



IV Semana Nacional de Ingeniería Electrónica

Memorias del Congreso

Universidad Panamericana-Bonatererra

**1 al 3 de Octubre de 2008
Aguascalientes, Ags. México**

ISBN 978-970-31-0944-9

Editor

Universidad Autónoma Metropolitana-Azcapotzalco
Av. San Pablo 180 Col. Reynosa Tamaulipas
Del. Azcapotzalco, 02200 México D.F.

Coeditor

Universidad Panamericana-Bonaterza
Josemaría Escriba de Balaguer 101
Fracc. Rústicos Calpulli
20290 Aguascalientes, Ags.

Autor

Varios

Compiladores

Miguel Magos Rivera
Ramiro Velázquez Guerrero

IV Semana Nacional de Ingeniería Electrónica

Memorias del Congreso

ISBN 978-970-31-0944-9

México D.F. 2008

Comité Organizador

M. en C. Eduardo de la Vega Segura
Director Escuela de Ingeniería
Universidad Panamericana-Aguascalientes

M. en C. José Raymundo Lira Cortés
Jefe del Departamento de Electrónica
UAM-Azcapotzalco

Dr. Miguel Magos Rivera
Presidente

Dr. Ramiro Velázquez Guerrero
Coordinador Local

M. en C. Ricardo Macías Quijas
Coordinador Logístico

Ing. Ricardo Godínez Bravo
Coordinador Exposición Industrial

Comité Técnico

Dr. Aviles Cruz Carlos	UAM-Azcapotzalco
Ing. Abundes Gallegos Oscar	Automation Solutions Factory S.A. de C.V.
Dr. Adam Medina Manuel	CENIDET-Cuernavaca
Dra. Alcaraz Mejía Mildreth Isadora	Centro Universitario de la Ciénega-UDG
Dra. Abascal Mena Ma. del Rocío	UAM-Cuajimalpa
Dr. Astorga Zaragoza Carlos	CENIDET-Cuernavaca
Dr. Campos Rodríguez Raúl	Centro Universitario de la Ciénega-UDG
Dr. Durán Gómez José Luís	Instituto Tecnológico de Chihuahua
Dr. Ferreyra Ramírez Andrés	UAM-Azcapotzalco
Dr. Herrera Alcántara Oscar	UAM-Azcapotzalco
Dr. Leyva Hernández Gerardo	Universidad Autónoma de Aguascalientes
Dr. López Mellado Ernesto	Cinvestav-Guadalajara
Dr. López Ornelas Erick	UAM-Cuajimalpa
Dr. López Orozco Omar	ITESM-Morelia
Ing. Meré Broissin Alfredo	Mega Control y Suministros, S. A de C.V.
Dr. Mota Grajales Rafael	Instituto Tecnológico de Tuxtla Gutiérrez
Dr. Magos Rivera Miguel	UAM-Azcapotzalco
M. en C. Macías Quijas Ricardo	Universidad Panamericana-Aguascalientes
Ing. Olvera Menchaca Ramón	Automation Solutions Factory S.A. de C.V.
Dr. Orozco Mora Jorge	Instituto Tecnológico de Aguascalientes
Dr. Ramírez Treviño Antonio	Cinvestav-Guadalajara
Dr. Ramos Corchado Félix	Cinvestav-Guadalajara
Dra. Ruiz Beltrán Elvia	Instituto Tecnológico de Aguascalientes
Dr. Velázquez Guerrero Ramiro	Universidad Panamericana-Bonaterza
Dr. Varona Salazar Jorge	Snowbush-Aguascalientes
Dr. Zavala Río Arturo	Instituto Potosino de Investigación Científica y Tecnológica

Presentación

Con la realización de la IV Semana Nacional de Ingeniería Electrónica, la División de Ciencias Básicas e Ingeniería de la Universidad Autónoma Metropolitana Azcapotzalco y la Escuela de Ingeniería de la Universidad Panamericana Bonaterra cumplen con el objetivo de proporcionar un foro para la discusión de ideas, la comparación e intercambio de resultados y la divulgación de los trabajos de investigación relacionados con la Ingeniería Electrónica y sus diversas ramas.

Como en las ediciones anteriores, a la par de las presentaciones de los trabajos, se llevan a cabo Conferencias Magistrales y Técnicas sobre temas de interés. Asimismo, se realiza una exposición industrial la cual cuenta con la presencia de empresas líderes en su ramo.

En esta ocasión el evento se lleva a cabo en las instalaciones de la Universidad Panamericana Bonaterra, institución privada de reconocido prestigio ubicada en el centro del país.

Las memorias incluyen 67 artículos en extenso, los cuales fueron seleccionados por el Comité Técnico de entre casi 100 trabajos enviados por autores de todo el país. Cabe señalar que se realizó un esfuerzo importante en el arbitraje de los trabajos con el objetivo de asegurar la calidad de los mismos. Nuestro reconocimiento a los miembros del Comité Técnico por la labor realizada.

Finalmente, agradecemos a los miembros del Comité Organizador Local por su esfuerzo para lograr que este evento se realice en las mejores condiciones.

Aguascalientes, Ags., octubre de 2008.

Comité Organizador de la IV Semana Nacional de Ingeniería Electrónica.

Miércoles 1 de Octubre

Horario	Sala Calvillo	Sala Rincón de Romos	Sala San José de Gracia
9:00 – 10:00		Inscripción	
10:00 – 10:30		Inauguración	
10:30 – 11:30		Magistral 1 <u>Auditorio IPADE</u> Recolección y conversión de energía en la microescala <i>M. en C. Jorge Varona Salazar</i> Snowbush Microelectronics de México	
11:30 – 11:50		Receso	
11:50 – 12:20		Conferencia Técnica 1 <u>Sala Calvillo</u> Agrupamientos industriales en Aguascalientes: Caso CELESA <i>M. en C. Alejandro López López</i> Cluster de Electrónica de Aguascalientes	
12:20 – 12:25		Receso	
12:25 – 12:50	<i>Ley Federal de Telecomunicaciones y pérdida de la órbita geostacionaria 109,2° oeste</i> Cintya Maigre Pérez Instituto Politécnico Nacional	<i>Implementación en FPGA de un algoritmo novedoso para detección en línea de burras rotas en motores de inducción</i> Ismael Urbina Salas Universidad de Guanajuato	<i>Control difuso de temperatura implementado en un CPLD</i> Rubén Herrera Galicia Instituto Tecnológico de Tuxtla Gutiérrez
12:50 – 13:15	<i>Diseño de un receptor para procedimiento genérico de entramado (GFP) para redes de transporte ópticas</i> Ricardo Macías Quijas Universidad Panamericana-Bonaterra	<i>Cálculo del rendimiento de la técnica de control por ángulo simétrico para un rectificador controlado monofásico</i> José Ángel Lerma García Instituto Tecnológico de Veracruz	<i>Caracterización de una red neuronal difusa evolutiva y su aplicación en problemas de predicción</i> Marco Antonio Sánchez Trejo Universidad Autónoma Metropolitana-Azcapotzalco
13:15 – 13:40	<i>Bluetooth: desempeño en redes inalámbricas de área personal</i> José Antonio Michel Macarty Universidad Autónoma del Estado de Baja California	<i>Instalación de filtros de armónicas en la Comisión Federal de Electricidad, División de Distribución Centro Sur, para disminuir el efecto de las armónicas</i> Antonio Sánchez Isais Comisión Federal de Electricidad	<i>Propuesta para predecir eventos en el sistema de monitoreo volcánico en tiempo real basado en la informática bio-inspirada y algoritmos de redes neuronales</i> Luis Enrique Colmenares Guillén Benemérita Universidad Autónoma de Puebla
13:40 – 14:05	<i>Sistema mejorado de telemetría de bajo costo para aplicaciones agrícolas</i> Daniel Ortiz Barrero Universidad Autónoma de Baja California	<i>Cargador de baterías para aplicaciones fotovoltaicas con seguidor de la corriente máxima de carga</i> Adolfo Rafael López Núñez Instituto Tecnológico de Celaya	<i>Sistema de detección de fallas en conectores para amplificadores de RF</i> Ruth Esther Ledesma Angulo Universidad Autónoma de Baja California-Tijuana
14:05 – 15:30		COMIDA	
15:30 – 16:00		Conferencia Técnica 2 <u>Sala Calvillo</u> Aplicaciones con el microcontrolador AVR <i>Ing. Pavel Elíosa Jara</i> Haltica Automatización	
16:00 – 16:15		Receso	
16:15 – 16:40	<i>Implementación de un termómetro digital con interface RS232 con el microcontrolador MC9RS08KA2</i> Adalberto Bustamante López Centro de Enseñanza Técnica y Superior- Tijuana	<i>Diseño físico de CI's en procesos nanométricos</i> Emilio Saenz Irigoyen Universidad Panamericana-Bonaterra	<i>Análisis modal y estructural de microscopios de anillo vibrante mediante modelos de elementos finitos</i> Agustín Leobardo Herrera May Universidad Veracruzana
16:40 – 17:05	<i>Sistema de monitoreo por radiofrecuencia mediante el uso de una PDA</i> Gabriel Elías Aranda Castillo Universidad Autónoma de San Luis Potosí	<i>Construcción de un microcontrolador para aplicaciones empotradas en FPGAs, a partir del núcleo Picoblaze</i> Felipe Santiago Espinosa Universidad Tecnológica de la Mixteca	<i>Espectrómetro de voz usando la tecnología MEMS</i> Edgar Julian Cabrera Magaña / Rubén Cabrera Magallón Universidad Autónoma de Guadalajara
17:05 – 17:30	<i>Diseño y aplicación de un medidor de frecuencia "frecuencímetro" sobre FPGA's</i> Juan Sifuentes Mijares Instituto Tecnológico de la Laguna	<i>Módulos aritméticos CMOS VLSI para sistemas multimedia</i> Mónico Linares Aranda Instituto Nacional de Astrofísica, Óptica y Electrónica	<i>Determinación de insecticia en pimiento morrón mediante el uso de tecnologías MEMS</i> Juan Hernández Vela Instituto Tecnológico Superior de Irapuato

Jueves 2 de Octubre

Horario	Sala Calvillo	Sala Rincón de Romos	Sala San José de Gracia
9:00 – 9:25	Consideraciones de diseño de antenas reconfigurables MEMS para uso en sistemas de comunicación miniaturizados Miguel Ángel Sosa Torres Instituto Tecnológico Superior de Irapuato	Un nuevo concepto de interfaz táctil basado en estimulación latero-táctil María Lilia Martínez Rico Universidad Panamericana-Bonaterra	SCAM: Sistema de Control de Acceso al Macrocentro de Cómputo Elvia E. Aispuro Félix Universidad Autónoma de Baja California Sur
9:25 – 9:50	Circuito interfaz integrado para registro simultáneo de un microsistema MEMS Neutona-Chip Francisco López Huerta Benemérita Universidad Autónoma de Puebla	Diseño y construcción de un sistema automático para la perforación de circuito impreso Juan Carlos Pérez Merlos Universidad Autónoma del Estado de México	Replicas de información utilizando agentes móviles Israel Guzmán Pérez Benemérita Universidad Autónoma de Puebla
9:50 – 10:15	Diseño parametrizado de microestructuras tipo engrane Raúl Campos Rodríguez Universidad de Guadalajara	Control en tiempo real de un manipulador flexible de dos grados de libertad usando el microcontrolador Freescale MC9S12 y Matlab Rogelio Francisco Antonio Universidad Politécnica del Valle de México	
10:15 – 10:25		Receso	
10:25 – 11:25		Magistral 2 <u>Auditorio IPADE</u> Dr. Carlos Delgado Mata Universidad Panamericana Bonaterra	
11:25 – 11:35		Receso	
11:35 – 12:05		Conferencia Técnica 3 A confirmar	
12:05 – 12:35		Conferencia Técnica 4 A confirmar	
12:35 – 12:55		Receso	
12:55 – 13:20	Model synthesis of concurrent discrete event systems using interpreted Petri nets María Elena Meda Campana Universidad de Guadalajara	A constraint-based stochastic method to address school course timetabling problem Hugo Iván Piza Dávila Universidad de Guadalajara	Proceso de fabricación de microestructuras para aplicaciones en MEMS utilizando micromaqueinado superficial José Andrés Alanís Navarro Instituto Nacional de Astrofísica Óptica y Electrónica
13:20 – 13:45	Diseño de una herramienta de simulación de sistemas industriales basada en Redes de Petri Alejandra Santoyo Sanchez Universidad de Guadalajara	Sistema de identificación de rostros humanos a través de redes neuronales Marco Antonio Sánchez Trejo Universidad Autónoma Metropolitana-Azcapotzalco	Acondicionamiento de señal para un sensor de presión MEMS Rodrigo Montiel Uribe Instituto Tecnológico Superior de Irapuato
13:45 – 14:10	Diagnóstico de faltas de control en RPI Lizette Rubio Gómez Cinvestav-Guadalajara	EncuentraCel – sistema de localización y búsqueda de sitios de interés para Web y teléfonos celulares Luis Enrique Bustos Méndez Instituto Politécnico Nacional	
14:10 – 15:30		COMIDA	

Viernes 3 de Octubre

Horario	Sala Calvillo	Sala Rincón de Romos	Sala San José de Gracia
9:00 – 9:25	Análisis de la técnica de aprendizaje basada en imagen José Adriel Alonso Castro Centro de Investigación en Computación-IPN	Diseño de un electrocardiógrafo digital usando como plataforma el kit de desarrollo Spartan-3E de Xilinx Celia del Carmen Martínez Chifñas Instituto Tecnológico de Tuxtla Gutiérrez	Diagnóstico de fallas usando redes de Petri interpretadas: <i>Aproximación estructural</i> Alejandra Santoyo Sánchez Universidad de Guadalajara
9:25 – 9:50	Manejo de eventos compuestos en sistemas RFID Omar Alfredo González Padilla Cinvestav-Guadalajara	Sistema de control para una cama vibratoria para pruebas oscilatorias Andrés Gerardo Fuentes Covarrubias Universidad de Colima	<i>Methods for obtaining reduced Petri net models to detect and locate faults</i> Elvia Ruiz Beltrán Instituto Tecnológico de Aguascalientes
9:50 – 10:15	Diseño e implementación de un detector de complejos QRS en la línea del ECG Raúl Campos Rodríguez Universidad de Guadalajara	Sistema auxiliar para el diagnóstico del carcinoma epidermoide invasor en tejido cervix empleando tecnología láser Aurora Aparicio Castillo ESIME-Azcapotzalco	<i>Fault diagnosis in discrete event systems using distributed diagnosers</i> Jesús Arámburo Lizárraga Cinvestav-Guadalajara
10:15 – 10:25		Receso	
10:25 – 11:25		Magistral 3 Auditorio IPADE Control de sistemas de eventos discretos	
11:25 – 11:35		Dr. Antonio Ramírez Treviño Cinvestav Guadalajara	
11:35 – 12:05		Receso	
12:05 – 12:35		Conferencia Técnica 5 Sala Calvillo Técnicas de simulación de transporte para evitar daños al producto durante su distribución Ing. Javier Ochoa Flextronics	
12:05 – 12:35		Conferencia Técnica 6 Sala Calvillo laboratorios de metrología con sistemas de gestión de calidad ISO17025 Ing. Oscar Narciso Hernández Tovar Flextronics	
12:35 – 12:55		Receso	
12:55 – 13:20	<i>The two ray technique applied to the RFID technology</i> Salvador Ricardo Meneses González ESIME-Zacatenco Instituto Politécnico Nacional	<i>Formación de robots basada en interacciones locales</i> Gustavo A. Torres Blanco Cinvestav-Guadalajara	<i>Cálculo distribuido del grafo de alcanzabilidad de una Red de Petri</i> Mildreth Isadora Alcaraz Mejía Universidad de Guadalajara
13:20 – 13:45	<i>Redes de radios cognitivos. Un estudio exploratorio</i> Jesús Zukhov Sánchez Garrido Instituto Tecnológico Autónomo de México	<i>Animation of autonomous avatars over the GeDA-3D agent architecture</i> Héctor Rafael Orozco Aguirre Cinvestav-Guadalajara	<i>Aplicación de redes de Petri para el diseño de control lógico</i> Bernardo Haro Martínez Universidad Autónoma De Guadalajara
13:45 – 14:10	<i>Análisis paramétrico del modelo de capa acoplada perfectamente en el método de diferencias finitas en el dominio del tiempo</i> Alberto Manuel Benavides Cruz Instituto Politécnico Nacional	<i>E-entertainment: Sistema Web de comercio electrónico móvil para eventos culturales y de entretenimiento</i> Julio Jesús Florido Paredes UPIITA- Instituto Politécnico Nacional	<i>Lenguaje para la descripción de la información geográfica de sistemas de tráfico urbano</i> Emmanuel López Neri Cinvestav-Guadalajara
14:10 – 15:45		COMIDA	
15:45 – 16:10	<i>Notas acerca del problema del péndulo invertido</i> Enrique Preza Hernández Universidad Panamericana-Bonaterra	<i>Control GPI de un motor sincrónico de imanes permanentes</i> Jesús Antonio Mendieta Antúnez Universidad Autónoma Metropolitana-Azcapotzalco	<i>Composición de protocolos de interacción utilizando obligaciones</i> José Octavio Gutiérrez García Cinvestav-Guadalajara
16:10 – 16:35	<i>Diseño de un sistema de instrumentación para un sistema de generación de fuerza en un simulador de cadera mecánico</i> Jorge Antonio Rangel Carrillo Universidad Tecnológica de Torreón	<i>Instrumento de bajo costo basado en FPGA para el monitoreo de la calidad de la energía eléctrica</i> Jorge Lara Cardoso Universidad de Guanajuato	<i>Modelado declarativo de ambientes virtuales basado en explotación del conocimiento</i> Jaime Alberto Zaragoza Ríos Cinvestav-Guadalajara
16:35 – 17:00	<i>Diseño de un vehículo eléctrico computarizado controlado a distancia mediante el estándar WIFI @ 802.11</i> Francisco Javier Díaz Delgado Centro de Enseñanza Técnica y Superior-Tijuana	<i>Cálculo del rendimiento de la técnica PWM de múltiples pulsos para un rectificador controlado monofásico</i> Mario Cruz Cruz Instituto Tecnológico de Veracruz	<i>Generation of behavior's agents using evolutive systems and anticipation</i> Marco Antonio Ramos Corchado Universidad Autónoma del Estado de México
17:00 – 17:25	<i>Robot de posicionamiento angular de tres grados de libertad controlado mediante cámara de video</i> Héctor Demián Vargas Mejía Universidad Panamericana-Bonaterra	<i>Controlador de cargas eléctricas con el protocolo MODBUS para el control de la demanda máxima</i> Fernán Hugo Ramírez Leyva Universidad Tecnológica de la Mixteca	<i>Procesamiento digital de video en tiempo real y "video-wall" con PC</i> Carlos Avilés Cruz Universidad Autónoma Metropolitana-Azcapotzalco
17:25 – 17:35		Receso	
17:35 – 18:30		Clausura y Brindis	

INDICE

Clic el Título de la ponencia para llevarlo a la información

Título de la Ponencia	Página
Control difuso de temperatura implementado en un CPLD	1
The two rays technique applied to the RFID technology	10
Redes de radios cognitivos. Un estudio exploratorio	16
Análisis paramétrico del modelo de capa acoplada perfectamente en el método de diferencias finitas en el dominio del tiempo	25
Ley Federal de Telecomunicaciones y perdida de la orbita geoestacionaria 109.2° oeste	32
Diseño de un receptor para procedimiento genérico de entramado (GFP) para redes de transporte ópticas	39
Bluetooth: desempeño en redes inalámbricas de área personal	47
Sistema mejorado de telemetría de bajo costo para aplicaciones agrícolas	55
Diseño físico de CI's en procesos nanométricos	63
Construcción de un microcontrolador para aplicaciones empotradas en FPGAs, a partir del núcleo Picoblaze.	74
Módulos aritméticos CMOS VLSI para sistemas multimedia	84
Implementación en FPGA de un algoritmo novedoso para detección en línea de barras rotas en motores de inducción	95
Instrumento de bajo costo basado en FPGA para el monitoreo de la calidad de la energía eléctrica	102
Instalación de filtros de armónicas en la Comisión Federal de Electricidad, División de Distribución Centro Sur, para disminuir el efecto de las armonicas	109
Control GPI de un motor síncrono de imanes permanentes	116
Cargador de baterías para aplicaciones fotovoltaicas con seguidor de la corriente máxima de carga	122
Cálculo del rendimiento de la técnica de control por ángulo simétrico para un rectificador controlado monofásico	131
Cálculo del rendimiento de la técnica PWM de múltiples pulsos para un rectificador controlado monofásico	139
Implementación de un termómetro digital con interface RS232 con el microcontrolador MC9RS08KA2	147
Controlador de cargas eléctricas con el protocolo MODBUS para el control de la demanda máxima	156
Sistema de monitoreo por radiofrecuencia mediante el uso de una PDA	166
Diseño de un electrocardiógrafo digital usando como plataforma el kit de desarrollo Spartan-3E de Xilinx	174
Propuesta para predecir eventos en el sistema de monitoreo volcánico en tiempo real basado en la informática bio-inspirada y algoritmos de redes neuronales	181
Sistema de control para una cama vibratoria para pruebas oscilatorias	191
Sistema auxiliar para el diagnostico del carcinoma epidermoide invasor en tejido cervix empleando tecnología láser	198
Diseño y aplicación de un medidor de frecuencia “frecuencímetro” sobre FPGA’s	208
Un nuevo concepto de interfaz táctil basado en estimulación latero-táctil	215
Notas acerca del problema del péndulo invertido	224
Control en tiempo real de un manipulador flexible de dos grados de libertad usando el microcontrolador freescale mc9s12 y matlab	232
Diseño y construcción de un sistema automático para la perforación de tablilla de circuito impreso.	239

Título de la Ponencia	Página
Diseño de un sistema de instrumentación para un sistema de generación de fuerza en un simulador de cadera mecánico	245
Diseño de un vehículo eléctrico computarizado controlado a distancia mediante el estándar WIFI @ 802.11	253
Robot de posicionamiento angular de tres grados de libertad controlado mediante cámara de vídeo	263
Análisis modal y estructural de microgiroscopios de anillo vibrante mediante modelos de elementos finitos	269
Espectrómetro de voz usando la tecnología MEMS	276
Determinación de insecticida en pimiento morón mediante el uso de tecnologías MEMS	282
Proceso de fabricación de microestructuras para aplicaciones en MEMS utilizando micromaquinado superficial	290
Acondicionamiento de señal para un sensor de presión MEMS	297
Consideraciones de diseño de antenas reconfigurables MEMS para uso en sistemas de comunicación miniaturizados	307
Circuito interfaz integrado para registro simultáneo de un microsistema MEMS Neurona-Chip	313
Diseño parametrizado de microestructuras tipo engrane	322
Sistema de detección de fallas en conectores para amplificadores de RF	328
Análisis de la técnica de aprendizaje basada en imagen	335
Sistema de identificación de rostros humanos a través de redes neuronales	343
Diseño e implementación de un detector de complejos QRS en la línea del ECG	353
Caracterización de una red neuronal difusa evolutiva y su aplicación en problemas de predicción	362
Procesamiento digital de video en tiempo real y “video-wall” con PC	373
Animation of autonomous avatars over the GEDA-3D agent architecture	387
E-ntertainment: Sistema Web de comercio electrónico móvil para eventos culturales y de entretenimiento	397
Composición de protocolos de interacción utilizando obligaciones	405
Modelado declarativo de ambientes virtuales basado en explotación del conocimiento	412
Generation of behavior’s agents using evolutive systems and anticipation	419
A constraint-based stochastic method to address school course timetabling problem	425
Manejo de eventos compuestos en sistemas RFID	432
EncuentraCel – sistema de localización y búsqueda de sitios de interés para Web y teléfonos celulares.	441
Formación de robots basada en interacciones locales	452
Lenguaje para la descripción de la información geográfica de sistemas de tráfico urbano	459
SCAM: Sistema de Control de Acceso al Macrocentro de Cómputo	468
Replicas de información utilizando agentes móviles	476
Aplicación de redes de Petri para el diseño de control lógico	483
Diseño de una herramienta de simulación de sistemas industriales basada en Redes de Petri	492
Diagnóstico de faltas de control en RPI	499
Cálculo distribuido del grafo de alcanzabilidad de una Red de Petri	506
Model synthesis of concurrent discrete event systems using interpreted Petri Nets	513
Diagnóstico de fallas usando redes de Petri interpretadas: Aproximación estructural	523
Methodologies for obtaining reduced Petri net models to detect and locate faults	533
Fault diagnosis in discrete event systems using distributed diagnosers	540

Control Difuso De Temperatura Implementado En Un CPLD

Herrera Galicia Rubén, Hipólito Gordillo José Alberto.

Área de Control y Automatización, Departamento de Eléctrica y Electrónica,
Instituto Tecnológico de Tuxtla Gutiérrez
Carretera Panamericana Km. 1080, Col. Teran, Tuxtla Gutiérrez, Chiapas, 29070, México
Tel. 6150380, correo electrónico: h_galicia24@hotmail.com, alberto_hipolito@hotmail.com

RESUMEN

El principal problema que se presenta cuando se diseña un control difuso en un microcontrolador son las limitaciones de software, las cuales están condicionadas por el tipo de datos que se manejan, enteros de 0 a 255, y por el conjunto de instrucciones, que hace que las tareas de programación sean más laboriosas en comparación con los lenguajes de mediano nivel. Al implementar un control difuso lo que se hace es: programar la fusificación a través de comparaciones y la implementación de las ecuaciones de las funciones de membresía, programar el conjunto de reglas a través de comparaciones de los valores de pertenencia para definir el valor difuso correspondiente a las variables de salida, y programar una suma de productos para la defusificación.

El problema que aquí se resuelve es implementar un control difuso con un mínimo de recursos de software y principalmente sin la necesidad de programar la fusificación, el conjunto de reglas y la defusificación. El procedimiento se basa en la hipótesis de que la programación de la fusificación, el conjunto de reglas y la defusificación puede ser sustituida por un análisis (incluido en el presente trabajo), el cual permite, a través de simplificaciones, obtener una fórmula de diseño sencilla, la cual representa por completo al control difuso. La fórmula de diseño tiene la característica de simplificar significativamente los cálculos necesarios para la

implementación del control difuso, lo cual reduce el nivel de dificultad de la programación.

Desde el punto de vista del dispositivo usado, es interesante implementar un control difuso en un CPLD por la posibilidad de procesamiento paralelo que ofrece este componente y por el procesamiento en un tiempo nulo, lo cual da una velocidad de respuesta limitada solo por la frecuencia del reloj y por la propagación de la señal a través de las macro celdas.

Palabras clave: Control, Lógica Difusa, Microcontroladores, CPLD.

I. Introducción

Los microcontroladores y los CPLD presentan diferentes características con las que pueden competir entre si, pero vale la pena considerar la posibilidad que tienen de complementarse para formar un dúo de excelentes cualidades. Los microcontroladores resuelven el problema de la conversión ADC y el manejo de LCD de manera eficiente. Los CPLD ofrecen la posibilidad de procesamiento concurrente a la par con procesamiento secuencial rápido y confiable. Un enfoque interesante es diseñar un modulo donde se repartan las tareas entre el CPLD y el microcontrolador de tal manera que el funcionamiento del sistema sea mas confiable y eficiente.

En este trabajo se presenta un sistema de control de temperatura basado en un microcontrolador y un CPLD. El microcontrolador tiene asignadas

como tareas: convertir de analógico a digital la señal proveniente de un sensor analógico de temperatura y presentar la temperatura en un LCD. En el CPLD se implementó el control difuso, encargado de determinar la demanda de potencia, además de generar la señal PWM encargada de regular la potencia entregada a la carga.

La principal ventaja que ofrece el sistema microcontrolador- CPLD es la diversidad de posibilidades que ofrece en la asignación de tareas incluyendo la posibilidad de procesamiento paralelo inherente a los dispositivos CPLD.

II. Dispositivos CPLD

1) *Arquitectura.* Un dispositivo lógico programable complejo CPLD es un dispositivo cuyas características pueden ser modificadas y almacenadas mediante programación. El CPLD es un dispositivo con complejidad entre el FPGA y las PAL, y con características arquitectónicas de ambos. El principio de funcionamiento de un dispositivo lógico programable se fundamenta en el hecho de que cualquier función booleana puede ser expresada como una suma de productos. El bloque principal del CPLD son las macroceldas, las cuales contienen arreglos específicos de compuertas lógicas. Las macroceldas se comunican entre sí utilizando una matriz programable de interconexiones. Al programar en VHDL en realidad lo que se está haciendo es programar a la matriz de interconexiones.

2) *Aplicaciones.* Los CPLD implementan más eficientemente diseños con parte combinacional abundante y dispositivos de configuración no volátil. Los FPGA son usados para diseños que manejan mayor transferencia de datos y registros, pudiéndose implementar memorias RAM, ROM, y FIFO. Las aplicaciones comunes de los CPLD incluyen control de motores DC por modulación de anchura de pulso PWM, control ADC/DAC, contador de eventos, divisor de frecuencia, generador de interrupción, máquinas de estado, teclados, interfaces de panel de control, comandos de habilitación para dispositivos externos, multiplexores y circuitos de retardo.

3) *Ventajas.* El uso de dispositivos CPLD trae consigo como ventajas: velocidad superior a la obtenida con un microcontrolador y mayor confiabilidad. Los microcontroladores y los CPLD ambos tienen como ventajas: ahorro de espacio, solución en una sola pastilla, bajos requerimientos de potencia, facilidad en la corrección de errores por ser reprogramables, proceso de ensamble rápido y a bajo costo.

4) *Desventajas.* El uso de dispositivos CPLD trae consigo como desventajas: el tipo de encapsulado, la mayoría son de montaje superficial, y las herramientas de desarrollo aun son deficientes, sobre todo las de programación.

5) *Programación.* El lenguaje usado para programar a un CPLD es el lenguaje de descripción de hardware VHDL. Las descripciones en VHDL son creadas a partir de dos estructuras: la entidad y la arquitectura. La entidad representa al circuito. Se piensa en ella como en una caja negra y solo se definen entradas y salidas. En la arquitectura se define el comportamiento interno de la entidad con instrucciones que se ejecutan de manera secuencial o en concurrencia. Una estructura programada en concurrencia siempre está activa y se llama proceso. Por lo que todo lo que se incluya dentro de un proceso se ejecutará al mismo tiempo que el resto de las instrucciones que están afuera.

III. Control Difuso

Las técnicas de razonamiento aproximado pueden ser aplicadas a diferentes niveles de control. Los sistemas de control difuso son una extensión de los sistemas expertos, pero superando los problemas prácticos que estos tienen para el razonamiento en tiempo real, causados por el aumento de las operaciones de cálculo indispensables para el análisis completo de amplias bases de reglas.

La lógica difusa es aplicable a todos aquellos sistemas que se pueden describir lingüísticamente y son una gran mejora en los sistemas no lineales, difíciles de describir matemáticamente. Un control difuso se divide en tres etapas; la

fusificación, la evaluación de reglas y la defusificación.

1). *La fusificación.* Esta etapa consiste en asociar un valor real, del universo de discurso de la variable lingüística, hacia sus correspondientes valores difusos. Estos son números reales en el intervalo cerrado $[0,1]$, definidos por las funciones de membresía de los conjuntos difusos de la variable de entrada. A cada valor real se le asocia un valor difuso por cada conjunto difuso de entrada. Dicho valor difuso representa el grado de pertenencia de un valor real dado a un conjunto difuso dado.

2). *La evaluación de reglas.* Esta etapa consiste en usar los valores difusos obtenidos en la fusificación para inferir (asignar) grados de pertenencia a los conjuntos difusos de la variable de salida. La inferencia incluye implicaciones y operaciones de lógica difusa; conjunción y disjunción.

3). *La defusificación.* Esta etapa es el proceso inverso de la fusificación y consiste en combinar a todos los valores difusos de la variable de salida para producir un valor único real. Este valor representa la respuesta del sistema a un conjunto de valores de entrada.

La literatura sobre aplicaciones de lógica difusa es amplia y los problemas que se resuelven con ella son de diversa índole, por eso el procedimiento que aquí se propone no puede ser aplicado a todos ellos. El procedimiento aquí descrito es aplicado al control de un parámetro físico, por ejemplo: la temperatura del aire, la temperatura de un líquido, o la humedad del aire. Existen múltiples aplicaciones donde las simplificaciones que aquí se hacen no son favorables, por la naturaleza misma de esos problemas, por ejemplo: control de un metro, ajuste de color en una TV, análisis de contaminación del agua, reconocimiento de escritura, planificación del tiempo de salida de autobuses, predicción de terremotos, diagnóstico de cáncer a partir de imágenes, clasificación de arritmias cardíacas, control de robots industriales, posicionamiento de helicóptero sin piloto [2- 6].

IV. Método De Diseño

El procedimiento para el diseño del control difuso fue publicado con anterioridad en [1]. Una descripción actualizada de dicho método se presenta aquí. El método consiste de ocho etapas y puede aplicarse por ejemplo a control de temperatura ó humedad del aire.

Primera etapa.- Definir las entradas que garanticen la mejor respuesta del sistema. Esto se hace experimentalmente por prueba y error, o utilizando el conocimiento del diseñador. En la literatura se usan dos tipos de variables de entrada para controles de temperatura; $[T_{ref}, Error]$ y $[Error, \Delta Error]$. Aquí se usara el primero de estos y se proponen como entradas el valor de referencia del parámetro a controlar X_{ref} y el Error. Aquí también se define la variable de salida, que generalmente es potencia entregada a una carga Pot. En la Fig.1 se presentan las entradas seleccionadas y la variable de salida. Las entradas son el valor de referencia del parámetro a controlar X_{ref} y el Error. X_{med} es el valor medido del parámetro de interés. Para calcular el Error se usa el valor de referencia y el valor medido; $Error = X_{ref} - X_{med}$.

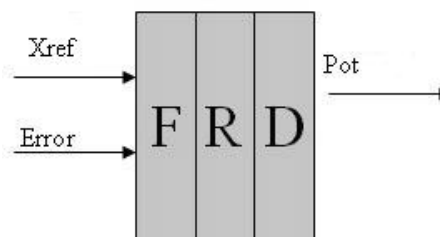


Fig.1 Sistema de Control Difuso.

Segunda etapa.- Proponer los conjuntos difusos para las variables de entrada y la variable de salida. Generalmente son suficientes tres conjuntos difusos para las variables de entrada. Aquí para X_{ref} se seleccionan: BAJA, MEDIA, ALTA, relacionados con el conjunto de etiquetas {BAJ, MED, ALT}. Y para el Error se seleccionan: POSITIVO, CERO, NEGATIVO con etiquetas {POS, CER, NEG}. Para la variable de salida generalmente son suficientes de tres a cinco conjuntos difusos. Aquí para la salida se

seleccionan cuatro: NULA, BAJA, MEDIA, ALTA y se relacionaran con el conjunto de etiquetas {NUL, BAJ, MED, ALT}.

Tercera etapa.- Dibujar las funciones de membresía y seleccionar el universo de discurso; es decir determinar la forma de la gráfica de la función de membresía de cada conjunto difuso del sistema. Las formas más comunes y suficientes son como las presentadas en la Fig.2. En el método aquí descrito [1] se propone hacer que el valor deseado de X_{ref} coincida con el punto máximo del conjunto media. Con esta consideración los grados de pertenencia de X_{ref} a los conjuntos difusos son: $MED=1$, $BAJ=0$, $ALT=0$. Esta simplificación no afecta al control difuso y tiene la ventaja, como se verá mas adelante, de que permite hacer una evaluación de las reglas sin tener que conocer los valores de las variables X_{ref} y Error. También reduce el conjunto de reglas a tres. Y además permite obtener como resultado final una sola ecuación, la cual simplifica significativamente los cálculos y el diseño.

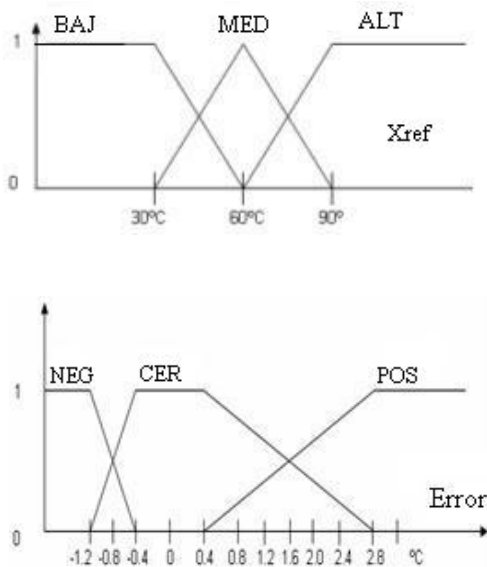


Fig.2 Funciones de Membresía para los conjuntos difusos de entrada.

En la Fig.2 se presentan las funciones de membresía para la variable X_{ref} . Si el valor deseado del parámetro a controlar X_{ref} , se hace coincidir con el valor máximo del conjunto difuso

MED , esto es $X_{ref}=60$, entonces se simplifican los cálculos. También en la Fig.2 se presentan las funciones de membresía para la variable Error. Aquí para la variable Error se considera como suficiente un universo de discurso de 4, pero vale la pena recordar que el diseño de un control difuso se basa en prueba y error. Por tanto en caso de obtener resultados deficientes, el universo de discurso deberá ajustarse empíricamente.

Cuarta etapa.- Construir una matriz de inferencia difusa en base al conocimiento del experto humano y usar esta matriz para construir el conjunto de reglas. Cuando se trabaja con control difuso comúnmente se habla de un experto humano, pero la aplicación de lógica difusa tiene la ventaja de que en muchos casos es suficiente un análisis del comportamiento esperado del sistema para construir esta matriz o construirla por semejanza a las matrices de inferencia presentadas en la literatura. Además, para facilitar su construcción, vale la pena observar que las matrices de inferencia se caracterizan por estar constituidas por curvas de nivel. Para controlar la temperatura o la humedad del aire es suficiente un análisis del comportamiento esperado del sistema para construir la matriz de inferencia y cumplir con este requisito.

Error	NEG	CER	POS
X_{ref}			
BAJ	<i>NUL</i>	<i>NUL</i>	<i>MED</i>
MED	<i>NUL</i>	<i>BAJ</i>	<i>ALT</i>
ALT	<i>NUL</i>	<i>MED</i>	<i>ALT</i>

Fig.3 Matriz de Inferencia Difusa

En la Fig.3 se presenta un ejemplo de una matriz de inferencia difusa para las variables X_{ref} y Error. Esta matriz esta compuesta por un conjunto de nueve reglas. Las reglas se construyen según las combinaciones y la respuesta esperada a cada combinación. Por ejemplo, considérese la combinación: X_{ref} es bajo y Error es negativo. Aquí un Error negativo sucede cuando el valor medido sobre pasa al valor de referencia. Es

decir que no existe la necesidad de más potencia. Entonces esperamos que para estos casos la potencia sea ninguna, es decir nula. Esto da lugar a la regla 1. Y con un razonamiento similar se construyen el resto de las reglas basado en las distintas combinaciones y el valor de salida esperado.

Regla 1. SI {Xref es BAJ Y Error es NEG}, ENTONCES {Pot es NUL}

Quinta etapa.- Evaluar el conjunto de reglas. Cada regla es una implicación, la cual se compone de dos antecedentes conectados por un operador AND y un consecuente. Dada esta estructura, la evaluación de una regla consiste en comparar los grados de pertenencia de los antecedentes y tomar el menor de ellos para asignar este valor como grado de pertenencia al consecuente. Los grados de pertenencia son valores del intervalo cerrado [0, 1]. Además se considera que todas las reglas están conectadas por un operador OR. Dada esta consideración, para los conjuntos difusos de salida que se repitan se comparan sus grados de pertenencia, y se toma el mayor de estos para asignar este valor como grado de pertenencia único de dicho conjunto difuso de salida.

En la etapa tres se hizo coincidir el valor deseado de Xref con el valor máximo del conjunto Media, lo cual arrojo como grados de pertenencia de Xref a los siguientes: $\mu_{MED} = 1$, $\mu_{BAJ} = 0$, $\mu_{ALT} = 0$. Esta simplificación permite evaluar al conjunto de reglas sin conocer los grados de pertenencia de la variable Error y considerarlos como no importa. Evaluando cada regla se obtienen las igualdades 1- 9. En estas ecuaciones x es no importa y representa a cualquier valor del intervalo [0, 1].

- $\mu_{NUL1} = \text{MIN}\{\mu_{BAJ}, \mu_{NEG}\} = \text{MIN}\{0, x\} = 0$ (1)
- $\mu_{NUL2} = \text{MIN}\{\mu_{BAJ}, \mu_{CER}\} = \text{MIN}\{0, x\} = 0$ (2)
- $\mu_{NUL3} = \text{MIN}\{\mu_{MED}, \mu_{NEG}\} = \text{MIN}\{1, x\} = x$ (3)
- $\mu_{NUL4} = \text{MIN}\{\mu_{ALT}, \mu_{NEG}\} = \text{MIN}\{0, x\} = 0$ (4)
- $\mu_{BAJ} = \text{MIN}\{\mu_{MED}, \mu_{CER}\} = \text{MIN}\{1, x\} = x$ (5)
- $\mu_{MED1} = \text{MIN}\{\mu_{BAJ}, \mu_{POS}\} = \text{MIN}\{0, x\} = 0$ (6)
- $\mu_{MED2} = \text{MIN}\{\mu_{ALT}, \mu_{CER}\} = \text{MIN}\{0, x\} = 0$ (7)
- $\mu_{ALT1} = \text{MIN}\{\mu_{MED}, \mu_{POS}\} = \text{MIN}\{1, x\} = x$ (8)
- $\mu_{ALT2} = \text{MIN}\{\mu_{ALT}, \mu_{POS}\} = \text{MIN}\{0, x\} = 0$ (9)

Para el conjunto de salida NUL se tienen cuatro valores. De aquí se obtiene una igualdad representada por la ecuación (10).

$$\mu_{NUL} = \text{MAX}\{\mu_{NUL1}, \mu_{NUL2}, \mu_{NUL3}, \mu_{NUL4}\} = \text{MAX}\{0, 0, x, 0\} = x \quad (10)$$

Para el conjunto de salida BAJ se tiene solo un valor. Para el conjunto de salida MED se tienen dos valores. De aquí se obtiene una igualdad representada por la ecuación (11).

$$\mu_{MED} = \text{MAX}\{\mu_{MED1}, \mu_{MED2}\} = \text{MAX}\{0, 0\} = 0 \quad (11)$$

Para el conjunto de salida ALT se tienen dos valores. Y con esto se obtiene una igualdad representada por la ecuación (12).

$$\mu_{ALT} = \text{MAX}\{\mu_{ALT1}, \mu_{ALT2}\} = \text{MAX}\{x, 0\} = x \quad (12)$$

Hacer que el valor deseado Xref coincida con el punto máximo del conjunto media, y que los grados de pertenencia para los conjuntos baja y alta sean cero, simplificó los cálculos y el conjunto de reglas prácticamente se redujo a tres. El conjunto de reglas se redujo a las reglas de la segunda fila de la matriz de inferencia, que son las únicas reglas con salidas distintas de cero.

SI {Xref es MED Y Error es NEG}, ENTONCES {Pot es NUL}

SI {Xref es MED Y Error es CER}, ENTONCES {Pot es BAJ}

SI {Xref es MED Y Error es POS}, ENTONCES {Pot es ALT}

Error Xref	NEG x	CER x	POS x
BAJ 0	<i>NUL</i> 0	<i>NUL</i> 0	<i>MED</i> 0
MED 1	<i>NUL</i> x	<i>BAJ</i> x	<i>ALT</i> x
ALT 0	<i>NUL</i> 0	<i>MED</i> 0	<i>ALT</i> 0

Fig.4 Evaluación de reglas

En la Fig.4 se incluyen los grados de pertenencia. Si se considera que Xref tiene asociado un grado de pertenencia igual a 1, el grado de pertenencia

del consecuente siempre será el del Error, puesto que se elige el mínimo de los dos y 1 es el grado máximo. Una vez que se evalúen todas las reglas 1- 9, y considerando las igualdades 10- 12, cada conjunto de salida tendrá asociado un grado de pertenencia indicado por las igualdades 13- 16. Así los grados de pertenencia de los conjuntos difusos de la variable de salida quedan únicamente en función de los grados de pertenencia de uno de los conjuntos difusos de entrada, los de la variable Error.

$$\mu_{NUL} = \mu_{NEG} \quad (13)$$

$$\mu_{BAJ} = \mu_{CER} \quad (14)$$

$$\mu_{MED} = 0 \quad (15)$$

$$\mu_{ALT} = \mu_{POS} \quad (16)$$

Sexta etapa.- Definir y simplificar la fórmula para la defusificación. Un método que se adapta bien para fines de defusificación es el método del centroide [2, 3]. En este método cada grado de pertenencia se multiplica por su singleton, se suman todos estos productos y el resultado se divide entre la suma de los grados de pertenencia. La fórmula para la defusificación esta dada por la ecuación (17). Sustituyendo 13- 16 en 17 y haciendo cálculos se obtiene 18. En la Fig.2 se puede observar que siempre $\mu_{NEG} + \mu_{CER} + \mu_{POS} = 1$, con esta simplificación se obtiene la ecuación 19.

$$\text{Pot} = (\mu_{NUL} * \text{NUL} + \mu_{BAJ} * \text{BAJ} + \mu_{MED} * \text{MED} + \mu_{ALT} * \text{ALT}) / (\mu_{NUL} + \mu_{BAJ} + \mu_{MED} + \mu_{ALT}) \quad (17)$$

$$\text{Pot} = (\mu_{NEG} * \text{NUL} + \mu_{CER} * \text{BAJ} + 0 * \text{MED} + \mu_{POS} * \text{ALT}) / (\mu_{NUL} + \mu_{BAJ} + \mu_{MED} + \mu_{ALT}) \quad (18)$$

$$\text{Pot} = (\mu_{NEG} * \text{NUL} + \mu_{CER} * \text{BAJ} + \mu_{POS} * \text{ALT}) \quad (19)$$

Séptima etapa.- Seleccionar los singletons para los conjuntos difusos de la variable de salida. Un singleton es el peso específico que el experto le da a cada conjunto difuso de salida en el universo de discurso de la variable de salida. En cada control difuso el valor asignado a los singletons es muy importante y se ve reflejado en los resultados experimentales de control. Cambiando el valor de los singletons, a prueba y error, el diseñador tiene la posibilidad de lograr un control fino de manera fácil y rápida. Un singleton responde a la pregunta: ¿Qué tanto es baja?, ¿Qué tanto es media? Si al inicio no se tienen

preferencias, se puede iniciar con los siguientes valores; NUL= 0%, BAJ= 10%, MED= 50%, ALT= 100%. Si se considera un peso igual a cero para el singleton NUL se obtiene la ecuación 20.

$$\text{Pot} = (\mu_{CER} * \text{BAJ} + \mu_{POS} * \text{ALT}) \quad (20)$$

La ecuación 20 es la ecuación del control difuso. En esta fórmula se visualiza la importancia de los singletons BAJ y ALT. Dado que ALT esta multiplicando a μ_{POS} , este influye en la rapidez con la que el sistema llega al valor deseado Xref. Dado que BAJ esta multiplicando a μ_{CER} , este valor influye en el equilibrio del sistema una vez alcanzado el valor Xref. Es decir que el valor asignado a BAJ debe corresponder exactamente al porcentaje de potencia necesario para mantener un equilibrio entre lo entregado al sistema y sus pérdidas. Estos valores son ajustados a prueba y error hasta lograr una respuesta óptima del sistema.

Octava etapa.- El último paso es usar la fórmula del control difuso para la programación. Las funciones de membresía para la variable Error, definidas y dibujadas en la tercera etapa, Fig.2, sirven de referencia para calcular los grados de pertenencia μ_{CER} y μ_{POS} en todo el universo de trabajo. Estos valores se sustituyen en la ecuación 20, junto con los valores de los singletons BAJ y ALT, para calcular el porcentaje de potencia que debe ser entregado a la carga Pot.

V. Aplicación; Un control de temperatura

De acuerdo con el método descrito se uso la ecuación 20 para implementar un sistema de control difuso para la variable temperatura. Es decir Xref= Tref y Xmed= Tmed. En la tabla I se presentan algunos valores de Tmed, su Error correspondiente y el porcentaje de potencia que debe ser entregado a la carga Pot. El valor de Pot se calculó con la ecuación 20, para el caso de BAJ= 25, ALT= 100 y Tref= 37.2°C. Los valores de esta tabla se usan para programar al CPLD. El control difuso en si es una tabla y dicha tabla se programa en el CPLD, se usan de esta tabla solo las columnas Tmed y Pot. La programación se compone básicamente de comparaciones, por ejemplo; si Tmed es 36.3°C, entonces Pot es

85%. Y al momento de activar al control el CPLD lo que hace es leer el dato correspondiente a la medición de la temperatura que le entrega el microcontrolador, compara ese dato con los valores de la tabla, identifica su valor correspondiente de potencia y genera el PWM. En base a las pruebas hechas, para optimizar la respuesta del sistema, se consideran como suficientes los siguientes ajustes: variar a prueba y error el universo de discurso y la pendiente de las funciones de membresía de la variable Error (Fig.2), variar a prueba y error los valores de los singletons BAJ y ALT (ecuación 20), que corresponden respectivamente a las pérdidas del sistema, y a la velocidad de respuesta del sistema.

Entre más pérdidas de calor tenga el sistema, el valor de BAJ requerido será mayor. Si se quiere que el sistema responda a la máxima velocidad, ALT debe ser 100. Una vez hecho esto, el tiempo real dependerá solo de la potencia de la fuente de calor utilizada.

Tmed	ERROR	μ_{Cer}	μ_{Pos}	Pot
36.1	1.1	0	1	100
36.15	1.05	0.05	0.95	96
36.2	1	0.1	0.9	93
36.25	0.95	0.15	0.85	89
36.3	0.9	0.2	0.8	85
36.35	0.85	0.25	0.75	81
36.4	0.8	0.3	0.7	78
36.45	0.75	0.35	0.65	74
36.5	0.7	0.4	0.6	70
36.55	0.65	0.45	0.55	66

Tabla I. Tabla para la programación.
(BAJ= 25, ALT= 100 y Tref= 37.2°C).

La gráfica de la ecuación 20 se muestra en la Fig.5. Los cálculos necesarios para graficarla se obtuvieron considerando BAJ= 25 y ALT= 100. Después de las modificaciones del universo de discurso, para ($Error \geq 1.2$) se seleccionó $\mu_{CER}=0$ y $\mu_{POS}=1$, por tanto en la ecuación 20 predomina μ_{POS} y Pot= 100. Para ($-0.2 \leq Error \leq +0.2$), se seleccionó $\mu_{CER}=1$ y $\mu_{POS}=0$, por tanto en la ecuación 20 predomina μ_{CER} y Pot= 25.

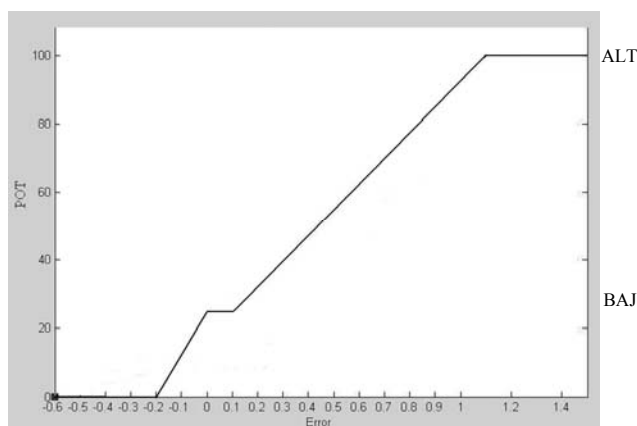


Fig.5 Pot vs. Error

VI. Resultados

Para pruebas se usó un prototipo de una incubadora de aves con una capacidad de 750 huevos de gallina. La tabla I se usó para programar a un CPLD XC95108. En el CPLD se implementó el control difuso y el PWM. También se implementó una etapa de medición de la temperatura que incluye un convertidor analógico digital integrado en un microcontrolador 16f877A. Además se construyó una etapa de potencia que consta de un control integral y un circuito de detección de cruce por cero. En el circuito electrónico de medición de la temperatura se usó un amplificador restador para mejorar la resolución del sistema, la cual quedó en 0.05°C/bit. Los resultados de control de la temperatura en función del tiempo se presentan en las Fig. 6- 8.

La Fig. 6 ilustra el comportamiento de la temperatura cuando se aplica el control difuso. Ahí se presentan los resultados del control para una temperatura de referencia Tref=38.2°C. El sistema tarda 18 min. para estabilizarse y alcanzar dicha temperatura. Por arriba del minuto 23 se mantiene variando entre 38.20 y 38.25°C, lo cual da una variación de +0.05°C.

En la Fig. 7 se presentan los resultados del control para una temperatura de referencia Tref=37.2°C. El sistema tarda 16 min. para llegar a 37.15°C. Y Por arriba del minuto 17 se mantiene variando entre 37.20 y 37.25°C, lo cual da una variación de +0.05°C.

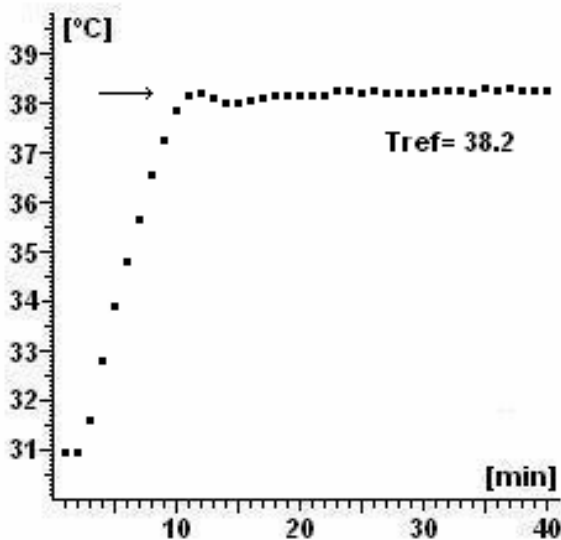


Fig. 6 Respuesta del Sistema para $T_{ref}= 38.2^{\circ}\text{C}$

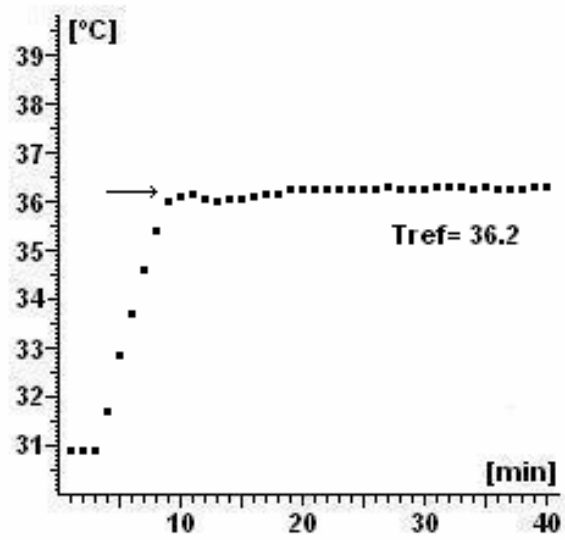


Fig. 8 Respuesta del Sistema para $T_{ref}= 36.2^{\circ}\text{C}$

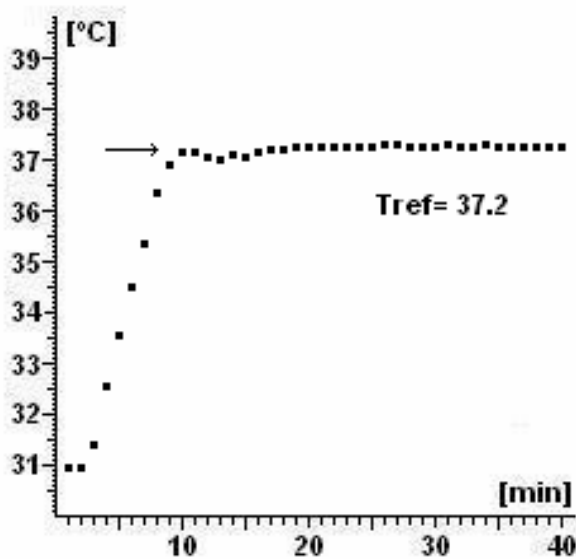


Fig. 7 Respuesta del Sistema para $T_{ref}= 37.2^{\circ}\text{C}$

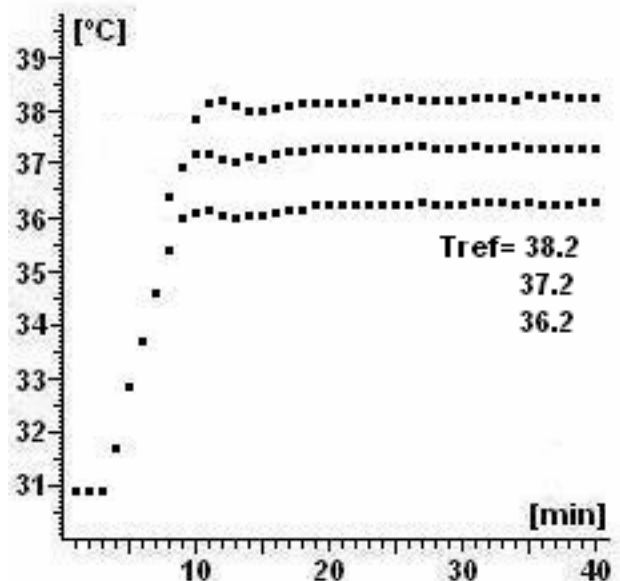


Fig. 9 Respuesta del Sistema a tres temperaturas de referencia.

En la Fig. 8 se presentan los resultados para una temperatura de referencia $T_{ref}=36.2^{\circ}\text{C}$. El sistema tarda 17 min. para llegar a 36.15°C . Por arriba del minuto 18 se mantiene variando entre 36.25 y 37.30°C , lo cual da una variación de 0.05°C . Sin embargo, aquí se pasa la temperatura y se requiere disminuir el valor de BAJ para que coincidan T_{med} y $T_{ref}= 36.2^{\circ}\text{C}$.

En la Fig. 9 se presentan los resultados en conjunto para las tres temperaturas de referencia usadas en las mediciones. Se escogieron estas temperaturas porque el control difuso fue desarrollado para incubadoras de aves. Y las temperaturas de incubación de las especies caen dentro de ese rango, ejemplo; avestruz 36.4°C , faisán 37.7°C , gallinas de rancho 37.0 a 38.0°C . En base a los resultados de la Fig. 9 se puede decir que el comportamiento del control difuso es igual para el rango de 36.2°C a 38.2°C . La única

desventaja es que a menor temperatura son menores las pérdidas de energía, por tanto se requiere introducir un factor de corrección para el singleton BAJ. Esta mejora se tiene prevista para trabajos próximos.

VII. CONCLUSIONES

En este trabajo se presentó un procedimiento para implementar un control difuso con un mínimo de recursos de software y principalmente sin la necesidad de programar la fusificación, el conjunto de reglas y la defusificación. El procedimiento se basa en la hipótesis de que la programación de la fusificación, el conjunto de reglas y la defusificación puede ser sustituida por un análisis (incluido en el presente trabajo), el cual permite, a través de simplificaciones, obtener una fórmula de diseño sencilla, la cual representa por completo al control difuso. La fórmula de diseño tiene la característica de simplificar significativamente los cálculos necesarios para la implementación del control difuso, lo cual reduce el nivel de dificultad de la programación.

Además en este trabajo se presentó un sistema de control de temperatura basado en un microcontrolador y un CPLD. En el CPLD se implementó el control difuso, encargado de determinar la demanda de potencia, además de generar la señal PWM encargada de regular la potencia entregada a la carga. El sistema de control de temperatura se caracterizó por su buen desempeño.

También se presentaron resultados de mediciones a distintos valores de la temperatura de referencia. Los resultados mostraron que el control difuso responde con precisión y de manera similar a distintas temperaturas, la única desventaja es que al cambiar la temperatura de referencia varían las pérdidas de calor del sistema y la temperatura del sistema tiende a desviarse de la temperatura de referencia. Una solución a esto es introducir un factor de corrección del singleton BAJ en función de X_{ref} .

REFERENCIAS

- [1] Herrera Galicia Rubén, Díaz Castellanos Mario., *Procedimiento Práctico Para Diseñar Sistemas de Control Utilizando Lógica difusa*, III Semana Nacional de Ingeniería Electrónica SENIE 07, Universidad Autónoma Metropolitana, México D.F., pp. 163- 170, 3- 5 OCT. 2007.
- [2] Joaquín Eduardo Domínguez Zenteno, *Sistema Experto Difuso Para Controlar Un Calentador Solar*, Tesis De Maestría, Instituto Tecnológico De Toluca, Metepec, México, Marzo 2001
- [3] Kevin M. Passino
Steven Yurkovich
Fuzzy Control
Addison-Wesley, Longman 2004
- [4] Roland S. Burns
Advanced Control Engineering
ButterWorth-Heinemann, 2005.
- [5] Fernando Lara-Rojo, Edgar N.Sanchez, Daniel Zaldivar- Navarro, *Real time fuzzy microcontroller for didactic level system*, ELECTRO 2001, Chihuahua, Chi. México.
- [6] Flavio Torres V., Carlos Muñoz P., Iván Velásquez B., *Control difuso para una planta piloto de secado por lecho fluidizado*, REVISTA FACULTAD DE INGENIERIA, U.T.A. (CHILE), Vol. 1, pp. 49-53, 2003.

The two ray technique applied to the RFID technology

S. Ricardo Meneses G., Roberto Linares, Laura Montes and Cristóbal Pérez
 Instituto Politécnico Nacional
 Escuela Superior de Ingeniería Mecánica y Eléctrica, Campus Zacatenco
 U.P.A.L.M. Edif. Z, Acc 3, 3er. Piso, Col. Lindavista, C. P. 07738, México, D. F.
rmeneseg@ipn.mx; rlinaresy@ipn.mx

Abstract — Multipath electromagnetic environment is very complex and assumptions have been made in order to obtain optimum models. The ray tracing model and the two ray model represent a tool for modeling the radio channel. In this work, measurement of the electromagnetic field and the power were taken in the field and in the electromagnetic compatibility laboratory, in order to know the electric field performance of the Radio Frequency Identification Communication System. The two rays model is applied to a communication system, when the transmitter is in a fix position meanwhile the receiver is moving.

Keywords— Incidence angle, reflection coefficient, multipaths.

I. Introduction

Radio wave propagation can be defined as the energy transfer by the electromagnetic radiation in the radiofrequency band. The radio wave propagates in several ways being dependent of the path geometry, the frequency and the electric properties of the propagation medium.

In a radio mobile typical application, the base station has a fix position meanwhile the mobile station is moving. Generally the propagation is suffering dispersion, reflection and diffraction, due the buildings, people, different obstacles that cancel the line of sight. The radio waves reach the mobile receiver from different paths, each one with different amplitudes, phases, delays. This situation is known as multipath propagation.

The radio channel is built with the sum of the all paths. The source, in the case of RFID systems, the antenna or radiator becomes in a source emitting rays to all directions, some of them reach the observation point, due to reflection, diffraction, refraction or direct way. These basic propagation mechanisms have an impact on the mobile systems communications propagation. The electric field and the received power are the most important predictable parameters used by the propagation models based in these mechanisms.

II. Statement of the problem

The effects on the wave propagation due to lossy terrain could be locally modeled by a plane surface. The homogenous surface effects, it is a Sommerfeld classic problem, an infinitesimal electric dipole radiating above of a lossy surface, and it can be applied to know the electric field performance in the RFID technology.

The radiation properties of the antennas in a stratified plane medium, have served to find approximate solutions in the electromagnetic waves area. The integrals Sommerfeld exact solution, where a source located in a plane, being an interface between two different mediums, with different permittivity and permeability, could be an option to determine the near and the far field and be applied to a RFID communication system. However, closed form solutions for fields depending in time and space, excited for distributed sources depending in time and space too, are difficult to obtain. There are various formal solutions for this kind of problem, but the explicit evaluation requires a complicated integration process, depending again of the time and space distributed sources.

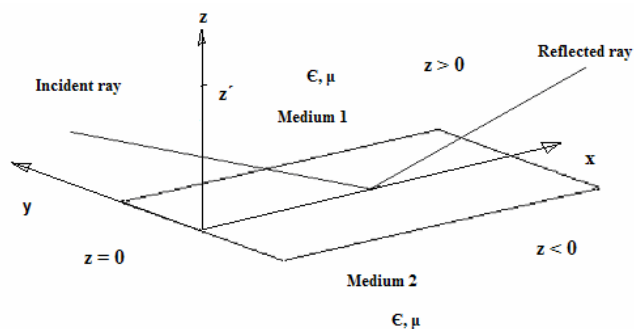


Fig.1. Two mediums model geometry

The Sommerfeld solution is expressed in terms of integrals of the Sommerfeld type, which cannot be evaluated in closed form due to their highly oscillatory nature are difficult to evaluate numerically. To evaluate the Sommerfeld integrals analytically standard asymptotic

techniques, such as the method steepest decent (saddle-point method), are applied. These techniques are valid when the distance between source and observation is large, far field. For source and observation near the surface the direct and reflected waves tend to cancel and higher order terms, Norton Surface Wave, in the asymptotic expansion are dominate.

Parhami et al [1] proposed a method to evaluate the Sommerfeld integrals numerically, valid for a vertical electric dipole, in which the integration contour is deformed to the steepest descent path. The integral is solved asymptotically when the distance between the image and the observation point is large, and numerically when the distance is small.

In order to improve the convergence properties of the Sommerfeld type integrals, the image theory is applied. In this method, an integral transform of a Laplace Transform is applied to the reflection coefficients in the Sommerfeld type integrals, [2] and [3].

The diffracted electric field due the Image Theory has the next expression [4]:

$$\begin{aligned} \bar{E}(\bar{r}, \bar{r}') = & \frac{kZ_0}{4\pi} \left\{ \hat{x} \int_0^\infty \frac{k_\rho}{2k_z} \left[\Gamma_h \begin{array}{l} -l_x (J_2(k_\rho \rho) \cos 2\phi) \\ + J_0(k_\rho \rho) \\ -l_y J_2(k_\rho \rho) \sin 2\phi \end{array} \right. \right. \\ & + \Gamma_v \left. \left. \begin{array}{l} \frac{2ik_z k_\rho}{k^2} l_z \cos \phi J_1(k_\rho \rho) \\ + \frac{k^2}{k^2} l_x \cdot (J_0(k_\rho \rho) \cos 2\phi) - \\ \frac{k_z^2}{k^2} l_y J_2(k_\rho \rho) \sin 2\phi \end{array} \right] \right\} e^{ik_z(z+z')} dk_\rho \\ & + \hat{y} \int_0^\infty \frac{k_\rho}{2k_z} \left\{ \Gamma_h \begin{array}{l} -l_x J_2(k_\rho \rho) \sin 2\phi - l_y (J_0(k_\rho \rho) \\ - J_2(k_\rho \rho) \cos 2\phi \end{array} \right\} + \\ & \Gamma_v \left. \begin{array}{l} \frac{2ik_z k_\rho}{k^2} l_z \sin \phi J_1(k_\rho \rho) - \frac{k_z^2}{k^2} l_x J_2(k_\rho \rho) \sin 2\phi \\ + \frac{k_z^2}{k^2} l_y (J_0(k_\rho \rho) + J_2(k_\rho \rho) \cos 2\phi \end{array} \right\} e^{ik_z(z+z')} dk_\rho \\ & - \hat{z} \int_0^\infty \frac{k_\rho}{k_z} \Gamma_v \left. \begin{array}{l} \frac{k_\rho^2}{k_z} l_z J_0(k_\rho \rho) + \\ \frac{ik_z k_\rho}{k^2} \cdot (l_x \cos \phi + l_y \sin \phi) J_1(k_\rho \rho) \end{array} \right\} e^{ik_z(z+z')} dk_\rho \end{aligned}$$

where:

$$\Gamma_h = \frac{\eta - k/k_z}{\eta + k/k_z}, \text{ horizontal reflection coefficient}$$

$$\Gamma_v = \frac{-\eta + k_z/k}{\eta + k_z/k}, \text{ vertical reflection coefficient}$$

$\eta = Z_1/Z_0 = \sqrt{1/(\epsilon' + i\epsilon'')}$, normalized impedance of the interface

$\epsilon_r = \epsilon' + i\epsilon''$, complex relative permittivity, medium 2

ρ , distance between the source and observation point

z , height of the observation point

ϕ , angle between ρ and x axis

J_0, J_1, J_2 , Bessel functions

z' , height of the source point

$$k_z^2 = k^2 - k_\rho^2$$

The above integral is a Sommerfeld type integral, which cannot be evaluated, highly oscillatory, with weak convergence properties. Numerically is difficult to evaluate.

For our study case showed in the figure 2, it is necessary determine the total electric field, the direct wave and the reflected wave, and/ the received power. In the face of this situation, to use the methods above mentioned is a hard and difficult work, so then, the two rays model is applied in order to know the electric field performance of the Radio Frequency Identification Communication System.

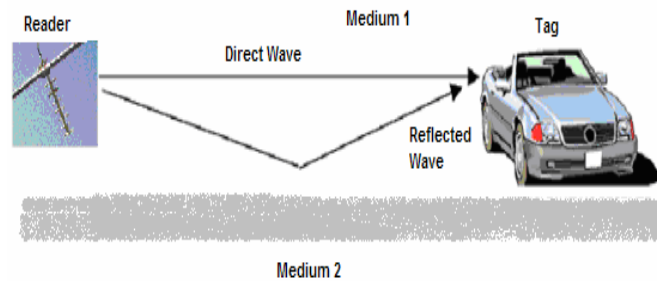


Fig.2. RFID technology applied to vehicular control

A. Two Ray Propagation Model

One direct path between the base station and the mobile station is rarely the only physical medium of propagation. This the reason, why the free space propagation model is imprecise model. The two rays ground reflection model showed in the figure 3 is a useful propagation model, it is

based in geometric optic (GO), as long as, the direct path and the reflected propagation path between the transmitter and the receiver is considered. This model has been found to be reasonably precise to predict large scale signal level for radio mobile systems.

The total electric field E_{TOT} is the sum of the direct component, line of sight component, E_{LOS} , and the reflected component E_g . The sum of the power due the direct ray and the power of the reflected ray, determine the total received power, but there is a 180° phase difference in this respect to the reflected ray.

The two rays model geometry is showed in the figure 3. h_1 is the height of the transmitter and h_2 is the height of the receiver.

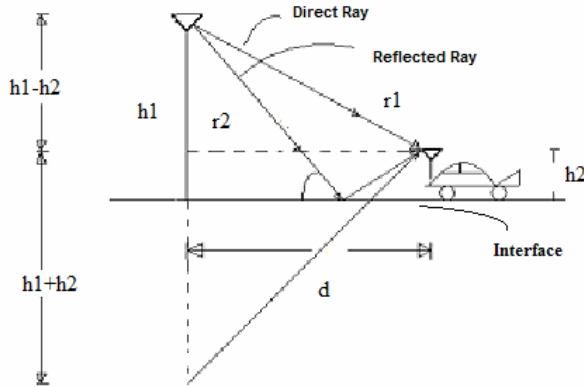


Fig.3. Two rays model geometry

This model considers that the reflection coefficient is based in the electric properties of the reflecting surface and the angle of incidence. So, due to contribution of each ray the received power is expressed in this way:

$$P_r = P_t \left(\frac{\lambda}{4\pi} \right)^2 \left| \frac{1}{r_1} \exp(-jkr_1) + \Gamma(\alpha) \frac{1}{r_2} \exp(-jkr_2) \right|^2 \quad (2)$$

where:

P_t , transmission power

d , distance between the transmitter and receiver

r_1 , distance of the direct ray

r_2 , distance of the reflected ray

$\Gamma(\alpha)$, reflection coefficient

being:

$$r_1 = \sqrt{d^2 + (h_1 - h_2)^2} = \sqrt{d^2 + h_1^2 + h_2^2 - 2h_1h_2} \quad (3)$$

$$r_2 = \sqrt{d^2 + (h_1 + h_2)^2} = \sqrt{d^2 + h_1^2 + h_2^2 + 2h_1h_2} \quad (4)$$

$$\sin \alpha = \frac{\sqrt{d^2 + (h_1^2 + h_2^2)}}{(h_1 + h_2)} \quad (4)$$

The reflection coefficient depends of the angle of incidence, signal polarization and the relative permittivity of the reflecting surface. The reflection coefficient is given by:

$$\Gamma(\alpha) = \frac{\cos(90 - \alpha) - a\sqrt{\epsilon_r - \sin^2(90 - \alpha)}}{\cos(90 - \alpha) + a\sqrt{\epsilon_r - \sin^2(90 - \alpha)}} \quad (5)$$

where:

α , angle of incidence

a , determined value given by the polarization

σ , conductivity of the reflecting surface

being:

Vertical polarization:

$$a_v = \frac{(k - j60\lambda\sigma)\sin \alpha - (k - j60\sigma\lambda - \cos^2 \alpha)^{1/2}}{(k - j60\sigma\lambda)\sin \alpha + (k - j60\sigma\lambda - \cos^2 \alpha)^{1/2}} \quad (6)$$

Horizontal polarization:

$$a_h = \frac{\sin \alpha - (k - j60\sigma\lambda - \cos^2 \alpha)^{1/2}}{\sin \alpha + (k - j60\sigma\lambda - \cos^2 \alpha)^{1/2}} \quad (7)$$

B. Performance of the electric field in this respect to the distance

The figure 4 shows the received power vs. distance graphic, using the two rays model expressed in (3). The operation frequency is 915 MHz, assigned frequency for this application, the relative permittivity of the reflecting surface is the asphalt permittivity, the height of the reader, the transmitter, 5m, and the height of the tag, the receiver, 1.2 m. Several measurements were taken when the speed of the mobile was 40 Km/h, 60 Km/h and 80 Km/h. The reference position is the reader position. This point is the 0 m position point.

The figure shows the constructive and the destructive contributions of the direct and the reflected rays. The measurement were taken in marked points between the reader and tag distance, each point was separated of the adjacent point by a 1 meter, as shown in the figure 5. The speed of the mobile was 60 Km/h.

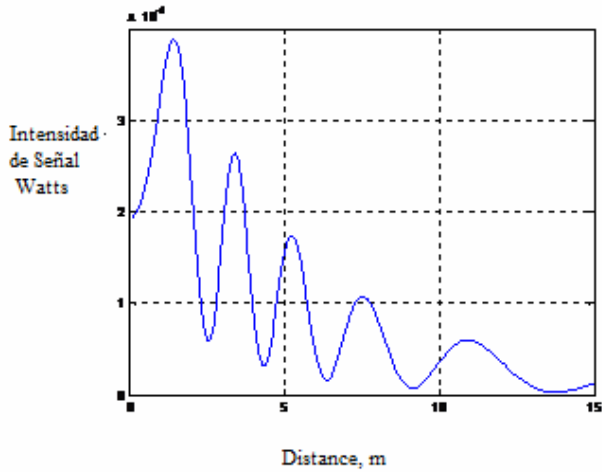


Fig.4. Received power vs. Distance, using the two rays model expressed in (2). The operation frequency is 915 MHz, vertical polarization, height of the reader 5 m, height of the tag 1.2 m, speed of the mobile, 60 Km/h.

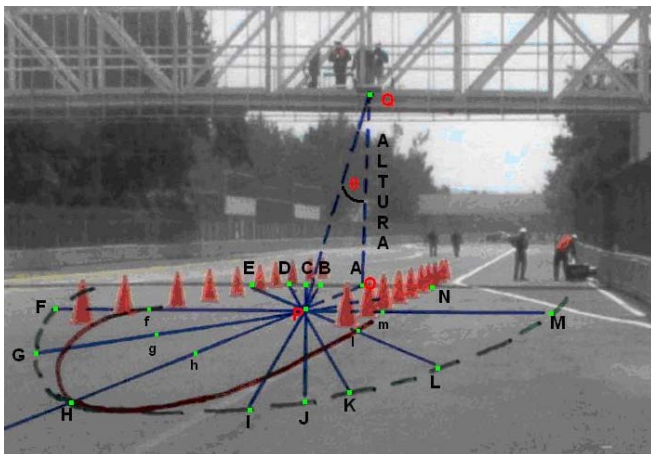


Fig.5. Measurements in the road

From the showed graphics, it is possible to observe that the points of maximum level of the received power is due to constructive distribution of the direct and reflected rays, and the points of minimum level is due to destructive contribution of the them. The destructive contribution increases when the height of the receiver is large, and it decreases when the height of the receiver is small.

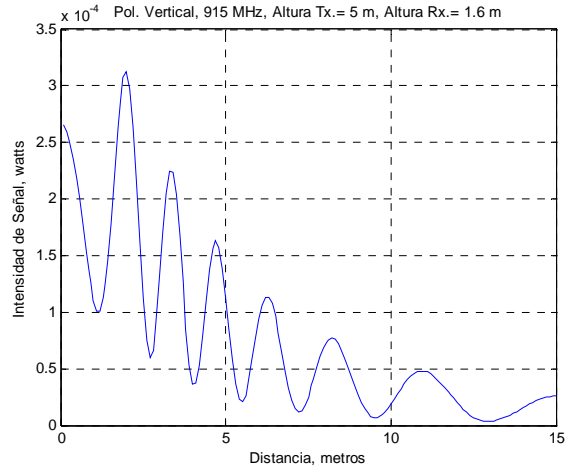


Fig.6. Received power vs. Distance, using the two rays model expressed in (2). The operation frequency is 915 MHz, vertical polarization, height of the reader 5 m, height of the tag 1.6 m, speed of the mobile, 60 Km/h.

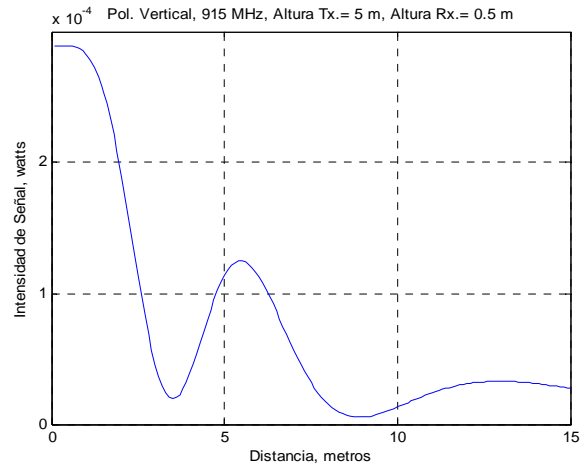


Fig.7. Received power vs. Distance, using the two rays model expressed in (2). The operation frequency is 915 MHz, vertical polarization, height of the reader 5 m, height of the tag 0.5 m, speed of the mobile, 60 Km/h.

C. Power spatial distribution

The measurements show a continuous transmission by the reader, however, the covered area does not have the same signal level. A power spatial distribution average around 15 m, was obtained, showed in the figure 9. The red color represents a maximum signal level, and the white color represents the minimum signal level. The measurements were taken using a log-periodic antenna, a spectrum analyzer and a data acquisition program.



Fig. 8. The reader is installed in the post and the tag is installed in the windshield of the car.

III. Conclusion

The present work shows the two rays model applied to RFID technology in order to know the power and the electric field performance. The angle of incidence, the electric properties of the reflecting surface, the frequency, the height of the reader and the tag determine the performance of the power and the electric field. The propagation model allows us to predict the power and the electric field spatial distribution, in order to optimize the receiver operation.

IV. References

- [1] P. Parhami, Y. Rahmat-Samii, and R. Mitra, "An efficient approach for evaluating Sommerfeld Integrals encountered in the problem of a current element radiating over lossy ground" *IEEE Trans. Antennas Propagation*, vol. 28, pp. 100-104, Jan 1980.
- [2] B. van der Pol, "Theory of the reflection of the light from a point source by a finitely conducting conducting flat mirror: With an applications to radiotelegraphy," *Physics*, vol. 2, pp. 843-853, 1935.
- [3] K. Furutsu, "On the excitation of the waves of proper solutions," *I.R.E. Trans. Antennas Propagation*, vol. 7, pp.209-218, 1959. Special Supplement.
- [4] K. Sarabandi, M. Casciato, I. Suez Koh, "Efficient calculation of the fields of a dipole radiating above an impedance surface", *IEEE Trans. on Antennas and Propagation*, Vol. 50, No. 9, September 2002.

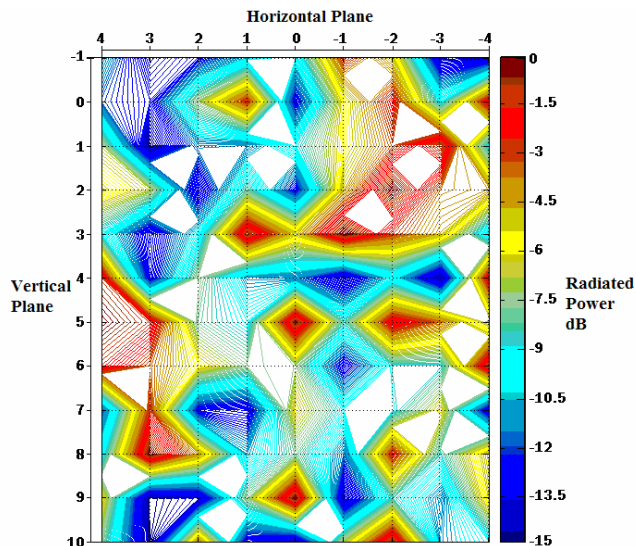


Fig. 9. Power spatial distribution

- [5] A. N. Sommerfeld, *Partial Differential Equations in Physics*. New York, Academic, 1949.
- [6] R. Linares y M., J. L. López Bonilla, M. Acevedo, and R. Meneses, "Multipath environment indoor analysis by Sommerfeld integrals, geometric optics, and Norton surface wave", *Indian Journal Physics*, 77B (3), June 2003.
- [7] R. Meneses, R. Linares y M., and J. L. López Bonilla, "The multipath electromagnetic environment, the two ray model in the RFID systems", *ICFAI University J. of Physics*, Vol. II, April 2008.
- [8] Anderson, R. Harry, "A Ray-Tracing Propagation Model for Digital Broadcast Systems in Urban Areas", *IEEE Transactions on Broadcasting*, Vol. 39, No. 3, September 1993.
- [9] Xia Howard., Bertoni H., Maciel L., Lindsay A., Rowe R. "Radio Propagation Characteristics for Line-of-Sight Microcellular and Personal Communications", *IEEE Transactions on Antennas and Propagation*, Vol. 41, No. 10, October 1993.
- [10] K. I. Blackard, M. J. Feuerstein, T. S. Rappaport, S. Y. Seidel. And H. H. Xia, "Path loss and delay spread models as function of antenna height for microcellular system design," *Proceedings of the 1992 IEEE Vehicular Technology Conference*, Denver, CO, pp. 333-337, May 1992.
- [11] Y. Oda, K. Tsunekawa, M. Hata, "Advanced LOS Path Loss Model in Microcellular Mobile Communications," *IEEE Transactions of Vehicular Technology*, Vol. 49, No.6, November 2000.

[12] Volkert W. Hansen, “Numerical Solution of Antennas in Layered Media”, Research Studies Press Ltd. 1989, England

[13] Chen To Tai, “Dyadic Green Functions in Electromagnetic Theory”, IEEE Press, 1993.

[14] T. Rappaport, “Wireless Communications Principles and Practice”, Prentice Hall, New Jersey 1996.

BIOGRAPHIES

Salvador Ricardo Meneses González.

Salvador Ricardo Meneses González. Communications and Electronic Engineer from ESIME IPN (1981). M. S. E. degree in electrical engineering from Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV IPN). He is actually professor and departmental head of Communications and Electronic Engineering of the ESIME Zacatenco, IPN.



Dr. Roberto Linares y Miranda, Ph. D. degree from Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV-IPN). He is actually professor in the Post-graduate Section of the ESIME Zacatenco.



REDES DE RADIOS COGNITIVOS. UN ESTUDIO EXPLORATORIO

Jesús Zukhov Sánchez Garrido, José A. Incera Diéguez

Instituto Tecnológico Autónomo de México
División Académica de Ingeniería
Río Hondo No. 1 Col. Tizapán Progreso, México, D.F. 01080, México
Tel. 56284000, correo electrónico: zukhov@hotmail.com, jincera@itam.mx

RESUMEN

Los radios cognitivos, un tipo de radio definido por software (SDR), pueden ofrecer una solución al problema de la escasez del espectro radioeléctrico, por su capacidad de reconfiguración basada en mediciones obtenidas de su medio operacional. Debido al gran interés que han adquirido recientemente, este artículo presenta el estado del arte en radios cognitivos. Además de una breve descripción de las tecnologías subyacentes, se identifican los principales actores que están desarrollando estas redes, en qué estado se encuentran los proyectos más relevantes, así como los trabajos que llevados a cabo por organismos regulatorios, de estandarización y académicos en el área.

Palabras clave: Radios cognitivos, radios definidos por software, espectro radioeléctrico, reconfigurabilidad.

I. INTRODUCCIÓN

La administración del espectro radioeléctrico se realiza en prácticamente todos los países, a través de políticas autónomas, pero siguiendo las recomendaciones de la Unión Internacional de Telecomunicaciones (ITU). Tradicionalmente esta administración consiste en asignar bandas del espectro a servicios específicos, llamados de uso primario. Así, se reconocen –de manera prácticamente universal- las bandas en que operan los distintos sistemas de radiodifusión, televisión, comunicaciones satelitales, redes de telefonía celular, entre otros.

Esta forma de administración, sin embargo, puede resultar sumamente ineficiente, pues algunas bandas asignadas pueden estar desocupadas tanto en alguna región (agujero espacial) como durante algún –posiblemente largo- periodo de tiempo (agujero temporal). La *Federal Communications Commission*, en los Estados Unidos, ha detectado que estas variaciones oscilan entre un 15% y un 85% de utilización del espectro asignado [1].

Si el espectro es un recurso limitado y además se utiliza de manera ineficiente, será cada vez más difícil satisfacer la creciente demanda de servicios de comunicación inalámbricos. Por ello se han buscado soluciones para hacer uso del espectro radioeléctrico de forma más eficiente.

Una primera opción es el uso de bandas de frecuencia libres que no requieren de licencia para operar, aunque sí deben respetarse restricciones técnicas y administrativas impuestas por las autoridades regulatorias, notablemente en cuanto a su potencia de radiación, lo que limita severamente el tipo de servicios que se pueden ofrecer con estas frecuencias [2].

Ejemplo de bandas de frecuencia libres son las bandas destinadas para aplicaciones industriales, científicas y médicas (ICM). En México ocupan los bloques de 902 – 928 MHz, 2.4 – 2.5 GHz y 5.725 – 5.875GHz. En estas bandas se encuentran servicios como controles remotos, teléfonos inalámbricos, y redes con tecnologías *Bluetooth* y *Wi-Fi*. Se ha observado que estas bandas se encuentran abarrotadas, lo cual se traduce en una

mayor interferencia en los equipos de comunicación [3].

Otra alternativa propuesta para mejorar la eficiencia en el uso del espectro radioeléctrico proviene de los radios definidos por software (SDR) los cuales forman un sistema reconfigurable de radiocomunicación mediante la modificación dinámica de sus parámetros de operación [4]. Esta configuración puede estar presente tanto en los equipos terminales como en las estaciones base y se logra mediante diversos módulos de software corriendo sobre una plataforma genérica de hardware [5].

En resumen, los SDR pueden operar en múltiples bandas de frecuencia utilizando diversos estándares y mediante una simple actualización de su software son capaces de adicionar nuevos servicios y acoplarse a nuevos estándares. Un tipo de SDR son los Radios Cognitivos.

II. RADIOS COGNITIVOS

Haykin [6] define los radios cognitivos o ágiles como *un sistema de comunicaciones inteligente que está consciente del entorno que lo rodea y usa la metodología de “understanding-by-building” para aprender del medio ambiente y adapta sus estados internos a variaciones estadísticas basadas en los estímulos de radiofrecuencia recibidos por medio de cambios en sus parámetros de operación* (por ej., potencia de transmisión, frecuencia de portadora, estrategia de modulación) *en tiempo real teniendo dos objetivos principales: tener un sistema de comunicaciones confiable donde se necesite y cuando se necesite, y una utilización eficiente del espectro radioeléctrico.*

Así, las características que deben estar presentes en los RC son: conocimiento de su entorno, aprendizaje, adaptabilidad a estímulos, eficiencia y confiabilidad. Estas características se encuentran en las tres tareas principales del ciclo cognitivo básico de los RC.

2.1. Ciclo Cognitivo

Los RC deben tener conocimiento de su entorno. Esto significa que deben tener comunicación con otros dispositivos para conocer sus de parámetros de operación (frecuencia y técnica de modulación utilizada, potencia de transmisión, localización y red de operación) con el fin de poder adoptar la configuración que le permita adaptarse al medio que lo rodea.

El proceso cognitivo comienza con el monitoreo de los estímulos y culmina con una acción, siguiendo el ciclo que se ilustra en la figura 1, el cual consiste de tres etapas básicas: monitoreo, análisis y decisión [6]. Las primeras dos etapas se llevan a cabo en el receptor y la última en el transmisor.

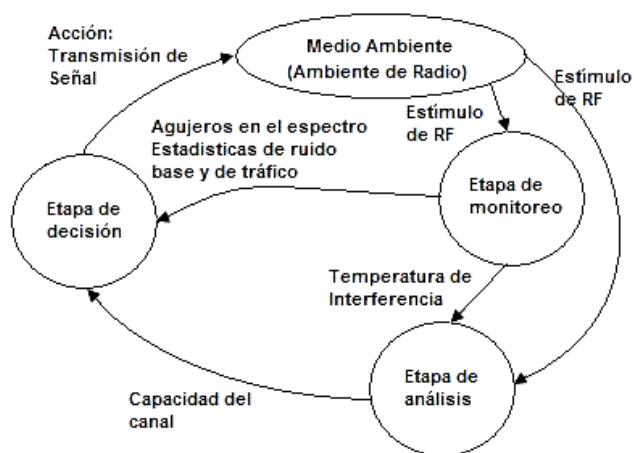


Figura 1: Ciclo básico de cognición [6].

a) Etapa de monitoreo

En esta etapa el RC recolecta información de su estado interno y del medio que lo rodea con el fin de detectar agujeros en el espectro. Para ello es necesario determinar las frecuencias que están siendo utilizadas y cuándo, inferir la posición de transmisores y receptores, así como los niveles ruido y de temperatura de interferencia.

La FCC introdujo el concepto de *Temperatura de Interferencia (TI)* como *la medida de potencia de radiofrecuencia generada por emisores no deseados así como por fuentes de ruido presentes*

en un sistema receptor por unidad de ancho de banda [7].

Con este modelo, los servicios que cuenten con una licencia de operación pueden tener un mejor conocimiento acerca del nivel máximo de interferencia presente, asegurando que no habrá una degradación o pérdida del servicio. Lo anterior se logra definiendo un límite de TI como el nivel máximo de potencia al que puede llegar el ruido debido a la presencia de nuevas fuentes de interferencia. Cualquier señal recibida con una potencia superior al límite de TI podrá ser reconocida por el receptor, como se muestra en la figura 2.

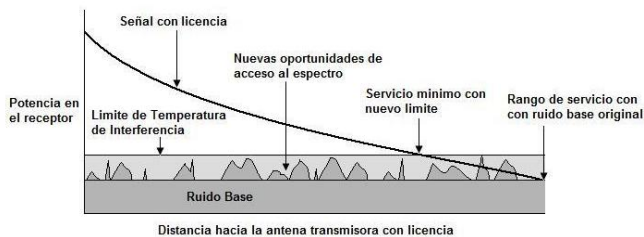


Figura 2: Modelo de Temperatura de Interferencia [7]

El modelo de TI también es aprovechado por los usuarios secundarios (aquellos que no tienen licencia sobre el servicio primario en una banda determinada), debido a que el mismo límite podrá ser usado como una barrera que delimita la cantidad máxima de potencia que puede ser introducida en una banda de frecuencia en particular sin que afecte a las transmisiones del usuario primario [7].

En caso de que el límite de TI sea rebasado, los transmisores deberán buscar otra frecuencia con niveles de TI inferiores. Si no se encuentra una disponible en ese momento, el transmisor deberá esperar a que en alguna de ellas el nivel disminuya lo suficiente para poder transmitir sin rebasar el límite. Otra opción es disminuir el nivel de potencia de su transmisión para así poder operar sin que sea rebasado el límite establecido.

Un agujero en el espectro es una banda de frecuencias asignada a un servicio primario, pero

que en un tiempo y zona geográfica particular, no está siendo utilizada [8]. Por consiguiente, se pueden identificar agujeros en el espacio y en el tiempo.

Un agujero espacial se ejemplifica en la figura 3, donde se observa que existen áreas en las que la potencia de transmisión de los operadores primarios para una banda determinada, es indetectable. En ellas, un usuario podría realizar sus comunicaciones con la potencia suficiente para no interferir con los primarios [9].

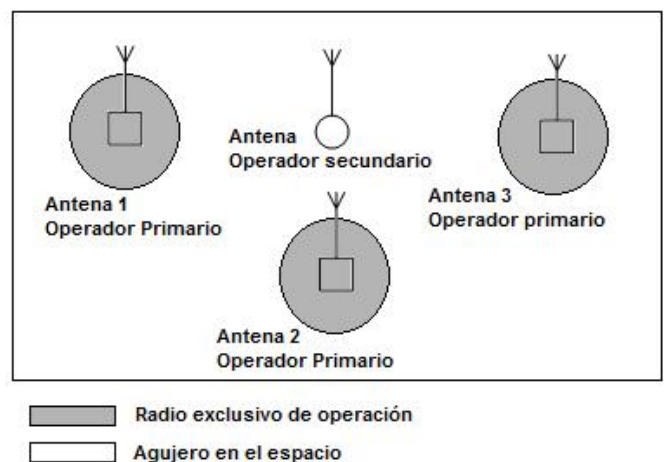


Figura 3: Agujero en el espacio [9].

Para que un usuario secundario sepa si se encuentra fuera del radio de operación de los sistemas primarios, se tienen dos opciones. La primera consiste en tener la localización de los transmisores primarios en un registro. El sistema secundario debe ser capaz de calcular su propia posición e inferir a qué distancia se encuentra de los transmisores primarios [9]. Sin embargo, esto implica que el usuario secundario dependerá en gran medida del sistema primario.

Otra manera de calcular la distancia hacia el transmisor primario es midiendo la potencia de la transmisión primaria en el receptor secundario (P). El sistema secundario podrá transmitir si:

$$P \leq P_t - 10 \log_{10}(r_n^\alpha + \Delta), \quad (1)$$

donde α es el factor de atenuación de la señal, P_t es la potencia de transmisión del transmisor primario, r_n es el radio del área global de silencio (r_n), y Δ representa los efectos de atenuación provocados por la multitrayectoria y *shadowing* [9].

Los agujeros del espectro en el tiempo ocurren cuando el operador primario realiza transmisiones interrumpidas, como se ejemplifica en la figura 4 donde se muestran tres fuentes que transmiten intermitentemente durante el periodo observado. Para poder aprovechar estos espacios, el usuario secundario debe ser capaz de predecir los tiempos de silencio, y éstos deben ser lo suficientemente largos para poder ser aprovechados [9].

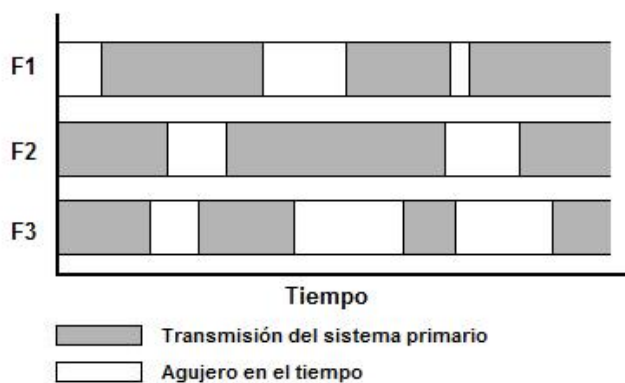


Figura 4: Agujeros en el tiempo.

b) Etapa de análisis

En esta etapa se estima la información del estado actual del canal y se predice la capacidad total que estará disponible.

Las características del canal son específicas para cada banda de frecuencia y varían a lo largo del tiempo. Su monitoreo constante es indispensable para determinar la banda más apropiada para las necesidades del sistema. Los datos que deben ser monitoreados para lograr una adecuada caracterización del canal son [1]:

- *Interferencia:* Con base en ella se obtiene la capacidad del canal. Los niveles de interferencia difieren en cada banda de frecuencia, en función de su uso.

- *Atenuación:* En general, la atenuación aumenta con la frecuencia, por lo que a frecuencias mayores, se deberán transmitir con mayor potencia si se desea mantener el mismo rango de operación.
- *Tasa de error:* Depende de varios factores, como la modulación utilizada y el nivel de interferencia presente en el canal.
- *Retardo en la capa de enlace de datos:* Debido a los factores anteriores, es necesario utilizar diversos protocolos de la capa de enlace en las distintas bandas de frecuencia utilizadas, lo cual se traduce en diferencias en los retardos de transmisión.
- *Tiempo de retención:* Representa el periodo durante el cual el usuario secundario puede hacer uso del canal antes de que el primario lo recupere.

A partir de estas mediciones, junto con cálculos del SNR y el modelo de TI presentado anteriormente, es posible determinar la frecuencia y la máxima potencia a la que se puede transmitir sin afectar al sistema primario [1].

c) Etapa de decisión.

Determina la acción más apropiada para reaccionar a los estímulos recibidos. Una vez que las diversas banda de frecuencia han sido caracterizadas, el RC debe auto-configurarse para trabajar en la banda de frecuencia que sea más apropiada para cumplir los requerimientos del usuario y de calidad de servicio. Algunos de los parámetros que se reconfiguran son la tasa de transmisión, frecuencia de operación, modulación, potencia de transmisión, tecnología de comunicación y el ancho de banda utilizados [1].

III. ARQUITECTURAS DE REDES DE RADIOS COGNITIVOS.

Las redes de RC, también llamadas *Dynamic Spectrum Access (DSA) Networks*, trabajan tanto en bandas libres como con licencia, por lo que es conveniente describir su arquitectura general como la conjunción de la red primaria (del usuario primario) y la red de RCs.

La *red primaria* es la infraestructura de los operadores con derecho de uso en una región y banda de frecuencia específica. Está compuesta por los *usuarios primarios*, quienes pueden utilizar la infraestructura del operador, y las *estaciones base primarias* que controlan el acceso de los usuarios primarios y también les conceden acceso a los usuarios secundarios, como se explica más adelante.

La *red secundaria* puede operar como una red *ad-hoc* o como una red basada en infraestructura. En este último caso, sus componentes principales son: los *usuarios secundarios* con radios cognitivos, las *estaciones base secundarias* las cuales dan acceso a otras redes, y el *Spectrum Broker* cuya función es permitir la coexistencia de redes secundarias.

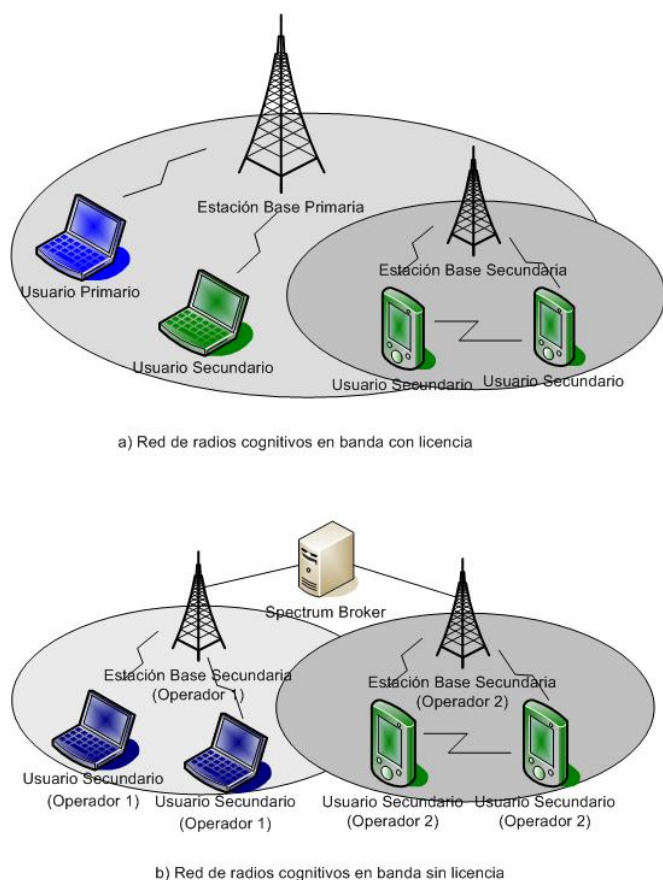


Figura 5: Tipos de acceso en redes de radios cognitivos según banda de operación [1].

Cuando las redes de RCs operan en bandas con licencia se tienen tres tipos de acceso: (1) *Acceso por la red de radios cognitivos*, donde los usuarios secundarios acceden por medio de su propia estación base; (2) *acceso por redes ad-hoc de radios cognitivos*, donde los usuarios secundarios puede comunicarse entre sí por medio de una conexión *ad-hoc*; (3) *acceso a la red primaria*, en donde el usuario secundario puede acceder directamente a la estación base primaria (figura 5a). En las bandas libres, las opciones de acceso son únicamente el *acceso por la red de radios cognitivos* y *acceso por redes ad-hoc de radios cognitivos* (figura 5b).

IV. ESTADO ACTUAL.

A continuación se presentan algunas de las arquitecturas de redes de RCs existentes.

Spectrum Pooling [10] La idea central consiste en que varios concesionarios de bloques de frecuencias en el espectro agregarán sus frecuencias en un grupo de donde podrán ser arrendadas por usuarios secundarios. Para lograr lo anterior, se utiliza OFDM como técnica de modulación. Con OFDM es posible dejar sub-portadoras sin modular para que un usuario secundario pueda hacer uso de las mismas.

La detección de sub-portadores libres y ocupadas por los usuarios primarios se realiza por poleo (sensado de portadoras) en los RC de los usuarios secundarios. Esta información se concentra en la estación base, la cual genera un vector de ocupación que es difundido a todos los usuarios secundarios.

CORVUS (*Cognitive Radio approach for usage of Virtual Unlicensed Spectrum*) [11]. Aquí los RC detectan las partes del espectro que están siendo sub-utilizadas y crean una banda libre virtual¹ que

¹ La creación virtual de una banda libre consiste en hacer uso de una banda de frecuencia asignada a un servicio primario como si se tratara de una banda libre cuando el usuario primario no está haciendo uso de ella.

puede ser utilizada sin dañar las transmisiones del usuario primario.

En esta arquitectura se definen grupos de usuarios secundarios, a los cuales les son asignados bloques de frecuencias para coordinar su comunicación. Los usuarios en un grupo se pueden comunicar directamente (configuración *ad-hoc*) o a través de una estación base. No se permite la comunicación entre grupos, aunque este es un problema que puede ser resuelto en capas superiores (a través de un mecanismo de enrutamiento).

IEEE 802.22 Este grupo de trabajo, conocido como *Wireless Regional Area Networks* (WRANs) [12], desarrolla una norma para dispositivos libres de licencia que trabajen en un esquema de no interferencia en las bandas UHF y VHF asignadas al servicio de televisión.

Especifica una comunicación punto-a-multipunto entre la estación base y los usuarios dentro de su área de cobertura, basada en una relación maestro-esclavo. La estación base autoriza a los RC el inicio de operaciones y notifica las características con las que éstos deben transmitir. También les solicita continuamente monitoreo del medio con el fin de proteger las eventuales transmisiones del operador primario [13].

DRiVE/overDRiVE Los proyectos DRiVE (*Dynamic Radio for IP Services in Vehicular Environments*) [14] y overDRiVE (*Spectrum Efficient Uni- and Multicast Services over Dynamic Multi-Radio Networks in Vehicular Environments*) [15] buscan desarrollar una red IP inalámbrica de alta calidad para entregar servicios multimedia a vehículos automotores. Para ello se sirve de redes celulares (GSM, GPRS y UMTS) y de difusión (DAB y DVB-T).

Utiliza un esquema de control de tráfico encargado de seleccionar la tecnología de acceso más apropiada en función del servicio requerido, tomando en consideración parámetros propios a la terminal (tipo, ubicación, capacidades), preferencias del usuario, requerimientos de calidad y estado de la red. Define también un Canal de

Coordinación Común (CCC) a través del cual se coordinan las distintas tecnologías para otorgar la asignación dinámica, eficiente y justa del espectro.

Nautilus Su objetivo es compartir el espectro de manera eficiente y escalable sin tener un esquema de control centralizado. Opera en configuraciones de red *Ad-hoc* con tres opciones de acceso [1]. La primera utiliza un algoritmo optimizado para nodos fijos, pues sus requerimientos computacionales hacen prohibitiva la reconfiguración continua en redes móviles. En estos casos, se recomienda la segunda opción en la que los nodos se organizan en *clusters* y la asignación de espectro se hace por cluster. La última opción, orientada a redes de sensores con fuertes restricciones de recursos, sugiere un mecanismo de acceso basado en reglas que deben ser consideradas, además de las observaciones locales del medio.

ORCA (OFDM-based Cognitive Radio). Dado que OFDM promete ser la tecnología dominante en redes inalámbricas, los RC en ORCA se basan en ella para sensar el medio y tomar decisiones rápidas de reconfiguración, disminuyendo así los efectos negativos del *handoff*. Una de sus características más llamativas es el acceso al espectro en modo dual, lo que le permite tener comunicación con redes ya existentes mientras sostiene comunicación simultáneamente con otros RC [1].

V. GRUPOS DE INTERÉS EN RADIOS COGNITIVOS

Por sus implicaciones en la manera en que se ha administrado el espectro tradicionalmente, resulta primordial conocer la actividad que sobre el tema están las entidades regulatorias, los grupos de estandarización, y las organizaciones industriales y académicas.

a) Entidades Regulatorias y de estandarización.

En Estados Unidos, la *Federal Communications Commission* creó el *Cognitive Radio Technologies Proceeding*² (CRTP) quien ha emitido una serie de

² www.fcc.gov/oet/cognitiveradio

reglas para regular el uso de radios cognitivos. Así, los RCs pueden hacer uso de bandas con licencia como operadores secundarios y permiten que el operador primario rente parte de su espectro radioeléctrico [16].

La *International Telecommunications Union*, por medio de su sector de radiocomunicación (ITU-R), está a cargo de la administración de las orbitas satelitales y el espectro radioeléctrico. Su *Study Group 1 (SG1) – Spectrum Management* emite principios generales de compartición y monitoreo del espectro, así como estrategias de utilización y económicas. Dentro de sus grupos de trabajo, el apartado 8A ha estado trabajando intensamente en definir las características, requerimientos, políticas y aplicaciones potenciales de radios cognitivos. El resultado de este grupo se presentará como recomendaciones ITU en 2010.

El *Institute of Electrical and Electronic Engineers* (IEEE) cuenta con tres proyectos enfocados al desarrollo de las redes de radios cognitivos. El primero de ellos, IEEE 802.22 ya ha sido tratado en este documento.

El IEEE *Standards Coordinating Committee 41* (IEEE SCC41, www.scc41.org), denominado también DySPAN (*Dynamic Spectrum Access Networks*), se encarga de analizar proyectos de asignación dinámica de espectro, radios cognitivos, gestión de interferencia y de asignación de espectro, entre otros.

Finalmente, el IEEE *Technical Sub-Committee on Cognitive Networks* (TCCN) [17] es un foro para el intercambio de ideas con la finalidad de identificar los mayores problemas en el desarrollo de las redes de radios cognitivos, así como de buscar posibles soluciones a los mismos.

En Europa, el *European Telecommunications Standard Institute* (ETSI) creó recientemente el comité técnico en Sistemas de Radio Reconfigurables (RRS) [18] los cuales engloban los SDRs y RCs. Este comité se encarga de las actividades de estandarización de los RRS, establecer sus principales requerimientos e

identificar temas relacionados que no sean cubiertos por ninguno de los otros estándares de la ETSI.

b) Empresas.

Prácticamente todas las empresas líderes en el ramo de radiocomunicaciones, como Philips y Motorola, tienen proyectos de investigación en radios cognitivos. Algunos ejemplos recientes que merecen ser resaltados son los siguientes.

Adapt4 (www.adapt4.com) es el creador del primer verdadero radio cognitivo. Basado en su tecnología XG1, que es un dispositivo basado en software que se adapta perfectamente a la definición de un RC, le permite a RC adaptarse a su entorno en canales con licencia compartiendo el medio en el tiempo cuando estos canales no se encuentran en uso.

Vanu (www.vanu.com) ha desarrollado la primera red de acceso inalámbrica basada en software que se encuentra comercialmente disponible. Esta red permite a los operadores trabajar simultáneamente con múltiples estándares tales como GSM, CDMA, iDEN, entre otros en una sola plataforma por medio de su solución *Vanu Anywhere*.

Intel ha desarrollado la tecnología *Radio Free Intel* con la que se busca integrar todas las capacidades de radio en los nuevos chips Intel, y así fabricar radios ágiles capaces de interactuar con diferentes redes inalámbricas a través de diversos protocolos como UWB, WiMAX o GPRS (*General Packet Radio Service*) [19].

c) Otros grupos de interés.

El *SDR Forum* (www.sdrforum.org) es una asociación internacional formada por alrededor de 100 miembros de los sectores gubernamental, comercial, académico, fabricantes de equipo, y desarrolladores de software. Su finalidad es apoyar el desarrollo y la implementación de los SDRs y RCs y difundir los avances que se tienen en el desarrollo de estas tecnologías, mediante la organización de reuniones y la publicación de reportes tanto de mercado como tecnológicos.

Son muchas las universidades que juegan un rol importante en el desarrollo e investigación en RCs. Por ejemplo, la Universidad de California en Santa Bárbara, por medio del laboratorio *Lab of Intelligent Networking* (LINK) [20] mejora la eficiencia, desempeño, tolerancia a fallos, seguridad y otros aspectos de las comunicaciones inalámbricas por medio de técnicas de acceso dinámico al espectro. Otro ejemplo es la Universidad de California en Berkley, quien a través de su proyecto *Cognitive Radio Research* (CORR) [21] desarrolla radios que puedan monitorear el espectro existente e identificar las bandas de frecuencia libres.

Otras universidades que llevan a cabo investigación sobre RCs son la Universidad de Karlsruhe en Alemania, el Instituto Politécnico de Virginia en Estados Unidos y el Politécnico de Cataluña en España.

Debido a que los RCs son radios definidos por software es importante conocer las herramientas con las que se cuentan para su configuración.

The MathWorks por medio de la familia de productos *Simulink* permite simular el comportamiento de SDRs y el análisis de su desempeño. También permite la generación del código para la configuración de los SDRs y la elaboración de modelos para analizar su comportamiento [22].

En el mundo del software libre, se ha desarrollado la herramienta llamada *GNU Software Radio* [23] que proporciona la API *signal processing runtime* y los bloques de procesamiento necesarios para la implementación de SDRs. Esta herramienta es utilizada tanto para fines académicos como comerciales y es capaz de crear ambientes de pruebas como implementar sistemas inalámbricos reales.

VI. CONCLUSIONES

Los radios cognitivos prometen grandes ventajas para optimizar el uso del espectro radioeléctrico que es cada vez más demandado. Es por ello que

se ha detectado en años muy recientes una muy febril actividad en investigación y desarrollo en este tema.

El presente artículo ofrece una panorámica del estado actual de esta tecnología, desde sus principios de funcionamiento y generalidades técnicas, hasta los proyectos que se están realizando en materia regulatoria, de estandarización y de iniciativas comerciales.

Los grupos interesados en el tema, y en particular las instituciones académicas, deben seguir muy de cerca los trabajos desarrollados por los distintos comités de IEEE y ETSI, así como de las recomendaciones que serán emitidas por ITU-R en 2010.

Los radios cognitivos pueden resultar una excelente opción para detonar el desarrollo de redes de acceso en aquellos países donde éstas son escasas, o sumamente costosas debido a la falta de un entorno competitivo como ocurre, desgraciadamente, en nuestro país. Los radios cognitivos podrían contribuir seriamente a reducir la brecha digital, llevando los beneficios de la llamada Sociedad de la Información a mayores segmentos de la población.

Para ello es necesario que los órganos encargados de emitir las políticas de gestión del espectro estudien cuidadosamente los reportes de la ITU-R, FCC y de otros países en los que estas tecnologías se discuten seriamente, para así detectar y preparar los cambios necesarios que en materia regulatoria deban hacerse para permitir la coexistencia de RCs con las concesiones existentes.

VII. REFERENCIAS

- [1] I. F. Akyildiz, W. Lee, M. C. Vuran y S. Mohanty, *NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey*, Computer Networks, May. 2006.
- [2] G. Faulhaber y D. Faraber, *Spectrum Management: Property rights, markets and the commons*, Joint Center, Dic. 2002.

- [3] J. Arredondo, A. Hibert, S. Jalife y C. González, *Resolución del pleno de la Cofetel para clasificar ciertas bandas de frecuencia conforme a la Ley Federal de Telecomunicaciones*, Jul. 2005.
- [4] T. Owen, *Software-Defined radios make their move*, Mar. 2003.
- [5] *Software-defined radio*, Wipro Technologies, Ago. 2002.
- [6] S. Haykin, *Cognitive Radio: Brain-empowered wireless communications*, IEEE JSAC, 23(2) Feb. 2005.
- [7] *Establishment of an interference temperature metric to quantify and manage interference and to expand available unlicensed operations in certain fixed, mobile and satellite frequency bands*, Federal Communications Commission, Nov. 2003.
- [8] P. Kolodzy *et al.*, *Next generation communications: Kickoff meeting, DARPA*, Oct. 2001.
- [9] R. Tandra, S. M. Mishra y A. Sahai, *What is a spectrum hole and what does it take to recognize one?*, Enviado a Procs. IEEE, 2008.
- [10] T. Weiss, J. Hildenbrabd, A. Krohn, F. Jondral, *Efficient signaling of spectral resources in spectrum pooling systems*, Proceeding of the 10th symposium on communications and vehicular technology, Nov. 2003.
- [11] R. Brodersen, A. Wolisz, *et al.*, *CORVUS: A Cognitive Radio Approach for usage of Virtual Unlicensed Spectrum*, Jul. 2006
- [12] *IEEE 802.22 WRAN WG* <http://ieee802.org/22/>, Consultado en May. 2008
- [13] C. Cordeiro, K. Challapali y M. Ghosh, *Cognitive PHY and MAC Layers for namic Spectrum Access and Sharing of TV Bands*, TAPAS '06 First international workshop on technology and policy for accessing spectrum, Ago. 2006.
- [14] L. Xu, R. Tonjes, T. Paila, W. Hansmann, M. Frank, M. Albrecht, *DRiVE-ing to the internet: dynamic radio for IP services in vehicular environments*, Proceedings of the 25th Annual IEEE Conference on local computer networks, Tampa, USA, Nov. 2000.
- [15] *University of Bonn, Institute of Computer Science, Prof. Martin Research. overDRiVE: Spectrum Efficient Uni- and Multicast Services over Dynamic Multi-Radio Networks in Vehicular Environments*, en <http://web.informatik.uni-bonn.de/IV/martini/Forschung/Projekte/index-en.html#overdrive>.
- [16] B. Romano, *FCC adopts rule changes for smart radios*, FCC, Mar. 2005.
- [17] W. Wang, *IEEE Technical Sub-Committee on Cognitive Radio Networks*, TCCN en <http://www.eecs.ucf.edu/tccn/>, Dic. 2007.
- [18] *ETSI Reconfigurable Radio Systems*, ETSI, <http://www.etsi.org/WebSite/technologies/RS.aspx>, 2008.
- [19] *Integrating radio in silicon everywhere, Radio Free Intel*, Intel Corporation en <http://www.intel.com/technology/techresearch/radio/index.htm>.
- [20] *Lab of Intelligent Networking*, en <http://www.cs.ucsb.edu/~htzheng/cognitive>.
- [21] *Cognitive Radio Home*, en <http://bwrc.eecs.berkeley.edu/Research/Cognitive/home.htm>.
- [22] *Developing Software Defined Radio Systems Using MATLAB® and Simulink®*, en <http://www.mathworks.com/company/events/webinars/wbnr30307.html?id=30307&p1=34318&p2=34320>.
- [23] *GNU Radio – GNU FSF Project*, Free Software Foundation Inc. en <http://www.gnu.org/software/gnuradio>, Mar. 2007.

Análisis Paramétrico del Modelo de Capa Acoplada Perfectamente en el Método de Diferencias Finitas en el Dominio del Tiempo

Miriam Nieto-Rodríguez⁽¹⁾, Alberto Manuel Benavides Cruz⁽¹⁾,
Mauro Enciso-Aguilar⁽¹⁾, Jorge Sosa-Pedroza⁽¹⁾

⁽¹⁾Instituto Politécnico Nacional, ESIME-SEPI, Departamento de Telecomunicaciones,
UPALM Edif. Z-4. 3er. piso. 07738. México. D.F.

RESUMEN— En este artículo presentamos un análisis de los parámetros que definen la distribución de conductividad eléctrica de la Capa Perfectamente Acoplada. Actualmente la mejor técnica para resolver el problema de las condiciones de frontera en el MDFDT es el modelo de Capa Perfectamente Acoplada, PML (*Perfect Matched Layer*) de Jean Berenger, pues presenta reflexiones sumamente pequeñas. La condición de Berenger define un nuevo espacio computacional alrededor del espacio de interés cuyas características reducen las ondas computacionales entrantes casi a cero, evitando cualquier reflexión de regreso al dominio computacional principal

Los resultados muestran que la obtención de los valores para estos parámetros que brinden niveles más bajos de reflexión permite simular espacios de propagación electromagnética infinitos y tener un mejor aprovechamiento de los recursos de cómputo.

Palabras Clave: Dominio computacional, MDFDT, PML

ABSTRACT—The parametric analysis of the electrical conductivity distribution for the Perfect Matched Layer (PML) method, is discussed in this communication. Currently the best technique to solve the problem of boundary conditions in Finite Differences Time Domain (FDTD) method is PML, introduced by Jean Berenger, since this technique presents reflections extremely small. The Berenger's PML defines a new computational space around the area of interest yielding to reduce any undesirable reflection that may perturb

the main computational domain.

The optimal parameters values for the conductivity distribution are obtained in order to provide lower levels of reflection, enabling us to model infinite electromagnetic propagation spaces and as a consequence, to optimize the computing resources.

Keywords — Computational Domain, MDFDT, PML

I. INTRODUCCIÓN

A partir de 1990, el Método de Diferencias Finitas en el Dominio del Tiempo MDFDT se usa como una herramienta importante para el modelado de variados problemas científicos y de ingeniería relacionados con ondas electromagnéticas [4].

Uno de los principales problemas que se deben enfrentar al hacer uso del MDFDT, en el modelado de áreas abiertas, son las reflexiones causadas una vez que la onda llega al final del mallado de simulación, pues se cuenta con una región de cálculo limitada con recursos computacionales finitos. Se sabe que entre más grande sea el dominio computacional a simular, se hace necesario el uso de herramientas más sofisticadas y costosas [5]. Para afrontar esta dificultad se hace uso de las condiciones de frontera adecuada para simular una extensión infinita, para que la solución sea válida en todo tiempo.

Desde la aparición del MDFDT de Kane Yee [2] se ha dado gran consideración a las condiciones de frontera absorbentes, desde las técnicas usadas por Engquist y Magda [7] aplicadas a las ondas acústicas y después la técnica de Gerrit Mur [6] aplicada a las ondas electromagnéticas.

Jean-Pierre Berenguer presentó en 1994 una propuesta, titulada “Capa Perfectamente Acoplada para la Absorción de Ondas Electromagnéticas” [1], (PML, *Perfectly Matched Layer*) la cual está basada en la descomposición del campo eléctrico o magnético en la región de absorción, asignándole pérdidas a cada uno de estas subcomponentes. Berenger derivó una novedosa formulación de las ecuaciones de Maxwell, donde cada componente vectorial se divide en dos componentes ortogonales [5].

La técnica utiliza una capa absorbente acoplada al espacio computacional principal, designada para absorber sin reflexión las ondas electromagnéticas salientes, de tal forma que el factor de reflexión de una onda plana sea nulo, independientemente de la frecuencia de la onda o del ángulo de incidencia al momento de interactuar con la frontera [3].

II. EL MEDIO PML

Ahora se define el medio PML en el modo TE. El punto clave de la definición es la descomposición de la componente de campo magnético H_z en dos sub-componentes que se denotan H_{zx} y H_{zy} . En el modo TE, un medio PML se define como un medio en el que el campo electromagnético tiene cuatro componentes, E_x , E_y , H_{zx} , H_{zy} , relacionadas a través de las siguientes ecuaciones:

$$\varepsilon_0 \frac{\partial E_x}{\partial t} + \sigma E_x = \frac{\partial(H_{zx} + H_{zy})}{\partial y} \quad (1)$$

$$\varepsilon_0 \frac{\partial E_y}{\partial t} + \sigma E_y = -\frac{\partial(H_{zx} + H_{zy})}{\partial x} \quad (2)$$

$$\mu_0 \frac{\partial H_{zx}}{\partial t} + \sigma_x^* H_{zx} = -\frac{\partial E_y}{\partial x} \quad (3)$$

$$\mu_0 \frac{\partial H_{zy}}{\partial t} + \sigma_y^* H_{zy} = \frac{\partial E_x}{\partial y} \quad (4)$$

donde los parámetros $(\sigma_x, \sigma_x^*, \sigma_y, \sigma_y^*)$ son homogéneos para las conductividades eléctricas y magnéticas. Una consecuencia importante de esto es que se puede ver al medio PML como una generalización de diferentes tipos de medio, dependiendo de la combinación de estas conductividades, si son iguales a cero o si son iguales entre sí, como se explica más ampliamente en [1].

Aún más, si la condición

$$\frac{\sigma}{\varepsilon_0} = \frac{\sigma^*}{\mu_0} \quad (5)$$

se satisface, entonces la impedancia de este medio es igual a la del vacío y no existirá ninguna reflexión cuando una onda plana se propaga normalmente a través de una interface entre el medio y el vacío, o cualquier tipo de medio bajo estudio.

Ahora bien, la clave de la implementación de la técnica PML es que las conductividades eléctrica y magnética varían con respecto a la distancia que existe entre la capa y la interfaz existente con la malla principal [10]. Retomando la ecuación para la distribución de la conductividad eléctrica se tiene que:

$$\sigma(\rho) = \sigma_m \left(\frac{\rho}{\delta}\right)^n \quad (6)$$

donde σ_m es la conductividad máxima, ρ es la distancia entre la capa y la interfaz en una capa dada y δ es la distancia total entre la interfaz y la última capa, n nos indicara la forma en que varia la conductividad, ya que es el grado del polinomio de distribución de conductividad eléctrica (si $n=0$, la conductividad es constante a lo largo de toda la capa; si $n=1$, la conductividad

varía linealmente; si $n=2$ o superior, la conductividad tiene una variación no lineal).

III NECESIDAD DEL ANALISIS

En este apartado, se mostrará la necesidad del estudio de los parámetros de distribución de conductividad eléctrica y magnética σ_m , n y δ , se verá cómo estos valores determinan el éxito de la implementación de las condiciones absorbente de frontera PML.

En la figura 1 se ve que la forma de la onda no es respetada, pues existen reflexiones considerables, así también en la figura 2 se ve claramente cómo en las celdas correspondientes a la capa PML es donde se incrementa la reflexión, sobre todo en las últimas celdas, además de que el problema se acentúa en las esquinas. En la gráfica de contorno (fig. 2) deberían apreciarse círculos concéntricos de diferentes niveles; sin embargo las circunferencias son deformadas y las pequeñas áreas entre círculos no son deseadas.

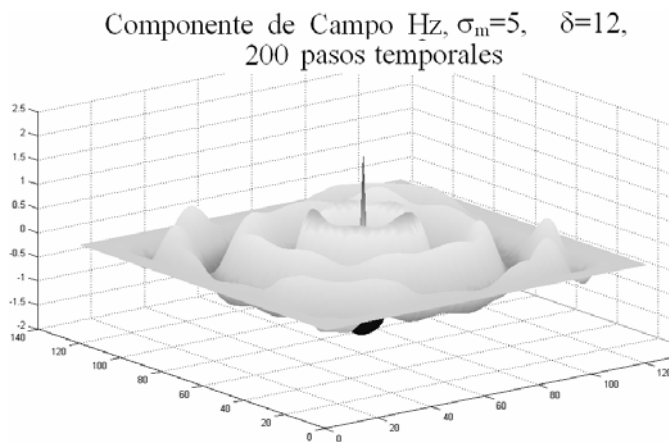


Fig. 1 Campo Hz en espacio libre para 200 pasos temporales. Se hace uso de la técnica de capa PML con $\sigma_m=5$, $n=1$ y $\delta=12$.

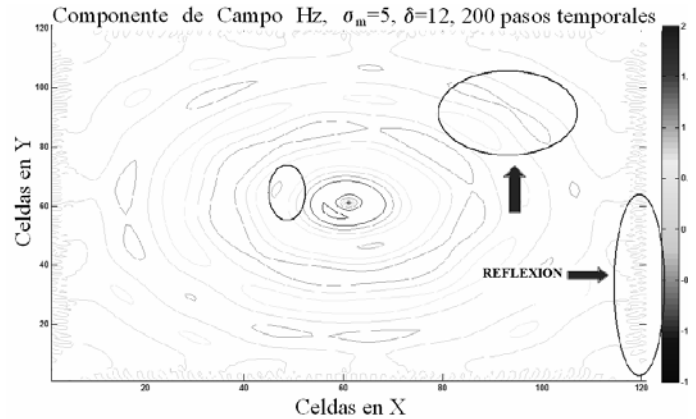


Fig.2. Gráfica de contorno para el campo Hz en espacio libre para 200 pasos temporales. Se hace uso de la técnica de capa PML con $\sigma_m=5$, $n=1$ y $\delta=12$.

En el conjunto de figuras 3 y 4, con los valores de $\sigma_m=1$ y $n=3$, $\delta=12$, se puede ver una gran mejoría con respecto a los casos anteriores, pues los círculos en la gráfica de contorno son concéntricos, conservando su forma ideal, además de que los lóbulos de reflexiones son muy reducidos, además la reflexión a los costados es nula y no existe el problema de las esquinas.

Con estos dos juegos de parámetros se demuestra que la obtención de los mismos es de suma importancia para el correcto uso de la técnica PML.

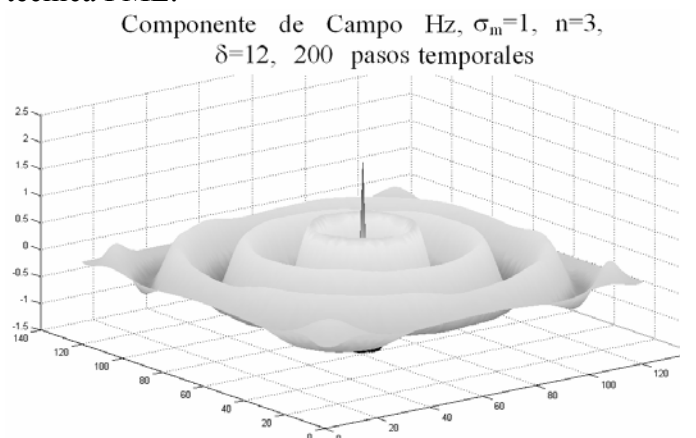


Fig. 3. Campo Hz en espacio libre para 200 pasos temporales. Se hace uso de la técnica de capa PML con $\sigma_m=1$, $n=3$ y $\delta=12$.

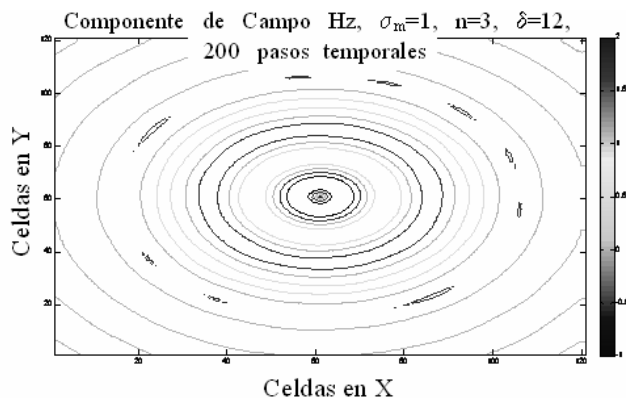


Figura 4. Gráfica de contorno para el campo Hz en espacio libre para 200 pasos temporales. Se hace uso de la técnica de capa PML con $\sigma_m=1$, $n=3$ y $\delta=12$.

IV RESULTADOS DE LA MEDICIÓN DE LA REFLEXIÓN

Para analizar el desempeño de las capas PML se tomó como referencia una región cuadrada ($\Delta=\Delta x=\Delta y$) rodeada por una capa PML con un variado valor para δ . El estudio se realizó para un modo de propagación transversal eléctrico TE reducido a dos dimensiones y propagándose en el sentido z positivo.

Se utilizó una fuente senoidal puntual centrada en la componente de campo magnético Hz ubicada en medio de la región de cálculo, operando a una frecuencia arbitraria de 6 GHz. El medio utilizado en el dominio computacional principal es el vacío, por lo tanto: $\Delta = 2.5\text{mm}$

Para la elección del escalonado temporal Δt , se retoma la *Condición de Courant*, [5], $\Delta t = \Delta / 2c_0 = 4.16\text{pseg}$. Para la determinación del tamaño total del dominio entero de cálculo, se tomo en cuenta una región principal cuadrada de 96x96 celdas (sin células extras para la capa PML), así el dominio computacional total fue variando de acuerdo al número de celdas PML requerido.

La siguiente ecuación, es una medida del error en el dominio computacional bajo estudio, como

una función del tiempo, es decir, nos indica cómo va variando el error en el dominio computacional en cada paso temporal [8].

$$L^2 = \sum_{i=1}^{i_{\max}} \sum_{j=1}^{j_{\max}} [H_z(i, j) - H_{zr}(i, j)]^2 \quad (7)$$

La ecuación (7) nos indica un índice de error en *todo el dominio bajo estudio*, la diferencia que hay entre el campo bajo estudio utilizando PML ($H_z(i, j)$) y un campo ideal, sin reflexiones de ningún tipo, donde no exista una capa PML ($H_{zr}(i, j)$), para cada una de las celdas.

Para la obtención para los mejores valores de n , σ_m y δ , se procede de la siguiente manera: se fija un valor arbitrario para σ_m y δ y se varía n , observándose el comportamiento y determinando qué valor es el mejor, el que proporciona menor nivel de reflexión. Con un valor fijo de n , se varía σ_m y se mantiene una δ . Encontrándose los mejores valores, se aprecia el comportamiento para un número de δ dado.

El efecto de una distribución de conductividad eléctrica $\sigma(\rho)$ constante es apreciado en la figura 5, donde un grado de polinomio $n=0$ es incluido en las curvas de comparación del error global, lo que provoca un niveles considerablemente más grandes que con distribuciones lineales ($n=1$) o no lineales ($n \leq 2$).

En la literatura disponible sobre el MDFDT no se han manejado valores no enteros para el polinomio de distribución, en el presente trabajo se emplearon con el fin de obtener una aproximación más exacta, encontrándose que el error efectivamente disminuye, **Se concluye así que el valor óptimo para $n=3.3$ o $n=3.4$** , pues en las pruebas realizadas, se comprobó ese comportamiento, lo que puede apreciarse en la figura 6. Los resultados de los experimentos, utilizaron una $\delta=12$ y una $\sigma_m=1.0$.

El siguiente experimento consiste en mantener $\delta=12$ y se escoge el valor de $n=3.4$, y ahora se variara σ_m para encontrar el mejor valor. Para empezar se realizaron variaciones para los primeros valores entero ($\sigma_m=1,2,3,4$) y para $\sigma_m=0.5,4.5$, pues σ_m muy grandes provocan mucho más reflexión.

La figura 7 nos indica que el mejor valor para $\sigma_m=1.0$, aunque se observa en la figura 8 que una mejor aproximación es el valor $\sigma_m=0.9$. Se observa que los valores que nos arrojan la reflexión más baja es ligeramente menor a uno, por lo que se puede considerar el valor para $\sigma_m=1.0$, todo depende del grado de aproximación que se desee.

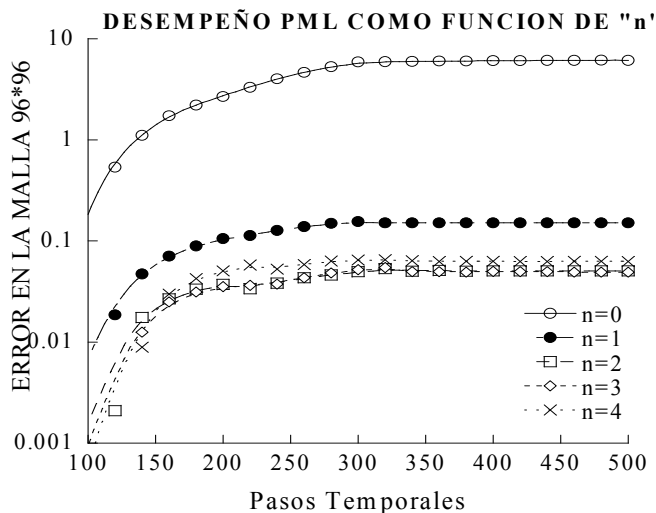


Fig. 5 Desempeño en función del grado del polinomio, incluyendo el valor de $n=0$ en escala logarítmica

Ahora bien, con los valores de σ_m y n definidos, el numero de capas PML δ es variado. Como lo indica la figura 9, al aumentar el número de capas, se reduce grandemente el error, mas de 5 veces para el caso de $\delta=16$ y $\delta=20$.

Berenger señala que el número de capas PML δ es el factor que influye significativamente en la disminución del error, ya que permite tener un aumento más lento de $\sigma(\rho)$, y por lo tanto un mejor acoplamiento de los medios bajo estudio en

el mallado DFDT principal (en el experimento el vacío) y la capa PML.

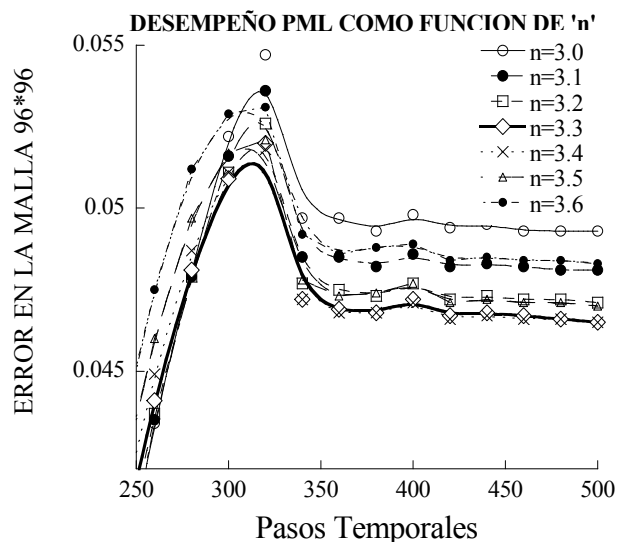


Fig. 6. Desempeño en función del grado del polinomio, con valores de n no enteros

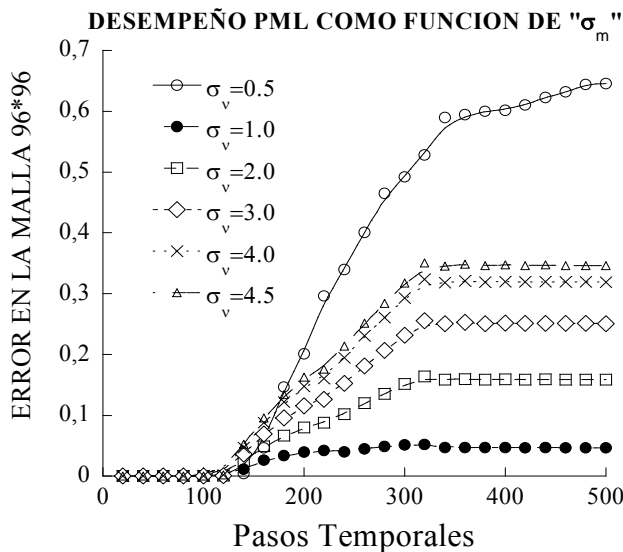


Fig 7. Desempeño en función del valor de σ_m

El aumento en el número de capas PML demanda el incremento de memoria, lo que puede llegar a representar un inconveniente, por esta razón se aborda primeramente la optimización los valores para n y σ_m que disminuyan el error sin tener la necesidad de usar este recurso, además de

que no es el único que afecta el error y la reflexión.

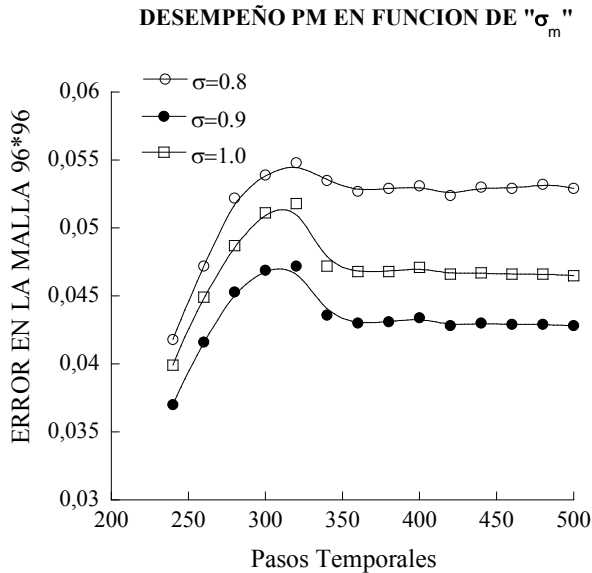


Fig. 8 Desempeño en función del valor de σ_m ,

con valores decimales

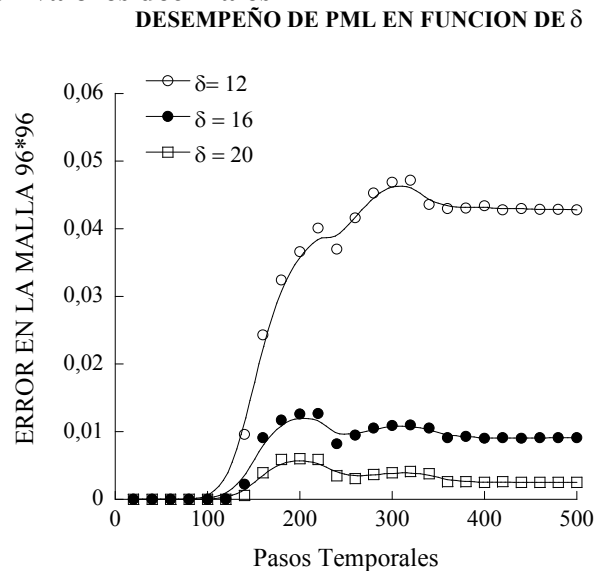


Fig. 9 Desempeño de PML en función de δ

V CONCLUSIÓN

La utilización del modelo de condición absorbente PML disminuye considerablemente las reflexiones provocadas por la incidencia de las ondas sobre la frontera de simulación. Esto permitirá que se puedan estudiar fenómenos electromagnéticos sin la necesidad de dominios computacionales extensos, además de poder contar con resultados más exactos y confiables.

De manera más recalable, un análisis paramétrico de la función de distribución de conductividad magnética para los parámetros que determinan la eficiencia de la capa PML, n , σ_m y δ arrojaron valores óptimos para el modelo de Capa Acoplada Perfectamente en dos dimensiones, pudiendo obtenerse valores muy pequeños de reflexión, menores al 1%; sin embargo se debe notar que si bien Berenger presenta valores aún más pequeños, en su análisis no utilizó una fuente senoidal continua sino un pulso no continuo compuesto de tres frecuencias, lo que hace que el error no se acumule ni se sume con las ondas emergentes de la fuente. Esto provoca que el error se vaya acumulando, aunque se reconoce un comportamiento estacionario en el que el error es constante no importando el paso temporal en que esta transcurriendo la simulación.

REFERENCIAS

- [1] Jean Pierre Berenger, "A Perfectly Matched Layer for the Absorption of Electromagnetic Waves", Journal of Computational Physics, vol. 114, No. 2, pp 185-200, Oct. 1994
- [2] Kane S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media, IEEE AP-S Transactions", Mayo 1966.
- [3] Allen Taflove, Morris Brodwin, "Numerical Solution of Steady-State

- Electromagnetic Scattering Problems Using The Time- Dependent Maxwell's Equations*", IEEE MTT, 1975.
- [4] Allen Taflove, Susan C.Hagness "Computational Electrodynamics The Finite Difference Time Domain Method", Artech House, 2000.
- [5] Sullivan, Dennis Michael, "Electromagnetic simulation using the FDTD method", New York: IEEE Press, 2000
- [6] G. Mur, "Absorbing Boundary Condition for the finite-difference approximation of the time-domain electromagnetic field equations", IEEE trans Electromagnetic Compatibility, vol EMC-23, pp. 377-382, Nov. 1981.
- [7] Enquist, B., and Majda, "Absorbing Boundary Conditions for the Numerical Simulation of Waves," Mathematics of Computation, vol 31, pp. 629-651, 1977.
- [8] Jeffrey G. Blaschak, Gegrory A. Kriegsmann "A Comparative Study of Absorbing Boundary Conditions" Journal of Computational Physics vol. 77, pp. 109-139, 1988
- [9] Alberto M. Benavides C., "Análisis de Campo en Guías de Onda y Resonadores Rectangulares en el Modo TE Utilizando el Método de Diferencias Finitas en el Dominio del Tiempo", Tesis de Maestría, SEPI-ESIME, Marzo 2001.
- [10] C. M. Calderón-Ramón, *Análisis de Campo en Guías de Placas Planas Paralelas utilizando el Método de Diferencias Finitas en el Dominio del Tiempo*. Tesis de Maestría en Ciencias, SEPI-ESIME IPN, Febrero 2007

“LEY FEDERAL DE TELECOMUNICACIONES Y PERDIDA DE LA ORBITA GEOESTACIONARIA 109.2° OESTE”

Cintya Maigre Pérez¹, Sergio Viñals Padilla², Sergio Vidal Beltrán³

¹ Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Zacatenco. Edificio Z-4, 3er Piso Colonia Lindavista, México DF

² Av. Zempoaltecas, esq. Manuel Salazar, Col. Ex-hacienda el Rosario, Delegación Azcapotzalco, 02420, México D.F.

³ Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Zacatenco. Edificio Z-4, 3er Piso Colonia Lindavista, México DF

Teléfono (55) 57296000 ext 54756 Celular (55)15321822 E-mail: cintya_maiper@hotmail.com

RESUMEN.- Debido a que la sociedad continuamente muestra el interés que tiene por los avances e innovación tecnológica que presenta el país y el resto del mundo, en cuanto a telecomunicaciones se refiere. Sin embargo, es importante hacer consciente a la misma sociedad sobre el daño que se le hace a la tecnología si se falta al cumplimiento de las leyes que rigen las telecomunicaciones del país. Por lo cual, este trabajo muestra mas a fondo los beneficios y las restricciones principales que presenta ‘La ley federal de telecomunicaciones’ en las concesiones satelitales, enfocado a la perdida de la orbita geoestacionaria 109.2°.

ABSTRACT.- Because the society continually sample the interest that has for the advances and technological innovation that it presents the country and the rest of the world, as for telecommunications refers. However, it is important to make aware to the same society on the that is harmed to the technology if it is missed the execution of the laws that they govern the telecommunications of the country. Reason why, this work shows but thoroughly the beneficioses and the main restrictions that it presents ‘The federal law of telecommunications’ in the concessions satelitales, focused to the lost one of it orbits her geoestacionaria 109.2°.

PALABRAS CLAVE

Satélite, orbita, ley, telecomunicaciones, UIT.

I. INTRODUCCION

El inicio de las telecomunicaciones se debe a un acto tan simple, pero tan importante para la comunicación. El 24 de mayo de 1844, Samuel Morse realizo el envió del primer mensaje público a través de el enlace telegráfico entre Washington y Baltimore. De aquí, que se comienza la expansión de los enlaces telegráficos no solo en el país, sino en el resto del mundo. Provocando que se generasen leyes adecuadas a las restricciones y beneficios que traería tal avance tecnológico.

Es así, como cada uno de los países antes de generar sus propias leyes, debieron generar una ley que funcionara en su totalidad para la sociedad mundial. Basados en esto, el 17 de mayo de 1865, después de dos meses y medio de negociaciones, 20 estados miembros fundadores firman en París el primer Convenio Telegráfico Internacional y crean la Unión Telegráfica Internacional, con objeto de facilitar posibles modificaciones posteriores a este acuerdo inicial. Hoy, luego de aproximadamente 140 años, los motivos que llevaron a la creación de la UIT siguen siendo de actualidad y los objetivos fundamentales de la organización son básicamente los mismos.

Para 1940 México crea lo que es las Ley de Vías Generales de Comunicación; en ella se describe de manera generalizada las restricciones que

tienen los concesionarios o permisionarios, sin un enfoque específico sobre un medio; es decir, en la actualidad cada medio de comunicación tiene una ley específica que dictamina su regulación.

En el caso de los satélites, contamos con la ‘Ley Federal de Telecomunicaciones’. Ley publicada en el diario oficial de la federación el 7 de junio de 1995. Durante la presidencia de la república del C. Ernesto Zedillo Ponce de León, presidente constitucional de los Estados Unidos Mexicanos, quien hace saber al pueblo; que el H. Congreso de la Unión, dirige el decreto de La Ley Federal de Telecomunicaciones.

II. DESARROLLO

Satélites

El tratamiento de este documento esta basado en el estudio de las leyes enfocadas a los satélites. Por tal motivo, a continuación se describen algunas características de los satélites con que cuenta México, y en que orbitas se encuentran.

El 17 de junio de 1985 desde Cabo Cañaveral se realiza el lanzamiento del satélite Morelos I. El satélite alcanza exitosamente su posición orbital 113° Oeste, con cobertura nacional. El 27 de noviembre del mismo año y en el mismo lugar, se realiza el lanzamiento del satélite Morelos II. El satélite ocupa la posición orbital 116.8° Oeste, con cobertura nacional.

Para 1993, el 17 de noviembre se realiza el lanzamiento del satélite Solidaridad 1. El satélite alcanza exitosamente su posición orbital 109.2°W Oeste. El 17 de octubre del siguiente año, se lanza el Solidaridad 2 desde Kourou, Guyana Francesa. Y de igual manera el satélite alcanza exitosamente su posición orbital 113° Oeste.

Para 1995, se inicia el proceso hacia la privatización de la Sección de Servicios Fijos Satelitales de Telecom (hoy Satmex). Entra en vigor la Reforma a la Ley de Telecomunicaciones.

El 5 de diciembre de 1998 el satélite Satmex 5 es lanzado al espacio desde Kourou, Guyana Francesa.

Para el 2001 se comienza la construcción del satélite Satmex 6, que sería lanzado a principios del 2003 en la posición orbital de 109.2° W. sin embargo, el 27 de mayo del 2006 se lanza el satélite Satmex 6 para ocupar la posición 113 ° Oeste. El satélite Solidaridad 2 se migra a la posición 114.9 ° Oeste.

Actualmente, México cuenta con tres satélites puestos en orbita por parte de SATMEX. El primero es Solidaridad II, este satélite se encuentra en una orbita geoestacionaria (GEO) a 114.9° Oeste.

Este satélite cuenta con un total de 48 transpondedores equivalentes en las bandas de frecuencias **C y Ku**. Con una cobertura y patrón de radiación en:

- Banda C: México, sur de los Estados Unidos, Caribe, Centro y Sudamérica.
- Banda Ku: México, costa este de los Estados Unidos, y las ciudades de San Francisco y Los Ángeles.

En la figura 1, se muestra la huella de cobertura más amplia con que cuenta este satélite:

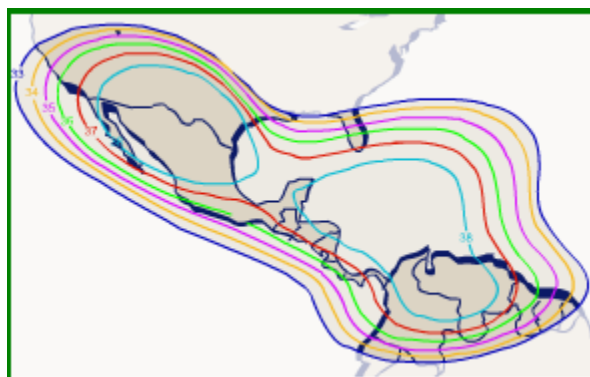


FIG. 1

El Solidaridad II tiene una vida media de 14 años, a partir de diciembre de 1994. Y uno de sus

principales beneficios es que tiene conectividad entre los principales nodos de los Estados Unidos y Latinoamérica.

El Satmex 5 se encuentra en una órbita geoestacionaria a 116.8° Oeste. Su diseño lo dota con más de 7,000 watts de potencia para la operación de la carga útil. Usando las bandas de frecuencia C y Ku. Las bandas anteriores abarcan las siguientes coberturas:

- Banda C: Continental
- Banda Ku: NAFTA (zona comprendida entre México, Estados Unidos y Canadá), y continental

En la figura 2, se muestra la huella de cobertura más amplia con que cuenta este satélite:

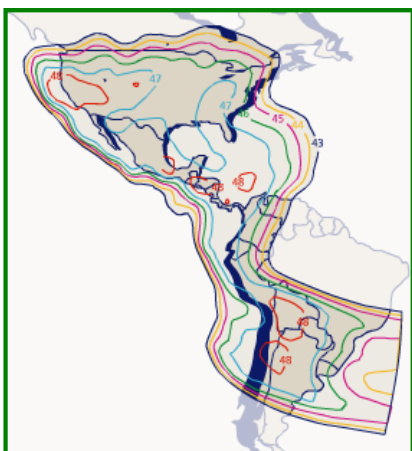


FIG. 2

Tiene una vida media de más de 15 años, contando desde enero de 1989. Sus principales beneficios son, que cuenta con una reducción en el tamaño de sus antenas y que la huella de la banda Ku esta totalmente dedicada a Norteamérica.

El satélite Satmex 6 se encuentra ubicado en una órbita geoestacionaria (GEO) a 113° Oeste.

El Satmex 6 es un satélite moderno que por su diseño ofrece grandes beneficios. Cuenta con un total de 60 transpondedores que dan servicio en

las bandas C y Ku. Las bandas anteriores abarcan las siguientes coberturas:

- Banda C: Estados Unidos, Sudamérica y continental
- Banda Ku: NAFTA y continental, con un haz de alta potencia sobre las principales ciudades de Sudamérica

En la figura 3, se muestra la huella de cobertura más amplia con que cuenta este satélite:



FIG. 3

La vida media con que cuenta es de más de 15 años. Algunos beneficios adicionales son, la reducción del tamaño de sus antenas y que sus coberturas son configurables.

En la tabla 1, se presenta una tabla en la que se describe de forma más general las ubicaciones de los satélites.

NOMBRE	FECHA DE LANZAMIENTO	POSICIÓN	ACTUALMENTE EN ACTIVIDAD
MORELOS I	17 DE JUNIO DE 1985	113° W	<input checked="" type="checkbox"/>
MORELOS II	27 DE NOVIEMBRE DE 1985	116.8° W	<input checked="" type="checkbox"/>
SOLIDARIDAD I	17 DE NOVIEMBRE DE 1993	109.3° W	<input checked="" type="checkbox"/>
SOLIDARIDAD II	17 DE OCTUBRE DE 1994	113° W	EMIGRO 114.9° W EN 2006 <input checked="" type="checkbox"/>
SATMEX 5	5 DE DICIEMBRE DE 1998	116.8° W	<input checked="" type="checkbox"/>
SATMEX 6	27 DE MAYO DE 2006	113° W	<input checked="" type="checkbox"/>

TABLA 1

Ley Federal de Telecomunicaciones

Es importante que cuando uno se dispone a leer una ley; en este caso, La ley federal de telecomunicaciones se comprendan en su totalidad cada uno de los términos a emplear durante toda la revisión. Por tal motivo, uno de los primeros artículos se encarga de la descripción de los términos mas empleados, como son:

- Banda de frecuencias: porción del espectro radioeléctrico que contiene un conjunto de frecuencias determinadas;
- Espectro **radioeléctrico**: el espacio que permite la propagación sin guía artificial de ondas electromagnéticas cuyas bandas de frecuencias se fijan convencionalmente por debajo de los 3,000 GHz;
- Estación terrena: la antena y el equipo asociado a ésta que se utiliza para transmitir o recibir señales de comunicación vía satélite;
- Orbita satelital: trayectoria que recorre un satélite al girar alrededor de la tierra;
- Posiciones orbitales geoestacionarias: ubicaciones en una órbita circular sobre el Ecuador que permiten que un satélite gire a la misma velocidad de rotación de la tierra, permitiendo que el satélite mantenga en forma permanente la misma latitud y longitud;
- Red de telecomunicaciones: sistema integrado por medios de transmisión,
- Sistema de comunicación vía satélite: el que permite el envío de señales de microondas a través de una estación transmisora a un satélite que las recibe, amplifica y envía de regreso a la Tierra para ser captadas por estación receptora, y
- Telecomunicaciones: toda emisión, transmisión o recepción de signos, señales, escritos, imágenes, voz, sonidos o información de cualquier naturaleza

El objetivo de la creación de esta ley es promover un desarrollo eficiente de las telecomunicaciones, para garantizar la soberanía nacional; así como, fomentar una sana competencia entre los diferentes prestadores de servicios de telecomunicaciones.

Los privilegios que se manejan en esta ley están totalmente direccionados a los mexicanos; ningún extranjero tendrá el beneficio de obtener la concesión o permiso de cualquiera de nuestras radiofrecuencias. De tal manera que, solo las personas físicas o morales con nacionalidad mexicana podrán ser concesionarios o permisionarios.

Otorgamiento de concesión

En principio, se requiere concesión de la Secretaría para:

Ocupar posiciones orbitales geoestacionarias y órbitas satelitales asignadas al país, y explotar sus respectivas bandas de frecuencias, y

Para la obtención de dicha concesión, se requiere que el solicitante participe en la licitación de la misma. La descripción general y específica para la participación adecuada de dicha concesión se realiza a partir del capítulo III de la Ley Federal de Telecomunicaciones.

Las concesiones se otorgan por parte de la secretaria; para fines satelitales, las cuestiones sobre las que se realizan dichas concesiones son tales como: ocupar posiciones orbitales geoestacionarias y órbitas satelitales asignadas al país, y explotar sus respectivas bandas de frecuencias.

En el artículo 12 se reitera que las concesiones solo son otorgadas a mexicanos; pero es posible que, en dichas concesiones se cuente con participación de extranjeros, solamente si ellos cuentan con menos de un 49% de acciones.

La licitación de las concesiones se publica en el diario oficial de la federación, por parte de la secretaria. La publicación deberá presentar las bases mínimas de la licitación:

- Los programas y compromisos de inversión, de cobertura y calidad de los servicios que se pretenden prestar;
- El plan de negocios;
- Las especificaciones técnicas de los proyectos, y
- Opinión favorable de la Comisión Federal de Competencia.
- Las bandas de frecuencias objeto de concesión, sus modalidades de uso y zonas geográficas en que pueden ser utilizadas;
- El período de vigencia de la concesión, y
- Los criterios para seleccionar al ganador

Un artículo de suma importancia en esta ley es el Artículo 19, el cual, nos dice que las concesiones sobre bandas de frecuencias se otorgarán por un plazo hasta de *20 años* y podrán ser prorrogadas hasta por plazos iguales a los originalmente establecidos, a juicio de la Secretaría.

Dentro de la ley, se tratan dos términos que aparentemente podrían dar a entender que la concesión o permiso se perdió. Estos términos en realidad son: el cambio o rescate de la frecuencia; y la terminación de la concesión o permiso.

El cambio o rescate de la frecuencia, se especifica en el artículo 23; la cual, solo es rescatada en los siguientes casos:

- Cuando lo exija el interés público;
- Por razones de seguridad nacional;
- Para la introducción de nuevas tecnologías;
- Para solucionar problemas de interferencia perjudicial, y
- Para dar cumplimiento a los tratados internacionales suscritos por el Gobierno de los Estados Unidos Mexicanos.

Para estos efectos, la Secretaría podrá otorgar directamente al concesionario nuevas bandas de frecuencias.

En cuanto a la terminación de concesiones y permisos se refiere, se especifica en el artículo 37; que dice que se termina en caso de:

- Vencimiento del plazo establecido en el título o, en su caso, en el permiso respectivo;
- Renuncia del concesionario o permisionario;
- Revocación;
- Rescate, y
- Liquidación o quiebra del concesionario o permisionario

La UIT

Organización más importante de las Naciones Unidas en cuanto a las tecnologías de la información y la comunicación. La función de la UIT abarca tres sectores fundamentales, a saber: radiocomunicaciones, normalización y desarrollo.

La UIT tiene su sede en Ginebra (Suiza) y está formada por 191 Estados Miembros y más de 700 Miembros de Sector y Asociados.

Para el desarrollo del presente, se efectuara un estudio solamente en el Sector de Radiocomunicaciones.

La gestión de los recursos internacionales del espectro de radiofrecuencias y la órbita de los satélites es parte fundamental de las actividades del Sector de Radiocomunicaciones de la UIT (UIT R).

La UIT es la institución encargada de inscribir asignaciones de frecuencia, posiciones orbitales y otros parámetros de satélites, con la finalidad de evitar toda interferencia entre las estaciones de radiocomunicaciones de los distintos países miembros.

Las principales tareas del UIT R también incluyen la elaboración de normas sobre sistemas de radiocomunicaciones que garanticen una utilización eficaz del espectro de radiofrecuencias y la realización de estudios relativos al desarrollo de sistemas de radiocomunicaciones.

El Reglamento de Radiocomunicaciones, se ha revisado y actualizado periódicamente debido a la demanda de utilización del espectro. Esto es necesario por la importancia de mantener un ritmo constante en la expansión de los sistemas actuales así como de las nuevas tecnologías inalámbricas avanzadas que se están desarrollando y tienen una gran demanda del espectro.

Perdida de la orbita

La orbita geoestacionaria 109.2°, es una de las mas solicitadas dentro de la UIT; ya que esta cuenta con grandes beneficios, entre los mas destacados está el hecho de que utilizara la banda Ka, la cual provee de servicios de banda ancha y de comunicación “prácticamente” en tiempo real, es una banda que se empleara para servicios de nueva generación en cuanto a tecnología satelital se refiere, además de ser una banda *no saturada*.

Dicha la órbita satelital 109.2° oeste, que cubre todo el continente americano y el archipiélago de Hawaii.

Esta orbita fue otorgada a México, con la consigna de que antes del día 5 de marzo del 2008 la orbita estaría en uso.

Luego de obtener México la concesión de la orbita, la COFETEL publica 2 licitaciones para otorgar la concesión a empresas mexicanas; sin embargo, es declarada desierta en ambas ocasiones.

El motivo por el que la orbita se perdiese, fue que las empresas satelitales no fueron capaces de llevar a cabo la construcción de un satélite con las

características adecuadas para ofrecer los servicios ya antes mencionados.

Que implica esta situación:

- Otro país será el que ofrecerá servicios de banda ancha a todo el continente americano
- El desarrollo tecnológico del país seguirá estancado
- El país pierde miles de dólares de inversión, sin haber cumplido el objetivo, *colocar un satélite en dicha orbita*
- El país pierde millones de dólares de ganancia por los servicios que se otorgarían

III. CONCLUSIONES

Después de realizado un análisis de las leyes que rigen las telecomunicaciones a nivel mundial, en conjunta con las que rigen nuestro país. Es imprescindible que las nuevas generaciones enfocadas a las telecomunicaciones sean capaces de generar no solo innovaciones tecnológicas en nuestro país; sino que, sean conscientes de los tiempos que deben cumplirse para lograr un beneficio común y real en el país.

Es fácil creer que el lento desarrollo tecnológico que existe en nuestro país, o el hecho de que el país es tercer mundista, se deba a que el resto de los países sean mejores que el nuestro; en realidad y como se muestra en este documento, nuestro gobierno genera leyes semejantes a las del resto del mundo; con algunas excepciones, como es el caso de la duración de la concesión. Las reglas en el mundo, por parte de la UIT están puestas en igual grado para todos; solo que, nuestro país no es estricto ni para el cumplimiento de las propias.

El gobierno debe cumplir; y nosotros también debemos realizar satisfactoriamente nuestras tareas como ingenieros, propiciando el desarrollo en mayor escala.

IV. REFERENCIAS

- [1] Neri Vela Rodolfo, Comunicaciones por Satélite, Thomson, 2003
- [2] Rosado, Carlos. Comunicación por satélite. México. Editorial Limusa, México 1999
- [3] <http://www.satmex.com.mx/>
- [4] <http://profesor.sis.uia.mx/aveleyra/comunicacion/leyes/lft.htm>
- [5] <http://www.eluniversal.com.mx/finanzas/57952.html>
- [6] <http://www.telecomysociedad.blogspot.com/>
- [7] www.senado.gob.mx/gace.php?sesion=2007/08/31/1&documento=54
- [8] <http://www.itu.int/net/about/history-es.aspx>

V. BIOGRAFIAS

Cintya Maigre Pérez. Estudiante de primer semestre del Posgrado de Ingeniería en Telecomunicaciones de la SEPI-ESIME del Instituto Politécnico Nacional. Sus áreas de interés son las Redes alámbricas e inalámbricas de computadoras.



M.C. Sergio Vidal Beltrán. Coordinador del Posgrado en Ingeniería en Telecomunicaciones de la SEPI-ESIME del Instituto Politécnico Nacional. Sus áreas de especialidad son las Redes de Datos, Multiservicios IP, Sistemas Distribuidos, Comunicaciones Inalámbricas.



Ing. Sergio Viñals Padilla. Profesor de tiempo parcial del Posgrado en Ingeniería de Telecomunicaciones de la SEPI-ESIME del Instituto Politécnico Nacional. Encargado de despacho del Centro de Difusión de Ciencia y Tecnología.



DISEÑO DE UN RECEPTOR PARA PROCEDIMIENTO GENÉRICO DE ENTRAMADO (GFP) PARA REDES DE TRANSPORTE ÓPTICAS

M.C. Macías Quijas Ricardo

Área de eléctrica, electrónica y control, escuela de ingeniería
Universidad Panamericana campus Bonaterra
Josemaría Escrivá de Balaguer #101, Fracc. Rústicos Calpulli, Aguascalientes, Ags. 20290, México
Tel. (449) 9106219 Ext. 7167, correo electrónico: reacias@ags.up.mx

RESUMEN

El presente artículo muestra una breve explicación del protocolo de telecomunicaciones llamado Procedimiento Genérico de Entramado (*GFP*, por sus siglas en inglés), además se presenta una arquitectura de un receptor de flujos del mismo.

Este dispositivo se enfoca en la recepción de flujos *Ethernet* sobre redes de transporte adaptados por medio del protocolo GFP. Se presentan los puntos principales del diseño de éste circuito, su verificación y los resultados obtenidos.

Palabras clave: GFP, Protocolo, Diseño electrónico, Comunicaciones digitales, Redes de Transporte, Ethernet.

1. INTRODUCCIÓN

El Procedimiento Genérico de Entramado, GFP[1], provee un mecanismo robusto y eficiente para la adaptación de tráfico de distintos tipos y distintos clientes sobre una red de transporte óptica, como lo son la Red Óptica Síncrona (*SONET*, por sus siglas en inglés), la Jerarquía Digital Síncrona (*SDH*, por sus siglas en inglés) y la Red de Transporte Óptica (*OTN*, por sus siglas en inglés).

Este protocolo fue desarrollado en octubre del 2001 por *TIXI*, el subcomité técnico responsable del desarrollo de estándares de redes de transporte de telecomunicaciones, parte del Instituto Nacional Americano de Estándares (*ANSI*, por sus siglas en inglés), quienes resaltan como una característica notable que el procedimiento utiliza un algoritmo de alineación de tramas robusto y eficiente.

GFP define dos modos de adaptación creados para distintas aplicaciones, estos modos son llamados “Mapeo por trama” y “Transparente”.

El primero, *GFP-F*, es un protocolo de adaptación orientado a unidades de datos. Este trabaja utilizando un mapeo trama a trama para encapsular una trama o paquete cliente dentro de una trama GFP y su uso está restringido sólo al tráfico de datos. Algunas de las señales cliente que utilizan *GFP-F* son *Ethernet MAC*[2] e *IP/PPP*.

El modo transparente, *GFP-T*, es un protocolo de adaptación orientado a bloques de códigos y provee un mecanismo para facilitar el transporte de señales codificadas en bloques 8B/10B para escenarios que requieren latencias de transmisión muy bajas. Ejemplos de tales señales de cliente incluyen *Fibre Channel*, *ESCON*, *FICON* y *Gigabit Ethernet*.

GFP ofrece algunas ventajas sobre otros protocolos de adaptación de datos sobre SONET, tales como:

- Soporte de configuraciones lineales y de anillo
- Inserta pocos bytes de encabezado
- Se aplica directamente a fibras oscuras
- Usa un algoritmo de alineación de tramas robusto mediante un campo de longitud de trama protegido con un código de corrección de error CRC-16.

El procedimiento genérico de entramado considera dos grandes aspectos generales: los aspectos “comunes” y los “específicos a cliente”.

Los aspectos comunes se aplican a todos los tráficos adaptados por GFP, mientras que los aspectos específicos de GFP dependen de la señal de cliente a ser adaptada.

Los principales objetivos del presente trabajo son:

- Describir el protocolo GFP.
- Establecer los requerimientos y la arquitectura para un receptor de GFP.
- Caracterizar el diseño y su verificación.

2. DESCRIPCIÓN DEL PROTOCOLO GFP

Con el fin de diseñar un receptor GFP es necesario considerar el formato de sus tramas.

GFP utiliza tramas de longitud variable, las cuales encapsulan la señal de cliente, y a su vez son encapsuladas dentro del área de carga de una trama de transporte.

La trama GFP consiste en un *encabezado principal* y un área de carga, como se indica en la figura 1. Los primeros cuatro bytes pertenecen al encabezado principal, dos bytes para el campo indicador de longitud de área de carga (*PLI*, *Payload Length Indicator*) y dos bytes para el campo de revisión de error del encabezado principal (*cHEC*, *core Header Error Check*).

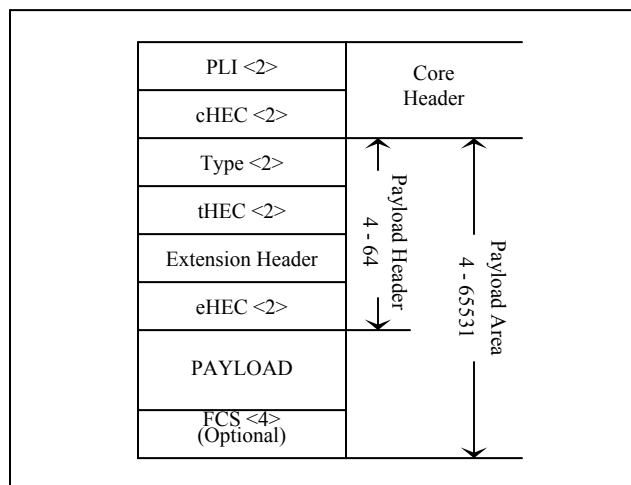


Figura 1. Trama GFP

El campo PLI contiene un número binario que representa la cantidad de bytes contenidos en el área de carga (*payload*) y está protegido por el campo cHEC, el cual contiene un código cíclico que aporta la característica de *FEC* al protocolo.

La longitud del área de carga varía entre 4 y 65535 bytes, además puede incluir un campo opcional de protección de trama, el cual contiene un código *CRC-32*, para proteger la integridad de los datos en el área de carga.

El área de carga, a su vez, contiene dos encabezados *Tipo* y *Extensión*. Además, los campos *Tipo* y *Extensión* son protegidos por medio de códigos *CRC-16* colocados en los campos *tHEC* y *eHEC*, respectivamente. Estos campos forman el encabezado de área de carga.

El encabezado del área de carga puede variar entre 4 y 64 bytes, dependiendo de la configuración del enlace, estos campos se explicarán con más detalle más adelante.

Las secuencias *CRC-16* son códigos cíclicos redundantes de 16 bits, utilizados para detectar y corregir errores sobre los bytes de datos. Estos códigos son calculados por medio de operaciones de polinomios. El circuito que calcula dichas operaciones se muestra en la figura 2, consiste en un registro de corrimiento con algunos sumadores módulo 2.

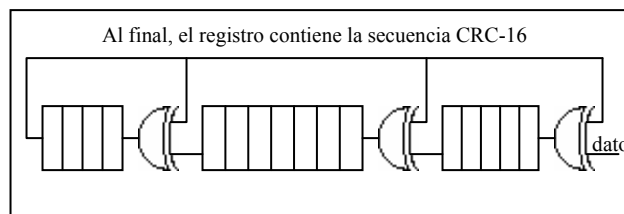


Figura 2. Circuito para el cálculo de CRC-16

GFP también define las tramas “vacías” que es un tipo especial de trama que consiste únicamente en el encabezado principal con todos sus valores en cero (0x0000) antes del proceso de aleatorización. Dichas tramas son utilizadas para rellenar los espacios entre tramas, esto para facilitar la adaptación del flujo GFP a cualquier medio de transporte dado.

El circuito para el cálculo del *CRC-16* utiliza el polinomio generador $x^{16} + x^{12} + x^5 + 1$, el cual ofrece corrección de bit sencillo y detección de bits múltiples.

3. REQUERIMIENTOS

Los requerimientos de diseño principales se listan a continuación:

- El dispositivo debe ser capaz de recibir, analizar y procesar el flujo de entrada GFP.
- El diseño deberá contener un puerto de inicialización general.
- El circuito deberá contar con un puerto de entrada configurable (*DELTA*) utilizada para cambiar el grado de robustez del algoritmo de alineación de tramas.
- El circuito deberá indicar el tipo de señal cliente contenida en la trama presente.
- El dispositivo deberá indicar la configuración y parámetros de la red correspondiente.
- La salida de datos debe ser un flujo Ethernet válido, el cual contendrá la información original de cliente, extraída del área de carga de la trama GFP.

4. ARQUITECTURA Y DISEÑO

El diagrama a bloques correspondiente a la arquitectura del receptor GFP se muestra en la figura 3. Esta arquitectura utiliza un bus de 32 bits y un reloj de 77.76Mhz, con estos parámetros el dispositivo trabaja a una velocidad de 2.5Gbps.

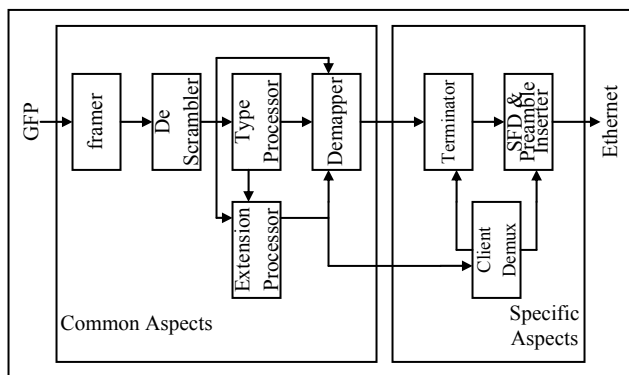


Figura 3. Arquitectura propuesta

Esta arquitectura que se propone esta claramente dividida en las dos partes generales que define el protocolo GFP:

- Aspectos comunes de GFP y
- Aspectos específicos a señales cliente, Ethernet, para éste caso.

4.1. ASPECTOS COMUNES DE GFP

Los aspectos comunes se aplican a todos los tipos de tráfico adaptados por GFP, es decir, son independientes de lo que contenga el área de carga.

4.1.1. FRAMER

La función principal de éste componente, también llamado *entramador*, es detectar los límites de las tramas GFP recibidas en el flujo, este proceso se conoce como alineación de tramas. Además utiliza un diagrama de estados para indicar el estado actual del proceso, ver figura 4.

El proceso de detectar el inicio de la trama se lleva a cabo buscando, byte por byte, un encabezado principal correctamente formato sobre la ultima secuencia de cuatro bytes recibida. Una vez que éste encabezado es encontrado, el proceso entra a un estado llamado de pre-sincronía.

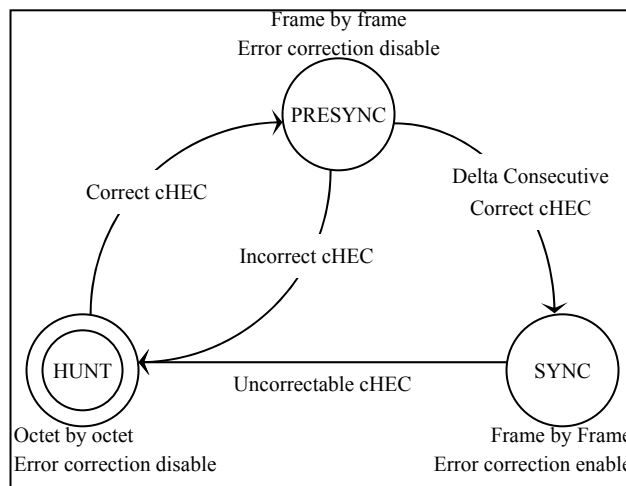


Figura 4. Maquina de estados para entramado

En éste punto, el proceso lleva a cabo la alineación de tramas revisando trama por trama que los campos cHEC sean correctos. Esto se realiza esperando una correspondencia en el presunto encabezado principal de la siguiente trama “candidato”, dicha trama se encuentra utilizando el campo PLI de la trama anterior. El proceso se repite hasta que cierto número de campos cHEC correctos consecutivos son confirmados, una vez que esto pasa se dice que el proceso esta sincronizado.

La MEF va del estado de pre-sincronía a la sincronización después de DELTA correspondencias en los encabezados principales, es decir, después de DELTA tramas GFP formateadas correctamente. El valor DELTA es un parámetro de robustez en el algoritmo de alineación de tramas y se recomienda un valor 1.

Una vez que el proceso se sincroniza, la máquina trabaja similar que en el estado de pre-sincronía, en cuyo caso la característica de corrección de error se habilita.

Si el proceso encuentra un cHEC incorrecto en los estados de pre-sincronía o sincronía, la máquina regresa a su estado inicial.

La arquitectura del entramador está compuesta por tres unidades principales, mostradas en la figura 5.

La unidad de distribución de datos toma el flujo y lo distribuye hacia la máquina de revisión del campo cHEC correspondiente, ésta decisión la toma en base a la información que obtiene del controlador.

El distribuidor de datos divide una palabra de datos en cuatro “candidatos” a encabezado principal o sólo uno dependiendo del estado actual del proceso.

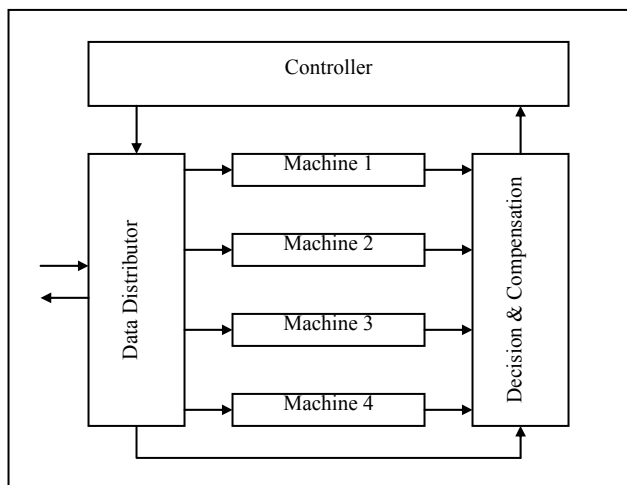


Figura 5. Arquitectura del entramador

Las máquinas de revisión de cHEC reciben un presunto encabezado principal y realizan el cálculo y comparación de los campos cHEC y PLI.

Como lo muestra la figura 5, hay cuatro instancias de esta máquina trabajando simultáneamente. Esto es necesario ya que cualquier cuarteto de bytes podría ser el encabezado principal dentro de dos flujos continuos de 32 bits.

La unidad de decisión y compensación recibe los resultados de las cuatro máquinas de revisión de cHEC además de información proveniente del distribuidor de datos para evaluarlos y tomar una decisión. Esta unidad decide si el encabezado principal actual es correcto. Además, esta unidad genera una señal de compensación para la búsqueda del siguiente encabezado principal.

El controlador contiene una implementación del diagrama de estados mostrado en la figura 4. Este calcula el estado actual del proceso haciendo uso de los parámetros entregados por la unidad de decisión y compensación. El estado actual lo reporta al distribuidor de datos para ajustar la modalidad de búsqueda.

4.1.2. DESCRAMBLER

El propósito de éste componente es realizar la *desaleatorización*, proceso inverso a la *aleatorización*.

El proceso de aleatorización tiene como finalidad garantizar una densidad de 1's en el flujo de datos de salida. Esto evita cadenas de unos o ceros consecutivos manteniendo suficientes transiciones para facilitar la función de recuperación de reloj. Esta función es realizada por un *scrambler*.

Además, el scrambler puede llevar a cabo la función de supresión de ceros tal como un código de línea. En general, la aleatorización de los datos se aplica para desempeñar las siguientes dos funciones:

- Aportar seguridad a los datos transmitidos
- Suprimir cadenas de bits consecutivos con el mismo valor.

Un aleatorizador es un generador pseudo-aleatorio implementado a partir de registros de corrimiento y sumadores módulo 2.

Para GFP la aleatorización se divide en dos procesos: aleatorización del encabezado principal y aleatorización del área de carga.

El encabezado principal es balanceado en DC por medio de una operación XOR con un número fijo de 32 bits. Este número es una secuencia “*tipo Barker*”, cuya finalidad es proveer un número máximo de transiciones y un mínimo de lóbulos laterales en el ancho de banda del flujo.

La aleatorización del área de carga es realizada por un scrambler auto-síncrono [3]. Este es un tipo especial de scrambler el cual tiene la propiedad de trabajar continuamente sin la necesidad de inicializarse en cada trama. Su polinomio generador es $1+x^{43}$.

El scrambler es habilitado al inicio del primer byte transmitido después del campo cHEC, y deshabilitado después del último byte transmitido de la trama GFP.

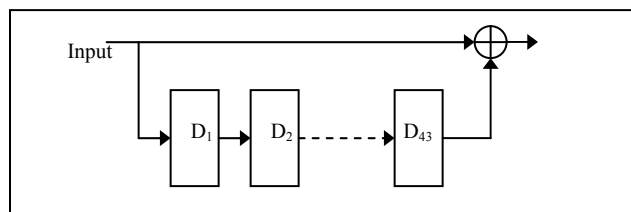


Figura 6. Desaleatorizador GFP

El *descrambler* ejecuta la operación inversa que el scrambler para decodificar los datos de entrada. Este es implementado usando el mismo polinomio generador que el aleatorizador, como se muestra en la figura 6.

En el lado de recepción el descrambler es habilitado sólo cuando el proceso de alineación de tramas está sincronizado y actúa solamente sobre los bytes contenidos en el área de carga de datos. La desaleatorización del encabezado principal es realizada antes del proceso de alineación de tramas.

4.1.3. PROCESADOR DE TIPO

Los primeros dos campos del encabezado del área de carga son los campos Tipo y tHEC, respectivamente. El campo Tipo tiene 16 bits que indican el contenido y formato del área de carga de la trama GFP. El formato del campo Tipo se ilustra en la figura 7.

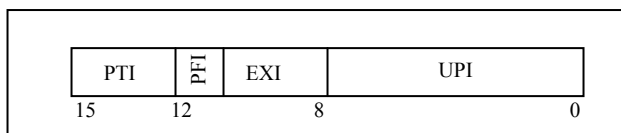


Figura 7. El campo Tipo

Como se puede ver, el campo Tipo contiene cuatro indicadores, los cuales son usados para distinguir entre servicios en un ambiente multi-servicio.

El indicador de tipo de área de carga (*PTI*) tiene valor ‘0’ para tramas de datos y ‘1’ para tramas de control, estas tramas de control se utilizan para enviar comandos, tales como una indicación o alarma de la señal de un elemento de cliente lejano. El campo *FCS* (*Frame Check Sequence*) en el área de carga de datos se asume existente siempre que el bit *PFI* tenga valor ‘1’, de lo contrario se asume que la trama no hace uso de dicha característica. El identificador de encabezado de extensión (*EXI*) contiene la configuración del enlace, a su vez esto indica la longitud y el formato del encabezado de extensión.

El identificador de área de carga de usuario indica el tipo de datos de cliente que contiene el área de carga de las tramas GFP, tales como Ethernet, PPP, FICON, etc.

El campo tHEC contiene un código CRC-16 que protege la integridad del contenido del campo Tipo.

El propósito del procesador de tipo es extraer el campo Tipo para obtener información sobre la configuración del enlace, el tipo de datos de cliente y determinar la presencia o ausencia del campo FCS en la trama GFP.

Este procesador extrae y procesa la información que ofrece el campo Tipo para ser utilizada en elementos posteriores.

4.1.4. PROCESADOR DE EXTENSIÓN

El encabezado de extensión es un campo extendido que varía su longitud entre 0 y 60 bytes. Se definen tres variantes para éste encabezado para dar soporte a redes de datos sobre configuraciones lógicas en anillo o configuraciones lógicas punto a punto.

Se define un encabezado de extensión nulo para configuraciones lógicas punto a punto. Este encabezado se implementa para escenarios donde la ruta de transporte es dedicada.

El encabezado de área de carga para una trama con encabezado de extensión nulo no contiene dicho campo.

El encabezado de extensión para tramas en configuración lineal se muestra en la figura 8a. Este se implementa para escenarios donde la ruta de transporte es compartida por varias señales de cliente y aplica a configuraciones lógicas punto a punto. El campo de identificador de canal (CID) es un número de 8 bits usado para indicar uno de 256 posibles canales de comunicación en un elemento extremo de GFP.

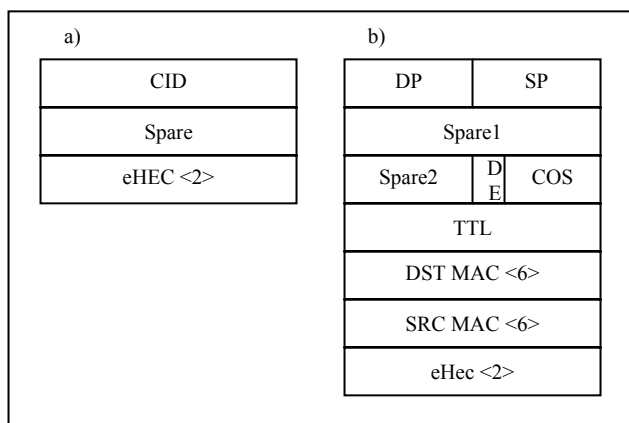


Figura 8. Encabezado de extensión:
a)lineal b)anillo

El encabezado de extensión para una configuración lógica en anillo se muestra en la figura 8b. Este encabezado consiste en 18 bytes que contienen dos

números de cuatro bits usados para indicar uno de 16 puertos destino(DP) y uno de 16 puertos origen(SP), respectivamente.

Además, el campo contiene otros componentes tales como un bit de elegibilidad para ser descartado (DE), clase de servicio (COS), y direcciones MAC destino y origen formados por 6 bytes cada uno. El campo de tiempo de vida (TTL) es un número de 8 bits que representa el número de puntos GFP que persistirá la trama. Un valor de cero indica que la unidad GFP asociada será terminada en el siguiente elemento de terminación GFP. El campo eHEC es un código CRC-16 que protege la integridad del contenido del campo de extensión.

El procesador de encabezado de extensión extrae los bytes que pertenecen a éste campo y determina el contenido de los parámetros anteriormente definidos. La información extraída de este encabezado depende directamente de la entrega del procesador de tipo acerca de la configuración del enlace.

4.1.5. DEMAPPER

La función “demapping” consiste en extraer la información original que fue encapsulada en una trama GFP, en éste caso un paquete de datos Ethernet, de cada trama. La información obtenida por éste componente incluye los bytes de Ethernet MAC desde la dirección destino hasta el campo FCS.

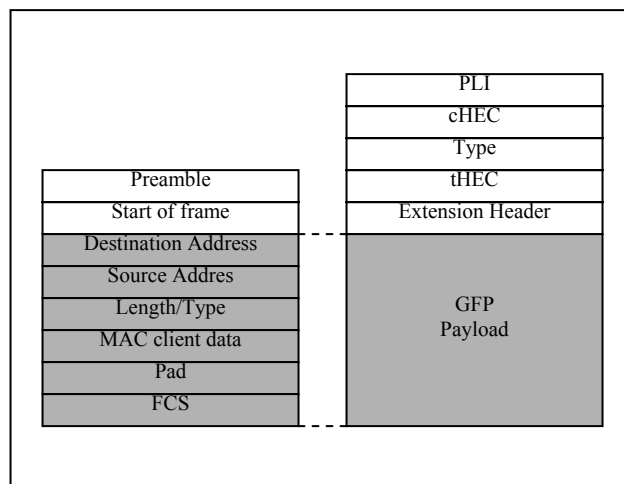


Figura 9. Encapsulado de trama Ethernet

La extracción de los datos de cliente a partir del contenedor GFP es un aspecto específico de GFP, específicamente para GFP-F.

La relación entre las tramas Ethernet MAC y las tramas GFP es ilustrada en la figura 9. Es necesario notar que los campos *Preámbulo* y *SFD* deben ser restaurados.

Otra función requerida es la restauración de los espacios entre paquetes (*IPG, Inter Pack Gaps*). Los IPG son espacios en blanco entre paquetes Ethernet, los cuales fueron suprimidos en el elemento de inicio de GFP, antes de la transmisión. Esta restauración se lleva a cabo después de que la trama es extraída del contenedor GFP por el *demapper*. Este proceso se realiza insertando bytes que contienen el patrón “00h”.

4.2. ASPECTOS ESPECIFICOS A CLIENTE

Los aspectos específicos a cliente de GFP difieren para cada tipo de señal de cliente adaptada. Estos son dependientes del área de carga.

4.2.1. MULTIPLEXOR DE CLIENTE

La actividad principal realizada por éste modulo es habilitar el hardware dedicado a realizar los aspectos específicos para cada tipo de cliente.

Este componente toma la información del tipo de cliente proveniente del procesador de Tipo y toma una decisión para habilitar el hardware correspondiente.

En este caso, éste módulo solamente validará el tipo de la señal de cliente. El módulo habilitará la parte de la arquitectura específica a cliente siempre y cuando los datos contenidos en la trama GFP correspondan al tipo Ethernet.

4.2.2. TERMINADOR DE DATOS INVALIDOS

Este módulo se encarga de rechazar las tramas inválidas que se encuentren en el flujo de datos, evitando así su transmisión a los siguientes elementos de cliente. Las tramas con errores no corregibles en algún encabezado o campo de datos son tramas inválidas. Además, el flujo se considera

inválido mientras el entramador se encuentre fuera de sincronía.

4.1.5. RESTAURADOR DE PREÁMBULO Y SFD

Los campos preámbulo y SFD no son encapsuladas en la trama GFP, éstos campos deberán ser restaurados antes de mandar la trama Ethernet hacia los elementos en la capa de cliente para su procesamiento. Para insertar esos campos es necesario un “*buffer*” para guardar temporalmente el resto de la trama Ethernet, esto permite la transmisión correcta de la trama Ethernet original y completa hacia la capa de cliente. La trama debería ser igual a la extraída a partir del flujo de datos de cliente en el lado de transmisión. El flujo de salida de éste módulo es un flujo de datos Ethernet válido.

5. VERIFICACIÓN

Tener un buen plan de verificación es una parte importante del ciclo de diseño. Esto debe ayudar a probar las funcionalidades del dispositivo bajo prueba (*DUT*).

La primera parte de la verificación es la simulación. Esto permite observar el comportamiento funcional y temporal del dispositivo y depurarlo. Como éste método es generalmente visual, es común que algunos errores persistan en el diseño.

La siguiente parte es la elaboración de un plan de verificación funcional, el cual consiste en un conjunto de casos de prueba. Para cada caso de prueba se aplica uno o más estímulos al dispositivo bajo verificación y se analiza su respuesta. En éste caso tanto el estímulo como la respuesta son manejados como archivos tipo CSV, esto se puede ver en la figura 10.

El plan de verificación se diseña con uno o más casos de prueba por cada requerimiento funcional del dispositivo. Estos requerimientos son documentados y relacionados con un archivo de estímulo.

Además es necesario implementar una cama de pruebas para aplicar el estímulo, analizar la respuesta y generar un reporte.

Los estímulos son creados por un generador de tramas GFP virtual, cuyo código se escribió en el lenguaje orientado a objetos C++.

El plan de verificación, para este dispositivo, llegó a más de doscientos casos de prueba para probar cada una de las funcionalidades de los componentes que forman la arquitectura. Cada función desempeñada por un componente debe ser asociado a uno o más casos de prueba.

La tabla 1, muestra un resumen del número de casos de prueba diseñados para cada componente. Además esta tabla muestra el porcentaje de código que se cubrió de cada componente.

Component	Test Cases (#)	Code Coverage (%)
Framer	145	99.7
Type P.	10	100
Extension P.	18	99.3
Demapper	17	96.9
Client Demux	5	100
Terminator	11	98.2
SFD Inserter	6	100
GFP Receiver TOP	13	99.1
TOTAL	225	99.1

Tabla 1. Resumen de plan de verificación

El porcentaje de cobertura de código nos proporciona una métrica de prueba que disminuye la posibilidad de pasar por alto errores funcionales, ya que nos asegura que la mayor parte de las funciones del circuito fueron probadas.

Para llevar a cabo esta medida es necesario un caso de prueba significativo, es decir, que estimule completamente al componente en una misma ejecución.

6. CONCLUSIONES

Este artículo presentó el protocolo GFP y el diseño de un dispositivo para recibir flujos de datos del mismo tipo.

La mayor contribución fue el desarrollo de una implementación funcional de un receptor de flujos GFP.

La implementación del dispositivo fue realizada en el lenguaje de descripción de hardware VHDL[4].

Se diseñó y ejecutó, además, un plan de verificación de más de doscientos casos de prueba obteniendo reportes positivos. El porcentaje de cobertura de código que permitió la cama de pruebas fue muy cercano al 100%, dando confiabilidad al diseño.

7. REFERENCIAS

- [1] Enrique Hernandez-Valencia, *T1X1.5/2001-024R4 Generic Framing Procedure (GFP)* (Holmdel, NJ: American National Standards Institute, Inc., 2001).
- [2] LAN/MAN Standards Committee of the IEEE Computer Society, *802.3 Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications* (New York, NY: The Institute of Electrical and Electronics Engineers, Inc., 2002).
- [3] Draft American National Standards for Telecommunications, *T1E1.2/93-020R2 Broadband ISDN Customer Installation Interfaces: Physical Layer Specification*.
- [4] Ulrich Heinkel, Lucent Technologies Nuremberg Germany, *THE VHDL REFERENCE, A practical Guide to Computer-Aided Integrated Circuit Design including VHDL-AMS*(New York: John Wiley & Sons LTD, 2000)
- [5] Paul Bonenfant and Antonio Rodriguez-Moral, Photuris Inc., *Generic Framing Procedure (GFP) The Catalyst for Efficient Data over Transport* (IEEE Communications Magazine, May 2002 Vol. 40 No. 5).

Bluetooth: desempeño en redes inalámbricas de área personal

Juan Iván Nieto Hipólito, Candolfi Arballo Norma, José Antonio Michel Macarty y
Elitania Jiménez García

Facultad de Ingeniería-Ensenada, UABC
{ jnieto, candolfi, michel, ejimenez @uabc.mx }

Resumen

En este trabajo se presentan los resultados de mediciones hechas al funcionamiento de Bluetooth v2.0 +EDR en aplicaciones de transferencias de archivos de tamaño variable en un ambiente indoor, donde la presencia de obstáculos es casi inevitable. Esta evaluación es importante porque indican el retardo promedio, porcentaje de pérdida y de retransmisión de paquetes. Conocer estas métricas es fundamental para el desarrollo de aplicaciones multimedia, donde se debe garantizar una tasa de retransmisiones y retardos mínimos. Los resultados muestran que el retardo promedio en la transferencia de archivos es lineal en función del tamaño del archivo sin considerar la distancia entre el par transmisor-receptor. También se observó que el desempeño de la versión 2.0 +EDR de Bluetooth no se degrada con la presencia de redes WiFi. La importancia de estudiar y analizar la tecnología Bluetooth es que se encuentra instalada en miles de millones de dispositivos.

I. Introducción

Las redes inalámbricas son cada vez más comunes en nuestro entorno diario. Hoy en día vemos redes en el hogar, oficinas e industrias donde los usuarios exigen servicios de transferencia de datos, voz y video de una manera rápida y con el mínimo de errores. Una tecnología que se ya se encuentra en los ambientes citados, es Bluetooth (BT). Bluetooth es una tecnología inalámbrica de telecomunicaciones de corto alcance, con una velocidad de transferencia de

hasta 3 Mbps en su versión 2.0+EDR (ratificada en noviembre de 2004) y cuyo objetivo es eliminar los cables en las conexiones entre dispositivos electrónicos [1]. Su versión anterior 1.2 logra transmisiones de hasta 1 Mbps. De aquí que se tiene un incremento de 3x en la velocidad de transmisión. En la referencia [3] se realiza una comparación mas profunda entre estas dos versiones de la tecnología BT.

La Tecnología BT, fue concebida por un Grupo de Interés Especial (SIG), fundado en 1998 con el fin de promover y desarrollar esta tecnología. Actualmente este SIG cuenta con más de 9.000 miembros y varias empresas promotoras. A la fecha la tecnología de BT se encuentra regulada bajo el estándar IEEE 802.15.1. La referencia [2] realiza un análisis más comprensivo del mencionado estándar.

El sistema de comunicación BT permite la conexión instantánea entre dispositivos, que se encuentran en un mismo radio (área) de operación. Las conexiones entre dispositivos BT, forman una red denominada Piconet. Una Piconet se forma con 8 dispositivos Bluetooth y se pueden enlazar hasta 10 Piconets. En una Piconet se tiene un nodo maestro y hasta 7 nodos esclavos. La unión de dos o más Piconet se conoce como una Scatternet. Para operar la tecnología Bluetooth es necesario formar al menos una Piconet, en la Figura 1 se muestra el esquema de una Piconet y Scatternet.

De la figura 1, la Piconet 1 está formada por el nodo 1 como Maestro y administrador de la red, y los nodos 2, 3 y 4 como clientes. En el caso de la Piconet 2, el nodo maestro es el nodo 5 y el nodo 4, su único nodo cliente. Por otro lado, se

ejemplifica como el nodo 4 pertenece a las dos Piconets, dando forma a lo que es una Scatternet.

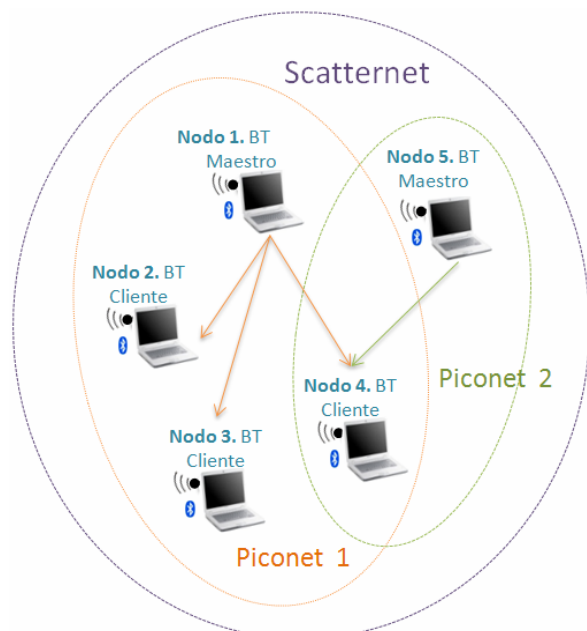


Fig. 1. Piconet y Scatternet Bluetooth.

Las redes en Bluetooth operan en una topología estrella. En donde el nodo maestro es el administrador de toda la red, y por medio de él se comunican los nodos cliente, siguiendo un protocolo de acceso tipo TDD (time division duplex). Cuando el nodo maestro de la Piconet, se desconecta, la red no se desintegra, los nodos clientes, se comunican y coordinan para asignar un nuevo maestro.

BT transmite en el Rango de Frecuencias de 2,402 GHz - 2,480 GHz. Con una potencia 1 mW para 10 metros y 100 mW para 100 metros. Utiliza la tecnología de Espectro Expandido en Secuencia Directa Híbrida SSSD y/o Saltos en Frecuencia SSFH, con el fin de evitar interferencia con las otras tecnologías inalámbricas con las que comparte esta banda. La señal salta entre 79 frecuencias en intervalos de 1 MHz para tener un alto grado de tolerancia a las interferencias.

El éxito de BT es tal que, a la fecha se encuentra embebido en alrededor de 3 mil millones de dispositivos. Entre estos dispositivos se encuentran: Teléfonos celulares, PDA's, auriculares, computadoras personales, televisiones,

entre otros. Además BT tiende a sustituir a la Tecnología IrDA (infrarrojo) [1].

Este trabajo esta organizado de la siguiente manera: la sección II describe la metodología utilizada. La sección III explica el desarrollo del experimento, muestra los resultados obtenidos, se examina el funcionamiento de BT en la presencia de redes WiFi y se realiza una comparación con resultados previos publicados para la versión 1.2 de BT. Finalmente en la sección IV se concluye el trabajo y se diserta sobre trabajos futuros.

II. Metodología de Evaluación

La Metodología de Evaluación se eligió en base a escenarios de uso cotidiano, lo cual permite darles a las mediciones un ambiente de aplicación más real. El equipo utilizado en los experimentos consistió de dos sniffers, un nodo servidor BT y 5 nodos cliente BT. Los dispositivos BT utilizados tanto para el servidor, clientes y sniffers fueron los nodos ANYCOM USB-200/250 Bluetooth 2.0+EDR. Los dispositivos ANYCOM USB-200/250 cumplen con las especificaciones BT tal como se citan en [1].

De los sniffers, uno (Sniffer A) se utilizo para la medición de paquetes Bluetooth, el cual estuvo inmerso en la Piconet, donde se efectuó la transmisión. El segundo sniffer (Sniffer B), se utilizo para detectar errores debido a interferencia de procesos externos tales como interferencia debido a transmisiones WiFi.

La computadora que alojo al *Sniffer A* se instalo el analizador de red Packet Analyzer Colasoft Capsa 6.7, el cual esta optimizado para analizar trafico BT. En la computadora que alojó el sniffer B se instaló el analizador WireShark 1.0, el cual es la nueva versión del analizador de redes Ethereal, ampliamente utilizado para el análisis de trafico WiFi. La tarjeta de medición WiFi fue Intel(R) PRO/Wireless 3945ABG.

III. Diseño del experimento

En el escenario de prueba se utilizo la métrica de retardo de transferencia de archivo (RTA). Este parámetro mide el tiempo promedio en que el

sistema Bluetooth transferirá archivos de un punto origen a un punto destino. El parámetro RTA se grafica en función de la distancia del par transmisor – receptor. Obtener estos resultados de RTA en función de la distancia es importante debido a la naturaleza inalámbrica de la transmisión, en donde el desvanecimiento de la señal puede provocar retransmisiones o pérdidas de paquetes. Las mediciones se realizaron a distancias de 3, 6 y 9 metros. La distancia de 9 metros se eligió como máxima debido a que en mediciones previas a una distancia mayor de 10 metros se observó una pérdida excesiva de paquetes, además los dispositivos utilizados especifican una distancia máxima de transmisión de 10 metros. Los tamaños de archivos utilizados fueron de 1, 3, 5, 10 y 20 Mb.

III.1 Escenario de prueba

La topología utilizada en el escenario de prueba se presenta en la figura 2.

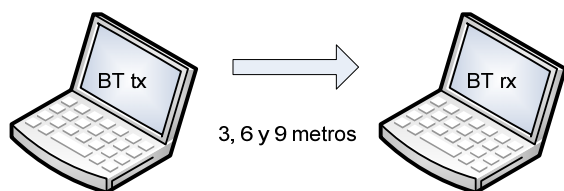


Figura 2. Topología de medición para el escenario 1.

En este escenario se transfiere un archivo de tamaño variable del Nodo 1 (BT transmisor Ttx) al Nodo 2 (BT receptor Rx). Esta transmisión se monitorea por el Sniffer A y Sniffer B, simultáneamente. Las mediciones se hicieron con obstáculos y sin obstáculos. Los obstáculos en las mediciones se deben a que Bluetooth opera generalmente en ambientes interiores, donde la presencia de barreras para la propagación de la señal es casi inevitable.

La tabla I muestra los resultados del RTA en segundos para diferentes tamaños de archivos transmitidos a diferentes distancias, sin que exista obstáculo entre el par transmisor – receptor. La figura 3 grafica estos resultados.

Distancia en metros	Tamaño del archivo en Mega Bytes				
	1	3	5	10	20
3	6.27	16.985	27.665	55.32	108.11
6	8.21	21.705	37.55	76.865	110.65
9	9.32	26.435	43.115	88.3	135.59

Tabla I. Retardo promedio en segundos para la transmisión de archivos de tamaño variable en función de la distancia **sin obstáculos** entre par Tx—Rx.

La Tabla II muestra los resultados del RTA en segundos para diferentes tamaños de archivos transmitidos a diferentes distancias, con una pared de concreto como obstáculo entre el par transmisor – receptor. La figura 4 grafica estos resultados.

Distancia en metros	Tamaño del archivo en Mega Bytes				
	1	3	5	10	20
3	7.50	18.82	35.76	81.70	137.84
6	8.96	23.76	38.42	81.10	153.70
9	12.56	32.22	51.86	99.33	202.43

Tabla II. Retardo promedio en segundos para la transmisión de archivos de tamaño variable en función de la distancia **con obstáculos** entre par Tx—Rx.

Los resultados mostrados en las tablas I y II son el promedio de realizar 30 veces la misma medición. Esto es transmitir 30 veces el archivo de un 1MBytes a una distancia de 3 metros y así sucesivamente para cada una de las mediciones. Comparando los resultados de la tabla 3 con la tabla 4 se observa un incremento en el tiempo de transferencia del archivo cuando existen obstáculos entre el par Tx (transmisor) – (Rx) receptor y sin importar la distancia en que se encuentren ambos. Este incremento se debe a que la señal

electromagnética pierde potencia al atravesar el obstáculo provocando errores en el paquete de datos y esto obliga al receptor solicitar la retransmisión de dicho paquete.

Los resultados obtenidos con una pared de concreto obstruyendo la transferencia, muestran mayor retardo que sin obstáculo, obteniéndose un incremento promedio de 1.5 segundos en archivos de 1 a 3 mega bytes, y a partir de los 5 a 20 mega bytes se observa un mayor incremento de 2 seg hasta 30 seg.

De este experimento es importante resaltar que la existencia de una pared de concreto como obstáculo puede ocasionar altas tasas de retardo. Este retardo se debe a la retransmisión de paquetes que se presenta al obstruir la transmisión, reducción de la potencia de la señal, y algunos paquetes no llegan a su destino.

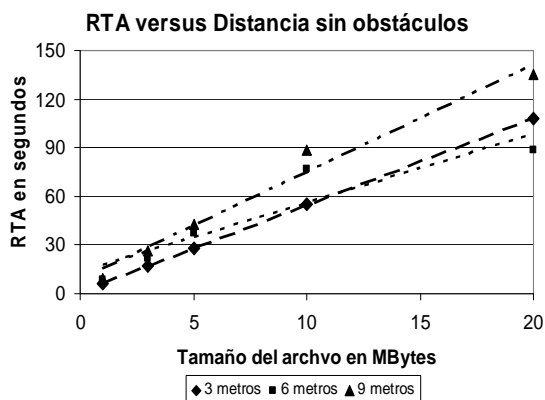


Fig. 3. RTA promedio en función de la distancia para diferentes tamaño de archivo.

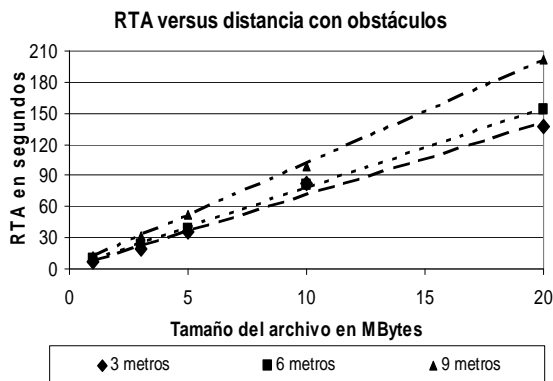


Fig. 4. RTA promedio en función de la distancia para diferentes tamaño de archivo. Con una pared como obstáculo entre el par Tx—Rx.

En la Figura 5 se muestran los retardos promedios de la transferencia de archivos en función del tamaño del archivo, ya sin tomar en cuenta la distancia. El eje de las “y” muestra el retardo promedio obtenido de promediar los valores de las columnas en las tablas I y II. Por ejemplo para transmitir un archivo de un mega bytes sin obstáculos (tabla I) se promedian los valores de $\{(6.27+8.21+9.32)/3=7.93\}$ y así sucesivamente. Este dato es importante porque en una aplicación real los usuarios no toman en cuenta la distancia para la transferencia de archivos entre dispositivos que incluyan esta tecnología.

La tabla III muestra los valores de retardo promedio en función del tamaño de paquete utilizados para graficar la figura 5.

	Tamaño del archivo en Mega Bytes				
	1	3	5	10	20
RTA sin obstáculos	7.930	21.708	36.110	73.495	118.116
RTA con obstáculos	9.670	24.933	42.013	87.376	164.656

Tabla III. RTA para diferentes tamaños de archivo sin tomar en cuenta la distancia.

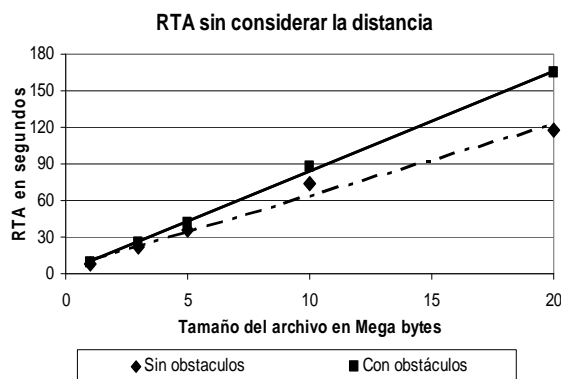


Fig. 5. RTA promedio sin considerar la distancia para diferentes tamaños de archivo. Con y sin una pared como obstáculo entre el par Tx—Rx.

La línea punteada muestra los promedios obtenidos sin obstáculos y la línea continua los promedios obtenidos con obstáculos. De la figura 5 se puede observar que el retardo promedio sin obstáculos (línea punteada) en función del tamaño del archivo sigue un modelo lineal, dado por la ecuación:

$$y = 5.8209x + 6.0682 \quad \dots (1)$$

El coeficiente de correlación obtenido de $R^2 = 0.9844$ comprueba la linealidad entre el RTA promedio en función de la distancia del par transmisor – receptor. Esta ecuación se obtuvo utilizando Microsoft Excel.

También en la figura 5 se observa que este modelo lineal se mantiene de igual manera con una pared de concreto como obstáculo entre los nodos transmisor y receptor. La ecuación:

$$y = 8.2234x + 1.5871 \quad \dots (2)$$

Modela este comportamiento y también se obtuvo utilizando Microsoft Excel. El coeficiente de correlación obtenido de $R^2 = 0.998$ comprueba la linealidad entre el RTA promedio en función de la distancia con una pared de distancia con una pared de concreto como obstáculo entre el par transmisor – receptor.

La existencia de una ecuación lineal para modelar el RTA sin y con obstáculos muestra que el desempeño de la tecnología BT dentro de su rango de operación es confiable y que las pérdidas por propagación debido a obstáculos también son estables dentro de este rango de operación, lo cual es útil para aplicaciones multimedia.

III.2 Comparación con versiones anteriores y Coexistencia con WiFi.

La versión 2.0 +EDR (Enhanced Data Rate) de la tecnología Bluetooth ratificada en noviembre de 2004 incrementa la velocidad de transmisión sin impactar a las capas superiores. Con un caudal eficaz real de 2.1 Mbps (sin considerar el sobre encabezado del paquete), mientras que la versión anterior solo lograba un caudal eficaz real de 723 kbps, también sin considerar el encabezado. Las únicas diferencias entre estas versiones de BT se

dan en la velocidad de transmisión y en la técnica de modulación empleada, tal como se muestran en la tabla IV.

	Bluetooth v1.2	Bluetooth v2.0+ EDR
Rate (Mbps)	1 Mbps	2 - 3 Mbps
Modulation	GFSK QPSK	$\pi/4$ DQPSK 8DPSK

Tabla IV. Comparación entre versiones 2.0 +EDR y 1.2 de Bluetooth.

Para mantener compatibilidad entre las versiones 1.2 y 2.0 +EDR, los paquetes BT v2.0 +EDR utilizan dos campos adicionales, tal como se muestran en la figura 6: un campo de banda de guarda y un campo de palabra de sincronía.

Debido a que BT 2.0 +EDR puede utilizar una de dos tipos de modulaciones para transmitir la carga útil, los campos de banda de guarda y palabra de sincronía se utilizan para indicar si los datos se transmitirán utilizando la modulación $\pi/4$ -DQPSK o DPSK. Los campos de código de acceso y encabezado se transmiten utilizando la modulación GFSK, al igual que los paquetes de BT v1.2, donde todo el paquete se transmite utilizando únicamente la modulación GFSK, tal como se ejemplifica en la figura 6.

Respecto al desempeño, en la referencia [5] se dan resultados de la versión 1.2 para el mismo experimento: RTA versus distancia. Para propósitos de comparación se utilizaron los tamaños de archivo de 1 y 5 Mbytes transmitidos a una distancia de 6 metros sin obstáculos. Estos resultados se muestran en la tabla V. Los retardos para la transmisión (RTA) de archivo están medidos en segundos.

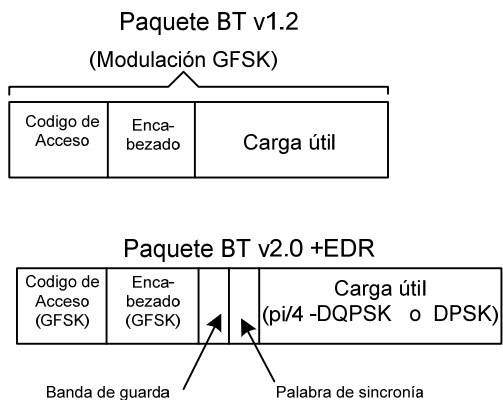


Fig. 6 Estructura del paquete +EDR.

Distancia 6 metros, sin obstáculos		
Tamaño de archivo	BT 1.2 *	BT 2.0 +EDR
1 MByte	25.7	8.21
5 Mbytes	124.6	37.55

Tabla V. Comparación entre versiones de BT para la transmisión de archivos de tamaño variable en función de la distancia. *Los datos de esta columna son los reportados en la referencia [5].

De la tabla V se puede concluir que efectivamente se tiene un incremento en la velocidad en la transmisión de archivos de 3X para esta nueva versión de la tecnología BT. Es importante remarcar esta conclusión, debido a que esto no necesariamente sucede con otras tecnologías, como es el caso de Ethernet o WiFi. Donde un incremento en la velocidad de transmisión de 10X no se traduce en un incremento similar en la velocidad de transferencia de archivos, esto debido principalmente al método de acceso empleado y a la agregación de campos en el encabezado del paquete de datos, necesarios para asegurar una compatibilidad entre versiones, principalmente con anteriores.

III.3 Coexistencia con WiFi.

Esta nueva versión coexiste mejor con otras tecnologías con las que comparte la banda ISM de 2.4 GHz.

Bluetooth y el 802.11 operan en el mismo espectro de frecuencias, de la banda ISM (que no necesita licencia). Por lo tanto se puede esperar que el desempeño de dispositivos usando estas tecnologías, se vea afectado adversamente, cuando operan simultáneamente en un mismo ambiente. Así, el grupo de trabajo de coexistencia 2 (TG2) del IEEE 802.15, ha propuesto un mecanismo adaptativo de salto de frecuencia (AFH). AFH toma en consideración la condición del canal, y cambia los saltos de frecuencia dinámicamente, permitiendo la coexistencia con otros dispositivos trabajando en la banda ISM. AFH consiste en 2 etapas: Clasificación del canal, y el Protocolo de control adaptativo. La clasificación del canal, consiste en elaborar una lista dinámica de canales buenos y malos, basándose en la calidad del canal. Esta información se intercambia, haciendo uso del Protocolo de control adaptativo. Con esta información el protocolo AFH, escoge un conjunto de saltos de frecuencias, de manera que se eviten, tanto como sea posible, los canales malos. Si el número de canales buenos, es menor de 15, (el menor número de canales que necesita Bluetooth para operar, de acuerdo con el estándar), entonces algunos canales malos tendrían que ser utilizados.

En el experimento descrito en la sección III.1 el sniffer B estuvo monitoreando la red WiFi (IEEE 802.11b) académica instalada y que funciono durante todo el experimento para observar el sí el desempeño de BT se degradaba con la presencia de esta. Durante las mediciones no se observo degradación del desempeño de BT y para comprobar los resultados, el experimento se realizo en dos ocasiones: con y sin la presencia de WiFi. No se observaron diferencias significativas, por lo que los datos reportados en este trabajo son los que arrojó el experimento coexistiendo con la presencia de WiFi. Estos resultados contrastan con los reportados en [2] y [8] para la versión 1.2 de BT.

En [9] se reporta la evaluación del desempeño de Bluetooth con usuarios móviles, cuando se presenta interferencia de tráfico intenso WiFi. Tanto para la versión 1 como para la versión 2 de Bluetooth, la interferencia de WiFi, en la descarga de archivos cuando el usuario es móvil, se vio

afectada, bajando la tasa de bit efectiva, en 30 %, en comparación con la descarga sin presencia de tráfico WiFi. Los autores de [9] concluyen que el mecanismo adaptativo AFH no funciona adecuadamente en el caso de usuarios móviles.

Sin embargo para el caso de usuarios relativamente estáticos, que se presenta en este trabajo, la descarga de archivos en presencia de obstáculos, no se afecta por la presencia de tráfico ligero WiFi.

IV. Conclusiones y trabajos futuros.

Como se menciona en la introducción de este trabajo, la importancia del estudio y análisis de la tecnología de BT radica en que se encuentra instalado en miles de millones de dispositivos portátiles. De aquí la importancia de desarrollar esta tecnología, que originalmente fue concebida para la eliminación de cables para dispositivos como teclados, mouse o impresoras. Ahora, debido a su ubicuidad, puede llegar a convertirse en la tecnología que reúna o conjunte a todos los dispositivos informáticos de uso personal y doméstico: cámaras de video, cámaras digitales, radios, tv, equipos de sonido, laptops, pda, etc. y posiblemente reemplazar a la tecnología de infrarrojo, principalmente en lo que se refiere a los mandos a distancia, utilizados en aparatos de uso doméstico: como televisiones y equipos de sonido, principalmente.

Para que BT llegue a convertirse en la tecnología de transporte de la información de los dispositivos mencionados, debe de ser capaz de poder transmitir video en tiempo real tal como lo citan en las referencias [4], [6], [7] y [8]. También debe ser capaz de poder mantener transmisiones de audio en tiempo real entre más de 3 nodos como citan en [3]. Para lograr estos dos retos, se deben desarrollar nuevos protocolos de transporte (tipo TCP, UDP o RTP) o modificarlos y nuevos esquemas de calendarización que tomen ventaja de la multiplexión en tiempo del canal de BT. Otro reto importante es lograr aumentar la velocidad de transmisión, para lo cual se esta pensando en la tecnología de Ultra Wide Band (UWB) [1]. Se prevé que las nuevas versiones de BT incluirán

este tipo de modulación y que logran velocidades de 20 a 50 mega bytes por segundo, con lo cual se podría lograr transmitir video en tiempo real y tener transmisiones simultaneas de audio de calidad entre los nodos que forman una piconet.

Respecto al desempeño de esta tecnología, se concluye que el retardo en la transferencia de archivos de tamaño variable es proporcional al tamaño del archivo transmitido y sigue un modelo lineal sin y con la presencia de obstáculos entre el origen y el destino. Otra conclusión importante es que la presencia de obstáculos entre el origen y el destino introduce un retardo sin importar la distancia en que estos se encuentren. Estos resultados son importantes en los que desee agregar calidad de servicio a transmisiones multimedia, debido a que se puede analizar analíticamente los retardos que tendrán y programar o desarrollar aplicaciones que soporten estos retardos.

Referencias

- [1] The Bluetooth Special Interest Group, <http://www.bluetooth.com>
- [2] F. P. Erina Ferro. Bluetooth and Wi-Fi Wireless Protocols: A survey and a Comparison. IEEE Wireless Communications Magazine. February 2005.
- [3] David McCall. Taking a walk inside Bluetooth EDR. <http://www.wirelessnetdesingline.com>
- [4] R. Razavi, M. Fleury, and M. Ghanbari. Video-streaming Applications Enabled Across Bluetooth v. 2.0 Interconnects. IADIS Wireless Applications and Computing, Lisbon, Portugal, 2007.
- [5] R. Rashid, R.A.; Yusoff. Bluetooth Performance Analysis in Personal Area Network (PAN). IEEE International RF and Microwave Conference Proceeding. September 12-14, 2006. Putrajaya, Malaysia.
- [6] Vilovic, I.; Zovko-Cihlar. Performance of the Bluetooth - based WPAN for Multimedia Communication. Video/Image Processing and Multimedia Communications, 2003. 4th

EURASIP Conference focused on Volume 2,
Issue , 2-5 July 2003 Page(s): 783 - 788 vol.2

- [7] M. F. Rouzbeh Razavi and M. Ghanbari. Unequal Protection of Video Streaming Through Adaptive Modulation with a Trizone Buffer over Bluetooth Enhanced Data Rate. EURASIP Journal on Wireless Communications and Networking, October 2007.
- [8] Shuaib, K. Boulmalf, M. S. F. L. A, Performance Analysis Co-Existence of IEEE 802.11g with Bluetooth. Wireless and Optical Communications Networks, 2005. WOCN 2005. Second IFIP International Conference on Volume , Issue , 6-8 March 2005.
- [9] Uichin Lee, Sewook Jung, Alexander Chang, Dae-Ki Cho, Mario Gerla. "Bluetooth-based P2P Content Distribution to Mobile Users ". UCLA Technical Report: TR-070013 (submitted to TMC).

SISTEMA MEJORADO DE TELEMETRIA DE BAJO COSTO PARA APLICACIONES AGRICOLAS

Daniel Ortiz Barredo*, Miguel E. Martínez-Rosas*†, Humberto Cervantes de Ávila*,
Eduardo A. Murillo Bracamontes*, Carlos Gómez Agis*.

* Facultad de Ingeniería-Ensenada
Universidad Autónoma de Baja California
Km. 103 carr. Tijuana-Ensenada, 22800, Ensenada, Baja California, México
Correo electrónico: emartine@uabc.mx, dbarredo@gmail.com

† Centro de Investigación Científica y de Educación Superior de Ensenada
Km. 107 carr. Tijuana-Ensenada, 22800, Ensenada, Baja California, México

RESUMEN

En este trabajo se presenta el diseño e implementación de una red de sensores inalámbrica económica, de baja velocidad y bajo consumo de potencia con el fin de monitorear variables medioambientales tales como temperatura, humedad relativa y luminosidad para aplicaciones agrícolas. En base a una red de topología estrella presentada en un trabajo anterior, se logró expandir el alcance de ésta creando una red jerárquica de dos niveles o red de redes estrellas. Se utilizó una topología tipo árbol, de tal manera que se crearon vecindades formadas por múltiples nodos y una base coordinadora, donde cada base responde a un coordinador general de la red. La comunicación utiliza el protocolo propietario SMAC de Freescale, ya que las funciones que este provee son suficientes para el desarrollo de la aplicación. Se optimizó el funcionamiento de nodos, bases y coordinador, logrando de esta manera el diseño de una red inalámbrica que cumple con las expectativas esperadas.

Palabras clave: Redes de sensores inalámbricas; Protocolo; SMAC; Nodos; Bases; Coordinador.

I. INTRODUCCIÓN

Una red de sensores inalámbrica consiste en un arreglo de miles (hasta millones) de sensores de diversos tipos interconectados a través de una red de comunicación inalámbrica. Los datos son utilizados como entradas en un sistema de

estimación distribuido cuya función es extraer información relevante de los datos disponibles [1]. El rápido desarrollo, la autoorganización y la tolerancia a fallas son características que hacen a las redes de sensores prometedoras para un gran número de aplicaciones civiles y militares.

El desarrollo de protocolos y arquitecturas para el diseño de redes de sensores inalámbricas se ha convertido en un campo de investigación muy importante en los últimos años [2]. El conocimiento de los cambios de variables críticas en un sistema permite tomar decisiones de control, lo que afectará directamente en la respuesta de éste. Por ejemplo, el conocimiento de la cantidad de agua y pesticidas utilizados en un cultivo pueden representar la diferencia entre una buena cosecha o una cosecha perdida, con sus consecuentes repercusiones económicas.

Desde hace algunos años han surgido nuevas tendencias en el sector agrícola. Gracias al desarrollo en el campo de redes de sensores inalámbricas, tales como la miniaturización de las tarjetas impresas, la agricultura de precisión ha comenzado a emerger [3]. Una faceta de la agricultura de precisión se concentra en el control de cosechas específicas. Esto abarca diferentes aspectos tales como la monitorización del suelo y el clima en el campo. Es posible generalizar el estado del campo de cultivo completo conociendo el valor en puntos específicos de algunas de las variables que influyen de manera crítica en la calidad de la cosecha. En la actualidad existen algunos ejemplos de redes de sensores

inalámbricas importantes, tal es el caso de Ambient Systems [4] y el de la compañía Grape Networks [5], la cual tiene instalada la red de sensores inalámbrica más grande del mundo, cubriendo aproximadamente 50 acres con más de 200 sensores en el valle de San Ramón, California, EU [6].

Una de las motivaciones clave para el uso de la tecnología inalámbrica es la reducción de costos por instalación. Los sistemas inalámbricos se han impulsado gracias a las nuevas tecnologías, con un alto grado de integración y con bajo costo de los componentes requeridos [7]. A la fecha existen muchos protocolos de comunicación diseñados para aplicaciones en redes de sensores inalámbricas, siendo los más utilizados IEEE 802.11, Bluetooth e IEEE 802.15.4. Este último presenta características ventajosas para su aplicación a redes de sensores, ya que, a diferencia de los primeros, está orientado hacia el desarrollo de redes inalámbricas de baja velocidad (es decir, aplicaciones en donde la transferencia de información no es muy alta), bajo costo y muy bajo consumo de potencia. El protocolo más robusto para el desarrollo de redes de sensores inalámbricas es Zigbee, el cual tiene como base la capa física y la capa MAC del estándar 802.15.4.

Los requisitos para cualquier aplicación de redes de sensores inalámbricas siempre dependerán estrechamente de la aplicación concreta, por lo que la elección del protocolo no debe de ser tomada a la ligera. Al realizar el diseño hay que tomar en cuenta tres parámetros de importancia. El primero de ellos es el bajo consumo de potencia. Al ser alimentados por baterías, tanto los nodos como el protocolo de comunicación de comunicación requieren ser diseñados con características específicas, buscando el máximo ahorro de energía posible. El segundo parámetro es la latencia, la cual representa la suma de retardos producidos por la demora en la recepción y transmisión de datos dentro de una red. La latencia presente en la comunicación de variables medioambientales no es un factor crítico, ya que la monitorización de éstas no presenta cambios abruptos ni limitaciones críticas. El tercer parámetro es la confiabilidad. El usuario final espera que los datos obtenidos a través de la red sean correctos, de tal forma

que las decisiones que se tomen a partir de ellos sean afines a la mejora del producto. Este trabajo presenta el diseño de una red de sensores inalámbrica a implementarse específicamente en un viñedo del Norte

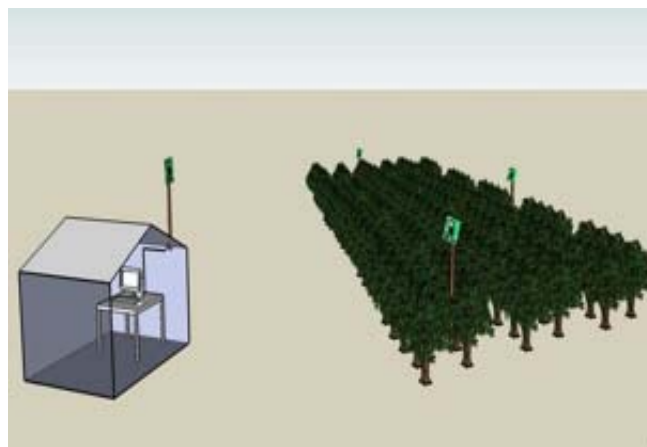


Fig. 1. Instalación de red estrella en un campo agrícola.

de México con la cual el usuario interactuará a través de una interfaz gráfica. Es importante notar que los usuarios de este sistema serán agricultores y trabajadores del campo, y que el principal interés de estos se encuentra enfocado a la producción. Ellos requieren datos que recomienden que decisión tomar y ahorre tiempo, no un sistema que genere trabajo adicional [8].

Con la implementación de este sistema en un área agrícola se busca otra meta de prioridad: la optimización de agua para el riego, ya que al ser éste un recurso limitado y compartido por todos, es imperante el racionarlo de manera óptima. La producción agrícola utiliza aproximadamente el 70 % de el agua dulce a nivel mundial, y en las poblaciones en crecimiento la demanda es mayor, por lo que el nivel de agua disponible disminuye mucho más rápido de lo que es recuperado [9]. Finalmente una de las grandes ventajas que presentan las redes de sensores inalámbricas contra las redes cableadas es su adaptabilidad, ya que con esta tecnología es posible monitorear áreas remotas, peligrosas y de difícil acceso [10]. Como ya se mencionó, otro punto muy importante es la minimización de costos. En [10] se menciona que al utilizar este tipo de redes es posible lograr una reducción del 20-80 % respecto al costo de una red alamburada. Estas dos características, sumadas al gran avance tecnológico

de esta área en los últimos años hacen posible proponer esta tecnología como una importante herramienta dentro del medio agrícola, industrial e incluso residencial.

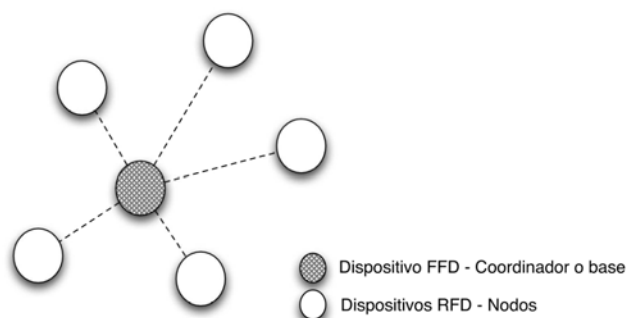


Fig. 2. Diagrama de topología estrella en redes de sensores inalámbricos.

Sin embargo, no todo está hecho. Aunque este es un tema en constante desarrollo, las redes de sensores inalámbricos son aún muy primitivas y propensas a fallas. En [11] se muestran los resultados obtenidos por el proyecto LOFAR-agro [12] al desplegar una red con más de 100 nodos para una prueba piloto de tres meses en un campo agrícola. La prueba no respondió como se esperaba, pero reveló muchos problemas de ingeniería que habían sido omitidos.

II. PRINCIPIOS DE OPERACIÓN

Una red de sensores inalámbrica está formada por múltiples nodos, los cuales a su vez cuentan con una unidad de control o microcontrolador, un transreceptor y sensores. Los nodos son posicionados en lugares estratégicos donde se requiera conocer el cambio de alguna variable específica. Dependiendo de su topología, una red de sensores inalámbrica puede ser conocida como red estrella o red malla. La topología estrella, mostrada en la Fig. 2 es la más común y sencilla de implementar, ya que el protocolo de comunicación necesario para su funcionamiento es mínimo.

La Fig. 1 muestra el escenario particular en el cual se implementará la red: se requiere diseñar una red inalámbrica fija, la cual monitorizará humedad relativa, temperatura y luminosidad en distintos puntos dentro

de un viñedo. Debido a que las variables a sensor no presentan un cambio abrupto, el tráfico de la red es bajo al igual que la cantidad de información a transmitir. Se requiere una unidad central en donde los datos sensados se concentren, se almacenen y sean desplegados de manera gráfica al usuario. El presente trabajo está basado en [13], en donde se muestran los resultados obtenidos para una red estrella básica. Para el desarrollo de esta nueva red se utilizó la topología árbol, la cual está basada en las redes estrella. Al modificar la topología se buscó cubrir una mayor extensión de terreno, lo cual, si bien aumenta el poder de la red, también aumenta la complejidad de ésta. Una de las consideraciones básicas a tomar en cuenta en esta nueva red es la sincronización de los elementos. A mayor cantidad de superficie cubierta, mayor cantidad de elementos de la red trabajando. Se debe de realizar un proceso de sincronización cada determinado tiempo con el fin de saber cuantos y cuales nodos se encuentran trabajando y en que vecindad se encuentran, para evitar que dos o más nodos transmitan su información a la misma base al mismo tiempo. Es decir, se busca reducir la pérdida de información en el aire por choques de tramas.

III. EXPERIMENTOS Y RESULTADOS

Para la implementación de la red se utilizaron los módulos PAN802154 de Panasonic como el mostrado en la Fig. 3. Estos están diseñados con los componentes necesarios para funcionar como dispositivos en una LR/WPAN (Low Rate Wireless Personal Area Network), tanto como coordinador, enrutador o nodo. Estos módulos están equipados con el microcontrolador MC9S08GT60CFBE de Freescale y el transreceptor MC13192. Las principales características con las que cuentan estos módulos son:

- Soporte completo a aplicaciones Simple MAC, 802.15.4 y Zigbee.
- Banda de trabajo ISM a 2.4 GHz.
- 16 canales con 5 Mhz de espaciamiento entre ellos.
- Transmisión a 250 Kbps.
- Puerto RS-232, convertidos A/D de 10 bits, dos entradas analógicas y 8 puertos digitales I/O.
- Potencia de salida típica de 0 dBm (1mW).
- Sensitividad del receptor: típicamente -92 dBm al 1 % de tasa de paquetes erroneos.
- Antena impresa en la tarjeta. Posibilidad de conectar una antena externa.
- Rango de energía de 2.2 3.4 VDC sin RS-232 y de 3 a 3.4 VDC con RS-232.
- Corriente DC típica de 35 mA.
- 5 µA en bajo consumo con interrupción en tiempo real.



Fig. 3. Módulo PAN802154 de Panasonic.

El protocolo seleccionado para la realización de este trabajo fue SimpleMAC o SMAC [14] de Freescale. SMAC es un protocolo sencillo basado en el estándar IEEE 802.15.4, implementado para

trabajar con los transreceptores manejados por microcontroladores de la misma compañía. SMAC no implementa de manera completa ni la pila de Zigbee ni la capa MAC completa de 802.15.4, sin embargo, es ideal para aplicaciones sencillas y de bajo costo. El código está basado en ANSI C y es distribuido en el software Beekit, el cual requiere

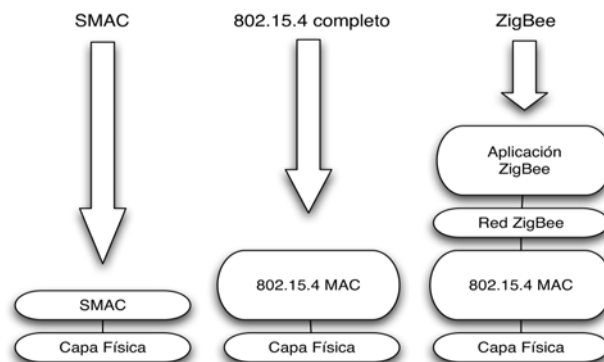


Fig. 4. Esquema a bloques comparativo de los protocolos SMAC, 802.15.4 y Zigbee.

de 4 a 8 Kb de memoria Flash. La Fig. 4 muestra comparativamente de manera gráfica y sintetizada las diferencias entre SMAC, 802.15.4 y Zigbee.

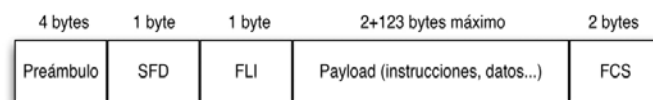


Fig. 5. Configuración de la trama SMAC al transreceptor MC13192.

La Fig. 5 muestra la estructura del paquete del transreceptor MC13192. Este está dividido en cinco partes principales. Los cuatro primeros bytes son conocidos como preámbulo. Cuando este es detectado por el receptor, se inicia una medición de energía recibida durante un periodo de 64 µS, la cual es llamada LQI (Link Quality Indicator). El byte SFD (Start of Frame Delimeter) indica el fin de la cabecera de sincronización y el inicio de la

trama de datos. En el byte FLI (Frame Length Indicator) se almacena la longitud de la trama. El siguiente bloque es conocido como Payload Data, con una longitud total de 125 bytes, donde los dos primeros son otorgados por la subcapa MAC y configuran a la trama en modo no promiscuo. En los 123 bytes restantes se colocan los datos, comandos o cualquier información que el programador considere necesaria. Finalmente, los últimos dos bytes son de FCS (Frame Check Sequence). En estos se introduce un código de 16 bits generados por un algoritmo CRC (Cyclical Redundancy Check) definido en [15]. Al recibir un paquete, el receptor calcula el FCS del paquete recibido y compara el resultado con los últimos dos bytes de este. Si los resultados son iguales significa que el paquete se transmitió sin errores.

Antes de iniciar el desarrollo del protocolo de la red se realizaron pruebas de directividad y potencia a los módulos con el fin de conocer el patrón de radiación de la antena impresa y estimar un rango de distancia seguro para la transmisión y recepción exitosa de tramas. Estas pruebas consistieron en transmitir un número conocido de paquetes de un módulo a otro empleando como base una plantilla provista por *Beekit* con el protocolo mínimo de enlace (el cual fue ligeramente modificado). En las pruebas de directividad se midió la potencia captada por el receptor variando gradualmente la posición de éste respecto al transmisor con el objetivo de determinar la orientación física óptima de ambos. Por medio de esta prueba se observó que la máxima potencia de radiación de la antena se tiene entre 180 y 280 grados, cuando el módulo es orientado en el plano horizontal, para la orientación en el plano vertical la máxima potencia se obtiene entre -40 y 40 grados. En la Fig. 6 se muestra el patrón de radiación normalizado a la máxima potencia obtenido con mediciones cada 20 grados.

Para las pruebas de potencia se colocó el módulo receptor en un punto fijo y el módulo transmisor a 20 m de éste manteniendo línea de vista. El receptor, al captar los paquetes enviados, estimaba el LQI con el que se recibía cada paquete. Esta prueba se repitió incrementando la distancia del transmisor al receptor en intervalos de 20 m hasta llegar a las 100 m. La Fig. 7 muestra los resultados obtenidos,

donde se observa una caída abrupta de potencia aproximadamente a los 80 m de distancia. Esta peculiaridad se presentó a lo largo de todas las pruebas realizadas.

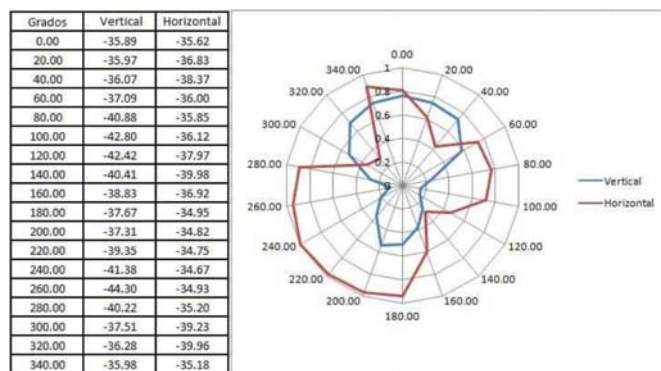


Fig. 6. Directividad de la antena impresa en el módulo PAN802154.

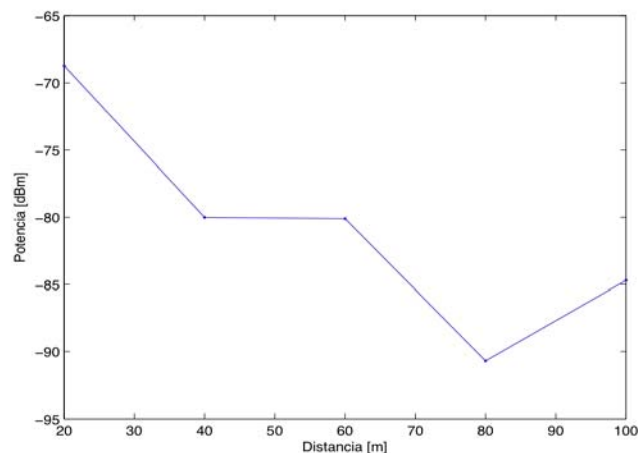


Fig. 7. Relación Potencia / Distancia de los módulos PAN802154 cada 20 m. a.

Para el diseño inicial de la red se manejó una topología estrella como la mostrada en la Fig. 2. En base a los resultados obtenidos en [13] fue posible ampliar ésta para formar una red jerárquica de dos niveles tal como se observa en la Fig. 8. Esta red está constituida por múltiples vecindades que responden a una estación coordinador. Cada vecindad está formada por múltiples nodos (dispositivos RFD de bajo consumo) y una base (dispositivo FFD), donde cada uno de estos

elementos tiene asignadas tareas específicas. Los nodos están dispersos en el área a sensar, enviando cada determinado tiempo información a la estación base y recibiendo comandos de control de ésta. Las bases recolectan los datos enviados por los nodos, sincronizan

Canal	Frecuencia	Canal	Frecuencia
0	2.405 GHz	8	2.445 GHz
1	2.410 GHz	9	2.450 GHz
2	2.415 GHz	10	2.455 GHz
3	2.420 GHz	11	2.460 GHz
4	2.425 GHz	12	2.465 GHz
5	2.430 GHz	13	2.470 GHz
6	2.435 GHz	14	2.475 GHz
7	2.440 GHz	15	2.480 GHz

Tabla 1. Relación de canales disponibles.

cada determinado tiempo los nodos de su vecindad y envían la información promediada de cada uno de sus nodos a la estación coordinador. Por su parte esta última toma la función de PANC (personal area network coordinator), la cual está conectada a una computadora utilizando la comunicación RS-232, de manera que al recibir información de las bases el usuario pueda obtener e interpretar los datos con la ayuda de una interfaz gráfica, los datos a su vez son almacenados en el disco duro de la computadora.

Cada vecindad está formada por una sola base, y los nodos sólo pueden agruparse a una vecindad a la vez. Con el fin de aprovechar el ancho de banda existente y el de no sobrecargar de trabajo a las bases, sólo existirá una vecindad en cada canal, de ésta manera cada base atenderá solo a los nodos que formen parte de su vecindad. Cada vez que una base es inicializada, ésta realiza un barrido de frecuencia emitiendo un mensaje de *broadcast* en el canal 0, buscando de ésta manera otra base en dicho canal. Si no obtiene respuesta, toma este como su canal de trabajo. En caso de recibir una respuesta de otra base, cambia al canal 1 y vuelve a emitir el mensaje de *broadcast*. Este proceso será repetido hasta encontrar un canal libre. De esta manera se logra manejar un máximo de 15 bases cada una en un canal diferente. La Tabla I muestra las frecuencias de

operación de cada uno de los 16 canales disponibles, en donde el ancho de banda de cada uno es de 2 MHz espaciados cada 5 MHz. Debido a que las bases deben permanecer en estado de recepción la mayor parte del tiempo, es recomendable que su fuente de energía sea constante.

Para que un nodo sea asociado a una red, éste emite un mensaje de petición de nombre en cada uno de los 16 canales, almacenando tanto la respuesta de cada base como el LQI. Una vez terminado el barrido de canales, el nodo se asociará a la vecindad de la base con la cual presentó un mejor LQI. Este proceso presenta dos grandes ventajas: la primera de ellas es la autoconfiguración. No existen vecindades preestablecidas; éstas se formarán al momento de inicializar la red de manera transparente al usuario, de tal forma que de requerirse trasladar un nodo de una vecindad a otra tan solo basta mover éste a su nueva posición e inicializarlo. La segunda ventaja es que al hacer una petición de adopción a todas la bases existentes, el nodo se asegura de conectarse a aquella con la cual tenga menos probabilidades de perder su enlace. Una vez asociado a una vecindad, el nodo está en estado de reposo salvo cuando transmite a su base la información provista por sus sensores. Ya que las variables a medir no presentan cambios críticos en periodos de tiempo muy cortos, los nodos realizan esta operación cada diez minutos, sin embargo este tiempo puede ser modificado a conveniencia del usuario.

Al estar los nodos asociados a una vecindad, estos sólo transmitiran información a su base; de la misma manera, una base (a excepción de la solicitud de nombre de un nuevo nodo) solo atenderá a los nodos de su vecindad. En caso que una base deje de funcionar, los nodos de su vecindad dejarán de recibir su trama de ACK, por lo que entrarán en estado de retransmisión del último dato un número predeterminado de veces. Después de estas retransmisiones, el nodo asume que su base está deshabilitada y procede de nuevo a solicitar la adopción a otra vecindad transmitiendo de nuevo una petición de nombre en cada canal. De esta forma, los nodos pasan a formar parte de otra vecindad y no quedan inutilizados. A diferencia de las bases y del coordinador, los nodos sólo requieren estar prendidos la menor parte del tiempo, por lo tanto es factible proveerles energía con baterías. Una importante tarea que deben cumplir las

bases es sincronizar todos los nodos de su vecindad cada cierto tiempo. Al recibir la información de sus nodos, cada estación

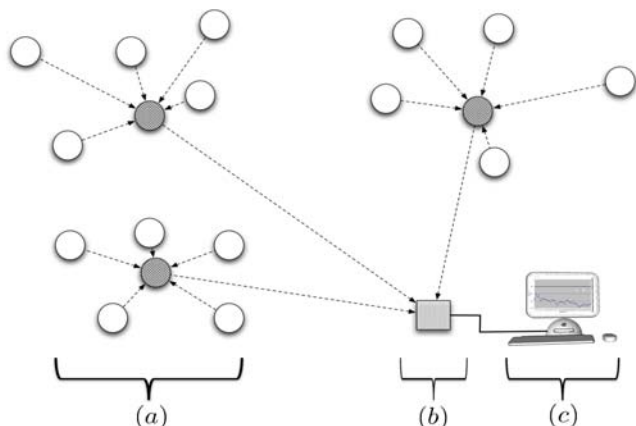


Fig. 8. Diagrama de topología de una red con tres vecindades. (a) Nivel 2. La base de cada vecindad envía sus datos al coordinador. (b) Nivel 1. El coordinador envía toda la información de la red a una PC via comunicación RS-232. (c) La PC despliega la información de la red.

base almacena en memoria tanto el nombre del nodo transmisor como el dato y el número de dato enviado. Cuando se llega a la décima muestra del primer nodo asociado, la base detiene a todos sus nodos, promedia los datos de cada uno de ellos, procede a enviar dicho promedio, junto con su trama de identificación a la estación coordinador y finalmente otorga a cada uno de sus nodos una ranura de tiempo diferente, de tal manera que 2 de los nodos sincronizados no transmitan su información al mismo tiempo, lo que garantiza una menor pérdida de paquetes y elimina la retransmisión de estos por colisiones en el medio. Este proceso se repite cada diez muestras recibidas por la base para de esta manera sincronizar nodos nuevos.

Como ya se mencionó, el coordinador debe estar comunicado con una PC. El coordinador tiene conocimiento en todo momento de cuantas bases existen en la red y en que canal se encuentra cada una de ellas. Cada vez que el coordinador recibe un paquete de datos de una base, envía este paquete a la PC, en donde podrán visualizarse los datos medidos y la dirección de los nodos transmisores utilizando para este fin una interfaz gráfica programada en lenguaje *LabView* de *National Instruments*TM. La Fig. 9 muestra de manera básica el

funcionamiento del coordinador. En [13] se muestran los diagramas tanto de los nodos como de las estaciones base.

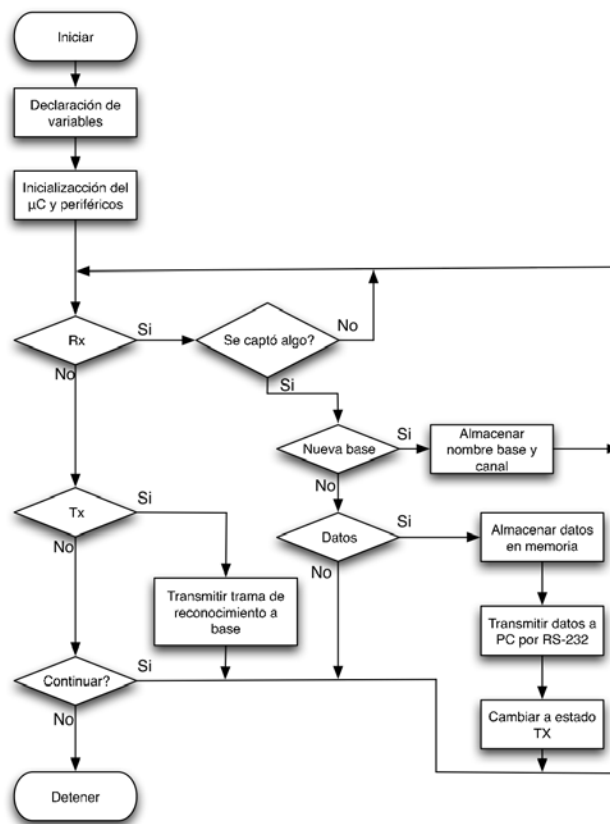


Fig. 9. Diagrama de flujo de el coordinador.

IV. CONCLUSIONES

Se diseñó de una red tipo árbol jerárquica de dos niveles ampliando la extensión de superficie cubierta en [13], con un radio de operación aproximado de 60 m por vecindad a una tasa de paquetes erróneos del 1 % y la cual utiliza un método de acceso al medio sencillo basado en el estándar 802.15.4. Se demostró la omnidireccionalidad de las antenas y que los resultados de potencia obtenidos en la red en relación a la distancia, aunque con una diferencia en los límites debida a la diferencia de

escenarios, fueron congruentes con los resultados de la caracterización Tx-Rx. Se planea la implementación de esta red en un importante campo vinícola de la región Noroeste de México, con lo que se prevé la disminución del consumo de agua en relación al riego sin monitorización.

V. REFERENCIAS

- [1] Mostafa I. Abd-El-Barr, Mohamen A.M. Youssef, Mariam M. Al-Otaibi. “*Wireless sensor networks-part 1: Topology and design issues*”, Canadian Conference on Electrical and Computer Engineering, 2005.
- [2] Abraham Menendez Marquez, Juan Jose Perez Solano, Jose Pelegr Sebastia. “*Red de sensores inalámbricos para monitorización de terrenos mediante tecnología ieee 802.15.4*”, Universidad de Valencia.
- [3] Aline Baggio. “*Wireless sensor networks in precision agriculture*”, Workshop on Real-World Wireless Sensor Networks, 2005.
- [4] Ambient Systems, en <http://www.ambient-systems.net>
- [5] Grape Networks, en www.grapenetworks.com
- [6] *Applications: World’s largest internet enabled, wireless mesh sensor network for agriculture*, en <http://www.measurementdevices.com/index.php?name=News&file=article&sid=918>
- [7] *IEEE 802.15.4 y Zigbee*, Jordi Mayne, http://www.bairesrobotics.com.ar/data/ieee_zigbee_silica.pdf
- [8] J. Burrell, T. Brooke, R. Beckwith, “*Vineyard computing: sensornetworks in agricultural production*”, Pervasive Computing, IEEE., 3(1):38–45, Enero-Marzo 2004.
- [9] Chibuzor Edordu, Lionel Sacks, “*Self organising wireless sensornetworks as a land management tool in developing countries: A preliminary survey*”, London Communications Symposium, 2006.
- [10] Ning Wang, Naiqian Zhang, Maojua Wang, “*Wireless sensors in agriculture and food industry—recent development and future perspective*”, Computers and Electronics in Agriculture, 2006.
- [11] K.G. Langendoen, A. Baggio, O.W. Visser, “*Murphy loves potatoes: Experiences from a pilot sensor network deployment in precision agriculture*”, 14th Int. Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), 2006.
- [12] LOFAR, en <http://www.lofar.org/p/Agriculture.htm>
- [13] Daniel Ortiz Barredo, Miguel E. Martínez Rosas, Humberto Cervantes de Ávila, Eduardo A. Murillo Bracamontes, Carlos Gómez Agis, “*Sistema de telemetría de bajo costo para aplicaciones agrícolas*”, Aceptado para su publicación en congreso *IEEE COLCOM*, 2008.
- [14] *Simple media access controller (smac) users guide*, Freescale Semiconductor en http://www.freescale.com/files/rf_if/doc/user_guide/SMACRM.pdf
- [15] *Part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for lowrate wireless personal area networks (wpans)*, IEEE Computer Society en <http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf>

DISEÑO FÍSICO DE CIRCUITOS INTEGRADOS EN PROCESOS NANOMÉTRICOS

Sáenz Irigoyen Emilio^{1,2}, Varona Salazar Jorge².

¹Escuela de Ingeniería, Universidad Panamericana campus Bonaterra.
Josemaría Escrivá No. 101, Aguascalientes, Ags., 20290, México
Tel. (449)9106200, correo electrónico: esaenz@ags.up.mx

²Snowbush Microelectronics Mexico
Venustiano Carranza No. 122, Aguascalientes, Ags., 20000, México
Tel. (449)9182823, correo electrónico: varona@ieeee.org

RESUMEN

Este artículo presenta un análisis sobre los retos de diseño de circuitos integrados (CIs) en tecnologías nanométricas así como algunas de las técnicas de diseño físico que deben emplearse para enfrentar los serios problemas tecnológicos que impone la fabricación de dispositivos en la escala de nanómetros. Conforme se reducen las dimensiones del transistor con cada nueva generación, se presenta una mayor desviación entre la operación estimada por los modelos matemáticos básicos del transistor MOS y el comportamiento real de los dispositivos así como una mayor dependencia de los efectos geométricos derivados del diseño físico del circuito. El diseño físico o “*layout*” de circuitos ha cobrado cada vez mayor relevancia en el proceso de diseño conforme los transistores reducen sus dimensiones más allá de los 100nm debido, entre otros, al uso de dimensiones críticas que alteran la física que gobierna el comportamiento de los dispositivos a esa escala. Además, en los modernos procesos litográficos de fabricación existe menor control sobre la variabilidad de las características de los dispositivos y menor confiabilidad. En vista de lo anterior, este artículo presenta algunas técnicas avanzadas de diseño físico para la implementación de circuitos en dimensiones moleculares y su caracterización experimental usando un proceso de maquinado superficial sobre un substrato de silicio.

Palabras clave: Diseño físico, *layout*, circuitos integrados, nanotecnología, litografía.

I. INTRODUCCIÓN

El éxito de la industria electrónica y de semiconductores depende del continuo mejoramiento en el desempeño de los circuitos integrados. Históricamente, esto se ha logrado primordialmente mediante la constante miniaturización de los dispositivos de estado sólido para permitir que una mayor cantidad de transistores sea colocada dentro de un área determinada de una oblea de silicio. El resultado inevitable de esta continua miniaturización al través de los años ha sido la transformación de la clásica microelectrónica en la nanoelectrónica de hoy día, con una nueva serie de reglas y métodos de diseño más apropiados de acuerdo con las características propias de la tecnología. En general, la nanotecnología se define como la investigación y desarrollo tecnológico al nivel atómico, molecular, o macromolecular en la escala de 1 a 100nm [1]. Durante décadas, la miniaturización de los dispositivos semiconductores fue sinónimo de un mejor desempeño de los transistores, menores tiempos de conmutación, mayor ancho de banda, menor consumo de potencia, etc. Sin embargo, el advenimiento de la nanoelectrónica ha significado el límite en cuanto al incremento en el desempeño ya que a esta escala, los parasíticos asociados a la estructura física del transistor y sus interconexiones son comparables con la resistencia y capacitancia intrínsecas de los dispositivos [2]. En otras palabras, en la escala nanométrica los parasíticos dominan el desempeño del transistor. Existen además otros factores que juegan un papel importante en la operación de los transistores a nivel molecular imponiendo serias barreras y retos

tecnológicos. Por ejemplo, en tecnologías menores al nodo de 65nm, mismo que entró en producción masiva en el año 2007, el grosor de la capa de óxido de silicio (SiO_2) de la compuerta del transistor se ha reducido a menos de 1nm (aproximadamente 5 átomos de espesor). A estas dimensiones, el dieléctrico de la compuerta experimenta importantes corrientes de fuga desde la compuerta al sustrato o canal y la resultante degradación del óxido de la compuerta. Una de las soluciones potenciales a este problema es el reemplazo del óxido de silicio por un material con mayor constante dieléctrica (k). Aislantes con alta k podrían depositarse en un espesor equivalente al del SiO_2 y reducir las corrientes de fuga de la compuerta al tiempo de mantener el campo eléctrico requerido. Paralelamente a la opción de materiales de alta k , se investiga actualmente el uso de compuertas metálicas o de polisilicio metalizado [3]. Así mismo, mientras menor es la longitud de la compuerta en un MOSFET, más cercanas se encuentran entre sí las regiones de drenaje y fuente, lo que ocasiona que electrones puedan saltar de una hacia la otra aún cuando el transistor supone estar en estado de apagado o corte. Las fugas de corriente no solamente incrementan dramáticamente el consumo de potencia en chips con millones de transistores, sino que además resultan en mayor disipación de calor, degradación de la señal y hasta pérdida de información en memorias DRAM [4]. A fin de obtener un mejor control electrostático del flujo de corriente a través del canal del transistor y evitar las corrientes de fuga entre drenaje y fuente, se están investigando actualmente arquitecturas disruptivas que pueden significar la aparición de transistores con múltiples compuertas tales como los llamados FinFETs o los MOSFETs de doble compuerta [5]. Probablemente el problema más crítico con la miniaturización de los dispositivos de estado sólido radica en los nanoscópicos cables metálicos que realizan las interconexiones del circuito. A esta escala, la capacidad de conducir corriente se ve severamente limitada, la resistencia de los cables ha incrementado en la proporción que su tamaño ha disminuido. Aunado al incremento en la resistencia, se tiene también un incremento en la capacitancia parasítica de los cables al reducirse

también su distancia respecto del sustrato y de los planos de tierra. La combinación de resistencia y capacitancia parasíticas tiene un impacto negativo en la velocidad máxima a la que puede operar el circuito ya que las conexiones actúan como un filtro pasabajos. Al coexistir los cables tan cerca unos de los otros (separados por tan solo unas decenas de nanómetros) se incrementa también la capacitancia entre ellos dando lugar a interferencia y acoplamiento de señales (ruido tipo *crosstalk*). Al contrario del dieléctrico usado en las compuertas de los transistores el cual requiere una alta k , en el caso de las interconexiones se requiere el desarrollo de un aislante con baja k que minimice la capacitancia parasítica entre metales de tal forma que se pueda incrementar la velocidad de las señales. Además, la baja resistencia de las interconexiones conlleva problemas graves de electromigración (EM) por la alta densidad de corriente que fluye a través de ellas. El fenómeno de la electromigración ocasiona un daño progresivo en los conductores metálicos del circuito integrado que puede derivar finalmente en la ruptura del cable o incluso en un corto circuito. Por último, otro de los fenómenos asociados con la física y geometría de los dispositivos en la escala nanométrica es la aparición del comportamiento cuántico de los materiales, el cual debe ser tomado en cuenta durante el diseño físico del circuito debido a que los macro-modelos utilizados actualmente para su simulación no pueden predecir adecuadamente estas situaciones.

También como resultado de la miniaturización de los dispositivos de estado sólido, los modernos circuitos integrados son sistemas extremadamente complejos que pueden contener hasta 1,000 Millones de transistores en la actualidad. Este incremento en la densidad de componentes impone un reto adicional al diseño, ya que para obtener una buena rentabilidad en la producción solo se puede admitir un cierto número de chips defectuosos en cada oblea de silicio. Si se considera el caso de un circuito con 1,000 Millones de transistores, esto representa un requerimiento de control de fallas superior a una parte por cada mil millones (1ppmm); mucho más exigente que cualesquier otra industria.

Es por todo lo anterior que el diseño físico de circuitos integrados ha cobrado una enorme importancia en el ciclo de diseño de un circuito integrado (especialmente en circuitos analógicos y de señal mixta). Ya que, ultimadamente, es en la etapa de diseño físico donde se definen los elementos parasíticos del circuito y donde se puede ejercer control sobre ciertas variables que determinarán la rentabilidad, la “manufacturabilidad”, y el desempeño general del microchip.

II. MÉTODO

Los modernos procesos litográficos utilizados en la fabricación de CIs se valen de una serie de recursos tecnológicos conocidos como Técnicas de Aumento de Resolución o “RET” para poder imprimir patrones menores a la mitad de la longitud de onda de la fuente de luz utilizada ($\lambda=193\text{nm}$). Las técnicas RET incluyen corrección de proximidad óptica (OPC), máscaras de corrección de fase (PSM), iluminación fuera de eje (OAI), doble impresión de patrones (DP), litografía de inmersión, entre otras [6]. A pesar de todos los recursos empleados, el proceso litográfico de manufactura introduce una variedad de efectos que hacen que el tamaño de los componentes difiera de manera significativa con respecto a los patrones diseñados originalmente; el proceso de fabricación puede inducir efectos relacionados con la interacción de componentes parasíticos, de compresión mecánica, variación de las características del material, fenómenos relacionados con la presencia de regiones de alta temperatura o “hot spots”, electromigración, etc.

Para diseñar un buen layout se deben entender las variaciones que existen durante el proceso de fabricación y cómo es que impactan a los dispositivos. Así, dos de las máscaras más importantes en el layout son las que definen la región activa y la del polisilicio de la compuerta. La intersección de estas dos máscaras es la que conforma la región del canal de un transistor MOS. En la figura 1 se muestra una versión simplificada de un transistor MOS y su layout equivalente.

La aparición de un gradiente debido a la falta de uniformidad en la distribución de los dopantes a lo

largo de la superficie de la oblea de silicio se ha convertido también en un problema crítico. El análisis por métodos determinísticos necesita ser reemplazado por análisis estadísticos y tomando en cuenta el comportamiento mecánicamente cuántico de los materiales. Por ejemplo, la figura 2 presenta el caso de cuatro inversores con las mismas dimensiones W y L . Mientras que el layout de la figura 2(a) presenta un inversor diseñado independientemente, la figura 2(b) muestra un arreglo de tres de estos inversores colocados uno tras del otro. El voltaje de umbral, V_{th} , de los transistores en el arreglo es diferente de aquel que es diseñado individualmente hasta en más de un 7%. Dicha variación es función directa del layout, ya que los dispositivos que están dispuestos uno junto al otro experimentan diferentes condiciones de frontera comparados con el que se diseña individualmente (como el área efectiva de las regiones de drenaje y fuente, y la proximidad de la trinchera de asilamiento que define el límite de la región activa), lo cual aunado al efecto del gradiente afecta el perfil de dopaje.

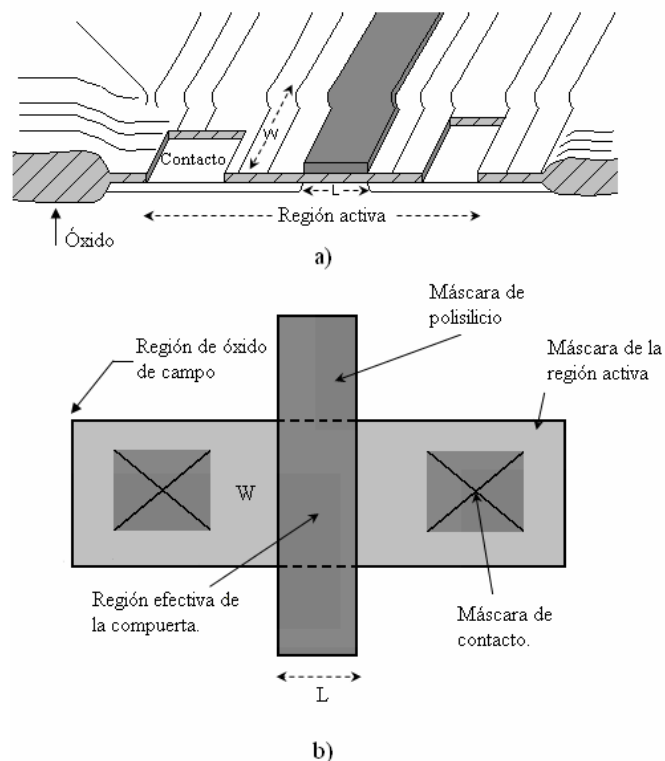


Figura 1: a) Corte transversal de la estructura física de un transistor MOSFET b) Layout de un transistor MOSFET.

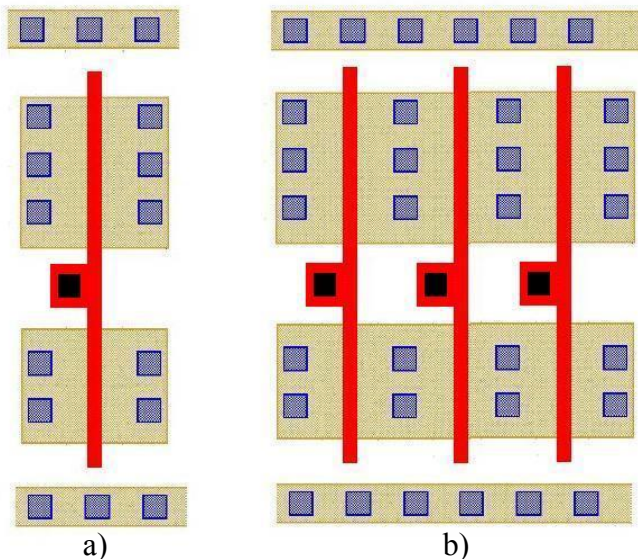


Figura 2: a) Layout de un inversor independiente. b) Layout de un arreglo de 3 inversores.

Uno de los fenómenos que ha cobrado gran relevancia en procesos nanométricos es el llamado de proximidad de contenedor o “well” (WPE). Durante el proceso de fabricación de regiones con un dopaje específico (por ejemplo formación de regiones o wells tipo-N en un sustrato tipo-P), los iones dopantes pueden rebotar en las paredes de la fotorresistencia usada para definir el límite del well, situación que incrementará la concentración de dopaje en la zona cercana a la fotorresistencia o límite del well como se muestra en la figura 3. Con el fin de evitar que transistores contenidos dentro del well exhiban variaciones importantes en el su voltaje de umbral y otras características eléctricas, el layout deberá tomar en cuenta el efecto del WPE evitando colocar dispositivos en la periferia cercana a la pared de éste [7]. Por ejemplo, en procesos de 65nm se recomienda mantener una distancia de los transistores a la orilla del well de por lo menos 1.5µm.

Uno más de los fenómenos derivados de la nanofabricación es el llamado efecto de trinchera de aislamiento (STI). La trinchera de aislamiento fue desarrollada para reemplazar al óxido de campo empleado en anteriores procesos micrométricos para aislar unos transistores de otros.

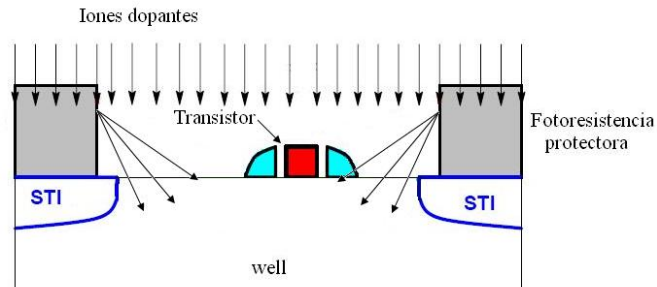


Figura 3: Representación del fenómeno de proximidad de well (WPE).

La trinchera consiste básicamente de una cavidad abierta en el sustrato y rellena de SiO₂ que delimita la región activa de un determinado transistor o grupo de transistores. El efecto de trinchera es básicamente un efecto piezoeléctrico que ocurre por el esfuerzo mecánico que la trinchera ejerce sobre la región activa del transistor. El efecto de trinchera afecta la movilidad de los electrones y por tanto la conductividad del silicio que se encuentra bajo compresión/tensión hasta en un 20% (el efecto es opuesto en regiones tipo-P y tipo-N) lo que resulta en desviaciones del valor de la corriente de saturación I_{dsat} y del voltaje de umbral V_{th} . Ya que el efecto de STI no es uniforme y disminuye con la distancia a la orilla de la trinchera, las características del transistor dependen una vez más de su posición en el layout. Para aliviar este problema se recomienda extender la región activa un mínimo de 1.5µm extra. Además, con la finalidad de mantener las mismas condiciones de frontera en todos los transistores, se sugiere colocar “transistores falsos” o “dummies” en la extensión de la región activa que guarda la distancia extra del STI (ver figura 4).

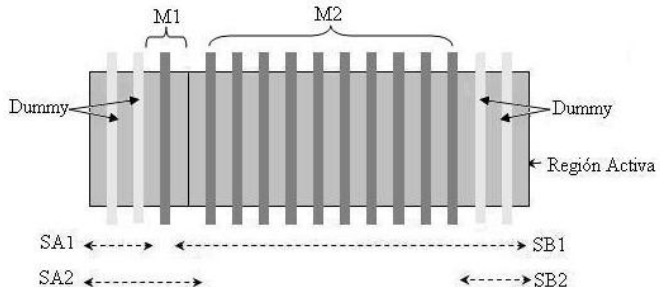


Figura 4: Layout diseñado considerando el efecto STI caracterizado por las dimensiones de las extensiones SA y SB.

Se ha demostrado que el efecto de STI sobre las características eléctricas del transistor puede ser descrito cualitativamente usando los parámetros geométricos SA y SB que representan la distancia de la compuerta del transistor de interés hacia cada una de las orillas de la región activa [8]. Se ha demostrado que los parámetros del transistor tales como el voltaje de umbral V_{th} , la transconductancia g_m , y la corriente de saturación I_{dsat} varían linealmente con el esfuerzo dado por:

$$Esfuerzo = \frac{1}{SA + L/2} + \frac{1}{SB + L/2} \quad (1)$$

La figura 5 ofrece una gráfica de la cantidad de esfuerzo mecánico ejercido por las trincheras sobre la región activa como función de las distancias SA y SB.

Los efectos de proximidad no se limitan al WPE y STI, sino que la sola presencia de otros dispositivos en el layout que se encuentren en las proximidades del circuito de interés afecta el desempeño del mismo al introducir esfuerzos y condiciones de frontera diferentes a lo largo de su perímetro. Los efectos de proximidad (SPE) se vuelven más prominentes conforme la tecnología decrece ya que el rango de interacción entre dispositivos es de hasta aproximadamente $2\mu\text{m}$. La figura 6 ilustra el rango de alcance del efecto de proximidad sobre una celda determinada.

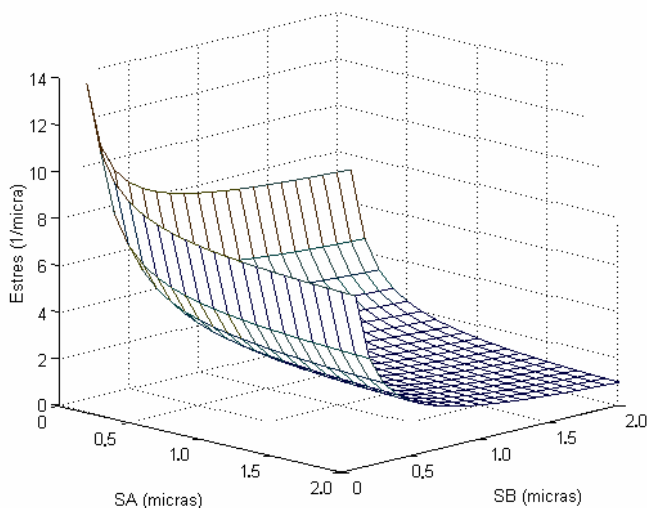


Figura 5: Esfuerzo mecánico por el efecto STI como función de las distancias SA y SB para una tecnología de L=90nm.

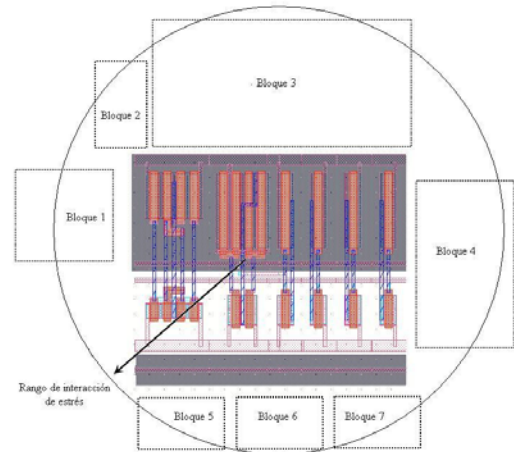


Figura 6: Representación del rango de alcance del efecto de proximidad por esfuerzo mecánico ejercido por elementos periféricos en el layout.

Otro ejemplo del efecto de proximidad es el comentado anteriormente respecto del arreglo de inversores mostrado en la figura 2. La solución a los problemas de proximidad en la periferia de cada celda de layout consiste en armonizar o igualar las condiciones de frontera de circuitos críticos cuidando que todos los elementos en su periferia sean iguales de tal forma que el esfuerzo inducido sea el mismo y uniforme a lo largo y ancho de la celda. Cuando esto no es posible debido a que los dispositivos o estructuras en la periferia son de distinta naturaleza, se deben poner estructuras falsas o dummies y guardar una distancia $>2\mu\text{m}$ con respecto al resto de las celdas. La movilidad está relacionada con el esfuerzo a través de la ecuación piezoresistiva:

$$\frac{\Delta\mu}{\mu_0} = k_x \cdot \sigma_{xx} + k_y \cdot \sigma_{yy} + k_z \cdot \sigma_{zz} \quad (2)$$

Un efecto muy común que sucede durante la fabricación de transistores se conoce como “sobre remoción” y resulta cuando se elimina de manera indeseada parte del polisilicio y metal de ciertas estructuras a pesar de las diferentes capas protectoras que se aplican durante el proceso de manufactura.

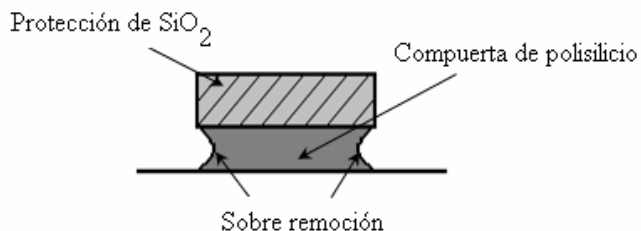


Figura 7: El efecto de sobre remoción en la compuerta de un transistor MOSFET.

La figura 7 muestra el efecto de sobre remoción sobre una compuerta que es más pequeña de lo que debería ser (el polisilicio debería ser grabado al ras de la capa protectora de SiO₂). Las variaciones en las dimensiones de las compuertas introducen cambios en el voltaje de umbral del transistor; de manera experimental se ha encontrado que:

$$S_{V_t} = \frac{C_{V_t}}{\sqrt{W_{eff} L_{eff}}} \quad (3)$$

donde S_{V_t} es la desviación estándar de las variaciones del voltaje de umbral y C_{V_t} es una constante [9, 10]. El valor de C_{V_t} se determina de forma empírica midiendo las variaciones aleatorias entre pares de transistores de diferentes tamaños y, por lo tanto, los resultados sólo aplican a los transistores que tienen similitudes con los utilizados para encontrar el valor de C_{V_t} . Así, en el layout analógico de precisión donde resulta de primordial importancia la armonización entre dispositivos, se debe mantener las mismas condiciones en la periferia del circuito para que los efectos de sobre remoción sean similares en los dispositivos críticos del diseño.

Como se mencionó anteriormente, los efectos de gradiente tienen un gran impacto en el rendimiento de los circuitos, y se refiere a todas aquellas variaciones y cambios en la homogeneidad de ciertas variables que presenta la oblea durante su fabricación (tales efectos incluyen los cambios de estrés, de temperatura, y cambios del grosor del óxido de silicio de la compuerta en distintas áreas del microcircuito). Para minimizar los efectos de gradiente es sumamente importante cuidar la simetría del circuito (incluso la orientación de los dispositivos) y, del mismo modo que con los

efectos de sobre remoción, se deben mantener las mismas condiciones de frontera en la periferia de los circuitos sensibles. Así, cuando se requiere de armonización de precisión entre dos transistores, cada transistor no sólo se realiza utilizando transistores unitarios sino que cada dedo de un transistor se “interdigita” con los dedos del segundo transistor. Esta práctica se conoce como “layout de centroide común” (ya que los dos dispositivos comparten su centroide a través de los ejes de simetría de la estructura) lo que ayuda a disminuir los efectos del gradiente a través del microcircuito [11]. Lo anterior se muestra en la figura 8.

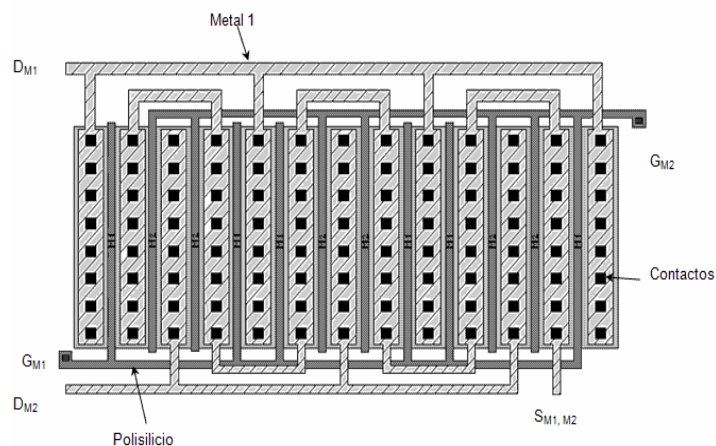


Figura 8: Estructura de dos transistores multidedo utilizando la técnica de centroide común.

Como se observa en la figura 8, cada transistor está compuesto por seis dedos conectados en paralelo. Los dedos internos se acomodan por pares (dos dedos para M1 y dos dedos para M2). Así, el layout es simétrico para los ejes X y Y de manera que cualquier gradiente a través del microcircuito afectará de igual forma a ambos transistores. Esta estructura es muy utilizada en el diseño físico de pares diferenciales donde las terminales D_{M1} , G_{M1} , D_{M2} y G_{M2} de la figura 8 son los drenajes y compuertas de los dos transistores que componen al par diferencial y $S_{M1,M2}$ es la fuente común del arreglo.

Un punto importante a considerar en layout nanométrico es el de rellenar inteligentemente los espacios vacíos con patrones metálicos a fin de cumplir con las densidades mínimas exigidas por

el proceso de fabricación. Tradicionalmente el rellenado de metal no se consideraba una tarea crítica, sin embargo, actualmente se pueden tener problemas serios de manufacturabilidad y rendimiento si la densidad de metal no es suficientemente uniforme a través del chip. Lo anterior se debe a que diferencias locales en la densidad de metal afecta el proceso de planarización que se efectúa con cada capa de metal que se deposita sobre el circuito (los procesos actuales pueden tener más de 10 capas de metal). El proceso de planarización se lleva a cabo utilizando la técnica de pulido químico-mecánico o CMP; si la superficie de cada capa de metal y de su aislante no está lo suficientemente plana y uniforme pueden introducirse variaciones significativas en la resistencia y capacitancia de las interconexiones. Así pues, el proceso de rellenado de metal debe hacerse de tal forma que se obtenga una densidad homogénea sobre la superficie del chip.

Algunos conceptos de layout que han sido introducidos recientemente en procesos nanotecnológicos incluyen el análisis de área crítica (CAA), y el diseño de layout compatible con las técnicas litográficas de OPC. El análisis de área crítica trata sobre la sensibilidad de un diseño determinado a los efectos causados por partículas aleatorias que interfieren con el proceso de fabricación y que pueden causar circuitos abiertos y corto circuitos principalmente. Entre más pequeñas son las estructuras en el layout más crítico se vuelve el efecto de una determinada partícula. El análisis de CAA se realiza con base en modelos estadísticos pero también es posible prevenir defectos evitando patrones demasiado densos en el layout, promoviendo la simetría, optimizando distancias y número de contactos en geometrías críticas, etc. Por otra parte, el hacer que un layout sea compatible con las técnicas de OPC reduciendo el riesgo de defectos de manufactura, requiere de un conocimiento un tanto más profundo de las tecnologías RET y el proceso de fabricación. La regla general consiste en evitar ciertos patrones geométricos y espaciamientos que se sabe de manera empírica son propensos a introducir errores litográficos por su poca afinidad con los algoritmos computacionales de OPC.

Por último, para demostrar experimentalmente algunas de las técnicas descritas para el diseño físico de CIs, en especial las referentes a sobre remoción, el uso de transistores dummy y la igualación de las condiciones de frontera, se diseñaron y fabricaron varias estructuras simples con compuertas de polisilicio según se describe a continuación.

Los dispositivos se diseñaron empleando una herramienta CAD especializada para el diseño físico de CIs y se fabricaron usando un proceso estándar de micromaquinado superficial sobre un sustrato de silicio. La figura 9 muestra el layout del arreglo de dispositivos diseñado para realizar mediciones experimentales y una imagen de microscopio electrónico (SEM) de la estructura fabricada. De la figura 9, las estructuras A y B se diseñaron con la resolución mínima del sistema mientras que el resto de las estructuras están diseñadas con una longitud de compuerta de $4\mu\text{m}$.

Todas las estructuras presentadas en el arreglo, tienen un ancho de $26\mu\text{m}$. La estructura A, mostrada en la figura 10, está diseñada con una longitud de compuerta de $2\mu\text{m}$. Se fabricó una sola compuerta para poder medir la variación en las dimensiones efectivas del dispositivo por los efectos de gradiente y, en especial, por la sobre remoción durante el proceso de fabricación de un transistor sin dispositivos dummy a sus lados.

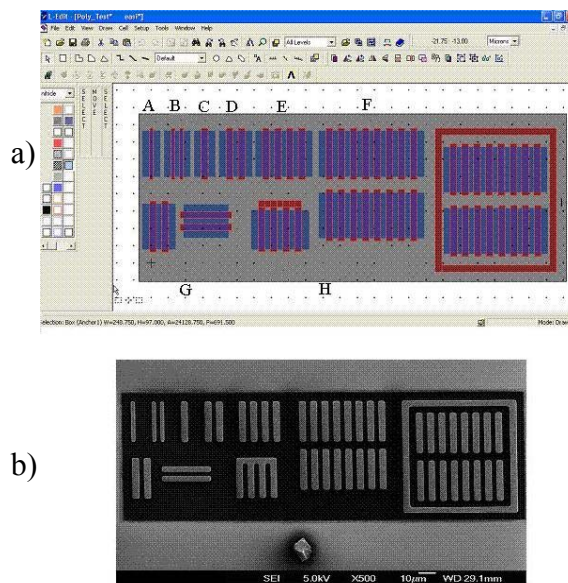


Figura 9: a) Layout del chip diseñado b) Microimagen SEM de la estructura fabricada.

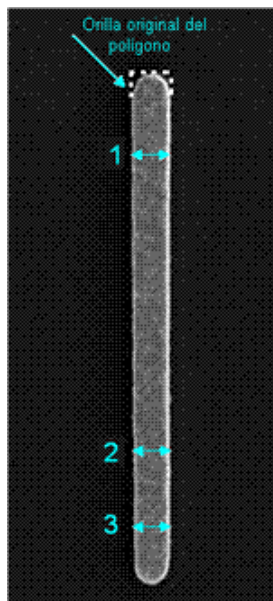


Figura 10: Imagen SEM de la estructura A.

En la figura 10 se observa con gran detalle como la sobre remoción hace que el transistor tenga sus extremos redondeados en lugar de rectangulares, también se han marcado tres puntos en la compuerta del transistor donde las dimensiones efectivas varían. El punto número 1 es la dimensión más cercana al valor original diseñado y se usará como el valor de referencia para medir el resto de las variaciones. Este valor se mantiene hasta la región marcada con el número 2; en este punto se tiene una disminución del valor de referencia de un 9.1% mientras que en el punto 3 la longitud del canal varía hasta un -13.6% respecto al valor de referencia y es donde se presenta la mínima longitud del canal. Estas variaciones en el largo de la compuerta del transistor, introducen cambios en el voltaje de umbral y en la transconductancia del transistor.

En la estructura B se diseñaron dos compuertas con la resolución mínima para medir las variaciones en las orillas de un transistor con factor M igual a 2, es decir 2 dedos. Así mismo, la estructura servirá para medir los efectos de la sobre remoción que existe en el espacio entre los dedos de los transistores. La figura 11 muestra la imagen de microscopio electrónico de la estructura B; nótese el efecto de redondeo sobre las orillas de los transistores en la misma proporción que en el dispositivo de la figura 10.

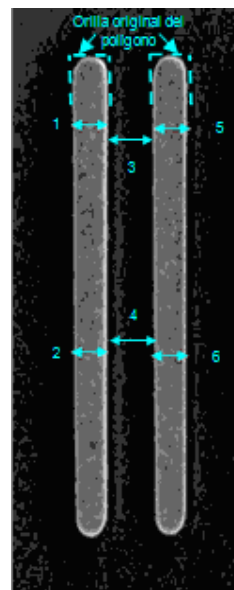


Figura 11: Imagen SEM de la estructura B.

Al igual que en el caso de un solo transistor, se marcaron diferentes puntos de interés en las compuertas. Los puntos 1 y 5 en la parte superior de la compuerta presentan el valor más cercano al definido en el layout (y serán utilizados como los valores de referencia). En los puntos 2 y 6 de la imagen, la longitud del canal decrece un 7.8% respecto al valor de referencia de los transistores. Las variaciones fueron iguales en ambos dispositivos y no existe una tercera variación como en el caso de un solo transistor (existe mayor uniformidad).

La variación en la longitud de las compuertas en la parte interna de los dedos (puntos 3 y 4) sirve para comprobar que el efecto de sobre remoción es más severo en las paredes externas de los transistores que en las caras internas. Del total de la disminución de la longitud efectiva del transistor, la sobre remoción de las paredes externas contribuye con un 68% de la variación total de la dimensión del transistor.

Para las estructuras C y D de la figura 12, se realizó el mismo procedimiento de medición que para las estructuras A y B.

La primer observación importante es que las estructuras presentan un borde plano en la orilla de la compuerta del transistor, lo indica que el efecto de sobre remoción es menos pronunciado en compuertas de mayor tamaño. Esto comprueba la conveniencia de la técnica de no usar dimensiones

mínimas en el diseño de dispositivos críticos. De manera similar que en las estructuras A y B, los valores más cercanos al valor definido en el layout se encuentran en la parte superior de la compuerta, lo que sugiere la presencia de un efecto de gradiente en el eje vertical y por lo que se comprueba que transistores que requieren armonizarse deberán diseñarse usando la misma orientación en el layout.

Las mediciones indican que la estructura C, tiene una disminución en la longitud efectiva de la compuerta de aproximadamente 6.5% entre el valor de referencia y el punto 2, y en esta estructura no existe un tercer punto de variación considerable. Para la estructura D, la variación en ambas compuertas del valor de referencia en los puntos 4 y 8 es de 5.8% y la sobre remoción de las paredes externas de las compuertas contribuye con el 73% de la desviación con respecto a la dimensión de referencia en los puntos 3 y 7.

La figura 13 muestra la estructura E que consiste de un transistor de dos dedos (M=2) pero que incluye un transistor dummy a cada lado para igualar las condiciones de frontera de los dedos internos y armonizar el efecto de la sobreremoción. Con la inclusión de los transistores dummy en los extremos del arreglo de compuertas, se tuvo una disminución en la longitud de las dos compuertas internas, respecto a los valores de referencia, de aproximadamente 5.5% mientras que en los transistores dummy la disminución fue de 8.33%.

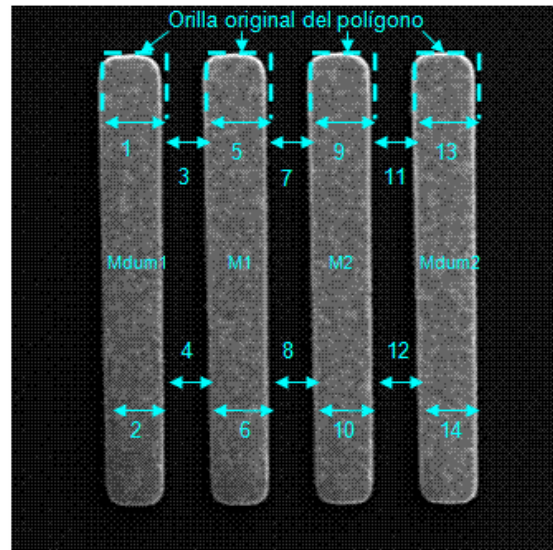


Figura 13: Imagen SEM de la estructura E.

El espaciamiento entre los dedos de los transistores tuvo una variación de 8.33% en todo el arreglo (puntos 4,8 y 12). Con esto se comprueba que los transistores dummies (Mdum1 y Mdum2) sirven su propósito de absorber las variaciones en los extremos mientras que los transistores reales (M1 y M2) quedan armonizados.

La estructura G se diseñó para medir las variaciones en la longitud de la compuerta debido a la orientación de los dispositivos. Para ello se hizo el layout de dos transistores verticales y dos transistores horizontales como se muestra en la figura 14.

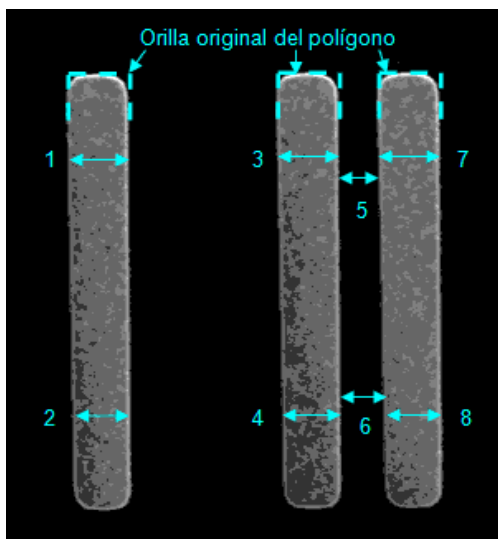


Figura 12: Imagen SEI de la estructura C y D.

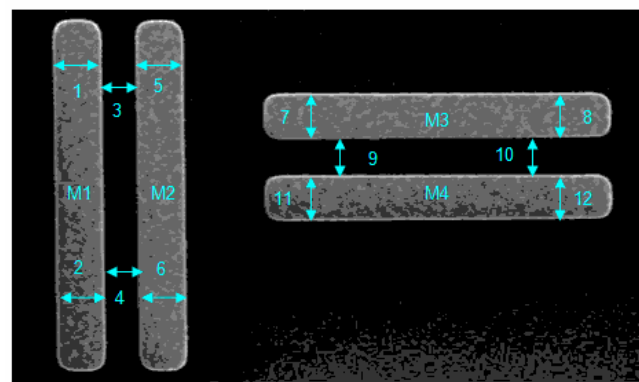


Figura 14: Imagen SEM de la estructura G.

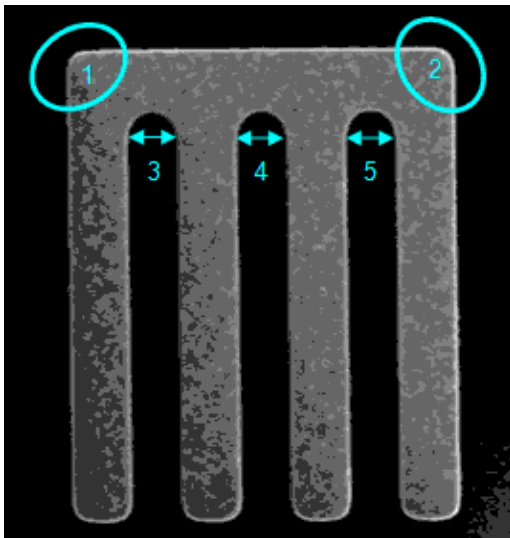


Figura 15: Imagen SEM de la estructura H.

Los resultados sobre las estructuras verticales fueron muy similares a los obtenidos en el caso de la estructura D, esto es, la dimensión efectiva de las compuertas varió alrededor del 5.7%. De igual forma, la sobre remoción es más pronunciada en la parte externa de los transistores contribuyendo en un 74% del total de la variación presentada en los transistores.

En el caso de los transistores horizontales, los valores de referencia (los cuales coincidieron con los transistores verticales) se localizaron en la parte izquierda del transistor (puntos 7 y 11 de la imagen) y la variación máxima presentada fue de 4.76%, lo que indica que aunque la distancia entre los dispositivos es relativamente corta, la orientación de los transistores es un factor importante en la variación de parámetros.

Para observar como se afecta la armonización de los transistores cuando las compuertas de una estructura multidedo se unen con polisilicio, se diseñó la estructura H la cual tiene 4 transistores conectados en forma de “peine”. La figura 15 muestra la imagen SEM de la estructura fabricada.

Como se observa, los bordes rectangulares del layout desaparecen en la estructura fabricada y, aunque en las orillas inferiores de los transistores se tienen partes planas, en las separaciones internas (puntos 3, 4 y 5) se tienen bordes redondeados como los presentes en las estructuras fabricadas con la mínima resolución.

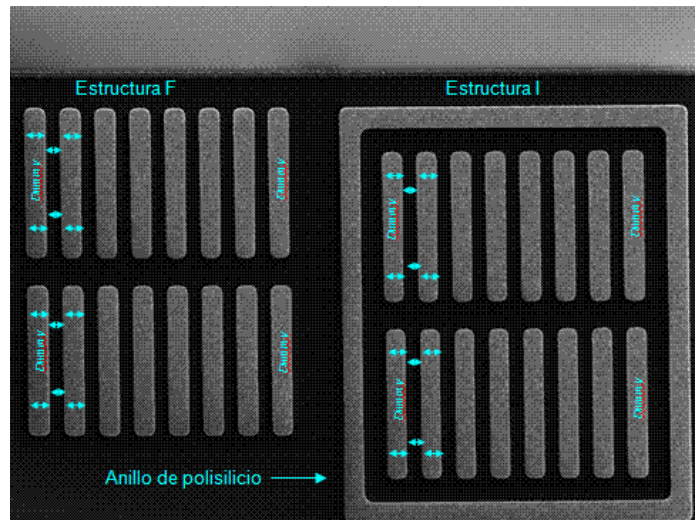


Figura 16: Imagen SEM de la estructura F e I.

Las principales causas que afectan la homogeneidad del arreglo son la diferencia que existe en la cantidad de polisilicio de cada compuerta; esto es, las compuertas de los extremos presentan una menor área de polisilicio debido a la sobre remoción en la esquinas superiores (marcadas como 1 y 2 en la figura) y, por otro lado, las variaciones en la remoción de los bordes internos (puntos 3, 4 y 5). Estas diferencias introducen variaciones en el comportamiento eléctrico de la compuerta, de ahí la recomendación de realizar las conexiones entre las compuertas de una estructura multidedo en el metal más bajo disponible en el proceso.

Por último las estructuras F e I de la figura 9 se diseñaron con el objetivo de ver si existen diferencias entre un arreglo de transistores que presenta las mismas condiciones de frontera en toda su periferia y uno que se encuentra libre (sin igualar sus condiciones de frontera). La figura 16 muestra la imagen SEM de las estructuras.

Ambas estructuras constan de dos renglones de 8 dedos cada uno. Los transistores en los extremos de cada renglón son dispositivos dummy para mejorar la armonización de los arreglos. Para igualar las condiciones de frontera de la estructura I, se colocó un anillo de polisilicio con un ancho de $4\mu\text{m}$ a su alrededor.

Los resultados en ambos renglones de la estructura F (la que no tiene anillo), son muy similares a los obtenidos en la estructura E donde los transistores dummy ayudaron a mantener la armonización del

arreglo. La disminución del ancho efectivo de la compuerta en los dedos internos de ambos renglones fue de alrededor del 5.5% respecto al valor de referencia mientras que para los dispositivos dummy fue 8.33%. Los efectos de gradiente y de sobre remoción a lo largo del eje Y no fueron perceptibles ya que ambos renglones de transistores presentaron las mismas variaciones en sus compuertas.

En el caso del arreglo de transistores de la estructura I, se obtuvieron mediciones muy similares en la disminución del tamaño de la compuerta efectiva y en la separación interna de todos los transistores incluyendo los dummies. En este caso todos los dispositivos del arreglo dentro del anillo experimentaron los mismos efectos de sobre remoción. Este experimento puede usarse como ejemplo de cómo la técnica de igualar las condiciones de frontera puede ser efectiva para armonizar otros efectos de proximidad como por ejemplo el esfuerzo mecánico.

Aún y cuando el proceso de fabricación utilizado para la medición de los resultados experimentales es de dimensiones relativamente grandes, se pudieron medir y comprobar muchos de los fenómenos que siguen afectando hoy en día a los transistores fabricados en los más modernos procesos a escalas nanométricas.

III. CONCLUSIONES

Conforme la industria de los semiconductores se adentra en la era nanotecnológica, la complejidad y relevancia del layout de circuitos ha aumentado considerablemente. Los resultados obtenidos experimentalmente a partir del prototipo fabricado, permitieron cuantificar y comprobar el efecto de diversas técnicas de diseño físico en las que mínimas modificaciones en el arreglo o geometría de los dispositivos pueden producir cambios dramáticos en el desempeño final del sistema.

El diseño físico de CIs se ha convertido en una tarea multidisciplinaria donde se involucran diversos dominios energéticos (eléctrico, mecánico, térmico, magnético, etc.) y requiere de un alto grado de especialización ya que las herramientas de software utilizadas para el diseño, a pesar del alto grado de automatización, no pueden identificar muchos de los problemas que

ocurren en el diseño de CIs en procesos nanométricos.

IV. REFERENCIAS

- [1] *Nanotechnology definition*, National Science Foundation, Febrero de 2000, en: www.nsf.gov/crssprgm/nano/reports/omb_nifty_50.jsp
- [2] S.E. Thompsona, S. Parthasarathy, “*Moore's law: the future of Si microelectronics*”, Elsevier Materials Today, Vol. 9, No. 6, pp. 20-25, 2006.
- [3] R. Chau et al., “*High-k/Metal-Gate Stack and Its MOSFET Characteristics*”, IEEE Electron Device Letters, Vol. 25, No. 6, pp. 408-410, 2004.
- [4] T.Y. Winarski, “*Dielectrics in MOS devices, DRAM capacitors, and inter-metalisolation*”, IEEE Electrical Insulation Magazine, Vol. 17, No. 6, pp. 34-47, 2001
- [5] D. Schmitt-Landsiedel, C. Werner, “*Circuits with innovative devices - challenges and chances*”, IEEE 9th International Conference on Ultimate Integration of Silicon, pp. 81 – 84, 2008.
- [6] F.M. Schellenberg, “*A History of Resolution Enhancement Technology*”, Optical Review, Vol. 12, No. 2, pp.83-89, 2005.
- [7] T.B. Hook, “*Proximity effects and VLSI design*”, International Conference on Integrated Circuit Design and Technology, pp. 167-170, 2005.
- [8] P.G. Drennan, et al., “*Implications of Proximity Effects for Analog Design*”, IEEE Custom Integrated Circuits Conference, pp. 169-176, 2006.
- [9] F.M. A. Hastings, *The Art of Analog Layout*, 2nd Edition, Pearson Prentice Hall, 2006.
- [10] K.R. Lakshmikumar, R.A. Hadaway and M.A. Copeland, “*Characterization and Modeling of Mismatch in MOS Transistors for Precision Analog Design*”, IEEE Journal of Solid-State Circuits, Vol. SC-21, No. 6, pp. 1057-1066, 1986.
- [11] D. Johns and K. Martin, *Analog Integrated Circuit Design*, Wiley, 1997.

CONSTRUCCIÓN DE UN MICROCONTROLADOR PARA APLICACIONES EMPOTRADAS EN FPGAs, A PARTIR DEL NÚCLEO PICOBLAZE.

Santiago Espinosa Felipe, Peña Toledo José.

Instituto de Electrónica y Computación
Universidad Tecnológica de la Mixteca
Carretera a Acatlima Km. 2.5, Huajuapán de León, Oaxaca, 69000, México
Tel. 9535320214, correo electrónico: fsantiag@mixteco.utm.mx, jostoledo555@hotmail.com

RESUMEN

En este artículo se describe la forma en que se acondicionó al procesador de 8 bits denominado Picoblaze (también conocido como KCPSM3), el cual es de distribución libre y fue desarrollado por Ken Chapman; ingeniero de desarrollo de Xilinx, Inc., se le agregaron módulos para el manejo de puertos, temporización e interrupciones, desarrollados en VHDL. Dando como resultado un microcontrolador simple pero funcional, el cual ha quedado disponible para el desarrollo de aplicaciones empotradas en FPGAs.

Palabras clave: Microcontrolador, Picoblaze, KCPSM3, VHDL, FPGA.

I. INTRODUCCIÓN

Picoblaze es el nombre de un microprocesador de 8 bits que puede implementarse en dispositivos Spartan-3, Virtex-II o dispositivos más recientes. Es de distribución libre y puede obtenerse como un componente de VHDL o Verilog, el cual puede ser instanciado y sometido a simulación [1].

Aunque puede ser usado para procesamiento de datos, generalmente es empleado para aplicaciones que requieren una compleja máquina de estados. Un factor importante es su tamaño, ocupa solo el 3% de los recursos en un dispositivo XC3S500 [2].

Requiere de un bloque de RAM para formar una ROM en donde es posible almacenar un programa de hasta 1024 instrucciones. Los programas con requerimientos mayores pueden ser redireccionados de manera que utilicen múltiples

procesadores, cada uno con su respectivo bloque de memoria, para distribuir tareas en diversos sistemas.

Su repertorio de instrucciones contiene 42 instrucciones que incluyen operaciones lógicas, aritméticas, de transferencia y bifurcaciones. Incluye memoria para datos, 16 registros de propósito general, banderas, unidad aritmética lógica (ALU, *Aritmethic Logic Unity*), una interrupción externa y la capacidad de manejar hasta 256 puertos de entrada y 256 de salida [2].

Sin embargo, el Picoblaze es solo un núcleo, para el manejo de periféricos es necesario el acondicionamiento de puertos; y si una aplicación requiere temporización, conteo de eventos o monitoreo de actividades externas, será necesario que estas tareas se asignen al programa principal, requiriendo tiempo de procesamiento.

Por esta razón, el presente trabajo se enfocó al diseño de un bloque de recursos, el cual incluye tres módulos con las funciones de temporización, interrupciones y manejo de puertos, con lo que se incrementaron las capacidades del núcleo [3]. Estos módulos se diseñaron en VHDL, empleando una metodología de diseño descendente [4].

En [5] se encuentra una serie de aplicaciones del Picoblaze, muchas de ellas se simplificarían si se emplea al procesador con los módulos desarrollados. Otras aplicaciones de procesadores empotrados se describen en [6], [7] y [8].

II. ORGANIZACIÓN DEL PICOBLAZE

En la figura 1 se muestra la estructura interna del núcleo y la forma en que se interconecta con la memoria de código [2]. Los módulos del núcleo que destacan son: el archivo de registros, conformado por 16 registros de 8 bits, la memoria interna de datos con 64 localidades, la ALU y el control.

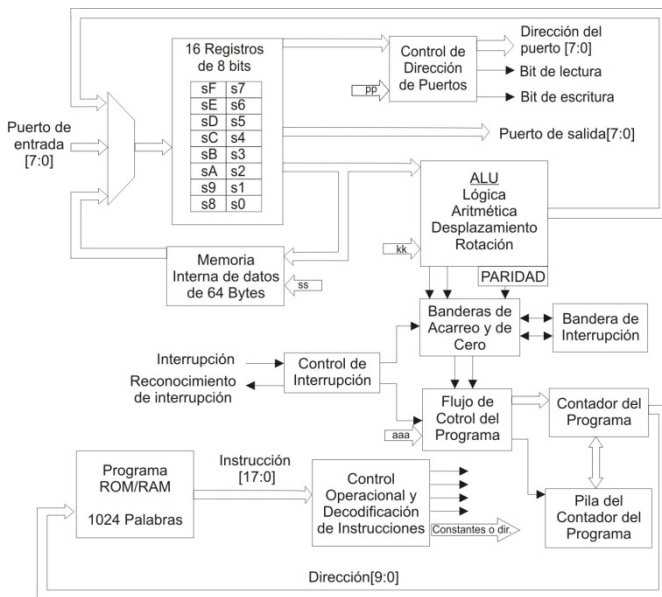


Fig. 1 Organización interna del PicoBlaze

El núcleo maneja una interrupción, la cual es solicitada al poner en alto la señal correspondiente y cuando la interrupción es aceptada por el núcleo, genera una señal de reconocimiento.

También se observan tres buses externos: el puerto de salida, el puerto de entrada y la dirección del puerto, todos de 8 bits. Estos buses son los que proporcionan flexibilidad al núcleo, a través de ellos se realizará la interconexión del bloque de recursos desarrollado. Las señales de lectura y escritura coordinan el flujo de información en estos buses.

La entrada de datos se realiza con la instrucción INPUT, el PicoBlaze coloca la dirección a leer, genera un pulso en el bit de lectura por un ciclo de reloj, lee el dato del bus del puerto de entrada y lo transfiere a un registro especificado. Para la salida de datos se tiene a la instrucción OUTPUT, con

ésta el PicoBlaze coloca la dirección a escribir, ubica un dato en el bus del puerto de salida, proveniente de un registro especificado y genera un pulso en el bit de escritura por un ciclo de reloj.

En la figura 2 se muestra el comportamiento de las instrucciones INPUT y OUTPUT para las operaciones de entrada y salida.

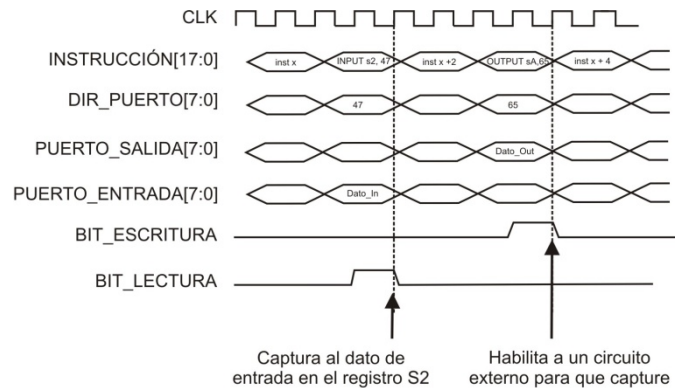


Fig. 2 Entrada y salida de datos del núcleo

III. DISEÑO DE LOS RECURSOS

En la figura 3 se muestra al PicoBlaze, como un bloque, y la forma en que se relaciona con su memoria de código, quedando disponibles las señales para el acondicionamiento de recursos.

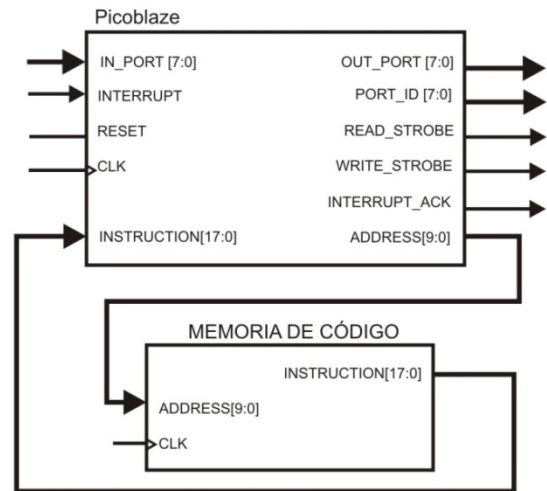


Fig. 3 El núcleo con su memoria de código

El bloque de recursos desarrollado se ubica al mismo nivel jerárquico de estos dos elementos, esto se muestra en la figura 4.

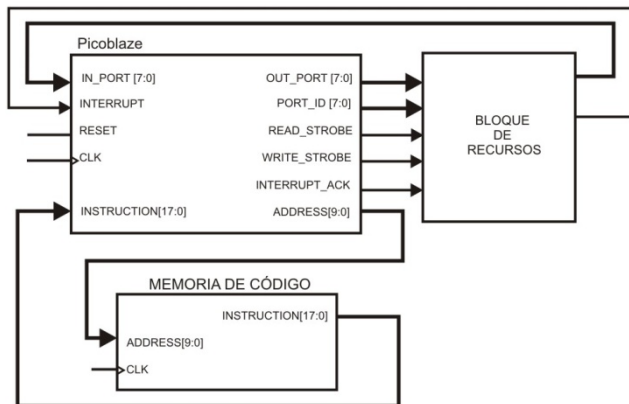


Fig. 4 Ubicación del bloque de recursos

El bloque de recursos se compone de tres módulos: El primero dedicado al manejo de los puertos de entrada y salida; el segundo dedicado al Temporizador/Contador y el último para dar atención a las interrupciones, esto se muestra en la figura 5.



Fig. 5 Módulos que integran al bloque de recursos

Para el manejo de cada uno de los módulos se debe definir un conjunto de registros externos al núcleo. La escritura de los registros permitirá la configuración de los recursos, mientras que su lectura será para conocer su estado actual. El número de registros está en función de las características y funcionalidad que se le atribuirá a cada uno de los módulos.

Módulo de Puertos

Se decidió incluir cuatro puertos de propósito general, de 8 bits, cada uno de ellos podrá configurarse como entrada o salida, de manera independiente, terminal por terminal. Por lo tanto, para cada puerto se requiere la definición de 3 registros:

- Un registro para configuración del puerto, en el que se definirá individualmente, para cada

terminal, si será utilizada como entrada o salida (REG_CONF_x).

- Un registro de entrada, para retener el último valor leído (REG_IN_x).
- Un registro de salida, para retener el último valor escrito (REG_OUT_x).

La *x* indica los posibles valores de 1, 2, 3 o 4. Entonces, para cada puerto se tendrá un bloque como el que se muestra en la figura 6.

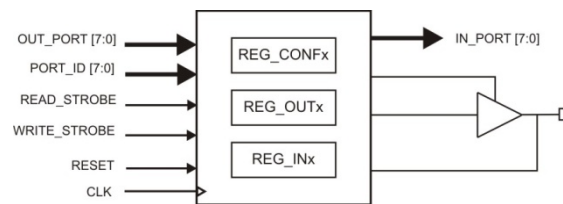


Fig. 6 Organización de los puertos

El buffer de tres estados evita que los datos de entrada colisionen con los que se han escrito en el registro de salida. Cabe aclarar que algunas terminales deberán multiplexarse para soportar una función alterna, como la entrada externa de un reloj para el temporizador o la detección de eventos externos.

Las direcciones que se han dejado disponibles para el manejo de los puertos se muestran en la tabla 1.

REGISTRO	DIRECCIÓN
REG_CONF1	0xF0
REG_IN1	0xF2
REG_OUT1	0xF1
REG_CONF2	0xF3
REG_IN2	0xF5
REG_OUT2	0xF4
REG_CONF3	0Xed
REG_IN3	0xEF
REG_OUT3	0xEE
REG_CONF4	0xEA
REG_IN4	0xEC
REG_OUT4	0xEB

Tabla 1. Registros para el manejo de los puertos

Módulo del Temporizador

Un temporizador básicamente es un registro de 8 bits que se incrementa automáticamente cada ciclo

de reloj o cuando ocurren eventos externos. Cuando el temporizador alcanza su máximo valor y ocurre una transición de éste a cero, genera una señalización [9]. Otras características que se consideraron para el diseño de este módulo fue el manejo de eventos por coincidencia en la comparación del temporizador con otro registro y la generación de señales moduladas por ancho de pulso (PWM, *Pulse Width Modulation*) [10].

El módulo del temporizador se integra de tres componentes, como se ilustra en la figura 7.

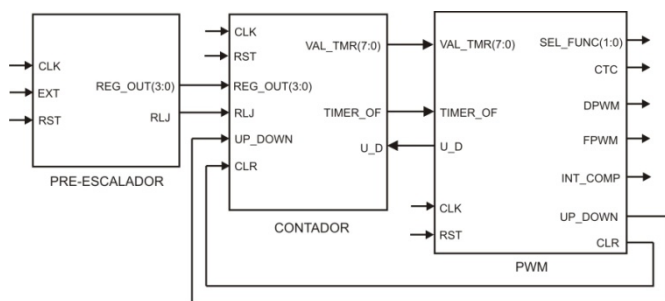


Fig. 7 Organización del módulo del temporizador

El pre-escalador permite alcanzar intervalos de tiempo más grandes, dividiendo la señal del reloj principal por diferentes factores. Se optó por incluir divisiones entre 8, 32, 64, 128, 256, 1024, 2^{17} y 2^{25} , con los últimos factores se alcanzan intervalos en el orden de segundos. La base del pre-escalador es un registro de 25 bits del cual se seleccionarán diferentes posiciones de bits a través de un multiplexor, para el cual también se consideró la anulación del reloj, una salida sin modificación y el manejo de un reloj externo. La salida del pre-escalador se proporciona en la señal RLJ.

Se requiere de un registro para definir el factor de pre-escala de la señal de reloj que se suministrará al contador. Este registro se denomina VAL_REG y se ubica en la dirección 0xFC, prácticamente sólo se emplean 4 de sus ocho bits, como se muestra en la tabla 2.

El contador básicamente es un registro de 8 bits que ascenderá o descenderá automáticamente con la señal RLJ. Este registro se denomina CONT_8B

y se ubica en la dirección 0xFF. Se acondicionó para que pueda ser leído o escrito durante la ejecución de un programa. También se incluyó un noveno bit para la detección de desbordamientos.

VAL_REG	RLJ (salida)
0000	0
0001	Reloj externo
0010	Reloj principal (clk)
0011	clk/8
0100	clk/32
0101	clk/64
0110	clk/128
0111	clk/256
1000	clk/1024
1001	clk/ 2^{17}
1010	clk/ 2^{25}
*Cualquier otro caso	0

Tabla 2. Selección del reloj para el temporizador

El componente de PWM es el que se encarga de las funciones especiales del temporizador, incluye un registro de comparación, denominado VAL_COMP, ubicado en la dirección 0xF9, este registro continuamente se compara con el temporizador y ante eventos de coincidencias por comparación, se realizarán algunas de las siguientes acciones:

- La limpieza del temporizador.
- La conmutación de una terminal de salida.
- Generación de una señal PWM de rampa simple, el contador trabaja de manera ascendente, limpiando una señal de salida ante el evento de comparación y ajustándola ante un desbordamiento.
- Generación de una señal PWM de rampa doble, el contador primero trabaja de manera ascendente y luego descendente, una señal de salida se ajusta al ocurrir la coincidencia ascendente y se limpia con la coincidencia descendente.

Se incluyó un registro para seleccionar estas funciones especiales. El registro se denomina CONF_COMP y se ubica en la dirección 0xF8, los

bits que se han implementado para este registro se muestran en la tabla 3.

CONF_COMP (0xF8)		
BIT	NOMBRE DEL BIT	DESCRIPCION
0	-	-
1	CTC	Habilita la función de limpieza del contador por comparación.
2	FPWM	Habilita la función PWM de rampa sencilla.
3	DPWM	Habilita la función del PWM de doble rampa.
4-7	-	Sin uso

Tabla 3. Definición de bits para el registro CONF_COMP

En la figura 8 se observa la forma es que se multiplexa una de las terminales del puerto 3 para soportar la función alterna de PWM de rampa sencilla, los bits de selección de función se determinarán por el contenido del registro CONF_COMP.

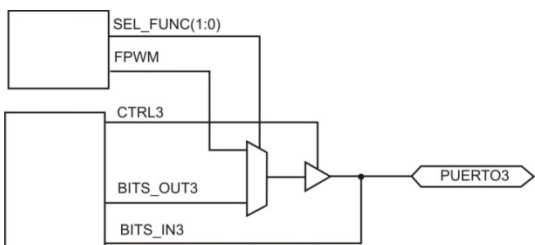


Fig. 8 Multiplexado de salidas para agregar funciones alternas

En la tabla 4 se muestra un resumen de los registros que se han definido para el módulo del temporizador.

REGISTRO	DIRECCIÓN
CONT 8B	0xFF
VAL_REG	0xFC
VAL_COMP	0xF9
CONF_COMP	0xF8

Tabla 4. Registros para el manejo del temporizador

Módulo de Interrupciones

Por medio de este módulo se podrán detectar los siguientes eventos:

- Interrupción externa.

- Interrupción por comparación.
- Interrupción por desborde.

Para ello, este módulo incluye dos registros, uno de configuración, denominado INT_CONF con dirección 0xFB y otro de estado, llamado INT_STATE ubicado en 0xFA. Ambos registros pueden leerse o escribirse por software, sin embargo, el registro de estado también se escribe automáticamente por hardware.

El núcleo incluye una entrada para el manejo de una interrupción externa, ésta se ha acondicionado por hardware para que pueda ser disparada por diferentes eventos, que deben ser habilitados en el registro de configuración. Por software se podrá determinar la fuente de la interrupción, a partir del contenido del registro de estado.

En la figura 9 se observa al módulo de interrupciones conformado por dos componentes, uno para la detección de los eventos y solicitud para su atención por parte del núcleo, y el otro para seleccionar como se detectarán los eventos externos, si serán flancos de subida o bajada, o niveles alto o bajo.

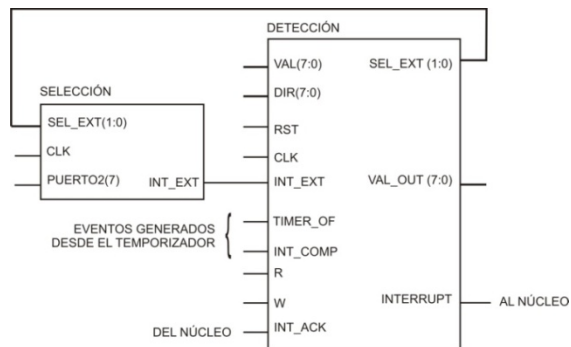


Fig. 9 Organización del módulo de interrupciones

La tabla 5 muestra la organización del registro de configuración y la tabla 6 corresponde al registro de estado.

INT_CONF (0xFB)		
BIT	NOMBRE DEL BIT	DESCRIPCION
0	HIS	Habilita la interrupción por desbordamiento del temporizador
1	HIE	Habilita la interrupción externa

2	HIC	Habilita la interrupción por coincidencia en la comparación
3	-	-
4	BS1	Configuran la interrupción externa: 00 - Por flanco de subida. 01 - Por flanco de bajada. 10 - Por nivel lógico alto. 11 - Por nivel lógico bajo.
5	BS0	
6	-	-
7	HGI	Habilitador global de interrupciones

Tabla 5. Definición de bits para el registro INT_CONF

INT_STATE (0xFA)		
BIT	NOMBRE DEL BIT	DESCRIPCION
0	BIS	Bandera de interrupción por desbordamiento del temporizador
1	BIE	Bandera de interrupción externa
2	BIC	Bandera de interrupción por comparación
3-7	-	Sin uso

Tabla 6. Definición de bits para el registro INT_STATE

IV. IMPLEMENTACIÓN

Todos los módulos se desarrollaron en VHDL, cada módulo incluye registros y la mayoría son de lectura y escritura. Los registros deben mantener su estado, por lo que básicamente se declaran como señales en VHDL. Por ejemplo, la declaración de los registros del puerto 1:

```
signal REG_CONF1: STD_LOGIC_VECTOR
    (7 downto 0):="00000000";
signal REG_OUT1: STD_LOGIC_VECTOR
    (7 downto 0):="00000000";
signal REG_IN1: STD_LOGIC_VECTOR
    (7 downto 0):="00000000";
```

La escritura de los registros es síncrona y se debe decodificar la dirección en cada operación para evitar accesos erróneos, además de revisar el estado de la señal de *reset*. Por ejemplo, la escritura del registro de configuración del puerto 1 se realiza con el siguiente proceso en VHDL:

```
writel: process(clk)
begin
    if clk'event and clk = '1' then
        if RST='1' then
            REG_CONF1<="00000000";
```

```
        elsif DIR="11110000" and W='1' then
            REG_CONF1<=INPUT;
        end if;
    end if;
end process writel;
```

Procesos similares se utilizan para la escritura de otros registros.

Para las lecturas, puesto que un módulo incluye diferentes registros, en cada operación de lectura se debe evaluar si es un registro del módulo el que se intenta leer, en caso de no ser así, el bus de salida debe estar en alta impedancia para evitar conflictos con las señales provenientes de otros módulos. Las lecturas no son síncronas porque son solicitadas por el núcleo, como un ejemplo, se muestra un proceso para la lectura de los registros del módulo de puertos:

```
read_process: process(R, DIR,REG_CONFIG1,
    REG_OUT1,REG_IN1)
begin
    if R='1' then
        case DIR is
            when "11110000" =>
                VAL_OUT<=REG_CONF1;
            when "11110001" =>
                VAL_OUT<=REG_OUT1;
            when "11110010" =>
                VAL_OUT<=REG_IN1;
            when others =>
                VAL_OUT <= "ZZZZZZZZ";
        end case;
    else
        VAL_OUT <= "ZZZZZZZZ";
    end if;
end process read_process;
```

Módulo de Puertos

Un aspecto que se planteó en el diseño fue que todos los recursos jerárquicamente se ubicarán en tres módulos dentro de un bloque de recursos. Sin embargo, los puertos no se pueden implementar en un FPGA bajo esta estructura, porque los buffers de tres estados se intentan implementar en los bloques lógicos configurables (CLB, *configurable logic block*) de un FPGA y se obtienen errores de asignaciones dobles a una señal. Los buffers de tres estados deben estar en los bloques de entrada y salida (IOB, *input output block*) y por lo tanto su declaración se debe realizar en el nivel de mayor

jerarquía, al mismo nivel que el núcleo con su memoria.

En la figura 10 se muestra el resultado de la implementación del puerto 1, con sus tres registros y sus terminales de salida.

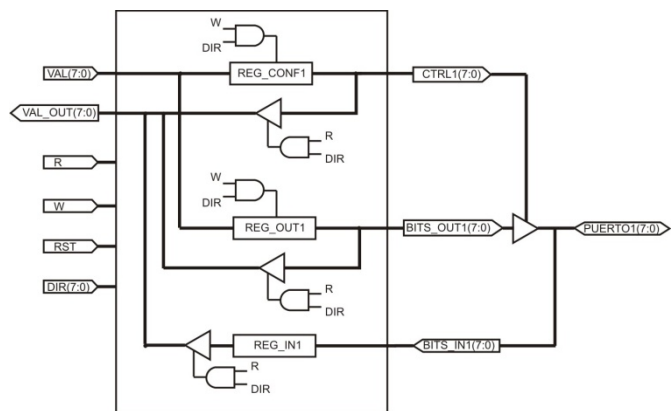


Fig. 10 Implementación del puerto 1

Los buffers de tres estados se implementan con las siguientes instrucciones de VHDL:

```

for i in 0 to 7 GENERATE
begin
    PUERTO1(i) <= BITS_OUT1(i) when
        CTRL1(i)='1' else 'Z';
    BITS_IN1(i) <= PUERTO1(i);
end GENERATE;
    
```

La construcción de los otros puertos es similar, excepto para aquellas terminales que tienen una doble función.

Módulo del Temporizador

Este módulo se integra de tres componentes, el primero de ellos es el pre-escalador, éste se basa en un registro de 25 bits que se incrementa cada ciclo de reloj, un multiplexor para obtener diferentes factores de división y un registro para almacenar la selección, denominado VAL_REG. La figura 11 muestra como se implementó al pre-escalador.

El multiplexor se crea con una estructura *case-when* de VHDL. El registro VAL_REG se inicializa con cero, por lo que después de un *reset* no hay señal de reloj.

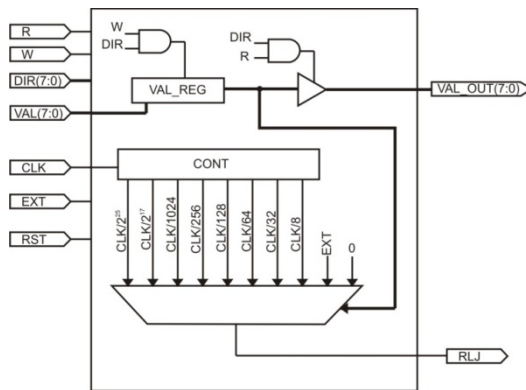


Fig. 11 Implementación del pre-escalador

El segundo componente de este módulo es el contador, el cual básicamente es un registro que se debe modificar con la señal RLJ que proviene del pre-escalador, esto crea un problema porque las escrituras de los registros se manejan en sincronía con la señal de reloj, para resolverlo, cuando ocurre un evento en la señal de reloj, se observa el estado actual y anterior de RLJ, con lo cual se determina si es necesario modificar al contador. El proceso que se encarga de esta evaluación es el siguiente:

```

flanco_subida: process(clk)
begin
    if clk'event and clk='1' then
        delay_RLJ <= RLJ;
        if RLJ='1' and delay_RLJ='0' then
            flanco <= '1';
        else
            flanco <= '0';
        end if;
    end if; end process flanco_subida;
    
```

Puesto que *delay_RLJ* es una señal, su escritura será efectiva cuando el proceso termine, por lo que para la comparación aún conserva su estado anterior. En la figura 12 se muestra como se implementó el contador. Otras señales que afectan al registro CONT_8B son: UP_DOWN para determinar si asciende o desciende y CLR, para su puesta en ceros. Estas señales provienen del componente PWM, el cual es el tercer componente del módulo del temporizador, para las funciones especiales.

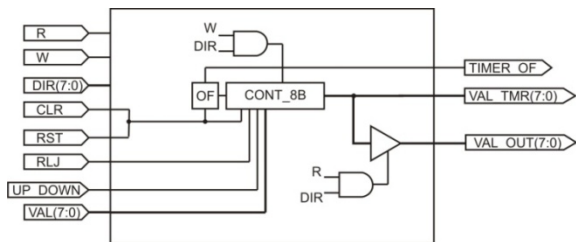


Fig. 12 Implementación del contador

La implementación del componente PWM se muestra en la figura 13. El componente recibe continuamente el valor del contador y la bandera de desbordamiento.

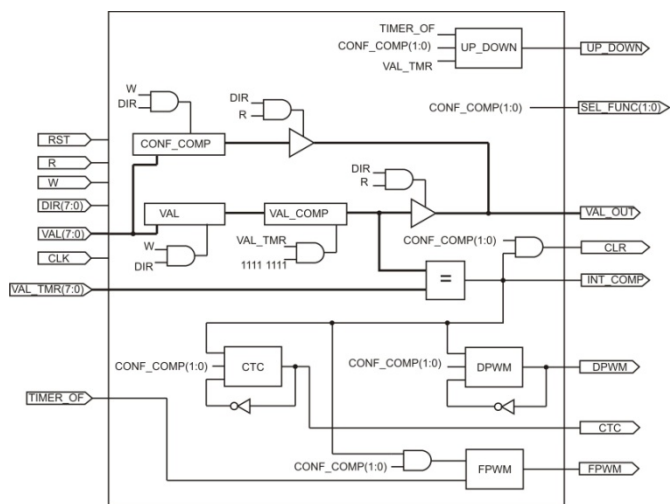


Fig. 13 Componente PWM

A través de procesos se generan las señales PWM con pendiente simple (FPWM) o con pendiente doble (DPWM), si tales funciones están habilitadas en el registro de configuración (CONF_COMP). El registro VAL_COMP contiene el valor con el que se va a comparar al temporizador en cada ciclo de reloj, su valor proviene de un registro temporal denominado VAL que es donde escribe el usuario. La escritura de VAL_COMP se realiza cuando el temporizador alcanza su valor máximo, con lo cual se evitan errores debidos a que el valor de comparación sea menor al valor actual del contador.

Módulo de Interrupciones

El módulo de interrupciones determina la validez de un evento externo, de desbordamiento del temporizador o de igualdad en una comparación, para disparar una interrupción, evaluando los

registros de estado y de configuración. De acuerdo a su diseño, este módulo contiene dos componentes. El componente de Detección incluye la lógica combinacional, con base en los bits del registro de configuración y estado, cuya salida va a activar la entrada INTERRUPT del núcleo.

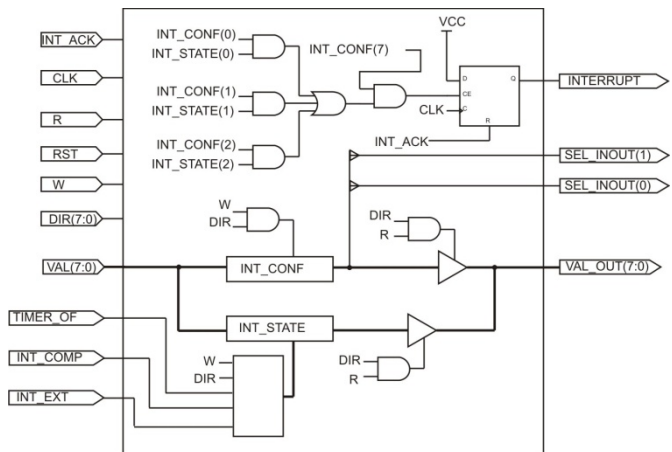


Fig. 14 Componente de Detección

El registro INT_CONF sólo se escribe por software. INT_STATE puede ser escrito por software para forzar un evento o limpiar las banderas; generalmente se escribirá por hardware.

El componente de selección determina la manera en que se activará la interrupción externa, para ello incluye un detector de flanco de subida, flanco de bajada y un inversor, para una activación por un nivel bajo de voltaje. Esto se observa en la figura 15.

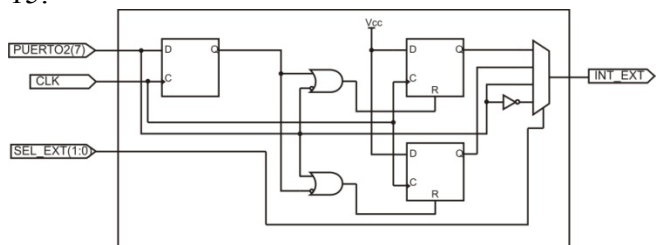


Fig. 15 Componente de Selección

Al integrar todos los elementos, el microcontrolador se puede ver como un único módulo, con sus 4 puertos y algunas terminales con doble función, esto se muestra en la figura 16.

Todos los módulos se desarrollaron en VHDL, las imágenes se dibujaron con base en las generadas

por la herramienta ISE versión 9.2 de Xilinx, en la vista que presenta la lógica de transferencia de registros (RTL).

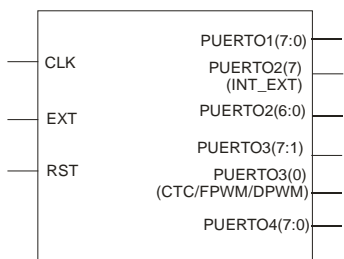


Fig. 16 El sistema completo

V. PRUEBAS Y RESULTADOS

Para evaluar al sistema de la figura 16, se implementó en una tarjeta de desarrollo *Spartan 3E Starter Board*, manufacturada por Digilent, la cual tiene un FPGA XC3S500 y un oscilador con una frecuencia de 50MHz, es decir, un período de 20ns [11].

En la tabla 7 se muestra la cantidad de recursos que ocupa el núcleo antes y después de agregar los módulos de recursos desarrollados.

RECURSOS	NÚCLEO Y MEMORIA	SISTEMA INTEGRADO
Número de IOBs externos	14 de 190 (7.3 %)	34 de 190 (19 %)
Número de bloques RAM	1 de 20 (5 %)	1 de 20 (5 %)
Número de Slices	91 de 4656 (1.9 %)	329 de 4656 (7 %)

Tabla 7. Comparación en el uso de recursos

La tabla 7 demuestra cómo la cantidad de recursos utilizados en el FPGA sigue siendo demasiado pequeña, en relación al total de recursos disponibles en el FPGA, a pesar de haber agregado al bloque de recursos.

Con el microcontrolador empotrado en el FPGA se realizaron diferentes pruebas para la evaluación de los recursos. En el siguiente código se muestra cómo generar una señal de 1 MHz ($T = 1\mu S$) activando la conmutación de una salida y limpieza del temporizador, ante una coincidencia por comparación (constantes en hexadecimal):

```

1:   LOAD          S1, 02
2:   OUTPUT       S1, VAL_REG
3:   LOAD          S1, 0F
4:   OUTPUT       S1, REG_CONF3
5:   LOAD          S1, 01
6:   OUTPUT       S1, CONF_COMP
7:   LOAD          S1, 18
8:   OUTPUT       S1, VAL_COMP
    
```

Con las líneas 1 y 2 se configura el reloj, sin factor de división ($f = 50 \text{ MHz}$, $T = 20 \text{ nS}$), en las líneas 3 y 4 se configura al nibble bajo del puerto 3 como salida y al alto como entrada, en las líneas 5 y 6 se habilita la limpieza del temporizador ante una coincidencia por comparación y en las líneas 7 y 8 se define el valor a comparar que es de 18 (24 en decimal), es decir, un total de 25 ciclos (0 a 24). La frecuencia será:

$$f = \frac{1}{T} = \frac{1}{2 \times 25 \times 20\text{nS}} = \frac{1}{1\mu S} = 1\text{MHz}$$

En la figura 17 se muestra el resultado observado en el osciloscopio.

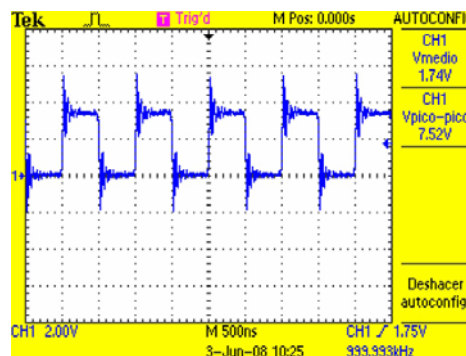


Fig. 17 Prueba del temporizador

En la figura 17 se observa como la frecuencia real coincide con la deseada. Y de acuerdo con el código anterior, es el recurso el que genera la señal, mientras el procesador podría estar haciendo otras tareas.

Otro ejemplo se muestra en el código siguiente, con el que se pretende obtener una señal PWM de rampa única, con una frecuencia de 195 kHz y un ciclo de trabajo del 80 %:

```

1:   LOAD          S1, 0F
2:   OUTPUT       S1, REG_CONF3
    
```

```

3:   LOAD      S1,02
4:   OUTPUT    S1,CONF_COMP
5:   LOAD      S1,CC
6:   OUTPUT    S1,VAL_COMP
7:   LOAD      S1,02
8:   OUTPUT    S1,VAL_REG

```

Las líneas 1 y 2 configuran al puerto 3, las líneas 3 y 4 habilitan el modo de PWM de rampa simple, el temporizador corre de 0 a 255 y reinicia. En las líneas 5 y 6 se carga como valor para comparación el número CC (204 en decimal), por que el 80 % de 256 es 204.8. En las líneas 7 y 8 se define al reloj sin factor de división, por que cuando se genera una señal PWM, la frecuencia sólo se puede modificar con el pre-escalador, con base en la siguiente ecuación:

$$f_{PWM} = \frac{f_{clk}}{256 \cdot n}$$

La variable n hace referencia al factor de pre-escala, para la frecuencia deseada (195 kHz) y el reloj de la tarjeta, se obtiene $n = 1.001$. En la figura 18 se muestran los resultados que se obtuvieron en el osciloscopio.

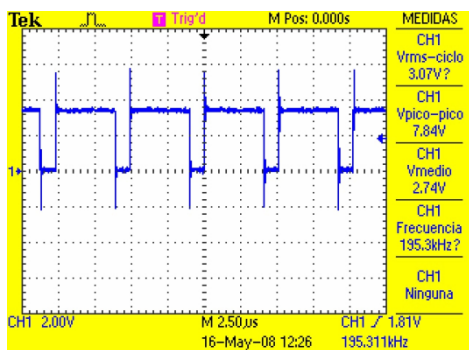


Fig. 18 Generación de PWM

VI. CONCLUSIONES

Se obtuvo un microcontrolador totalmente empotrado en un FPGA, el cual no requiere de conexiones externas y es inmune a ser obsoleto debido a que puede ser reasignado a futuras generaciones de FPGA's.

Es posible agregar otros recursos al núcleo, se tiene una gran flexibilidad, no sólo el módulo del Picoblaze es versátil, también el ambiente lo es.

Se realizaron otras pruebas para los diferentes recursos y en todos los casos los resultados fueron los esperados. Los errores en la generación de señales son debidos al redondeo, ya que para algunas frecuencias se pueden necesitar valores en de comparación en punto flotante, que no pueden ser cargados en los registros, porque sólo manejan números enteros.

VII. REFERENCIAS

- [1] Chapman, Ken: "PicoBlaze KCPSM3 8 bit Microcontroller"; Xilinx, Inc; June, 2005.
- [2] Chapman, Ken: "PicoBlaze 8-bit Embedded Microcontroller User Guide"; Xilinx, Inc; November, 2005.
- [3] Angulo Usategui, José María; Angulo Martínez, Ignacio: "Microcontroladores PIC Diseño práctico de aplicaciones"; McGraw-Hill, 2002
- [4] ROMERO, René: "Sistemas Digitales con VHDL"; Legaria Ediciones
- [5] Xilinx: Prototyping and Development Boards; Spartan-3e Starter Kit Design Examples, http://www.xilinx.com/products/boards/s3e_starter/reference_designs.htm
- [6] "Sistemas embebidos. Implementación en un FPGA del microcontrolador PicoBlaze."; Universidad Autónoma de Madrid, España; <http://arantxa.ii.uam.es/~edcd/lab/Practica_2.pdf>
- [7] JAMES, Lucero; "Reference System: PowerPC 440 System Simulation"; Xilinx, Inc; April, 2008; <http://www.xilinx.com/support/documentation/application_notes/xapp1003.pdf>;
- [8] SCHOEBERL, Martin; "Java Optimized Processor" February, 2008; <<http://www.jopdesign.com/index.jsp>>;
- [9] Mackenzie, I. Scout: "The 8051 Microcontroller"; Edit. Thomson Delmar Learning; Second Edition.
- [10] ATMEL, AVR: "ATmega8 DataSheet", 2002.
- [11] Spartan-3E Starter Kit Board User Guide, Xilinx, Inc., version 1.0a, March 9, 2006

MÓDULOS ARITMÉTICOS CMOS VLSI PARA SISTEMAS MULTIMEDIA

Mariano Aguirre Hernández, Mónico Linares Aranda

Instituto Nacional de Astrofísica, Óptica y Electrónica, Departamento de Electrónica
Luís Enrique Erro No. 1, Sta. María Tonantzintla, Apdo. postal 51, CP. 72000, Puebla, México
mariano.aguirre@intel.com , mlinares@inaoep.mx

RESUMEN

En este trabajo se presenta un conjunto de módulos aritméticos de alta velocidad y bajo consumo de potencia altamente atractivos para aplicaciones multimedia. En su diseño se utiliza una nueva estructura lógica interna y una combinación de estilos lógicos de diseño estáticos y dinámicos que permiten la reducción de potencia y la obtención de barridos de voltaje completos en sus nodos, aptos para aplicaciones digitales. Simulaciones SPICE muestran que los módulos son capaces de operar a 1.2GHz usando una alimentación de 3.3V con ahorros de potencia de hasta un 20%. Resultados experimentales de un multiplicador de 8x8 bits fabricado con una tecnología CMOS de 0.35 micrómetros confirman los resultados obtenidos mediante simulación.

Palabras clave: circuitos digitales, circuitos integrados, sistemas VLSI.

I. INTRODUCCIÓN

Actualmente, debido a la gran cantidad de información (datos) que se necesita procesar y a los requerimientos de portabilidad, los sistemas modernos tales como los sistemas multimedia, los microprocesadores de alto desempeño y los procesadores digitales de señales (DSP), son diseñados con desafiantes restricciones en términos de velocidad, consumo de potencia y ruido.

Si bien es posible aumentar la frecuencia de operación (velocidad) de un circuito mediante el uso de procesos tecnológicos avanzados de circuitos integrados (diseño con las dimensiones más pequeñas permitidas), sin embargo, este

aumento de velocidad así como el aumento del número de dispositivos por unidad de área y por ende el aumento de tamaño de los circuitos integrados, hacen que la disipación de potencia total sea hoy en día una de las restricciones de igual o mayor importancia incluso sobre el gasto de área, en la construcción de procesadores modernos. En este sentido, técnicas de “*clock-gating*”, deben ser aplicadas en todo diseño con el fin de optimizar el uso de las fuentes que les proporcionan energía; esto es principalmente importante si de equipo portátil se trata. *Clock-gating* es una técnica que reduce la disipación de potencia dinámica mediante la detención selectiva (deshabilitación) de la señal de reloj en secciones del circuito que están inactivas durante ciertos periodos de tiempo, y puede aplicarse en diferentes niveles jerárquicos. Por lo tanto, bajo este escenario común viene a ser extremadamente importante considerar el consumo de potencia de cualquier modulo propuesto cuando no hay transiciones (modo de stand-by) de los datos de entrada o actividad de la señal de reloj.

Con el fin de construir sistemas multimedia eficientes, en este artículo se presenta el diseño de diversos módulos aritméticos: compuertas XOR/XNOR, sumadores, flip-flops, y multiplicadores. El artículo es organizado como sigue: en la sección II se presentan los módulos aritméticos diseñados, sus resultados de simulación HSPICE y una comparación de los mismos. En la sección III se presenta el circuito fabricado, así como los resultados experimentales obtenidos. Finalmente en la sección IV se dan las conclusiones del presente trabajo.

II. MÓDULOS ARITMÉTICOS

A. XOR-XNOR

Las compuertas XOR y XNOR constituyen el componente básico de diferentes circuitos utilizados para realizar operaciones aritméticas tales como sumadores [1, 2] y comparadores [3], [4], entre otros. El desempeño individual de estas compuertas afecta el desempeño total de módulos mayores (flip-flops, sumadores, multiplicadores, etc.) por lo que es deseable que estas compuertas sean óptimas en términos de velocidad, consumo de potencia y área.

En la figura 3 se muestran varias de las estructuras XOR-XNOR publicadas [5-10]. Estas compuertas son diseñadas con estilos lógicos de compuertas de transmisión TG (b, c, g, h, i), de paso simple CPL, de paso dual DPL (*d, e, f* y *g*) y otros esquemas recientemente publicados *k* [10], *l* [6] y *m* [9]. El objetivo común de estos trabajos es obtener una estructura XOR-XNOR de bajo consumo de potencia considerando cada componente de la misma, esto es, dinámica, estática y de fuga. En [11] se presentó una comparación de desempeño (Tabla I) de estructuras XOR-XNOR con especial atención en esquemas con barrido de voltaje completo en sus nodos de salida compatibles con aplicaciones digitales. A partir de la Tabla I se observa que las compuertas XOR-XNOR con menor disipación de potencia son aquellas que tienen un barrido de voltaje incompleto (*d, h, i* y *j*); sin embargo considerando el producto potencia-retardo las mejores son: (*b*), (*d*), (*e*) y (*g*). Algunos de estos últimos diseños serán utilizados en la construcción de flip-flops y sumadores en las siguientes secciones del presente trabajo.

B. FLIP-FLOPS

Los “flip-flops”, referidos en este trabajo como elementos de sincronización, son bastante utilizados (hasta un 20% del total de elementos en un microprocesador) en sistemas temporizados como lo son los sistemas “pipeline” los cuales son bastante atractivos por las altas razones de procesamiento de datos que pueden lograrse, y

tienen un gran impacto en el desempeño (frecuencia de operación y consumo de potencia) de sistemas síncronos modernos [12]. Debido a ello, se ha buscado prioritariamente reducir su consumo de potencia dinámica. Así, la propuesta de [13] presentada en la Fig. 2, ha mostrado una gran reducción de potencia al aplicar la técnica “clock-gating” en forma individual a cada flip-flop. De acuerdo con [13] y la fig. 3, el uso de transistores de paso para el comparador (compuerta XOR) permite reducir el consumo de potencia debido al reducido barrido de voltaje en el nodo *comp*, mientras que utilizando MOS estándar interconstruido se logra mayor frecuencia de operación. Con el fin de mejorar el desempeño tanto en velocidad como en consumo de potencia proponemos un comparador construido en base a la compuerta XOR diseñada con lógica de transistor de paso y técnica “bootstrapped” como se muestra en la Fig. 4. Dado que la etapa *invFB* no es estrictamente necesaria ya que el nodo \overline{ckg} es recargado (a través del transistor PMOS) en cada semiciclo bajo de la señal de reloj, se obtiene una propuesta de circuito (Fig. 5) de mayor velocidad y mucha menor disipación de potencia. Estos esquemas de clock-gating fueron utilizados individualmente a un flip-flop maestro-esclavo [13], para obtener cinco flip-flops disparados por borde negativo. Los resultados de simulación HSPICE de velocidad y consumo de potencia para cada flip-flop se muestran en la Tabla II. A partir de esta tabla se puede ver que el esquema *bootstrap_no_fb* presenta menor consumo de potencia y mejora la velocidad del flip-flop. Esta reducción de consumo de potencia es mayor que para el esquema *pass_no_fb*. Debido al barrido de voltaje reducido en el nodo *comp*, obviamente su velocidad es reducida comparada al esquema *bootstrap_no_fb*, no obstante aún es más rápida que muchos de los otros esquemas.

C. SUMADORES

Varios artículos han sido publicados sobre el diseño de circuitos sumadores de bajo consumo de potencia considerando tanto la estructura interna como el estilo lógico utilizados para construir el módulo sumador.

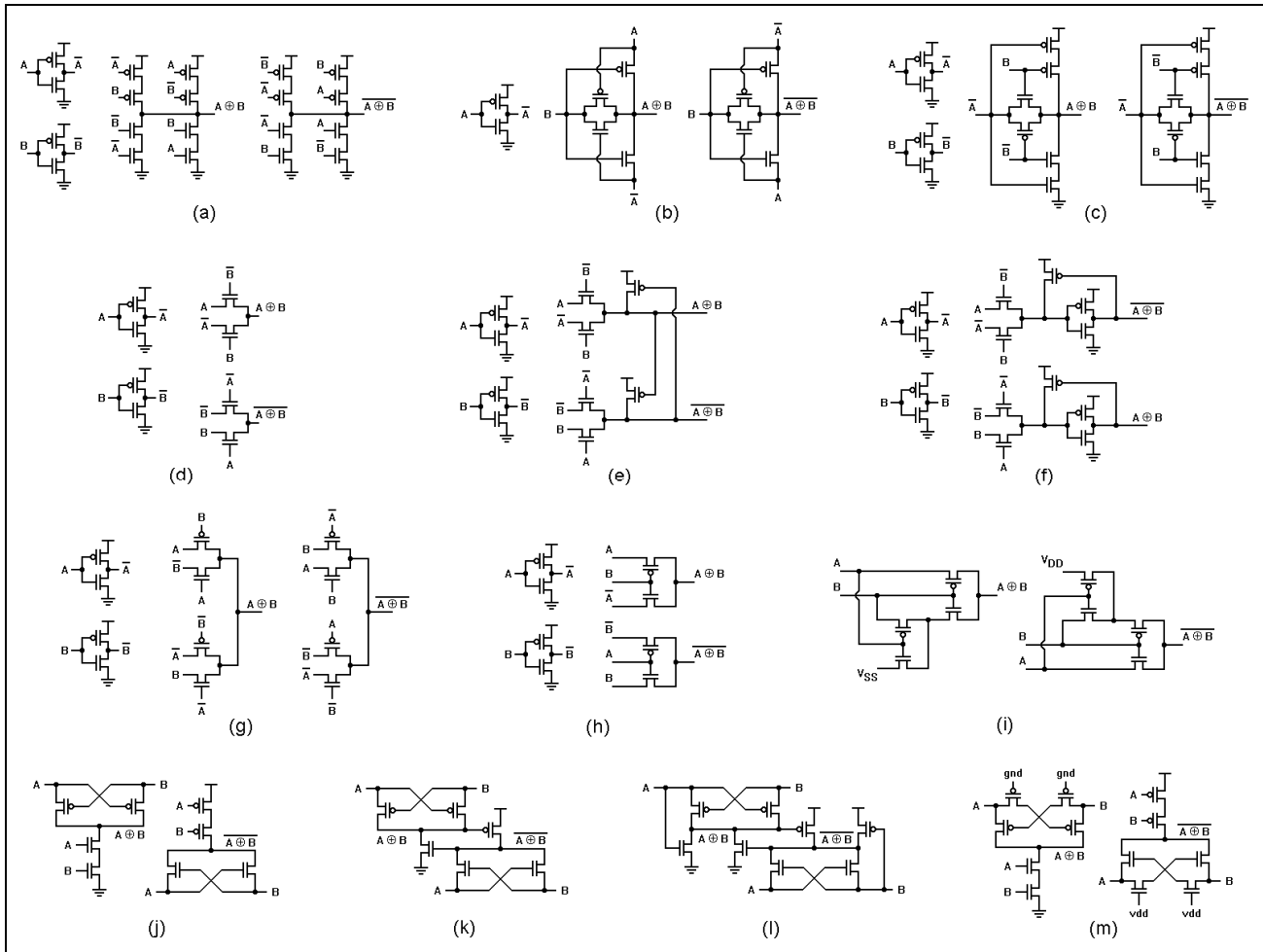


Fig. 1 Diseños previos de compuertas XOR-XNOR

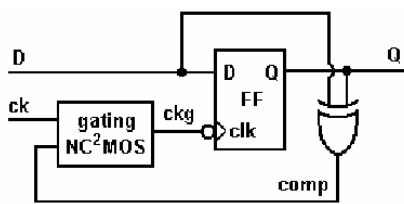


Fig. 2 Técnica de "Clock-gating aplicada a un Flip-flop"

Tabla 2. Características de retardo y consumo de potencia de circuitos "clock-gating"

Esquema	Retardo (ps)	Potencia (μ W)			
		250MHz	125MHz	62MHz	0MHz
Built in gating	424.5	2185.4	1287.7	946.8	518.3
Pass gating	451.3	2029.3	1191.6	872.1	468.9
Bootstrap gating	421.1	2168.1	1275.3	936.3	508.6
Bootstrap no_fb	390.4	2049.2	1220.1	904.6	507.2
Pass_no_fb	410.0	1833.2	1180.6	889.2	503.2

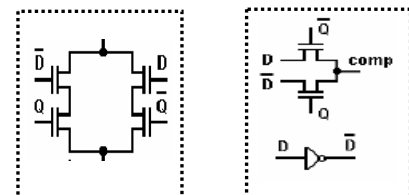
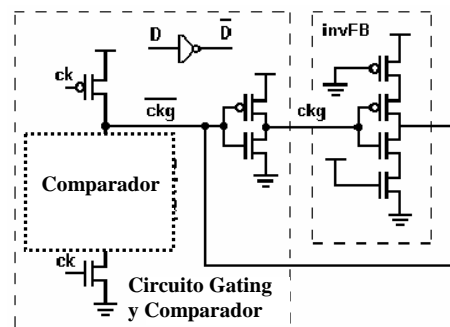


Fig. 3 Circuitería "Clock gating N²CMOS": a) comparador inter-construido NMOS (*built_in_gating*), b) comparador de transistores de paso (*pass_gating*).

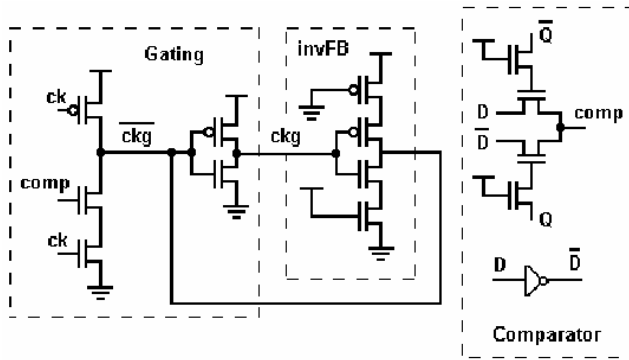


Fig. 4 (Arriba) Circuitería “Clock gating N²CMOS” con comparador en base a transistor de paso “bootstrapped” (bootstrapped_gating).

Fig. 5 (Derecha) Circuitería “Clock gating N²CMOS” sin inversores de retroalimentación y a) comparador en base a transistor de paso “bootstrapped” (bootstrap_no_fb), b) comparador en base a transistor de paso (pass_no_fb).

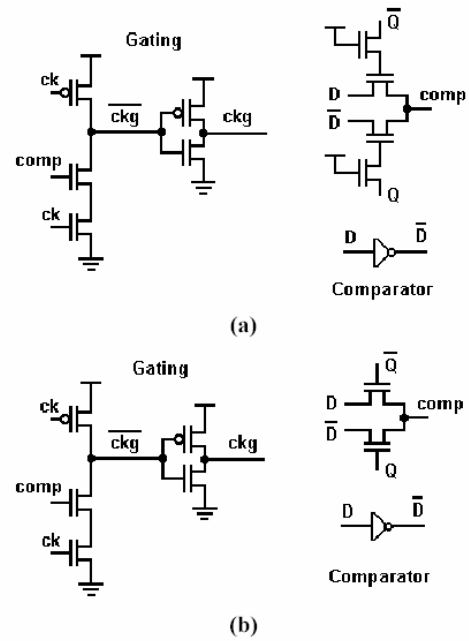


Tabla 1 Resultados de simulación de compuertas XOR-XNOR (Disipación de potencia en μW , Retardo en ns)

Esquema	Potencia Promedio	Pot de fuente	Potencia Dinámica	Potencia Estática	Potencia de Corto-circuito	% Pot. Gastada /Total	% de Potencia Xor/top	Retardo	Pot*Ret
a	Top	849.4	849.4	731.0	-	118.5	13.9	0.432	366.9
	Xor	320.8	320.8	291.4	-	29.0	9.1		
b	Top	573.9	573.9	458.7	-	115.2	20.0	0.281	161.3
	Xor	85.0	54.5	166.7	-	14.5	8.1		
c	Top	693.3	693.3	577.5	-	115.8	16.7	0.478	331.4
	Xor	239.1	238.9	218.5	-	20.5	8.6		
d	Top	739.9	739.9	418.1	17.2	304.3	43.4	0.219	162.0
	Xor	85.5	73.3	106.9	-	7.6	6.7		
e	Top	583.1	583.1	444.2	-	138.6	23.8	0.226	131.8
	Xor	138.1	133.3	137.9	-	37.3	21.3		
f	Top	716.8	716.8	535.6	-	181.2	25.3	0.414	296.8
	Xor	268.5	261.0	200.6	-	104.3	34.2		
g	Top	586.7	586.7	479.5	-	107.3	18.3	0.236	138.5
	Xor	131.5	106.3	196.4	-	25.7	11.6		
h	Top	771.2	771.2	440.6	16.8	313.5	42.8	0.247	190.5
	Xor	90.5	78.5	152.8	-	17.8	10.4		
i	Top	804.2	804.2	418.1	13.0	372.9	47.9	0.368	295.9
	Xor	52.4	27.7	129.0	-	5.0	3.6		
j	Top	689.4	689.4	474.2	5.0	215.2	31.2	0.300	206.8
	Xor	71.3	41.9	182.5	-	7.6	4.0		
k	Top	1008.5	1008.5	523.7	-	484.8	48.0	0.527	531.5
	Xor	334.2	258.1	214.2	-	328.4	60.5		
l	Top	1049.7	1049.7	484.4	112.9	451.8	53.8	0.397	416.7
	Xor	442.7	196.7	184.1	112.9	315.2	69.9		
m	Top	758.3	758.3	489.4	-	268.6	35.4	0.491	372.3
	Xor	98.3	52.5	255.1	-	11.2	4.3		

C. SUMADORES

Varios artículos se han publicado sobre el diseño de circuitos sumadores de bajo consumo de potencia considerando tanto la estructura interna como el estilo lógico utilizados para construir el módulo sumador.

C.1 SUMADORES ESTÁTICOS

Desde la realización CMOS estándar, varios sumadores completos han sido construidos. Así, considerando el estilo lógico de diseño, tenemos: Differential Cascode Voltage Switch (DCVS) [14], Complementary Pass- Transistor Logic (CPL) [15], Double Pass-Transistor Logic (DPL) [16], y Swing Restored CPL (SR-CPL) [17]. Sobre la base de estos estilos lógicos otros autores han construido nuevos sumadores cambiando la estructura lógica interna del módulo, tal como en [18] en donde con su propuesta se reduce el retardo de propagación total gracias a la reducción de capacitancias en nodos críticos. Asimismo, el retardo en las terminales de *suma* y *acarreo* pueden ser entonados individualmente ajustando compuertas XOR-XNOR como las presentadas en la sección A de este trabajo.

Considerando la estructura interna propuesta en [18] y presentada en la Fig. 6, proponemos las estructuras de sumador mostradas en la Fig. 7. Estas nuevas celdas sumadoras son construidas sobre la base de funciones Booleanas XOR/XNOR y AND/OR, para obtener las salidas de SUM y CARRY, respectivamente [5]. Nuestra primer propuesta (Figura 7a) es diseñada con el estilo lógico DPL (Ours_1), y la segunda propuesta (Figura 7b) usa compuertas XOR/XNOR diseñadas con el estilo lógico SR-CPL (Ours_2).

Las compuertas AND/OR y un multiplexor han sido construidas utilizando una configuración de transistor de paso. El multiplexor es utilizado para obtener la salida C_o . Los latches de salida son usados con el fin de poder comparar su desempeño con sumadores del tipo dinámico que se verán más adelante.

En la tabla III muestran los resultados de simulación de los sumadores propuestos y muchos otros sumadores propuestos por otros autores,

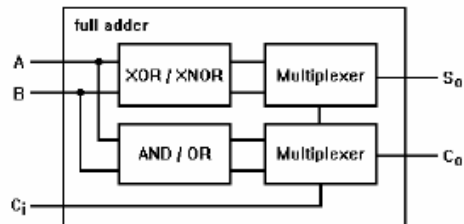


Fig. 6 Nueva estructura lógica interna del sumador

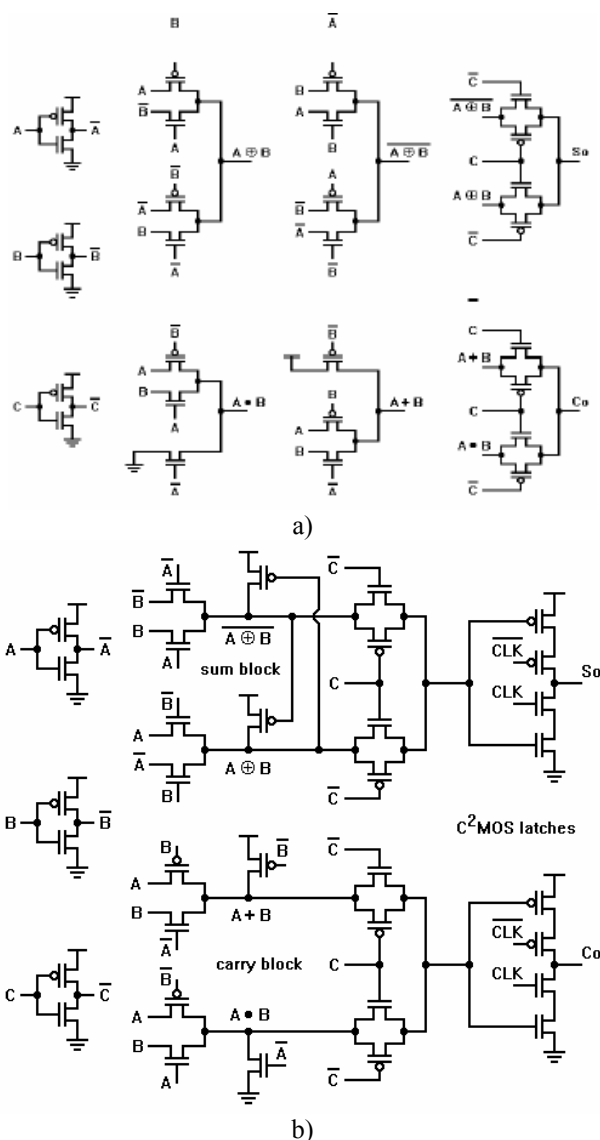


Fig. 7 Celdas sumadoras a) DPL, b) SR-CPL

respecto de disipación de potencia y retardo. Los sumadores de comparación son: cmos_26 [19], cmos_28, cpl [15], sr_cpl [17], dcvs [14], bay_10a [20], bay_10b [21], bay_14a [22], bay_14b [23], bay_16 [24], full_rest [25], mux_based [26], tran_funct [27], wey_chow [28], wu_ng [29].

Tabla 3 Resultados de simulación de sumadores (Disipación de potencia en μW , Retardo en ns , ancho en μm , V_{dd} en Voltios)

Esquema		Pot Prom	Pot de fuente	Pot. Dinámica	Pot. Estática	Pot. Corto-circuito	% Pot. Gastada /Total	% Sumador /top	Retardo	Pot*Ret	Σ Ancho	Vdd mín
Cmos26	Top	1286.4	1286.3	875.2	4.3	406.9	31.9	68.6	0.703	904.3	67.8	1.3
	Add	882.2	811.5	683.8	4.3	363.0	34.9					
Cmos28	top	1737.0	1736.8	1420.7	-	315.8	18.2	59.0	0.984	1709.2	184.8	1.3
	add	1025.0	1025.0	746.8	-	277.9	27.1					
Cpl	top	2976.1	2975.9	984.4	1991.6	37.6	66.9	86.6	0.781	2324.3	113.2	1.8
	add	2577.0	2504.4	702.6	37.6	1910.0	73.4					
Cpl_sr	top	2264.2	2264.1	1097.9	-	1165.9	51.5	81.7	0.812	1835.5	116.4	1.4
	add	1848.8	1804.8	810.2	-	1227.6	58.1					
Cpl_uye	top	2179.9	2179.7	1190.3	-	989.3	45.3	80.3	0.853	1859.5	116.4	1.4
	add	1750.0	1688.6	982.7	-	963.3	49.5					
Devs	top	2965.5	2965.4	1579.7	-	1385.3	46.7	84.8	1.107	3282.8	182.4	1.3
	add	2515.6	2515.6	1179.1	-	1336.2	53.1					
Bay_10a	top	1576.4	1576.4	632.0	55.1	889.4	60.0	80.0	1.955	3081.9	51.4	2.8
	add	1261.1	1237.5	436.9	55.1	829.6	67.0					
Bay_10b	top	1565.5	1565.5	960.6	2.0	602.6	38.6	71.5	1.157	1811.3	84.8	2.4
	add	1119.0	953.7	773.2	2.0	561.0	42.1					
Bay_14a	top	1221.0	1221.0	848.8	0.7	371.6	30.5	65.5	1.220	1489.6	68.7	2.4
	add	799.6	684.8	658.0	0.7	331.0	33.5					
Bay_14b	top	1290.6	1290.6	975.5	0.7	314.2	24.4	64.3	1.366	1763.0	84.0	2.4
	add	829.3	663.3	785.7	0.7	275.2	26.0					
Bay16	top	1343.8	1343.8	919.4	0.3	423.7	31.5	60.8	1.688	2268.3	80.4	2.8
	add	817.0	699.6	724.4	0.3	393.4	35.2					
Full_Rest	top	1795.5	1795.5	894.3	266.3	634.6	50.1	70.2	1.022	1835.0	72.7	1.8
	add	1260.3	1027.0	687.7	266.3	615.1	56.2					
Mux_Based	top	1560.2	1560.2	750.8	4.6	804.5	51.8	74.0	1.362	2125.0	65.6	2.4
	add	1155.3	1100.2	566.6	4.6	758.0	57.3					
Tran_Funct	top	1225.0	1225.0	796.6	-	428.0	34.9	71.2	0.932	1141.7	63.0	1.3
	add	871.6	808.0	612.2	-	379.8	38.3					
Wey_Chow	top	1346.4	1346.4	942.2	-	404.3	30.0	64.7	1.024	1378.7	75.8	1.7
	add	871.1	739.9	747.1	-	372.2	33.2					
Wu_ng	top	1161.6	1161.6	959.6	-	202.0	17.4	61.9	1.067	1239.4	79.4	1.7
	add	719.1	728.6	741.8	-	168.6	18.5					
Ours_1	top	843.8	843.8	751.1	-	92.7	11.0	46.8	0.716	604.2	52.8	1.5
	add	395.2	280.2	510.8	-	56.4	10.0					
Ours_2	top	835.6	835.6	710.8	-	124.7	15.0	50.6	0.734	613.3	50.4	1.5
	add	422.6	364.7	466.6	-	89.8	16.1					

Las simulaciones fueron realizadas utilizando Hspice [35] para medir el retardo de propagación en las terminales de salida *suma* y *acarreo* y Nanosim [36] para determinar las características de consumo de potencia. De la Tabla III se deriva lo siguiente:

- Con respecto a nuestras propuestas presentadas en este artículo, claramente puede verse la ventaja del uso de la estructura lógica interna [18] y presentada en la figura 6. Las realizaciones diseñadas sobre esta estructura (*Ours_1* y *Ours_2*) presentan el consumo de potencia, retardo (excepto por la realización *cmos_26*) y producto potencia-retardo más bajos.
- Por lo anterior, y ya que estas realizaciones no

presentan disipación de potencia estática, nuestras propuestas son muy atractivas para aplicaciones portátiles en donde se requieren de módulos de reducido consumo de potencia en modos de standby. En la figura 8 se muestran formas de onda del sumador propuesto *Ours_1*, para un patrón de entrada correspondiente a la tabla de verdad indicada sobre la misma figura.

C.2. SUMADORES DINÁMICOS

El diseño de multiplicadores para sistemas de altas razones de procesamiento (*throughput*) hacen uso de sumadores utilizando estilos lógicos dinámicos por su alta velocidad [30-32], por lo que resulta importante presentar algunos de ellos.

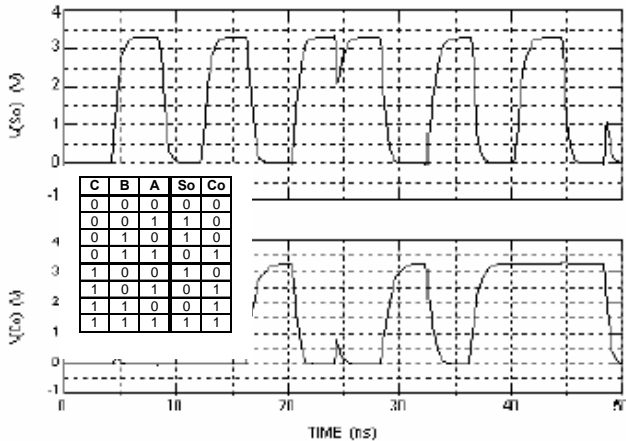


Fig. 8 Formas de onda de los términos Suma (S_o) y Acarreo (C_o) del sumador *Ours_1*

Así, en la Fig. 9 se muestra el sumador usado en [32], el cual es diseñado sobre un estilo lógico *quasidomino*. Este sumador si bien permite una operación rápida lo hace a costa de consumo de potencia. Este mismo sumador ligeramente modificado (el bloque de acarreo es construido con lógica estática) también presentado en [32] y mostrado en la figura 10 (denominado *estatic & dynamic*) permite acelerar el término de acarreo y por tanto de la suma total. Sin embargo, su gran desventaja es la alta disipación de potencia que podría ocurrir si el árbol lógico N (bloque de Acarreo) es encendido simultáneamente en ambas fases de precarga y evaluación.

En [33] se establecen otras desventajas respecto de velocidad para las anteriores implementaciones, por lo que se propone la estructura de la Fig. 11 (*static & pass transistor*) en la cual el término de suma ya no depende de la evaluación del término de acarreo ya que se utilizan etapas separadas para calcular ambos términos. El término de SUMA se obtiene en base al uso de compuertas XOR/XNOR construidas con el estilo lógico de transistor de paso. Desafortunadamente este circuito aún puede disipar de potencia estática debida a la etapa pseudo-nmos utilizada para computar el término de acarreo.

Resumiendo, todas las mejoras hechas en los anteriores sumadores tratando de incrementar su velocidad, introducen nuevos componentes de disipación de potencia o empeoran los ya existentes. La justificación que se da a la utilidad

de estas propuestas es que al operar estos sumadores a alta frecuencia la disipación de potencia es despreciable. Sin embargo, este escenario ha cambiado actualmente ya que incluso el consumo de potencia estático es el que determina en muchos de los casos los actuales diseños VLSI para sistemas portátiles de alto desempeño.

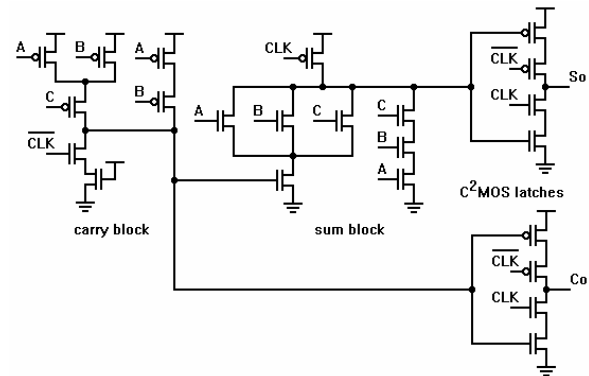


Fig. 9 Sumador completo *Quasi-domino*

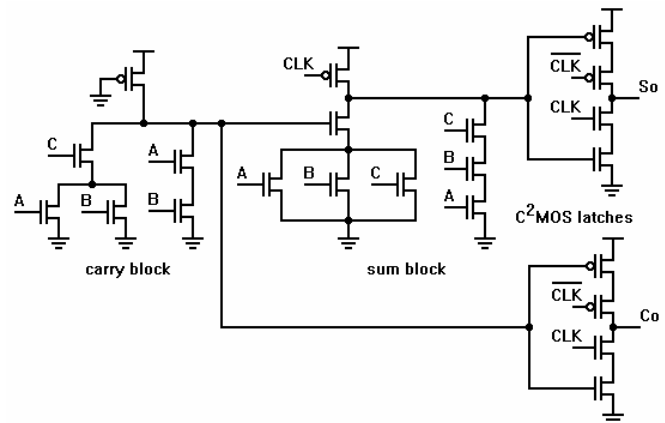


Fig.10 Sumador completo *Static&dynamic*

D. MULTIPLICADORES

La multiplicación es la operación fundamental en muchos algoritmos de procesamiento, y los circuitos que la hacen posible (multiplicadores) son la base del diseño VLSI de microprocesadores de alta velocidad. De hecho, el 8.72% de todas las instrucciones en un programa científico típico lo son multiplicaciones. Por lo tanto, contar con multiplicadores de alta velocidad es clave para lograr un buen desempeño de los procesadores. Bastante trabajo se ha realizado sobre multiplicadores en diferentes niveles de

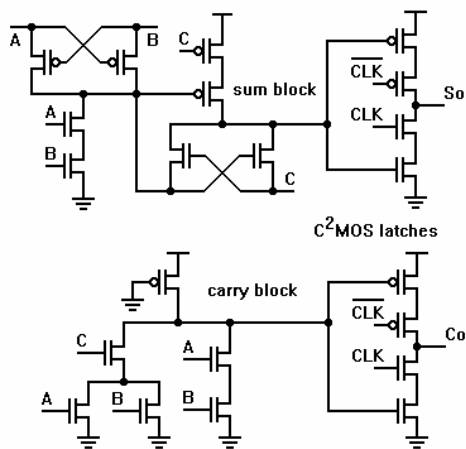


Fig. 11 Sumador completo *Static&pass_transistor*

abstracción: tecnología, patrón geométrico (layout), circuito y arquitectura. La idea básica de todos los diseños de multiplicadores modernos es la implementación rápida de la suma de productos parciales. Los multiplicadores más rápidos son los multiplicadores paralelos, sin embargo, sufren de regularidad; así que los multiplicadores “pipeline” son la mejor selección [34] cuando los parámetros principales son: regularidad, alto desempeño y bajo consumo de potencia. Los multiplicadores pipeline son útiles en sistemas donde el cómputo aritmético es más importante que el retardo; sin embargo, este aumento de procesamiento es obtenido a costa del uso extensivo de registros (flip-flops), incrementándose así el área que ocupa el multiplicador. Esta es una de las razones por la que es importante que los flip-flops no solo sean efectivos en velocidad y consumo de potencia sino también en el área que ocupan.

En todo multiplicador la velocidad queda determinada principalmente por la arquitectura y los circuitos que la conforman, por lo que la velocidad de un multiplicador puede expresarse en términos del número de retardos de celdas a lo largo de la trayectoria crítica. Ya que el retardo de celda, normalmente es determinado por el retardo de un sumador, los cuatro multiplicadores que aquí se presentan se basan en el multiplicador pipeline cuyo diagrama a bloques se muestra en la Fig. 12, aplicando las cuatro celdas sumadoras: *quasi-domino* (Fig. 9), *static&dynamic* (Fig. 10), *static&pass-transistor* (Fig. 11) y *our proposal* (Fig. 7b) presentada en la sección C.1).

Los diferentes multiplicadores 2-complemento fueron diseñados usando parámetros eléctricos de una tecnología AMS 0.35 μm , y simulados utilizando el modelo *BSIM3v3* (level 49) de HSpice [35] y Nanosim [36] alimentados con 3.3 V. Las simulaciones Hspice se realizaron para diferentes escenarios: a) Con transición de datos en las entradas y señales de reloj activadas (*CLK* y \overline{CLK}), para simular una operación normal. b) Con datos estables a las entradas y señales de reloj activadas para simular la técnica “*aislamiento de módulo*” aplicada al multiplicador y circuitería periférica, y c) Con datos estables a las entradas y señales de reloj detenidas para simular la técnica *deshabilitación de reloj* aplicada al multiplicador.

En la tabla IV se presentan los resultados de simulación obtenidos. Es importante mencionar que el multiplicador *our proposal* usando el sumador con lógica SR-CPL (Fig. 7b) fue capaz de operar hasta 1.2 GHz disipando una potencia de 164.2 mW, mientras que la máxima frecuencia de operación de los otros multiplicadores fue de apenas 1GHz con mayores consumos de potencia.

De acuerdo con la tabla IV se tiene lo siguiente:

- Bajo condiciones normales de operación (datos de entrada cambiantes y señal de reloj habilitada), el multiplicador diseñado con el estilo lógico *quasi-domino* es el que presenta mayor consumo de potencia debido a que el circuito tiene dos etapas dinámicas las cuales son precargadas en cada ciclo de reloj.
- El multiplicador diseñado con el estilo lógico *static & dynamic*, reduce la disipación de potencia dinámica mediante la eliminación de la fase de precarga en una de las etapas, pero aún presenta la desventaja de disipación de cortocircuito y el incremento de la disipación de potencia estática debido a la etapa pseudo-NMOS.
- El multiplicador con lógica *static & pass-transistor* muestra menos disipación de potencia debido a que no hay etapas dinámicas, sin embargo, aún existe potencia estática debido al barrido de voltaje incompleto de los latches C²MOS así como a la etapa pseudo NMOS.
- El multiplicador *Our proposal* reduce en 10% la disipación de potencia respecto de *static & pass-transistor*, y más del 20% respecto de las demás, debido a que no hay ni disipación estática

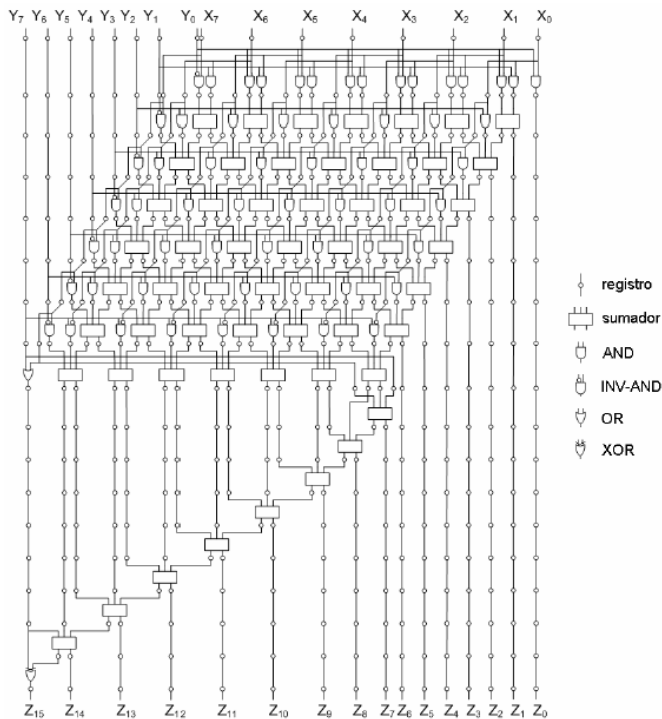


Fig. 12 Diagrama de bloques del multiplicador pipeline

Tabla 4 Resultados de simulación de multiplicadores

Sumador completo	Consumo de potencia @ 1 GHz (mW)		
	operación normal	Entradas estables	Reloj deshabilitado
quasi-domino	190.5	31.7	112.6
static & dynamic	182.6	53.5	120.3
static & pass-tx	161.9	33.4	51.7
Our proposal	144.4	1.5	9.5

ni consumo de cortocircuito, excepto por la disipación en los inversores de entrada.

- En el caso de tener datos de entrada estables y señales de reloj corriendo libremente, el consumo de potencia es reducido para todos los multiplicadores, debido a que no hay componente de disipación dinámica en el arreglo multiplicador, excepto por los buffers de reloj. Asimismo, debido a que esta propuesta no tiene etapas pseudo-NMOS, el consumo de potencia es muy pequeña, con ahorros del 95% respecto de los otros multiplicadores.

- Para el caso de señales de reloj desactivadas, *our proposal* exhibe un ahorro del 80% respecto del multiplicador *static & pass*, y del 90% respecto de los otros multiplicadores.

III. RESULTADOS EXPERIMENTALES

Con el fin de verificar los resultados obtenidos en simulación del multiplicador pipeline *our proposal*, éste fue fabricado utilizando el proceso de fabricación de 0.35 μm CMOS de Austria MicroSystem, a través del servicio EURO PRACTICE. Una fotografía del circuito integrado (chip) fabricado conteniendo el multiplicador se muestra en la Fig. 13. El tamaño del circuito integrado es de 0.6 mm^2 .

El circuito integrado contiene los siguientes bloques (ver Fig. 14):

- Arreglo multiplicador 8x8 bits 2-complemento (1). Ya tratado.
- Registro de corrimiento con entrada paralela (2). Este circuito convierte los 16 bits de salida del multiplicador en una cadena serial de 20 bits (se incluyeron 4 ceros como “banda de protección” para evitar el traslape de los bits del resultado), esto fue con el fin de reducir la cantidad de terminales de entra/salida (E/S). Una vez que el registro de corrimiento se carga con los 16 bits en paralelo del resultado del multiplicador, con un pulso en su entrada “LOAD” se recorre su contenido hacia la izquierda incorporando los bits que abandonan el extremo respectivo en la entrada del registro del extremo derecho. Así, el resultado se mantiene almacenado y circulando a través del registro para ser observado en forma serial en una de las terminales E/S.
- Oscilador controlado por voltaje VCO (3). Este circuito genera una señal de reloj en el rango de 200 MHz a 1.6 GHz para voltajes de control en el rango de 0.8 V a 1.6 V, suficientes para sincronizar la operación del arreglo multiplicador en su frecuencia máxima de 1 GHz. [33].
- Red de distribución de reloj (4). Utilizada para distribuir la señal de temporización y sincronización de los circuitos a partir de la señal generada por el VCO interno o bien de un generador externo.
- Circuito divisor de señal de reloj entre 16 (5). Este circuito permite reducir la frecuencia de oscilación del VCO para que pueda ser medida en una terminal (E/S) del circuito integrado, y cuyas características de retardo permiten manejar señales con una frecuencia máxima de 250 MHz.

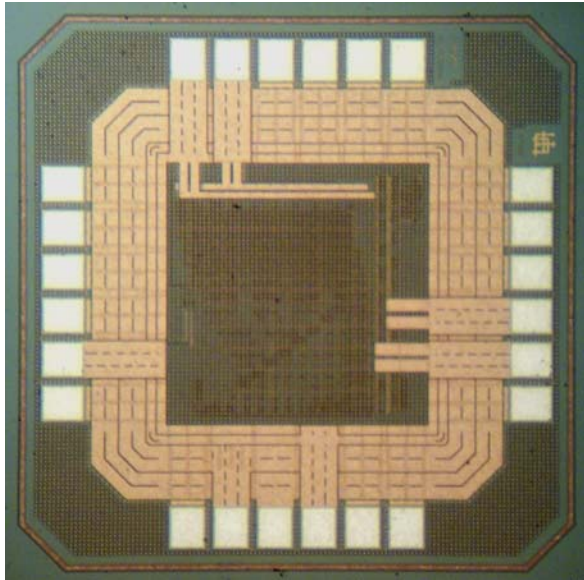


Fig. 13 Fotografía del circuito integrado fabricado conteniendo el multiplicador pipeline

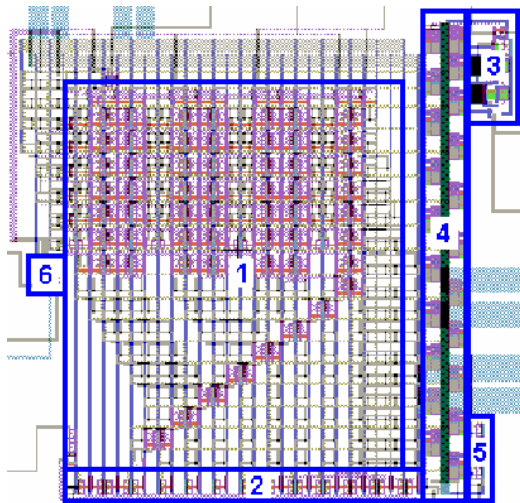


Fig. 14 Patrón geométrico del multiplicador de 8x8 bits

- Multiplexor. Debido a la limitación de respuesta máxima en frecuencia de las terminales E/S, no es posible obtener el resultado de multiplicación en forma serial a través de una terminal E/S a la frecuencia máxima de operación del VCO (1.2GHz). Así que un multiplexor fue usado para seleccionar entre la señal de reloj del VCO y una señal de reloj externa de baja frecuencia una vez que el resultado de multiplicación ha sido capturado en el registro de corrimiento.
- Celda sumadora de 1-bit (6). Por último, con el fin de poder medir el consumo de potencia de forma individual, se colocó una celda sumadora de

1-bit en forma aislada, con sus terminales de alimentación independientes de cualquier otro módulo del circuito integrado. Cabe mencionar también que el arreglo multiplicador tiene una terminal de alimentación independiente a la que utiliza el VCO y la red de distribución de reloj. En la figura 15 se presentan formas de onda experimentales del circuito realizando la operación de multiplicación a una frecuencia de reloj de 50MHz. Se puede observar que la operación de multiplicación experimental (Fig. 15a) concuerda muy bien con la obtenida mediante simulación (Fig. 15b).

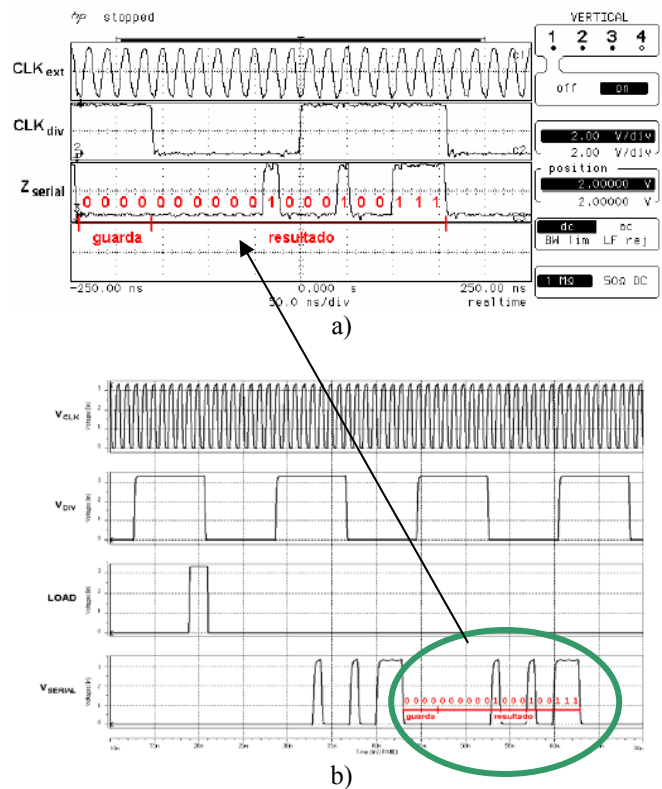


Fig. 15 Formas de onda de la multiplicación $-65 \times 91 = -5915$ (1110 1000 1110 0101)₂ utilizando una frecuencia externa de reloj de 50MHz: a) Experimental, b) Simulación.

IV. CONCLUSIONES

Se ha presentando un conjunto de módulos aritméticos de alto desempeño altamente atractivos para su aplicación en sistemas portátiles multimedia. Algunos módulos propuestos incorporan nuevas estructuras lógicas, usan estilos lógicos de diseño estáticos y dinámicos con el fin de lograr alta velocidad, área reducida y bajos

consumos de potencia. Un arreglo multiplicador de 8×8-bits con arquitectura pipeline considerado como el módulo principal y de mayor tamaño, fue diseñado y fabricado bajo una tecnología CMOS de 0.35micrometros. Los resultados experimentales obtenidos confirman su funcionalidad.

Agradecimientos

Al Conacyt-México por el apoyo parcial a través del proyecto 51511-Y.

V. REFERENCIAS

- [1] E. Abu-Shams, A. Elchouemi, S. Sayed and M. Bayoumi, "An Efficient Low Power Basic Cell for Adders", IEEE ISCAS 1995, pp. 306-308.
- [2] A. Shams and M. Bayoumi, "A Novel Low-Power Building Block CMOS Cell for Adders", ISCAS 1998.
- [3] M. Elgamel, S. Goel and M. Bayoumi, "Noise Tolerant Low-Voltage XOR-XNOR for Fast Arithmetic", GLSVLSI 2003, pp. 285-288.
- [4] Daniel Sanchez, Luke Yen, Mark D. Hill, Karthikeyan Sankaralingam. "Implementing Signatures for Transactional Memory", 40th Annual IEEE/ACM, 2007.
- [5] H. T. Bui, A. K. Al-Sheraidah and Y. Wang, "New 4 Transistor XOR and XNOR designs", IEEE Asia Pacific Conference on ASICs 2000, pp. 25-28.
- [6] M. Elgamel, S. Goel and M. Bayoumi, "Noise Tolerant Low-Voltage XOR-XNOR for Fast Arithmetic", GLSVLSI'03, April 2003, USA. pp. 285-288.
- [7] A. Fayed and M. Bayoumi, "A Low-Power 10-Transistor Full Adder Cell for Embedded Architectures", IEEE ISCAS 2001, pp. 226-229.
- [8] D. Radhakrishnan, "Low-Voltage Low-Power CMOS Full Adder", IEE Proceedings on Circuits Devices and Systems, Vol. 148, No. 1, February 2001, pp. 19-24.
- [9] K. Fujii and T. Douseki, "A Sub-IV bootstrap pass-transistor logic", IEICE Trans. Electronics, Vol. E86-C, No. 4, April 2003, pp.604-611.
- [10] R. Zimmerman and W. Fichtner, "Low-Power Logic Styles: CMOS Versus Pass-Transistor Logic", IEEE ISSC, Vol. 32, No. 7, July 1997, pp. 1079-1090.
- [11] M. Aguirre Hernández, M. Linares Aranda. "Low-power low voltaje 1-bit CMOS full-adder for energy-efficient multimedia applications". ICED 2004, Veracruz, Méx.
- [12] S. Heo, R. Krashinsky, and K. Asanovic, "Activity-Sensitive Flip-Flop and Latch Selection for Reduced Energy", IEEE Trans. on VLSI Systems, vol. 15, no. 9, sep. 2007. pp. 1060-1064.
- [13] A. G. M. Strollo, E. Napolli, and D. De Caro, "New Clock-Gating Techniques for Low-Power Flip-Flops", ISLPD 2000, pp. 114-119.
- [14] K. M. Chu and D. Pulfrey, "A comparison of CMOS circuit techniques: differential cascode voltage switch logic versus conventional logic", IEEE JSSC, Vol. sc-22, No. 4, August 1987, pp. 528-532.
- [15] K. Yano, et al, "A 3.8ns CMOS 16 × 16-b multiplier using complementary pass-transistor logic", IEEE JSSC, Vol. 25, April 1990, pp. 388-395.
- [16] M. Suzuki, et al, "A 1.5ns 32-b CMOS ALU in double pass transistor logic", IEEE JSSC, Vol. 28, No. 11, November 1993, pp. 1145-1150.
- [17] R. Zimmerman and W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic", IEEE JSSC, Vol. 32, No. 7, July 1997, pp. 1079-1090.
- [18] M. Aguirre Hernández, M. Linares Aranda. "An alternative logic approach to implement high-speed low-power full adder cells". 18th SBCCI. 2005. pp. 166-171.
- [19] N. Weste and K. Eshraghian, *Principles of CMOS design, A system perspective*, Addison-Wesley, 1988.
- [20] A. M. Shams and M. Bayoumi, "A 10-transistor low-power high-speed full adder cell", IEEE ISCAS 1999, pp. 43-46.
- [21] A. M. Shams and M. Bayoumi, "A Low Power 10-Transistor Full Adder Cell for Embedded Architectures", IEEE ISCAS 2001, pp. 226-229.
- [22] A. M. Shams and M. Bayoumi, "A new cell for low power adders", MWSCAS, 1995.
- [23] E. Abu-Shams, A. Elchouemi, S. Sayed and M. Bayoumi, "An Efficient Low Power Basic Cell for Adders", IEEE ISCAS 1995, pp. 306-308.
- [24] A. Shams, et al "A novel low-power building block CMOS cell for adders", IEEE ISCAS 1998.
- [26] B. Alhalabi and A. Al-Sheraidah, "A novel low-power multiplexer-based full adder cell", IEEE ICECS 2001, Vol. 3, September 2001, pp. 1433-1436.
- [27] N. Zhuang and H. Wu, "A new design of the CMOS full adder", IEEE JSSC, Vol. 27, No. 5, 1992, pp. 840-844.
- [28] I. Wey, C. Huang and H. Chow, "A new low-voltage CMOS 1-bit full adder for high performance applications", Proceedings of the IEEE APC-ASIC, 2002, pp. 21-24.
- [29] A. Wu and C. K. Ng, "High performance low power low voltage adder", Electronic Letters, Vol. 33, No. 8, 1997.
- [30] T. G. Noll et al, "A pipelined 330-MHz multiplier", IEEE JSSC, Vol. SC-21, pp. 411-416, June 1986.
- [31] F. Lu and H. Samueli, "A 200-MHz CMOS pipelined multiplier accumulator using a quasi-domino dynamic full-adder cell design", IEEE JSSC, Vol. 28, No. 2, pp. 123-132, February 1993.
- [32] S. Jou, C. Chen, E. Yang and C. Su, "A pipelined multiplier- accumulator using a high-speed low-power static and dynamic full adder design", IEEE JSSC, Vol. 32, No. 1, pp. 114-118, January 1997.
- [33] A. M. Shams, et al. "An enhanced low-power computational kernel for a pipelined multiplier-accumulator unit", ICM 1998, pp. 33-36.
- [34] M. R. Satoro and M. A. Horowitz, "SPIM: A pipelined 64x64-bit Iterative Multiplier", IEEE JSSC, Vol. 24 No. 2, 1989.
- [35] HSPICE User Guide, Synopsys, December 2007.
- [36] NanoSim. User Guide (A-2008-03), Synopsys, March 2008.

Implementación en FPGA de un Algoritmo Novedoso para Detección en Línea de Barras Rotas en Motores de Inducción

J. J. Rangel-Magdaleno, R. J. Romero-Troncoso, I. Urbina-Salas
 Electronics Department – HSPdigital FIMEE - Universidad de Guanajuato
 Tampico 912, Col. Bellavista, Salamanca, Gto. Mexico, 36730
 Tel/Fax: +52 (464) 6479940
 e-mail: {jjrangel, troncoso, iurbina }@hspdigital.org

Resumen- *El mantenimiento preventivo es uno de los intereses principales en la industria moderna donde la detección de fallas en motores permite aumentar el ciclo de vida de la maquinaria. La ruptura de barra es uno de los casos más comunes en motores y la supervisión de esta condición es una tarea obligatoria para la industria. Trabajos anteriores sobre la detección de ruptura de barra están basados en el análisis del espectro de corriente o de vibración fuera de línea, además de que su detección es realizada bajo ciertas condiciones de funcionamiento. La novedad de este trabajo es la implementación de un algoritmo de correlación que combina los espectros de corriente y de vibración para mejorar la detección donde otros trabajos fallan. La contribución de este trabajo consiste en que el algoritmo propuesto, a pesar de la complejidad computacional, su implementación es de bajo costo en FPGA dando una solución SOC para la operación en línea, gracias al desarrollo de una unidad especial para el procesamiento de señales en hardware. Varios casos de estudio son presentados para demostrar el funcionamiento de la implementación.*

Palabras clave – *FPGA, SOC, Barras rotas, Correlación.*

1. INTRODUCCIÓN

Hoy en día, el mantenimiento preventivo es uno de los intereses principales en la industria moderna donde la detección de fallas en equipos eléctricos de potencia como transformadores, generadores y motores permite el aumento del ciclo de vida de la maquinaria para evitar daños catastróficos y reducir el costo de producción.

Por otra parte, los motores de inducción de jaula de ardilla representan el 85% de consumo de electricidad en la industria donde la presencia de barras rotas sobre el rotor es una de las fallas más importantes en estos motores, ya que el motor sigue trabajando aparentemente en buenas condiciones sin que se note la falla, pero crece notablemente el consumo de electricidad, el costo de producción y la disminución del ciclo

de vida con vibraciones indeseadas sobre los mecanismos conectados como: engranajes, enganches, bandas de transmisión, etc. Este interés ha promovido el desarrollo de varias técnicas de supervisión de fallas sobre motores de inducción como: análisis de parámetros eléctricos [1-2] análisis de vibración, análisis térmico, radiografía, tintas penetrantes y ultrasonido [3]; sin embargo, la mayor parte de estas técnicas son de naturaleza invasiva o son realizadas fuera de línea. Debido a su naturaleza no invasiva, el análisis de corriente eléctrica es una técnica popular para la supervisión de fallas en los motores donde la instrumentación es realizada mediante un sensor tipo gancho de corriente conectado a la línea de conducción eléctrica que alimenta el motor.

Esta técnica para la detección de barras rotas es realizada mediante la medición de varios componentes espectrales, llamadas f_{srr} , las cuales aparecen cerca de la componente principal espectral que corresponden a la frecuencia de alimentación, llamada f_s , donde la separación entre estas componentes espectrales dependen del deslizamiento s del rotor [3], que depende de la carga mecánica. f_{srr} y f_s son localizadas muy cerca y la detección es comprometida, requiriendo el análisis de un técnico altamente entrenado en el área.

Otra desventaja del uso de la técnica citada es el equipo de análisis espectral que tiene un alto precio y es de uso general, el cual no fue diseñado para la tarea específica de análisis de corriente a bajas frecuencias. La mayor parte de las investigaciones recientes para la detección de barras rotas ha demostrado la ineficiencia para ciertas condiciones de cargas mecánicas en el motor [5-7], donde la separación entre la f_{srr} y f_s es muy estrecha para ser detectada. Otra investigación señala la imposibilidad de la detección de la falla de barras rotas para motores sin carga mecánica [4] con el análisis de vibraciones. Debe notarse que ninguna investigación ha sido aplicada tanto en análisis de vibraciones como de corriente para mejorar la detectabilidad. La contribución de este trabajo es presentar una técnica nueva para la detección de barras rotas que combina el análisis de vibraciones y corriente mediante la correlación de los espectros de dichas señales para mejorar la detectabilidad en las condiciones de funcionamiento sin carga mecánica del motor en donde otras

técnicas aisladas son incapaces de detectar. Además, una unidad de propósito especial para el Procesamiento de Señales en Hardware (HSP) es implementada en un circuito de bajo costo del tipo FPGA (Arreglo de Campo de Compuertas Programables), es desarrollada para la detección en línea del algoritmo de correlación que no requiere el diagnóstico de un técnico experto sumamente entrenado para dar la condición de motor. Junto al algoritmo propuesto, implementado en el FPGA, puede ser configurado de nuevamente para lograr condiciones de medida específicas en varios motores y el equipo industrial sin modificaciones de hardware.

Este trabajo es organizado como sigue: la sección II presenta el fondo teórico de la corriente y el análisis de vibración, el algoritmo propuesto es desarrollado en la sección III, la sección IV muestra la implementación en FPGA del algoritmo propuesto, la sección V presenta el sistema experimental, varios casos de estudio son presentados en la sección VI y la sección VII presenta la discusión y conclusiones.

II. ANÁLISIS DE CORRIENTE Y VIBRACIONES

Como quedo demostrado por [4,9], varias disimetrías en el estator y el rotor pueden ser determinadas por el análisis del espectro de corriente y de vibración en el motor. Las disimetrías eléctricas inducen componentes espectrales cerca de la f_s como componente de la frecuencia principal como lo declarado en (1).

$$\begin{aligned} f_{fLk} &= f_s(1 - 2ks) \\ f_{fRk} &= f_s(1 + 2ks) \end{aligned} \quad k=1,2,3,.. \quad (1)$$

Donde la f_{fLk} es la frecuencia falsa izquierda, f_{fRk} es la frecuencia falsa derecha y s es el deslizamiento asincrónico entre la n_m de velocidad mecánica y la n_s de velocidad rotativa, como lo definido en (2).

$$s = \frac{n_s - n_m}{n_s} \quad (2)$$

Para el motor con carga mecánica, la s de deslizamiento es máxima y la separación entre f_s y los componentes falsos es máxima (ver Fig. 1a); sin embargo, para un motor sin carga, la s es mínima y la f_s es muy cercana a los componentes falsos (ver Fig. 1b).

Fig 1) es una fuente sinusoidal $V(t)$ está conectada a una carga no lineal que dibuja una corriente no lineal $I(t)$ conforme la eq. (1):

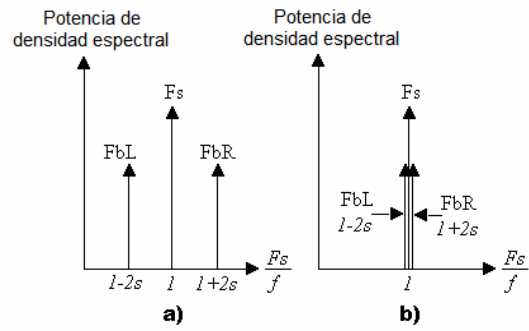


Fig. 1. Características de frecuencia para condiciones de barra rota en motores a) Carga completa. b) Estudio del Circuito Eléctrico sin carga.

III. ESPECTRO DE CORRELACIÓN DE CORRIENTE Y VIBRACIÓN

La investigación anterior [4-7] ha reportado que la detección de condición de barra rota en motores sin carga es comprobada solo por el análisis de corriente. La investigación de análisis de vibración [8-10] también da informes de baja detección para la condición de motor sin carga, que requiere el diagnóstico de un técnico sumamente entrenado, dando la lectura incorrecta de medición. Para vencer el problema bajo detectabilidad con las técnicas citadas cuando sea aplicado solo el algoritmo propuesto tiene la novedad de combinación de ambas técnicas para mejorar la detectabilidad. La mejora es obteniendo por correlación de espectros de corriente y vibración basados en una FFT, ambas señales son adquiridas y procesadas simultáneamente. La corriente es supervisada por un sensor de corriente de gancho estándar, mientras la vibración es supervisada con un acelerómetro de 3 ejes basado en un MEMS económico donde la correlación entre la corriente y la vibración puede ser hecha con cualquiera de los 3 ejes. La correlación de espectros es definida como lo declarado en (3).

$$R_{ss}(m) = E\{I_{ss(n+m)}A_{ss(n)}^*\} = E\{I_{ss(n)}A_{ss(n-m)}^*\} \quad (3)$$

Donde I_{ss} es el espectro de corriente, A_{ss} es el espectro de aceleración, que son procesados conjuntamente, la E es el estimador espectral. Realizando la correlación, las semejanzas sobre las señales pueden ser mejoradas para analizar correctamente las condiciones de barras rotas para el motor sin carga mecánica cuando el análisis de las señales individuales da la lectura incorrecta (ver Fig. 2).

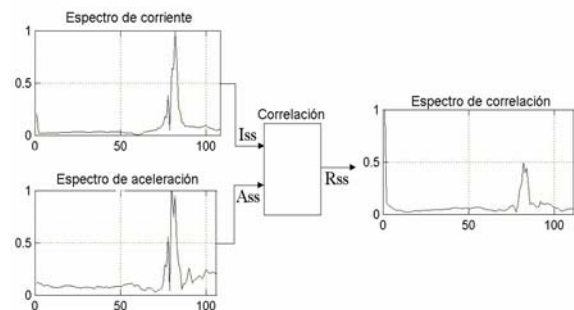


Fig. 2. Espectro de correlación de corriente y vibración.

La figura 3) muestra el diagrama a bloques del algoritmo propuesto. Es realizado por dos FFT simultáneas de 1024-puntos y un procesador de correlación subsecuente, cuyo resultado más lejano puede ser analizado por un bloque de toma de decisiones que simplemente declara la condición de motor en dos valores posibles: bueno o defectuoso, haciendo el proceso en línea sin necesidad de un técnico experto para el diagnóstico.

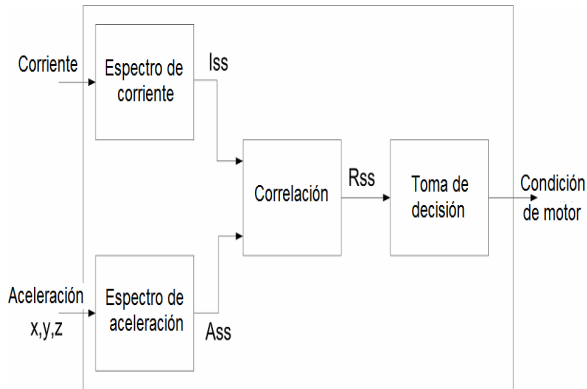


Fig. 3. Diagrama a bloques del algoritmo propuesto.

IV. IMPLMENTACIÓN EN FPGA

La implementación en FPGA tiene la novedad, comparada con los analizadores de espectro disponibles en el mercado de uso general usados con los trabajos citados [4-10], de ser capaz de analizar simultáneamente dos canales (la corriente y un eje de vibración seleccionado por el usuario) y más lejos correlacionar los espectros en SOC (Sistema de Chip) de bajo costo. El plano de edificio de la unidad HSP para el algoritmo propuesto en la Fig. 3) es presentado en la Fig. 4). Una máquina de estados finitos (FSM) realiza el control total de la unidad. El sistema de adquisición de datos (DAS) es controlado por el conductor de datos (DAS), que supervisa el proceso de adquisición de datos del canal de señal de corriente y uno de los tres ejes de aceleración para adquirir la mayor cantidad de datos en el bloque RAM para el cómputo simultáneo de las FFT de dos canales.

A. Módulo FFT

El módulo FFT implementado (ver Fig. 5) fue desarrollado por los autores para la optimización en el área que se tiene en mente para reducir el costo. La estructura de FFT tiene dos bloques de RAM de puerto duales con acceso para lectura y escritura de datos; cuatro multiplexores controlan el camino de los datos al corazón FFT de los datos entrantes al DAS y los datos de regeneración de la raíz 2-mariposa. Una LUT (Look Up Table) el bloque de memoria sólo de lectura es

usada para almacenar los factores del algoritmo FFT. La FFT causa la representación de un número complejo, almacenado en los bloques de RAM: RAM-Real y la RAM-Imag, más tarde es procesado por una unidad de cómputo de módulo ABS, que tiene acceso a los bloques de RAM con dos multiplexores adicionales para el camino del control de datos a las siguientes etapas.

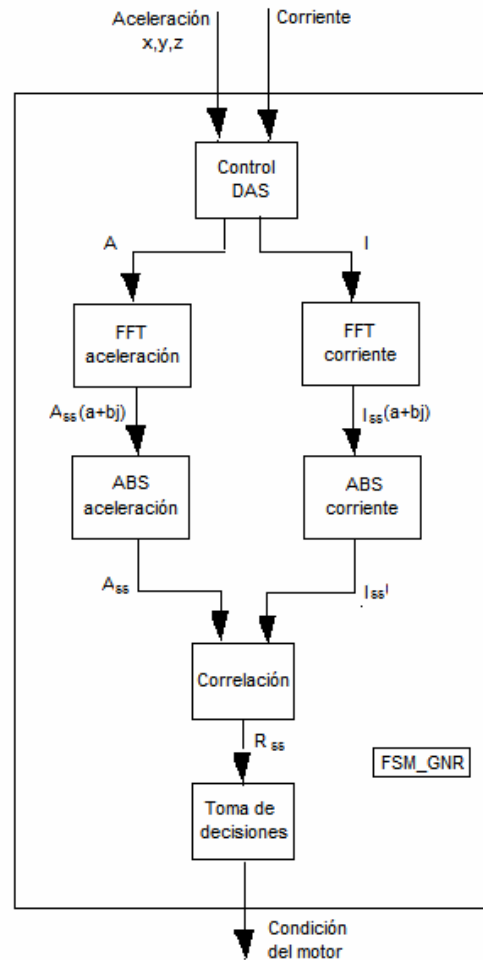


Fig. 4. Implementación en FPGA

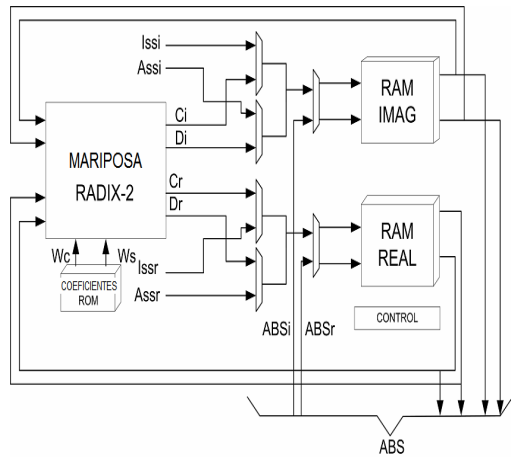


Fig. 5. Módulo FFT

B. Módulos de operación de bloque

La FFT solo da la descomposición de frecuencia de la señal en números complejos como lo declarado en (4) donde $X(k)$ es la componente compleja espectral y la k es el índice del componente espectral. La valoración espectral es realizada por la operación de módulo de los componentes espectrales definidos por (5), donde $|X(k)|_{ss}$ es la magnitud de la valoración espectral de $X(k)$ en el dominio verdadero.

$$X(k) = \text{Re}[X(k)] + j\text{Im}[X(k)] \quad (4)$$

$$|X_{ss}(k)| = |X(k)| = \sqrt{\text{Re}[X(k)]^2 + \text{Im}[X(k)]^2} \quad (5)$$

El bloque de operación del módulo (ver Fig. 6) tiene tres multiplicadores de 18x18-bits, dos de ellos usados para la función cuadrática de las partes reales e imaginarias y el tercer multiplicador como la función inversa del registro de aproximación sucesivo (SAR) el algoritmo para la raíz cuadrada. El resultado es almacenado en un bloque de RAM independiente, llamado RAM-ABS.

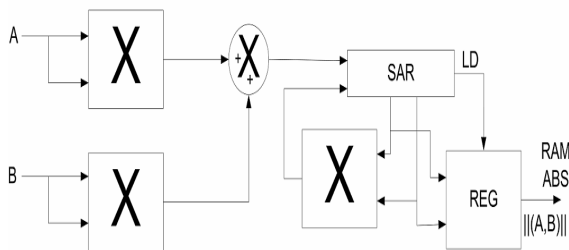


Fig. 6. Módulos ABS

C. Unidad de correlación

La unidad de correlación de la estructura digital (ver Fig. 7) tiene un generador de dirección (ADD-GNR), una unidad de

acumulador de multiplicador (MAC) y un control FSM. El generador de dirección indica la dirección de datos, almacenados en la RAM ABS del bloque de cada estimador espectral, la correlación y el almacenaje de los resultados se realiza en un bloque de RAM independiente, llamado RAM CORR.

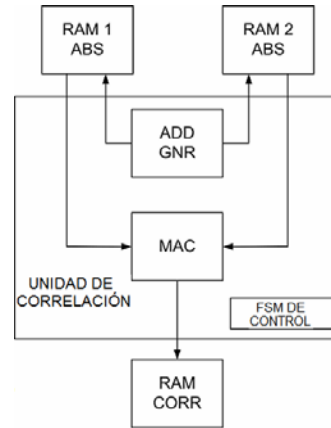


Fig. 7. Unidad de correlación de la estructura digital

Generador de dirección. El generador de dirección de la estructura digital de la unidad de correlación indica las direcciones de datos de los espectros de aceleración y corriente. Esta unidad es construida con dos contadores programables, un bloque de complemento a 2, dos sumadores y una unidad de control (ver Fig. 8).

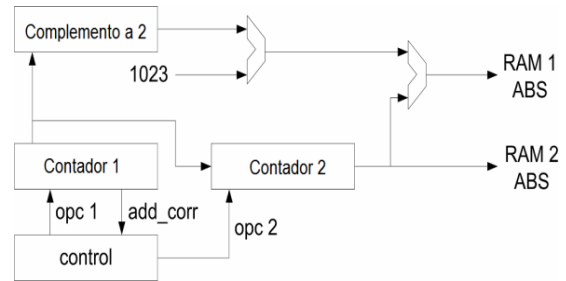


Fig. 8. Estructura digital del generador de dirección.

Unidad MAC. La unidad MAC de correlación es la estructura iterativa conocida (ver Fig. 9), que contiene un multiplicador de 18x18-bits, un sumador, una registro acumulador ACC, un registro de resultado REG y un desplazador a la izquierda BS para el ajuste de datos de punto fijo.

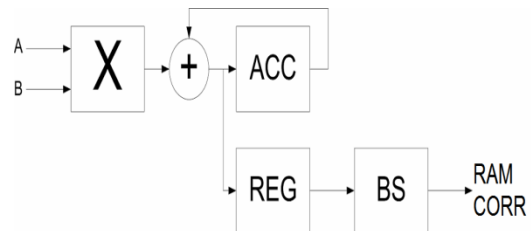


Fig. 9. Estructura digital de la MAC

D. Toma de decisiones

Una vez que la correlación es realizada, una unidad de toma de decisiones simple basada en un umbral comparador declara la condición de motor, indicando si el motor esta en buenas condiciones o en una condición de barra rota (ver Fig. 10).

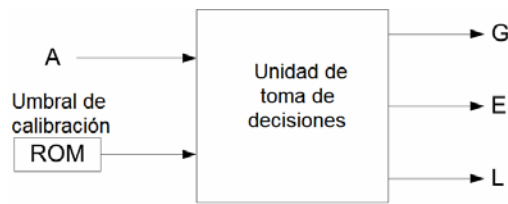


Fig. 10. Comparador de toma de decisiones

La Fig. 11 muestra la estructura digital del sistema total implementado en el FPGA. El proceso entero es realizado como sigue: el conductor DAS controla los datos de entrada de la corriente y el eje de aceleración seleccionado por el usuario para llenar el bloque de RAM-Real para cada unidad FFT mientras se inicializa el bloque RAM-Imag con el valor cero, después los datos son adquiridos; las dos unidades FFT realizan su cómputo con los datos almacenados y los bloques de ABS comienzan la valoración del espectro de cada señal y almacenan los resultados sobre los bloques de RAM-1-ABS y el RAM-2-ABS; entonces la unidad de correlación realiza el cómputo y almacena los datos en el bloque de RAM-CORR; finalmente, la unidad de toma de decisiones compara el resultado de correlación con el umbral calibrado para dar la condición de motor.

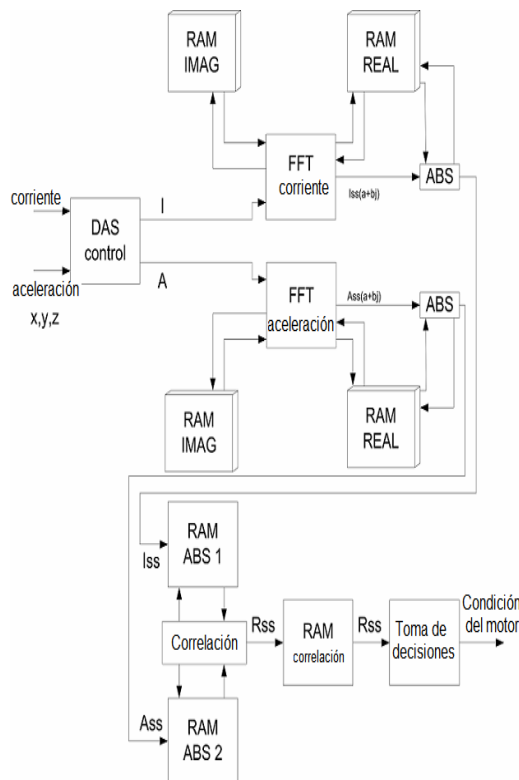


Fig. 11. Sistema total

V. SISTEMA EXPERIMENTAL

Para validar el algoritmo implementado, fueron realizadas varias pruebas. Un banco de pruebas (ver Fig. 12) con un motor de 1-hp del tipo jaula de ardilla sin carga mecánica para condiciones de rotor diferentes: buen estado, una barra rota y dos barras rotas. El sensor de vibración es un acelerómetro MEMS triaxial, que es integrado en el DAS que contiene 12-bits de 3 canales ADC y el acondicionamiento de señal más la interfaz para la transmisión de información sucesiva. La corriente es sensada por un sensor amperimétrico de gancho que es conectado al ADC de 16-bits. La sección digital del DAS (12-bits 3-canales ADC y ADC 16-bits) es conectada al FPGA donde los conductores para la adquisición de datos son integrados como parte del sistema total. La tarifa de muestreo es puesta a 1.5 KSPS (kilo muestras por segundo).

El FPGA seleccionado es la Spartan-3E XC3S500E de Xilinx. La Tabla 1 muestra el uso de recursos del sistema total. La frecuencia máxima, como lo relatado por el proceso de síntesis, es de 27.94 MHz y el sistema trabaja a 25 MHz. Los cálculos enteros algorítmicos son realizados en 102.6 ms.

Tabla 1. Informe de síntesis para uso de recursos

	Usado	Disponibl e	%
Deslizamientos	2301	4656	49
4-entradas LUT	4305	9312	46
Bloques RAM	7	20	35
Multiplicadores de 18*18	20	20	100

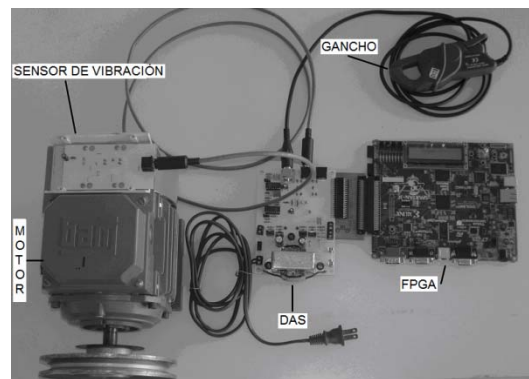


Fig. 12. Banco de pruebas para el sistema experimental

VI. CASO DE ESTUDIO

Tres motores de inducción de jaula de ardilla de 1 HP en condiciones diferentes fueron usados como casos de estudio: en buen estado, una barra rota y dos barras rotas. La fractura

de barra fue hecha por fuera perforando el rotor. Para los tres casos, varias medidas fueron hechas, correlacionando la corriente con cada eje de vibración (x , y y z). Los mejores resultados fueron obtenidos con la correlación en el eje z (ver Fig. 13). La Fig. 13a muestra la correlación de espectros para un motor en buen estado; la Fig. 13b muestra la correlación de espectros para un motor con una barra rota y la Fig. 13c muestra los resultados para un motor con dos barras rotas.

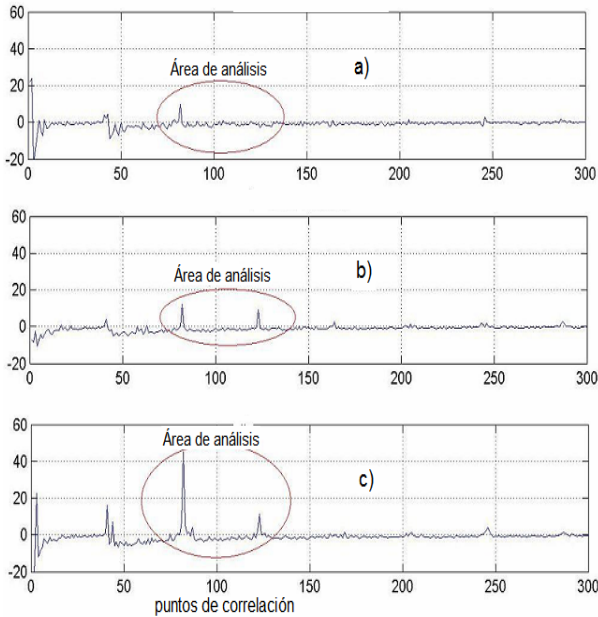


Fig. 13. Resultados de la correlación entre la corriente y el eje-z. a) Motor en buen estado. b) Motor con 1 barra rota. c) Motor con 2 barras rotas.

El parámetro de valoración S_x para el análisis de área entre los puntos de correlación marcó de 75 a 130 (ver Fig. 13) y está basado en la norma de la correlación espectral R_{ss} como lo declarado en (6).

$$S_x = \sqrt{\left(\sum_{x=75}^{130} [R_{ss}(x)]^2 \right)} \quad (6)$$

Para establecer los niveles de umbral para cada condición de motor para la calibración, 25 pruebas fueron hechas para cada condición de motor: en buen estado, una barra rota, dos barras rotas. Muestran la estadística de experimentación en la tabla 2, donde $\mu(S_x)$ es el promedio y $\sigma(S_x)$ es la desviación estándar del parámetro de estimación.

Tabla 2. Umbrales de condición del motor. ($\times 10^{-2} \text{ m/s}^2$)

Condiciones del motor	$\mu(S_x)$	$\sigma(S_x)$
Buen estado	1.563	0.4163
1 barra rota	2.720	0.4952
2 barras rotas	3.710	0.3978

En la Fig. 14., se muestra la experimentación entera para 25 pruebas de cada condición de motor. En esta figura puede verse el promedio y el promedio máximo/mínimo de la desviación estándar para el parámetro estimado en el análisis del área, que establece tres regiones diferentes de detección para la condición particular del motor.

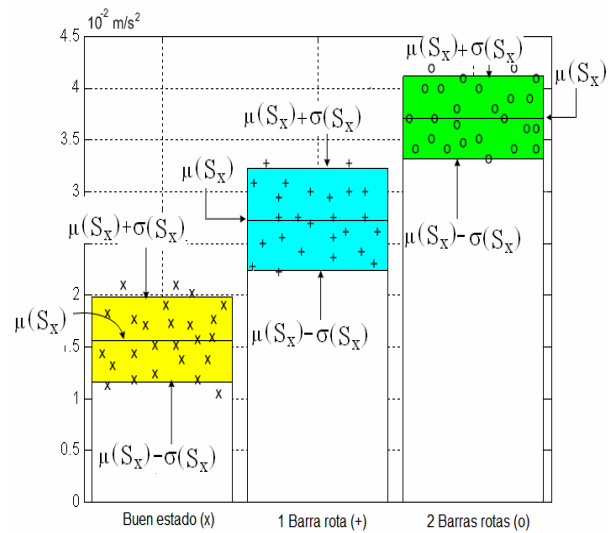


Fig. 14. Resultados experimentales para 25 pruebas con cada una de las condiciones del motor y sus regiones de detección.

VII. CONCLUSIONES

Se propuso un algoritmo nuevo, basado en los espectros de correlación de corriente y aceleración, para la detección de barras rotas en motores de inducción sin carga mecánica. El algoritmo mejora la detectabilidad hasta la condición del motor sin carga mecánica donde otros trabajos reportados fueron incapaces de detectar las fallas [4-10]. Además, por usar un acelerómetro de 3 ejes, la correlación puede ser realizada con cualquier eje y la corriente para mejorar la detección de fallas. Una extensa experimentación de 25 pruebas para cada condición de motor: en buen estado, 1 barra rota, 2 barras rotas, fueron realizadas para verificar la eficiencia de la instrumentación desarrollada. De los resultados puede ser establecido que el algoritmo propuesto da tres umbrales bien diferenciados para cada condición de motor. La contribución de este trabajo consiste en que el algoritmo propuesto, aun con la complejidad en la carga computacional, es puesto en práctica dentro del FPGA de bajo costo que da un objetivo especial SOC para la solución en la

operación en línea, comparada con los algoritmos estándar que requieren la interpretación de un técnico experto que usa analizadores de espectro de objetivos generales.

REFERENCIAS

- [1] M. E. H. Benbouzid, M. Vieira and C. Theys, "Induction motors' faults detection and localization using stator current advanced signal processing techniques," *IEEE Trans. Power Electron.*, vol. 14, no.1, pp. 14-22, Jan. 1999..
- [2] A. Bellini, F. Filippetti, G. Franceshini, C. Tassoni, "Quantitative evaluation motor broken bars by means of electrical signature analysis," *IEEE T. Ind. Appl.*, vol. 37, no.5, pp. 1248-1254, Sept./Oct. 2001. C. Harris, "The detection of rotor defects in induction motors", *Proc. IEE Int. Conf. Electrical Machines, Desing and Applcation*, London, U.K., 1982, pp. 216-220.
- [3] B. Liang, S. D. Iwnicki, "Asymmetrical Stator and Rotor Faulty Detection Using Vibration, Phase Current and Transient Speed Analysis," *Mech Syst Signal Pr*, 2003, pp. 857-869.
- [4] J. Antonio, M. Riera, J. roger-Folch, M.P. Molina, "Validation of a New Method for the Diagnosis of Rotor Bar Failures via Wavelet Transform in Industrial Induction Machines," *IEEE T. Ind. Appl.*, Vol. 42, No. 4, July/Aug. 2006, pp. 990-996.
- [5] H. Douglas, P. Pillay, and A. K. Ziarani, "A new algorithm for transient motor current signature analysis using wavelets," *IEEE T. Ind. Appl.*, Vol. 40, No. 5, Sep/Oct. 2004, pp. 1361 1368.
- [6] H. Douglas, P. Pillay, and A. K. Ziarani, "Broken rotor bar detection in induction machines with transient operating speeds," *IEEE T. Energy Conver.*, Vol. 20, No. 1, Mar. 2005, pp. 135 141.
- [7] S. A. Saleh, A. Kazzaz, G.K. Singh, "Experimental investigations on induction machine condition monitoring and fault diagnosis using digital signal processing techniques", *Elsevier Electric Power Systems Research*, pp. 197-221, 2003.
- [8] F. Filippetti, G. Franceschini, C. Tassoni and P. Vas, "AI techniques in induction machines diagnosis including the speed ripple effect," *IEEE T. Ind. Appl*, vol. 34, no. 1, pp. 98-108, Jan./Feb. 1998.
- [9] G. Niu, T. Han, B-S Yang, A. Chit, C. Tan, "Multi-agent decision fusion for motor fault diagnosis," *Mech Syst Signal Pr*, pp 1285-1299, 2007.

INSTRUMENTO DE BAJO COSTO BASADO EN FPGA PARA EL MONITOREO DE LA CALIDAD DE LA ENERGIA ELECTRICA

Jorge Lara-Cardoso, Rene de Jesus Romero-Troncoso, Arturo García-Perez

Departamento de Electrónica – HSPDigital FIMEE - Universidad de Guanajuato

Tampico 912, Col. Bellavista, Salamanca, Gto. México, 36730

Tel/Fax: +52 (464) 6479940

correo-electrónico: {jlara, troncoso, agarcia}@hspdigital.org

Resumen – El monitoreo de la calidad de la energía eléctrica (CEE) es uno de los temas de mayor interés en la industria moderna, con el objetivo de incrementar la vida útil de las máquinas eléctricas, ahorrar dinero y cumplir con los estándares internacionales. El contenido armónico en la línea eléctrica debido a cargas no lineales causa muchos problemas en la industria, siendo necesario su constante análisis y monitoreo. Los trabajos previos acerca del monitoreo de la calidad de la energía dependen de una computadora para el procesamiento de los datos o el despliegado de unos pocos índices de la calidad de la energía eléctrica (ICEE), por otro lado, los instrumentos comerciales que realizan este análisis tienen la desventaja de ser muy costos e inadecuados para un monitoreo continuo. La novedad de este trabajo es la propuesta de un instrumento autónomo para analizar los armónicos de la línea eléctrica debidos a cargas no lineales, siendo capaz de calcular suficientes ICEE y realizar una evaluación completa y confiable de la CEE. La contribución del artículo es que el algoritmo para este análisis es implementado en un FPGA (Arreglo de compuertas programables en campo) de bajo costo, ofreciendo una solución de propósito especial accesible para la mayoría de las industrias y que satisface la necesidad de monitoreo continuo. La distribución de los recursos utilizados del FPGA y el tiempo de ejecución logrado son reportados. Finalmente, el análisis de dos casos de estudio de sistemas eléctricos reales se utilizan para demostrar el desempeño del instrumento desarrollado y se validan los resultados mediante comparación con un instrumento comercial.

Palabras Clave - FPGA, instrumento autónomo, analizador de armónicos, calidad de la energía, monitoreo en tiempo real.

I. INTRODUCCIÓN

La calidad de la energía eléctrica es un término que expresa las desviaciones de las señales de voltaje y corriente consumidas por la carga, respecto a una forma de onda senoidal ideal, caracterizada por tener una sola componente fundamental y sin variaciones en amplitud y frecuencia. Actualmente, la CEE es un tema muy importante debido al impacto económico que genera en la compañía que ofrece el servicio, así como en los usuarios domésticos e industriales, sufriendo este sector pérdidas millonarias a causa de una mala calidad de la energía caracterizada por la presencia de armónicos en la línea eléctrica [1], produciendo una operación incorrecta de los sistemas de protección, lecturas erróneas de medidores Watt-hora calibrados para ondas senoidales puras, corrientes excesivas en el neutro en sistemas trifásico de 4 hilos, sobrecalentamiento de motores, transformadores, bancos de capacitores y del cableado en general, disminuyendo considerablemente el ciclo de vida útil de los sistemas eléctricos. Asimismo, las cargas no lineales

tales como: hornos de arco eléctrico, fuentes conmutadas de equipo de cómputo, variadores de velocidad y en general cargas controladas por dispositivos eléctricos, consumen una corriente no senoidal, aumentando el contenido armónico de la línea eléctrica y distorsionando el punto común de consumo (PPC). Debido a estos problemas, la CEE está regulada por estándares internacionales como el IEEE 519 [2], IEC 61000 [3], EN 50160 [4], entre otros. El análisis y monitoreo de las señales de la línea eléctrica son esenciales para evaluar la CEE, con el objetivo de encontrar puntos contaminados en la red, implementar técnicas de compensación de armónicos, reducir costos e incrementar la productividad.

Hasta ahora, han sido identificadas dos soluciones para el monitoreo de la CEE, los reportados en la literatura técnica y los instrumentos comerciales. Algunos de los sistemas de monitoreo presentados en la literatura están basados en una PC [5-8] donde el sistema operativo inherente limita el análisis armónico en tiempo real. El resto de los sistemas están basados en DSP-PC [9-15], que inevitablemente dependen de una computadora para la adquisición de señales, el procesamiento de los datos o el desarrollo de la interfaz gráfica de usuario (GUI), por consiguiente no logran ser sistemas autónomos. Aunado a estas desventajas, estos sistemas [8, 13-14] calculan muy pocos ICEE, los cuales son insuficientes para un monitoreo correcto y confiable de la línea eléctrica. Por otro lado, los instrumentos comerciales para el análisis armónico [16] tienen un costo excesivamente alto, por lo que pequeñas y medianas industrias no pueden adquirirlo e incluso empresas grandes no tienen la opción de contar con más instrumentos para monitorear varios puntos de la red al mismo tiempo y de forma continua durante las 24 horas. Una solución óptima al problema planteado sería un sistema autónomo para el monitoreo en tiempo real de la CEE, que no dependa de una computadora, que calcule suficientes ICEE para una buena evaluación de las condiciones de la línea y que tenga la ventaja de ser de bajo costo a fin de que sea accesible para la mayoría de las industrias y así puedan monitorear continuamente sus instalaciones eléctricas.

La contribución de este artículo es el desarrollo de un instrumento basado en tecnología FPGA para el monitoreo en tiempo real de la CEE, siendo un sistema autónomo que no depende de una PC para la adquisición de las señales, el procesamiento de la información o la exhibición de los

resultados, gracias a que cuenta con su propio sistema de adquisición de datos (SAD) y a que maneja directamente un monitor VGA estándar para mostrar gráficamente la magnitud de los primeros 31 armónicos de voltaje y corriente. El instrumento desarrollado calcula hasta 20 ICEE, los cuales son suficientes para una buena evaluación de las condiciones de la línea eléctrica. Estos ICEE son exhibidos uno a uno en una pantalla LCD. Opcionalmente, las señales capturadas de voltaje y corriente, la magnitud y fase del espectro y los ICEE son enviados a una PC a través de la interfaz serial RS-232 con el propósito de su almacenamiento. Todas estas funciones son ejecutadas en línea y dentro de un solo chip, contrario a los trabajos citados [5-15]. Gracias a la implementación en un FPGA de bajo costo, la mayoría de las industrias pueden adquirirlo para monitorear continuamente sus instalaciones eléctricas.

El artículo está organizado como sigue: la sección 2 presenta el fundamento teórico para obtener los ICEE, el desarrollo del instrumento y la implementación FPGA se describe en la sección 3, la sección 4 muestra el montaje del experimento para los casos de estudio, los resultados obtenidos se presentan en la sección 5 y finalmente la sección 6 marca las conclusiones del trabajo.

II. FUNDAMENTO TEÓRICO

A fin de realizar un monitoreo completo de la CEE, el instrumento desarrollado es capaz de calcular para voltaje y corriente los siguientes ICEE [17-18]: valor rms, valor pico, factor cresta, distorsión armónica total referida al valor rms, distorsión armónica total referida a la fundamental, contribución armónica, factor total de potencia, factor de potencia de distorsión, factor de desplazamiento, así como las potencias pico, real, reactiva, aparente y de distorsión.

El circuito eléctrico bajo estudio (ver Fig. 1) es una fuente no senoidal $V(t)$ conectada a una carga no lineal que consume una corriente no lineal $I(t)$ de acuerdo a la ec.(1):

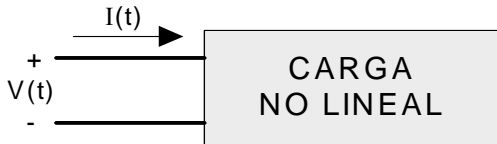


Fig. 1. Circuito eléctrico bajo estudio

$$I(t) = \sum_{n=1}^N I_n \sin(n\omega_0 t + \phi_n) \quad (1)$$

$$V(t) = \sum_{n=1}^N V_n \sin(n\omega_0 t + \theta_n)$$

donde V_n e I_n son las amplitudes de los armónicos de voltaje y corriente, N es el número máximo de armónicos a ser analizados, ω_0 es la frecuencial fundamental de la línea, θ_n y

ϕ_n son la fase de los armónicos de voltaje y corriente, respectivamente.

Las fórmulas matemáticas para calcular algunos de los ICEE se muestran a continuación:

Voltaje rms (V_{rms}) y corriente rms (I_{rms}) son el valor efectivo o root mean square de la señal:

$$V_{rms} = \frac{1}{\sqrt{2}} \sqrt{\sum_{n=1}^N V_n^2} \quad I_{rms} = \frac{1}{\sqrt{2}} \sqrt{\sum_{n=1}^N I_n^2} \quad (2)$$

Factor cresta de voltaje (V_{crest}) y factor cresta de corriente (I_{crest}) es la relación del valor pico de la señal y su valor rms:

$$V_{crest} = \frac{V_{peak}}{V_{rms}} \quad I_{crest} = \frac{I_{peak}}{I_{rms}} \quad (3)$$

donde V_{peak} e I_{peak} son el voltaje pico y corriente pico, respectivamente.

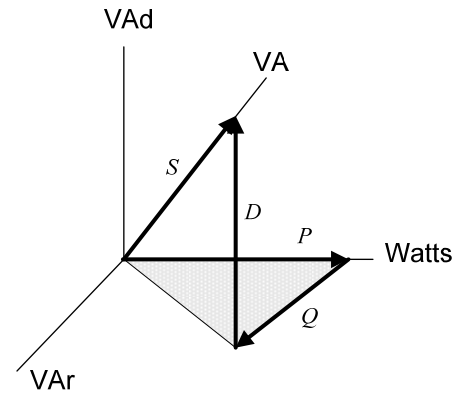


Fig. 2. Triángulo de potencias para cargas no lineales

El triángulo de potencias para cargas no lineales (ver Fig. 2), expresa que la potencia total o aparente S [VA] está formada por dos componentes: la potencia de distorsión D [VAd] y la resultante en el plano horizontal de la potencia real P [Watts] y la potencia reactiva Q [VAR].

Potencia real (P) es la potencia consumida por la parte resistiva de la carga y que se transforma en trabajo útil:

$$P = \frac{1}{2} \sum_{n=1}^N V_n I_n \cos[\theta_n - \phi_n] \quad (4)$$

Potencia reactiva (Q) es la potencia almacenada en forma de campo eléctrico y magnético en la parte capacitiva e inductiva de la carga, representando un consumo de energía pero sin generar trabajo útil:

$$Q = \frac{1}{2} \sum_{n=1}^N V_n I_n \sin[\theta_n - \phi_n] \quad (5)$$

Potencia de distorsión (D) es una componente de la potencia aparente definida como:

$$D = \sqrt{S^2 - Q^2 - P^2} \quad (6)$$

Distorsión armónica total de voltaje referida a la fundamental ($THDV_{fund}$) y distorsión armónica total de corriente referida a la fundamental ($THDI_{fund}$) son una medida porcentual del nivel de distorsión de la señal debida a la presencia de armónicas:

$$THDV_{fund} = \frac{\sqrt{\sum_{n=2}^N V_n^2}}{V_1} \times 100 \quad THDI_{fund} = \frac{\sqrt{\sum_{n=2}^N I_n^2}}{I_1} \times 100 \quad (7)$$

Factor total de potencia (TPF) es una medida de la eficiencia de los sistemas eléctricos, definido como la relación de la potencia real consumida por la carga y la potencia aparente:

$$TPF = \frac{P}{S} \quad (8)$$

III. DESARROLLO DEL INSTRUMENTO E IMPLEMENTACIÓN FPGA

Ningún sistema de monitoreo de los trabajos citados [5-15] está basado en FPGA ni ha logrado la ventaja costo-beneficio del instrumento aquí presentado. La operación general del sistema desarrollado (ver Fig. 3) cuenta con un SAD diseñado por los autores para monitorear simultáneamente el voltaje V a través de un transformador y la corriente I mediante un sensor de efecto Hall.

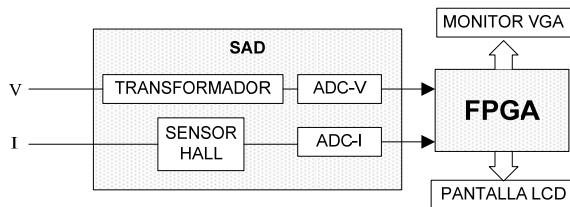


Fig. 3. Operación general del sistema

Estas señales son convertidas a datos digitales mediante los convertidores analógico-digital (ADC) para poder ser almacenadas dentro del FPGA y así llevar a cabo todo el procesamiento de las señales en hardware (HSP), obtener los armónicos de voltaje y corriente y desplegarlos gráficamente en un monitor VGA, así como exhibir los ICEE en una pantalla LCD. El diagrama a bloques de la unidad HSP propuesta para el procesamiento de las señales así como la interfaz con el FPGA se muestra en la Fig. 4.

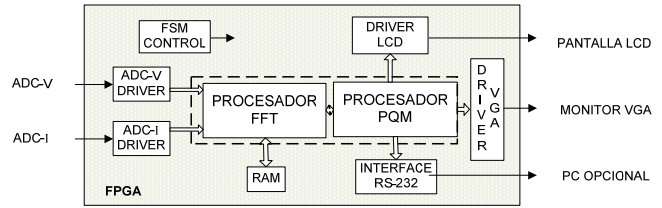


Fig. 4. Diagrama a bloques de la solución hardware propuesta

Los drivers para los convertidores ADC-V y ADC-I permiten recibir los bits seriales de la conversión A/D a fin de almacenar palabras paralelas dentro del FPGA. Una máquina de estados finitos (FSM) controla el proceso de almacenamiento de los 1024 puntos muestra de cada una de las señales de voltaje y corriente dentro de los bloques RAM del FPGA y así proceder al cómputo de su Transformada Rápida de Fourier Radix-2 mediante el PROCESADOR FFT, obteniendo las primeras 31 magnitudes y fases de los armónicos de voltaje y corriente. Este espectro es transferido al PROCESADOR PQM para calcular hasta 20 ICEE, los cuales son mostrados en una pantalla LCD mientras que los armónicos son desplegados gráficamente en un monitor VGA.

A. Procesador FFT

El módulo FFT (ver Fig. 5) es un procesador optimizado para el cómputo de la FFT dual de 1024-puntos Radix-2 con diezmo en tiempo.

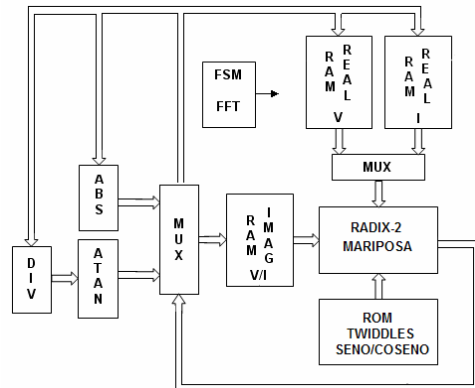


Fig. 5. Ruta de datos para la FFT dual

La estructura digital tiene 3 bloques RAM de doble acceso para lectura y escritura. Inicialmente, los bloques RAM REAL V y RAM REAL I contienen las señales de voltaje y corriente capturadas por el SAD. El proceso comienza con el cálculo de la FFT de la señal de voltaje, almacenando los resultados reales intermedios provenientes de la unidad operativa MARIPOSA en el mismo bloque RAM REAL V, mientras que la parte imaginaria se guarda en el bloque RAM IMAG V/I. Una vez que termina el cómputo de la FFT, se calcula la magnitud de los números complejos obtenidos haciendo uso del módulo ABS. Para obtener la fase de estos números complejos, primero se realiza la división de la parte

imaginaria entre la parte real mediante el bloque DIV y posteriormente se evalúa la función tangente inversa con el módulo ATAN, obteniendo así la fase de los primeros 31 armónicos de voltaje. Ahora, simplemente controlando la ruta de datos mediante el multiplexor MUX y reutilizando el bloque común RAM IMAG V/I, se realiza el mismo procedimiento con la señal de corriente para obtener también su espectro (magnitud y fase). Los factores de fase requeridos por el algoritmo FFT son almacenados en una tabla de consulta de datos (LUT).

B. Procesador PQM

El módulo PQM (ver Fig. 6) es un procesador especial que trabaja en paralelo con el procesador FFT a fin de calcular los ICEE.

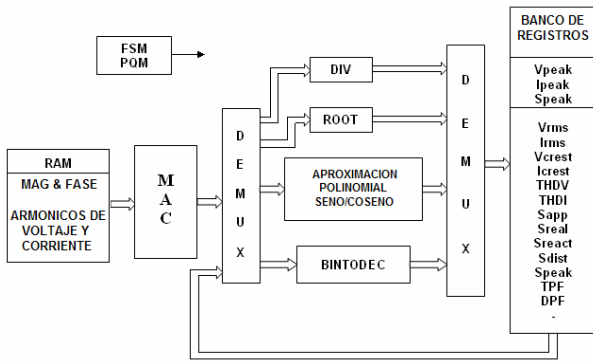


Fig. 6. Estructura digital para el cálculo de los ICEE

La magnitud y fase de los primeros 31 armónicos de voltaje y corriente previamente calculados con el procesador FFT, son copiadas en un nuevo bloque RAM de acceso simple para realizar lecturas simultáneas y facilitar así el cálculo de los ICEE, principalmente la potencia real y la potencia reactiva. La unidad MAC (multiplicador-acumulador) es utilizada para efectuar las multiplicaciones requeridas y la acumulación sucesiva de los resultados parciales y así llevar a cabo la evaluación de las fórmulas de la CEE. De acuerdo a cada una de estas fórmulas, es ejecutada alguna de las siguientes operaciones: la raíz cuadrada mediante el bloque ROOT, la evaluación de la función seno o coseno con el bloque APROXIMACION POLINOMIAL o la división de factores con el módulo DIV. Después de realizar la operación correspondiente, el resultado obtenido es almacenado en un banco de registros de 24-bits. Para el cálculo de un ICEE que depende de valores previamente encontrados, como es el caso del factor cresta de corriente, primero son leídos del banco de registros el voltaje pico y el voltaje rms, y a través de la ruta de retroalimentación se efectúa la operación de división y se sobrescribe el registro con el nuevo valor. Una vez que son calculados todos los ICEE, se lleva a cabo la conversión de código binario a decimal mediante el módulo BINTODEC. El formato de salida es de 3 dígitos enteros y 3 fraccionarios.

Aproximación Polinomial. La estructura digital para realizar la aproximación polinomial (ver Fig. 7) es similar a la unidad MAC, solamente que ahora se realiza primero la acumulación y después la multiplicación.

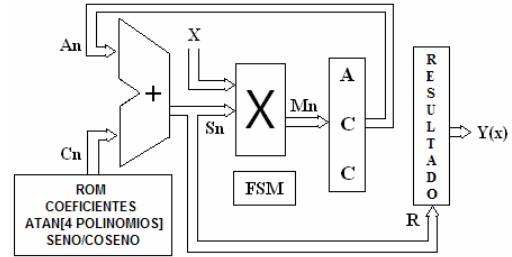


Fig. 7. Estructura digital para la Aproximación Polinomial

Un polinomio expresado de la forma:

$$y(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 \quad (9)$$

es evaluado de forma iterativa como:

$$y(x) = \left[\left[(0 + a_n) x + a_{n-1} \right] x + \dots + a_1 \right] x + a_0 \quad (10)$$

Este módulo se utilizó para evaluar la función tangente inversa y obtener la fase de los armónicos de voltaje y corriente, así como para evaluar las funciones seno y coseno y poder calcular la potencia real, la potencia reactiva y el factor de desplazamiento.

C. Módulo GUI

El módulo digital para el control de la interfaz gráfica de usuario se muestra en la Fig. 8.

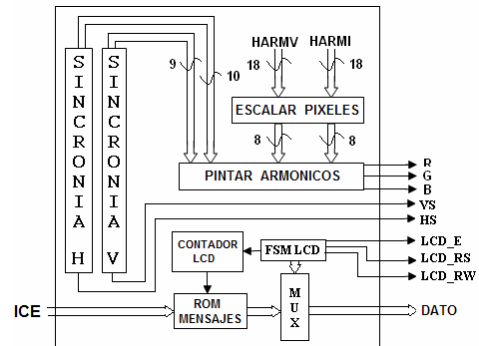


Fig. 8. Módulo GUI

Esta estructura digital maneja directamente un monitor VGA con una resolución de 640x480 pixeles, el cual requiere de la SINCRONIA H horizontal y la SINCRONIA V vertical para poder hacer el barrido sobre la pantalla. El bloque ESCALAR PÍXELES ajusta la magnitud de los armónicos a un máximo de 170 pixeles, mientras que el bloque PINTAR ARMONICOS dibuja los armónicos de voltaje HARMV y corriente HARM I, respectivamente.

IV. MONTAJE EXPERIMENTAL

A fin de validar los resultados arrojados por el instrumento desarrollado, se analizaron dos casos de estudio basados en cargas no lineales con alto contenido armónico. El primer caso de estudio (ver Fig. 9) utiliza el banco de pruebas eléctricas Lab-Volt, el cual consiste de un motor de inducción tipo jaula de ardilla, con la siguientes características: ¼ HP, cuatro-polos, 1670 rpm y una frecuencia de trabajo de 60 Hz. Este motor se conectó en configuración estrella y se alimentó mediante un inversor basado en transistores MOSFET.

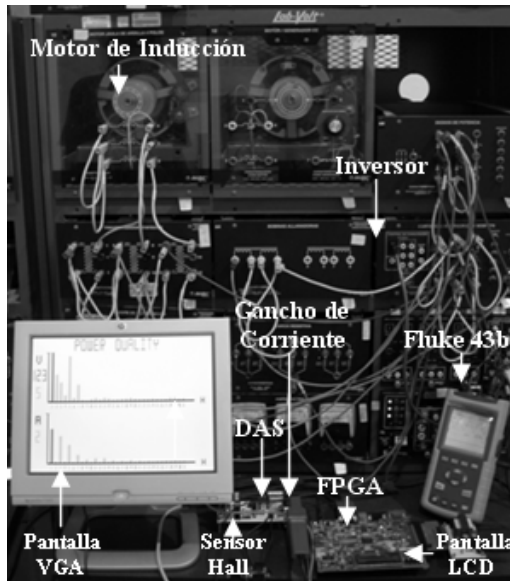


Fig. 9. Banco de pruebas eléctricas Lab-Volt

El segundo caso de estudio consiste en el monitoreo de la CEE de la fuente conmutada de una laptop HP modelo dv6324us. Ambos casos de estudio fueron analizados con el instrumento desarrollado y con el instrumento comercial Power Quality Harmonics Analyzer Fluke 43B [16], con el objetivo de calibrar, comparar resultados y validar el instrumento. El sistema desarrollado cuenta con su propio SAD de dos canales para el monitoreo simultáneo de voltaje y corriente. Los convertidores de aproximaciones sucesivas ADS7886 [19] utilizados en el diseño tienen una resolución de 12 bits y una frecuencia de 20 MHz para la transmisión de datos seriales. El voltaje es sentido a través de un transformador modelo PC-12-100 [20] con 115V de entrada y 6.3V de salida, mientras que la corriente es monitoreada mediante un sensor de corriente de efecto Hall, marca TAMURA modelo L08P100D15 [21]. La sección digital del SAD se conecta directamente al FPGA sin problema alguno, pues los drivers para la adquisición de los datos están embebidos como una parte de toda la implementación FPGA. La frecuencia de muestreo del SAD es de 5,120 Hz mientras que el instrumento trabaja a 25MHz.

V. RESULTADOS

Un motor de inducción tipo jaula de ardilla de ¼ HP alimentado mediante un inversor a base de transistores MOSFET se utilizó como primer caso de estudio. Las señales de voltaje y corriente capturadas con el SAD, así como el espectro obtenido se muestran en la Fig. 10.

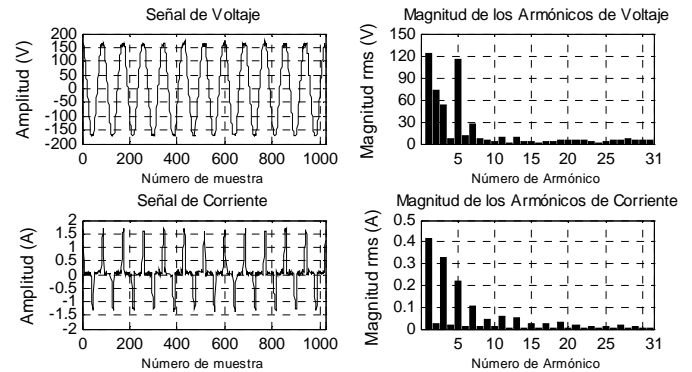


Fig. 10. Señales temporales y espectro del sistema motor-inversor

Las señales temporales y el espectro obtenido para el segundo caso de estudio, consistente en la fuente conmutada de la laptop se muestran en la Fig. 11.

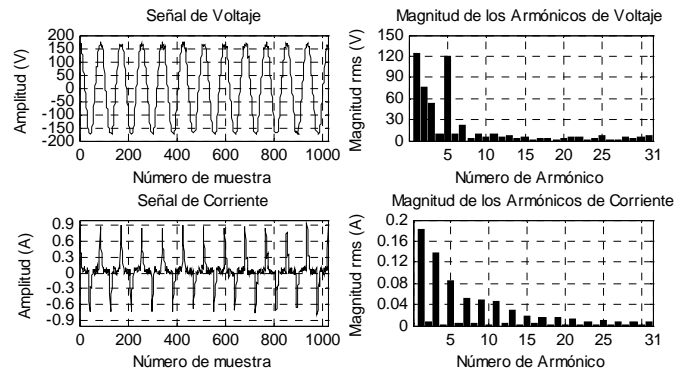


Fig. 11. Señales temporales y espectro de la fuente conmutada de la laptop

A fin de realizar un análisis estadístico de los resultados (media y desviación estándar-SD) se llevaron a cabo 25 pruebas para cada caso de estudio. Este análisis se muestra para algunos ICEE (2)-(8) en las Tablas 1 y 2.

Tabla 1. Resultados del monitoreo del sistema motor-inversor

ICEE	FPGA			FLUKE		
	Media	S.D.	Error (%)	Media	S.D.	Error (%)
Vrms (V)	122.33	0.57	0.46	122.37	0.56	0.03
Icrest	2.49	0.018	0.72	2.50	0.034	0.40
Ipeak (A)	1.718	0.011	0.64	1.708	0.04	0.58
THDIrms (%)	71.16	0.46	0.65	71.32	0.49	0.22
DistPF	0.702	0.004	0.57	-	-	-
D (VAd)	59.48	0.01	0.02	-	-	-

Tabla 2. Resultados del monitoreo de la fuente conmutada de la laptop

ICEE	FPGA			FLUKE		
	Media	S.D	Error (%)	Media	S.D.	Error (%)
Vpeak (V)	170.43	1.08	0.63	170.26	0.97	0.10
Vcrest	1.38	0.01	0.72	1.404	0.02	1.74
dPF	0.996	0.005	0.50	0.989	0.005	0.70
TPF	0.683	0.005	0.73	0.680	0.002	0.44
VHC (V)	6.892	0.222	3.20	-	-	-
THDIfund (%)	93.29	2.16	2.31	-	-	-

A. Recursos Utilizados del FPGA

El reporte de los recursos utilizados por cada módulo dentro del FPGA (Spartan-3E 3S500E de Xilinx) se muestra en la Tabla 3. Nótese que sólo el 23% de las compuertas son utilizadas para el desarrollo de la GUI, mientras que la mayor parte de los recursos son dedicados al Procesador FFT (32%) y al Procesador PQM (38%).

Tabla 3. Recursos utilizados de la Spartan 3E-3S500E

Recursos	Utilizado	Disponible	%		
			Etapa	Module	%
Slices	4571	4656	Adquisición	DAS drivers	2
			Procesamiento	Procesador FFT	32
				Procesador PQM	38
			Interfaz Gráfica de Usuario	LCD driver	7
				VGA driver	16
				RS-232 driver	3
LUT 4 entradas	7988	9312	98		
Bloques RAM	7	20	85		
Mult 18x18b	14	20	35		
			70		

B. Tiempos de Procesamiento

El tiempo de ejecución para el cómputo del espectro de las señales de voltaje y corriente, incluyendo 2-FFT y el cálculo de la magnitud y la fase de los armónicos, le toma al FPGA tan sólo 0.955ms, mientras que los 20 ICEE son calculados en 0.070ms, dando un tiempo total de procesamiento de 1.025ms, garantizando así el monitoreo en tiempo real de la CEE y la operación en línea del instrumento. Con el fin de hacer notar la ventaja del tiempo total de procesamiento al utilizar tecnología FPGA, el mismo algoritmo de análisis armónico se implementó en una PC con dos procesadores Turion a 1.6GHz y 1Gb en RAM, tomándole 203ms, alrededor de 200 veces más lento que la implementación en FPGA. Este tiempo de ejecución supera fácilmente a los sistemas basados en computadora, así como a las soluciones DSP, a las cuales les toma 2.553ms tan sólo para el cómputo de una FFT de 1024 puntos [22].

VI. CONCLUSIONES

Un novedoso instrumento para el monitoreo en tiempo real de la calidad de la energía eléctrica ha sido diseñado, implementado y evaluado. Siendo un instrumento autónomo, al manejar un monitor VGA estándar para mostrar gráficamente los primeros 31 armónicos de voltaje y corriente y al controlar un pantalla LCD para exhibir hasta 20 ICEE, suficientes para realizar una evaluación armónica completa y confiable, presentando todas estas ventajas sobre los trabajos reportados [5-15], los cuales son incapaces de eliminar su dependencia con la PC, calculando pocos ICEE y teniendo tiempos mayores de procesamiento. La contribución de este trabajo es la implementación del algoritmo en un FPGA de bajo costo, ofreciendo una solución accesible para la mayoría de las industrias a fin de que monitoreen sus instalaciones eléctricas de forma continua. El tiempo total de procesamiento en el FPGA es de tan sólo 1.025ms, incluyendo el cómputo simultáneo de 2-FFT, el cálculo de la magnitud y fase de los armónicos y el cómputo de los 20 ICEE, garantizando el monitoreo en tiempo real de la CEE y la operación en línea del instrumento. De los resultados se observa que el instrumento desarrollado tiene un error máximo de 3% en comparación con el Fluke 43B, verificando su correcto funcionamiento y ofreciendo así, una solución óptima para el monitoreo de la CEE y el análisis armónico en cargas no lineales.

RECONOCIMIENTOS

Este proyecto de investigación fue apoyado parcialmente por el CONACYT con el número de beca 207681.

REFERENCIAS

- [1] M. H. J. Bollen, "What is power quality?", Electric Power Systems Research, Volume 66, no. 1, July 2003, pp. 5 – 14.
- [2] IEEE Standard 519-1992, IEEE Recommended Practices and Requirements for Harmonic Control in Electrical Power Systems". April 1999.
- [3] IEC 61000 Standard, Electromagnetic compatibility, International Electrotechnical Commission, 1998.
- [4] EN 50160 Standard, Voltage characteristics of electricity supplied by public distribution systems, European Norm, 1999.
- [5] M. Aiello, A. Cataliotti, and S. Nuccio, "A PC-based instrument for harmonics and interharmonics measurements in power supply systems", Trans. on Measurement, vol. 35, 2004, pp. 371–380,
- [6] J. Batista, J. Alfonso, and J. Martins, "Low-cost power quality Monitor based on a PC", Proc. IEEE Int. Symp. Industrial Electronics, RJ (Brazil), 9-11 June, 2003, pp. 323–328.
- [7] L. Podesta, S. Sangiovanni, and E. Zappitelli, "A new system for distorted three-phase sinusoidal quantities measurement", Trans. on Measurement, vol. 32, 2002, pp. 1–6.
- [8] C. J. Farhat, "Power quality monitoring and analysis", Proc. of International Conference on Energy Management and Power Delivery, Singapore, Volume 2, 3-5 March, 1998, pp.1 – 4.

- [9] J. Gou, B. Xie, P. Enjeti, "A DSP based real time power quality measurement system", Proc. Applied Power Electronics and Exposition, Boston (USA), 23-27 February, 1992, pp. 299 – 302.
- [10] A. R. Abdullah, A.Z. Sha'ameri, "Real Time Power Quality Monitoring System Based On TMS320CV5416 DSP Processor", Proc. of International Conference on Power Electronics and Drives and Systems, Kuala Lumpur (Malaysia), 28 November - 1 December, 2005, pp. 1668 – 1672.
- [11] M. E. Salem, A. Mohamed, S. A. Samad, R. Mohamed, "Development of a DSP-Based Power Quality Monitoring Instrument for Real-Time Detection of Power Disturbances", Proc. of International Conference on Power Electronics and Drives and Systems, Kuala Lumpur (Malaysia), 28 November - 1 December, 2005, pp. 304 – 307.
- [12] D. Gallo, C. Landi, N. Rignano, "Real-Time Digital Multifunction Instrument for Power Quality Integrated Indexes Measurement", Proc. Instrumentation and Measurement Technology, Sorrento (Italy), 24-27 April, 2006, pp. 2271 – 2276.
- [13] A. Lakshmikanth, M. Morcos, "A Power Quality Monitoring System: A Case of Study in DSP-based Solutions for Power Electronics", IEEE Trans. on Instrumentation and Measurement, Volume 50, no. 3, June 2001, pp. 724 – 731.
- [14] A. Lakshmikanth, M. Morcos, and W. White, "A real time system for power quality testing ", IEEE Trans. on Instrumentation and Measurement, vol. 47, Issue 6, Dec. 1998, pp. 1464 – 1468.
- [15] K. Wei , Y. Xiang-wu, L. He-ming, and Z. Li-xia, "Design and realization of power quality monitoring system based on DSP and PCI technique", Proc. 12th Int. Conf. Power Electronics and Motion Control, Portoroz (Slovenia), 30 August-1 September, 2006, pp. 420 – 424.
- [16] Fluke Corporation, Fluke 43B Power Quality Analyzer user's guide, 2001 – 2005.
- [17] IEEE Standard 1459-2000, IEEE Trial-Use Standard Definitions for the Measurement of Electric Power Quantities under Sinusoidal, Nonsinusoidal, Balanced or Unbalanced Conditions", January 2000.
- [18] T. Kneschke, "Distortion and power factor of nonlinear loads", Proc. of ASME/IEEE Joint Railroad Conference, Dallas (USA), 13-15 April, 1999, pp. 47 – 54.
- [19] Burr-Brown Corporation, Miniature SAR analog to digital converter ADS7886 data sheet, Texas Instruments Inc., 2006.
- [20] Signal Transformer Corporation, Miniature low power transformer PC-12-100 data sheet, 2007.
- [21] Avio & Industrial devices B.U., Open loop current sensor Hall effect technology L08P100D15 data sheet, Tamura Corporation., 2007.
- [22] Regional Technology Center, An Implementation of FFT, DCT, and Other Transforms on the TMS320C30 data sheet, Texas Instruments Inc., 1997.

INSTALACIÓN DE FILTROS DE ARMÓNICAS EN LA COMISIÓN FEDERAL DE ELECTRICIDAD, DIVISIÓN DE DISTRIBUCIÓN CENTRO SUR, PARA DISMINUIR EL EFECTO DE LAS ARMONICAS.

M.A. Antonio Sánchez Isais
antonio.sanchez@cfe.gob.mx
Comisión Federal de Electricidad
Unidad de Ingeniería Especializada
Jefe del Departamento de Electrónica

M.C. Salvador López Arce
salvador.lopez@cfe.gob.mx
Comisión Federal de Electricidad
Unidad de Ingeniería Especializada
Departamento de Electrónica

Resumen.

Este trabajo pretende dar a conocer la importancia que la Comisión Federal de Electricidad le da a la Calidad de la Energía (Power Quality), así como las medidas que se toman en la compensación paralelo de los Sistemas de Distribución, se pretenden mostrar las experiencias que se han tenido en la puesta a punto y energización tanto de bancos de capacitores como de filtros de armónicas a nivel nacional y en particular se hablará de las experiencias de la energización del filtro de 22.5 MVARS ubicado en la S.E. Tejupilco, en el Estado de México.

En este trabajo se mencionan muchos aspectos que deben de tomarse en cuenta para la correcta instalación y operación de filtros de armónicas.

Introducción

Desde hace aproximadamente 6 años se están instalando un gran número de bancos de capacitores y filtros en la República Mexicana para efectos de mejorar la regulación de voltaje, para la compensación de la potencia reactiva, así como para filtrar las componentes armónicas presentes en algunos nodos del Sistema Eléctrico Nacional. Dentro de esta compensación se encuentran el paquete 33 con 86 bancos de capacitores a Nivel Nacional.

Algunos bancos de capacitores instalados, no están operando adecuadamente debido a que la capacidad del banco no es la adecuada originando escalones de voltaje muy elevados, o porque en lugar de banco de capacitores se

requería un filtro para corregir problemas de armónicas.

Posteriormente dentro del paquete 405 de compensación reactiva en alta tensión se incluyeron algunos filtros, uno de los cuales es tema del presente trabajo.

En el paquete de compensación en alta tensión 705 nuevamente se incluyen filtros. Es muy importante conocer las características particulares del lugar en donde se van a instalar el filtro, para que el diseño de este, sea lo más apegado a la realidad.

Dimensionamiento del banco de capacitores o filtro.

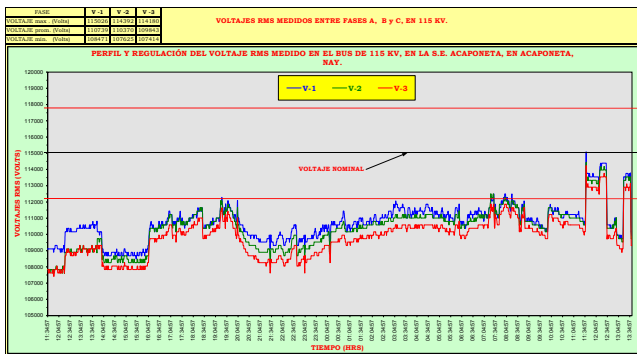
El dimensionamiento del banco de capacitores o filtro depende de los requerimientos de potencia reactiva del nodo y de los pasos con que éste debe entrar a la variación del voltaje. Cuando alguno de estos equipos es conectado o desconectado, el voltaje del sistema se incrementa o disminuye respectivamente.

Para tener un efecto mínimo en las cargas de los consumidores, este cambio de voltaje se debe limitar de un 2 a un 3 % del voltaje nominal, en donde se va a conectar el filtro.

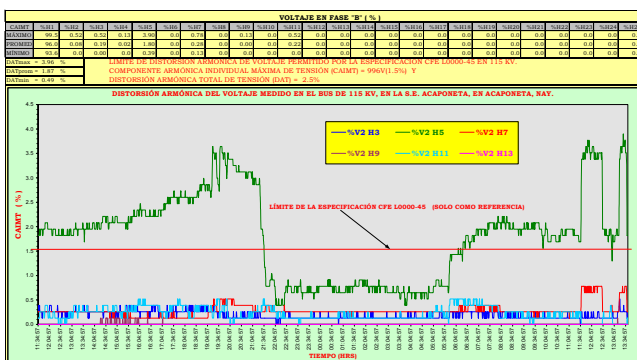
Para el dimensionamiento del filtro no solamente se deben tomar en cuenta estudios de flujos de potencia, si no que es importantísimo realizar mediciones, para conocer las características particulares del nodo en donde se pretende

En este caso se observa que hicieron falta mediciones de calidad de la energía antes de decidir entre un banco de capacitores o bien un filtro de armónicas, puesto que las mediciones hubieran proporcionado información para decidirse por la instalación de un filtro de 5ª armónica, en lugar de un banco de capacitores.

Banco de Capacitores de 7.5 MVARS, en el bus de 115 kV, en la S.E. Acaponeta, en Acaponeta Nayarit.



Gráfica 3. Gráfica del perfil de voltaje presentado al energizar el banco de capacitores.



Gráfica 4. Gráfica de las componentes individuales de voltaje registradas durante la energización del banco de capacitores.

De las gráficas anteriores podemos decir que se requerían de mediciones de calidad de la energía antes de realizar la construcción de estos bancos de capacitores.

Para el Banco de Capacitores de la S.E. Villa de arista (Gráficas 1y 2), posteriormente se encontró que la componente de 5ª armónica era muy elevada debido a la **cercanía de fuentes armónicas**.

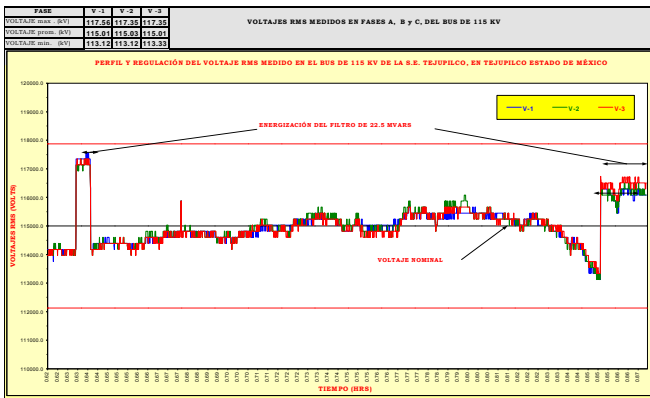
Para el segundo caso, el de la S.E. Acaponeta (Gráficas 3 y 4), se concluyó que debido a que era una subestación cola de red, presentaba características de nodo muy débil, lo que hacía que funcionara prácticamente como sumidero de las componentes armónicas presentes en la red.

Estos dos bancos de capacitores pueden operar, de forma adecuada y sin el riesgo de que entren en resonancia con el sistema y se dañen, si los convertimos en filtros, solamente adicionando reactores en serie para sintonizarlos a una frecuencia de 4.5 p.u. para filtrar 5ª armónica.

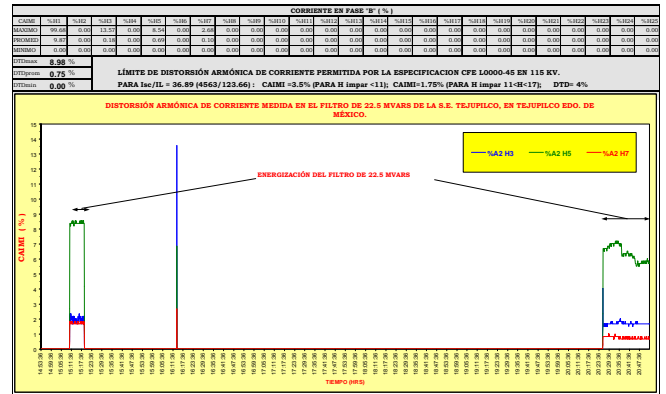
Hay que tener en cuenta muchos aspectos al modificar de banco de capacitores a un filtro, por ejemplo el voltaje de nominal de operación y el voltaje máximo de diseño de las unidades capacitoras existentes es un aspecto importantísimo al realizar un estudio en donde se tome en cuenta el reactor. Se requiere verificar si estas unidades pueden trabajar satisfactoriamente como filtro o bien realizar alguna reconfiguración al banco de capacitores antes de convertirlo a filtro.

Energización del Filtro de 22.5 Mvars en el Bus de 115 kv de la S.E Tejupilco, en Tejupilco Estado de México.

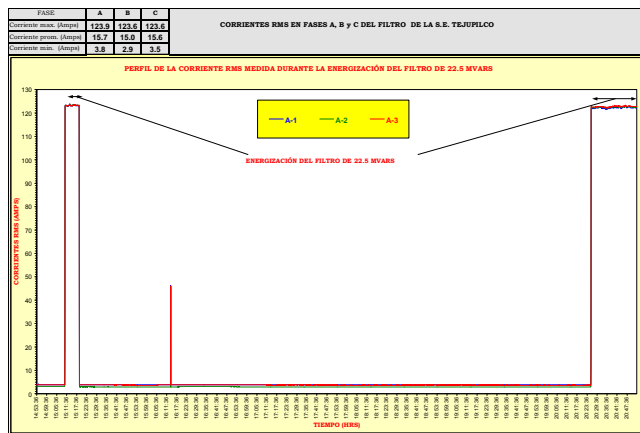
La energización del filtro de 22.5 Mvars ubicado en Tejupilco, en el Estado de México, ayuda en gran medida abatiendo el contenido de 5ª armónica presente en este punto del sistema.



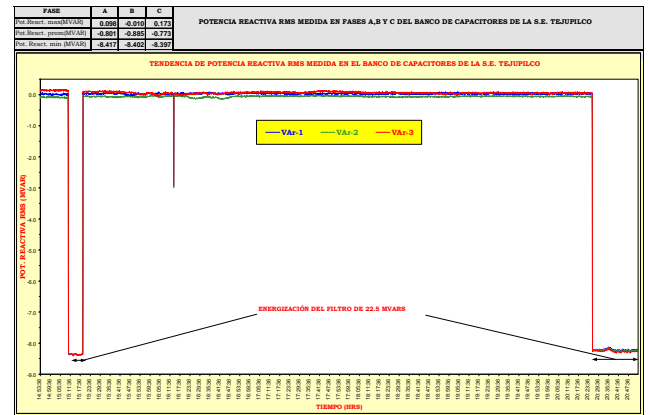
Gráfica 5. Gráfica del perfil de voltaje presentado al energizar el filtro



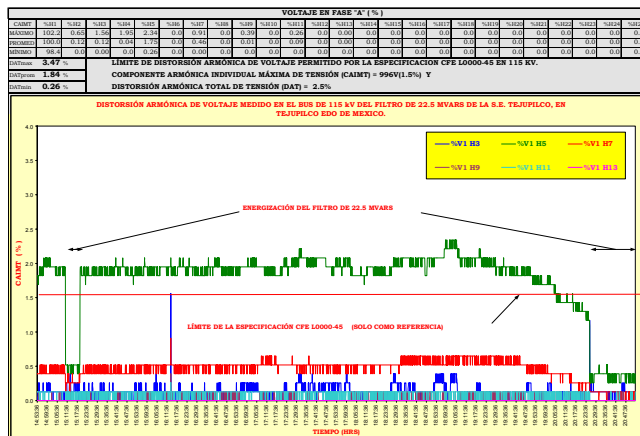
Gráfica 8. Gráfica de las componentes individuales de corriente registradas durante la energización del filtro de 5ª armónica



Gráfica 6. Gráfica del perfil de corriente en el filtro de 5ª armónica.



Gráfica 9. Gráfica de la potencia reactiva suministrada durante la energización del filtro de 5ª armónica.

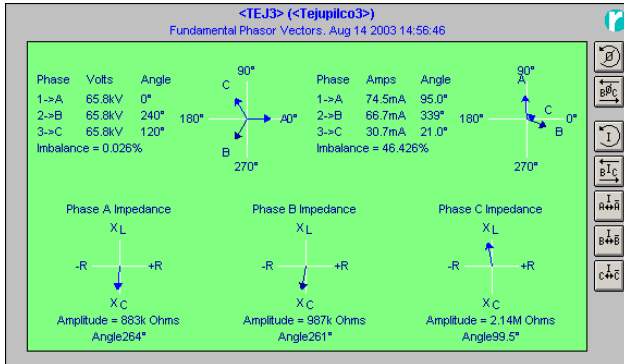


Gráfica 7. Gráfica de las componentes individuales de voltaje registradas durante la energización del filtro de 5ª armónica.

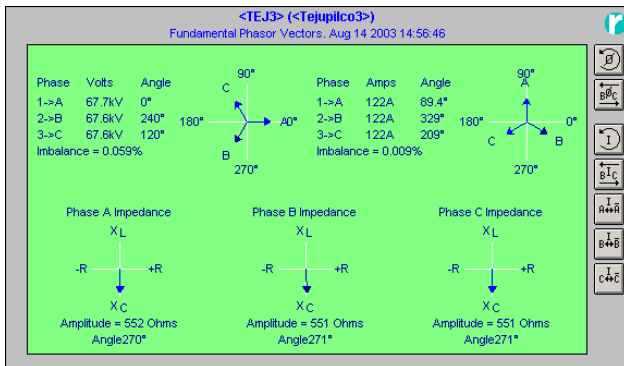
Como se puede observar en las gráficas, se realizaron dos energizaciones del filtro y a continuación se muestran los diagramas vectoriales de estas energizaciones:

PRIMERA ENERGIZACIÓN

La figura de abajo muestra los diagramas vectoriales justamente antes de energizar el filtro, se observa que no existen corrientes.

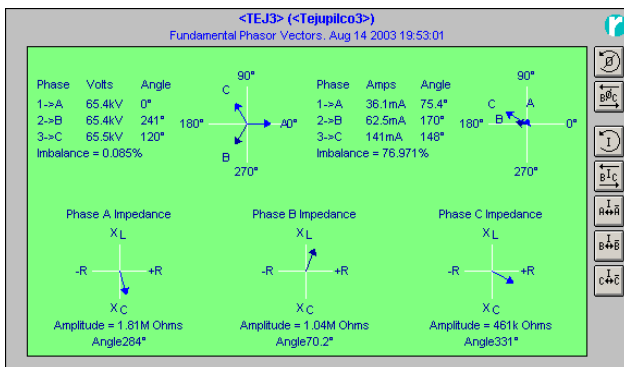


La figura de abajo muestra los diagramas vectoriales en el momento de energizar el filtro.



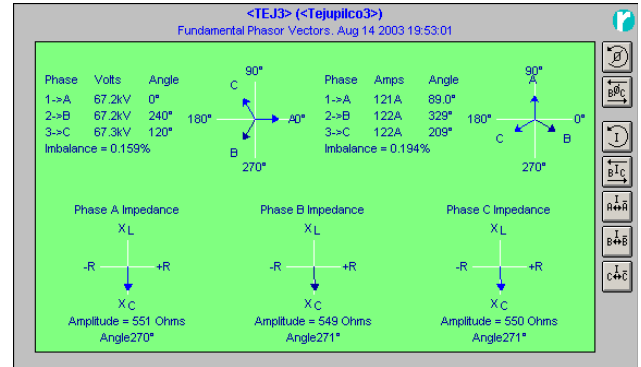
DESCONEXIÓN DEL FILTRO

La siguiente gráfica muestra el diagrama vectorial en el instante en que se desconecta el filtro.



SEGUNDA ENERGIZACIÓN

Esta gráfica muestra la segunda energización del filtro.



VALORES MONITOREADOS

EN EL MOMENTO DE LA 1ª ENERGIZACION TENEMOS:

Escalón de voltaje medido = $(67.6 \text{ kV} - 65.8 \text{ kV}) \times \sqrt{3}$
 = 3.11 kV
 I del filtro = 122 Amperes
 Potencia suministrada por el filtro = 8.37 + 8.37 + 8.35 = 25.09 MVAR'S

EN EL MOMENTO DE LA 2ª ENERGIZACION TENEMOS:

Escalón de voltaje medido = $(67.2 \text{ kV} - 65.4 \text{ kV}) \times \sqrt{3}$
 = 3.11 kV
 I del filtro = 122 Amperes

CALCULOS REALIZADOS :

$I_{cc} 3f = 4563$ Amperes
 $MVA_{cc} 3f = 908.9$ MVAS
 $\Delta V = 2.47\% = 2.84$ kV
 $C = 2.256 \times (10/5) = 4.512$ μF
 $X_C = (1000 \ 000)/(377 \times 4.512) = 587.88$ Ω
 $X_L = 28.865$ Ω
 $L = 76.565$ mH
 Impedancia del Filtro = 559.01 Ω
 Potencia monofásica del banco = $(118/\sqrt{3})/559.01 = 8.30$ MVARs
 Potencia Total del Filtro = 24.9 MVARs

CONCLUSIONES.

Usualmente la solución al problema de armónicos es eliminar los síntomas y no el origen, los aparatos que crean los armónicos generalmente constituyen una pequeña parte de la carga, eliminar su uso no es posible, modificar esos equipos para que no causen armónicos tampoco es factible. Lo que nos queda es reducir los síntomas ya sea incrementando la tolerancia del equipo y del sistema a los armónicos o modificar los circuitos y los sistemas para reducir su impacto, atrapar, o bloquear los armónicos con filtros. Por supuesto hay excepciones. En casos de sobrecarga, daño de equipo o diseño inapropiado, estas causas que generan armónicos pueden ser corregidas, similarmente un aparato o equipo particular que produce un alto nivel de armónicos debe ser modificado o reemplazado.

Un aspecto que con frecuencia es mal evaluado, es que los armónicos han sido un problema reciente debido al efecto de adición y multiplicación de los mismos, la presencia de estos efectos es lo que causa problemas, individualmente ninguno es problemático por sí mismo. Por ejemplo, la distorsión armónica causada por un motor de inducción, que se usa para hacer circular aire para uso agrícola, puede haber sido tolerado por muchos años, pero inesperadamente causa problemas de flicker porque el conductor neutro se abrió. Es común en el caso de severos problemas de armónicas, que se ligen dos o más factores que contribuyan a agravar el problema, particularmente cuando se adiciona equipo nuevo o que existen cambios de equipo, siendo la sospecha del problema los nuevos equipos. Cuando se presentan causas simultáneas que generan altos niveles de armónicas, usualmente sólo una es la mayor causa del problema, contribuyendo las otras causas a crear resonancia o a ayudar en su propagación.

El primer paso que se recomienda en cualquier investigación sobre el problema de armónicas es inspeccionar el equipo y el circuito eléctrico.

Estos problemas son causados o empeorados por cargas desbalanceadas, mala conexión a tierra, problemas con el conductor neutro, por problemas con equipo o por uso inapropiado. Esto puede ser identificado con una inspección cuidadosa con equipo apropiado.

Desde el punto de vista de sobretensiones transitorias y armónicas, disminuyendo la impedancia a tierra e incrementando la ampacidad del neutro con frecuencia se resuelven problemas de calidad de energía (incluyendo armónicas). Los problemas de puesta a tierra contribuyen de un 33 a un 40% de los problemas relacionados con la calidad de energía

Los filtros se utilizan para bloquear o atrapar la energía de los armónicos de tal manera que no fluya por los equipos o que no entre al sistema, son las dos soluciones más usadas para el problema de las armónicas.

Aunque los filtros son elementos cuya impedancia varía con la frecuencia, también tienen el potencial de crear y amplificar el problema de las armónicas, a menos que cuidadosamente sean localizados y diseñados.

Los filtros pasivos son usados por las empresas como circuitos en paralelo en la entrada de los servicios con problemas de generación de armónicas, evitando de esta manera que entren al sistema de distribución, los filtros pueden instalarse directamente en un equipo particular donde existe un grave problema de generación de armónicas, evitando de esta manera que circulen en la propia instalación eléctrica del usuario. También se pueden instalar filtros en las Subestaciones Eléctricas a fin de evitar que las armónicas circulen tanto por los clientes como por el propio sistema de la CFE.

El comportamiento de los filtros pasivos es ser sensitivos a la impedancia del sistema para los cuales ellos fueron ajustados. La impedancia del sistema puede cambiar a lo largo del tiempo,

como el equipo altera su comportamiento de volts/var, siendo difícil estimar su exactitud si no se tienen mediciones, de aquí que la importancia de las mediciones de calidad de la energía.

Los filtros pasivos con frecuencia no proporcionan un comportamiento satisfactorio, bajo ciertas circunstancias pueden causar problemas de resonancia sobre el sistema donde están conectados.

BIBLIOGRAFÍA

IEEE Std. 519-1992 IEEE Recommended practices and requirements for harmonic control in electrical power systems.

CFE L0000-45 Perturbaciones permisibles en las formas de onda de tensión y corriente del suministro de energía eléctrica.

IEEE Std. 18-1992 IEEE Standard for shunt power capacitors

IEEE Std. 1036-1992 IEEE Guide for application of shunt power capacitors

Curriculum



M.C. SALVADOR LÓPEZ ARCE

Ingeniero mecánico electricista egresado de la Esime (IPN) 1992-1996

Maestría en sistemas eléctricos de potencia S.E.P.I. IPN 2000-2001

1997- a la fecha. Comisión Federal de Electricidad, Puestos ocupados:

Ingeniero especializado del Departamento de Compensación de la Red y Estudios Especiales.

Ingeniero Especializado del Departamento de Electrónica



M.A. ANTONIO SÁNCHEZ ISAÍS

Ingeniero mecánico electricista egresado de la U.N.A.M. 1982-1986

Maestría en Administración de Empresas en la Universidad Anahuac del Sur 2000-2001.

1986- 1987. Instituto de Investigaciones Eléctricas

1987- A la fecha. Comisión Federal de Electricidad,

Puestos ocupados:

Ingeniero Especializado

Subjefe del Departamento de Compensación de la Red y Estudios Especiales

Actualmente Jefe de Departamento en la Unidad de Ingeniería Especializada en CFE.

CONTROL GPI DE UN MOTOR SÍNCRONO DE IMANES PERMANENTES

Mendieta Antúñez, Antonio, Campero Littlewood, Eduardo y Olivares Galván, J. Carlos

Departamento de Energía, Unidad Azcapotzalco, Universidad Autónoma Metropolitana
Av. San Pablo No. 180, Col. Reynosa Tamaulipas, México, D.F. 02200, México
Correo electrónico: teoma_i@hotmail.com

RESUMEN

En este artículo se plantea un control de velocidad para un motor síncrono de imanes permanentes que alimenta a una carga de par desconocido. Se diseña una ley de control utilizando la técnica de control integral proporcional generalizado para rechazar la perturbación del par y lograr un seguimiento de la velocidad del rotor. La ley de control se aplica a un motor específico, se simula el comportamiento del motor y se analizan los resultados.

Palabras clave: Motor Síncrono de imanes permanentes, control Integral Proporcional Generalizado, rechazo a perturbaciones.

I. INTRODUCCIÓN

El motor síncrono (MS) de imanes permanentes (IP) tiene características que lo hacen adecuado para el control de velocidad y par, y por lo tanto para su aplicación como servomotor o como accionador eléctrico para posicionamiento. De esta manera resulta ser un buen candidato para usarse como fuente de movimiento controlado en equipos tales como robots, máquinas herramientas y ascensores. De manera sencilla puede decirse que el MS de IP consiste de un estator embobinado de tal forma que produce una fuerza magnetomotriz (FMM) que gira a una velocidad que está en sincronía con la frecuencia de las corrientes que circulan por las bobinas. El rotor tiene acondicionados unos imanes permanentes cuya FMM interactúa con la FMM del estator de forma que se produce un par. Este par depende de la magnitud de las FMM y del ángulo entre ellas. El control de MS de IP puede lograrse con diferentes técnicas. Por ejemplo con control por campo orientado usando retroalimentación estática de los estados [1] o mediante la técnica de interconexión y asignación de amortiguamiento (IDA, por sus

siglas en inglés) del control basado en pasividad [2], además se tienen esquemas adaptivos [3]

En este artículo se utiliza la técnica de control Integral Proporcional Generalizado (GPI, por sus siglas en inglés) [4] para controlar la velocidad en un MS bifásico que tiene sus IP con una geometría tal que el campo magnético radial que produce en el entrehierro tiene una distribución senoidal. Este trabajo incluye una simulación donde se aprecia el comportamiento de la velocidad a un par que varía de manera senoidal.

En la sección II se describe el modelo matemático y la transformación $d-q$ que nos lleva a un modelo más sencillo para lograr encontrar la ley de control que nos garantice seguimiento de trayectorias en la velocidad del rotor. En la sección III se diseña la ley de control usando campo orientado y se logra el rechazo del par mediante el GPI. En la sección IV se muestran los resultados de la simulación, que se lleva a cabo utilizando Simulink de Matlab. Finalmente en la sección V aparecen las conclusiones del trabajo.

II. MODELO MATEMÁTICO

A continuación se muestra el modelo matemático del MS de IP propuesto en [1] y que se utiliza en este trabajo para el diseño del controlador:

$$\begin{aligned}\frac{di_{sa}}{dt} &= -\frac{R_s}{L_s}i_{sa} + \frac{k_m}{L_s}\sin(n_p\theta)\omega + \frac{1}{L_s}u_{sa} \\ \frac{di_{sb}}{dt} &= -\frac{R_s}{L_s}i_{sb} - \frac{k_m}{L_s}\cos(n_p\theta)\omega + \frac{1}{L_s}u_{sb} \\ \frac{d\omega}{dt} &= \frac{k_m}{J}[-i_{sa}\sin(n_p\theta) + i_{sb}\cos(n_p\theta)] - \frac{1}{J}T_L \\ \frac{d\theta}{dt} &= \omega\end{aligned}\quad (1)$$

donde:

i_{sa} : es la corriente de la fase a del estator.

i_{sb} : es la corriente de la fase b del estator

ω : es la velocidad angular del rotor

Cabe mencionar que este modelo de máquina síncrona bifásico puede ser también considerado como una transformación del modelo de un estator trifásico [1].

Ahora se aplica la transformación a los ejes d - q para referir las componentes de corriente y voltaje al marco de referencia del rotor (el eje directo d coincide con el eje del campo magnético del rotor y el eje de cuadratura q está a 90° del eje directo) [5]; que para los voltajes resulta

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = [Q] \begin{bmatrix} u_{sa} \\ u_{sb} \end{bmatrix} \quad (2)$$

y para las corrientes

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = [Q] \begin{bmatrix} i_{sa} \\ i_{sb} \end{bmatrix} \quad (3)$$

donde:

$$Q = \begin{bmatrix} \cos(n_p \theta) & \sin(n_p \theta) \\ -\sin(n_p \theta) & \cos(n_p \theta) \end{bmatrix} \quad (4)$$

Aplicando (2) y (3) se logra transformar el sistema (1)

$$\begin{bmatrix} \frac{di_d}{dt} \\ \frac{di_q}{dt} \end{bmatrix} = [Q] \begin{bmatrix} \frac{di_{sa}}{dt} \\ \frac{di_{sb}}{dt} \end{bmatrix} + \left[\frac{dQ}{dt} \right] \begin{bmatrix} i_{sa} \\ i_{sb} \end{bmatrix}$$

A continuación el nuevo sistema:

$$\begin{bmatrix} \frac{di_d}{dt} \\ \frac{di_q}{dt} \\ \frac{d\omega}{dt} \\ \frac{d\theta}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R_s}{L_s} i_d + n_p \omega i_q + \frac{1}{L_s} u_d \\ -\frac{R_s}{L_s} i_q - n_p \omega i_d - \frac{k_m}{L_s} \omega + \frac{1}{L_s} u_q \\ \frac{k_m}{J} i_q - \frac{1}{J} T_L \\ \omega \end{bmatrix} \quad (5)$$

III. DISEÑO DE LA LEY DE CONTROL.

El objetivo de control es llevar a $\omega \rightarrow \omega^*$ (valor deseado de la velocidad) con el inconveniente que T_L varía en el tiempo.

Entonces se propone la linealización de la siguiente forma:

$$\begin{aligned} u_d &= L_s \left[\frac{R_s}{L_s} i_d - n_p \omega i_q + v_d \right] \\ u_q &= L_s \left[\frac{R_s}{L_s} i_d + n_p \omega i_q + \frac{k_m}{L_s} \omega + v_q \right] \end{aligned} \quad (6)$$

Retroalimentando se tiene:

$$\begin{aligned} \frac{di_d}{dt} &= v_d \\ \frac{di_q}{dt} &= v_q \\ \frac{d\omega}{dt} &= \frac{k_m}{J} i_q - \frac{1}{J} T_L \end{aligned} \quad (7)$$

Considerando a

$$v_d = -k_d (i_d - i_d^*) + \frac{di_d^*}{dt}$$

y

$$v_q = -k_q (i_q - i_q^*) + \frac{di_q^*}{dt},$$

donde k_d y k_q son constantes reales positivas, se tiene:

$$\begin{aligned} \frac{di_d}{dt} - \frac{di_d^*}{dt} + k_d (i_d - i_d^*) &= 0 \\ \frac{di_q}{dt} - \frac{di_q^*}{dt} + k_q (i_q - i_q^*) &= 0 \end{aligned} \quad (8)$$

La expresión (8) se reescribe como:

$$\begin{aligned} \dot{e}_d + k_d e_d &= 0 \\ \dot{e}_q + k_q e_q &= 0 \end{aligned}$$

Se propone entonces la siguiente función de Lyapunov:

$$V = \frac{1}{2} e_d^2 + \frac{1}{2} e_q^2 \geq 0 \quad (9)$$

Derivando la expresión (9) con respecto al tiempo se tiene:

$$\begin{aligned}\dot{V} &= e_q \dot{e}_q + e_d \dot{e}_d = e_q(-k_q e_q) + e_d(-k_d e_d) \\ &= -e_q^2 k_q - e_d^2 k_d\end{aligned}$$

Lo cual implica que $i_d \rightarrow i_d^*$ y $i_q \rightarrow i_q^*$ cuando $t \rightarrow \infty$.

Ahora el objetivo de control se centra en diseñar las trayectorias deseadas i_d^* e i_q^* dada la velocidad deseada del rotor ω^* . Como se observa en (7), i_q es la corriente que actúa directamente sobre ω . Entonces el diseño de i_d debe estar basado en el sistema en estado estable [1], es decir:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{R_s}{L_s} i_d + n_p \omega i_q + \frac{1}{L_s} u_d \\ -\frac{R_s}{L_s} i_q - n_p \omega i_d - \frac{k_m}{L_s} \omega + \frac{1}{L_s} u_q \\ \frac{k_m}{J} i_q - \frac{1}{J} T_L \end{bmatrix} \quad (10)$$

Dado el sistema anterior, se obtienen u_d y u_q :

$$\begin{aligned}u_d &= -R_s i_d + L_s n_p \omega i_q \\ u_q &= -R_s i_q - L_s n_p \omega i_d - k_m \omega\end{aligned}$$

Entonces para obtener i_d se maximiza la siguiente función según [1]:

$$T_L(i_d, i_q) = k_m i_q$$

dada la restricción:

$$u_d^2 + u_q^2 \leq V_{\max}^2$$

Para ello, se usa la técnica de los multiplicadores de Lagrange, donde se debe resolver la siguiente ecuación:

$$\nabla[T_L(i_d, i_q) - \lambda g(i_d, i_q)] = 0$$

donde:

$$\begin{aligned}g(i_d, i_q) &= (-R_s i_d + L_s n_p \omega i_q)^2 \\ &+ (-R_s i_q - L_s n_p \omega i_d - k_m \omega)^2 - V_{\max}^2\end{aligned}$$

entonces se tiene que:

$$\frac{\partial T_L(i_d, i_q)}{\partial i_d} = 0$$

$$\frac{\partial T_L(i_d, i_q)}{\partial i_q} = k_m$$

y

$$\frac{\partial g(i_d, i_q)}{\partial i_d} = 2[-R_s u_d + L_s n_p \omega u_q]$$

$$\frac{\partial g(i_d, i_q)}{\partial i_q} = 2[L_s n_p \omega u_d + R_s u_q]$$

Como sólo nos interesa el diseño de i_d , tomamos la siguiente igualdad:

$$\frac{\partial T_L(i_d, i_q)}{\partial i_d} - \lambda \frac{\partial g(i_d, i_q)}{\partial i_d} = 0$$

lo que nos lleva a:

$$i_d^* = \frac{-k_m n_p L_s (\omega^*)^2}{R_s^2 + (n_p L_s \omega^*)^2}$$

Ahora nos concentraremos en el subsistema mecánico:

$$\frac{d\omega}{dt} = \frac{k_m}{J} i_q - \frac{1}{J} T_L \quad (11)$$

Se desea diseñar i_q para lograr que $\omega \rightarrow \omega^*$ rechazando la perturbación T_L [2]. Supongamos que la carga T_L se puede aproximar localmente a través de un polinomio de grado 4, es decir:

$$T_L = a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

Propongamos a i_q como:

$$i_q = \frac{J}{k_m} \left(\frac{d\omega^*}{dt} - G(s) e_\omega \right) \quad (12)$$

donde:

$$e_\omega = \omega - \omega^*$$

$$G(s) = \frac{k_5s^5+k_4s^4+k_3s^3+k_2s^2+k_1s+k_0}{s^5(s^5+k_{10}s^4+k_9s^3+k_8s^2+k_7s+k_6)}$$

Sustituyendo (12) en (11)

$$\frac{d\omega}{dt} = \frac{d\omega^*}{dt} - \frac{k_5s^5+k_4s^4+\dots+k_0}{s^5(s^5+k_{10}s^4+\dots+k_6)}e_\omega - \frac{1}{J}T_L \quad (13)$$

Multiplicando la expresión anterior por s^5 y simplificando se tiene

$$s^5 \dot{e}_\omega = -\frac{k_5s^5+k_4s^4+\dots+k_0}{(s^5+k_{10}s^4+\dots+k_6)}e_\omega - \frac{1}{J}s^5 T_L \quad (14)$$

Como se observa al multiplicar T_L por s^5 , se esta eliminando el polinomio de cuarto grado que se había supuesto, ya que al multiplicar por s^5 significa que se esta derivando cinco veces en el tiempo la función T_L . Por lo tanto (14) queda como

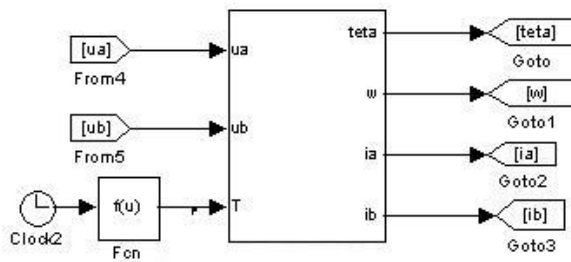
$$s^5 \dot{e}_\omega = -\frac{k_5s^5+k_4s^4+k_3s^3+k_2s^2+k_1s+k_0}{s^5+k_{10}s^4+k_9s^3+k_8s^2+k_7s+k_6}e_\omega \quad (15)$$

Simplificando (15)

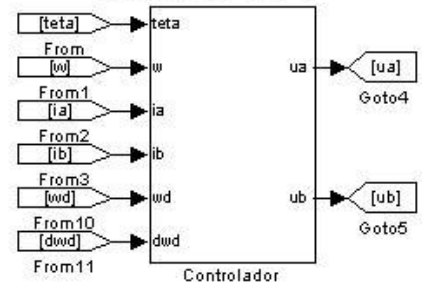
$$(s^{11} + k_{10}s^{10} + k_9s^9 \dots + k_2s^2 + k_1s + k_0)e_\omega = 0$$

donde

Motor Síncrono de Imanes Permanentes



Control GPI



Velocidad Deseada

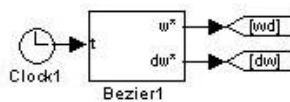


Figura 1. Esquema de Simulación.

$$s^{11} + k_{10}s^{10} + k_9s^9 \dots + k_2s^2 + k_1s + k_0$$

Es el polinomio característico de e_ω , de tal forma que al elegir $k_{10} \dots k_0$ para que todos los polos se encuentren en el semiplano izquierdo complejo, se logre estabilidad asintótica del equilibrio $e_\omega = 0$, lo que nos lleva a concluir que $\omega \rightarrow \omega^*$ cuando $t \rightarrow \infty$.

Así pues se hace la siguiente elección de $k_{10} \dots k_0$ igualando el polinomio característico a una dinámica deseada:

$$s^{11} + k_{10}s^{10} + \dots + k_0 = (s^2 + 2\xi\omega_n s + \omega_n^2)^5 (s + p) \quad (16)$$

donde:

- $k_0 = \omega_n^{10} p$
- $k_1 = \omega_n^{10} + 10\xi\omega_n^9 p$
- $k_2 = 40\xi^2\omega_n^8 p + 10\xi^2\omega_n^9 p + 5\omega_n^8 p$
- $k_3 = 80\xi^3\omega_n^7 p + 5\omega_n^8 p + 40\xi\omega_n^7 p + 40\xi^2\omega_n^8$
- $k_4 = 80\xi^3\omega_n^7 p + 10\omega_n^6 p + 40\xi\omega_n^7 p + 120\xi^2\omega_n^6 p + 80\xi^4\omega_n^6 p$
- $k_5 = 10\omega_n^6 p + 60\xi\omega_n^5 p + 160\xi^3\omega_n^5 p + 80\xi^4\omega_n^6 p + 120\xi^2\omega_n^6 p + 32\xi^5\omega_n^5 p$
- $k_6 = 10\omega_n^4 p + 120\xi^2\omega_n^4 p + 160\xi^3\omega_n^5 p + 80\xi^4\omega_n^4 p + 32\xi^5\omega_n^5 p + 60\xi\omega_n^5$
- $k_7 = 120\omega_n^4 \xi^2 + 80\xi^3\omega_n^3 p + 80\xi^4\omega_n^4 p + 10\omega_n^4 p + 40\xi\omega_n^3 p$
- $k_8 = 80\omega_n^3 \xi^3 + 40\xi^2\omega_n^2 p + 40\xi\omega_n^3 p + 5\omega_n^2 p$
- $k_9 = 10\omega_n \xi p + 40\xi^2\omega_n^2 p + 5\omega_n^2$
- $k_{10} = p + 10\xi\omega_n k_0$

IV. RESULTADOS EN SIMULACIÓN

Los parámetros del MS de IP que se utilizó para simular una prueba del control fueron tomados de [1]:

$$\begin{aligned} L_s &= 1.5 \times 10^{-3} \text{ H} \\ R_s &= 0.55 \text{ } \Omega \\ n_p &= 50 \\ k_m &= 0.19 \text{ N.m/A} \\ J &= 4.5 \times 10^{-5} \text{ kg.m}^2 \end{aligned}$$

La Figura 1 muestra el esquema de simulación que se llevó a cabo utilizando MatLab y Simulink (MatLab y Simulink son marcas registradas de The MathWorks Inc.).

En la Figura 2 aparece la onda del par propuesto: $T_L = 0.2 \sin(t)$. Este par es desconocido para el control y se entiende como una perturbación que se espera sea “absorbida” por el control GPI. Es importante señalar que a pesar de que la función seno no tiene una representación finita polinómica, la aproximación mediante un polinomio de cuarto orden bastó para que el control rechazara el par.

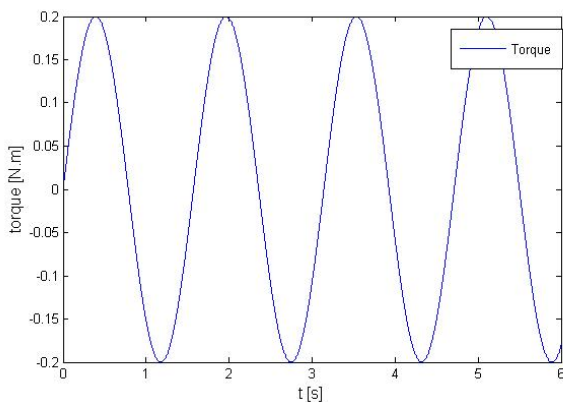


Figura 2. Par aplicado al motor síncrono.

En la Figura 3 se muestra el seguimiento de la velocidad ω^* (deseada) y la trayectoria de la velocidad real ω . Se debe mencionar que la referencia de la velocidad debe corresponder a una trayectoria derivable al menos dos veces, es decir, debe ser suave, esto debido a que en la ley de

control se requiere su derivada. Las trayectorias propuestas de la velocidad del rotor no deben tener discontinuidades, es por esta razón, que se usa una trayectoria C^∞ , es decir, un polinomio de Bezier.

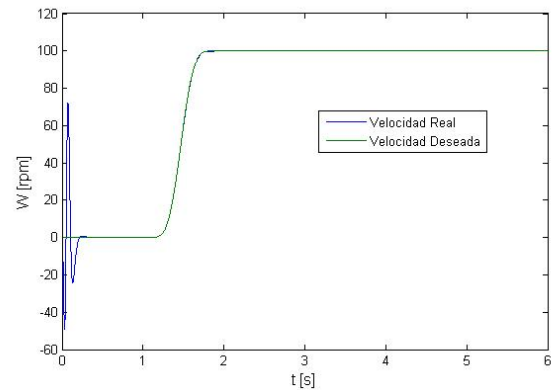


Figura 3. Seguimiento de velocidad.

En los primeros instantes del seguimiento de la Figura 3 se puede observar un sobreimpulso que surge por el inicio de la variación del par en forma senoidal, que tiene una amplitud de 0.2 Nm. De esta manera el motor debe permanecer en $\omega=0$ a pesar del desconocimiento del par. La velocidad tiene una trayectoria programada de forma que en el instante correspondiente a dos segundos ya debe de estar en 100 rpm. Esto sin importar que el par siga variando.

En la Figura 4 se muestran las entradas de control transformadas a un sistema trifásico, es decir, el control nos ofrece dos entradas, que corresponden al modelo bifásico del MS de IM y se mapean como voltajes trifásicos a través de la transformación de Clarke [6].

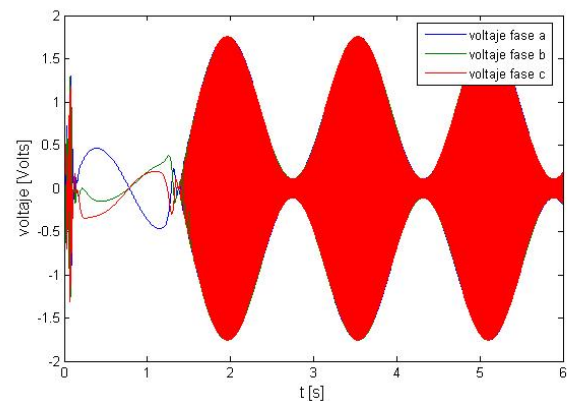


Figura 4. Voltajes de control.

V. CONCLUSIONES

En el presente artículo se muestra el desempeño del control GPI para rechazar perturbaciones de par en un MS de IP. Se utiliza el diseño del control por campo orientado, reportado en [1], para lograr que en el sistema transformado (d-q) la corriente del eje directo esté desacoplada de la velocidad, o dicho de otra forma, para lograr que la corriente de cuadratura sea la única entrada de control para la velocidad. Cabe mencionar que el sobreimpulso en la velocidad que aparece en los primeros instantes de la simulación puede ser corregido usando otra forma del control GPI, que modifique la ubicación de los polos.

Como trabajo futuro se pretende estudiar el problema de la sintonización del GPI, ya que no es evidente como deben ser las ganancias ξ , ω_n y p , para lograr un mejor desempeño en el MS de IP. Finalmente debe estudiarse la posibilidad de utilizar un observador para reconocer el par desconocido y poder mejorar la respuesta del control, de forma que minimice los sobreimpulsos.

VI. REFERENCIAS

- [1] John Chiasson, *Modeling and High-Performance Control of Electric Machines*, IEEE Press Series on Power Engineering, 2005
- [2] Vladan Petrovic, Romeo Ortega, and Aleksandar M. Stankovic, "Interconnection and Damping Assignment Approach to Control Synchronous Motors of PM", IEEE Transactions on Control Systems Technology, Vol. 9, No. 6, November 2001.
- [3] Vladan. Petrovic, R. Ortega, A. M. StankoviC, G. Tadmor "An Adaptive Controller for Minimization of Torque Ripple in PM Synchronous Motors", Proceedig of the 38th Conference on Decision & Control, December 1999
- [4] J. Becedas, V. Feliu and H. Sira-Ramirez, "GPI Control for a Singel-Link Flexible Manipulator", International Conference on Modeling, Simulation and Control. San Franciso, EUA, pp 24-25, October 2007
- [5] Paul C. Krause, O. Wasynczuk and S.D Sudhoff, *Analysis of Electric Machinery and Drive Systems*, Wiley Inter-Science, 2002.
- [6] Leonhard, W., Control of electrical drives, Springer-Verlag, Berlin, 1985

CARGADOR DE BATERÍAS PARA APLICACIONES FOTOVOLTAICAS CON SEGUIDOR DE LA CORRIENTE MÁXIMA DE CARGA

G. Tapia Cecilia, López Adolfo, Rodríguez Elías, Hurtado José Luis, Hernández Benjamín, Vázquez Nimrod.

Instituto Tecnológico de Celaya, Departamento de Electrónica
Av. Tecnológico y A. García Cubas S/N, Col. Alfredo Bonfil, Celaya, Guanajuato, C.P. 38010
Tel. (461) 611 75 75, correo electrónico: elias@itc.mx

Resumen—Este trabajo presenta un nuevo algoritmo de control de carga de baterías para aplicaciones en sistemas fotovoltaicos autónomos. Se diseñó un cargador de baterías basado en un convertidor reductor síncrono para comprobar el algoritmo de control. Adicionalmente se diseñó un sistema de adquisición de datos para presentar, almacenar y analizar el comportamiento del sistema fotovoltaico. Se presentan también los resultados experimentales del algoritmo implementado. Palabras clave: cargador, seguidor, convertidor reductor, corriente de carga, SIADA.

1. INTRODUCCION

Las energías renovables son la mejor opción para llevar energía eléctrica a zonas rurales o de difícil acceso. Además, ahora que temas como la contaminación del aire y el calentamiento global se han vuelto comunes y que las reservas de los recursos no renovables como lo es el petróleo se están agotando, se ha promovido el uso de energías limpias como lo son: solar, eólica, biomasa, etc. con el objeto de mitigar dichos problemas ambientales. Por lo anterior, los sistemas fotovoltaicos son una alternativa prometedora para la generación de energía eléctrica ya que ésta proviene de un recurso poderoso e inagotable como lo es el sol por lo que es gratis, libre de contaminantes y además, esta distribuida por todo el planeta.

Existen dos variantes para estos sistemas que son los sistemas fotovoltaicos conectados a la red y los sistemas fotovoltaicos autónomos los cuales almacenan la energía producida en un banco de baterías para después ser distribuida, y es precisamente a estos sistemas a los que va dirigido este trabajo. Un elemento importante dentro de los

sistemas fotovoltaicos son las baterías las cuales determinan el tiempo de respaldo del sistema, por lo que la correcta carga y descarga de estas es de vital importancia considerando que éstas representan una parte significativa en el costo total del sistema. Una característica importante del cargador de baterías es su eficiencia, ya que como se sabe la eficiencia total del sistema es el producto de las eficiencias individuales de cada etapa del sistema, por lo que el cargador debe presentar una alta eficiencia para aprovechar al máximo la energía solar.

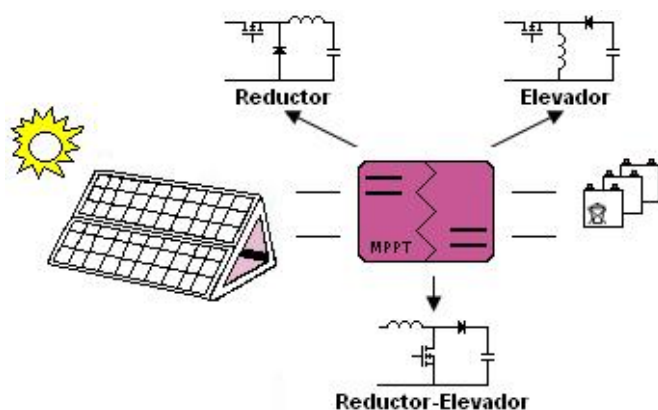
La corriente de salida del cargador (I_B) debe ser una fracción de la corriente nominal que se requiere para cargar completamente a la batería en una hora, a esta corriente se le llama capacidad nominal de la batería (C). Otro aspecto a considerar es el tiempo de sol disponible para cargar la batería, ya que esto es un factor determinante en la elección del tipo de carga que se va a realizar. Existen varios métodos para cargar una batería dependiendo del tiempo de carga con el que se cuente, y se clasifican en: carga lenta, carga rápida y carga pulsante. La diferencia principal entre cada método es el tiempo requerido para cargar a la batería, a mayor razón de carga la batería se carga en un menor tiempo, todo depende de la aplicación. En la tabla 1 se muestra la razón de carga y el tiempo de carga de cada uno de los métodos de carga antes mencionados [1].

Para la implementación del cargador de baterías se pueden utilizar diferentes topologías, las tres principales son: reductor, elevador y reductor-elevador. La elección depende de las características de voltaje de las celdas fotovoltaicas y del arreglo de baterías, como se muestra en la figura 1.

Tabla 1 Características de los métodos de carga de las baterías.

Método de Carga	Razón de Carga [Amperes]	Tiempo de Carga [Horas]
Pulsante	0.01 – 0.03 C	100 - 33
Lenta	0.05 – 0.1 C	20 - 10
Rápida	0.2 – 0.5 C	5 - 2
Ultra rápida	1 C o más	1 o menos

El voltaje y la corriente de las celdas fotovoltaicas presentan un comportamiento no lineal, el cual se ve afectado por diferentes factores como la temperatura de las celdas fotovoltaicas y el nivel de insolación, por lo que es muy complejo localizar el punto de máxima potencia de las celdas fotovoltaicas. Para localizar el punto de máxima potencia surgieron algoritmos de control llamados MPPT (Maximum Power Point Tracking) cuya función es mover las condiciones eléctricas de operación de la celda fotovoltaica para que ésta sea capaz de entregar una potencia constante.

**Figura 1 Diagrama a bloques del cargador de baterías.**

2. SEGUIDORES DEL PUNTO DE MAXIMA POTENCIA

Se han desarrollado e implementado varios métodos para seguir el punto de máxima potencia de las celdas fotovoltaicas, algunos más sofisticados que otros. Cada método varía en complejidad, sensores requeridos, velocidad de respuesta, confiabilidad, costo, etc. Algunos de los métodos son: perturbar y observar, incremento de la conductancia, además, hay algunos más sofisticados donde se utiliza control difuso ó redes

neuronales [2-6] pero todo depende de la complejidad que se desee. La elección del método también depende de la aplicación, en el caso de sistemas fotovoltaicos autónomos dirigidos a aplicaciones rurales, el arreglo fotovoltaico se instala en áreas despejadas y al no haber edificios u obstáculos que reduzcan los niveles de irradiación de las celdas fotovoltaicas no se necesita de sistemas muy complejos.

Debido a su fácil implementación y a su bajo costo que los métodos perturbar y observar e incremento de la conductancia son los más empleados para estas aplicaciones. Existen dos variantes principales del algoritmo observar y perturbar, en la primera se mide la variación de potencia y la variación de voltaje para determinar el cambio del punto de máxima potencia [2]. Si el signo de $\Delta P/\Delta V$ es positivo, se incrementa la referencia de voltaje y viceversa como se muestra en la figura 2 y con eso se aumenta la corriente de salida del cargador de baterías. En el otro caso se mide la variación de corriente para encontrar el punto de máxima potencia [3], de la misma forma si el signo de $\Delta P/\Delta I$ es positivo se incrementa la referencia de voltaje y viceversa. Pero con el objetivo de eliminar las características negativas de este método es que surgió el método del incremento de la conductancia, en el cual se checa el signo de $\Delta I/\Delta V$ [4]. Después de esto han surgido demasiadas variantes de estos métodos con los cuales se tratan de mejora aspectos como: el tiempo de respuesta, precisión, reducir las pérdidas, entre otros. La principal ventaja de estos métodos son su fácil implementación y la simplicidad de su estructura de control, aunque presentan la desventaja de no ser capaz de responder a cambios bruscos en las variables. Estos algoritmos de control se concentran en obtener de las celdas fotovoltaicas la máxima potencia y entregarle a la batería la mayor cantidad de energía (corriente) en el menor tiempo. Dado lo anterior han surgido una infinidad de variantes de estos algoritmos en los cuales se busca llegar al punto de máxima potencia lo más pronto posible, que el tiempo de respuesta ante cambios atmosféricos sea casi instantáneo y que sean más precisos, pero en

la aplicación en cuestión estas características no son imprescindibles.

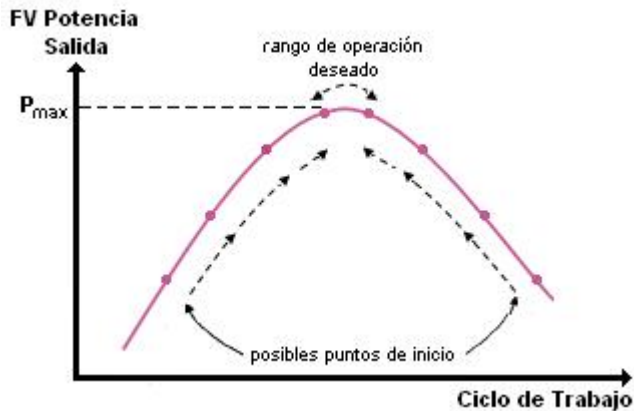


Figura 2 Funcionamiento del algoritmo perturbar y observar.

Pero también es bien sabido que si las baterías son cargadas a razones de carga muy grandes, las baterías aparentan haber alcanzado su nivel máximo de carga pero después de un tiempo en el que el proceso químico de las baterías se estabiliza el nivel de carga disminuye y alcanza su valor real. Además, al aplicar razones de carga muy grandes a las baterías se disminuye el tiempo de vida promedio de estas y se corre el riesgo dañarlas definitivamente. Otro aspecto a considerar es que cuando las baterías se encuentran a punto de alcanzar su carga máxima, se debe disminuir la razón de carga para evitar causarle daños irreversibles a las baterías. Es por eso que surgió la idea de un algoritmo de control que se concentre en la correcta carga de las baterías y no en maximizar la potencia demandada a las celdas fotovoltaicas, al cual hemos llamado seguidor de la corriente máxima de carga o MCCT (maximum charge current tracking)

3. SEGUIDOR DE LA CORRIENTE MÁXIMA DE CARGA

3.1 Cargador de Baterías

La elección de la topología surgió de las condiciones de entrada y salida del cargador las cuales son: $V_{FV} = 22 - 40$ volts, $V_B = 12$ volts, $I_{Bmax} = 3$ A, por lo que se requiere de un convertidor reductor. En la búsqueda de mejorar la eficiencia del convertidor se utilizó rectificación síncrona como se muestra en la figura 3.

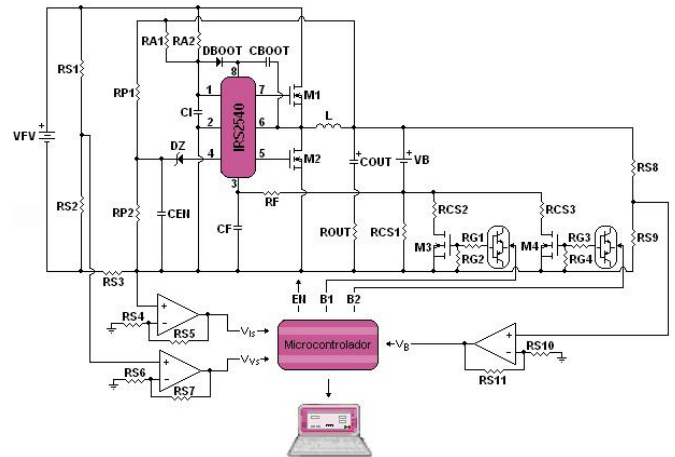


Figura 3 Diagrama esquemático del cargador de batería seguidor de la máxima corriente de carga.

3.2 Etapa de Control

Buscando un control fácil de implementar pero a la vez eficaz, se llegó al circuito integrado IRS2540 [8] el cual es un regulador de corriente. El circuito integrado emplea un método de control de libre corrimiento de frecuencia por lo que no se requiere de componentes externos para fijar la frecuencia de operación, es fácil implementar la autoalimentación (requiere una corriente de arranque menor a 500 uA), presenta reencendido automático y una terminal para desactivarlo. Su operación es muy sencilla, una vez que el voltaje de alimentación del circuito integrado alcanza el voltaje de umbral la salida HO del CI (pin 7) se pone en alto por un periodo de tiempo predeterminado lo que inicia un proceso de carga del capacitor de bootstrap el cual determina la fuente flotada que opera como referencia de la salida LO (pin 5), mientras que la salida HO se encuentra en alto la salida LO se mantiene en bajo. Bajo condiciones normales de operación la corriente de salida se regula retroalimentando al CI la muestra de corriente tomada a través de R_{CS} . Esta muestra de corriente de salida se pasa por un filtro antes de llegar a la terminal IFB (pin 3), este voltaje retroalimentado se compara con un voltaje de referencia interno (V_{IFBTH}). Mientras que V_{IFB} sea menor que V_{IFBTH} la salida HO se mantendrá en alto transfiriendo una corriente de V_{FV} a la carga y al mismo tiempo se almacena energía en el

inductor, esto provoca un incremento en V_{IFB} y en el momento que V_{IFB} supere a V_{IFBTH} el lazo de control pone la salida HO en bajo y LO en alto liberando la energía almacenada en el inductor hacia la carga y el valor de V_{IFB} comienza a disminuir. Lo anterior provoca que la salida HO se vuelva a poner en alto y LO en bajo volviéndose a repetir el proceso, y así continua para obtener la corriente promedio deseada. La relación entre la corriente deseada y el V_{IFBTH} y la resistencia de sentido es:

$$I_B = \frac{V_{IFBTH}}{R_{CS}} \quad (1)$$

donde: $V_{IFBTH} = 0.5$ volts, esta relación se cumple siempre que el rizo de corriente en I_{out} y I_{FB} sean pequeñas. Para evitar la condición de corto circuito entre ambos interruptores de forma implícita las señales de control presentan un tiempo muerto de aproximadamente 140 ns, además, de tiempos de retardo tanto en el encendido como en el apagado de las señales de salida.

3.3 Selección del Inductor y Capacitor de Salida

Como ya se mencionó el circuito integrado no requiere de elementos externos para fijar la frecuencia de operación, por lo que utiliza las características de entrada y salida del convertidor para lograrlo. Pero además, los valores del inductor y del capacitor de salida causan un gran efecto en el ajuste de la frecuencia de operación, por lo que es importante hacer una buena selección de los mismos ya que a mayor frecuencia de operación las pérdidas por conmutación también son mayores. El comportamiento de la frecuencia de operación reportado en las hojas de especificaciones del circuito integrado ante cambios en los valores de la inductancia y del capacitor de salida, nos dice que entre mayor sea la inductancia menor será la frecuencia de operación, de la misma forma entre mayor sea la capacitancia menor será la frecuencia de operación con la diferencia de que hay un punto en el que aunque se aumente la capacitancia el efecto será mínimo. Es por eso que considerando los datos reportados y pruebas experimentales se eligieron los valores de $L = 1\text{mH}$ y $C_{out} = 33 \text{ uF}$ para obtener una

frecuencia de operación de aproximadamente 50KHz y una corriente máxima de salida de 3 A.

Para el diseño del inductor nos basamos en el método reportado en [9], donde primero hay que determinar el tamaño mínimo del núcleo por medio del producto de áreas:

$$A_e A_c = \frac{(5.067 * L * I_L * D^2)}{K * B_{max}} \approx 1.16 \text{ cm}^4 \quad (2)$$

donde:

A_e = área efectiva del núcleo,
 A_c = área de ventana,
 D = diámetro del conductor,
 $K = 0.8$,
 $B_{max} = 2500 \text{ G}$.

Del catálogo de ferritas el núcleo RM12 tiene un $A_e = 1.46 \text{ cm}^2$ y $A_c = 0.774 \text{ cm}^2$, lo que nos da un producto de áreas de 1.13 cm^4 por lo que es posible utilizar este núcleo. La longitud del gap requerido se obtiene de:

$$l_g = \frac{(0.4 * \pi * L * I_L^2 * 10^8)}{A_e * B_{max}^2} \approx 0.1239 \text{ cm} \quad (3)$$

y el número aproximado de vueltas se obtiene de:

$$N = \frac{B_{max} * l_g}{(0.4 * \pi * I_L)} \approx 82 \text{ vueltas} \quad (4)$$

3.4 Sistema de Adquisición de Datos (SIADA)

Este sistema consta de una etapa de sentido y una etapa de adquisición, en la figura 4 se muestra el diagrama esquemático de este sistema. Para la implementación se empleo un microcontrolador PIC18F258 de Microchip, el cual cuenta con un módulo convertidor analógico digital de 10 bits multicanal y tecnología Ecan la cual permite comunicación entre la computadora y el cargador de baterías. Primeramente se sensa el voltaje y la corriente que entregan las celdas fotovoltaicas ($V_{FV} - I_{FV}$) y el voltaje de la batería (I_B), esta información llega al microcontrolador a través de los canales AN0, AN1 y AN2.

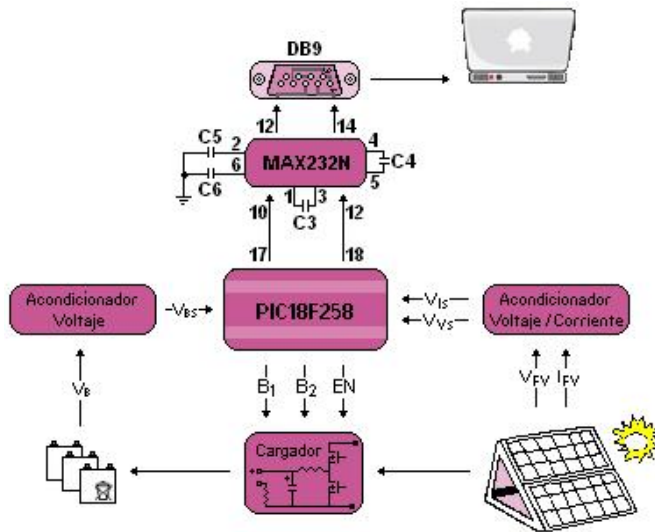


Figura 4 Diagrama esquemático del sistema de adquisición de datos (SIADA).

Los datos obtenidos son señales analógicas por lo que se emplea el convertidor analógico digital interno del microcontrolador para convertir las señales analógicas a señales digitales. La conversión de cada dato se realiza una a la vez ya que el tiempo mínimo de muestreo es de 1 minuto por lo que no se requiere de un convertido A/D para cada señal. Los datos digitales funcionan como variables del algoritmo de control. Además, se requiere comunicar el microcontrolador con una computadora para lo cual se emplea el circuito integrado MAX232N el cual convierte las señales digitales al protocolo de comunicación RS-232 (comunicación serial), en la figura 4 se muestra el diagrama esquemático de la comunicación entre el microcontrolador y la computadora.

La herramienta diseñada para la adquisición de datos se realizó con ayuda de Visual Basic mediante objetos DAO (Data Access Object). El objeto DAO es una compilación de DLL's los cuales no son más que un componente de Microsoft Access que podemos usar al igual que lo hace Access para generar una base de datos [10]. En el instante que el dato de voltaje y corriente de las celdas fotovoltaicas se encuentra en la computadora se obtiene la potencia entregada por las celdas fotovoltaicas, y después se guarda la información de voltaje, corriente y potencia de las celdas fotovoltaicas en la respectiva base de datos

previamente creada. El rango de muestreo se puede seleccionar desde minutos hasta horas y después de cada muestreo se despliegan los valores actuales en la ventana de control del sistema de adquisición, la cual se muestra en la figura 5. La computadora despliega la información almacenada del voltaje, corriente y potencia de las celdas fotovoltaicas en forma de gráficas y de tablas de datos.



Figura 5 Ventana de control del sistema de adquisición de datos (SIADA).

3.5 Funcionamiento del Cambio de Carga

Debido a que el circuito integrado IRS2540 fija la corriente de salida dependiendo del valor de la resistencia de sensado R_{CS} , si variamos la resistencia de sensado podemos obtener diferentes corrientes de carga para la batería. Como la corriente a través de la resistencia de sensado es de por lo menos 1A, se decidió que los interruptores fueran mosfet's. Para esta aplicación se pretende tener cuatro diferentes corrientes de carga: 0.5A, 1A, 2A y 3A, en la figura 6 se muestra un diagrama esquemático del circuito de cambio de carga, los mosfet's son controlados por la salida B de dos bits del microcontrolador cuyo algoritmo de control selecciona que mosfets deben estar encendidos. Cada combinación de la salida B corresponde a una de las posibles corrientes de salida del cargador de baterías, como se muestra en la tabla 2.

3.6 Algoritmo de Control

El objetivo del algoritmo de control es determinar cual es la corriente a la cual debe ajustarse el cargador de baterías, este algoritmo consta de tres etapas principales y su diagrama de flujo se muestra en la figura 7.

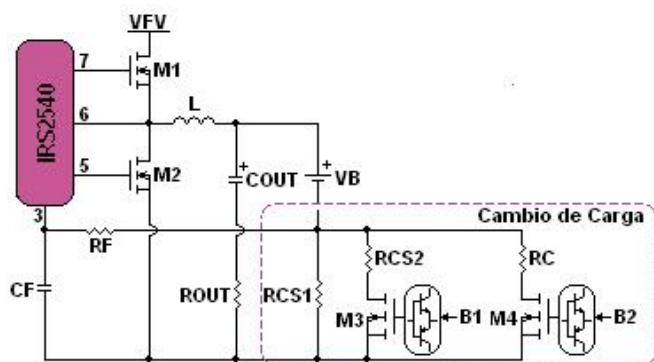


Figura 6 Diagrama esquemático del circuito de cambio de carga.

Para la implementación se utilizó el mismo microcontrolador PIC18F258 de la etapa de adquisición de datos. En la primera etapa se realiza el sensado de las señales de voltaje y corriente de las celdas ($V_{FV} - I_{FV}$). Se utiliza el convertidor analógico digital del microcontrolador para realizar primero la conversión de la señal de voltaje correspondiente al voltaje de la celda, la conversión de la señal de voltaje correspondiente a la corriente de la celda y por último la conversión de la señal del voltaje de la batería, una a la vez.

Tabla 2 Combinaciones de la salida B del microcontrolador.

Salida B		I_B	Espacio Memoria
B ₁	B ₂		
0	0	0.5 A	P0.5A
0	1	1 A	P1A
1	0	2 A	P2A
1	1	3 A	P3A

Después comprueba que el voltaje de la celda sea mayor a un voltaje preestablecido por lo que en caso de ser menor desactiva al cargador hasta que se alcance el nivel de voltaje de celda mínimo, y por último obtiene la potencia instantánea de la celda fotovoltaica. Debido a la aplicación de este sistema no se requiere de una respuesta rápida, por lo que el muestreo mínimo es de 1 minuto, lo cual permite tomar un número fijo de muestras tanto para el voltaje como para la corriente de la celda y promediarlas y de esta forma obtener un valor más cercano a la realidad. La segunda etapa se encarga

de guardar en un espacio de memoria determinado la potencia instantánea (P_i) para cada combinación de la salida B comenzando con la combinación menos significativa como se muestra en la tabla 2 y terminando con la más significativa, esto ocurre en dicho orden por la naturaleza propia del programa.

El algoritmo guarda un dato a la vez y al finalizar se pone en alto una bandera que indica que no se debe volver a entrar a esa rutina a menos que se reinicie el programa. Como se sabe el comportamiento de las celdas depende fuertemente de la estación del año por lo que si fijamos los límites de potencia mínima para cada corriente demanda a las celdas, puede que para la siguiente estación difieran demasiado, así que con este método dichos límites se generan de forma automática cada vez que se alcanza el voltaje mínimo para arrancar al integrado. En la tercera etapa se realiza una comparación entre la potencia instantánea y la potencia anterior (P_a) la cual se inicializa en cero, si el resultado indica que $P_i \geq P_a$ el valor de la salida B debe aumentarse en un bit, esto ocurre por lo menos en las cuatro primeras corridas del programa, después depende si se puede cumplir con la potencia mínima requerida para P3A ya que si no fuese así se debe tomar sobre la salida B la acción correspondiente a la potencia que se pueda satisfacer. Si se alcanza el valor máximo para la salida B y la potencia de la celda sigue aumentando, el algoritmo mantiene la salida B con el mismo valor.

Caso contrario, si el resultado de la comparación indica que $P_i < P_a$ hay dos posibles razones, una es que se haya presentado una disminución de la potencia de la celda pero que dicha potencia siga dentro del rango de P3A y entonces no debe haber cambio en la salida B, o por que se haya presentado una disminución en la potencia de la celda y la potencia instantánea se encuentre fuera del rango de P3A lo cual obliga al sistema a buscar el rango de potencias en el que se encuentra y dependiendo de eso toma un valor para la salida B. Por otro lado, para evitar un descarga profunda de las baterías o un exceso de carga el nivel de carga de las baterías siempre se está monitoreando, por

lo que si el microcontrolador detecta que se esta a punto de llegar a la carga máxima de la batería manda a desactivar el cargador y de igual forma si ve que el nivel de carga de la batería esta por abajo del deseado enciende el cargador de baterías.

corriente obtenidas del monitoreo de las celdas fotovoltaicas por un periodo de aproximadamente 2 horas.

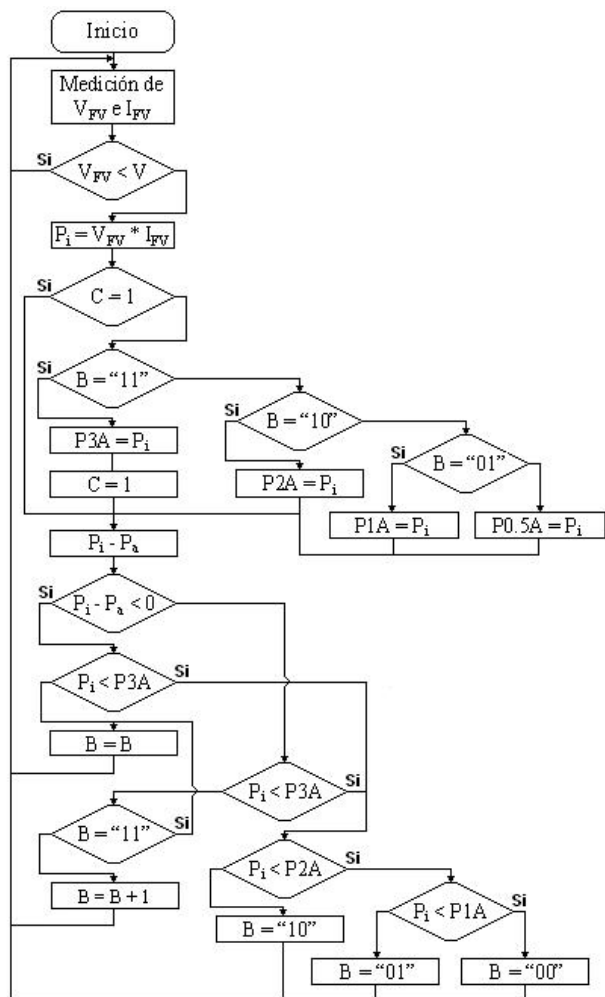


Figura 7 Diagrama de flujo del algoritmo de control del seguidor de la corriente máxima de carga.

4. RESULTADOS EXPERIMENTALES

Se implemento un prototipo experimental de 100 watts con las siguientes características: $V_{entrada} = 22-40$ volts, $V_{salida} = 12$ volts, $I_{m\acute{a}x} = 3$ amper, $L = 1$ mH, $C_{out} = 33$ uF. La fuente primaria de alimentación es un arreglo fotovoltaico que consiste de dos módulos fotovoltaicos de 100 watts conectados en serie y el respaldo es una batería plomo-ácido de 12 volts 24 Ah. En las figuras 8 y 9 se pueden observar las gráficas de voltaje y



Figura 8 Gráfica de voltaje obtenida del sistema de adquisición de datos (SIADA).

Estas dos graficas representan al mismo tiempo el voltaje y la corriente de entrada del cargador de baterías el cual estaba ajustado para entregar a la batería una corriente de carga de 1.2A.

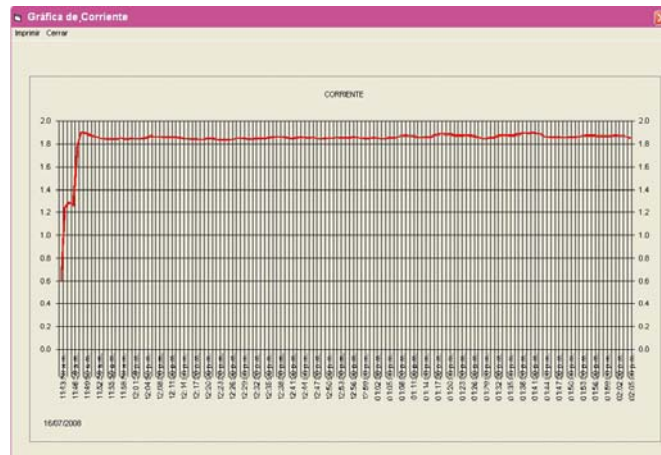


Figura 9 Gráfica de corriente obtenida del sistema de adquisición de datos (SIADA).

En la figura 10 se muestran las formas de onda de corriente de entrada, voltaje y corriente de salida, además de la frecuencia de operación del cargador de baterías, esto para cuando se demanda la máxima corriente de carga. Hay que hacer notar que la eficiencia promedio resultante de muestras obtenidas bajo las cuatro diferentes condiciones de carga es del 92%.

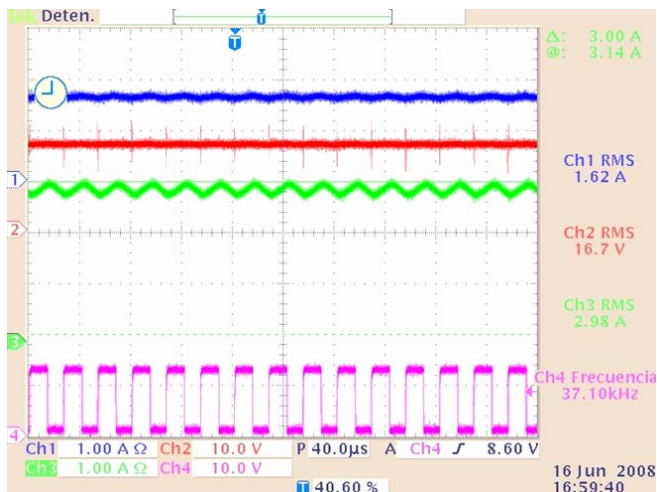


Figura 10 Formas de onda: (a) Voltaje de entrada, (b) Corriente de salida, (c) Voltaje de salida, (d) Corriente de salida.

En la figura 11 se muestra el prototipo experimental del cargador de baterías, el prototipo consiste de acondicionador de la señal de entrada, acondicionador del voltaje de batería, cargador de batería y el sistema de adquisición de datos.

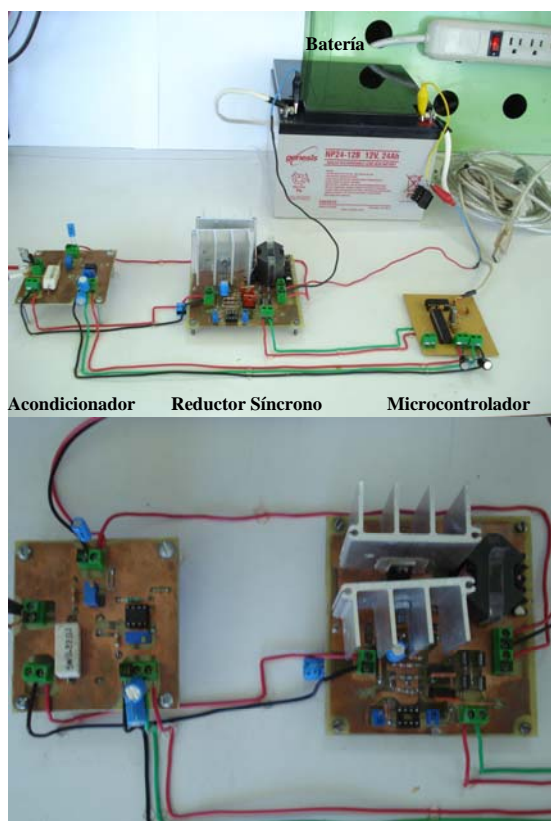


Figura 11 Prototipo experimental del cargador de baterías.

5. CONCLUSIONES

Se implementó un nuevo algoritmo para aplicaciones fotovoltaicas, el cual ajusta la corriente de carga de las baterías dependiendo de la capacidad de las celdas fotovoltaicas para alimentar al cargador de baterías con el voltaje mínimo para cada corriente demandada. Este cargador cuenta con cuatro corrientes de carga que son: 0.5A, 1A, 2A y 3A, las cuales se ajustan dependiendo de las decisiones que se tomen en el algoritmo de control, y también cuenta con una función de paro para que en el momento que la batería este por alcanzar su nivel de máxima carga se desactive. Para la implementación del algoritmo de control se diseñó un convertidor reductor síncrono para maximizar la eficiencia del sistema fotovoltaico. Además, se desarrolló un sistema de adquisición de datos que nos permite monitorear el comportamiento de las celdas fotovoltaicas y observar el funcionamiento de los cambios de carga del cargador de baterías. El algoritmo desarrollado pretende ser una alternativa para maximizar el tiempo de vida útil de las baterías.

6. REFERENCIAS

- [1] *New Technology Batteries Guide* [Online]. Disponible: <http://www.ncjrs.gov/pdffiles/172868.pdf>
- [2] H. Dong, H. Sugimoto, and N. Nishio, "A maximum power tracking control method for photovoltaic power generation system based on derivation of output power with respect to output voltage" *Trans. Inst. Electr. Eng. Jpn.*, vol. 118-D, no. 12, pp. 1435–1442, 1998.
- [3] T. Kitano, M. Matsui, and D. Xu, "A maximum power point tracking control scheme for PV system based on power equilibrium and its system design" *Trans. Inst. Electr. Eng. Jpn.*, vol. 121-D, no. 12, pp. 1263–1269, 2001.
- [4] Y.-C. Kuo, T.-J. Liang, and J.-F. Chen, "Novel maximum-power-pointtracking controller for photovoltaic energy conversion system" *IEEE Trans. Ind. Electron.*, vol. 48, no. 3, pp. 594–601, 2001.
- [5] Jung-Min Kwon, Kwang-Hee Nam and Bong-Hwan Kwon, "Photovoltaic Power Conditioning System With Line Connection" *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp.1048-1054, 2006.

- [6] Weidong Xiao, Magnus G. J. Lind, William G. Dunford and Antoine Capel, “*Real-Time Identification of Optimal Operating Points in Photovoltaic Power Systems*” IEEE Trans. Ind. Electron., vol. 53, no. 4, pp.1017-1026, 2006.
- [7] *Sealed Lead-Acid Batteries, Technical handbook* [Online]. Disponible: http://cardi.igeofcu.unam.mx/techdocs/PowerSonic_batteries.pdf
- [8] “*Datasheet LED Buck Regulador Control IC*”, Internacional Rectifier.
- [9] Chrysis George C. “*High Frequency Switching Power Supplies: Theory and Design*”, Segunda Edición.
- [10] García de Jalon Javier “*Practique Visual Basic 6.0 como si estuviera en primero*” Universidad de Navarra 1999.

CÁLCULO DEL RENDIMIENTO DE LA TÉCNICA DE CONTROL POR ÁNGULO SIMÉTRICO PARA UN RECTIFICADOR CONTROLADO MONOFÁSICO

Lerma García José Ángel, Gómez González Francisco Javier, Méndez Ancona Javier

Departamento de Ingeniería Eléctrica y Electrónica

Instituto Tecnológico de Veracruz

Calzada. Miguel Ángel de Quevedo No. 2779, Col. Formando Hogar, Veracruz Ver. 91860, México

Tel. 229-9341500, correo electrónico: lerma39@hotmail.com

RESUMEN

En este artículo se presenta la implementación de la técnica de control por ángulo simétrico de un rectificador controlado monofásico, para llevar a cabo el cálculo de los parámetros de rendimiento, como son la eficiencia, el valor efectivo (rms) de la componente de CA, el factor de forma y el factor de rizo. Se presenta un análisis comparativo del comportamiento de los parámetros mencionados para diferentes ángulos de conducción, con la intención de extraer conclusiones.

Palabras clave: Rectificador controlado monofásico, control por ángulo simétrico, parámetros de rendimiento, semiconvertidor y convertidor completo.

I. INTRODUCCIÓN

En la electrónica hay muchas aplicaciones que requieren de una fuente de potencia de CD, misma que se puede obtener de la red de alimentación monofásica de CFE, mediante circuitos encargados de convertir la energía de CA a CD, conocidos como rectificadores. Una de las subclasificaciones de los rectificadores son los llamados rectificadores controlados, en los cuales se pueden sustituir parcial o totalmente los diodos rectificadores por dispositivos controlados, como los rectificadores controlados de silicio (SCRs). La ventaja de estos sistemas es que permiten la regulación del voltaje de salida entregado a la carga.

Este tipo de rectificadores se usa en forma extensa en aplicaciones industriales, en especial en propulsores de velocidad variable, desde potencia fraccionaria hasta niveles de megawatts [1].

La técnica tradicional para el control de los rectificadores controlados con SCRs es la de control por ángulo de fase, en la cual la potencia entregada a la carga se regula variando el ángulo de disparo de los tiristores.

En la técnica de control por ángulo simétrico se controla el voltaje de salida, haciendo variar el ángulo de conducción β [1].

En la **Figura 1** se muestran las formas de onda de entrada y salida para un rectificador controlado monofásico, así como los pulsos de control de ángulo simétrico sincronizados para cada fase.

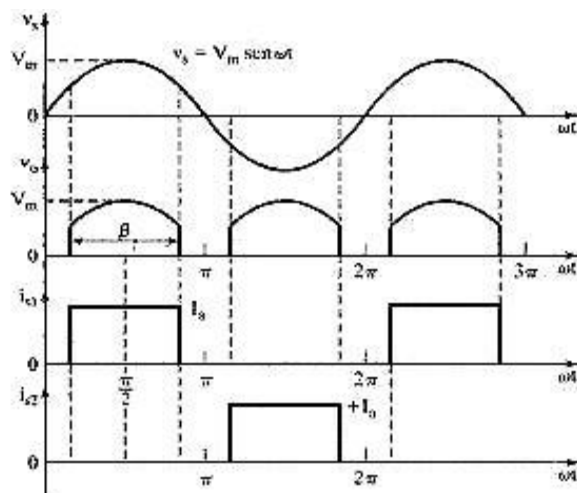


Fig.1 Formas de onda para un rectificador controlado monofásico

En la **Figura 2** se pueden observar las formas de onda de los pulsos de salida de control por ángulo simétrico. Se muestra una señal triangular sincronizada con los cruces de la señal senoidal de alimentación del rectificador controlado, la cual se compara con un nivel de voltaje de referencia para generar el ángulo de conducción β que controlará la magnitud del voltaje de salida.

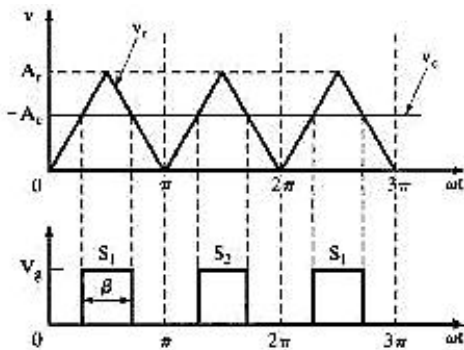


Fig.2 Formas de onda de la técnica de control por ángulo simétrico

A partir de este principio se ha podido diseñar el circuito correspondiente para generar las señales de control.

Se han presentado trabajos relacionados con las técnicas de modulación por ancho de pulso, sin embargo no presentan el uso de éstas en la estimación de los parámetros de rendimiento de un rectificador controlado [2, 3 y 4].

II. MÉTODO

Circuito de Control

En la **Figura 3** se muestra el diagrama a bloques de la técnica de control por ángulo simétrico, estructurada en varios módulos, de los cuales se explica el funcionamiento y se muestra el diagrama esquemático correspondiente.

La mayoría de los módulos están implementados con circuitos electrónicos (analógicos y digitales), entre ellos compuertas TTL, optoacopladores y amplificadores operacionales dispuestos en diversas configuraciones que permiten obtener las señales requeridas [5].

Un amplificador operacional u opamp, es un amplificador diferencial de muy alta ganancia que posee alta impedancia de entrada y baja impedancia de salida [6].

Los amplificadores operacionales se utilizan para proporcionar cambios en la amplitud de voltaje, también en osciladores, en filtros y en circuitos de instrumentación [6].

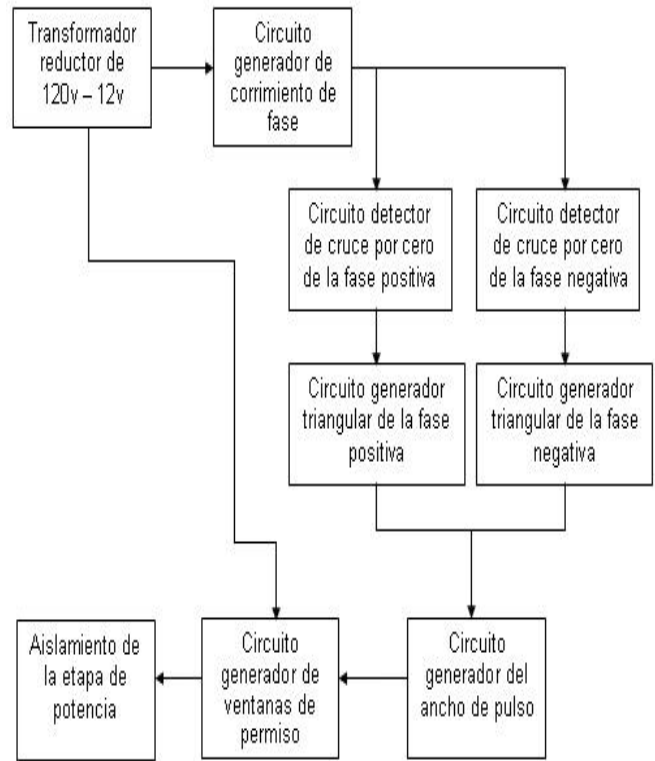


Fig. 3 Diagrama a bloques del circuito de control por ángulo simétrico

Cada módulo es primordial para que el funcionamiento del circuito completo sea óptimo; a continuación se explica la función de cada etapa del circuito.

El transformador se encarga de reducir el voltaje de línea de alimentación de CFE de 120v a 6v, para proporcionar la sincronía que requiere el circuito.

En la **Figura 4** se muestra el circuito generador de corrimiento de fase, llamado también desfasador, que es el que se encarga de desfasar la señal del

transformador 90°, para que la señal de onda triangular esté sincronizada con los cruces por cero de la señal senoidal del transformador.

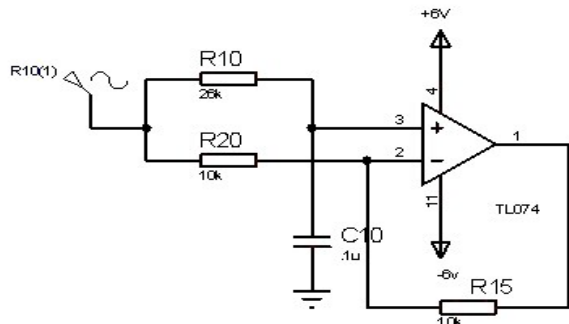


Fig.4 Circuito generador de corrimiento de fase

En las Figuras 5 y 6 se muestran los detectores de cruce por cero de la fase positiva y negativa, implementados con circuitos comparadores, obteniendo una señal de salida en saturación utilizada para generar las señales triangulares de cada fase.

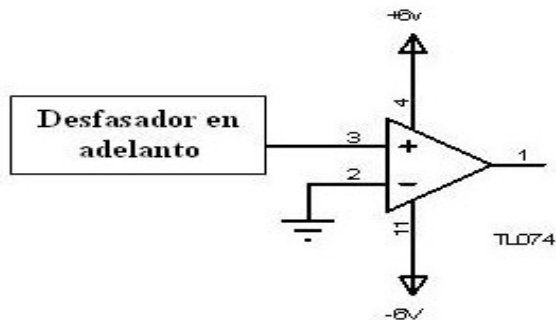


Fig.5 Detector de cruce por cero positivo

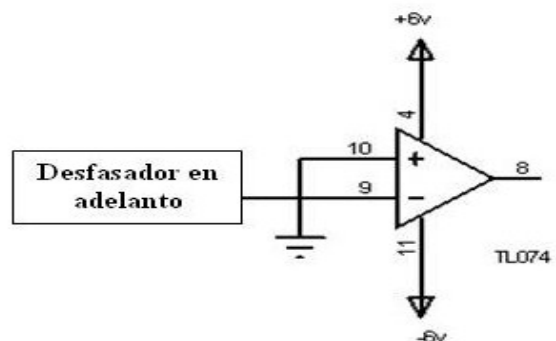


Fig.6 Detector de cruce por cero negativo

En las Figuras 7 y 8 se observan los circuitos para obtener las señales triangulares generadas por un integrador, que se encargan de convertir la señal de

salida de los detectores de cruces por cero, a señales triangulares para cada fase de la señal senoidal del transformador.

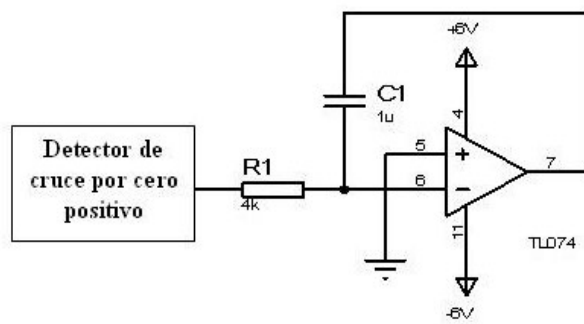


Fig.7 Generador de onda triangular positiva

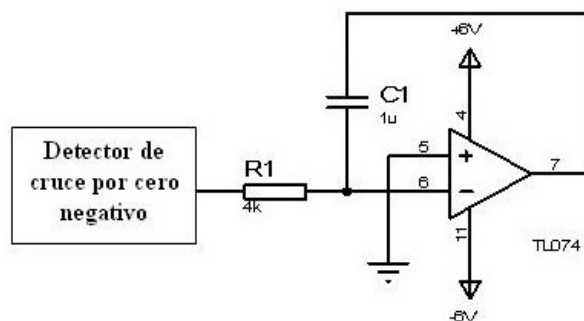


Fig.8 Generador de onda triangular negativa

Para obtener el tren de pulsos triangular sincronizado con las fases del transformador, se suman las salidas de los generadores de onda triangular.

El circuito generador de ancho de pulso está implementado con un comparador, el cual compara la señal triangular con un voltaje de referencia de CD, obteniendo así la variación del ancho de pulso de la técnica de control por ángulo simétrico; el circuito se observa en la Figura 9.

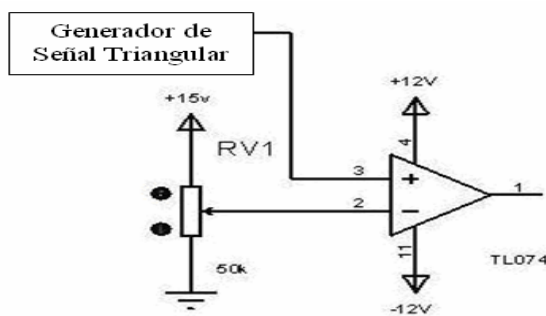


Fig.9 Generador de ancho de pulso

El circuito generador de la ventana de permiso habilita los pulsos para un semiciclo específico de la alimentación de CA, evitando así que los dispositivos controlados por ángulo de conducción, estén activos de forma simultánea, provocando un cortocircuito en la etapa de potencia.

El circuito está implementado con dos detectores de cruce por cero y compuestas AND, para aislar los pulsos para cada semiciclo; es necesaria esta etapa porque protege tanto a los dispositivos de conmutación como a la carga. Se observa el diagrama esquemático en la **Figura 10**.

Los dispositivos TTL se emplean como un puente para conectarse a los dispositivos más complejos en los sistemas digitales. También se usan como circuitos de interfaz para dispositivos que requieren el manejo de alta corriente [7].

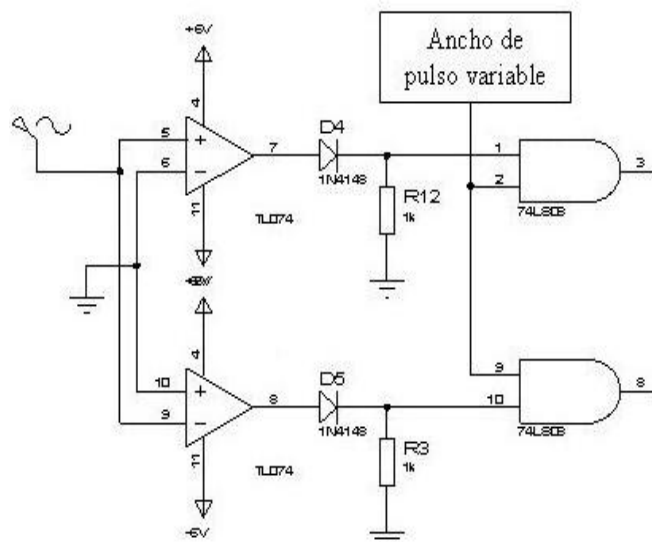


Fig.10 Circuito generador de las ventanas de permiso

El módulo de la etapa de aislamiento de potencia se utiliza para evitar que el circuito de control sufra algún daño, debido a que el circuito de potencia trabaja con voltajes y corrientes elevados; por ello es necesaria esta etapa para la protección del circuito. Se muestra en las **Figuras 11 y 12** el aislamiento de los pulsos implementado con optoacopladores para las dos fases de la señal senoidal.

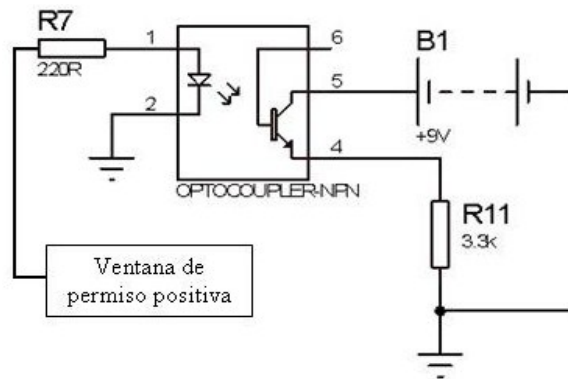


Fig. 11 Etapa de aislamiento positiva

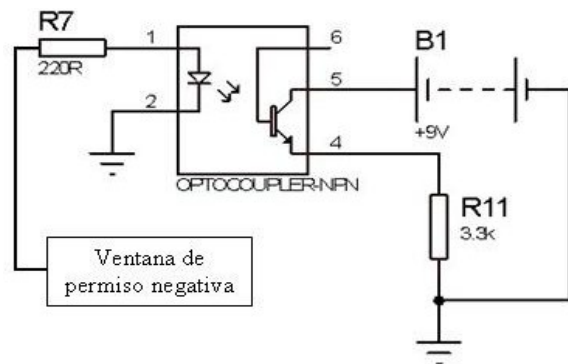


Fig.12 Etapa de aislamiento negativa

La técnica de control por ángulo simétrico al ser considerada una técnica de conmutación forzada, no es recomendable aplicarla para el control de tiristores (SCRs), debido a la dificultad para apagarlos antes del cruce por cero de la señal de corriente de CA. Por esta razón, se optó por utilizar MOSFETs como dispositivos de conmutación, así como por cuestiones de accesibilidad, bajo costo y por tener las características eléctricas óptimas que se necesitan.

Además de los MOSFET existen otros dispositivos que son recomendables para la utilización en técnicas de conmutación forzada, tal es el caso de los GTOs e IGBTs.

Etapa de potencia

Una de las consideraciones que debemos tomar con respecto a los MOSFET es que son dispositivos de conmutación, por consiguiente se debe hacer un arreglo implementado con cuatro diodos formando un puente rectificador y los

MOSFT, uno ó dos para cada rama de la fase dependiendo del tipo de rectificador, ya sea semicontrolado o totalmente controlado, siendo estos los que permiten regular el voltaje de salida en la carga.

Las hojas de datos de los parámetros eléctricos, configuraciones y características de los MOSFET IRF840 que se utilizaron en la implementación de rectificadores controlados se encuentran en [8].

En la **Figura 13** se muestra el diagrama de la etapa de potencia implementando un rectificador semicontrolado.

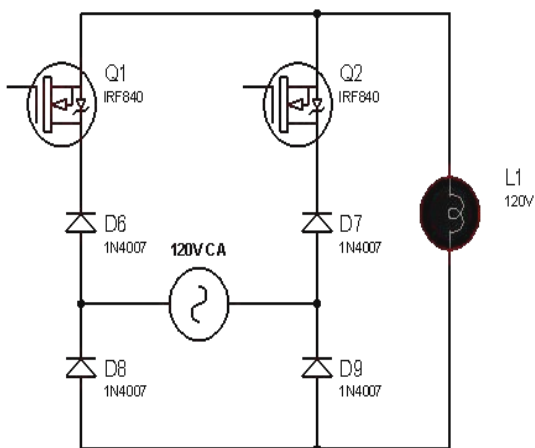


Fig.13 Rectificador semicontrolado

En la **Figura 14** se observa el diagrama de la etapa de potencia implementando un rectificador totalmente controlado.

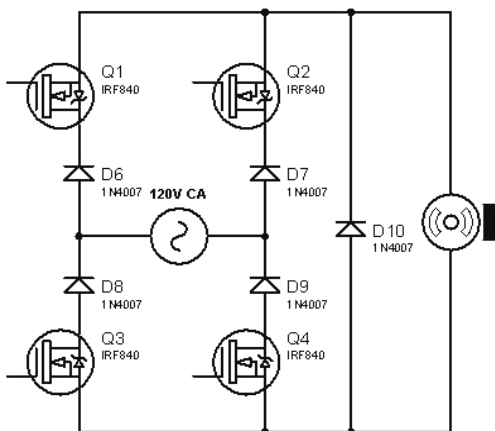


Fig. 14 Rectificador totalmente controlado

Ambos circuitos funcionan tanto para carga resistiva como para carga inductiva. Para el caso de la carga inductiva se debe agregar un diodo de rueda para darle continuidad a la corriente, ya que de lo contrario podría dañar a los MOSFET.

Resultados obtenidos de las pruebas realizadas a los rectificadores controlados monofásicos.

Se realizaron las pruebas correspondientes para el rectificador controlado tanto con carga resistiva como con carga inductiva, con un motor de CD de ¼ HP, midiendo los voltajes promedio y eficaz en la carga para calcular el rendimiento de los parámetros tales como el V_{ca} , η , FF y FR. De los valores calculados se obtienen las gráficas para analizar el comportamiento de los parámetros mencionados.

En la **Tabla 1** se observan los parámetros calculados de un rectificador semicontrolado para carga resistiva a diferentes ciclos de trabajo; mismos que no pudieron ser medido al 100% ya que el ángulo de conducción no logró controlar más del 80% del ciclo de trabajo en la carga.

Ciclo	Vrms	Vdc	Vca	η	FF	FR
5	41.59	8.86	40.53	4.53	470	458
10	56.5	17.35	53.5	9.51	324.7	308.6
15	68	25.77	62.74	14.43	263.6	243.8
20	78.51	35.17	70.19	20.06	223.3	199.4
25	86.6	43.1	75.11	24.76	200.1	174.7
30	94.46	51.68	79.06	29.93	182.5	152.6
35	99.6	58.33	80.73	34.29	170.3	138
40	105.9	66.88	82.1	39.88	158.1	122.2
45	110.3	73.95	81.83	44.94	149.3	110.6
50	114.3	80.65	80.99	49.78	141.9	100
55	116.4	85.6	78.87	54.08	135.2	92.7
60	120.1	92.8	76.23	59.7	129.4	82.3
65	121.3	98.14	71.28	65.46	123	72.4
70	122.3	102.3	67.02	69.93	119.9	65.56
75	124.8	108.8	61.13	76	114.8	56.24

Tabla 1 Parámetros de rendimiento del rectificador controlado para carga resistiva

En la **Figura 15** se observa el comportamiento de los voltajes V_{rms} , V_{dc} y V_{ca} de salida del rectificador; a medida que el ciclo de trabajo aumenta, el voltaje de la componente de CA disminuye debido a que los voltajes V_{rms} y V_{dc} se aproximan.

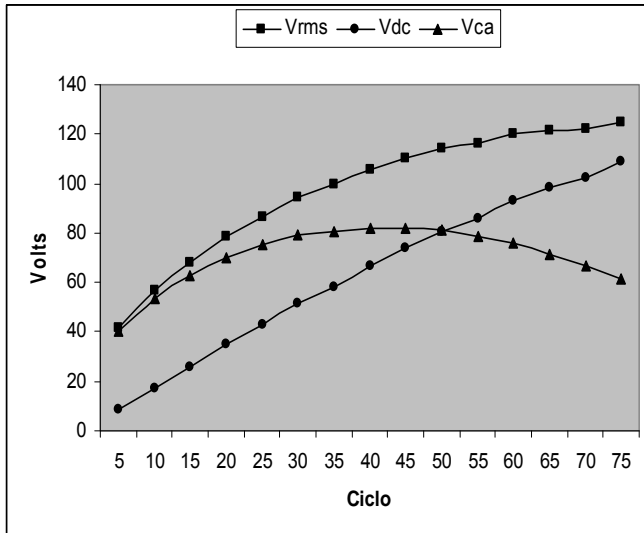


Fig. 15 Voltajes de salida del rectificador controlado para carga resistiva

En la **Figura 16** muestra el comportamiento del factor de forma y del factor de rizo, ambas tienden a mejorar al incrementarse el ciclo de trabajo del rectificador.

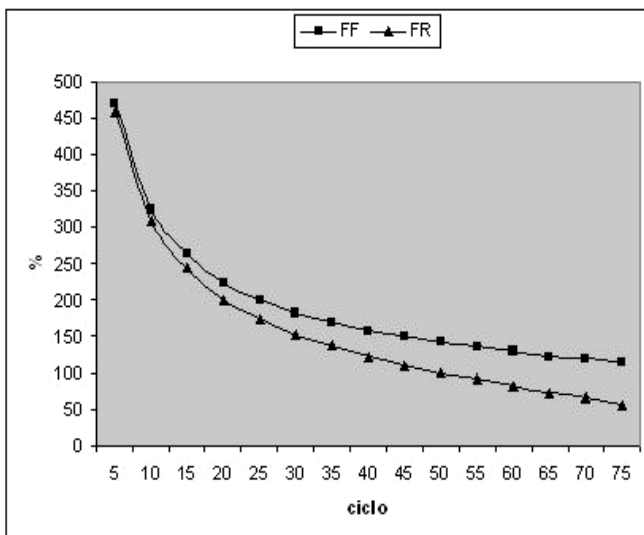


Fig. 16 Parámetros de rendimiento calculados para carga resistiva

En la **Figura 17** se presenta la gráfica de la eficiencia del rectificador controlado para carga resistiva; se observa que la eficiencia tiene una tendencia lineal, incrementando proporcionalmente cuando el ciclo de trabajo aumenta.

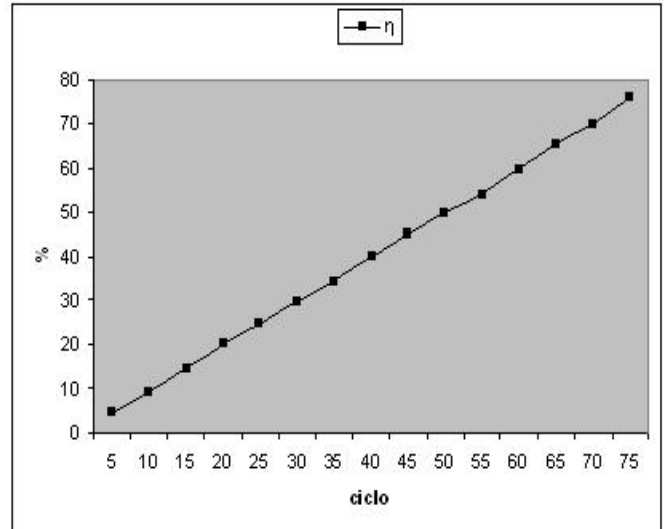


Fig. 17 Estimación de la eficiencia del rectificador semicontrolado para carga resistiva

En la **Figura 18** se observa la forma de onda para el voltaje de salida considerando una carga resistiva, para un ciclo de trabajo del 70% y un ángulo de conducción β aproximado de 150° .

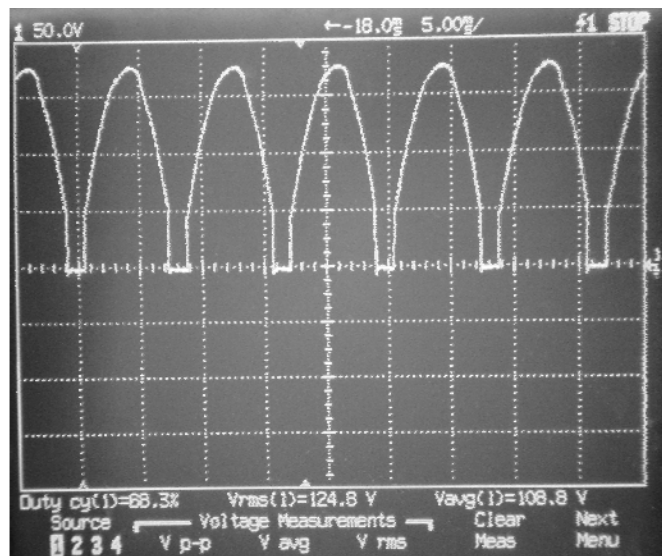


Fig. 18 Voltaje de salida de un rectificador controlado para carga resistiva

En la **Tabla 2**, se muestra el cálculo de los parámetros de rendimiento del rectificador considerando una carga inductiva; la fuerza contraelectromotriz que genera el motor utilizado como carga, provoca que el ciclo de trabajo no se estime adecuadamente.

Ciclo	Vrms	Vdc	Vca	η	FF	FR
25	44	21.9	38.16	24.77	200.6	174.15
60	73.94	58.5	45.22	62.6	126.4	77.08
80	100	89.37	44.86	79.86	111.9	50.41
85	107.2	97.2	45.21	82.21	110.5	46.78
90	115.7	106.7	44.73	85	108.1	41.91
95	149.1	144.1	36.74	93.9	103.7	25.48
100	152	150.7	19.83	98.3	100.9	13.06

Tabla 2 Parámetros de rendimiento del rectificador controlado para carga inductiva

En la **Figura 19** se muestra el comportamiento de los voltajes de salida del rectificador para carga inductiva a diferentes ciclos de trabajo.

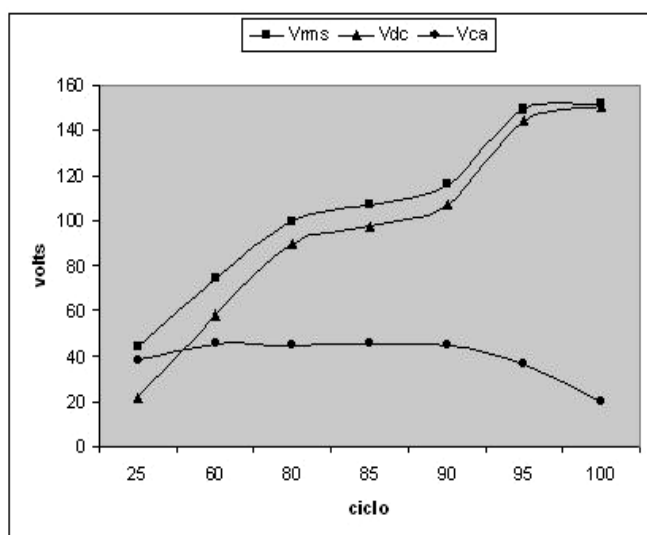


Fig. 19 Voltajes de salida del rectificador controlado para carga inductiva

La figura anterior demuestra que el ciclo de trabajo influye directamente en la eficiencia del rectificador, lo que se comprueba con el resultado mostrado en la **Figura 20**.

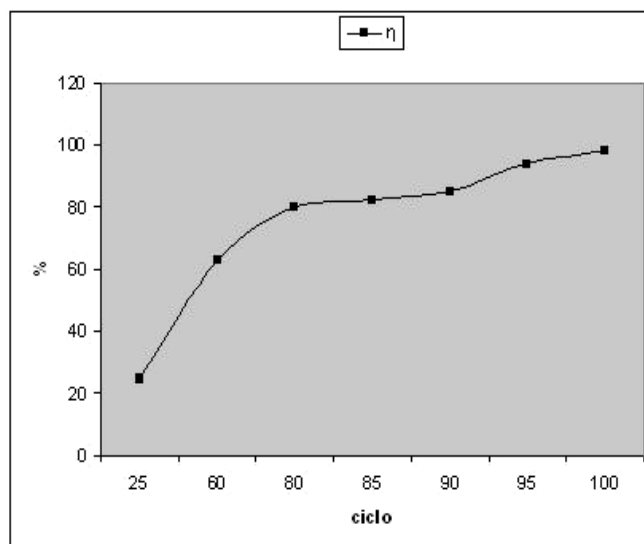


Fig. 20 Estimación de la eficiencia de un rectificador controlado para carga inductiva

De igual manera los factores de forma y de rizo mejoran debido al incremento del ciclo de trabajo, como se observa en las gráficas de la **Figura 21**. Se aprecia una mejora en estos parámetros respecto a los obtenidos con carga resistiva.

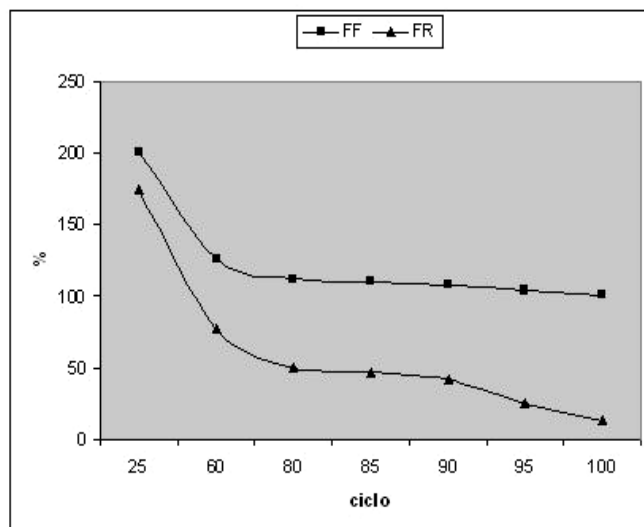


Fig. 21 Parámetros de rendimiento calculados para carga inductiva

En la **Figura 22** se muestra la forma de onda de salida de un rectificador controlado monofásico para carga inductiva con un ciclo de trabajo estimado del 100% con un ángulo de conducción β aproximado de 180°.

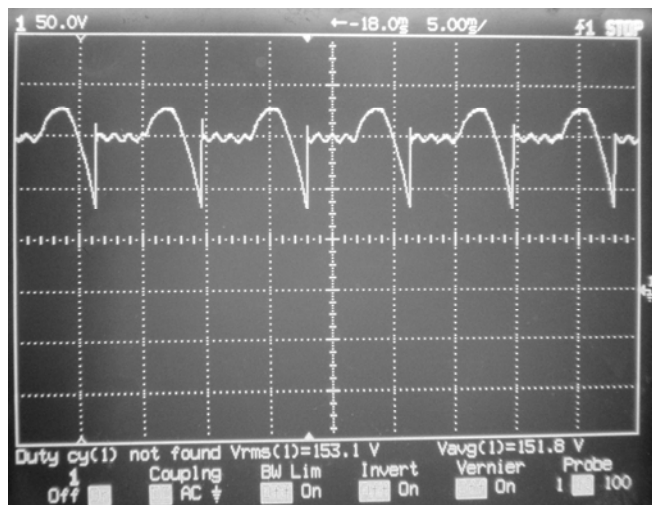


Fig. 22 Voltaje de salida de un rectificador controlado monofásico para carga inductiva

Los parámetros de rendimiento ideales para un rectificador que pueden utilizarse como comparativo con los resultados obtenidos, son los siguientes [1]:

- Eficiencia del 100%.
- Valor efectivo (rms) de la Componente de CA de 0%.
- Factor de rizo 0%.
- Factor de forma de 100%.

III. CONCLUSIONES Y TRABAJOS FUTUROS

Se han presentado los resultados de los cálculos realizados para obtener los parámetros de rendimiento de un rectificador semicontrolado monofásico con cargas resistiva e inductiva.

Para el caso de carga resistiva se obtuvo una eficiencia del 76% aproximadamente, un voltaje efectivo (rms) de la componente de CA de 61.3V, un factor de forma de 114% y un factor de rizo de 56%, tomando en cuenta que el ciclo de trabajo máximo fue del 75%.

Por otra parte, con carga inductiva se observó una mejora en los parámetros obtenidos, ya que para este caso, se obtuvo una eficiencia del 98.3%, el voltaje efectivo (rms) de la componente de CA se redujo hasta 19.8V, el factor de forma mejoró considerablemente a 100.9% y el factor de rizo se redujo hasta un 13% para un ciclo de trabajo del 100%.

Se puede concluir de los resultados obtenidos, que el rectificador presenta un mayor rendimiento con carga inductiva que con carga resistiva.

Para trabajos futuros se investigará si la técnica de control por ángulo simétrico mejora otros parámetros de rendimiento tales como:

- El factor de potencia.
- El factor de utilización del transformador.
- El factor armónico.

Así mismo se realizarán todos los análisis presentados y propuestos en este artículo utilizando un rectificador totalmente controlado.

IV. REFERENCIAS

- [1] M. H. Rashid. “*Electrónica de Potencia. Circuitos, Dispositivos y Aplicaciones*”. 2ª edición, Ed. Prentice Hall.
- [2] R. Wu, S. B. Dewan y G. R. Slemon; “*Analysis o an AC-to-DC voltage source converter using PWM with phase and amplitude control*”. IEEE Transactions on Industry Applications, Vol. 27 No. 2.
- [3] X. Ruan, L. Zhou y Y. Yan. “*Sof-switching PWM three-level converters*”. IEEE Transactions Electronic, Vol. 16, No. 5.
- [4] M. A. Boost y P. Ziogas. “*State-of Art PWM techniques*”. IEEE Transactions on Industry Applications, Vol. 24, No. 2.
- [5] Robert F. Coughlin, Federrik F. Driscoll. *Amplificadores Operacionales y Circuitos integrados Lineales*. 4ª ed. Ed. Prentice Hall. Año1993
- [6] Robert L. Boylestad, Louis Nashelsk; *Teoría de Circuitos y Dispositivos Electrónicos*. 8va ed. Ed. Prentice Hall.
- [7] Ronald J. Tocci, Javier León Cárdenas, Neal S. Widmer, Widmer Neal S; *Sistemas Digitales, Principios y Aplicaciones*. 8va ed. Ed. Prentice Hall.
- [8] TMOS POWER MOSFET, *Transistor Data* Motorola.

CÁLCULO DEL RENDIMIENTO DE LA TÉCNICA PWM DE MÚLTIPLES PULSOS PARA UN RECTIFICADOR CONTROLADO MONOFÁSICO.

Cruz Cruz Mario, Gómez González Francisco Javier, Muñoz Arzate Guillermo.

Departamento de Ingeniería Eléctrica y Electrónica.

Instituto Tecnológico de Veracruz.

Calzada Miguel Ángel de Quevedo No. 2779, Col. Formando Hogar, H. Veracruz, Veracruz.

Tel. 229-9381930, correo electrónico: ruzmario@hotmail.com

RESUMEN

En este artículo se presenta la aplicación de la técnica de modulación por ancho de pulso (PWM) de múltiples pulsos en el control de un rectificador controlado monofásico, que permite el cálculo de los parámetros de rendimiento tales como la eficiencia, el factor de forma y el factor de rizo. Se muestran gráficas comparativas de los parámetros mencionados para diferente número y ancho de pulsos, con la finalidad de extraer conclusiones.

Palabras clave: PWM de múltiples pulsos, rectificador controlado, parámetros de rendimiento, eficiencia, factor de forma, factor de rizo.

I. INTRODUCCIÓN

La técnica de modulación por ancho de pulso ha sido tratada en diferentes artículos [3, 4, 7 y 9]; sin embargo, éstos no tratan su aplicación en el cálculo del rendimiento de un rectificador controlado. Lo anterior, motivó la elaboración de este artículo.

En algunas aplicaciones tales como cargadores de baterías y en algunas clases de controles de velocidad para motores de CD y CA, es necesario contar con un voltaje de alimentación de CD controlable. La conversión de CA a CD controlable se consigue con convertidores controlados por fase por medio de tiristores [5].

Tradicionalmente, la potencia de salida en un rectificador controlado se consigue variando el ángulo de disparo de los tiristores (normalmente SCRs) que constituyen su topología. A esta técnica

de control se le conoce como técnica de control por ángulo de fase [2].

La técnica de control PWM de múltiples pulsos consiste en generar varios pulsos durante cada medio ciclo de la señal de alimentación de CA. Los pulsos se obtienen comparando una señal triangular (señal portadora) con una señal de referencia, tal y como se muestra en la figura 1.

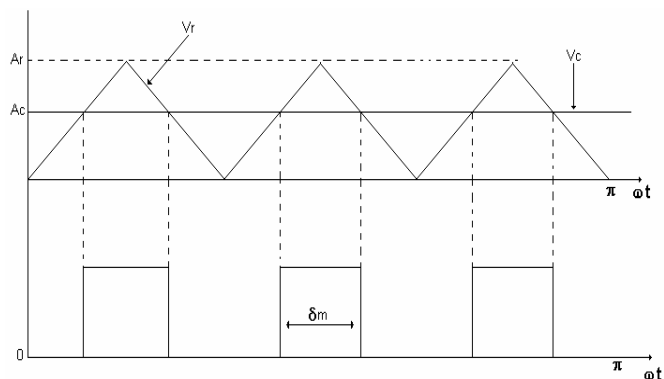


Figura 1.- Control PWM de múltiples pulsos.

En esta técnica, la frecuencia de la señal portadora determina el número de pulsos por cada medio ciclo, mientras que la magnitud de la señal de referencia permite el control del ancho de pulso. Es importante notar que la señal triangular deberá estar en todo momento en sincronía con la señal de alimentación de CA.

La figura 1 ejemplifica el caso de 3 pulsos en un medio ciclo y la figura 2 muestra la sincronía de la señal portadora con la señal de alimentación, caso en el que se obtiene un solo pulso por cada medio ciclo.

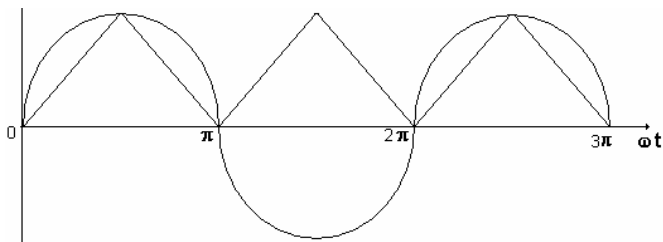


Figura 2.- Señal triangular sincronizada para un solo pulso por cada medio ciclo.

II. MÉTODO.

En la figura 3 se muestra el diagrama a bloques del circuito de control diseñado. Las etapas que lo componen son:

- 1.- Transformador reductor.
- 2.- Filtro pasa todo en adelante.
- 3.- Detector de cruce por cero.
- 4.- Multiplicador de frecuencia.
- 5.- Generador de ciclos negativos.
- 6.- Generador de la señal triangular.
- 7.- Generador de ventanas de permiso.
- 8.- Circuito generador de ancho de pulso.

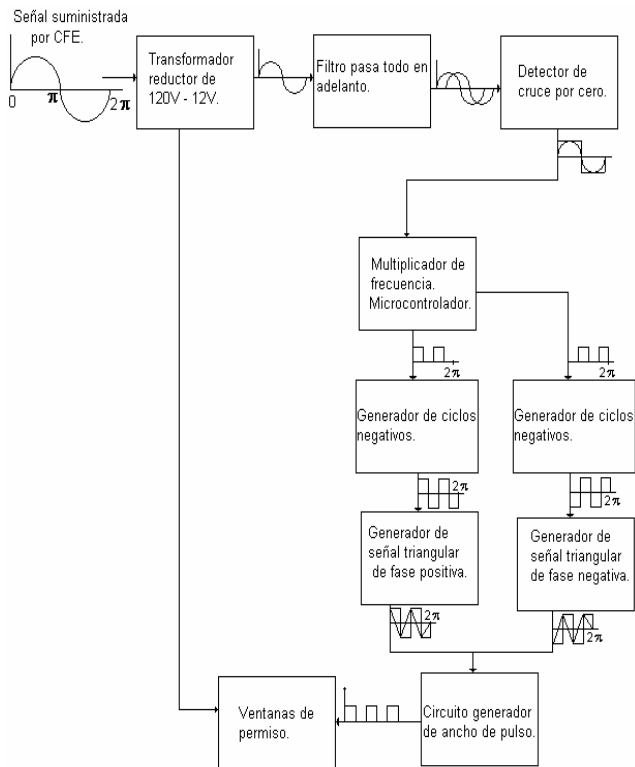


Figura 3.- Diagrama a bloques de la etapa de control.

La función principal del transformador reductor es la de disminuir el voltaje de alimentación de 127 a 12 volts para trabajar adecuadamente con el resto de las etapas.

Por su parte, el filtro pasa todo en adelante realiza un desfase de la señal de entrada, necesario para lograr la sincronía de la señal portadora con la señal senoidal de alimentación.

Se optó por usar un potenciómetro para modificar el grado de desfase de la señal de sincronía, de acuerdo con el número de pulsos por cada medio ciclo requerido.

En la figura 4 se ilustra el circuito del filtro pasa todo en adelante.

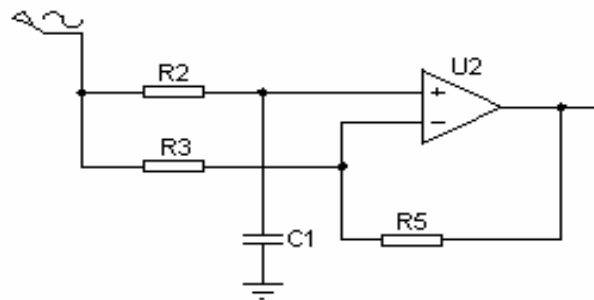


Figura 4.- Filtro pasa todo en adelante.

El detector de cruce por cero mostrado en la figura 5, tiene la función de convertir la señal senoidal en una señal cuadrada con sus dos semiciclos, positivo y negativo; el semiciclo negativo es eliminado usando un diodo rectificador, mientras el positivo es enviado al microcontrolador, quien se encarga a su vez de procesar la señal del detector de cruce por cero y enviar la señal a los generadores de ciclos negativos.

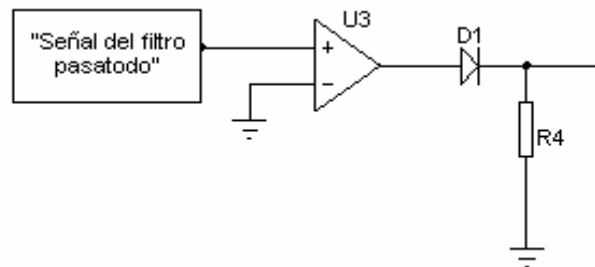


Figura 5.- Detector de cruce por cero.

Con base en el número de pulsos seleccionados, el microcontrolador proporciona dos señales digitales. La conexión del microcontrolador se muestra en la figura 6.

Como se puede notar, se cuentan con dos *push-buttons* para seleccionar el número de pulsos y un *display* para visualizar los pulsos seleccionados.

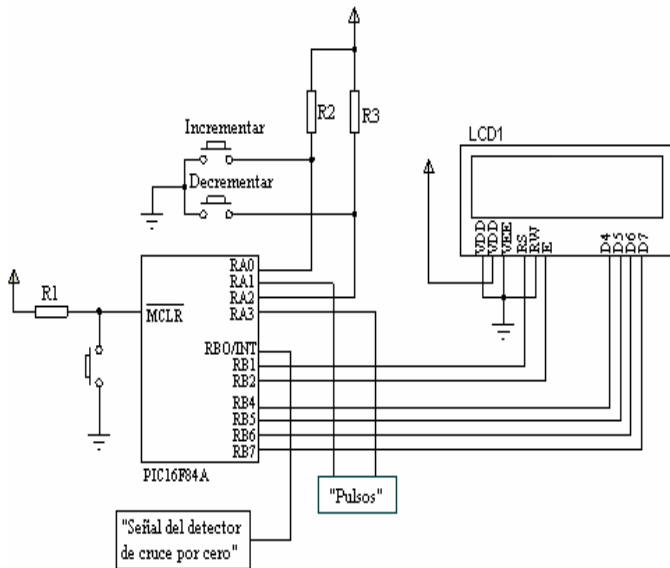


Figura 6.- Conexión del microcontrolador.

Los pulsos generados por el microcontrolador son enviados al generador de ciclos negativos el cual esta configurado como comparador, cuando el comparador detecta cero volts a la entrada automáticamente a la salida genera un voltaje negativo. Esto se realiza con el circuito mostrado en la figura 7.

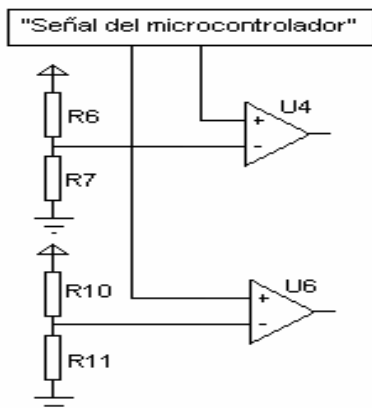


Figura 7.- Generador de ciclos negativos.

Las señales de salida del generador de ciclos negativos son procesadas por un par de generadores de señal triangular como los que se pueden apreciar en la figura 8.

La señal que se genera con estos circuitos es rectificada a través de un diodo, con la finalidad de eliminar los semiciclos negativos.

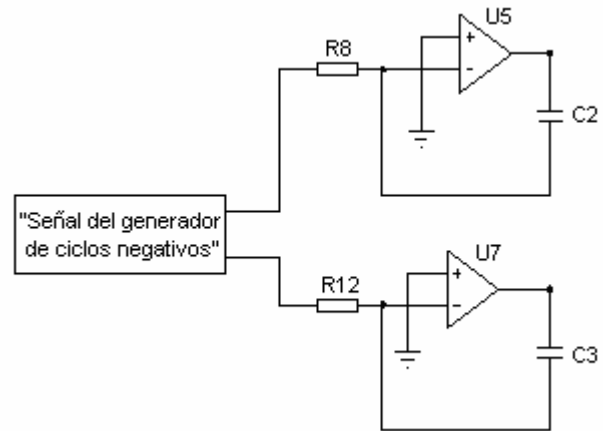


Figura 8.- Generador de la señal triangular.

De este circuito se obtiene la señal portadora, que comparada con la señal de referencia, permite la modulación del ancho de los pulsos generados. El circuito del comparador implementado se muestra en la figura 9.

La figura 10 muestra el esquema del circuito generador de las ventanas de permiso, útiles para la creación de los pulsos de disparo para los dispositivos de potencia que forman el arreglo del rectificador controlado.

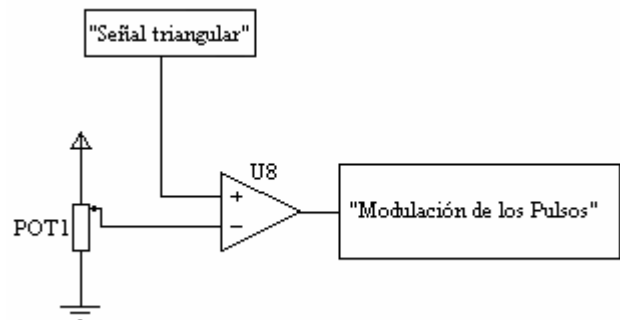


Figura 9.- Circuito comparador.

Este circuito permite que los pulsos queden específicamente activos para cada semiciclo de la

señal senoidal de alimentación, con el objetivo de evitar cortocircuitos en la etapa de potencia y proteger así a los dispositivos.

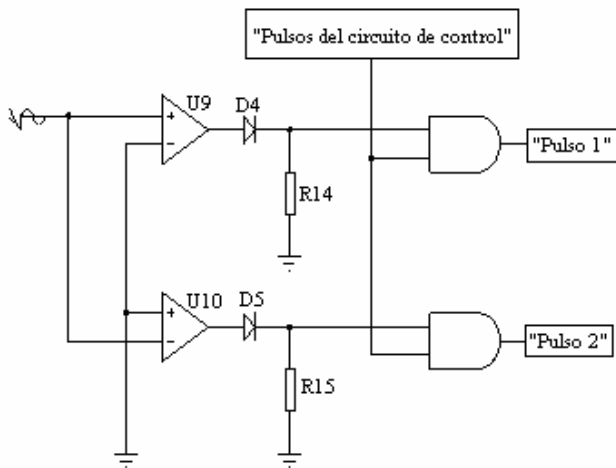


Figura 10.- Ventanas de permiso.

En la figura 11 se presenta el circuito correspondiente a uno de los rectificadores controlados monofásicos implementados para la validación del circuito de control diseñado. Se trata de la topología de un rectificador semicontrolado monofásico, el cual emplea como elementos principales de conmutación un par de MOSFETs.

También se aprecian un par de optoacopladores (opto transistores), los cuales tienen la función de aislar eléctricamente las etapas de control y de potencia.

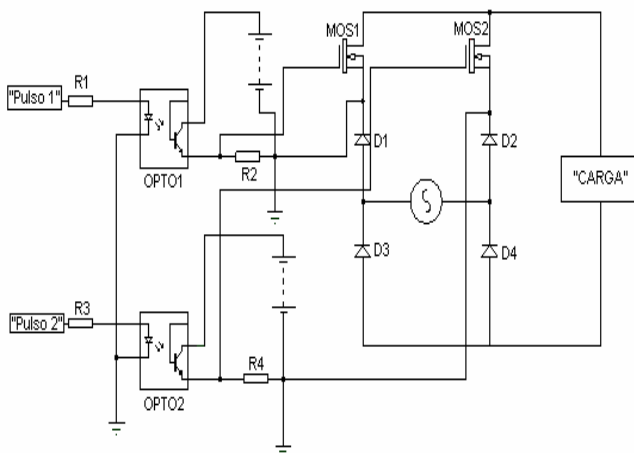


Figura 11.- Etapa de potencia.

Una vez implementadas ambas etapas, se procedió a llevar a cabo las pruebas que permitieron obtener resultados.

PRUEBAS REALIZADAS.

Primeramente se comprobó el funcionamiento del circuito haciendo uso de software de simulación. Los resultados obtenidos en la simulación fueron los esperados, por lo que se procedió a la implementación física.

Posteriormente se realizaron pruebas en el laboratorio de Ingeniería Electrónica del Instituto Tecnológico de Veracruz. Las pruebas consistieron en la medición de voltajes (Vrms y Vcd) para diferente número y ancho de pulsos, tanto para carga resistiva como para carga inductiva.

Se llevaron a cabo dos grupos de mediciones, uno para un rectificador semicontrolado y otro para un rectificador totalmente controlado, ambos monofásicos. Con las mediciones obtenidas se calcularon los siguientes parámetros: eficiencia, factor de forma y factor de rizo. Todas estas mediciones se realizaron para uno y dos pulsos por cada medio ciclo de la señal de alimentación de CA.

RESULTADOS OBTENIDOS.

I.- Para el rectificador semicontrolado monofásico y un pulso por cada medio ciclo (carga resistiva).

En la figura 12 se muestra la gráfica para la eficiencia.

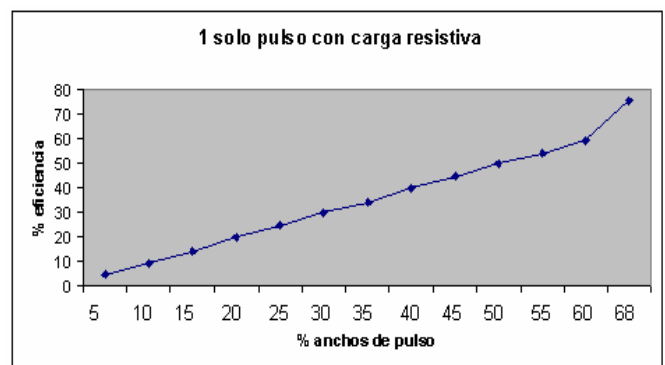


Figura 12.- Eficiencia con carga resistiva.

Como se observa en dicha figura, a mayor ancho de pulso la eficiencia del rectificador tiende a incrementarse.

En la figura 13 se aprecia el comportamiento del factor de forma para diferentes anchos de pulso. Conforme se incrementa el ancho de pulso, el factor de forma disminuye, es decir, mejora.

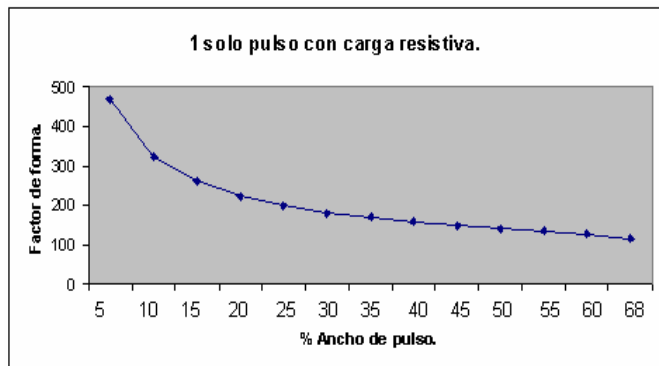


Figura 13.- Factor de forma con carga resistiva

La figura 14 presenta el factor de rizo obtenido. Como se puede ver en la gráfica, el factor de rizo es bajo para anchos de pulso grandes.

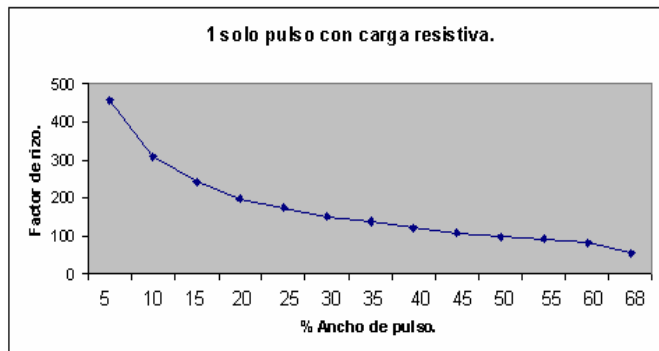


Figura 14.- Factor de rizo con carga resistiva.

II.- Para el rectificador semicontrolado monofásico y un pulso por cada medio ciclo (carga inductiva).

La figura 15 presenta la gráfica de la eficiencia obtenida, utilizando ahora una carga inductiva. Se puede notar que para el ancho de pulso máximo, la eficiencia del rectificador es casi del 100%, lo cual se aproxima al valor ideal correspondiente a este parámetro.

Comportamientos similares se obtuvieron para el factor de forma y el factor de rizo.

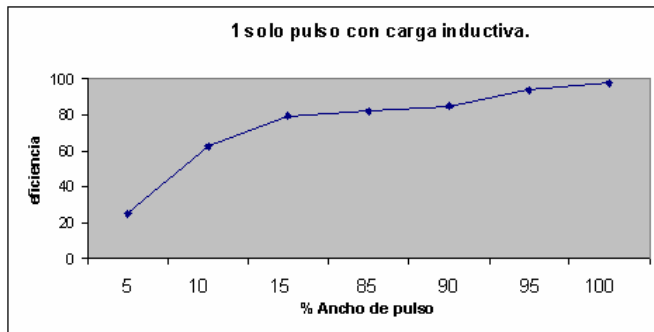


Figura 15.- Eficiencia con carga inductiva.

La figura 16 confirma la mejora del factor de forma para el ancho de pulso máximo. Se puede notar que bajo esta condición, este parámetro es prácticamente del 100%.

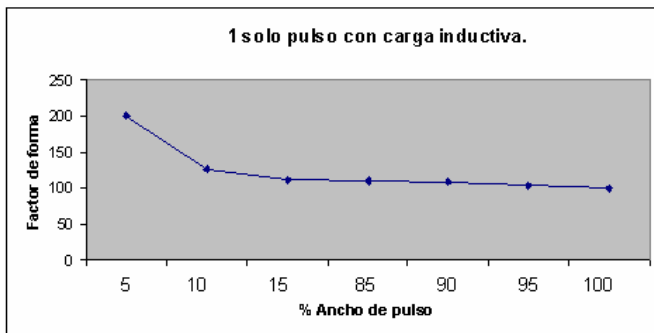


Figura 16.- Factor de forma con carga inductiva

Al igual que para los parámetros anteriores, el factor de rizo mejora considerablemente. En la figura 17 se comprueba que para el ancho de pulso máximo, el factor de rizo se acerca al valor ideal del 0%.

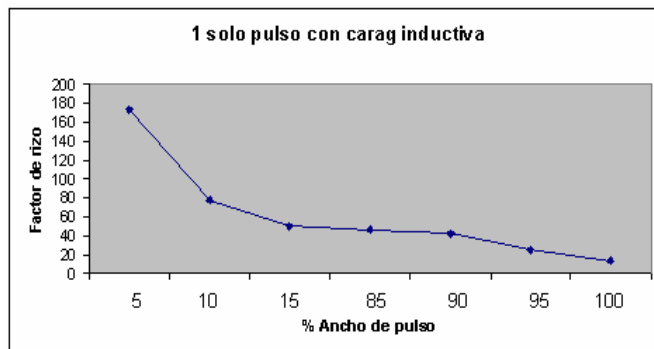


Figura 17.- Factor de rizo con carga inductiva.

III.- Para el rectificador semicontrolado monofásico y dos pulsos por cada medio ciclo (carga resistiva).

Para este caso, la gráfica de la eficiencia se muestra en la figura 18. Comparando este resultado con el obtenido para el caso de un solo pulso por cada medio ciclo, se aprecia una mejora en este parámetro. Como se observa, a mayor ancho de pulso la eficiencia del rectificador se incrementa.

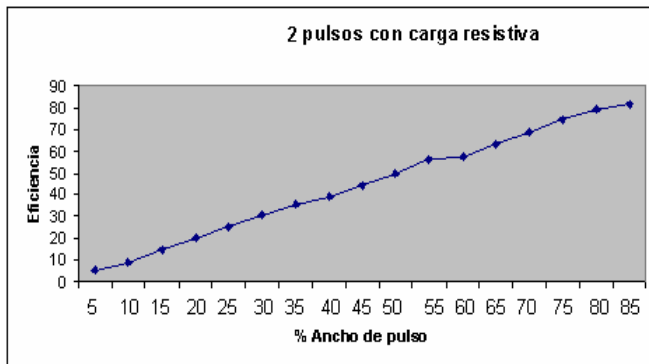


Figura 18.- Eficiencia con carga resistiva.

En la figura 19 se muestra el factor de forma del rectificador semicontrolado. Analizando y comparando esta gráfica con la de la figura 13, se observa que el factor de forma mejora también considerablemente.

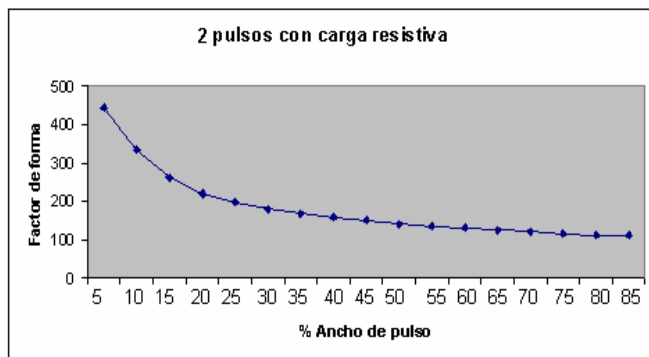


Figura 19.- Factor de forma con carga resistiva.

La figura 20 presenta el factor de rizo obtenido. Como se puede ver en la gráfica, el factor de rizo es bajo para anchos de pulso grandes. Si comparamos ésta con la de la figura 14, se nota que el factor de rizo se reduce.

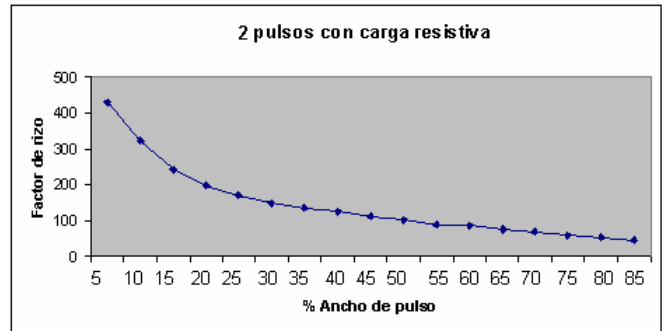


Figura 20.- Factor de rizo con carga resistiva.

IV.- Para el rectificador semicontrolado monofásico y dos pulsos por cada medio ciclo (carga inductiva).

En la figura 21 se muestra la eficiencia considerando la carga inductiva. Como se observa en dicha gráfica, para anchos de pulso cercanos al máximo, la eficiencia es mayor al 90%.

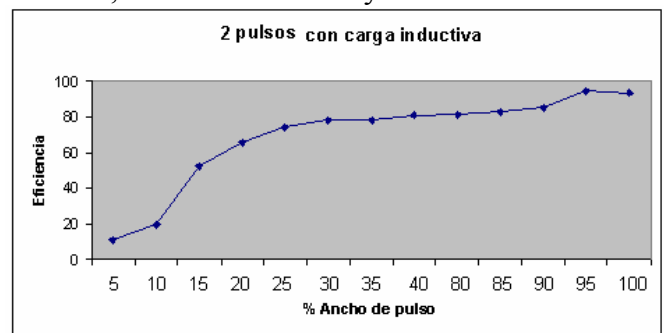


Figura 21.- Eficiencia con carga inductiva.

La figura 22 confirma la mejora del factor de forma para el ancho de pulso máximo. Se puede notar que bajo esta condición, este parámetro es prácticamente del 100%.

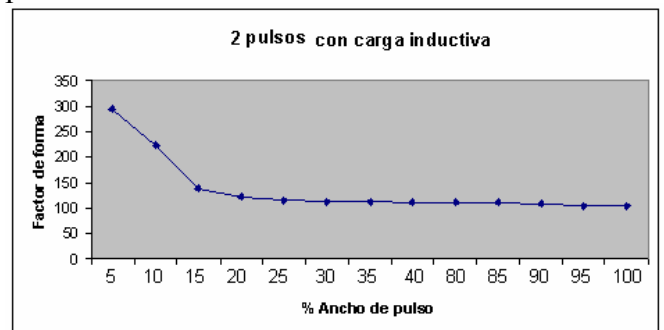


Figura 22.- Factor de forma con carga inductiva.

La figura 23 presenta el factor de rizo obtenido. Como se puede ver en la gráfica, este parámetro es bajo para anchos de pulso grandes.

Como se puede ver en las gráficas con carga inductiva, para el rango entre 15 y 85% de ancho de pulso, los parámetros se mantienen con variaciones mínimas.

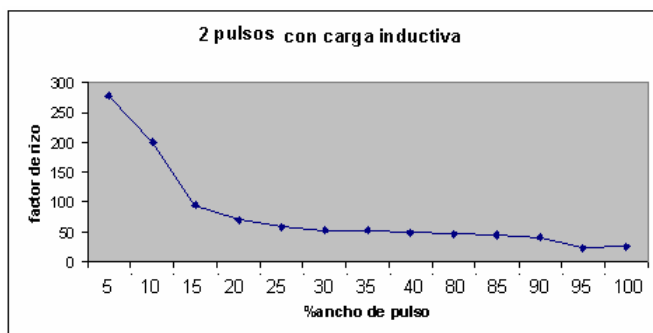


Figura 23.- Factor de rizo con carga inductiva.

Todas las pruebas realizadas con carga inductiva fueron con un motor de CD de excitación separada.

La figura 24 corresponde a la forma de onda del voltaje de salida considerando dos pulsos por cada medio ciclo. Aquí se aprecia el efecto de la fuerza contra-electromotriz producida por el motor utilizado en las pruebas.

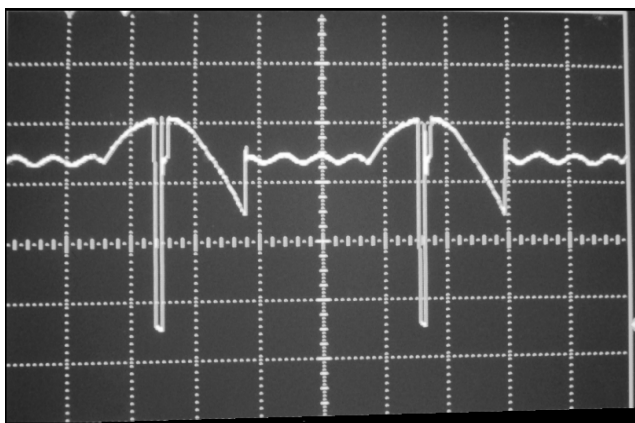


Figura 24.- Forma de onda con carga inductiva.

En la figura 25 se muestra la forma de onda para una carga resistiva con dos pulsos, esta corresponde a un ancho de pulso cercano al 100%.

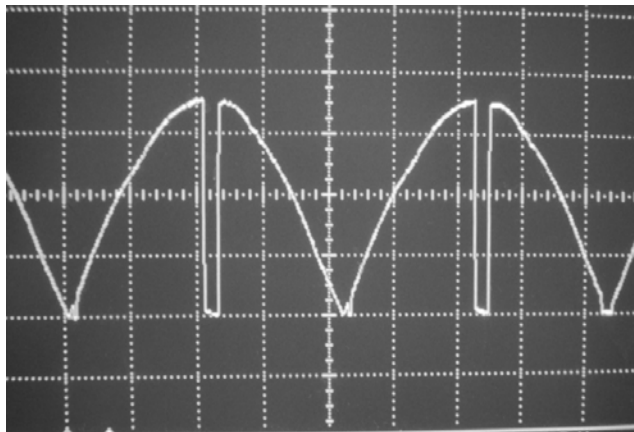


Figura 25.- Forma de onda con carga resistiva.

III.- CONCLUSIONES.

Se presentaron los resultados obtenidos de la evaluación de la técnica por modulación por ancho de pulso (PWM) de múltiples pulsos, para el cálculo de los parámetros de rendimiento de un rectificador semicontrolado monofásico.

Las pruebas se realizaron para carga resistiva e inductiva, considerando para ambos casos uno y dos pulsos por cada medio ciclo de la señal de alimentación de CA. Sin embargo, cabe mencionar que el diseño permite la operación para tres, cuatro y hasta cinco pulsos por cada medio ciclo.

De los resultados mostrados en este artículo se concluye que, para el caso de cargas resistivas, la técnica permite un mayor rendimiento para dos pulsos por cada medio ciclo. Por otra parte, para una carga inductiva (motor CD), los parámetros de rendimiento fueron superiores para un solo pulso por cada medio ciclo.

Como trabajo futuro, se estimarán los parámetros presentados en este artículo para el caso de tres, cuatro y cinco pulsos por cada medio ciclo, con la intención de compararlos con los aquí documentados.

También se estudiará si la técnica es adecuada para mejorar otros parámetros importantes, como son el factor de potencia, el factor de utilización del transformador y el factor armónico.

IV.- REFERENCIAS

- [1] *Enrique Palacios, Fernando Remiro Domínguez-, Microcontrolador PIC16F84. Desarrollo de proyectos. 2 Edición. Editoria Alfa-Omega.*
- [2] *M. H. Rashid; Electrónica de Potencia. Circuitos y Aplicaciones. 2 Edición. Ed. Prentice Hall.*
- [3] *M.H. Rashid y M. Aboudina, “Análisis of forced-commutated techniques for ac-cd converters”. 1st European Conference on power Electronics and Applications, Bruselas.*
- [4] *M. A. Boost y P. Ziogas. “State-of Art PWM techniques”. IEEE Transactions on Industry Applications, Vol. 24, No. 2.*
- [5] *Ned Mohan, Tore M. Undeland, William P. Robbins. Power Electronics. Converters, Applications, and design. Second. Edición.*
- [6] *Robert F. Coughlin, Federrik F Driscoll. Amplificadores Operacionles. 2 Edición. Ed. Prentice Hall.*
- [7] *R. Wu, S. B. Dewan y G. R. Slemon; “Analysis o an AC-to-DC voltage source converter using PWM with phase and amplitude control”. IEEE Transactions on Industry Applications, Vol. 27 No. 2.*
- [8] *Tmos Power MOSFET. Transistor Data. Motorola.*
- [9] *X. Ruan, L. Zhou y Y. Yan. “Sof-switching PWM three-level converters”. IEEE Transactions Electronic, Vol. 16, No. 5.*

Implementación de un termómetro digital con interface RS232 con el microcontrolador MC9RS08KA2

Adalberto Bustamante López

CENTRO DE ENSEÑANZA TÉCNICA Y SUPERIOR

Campus Tijuana

Av. Grandes Lagos s/n Fracc. El Lago

Tijuana, Baja California, México

Resumen— En este documento se presenta el diseño e implementación de una aplicación para la medición y muestreo de temperatura basada en el microcontrolador MC9RS09KA2 de Freescale®. La problemática abordada es referente al diseño de sistemas de monitoreo de temperatura donde el costo y opción de integración modular son factores primordiales, por lo que se propone una solución basada en el en componentes estándares y una interfaz ampliamente utilizada por diversos sistemas industriales. La medición de temperatura se realiza mediante un convertidor analógico-digital implementado programáticamente en el MC9RS09KA2, utilizando un temporizador, un comparador analógico interno, además de un circuito RC, y un divisor de voltaje con una resistencia NTC. El resultado de la conversión se transmite mediante una interface RS232 a un puerto COM de una PC, donde es monitoreado mediante un programa visual realizado en C#.NET.

Términos utilizados —ADC, Comparador, Controlador, Filtro, IDE, Instrucciones, MCU, Registros, RS08, Temperatura, Timer, RS232, Serie

I. INTRODUCCIÓN

EL desarrollo de aplicaciones embebidas de bajo costo es una de las razones por las cuales las

tecnologías de microcontroladores de 8 bits continúan siendo vigentes hoy en día.

Actualmente, una gama impresionante de microcontroladores ofrecen dentro de su arquitectura los módulos ADC de alta resolución e interfaces como SPI, I²C, etc., a un precio razonable. Sin embargo, algunos microcontroladores de 8 bits, como los de la serie RS08 de Freescale®, son de muy bajo costo, por lo que pueden reducir significativamente el costo de producción a gran escala en comparación con utilizar un microcontrolador de mayor potencia o circuitos integrados digitales/analógicos convencionales.

En la presente aplicación, se utiliza el microcontrolador MC9RS09KA2 para implementar un termómetro digital con interface a través del protocolo RS232 hacia un puerto de comunicaciones (COM) de una computadora personal. La razón por la que se ha implementado bajo comunicación serie, es debido a la poca periferia que nos ofrecen los microcontroladores de la familia RS08, siendo que el MC9RS09KA2 contiene sólo 6 terminales disponibles como entradas y/o salidas¹.

El transductor que se utiliza para obtener la temperatura es un termistor o resistencia NTC, ya

¹ Para obtener información específica sobre la periferia disponible en el controlador, revisar el documento [1] en la sección *Chapter 6. Parallel Input/Output Control*.

que estos sensores son de una fracción del costo en comparación con otros, como los sensores lineales. Para llevar a cabo la lectura del termistor, se ha configurado un divisor de voltaje en conjunto con una red RC, ya que la conversión analógica-digital propuesta se basa en medir el tiempo en el ciclo de carga de un capacitor hasta llegar al valor de voltaje presente en el divisor de voltaje, que representa la temperatura actual.

La aplicación ha sido desarrollada en el lenguaje ensamblador propietario de la compañía Freescale® para su arquitectura RS08, bajo el entorno de desarrollo Freescale CodeWarrior [2]. La aplicación de PC que recibe la información proveniente del microcontrolador ha sido desarrollada en lenguaje C#.NET, en el entorno de desarrollo Microsoft Visual C# Express 2008.

II. DESARROLLO ELECTRÓNICO

A. Sensor de temperatura

Para sensar la temperatura se ha seleccionado una resistencia NTC, o también llamada termistor, la cual hace variar su resistencia en función de la temperatura presente en el ambiente. La configuración para obtener un nivel de voltaje que represente una cierta temperatura se realiza formando un divisor de voltaje con la resistencia NTC como se observa en la *figura 1*. Existen otros métodos para poder convertir el valor de resistencia en el termistor a voltaje, como el Puente de Wheatstone, pero requieren de más circuitería externa, además de un convertidor analógico-digital disponible en microcontrolador para ser interpretado.

El microcontrolador MC9RS08KA2 no tiene integrado un módulo para la conversión analógica-digital, por lo que éste se debe implementar utilizando algún algoritmo convencional. De suerte, el MC9RS08KA2 contiene un comparador analógico, con el cual se puede realizar la conversión apoyándose de una red RC sencilla.

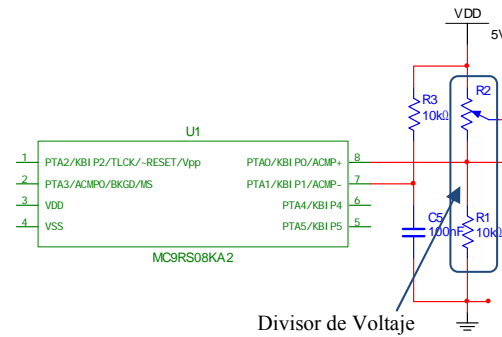


Figura 1: Red RC y divisor de voltaje con la Resistencia NTC conectados al MC9RS08KA2

Para obtener un valor digital del nivel de voltaje presente en la resistencia NTC, es necesario utilizar un ciclo de carga de la red RC y utilizar el comparador analógico para detectar el momento en el cual el capacitor se carga al nivel de voltaje del nodo formado por la resistencia NTC y la resistencia de 10.0K. De esta manera, la conversión analógica-digital se hará *midiendo el tiempo en el que el capacitor adquiere un nivel de voltaje igual² al nodo de la resistencia NTC*.

Para que la conversión sea exitosa, se debe configurar el periodo del *Modulo Timer (MTIM)*³ ajustándose al tiempo de establecimiento de los componentes que conforman la red RC. El capacitor se carga de manera logarítmica como se aprecia en la *figura 2*, por lo que para lograr la linealidad de la medición se requiere una corrección exponencial en software⁴.

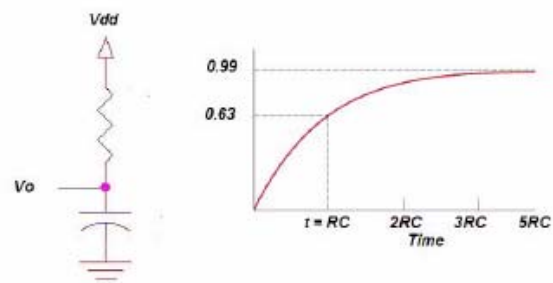


Figura 2: Carga de la red RC

La constante de tiempo de carga se calcula en base a la relación

$$t = 5RC$$

² Para que se genere una interrupción en el ACMP (Comparador analógico) el voltaje del capacitor debe sobrepasar ligeramente el valor de voltaje en el nodo de la terminal no inversora (la resistencia NTC).

³ El *Modulo Timer* del MC9RS08KA2 es un temporizador que se basa en subdividir la frecuencia del oscilador interno de 32 KHz del microcontrolador.

⁴ Debido a las limitaciones de memoria del MC9RS08KA2 la corrección exponencial se realiza dentro de la aplicación de C#.

Para la presente aplicación se utiliza una resistencia de 10.0k y un capacitor de Tantalio de 0.1 uF. Por lo tanto, el tiempo de establecimiento es

$$t = 5(10^4)(0.1 \times 10^{-6}) = 5 \text{ ms}$$

El tiempo en que ocurre una interrupción por sobreflujo en el MTIM debe ser menor o igual al tiempo de establecimiento, de otra manera, se desbordaría el Timer-Counter (MTIMCNT) antes de que ocurriera la interrupción en el comparador analógico y pudiera guardarse dicho valor.

Bajo la selección de un tiempo de establecimiento de $t = 5 \text{ ns}$, y considerando el oscilador interno del microcontrolador en 32Khz, el divisor de frecuencia se calcula por:

$$\text{prescaler} = 32000\text{hz} \times 0.005\text{s} = 160$$

El valor de 160 para el divisor de frecuencia no se puede establecer directamente en el microcontrolador, ya que éste solo tiene ciertos valores fijos que pueden ser utilizados. Los valores más cercanos son 128 y 256, por lo que se escoge 128, ya que un divisor más alto, haría que el módulo timer no pudiera registrar tiempos para valores de voltaje altos en el nodo del termistor y la resistencia de 10.0K, como ya se ha mencionado anteriormente.

Para iniciar el algoritmo de conversión es necesario que el capacitor esté totalmente descargado. Para esto, se efectúa un ciclo de descarga previo, el cual consiste en mandar un cero lógico al nodo del capacitor. Se debe utilizar un temporizador sencillo que proporcioné el tiempo suficiente para que se realice la descarga, como lo puede ser un ciclo en el que un contador cuente hasta 254, y después continúe la aplicación.

Una vez descargado el capacitor, se habilita el comparador analógico por interrupción, el MTIM en modo de free-running y se activa la salida lógica en "1". El programa entonces, debe entrar en un modo de espera hasta que se produzca una interrupción del comparador analógico, y se debe capturar el valor actual del registro MTIMCNT. Hasta este momento se debe tener una conversión analógico-digital exitosa, pero aún sin ajuste lineal. El dato en

MTIMCNT se transmite hacia una computadora que va procesar y a desplegar el valor representativo de temperatura.

B. Diagrama electrónico

A continuación se presenta el diagrama esquemático completo de la aplicación implementada, seguido de la lista de componentes necesarios. La resistencia NTC fue seleccionada similar a la empleada en el kit de desarrollo de Softec ® DEMO9RS08KA2, en donde se diseñó la aplicación. Para mayor referencia, veáse [4].

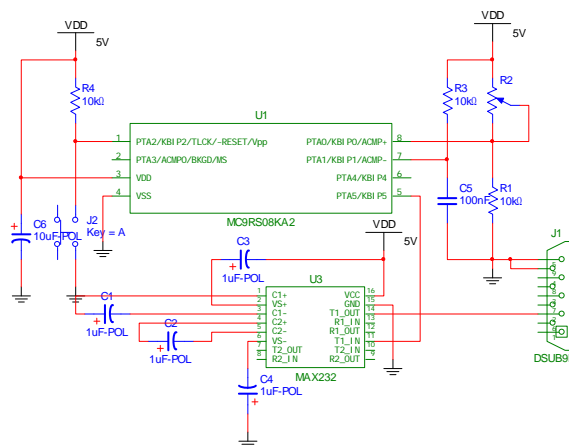


Figura 3: Diagrama esquemático completo de la aplicación

Cantidad	Descripción	Referencia Designada
1	Conector DB9	J1
1	C.I. MAX232	U3
4	Capacitor Electrolytico, 1uF	C1, C2, C3, C4
3	Resistencia de 10.0kΩ 5%, ¼ Watt	R4, R3, R1
1	Switch de pulso miniatura	J2
1	Capacitor de Tantalio, 100nF	C5
1	NTC Termistor 10K 10% 0805, B57620-C103-J62	R2
1	MC9RS08KA2	U1

Tabla 1: Lista de Componentes⁵.

Nótese que se está utilizando un circuito integrado MAX232 para convertir al estándar eléctrico de transmisión RS232, y sólo se está utilizando en esta ocasión el pin de transmisión hacia un conector DB9, que puede ser una computadora personal o cualquier otro dispositivo con interface serie.

⁵ El diseño del prototipo inicial se basó en la circuitería que utiliza el DEMO9RS08KA2. Ver [5]

III. ESTIMACIÓN DE LA TEMPERATURA POR LA ECUACIÓN DE STEINHART-HART SIMPLIFICADA

En base al diseño electrónico presentado, se debe establecer una relación entre el valor de voltaje obtenido por el convertidor y el valor actual de la temperatura.

Supóngase que el valor obtenido por la conversión fue 100. Para obtener el tiempo que duró en cargarse el capacitor en base al valor digital, se utiliza la relación existente entre el periodo del *MTIM* y la lectura del *MTIMCNT* recibida:

$$t = \left(\frac{MTIM_{OVERFLOWTIME}}{255} \right) (MTIMCNT)$$

Como el preescaler se ha elegido en 128, significa que la frecuencia de oscilación de **32kHz** se va a dividir sobre 128, por lo que el tiempo para que exista un desbordamiento en *MTIM* es:

$$MTIM_{OVERFLOWTIME} = \left(\frac{32000}{128} \right)^{-1} = 4.0 \text{ ms}$$

Sustituyendo los valores utilizados en la aplicación se tiene que:

$$t = \left(\frac{0.004s}{255} \right) (100) = 1.6 \text{ ms}$$

El voltaje presente en el nodo de la resistencia NTC y la resistencia de 10.0K está dado por la siguiente ecuación

$$V_o = V_{in} \left(1 - e^{-\frac{t}{RC}} \right)$$

Sustituyendo los valores actuales se obtiene que

$$V_o = 3.3 \left(1 - e^{-\frac{-1.6 \times 10^{-3}}{(10^4)(0.1 \times 10^{-6})}} \right) = 2.6V$$

Teniendo el valor de voltaje en la terminal, se debe obtener la temperatura en base a una relación con la resistencia actual del termistor.

La resistencia actual del termistor se puede obtener despejando del divisor de voltaje presente en el nodo de la terminal no inversora. De la relación

$$V_{out} = \frac{R_1 V_{in}}{R_1 + R_T}$$

Se despeja R_T

$$V_{out} (R_1 + R_T) = R_1 V_{in}$$

$$R_1 (V_{out} - V_{in}) = -R_T V_{out}$$

Y se obtiene

$$R_T = \frac{R_1 (V_{in} - V_{out})}{V_{out}}$$

El valor de resistencia actual en el termistor es entonces

$$R_T = \frac{10^4 (3.3 - 2.6)}{2.6} = 2.6 \times 10^3 \Omega$$

Una aproximación bastante precisa para la temperatura en función de la resistencia de un termistor, está dada por la ecuación de Steinhart-Hart [3]:

$$\frac{1}{T} = a + b \ln(R) + c \ln^3(R)$$

Donde a, b y c, son los coeficientes de Steinhart-Hart.

Sin embargo, una ecuación muy utilizada para los cálculos de temperatura en termistores, es una derivación de la ecuación de Steinhart-Hart utilizando el parámetro B , que es cuando se toma $c = 0$.

La ecuación es la siguiente

recibidos del Módulo timer del MC9RS08KA2, dada por la ecuación de parámetro B y las ecuaciones de apoyo presentadas anteriormente. El código es el siguiente:

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \ln\left(\frac{R_T}{R_0}\right)$$

Donde T es la temperatura del ambiente en Kelvins, R_0 es la resistencia a la temperatura T_0 en Kelvins, R es la resistencia a la temperatura del ambiente. El parámetro B se puede obtener experimentalmente o con las curvas R/T de las hojas de datos del fabricante.

A su vez, del número de serie del termistor es posible obtener estos parámetros, como se explica en [4]. Para un B57620-C103-J62, el valor de la constante, según la hoja de datos es

$$\frac{B_{25}}{100} = 4575 \text{ } ^\circ\text{K} \pm 2\%$$

Que es obtenida a partir del número “62” de la serie.

Los parámetros R_0 y T_0 son denotados como los parámetros de referencia. R_0 representa la resistencia que tendría el termistor a $T_R = 25 \text{ } ^\circ\text{C}$. Estos valores se pueden obtener a través de la serie “103”, cuya interpretación se da de la siguiente manera:

$$103 = 10 \times 10^3 \Omega = 10000 \Omega$$

En la ecuación de parámetro B se sustituyen los valores pertinentes y la temperatura en escala Kelvin es:

$$\frac{1}{T} = \frac{1}{273.15 + 25} + \frac{1}{4575} \ln\left(\frac{2.6 \times 10^3}{10^3}\right) = 3.6 \times 10^{-3}$$

$$T = \frac{1}{3.6 \times 10^{-3}} = 280.5 \text{ } ^\circ\text{K}$$

En escala Celsius, esta temperatura es

$$T_C = 280.5 - 273.15 = 7.31 \text{ } ^\circ\text{C}$$

A través de MATLAB® se obtuvo una gráfica de la temperatura del ambiente en función de los bytes

```
r1 = 10e3;
c = 0.1e-6;
ro = 1000;
b = 4575;
tok = 25+273.15;
vin = 3.3;
mtim_of_time = 1/(32000/128);

counter = 1:1:255;

t = mtim_of_time/255 * counter;

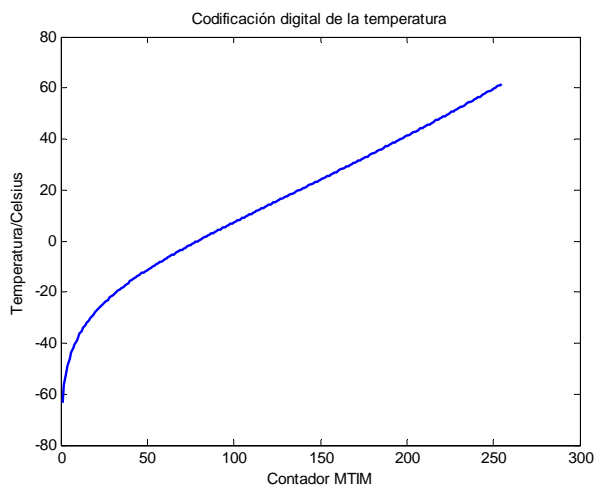
vo = vin*(1-exp(-t/(r*c)));

rt = r1*(vin - vo)./vo;

tk = 1./(1/tok + 1/b*log(rt./ro));

tc = tk - 273.15;

plot(counter, tc, 'LineWidth',2);
xlabel('Contador MTIM');
ylabel('Temperatura/Celsius');
title('Codificación digital de la temperatura');
```



Gráfica 1: Codificación digital de la temperatura con Matlab®

IV. DISEÑO DE SOFTWARE EMBEBIDO

Una vez obtenidas las ecuaciones necesarias para obtener un valor representativo de temperatura, se procede a implementar la aplicación sobre el MC9RS08KA2.

La inicialización de los registros del MC9RS08KA2 se realizó mediante la aplicación gráfica Processor Expert®, la cual es parte de las

herramientas de desarrollo que ofrece la IDE de Codewarrior.

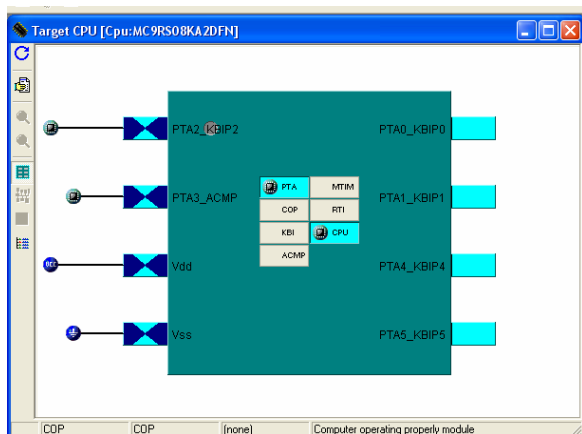


Figura 4: Inicialización del dispositivo mediante el entorno gráfico de Processor Expert®.

El código de inicialización generado para la aplicación se encuentra en el Apartado A del Apéndice.

Dentro de la inicialización, cabe remarcar que se inicializa registro SOPT para poder correr el microcontrolador en *STOP Mode*. Se configura al PTA5 como salida, y se configura su valor con un “1” lógico, ya que como se verá más adelante, el protocolo de transmisión serie debe mantener en estado alto la salida, hasta que se empiece a transmitir una cadena de datos. Se habilitan las interrupciones del comparador analógico y ésta a su vez en *Rising Edge*, pues sólo interesa conocer cuando el voltaje del capacitor rebasa el voltaje en la resistencia NTC. Para más información sobre la inicialización del comparador analógico, referirse a la hoja de datos del MC9RS08KA2, págs. 84-85 [1].

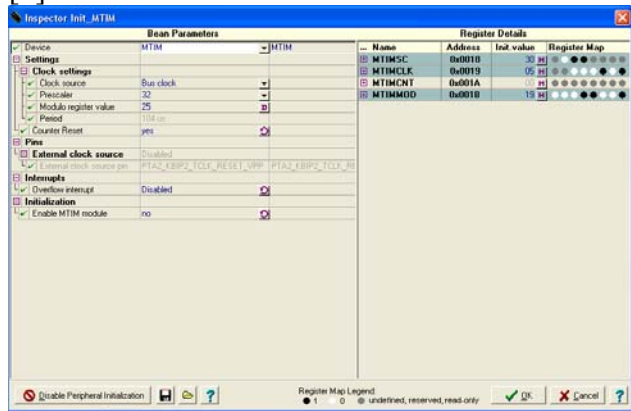


Figura 5: Configuración de MTIM para la comunicación serie, con un periodo de 104 us.

Una vez inicializados los registros necesarios, se procede a efectuar un ciclo del algoritmo, el cual se puede observar en el siguiente esquema:

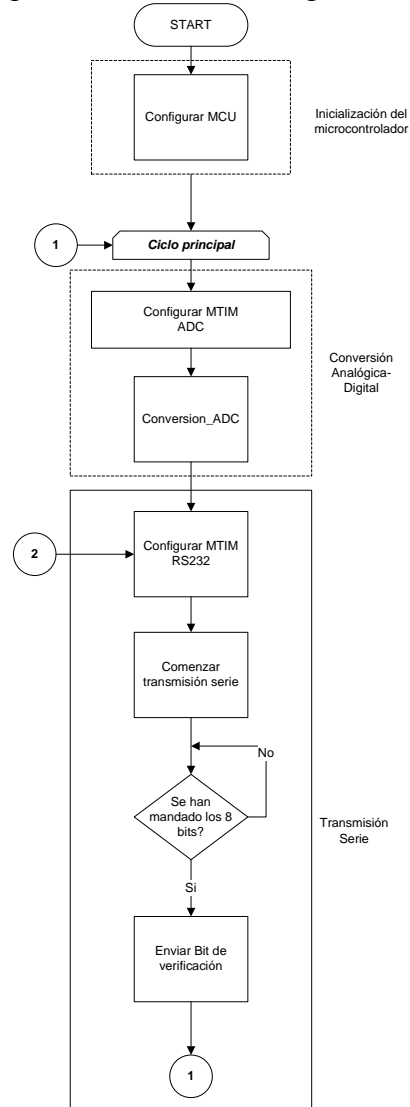


Figura 6: Algoritmo para capturar la temperatura mediante ADC y transmitirla mediante una interface serie

En el ciclo principal se observa que se debe configurar el módulo timer para efectuar la operación de conversión. Más adelante aparece una nueva configuración para la transmisión serie, puesto que ambos módulos trabajan a diferentes frecuencias y bajo distinto modo de interrupción.

El algoritmo de para efectuar una conversión analógica, refiriéndose al módulo Conversion_ADC utilizando el comparador analógico es la siguiente:

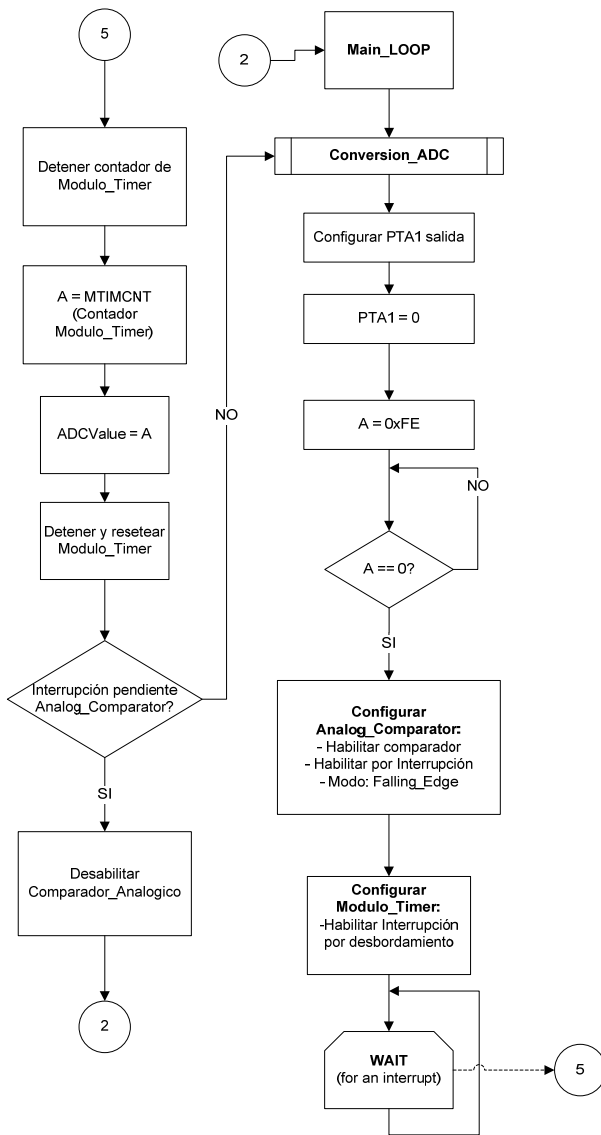


Figura 7: Algoritmo de conversión analógico-digital

Una vez capturado el valor de temperatura por medio del contador de módulo timer, el valor es guardado en algún lugar de la memoria para utilizarse más adelante.

Al finalizar la conversión analógico-digital, se configura el módulo timer para adaptarse al estándar de tiempo de la transmisión RS232, en la cual cada bit debe ser transmitido en un intervalo de 104 µs por bit enviado.

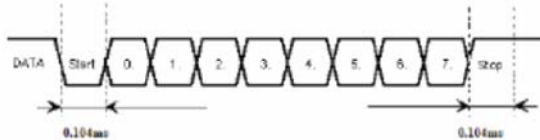


Figura 8: Protocolo de transmisión por serie RS232

Como se aprecia en la figura 8, el puerto de salida debe mantenerse en alto mientras no se esté transfiriendo información. El protocolo de comunicación serie establece que cuando la señal cambia de alto a bajo el dispositivo está indicando que se va a enviar un paquete de datos, por lo que los dispositivos receptores se preparan para recibir los datos.

Para enviar un paquete de 8 bits, es necesario cargar el valor de 8 a un contador, y verificar al terminar cada ciclo hasta que el valor sea cero. Para saber que bit se va a enviar, es necesario declarar una variable de máscara, que en esta aplicación se denomina *roll_bit*, sobre la cual se estará aplicando un corrimiento unitario a la izquierda cada vez que un bit sea transmitido, y ésta debe enmascarar el dato a enviar, con el fin de obtener el n-ésimo bit del byte a enviar.

Una vez enviados los 8 bits, se envía un último bit en nivel lógico de “1” para terminar la comunicación.

El siguiente diagrama muestra el algoritmo de transmisión serie:

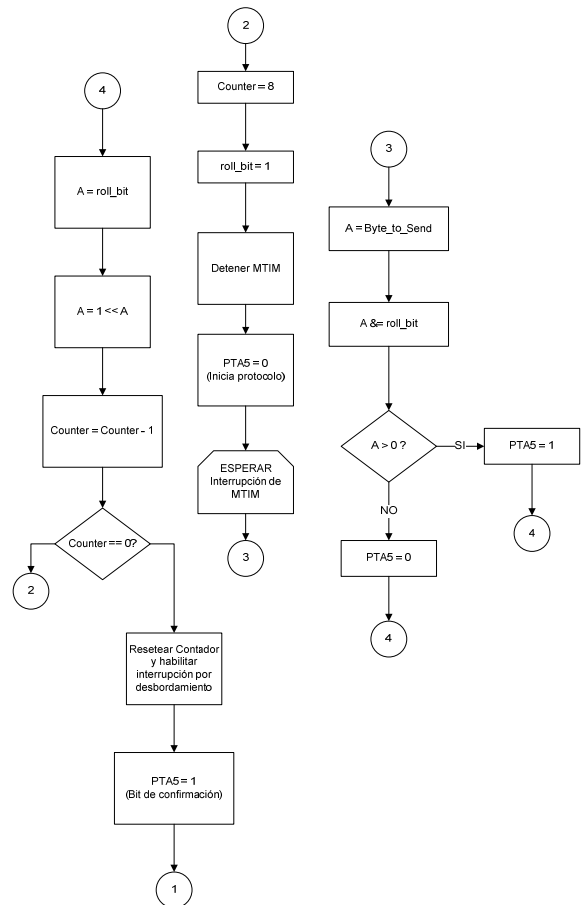
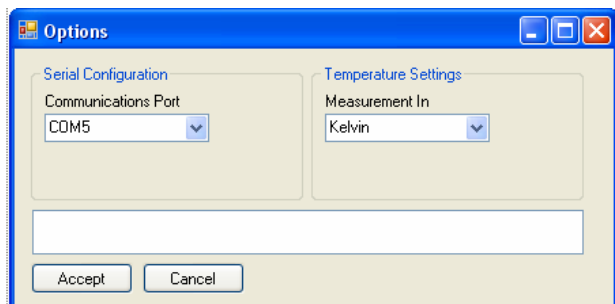


Figura 9: Algoritmo de transmisión serie por el protocolo RS232

V. SOFTWARE DE MONITOREO Y RESULTADOS

Para interpretar los datos enviados por la interface serie en el microcontrolador MC9RS08KA2, se ha escrito una aplicación en C#.NET, y se ha utilizado el componente *SerialPort* que ofrece la API de .NET.

Sobre el componente *SerialPort* se debe ajustar el BaudRate a 9600 y el puerto COM donde se ha conectado el dispositivo. Se puede utilizar una rutina que busque en los puertos COM disponibles hasta que detecte el termómetro, aunque para fines de esta aplicación se ha realizado una pantalla de configuración como la que se muestra en la figura 7c.



(c) Configuración de Opciones

Figura 10: Aplicación de PC en C#.NET

El objeto *SerialPort* contiene un Buffer al cual se puede acceder mediante la instrucción

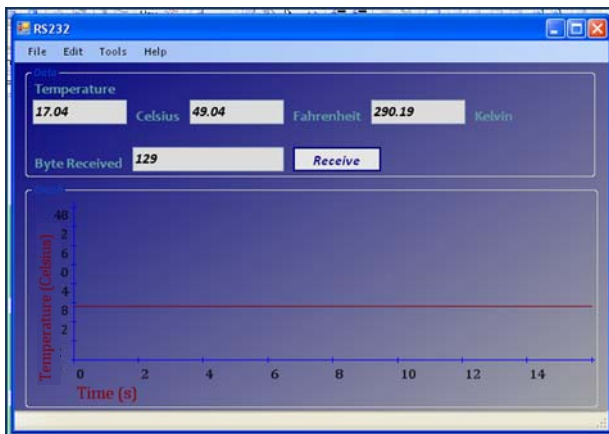
```
int data = serialPort1.ReadByte();
```

La aplicación embebida transmite información continuamente, por lo que el buffer se puede llenar en poco tiempo de mucha información. El método *ReadByte* lee secuencialmente del Buffer, por lo que debe efectuarse hasta encontrar el último valor guardado, ya que si no, se puede llegar a obtener un dato que no esté actualizado. De no utilizar el buffer contenido en el componente *serialPort* es recomendable utilizar una estructura de lista circular, garantizando que los valores nuevos de temperatura reemplacen de manera ordenada a los anteriores, debido a la gran cantidad de datos arrojada por el microcontrolador.

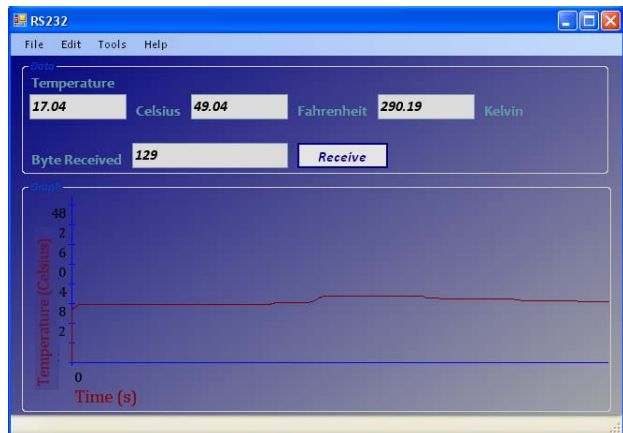
Hasta este momento los datos contenidos en el Buffer no son valores representativos de alguna escala de temperatura, sino es la simple lectura del contador de módulo timer del RS08. Para ello, es necesario aplicar las formulas obtenidas en la sección III, donde se desarrolló la ecuación de Steihart-Hart en relación al valor de la conversión.

VI. CONCLUSIÓN

La aplicación describe a detalle la manera de implementar un termómetro de temperatura que pueda ser monitoreado por una computadora personal por un tiempo indeterminado. La ecuación derivada de la ecuación de Steihart-Hart empleada para obtener la temperatura posee limitaciones de precisión, ya que en la práctica, existen factores como la potencia disipada en el termistor, que



(a) Temperatura constante



(b) Registro de temperatura con variaciones en el tiempo

afectan en poca o gran medida la lectura. En la aplicación se utilizaron resistencias grandes, del orden de Kilo-Ohms como liadoras de corriente, para evitar sobrecargas en el Termistor.

Sin embargo, la ecuación es útil para sistemas sin alta precisión. En esta aplicación, la precisión de las lecturas están limitadas por la resolución en el convertidor analógico, que es de 8 bits. Esta resolución puede ser mejorada si se utiliza una conversión ADC de doble muestreo, utilizando dos arreglos RC.

El protocolo serie implementado en el MC9RS08KA2 es muy sencillo y no posee ningún mecanismo de validación, sino simplemente arroja datos que deben ser interpretados por algún sistema externo.

El sistema de monitoreo de temperatura muestra gráficamente las variaciones de la temperatura registrada, como se puede apreciar en la figura 10b.

Las aplicaciones que pueden darse son variadas, desde ser aplicado en sitios donde se facilita la interface a una computadora personal para obtener un monitor visual, para programación de alarmas bajo establecimiento de parámetros límites, hasta aplicaciones donde se pretende hacer un análisis estadístico de las muestras de temperatura.

ABREVIACIONES

ADC: Convertidor Analógico Digital
 PC: Computadora Personal
 SPI: Serial Peripheral Interface (Interface serie)
 I²C: Inter-Integrated Circuit

APÉNDICE

Apartado A. Diagrama de Circuito Impreso

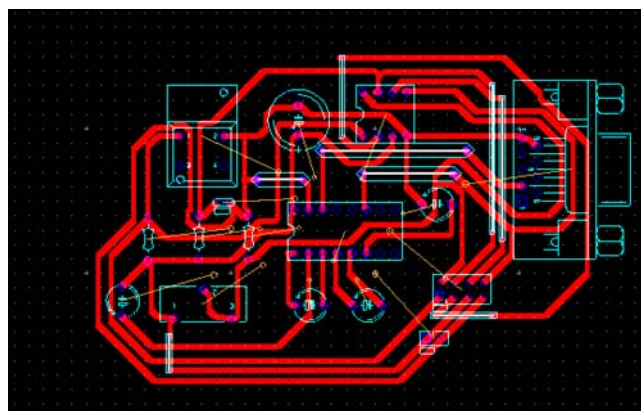


Figura 11: Diagrama de circuito impreso de termómetro digital de temperatura con el MC9RS08KA2.

REFERENCIAS

- [1] Freescale Semiconductor, *MC9RS08KA2 Series Data Sheet*, Rev 3. Freescale™ Semiconductor, Inc, Septiembre 2007, http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC9RS08KA2.pdf
- [2] Freescale Semiconductor, *Implementing an Analog-to-Digital Converter (ADC) on the MC9RS08KA2*. RS08 Peripheral Module Quick Reference, México: RTAC Americas, 2006, http://www.freescale.com/files/microcontrollers/doc/user_guide/RS08QRUG.pdf
- [3] Steinhart, J. S. and S. R. Hart, "Calibration curves for thermistors", *Deep Sea Res.*, 15, 497-503, 1968.
- [4] Epcos, *NTC Thermistors General Technical Information*, Epcos © AG March 2006, http://www.epcos.com/web/generator/Web/Secti ons/ProductCatalog/NonlinearResistors/NTCTh ermistors/PDF/PDF_General_technical_inf ormation,property=Data_en.pdf;/PDF_Gener al_technical_information.pdf
- [5] Freescale Semiconductor, *DEMO9RS08KA2 Schematic and Bill of Material*. Freescale™ Semiconductor, Inc, Septiembre 2006. http://www.freescale.com/files/microcontrollers/hardware_tools/schematics/DEMO9RS08KA2S CH.pdf

CONTROLADOR DE CARGAS ELECTRICAS CON EL PROTOCOLO MODBUS PARA EL CONTROL DE LA DEMANDA MÁXIMA

Ramírez Leyva F. Hugo, Paz Carreño Rodolfo, Ramírez González A. Ernesto

Instituto de Electrónica y Computación

Universidad Tecnológica de la Mixteca (UTM)

Carretera a Acatlima Km. 2.5 Huajuapán de León, Oaxaca, C.P. 69000, Tel. (01953) 5320399 Ext. (200)

Correo electrónico: hugo@nuyoo.utm.mx, fito21091980@hotmail.com, oetxim@mixteco.utm.mx

RESUMEN

El presente artículo describe el diseño, construcción y ejecución de un dispositivo electrónico llamado “controlador de cargas serial”, que se encarga de activar o desactivar sus salidas a relevador en función de los comandos MODBUS que recibe por su puerto de comunicaciones. Las principales características de este dispositivo es la capacidad de controlar hasta 8 cargas, poseer un puerto RS-232, uno RS-485 y uno USB. El sistema fue desarrollado con base en el microcontrolador ATmega32 de la firma ATMEL, y para controlar todas las opciones del dispositivo se diseñó un programa en MATLAB. Este sistema forma parte de un sistema para el control de la demanda máxima en forma distribuida.

Palabras clave: Controlador de cargas, protocolo de comunicaciones MODBUS, demanda máxima.

I. INTRODUCCIÓN

Actualmente la demanda de energía eléctrica es superior a la capacidad de generación en el país, por lo que se han implementado medidas para hacer un uso más eficiente de la misma [1]. Las alternativas existentes son la instalación de nuevas plantas generadoras, o establecer medidas para hacer un buen uso de la energía disponible, que a grandes rasgos consisten en la implementación de dispositivos enfocados al ahorro de energía.

El gobierno federal ha impulsado la segunda alternativa a través de programas que fomenten el consumo más eficiente y racional de la energía eléctrica. Uno de ellos es el Fideicomiso para el

Ahorro de Energía (FIDE), el cual otorga estímulos fiscales a las empresas que emprendan medidas para ahorrar energía [1]. La forma de ahorrar energía, y así disminuir su demanda, es usando equipo más eficiente como: lámparas ahorradoras, motores de alta eficiencia, controladores automáticos de encendido y apagado del alumbrado, etc.

A los usuarios con un consumo mayor a 25 kW, la Comisión Federal de Electricidad (CFE) les factura la energía consumida, el factor de potencia (FP), el derecho de alumbrado público y la demanda máxima.

La demanda máxima es una medida de la forma en la cual el usuario está consumiendo la energía eléctrica. Lo ideal es que el consumo sea constante, pero en casos donde sea irregular, la CFE debe tener la capacidad de suministrar la energía que le es requerida, lo que le implica gastos de operación más altos.

De acuerdo a la CFE [2], “Las Demandas Máximas medidas en los distintos periodos se determinarán mensualmente por medio de instrumentos de medición, que indican la demanda media en kW, durante cualquier intervalo de quince minutos del periodo en el cual el consumo de energía eléctrica sea mayor que en cualquier otro intervalo de quince minutos, en el periodo correspondiente”. Cabe la pena aclarar que la demanda máxima a lo largo del día se divide en tres intervalos para efectos de facturación (base, intermedio y punta) y época del año. Una forma de controlarla es

mediante la instalación de un controlador de demanda máxima.

Un controlador de demanda máxima (CDM) es, de acuerdo a Wildi [4]: “Dispositivo que mide la energía eléctrica que una planta está demandando y predice su comportamiento al final del intervalo de medición y, en caso de ser necesario, apaga las cargas eléctricas que no son prioritarias para la producción, de modo que la demanda máxima no exceda un umbral de consumo preestablecido”.

En el mercado existen varios tipos de CDM comerciales, los cuales adquieren la información del consumo de la planta a partir de las salidas pulsadas KYZ del medidor. En la figura 1 se muestra un CDM típico, el cual recibe las señales KYZ del medidor de potencia, calcula la demanda máxima y, con base en los límites preestablecidos, conecta o desconecta las cargas que controla. También posee un puerto de comunicaciones que se utiliza para enviar la información de las operaciones que ha realizado, así como el registro de energía que ha consumido la planta, o simplemente para configurar sus diferentes parámetros.

La principal desventaja de este tipo de CDM es que todas las cargas que controla, así como las señales KYZ del medidor, deben estar cableadas. Lo que en ocasiones resulta complicado, además de incrementar los gastos de instalación. Una alternativa a esta configuración, es un CDM distribuido, el cual con un enlace de comunicaciones, que puede ser alámbrico o inalámbrico, realiza las diferentes funciones del controlador.

En la figura 2 se muestra el diagrama a bloques del CDM distribuido propuesto, el cual está constituido por: La Estación Maestra, el Codificador de Pulsos KYZ, el Controlador de Cargas y el bus de campo.

El codificador de Pulsos KYZ se conecta a las salidas pulsadas del medidor de CFE, lleva la cuenta del número de pulsos y su duración, y envía esta información al controlador de demanda para

que haga las estimaciones del consumo de energía. Posee un puerto de comunicaciones RS-232 y RS-485 para que pueda ser conectado a una bus de campo [5].

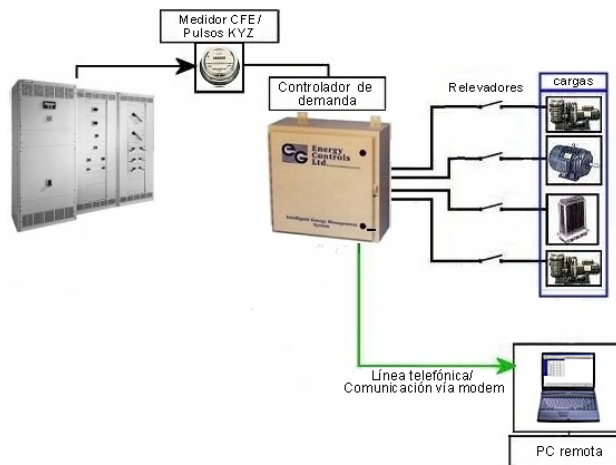


Fig. 1. Diagrama a bloques de un controlador de demanda típico

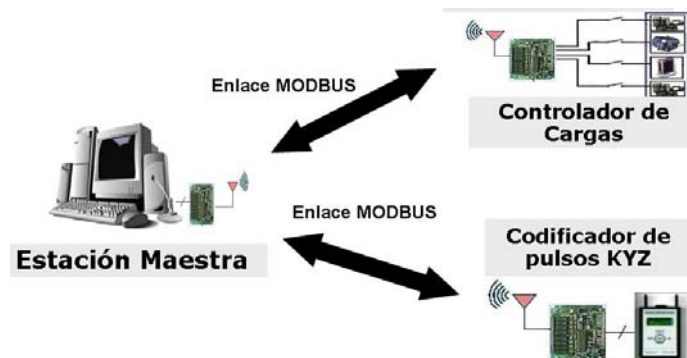


Fig. 2. Controlador de demanda máxima distribuido

La Estación Maestra (en fase de desarrollo), es en sí el CDM y está basada en una computadora personal. Se conecta a los demás módulos por una interfaz de comunicaciones, que puede ser alámbrica o inalámbrica. Con base en la información del codificador de pulsos, estima la demanda máxima y cuando determina que se debe apagar alguna carga, le envía la orden al controlador de cargas.

El control de cargas activa y desactiva sus salidas digitales en función de las órdenes que recibe del controlador de demanda máxima. De tal manera que apaga o enciende las cargas eléctricas que

tenga conectadas. Al igual que el codificador de pulsos, posee un puerto de comunicaciones para que se pueda conectar a un bus de campo o una computadora portátil.

El bus de campo es una denominación que se le ha dado a los diferentes protocolos de comunicaciones de datos, para redes industriales, que existen en el mercado. Entre los más importantes se encuentran Profibus, CAN, DeviceNet, MODBUS, Fieldbus Foundation, HART, DNP 3 y Ethernet [6].

Se escogió trabajar con MODBUS por que se tiene experiencia en el desarrollo de aplicaciones basadas en él, y porque la interfaz eléctrica es muy simple de realizar con microcontroladores.

Este protocolo es un estándar que ha sido ampliamente aceptado en redes industriales por ser robusto y de fácil uso, existiendo en el mercado una gran cantidad de dispositivos basados en él. Sus principales características son: Arquitectura maestro-esclavo, capacidad de manejar hasta 255 esclavos, y manejo de los errores por codificación CRC de 16 bits [7, 8].

En la figura 3 se muestra la estructura de una trama MODBUS, la cual consta de los campos: Dirección del Esclavo con el que establece comunicación el Maestro; Código de la función que el Esclavo debe ejecutar; Datos necesarios para ejecutar la función (si los requiere); campo de comprobación de error en la trama.

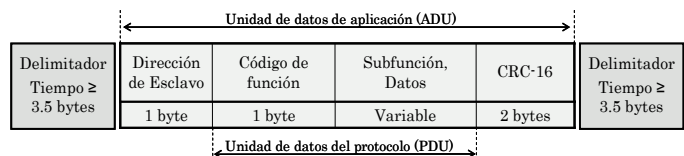


Fig. 3. Estructura general de una trama MODBUS.

El tema principal de este trabajo es la descripción de la forma en que fue realizado el control de cargas. Éste funciona como un esclavo en una red MODBUS, por tal motivo se le dotó de interfaces de comunicación serial. Para probarlo se desarrolló un programa para computadora en MATLAB [9], con una interfaz gráfica, desde el cual se le envían

las tramas MODBUS que activan o desactivan sus salidas.

II. HARDWARE DEL CONTROLADOR DE CARGAS

El controlador de cargas, que denominaremos CDECA, tiene capacidad para manejar 8 salidas a relevador, con las cuales controla la activación o desactivación de las cargas eléctricas a las que se conecta. La apertura o cierre de estas salidas es en función de la trama MODBUS que recibe del controlador. Las principales características del CDECA son:

- 8 salidas a relevador, capaces de manejar cargas de DC o AC de hasta 10 A.
- Soporte de comunicación bajo el Protocolo MODBUS.
- Puerto de comunicaciones RS-485 para recibir y transmitir tramas seriales en una red MODBUS multipunto.
- Módulo convertidor de USB y RS-232 a TTL para configurar los parámetros del controlador de cargas. Este módulo es independiente, para que pueda ser usado en otras aplicaciones.
- Sistema con base en un microcontrolador de la firma ATMEL, programado completamente en lenguaje C para facilitar las actualizaciones del software.

En la figura 4 se muestra un diagrama a bloques del CDECA, el cual, de forma funcional se divide en 3 módulos que son: Módulo de comunicaciones, Módulo de control, Módulo de carga.

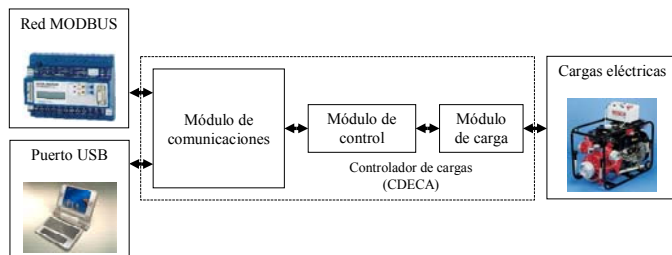


Fig. 4. Diagrama a bloques del controlador de cargas (CDECA)

La alimentación de CDECA es suministrada por un eliminador de pilas externo con voltaje de salida entre 4.75V y 5.25V. Se decidió que fuera de esta forma para reducir el tamaño del sistema.

Módulo de carga

Este módulo se encarga de controlar la activación y desactivación de las cargas, es decir, encender y apagar los relevadores de salida de acuerdo a las órdenes recibidas por el dispositivo maestro. Los relevadores pueden conmutar cargas de DC de hasta 240 volts o cargas AC de hasta 120 volts.

La activación del relevador es realizada por las salidas digitales del microcontrolador. Proporcionan voltajes con niveles TTL, con una corriente de salida máxima de 20 mA. Por tal motivo se utiliza un transistor BC547 como interruptor para incrementar la capacidad de corriente.

Se tienen 8 bloques de este tipo, uno para cada una de las 8 salidas con las que cuenta el sistema. Para hacerlo más práctico, se dividió esta etapa en 2 módulos de 4 salidas cada uno, donde uno de ellos es desmontable con 4 salidas (figura 5) y las 4 salidas restantes se tienen con la demás circuitería.



Fig. 5. Fotografía del módulo de cargas desmontable

Como se puede ver en la figura 5, los cuadros amarillos corresponden a los 4 relevadores, en la

parte izquierda están los transistores para el disparo. Las terminales corresponden a las salidas del relevador y la señal de disparo.

Módulo de control

Este módulo se encarga de recibir las tramas seriales MODBUS que vienen del módulo de comunicaciones, decodificarlas y si son válidas, ejecutar la tarea solicitada. También envía las señales de disparo que activan a cada una de las 8 salidas del módulo de cargas. Para preservar a los registros mantenidos, especificados por MODBUS, debe contar con una memoria no volátil. Además, debe tener suficiente memoria de programa para que sea programado en C.

Para lograr los requerimientos planteados, el circuito integrado que cumple con ellos es un microcontrolador. En nuestro caso, el microcontrolador que se usó fue el ATmega32 de la firma ATMEL [12]. Se seleccionó este micro por ser muy rápido, contar con suficiente memoria de programa (32 kbytes) y porque se tiene experiencia en el desarrollo de aplicaciones basadas en él. Sus principales características son:

- 32 registros de propósito general, cada uno de 8 bits.
- La gran mayoría de instrucciones se realizan en un solo ciclo de reloj.
- 32 kbytes de memoria flash, 1024 bytes de memoria EEPROM y 2 kbytes de memoria SRAM.
- 2 timer/counters de 8 bits.
- Watchdog timer programable.
- 4 puertos, cada uno con 8 líneas de entrada/salida.
- Voltaje de alimentación de 2.7 a 5.5 volts.
- 44 terminales con encapsulado TQFP.

El ATmega32 puede operar hasta 16 MIPS con un oscilador de 16MHz. Sin embargo, se decidió trabajar con un cristal de 7.3728 MHz para alcanzar los escalamientos en frecuencia requeridos para generar el baud rate en el rango de 1200 bps hasta 230.4 kbits/s. En la parte de

entada/salida se tiene 32 pines y solo se ocupan 10. Sin embargo, los 32 Kbytes de memoria de programa garantizan que se tenga suficiente memoria para almacenar el programa.

Se le implementó un puerto, compatible con los programadores comerciales de la firma ATMEL, para que se actualice el firmware. Como existen variantes de programadores que requieren alimentación o no, se dejó un jumper con el que se selecciona la opción. Se recomienda leer las hojas de especificaciones del programador que se utilizará para evitar posibles daños al programador o al microcontrolador.

Módulo de comunicaciones

Esta etapa se encarga de acondicionar las señales eléctricas para que se pueda establecer el intercambio de datos entre el CDECA y un dispositivo maestro. La comunicación alamburada se lleva a cabo por medio de los puertos de comunicaciones seriales USB y RS-485 de 2 ó 4 hilos.

En la figura 6 se muestra un diagrama a bloques con las partes que conforman al módulo de comunicaciones. Como se puede apreciar, está integrado por 3 secciones que son: el puerto RS-485, el puerto USB y el selector del puerto (interruptor). Esta última parte es necesaria, ya que el microcontrolador sólo tiene una salida de transmisión (TX) y recepción de datos (RX). En las siguientes secciones se explicarán cada una de ellas.

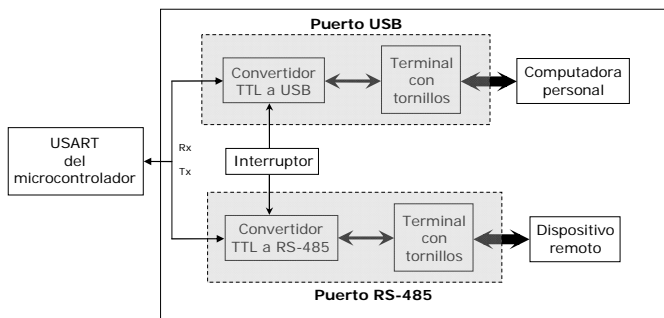


Fig. 6. Diagrama a bloques del módulo de comunicaciones

Para adaptar los niveles TTL del microcontrolador y generar las señales del puerto RS-485 de 2 ó 4 hilos siguiendo la norma EIA/TIA-485, se usa el CI MAX489 de la firma MAXIM [11]. Este CI proporciona una comunicación de tipo full-duplex, en la cual se pueden transmitir y recibir datos al mismo tiempo a una velocidad de hasta 250kbts/s. Permite conectar hasta 32 transeptores de su mismo tipo en el BUS, en formato de 2 hilos (half-duplex) y 4 hilos (full duplex). En una red MODBUS, se utiliza half-duplex.

En la figura 7 se muestra la fotografía del CDECA. En ella se pueden ver los 4 relevadores para controlar al mismo número de cargas. En la parte superior izquierda están las terminales del puerto RS-485 en configuración half-duplex. En la figura 7, mostrada previamente, se ve la parte posterior, en la cual se encuentra el microcontrolador.

En la figura 8 se muestra el diagrama esquemático de todo el controlador de cargas. La parte encuadrada corresponde al módulo desmontable que controla 4 cargas.



Fig. 7. Fotografía del CDECA

Se implementó un convertidor de RS-232 y USB a TTL, el cual se denominará ComUSB. Es un módulo independiente y desmontable con el fin de que pueda ser utilizado en otras aplicaciones, además de que estas interfaces son las que se utilizan cuando se conecta a una computadora personal. Para que funcione, requiere un voltaje entre 4.4 y 5.25V, el cual se le suministra mediante una fuente de voltaje externa. En las figuras 9 y 10 se muestran el diagrama esquemático del ComUSB

y la vista superior del módulo terminado, respectivamente.

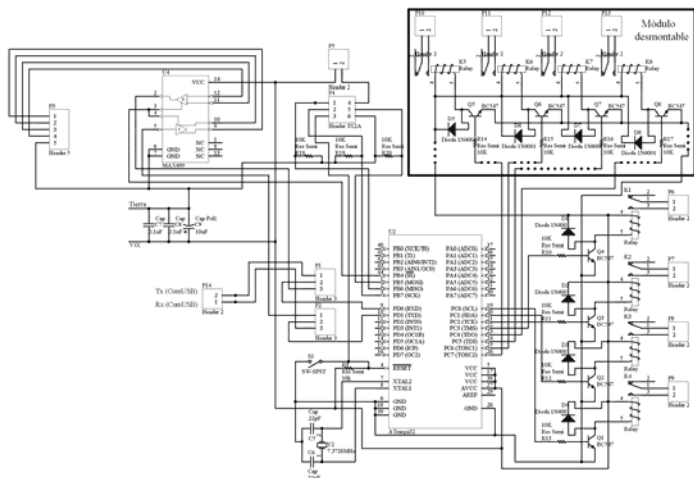


Fig 8. Diagrama del Controlador de Cargas

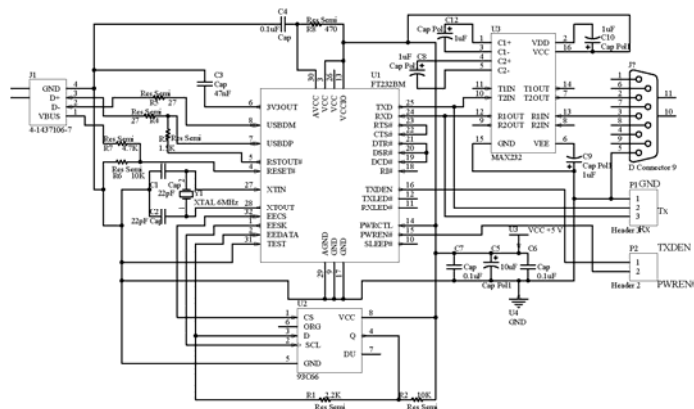


Fig. 9. Diagrama esquemático del ComUSB



Fig. 10. Vista superior del ComUSB

El puerto RS-232 fue implementado con el circuito integrado MAX-232, que cumple con todas las características eléctricas de esta interfaz. Para cubrir con la parte mecánica, se utilizó un conector DB9 para impreso, el cual se muestra en la parte

inferior de la figura 10. Se decidió incluirlo por que en el área de instrumentación electrónica todavía se sigue usando, además de que permite interconectarlo a radio modems para transmisiones inalámbricas.

El puerto USB transfiere información serial desde y hacia una computadora a una velocidad mucho mayor que el RS-232 o el RS-485. Sin embargo su manejo es mucho más complejo que los 2 anteriores, ya que involucra el manejo de señales de sincronización, manejo de errores (mecanismos de recuperación de datos ante algún fallo), inserción dinámica de dispositivos, soporte para la identificación de dispositivos defectuosos, etc. También su estándar especifica el tipo de conector a usar y la forma en que se transmiten las tramas.

Para que una computadora personal (PC) reconozca a un dispositivo conectado al puerto USB, se requiere desarrollar un programa especial, el cual servirá de intermediario entre un dispositivo de hardware y el sistema operativo. A este programa se le conoce como driver o controlador. El desarrollo de un controlador de USB es bastante complejo, ya que es necesario un conocimiento profundo del sistema operativo y el hardware.

En el mercado existen varios fabricantes de circuitos integrados que han diseñado manejadores de USB y proporcionan el driver para que sea usado su dispositivo con diferentes sistemas operativos. Uno de estos dispositivos es el FT232BM de la firma FTDI (Future Technology Devices Intl.) [10]. Se escogió este módulo por que es la base de muchos manejadores de USB que existen en el mercado. Además de que es fácil de conseguir en México.

Para el sistema operativo Windows®, una vez que se instala el driver, el sistema operativo lo reconoce como un puerto de comunicaciones de tipo COM, facilitando el desarrollo de aplicaciones con cualquier lenguaje de programación.

El FT232BM requiere de un reloj a 6MHz, por tal motivo se le puso un cristal a esa frecuencia. Internamente tiene preescaladores de frecuencia

con lo que consigue generar señales de reloj de 12MHz y 48MHz. Sus salidas del módulo marcado como UART poseen todas las terminales de un manejador de comunicaciones seriales convencional.

La velocidad de trasferencia de datos del FT232BM hacia la computadora es a 12 Mbits/s (modo full speed), mientras que en la conexión con el dispositivo final la velocidad es configurable en el rango de 1.2kbits/s hasta 120kbits/s. Para compensar estas diferencias de velocidades, el CI tiene implementados buffers para almacenar los bytes que se reciben o envían.

III. SOFTWARE DEL CODIFICADOR DE PULSOS

El programa que controla al microcontrolador ATmega32, es el que se encarga de coordinar todas las tareas que éste ejecuta. Siendo las más importantes, la implementación del protocolo MODBUS y el control del módulo de control. El conjunto de tareas que desarrolla son:

- Recepción de la trama MODBUS y decodificación del mensaje.
- Actualización de las salidas en función de la petición realizada.
- Manejo de los contadores de eventos.
- Ejecución de la función solicitada.
- Construcción de la respuesta MODBUS en función de la petición realizada por el maestro.
- Construcción de la respuesta MODBUS de excepción, en caso de que el dispositivo no soporte la función solicitada u ocurra un error de lectura/escritura de un registro.

Todo el programa se desarrolló en C, utilizando el compilador WinAVR en su versión 20070122. En la figura 11 se muestra el diagrama de flujo del programa. Lo primero que realiza es configurar los parámetros de los pines de entrada/salida (función configurar_puertos) y los parámetros de comunicaciones (función configurar_USART).

La función configurar_temporizador, es la que se encargará de configurar los parámetros necesarios para el temporizador (Timer Counter 2 del microcontrolador), la cual sirve como base para medir los tiempos de espera especificados por MODBUS. Las velocidades que soporta son: 2.4, 4.8, 9.6, 19.2, 38.4, 57.6, 115.2, 230.4 kbits/s.

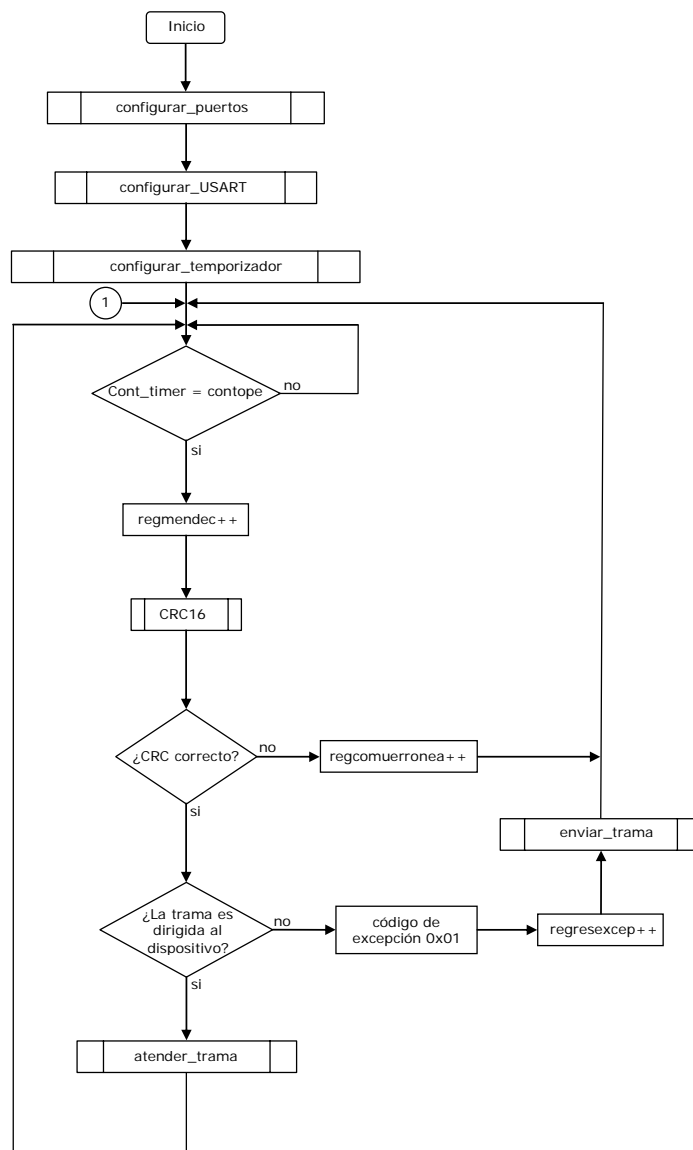


Fig. 11. Diagrama de flujo del firmware del CDECA

Una vez terminada la configuración, entra en un ciclo infinito, el cual está a la espera de recibir una trama MODBUS. En caso de que llegue alguna trama válida, verifica el código de errores y comprueba si es un mensaje para él. Cuando el

mensaje es para él, atiende la petición y envía la respuesta. En caso contrario la ignora y vuelve a su estado de espera. En ambos casos incrementa el contador de tramas.

Protocolo de comunicaciones

El protocolo de comunicaciones MODBUS especifica registros mantenidos que almacenan información del sistema que no debe perderse al apagar el dispositivo. La dirección física de la memoria RAM no corresponde con la dirección que maneja el protocolo. Para el CDECA, del registro 30001 al 30008 se encuentran los estados de las salidas de los relevadores. En las direcciones 40001, 40002 y 40003, se tiene el registro de la velocidad de transmisión, paridad y dirección del esclavo, repectivamente.

Dentro de las funciones MODBUS que se implementaron en el sistema, están la de lectura del estado de las bobinas y escritura de los registros mantenidos (0x01H y 0x10H), diagnóstico y lectura de contador de eventos (0x08H a 0x0BH), e identificación del dispositivo.

IV. PROGRAMA DE CONTROL DEL CDECA

Para verificar el correcto funcionamiento del CDECA, se creó un programa para computadora con base en el lenguaje de programación MATLAB®. Este lenguaje normalmente es usado para realizar cálculos de índole científico, ya que cuenta con funciones para obtener una solución más rápida de problemas matemáticos. Por fortuna, las últimas versiones permiten tener accesos a puertos de comunicaciones, en particular a los de tipo COM. Las características más importantes del programa son:

- Comunicación a través del puerto RS-232 y USB.
- Menú de configuración desde el cual se establecen las opciones de comunicación.
- Menú de configuración donde se seleccionan las opciones del protocolo.

- Recepción y transmisión de las tramas MODBUS y despliegue del estado de las cargas.
- Ventana de errores.
- Detección de los puertos seriales disponibles en la PC.

Para cumplir con los requerimientos, el programa se dividió según la estructura jerárquica mostrada en la figura 12. En forma funcional, el programa se divide en tres módulos que son: Selección de opción, Procesamiento y Reporte.

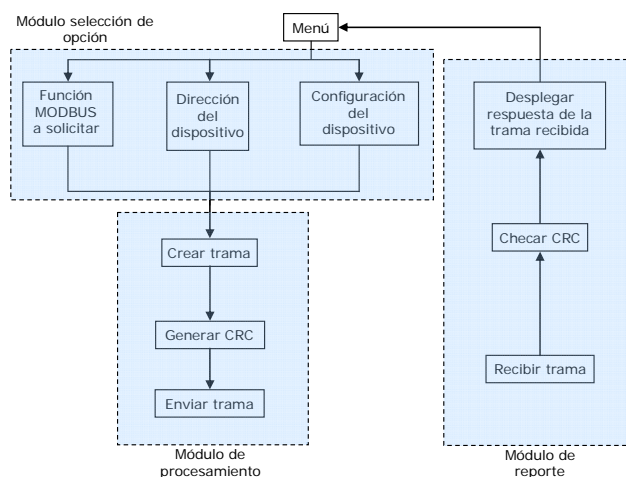


Fig. 12. Diagrama de flujo del programa implementado en el CDECA

El módulo de Selección de opción, es la parte en la cual se definen los parámetros de comunicaciones del esclavo con el que se va a establecer comunicación, la velocidad de transmisión y la paridad, así como también la función MODBUS a ejecutar. El módulo de procesamiento elabora y envía la trama MODBUS por el puerto serial, con los datos configurados en el módulo Selección de opción. El módulo de reporte es el responsable de obtener los datos recibidos del CDECA, que envía como respuesta, y mostrar al usuario el resultado de forma gráfica.

En la figura 13 se muestra la interfaz de usuario del programa, elaborada usando la herramienta que proporciona el MATLAB para este fin.

V. PRUEBAS Y RESULTADOS

Por ser este sistema una parte del controlador de demanda distribuido, y al estar éste en fase de desarrollo, aún no se puede probar en una red eléctrica. Por tal motivo se probó en forma independiente, usando una carga eléctrica de corriente alterna.

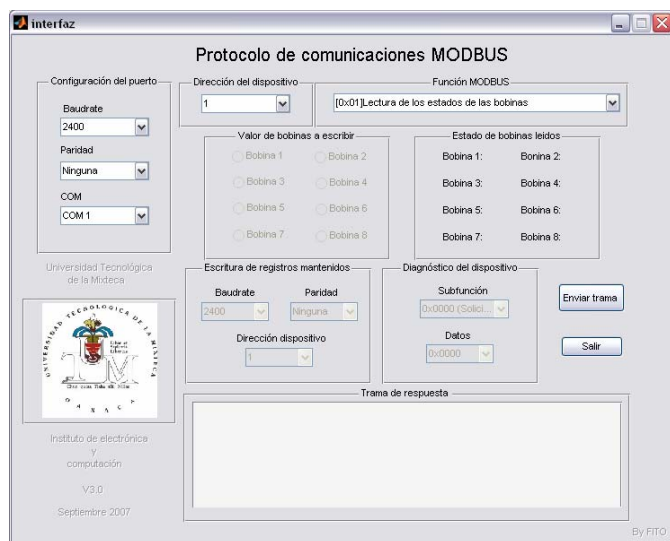


Fig. 13. Interfaz gráfica de usuario del CDECA

En general, una vez instalado el dispositivo, su funcionamiento es relativamente simple; se envía la trama MODBUS con la petición que se desea ejecutar, recibe la trama, la decodifica y en caso de que sea correcta la petición, la ejecuta y envía la respuesta. Por tal motivo sólo se realizaron 3 tipos de pruebas: Enlace de comunicaciones y tasa de error, Apertura y cierre del CDECA y Enlace RS-485.

La prueba de enlace de comunicaciones se hizo generando una trama de 255 datos, los cuales se enviaron y recibieron 1000 veces a las diferentes velocidades que maneja el CDECA. El resultado obtenido fue de una tasa de error del 0%, es decir todas las tramas fueron recibidas sin error. La prueba del enlace RS-485, se hizo conectado el controlador a la computadora, a través de un cable de 300m, el resultado de la prueba fue que el dispositivo funcionó adecuadamente.

Finalmente, le prueba de apertura y cierre se realizó usando un foco como carga, como se muestra en la figura 14. Usando el programa de control se ordenó apagar y encender el foco varias veces, ejecutándose sin ningún problema.

VI. CONCLUSIONES

Se ha presentado un sistema que controla la apertura o cierre de cargas eléctricas, con el fin de desarrollar un controlador de demanda máxima en forma distribuida con el protocolo de comunicaciones MODBUS. Además, se mostró la forma en la cual se implementó un manejador de comunicaciones seriales para USB.

El programa para la computadora, desarrollado en MATLAB, y cuya finalidad fue la de probar el rendimiento del sistema en aplicaciones que se comuniquen con instrumentos, tuvo como inconveniente el quedarse bloqueado, y suponemos que se debe a que la interfaz gráfica demanda demasiados recursos que hacen que el sistema se sature.

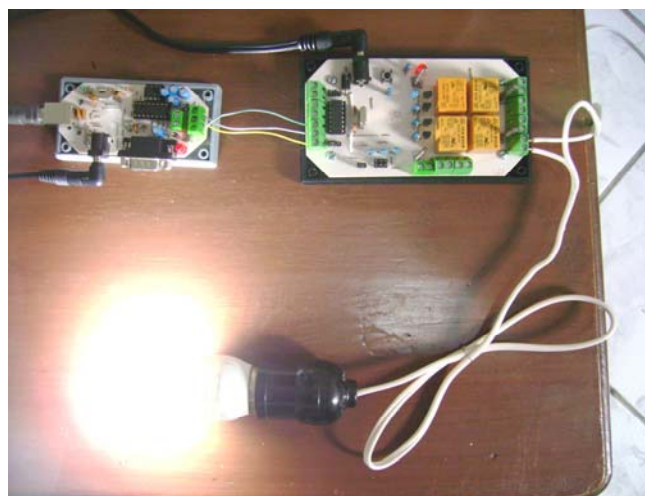


Fig. 14. Prueba de apertura y cierre del CDECA

El funcionamiento del ComUSB fue mejor de lo esperado, ya que al realizar las pruebas y obtener el porcentaje de error se observó que este es del 0%. Hay que aclarar que esta cifra se obtuvo después

de una serie de pruebas realizadas utilizando el lenguaje MATLAB® y el ComUSB.

Como posibles trabajos futuros, se pretende terminar las demás partes del controlador de demanda distribuido, experimentar con interfaces Ethernet para que el sistema pueda ser conectado a una red de Área Local, y trabajar con X10, por la factibilidad de usar los mismos cables para transmitir la orden de apertura y cierre usando el mismo cableado con que se alimentan los equipos.

VII. REFERENCIAS

- [1] “Medidas para el ahorro de energía” Fideicomiso para el Ahorro de Energía, en http://www.fide.org.mx/el_fide/resultados.html.
- [2] “Demanda Máxima Medida”, Comisión Federal de Electricidad, en <http://www.cfe.gob.mx>.
- [3] Santiago, F. y Ramírez, F.: “Diseño y Construcción de un Simulador de Medidor de Energía Eléctrica para Controladores de Demanda Máxima”. Memorias SOMI XIV: Congreso de Instrumentación, Puebla, México, Octubre de 1999.
- [4] Wildi, T.: Electrical Machines, Drives and Power Systems. Prentice Hall, USA, 2002.
- [5] Ramírez Leyva F. Hugo, Aparicio Velásquez Enmanuel, Ramírez González A. Ernesto: “Codificador de Pulsos para el Registro del consumo de Energía Eléctrica y Demanda Máxima con el Protocolo Modbus”. Memorias del 3ª semana nacional de Ingeniería Electrónica SENIE07, pagina 30-39. Octubre 2007.
- [6] Mariño Espiñeira P.: Las Comunicaciones en la Empresa: Normas, Redes y Servicios, Madrid, RA-MA, España 1995.
- [7] MODBUS-IDA.ORG: MODBUS Application Protocol Specification V1.1, USA, 2002.
- [8] MODBUS-IDA.ORG: MODBUS over Serial Line (V1) Specification and Implementation Guide V1.0, USA, 2002.
- [9] “Descripción de los Toolboxes de Matlab®”, Pagina de Mathworks en: <http://www.mathworks.com/products/distri/btb>
- [10] FT232BM USB UART (USB – Serial) I.C; V1.8; Future Technology Devices International Ltd.; USA; 2005
- [11] “Convertidor de TTL a RS-485”, Fabricante MAXIM, en <http://www.maxim-ic.com>
- [12] Atmel, ” 8-bit Microcontroller with 32K Bytes In-System Programmable Flash”, Atmel, No de parte 2503M-AVR-05/08.

SISTEMA DE MONITOREO POR RADIOFRECUENCIA MEDIANTE EL USO DE UNA PDA

G. E. Aranda Castillo, J. M. Luna Rivera

Departamento de Electrónica, Facultad de Ciencias
Universidad Autónoma de San Luis Potosí
Av. Salvador Nava Mtz. s/n, Zona Universitaria, San Luis Potosí, S.L.P., CP. 78290, México
Tel. +52 (444) 8262387, correo electrónico: mlr@fciencias.uaslp.mx

RESUMEN

En este artículo se propone un sistema para el monitoreo de máquinas en las líneas de producción basado en los recursos que dispone una PDA. La elección en usar una PDA corresponde principalmente a sus características como son el tamaño, flexibilidad y versatilidad en el intercambio de información con otros dispositivos de monitoreo tradicional.

Palabras clave: Monitoreo por Radiofrecuencia, PDA, Bluetooth, LabVIEW.

I. INTRODUCCION

En la actualidad las empresas participan en un mercado cada vez más global, en donde la fabricación de productos en grandes cantidades y con tiempos limitados resulta un proceso crítico en el sector industrial. Para satisfacer estas demandas, es vital el desarrollo de nuevas estrategias de monitoreo a las líneas de producción con el propósito de ayudar a reducir los tiempos de paro en la producción y los gastos de mantenimiento. Adicionalmente estas estrategias pueden contribuir a la reducción de los riesgos de accidentes para el personal responsable de su uso y mantenimiento.

Tres de las desventajas principales en los procesos de mantenimiento actuales dentro de la industria son:

- Pérdida en la producción por paro de máquinas.
- Alto costo de los mecanismos de mantenimiento.
- Suspensión de labores para el personal que opera las máquinas.

De esta manera, uno de los grandes retos en las empresas es entonces el establecimiento de un programa de mantenimiento eficiente y oportuno que evite contratiempos en la producción al mantener los equipos que conforman una línea de producción en buen funcionamiento. En este contexto, este artículo propone un sistema de monitoreo basado en una plataforma PDA cuya comunicación se logra a través de la tecnología Bluetooth. Este sistema tiene el propósito de ofrecer al usuario una alternativa para la verificación de eventos en equipo o dispositivos electrónicos de manera continua y sin la necesidad de suspender su funcionamiento. El enfoque de este trabajo es sustituir cualquier tipo de equipo de medición que requiera una conexión física en el proceso de monitoreo, y cambiarlo por una PDA la cual representa una unidad móvil y de pequeñas dimensiones. Para llevar a cabo la comunicación se utilizará la tecnología Bluetooth la cual ofrece ventajas muy particulares gracias a su estandarización en un gran número de dispositivos [1], [2]. La principal contribución de este trabajo radica en la integración de la PDA al proceso de monitoreo por radiofrecuencia con el propósito de ofrecer un

diagnóstico oportuno en el proceso de mantenimiento de equipo en una línea de producción. La PDA puede ofrecer capacidades similares a la versatilidad de una PC de escritorio con la ventaja de la movilidad y un ambiente gráfico para la interpretación de los datos. El sistema propuesto será validado de forma experimental en la aplicación de monitoreando de un motor trifásico.

II. SISTEMA DE MONITOREO Y ADQUISICION DE DATOS

La Figura 1 muestra el diagrama a bloques del sistema de monitoreo y adquisición de datos que se propone en este trabajo. El sistema se compone de 4 etapas: 1) Fuente de Información, 2) Unidad de Adquisición y Procesamiento de Datos, 3) Enlace por Radiofrecuencia, y 4) Unidad de Monitoreo Remoto (PDA).

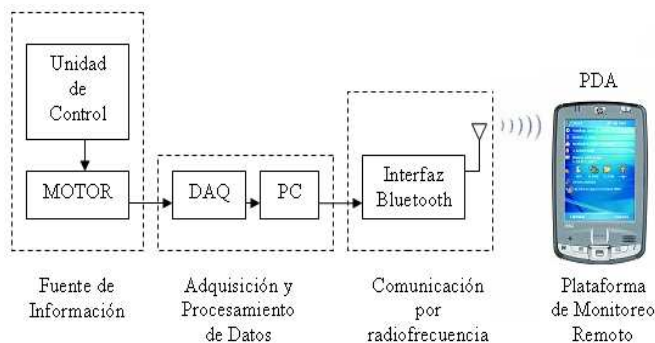


Figura 1: Sistema de monitoreo por radiofrecuencia para el diagnóstico de fallas en equipo industrial.

II.A. Fuente de Información

La primera etapa consiste de la fuente de información que en este caso representa el equipo o dispositivo a monitorear. El sistema aquí propuesto incluye como ejemplo de aplicación el monitoreo y adquisición de datos de un motor trifásico cuyas características se presentan en la Tabla 1. El sistema de monitoreo incluye un actuador de potencia (inversor trifásico) [3] necesario para el accionamiento

del motor de inducción. El objetivo del inversor trifásico es proveer un voltaje de magnitud y frecuencia variable para alimentar el motor, y

Tabla 1. Características del motor trifásico SIEMENS usado para el monitoreo.

Parámetros	Valores
Serie	1LA2
Tensión	220/380 V
Potencia	0.25 kW
Horse power	1/4 HP
Revoluciones/minuto	1340 rpm

así poder modificar la velocidad de rotación y el par eléctrico producido. Este dispositivo consiste de 3 bloques: control, aislamiento y potencia [3].

El bloque de control genera los pulsos para disparar los IGBTs de la etapa de potencia. La generación de estos pulsos depende de dos potenciómetros (tensión y frecuencia de alimentación) que deben ser operados manualmente para poder controlar al motor. Las señales de conmutación son generadas bajo un esquema de modulación espacio-vectorial en un microcontrolador PIC-16F819.

El segundo bloque realiza la tarea de aislamiento de los patrones de conmutación generados por la etapa de control. Este aislamiento se realiza por medio de optoacopladores que proveen un aislamiento óptico y evitan dañar esta etapa en caso de un corto o falla dentro del inversor de frecuencia.

Por último, existe el bloque de censado de las variables eléctricas del motor: corrientes de línea y voltajes fase-fase. Las corrientes son medidas por medio de una resistencia shunt de bajo valor (0.01Ω) y un amplificador operacional aislado HCPL-7840. De manera que se tiene un voltaje proporcional a la corriente que circula por la resistencia. Estos mismos integrados se utilizan para censar los voltajes fase-fase utilizando un divisor de tensión.

II.B. Unidad de Adquisición y Procesamiento de Datos

Esta segunda etapa tiene la tarea de procesar la información y proveer la interfase para su transmisión por radiofrecuencia. Para llevar a cabo dicha tarea, la señal proveniente de la fuente de información (motor trifásico) debe ser primero digitalizada para después ser procesada en la PC. En el sistema utilizado, se emplea el dispositivo *DAQ NI-6211* de *National Instruments* cuyas características se presentan en la Tabla 2 [4].

Tabla 2. Características del DAQ NI-6211.

Características	Valores
Entradas analógicas	32 con representación en 16 bits
Muestras por segundo	250kS/s
Salidas analógicas	2 con representación en 16 bits
Líneas digitales TTL/CMOS	16 líneas

Es importante mencionar que por facilidad y para propósitos experimentales, el sistema propuesto incluye una PC como interfase entre la unidad de adquisición y el dispositivo de comunicación Bluetooth. Sin embargo, una solución más conveniente y práctica sería la utilización de un sistema de procesamiento más económico y de propósito más específico como puede ser un microcontrolador, DSP o FPGA. El dispositivo *DAQ NI-6211* posee una etapa de conversión analógico-digital (ADC, por sus siglas en inglés) de 16 bits, la cual es utilizada para digitalizar las señales de voltaje y corriente obtenidas del motor de inducción. A la salida del *DAQ NI-6211*, la señal digital representada con palabras binarias de 16 bits se procesa en la PC mediante el uso del programa LabVIEW [5]. Este lenguaje de programación desarrollado por National Instruments representa una herramienta gráfica muy útil para la realización de pruebas, control, y diseño mediante la programación. En particular, este proyecto hace uso de las librerías

para la comunicación con Bluetooth que están disponibles en LabVIEW. De esta manera, dichas librerías permitirán realizar la interfase entre la unidad de adquisición y el dispositivo Bluetooth para la transmisión por radiofrecuencia de los datos. En la siguiente sección se describe más a detalle la etapa de comunicación por radiofrecuencia.

II.C. Enlace por Radiofrecuencia

La tercera etapa en el sistema representa el proceso de transmisión de datos por el medio. Para esto, se propone el uso de la tecnología Bluetooth como medio de transporte. La elección de usar esta tecnología se basa principalmente por su estandarización en una amplia gama de dispositivos disponibles en el mercado [2].

Bluetooth es una tecnología que opera en la banda 2.4 GHz, una de las bandas de radio industrial, científica y médica (ISM, por sus siglas en inglés) que no requiere licencia. La potencia de transmisión es 1mW (0 dBm) lo cual permite tener una cobertura de hasta 10 m aproximadamente dentro de un entorno sin obstáculos. Bluetooth esta diseñada para aplicarse a dispositivos que generalmente funcionan con batería, por lo tanto el bajo consumo de energía es una prioridad para este estándar. Los datos se pueden intercambiar entre dispositivos a velocidades hasta 1 Mbits/s (se esperan 3 Mbits/s en la segunda generación de esta tecnología). Otra de las características importantes es el uso de la tecnología de saltos en frecuencia (FHSS, frequency hopping spread spectrum) que permite a los dispositivos comunicarse inclusive en áreas donde existe una gran interferencia electromagnética, además provee de esquemas de encriptación y verificación.

Una comparación del estándar Bluetooth con otras tecnologías de características similares se presenta en la Tabla 3. Como se observa, Bluetooth alcanza una mayor velocidad en comparación con las otras tecnologías aunque

Tabla 3. Comparación de la tecnología Bluetooth con otras tecnologías existentes.

Tecnología	Velocidad (Kb/s)	Frecuencia (GHz)	Alcance (m)
Infrarrojo	9.6	300	0.08
Bluetooth (omni-direccional)	1	2.4	10
ZigBee (omni-direccional)	20	2.4	75
Wibree	Compatible Con Bluetooth	2.4	10

existen tecnologías inalámbricas que son de algún propósito más específico y que pueden sustituir la tecnología Bluetooth para ciertas aplicaciones. Sin embargo, estas otras tecnologías llegan a tener ciertas desventajas, como por ejemplo, la comunicación infrarroja es de muy corto alcance y requiere línea de vista para la comunicación. Por su parte, ZigBee y Wibree están enfocadas a aplicaciones que requieren una baja tasa de transmisión y un bajo consumo de potencia y además son compatibles con Bluetooth. Sin embargo, estas tecnologías son relativamente nuevas y por lo tanto pocos son los dispositivos que cuentan con su implementación. De esta manera, para el sistema propuesto en este artículo se hace uso de un dispositivo Bluetooth cuyas características se enlistan a continuación:

- Adaptador USB 2.0 compatible con 1.1.
- Versión de Bluetooth 2.0+EDR.
- Bluetooth de clase 2.
- Banda de frecuencia 2.402 a 2.48 GHz.
- Rango de hasta 10 m.
- Tasa de transmisión de hasta 3 Mb/s.
- Rango de temperatura 0 a 50 °C.
- Consumo máximo de corriente, 50mA.

En la actualidad existen 4 versiones de Bluetooth las cuales se pueden dividir en 3 clases según la potencia de transmisión del

dispositivo. La versión 2.0+EDR incorpora la técnica Enhanced Data Rate (EDR), esta permite mejorar las velocidades de transmisión en hasta 3Mbps a la vez que intenta solucionar algunos errores de la versión 1.2 [2]. Por último, los requerimientos mínimos de la PC para trabajar con el modulo Bluetooth (USB) son: Windows XP con Service Pack 2 y controladores para el adaptador USB de Bluetooth.

II.D. Unidad de Monitoreo Remoto (PDA)

La cuarta y última etapa consiste en la recuperación de los datos provenientes de la fuente de información a través de una PDA (Personal Digital Assistant) o Pockets PC. Esta plataforma representa una opción con gran flexibilidad debido a que hoy en día cuentan con la capacidad de procesamiento y almacenamiento de datos mediante el uso de Word, Excel, PowerPoint, lectura de archivos (audio y video) y dispositivos de comunicación inalámbrica como Bluetooth, Irda, Wi-Fi, y GPS. Además ofrece al usuario el recurso de la movilidad. Para este trabajo se utilizó una PDA con la versión de Windows Mobile 5.0 [6] como unidad de monitoreo remoto. Una importante propiedad de esta versión es que soporta el software LabVIEW 8.2 el cual es utilizado para el procesamiento y representación de los datos en la PDA. Cabe mencionar que la versión Windows Mobile 2003 (second edition) también soporta LabVIEW 8.2.

La tarea de monitoreo requiere que la unidad remota pueda interpretar los datos enviados desde la unidad de adquisición y procesamiento de datos. De esta manera, uno de los retos en el trabajo es la programación de la PDA para lograr la conectividad mediante un dispositivo Bluetooth, interpretar la información recibida y visualizarla de manera gráfica. Para resolver este problema se desarrollo un programa basado en el entorno de LabVIEW el cual permite la conexión entre el módulo Bluetooth (USB) y la

PDA. Cabe hacer mención que el módulo USB usa la versión 2.0 + EDR de Bluetooth y la PDA la versión 1.2, comprobando así la compatibilidad entre las diferentes versiones de Bluetooth.

Una vez que el módulo Bluetooth (USB) y la PDA están conectados, los datos que se generan por la fuente de información se despliegan de manera remota en la PDA. Estos datos se muestran después de haber sido procesados e interpretados por el programa en la PDA. Todos los datos que se muestran en la pantalla de la PDA son en tiempo real y tienen una frecuencia de muestreo de 500Hz, este valor fue seleccionado debido a que la frecuencia máxima a la que puede operar el inversor de frecuencia oscila alrededor de los 200 Hz, por lo tanto un valor mayor de 400 Hz garantizará que no exista el efecto de solapamiento (aliasing) en el sistema.

III. APLICACIÓN DE MONITOREO SOBRE UN MOTOR DE INDUCCIÓN

Un problema muy común en la industria es el monitoreo de motores. En particular, en un motor de inducción es importante contar con la información de ciertas variables como la demanda de corriente al momento del arranque. Al arranque del motor, es necesario que este tenga la fuerza suficiente para vencer la fricción mecánica y par de carga. Es decir, el motor demanda más corriente al momento del arranque que cuando se encuentra en movimiento, sin embargo, en ocasiones llegan a tener una sobre carga que genera una demanda mayor de corriente aún cuando se encuentran en movimiento. Ahora, si el motor se encuentra operando de forma continua, fuera de sus condiciones nominales de uso, como consecuencia se pueden desarrollar fallas en sus partes eléctrica o mecánica.

En este sentido, una manera de monitorear la "salud" del motor es por medio del censado en tiempo real de sus voltajes y corrientes de estator, y en seguida generando señales de

residuo que correlacionen estas mediciones [7]. Para este propósito se utilizan la potencia total de estator y el radio del Vector de Park para las corrientes de línea [8]. Posteriormente se realiza un análisis en el dominio de la frecuencia de estos residuos para identificar los componentes característicos de falla. Para el desarrollo de este proceso se implementó un banco de pruebas que incluye un inversor trifásico con modulación vectorial para operar un motor de inducción de 1/4 HP, una tarjeta de censado (voltajes fase-fase y corrientes de línea) y un programa para análisis desarrollado en LabVIEW.

En la siguiente sección se describe el funcionamiento y los resultados experimentales del sistema de monitoreo propuesto al aplicarse a un motor de inducción.

IV. RESULTADOS EXPERIMENTALES

El sistema se divide en dos partes; la primera parte conforma el programa encargado de capturar los datos y enviarlos a través de un dispositivo Bluetooth y la segunda parte tiene la tarea de capturar, procesar y visualizar la información enviada.

IV.A. Enlace Bluetooth

Para poder realizar un enlace con Bluetooth es necesario que exista un servidor y un cliente, en nuestro sistema se usa la PC como servidor y la PDA como cliente. En el caso de la PC, la función *Bluetooth Create Listener* de LabVIEW se encarga de crear un servicio (*Bluetooth Serial Port* o *File Transfer*), la cantidad de servicios que puede ofrecer un dispositivo Bluetooth depende de los recursos del diseño. Cada servicio está identificado por un número con un total de 32 dígitos en hexadecimal llamado UUID (Identificador Único Universal, por sus siglas en inglés). En cada dispositivo Bluetooth existe un UUID diferente para cada servicio de un total de $2^{128} = 3.4 \times 10^{38}$ UUIDs que pueden ser creados, lo que resulta casi imposible de agotar los posibles valores de un UUID. En

LabVIEW existen 2 maneras distintas de generar un servicio, una es mediante la función *Create Listener* que debe coincidir con el servicio que ofrece la PDA y la otra es indicando en la función un canal de transmisión y la dirección Bluetooth de la PDA.

Una vez que se ejecuta el programa de LabVIEW en la PC, la PDA será capaz de hacer la petición de conexión al servidor. La función *Bluetooth Wait On Listener* de LabVIEW es la encargada de tomar la solicitud de conexión y hacer coincidir los datos que se cargan en la función *Bluetooth Create Listener*. En caso de que exista una coincidencia entonces la función *Bluetooth Wait On Listener* realizará el enlace, de lo contrario el servidor rechazará la petición y no habrá conexión alguna. La PDA tiene 30 segundos antes de que el servidor rechace cualquier petición de conexión.

Por otro lado, una de las tareas que tiene que realizar un cliente es buscar los dispositivos Bluetooth disponibles y realizar la petición de conexión. LabVIEW usa la función *Bluetooth Discover* para localizar todos los dispositivos Bluetooth que se encuentren a su alcance. La información generada se despliega en la pantalla de la PDA donde el usuario selecciona al dispositivo Bluetooth que se desea conectar. Finalmente, cuando el usuario decide realizar la conexión, la función *Bluetooth Open Connection* realiza la petición al servidor y después de ser aceptada se inicia la transferencia de datos entre ambos dispositivos Bluetooth.

IV.B. Resultados Experimentales

El objetivo de esta sección es mostrar el funcionamiento de la plataforma portátil para el monitoreo de datos basado en un dispositivo PDA. Como se mencionó anteriormente, la aplicación de interés se centra en el monitoreo de un motor trifásico de 1/4 HP, así como su análisis en frecuencia. El DAQ por su parte es el dispositivo encargado de digitalizar las lecturas que provienen de la resistencia shunt. Este dispositivo esta programado para realizar una

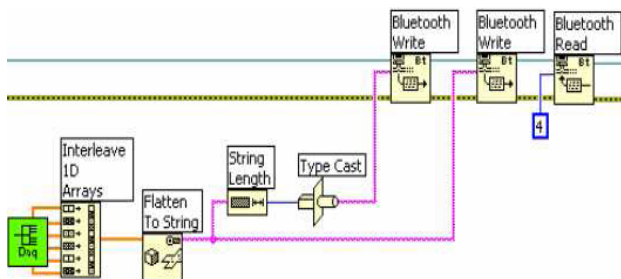


Figura 2: Diseño del diagrama a bloques en LabVIEW para enviar datos a través de Bluetooth.

lectura de 100 muestras a una frecuencia de muestreo de 500 Hz, este valor es suficiente para garantizar que no habrá efecto de solapamiento. En este caso, el DAQ toma la lectura de 6 señales, 3 de ellas son de voltajes y las otras 3 corresponden a las corrientes en el motor.

Una limitante importante que se detectó en LabVIEW es que las funciones de Bluetooth solo pueden enviar arreglos en 1D y con formato String. Por este motivo es necesario que los datos capturados por el dispositivo DAQ deben ser entrelazados a través de la función *Interleave 1D Arrays* para después convertidos al formato String a través de la función *Flatten To String*. Una vez capturados los datos por el DAQ, estos pueden ser enviados a través del protocolo de Bluetooth. En la Figura 2 se muestra el diagrama a bloques en LabVIEW que realiza el proceso de transmisión. Para poder enviar datos, LabVIEW debe conocer el número de datos, para esto se usan las funciones *String Length* y *Type Cast* que especifica la cantidad de datos a enviar y se coloca en una función *Bluetooth Write*, después la función *Bluetooth Write* se encarga de enviar los datos para que finalmente la función *Bluetooth Read* realice la sincronización entre servidor y cliente.

Por otra parte, en la Figura 3 se muestra el proceso de la recepción de datos en la PDA. Como se observa, primero se usa la función *Bluetooth Read* que adquiere el tamaño de los

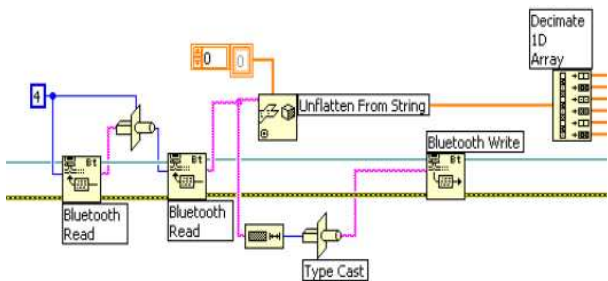


Figura 3: Diseño del diagrama a bloques en LabVIEW para recibir datos a través de Bluetooth.

datos que se envían desde el servidor. Una vez más se aplica la función *Bluetooth Read* pero en esta segunda ocasión es usada para la lectura de los datos. Finalmente la función *Bluetooth Write* es usada para realizar la sincronización entre el servidor y el cliente. Los datos capturados en la segunda llamada de la función *Bluetooth Read* se convierten del formato *String* al formato *Array 1D* mediante la función *Unflatten From String*. Por último, la función *Decimate 1D Array* es aplicada a los datos para separarlos y alinearlos en la forma original en la que fueron capturados por el DAQ. Después de la función *Decimate 1D Array* cada señal opera independientemente hasta obtener una visualización gráfica en la pantalla de la PDA que muestra la información que viene del motor.

En la Figura 4 se presenta un ejemplo de la captura de las señales de voltajes y corrientes para una condición de velocidad constante en el motor y sin carga. Por otro lado, la Figura 5 ilustra la potencia total y el vector de Park para esa misma condición de operación. Finalmente, el análisis en frecuencia de los voltajes y corrientes se muestra en la Figura 6.

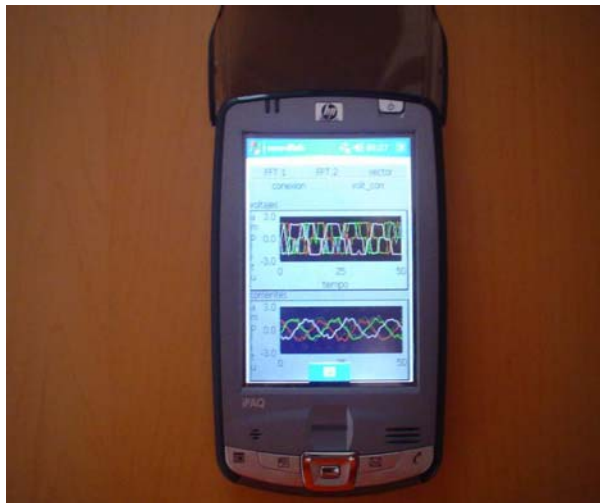


Figura 4: Visualización de las señales de voltajes y corrientes de un motor trifásico en la PDA.



Figura 5: Visualización de la Potencia de Entrada y el Vector de Park en la PDA.

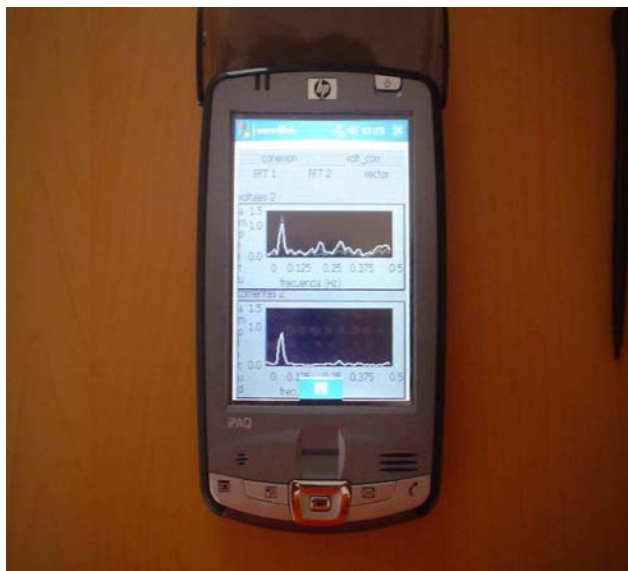


Figura 6: Visualización de la transformada de Fourier correspondientes a voltajes y corrientes en la PDA.

V. CONCLUSIONES

En este trabajo se presentó la integración e implementación de un sistema de monitoreo de datos por radiofrecuencia. Como ejemplo experimental se realizó el monitoreo de un motor trifásico mediante una PDA. Dos de las características principales del sistema propuesto se presentan a continuación:

- Monitoreo de datos de forma continúa. A diferencia de otros sistemas existentes, el sistema propuesto ofrece la capacidad de comunicación móvil y permanente a través de la tecnología inalámbrica Bluetooth.
- Unidad de Monitoreo Pequeña. A diferencia de los instrumentos de medición tradicionales, una PDA tiene la capacidad mediante la instrumentación virtual de realizar el monitoreo de los equipos o máquinas de interés a través de un dispositivo pequeño y de bajo costo.

Por un lado, la tecnología Bluetooth es un estándar global cuyas principales características

son su fiabilidad, es de bajo consumo de energía y de un costo bajo. Por otra parte, el uso del lenguaje de programación de LabVIEW permite que el dispositivo PDA sea una herramienta muy versátil con la capacidad de extender su funcionamiento en una amplia gama de aplicaciones.

En conclusión, la contribución principal de este trabajo se centra en mejorar y simplificar las estrategias de mantenimiento en líneas de producción del sector industrial donde el funcionamiento adecuado y continuo de las máquinas es vital. De esta manera, el sistema propuesto representa una solución alternativa en el problema de monitoreo.

VI. REFERENCIAS

- [1] Jaap C. Haartsen, *The Bluetooth Radio System. IEEE Personal Communications*, February 2000.
- [2] *Specification of the Bluetooth System, Version 2.0+ EDR*, Bluetooth Special Interest Group (SIG).
- [3] Rafael C. Gonzalez & Richard E. Woods *Digital Image Processing, 2nd Edition*, Prentice Hall.
- [4] <http://www.ni.com/>
- [5] <http://www.ni.com/labview/>
- [6] <http://www.microsoft.com/windowsmobile> Página Oficial de Windows Mobile.
- [7] M.E.H. Benbouzid, "A Review of Induction Motors Signature Analysis as a Medium for Faults Detection", *IEEE Trans. Ind. Electr.*, Vol. 47, No. 5, pp 984-993, 2000.
- [8] D.U. Campos-Delgado, J.S.Murguía-Ibarra y O. Ramírez-Rodríguez, "Multivariable Fault Diagnosis in Induction Motors", *Congreso Nacional de Control Automático 2007*, Monterrey, N.L., 24-26 de Octubre, 2007, (ISBN-978-970-32-5089-9).

Diseño De Un Electrocardiógrafo Digital Usando Como Plataforma El Kit De Desarrollo Spartan-3E De Xilinx

Herrera Galicia Rubén, Martínez Chiñas Celia Del Carmen, Urbina Brito Norberto, Díaz Gutiérrez Roger Alejandro, Ramón Paz Sergio Antonio.

Área de Control y Automatización, Departamento de Eléctrica y Electrónica,
Instituto Tecnológico de Tuxtla Gutiérrez
Carretera Panamericana Km. 1080, Col. Teran, Tuxtla Gutiérrez, Chiapas, 29070, México
Tel. 6150380, correo electrónico: h_galicia24@hotmail.com, cely73@hotmail.com, drakonian_6@hotmail.com

RESUMEN

El trabajo aquí descrito está enfocado a diseñar un sistema capaz de adquirir señales bioeléctricas provenientes del corazón de un ser humano y procesarlas por medio de un FPGA obteniendo así una señal con la menor interferencia y ruido posibles que garanticen un registro confiable de la intensidad y duración de los impulsos del corazón y que pueda ser transportado por un medio de comunicación para finalmente ser visualizado en el monitor de una computadora.

I. Introducción

Un electrocardiograma (ECG) es un registro gráfico de los potenciales eléctricos producidos por el tejido cardíaco. El corazón es especial entre los músculos del cuerpo en vista de que posee la propiedad de la generación de un impulso eléctrico automático. Este impulso provoca la excitación de las fibras musculares y resulta en la contracción cardíaca. La conducción de este impulso origina pequeñas corrientes eléctricas que se propagan a todo el cuerpo. El ECG se obtiene colocando electrodos en varios sitios de la superficie del cuerpo que se conectan a un aparato de registro.

Un electrocardiograma consiste en una serie de formas de onda que ocurren en un orden repetido. Estas formas de onda provienen de una línea de

fondo plana llamada línea isoelectrica. Cualquier desviación de esta línea indica actividad eléctrica. Un ciclo de corazón es representado por un grupo de cinco desviaciones principales designadas por las letras P, Q, R, S, T. La onda P representa la despolarización del atria y es asociada con su contracción. La primera desviación negativa es la onda Q y es seguida de una desviación positiva llamada onda R. El complejo se termina con una desviación negativa conocida como onda S. El complejo QRS denota la despolarización de los ventrículos y está asociado con su contracción. Algunos tiempos típicos para las partes del ECG son: P-R (0.2 s), QRS (0.1 s), Q-T (0.38 s).

Para interpretar un electrocardiograma se recomienda tomar una serie de medidas básicas: la frecuencia cardíaca, ritmo, eje del complejo QRS en el plano frontal, intervalo PR, duración del QRS. Con esta secuencia de lectura se detecta prácticamente cualquier alteración electrocardiográfica.

En México existen deficiencias relacionadas con equipo de diagnóstico de enfermedades que den una respuesta eficaz, a precio competitivo, y que sea transportable. Las patologías cardíacas son de las principales causas de deceso en la población de edad adulta, podrán evitarse muchas de estas muertes si se detectan a tiempo estos problemas cardíacos.

Un ECG tiene significado de diagnóstico. Un paciente con enfermedad cardiaca puede tener un ECG normal y un individuo normal puede tener un ECG anormal. No obstante, un paciente puede recibir la seguridad garantizada de que no tiene padecimiento cardiaco en base exclusivamente de un ECG normal.

Un FPGA ofrece ventajas muy favorables para el diseño y desarrollo de un electrocardiógrafo: flexibilidad, capacidad de procesamiento en paralelo y velocidad. Entre esas ventajas resalta la flexibilidad, cualidad inherente a los circuitos de lógica programable, que permite hacer cambios en el funcionamiento y hacer otro procesamiento digital de la señal sin cambiar el hardware.

Para implementar el electrocardiógrafo digital propuesto en el presente trabajo se usó el kit de desarrollo Spartan 3E de Xilinx. El sistema se compone de tres parches de electrodo para adquisición de la señal ECG de la persona, un amplificador de instrumentación basado en el AD624 con amplificación de 1000 V/V, dos filtros analógicos, un kit de desarrollo Spartan 3E, una interfase de comunicación USB-RS232, y una PC. El diseño completo del sistema se dividió en cinco unidades: sensado, amplificación, digitalizado, filtrado, y comunicación. Un diagrama a bloques del sistema se presenta en la Fig. 1.

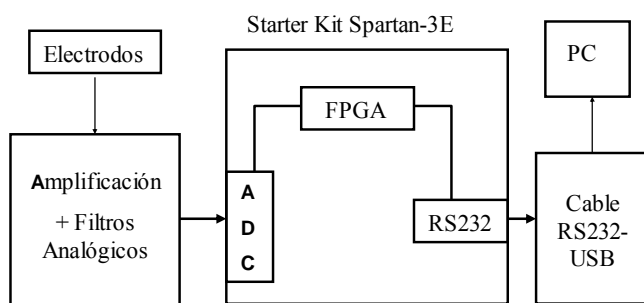


Fig. 1 Electrocardiógrafo digital; Diagrama a bloques.

El desarrollo del presente electrocardiógrafo se hace con el propósito de ayudar a la comunidad médica a contar con un aparato que dé una

respuesta eficaz, a precio competitivo, que sea transportable y capaz de auxiliar al especialista en la elaboración de un diagnóstico acertado.

Este proyecto es importante porque es una herramienta útil para diagnosticar enfermedades cardiacas. Con su desarrollo se está colaborando con el sector salud. Y tiene la ventaja de que al utilizar un FPGA permite hacer modificaciones al diseño sin tener que cambiar el hardware.

II. Bases teóricas

1) *Programación del FPGA.* El lenguaje usado para programar al FPGA es el lenguaje de descripción de hardware VHDL. Este lenguaje es utilizado para describir circuitos en un nivel alto de abstracción y ha sido aceptado como un medio estándar de diseño. El lenguaje VHDL está creado específicamente para el diseño de hardware, se pueden implementar con él circuitos lógicos, tanto combinacionales como secuenciales. Éste lenguaje también permite describir elementos más complejos, como CPU y retrasos de tiempo.

Un programa en VHDL consta de dos partes. La entidad que sirve para relacionar al diseño con el mundo exterior. Aquí se declara lo que se va a crear como una caja negra, de la que se especifican sus entradas y sus salidas. En la segunda parte, la arquitectura, se describe como trata el circuito la información correspondiente a las entradas para obtener las salidas. Dentro de la arquitectura pueden haber subprogramas, cada uno cumpliendo una función y todos ellos gobernados por un bloque principal. Los subprogramas en su funcionamiento pueden ser secuenciales o concurrentes. La estructura de programación es similar a la de los lenguajes de mediano nivel: C, ada. Usa declaraciones de tipos de datos, variables, y arreglos. También usa estructuras de selección (*if, case*), y estructuras de repetición (*for, while*). Los puertos de las entidades se definen siempre para los tipos *std_logic* y *std_logic_vector*. Las operaciones aritméticas estándares solo están definidas para los tipos *signed* y *unsigned*.

En VHDL la concurrencia consta de un conjunto de procesos ejecutándose asincrónicamente y comunicándose mediante una red de señales. En el interior de un proceso las acciones se suceden en el orden secuencial marcado por las construcciones, pero en un tiempo nulo.

2) *Filtros digitales*. En lo básico de su funcionalidad un filtro digital se comporta de igual manera que un analógico. Un filtro es un sistema encargado de alterar el contenido de la información espectral de una señal de entrada $X(t)$, produciendo una señal de salida $Y(t)$. Los filtros analógicos son implementados mediante el uso de dispositivos discretos que operan sobre formas de onda continuas. Los filtros digitales son implementados mediante el uso de circuitos lógicos o en programas de computador que operan sobre secuencias de números que son obtenidos por el muestreo de ondas continuas.

Los filtros digitales tienen un extendido uso gracias a la facilidad que se presenta para montar estos diseños en dispositivos de procesamiento digital DSP. En otros casos son diseñados e implementados en circuitos lógicos programables como el FPGA.

Los filtros digitales tienen ventajas sobre los filtros analógicos y estas son algunas de ellas: Un filtro digital en sí mismo no introduce ruido, gracias a la forma en que se implementa (Software o Circuito digital). La precisión de un filtro digital depende del error de redondeo, el cual está determinado por el número de bits usado para representar a las variables en el filtro. Es fácil y barato cambiar las características de operación del filtro. Esto a diferencia de los filtros analógicos donde se requiere una reconstrucción del hardware. Además su desempeño no está en función del deterioro de sus componentes, de las variaciones de la temperatura o de las variaciones del voltaje de la fuente de alimentación.

Esta clase de sistemas se encuentra caracterizada por una ecuación lineal en diferencias con coeficientes constantes, ec. (1). Mediante la

transformada Z se pueden caracterizar como una función de transferencia racional, ec. (2).

$$y(n) = -\sum_{k=1}^N a_k y(n-k) + \sum_{k=0}^M b_k x(n-k) \quad \text{ec. (1)}$$

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad \text{ec. (2)}$$

En las ecuaciones (1) y (2) los polos y los ceros del sistema se encuentran dados por los $\{b_k\}$ y $\{a_k\}$, los cuales determinan las características de la respuesta en frecuencia del sistema. Existen dos tipos básicos de filtros digitales, los No-recursivos y los recursivos. Los filtros No-recursivos se caracterizan por no poseer realimentaciones de lo cual se aprecia que la salida se encuentra dada solo en función de la entrada actual y las entradas pasadas, ec. (3).

$$y(n) = \sum_{k=0}^{M-1} b_k x(n-k) \quad \text{ec. (3)}$$

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} \quad \text{ec. (4)}$$

Para los filtros recursivos la ecuación en diferencias se encuentra expresada como una suma de dos funciones polinomiales, ec. (5), la cual lleva a encontrar una función de transferencia dada por la ec. (6).

$$y(n) = -\sum_{k=1}^N a_k y(n-k) + \sum_{k=0}^M b_k x(n-k) \quad \text{ec. (5)}$$

$$H(z) = \frac{\sum_{k=0}^M a_k z^{-k}}{1 - \sum_{k=1}^N b_k z^{-k}} = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}}{1 - b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n}} \quad \text{ec. (6)}$$

A los primeros pertenecen los filtros tipo FIR, los cuales se caracterizan por no poseer realimentación y a los segundos los filtros tipo IIR en los cuales la salida se encuentra dada en función de la entrada actual y pasadas y de las salidas pasadas.

III. Diseño del electrocardiógrafo

1) *Unidad de adquisición analógica.* Debido a que la corriente que genera el cuerpo humano es del orden de milivolts se debe tener extrema precaución en esta etapa ya que es muy fácil que se presenten ruidos no deseados junto con la señal útil. Las probables fuentes de ruido en un ambiente de hospital son: la red eléctrica, las lámparas fluorescentes, los motores eléctricos u otro equipo medico.

Para la adquisición de la señal ECG de la persona se usan tres parches de electrodo. Se coloca firmemente el primer electrodo sobre la parte interna del tobillo derecho. Un segundo electrodo a un costado de la costilla izquierda, a la altura del codo izquierdo. Un tercer electrodo sobre el corazón, sobre los músculos del ventrículo derecho. Se conectan los caimanes del cable a los conectores de los parches de los electrodos. Se conecta el caimán negro, la referencia, al tobillo derecho. Esto es el punto de referencia para la línea isoelectrica. Se conecta el caimán verde, negativo, a la costilla izquierda. El caimán rojo, positivo, al corazón.

2) *Unidad de amplificación.* Para amplificar la señal proveniente de los electrodos se uso el amplificador de instrumentación AD624 con una amplificación total aproximada de 1000 V/V, en combinación con un filtro analógico pasabanda de 0.05 Hz a 100 Hz, de segundo orden, y un filtro rechaza banda de 60 Hz. Para acoplar el voltaje que entrega el sensor (Offset = 1V) al voltaje de 1.65v, requerido como nivel cero por el convertidor ADC del Kit Spartan 3E, se agrego un amplificador con ganancia 1.65 V/V.

3) *Unidad de digitalizado.* En esta parte se convierte la señal analógica del cuerpo (voltaje) a

una palabra digital de 14 bits con signo en notación de complemento a dos. Esto se hace mediante el convertidor analógico - digital incluido en el LTC1407A-1 del kit de desarrollo Spartan 3E. Este ADC convierte voltajes de $\pm 1.25v$ relativos a 1.65v.

4) *filtrado digital.* En esta etapa se pretende mejorar la señal aplicando distintos tipos de filtros; pasa altas (0.05 Hz), rechaza banda (60Hz) y pasa bajas (150Hz). El objetivo de éstos es eliminar frecuencias menores a 0.1Hz, mayores a 150Hz y el ruido de la fuente de alimentación. La frecuencia de cualquier pulso cardiaco en reposo no sobrepasa los 20 Hz, y así se evitan señales extrañas provocadas hasta por los mismos circuitos.

Aquí se reportan resultados del filtro pasabajas, dado que la programación de los otros filtros aun esta en proceso de pruebas. Para la programación se usa lenguaje VHDL para ser descargada al FPGA Spartan 3E del Kit de desarrollo, usando para esto el entorno de programación ISE 8.2i.

5) *Unidad de comunicación.* La comunicación del kit con la computadora es a través del puerto serie RS232 usando para esto un cable conversor de puerto serie RS232 a USB.

IV. Programación

Se escribió un programa en VHDL aplicando la estructura de una maquina de estados sensible a la entrada de reloj del sistema, 50 Mhz. El programa consta de cuatro partes: configuración de la amplificación del amplificador LTC6912-1 incluido en el kit SPARTAN 3E, control del convertidor A/D LTC1407A-1, la etapa de filtrado y el cuarto bloque se uso para enviar la salida digital de los filtros a través del conector RS232-USB hasta una PC. Los retrasos de tiempo están calculados para obtener 512 muestras por segundo. Un ejemplo de la programación se presenta a continuación para el caso de un filtro pasa bajas tipo FIR. Dicho filtro es un promediador que utiliza ocho muestras. Se escogió este para mostrarlo como ejemplo por su

sencillez. Pero la idea es usar distintas estructuras de filtros y probarlas sustituyendo al filtro promediador por la estructura deseada. La programación es directa en VHDL y no requiere el uso de PICOBLAZE.

```

library IEEE;
use IEEE.NUMERIC_STD.ALL;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
-----
-- Control del tiempo: SPI_SCK
-- Entrada de bits de datos: SPI_SDO
-- Salida bits de configuración AMP: SPI_MOSI
-- Salida habilitación AMP: SPI_AMP_CS
-- Inicio de conversión ADC: SPI_ADC_CONV
-- Salida a TX RS232: TxD
-- Conector FX2 : LED
-----
entity ADC_CANAL is
  port ( SPI_SCK : out std_logic; -- "U16"
        SPI_SDO : in std_logic;  -- "N10"
        SPI_MOSI : out std_logic; -- "T4"
        SPI_AMP_CS : out std_logic; -- "N7"
        SPI_ADC_CONV : out std_logic; -- "P11"
        BTN_SOUTH : in std_logic;  -- "K17"
        STRATAFLASH_OE : out std_logic; -- "C18"
        STRATAFLASH_CE : out std_logic; -- "D16"
        STRATAFLASH_WE : out std_logic; -- "D17"
        PLATFORMFLASH_OE : out std_logic;
                                     -- "B18"

        TxD : out std_logic; -- "M14"
        CLK : in std_logic;  -- "C9"
        LED : out std_logic_vector(7 downto 0));
end ADC_CANAL;
-----
architecture RTL of ADC_CANAL is

  type adcStateType is (startAmp, sendBit,
                        clockHigh, clockLow,
                        csHigh, convHigh, convLow,
                        requestBit, readData,
                        firFiltro, txOut, Delay);

  signal adcState : adcStateType := startAmp;
  signal adcData : std_logic_vector (33 downto 0);
  signal xData : std_logic_vector (13 downto 0);

```

```

  signal yData : std_logic_vector (13 downto 0);
  signal adcMsb : std_logic_vector (7 downto 0);
  signal txBuff : std_logic_vector (7 downto 0);
  signal ampData : std_logic_vector (7 downto 0);
  signal event_1hz : std_logic;
  constant BITperiod : time := 104166 ns; -- 9600
  -- baudios
  -----
begin
  -----
  STRATAFLASH_OE <= '1';
  STRATAFLASH_CE <= '1';
  STRATAFLASH_WE <= '1';
  PLATFORMFLASH_OE <= '0';
  -----
  process(CLK, btn_south)
    variable bitCounter : integer range 0 to 33;
    variable S0,S1,S2,S3 : integer range 0 to 16383;
    variable S4,S5,S6,S7 : integer range 0 to 16383;
    variable Sample : integer range 0 to 16383;
    variable yOut : integer range 0 to 16383;
    begin
      -----
      if CLK'event and CLK='1' then
        if btn_south = '1' then
          SPI_ADC_CONV <= '0';
          SPI_SCK <= '0';
          TxD <= '1';
          adcState <= startAmp;
        else
          case adcState is
            ----- Gain = -5
            when startAmp =>
              ampData <= x"33";
              SPI_SCK <= '0';
              SPI_AMP_CS <= '0';
              bitCounter := 7;
              adcState <= sendBit;
            -----
            when sendBit =>
              SPI_MOSI <= ampData(7);
              ampData <= ampData(6 downto 0) & "0";
              adcState <= clockHigh;
            ----- 100 ns (Retardo)
            when clockHigh =>
              wait for 100 ns;
              SPI_SCK <= '1';
              adcState <= clockLow;

```

```

-----
when clockLow =>
  wait for 100 ns;
  SPI_SCK <= '0';
  if bitCounter = 0 then
    adcState <= csHigh;
  else
    bitCounter := bitCounter - 1;
    adcState <= sendBit;
  end if;
-----

when csHigh =>
  SPI_AMP_CS <= '1';
  adcState <= convHigh;
-----

when convHigh =>
  SPI_ADC_CONV <= '1';
  adcState <= convLow;
-----34 ciclos CLK

when convLow =>
  SPI_ADC_CONV <= '0';
  bitCounter := 33;
  adcState <= requestBit;
-----

when requestBit =>
  SPI_SCK <= '1';
  adcState <= readBit;
-----

when readBit =>
  SPI_SCK <= '0';
  adcData(0) <= SPI_SDO;
  adcData <= adcData(32 downto 0) & "0";
  if bitCounter = 0 then
    adcState <= readData;
  else
    bitCounter := bitCounter - 1;
    adcState <= requestBit;
  end if;
-----

when readData =>
  xData <= adcData(31 downto 18);
  LED <= adcData(31 downto 24); -- Msb
  Sample := to_integer(unsigned(xData));
  adcState <= firFiltro;
-----

when firFiltro =>
  s7:=s6;
  s6:=s5;
-----
  S5:=S4
  S4:=S3;
  S3:=S2;
  S2:=S1;
  S1:=S0;
  S0:=Sample;
yOut <= (S0+S1+S2+S3+S4+S5+S6+S7)/8;
yData <=
  std_logic_vector(to_unsigned(yOut,14));
  txBuff <= yData(13 downto 6);
  adcState <= txOut;
-----

when txOut =>
  TxD <= '0';
  wait for BITperiod;
  for i in 0 to 7 loop
    TxD <= txBuff(i);
    wait for BITperiod;
  end loop;
  TxD <= '1';
  wait for BITperiod;
  adcState <= Delay;
-----

-- 512 muestras/ segundo ocupan 1,953,125 ns
-- por muestra. El programa hasta aquí se tarda
-- 1,043,940 ns. Faltan 909185 ns.
-----

when Delay =>
  wait for 909185 ns;
  adcState <= startAmp;
-----

end case;
end if;
end process;
-----

end architecture RTL;

```

V. Resultados

Para la adquisición de la señal ECG se usaron tres parches de electrodo. Se colocó el electrodo de referencia sobre la parte interna del tobillo derecho. El electrodo negativo a un costado de la costilla. El tercer electrodo sobre el corazón. Para conectar a los electrodos con la entrada del amplificador se usó un cable modelo KENDALL-LTP marca TYCO HelthCare. Después las

señales ECG fueron amplificadas y convertidas de analógico a digital para su procesamiento en el FPGA. En la Fig. 2 se muestra una imagen de la señal ECG a la entrada del convertidor vista en el osciloscopio. En dicha figura se puede observar que la señal obtenida del ECG es de buena calidad.

Una vez aplicado el filtro digital FIR sobre la señal digitalizada del ECG, los resultados obtenidos a la salida del mismo fueron transmitidos a través del puerto serie a una PC portátil para su visualización.



Fig. 2 ECG a la entrada del convertidor A/D LTC1407A-1.

En la Fig. 3 se muestra una imagen de la señal ECG mostrada en la PC.

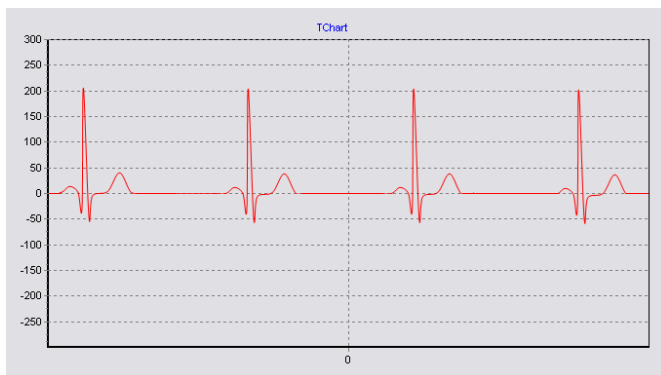


Fig. 3 Imagen del ECG digital en la PC.

VI. CONCLUSIONES

Se implemento un sistema para adquirir señales bioeléctricas provenientes del corazón de un ser humano con bajo nivel de ruido e interferencia para procesarlas por medio de un FPGA obteniendo una señal que garantiza un registro confiable de la intensidad y duración de los impulsos del corazón que fue transportado por una interfase de comunicación RS232-USB para finalmente ser visualizado en el monitor de una computadora.

Se implemento un electrocardiógrafo digital usando para esto el kit de desarrollo Spartan 3E de Xilinx. El uso de un FPGA tiene la ventaja de que permite hacer modificaciones al diseño sin tener que cambiar el hardware.

REFERENCIAS

- [1] Victor Borshevich, Alexandr V. Mustyatsa, Wiacheslav L. Oleinik, "Fuzzy Spectral Analysis of Heart's Rhythms", Fifth IFSA World Congress 1993; págs:561-563; 1993
- [2] Ken Chapman, "PicoBlase Amplifier and A/D Converter Control for Spartan 3-E Starter Kit", Xilinx Ltd, 23 february 2006
- [3] "Spartan-3E Starter Kit Board User Guide", UG230(v1.0) March 9, 2006 en <http://www.xilinx.com>
- [4] "IEEE Standard VHDL Language Reference Manual", IEEE Std 1076-1987
- [5] Miguel Alberto Melgarejo Rey, "Desarrollo de un filtro digital promedio en FPGA", en

<http://www.udistrital.edu.co/comunidad/grupos/glogica/index.html>

PROPUESTA PARA PREDECIR EVENTOS EN EL SISTEMA DE MONITOREO VOLCÁNICO EN TIEMPO REAL BASADO EN LA INFORMÁTICA BIO-INSPIRADA Y ALGORITMOS DE REDES NEURONALES.

Omar Ariosto Niño Prieto, Enrique Colmenares Guillen, Manuel Martín Ortiz,
FCC-BUAP

Facultad de Ciencias de la Computación de la
Benemérita Universidad Autónoma de Puebla
Av. San Claudio y 14 sur, Ciudad Universitaria, Puebla, Pue. México.
Tel. 2295500, correo electrónico: omar.ariosto@gmail.com,
lecolme@cs.buap.mx, mmartin@cs.buap.mx

RESUMEN

En el siguiente trabajo se presenta una propuesta para predecir eventos volcánicos basada en la informática bio-inspirada y algoritmos de redes neuronales con una Red Recurrente Retardada en el Tiempo. Un sistema de monitoreo volcánico en tiempo real se utiliza para automatizar y facilitar la obtención de datos de la actividad magnética de un volcán. La predicción de eventos en un volcán puede ser útil para diseñar planes de contingencia, para salvar el mayor número de vidas humanas y disminuir las pérdidas económicas de la región.

I INTRODUCCIÓN

Los volcanes activos son un riesgo para las sociedades que viven cerca de estos colosos, por ello es necesario estudiarlos para poder predecir algún acontecimiento catastrófico que incluye grandes pérdidas humanas y materiales, dañando la economía de la región. Estos eventos no se pueden evitar, pero si se pueden prevenir con el objetivo de tener las menores pérdidas posibles ya sea humanas o materiales.

Los sistemas de software nos ayudan a predecir estos eventos, y gracias a la tecnología, se puede saber si habrá algún evento con un poco más

de anticipación que antaño; por lo que la tecnología es una herramienta indispensable para prevenir a la población de este tipo de eventos.

El monitoreo volcánico es un trabajo de riesgo, que además, requiere de tiempo para determinar manualmente el estado de un volcán activo utilizando herramientas sofisticadas para lograr el objetivo. Por lo que se necesitan sistemas informáticos que faciliten esta tarea. Actualmente existen sistemas que miden la actividad del volcán Popocatepetl en la Universidad Nacional Autónoma de México (UNAM) y el Centro Nacional de Prevención de Desastres (CENAPRED) con su sistema de monitoreo [3], el sistema de monitoreo MAGNET en Italia [2] y el sistema de monitoreo volcánico en Ecuador [1] entre otros. La Benemérita Universidad Autónoma de Puebla (BUAP) estudia la actividad del volcán Popocatepetl con la ayuda del Centro Universitario de Prevención de Desastres Regionales (CUPREDER), dependencia de la BUAP. Los instrumentos que permiten estudiar la actividad volcánica, son: algunos sensores de tierra, electrodos, así como dos magnetómetros. En la actualidad, la información es obtenida por medio del trabajo de campo y es recolectada con la ayuda de un magnetómetro. Los resultados son analizados por especialistas que pueden dar un pronóstico de la actividad volcánica.

La deficiencia de tener una mejor planeación en una contingencia es porque se carece de un sistema de tiempo real y además los datos obtenidos son insuficientes. El equipo de trabajo conformado por personas de diferentes áreas, como geólogos, vulcanólogos, ingenieros, se tienen que trasladar directamente al volcán para obtener los datos. Por cada sesión es necesario invertir un día entero para obtener información del volcán, además de los gastos económicos de traslado, alimentación y de personal.

Los beneficios sociales a largo, mediano y a corto plazo son conocer la información de la actividad del volcán Popocatepetl para poder implementar programas de protección civil en un menor tiempo y en caso de que exista actividad intensa, salvar el mayor número de vidas y tratar de tener un impacto económico positivo debido a la prevención. Los usuarios reales serán los geólogos y vulcanólogos de la BUAP y CUPREDER quienes analizarán la información recibida para tener más elementos de planeación y de protección civil. Los beneficiados serán todas las personas que habitan en comunidades cercanas al volcán y los habitantes de la ciudad de Puebla.

El siguiente artículo se divide en las secciones *I introducción*, se presenta el planteamiento del problema, *II trabajo relacionado* se presenta de una manera generalizada, la informática Bio-inspirada, *III Redes neuronales artificiales* se analiza la teoría y la propuesta para predecir eventos dentro del Sistema de Monitoreo Volcánico en Tiempo Real, en la sección *IV Sistema propuesto* se discute la arquitectura general del sistema y los objetivos del mismo, en la sección *V Metodología del sistema de monitoreo volcánico en tiempo real* [9] se discute la ingeniería de software específica con la propuesta de aplicación de las Redes Neuronales Artificiales en el sistema. Por último en la sección *VI*, se dan conclusiones y el trabajo a futuro.

II TRABAJO RELACIONADO

Una rama de las Ciencias de la Computación se encarga de estudiar el tratamiento racional de la información a través de máquinas automatizadas que ayudan al desarrollo del conocimiento y su comunicación. Por medio de la observación podemos encontrar diversos sistemas cognitivos capaces de manipular los símbolos a nuestro alrededor; estos sistemas pertenecen en la mayor parte al Reino Natural y ejemplos de ellos son el cerebro, el sistema nervioso central, la evolución natural de las especies a través de los códigos genéticos, el sistema inmunitario, los sistemas sociales y algunos sistemas psico-químicos.

En nuestro entorno, existen una gran diversidad de problemas complejos que se resuelven mediante la descomposición de los problemas en subproblemas sin importar que dicha descomposición nos puede llevar a un tiempo de resolución demasiado grande, con una complejidad *NP-Difficiles* [4], pero un problema complejo no se puede descomponer sin perder la naturaleza del mismo problema mediante su característica unitaria; un mismo tipo de problema puede ser tratado de una manera complicada o de una manera compleja teniendo en cuenta que su solución es complicada como lo es el problema computacional del “agente de viajero”.

La complejidad en los sistemas agrupa un gran número de entradas de diversos elementos que son ligados mediante diversas interacciones cicladas y/o recurrentes y el comportamiento de sus diferentes componentes son interdependientes y existen propiedades globales emergentes (propiedades de un sistema que presenta características novedosas relacionadas con las propiedades y características de otro tipo de sistema) que no siempre son visibles, dependiendo del tipo de sistema.

Los sistemas inspirados en la biología nos pueden ayudar a construir algoritmos más eficaces que nos ayudan a resolver problemas complejos, dentro de una de las ramas de la Inteligencia

Artificial. La informática bio-inspirada nos permite la construcción de sistemas artificiales (Hardware o Software) que reproducen algunos aspectos de la inteligencia humana o animal y la construcción de sistemas artificiales capaces de ayudar a resolver o reemplazar tareas humanas que necesiten de inteligencia. A través de la historia, la informática bio-inspirada ha ido evolucionando, desde el desarrollo de artefactos mecánicos, el desarrollo de la cibernética, el razonamiento lógico en base a modelos matemáticos, la representación del conocimiento, los sistemas expertos, los algoritmos de aprendizaje y el conocimiento perfeccionado gracias a la evolución del ser humano y su entorno. El conocimiento se convierte en un nuevo paradigma que reemplaza la metáfora del cerebro-ordenador y guía los sistemas inteligentes hacia la resolución de los problemas de una manera cognitiva, ya sea interactuando con su medio ambiente y/o con un comportamiento adaptativo elemental al medio. Para algunos ingenieros resulta más eficaz inspirarse de los organismos vivos y de la propia introspección humana, de esto derivan campos de estudio como el reconocimiento de patrones mediante redes neuronales *Feed-Forward*, el control de movimiento mediante redes neuronales recurrentes, la optimización mediante algoritmos genéticos, entre otros.

El cerebro es uno de las cumbres de la evolución biológica, ya que es un gran procesador de información. Entre sus características podemos destacar, que es capaz de procesar a gran velocidad grandes cantidades de información procedentes de los sentidos, combinarla o compararla con la información almacenada y dar respuestas adecuadas. Además es de destacar su capacidad para aprender a representar la información necesaria para desarrollar tales habilidades, sin instrucciones explícitas para ello. El cerebro es un conjunto de millones y millones de células especiales llamadas neuronas, y que están interconectadas entre ellas por sinapsis.

Los científicos llevan años estudiándolo y se han desarrollado algunos modelos matemáticos que tratan de simular su comportamiento. Estos modelos se han basado sobre los estudios de las

características esenciales de las neuronas y sus conexiones.

Aunque estos modelos no son más que aproximaciones muy lejanas de las neuronas biológicas, son muy interesantes por su capacidad de aprender y asociar patrones parecidos lo que nos permite afrontar problemas de difícil solución con la programación tradicional. Se han implementado en computadoras y equipos especializados para ser simulados [9].

Según la informática tradicional, se definen programas específicos para resolver algún problema determinado mediante el uso de algoritmos de resolución y de conceptos matemáticos. Pero dentro de la informática ligada a la Inteligencia Artificial un programa genérico manipula los datos que contienen la solución de algún problema dado; en el caso de la Inteligencia Artificial *clásica* existen sistemas como los motores de inferencias y los Sistemas Expertos, mediante una base de reglas y de conocimiento.

Dentro de la informática bio-inspirada, a diferencia de la Inteligencia Artificial simbólica tradicional, los datos son subsimbólicos; la inspiración biológica que proporciona los algoritmos de exploración mediante: la aproximación funcional, la exploración parcialmente aleatoria, las redes neuronales (elasticidad sináptica), la secuencia genética (selección natural), los caminos de feromonas (inspiración en las hormigas, termitas) y el sistema inmunitario (respuesta inmunitaria con anticuerpos) entre otros. En los algoritmos de redes neuronales, un mismo programa desarrollado puede ser aplicado a toda una clase de problemas sin modificar el código.

Una red de neuronas artificial es un conjunto de células lógicas que intercambian mensajes a través de un grafo de conexiones ponderadas con el objetivo de realizar una función de entrada y salida; los principios generales de las redes neuronales son inspirados directamente de la estructura del sistema nervioso del ser humano.

III REDES NEURONALES ARTIFICIALES

Con el paso de los años, los modelos de neuronas han aumentado la complejidad. Por ejemplo, se han introducido nuevos conceptos y nuevos paradigmas de computación basado en el comportamiento de las neuronas. El funcionamiento del cerebro humano es complejo, en la Tabla I [4] se hace un comparativo entre el funcionamiento del cerebro y el funcionamiento de las computadoras.

En la misma *Tabla I* se muestra una breve comparación de las operaciones que efectúa el cerebro humano y la computadora, como ejemplo podemos mencionar que para lograr una visión parecida a la que realiza el ser humano es necesario utilizar 1000 super computadoras, sin embargo la computadora memoriza detalles arbitrarios de una forma mucho más rápida.

Para memorizar detalles que normalmente los humanos realizamos todos los días, el cerebro humano es capaz de hacerlo de una manera mucho más eficaz, a diferencia de las computadoras que deben de ser programadas explícitamente para resolver un tipo de problema. El cerebro puede aprender de la experiencia.

La estructura paralela del cerebro ofrece muchas ventajas, ofrece “cálculos” paralelos muy intensos, tolerancia a los errores fuertes, tolerancia al ruido extremo, tiene degradación lenta del desempeño, generalización y extrapolación. Para el sistema, se revisaron diversos modelos de redes neuronales como el perceptrón, y los algoritmos de retropropagación de error para generar aprendizaje. [4, 10, 12].

En el Sistema de Monitoreo Volcánico en Tiempo Real se aplica los conceptos de la informática bio-inspirada mediante los algoritmos de redes neuronales para predecir eventos dentro del proceso *P6 PREDICE* y su función *PREDECIR()* de la Fig. 3.

	COMPUTADORA	CEREBRO
Estructura y complejidad	Estructura Jerárquica, procesador secuencial.	10^{10} neuronas “equivalentes” 10^4 sinapsis/neurona.
Velocidad	10^4 operaciones específicas por segundo.	100 operaciones específicas por segundo y por neurona
Potencia de Cálculo	Una operación a la vez sobre 1 o 2 datos de talla reducida	10^{10} (al menos) operaciones a la vez sobre millones (al menos) de datos.

TABLA I: COMPARACIÓN ENTRE EL CEREBRO Y LA COMPUTADORA

A) LA NEURONA ARTIFICIAL

Las neuronas artificiales son modelos que tratan de simular el comportamiento de las neuronas, una red Neuronal Artificial corresponde a una colección de Elementos Procesadores interconectados. Los pesos sinápticos o poder de las conexiones, se adaptan tal que la salida de la red ajusta una respuesta esperada, y contiene adaptación que es la capacidad para cambiar parámetros de algún sistema según una regla que minimiza el error. Estas características de procesamiento distribuido y adaptación son la marca de calidad de sistemas de procesamiento de información biológica [10]. Los modelos clásicos desarrollados en las neurociencias computacionales podemos destacar los de Hodkin-Huxley (HH), Integrate and Fire (I&F), Modelo Frecuencial [4].

Las neuronas son autómatas simples en donde la lógica de funcionamiento es independiente del estado de toda la red, tienen una topología de organización de conexiones intracelulares (i.e. Estructura de un grafo ponderado). Contiene leyes dinámicas con la evolución del estado de los autómatas y la evolución de las ponderaciones del grafo de conexiones.

Una neurona artificial es una función algebraica no lineal que tiene parámetros con valores limitados. Una neurona formal se representa de la siguiente manera (Fig. 1) [4] y la ecuación de salida representada con la ecuación 1.1:

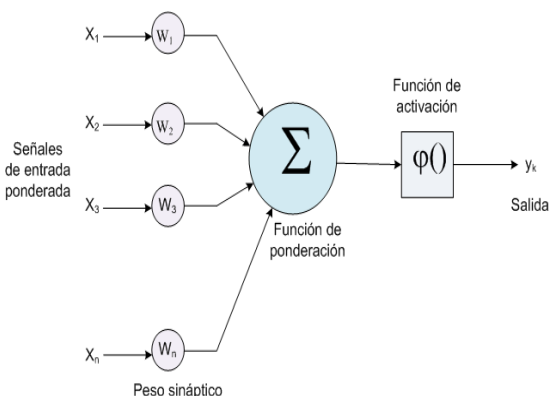


FIG. 1. LA NEURONA FORMAL

$$Y_k = \phi \left(\sum^n W_{ki} X_i + \theta_k \right) \quad (1.1)$$

Por las entradas \$X_i\$ llegan unos valores que pueden ser enteros, reales o binarios. Estos valores equivalen a las señales que se envían otras neuronas a través de las dendritas. En la neurona biológica, los pesos que hay en las sinapsis \$W_i\$, equivalen a los mecanismos que existen en las sinapsis para transmitir la señal. De forma que la unión de estos valores (\$X_i\$ y \$W_i\$) equivalen a las señales químicas inhibitorias y excitadoras que se dan en las sinapsis y que se induce a la neurona a cambiar su comportamiento [8].

El Sistema de Monitoreo Volcánico en Tiempo Real contiene el módulo P6: *PREDICE* (ver Fig. 3), cuyo objetivo es realizar las predicciones del comportamiento en el campo geomagnético del volcán Popocatepetl. El sistema muestra los eventos magnéticos que se registran en los magnetómetros y que se almacenan en una Base de Datos; Es posible diseñar modelos de predicción con algoritmos de aprendizaje para predecir eventos volcánicos utilizando la información almacenada durante un periodo de tiempo con los algoritmos de redes neuronales.

Para ello las Redes Dinámicas y las Redes Recurrentes Retardadas en el Tiempo relacionadas mediante las memorias, (*Gamma* y *Laguerre* [10]), son el mejor modelo para realizar el trabajo de predicción de fenómenos naturales.

Las topologías de Redes Dinámicas son una clase muy importante de Redes Neuronales ya que son hábiles para procesar varias señales en el tiempo [10]. Una red Neuronal Dinámica se crea a partir de una Red Neuronal Estática con un mecanismo de memoria extendida que es capaz de almacenar valores anteriores de los datos correspondientes a los valores de entrada [10]; la memoria es importante ya que permite decisiones con base en el comportamiento de la entrada en un periodo de tiempo. Un predictor estático hace decisiones con base solamente a la entrada actual, por lo que puede desempeñar funciones que no involucran conocimiento sobre la historia de la señal de entrada [10].

Las Redes Recurrentes Retardadas en el Tiempo tienen como estructura de memoria los filtros lineales donde la memoria es una referencia de lo esperado en línea, o de lo almacenado en el pasado de la entrada [12]. Las conexiones recurrentes se usan como memoria y son llamadas unidades de contexto [12]. Las Redes Recurrentes Retardadas en el Tiempo con capa de memoria restringida a la salida pueden representarse como un procesador de entrada [12]. Lo importante de este tipo de red es la representación de la información en el tiempo en vez de la información dada únicamente por los valores de entrada. Este tipo de redes utiliza la memoria Gamma, que se identifica como una estructura de conexiones en cascada recurrentes a ellas mismas, es una estructura de retroalimentación local que amplía la unidad de contexto con almacenamiento mucho más versátil [12]. Las Redes Recurrentes Retardadas en el Tiempo son la mejor opción para proponer un módulo de predicción en el Sistema de Monitoreo Volcánico en Tiempo Real con el fin de prever alguna variación anormal en el campo geomagnético del volcán Popocatepetl y así evitar alguna catástrofe, con el fundamento de la

informática bio-inspirada y las Redes Neuronales para construir sistemas complejos.

IV SISTEMA PROPUESTO

Los sistemas de monitoreo y vigilancia son muy importantes en la evaluación y en los pronósticos de la actividad de los volcanes. Por lo que es vital pronosticar, monitorear y evaluar la actividad volcánica de forma automatizada.

Se tiene como objetivo general proponer un sistema automatizado en tiempo real para la monitorización del volcán Popocatepetl que facilite la gestión de datos en la prevención de desastres naturales. Como objetivos específicos de la aplicación se consideran los siguientes puntos:

- [1] Crear una implementación para un sistema empotrado (embedded) basado en los datos de los magnetómetros (dispositivos de hardware).
- [2] Proporcionar información automatizada de la actividad volcánica a la ciudad más próxima al volcán Popocatepetl: Puebla.
- [3] Obtener información inmediata para implementar acciones de prevención, protección civil, alerta y evacuación.
- [4] Predecir algún evento anormal dentro de la actividad magnética del volcán, mediante el estudio de algoritmos de redes neuronales y su futura implementación gracias al modelo de una Red Recurrente Retardada en el Tiempo para pronosticar eventos volcánicos [4, 10].

El sistema está dentro del nivel de exigencias de tipo A, teniendo riesgo de pérdidas humanas numerosas (catástrofe), desastre financiero, servicios de información y seguridad críticos para las empresas, y deterioro del medio ambiente irreversible según la Norma internacional *ISO/CEI 9126* que define los niveles de exigencia para un dominio de aplicación, según los riesgos o los daños ocurridos después de la falla de un *Sistema de Información* [5, 13].

Se utilizan 2 magnetómetros (M1, M2)

sobre 2 puntos estratégicos sobre las faldas del volcán Popocatepetl para monitorear el magnetismo de manera autónoma; dicha autonomía también es referente al modo de alimentación de los dispositivos, ya que mediante un estudio de uso, los dispositivos pueden ser alimentados mediante placas solares y tarjetas electrónicas diseñadas para ello.

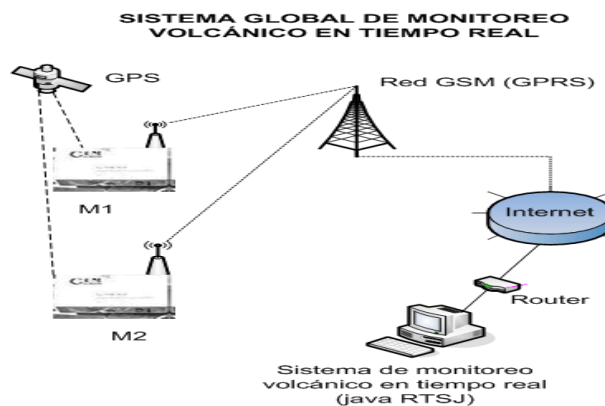


FIG. 2. DIAGRAMA GENERAL DEL SISTEMA.

Dichos dispositivos necesitan de una interfaz de Hardware para transmitir la información magnética de forma inalámbrica a una red de tipo GSM, que a su vez la transmitirá por Internet y finalmente los datos serán recibidos por el Sistema de Monitoreo en Tiempo Real basado en *RTSJ (Real Time Specification for Java)* para efectuar el monitoreo. El sistema también contaría con la comunicación mediante el sistema GPS para identificar la ubicación de los magnetómetros. (Fig.2).

Se propone un sistema de Tiempo Real que use los dispositivos existentes: dos magnetómetros instalados en las cercanías del volcán Popocatepetl que registran la actividad magnética del volcán. Se incorpora un sistema de transmisión de datos mediante la conversión de la señal analógica a digital y transportada por medio inalámbrico hasta el centro de investigación del CUPREDER de la BUAP.

El sistema de cómputo incluye una

computadora con una capacidad de almacenamiento de 250 GB para guardar toda la información proveniente del monitoreo magnético en una Base de Datos creada para ello. El motor de Base de datos es MySQL. Tiene como Sistema Operativo alguna distribución de LINUX compatible con las librerías del RTSJ de JAVA para los sistemas de tiempo real i.e.: SUSE 10, DEBIAN, Fedora, y como nueva propuesta Red Hat y FREE BSD. El software está desarrollado utilizando las librerías del RTSJ que reciben las señales enteras provenientes de los magnetómetros. Posteriormente dichos datos se guardan y son interpretados por otro programa con una interfaz gráfica de usuario (GUI). El proyecto tiene integrado un sistema de alarma automático, que alertará a las autoridades y personas competentes automáticamente en caso de registro de actividad volcánica anormal.

V METODOLOGÍA DEL SISTEMA DE TIEMPO REAL

El sistema fue diseñado para el monitoreo de la información magnética del volcán Popocatepetl recibida en tiempo real, utilizando herramientas de última generación. La metodología de ingeniería de software utilizada para desarrollar el sistema fue SA-RT (*Structured Analysis for Real-Time*)[6, 9], que nos proporciona una visión coherente y estructurada sobre el desarrollo de sistemas de tiempo real, así como la implementación del lenguaje formal LACATRE (*Langage d'Aide à la Conception d'Applications en Temps Réel*) [7, 14, 11] para comprender fácilmente la estructura del programa desarrollado para efectuar el monitoreo volcánico.

El diseño del software está representado mediante el lenguaje formal gráfico LACATRE para representar las aplicaciones desarrolladas en tiempo real, y facilitar el diseño del sistema. (ver Fig. 3). El programa fuente se divide en diferentes módulos; la función principal P1, *main* () trabajará directamente con los hilos (*Threads*) siguientes:

- P2: MAGNETOMETRO_1, con prioridad 1.
- P3: MAGNETOMETRO_2, con prioridad 2.
- P4: MONITOREO, con prioridad 3.
- P5: ALARMA, con prioridad 4.
- P6: PREDICE, con prioridad 5.

Todos los hilos están sincronizados mediante semáforos para asegurar que todos los procesos estén sincronizados y con los datos correctos. Las señales provenientes de los magnetómetros son digitalizadas y obtenidas mediante un protocolo de comunicación de datos, y son transmitidas mediante un Recurso de Datos (*R.S.*) a los hilos P2 y P3. En la actividad actual se muestra el llamado de la función *MOSTRAR* () que contiene los procedimientos para mostrar los datos numéricos en pantalla. Posteriormente los datos numéricos se escriben en una Base de Datos (*B.D.*) y los mensajes se envían a una estructura de datos de tipo FIFO de tamaño n , llamada *MsgA*. Estos mensajes son transmitidos de manera independiente al hilo P4 quién inicia el monitoreo de la actividad mediante el llamado a las funciones: *GRAFICAR* (), *M_CRITICO* () y *M_NORMAL* (). Estas funciones contienen la información magnética del volcán, se actualiza en un tiempo determinado, y se almacena en la B.D.

Si existe una actividad anormal, la función *M_CRITICO*(), envía un mensaje a otra estructura de datos FIFO llamado *MsgB* que contiene el tipo de anormalidad y su grado de peligro. Los mensajes de *MsgB* son transmitidos al hilo P5, que escribe un informe del problema en la Base de Datos y posteriormente se activa una alarma dependiendo del tipo de severidad del problema. Esto se mide mediante el tipo de datos n proveniente de *MsgB*. La función *GRAFICAR*() muestra un mensaje de la actividad anormal en pantalla mediante el llamado de la función *MOSTRAR* () .

El sistema también cuenta con un proceso P6 llamado *PREDICE* que tiene acceso a la Base de Datos y a la función *PREDECIR* (). Esta

función hace una predicción de la actividad magnética del volcán usando algoritmos de redes neuronales [8]. En caso de un evento anormal, se enviará un mensaje a *MsgB* y activará otro tipo de alarma para prevenir sobre un posible evento anormal o una catástrofe.

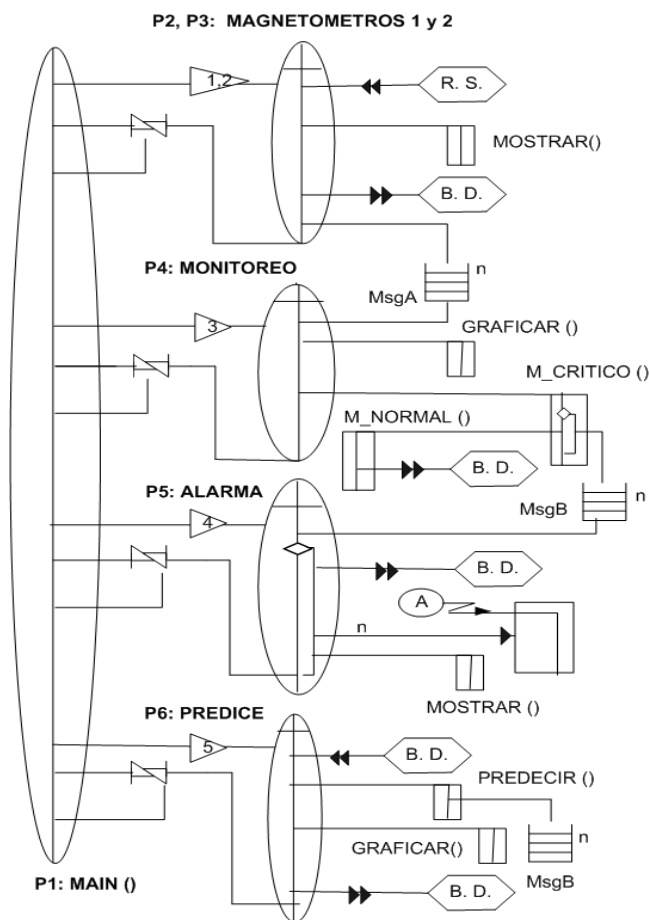


Fig. 3 . Diagrama LACATRE

VI CONCLUSIÓN Y TRABAJO FUTURO

Las principales aportaciones son: (a) Dar una propuesta de aplicación de las Redes Neuronales Artificiales para predecir eventos volcánicos de importancia. (b) Contar con un sistema de última generación para el desarrollo y uso de la tecnología en México; (c) se promueva la colaboración de diferentes disciplinas.

Para el trabajo a futuro, el sistema general gracias a la ayuda del concepto de las Redes

Neuronales Artificiales, será implementado en forma real una Red Recurrente Retardada en el Tiempo que podrá arrojar las primeras predicciones después de cierto tiempo, con la información recolectada con el módulo *Predice* del sistema. También se propuso los fundamentos teóricos para la implementación del módulo *P6 PREDICE* en el software para realizar predicciones de eventos volcánicos. Este tema se puede profundizar y posteriormente implementar como el proyecto realizado para pronosticar la velocidad el viento en el gran *Telescopio Milimétrico* en el INAOE, Puebla con el proyecto titulado: “Una Red Recurrente Retardada en el Tiempo para Pronosticar la Velocidad del Viento y su Frente de Onda” [10].

Sus principales aportaciones fueron el diseño de ingeniería de una parte del Sistema de Monitoreo Volcánico en Tiempo Real, su justificación para profundizar en el tema, y la metodología utilizada para el diseño e implementación del sistema de software y de la arquitectura general del Sistema.

VII REFERENCIAS

- [1] La Erupción Inalámbrica de un Volcán, http://astrobiologia.astroseti.org/astrobio/articulo_1_515_la_erupcion_inalambrica_volcan.htm revisado en 2008.
- [2] Automated System for Magnetic Surveillance of Active Volcanoes, <http://maglab.ct.ingv.it/en/index.html> revisado en 2008.
- [3] Enrique Guevara O, Roberto Quaas W, Carlos Valdés, Gilberto Castelán P, Alicia Martínez B, Javier Ortiz C., Francisco Galicia y Ángel Gómez V. “El sistema de monitoreo y vigilancia del volcán Popocatepetl. 6.5 años de operación.” Centro Nacional de prevención de Desastres, CENAPRED, México. Página de internet: <http://www.ugm.org.mx/pdf/geos01-3/REDES01-3.pdf> revisado en 2008.

- [4] BELSON Guillaume, cours 5IF “*Informatique Bio-Inspirée 5IF-IBI*” (Année 2007-2008), Institut National des Sciences Appliquées de Lyon, (INSA de Lyon, France). (Notas de curso).
- [5] AUBRI Régis, cours 4IF “Assurance Qualité logicielle 4IF-QL” (Année 2007-2008), Institut National des Sciences Appliquées de Lyon, (INSA de Lyon, France). (Notas de curso).
- [6] Babau, Jean-Philippe, SA-RT Structured Analysis for Real-Time. <http://www.if.insa-lyon.fr/chercheurs/jpbabau/cours/sart.pdf> Revisado en 2008.
- [7] BABAU Jean-Philippe, cours 4IF “Systèmes d'Exploitation avancés 4IF-SEA, *conception Multitâches Lacatre/VxWorks*”, (Année 2007-2008), Institut National des Sciences Appliquées de Lyon, (INSA de Lyon, France). (Notas de curso).
- [8] Neural Network Frameworks, <http://www.redes-neuronales.netfirms.com/tutorial-redes-neuronales/que-son-las-redes-neuronales.htm> revisado en 2008.
- [9] Niño Omar, Colmenares Enrique, Martín Manuel, “Sistema de Monitoreo Volcánico en Tiempo Real” Paper submit para el congreso de IEEE Andescon 2008.
- [10] Soancatl Aguilar Leobardo, “Una Red Recurrente Retardada en el Tiempo para Pronosticar la Velocidad del Viento y su Frente de Onda”, Tesis de Maestría, BUAP 1999.
- [11] J. J. SCHWARZ, J. J. SKUBICH, Graphical programming for Real-Time Systems, *Control Engineering Practice*, Vol. 1, No. 1, pp. 43–49, 1993.
- [12] Lang K., Waibel A. and Hinton G. “A time delay neural network architecture for isolated word recognition” *Neural Networks* 3 (1), 23-44. 1990
- [13] J.A. McCall, Factors in Software Quality, General Electric no. 77C1502, juin 1977.
- [14] Piotr Szwed, *Lacatre Reference Guide*, programming tool, http://pszwed.ia.agh.edu.pl/RT/La4_rm/La4_rm.html revisado en 2008.

Niño Prieto Omar Ariosto, Nació en Puebla, Pue. México el 2 de enero de 1984. Realizó estudios de preparatoria en el Instituto Oriente de Puebla, Bachillerato CCH incorporado a la UNAM. Ingresó a la Licenciatura en Ciencias de la Computación en la BUAP (2002), realizando una estancia de estudios en Université Laval, Quebec, Canadá, (2005-2006). Posteriormente terminó los créditos obligatorios de la licenciatura en la FCC-BUAP (2006-2007) y realizó una estancia de estudios en l'Institut National des Sciences Appliquées de Lyon, (INSA de Lyon, Francia) (2007-2008) para finalizar el currículo de la Licenciatura. El Sistema de Monitoreo Volcánico en Tiempo Real, es la Tesis de licenciatura realizada para obtener el grado de Licenciado en Ciencias de la Computación, aplicando los conocimientos adquiridos en la BUAP y en el INSA de Lyon.

Durante el verano 2005, realizó prácticas profesionales en el Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE) en el departamento de informática. Tiene la asistencia a diversos congresos de computación e informática en Puebla, así como el dominio de los idiomas Español, Francés e Inglés. Sus intereses son los Sistemas de Tiempo Real, las Imágenes Digitales, la Informática Bio-inspirada la Inteligencia Artificial, así como la Organización y Gestión de la Producción en las empresas utilizando los Sistemas y Tecnologías de la Información.

Luis Enrique Colmenares Guillén, Realizó sus estudios de la Licenciatura en Computación en la Benemérita Universidad Autónoma de Puebla en la Facultad de Ciencias Físico-matemáticas. Los estudios de Maestría en la Universidad de las Américas Puebla obteniendo el título de Maestro en Ciencias. El doctorado fue realizado en la Universidad Politécnica de Cataluña en Barcelona España, en la especialidad de Sistemas distribuidos

en el Departamento de arquitectura de computadores.

Actualmente está como profesor investigador de tiempo completo en la Facultad de Ciencias de la Computación de la Benemérita Universidad autónoma de Puebla. Imparte asignaturas de las curriculas de Licenciatura, Ingeniería y Postgrado. Pertenecer al cuerpo académico de Sistemas de Información, y colabora con la Dirección General de Innovación educativa. Esta en diferentes proyectos de Sistemas Distribuidos, Sistemas de Tiempo Real, Cómputo Pervasivo y Cómputo Ubicuo.

Manuel Martín Ortiz, Realizó sus estudios de Física en La facultad de Ciencias físico-matemáticas de la Benemérita Universidad Autónoma de Puebla, obtuvo su maestría y doctorado en el Instituto de Física de la misma universidad.

Actualmente está como profesor investigador de tiempo completo en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. Imparte asignaturas de las curriculas de la Licenciatura, Ingeniería, Postgrado y Diplomado. Es responsable del cuerpo académico de Sistemas Distribuidos y del Centro de Información "Alejandro Reyes Cristiani" de la F.C.C. Es responsable de la "Olimpiada Mexicana de Informática" a Nivel Estatal (Puebla), es miembro de la Comisión de Tesis de la F.C.C. y es corresponsable del Area de Servicios de Red de la F.C.C. Tiene relación de numerosos artículos nacionales e internacionales arbitrados así como numerosas Tesis dirigidas de nivel licenciatura y postgrado.

Es Miembro de la American Computer Machinery (ACM) y tiene experiencia como profesor desde 1978 en la enseñanza superior. Cuenta con 18 publicaciones Nacionales y Locales y 12 publicaciones internacionales arbitradas. Ha impartido diversas conferencias a nivel nacional e internacional.

SISTEMA DE CONTROL PARA UNA CAMA VIBRATORIA PARA PRUEBAS OSCILATORIAS

Andrés G. Fuentes Covarrubias

Instituto de Innovación y Desarrollo Tecnológico,
Facultad de Ingeniería Mecánica y Eléctrica, Universidad de Colima
Coquimatlán, Colima 28400, México
e-mail: fuentesg@ucol.mx

Ricardo Fuentes Covarrubias

Instituto de Innovación y Desarrollo Tecnológico,
Facultad de Ingeniería Mecánica y Eléctrica, Universidad de Colima
Coquimatlán, Colima 28400, México
e-mail: fuentesr@ucol.mx

Ricardo E. Macías Fernández

Facultad de Ingeniería Mecánica y Eléctrica, Universidad de Colima
Coquimatlán, Colima 28400, México

David Díaz Delgado

Facultad de Ingeniería Mecánica y Eléctrica, Universidad de Colima
Coquimatlán, Colima 28400, México

RESUMEN

El presente trabajo trata sobre el diseño e implementación de un sistema de control en tiempo real para un simulador sísmico. Dado que el estado de Colima, se encuentra dentro de una zona considerada de alta sismicidad; conocer el comportamiento que guardan los materiales de construcción de casas-habitación ante un evento sísmico es crítico. El instrumento a medir es una plataforma de concreto armado de 4 m², la cual constituye el espécimen de estudio. Esta plataforma es de un grosor de 15 cm., lo cual es común en las edificaciones de la zona. El sistema de poder para producir las vibraciones del espécimen esta conformado por cuatro actuadores neumáticos que provocan una vibración en el eje vertical y dos actuadores en un eje horizontal. El control se efectúa mediante válvulas neumáticas proporcionales y como señal de control se generan señales PWM que deben reproducir la simulación de un evento sísmico. Para la generación del sistema de vibraciones se utiliza un sistema embebido y un sistema operativo para tener un control total sobre la temporización y sincronía de las

señales de control. Para el estudio de las vibraciones generadas se utilizaron dos geófonos sísmicos calibrados en el eje vertical.

Palabras Claves: Sistema embebido, tiempo real, procesamiento digital de señales, Control, eventos sísmicos.

1. INTRODUCCIÓN

Actualmente el Laboratorio de Instrumentación y Robótica de la Facultad de Ingeniería Mecánica y Eléctrica atiende a cuatro líneas de investigación vinculadas con el Cuerpo Académico UCOL-CA46, “Sistemas Inteligentes”, una de ellas es Control Basado en Microprocesador. Como parte de las actividades para lograr el arranque de esta LGAC, se ha trabajado con la Facultad de Ingeniería Civil en la rehabilitación y puesta en marcha de los equipos de monitoreo sísmico SISMO1. Esos equipos están dotados de un acelerógrafo con el fin de adquirir datos y caracterizar movimientos telúricos. Por otro lado, es posible la caracterización de mecánica de suelos mediante estudios de prospección geofísica provocando ondas sísmicas de

manera artificial mediante un golpe de ariete en las inmediaciones del terreno a estudiar.

2. ANTECEDENTES

Los equipos de sensado sísmico[2] SISMO1 es un dispositivo de almacenamiento de señales digitales sísmicas, el cual automáticamente detecta la señal sísmica, almacena la onda sísmica y transmite estadísticas relacionadas con la actividad sísmica a través del sistema Argos. SISMO1 está diseñado para implementar estaciones sísmicas portátiles de largo término, para aplicaciones en campo y monitorear la actividad sísmica, cuenta con un acelerógrafo unidireccional lo cual permite la perspectiva de ser utilizado en aplicaciones de estudios de prospectiva geofísica.

La Prospección Geofísica es la caracterización del subsuelo basada en las variaciones laterales o verticales de las distintas propiedades físicas de los materiales que lo componen, mediante mediciones no intrusivas, de estas distintas propiedades [3].

Es una herramienta de ayuda a la hora de estudiar el subsuelo que nunca sustituye cualquiera de los métodos directos aplicados a ese estudio (Sondeos mecánicos), pero si delimita las zonas de interés y ayuda a localizar las estructuras que se buscan.

Todas las técnicas geofísicas intentan distinguir o reconocer las formaciones geológicas que se encuentran en profundidad mediante algún parámetro físico, por ejemplo en sísmica por la velocidad de transmisión de las ondas o en prospección eléctrica por la resistividad del suelo.

De lo anterior se desprende el que se puede lograr el análisis de mecánica de suelos mediante una solución eléctrica, sin embargo ello involucra el tener que efectuar el diseño del equipo de monitoreo completo.

Eventos sísmicos en la región [1]

La caracterización sísmica en México es importante dado que bordea su territorio la placa de subducción continental, en el caso del estado de Colima confluyen dos fallas

geológicas las cuales son la de “cocos” y la “clarión”, esto ha provocado que en las últimas décadas la región sea epicentro de eventos sísmicos de considerable magnitud:

AÑO	INTENSIDAD RICHTER	EPICENTRO
1972	6.6	COSTA DE JALISCO
1973	7.6	COSTA DE MICHOACAN
1981	7.3	COSTA DE MICHOACAN
1985	8.1	COSTA DE GUERRERO
1986	7.0	COSTA DE MICHOACAN
1989	6.5	COSTA DE COLIMA
1995	8.0	COSTA DE COLIMA
2003	7.6	COSTA DE COLIMA

Tabla 1. Eventos sísmicos más importantes en la región [1]

La estructura

De la tabla 1 se desprende el que la actividad en la región es importante, por ende el daño a las edificaciones y a la población civil justifica el contar con una herramienta de simulación que permita en primer instancia el poder efectuar pruebas en los elementos estructurales y en segunda instancia servir como plataforma para educación y entrenamiento de la población en general.

En la siguiente figura se aprecia un detalle de la republica mexicana y las principales fallas y su convergencia con la placa continental:

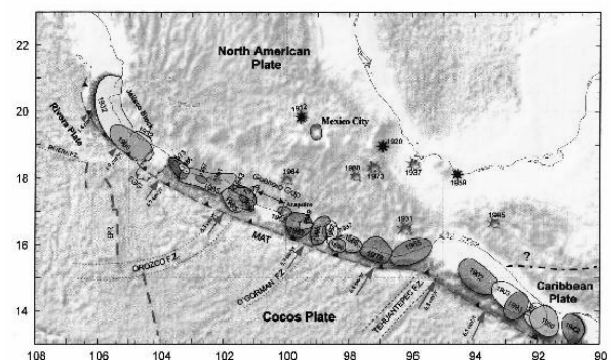


Figura 1. Principales fallas tectónicas de la región [1]

3. EL PROTOTIPO PROPUESTO

Estructura

Un prototipo de cama sísmica es el medio común de efectuar pruebas sobre elementos estructurales para que el profesional de la

ingeniería civil en México pueda inferir modelos de comportamiento en mecánica de materiales ante fenómenos de naturaleza vibratoria. A nivel nacional pocas son las instituciones que cuentan con un dispositivo de esta naturaleza, este es un elemento de dos grados de libertad y se trata de examinar los modelos en cuanto a su comportamiento en el eje Y (tremores sísmicos) y en el eje X (movimientos oscilatorios), los cuales son característicos y típicos del movimiento ondulatorio provocado por un evento.



Figura 2. Detalle de la estructura de la cama sísmica

En la figura 2 se puede apreciar un detalle de la estructura diseñada, los apoyos para los actuadores asimismo en la parte superior de la plancha del espécimen un elemento sensor o acelerómetro para graficado y retroalimentación de la señal.

El sistema de control

El sistema está compuesto por un sistema embebido con base en el microcontrolador PIC18F452. La razón de utilizar este microcontrolador fue que sus recursos permiten montar un pequeño sistema operativo a este. Por otro lado, el sistema operativo elegido es de arquitectura abierta y en base a la norma OSEK/VDX. Un juego de actuadores neumáticos permite articular los dos grados de libertad de la estructura. Finalmente un sistema de sensorica implementado por el SISMO1.

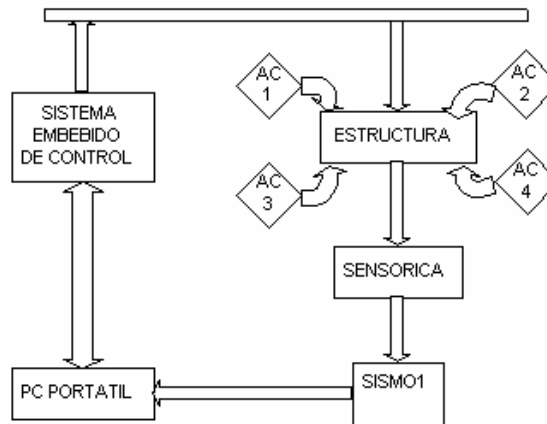


Figura 3. Diagrama a bloques del sistema

Para la parte de control, se propone el uso de una tarjeta de desarrollo basada en el PIC18F452 corriendo a 4 Mhz, Un programador del microcontrolador para bajar los programas al PIC. El lenguaje base será el C++ y se proponen dos RTOS como base, el FreeRTOS por Richard Barry y el PICOS18-OSEK por la empresa SoftElect y Pragmatec, dado que ambos son de dominio público protegidos por la GPL del GNU y permiten su uso y modificación sin el pago de regalías.

Razones de usar un sistema operativo

Hay objetivos claramente definidos que establecen las razones para usar un Sistema Operativo en una aplicación. En primera instancia, es necesario crear una plataforma de uso transparente al usuario de los recursos informáticos, si un usuario de una aplicación necesita controlar la velocidad de un motor o producir el movimiento de un actuador, este no debe saber si se requiere activar un bit o un puerto. También, los desarrolladores de aplicaciones necesitan una plataforma más amigable y transparente de acceso a los recursos, esto produce código más portable e intuitivo. Hay que pensar que si un programador está desarrollando una aplicación o un módulo de un sistema y requiere hacer uso de un recurso, tal como un puerto serial o un timer, le sería muy difícil reescribir el código para cada plataforma en la cual está desarrollando.

Al desarrollar una aplicación en un sistema embebido, el programador o diseñador de sistemas debe preguntarse si por la complejidad del problema vale la pena usar un sistema operativo. Por lo general, para solucionar problemas pequeños, y si los recursos del microprocesador o microcontrolador son suficientes, un programa en base a una secuencia es mas que suficiente ya que el nivel de complejidad es bajo. Sin embargo, para aquellas soluciones en las cuales se requiere ejecutar mas de un proceso, que los procesos compartan estructuras de datos, o bien, que estos tengan que competir por el uso de un recurso, una solución en base a una secuencia no es lo suficientemente eficaz para la administración de los recursos, asimismo, para obtener una respuesta en tiempo lo suficientemente como para que se hable de una aplicación en tiempo real.

4. EL KERNEL DEL OSEK

Definición

La especificación del sistema operativo OSEK representa un ambiente uniforme el cual soporta la utilización eficiente de los recursos para el control automotriz y de sistemas empotrados y de unidades de control y sistemas empotrados. El sistema operativo OSEK esta enfocado a un procesador único lo cual significa que deberá ser utilizado en unidades de control o sistemas empotrados utilizando multitarea.

La especificación de servicios OSEK/VDX

Manejo de procesos: Activación y terminación de tareas, Manejo de estado de procesos e interrupción de tareas (preemptive task control)

Sincronización de tareas: Manejo de recurso, control de acceso por operaciones inseparables a los recursos vinculados a un dispositivo, control por medio del flujo de programa

Manejo de interrupciones: Servicios para el manejo de interrupciones

Alarmas: Alarmas relativas y absolutas, de tipo estáticas (definidas en tiempo de compilación) y dinámicas (definidas en tiempo de ejecución)

Manejo de excepciones: Manejo de errores, mecanismo soportado por el usuario en caso de varios errores, el código de la aplicación deberá manejar las excepciones

Especificación del sistema operativo

La versión v1.11 de respeta la norma para los OS estándar de OSEK. Por otra parte, no es al momento 100% compatible con este estándar y algunas fallas pueden persistir en el código. El objetivo básico es así, la reunión de los componentes básicos del kernel de este estándar, después integrar los medios de comunicación definidos por COM y finalmente agregar a OSEK los servicios de red de acuerdo con el estándar.

Características principales del kernel de OSEK:

Tipo de kernel	Multitarea, preemptive
Procesador	Familia microchip PIC18
Requerimientos del kernel (ROM)	< 1 kb
Requerimientos del kernel (RAM)	7 octetos
Requerimientos de los servicios (ROM)	4 kb
Requerimientos de los servicios (RAM)	121 octetos
Tamaño de la pila hardware	32 llamadas a funciones para todos los procesos
Tamaño de la pila software	128 octetos
Tiempo de latencia del scheduler	25uS, FreqXtal=4Mhz
Número de procesos simultáneos	8
Número de eventos por proceso	8
Número de prioridades	8
Número de timers lógicos	Sin limites
Tamaño del contexto de un proceso	20 octetos

Tabla 2. Especificaciones del sistema operativo

Estas características son suficientes para el desarrollo de aplicaciones de mediana envergadura con la familia de microcontroladores de microchip PIC18xxx, para el caso de este tutorial, se trabajará sobre la base del PIC18F452-I/P, ya que cuenta con buenas prestaciones en cuanto a memoria y recursos, además, es compatible pin a pin con el PIC16F877. Es de observar que el kernel, los servicios y la pila monopolizan los primeros 5Kb de memoria de programa así como el primer banco de memoria RAM.

El kernel del OSEK esta compuesto por los siguientes bloques:

CONTROL DE EVENTOS	ADMINISTRADOR DE PROCESOS	CONTROL DE INTERRUPCIONES
CONTROL DE ALARMAS	ADMINISTRADOR DE TAREAS (TASK MANAGER)	
	PLANIFICADOR (SCHEDULER)	ARRANCADOR (BOOTLOADER)
		SERVICIOS DE ATENCION DEL S.O.

Figura 4. El kernel según la especificación OSEK/VDX

En esta versión el tutorial está validado para la versión bajo el ambiente de desarrollo de MPLAB/Windows versión 6.40 con el compilador C18 versión 2.20.04. El microcontrolador es un PIC18F452.

5. RESULTADOS

Sistema embebido implementado

Como se mencionó en párrafos anteriores fue implementada una tarjeta de sistema embebido con base en el microcontrolador PIC18F452 de Microchip. El detalle es el que ilustra la figura 5 a continuación.

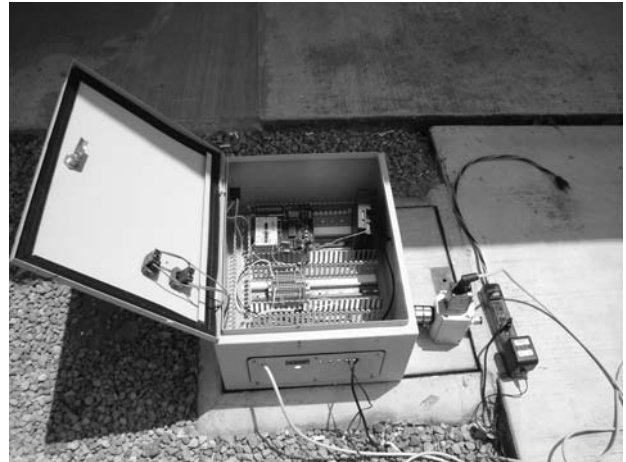


Figura 5. El sistema embebido y válvula proporcional

Estructura final de la cama sísmica

Se efectuaron pruebas monocanal sobre la plancha de concreto, para el caso se utilizó el canal de tremor sísmico sobre el eje Y utilizando cuatro actuadores neumáticos de 90 ciclos x s, a carga plena. Para la prueba se sintetizó una señal senoidal a una frecuencia de 1 ciclo por s enviando muestras PWM cada 10ms.



Figura 6. Detalle de la prueba monocanal

Respuesta del módulo de sensórica y de SISMO1

Durante las pruebas monocanal el sistema de sensórica utilizó un geófono de baja velocidad y para medir aceleraciones verticales. Fue configurado para una ventana de muestreo de

10s , se utilizó un convertidor analógico-digital de 12 bits. Se configuró una corrida de 35 s, a carga plena con una sobrecarga de 160 kg.

Módulos del software

Los módulos del kernel que fueron utilizados son los siguientes:

18f452.lkr	Archivo de la escritura del linker. Fija las diversas secciones en la memoria.
alarm.c, alarm.h	Manejador de alarmas. Punto de entrada del programa. Llama inmediatamente la rutina principal.
boot.c	
even_man.c, even_man.h	Manejador de eventos.
int.asm	Tabla de vectores de interrupción. El TIMER0 INTERNO está situado en el archivo de timer.c.
int_man.c, int_man.h	Manejador de interrupciones.
main.c	Definición de tareas, procedimiento de inicio y llamadas al kernel.
kernel.asm	Base del núcleo. Define y activa la siguiente tarea.
pro_man.c, pro_man.h	Manejador de procesos.
device.h, device.inc	Recoge las definiciones relacionadas con los procesos.
taskdesc.c	Declaración de procesos (prioridad, dirección de inicio del proceso, dirección de la pila del proceso).
timer.c	Manejador de alarmas y contadores.
Tsk_Ac1	Manejador del actuador 1
Tsk_Ac2	Manejador del actuador 2
Tsk_Ac3	Manejador del actuador 3
Tsk_Ac4	Manejador del actuador 4

6. CONCLUSIONES

En general las pruebas preliminares al sistema son satisfactorias, en cuanto al software del sistema embebido su respuesta es satisfactoria en función de que su tiempo de latencia para el software desarrollado es bastante buena, 25 ms, tomando en consideración que se debe controlar un actuador cuya máxima velocidad será de 4 cm. x s., esto considerando que se simularía un tremor similar al del epicentro del sismo. En cuanto a las ondas largas u oscilatorias una frecuencia de 2 ciclos x s, es suficiente para probar especímenes de aproximadamente 200 kgs. x m² .

Es de mencionar que los requerimientos de tiempo de respuesta permiten construir un sistema no estricto en función de que si bien es importante saber que esta ocurriendo un evento, la velocidad de muestreo es baja en función de la temporalidad del sistema.[5]

Sin embargo, cabe hacer mención que en las pruebas monocanal se detectó un sobredimensionamiento de la señal en aproximadamente un 10% por pico. Ello llevará a valorar la eficacia de la etapa de amplificación ya que para simular tremores cercanos a un epicentro con emisión de energía cercano a valores richter arriba de los 6 grados el sobredimensionamiento crecerá de manera logarítmica haciendo inútil el instrumento.

No se efectúan comentarios respecto de pruebas biaxiales debido a que aun no se tienen resultados concluyentes al respecto, en función de que se requiere la preparación del espécimen de prueba por parte de la Facultad de Ingeniería Civil, así mismo su instrumentación, sin embargo esta es una etapa futura del proyecto.

7. REFERENCIAS

- [1] Instituto de Geofísica, Servicio Sismológico Nacional, México: Universidad Autónoma de México, página oficial, www.ssn.unam.mx, 2005.
- [2] SISMO1, Users Manual, 1994.
- [3] Astier, J.L., Géophysique Appliquée a l'Hydrogéologie, Mason, 273 pp, Ed. Paraninfo

- [4] OSEK/VDX, Home Page of OSEK/VDX Operating System; http://www.osek-vdx.org/osekvdx_OS.html, 2004.
- [5] Burns, A., Sistemas de Tiempo Real y Lenguajes de Programación, Ed. Addison-Wesley, 3ª Ed., 2003, pp 2-3.
- [6] Dyke, S. J., 1998, "Design and development of the Washington University seismic Simulator facility", Proc. 12th Eng. Mech. Conf. '98, ASCE, pp. 762-765, UC San Diego USA.
- [7] Nagarajaiah S y E. Godowski, 1998, "Unidireccional Shaking Table for testing small scale structural models: desingn and development", proc. 12th Eng. Mech. Conf. '98, ASCE, pp. 779-782, UC San Diego USA.

SISTEMA AUXILIAR PARA EL DIAGNOSTICO DEL CARCINOMA EPIDERMÓIDE INVASOR EN TEJIDO CERVIX EMPLEANDO TECNOLOGIA LÁSER

Aparicio Castillo Aurora¹, De la Rosa Vázquez José Manuel², Espinoza de los Monteros Aguilar Erick¹, Ávila Salazar Carlos Alberto¹

¹ Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Azcapotzalco
Av. De las Granjas No. 682 Col. Sta. Catarina Del. Azcapotzalco, México, D.F.
Tel. 5729 6000 Ext. 64414 Correo electrónico: eresag@gmail.com

² Sección de Estudios de Posgrado e Investigación ESIME-IPN
Unidad Profesional Adolfo López Mateos, Av. IPN s/n, C.P. 07738, México, D. F

RESUMEN

La fluorescencia inducida por laser es una técnica que ha sido utilizada recientemente para la detección de cáncer in vivo e in vitro. El desarrollo de un sistema óptico para la detección de fluorescencia en muestras de tejido cervicouterino permite obtener la intensidad con que brilla la biopsia así como su longitud de onda para realizar el análisis de las muestras. A través de un láser de N₂ laser, una lente biconvexa, fibra óptica, un espectrómetro, un fotomultiplicador y un osciloscopio, se pueden registrar las diferentes longitudes de onda en una computadora personal para después ser tratadas por una red neuronal y diagnosticar así a la biopsia.

Palabras clave: Fluorescencia, cáncer, cérvix, láser

I. INTRODUCCIÓN

Según la Organización Mundial de la Salud, el cáncer cervicouterino es la segunda mayor causa de mortalidad femenina por cáncer en todo el mundo, con unas 300,000 muertes al año. El 80% de los casos corresponden a los países en vías de desarrollo (Parkin, Whelan et al. 1997) y cerca de 500,000 casos nuevos se presentan cada año. Tan solo en el año 2002 se presentaron 493,243 nuevos casos y 273,505 decesos (Ferlay, Bray et al. 2004).

En México, en el año 2002, se presentaron 12,512 nuevos casos de cáncer cervicouterino, y 5,777 decesos (Ferlay, Bray et al. 2004). Esta enfermedad

fue la primera causa de muerte entre las mujeres mexicanas con cáncer, ocupando un 16.6% del total.

La detección temprana y el tratamiento oportuno de la infección por VPH en lesiones precancerosas pueden prevenir la progresión a cáncer. Los métodos principales de diagnóstico han sido histopatología y métodos citológicos como el Papanicolaou, que busca cambios en las células de la zona de transformación, así como el estudio de colposcopia (López and Lizano 2006). Recientemente, se han utilizado nuevas técnicas de diagnóstico tales como la colposcopia digital y la espectroscopia de fluorescencia.

Estos nuevos sistemas podrían ayudar a reducir los porcentajes de falsos negativos (entre un 15 y un 50 %) y falsos positivos (entre un 5 y un 10 %) respecto de los métodos tradicionales como la citología y la histopatología, y así evitar el sobre-diagnóstico y sobre-tratamiento de las pacientes.

Particularmente, en la técnica LIF (Laser Induced Fluorescence) se analizan los espectros de emisión de fluorescencia que son provocados por una fuente de emisión de luz; un espectro de emisión de fluorescencia es la construcción de una gráfica de la intensidad de la fluorescencia en función de la longitud de onda.

II. DESARROLLO DEL SISTEMA

Se plantea diseñar y construir un sistema que permita medir los espectros de fluorescencia

emitidos por una biopsia para que posteriormente se pueda diagnosticar.

Los espectros de fluorescencia son obtenidos midiendo la emisión de luz de la muestra de tejido cervical bajo estudio cuando se le excita con una fuente que emite una longitud de onda específica (De la Rosa and Bautista F. 2005).

Como fuente de excitación se utilizó un láser de N₂ (Majumder S., Kumar et al. 2004) cuyo haz es alineado hacia el sistema divisor de haz por medio de dos espejos, el divisor de haz se utilizó para reflejar una parte de esa luz hacia un fotodetector de referencia que sirve para medir la potencia del láser y sincroniza la operación del osciloscopio. La luz que no se refleja llega a una lente biconvexa que la enfoca sobre la muestra que se este analizando, la fluorescencia emitida por esta se colecta, utilizando un par de esferas de vidrio, en una fibra óptica que le lleva al espectrómetro para realizar su análisis espectral. El fotomultiplicador convierte las intensidades ruinosas en forma de voltaje para poder ser monitoreada en el osciloscopio.

Éste último envía las formas de onda, por medio de un interfaz GPIB, a una PC que se encarga de registrarlas e ir extrayendo la información para generar el espectro de fluorescencia. El espectrómetro es controlado desde la PC por medio de un puerto serie (RS-232). El software fue desarrollado en LabView 6 de National Instruments.

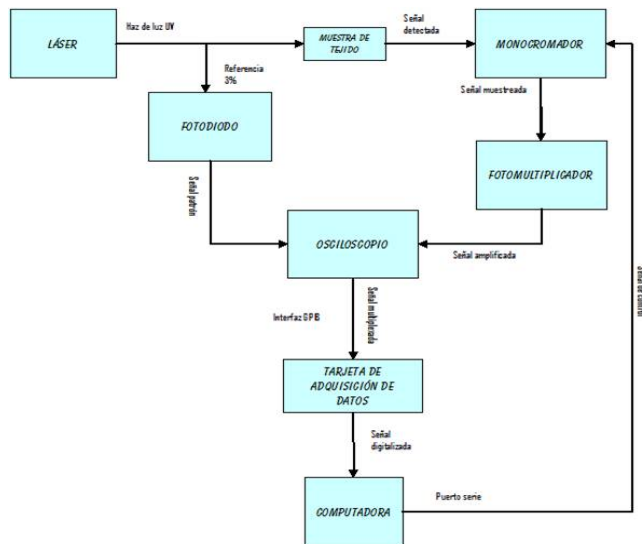


Fig. 1 Diagrama de bloques del arreglo experimental

Método

Fuente de Excitación (Láser de Nitrógeno(Thomas, Chakrapani et al. 1977))

La fuente de excitación es un láser de nitrógeno, que emite pulsos de luz con una longitud de onda de 337.1nm (UV) y tienen una duración de 4ns.El diagrama eléctrico del láser de nitrógeno se muestra en la Figura 2

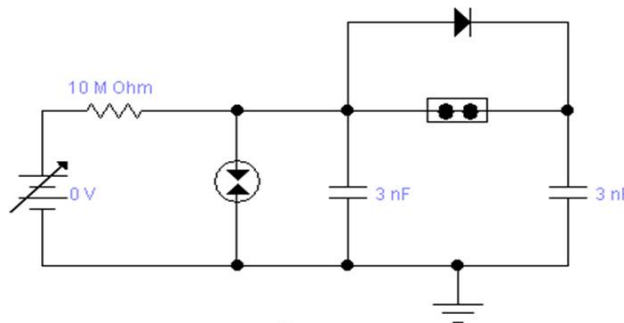


Fig. 2 Diagrama eléctrico del láser de nitrógeno.

La tasa de repetición de pulsos del láser para estas mediciones se estableció en 35Hz aproximadamente, aunque puede variar dependiendo de su tensión de alimentación y la separación entre las esferas del interruptor de chispa.

Las partes que conforman el láser de nitrógeno son las siguientes:

- Fuente de Alta Tensión.- La fuente de alta tensión va desde 0v hasta 18kV la cual sirve para alimentar la cámara de descarga.
- Regulador de Voltaje.- Este dispositivo permite regular el voltaje de entrada hacia la cámara de descarga. El regulador se coloca en el valor de 60% de los 18kV para que comience la descarga eléctrica en la cámara de descarga.
- Bomba de vacío.- Este dispositivo se encarga solamente de extraer el aire que pueda estar contenido en la cámara de descarga y establecer una presión de 60mbar de N₂ dentro de la cámara de descarga.
- Interruptor de chispa.- Es un interruptor con aire como dieléctrico que sirve para iniciar la descarga en el circuito y tiene las siguientes dimensiones:
 - Altura: 50mm
 - Anchura: 60mm
- Cámara de descarga.- Realizada en Vidrio y sellada con resina epóxica, los electrodos son de aluminio y tienen forma de V.
- Nitrógeno.- Gas industrial

En la figura 3 se muestra físicamente el circuito de excitación empleada en el sistema LIF.



Fig. 3 Laser de nitrógeno.

Sistema óptico (Álvarez, Franco et al. 2005)

El sistema óptico es el encargado de adecuar la luz del láser para dividirla en el haz que servirá como referencia y el haz que será concentrado y enfocado en la muestra que se esté estudiando. También consta de lentes que se encargan de concentrar la mayor cantidad de luz emitida por fluorescencia de la muestra y dirigirla a una fibra óptica.

El sistema óptico consta de los siguientes elementos:

- Divisor de haz (vidrio portaobjeto).- Permite desviar 18.2% de la luz del haz láser incidente hacia el fotodetector de referencia.
- Lente biconvexo (BK7) concentran el haz de luz en una sola zona de 1.7 mm x 0.9 mm.
- Fotodetector MRD500.- El cual esta aislado dentro de un tubo de aluminio, y cuenta con una salida hacia el osciloscopio.
- Fibra óptica (cuarzo).- Para conducir la fluorescencia hacia el espectrómetro.



Fig. 4 Arreglo óptico

Divisor de haz

Como divisor de haz se utiliza un vidrio portaobjetos, el cual tiene un índice de reflexión (n_i) de 1.5 y un espesor de 1.0mm. El láser incide

en él con un ángulo de 45° , el arreglo básico se muestra en la figura 5.

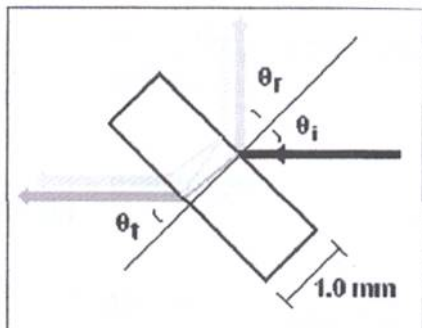


Fig. 5 Divisor de haz

Enfoque de la Radiación sobre la Muestra

En esta etapa del sistema se utiliza un lente biconvexo de un diámetro de 3.5cm y una longitud focal de 15cm. que se encuentra después del divisor de haz y tiene el fin de concentrar la luz transmitida en una zona reducida de la muestra.

Una característica de la luz emitida por el láser de N_2 es su alta divergencia conforme se aleja de su origen, debido a esto es necesario volver a concentrar este haz en un área lo mas pequeña posible para obtener mayores intensidades de fluorescencia. Las dimensiones del haz una vez enfocado en la muestra son de 1.7mm x 0.9mm logrando con esto un área de 1.53 mm^2 .

En el diagrama de la figura 6 se muestran los detalles.

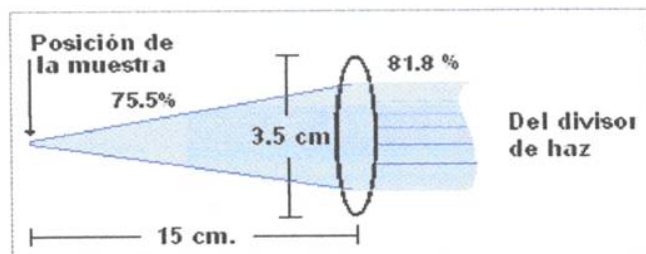


Fig. 6 Lente biconvexo para enfocar la figura.

El ángulo de incidencia en la lente es de 0° por lo que la reflexión es del 4%, al entrar a la lente. Como en este lente también existen reflexiones internas, los porcentajes de luz transmitida y reflejada son:

Luz reflejada = 7.68%
Luz transmitida = 92.3%

Se estima que la energía luz total con que se radia la muestra es el 75.5% de la luz emitida por el láser, equivalente a $0.83 \mu\text{J}$.

Concentración de la Fluorescencia

Una vez que el haz ha sido dirigido a la muestra, la fluorescencia será emitida en todas direcciones. La porción de luz detectada será determinada por la óptica de colección de la fluorescencia.

El propósito de esta etapa es captar la mayor parte de la fluorescencia provocada por el haz láser y dirigirla a un punto que puede ser directamente un fotodetector o una fibra óptica (como en este caso).

Para la realización del sistema de colección se utilizó un par de lentes esféricas (Edmund Industrial Optics L32-748) en una configuración para acoplar fibras ópticas. Estos lentes esféricas son unas excelentes herramientas para mejorar el acoplamiento de la señal entre fibras ópticas, emisores y detectores ópticos. El material de que están hechos es BK7 y el índice de refracción es 1.51, en este caso los lentes esféricos tienen un diámetro de 10mm.

Sistema espectrométrico (Álvarez, Franco et al. 2005)

Esta parte la componen la fibra óptica, el espectrómetro, el tubo Fotomultiplicador y el fotodetector de referencia. El trabajo principal del sistema espectrométrico es tomar la muestra de fluorescencia y obtener la intensidad de la variación en función de la longitud de onda.

Este sistema está constituido por los siguientes elementos:

- Fibra óptica.- La fibra óptica se encarga de conducir la fluorescencia inducida por el láser hacia el espectrómetro, mismo que la descompondrá en sus componentes, la

intensidad de cada componente será convertida por el fotomultiplicador a señal eléctrica y observada en un osciloscopio.

- Espectrómetro.- Su función es la de realizar el análisis espectral de la luz fluorescente, la intensidad de cada componente espectral es convertida por el fotomultiplicador a señal eléctrica y observada en un osciloscopio. El espectrómetro es controlado por la PC por medio de la interfaz serial (RS-232).
- Fotomultiplicador.- Como se menciona anteriormente este dispositivo convierte la irradiancia luminosa en voltajes, los cuales son capturados por el osciloscopio para después ser enviados a la tarjeta GPIB.

Fibra óptica

La fibra óptica conduce la fluorescencia emitida por la muestra hacia el espectrómetro.

La fibra óptica utilizada es de cuarzo, diseñada para aplicaciones con luz UV (de Edmund Industrial Optics 138-956). Esta fibra presenta fluorescencia a 245nm aproximadamente y no afecta en los espectros obtenidos por estar fuera del intervalo de análisis. Tiene una cubierta de plástico que provee una máxima eficiencia en la transmisión debido a la razón núcleo-revestimiento que es 10:1, la funda es de PVC y está terminada con puntas de acero inoxidable. Su longitud total es de 36" (92.5cm) y tiene un diámetro de 0.250" (6.0mm).

Espectrómetro (Acton Research®)

Un espectrómetro es un sistema con un elemento dispersivo (rejilla de difracción) que tiene la función de separar angularmente las distintas longitudes de onda de un haz de luz policroma. Se compone de una ranura de entrada y una ranura de salida que permiten establecer la resolución del sistema.

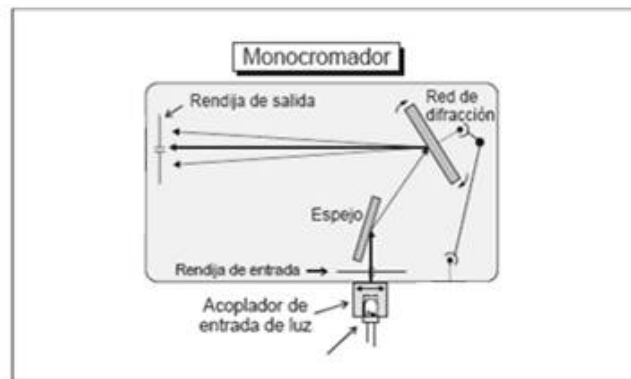


Fig. 7 Representación esquemática de un espectrómetro.

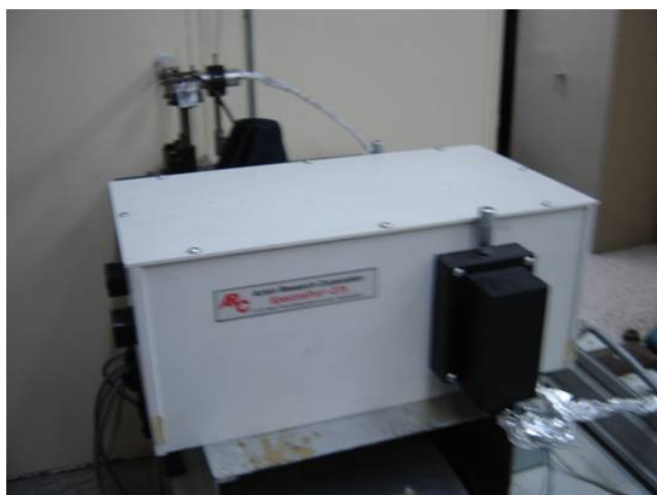


Fig. 8 Espectrómetro.

Para cambiar la longitud de onda de medición, o para hacer un barrido espectral, se mueve el elemento dispersor o algún espejo por medio de un motor por pasos.

Los componentes de un espectrómetro son:

Rendija de entrada (D1 de anchura d_1): Sirve como una fuente para el espectrómetro. Es un rectángulo largo y estrecho que proporciona una mayor resolución de la longitud de onda de la energía dispersada, y su longitud permite tener suficiente energía en el sistema como para producir una señal medible

Rendija de salida (D2 con anchura d_2): Determina la banda de paso en términos de la longitud de onda de la radiación que alcanza el detector de la muestra.

Colimador espejo o lente (M1): Permite dirigir los rayos que emanan de la rendija de entrada de

forma que los rayos provenientes de todas las posiciones de la rendija llegan al elemento de dispersión con el mismo ángulo. El segundo colimador forma una imagen de la rendija de entrada, después de que ha sido dispersada como función de la longitud de onda, en la rendija de salida.

Elemento de dispersión (rejilla de difracción): Es la parte más importante del espectrómetro. En nuestros espectrómetros se utiliza la rejilla como elemento de dispersión de la luz y se selecciona la longitud de onda deseada a través de una rendija estrecha.

El espectrómetro utilizado es de la compañía Acton Research® modelo Spectra Pro-275 de 0.275m de longitud focal. Tiene 3 rejillas distintas que permiten seleccionar la resolución del barrido y es posible controlarlo remotamente por medio del puerto serie de una PC (RS-232). Debido a que la fluorescencia inducida por láser se encuentra dentro del espectro visible, se utiliza la rejilla 1, que es capaz de realizar un barrido desde 100 a 750nm.

Tubo Fotomultiplicador (Hamamatsu)

En la salida del espectrómetro se pueden tener niveles de luz extremadamente pequeños por lo que se usó un tubo fotomultiplicador modelo Hamamatsu® montado en un arreglo que proporciona niveles de tensión equivalentes a la cantidad de luz en la salida del espectrómetro.

Tabla 1.- Principales características del fotomultiplicador

Ancho espectral	200 a 900nm
Tiempo de respuesta	2ns
Sensitividad	4x100 A/W a 400nm
Corriente de oscuridad	3nA
Tensión de alimentación	1000V máx.

El tubo fotomultiplicador se encierra en un compartimiento que impide que llegue luz no deseada y es alimentado por una fuente regulada de 0-15V. Es importante aclarar que el tubo fotomultiplicador utilizado tiene una curva de

respuesta no lineal, es decir la forma en que convierte la radiación incidente en voltaje, depende de la longitud de onda de dicha radiación. El rango en el cual trabaja este dispositivo es entre los 160nm y los 950nm. La respuesta del tubo R955 de Hamamatsu se presenta en la figura 9.

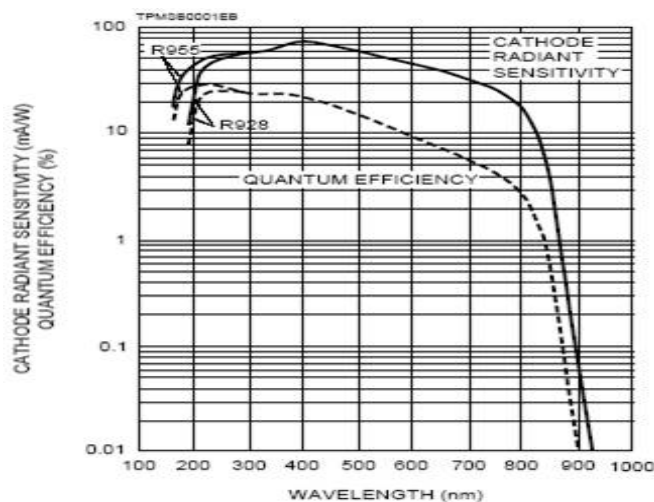


Fig. 9. Respuesta típica espectral del tubo fotomultiplicador R955.

Fotodetector para muestreo de la radiación láser (Motorola)

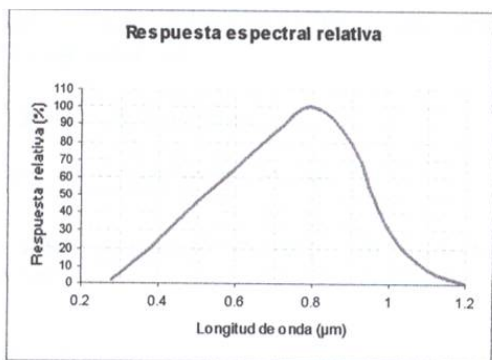
El fotodetector para muestreo de la radiación láser es un MRD500 de Motorola® y es utilizado para medir la intensidad de la radiación que proviene del láser y sincronizar el osciloscopio de tal manera que asegure la captura de la señal generada por el fotomultiplicador. El uso de este dispositivo es de gran importancia ya que la señal fluorescente obtenida depende de la muestra que se está analizando y, de acuerdo a la naturaleza de dicha muestra se pueden obtener niveles tan pequeños de señal que no sean capaces de disparar el osciloscopio. El fotodiodo recibe una intensidad de luz láser casi constante proveniente del divisor de haz lo suficientemente grande como para que sea captada por la electrónica de disparo del osciloscopio.

Las características principales del fotodiodo utilizado se muestran en la Tabla 2.

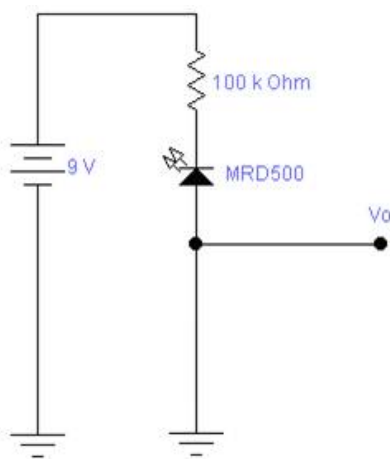
Tabla 2. Principales características del fotodiodo MRD500

Ancho espectral	300 a 1150nm
Tiempo de respuesta	1nS
Sensitividad	1.2– 6.6 μ A/mW/cm ² a 800nm
Área de detección	0.1225 cm ²

La gráfica mostrada en la figura 10 muestra la curva de respuesta espectral del fotodiodo. Esta curva es útil para conocer la respuesta real de éste en función de la longitud de onda de la luz incidente. Para este caso, la longitud de onda de la luz UV del láser de nitrógeno es 337.1nm y su tensión de salida será el 10% de la tensión obtenida si se excitara con luz de una longitud de onda de 800nm (respuesta máxima).

**Fig. 10 Respuesta espectral relativa del fotodiodo MRD500**

El circuito que se muestra en la figura 11 se utiliza para polarizar al fotodiodo empleado

**Figura 11. Circuito eléctrico del detector**

Adquisición de Radiación Láser y Fluorescencia (Majumder S., Kumar et al. 2004)

Las señales de la radiación láser y la fluorescencia proporcionadas por los fotodetectores son registradas en un osciloscopio digital Tektronix®2440.

Algunos osciloscopios digitales permiten guardar las señales registradas en un disco y posteriormente observar estas imágenes en una PC, aunque esta es una gran ventaja, la grabación de señales en disco es lenta y si se deseara realizar alguna operación matemática con dos o más señales esto es aún más lento y tedioso.

El osciloscopio utilizado no tiene la capacidad de registrar estas señales en un disco, sin embargo cuenta con una interfaz de comunicación GPIB con el estándar IEEE 488 que le permitirá comunicarse con una PC para transferir todas las señales que se requieran en un tiempo considerablemente corto.

Interfaz GPIB bajo la norma IEEE 488 (GPIB-USB Controller 4.2)

Utilizando una interfaz estándar, se pueden diseñar instrumentos que tengan un nivel básico de compatibilidad con otros instrumentos que cumplan con la norma IEEE488. La norma IEEE 488 define tres características de la interfaz de un instrumento:

- Mecánica. El conector y el cable
- Eléctrica. Los niveles eléctricos para las señales lógicas y como las señales con enviadas y recibidas
- Funcional. Las tareas que la interfaz de un instrumento puede realizar tales como, enviar datos o recibirlos.

El conector GPIB tiene 24 pines, 16 asignados a señales específicas y 8 para tierras. Los instrumentos pueden conectarse en una

configuración lineal, de estrella o una combinación de ambas. La forma lineal es donde el cable GPIB está conectado de un instrumento al siguiente. En la configuración de estrella todos los cables de los instrumentos están conectados a un punto en común.

Para mantener las características eléctricas de potencia en el bus, un dispositivo debe estar conectado mediante un cable de 2 m de longitud como máximo y al menos dos terceras partes de los instrumentos conectados al bus deben estar encendidos.

Las tensiones requeridas en todos los conectores del bus se basan en la tecnología TTL y utilizan lógica negativa.

Un "0" lógico corresponde a tensiones $> 2V$ y $< 5.2V$ Un "1" lógico corresponde a tensiones $> 0V$ y $< 0.8V$.

Cada instrumento conectado al bus tiene sólo una dirección primaria. Se puede asignar una dirección primaria a cada instrumento en el rango de 0 a 31.

Con esto se observa que un solo bus es capaz de controlar hasta 32 instrumentos.

Tarjeta GPIB

Para poder establecer la comunicación entre la PC y el osciloscopio por medio de la interfaz GPIB es necesario contar con una tarjeta que nos permita realizar este proceso de control.

Se utilizó una tarjeta GPIB de *National Instruments®* AT -GPIB/TNT con el fin de manejar los instrumentos por medio de una PC.

Algunas características de esta tarjeta son las siguientes:

- Convertidor analógico–digital con resolución de 12 bits del tipo de aproximaciones sucesivas y 24 líneas de entrada/salida compatibles con tecnología TTL y tres contadores / timers de 16 bits.

- Frecuencia de muestreo de ADC $F_m=20MHz$.
- 8 canales de entrada analógicas configurable (como 4 entradas diferenciales u 8 entradas simples)
- Transferencia máxima de datos de 1.5 Mbytes/s, normalmente 1Mbyte
- Compatible con sistemas Windows 95/98/ME
- Control simultaneo de hasta 32 instrumentos.
- Conexión a la PC en las ranuras de expansión ISA

Muestras de tejido

Las muestras de tejido cervical después de ser tomada del paciente fueron puestos en formaldehido al 10% durante 24 horas en un cuarto a temperatura constante, posteriormente son incrustados en parafina para su corte. Secciones de 4 μm fueron cortadas para ser analizadas empleando un microscopio. Se tiñeron las muestras con hematoxilina y eosina. En este estudio se utilizaron 15 muestras de 15 diferentes pacientes. El diagnóstico histopatológico fue tomado como el diagnostico correcto con el cual posteriormente se comparara el resultado entregado por el sistema LIF. 10 de las muestras fueron empleadas para entrenar la red neuronal, la cual posteriormente se empleo para clasificar las restantes 5 muestras de tejido y así validar la red neuronal. Todas las muestras de tejido tenían una antigüedad promedio de 3.5 años.

III. MANEJO DEL SISTEMA

Una vez construido el sistema, los pasos que se siguen para obtener las mediciones de fluorescencia de las muestras tisulares fijadas en el vidrio porta y cubre objetos, son los que se describen a continuación:

- Mediante una bomba de vacío se genera el vacío adecuado en la cámara de descarga

del láser de N₂, abrir la llave del tanque de Nitrógeno, verificar que la presión sea la adecuada (60mbar) y aplicar alto voltaje hasta observar la descarga del láser.

- Verificar que los elementos ópticos estén perfectamente alineados con respecto al haz láser, es decir que al pasar por el divisor de haz efectivamente una parte del láser se desvíe hacia el fotodiodo (referencia), para verificar dicho efecto es recomendable conectar la salida del fotodiodo a un canal del osciloscopio y observar la señal desplegada en la pantalla del instrumento, lo que nos asegura el correcto funcionamiento del láser (≈3%).
- Verificar que la otra parte del haz láser (≈81.2%) llega al sistema óptico e incide en la parte central del punto donde quedará la muestra, de ser así se apaga el láser y se coloca la muestra bajo medición y se fija.
- En la PC en la barra de INICIO seleccione y elegir LabView.
- Una vez en LabView pulse la pestaña ABRIR y elegir el programa que tiene la siguiente ruta:
C:\WINDOWS\Escritorio\flu\Tek2440\Graphed12a.vi, y se elige la aplicación que controla el sistema.
- Posicionar el control de acción en inicializar espectrómetro, encienda el espectrómetro, al tiempo que aparece en el espectrómetro el mensaje de inicializando debe dar clic al botón de correr continuamente en la barra de ejecutar en la PC.
- Verificar que aparece el mensaje de: COMPUTER CONTROL en el espectrómetro y en la PC debe mostrarse el mismo mensaje, en caso contrario repita el proceso.

- Establecer las longitudes de onda para el barrido que realizará el espectrómetro, en este caso los controles de posición del espectrómetro se colocan de tal forma que inicien en 330nm hasta 650nm con incrementos de 5.
- Indicar una ruta para guardar el archivo de Excel que se generara al término de este proceso.
- Indicar una ruta para guardar el archivo de texto que se generara al término de este proceso.
- Posicionar el control de acción a realizar en posición inicial.
- Dar clic al botón de correr continuamente en la barra de ejecutar.
- Verificar que el espectrómetro se posicione en 330nm, si es así detenga el programa haciendo clic en el botón STOP de la barra ejecutar.
- Encender la fuente de alimentación del fotomultiplicador.
- Fijar a un voltaje de 15V ±1.
- Una vez colocada la muestra posicionar el control de acción a realizar en barrido.
- Dar clic al botón de correr continuamente en la barra de ejecutar.
- Una vez que el espectrómetro llegue a la posición final (650nm) debe dar clic en el botón Stop de la barra ejecutar.

IV. CONCLUSIONES

Utilizando el sistema de medición implementado y siguiendo la serie de pasos descritos para obtener las mediciones de fluorescencia de las muestras de

tejidos de cérvix de útero fijadas en vidrio portaobjetos, se puede logra analizar los diferentes espectros.

La Figura 12 muestra la intensidad de fluorescencia promedio contra la longitud de onda obtenida en 10 diferentes puntos de cada una las muestras. La fluorescencia de las muestras de tejido saludable es mayor que la fluorescencia de otras en las bandas de 350 a 540nm y de 575 a 650nm, y este podría ser un criterio para separar muestras malignas y no malignas (Richards - Kortum and Sevick -Muraca 1996). Para una excitación de 320nm algunos autores (Glassman, Liu et al. 1992) han propuesto usar solo la relación de la intensidad de la fluorescencia a 383 y 460nm para clasificar muestras malignas y no malignas de biopsias cervicales in-vitro obteniendo un 33% en los diagnósticos falso-negativo.

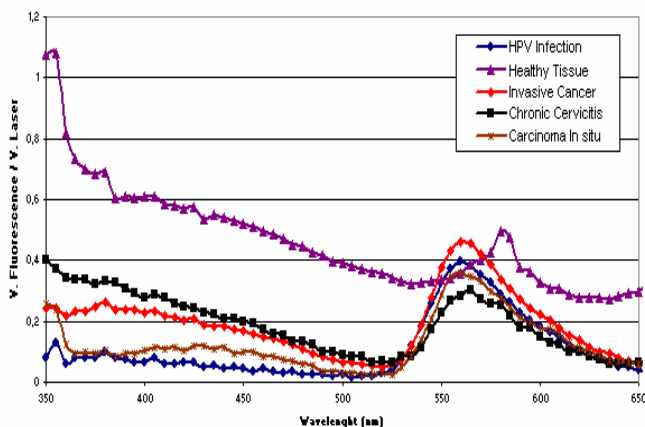


Figura 12. Espectros de fluorescencia para diferentes tejidos de cérvix

V. REFERENCIAS

Acton Research®. "Especificaciones del Monocromador y el sistema de scanner utilizado Modelo Spectra Pro-275."

Álvarez, R., E. B. Franco, et al. (2005). Sistema Auxiliar para el diagnóstico de carcinoma epidermoide invasor en tejido cérvix empleando tecnología láser. Trabajo Terminal No. 633. ESCOM, IPN. **Licenciatura**.

De la Rosa, J. and J. Bautista F. (2005). "Optical properties of paper at 337.1nm." Revista Mexicana de Física **51**(1): 110-113.

Ferlay, J., F. Bray, et al. (2004). "GLOBOCAN 2002; Cancer incidence, mortality and prevalence worldwide." IARC Cancer Base No. 5.

Glassman, W. S., C. H. Liu, et al. (1992). "Ultraviolet excited Fluorescence spectra form non-malignant tissues of the gynecological tract." Lasers in the Life Sciences **5**: 49-58.

GPIB-USB Controller 4.2
<http://prologix.googlepages.com/manual2>.

Hamamatsu. "Hojas de especificaciones del fotomultiplicador R955."

López, A. and M. Lizano (2006). Cáncer Cervicouterino y el Virus del Papiloma Humano. Cancerología. México, Unidad de Investigación Biomédica en Cáncer UNAM- INCAN. Subdirección de investigación básica. Instituto Nacional de Cancerología. **1**: 31-55.

Majumder S., K., R. Kumar, et al. (2004). "Cancer Diagnosis Using N2 laser excited autofluorescence spectroscopy of formalin-fixed human tissue." Conference Proceeding of Photonics.

Motorola. "Hojas de especificaciones del fotodiodo MRD500."

Parkin, D. M., S. L. Whelan, et al. (1997). "Cancer incidence in five continents." International Agency for Research on Cancer Scientific Publications **143**.

Richards - Kortum, R. and E. Sevick -Muraca (1996). "Quantitative Optical Spectroscopy for Tissue Diagnosis." Rev. Phys. Chem. **47**: 555 - 606.

Thomas, G. C., G. Chakrapani, et al. (1977). "Coherent radiation from a nitrogen laser." Applied Physics Letter **30**(12): 633-635.

DISEÑO Y APLICACIÓN DE UN MEDIDOR DE FRECUENCIA “FRECUENCÍMETRO” SOBRE FPGA’S

De la Torre Cantú Jorge, Sifuentes Mijares Juan.

Departamento de Eléctrica y Electrónica, Departamento de Posgrado e Investigación.
Instituto Tecnológico de la Laguna.
Boulevard Revolución y Calzada Cuauhtémoc, Torreón, Coahuila, C.P. 27000, México.
Tel. 8717051331 Ext130, correo electrónico: juansm@itl.laguna.edu.mx

RESUMEN

En este artículo se presenta el diseño y desarrollo de un medidor de frecuencia “Frecuencímetro”, con rango de medición de 0 a 9999 Hz, que fue implementado sobre un dispositivo lógico programable “FPGA”. Se presenta este trabajo como una muestra de la versatilidad de estos dispositivos tanto para aplicaciones, como en este caso de la instrumentación y diseño de equipos, como en otras áreas de la electrónica.

Palabras clave: Frecuencímetro, FPGA, VHDL.

I. INTRODUCCIÓN

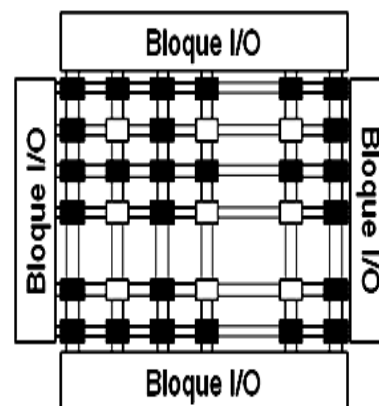
Los más recientes avances en cuanto a prototipos y diseño rápido de componentes digitales no apuntan hacia el diseño en ASIC (Application-Specific Integrated Circuit), sino al uso de un FPGA (Field Programmable Gate Array) por medio del lenguaje VHDL, Verilog, u otros.[1] Se tienen muchos avances sobre todo en diseños que necesitan un procesador embebido y componentes digitales. [2] Algunos artículos recientes hacen referencia a la paralelización de métodos y algoritmos digitales para imágenes, filtrado de ruido, receptores y transmisores inalámbricos además de otros. [3][4]

El FPGA, que es un dispositivo lógico programable de muy alta escala de integración, representa uno de los últimos avances en tecnología de dispositivos lógicos programables. [5] Es importante señalar, que cuando se diseña con un FPGA, a diferencia de lo que normalmente se conoce como sistema programado (microcontrolador, microprocesador, etc.) en donde un hardware fijo es capaz de interpretar y

ejecutar un programa especificado como un conjunto de instrucciones secuenciadas, mientras que con los FPGA’s lo que se tiene es un hardware que se re-configura realizando conexiones físicas que son especificadas por un programa o mejor dicho por una plantilla de conexiones.

Al realizar un diseño con el FPGA se presentan los mismos inconvenientes que al realizar un sistema con componentes discretos, es decir, toman relevancia los fenómenos de retardo de propagación de la señal entre los dispositivos lógicos internos y en general todo lo relacionado con la sincronización de sus señales.

Los primeros dispositivos lógicos programables que aparecieron fueron los PAL’s o los PLD’s (Dispositivos Lógicos Programables), estos presentaban compuertas (AND/OR) fijas que podían ser programadas para responder a determinadas funciones combinatoriales.



□ Bloque Lógico (LUT)
■ Llaves de Interconexión

Fig. 1 Estructura general interna de un FPGA.

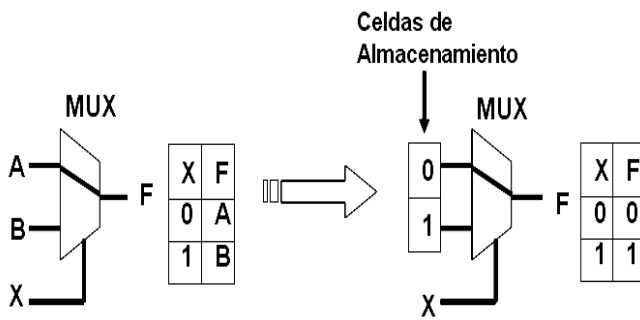


Fig. 2 Estructura de una LUT.

Los FPGA's a diferencia de los PLD's y PAL's, es que su estructura no está compuesta por compuertas AND/OR, en su lugar contienen bloques lógicos para implementar las funciones requeridas, como se muestra en Fig. 1. [5] [6]

Cada bloque lógico en un FPGA, típicamente tiene un pequeño número de entradas y salidas. Los componentes disponibles de cada bloque varían de fabricante a fabricante.

El bloque más comúnmente utilizado es una LUT (Lookup Table - Tabla de estimación), la cual contiene celdas de almacenamiento que se utilizan para implementar funciones lógicas simples. Cada celda es capaz de mantener un valor lógico, ya sea un '0' o un '1'. Este valor almacenado (1 o 0) es también recuperado a la salida de la celda de almacenamiento.

En la Fig. 2 se observa la estructura de una LUT. Los FPGA's, en general se complementan con circuitos adicionales además de las LUT's, como se observa en la Fig. 3.

La programación de estos dispositivos se puede lleva a cabo desde un nivel de diseño gráfico de un circuito esquemático hasta la escritura en un lenguaje de alto nivel como VHDL o Verilog.

II. MÉTODO

En la Fig. 4 se muestra un diagrama a bloques del frecuencímetro, sus componentes y conexiones.

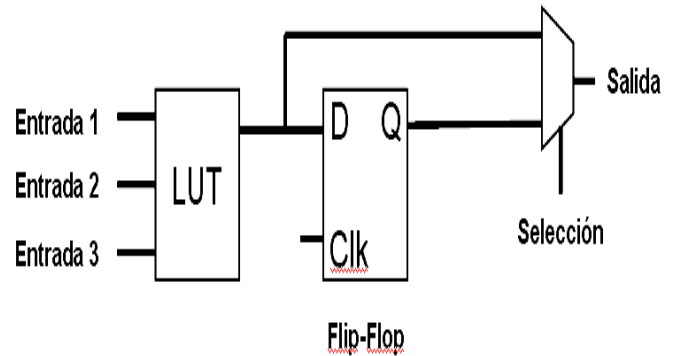


Fig. 3 Elementos internos en los FPGA's.

Para este diseño se utilizó una tarjeta SP3 de la familia Spartan de Xilinx.

El único circuito adicional al dispositivo FPGA, es el circuito acondicionador de señal. Este se realizó externamente con circuitos discretos, la razón de esto es que nuestro FPGA solo maneja señales digitales. [7]

El circuito acondicionador de señal se encarga de convertir la entrada analógica en una señal digital, dándole los niveles de voltaje adecuados, para ser procesada por la tarjeta Spartan3E.

El resto de los bloques fueron implementados con circuitos internos de la propia FPGA.

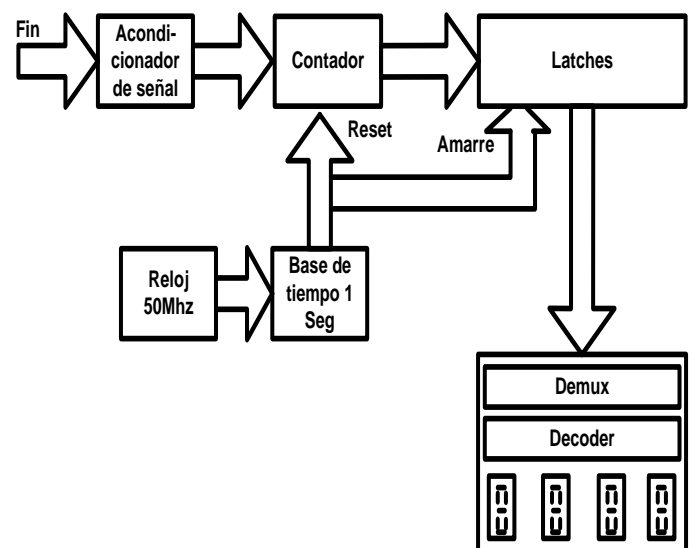


Fig. 4 Diagrama a bloques del Frecuencímetro.

Programación en VHDL

VHDL es el acrónimo de la combinación de VHSIC (Very High Speed Integrated Circuit) y HDL (Hardware Description Language). [6] [8]

El lenguaje VHDL está creado específicamente para el diseño de hardware, es decir, podremos implementar con él multitud de circuitos lógicos, tanto combinacionales como secuenciales. Éste lenguaje también nos permite describir elementos más complejos, como CPU's (Unidad Central de Procesamiento), retardadores de tiempo, etc. Además de poder manipular los archivos en VHDL ya sea reutilizando o reciclando los diseños creados previamente

Un programa básico en VHDL consta de dos partes:

La primera parte.-La entidad, nos sirve para relacionar nuestro diseño con el proceso a realizar, es decir, los recursos de datos con los que se cuenta (entradas, salidas y disposición de las mismas) para la programación.

La segunda parte.- La arquitectura, describe la circuitería interna y cómo esta procesa la información correspondiente a las entradas para obtener las salidas. [8]

```

A {
    library IEEE;
    use IEEE.STD_LOGIC_1164.ALL;
    use IEEE.STD_LOGIC_ARITH.ALL;
    use IEEE.STD_LOGIC_UNSIGNED.ALL;

B {
    entity nand2a is
        Port ( a : in std_logic;
              b : in std_logic;
              c : out std_logic
            );
    end nand2a;

C {
    architecture archi_nand2a of nand2a is
    begin
        c <= a nand b;
    end archi_nand2a;
  
```

Fig. 5 Esqueleto de programación.

En Fig. 5 aparecen las tres divisiones del esqueleto de programación: A) Librerías, B) Puertos y C) Arquitectura del programa.

El programa que se encuentra dentro de la arquitectura de Fig. 5, es simplemente una NAND, pero nos sirve como ejemplo base para mostrar una forma de codificación en VHDL.

El código VHDL del frecuencímetro

A continuación se muestran los procesos que generan los diferentes bloques en VHDL, que conforman el frecuencímetro y una breve descripción de cada uno de ellos. [9] [10] [11]

Proceso: Base de tiempo 1 segundo

Dado que las lecturas serán en Hz es necesario tener una base de tiempo de 1 segundo, la cual es obtenida a partir de la división de la frecuencia de 50 MHz, que proviene del cristal de la tarjeta, hasta dejarla en 1Hz.

Esto se hace mediante la comparación del valor de un contador de carrera libre contra el valor de 50,000,000, como se puede apreciar en el siguiente proceso.

```

--Generador de base tiempo de un segundo.
process(rst,clk)
begin
    if rst = '1' then
        b <= "00000000000000000000000000000000";
    elsif clk' event and clk='1' then
        if b < "10111110101111000010000000"
            b <= b + 1;
        else
            b <= "00000000000000000000000000000000";
        end if;
    end if;
end process;
  
```

Es necesario generar la señal que reinicia los contadores y que amarra los latches, para poder iniciar así una nueva cuenta cada segundo con el siguiente proceso:

```
--Reset de los contadores y amarre de latches.
process(b)
begin
if b = "10111110101111000001111110" then
limpiar <= '1';
else
limpiar <= '0';
end if;
end process;
```

Proceso: Contadores

La cuenta de los pulsos por segundo se realiza por medio una serie de 4 contadores en cascada: de unidades, de decenas, de centenas y de unidades de millar. Si algún contador está en 9 (en binario "1001") y recibe un pulso de reloj entonces su cuenta volverá a cero, a la vez se genera un pulso de reloj hacia el siguiente contador, donde este incrementará en uno su cuenta, hasta que finalmente llegue también a 9 y así sucesivamente con el resto de los contadores. De esta manera la cuenta no se hace en binario sino en BCD evitándose así la conversión de un sistema a otro.

Lo anterior se puede observar en los procesos que se muestran a continuación:

```
-- Primera cuenta de 0 a 9 unidades
process(rst,pulsos,limpiar)
begin
if rst = '1' then
contaA <= "0000";
elsif limpiar = '1' then
contaA <= "0000";
elsif pulsos = '1' and pulsos'event then
if contaA < "1001" then
contaA <= contaA + 1;
clk2 <= '0';
else
contaA <= "0000";
clk2 <= '1';
end if;
end if;
end process;
```

```
-- Segunda cuenta de 0 a 9 decenas
process(rst,clk2,limpiar)
begin
```

```
if rst = '1' then
contaB <= "0000";
elsif limpiar = '1' then
contaB <= "0000";
elsif clk2 = '1' and clk2'event then
if contaB < "1001" then
contaB <= contaB + 1;
clk3 <= '0';
else
contaB <= "0000";
clk3 <= '1';
end if;
end if;
end process;
```

```
-- Tercera cuenta de 0 a 9 centenas
process(rst,clk3,limpiar)
begin
if rst = '1' then
contaC <= "0000";
elsif limpiar = '1' then
contaC <= "0000";
elsif clk3 = '1' and clk3'event then
if contaC < "1001" then
contaC <= contaC + 1;
clk4 <= '0';
else
contaC <= "0000";
clk4 <= '1';
end if;
end if;
end process;
```

```
-- Cuarta cuenta de 0 a 9 unidades de millar
process(rst,clk4,limpiar)
begin
if rst = '1' then
contaD <= "0000";
elsif limpiar = '1' then
contaD <= "0000";
elsif clk4 = '1' and clk4'event then
if contaD < "1001" then
contaD <= contaD + 1;
else
contaD <= "0000";
end if;
end if;
end process;
```

Proceso: Latches

Es necesario que durante la medición presente se mantenga la información anterior, que se envía hacia los displays, y que esta se actualice hasta el momento de cumplirse el tiempo del siguiente segundo. La codificación se muestra en seguida:

```
-- Latches
process(rst,b)
begin
if rst='1' then aux_contaA <= "0001";
aux_contaB <= "0011";
aux_contaC <= "0111";
aux_contaD <= "1001";
elsif b = "10111110101111000001111100" then
aux_contaA <= contaA;
aux_contaB <= contaB;
aux_contaC <= contaC;
aux_contaD <= contaD;
else aux_contaA <= aux_contaA;
aux_contaB <= aux_contaB;
aux_contaC <= aux_contaC;
aux_contaD <= aux_contaD;
end if;
end process;
```

Proceso: Demux

La salida de cada uno de los 4 contadores (unidades, decenas, centenas y unidades de millar), deben enviarse, hacia su respectivo display una a una de la siguiente forma:

Se controla en 4 tiempos diferentes la actualización de cada display de 7 segmentos. Cada cambio en la señal div2, la cual es de dos bits ("00", "01", "10", "11"), se hace en 20.48 μ s y como necesita iniciar la cuenta a ceros (00) transcurren 4 tiempos de 20.48 μ s para que la actualización quede completa, de aquí que en 81.92 μ s se actualiza cada display de la tarjeta, teniéndose un total de 12,207 actualizaciones por segundo. Esto es tan rápido para la vista, que se tiene la idea de que la actualización de los valores en los displays es en tiempo real.

La codificación correspondiente al multiplexor quedó de la siguiente manera:

```
-- Demux
div2 <= b ( 12 downto 11);
process(div2)
begin
case (div2) is
when "00" => anodos <= "1111";
aseg <= aux_contaA;
when "01" => anodos <= "1101";
aseg <= aux_contaB;
when "10" => anodos <= "1011";
aseg <= aux_contaC;
when "11" => anodos <= "0111";
aseg <= aux_contaD;
when others => null;
end case;
end process;
```

Proceso: Decoder

Este proceso se encarga de cambiar el código de BCD a 7 segmentos, para aplicarse finalmente a los displays, de tal forma que con un "0000" se enciendan únicamente los segmentos del display que nos muestren un cero, todos encendidos menos el central, con un "0001", nos muestre el uno, solo los 2 segmentos de la derecha encendidos, y así sucesivamente hasta completar las 10 combinaciones posibles como se puede observar en el código:

```
--Decoder
process(aseg)
begin
case aseg is
when "0000"=> disp<="11000000";
when "0001"=> disp<="11111001";
when "0010"=> disp<="10100100";
when "0011"=> disp<="10110000";
when "0100"=> disp<="10011001";
when "0101"=> disp<="10010010";
when "0110"=> disp<="10000011";
when "0111"=> disp<="11111000";
when "1000"=> disp<="10000000";
when "1001"=> disp<="10011000";
when others=> disp<="11111111";
end case;
end process;
```

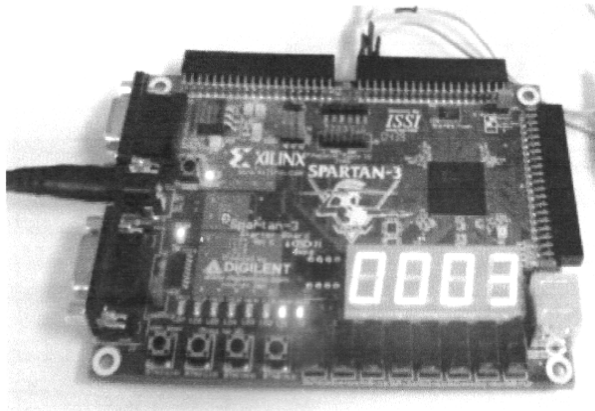


Fig. 6 Prueba del frecuencímetro.

III. RESULTADOS

Los resultados de utilización de recursos internos del FPGA, nos muestran que es muy bajo, ya que solo se emplea un máximo del 3% de utilización de los slides disponibles. Con los resultados en velocidad se puede observar que el dispositivo creado pudiera trabajar a una velocidad 5 veces superior a la empleada de 50Mhz, ver Tabla 1.

Tabla 1 Resumen de utilización y tiempos en el FPGA.

Device utilization summary:	
Selected Device : 3s200ft256-4	
Number of Slices:	65 out of 1920 3%
Number of Slice Flip Flops:	61 out of 3840 1%
Number of 4 input LUTs:	111 out of 3840 2%
Number of bonded IOBs:	13 out of 173 7%
Number of GCLKs:	2 out of 8 25%
Timing Summary:	
Speed Grade: -4	
Minimum period: 5.782ns (Max Frequency: 172.951MHz)	
Minimum input arrival time before clock: 4.993ns	
Maximum output required time after clock: 8.974ns	
Maximum combinational path delay: No path found	

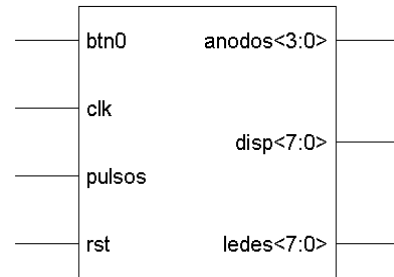


Fig. 7 Diagrama general del componente creado.

En la Fig. 6 se puede observar el diseño ya cargado en la tarjeta SP3 y en funcionamiento.

En la Fig. 7 se muestra el diagrama esquemático general del componente creado una vez sintetizado por el software de diseño para VHDL.

En la Fig. 8 se puede observar parte del diagrama interno del componente creado, en el cual, debido a su complejidad y tamaño no se incluyó en su totalidad

Cabe mencionar que, aunque el diagrama de la Fig. 8, pudiera no ser muy claro, el principal objetivo del presente artículo es mostrar la facilidad con la que se pueden realizar diseños utilizando el código VHDL, ya que finalmente es la herramienta de software la que se encarga de realizar la sintetización para entregarnos a la postre los diagramas esquemáticos del hardware diseñado.

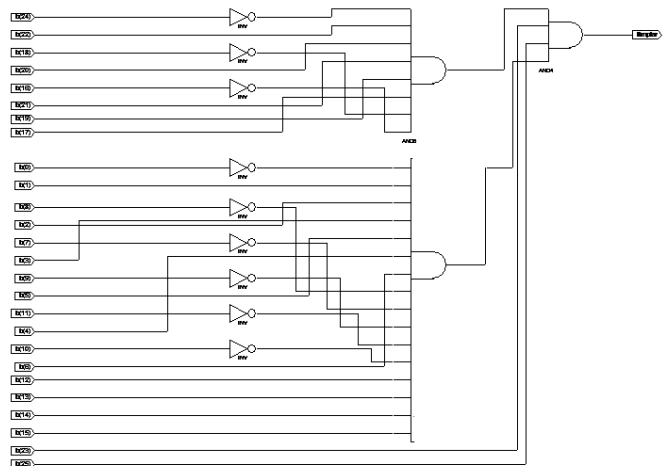


Fig.8 Diagrama esquemático de uno de los bloques implementado dentro del FPGA.

IV. CONCLUSIONES

Como se puede observar, el diseño del frecuencímetro, se implemento de una manera ágil y sencilla, además de tener mucha fiabilidad en su funcionamiento sobre el FPGA. Esto puede ser motivante para introducirse más afondo en el manejo de los FPGA's y llevarnos a su vez a la implementación de diseños más complejos y especializados donde se explote todo el gran potencial que involucra este avance tecnológico para beneficio no solo de la instrumentación electrónica sino de todos los campos en los que se involucre el uso de los FPGA's, repercutiendo finalmente en una modernización tecnológica del país.

V. REFERENCIAS

- [1] Liang Dong and Massood Z. Atashbar, “*An FPGA Experience in ASIC Design*”, Proceedings of the 2005 ASEE North Central Conference, American Society for Engineering Education, 2005.
- [2] Constantin Siriteanu et al, “*FPGA-Based Communications Receivers for Smart Antenna Array Embedded Systems*”, EURASIP Journal on Embedded Systems, 2006, Article ID 81309, Pages 1–13, 2006.
- [3] R. Cicalese et al, “*Implementation of High-Resolution Time-to-Digital Converters on two different FPGA devices*”, WSPC Proceedings, October 6, 2007.
- [4] Mihajlo Katona et al, “*A Real- Time Wavelet-Domain Video Denoising Implementation in FPGA*”, EURASIP Journal on Embedded Systems, Article ID 16035, Pages 1–12, 2006.
- [5] Xilinx, manuales de la tarjeta de desarrollo SP3, Fecha de consulta 30 Junio2008, <http://www.xilinx.com/support/documentation/index.htm>
- [6] Juan Ángel Garza Garza, “*Sistemas Digitales y Electrónica Digital*”, Prentice Hall, 2006.
- [7] Robert F. Coughlin, Frederick Driscoll, Prentice Hall, 1993.
- [8] Spartan-3 Starter Kit Board User Guide, UG130 (v1.1) May 13, 2005
- [9] M. Morris Mano, “*Logic and Computer Design Fundamentals*”, Prentice Hall, 2001,
- [10] Keshab K. Parhi, “*VLSI Digital Signal Processing Systems*”, Wile Inter Science, 1999.
- [11] John F. Wakerly “*Digital Design*”, Prentice Hall, 2000.

UN NUEVO CONCEPTO DE INTERFAZ TÁCTIL BASADO EN ESTIMULACIÓN LATERO-TÁCTIL

Martínez Rico María Lilia, Velázquez Guerrero Ramiro

Mecatrónica y Control de Sistemas - Universidad Panamericana,
Josemaría Escrivá de Balaguer 101, Fracc. Rústicos Calpulli,
20290, Aguascalientes, Ags., México

Tel: (449) 9106200, web: www.robotica-up.org, correo electrónico: rvelazquez@ags.up.mx

RESUMEN

Este artículo presenta un concepto novedoso de interfaz táctil basado en el movimiento lateral de los pines de contacto o pines táctiles así como su evaluación preliminar en transmisión de información. A diferencia de las interfaces táctiles típicas formadas por arreglos matriciales de pines que suben y bajan, este concepto permite también desplegar información 2D mediante el movimiento lateral de una sola fila de actuadores sobre un riel. La ventaja de este concepto reside en el hecho de que con una sola fila de N actuadores se puede desplegar información que hasta ahora había sido tradicionalmente desplegada en matrices de $N \times N$, reduciendo así significativamente el número de actuadores, simplificando su electrónica de control y permitiendo el desarrollo de interfaces táctiles ligeras, compactas, altamente portátiles y a muy bajo costo para el usuario. Tanto el diseño de la interfaz latero-táctil y la implementación del prototipo así como el diseño y desempeño de los actuadores y los resultados de las pruebas de percepción serán presentados y discutidos.

Palabras clave: Interfaz latero-táctil, pin táctil, movimiento lateral, motor linear miniatura, percepción táctil.

I. INTRODUCCIÓN

Las interfaces táctiles típicas son dispositivos que consisten en arreglos matriciales de actuadores y tienen como objetivo estimular la piel, particularmente la yema de los dedos, con sensaciones de contacto.

Por décadas, estos dispositivos habían sido utilizados principalmente para la transmisión de información a los invidentes o débiles visuales, pero no es hasta ahora que su verdadero potencial se ha empezado a descubrir y explotar. Una gran variedad de nuevas aplicaciones ha surgido ahora que el tacto ha comenzado a utilizarse como una nueva modalidad de interacción con las computadoras: realidad virtual, robótica, telecirugía, tele-manipulación, técnicas de entrenamiento, rehabilitación, juegos y entretenimiento (Fig. 1), entre otras.

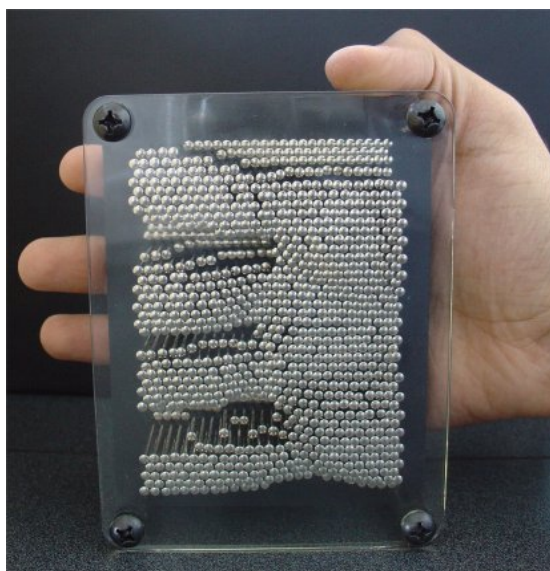


Fig. 1. Ejemplo de un dispositivo táctil de entretenimiento basado en un arreglo matricial de pines.

Una gran cantidad y variedad de dispositivos se ha desarrollado utilizando un amplio número de conceptos y tecnologías. Las tecnologías empleadas en el diseño de los actuadores van desde

servomotores, bobinas electromagnéticas, cerámicos piezoeléctricos y neumática [1-4], consideradas tecnologías tradicionales hasta el uso de las más recientes como las aleaciones de memoria de forma (SMAs: por shape memory alloys), polímeros electroactivos (EAP) y fluidos electroreológicos (ER) [5-7].

Considerables esfuerzos se han realizado buscando optimizar el concepto matricial en términos de integración, costo y desempeño pero muy pocos trabajos se han enfocado en buscar nuevas formas de estimulación táctil para lograr el mismo (o similar) efecto.

Este artículo propone un concepto novedoso de dispositivo táctil basado en el movimiento lateral de los pines táctiles. La idea consiste en una sola fila de actuadores que mueve los pines lateralmente en vez de arriba/abajo y que es capaz de desplegar información de manera matricial en 2D. La gran ventaja de este concepto es que una fila de N actuadores puede desplegar cierta información que había sido tradicionalmente desplegada en matrices de $N \times N$, reduciendo así significativamente el número de actuadores, simplificando su electrónica de control y ofreciendo características que resultan interesantes como bajo costo, peso muy ligero, dimensiones compactas y alta portabilidad.

Basado en este concepto, se ha desarrollado un primer prototipo de 8 pines táctiles. Sus dimensiones compactas ($12 \times 9.8 \times 8.3 \text{ cm}^3$) y su masa de 500 g lo convierten sin duda en un dispositivo portátil fácil de cargar por un usuario. Su costo de laboratorio es de tan solo 200 USD.

El resto del artículo está organizado de la siguiente manera: en la Sección 2 se introduce el concepto de movimiento lateral y el diseño del prototipo. En la Sección 3 se presenta el diseño de los actuadores que permite el movimiento lateral y la evaluación de su desempeño, mientras que en la Sección 4 se presenta y discute la implementación del dispositivo. En la Sección 5 se evalúa su funcionamiento y rendimiento en términos de transmisión de información a través de un conjunto de experimentos de percepción táctil aplicados a un grupo de 20 voluntarios. Por último, la Sección 6 concluye resumiendo las aportaciones principales y las perspectivas de trabajo futuro.

II. ESTIMULACIÓN LATERO-TÁCTIL: CONCEPTO Y DISEÑO

Los dispositivos táctiles que despliegan formas y figuras consisten generalmente en arreglos matriciales muy densos de pines táctiles que se mueven ortogonalmente a la superficie de la piel, [8].

Lo que buscamos conseguir es un efecto cercano o similar pero reduciendo considerablemente el número de actuadores para su implementación.

La fig. 2 muestra una representación conceptual del dispositivo táctil de movimiento lateral considerando un primer prototipo de 8 pines táctiles. Se puede observar que los pines se mueven lateralmente sobre un riel (y paralelamente a la superficie de la piel) para desplegar formas y transmitir información táctil como lo haría una matriz 2D. Los pines táctiles se mueven de manera independiente mediante un control de velocidad y posición que le permite ir a cualquier punto del riel de manera rápida y precisa.

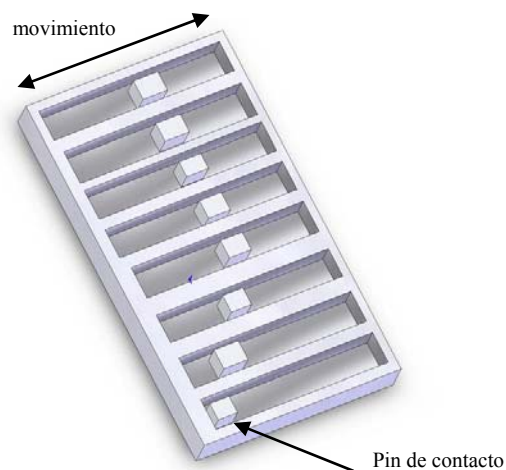


Fig. 2. Representación conceptual de la interfaz latero-táctil, según [9]

III. ACTUADOR

Se consideraron varias propuestas para implementar los actuadores y lograr el movimiento lateral decidiéndonos finalmente por una solución electromagnética, particularmente los motores lineales de CD en miniatura. Las ventajas que ofrecen son su forma compacta, portabilidad, bajo costo y facilidad para su control e implementación.

Además cuentan con varios parámetros como el ancho de banda, fuerza entregada y carrera que los hacen adecuados y atractivos para esta aplicación.

Para poder lograr el movimiento lateral de los pines táctiles se diseñó e implementó el actuador de la fig. 3.

El actuador consiste de un mini-motor lineal de CD cuyo torque y desplazamiento angular del eje se transforma en fuerza y desplazamiento traslacional a través de un mecanismo tipo tornillo sin fin. Adjunto a éste se encuentra un potenciómetro deslizante. De esta manera la posición del potenciómetro deslizante es la posición del pin táctil y es equivalente a un valor de resistencia. (Fig. 3(a)). La Fig. 3(b) muestra el prototipo del actuador. Este actuador miniatura mide 60 mm de largo por 23 mm de alto y 14 mm de ancho, su masa es de 30 g y el costo de laboratorio es de tan solo 5 USD [9].

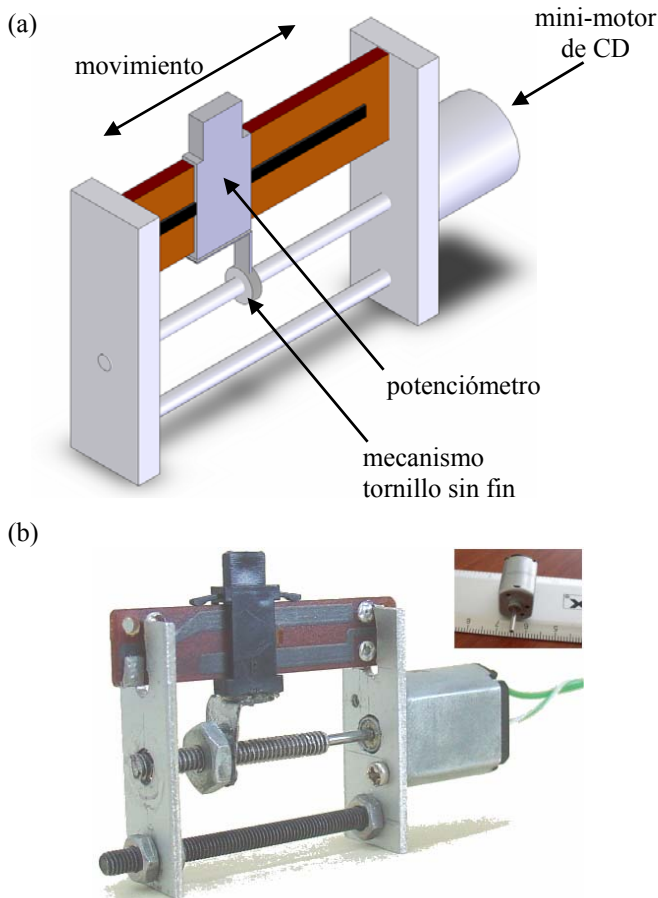


Fig. 3. Actuador electromagnético basado en motores miniatura de CD: (a) diseño y (b) prototipo.

El diagrama de bloques del actuador se muestra en la fig. 4. El actuador esta implementado con un motor de CD en el que la variable de control es el voltaje de entrada U y la salida es el torque del rotor. El torque aplicado a una carga rotativa genera una velocidad θ' . La velocidad angular convertida a velocidad de traslación ($\theta' \rightarrow x'$) se lleva a cabo en el mecanismo tipo tornillo sin fin que puede ser definido cinemáticamente como una junta helicoidal. El movimiento traslacional resultante x es proporcional al valor de la resistencia eléctrica R que se utiliza como señal de retroalimentación.

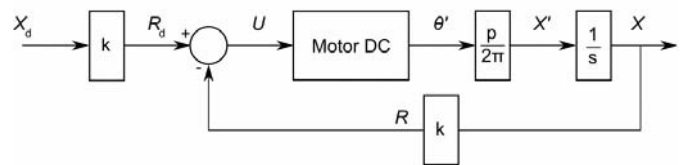


Fig. 4. Diagrama de bloques del actuador.

Seleccionando el vector de estado $X=[i \ \theta' \ x]^T$ (donde i es la corriente del motor), el comportamiento dinámico del actuador puede ser representado mediante las siguientes ecuaciones de estado:

$$X' = \begin{bmatrix} -\frac{R_m}{L} & -\frac{k_e}{L} & 0 \\ \frac{k_t}{J} & -\frac{b}{J} & 0 \\ 0 & \frac{p}{2\pi} & 0 \end{bmatrix} X + \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \end{bmatrix} U \tag{1}$$

$$y = [0 \ 0 \ 1]X$$

en donde R_m y L son la resistencia eléctrica e inductancia del motor respectivamente, k_t y k_e las constantes de torque y fuerza electromotriz, J es la masa en rotación, b el amortiguamiento del sistema mecánico y p es el paso de rosca del tornillo del mecanismo de tornillo sin fin.

De acuerdo a este modelo, la retroalimentación se toma de la resistencia eléctrica R del potenciómetro deslizante que se encuentra relacionado con la posición x del actuador mediante el parámetro k . La fig. 5 muestra la

evolución de R en función de x . Las mediciones fueron realizadas mediante un multímetro incrementando manualmente el desplazamiento del potenciómetro. Las muestras fueron tomadas cada milímetro. Se puede observar un comportamiento no lineal que puede ser linearizado de manera sencilla. Para poder modelar un comportamiento más cercano, se puede establecer una aproximación empírica válida para $1 \leq x \leq 18$ mm al integrar los datos experimentales como sigue:

$$R(x) = \begin{cases} x & \text{para } 1: 0 \leq x \leq 1 \\ 7x - 6 & \text{para } 2: 1 < x \leq 14 \\ 1.1x + 79 & \text{para } 3: 14 < x \leq 18 \end{cases} \quad (2)$$

La fig. 5 muestra los datos experimentales aproximados por la ecuación (2). Los valores para el resto de los parámetros y su forma de obtención se muestran en la tabla I.

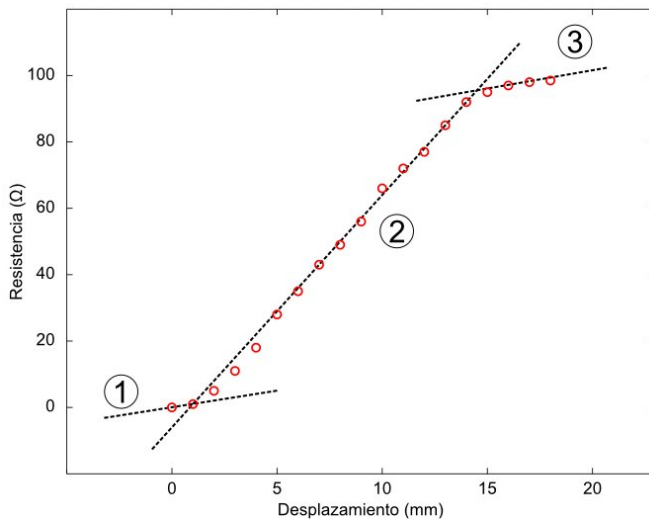


Fig. 5. Parámetro k : Evolución de R en función del desplazamiento x y datos experimentales aproximados mediante la ecuación empírica (2).

Parámetro	Valor	Unidad	Referencia
J	0.015	kg-m ²	Medición
L	0.5	H	Fabricante
R_m	1	Ω	Fabricante
k_t	0.0019	N-m/A	Fabricante
k_e	0.0041	V-s/rad	Fabricante
b	0.1	N-m-s/rad	Medición
p	1	mm	Medición

TABLA I. Resumen de los parámetros que intervienen en el modelado del actuador.

En la fig. 6 se observa la respuesta en lazo abierto del actuador a un comando en posición de 1 mm. Para poder mejorar el desempeño del actuador (precisión, velocidad de respuesta y estabilidad) se implementó un sistema de control basado en el principio de retroalimentación.

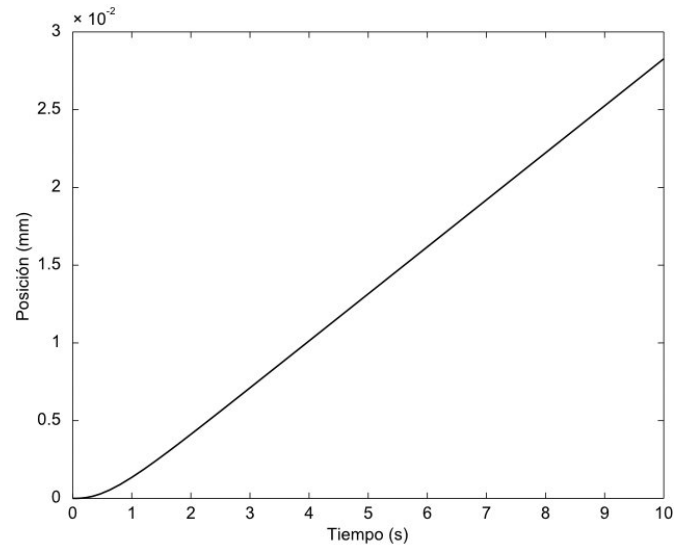


Fig. 6. Respuesta dinámica del actuador en lazo abierto.

Hasta ahora, dos opciones han sido exploradas e implementadas: control proporcional (P) y proporcional derivativo (PD). En la Fig. 7 se comparan ambas respuestas. Se observa que mientras el controlador P ofrece un tiempo de estabilización de 5 s y un error en estado estable de 0.04%, el controlador PD ofrece resultados de 1.2 s y 0.01%.

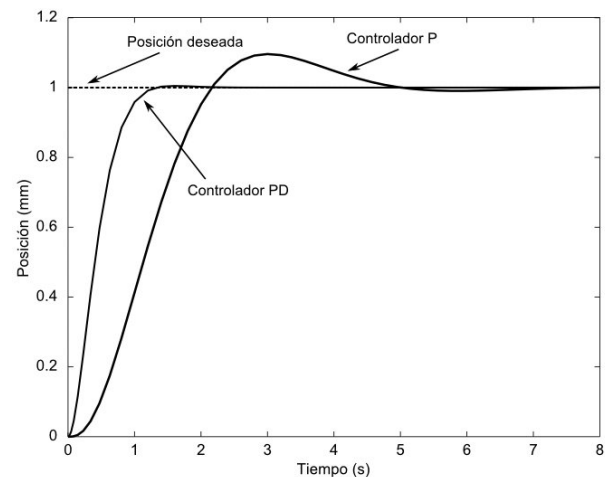


Fig. 7. Mejoramiento en la respuesta utilizando controladores P y PD.

La fig. 8 muestra la respuesta del actuador a una serie de comandos en posición utilizando el controlador PD. Note que se espera contar con un ancho de banda de 1Hz para desplegar información táctil.

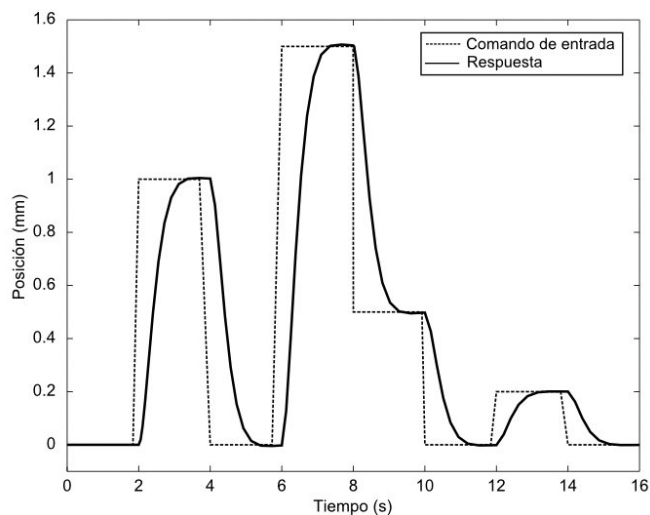


Fig. 8. Respuesta a una serie de comandos en posición compensada mediante un controlador PD.

IV. INTERFAZ TÁCTIL

El diseño que se utilizó para implementar el primer prototipo de interfaz latero-táctil se basa en un concepto monobloque que permite y facilita los procesos de ensamblaje y el de remplazar los pines táctiles en caso de ser necesario, manteniéndolos a su vez perfectamente alineados. La fig. 9(a) de lado izquierdo muestra los 8 pines táctiles integrados en la estructura monobloque comentada anteriormente.

A la derecha de la fig. 9(a) se observa el primer prototipo desarrollado. Este dispositivo consiste de 8 pines táctiles con una distancia de separación de 10 mm que permite desplazar pines de 2 mm de altura a través de un riel de 18 mm de largo con una frecuencia de operación de hasta 1 Hz. El dispositivo completo pesa 500 g y es de dimensiones compactas ($12 \times 9.8 \times 8.3 \text{ cm}^3$, control electrónico incluido como se muestra a la izquierda de la fig. 9(a)), lo que lo hace completamente portátil para el usuario. El área de contacto efectiva es de 28.5 cm^2 en un espacio de

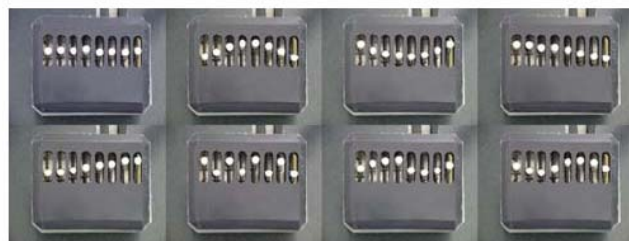
$9.5 \times 3 \text{ cm}$ pensado para ser utilizado con las 4 yemas de los dedos.

El prototipo tiene un costo de laboratorio de 200 USD. El diseño es robusto y fácil y rápido de fabricar. Otra ventaja que otorga este diseño es que permite escalar el número de pines táctiles sin dejar de ser atractivo en términos de costo.

La fig. 9(b) muestra diferentes ejemplos de formas táctiles en 2D desplegadas utilizando este dispositivo.



(a)



(b)

Fig. 9. Interfaz táctil de movimiento lateral: (a) estructura monobloque y primer prototipo. (b) Diferentes formas táctiles desplegadas en el dispositivo.

V. EVALUACIÓN PRELIMINAR EN TRANSMISIÓN DE INFORMACIÓN

Una serie de experimentos psicofísicos preliminares fueron realizados para evaluar la habilidad del prototipo para transmitir información táctil al usuario. Tres experimentos se llevaron a cabo para lograr un mejor entendimiento de las capacidades que ofrece el concepto de estimulación latero-táctil.

Veinte individuos (3 hombres y 7 mujeres, 19 diestros y 1 zurdo) de la Universidad Panamericana en Aguascalientes participaron de

manera voluntaria en los experimentos. Los individuos fueron seleccionados solo en base a su edad: se dividieron en 2 grupos de 10 personas cada uno. El primer grupo (A) estaba integrado por gente entre 17-25 años (estudiantes) mientras que el segundo grupo (B) por personas entre 40-55 años (profesores de la universidad).

Durante los experimentos, los individuos se sentaban frente al dispositivo táctil con los ojos vendados para evitar cualquier asociación visual. Antes de cada sesión, los individuos desconocían por completo los aspectos en los que consistiría la prueba y se les daban las instrucciones generales para llevarla a cabo. Ninguna familiarización con el dispositivo se les permitía antes de las pruebas. Los individuos podían explorar la superficie del dispositivo libremente sin restricciones de tiempo y sin imponer un criterio específico respecto a la posición de los dedos o la mano.

V.I. Experimento I – Reconocimiento de figuras simples

Método:

El propósito de esta primera prueba consistía en determinar si el sujeto era capaz de utilizar la interfaz táctil para identificar varias formas geométricas simples así como el nivel de comprensión de la actividad.

Se les pidió a los 20 individuos explorar la superficie táctil del dispositivo para posteriormente dibujar en una hoja de papel la forma geométrica que habían percibido. En cada prueba se tomó el tiempo de duración.

Resultados y discusión:

La fig. 10 integra los resultados obtenidos: muestra la forma desplegada en la interfaz táctil y las formas interpretadas y dibujadas por 5 integrantes de cada grupo. Los dibujos fueron seleccionados con el fin de mostrar un entendimiento excelente, bueno (bastante aceptable) y malo. Observe que en general, los individuos del grupo B obtuvieron mejores resultados que los individuos del grupo A.

Una hipótesis preliminar sugería mejores resultados para el grupo A debido a que se esperan

ciertas habilidades y destreza en gente joven y a la pérdida de sensibilidad táctil debido a la edad de los individuos del grupo B, sin embargo los resultados contradijeron esta hipótesis. Creemos firmemente que la educación y la experiencia aseguran mejores resultados: el grupo B está integrado en su mayoría por profesores acostumbrados al razonamiento espacial y a la organización geométrica, por lo que es razonable esperar mejores resultados que entre los estudiantes.

El tiempo de exploración fue evaluado utilizando un análisis de varianza (ANOVA). Los resultados para cada forma desplegada se presentan en la fig. 11(a). Los resultados muestran que el grupo B dominó la prueba no solo en rendimiento sino también en términos de rapidez.

En la fig. 11(b) y 11(c) se observan los tiempos de exploración de 6 individuos de ambos grupos A y B, respectivamente. Dos tendencias pueden ser identificadas: los individuos del grupo A mejoraron su desempeño en la exploración de la interfaz conforme avanzaba la prueba, mientras que los individuos del grupo B presentan un tiempo de exploración más uniforme durante todo el proceso. Estos resultados sugieren una rápida habituación y aprendizaje del dispositivo en gente joven y, conforme al uso y práctica, se pueden esperar mejores o similares resultados que los obtenidos por los profesores.

V.II. Experimento II – Información dinámica

Método:

El propósito de esta prueba consistió en determinar si el individuo era capaz de reconocer la dirección de movimiento cuando se despliega información dinámica.

Las tres secuencias mostradas en la fig. 12 fueron desplegadas a ambos grupos. Se les pidió a los participantes que exploraran la interfaz táctil y expresaran verbalmente la dirección del movimiento que percibieron. Todos los individuos escogieron explorar de manera pasiva el dispositivo, es decir, mantenían los dedos estáticos sobre la superficie de la interfaz.

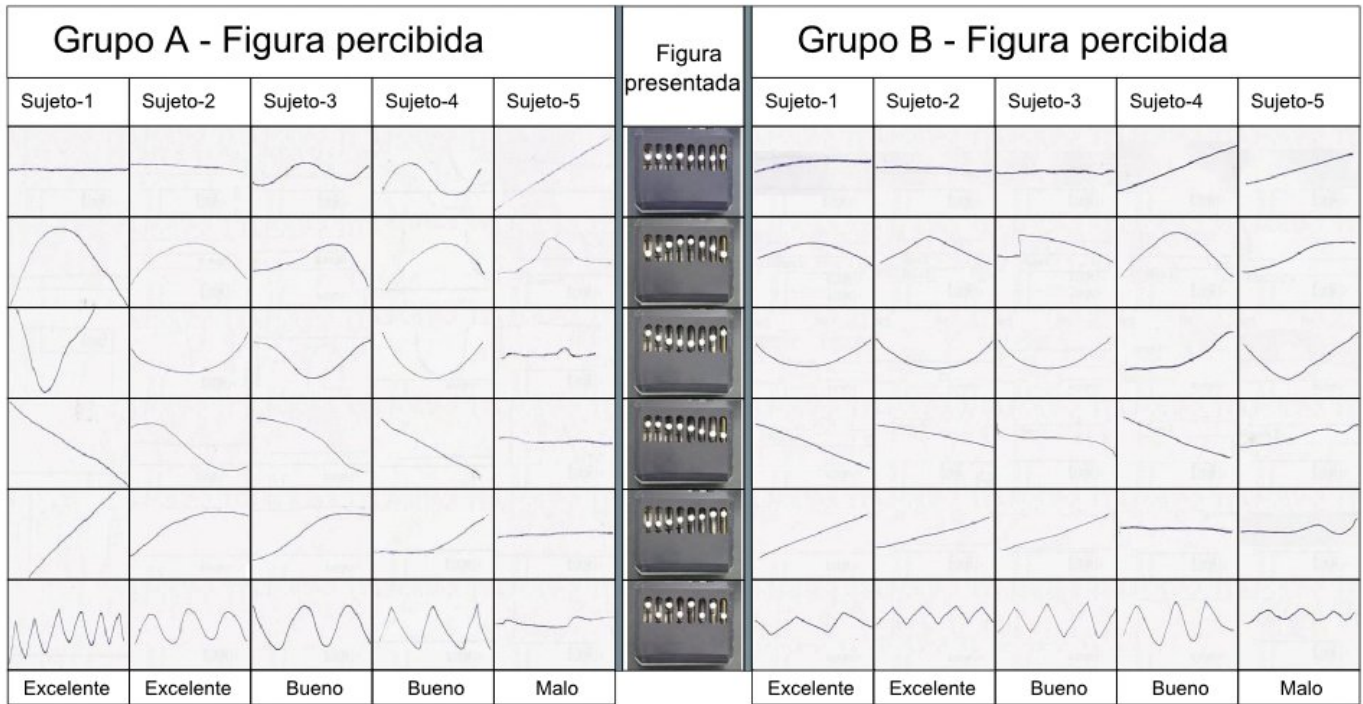


Fig. 10. Resultados representativos de ambos grupos en la prueba de reconocimiento de formas táctiles 2D.

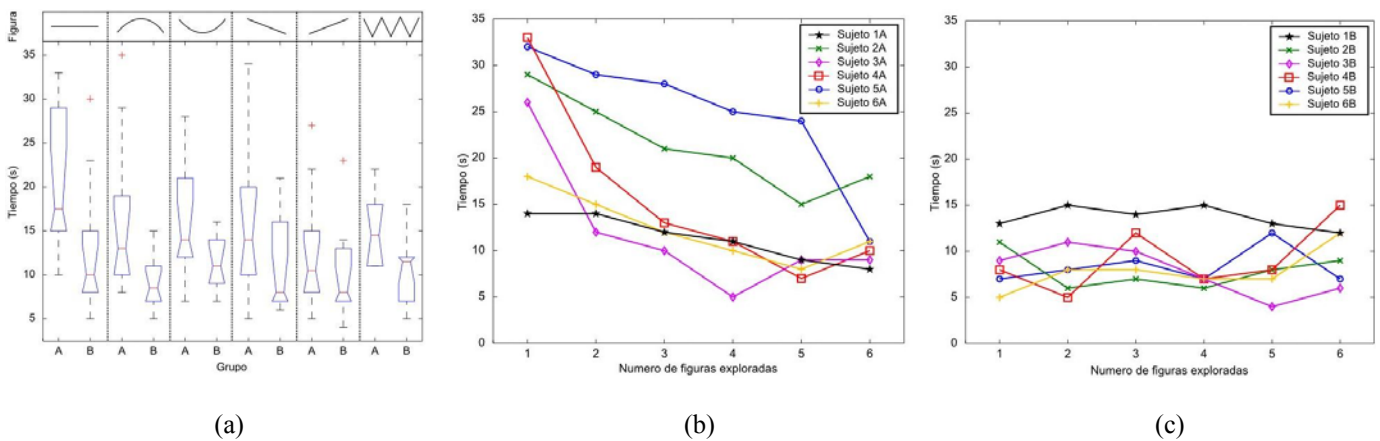


Fig. 11. (a) Resultados ANOVA del tiempo de exploración por figura: el grupo B es definitivamente más rápido que el grupo A. Tendencias en el tiempo de exploración: (b) grupo A y (c) grupo B.

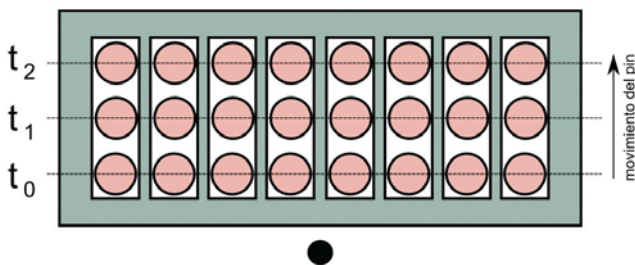


Fig. 12. Configuración dinámica utilizada en la prueba de reconocimiento de dirección.

Resultados y discusión:

Los resultados obtenidos fueron muy satisfactorios: los 20 individuos pudieron percibir la línea formada por los pines táctiles alejándose de ellos (i.e. una tasa de reconocimiento del 100%).

V.III. Experimento III – Relación visión –tacto

Método:

Las cuatro formas geométricas que se muestran en la fig. 13 fueron utilizadas en el tercer experimento. A ambos grupos se les pidió explorar la interfaz táctil mientras observaban una hoja de papel con la fig. 13 impresa (no fue permitido ver el dispositivo en ningún momento). La prueba consistía en relacionar lo que percibían de manera táctil con las figuras en la hoja. Las fig. 13(a), (b) y (c) fueron desplegadas en la interfaz mientras la fig. 13(d) fue utilizada únicamente para descartar la eliminación. Los participantes tenían solo una oportunidad para escoger la figura correcta en cada prueba y no se les permitía corregir su respuesta.

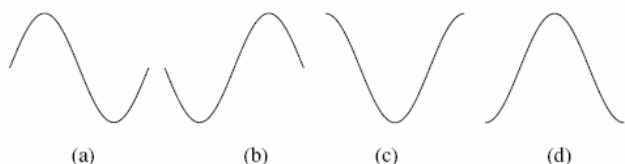


Fig. 13. Formas desplegadas en el experimento III.

Resultados y discusión:

La tabla II integra los resultados obtenidos. Se puede observar que en general se obtuvo un buen rendimiento en la prueba de relacionar visión y tacto: 63.33% para el grupo A y 66.66% para el grupo B. Nuevamente el grupo B obtuvo mejores resultados, aunque por muy poca diferencia. De nuestras observaciones no pudimos concluir que un mayor tiempo de exploración conduzca a mejores resultados.

		respuestas (%)						respuestas (%)			
		A	∩	∪	∩	B	∩	∪	∩	∩	
presentado	∩	60	10	20	10	presentado	∩	70	0	30	0
	∪	10	60	30	0		∪	40	60	0	0
	∩	20	10	70	0		∩	0	20	70	10

TABLA II. Distribución de respuestas (%) en la prueba relación visión – tacto.

VI. CONCLUSIONES

Este artículo ha introducido un nuevo concepto de interfaz táctil basado en movimiento lateral de los pines de contacto. Se utilizó una solución electromagnética para el diseño e implementación de los actuadores con el fin de realizar un dispositivo compacto, ligero, portátil y de bajo costo, que además fuera simple y rápido de fabricar.

La ventaja de este concepto sobre los dispositivos matriciales es la optimización del número de actuadores y los recursos de control que permiten desplegar cierta información. Queda bastante claro que la interfaz latero-táctil es capaz de desplegar únicamente información táctil mediante un punto por columna por lo que podría considerarse limitado en este sentido, pero creemos que es apropiado e incluso presenta ventajas importantes si se quiere desplegar contornos o límites de objetos visuales así como información tipo línea, etc.

Nuestro objetivo primordial era validar el concepto de movimiento lateral. Para este propósito, se implementó un primer prototipo de 8 pines táctiles con una resolución espacial de 10 mm y un ancho de banda de 1 Hz. Los resultados obtenidos en percepción táctil fueron bastante motivadores: la tasa de reconocimiento en todas las pruebas fue satisfactoria, lo que sugiere que la información táctil desplegada es fácil y rápida de entender. El objetivo a futuro es realizar un segundo prototipo con 2.54 mm de separación entre los pines (el estándar Braille) al igual que realizar pruebas de percepción con individuos invidentes.

VII. REFERENCIAS

[1] C. Wagner, S. Lederman and R. Howe, “Design and performance of a tactile shape display using RC servomotors”, *Proc. of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Orlando, Fl, USA, pp 354-355, 2002.

- [2] H. Fischer, B. Neisius and R. Trapp, "Tactile feedback for endoscopic surgery", *Chapter 19 in Interactive Technology and the New Paradigm for Healthcare*, IOS Press, pp 114-117, 1995.
- [3] I. Summers and C. Chanter, "A broadband tactile array on the fingertip", *Journal of the Acoustical Society of America*, vol. 112, pp 2118-2126, 2002.
- [4] F. Vidal, M. Madueño and R. Navas, "Thermopneumatic actuator for tactile displays and smart actuation circuitry", *Proc. of SPIE International Symposium on Microtechnologies for The New Millenium*, Sevilla, Spain, pp 484-492, 2005.
- [5] R. Velazquez, E. Pissaloux, M. Hafez and J. Szewczyk, "Towards low-cost highly-portable tactile displays with shape memory alloys", *Applied Bionics and Biomechanics*, vol. 4, no. 2, pp 57-70, 2007.
- [6] M. Konyo, S. Tadokoro and T. Takamori, "Artificial tactile feel display using soft gel actuators", *Proc. of IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, pp 3416-3421, 2000.
- [7] P. Taylor, D. Pollet, A. Hosseini and C. Varley, "Advances in an electrorheological fluid based tactile array," *Displays*, vol. 18, pp 135-141, 1998.
- [8] K. Shimoga, "Survey of perceptual feedback issues in dexterous telemanipulation: Part 2. Finger touch feedback", *Proc. of IEEE Annual Virtual Reality International Symposium*, Seattle, USA, pp 271-279, 1993.
- [9] S. Gutierrez and R. Velazquez, "Lateral motion pin device for tactile communication: an approach with DC mini-motors", *Proc. of 2006 Multiconference on Electronics and Photonics*, Guanajuato, Mexico, pp 146-149, 2006.

NOTAS ACERCA DEL PROBLEMA DEL PÉNDULO INVERTIDO

Preza Hernández Enrique, Hernández Mücke Hermes,
Flores Marín Berenice, Velázquez Guerrero Ramiro

Mecatrónica y Control de Sistemas - Universidad Panamericana,
Josemaría Escrivá de Balaguer 101, Fracc. Rústicos Calpulli,
20290, Aguascalientes, Ags., México

Tel: (449) 9106200, web: www.robotica-up.org, correo electrónico: rvelazquez@ags.up.mx

RESUMEN

Este artículo presenta una revisión completa del problema del péndulo invertido con el fin de ser empleado como un tutorial o material de apoyo en la comprensión de este sistema. Para ello, se presenta el diseño, modelado y control de un péndulo invertido así como la implementación mecatrónica de un prototipo. Se proponen varias estrategias de control lineal, en particular controladores PD y PID para estabilizar el sistema y ambos desempeños teórico y experimental se comparan para evaluar la precisión del modelo.

Palabras clave: Péndulo invertido, modelado, métodos de control, estabilidad.

I. INTRODUCCIÓN

Muchas universidades e institutos alrededor del mundo poseen un péndulo invertido para probar teorías de control. La razón por la cual este problema es interesante es porque, aparte de que ilustra muchas de las dificultades asociadas con problemas de control del mundo real, se basa en un modelo general que tiene múltiples aplicaciones: desde el control de misiles hasta el análisis de la biomecánica de la marcha, balance y postura humana.

Los péndulos invertidos son toda una familia de mecanismos: péndulo con rueda inercial, péndulo rotatorio de Furata, péndulo invertido esférico, doble péndulo invertido, el acrobot, el pendubot, [1-6] por mencionar solo algunos. El más estudiado es el péndulo instalado sobre un vehículo o carro.

El sistema consiste en un péndulo o varilla que gira libremente por uno de sus extremos mediante una articulación situada sobre el carro. El carro se mueve sobre una trayectoria rectilínea horizontal bajo la acción de una fuerza F (Fig. 1). Inicialmente, el péndulo se sitúa manualmente en perfecta posición vertical ($\theta = 0$), se suelta y el problema de control consiste en que de forma autónoma el péndulo continúe en esa posición mediante el adecuado posicionamiento del carro.

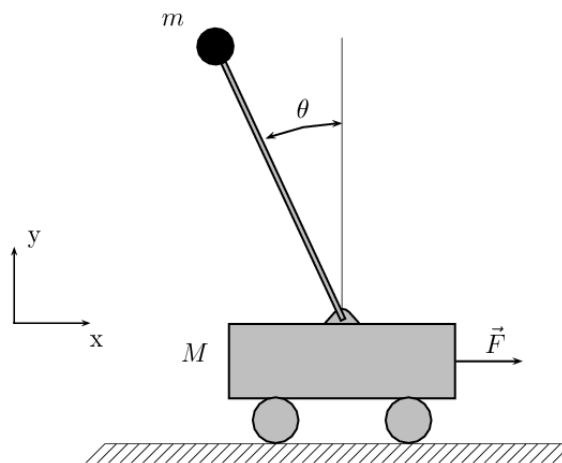


Fig.1. Péndulo en un carro.

Este artículo revisa en detalle este problema clásico de control y tiene 2 objetivos: presentar un sistema mecatrónico de péndulo invertido a bajo costo que actualmente está siendo utilizado para la enseñanza de teoría de control automático y presentar el análisis completo de este sistema de manera didáctica de manera que pueda ser fácilmente comprendido y tomado como base para el diseño de estrategias de control más avanzadas.

El resto del artículo está organizado de la siguiente manera: en la Sección 2 se presenta el modelo matemático que describe el comportamiento dinámico del sistema de péndulo invertido. La Sección 3 presenta la implementación de un prototipo mientras que en la Sección 4 se proponen una serie de controladores lineales para mejorar el desempeño del péndulo invertido. La Sección 5 concluye el trabajo resumiendo los conceptos, aportaciones principales y perspectivas de trabajo futuro.

II. MODELO DEL COMPORTAMIENTO DINÁMICO

Considere el sistema de péndulo invertido de la fig. 1. Este sistema se puede descomponer en 2 subsistemas: carro y péndulo (fig. 2).

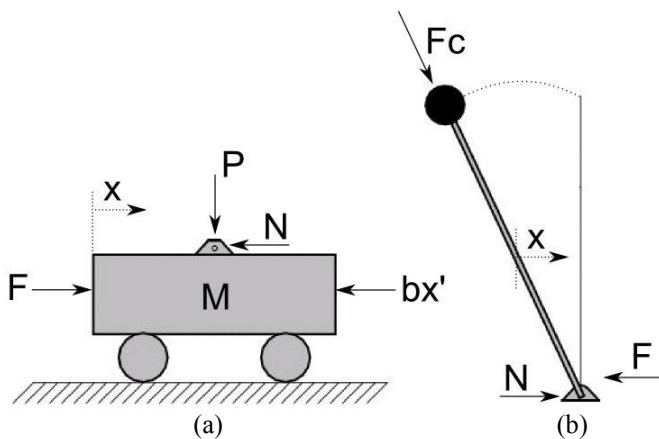


Fig.2. Subsistemas del péndulo invertido.

La dinámica del carro se puede deducir del diagrama de cuerpo libre de la fig. 2(a):

$$Mx'' = F - bx' - N \tag{1}$$

donde M es la masa del carro, F es la fuerza o impulso de entrada, b es el coeficiente de fricción y N es la fuerza horizontal que ejerce el péndulo sobre el carro (la fuerza vertical P por el momento no es significativa).

Igualmente, la dinámica del péndulo se puede deducir de la fig. 2(b):

$$mx'' = N - F_x + F_{c_x} \tag{2}$$

donde m es la masa del péndulo, F_x es la componente horizontal de la fuerza ejercida por el

momento del péndulo y F_{c_x} es la componente horizontal de la fuerza centrípeta presente al seguir el péndulo una trayectoria circular.

Recuerde que el momento del péndulo puede expresarse como el producto cruz: $T = l \times F = J\theta''$ donde l es la distancia del pivote al centro de gravedad del péndulo y J es la inercia o masa en rotación. Entonces la componente horizontal de F puede escribirse como $F_x = J\theta'' \cos\theta / l$.

De igual forma, la fuerza centrípeta se expresa tradicionalmente como $F_c = J\theta'^2 / l$. Entonces su componente horizontal F_{c_x} se puede expresar como: $F_{c_x} = J\theta'^2 \sin\theta / l$.

Integrando las nuevas expresiones para F_x y F_{c_x} y recordando que $J = ml^2$, la ecuación (2) se transforma en:

$$mx'' = N - ml\theta'' \cos\theta + ml\theta'^2 \sin\theta \tag{3}$$

Combinando (3) y (1) se obtiene la primera ecuación de movimiento:

$$(M + m)x'' + bx' + ml\theta'' \cos\theta - ml\theta'^2 \sin\theta = F \tag{4}$$

La segunda ecuación de movimiento se puede obtener sumando todas las fuerzas perpendiculares al péndulo (fig. 3):

$$P \sin\theta + N \cos\theta - mg \sin\theta = ml\theta'' + mx'' \cos\theta \tag{5}$$

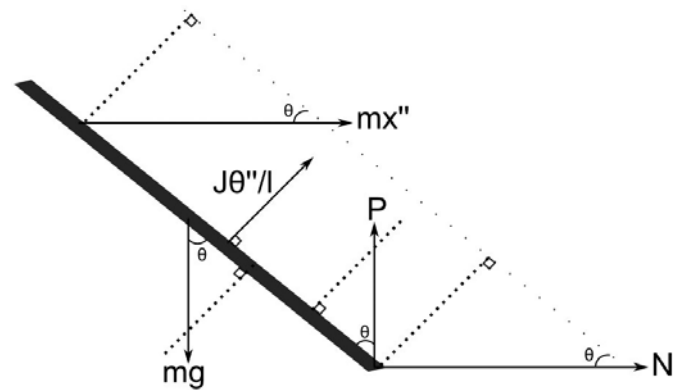


Fig.3. Fuerzas que actúan perpendicularmente al péndulo (en líneas discontinuas).

Sumando los momentos de P y N alrededor del centro de gravedad del péndulo se obtiene:

$$J\theta'' = -Pl \sin\theta - Nl \cos\theta \tag{6}$$

Combinando (5) y (6) se obtiene la segunda ecuación de movimiento:

$$(ml^2 + J)\theta'' + mgl\sin\theta + mlx''\cos\theta = 0 \quad (7)$$

Ambas ecuaciones (4) y (7) son fuertemente no-lineales y un punto de operación debe ser seleccionado para linealizarlas. Es evidente que el péndulo trabaja a π rad de su posición estable, por lo que la variable lineal es $\theta^* = \pi + \delta\theta$.

Después de linealización, las ecuaciones (4) y (7) se pueden expresar como:

$$(M + m)x'' + bx' - ml\theta^{*''} = F \quad (8)$$

$$(ml^2 + J)\theta^{*''} - mgl\theta^* - mlx'' = 0 \quad (9)$$

Seleccionado como vector de estado:

$X = [x \quad \dot{x} \quad \theta^* \quad \dot{\theta}^*]^T$, el modelo lineal del sistema de péndulo invertido puede escribirse como [7]:

$$X' = \begin{bmatrix} \frac{-b(J + ml^2)}{\Delta} & 0 & 0 & \frac{m^2 gl^2}{\Delta} \\ \frac{1}{\Delta} & 0 & 0 & \frac{\Delta}{0} \\ \frac{-mlb}{\Delta} & 0 & 0 & \frac{mgl(M + m)}{\Delta} \\ \frac{\Delta}{0} & 0 & 1 & \frac{\Delta}{0} \end{bmatrix} X + \begin{bmatrix} \frac{J + ml^2}{\Delta} \\ \frac{\Delta}{0} \\ \frac{ml}{\Delta} \\ \frac{\Delta}{0} \end{bmatrix} F$$

$$Y = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} X \quad (10)$$

$$\text{Con: } \Delta = J(M + m) + Mml^2$$

III. PROTOTIPO Y PRINCIPIO DE ACCIONAMIENTO

La fig. 4 muestra el prototipo de péndulo invertido desarrollado.

La parte mecánica consta de un péndulo de 15 cm de longitud montado sobre una base de impresora que hace el papel del carro. Este conjunto de péndulo y carro presenta un grado de libertad.

El carro se desliza por un riel ajustado a una banda dentada la cual es accionada por un motor de CD (fig 5(a) y 5(b)). La posición del carro respecto a su riel presenta el segundo grado de libertad.

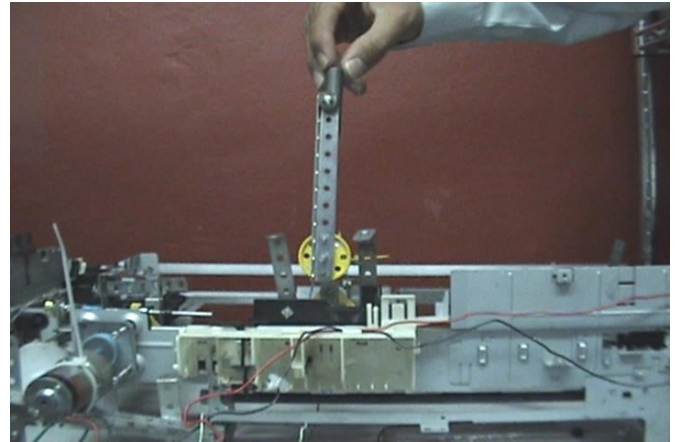


Fig.4. Prototipo de péndulo invertido.

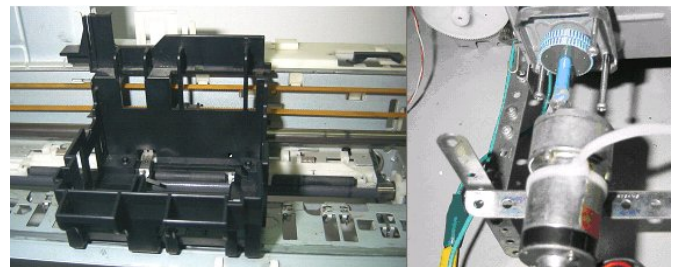


Fig.5. Subsistemas de accionamiento: (a) riel y banda dentada, (b) actuador (motor CD acoplado con reductores).

Como se puede apreciar e intuir en las figs. 4 y 5, las piezas mecánicas utilizadas son de alta disponibilidad y fácil acceso: motores, bandas y engranajes de aparatos de uso común como impresoras, video-caseteras, etc. En consecuencia, tanto el costo total como el de mantenimiento son extremadamente bajos, lo que lo hace perfecto para fines didácticos.

Para su control, el sistema cuenta con una conexión serial a la computadora. El software desarrollado es capaz de representar en tiempo real la posición del péndulo y sus principales valores y constantes características como posición, velocidad y tiempo. La fig. 6 muestra un ejemplo de la pantalla del usuario.

El software de control se comunica a la parte mecánica por medio de una interfaz electrónica que consiste de un microcontrolador (PIC) y una etapa de potencia. El microcontrolador es responsable de tratar las señales de 3 sensores (sensor de posición angular del péndulo, sensor de posición angular del motor de CD y el sensor de corriente del motor) así

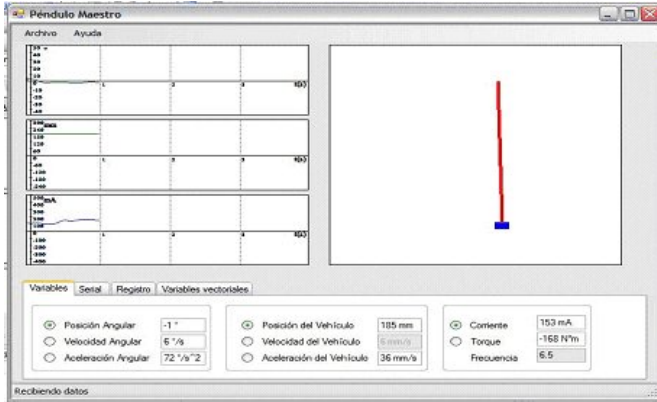


Fig.6. Ejemplo de la interfaz gráfica del usuario.

como de la transmisión de datos a la computadora.

Por otro lado, la etapa de potencia es responsable de acoplar los niveles digitales y analógicos para el funcionamiento del motor y de eliminar corrientes parásitas de la parte analógica hacia la digital.

Para completar el modelo de este sistema de péndulo invertido en particular, es necesario modelar también el mecanismo de accionamiento tipo impresora, esto es el motor de CD, las poleas y la banda sobre la cual esta montada el péndulo (fig. 7).

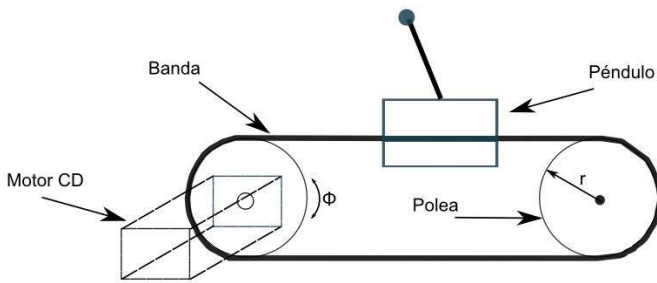


Fig.7. Sistema de accionamiento: motor-polea-banda.

La inercia o masa total en rotación J_t aplicada al eje del motor es la inercia del sistema de péndulo invertido (carro y péndulo), por lo que el torque requerido del motor es [8]:

$$\tau_m = J_t \Phi'' = (M + m)r^2 \Phi'' \quad (11)$$

donde r es el radio de la polea y Φ es el desplazamiento angular del motor.

El modelo convencional de un motor eléctrico de CD es conocido y se omitirá en este trabajo.

La fig. 8 muestra el diagrama de bloques de este sistema acoplado con el del péndulo invertido.

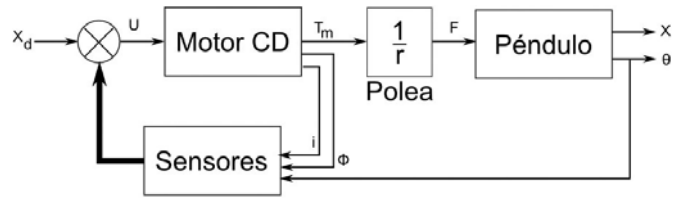


Fig.8. Diagrama de bloques del péndulo invertido accionado por un sistema motor-polea-banda.

Como se mencionó anteriormente, tres variables pueden ser utilizadas como retroalimentación: corriente del motor i , desplazamiento angular del motor Φ y desplazamiento angular del péndulo θ . La comparación de estas 3 señales con un comando de entrada de posición del carro X_d genera el voltaje U de entrada al motor.

Los valores de los parámetros involucrados en el modelado del sistema y su forma de obtención se muestran en la tabla I.

TABLA I. Resumen de los parámetros que intervienen en el modelado del sistema de péndulo invertido.

Parámetro	Valor	Unidad	Referencia
Péndulo (ecuación 10)			
M	0.7	kg	Medición
m	0.1	kg	Medición
l	0.075	m	Medición
J	5.62e-4	kg·m ²	Calculado
b	0.0041	N·m·s/rad	Medición
Motor-polea-banda			
r	0.023	m	Medición
R_m	1	Ω	Fabricante
L	0.5	H	Fabricante
k_t	0.0019	N·m/A	Fabricante
k_e	0.0041	V·s/rad	Fabricante
D	0.1	N·m·s/rad	Medición

* Constantes del motor: R_m y L son la resistencia eléctrica e inductancia de la armadura del motor respectivamente, k_t y k_e son las constantes de torque y fuerza electromotriz. D es el coeficiente de rozamiento viscoso del motor.

La fig. 9 muestra la respuesta del sistema en lazo abierto a un impulso de entrada. Note en el diagrama de polos y ceros que el sistema tiene una raíz inestable, por lo que el ángulo θ del péndulo tiende rápidamente hacia la inestabilidad.

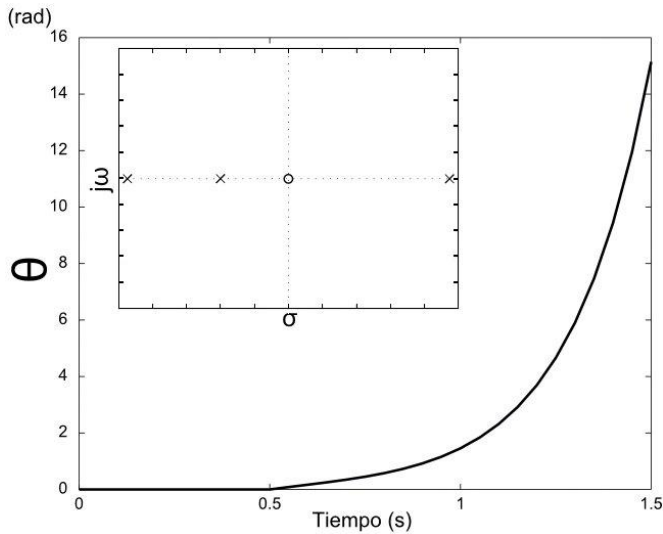


Fig.9. Diagrama de polos y ceros y respuesta del sistema a un impulso en lazo abierto.

Una forma interesante de visualizar el movimiento del péndulo es desde el punto de vista de trayectorias de movimiento para diferentes condiciones iniciales [9].

La fig. 10(a) muestra las curvas de nivel para el sistema de péndulo invertido sin tomar en cuenta ningún tipo de rozamiento. Note que el sistema es estable en π y $-\pi$ que es cuando el péndulo está paralelo al carro y altamente inestable en $\theta=0$ que es la posición deseada.

La fig. 10(b) muestra las curvas de nivel para el sistema bajo la acción de algún tipo de fricción suponiendo que el péndulo pudiera girar libremente n vueltas alrededor de su eje. Note que ahora las trayectorias tienden a π y $-\pi$ al perder el sistema energía debido al rozamiento.

Para poder estabilizar el sistema es necesario implementar una estrategia de control. La siguiente sección discute el diseño y desempeño de una serie de controladores.

IV. ESTRATEGIAS DE CONTROL

Con el fin de estabilizar el comportamiento de θ , se han explorado hasta el momento 2 opciones lineales: control proporcional derivativo (PD) y proporcional-integral-derivativo (PID). Dichos controladores tienen como objetivo modificar el lugar geométrico de raíces (LGR) mediante la

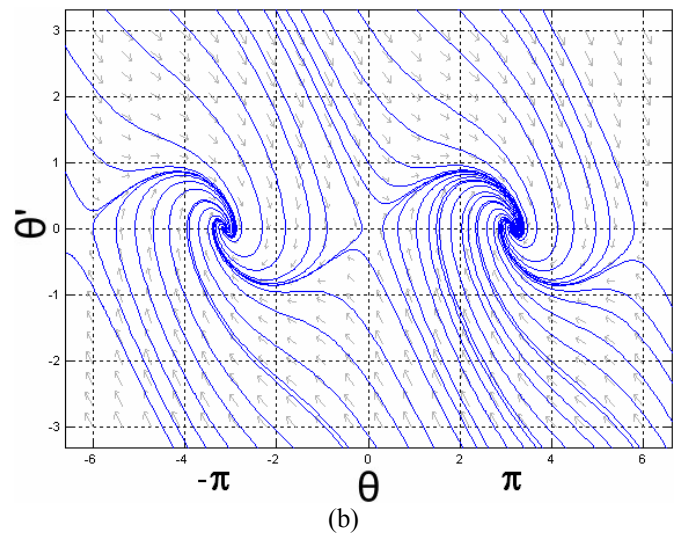
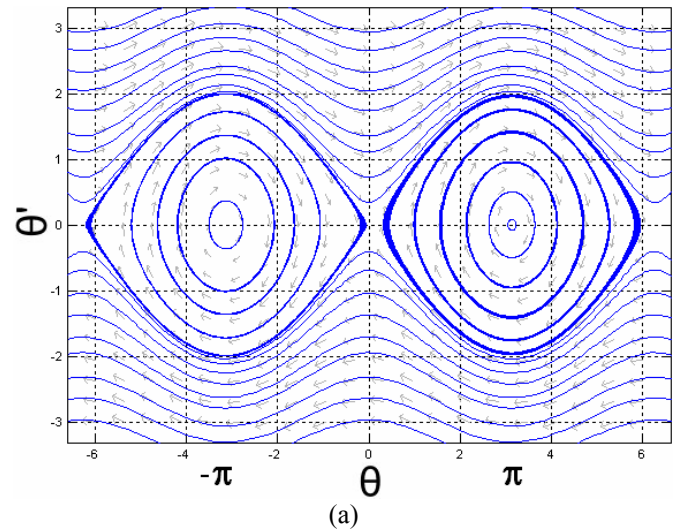


Fig.10. Curvas de nivel para el sistema de péndulo invertido: (a) sin y (b) con rozamiento.

adición de polos y ceros de tal forma que para cierto rango de valores todas las raíces del sistema sean estables [10].

Las consideraciones para la respuesta deseada del sistema de péndulo invertido son las siguientes:

- **Sobrepaso máximo del 20%.** Esta magnitud, que podría considerarse como grande, realmente no es significativa siempre y cuando el péndulo se estabilice rápidamente además de que permite mayor libertad para calibrar otros parámetros.
- **Tiempo de estabilización menor a 0.5 s** para asegurar rápidamente la posición estable del péndulo.
- **Error en estado estable igual a cero.** El péndulo debe estar en perfecta posición vertical.

La fig. 11 resume los resultados obtenidos. La fig. 11(a) muestra el LGR del sistema en lazo abierto. En esta gráfica se aprecia claramente que el sistema no puede ser controlado mediante simple retroalimentación unitaria: una rama del LGR se encuentra en la región inestable. Por ello, es necesario reordenar el LGR mediante la adición de polos y ceros.

La fig. 11(b) compara las respuestas analítica y experimental del sistema para el caso de lazo abierto. Note que, efectivamente el comportamiento es inestable, esto es θ diverge al infinito o en términos físicos θ es tan grande que provoca que el péndulo se caiga.

La fig. 11(c) presenta el LGR reordenado mediante la introducción de un cero por el controlador PD. Note la trayectoria cerrada en el LGR. Para esta trayectoria existe una serie de valores en los cuales el sistema es estable.

Tomando en cuenta las consideraciones deseadas en la respuesta del péndulo, la fig. 11(d) muestra la respuesta analítica del sistema compensado con un controlador PD a un escalón unitario. Note que ahora es posible estabilizar rápidamente θ .

De igual forma, la fig. 11(e) presenta el LGR del sistema de péndulo con la introducción de 2 ceros y 1 polo característicos al control PID. Note que el LGR es muy similar al PD de la fig. 11(c).

La fig. 11(f) presenta la respuesta analítica del sistema compensado con un controlador PID nuevamente a un escalón unitario. Note que el desempeño mejora sobretodo en magnitud del sobrepaso. Se concluye que el controlador PID es la mejor opción lineal para controlar este sistema.

Para validar el desempeño real del controlador PID, es necesario someter el péndulo a un impulso y verificar que efectivamente permanece en su estado de equilibrio $\theta=0$.

La fig. 12 compara las respuestas analítica y experimental del sistema compensado con un controlador PID a un impulso de entrada. Note que θ tiende rápidamente a su posición de equilibrio (0.5s) y que el comportamiento experimental es muy cercano al analítico. Para todos los casos se obtienen muestras de los

sensores cada 16 ms, esto es una frecuencia de muestreo de 60 Hz.

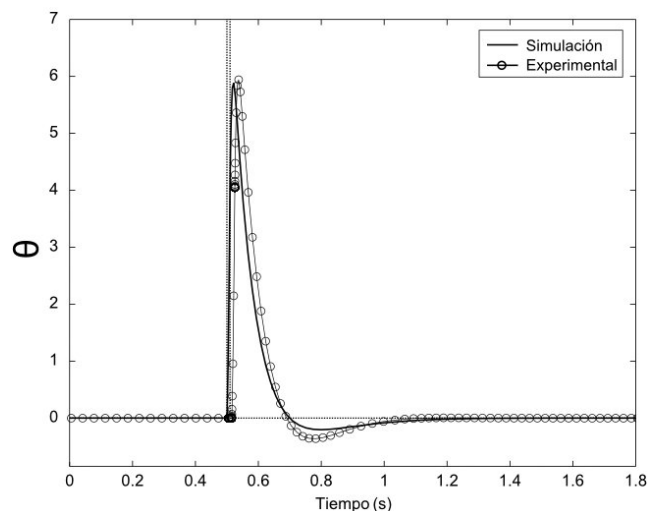


Fig.12. Comparación de las respuestas analítica/experimental a un impulso del sistema de péndulo invertido con un controlador PID. El péndulo tiende rápidamente a su posición de equilibrio ($\theta=0$).

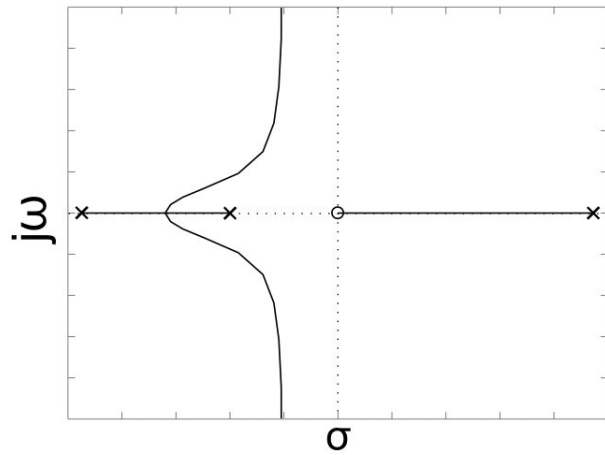
V. CONCLUSIONES

En este artículo se ha presentado una revisión completa del modelado del sistema de péndulo invertido. En particular, se analizó un prototipo accionado por un sistema motor-polea-banda el cual sirvió de plataforma para diseñar, implementar y evaluar varios controladores lineales.

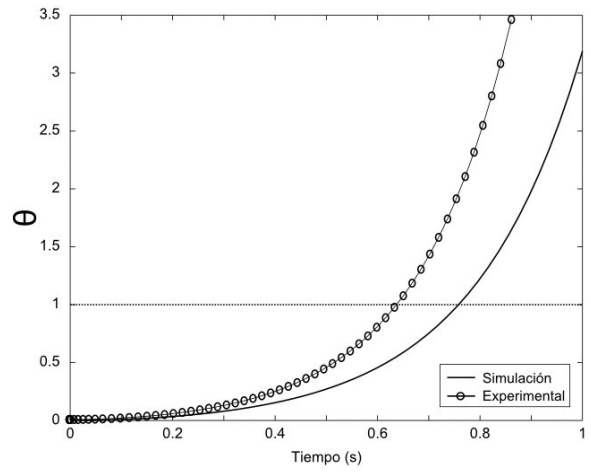
Los resultados experimentales son muy cercanos a los obtenidos analíticamente lo que demuestra que el modelo aquí formulado es confiable.

Por otra parte, se ha puesto de manifiesto el problema de la estabilidad que justifica el estudio desde el punto de vista del control no-lineal. Entre las perspectivas de trabajo futuro se incluyen el diseño de controladores basados en la teoría de Lyapunov y controladores adaptativos.

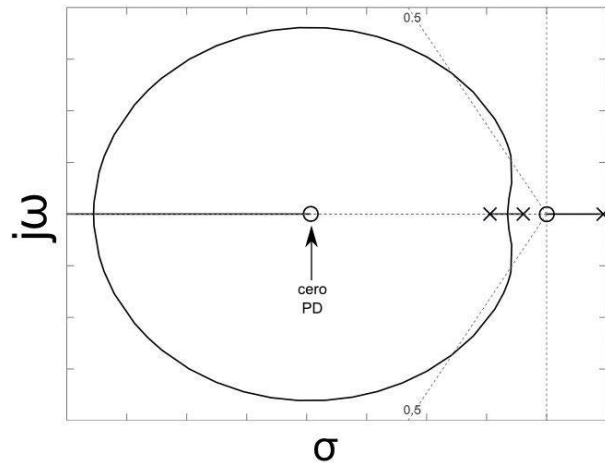
Este proyecto es una contribución al laboratorio de sistemas automáticos de la escuela de ingeniería de la Universidad Panamericana y tiene como objetivo servir de plataforma para que los alumnos puedan seguir profundizando en sus conocimientos de control al aplicarlos en un sistema real.



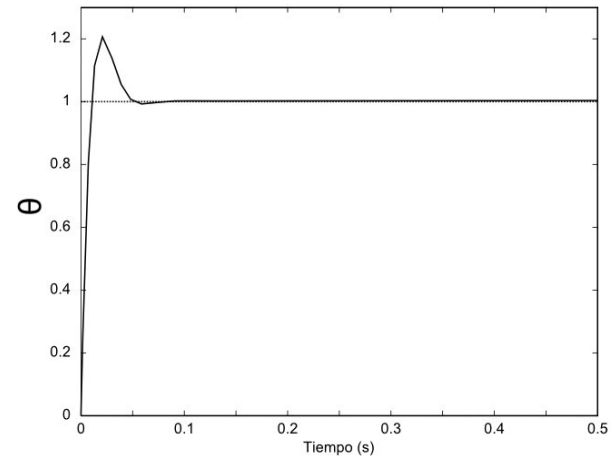
(a)



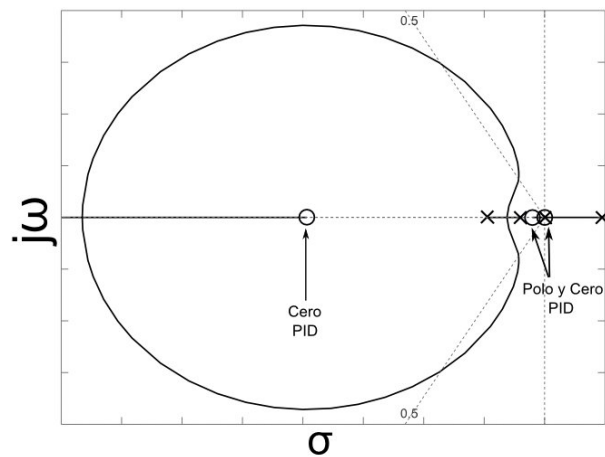
(b)



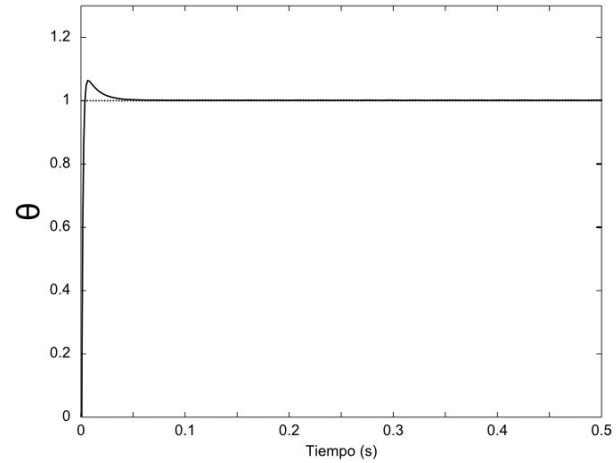
(c)



(d)



(e)



(f)

Fig.11. Lugar geométrico de raíces (LGR) y respuesta del sistema de péndulo invertido a una comando en posición: (a)-(b) lazo abierto, (c)-(d) lazo cerrado con control PD, (e)-(f) lazo cerrado con control PID.

VI. REFERENCIAS

- [1] D. Angeli, “Almost global stabilization of the inverted pendulum via continuous state feedback”, *Automatica*, Vol. 37, pp. 1103-1108, 2001.
- [2] A. Bloch, N. Leonard and J. Marsden, “Stabilization of the pendulum on a rotor arm by the method of controlled Lagrangians”. *Proc of IEEE International Conference on Robotics and Automation*, pp. 500-505, 1999.
- [3] K. Furuta, “Control of pendulum: from super mechano-system to human adaptive mechatronics”, *Proc. of IEEE Conference on Decision and Control*, pp. 1498-1507, 2003.
- [4] M. Spong, “The swing up control problem for the acrobot”, *IEEE Control Systems Magazine*, Vol. 15, No. 1, pp. 49-55, 1995.
- [5] R. Lozano, I. Fantoni and D. Block, “Stabilization of the inverted pendulum around its homoclinic orbit”, *Systems and Control Letters*, Vol. 40, No. 3, pp. 197-204, 2000.
- [6] C. Aguilar, O. Gutierrez, H. Sossa, “Lyapunov approach for the stabilization of the inverted spherical pendulum”, *Proc. of IEEE Conference on Decision and Control*, pp. 6133-6137, 2006.
- [7] Control tutorials for Matlab - Modeling an inverted pendulum:
<http://www.engin.umich.edu/group/ctm/examples/pend/invpen.html>
- [8] A. Mirza and S. Hussain, “Robust controller for nonlinear and unstable system: inverted pendulum”, *AMSE Journal of Control and Design Simulation*, Vol. 55, No. 3, pp. 49-60, 2000.
- [9] J-J. Slotine and W. Li, *Applied nonlinear control*, Ed. Prentice Hall, New Jersey, 1999.
- [10] N. Nise, *Control systems engineering*, Ed. Wiley, New York, 2000.

CONTROL EN TIEMPO REAL DE UN MANIPULADOR FLEXIBLE DE DOS GRADOS DE LIBERTAD USANDO EL MICROCONTROLADOR FREESCALE MC9S12 Y MATLAB

Rogelio Francisco Antonio¹, Jorge Cruz Mancilla²

1. División de Ingeniería Mecánica Electrónica, Universidad Politécnica del Valle de México

Mail: rfrancisco2002@yahoo.com.mx

2. División de Ingeniería Mecatrónica, Tecnológico de Estudios Superiores de Ecatepec

Mail: jcm21mafy@hotmail.com

RESUMEN

El presente trabajo muestra el control en tiempo real de un manipulador flexible de dos grados de libertad usando una interface con matlab, donde se usa un microcontrolador Freescale mc9s12. Además se presenta el diseño de la tarjeta y la construcción del prototipo donde el objetivo principal es controlar la posición del manipulador y usar el ambiente matlab simulink en tiempo real.

Palabras clave: Microcontrolador Freescale, Control, Manipulador, Interface.

I. INTRODUCCIÓN

En el presente trabajo se presentan los logros alcanzados para el diseño, simulación y control en tiempo real de un manipulador de dos grados de libertad en el entorno de Simulink utilizando un Microcontrolador Freescale mc9s12. La arquitectura de un MC9S12DP256 tiene las ventajas de un microcontrolador de 16 bits y un dsp. Las librerías que se utilizan en este trabajo son de uso libre, están disponibles en la pagina web de la Universidad de Adelaida en Australia. Las librerías contienen bloques de entrada y salida digital, periféricos de entrada/salida, manejo del puerto serial, manejo de puerto spi; la correcta configuración del dispositivo maestro para la operación del microcontrolador y el manejo adecuado de las señales entre los bloques nos lleva a una completa compilación del modelo y la creación del código hexadecimal. Cada microcontrolador 9S12, tiene la forma de operar, cantidad de instrucciones que maneja el oscilador. En este trabajo se utiliza un *MC9S12DP256*, que está integrado en la tarjeta de desarrollo DRAGON12PLUS.

El constructor de Simulink hace una comunicación a Metrowerks' para compilar el código C generado en tiempo real el cual puede ser descargado en la memoria ROM Flash del microcontrolador MC9S12DP256. Las librerías RTMC9S12 son el medio de desarrollo para el control en Simulink. El compilador que se utiliza es del programa Code Warrior de Freescale. Los avances de este trabajo consisten en el manejo adecuado de las librerías RTMC9S12, así como el desarrollo de un simple diagrama a bloques en simulink que construya un PWM para el control de los servomotores que controlan las articulaciones del manipulador de dos grados de libertad. La construcción de las articulaciones del manipulador están construidas con dos servomotores tipo S3305 de la marca Futaba. En el trabajo se presenta el diseño del control PWM en Simulink con los bloques de las librerías RTMC9S12.

El objetivo principal de este trabajo es que a partir de un diagrama de bloques en Simulink, donde se realice un simple PWM para el control de servomotores, se traduzca en código C utilizando el toolbox de Simulink del Real-Time Workshop Embedded Coder. El constructor de Simulink hace una llamada a Metrowerks' para compilar el código C generado en tiempo real el cual puede ser descargado en la memoria ROM Flash del microcontrolador MC9S12DP256

Además con este trabajo se pretende demostrar el uso de las nuevas tecnologías de la empresa Wytec que contribuyan a la construcción de nuevos prototipos rápidos en tiempo real. A continuación se describen las etapas en que consiste este trabajo:

- Construcción de un manipulador de dos grados de libertad que está compuesto en cada articulación por un servomotor S3305 de la firma FUTABA.
- Diseño de un sistema PWM para el control del manipulador en el entorno de MATLAB y SIMULINK con las librerías RTMC9S12.
- Control en tiempo real del manipulador en lazo abierto.

II. MÉTODO

La metodología que se emplea en el desarrollo de este trabajo consiste en los siguientes:

- Manejo y operación de la tarjeta de desarrollo Dragón 12 plus.
- Instalación, configuración de las librerías RTMC9S12.
- Instalación, configuración del compilador CodeWarrior de la firma Metrowerks.
- Manejo de las librerías RTMC9S12
- Creación del modelo PWM para el control de los servomotores de las articulaciones del manipulador.
- Compilación del modelo PWM.
- Programación de los archivos hexadecimales por el PUERTO SERIAL.
- Construcción del manipulador de dos grados de libertad.

Consolidación del área de trabajo y movimientos de las articulaciones del manipulador de dos grados de libertad.

III. DESCRIPCIÓN DEL SISTEMA Y MODELO MATEMÁTICO

El sistema está compuesto por tres elementos, el hardware, que realiza la interfaz entre el proceso y el computador, el software de control de las aplicaciones para cerrar el lazo de control, un diagrama esquemático puede verse en la figura 1 donde se muestran los elementos de l sistema.



Figura 1 Esquema del sistema.

El manipulador de dos grados de libertad se muestra en el siguiente esquema de la figura 2:

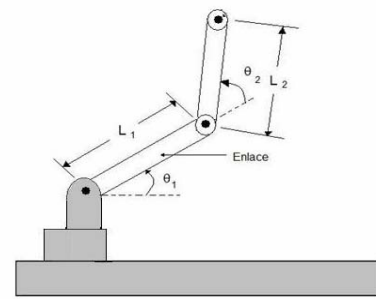


Figura 2 Manipulador de dos grados de libertad.

A. Modelo Matemático del sistema

A continuación se obtiene el modelo matemático que describe el comportamiento del manipulador. Considerando la siguiente figura, donde se describe cada una de las variables que intervienen en el sistema.

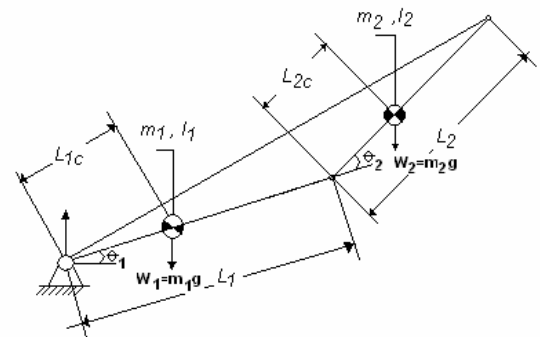


Figura 3. Estructura manipulador de dos grados de libertad

El método que se emplea para obtener el modelo matemático es el de Euler-Lagrange, entonces ahora se obtienen la energía cinética y potencial del sistema.

La energía cinética K y la energía potencial V para el primer enlace está definido por:

$$K_1 = \frac{1}{2} m_1 \dot{\theta}_1^2 + \frac{1}{2} I_1 \dot{\theta}_1^2 \tag{1}$$

$$V_1 = m_1 l_{1c} \sin \theta_1 \tag{2}$$

Para el segundo enlace, es necesario encontrar los componentes de posición y velocidad para el centro de gravedad, de forma similar las ecuaciones siguientes se pueden derivar:

$$x_{2c} = l_1 \cos \theta_1 + l_{2c} \cos (\theta_1 + \theta_2)$$

$$y_{2c} = l_1 \sin \theta_1 + l_{2c} \sin (\theta_1 + \theta_2)$$

$$\dot{x}_{2c} = -l_1 \dot{\theta}_1 \sin \theta_1 - l_{2c} (\dot{\theta}_1 + \dot{\theta}_2) \sin (\theta_1 + \theta_2)$$

$$\dot{y}_{2c} = l_1 \dot{\theta}_1 \cos \theta_1 + l_{2c} (\dot{\theta}_1 + \dot{\theta}_2) \cos (\theta_1 + \theta_2)$$

Observe que la velocidad angular del enlace L2 es $(\dot{\theta}_1 + \dot{\theta}_2)$. De aquí se observa que la energía cinética para el enlace L2 satisface:

$$K_2 = \frac{1}{2} m_2 (\dot{x}_2^2 + \dot{y}_2^2) + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \quad (3)$$

Entonces se puede obtener lo siguiente:

$$\begin{aligned} (m_2 \dot{x}_2)^2 + (m_2 \dot{y}_2)^2 = \\ I_2^2 \dot{\theta}_1^2 + I_2 \dot{\theta}_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + I_2 l_1 l_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2 \end{aligned} \quad (4)$$

Sustituyendo la ecuación (4) en la ecuación (3) obtenemos:

$$\begin{aligned} K_2 = \frac{1}{2} m_2 I_2^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \\ + m_2 l_1 l_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2 + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \\ K_2 = \frac{1}{2} m_2 I_2^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 I_2 (\dot{\theta}_1^2 + 2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \\ + m_2 l_1 l_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2 + \frac{1}{2} I_2 (\dot{\theta}_1^2 + 2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \end{aligned} \quad (5)$$

La energía potencial en el enlace L2 es:

$$V_2 = m_2 g l_1 \sin \theta_1 + m_2 g l_2 \sin (\theta_1 + \theta_2) \quad (6)$$

Ahora obtenemos el Lagrangiano que está definido por $L=k-v$, entonces para el manipulador de dos grados de libertad se tiene que:

$$L = (K_1 + K_2) - (V_1 + V_2) \quad (7)$$

Sustituyendo (1), (2), (5) y (6) en (7) obtenemos

$$\begin{aligned} L = \frac{1}{2} m_1 I_1^2 \dot{\theta}_1^2 + \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} m_2 I_2 \dot{\theta}_1^2 \\ + \frac{1}{2} m_2 I_2^2 (\dot{\theta}_1^2 + 2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) + m_2 l_1 l_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2 \\ + \frac{1}{2} I_2 (\dot{\theta}_1^2 + 2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \\ - m_1 g l_1 \sin \theta_1 - m_2 g l_1 \sin \theta_1 - m_2 g l_2 \sin (\theta_1 + \theta_2) \end{aligned}$$

Derivando el Lagrangiano L, obtenemos las ecuaciones de movimiento:

$$\begin{aligned} \frac{\partial L}{\partial \theta_1} = m_2 I_2^2 \dot{\theta}_2 + I_2 \dot{\theta}_1 + m_2 I_2 \dot{\theta}_1 + m_2 I_2 \dot{\theta}_1 \\ + m_2 I_2^2 \dot{\theta}_2 + 2m_2 l_1 l_2 \dot{\theta}_1 \cos \theta_2 \\ + m_2 l_1 l_2 \dot{\theta}_2 \cos \theta_2 + I_2 \dot{\theta}_1 + I_2 \dot{\theta}_2 \end{aligned}$$

Observe que se satisface que

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_1} \right) = [m_2 I_2^2 + I_2 + m_2 I_2^2 + m_2 I_2^2 + 2l_1 m_2 l_2 \cos \theta_2 + I_2] \dot{\theta}_1 \\ + [m_2 I_2^2 + m_2 l_1 l_2 \cos \theta_2 + I_2] \dot{\theta}_2 \\ - 2l_1 m_2 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 - l_2 m_2 l_2 \dot{\theta}_2^2 \sin \theta_2 \\ \frac{\partial L}{\partial \theta_1} = -m_2 g l_1 \cos \theta_1 - m_2 g l_2 \cos (\theta_1 + \theta_2) \\ - m_2 g l_2 \cos (\theta_1 + \theta_2) \end{aligned}$$

De la misma formase obtiene lo siguiente

$$\frac{\partial L}{\partial \theta_2} = m_2 I_2^2 \dot{\theta}_2 + m_2 I_2 \dot{\theta}_1 + m_2 l_1 l_2 \dot{\theta}_1 \cos \theta_2$$

$$\begin{aligned} + I_2 \dot{\theta}_1 + I_2 \dot{\theta}_2 \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_2} \right) = [m_2 I_2^2 + I_2 + m_2 l_1 l_2 \cos \theta_2] \dot{\theta}_1 \\ + [m_2 I_2^2 + I_2] \dot{\theta}_2 - m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 \end{aligned}$$

entonces

$$\begin{aligned} \frac{dL}{d\theta_2} = -m_2 l_1 l_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \sin \theta_2 \\ - m_2 g l_2 \cos (\theta_1 + \theta_2) \end{aligned}$$

Finalmente, las ecuaciones que describen el comportamiento del manipulador está definido por:

$$\begin{aligned} \tau_1 = \\ [m_1 I_1^2 + I_1 + m_2 I_1^2 + m_2 I_2^2 + 2l_1 m_2 l_2 \cos \theta_2 + I_2] \dot{\theta}_1 \\ + [m_2 I_2^2 + m_2 l_1 l_2 \cos \theta_2 + I_2] \dot{\theta}_2 \\ - 2l_1 m_2 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 - l_2 m_2 l_2 \dot{\theta}_2^2 \sin \theta_2 \\ + (m_2 l_1 g + m_2 l_2 g) \cos \theta_1 - m_2 g l_2 \cos (\theta_1 + \theta_2) \end{aligned}$$

$$\begin{aligned} \tau_2 = \\ [m_2 I_2^2 + I_2 + m_2 l_1 l_2 \cos \theta_2] \dot{\theta}_1 \\ + [m_2 I_2^2 + I_2] \dot{\theta}_2 + m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 \\ + m_2 l_1 l_2 \dot{\theta}_1^2 \sin \theta_2 + m_2 g l_2 \cos (\theta_1 + \theta_2) \end{aligned}$$

y en la forma matricial se puede representar como:

$$\begin{aligned} \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} = \\ \begin{bmatrix} m_1 I_1^2 + I_1 + m_2 I_1^2 + \\ m_2 I_2^2 + 2l_1 m_2 l_2 \cos \theta_2 + I_2 & m_2 I_2^2 + m_2 l_1 l_2 \cos \theta_2 + I_2 \\ m_2 I_2^2 + m_2 l_1 l_2 \cos \theta_2 + I_2 & m_2 I_2^2 + I_2 \end{bmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \\ + \begin{bmatrix} -2l_1 m_2 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 & -l_2 m_2 l_2 \dot{\theta}_2^2 \sin \theta_2 \\ m_2 l_1 l_2 \dot{\theta}_1^2 \sin \theta_2 & m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 \end{bmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \\ + \begin{bmatrix} (m_2 l_1 g + m_2 l_2 g) \cos \theta_1 - m_2 g l_2 \cos (\theta_1 + \theta_2) \\ m_2 g l_2 \cos (\theta_1 + \theta_2) \end{bmatrix} \end{aligned}$$

IV. DESARROLLO DEL TRABAJO

A continuación se describen las etapas para la implementación del prototipo en tiempo real.

A. Desarrollo del sistema de control en lazo abierto.

La construcción del código hexadecimal para el sistema de control tiene las siguientes etapas que se muestran en la figura 4.. En esta figura 4 se puede apreciar la conversión del diagrama a bloques de Simulink en código C, enseguida el constructor de Simulink llama al compilador CodeWarrior de la

firma Metrowerks y se genere el código hexadecimal que se envía en tiempo real por el puerto serie de la pc.

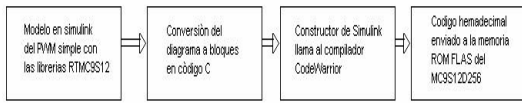


Figura 4 Sistema de control entre el modelo PWM de Simulink y la tarjeta de desarrollo Dragon12plus.

B. Proceso de generación del código hexadecimal a partir de RTW Y TCL

La generación del código hexadecimal y C. puede verse en la figura 5, esto se genero una vez que el modelo en Simulink ha sido compilado correctamente a través de la interfaz con el compilador CodeWarrior de la firma Metrowerks .

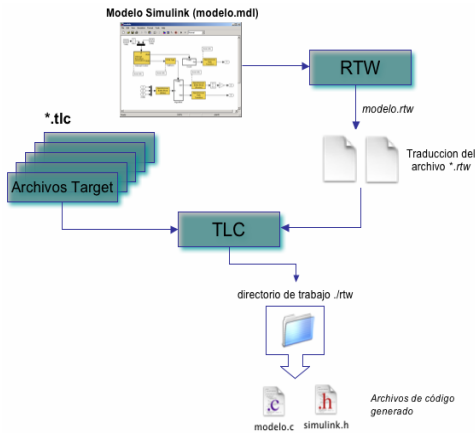


Figura 5 Generación del código hexadecimal.

C. Tarjeta de desarrollo Dragon12plus

La tarjeta que se emplea en este trabajo se muestra a continuación en la figura 5:



Figura 6 Tarjeta de Desarrollo Dragon12

D. Librerías del Real-Time MC9s12 Toolbox

Las librerías MC9S12TOOL se muestran continuación una vez que son instaladas y

configuradas tienen el siguiente aspecto en el entorno de Simulink, como en la figura 6:

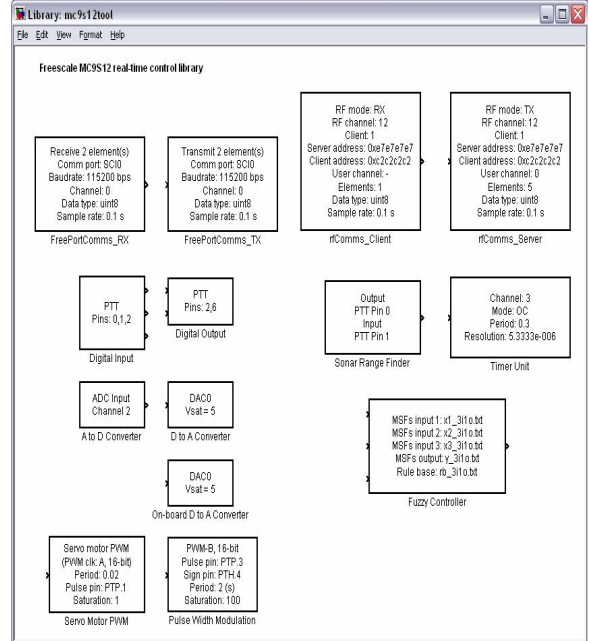


Figura 7. Librerías Real Time RTMC9S12 Toolbox Arquitectura de la tarjeta de desarrollo Dragon12 plus.

1. Controlador CAN.
2. Puerto CAN.
3. Área para prototipos.
4. Oscilador X1.
5. Oscilador X2.
6. Base para el MC9S12D256.
7. Cuatro pushbutton.
8. 8 leds conectados al puerto B
9. Comunicación RS485
10. Sensor de temperatura.
11. Dos puertos de comunicación SCI por puerto serial DB9.
12. Salida optoacoplada
13. Potenciómetros variables como entrada
14. Convertidor analógico-digital.
15. Interface I²C
16. Display grafico LCD de 16 líneas.
17. Switch de reinicio.
18. TIMER de 8 a 16 bits.
19. Regulador de voltaje de la tarjeta.
20. Diodo emisor de luz de encendido de la tarjeta.
21. Puerto serial RS-232
22. Puerto RS-485/ RS-422

E. Configuración de las librerías para el modelo de control de los servomotores.

A continuación describiremos algunas de las configuraciones de las librerías de REAL-TIME MC9S12 TOOLBOX.

E.1 Freeport bloques de comunicación

Algunas aplicaciones requieren una sencilla y fiable comunicación entre varias plataformas de un microcontrolador y aplicaciones basadas en host distinto de MATLAB.

Esta necesidad se atiende por los bloques de comunicación Freeport del toolbox. Estos bloques permiten la transmisión/recepción de mensajes con formato corto, ya sea a través de los interfaces de comunicación serie del 9S12 (SCI0 / SCI1). Los parámetros de los bloques de comunicación Freeport son: Tiempo de muestreo, puerto de comunicación, Baudrate, número de canal, número de elementos y el tipo de datos. Como se observa en la siguiente figura 8.

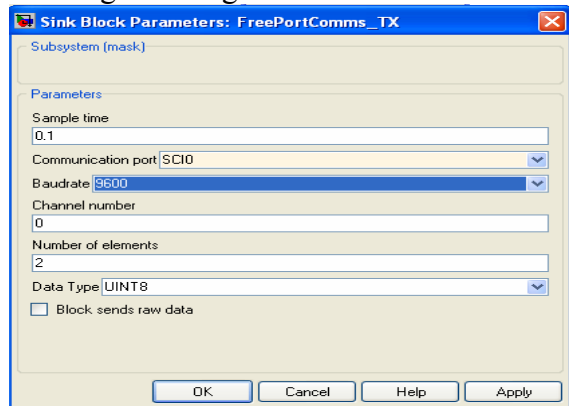


Figura 8 Parámetros del bloque de comunicación.

E.2 Convertidor analógico digital

El convertidor A/D del microcontrolador MC9S12D256, se puede acceder usando el bloque de envío del Toolbox ADC de entrada. Este bloque tiene cinco parámetros: La unidad de conversión del microcontrolador (ATD0, ATD1), la gama de canales que desea convertir (0-7), la resolución del ADC (8-bit, 10-bit), el formato de salida (datos brutos, normalizado a [0 .1], también a [0 . 5]) y tiempo de muestreo. La figura 9 muestra el parámetro solicitante del bloque de entrada.

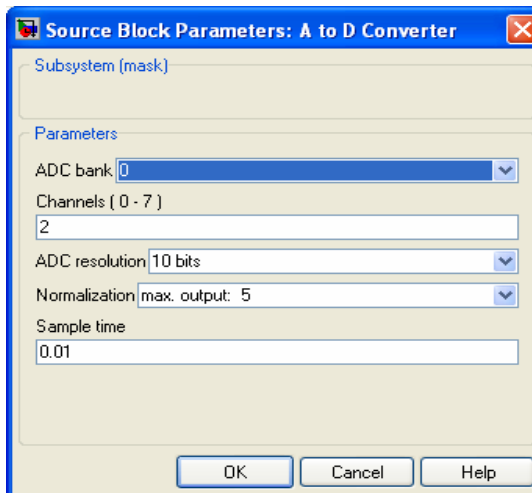


Figura 9 Parámetros del bloque del convertidor analógico digital.

E.3 Unidad para servomotor con modelación de ancho de puls

Los servomotores están manipulados por modulación de ancho de pulso (PWM) con señales de un ciclo de trabajo muy corto (normalmente 2% - 12%). Estas señales se pueden generar utilizando el bloque llamado servo motor Pulse Width Modulation. La versión actual de *rtmc9s12* dispone de 4 a 8 canales de PWM, dependiendo de la resolución de cada canal. La unidad se puede configurar para hasta 4 de 16 bits o canales de PWM hasta 8 de 8 bits de canal. Los bloques de entrada tienen que ser configurados sin signo. El bloque permite los siguientes parámetros: tiempo de muestreo, Resolución, periodo PWM, pines de pulso, mínimo ancho de pulso, máximo ancho de pulso y saturación nivel de entrada. se muestra a continuación en la figura 10.

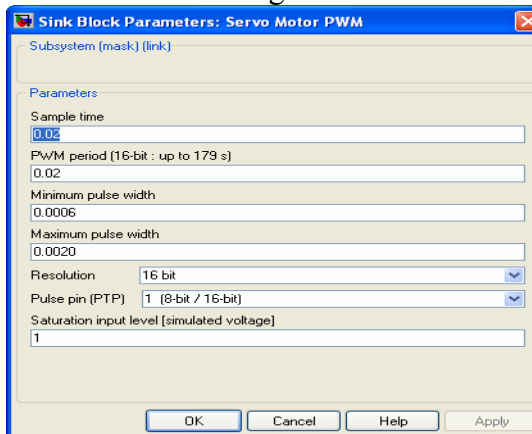


Figura 10 Parámetros del bloque Servo Motor PWM

E. 4 Diagrama a bloques para el control del movimiento de los servomotores

El diagrama a bloques para el control de servomotores se muestra a continuación en la figura 11. En este diagrama podemos observar dos bloques Servomotor PWM para las dos articulaciones con un periodo de 2 milisegundos cada uno, con saturación a 5 volts, con salida en dos pines, el 1 y 3. Todo esto solo se ejecuta en tiempo real.

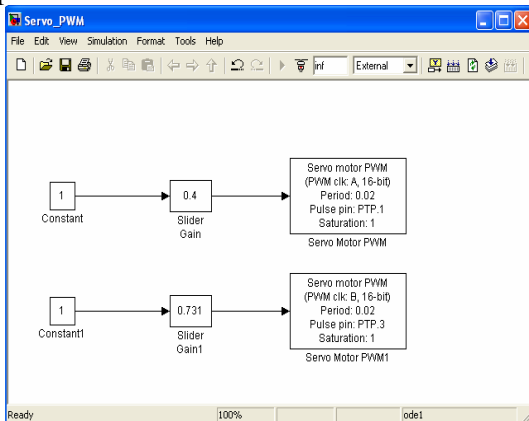


Figura 11 Modelo para el sistema para el control de servomotores.

E.5 Programación en tiempo real del código hexadecimal en el microcontrolador mc9s12d256

La programación en tiempo real del código hexadecimal que genera el compilador CodeWarrior se lleva a cabo dentro de la memoria ROM FLASH del microcontrolador, por medio del puerto serie de la PC. Se muestra a continuación en la figura 12 donde se muestra el manipulador de dos grados de libertad construido para este propósito.

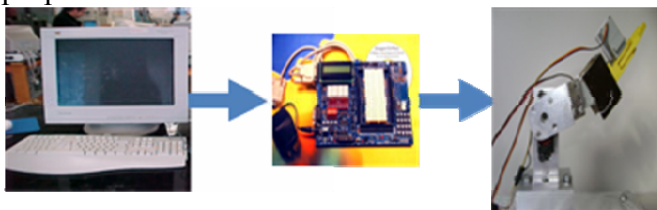


Figura 12 Sistema con el manipulador de dos grados de libertad.

V. RESULTADOS Y CONCLUSIONES

Los resultados obtenidos se pueden resumir en tres etapas; la primera es la instalación de las librerías Real Time RTMC9S12 Toolbox, la segunda es la

configuración adecuada de los parámetros de los bloques del Toolbox para llevar a cabo la interfaz con la tarjeta de desarrollo Dragon12, y la tercera es la construcción, configuración y compilación adecuada de modelos desarrollados en Simulink con los bloques de función de las librerías Real Time RTMC9S12 Toolbox.

La siguiente figura 13 se muestra la construcción del manipulador operando con la tarjeta de desarrollo Dragon12 plus.

En la figura 13 se muestra el control de las señales del control PWM que controlan las articulaciones del manipulador de dos grados de libertad.



Figura 13 Señal PWM para el control de los servomotores de las articulaciones del manipulador.

Por lo tanto el trabajo realizado hasta ahora permite vislumbrar avances importantes en el diseño de sistemas en tiempo real.

VI. REFERENCIAS

- [1] Barrientos A. , Peñin L. F. , *Fundamentos de Robótica*, 2ª. Edición, Mc. Graw Hill, 2007.
- [2] Ferreira J. C., *Microcontroladores Motorola -- Freescale*, Alfaomega, 2006.
- [3] Geschke : Geschke C., *A robot task using visual tracking.*, Robotics Today, 39-43, Winter, 1999.

- [4] Kelly R., Robust asymptotically stable visual servoing of planar robots., *IEEE Trans. Robotics and Automation*, Vol. 12., 759-766, 1999.
- [5] Kimura H., Akai K., Oike T., Robust stabilization of a flexible arm. *Proc. IEEE Conf. Decision and Control*, Tampa, Florida, Dec. 1989.
- [6] Spong M. W., M. Vidyasagar, *Robot dynamics and control*, John Wiley & Sons, 1989.
- [7] Usategui J. M. A., *Microcontroladores avanzados dspic. Controladores digitales de señales. Arquitectura, programación y aplicaciones*, Thomson, 2006.
- [8] Usategui J. M. A., dsPIC, *Diseño práctico de aplicaciones*, Mc Graw Hill, 2007.

DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA AUTOMÁTICO PARA LA PERFORACIÓN DE TABLILLA DE CIRCUITO IMPRESO.

Juan Carlos Pérez Merlos, Juan Carlos Portillo Medina.

Universidad Autónoma del Estado de México.

Facultad de Ingeniería.

División de Electrónica.

Dirección : cerro de Coatepec. s/n Ciudad Universitaria, Toluca Edo. De México.

Teléfono : (01) (722) (2-14-08-55) ext. 111, (2-15-14-53) Fax (2-15-45-12)

Correo electrónico : jcjc63@yahoo.com, pmjc_74@yahoo.com.mx

RESUMEN

En la Facultad de Ingeniería de la Universidad Autónoma del Estado de México existen las carreras de ingeniería electrónica y computación, las dos contienen asignaturas con prácticas de laboratorio y terminan con un proyecto o al final de la carrera se realiza una tesis, en ambos casos los diseños de circuitos electrónicos se implementan en protoboard y pocos los terminan con tablillas de circuito impreso por lo laborioso de éstos, incluyendo que la perforación de puntos se hace de manera manual. Este problema se presenta en muchas instituciones de educación media y superior. Con el fin de apoyar a que los proyectos tengan una mejor calidad surge la idea del trabajo “Diseño y construcción de un sistema automático para la perforación de tablilla de circuito impreso”. En el desarrollo del proceso se utilizó la herramienta de Eagle (software libre) para generar los circuitos impresos y a la vez obtener las coordenadas de los puntos a perforar en la tablilla del circuito impreso, asimismo se diseñó una estructura mecánica simple con motores de pasos para ubicar la coordenadas “x”, “y” y “z” en la tablilla. Se hizo un programa en lenguaje “C” para el control de los motores utilizados y otro en lenguaje ensamblador de un microcontrolador para manipular el eje “z” que es el que perfora la tablilla.

Palabra clave : Instrumentación, control con microcontrolador, automatización.

I INTRODUCCIÓN

A nivel nacional hay muchas instituciones de nivel medio superior y superior que tiene carreras de electrónica, computación y mecatrónica o alguna similar en la que se requiere hacer circuitos impresos. Tal es el caso de la Facultad de Ingeniería de la Universidad Autónoma del Estado de México, que cuenta con cuatro carreras de nivel superior, dos de ellas están relacionadas, ingeniería en electrónica e ingeniería en computación, en ambas se contemplan asignaturas en las que se hacen prácticas y proyectos de electrónica o se generan tesis donde se involucran sistemas electrónicos los cuales se montan en tablillas protoboards y difícilmente se logran llevar a una tablilla de circuito impreso (tablilla de baquelita cubierta de cobre donde se montan los dispositivos electrónicos bajo un diseño especificado), debido al proceso que éste lleva. El proceso inicia realizando el diseño electrónico en un software especial (Orcad, Pcb, Eagle, Protel, Tango) el cual puede generar comúnmente el diseño del circuito impreso correspondiente, en otras palabras las conexiones de los dispositivos electrónicos usados, éstas pueden estar en una sola cara o en doble cara. Después de ello, el diseño del circuito impreso generado es plasmado en el cobre de la tablilla de baquelita a través de técnicas que van desde manuales hasta un poco más elaboradas como son tinta indeleble, logotipo, transferencia térmica, fotográfica y serigráfica. Enseguida se pasa a la

perforación de los puntos de conexión con un taladro que permita trabajar con brocas de tamaño 1.588 o 7.9375 milímetros (mm) (tamaño de brocas comunes utilizadas para las perforaciones de circuitos impresos). El proceso de perforación típicamente es manual, utilizando un banco donde se sujeta la tablilla y se perfora cada punto esto es muy cansado y cuando son muchos puntos esto hace que se hagan perforaciones con muy baja precisión.

II MÉTODO

El proyecto se inició con el establecimiento de los requerimientos los cuales se desglosan a continuación:

Requerimientos para el diseño del sistema:

La tablilla a perforar estará limitada las dimensiones de 200*200 mm. o menor.

Mejorar el tiempo de respuesta de la perforación de puntos de manera manual.

Que la lectura de las coordenadas generadas sea automática.

Que el sistema mecánico sea portátil y ligero.

Que el proceso sea totalmente automático.

Que la precisión sea la suficiente para que se perfora en el punto y éste pueda conectarse.

Partiendo de lo anterior se inició el proyecto con la selección del software base que se utilizaría para obtener las coordenadas de los puntos a perforar del circuito impreso, Eagle fue el seleccionado debido a que es un software libre y se utiliza ampliamente incluso en nuestra Facultad. Eagle genera un archivo con extensión *.dri que contiene todas las coordenadas (las puede presentar para varias dimensiones de brocas identificadas como T01, T02 etc.) de los puntos de interconexión del diseño electrónico respectivo, este archivo es el que se utilizó para obtener los datos de los puntos a perforar. Para leer el archivo de coordenadas se hizo un

programa en Turbo C que lo primero que hace es abrir e identificar el dato de la coordenada “x” e “y”, esto se hizo utilizando la lectura del archivo usando Tokens y dejar las coordenadas en forma de vectores para después ser enviados al puerto paralelo de la computadora.

Con los datos en forma de vectores, se hizo enseguida una propuesta de diseño del sistema el cual se decidió dividirlo en una estructura mecánica, un sistema electrónico y un programa de control como se muestra en la figura 1.

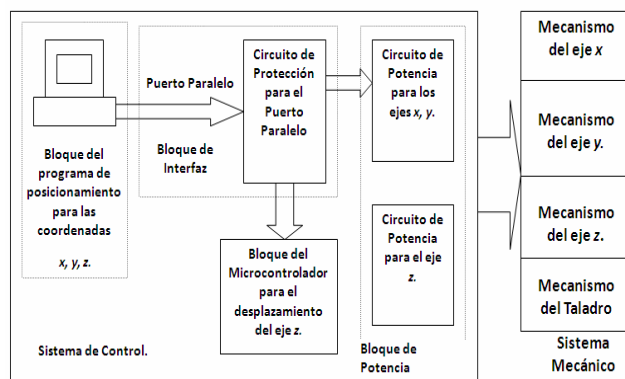


Figura 1.- Conjunto de partes del sistema de perforación propuesto.

Para el sistema mecánico se diseñó un croquis de situación para ubicar la distribución y cantidad de componentes como se muestra en la figura 2.

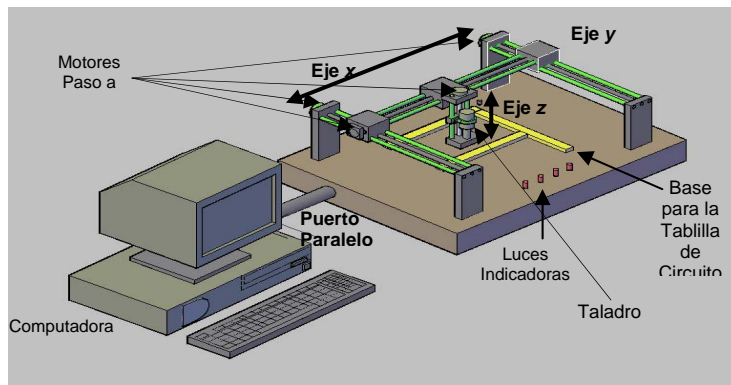


Figura 2.- Propuesta del sistema mecánico (croquis de situación).

La propuesta para la interconexión del eje “x” y el “z” quedó como lo muestra la figura 3. Los tres orificios de la parte superior son para el ensamble con el eje “x”, la parte circular identifica la colocación del motor a pasos para el deslizamiento del eje “z”, y el orificio inferior es para colocar el motor de corriente directa que realiza la operación de taladrar.

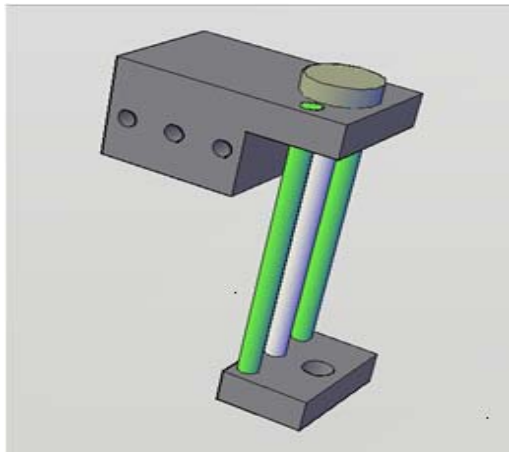


Figura 3.- Mecanismo de interconexión del eje “x” y “z”.

La propuesta del taladro quedó de la siguiente forma, una base de sujeción con el mecanismo del eje “z”, una abrazadera para sujetar al motor de corriente directa el cual lleva la broca que va a perforar la tablilla haciendo la función de taladro, como lo muestra la figura 4.

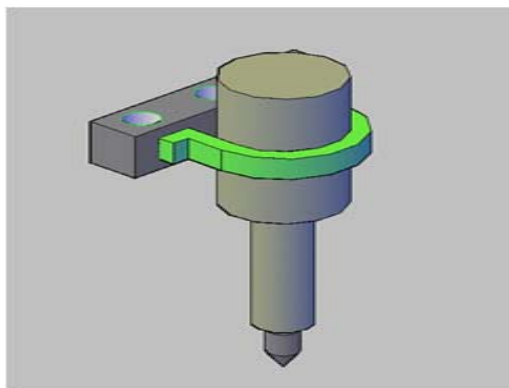


Figura 4.- Propuesta del taladro.

Después de la propuesta se construyó cada parte y quedó de la siguiente manera: para el eje “y” dos estructuras paralelas montadas sobre dos pilares fijos (de 170 mm. de alto, 50 mm. de largo y 12 mm. de ancho), cada una formada por dos rieles paralelos hechos de varillas de aluminio (de diámetro 9.525 mm. y 320 mm. de longitud) y entre ellos un tornillo de fuerza el cual tiene en su extremo un motor a pasos bipolar de 15° de giro por paso, cuando gira el motor el tornillo transforma el giro en movimiento. Estas dos estructuras están unidas a través de otra similar puesta perpendicular a los anteriores la cual forma el eje “x”, también con sus dos rieles de aluminio (diámetro de 9.525mm. y 400 mm. de largo) y su tornillo de fuerza (diámetro 9.525mm y 400 mm. de largo) entre los dos rieles con su respectivo motor a pasos bipolar de 1.8° de giro por paso ver la figura 5.

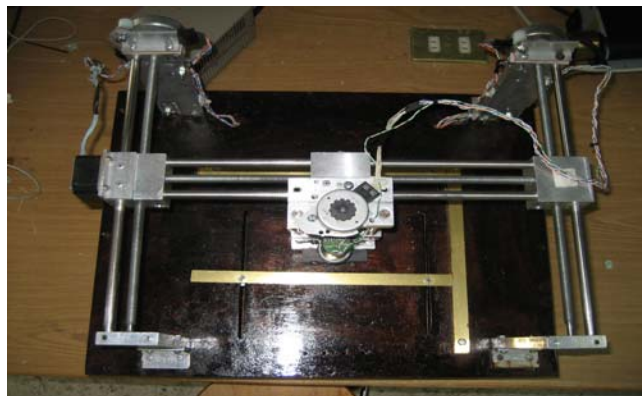


Figura 5.- Sistema mecánico de perforación de tablilla de circuito impreso vista superior.

En el eje “x” esta montado sobre el riel otra pequeña estructura similar a las anteriores, formada por dos rieles paralelos (diámetro de 9.525 mm. 166 mm. de longitud) con su respectivo motor a pasos unipolar de 15° de giro por paso en el tornillo de fuerza (diámetro 7.9375 mm.) que a su vez sostiene una base que sujeta con una abrazadera a un motor de corriente directa que forma el eje “z”, este motor se acondicionó en su flecha, con una punta de aluminio de 45 mm de longitud y 3 mm de

diámetro, perforada por un extremo para poder colocar la broca que tiene contacto directo con la tablilla y ejecuta la perforación de la misma, ver la figura 6

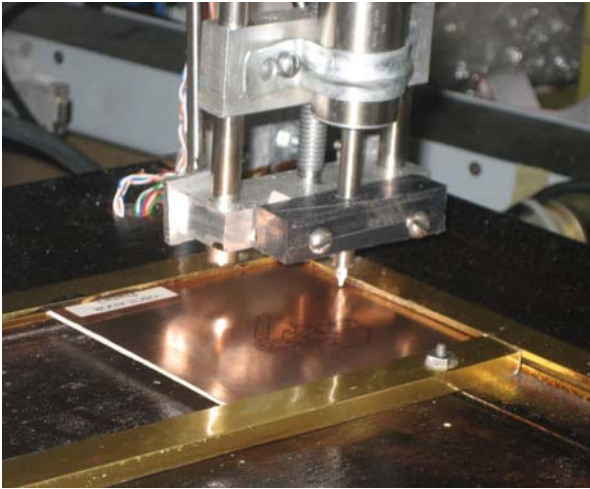


Figura 6. Muestra el motor de corriente directa, montado en el eje “z” y la estructura que sujeta a la tablilla de circuito impreso.

Todo está montado sobre una base de madera cuyas dimensiones son 500*400*150 mm. de largo, ancho y alto respectivamente.

La construcción final de la estructura se puede observar en la figura 5, es una vista superior la cual muestra las barras de aluminio de los ejes “x” e “y”. En la parte inferior se puede distinguir el sujetador de la tablilla de circuito impreso. En el centro del eje “x” esta el motor a pasos y allí mismo el taladro que conforma el eje “z”.

El diseño electrónico consta de una etapa de potencia para el manejo de los motores, el driver push-pull L293B proporciona la corriente suficiente para el motor bipolar del eje “x” y el motor del taladro que es de c.d., un TIP31 sostiene la corriente de los motores unipolares del eje “y” (dos motores sincronizados). Del puerto paralelo de la computadora se utilizó el puerto de datos 4 bits (1:3) para controlar los

motores del eje “y”, 4 bits (4:7) para el eje “x”, dos bits del puerto de control (1:2) para luces indicadoras de activación de los ejes “y” y “x”, un bit (0) para el control de microcontrolador el cual trabaja como esclavo de la computadora, este es el PIC 16F84 de la familia de Microchip, controla a través de 4 bits (RB0:RB3) de su puerto RB al motor del eje “z”, y dos bits del mismo puerto (RB6:RB7) para el taladro el cual gira en ambos sentidos, uno para perforar y el otro para salir del agujero sacando la rebaba. El diagrama electrónico a bloques se muestra a continuación en la figura 7.

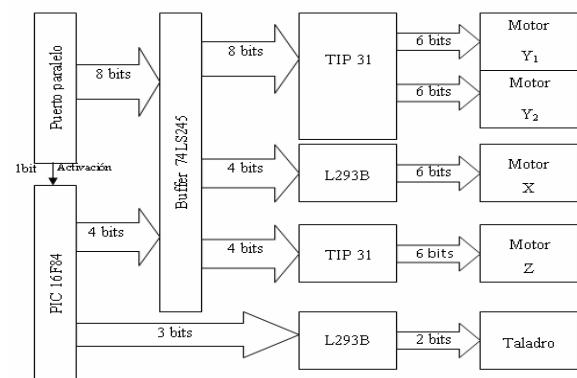


Figura 7.- Diagrama a bloques del sistema electrónico.

El programa hecho en lenguaje “C” en la computadora tiene como funciones principales, número uno, leer el archivo de coordenadas (generado por Eagle), número 2, mandarlas al puerto paralelo para mover los motores del eje “x” e “y”, número 3 activar al microcontrolador (el microcontrolador tiene un programa en ensamblador que maneja al eje “z” y al motor de c.d. para hacer la perforación).

Para perforar la tablilla de circuito impreso, esta se coloca en un riel donde se tiene un punto de referencia previamente identificado, cuando se llega a dicho punto la tablilla es sujeta por medio de unos tornillos para que no se mueva debido a la inercia del giro del taladro. La alimentación del sistema es con tres niveles de voltajes 5 voltios para microcontrolador y

buffers, 5 voltios para los drivers de corriente y 12 voltios para el bloque de potencia. El sistema completo se conecta a 110 volts de c.a. y consume una corriente de 4 amperes, cuenta con un interruptor de arranque y luces indicadoras de el eje que se esta moviendo y otra para el taladro, el equipo completo se puede observar en la figura 8.

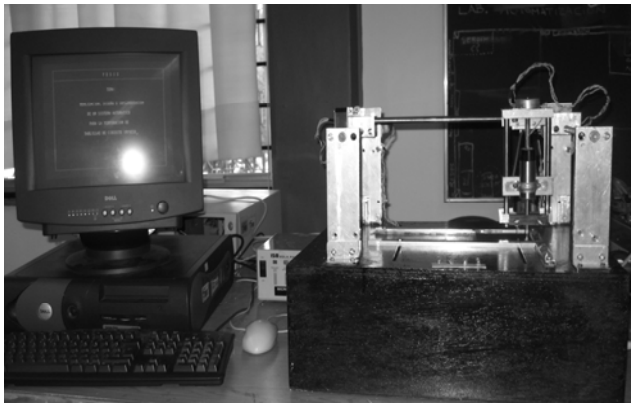


Figura 8.- Sistema de perforación completo

Las pruebas realizadas fueron inicialmente ejecutadas parte por parte y luego se hizo una integración de ellas hasta completar todo el sistema. En primer lugar se tuvo problemas con los rieles del eje “y”, inicialmente no giraban los motores debido a un descuadre de los dos rieles que lo conforman, se corrigió y funcionó adecuadamente. Cuando se deseaba perforar, la broca se desplazaba al contacto con la superficie de la tablilla y no la penetraba, esto se eliminó poniendo dos varillas en la parte trasera del taladro para dar más firmeza a la punta del taladro.

Se realizó un diseño de un oscilador, de donde se obtuvieron 20 coordenadas de puntos a perforar, el sistema lo hizo en 7 minutos 40 segundos, manualmente se hizo en 15 minutos.

La tasa de error se mantuvo dentro $\pm 1\text{mm}$ en los primeros 50 puntos y después aumenta, esto es debido al tipo de motor de pasos utilizado, se puede mejorar cambiando los motores de 15° por otros de 1.8° , aquí se solucionó por programación, después de 50 puntos se hace

regresar al taladro al punto de referencia para que vuelva a comenzar con la siguiente coordenada y mantener la perforación dentro de la precisión aceptable, ver la figura 9.

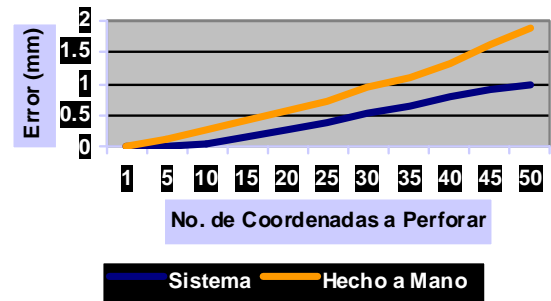


Figura 9. Error presentado del sistema automático (oscila entre $\pm 1\text{mm}$).

Otro punto determinante fue el caracterizar la relación por vuelta de cada motor y la distancia recorrida, para esto se observó que el motor del eje “y” debe de dar 768 pasos para recorrer una pulgada y el del eje “x” 3200 (debido a los grados de giro por pulsos). Con relación al tiempo de perforación de la tablilla de circuito impreso se hizo una comparación entre una práctica manual y la automática, se puede observar que en 50 puntos el sistema es más rápido, como se demuestra en la figura 10.

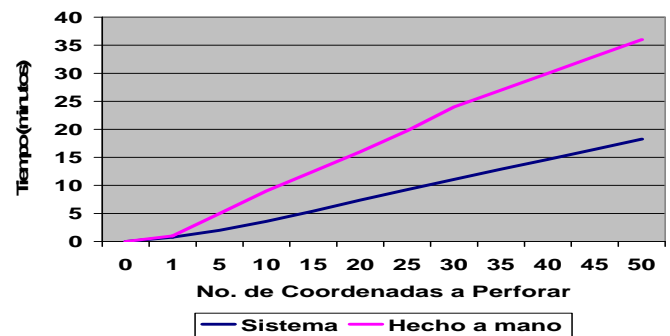


Figura 10.- Comparación de perforar 50 puntos hechos a mano y en automático.

El sistema en su conjunto usa elementos tanto mecánicos, electromecánicos y electrónicos que se consiguen fácilmente en el mercado, lo cual puede representar una ventaja para su reproducción. Existen trabajos al respecto, ejemplo de ello es el control de movimientos coordinados de una mesa trazante [2]. En este proyecto se controlaron motores de corriente directa con microprocesadores, por lo que la electrónica es compleja y además ocupa gran espacio, otra característica es que la mesa es pesada.

El trabajo aquí expuesto fue hecho con motores a pasos controlados con la computadora y un microcontrolador y de una estructura ligera.

III CONCLUSIONES

El sistema diseñado cumple con el propósito de perforar tablillas de circuito impreso de 50*50 y hasta 200*200 mm.

Entre más cantidad de puntos a perforar el sistema automático supera en mucho a la práctica manual.

En el tiempo de perforación de un punto influye la distancia a la que se encuentra de la referencia, la perforación se realiza en solo 15 segundos.

El sistema puede ser reproducido para tabillas más grandes, dado que se cuenta con todo el diseño.

Este sistema es económico en cuestión de los componentes ya que costo fue de \$3,500.00 aproximadamente y los dispositivos pueden ser fácilmente conseguidos.

Aunque el funcionamiento del sistema es sencillo se realizó su manual de operación.

Los requerimientos de la computadora utilizada son mínimos, se puede decir que funciona con cualquiera.

Los productos obtenidos fueron una tesis, un manual de operación y mantenimiento, y el sistema de perforación.

El sistema trabaja en lazo abierto se espera en trabajos futuros hacerlo en lazo cerrado para

garantizar la perforación, que pueda leer coordenadas de otro software y cambiar los motores de mejor resolución para mejorar la precisión.

Lo invertido en este sistema realmente es poco con respecto a los de marca que suelen costar mucho (como el Unimatic modelo CNC 20 de Inteltech S.A. de C. V.), en algunos casos hasta \$300,000 pesos M/N (y su reparación en caso de daño hasta 50,000 pesos M/N). La compra de un equipo de marca de esta magnitud es aconsejable y justificable si alguna institución se dedica a realizar circuitos impresos en masa y de manera continua de allí se puede recuperar la inversión. Comúnmente no es el caso de las instituciones educativas.

IV REFERENCIAS

- [1].- Circuitos Impresos Teoría, Diseño y Montaje, José González Calabuig, Paraninfo, Madrid España, 1997.
- [2].- Diseño de un Sistema de Control para Movimientos Coordinados en tres ejes, Mejía Vázquez Hector, CICESE, 1991.
- [3].- Microchip “ PIC 16F84 Data Sheet 18 pin Enhanced FLASH/EEPROM 8 Bit Microcontroller”, Microchip Technology Inc., 2001.
- [4].- The Waite Group's Essential Guide to Turbo C, Howard W. Sams & Company, 1991.
- [5].- Ingeniería de Control Moderna, Ogata Katsuhiko, Perntice Hall , 2004.
- [6].- Stepping motors and their microprocessor controls, Takashi Kenjo, Clarendon Press Oxford, 1986.
- [7].- Electrónica de Potencia, Muhammad H. Rashid, Prentice Hall, 1995.

DISEÑO DE UN SISTEMA DE INSTRUMENTACIÓN PARA UN SISTEMA DE GENERACIÓN DE FUERZA EN UN SIMULADOR DE CADERA MECÁNICO

Rangel Carrillo Jorge Antonio.

Departamento de Mecatrónica
Universidad Tecnológica de Torreón
Carr. Torreón – Matamoros Km. 10 s/n, Ejido El Águila, Torreón, Coah. C.P. 27400, México
Tel. 01 (871) 7297400 Ext. 244, correo electrónico: jrangel@utt.edu.mx

RESUMEN

El presente artículo trata de una investigación que está orientada a proponer un sistema de instrumentación que pueda utilizarse en un simulador de cadera mecánico, el cual es un equipo usado en estudios de tribología de las prótesis totales de cadera. Este sistema de instrumentación consiste básicamente en la generación de una fuerza variable en un periodo de tiempo, junto con un sistema de adquisición de datos, durante un ciclo de marcha normal en un humano, con la ayuda del software de instrumentación LabVIEW 8.0. Se presentan los resultados obtenidos en un simulador mecánico de cadera que se encuentra en el laboratorio de biotribología de la Universidad Autónoma de Nuevo León.

Palabras clave: Simulador de cadera, prótesis total de cadera, fuerza variable, adquisición de datos, LabVIEW.

I. INTRODUCCIÓN

Existen diferentes líneas de investigación que se han enfocado a mejorar la calidad de vida en el ser humano. Una de ellas es la referente al estudio del desgaste en las articulaciones artificiales (biotribología), tal es el caso para los implantes de cadera que sustituyen la articulación natural debido a enfermedades como osteoartritis o fracturas por accidentes [1].

Estas articulaciones artificiales de cadera implantadas en el cuerpo humano, desprenden

millones de partículas debido al desgaste, provocando debilitamiento de la unión (osteólisis), reacciones adversas en los tejidos y daño en los huesos, lo cual obliga a una reintervención quirúrgica para reemplazar nuevamente el implante.

Actualmente en México existe una creciente demanda de implantes quirúrgicos para articulaciones, la mayoría de las prótesis son importadas a precios muy altos, y en algunos casos, con características poco aceptables en cuanto al diseño. Esto genera un área de oportunidad la cual se ha convertido en un reto multidisciplinario para la autosuficiencia de endoprótesis [2].

El esfuerzo de la comunidad científica mundial en los últimos años ha sido el desarrollar una prótesis total de cadera satisfactoria, es decir, con una mayor longevidad.

El diseño apropiado de una prótesis total de cadera, involucra la evaluación del diseño final bajo condiciones in vitro. Lo anterior lo podemos lograr con la ayuda de un simulador de cadera, que es un equipo utilizado para la evaluación del desempeño tribológico de las prótesis totales de cadera.

Cadera humana

La articulación de la cadera es del tipo esférico, que además de permitir el movimiento, ofrece la estabilidad necesaria para soportar el peso del

cuerpo. Esta formada por una bola que es el final del hueso fémur y un recipiente articular en forma de cavidad, para ajustarse a esta bola femoral depende del hueso de la pelvis [3].

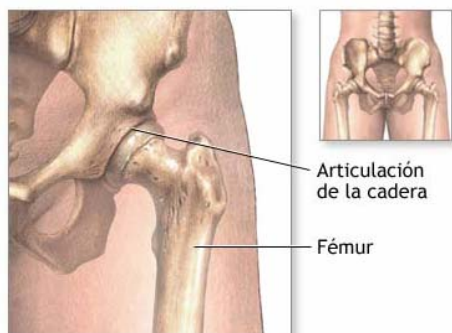


Figura 1. Articulación de la cadera humana.

La función de la articulación de la cadera es orientar al miembro inferior en todas las direcciones del espacio. Por lo que va a estar dotada de movimientos en las tres direcciones, gracias a la disposición de una superficie esférica sólida que penetra en el interior de una superficie esférica hueca.

Por otra parte, el implante de cadera está sometido a una carga variable según las investigaciones de John Paul [4], quien encontró que la carga con respecto al tiempo de un ciclo de marcha se comporta según la curva mostrada en la figura 2, esto es debido a la distribución de peso entre las dos caderas durante un ciclo de marcha cuya frecuencia es de 1.1 Hz. Este se le conoce como el ciclo de Paul.

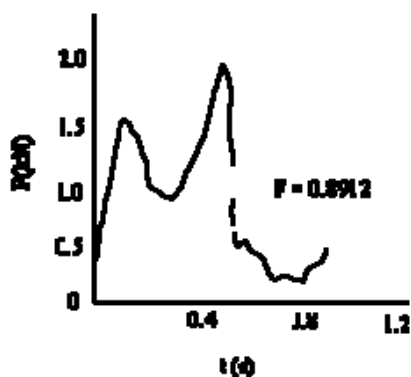


Figura 2. Carga que actúa en una cadera humana durante un ciclo de marcha.

Prótesis de cadera

Sin duda alguna la prótesis de cadera es el sistema protésico que ha sido más ampliamente estudiado, tanto por ser un sistema relativamente sencillo como por ser la articulación cuyo reemplazo fue abordado en primer lugar de forma masiva y con un éxito relativo. Lo anterior ha dado pie al desarrollo de sistemas y técnicas destinadas al reemplazo de otras articulaciones, de tal suerte que las causas de los éxitos y fracasos de las prótesis de cadera, han servido y sirven todavía de base para el desarrollo de otras prótesis articulares.

Los componentes óseos dañados de la articulación de cadera que son la copa acetabular de pelvis y cabeza femoral, son sustituidos por un implante el cual se compone por una copa o acetábulo que se fija en la pelvis y funciona como asiento para un segundo elemento: la esfera artificial la cual esta sujeta a un vástago que se fija en el hueso fémur. Lo anterior se conoce como prótesis total de cadera, y se puede observar en la figura 3.

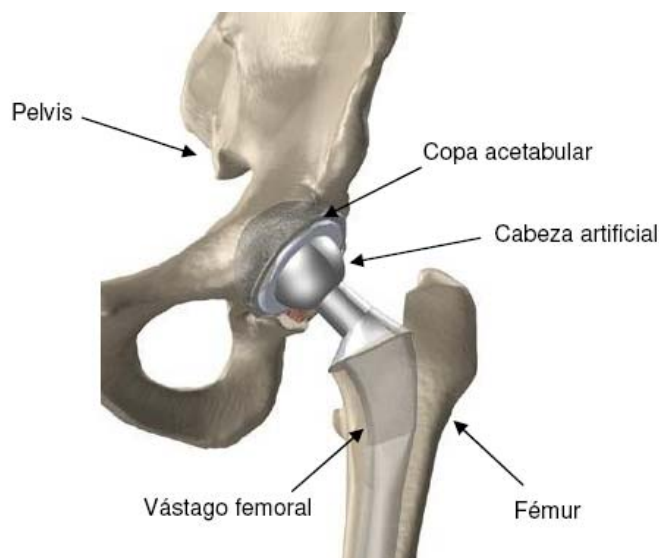


Figura 3. Prótesis total de cadera.

Simulador de cadera

Un simulador de cadera tiene como objetivo reproducir las condiciones de carga variable y movimientos complejos en tres dimensiones en las superficies de contacto, tan cercanamente posible a

las condiciones “in vivo”. Por lo tanto, los factores que necesariamente se tienen que considerar son: velocidad, dirección del movimiento, dirección y magnitud de la carga, composición del lubricante y temperatura [5]. Los simuladores de cadera se han convertido en una importante herramienta tanto para el área de investigación, como para evaluaciones preclínicas, minimizando el riesgo del paciente cuando recibe nuevos tipos de implantes quirúrgicos. Además, puede proporcionar información valiosa para mejorar el diseño de la nueva generación de implantes de cadera metal-metal. [6]. Bajo este contexto, el diseño de un simulador debe considerar el problema de la generación de ciclos de movimiento relativo entre dos superficies articuladas, mientras se aplica una fuerza de contacto que varía en magnitud durante un ciclo de marcha [7]. En esta clase de equipos tenemos la opción de probar varias prótesis al mismo tiempo, lo cual reduce significativamente los tiempos de prueba.

Actualmente los principales fabricantes de simuladores de cadera en el mundo son INSTRON, PLINT y ProSim Ltd., los cuales ofrecen equipos sofisticados y de alto costo.

Simulador de cadera FIME II

En el laboratorio de biotribología de la Facultad de Ingeniería Mecánica y Eléctrica de la Universidad Autónoma de Nuevo León, a cargo del Dr. Marco A. L. Hernández Rdz. y el M.C. Javier Alonso Ortega Sáenz, se diseñó y fabricó un simulador de cadera mecánico el cual provee ciclos de carga variable con parámetros muy cercanos a los de la fisiología humana durante caminata y trote, además incluye el fenómeno de microseparación y los movimientos de la cadera de flexión-extensión (FE), abducción-adducción (AA), rotación interna-externa (IER), con las amplitudes de: $\pm 23^\circ$ para FE, $\pm 23^\circ$ para AA y $\pm 8^\circ$ para IER. Por otra parte, el simulador cuenta con un sistema de lubricación similar al del cuerpo humano [8]. Además se tiene un motor eléctrico monofásico de 1 HP de potencia, el cual provee el movimiento rotacional a cada una de las estaciones a una velocidad de 1.4

Hz. Lo anterior se logra mediante un sistema de reducción de velocidad, compuesta por una transmisión acoplada al motor, con una relación de 60:1 en conjunto con un sistema de engranes [9].



Figura 4. Simulador de cadera FIME II.

II. MÉTODO

Debido a que el simulador anterior carecía de un sistema de instrumentación, generándose una área de oportunidad para llevar a cabo este proyecto, en donde se propuso implementar un sistema de control para éste simulador. Para la realización del sistema de control se analizaron los comportamientos de una celda de carga como sensor de fuerza, un regulador electroneumático como elemento de mando, un músculo neumático como elemento actuador y un circuito de acondicionamiento, todo interconectado de una manera lógica y en conjunto con el programa LabVIEW 8.0 y una tarjeta adquisitora de datos para monitorear y controlar el sistema completo. Por último, se estudiaron los resultados que se generaron al unir la parte electrónica con la parte mecánica, dando como resultado un control flexible [10]. Dicho sistema consta de dos partes:

- Sistema de Adquisición de Datos.
- Sistema Generador de Fuerza.

En la figura 5 se presenta el diagrama a bloques del sistema de adquisición de datos.

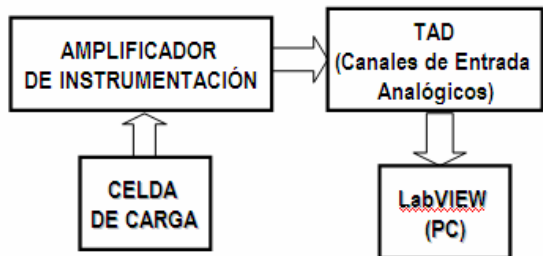


Figura 5. Sistema de adquisición de datos.

Para el sensor de fuerza se seleccionó una celda de carga marca HBM, INC. con las características que se muestran en la tabla 1.

Modelo:	S35-1.5K-20575
Resistencia de Entrada:	386.9 Ω
Resistencia de Salida:	350.3 Ω
Sensibilidad:	2.9621 mV/V
Carga Máxima:	6672.332 N
Voltaje de Excitación:	5 vcd

Tabla 1. Características del sensor de fuerza.

La carga máxima que puede sensar este dispositivo es de 6672.332 N. Partiendo de la idea de que la fuerza máxima que se presenta en el simulador es de 2000 N, según el ciclo de Paul, con esta celda estamos sobrados en capacidad para tener lecturas por debajo del posible máximo valor sensado, lo anterior para no forzar al dispositivo sensor de fuerza, y por consiguiente se presente una vida útil más larga.

Para el amplificador de instrumentación se utilizó uno compuesto por tres amplificadores operacionales con una ganancia de 337.6, ya que el voltaje de salida máximo que se presenta en la celda de carga es de 14.81 mV y el voltaje necesario en la tarjeta adquisitora de datos es de 5 vcd.

En la figura siguiente se muestra el circuito final utilizado.

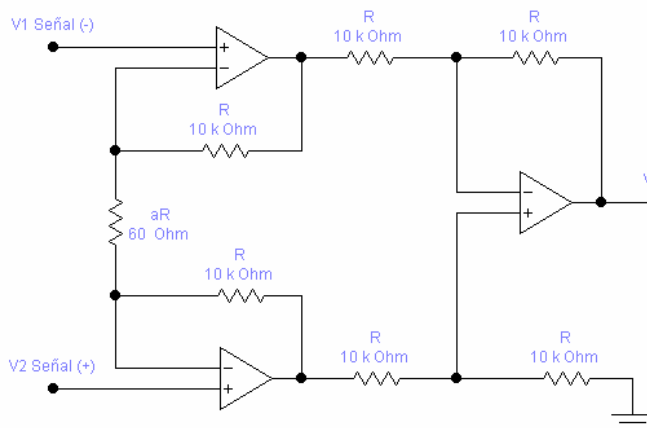


Figura 6. Amplificador de instrumentación.

La señal de salida del amplificador de instrumentación se conecta a uno de los canales de entrada de una tarjeta adquisitora de datos de National Instruments NI USB-6008. Como el simulador mecánico tiene cuatro estaciones de prueba, es necesario fabricar cuatro amplificadores de instrumentación.

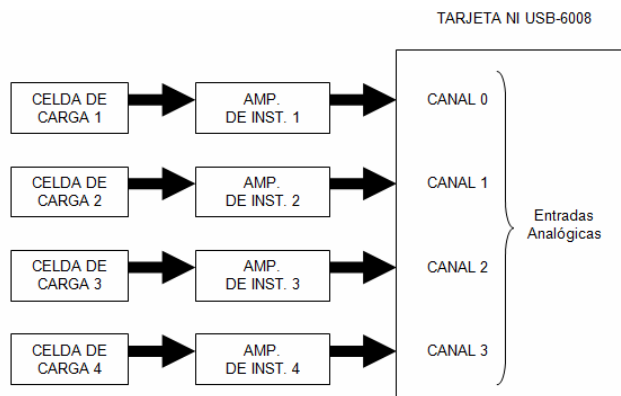


Figura 7. Conexión de las celdas de carga, los amplificadores de instrumentación y la tarjeta NI USB-6008.

El graficado en tiempo real de las señales obtenidas de las celdas de carga, se hizo con la ayuda del software de instrumentación LabVIEW 8.0. Los bloques empleados para generar la rutina

para leer los datos de las celdas, se muestran en la figura 8.

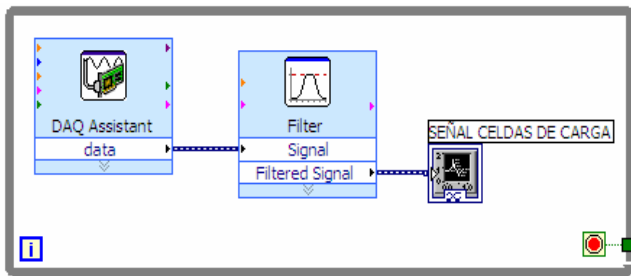


Figura 8. Rutina para leer las señales de celdas de carga.

Por otro lado, para el sistema generador de fuerza el cual se muestra en la figura 9, la idea es la de gobernar la operación de un regulador electroneumático a través de una señal de voltaje generada en un canal de salida de la misma tarjeta NI USB-6008, de tal forma de usarla también para esta parte del sistema de instrumentación del simulador. A su vez, el regulador electroneumático controla el accionamiento y la fuerza generada en un músculo neumático de la marca FESTO. Esta fuerza es la que se aplica a cada prótesis de cadera en el equipo simulador.

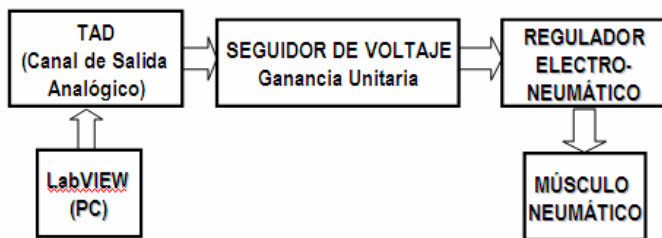


Figura 9. Sistema generador de fuerza.

Primeramente se generó una señal de voltaje variable en el tiempo, para luego mandarla a escribir a un canal de salida de la tarjeta NI USB-6008. Para proteger la tarjeta de un posible daño eléctrico y ganar corriente, se decidió usar un seguidor de voltaje entre la tarjeta y el regulador electroneumático.

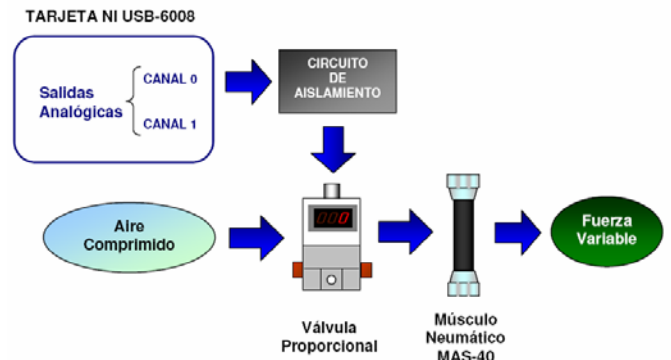


Figura 10. Conexión entre la tarjeta de control y el regulador electroneumático.

En la figura siguiente se observa la programación generada en LabVIEW, donde se configura la señal de voltaje, de tal forma que a través de esta señal estamos controlando una presión de aire y a su vez, una fuerza variable.

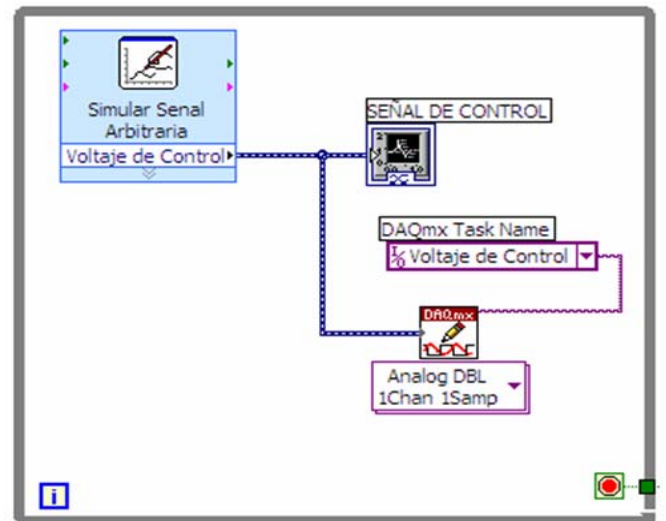


Figura 11. Rutina para generar la señal de voltaje variable.

Si la señal de voltaje variable tiene un valor de 0 V, la fuerza medida en las celdas de carga corresponde a un valor de 0 N, tal como se muestra en la figura 12.

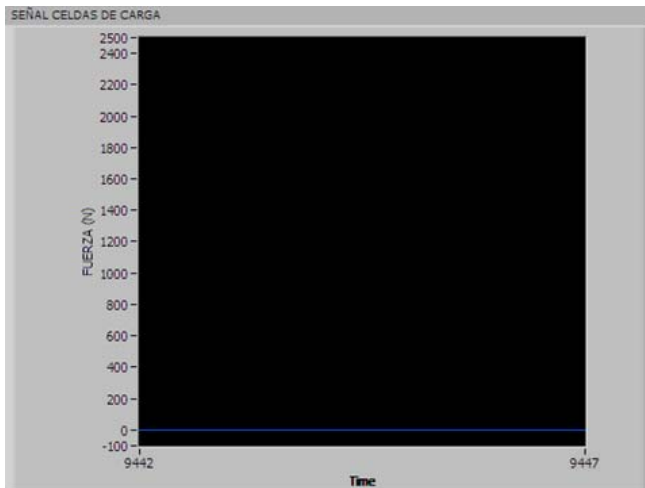


Figura 12. Fuerza medida en celdas a 0 V.

Por otro lado, si aumenta la señal de voltaje variable a 5 V, la fuerza aplicada a las prótesis de cadera aumenta a un valor aproximado de 5690 N, como se puede observar en la siguiente figura.

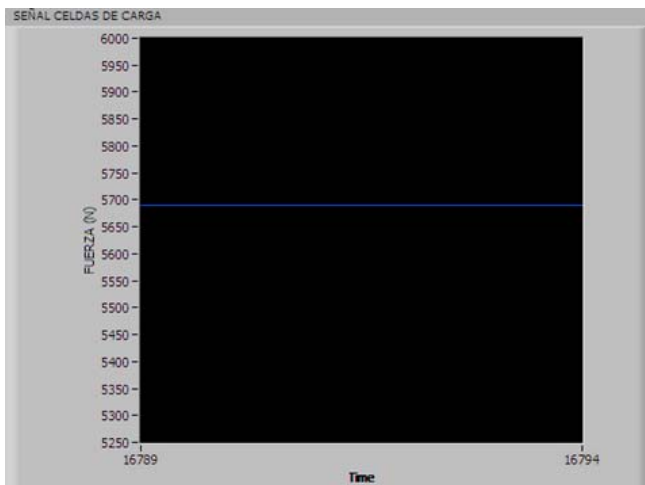


Figura 13. Fuerza medida en celdas a 5 V.

La señal de voltaje variable define la fuerza que se presenta en cada una de las prótesis en el simulador, por lo que es necesario generar y ajustar una señal de tal forma que la fuerza generada sea lo más cercana posible al ciclo de Paul.

En la figura 14 se muestra la señal de voltaje necesaria para generar la fuerza variable en un ciclo de marcha en el simulador. Mientras que la

figura 15 corresponde a la lectura que se obtuvo en una celda de carga. Nótese que existe una gran similitud al ciclo de Paul, igualando en gran medida las condiciones que se presentan en una cadera humana.

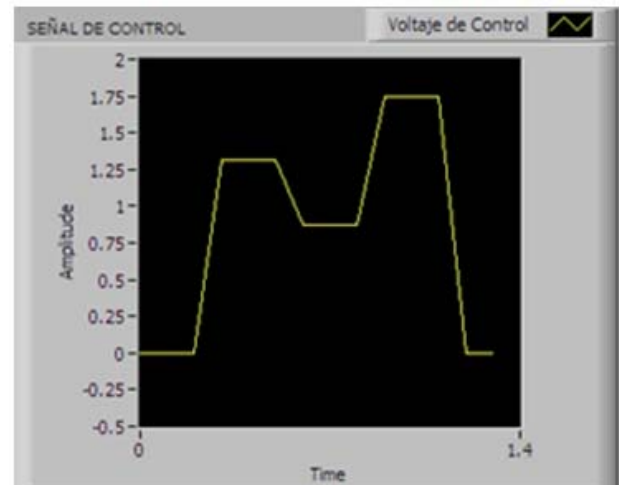


Figura 14. Señal de voltaje para el regulador electro neumático.

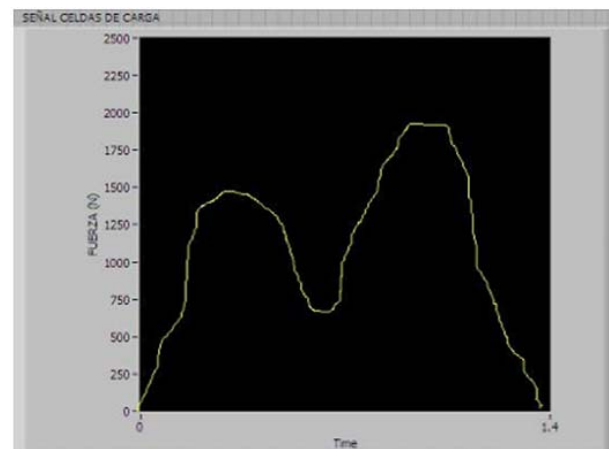


Figura 15. Fuerza aplicada en una estación del simulador de cadera.

Por último la tabla 2 muestra la relación final voltaje-fuerza que se obtuvo en el sistema de instrumentación implementado en el simulador de cadera mecánico. La señal de voltaje puede oscilar en un rango de 0 a 5 vcd, proporcional a una fuerza de 0 a 5700 N.

VOLTAJE (vcd)	FUERZA (N)
0	0
1	1140
2	2280
3	3420
4	4560
5	5700

Tabla 2. Relación voltaje-fuerza en el simulador de cadera.

III. CONCLUSIONES

Se ha presentado y comprobado la instrumentación necesaria en un simulador de cadera mecánico, con la finalidad de controlar y monitorear una fuerza variable a la cual están sometidas las prótesis de cadera durante un estudio biotribológico, y con ello, ayudar en la obtención de resultados lo más pegados a la realidad y así contribuir en el buen diseño de prótesis totales de cadera.

La aportación de este estudio se limita a ofrecer una herramienta que sea utilizada para futuras investigaciones en el campo de la biotribología, además en México sólo existe el simulador mecánico de cadera que se presentan en este artículo, por lo que le da más valor a la investigación descrita.

Agradecimientos

El autor de este estudio agradece a la Facultad de Ingeniería Mecánica y Eléctrica de la Universidad Autónoma de Nuevo León (FIME-UANL) y a la Universidad Tecnológica de Torreón (UTT), por el soporte de infraestructura para esta investigación.

IV. REFERENCIAS

- [1] J. A. Rangel Carrillo, M.A.L. Hernández Rdz., J. Ortega Sáenz, R. Mercado Solís, L. M. Torres Treviño, F. A. Reyes Valdés. Desarrollo de la instrumentación necesaria en un simulador de cadera mecánico. V Congreso Internacional de Metal-Mecánica 2006, Instituto Tecnológico de Durango. Octubre de 2006.
- [2] M. A. L. Hdz. Rdz., A. Pérez, R. Mercado, D. Martínez, M. Cantú, "Wear of cast metal-metal pairs for total replacement hip prostheses" *Wear*, Vol. 259 part 2, p. 958-963, 2005.
- [3] M. Comím, R. Dejoz, C. Atienza, J. Prat, J. L. Peris, P. Vera, A. Gil, C. Reig, *Biomecánica articular y sustituciones protésicas*, primera edición, Valencia, 1998.
- [4] J. P. Paul, Approaches to desig-force action transmitted by joints in the human body, *Proc. R. Soc. Lond. B.* 1976. p. 163-172.
- [5] A. A. Goldsmith, D. Dowson, Development of a ten-station, multi-axis hip joint simulator, *Journal of Engineering in Medicine*, Vol. 213 Issue 4, p 311-316, 1999.
- [6] J. B. Medley, J. J. Krygier, J. D. Bobyn, F. W. Chan, A. Lippincott, M. Tanzer. Kinematics of the MATCO™ hip simulator and issues related to wear testing of metal-metal implants. *Proc. Instn. Mech. Engrs.* Vol. 211, Part H. 1995.
- [7] M. Viceconti, G. Cavallotti, A. O. Andrisano, A. Toni. Discussion on the design of a hip joint simulator. *Med. Eng. Phys.* Vol. 18, No. 3, p. 234-240, 1996.
- [8] J. A. Ortega Sáenz. Desarrollo de un simulador de cadera incluyendo microseparación para evaluación de prótesis totales de cadera. Tesis de maestría en ingeniería de materiales. Facultad de Ingeniería Mecánica y Eléctrica de la Universidad Autónoma de Nuevo León. 2007.

- [9] J. A. Ortega Sáenz, M. A. L. Hdz. Rdz., Diseño de un simulador de cadera con movimiento triaxial, carga variable y microseparación para realizar pruebas de desgaste en implantes quirúrgicos. Facultad de Ingeniería Mecánica y Eléctrica, Universidad Autónoma de Nuevo León, 2006.
- [10] J. A. Rangel Carrillo. Desarrollo de un sistema de control para un simulador de cadera mecánico. Tesis de maestría en ciencia y tecnología con opción terminal de diseño y desarrollo de sistema mecánicos. Corporación Mexicana de Investigación en Materiales. Saltillo, Coah. 2007.

Diseño de un vehículo eléctrico computarizado controlado a distancia mediante el estándar WIFI ® 802.11

Adalberto Bustamante López, Nataly Medina Rodríguez, Francisco Javier Díaz Delgadillo,
Arturo Sevilla Covarrubias,
CENTRO DE ENSEÑANZA TÉCNICA Y SUPERIOR

Campus Tijuana

Av. Grandes Lagos s/n Fracc. El Lago

Tijuana, Baja California, México

Teléfono: (664) 903 1800

Resumen: El presente documento describe el proceso de diseño de los sistemas mecánico, de control y de comunicación de un vehículo controlado a distancia por computadora, mediante el estándar WI-FI ® 802.11. El diseño mecánico del prototipo se concentra principalmente en el diseño de una estructura capaz de soportar impactos hasta $2.482 \times 10^8 \frac{N}{m^2}$ de esfuerzo. La simulación del prototipo estructural ha sido realizado COSMOSXpress, una herramienta para diseño en 3D de SolidWorks®.

El sistema de comunicación es una implementación embebida del protocolo TCP/IP sobre un microcontrolador ATmega128, el cual contiene una interfaz paralela con una tarjeta CompactFlash ® 802.11b/g. El sistema de comunicación envía los paquetes recibidos al microcontrolador central, el cual procesa en tiempo real los comandos recibidos por el módulo de comunicación y envía señales PWM de control al sistema de potencia, que a su vez regulan la energía suministrada a los motores de DC.

I. INTRODUCCIÓN

El proyecto XRunner es un prototipo de vehículo diseñado para resistir impactos y poseer una gran versatilidad para desarrollar movimientos de evasión y ataque frontal. Fue construido para concursar en el evento Sony Robotika 2007 en la región de Tijuana, B.C. El concurso culminaba en un enfrentamiento de robots de combate, sin embargo su finalidad consistía en someter a los equipos a un arduo proceso de diseño ingenieril, desde la investigación hasta el desarrollo del mismo prototipo. Aun cuando el diseño mecánico y electromecánico del proyecto XRunner fue enfocado a este evento (esto es, un robot de combate), la finalidad que nuestro equipo perseguía fue la crear una plataforma de investigación y desarrollo para todo tipo de pruebas mecánicas, electromecánicas, electrónicas y de software a lo largo de nuestras carreras de ingeniería.

El prototipo se creó en un periodo de aproximadamente cuatro meses. Se logró construir una estructura reforzada con un peso aproximadamente de 35 lb, dando un peso total en el vehículo de 120 lb, incluyendo todos sus demás accesorios. Se optó por utilizar tubular de acero galvanizado y se planeó un diseño triangulado enfocándose en la protección de las ruedas, la

distribución de la fuerza en colisiones frontales, y la altura necesaria para que el vehículo pudiera ser reversible. Todos estos aspectos fueron logrados exitosamente.

El vehículo contiene tracción individual en las cuatro ruedas, destinando un motor de 350 Watt (aprox. 0.5 Caballos de Fuerza), con el fin de tener alrededor de dos caballos de fuerza totales. Por otra parte, XRunner posee un mecanismo de control electrónico capaz de controlar independientemente dos ejes de su movimiento: eje lateral izquierdo y eje lateral derecho. Además de ello, cada eje puede ser regulado en cuanto a velocidad y dirección, además de un frenado electrónico de emergencia. Esto permite que la maniobrabilidad del vehículo sea versátil, puesto que puede efectuar movimientos regulares hacia delante y hacia atrás, pero además puede efectuar vueltas sobre su centro de masa.

El vehículo XRunner cuenta con un sistema de comunicación inalámbrica WI-FI®, o también conocida como IEEE 802.11 b/g, por lo cual puede ser controlada con cualquier computadora personal conectada a un *Access Point* mediante un programa de aplicación. El módulo de comunicación se comporta de manera deseable bajo condiciones regulares, sin embargo, dentro de las pruebas tuvo algunas pérdidas de conexión debido a interferencias y a la vibración excesiva. A continuación se muestra la implementación final del vehículo.



Fig. 1 Diseño final del proyecto XRunner.

II. DISEÑO MECÁNICO

El diseño mecánico del vehículo es uno de los aspectos más importantes para lograr un desempeño

exitoso durante las pruebas de impacto a las que se sometería el dispositivo. El plano estructural de XRunner fue diseñado utilizando herramientas de modelación CAD en 3D, específicamente SolidWorks® por sus ventajas superiores. Algunos puntos a favor radican en la posibilidad de simular los prototipos estructurales para poder apreciar su resistencia a la aplicación de fuerzas. Por otra parte, se pudieron lograr adaptaciones y cambios en el diseño mecánico de manera ágil y sin dificultades, lo que permitió comparar diseños preliminares antes de definir el final.

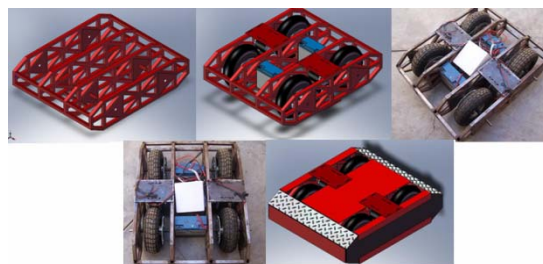


Fig. 2 Comparación del Diseño mecánico CAD de XRunner y el diseño real

La estructura principal de XRunner tiene el siguiente diseño:



Fig. 3 Estructura principal

El diseño básico consiste en tres elementos que soportan los impactos primarios, dos de los ejes distribuyen las fuerzas hacia los costados de la estructura, mientras el elemento principal que se encuentra exactamente entre estos dos elementos soporta la mayor parte de las fuerzas.

En la Fig. 4 se observa una fuerza aplicada a la estructura de aproximadamente $2.482 \times 10^8 \frac{N}{m^2}$. Se puede apreciar que los elementos de los costados de la estructura se encuentran en compresión, por lo que la estructura no es dañada drásticamente, lo cual es característica principal del prototipo: proteger los componentes que se encuentran dentro del mismo, el cual incluye tarjetas con

microcontroladores y controladores de motores.

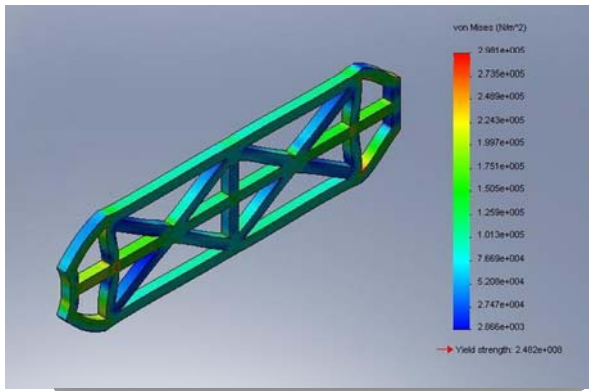


Fig. 4 Stress Distribution Analysis de CosmosWorks®

Llantas

XRunner utiliza llantas de 10.5" de diámetro, ya que el diseño consiste en lograr la mayor tracción posible y la reversibilidad del vehículo.



Fig. 5 Comparación entre el diseño CAD de las llantas y su sistema de engranaje, con el utilizado en el prototipo.

Transmisión

La transmisión mecánica está basada en un sistema de cadenas y sprockets¹, adaptando las llantas mediante una pieza maquinada para lograr la sujeción óptima con el sprocket.

¹ Sprocket – Rueda dentada

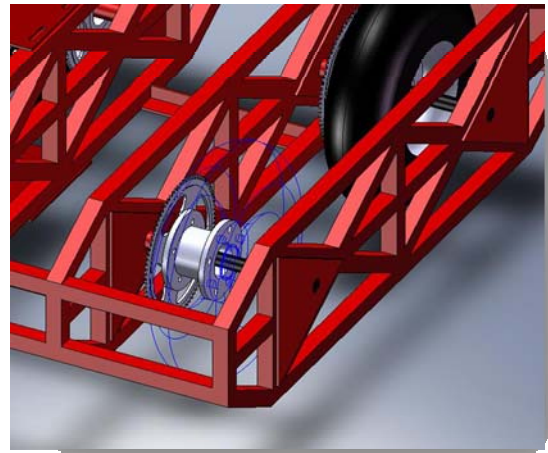


Fig. 6 Diseño de la Transmisión Mecánica

Las especificaciones de los motores utilizados en el prototipo son las siguientes:

Voltaje	24V
Corriente	18.5A
Velocidad	2650 RPM
Potencia	350 W
Sprocket diámetro	1"

Fig. 7 Especificaciones eléctricas de los motores, para la construcción de la transmisión de XRunner.



Fig. 8 Comparación entre el diseño CAD de la transmisión y el sistema real de XRunner

Se diseñó la transmisión para que esta tuviera velocidades de 9MPH, por lo que los cuatro motores utilizados, trabajando a $\frac{3}{4}$ de su capacidad ($2650 \times 0.75 = 1985RPM$), necesitan una proporción 1:6 de sistema de engranaje para las llantas de 10".

Considerando x como las revoluciones a las que se pretende hacer la reducción, y fijando un valor aproximado de $x \approx 300RPM$ después de la reducción, se obtiene que la relación r en la transmisión es

$$r = \frac{1985RPM}{300RPM} = 6.6$$

De lo cual, $6 \frac{1}{2}$ es la reducción de engranaje seleccionada por medio de los datos anteriores, ya que 6.6 no es una medida estándar.

El cálculo utilizado para determinar la velocidad, en millas por hora, es el siguiente:

$$v_{MPH} = \frac{Tire_Diameter \cdot Axle_Shaft_Speed}{336} \quad (1)$$

$$Tire_Diameter = 10in$$

$$Axle_Shaft_Speed = \frac{(2650RPM)(0.75)}{6.5} = 305.76RPM$$

$$v_{MPH} = \frac{(10in)(305.76RPM)}{336} = 9.1MPH$$

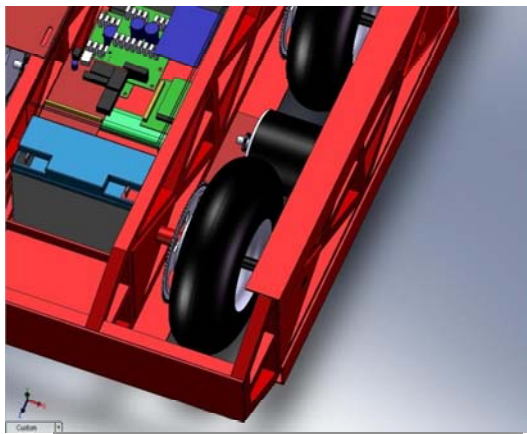


Fig. 9 Transmisión junto con ensamblaje de las placas laterales y visualización del ensamblaje de componentes electrónicos.

El sistema electrónico fue colocado en el centro de la estructura, cubierto con un material elástico para amortiguar impactos. El diseño se muestra en la figura 10.

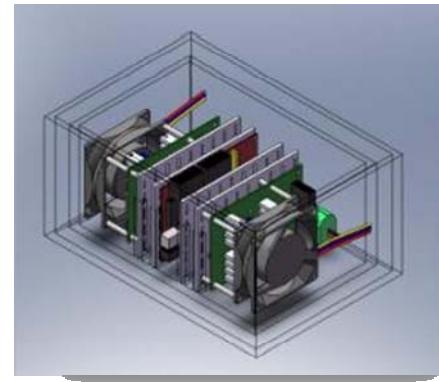


Fig. 10 Módulos de hardware electrónico. Los abanicos cubren los controladores de potencia y el microcontrolador y el módulo WI-FI se encuentran en la parte central. Las baterías se muestran al costado.

III. DISEÑO ELECTRÓNICO.

Es deseable que un sistema electrónico sea compacto, seguro y versátil. XRunner contiene en su mayoría módulos de propósito general los cuales han sido programados para desempeñar funciones deseadas; además de que el proyecto utiliza sólo tecnología de estado sólido, como EEPROMs y FLASH, aumentando la garantía de su desempeño en situaciones adversas. El diseño electrónico del vehículo consiste en los siguientes módulos:

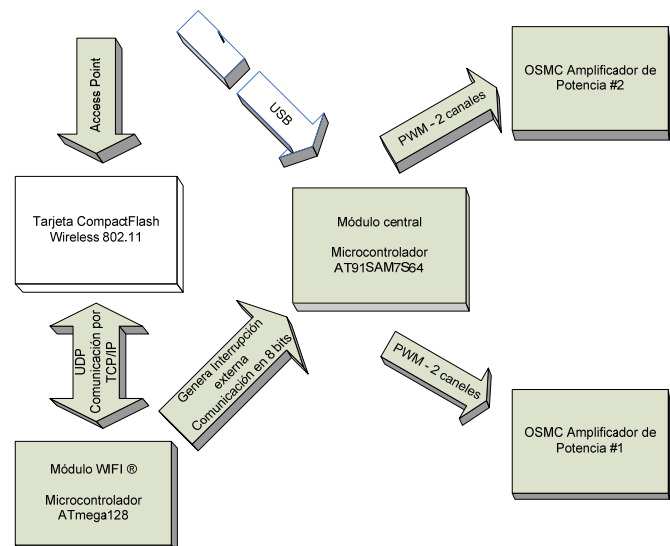


Fig. 11 Organización de los módulos electrónicos

A) Controlador central

El módulo principal es un microcontrolador Atmel AT91SAM7S64 de 32bits. El

microcontrolador central se encarga de recibir los datos del módulo WI-FI mediante una comunicación asincrónica de 8 bits unidireccional (recepción), además de los datos provenientes del USB, y es encargado de enviar señales de tipo PWM (modulación de ancho de pulso) a los controladores de potencia en cuatro canales:

- Canal 1: Regulación de la velocidad en dirección frontal en los motores de la lateral derecha.
- Canal 2: Regulación de la velocidad en dirección de reversa en los motores de la lateral derecha.
- Canal 3: Regulación de la velocidad en dirección frontal en los motores de la lateral izquierda.
- Canal 4: Regulación de la velocidad en dirección de reversa en los motores de la lateral izquierda.

Los controladores de potencia son dos controladores Puente-H de 50V a una corriente máxima de 160A, de tecnología MOSFET, los cuales reciben señales con el ciclo de trabajo y mediante un circuito integrado HIP4081 regulan la potencia y el sentido de la corriente que es enviada a los motores.

B) Módulo WI-FI

El sistema de comunicación por WI-FI® contiene un microcontrolador ATmega128 dedicado al procesamiento y comunicación con una tarjeta CompactFlash®, proporcionándole a la misma, las rutinas de inicialización y extracción de datos a través del protocolos diversos, que son descritos en el siguiente apartado.

IV. SOFTWARE

El diseño del software es una pieza para lograr un gran nivel de funcionalidad. Hubo la necesidad de usar de manera conjunta la programación de bajo y alto nivel para desarrollar los programas tanto de los microcontroladores como de la aplicación de PC.

Dentro de la programación del hardware embebido se optó por utilizar lenguaje C, debido a

la diversidad de aplicaciones desarrolladas en este lenguaje, además de su poder y eficiencia en dispositivos de características más limitadas. Por otra parte, se utilizó C++ para desarrollar el controlador de USB para la PC, así como C# para el programa de aplicación gráfico.

Dentro del software embebido se encuentran los siguientes componentes:

- Librería en C para el control de motores mediante señales PWM.
- Librería extensa de manejo de interrupciones externas (comunicación con el módulo de WI-FI®), interrupciones de timers, controlador de dispositivo USB, etc.
- Rutinas y macros para la ejecución de funciones de control.

Sin embargo, el software contenido dentro del vehículo se encuentra dividido en dos partes lógicas: el módulo central que se encarga de recibir interrupciones externas, interpretarlas y ejecutar comandos, asimismo de efectuar movimiento debido a la manipulación de señales PWM; y, el módulo de red encargado de manejar distintos protocolos de comunicación como TCP, IP, UDP, ICMP y ARP.

A. Módulo de red

El desarrollo del módulo WI-FI está basado en el Aidrop A® diseñado por Fred Eady [1]. El dispositivo CompactFlash es utilizado para cubrir la capa física del modelo OSI, es decir, es el encargado de enviar las señales de radiofrecuencia, asimismo como recibir datos y colocarlos en un búfer temporal en espera de ser recolectados por el microcontrolador. El dispositivo CompactFlash está basado en la tecnología PRISM desarrollada por Conexant® y frecuentemente utilizado por otros dispositivos inalámbricos Wi-Fi que funcionan bajo CompactFlash o USB. PRISM ofrece un poco más que solo proveer la transmisión de datos, ya que permite la configuración de la tarjeta, simulación del protocolo IEEE 802.3 (Ethernet) y la configuración de otros datos como la dirección IP así como la obtención de la dirección MAC del mismo dispositivo. Esta es otra de las razones de la elección del dispositivo CompactFlash ya que trae consigo una dirección MAC única, legal y validada. Sin embargo, a pesar de ofrecer facilidades para

manejar la capa de datos con la simulación de IEEE 802.3 (ya que es un protocolo más sencillo que 802.11), el microcontrolador se debe encargar de todos los aspectos relacionados con el protocolo de Ethernet, como la especificación de la cabecera, la dirección destino y origen, el cálculo de la suma de verificación, entre otros.

El firmware del vehículo incluye la implementación del protocolo TCP para el intercambio de instrucciones. La elección de TCP se debe a que permite la transmisión de datos de una manera confiable, algo que otros protocolos que a pesar de menos exigentes y más eficientes, como UDP, no pueden ofrecer.

Las principales razones de la modularidad de la electrónica y el software del vehículo fueron en un principio debido a la capacidad de procesamiento de los mismos módulos. El procesamiento de los mensajes posterior a su transmisión es un proceso tardado y caro en términos de memoria pues se requiere calcular la suma de verificación, transmitir datos desde el búfer del procesador a la tarjeta PRISM y construir el mensaje según los protocolos de red. Sin embargo la modularidad que se elaboró permitió la encapsulación y la abstracción de la comunicación de la que es capaz el vehículo, ya que el módulo central no conoce nada de protocolos de redes ni el módulo de red conoce de señales PWM. Debido a esto el canal de comunicación entre ambos módulos es muy simple.

Otro beneficio que permitió la modularidad del proyecto es la incorporación de un controlador de USB, esta variante permitió realizar pruebas sobre XRunner sin activar sus capacidades de comunicación inalámbrica.

B. Aplicación de PC

Dentro del software de aplicación para PC (alto nivel), se puede encontrar una aplicación que utiliza dos librerías de comunicación:

- Driver de comunicación inalámbrica basada en el protocolo TCP/IP
- Driver de comunicación mediante USB

Además, la aplicación utiliza un ambiente gráfico para realizar pruebas de conectividad, registro de comandos enviados, y esquema gráfico para reconocer el estatus del vehículo. Además, la

aplicación reconoce el uso de un Gamepad para controlar el vehículo de manera sencilla.

La mayoría del diseño de software presente en el sistema XRunner utiliza la técnica de *divide y conquistarás*. Dicho procedimiento se puede observar a través del análisis del siguiente algoritmo, empleado para realizar la aplicación de PC.

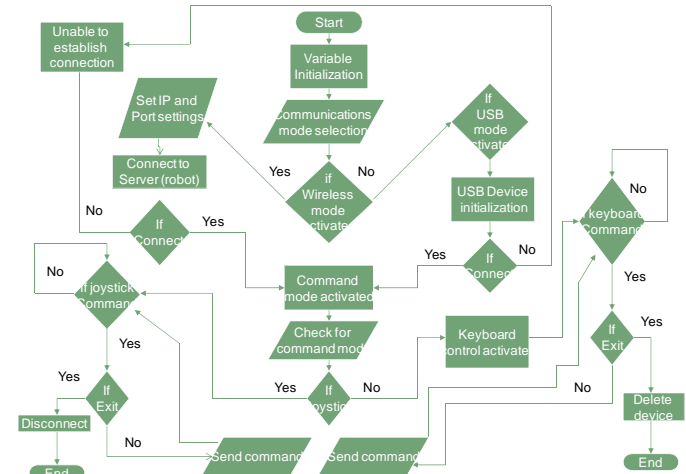


Fig. 12 Algoritmo de comunicación

C. Proceso de desarrollo

Existen cuatro principios sobre los cuales el desarrollo del software del sistema completo de XRunner se encuentra basado:

- Concurrencia
- Manejo de eventos
- Distribución de componentes (modularidad).
- Tolerancia de errores y manipulación

Debido a que durante el movimiento de un vehículo computarizado muchos eventos pueden ocurrir al mismo tiempo es necesario tener clara la noción de concurrencia y como evitar los problemas inherentes. El software está planeado para que estos errores solo ocurran en la capa de comunicación entre los distintos módulos, es decir en el manejo de eventos.

Debido a la modularidad del proyecto fue necesario implementar la comunicación entre los módulos. El modelo que se siguió fue de la programación orientada a eventos, dicho modelo fue implementado a través de interrupciones de hardware y cuidando que las operaciones realizadas durante las interrupciones no afectara el contexto actual del procesador. Este método evita errores con

la transferencia de datos, ya que los procesadores tienen relojes con distintas frecuencias.

A pesar de que los errores en software deben mantenerse a lo mínimo, algunos son inevitables pero responden a escenarios específicos. Sin embargo para evitar que los mismos escalen a todo el sistema, el manejo de errores se lleva dentro del mismo modulo afectado.

El algoritmo presente en el modulo de comunicación WI-FI® sigue las recomendaciones OSI [2] de distintas capas lógicas de comunicación sin perder un control óptimo, puesto que el software se diseñó para evitar casos de redundancia de operaciones que sólo debilitan el poder operacional del vehículo.

D. Capa Física

El vehículo XRunner utiliza la tecnología CompactFlash para implementar la capa física del modelo OSI. Esto permitió que el trabajo se enfocara en la lógica del funcionamiento más que en el alambrado de antenas, por ejemplo. Conforme a la especificación CompactFlash 3.0 [3] existen tres modos de operación: modo de memoria, I/O o TRUE IDE. El estándar PRISM [4] utiliza el modo de memoria para configurar el dispositivo, pero el modo I/O para efectuar operaciones de transmisión.

El modo de memoria es el default cuando la tarjeta se enciende, y la primera tarea que el módulo de comunicación procede a realizar es la verificación del funcionamiento del dispositivo PRISM para posteriormente introducirse al modo I/O. En la configuración del dispositivo existen registros pero a su vez tres tipos de estructuras muy importantes: BAPs (Búfer Access Path), FIDs (Frame Identifier) y RIDs. Los BAPs son utilizados para conocer donde los mensajes recibidos de manera asíncrona se encuentran localizados en la memoria del dispositivo CompactFlash. De manera similar los FIDs proveen acceso a las cadenas de mensajes alocaadas de manera dinámica, por ejemplo para la transmisión de datos. Los RIDs son similares a los FIDs pero proveen de apuntadores a regiones de configuración de la tarjeta.

Otra estructura de datos muy importante es el CIS (Card Information Structure); este describe la funcionalidad básica de la tarjeta como el voltaje necesario, dirección de distintos registros e

idealmente la dirección MAC, sin embargo muchas tarjetas reportan incorrectamente este dato a través de esta estructura. XRunner obtiene la dirección MAC a través del uso de un RID.

Para conocer si existen mensajes recibidos aun no procesados existen los registros de eventos. Dichos registros también reportan el estado sobre otras operaciones como la transmisión de datos.

E. Capa de datos

La tarjeta PRISM provee otra técnica útil para administrar el flujo de datos: el procesador del módulo de comunicación nunca analiza si el mensaje proveniente es para el vehículo (en términos de dirección MAC, ya que el módulo sí verifica la dirección IP), este procesamiento lo realiza la tarjeta PRISM antes de haber notificado al procesador del modulo, de manera que actúa como un filtro del espectro 2.4 GHz.

Es en esta misma capa donde se escoge bajo qué modo debe funcionar la tarjeta (manejado o ad-hoc), la tasa de transmisión y el SSID. Actualmente el firmware los implementa como datos definidos por el procesador de C.

Sin embargo la característica más importante que la tarjeta PRISM puede ofrecer a la construcción del firmware es la simulación del protocolo 802.3. A continuación se muestra una pequeña muestra de código que activa dicha funcionalidad utilizando tanto un búfer localizado en BAP que posteriormente se carga en un RID.

```
comframe_búfer[12] = 0x04; /* use
802.3 hdr style */
    comframe_búfer[13] = 0;
    #if USE802_11HDR == 1
    comframe_búfer[14] = 0x08;
    comframe_búfer[15] = 0x01;
    #else
    for (i = 0; i < 6; i++)

    comframe_búfer[TXDESTADDDROFFSET+i
] = macaddr[i];
    #endif
```

F. Capa de red

El firmware de XRunner implementa los siguientes protocolos: IP, y parcialmente ARP e ICMP. El modulo de comunicación nunca ejecuta una

solicitud de ARP, sin embargo construye una tabla que asocia direcciones IP y MAC similares a las entradas de ARP en un sistema operativo moderno. El comando ECHO de ICMP es la única parte del protocolo del mismo protocolo implementado en el módulo, ya que permite monitorear la conectividad del vehículo.

G. Capa de transmisión

Como se ha mencionado con anterioridad, XRunner implementa el protocolo TCP y UDP. Sin embargo para evitar perder procesamiento en la construcción del mensaje cuando se utiliza UDP la suma de verificación no se calcula, dicho comportamiento es completamente legal con las especificaciones del protocolo.

La capa de transmisión sigue el modelo cliente-servidor, donde el servidor corresponde al vehículo y el cliente a la aplicación que lo controla. El número de conexiones posibles depende de la cantidad de memoria disponible por el módulo de comunicación. Es importante hacer notar que debido a esta característica el vehículo puede ser controlado por más de un solo cliente (cuando el número de conexiones sea obviamente mayor a uno).

H. Capa de aplicación

Debido a que el trabajo más importante durante la implementación de los protocolos TCP e IP es la transmisión de comandos, se desarrolló un protocolo de instrucciones. Las instrucciones son empaquetadas en conjuntos de dos bytes. La estructura del paquete se muestra a continuación.

4 bit	4	4	0-4
Identificador del Protocolo XRunner	Motor izquierdo	Motor derecho	Comandos de control

Fig. 13 Tabla de especificación de un paquete de comandos.

La sintaxis del protocolo de comunicación desarrollado para la comunicación entre el vehículo y la aplicación se basa en el principio de *identificadores mágicos* o constantes que expresan la validez del formato, aún cuando éste sea implementado sobre TCP. Algunos otros ejemplos

de protocolos con este estilo son HTTP [5], del cual las respuestas del servidor comienzan con el identificador HTTP/versión; e IP [6], donde la versión del protocolo antecede al resto del datagrama en los bits 0 al 3 de la cabecera.

I. Pruebas de funcionamiento del módulo de red

Las pruebas pertinentes se realizaron utilizando en un principio un arreglo de LEDs para indicar que existía comunicación entre el controlador de la tarjeta de red y una aplicación de prueba. Dicho programa prueba se desarrolló en C bajo GNU/Linux y utilizando la librería de BSD sockets común en todos los sistemas UNIX (y parecidos).

Para comprobar el funcionamiento del sistema de comunicación se utilizó Wireshark, un analizador de paquetes. El propósito de usar dicho software era analizar que el paquete estuviera correctamente formado conforme al protocolo de instrucciones de XRunner que el equipo desarrolló y conforme a los protocolos de comunicación como TCP, IP, IEEE 802.11. En un principio la mayor cantidad de errores residían en la generación de la suma de verificación ya que su alineación era de 32 bits y debía ser transmitida hacia la tarjeta PRISM en formato Big Endian; pero el procesador tiene registros de 16 bits y trabaja con el formato Little Endian, lo que provocó numerosos cambios en la producción de dicha suma de verificación.

Durante la etapa final del desarrollo del firmware del módulo de comunicación, se realizaron muchas pruebas en el manejo de un número excesivo de paquetes. Debido a que el software contenido en dicho componente incluye la implementación para el comando ECHO del protocolo ICMP, las pruebas se realizaron ejecutando simples “pings” desde otra máquina. Para realizar las pruebas se ejecutó el comando ‘ping -f 192.168.1.100’ en una terminal de un sistema operativo GNU/Linux (sustituyendo la dirección IP por la apropiada en cada prueba); este comando “inunda” la red de transmisión con comandos ECHO. El comando era repetido de manera concurrente; al hacer pruebas con otros sistemas, como máquinas portátiles, el tiempo de respuesta llegó a demorarse hasta cinco segundos. Bajo la misma situación el módulo efectúa un reinicio del firmware debido a que la tarjeta PRISM no es capaz de manejar y almacenar tantos mensajes al mismo tiempo; esta es una de las limitaciones del

sistema que debido a que el tiempo estipulado para concluir el proyecto fue limitado este problema no fue solucionado. Sin embargo este problema no afecta seriamente ya que solo ocurre bajo premeditación de la inundación de la red de transmisión: ni siquiera durante un alto, pero normal, tráfico de mensajes se manifiesta el error.

Una vez que el componente de red fue instalado en el vehículo, las pruebas de velocidad y manejo se desarrollaron usando tanto el modulo USB (orientando las pruebas al manejo y velocidad) y el de red (orientando las pruebas a la conectividad y la velocidad). Al realizar dichas pruebas se comprobó que el alcance del vehículo consistía con el provisto por el *Access Point* (Linksys WRT54G).

J. Resultados del módulo de red

Se pudo constatar que el protocolo utilizado experimentaba una cierta limitación si se desearía incluir mas 255 opciones de control. Sin embargo, debido a la simplicidad de los comandos implementados se pudo comprobar que fue un modelo bien diseñado en acorde a los objetivos buscados. De igual manera debido a la gran opción de asignaciones libres en el protocolo, durante la fase de pruebas fue muy simple extender de manera temporal los comandos disponibles para realizar pruebas de configuración o instrucciones experimentales, así como el rechazo de comandos no reconocidos.

De acuerdo a la prueba de inundación de paquetes se pudo analizar que existe una vulnerabilidad en cuanto a que el modulo no es capaz de manejar una cantidad exagerada de paquetes ECHO del protocolo ICMP. Sin embargo, esto parece ser una debilidad impresa no solo en los recursos utilizados por el proyecto XRunner, sino de todas las tarjetas PRISM basadas en PCMCIA, ya que se desarrollo la misma prueba a dos computadoras personales de tipo portátil: Una donde el componente de red era basado en PCMCIA (muy parecido al CompactFlash utilizado por el vehículo) y otra donde con Intel Centrino donde, según datos de Intel, la tarjeta de red se encuentra ligada de manera directa con el procesador. Los resultados arrojaron que la computadora portátil con PCMCIA como base del componente de red con alrededor de 15 sesiones de Shell en GNU/Linux ejecutando el comando ‘ping -f 192.168.1.106’ demoraba hasta 5

segundos en devolver respuesta al paquete ECHO; sin embargo, la portátil con tecnología Centrino no se veía afectada de manera muy notoria como se puede apreciar en la siguiente muestra de los datos que seguidamente se proporcionan:

```

Respuesta al paquete ECHO de la computadora portátil con PCMCIA:
64 bytes from 192.168.1.101: icmp_seq=12 ttl=128 time=5535.5 ms
64 bytes from 192.168.1.101: icmp_seq=13 ttl=128 time=4985.23 ms
64 bytes from 192.168.1.101: icmp_seq=14 ttl=128 time=3019.33 ms
64 bytes from 192.168.1.101: icmp_seq=15 ttl=128 time=4014.43 ms
Respuesta al paquete ECHO de la computadora portatil con Centrino:
64 bytes from 192.168.1.101: icmp_seq=1 ttl=128 time=7.19 ms
64 bytes from 192.168.1.101: icmp_seq=2 ttl=128 time=18.2 ms
64 bytes from 192.168.1.101: icmp_seq=3 ttl=128 time=7.28 ms
64 bytes from 192.168.1.101: icmp_seq=4 ttl=128 time=14.5 ms

```

Es apreciable la diferencia entre ambos sistemas. De esta manera se puede concluir que a pesar de las limitaciones del modulo de red, la capacidad de transferencia entre el componente CompactFlash y el controlador, la capacidad de los búferes (BAPs, de acuerdo a la terminología de CompactFlash), la capacidad de procesamiento de los controladores y así como la implementación de los protocolos, son responsables directos del comportamiento que se presentó durante las pruebas del vehículo.

V. IMPLEMENTACIÓN E INTEGRACIÓN DE LOS SISTEMAS

Durante la etapa de implementación surgieron problemas inevitables referentes al software, además de fallas en la intercomunicación de los módulos electrónicos, y pérdidas de potencia por el sistema eléctrico. Los errores más propensos en el proyecto XRunner se debieron a la comunicación entre la tarjeta wireless CompactFlash® y su respectivo microcontrolador ATmega128, además de la comunicación de éste modulo con el módulo central.

Cabe considerar que el driver programado para controlar la tarjeta wireless funciona de manera adecuada con las tarjetas NETGEAR, pero no se puede garantizar que trabaje correctamente con otras marcas. Esto se debe a que cada tarjeta tiene pequeñas variaciones en su funcionamiento interno.

Otro aspecto considerable de implementación es la problemática de sincronización entre los microcontroladores, ya que manejan distintas frecuencias. Para ello se manejó el concepto de programación por interrupciones y se calculo el tiempo máximo de retardo que se necesita para que

el módulo central reciba los datos que el microcontrolador de WIFI®.

La integración de los módulos en el sistema electromecánico es otro aspecto considerable, puesto que el sistema puede responder de manera abrupta si no se utiliza la frecuencia adecuada en la señal enviada a los controladores de potencia. Para hacer las pruebas correspondientes, se probó el sistema electromecánico en su mínima expresión mediante el controlador USB como se aprecia en la siguiente imagen:



Fig. 14 Pruebas de funcionamiento del controlador de motores.

A continuación se muestra una vista de diseño de la construcción final del prototipo.



Fig. 15 Diseño final implementado

VI. CONCLUSIONES

La integración de diversas tecnologías, y en especial el desarrollo de las mismas, es un proceso arduo pero que permite una funcionalidad muy superior a los módulos ya comerciales, pues es

escalable. XRunner es un sistema de lazo abierto, por lo que no existe retroalimentación alguna para lograr un control automático, ya que se manipula unidireccionalmente por el estándar Wifi 802.11.

El uso de herramientas de desarrollo en 3D nos permitió realizar prototipos a los cuales se le pueden obtener diversos parámetros físicos, reduciendo drásticamente el tiempo, esfuerzo y dinero al no tener que crear diversos prototipos físicos. La herramienta utilizada en este proyecto está basada en un esquema matemático, en el cual, el uso de vectores permite el análisis de pruebas de esfuerzo y de impacto, análisis claves para nuestro proyecto, ya que es una característica importante en nuestro diseño.

Se pueden apreciar que algunas consideraciones básicas del diseño como la conservación de simetría, un centro de masa bajo y la protección de los circuitos electrónicos, han sido ventajosos en el desempeño exitoso de este dispositivo.

Por otra parte, la integración de los módulos electrónicos fue más ardua dado que no se diseñaron circuitos impresos, sino se conservaron los módulos de desarrollo convencionales.

XRunner busca ser un prototipo de investigación el cual pueda ser destinado para distintos fines, no sólo bélicos, sino como un vehículo buscador de minas, un dispositivo de manipulación de objetos en cámaras peligrosas para humanos, excavador de minas, etc.

ABREVIACIONES

EEPROM: Electronically Erasable Programmable Read Only
GCC: GNU Compiler Collection
PWM: Pulse-Width Modulation

REFERENCIAS

- [1] F. Eady, *Implementing 802.11 with Microcontrollers: Wireless Networking for Embedded Systems Designers*, Embedded Technology & Elsevier Inc, United States, 2005, pp. 105-107
- [2] A. S. Tanenbaum, *Computer Networks*, Pearson, New Jersey 2003.
- [3] CompactFlash Association, *CompactFlash. Specification. Revision 3.0.*, Palo Alto, CA, United States, December 2004. Available in rumkin.com/reference/aquapad/files/cfspec3_0.pdf
- [4] IDEAlliance, *PRISM 2.0 Specifications*, United States, 2001. Available in <http://www.primstandard.org/specifications/2.0/PRISM2.0Errata.zip>
- [5] Fielding, et al., *Hypertext Transfer Protocol -- HTTP/1.1*. RFC2616, Standards Track, June 1999, Available from <http://tools.ietf.org/html/rfc2616>
- [6] University of Southern California, "Internet Protocol Specification", DARPA Internet Program, California, United States, September 1981. Available from <http://tools.ietf.org/html/rfc791>

ROBOT DE POSICIONAMIENTO ANGULAR DE TRES GRADOS DE LIBERTAD CONTROLADO MEDIANTE CÁMARA DE VÍDEO

Azcona Guerrero Francisco, Vargas Mejía Héctor, Velázquez Guerrero Ramiro

Mecatrónica y Control de Sistemas - Universidad Panamericana,
Josemaría Escrivá de Balaguer 101, Fracc. Rústicos Calpulli,
20290, Aguascalientes, Ags., México

Tel: (449) 9106200, web: www.robotica-up.org, correo electrónico: javier.azcona@up.edu.mx

RESUMEN

En este artículo se propone el uso de una cámara de video para controlar la posición del efector final en un plano de movimiento de un brazo robótico de tres grados de libertad. La cámara detecta objetos de un color y forma particular así como su movimiento en el espacio para calcular, a través de métodos de cinemática inversa, los movimientos a efectuarse en cada articulación del robot. Como interfaz electrónica se propone el uso de un FPGA debido a su gran versatilidad para realizar cambios en su configuración de hardware interna.

Palabras clave: Robótica, cinemática directa, cinemática inversa, FPGA, procesamiento de imágenes.

I. INTRODUCCIÓN

Hoy en día es cada vez más común el uso de robots, máquinas y sistemas inteligentes para asistir al hombre en tareas repetitivas, tediosas y dañinas.

Existen un sinnúmero de definiciones para el término robot. Sin embargo, con el fin de hacer referencia al presente trabajo, se propone la definición de Schilling: *Un robot es un dispositivo mecánico controlable por software que utiliza sensores para guiar uno o más efectores a través de movimientos programados en un espacio de trabajo para manipular objetos físicos* [1].

A nivel industrial, desde hace varios años se utilizan los llamados brazos robóticos para el desarrollo de algunas tareas pero no ha sido sino

hasta fechas recientes que se comenzó a hacer un uso combinado de este tipo de manipuladores con técnicas de visión. La razón principal de esta integración son las ventajas que ofrece el uso de cámaras sobre los sensores activos: tecnología no-invasiva, percepción global del espacio, scanning no secuencial, velocidad de muestreo, costo y la posibilidad de trabajar en 3D [2]. Si bien es cierto que el uso de cámaras implica una mayor carga de procesamiento, también es cierto que la mayoría de los cálculos en el análisis de imágenes se pueden realizar en tiempo real con el uso de modernas computadoras y que en el campo de manipuladores, el procesamiento digital de imágenes puede ser bastante más rápido que la velocidad y desplazamiento de mecanismos y actuadores.

Existen distintos tipos de brazos robóticos clasificados esencialmente por el tipo de movimiento que pueden realizar. Las configuraciones clásicas son: cartesiano, cilíndrico, polar o esférico, angular o rotacional o antropomórfico. Además, existen otras configuraciones llamadas no clásicas, un ejemplo es el robot tipo SCARA. En la fig. 1 se muestran estas configuraciones [3].

En este artículo se presenta un brazo de tipo angular cuya representación conceptual se muestra en la fig. 2. Este brazo ha sido diseñado e implementado en la Universidad Panamericana con el fin de servir como plataforma a muy bajo costo para el estudio de la cinemática de manipuladores y la optimización de trayectorias de movimiento.

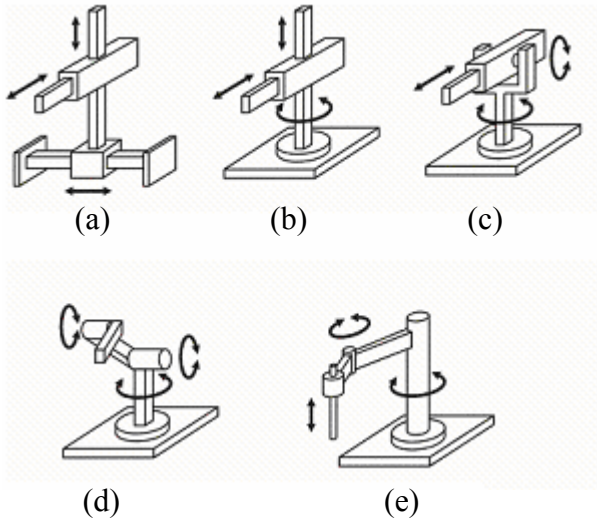


Fig. 1. Configuraciones clásicas de brazos robots: (a) cartesiano, (b) cilíndrico, (c) polar o esférico, (d) angular o antropomórfico y (e) SCARA.



Fig. 2. Representación conceptual de un brazo robótico de tipo angular.

Este brazo de tipo antropomórfico presenta 3 grados de libertad y es capaz de alcanzar una gran variedad de posiciones, dentro de su espacio de trabajo, en el plano xyz .

El resto del artículo está organizado de la siguiente manera: la Sección 2 presenta una revisión del problema de cinemática directa e inversa para el brazo de tipo angular y la implementación de un prototipo mecatrónico. La Sección 3 presenta los fundamentos del control de posicionamiento de dicho brazo basado en visión mono-estéreo. La sección 4 presenta la implementación de la interfaz electrónica del brazo robot mediante FPGAs. Finalmente, la Sección 5 presenta las conclusiones y perspectivas de trabajo futuro.

II. ANÁLISIS DE LA CINÉMÁTICA DEL SISTEMA

En el diseño y realización de un robot existen muchos detalles que deben ser contemplados pues se trata de un dispositivo que incluye tanto partes mecánicas como electrónicas y de software que deben trabajar en conjunto para realizar una tarea.

Debido a que la mecánica es quizás el tema más estudiado en términos generales, se comenzará por explicar los fundamentos del análisis necesario para la elaboración de un sistema de brazo robótico.

En términos generales el primer punto a analizar es el hecho de que un brazo robótico es un conjunto de eslabones unidos mediante articulaciones o juntas mecánicas, las cuales poseen un cierto número de grados de libertad (GDL). En esencia, un GDL es la libertad de movilidad de la cual dispone un objeto para desplazarse. Un objeto localizado únicamente en un plano tendrá tres GDL como máximo, mientras que a nivel espacial un objeto puede tener hasta seis GDL.

A nivel de mecanismos de lazo cerrado, el número de GDL está dado por el criterio Grübler-Kutzbach:

$$GDL = \lambda(n - 1 - j) + \sum_{i=0}^n F_i + \sum_{i=0}^n Fp_i \quad (1)$$

donde λ es el número máximo de GDL alcanzables, n el número de uniones en el mecanismo, j el número de articulaciones de un mecanismo, Fp el número de articulaciones redundantes y F el grado de movimiento relativo de cada articulación.

Para el caso de mecanismos de lazo abierto, se considera usualmente que la suma de GDL de sus articulaciones es el GDL del mecanismo.

Una vez que se tiene contemplada la cantidad de GDL del sistema, se realiza el análisis matemático de los movimientos que puede realizar el mecanismo, es decir, se realiza el análisis de cinemática directa, y aún más importante, el análisis de cinemática inversa. En otras palabras, modelar la cinemática de un sistema es describir matemáticamente la relación que guardan las coordenadas de las articulaciones con la relación de la posición y orientación del elemento terminal.

El análisis de cinemática directa resuelve las posiciones que puede alcanzar el sistema (coordenadas espaciales) a partir de los datos que se debe mover cada articulación. Esto es de utilidad si se sabe exactamente que ángulo necesita rotar una articulación, o en el caso de un movimiento lineal, si se conoce que sucede en el espacio cuando el sistema se desplaza una distancia dada. Típicamente el problema de cinemática directa se puede resolver de distintas maneras. Los métodos más usuales son:

- Método Geométrico
- Matrices de transformación homogénea
- Método Denavit-Hartenberg

Los tres métodos son igualmente eficientes para el modelado de todo tipo de problema en robótica. Sin embargo conforme el número de GDL aumenta, la dificultad se incrementa de igual manera en los dos primeros métodos. Por su simplicidad, se utiliza el método Denavit-Hartenberg para resolver los problemas más complejos.

Las ecuaciones (3) y (4) son las ecuaciones típicas de resolución por el método geométrico y el método de Danevit para un robot angular de tres GDL, respectivamente.

$$\begin{aligned} X &= L_2 \cos \theta_2 \cos \theta_1 + L_3 \cos(\theta_2 + \theta_3) \cos \theta_1 \\ Y &= L_2 \cos \theta_2 \sin \theta_1 + L_3 \cos(\theta_2 + \theta_3) \sin \theta_1 \\ Z &= L_1 + L_2 \sin \theta_2 \cos \theta_2 + L_3 \sin(\theta_2 + \theta_3) \end{aligned} \quad (3)$$

$${}^{n-1}T_n = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

donde:

X, Y, Z: Movimiento en cada uno de los ejes.
 L_i : Eslabón o pieza del robot
 θ_i : Ángulo entre los eslabones
 α_i : Ángulo perpendicular que existe entre los ejes
 d_i : Distancia entre eslabones
 a_i : Distancia entre los ejes de las articulaciones.

Como se puede intuir, el cálculo de la cinemática directa es relativamente sencillo pero al momento de analizar su funcionalidad en términos de operatividad, carece de una importancia visible.

La cinemática inversa se encarga de analizar el movimiento necesario de una articulación con el fin de alcanzar un punto en el plano cartesiano. En términos prácticos esto es más común pues usualmente se puede definir la posición espacial de un objeto de manera sencilla. Al igual que en el caso de cinemática directa este cálculo se puede realizar por distintos métodos. Las ecuaciones en (5) se utilizan en el caso de brazos robóticos de 3 GDL.

$$\begin{aligned} \theta_1 &= \tan^{-1}\left(\frac{Y}{X}\right) \\ \theta_2 &= \tan^{-1}\left(\frac{Z - L_1}{R}\right) + \tan^{-1}\left(\frac{\sqrt{1 - \cos B^2}}{\cos B}\right) \\ \theta_3 &= -\frac{\sqrt{1 - \cos C^2}}{\cos C} \end{aligned} \quad (5)$$

con:

$$\begin{aligned} R &= \sqrt{X^2 + Y^2} & \cos B &= \frac{L_2^2 + t^2 - L_3^2}{2L_2L_3} \\ t &= \sqrt{R^2 + Z^2} & \cos C &= \frac{t^2 - L_3^2 - L_2^2}{2L_2L_3} \end{aligned}$$

De manera similar se puede realizar el cálculo mediante matrices inversas por el método de Denavit-Hartenberg. Con dicho método se obtienen 9 ecuaciones para el cálculo de cada ángulo pudiendo resolver con cualquiera de ellas para el ángulo dado [4].

A - Prototipo

La fig. 3 muestra el prototipo realizado. Se trata de un brazo robótico de 3 GDL, que consta de una estructura de barras de aluminio, con una distancia entre ejes: $L_1=19.8$ cm, $L_2=11.3$ cm y $L_3=12$ cm, donde L_1 es la distancia de la base a la articulación del hombro, L_2 la distancia entre las articulaciones codo y hombro y L_3 la distancia entre la articulación del codo a la punta del mecanismo.

Cada articulación cuenta con distintos grados de movilidad: la cintura cuenta con 270° dividiendo su interpretación de -135° a 135° , el hombro cuenta con 60° dividiendo su interpretación de -25° a 45° y el codo cuenta con 170° dividiendo su interpretación de -75° a 95° , por lo que el área máxima de trabajo se denota por una semiesfera de radio 22.3cm.



Fig. 3. Prototipo de brazo robótico de 3 GDL.

III. CÁMARA DE VIDEO

Existen diversos métodos para detectar objetos que se encuentran en un área dada, desde sensores infrarrojos (los más básicos que existen) hasta dispositivos complejos a base de sonar o cámaras de video. Sin embargo no todos los sensores tienen las mismas capacidades en términos de detección, por lo cual a cada tipo de sensor se le deben aplicar distintos tipos de filtros para la detección de objetos [5].

El hecho es que si en realidad se desea detectar un objeto de manera similar a como lo hace el hombre, se requiere de un sensor capaz no sólo de recabar información en tiempo real (hay que recordar que incluso el ojo humano sólo es capaz de captar información en un rango de entre 0 y 50 cuadros por segundo aproximadamente, lo cual hace que movimientos mayores a estas frecuencias no sean captados en su totalidad); más aún, se requiere un sensor capaz de discernir cuando menos el color y la forma de un objeto en caso de que se presenten distintos objetos similares en el ambiente.

Una cámara de vídeo resulta una opción muy interesante para realizar el rastreo de objetos que posean un color y una forma definida. En nuestra aplicación el interés se sitúa en la detección y rastreo de esferas. Sin embargo con ligeras modificaciones este rastreo se puede expandir a otro tipo de objetos.

Dado que el interés de este artículo recae también en la identificación de colores y no sólo en la identificación de objetos, se decidió utilizar un sistema común de clasificación en RGB; también es posible utilizar un sistema HSI (sistema que es más similar en términos de visión a lo que realiza el ojo humano), pero la conversión a este tipo de sistema implica un mayor tiempo de procesamiento, que se prefirió utilizar en otro tipo de procesos.

Las etapas del procesamiento digital de imágenes para el control del brazo robótico son:

- Obtención de la imagen
- Detección del color
- Detección de forma
- Localización del centro de la forma en un plano dado

En términos de procesamiento la etapa más complicada es la detección de forma pues dependiendo del color existe una alta variación en lo que se puede detectar dependiendo de la cantidad de luz ambiental (iluminación). Por ejemplo, en el caso del color rojo como se observa en la imagen de la fig. 4, no sólo se detecta el color de la esfera, sino también partes de la piel, razón por la cual es necesario una siguiente etapa de detección de formas.



Fig. 4. Detección de color de un objeto. Note que el color del objeto puede confundirse con otros objetos.

Como se observa en la fig. 4, la esfera no se detecta de manera completa por lo que el detector de forma debe de tener la capacidad de detectar el objeto en relación a un porcentaje del objeto dado. Se implementó un algoritmo que utiliza un porcentaje de similitud de entre 85 y 90%, dependiendo del brillo o iluminación que se encuentre en la habitación (fig. 5). Además, una vez que se tienen aislados las distintas esferas, el algoritmo es capaz de discriminar el tamaño del objeto.

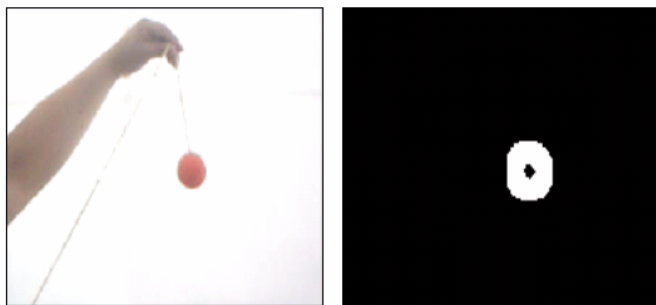


Fig. 5. Detección de un objeto (esfera) utilizando una relación de porcentaje de su forma.

Debido al rango de valores utilizados para la detección de la esfera, se observa que ya no es posible tomar directamente el objeto con la mano, pues distorsiona su forma al estar dentro del rango de valores detectados.

Una vez que se tiene perfectamente aislado el objeto se procede a la detección de su centro asignando un eje coordenado correspondiente para el movimiento (fig. 5). Por conveniencia, se seleccionó la vertical descendente como eje X positivo mientras que la horizontal representará el eje Y con secciones positivas y negativas respectivamente, esto con el fin de tener un movimiento sobre la base de cuando menos 180°.

Una vez que el centro se interpreta de acuerdo al nuevo sistema de coordenadas, se puede proceder a hacer un análisis de cinemática inversa y finalmente a hacer el acoplamiento con las interfaces electrónica y de control que en este caso se realizan con una tarjeta FPGA.

IV. INTERFAZ ELECTRÓNICA CON FPGA

Después de concluir el procesamiento de imágenes se debe implementar una interfaz entre el software y la parte mecánica. Con este fin se optó por explorar la tecnología de FPGAs.

La implementación en FPGA (Arreglo de Compuertas Programables en Campo) tiene la gran ventaja sobre los microcontroladores de ser capaz de analizar y ejecutar simultáneamente todas las variables de proyecto, además de incluir una gran variedad de puertos: USB, RS232 y la posibilidad de correlacionarlas para obtener un resultado óptimo.

El diagrama bloques de la interfaz electrónica con FPGAs se muestra en la fig. 6. Esta interfaz consiste principalmente de un sistema de adquisición de datos (DAS) el cual se encarga del procesamiento de los valores de retroalimentación de la cintura del brazo.

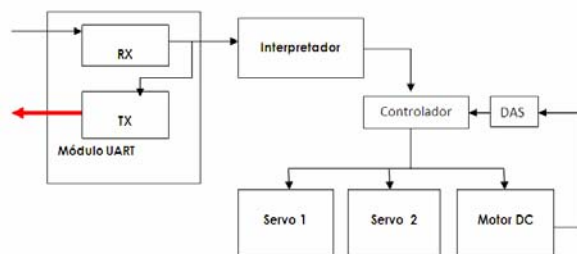


Fig. 6. Implementación de la interfaz electrónica con FPGAs.

V. CONCLUSIÓN

En este artículo se han presentado los principios de un método de detección y seguimiento de objetos diferente a las propuestas tradicionales en las cuales se utilizan sensores infrarrojos o de sonido. Encontrando que una cámara de vídeo contiene un alto grado de eficiencia para esta aplicación con la desventaja que se debe controlar la iluminación del medio, este método de visión también nos permite detectar formas por lo cual es factible su implementación en la industria en áreas como control de calidad.

Este sistema de visión mono-estéreo fue instalado en un brazo articulado de 3 GDL con el fin de controlar los movimientos de sus articulaciones.

La perspectiva de trabajo futuro es añadir 3 GDL más con el fin de tener un mayor alcance de posiciones en la esfera. También se espera añadir una pinza con el fin de manipular objetos dentro del campo de visión.

Este proyecto es una contribución al laboratorio de sistemas automáticos de la escuela de ingeniería de la Universidad Panamericana Campus Bonaterra y tiene como objetivo servir de plataforma para que los alumnos puedan seguir profundizando en sus conocimientos de Robótica y Visión al aplicarlos en un sistema real.

VI. REFERENCIAS

- [1] A. Barrientos y L. Peñin, “Fundamentos de Robótica”, McGraw-Hill, 1997.
- [2] R. Gonzalez y R. Woods, “Digital Image Processing”, Prentice Hall, 2001.
- [3] J. Craig, “Introduction to robotics: mechanics and control”, Addison Wesley, 1989.
- [4] K.S. Fu, R.C. González y C.S. Lee, “Robótica control, detección, visión e inteligencia”, McGraw-Hill 1989.
- [5] M. Groover y P. Mikell, “Industrial Robotics: Technology, programming and applications”, Mc Graw Hill, 1986.

Análisis modal y estructural de microgiroscopios de anillo vibrante mediante modelos de elementos finitos

A L Herrera May^{1,2}, M A Domínguez Salgado², E Morales González², L A Aguilera Cortés³

¹Centro de Investigaciones en Micro y Nanotecnología, Universidad Veracruzana, 94292, Boca del Río, Ver., México

²Facultad de Ingeniería, Universidad Veracruzana, 94292, Boca del Río, Ver., México.

³FIMEE Campus Salamanca, Universidad de Guanajuato, Carretera Salamanca-Valle de Santiago km 3.5+1.8, Salamanca, Gto., México

E-mail: leherrera@uv.mx; aguilera@salamanca.ugto.mx

RESUMEN

El análisis modal y estructural de dos microgiroscopios de anillo vibrante de polisilicio con alta razón de aspecto (20:1) es desarrollado mediante modelos tridimensionales de elementos finitos. La estructura de los microgiroscopios tiene anillos vibrantes con radios externos de 550 μm y 750 μm , una profundidad de 80 μm y un espesor de 4 μm . Estos anillos están conectados a una base rígida mediante ocho soportes semicirculares y son excitados en su primer modo flexionante de forma elíptica. Los cambios obtenidos en el desplazamiento del anillo vibrante son proporcionales a la velocidad de rotación. El máximo esfuerzo de Von Mises (5.6 MPa) en la estructura de los microgiroscopios se presenta en la conexión entre el anillo vibrante y los soportes semicirculares. Los microgiroscopios presentan una respuesta lineal para velocidades angulares de 62.5 $\%/\text{s}$ a 2000 $\%/\text{s}$ con un factor de calidad de 1200. La respuesta en la amplitud de la oscilación de los microgiroscopios es obtenida en un intervalo para el factor de calidad de 500 a 20000. Un desplazamiento de 483.9 nm es obtenido para una velocidad de rotación de 1000 $\%/\text{s}$ con un factor de calidad de 20000.

Palabras clave: Análisis modal, esfuerzo de Von Mises, método de elementos finitos, microgiroscopio.

I. INTRODUCCIÓN

El giroscopio es un dispositivo utilizado para la medición de la velocidad angular. Un giroscopio convencional posee una masa de prueba que vibra en la dirección de un eje y cuando una velocidad es aplicada, el giroscopio experimenta una segunda vibración en la dirección del otro eje. La magnitud de la segunda vibración es proporcional a la magnitud de la velocidad angular de entrada. Por ende, el esquema de sensado del giroscopio mide la magnitud de la velocidad angular de entrada considerando la magnitud de la segunda vibración.

Avances recientes en la tecnología de micromaquinado han permitido el diseño y fabricación de microgiroscopios. Estos dispositivos presentan un tamaño pequeño, mínimo consumo de potencia y bajo costo en producción en serie con respecto a los giroscopios convencionales [1]. Desde 1993, Draper and Rockwell International han comercializado microgiroscopios para aplicación automotriz [2]. Además, General Motors, Analog Devices y Samsung han realizado extensas investigaciones y desarrollo acerca de microgiroscopios [3].

Actualmente los microgiroscopios tienen un enorme potencial en aplicaciones de la electrónica de consumo, incluyendo video juegos, mouse inercial para computadoras, juguetes electrónicos,

estabilización de imágenes para video cámaras; aplicaciones automotrices en el control activo de la tracción y de la suspensión y en sistemas de seguridad del vehículo; en aplicaciones militares tales como el control de satélites y microaeroplanos [4-6]; en biomedicina en el desarrollo de prótesis vestibular [7] y en aplicaciones de robótica.

Los microgiroscopios de anillo vibrante tienen una masa sísmica que consiste de un anillo circular, el cual está conectado por soportes elásticos y utiliza electrodos para la excitación y detección [8]. La compañía Delphi Delco Electronics Systems fabricó un microgiroscopio de anillo vibrante con su parte central anclada al sustrato [9]. Este dispositivo presentó frecuencias degeneradas con modos resonantes desfasados 45° con respecto a cada uno. Ayazi y Najafi [10] desarrollaron un microgiroscopio de anillo vibrante de polisilicio con una alta razón de aspecto (20:1) y con electrodos para la detección capacitiva. El factor de calidad del microgiroscopio fue de 1200 y su rango dinámico de ± 250 %/s.

Los microgiroscopios de anillo vibrante ofrecen importantes ventajas con respecto a giroscopios convencionales basados en ruedas vibrantes tales como menor tamaño, menor costo, disminución del ruido y su estructura simétrica es menos sensible a vibraciones espurias. Estas vibraciones pueden ser inducidas por el medio externo únicamente cuando la geometría del anillo tiene asimetrías en su rigidez o masa [11]. La sensibilidad es incrementada al operar a bajas presiones (disminución del amortiguamiento del aire e incremento del factor de calidad). El anillo vibrante es menos sensible a cambios en la temperatura, porque los modos de vibración idénticos son afectados de forma similar por la temperatura. La sintonización electrónica de la estructura del microgiroscopio es posible. Cualquier error en el acoplamiento entre los modos de resonancia de excitación y sensado, originados por proceso de fabricación, puede ser electrónicamente compensado mediante la

utilización de electrodos de ajuste que son localizados alrededor de la estructura.

Este trabajo se reporta el estudio del comportamiento mecánico de microgiroscopios de anillo vibrante de polisilicio, mediante el desarrollo de modelos de elementos finitos en 3D con el software ANSYS. Las respuesta estructural de los microgiroscopios con respecto a la variaciones en la velocidad angular y el factor de calidad son analizados.

II. FUNCIONAMIENTO DEL MICROGIROSCOPIO

La estructura del microgiroscopio consiste de un anillo circular externo soportado mediante ocho elementos semicirculares conectados a una base rígida (ver figura 1). A una distancia fija del perímetro del anillo se colocan electrodos para la excitación electrostática y la detección de la variación de la capacitancia. Esta configuración garantiza que el microgiroscopio presente dos modos flexionantes de forma elíptica a la misma frecuencia y con un desfase de 45° entre ellos. El anillo empieza a vibrar en su primer modo flexionante, con una amplitud fija, cuando es excitado electrostáticamente. Un cambio en la velocidad de rotación provoca que la energía se transfiera del primer modo flexionante (modo de excitación) a su segundo modo flexionante (modo de sensado). La distancia entre el electrodo de sensado y la superficie de anillo es alterada, lo cual ocasiona una variación en la capacitancia.

La frecuencia de resonancia para los modos flexionantes circunferencial-radial de una membrana cilíndrica con un radio r y un espesor t es dado por [12]

$$f_i = \frac{i(i^2 - 1)t}{2\pi\sqrt{2}r^2\sqrt{i^2 + 1}} \sqrt{\frac{E}{\rho}} \quad (1)$$

donde E es el módulo de Young y ρ es la densidad del material de la membrana. Las frecuencias naturales de los modos flexionantes son independientes de la altura de la estructura de la

membrana. Mediante la Ec. (1), es posible calcular aproximadamente la frecuencia de resonancia flexionante de un giroscopio de anillo. Si $f_i = (1/2\pi)\sqrt{k_i/M}$ y la masa $M \propto r$, entonces la rigidez $k_i \propto (1/r^3)$. En el caso de un giroscopio en donde cada soporte semicircular tiene un radio aproximadamente igual a 0.43 veces el radio del anillo, entonces la rigidez flexionante de cada soporte es 6.4 veces superior a la del anillo sin soportes semicirculares (observe que dos soportes semicirculares forman un anillo completo, así la rigidez de cada soporte semicircular es la mitad de la rigidez de un anillo circular). Por lo tanto, la rigidez para el modo flexionante de un microgiroscopio de anillo con ocho soportes semicirculares será 7.4 veces mayor que la rigidez de un anillo sin soportes. Si la masa efectiva de un soporte semicircular es un tercio de su masa total, entonces la frecuencia de resonancia para el primer modo flexionante de un anillo puede ser estimado por ($i=2$ in Ec. (1)) [10]

$$f_g \cong \frac{\sqrt{3}t}{2\pi r^2} \sqrt{\frac{E}{\rho}} \quad (2)$$

La fuerza de Coriolis que actúa en el microgiroscopio de anillo vibrante (con dos modos de vibración flexionantes de forma elíptica a frecuencias similares) sujeta a una rotación angular alrededor de su eje normal está dada por [10]:

$$F_{Coriolis} = 4A_g \Omega_z M \omega_0 q_{drive} \quad (3)$$

donde $A_g \approx 0.37$ es la ganancia angular de la estructura tipo anillo (que depende de la geometría del sensor) [10], Ω_z es la velocidad de rotación, M es la masa de la estructura, ω_0 es la frecuencia angular de la estructura y q_{drive} es la amplitud de vibración en el modo de excitación, que depende de la magnitud del voltaje aplicado en los electrodos. Para los modelos de elementos finitos se considerará una amplitud de vibración de $0.15 \mu\text{m}$ [10].

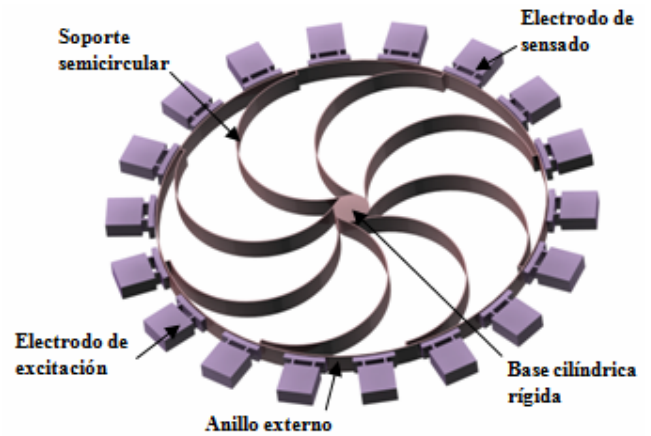


Figura 1. Configuración de un microgiroscopio de anillo vibrante.

III. MODELOS DE ELEMENTOS FINITOS

Modelos de elementos finitos en 3D de la estructura de dos microgiroscopios de anillo vibrante, con radios externos de $550 \mu\text{m}$ y $750 \mu\text{m}$, se realizaron con el software ANSYS (figura 2). Se utilizó elementos solid95, éstos tienen tres grados de libertad en cada nodo (desplazamientos en las direcciones x , y y z). Los estructuras tienen en común las siguientes dimensiones: profundidad de $80 \mu\text{m}$, base cilíndrica sólida con $120 \mu\text{m}$ de diámetro, espesor $t=4 \mu\text{m}$ para el anillo exterior y ocho placas rectangulares de $20 \times 80 \times 4 \mu\text{m}$ que conectan los soportes semicirculares con el anillo del microgiroscopio. La base cilíndrica fue restringida en su movimiento en las tres direcciones x , y y z .

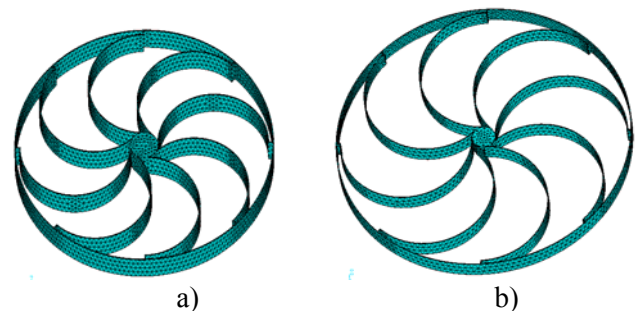


Figura 2. Mallado de la estructura de dos microgiroscopios de anillo vibrante con un radio externo de a) $550 \mu\text{m}$ y b) $750 \mu\text{m}$.

IV. ANÁLISIS Y DISCUSIÓN

En esta sección se presenta los resultados del comportamiento modal y estructural de las dos

estructuras de los microgiroscopios. El material de los microgiroscopios es polisilicio con un módulo de Young $E=150$ GPa y una densidad $\rho=2328$ kg/m³. Las primeras diez frecuencias de resonancia fueron obtenidas para cada modelo. La estructura de 550 μm de radio registró el primer modo de vibración torsional a 11124 Hz, como se muestra en la figura 3 (a). Sus primeros dos modos de vibración de traslación se presentaron a 20671 Hz y 20709 Hz, como se muestra en la figura 3 (b-c). Los primeros dos modos flexionantes de forma elíptica fueron registrados a 29793 Hz y 29844 Hz. Estos modos flexionantes están desfasados 45° entre ellos, como se muestra en la figura 3(d-e). La diferencia en el valor de las frecuencias entre estos modos degenerados flexionantes fue de 51 Hz.

Los resultados del análisis modal con el microgiroscopio de 750 μm de radio presentaron sus primeras frecuencias de resonancia con magnitudes menores en relación con el microgiroscopio de 550 μm . El primer modo torsional es obtenido a 5817 Hz, ver la figura 4(a). Los siguientes dos modos son de traslación (10604 Hz y 10648 Hz) como se muestran en la figura 4(b-c). Sus primeros dos modos flexionantes con forma elíptica (desfasados 45° uno con respecto al otro) se registraron a 15530 Hz y 15557 Hz [Figura 4(d-e)]. La diferencia entre los modos degenerados tipo flexionante es de sólo 27 Hz. Las frecuencias de resonancia de estos modos degenerados fueron comparadas con los obtenidos con la Ec. (2), ver la tabla 1.

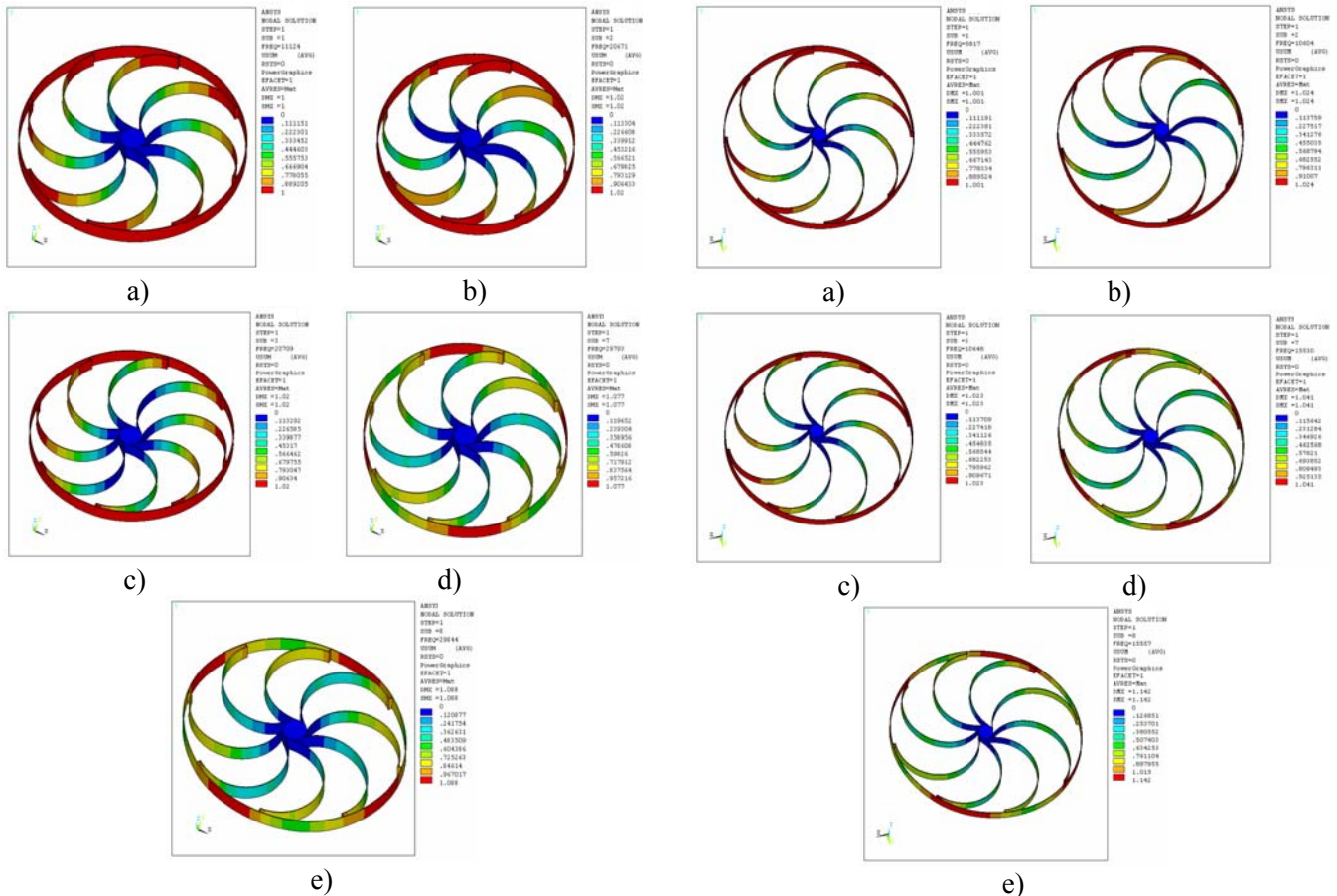


Figura 3. Modos de vibración de un microgiroscopio de anillo vibrante de 550 μm de radio: a) 1er modo torsional; b) 1er y c) 2do modo de traslación; d) 1er y e) 2do modo flexionante de forma elíptica.

Figura 4. Modos de vibración de un microgiroscopio de anillo vibrante de 750 μm de radio: a) 1er modo torsional; b) 1er y c) 2do modo de traslación; d) 1er y e) 2do modo flexionante de forma elíptica.

Tabla 1. Comparación de la frecuencia del primer modo flexionante de forma elíptica de un microgiroscopio de anillo vibrante.

Modelo	Radio del anillo	
	550 μm	750 μm
	Frecuencia (Hz)	
Elementos finitos	29793	15530
Analítico	29259.7	15735.2
% Diferencia relativa	1.8	1.3

En los dos modelos de elementos finitos se consideró un factor de calidad $Q=1200$ y una amplitud inicial de vibración de $0.15 \mu\text{m}$ [8] en el primer modo flexionante de forma elíptica. Mediante la Ec. (3) se obtiene la fuerza de Coriolis para una razón de rotación de $62.5 \text{ }^\circ/\text{s}$ hasta $2000 \text{ }^\circ/\text{s}$. El valor de la fuerza de Coriolis disminuye conforme se incrementa la magnitud del radio del anillo, porque la frecuencia de resonancia decae. Esta fuerza se aplicó en la estructura del microgiroscopio cuando opera en su modo degenerado flexionante. Posteriormente se encontró la respuesta del desplazamiento de los microgiroscopios como resultado de la fuerza de Coriolis ocasionadas por cambios en la velocidad de rotación. El valor del desplazamiento fue obtenido en el extremo de la elipse que se forma en el segundo modo flexionante. Los resultados indican un comportamiento lineal del desplazamiento con respecto a la razón angular (ver la figura 5). Este desplazamiento se incrementa conforme el radio del anillo vibrante aumenta de $550 \mu\text{m}$ a $750 \mu\text{m}$. El valor máximo de desplazamiento de 118.4 nm se obtuvo a $2000 \text{ }^\circ/\text{s}$ para la estructura de $750 \mu\text{m}$, el cual es 1.9 veces mayor que el obtenido en la estructura de $550 \mu\text{m}$.

Los esfuerzos máximos de von Mises fueron obtenidos en las dos estructuras de los microgiroscopios. Estos resultados tienen un incremento lineal con respecto a la velocidad de rotación (ver la figura 6). El microgiroscopio de $750 \mu\text{m}$ registró el máximo esfuerzo de 847.2 kPa con una velocidad de rotación de $2000 \text{ }^\circ/\text{s}$, el cual es 1.39 veces superior al obtenido en el de 550

μm . En la figura 7 se muestra la distribución de los esfuerzos de von Mises en el microgiroscopio de menor tamaño a una velocidad de rotación de $1500 \text{ }^\circ/\text{s}$ y un factor de calidad de $Q=1200$. Los esfuerzos máximos se presentan en la conexión del anillo externo y los soportes semicirculares. Este esfuerzo es menor al esfuerzo de ruptura del polisilicio.

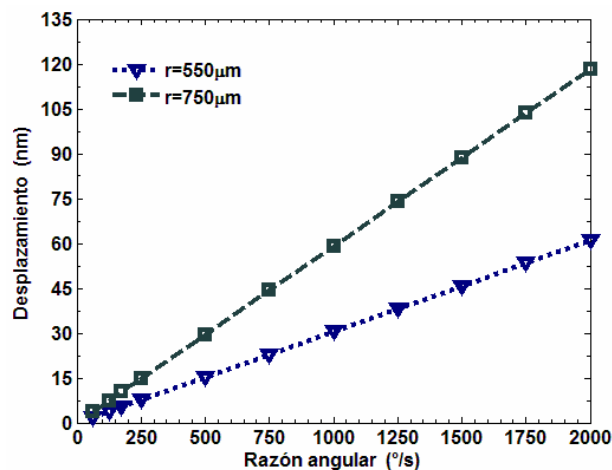


Figura 5. Desplazamiento (nm) vs velocidad de rotación para dos microgiroscopios de anillo vibrante.

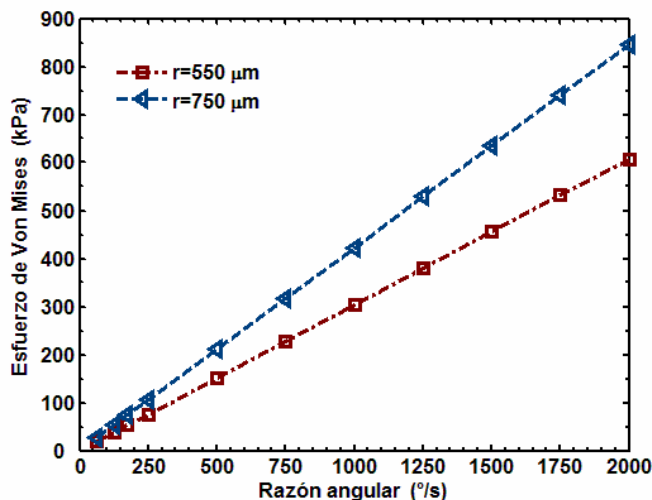


Figura 6. Esfuerzos máximos de Von Mises (nm) vs velocidad de rotación para dos microgiroscopios de anillo vibrante.

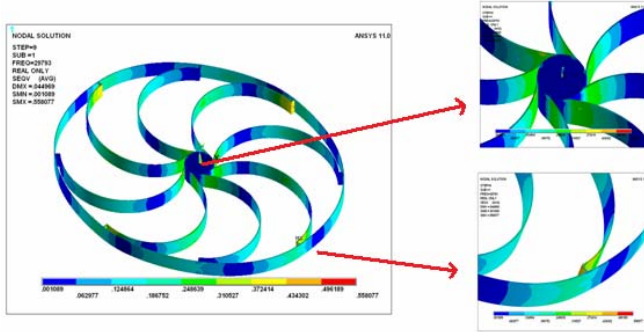


Figura 7. Distribución de los esfuerzos de von Mises (MPa) provocados por una velocidad de rotación de 1500 %/s en un microgiroscopio de 550 μm con un factor de calidad de 1200.

En el microgiroscopio 750 μm se estudio el comportamiento estructural en función a su factor de calidad. Se utilizó un rango de 500 a 20000 para una velocidad de rotación de 1000 %/s. La figura 8 muestra los resultados del desplazamiento máximo en el segundo modo flexionante de forma elíptica. Esta respuesta tiene incrementos significativos cuando el factor de calidad es superior a 2500. El máximo desplazamiento de 483.9 nm se obtiene con un factor de calidad de 20000. Este valor es 8.2 veces superior al obtenido con un factor de calidad de 1200. Un incremento en el factor de calidad es obtenido con un empaquetado a presiones muy bajas. La respuesta de los esfuerzos en función del factor de calidad se muestra en la figura 9. El incremento en los esfuerzos muestra una relación no lineal con relación al factor de calidad. El mayor esfuerzo es de 5.6 MPa para un factor de calidad de 20000. Este valor es 13.4 veces superior con respecto al obtenido con un factor de calidad de 1000. En la figura 10 se muestra la distribución de los esfuerzos de von Mises para una velocidad de rotación de 1000 %/s con un factor de calidad de 15000. Los esfuerzos de mayor magnitud se encuentran en la conexión de los soportes semicirculares con el anillo vibrante.

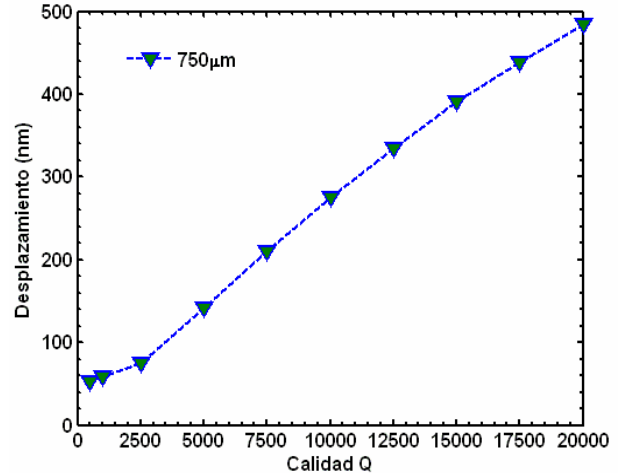


Figura 8. Desplazamiento (nm) vs el factor de calidad de un microgiroscopio de 750 μm con una velocidad de rotación de 1000 %/s.

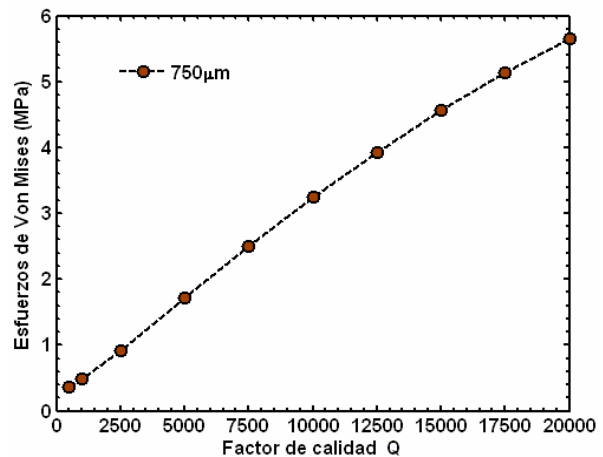


Figura 9. Esfuerzo de von Mises (nm) vs el factor de calidad de un microgiroscopio de 750 μm con una velocidad de rotación de 1000 %/s.

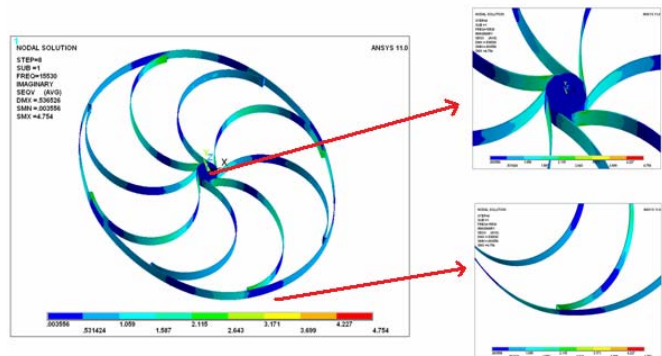


Figura 10. Distribución de los esfuerzos de von Mises (MPa) provocados por una velocidad de rotación de 1000 %/s en un microgiroscopio de 750 μm con un factor de calidad de 15000.

V. CONCLUSIONES

Modelos de elementos finitos en 3D se utilizaron para el análisis estructural y modal de microgiroscopios de anillo vibrante con un radio de 550 μm y 750 μm , respectivamente. Estos microgiroscopios tienen dos modos flexionantes de forma elíptica desfasados 45° entre ellos. Un conjunto de electrodos alrededor del anillo puede excitar electrostáticamente el microgiroscopio y otro conjunto de electrodos sensa capacitivamente el desplazamiento del anillo cuando existe un cambio en la velocidad de rotación. La estructura de 750 μm presentó su primer modo flexionante a la frecuencia de 15530 Hz. El desplazamiento máximo del anillo vibrante mostró un comportamiento lineal con respecto a la velocidad de rotación para un valor fijo del factor de calidad ($Q=1200$). En cambio, la respuesta del desplazamiento máximo del anillo vibrante registró una relación no lineal con respecto al factor de calidad. Para calidades superiores a 2500 se presentaron incrementos significativos en la respuesta estructural de los microgiroscopios. Los máximos esfuerzos de von Mises se concentraron en los extremos de los soportes semicirculares. Para un factor de calidad de 20000 y una velocidad de rotación de 1000 °/s se encontró un esfuerzo máximo de 5.6 MPa.

VI. REFERENCIAS

- [1] N. F. Yazdi, K. Najafi, “*Micromachined Inertial Sensor*” *Procc. IEEE*, Vol. 86, pp. 1640-1659, 1998.
- [2] M. Weinberg, J. Borenstein, J. Campbell, J. Cousens, B. Cunningham, R. Fields, P. Greiff, B. Hugh, L. Niles, J. Sohn, “*Micromachining Inertial Instruments*”, *Proc. SPIE 1996 Symp. Micromachining and Microfabrication*, Austin, Texas, 1996.
- [3] C. Song, “*Commercial Vision of Silicon based Inertial Sensors*”, *Tech. Dig. 9th Int. Conf. Solid-State Sensors and Actuators (Transducers’97)*, Chicago, Illinois, pp 839-842, 1997.
- [4] M. Kraft, “*Micromachined Inertial Sensors: The State of the Art and a Look into Future*”, *IMC Meas. Control*, Vol 3, pp. 164-168, 2000.
- [5] S. Park, “*New Adaptive Mode of Operation for MEMS Gyroscope*”, *J. Dyn. Sys., Meas., Control*, Vol. 126, pp. 800-810, 2004.
- [6] H. Kuisma, “*Inertial Sensors for Automotive Applications*” *Tech. Dig. 11th Int. Conf. Solid-State Sens. Act. (Transducer’01)*, pp. 430-433, 2001.
- [7] M. S. Weinberg, C. Wall, J. Robertsson, E. O’Neil, K. Sienko, R. Fields, “*Tilt Determination in MEMS Inertial Vestibular Prosthesis*”, *J. Biomech. Eng.* Vol. 128, pp. 943-956, 2006.
- [8] M. W. Putty, *A Micromachined Vibrating Ring Gyroscope*, Ph. D. tesis de doctorado, Michigan University, Marzo 1995.
- [9] S. Chang, et al., “*An Electroformed CMOS Integrated Angular Rate Sensor*”, *Sensor Actuators A*, Vol. 66, pp. 138-143, 1998.
- [10] F. Ayazi, K. Najafi, “*A HARPSS Polysilicon Vibrating Ring Gyroscope*”, *J. Microelectromech. Syst.* Vol. 10, pp. 169-179, 2001.
- [11] A. Lawrence, *Modern Inertial Technology: Navigation, Guidance, and Control*, 2da. edición, Springer-Verlag, 1998.
- [12] H. A. Rothbart, T. H. Brown, *Mechanical Design Handbook*, New York: McGraw-Hill, 2006.

ESPECTRÓMETRO DE VOZ USANDO LA TECNOLOGÍA MEMS

Edgar Julian Cabrera Magaña
Rubén Cabrera Magallón

edgy_canel@hotmail.com
cabreramagallon@hotmail.com

Área de Mecatrónica, Departamento de Electrónica
Universidad Autónoma de Guadalajara
Av. Patria No. 1201.

RESUMEN

Este artículo presenta el diseño de un espectrómetro de voz utilizando microsistemas electromecánicos (o Mems por sus siglas en inglés). El espectrómetro de voz está diseñado en base a un banco de resonadores, los cuales son filtros mecánicos de segundo orden pasa bajas. Se aprovecha la zona de la frecuencia natural del sistema como zona de trabajo, con esto se logra simular un alto factor de calidad y se puede separar la voz en una gran cantidad de armónicos.

Cada resonador del banco está formado por dos placas paralelas, las cuales forman un capacitor. Una de las placas se encuentra fija mientras que la otra está suspendida en el aire con ayuda de un resorte, formando el filtro de segundo orden.

Palabras clave: Mems, resonador, espectrómetro.

I. INTRODUCCIÓN

Las cuerdas vocales que están en el aparato fonador de cada ser humano son las que en última estancia permiten emitir el habla [5]. Desafortunadamente estas cuerdas se pueden dañar por diversas causas, tales como golpes, desgaste, accidentes, etc., lo que ocasiona que se pierda la voz. Las cuerdas vocales pueden ser recuperadas mediante intervenciones quirúrgicas y rehabilitación. Durante la rehabilitación, la persona debe emitir sonidos a frecuencias preestablecidas para una correcta recuperación.

Para determinar si la persona está emitiendo la frecuencia correcta se utiliza un espectrómetro de voz. Los espectrómetros actuales están basados en bancos de filtros implementados en microprocesadores o procesadores digitales de señales (DSP). Desafortunadamente se requieren aproximadamente 20 filtros de un alto factor de calidad para que el espectrómetro sea adecuado para la rehabilitación, lo cual hace que los sistemas sean muy costosos.

En este trabajo se presenta el diseño de un espectrómetro de voz de bajo costo, ya que los filtros usados son mecánicos tipo Mems [3], los cuales evitan el tratamiento de la señal de forma digital y son muy baratos en su construcción. Al usar estructuras micrométricas se logra un alto factor de calidad y una gran exactitud en la medición de la frecuencia.

Existen varios tipos de resonadores. Los hay de membrana, los cuales constan de dos placas que se traslapan entre si, dejando un hueco de aire entre ellas, haciendo que cuando se le aplique cierta fuerza a una de las placas, en el hueco se produzcan vibraciones a ciertas frecuencias. En estos resonadores se mide la frecuencia de resonancia a la cual comienza a vibrar y su valor en capacitancia. También los hay de placa, que consta de dos placas separadas entre si por un espacio de aire, teniendo el mismo principio del resonador de Membrana.

En este artículo vamos a utilizar el resonador de Placa, ya que de los cálculos realizados para

diversos resonadores, este tipo de resonador resultó ser el que ofrece una alta capacitancia, un alto factor de calidad y se presta a su fabricación en tecnologías baratas disponibles para Mems.

Este artículo está organizado como sigue. La sección II introduce brevemente al diseño de Mems, las tecnologías de fabricación, herramientas de diseño y análisis comúnmente usadas en esta área. Nos muestra el panorama de lo que es un sistema electromecánico, sus usos más comunes en la actualidad y uno de los medios por medio del cual se pueden diseñar dichos sistemas. La parte III corresponde al Diseño de los resonadores. Esta parte nos muestra el diseño de uno de los resonadores centrado en los 200Hz, también se muestra el diagrama de cuerpo libre del resonador, así como su función de transferencia, los cálculos efectuados para la constante del resorte, la masa de las placas, su volumen y la capacitancia. La sección IV es la simulación del resonador en software como Matlab y Ansys. Aquí se muestran graficas obtenidas de la función de transferencia y su interpretación para nuestros fines, así como la simulación del resonador aplicándole una fuerza.

II. TECNOLOGÍA MEMS

Si bien un Mems nació como un dispositivo electromecánico cuyo tamaño oscila entre unos trescientos nanómetros a un milímetro, en la actualidad las aplicaciones se han diversificado enormemente, ya que se tienen dispositivos Mems especializados en óptica, en radiofrecuencia [1], en procesos biológicos, fluidos, etc. Los Mems son una tecnología que ha permitido resolver muchos problemas y hay dispositivos que comercialmente han sido exitosos, por ejemplo los sensores de presión, los micrófonos capacitivos, los cartuchos de inyección de tinta para impresoras, los acelerómetros que disparan las bolsas de aire de los automóviles, los procesadores digitales de luz (DLP) que permiten proyectar la imagen del monitor de una computadora y muchos otros más.

Existen varias tecnologías para la fabricación de MEMS, y cada una de ellas tiene sus propios procesos de fabricación. Una de las tecnologías

más difundidas es la de PolyMUMPs. Con esta tecnología se pueden hacer Mems superficiales. PolyMUMPs tiene tres capas de polisilicio y una de metal. Dos capas de polisilicio son libres (se pueden mover) y la otra está anclada. El metal siempre se deposita sobre una capa de polisilicio con el fin de tener buena conducción eléctrica [4]. La tecnología PolyMUMPs es popular ya que es una de las más baratas donde se pueden desarrollar prototipos.

También existen editores de “layouts”. Estos editores gráficos presentan las capas de polisilicio que tiene PolyMUMPs y los trabajos que se pueden realizar. Este tipo de editores no es nuevo, ya que es el que se usa para especificar las máscaras de grabado de circuitos integrados. En el presente trabajo se utilizó el editor L-Edit. Este editor trae un archivo de texto que le indica al editor la tecnología que se está trabajando. En nuestro caso este archivo le indica que se usa PolyMUMPs. Con esto el editor restringe el trabajo a las tres capas de polisilicio, a los trabajos de anclado y vías de polisilicio.

III. DISEÑO DE RESONADORES

Como muestra del diseño de resonadores, se presenta el diseño de uno de ellos centrado en los 200Hz y con un ancho de banda no mayor a 100Hz. Además debe ser de tamaño micrométrico, ya que 20 de ellos y el equipo total debe ser portátil. Dado el ancho de banda, se tiene que el factor de calidad debe ser alto (sobre todo para frecuencias altas, a unos 2000 Hz). La señal a medir debe ser alta para no confundirla con ruido, en el presente caso indica que la capacitancia debe ser alta.

De los diferentes tipos de resonadores se seleccionó el resonador de placas paralelas, el cual consta de dos placas, una móvil y otra fija, un par de resortes y conectores hacia las dos placas. La placa móvil está suspendida con los resortes y queda paralela a la placa fija, sin tocarla. La figura 1 muestra la disposición de los elementos de este tipo de resonador. En dicha figura se observan los resortes [2], la placa móvil (la placa fija está justo debajo de la placa móvil) y los conectores.

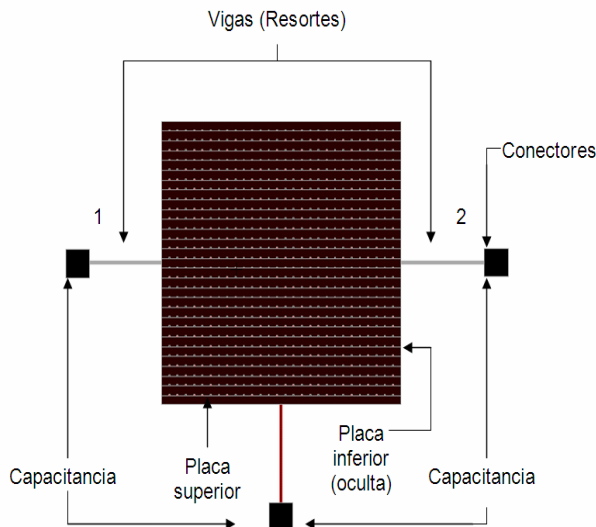


Fig.1. Vista superior del resonador.

Diagrama de cuerpo libre

El diagrama de cuerpo libre del resonador propuesto se muestra en la figura 2. Con este diagrama se procede a obtener la función de transferencia del resonador.

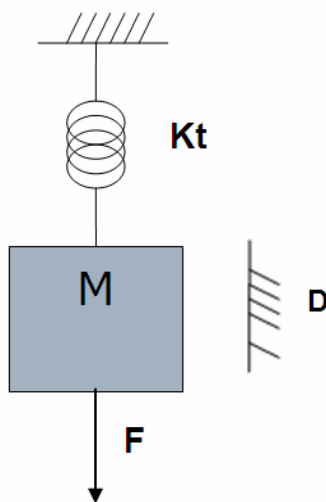


Fig.2. Sistema masa resorte amortiguador

Del diagrama de cuerpo libre se obtiene la ecuación diferencial de movimiento del sistema. En esta ecuación m es la masa de la placa móvil, D es el coeficiente de fricción del aire con la placa móvil y K es la constante de elasticidad del resorte. Note también que la constante de elasticidad está relacionada con el momento de inercia de las vigas de la figura 1.

$$F = m \frac{d^2 x}{dt^2} + D \frac{dx}{dt} + Kx$$

Ahora aplicamos Laplace y obtenemos:

$$F(s) = ms^2 X(s) + DsX(s) + kX(s)$$

$$F(s) = X(s) * (ms^2 + Ds + K)$$

Por lo tanto la función de Transferencia queda de la siguiente forma.

$$\frac{X(s)}{F(s)} = \frac{1}{ms^2 + Ds + K}$$

Sabemos que el polisilicio tiene un módulo de Young (E) de 127 Gpa y una densidad de 2.3 g/cm³. En PolyMUMPs la separación entre las placas es de $H=3\mu\text{m}$ y el grosor de la placa de $W=1.5\mu\text{m}$. Como existen muchas variables y pocas ecuaciones, se puede proponer el valor de algunas de ellas, en el presente caso las distancias propuestas son $L_1 = 300\mu\text{m}$ y $L_2 = 350\mu\text{m}$.

Utilizando el momento de inercia de las vigas uno y dos de la figura 1, podemos calcular la constante de elasticidad del resorte K la cual es:

$$k = \left(\frac{\Pi^4}{6} \right) * \left(\frac{E * W * H^3}{(2L_1)^3 + (2L_2)^3} \right)$$

$$k = \left(\frac{\Pi^4}{6} \right) * \left(\frac{127\text{GPa} * 1.5\mu\text{m} * 3\mu\text{m}^3}{(600\mu\text{m})^3 + (700\mu\text{m})^3} \right)$$

$$k = 0.149 \text{ N/m}$$

$$k_{total} = 0.298 \text{ N/m}$$

Sabiendo que la frecuencia de resonancia $W_r = 1000 \text{ rad/seg}$ (para una frecuencia aproximada de 150 HZ) y que la $K = 0.298 \text{ N/m}$, calculamos la masa del sistema con:

$$W_r = \sqrt{\frac{k}{m}}$$

$$1000 \text{ rad/seg} = \sqrt{\frac{0.149 \text{ N/m}}{m}}$$

Despejando m tenemos que:

$$m = 1.49 \text{ E}^{-7}$$

Ahora calculamos el volumen, sabiendo que la densidad del polisilicio $d = 2.3 \text{ gr/cm}^3$ y que la masa $m = 1.49 \text{ E}^{-7}$:

$$d = \frac{m}{v}$$

Despejando v tenemos:

$$v = \frac{1.49 \text{ E}^{-7}}{2331 \text{ kg/m}^3}$$

$$v = 6.39 \text{ E}^{-11}$$

Considerando la fricción con el aire, podemos calcular el factor de amortiguamiento D tenemos:

$$D = \frac{nA}{h}$$

El valor de la constante n es:

$$n = 18 \mu\text{m}$$

$$D = \frac{18 \mu\text{m} * (1000 \mu\text{m} * 1.5 \mu\text{m})}{3 \mu\text{m}}$$

$$D = 9 \text{ E}^{-9}$$

Por último calculamos la Capacitancia con:

$$c = \frac{E_o * A}{d}$$

donde:

c = Capacitancia

E_o = Permitividad del aire = $8.85 \text{ E}^{-12} \text{ F/m}$

A = Área

d = Grosor de las Placas = $2 \mu\text{m}$

$$c = \frac{8.85 \text{ E}^{-12} * (1000 \text{ E}^{-6} * 1000 \text{ E}^{-6})}{2 \text{ E}^{-6}}$$

$$c = 4.425 \text{ E}^{-12}$$

IV. SIMULACIÓN

Sabiendo que la función de transferencia es:

$$\frac{X(s)}{F(s)} = \frac{1}{ms^2 + Ds + K}$$

y sustituyendo en ella los valores anteriormente obtenidos tenemos:

$$\frac{X(s)}{F(s)} = \frac{1}{1.49 \text{ E}^{-7} s^2 + 9 \text{ E}^{-9} s + 0.298}$$

Ya contando con la función de Transferencia, podemos ingresarla a Matlab y obtener la grafica de Bode de la siguiente manera:

```
>> t=tf([1],[1.49e-7 9e-9 .298])
>> bode(t)
```

lo cual arroja como resultado la siguientes gráficas:

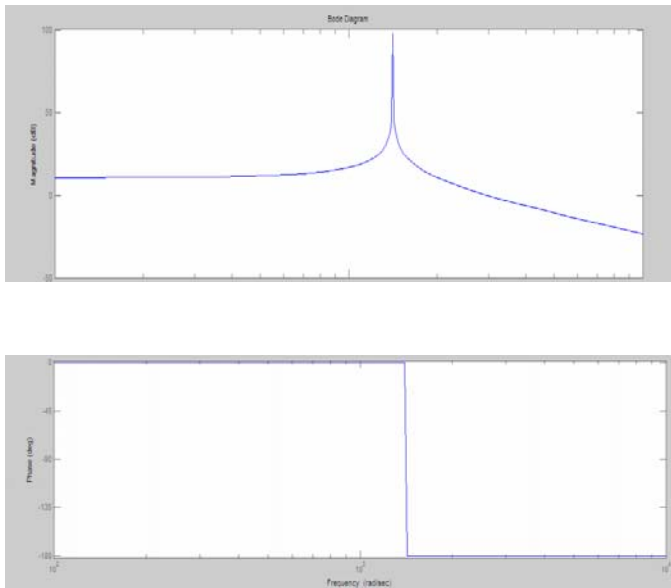


Fig.3. Gráficas de magnitud y fase del resonador.

En las gráficas se puede observar que la forma de onda corresponde a un filtro pasa bajas, con una frecuencia natural de 1410 rad/seg que corresponden a 224 Hz aproximadamente.

Un análisis semejante se puede realizar con simuladores de elemento finito, por ejemplo Ansys. En ese simulado se debe introducir la geometría del dispositivo y los materiales usados. También se le debe indicar los elementos que estarán fijos y por ende, el resto serán elementos móviles. Finalmente se debe introducir las fuerzas a las que estará sometido el dispositivo y Ansys podrá hacer un análisis de esfuerzos, de movimiento y modal, entre otros. Como ejemplo se hace una simulación del resonador propuesto aplicándole una fuerza de $F= 6.324E^{-17} \text{ N}/\mu\text{m}^2$ (equivalente a unos 10 db de audio) para estudiar el desplazamiento de la placa suspendida en el aire.

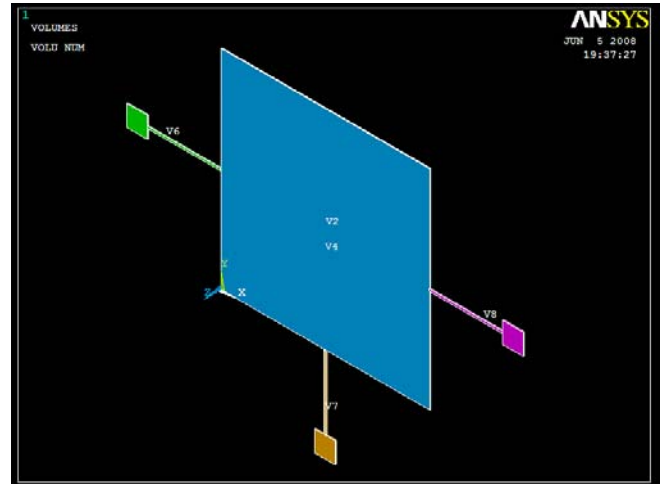


Fig.4. Vista tridimensional del resonador en el software ansys.

Los resultados arrojados de acuerdo a la fuerza aplicada muestran un desplazamiento vertical de la placa suspendida sin rupturas ni perturbaciones.

En base a los resultados obtenidos en la simulación podemos decir que el resonador propuesto es funcional para frecuencias con un rango de 200 a 250 Hz.

Por último se muestra el diagrama de bloques que contiene los componentes mecánicos y eléctricos con los cuales se va a probar el espectrómetro de voz.

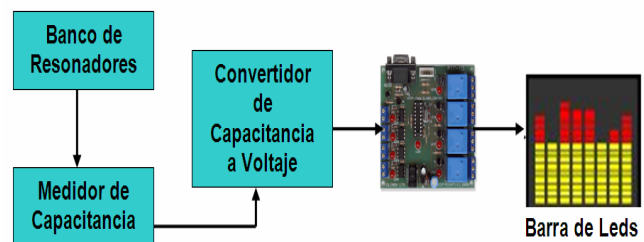


Fig.5. Diagrama de bloques del Banco de resonadores, medidor y convertidor de capacitancias, microprocesador y barra de leds.

V. CONCLUSIONES

En este proyecto se ha presentado el prototipo de uno de los resonadores del espectrómetro de voz, el cual según los resultados obtenidos en la simulación muestra que es un sistema viable para formar parte del banco de resonadores que se va a emplear para la medición de las frecuencias provenientes de la voz.

VI. TRABAJO A FUTURO

Dicho resonador ya ha sido mandado a fabricar, por lo que solo resta esperar un lapso de dos a tres meses a que llegue para hacer las pruebas necesarias. Además no se descarta la posibilidad de modificar el diseño, en caso de ser necesario, si surgieran nuevas variantes o especificaciones.

VII. REFERENCIAS

- [1] *RF MEMS Circuit Desing for Wireless Communications*, Héctor J. De Los Santos, Artech House Boston London, 2002.
- [2] Dirk Lange, Oliver Brand, Henry Baltes, *CMOS Cantilever Sensor Systems*, Springer, Enero 2002.
- [3] Stephen D. Senturia, *Microsystem Desing*, Kluwer Academic Publishers, 2001.
- [4] *PolyMUMPs Desing Handbook*, Jim Carter, Allen Cowen, Busbee Hardy, Ramaswamy Mahadevan, Mark Stonefield, and Steve Wilcenski, MEMSCAP Inc., Revision 11.0
- [5] Gayton, *Tratado de Fisiología Médica*, Robert Gayton, Mc Graw Hill, Septiembre 2003.

Determinación de insecticida en pimiento morrón mediante el uso de tecnologías MEMS

Juan Hernández Vela, Akira Torreblanca Ponce, Nicolás Ortega

Departamento de MEMS, ingeniería bioquímica,
Universidad Instituto Tecnológico Superior de Irapuato
Carretera Irapuato-Silao, kilometro 2.5 Irapuato, Gto.
Teléfono (462) 60 67 900 e –mail: itesy@hotmail.com

Resumen

Por medidas de higiene y inocuidad de los alimentos se determinara la presencia de organofosforados en el pimiento morrón, utilizando tecnología de punta para la satisfacción de bienestar del agricultor al momento de exportar su producto, se creo un biosensor, por medio de electrodos, con ayuda de la tecnología MEMS; ya que este es uno de los principales problemas en cuanto a exportación originando perdidas considerables.

Actualmente se tiene el primer diseño fabricado de dicho sensor y se están realizando las pruebas de caracterización con el fin de realizar el rediseño del sensor, el cual proporcione la medición de la presencia o ausencia del organofosforado.

Palabras clave:

Acetilcolinesterasa, sensor, organofosforados, pimiento morrón, MEMS.

I. Introducción

La globalización de los mercado y la competencia creciente en el mercado nacional, han generado un entorno totalmente nuevo donde los productores y empacadores de nuestro país tienen retos importantes.

- Los patrones del consumo de alimentos han cambiado, pues los consumidores se preocupan cada vez más por la salud y calidad de los alimentos que consumen.
- Los países están cada vez más preocupados por proteger la salud de sus habitantes.

- En los principales mercados compradores existe un acelerado desarrollo de normas que garanticen la calidad de los alimentos que se comercializan y consumen.

México ocupó el segundo lugar en producción de pimientos frescos con una producción de 14, 593. 000 toneladas en el año 2006.

Los productos agrícolas en México representan cerca del 6% del total de las exportaciones. En el año 2006 los productos agrícolas registraron exportaciones por más de 13 mil millones de

dólares, de los cuales el 30% correspondió a frutas y hortalizas. El pimiento dulce se cultiva principalmente en el estado de Sinaloa, en donde se produce en campo abierto en suelo arenoso con acolchados de plástico y con sistemas de riego por goteo. (Lisa y col. 2008)

Las nuevas tecnologías de invernadero permiten la producción de estos pimientos en la mayoría de climas fríos-cálidos del México central. (Damasio H. y col. 2001) Estas nuevas instalaciones de producción de pimiento de invernadero están localizadas en Guanajuato, Jalisco, San Luís Potosí y la península de Yucatán, en donde los rendimientos y la calidad son supremos teniendo un peso de 500g y la planta llega a medir hasta 2.5m. (Lisa y col. 2008)

El producto deberá producirse y manipularse de acuerdo con lo establecido en los Lineamientos para la Certificación de Buenas Prácticas Agrícolas y Buenas Prácticas de Manejo en los Procesos de Producción de Frutas y Hortalizas para Consumo Humano en Fresco, publicados por el Servicio Nacional de Sanidad, Inocuidad y Calidad Agroalimentaria (SENASICA). (firco. 2002).

En la actualidad Irapuato cuenta con agricultores que exportan su producto al extranjero el cual puede ser regresado por tener presencia de organofosforados teniendo el productor pérdidas considerables.

Estos insecticidas son utilizados para la eliminación de algunas plagas tales como la mosca blanca, araña roja, araña blanca, mosca blanca pulgón, cochinillas. (Manuel llanos. 2006) La lucha contra las plagas del pimiento por métodos biológicos cuenta con un amplio espectro de depredadores y parásitos, así como de microorganismos patógenos, que atacan o viven a expensas de algunas de ellas. (Altieri y col. 2000)

En la lucha química se recomiendan los tratamientos con organofosforados (clorpirifos, carbamatos (acefato, metomilo) y tratamientos

órgano clorados que fue lo que se utilizaron mucho antes que los organofosforados, los compuestos órgano clorados tales como DDT, aldrin, dieldrin, toxafeno y endrin fueron los primeros insecticidas comercializados. Los efectos de estos órgano clorados, se ha comprobado que el DDT reduce las reservas de vitamina A en madres que están amamantando; los síntomas de sobre exposición son mas comunes después de la ingestión, se incluyen la hiperexcitabilidad, temblores y convulsiones en casos extremos (Obiols J.1998). Los organofosforados son el resultado de un proceso industrial de síntesis química, y se han convertido en la forma dominante del componente a las plagas, (Aboudonia y col. 1990) estos compuestos son considerados como a polares lo que significa que desde el punto de vista químico la mayoría son escasamente solubles en agua, aunque con grandes diferencias de un compuesto a otro, y desde el punto de vista biológico tienden a disolverse en grasas. (Methner y col. 1994) La estabilidad de los organofosforados depende del PH fuertemente alcalino se descomponen, lo que puede ser utilizado para destruirlos. Los plaguicidas organofosforados constituyen un amplísimo grupo de compuestos de síntesis, en general altamente tóxicos.

Los compuestos organofosforados, su exposición pueden tener lugar por tres vías clásicas: digestiva, por ser inhalada y dérmica. La vía digestiva directa se suele considerar como accidental (ingestión de una solución por error o con fines suicidas, o de alimentos directa o indirectamente contaminados); es bien conocidos que las vías inhalatorias y la dérmica están muy estrechamente relacionadas con la exposición en las distintas operaciones en que se pueden manipular este tipo de productos por parte de operarios con distintas actividades. (Aboudonia y col. 1990)

Una vez absorbidos, los organofosforados y sus metabolitos se distribuyen rápidamente por todo los órganos y tejidos, aunque las concentraciones

mas elevadas se alcanzan en el hígado y los riñones, antes de ser eliminados de manera prácticamente total por la orina y las heces. No obstante, los compuestos mas lipofílicos pueden almacenarse en pequeña proporción en los tejidos grasos y el tejido nervioso, dada su riqueza en lípidos, en donde pueden ser posteriormente liberados. (Hartman 1989).

El catabolismo (descomposición en sustancias más sencillas) de los compuestos organofosforados una vez absorbidos tiene lugar, en parte, a través de las llamadas esterases “A”, enzimas que los hidrolizan a una velocidad considerable, actuando como destoxificadoras. Las esterases “B” no tienen, en general, esta función y, muy al contrario, son las moléculas diana sobre las que los organofosforados actúan en el organismo, ejerciendo así su acción tóxica, como es el caso de la acetilcolinesterasa (con una muy destacada función fisiológica en el sistema nervioso) cuya actividad bioquímica resulta inhibida, con una rapidez e intensidad que dependen de la naturaleza del propio compuesto, además de su concentración (Peyster 1993). La acetilcolinesterasa, además de encontrarse en los glóbulos rojos, donde no se le conoce acción fisiológica, regula la transmisión de los impulsos nerviosos en las terminaciones colinérgicas (por hidrólisis de la acetilcolina, que actúa como neurotransmisor, una vez ha alcanzado su destino) de las neuronas preganglionares del sistema simpático y parasimpático (Hartman 1989).

El acumulo de acetilcolina en cualquiera de esos puntos que se acaban de citar, por inhibición de la actividad colinesterásica, trae como consecuencia la aparición de trastornos de mayor o menor intensidad y de naturaleza distinta (Hartman 1989).

En la actualidad el monitoreo o la detección de organofosforados se hace mediante la cromatografía de gases en donde La cromatografía de gases es la técnica a elegir para la separación de compuestos orgánicos e inorgánicos térmicamente estables y volátiles. El inconveniente de este tipo de técnica es su costo al publico un aparato de estos cuesta aproximadamente 100 000.00 pesos, en laboratorios una prueba de esta técnica cuesta aproximadamente 6 000. 00 pesos siendo un costo muy excesivo, otro inconveniente es el tiempo de determinación que es de 45 min. Por muestra (UNAM. 2005).

Al tener esta problemática nosotros proponemos una nueva tecnología la cual será fácil utilizarla, bajo costo, y su medición será de 3 min, y una ventaja es su tamaño, que será fácil de transportarse y hacerse mediciones en el momento (in situ).

Donde el MEMS son pequeñas estructuras electro-mecánicas que consisten de componentes en micrómetros que se usa primordialmente el silicio ya que, este es compatible con algunos metales, también es utilizado por sus limites de elasticidad que es de $7E9$ n/m² a comparación con el metal que es de $4.4E9$ n/m, es mas ligero cuya densidad es de 2.3 g/cm³ y la del acero es de 7.9g/cm³. (Ryo Y. y col. 2001)

MEMS es una tecnología que permita el desarrollo de productos inteligentes, aumentar la capacidad computacional de la microelectrónica con la percepción y capacidades de control de microsensores y microactuadores y ampliar el espacio de posibles diseños y aplicaciones. (MEMS. 2005)

Los dispositivos MEMS permiten el procesamiento de especímenes, reacciones bioquímicas, y la detección del producto

resultante en un formato de flujo, con un control electrónico de diversos componentes tales como bombas termo-neumáticas, micro-calentadores y sensores de temperatura, detectores de fluorescencia miniaturizado, concentrador de muestra y sustancias químicas de proceso, y filtros. (Ryo Y. y col. 2001)

El término “sensar” definido como “una habilidad para juzgar, discriminar, o estimar condiciones externas” es notablemente diferente de la definición de “detectar” como “descubrir algo oculto” (Romani A. y col. 2000). Se les llama *biosensores amperométricos* a este tipo de dispositivos debido al uso de material biológico para producir la interrupción o el paso de corriente a través de un par de electrodos. (Teshome T. Y col. 1999)

PLANTEAMIENTO DEL PROBLEMA

- La principal problemática que presenta la exportación del pimiento es la presencia de insecticidas motivo por el cual puede ser rechazado en la frontera ocasionando pérdidas económicas considerables para el productor.
- Donde actualmente existen aparatos de detención de organofosforados, estos aparatos son muy costosos, de difícil utilización, si no estar antes capacitado para utilizarse, y de larga duración.
- Al tener esta problemática nosotros proponemos una nueva tecnología la cual será fácil utilizarla, bajo costo, y su medición será de 2min, y una ventaja es su tamaño, que será fácil de transportarse y hacerse mediciones en el momento.

II. Metodología

El trabajo es realizado por el tecnológico superior de Irapuato, el cual se inicio por la búsqueda de la información es te consiste en realizar visitas de campo con los productores de pimiento morrón bajo invernadero en el estado de Guanajuato, con el propósito de detectar la problemática de dicho producto. Una vez detectada la problemática se realizó investigación documental para poder atacarla.

Luego se diseño el sensor este consiste en un sistema de micro electrodos de níquel recubiertos de oro mediante la utilización del proceso METALMUMPS este se puede ver en la fig. 1.

Donde el diseño fue fabricado por la empresa MEMSCAPS en Estados Unidos de Norte América. Lugo se realizo una caracterización en esta etapa se realizan mediciones de impedancia y reactancias a frutos contaminados con organofosforados a diferentes concentraciones con el sensor propuesto.

Se prepararon 10 diferentes concentraciones de órgano fosforado comercial por prueba de caracterización empezando a concentraciones elevadas iniciando con concentraciones en gr., ml gr., nano gr., la medición de gramos fue para ver si el sensor era capaz de medir la concentración administrada.

Con el fin de determinar la sensibilidad del sensor es necesario realizar pruebas de caracterización con concentraciones a nivel nano gramos lo cual establece la norma oficial mexicana.

Posteriormente se fumigo el pimiento durante una semana por inmersión en la concentración de órgano fosforado correspondiente, los pimientos fueron almacenados a 15 C. Una vez concluido

el tiempo se realizó una extracción de cada pimiento sometido a fumigación mediante una molienda manual, los extractos fueron colocados en tubos falcón con aproximadamente 15ml. de muestra.

A cada uno de los extractos obtenidos se le realizaron mediciones de monitoreo de conductividad, resistividad y pH, contando con un control negativo, el cual no fue sometido a contaminación por organofosforado.

Al final se hizo una integración de los datos para obtener su resultado.

III. Resultados y discusión

El sensor propuesto para la detección de insecticidas en este momento está ya fabricado (ver figuras 1 y 2) y se está realizando las pruebas de caracterización de dicho sensor, durante el desarrollo de ciertas prácticas concernientes a la caracterización de dicho sensor, se realizarán las modificaciones pertinentes propuestas en el cronograma de actividades presentado en este documento.

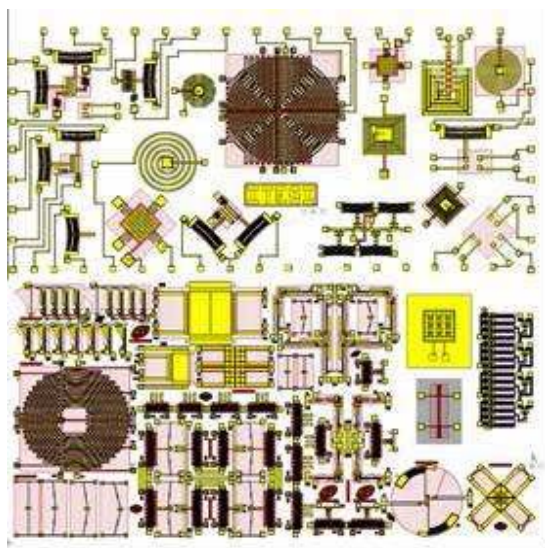


Figura 1. Diseño de fabricación por el método METALMUMPS Junio 2006.

Así mismo se plantea además de rediseñar el sensor de insecticidas mediante un sistema de electrodos, realizarlos mediante un sistema denominado cantiliver, el cual consiste en inmovilizar la enzima y hacer el monitoreo mediante un incremento en el peso de la viga, monitoreando la deflexión de la misma.

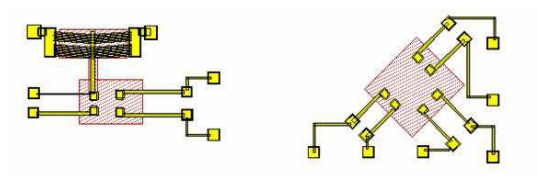
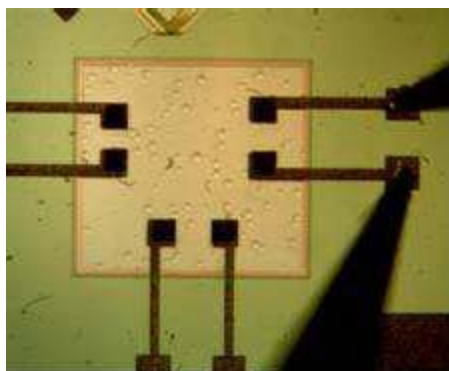


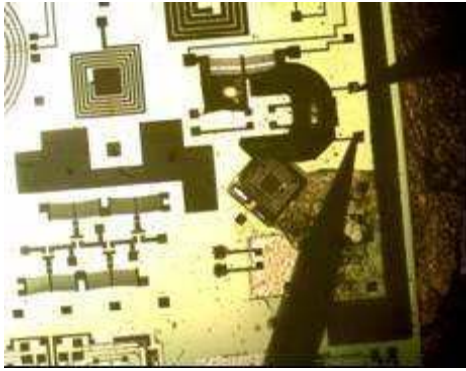
Figura 2. Diseño del sensor de determinación de insecticidas mediante el sistema de electrodos. Fabricados bajo la tecnología METALMUMPS (junio 2006).

- a) Electrodos variables de acuerdo al desplazamiento lineal de la oscilación del chevron.
- b) Sistema de electrodos para la medición de insecticidas en frutos de interés.

Todo esto con el fin de comparar los dos diseños propuestos y poder integrar la electrónica al diseño con mayores beneficios, ya sea económicos, factibilidad, etc.

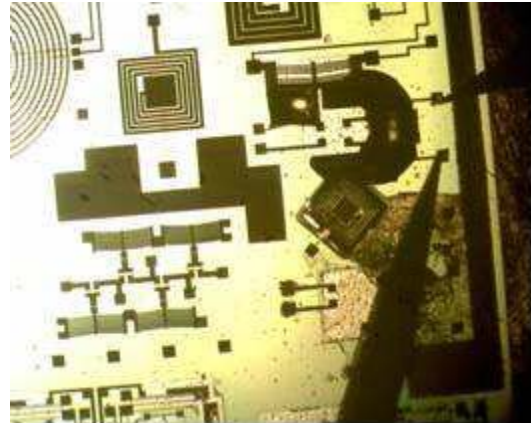


a)



b)

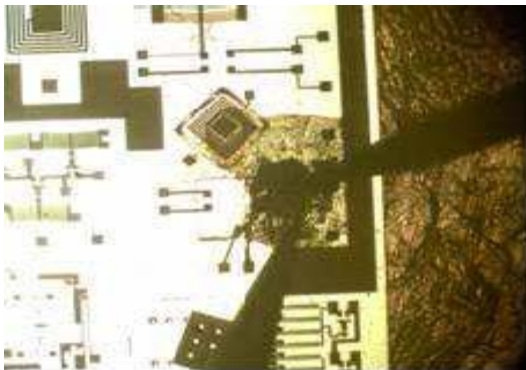
Figura 3. a) Sensor para la determinación de insecticidas vista con el micro manipulador en el objetivo 40X. b) Colocación de la muestra en el contenedor de medición para la determinación de organofosforados.



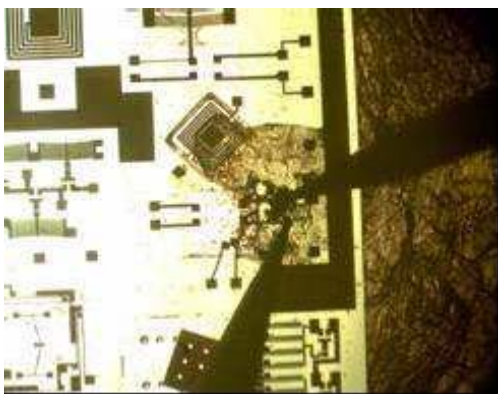
c)

Figura 4. Determinación de insecticidas en una muestra testigo.

En las figuras 3(b), 4(a, b y c) puede observarse la medición de insecticidas en muestras que contenían organofosforados a diferentes concentraciones. La determinación es obtenida mediante el monitoreo de la reacción, la cual es convertida a una señal eléctrica y monitoreada con un sistema de medición electrónico (multímetro y osciloscopio).



a)



b)

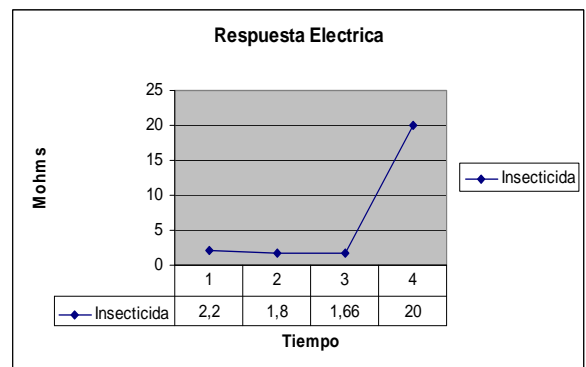
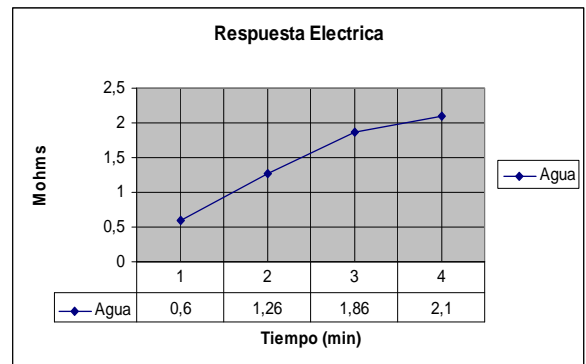


Figura 5. Grafica de respuesta eléctrica.

A) Testigo Negativo. Muestra correspondiente a agua sin organofosforados b) Muestra Testigo positivo. Muestra que contenía organofosforados. Como puede observarse en la figura 5 si existe diferencia en la respuesta eléctrica en la muestra que no contenía organofosforados y en la que si contenía.

En la figura 5(A) puede observarse que la respuesta incrementa de manera lineal, comportamiento típico en la medición de Agua. En cambio en la figura 5(B) puede observarse que la señal permanece constante durante los primeros dos minutos para posteriormente incrementarse de manera rápida lo cual podría ser el indicativo de la detección del organofosforados, sirviendo esto como base en el diseño de circuitería y diseño electrónico del sensor.

IV. Conclusión

En esta primera etapa los cambios son visibles en las graficas anteriores, dependiendo del grado de concentración del insecticida que se encuentra en cada una de las muestras que se utilizaron para hacer el estudio del pimiento morrón. Para ello se utilizo un sofisticado software, en donde a cada una de las muestras se les aplico una variación de frecuencia, haciendo varias mediciones de cada una de las muestras.

Existe evidencia significativa del 95% de que la variación de concentración de insecticida influye en la reactancia y resistencia la cual indica que el sensor es posible detectar insecticidas organofosforados en el pimiento.

V. Referencias

- [1] ABOUDONIA, M., LAPADULA, D. M.(1990). *Mechanisms of organophosphorus esterinduced delayed neurotoxicity: Type I and Type II. Annu Rev Pharmacol Toxicol* 30: 405-440.
- [2] Dámasio Hornero- Méndez, and M. Isabel Mínguez-Mosqueda, “*Rapid Spectrophotometric Determination of Red and Yellow Isochromic Carotenoid Fractions in Paprika and Red Pepper Oleoresins*”, *Journal of Agricultural and Food Chemistry*, Vol.49, No.8, pp. 3584-3588,2001.
- [3] HARTMAN. E. D.Neuropsychological Toxicology.(1989). *Identification and Assessment of Human neurotoxic Syndroms.Pergamon Press, New York, second printing.*
- [4] Lisa M. Papageorge, Roger F. (2008). McFeeters, and Henry P. Fleming, “*Factors Influencing Texture Retention of Salt-Free, Acidified, Red Bell Peppers during Storage*”, *Journal of Agricultural and Food Chemistry*, Vol. 51, No. 5, pp. 1460-1463.
- [5] METHNER, M. M., FENSKE, R. Pesticide Exposure During Greenhouse Applications, *Part II. Chemical Permatation Through Protective Clothing in Contact with Treated Foliage. Appl Occup Environ Hyg* 9: 567-574. (1994).
- [6] OBIOLS, J. *O.Plaguicidas organofosforados. (Capítulo 16) págs, 253-288. En: Control biológico de los trabajadores expuestos a contaminantes químicos. Instituto Nacional de Seguridad e Higiene en el Trabajo, Madrid, 1998.*

- [7] Altieri M., Nicholls C. (2000) *AGROECOLOGÍA. (Capítulo 7) pags; 147-153. Teoría y práctica para una agricultura sustentable primera edición.*
- [8] PEYSTER, A., WILLIS, W.O., y cols. (1993). *Cholinesterase and Selfreported Pesticide Exposure among Pregnant Women. Arch Environ Health 48: 348-352.*
- [9] Romani, A. M. Minunni, N. Mulinacci, P. Pinelli, and F. F. Vincieri, (2000) "Comparison among Differential Pulse Voltammetry, Amperometric Biosensor, and HPLC/DAD Analysis for Polyphenol Determination", *Journal of Agricultural and Food Chemistry, Vol. 48, No. 4, pp. 1197-1203.*
- [10] Ryo Yamauchi, Koichi Aizawa, Takahiro Inakuma, and Koji Kato, (2001) "Analysis of Molecular Species of Glycolipids in Fruit Pastes of Red Bell Pepper (*Capsicum annuum* L.) by High-Performance Liquid Chromatography-Mass Spectrometry", *Journal of Agricultural and Food Chemistry, Vol. 49, No. 2, pp. 622-627.*
- [11] Teshome Tadesse, Michael A. Nichols, and Keith J. Fischer, (1999). "Nutrient Conductivity Effects on Sweet Pepper Plants Grown using a Nutrient Film Technique. 2. Yield and Fruit Quality", *New Zealand Journal of Crop and Horticultural Science, Vol. 21, pp. 229-237.*
- [12] Manuel Ilanos (2006). (Control y tratamientos fitosanitarios en el cultivo del pimiento). <http://www.eumedia.es/articulos/vr/horto/frut/83pimiento.html>. (2008)
- [13] Micro Electro Mechanical Systems (MEMS) (2005), www.memsnet.org/mems/what-is.html, (2008)
- [14] Producción de Pimiento Morrón bajo condiciones de invernadero, (2002), www.firco.gob.mx/agronegocios/docs/FO MAGRO2002-2006ProyectosExito.pdf. (2008).
- [15] UNAM (Universidad autónoma de México (2005.) *Instrumentación en cromatografía de gases (2008).* www.dacb.ujat.mx/material_academico/d_escarga/instrum_cromatografica.ppt.

PROCESO DE FABRICACIÓN DE MICROESTRUCTURAS PARA APLICACIONES EN MEMS UTILIZANDO MICROMAQUINADO SUPERFICIAL

A. Alanís, D. Díaz, M. Landa, P. Alarcón, N. Carlos, A. Torres, J. Hidalgo, M. Linares, J. Molina, P. Rosales, C. Zúñiga, C. Reyes y W. Calleja

INAOE

Laboratorio de Microelectrónica y Centro de Diseño de MEMS

A. P. 51, C. P. 72000, Puebla, México

wcalleja@inaoep.mx

RESUMEN

En este trabajo se presenta un proceso de fabricación de microestructuras utilizando la técnica de micromaquinado superficial. El proceso utiliza tres niveles de *Silicio Policristalino* (polisilicio, poly-Si), *PSG* como material de sacrificio, y *Aluminio* como material de interconexión eléctrica. Las estructuras fabricadas son de operación estática o dinámica. Los dispositivos de operación dinámica pueden funcionar como sensores y/o actuadores. Los dispositivos de operación estática son principalmente monitores para determinar cualitativa y cuantitativamente algunas características mecánicas del poly-Si.

Este proceso de fabricación es caracterizado utilizando un circuito de pruebas que incluye: dispositivos eléctricos tales como resistencias de película delgada, resistencias de contacto, resistencia superficial; elementos ópticos para monitoreo de dimensiones; además de algunos parámetros mecánicos tales como esfuerzos mecánicos residuales (σ_r) y gradientes de esfuerzos ($\Delta\sigma/z$); los que en conjunto permiten monitorear *in-situ* y *ex-situ* el proceso de fabricación de microestructuras con polisilicio denominado PolyMEMS INAOE.

Palabras clave: MEMS, micromecanismos, sensores, actuadores, micromaquinado superficial, silicio policristalino, esfuerzos residuales, gradientes de esfuerzo.

I. INTRODUCCIÓN

El poly-Si es un material ampliamente utilizado como material estructural en micromaquinado superficial, para la fabricación de sensores y actuadores con aplicaciones principalmente en MEMS. Las propiedades mecánicas del poly-Si dependen fuertemente de su estructura, y ésta a su vez, de las condiciones de depósito y de los tratamientos térmicos subsecuentes [1]. Las propiedades mecánicas (entre ellas el *módulo de young o elasticidad*) del poly-Si dependen de la orientación dominante de los granos, y en general de su estructura [2]. Específicamente, las fronteras de grano influyen directamente en las propiedades mecánicas como la resistencia a la deformación (*yield strength*) [3], y la resistencia a la fractura (*fracture strength*), debido a que tienden a modular el desplazamiento de las dislocaciones (dichos desplazamientos originan la deformación plástica ó permanente de los materiales). El hecho de influir en el tamaño de grano (o controlar la densidad de granos) posibilita el uso optimizado del poly-Si en condiciones elásticas y así evitar la condición de fractura [4]. El poly-Si depositado por LPCVD a temperaturas superiores a 600°C presenta una estructura de tipo columnar, con paredes orientadas preferencialmente en la dirección $\langle 110 \rangle$ [5] y con esfuerzos residuales en el rango de 400MPa [6]. Las películas de poly-Si dopadas n^{++} por difusión (principalmente con fósforo) a temperaturas superiores a 900°C, crecen con una orientación preferente $\langle 111 \rangle$, con esfuerzos residuales menores, sin embargo el tamaño de grano

incrementa considerablemente [5]. Los esfuerzos residuales influyen negativamente en el comportamiento mecánico de las microestructuras, siendo este tema crucial en los procesos de fabricación relacionados [7].

II. DESARROLLO EXPERIMENTAL

El proceso de fabricación utiliza cinco materiales, y nueve niveles de mascarillas. Los materiales se listan en la Tabla 1, en la última columna se especifica la técnica de obtención de cada material.

1)	Dióxido de Silicio	SiO ₂	Térmico
2)	Nitruro de Silicio	Si ₃ N ₄	LPCVD
3)	Silicio Policristalino	p-Si	LPCVD
4)	Vidrio de fosfosilicato	PSG	APCVD
5)	Aluminio	Al	PVD

Tabla 1: Materiales utilizados en el proceso de fabricación, se indica su método de obtención.

El patrón geométrico del circuito (chip) de pruebas PolyMEMS INAOE se muestra en la Figura 1.

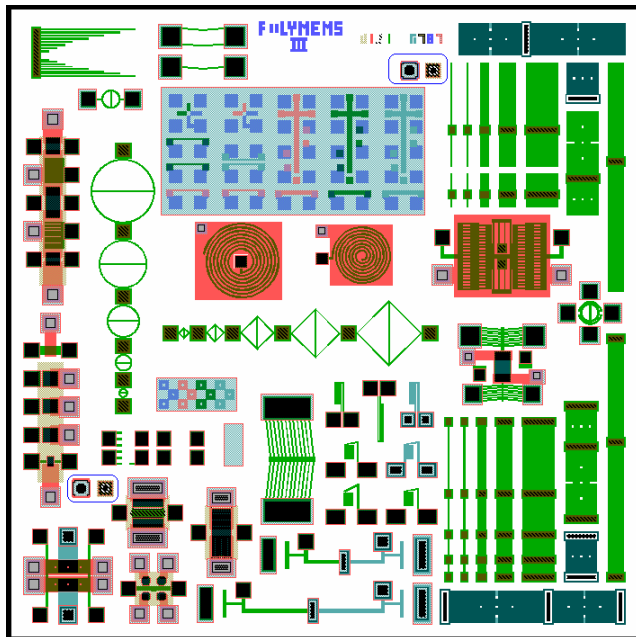


Figura 1: Patrón geométrico del circuito de pruebas PolyMEMS III del INAOE.

Descripción de las Microestructuras

Las microestructuras tienen dimensiones mínimas de 5µm, y máximas de 800µm. Los monitores de esfuerzos mecánicos residuales son estructuras tipo Puente, Anillo, Diamante y tipo Vernier. Los Puentes, Anillos y Diamantes, fueron diseñados con diferentes dimensiones con el objetivo de obtener condiciones críticas de operación con precisión [7]. Los monitores de gradientes de esfuerzo son estructuras tipo Trampolín y Espirales. Algunos sensores/actuadores incluidos son de naturaleza electrostática, tal como los *microspejos* y *resonadores*. Otros sensores/actuadores pero de naturaleza electrotérmica son las estructuras dinámicas de lazo sencillo *STA* (Single Two Arms) ó *Pseudo-biomorfo*, y *Chevron* de 1, 2 y 12 elementos. El chip de pruebas contiene estructuras de propósito general como *monitores ópticos*, que sirven para verificar la fidelidad en la transferencia de los patrones geométricos en la resina y con ello monitorear estrechamente el proceso de grabado de los materiales. En la Tabla 2 se presenta un resumen de cada una de las estructuras del circuito, se indica su función y el modo de operación.

Estructura	Función	Modo de Operación
Trampolín	Determina gradientes de esfuerzo residual (+/-) en el eje z.	Mecánico
Puente	Determina esfuerzos residuales de estiramiento en el plano x, y.	Mecánico
Espiral	Determina gradientes de esfuerzo residuales en el eje z.	Mecánico
Anillo & Diamante	Determina esfuerzos residuales de estiramiento en el plano x, y.	Mecánico
Indicador Vernier	Determina esfuerzos residuales de estiramiento o compresión.	Mecánico
Resonador	Generar un movimiento mecánico oscilatorio, o cambios de Aceleración (sensor y/o actuador).	Electrostático
Espejo	Deflectar un haz de luz (interruptor óptico).	Electrostático
STA / Chevron	Generar un movimiento mecánico modulado por una corriente eléctrica.	Electrotérmico

Tabla 2: Descripción de las estructuras del circuito PolyMEMS – INAOE, se indica su función y modo de operación.

El proceso de Fabricación completo del circuito PolyMEMS se describe de manera general con los siguientes pasos:

- 1) Crecimiento de SiO₂ (Térmico)
- 2) Depósito de Si₃N₄ (LPCVD)
- 3) Depósito & Dopado de poly-Si 1 (LPCVD)
- 4) Litografía & Grabado de poly-Si 1.
- 5) Depósito, Reflujo, & Litografía de PSG 1 (APCVD)
- 6) Depósito & Dopado de poly-Si 2
- 7) Litografía & Grabado de poly-Si 2
- 8) Depósito, Reflujo, & Litografía de PSG 2
- 9) Depósito & Dopado de poly-Si 3
- 10) Litografía & Grabado de poly-Si 3
- 11) Depósito, Reflujo, & Litografía de PSG 3
- 12) Depósito & Litografía Aluminio
- 13) Liberación mecánica, y Aleación de Aluminio

Los materiales se obtuvieron mediante las siguientes técnicas: PSG, depósito químico en fase vapor (CVD); Polisilicio, CVD a baja presión (LPCVD); Si₃N₄, CVD a presión atmosférica (APCVD) y Aluminio, evaporación física (PVD). La litografía se realizó mediante una alineadora de contacto utilizando luz ultravioleta. El grabado se realizó por técnicas de grabado húmedo y seco. En la Tabla 3 se listan las técnicas y condiciones de grabado utilizadas en cada material, su razón de grabado, y el material enmascarante. Las películas de SiO₂ y Si₃N₄ funcionan como aislantes (eléctricos y térmicos), por lo que no se efectúa proceso litográfico alguno sobre estos materiales.

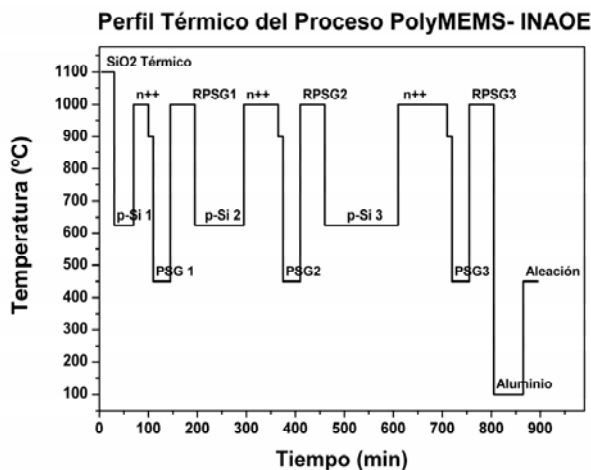


Figura 2: Perfil del proceso PolyMEMS-INAOE.

En la Figura 2 se muestra el perfil térmico del proceso de fabricación; se muestra la variación de la temperatura en los distintos pasos del proceso, en donde n⁺⁺ indica el proceso de dopado de poly-Si.

Material	Condiciones de Grabado	Enmascarante	Razón de Grabado [nm/min]
poly-Si	KOH 45% wt.	SiO ₂ 800Å	225
PSG	(CH ₃ COOH: NH ₄ F: H ₂ O) 33%.	a) Resina (-)* b) Resina (+)**	625
Al	Al-etch	Resina (+)	600

Tabla 3: Técnicas y condiciones de grabado para los materiales y la razón de grabado. En la columna 3 se especifican los materiales enmascarantes.

* Para Apertura de ventanas, ** Para Liberación mecánica.

Liberación Mecánica de las Estructuras

Para la liberación mecánica de las estructuras se utiliza una solución ácido acético-fluoruro de amonio-agua, (CH₃COOH:NH₄F:H₂O)33% [8] a 50°C, por su selectividad respecto a la resina y aluminio, 500:40, que presenta mejores grabados respecto a aquellas soluciones a base de ácido fluorhídrico (HF). Esta solución sin HF permite utilizar tiempos de grabado relativamente cortos, disminuyendo el número de etapas de recocido de la resina y sin afectar al aluminio, o incluso, realizar con las zonas metalizadas expuestas al grabante.

Caracterización de las Microestructuras

Rugosidad: la rugosidad de las películas de poly-Si incrementa por diversos factores: 1) temperaturas de depósito superiores a 650°C, 2) al incrementar la presión parcial del gas precursor (SiH₄); c) al incrementar la temperatura de los tratamientos térmicos subsecuentes tales como oxidación/dopado del polisilicio y reflujo del PSG [9].

En los procesos de fabricación de CMOS se utilizan temperaturas típicas de (1000 y 1100)°C [10] para el reflujo del PSG (RPSG) conteniendo (6-8)%wt. P [11]. Por lo anterior se realizó un análisis de la rugosidad de la superficie de las estructuras fabricadas en poly-Si utilizando ambas temperaturas. El estudio se realizó mediante un microscopio de fuerza atómica (AFM: *Atomic Force Microscope*), y su efecto sobre la definición de los bordes mediante un microscopio de barrido de electrones (SEM: *Scanning Electron Microscope*).

En la Figura 3 (a y b) se muestran los resultados obtenidos en procesos con T=1100°C, la rugosidad RMS medida es ~76.5nm. En la Figura 3 (c y d) se muestran resultados del análisis de la morfología en procesos con T=1000°C, la rugosidad RMS medida es ~9.1nm. En la estructura de la Figura 3a, se puede observar la pobre definición de los bordes. En la Figura 4b se observa una mejor definición en los bordes de las pistas de polisilicio, como resultado de reducir la temperatura del proceso RPSG.

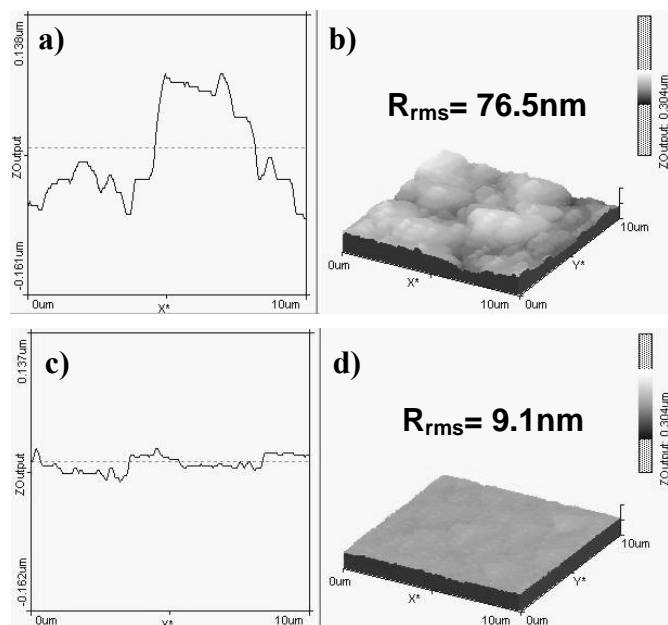


Figura 3: Micrografías obtenidas por AFM, a) vista lineal, b) vista tridimensional sobre una estructura obtenida con T=1100°C; c) vista lineal, d) vista tridimensional de una estructura obtenida con T=1000°C.

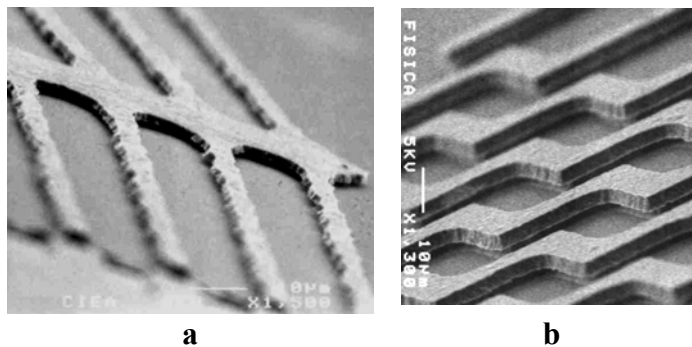


Figura 4: Micrografías obtenidas por SEM, a) de una estructura obtenida con T=1100°C; b) de una estructura obtenida con T=1000°C.

Del análisis superficial de las muestras presentadas en las Figuras 3 y 4, se pudo determinar una reducción de la rugosidad en un 88%. El reducir la rugosidad y mejorar la definición de los bordes en las pistas de polisilicio, implica un incremento en la resistencia a la deformación [3].

En las Figuras 5, 6, y 7 se muestran fotografías obtenidas por SEM de algunas microestructuras suspendidas fabricadas con el proceso PolyMEMS del INAOE.

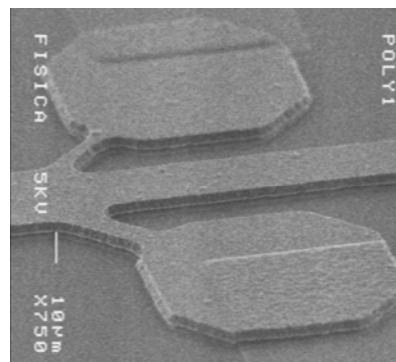


Figura 5: Microespejos Electrostáticos de flexión.

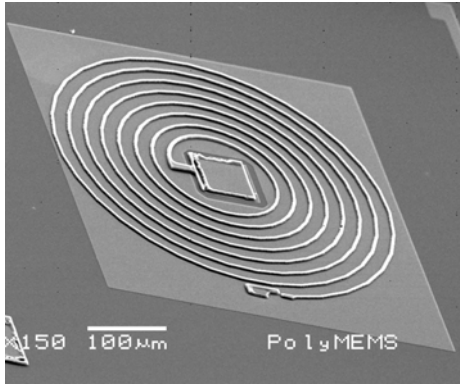


Figura 6: Estructura tipo Espiral, monitor de gradientes de esfuerzo.

Análisis sobre los Esfuerzos Residuales

La presencia involuntaria de defectos en el interior de los materiales, origina deformaciones mecánicas en su estructura; estos defectos pueden ser: impurezas, ausencia de átomos (*vacancias*) en la estructura del material, o por la presencia de átomos en posiciones intermoleculares (*intersticiales*). Los defectos inducen esfuerzos mecánicos [5] que pueden disminuir el rendimiento de las estructuras. Estos defectos se pueden clasificar en dos grandes grupos: 1) *Esfuerzos de Compresión* ($-\sigma_r$), los cuales tienden a incrementar el volumen (superficie, longitud) de las estructuras, y 2) *Esfuerzos de Estiramiento* ($+\sigma_r$), los cuales tienden a disminuir el volumen de las estructuras [12].

Se determinaron los esfuerzos residuales en las películas de poly-Si utilizando las estructuras de pruebas mecánicas *Puente* y *Anillo*. Los esfuerzos residuales se calcularon mediante la Ecuación 1, en donde E es el módulo de elasticidad o Young, $E=160GP$ [13]. La deformación mecánica (ϵ) para un puente está dada por la Ecuación 2, en donde z es el grosor de la película de poly-Si ($2.5\mu m$). La deformación para un anillo está dada por la Ecuación 3, en donde $f_{(R)}$ es una función del anillo ($f_{(R)}=0.2$) [12, 14].

$$\sigma_r = \epsilon E \tag{1}$$

$$\epsilon = \frac{\pi^2 z^2}{3L_{cr}^2} \tag{2}$$

$$\epsilon = \frac{\pi^2 z^2}{48 f_{(R)} R_c^2} \tag{3}$$

En la Figura 7 se muestra un puente con una *longitud crítica* $L_c=400\mu m$, y en la Figura 8 se muestra una estructura tipo anillo con un *radio crítico* $R_c=200\mu m$. A partir de estas dimensiones críticas de deformación se calcularon los esfuerzos residuales de compresión para las estructuras tipo puente y anillo, resultando $-20.6MPa$ y $-25.7MPa$ respectivamente. Una magnitud considerada como “cero” Esfuerzo Residual es $\sigma_r < |5|MPa$ [1].

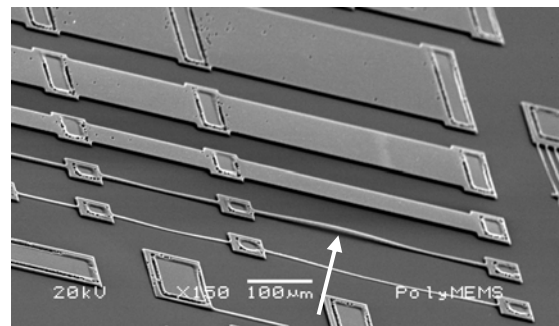


Figura 7 Estructura tipo Puente, fabricada con una longitud crítica $L_{cr}=400\mu m$.

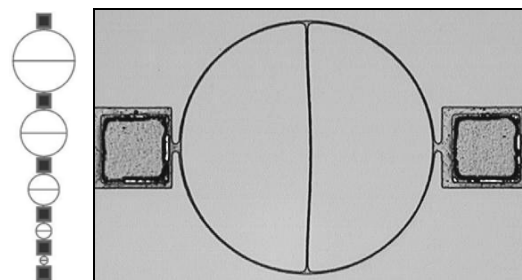


Figura 8: Serie de estructuras tipo Anillo (izquierda), y (derecha) Anillo de diámetro mayor mostrando un radio crítico de deformación $R_c=200\mu m$.

Análisis sobre los Gradientes de Esfuerzo

Los gradientes de esfuerzo son variaciones en los esfuerzos de compresión o estiramiento, en dirección normal al crecimiento de la película [6]. Se observó cualitativamente mediante microscopía óptica, utilizando estructuras de pruebas tipo espiral, la dependencia de los gradientes de esfuerzo con la temperatura. En la Figura 8 se muestra la vista superior de una estructura tipo espiral obtenida de un proceso a 1100°C, la cual presenta gradientes de esfuerzo positivos ($+\Delta\sigma/z$). Por otra parte, en la Figura 9 se muestra un espiral suspendido sin gradientes de esfuerzo. Con este segundo análisis también se concluye que la temperatura de 1100 °C no es apropiada para el proceso de fabricación de las microestructuras de polisilicio.

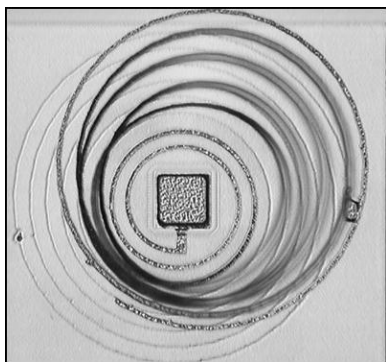


Figura 8: Estructura tipo Espiral de un proceso a 1100°C, se observa una deformación debida a los gradientes de esfuerzo positivos.

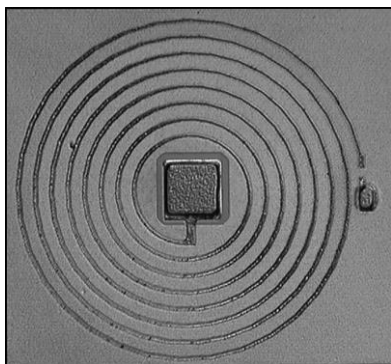


Figura 9: Estructura tipo Espiral de un proceso a 1000°C sin gradientes de esfuerzo.

III. CONCLUSIONES

Se ha presentado un proceso de fabricación de mecanismos y estructuras mecánicamente estables, con dimensiones de $5\mu\text{m} - 800\mu\text{m}$ compatible con procesos de fabricación de CI's convencionales como CMOS, Bipolar, o BiCMOS. El proceso PolyMEMS – INAOE es un proceso de fabricación diseñado para una integración monolítica *intra-proceso*, con la posibilidad de utilizar líneas de interconexión eléctrica de poly-Si n^{++} .

La razón de aspecto (5:3) de las geometrías permite la fabricación de una serie de sensores/actuadores con deflexiones en el plano perpendicular a la superficie de la oblea, *i.e.* *microespejos de flexión*.

Los esfuerzos residuales de las películas de poly-Si disminuyen al incrementar la temperatura, no obstante el tamaño de grano aumenta considerablemente, y como consecuencia una pobre definición de las geometrías. Los gradientes de esfuerzo por otra parte, incrementan con la temperatura y tiempo de los tratamientos térmicos subsecuentes.

Los procesos con $T=1000^\circ\text{C}$ presentan mejores resultados en cuanto a definición de los bordes de las geometrías, exhibiendo esfuerzos residuales y gradientes de esfuerzo mínimos. El proceso RPSG desarrollado a $T=1000^\circ\text{C}$ no influyó de manera negativa en el proceso.

IV. AGRADECIMIENTOS

Los autores agradecen a los técnicos del laboratorio de Microelectrónica del INAOE por su colaboración en el proceso de fabricación. A *Alicia del Real* encargada de Microscopía Electrónica del Centro de Física Aplicada y Tecnología Avanzada (CFATA) por las microfotografías (Figuras 6 y 7).

V. REFERENCIAS

- [1] K. H. Bushow, W. R. Cahn, M. C. Flemings, B. Ilshner, J. Kramer, S. E. Mahajan, 2001, “*Encyclopedia of Materials: Science & Technology*”, Vol. 8, Elsevier, pp 7724-7735.
- [2] Ted. Kamins, 1998, *Polycrystalline Silicon For Integrated Circuits & Displays*, Kluwer Academia Publishers, pp 97-98, 311, 313-314.
- [3] P.L. Mangonon, 1999, “*The Principles of Materials Selection for Engineering Design*”, Prentice Hall, Pp 135-136.
- [4] J. F. Shackelford, 2000, “*Introduction for Materials Science for Engineers*”, 5th Ed, Prentice Hall, Pp 212-213.
- [5] G. H. Mohamed, 2006, “*MEMS: Design and Fabrication*”, CRC, 3-114, 3-165-168.
- [6] S. Sedky, 2006, “*Post-Processing Techniques for Integrated MEMS*”, Artech House, pp 70, 121.
- [7] D. D. Alonso, 2008, *Tesis de Licenciatura: “Estudio de los Mecanismos de Colapso en Microactuadores y su influencia en Microsistemas”*, INAOE.
- [8] J. F. L. Goosen, et al. “*Problems of sacrificial etching in the presence of aluminium interconnect*”, 1997, *Sensors and Actuators A62*, pp 692-697.
- [9] S. Wolf, 2004, “*Microchip Manufacturing*”, Lattice Press, pp 113, 289.
- [10] S. M. SZE, 1987, “*VLSI Technology*”, McGraw-Hill, pp 113-115.
- [11] B. CLARK, 1977, “*Phosphorus Concentration in Low Temperature Vapor Deposited Oxide*”, *Solid State Technology*.
- [12] J. G. Korvink, 2006, “*MEMS: A practical Guide to Design, Analysis and Applications*”, William Andrew Publishing, pp 51, 56-57, 77-78.
- [13] French, P. J., 2002, “*Polysilicon: a versatil material for Microsystems*”, *Sensors and Actuators A 99*, pp 7.
- [14] J. Laconte, D. Flandre, and J.-P. Raskin, “*Micromachined Thin-Film Sensors for SOI-CMOS Co-Integration*”, 2006, Springer, pp 71.

ACONDICIONAMIENTO DE SEÑAL PARA UN SENSOR DE PRESIÓN MEMS

Montiel Uribe Rodrigo, Navarro Alvarado J. Marcos, Sosa Torres Miguel Ángel.

Centro de Investigación en Micro y Nano Tecnología (CIDEMyNT)
Instituto Tecnológico Superior de Irapuato (ITESI)
Carr. Irapuato-Silao Km. 12.5, Irapuato, Guanajuato, México, C.P. 36821
Tel. 01 (462) 6067900, Ext. 146, Fax. Ext.105, Apdo. Postal 146
Correo electrónico: romontiel@itesi.edu.mx, misosa@itesi.edu.mx, jonavarro@itesi.edu.mx,
www.itesi.edu.mx

RESUMEN

Este trabajo está orientado hacia el diseño de un sistema de acondicionamiento de señales para un sensor de presión capacitivo basado en MEMS (*MicroElectroMechanical System*, Sistema microelectromecánico). MEMS es la tecnología que permite integrar sobre un mismo sustrato de silicio la parte electrónica como son los sensores, actuadores y diversos elementos mecánicos; con la particularidad de que los elementos mecánicos están en la escala de micras. También se presenta la propuesta de un sensor capacitivo de tipo membrana que se realizó con el proceso de diseño PolyMUMS, con el fin de tener una referencia de las variaciones de capacitancia que se pueden tener en este tipo de sensores de presión basados en MEMS. Se muestran en la parte final de este artículo resultados de la simulación del acondicionamiento de señal.

Palabras clave: Acondicionamiento de señal, MEMS, Sensor capacitivo, PolyMUMS.

I. INTRODUCCIÓN

El hombre siempre ha tenido la necesidad de medir, ya que percibe la información del mundo que le rodea a través de sus sentidos y adquiere el conocimiento, sobre todo el científico, cuando es capaz de cualificar y/o cuantificar las magnitudes que percibe, es decir, a través de la medida. La Instrumentación Electrónica es la técnica que se ocupa de la medición de cualquier magnitud física, de la conversión de la misma a magnitudes eléctricas y de su tratamiento para proporcionar la información adecuada a un operador

(visualización), a un sistema de control o a ambos. El sector industrial demanda dispositivos electrónicos cada vez más confiables que puedan ayudar a medir variables físicas de interés.

Un sistema de medida consiste de un elemento sensor, un acondicionador de señal y un sistema de visualización. El elemento sensor es aquel dispositivo en el que un cambio en una magnitud física proporciona un cambio de una propiedad eléctrica. La magnitud del cambio, así como la propiedad eléctrica que se ve afectada con el cambio en la magnitud de la variable física son muy variadas. El acondicionador de señal es aquel elemento que se encarga de convertir esa variación en una propiedad eléctrica y por consiguiente un cambio de esta, de tipo y magnitud adecuada para el elemento de visualización. El elemento de visualización se encarga de que esa señal eléctrica proveniente del acondicionador de señal sea entendible para un operador o para un sistema de control.

Por estas razones, el acondicionamiento de señal es una parte muy importante de un sistema de medida debido a que si no se realiza un buen acondicionamiento el sistema de medida no funcionará adecuadamente.

Problemática

Las llantas tienen una presión óptima de operación. Conducir con una llanta que tiene los niveles de presión fuera de rango (alto o bajo) puede acortar la vida útil de esta, debido a que el desgaste a una presión no-adeuada es mayor. En el peor de los casos una llanta que no opera a la

presión adecuada puede causar un accidente de consecuencias fatales.

En Estados Unidos entró en vigor una nueva ley que obliga a los vehículos de pasajeros a contar con un sistema de monitoreo de presión y temperatura de la llanta si viajan por una autopista. Además, los automóviles de lujo cuentan ya con un sistema de monitoreo de la presión de las llantas como parte de su equipamiento de serie.

Hasta ahora los equipos de monitoreo de presión de la llanta se montan sobre el rin, estos equipos son voluminosos y pueden interferir en el funcionamiento de esta debido a que pueden llegar a desbalancearla. El sensor de presión puede ser parte de la llanta si tiene dimensiones muy pequeñas, esto es posible con tecnología MEMS.

El trabajo de investigación que aquí se presenta es parte de un proyecto de mayor alcance, el cual tiene como objetivo crear un sistema de monitoreo para las llantas de vehículos de autotransporte. Las variables que se pretende monitorear son: Presión, temperatura, kilometraje, en el cual cada llanta tendrá un código de identificación único. La información será transmitida con tecnología RFID a una computadora central que almacenará los datos obtenidos.

Un sensor no puede funcionar por sí solo, requiere de varias etapas para que pueda convertirse en un sistema de medida. Parte importante de este sistema es el acondicionamiento de señal del elemento sensor, y más aún si es de tipo pasivo.

II. MÉTODO

Capacitancia

Los capacitores están diseñados de manera que la carga se almacene en un material polarizado entre dos conductores. La carga que se almacene dependerá de la capacitancia, que a su vez depende del diseño del capacitor y el dieléctrico utilizado.

El material entre los conductores debe polarizarse con facilidad, para que la constante dieléctrica sea alta, y al mismo tiempo, tenga una resistividad eléctrica elevada, para impedir que la carga pase de una placa a la otra. La aplicación de un campo eléctrico genera la formación y el movimiento de dipolos contenidos en el material. Los dipolos son átomos o grupos de átomos que tienen carga desequilibrada. En un campo eléctrico aplicado, los dipolos se alinean causando polarización. Cuando se aplica un campo eléctrico a un material, se induce la creación de dipolos dentro de la estructura atómica o molecular, la cual se alinea en la dirección del campo, la alineación con el campo de los dipolos permanentes presentes en el material generan la polarización. La polarización es causada por cuatro mecanismos: (1) la polarización electrónica, (2) la polarización iónica, (3) la polarización molecular y (4) las cargas espaciales.

Sensor tipo membrana

Los principios de sensado de presión más comunes son: el de membrana piezorresistiva y el capacitivo. Aunque también existen los sensores de presión de diamante policristalino y sensores de presión de fibra óptica. La sensibilidad de un sensor de presión capacitivo es mucho mayor que la del sensor piezorresistivo, aproximadamente 10 veces mayor, y se encuentra en el rango de 1000 ppm/Torr. Estas propiedades muestran muchas ventajas, aunque su principal desventaja es la no linealidad y el rango dinámico.

Las características esenciales de la mayoría de los sensores de presión micromaquinados es un diafragma soportado por los bordes que se deforma en respuesta a una presión diferencial transversal a través de él. Esta deformación es detectada típicamente midiendo los esfuerzos en el diafragma, o midiendo el desplazamiento del diafragma.

La deflexión de un diafragma y su esfuerzo asociado se puede calcular analíticamente en muchos casos. Generalmente, es útil hacer algunas

suposiciones con el fin de simplificar prestando atención a las dimensiones y a las condiciones límite. Una aproximación es suponer que los bordes están simplemente soportados. Esta es una suposición razonable si el grosor del diafragma, h , es más pequeño que su radio, a . Esta condición previene desplazamientos transversales de la superficie neutral en el perímetro, mientras permite un desplazamiento rotacional y longitudinal. Matemáticamente, esto permite que la segunda derivada de la deflexión sea cero en los bordes del diafragma. Sin embargo, la suposición preferida es que los bordes del diafragma están rígidamente fijados al soporte alrededor de su perímetro. Bajo esta suposición el estrés sobre la superficie inferior de un diafragma circular puede ser expresado en coordenadas polares por las ecuaciones:

$$\sigma_r = \frac{3 \cdot \Delta P}{8h^2} [a^2(1+\nu) - r^2(3+\nu)] \quad (1)$$

$$\sigma_t = \frac{3 \cdot \Delta P}{8h^2} [a^2(1+\nu) - r^2(1+3\nu)] \quad (2)$$

donde σ_r denota la componente radial y σ_t la componente tangencial, a el radio y h el grosor del diafragma, r es la coordenada radial, ΔP es la presión aplicada en la superficie superior del diafragma y ν es la razón de Poisson.

Estas ecuaciones indican que las componentes del esfuerzo varían desde una tensión máxima en el centro del diafragma hasta una compresión máxima en la periferia. Ambas componentes son cero en valores separados de r entre cero y a . En general, piezoresistores en los puntos de mayor esfuerzo en compresión y tensión proveerán la mayor respuesta. La deflexión de un diafragma circular bajo los estados supuestos está dado por:

$$d = \frac{3 \cdot \Delta P(1+\nu^2)(a^2 - r^2)^2}{16Eh^3} \quad (3)$$

donde E es el modulo de Young del material estructural. Esto es válido para un diafragma delgado simplemente soportado por los bordes, suponiendo una deflexión pequeña.

En la Ecuación [3] se puede notar que la distancia entre las placas es directamente proporcional al cambio de presión aplicado lo que da grandes posibilidades de tener un sistema de medida lineal.

Para un capacitor de placas paralelas de área A , y separadas por una distancia d , la capacitancia C está dada por

$$C = \frac{\epsilon_0 \epsilon_r A}{d}, \quad (4)$$

donde $\epsilon_0 = 8.85$ pF/m es la permitividad del espacio libre y ϵ_r es la permitividad relativa del material dieléctrico.

De esta manera, cualquier fenómeno o magnitud que produzca una variación en ϵ_r , A o d provocará un cambio en la capacitancia C .

El sensor que se propone para medir la presión es un sensor capacitivo de membrana, como el que se muestra en la Figura 1. Entre las aplicaciones más inmediatas de los sensores capacitivos, están la medida de desplazamientos lineales y angulares y los detectores de proximidad.

Los sensores de presión capacitivos con diafragma de silicio emplean diafragmas más finos que los equivalentes piezoresistivos y por ello pueden medir presiones menores. En la Figura 1, se muestra la propuesta para el sensor.

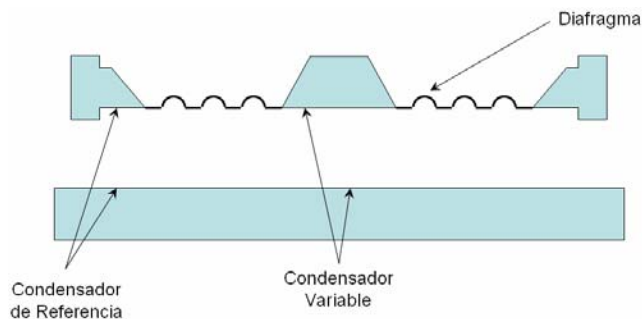


Figura 1. Sensor de presión capacitivo de tipo membrana.

Como en un sensor capacitivo de diafragma la distancia entre las placas no es uniforme cuando

se flexiona el diafragma, comúnmente se utiliza el análisis de elemento finito para calcular la respuesta de un sensor de presión capacitivo. Sin embargo se puede demostrar que para una deflexión que es más pequeña comparada al grosor del diafragma, la sensibilidad, definida como el cambio fraccional en la capacitancia por unidad de cambio en la presión, tiene la siguiente proporcionalidad:

$$S_{cap} = \frac{\Delta C}{C \cdot \Delta P} = \frac{1}{PA d} \int_A w dA = 0.0746 \frac{1-\nu^2}{E} \frac{a^4}{h^3 d} \quad (5)$$

donde d es la distancia nominal entre el diafragma y el electrodo y las demás variables son las mismas que se han utilizado anteriormente.

El sensor es de tipo pasivo debido a que no genera energía por si mismo, este sensor cambia sus propiedades eléctricas en respuesta a una deformación en el diafragma, para este caso en particular el cambio en la distancia entre el diafragma y el electrodo fijo provoca un cambio de capacitancia. La capacitancia de un sensor MEMS puede variar en un rango comprendido entre 0.8 y 38 pF. La capacitancia de un capacitor de placas paralelas de área, A , separadas por una distancia, d , y con un material dieléctrico de permitividad relativa ϵ_r , está dado por la Ecuación [4].

La capacitancia de un capacitor depende solo de sus dimensiones y del material dieléctrico entre las placas. Por estas razones el sensor propuesto es una membrana de 1 mm de diámetro de polysilicio, el material dieléctrico será el vacío, de acuerdo a las reglas de diseño de PolyMUMPS, la distancia entre las placas será de 5 a 1 μm debido al grosor de las capas, tal como se muestra en la Figura 2. Se consideró una membrana en PolyMUMPS solo como referencia en cuanto a las dimensiones, sin embargo, es posible que este no sea el diseño de sensor final, el cual se está trabajando en otro proyecto de investigación.

El diseño de la membrana se realizó siguiendo las reglas de diseño que MEMSCAP

tiene para el estándar PolyMUMPs, usando el módulo L-EDIT del software MEMS pro V4.0. El resultado del diseño se muestra en la Figura 3 y en la Figura 4 se muestra una vista 3D del diseño de la membrana.

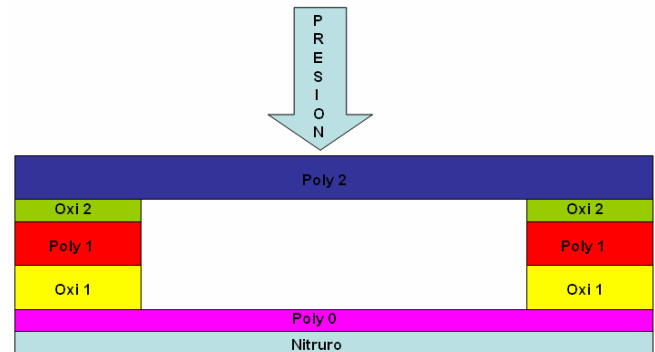


Figura 2.- Espesor de las capas en un dispositivo MEMS en el proceso PolyMUMS.

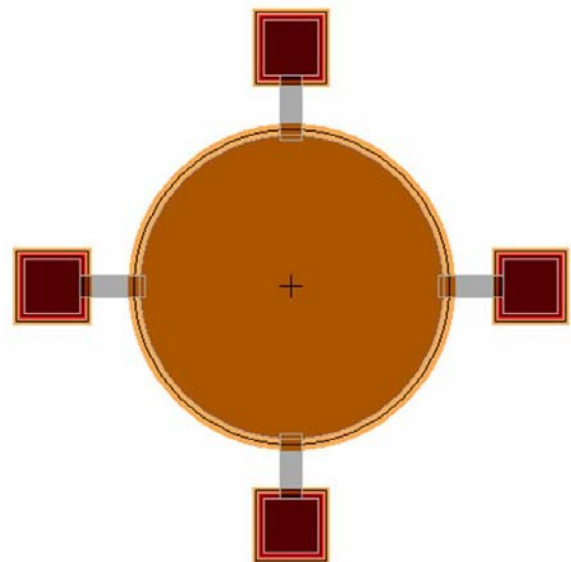


Figura 3.- Diseño de la membrana en PolyMUMPS, vista superior en L-Edit.

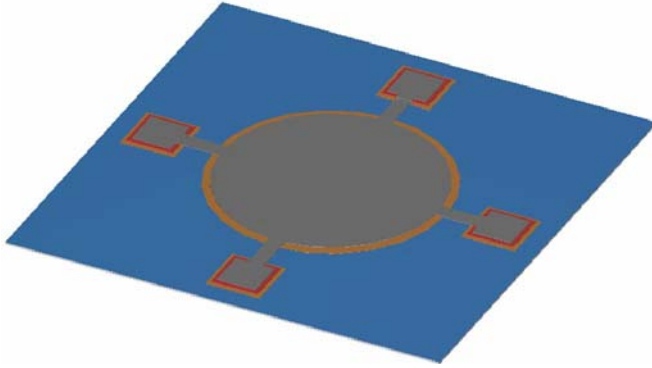


Figura 4.- Diseño de la membrana en PolyMUMPS, vista 3D.

Considerando estas dimensiones de la membrana, y utilizando la Ecuación [4], donde el área sería de $A=\pi(0.0005^2)=0.7854 \times 10^{-6} \text{ m}^2$, la permitividad relativa del espacio libre es de $\epsilon_0 = 8.85 \text{ pF/m}$, por lo tanto, la capacitancia máxima se da cuando la distancia es mínima, $1 \mu\text{m}$ y está dada por:

$$C = \frac{8.85 \times 10^{-12} (0.7854 \times 10^{-6})}{10^{-6}} = 6.95 \text{ pF} \quad (6)$$

y la capacitancia mínima se presenta cuando la distancia es máxima y es

$$C = \frac{8.85 \times 10^{-12} (0.7854 \times 10^{-6})}{5 \times 10^{-6}} = 1.39 \text{ pF} \quad (7)$$

Con este diseño se da una idea clara de los cambios de capacitancia que puede presentar un sensor de presión de tipo membrana de dimensiones tan minúsculas.

Circuito de Acondicionamiento propuesto

Para obtener una señal útil a partir de una capacitancia es necesario disponer de una fuente de tensión alterna para alimentar el componente y de un medio para detectar las variaciones producidas en respuesta a la magnitud detectada. En este caso se ha de utilizar un convertidor A/D, la señal entregada a la entrada de este debe ser además continua y con un margen de valores normalizado.

Para los sensores capacitivos, las capacidades que se manejan son inferiores a los 100 pF. Esto obliga a trabajar a frecuencias altas para tener impedancias manejables, pero no suele pasar de 100 MHz ni de menos de 10 kHz. Al ser la impedancia de salida alta, conviene que los cables de conexión estén apantallados. Esto equivale a añadir una capacitancia en paralelo con el capacitor, lo que trae como consecuencia pérdida de sensibilidad y deterioro de la linealidad. Si además hay un movimiento relativo entre los conductores del cable y el dieléctrico entre ellos, el error puede ser muy grave. La solución consiste en poner la electrónica cerca, con cables cortos, incluso rígidos y emplear técnicas de guarda activa o transformadores de impedancia. Para nuestro caso se pretende que el sensor y la microelectrónica del acondicionador estén integrados en el mismo microcircuito, de tal manera que se minimizarán estos efectos nocivos.

En el circuito de la Figura 5, denominado amplificador de carga, se excita el sensor con una fuente de corriente alterna constante, y la corriente que resulta se convierte en una tensión mediante el amplificador operacional.

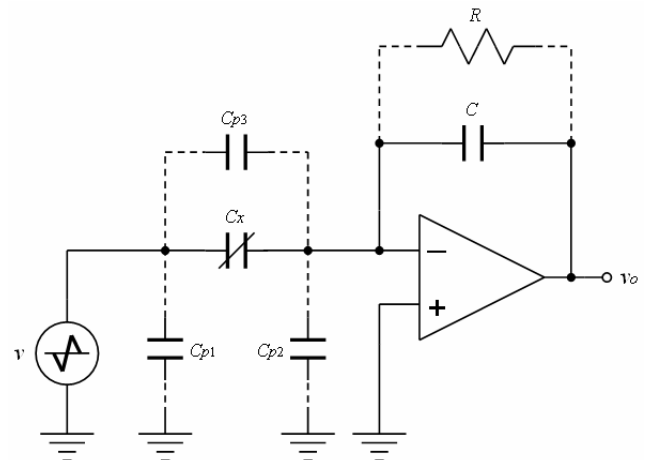


Figura 5.- Circuito inmune a capacitancias parásitas.

En este circuito, el voltaje de salida del circuito es lineal con respecto al cambio de capacitancia C_x debida a las variaciones en la distancia entre las placas.

$$v_0 = -v \frac{C_x}{C} \quad (8)$$

La capacitancia C puede ser un capacitor de referencia externo o puede ser otra membrana fabricada en el mismo dispositivo, pero que no está sometida a la presión, por lo tanto, ésta se mantiene constante, a este tipo de sensores se les denomina pasivos, ya que solo sirven de referencia. La capacitancia parásita C_{p3} se puede reducir apantallando los cables de conexión del sensor y por eso se ha considerado que no influye en v_0 . Las capacitancias parásitas C_{p1} y C_{p2} , debidas a los cables de conexión del sensor, no influyen la medida, aunque una C_{p2} muy alta puede hacer que el circuito oscile. Un inconveniente es que la salida es inversamente proporcional a la distancia entre las placas de C_x , pero con ese método se han medido capacitancias de sólo 0.03 pF.

El acondicionador de señal que se muestra en la Figura 5 presenta linealidad entre la capacitancia y el voltaje de salida, sin embargo, el voltaje de salida es no lineal con respecto a los cambios de distancia entre las placas.

Si invertimos el capacitor C con el capacitor C_x en el circuito de la Figura 5, resulta el circuito de la Figura 6, aquí, el voltaje de salida y la distancia entre las placas del sensor capacitivo guardan una relación lineal, esto se puede notar en la siguiente ecuación:

$$v_0 = -v \frac{Z_x}{Z} = -v \frac{C}{C_0} (1 + x) \quad (9)$$

donde x es la variación de la distancia entre las placas.

El circuito de la Figura 6, será el circuito idóneo para nuestra aplicación, ya que cumple con los requisitos de linealidad entre voltaje de salida y distancia entre las placas. Además, en la Ecuación [3] se vio que bajo ciertas restricciones, la distancia entre las placas y la presión aplicada guardan una relación lineal. Por lo tanto, habrá una

relación lineal entre el voltaje de salida del circuito de la Figura 6 y la presión aplicada.

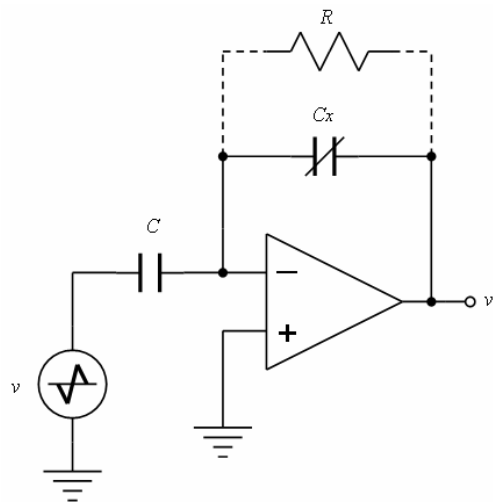


Figura 6.- Circuito linealizador.

La siguiente etapa a considerar es la conversión de una señal de corriente alterna, ca, a una señal de corriente directa, cd. Para esto existen muchas posibilidades, aunque se consideró un circuito rectificador de precisión con amplificadores operacionales, tal como se muestra en la Figura 7.

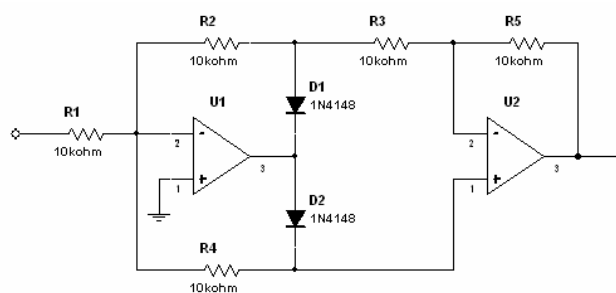


Figura 7.- Circuito rectificador de precisión.

Este circuito convierte una señal alterna en directa. La primera etapa es un rectificador lineal de media onda con salida negativa, la segunda etapa es solo un amplificador inversor, al circuito completo se le llama rectificador de precisión o circuito de valor absoluto. Este circuito transmite una polaridad de la señal de entrada e invierte la otra. Por lo tanto, ambos medios ciclos de un

voltaje alterno se transmiten pero se convierten en una polaridad única a la salida del circuito. El rectificador de precisión de onda completa puede rectificar voltajes de entrada con amplitudes de milivolts. Después de esta etapa puede colocarse un filtro pasa bajas tipo Butterworth de 40 dB/década, para obtener una señal de cd pura.

El nivel y rango del voltaje de salida después del filtro puede no ser adecuado para una etapa de postprocesamiento, un sistema de visualización o un convertidor analógico a digital. En nuestro caso, se desea que la señal sea adecuada para un convertidor analógico a digital (ADC, por sus siglas en inglés). El nivel de voltaje requerido para un ADC generalmente está normalizado y en general está en un rango de 0 a 5 V. Para poder cambiar el rango y el nivel de la señal se utiliza un amplificador de instrumentación. El amplificador de instrumentación es de los más útiles, precisos y versátiles disponibles en la actualidad. La mayoría de los sistemas de adquisición de datos los utilizan y están hechos con tres amplificadores operacionales y siete resistencias tal como se muestra en la Figura 8.

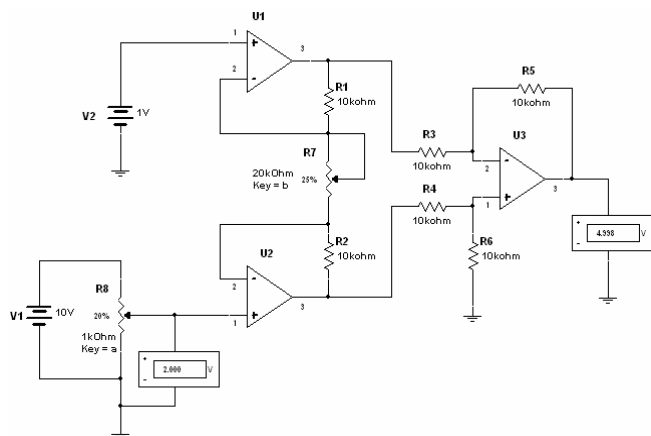


Figura 8.- Amplificador de Instrumentación.

Este circuito está construido con un diferenciador básico al que se la ha colocado en la entrada un amplificador de aislamiento. Este amplificado tiene una entrada diferencial, por lo que se pueden amplificar señales diferenciales o señales de simple terminación. La ganancia del

amplificador está determinada únicamente por la resistencia R_7 , por la relación,

$$V_o = (V_i - V_2) \left(1 + \frac{2}{a} \right) \quad (10)$$

donde V_i es el voltaje de entrada que resulta del divisor de voltaje formado por R_8 y V_2 , que para este caso $V_i = 2 \text{ V}$, y

$$a = \frac{aR}{R} = \frac{5\,000}{10\,000} = \frac{1}{2} \quad (11)$$

donde $V_2 = 1 \text{ V}$, por lo tanto, el voltaje de salida para la simulación que se presenta en la Figura 7, es de 5 V.

En el diseño del sensor se determinaron los valores probables para un sensor de presión por capacitancia de una membrana micromáquinada en el proceso de diseño PolyMUMPs. La capacitancia encontrada varía de 1.39 a 6.95 pF, cuando la distancia entre las placas varía de 5 a 1 μm respectivamente. La curva de capacitancia contra distancia entre las placas se muestra en la Figura 8.

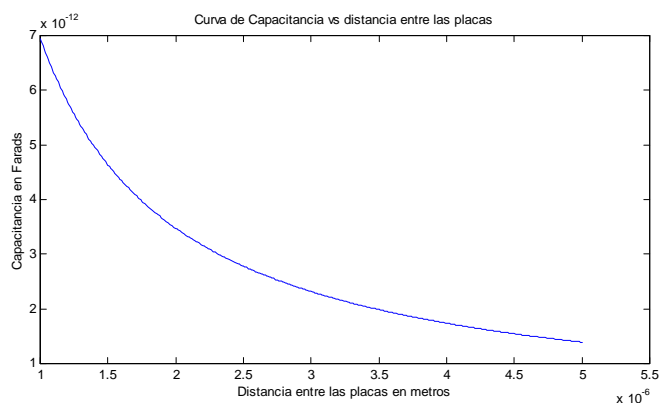


Figura 8.- Curva características de capacitancia vs distancia entre las placas.

Considerando los valores teóricos de capacitancia se realizó la simulación en Multisim 2001, la cual se muestra en la Figura 9. La señal aplicada es una señal senoidal de 10 V de amplitud y 100 kHz de frecuencia.

El capacitor C representa la capacitancia de referencia que en este caso lo suponemos de 10 pF y el capacitor C_x representa la capacitancia variable de la membrana, si lo suponemos de 7 pF, al 20 % aproximadamente tendrá la capacitancia mínima de la membrana y al 99 % la capacitancia máxima de la membrana. El resultado teórico, es de

$$v_0 = -v \frac{C}{C_x} = -10 \frac{1}{1.39} = -7.19 \text{ V} \quad (12)$$

para una capacitancia mínima y para una capacitancia máxima tenemos

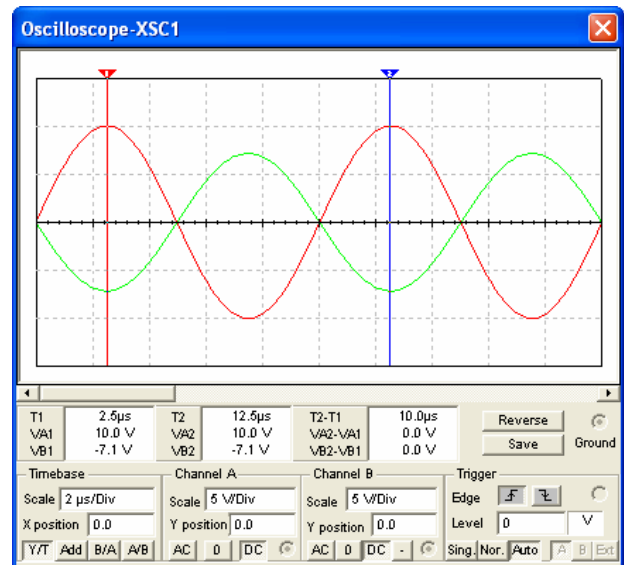
$$v_0 = -v \frac{C}{C_x} = -10 \frac{1}{6.95} = -1.44 \text{ V} \quad (13)$$

En la Figura 9 se pueden ver los resultados de la simulación para los valores extremos de la capacitancia. En la Figura 9(a) se muestra el voltaje de salida para una capacitancia de 1.39 μF y en la Figura 9(b) se puede observar el resultado de la simulación para la capacitancia de 6.95 μF .

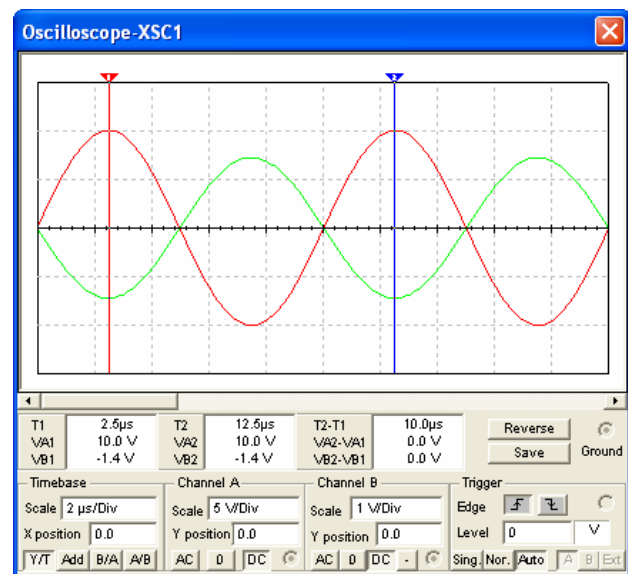
La curva de voltaje de salida contra distancia entre las placas es lineal, tal como se muestra en la Figura 10.

Se puede notar en los resultados de la simulación que, efectivamente, las consideraciones para este circuito linealizador se cumplen tal y como lo establecen los conceptos teórico. La variación de la distancia entre las placas es lineal con respecto al voltaje de salida. Esto nos indica una gran ventaja de este acondicionador de señal para ser considerado como una posibilidad en el acondicionamiento de señal de nuestro sensor MEMS.

Un circuito rectificador de onda completa a diodos tipo puente puede convertir una señal alterna en una señal de corriente continua, sin embargo, la caída de voltaje a través de los diodos puede darnos un error. La caída de voltaje a través de un diodo es de 0.7 V aproximadamente, debido al voltaje de umbral necesario para polarizar el diodo y pueda entrar en conducción.



(a)



(b)

Figura 9.- Resultados de la simulación para (a) Tensión de entrada salida para la capacitancia mínima y (b) Tensión de entrada salida para la capacitancia máxima.

Si en un puente rectificador de onda completa de tipo puente la señal debe pasar por dos diodos en serie, entonces, la caída de tensión es de 1.4 V aproximadamente. Esto es un inconveniente si se quiere rectificar una señal de corriente alterna cuya amplitud está variando, que es el caso de la

señal que se obtendrá del circuito linealizador debido a las variaciones en la capacitancia del sensor.

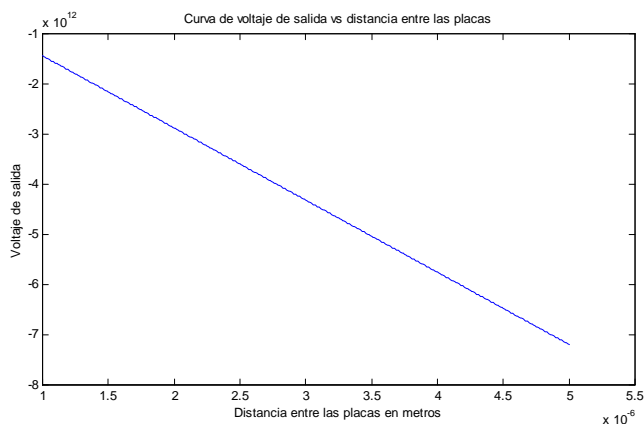


Figura 10.- Curva características de voltaje de salida vs distancia entre las placas.

La solución a este problema es el uso de un rectificador de precisión, el cual se presentó anteriormente. El circuito rectificador de precisión de la Figura 7 es el que se simuló. Los resultados de la simulación se muestran en la Figura 11.



Figura 10.- Resultados de la simulación del rectificador de precisión.

En el resultado de la simulación del circuito rectificador se puede ver que este circuito es capaz de rectificar una señal de 1 V con bastante

precisión. Por tanto es una opción adecuada para los fines que se persiguen.

III. CONCLUSIONES

Se ha presentado una propuesta de acondicionamiento de señal para un sensor basado en MEMS de capacitancia variable.

El circuito linealizador presenta una relación entre voltaje de salida y distancia entre las placas constante, lo cual es una gran ventaja de este circuito por sobre las demás opciones también analizadas. El acondicionamiento de señal para este tipo de sensores no está plenamente identificado, de tal forma que existen muchas alternativas cada una con sus ventajas y desventajas. Sin embargo, las simulaciones muestran que este es el indicado.

El rectificador de precisión resultó ser la mejor opción para el convertidor de ca a cd, debido a que este circuito entrega una tensión constante igual a la amplitud de la señal entregada por el circuito linealizador.

El Amplificador de Instrumentación es el amplificador más utilizado en los sistemas de adquisición de datos, su alta impedancia de entrada diferencial lo hacen ideal para este tipo de aplicaciones. En las simulaciones se comprobó su funcionamiento eficaz para sus fines.

Las simulaciones muestran que los circuitos propuestos, son los idóneos para la aplicación.

IV. REFERENCIAS

- [1] Bolton, W., *Instrumentación y Control Industrial*, (2ª Ed.), España: Paraninfo, (1999).
- [2] Nikolay V., Kirianaky; Sergey Y. Yurish; Nestor O., Shpak; Vadim P., Deniega, *Data Acquisition and Signal Processing for Smart Sensors*, England, Jhon Wiley & Sons, LTD, 2002.

- [3] Coughlin F., Robert; Driscoll F., Frederick, *Amplificadores Operacionales y Circuitos Integrados Lineales*, (4ª Ed.), México, Prentice Hall, (1993).
- [4] Yogesh B. Gianchandani, Chester G. Wilson and Jae-Sung Park, *Micromachined Pressure Sensors: Device, Interface Circuits and Performance Limits*, MEMS Applications, The MEMS Handbook, Edited by Mohamed Gad-el-Hak, Second Edition, CRC Press Taylor & Francis Group, USA, 2006.
- [5] Ramón Pallás Areny, *Sensores y Acondicionadores de Señal*, 3ª edición, Editorial Alfaomega Marcombo, México 2001.
- [6] Li Cao, Tae Song Kim, Susan C. Mantell and Dennis L. Polla, *Simulation and fabrication of piezoresistive membrane type MEMS strain sensors*, This research is funded by the US Naval Research Laboratory (NRL), contract number N00014-94-C-2231.
- [7] Maxim Integrated products, *Desing and Manufacture of Automotive Pressure Sensors*, Aplication note 762, USA, June, 29, 2001.
- [8] John J. Neumann, Jr. and Kaigham J. Gabriel, *CMOS-MEMS Membrane for Audio Frecuency Acoustic Actuation*, Electrical and Computer Engineering Department, Carnegie Mellon University. Pittsburgh, PA 15213, USA.
- [9] Nilmoni Deb and R. D. (Shawn) Blanton, *Built-In Self-Test of MEMS Accelerometers*, Journal of Microelectromechanical Systems, vol. 15, no. 1, pp. 52-68, february 2006.
- [10] Darrel R. Wur, Jimmy L. Davidson, *Senior Member, IEEE*, Weng Po0 Kang, *Member, IEEE*, and Donald L. Kinser, *Polycrystalline Diamond Pressure Sensor*, Journal of Microelectromechanical Systems, vol. 4, no. 1, pp. 34-41, march 1995.
- [11] Ni xiao-qi, Wang Ming, Chen Xu-xing, Ge yi-xian, Rong Hua, *A Novel Optical Fibers MEMS Pressure Sensor*, Jiangsu Key Lab of Opto-electronic Technology, School of Physical Science and Technology, Nanjing Normal University, 210097, Nanjing China, Institute of Physics Publishing, Journal of Physics: Conference Series **34** (2006) 996–1001, International MEMS Conference, pp. 996-1001, 2006.
- [12] Nadim Maluf, *An Introduction to Microelectromechanical Systems Engineering*, 2ª Ed., USA, Artech House, 2000.
- [13] Jim Carter, Allen Cowen, Busbee Hardy , Ramaswamy Mahadevan, Mark Stonefield, and Steve Wilcenski, *PolyMUMPs Design Handbook*, 11ª revisión, MEMSCAP, Inc.
- [14] *PolyMUMPs FAQs*, 2ª revisión, MEMSCAP, Inc.
- [15] S. P. Beeby, M. Stuttle and N. M. White, *Design and fabrication of a low-cost microengineered silicon pressure sensor with linearised output*, *IEE Proccedings Science Meas Technology*, Vol. 147, No. 3, pp. 127-130, may 2000.
- [16] Don C. Abeysinghe, Samhita Dasgupta, Joseph T. Boyd and Howard E. Jackson, *A Novel MEMS Pressure Sensor Fabricated on an Optical Fiber*, *IEEE Photonics Technology Letters*, Vol. 13, No. 9, pp. 993-995, september 2001.

CONSIDERACIONES DE DISEÑO DE ANTENAS RECONFIGURABLES MEMS PARA USO EN SISTEMAS DE COMUNICACIÓN MINIATURIZADOS”

Sosa Torres Miguel Ángel, Navarro Alvarado José Marcos, Montiel Uribe Rodrigo.

Centro de Investigación y Diseño en Micro y Nano Tecnología (CIDEMyNT), Departamento de Electrónica
Instituto Tecnológico superior de Irapuato
Km. 12.5 Carretera Irapuato-Silao, Irapuato, Gto. 36821, México
Tel. (462)6067900, correo electrónico: misosa@itesi.edu.mx, jonavarro@itesi.edu.mx, romontiel@itesi.edu.mx

RESUMEN

Se describe la arquitectura de las antenas reconfigurables, que pueden ser utilizadas en sistemas de comunicación miniaturizados. Las consideraciones que se deben de tener para su diseño y su fabricación.

Palabras clave: Antena reconfigurable, Interruptores MEMS, Micro-tiras.

I. INTRODUCCIÓN

En la actualidad, se hacen grandes esfuerzos en el desarrollo de antenas reconfigurables, debido a las necesidades de mayores anchos de banda. Estas antenas han tenido gran aceptación en aplicaciones de comunicaciones [1]. Son convenientes para muchas aplicaciones electrónicas debido a las propiedades de seleccionar su frecuencia, su ancho de banda, su polarización y ganancia. Estas tienen la ventaja de bajo costo, tamaño y portabilidad. Estas pueden operar en más de una banda de frecuencia, reduciendo el número de antenas de comunicación. Es posible controlar el patrón de radiación, cambiando los elementos de transmisión o recepción [1], [4], [6], [7].

II. MÉTODO

El elemento principal en el diseño de las antenas reconfigurables, es el interruptor, el cuál es el encargado de configurar las diferentes estructuras de las antenas, afectando de esta manera la apertura de los elementos radiadores y en consecuencia la radiación de la antena. Comúnmente se han

utilizado interruptores de estado sólido para modificación de la apertura, pero estos están siendo sustituidos por interruptores micro-electromecánicos (MEMS) que necesitan bajos voltajes para su activación.

El diseño de las antenas reconfigurables, los interruptores MEMS, el sistema de control para llevar a cabo la activación y la implementación en un sistema multicapas es un problema de diseño muy complejo. Un gran problema es la falta de un sistema computacional eficiente para el modelado en el diseño de las capas.

La figura 1 muestra una estructura de un sistema de antenas de apertura reconfigurable, donde se puede identificar los siguientes componentes.

- 1.- Arreglo de antenas y sus redes de alimentación
- 2.- Red de Interruptores y sus circuitos de activación
- 3.- Estructuras planas que consisten de Superficies selectoras de frecuencia (FSS) y estructuras de la banda fotónica [1].

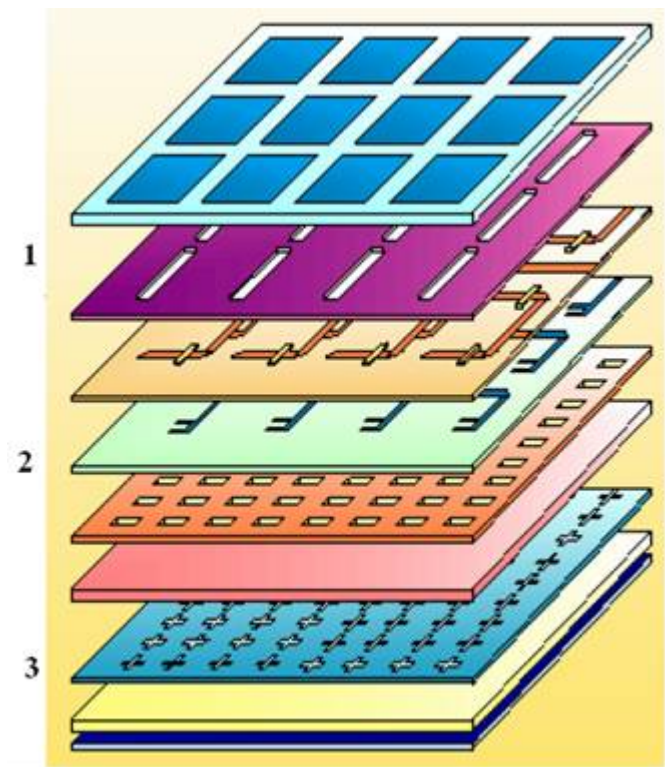


Figura 1. Diagrama de un sistema de antenas reconfigurable

Para realizar un buen modelado de las antenas, se tiene que recurrir a técnica numéricas de onda completa. La simulación de onda completa para una antena reconfigurable, con la estructura anterior, es prácticamente imposible debido a la gran cantidad de cálculos numéricos y a los recursos de cómputo necesarios. La única manera de atacar la simulación de una antena reconfigurable, es descomponer en pequeños subdominios más manejables. Se pueden plantear macro-modelos [1].

Es muy complejo realizar la simulación numérica de todo el sistema, es mejor realizar la simulación en cada componente, adecuado a determinada técnica.

Los arreglos de antenas planas, son modelados mediante el método de los momentos (MoM). Cuando el arreglo de las antenas es muy grande, este método es inútil. En este caso se pueden aplicar los macro-modelos para la simulación del arreglo de las antenas.

Se debe de tomar en cuenta los efectos de acoplamiento entre las antenas, por medio del acoplamiento de los macro-modelos.

Tanto para las estructuras infinitas periódicas como para las superficies selectoras de frecuencia (FSS) y las estructuras de la banda fotónica, el método más eficiente y preciso de análisis es la implementación periódica del método de momentos con funciones periódicas de Green [2].

INTERRUPTORES MEMS

Los interruptores MEMS y sus circuitos de polarización, pueden involucrar geometrías intrínsecas tridimensionales con materiales complejos variantes. En este caso, el método más adecuado es el método del elemento finito (FEM). El FEM es el método adecuado para la simulación en 3D de las antenas que involucran geometrías complejas y materiales. Cuando el tamaño de los interruptores es muy pequeño comparado con la longitud de onda, se emplea un FEM estático para obtener resultados óptimos. El interruptor, se representa por un circuito equivalente que puede ser añadido en el análisis de onda completa del elemento radiador [3], [5].

Los interruptores, son los elementos que se encargan de activar o desactivar los elementos radiadores de las antenas. La precisión en la simulación numérica de los dispositivos MEMS es un reto, debido a las altas relaciones de aspecto de estos y aunque son muy pequeños, permiten predecir su comportamiento eléctricamente. Cuando se colocan cerca de los elementos radiadores, el modelo del interruptor en conjunto con el elemento radiador cambia drásticamente debido a que los elementos radiadores son comparables en tamaño con la longitud de onda y entonces se debe de usar un modelo dinámico para su simulación.

En el laboratorio de Investigaciones de la Fuerza Aérea (AFRL) en Wright Patterson, OH, se han caracterizado interruptores en paralelo de los que se muestran en la figura 2 [8].

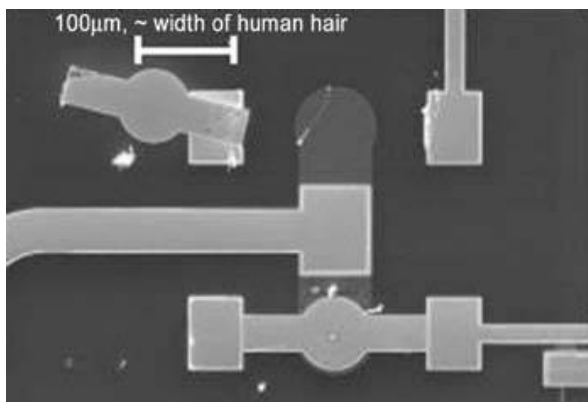


Fig. 2 Interruptor MEMS RF fabricada en el AFRL

En la figura 3, se muestra la estructura de estos interruptores.

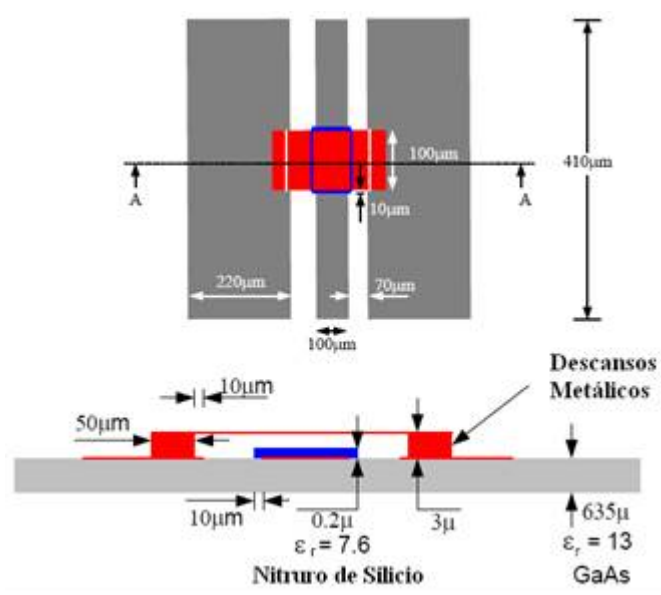


Fig. 3 Dispositivo MEMS fabricada en AFRL, consiste de un puente sobre una línea CPW, el cuál es empujado hacia abajo cuando se aplican 27 V DC entre el centro y los conductores de tierra del CPW.

Este interruptor consiste de una configuración de guía de onda coplanar (CPW) montada en un sustrato de 635 μm , el puente está anclado al plano de tierra por postes metálicos y suspendido por aproximadamente 3 μm sobre el centro del conductor. Un aislante dieléctrico de Nitruro de Silicio (Si_3Ni_4) de 0.2 μm de grosor colocado en el

centro del conductor crea un contacto capacitivo. Aplicando un voltaje de 27 VDC, es posible cerrar el interruptor. El ancho del puente se puede variar para diferentes voltajes.

Se elige el método del elemento finito (FEM) por su flexibilidad en el modelado de pequeños detalles geométricos y variación del material. El interruptor es modelado como un capacitor de placas paralelas el cual es analizado con un análisis FEM estático. Se crea un campo eléctrico entre el puente y el centro del conductor del CPW por medio de la aplicación de una diferencia de potencial entre ellos. Esto también sirve para la excitación en el problema numérico.

Se necesita un blindaje metálico para la terminación del mallado en FEM.

La diferencia de potencial causa la acumulación de una carga total igual en cantidad pero de polaridad opuesta sobre dos superficies potenciales. Teniendo el cálculo del campo eléctrico en cualquier dirección, la carga total Q se calcula integrando la componente normal del campo sobre una superficie cerrada encerrando una superficie potencial (el conductor central en este caso).

Entonces la capacitancia C es dada por Q/V , donde V es la diferencia de potencial. Los parámetros "S" se calculan mediante formulas de la teoría de líneas de transmisión. Si el ancho del puente crece de 25 a 100 μm , el parámetro S_{21} crece de 2.2 a 4 dB.

En la figura 4 se comparan los resultados medidos y obtenidos en el AFRL para el interruptor CPW para las dos posiciones del puente. En la posición normal (superior), la concordancia es excelente. El desafío es cuando está en la posición inferior (corto RF).

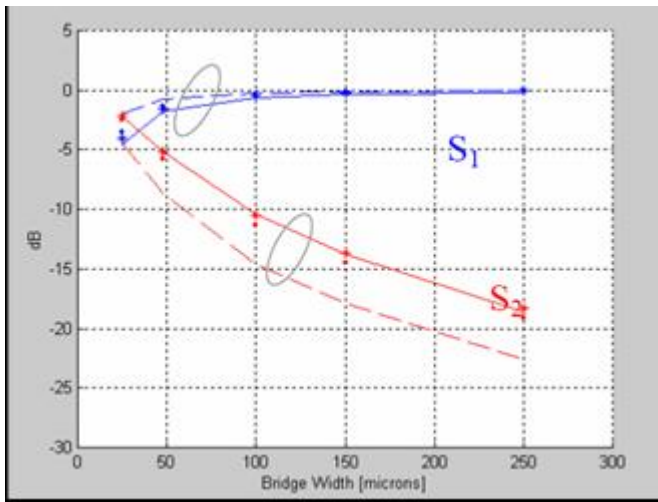


Fig. 4 Parámetros-S medidos contra simulados como una función del ancho del puente en la posición inferior. La línea sólida es la medida y la línea punteada es la simulada.

Existen varios factores a tomar en cuenta en los cálculos que explican la diferencia entre los valores medidos y los simulados. En la posición inferior, el campo dentro de la capa de Si₃Ni₄ domina la capacitancia y la exactitud estimada de su constante dieléctrica es muy importante. Una constante de permitividad relativa ε_r más pequeña permite una capacitancia menor y entonces un valor mayor de S₂₁. También influye el hecho de que no se toman en cuenta las rugosidades de las superficies y no se produce un contacto perfecto entre la parte inferior del puente metálico y la parte superior de la capa de Si₃Ni₄. Esto conduce a una capacitancia menor total.

Las rugosidades entre las superficies pueden ser representadas como una brecha de aire. Estas brechas de aire actúan como capacitores conectados en serie al capacitor de la capa de Si₃Ni₄.

La capacitancia total, se puede calcular mediante la formula

$$C_{tot} = \frac{C_{air} C_{SiN}}{C_{air} + C_{SiN}} = C_{SiN} \left(\frac{1}{1 + \epsilon_{SiN} \left(\frac{\Delta}{h} \right)} \right)$$

Cuando la capa de aire desaparece (Δ=0), se obtiene que C_{tot} = C_{Si₃Ni₄}. Que es cuando existe un contacto perfecto entre el puente metálico superior y la parte superior de la capa de Si₃Ni₄. En el caso de incrementos de la rugosidad (Δ>0), la capacitancia total va disminuyendo y el interruptor presenta un peor aislamiento. Un rango aceptable en las brechas de aire es de 13 a 17 nm. En la gráfica de la figura 4, se muestran las mediciones realizadas con estos valores de la brecha de aire para 13 nm (.) y 17 nm (*).

Los interruptores MEMS presentan muchas ventajas y desventajas sobre otros dispositivos de conmutación como son:

Ventajas

- Ancho de Banda
- Alto Aislamiento
- Baja pérdida de inserción
- Bajo consumo de potencia
- Linealidad

Desventajas

- Baja velocidad de conmutación
- Bajo manejo de potencia
- Alto voltaje de actuación
- Baja confiabilidad
- Empaquetamiento

La tabla 1 muestra una comparación de las características técnicas de los interruptores MEMS con otros dispositivos.

	Mechanical relay	PIN diode	FET	MEMS
Insertion loss at 1 GHz (dB)	0.25	0.5-1.0	0.5-1.0	< 0.1
Isolation at 1 GHz (dB)	70	20-40	40	> 40
Switching speed (s)	2 · 10 ⁻³	650 · 10 ⁻⁹	10 ⁻⁹ -10 ⁻⁸	10 ⁻⁶
Bandwidth (MHz)	dc-1200	20-2000	?	dc-100000
Actuation voltage (V)	100-200	3-5	5-50	3-30
Bias current (μA)	1-2	10 ⁴	< 10	< 10

Tabla 1. Comparación de interruptores de RF

Otro aspecto muy importante a considerar en el diseño de este tipo de antenas, es el patrón de radiación que se distribuye de acuerdo a la

cantidad de interruptores que se encuentran activados y su posición [3].

En la figura 5 se muestran algunos patrones de radiación generados por una antena con 4 interruptores en diferentes estados.

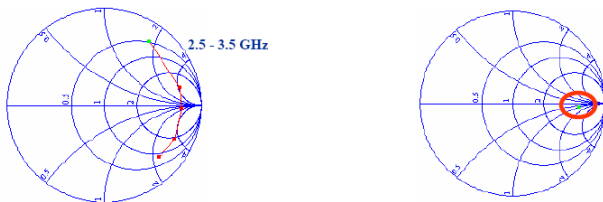
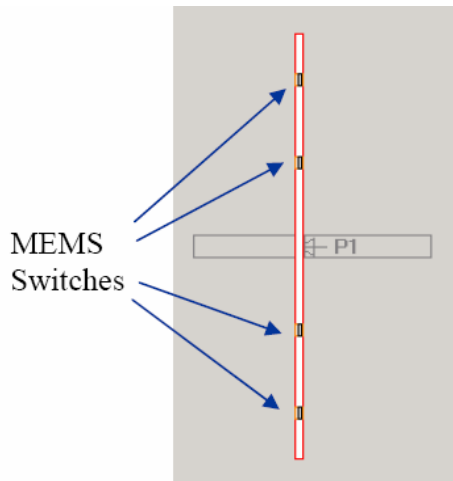


Fig. 5 Antena con interruptores MEMS y patrón de radiación

Cuando se operan en conjunto varias antenas, se puede modificar el patrón de radiación y la potencia de transmisión. En la figura 6, se muestra un arreglo de antenas de 3x3 y su patrón de radiación.

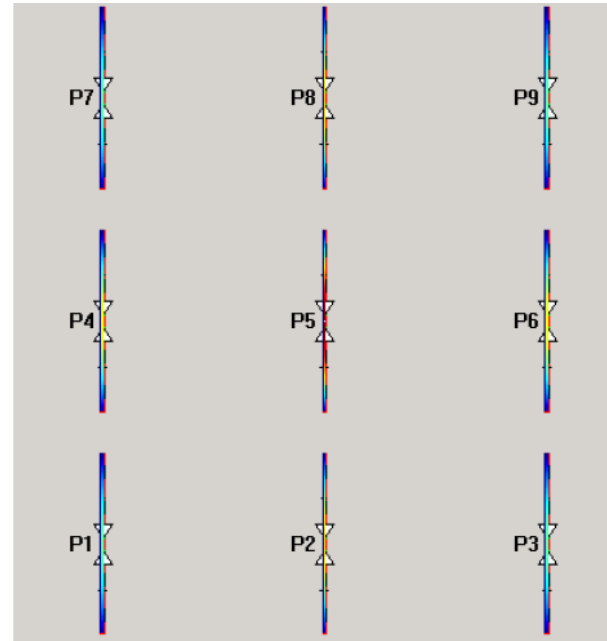


Fig. 6 Arreglo de antenas de 3x3 con los interruptores externos en posición inferior.

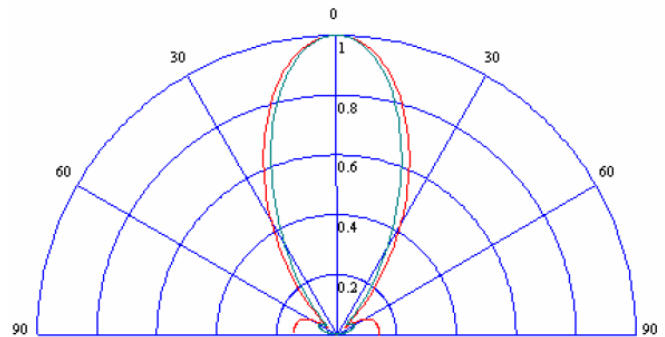


Fig. 7 Patrón de radiación para el arreglo de antenas de 3x3.

III. CONCLUSIONES

Debido a la complejidad de las antenas reconfigurables, en la simulación numérica, no es conveniente utilizar únicamente un método, sino los métodos adecuados para cada componente del sistema de antenas.

El diseño y la realización de los interruptores es uno de los aspectos más importantes, ya que estos son los que determinan la frecuencia de operación más óptima para la que deben responder las antenas. Estos deben ser controlados mediante un circuito electrónico que sea capaz de responder a la

frecuencia necesaria de conmutación a las diferentes frecuencias.

En el desarrollo de las tecnologías MEMS, es todo un reto realizar estos interruptores, ya que se tienen que adecuar a los procesos existentes de fabricación.

Antennas an Propagation library, Norwood, MA 2001.

[8] <http://www.afrl.af.mil>

IV. REFERENCIAS

- [1] T. Ozdemir, K.F. Sabet, et al, “A *Hybrid-statistical approach for accurate characterization of MEMS on complex platforms*”, 2001 IEEE AP-S Intl. Symp. Digest, Boston, MA, Julio 2001.
- [2] K.F. Sabet, D.P. Jones, J-C. Cheng, K. Sarabandi and L.P.B. Katehi. *Efficient printed antenna array synthesis including coupling among radiating elements using evolutionary genetic algorithms*, 1999 IEEE AP-S Intl. Symp Digest, Orlando, Florida, Junio 1999
- [3] Cummings Nathan P. “*Active Antenna Bandwith Control Using Reconfigurable Antenna Elements*”, Virginia Tech Antenna Group, Bradley Department of Electrical & Computer Engineering, Blacksburg Virginia, 2003
- [4] De los Santos Hector J. “*RF MEMS Circuit Design for wireless communications*”, Artech House, microelectromechanical systems series), 2002, USA.
- [5] Brown E.R. “*RF MEMS Switches for Reconfigurable Integrated Circuits,*” IEEE Trans. Microwave Theory Tech., Vol. 46, Noviembre 1998, pag. 1868-1880.
- [6] Wong, Kin Lu. “*Planar Antennas for wireless Communications,*” Wiley series in microwave and optical engineering, 2003, USA.
- [7] Ramesh Garg, Prakash Bhartia, Inder Bahl, Apisak Ittipiboon, “*Microstrip Antenna Design Handbook,*” Artech House

CIRCUITO INTERFAZ INTEGRADO PARA REGISTRO SIMULTÁNEO DE UN MICROSISTEMA MEMS NEURONA-CHIP

B. S. Soto-Cruz¹, F. López-Huerta², J. J. Estrada-López^{3, 5}, I. Juárez Ramírez⁴, A. L. Herrera-May⁶

¹ Centro de Investigación en Dispositivos Semiconductores, BUAP, Puebla, Pue.

² Facultad de Ciencias Físico- Matemáticas, BUAP, Puebla, Pue.

³ Lab. de Instrumentación, Control, Visión y Robótica. Facultad de Matemáticas, UADY. Mérida. Yucatán.

⁴ Lab. de Microelectrónica, Instituto Nacional de Astrofísica Óptica y Electrónica. Puebla, Pue.

⁵ Departamento de Ingeniería Eléctrica y Electrónica, Instituto Tecnológico de Mérida. Mérida. Yucatán

⁶ Centro de Investigación en Micro y Nanotecnología, Universidad Veracruzana, Boca del Río, Ver.
Av. 14 Sur y Blvd. Valsequillo, Edificio 136, Col. San Manuel, C.U. Puebla, Puebla. 72590, México
Tel. 2295500, correo electrónico: blanca.soto@icbuap.buap.mx

RESUMEN

Se presenta el desarrollo de un circuito interfaz analógico integrado CMOS para la elaboración de biopotenciales dentro de un microsistema MEMS de registro simultáneo *in vitro* a través de una matriz de microelectrodos extracelulares de 4 X 3. Se presenta la realización de la matriz de microelectrodos y la interconexión a cada uno con el circuito interfaz. Se propone un arreglo de transistores par-acoplado complementarios operando en la región de subumbral para realizar una sintonización de la frecuencia de corte bajo a través de la polarización con el objetivo de seleccionar el periodo de los biopotenciales de neurona. El microsistema neurona-chip es alimentado con un par de baterías de 1.5V consumiendo una corriente de polarización de 121 μA ; el circuito interfaz completo ocupa 0.0664 mm^2 de un dado de silicio que mide 4.84 mm^2 fabricado con tecnología CMOS 1.5- μm . Se obtuvo un rango dinámico de 61dB para el máximo de la señal de entrada. La caracterización del circuito mostró un offset de 20mV y un consumo de micropotencia de alrededor de 363 μW . La caracterización experimental de todos los bloques constituidos del circuito interfaz demostró una funcionalidad adecuada para señales con amplitudes del orden de μV .

Palabras clave: Microsistemas, MEMS, Micro Electro Mechanical Systems, Biopotencial, Microelectrodo.

I. INTRODUCCIÓN

Los potenciales eléctricos existen a través de las membranas que envuelven las células vivas, tales como neuronas, nervios, músculos y glándulas, así como en diversas células de las plantas exhiben este fenómeno de generar biopotenciales [1-4].

Algunos sistemas de registro utilizan generalmente métodos electrofisiológicos a través de microelectrodos intracelulares o extracelulares invasivos o no, para registrar el biopotencial generado por la célula bajo condiciones con o sin estímulo; sea en especies *in vitro* o *in vivo*; un ejemplo de ello es el caso del espectro de EEG (electroencefalograma), el cual ha servido para vislumbrar las patologías intracranianas, sin embargo, es bien sabida la controversia sobre las lesiones irreversibles que esta técnica provoca [5]. En neurobiología, existen grandes expectativas por entender la riqueza y complejidad del pensamiento y del comportamiento del ser humano, siendo una de las áreas más apasionantes dentro de la neurociencia el estudio de los mecanismos del comportamiento global de la actividad cerebral. Numerosos han sido los esfuerzos en identificar y analizar este comportamiento, en décadas pasadas los sistemas que realizaban un registro de los biopotenciales neuronales tenían el problema de baja resolución espacial consecuencia de las técnicas de litografía utilizadas en la fabricación de los microelectrodos, las cuales presentaban serios problemas en el control de las dimensiones de cada sitio y del propio arreglo. Adicionalmente a este

problema las investigaciones sobre las características de organización estructural, capacidad dinámica y respuesta transitoria de la red neuronal en animales avanzados permanecían inexploradas principalmente porque la información generada de la actividad, reconocimiento, manipulación y almacenaje requiere una respuesta rápida en el acondicionamiento y procesamiento de la señal registrada, [6]. No es sino hasta 1940, cuando se extiende el uso de los instrumentos auxiliares de acondicionamiento (amplificadores electrónicos, osciloscopios, etc.) comenzando de manera macroscópica a estudiarse los sistemas de registro de biopotenciales a través de microelectrodos. El mayor desarrollo se obtuvo en el registro de la actividad intracelular. En este, un microelectrodo de vidrio o pipeta perfora el tejido midiendo la actividad de la célula. Sin embargo, este método tiene limitaciones tecnológicas y metodológicas importantes: el uso simultáneo de más de dos sitios (microelectrodos intracelulares) conectados a una célula era técnicamente irrealizable dado que el registro intracelular de 5–10 neuronas era logísticamente imposible [7], además de no poder mantener los registros a largo término. Debido a estas limitaciones y al creciente interés en profundizar el estudio de la actividad de una red neuronal, el objetivo en estos últimos 20 años es el desarrollo de técnicas de registro no invasivas utilizando métodos ópticos o técnicas de microelectrodos de película delgada, las cuales siguen aún en continua evolución [8]. Este tipo de trabajos regularmente hacen uso de electrodos de metal noble como son platino y oro entre los más utilizados, sin embargo, la interfaz generada entre el metal/electrolito presenta una muy alta impedancia lo que provoca una atenuación del biopotencial registrado. Además de esto, son pocos los trabajos que se han desarrollado para integrar tanto los microelectrodos como los circuitos electrónicos sin tener limitaciones técnicas o tecnológicas. Por ejemplo, existen sistemas implantables invasivos (del tipo punta) que se insertan en la masa encefálica registrando la actividad neuronal, sin embargo cuentan con un número limitado de microelectrodos los cuales están hechos de platino [9]. Por estas razones, generalmente los sistemas de registro de

biopotenciales extracelulares, compuestos de un arreglo de microelectrodos de metal e instrumentos de acondicionamiento, han sido pensados por caminos diferentes. Considerando lo anterior, es justificable continuar con los estudios relacionados con sistemas de registro de biopotenciales que vislumbren la comunicación que existe entre las neuronas de forma no invasiva y simultánea por largos periodos de tiempo. Este trabajo presenta la propuesta de realización de un microsistema neurona-chip con el uso de la tecnología de silicio comercial y por tanto, de la miniaturización de la matriz de microelectrodos extracelulares y circuito interfaz de lectura integrado. Los experimentos realizados con microelectrodos basados en silicio presentan una menor impedancia que su contraparte de metal. Se incluye el circuito de lectura completamente integrado acoplado en DC, lo que proporciona una detección simultánea en los sitios de registro. En una primera aproximación se realiza un arreglo de 12 microelectrodos con un circuito amplificador asociado a cada uno de ellos. Los amplificadores tienen un control que selecciona la frecuencia de corte bajo para biopotenciales entre $\sim 10\text{Hz}$, 50Hz , 100Hz y 200Hz y es insensible a variaciones base CD. Se realiza la descripción del diseño del circuito electrónico y las mediciones obtenidas de la caracterización eléctrica por medio de una señal generada idealmente. Finalmente, se discuten las condiciones propuestas para la medición biológica de una red de neuronas de ratas neonatas Wistar crecidas in vitro [10].

II. METODOLOGÍA DE DISEÑO DEL CIRCUITO INTERFAZ

El acondicionamiento del biopotencial en un sistema de registro, de manera general involucra una reducción de la impedancia de entrada, un filtraje (ajustado a un ancho de banda específico dependiendo de la aplicación) y una amplificación. Dado que las señales tomadas de los microelectrodos extracelulares son del rango de microvolts, de la etapa de amplificación dependerá la viabilidad del procesamiento y análisis en los dispositivos electrónicos disponibles, tales como convertidores, analizadores o computadoras.

Además, uno de los mayores retos en la interface electrónica del microelectrodo de registro es el voltaje aleatorio asociado a la interface electroquímica microelectrodo/electrolito. El voltaje en CD entre el electrolito y el microelectrodo es objeto de variaciones substanciales que pueden ser tan altas como ± 50 mV [11]. Consecuentemente, la etapa de entrada del amplificador debe ser insensible a estas variaciones de DC y permitir registrar señales para ser reproducidas sobre un ancho de banda típico de 100 Hz a 10kHz [12].

Finalmente, los parámetros de diseño que se deben considerar son: tamaño, consumo de potencia, voltaje de operación, voltaje de offset en la entrada, con consideraciones de protección contra daños por descarga eléctrica, oscilaciones, incrementos en el slew rate y ruido de salida.

El diseño del amplificador en lazo abierto cuenta con dos etapas, la primera hace uso de dos transistores de entrada tipo PMOS para disminuir el ruido. La segunda etapa se realiza a través de dos transistores acoplados complementarios PMOS y NMOS. Para que el desempeño sea estable se agregó una etapa de compensación, la cual se realiza a través de un capacitor de compensación C_c conectado entre la entrada (etapa uno) y la salida de la etapa dos. Inspeccionando la función de transferencia del amplificador observamos dos polos y un cero cuyas posiciones dependen inversamente del producto de las conductancias drenador-fuente de los transistores asociados y del capacitor de compensación. Esto hace que el efecto del capacitor sea separar los polos, es decir, empujar el polo 1 hacia las bajas frecuencias mientras que el polo 2 sea llevado a las altas frecuencias. Con respecto al cero, se tiene que la ubicación sea el resultado de la razón de la transconductancia del transistor de salida de la segunda etapa y el capacitor de compensación, de esta forma, el desplazamiento es hacia el lado derecho del plano s lo que no mejora el margen de fase. Para resolver este efecto, se usó un resistor complementario MOS conectado entre el nodo de salida y el capacitor de compensación [13]. En la figura 1 se muestra la comparación del diagrama de polos y ceros del circuito interfaz en lazo abierto compensado con un solo capacitor y el

compensado por el arreglo capacitor-resistor complementario MOS.

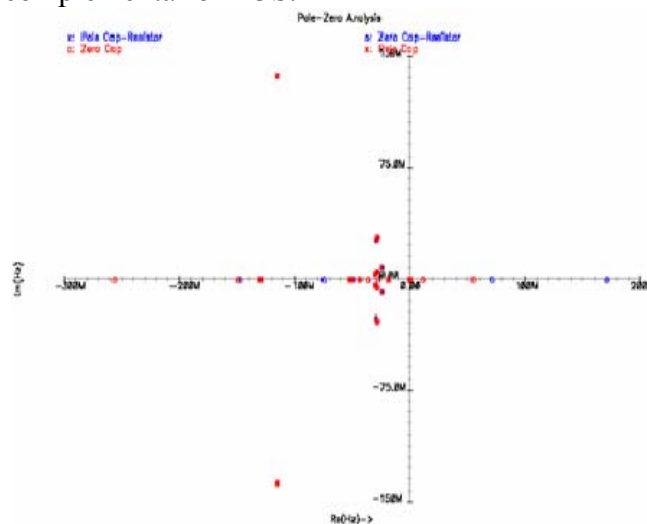


Figura 1. Diagrama de polos y ceros del circuito interfaz para \square capacitor –resistor y \square capacitor.

Puede observarse que los polos dominantes se mantienen prácticamente sin cambio, pero los ceros dominantes en cambio, han sufrido un corrimiento de en el arreglo capacitor-resistor complementario MOS alejándose del origen hacia el lado derecho, reduciendo con esto su efecto sobre el margen de fase.

En cuanto a la frecuencia de ganancia unitaria f_{0dB} y la frecuencia de fase a f_{180} es muy cercana en este amplificador, lo que significa que la ganancia de margen y el margen de fase es cercana a cero, y el amplificador por tanto, es estable.

Dado que el amplificador deberá atenuar la interferencia proveniente de las líneas de alimentación, regularmente esto se logra con técnicas como uso de capacitores conmutados [14], o diodos polarizados inversamente [15-16], el diseño del amplificador en lazo cerrado presentado en este trabajo, realiza la implementación de un resistor MOSFET operando en la región subumbral profunda para lograr un muy alto valor de resistencia para sintonizar o controlar la frecuencia de corte bajo con el objetivo de minimizar tanto el ruido de baja frecuencia como la interferencia proveniente de las líneas de alimentación, y el voltaje de CD entre el electrolito y el microelectrodo.

II.I. CIRCUITO INTERFAZ DE LECTURA DEL MICROSISTEMA MEMS NEURONA-CHIP.

El amplificador en lazo cerrado realiza la retroalimentación por un divisor capacitivo C1 y C2 para ajustar la ganancia y disminuir el consumo de potencia, además se incluye la resistencia RMOS para sintonizar la frecuencia de corte bajo. La topología del primer bloque del circuito de lectura se muestra en la figura 2.

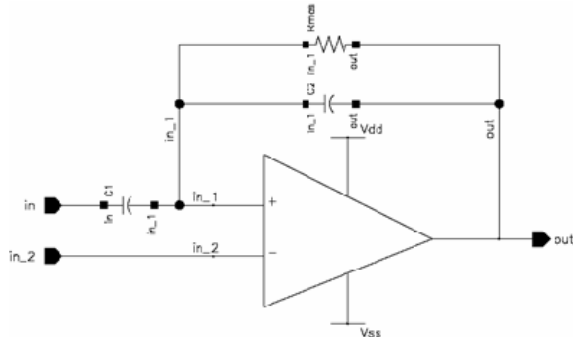


Figura 2. Bloque de entrada del circuito interfaz.

La ganancia del amplificador está dada por la razón de los capacitores C1 y C2, y la frecuencia de corte bajo es inversamente proporcional al producto RMOS y C2. La frecuencia de corte alto, en cambio, es directamente proporcional a la transconductancia del transistor de entrada e inversamente proporcional al producto de la ganancia en lazo abierto y el capacitor de compensación Miller.

II.II. RESISTENCIA MOS, RMOS

La síntesis de resistores usando transistores MOSFET, facilita la implementación de resistencias de grandes valores en una pequeña fracción de área del chip cuando se operan en la región de subumbral profunda. Por definición, en la región de subumbral profunda solo la corriente de difusión está presente, de tal manera que la corriente a una polarización de cuerpo cero, usualmente se modela por la expresión exponencial de la ecuación (1), [17]:

$$I_D = I_s \left[1 - \exp\left(-\frac{V_{GS}}{v_t}\right) \right] \exp\left[\frac{V_{GS}-V_{th}}{n v_t}\right] \quad (1)$$

donde V_{th} es el voltaje de umbral, v_t es el voltaje térmico, V_{GS} es voltaje de polarización compuerta-

fuerza, y n es el factor de calidad conocido como la pendiente de umbral. I_s es el coeficiente que depende de la capacitancia de compuerta por unidad de área, del tamaño efectivo, de la movilidad efectiva del canal, del voltaje térmico y del voltaje de drenaje-fuente [18]. La región de transición entre los mecanismos presentes nos determina el rango en donde la corriente está dominada por difusión. El cambio de la característica I_D-V_{GS} entre las regiones del comportamiento exponencial a lineal puede obtenerse encontrando el punto de intersección entre las asíntotas (V_{th}).

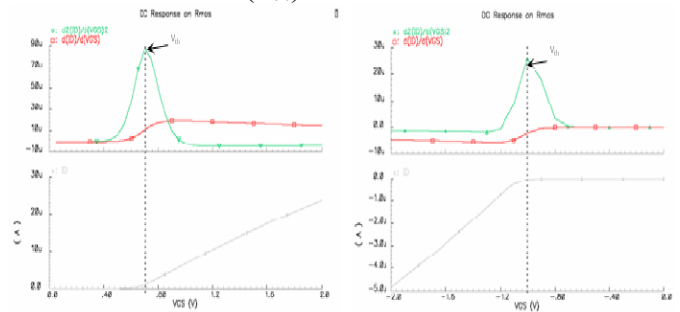


Figura 3. Regiones de transición exponencial y lineal para el transistor acoplado NMOS y PMOS sobre el comportamiento de I_D , con dI_D/dV_{GS} y d^2I_D/dV_{GS}^2 . Punto transición entre las regiones V_{th} .

Existen diferentes topologías de circuitos para realizar una resistencia MOS, los resistores flotantes [19], el resistor lineal controlado por voltaje [20-21], etc., en este trabajo se presenta el resistor MOS par-acoplado complementario operando en la región de subumbral profunda implementado con tecnología comercial CMOS 1.5- μm .

La propuesta para el circuito resistor MOS se realiza utilizando una combinación de transistores complementarios NMOS y PMOS, como se muestra en la figura 4. Las relaciones de forma calculadas para los transistores M1 a M4 de R_{mos} fueron $W/L=10\mu\text{m}/1.6\mu\text{m}$. Todos los transistores NMOS M1 y M2 y PMOS M3 y M4, operan en la región de subumbral.

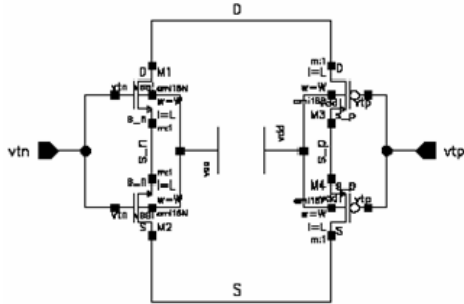


Figura 4. Topología de la resistencia MOS.

La operación de control se ajusta por medio del voltaje de la compuerta V_{GS} de los transistores NMOS y PMOS como v_{tp} y v_{tn} respectivamente. La figura 5 muestra los valores obtenidos en la resistencia R_{mos} operando en la región subumbral profunda con variación del voltaje de drenaje-fuente de 220mV a diferentes valores de polarización de compuerta-fuente v_{tp} para un valor fijo v_{tn} de 600mV.

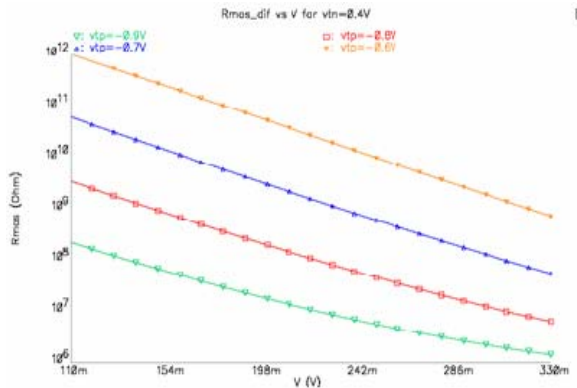


Figura 5. Curva R_{mos} vs. V para la resistencia MOS a diferentes valores de polarización de la compuerta (v_{tp} , v_{tn}).

La resistencia vista en las terminales “D” y “S” (figura 4) decae lentamente al incrementar el voltaje de compuerta y los rangos se encuentran en valores de decenas de $G\Omega$. Esto permite un menor paso de control en la selectividad de la frecuencia de corte bajo.

II.III. DISEÑO DEL CIRCUITO INTERFAZ

Regularmente las interconexiones largas son causantes de crosstalk entre los diferentes canales o microelectrodos, además se agrega a esto el ruido debido a fuentes eléctricas en la vecindad del área de registro y/o por las propiedades intrínsecas de la solución fisiológica (ruido térmico). De forma

general, para minimizar este ruido externo e interferencias regularmente se recurre a los sistemas de blindaje como la caja de Faraday. En circuitos integrados, se opta por el uso de los anillos de guarda los cuales deben estar referidos a un mismo punto de tierra. En la figura 6 se muestra la disposición del diseño del circuito interfaz con anillo de guarda referido al potencial más bajo.

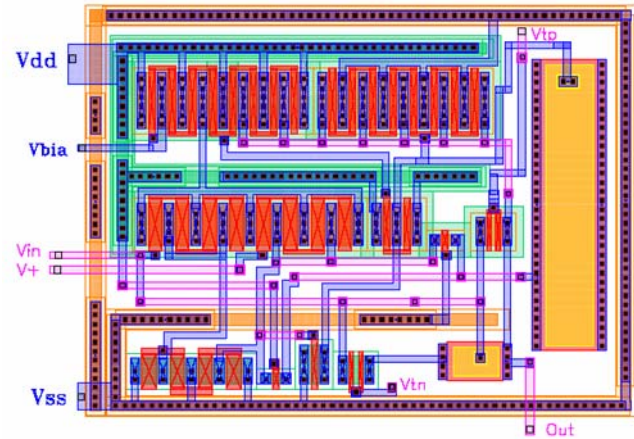


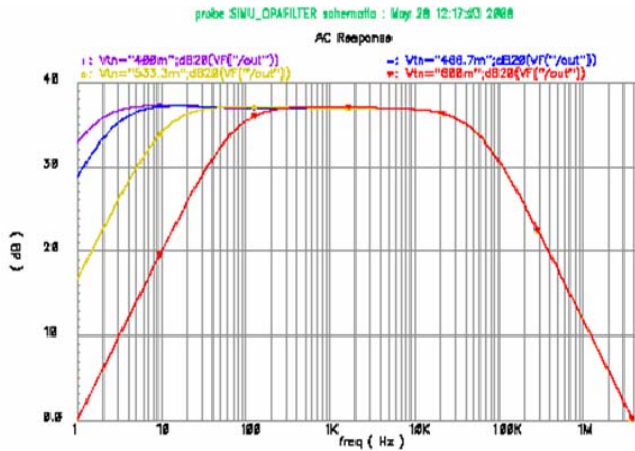
Figura 6. Diseño físico o plano de layout para el circuito interfaz integrado.

La razón de aspecto para cada transistor del circuito interfaz se muestran en la Tabla 1. El área obtenida es de $170\mu\text{m} \times 190\mu\text{m}$ sin incluir el capacitor C1.

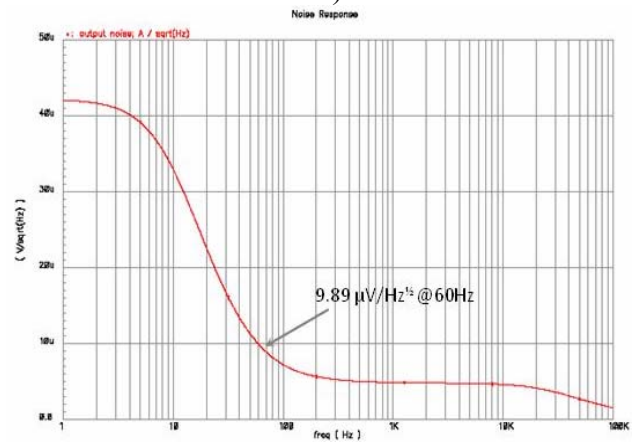
M1-M2	61u/4.8u
M3-M4	22.4u/5.2u
M5-M6	40u/2.4u
M7	30.4/2.4u
M8	14u/2.4u
M9	80u/2.4u
M10	160u/2.4u
M11-M12	4u/2.4u
M13-M14	10u/1.6u
M15-M16	10u/1.6u

Tabla 1. Razón de aspecto de los transistores del circuito interfaz.

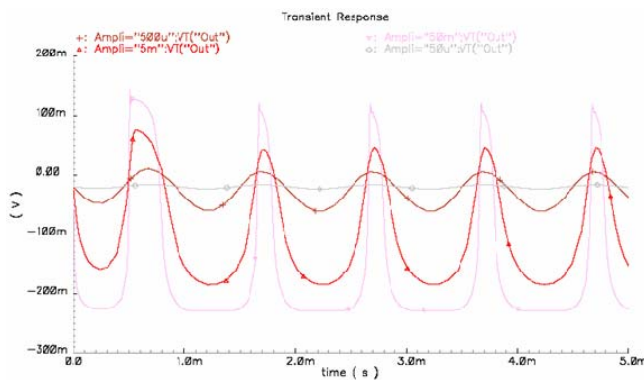
En la figura 7 se muestra la respuesta en frecuencia, el análisis de ruido y la respuesta en el dominio del tiempo para el circuito interfaz considerando los elementos parásitos del dispositivo.



7 a)



7 b)



7 c)

Figura 7. Análisis de a) Respuesta en frecuencia b) Ruido a la salida del circuito interfaz a diferentes valores de V_{tn} y c) Respuesta en el dominio del tiempo para diferentes amplitudes de la señal de entrada.

La atenuación de las señales ocurre debido a la capacitancia distribuida en los alambres del blindaje que conectan el microelectrodo al amplificador. En nuestro caso para el peor caso (microelectrodos centrales), el amplificador se encuentra a $450\mu\text{m}$ de separación.

III. DISEÑO DE MICROSISTEMA MEMS NEURONA-CHIP

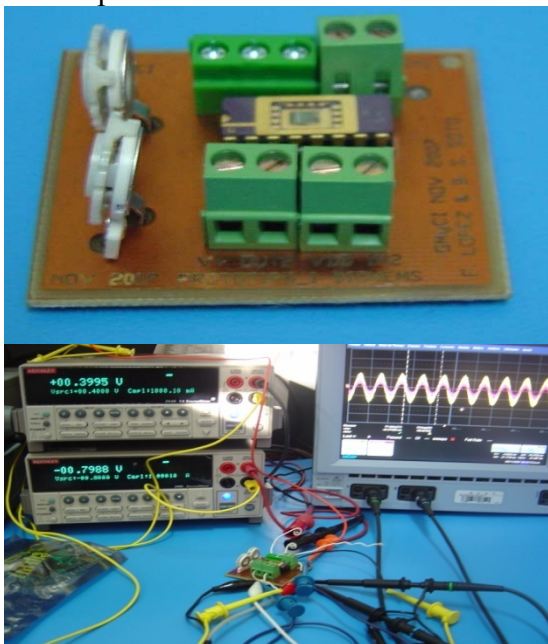
El chip fabricado cuenta con dimensiones de 2.4mm por lado, dando un espacio central para generar la matriz de microelectrodos cuadrados de 1.17mm^2 . Se decidió por un arreglo de 12 microelectrodos debido a que el arreglo completo tiene 4 columnas y 4 filas, con 4 electrodos de referencia. La matriz de microelectrodos de registro es de 3×4 , cuyos microelectrodos en el primer arreglo de 3×3 tienen dimensiones de $100\mu\text{m}$ por lado mientras que la última fila los microelectrodos tienen dimensiones de $200\mu\text{m}$. La separación entre microelectrodos es de $50\mu\text{m}$. El diseño físico de la matriz de microelectrodos o microestructuras se realizó a través de las capas del proceso tecnológico comercial CMOS $1.5\text{-}\mu\text{m}$ siendo el substrato de partida tipo P. Se formó un contacto óhmico entre la capa de aluminio del microelectrodo a través de la difusión del silicio y se realizó una abertura en el vidrio para exponer los microelectrodos al exterior. En la Figura 8 se muestra un detalle lente $50\times$ de la última fila de la matriz de microelectrodos.



Figura 8. Microfotografía (50x) de microelectrodo de $200\mu\text{m}$ por lado.

IV. MEDICIÓN Y CARACTERIZACIÓN ELÉCTRICA DEL CHIP

Se realizó la medición eléctrica de cada circuito interfaz. Primero se monto el chip en el encapsulado del tipo CSBB04833 de Spectrum Semiconductor Materials, para posteriormente alambrar los PADS del chip a las 32 patas (PINs). Las condiciones de la medición fueron las siguientes: se generaron funciones variantes en tiempo por medio de un generador de funciones arbitrarias Agilent, la mínima amplitud generada fue de 100mV_{pk} . La interfaz gráfica para la medición de la respuesta en tiempo se formó a través del osciloscopio digital Wave Surfer de Lecroy, así como la inyección del barrido en voltaje de la fuente y la polarización fueron generadas por medio de las fuentes programables Keithley modelo 24400, 2440 y Agilent modelo E36620A. La estación de pruebas para las mediciones del chip a nivel oblea fue realizada por medio de la estación J MicroTechnology JR-2727. Se realizaron mediciones a temperatura ambiente para obtener los valores de voltaje y tiempo de la señal generada por el circuito interfaz y el rango dinámico para la distorsión de la señal. La figura 9 muestra el chip encapsulado montado en la tarjeta PCB y algunas de las formas de onda obtenidas en el osciloscopio.



9 a)



9 b)

Figura 9. Condiciones de medición a) Chip encapsulado y b) Formas de onda en el osciloscopio.

Regularmente, el rango esperado se da entre 100Hz a 6kHz, por lo que la polarización preajustada en la compuerta de la resistencia R_{mos} fue dada para sintonizar a no menos de 100Hz. Por tal motivo, para la frecuencia de corte bajo se pudo obtener una medición preliminar variando la frecuencia de entrada de la fuente senoidal, obteniéndose resultados de ganancia máxima alrededor de 1KHz. Esta frecuencia es el valor típico de caracterización y en donde regularmente los biopotenciales se producen en las ratas Wistar.

V. CONCLUSIONES

En este trabajo se presentó la implementación de un circuito interfaz para registrar biopotenciales dentro de un sistema MEMS integrado con procesos CMOS comercial $1.5\text{-}\mu\text{m}$ el cual ocupa un área del chip de 0.0664 mm^2 . Se describió las consideraciones de diseño para la realización de una resistencia de lazo cerrado a través del arreglo de transistores MOS par-acoplados complementarios obteniéndose valores de unos

cuantos $G\Omega$ para sintonizar la frecuencia de corte bajo. Se obtuvieron valores de -3dB para frecuencias por debajo de 100Hz polarizando la compuerta de la resistencia R_{mos} a 0.4V y -0.8V para el par NMOS y PMOS respectivamente. Las características cumplieron con las especificaciones de moderada ganancia, bajo consumo y área reducida, obteniéndose una ganancia de 100, corriente de polarización alrededor de 120 μ A, potencia de 363 μ W con un offset de 20mV. Finalmente, se demuestra que es posible la integración del microsistema MEMS neurona-chip de registro portátil, cumpliendo las recomendaciones generales y la figura de mérito en cuanto a cercanía entre el sitio de registro y la electrónica asociada, área reducida y bajo consumo de potencia.

VI. AGRADECIMIENTOS

Los autores agradecen a MOSIS la oportunidad de fabricar los dispositivos a través del programa Educativo y de Investigación MEP MOSIS, de la cuenta MOSIS 4683-MEP-RES/BUADP/CDIDS, y el financiamiento de SEP-CONACYT dentro del proyecto ref. 45043.

VII. REFERENCIAS

- [1] *Studying Some Features of Maize Genetics and Developmental Biology Using Electrophysiological Techniques*. V.N. Lysikov, Institute of Genetics, Academy of Sciences of Moldova, Kishinau
- [2] K. Intabon, T. Maekawa, K. Sato, “*Studies on the Monitoring Techniques and Application of Bioelectric Potentials in Bioproduction*” J. of Soc. Of Agri. Struc., Japan. Vol 27, No. December 1196, 141-148.
- [3] Jorge R. Gersenowies R., Temas de Neurobiología Comparada N° 2, ‘Métodos en el estudio de la neurobiología’, UNAM México 2000.Y.
- [4] E. Neher, *Ion channels for communication between and within cells*. Science 256 (1992) 498-502.
- [5] L. Kiloh, J. McTomas, W. Osselton. *Clinical Electroencephalography*, 3rd ed. London, 1972.
- [6] G. Edelman, Neural Drawinism, *Basic Books Inc*. New York 1987.
- [7] “*Medical Instrumentation, Application and Design*”, John G. Webster, John Wiley & Sons Inc., pg. 183-231, 1998.
- [8] Gross GW and Lucas JH. (1982) *Long-term monitoring of spontaneous single unit activity from neuronal monolayer networks cultured on photoetched multielectrode surfaces*. J. Electrophys. Tech. 9, 55-69.
- [9] K. Najafi, J. Ji, K. Wise, “*Scaling limitations of silicon multichannel recording probes*”, IEEE Trans. Biomed. Eng. 37 (1990) 1-11.
- [10] Mercado F., López I. A, Acuna D., Vega R. and Soto E., “*Acid-sensing ionic channels in the rat vestibular endorgans and ganglia*”. J. Neurophysiology, 96 (2006) 1615-1624.
- [11] Christian C.Enz “*Circuit techniques for reducing Effects of Op-Amp Imperfections: Autozeroing, Correlated Double Sampling, and Chopper Stabilitation*”. Proc. of the IEEE Vol.84 No. 11 November 1996. 1584-1614.
- [12] Q. Bai and K.D. Wise. “*Single-unit-neuronal recordings with active microelectrode arrays*”. IEEE Trans. Biomed. Engineering. Vol. 48. No. 6, pp 911-920. Aug.
- [13] Reid R .Harrison. “*A Low-Power Low-Noise CMOS Amplifier for neural recording Applications*”. IEEE Journal of Solid-State Circuits. vol.36, No. 6, June 2003.
- [14] K. D. Wise, J. Angell, “*A low capacitance multielectrode probe for use in extracellular neurophysiology*”, IEEE Trans. Biomed. Eng. 22 (1975) 212-219.S.
- [15] S. Bement, K. Wise, D. Anderson, K. Najafi, K. Drake, “*Solid-State Electrodes for multichannel multiplexed intracortical neuronal recording*”, IEEE Trans. Biom. Eng. 33 (1986) 230-241.I.

- [16] Roy H. Olsson. “*Band-Tunable and Multiplexed Integrated Circuits for simultaneous Recording and Stimulation With Microelectrode Arrays*”. IEEE Trans. Biomed. Engineering. Vol.52, No.7, July 2005.
- [17] Arora. N, “*MOSFET Models for VLSI Circuit Simulation*” apringer-Verlag 1993.
- [18] Department of Electrical Engineering and computer Sciences University of California, Berkeley, “*BSIM3v3 User’s manual (Final Version)*”, 1996.
- [19] P. Singh, et al, “*A new floating resistor for CMOS technology,*” *IEEE Trans. Circuit Syst.*, vol. 36, pp. 1217-1220, Sept. 1989.
- [20] Han and s. B. Park, “*Voltage-Controlled Linear Resistor by Two MOS Transistors and its Applications to Active RC Filter MOS Integration,*” *Proc. IEEE*, vol. 72, pp. 1655-1657, Nov. 1984.
- [21] J. Ramirez-Angulo, R. G. Carvajal, A. Torralba, J. Galán, A. P. VegalLeal, and J. Tombs. “*The Flipped Voltage Follower: a useful cell for low-voltage low power circuit design*” *Proc. ISCAS’02*, vol. 3, pp 615-618, 2002.

DISEÑO PARAMETRIZADO DE MICROESTRUCTURAS TIPO ENGRANE

R. Campos-Rodríguez, M. Alcaraz-Mejía, L. García-Contreras, A. Aceves-Paredes, M. Aguirre-Villa

Departamento de Ciencias Tecnológicas, División de Desarrollo Bio-Tecnológico
 Centro Universitario de la Ciénega, Universidad de Guadalajara
 Av. Universidad 1115, Col. Linda Vista, C.P. 47810, Ocotlán, Jalisco, México
 Tel. +52 (392) 9259400, correo electrónico: {raul.campos, mildreth.alcaraz}@cuci.udg.mx

RESUMEN

Los engranes MEMS son ampliamente utilizados para transmitir movimiento rotacional a otras microestructuras o bien para convertir movimiento rotacional en movimiento lineal y viceversa. Este artículo presenta una metodología para el diseño parametrizado de microestructuras tipo engrane. La técnica propuesta permite manejar el diseño de engranes desde su concepción y análisis hasta la generación de modelos sólidos 3D y su análisis mecánico en herramientas de simulación numérica. Finalmente, la metodología propuesta permite la generación de máscaras para la fabricación de las microestructuras tipo engrane.

Palabras clave: Engranes MEMS, Diseño Parametrizado, AutoCad, Ansys

I. INTRODUCCIÓN

La tecnología de los microsistemas o MEMS propone un nuevo paradigma para la creación de sistemas al permitir su miniaturización. Las resoluciones de fabricación para MEMS se encuentran dentro del rango de los micrómetros (1×10^{-6} m), muy superiores a las del acero (1×10^{-3} m), principal material para la fabricación de maquinaria. Los procesos de fabricación de microsistemas se derivan de los procesos de fabricación presentes en la industria de los circuitos integrados. Por esta razón se emplea fundamentalmente silicio para construir estructuras mecánicas microscópicas [1]. Sin embargo, no se limita a utilizar solo este material, ya que es posible depositar polímetros y otros metales como oro, plata, cromo, níquel y titanio, entre otros [2]. El resto de este documento se organiza de la siguiente manera: la sección II. presenta una

revisión de la tecnología MEMS. La sección III. presenta los aspectos principales del proceso de fabricación PolyMUMPs. En la sección IV. se aborda el problema del diseño parametrizado de engranes MEMS. Finalmente, la sección V. presenta conclusiones y trabajo futuro.

II. TECNOLOGÍA MEMS

Los procesos de fabricación de microsistemas incluyen etapas de depósito, fotolitografía y remoción de materiales. Cada una de estas etapas plantea en sí misma retos importantes.

DEPÓSITO DE MATERIALES

El depósito de materiales en capas delgadas, de 2 μ m a 5 μ m, y gruesas, de 6 μ m a 20 μ m, es una de las etapas cruciales en la creación de microsistemas. Los procesos más comúnmente utilizados para el depósito de materiales son el Depósito de Vapor Químico (Chemical Vapor Deposition CVD), el Depósito de Vapor Físico (Physical Vapor Deposition PVD), Electrodepósito y depósito por medio de rociado (Sputtering) entre otros [1], [2].

En el proceso CVD, un sustrato es expuesto a uno o más gases en una cámara de depósito. La reacción química de los gases provoca el depósito de materiales en forma monocristalina, policristalina, epitaxial y amorfa. Los materiales más comúnmente utilizados en este proceso son el silicio, carbono, tungsteno, nitrato de silicio y nitrato de titanio [1], [2].

En el proceso PVD, un sustrato es expuesto a un solo gas del cual se depositan una o varias capas mediante procesos de condensación. Los materiales utilizados en el proceso PVD incluyen cobre, silicio, aluminio, oro y otros [1], [2].

El Electrodepósito es similar al proceso de galvanización. Se utiliza una corriente eléctrica para depositar capas delgadas de un metal sobre una superficie. Controlando el tiempo de exposición de la superficie a la acción del Electrodepósito es posible controlar el grosor de las capas depositadas. Los materiales utilizados en Electrodepósito son comúnmente metales [1], [2]. El depósito de materiales por medio de Sputtering es una variante del proceso PVD, en el cual se inyectan átomos de material deseado a la cámara de depósito directo sobre el sustrato. La cámara de depósito está diseñada con paredes que atrapan aquellos átomos que “rebotan” sobre el sustrato y no se adhieren a éste [1], [2].

FOTOLITOGRAFÍA

La fotolitografía es un proceso que permite grabar patrones geométricos sobre materiales fotosensitivos. Las resoluciones típicas de los procesos de fotolitografía son del orden de 2µm y 4µm. Pero existen procesos submicrométricos, de 0.5µm y menores, generalmente utilizados en microelectrónica [3]. La fotolitografía es un proceso ampliamente difundido en la industria electrónica, la cual utiliza un haz de luz y una máscara en cristal para transferir formas geométricas sobre materiales fotosensitivos. Este material protege el patrón geométrico de la acción de ácidos que se aplican durante la etapa de remoción de materiales.

REMOCIÓN DE MATERIALES

Una vez transferido el patrón deseado, el siguiente paso consiste en remover el material no deseado. Existen básicamente dos procesos para remover materiales, conocidos como procesos húmedos y procesos secos. Los procesos húmedos consisten en sumergir el sustrato en una solución que disuelve el material no protegido por el fotoresistivo. Los procesos secos consisten en atacar el material no protegido por el fotoresistivo mediante iones reactivos o vapores disolventes.

Ambos procesos de remoción de materiales producen dos tipos de micromaquinado, clasificados como superficial y de volumen. En el primero, las estructuras alcanzan un grosor de 2µm a 5µm, mientras que en el último llegan hasta los 20µm [2], [4].

III. PROCESO DE FABRICACIÓN MULTIUSUARIO

Uno de los procesos comerciales de fabricación de microsistemas más difundidos es el Proceso de Fabricación Multiusuario PolyMUMPs. Este proceso emplea una técnica de micromaquinado superficial con tres capas de polisilicio mecánicamente identificadas como Poly0, Poly1 y Poly2. La capa Poly0 se encuentra anclada al sustrato mientras que las capas Poly1 y Poly2 son móviles. Finalmente, el proceso puede depositar una capa de 0.75µm de oro como material de terminación. El oro se emplea para realizar conexiones eléctricas entre microestructuras, además de ser utilizado como superficie reflejante para microespejos. En la Figura 1 se muestra la sección transversal de un rotor fabricado con tecnología PolyMUMPs [7]. El material del centro corresponde a la capa Poly2 y se encuentra anclado a Poly0. A su vez, la capa Poly0 está anclada al sustrato, por lo cual ambas capas no presentan movimiento. El material Poly1 central se encuentra libre y puede girar en torno al pivote en Poly2.

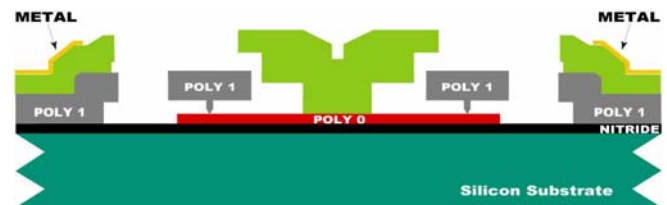


Figura 1: El Proceso de Fabricación PolyMUMPs

IV. DISEÑO PARAMETRIZADO DE ENGRANES

El engrane es un elemento ampliamente utilizado en la industria como mecanismo para transferir movimiento rotacional. Mediante arreglos de engranes, se puede multiplicar o dividir el movimiento [5]. Adicionalmente, con el uso de una cremallera, el movimiento rotacional se puede transformar en movimiento lineal o viceversa.

ENGRANES DE INVOLUTA

El engrane más utilizado es el de tipo involuta, debido a que minimiza la fricción entre los dientes de los engranes [5]. El perfil de un engrane de involuta se muestra en la Figura 2.

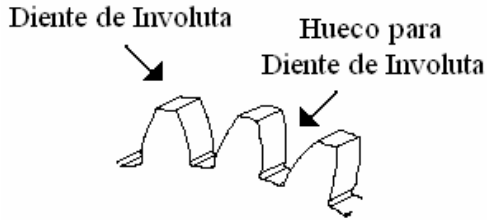


Figura 2: Perfil de un Engrane de Involuta

La forma peculiar de los dientes en este tipo de engranes, conocida como involuta, permite que exista en todo momento un solo punto de contacto entre los dientes de dos engranes, lo que reduce la fricción. Una involuta se forma cuando se traza una línea de tamaño fijo tangente a un círculo y luego se le hace girar entorno al punto de tangencia. La forma imaginaria que traza el extremo de esta línea se conoce como involuta [5]. La Figura 3 muestra una involuta.

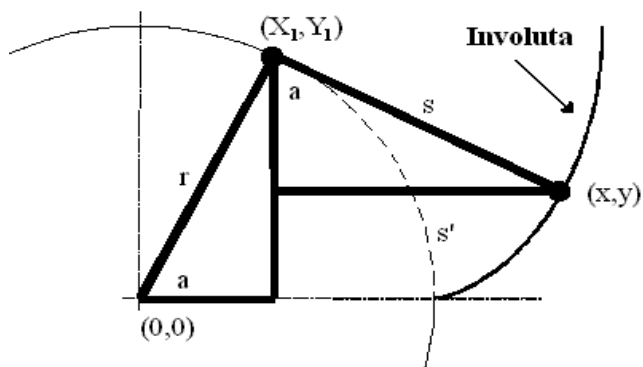


Figura 3: Involuta

La ecuación de la involuta define los valores del punto (x,y) en función del ángulo a, como sigue:

$$\begin{aligned} x &= X_1 + s * \sin(a) \\ y &= Y_1 - s * \cos(a) \end{aligned} \quad \text{Ecuación 1}$$

El punto (X₁, Y₁) en la Ecuación 1 se calcula como:

$$\begin{aligned} X_1 &= r * \cos(a) \\ Y_1 &= r * \sin(a) \end{aligned} \quad \text{Ecuación 2}$$

La Ecuación 2 es puramente trigonométrica y se deriva directamente de la Figura 3. Los parámetros requeridos para determinar la involuta se muestran en la **Figura 4**.

DISEÑO PARAMETRIZADO MODELOS SÓLIDOS

El primer paso para la creación de microestructuras tipo engrane con tecnología MEMS consiste en realizar un modelo sólido de la forma que se requiere que tenga. Se debe calcular el número de dientes que debe tener, su altura y separación, así como los diámetros interno y externo. Para esta tarea, lo más conveniente es utilizar una herramienta de diseño sólido como AutoCad para generar modelos tridimensionales.

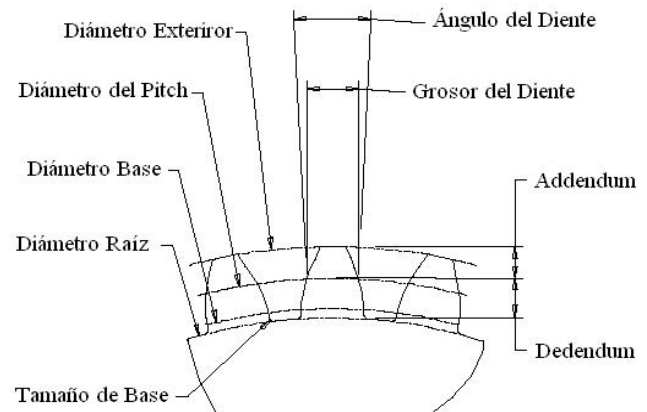


Figura 4: Parámetros de la Involuta

Sin embargo, aún con una herramienta como AutoCad, el diseño de un engrane de involuta es todo un desafío si se utilizan los métodos tradicionales de arrastrar, copiar y cortar. Afortunadamente, AutoCad permite el uso de lenguajes de programación como VisualBasic, AutoLisp y C++ para automatizar algunas de las tareas de diseño [6]. Con el uso de estos lenguajes, resulta sencilla la implementación de la Ecuación 1 y Ecuación 2 para la generación de los dientes de involuta en este tipo de engranes.

Aún más interesante resulta el hecho de que las funciones para generar los engranes pueden recibir parámetros como el tamaño de la base, el diámetro raíz, el diámetro base, el diámetro exterior y el número de dientes para generar automáticamente engranes de diferentes dimensiones de acuerdo a los parámetros proporcionados. La Figura 5 muestra un engrane de involuta de 16 dientes diseñado en AutoCad mediante automatización. La Ecuación 1 y Ecuación 2 permiten generar la forma del engrane en 2D. Al aplicar una operación de extrusión se obtiene el modelo en 3D. Las

operaciones de extrusión son funciones estándar soportadas por las herramientas de diseño sólido.

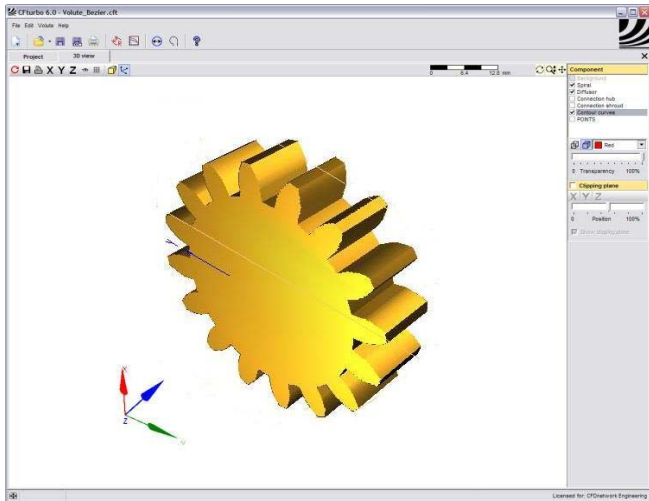


Figura 5: Diseño Sólido 3D del Engrane de Involuta

DISEÑO PARAMETRIZADO DE CADENAS DE ENGRANES

Encadenando engranes de involuta es posible construir mecanismos de transmisión de movimiento para multiplicar el movimiento rotacional [5]. Por ejemplo, la Figura 6, muestra un arreglo de dos engranes de 20 y 16 engranes respectivamente. Los engranes encadenados como los de la Figura 6 se relacionan de la siguiente manera:

Ecuación 3

$$T_R = \frac{GR_1}{GR_2}$$

donde GR_1 y GR_2 representan el número de dientes en el Engrane 1 y Engrane 2, respectivamente. De esta manera, si Engrane 1 gira n_1 vueltas, entonces el Engrane 2 gira $n_1 * T_R$ vueltas. Para el arreglo de engranes de la Figura 6 considerados de izquierda a derecha y utilizando la Ecuación 3, se tiene que $T_R=20/16=1.25$. Es decir, si el engrane de la izquierda gira 8 vueltas, entonces el engrane de la derecha gira 10 vueltas. Esto tiene el efecto de multiplicar el movimiento rotacional. Al considerar el arreglo en sentido inverso, es decir, de derecha a izquierda, $T_R=0.8$, lo que resulta en una división del esfuerzo.

Para un arreglo de dos engranes, como el mostrado en la Figura 6, las coordenadas de la involuta de ambos engranes se relacionan mediante la siguiente formula:

Ecuación 4

$$x_2 = R + (X_1 + s * \sin(a))$$

$$y_2 = D + (Y_1 - s * \cos(a))$$

El parámetro R en la Ecuación 4 representa la traslación horizontal del eje de un engrane respecto al otro y su valor se deriva directamente de la Ecuación 1 y Ecuación 4 una vez determinada la relación T_R en la Ecuación 3. El parámetro D permite trasladar verticalmente un engrane respecto al otro. Si $D=0$, entonces los engranes quedan alineados respecto a la vertical. Este procedimiento se puede generalizar para encadenar más de dos engranes.

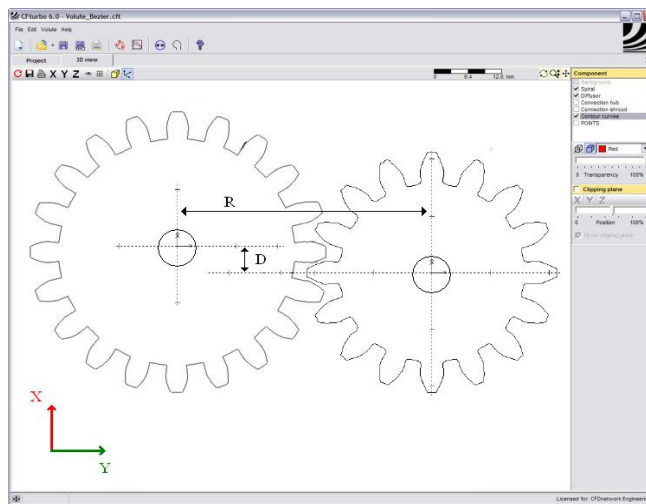


Figura 6: Sistema de Transmisión de Engranes de Involuta

GENERACIÓN DE MODELOS SÓLIDOS PARA SIMULACIÓN NUMÉRICA

Una vez generados los modelos sólidos, el siguiente paso consiste en realizar sobre ellos un análisis estructural. El análisis estructural permite estimar esfuerzos sobre el engrane, puntos de fractura y fatiga del material, entre otros [8]. Para este fin, se puede utilizar una herramienta de análisis de elemento finito como Ansys. Mediante la discretización en elementos finitos de un dominio particular, Ansys es capaz de aproximar la

solución a un sistema de ecuaciones diferenciales mediante diferentes métodos como el de los residuos [8].

Los pasos necesarios para la realización de un análisis de elemento finito son la generación del modelo sólido, la discretización del dominio, el ensamble de la matriz global del sistema, la solución del sistema global y el procesamiento posterior para visualizar los resultados. Todos estos pasos se pueden realizar de manera relativamente automatizada, excepto la generación del modelo sólido. Afortunadamente, las herramientas de diseño como AutoCad permiten exportar los modelos sólidos a herramientas de simulación numérica como Ansys. La Figura 7 muestra el modelo sólido para uno de los engranes de involuta exportado a Ansys desde AutoCad.

Mediante el uso de estos mecanismos de importación y exportación de modelos, es posible trabajar en el diseño de un único modelo sólido y compartirlo entre las distintas aplicaciones.

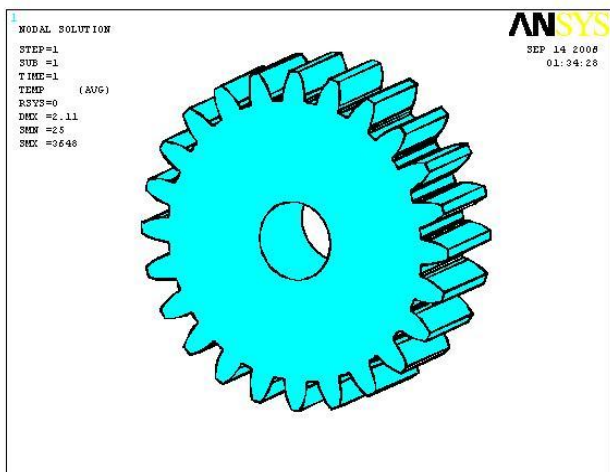


Figura 7: Modelo Sólido para Simulación en Ansys

Los resultados de las simulaciones obtenidos en Ansys se utilizan para retroalimentar el modelo sólido. De esta manera, si el diseño actual no cumple con algunos de los requerimientos de desempeño, entonces se modifica el diseño sólido y se vuelve a realizar el análisis estructural hasta obtener los resultados esperados.

GENERACIÓN DE MÁSCARAS DE FABRICACIÓN

Una vez que el diseño sólido ha quedado definido, el siguiente paso consiste en diseñar las máscaras

de fabricación. Las máscaras permiten transferir los patrones geométricos que definen el engrane o cadena de engranes al sustrato donde finalmente quedarán fabricados. Una ventaja adicional de utilizar una herramienta como AutoCad para generar el diseño del engrane es la posibilidad de exportar los modelos sólidos a herramientas de edición de máscaras de fabricación como LEdit de Tanner, Inc. Mediante el uso de formatos de intercambio de archivos, como CIF o GDSII, las herramientas de diseño pueden compartir modelos generados en una y otra herramienta. AutoCad permite exportar modelos sólidos a formato CIF, GDSII y otros. La Figura 8 muestra la máscara en Poly1 para el engrane de involuta de 16 dientes.

Uno de los trabajos más importantes en la generación de máscaras de fabricación es la fractura en polígonos de las figuras geométricas. Esto es necesario debido a que los procesos de fabricación de microsistemas encuentran dificultades para reproducir patrones geométricos con esquinas redondeadas. Por esta razón, el engrane de involuta de la Figura 8 debe aproximarse mediante polígonos. La tarea del diseñador de máscaras consiste en aproximar lo mejor posible, mediante polígonos, las figuras geométricas con esquinas redondeadas [3].

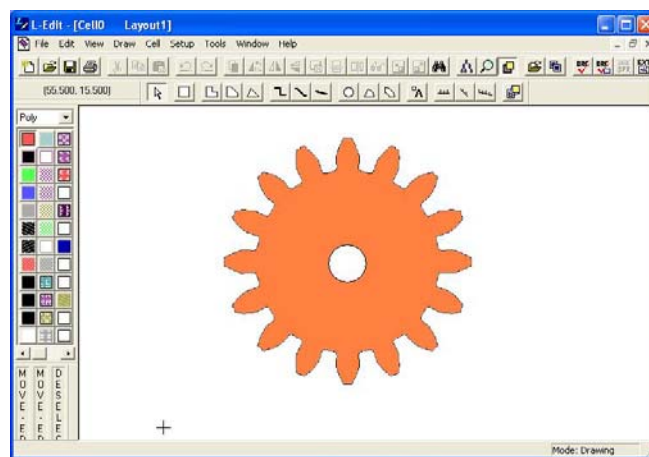


Figura 8: Máscara para Engrane en Capa Poly1

Cada proceso de fabricación impone restricciones propias respecto a la forma y tamaño mínimo de los polígonos utilizados para aproximar las figuras geométricas. Por ejemplo, el proceso de fabricación PolyMUMPs permite incluir hasta mil polígonos por figura geométrica. Una figura

geométrica que requiera más de esta cantidad de polígonos para su aproximación, deberá ser dividida por lo menos a la mitad [7].

V. CONCLUSIONES

Este artículo presenta una metodología para el diseño de microestructuras tipo engrane. El engrane diseñado es el de tipo involuta. El proceso de diseño inicia con la creación de un modelo sólido para el engrane de involuta en Autocad mediante automatización. Se utilizan los lenguajes de programación como AutoLisp y VisualBasic para crear funciones paramétricas que permiten crear engranes de diferentes tamaños basados en los parámetros proporcionados. Como segundo paso, se exporta el modelo sólido a una herramienta de simulación numérica como Ansys. Con el modelo en Ansys, se realiza análisis de elemento finito para validar el modelo sólido generado en Autocad. Se utiliza la información proporcionada por Ansys para ajustar el modelo en Autocad en caso necesario. Una vez que se ha construido el modelo del engrane en Autocad, se exporta en formato CIF o GDSII. Entonces se utiliza una herramienta de edición y generación de máscaras de fabricación como LEdit de Tanner, Inc. Las máscaras se fracturan en polígonos para su correcto procesamiento durante la fabricación de las estructuras en un proceso como PolyMUMPs.

VI. REFERENCIAS

- [1] Fundamentals of MICROFABRICATION: The Science of Miniaturization, Second Edition, by Marc J. Madou. Pub. CRC, 2nd edition, 2002. ISBN: 0849308267.
- [2] Micromachined Transducers Sourcebook, by Gregory T. Kovacs. Pub. McGraw-Hill; 1st edition, 1998. ISBN: 0072907223.
- [3] Microlithography: Science and Technology, 2nd Edition, by Kazuaki Suzuki and Bruce W. Smith. Pub. CRC. ISBN: 0824790243.
- [4] Microsystem Design, by Stephen D. Senturia. Pub. Springer; 1st edition, 2000. ISBN: 0792372468.
- [5] Handbook of Practical Gear Design, by Darle W. Dudley. Pub. CRC, 1994. ISBN:1566762189
- [6] Designing with Autocad,
- [7] El Proceso de Fabricación Multiusuario PolyMUMPs, <http://www.memscap.com/>.
- [8] The Finite Element Method and Its Application to Engineering using Ansys.

SISTEMA DE DETECCIÓN DE FALLAS EN CONECTORES PARA AMPLIFICADORES DE RF

José Jaime Esqueda Elizondo, Olivia Mendoza Duarte, Ivette Arely García Dávalos,
Francisco Roberto González Gutiérrez, Ruth Esther Ledesma Angulo, Daniel
Mendoza Damas, Shantal Alejandra Mora Camacho.

Área de Procesamiento de Señales, Facultad de Ciencias Químicas e Ingeniería.
Universidad Autónoma de Baja California
Calzada Universidad No. 14418. Mesa de Otay, C.P. 22390. Tijuana, B.C., México

TEL: (664)6821033, ext. 5800, correo-e: jjesqueda@uabc.mx, omendoza@uabc.mx,
ivette_arely@hotmail.com, t_rocket17@hotmail.com, mirocaescristo@hotmail.com, danymax25@hotmail.com,
shantalejandra@hotmail.com.

RESUMEN

El trabajo que se presenta es un sistema de inspección de piezas basado en reconocimiento de imágenes. El prototipo de dicho sistema incluye una aplicación desarrollada en el lenguaje de programación Visual Basic (perteneciente a Visual Studio 6.0 de Microsoft) y opera en una computadora personal. Dicho programa recibe imágenes por medio de una cámara conectada a la computadora por el puerto USB. El prototipo cuenta también con una plataforma que permite girar la pieza a inspeccionar para la adquisición de imágenes desde distintos ángulos con el propósito de que los resultados de la aplicación sean más confiables, mejorando el proceso de inspección, perteneciente al área del control de calidad. Como es de imaginar, los diferentes tipos de errores en las piezas a inspeccionar son muy variados, en este prototipo nos centraremos en los cuatro más frecuentes para un mejor análisis.

Palabras Clave: Control de calidad, inspección, reconocimiento de imágenes, redes neuronales, visión por computadora.

I. INTRODUCCIÓN

Se dice que la industria actual posee una necesidad de mejora continua. Ésta se debe, entre otras cosas, a la creciente importancia dada por el consumidor a la calidad de los productos, incluso por encima del precio. Lo anterior convierte al control de calidad en una etapa clave de todo proceso de producción.

La inspección consiste en la evaluación de la calidad de un producto mediante su comparación con estándares establecidos [5] y es un componente esencial de todos los sistemas de aseguramiento de la calidad. Se propone, pues, un sistema de inspección basado en reconocimiento de imágenes, el cual serviría como apoyo en el proceso del control de calidad.

Como caso de estudio, se tiene la inspección de amplificadores de radiofrecuencia (ver Fig. 1) en busca de defectos, proceso realizado actualmente en la industria de las comunicaciones.

En secciones posteriores del presente escrito se explicarán las diversas técnicas utilizadas para lograr el objetivo del sistema.



Fig. 1. Conectores para amplificadores de radiofrecuencia

II. FUNDAMENTOS DE IMÁGENES

Las imágenes llamadas *mapa de bits* o *bitmap*, son representadas por una matriz rectangular donde cada elemento es un píxel (abreviatura de *Picture Element*). Un píxel es la mínima unidad en un mapa de bits, se forma por la combinación de los 3 colores básicos rojo, verde y azul, *RGB*.

A continuación se describen algunas técnicas de procesamiento de imágenes utilizadas.

Recorte de la imagen

Una imagen tiene el tamaño en píxeles que la cámara tenga configurada, pero si la imagen necesita ser de un tamaño más pequeño o bien se necesita eliminar “basura”, se utiliza la técnica de corte de la imagen original. Esto es eliminando los píxeles no deseados tanto en el eje de las X y de las Y. (Ver Fig. 10.a)

Es necesario aplicar este algoritmo ya que entre menor sea el tamaño de píxeles o bien traducidos en matriz, el menor número de datos, menor es el tiempo de procesamiento en cada una de las imágenes.

Conversión a escala de grises

Así como un píxel tiene la combinación de R, G y B para formar un color, en la escala de grises el píxel utiliza matices de negro para representar objetos. Los objetos en escala de grises tienen un valor de brillo comprendido entre el 0% (blanco) y el 100% (negro).

Los valores en escala de grises deben ser enteros entre 0 y 255, donde el píxel tiene el mismo valor para los 3 canales. [2] (Ver Fig. 10.b)

Convolución bidimensional

La convolución bidimensional es la base de algunos procesamientos comunes, como el filtrado de imágenes. Dentro del campo del procesamiento de imágenes, la convolución se realiza entre la imagen y una matriz (los coeficientes del filtro) llamada máscara para filtrar una imagen. [1]

Filtros de Sobel

Es una técnica de filtrado para detección de contornos que consiste en 2 máscaras (o matrices) de 3 X 3 con los siguientes valores, $MH = [-1, -2, -1; 0, 0, 0; 1, 2, 1]$ y $MV = [1, 2, 1; 0, 0, 0; -1, -2, -1]$, éstos se aplican sobre la imagen original, la cual debe estar representada por una matriz cuadrada, sea en color o escala de gris.[3]

Fórmula de Intensidad (gradiente de nivel de gris)

El gradiente de una imagen $f(x, y)$ u operador de Sobel es aplicado a una imagen en escala de gris, es útil para calcular el gradiente de la intensidad de brillo de la imagen en cada píxel. El resultado muestra qué tan brusco o suave es el cambio de la imagen en ese punto, y determina si ese punto analizado representa un borde y hacia dónde tiende ese borde. [3]

Detección de contorno.

Un borde es la frontera entre dos regiones con propiedades de nivel de gris relativamente distintas, para detectar el contorno de una imagen en escala de gris, se aplican 2 convoluciones con los filtros de Sobel (máscaras MH y MV). Una vez obtenidas 2 matrices se les aplica el operador gradiente. (Ver Fig. 10.c)

Normalización

La normalización es una técnica para crear cierta independencia de las propiedades de la imagen, como lo son el brillo y el contraste, el uso de esta técnica permite restablecer los valores de la matriz en un rango definido, en el caso de las imágenes en escala de gris es de 0 a 255, para la cual se usa la siguiente ecuación: [4] (Ver Fig. 10.d)

$$N(x, y) = \frac{(N.Niveles - 1)}{(\max(I) - \min(I))} (I(x, y) - \min(I)) \quad (1)$$

donde:

- $I(x, y)$, nivel de gris de la imagen original en la coordenada (x, y) .
- $\min(I)$, $\max(I)$: mínimo y máximo nivel de gris en la imagen original.
- $N(x, y)$, nivel de gris de la imagen normalizada en la coordenada (x, y) .

III. REDES NEURONALES ARTIFICIALES

Una red neuronal artificial es un sistema de procesamiento, hardware o software, que imita esquemáticamente la estructura neuronal del cerebro para tratar de reproducir sus capacidades. [6]

La idea que subyace en los sistemas neuronales artificiales (o ANS, por sus siglas en inglés) es que, para abordar el tipo de problemas que el cerebro resuelve con eficiencia, puede resultar conveniente construir sistemas que copien en cierto modo la estructura de las redes neuronales biológicas, con el fin de alcanzar una funcionalidad similar.

Los sistemas neuronales artificiales cuentan con una estructura (ver Fig. 2) similar a la de su contraparte biológica. El elemento esencial de partida es la neurona artificial, la cual se organiza

en capas. Varias capas constituirán una red neuronal y por último, una red neuronal o un conjunto de ellas, junto con las interfaces de entrada y salida, más los módulos convencionales adicionales necesarios, constituirán el sistema global de proceso.

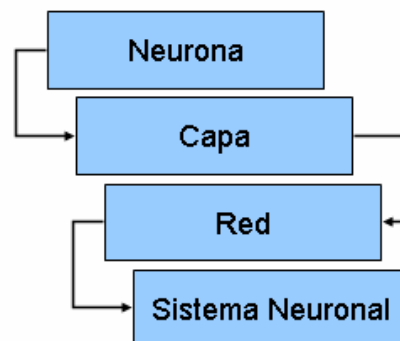


Fig. 2. Estructura jerárquica de un ANS

Funcionamiento

Clásicamente, se distinguen dos modos de operación en los sistemas neuronales: el modo recuerdo o ejecución, y el modo aprendizaje o entrenamiento. [6]

Para que una red neuronal resulte operativa es necesario entrenarla, lo que constituye la *fase de aprendizaje o convergencia*. En la mayoría de los casos, el aprendizaje consiste en determinar un conjunto de pesos sinápticos (que representan la intensidad de interacción entre neuronas) el cual permita a la red realizar correctamente el tipo de procesamiento deseado. El proceso es usualmente iterativo, actualizándose los pesos una y otra vez, hasta que la red neuronal alcanza el rendimiento deseado.

Generalmente, una vez que el sistema ha sido entrenado, el aprendizaje “se desconecta”, por lo que los pesos y la estructura quedan fijos, estando la red neuronal ya dispuesta para procesar datos. Este modo de operación se denomina *modo de recuerdo o de ejecución*.

IV. PROTOTIPO

Componentes del Sistema

El sistema propuesto está compuesto por los siguientes elementos: una cámara, una plataforma giratoria y una computadora (ver Fig. 3).

La cámara es el dispositivo utilizado para la entrada de datos al sistema.

La plataforma giratoria, como su nombre lo indica, tiene la particularidad de girar con la ayuda de un motor DC (DF10BE22-01). Este dispositivo es el que contiene la pieza a analizar, en el momento del reconocimiento.

La computadora muestra la interfaz gráfica que permite operar el sistema, guarda las muestras de las imágenes tomadas a las piezas y es la que realiza el proceso de evaluación de calidad de las piezas en tiempo real.

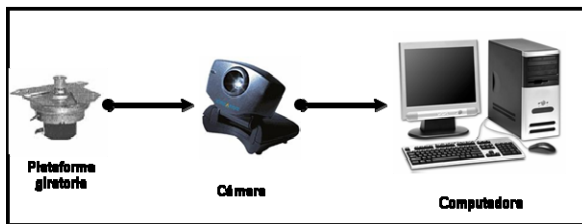


Fig. 3. Componentes del Sistema.

Proceso de identificación de piezas mediante redes neuronales

1. Muestras para entrenamiento

- Capturar imágenes de componentes de buena y mala calidad por separado.
- Hacer un pre-procesamiento de las imágenes por medio de detección de bordes, utilizando las técnicas explicadas anteriormente, para descartar el color de fondo o niveles de iluminación.
- Un segundo pre-procesamiento es realizar la normalización de la imagen. Esta función arroja como resultado el promedio de los tonos de gris cercanos a cero y una desviación estándar cercano a uno.

- Crear una sola matriz que contenga una columna por cada imagen con la que se cuenta.

2. Estructura y entrenamiento de la red neuronal

- El tipo de red neuronal utilizada es *feed-forward*, entrenada con un método supervisado de retropropagación del error.
- La estructura de la red neuronal es de 3 capas: 2 capas ocultas de 50 neuronas y la capa de salida de 2 neuronas, para indicar si la pieza es defectuosa o no.
- Los *datos de entrenamiento* utilizados son los de la matriz construida en el paso 1.
- Los *datos objetivo* para el entrenamiento supervisado están dados por una matriz identidad de 2 por 2, ya que la red neuronal clasificará en dos categorías.
- Como el entrenamiento de la red neuronal no es un proceso determinístico, es decir, el nivel de entrenamiento es variable debido a la aleatoriedad de los parámetros iniciales, se realizaron múltiples entrenamientos y se eligió el más acertado.

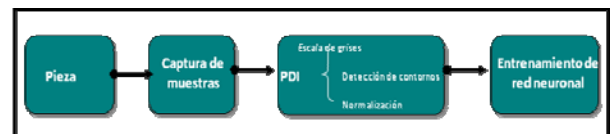


Fig. 4. Diagrama a bloques del proceso de entrenamiento de la red neuronal el cual se realiza como preparación.

3. Pruebas de clasificación

- Se elaboró un programa que realiza la prueba de clasificación de las piezas utilizando la misma matriz creada en el paso 1, sólo para asegurar el buen entrenamiento de las redes neuronales.
- Se realiza la identificación de las piezas en tiempo real.
- Para la presentación de los resultados finales se elaboró una interfaz gráfica. La aplicación del sistema está desarrollada en el lenguaje de programación Visual Basic perteneciente a Visual Studio 6.0. La interfaz de usuario se muestra en la Fig. 5.



Fig. 5. Interfaz Gráfica de Prototipo.

La interfaz está dividida en dos secciones, la de *muestras* y *resultados*.

La sección *muestras* permite realizar la captura de las imágenes de la pieza que se va a evaluar.

En *resultados*, una vez realizado el proceso de evaluación de calidad de la pieza que se está analizando, se muestra si ésta es de buena o mala calidad a través de un mensaje. En esta misma sección, se informa el estado de la pieza.

Simulación del proceso de inspección

En la plataforma giratoria del prototipo se colocan las piezas para su futuro análisis y destino (continuar en el proceso de producción o ser desechada).

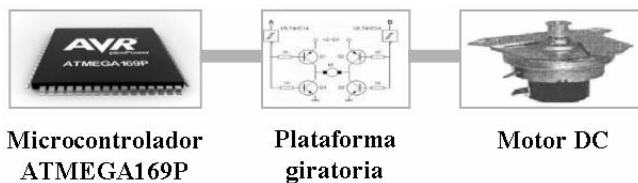


Fig. 6. Componentes del proceso de inspección.

Para cambiar de posición la pieza, se utilizó un motor de DC. Sobre el eje de dicho motor se monta la pieza para que gire, de tal manera que se pueda detener y avanzar el giro del motor conforme sea requerido.

Con este fin se incorporó un contador por software en el microcontrolador ATMEGA 169P,

donde un bit activará el motor hasta que el contador detenga el bit de activación y brindará una pausa suficiente para capturar una imagen, mediante la cámara. Posteriormente se reactivará el contador para repetir el proceso de movimiento de la pieza y la pausa para captura de imagen.

Se empleó el puente H en el C.I. L298 para una etapa de potencia capaz de generar el voltaje y la corriente suficiente para mover el motor.

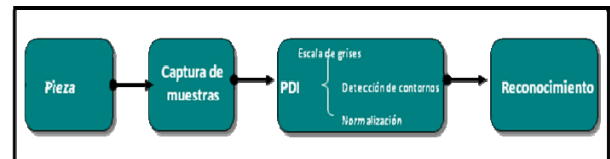


Fig. 7. Diagrama a bloques de la etapa de inspección de piezas.

V. APLICACIÓN

El sistema propuesto es considerado como de aplicación industrial pues, como se ha mencionado anteriormente, éste apoyaría en la fase de inspección del control de calidad en las industrias que requieran la detección de defectos. En particular interesan los defectos que pueden ser percibidos visualmente.

Se dice que hay dos aspectos de la inspección que se ven afectados por una variedad de factores relacionados con el desempeño del personal: la velocidad y la exactitud. La velocidad se refiere a la cantidad de tiempo que el inspector necesita para tomar la decisión de si el producto cumple o no con los requisitos establecidos. La exactitud es el resultado de los dos tipos de error que pueden darse en dicha decisión: que una unidad que sea conforme sea clasificada como no conforme y que una unidad que no sea conforme la clasifiquen como conforme. [5]

El sistema de detección de fallas mediante reconocimiento de imágenes permite mejorar ambos aspectos. Si en la actualidad, los inspectores necesitan corroborar visualmente el estado de los conectores (lo cual toma una cantidad de tiempo considerable), al implementar

el método propuesto el mismo operador manejaría el software, el cual mostraría el estado de la pieza en un monitor.

VI. RESULTADOS

En la presente sección se presentan algunos de los resultados obtenidos durante la realización de pruebas con el prototipo.

Se muestra a continuación una comparación del resultado del proceso de detección de bordes. La Fig. 8 muestra el resultado de dicho proceso en el sistema realizado, así como el procedimiento realizado con funciones predefinidas en el software MATLAB.

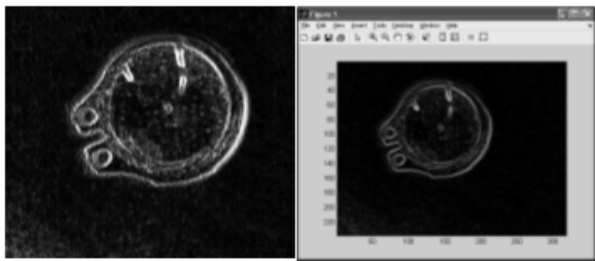


Fig. 8. Detección de bordes utilizando el prototipo (izquierda), y MATLAB (derecha)

El sistema guarda la imagen original capturada (Ver Fig. 9), tanto en mapa de bits como en una matriz de pixeles para el posterior procesamiento de dicha imagen.



Fig. 9. Imagen Original capturada (160 x 120)

En la Fig. 10 se pueden observar las diferentes y consecutivas etapas por las que la imagen tiene que atravesar para determinar el estado de la pieza. Dichas etapas fueron explicadas en la sección de II (Fundamentos de imágenes).

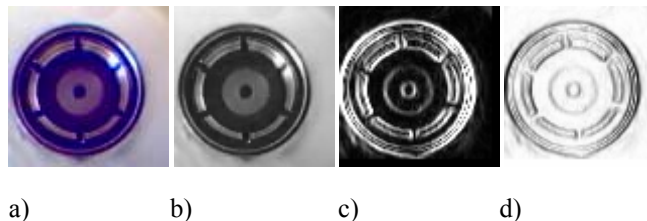


Fig. 10. Procesamiento Digital de Imágenes { a) Imagen recortada en 80 x 80, b) Imagen en Escala de Gris, c) Pieza Imagen con detección de bordes, d) Imagen Normalizada Inversa }

La siguiente es la pantalla de la interfaz gráfica actual, en la cual se muestran las imágenes obtenidas para su comparación con la base de conocimientos, así como el estado de la pieza inspeccionada en ese momento, que en éste caso es aceptable. Esto se muestra con el mensaje “Estado de la pieza: Aceptable”. En caso de presentar algún defecto (Ver Fig. 11), el mensaje correspondiente será “La pieza se considera en mal estado”.



Fig. 11. Resultado del procesamiento de la imagen en el cual la pieza se consideró “aceptable”.

Como es de comprender, en el proceso de elaboración de piezas pequeñas y complejas como las que se están manejando, hay una gran variedad de defectos. Debido a que este sistema hasta el momento es un prototipo, se pretende enfocar en los 4 tipos de defectos más frecuentes, (Ver la Fig. 12), esto es con el fin de optimizar el tiempo de respuesta del sistema, dado que cada imagen representa un conjunto de datos que

repercuten en el tamaño de la base de conocimientos del sistema (explicado en la sección III, Redes neuronales) haciendo que el procesamiento sea mayor.

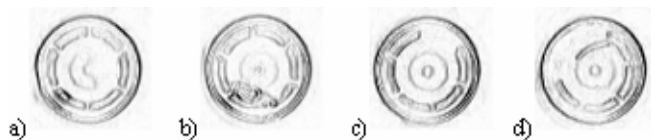


Fig. 12. Tipos de piezas defectuosas que el sistema reconoce { a) Pieza incompleta del centro, b) Pieza con limadura de hierro, c) Pieza incompleta de una placa lateral, d) Pieza con placa lateral volteada

VII. CONCLUSIÓN

El sistema de detección de fallas mediante reconocimiento de imágenes responde a la necesidad de innovación en los procesos de producción de la industria actual, en especial en el control de calidad.

La implementación del prototipo es una propuesta para la mejora de los procesos de control de calidad, la cual hace uso de técnicas de cómputo actuales, específicamente en el área de inteligencia artificial y procesamiento digital de imágenes. Es posible que de ser implementado el sistema y en caso de obtenerse buenos resultados, las industrias se vean motivadas a utilizar nuevas tecnologías para su beneficio y el de los consumidores.

Se cree que los sistemas de visión por computadora serán ampliamente usados conforme la lógica difusa, las redes neuronales y demás ramas de la inteligencia artificial mejoren su confiabilidad.

VIII. REFERENCIAS

[1] Esqueda, J., Palafox, L., *Fundamentos de Procesamiento Digital de Imágenes*, Editorial Universitaria UABC, Mexicali, 158 pp, (2006).

[2] Frederic, P., (12 de Julio del 2007), <http://yov408.free.fr>.

[3] Pájares, G., De la Cruz, J., Molina, J., Cuadrado, J., López, A., *Imágenes Digitales: Procesamiento práctico con Java*, RA-MA Editorial, Madrid, 193 pp, (2004).

[4] Pérez de la Blanca, N., (8 Agosto 2007), <http://ftp.uniovi.es>.

[5] Pyzdek, B., *Manual de Control de Calidad en la Ingeniería*, McGraw Hill, México D. F., 290 pp, (1996).

[6] Martín del Brío, B., Sanz, A., *Redes Neuronales y Sistemas Borrosos*, Alfaomega, México D. F., 404 pp, (2007).

Análisis de la Técnica de Aprendizaje Basada en Imagen

J.A. Alonso-Castro, K. Ramírez-Amaro y J.C Chimal-Eguía.

Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Bátiz s/n Unidad Profesional Adolfo López Mateos
Col. Nueva Industrial Vallejo, México, D.F. 07738, México
acastr07@sagitario.cic.ipn.mx, kramirez@ipn.mx, chimal@cic.ipn.mx

RESUMEN

En este artículo, se analiza una técnica desarrollada por Ramírez et al [3] para aprender el comportamiento dinámico de las series de tiempo a través de la información obtenida de su eje imagen. Dicha técnica consiste en dividir la serie de tiempo en regiones llamadas cajas, las cuales proporcionan información independiente sobre la serie de tiempo. En este trabajo se analizarán diferentes tipos de división de la imagen de la serie de tiempo, la imagen en la técnica se considera como el eje vertical o eje y de una serie de tiempo. Para demostrar el alcance y potencialidad de la técnica, probamos el rendimiento de la misma con diferentes series de tiempo que muestran comportamientos dinámicos desde periódicos hasta estocásticos [9]. El aprendizaje que se obtiene al analizar una serie de tiempo con la técnica propuesta puede servir para diversas aplicaciones como por ejemplo la predicción.

Palabras clave: Series de tiempo, aprendizaje, cajas, amplitud.

I. Introducción

Antes de mencionar los parámetros de entrada y analizar el aprendizaje que se obtiene de la técnica basada en imagen para cada uno de los diferentes tipos de división de la imagen de la serie de tiempo, es necesario dar una definición formal de lo que es una serie de tiempo.

Una serie de tiempo [1] [9] es un conjunto de observaciones que expresan la dinámica de las

variables de un sistema. Así, una serie de tiempo s se puede definir como un conjunto de valores reales consecutivos $s = \{x_t \in \mathbb{R} | x_1, x_2, \dots, x_n\}$ [1], los cuales se encuentran en un intervalo $t = n$, que cumpla la siguiente condición:

$$t_1 < t_2 < t_3 < \dots < t_n \quad (1)$$

Un ejemplo de serie de tiempo lo podemos observar en la figura 1, donde el eje y, representa la amplitud o imagen de la ST y el eje x representa al tiempo t .

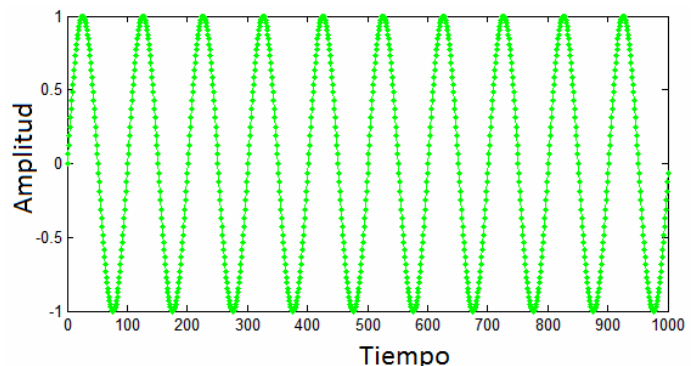


Figura 1. Ejemplo de serie de tiempo (seno).

Las ST contienen información sobre las variables independientes de un sistema que determinan su dinámica. La extracción de esta información es un problema que se estudia mediante el análisis (caracterización), predicción y modelado de las series de tiempo [2] [5].

En el trabajo presentamos la siguiente organización: en la sección II es presentada la metodología para representar la información, en la

sección III es explicada la metodología para dividir la imagen de la serie de tiempo, en la sección IV se da una breve explicación de la técnica de aprendizaje y finalmente en la sección V se muestran los resultados obtenidos.

II. Representación de la Información

Para comprender la técnica es necesario definir los datos de entrada para el proceso de aprendizaje y algunos conceptos importantes que nos ayudarán a entender el proceso de división de la serie de tiempo, la cual es la base de realizar un óptimo aprendizaje del comportamiento de la serie de tiempo. La técnica de aprendizaje consiste en dividir la amplitud de la serie de tiempo en un determinado número de cajas [3]. Una caja o **[Box]** es un vector de intervalos [7] y se puede expresar matemáticamente de la siguiente manera:

$$[Box] = [y_1, \bar{y}_1] \times [y_2, \bar{y}_2] \times \dots \times [y_N, \bar{y}_N]. \quad (2)$$

De acuerdo a lo anterior, podemos definir lo siguiente:

Definición 1.1. El área_f de una caja es posible definirla mediante la siguiente función [3]:

$$\text{área}_f = [Box] \rightarrow N \cup \{0\} \cup \{\infty\}. \quad (3)$$

Como resultado la nueva función área_f tiene como dominio los intervalos de la imagen $[y_1, \bar{y}_1]$ de la función f y su contradominio son los números N incluyendo el cero y el infinito. Entonces, la función área_f se puede escribir de la siguiente manera [3]:

$$\text{área}_f [Box] = \text{card}([y_1, \bar{y}_1] \cap [x_1, \bar{x}_1]), \quad (4)$$

donde card indica el número de puntos que se encuentran en el área_f de cada caja o vector de intervalos. La expresión anterior define la cantidad de puntos que se encuentran dentro de cada caja (vector de intervalos). En la figura 2 se muestra el área de una caja.

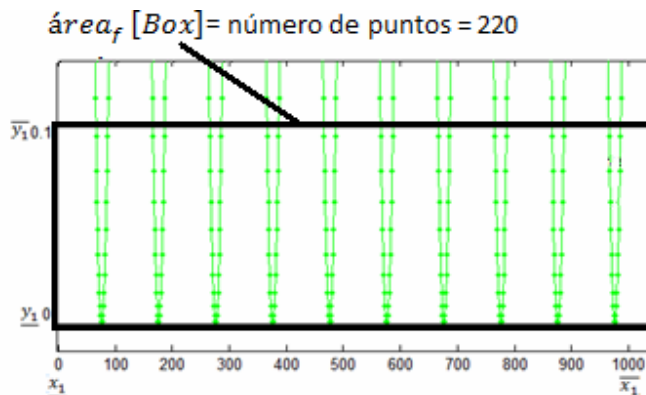


Figura 2. Ejemplo de puntos contenidos dentro de una caja (área_f).

Para realizar el algoritmo de aprendizaje encontramos parámetros característicos dentro de cada caja los cuales nos dan información de la dinámica de la serie de tiempo. Los parámetros que se consideraron dentro de cada caja son los siguientes:

Tendencia: Este parámetro indica la dirección que toman los puntos en la serie de tiempo [8] [3]. El parámetro tiene un valor binario, es decir, cuando la dirección de los puntos es descendente se toma un valor de 0 y toma un valor de 1 cuando la dirección de los puntos es ascendente.

Visita: Este parámetro indica el número de veces que un conjunto de puntos entra en una misma caja y con la misma tendencia [8] [3]. Para ejemplificar lo anterior se puede observar la figura 3b) la cual muestra dos rectángulos punteados, el primer rectángulo punteado contiene los puntos $x(1)$ hasta $x(23)$, este conjunto de puntos representa la visita 1, se vuelve a presentar la misma tendencia por segunda ocasión en los puntos $x(27)$ hasta $x(30)$ por lo cual el conjunto de puntos en este intervalo de tiempo representa la visita 2 y así sucesivamente.

En la figura 3, se muestra un esquema que representa la información obtenida de la imagen o

amplitud de la serie de tiempo, en la figura 3a) se muestran los puntos de la serie de tiempo, y en la figura 3b) muestra la división de la amplitud de la serie de tiempo en cajas, en la misma figura se observa la división de la imagen de la serie de tiempo en 3 cajas. Finalmente, la figura 3c) muestra los diferentes datos de entrada (tendencia, visita, número de caja) para los puntos de la serie de tiempo, los cuales están ubicados en diferentes cajas con tendencia de 0 (descendente) y visita de 1.

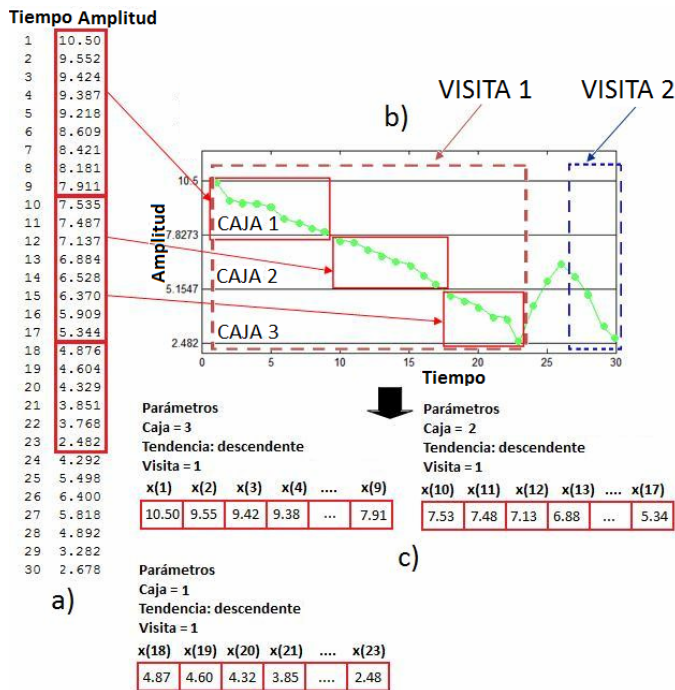


Figura 3. Esquema de representación de la información para la técnica TABI.

III. Metodología para dividir la imagen de la serie de tiempo

Los conceptos anteriores nos permiten conocer información local de la serie de tiempo y poder analizarla. La técnica de aprendizaje basada en la imagen (TABI) de la ST [3], toma como idea fundamental la metodología de submetas [10] (dividir un problema difícil en varios problemas fáciles) y divide la serie de tiempo en cajas [7], las cuales contienen un determinado número de puntos contenidos en cada una de ellas.

Uno de los principales pasos para realizar un buen aprendizaje del comportamiento de la serie de tiempo es realizar una división óptima de imagen de la serie de tiempo. Por lo que se realizó un análisis de cada uno de los tipos de división propuestos originalmente para la técnica “TABI” con el propósito de conocer cuáles son los tipos de división ideales para obtener mejores resultados en el proceso de aprendizaje, a continuación se explican los tipos de divisiones con los que se realizaron los experimentos.

Misma anchura. Como su nombre lo indica, la imagen de la serie de tiempo o amplitud se divide en N número de cajas y cada caja tiene un intervalo con la misma anchura. La anchura de cada caja se genera por la siguiente expresión:

$$a(y_1, y_2) = \frac{y_2 - y_1}{N}, \quad (5)$$

donde y_1, y_2 son cotas de la imagen y N indica el número total de cajas en las cuales será dividida la imagen de la serie de tiempo.

En la figura 4 se muestra un ejemplo de la división con misma anchura, donde suponemos que el número de cajas es $N = 3$, entonces el cálculo de la anchura para cada caja es:

$$a(2.482, 10.500) = \frac{10.500 - 2.482}{3} = 2.672.$$

Por lo que obtenemos tres intervalos para cada caja los cuales son: Caja1 (2.482,5.154), Caja2 (5.154,7.827), Caja3 (7.827,10.500).

Una de las desventajas que presenta este tipo de división, es la distribución de los puntos por cada caja, que en la mayoría de veces presentaba una distribución Gaussiana, es decir se podía observar que al graficar las frecuencias por cada caja de la imagen de la serie de tiempo, las cajas que se localizan al centro de la serie de tiempo contienen la mayor concentración de puntos, lo que significa que presentaban una frecuencia mayor de puntos que las cajas de las orillas.

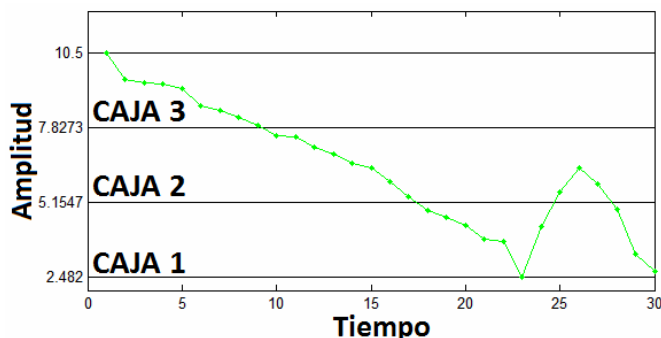


Figura 4. División misma anchura de la ST.

Por frecuencias. Este tipo de división, se realiza de la siguiente manera: tomando la ecuación (4) donde se define el concepto área_f , podemos asignar las cotas superior e inferior de cada caja de una manera dinámica, por lo que la principal diferencia con la división anterior es que la división por frecuencias se realiza por el número de puntos contenidos dentro de cada caja y no de acuerdo a su anchura. Realizando esta división obtenemos que cada caja contiene un número de puntos uniforme o información similar al área propuesta (área_f) [8].

Antes de definir las cotas que tendrán las nuevas cajas es importante definir el área_f . Por lo cual, primero tendremos que realizar la división de las cajas por misma anchura obteniendo de esta forma las frecuencias de cada caja, donde la frecuencia se puede expresar de la siguiente forma:

$$\text{Frec}(i) = \text{área}_f(y_{1i}, y_{2i}) \text{ para } i = 1, 2, \dots, N \quad (6)$$

Donde $\text{Frec}(i)$ es un vector que contiene las frecuencias acumuladas de los puntos que pertenecen a cada caja (y_{1i}, y_{2i}), las cuales son calculadas dividiendo la serie de tiempo en N cajas por medio de la división de misma anchura. Obteniendo las frecuencias de cada caja el número de puntos adecuado para las nuevas cajas, se realiza mediante el cálculo del área_f , de acuerdo a la siguiente expresión:

$$\text{área}_f = \frac{\max(\text{Frec}(i)) + \min(\text{Frec}(i))}{2}, \quad (7)$$

donde $\max(\text{Frec}(i))$ se refiere a la caja con mayor cantidad de puntos en la división realizada con misma anchura y $\min(\text{Frec}(i))$ se refiere a la caja con menor cantidad de puntos en la división con misma anchura. Por ejemplo, tomando la serie de tiempo de la figura 4, donde se dividió la imagen de la serie de tiempo en 3 cajas se obtiene que la $\max(\text{Frec}(i)) = 11$ y la $\min(\text{Frec}(i)) = 9$. Así entonces, se obtiene un $\text{área}_f = 10$ puntos dentro de cada caja. Por lo que cada caja, al definir su cota inferior y superior, tendrá que contener un número aproximado de 10 puntos, como resultado se obtendrá una distribución de frecuencias de puntos dentro de cada caja uniforme y las cotas serán: Caja1 (2.482,4.881), Caja2 (4.881,7.141), Caja3 (7.141,9.621), Caja4 (9.621,10.500).

Por porcentaje de información total. Un tercer tipo de división llamada “Por porcentaje de información total”, en la cual como su nombre lo indica, determina el número de puntos dentro de cada caja mediante un cierto porcentaje aplicado al número total de puntos de la serie de tiempo.

El cálculo de la nueva área_f por este tipo de división es obtenido de la siguiente manera [8],

$$\text{área}_f = \text{size}(ST) * (x/100), \quad (8)$$

donde $\text{size}(ST)$ corresponde al número total de puntos de la serie de tiempo y $(x/100)$ expresa la cantidad de porcentaje que se tomará de la información total. Por ejemplo, tomando la serie de tiempo de la figura 4, donde el número de puntos de la serie de tiempo es igual a 30, se obtiene el $\text{área}_f = \text{size}(30) * (10/100) = 3$ puntos por cada caja, donde el 10/100 representa el 10% de la información total de la serie de tiempo y las nuevas cotas de las cajas serán: Caja1 (2.482,3.551), Caja2 (3.551,4.301), Caja3(4.301,4.878), Caja4(4.878,5.680), Caja5 (5.680,6.320), Caja6 (6.320,6.890), Caja7 (6.890,7.693), Caja8 (7.693,8.495), Caja9 (8.495,9.421), Caja10 (9.421,10.500).

IV. Técnica de aprendizaje basada en su imagen

Ahora bien, deseamos obtener información de la serie de tiempo de tal manera que podamos aprender su comportamiento dinámico, para ello analizamos los tipos de división de la técnica TABI, la cual propone un algoritmo probabilístico [2].

El algoritmo que realiza el aprendizaje de la técnica TABI es el siguiente y se observa en la Figura 5 [2].

Paso 1: Recorrer la serie de tiempo original desde el punto inicial de la ST hasta el punto final.

Paso 2: Identificar la caja a la cual pertenece el punto original de la ST.

Paso 3: Determinar la tendencia de los puntos de la serie de tiempo.

Paso 4: Si es la primera visita realizar el *Paso 5*, en otro caso seguir con el *Paso 6*.

Paso 5: Obtener el aleatorio ponderado $\tilde{x} = rand(y_{1t}, y_{2t})$ y guardarlo en la matriz U_a y saltar al *Paso 7*.

Paso 6: Leer el último error de la Matriz U_e si cumple con la siguiente condición: último error > 0.009 | último error ≤ -0.009 significa que el punto aprendido anteriormente en la pasada visita no fue óptimo, por lo tanto saltar al *Paso 5*. De otra forma el punto aprendido tiene un valor óptimo similar al punto original de la serie de tiempo y realizamos el *Paso 7*.

Paso 7: Calcular Error de aprendizaje, el cual se obtiene de la diferencia del punto generado (U_a) con el punto original de la serie de tiempo y guardarlo en la matriz U_e , regresar al *Paso 1* para tomar el siguiente punto de la serie de tiempo.

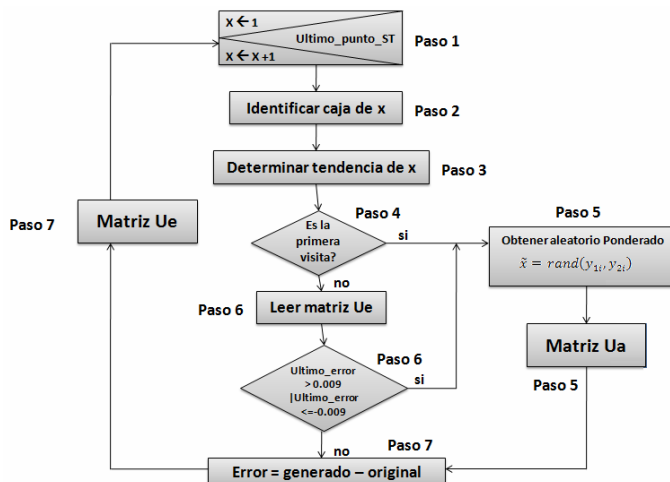


Figura 5. Proceso de aprendizaje realizado con la técnica TABI.

V. Resultados

Nuestra hipótesis es que los distintos tipos de división influyen en el proceso de aprendizaje. Debido a lo anterior analizamos los errores locales de cada caja que genera la técnica TABI con series de tiempo de diferente tipo de comportamiento como por ejemplo: periódico, cuasi-periódico, caótico, complejo y estocástico [5][6]. Por motivos de espacio no es posible mostrar todos los experimentos realizados, así que únicamente mostramos algunos resultados que consideramos más significativos del análisis de la técnica.

La metodología que se usó para el análisis de la técnica de aprendizaje es la siguiente:

1. Las series de tiempo fueron normalizadas dentro del intervalo $[0,1]$.
2. Posteriormente por cada serie de tiempo se obtuvo el aprendizaje utilizando los tres tipos de división mencionados en la sección III y por cada análisis se obtuvo lo siguiente:
 - a) Una gráfica donde se observa la serie de tiempo original contra la ST aprendida.
 - b) Una gráfica de errores calculados por cada caja, que nos permite observar cómo influyen estos en el aprendizaje.

- c) Una gráfica que muestra un acercamiento a la primera visita de una determinada caja de la ST.
- d) Una gráfica que muestra un acercamiento a la última visita de una determinada caja de la ST.

En la figura 6, observamos los resultados de aplicar el tipo de división de porcentaje de la información total de la serie de tiempo seno, la cual muestra un comportamiento periódico. Primeramente realizamos el aprendizaje de la ST con la técnica TABI, y se puede observar que el seguimiento de los puntos aprendidos con respecto a los puntos de la serie de tiempo original se realiza de una manera muy buena, posteriormente graficamos los errores locales de la caja 3 con tendencia descendente los cuales se muestran en la figura 6b) y podemos observar que los primeros puntos calculados de la primera visita presentan errores mayores pero conforme van evolucionando los puntos en la visita, disminuye el error y tiende a converger a cero. A partir de la visita 3, hasta la última visita se observa que los errores muestran un patrón parecido, esto significa que se tiene un valor óptimo en los puntos aprendidos y no es necesario recalculer nuevamente los puntos.

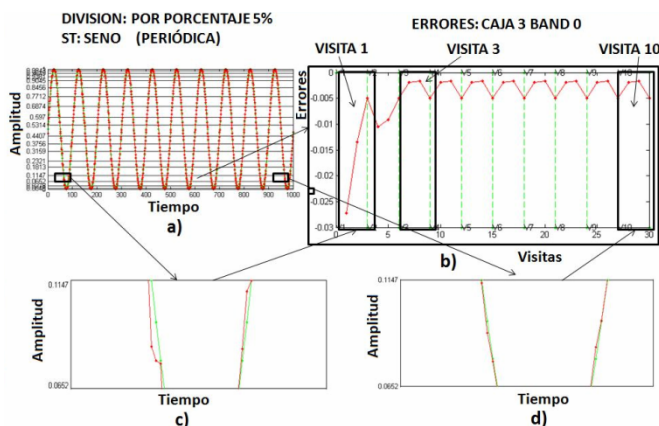


Figura 6. a) Gráfica de aprendizaje realizado a la serie de tiempo seno. **b)** gráfica de errores de la caja 3 con tendencia descendente, en c) se observa un acercamiento realizado para la primera visita de la caja 3 de la ST y d) muestra la última visita donde se observa que se aprendió el comportamiento de la ST.

La figura 7 muestra los resultados obtenidos de aplicar la división de porcentaje de la información total a la serie de tiempo Lorenz, la cual muestra un tipo de comportamiento caótico, el porcentaje que se utilizó para la división de la imagen de la ST fue del 5%, el cual muestra muy buenos resultados en el aprendizaje [3]. En la figura 7a) se muestra el seguimiento realizado a la serie de tiempo original el cual es óptimo y en la figura 7b) se observa la gráfica de errores de la caja 1 de la imagen de la ST con tendencia descendente, se puede ver en la gráfica de errores que la evolución de los puntos depende del punto anterior calculado, al ser una técnica probabilística mientras mejores resultados tengan los puntos calculados anteriormente se mostrara una mejor evolución de los puntos en las visitas sucesivas, también se puede observar que los errores oscilan mucho debido al comportamiento de la serie de tiempo. En la figura 8c) se muestra los puntos generados en la primera visita, posteriormente en la figura 8d) se puede apreciar como los puntos generados por el algoritmo han aprendido el comportamiento de la ST.

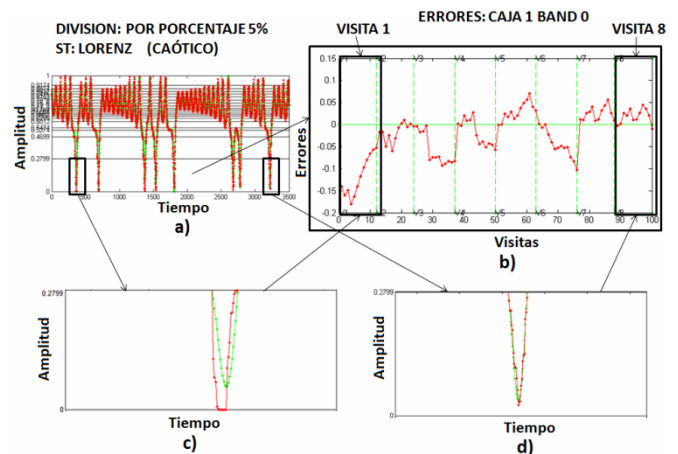


Figura 7. a) Gráfica de la serie de tiempo Lorenz con el tipo de división de porcentaje de la información total, se muestra el aprendizaje realizado. **b)** gráfica de errores de la caja 1 con tendencia descendente. **c)** se observa un acercamiento realizado para la primera visita de la caja 1 de la ST y d) muestra la última visita donde se observa que se aprendió el comportamiento de la ST.

Otro ejemplo de tipo de división con la que se analizó la serie de tiempo Lorenz fue la división por frecuencias, la cual se puede observar en la figura 8, el proceso de aprendizaje se realiza de una manera óptima semejante a la división por porcentaje, en la figura 8a) se observa el seguimiento que realiza de la serie de tiempo original, en la figura 8b) se observa la gráfica de los errores de la caja 1, al revisar la gráfica encontramos que se presentan errores mayores y conforme va evolucionando el seguimiento de esa visita el valor de los errores disminuye, en la figura 8c) se muestra los puntos generados en la primera visita, posteriormente en la figura 8d) se puede apreciar como los puntos generados por el algoritmo han aprendido el comportamiento de la ST, en la grafica se puede observar que no realiza un seguimiento perfecto de los puntos de la serie de tiempo original pero la amplitud entre cada punto es similar. Es posible observar que al ejecutar este tipo de división, la precisión con la que se realiza el aprendizaje no es tan buena como en la división por porcentaje de la información total.

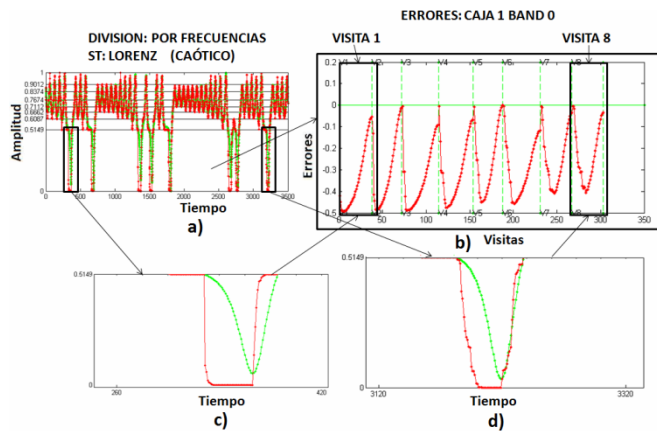


Figura 8. a) Gráfica de la serie de tiempo Lorenz con el tipo de división por frecuencias, en la cual se muestra el aprendizaje realizado. b) gráfica de errores de la caja 1 con tendencia descendente. c) se observa un acercamiento realizado para la primera visita de la caja 1 de la ST y d) muestra la última visita donde se observa que la amplitud de los puntos aprendidos es similar a los puntos originales de la ST.

La figura 9 muestra los resultados obtenidos de aplicar la división de porcentaje de la información total a la serie de tiempo Ruido Blanco (white noise) la cual muestra un tipo de comportamiento estocástico [3], el porcentaje que se utilizó para la división de la imagen de la ST fue del 5%. En la figura 9a) se muestra el seguimiento realizado a la serie de tiempo original, la cual a pesar de tener un comportamiento difícil de aprender y con una tendencia aleatoria, lo realiza de una manera óptima. Posteriormente en la en la figura 9b) se observa la gráfica de errores de la caja 1 de la imagen de la ST con tendencia descendente, se puede ver en la gráfica que los errores oscilan de una manera constante, debido a que la serie de tiempo en esa caja muestra un punto por cada visita, el cual se ubica en diferente posición en la amplitud de la caja 1 en cada una de las visitas.

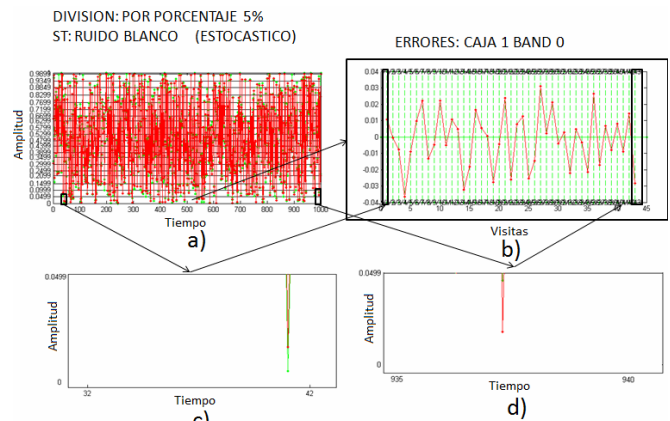


Figura 9. a) Gráfica de la serie de tiempo ruido blanco (White noise) con el tipo de división de porcentaje de la información total, se muestra el aprendizaje realizado. b) gráfica de errores de la caja 1 con tendencia descendente. c) se observa un acercamiento realizado para la primera visita de la caja 1 de la ST y d) muestra la última visita y el punto aprendido en dicha visita.

VI. Conclusiones

En este artículo hemos presentado un análisis del funcionamiento de la técnica TABI con los 3 tipos de división propuestos originalmente [3], el análisis nos permitió obtener la información

relevante para la técnica y extraer patrones de comportamiento en el proceso de aprendizaje, con la finalidad de proponer en un trabajo futuro nuevos tipos de división de la imagen de la ST para obtener mucho mejores resultados en el aprendizaje.

Por medio del análisis de los errores por cada caja se obtuvieron diferentes características de comportamientos en la evolución del aprendizaje que nos permiten entender mejor el funcionamiento de la técnica explorada, este trabajo es importante debido a que los buenos resultados de la técnica dependen del tipo de división que se realice, con lo cual podemos tener una buena representación de la información. Por medio de los experimentos realizados se observó que de los 3 tipos de división mencionados en la sección III, la división por porcentaje de la información total con un porcentaje de 5% tiene los mejores resultados en el aprendizaje que los otros tipos de divisiones.

En este artículo analizamos con la técnica TABI el seguimiento del comportamiento dinámico de una serie de tiempo. Para reflejar los resultados obtenidos en este proceso es necesario enfocarlo a una aplicación específica que permita comparar los resultados con otras técnicas existentes. No podemos realizar una comparación del aprendizaje o fase de entrenamiento con otras técnicas clásicas debido a que presentan procesos de entrenamiento diferentes a la técnica propuesta [3]. Pero si es posible realizar una comparación entre esta nueva forma de representar los datos con técnicas clásicas, enfocándonos a alguna aplicación como lo es la predicción, en la cual se obtienen mejores resultados que las técnicas clásicas [3].

VII. Referencias

- [1] Bautista-Thompson, E., Guzmán-Ramirez E. y Figueroa-Nazuno J. *Predicción de Múltiples Puntos de Series de Tiempo Utilizando Support Vector Machines*, Centro de Investigación en Computación, IPN, 2004.
- [2] Ramírez-Amaro, K. y Figueroa-Nazuno, J.G. *Empleo de la Técnica Mapa Recurrente en el Análisis de los Índices de Teleconexión Atmosféricos*. Centro de Investigación en Computación, IPN, 2006. Reporte Técnico. ISBN: 970-36-0330-0.
- [3] Ramírez-Amaro, K. *Técnica de Aprendizaje de Series de Tiempo Estacionarias a partir de la información de su Imagen*, Centro de Investigación en Computación (CIC-IPN). Tesis de Maestría en Ciencias de la Computación, 2007.
- [4] Michie D., Chambers R.A. *Boxes: An experiment in Adaptive Control*. Eds. Dale E. y Michie D. Machine Intelligence 2. 1968.
- [5] Kantz, H. y Schreiber, T. *Nonlinear Time Series Analysis*. Cambridge University Press, 1997.
- [6] Sprott, J.C. *Chaos and Time-Series Analysis*. Oxford University Press, 2003.
- [7] Jaulin L., Kieffer M. y Didrit O., Walter E. *Applied Interval Analysis*. Springer-Verlag London, 2001.
- [8] Ramírez-Amaro, K. y Chimal-Eguía, J.C. *New Learning Technique to time Series Forecasting*, Centro de Investigación en Computación (CIC-IPN). 2006.
- [9] Alonso-Castro, J.A., Ramírez-Amaro, K. y Chimal-Eguía, J.C. *Técnicas no lineales para el análisis de series de tiempo*. Centro de Investigación en Computación, IPN, 2006. Reporte Técnico.
- [10] Newell A., Shaw J.C., Simon H.A. *A Variety of Intelligent Learning*. Eds. Marshall, C. Yovits y Cameron, S. London: Pergamon: in a general problem solver in Self-organizing Systems.

“SISTEMA DE IDENTIFICACIÓN DE ROSTROS HUMANOS A TRAVÉS DE REDES NEURONALES”

Andrés Ferreyra Ramírez, Javier Alducin Castillo, Marco Antonio Sánchez

Área de Instrumentación, Departamento de Electrónica, División de CBI
Universidad Autónoma Metropolitana--Azcapotzalco
Av. San Pablo No. 180, Col. Reynosa Tamaulipas, México, D.F. 02200, México
Tel. 53189000, Ext. 1085, correo electrónico: fra@correo.azc.uam.mx

RESUMEN

Los seres humanos reconocemos a otros seres en parte a través del rostro, dar esta capacidad a un sistema informático permite un reconocimiento de acuerdo a la técnica utilizada en el procesamiento del mismo. “Utilizando las principales características de las redes neuronales como son: *memoria y clasificación de patrones*”[1], se puede dar al sistema un nivel de confiabilidad mayor, sin tener que recurrir a técnicas biométricas. Se utilizó una red neuronal *feed-forward*, con *backpropagation*, la cual constó de 3 capas, variando la cantidad de neuronas en cada una, observando su comportamiento y comparándolo.

Palabras clave: redes neuronales, backpropagation, feed-forward.

I. INTRODUCCIÓN

En el reconocimiento facial, influyen muchos factores importantes como:

- *Iluminación:* En condiciones controladas, la iluminación es en la medida de lo posible equitativa al tomar las imágenes de los rostros de cada persona, pero ésta en algún momento afectará de manera importante el reconocimiento o no de una persona [2].
- *Expresiones Faciales:* Los gestos como la risa, facciones de enojo, y sorpresa

proporcionan una variable más, estas expresiones ofrecen información conforme el estado de ánimo de la persona, es decir, no siempre el rostro de una persona presenta la misma fación.

- *Rotación del rostro.* Una variable a considerar ya que las imágenes de un rostro no siempre será idealmente de frente.
- *Accesorios.* Como lentes oscuros, gorras, pelucas, maquillaje, dificultan el reconocimiento de una persona.

Estas variables de entorno, en conjunto afectan de manera significativa el reconocimiento de un ser humano, ahora sí, trasladamos estas mismas, a un sistema informático, el grado de complejidad del mismo se va incrementando al tratar de cumplir con cada de una de las variables anteriores.

Es por esto que los sistemas de reconocimiento facial, tienen en la seguridad su meta principal:

- Autorizar el acceso a personas en áreas restringidas.
- Acceso a información dentro de una red particular, acceso a bancos, acceso a computadoras personales.
- Aplicación de leyes, identificación de delincuentes en lugares públicos.
- Reconocimiento en medicina forense, reconstrucción de rostros a través de restos humanos.

Estado del arte

En la actualidad han surgido diferentes técnicas de reconocimiento y se podrían clasificar de la siguiente manera [3]:

- *Características Geométricas*: El análisis de características como ángulos, distancias entre rasgos morfológicas del rostro permite el uso directo de las técnicas estadísticas para reconocimientos de patrones. En esta clasificación destaca el análisis biométrico.
- *Eigenfaces*: Turk y Pentland [4] presentaron un esquema de reconocimiento del rostro, en el cual las imágenes de la cara fueron proyectadas sobre los componentes principales del sistema del set original de fotos de entrenamiento.
- *Template Matching* (Emparejamiento de plantillas): En esta técnica se realizan correlaciones directas de los segmentos de la imagen. Es generalmente eficaz cuando las imágenes de prueba tienen la misma escala, orientación e iluminación que el entrenamiento realizado.

Abundando en la clasificación de características geométricas la biométrica es la más usada actualmente ya que “*permite la identificación de personas a través de características morfológicas, propias y únicas del individuo, conocidas como autenticadores* [5].

Dentro de la biométrica se consideran autenticadores las huellas dactilares, la geometría de la mano, la termografía facial, el iris, la retina, la voz. Esto permite ventajas y desventajas de acuerdo al identificador que se desea analizar. Con respecto a la *termografía facial* (identificar mediante el rostro humano), las ventajas que representa este sistema son: sistema no invasivo (no hay contacto con el

sistema de reconocimiento), permite la identificación de personas en movimiento, reconocimiento sin conocimiento de la persona. Sus desventajas radican en que el sistema de captura necesita una fuente de luz auxiliar, por lo cual es susceptible a problemas de iluminación. También es vulnerable al reconocimiento de sujetos sometidos a intervenciones quirúrgicas, y a la portación de accesorios por parte de la persona. Otra desventaja es que no son suficientemente rápidos en zonas con gran afluencia de gente, ante lo cual necesita mucha memoria y gran tiempo y velocidad de cómputo, dos características fundamentales para acortar el tiempo de ejecución.

Redes neuronales

Las redes neuronales no son más que otra forma de emular ciertas características propias de los humanos, como la *capacidad de memorizar* y de *asociar hechos*. Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro. Por ejemplo, son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan información irrelevante, etc. A grandes rasgos, estas aplicaciones son aquellas en las cuales se dispone de un registro de datos y nadie sabe exactamente la estructura y los parámetros que pudieran modelar el problema. En otras palabras, grandes cantidades de datos y mucha incertidumbre en cuanto a la manera de como estos son producidos.

Entre las ventajas de las redes neuronales podemos mencionar:

- *Aprendizaje Adaptativo*: capacidad de aprender a realizar tareas basadas en un entrenamiento o en una experiencia inicial.

- Auto-organización: una red neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.
- Tolerancia a fallos: la destrucción parcial de una red conduce a una degradación de su estructura; sin embargo, algunas capacidades de la red se pueden retener, incluso sufriendo un gran daño.
- Operación en tiempo real: los cálculos neuronales pueden ser realizados en paralelo; para esto se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.
- Fácil inserción dentro de la tecnología existente: se pueden obtener chips especializados para redes neuronales que mejoran su capacidad en ciertas tareas. Ello facilitará la integración modular en los sistemas existentes.

Backpropagation

El estudio de características, ventajas, desventajas, así como de dificultades encontradas en el desarrollo y entrenamiento de aplicaciones concretas, llevaron a la selección de una RNA del tipo backpropagation. Fue decisiva para esta elección, su capacidad de aprender a asociar patrones de entrada a la red (ejemplos) con las clases correspondientes propagando el error hacia atrás, es decir, su capacidad de autoadaptar los pesos de las neuronas de las capas intermedias para aprender la relación que existe entre un conjunto de patrones dados como ejemplo y sus salidas correspondientes. Esto posibilita que después del entrenamiento, se reconozca a nuevos vectores de entrada con ruido o incompletos, dando una salida activa si la nueva entrada es parecida a las presentadas durante el

aprendizaje. Esta característica importante, que se exige a los sistemas de aprendizaje [6], es la capacidad de generalización, entendida como la facilidad de dar salidas satisfactorias a entradas que el sistema no ha visto nunca en su fase de entrenamiento.

II. MÉTODO

Objetivos, restricciones y descripción

El siguiente trabajo está centrado sobre la idea de, un sistema que pueda reconocer rostros humanos, por medio de redes neuronales. Observar el comportamiento de la arquitectura sin tener una etapa de preprocesamiento de datos con técnicas como: análisis de componentes principales (PCA) [7,8] o Análisis de discriminante lineal (LDA).

Así mismo observar los resultados con un hardware no especializado en la toma de imágenes, sin una etapa de preprocesamiento de datos, es decir, procesar en su totalidad la información obtenida de las imágenes sin reducción de datos ni separación espacial.

En las siguientes líneas se describirá brevemente en que consistieron las pruebas, pero cabe adelantar que la iluminación con la cual fueron tomadas todas las fotografías fue controlada y similar, lo que permitió tener una variable menos en las posteriores pruebas.

El proyecto está dividido en tres partes fundamentales:

- *Etapas de Adquisición de datos:* En esta etapa se obtendrán las imágenes de los rostros humanos en formato .JPEG, y en escala de grises, se procesarán para obtener la entrada y salida deseada correspondientes requeridos para el entrenamiento y simulación de la red neuronal.

- **Etapa de Aprendizaje y entrenamiento:** En esta etapa la red procesará los datos presentados en la etapa de entrenamiento (entrada y salida deseada), los comparará con la salida obtenida y la salida deseada y por medio de la regla *delta*, ajustará los pesos de las conexiones hacia las capas anteriores a la salida hasta llegar a la capa de entrada, tratando de reducir con cada época el error entre datos de entrada y salida.
- **Etapa de Reconocimiento y visualización:** La última etapa del sistema concentrará la simulación de la red con cualquiera de los sujetos de prueba, visualizándose la imagen del sujeto de frente escogida como salida deseada. Se compararán los resultados obtenidos.

Para la realización se utilizó el paquete de software MATLAB® en su versión 7.0.0 (R14), ya que este software, provee de una variedad de funciones de manejo de imágenes, matrices, así como la creación y manipulación de redes neuronales útiles y esenciales para esta aplicación.



Figura 1: Diagrama a bloques del trabajo.

El hardware utilizado durante todo el desarrollo del proyecto fue el siguiente:

- Notebook XPS M140 Dell.
- Procesador Intel Centrino M a 1.73GHz.
- 1 GB de Memoria Ram DDR2.
- S.O. Windows Xp Profesional Media Center Edition.
- Cámara de Teléfono Móvil Benq modelo A500.

Etapa de Adquisición de datos.

Se utilizaron fotografías del rostro de 11 personas, 6 hombres y 5 mujeres, entre un rango de edad de 21 a 27 años de edad, de distintas facciones. Estas fueron tomadas con una cámara integrada en un teléfono celular (BENQ A500) en escala de grises y con un tamaño de 96*64 pixeles, cada una y almacenada en la computadora vía transferencia usb.

Los set's de imágenes, para entrenamiento y simulación de la red, fueron conformados de la siguiente manera: para el set de entrenamiento, se ocuparon por cada individuo 5 imágenes diferentes en cuanto a la rotación de rostro, a los accesorios y los gestos, así mismo, para la simulación se tuvo un set de 5 fotografías de cada persona, diferentes a las ocupadas en el entrenamiento de la red, y se eligió una de ellas para la simulación. La salida deseada, la cual se mostró a la red, fue una sola fotografía de cada persona, mirando de frente con su postura cotidiana, con lo cual se ven las diferentes facciones de cada una de ellas.

Como se observa en la Figura 2 y 3, las fotografías del rostro de cada persona fueron tomadas en diferentes posiciones, con diferentes gestos y expresiones marcadas entre sí, además de tener en algunos casos accesorios como gorra y anteojos.

Otro características de las imágenes es que la rotación del rostro para cada sujeto no es

constante, es decir, la rotación va desde los 5° hasta el caso máximo de los 85° (ver sujeto 3, fotografía 5 de la Figura 2), llegando a tener una imagen del perfil de la persona, esto sin duda da como resultado, un proceso de reconocimiento complejo y muy difícil de realizar. Los gestos y expresiones fáciles son diferentes para cada sujeto.

A continuación se muestran las imágenes utilizadas como entrada para la etapa de entrenamiento de la red neuronal, se nombró a cada foto de manera progresiva de 1 a 5, colocadas en carpetas para cada sujeto nombradas s1 a s11.

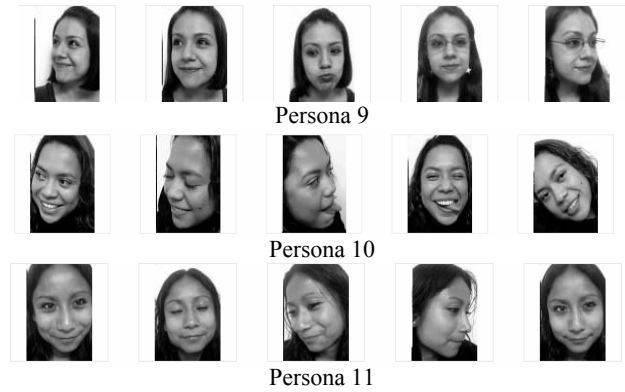
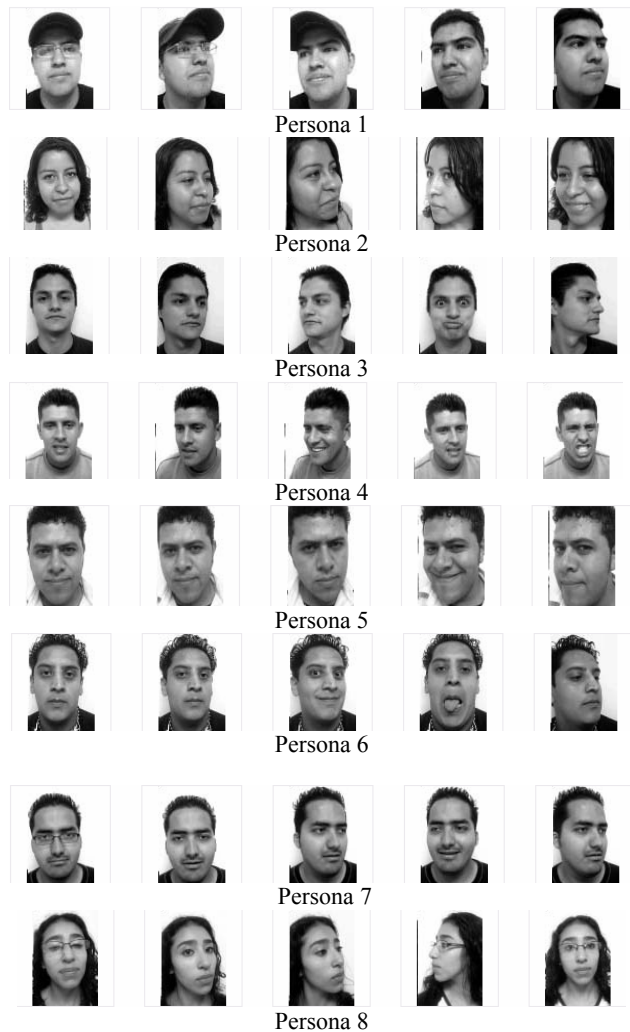
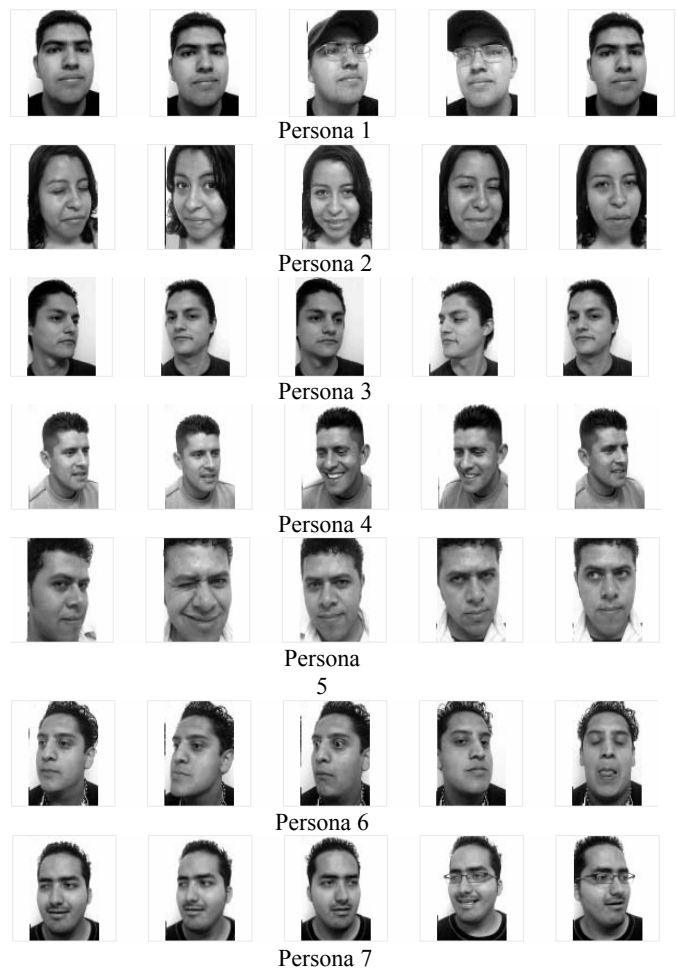


Figura 2. Base de fotografías para entrenamiento.

En la Figura 3 se muestran las imágenes utilizadas para la etapa de simulación, de cada uno de los sujetos.



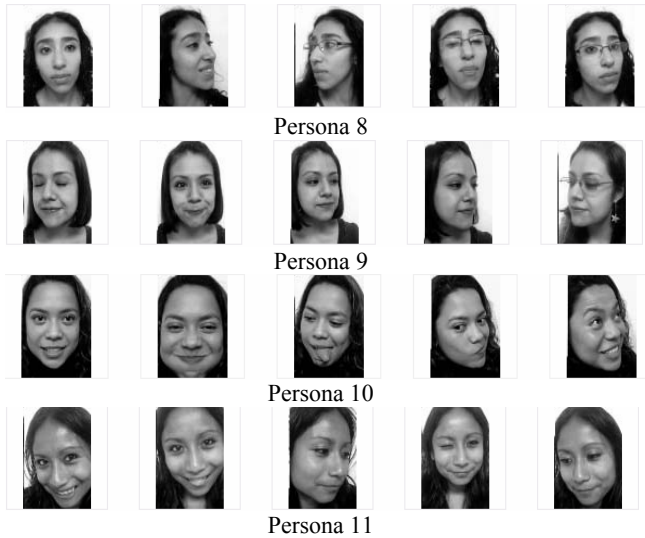


Figura 3. Set de fotografías para simulación.

En la Figura 4 se muestran las fotos de salidas deseadas para cada persona, se observa que se buscó que esta salida deseada, mostrará el rostro de manera frontal, con el gesto particular de cada persona.

Etapa de aprendizaje y entrenamiento

Como se mencionó el tamaño de cada foto es de 96x64 pixeles, y el tipo de datos es 'int8', los cuales pueden tomar 255 valores mostrados en hexadecimal de 00 a FF, cada uno de los valores representa un nivel de gris (escala de grises). Para poder presentárselas de mejor manera a la red neuronal se convirtieron los datos de 'int8' a 'double' para poder trabajar en un rango de 0 a 1. Y poder utilizar la función de activación sigmoïdal en cada capa de la red.



Figura 4. Set de fotos de salida ideal

Se formaron dos matrices de datos, la matriz de entrada con datos de 55 fotografías (figura 2) con una dimensión de 6144*55 y una matriz de salida con 11 fotografías (figura 4). Con estas matrices se entrenó a la red neuronal. Es importante recordar que no se realizó ningún tipo de preprocesamiento para disminuir la dimensionalidad de los datos ya que se tenía el interés de evaluar la capacidad de generalización de la red neuronal, por lo que, solo se consideró la normalización de los datos para poder trabajar con la red neuronal.

Etapa de reconocimiento y visualización

La simulación del sistema se llevó a cabo, con una sola fotografía de la persona a reconocer, seleccionada del set de fotos mostrado en la figura 3. La red procesa la información y nos da como resultado un vector columna con 6144 datos con el rango de 0 a 1, se convierte el tipo de datos 'double' a 'int8' para poder reconstruir la imagen deseada, se "forma" una nueva matriz de tamaño 96*64 con los datos de salida, obteniendo la fotografía reconstruida a partir de la salida de la red neuronal.

Resultados

Primer resultado. Al crear la red con 60 neuronas en la capa de entrada, 30 en la capa oculta y 6144 de salida y entrenarla con 5 individuos (individuos 1, 2, 3, 4 y 5) y simularla para 3 individuos (individuos 1,2 y 3, las fotos

se muestran en la Figura 3, línea 1 foto 3, línea 2 foto 2 y línea 3 foto 4).

Como se observa en la Figura 6, el sistema reconoció exitosamente las tres personas: para la persona 1, se utilizó para la simulación una fotografía de su rostro con accesorios (anteojos y gorra), además una rotación del rostro de 35° , la imagen reconstruida muestra pixeles fuera del rostro lo que indica que algunos de estos no fueron reconocidos por completo, haciendo Un reconocimiento, sino perfecto, bueno.

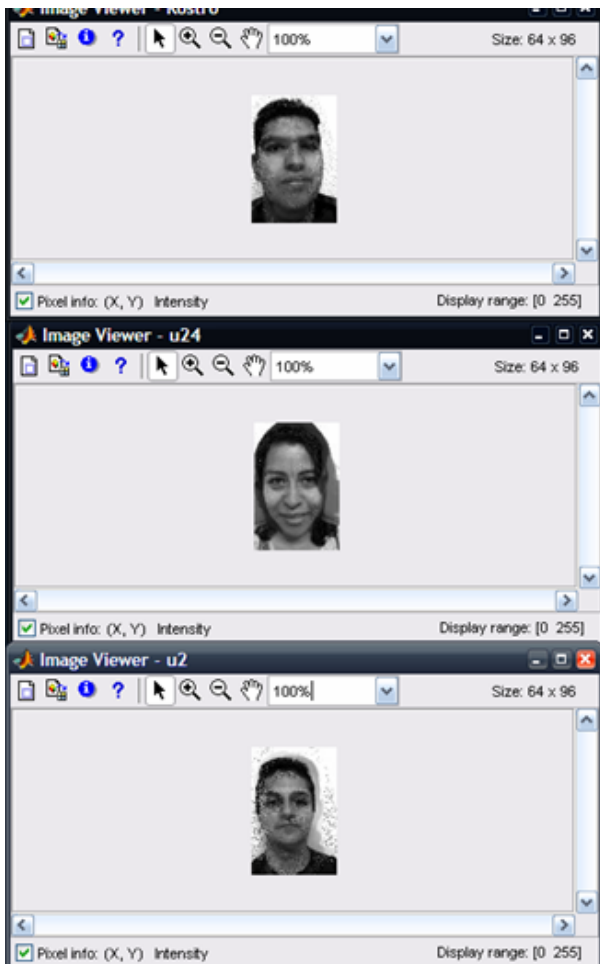


Figura 6. Salida de la red neuronal.

En el caso de la persona numero 3, la foto para simulación es de perfil cerca de los 70° de rotación. Observando la salida, el sistema reconoció a la persona, a pesar de la rotación del

rostro, pero sobresale una sombra del lado derecho de la imagen resultante y pixeles no reconocidos con éxito.

Con este resultado podemos adelantar que el sistema funciona para imágenes con rostros de perfil, pero necesita más capacidad para eliminar pixeles no aprendidos.

Segundo Resultado.

Para ésta prueba, se creó una red con 100 neuronas en la capa de entrada, 50 neuronas en la capa oculta y 6144 neuronas en la capa de salida. Se entrenó con las 11 personas, y se simuló para 4 personas (personas 1, 2, 3 y 9) los resultados fueron los siguientes.



Figura 7. Set de fotos para simulación.

En esta prueba se observa una mejor calidad en las fotos resultantes de los individuos 1 y 2 (Figura 8.1) pero en las personas 3 y 4 (Figura 8.2) el reconocimiento es incierto con pixeles borrosos y mal reconocidos.

Sin embargo con este resultado, se observa una tendencia de mejora: aumentar la cantidad de neuronas en la capa de entrada y capa oculta de la red neuronal para mejorar el comportamiento de la red.



Figura 8.1. Set de fotos salida de la red.



Figura 8.2. Set de fotos salida de la red.

Tercer Resultado.

Para una tercera prueba (en el trabajo en realidad se realizaron más de 11 pruebas, presentándose aquí las más importantes), se creó una red neuronal con 1500 neuronas de entrada, 500 neuronas de la capa oculta y 6144 neuronas en la capa de salida. Entrenada para las 11 personas y simulando para 6 personas (Figura 9).

Se observa en la Figura 10, una comparación de las salidas obtenidas de las diferentes simulaciones de la red neuronal, con las salidas deseadas mostradas a la red neuronal en la etapa de entrenamiento. Obteniendo buenos resultados, se logra identificar a todas las personas, excepto el caso de la persona10, a causa de los píxeles confusos.

Los píxeles confusos o que se muestran fuera del rostro de las personas, son aquellos que la red no pudo aprender de manera correcta y que por tanto presenta información mezclada.



Figura 9. Set de fotos para simulación



Figura 10. Imágenes de comparación. A la izquierda las imágenes reconstruidas a partir de la red neuronal. A la derecha las salidas deseadas presentadas a la red neuronal.

III. CONCLUSIONES

La red neuronal utilizada en la implementación del sistema para la identificación de rostros humanos funciona adecuadamente pero bajo ciertas limitaciones establecidas por el hardware; el poder de procesamiento (procesador moderno) y los recursos (memoria física y de paginación) de la computadora empleada son afectados principalmente por la cantidad de neuronas que contiene la red a consecuencia de la cantidad de datos que se manejan. Por lo tanto, es necesario realizar una etapa de preprocesamiento de los datos a través de técnicas estadísticas (análisis de componentes principales, análisis discriminante lineal, etc.) [4,7,10] que permita disminuir la dimensionalidad de los datos de entrenamiento y prueba con el fin de reducir el número de neuronas tanto de la capa de entrada como de la capa oculta de la red, demandando con esto menos recursos de hardware.

La red neuronal funcionó correctamente con un universo de personas (muestras) pequeño, al aumentar el número de personas (fotografías) tanto el ajuste de la arquitectura de la red neuronal como el ajuste de sus parámetros es más difícil de lograr, lo que pone de manifiesto la importancia de contar con una etapa de preprocesamiento de los datos del sistema.

Otro aspecto importante: el número de capas ocultas necesarias para el buen funcionamiento de la red neuronal, en los programas mostrados sólo se utilizó una capa oculta, esto también por la limitante de los recursos de la computadora, sin embargo como se comentó, la capa oculta o capas ocultas ayudan a la red neuronal para aprender las diferentes características de las diferentes personas. El caso máximo de individuos fue de 11 personas con una sola capa oculta, obteniendo buenos resultados, es decir, si se necesitarán más personas, se tendría que aumentar el número de capas ocultas en razón de individuos.

REFERENCIAS

- [1] Freeman, James A. “*Redes neuronales: algoritmos, aplicaciones y técnicas de programación*”. pag 10, 1993.
- [2] Montes Diez, Raquel. Conde, Cristina “*Automatic Detection of the Optimal Acceptance Threshlod in a Face Verification System.*”. Universida Rey Juan Carlos.España.
- [3] R. Chellappa, C. L. Wilson and S. Sirohey.
“*Human and Machine Recognition of Faces: A Survey*”.
Proceedings of the IEEE.
Vol 83. N° 5. pp 705-740. May 1995.
- [4] Rose Gómez, César Enrique.
“*Reconocimiento e identificación de rostros usando análisis de los principales componentes.*”. Instituto Tecnológico de Hermosillo.Mexico.
- [5] Espinosa Duró, Virginia.
“*Evaluación De Sistemas de Reconocimiento Biométrico*”.
Departamento de Electrónica. Escuela Universitaria Politécnica de Mataró (Barcelona).
- [6] J. Hilera and V. Martínez.
“*Redes neuronales artificiales. Fundamentos, modelos y aplicaciones*”.
Addison-Wesley Iberoamericana,
México, 1995.
- [7] P. S. Penev, L. Sirovich.
“*The global dimensionality of face space.*”
Proc. Fourth IEEE International, Conference on Face and Gesture Recognition. . P 264-270. 2000
- [8] J. Zhang, Y. Yan, and M. Lades.
“*Face recognition: eigenface, elastic matching and neural nets*”.
Proceedings of the IEEE, 85(9),
September 1997.

DISEÑO E IMPLEMENTACIÓN DE UN DETECTOR DE COMPLEJOS QRS EN LA LÍNEA DEL ECG

R. Campos-Rodríguez, M. Alcaraz-Mejía, S. Sahagún-Velasco

Departamento de Ciencias Tecnológicas, División de Desarrollo Bio-Tecnológico
 Centro Universitario de la Ciénega, Universidad de Guadalajara
 Av. Universidad 1115, Col. Linda Vista, C.P. 47810, Ocotlán, Jalisco, México
 Tel. +52 (392) 9259400, correo electrónico: {raul.campos, mildreth.alcaraz}@cuci.udg.mx

RESUMEN

Este artículo presenta un detector de Complejos-QRS diseñado para correr en sistemas embebidos. Por sus características, el detector es pequeño, ya que puede ser implementado en apenas 80 líneas de código ANSI-C. Adicionalmente, cuenta con propiedades de tiempo real al presentar un retardo de tan solo 2.7ms para detectar un Complejo-QRS cuando la señal electrocardiográfica se captura a 360 muestras por segundo. Los resultados en simulación muestran una exactitud del 97.98% en la detección positiva de Complejos-QRS, cuando se utilizan las bases de datos de MIT Arrhythmia Database en el sitio web de Physionet.

Palabras clave: Detector, Complejos-QRS, ECG, ANSI-C, Sistema Embebido

I. INTRODUCCIÓN

El corazón es el músculo encargado de bombear la sangre a todo el cuerpo mediante contracciones rítmicas y repetidas de aurículas y ventrículos [1]. Con un promedio de 70 latidos por minuto, el corazón de una persona de 66 años habrá latido más de 2,500 millones de veces [2]. Por otro lado, las enfermedades cardiacas representan una de las principales causas de muerte a nivel mundial [3]. El Electrocardiograma (ECG) es un procedimiento no invasivo que permite observar el estado general del corazón mediante el estudio de su actividad eléctrica [4], [5]. La actividad eléctrica del corazón se mide a través de un conjunto de electrodos colocados en la piel del paciente. La magnitud del vector resultante de esta actividad, de apenas unos cuantos milivoltios, es amplificada hasta alcanzar en promedio un voltio [4]. Posteriormente, pasa a un proceso de digitalización para su procesamiento

en computadora. Al graficar la señal amplificada en una base de tiempo adecuada, se obtiene lo que comúnmente se conoce como ECG [4], [5].

Mediante el estudio del ECG es posible extraer información simple como el ritmo cardiaco, que indica la frecuencia de las contracciones del músculo del miocardio. Además, en el espectro se encuentran datos acerca de la frecuencia respiratoria, el tamaño y ubicación de las cámaras del corazón, el estado general de la arterias y otros parámetros útiles para un diagnóstico cardiovascular [2], [4], [5].

El espectro de frecuencias de un ECG típico revela que la máxima energía se encuentra entre las bandas de 3Hz y 15Hz [2]. Ahí se localiza la información del Complejo-QRS y del segmento ST, que determinan, respectivamente, los periodos de las contracciones del miocardio y su posterior tiempo de relajación [4], [5]. Existen algoritmos para extraer información fuera de línea a partir de ECG's digitalizados. A su vez, comienzan a aparecer variantes para trabajar en línea y con restricciones de tiempo real.

En [7], por ejemplo, los autores proponen un algoritmo de detección de Complejos-QRS basado en el análisis del flanco de subida en una ventana de tiempo mediante el uso de valores de umbral. La señal electrocardiográfica es pre-procesada a fin de acentuar las diferencias entre los complejos y otras ondas en el ECG. En [2] y [6] los autores siguen un procedimiento similar considerando el uso de técnicas multi-resolución. En [8] los autores realizan una combinación de la técnica presentada en [7] con técnicas basadas en wavelets. En [9], los autores realizan una convolución de la señal electrocardiográfica con una función dinámica wavelet para detectar Complejos-QRS en ECG's con baja relación señal a ruido. Con un enfoque

similar, los autores en [10] utilizan una red neuronal y una transformación wavelet para detectar contracciones prematuras de ventrículo a partir de información recolectada por un monitor Holter. En otros trabajos, técnicas similares han sido extendidas para extraer información de la actividad cardiaca de los fetos [12], [11].

Este trabajo propone el diseño de un detector de Complejos-QRS para sistemas embebidos siguiendo el enfoque de ventanas de tiempo y flancos de subida presentado en [7]. Por sus características, este detector es pequeño, ya que puede ser implementado en 80 líneas de código ANSI-C. El retardo de este algoritmo es de tan solo 2.7ms, cuando la señal se digitaliza a 360 muestras por segundo, lo cual lo convierte en un excelente detector de Complejos-QRS de tiempo real. Su desempeño es muy bueno aún con resoluciones de conversión analógico/digital tan bajas como 11bits, como en el caso de la base de datos MIT Arrhythmia Database en el sitio web de Physionet [13]. Los resultados en simulación y fuera de línea para esta base de datos muestran una exactitud del 97.98% al detectar Complejos-QRS positivos.

El resto de este documento se organiza de la siguiente manera: la sección II. hace una revisión de la morfología del Complejo-QRS y su importancia en electrocardiografía. La sección III. explica el diseño del detector y muestra los puntos más importantes de su implementación. La sección IV. proporciona resultados experimentales. La sección V. ofrece una declaratoria de confidencialidad, y finalmente la sección VI. ofrece las conclusiones y trabajo futuro.

II. EL COMPLEJO-QRS

En la Figura 1 se muestra el esquema de un ECG típico medido desde el Lead-II. La Onda P está asociada a la contracción de las aurículas, mientras que el pico más prominente representa al Complejo-QRS que constituye la actividad eléctrica del músculo del miocardio. El punto más alto de Complejo-QRS se denomina punto R. El inverso del intervalo RR, es decir la duración entre picos R consecutivos, proporciona directamente la frecuencia cardiaca [4], [5].

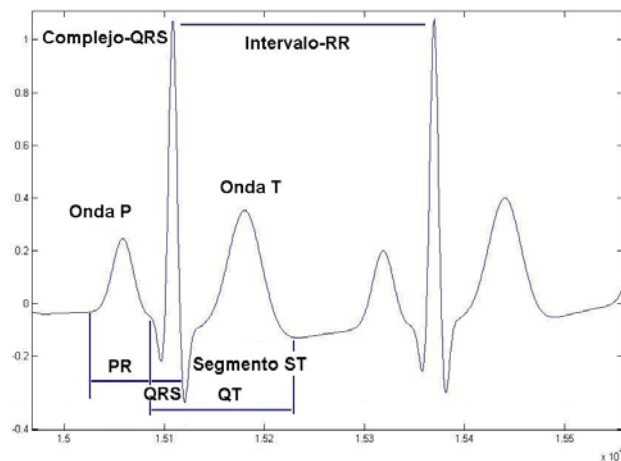


Figura 1: Esquema de un Complejo-QRS Típico

El Complejo-QRS representa tanto la contracción de los ventrículos como la relajación de las aurículas. Sin embargo, debido a que la masa muscular de los ventrículos supera en varios ordenes a la masa muscular de las aurículas, la relajación de estas últimas se ve oculta en el Complejo-QRS [5]. Debido a que el Complejo-QRS está directamente asociado a la contracción de los ventrículos, su identificación permite verificar el estado del músculo del miocardio, permite identificar contracciones prematuras del ventrículo (PVC) y otros factores de riesgo [2]. Adicionalmente, su localización dentro de la línea del ECG permite ubicar otros puntos y segmentos de gran importancia en electrocardiografía, como la Onda P, el segmento ST y la onda T [2].

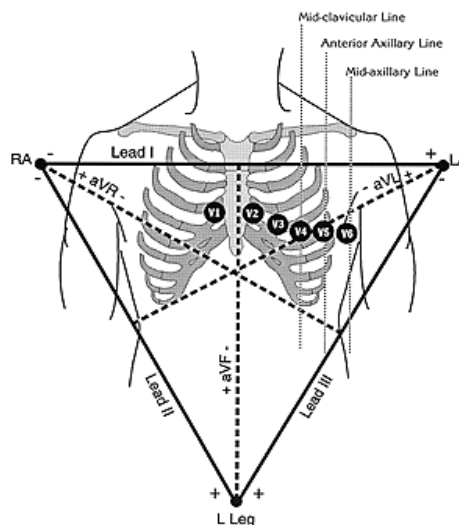


Figura 2: Distribución de los 12 Electrodo Estándar

El espectro de frecuencias del ECG revela que la mayor concentración de energía se encuentra entre las bandas de 3Hz a 20Hz, como se muestra en la Figura 3. En esta región se localiza la información del Complejo-QRS [2].

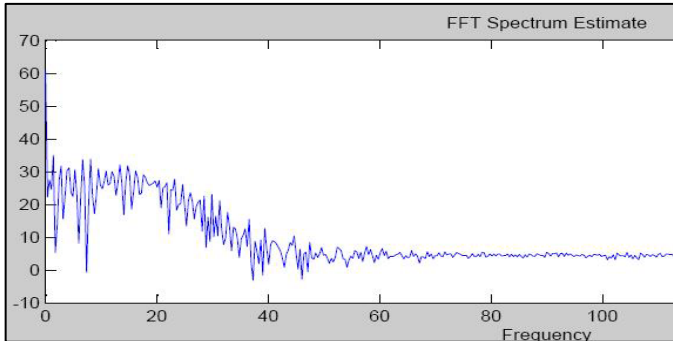


Figura 3: Espectro de Frecuencias de un ECG

III. DISEÑO DEL DETECTOR DE COMPLEJO-QRS

La Figura 4 muestra un diagrama a bloques del detector de Complejos-QRS propuesto.

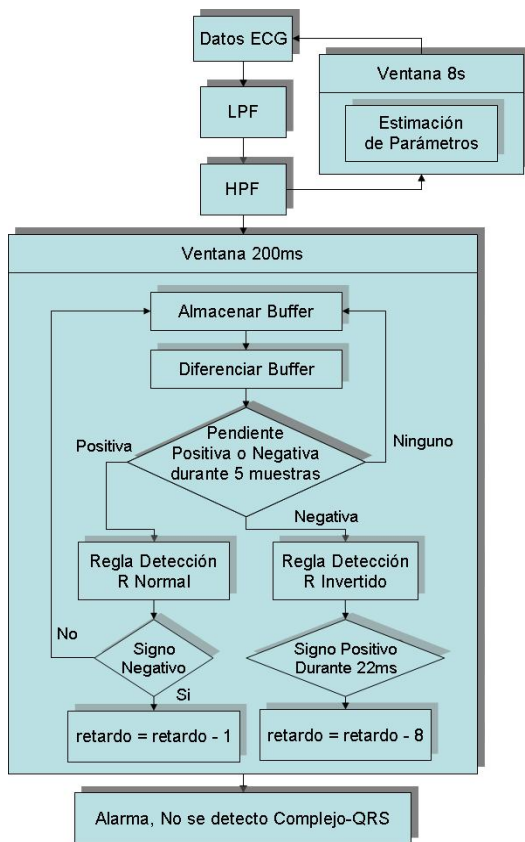


Figura 4: Diagrama Conceptual del Detector

Existe una ventana de 8s que permite al algoritmo estimar algunos parámetros como las pendientes de

subida y bajada más pronunciadas en la línea del ECG, que corresponde a los Complejos-QRS, así como su duración; el valor en milivoltios del punto R, el valor promedio de intervalos de puntos R consecutivos y otros. Esto ayuda a que el detector se “adapte” para detectar complejos en señales provenientes de distintos individuos. Un filtro pasa bandas formado por la combinación en cascada de un filtro pasa bajas (LPF) y un filtro pasa altas (HPF) mejora la calidad de la señal para su procesamiento digital.

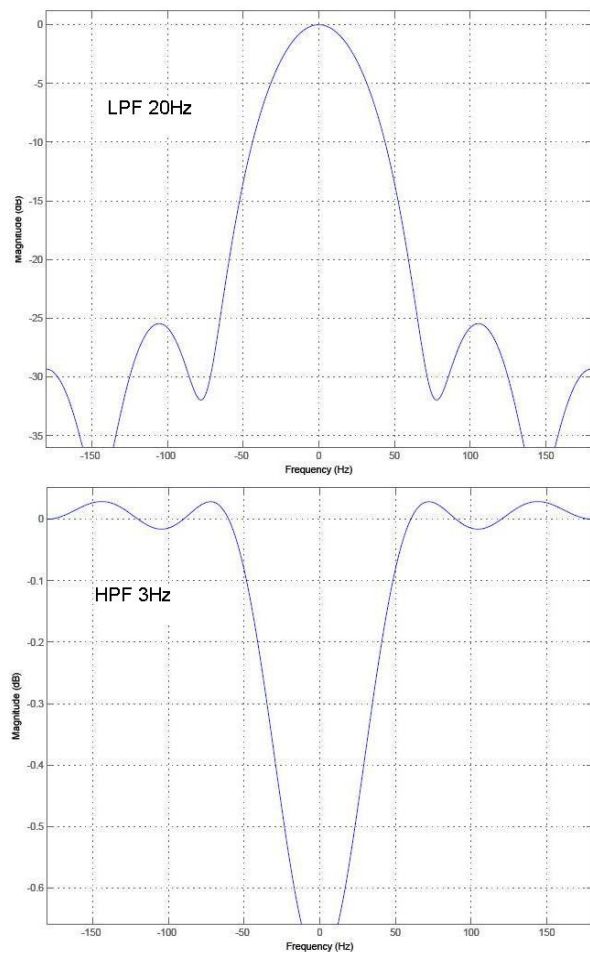


Figura 5: Kerneles de los filtros digitales de 3Hz y 20Hz

Usando la información en el espectro de frecuencias de la Figura 3 se diseñan los filtros digitales. Los kerneles utilizados, de 20Hz (LPF) y 3Hz (HPF), se muestran en la Figura 5. Mediante la herramienta FDATool y el Filter Design Toolbox ambos de Matlab, los kerneles se transforman en estructuras de datos tipo entero en ANSI-C para su utilización en el detector de

Complejos-QRS. Los filtros se prefieren en precisión entera (integer) a precisión doble (double) por dos razones principales. Primero, la aplicación del filtro sobre la señal mediante el método de respuesta finita al impulso (FIR) se ejecuta considerablemente más rápido con valores enteros que con valores en precisión doble. Segundo, los filtros con valores enteros se pueden utilizar en plataformas embebidas con aritmética de punto fijo. Los procesadores embebidos de punto fijo son generalmente más pequeños que los de punto flotante, consumen menos energía y son mucho más baratos.

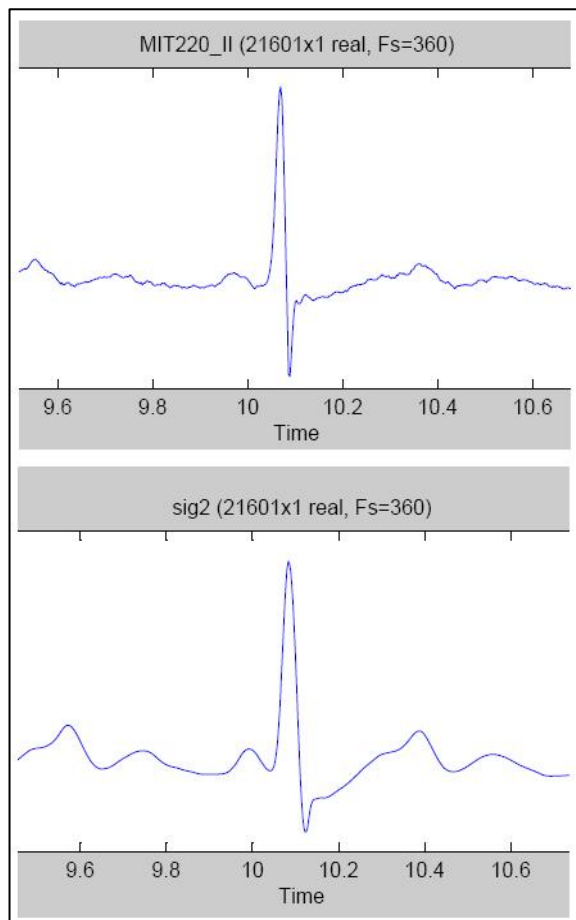


Figura 6: MIT220 Lead II antes y después del filtro

Mediante la convolución de estos filtros con la señal del ECG, los registros en las bases de datos del sitio web de Physionet se mejoran considerablemente. Por ejemplo, la Figura 6 muestra parte de las señales del Lead-II en el archivo MIT220 de la base de datos MIT Arrhythmia Database. La señal se suaviza al

desaparecer los rizos a frecuencias superiores a los 20Hz. Esto ayuda considerablemente al detector de Complejos-QRS, ya que se eliminan muchos cambios de signo en la derivada de la señal, lo que evita que el detector emita falsos positivos y se ahorra mucho tiempo de procesamiento.

Una vez filtrada la señal, el algoritmo entra en la fase de detección de Complejos-QRS. Una ventana de 200ms proporciona un marco de referencia sobre el que trabaja el detector. El número de ciclos en cada ventana de tiempo se calcula como:

$$ciclos = \lceil 0.2 fs \rceil \quad (1)$$

Mediante el ajuste de la constante 0.2 en la ecuación (1) se pueden incluir ventanas de tiempo más amplias, desde 0.2s hasta 1.0s. Sin embargo, los mejores resultados se obtienen utilizando un valor cercano o igual a 0.2s [14].

Las muestras filtradas se almacenan de manera consecutiva en un buffer tipo FIFO, para conservar la última porción de la señal. El tamaño del buffer esta determinada por:

$$sz = \left\lceil \frac{0.06 fs}{2} - 4 \right\rceil \quad (2)$$

La ecuación (2) se explica de la siguiente manera. Datos clínicos muestran que la duración mínima de un Complejo-QRS típico es de 60ms [5]. Al multiplicarlo por fs se obtiene el número de muestras promedio por complejo. La ecuación (2) considera la mitad de esta cantidad menos 4. El dato “-4” es un parámetro experimental que permite ajustar la derivada del flanco de subida del Complejo-QRS a una curva de segundo orden.

El buffer es diferenciado para obtener valores consecutivos de las pendientes en la línea del ECG. Entonces, se usan los valores encontrados en la etapa de “Estimación de Parámetros” para determinar si las pendientes corresponden a un flanco de subida o bajada. Una vez que se detecta un flanco de un Complejo-QRS, el algoritmo usa heurísticas para determinar un punto R normal o invertido. En el caso de un R normal, el detector espera el siguiente cambio de signo en la pendiente

para identificar el punto R. El intervalo del complejo se calcula como el intervalo actual menos uno en la ventana de 200ms. Entonces, la latencia al detectar el complejo se calcula como:

$$latencia = \frac{1}{f_s} \quad (3)$$

De la ecuación (3) se deduce que la latencia puede ser mejorada aumentando f_s . Para el caso de la base de datos MIT Arrhythmia Database, con $f_s=360$, la latencia es de tan solo 2.7ms.

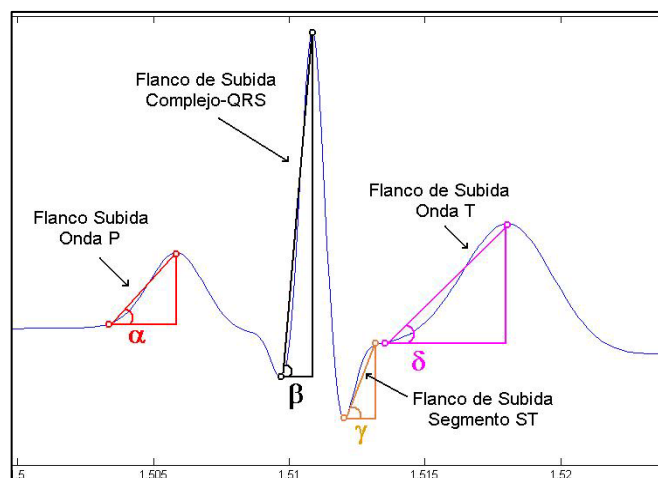


Figura 7: Pendientes en los flancos de subida

La heurística de detección de puntos R considera la pendiente en la línea del ECG. La Figura 7 muestra los flancos de subida de la Onda P, el Complejo-QRS, el Segmento ST y la Onda T. De todos estos, el Complejo-QRS se caracteriza por un considerable aumento en la línea de voltaje en un periodo corto de tiempo. Esto hace que el ángulo β se distinga claramente de los ángulos α , γ y δ . Lo anterior es aprovechado por el algoritmo para determinar el flanco de subida típico de un Complejo-QRS. Una vez que el detector ha encontrado un flanco de subida típico, espera el siguiente cambio de signo en la derivada para determinar que un punto R se ha alcanzado.

Los parámetros de detección se ajustan con cada complejo encontrado, a fin de que el algoritmo consiga mayor adaptación a la morfología del ECG de cada individuo. Sin embargo, este ajuste se trata con gran cuidado, ya que el algoritmo puede adaptarse para detectar Complejos-QRS anormales [2], [7]. Lo anterior debe complementarse con

información clínica para determinar los límites entre un complejo normal y uno que podría representar un problema cardiovascular [4].

El análisis de un punto R invertido es similar al procedimiento anterior, solo que ahora se usan los flancos de bajada en lugar de los de subida.

IV. RESULTADOS EXPERIMENTALES

La familia TMS320 de procesadores digitales de señales de Texas Instruments, proporcionan el más alto rendimiento en su tipo, lo que lo convierte en una plataforma ideal para procesamiento embebido. La arquitectura con un solo núcleo llega a velocidades de 1.2GHz y pueden desarrollar más de 900MIPS. La memoria interna van desde 32KB de cache L1/SRAM y 2048KB de cache L2/SRAM [15].

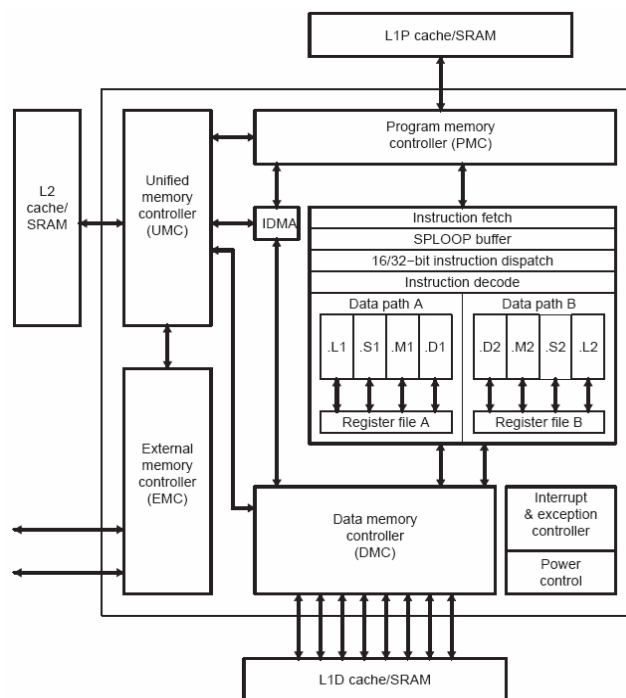


Figura 8: Arquitectura de CPU C60x

A fin de verificar el correcto funcionamiento del algoritmo propuesto, se utilizó la plataforma C60x mostrada en la Figura 8. El procesador proporciona hasta ocho ciclos de instrucciones de 32-bits y dos niveles de cache L1 y L2, con hasta 9600 MMAC de 16 bits. Adicionalmente, la arquitectura está complementada con un amplio conjunto de periféricos, como puertos serie McBSP,

controlador Ethernet 10/100/1000 Mb/s y un puerto I2C, entre otros [16].

Desde el punto de vista del programador, el procesador C60x puede ser controlado desde ANSI-C mediante el IDE CCStudio [17]. El IDE de programación se integra con el sistema operativo de tiempo real DSP/BIOS propietario de Texas Instruments. La plataforma integrada proporciona un conjunto de manejadores de periféricos, un planificador de tareas de tiempo real y manejo automático de memoria. Adicionalmente, el DSP/BIOS proporciona el concepto de semáforos, buzones, tareas prioritarias y manejo de interrupciones de hardware y software [18]. El enlace RTDX de comunicación de tiempo real con la PC permite agregar elementos de instrumentación para verificar “en sitio” los programas implementados en la plataforma. La comunicación mediante el enlace RTDX, entre la plataforma embebida y el IDE CCStudio en la PC, se efectúa durante el ciclo “IDLE” de más baja prioridad del sistema operativo DSP/BIOS para no afectar el desempeño de los algoritmos implementados [19].

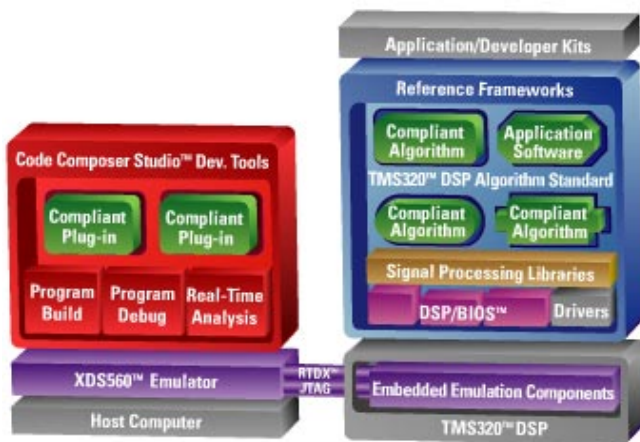


Figura 9: Plataforma DSP/BIOS

El IDE CCStudio de la Figura 9 se puede utilizar en modo simulador donde la plataforma de la Figura 8 se emula dentro de la PC donde se ejecuta el IDE, lo que permite una rápida verificación del código que implementa el detector de complejos.

Una sección del código en ANSI-C se muestra en el Algoritmo 1.

Algoritmo 1: Implementación ANSI-C

```
#include <cs1.h>
#include <math.h>
#include <tms320.h>
#include <dsplib.h>

...

/*LPF coeffs*/
#include "win_blk_fc_5.h"
/*HPF coeffs*/
#include "win_blk_fc_05.h"

...

main() {

...

while (TRUE) {

    x=c;

    //LPF
    fir((DATA*)&x, (DATA*)&B[0], (DATA*)&x,
&dbfr[0], 1, BL);
    //HPF
    fir((DATA*)&x, (DATA *)&BH[0], (DATA*)&x,
&dbfrH[0], 1, BLH);

    delay = DetectorQRS(x);

}

...

}
```

Se utiliza la versión optimizada de un filtro de respuesta finita al impulso implementado mediante la función `fir()` de la librería de procesamiento de señales `dsplib` [20]. Se utilizan los filtros de la Figura 5 implementados en los arreglos enteros `B` y `BH` para formar un filtro paso-banda entre 3Hz y 20Hz. Una vez filtrada la señal, se pasa a la función `DetectorQRS()`, la cual devuelve el retraso entre el Complejo-QRS actual y el inmediato anterior. Los detalles técnicos de la función `DetectorQRS()` no se muestran debido a un proceso de propiedad intelectual derivado de este trabajo.

Se utilizaron las capacidades de automatización que CCStudio ofrece para Matlab y Simulink a fin de desarrollar un marco de trabajo para el algoritmo. De esta manera, en Matlab y Simulink se diseñaron y verificaron los kernels de la Figura 5. Enseguida, usando el enlace de automatización, se verificó su correcto funcionamiento en un entorno de tiempo real creado en CCStudio.

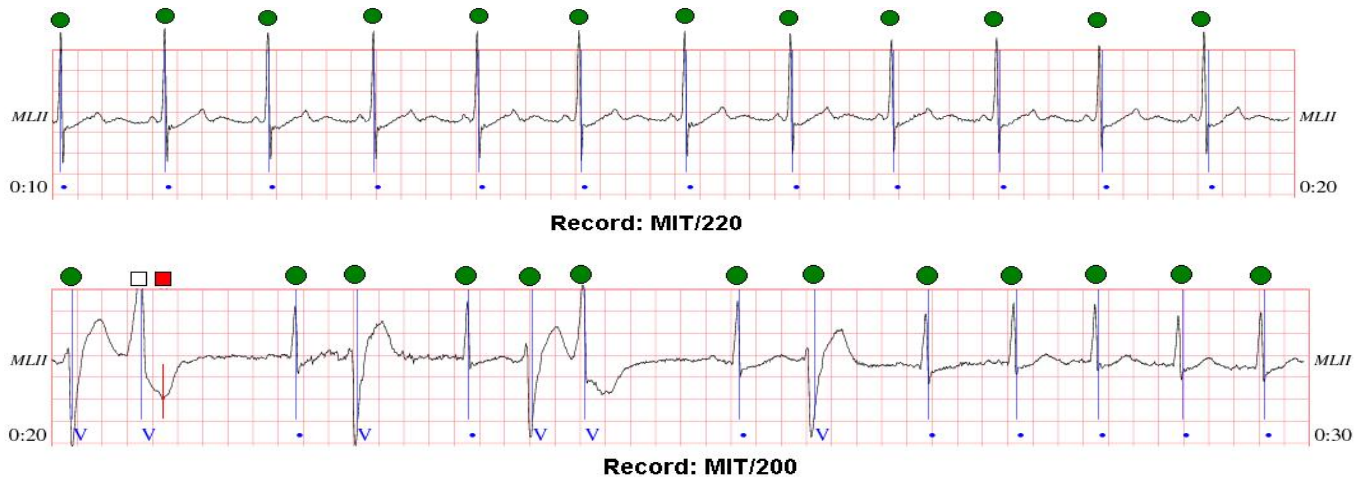


Figura 10: Resultados Experimentales sobre las Señales de la Base de Datos MIT Arrhythmia Database

Se utilizaron las bases de datos MIT Arrhythmia Database del sitio web Physionet.org. Este sitio ofrece una gran cantidad de señales biológicas digitalizadas en 9, 11 y 14 bits de precisión, a diferentes frecuencias de muestreo. Para la primera etapa de experimentación, se seleccionaron aleatoriamente registros digitalizados a 11bits de resolución con una velocidad de conversión de 360 muestras por segundo. Estas señales pertenecen a diferentes pacientes con diversos trastornos cardiovasculares documentados en el sitio web de Physionet [13]. Las señales fueron “amplificadas” en software y se les añadió una constante a fin de que pudieran ser tratadas como valores enteros dentro de la plataforma embebida. Para los experimentos, sólo se utilizó la señal capturada mediante el Lead-II. Los resultados para diez pacientes se muestran en la Tabla 1.

Tabla 1: Señales MIT-BIH Arrhythmia Database

Señal	%Aciertos	%Errores
100	100.00	0.00
101	100.00	0.00
103	100.00	0.00
119	98.46	1.54
124	100.00	0.00
200	96.55	3.45
220	100.00	0.00
221	100.00	0.00
222	100.00	0.00
232	84.79	15.21
Total	97.98%	2.02%

Para introducir estas señales al sistema, se crearon varios puntos de interrupción en el entorno de tiempo real desarrollado en CCStudio. La línea de código “x=c” en el Algoritmo 1 se utiliza para este fin.

Fragmentos de los resultados para los registros MIT/220 y MIT/200 se muestran en la Figura 10. Para algunos registros con señales relativamente normales, como el MIT/220, el algoritmo detectó el 100% de los Complejos-QRS con retardos típicos de 2.7ms. Sin embargo, la mayoría de las señales en la base de datos de Physionet presenta anomalías. Aun así, el algoritmo muestra buenos resultados, como en el caso del registro MIT/200, donde a pesar de las irregularidades en la señal, solo tuvo dos errores y cincuenta y seis aciertos, para una precisión del 96.55%. Uno de los errores presentado por el algoritmo se muestra con recuadros en blanco y rojo en la Figura 10. En este caso el Complejo-QRS es el marcado por el recuadro en blanco. Sin embargo el algoritmo lo detectó como un punto R invertido. Esto se debe a una de las Reglas de Detección de Puntos R Invertidos, en el diagrama de la Figura 4. Esta regla establece que si la señal baja a un 200% del valor del punto máximo anterior, este debe ser considerado un R invertido. El error se puede solucionar al agregar nuevas heurísticas basadas en datos clínicos que permiten capturar Complejos-QRS anormales. Sin embargo, el detector fue

diseñado pensando en un algoritmo pequeño y rápido, con posibilidades de correr en entornos de tiempo real como la plataforma TMS320, con restricciones en cuanto a memoria se refiere.

V. DECLARACIÓN DE CONFIDENCIALIDAD

Varios de los detalles técnicos más significativos de este trabajo se han omitido debido a un proceso de protección de la propiedad intelectual derivado de este trabajo.

VI. CONCLUSIONES

Este trabajo presenta un detector de Complejos-QRS en la línea del ECG. Por sus características el algoritmo es pequeño, implementado en tan solo 80 líneas de código en ANSI-C. Adicionalmente, cuenta con características de Tiempo Real ya que presenta un retardo de solo 2.7ms cuando la señal es digitalizada a 360 muestras por segundo. Estos atributos lo hacen ideal para su implementación en sistemas embebidos, como Holvers y otros sistemas de monitoreo cardiaco [14].

El detector presenta un excelente desempeño para identificar Complejos-QRS normales, con un retardo de $1/fs$, donde fs es la frecuencia de muestreo. Dicha latencia se puede mejorar incrementando la frecuencia de muestreo. Para el caso de complejos anormales, el detector tiene implementadas heurísticas para identificarlos. Es posible agregar al algoritmo más reglas de detección basadas en datos clínicos. Sin embargo, un aumento en las reglas de detección aumentaría considerablemente el tamaño del código final e impactaría directamente en el tiempo de respuesta, afectando su capacidad de dar respuesta en entornos de Tiempo Real.

El trabajo futuro sobre este trabajo consiste en utilizar la información de Complejo-QRS para identificar otras secciones en el ECG, como la onda P, el segmento ST y la onda T.

VII. REFERENCIAS

[1] Texas Heart Institute, <http://www.texasheartinstitute.org/hic/anatomy/anatomy2.cfm>.

- [2] Rajendra Acharya U, Jasjit S. Suri, Jos S.E. Spaan, S.M. Krishnan, "Advances in Cardiac Signal Processing", Ed. Springer Berlin Heidelberg, 2007, ISBN: 978-3-540-36674-4.
- [3] Organización Mundial de la Salud, <http://www.who.int/es/>.
- [4] Barbara Aehlert, "ECGs Made Easy", Pub: Mosby/JEMS; 3rd, 2005. ISBN: 0323039693.
- [5] ECG Learning Center, <http://library.med.utah.edu/kw/ecg/>.
- [6] Gari D. Clifford, Francisco Azuaje and Patrick McSharry, "Advanced Methods And Tools for ECG Data Analysis", Pub.: Artech House; 1st edition, 2006. ISBN: 1580539661.
- [7] Pan, Jiapu; Tompkins, Willis J., "A Real-Time QRS Detection Algorithm", IEEE Transactions on Biomedical Engineering, Volume: BME-32, Issue: 3, March 1985, Page(s): 230-236
- [8] Meyer, C.; Gavela, J.F.; Harris, M., "Combining Algorithms in Automatic Detection of QRS Complexes in ECG Signals", IEEE Transactions on Information Technology in Biomedicine, Volume 10, Issue 3, July 2006 Pages: 468 – 475
- [9] Kadambe, S.; Murray, R.; Boudreaux-Bartels, G.F., "Wavelet transform-based QRS complex detector", IEEE Transactions on Biomedical Engineering, Volume 46, Issue 7, Jul 1999 Page(s):838 – 848
- [10] Liang-Yu Shyu; Ying-Hsuan Wu; Hu, W., "Using wavelet transform and fuzzy neural network for VPC detection from the holter ECG", IEEE Transactions on Biomedical Engineering, Volume 51, Issue 7, July 2004 Pages: 1269 – 1273
- [11] Karvounis, E.C.; Tsipouras, M.G.; Fotiadis, D.I.; Naka, K.K., "An Automated Methodology for Fetal Heart Rate Extraction From the Abdominal Electrocardiogram", IEEE Transactions on Information Technology in Biomedicine, Volume 11, Issue 6, Nov. 2007 Pages:628 - 638

- [12] Kuei-Chiang Lai; Shynk, J.J., “A successive cancellation algorithm for fetal heart-rate estimation using an intrauterine ECG signal” IEEE Transactions on Biomedical Engineering, Volume 49, Issue 9, Sep 2002 Pages: 943 - 954
- [13] Chart-O-Matic: View signals and annotations. En: <http://physionet.org/cgi-bin/chart>.
- [14] “ACC/AHA Guidelines for Ambulatory Electrocardiography: Executive Summary and Recommendations”. ACC/AHA Practice Guidelines. (Circulation. 1999; 100:886-893.) © 1999 American Heart Association, Inc.
- [15] TMS320C6455, Procesador Digital de Señales de Punto Fijo Status, <http://focus.ti.com/docs/prod/folders/print/tms320c6455.html>
- [16] TMS320C6455 Fixed-Point Digital Signal Processor (Rev. H) (tms320c6455.pdf), 02 Oct 2007, <http://www.ti.com/lit/gpn/tms320c6455>
- [17] Code Composer Studio IDE <http://focus.ti.com/dsp/docs/dspsupportatn.jsp?sectionId=3&tabId=415&familyId=44&toolTypeId=30>
- [18] DSP/BIOS Real-Time Kernel, <http://focus.ti.com/docs/toolsw/folders/print/dspbios.html>
- [19] High Speed RTDX (HSRTDX), [http://tiexpressdsp.com/wiki/index.php?title=High_Speed_RTDX_\(HSRTDX\)](http://tiexpressdsp.com/wiki/index.php?title=High_Speed_RTDX_(HSRTDX))
- [20] C64x+DSPLIB, <http://focus.ti.com/docs/toolsw/folders/print/sprc265.html>

CARACTERIZACIÓN DE UNA RED NEURONAL DIFUSA EVOLUTIVA Y SU APLICACIÓN EN PROBLEMAS DE PREDICCIÓN

Andrés Ferreyra Ramírez, Marco Antonio Sánchez Trejo, Javier Alducin Castillo

Área de Instrumentación, Departamento de Electrónica, División de CBI
Universidad Autónoma Metropolitana--Azcapotzalco
Av. San Pablo No. 180, Col. Reynosa Tamaulipas, México, D.F. 02200, México
Tel. 53189000, correo electrónico: fra@correo.azc.uam.mx

RESUMEN

En el presente trabajo, se caracteriza la arquitectura de un modelo computacional basado en una red neuronal difusa evolutiva (Evolving Fuzzy Neural Network- EfuNN); esta es aplicada en la predicción de una serie de tiempos con el fin de analizar y evaluar su rendimiento. El objetivo es enfatizar las ventajas y desventajas tanto de la arquitectura como del algoritmo de entrenamiento en este tipo de aplicaciones.

Palabras clave: Sistemas Difusos, Redes Neuronales, Sistemas Neuro-Difusos, Fuzzy Neural Networks.

I. INTRODUCCIÓN

La adquisición del conocimiento es un asunto muy importante en muchos campos tales como control, sistemas expertos, reconocimiento de patrones, comunicaciones, etc. El propósito de la adquisición del conocimiento es extraer información acerca de la distribución de los datos a partir de datos observados [1]. Hasta el momento, muchos planteamientos, por ejemplo, redes neuronales [2] y sistemas difusos [3,4], han sido propuestos para resolver este problema. El método más popular en años recientes ha sido el modelado neuro-difuso [5]. El modelado neuro-difuso es uno de los métodos más eficientes para construir un sistema aproximado a partir de un conjunto de datos de entrada-salida (E/S). En general, esto implica dos pasos principales, la identificación de la estructura y el ajuste de los parámetros [6]. La meta del

primer paso es identificar la estructura del sistema difuso (SD) para obtener algunas reglas difusas a partir de los datos observados. Por lo tanto, el número de reglas, las funciones de membresía (FM) en cada regla, y el mecanismo de razonamiento difuso deben de ser establecidos en este primer paso. Para propósitos de mejorar la precisión, las redes neuronales adaptables son adoptadas en el segundo paso para ajustar los parámetros de las reglas difusas. La ventaja del modelado neuro-difuso es su adaptabilidad (capacidad de aprendizaje), rápida convergencia y su alta exactitud.

Los sistemas neuro-difusos son capaces de aprender y optimizar tanto reglas lingüísticas como funciones de membresía. Cuando se trabaja con reglas, el sistema neuro difuso crea un conjunto de reglas utilizando los datos de entrada-salida y las funciones de membresía iniciales, posteriormente estas reglas son evaluadas utilizando datos de entrenamiento con el fin de aprender nuevas reglas o eliminar algunas redundantes. Aprender y optimizar funciones de membresía es menos complejo que adaptar una base de reglas. Las funciones de membresía pueden describirse fácilmente por parámetros, que pueden ser optimizados con respecto a la medida del error global. La adaptación de parámetros es una tarea estándar para las RNs, y hay muchas combinaciones con SD para aprender funciones de membresía.

En este trabajo, se introduce un paradigma de modelado neuro-difuso que plantea la construcción

de su estructura en línea, ya que puede acomodar datos de entrada nuevos, incluyendo nuevas características, nuevas clases, etc. Durante la operación del sistema nuevas conexiones y nuevas neuronas son creadas (a través de un aprendizaje supervisado/no-supervisado). Este sistema neuro difuso es de varios órdenes de magnitud más rápido que las redes neuronales ordinarias y difusas entrenadas con el algoritmo de backpropagation [7]. Este paradigma neuro difuso pertenece a una nueva generación de sistemas inteligentes adaptables llamados sistemas conexionista evolutivos. De igual manera se pretende también ilustrar las características de la EFuNN en varios casos de estudio con conjuntos de datos para predicción de series de tiempos.

II. MÉTODO

II.1 Red neuronal difusa evolutiva [8, 9, 10].

Una red neuronal difusa evolutiva (Evolving Fuzzy Neural Network, EFuNN) consta básicamente de una red neuronal perceptrón multicapa de 5 capas de neuronas entrenada con un algoritmo de backpropagation modificado [Fig. 1]; en donde los nodos y las conexiones entre neuronas son creadas o unidas durante el proceso de evolución; es decir cuando se presentan y se adaptan datos al sistema. EFuNN cuenta con una capa extra de memoria opcional a corto plazo que permite realizar asociaciones temporales entre los datos de entrada del sistema, esta puede ser considerada como una capa de conexión de regeneración de reglas [Fig 2].

En EFuNN la primera capa o *capa de entrada* representa variables de entrada, es decir en esta se reciben los datos que llegan al sistema.

La segunda capa o *capa de entradas difusas*, realiza la cuantificación difusa de cada espacio de la variable de entrada, es decir, realiza un mapeo del mundo real al mundo difuso (fusificación). Las neuronas dentro de esta capa representan las FMs de las variables de entrada, las cuales pueden ser modificadas dinámicamente durante el proceso de aprendizaje de la red.

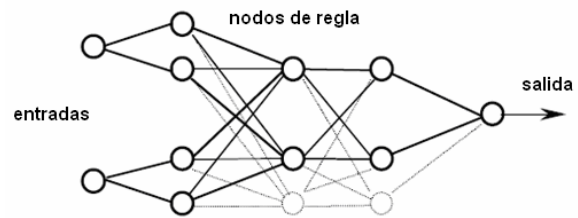


Fig.1.- La estructura básica de EFuNN.

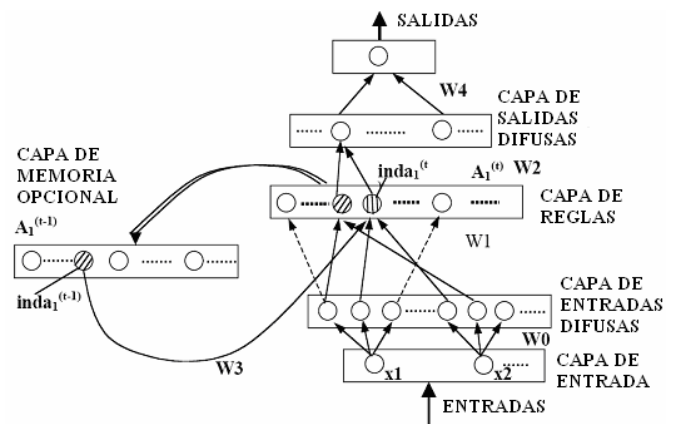


Fig.2.- EFuNN con una memoria de corto plazo y una conexión de regeneración.

En la tercera capa o *capa de reglas*, cada nodo representa una regla de inferencia difusa la cual constituye un prototipo de asociaciones de datos de entrada-salida difusos que se pueden representar como una asociación de hiper-esferas de espacios correspondientes a la entrada y salida. Cada regla de nodo que denotaremos como r es definida por dos vectores de pesos de conexión; $W1(r)$ para la entrada y $W2(r)$ para la salida, el peso de $W1(r)$ es ajustado por estudio supervisado basado en el error de salida, y el peso de $W2(r)$ es ajustado por el estudio no-supervisado basado en la semejanza que se mide dentro de un área local del espacio del problema. Las neuronas de esta capa utilizan funciones de activación lineal o gaussiana.

La cuarta capa de neuronas representa la cuantización difusa para las variables de salida. Aquí, una función de entrada cargada de la suma y una función lineal saturada de la activación se utilizan en las neuronas para calcular el grado de membresía para el vector de la salida que se asoció al vector actual de la entrada y que pertenecen a cada uno a la función de membresía respectiva.

La quinta capa representa los valores verdaderos para las variables de salida. Aquí se utiliza una función de activación lineal para calcular la defusificación para los valores de las variables de salida.

Cada nodo de regla, que esquematizaremos como r_j , y nos representa una asociación entre una hiper-esfera del espacio en la entrada difusa y una hiper-esfera del espacio de salida difuso (Fig. 4a), el $W1(r_j)$ representa los pesos de conexión de las coordenadas del centro de la esfera en el espacio de entrada difuso, y el $W2(r_j)$ las coordenadas en el espacio de salida difuso. El radio de una hiper-esfera de entrada de un nodo de regla es definido como $(1-Sthr)$, donde $Sthr$ es el parámetro de umbral de sensibilidad que define el mínimo de activación de un nodo de regla; por ejemplo, r_1 , antes desarrollado para representar un punto de datos $(Xd1, Yd1)$ a un vector (por ejemplo, $(Xd2, Yd2)$) para la entrada nueva del vector para ser asociado con este nodo de regla.

Dos pares de vectores de datos de entrada-salida difusos $d1 = (Xd1, Yd1)$ y $d2 = (Xd2, Yd2)$ son asignados al primer nodo de regla $r1$ si ellos caen en la esfera de entrada de $r1$ y en la esfera de salida $r1$; y si por supuesto la diferencia local normalizada difusa entre $Xd1$ y $Xd2$ es más pequeña que la r de radio $(1-Sthr)$ y la diferencia local normalizada difusa entre $Yd1$ y $Yd2$ es más pequeña que un umbral de error $Errthr$.

La diferencia local normalizada difusa entre dos vectores de membresía difusos $d1f$ y $d2f$ que representa los grados de membresía a que dos verdaderos valores de datos $d1$ y $d2$ pertenecen a una FM predefinida; es calculada como:

$$D(d1f, d2f) = \frac{\text{sum}(\text{abs}(d1f - d2f))}{\text{sum}(d1f + d2f)}$$

Por ejemplo, si $d1f = (0,0,1,0,0,0)$ y $d2f = (0,1,0,0,0,0)$, $D(d1, d2) = (1+1)/2=1$ que es el valor máximo para la diferencia local normalizada difusa. [Fig. 3]

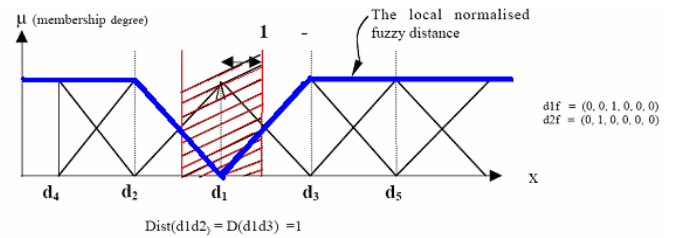


Fig.3.- Funciones de membresía (FM) y la distancia local normalizada difusa

Si el ejemplo de datos $d1 = (Xd1, Yd1)$, donde $Xd1$ y $Xd2$ son en proporción la entrada y la salida el grado de vectores de membresía difusos, y el ejemplo de datos es asociado con un nodo de regla $r1$ con un centro $r1^1$, que los nuevos datos señalan $d2 = (Xd2, Yd2)$, que esta dentro del área protegida del nodo de regla como se puede apreciar en la fig. 4a, será asociado con este nodo de regla también por que cae dentro del radio de estos nodos de regla antes de ser optimizado con los nuevos datos de entrada.

Por el proceso de asociar y de estudiar nuevos datos indica que en un nodo de regla, los centros de estas hiper-esferas de nodo se adaptan (optimizan) en el espacio de entrada difuso dependiendo el rango que aprende $Ir1$, y en el espacio de salida difuso dependiendo el rango que aprende $Ir2$, como lo muestran en la fig. 4a sobre los dos datos señala $d1$ y $d2$.

$Ir1$ e $Ir2$ identifican el rango o distancia que se van recorriendo los centros de las hiper-esferas al ir ajustando datos nuevos que se puedan agrupar dentro de ellas en el proceso de aprendizaje de un EFUNN.

El ajuste del centro $r1^1$ a su nueva posición $r1^2$ puede ser representado matemáticamente por el cambio de los pesos de conexión del nodo de regla $r1$ de $W1(r1^1)$ y $W2(r1^1)$ a $W1(r1^2)$ y $W2(r1^2)$ según las operaciones siguientes vector:

$$W2(r1^2) = W2(r1^1) + Ir2 \cdot Err(Yd1, Yd2) \cdot A1(r1^1)$$

$$W1(r1^2) = W1(r1^1) + Ir1 \cdot Ds(Xd1, Xd2)$$

Donde: $Err(Yd1, Yd2) = Ds(Yd1, Yd2) = Yd1 - Yd2$ es el valor firmado más bien que el valor absoluto de la diferencia del vector difuso;
 $A1(r1^1)$ = es la activación del nodo de regla $r1^1$ para la entrada vector Xd ; una función lineal simple se puede utilizar para $f1$; ejemplo:

$$A1(r1^1) = 1 - D(W1(r1^1), Xd)$$

Cuando un nuevo ejemplo es asociado a un rj del nodo de regla será localizado en el espacio de la entrada, su radio Rj , y su umbral $Sthr$ de la sensibilidad, cambian como sigue:

$$R_{j(t+1)} = R_{j(t)} + D(w1(R_{j(t+1)}), w1(R_{j(t)}))$$

Respectivamente:

$$Sthr_{(t+1)} = Sthr_{(t)} - D(w1(R_{j(t+1)}), w1(R_{j(t)}))$$

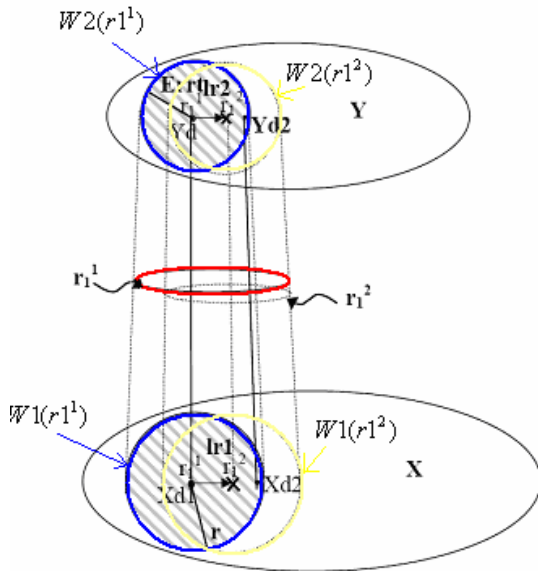


Fig.4a.- Mapeo y estudio de la Entrada / Salida

El proceso de aprendizaje en el espacio de entrada difuso es ilustrado en la Fig. 4b sobre cuatro datos señala $d1, d2, d3$ y $d4$. La Fig. 4c muestra como el centro del nodo de regla $r1$ se adapta después del estudio de cada nuevo punto de datos cuando el estudio de dos fases es aplicado. Si $lrn1=lrn2=0$, una vez establecido, los centros de los nodos de reglas no se mueve.

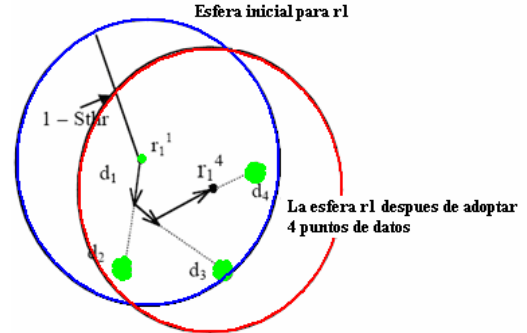


Fig.4b.- Proceso de aprendizaje

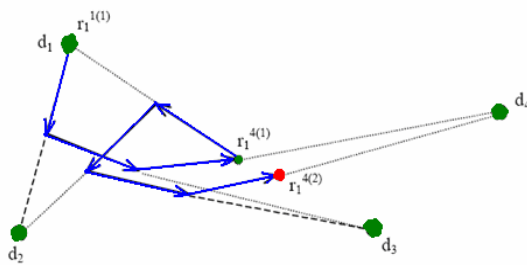


Fig.4c.- Adaptación del nodo de regla

La idea de creación dinámica de nuevos nodos de regla con el tiempo para los datos de serie de tiempo gráficamente es ilustrada en la fig. 4d.

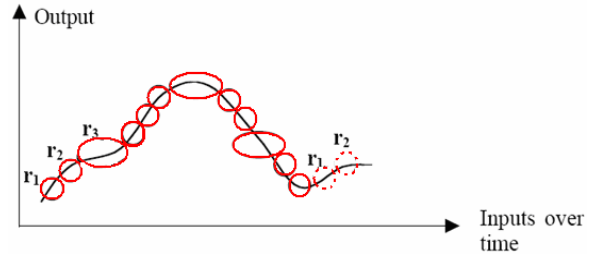


Fig.4d.- Creación Dinámica de nuevos nodos de regla con el tiempo

La capa temporal junto con el peso $W3$ de la conexión de la memoria opcional a corto plazo [Ver Fig.2] captura dependencias temporales entre los ejemplos consecutivos de los datos; es decir si se establece el nodo de la regla que ganaba en el momento $(T-1)$ (a cuál era asociado el vector de datos de entrada en el momento $(T-1)$) era $r1=inda1(t-1)$, y el nodo que gana en el momento que t es $r2=inda1(t)$, existe un acoplamiento y adaptación entre los dos nodos y se realiza de la siguiente forma:

$$W3(r1^{(t-1)}, r2^{(t)})^{new} = W3(r1^{(t-1)}, r2^{(t)})^{old}$$

$$+ Ir3.A1(r1)^{(t-1)}.A1(r2)^{(t)}$$

Donde: $A1(r)$ (t) denota la activación de un momento del nodo r de la regla a la vez (t); $Ir3$ define el grado a el cual el EFuNN asocia acoplamientos entre las reglas (racimos, prototipos) que incluyen ejemplos consecutivos de los datos (si $Ir3=0$, ningunas asociaciones temporales se aprenden en una estructura de EFuNN y el EFuNN de fig.2 se convierte en el que esta de fig.1).

En EFuNN se ajusta la posición y el radio de un r_j existente del nodo de regla para acomodar este ejemplo. Después de cierto rato (cuando cierto número de ejemplos se ha presentado) algunas neuronas y conexiones pueden ser podadas o ser agregadas. Diversas reglas de la poda pueden ser aplicadas para una poda acertada de nodos y de conexiones innecesarios.

Una de ellas es:

IF (Age (r_j) > OLD) AND (la activación total es TA (r_j) es menor que el tiempo del parámetro de poda Pr Age (r_j)). THEN poda el r_j del nodo de la regla.

Donde Age (r_j) se calcula como el número de ejemplos que se han presentado a EFuNN después de que el r_j hubiera sido creado, OLD es un límite de edad predefinido; Pr es el parámetro de poda en el rango [0, 1], y la activación total TA (r_j) se calcula como el número de ejemplos por los cuales el r_j ha sido un nodo que ganaba correcto, es decir el número de ejemplos que activaban este nodo.

La regla de la poda y la manera los valores para los parámetros de la poda se definen, dependen de la tarea del uso.

II.2 Inserción, extracción y agregación de la regla en EFuNN [8, 9, 10].

Inserción de la regla de EFuNN. La inserción de reglas difusas se alcanza con fijar un nuevo r_j del nodo de regla para cada nueva regla R, tal que los

pesos de la conexión $W1(r_j)$ y $W2(r_j)$ del nodo de regla representan la regla R.

Por ejemplo, la regla difusa (IF $x1$ es pequeño AND $x2$ es pequeño THEN y es pequeña) se puede insertar en una estructura de EFuNN fijando las conexiones de un nuevo nodo de la regla a los nodos difusos de la condición $x1$ - pequeñas y $x2$ -pequeña y al nodo difuso de la salida y -Pequeña a un valor de 1 por cada uno. El resto de las conexiones se fija a un valor de cero. De igual manera, una regla exacta se puede insertar en una estructura de EFuNN, por ejemplo; IF $x1$ es 3.4 AND $x2$ es 6.7 THEN y es 9.5, pero aquí los grados de membresía a los cuales los valores $x1=3.4$ y $x2=6.7$ de la entrada, y al valor $y=9.5$ de la salida pertenecen a los valores difusos correspondientes se calculan y se unen a los pesos correspondientes de la conexión.

Regla de extracción y de agregación. Cada nodo r , que de la regla nos representa un prototipo, la regla, ejemplar del espacio del problema, se puede describir por sus pesos de la conexión $W1(r)$ y $W2(r)$ y que nos definen la asociación de las dos hiperactivo-esferas correspondientes de la entrada difusa y de los espacios difusos del problema de la salida. La asociación se expresa como regla difusa, por ejemplo: IF $x1$ es 0.85 pequeño AND $x1$ es 0.15 medio AND $x2$ es 0.7 pequeño AND $x2$ es 0.3 media THEN y es 0.2 pequeño AND y es 0.8 grande.

Los números unidos a las etiquetas difusas denotan el grado (peso) a el cual los centros de la entrada y las hiper-esferas de la salida pertenecen a la función de membresía respectiva. El proceso para extraer reglas se pueden realizar agregando varios nodos de regla en hiper-esferas mas grandes como se puede apreciar en la figura 5a y 5b en un ejemplo de dos nodos de regla $r1$ y $r2$; para la agregación de dos nodos $r1$ y $r2$ de la regla, se utiliza la regla siguiente de la agregación:

IF (D ($W1(r1)$, $W1(r2)$) <= Thr1) AND (D ($W2(r1)$, $W2(r2)$) <= Thr2) THEN; se agrega $r1$ y $r2$ en r_{agg} y calcula los centros del nuevo nodo de la regla como:

$W1(r_{agg}) = \text{el promedio}(W1(r1), W1(r2)), W2(r_{agg}) = \text{el promedio}(W2(r1), W2(r2)).$

Por lo tanto aquí el centro geométrico entre dos puntos en un espacio difuso del problema se calcula con el uso de una operación media del vector sobre los dos vectores difusos. Esto se basa en una función por trozos lineal presumida entre dos puntos provenientes definidos a través de los parámetros S_{thr} y Err_{thr} entrados y de las hiper-esferas difusas de la salida.

Ejemplo: Las dos reglas siguientes $r1$ y $r2$ se pueden agregar para $Thr1=0.15$ y $Thr2=0.05$ en un nuevo r_{agg} de regla mientras que se demuestra abajo:

$r1$: IF $x1$ is Small 0.85 and $x1$ is Medium 0.15 and $x2$ is Small 0.7 and $x2$ is Medium 0.3 THEN y is Small 0.1 and y is Medium 0.9

$r2$: IF $x1$ is Small 0.80 and $x1$ is Medium 0.2 and $x2$ is Small 0.8 and $x2$ is Medium 0.2 THEN y is Small 0.12 and y is Medium 0.88

$$D(W1(r1), W1(r2)) = (0.05 + 0.05 + 0.1 + 0.1) / 2 / 2 = 0.075 < Thr1 = 0.15;$$

$$D(W2(r1), W2(r2)) = (0.02 + 0.02) / 2 / 1 = 0.005 < 0.02 < Thr2 = 0.05;$$

r_{agg} : IF $x1$ is Small 0.825 and $x1$ is Medium 0.175 and $x2$ is Small 0.75 and $x2$ is Medium 0.25 THEN y is Small 0.11 and y is Medium 0.89

Los nodos de la regla agregarán solamente si el radio del campo receptivo agregado del nodo r_{agg} es menos que un radio máximo predefinido R_{max} .

Agregación y abstracción por ECO-aprendizaje:
La agregación de los nodos de regla para que se pueda realizar la asociación de hiper-esferas más grandes de la entrada y del espacio de la salida se lograron el uso del método que aprende de ECO, cuando los pesos de la conexión $W1(1)$ y $W2(1)$ de un EFuNN1 desarrollado se utilizan como ejemplares difusos para desarrollar un EFuNN2

para valores más pequeños del umbral S_{thr} de la sensibilidad y del umbral Err_{thr} del error (Fig.6). Este proceso se puede continuar más lejos para desarrollar un EFuNN3 nuevo con un número más pequeño de los nodos de regla, por lo tanto un número más pequeño de reglas. En caso de que de las tareas de la aproximación de la función, la exactitud de la generalización en este caso pueda disminuir dependiendo de los umbrales elegidos $Thr1$ y $Thr2$ mientras que la agregación significa la creación de prototipos más grandes que acomoden más ejemplos que tienen vectores similares de la entrada y vectores similares de la salida.

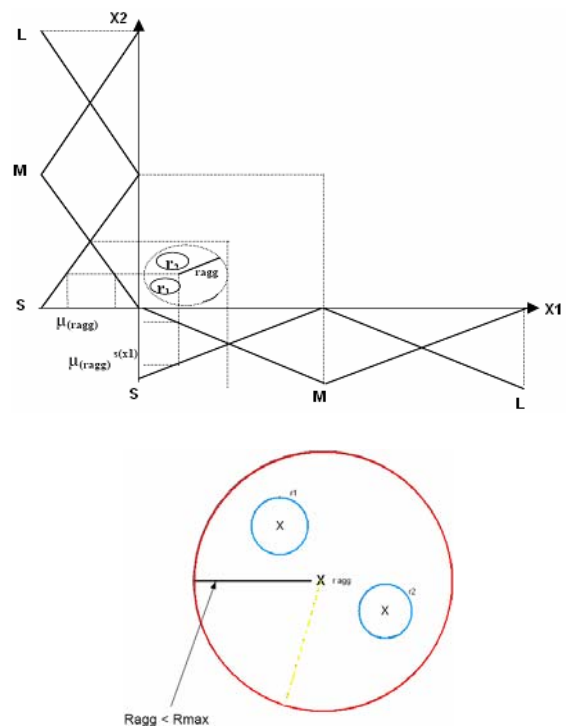


Fig.5a y Fig5b.- El proceso de la agregación de dos nodos de regla $r1$ y $r2$ en un nodo de racimo r_{agg} ;

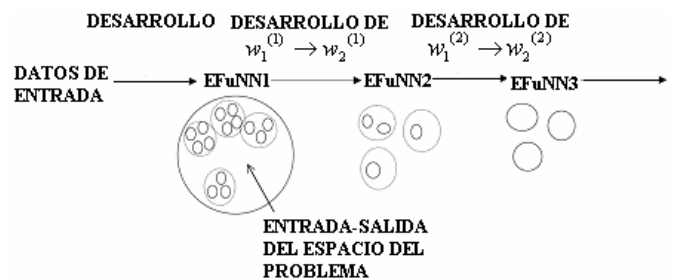


Fig.6. Agregar nodos de regla con aprender de ECO

*Cambia la función de membresía durante la operación. En EFuNN hay varias posibilidades para implementar cambios dinámicos en las funciones de membresía que son ilustradas gráficamente en fig.7a, b, c. Éstos son:

(a) Los ejemplares difusos almacenados en W1 y W2 que tengan tres FMs son defusificados (ejemplo, a través de la técnica de defusificación del centro de gravedad) y que utilizados desarrollan una nueva estructura de EFuNN que tenga, por ejemplo, cinco FMs (Fig.7a);

(b) La nueva FM se puede crear (insertar) sin una necesidad de cambiar las viejas (fig.7b). Para el cual cada centro del racimo (cada nodo de regla) pertenece a la nueva FM se puede calcular defusificando los centros igual que en el inciso (a);

(c) Cuando la agregación de los nodos de regla se aplica después de muchas épocas, es posible que la entrada o la salida de la función de membresía es difusa mientras que los centros de las hiper-esferas de regla se mueven, es decir los pesos de la conexión se mueven a sus nuevos valores verdaderos de la entrada como el caso en fig.7a y 7b (Fig.7c).

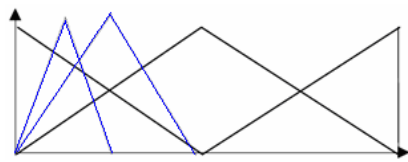


Fig.7a.- Se insertan nuevas FMs sin la modificación, si existen algunas.

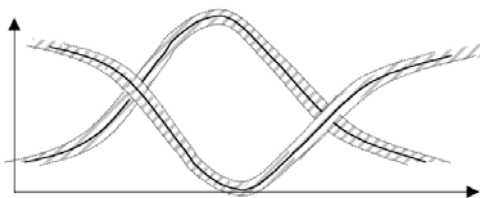


Fig.7b.- FMs difusas.

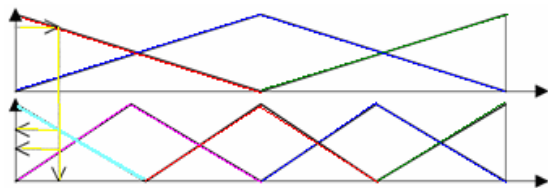


Fig.7c.- El número de las FMs aumenta a partir de 3 a 5

II.3 La serie de tiempo caótica Mackey-Glass

Una serie de tiempo es un conjunto de observaciones que expresan la dinámica de la variable de un sistema. Así una serie de tiempo S se puede definir como un conjunto de valores reales consecutivos. $S = \{X \in R / X_1, X_2, \dots, X_N\}$

Uno de los principales objetivos del estudio de las series de tiempo es la predicción; dicho problema puede ser representado separando la serie $S = \{X \in R / X_1, X_2, \dots, X_N\}$ en ventanas $w = (X_i, \dots, X_{i+p-1})$ de tamaño p, entonces hay que encontrar una función $f : R^k \rightarrow R$ tal que $f(X_i, \dots, X_{i+p-1}) = X_{i+p}$; esto es para cada $i \in \{0, N - p\}$.

La aplicación de EfuNN se realiza aquí para modelar y predecir los valores futuros de una serie de tiempo caótica que es la Mackey-Glass (MG); esta serie temporal se crea con el uso del tiempo-retraso diferencial con la ecuación MG que se puede definir de la siguiente manera:

$$\frac{dx}{dt} = \frac{0.2x(t - \tau)}{1 + x^{10}(t - \tau)} - 0.1x(t)$$

Esta serie se comporta como serie de tiempo caótica para algunos valores; es por eso que aquí $x(0) = 1.2$, $\tau = 17$ y $x(t) = 0$ para $t < 0$; por lo que $t > \tau$. Los datos de entrada-salida para desarrollar un EFuNN de los datos de la serie de tiempo de MG son presentados de la siguiente manera: vector de la entrada: $[x(t), x(t-6), (t-12) x(t-18)]$; vector de la salida $[x(t+6)]$. Aquí el objetivo es predecir los valores futuros $x(t+6)$ a partir de 4 puntos espaciados en seis intervalos de tiempo en el pasado.

Los datos son generados con un valor que se le asigna a $\tau=17$ con la ayuda de el método de Runge-Kutta de 2º orden con un intervalo de paso de 0.1 para las cuatro entradas que son $x(t)$, $x(t-6)$, $(t-12) x(t-18)$ y una salida que es la de $x(t+6)$.

Se utilizaran 988 datos para predecir el valor de la salida $x(t+6)$ a partir del vector de entrada.

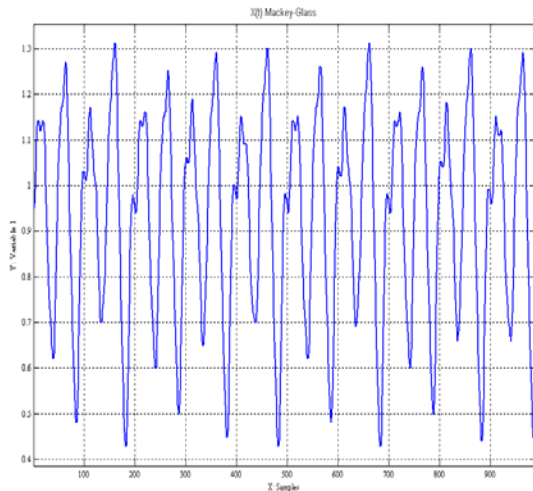


Fig 8.- Datos generados en dos dimensiones para el vector de entrada del primer punto $x(t)$.

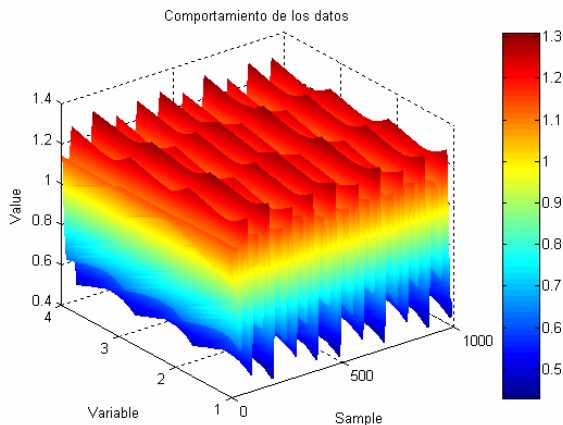


Fig 9. Comportamiento los datos empleados en el vector de entrada que tiene 4 puntos: $[x(t), x(t-6), x(t-12), x(t-18)]$.

II.4 Experimentos y resultados

Los parámetros de EfuNN que se ajustaron para los diferentes experimentos son:

1. Umbral de sensibilidad S_{thr} .
2. Umbral del error Err_{thr} .
3. Número de funciones de membresía (FMs).
4. Rango de aprendizaje (Ra) para w_1, Ir_1, w_2, Ir_2 .
5. Podamiento (Pod) y edad del nodo (En).

6. Radio máximo (R_{max}) para la agregación
7. Numero de nodos de regla activados (NNRA).
8. Conexión Recurrente (CR).

Las cuales se muestran en la tabla 1.

Parámetros de EfuNN									
	S_{thr}	Err_{thr}	FMs	Ra		Pod. (En)	Add	NNRA	CR
				w_1 Ir_1	w_2 Ir_2		R_{max}		
1	0.9	0.1	3	0.1	0.1	No	0.5	3	No
2	0.9	0.05	5	0.3	0.3	70	0.5	5	0.5
3	0.9	0.05	5	0.3	0.3	60	0.5	5	0.5
4	.98	0.01	5	.05	.05	No	0.5	5	0.5
5	0.9	0.01	7	0.1	0.1	60	0.5	2	1.0
6	0.9	.005	10	0.1	0.1	60	0.5	1	1.0
7	0.7	.001	10	.01	.01	40	1.0	1	1.0

Tabla 1. Parámetros de EfuNN utilizados en los diferentes experimentos.

Los parámetros de EfuNN que utilizaron para medir su rendimiento son:

1. Numero de nodos de regla generados (NRG).
2. Error RMSE.
3. Error NDEI.

Los cuales se muestran en la tabla 2.

Exp. Num	Resultados		
	NRG	Error	
		RMSE	NDEI
1	195	0.0511	0.2256
2	70	0.0752	0.3316
3	58	0.0782	0.3451
4	907	0.0458	0.2021
5	60	0.0475	0.2098
6	60	0.0334	0.1473
7	38	0.0334	0.1472

Tabla 2. Resultados de los experimentos.

Como se puede apreciar en la tabla 2, EfuNN reduce el número de nodos de reglas al aplicar la técnica de poda; pero a cambio de esto se aumenta el Error RMSE y el error NDEI. En la medida en que se varían los diferentes parámetros de la red se reduce considerablemente el número de nodos y se afecta mínimamente al error. El hecho de establecer parámetro que aumenten considerablemente el número de reglas no garantiza una reducción del error. Ver figuras 10-17.

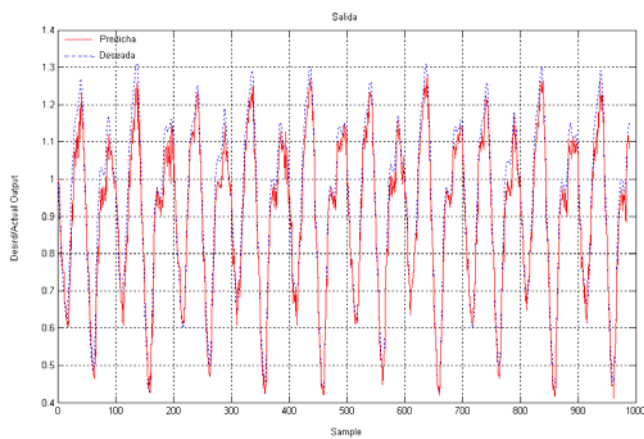


Fig 10. Salida deseada vs predicción de EfuNN para el experimento 1.

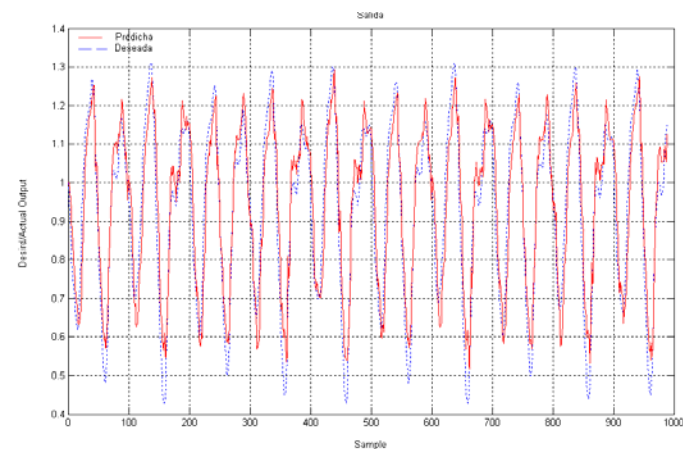


Fig 11. Salida deseada vs predicción de EfuNN para el experimento 2.

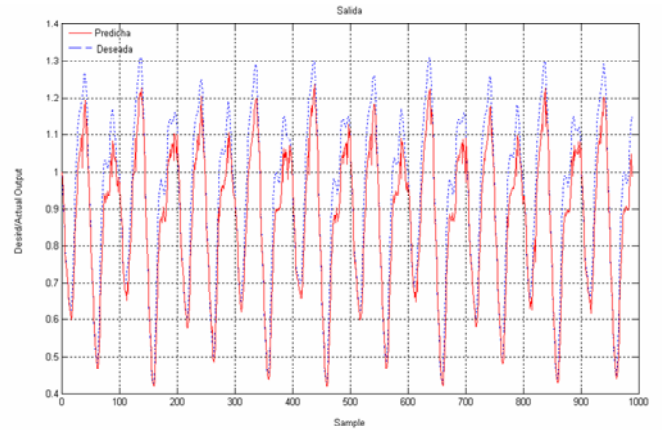


Fig 12. Salida deseada vs predicción de EfuNN para el experimento 3.

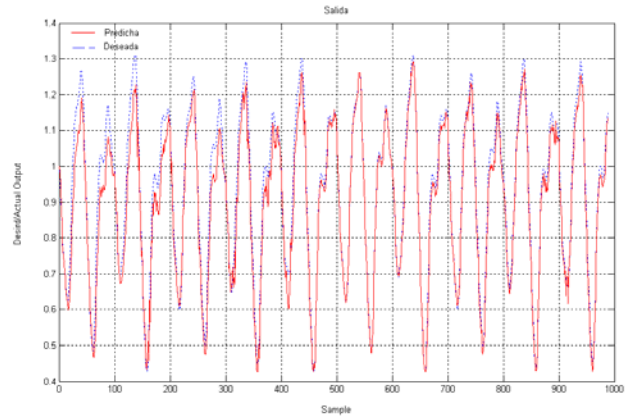


Fig 13. Salida deseada vs predicción de EfuNN para el experimento 4.

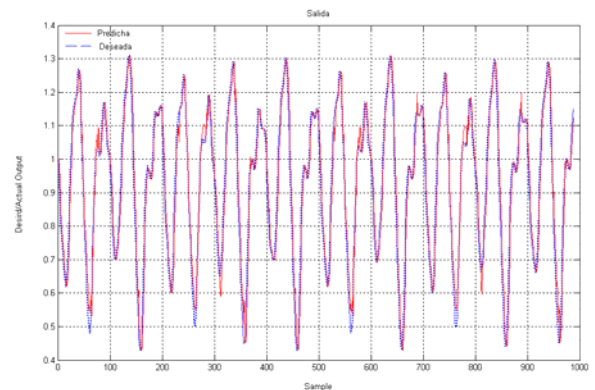


Fig 14. Salida deseada vs predicción de EfuNN para el experimento 5.

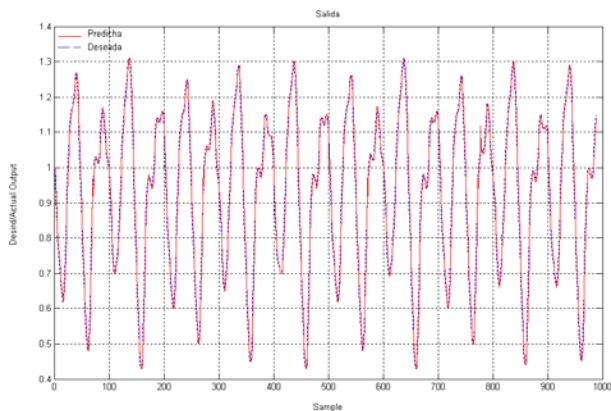


Fig 15. Salida deseada vs predicción de EfuNN para el experimento 6.

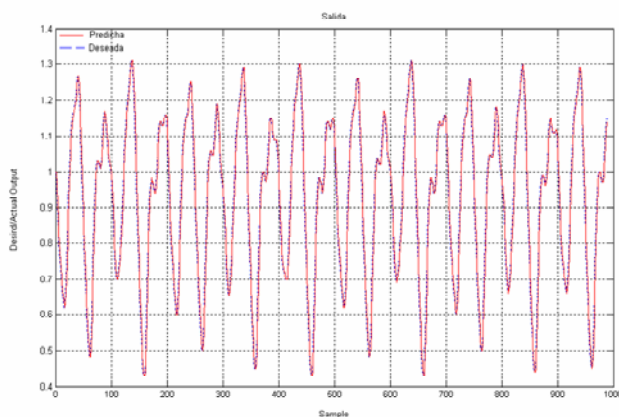


Fig 16. Salida deseada vs predicción de EfuNN para el experimento 7.

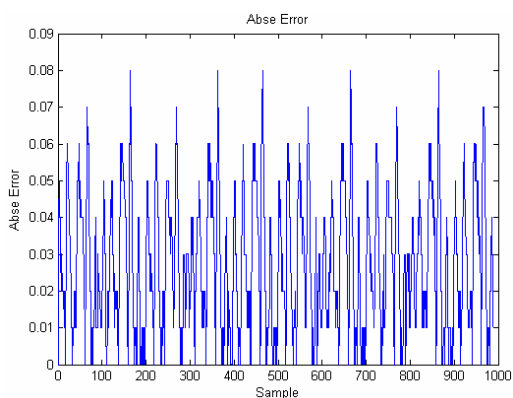


Fig 17. Error generado para el experimento 7.

III. CONCLUSIONES

La Red Neuronal Difusa Evolutiva (EFuNN) es desarrollada por un sistema conexionista evolutivo que es construido en línea permitiéndonos que el

desarrollo, la adaptación y el aprendizaje se den de una manera rápida con forme se van presentando al sistema los datos y de esta manera se optimizan los parámetros como son las neuronas y sus respectivas conexiones por lo que se puede considerar que EFuNN emplea un aprendizaje adaptable.

Algunas de las desventajas de EFuNN es que se deben de establecer los parámetros a prueba y error hasta encontrar los óptimos.

EFuNN puede ser empleada para modelar y extraer reglas difusas de sistemas cambiantes complejos; por lo que puede ser considerada como de alta precisión en predicción de series de tiempo además de presentar un entrenamiento rápido sobre otros sistemas que son empleados para la predicción.

Tomando en cuenta los experimentos anteriores se tiene que el valor que toma el umbral de error Err_{thr} se debe de minimizar lo mas posible para que la EFuNN nos genere valores de error a la salida mínimos; tomando en cuenta los rangos de aprendizaje tanto de los pesos W_1 y de W_2 para que a la hora de que son generados los nodos de regla estos tiendan a tomar un numero adecuado de datos tanto de la entrada como de la salida difusa, se debe de considerar alguna técnica de poda para que se tengan menos nodos de regla y por lo tanto menos reglas; teniendo en cuenta de que la edad de activación de un nodo este en un rango considerable de tal forma que no afecte el desarrollo y adaptación de la EFuNN para que la salida que nos proporciona no se vea afectada de la forma en que se podan nodos que tal ves eran necesarios o en su defecto que se tendrían que podar nodos que ya no eran activados; otro parámetro importante es la agregación de nodos de regla según lo explicado en la parte de agregación de nodos de regla para la EFuNN; es decir que si dos o mas nodos vecinos son altamente activados estos se pueden agregar en uno solo para reducir el numero de nodos de regla.

IV. REFERENCIAS

- [1] Chen-Sen and Shie-Jue Lee, A hybrid Algorithm for Structure Identification of Neuro-Fuzzy Modeling, IEE trans. Systems, Man, and Cybernetics, vol 5, pp. 3611-3616, 2000.
- [2] K. Hornik, M. Stinchcombe and H. While, "Multilayer feedforward networks are universal approximators", Neural Networks, vol 2, pp359-366.
- [3] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control", IEEE trans. Systems, Man and Cybernetics, SMC-15(1),pp. 116-132.
- [4] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling", IEEE Trans. Fuzzy Systems, vol. 1, no. 1, pp. 7-31, 1993
- [5] A. Ferreyra, L. Jiménez, "Diseño optimo de arquitecturas neuro-difusas para control inteligente", SOMI XIX Congreso Nacional de instrumentación, AFR19179,
- [6] Andrés Ferreyra, Wen Yu, "On-line fuzzy neural modeling with structure and parameters updating", IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, CIMSA 2004, Boston, MD, USA,14-16 July 2004.
- [7] N. Kasabov, Evolving Fuzzy Neural Networks - Algorithms, Applications and Biological Motivation
- [8] N. Kasabov, Evolving fuzzy neural networks for supervised/unsupervised, IEEE Trasc. Of systems, man and cibernetics, Part B, Vol. 31, No. 6, (2001), 902-918.
- [9] N. Kasabov, Evolving connectionist systems for on-line, knowledge-based learning: principles and applications, IEEE Systems, Man, and Cybernetics, (1999), 1-31.
- [10] N. Kasabov, Evolving Fuzzy Neural Networks for On-line Knowledge Discovery, (2001), 1-23.

PROCESAMIENTO DIGITAL DE VIDEO EN TIEMPO REAL Y VIDEO-WALL CON PC

Coronillas-Contreras Ukranio, Avilés-Cruz Carlos y Andrés Ferreyra-Ramírez

Departamento de Electrónica. Universidad Autónoma Metropolitana – Azcapotzalco.
Av. San Pablo No. 180 Col. Reynosa. C. P. 02200, México D. F.
caviles@correo.azc.uam.mx

RESUMEN

En el desarrollo de la presente trabajo se abordaron tres tipos de problemas, el primero fue establecer el hardware mínimo necesario para llevar a cabo el despliegue de una imagen de video en forma de videowall (arreglo cuadrado de monitores que presenta una sola imagen), haciendo uso de la PC y con cuatro monitores. El segundo problema consistió en hacer uso del sistema operativo para controlar las cuatro tarjetas de video, lo cual corresponde a un procesamiento de escalamiento en la imagen, y finalmente el problema fundamental de construir el software que permita el procesamiento de video en tiempo real, el cual se apoya en el sistema de desarrollo SDK de DirectX y en particular la herramienta DirectShow. Se elaboraron procesamientos básicos de video que consisten en filtrado de componentes de color, adición de ruido a la imagen, superposición de imagen al video, eliminación de color, posterizado y realce, entre otros.

1. INTRODUCCION

El procesamiento digital de señales y sus aplicaciones envuelven hoy día una gran cantidad de áreas como la medica, espacial, de telefonía, comercial, científica, industrial, entre otras. Sus orígenes se ubican en las décadas de los 60's y 70's (Watkinson, 1994; Luther, 1997), momento en el cual las computadoras digitales comenzaron a estar disponibles. Los desarrollos iniciales se dieron solo en áreas estratégicas como radar y sonar, exploración petrolera y espacial, además de imágenes médicas (Luther, 1997; Poynton, 1996). A partir de los 80's y 90's la revolución en las computadoras comenzó a expandir el campo del procesamiento digital de señales en nuevas aplicaciones, principalmente comerciales como telefonía móvil, reproductores de discos compactos, y video conferencia (Poynton, 1996).

Normalmente el procesamiento digital se lleva a cabo con dispositivos digitales de propósito específico denominados DSP (Digital Signal Processors) (Wallace, 1998). Aunque en los últimos años el desarrollo de los DSP ha permitido su incorporación en diversos dispositivos electrónicos, actualmente el procesamiento

de video a través de estos es caro debido a que el mercado no es muy extenso aún y su demanda es todavía reducida (Brey, 1995).

La ventaja del uso de microprocesadores de propósito general respecto a los DSP, radica en la celeridad con la que crece su mercado, provocando una disminución de costo y el consiguiente aumento en los rendimientos y capacidades de procesamiento (Brey, 1995). Las características de hardware paulatinamente van semejando mucho a las de un DSP. Baste como ejemplo citar la tecnología MMX que los procesadores Pentium de Intel han añadido a sus procesadores (Corporación Intel, 2001):

“LOS DISEÑADORES DE INTEL HAN AÑADIDO 57 NUEVAS Y PODEROSAS INSTRUCCIONES ESPECÍFICAMENTE DISEÑADAS PARA MANIPULAR Y PROCESAR VIDEO, AUDIO Y DATOS GRÁFICOS DE MANERA EFICIENTE. ESTAS INSTRUCCIONES ESTÁN ORIENTADAS A CONSEGUIR SECUENCIAS ALTAMENTE PARALELAS Y REPETITIVAS ENCONTRADAS CONTINUAMENTE EN OPERACIONES MULTIMEDIA.”¹

Estas novedosas capacidades son explotadas en el presente trabajo al construirse el software que permite llevar a cabo procesamiento de video en tiempo real en una computadora personal. Dicho software se apoya en el sistema de desarrollo SDK de DirectX-DirectShow el cual ofrece acceso directo a tarjetas de video y de audio conectadas en la PC. (Bargen, 1998; Dowsey, 2002).

El procesamiento de video puede incluir un amplio rango de efectos, que van desde la eliminación de componentes de color hasta los complejos que incluyen introducir parte de un video (por ejemplo la imagen en movimiento de una persona) dentro de otro video.

¹[HTTP://WWW.INTEL.COM/ESPAÑOL/INDEX.HTM](http://www.intel.com/espanol/index.htm) (MAYO 2001)

Asimismo se puede realizar procesamiento sobre video en tiempo real, y procesamiento de video en modo de edición cuadro por cuadro sobre cada imagen, para su posterior presentación. Este último tipo de procesamiento se realiza en edición de películas para añadir efectos visuales especiales y requiere costosas estaciones de trabajo con más de un procesador si se quiere ahorro de tiempo aunque una PC lo puede hacer con ciertas restricciones.

En la actualidad se dispone en el mercado de software para edición de video, pero no de software comercial para procesamiento de video en tiempo real para la PC, lo cual constituye un atractivo aliciente para este trabajo. Cabe resaltar que existen sistemas de procesamiento de video en la PC pero estos incluyen tarjetas con procesadores digitales de señales (DSP) externos.

En este proyecto se realiza solo procesamiento de video en tiempo real básico, y dado que la complejidad permisible del procesamiento viene determinada fuertemente por el hardware, se experimenta la capacidad de procesamiento de la computadora personal utilizada para establecer sus limitaciones.

Se elaboraron procesamientos básicos de video que consisten en filtrado de componentes de color, adición de ruido a la imagen, superposición de imagen al video, eliminación de color, posterizado y realce. Algunos efectos que requerían mayor capacidad de procesamiento no pudieron llevarse a cabo, sin embargo el aumento en las capacidades del hardware seguramente solventará pronto dichas restricciones.

La versatilidad de la computadora y el software actuales en la elaboración de productos multimedia interactivos, aunada a las imágenes en movimiento que pueden generarse con algún reproductor de video digital o analógico, permite captar la atención del televidente, brindando así un excelente medio de aprendizaje.

Para la enseñanza en aulas escolares es importante además, la generación de imágenes amplias que faciliten a todos los alumnos la percepción clara de los detalles en texto y esquemas. Esto se logra mediante el uso de cañoneras o video proyectores, sin embargo tienen limitaciones en cuanto a la cantidad de lúmenes que producen, y por ende esta restringido su uso a lugares con poca iluminación.

Es en este contexto que surge el interés por elaborar un dispositivo de bajo costo que permita manipular señales de video en tiempo real provenientes de una video casetera, una videocámara, un DVD o la computadora personal; y que proporcione una imagen ampliada en un

arreglo cuadrado de monitores o televisores para su uso en aulas iluminadas o inclusive al aire libre.



Figura 1 Aspecto general de un videowall de 5x5 monitores

Al arreglo de monitores o dispositivos de proyección que están conectados a un procesador digital de señales vía hardware y software, se le denomina comercialmente videowall. Sus usos comprenden anuncios de comerciales, presentación de eventos al aire libre, presentación de videos, etc. (ver figura 1, como un ejemplo de video-wall comercial). El costo de un videowall se incrementa con el número de monitores que es capaz de manejar. Como ejemplo la empresa VIDEOWALLSUSA (www.vwusa.teklab.it) ofrece un procesador videowall 2x2 (4 monitores) en US\$2,495. Por otra parte Minnesota Video Productions (www.tccom.com) ofrece la renta de un videowall 3x3 (9 monitores) por 4 días con algunos servicios adicionales en US\$3,500.

Finalmente se ha de combinar el videowall con el procesamiento del video en tiempo real aunque esto añade una gran sobrecarga a la computadora. En el presente proyecto, se pudo diseñar e implementar el videowall y correr los efectos de video simultáneamente (diseñando software propietario).

A últimos días el procesamiento digital de video en tiempo real ha tomado gran importancia y sus aplicaciones van desde la edición de películas, eliminación de ruido en la transmisión de video, compresión de video y adición de efectos especiales entre otras.

Uno de los centros con mayor producción de trabajos en este campo es el laboratorio de procesamiento de imágenes y video *VIPER* (Video and Image Processing Laboratory) de la Universidad de Purdue. Los trabajos más recientes se pueden encontrar en: <http://dynamo.ecn.purdue.edu/~ace/delp-pub.html>

En este caso el procesador asociado TMS320C80 tiene un costo de US\$993, pero un sistema de desarrollo completo requiere de un kit de herramientas y una computadora adicional cuyo costo puede ascender a los US\$4500.

Por otro lado, el grupo de investigación de procesamiento de video en el departamento de electrónica e ingeniería eléctrica de la universidad de Dublin <http://www.wdsp.ucd.ie> realizan el filtrado para eliminar errores en la imagen producidas durante el proceso de compresión, en todos los casos las pruebas e implementación se hacen en DSP para procesamiento de video.

Una investigación utilizando DirectShow se encuentra disponible en Internet bajo la dirección: <http://keeper.warhead.org.uk>. El título es "DirectShow TV display with Real Time video processing". El proyecto combina técnicas de tiempo real para la reducción de ruido y decorrelaciones derivado de publicaciones con un carácter puramente teórico. Ambos procesamientos se realizan bajo el marco de DirectShow. El filtro de procesamiento de video es un componente COM y puede ser reutilizado sin modificaciones en otros proyectos DirectShow. El filtro implementa una serie de algoritmos de reducción de ruido y decorrelaciones para ser utilizados en computadoras personales de varias velocidades.

Salvo este último caso no fue posible encontrar algún otro proyecto de investigación que utilice a DirectShow como herramienta de procesamiento de video en tiempo real. Cabe destacarse que la herramienta DirectShow tuvo su primera aparición en la versión 8.0a de DirectX disponible en 1999.

En el ámbito nacional dentro del IIMAS en la UNAM el Dr. Fabián García Nocetti trabaja en la investigación y desarrollo de arquitecturas computacionales de alto desempeño y de algoritmos eficientes para implementar sistemas paralelos en aplicaciones de tiempo real. En su centro de investigación se desarrollan arquitecturas computacionales heterogéneas, integradas por procesadores paralelos y procesadores digitales de señales (DSPs).

El resto del presente artículo está organizado de la siguiente forma: en la sección II se plantea la solución tomada, la sección III referencia a la implementación y resultados; y finalmente, en la sección IV se plantean las conclusiones y perspectivas.

2. SOLUCION PROPUESTA

2.1. HARDWARE

Los elementos de hardware utilizados para la elaboración del videowall son básicamente además de la computadora, las tarjetas de video adicionales y sus respectivos monitores.

2.1.1. MONITORES

Se realizó una comparación entre varios monitores de computadora cuyos precios oscilaban entre \$1100 y \$1600 obteniéndose la siguiente tabla 1:






					
Marca	BENQ	DAEWOO	DTS	LG	SAMSUNG
Modelo	V551	531X	1528	563N	551V
Precio	\$1,366	\$1,238	\$1,298	\$1,329	\$1,206
Tipo	CRT color	CDT	Convencional	CRT color	CRT color
Área visible	14"	14"	13.9"	13.8"	13.9"
Tratamiento	Ninguno	Antiestático, antireflejante	ninguno	Anti Glare	Anti Glare
Resolución	1024x768	1024x768	1280x1024	1024x768	1024x768

Tabla 1: Diferentes tipos de monitores considerados en el proyecto

El principal motivo de la selección reside en la resolución permisible, dado que esto expone mayor calidad en la imagen. Los tratamientos efectuados sobre la superficie del cinescopio influyen principalmente en los reflejos externos producidos, sin embargo no es una característica más importante que la anterior. Los monitores adquiridos son entonces de marca DTS y de modelo 1528.

2.1.2 TARJETAS DE VIDEO

Las tarjetas de video se han escogido de diferentes tipos con la intención de obtener sus respectivos rendimientos y realizar comparaciones. Las más comunes en el

mercado fueron seleccionadas teniéndose solo en consideración que fuesen 3 tarjetas con puerto PCI y

una tarjeta con puerto AGP, como se pueden apreciar en la tabla 2.





				
Marca y modelo	ATI 3D Rage Pro	3Dblaster Savage 4	Voodoo 4 4500 PCI	Cirrus Logic 5446 PCI
Memoria de video	4MB	8MB	32M	2M
Bus	PCI	AGP	PCI	PCI
Resolución máxima	1600x1200	1600x1200	1600x1200	1024x768
Características adicionales	Incluye acelerador gráfico de 64 bits con soporte de 2D, video y 3D	Soporta Directdraw 2D y 3D	Acelerador gráfico de 128bits 2D/3D/Video	

Tabla 2: Diferentes tipos de tarjetas de video utilizadas en el proyecto

Para la adquisición de video se ha contemplado el uso de la tarjeta WinTV-PCI de la empresa Hauppauge (www.hauppauge.com) con las siguientes características técnicas:

- El video digitalizado se envía sobre el bus PCI hacia la memoria de la tarjeta de presentación VGA, sin necesidad de uso del CPU.



Figura 2: Tarjeta de adquisición de video utilizada en el proyecto

- Conectores de entrada auxiliares de audio/video para video cámaras, VCR u otras fuentes similares.
- Captura imágenes con calidad de 24-bits y digitalización YUV 4:2:2 en un formato de color a 60 frames por segundo. El máximo tamaño de imagen digitalizada es de 640x480 para fuentes de video NTSC.

2.1.2. PROCESADOR

Se ha seleccionado el microprocesador Pentium III de Intel, a 550MHz, debido a que en el momento presente expone las mejores prestaciones y capacidad de procesamiento para aplicaciones de video en tiempo real. En particular, el Pentium III incluye instrucciones Single Instruction Multiple Data (SIMD) en punto flotante, las cuales pueden incrementar significativamente la velocidad de los algoritmos de procesamiento. Por otro lado dispone de la tecnología MMX (Matrix Math extensión) la cual es muy similar a

las instrucciones SPARC VIS que realizan operaciones con enteros en vectores de palabras de 8, 16, o 32bits. Esto facilita el procesamiento de video así como la estimación de movimiento y la interpolación.

2.1.3 SOPORTE DE MULTIMONITOR

La característica **multimonitor** le permite al sistema operativo controlar más de una tarjeta de video conectada en la misma tarjeta madre. Esta característica reduce considerablemente los costos y complejidad del videowall, y es soportada por Windows 98/Me/2000, y también por LINUX en sus versiones 7.0 de la distribución Mandrake y superiores. En todos los casos las tarjetas deben ser PCI o AGP, dado que las anteriores ISA/EISA/VESA no son soportadas. Se ha escogido Windows 98 debido a que soporta una mayor cantidad de hardware y considerando la herramienta de desarrollo disponible DirectX.

Para cada monitor se puede tener su propia resolución y cantidad de colores, y Windows permite especificar la posición de cada monitor respecto a cualquier otro. Una desventaja es que el modo de **pantalla-completa** (que se refiere al modo de video que se obtendría en una sesión de DOS en toda la pantalla o de una aplicación DirectX que ocupe toda la pantalla) solo se puede obtener en un solo monitor denominado monitor principal. Este es un problema que se espera resuelvan los desarrolladores de DirectX en el futuro.

2.2 SOFTWARE

La parte del software diseñado para esta aplicación, se llevo a cabo en visual c++, utilizando las librerías de DIRECTX y los modelos de componentes COM, los cuales se describen a continuación, la parte del programa principal se describen en el capítulo de resultados.

2.2.1 DIRECTX

DirectX® de Microsoft® es el software que permite manipular el hardware del sistema a bajo nivel. DirectX ofrece un componente de desarrollo denominado DirectShow®, el cual implementa librerías para la manipulación de elementos multimedia. Desafortunadamente no se dispone actualmente de bibliografía para el uso de DirectShow, y toda la información al respecto ha sido obtenida del sitio oficial de DirectX:

<http://www.microsoft.com/directx>

DirectX proporciona acceso de bajo nivel al hardware multimedia de forma independiente del dispositivo, adicionalmente aprovecha los últimos desarrollos en hardware a medida que se van originando como es el caso de la tecnología MMX de Intel®. Esto se logra simplemente instalando la ultima versión de DirectX.

DirectX incluye varios componentes (Figura 3) entre los que se encuentran:

- DirectDraw®. Este proporciona animación realista usando intercambio de paginas de video, acceso a coprocesadores gráficos especializados y administración de la memoria de video. También sirve de base para otros componentes como DirectShow® y Direct3D®.
- Direct3D®. Proporciona interfaces de alto y bajo nivel para generar polígonos con texturas en 3D por software y por hardware.
- DirectSound®. Proporciona sonido estéreo y 3D con mezcla de sonido por hardware, así como administración de la memoria de la tarjeta de sonido.
- DirectPlay®. Incluye servicios transparentes de mensajería independientes del medio para crear juegos con varios jugadores, así como las funciones necesarias para organizar y ejecutar un juego multijugador.
- DirectShow®. Proporciona una interfaz para el manejo de flujos multimedia provenientes de archivos o dispositivos de adquisición de audio y video.

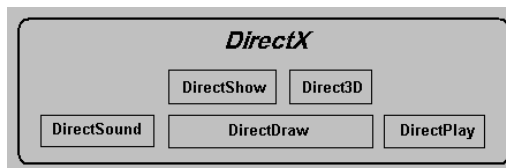


Figura 3: Componentes de DirectX

Directshow es la herramienta a utilizar en el presente trabajo, y para su comprensión es necesario revisar los conceptos sobre el modelo de objetos componentes COM desarrollado por Microsoft.

2.2.2 EL MODELO DE COMPONENTES COM

COM (Component Object Model) es un estándar creado por Microsoft que define a nivel binario como los objetos se deben crear, destruir e interactuar entre ellos. El hecho de que sea un estándar a nivel binario implica que es independiente del lenguaje que se use para trabajar con los objetos, siempre y cuando el lenguaje en cuestión sea capaz de manejar apuntadores a funciones.

Los componentes de software COM son independientes de cualquier aplicación y residirán en el sistema en

forma de DLL(Dynamic Link Library). Cada componente dispone de un identificador global único o GUID(globally unique identifier) que lo caracteriza(este identificador consiste de un entero de 128 bits y se puede generar mediante el programa guidgen.exe). Si el componente está registrado en el sistema, entonces podremos crear un objeto concreto de esa clase. Estar registrado en el sistema significa que el sistema operativo conoce donde reside "físicamente" la implementación de la clase identificada por el GUID. En el sistema Windows esta información se almacena en una base de datos centralizada llamada **registro**. Este registro se utiliza para configurar el software y hardware que se ejecute en el ambiente Windows. De esta manera, podemos disponer de toda una librería de componentes que podemos usar libremente, siempre y cuando estén registrados en nuestro sistema.

Una **interfase** es un conjunto de métodos (funciones) y atributos (datos) que tienen una fuerte relación lógica entre ellos. Un **componente** es la implementación de **una o más** interfaces y queda definida por las interfaces que implementa. La herencia de un componente se concibe a nivel de interfaces. Así todas las interfaces derivan de la interfase **IUnknown**. Esta interfase se encarga de la gestión de memoria de los objetos. La gestión de memoria se basa en un contador de referencias, y la interfase IUnknown esta formada por tres métodos **AddRef**, **Release** y **QueryInterface** para controlar este contador. Las dos primeras permiten llevar la cuenta de si el componente está siendo utilizado y cuantos clientes lo están utilizando; de esta forma el sistema puede descargar un componente cuando no se esté utilizando. La tercera permite averiguar si el objeto en cuestión implementa una interfase en particular o no, además de obtener un acceso al objeto a través de esa interfase.

3. IMPLEMENTACIÓN Y RESULTADOS

APLICACIÓN

La aplicación construye en primera instancia el gráfico de filtros completo de manera automática. La configuración obtenida depende de la fuente seleccionada por el usuario (tarjeta de TV o un archivo multimedia), si la fuente es un archivo, se debe crear una instancia del administrador del gráfico de filtros que cree el gráfico de filtros y lo ponga en ejecución.

Por otra parte debe proporcionar la ventana de video al sistema operativo para su despliegue en la ventana de la aplicación. También se requiere desplegar la página propietaria.

Si el usuario seleccionó la tarjeta de captura de video entonces se requiere la elaboración automática del gráfico de captura, para después insertar el filtro de efectos y finalmente la reproducción del video en la ventana de la aplicación. Veremos en los siguientes apartados cada uno de los pasos para la elaboración de nuestra aplicación.

2.1.3. EJECUCIÓN DE UN ARCHIVO

Una de las funciones de la aplicación es ejecutar un archivo multimedia. Esto incluye cuatro pasos básicos:

1. Crear una instancia del Manejador del Gráfico de filtros.
2. Usar el Manejador del gráfico de filtros para crear un gráfico de filtros.
3. Usar el Manejador del gráfico de filtros para ejecutar el gráfico de filtros
4. Esperar a que la ejecución se complete

Para llevar a cabo estas operaciones se hace uso de las siguientes interfaces COM:

- IGraphBuilder: Construye el grafico de filtros.
- IMediaControl: Maneja el flujo de datos en el gráfico de filtros.
- IMediaEvent: Maneja los eventos en el gráfico de filtros.

Iniciamos llamando a la función **CoInitialize**, la cual inicializa la librería COM. Posteriormente, llamamos a la función **CoCreateInstance** para crear el Manejador del gráfico de filtros.

La función **CoCreateInstance** regresa un apuntador a la interfaz **IGraphBuilder** del Manejador del gráfico de filtros. Utilizamos este apuntador para acceder a las otras dos interfaces **IMediaControl** y **IMediaEvent**. Posteriormente se coloca el corazón del programa:
`pGB->RenderFile(L"C:\\Example.avi", NULL);`
`pMediaControl->Run();`

El método **IGraphBuilder::RenderFile** construye un gráfico de filtros que ejecutará el archivo especificado. El prefijo "L" convierte una cadena ASCII a una cadena Unicode™.

Después que el Manejador del gráfico de filtros ha construido el grafico, este se encuentra listo para ejecutarse. El método Run lo lleva a cabo, y los datos multimedia comienzan a circular en el grafico de filtros.

El gráfico de filtros queda configurado entonces como en la figura 4.



Figura 4: Configuración automática del gráfico de filtros

En el presente trabajo se requiere añadir el filtro de efectos que construimos por lo cual se procede a desconectar el filtro renderer y realizar la conexión de nuestro filtro como sigue:

Primeramente, se hace la desconexión de los filtros “MPEG video decoder” y “Video render”, posteriormente, se dan de alta los filtros generados por nosotros llamados “Image Effects” y “Color space converter”; se interconectan los dos filtros anteriores para posteriormente conectarlos a los filtros “MPEG video decoder” y “Video render”, como se puede apreciar en la figura 5. Se recomienda hacer una comparación visual de la figura 4 y la figura 5.

El Gráfico de filtros resultante entonces se muestra en la figura 5.

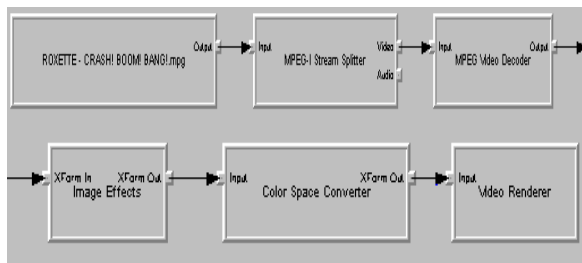


Figura 5: Gráfico con el filtro de efectos incorporados: “Image Effects” y “Color space converter”.

2.1.4. AJUSTE DE LA VENTANA DE VIDEO

Cuando se presenta un archivo de video, el grafico de filtros contendrá un filtro “video renderer”. El video renderer toma datos de video sin comprimir como entrada y presenta estos hacia la pantalla dentro de una ventana. La ventana de video “playback” es una ventana totalmente independiente con sus propios bordes y barra de titulo. Pero si se requiere que aparezca en una ventana creada por nuestra aplicación, se debe hacer que la ventana de video sea una hija de la ventana de la aplicación. Esto se logra ajustando las propiedades de la ventana de video para especificar el propietario, el estilo, y posición de la ventana, mediante la interfaz **IVideoWindow**.

Para pegar la ventana de video en la de la aplicación, entonces se hace un llamado a la función **IVideoWindow::put_Owner** y se le pasa un handle a

la ventana propietaria. Este método toma una variable del tipo OAHWND.

Se cambia el estilo de la ventana de video a una ventana hija. Esto mediante el método **IVideoWindow::put_WindowStyle** pasándole una combinación de banderas. La bandera **WS_CHILD** indica que la ventana es una ventana hija; la ventana **WS_CLIPSIBLINGS** previene a la ventana de dibujar dentro del área cliente de otra ventana hija.

Se ajusta la posición de la ventana de video mediante una llamada al método **IVideoWindow::SetWindowPosition**. Este método toma las coordenadas del dispositivo especificando la esquina superior izquierda, ancho y alto de la ventana.

Antes de que la aplicación salga, es importante que se ajuste la visibilidad de la ventana de video a false. De otro modo, un remanente de imagen de video quedaría en la pantalla y el usuario no podría quitarlo. Después se restaura el valor de owner a NULL; de otro modo los mensajes serán enviados a la ventana equivocada causando errores.

2.1.5. DESPLIEGUE DE LA PÁGINA PROPIETARIA

El filtro de efectos múltiples que añadimos soporta una página propietaria mediante la cual podemos cambiar los efectos.

Los filtros con página propietaria exponen la interfaz **ISpecifyPropertyPages**. Para determinar si un filtro define una página propietaria, se consulta al filtro mediante **QueryInterface**.

Dado que nosotros creamos una instancia del filtro, ya tenemos un apuntador al filtro de efectos, entonces solo requerimos llamar al método **ISpecifyPropertyPages::GetPages**. Este método llena un arreglo de conteo de identificadores globales únicos (GUIDs) con los identificadores de clases (CLSID) de cada página propietaria. Un arreglo de conteo es definido por una estructura **CAUUUID**, la cual se debe asignar pero no utilizar. El método **GetPages** localiza el arreglo, el cual está contenido en el miembro **pElems** de la estructura **CAUUUID**. Cuando esto se haya realizado, se libera el arreglo mediante una llamada a la función **CoTaskMemFree**.

La función **OleCreatePropertyFrame** provee una forma simple de desplegar la página propietaria dentro de una caja de dialogo. La figura 6 muestra la página propietaria con 9 efectos posibles:

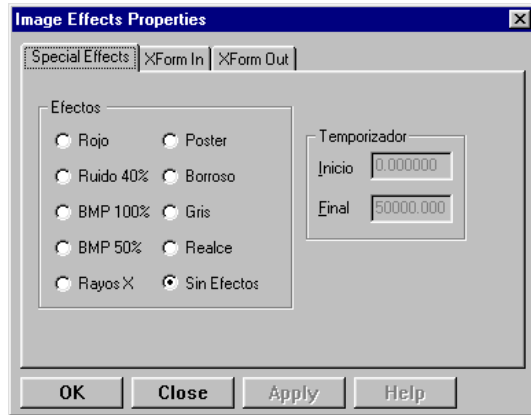


Figura 6: Página propietaria de nuestra aplicación

2.1.6. CAPTURA DE VIDEO WINTV

El manejo de video proveniente de la tarjeta de televisión winTV², previamente instalada tanto en hardware como en software, será manipulado por nuestra aplicación con el fin de poder aplicarle los efectos antes descritos, así como desplegar la imagen proveniente de wintv, en el video-wall.

DirectShow maneja la captura de video también a través de filtros. Este tipo de gráfico se denomina *gráfico de captura*.

Si un dispositivo de captura utiliza un controlador Windows Driver Model(WDM), el gráfico puede necesitar ciertos filtros “upstream” del filtro de captura. Estos filtros denominados filtros manejadores de clase flujo, soportan funcionalidades adicionales provistas por el hardware. Por ejemplo, una tarjeta de sintonía de TV tiene entre otras funciones como cambio de canal, volumen, etc.

El figura 7, muestra el diagrama de los tipos de filtros en un gráfico de captura típico. Las líneas punteadas indican donde los filtros adicionales pueden requerirse, tales como filtros decodificadores o filtros divisores.

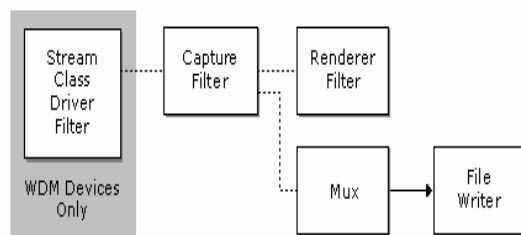


Figura 7: Gráfico de captura típico para manejo de video proveniente de la tarjeta sintonizadora de televisión (wintv).

² Tarjeta sintonizadora de televisión de la marca Hauppauge, se instala en cualquier versión de Windows

Para simplificar el proceso de construcción de gráficos de captura, DirectShow provee una componente llamada “**Capture Graph Builder**”. Este expone la interfaz **ICaptureGraphBuilder2**, la cual tiene métodos para construir y controlar gráficos de captura.

Un filtro de captura entrega datos a través de uno o más pins de salida. Un pin de salida puede ser clasificado por el tipo de datos media que éste entrega, y por su “*pin category*”.

El tipo de media se representa por un “**major type GUID**”. Los tipos más comunes de datos media se muestran en la siguiente tabla.

Tipo de Media	Major type GUID
Audio	MEDIATYPE_Audio
Interleaved digital video (DV)	MEDIATYPE_Interleaved
MPEG Flujo	MEDIATYPE_Flujo
Video	MEDIATYPE_Video

“Pin category” describe el propósito o función del pin, y es representada por un GUID propietario. Hay varios “pin category”, pero las dos utilizadas son:

- PIN_CATEGORY_CAPTURE: El pin provee datos para capturar a un archivo.
- PIN_CATEGORY_PREVIEW: El pin provee datos para visualizar.

Se puede consultar a un pin directamente por su tipo de media y su “pin category”. Para obtener el tipo de media, se llama al método **IPin::EnumMediaTypes**. Para obtener el pin category, se llama al método **IKsPropertySet::Get**. Utilizando los métodos, se puede limitar las operaciones a un tipo de media determinado o pin category, sin consultar a cada pin. A pesar de que por lógica debiéramos utilizar PIN_CATEGORY_PREVIEW en las pruebas con las cuatro pantallas no funciona, sin embargo con PIN_CATEGORY_CAPTURE si, una posible razón es el tipo de tarjeta de adquisición de video utilizada que no es completamente compatible con las funciones de DIRECTSHOW.

2.1.7. Gráfico de captura

El primer paso para crear el gráfico de captura es crear el “**Manejador del gráfico de filtros**” y el “**Capture Graph Builder**”. Entonces se llama para asociar el gráfico de filtros con el “capture graph builder”. como sigue:

```
IGraphBuilder *pGraph = NULL;
ICaptureGraphBuilder2 *pBuilder = NULL;
```

```
// Crea el FGM
CoCreateInstance(CLSID_FilterGraph, NULL,
CLSCTX_INPROC, IID_IGraphBuilder, (void
***)&pGB);
// Crea el Capture Graph Builder.
CoCreateInstance(CLSID_CaptureGraphBuilder2,
NULL, CLSCTX_INPROC,
IID_ICaptureGraphBuilder2, (void ***)&pBuilder);
// Asocia el gráfico con el builder
pBuilder->SetFiltergraph(pGB);
El siguiente paso es seleccionar un dispositivo de
captura, utilizando el “System Device Enumerator”.
El siguiente código enumera los dispositivos de captura
de video y selecciona el primer dispositivo:

// Crea el system device enumerator.
ICreateDevEnum *pDevEnum = NULL;
CoCreateInstance(CLSID_SystemDeviceEnum, NULL,
CLSCTX_INPROC, IID_ICreateDevEnum, (void
***)&pDevEnum);

// Crea un enumerador para video capture devices.
IEnumMoniker *pClassEnum = NULL;
pDevEnum-
>CreateClassEnumerator(CLSID_VideoInputDeviceCat
egory, &pClassEnum, 0);

ULONG cFetched;
IMoniker *pMoniker = NULL;
IBaseFilter *pSrc = NULL;
if (pClassEnum->Next(1, &pMoniker, &cFetched) ==
S_OK)
{
// Liga el primer moniker a un objeto filtro.
pMoniker->BindToObject(0, 0, IID_IBaseFilter,
(void***)&pSrc);
pMoniker->Release();
}
pClassEnum->Release();
pDevEnum->Release();
```

Ahora se añade el filtro de captura al gráfico:

```
// Se añade el filtro de captura a nuestro gráfico
hr = pGB->AddFilter(pSrc, L"Video Capture");
```

El paso final en la construcción de un gráfico de captura es presentar el flujo. El método **ICaptureGraphBuilder2::RenderFlujo** realiza dicha función.

```
// Muestra el pin de captura en el filtro de captura de
video.
// Se usa esta instrucción en lugar de pBuilder-
>RenderFile
```

```
hr = pBuilder->RenderFlujo
(&PIN_CATEGORY_CAPTURE
,
&MEDIATYPE_Video,
pSrc, NULL, NULL);
```

Los primeros dos parámetros especifican el pin category y el tipo de media. Los siguientes tres parámetros son apuntadores a los siguientes filtros:

- El filtro de captura
- Un filtro de compresión adicional (no usado)
- El multiplexor o filtro renderer. (no usado)

Finalmente, se incorpora el filtro desarrollado por nosotros llamado “**filtro multi efectos**”.

2.2. VIDEO-WALL

Se logró la elaboración del filtro aunque su inclusión en la aplicación resultó problemática, debido a que no se quería reconocer la existencia del filtro como librería de enlace dinámico.

En la figura 8 se muestra una imagen en pantalla completa del videowall funcionando. El video ofrece una presentación aceptable aunque se aprecia un ligero retardo en la actualización de la imagen, principalmente en las pantallas secundarias. Al ser reproducido el video a partir de la tarjeta de video WinTV, se observan los mismos problemas anteriores pero de un modo mas significativo. También se aprecia un defasamiento entre el audio y el video, presuponemos que otra tarjeta de video pudiera dar resultados distintos.



Figura 8: Video-wall de 4 monitores a pantalla completa

Se han elaborado nueve procesamientos distintos sobre la imagen, todos incluidos dentro de la función “Transform”. Se explica cada uno de ellos junto con las limitaciones encontradas. Considerese en todos los casos a $\overline{F}(x, y)$ como la función vectorial de imagen original con tres componentes R, G y B para cada color

del espacio RGB y $\overline{G}(x, y)$ como la imagen resultado de la transformación. Cabe remarcar que todos los filtros implementados funcionan en tiempo real.

Efecto rojo

Este es un filtro sencillo que elimina las componentes de color verde y azul del espacio de color RGB.

$$g_R(x, y) = \overline{F}(x, y) - (f_G(x, y) + f_B(x, y))$$

Lo único que se hace es igualar las componentes no deseadas a cero y esto resulta bastante sencillo dado que se tienen un formato de video en píxeles RGB. La estructura RGBTRIPLE mantiene un miembro para cada componente de color. numPixels es el número total de píxeles en la ventana de video. Prgb mantiene un apuntador al píxel actual. En la figura 9.a se puede apreciar en una imagen dicho efecto.

Efecto ruido

Se agrega ruido Gausiano a la imagen usando la función de generación de números aleatorios rand(), por lo tanto:

$$\overline{G}(x, y) = \overline{F}(x, y) \text{ con una probabilidad } p = \frac{2}{3}$$

$$\text{y } \overline{G}(x, y) = 255 \text{ con una probabilidad } p = \frac{1}{3}$$

Dado que se tiene 1 byte por cada color, en decimal el máximo valor es 255 para cada componente. Igualando todas las componentes a 255 obtenemos el blanco y esto lo hacemos de manera aleatoria en cada píxel produciéndose un efecto de ruido en la proporción deseada. Dado que rand() produce números entre 0 y 2 entonces la tercera parte de los píxeles en la pantalla serán blancos. En la figura 9.b se puede apreciar en una imagen dicho efecto.

Efecto imagen BMP

Aquí se superpone una imagen previamente almacenada en un archivo del disco duro con formato BMP. La imagen se carga en un arreglo temporal, sin embargo esta imagen no puede contener un número de píxeles mayor a 5000, porque el sistema se bloquea, considero que el motivo se debe a una cantidad de memoria preasignada al filtro y sobrepasarla la colapsa. Se pensó que el apuntador al píxel actual prgb siempre se encontraba en la esquina superior derecha de la ventana de video, sin embargo dependiendo de las características de la computadora y el tipo de procesamiento el origen se puede ubicar a en la izquierda y a la mitad del alto de la ventana de video. En nuestro caso se ubicó en la esquina inferior izquierda y el conteo de píxeles es hacia arriba y hacia la derecha. Es por ese motivo que la imagen BMP se debe almacenar invertida (de cabeza) para evitar un procesamiento adicional. La imagen se almacenó en el arreglo bidimensional bmiColors. El color verde puro se

utilizó como un color transparente dado que nuestra imagen no lo contenía. En la figura 9.c se puede apreciar en una imagen dicho efecto.

Efecto imagen BMP diluida

En este efecto se diluye la imagen con el video, promediando los valores de la imagen y del video obteniéndose un 50% de cada una. Se puede obtener cualquier proporción de mezcla sin mayor problema. Así entonces solo para la imagen BMP:

$$\overline{G}(x, y) = \frac{\overline{F}(x, y) + \overline{F}_{BMP}(x, y)}{2}$$

En la figura 9.d se puede apreciar en una imagen dicho efecto.

Efecto Rayos X

Este es un efecto que produce el negativo de la imagen, aquí se invierten los bits, 1 por 0 y viceversa en la imagen digitalizada.

$$\overline{G}(x, y) = NOT(\overline{F}(x, y))$$

En la figura 9.e se puede apreciar en una imagen dicho efecto.

Efecto Poster

Se obtiene un **plano de bit** correspondiente a los tres bits más significativos, esto es, se hacen cero los 5 bits menos significativos para cada componente de color, esto logra que las variaciones ligeras de color sean ignoradas y se tengan pocos colores en la imagen. Obsérvese que se logra haciendo una operación AND a nivel de bits del píxel con el byte 11100000.

$$\overline{G}(x, y) = \overline{F}(x, y) AND(11100000)$$

En la figura 9.f se puede apreciar en una imagen dicho efecto.

Efecto Borroso

Para cada píxel se toma un píxel y el siguiente a dos píxeles a su derecha, los promedia y lo sobrescribe, lográndose así un efecto de imagen borrosa.

$$\overline{G}(x, y) = \frac{\overline{F}(x+1, y) + \overline{F}(x+2, y)}{2}$$

Obsérvese la operación de bits que también podía ser aplicada al efecto BMP diluida, sin embargo dificultaría otras proporciones de la imagen. En la figura 9.g se puede apreciar en una imagen dicho efecto.

Efecto Gris

En este efecto se cambia la imagen a una escala de grises. Un cálculo estándar en base a las componentes RGB es el siguiente:

$$g(x, y) = \frac{30f_R(x, y) + 59f_G(x, y) + 11f_B(x, y)}{100}$$

Una mayor velocidad se obtiene con la siguiente simplificación:

$$g(x, y) = \frac{f_R(x, y) + f_G(x, y)}{2}$$

En la figura 9.h se puede apreciar en una imagen dicho efecto.

Efecto Detección de Contornos

En este caso hacemos una detección de borde aplicando el gradiente a la función imagen.

$$\nabla f(x, y) = [G_x, G_y] = \left[\frac{\Delta f}{\Delta x}, \frac{\Delta f}{\Delta y} \right]$$

Representada por las mascarar:

$$G_x = \frac{\Delta f}{\Delta x}$$

$$\begin{bmatrix} -1 & 1 \end{bmatrix} * f(x, y)$$

$$G_y = \frac{\Delta f}{\Delta y}$$

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} * f(x, y)$$

Si los valores de grises no son diferentes entonces se substituye un valor de gris intermedio, es decir (128, 128, 128). Si hay grandes diferencias (en contornos) entonces se alejarán más de la escala de gris intermedio. El efecto se puede apreciar en la figura 10.



Figura 9: (a) Filtro rojo, (b) Adición de ruido, (c) Superposición de imagen BMP, (d) Superposición de imagen BMP diluida, (e) Efecto Rayos X, (f) Efecto pasterizado, (g) Efecto imagen borrosa, (h) Efecto imagen en niveles de grises

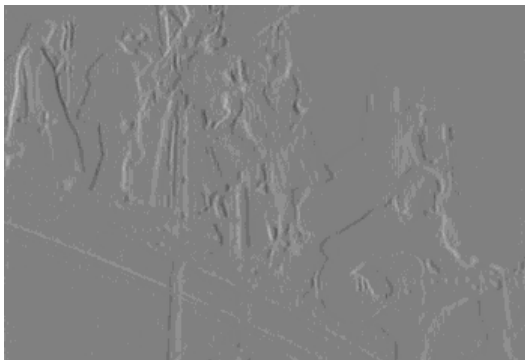


Figura 10: Efecto de realce, detección de contornos

3. CONCLUSIONES Y FUTURO TRABAJO

Como conclusiones se tiene que se logró satisfactoriamente el análisis, diseño e implementación del sistema llamado VideoWall. El sistema comprendió tanto la parte de hardware como el software de explotación respectivo. El sistema está basado en una computadora personal (con procesador de propósito genérico). Se ha obtenido el software para el procesamiento de video en tiempo real, cuyas capacidades se han explorado de manera experimental y han dejado ver las características que exhiben los modernos procesadores de propósito general. Dado que el procesamiento de video cubre un amplio rango de posibilidades y complejidad, se comenzó realizándose procesamiento básico que involucrase pocas operaciones matemáticas, y fue incrementándose la complejidad de dichas operaciones hasta obtenerse defectos en el video.

Considérese la dificultad de determinar las capacidades del presente software con antelación, considerando que se desconoce el porcentaje del procesador utilizado en las tareas básicas del sistema operativo. Entre las que se añadió el control simultáneo de varias tarjetas de video.

El procesamiento de video en tiempo real para los efectos descritos se ha logrado de manera satisfactoria solo en el caso de una pantalla, esto es, no trabajando simultáneamente con el videowall ya que tiene una considerable pérdida en la calidad visual. El procesamiento que requiere la división de imágenes para formar un videowall se obtiene también de modo aceptable al trabajar de manera independiente a los efectos. Los efectos implementados fueron nueve: (a) Filtro rojo, (b) Adición de ruido, (c) Superposición de imagen BMP, (d) Superposición de imagen BMP diluida, (e) Efecto Rayos X, (f) Efecto pasterizado, (g) Efecto imagen borrosa, (h) Efecto imagen en niveles de grises, Efecto de realce y finalmente, el efecto de detección de contornos. Los efectos anteriormente citados, funcionan adecuadamente (en tiempo real) sobre un solo monitor del videowall. Los efectos de los filtros se probaron sobre video proveniente de a) un archivo (video clip), video proveniente de una webcam y también con video de la tarjeta de adquisición de video winTV.

Las herramientas de software DirectX sobre las cuales se ha construido nuestra aplicación han sido fundamentales para el presente trabajo, algunas funciones que se han implementado en forma de objetos componentes han sido posibles también solo gracias al apoyo del grupo de usuarios en internet. En otros casos se detectaron errores en el software de desarrollo DirectShow que produjeron salidas inesperadas y que se solventaron con algún truco no documentado.

La determinación de la variación de la calidad del video respecto a la cantidad de operaciones matemáticas efectuadas, no se realizó, pero ésta puede ser también una de las tareas siguientes sobre todo por la necesidad de evaluar el desempeño en éste ámbito con los nuevos procesadores y desarrollos de hardware.

Es recomendable que se realicen pruebas con otros procesadores distintos al Intel® y con algunos dispositivos de adquisición de video diferentes, se espera también que otros sistemas operativos proporcionen características adicionales que puedan mejorar el desempeño.

Dentro de los tipos de procesamiento de video en tiempo real se tiene una amplia área de trabajo, ya sea desarrollando métodos de compresión o realizando procesamiento experimental sobre video en tiempo real.

Otra alternativa interesante es el procesamiento de video distribuido, considerando que se tienen velocidades altas en la transmisión de red local y que se dispone de mecanismos diversos de compresión se puede realizar algunos proyectos con más de una computadora.

Se espera que el presente trabajo sea de utilidad para aquellos que se interesen en procesamiento de video a bajo costo, por lo pronto es difícil la inclusión de una mayor cantidad de monitores no solo por la limitante del número de puertos PCI sino también porque se requiere mayor velocidad de procesamiento. Se puede incrementar el rendimiento si en posteriores versiones de DirectX se puede manejar pantalla completa en los monitores secundarios. Dentro de las pruebas hechas en LINUX el rendimiento fue muy bajo en el uso de multimonitor, aunque se tiene la ventaja de acceso a modificación del kernel.

Las aplicaciones de tiempo real en la actualidad también hacen uso de sistemas operativos dedicados y esta parece otra gran alternativa para mantener el bajo costo y aumentar el desempeño del procesamiento.

BIBLIOGRAFIA

- [1] **BARGEN BRADLEY**, *INSIDE DIRECTX* , **MC GRAW HILL, 1998**
- [2] Bennett David, *Visual C++ 5 Para Desarrolladores*, Prentice Hall, 1999
- [3] Brey B. B., *Los microprocesadores intel, arquitectura, programación e interfases*, Prentice Hall, 1995
- [4] Cevallos Fco. Javier, **Visual C++ 6 Aplicaciones para Win32**, Alfaomega, 2001.
- [5] Cevallos Fco. Javier, **Visual C++6 Programación Avanzada en Win32**, Alfaomega, 2002.
- [6] Dowsey Andrew M., **DirectShow TV Display with Real -Time Video Processing**,
<http://keeper.warhead.org.uk>
- [7] Eddon G. and Eddon H., **Inside Distributed COM**, Microsoft Press, (1998)
- [8] Gonzalez Rafael C., *Procesamiento digital de imágenes*, Addison Wesley, 1995
- [9] Luther, Arch C., *Principles of digital audio and video*, Artech House, Inc., 1997
- [10] Mattison P E., *Practical digital video with programming examples in C*, John Wiley & Sons, 1994.
- [11] Pascual Jorge y Charre Francisco, *Programación Avanzada en Windows 2000*, Mc Graw Hill, 1994.
- [12] Poynton Charles A., *A technical introduction to digital video*, John Wiley & Sons, 1996
- [13] Robin M. & Poulin M., **Digital Television Fundamentals**
- [14] Wallace Sean, *Digital video effects processor*, Sr. Design Project, 1998
- [15] Watkinson J., *An introduction to digital video*, Focal Press, 1994

REFERENCIAS EN INTERNET

- [16] Advanced Micro Devices (AMD)
<http://www.amd.com>
- [17] ATI Technologies Inc. <http://www.ati.com>
- [18] AV Science Forum, Section Home Theatre Computers:
- [19] <http://www.avforum.com/ubb.cgi/Ultimate.cgi?action=intro&showall=1>
- [20] AVerMedia <http://www.avermedia.com>
- [21] Corporación Intel: <http://www.intel.com>
- [22] DirectX Low-Level Extensions for Windows, Microsoft Corporation, <http://www.microsoft.com/directx>
- [23] Documentación Microsoft DirectShow,
- [24] http://msdn.microsoft.com/library/default.asp?URL=/library/psdk/directx/dx8_c/ds/default.htm
- [25] Hauppauge Computer Works
<http://www.hauppauge.com>

- [26] MSDN NEWSGROUP:
<HTTP://WWW.MICROSOFT.COM/ISAPI/REDIR.DLL?PRD=IE&PVER=5.0&AR=ISTART>
- [27] Procesamiento en tiempo real usando DSP
<http://dynamio.enc.purdue.edu/~ace/delp-pub.html>
- [28] Sun Microsystems (JMF) Java Media Framework:
- [29] <http://java.sun.com/products/java-media/jmf/index.html>.
- [30] Video4Linux2 <http://www.thedirks.org/v4l2/>.

ANIMATION OF AUTONOMOUS AVATARS OVER THE GEDA-3D AGENT ARCHITECTURE

Orozco Aguirre Héctor Rafael^a, Zaragoza Rios Jaime Alberto^a and Thalmann Daniel^b.

^aCentro de Investigación y de Estudios Avanzados del I.P.N.
Unidad Guadalajara

Av. Científica 1145, Col. El Bajío 45010 Zapopan, Jal., México
E-mail: {horozco, jzaragoz}@gdl.cinvestav.mx

^bÉcole Polytechnique Fédérale de Lausanne
EPFL IC ISIM VRLAB Station 14 CH-1015 Lausanne, Switzerland
E-mail: daniel.thalmann@epfl.ch

ABSTRACT

All interactive applications, such as computer games, video games and collaborative virtual environments, are in need of believable virtual entities. But actually, the behavior of the avatars or virtual creatures (VC) in current applications and systems is still very artificial. This work is focused on the definition of the minimum conscience of the avatars, within the GeDA-3D Agent Architecture using Conscious and Affective Personified Emotional (CAPE) Agents. The conscience and cognitive processes of the avatars allow them to solve the animation and behavior problems in a more natural way. The minimum conscience gives the avatars knowledge of how their structure is formed and of their environment. Thus, the avatars can learn movements and compute motion planning activities for acquiring basic skills. We use the CAPE Agents to control the behavior of the avatars in 3D virtual environments and we apply cognitive processes involving emotional information for animating the avatars. These processes include mainly: appraising and expressing emotions and regulating emotion in the self and others, and using emotions in adaptive ways.

Keywords: Avatar, conscience, GeDA-3D, knowledge base, agent, behavioral animation.

I. INTRODUCTION

In the last years the graphical representation and animation of VC has been focused on the use and

manipulation of predetermined animated forms and sequences. Nevertheless, in order to have autonomous avatars or VC, it is necessary to consider them as 3D semantic entities, with well-defined characteristics and functionalities. In order to make virtual environments more realistic, the avatars should exhibit a complex and believable behavior. A lot of studies about emotions exist and there are several models of emotion proposed in the literature. However, most of designed computational models of emotion only represent specific situations and respond in predetermined way to them. Nowadays, an important trend in the development of 3D dynamic virtual environments is to integrate proper characteristics of human being, such as personality, moods and emotions, into virtual characters or avatars with the aim of making them more believable and conceivable. Avatars need a convenient emotion model in order to synthesize emotions and express them. The emotion model should enable the avatars to distinguish and manage emotions in the same way that human beings do.

The human mind has been studied for many philosophers for a long time. The human consciousness is considered one of the most interesting topics in the philosophy. This topic is called philosophy of mind. An important aspect of human consciousness is the self-knowledge or self-awareness, defined as the ability to perceive and reason about oneself. This aspect is highly developed in the human being in comparison with other animals and it is considered very important

for making agents with an intelligent behavior. A human being unaware of his or her personal characteristics, abilities and skills does not know that he or she can do and cannot do and he or she will have difficulties for interacting with others in a natural way. The conscience in general is defined as the knowledge that the human being has of itself and of its environment. In this work, we define the conscience of an avatar as the notion that it has of its sensations, thoughts and feelings in a given moment in its environment. That is to say, the avatar conscience is the understanding of its environment and its self-knowledge. In other words, it is the notion that it has of its sensations, thoughts and feelings in the determined environment. Therefore, the self-knowledge makes the avatars have self-conscious and an internal representation of themselves. We use CAPE agents to develop the ability of avatar to perceive and reason about itself on the basis of the following: *consciousness* (involve thoughts, sensations, perceptions, personality, moods and emotions), *stimuli and sensorial entrances* (relevant events), *introspection* (ability of avatar to reason about of its perceptions and any conscious mental event), *awareness* (ability of avatar to be conscious and comprises perceptions and cognitive reactions to events, it does not necessarily imply understanding), *self-consciousness* (awareness and understanding of avatar, it gives the avatar the knowledge that it exists as an virtual entity separate from other avatars and virtual objects), and *qualia* (subjective properties of the perceptions and sensations of avatar).

In this work we use a knowledge base for animating avatars. We propose an ontology to provide the semantic definition and awareness of the internal structure of avatar (skeleton), its behavior (personality, emotions and moods) and its learned skills. In fact, the avatar uses the knowledge base first as a part of its conscience and second to implement a set of algorithms that constitute its cognitive knowledge. Therefore, an avatar as a human being needs to have conscience to keep equilibrium or achieve successfully complex activities. We argue that consciousness is very important and plays a crucial role in making

emotional agents with human abilities and skills. Thus, the avatars can have awareness of how its skeleton is formed (considering its mobility and physical restrictions) and also of the rules that govern its environment. At this moment, we are working at the design of autonomous avatars able to act in 3D virtual environments, maintaining the idea that an avatar is an animated virtual creature, whose motions do not need to be defined previously. But the generation of dynamic autonomous movements with high degree of realism is too complicated. However, it is possible to make models of interactions between avatars and their environment in applications of computer animation and simulation [1]. For example, the virtual humans can be used as virtual presenters, virtual guides, virtual actors or virtual teachers. Thus, the behavior of the virtual creatures such as virtual humans can be controlled using conscious emotional agents to show how humans behave in various situations.

Our main objective is not modeling the complexity of human being's behavior, but simulating conscious emotional agents with a personality and dynamic emotional behavior. Thus, the decision making and action selection of the CAPE Agents must be regulated and controlled not only by external stimuli, but also by their personality, emotions and moods. In the next section we will give an overview of related work. The third and fourth sections are dedicated to the proposal of this work. In the last section we will present the conclusions obtained from this work.

II. RELATED WORK

An avatar (autonomous virtual creature) or virtual character is a virtual entity that lives and interacts in a 3D virtual environment. Avatars are entities composed by well defined features and functionalities. Articulated models are very used for modeling avatars. The animation of such models is often based on motion capture or procedurally generated motions. Despite the availability of such techniques, the manual design of postures and motions is still widespread, because it is possible to have a total control over the results. However, it is a laborious task because

of the high number of degrees of freedom present in the models. In this section we will summarize the most important related topics and we give our opinion about them.

Motion Planning Algorithms

Motion planning has multiples applications. In Robotics is dedicated to endow robots of intelligence (autonomy) so they can plan their own movements. The problem of planning consists in finding a path for the robot since an initial point to a goal point without colliding with the obstacles in the environment [2]. In Artificial Intelligence (AI) the term planning takes a more interesting meaning. In this area the problems of planning are modeled with continuous spaces [3]. The problem of planning seems more natural and consists in defining a finite set of actions that can be applied to a discrete set of states and construct a solution by giving the appropriate sequence of actions. In [4] was presented a motion planner for computing animations for virtual mannequins cooperating to move bulky objects in cluttered environments. In this work were considered two kinds of mannequins: human figures and mobile robot manipulators.

Inverse Kinematics Algorithms

The kinematics algorithms are widely used for the animation of avatars. These algorithms use information like the position of joints angles and limbs lengths. In order to animate the avatars, structures hierarchical are formed by using a parent-child system similar to a tree. Each limb of the structure is a child node in the tree whose parent node provides a reference system from which it is describe. The parents are themselves child nodes of limbs above in the hierarchy and this recursive relationship continues up to a root node and the other end of the tree are leaf nodes which are children that have not descendants.

Reinforcement Learning Algorithms

In [5] two well-known reinforcement learning algorithms are presented. These algorithms are used for exploration, learning and visiting a virtual

environment. A different approach for animating humanoids is proposed in [6]. However, this approach has many restrictions in the used models. In this work our approach is different, because we use reinforcement learning as a cognitive process that allows the avatar to learn new skills within a certain context. Our approach is very distinct because it works with conscience that is not just knowledge but cognitive processes, which allow us to animate avatars in a most natural way.

Motion Capture and Skeleton-Based Motion Planning

In recent years the films have been successful exploding the technologies of motion capture and virtual actors. Motion capture is the process of capturing the live motion from a person or animal in order to animate an avatar or virtual character [7]. Motion capture provides an impressive ability to replicate gestures, synthetic reproduction of large and complicated movements, behavior analysis, among others. At the moment, the motion capture systems allow the collection of information for illustrating, studying and analyzing the characteristics of body limbs and joints during various motions, such as walking, running, etc. However, though impressive in the ability to replicate movement, the motion capture process is far away from perfect. Despite the longer time required to visualize the captured motion, the optical motion capture it is often preferred to magnetic technology. The avatars animation design generates libraries of postures and motion sequences using a motion capture system and later combined the obtained data with standard editing tools. In the real-time motion generation, the generation of avatars motions in real-time is based on the combination of pre-recorded sequences or dynamic motion capture allows avoiding the recording stage.

Role of the Emotions in Emotional Agents

Through the time, several models have been proposed in a broad range of scientific areas to describe the functioning of the human mind. Emotions in special have received increasing

attention and interest in several fields related to AI, mainly in Human-Computer Interaction (HCI) and Human-Robot Interaction (HRI), where emotional receptivity (perceiving and interpreting of facial expressions) and emotional expressivity (expressing emotions) are crucial and play an important role. Multi-Agent Systems (MAS) cover problems related to the autonomy, the cooperation and coordination between agents as well as the interaction of believable agents in virtual environments. In MAS, individual agents are assumed to be autonomous. That is to say, they should have the capability to deliberate and decide which actions to take or which tasks to perform in order to reach their goals. Several emotional architectures of agents have been proposed. The researchers have been particularly interested in designing models in order to make realistic and improve believability of the agents, which are applied in artificial situations, such as simulations in virtual environments.

Emotional agents are used to simulate the human being's reasoning by means of the influence and effect of emotions. We argue that it is necessary to add human characteristics, such as personality, mood and emotion, in order to design much more conceivable and believable agents. In [8] is described an architecture based on the Emotional-Belief-Desire-Intention (EBDI) agents by using emotional updating functions into four components: Emotion, Belief, Desire and Intention. This architecture demonstrated that EBDI agents have better performance than rational agents, because they have more flexibility and ability to be adaptable and survive in dynamic environments. Main roles of the emotions in agents are: action selection, motivation, adaptation, social regulation, goal management, attention focus, strategic processing and self-model.

Emotion and Personality Models Applied to Agents

The OCC model [9] is considered as a standard model for emotion synthesis and as the best model of categorization of emotions. The OCC model explains the human emotions and tries to predict under which situations which emotions can be

experimented. Emotions are divided into the following groups: reactions to events, actions and objects. This model specified 22 emotion categories based on positive or negative reactions. Thus, the consequences of an event can please or displease the agent (*pleased/displeased*), the agent can accept or reject actions (*approve/disapprove*), and the characteristics of an object can attract or not the attention of agent (*like/dislike*). The OCC model was developed to understand the emotions instead of simulating them.

The model FLAME (Fuzzy Logic Adaptive Model of Emotions) was created to produce emotions and simulate the emotional intelligence process [10]. This model uses fuzzy rules to explore the capability of fuzzy logic for modeling emotional processes, and capturing the fuzzy and complex nature of emotions. Nonetheless, it is very important to consider the personality for determining the consistency of emotional reactions over time.

The OCEAN model or Five Factor Model [11] is a purely descriptive model of personality, it groups personality traits of human being in five dimensions: *Openness*, *Conscientiousness*, *Extraversion*, *Agreeableness* and *Neuroticism*. Although this model is widely accepted, it has many criticisms, because it does not indicate how exactly the personality is affected by obtained stimuli and experienced situations.

Masuch, Hartman and Schuster [12] presented a different model represented by seven personality dimensions: *Suspicion*, *Curiosity*, *Sociability*, *Aggression*, *Helpfulness*, *Vividness* and *Conscientiousness*. These seven personality dimensions are closely related to specific aspects of more general dimensions of the OCEAN model. Due to the direct correspondence between emotions and facial expressions, many researchers prefer to employ the Ekman's six basic emotions (joy, fear, sadness, dislike, anger and surprise) for facial expression classification [13] and the OCEAN model or the OCC model in combination with the OCEAN model.

III. GeDA-3D: AN ARCHITECTURE FOR DEVELOPING CAPE AGENTS

In this work we present an extension of the GeDA-3D Agent Architecture [14, 15]. This architecture provides a platform useful for integrating and managing distributed applications, and it offers facilities to manage the communication between intelligent emotional agents. Figure 1 shows the GeDA-3D Agent Architecture. This architecture has been grouped in different modules: Virtual environments editor (VEE), rendering, GeDA-3D Kernel, agents community (AC) and context descriptor (CD). The striped rectangle encloses the main components of GeDA-3D: scene control, agent control and scene editor. The VEE includes the scene descriptor, interpreter, congruency analyzer and constraint solver. The VEE provides an interface between the GeDA-3D platform and the user; it specifies the physical laws that govern an environment and it describes a virtual scene taking place in such environment. Rendering addresses all the issues related to 3D graphics representation in a visual medium like the screen or the monitor; it allows the design of virtual objects and displaying of the scene. The EAC is composed by the agents that are in charge of ruling virtual objects behavior. The scene gives to the agents a detailed description about what we want them to do instead of how we want them to do it. Furthermore, this scene might involve a set of goals for a single agent, and it is not necessary that these goals are reached in a sequential way. It is also not necessary to give the avatar a set of actions to perform; we only need to determine a set of goals in a sequence of primitive actions. So, we need agents able to add shared skills into their global behavior. The user is enabled to construct the scene using a high level language similar to human language. The user does not provide the sequence of actions that the avatar must perform; he only specifies the goals that should be achieved by the avatars. That is, the agents behavior is in charge of trying to reach the goals specified in the scene description. The reason of creating this kind of scenes is to visualize a behaviors-based simulation, in other words, the user specifies what the agents must do, instead of

how they have to do it. Therefore, two similar specifications might produce different simulations.

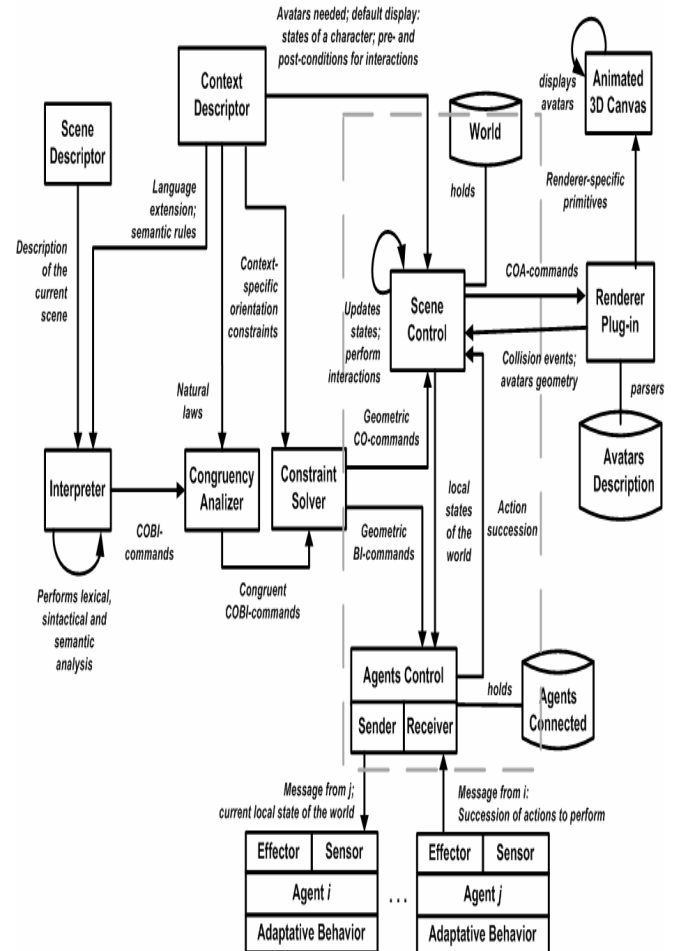


Figure 1. GeDA-3D Agent Architecture

Autonomous Avatars Animation over GeDA-3D

In order to animate avatars it is necessary to use dynamic planning and learning algorithms to compute motions. Avatars should be conscious of their internal structure (skeleton) and know how to combine simple or primitive movements to make complex activities or motions, which allow them to learn several skills and abilities. Figure 2 shows all the necessary elements and considerations to take into account, to animate articulated virtual creatures over the GeDA-3D Agent Architecture. In addition, it is necessary to use sensors in order to obtain stimuli, events and influences of the environment that can alter the behavior and motion of avatars, according to their personality, emotions and moods. For example, if an avatar is happy or

angry, its facial expression indicates its emotion, and its movements and behavior are changed; when it is sad its movements are slower than when it is happy.

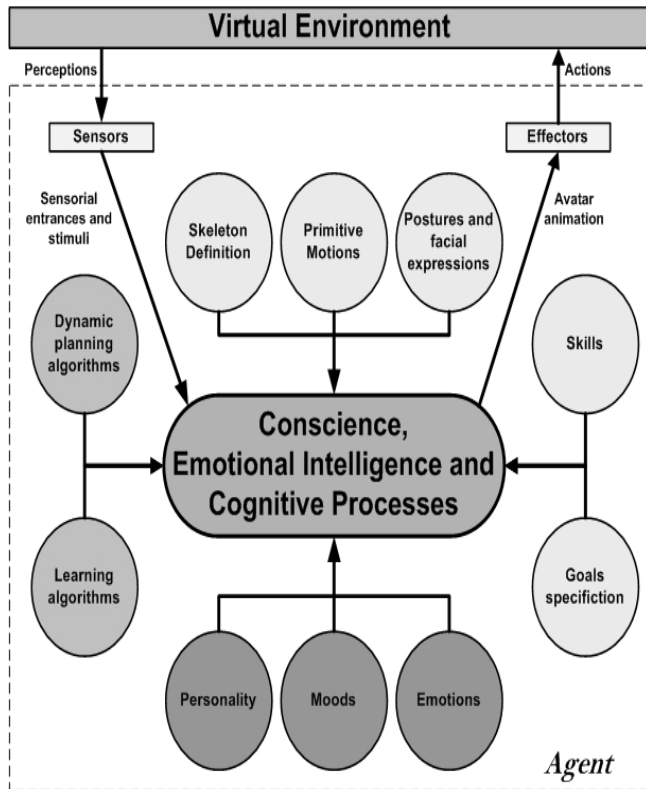


Figure 2. Necessary elements to animate avatars using CAPE Agents

In the scheme shown in figure 3 the posture and the motions of avatar are coordinated. The posture control is performed by references that indicate the direction and the required degree of stability of the avatar body. The sensorial entrances produce readjustments in the position of avatar and contribute to the modification of its corporal scheme. When the avatar makes a motion, advance adjustment of posture is produced (pro-action). The brain of the avatar contains an internal model of its corporal segments, relative sizes, their relations and positions. Therefore, the corporal scheme of avatar is the source of its corporal perception. Thus, the avatar adopts a posture and a facial expression that reflect its emotions and moods. The corporal scheme of avatar is a representation of its skeleton and its possible actions. This scheme is defined using: semantic and lexical information on the parts of the body

and the skeleton of avatar, visual-spatial representations of the avatar body and the objects of its environment (for instance, the nose is situated in the middle of the face, the ears one to each side of the head, the mouth under the nose, the eyes above the nose and one to each side of the face), and corporal references and composition of the motions on the basis of the corporal perception of avatar and its influences.

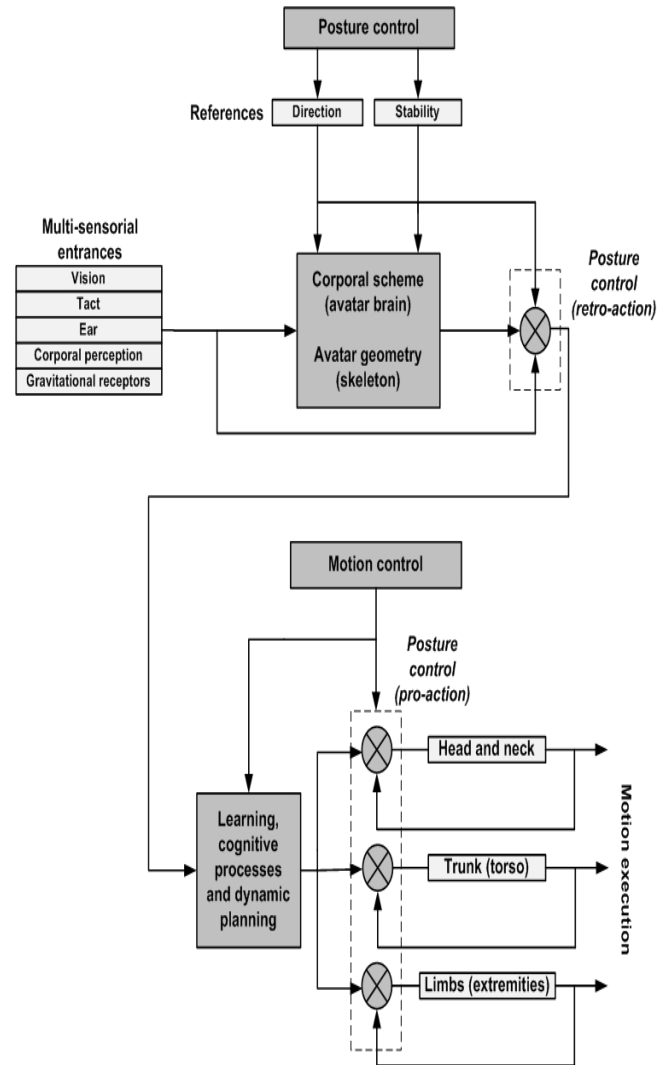


Figure 3. Control and coordination of the posture and motion of avatar

Figures 4 and 5 present the proposed modules of learning and dynamic planning for the GeDA-3D Agent Architecture. The personality, emotions and moods are related to learning and dynamic motion planning of avatar. The avatar should explore its body to know its structure and to learn a set of primitive motions, the basis for generating

complex motions. In addition, the avatar receives a set of goals and plans, and a series of motions necessary to fulfil them.

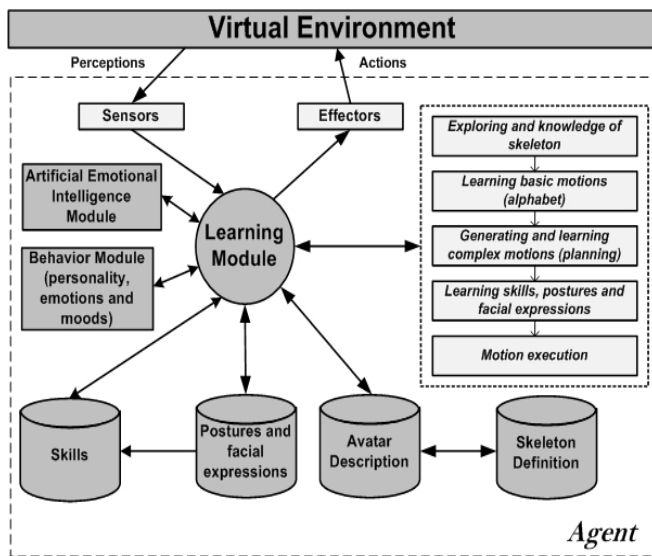


Figure 4. Learning module

The avatar explores its body to know its structure and to learn a set of primitive motions, the basis for generating complex motions. In addition, the avatar receives a set of goals and plans, and a series of motions necessary to fulfil them.

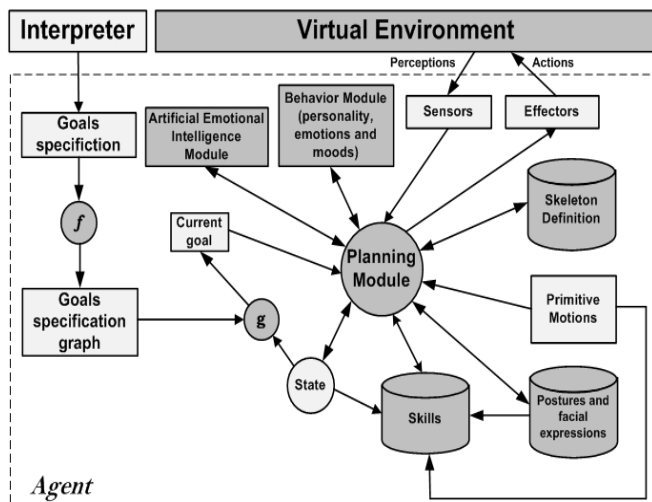


Figure 5. Dynamic motion planning module.

In this work we propose the use of synergies to support the idea that the avatar cannot control all the degrees of freedom of its skeleton. For this reason a set of simple or primitive motions is selected (natural motions) to generate complex motions. Therefore, synergies are the base of the motions of avatars and can be manipulated by

means of learning and dynamic planning algorithms.

IV. AN ONTOLOGY FOR CAPE AGENTS

In this work we propose an ontology to define the internal structure of avatar (skeleton), its behavior (personality, emotions and moods) and its learned skills. The main objective is the exploitation and use of the information and knowledge offered by the ontology in order to create autonomous animations of avatars. This ontology provides the definition of the internal structure of avatar, its behavior and its skills, and it permits share semantic information of avatar among CAPE Agents in 3D dynamic virtual environments. Thus, the proposed ontology is used as basis for applications of motion planning and motion learning of avatars. This ontology offers a formal description of CAPE Agents. Figure 6 shows the relationships between the main classes of the proposed ontology. An avatar is defined using a morphology description (qualitative description) that defines its skeleton (geometry of avatar) and the anthropometry description (quantitative description) that offers information about its age, gender, weight and height. In addition, as a part of its behavior, an avatar has a personality, emotions and moods.

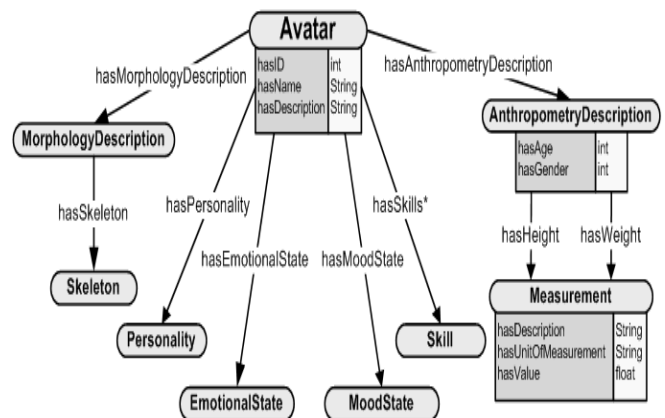


Figure 6. Semantic definition of avatar

The internal skeleton of an avatar (see figure 7) is formed by several parts, bones and joints in specific (skeleton parts). Each joint has a name and can have joints parents and/or joints children. There are motion constraints defined for each joint and a set of simple motions that define the alphabet

of basic movements (micro-animation) that will be used to generate complex motions by means of combination (macro-animation) between them. Also each bone can be united to one or more joints, and each joint has its position in the skeleton of avatar. Each bone of avatar has its measures that can be expressed in a predetermined unit of measurement, for example, in centimeters or decimeters.

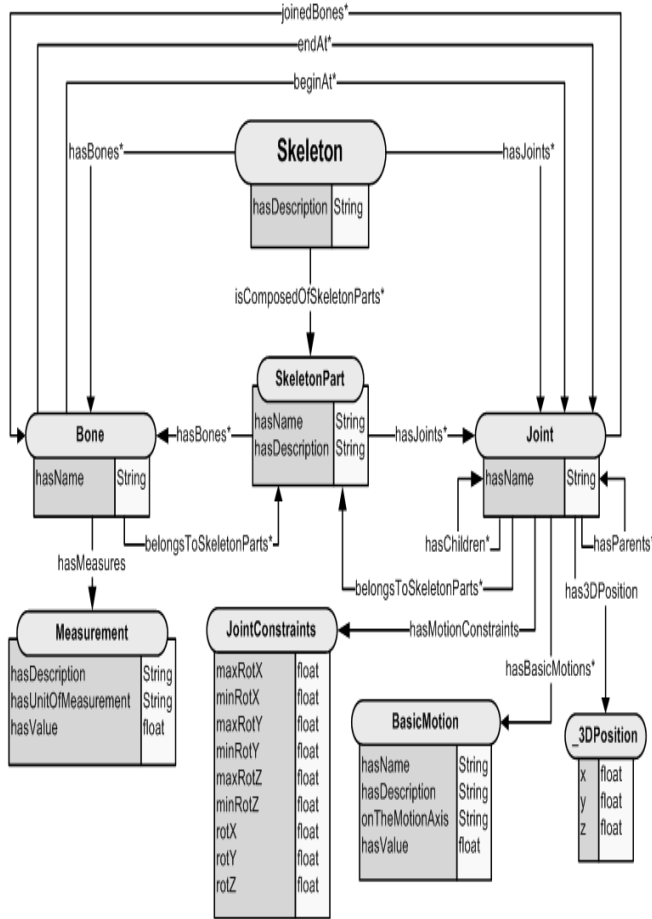


Figure 7. Skeleton definition of avatar

The emotional state of avatar is shown in figure 8. The emotional state is a set of emotions with an emotional history. A set of facial expressions and a set of postures associated with them correspond to every emotion of avatar. The facial expressions are produced with the help of facial markers (similar to joints). Body markers provide each posture with its own set of animation sequences. Emotions can be positive or negative, active or passive, but they always are of short duration. If an avatar is sad, it

is able to adopt a facial expression and posture that show its emotional state.

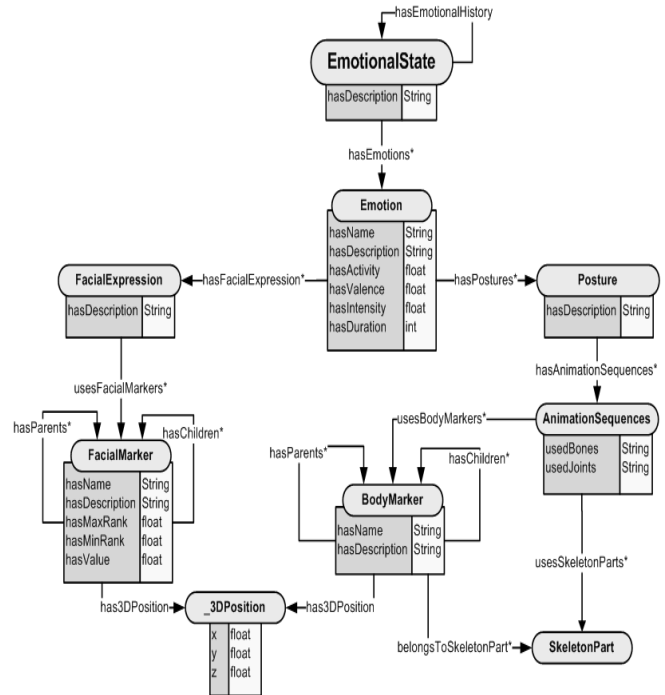


Figure 8. Emotional state of avatar

In the same way, the mood state of avatar is a set of moods with a mood history. Moods can also be positive or negative and active or passive, but also neutral. The personality of avatar is defined with the help of personality traits that can be positive or negative and active or passive. Personality traits can determine the behavior of avatar under the influence of certain events and stimuli. Finally, each skill that avatar learns is defined using a set of animation sequences. These sequences imply skeleton parts of avatar in motion and a set of corresponding facial expressions associated with them.

V. CASE STUDIES

Battle: This is a small example that shows a battle between two agents in a 2D environment (see figure 9). They are represented graphically as frogs. In this battle one frog fights against the other throwing bullets and trying to hit its rival. Both frog agents have the same kind of behavior. The behavior is based on basic actions like *move to the left or right* and *shoot*. The mixture of these three simple actions defines the decision making.

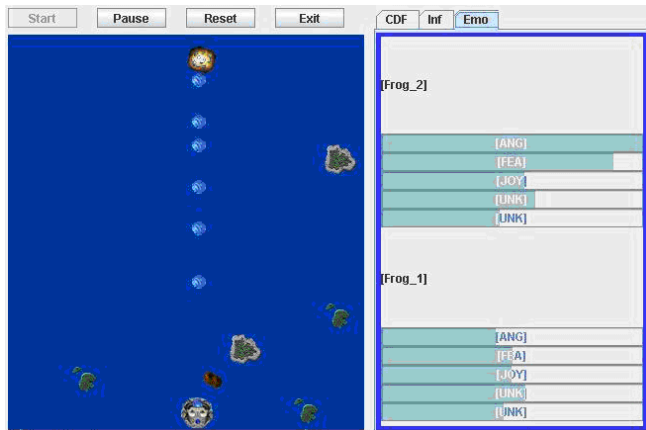


Figure 9. Frogs battle in 2D

Persecution: This case study was implemented using a virtual 3D environment which runs over the GeDA-3D Agent Architecture. Two kinds of agents are used, one is offensive and the other is defensive. The offensive agent persecutes the defensive one. The defensive agent tries to avoid the offensive agent's persecution, evading it when it is near and fleeing when it is closer. Both are emotional agents (see figure 10). Three different emotions have been defined for these agents: anger, joy and fear. The avatars modify their facial expressions and postures showing the emotion predominant in their emotional state. The implemented case studies were made using the PEM model [16] for updating the emotional and mood states of the avatars.



Figure 10. An offensive avatar that persecutes a defensive avatar

VI. CONCLUSIONS

In order to provide the avatars with a minimum conscience it is necessary to give them the capacity to understand their emotions. To understand a concept means to be able to express emotions. In this work, we have presented a methodology to determine how the agents must express their emotions. For each emotion we have defined two types of representation: a cognitive representation and a somatic representation. The cognitive representation determines the causes of the emotion. The interpretation of perceived events is based on the cognitive representation of the emotions, determined by the stimuli perceived and by the personality traits of avatar. Thus, emotion is triggered by particular interpretation of an event or stimulus that corresponds to the cognitive representation of the cause of a certain emotion. We have defined the emotional mental state as the cognitive representation of its inductive cause. The somatic representation determines how the avatars express their emotions. These representations allow to identify the mental states correspondent to each emotional state, and to determine the intensity of the emotions taking into account the influence of the avatar's personality.

In this work we presented an extension of the GeDA-3D Agent Architecture, and we have showed how the CAPE Agents must express their emotions using a knowledge base. The emotional agents' behavior and their interactions are influenced by their emotions and moods. Thus, the intelligence, the conscience and the cognitive processes help the avatars to take most accurate decisions in a more natural way. These decisions keep the agents' emotional status balanced and keep them alive in the environment. Sensorial entrances and influences the avatars receive from their environment alter their behavior and motion and also affect their emotions and mood. The GeDA-3D Agent Architecture owns the features defined in MAS theory, but it offers some extra features specific to the problem. This architecture contributes, for example, to the topics of goal-specification, skill based behaviors, collective knowledge bases, posture descriptors and facial animation descriptors.

VII. REFERENCES

- [1] H. Schmidl and M. Lin. *Geometry-Driven Physical Interaction Between Avatars and Virtual Environments*. Computer Animation and Virtual Worlds, Vol. 15, non. 3-4, pp. 229-236, 2004.
- [2] F. Schwarzer, M. Saha and J. Latombe. *Adaptive Dynamic Collision Checking for Single and Multiple Articulated Robots in Complex Environments*. IEEE Transactions On Robotics, Vol. 21, no. 3, pp. 338-353, June 2005.
- [3] S. M. LaValle. *Planning Algorithms*, Cambridge University Press, 2006.
- [4] C. Esteves, G. Arechavaleta and J. P. Motion Planning for Human-Robot Interaction in Manipulation Task. IEEE International Conference on Mechatronics and Automation, Vol. 4, pp. 1766- 1771 Laumond, 2005.
- [5] T. CondeW. Tambellini and D. Thalmann, *Behavioral Animation of Autonomous Virtual Agents Helped by Reinforcement Learning*. Lecture Notes in Computer Science, vol. 272, pp. 175-180, Springer-Verlag: Berlin, 2003.
- [6] J. Peters, S. Vijayakumar, and S. Schaal, *Reinforcement Learning for Humanoid Robotics*. International Conference on Humanoid Robots, pp. 1-20, Karlsruhe, Germany, September 2003.
- [7] L. Herda, P. Fua and D. Thalmann. *Skeleton-Based Motion Capture for Robust Reconstruction of Human Motion*. Computer Animation, pp. 77-83, Philadelphia, PA, USA, 2000.
- [8] D. Pereira, E. Oliveira, N. Moreira and L. Sarmiento. *Towards an Architecture for Emotional BDI Agents*. Proceedings of 12th Portuguese Conference on Artificial Intelligence, pp. 40-47, 2005.
- [9] A. Ortony, G. L. Clore and A. Collins. *The Cognitive Structure of Emotions*. New York: Cambridge Universty Press, 1988.
- [10] M. S. El-Nasr, J. Yen and T. R. Ioerger. *FLAME - Fuzzy Logic Adaptive Model of Emotions*. Autonomous Agents and Multi-Agent Systems, Vol 3, no. 3, pp. 219-257, 2000.
- [11] R. McCrae and O. John. *An Introduction to the Five-Factor Model and its Application*. *Journal of Personality*, vol. 60, no. 2, pp. 175-215, 1992.
- [12] M. Masuch, K. Hartman and G. Schuster. *Emotional Agents for Interactive Environments*. Proceedings of the Fourth International Conference on Creating, Connecting and Collaborating through Computing, pp. 96-102, 2006.
- [13] P. Ekman. *Moods, Emotions, and Traits, the Nature of Emotion: Fundamental Questions*. New York: Oxford University Press, 1994.
- [14] F. Zúñiga, F. F. Ramos and I. Piza. *GeDA-3D Agent Architecture*. Proceedings of the 11th International Conference on Parallel and Distributed Systems, Fukuoka, Japan, pp. 201-205, 2005.
- [15] H. I. Piza, F. Zúñiga and F. F. Ramos. *A Platform to Design and Run Dynamic Virtual Environments*. Proceedings of the 2004 International Conference on Cyberworlds, pp. 78-85, 2004.
- [16] A. Egges, S. Kshirsagar, and N. Magnenat-Thalmann. *Generic Personality and Emotion Simulation for Conversational Agents*. Computer Animation and Virtual Worlds, Vol. 15, pp. 1-13, 2004.

E-ntertainment: Sistema Web de comercio electrónico móvil para eventos culturales y de entretenimiento.

G. Guerrero¹, J. Florido², F. Mata³, B. Rico⁴

^{1,2,3,4} *Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas, IPN.
Avenida Instituto Politécnico Nacional No. 2580, Colonia Barrio la Laguna Ticomán,
Delegación Gustavo A. Madero, CP. 07340 México D.F*

¹ iran_g84@hotmail.com, ² julioflorio@hotmail.com, ³ migfel@sagitario.cic.ipn.mx, ⁴ blanca_alicia_rico@hotmail.com

RESUMEN

Presentamos el diseño de *E-ntertainment*, un sistema Web de comercio electrónico móvil que ofrece servicios de difusión del turismo, en particular, la búsqueda de eventos culturales y de entretenimiento a través de su ubicación. *E-ntertainment* muestra en un mapa la ubicación geográfica y la información descriptiva de eventos de tipo cultural y de entretenimiento, tanto para clientes Web como de telefonía celular. Adicionalmente, se ofrece un servicio de compra de boletos electrónicos para el evento, este servicio incluye el envío de boletos electrónicos a través de un mensaje SMS para usuarios de telefonía celular.

El sistema se compone de tres módulos, una aplicación WEB, un servidor de mapas, un servicio de SMS (Servicio de Mensaje Corto), un módulo para procesar compras, y una base de datos geográfica. El proceso de compra utiliza un protocolo de seguridad para proteger los datos del cliente durante la compra de un boleto electrónico. Se presenta el diseño general del sistema, así como su arquitectura y los resultados preliminares.

Palabras Clave: Sistema Web, comercio electrónico, mapas, mensaje SMS, base de datos, servidor de mapas.

I. INTRODUCCIÓN

Hoy en día, el avance vertiginoso del cómputo móvil y de las tecnologías de comunicación ha permitido que aparezcan nuevas y vistosas aplicaciones que antes sólo podían ejecutarse en una computadora de escritorio y que hoy están disponibles en los dispositivos móviles, como es el caso del comercio móvil, la publicidad y la difusión de eventos turísticos.

También, las tecnologías Web han facilitado el desarrollo de portales turísticos, por ejemplo, para reservar hospedaje y adquirir servicios de transporte generando áreas de estudio como e-Turismo (turismo electrónico) que consiste en aplicar las tecnologías de información en la industria turística [10, 11, 12]. Además, la navegación de sitios Web a través de dispositivos móviles es cada vez mas popular, ya que los contenidos ya no deben adaptarse para su visualización y despliegue en el móvil, sino que ya pueden desplegarse en los navegadores WAP que poseen la mayoría de los teléfonos celulares de hoy en día.

Por otra parte, las tecnologías Web y de Sistemas de Información Geográfica, convergen de tal forma que las aplicaciones que comúnmente un especialista SIG maneja, ahora puedan ser utilizadas por el público en general (usuarios de telefonía móvil). En otras palabras, los SIG se han popularizado permitiendo el despliegue de

mapas en pantallas de dispositivos móviles, conteniendo las funcionalidades básicas de manipulación de datos geográficos, tales como: acercamientos, alejamientos, consultas geográficas, y servicios basados en localización. Un ejemplo de lo anterior, es el portal Web Kayak.com [4] de turismo electrónico.

El rápido crecimiento y avance tecnológico en el mercado móvil se ha caracterizado por las nuevas capacidades de los teléfonos celulares, como la cámara digital, los mensajes multimedia, el acceso a Internet, la conexión infrarroja y/o de Bluetooth¹ entre otras. Todas estas funcionalidades y capacidades de conexión pueden ser aprovechadas para generar nuevas aplicaciones, como los esquemas de pago a través de los esquemas utilizados y bien probados del comercio electrónico. A la fecha estas aplicaciones han generado una nueva línea de trabajo denominada el comercio móvil. En [1,2] se trabaja bajo dicho enfoque, de donde consideramos algunos aspectos, pero enfocándonos en los elementos funcionales que en nuestro caso de estudio que es la difusión del turismo móvil.

Por otro lado, de acuerdo a la Organización para el Desarrollo y Cooperación Económica, el turismo es una de las industrias más grandes y dinámicas en la mayoría de los países desarrollados y se espera que su crecimiento siga en ascenso en los próximos años [3]. Por su parte, el Comercio Electrónico (EC) es un área ya bien establecida y con la madurez requerida para realizar transacciones de negocio a través de redes de telecomunicaciones, especialmente internet. En particular, el EC describe el proceso de comprar, vender e intercambiar productos, servicios e información vía redes de computadoras [4].

Por lo tanto, al converger el EC y el turismo podemos obtener aplicaciones que benefician al turista y al sector turístico- comercial. El turismo y los viajes están demostrando como se puede cambiar la estructura de una industria y en el

proceso crear nuevas oportunidades de negocio, como es el caso del B2B2C (negocio a negocio a consumidor). El turismo de comercio electrónico se refiere a un sistema de negocios basado en internet con distribución electrónica, utilizando información turística y un banco de comercio electrónico [5]. Las aplicaciones desarrolladas en este sentido, se han basado en las siguientes estadísticas, los consumidores utilizan los sitios web de la industria turística para obtener mapas de caminos, rutas (59%), alojamiento (54%), programación de actividades (46%), tarifas aéreas (45%), restaurantes y entretenimiento (36%) y calendarios de eventos locales (26%) de todos los sectores de la industria (Scott's Business Directory, 2002). Por ello, la industria del turismo ha experimentado un éxito mayor gracias a la adopción del EC [6].

Nuestra propuesta es un sistema Web de comercio electrónico móvil, que permite a un usuario Web, buscar eventos culturales y de entretenimiento, por su nombre o categoría (exposición, películas), desplegando resultados en un mapa con las ubicaciones sede de los eventos. En particular, la búsqueda se realiza en dos modalidades: por nombre del evento o por nombre de recinto. Entertainment mostrará la información descriptiva y de ubicación del recinto: cartelera, horario, fecha, día. En adición, el usuario puede seleccionar unos de los puntos desplegados y efectuar la compra de boletos para dicho evento (película, obra teatral, concierto).

En el escenario de compra, cuando se adquieren boletos electrónicos para cines o teatros, se utiliza el esquema del comercio electrónico donde el usuario ingresa datos de identificación personal, y de su tarjeta de crédito. Una vez realizado el pago, el sistema envía un mensaje SMS al teléfono celular del cliente, este mensaje contiene la confirmación de pago y el boleto electrónico para ingresar al evento de su preferencia.

¹ <http://spanish.bluetooth.com>

II. PLANTEAMIENTO DEL PROBLEMA

Actualmente existen eventos culturales y de entretenimiento, como cadenas de cine, teatros y auditorios donde algunos tienen una página web, para la consulta de horarios, programación, y costos relacionados con el evento. Esta información se despliega en formatos de imagen o con texto, limitando al usuario en la exploración de la información. Por otra parte, los sistemas actuales, no ofrecen el servicio de compra y de localización en conjunto, es decir, se requiere primero localizar el lugar y después realizar la compra o reservación. Integrar este proceso en una aplicación, generará los siguientes beneficios: reducción de tiempo de espera, facilitar la exploración de un sitio a través de un mapa, ofrecer un servicio de reserva o incluso de compra para los eventos encontrados. Todo lo anterior, aprovechando los protocolos de seguridad que se han consolidado en el comercio electrónico de hoy en día, como es el caso del protocolo de seguridad, *Secure Socket Layer* (SSL) el cual está bien establecido y está funcionando en el ambiente de comercio electrónico (*e-commerce*).

Un sistema que sea accesible vía el teléfono celular y que además permita recibir el boleto electrónico a través de un SMS, beneficiará al usuario móvil y a quien oferta el espectáculo, lo anterior aprovechando que el servicio de mensajes está disponible en la mayoría de los teléfonos celulares que utiliza la población de México, DF.

III. SOLUCIÓN PROPUESTA

E-ntertainment, integra un conjunto de herramientas para administrar, gestionar y difundir espectáculos y eventos de entretenimiento, así como para efectuar la venta y expedición de boletos, desde un dispositivo móvil.

E-ntertainment es un sitio WEB de comercio electrónico móvil que extiende sus capacidades al integrarse con un sistema de información geográfica, para ello, utiliza la interfaz de mapas de Google (*Google Maps*) permitiendo visualizar

sitios de interés. El sistema contiene tres bases de datos, la primera es el banco, la segunda contiene datos geográficos, y la tercera almacena datos descriptivos.

La búsqueda de eventos culturales o de entretenimiento, se realizará mediante dos opciones: por nombre del recinto o nombre del evento, la venta de boletos se realiza usando el protocolo de seguridad SSL. Se utiliza tecnología Web 2.0, el sistema de información geográfica de *ArcView*, la API de *Google Maps* [7] y el protocolo de seguridad SSL para las transacciones comerciales. *E-ntertainment* reducirá los tiempos de espera al recibir los boletos vía un SMS, con este proceso ya no se requieren impresiones ni hacer filas de espera.

IV. ARQUITECTURA DEL SISTEMA

En el funcionamiento del sistema intervienen los elementos mostrados en la siguiente figura 1.

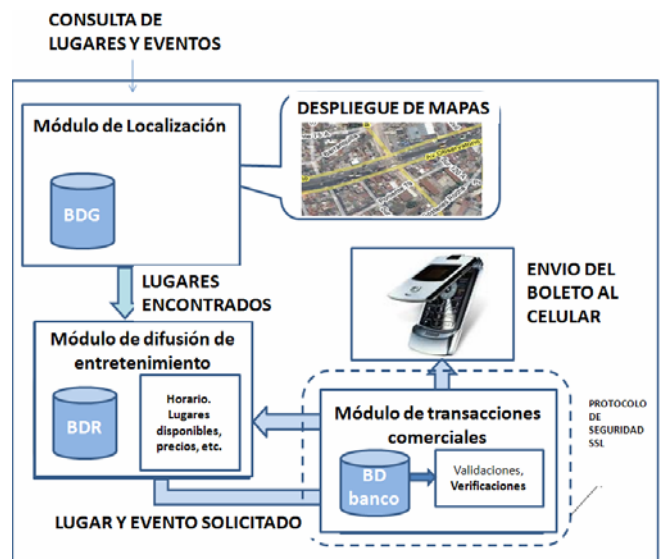


Fig. 1 Arquitectura general del sistema.

Como se aprecia en la Figura 1 se tienen cinco módulos que son: localización, información geográfica, información descriptiva, transacciones comerciales, y el de Mensajes SMS. A continuación se detalla cada uno de ellos.

IV.1 Módulo de Localización.

Este módulo procesa la información de los usuarios que buscan eventos culturales y/o de entretenimiento, tales como teatros, cines ó auditorios. Las consultas pueden incluir el nombre del teatro, cine ó auditorio. También pueden únicamente indicar el nombre del evento, se aplica un analizador sintáctico para eliminar palabras vacías (“de”, “a”, entre otras) y se procede a extraer la información asociada de la base de datos geográfica.

IV.2 Módulo de información geográfica.

En este proceso se recibe la consulta del módulo de localización, y extrae la información geográfica que coincide con la solicitud, desplegándola en un mapa y asociándola con la información descriptiva (extraída del correspondiente módulo, por ejemplo, avisos y eventos próximos a realizarse en un recinto específico). La cartografía digital utilizada en este proyecto se proceso usando *ArcView* y posteriormente utilizando la API de *Google Maps* se presenta en la Web. La figura 2 muestra una sección de la zona de Ticomán en el programa *ArcView*

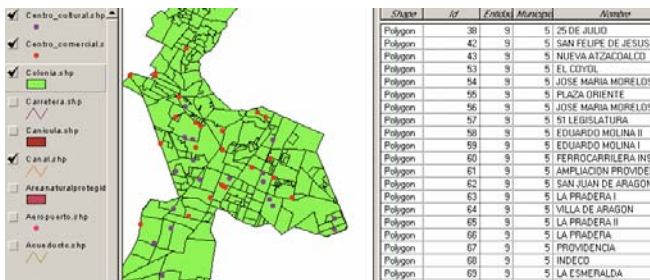


Fig. 2 Zona geográfica de Ticomán procesada en el programa *ArcView*.

IV.3 Módulo de información descriptiva.

Se encarga de mostrar la información relacionada y descriptiva de cada lugar desplegado en el mapa. Es una interfaz donde se mostrarán los horarios, descripciones, y lugares disponibles para el evento, así como la cantidad de boletos a comprar y precios. En la figura 2 también se muestra una

sección de los datos descriptivos asociados a cada punto geográfico.

IV.4 Módulo de transacciones comerciales.

Gestiona las transacciones comerciales desde que inician y hasta que finalizan, garantizando la confiabilidad y seguridad de la información. Su funcionamiento tiene lugar cuando un usuario selecciona el evento, el horario y la sección y lugar que dispondrá durante el evento. Este proceso se realiza de forma muy similar a la que se tienen en un escenario de compra vía la Web donde se proporciona el número de tarjeta de crédito para realizar la compra.

Una de las metas de este módulo es garantizar en cierto grado la seguridad de que su información personal no será publicada o accedida por terceros durante la transacción. Esto se logrará por medio de un cifrado de la información, mediante protocolos de seguridad (SSL) para transacciones electrónicas. Esta parte del sistema emula las operaciones de verificación de los datos proporcionados por él usuario (como número de tarjeta de crédito y número de cuentas, así como NIP) en las instituciones bancarias correspondientes (emuladas por un programa en nuestro caso de estudio). En este módulo se incluye un proceso de firmado digital que se describe en la sección IV 4.1.

IV.4.1 ESPECIFICACIONES PARA EL FIRMADO DIGITAL

Existe una JSR que especifica los servicios de autenticación, firmado y otros servicios de seguridad para dispositivos móviles como celulares y PDA's ; éste es el JSR 177, SATSA (*Security And Trust Services API*)². Esta especificación para J2ME es relativamente nueva y por lo tanto no es soportada por la mayoría de los celulares en la actualidad, es por ello, que se decidió trabajar con una API de uso libre del

² <http://jcp.org/en/jsr/detail?id=177>

grupo “Bouncy Castle”³, aunque tiene como desventajas la poca documentación y soporte nulo. En el caso de la firma digital en el *SalesPoint*, se usará el paquete de *java.security*. Todos estos API’S e implementaciones están definidas dentro de la JCA (Java Crypto Architecture).

Al contar con la confirmación de que los datos proporcionados por el usuario son correctos, se realizan los cargos del costo de los boletos comprados y se generan los comprobantes de estos boletos (boleto digital) para su envío vía SMS, este proceso es descrito en el módulo de envío de mensajes SMS. En este caso si el número de la tarjeta de crédito es rechazado, se despliega de nueva cuenta la interfaz de compra.

IV.5 Módulo de SMS

Envía mensajes de texto (SMS) al usuario notificando y confirmando la compra, así como el boleto electrónico en formato XML o en un formato de texto plano.

Esta confirmación será su pase de acceso al presentarse al evento cultural o de entretenimiento, reduciendo tiempo de espera ya que solo tendrá que mostrar el mensaje en su celular.

V. RESULTADOS PRELIMINARES

A continuación de presentan imágenes de los primeros resultados obtenidos con el sistema, estos módulos están funcionando de forma separada, el último paso consiste en su integración que es la que está actualmente en desarrollo.

Para explicar el caso de estudio del sistema *E-ntertainment* nos basaremos en el siguiente ejemplo: suponga un usuario que desea saber donde se presenta el evento “*Disney sobre*

Hielo”, entonces accede al sistema y se despliega un mapa y los campos de texto para la consulta y tipo de búsqueda.

Una vez desplegada la interfaz, el usuario puede seleccionar el tipo de búsqueda con base en dos opciones: por nombre del evento (si se conocen los datos específicos del evento al que deseamos asistir) o por nombre del recinto.

Por ejemplo, si se desea conocer los recintos donde tendrá lugar el espectáculo “Disney sobre hielo”, obtenemos el mapa con la ubicación de los recintos en panel derecho (usando la API de Google Maps) y en el panel izquierdo los nombres de eventos y su descripción asociada. La Figura 3 ilustra este proceso.



Fig.3 Resultado de búsqueda

Como se aprecia en la Figura 3 se muestran dos sitios que presentan el espectáculo buscado, en este punto si al usuario le interesa obtener mayor información sobre el recinto o el espectáculo “Disney sobre Hielo”, podrá hacer clic en el ícono que representa a dicho recinto. Entonces se desplegará la cartelera existente en este recinto, información descriptiva del evento, y sugerencias de cómo llegar al lugar. Este proceso se muestra en la Figura 4.

³ <http://jcp.org/en/jsr/detail?id=177>

CARTELERA SELECCIONADA		
EVENTO	FECHA	HORARIO
	SABADO 24/AGOSTO/2008	6:00 PM,7:00PM
	DOMINGO 25/AGOSTO/2008	5:00PM,8:00PM
COMPRAR BOLETOS		

Fig. 4 Visualización de cartelera

Como se aprecia en la Figura 4, se tienen los horarios e información relacionada con el evento, en este punto si el usuario desea realizar la compra de un boleto y asistir al espectáculo “**Disney sobre Hielo**”, primero deberá seleccionar las secciones y lugares disponibles y horarios, al seleccionarlos se muestran precios, fechas y detalles generales de su selección.

En este punto inicia el proceso de compra y donde interviene el protocolo de seguridad. Si el usuario desea realizar la compra deberá proceder a llenar un formulario con los datos de su tarjeta de crédito, nombre, número y fecha de vencimiento. El monto de la compra se generará automáticamente dependiendo de la sección y el número de boletos que el usuario seleccionó.

La comunicación que se realice desde el cliente hacia el banco emisor de la tarjeta de crédito estará protegida por un protocolo de seguridad de transacciones electrónicas (SSL), evitando así los fraudes electrónicos a los cuales la mayoría de los usuarios de internet están expuestos. El último paso, es la generación de los métodos de facturación para que el usuario logre ingresar al evento sin la necesidad de hacer largas filas esperando a que le entreguen su boleto reservado o una constancia que le avale que compró el lugar en el horario establecido.

E-ntertainment incluye dos modalidades para el envío del boleto electrónico, la primera es por medio de un correo electrónico y la segunda es a través de un mensaje SMS a un teléfono que el

usuario indique, en este caso el usuario puede elegir si se envía el boleto vía correo electrónico o en un mensaje SMS. Los datos de este boleto se recuperan a partir de la información recopilada durante la selección del evento hasta el ingreso de datos del usuario para realizar la compra. El formato del boleto generado es en un archivo XML, el cual servirá como pase de acceso al evento que el usuario indicó. Un fragmento del formato de este mensaje se muestra en la figura 5.

```

- <Boleto>
  - <Evento>
    <ID> 3674 </ID>
    <Nombre>Disney On Ice</Nombre>
    <Hora>15:00</Hora>
    <Lugar>auditorio</Lugar>
  </Evento>
</Boleto>

```

Fig. 5 Formato XML de boleto electrónico

La figura 6 muestra un ejemplo del mensaje desplegado en la pantalla del teléfono. El formato de este mensaje es sencillo, hasta el momento solo contiene los datos básicos del evento. Se está trabajando en éste módulo para integrar los elementos de seguridad y especificaciones para su validez e identificación segura.



Fig. 6 Envío de mensaje SMS

Es posible incluir otros elementos para una identificación del boleto más precisa. Este último aspecto se está considerando dentro del desarrollo que se está llevando a cabo actualmente.

El siguiente diagrama a bloques resume los pasos a seguir en el sistema para realizar búsquedas de otro evento.

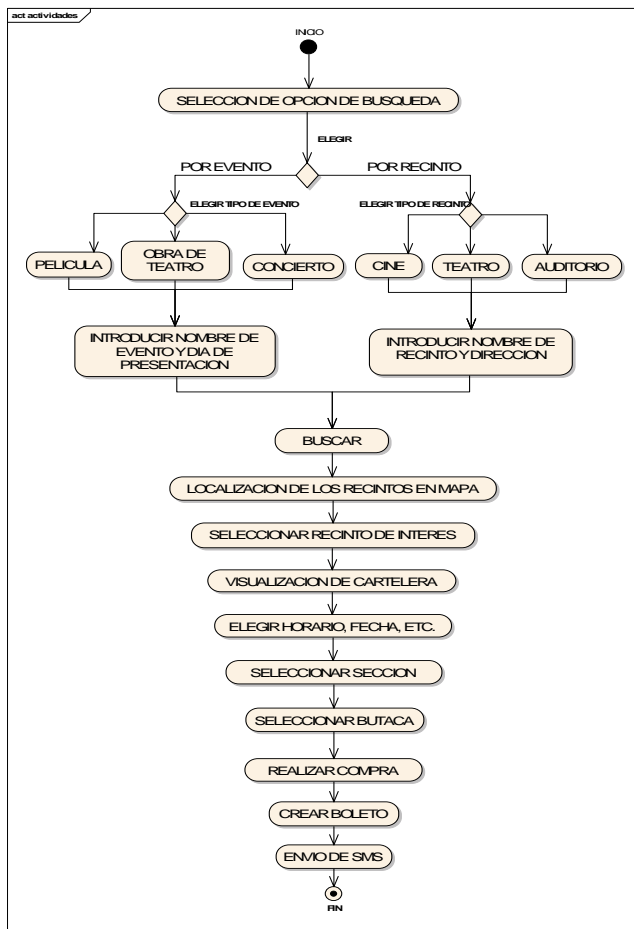


Fig. 7 Diagrama a Bloques del Sistema

CONCLUSIONES

Se ha presentado un sistema que integra un servicio de compra electrónica, con un sistema GIS, y un sistema WEB para resolver consultas en el ámbito turístico y de entretenimiento. En particular, se mostró el caso de estudio para eventos culturales y de entretenimiento.

Como trabajo a futuro se plantea integrar otras fuentes de información, como es el caso de video y de voz.

El módulo de comercio electrónico también puede integrarse con sistemas de comercio móvil, que

trabajan con mecanismos de comunicación Bluetooth.

Se requieren pruebas de desempeño y rendimiento cuando existan un mayor número de usuarios usando el sistema. Además, plantear y analizar los escenarios de concurrencia que pueden presentarse en el sistema.

El sistema cuenta ya con la fase de diseño y análisis terminado, mientras que la fase de desarrollo tiene módulos separados con un funcionamiento correcto y esperado, actualmente se está trabajando en la integración de los módulos para obtener el producto final.

Este sistema es parte de *Turicel* un sistema integral que ofrece servicios de turismo en general, para la población del Distrito Federal. Actualmente este proyecto está en desarrollo en la UPIITA-IPN en conjunto con el Instituto de Ciencia y Tecnología del DF, quien lo patrocina.

AGRADECIMIENTOS

Los autores de este artículo agradecen al Instituto Politécnico Nacional, a la Comisión de Operación y Fomento a las Actividades Académicas del IPN (COFAA), a la Secretaria de Investigación y Posgrado (SIP) y al Instituto de Ciencia y Tecnología del DF por el apoyo brindado a este trabajo; así como a los revisores de este trabajo por sus útiles comentarios.

REFERENCIAS

- [1] Zakaria Maamar, "Commerce, e-commerce, and m-commerce: what comes next?" Communications of the ACM, Volume 46 , Issue 12 (December 2003) ISSN:0001-0782.
- [2] Gary Shih, Simon S. Y. Shim. Service management framework for M-commerce applications. Mobile Networks and Application, ISSN:1383-469X Volume 7 , Issue 3, June 2002. Hingham, MA, USA
- [3] http://strategis.ic.gc.ca/epic/internet/indsibtour.nsf/en/h_qq0

0102e.html, Tourism Counts: A Consultation Framework for a National Tourism Strategy, Nov, 05, 2003.

[4] Efraim Turban, Jae Lee, David King and H. Michael Chung. Electronic Commerce: A Managerial Perspective. Higher Education Press, Pearson Education, 5(2001), 4-5.

[5] Chen Guogui, Hou Weihua and Wang Yanzhang, the development and countermeasures research of Chinese tourism e-commerce, Academy Journal of Dalian University of Technology, 10.

[6] Schuster, A. A Delphi survey on electronic distribution channels for intermediaries in the tourism industry: The situation in German speaking countries. In Proceedings of the Enter 2002 Conference. Springer Verlag, Wien, NY, 224-234.

[7] Google Maps, www.googlemaps.com

[8] <http://www.ciies.ipn.mx/inicioE.html#>

[9] Available: <http://www.kayak.com/help/about.html>

[10] http://en.wikipedia.org/wiki/Travel_technology

[11] J. Cardoso. E-Tourism: Creating Dynamic Packages using Semantic Web Processes (2006)

[12] World Tourism Organization. WTO. Tourism 2020 Vision. <http://www.unwto.org/facts/eng/vision.htm>

COMPOSICIÓN DE PROTOCOLOS DE INTERACCIÓN UTILIZANDO OBLIGACIONES

Gutiérrez-García J. Octavio^{1, 2}, Koning Jean-Luc¹ y Ramos-Corchado Félix F.²

¹Grenoble Institute of Technology, LCIS Research Laboratory,

50, rue Barthélémy de Laffemas, BP 54, 26902 Valence Cedex 9, France

²Centro de Investigación y de Estudios Avanzados del IPN, Unidad Guadalajara,

Av. Científica 1145, Col. El Bajío, Zapopan 45015, Jalisco, México

jgutierrez@gdl.cinvestav.mx, jean-luc.koning@grenoble-inp.fr, framos@gdl.cinvestav.mx

RESUMEN

En este artículo proponemos la composición de protocolos de interacción a través del uso de obligaciones, las cuales proporcionan una capa semántica, que brinda instrumentos para razonar sobre los protocolos. Además definimos una metodología basada en el enfoque de componentes, para desarrollar módulos independientes con interfaces adecuadas para su extensión, aumentando reusabilidad de los protocolos de interacción.

Palabras clave: Sistemas Multi-Agentes, Normas sociales, obligaciones, protocolos de interacción, composición y reusabilidad.

I. INTRODUCCIÓN

Actualmente, controlar la interacción y coordinación en los Sistemas Multi-Agentes se ha convertido en un desafío, debido a la dinamicidad y complejidad de las aplicaciones en las cuales son aplicados. Dicha interacción es regida por un conjunto de reglas, llamadas protocolos de interacción.

La reutilización de los protocolos es actualmente un área de investigación activa. Usualmente las reglas de interacción son especializadas y adaptadas para cada caso en particular, lo cual anula la posibilidad de reutilizar los protocolos de interacción. Igualmente las reglas de interacción carecen de semántica, lo que dificulta aun más su categorización y caracterización, lo cual podría ayudar a su reutilización; otro problema con el cual

nos enfrentamos es la división y encapsulación de las reglas de interacción, ya que estas se encuentran fuertemente entrelazadas en todo el proceso.

Los protocolos de interacción han sido representados usando lenguajes de modelado formales e informales, tales como: A-UML [1], máquinas de estado finito [3], redes de Petri [2], entre otros (para más detalles ver [5]); sin embargo la mayoría de las técnicas de modelado generan una representación compleja y monolítica del protocolo, entrelazando fuertemente el proceso de interacción, deshabilitando así, la reutilización de los protocolos. Esfuerzos para aminorar este problema se encuentran en [6 & 8], ambos enfoques presentan una propuesta para encapsular la interacción en micro-protocolos, posteriormente estos son ensamblados entre sí para construir el protocolo de interacción, facilitando así la reusabilidad y verificación formal de sus propiedades, sin embargo la definición de las reglas de interacción carece de expresividad que facilite la caracterización de los micro-protocolos para su posterior clasificación y reutilización.

La solución propuesta en éste artículo, involucra la adición de una capa semántica a los protocolos, dicha capa esta basada en la utilización de normas sociales que proveen información descriptiva de las secuencias de mensajes enviados. Con el apoyo de la semántica provista, los módulos de interacción son agrupados, utilizando un enfoque basado en componentes, donde cada modulo es equipado con una interfaz para acoplarse a otros módulos. La selección y acoplamiento de los

módulos es asistida por la semántica de las normas sociales, en éste caso, las obligaciones, las cuales definen la funcionalidad del módulo, facilitando su clasificación y caracterización, además de guiar la manera en la que los componentes interactúan.

Éste artículo esta estructurado de la siguiente manera: en la sección 2, se explica la definición y manejo de las obligaciones como normas sociales que dirigen los protocolos; la sección 3 contiene la descripción del modelo basado en componentes; la sección 4 incluye la metodología utilizada para seleccionar y componer los protocolos; en la sección 5 se presenta un ejemplo de aplicación de la técnica propuesta; finalmente, en la sección 6 se presentan las conclusiones obtenidas y las direcciones del trabajo futuro.

II. OBLIGACIONES COMO REGULADORES DE INTERACCIÓN

Una obligación es una imposición moral a sí mismo de alcanzar algún estado en particular, es representada simbólicamente de la siguiente manera: $O(\text{agt}, f)$, lo cual significa que el agente **agt** esta obligado a alcanzar el estado **f**. Otro tipo de obligaciones son las condicionales, denotadas por $CO(\text{agt}, f, g)$, la cual depende del estado **g** para activar o desactivar la obligación **f** al agente **agt**.

A continuación presentamos un listado de las operaciones que tienen efecto sobre las obligaciones:

- **CrearO(a(agt₁), O(agt₂, f))**: operación utilizada para crear obligaciones, el símbolo **a(agt₁)** indica que la acción **a** es ejecutada por el agente **agt₁**, dando como resultado la creación de la obligación **f** sobre el agente **agt₂**. Cabe mencionar que una obligación puede ser adoptada por uno mismo o puede ser inducida socialmente por otro agente, por ejemplo cuando se emite un saludo a una persona, la obligación de contestar el saludo es inducida, y en caso de que la obligación sea aceptada por el agente, éste regresa el

saludo, en ese momento decimos que pasamos de las obligaciones a la interacción.

- **LiberarO(a(agt₁), O(agt₂, f))**: la liberación de una obligación toma lugar, cuando el estado **f** ha sido alcanzado, a través de la ejecución de la acción **a**. Un agente puede liberarse de sus propias obligaciones, en el caso de obligaciones autoinducidas, por otra parte la liberación de obligaciones externamente inducidas se lleva acabo por los agentes que supervisan el cumplimiento de la obligación.
- **CancelarO(a(agt₁), O(agt₂, f), F^c, F^d)**: La cancelación de una obligación lleva consigo la creación de obligaciones adicionales que compensen su cancelación, denotado por el conjunto F^c , además es necesario considerar un efecto en cascada de cancelación de obligaciones vinculadas a la primera, esto es denotado por el conjunto F^d .

La definición de las obligaciones esta formalizada con el cálculo de eventos [7], el cual es un formalismo basado en lógica temporal para representar eventos y sus efectos. En [4] presentamos la definición formal de las obligaciones, así como los axiomas que controlan la evolución de las mismas.

III. UN ENFOQUE DE COMPONENTES PARA LA DEFINICIÓN DE PROTOCOLOS

Nosotros tomamos como base los principios que fundamentan los enfoques orientados a componentes, para desarrollar una estructura que soporta la modularización de los protocolos de interacción, dicha estructura es llamada organización. Una organización contiene un dominio de interacción, el cual define el perfil de la misma, expresando sus funciones y estructura. Cada organización esta compuesta de un conjunto de roles, que representan abstracciones de las funciones de la organización, cada rol cuenta con su propio contexto de interacción, obligaciones

Contexto de Interacción	$W = \{ \text{Obligación}_1 \uparrow \varpi^*, \text{Obligación}_2 \downarrow, \text{Obligación}_3 \downarrow \uparrow^*, \dots \}$
Obligaciones iniciales	$I = \{ \text{Mantiene}(\text{O}(\text{ROL}_A, \text{Obligación}_1), 0) \}$
Precondiciones	$P_1: \{ \text{Mantiene}(\text{O}(\text{ROL}_A, \text{Obligación}_1), t) \}$
Mensajes	$M_1: \text{Mensaje}_1(x);$
Efectos	$E_1: \{ \text{LiberaO}(\text{Mensaje}_1(x), \text{O}(\text{ROL}_A, \text{Obligación}_1),$ $\text{CrearO}(\text{Mensaje}_1(x), \text{O}(\text{ROL}_B, \text{Obligación}_4)) \}$

Figura 1. Ejemplo de definición genérica de un rol

iniciales, mensajes y estado de interacción (como se observa en la figura 1). A continuación presentamos la definición detallada de cada uno de éstos componentes:

- a) **Contexto de la interacción (W):** éste se encuentra definido por un conjunto de obligaciones, las cuales representan las funciones del rol y la interfaz para acoplarse con los demás roles pertenecientes a la organización. Cada una de las obligaciones listadas en el contexto de interacción puede tener cuatro propiedades, las cuales indican como es tratada la obligación dentro del rol, a continuación presentamos su definición:
- **Propiedad de Auto-Inducción (ϖ):** indica que el agente adopta la propiedad por sí mismo.
 - **Propiedad de Inducción de adentro hacia fuera (\uparrow):** indica que la obligación puede ser inducida a un rol externo.
 - **Propiedad de Inducción de afuera hacia dentro (\downarrow):** indica que la inducción de la obligación puede ser realizada por un rol externo.
 - **Propiedad Final (*):** Indica que la obligación es liberada por el rol.
- b) **Obligaciones iniciales (I):** son las obligaciones que un agente adopta por el simple hecho de interpretar el rol, dichas obligaciones pueden ser interpretadas como los estados iniciales de la interacción.

c) **Mensajes (M):** La ejecución de los mensajes esta controlada por precondiciones, definidas en términos de obligaciones. Los efectos de un mensaje están representados por la creación y liberación de obligaciones.

d) **Estado de interacción.** El estado actual de interacción esta determinado por las obligaciones que un agente esta comprometido a alcanzar en determinado momento.

Una vez definidos los roles, se procede con su agrupación en una organización, la cual representa su funcionalidad con la unión de los contextos de interacción de cada uno de los roles que la conforman.

El estado de la interacción de una organización se mide en base a las obligaciones que cada agente que ejecuta el protocolo tiene con los demás participantes. El contexto de interacción de la organización juega dos papeles importantes: el primero es describir la funcionalidad de la organización y el segundo es fungir como interfaz con otras organizaciones. El primer papel se puede lograr vinculando las obligaciones a una ontología común, después, las organizaciones pueden ser indexadas, facilitando el acceso a los diseñadores dentro de un repositorio. En lo que respecta al segundo papel, cada una de las obligaciones que se encuentran en el contexto de interacción esta etiquetada con propiedades de inducción hacia

fuera o hacia dentro, las cuales permiten la interconexión entre las organizaciones.

IV. COMPOSICIONES DE PROTOCOLOS

Razones tales como delimitación de funciones y reutilización de protocolos de interacción requiere que los protocolos sean vistos como módulos, los cuales permitan ser agregados, modificados o retirados, sin afectar las otras partes que conformen el protocolo, sin embargo uno de los problemas a vencer, es determinar el procedimiento adecuado para llevar a cabo estas operaciones sobre los protocolos manteniendo consistencia en la interacción.

Podemos encontrar dos tipos de composición de protocolos, la primera es la secuencial en la cual un nuevo componente es agregado al inicio o al final de otro componente; el segundo tipo de composición es la entrelazada, ésta se refiere a la ejecución alternada de los protocolos compuestos. La composición entrelazada requiere mayor atención para mantener la consistencia de la interacción. La definición del estado de interacción por medio de obligaciones, permite sostener interacción basada en objetivos y no en secuencias de mensajes, esto facilita en gran medida la composición secuencial y entrelazada, ya que el proceso de composición es llevado a cabo de una manera incremental, añadiendo obligaciones y mensajes que asignan o liberan obligaciones dentro del protocolo.

Como interfaz de los protocolos, utilizamos las obligaciones que no pueden ser liberadas dentro de una organización dada. Una organización es diseñada como un componente en el cual su función principal esta completamente cubierta por su protocolo de interacción, pero sus actividades secundarias están vinculadas a otras organizaciones, esto permite la delimitación y especialización de funciones. Las interfaces están representadas en el contexto de interacción, donde se especifica que obligaciones pueden ser inducidas hacia otras organizaciones, así como las

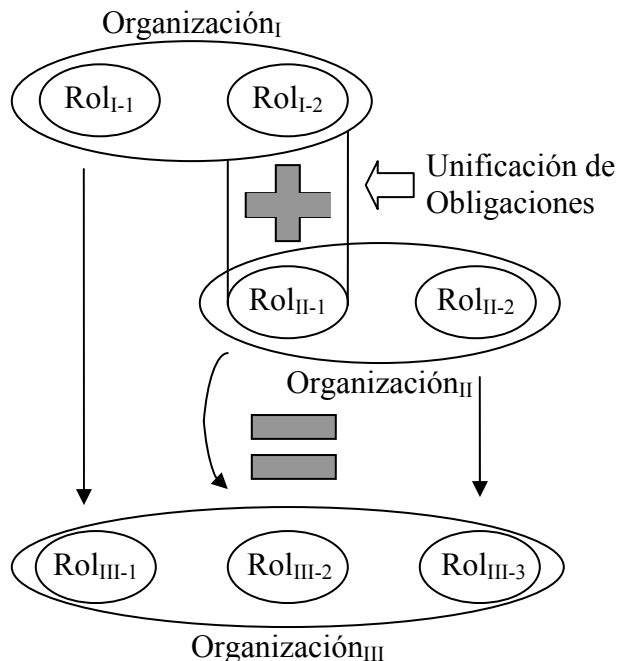


Figura 2. Composición de organizaciones

obligaciones que son aceptadas y liberadas dentro de la organización.

Una obligación interfaz es toda aquella obligación que no puede ser liberada dentro del contexto de una organización; éstas son identificadas por la ausencia de la propiedad final (*) en la definición del contexto, en el ejemplo presentado en la figura 1 la **Obligación₂** es una obligación interfaz. Una vez identificadas las obligaciones interfaces es necesario realizar una unificación de las obligaciones de ambas organizaciones. Al hacer la unificación de las organizaciones no solo estamos vinculando una obligación con otra, sino que también unimos los roles involucrados en cada organización (ver figura 2). Existen dos requisitos para realizar la unión de una organización:

- El primero tiene que ver con la correspondencia semántica entre las dos obligaciones, esto puede llevarse a cabo por medio de técnicas de representación de conocimiento tales como ontologías, o bien puede ser validado manualmente por el diseñador del sistema.

- El segundo requisito tiene que ver con la estructura del contexto de interacción de cada organización, donde una de las obligaciones tiene que permitir la inducción de adentro hacia fuera (\uparrow) y la otra de afuera hacia dentro (\downarrow).

La composición de dos organizaciones Org_I y Org_{II} se lleva a cabo de la siguiente manera: se identifican las obligaciones interfaz O_I y O_{II} , pertenecientes al Rol_{I-2} y Rol_{II-1} de las organizaciones Org_I y Org_{II} respectivamente (ver figura 2). La obligación resultante de la unificación se denomina O_{III} , la cual sustituye toda instancia de las obligaciones O_I y O_{II} , en sus respectivos roles, ahora ambos roles quedan unidos debido a la unificación de las obligaciones, dando paso al surgimiento de un nuevo rol Rol_{III-2} , el cual contiene todas las obligaciones de los roles Rol_{I-2} y Rol_{II-1} . Con las obligaciones y roles unificados solo resta realizar las siguientes uniones para generar la organización resultante Org_{III} .

$$\begin{aligned}W_{III} &= W_I \cup W_{II} \\ I_{III} &= I_I \cup I_{II} \\ M_{III} &= M_I \cup M_{II}\end{aligned}$$

La composición de las organizaciones (denotada por el símbolo \diamond) tiene las siguientes propiedades:

Idempotencia:

$$Org_1 \diamond Org_1 = Org_1$$

Conmutativa:

$$Org_1 \diamond Org_2 = Org_2 \diamond Org_1$$

Asociativa:

$$(Org_1 \diamond Org_2) \diamond Org_3 = Org_1 \diamond (Org_2 \diamond Org_3)$$

Distributiva:

$$Org_1 \diamond (Org_2 \diamond Org_3) = (Org_1 \diamond Org_2) \diamond (Org_1 \diamond Org_3)$$

La demostración de estas propiedades se puede obtener fácilmente, considerando la relación que tiene la composición de las organizaciones con la unión de conjuntos.

V. EJEMPLO DE FUNCIONAMIENTO

En esta sección mostraremos la aplicación de nuestra técnica de modelado en la composición de protocolos, para aumentar su funcionalidad y reducir tiempo de diseño de los sistemas. Se presentara el modelado parcial de 3 organizaciones; el objetivo es generar un protocolo de interacción que pueda ser utilizado para controlar un servicio organizador de viajes. Las tres organizaciones participantes son: Hotel (H), Aerolínea (A) y Agencia de Viajes (V). La organización hotel ofrece un protocolo que cuenta con 2 roles, el cliente y el administrador del hotel, por su parte la Aerolínea cuenta igualmente con 2 roles, de los cuales uno representa el cliente y el otro representa el ejecutivo de ventas, por último la agencia de viajes contiene 2 roles, cliente y organizador. Ahora definimos el contexto de interacción de cada una de las organizaciones:

ORGANIZACIÓN AGENCIA DE VIAJES (Org_V)

$$\begin{aligned}R_V &= \{\text{Cliente, Organizador}\} \\ W_V &= \{\text{ReservarVuelo}\downarrow\uparrow, \text{ReservarHotel}\downarrow\uparrow, \\ &\text{EntregarBoletos}\uparrow\downarrow^*, \text{ListarDestinos}\downarrow\uparrow^*, \\ &\text{DesplegarItinerarios}\downarrow\uparrow^*, \text{Pagar}\downarrow\uparrow^*, \\ &\text{AnunciarPromocionesViajes}\square^*, \dots\}\end{aligned}$$

ORGANIZACIÓN HOTEL (Org_H)

$$\begin{aligned}R_H &= \{\text{Cliente, Administrador}\} \\ W_H &= \{\text{ReservarCuarto}\downarrow\uparrow^*, \text{ConfirmarReservación}\downarrow\uparrow^*, \\ &\text{CancelarReservación}\downarrow\uparrow^*, \dots\}\end{aligned}$$

ORGANIZACIÓN AEROLÍNEA (Org_A)

$$\begin{aligned}R_A &= \{\text{Cliente, EjecutivoVentas}\} \\ W_A &= \{\text{ReservaVuelo}\downarrow\uparrow^*, \text{ReservarAsiento}\downarrow\uparrow^*, \\ &\text{ConsultaTarifas}\downarrow\uparrow^*, \text{ConsultaVuelos}\downarrow\uparrow^*, \text{EnviarBoletos}\downarrow\uparrow^*, \\ &\dots\}\end{aligned}$$

Con la definición del contexto de interacción de las organizaciones, se puede obtener una descripción de las funciones que brinda una organización, además de contar con las características de cada una de las obligaciones, por ejemplo en la organización **aerolínea**, podemos observar la obligación **ReservarVuelo** $\downarrow\uparrow^*$, de la cual podemos deducir que su protocolo cuenta con la estructura suficiente para inducir y aceptar la obligación, además de especificar que la obligación es liberada dentro de la organización, caso contrario en la

organización de **agencias de viajes**, la cual también cuenta con la obligación **ReservarVuelo**↓↑, pero sin manera de liberarla, sin embargo acepta la obligación. Es ésta dependencia la cual indica el punto de unión con otra organización, que contenga el protocolo para liberar esa obligación.

La composición de las 3 organizaciones es denotada de la siguiente manera:

$$\text{Org}_{VAH} = (\text{Org}_V \diamond \text{Org}_A) \diamond \text{Org}_H$$

Las obligaciones interfaz para la composición $\text{Org}_V \diamond \text{Org}_A$ son:

Org_V .ReservarVuelo↓↑ y Org_A .ReservaVuelo↓↑*

Posteriormente obtenemos $\text{Org}_{VA} \diamond \text{Org}_H$ con las siguientes obligaciones:

Org_{VA} .ReservarHotel↓↑ y Org_H .ReservarCuarto↓↑*

Para determinar cuales son los roles a ser unificados, basta con verificar los contextos de interacción de cada uno de los roles que componen las organizaciones, para encontrar el rol que contienen específicamente la obligación interfaz marcada con “↑”, indicando ser el rol que induce la obligación y por tanto es el rol que será unificado, con el rol que promovió la composición de las organizaciones, para nuestro ejemplo el rol **Organizador** de la organización Org_V es unificado con el rol **Cliente** de la organización Org_A , posteriormente es unificado con el rol **Cliente** de la organización Org_H , estos tres roles, conforman un nuevo rol denominado R_{OCC} , el cual forma parte de la nueva organización, la cual queda definida de la siguiente manera:

$R_{VAH} = \{\text{Cliente}, R_{OCC}, \text{EjecutivoVentas}, \text{Administrador}\}$

$V_{VHA} = \{\text{ReservarVuelo}↓↑^*, \text{ReservarHotel}↓↑^*,$
 $\text{EntregarBoletos}↑↓^*, \text{ListarDestinos}↓↑^*,$
 $\text{DesplegarItinerarios}↓↑^*, \text{Pagar}↓↑^*,$
 $\text{AnunciarPromocionesViajes}□^*, \text{ConfirmarReservación}↓↑^*,$
 $\text{CancelarReservación}↓↑^*, \text{ReservarAsiento}↓↑^*,$
 $\text{ConsultaTarifas}↓↑^*, \text{ConsultaVuelos}↓↑^*, \text{EnviarBoletos}↓↑^*$
 $\dots \}$

El resultado final es una organización que engloba todos los servicios de las 3 organizaciones que la conforman. Manteniendo sus propiedades de modularidad y características descriptivas.

VI. CONCLUSIONES Y TRABAJO FUTURO

En éste trabajo se presentó un mecanismo de composición de protocolos, añadiendo un nivel semántico generado por la definición de la interacción en términos de obligaciones, las cuales proveen una interacción orientada a metas en vez de secuencias de mensajes. Además de controlar la ejecución del protocolo, las obligaciones proveen una descripción de las funcionalidades del protocolo y definen las interfaces para interconectar diferentes módulos de interacción denominados organizaciones. Las contribuciones presentadas en este artículo, permiten la reducción del tiempo de diseño de los protocolos, además de aumentar su reusabilidad y modularidad.

El trabajo futuro esta dirigido a desarrollar una metodología de representación del conocimiento, en la cual se puedan representar las obligaciones, de manera que un agente pueda analizar y seleccionar las obligaciones a unificar, automatizando el proceso de composición.

VII. REFERENCIAS

- [1] B. Bauer, J. Muller, J. Odell. “*Agent uml: A formalism for specifying multiagent interaction*”, 2001.
- [2] R. S. Cost, Y. Chen, T. Finin, Y. K. Labrou, Y. Peng. “*Modeling agent conversations with colored petri nets*”, in Working notes of the Autonomous Agents ’99 Workshop on Specifying and Implementing Conversation Policies, Seattle, Washington, Mayo 1999.
- [3] R. Fernandez, U. Endriss. “*Abstract models for dialogue protocols*”, J. of Logic, Lang. and Inf., 16(2):121–140, 2007.

- [4] J. O. Gutiérrez-García, J.-L. Koning, F. F. Ramos-Corchado. “*From obligations to Organizational Structures in Multi-Agent Systems*”, por aparecer en: The 11th Pacific RIM International Conference on Multi-Agents. Hanoi, Vietnam, Diciembre 2008.
- [5] M.-P. Huget, J.-L. Koning. “*Engineering interaction protocols for multiagent systems*”. En Cornelius T. Leondes, editor, *Intelligent Knowledge-Based Systems: Business and Technology in the New Millennium*, vol 4, capítulo 9. Kluwer Academic Publishers, ISBN: 1-4020-7746-7, 2004.
- [6] J. L. Koning, M. P. Huget. “*A component-based approach for modeling interaction protocols*”, en H. Kangassalo and E. Kawaguchi, editors, 10th European-Japanese Conference on Information Modeling and Knowledge Bases, *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2000.
- [7] M.P. Shanahan. “*The event calculus explained*”, en *Artificial Intelligence Today*, ed. M. J. Wooldridge and M. Veloso, Springer Lecture Notes in Artificial Intelligence no. 1600, Springer, pp. 409-430, 1999.
- [8] B. Vitteau, M.-P. Huget. “*Modularity in interaction protocols*”, en *Workshop on Agent Communication Languages*, pp. 291-309, 2003.

MODELADO DECLARATIVO DE AMBIENTES VIRTUALES BASADO EN EXPLOTACIÓN DEL CONOCIMIENTO.

Jaime Alberto Zaragoza Rios¹, Hector Rafael Orozco¹, Veronique Gaildrat²

Centro de Investigación y de Estudios Avanzados del Politécnico Nacional¹
Unidad Guadalajara, Institute de Recherche en Informatique de Toulouse²

RESUMEN

El modelado de escenarios virtuales, su animación y la interacción con tales modelos son un tema complicado para el usuario no experto. En el presente artículo proponemos una metodología que permite al usuario no experto generar e interactuar con tales modelos, mediante el uso del modelado declarativo y la incursión de la explotación de bases conocimiento en ésta metodología, específicamente, para la resolución de conflictos de carácter geométrico.

Palabras clave: Modelado declarativo, ambientes virtuales, bases de conocimiento, mundos virtuales, animación por computadora.

I INTRODUCCIÓN

La creación y animación de ambientes tridimensionales complejos es una tarea complicada y que requiere de conocimientos y herramientas de complejidad notoria. Los mundos presentados en video juegos y películas toman normalmente varios meses de trabajo y requieren de un personal capacitado para el manejo de las herramientas, así como el talento necesario para la creación de los mundos y personajes.

Nuestro objetivo es el estudio de una metodología que permita la creación de una herramienta de fácil aprendizaje y uso, que permita al usuario común la creación, modificación y animación de ambientes virtuales complejos, a través de una visualización tridimensional y por medio de diversas interfaces.

Con el fin de proveer al usuario con un método sencillo y transparente para la generación del ambiente, se presenta un modelador declarativo, basado en una base de conocimientos, la cual

contiene toda la información necesaria para la creación y validación del modelo solicitado por el usuario. El modelado declarativo es una técnica que permite al usuario la descripción de un escenario, diseñándolo de manera intuitiva, al dar solo algunas propiedades esperadas y dejando al sistema de cómputo encontrar una solución, si existe alguna, que satisfaga tales restricciones [1]. Siendo la base para la creación de tales mundos virtuales una base de conocimientos, ésta también contiene la información necesaria para la validación semántica, lógica y geométrica de las restricciones presentadas por el usuario.

La base para la validación geométrica, el tema principal de este artículo, es el uso de algoritmos para la solución de problemas de satisfacción de restricciones, o CSP, los cuales nos permiten encontrar conflictos tanto de posicionamiento como colisiones.

Este trabajo forma parte del proyecto GeDA-3D [2] una plataforma genérica distribuida para la creación y manipulación de mundos virtuales en un ambiente en 3D. También conforma el modelador declarativo del proyecto DRAMA, en desarrollo en el Institute de Recherche en Informatique de Toulouse (IRIT), en Toulouse, Francia.

II TRABAJOS EN MODELADO DECLARATIVO

Existen diversos trabajos enfocados en el modelado declarativo, algunos enfocados completamente en aspectos arquitectónicos, otros dirigidos a la generación de escenarios virtuales, y otros como herramientas de esbozo o diseño.

Entre las herramientas dedicadas al desarrollo arquitectónico podemos destacar las siguientes: El sistema FL, por Jean-Eudes Marvie y otros [3], un

trabajo basado en Sistemas Lindenmayer, pero enfocado completamente en la generación de modelos complejos de ciudades. La entrada al sistema está definida en una gramática especializada, y puede generar modelos de ciudad de complejidad variable. El sistema se basa completamente en una variante de un Sistema-L, utilizando VRML97 para generar la visualización del modelo. CityEngine [4], por Yoav I H Parish y Pascal Müller, es un proyecto enfocado completamente al modelado de ciudades completas, recibiendo como entrada datos estadísticos e información geográfica. También basa su mecanismo de diseño en un Sistema-L, utilizando una gramática especializada para este fin. Sin embargo, nuestro punto de interés son aquellos trabajos enfocados en la creación de escenarios virtuales, donde destacan los siguientes trabajos:

WordsEye, un sistema automático de conversión texto-a-escena, desarrollado por Bob Coyne y Richard Asproad en los laboratorios AT&T. Permite al usuario generar un escenario en 3D, a partir de una descripción en un lenguaje natural. Utiliza marcado de texto y analizadores de parte del discurso y estadístico. La representación gráfica se genera a partir de representadores (especificación gráfica de bajo nivel) asignado a cada elemento semántico, modificados para coincidir con las poses y acciones descritas en el texto, por medio de cinética inversa. [5]. Utiliza algunas técnicas, como textualización, emblematización, caracterización, literalización o personificación en casos donde no existe un representador propiamente establecido.

DEM²ONS, un modelador declarativo de alto nivel para aplicaciones de gráficos en 3D diseñado por Ghassan Kwaiter, Véronique Gaildrat y René Caubet. Permite al usuario construir escenas en 3D de manera natural, con un alto nivel de abstracción. Está compuesto de dos partes: Interfaz modal y modelador de escenario en 3D [6]. La interfaz modal permite la comunicarse con el sistema, usando simultáneamente múltiples métodos de entrada (guantes de datos, sistema de

reconocimiento del habla, spaceball, ratón). El modelador de escenario utiliza ORANOS, un resolvidor de restricciones con varias características que le permiten expandir el rango de aplicaciones de modelado declarativo. Los objetos son modelados y representados por el Conjunto de Herramientas Inventor, que provee la Interfaz de Usuario Gráfica. Este sistema resuelve cualquier problema de restricción, pero solo permite objetos estáticos, sin soporte para avatares.

Multiformes es un modelador declarativo de propósito general especialmente desarrollado para esbozar escenas en 3D, presentado por William Ruchaud y Dimitri Plemenos. El trabajo en un escenario con MultiFormes es manejado esencialmente a través de su descripción, esto es, la forma en que el diseñador introduce todas las características de los elementos geométricos de un escenario y las relaciones entre ellos [7]. La característica más importante de Multiformes es su habilidad para explorar automáticamente todas las posibles variaciones en de un escenario, pues no fuerza una sola interpretación de cada propiedad imprecisa. La descripción de un escenario incluye dos conjuntos: el conjunto de objetos geométricos presentes en ese escenario y el conjunto de relaciones existentes entre los objetos geométricos. Gracias a su resolvidor de restricciones, Multiformes es capaz de explorar diversas variaciones de un esbozo, satisfaciendo la misma descripción, siendo el resolvidor de restricciones geométricas el corazón del sistema. Este sistema obtiene sus soluciones de manera incremental, y es capaz de resolver las restricciones solicitadas por el usuario, pero éste debe de indicar al sistema como se debe de construir el escenario.

CAPS es un sistema de posicionamiento automático basado en restricciones [8], desarrollado por Ken Xu, James Stewart y Eugene Fiume. Hace posible el modelado de escenarios grandes y complejos, utilizando un conjunto de restricciones de posicionamiento intuitivas que permiten la manipulación de múltiples objetos simultáneamente, mientras se utilizan pseudo-físicas para asegurar que el posicionamiento sea físicamente estable. Utiliza métodos de entrada con

alto nivel de grado de libertad, como el SpaceBall o el DataGlove. También emplea técnicas semánticas para el posicionamiento de los objetos, utilizando conceptos como fragilidad, utilidad o interacción con otros objetos. La disposición de los objetos se hace uno a la vez. Permite también la interacción directa con los objetos, manteniendo las relaciones entre ellos, por medio de seduofísicas o agrupamientos. Los métodos y herramientas integrados a este sistema lo hacen una herramienta de diseño, orientado principalmente a la visualización de un escenario, sin la posibilidad de que éste evolucione por su cuenta.

. III GENERIC DISTRIBUTED ARCHITECTURE 3D

GeDA-3D [2] es un proyecto actualmente en desarrollo, y del cual se desprende la investigación presentada en este artículo. GeDA-3D es una plataforma para crear, diseñar y ejecutar escenas dinámicas virtuales en 3D en una infraestructura distribuida, basada en el paradigma de agentes móviles y está formada por varios módulos: Editor de Ambientes Virtuales (VEE), Visualización, Núcleo de GeDA-3D, Comunidad de Agente (AC) y Descriptor del Contexto, todos representados en la Figura 1. Los componentes principales de GeDA-3D son el control de escena y el control de agentes.

EL VEE incluye el descriptor de escenarios, analizador de congruencia, resolvidor de restricciones y editor de escena. Provee una interfaz entre la plataforma y el usuario, especifica las leyes físicas que gobiernan un ambiente y describe la escena virtual que se lleva a cabo en tal ambiente.

La visualización se encarga de todo los detalles relacionados a los gráficos en 3D, principalmente, representar la evolución en 3D de la escena.

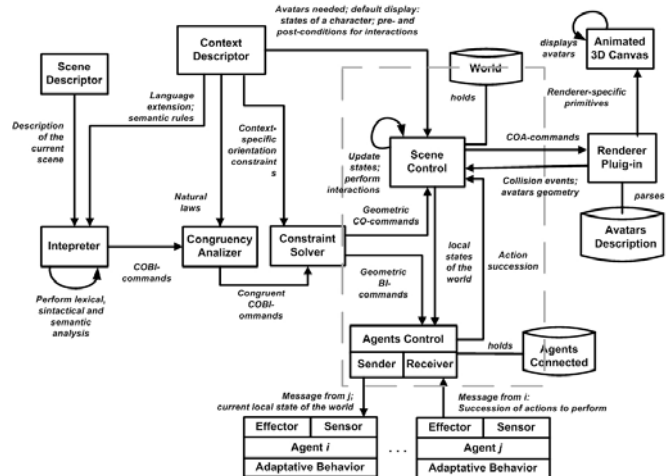
La comunidad de agentes (AC) representa los agentes encargados de dominar el comportamiento de los objetos virtuales y los avatares.

La descripción de escena proporciona a un agente una detallada descripción acerca de los objetivos que el usuario desea que haga, mas no la manera en que deben ser conseguidos.

Una escena puede incluir un conjunto de objetivos para un solo agente, sin que sea necesario que tales

objetivos sean alcanzados de una manera secuencial [9].

Figura 1. Arquitectura del proyecto GeDA-3D.



. IV MODELADOR DECLARATIVO PARA AMBIENTES VIRTUALES

La creación del modelo que representa el ambiente virtual solicitado por el usuario es llevada a cabo por un modelador declarativo, parte del Editor Virtual, recibiendo como entrada una descripción escrita en un lenguaje definido explícitamente con el fin de proporcionar al usuario un método sencillo, estructurado y con expresividad suficiente para indicar los elementos a incluir en el escenario así como las propiedades tanto del ambiente como de los elementos que existirán dentro de este. A este lenguaje lo hemos llamado VEDEL, Lenguaje para Descripción de Ambientes Virtuales (Virtual Environments Description Language) [10,11].

En una descripción en VEDEL se pueden identificar 3 secciones o párrafos, delimitados por etiquetas de sección, como se muestra en la figura 2. Cada sección consiste de varias oraciones, las cuales están formadas por declaraciones separadas por comas, y deben estar finalizadas por un punto. La declaración inicial debe ser el tipo de entidad a representar (ambiente, actor u objeto), seguido de manera opcional de un identificador único, en el caso de actores u objetos. El resto de las declaraciones corresponden a las propiedades deseadas para la entidad y los valores a ser asignados a estas. Tales propiedades pueden incluir posicionamiento, tamaño, propiedades físicas (color, forma, transparencia) o internas (energía,

estado emocional). Cada concepto expresado en la descripción es verificado en la base de conocimientos, con la excepción del identificador individual, de manera que aquellos conceptos no incluidos dentro de la misma serán omitidos o presentados como errores al usuario, según el método de funcionamiento del modelador.

```
[ENV]
  Desert, cloudy, night.
[/ENV]
[ACTOR]
  Knight Arthur, tall, wherever.
  YoungWoman Betty, left Arthur.
[/ACTOR]
[OBJECT]
  House, center.
  Chair, front house, color red, facing House.
[/OBJECT]
```

Figura 2. Ejemplo de descripción escrita en VEDEL.

La descripción es enviada a un analizador sintáctico, el cual analiza y clasifica la entrada para el modelador. El analizador es una máquina de estados, que busca errores de tipo léxico y sintáctico, los señala y resuelve, además envía una estructura de datos organizada de manera jerárquica al modelador. El modelador utiliza esta estructura para generar el modelo del ambiente virtual, utilizando además una base de conocimientos para realizar la verificación semántica de los conceptos solicitados por el usuario. El modelador extrae la información necesaria de la base de conocimientos para su correcta representación en el modelo, primero obteniendo las propiedades internas y visuales de cada entidad, y posteriormente con las propiedades geométricas: tamaño, posición y orientación. Para este proyecto, la base de conocimientos esta formada por 4 clases base: Environment, que incluye todos la ambientes *base*, Actors, donde se almacena la información pertinente a los actores, Objects, que incluye los datos para las entidades tipo objeto, y Keywords, que son todas aquellas palabras que indican propiedades de cualquier

elemento del escenario. Así mismo, se establecen diversas propiedades de tipo objeto y dato, desde relaciones “*Contrario A*” hasta conceptos como “*Color*” o “*Tamaño*”. Estas propiedades se asignan con los valores *por defecto* para la entidad o concepto en al menos un individuo por cada clase. Este individuo debe llevar el nombre de la clase y terminar con el sufijo “_default”. Si se desea establecer varios individuos, esto es, varios conceptos cuyo morfología básica sea similar, dentro de una misma clase, estos pueden llevar el nombre propio de la entidad o concepto, pero siempre terminados con el sufijo “_default”.

El caso del medio ambiente solicitado, el conocimiento almacenado en la ontología es utilizado para establecer el contexto que la arquitectura subyacente utilizara para desarrollar la escena, incluyendo las leyes que gobernarán sobre el ambiente y las entidades, y las instrucciones que permitirán a los agentes el desarrollo de las entidades que tengan asignadas, tales como las acciones que pueden ejecutar o sufrir, los parámetros que regirán su evolución, o las emociones que puedan representar.

Para las entidades, la información es extraída de los diversos individuos, y las propiedades internas y externas son interpretadas de acuerdo al conocimiento almacenado en la ontología, esto es, conceptos como “color”, “cabello”, “energía” o “tristeza” son transformados en valores numéricos o estructuras de datos interpretables para el resto de la arquitectura, a través de una función de inferencia que revisa la base de conocimientos para validar, convertir y, de ser necesario, inferir la información necesaria de otros apartados dentro de la misma base.

Una vez que completada la primera fase, el modelador prosigue con el posicionamiento de las entidades dentro del escenario, siempre guiándose por las solicitudes hechas por el usuario. Esta fase es auxiliada por un algoritmo CSP [12], el cual se encarga de verificar y validar que la posición asignada a una entidad sea válida con respecto a la descripción y a la propiedades intrínsecas de la entidad y las entidades a su alrededor.

Nuestro CSP está formado por la tupla siguiente:

- Un conjunto finito V de x de variables, en nuestro caso, las entidades del escenario.

- Un conjunto D de valores posibles para cada variable en V , como se muestra en la figura 3.
- Un conjunto C de restricciones, el cual está compuesto por las ecuaciones listadas en la figura 3., ecuaciones (1.a) a (1.c).

Por tanto, una solución válida a una descripción cualquiera es una asignación de valores de los dominios de D a las variables en V , tales que ninguna restricción en el dominio C sea violada [13].

Las ecuaciones seleccionadas para validar posicionamiento y resolver conflictos espaciales y colisiones fueron seleccionadas debido a que pueden ser resueltas en lapso corto, y que sus parámetros permiten moldear el volumen que representan a la forma de las entidades, de manera que al utilizar varios volúmenes se cubra toda la entidad, sin que esto impacte en la complejidad computacional o la velocidad de cálculo en el proceso.

Cada entidad y concepto tiene asignada una o varias funciones en su entrada en la base de conocimientos, llamadas etiquetas, las cuales modifican de acuerdo con los valores asignados a la entidad, y posteriormente validados contra el resto de las entidades. Adicionalmente, se asigna una serie de etiquetas para indicar los puntos característicos de la entidad, mismas que son utilizados para validar la posición de un entidad con respecto a otra (Figura 4).

Conjunto $V = \{x_1, \dots, x_n\}$ de variables donde $x = \{P, O, S\}$ tal que

- $P = \{x, y, z\}$ (posición de la entidad).
- $O = \{\theta_x, \theta_y, \theta_z\}$ (orientación de la entidad).
- $S = \{S_x, S_y, S_z\}$ (escala de la entidad).

Conjunto D para el conjunto V tal que

- $D(P) = \mathbb{R}^3$
- $D(O) = [0, 2\pi], \forall \theta \in O$
- $D(S) = \mathbb{R}^3$.

Conjunto C de restricciones, formado por las siguientes funciones:

$$(1.a) \quad (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 - (r_1 + r_2) = 0$$

$$(1.b) \quad \left(\frac{x}{p}\right)^2 + \left(\frac{y}{q}\right)^2 - 2z = 0$$

$$(1.c) \quad \left(\frac{x}{dx}\right)^2 + \left(\frac{y}{dy}\right)^2 + \left(\frac{z}{dz}\right)^2 - 1 = 0$$

Figura 3. Definición del CSP para el modelador.

La descripción ya analizada es revisada para encontrar dependencias entre las entidades, y a partir de éste análisis, se genera una lista que el modelador utiliza para asignar los valores iniciales a cada entidad, haciendo uso de la función de inferencia para obtener los valores y funciones asignados a cada concepto. Aquí se definen las entidades *pivote*, es decir, aquellas cuyo posición no depende de otras.

El modelo es revisado para verificar que no existan colisiones entre las entidades. Si estas ocurren, se procede a la modificación de los valores de las entidades en conflicto. Una vez realizados los cambios en el posicionamiento, se procede con la verificación de las restricciones, de tal forma que las nuevas posiciones se validen, o se computen nuevas posiciones si la entidad entra en un conflicto de posicionamiento. El proceso se repite hasta encontrar una solución que satisface las restricciones, o un punto cerrado, esto es, una solución que no cumple las restricciones y que no puede ser modificado para cumplirlas. En el caso de un punto cerrado, se procede a la modificación de los valores de las entidades que fungen como pivotes. Si el proceso no encuentra solución alguna después de estas modificaciones, se procede con un informe de error al usuario, o se excluyen las entidades en conflicto, según la metodología seleccionada para el modelado.

El modelo final es enviado a una función que se encarga de generar las salidas necesarias para que la arquitectura pueda comenzar con la representación del escenario, incluyendo la representación visual y el contexto para el núcleo y los agentes. Esta función utiliza un método Controlador Vista-Modelo, de forma que la salida pueda ser modificada sin necesidad de modificar el código de la aplicación.

V. CONCLUSIONES Y RESULTADOS

El prototipo más reciente de nuestro modelador ha sido modificado para cumplir con los requerimientos de integración con el proyecto DRAMA [14], actualmente en desarrollo en el IRIT, como una colaboración con el equipo de Sistemas Distribuidos del CINVESTAV, Guadalajara. Fue codificado en lenguaje Java,

utilizando el más reciente SDK, versión 6, así como la API del proyecto Protégé, la cual permite el acceso a la base de conocimientos. Finalmente, el Modelo Vista-Controlador es manejado por la clase FreeMarker, lo cual permite modificar la apariencia de las entidades sin acceder al código fuente de la aplicación.

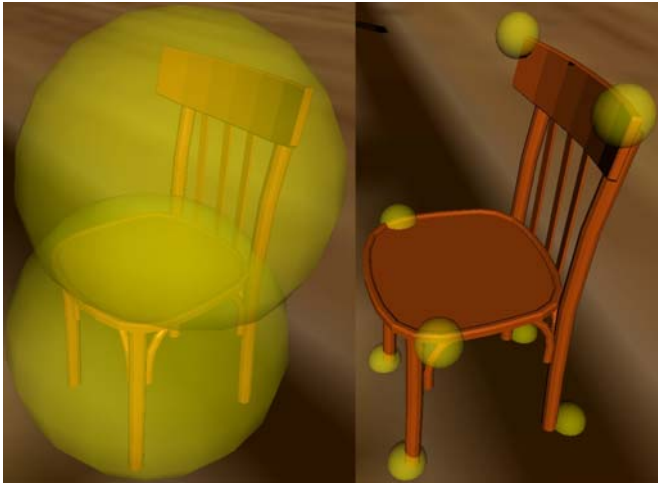


Figura 4. Ejemplos de etiquetas de colisión y de puntos característicos.

La figuras 5, 6 y 7 muestran algunos ejemplos de los escenarios obtenidos a través del modelador, utilizando estándar 3XD para la visualización en 3D. La cantidad de ambientes posibles se puede expandir al ingresar nuevos conceptos a la base de conocimientos, asignando también los modelos que representaran tales conceptos en el caso de entidades físicas. Los conceptos intrínsecos o abstractos son manejados por la arquitectura subyacente, siendo necesario que ésta pueda representarlos para que tomen efecto dentro del ambiente virtual.

A la fecha de redacción del presente artículo, resta conducir la integración del modelador tanto con el proyecto DRAMA como con el resto de la arquitectura GeDA-3D. Así mismo, es necesario el diseño de un verificador de congruencia semántica completo, que permita encontrar inconsistencias dentro del contexto de la descripción. Finalmente, se trabaja en la generación del contexto, encargado de indicar a la arquitectura las reglas que deben regir sobre el ambiente virtual, las propiedades y

acciones permitidas dentro del mismo, y las acciones a llevar a cabo para cada suceso posible dentro del contexto del ambiente, en concordancia con los requerimientos del usuario.



Figura 5. Composición simple.

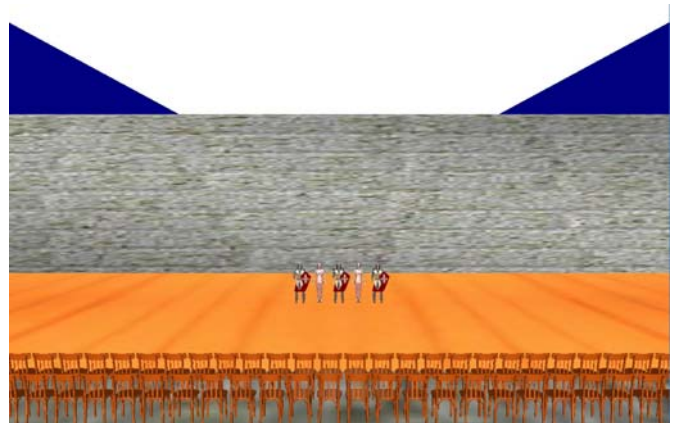


Figura 6. Ejemplo de ambiente "Teatro".



Figura 7. Composición simple en ambiente "Bosque".

. VI REFERENCIAS

- [1] Demitri Plemenos, Georges Miaoulis, and Nikos Vassilas. Machine learning for a general purpose declarative scene modeller. In *International Conference GraphiCon '2002*, Nizhny Novgorod (Russia), September 15-21, 2002.
- [2] Felix Ramos, Fabiel Zúñiga, and Hugo I. Piza. A 3D-space platform for distributed applications management, International Symposium and School on Advanced Distributed Systems 2002. Guadalajara, Jal., México, November 2002.
- [3] Marvie, Jean-Eudes and Perret, Julien and Bouatouch Kadi. The FL-system: a functional L-system for procedural geometric modeling, *The Visual Computer* 5-21, pages 329 -339, June 2005.
- [4] Parish, Y. I. and Müller, P. 2001. Procedural modeling of cities. In *Proceedings of the 28th Annual Conference on Computer Graphics and interactive Techniques SIGGRAPH '01*. ACM, New York, NY, 301-308.
- [5] Bob Coyne and Richard Sproat. Wordseye: An automatic text-to-scene conversion system. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 487-496. AT&T Labs Research, 2001.
- [6] G. Kwaiter, V. Gaildrat, and R. Caubet. Dem2ons: A high level declarative modeler for 3D graphics applications. In *Proceedings of the International Conference on Imaging Science Systems and Technology, CISST'97*, pages 149-154, 1997.
- [7] William Ruchaud and Dimitri Plemeno. Multiformes: A declarative modeller as a 3D scene sketching tool. In *ICCVG*, 2002.
- [8] Ken Xu and James Stewart and Eugene Fiume. Constraint-Based Automatic Placement for Scene Composition, Proc. Graphics Interface, May 2002, Calgary, Alberta
- [9] Fabiel Zúñiga. Agent Convenable Behavior in Dynamic Virtual Environments. PhD thesis, CINVESTAV GDL, México, 2007.
- [10] Jaime Alberto Zaragoza Rios. Representation and exploitation of knowledge for the description phase in declarative modeling of virtual environments. Master's thesis, Centro de Investigación y de Estudios Avanzados del IPN, Unidad Guadalajara, 2006.
- [11] Jaime Zaragoza, Félix Ramos, Héctor Rafael Orozco, Véronique Gaildrat. Creation of Virtual Environments Through Knowledge-Aid Declarative Modeling. Dans : *Congress of Logic Applied to Technology (LAPTEC 2007)*, UNISANTA - Santa Cecília University, Santos, Brésil, 21/11/2007-23/11/2007, IOS Press, p. 1-8, novembre 2007.
- [12] Daniel Hunter Frost. Algorithms and heuristics for constraint satisfaction problems. PhD thesis, University of California, 1997. Chair-Rina Dechter.
- [13] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.
- [14] Andriamarozakaniaina T, Pouget M, Zaragoza R, Gaildrat V. DRAMAtexte : indexation et base de connaissances. *Premier colloque international sur la notation informatique du personnage. 16-17 mai 2008*. A paraître.

GENERATION OF BEHAVIOR'S AGENTS USING EVOLUTIVE SYSTEMS AND ANTICIPATION.

Marco A. Ramos.

Alain Berro **

UAEM TOLUCA.
marco.corchado@gmail.com

IRIT/UT1, Francia.
berro@univ-tlse1.fr

Cerro de Coatepec S/N Universidad Autónoma del Estado de México Toluca.

**Rue du Doyen-Gabriel-Marty, Toulouse, France

Abstract

This article introduce an approach to behavior of agents based on adaptive learning. The approach is based on the concept of anticipation phenomenon of the human brain. In complex systems, the agents need to take their decisions based in the state of world. The concept of anticipation can be useful for real time for improving and facilitating decisions. The work in this paper deals with anticipative agents, and are developed situations, actions, and changes in the world. These situations are useful for an agent to compare previous situations, actions and results, and decide which action could lead to a situation with the best utility or satisfaction degree. Also is proposed a model of an multiagent system for evolving agents in virtual environment based on classifiers systems and evolutionary methods.

Keywords: Agents, anticipation, virtual reality, evolution.

I. INTRODUCTION

Anticipation is the capacity for creating predictive models thanks to learning process. The anticipatory behaviour considers the actions taken in the environment [9] in fact the anticipations are predictions of future values that modify the environment and the current behaviour of the agent. According to the classical work of Rosen [14], an "anticipatory system is a system containing a predictive model of itself and/or its environment which allows it to change its state at an instant in accord with the model's predictions pertaining to a latter instant (p. 339)". The

approach is centred in using the predictions for represent the anticipation and its consequences to choose apply an action in the environment. A learning anticipatory system, learns a predictive model of an encountered environment specifying the consequences of each possible action in each possible situation [2]. Many approaches has been used for the study of anticipation behaviour like the use of neural network, fuzzy logic etc., also exist some good review that present the effort to use the anticipation in complex systems [4, 15].

II. COGNITIVE MEMORY

The production systems, was the first cognitive architectures based in the human reasoning (Anderson, 1983), introducing the terms like short term memory and long term memory. The skill is recognised the patterns complex for the stimuli of perceptions thanks to sensory. Sensory information is stored in the human brain for neural net, so the information is preserved in different cell the problem here is that this information is preserved for a short time. It's the time of stimuli that produced that the cells in the human brain can store the information long time. We can classifier two types of memories, like visual information and sound information. These two types of memories work together for store the information and generally called sensory memory [1]. In fact the human memory is classified in three modules: episodic memory saves facts on their happening, semantic memory this memory construct knowledge in structural form, and procedural memory help to take decisions for actions to do. Sensory information "signals" like sound, view,

touch, smell is received by receptors for storage in sensory register (memory) this storage is temporary, only if the signal is continued and strong the information is recorded in long term, figure 1.

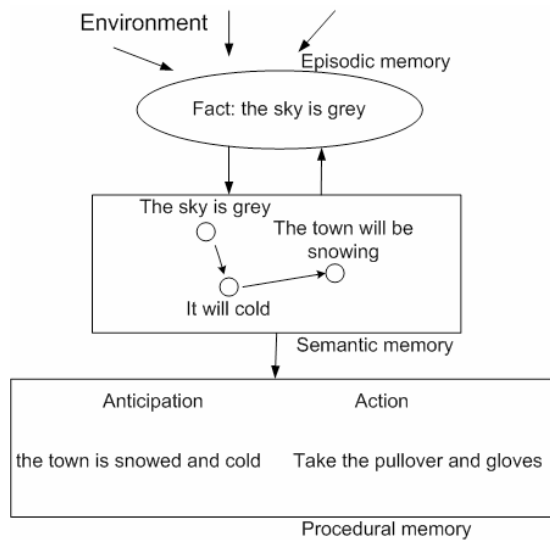


Figure 1: Architecture of Cognitive Memory.

Episodic memory stores information about happened events and their relationship. The semantic memory represents knowledge and does not change frequently and the procedural memory save procedures for execution of a task. Cognitive Psychology give a reference for put in evidence that this learning mechanism is the learning process in the human [16].

III. ANTICIPATION

Anticipation is based on predictive model of the system itself and its environment, we used this definition for this first work. We understand anticipation as the capacity for creating predictive models thanks to learning process. The anticipatory behaviour considers the actions taken in the environment [14]. In fact the anticipations are predictions of future values modifying the environment and the current behaviour of the agent.

THEORY OF ANTICIPATION

Anticipation is the generation of a multitude of models for creates of actions in the environment and the resolution of their conflict. The theory of anticipatory systems proposed by Hoffman [15, 14] is based in three states S-R-E where S it's the perceptions observed in the world, R is the reactions produced for the perceptions and the generations of models of decisions for change to state in the world and E is the effect produced for an change of state. It's clear that the anticipation is part important for the creation of behavior and for us the use of states passes has a principal roll in the model cognitive of human brain. This idea is that the dynamical behavior of a system is completely determined by its past. This kind of system is called a causal system.

$$w_{t+1} = f(w_0, w_2, \dots, w_t) \quad (1)$$

where, w_t , $0 \leq t \leq t+1$ is the state of the system at time t , and f represents the functional dependence between the present state and the past. The structure of these systems varies essentially with time, but, does not formalize the influence of the possible future on the present behavior of the system. An anticipatory system is thus a system which uses knowledge of future states to decide what action to take in the present.

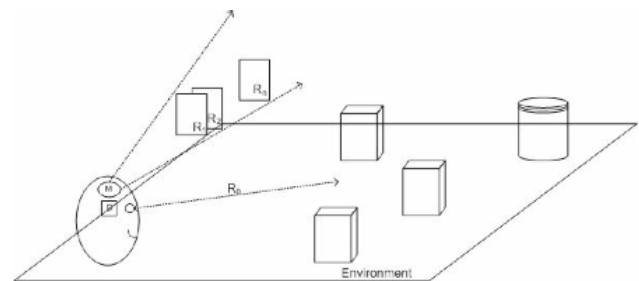


Figure 2: Anticipatory agent model: R0 is the perception of reality environment that the agent interacts with. D is an internal cognition of the agent's environment R0 and of the agent itself, while M is an internal interpretation of this description made by the agent in order to foresee the future. M gives rise to a number of comprehended realities, Ris. The variations among these are partly caused by different possible events in the environment, and partly caused by interactions between the environment and the agent itself.

This is an idea clearly stated for the agent [5]. That is, we can consider agents as anticipatory agents. That is, anticipatory agents will act according to the past states of the system and according to the desirable and possible future states. Formally,

$$w_{t+1} = f(w_0, w_2, \dots, w_t, \hat{w}_{t+1}^t, \dots, \hat{w}_{t+k(t)}^t) \quad (2)$$

where $\hat{w}_i^t, t+1 \leq t \leq t+k(t)$, is a prediction of w_i done at time t , and $k(t)$ is the size of the forecasting interval at time t . This formulation allows the agents dynamic behavior to be modeled in a more realistic form, gaining more explanatory power than a mechanistic conception. However, there is the problem of analytic solution determinations.

An anticipatory system uses the knowledge acquired to the past for applying them in the future next, that helped him to take better decisions than they provide appropriate solutions to the conditions of the environment see figure 2.

IV. MODEL OF ANTICIPATORY SYSTEM

Anticipatory systems can be defined and can be tested in a wide variety of artificial worlds that evolve over extended periods of time. The resulting "complex adaptive systems" can be examined both computationally and analytically, offering new ways of experimenting with and theorizing. Anticipatory system consists in some principal parts as: action module part of the system that perform the changes of the system (through the actuators), a set of sensor that perceive the environmental inputs that the cognitive module interpret for the creations of models of possible future states, when its not possible obtain a model that solve one situation in the environment its necessary a system evolutionist that let the generation of new models [10, 11] yield the possibility to obtain other parameters to satisfy the reality. The basic architecture of an anticipatory system illustrated in Figure 3. The structure of our Anticipatory System consist in diferent modules:

- A set of sensors that perceive and receive the environmental inputs.

- Cognition is a collection of pairs that joins states of the world and actions taken in these states.
- Rule population that is a collection of triples of conditions, action and strength (in the form of classifiers [10, 11, 3]) that represent the conditions of the world.
- Prediction is a value that expresses the expectation of the agent's happiness after the following action.
- Genetic Algorithm is the reinforcement learning mechanism that evolves the rule set. The fitness of the classifiers (their strength) is evaluated by comparing the current utility and the expected utility.
- Matching is the processes that select the rules that have its conditions satisfied by the state of the world now.
- Action is the processes that choose the winning rule.
- The winner acts via effectors and actualizes in memory expected utility and history of the world.

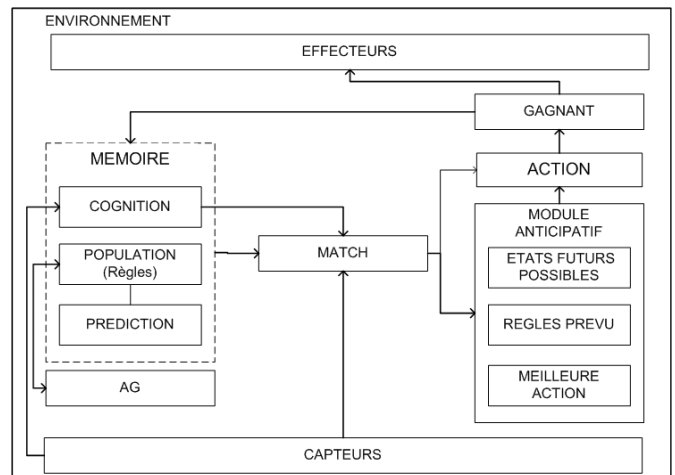


Figure 3: Architecture of Anticipatory System.

V. DESCRIPTION OF PROBLEM

The creation of emergent behavior in complex systems is very important and the anticipation is a roll important in this. Imagine that there is a shepherd-dog agent, which has the task of controlling in real time a group of agents sheep's [6] Figure 4. The dog can sense the objects that are in the environment and it can recognise which of them represent a danger to the sheep's. This

agent's task is to foresee the conditions that endanger this group of sheep-agents and to anticipate the movements that lead them to a safe state. The agent sheep's has a behavior to find the food. To perform such task an agent must possess capabilities for: perceptionacquiring and interpreting sensed data to obtain knowledge of external entities, cognition-knowledge-based reasoning to assess situations, solve problems, and determine actions, and actions actuating effectors to execute intended actions and influence external entities [7]. In a complex environment, an agent's opportunities for perception, action and cognition typically exceed its computational resources.

The design to situations ant description de actions is based to mechanism of learning system that can make the evolution of these situations represented by rules, for example the behavior for the shepherd-dog agent:

- go to back to herd
- go to left to herd
- go to right to herd
- save to danger sheep
- push the herd

The more important here is can evolving the rules for find the good learning cognitive for applied to the environment, and do emerge the behavior to shepherd-dog agent. The mechanism to make the evolution of the rules is a classifier system [11]. A classifier system is a learning system that learns syntactically simple string rules (called classifiers) to guide its performance in an arbitrary environment. A classifier system has three main components.

- Rule and message system.
- Apportionment of credit system.
- Genetic algorithm.

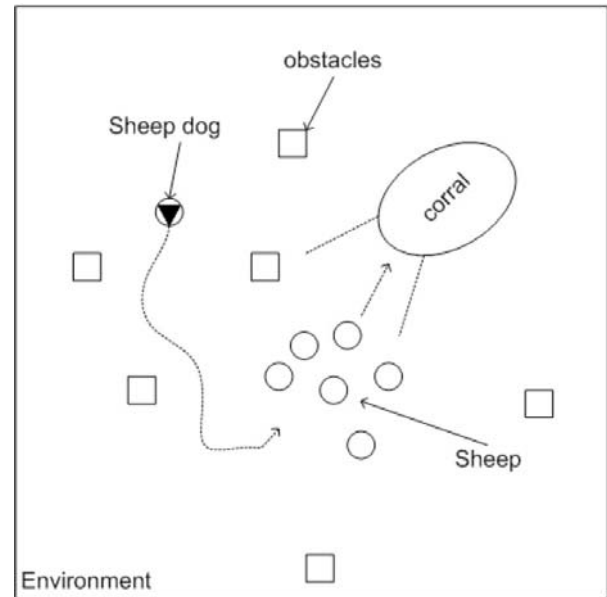


Figure 4: Sheep-dog environment.

The structure of a classifier system is a triple, (c, a, f) , figure 5., that joins a condition c “a state of the world or a situation”, an action a that is a “simple or composed action”, and a strength f . We use a ternary alphabet $(0, 1, \#)$ to represent the conditions and actions of the classifier. The symbol $\#$ has the meaning of “don't care”. For the similarity determination of conditions ”situations” and decision we consider that all elements of the alphabet have distance 0 to $\#$, [3]. For example, $\|(1, \#, 0) - (0, 1, \#)\|_3 = (1 + 0 + 0)^{\frac{1}{2}} = 1$. Apportionment of credit system.

The classifiers strength at time t , (f_t) , is adjusted in the following way:

- All rules have to pay a *LifeTax* (tax_{life}) at each iteration. That means $f_{t+1} = (1 - tax_{life})f_t$, where tax_{life} , $0 < tax_{life} < 1$, is a value near 0.
- All rules that match the present situation that is the rules that are going to the action, have furthermore to pay a *BidTax* (tax_{bid}). That is, the rules that participate in the action have its strength adjusted by $f_{t+1} = (1 - tax_{life} - tax_{bid})f_t$, where tax_{bid} , $0 < tax_{bid} < 1$, is a value near 0.

- The winner of the decision at time t has the strength computed using the difference between the expected utility,

$$\hat{U}_{t+1}, \text{ at time } t \text{ for decision } d \text{ and the real utility, } \hat{U}_{t+1}, \text{ observed by the agent at time } t + 1, \text{ that is } f_{t+1} = (1 - tax_{life} - tax_{bid})ft + u_{t+1} - \hat{U}_{t+1}.$$

is far to herd	is behind to herd	is bad way herd	is good way herd	is left to herd	is right to herd	is grouped the herd	Action	Strength
0	1	#	1	1	0	1	go to vision to herd	0.6
0	1	#	#	#	1	1	go to near to herd	0.5
1	#	1	#	1	1	1	go to side to herd	0.3
1	#	#	1	#	#	1	guide to corral	0.7
1	#	#	#	1	1	1	go to behind to herd	0.9

Figure 5: Classifier sheep-dog behavior.

VI. SOME RESULTS OF SIMULATION

The evolution of the rules is actually done by the genetic algorithm (GA)[10]. The execution of the GA occurs at certain number of iterations stochastically determined, called an epoch. In each epoch only a small part of the rule set is selected for evolution, if in the memory exist a solution or other whit a good degree of similarite this is used for aply one action in the environment, where the rules with the lower strength are the rules with greater selection probability. The rules of this small subset are substituted by new rules determined through crossover and mutation. The mutation process in the GA corresponds to random modification of some rules, randomly selected. The learning consists in building the models of the dynamics of environment, guaranty the decision and effect for the classifier. Also is very important the anticipation in the classifier for take of decision and update the cognition for futures states in the world.

We called our system Cognitive Anticipatory Classifier System CACS. This incorporates a module that saves the present conditions to the

environment that they were solved successfully and they are potential candidates to be used in the future. The first tests made, was to evaluate the systems with other systems that use classifier system to generate behavior like Extend Classifies System XCS and General Extend Classifier System GXCS. The results show the behavior of different systems, we can see in the side to stability that CACS obtain a good performance Figure 6., however XCS and GXCS have similar behavior this is a good news because show that CACS can solve better the complex problems.

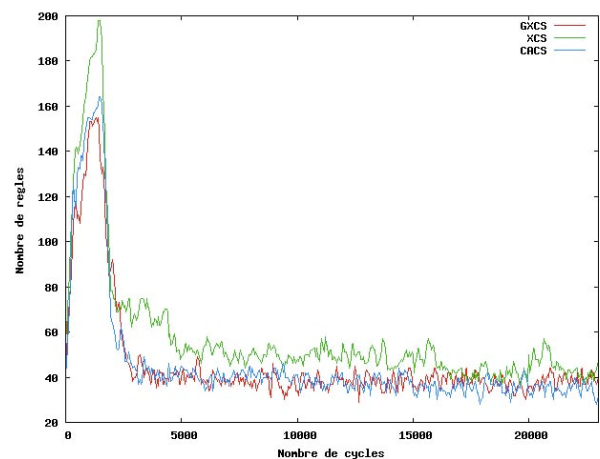


Figure 6: Evolution of knowledge of classifier.

Concerning to performance Figure 7., we can see that CACS at first time does many mistakes but its can stabilized quickly this is thanks to the cognitive module.

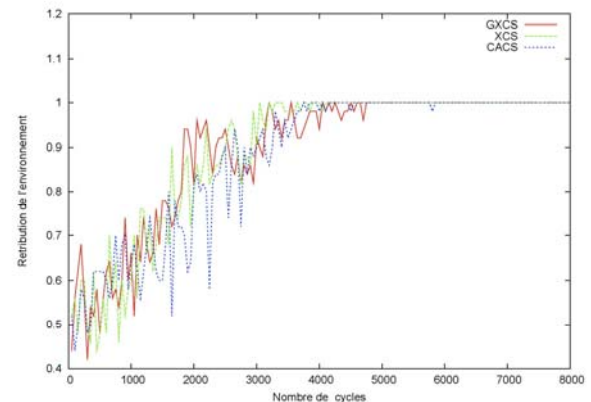


Figure 7. Performance of classifier.

VII. SUMMARY AND CONCLUSIONS

The phenomena of anticipation are very important in the real life the human and the other creatures use this phenomenon. In fact the anticipation creates the emergence of behavior in the human and animals. It's clear that anticipation it's not only the internal representation of itself and the environment, the process of cognition is part important in the phenomena of anticipation. The study of real time anticipatory systems using agents requires an architecture that is sufficiently complex to accommodate representations of itself, about the environment and the cognition. Even more the architecture must be able not just to have a model to recreate external world, but also entertain internal representations of relevant aspects of the environments in terms of the ranges of possible further developments. That is, think in function of the future. Thus anticipation must be considered as an analytical requirement in complex systems. In this article, this additional complexity was solved by using the time dimension not as a given, rather considering it as another degree of freedom available to the system. This degree of freedom allows for active adaptation to changes in the environment through anticipation by inverting the arrow of time.

References

- [1] Atkinson, R. C. and Shiffrin, R. M. (1968). *Human memory: A proposed system and its control processn in The Psychology of Learning and Motivation: Avances in Research and Theory*, Spence, K. W. and Spence, J. T., Eds., vol 2, Academic Press, New York.
- [2] Butz, M. Goldberg, D. (2003). *Generalized state values in an anticipatory learning classifier system. In M. Butzn O. Sigaud P. Gerard Anticipatory Behavior in Adative Learning Systems. LNAI 2684, Springer Verlag, p. 282-301.*
- [3] Butz, M. Sigaud, O. and Gerard P. (2003) *Anticipatory Behavior in Adative Learning Systems. LNAI 2684, Springer Verlag.*
- [4] Butz, M. (2002). *Anticipatory Learning Classifier Systems. Kluwer Academic Publishers.*
- [5] B. Ekdahl. *The concept of models in anticipatory agents. Technical Report LU-CS-TR: 94-124, Dept. of Computer Science, Lund University, Lund, Sweden, 1994. In preparation.*
- [6] Cliff, D. and Miller, G. F. (1996). *Co-Evolution Of Pursuit And Evasion II Simulation Methods And Results, From animals to animats 4.*
- [7] Dellaert, F. and Beer, R. D., (1996). *A Developmental Model for the Evolution of Complete Autonomous Agents. A developmental model for the evolution of complete autonomous agents., pages 393-401.*
- [8] Floreano, D. and Nolfi, S. (1999). *Learning and Evolution. Autonomous Robots, 7(1), 89-113.*
- [9] Fleischer J., Marshland S., and Shapiro J. (2003) *Sensory anticipation for autonomus selection of robot landmarks. In M. Butz, O. Sigaud, P. Gerard, Anticipatory Behaviour in adaptive Learning Systems. LNAI 2684, Springer Verlang, p. 110-132*
- [10] Golberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning. Addison Wesley.*
- [11] Hollan, J. H. (1975). *Adaptation in Natural and Artificial Systems. The University of Michigan Press.*
- [12] Koza, J.R. (1992). *Genetic Programming. MIT Press.*
- [13] Pal, S. and Konar, A. (1996). *Cognitive reasoning with fuzzy neural nets. IEEE Trans. on Systems, Man and Cybernetics, Part - B.*
- [14] Rosen, R. (1985). *Anticipatory Systems, p. 339. Pergamon Press.*
- [15] Stolzmann, W. (1998). *Anticipatory Classifier Systems. In Koza John R. et al. (editors). Genetic Programing 1998: Proceedings of the third Annual Conference, University of Wisconsin. Morgan Kaufmann.*
- [16] Tulving, E. (1987). *Multiple memory systems and consciousness. Human Neurobiology, vol. 6, pp. 67-80.*

A CONSTRAINT-BASED STOCHASTIC METHOD TO ADDRESS SCHOOL COURSE TIMETABLING PROBLEM

Piza Dávila H. Iván*, Campos Rodriguez Raul†.

*Centro Universitario de los Valles. Universidad de Guadalajara
Carretera Guadalajara - Ameca Km. 45.5 C.P. 46600. Ameca, Jalisco, México
Tel. +52 (375) 758 0500, correo electrónico: ivan.piza@profesores.valles.udg.mx
†Departamento de Ciencias Tecnológicas, División de Desarrollo Bio-Tecnológico
Centro Universitario de la Ciénega, Universidad de Guadalajara
Av. Universidad 1115, Col. Linda Vista, C.P. 47810, Ocotlán, Jalisco, México
Tel. +52 (392) 9259400, correo electrónico: raul.campos@cuci.udg.mx

ABSTRACT

School-course timetabling is a highly constrained combinatorial problem very hard to solve by conventional methods since it has a NP-hard nature. In many researches, it has been handled by means of evolutionary techniques [Mahdi03, Colomi93, Cooper93]. Genetic Algorithms [Holland75] is an adaptive and heuristic search algorithm based on the evolutionary ideas of natural selection; it is capable to find good solutions in a reasonable period of time. In this work, we propose an implementation of genetic algorithms which assigns lecturers and schedules to a given set of courses (academic offer) based on a number of fixed constraints, including: a) preferred subjects of lecturers, b) preferred schedules of subjects, c) preferred schedules of lecturers. Main goal is to minimize the number of conflicts in the resulting timetable. More kinds of constraints can be defined by subclassing class Constraint and overriding method sigma.

Keywords: Evolutionary computing, Constraint Satisfaction Problem, Genetic Algorithms, Scheduling, Combinatory.

I. INTRODUCTION

At the beginning of every school term, career coordinators at University of Guadalajara prepare a list of courses to offer, based on students' academic needs. Then, the various departments deal with the difficult task of assigning every course a lecturer and a schedule, trying to avoid conflicts.

Not only collisions in time-periods result in conflict but also giving a lecturer a subject he/she does not usually teach or a time period during which he/she is not available, to mention a few. Figuring out a reasonable assignment typically takes days (even weeks) and, very often, changes in teachers and/or schedules are carried out on the run (when the semester has already started).

A good academic offer should take into consideration most –or all– of the conditions (like the ones mention above) given by the department. Typically, some conditions –or constraints– are more critical than others. The problem intended to solve here belongs to the domain of constraint satisfaction problems (CSP), a task with a NP-complete degree of complexity [Piola92].

Seeing course timetabling problem as a constraint satisfaction problem allows the use of numerous techniques developed to solve CSP [Baykan91]. Stochastic methods represent a major approach to compute solutions. One of the most popular stochastic methods is Genetic Algorithms.

This paper is organized as follows. Section 2 introduces Genetic Algorithms. Section 3 describes the problem we intend to solve. Section 4 deepens in the solution proposed, emphasizing in individual representation and fitness calculation. Section 5 describes important details about implementation. Section 6 shows some experimental results and Section summarizes the conclusions of this work.

II. GENETIC ALGORITHMS

Genetic algorithms were formally introduced by John Holland at University of Michigan [Holland75] in the 1970s. A Genetic Algorithm is an adaptive heuristic search algorithm based on the evolutionary ideas of natural selection [Darwin64] and genetics. In [Reeves93], the author defines a heuristic procedure as a technique which finds good solutions (quasi-optimal) with a reasonable computational cost but with no promise of optimality or feasibility.

A typical genetic algorithm creates a population of solutions –called individuals– and applies genetic operators to them in order to find the best one(s). Three major aspects to consider when implementing a problem-specific genetic algorithm are the following:

1. Definition of the objective function to evaluate each solution. This function is used to calculate the fitness of each individual. Greater fitness values denote good solutions. Individuals with great values of fitness tend to preserve along generations.
2. Definition and implementation of the genetic representation. Every individual has a phenotype and a genotype. The phenotype stores the values used in the objective function. The genotype is represented by a chromosome. A chromosome is a chain of values or genes. Typically, a chromosome is in the form of a chain of bits or an integer array. A GA-implementation should include a method to translate genotypes into phenotypes.
3. Definition and implementation of the genetic operators: selection, crossover and mutation. Genetic Algorithms give big significance to crossover operator. These operators work with genotypes, instead of phenotypes. There exist several techniques for each of these operators.

Once these three aspects have been defined, the genetic algorithm should work fairly well. A typical genetic algorithm consists of the following steps:

1. Create a random population of solutions.
2. Calculate fitness of each individual
3. Repeat
 - a. Selection
 - b. Crossover
 - c. Mutation
 - d. Translate genotypes into phenotypes
 - e. Calculate fitness of each individual
 - f. Increase generation index in 1

Until best solution is good enough or generation index exceeds a given maximum.

III. PROBLEM DESCRIPTION

The problem considered in this work is informally defined as follows. Consider the beginning of every semester in a given university. Career coordinators must analyze student's academic demands to figure out which subjects should be offered in the next semester. Some subjects have such a demand that two or more groups are required to be offered. However, there are cases in which no courses are opened for a particular subject whose demand is quite low. Moreover, coordinators must also consider the available resources for supporting academic demands, in this case, lecturers and their timing constraints and preferences. The timetabling problem derived from selecting subject, lecturer and timing is addressed in this work.

The solution proposed in this paper was tested in a real scenario: the department of Computer Science and Engineering from University of Guadalajara, Campus Valles, at Ameca, Jalisco, Mexico. The final arrangement produced by the algorithm should avoid collisions in schedules and take into consideration all the constraints inherent to academic institutes, including, among others, the following:

- Professional profile of lecturers,
- Preferred schedules of subjects, and
- Availability of lecturers.

Constraints about classrooms availability are not considered in this version of the solution.

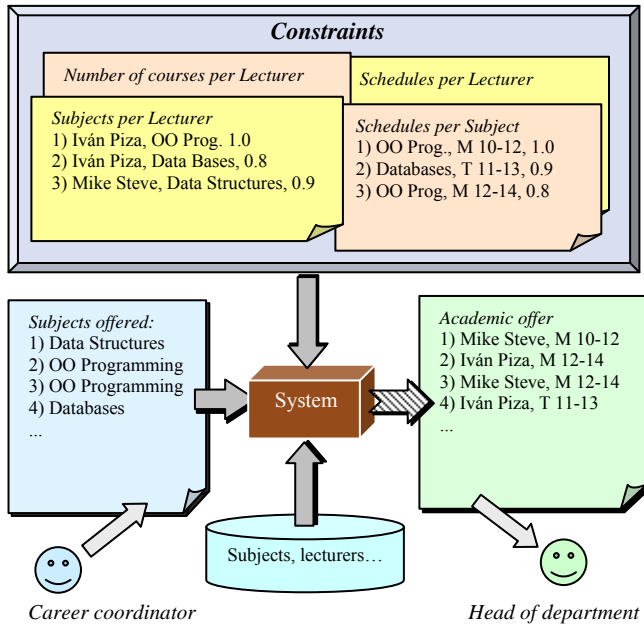


Figure 1: Problem Formulation Conceptual Diagram

Figure 1 depicts a conceptual diagram for the problem formulation. The system’s input includes: a) a set of subjects to be offered, b) lecturers available and c) constraints to be satisfied. The problem consists on finding out an academic offer which best fulfills a given set of constraints in a reasonable period of time.

Formally, an academic offer is a 6-tuple $AO = \langle S, L, \delta, T, G, C, \alpha \rangle$, where:

S is a non-empty finite set called the *subjects*

L is a non empty indexed set called the *lecturers*

$\delta : L \rightarrow N$ is a function representing the number of courses preferably assigned to a lecturer.

T is a non-empty indexed set called the *schedules*, or time periods, such that every $t \in T$ is formed by a 3-tuple $\langle day_of_week, initial_time, duration \rangle$.

$G = \{g_k \mid g_k \in S, 1 \leq k \leq n\}$ is a non-empty finite indexed set called the *groups* (subjects offered).

$C \subset (L \times S \times \mathbb{R}^+) \cup (L \times T \times \mathbb{R}^+) \cup (S \times T \times \mathbb{R}^+)$ is a finite set called constraints.

$\alpha : G \rightarrow (L \times T)$ is the assignment function. Every subject offered is imparted once a week by the same lecturer the whole semester.

Two kinds of constraints, known as *hard* and *soft*, are considered in this work. On the one hand, *hard constraints* are those to which an academic offer has to stick in order to be feasible. For example, each lecturer can teach only one class at a given time is a hard constrain. On the other, *soft constraints* are those which may be satisfied in order to reach a more suitable assignment of teachers and schedules. The more number of *soft constraints* are satisfied the more appropriate is the timetabling solution. Soft constraints are weighted for denoting how bad it would be not satisfying such a constraint.

Current implementation of the solution considers three classes of soft constraints:

1. Every lecturer is expert in one or two subjects, and has knowledge in some others.
2. Some subjects are preferably imparted at specific periods of time.
3. Some time periods are inconvenient to some lecturers.

IV. PROPOSED SOLUTION

The problem of finding out the best academic offer is a combinatorial problem: every assignment of time period and lecturer is a permutation with repetition, leading to an NP-Hard problem [Graham96].

Let n, l, t be the amount of courses offered, lecturers available and existing time periods, respectively. The number of the entire possible assignments is calculated as follows:

$$(OR_{l,n})(OR_{t,n}) = l^n t^n \tag{Equation 1}$$

Equation 1 represents an exponential number of possibilities. A typical academic offer from the Department of Computer Science at the Centro Universitario de los Valles, includes around sixty

courses and considers twenty lecturers and thirty time periods. Hence, in the worst case, where there is no limit in the number of courses per lecturer and time period, there is approximately 4.88×10^{166} different academic offers to evaluate in order to find the best one.

In a simpler case, where every lecturer and time period is assigned the same amount of courses (3 and 2, respectively), this number reduces to 1.85×10^{145} , still a huge number of possibilities.

However, genetic algorithms have proved to reach feasible solution in a reasonable period of time when dealing with combinatorial problems [Holland75].

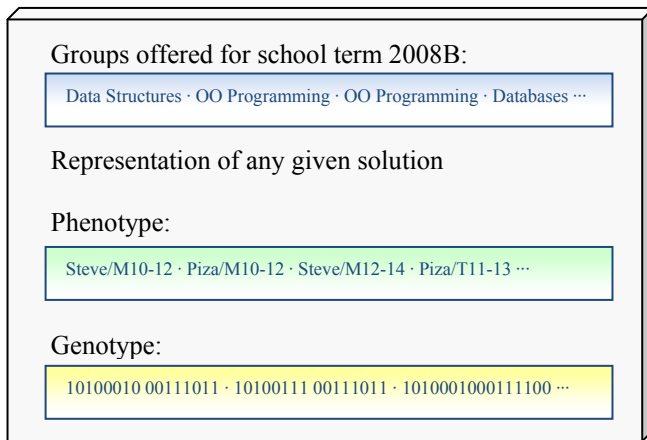


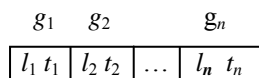
Figure 2: Solution Representation

Solution Codification

Figure 2 shows the codification of the solution as genetic structures. Every individual represents a potential academic offer and consists on the following elements:

- a) *Phenotype* φ : a vector of n pairs $\langle l, t \rangle$, such that:
 - n is the number of offered groups
 - $l \in L$ is a lecturer, and
 - $t \in T$ is a time-period.

- b) *Genotype* γ : a chain of n 16-bit strings of the form:



where:

- $1 \leq l_i \leq l$ is an 8-bit index of a lecture in L

- $1 \leq t_i \leq t$ is an 8-bit index of a time period in T
- c) *Fitness*: a double-precision number
- d) *Expected value*: a double-precision number used for *Roulette Selection*

Given N , the number of courses preferably assigned to every lecturer in L , i.e., $N = \sum_{l \in L} \delta(l)$, as shown in Figure 3, and the genotype γ for any individual of the population, the following algorithm computes its phenotype φ .

Algorithm 1: Phenotype Construction

```

For every  $0 \leq k \leq n$ , do:
Let  $\gamma_k$  be the  $k^{\text{th}}$  16-bit chain from genotype  $\gamma$ 
The  $k^{\text{th}}$  pair  $\langle l_k, t_k \rangle$  for  $\varphi$  is computed as:
value  $\leftarrow (\gamma_k \& 0 \times \text{FF}00) \gg 8$  [8 msb of chain  $\gamma_k$ ]
nvalue  $\leftarrow (\text{value} \times N) / 0 \times \text{FF}$  [normalized to  $0 \dots N$ ]
index  $\leftarrow \text{getIndex}(\text{nvalue})$  [index of lecturer]
 $l_k = L_{\text{index}}$  [lecturer selected]
value  $\leftarrow \gamma_k \& 0 \times 00\text{FF}$  [8 lsb of chain  $\gamma_k$ ]
index  $\leftarrow (\text{index} \times |T|) / 0 \times \text{FF}$  [normalized  $0 \dots |T|$ ]
 $t_k = T_{\text{index}}$  [time-period selected]
End-The
End-For
    
```

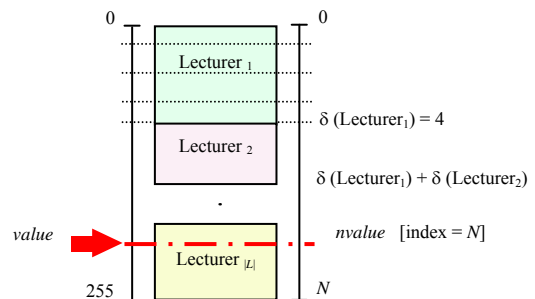


Figure 3: Preferred number of groups per lecture

Fitness Evaluation

The *fitness* function is a key aspect in the design of the proposed solution. For this implementation, the fitness of an individual is directly affected by the number of non-satisfied constraints for a given academic offer. The least constraints are violated, greater is the fitness value. It is clear to see that the more constraints exist, harder and more time-consuming is to find a conflict-free solution.

Algorithm 2 estimates fitness for each individual in the current solution. Notice that academic offers with collisions in time-periods (hard constraint) are not completely discarded since they may be close to a suitable solution that may be re-arranged by the academic coordinator in a post-processing procedure. After running this algorithm, the fitness ends with a value between 0 and 1.

Algorithm 2: Fitness Estimation

```

For every individual I in population, do:
  Get I.phenotype from I.Genotype
  Fitness ← 1.0
  If I.phenotype has time-collisions, then
    Fitness ← 0.5
  Sigma ← 1.0
  For each constraint c ∈ C, do:
    Sigma ← Sigma + c.sigma()
  End-For
  Fitness ← Fitness / Sigma
  I.fitness ← Fitness
End-For

```

V. SOLUTION IMPLEMENTATION

The system implementation follows an object-oriented approach. Figure 4 depicts a class diagram of the system architecture. The classes Group, Population, Individual, Algorithm, Assignment and Constraint conforms the current version of the implementation.

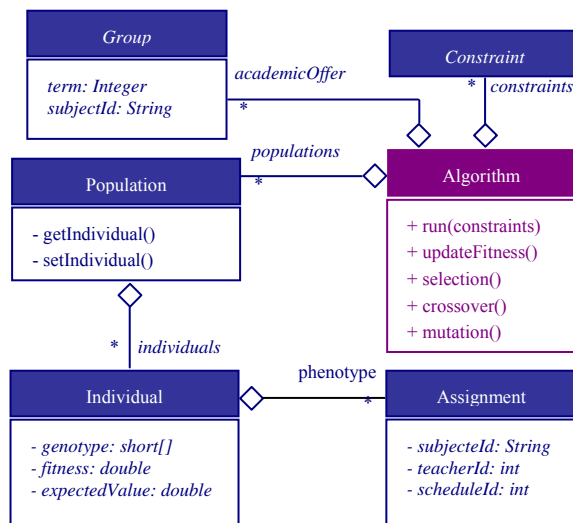


Figure 4: Class Diagram of System Architecture

Algorithm class implements all the methods inherent in a genetic algorithm: selection, crossover and mutation operators. Also, it contains a population of individuals. Each individual includes a list of assignments $\langle \text{subject}, \text{teacher}, \text{schedule} \rangle$. Besides, class *Algorithm* has:

- a list of subjects proposed –the *academic offer*– and
- a set of constraints

Figure 5 shows a class diagram for constraints implementation. As shown, every constraint is actually an instance of a subclass of an abstract class called *Constraint*, which overrides its method *sigma()*. This means that every constraint computes *sigma* in a particular way. Algorithm 3 is the implementation of the abstract method *sigma(phenotype)* in subclass *Teacher-Subject*.

Algorithm 3: Sigma Calculation

Let n be the amount of courses offered.

```

Sigma(phenotype φ):
  found ← false
  k ← 0
  While k ≤ n and not found:
    Let <lecturer, time_period> be the kth pair
    from φ
    subject ← get subject indexed i from
    academic offer
    If lecturer.Id ≠ value1
      continue with loop k + 1
    End-If
    subject ← get subject indexed i from
    academicOffer
    If subject.Id = value2
      found ← true
    End-If
  End-While
  If found return 0, else return weight
End-Sigma

```

Genetic operators

As in any typical genetic algorithm, fitness evaluation is performed on phenotypes and genetic operators act on genotypes. Individuals are chosen by an elitist *Roulette* selection. The best individual

always preserves. Better solutions are more likely to pass to the next generation than bad ones.

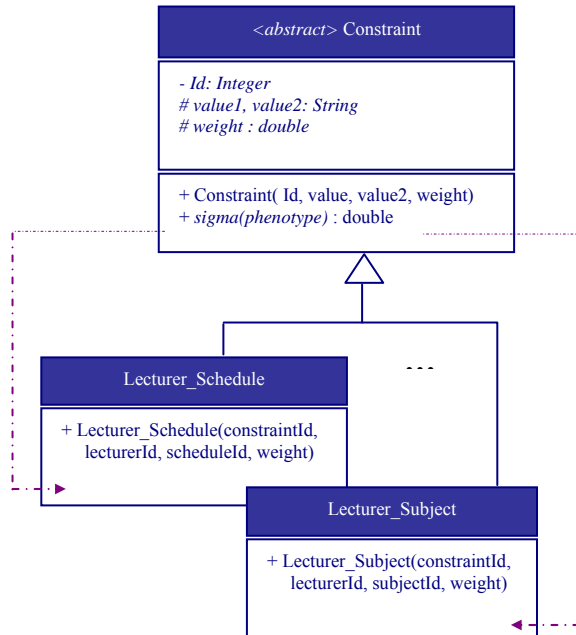


Figure 5: Constrains Implementation

Crossover technique employed is *two-point*. It is carried out between pairs of adjacent individuals ($k, k + 1$), where k is an even index ($0, 2 \dots$), and $k, k + 1 \neq$ best individual index. As can be seen in Figure 6, crossover exchanges 16-bit chains. Mutation algorithm has three steps:

- 1) Change a randomly selected bit, as shown in Figure 7, and evaluate,
- 2) Change a randomly selected group of 16 bits to a random 16-bit value and evaluate,
- 3) Keep the better one.

Mutation is performed only if $k \neq$ best individual index.

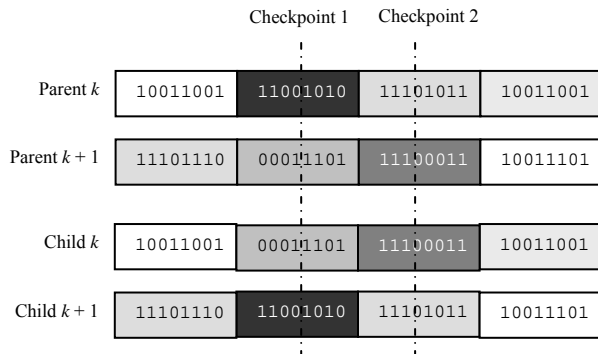


Figure 6: Two-point Crossover

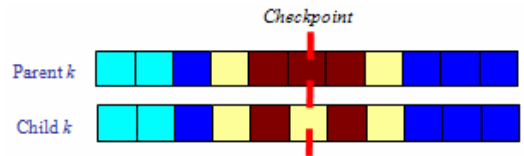


Figure 7: Uniform Mutation

VI. EXPERIMENTAL RESULTS

The proposed algorithm was tested in a real scenario, the timetabling problem at the University of Guadalajara Campus Ciudad Valles. The academic offer at the campus is as follows:

- Groups: 467
- Professors: 144
- Subjects: 317
- Time periods: from 07:00hrs to 19:00hrs

The time is divided in periods of two hours, for a total of eight time periods. Using previous data in Equation 1, it results in approximately $3,022^{970}$ possible combinations.

Table 1 shows a reduced population of the timetabling estimated by the algorithm while Table 2 shows the ideal timetabling for the same population. From the tables, it can be shown that only the courses Introduction to Computing Theory and Introduction to the Programming Languages was assigned to a professor other than the ideal. Moreover, the timing for the course is very close to that one preferred by the professors.

Table 1: Section of the Estimated Timetabling

DATABASE MANAGEMENT	12:00hrs	MON	PIZA DÁVILA HUGO IVÁN
FILE STRUCTURES IN C++	10:00hrs	MON	BARBOZA RODRÍGUEZ RODOLFO
FILE STRUCTURES IN C++	10:00hrs	THU	DE LA TORRE GOMORA MIGUEL ANGEL
DATA STRUCTURES IN C++	08:00hrs	FRI	ANCONA VALDEZ MARIA DE LOS ANGELES
DATA STRUCTURES IN C++	11:00hrs	TUE	ESTEVEZ SIORDIA MIGUEL ALEJANDRO
SOFTWARE ENGINEERING I	08:00hrs	TUE	ALVAREZ TOSTADO MARTINEZ ERENDIRA
SOFTWARE ENGINEERING II	11:00hrs	TUE	ALVAREZ TOSTADO MARTINEZ ERENDIRA
INTRODUCTION TO COMPUTING THEORY	12:00hrs	MON	ESTEVEZ SIORDIA MIGUEL ALEJANDRO
PROGRAMMING LANGUAGES	12:00hrs	TUE	PIZA DÁVILA HUGO IVÁN
PROGRAMMING LANGUAGES	10:00hrs	TUE	ANCONA VALDEZ MARIA DE LOS ANGELES

This experiment that computes the results in Table 1 was performed including the following elements, selected by a random process:

- 32 groups, i.e., 6.85%,
- 11 professors, i.e., 7.64%,
- 23 subjects, i.e., 7.26%

Table 2: Section of the Ideal Timetabling

DATABASE MANAGEMENT	11:00hrs	TUE	PIZA DÁVILA HUGO IVÁN
FILE STRUCTURES IN C++	08:00hrs	TUE	DE LA TORRE GOMORA MIGUEL ANGEL
FILE STRUCTURES IN C++	08:00hrs	TUE	BARBOZA RODRÍGUEZ RODOLFO
DATA STRUCTURES IN C++	10:00hrs	MON	ANCONA VALDEZ MARIA DE LOS ANGELES
DATA STRUCTURES IN C++	10:00hrs	MON	ESTEVEZ SIORDIA MIGUEL ALEJANDRO
SOFTWARE ENGINEERING I	11:00hrs	TUE	ALVAREZ TOSTADO MARTINEZ ERENDIRA
SOFTWARE ENGINEERING II	10:00hrs	MON	ALVAREZ TOSTADO MARTINEZ ERENDIRA
INTRODUCTION TO COMPUTING THEORY	14:00hrs	FRI	ALARCÓN MARTÍNEZ TERESA EFIGENIA
PROGRAMMING LANGUAGES	10:00hrs	TUE	ANCONA VALDEZ MARIA DE LOS ANGELES
PROGRAMMING LANGUAGES	10:00hrs	TUE	ALARCÓN MARTÍNEZ TERESA EFIGENIA

The parameters for the genetic algorithm were as follows:

- Crossover rate: 0,8
- Mutation rate: 0,3
- Generations: 3000
- Population: 200

The best fitness, with value 0.082, was found in generation number 3000. The delay of the algorithm in this experiment was 650 seconds.

VII. CONCLUSION

In this paper, we have proposed a Genetic Algorithm to handle the timetabling problem inherent in Universities academic planning. This is a NP-hard optimization problem that should be handled using stochastic methods instead of deterministic ones because of the combinatory explosion. We have tested the method proposed to assign lecturers and schedules to a fixed set of subjects offered at Centro Universitario de los Valles, Universidad de Guadalajara. Considering that every lecture has preferred subjects and schedules, and every subject has preferred schedules too, we have treated timetabling problem as a constraint satisfaction problem. Every individual denotes a potential academic offer <Subject, Teacher, Schedule>*. Fitness algorithm is in terms of the amount of constraint violations. The more constraints are considered the harder and time-consuming is to find an optimal solution. Considering 20 lecturers, 30 time-periods, 60 courses offered and 100 constraints, we have found acceptable solutions using around 2000 generations and 3000 individuals in the population with 1-hour of processing in a desktop computer.

VIII. REFERENCES

- [Baykan91] C. Baykan, M. Fox. Constraint satisfaction techniques for spatial planning, in Intelligent CAD Systems III - Practical experience and Evaluation, pp. 187-204, 1991.
- [Colorni93] A. Colorni, M. Dorigo. Genetic Algorithm To Solve The Timetable Problem. Politecnico di Milano, Italy, 1993.
- [Cooper93] Cooper, T.B., Kingston, J.H. The solution of real instances of the timetabling problem. The Computer Journal, 36(7), 645-653. 1993.
- [Darwin64] Charles Robert Darwin. On the Origin of Species by Means of Natural Selection Or the Preservation of Favoured Races in the Struggle for Life. Cambridge University Press, Cambridge, UK, sixth edition, 1964. Originally published in 1859.
- [Mahdi03] El Mahdi, O.; Aion, R.N.; Zainuddin, R. Using a Genetic Algorithm Optimizer Tool to solver University Timetable Scheduling Problem. Electronics, Circuits and Systems, 2003. ICECS 2003. December 2003.
- [Graham96] Graham, R.L., Groetschel M., and Lovász L., eds. (1996). Handbook of Combinatorics, Volumes 1 and 2. Elsevier (North-Holland), Amsterdam, and MIT Press, Cambridge, Mass. ISBN 0-262-07169-X.
- [Holland75] J. H. Holland. "Adaptation in Natural and Artificial Systems". University of Michigan Press, Ann Arbor, 1975
- [Piola92] R. Piola. Evolutionary solutions to a highly constrained combinatorial problem. Dipartimento di informatica, Univeristá degli di Torino, Italy, 1992.
- [Reeves93] Colin B. Reeves, editor. Modern Heuristic Techniques for Combinatorial Problems. John Wiley & Sons, Great Britain, 1993.

MANEJO DE EVENTOS COMPUESTOS EN SISTEMAS RFID

González Padilla Omar, Ramos Corchado Félix, Unger Herwig

CINVESTAV Unidad Guadalajara, Av. Científica 1145, Col. El Bajío, 45010 Zapopan, Jal. México
Email: {oglez,framos}@gdl.cinvestav.mx, Herwig.Unger@fernuni-hagen.de

Resumen. Éste artículo propone un nuevo enfoque para desarrollar sistemas RFID. La idea principal es incluir dos componentes para manejar eventos compuestos. Dichos componentes son un lenguaje declarativo y una capa de reconocimiento. El lenguaje declarativo, llamado RFID-CEDL permite a las aplicaciones definir los eventos compuestos que les interesen. La capa de reconocimiento analiza el flujo de observaciones RFID buscando ocurrencias de eventos de interés. Cuando la ocurrencia de un evento es detectada, la aplicación correspondiente es notificada.

Palabras clave: RFID, evento compuesto, evento de alto nivel, reconocimiento, detección, situación.

I. INTRODUCCIÓN

La identificación por radio frecuencia (RFID) es una tecnología utilizada para la identificación automática de personas y objetos. La base de RFID es almacenar datos en etiquetas (generalmente un ID único [1]), pegar dichas etiquetas en objetos y obtener el contenido de las etiquetas mientras los objetos se mueven utilizando lectores estratégicamente colocados. El uso de señales de radio frecuencia permite una comunicación remota que no requiere línea de visión.

Hoy en día, muchos sistemas RFID acatan los estándares de EPCglobal [2]. Dichos estándares definen la arquitectura de los sistemas RFID, sus componentes y su comportamiento. La figura 1 ilustra los componentes básicos de la arquitectura EPCglobal.

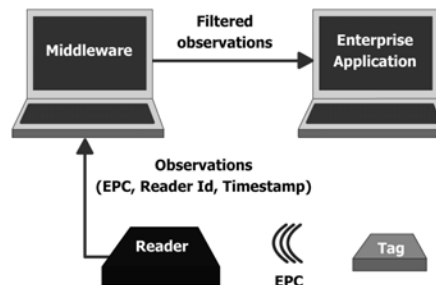


Fig. 1. Componentes principales de EPCglobal.

Los Códigos Electrónicos de Producto (EPC) son identificadores almacenados en las etiquetas. Cada EPC debe ser único, de manera que identifique unívocamente a un objeto. La infraestructura de hardware RFID comprende lectores y etiquetas; cada vez que una etiqueta está dentro del rango de lectura de un lector, el EPC correspondiente es transmitido por la etiqueta y recibido por el lector. Cuando un lector recibe un EPC se produce una *observación* (la cuál contiene el EPC, el ID del lector, y una marca de tiempo). El middleware obtiene observaciones de varios lectores y su función principal es filtrar la información, eliminando lecturas duplicadas o erróneas [3]. Después de su paso por el middleware, las observaciones son enviadas a las aplicaciones, las cuales están a cargo de analizarlas con el objetivo de reconocer patrones y detectar situaciones más complejas que ocurren en el entorno.

A pesar de que el proceso de filtrado realizado por el middleware reduce errores en la información, las observaciones RFID son difíciles de analizar porque se generan continuamente y proveen información de bajo nivel. La granularidad de las observaciones RFID generan dos inconvenientes:

- La mayoría de las aplicaciones están interesadas en eventos más complejos que simples

observaciones. Por lo tanto, tales aplicaciones deben realizar un análisis sobre las observaciones para poder detectar patrones relevantes o relaciones entre las observaciones. Realizar éste análisis incrementa la complejidad al desarrollar las aplicaciones.

- Muchas observaciones que son enviadas a las aplicaciones no son relevantes para detectar ningún evento de interés. Consecuentemente, se utilizan recursos de red innecesariamente, lo cual afecta al desempeño.

En la mayoría de los sistemas RFID actuales, la interacción entre las aplicaciones y el middleware es realizada mediante el uso de una interfaz llamada Application Level Events (ALE) [4], la cual permite a las aplicaciones especificar un conjunto de etiquetas de interés y ajustar algunos parámetros para contar, agrupar y reportar observaciones de dichas etiquetas. Usando ALE, los lectores físicos pueden ser organizados como lugares lógicos. Usando ésta información, el middleware puede seleccionar la información importante para la aplicación. Otra ventaja de ALE es que permite a las aplicaciones decidir como recibir la información; es posible obtener el conjunto completo de observaciones, o reportes diferenciales de observaciones (inclusiones o remociones en relación al reporte previo). La Figura 2 muestra la interacción entre las aplicaciones y el middleware utilizando ALE.

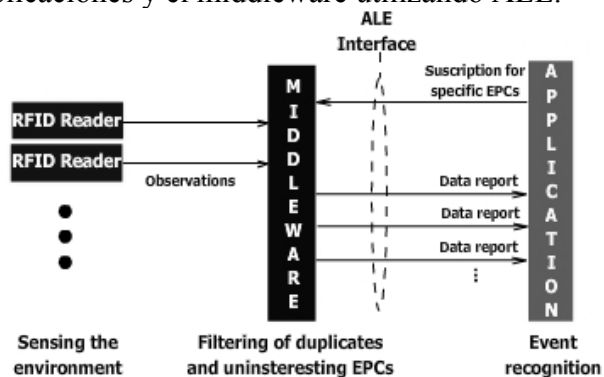


Fig. 2. Interacción mediante ALE.

El uso de ALE proporciona información más fácil de manejar; sin embargo, no resuelve los problemas previamente mencionados.

En nuestra propuesta, hemos modificado la arquitectura actual para poder proveer manejo de eventos compuestos [5]. Presentamos un nuevo enfoque para el desarrollo de aplicaciones RFID en el que las aplicaciones definen un conjunto de eventos de interés utilizando un lenguaje declarativo. Cada aplicación es notificada sólo cuando un evento relevante ocurre, en lugar de recibir continuamente información de bajo nivel acerca de la localización de los objetos.

El problema de especificar eventos compuestos utilizando información RFID fue tratado en [6]; éste enfoque está enfocado en definir eventos compuestos como patrones en un historial de eventos en lugar de tomar en cuenta las posiciones de los objetos. En [7], se presenta una propuesta para definición y reconocimiento de eventos mediante grafos conceptuales, donde los nodos son actores de la situación y los arcos son relaciones entre ellos; sin embargo ésta propuesta es general y no está enfocada en información RFID. Un sistema para consultar y obtener eventos de alto nivel compuestos básicamente por secuencias de observaciones atómicas es presentado en [8] y extendido en [9] con la inclusión de la cerradura de Kleene para definir patrones de eventos. En [10], se presenta un sistema para definir y reconocer eventos con datos RFID, éste enfoque utiliza probabilidad para manejar el ruido y los errores inherentes a la radio frecuencia.

El resto del artículo está organizado de la siguiente manera: primero, presentamos en detalle nuestra propuesta (sección 2); después, presentamos los tipos de evento manejables en nuestra plataforma (sección 3); en la sección 4, introducimos el conjunto de operadores disponibles para construir eventos; en seguida, presentamos nuestro lenguaje para definir de eventos (sección 5); en la sección 6, presentamos el componente que se encarga de analizar los datos RFID y reconocer ocurrencias de eventos; finalmente, presentamos algunas conclusiones (sección 7).

II. PROPUESTA GENERAL

Para superar los problemas presentados en la sección anterior, agregamos una capa entre el middleware y las aplicaciones. Ésta capa se encarga del reconocimiento de eventos actuando

como un intermediario entre el middleware y las aplicaciones: por un lado, obtiene de las aplicaciones el conjunto de eventos de interés especificados usando nuestro lenguaje; por otro lado, utiliza ALE para interactuar con el middleware y obtener información filtrada necesaria para detectar la ocurrencia de dichos eventos. A pesar de que la capa de reconocimiento es un componente independiente, está situado en la misma computadora que alberga al middleware, ésta característica reduce el tráfico en la red considerablemente. La capa de reconocimiento analiza la información proveniente del middleware y notifica a las aplicaciones cuando reconoce eventos interesantes. La capa de reconocimiento también almacena un historial de ocurrencias de eventos, el cuál es útil para reconocer eventos definidos como patrones de ocurrencias de eventos más simples. La figura 3 ilustra nuestro enfoque de manera general.

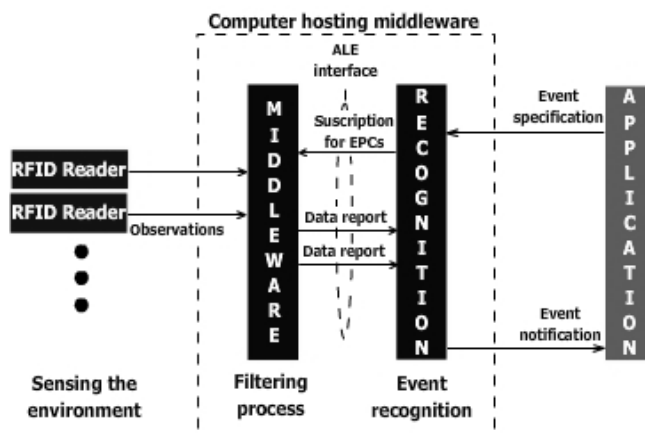


Fig. 3. Enfoque general de la propuesta.

III. EVENTOS

Es indispensable para nuestro trabajo definir los tipos de evento que puede manejar la capa de reconocimiento. Hemos tomado conceptos de Procesamiento de Eventos Complejos [5] y los hemos extendido para trabajar con información RFID. En esta sección presentamos nuestro modelo de eventos.

Incluimos dos categorías de eventos: *eventos primitivos* y *eventos compuestos*. Los eventos primitivos son generados automáticamente por la capa de reconocimiento a partir del flujo de

observaciones RFID provenientes del middleware. Los eventos compuestos se construyen combinando eventos primitivos y otros eventos compuestos usando un conjunto de operadores predefinidos.

Eventos Primitivos

Los eventos primitivos son el punto de partida para realizar el reconocimiento de eventos. Hay cuatro tipo de eventos primitivos: *Observation*, *Arrival*, *Departure*, y *Stance*.

Los eventos de tipo *Observation* son generados por los lectores RFID y son la unidad más pequeña de información manejada por la capa de reconocimiento. Un *Observation* es una tripleta $\langle \text{EPC}, \text{lector}, \text{marca de tiempo} \rangle$ producida siempre que un objeto etiquetado esta dentro del rango de lectura de un lector. Definimos un *Observation* como:

$$\text{Observation}(x, y, t)$$

Donde:

x es el id de la etiqueta observada.

y es el identificador del lector que detecta la etiqueta.

t es la marca de tiempo de la lectura.

Analizar directamente el flujo de observaciones es difícil dado que los lectores tienen altas tasas de lectura y el volumen de información generado es enorme. En lugar de realizar el reconocimiento de eventos compuestos analizando observaciones directamente, usamos *información diferencial*. La información diferencial es información acerca de entradas y salidas de etiquetas con respecto a los rangos de lectura de los lectores. Definimos otros dos eventos primitivos para expresar la información diferencial: *departure* y *arrival*.

Un evento *departure* para la etiqueta x y el lector y se produce cuando x es observada por y y después de dicha observación, x no es observada por y durante un periodo de tiempo p . La duración de p depende de la tasa de lectura y la precisión de los lectores.

$$(\exists x, y, t_1)((\text{Observation}(x, y, t_1) \wedge (\forall t_2)(\text{Observation}(x, y, t_2) \rightarrow ((t_2 \leq t_1) \vee (t_2 \geq t_1 + p)))))) \rightarrow \text{Departure}(x, y, t_1)$$

Un evento *arrival* de la etiqueta x al lector y se produce cuando x es observada por y y para todas las observaciones previas de x por y existe una salida posterior.

$$(\exists x,y,t_1)(Observation(x,y,t_1) \wedge (\forall t_2)((Observation(x,y,t_2) \wedge (t_2 < t_1)) \rightarrow (\exists t_3)(Departure(x,y,t_3) \wedge (t_3 > t_2) \wedge (t_3 < t_1))) \rightarrow Arrival(x,y,t_1))$$

La información que proveen las entradas y salidas puede comprimirse creando un evento *stance*. Un evento *stance* es creado por cada llegada y su salida correspondiente.

$$(\exists x,y,t_1,t_2)((Arrival(x,y,t_1) \wedge Departure(x,y,t_2) \wedge (t_1 < t_2) \wedge (\neg \exists t_3)(Arrival(x,y,t_3) \wedge (t_3 > t_1) \wedge (t_3 \leq t_2))) \rightarrow stance(x,y,t_1,t_2))$$

Definimos un termino llamado *stays* el cual evalúa si una etiqueta x está localizada dentro del rango de lectura de un lector y en un momento dado t . El valor de *stays* es obtenido en base a los eventos *stance*.

$$(\exists x,y,t_1,t_2,t)((stance(x,y,t_1,t_2) \wedge (t \geq t_1) \wedge (t \leq t_2)) \rightarrow stays(x,y,t))$$

Eventos compuestos

Los eventos compuestos se definen utilizando un conjunto de operadores predefinidos. Cuando un evento compuesto es definido debe tener un nombre n y un conjunto de campos $F := \{f_1, f_2, \dots, f_n\}$. Por cada evento compuesto, la capa de reconocimiento genera un tipo de evento análogo que además incluye un campo extra llamado t . El valor de t almacena la marca de tiempo para cada ocurrencia del tipo de evento. De acuerdo a esto la estructura general de un evento compuesto dentro de la capa de reconocimiento es:

$$n(f_1, f_2, \dots, f_n, t)$$

Si se produce una ocurrencia de un evento de tipo n en el momento t , la capa de reconocimiento almacena sus valores v_1, v_2, \dots, v_n . Representamos una ocurrencia de un evento compuesto como:

$$n(v_1, v_2, \dots, v_n, t)$$

IV. OPERADORES

Un evento compuesto se construye utilizando eventos más simples, propiedades de objetos etiquetados, ubicación de objetos etiquetados e información adicional. En esta sección presentamos el conjunto de operadores disponibles en nuestra propuesta para construir eventos compuestos.

Operadores de ubicación

Los eventos primitivos proveen información acerca de la ubicación de un único objeto. Basados en esta restricción, identificamos la necesidad de agrupar información acerca de la ubicación de varios objetos. En consecuencia incluimos tres operadores: *all*, *any*, y *none*; cada uno de estos operadores es evaluado como un valor booleano. Sus valores se calculan de acuerdo a un conjunto de objetos $X := \{x_1, x_2, \dots, x_n\}$, un lector y , y una marca de tiempo t . A continuación definimos dichos operadores.

$$All(x_1, x_2, \dots, x_n, y, t) \leftrightarrow stays(x_1, y, t) \wedge stays(x_2, y, t) \wedge \dots \wedge stays(x_n, y, t)$$

$$Any(x_1, x_2, \dots, x_n, y, t) \leftrightarrow stays(x_1, y, t) \vee stays(x_2, y, t) \vee \dots \vee stays(x_n, y, t)$$

$$None(x_1, x_2, \dots, x_n, y, t) \leftrightarrow \neg stays(x_1, y, t) \wedge \neg stays(x_2, y, t) \wedge \dots \wedge \neg stays(x_n, y, t)$$

Operadores de Ocurrencia

Incluimos dos operadores para definir eventos compuestos utilizando el historial de ocurrencias de eventos. Estos operadores son: *sequence* y *combination*.

Una secuencia de eventos es definida especificando una lista de nombres de tipos de eventos, una marca de tiempo t , y un entero i . Una secuencia es reconocida si todos los eventos en la lista de eventos ocurren ordenados dentro del periodo comprendido de entre $t-i$ y t .

$$Sequence(ev_1, ev_2, \dots, ev_n, t, i) \leftrightarrow ev_1(v_{11}, v_{12}, \dots, v_{1n}, t_1) \wedge$$

$$ev_2(v_{21}, v_{22}, \dots, v_{2m}, t_2) \wedge \dots \wedge ev_n(v_{n1}, v_{n2}, \dots, v_{nm}, t_n) \wedge (t-i) \leq t_1 < t_2 < \dots < t_n \leq t$$

Una combinación de eventos es definida similarmente a una secuencia; sin embargo para una combinación el orden de ocurrencia de los eventos no es relevante.

$$\text{Combination}(ev_1, ev_2, \dots, ev_n, t, i) \leftrightarrow ev_1(v_{11}, v_{12}, \dots, v_{1n}, t_1) \wedge ev_2(v_{21}, v_{22}, \dots, v_{2m}, t_2) \wedge \dots \wedge ev_n(v_{n1}, v_{n2}, \dots, v_{nm}, t_n) \wedge (t-i) \leq t_j \leq t$$

Donde:

$$j=1, 2, \dots, n$$

Operador de Existencia

En nuestra propuesta es posible definir eventos utilizando información adicional indirectamente relacionada a los objetos etiquetados. Por ejemplo para reconocer medicaciones alérgicas en un hospital es necesario saber, a que medicinas es alérgico cada paciente. El operador *exists* toma como argumentos un conjunto de valores específicos y es evaluado como verdadero solamente si se encuentran valores correspondientes dentro de la información adicional.

Operadores Booleanos

Los operadores de ubicación, ocurrencia y existencia, son evaluados como valores booleanos. Es posible usar operadores booleanos para combinar sub expresiones compuestas por tales operadores. Permitimos el uso de conjunciones, disyunciones y negaciones. Decidimos usar este subconjunto de operadores booleanos debido a que es funcionalmente completo.

Operadores de Comparación

Tanto los eventos como los objetos etiquetados se componen de campos. Cada instancia de un evento o de un objeto incluye valores para los campos correspondientes. Al definir un evento, es posible comparar los valores de los campos usando operadores de comparación (<, ≤, >, ≥, ≠, =). En la siguiente sección mostramos como definir eventos y objetos.

V. RFID-CEDL

En las secciones anteriores hemos definido los tipos de eventos primitivos y el conjunto de operadores disponibles para definir tipos de eventos compuestos. En esta sección presentamos el lenguaje que permite definir eventos compuestos. Dicho lenguaje, llamado RFID-Composite Event Definition Language (RFID-CEDL) es declarativo y tiene una estructura basada en XML. La principal ventaja de usar una sintaxis XML es su facilidad de uso y validación.

Para definir el conjunto de eventos que le interesan a una aplicación, es necesario crear una especificación. Una especificación es un documento XML que debe cumplir con cierta estructura. En esta sección presentamos dicha estructura y sus cinco secciones principales; ilustramos cada sección con ejemplos de su uso en un entorno hospitalario.

Clases

Dentro de la sección *classes*, definimos las estructuras de datos necesarias para definir eventos compuestos. Cada clase se compone de campos (*fields*); un *field* es un atributo de la clase y es definido mediante un tipo de dato y un nombre.

Las clases pueden corresponder a un tipo de objeto etiquetado o no. El proceso de definir clases es similar a definir la estructura de bases de datos de un sistema; los desarrolladores deciden que clases son necesarias y que campos requiere cada una de ellas.

Por ejemplo, para reconocer medicaciones alérgicas en un entorno hospitalario, es necesario definir una clase *paciente*, una clase *medicina*, y una clase *alergias*, la cual define a que medicinas es alérgico cada paciente. Note que en este ejemplo, tanto *paciente* como *medicina* corresponden a objetos etiquetados en el entorno, mientras que *alergia* no. En seguida mostramos un ejemplo de la definición de una clase.

```
<class name="medicine">
  <field>
    <string name="name"/>
  </field>
  <field>
    <date name="caducity"/>
  </field>
  <field>
    <string name="activesubstance" />
  </field>
</class>
```

```
</field>
</class>
```

Instancias

Una vez que se han definido las clases apropiadas para el escenario de la aplicación, se debe proveer información acerca de los objetos etiquetados. Esta tarea es realizada mediante la sección *instances* de nuestro lenguaje. Dentro de esta sección, ligamos el identificador de una etiqueta a un objeto real definiendo la clase a la que pertenece el objeto y proveyendo valores para los campos definidos en tal clase. Por ejemplo podemos instanciar una medicina de acuerdo al ejemplo de la sub sección anterior de la siguiente manera:

```
<instance epc="urn:upc:id:gid:10.1002.2"
class="medicine">
  <attribute>Aspirin</attribute>
  <attribute>2008-12-15</attribute>
  <attribute>Acetilsalicilic acid</attribute>
</instance>
```

El atributo *epc* es el id almacenado en la etiqueta del objeto que se está definiendo; este atributo es la liga entre las especificaciones y los objetos físicos. Es importante resaltar que los tipos de datos de los valores proporcionados para los atributos deben corresponder a los tipos de datos de los campos de la clase correspondiente.

Datos

En la sección *data* proveemos información adicional que es necesaria para definir los eventos, pero que no se refiere directamente a ningún objeto físico.

La sección de datos es similar a la sección *instances*, pero dado que en la sección *data* no se definen objetos tangibles, no proveemos el id de una etiqueta (atributo *epc*). Para el ejemplo de reconocer medicaciones alérgicas, es necesario saber que alergias sufre cada paciente; un ejemplo de tal información en tal especificación se presenta en seguida:

```
<data class="patientallergy">
  <attribute>Omar González</attribute>
  <attribute>Aspirin</attribute>
</data>
```

Cada definición de un dato debe corresponder a una clase previamente definida; ésta

correspondencia se define a través del atributo llamado *class*. Para nuestro ejemplo, debe existir una clase llamada *patientallergy* previamente definida en la sección *classes*, y tal clase debe contener dos campos de tipo string.

Locaciones

Para poder utilizar las observaciones al definir eventos, debemos considerar la posición de los lectores. Dentro de la sección *locations*, se definen los lugares relevantes en el entorno de cada aplicación. Para definir una locación, proveemos su nombre y un conjunto de identificadores de lectores. Al definir una locación como un conjunto de lectores, podemos manejar varios lectores, como una única fuente de observaciones. A continuación mostramos como definir locaciones.

```
<location name="room1">
  <reader name="192.168.1.2" />
</location>
<location name="corridor">
  <reader name="192.168.1.2" />
  <reader name="192.168.1.4" />
</location>
```

Como se muestra en el ejemplo, el lector con el identificador 192.168.1.2 es utilizado para definir dos locaciones diferentes, lo cual ilustra que el uso de lectores para definir locaciones no es exclusivo; podemos usar el mismo lector para definir dos o más locaciones.

Eventos

La sección *events* es la sección más importante del lenguaje porque es donde se utiliza la información de todas las secciones previas para definir los eventos de interés de la aplicación. Para cada evento se deben definir variables, fields y una condición.

Las variables de un evento proveen la información necesaria para definir tal evento. Hay tres tipos de variables: objetos etiquetados (que son definidos en la sección *instances*), información (que es definida en la sección *data*), y eventos más simples previamente definidos. Cada variable debe tener un nombre y un tipo: para objetos etiquetados y para información, el tipo debe ser el nombre de una clase; para eventos el tipo debe ser el nombre de un evento previamente definido.

Los campos (*fields*) de un evento almacenan información relacionada a cada ocurrencia de ese tipo de evento. Automáticamente, cada vez que un evento es reconocido, su nombre y su marca de tiempo es almacenada; sin embargo si especificamos campos para el evento, podemos almacenar mayor información relacionada a la ocurrencia del evento. Los campos obtienen sus valores a partir de las variables del evento. Como se menciono anteriormente los campos de un evento son útiles para definir eventos a partir de otros eventos. Por ejemplo, cada vez que se detecte una medicación se puede almacenar el nombre de la medicina y el nombre del paciente, como campos; después se podrían usar esos valores para reconocer sobredosis potenciales de una medicina para un paciente.

La condición de un evento es una expresión lógica la cual es evaluada por la capa de reconocimiento para determinar cuando un evento particular ocurre. Una condición se construye utilizando los operadores presentados en la sección 4. Por ejemplo, suponga que una medicación errónea debe detectarse en un entorno hospitalario. Para nuestro ejemplo, suponga también que las medicinas siempre las suministra una enfermera y que una medicación es errónea si se hace a un paciente alérgico o si el paciente ha recibido una transfusión durante de los cinco minutos previos a la medicación. La condición de este evento es:

```
<event name="wrongmedication">
  <var class="nurse" name="n1"/>
  <var class="medicine" name="m1"/>
  <var class="patient" name="p1"/>
  <var event="transfusion" name="t1"/>
  <fields>
    <string name="patient" value="p1.name"/>
    <string name="medicine" value="m1.name"/>
  </fields>
  <condition>
    <and>
      <all>
        <object>p1</object>
        <object>m1</object>
        <object>n1</object>
        <location>room1</location>
      </all>
      <or>
        <and>
          <sequence within="300">
            <occurrence>t1</occurrence>
          </sequence>
          <equal>
            <value>t1.patient</value>
            <value>p1.name</value>
          </equal>
        </and>
      </or>
    </and>
  </condition>
</event>
```

```
</and>
<exists type="allergicmedicine">
  <value>p1.name</value>
  <value>m1.name</value>
</exists>
</or>
</and>
</condition>
</event>
```

Note que para que la definición de este evento sea correcta, previamente deben estar definidas algunas clases (nurse, medicine, patient, and allergicmedicine), un evento más simple (transfusión), una locación (room1), e instancias para cada clase.

VI. RECONOCIMIENTO DE EVENTOS

En esta sección presentamos cómo la capa de reconocimiento realiza el reconocimiento de eventos. Presentamos las estructuras de datos requeridas y su funcionamiento en general. También describimos cómo la información fluye entre el middleware y la capa de reconocimiento a través de ALE, y la manera de importar definiciones de eventos.

Adquisición de Datos RFID

Una vez que la capa de reconocimiento ha recibido una especificación hecha con RFID-CEDL, debe usar ALE para definir qué datos obtener del middleware. La capa de reconocimiento debe establecer tres parámetros para lograr esto: lectores de interés, etiquetas de interés y el tipo de reportes. El conjunto de lectores de interés se define de acuerdo a las locaciones definidas en la especificación. El conjunto de etiquetas que reportará el middleware se define como el conjunto de etiquetas que se mencionan dentro de la sección instancias de la especificación. El tipo de reportes a generarse debe ser reportes diferenciales, esto es, información acerca de llegadas y salidas de objetos etiquetados respecto a los rangos de lectura de los lectores de interés.

Almacenamiento de Datos RFID

La capa de reconocimiento almacena la información proveniente del middleware en una tabla llamada *t_stance*. Cada fila en *t_stance* almacena el identificador de una etiqueta, el identificador de un lector, una marca de tiempo

inicial, y una marca de tiempo final. Cada fila en esta tabla indica que un objeto estuvo (o está) situado en una locación durante el periodo de tiempo comprendido entre las marcas de tiempo correspondientes.

Siempre que la capa de reconocimiento es notificada acerca de la llegada de un objeto al rango de lectura de un lector, se crea un registro con la marca de tiempo final en *null*; cuando la capa de reconocimiento es notificada acerca de un objeto abandonando el rango de lectura de un lector, la marca de tiempo final correspondiente es rellenada.

Almacenamiento del Historial de Eventos

Para evaluar *sequences* y *combinations*, es necesario mantener un historial de los eventos reconocidos. Éste historial se almacena en una tabla llamada *t_occurrence*. Cada fila en *t_occurrence* almacena el nombre de un tipo de evento, una marca de tiempo y una lista de valores correspondientes a los campos de cada ocurrencia del evento.

Manejo de definiciones de eventos

La capa de reconocimiento maneja las condiciones de los eventos como árboles de sintaxis, donde la raíz es el operador principal de la condición. La condición se descompone por niveles hasta las hojas, las cuales son sub expresiones de un único operador. El proceso de importar definiciones como árboles de sintaxis es directo dada la estructura de árbol inherente a la sintaxis XML. La figura 4 muestra el ejemplo del árbol de sintaxis correspondiente al evento presentado en la sección 5.5.

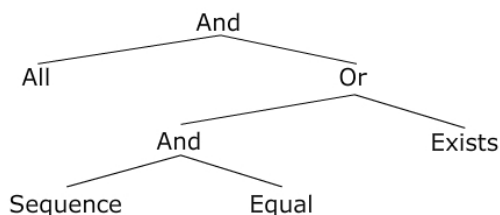


Fig. 4. Ejemplo de un árbol de sintaxis.

Reconocimiento de Situaciones

De acuerdo con nuestro modelo, el reconocimiento de eventos es disparado por los movimientos de los objetos. Supongamos que el objeto x cambia de posición. La capa de reconocimiento busca todos los árboles de sintaxis donde x es parámetro de algún operador de locación (*all*, *any* o *none*). Para cada árbol evaluado como verdadero, una ocurrencia del tipo de evento correspondiente se detecta y se crea la fila respectiva en *t_occurrence*. Cuando un evento de tipo e es reconocido, todos los árboles de sintaxis donde un evento de tipo e es parámetro de un operador de ocurrencia (*sequence* o *combination*) son evaluados. Si alguna otra ocurrencia es reconocida, se crea la respectiva fila en la tabla *t_occurrence*.

Ahora presentamos como la capa de reconocimiento evalúa cada operador.

Los operadores de locación son evaluados analizando el contenido de *t_stance*. $All(x_1, x_2, \dots, x_n, y)$ es evaluado como verdadero si para cada objeto x_1, x_2, \dots, x_n existe una fila con el lector igual a y y una marca de tiempo final *null*. $Any(x_1, x_2, \dots, x_n, y)$ es evaluado como verdadero si para por lo menos un objeto x_1, x_2, \dots, x_n existe una fila con el lector igual a y y una marca de tiempo final *null*. $None(x_1, x_2, \dots, x_n, y)$ es evaluado como verdadero si para ningún objeto x_1, x_2, \dots, x_n existe una fila con el lector igual a y y una marca de tiempo final *null*.

Las secuencias son evaluadas utilizando autómatas finitos no deterministas (NFA). Para cada secuencia se construye un NFA enlazando estados para elementos sucesivos en la secuencia. Cada transición es etiquetada con el tipo de evento correspondiente en la secuencia. El último estado, es el único estado de aceptación del autómata. La figura 5 ilustra el NFA a la secuencia ev_1, ev_2, ev_3 .

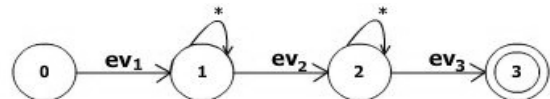


Fig. 5. Ejemplo del NFA para una secuencia.

Una vez que se construye el NFA correspondiente, el operador *sequence* es evaluado analizando el contenido de *t_occurrence*. Para evaluar la secuencia de eventos representada por $Sequence(ev_1, ev_2, \dots, ev_n, i)$ la capa de

reconocimiento filtra el conjunto de ocurrencia de eventos registradas en los últimos i segundos. Una vez filtrados los eventos se proveen como entrada del NFA ordenados por marca de tiempo. Todos los caminos desde el estado inicial al estado de aceptación son secuencias satisfactorias de eventos.

Una *combination* también se evalúa buscando en $t_{occurrence}$. Para evaluar la combinación de eventos representada por $Combination(ev_1, ev_2, \dots, ev_n, i)$ la capa de reconocimiento filtra el conjunto de de ocurrencias de eventos registrados dentro de los i segundos. El conjunto filtrado de ocurrencias se divide en clases de equivalencia de acuerdo al tipo de evento. Sean tales clases de equivalencia los subconjuntos EV_1, EV_2, \dots, EV_n . el conjunto de combinaciones satisfactorias es $EV_1 \times EV_2 \times \dots \times EV_n$. Si un tipo de evento es incluido más de una vez como parámetro de la combinación debe ser agregado el mismo número de veces en el producto Cartesiano.

VII. CONCLUSIONES

La representación y reconocimiento de eventos compuestos le permite a las aplicaciones manejar eventos compuestos en lugar de analizar información de bajo nivel. En este artículo, hemos presentado una plataforma para manejar eventos compuestos en sistemas RFID; hemos propuesto y justificado un lenguaje llamado RFID-CEDL para definir eventos compuestos. También mostramos cómo reconocer eventos utilizando información proporcionada por cualquier middleware que implemente la interfaz Application Level Events.

VIII. REFERENCIAS

- [1] EPCglobal, "EPCglobal Tag Data Standards Version 1.3.1", EPCglobal, march 2006.
- [2] EPCglobal, "The EPCglobal Architecture Framework", EPCglobal, September 2007
- [3] H. González, J. Han, X. Li, D. Klabjan, "Warehousing and Analyzing Massive

- RFID Data Sets", Proceedings of the 22nd International Conference on Data Engineerin, ACM, 2006, pp. 83.
- [4] EPCglobal, "The Application Level Events (ALE) Specification Version 1.0", September 2005.
- [5] David Luckham, "The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems", Addison Wesley, 2002.
- [6] F. Wang, S. Liu, P. Liu, Y. Bai, "Bridging Physical and Virtual Worlds: Complex Event Processing for RFID Data Streams", 10th International Conference on Extending Database Technology, Germany, 2006, pp.588-607.
- [7] N. Museux, J. Mattioli, C. Laudy, H. Soubaras, "Complex Event Processing approach for Strategic Intelligence", Proceedings of the 9th International Conference on Information Fusion, Italy, 2006, pp.1-8
- [8] E. Wu, Y. Diao, S. Rizvi, "High-performance complex event processing over streams", Proceedings of the 2006 ACM SIGMOD international conference on Management of data, USA, 2006, pp. 407 - 418
- [9] Y. Diao, N. Immerman, D. Gyllstrom, "SASE+: An Agile Language for Kleene Closure over Event Streams", UMass Technical Report, 2007.
- [10] N. Khoussainova, M. Balazinska, D. Suciu, "PEEX: Extracting Probabilistic Events from RFID Data", Proceedings of the 24th International Conference on Data Engineering, México, 2008.

***EncuentraCel* – sistema de localización y búsqueda de sitios de interés para Web y teléfonos celulares.**

Félix Mata¹, Enrique Bustos², Oriana Santillán³, Karina Zamudio⁴

^{1,2,3,4} *Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas, IPN.
Avenida Instituto Politécnico Nacional No. 2580, Colonia Barrio la Laguna Ticomán,
Delegación Gustavo A. Madero, CP. 07340 México D.F
mmatar@ipn.mx, migfel@sagitario.cic.ipn.mx*

Resumen

EncuentraCel es un sistema que permite localizar sitios de interés a usuarios de dispositivos móviles o de la Web. El sistema consta de una base de datos geográfica, una aplicación móvil para teléfonos celulares, y una aplicación Web-Mapping.

Se ofrecen servicios basados en localización, desplegando mapas con la ubicación de uno o más sitios de interés, tanto en versión Web como para teléfonos celulares. En la web se ofrecen funciones de acercamiento, desplazamiento, consultas geográficas e identificación de objetos geográficos. Mientras que en los dispositivos móviles se despliegan los mapas a través de un navegador WAP o con mensajes SMS.

El sistema cuenta con los servicios de consultas espaciales y de análisis espacial como el de buffer¹, obteniendo resultados en formato *raster* y *KML*². Básicamente, se resuelven tres necesidades: conocer la ubicación de una persona, encontrar un lugar específico y encontrar sitios de interés (centros comerciales, restaurantes) que están al alrededor de un usuario.

¹ La operación espacial de buffer, también es conocida como análisis de proximidad y su principal aplicación es poder definir las áreas de afectación que ocasiona un fenómeno geográfico.

² Keyhole Markup Language, formato para desplegar mapas del programa cliente *Google Earth*

Palabras clave

GPS, KML, Web-Mapping, GIS, WAP, Servicios basados en localización.

I. INTRODUCCIÓN

Hoy en día, la ubicación geográfica se procesa en sistemas de información geográfica (GIS) en programas de escritorio y en la Web. Recientemente, la ubicación también se ha procesado en dispositivos móviles, gracias a la convergencia del cómputo móvil y los sistemas de posicionamiento global (GPS). Esta convergencia ha generado nuevos sistemas y aplicaciones que facilitan diversas tareas a los usuarios de telefonía celular. Un ejemplo de ello son los servicios como la exploración y despliegue de mapas en teléfonos celulares y PDA's. Esto ha permitido realizar tareas que antes sólo estaban disponibles teniendo una conexión a Internet. Como muestra de ello, podemos mencionar, a los Servicios Basados en localización (Location Based Services - LBS) los cuales proveen información de ubicación específica para los usuarios móviles.

En este sentido, procesar la ubicación geográfica ha permitido generar nuevos tipos de búsqueda. Por ejemplo, la búsqueda de eventos (películas, obras teatrales) se ha extendido y mejorado al relacionarlas con la localización de los lugares (cines, museos, auditorios). De esta forma, al integrar estas funcionalidades en un sistema para usuarios móviles automatizaría y facilitaría tareas

de búsqueda de eventos y de lugares, permitiendo su acceso desde un dispositivo móvil, como lo es el teléfono celular.

En este sentido, los LBS ofrecen información accesible por dispositivos móviles aprovechando las redes móviles y las capacidades de localización de los teléfonos celulares [11]. Los LBS se definen en el internacional OpenGeospatial Consortium [4] como: Un servicio inalámbrico basado en IP, que procesa información geográfica para servir a un usuario móvil. Mientras que en [20] los LBS se definen como la intersección que existe entre los sistemas de telecomunicaciones, dispositivos móviles, internet, y los sistemas de información geográfica. Los LBS permiten a los usuarios móviles la búsqueda de sitios sin tener que especificar manualmente códigos postales u otros identificadores de localización para encontrar un sitio (hoteles, restaurantes). Esto es posible porque existen mecanismos para obtener la ubicación de un usuario o servicio, tales como los dispositivos GPS y los métodos de triangulación de la red celular. Las aplicaciones LBS de este tipo ayudan al usuario a reconocer y llegar un punto de interés dentro de un área determinada. Esto es muy útil en actividades de esparcimiento o entretenimiento.

Por otro lado, desplegar un mapa en un dispositivo de cómputo móvil (e.g. PDA) implica hablar de aplicaciones de Web-Mapping, las cuales consisten en implementar, generar y desplegar mapas en la Web. Estos sistemas en conjunción con los LBS requieren procesar atributos de referencia como la latitud y la longitud geográfica, así como referencias implícitas: direcciones, código postal, y nombres de calles. Algunos incluso resuelven tareas de carácter geográfico [18], como el servicio de localización geográfica que brinda *Geonames* [8] y el servicio de obtención de mapas de *Google Maps*.

Por lo tanto, realizar aplicaciones que aprovechen el amplio uso de los teléfonos celulares y las bondades de los LBS, ofrecerá a los usuarios una nueva forma de búsqueda, la cual extiende las

búsquedas actuales de la Web y que estarán disponibles desde un teléfono celular. Este trabajo pertenece a esta línea de investigación en donde convergen los sistemas GIS, Web-Mapping y de LBS.

II. TRABAJOS RELACIONADOS

Hoy en día, existen muchas soluciones para la visualización de mapas en la web. Por ejemplo, la ubicación de lugares desde la Web como es el caso de [23]. Sin embargo, este sitio no cuenta con el servicio para usuarios móviles. También existen una serie de servicios para teléfonos celulares, los cuales muestran la posición geográfica real de un usuario. Estos servicios utilizan el protocolo WAP y son mostrados en un mapa con formato adecuado para teléfono celular. Ejemplos de estos servicios son: *Ubícame* y *Ubicacel* ofrecidos por los proveedores *Iusacel* y *Movistar* en México. Estos servicios utilizan la red GSM o 3G para localizar vía el método de triangulación celular al usuario (le llaman GPS asistido). Este servicio trabaja con un control de localización, el proveedor de servicio le da la ubicación basándose en la intensidad de la señal de la torre móvil más cercana (para teléfonos sin características GPS). La primera desventaja de este enfoque es que puede ser bastante lento, ya que en el caso de la red GSM se utiliza el canal de voz, mientras que la segunda desventaja es el alto costo del servicio de datos. Otra desventaja es que la cartografía que ofrece solo está disponible para su despliegue pero no para descargarla. Es decir, la funcionalidad está dirigida al público en general y no a los especialistas GIS (se despliegan mapas en formato de imagen SVG³).

Continuando con el procesamiento de ubicación geográfica, revisemos ahora el caso de las aplicaciones de Web-Mapping, las cuales se han utilizado para diferentes fines. En el caso de México podemos obtener rutas y estimar el tiempo

³ *Standar Vector Graphics*, formato de imagen basado en XML,

de viaje entre dos puntos en el país, como es el caso de la aplicación Web de “*Traza tu Ruta de la SCT*” [22]. Dentro de este tipo de aplicaciones destaca *Google Maps*, ya que ha permitido que ciertas funciones de GIS sean utilizadas con facilidad por cualquier usuario de telefonía móvil o en la Web. La única desventaja con *Google Maps* es que no se puede procesar la cartografía que presenta (son imágenes) y no cuenta con toda la cartografía de México, limitando su uso en tareas de análisis y asistencia en toma de decisiones para la república mexicana.

Esto se puede solucionar utilizando la tecnología Web-Mapping ya que se pueden manejar y analizar estructuras de datos espaciales a través de Internet. La información geográfica se puede desplegar y manipular de forma muy similar a como se procesa con un GIS de escritorio. En Web Mapping se despliegan mapas en formato *raster* (por ejemplo, formato de imagen PNG) y en formato vectorial (como *shapefiles*). Lo cual permite mostrar las características de un fenómeno de una manera estática, y además añadir variables dinámicas que modelen fenómenos geográficos para su observación y análisis.

Por otro lado, procesar información geográfica en dispositivos móviles ofrece una nueva perspectiva para los GIS's que trabajan en un entorno móvil (Mobile GIS) [14]. Los Mobile GIS se utilizan como asistentes en la obtención de datos geográficos a nivel de campo [3, 15, 16]. Y también en tareas como la ingeniería en construcción, el suministro de agua, entre otras.

Por otra parte, un GIS almacena información sobre el mundo como una colección de capas temáticas que describen la geografía. Este concepto es simple, pero extremadamente poderoso y versátil que ha probado ser invaluable para la solución de muchos problemas, como el rastreo de vehículos de reparto y en tareas de planificación.

La integración de aplicaciones WAP con sistemas de información geográfica, ha simplificado las tareas que comúnmente realizan especialistas GIS, pero ahora desde un dispositivo móvil. Un ejemplo de estos, son las tareas de localización geográfica, y para el caso de usuarios neófitos en GIS permiten asociar información descriptiva a ubicaciones geográficas de forma intuitiva (por ejemplo, mi café favorito). Esto es posible gracias a los GIS móviles, que fungen como una extensión de los GIS desde la oficina hacia al campo.

Al integrar todas estas tecnologías se extienden y amplían las funcionalidades de los sistemas GIS. *EncuentraCel* es una aplicación que extiende y amplía las funcionalidades GIS al ponerlas al alcance del usuario no especialista en GIS. A continuación se muestra en la figura 1, la arquitectura del sistema para encontrar sitios de interés.

III. ENCUENTRACEL: ARQUITECTURA GENERAL

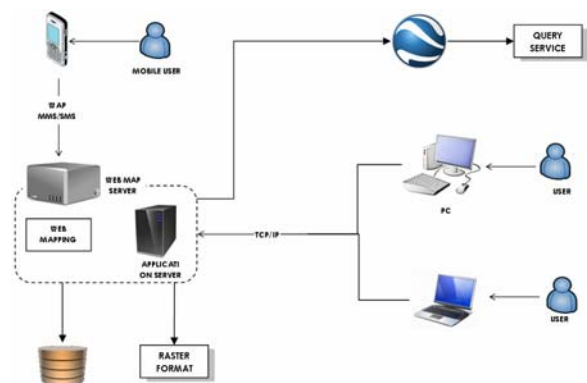


Figura 1. Arquitectura general de *EncuentraCel*

Como puede observarse en la Figura 1, el sistema funciona cuando un cliente (teléfono celular o computadora) hace una petición, que se procesa en el servidor, extrae la información necesaria de la base de datos geográfica. La información extraída se envía al servidor de aplicaciones para construir un mapa en formato de imagen PNG o

SVG. El siguiente paso es presentar el mapa en una interfaz web con las herramientas básicas para la manipulación del mapa. Este proceso es realizado por el módulo Web Mapping que incrusta en una página Web un mapa en formato de imagen, y lo configura de manera que el usuario visualiza la información de forma ordenada permitiéndoles realizar nuevas consultas sobre la base de este mapa.

Adicionalmente, *EncuentraCel* genera archivos KML para poder procesarlos y visualizarlos en el programa *Google Earth*. Esto con el objetivo de utilizar los datos, en otras tareas de análisis espacial facilitando interoperabilidad entre aplicaciones. Esta última función está orientada hacia especialistas GIS. El archivo KML despliega la ubicación y la información descriptiva de un sitio concreto. Por ejemplo, muestra un restaurante, su sitio web, y detalles como los precios, teléfonos, dirección completa, entre otras. Estos datos están orientados a satisfacer las consultas cuyos usuarios cumplan el perfil de un usuario GIS (los usuarios definen un perfil al ingresar por primera vez al sistema y con base en éste se procesan sus peticiones).

A continuación se explicaran los módulos que componen al sistema. En la sección 3.1, se explica el módulo principal de Web-Mapping, detallando la construcción de los datos, así como la publicación para clientes WEB y WAP. Mientras la sección 3.2, se encarga de describir la construcción de la base de datos geográfica. En la sección 3.3, se explica el módulo móvil, así como el proceso de extracción de las coordenadas geográficas a partir de la ubicación actual del usuario, para su uso en el módulo de Web-Mapping y WAP.

3.1 Módulo Web-Mapping

El módulo Web-Mapping es una aplicación que realiza la gestión y el despliegue de mapas, a partir de las consultas y la codificación del mapa en formatos de imagen como JPG, PNG o SVG. Se utiliza el servidor de aplicaciones de *ASPMAP*

[23], el cual trabaja en conjunto con un servidor Web. En particular se utilizaron dos servidores Web, el primero es el servidor IIS para las consultas Web y el segundo es Tomcat quien atiende las consultas WAP.

La aplicación solicita la información a procesar, la cual está organizada en archivos denominados capas de datos. Por ejemplo, si tenemos calles y restaurantes, necesitamos una capa de calles y otro de restaurantes. Entonces, es necesaria procesar las capas que constituyen el mapa, para su despliegue en formato de imagen JPG o PNG.

Este proceso se conoce como la digitalización del mapa. Esta fase consta de tres etapas, la primera es obtener la imagen, la segunda se encarga de la georeferenciación de imágenes y, la tercera es la digitalización de la imagen. Para el primer caso se uso *Google Earth*, mientras que para el segundo y el tercer paso, se utilizó el programa *ArcView*⁴.

Para el primer paso, se obtiene una imagen del área de Zacatenco usando el programa *Google Earth*, para ello se llevó a cabo la unión y conversión a formatos JPG y TIF. La extracción de las imágenes se realizó en dos etapas, la primera consistió en establecer los puntos que definen el lugar en este caso una superficie de 6,20 km². La segunda, es la combinación de cada uno de los puntos guardados en el paso anterior, entonces se obtiene una imagen completa de toda la zona. El resultado se muestra en la Figura 2.

⁴ www.esri.com



Figura 2. Imagen satelital de la zona de Zacatenco.

La Figura 2 muestra una parte de la zona de Zacatenco, que se utilizó como base para el sistema. Esta imagen debió ser georeferenciada para vincular las coordenadas obtenidas por el dispositivo GPS con el punto correspondiente en la imagen. Es decir, que sea capaz de indicar la localización de un sitio en función de sus coordenadas geográficas.

3.1.1 Proceso de digitalización

El tercer y último paso consiste en digitalizar las imágenes para su manipulación y despliegue en los clientes WEB y WAP. Este proceso permite el procesamiento de la solicitud de acuerdo a los entornos Web y WAP. La Figura 3 muestra este resultado.

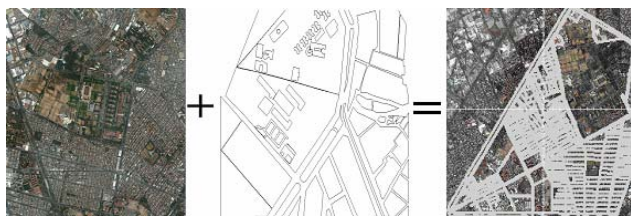


Figura 3. Proceso para la digitalización de mapas.

El mapa final consta de cinco capas principales, que son las siguientes:

Capa de la imagen del mapa: Se refiere a la

imagen que muestra el área de cobertura total en la que nuestro sistema ofrece sus servicios.

Capa de zonas: Se refiere a la representación vectorial de las zonas de polígonos.

Capa de calles: Es una capa de objetos de tipo línea que representan cada una de las calles en la zona.

Capa de Sitios de Interés: Esta capa se utiliza para representar diversos puntos de sitios de interés, maneja 4 categorías diferentes: escuelas, bancos, restaurantes y tiendas.

Capa de Buffer: Esta es otra capa de puntos que se han colocado en 10 intersecciones de avenidas principales para identificar las áreas de afectación del tráfico vehicular.

Como se puede ver, la primera capa se compone de una imagen obtenida a través de Google Earth, y referenciada usando el programa ArcView a través de la herramienta de georeferenciación. El resto de las capas se construyeron teniendo como base esta imagen. Cada uno de los elementos de las capas se asocia con un registro en una tabla de datos que el usuario puede consultar. Estos registros contienen información sobre un elemento de una calle, una manzana, un sitio de interés o una zona de influencia.

3.1.2 Funciones de navegación

Se ofrece cuatro funciones básicas en el manejo de la ruta, estos se definen a continuación

Función Zoom: Permite realizar acercamientos o alejamientos en una zona del mapa. La primera es la solicitud de una nueva imagen que cubre un área más pequeña con mayor detalle, en el segundo, esto se invierte, es decir, hay una solicitud de una imagen de una zona de mayor tamaño, pero con un menor nivel de detalle.

Función Pan: Esta función le permite desplazarse por el mapa, en primer lugar se obtienen las

coordenadas que el usuario selecciona y luego hace la solicitud de una nueva imagen centrada en las coordenadas obtenidas.

Función de Identificación: Recupera datos descriptivos con base en la posición de un objeto geográfico. Cuando el usuario selecciona un punto sobre el mapa obtendrá un resultado en una tabla. En otras palabras, los datos que están asociados con ese punto geográfico se muestran en una tabla.

3.1.3 Construcción de archivos KML

El formato KML (*Keyhole Markup Language*) es una implementación de XML para el despliegue de objetos geográficos en los programas *Google Maps* y *Google Earth*. Se procesan similarmente a cómo los navegadores Web procesan los archivos HTML y XML. De este modo, *Google Earth* y *Google Maps* actúan como navegadores de archivos KML. Un fragmento de archivo KML generado por el sistema se muestra en la Figura 4.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.2">
  <Placemark>
    <name>UPIITA </name>
    <description>Prueba numero 1
    </description>
    <Point>
      <coordinates>19.826752, -99.179045
      </coordinates>
    </Point>
  </Placemark>
</kml>
```

Figura 4. Ejemplo de archivo KML

Para la generación de archivos KML recuperamos las coordenadas en píxeles cuando el usuario hace clic en el punto que desea generar el archivo, una vez que la solicitud fue enviada al servidor, esto lo convierte en coordenadas geográficas para escribir el archivo. Al terminar la escritura en el archivo, el usuario obtiene un archivo con extensión *kml*. La Figura 5 muestra el proceso.

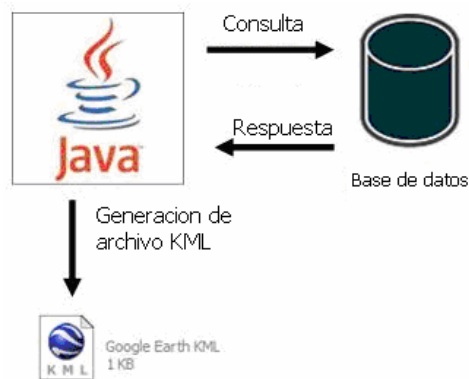


Figura 5. Generación de una consulta KML

La figura 5 muestra la generación de un archivo KML para desplegarlo en el programa Google Earth.

3.1.4 Generación de Mapa en formato SVG

Nuestro sistema utiliza el formato SVG el cual es un lenguaje para describir gráficos en dos dimensiones usando XML. Este formato fue elegido porque con él es posible generar imágenes SVG de forma dinámica, con base a la última imagen del mapa que le aparece al usuario. Además de reducir el tamaño en bytes, cada imagen ocupa cerca de 4 KB. Lo cual lo hace ideal para su despliegue en dispositivos móviles y para su transmisión en redes de ancho de banda limitado, siendo ideal para navegadores WAP o para incluirlo en mensajes SMS.

3.2 Módulo de la base de datos geográfica

La creación de la base de datos geográfica se realiza en el momento de la digitalización de la imagen del mapa, a partir de la cual se crearon los archivos SHP (Shapefile, formato propietario de ESRI), con datos sobre las distintas capas de información utilizadas: Calles, Sitios de interés y de Buffer. Los shapefiles son archivos ampliamente utilizados por muchas aplicaciones relacionadas con la geolocalización y la representación. Es un formato de archivo compuesto por tres tipos de archivos: shp, shx, y dbf. Los dos primeros son fundamentales y se encuentran relacionados para almacenar las

primitivas de representación (puntos, líneas, polígonos) junto con sus coordenadas geográficas. Por otra parte, los archivos con la extensión "dbf" son archivos complementarios que contienen la información descriptiva asociada a cada objeto geográfico. Esto se ilustra en la Figura 6.

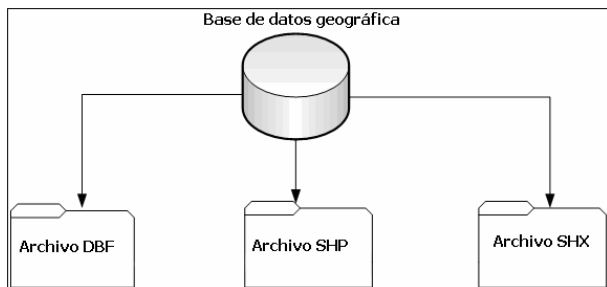


Figura 6. Base de Datos Geográficos utilizando archivos de forma

El archivo *shx* almacena índices (identificadores) entre los datos y los objetos geográficos. Es decir, asocia coordenadas geográficas con información descriptiva.

El archivo *dbf* contiene las tablas con los datos que se introdujeron con el fin de proporcionar una descripción los objetos geográficos. Por ejemplo, los atributos de la capa de calles, el nombre de la calle y la colonia a la que pertenece.

3.3 Módulo móvil

En éste módulo se desarrolló una aplicación móvil para extraer las coordenadas del dispositivo GPS, que se almacenan en formato GML (Geographic Markup Language). En este formato los datos de localización son separados por comas, de esta manera podemos conocer las diferentes coordenadas y almacenarlas.

Se utilizó el teléfono celular HP iPAQ 6515, el cual no incluye una máquina virtual de java, por lo que fue necesario encontrar una máquina virtual compatible con el teléfono celular. El nombre del proyecto asociado a esta máquina virtual es JEODE. Para procesar las coordenadas, se construyó un programa que extrae las coordenadas obtenidas por el GPS. Una vez obtenidas las

coordenadas, la aplicación se encarga de enviarlas desde el teléfono celular hacia el servidor usando una conexión WAP o en un mensaje SMS dirigido a un número telefónico asociado al servidor. En el momento en que el servidor de mapas recibe las coordenadas, este extrae los datos de la base de datos geográfica de acuerdo a los parámetros de la consulta y genera la imagen del mapa, enviándole al cliente una imagen del mapa, el cual muestra la ubicación actual del usuario.

IV. FUNCION DE ENVIO SMS

El servicio de mensajes SMS es un servicio designado para enviar avisos o promociones de un sitio de interés. Por ejemplo, si un usuario busca una tienda o artículo, y éstas tienen promociones vigentes para ciertos artículos, entonces se enviarán mensajes SMS con la información de la promoción. El SMS es enviado con base en la información que el usuario llenó en su perfil la primera vez que ingresó al sistema. Este proceso es mostrado en la Figura 7.

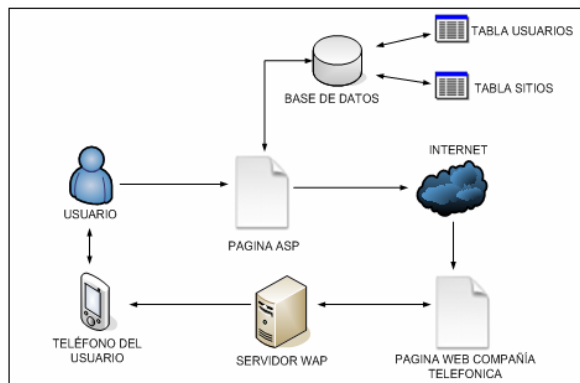


Figura 7. Envío de Servicio de Mensajes Cortos (SMS)

Como se aprecia en la figura 7, el sistema funciona de la siguiente manera: cuando un usuario busca un sitio de interés (por ejemplo, un centro comercial) el sistema le regresa los sitios encontrados. Si dentro de los sitios existen algunos que ofertan promociones, entonces se le envía un SMS al teléfono del usuario indicándole la promoción. El usuario puede elegir en su perfil si desea recibir promociones o avisos.

V. RESULTADOS

Las pruebas fueron generadas usando clientes WAP y WEB, en particular usando el teléfono Sony Ericsson w800. En la figura 8 se muestra la interfaz WEB así como la interfaz WAP.

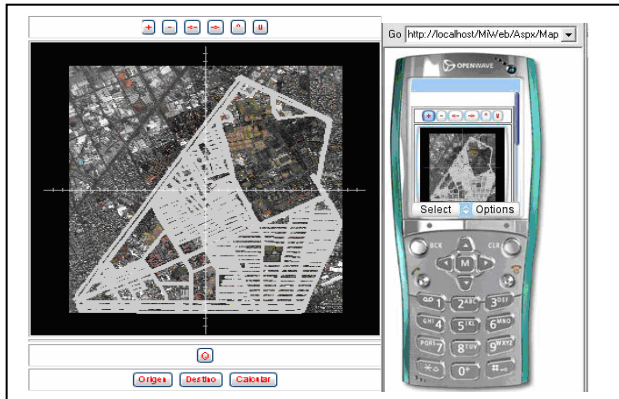


Figura 8. Interfaz de usuario para WEB y WAP

La figura 8 muestra el mapa y los controles para el manejo de éste, aprovechando las ventajas que tiene el navegador WAP. Una vez que el usuario accedió desde el navegador WAP en su teléfono celular y escribe la dirección específica en el sistema *EncuentraCel*, el usuario recibe un mapa, el cual está listo para ser manipulado usando las herramientas que se encuentran en él. La figura 9 muestra los acercamientos y alejamientos, desplazamientos e información relacionada con el área de Zacatenco.



Figure 9. Punto de interés en Zacatenco con acercamiento y alejamiento.

Para la búsqueda por área de interés es seleccionado el sitio y criterio de búsqueda. El resultado de la búsqueda de sitios activa la capa de sitios (puntos) y los resultados se muestran en

color rojo (puntos) tal y como se aprecia en la figura 10. Cabe señalar que es posible mostrar la información asociada a cada punto seleccionando la función “Identificar” y haciendo clic en el punto deseado.

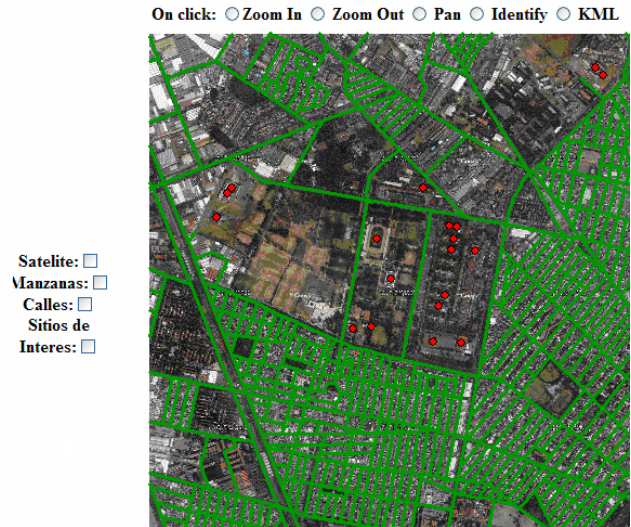


Figure 10. Resultado de la búsqueda por categoría “Estudiantes”

Por otra parte, la figura 11 muestra el despliegue del archivo KML dentro del programa *Google Earth*. Cuando en *EncuentraCel* se formuló la consulta {"UPIITA"} que es una escuela del IPN el resultado muestra la ubicación de dicha escuela.



Figure 11. KML resultado de la consulta UPIITA.

A continuación en la figura 12, se muestran los resultados para la función buffer. Esta función permite delimitar una zona de afectación para un fenómeno geográfico particular. En este caso, se

despliegan zonas afectadas por el tránsito vehicular. El escenario de pruebas se refiere a las áreas y avenidas afectadas por el tráfico vehicular en la delegación Gustavo A. Madero, que colinda con las delegaciones Cuauhtémoc y Venustiano Carranza, así como con los municipios de Tlalnepantla, Tultitlán, Coacalco y Ecatepec. La afluencia vehicular de estos municipios hacia la delegación Gustavo A. Madero provoca que muchas avenidas se congestionan en ciertas horas del día (alto tráfico). La extensión del área afectada es definida por una función que determina el grado de concentración del tráfico. Las pruebas para el sistema de Web Mapping fueron realizadas usando datos propuestos por las principales avenidas. La Figura 12 muestra un círculo azul (buffer), lo que representa la zona de afectación en las avenidas de Politécnico y Montevideo, y Ticomán.



Figura 12. Áreas afectadas en la esquina avenida politécnico - montevideo y la esquina de las avenidas politécnico – ticoman

La figura 12 muestra el resultado del uso de la función buffer, esta es una de las herramientas principales de cualquier GIS, en este caso particular esta función esta implementada sobre las herramientas WEB, lo que incrementa sus ventajas al no requerir una instalación de un GIS. Además de que pueden ser accedida y utilizada desde un teléfono celular.

VI. CONCLUSIONES Y TRABAJO FUTURO

Presentamos *EncuentraCel* un sistema de Geolocalización para dispositivos móviles. *EncuentraCel* ofrece servicios basados en la ubicación del usuario. Adicionalmente agrega la función de análisis espacial de *Buffer*, la cual indica una zona afectada por un fenómeno geográfico particular. En este caso, se midió el tránsito vehicular en la zona de Zacatenco. *EncuentraCel* también permite identificar el lugar donde el usuario está ubicado y los puntos de interés alrededor de él. Otra de sus funcionalidades consiste en localizar sitios mediante el nombre o categoría a la que pertenecen. Como valor agregado cuenta con un módulo que permite el envío de promociones u ofertas a través del servicios de mensajes SMS.

Existen diversas características que distinguen nuestro sistema y lo hacen una buena opción en comparación con otros sistemas que ofrecen soluciones de Geolocalización. Algunas de estas características son las siguientes:

- *EncuentraCel* no solo permite desplegar información geográfica, sino también recuperar datos con base en su localización.
- Comparado con servicios como *Ubicacel* o *Ubícame*, que pueden ser servicios muy lentos cuando utilizan el canal de voz, aunado al alto costo del servicio de datos. *EncuentraCel* ofrece la posibilidad de descargar la cartografía. Lo cual permite realizar análisis espacial a especialistas GIS.
- La integración con otras fuentes de datos es transparente ya que *EncuentraCel* utiliza una implementación de XML (el formato SVG) para desplegar los mapas en dispositivos móviles.
- Al permitir un servicio basado en consultas el trabajo puede extenderse para recuperar

información geográfica con un enfoque diferente al de palabra clave.

- Actualmente se está trabajando para integrar un motor semántico que procese las consultas basándose en la semántica espacial.

Hay varias opciones que podríamos explorar para hacer nuestro sistema más eficiente y proporcionarle una mayor funcionalidad. Actualmente, nuestro sistema está basado en el uso exclusivo de la tecnología GPS, por lo que se investigan otras posibilidades alternas para la localización. Se está trabajando en una versión usando redes Bluetooth para áreas geográficas de menor tamaño. Finalmente, cabe señalar que el área de cobertura fue elegida para probar el funcionamiento del prototipo. Pero, para dotar a nuestro sistema de funcionalidad real esta área debe ampliarse para abarcar una zona de mayor extensión geográfica. También esta tarea se está llevando a cabo como parte del Sistema *Turicel* que se realiza en convenio con el IPN y el Instituto de Ciencia y Tecnología del DF (ICyTDF).

AGRADECIMIENTOS

Los autores de este artículo agradecen al Instituto Politécnico Nacional, a la comisión de operación y fomento a las actividades académicas del IPN (COFAA), a la secretaria de investigación y posgrado (SIP) y al instituto de ciencia y tecnología del DF (ICyTDF) por el apoyo brindado a este trabajo; así como a los revisores de este trabajo por sus útiles comentarios.

REFERENCIAS

- [1] Dru, M-A., Saada, S., 2001. Location-based mobile services: The essentials. Alcatel Telecommunications Review, first quarter, 71-76.
- [2] Edwardes, A., Burghardt, D., Weibel, R., WEBPARK – Location based Services for Species Search in Recreational Areas. In: Proceedings of the 21st International Cartographic Conference. 2003, Durban, South Africa.

[3] Espinoza, F., Persson, P., Sandin, A., Nyström, H., Cacciatore, E. and Bylund, M., 2001. GeoNotes: Social and Navigational Aspects of Location-Based Information Systems. In: Abowd, Brumitt and Shafer, ed. Ubicomp 2001: Ubiquitous Computing., International Conference, September 30 – October 2, Atlanta, Georgia.

[4] Open Geospatial Consortium (OGC), 2005. Open Location Services 1.1.

[5] Persson, P., Espinoza, F., Fagerberg, P., Sandin, A. and Cöster, R., GeoNotes: A Location-based Information System for Public Spaces. In: Kristina Höök, David Benyon and Alan Munro, ed. Readings in Social Navigation of Information Space. 2002. Berlin: Springer, 151-173.

[6] Reichenbacher, T., Adaptive Methods for Mobile Cartography. In: Proceedings of the 21st International Cartographic Conference ICC: Cartographic Renaissance, 10-16th August 2003, Durban, South Africa.

[7] Reichenbacher, T., Mobile Cartography - Adaptive Visualisation of Geographic Information on Mobile Devices. 2004 (PhD).

[8] Schiller, J. H., Mobile Communication. Addison Wesley. Schilit, B., Adams, N. and Want, R., 1994. Context-aware computing applications. In: Proceedings of IEEE Workshop on Mobile Computing Systems and Applications. 2003, Santa Cruz, California., 85- 90.

[9] Timpf, S. and Kuhn, W., Granularity Transformations in Wayfinding. 2003. In: C. Freksa, W. Brauer, C. Habel and K.F. Wender (eds). Spatial Cognition III. Lecture Notes in Artificial Intelligence. Berlin- Heidelberg: Springer Verlag, 77-88.

[10] Tomtom, 2005. Car Navigation [online].

[11] Varrantaus, K., Markkula, J., Garmash, A., Terziyan, Y.V., Developing GIS-Supported Location- Based Services. In: Proc. of WGIS'2001 – First International Workshop on Web Geographical Information Systems., Kyoto, Japan.

[11] Linbing Ma, Jianya Gong, ChunSen Zhang, Research on an application solution and key technology of mobile GIS. http://www.isprs.org/commission2/proceedings/paper/057_121.pdf

[12] Liu Yong, Li Qing Quan, Xie Zhi Ying, Wang Chong, Research of mobile GIS application based on handheld computer. <http://www.isprs.org/commission4/proceedings/pdfpapers/475.pdf>

[13] Fangxiong W, Zhiyong J, Research on a distributed architecture of mobile GIS based on WAP. <http://www.isprs.org/istanbul2004/comm2/papers/220.pdf>

[14] North K, Field information systems for managing your assets, engineering the benefits of geographical information systems, 1997. IEEE Colloquium, 6/1-6/7. 5.

[15] Wenbin Z, Dengrong Z, the development research and application perspectives of GIS based on mobile computing (2003). Remote Sensing Information.

[16] Bin Jiang and Xiaobai Yao, Location-based services and GIS in perspective. Computers, Environment and Urban Systems, Volume 30, Issue 6, November 2006, Pages 712-725

[17] Ubicacel, servicio de localización celular de IUSACELL mediante triangulación celular, <http://www.iusacell.com.mx/3G/Ubicacel/>

[18] Available: <http://www.geonames.org/>

[19] M. Tsou, Z. Peng: "Internet GIS: Distributed Geographic Information Services for the Internet". John Wiley and Sons, 2003

[20] Narushige Shiode, Chao Li, Michael Batty, Paul Longley, David Maguire, The impact and penetration of location-based services. Centre for Advanced Spatial Analysis, paper 50, <http://eprints.ucl.ac.uk/246/1/Paper50.pdf>

[21] [<http://www.esri-es.com/index.asp?pagina=501>].

[22] Pagina Web de traza tu ruta de la secretaria de comunicaciones y transporte, http://aplicaciones4.sct.gob.mx/sibuac_internet/ControllerUI?action=cmdEscogeRuta

[23] Sitio de búsqueda de sitios, www.xaqui.com

[24] ASPMAP, Componentes Web-Mapping, <http://www.vdstech.com/aspmap.htm>

FORMACIÓN DE ROBOTS BASADA EN INTERACCIONES LOCALES

Miguel A. Sánchez-Acevedo, Gustavo A. Torres-Blanco, Salvador Jauregui-Ortiz, Ernesto López-Mellado, Félix Ramos-Corchado

CINVESTAV Unidad Guadalajara, Av. Científica 1145, Col. El Bajío. 45010 Zapopan, Jal. México
Email: {sancheza, sjauregu, gtorres, elopez, framos}@gdl.cinvestav.mx

RESUMEN

En este artículo se presenta un algoritmo para establecer una formación de robots móviles. La formación es establecida utilizando principios de auto-organización. Un grupo de robots móviles es auto-organizado en una formación de cuña. Cada robot móvil es capaz de comunicarse con sus vecinos con el fin de identificar su posición en la formación. La formación es balanceada cuando se pierde el balance debido a la incorporación de nuevos elementos. Para validar la propuesta, el algoritmo es implementado en robots NXT.

Palabras clave: Formación de robots móviles, principios de auto-organización, robots NXT.

I. INTRODUCCIÓN

El establecimiento de una formación de robots móviles trae como consecuencia una estructura robusta para realizar diversas tareas. Con esta estructura, las tareas más complejas pueden ser distribuidas en el grupo, requiriendo menos capacidad de procesamiento por parte de cada robot. Dentro de las aplicaciones que pueden ser realizadas por un grupo de robots móviles tenemos: exploración, operaciones de rescate, distribución de sensores, entre otras. Los principios de auto-organización reducen la necesidad de conocimiento global acerca del estado de la formación debido a que la formación se obtiene como resultado de interacciones locales [1], [8], [9].

Varias propuestas han sido presentadas con el fin de obtener una formación de robots. Algunos trabajos están basados en la dinámica de los robots con el fin de mantener su posición dentro de la formación [4], [5], [14]. Por otra parte, se ha obtenido el control de la formación a través de interacciones locales [2], [6], [7], [11], [13], [15]. En este trabajo la formación se obtiene como resultado de la interacción entre robots vecinos;

este comportamiento está basado en los principios de auto-organización observados en la naturaleza [3], [10], [12], [16].

Una aplicación interesante, la cual está basada en la dinámica de los robots, es presentada en [4]. En ese trabajo se define una entrada no lineal de control con retroalimentación para rastrear la velocidad y se definen leyes de control para mantener la distancia con respecto al líder. La distancia de separación se mide a partir de la parte trasera del líder para evitar colisiones. Sin embargo, en ese trabajo solo se presentan las leyes de control necesarias para mantener la posición asignada en la formación.

La propuesta presentada en [15] muestra como se puede obtener una formación de robots móviles utilizando interacciones locales. La formación crece dinámicamente a partir de robots independientes hasta que se obtiene una formación compleja. Cada robot es inicializado con la forma y el tamaño de la formación a crear. El líder periódicamente inicia un protocolo de descubrimiento para determinar el estado global de la formación. Cuando se detecta que la formación no está balanceada, el líder inicia el proceso de balanceo. La propuesta presentada en este trabajo difiere de [15] en que los robots no necesitan saber cuántos robots hay en la formación ni quien es el líder.

Con el fin de eliminar la necesidad de un líder que decida el objetivo que cada robot tiene que seguir, la propuesta presentada en [10] introduce el término de "plantillas locales". Estas plantillas codifican información en múltiples regiones sectoriales para generar enlaces virtuales entre vecinos. Estos enlaces virtuales son detectados y seguidos por otros robots. Sin embargo, solo se pueden agregar nuevos robots a la formación en

lugares donde el enlace virtual no esté ocupado. El balanceo de la formación no se permite en este esquema.

El objetivo de este trabajo es organizar múltiples robots en una formación de cuña sin coordinación centralizada y usando interacciones locales. La formación es establecida utilizando robots NXT. Cada robot ejecuta el mismo controlador para interactuar con sus vecinos. La formación es establecida como resultado de estas interacciones. Se pueden agregar nuevos robots a la formación en cualquier momento y en cualquier lugar.

La descripción del hardware de los robots utilizados en este trabajo es presentada en la sección II. La sección III presenta una descripción del algoritmo ejecutado por cada robot con el fin de establecer la formación. La sección IV describe la implementación utilizando robots NXT. Los resultados se muestran en la sección V. Finalmente las conclusiones obtenidas se presentan en la sección VI.

II. DESCRIPCIÓN DEL HARDWARE

Para llevar a cabo la formación de robots móviles se utilizaron los robots Mindstorms NXT de Lego. Cada robot ha sido diseñado para proveer movilidad básica y permitir interacciones locales. Cada robot está equipado con una brújula la cual determina la orientación del robot. Para determinar la distancia de un robot con respecto a otro se utiliza un sensor ultrasónico. Dos llantas con un radio de 56mm se utilizan para permitir que el robot se mueva en el entorno. Una tercera llanta de 24mm es utilizada para proveer estabilidad al robot. En la Figura 1 se muestra una fotografía del robot utilizado.

Con el fin de llevar a cabo la comunicación entre los robots, se utilizan los dispositivos Bluetooth integrados en los robots. Cada robot se mantiene en un estado de espera para recibir conexiones entrantes. Cada robot mantiene una lista de todas las direcciones de los robots que están participando. Cuando un robot desea comunicarse con otro, se establece la conexión con el robot deseado, envía los mensajes y cierra la conexión. Una vez que la conexión ha sido cerrada, el robot está listo para recibir nuevas conexiones entrantes. Debido a las limitaciones de los dispositivos Bluetooth de los robots NXT no es

posible establecer una comunicación bidireccional en la misma conexión. El robot que necesite enviar un mensaje primero establece la conexión, envía el mensaje y cierra la conexión una vez que el mensaje fue enviado.

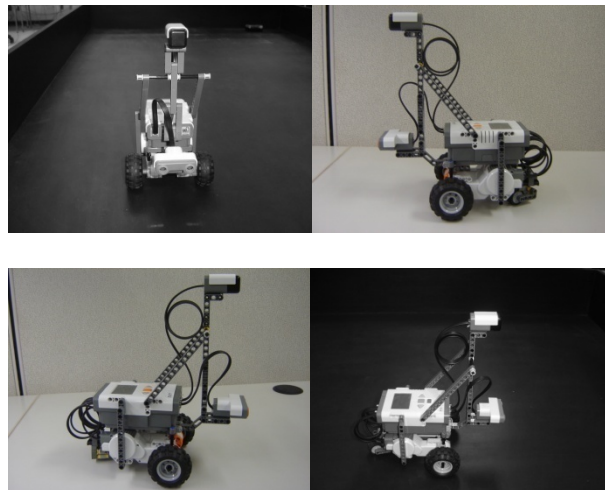


Figura 1. Robot Mindstorm NXT

El comportamiento de cada robot depende de los mensajes recibidos de sus vecinos y el estado actual del robot. En cada mensaje se envía la orientación y la dirección de los robots vecinos. Esta información es utilizada para identificar el curso a seguir en la formación. Las direcciones recibidas se utilizan para actualizar la información de los vecinos del robot.

En la siguiente sección se describen los mensajes y las acciones requeridas para establecer la formación de cuña. Cada vez que un robot recibe un mensaje de su vecino, el decide la acción a realizar y los mensajes que tiene que enviar a sus vecinos de acuerdo a la información contenida en el mensaje recibido. Este comportamiento individual trae como consecuencia la formación del grupo de robots móviles.

III. FORMACIÓN EN CUÑA

Debido a que no se mantiene información acerca del número de robots en la formación, la formación es configurada conforme nuevos robots son integrados a la formación. Para iniciar la formación, un robot es inicializado con la dirección a seguir y este robot toma el rol de líder. Los otros robots tratan de encontrar a algún robot que se encuentre en la formación. Una vez que se encuentra a un robot en la formación, el nuevo robot inicia un

intercambio de mensajes con el fin de determinar su posición en la formación.

La posición del robot en la formación es determinada de acuerdo al estado del robot vecino. Para determinar el estado global del robot se utilizan cinco variables de estado: Rol(Líder, Seguidor), EstadoFormación(Libre, EnFormación), Alineación(Derecha, Izquierda), RolVecino(Líder, Seguidor), y Balance(Balanceado, SemiBalanDer, SemiBalanIzq, DesBalanDer, DesBalanIzq). De acuerdo al estado del robot y el mensaje recibido de su vecino, el robot ejecuta ciertas acciones con el fin de obtener su posición en la formación. En algunos casos se envía un nuevo mensaje a los vecinos para mantener la forma de la formación.

Las acciones que un robot puede realizar y las restricciones que tienen que se tienen que satisfacer se describen en la tabla 1.

MensajeRec	Estado	MensajeTrans	Acciones
Robot Encontrado	Libre	Pedir Información	B
Pedir Información	Líder, EnFormación	Respuesta robot	B- SA
Pedir Información	Seguidor, EnFormación, Derecha	Respuesta robot	C- SA
Pedir Información	Seguidor, EnFormación, Izquierda	Respuesta robot	D- SA
Respuesta robot	B- Libre	L-Avanzar, R-Mover	LNSA, RNRN, MR, SL, SR
Respuesta robot	C- Libre	L-Avanzar, R-Mover	LNSA, RNRN, MR, SL, SR
Respuesta robot	D- Libre	R-Avanzar, L-Mover	RNSA, LNLN, ML, SR, SL
L-Avanzar	EnFormación	L-Avanzar	F, RNSA, SL
R-Avanzar	EnFormación	R-Mover	F, LNSA, SR
R-Mover	EnFormación	R-Mover	MR, LNSA, SR
L-Mover	EnFormación	L-Mover	ML, RNSA, SL

Tabla 1. Acciones a ser realizadas por los robots con el fin de integrar un nuevo robot.

Puesto que nuevos robots son integrados a la formación en el lugar donde un robot ya formado fue encontrado, el robot cuyo rol es el de líder inicia el proceso de balanceo cada vez que la formación pierde el balance. Cada vez que un robot se integra a la formación, un mensaje es retransmitido hasta que el líder lo recibe; una vez que el líder se entera que un nuevo robot ha sido integrado a la formación, el líder actualiza su contador de vecinos izquierdos y derechos. Si ambos contadores son iguales, estos son reinicializados a cero. Si uno de los contadores tiene una diferencia de dos con respecto al otro,

entonces el proceso de balanceo es iniciado. Las acciones llevadas a cabo por los robots para balancear la formación son descritas en la tabla 2. La descripción de cada una de las acciones presentadas en las tablas 1 y 2 se presenta en la tabla 3.

MensajeRec	Estado	MensajeTrans	Acciones
Ninguna	DesBalanDer	R-Balance	SR
Ninguna	DesBalanIzq	L-Balance	SL
R-Balance	RolVecino Seguidor	= R-Balance	F2, SR
R-Balance	RolVecino Líder	= R-Balance	F2, SR, SB
L-Balance	RolVecino Seguidor	= L-Balance	F2, SL
L-Balance	RolVecino Líder	= L-Balance	F2, SL, SB
LR-Actualizar	RolVecino Seguidor	= LR-Actualizar	Ninguna

Tabla 2. Acciones a ser realizadas por los robots con el fin de balancear la formación.

Cuando el líder detecta que la formación no está balanceada, éste solicita a su vecino, el cual se encuentra del lado de la formación con más elementos, que avance dos posiciones hacia el frente con el fin de balancear la formación. Esta petición es transmitida al siguiente vecino con la misma alineación hasta que ya no hay vecinos a quien transmitir. Al iniciar la maniobra de balance, si existen más seguidores izquierdos que derechos, entonces el líder cambia su rol a seguidor derecho y le pide a su vecino izquierdo que tome el rol de líder. Por otro lado, si existen más seguidores derechos que izquierdos, entonces el líder cambia su rol a seguidor izquierdo y le pide a su vecino derecho que tome el rol de líder. Una vez que este proceso ha terminado, el líder inicializa sus contadores de seguidores izquierdos y seguidores derechos a cero. El líder cambiará los contadores conforme nuevos elementos sean integrados en la formación.

El controlador de cada robot es generado a partir de las tablas descritas anteriormente. De esta forma, conforme nuevos mensajes son recibidos y las restricciones son satisfechas, el robot determina las acciones a realizar. Una vez que las acciones han sido realizadas, en algunos casos nuevos mensajes son enviados hacia los vecinos. Como resultado de estas interacciones locales se obtiene el comportamiento global el cual consiste en organizar al grupo de robots en una formación de cuña.

Acción	Descripción
B	Transmitir el mensaje a todos
SA	Enviar el mensaje a la dirección del último mensaje recibido
SB	Enviar el mensaje a ambos vecinos
SL	Enviar el mensaje al vecino izquierdo
SR	Enviar el mensaje al vecino derecho
ML	Moverse a la izquierda
MR	Moverse a la derecha
F	Avanzar una posición
F2	Avanzar dos posiciones
LNLN	Almacenar la dirección recibida en la variable <i>ln</i> como vecino izquierdo
LNSA	Almacenar la dirección del último mensaje recibido como vecino izquierdo
RNRN	Almacenar la dirección recibida en la variable <i>m</i> como vecino derecho
RNSA	Almacenar la dirección del último mensaje como vecino derecho

Tabla 3. Acciones que pueden ser realizadas por los robots.

IV. IMPLEMENTACIÓN UTILIZANDO ROBOTS NXT

Para validar la propuesta presentada en este artículo, la formación de cuña fue realizada utilizando robots reales. Se utilizaron robots NXT. Inicialmente, los robots móviles son distribuidos aleatoriamente en un espacio libre de obstáculos. Los robots no tienen información acerca de la posición que tienen que seguir en la formación. Las funciones que los robots tienen que llevar a cabo con el fin de completar las acciones que tiene que realizar en base a su estado se describen a continuación. Estas funciones permiten que un robot alcance su posición en la formación.

EncontrarRobot. Esta función permite que un robot encuentre a otro robot en la formación. Cuando un robot es detectado, se inicia la comunicación con él a fin de identificar la posición que le corresponde al nuevo robot dentro de la formación. Cada robot mantiene una lista de las direcciones de todos los robots en el entorno. Los robots siguen el siguiente procedimiento con el fin de identificar la dirección correspondiente al robot encontrado.

```

If ( distancia_sensor_ultrasonico < d)Then
    Intentar conectarse con cada robot de la lista

    If (conexion_es_establecida) Then
        Pedir al robot vecino moverse hacia adelante una distancia k.
        Pedir al robot vecino retroceder a su posición original.
    End If
End If

```

```

If (sensor_ultrasonico_detecta_cambio) Then
    La dirección es almacenada y utilizada para iniciar el proceso de comunicación con el fin de determinar la posición del nuevo robot.

```

```

End If
End If

```

ObtenerPosición. Para obtener la posición dentro de la formación, el robot tiene que desplazarse de su posición actual hacia la posición determinada como resultado del intercambio de mensajes con el robot en la formación. Una vez que la posición es determinada, los robots realizan los siguientes cálculos con el fin de determinar su movimiento (Figura 2).

```

If ((PosicionActual=izquierda and PosiconFinal = izquierda)
or
(PosicionActual =derecha and PoscionFinal = derecha))
Then
     $y = 90 - x^2$ 
Else
    If((PosicionActual = izquierda and PosicionFinal = derecha) or
(PosicionActual = derecha and PosicionFinal =izquierda)) Then
         $y = 90 + x^2$ 
    End If
End If

```

$$d_3 = \sqrt{d_1^2 + d_2^2} - d_2 \cos y$$

$$z = \frac{\sin^{-1}(d_1 + \sin y)}{d_3}$$

$$w = 180^\circ - y - z$$

$$s = 90^\circ - w \quad (L_1 \parallel L_2 \therefore s = 90^\circ - w)$$

Balance. En el momento en que nuevos robots se integran a la formación, estos robots envían un mensaje a sus vecinos indicando que un nuevo robot ha sido integrado a la formación. Este mensaje es retransmitido hasta que el líder lo recibe. Si el líder detecta que la formación ha perdido el balance, entonces le pide a su vecino que se encuentra ubicado en el lado con más elementos que avance dos posiciones. Una vez que el robot avanza, le pide a su vecino que se encuentra del mismo lado, que avance dos posiciones. Este proceso se lleva a cabo por cada robot ubicado en el lado con más elementos. Como

resultado la formación sera balanceada y un nuevo robot asumirá el rol de líder.

balanceo puede observar en las tres últimas fotografías.

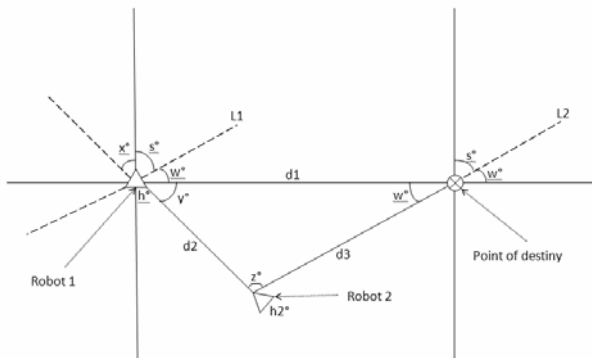


Figura 2. Trayectoria a seguir para colocarse a la derecha del robot encontrado.

Para permitir que un robot se mueva una distancia d se consideran las propiedades físicas de las llantas de los robots como se muestra en la figura 3.

Perimetró = $\pi(5.6) \approx 17.5929$ cm
 360 velocidad = rotar 360° (especificación de hardware)

```
If (t = 1000 milisegundos )
    Rotar  $1^\circ$  por unidad de velocidad
    Rotar 1 revolución por segundo
End If
```

Por lo tanto 17.5929 cm se recorren en 1 segundo con una velocidad de 360



Figura 3. Distancia recorrida en un segundo a una velocidad de 360.

V.RESULTADOS

El algoritmo propuesto para realizar la formación en cuña es probado con un grupo de 3 robots NXT. Al iniciar un robot es inicializado como líder, el segundo robot se une a la formación llegando por detrás del líder. El tercer robot se integra a la formación llegando por detrás de un seguidor derecho. La figura 3 muestra las fotografías tomadas durante el experimento. En esta figura se pueden observar las acciones llevadas a cabo por los robots para unirse a la formación. El proceso de

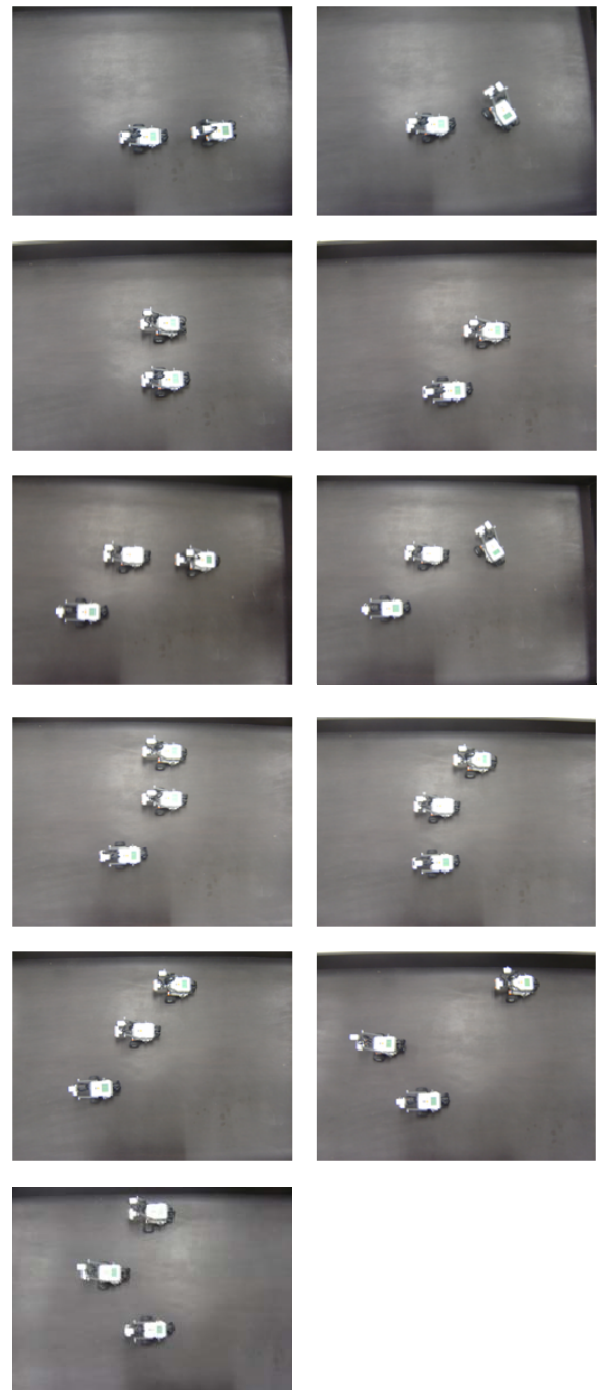


Figura 4. Fotografías tomadas durante el experimento.

VI. CONCLUSIONES

En este trabajo se logro establecer la formación de un grupo de robots NXT basado en principios de auto-organización. Las interacciones entre los robots vecinos generan como consecuencia el comportamiento global esperado (se estableció la

formación). Los resultados mostraron que un grupo de robots con capacidad limitadas puede auto-organizarse con el fin de realizar tareas complejas. La comunicación se reduce a la simple interacción entre robots vecinos. Las propiedades de escalabilidad y adaptabilidad se mantienen en la formación. Cada robot decide su posición en la formación de acuerdo a la información recibida de sus vecinos. Debido a que nuevos robots pueden ser integrados a la formación en cualquier parte de ella, la formación puede perder el balance y por lo tanto se lleva a cabo una maniobra de balanceo para recuperar la figura. El rol de líder es tomado por el robot que se encuentra en el centro de la formación. Este robot decide cuando se requiere una maniobra de balanceo.

VII. REFERENCIAS

- [1] P Bak, *How Nature Work: The Science of Self-Organized Criticality*, Springer-Verlag, Copernicus New York, September 1996, pp. 33-48.
- [2] T. Balch and R. C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926-939, December 1998.
- [3] T. Balch and M. Hybinette. Social potentials for scalable multi-robot formations. *Proceedings of the 2000 IEEE International Conference on Robotics and Automation, ICRA '00*, 1:73-80, April 2000.
- [4] T. Dierks and S. Jagannathan. Control of nonholonomic mobile robot formations: Backstepping kinematics into dynamics. *Proceedings of 16th IEEE International conference on Control Applications CCA*, 1:94-99, October 2007.
- [5] K.D. Do and J. Pan. Nonlinear formation control of unicycle-type mobile robots. *Robotics and Autonomous Systems*, 55(3):191-204, March 2007.
- [6] J. Fredslund and M. J. Mataric. A general algorithm for robot formations using local sensing and minimal communication. *IEEE Transactions on Robotics and Automation*, 18(5):837-846, October 2002.
- [7] J. Fredslund and M. J. Mataric. Robots in formation using local information. *Proceedings of 7th International Conference on Intelligent Autonomous Systems IAS-7*, 1:100-107, March 2002.
- [8] F Heylighen, *The science of self-organization and adaptivity*. L D Kiel, *The Encyclopedia of Life Support Systems: Knowledge Management, Organizational Intelligence and Learning, and Complexity*, EOLSS Publishers, Oxford, UK, 1999, pp. 253-280.
- [9] S A Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, New York, New York, May 1993, pp. 173-233.
- [10] K. N. Krishnanand and D. Ghose. Formations of minimalist mobile robots using local templates and spatially distributed interactions. *Robotics and Autonomous Systems*, 53(3-4):194-213, December 2005.
- [11] M. Lemay, F. Michaud, D. Létourneau, and J.-M. Valin. Autonomous initialization of robot formations. *Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA '04*, 3:3018-3023, April 26 – 1 May 2004.
- [12] N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. *Proceedings of the 40th IEEE Conference on Decision and Control*, 3:2968-2973, December 2001.
- [13] M. López-Sánchez. Robot behavior adaptation for formation maintenance. *ICINCO-RA*, 1:283-288, August 2006.
- [14] S. Mastellone , D. M. Stipanovic, and M. W. Spong. Multi-agent formation control and trajectory tracking via singular

perturbation. *Proceedings of 16th IEEE International Conference on Control Applications CCA*, 1:557-562, October 2007.

- [15] D. J. Naffin and G. S. Sukhatme. Negotiated formations. *Proceedings of the Eighth Conference on Intelligent Autonomous Systems*, 1:181-190, March 2004.
- [16] W. M. Spears , R. Heil, and D. Zarzhitsky. Artificial physics for mobile robot formations. *Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics, SMC*, 3:2287-2292, October 2005.

LENGUAJE PARA LA DESCRIPCIÓN DE LA INFORMACIÓN GEOGRÁFICA DE SISTEMAS DE TRÁFICO URBANO

LÓPEZ-NERI EMMANUEL, LÓPEZ-MELLADO ERNESTO, RAMÍREZ-TREVIÑO ANTONIO.

CINVESTAV, Unidad Guadalajara

Av. Científica 1145, 45015 Zapopan, Jal. México

correo electrónico: {eneri, elopez & art}@gdl.cinvestav.mx

RESUMEN

Se presenta un lenguaje para la especificación de las características geográficas, topológicas y de tránsito de un sistema de tráfico urbano. El lenguaje propuesto (UTYiL), incrementa la posibilidad de compartir el modelo de red del sistema de tráfico urbano entre las diferentes herramientas de simulación existentes al ser descrito como un meta-lenguaje de descripción de red. El lenguaje esta basado en el formato XML, lo que le permite ser un lenguaje abierto acorde con la tendencia de aplicaciones tipo GIS. Además es implementado un compilador para dicho lenguaje, que incrementa la capacidad de validación del modelo obtenido del sistema de tráfico urbano.

Palabras clave: Tráfico Urbano, modelo microscópico, GIS, lenguaje descripción formal geográfico.

I. INTRODUCCIÓN

La simulación microscópica de sistemas de tráfico urbano es una herramienta ampliamente utilizada por la comunidad científica para evaluar el impacto en el comportamiento del tráfico de la red urbana por la implementación de nuevas estrategias de control.

El trabajo previo a la simulación resulta ser una tarea compleja, ya que es necesario capturar la información geográfica del sistema de tráfico urbano (STU) que se pretende simular, por ejemplo: la conexión entre segmentos de la red, señalamientos en los segmentos, elevación, etc. En esta etapa pueden generarse algunos errores de codificación, ya que es necesario traducir la información geográfica en componentes del modelo de simulación seleccionado, provocando que los ingenieros de tráfico no dediquen la mayor

parte del tiempo en el diseño de la solución al problema a resolver, si no a problemas de captura del modelo del STU. Debido a esto, existe la necesidad de un mecanismo que permita manipular las especificaciones del modelo del STU, de manera computacional, permitiendo un análisis automatizado del espacio diseñado.

Una especificación en un lenguaje de alto nivel permite un rápido desarrollo y provee de mecanismos de validez y correctud de dicho diseño. Al ser las especificaciones una traducción de archivos generados por un sistema de información geográfica (GIS, por sus siglas en inglés), es necesario seguir la tendencia del desarrollo de software para este tipo de aplicaciones y las necesidades de los usuarios de dicha información (geógrafos, ingenieros civiles, arquitectos, etc.), es decir, mantener interoperabilidad entre los sistemas utilizando estándares abiertos [7][8]. En general se comienza a reconocer que la aplicación de software en sí, no es lo más importante, si no por el contrario, lo son las bases de datos (la información) que se han construido y recolectado a lo largo de los años. De ahí que es importante el tipo de formato que permita almacenar la información con el paso del tiempo. Algunos estándares abiertos son: OGC[18], GML [9] y XML [5][6].

Existen una gran cantidad de herramientas de simulación que fueron diseñadas para cubrir aspectos muy específicos de análisis de los STU. Cada herramienta tiene su propio formato para capturar el modelo del STU. Esto provoca que aspectos no contemplados en alguna, el modelo no pueda ser reutilizado por alguna otra herramienta que si lo cubre. Este problema también se presenta en el área de telecomunicaciones con la simulación de redes [2].

La mayoría de las herramientas de simulación, por ejemplo, SUMO [3], CALTRANS [4], MatSim [12] entre otras, carecen de mecanismos de formalización y no utilizan formatos o estándares comunes entre la comunidad geográfica. ATLAS [1] es un lenguaje formal para la descripción de un STU, e inclusive se ha desarrollado un compilador para el mismo. Este lenguaje describe el modelo del STU utilizando como base teórica el formalismo CELL-DEVS [13]. Existen otros simuladores como [10] y [11] que utilizan un lenguaje denominado de descripción de escenarios (SDL, por sus siglas en inglés) para describir la estructura de red. Desafortunadamente estos lenguajes son muy específicos del formalismo base teórico, eliminando otras características necesarias de otros modelos eliminando potencialmente la reutilización de los modelos de STU.

En este artículo se presenta un lenguaje que permite la especificación del modelo microscópico del STU (UTYiL, por sus siglas en inglés). La característica formal del lenguaje permite sea automatizado y por lo tanto disminuye los errores de captura del modelo. El lenguaje propuesto permite capturar la información geográfica, topológica y de tránsito de un STU, permitiendo al ingeniero de tráfico modelar con gran nivel de detalle (microscópico) la sección de una ciudad en particular. El lenguaje está basado en el formato de marcado extensible (XML de las siglas en inglés), esto permite el almacenamiento de grandes cantidades de información y sigue la tendencia para el almacenamiento de mapas geográficos, tal como el proyecto LandXML, el cual es un estándar basado en XML que está siendo utilizado en las aplicaciones de ingeniería civil y GIS [15]. Además, se muestra como la información capturada por el lenguaje propuesto puede ser utilizada en otros modelos o simuladores como un meta-lenguaje.

II. COMPONENTES ESTÁTICOS DEL STU

En [14] se presenta un modelo discreto orientado a eventos de un. En la fig. 1, se muestran los componentes estáticos de dicho modelo: La estructura de red y los señalamientos estáticos del

sistema. Se desean capturar todos los elementos que no cambian de valor o que no se mueven en la red. Cada señalamiento tiene asignada una posición en el segmento. Los segmentos están relacionados entre ellos formando una red conectada de segmentos.

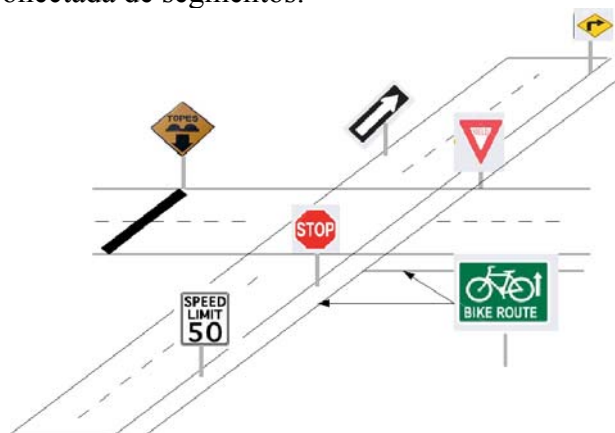


Fig 1: Componentes estáticos del sistema de tráfico urbano

III. DESCRIPCIÓN FORMAL DE COMPONENTES ESTÁTICOS DEL STU

A. Objetos

Los señalamientos de tráfico estático son representados como *objetos*. Cada objeto es representado por la 2-tupla:

$$o_m = (type_m, w_m)$$

Donde:

- $type_m = (signal, descriptor)$, y *signal* puede ser cualquier señalamiento de tráfico en el STU (señalamiento de prioridad, paro total, prioridad a la derecha, límite de velocidad, no estacionarse, dirección incorrecta, etc.) y *descriptor* es información suplementaria del señalamiento, por ejemplo, 80 km/hr para el señalamiento de límite de velocidad
- $w_m \in [0, size\ of\ road]$, es la posición del objeto en la calle.

El conjunto $O = \{o_m \mid o_m \text{ es un objeto}\}$ representa los objetos del modelo.

B. Segmentos

Las calles son representadas usando el componente del modelo *segmento*. Cada *segmento* tiene una longitud y contiene a un conjunto de objetos y es representado usando la 4-tupla:

$$s_i = (a_i, b_i, O_i, type_i)$$

Donde:

- $a_i, b_i = (lat_i, long_i)$ y $lat_i, long_i \in \mathbb{R}$, los cuales son la latitud y la longitud de los puntos extremos del segmento.
- O_i es el conjunto de objetos en el segmento i , $O_i \subseteq O$
- $type_i \in \{calle, crucePeaton, intersección\}$, representa el tipo del segmento.

El conjunto $S = \{s_i \mid s_i \text{ es un segmento}\}$ representa los segmentos del modelo. Las siguientes dos relaciones entre los segmentos son usadas para representar la conexión de los segmentos en la red.

Relación 1: Vecindad Adyacente (Segmentos Paralelos)

$$NC = \{(s_i, s_j) \mid s_i, s_j \in S, s_i \text{ es un vecino adyacente de } s_j\}$$

Relación 2: Vecindad Secuencial (Segmentos secuenciales)

$$NS = \{(s_i, s_j) \mid s_i, s_j \in S, s_i \text{ es un vecino secuencial de } s_j\}$$

En la fig. 2 se muestran las relaciones posibles entre los segmentos que permiten la construcción de la estructura de red del STU.

A continuación se introducen los conceptos de segmentos fuente y segmentos pozo. Esto permite describir los segmentos borde de un STU finito.

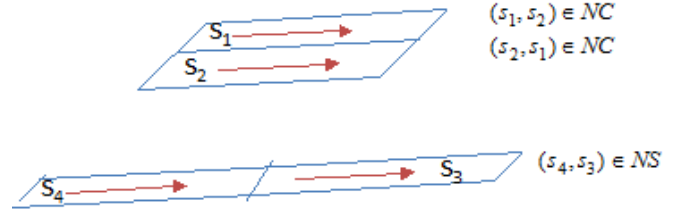


Fig. 2: Representación gráfica de los componentes Segmento y sus relaciones

Definición 1:

Sea $s_i, s_j \in S$, s_i es un segmento fuente iff $\neg \exists s_j$ tal que $(s_j, s_i) \in NS$. s_i es un segmento pozo iff $\neg \exists s_j$ tal que $(s_i, s_j) \in NS$.

La intersección es un caso especial del STU y es representado utilizando el concepto de segmentos. Usando la relación de vecindad secuencial es posible conectar más de un segmento como entrada o salida a otro segmento. En la figura fig. 3(b) el segmento S_9 y S_{10} representan a la intersección I_{10} de la fig. 3(a).

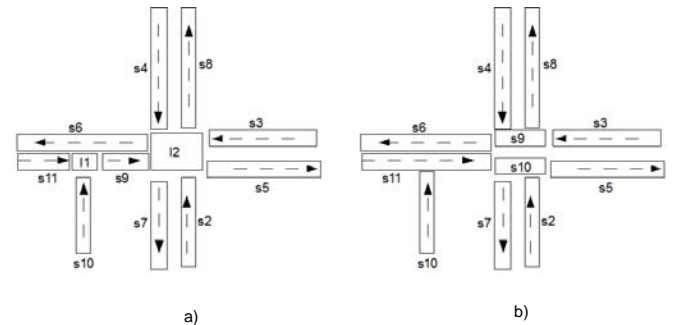


Fig. 3: Definición de intersección utilizando los componentes segmento

IV. META-Lenguaje PARA LA ESPECIFICACIÓN DE SISTEMAS DE TRÁFICO URBANO

A. Modelo de la Red de Tráfico Urbano

En la fig. 4 se muestra la descripción de un modelo completo de STU utilizando componentes del lenguaje UTYiL, los cuales son: *Segment*, *Object* y *Connection*. Esta representación jerárquica es resultado de la descripción formal presentada en la sección 3.

representada por n segmentos, donde n es el número de carriles de la calle.

La regla sintáctica que define a un segmento está formado por:

```
SEGMENT := <segment> id=[ID]
segTy=[SEGMENTTYPE] x1=[X1] y1=[Y1]
x2=[X2] y2=[Y2] [OBJECT|OBJECT OBJECT]
</segment>
```

La longitud del segmento es obtenida a partir de los puntos x_1, y_1, x_2, y_2 .

La calle “Escalona” de la fig. 6, es dividida en 3 segmentos (dicha calle tiene solo un carril), ya que la intersecan Valley y Breschka. Sin embargo, en el segmento 2 de la calle Escalona, existe un objeto, es un tope que necesita ser integrado al segmento.

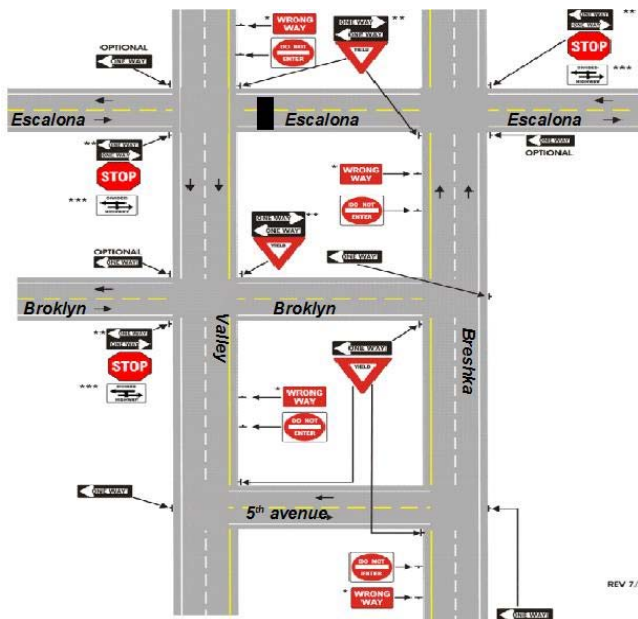


Fig. 6: Estructura de red con dos carriles

Entonces el ejemplo de la fig. 6 quedaría capturada utilizando UTYiL como sigue:

```
<segment>
id="2" x1=20.0 y1=30.0 x2=20.0 y2= 35.0
```

```
<object id="1" type="bump" pos="0.0"
value="0"/>
<object id="2" type="direction" pos="0.0"
value="p2"/>
<object id="3" type="speedLimit" pos="30.0"
value="15.0"/>
<object4 type="warningBump" pos="15.0"
value="15.0"/>
</segment>
```

D. Etiqueta Conexiones

La conexión entre los segmentos está definida por las relaciones NS y NC del apartado 3. En la fig. 6 se muestra un segmento conectado de manera paralela al segmento 2 y secuencial con el segmento 3. Es traducida en el lenguaje de la siguiente manera:

```
<Connections>
<connectionNS from="1" to="3"/>
<connectionNC from="1" to="2"/>
</Connections>
```

E. Descripción formal de UTYiL

UTYiL puede ser descrito de manera formal utilizando la siguiente gramática:

$$G = (N, \Sigma, S, P)$$

Donde:

- $N = \{ \text{OBJECT, SEGMENT, CONNECTION, RELATIONS, TOPOLOGY, STRUCTURE, SIGNTYPE, REF, POS, VALUE, STU, ID, X1, Y1, X2, Y2} \}$, es una colección finita de símbolos no terminales.
- $\Sigma = \{ \text{bump, direction, speedLimit, warningBump, p1, p2, connectionNS, connectionNC, <network>, </network>, <connections>, </connections>, <segments>, </segments>} \}$, es un alfabeto (o símbolos terminales).
- $S = \text{STU}$, es un símbolo no terminal llamado símbolo inicial.

- $P = \{(STU, \langle network \rangle STRUCTURE \langle /network \rangle), (STRUCTURE, \langle segments \rangle TOPOLOGY \langle /segments \rangle \langle connections \rangle RELATIONS \langle /connections \rangle), (TOPOLOGY, SEGMENT), (TOPOLOGY, SEGMENT SEGMENT), (RELATIONS, CONNECTION), (RELATIONS, CONNECTION CONNECTION), (SEGMENT, \langle segment \rangle id = ID segTy = SEGTYPE x1 = X1 y1 = Y1 x2 = X2 y2 = Y2 OBJECT \langle /segment \rangle), (OBJECT, \langle object id = REF type = SIGNTYPE pos = POS value = VALUE \langle / \rangle), (CONNECTION, \langle connectionNS seg1 = ID seg2 = ID \rangle), (CONNECTION, \langle connectionNC seg1 = ID seg2 = ID \rangle) \}$

P es el conjunto de reglas de sustitución llamadas producciones y tienen la forma $A \rightarrow w$, donde $A \in N$ y w es una cadena sobre $\Sigma \cup N$. Ya que las producciones son pares de no terminales de N con cadenas de Σ , entonces por conveniencia son representados como pares ordenados $N \times \Sigma$. Entonces el para (x,y) de $N \times \Sigma$ representa la producción $x \rightarrow y$.

De lo anterior puede ser construido un autómata no determinista de pila que puede realizar una derivación por la izquierda

$Q = \{q_1, q_2, q_3\}$
 $\Gamma = \{N \cup \Sigma \cup STU\}$, donde STU es el símbolo inicial de la pila.
 $F = \{q_3\}$, donde q_3 es el estado de aceptación
 $s = q_1$, donde q_1 es el estado inicial

El autómata tiene la siguiente regla de transición:

- $\Delta(q_1, \epsilon, STU) = \{(q_2, STU)\}$
- $\Delta(q_2, \epsilon, STU) = \{(q_2, \langle network \rangle STRUCTURE \langle /network \rangle)\}$
- $\Delta(q_2, \epsilon, STRUCTURE) = \{(q_2, \langle segments \rangle TOPOLOGY \langle /segments \rangle \langle connections \rangle RELATIONS \langle /connections \rangle)\}$
- $\Delta(q_2, \epsilon, TOPOLOGY) = \{(q_2, SEGMENT), (q_2, SEGMENT SEGMENT)\}$

- $\Delta(q_2, \epsilon, RELATIONS) = \{(q_2, CONNECTION), (q_2, CONNECTION CONNECTION)\}$
- $\Delta(q_2, \epsilon, SEGMENT) = \{(q_2, \langle segment \rangle id = ID segTy = SEGTYPE x1 = X1 y1 = Y1 x2 = X2 y2 = Y2 OBJECT \langle /segment \rangle)\}$
- $\Delta(q_2, \epsilon, OBJECT) = \{(q_2, \langle object id = REF type = SIGNTYPE pos = POS value = VALUE \langle / \rangle), (q_2, \langle object id = REF type = SIGNTYPE pos = POS value = VALUE \langle / \rangle), (q_2, \epsilon)\}$

V. COMPILADOR Y GENERACIÓN DE ESTRUCTURAS

Siguiendo la gramática propuesta, se desarrolla un compilador que analiza y traduce en las estructuras necesarias lo que ha sido capturado con UTYiL. El lenguaje es usado de manera nativa por la herramienta de simulación CiudadelaSim [14] como es mostrado en la fig. 7.

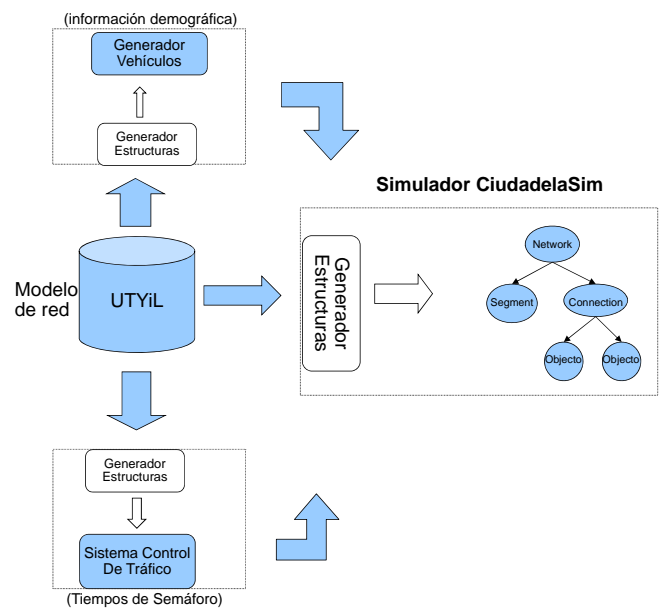


Figura 7. Uso del archivo EXE en el simulador

Una vez compilado se generan las estructuras necesarias que representan el modelo STU en la máquina de simulación. En la fig. 8 se muestra un ejemplo de dicha estructura dinámica.

Además del simulador, el modelo del STU capturado con UTYiL es utilizado por el control de semáforos que lo utiliza para conocer la estructura de red y de acuerdo a la estrategia de control asignar los tiempos a cada segmento, así como por el módulo generador de vehículos.

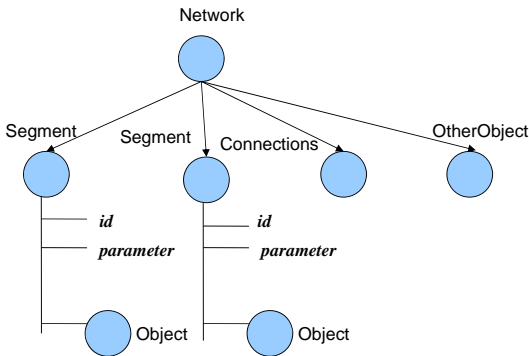


Fig. 8: Componentes del lenguaje UTYiL y estructuras de datos generadas por el compilador

VI. UTYiL COMO META-LENGUAJE

Desacoplar el lenguaje de cualquier herramienta de simulación permite incrementar de manera potencial el rehuso de un modelo de red. Todos los datos del modelo de red son trasladados al lenguaje de descripción de la red deseado, es decir, el modelo puede ser utilizado por cualquiera de las herramientas de simulación existentes (ver fig. 9). A continuación se describe la metodología de conversión ejemplificando el uso de UTYiL como meta-lenguaje de otras herramientas de simulación.

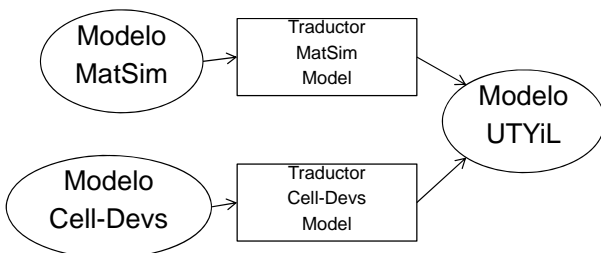


Fig 9: Convertidor a modelo UTYiL

A. MatSim usando UTYiL

El formato de archivo de MatSim [12] utiliza dos archivos uno llamado nodo y otro enlace. En la tabla 1 se muestran algunas de las características más importantes y distintivas del modelo. La

dirección del carril de A-B o viceversa y el número de carriles es descrito por campo *permLanesB*, en este caso indica tres carriles hacia el punto B. Los límites de velocidad son descritos por el campo *speedLmtA*. El campo *rightPacktA*, describe una configuración con un carril llamado de tipo pocket (fig. 10.a) que permiten acelerar la salida de los vehículos a la derecha o izquierda. Los carriles tipo pocket, son segmentos conectados de manera paralela (relación *NC*) con algún segmento central. Para capturar la restricción de solo vuelta a la derecha o izquierda se utiliza el objeto “*rightOnly*” o “*leftOnly*” El campo *twoWayTurn*, describe una configuración especial inglesa, con un carril central para dar vuelta a la izquierda (fig 10.b).

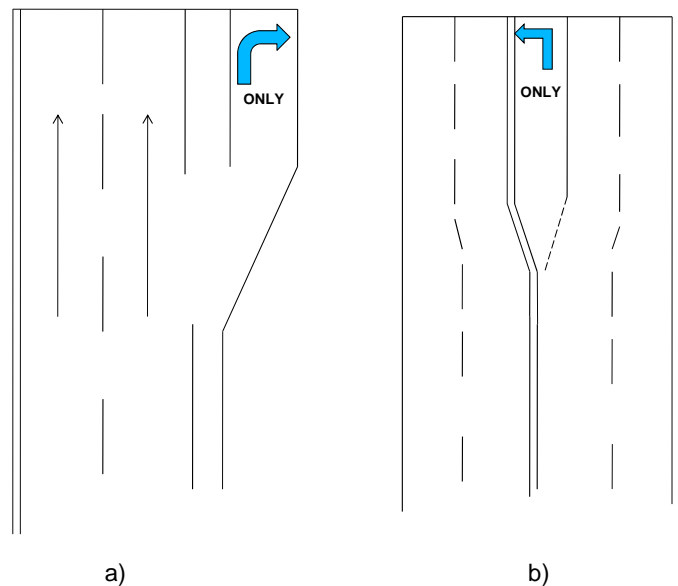


Fig. 10: a) Carril tipo pocket, b) Carril dos vías para vuelta a la izquierda

Tabla 1. Características Distintivas del Modelo

PERMLANESB	RIGHTPACKTSA	SPEEDLMTA	TWOWAYTURN
3	1	60	true

Utilizando UTYiL es posible describir la configuración de la tabla 1:

```
<segment> id="1" x1="20.0" y1="30.0"
x2="20.0" y2=" 35.0" z="30.0" slope="15"
<Object id="2" type="speedLimit" pos="0.0"
value="60"/>
</segment>
```

```

<segment> id="2" x1="20.0" y1="30.0"
x2="20.0" y2="35.0" z="30.0" slope="15"
<Object id="2" type="speedLimit" pos="0.0"
value="60"/>
</segment>
<segment> id="3" x1="20.0" y1="30.0"
x2="20.0" y2="35.0" z="30.0" slope="15"
<Object id="1" type="righthOnly" pos="0.0"
value="0"/>
<Object id="2" type="speedLimit" pos="0.0"
value="60"/>
</segment>

```

B. CELL-Devs usando UTYiL

Cell-devs es un formalismo que utiliza la teoría de eventos discretos y autómatas celulares. Cada calle está formada por varios carriles. ATLAS, es el lenguaje utilizado para capturar el modelo y traducirlo a CELL-Devs, una de las capacidades es poder capturar áreas en reparación y áreas de estacionamiento. Ver Fig. 11.

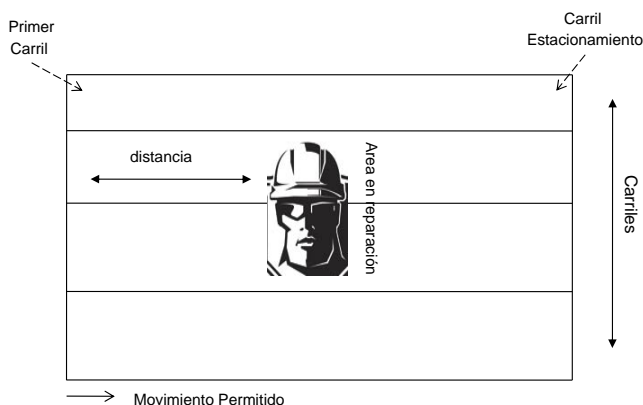


Fig. 11 Carriles de estacionamiento y en reparación

Se declaran los objetos *parkingLane* que indicarán al conductor que es un carril para poder estacionarse y el objeto *jobSite* que indican que el carril está obstruido por reparación.

```

<segment> id="1" x1="20.0" y1="30.0"
x2="20.0" y2="35.0"
<object id="1" type="parkLane" pos="0.0"
value="AllLane"/>
</segment>
<segment> id="2" x1="20.0" y1="30.0"
x2="20.0" y2="35.0"

```

```

<object id="1" type="jobSite" pos="50.0"
value="null"/>
</segment>
<segment> id="3" x1="20.0" y1="30.0"
x2="20.0" y2="35.0"
<object id="1" type="jobSite" pos="50.0"
value="null"/>
</segment>
<segment> id="4" x1="20.0" y1="30.0"
x2="20.0" y2="35.0"
</segment>

```

VII. CONCLUSIONES

En este artículo se ha propuesto una meta-lenguaje para la especificación de las características geográficas, topológicas y de tránsito de un sistema de tráfico urbano (UTYiL). UTYiL está basado en el formato XML. Utiliza como base teórica el modelo discreto para STU propuesto en [14]. Utilizando componentes del lenguaje UTYiL, se mostró la factibilidad de describir el modelo de red de otras herramientas de simulación en aquellos aspectos en los que las herramientas son diferentes entre ellas. UTYiL incrementa la posibilidad del reuso del modelo entre dichas herramientas.

Actualmente se está trabajando en incrementar la semántica del lenguaje al incluir la descripción de los flujos de tráfico y la descripción en particular de eventos que describen la parte dinámica del STU. Además se está desarrollando un convertidor del formato openStreetMap [16][17] al lenguaje UTYiL, esto permitirá utilizar la interfaz gráfica de openStreetMap para la descripción de la red, acelerando el tiempo de diseño y captura del modelo del STU.

VIII. REFERENCIAS

- [1] G. Wainer, "ATLAS: A language to specify traffic models using Cell-DEVS". In *Simulation Modelling Practice and Theory*. Vol 14. 2006, pp. 313-337.
- [2] Rahman, M. A., A. Pakstas and F. Z. Wang. (2005). "An Approach to Integration of Network Design and Simulation Tools". In 8th

- International Conference on Telecommunications – ConTel.
- [3] Krajzewicz, D., G. Hertkorn, C. R. P. Wagner, (2002) "*An Example of Microscopic Car Models Validation using the open source Traffic Simulation SUMO*"; In: Proceedings of Simulation in Industry, 14th European Simulation Symposium, SCS European Publishing House, pp. 318-322, Simulation in Industry, 14th European Simulation Symposium.
- [4] Church, R. L., and V. Noronha., (2003). "Caltrans Traffic Operations". User's manual, S2P Version 3.0d. Vehicle Intelligence and transportation Analysis Laboratory.
- [5] Milo, T., S. Abiteboul, B. Amann, O. Benjelloun, F. D. Ngoc, "Exchange intensional XML data", in *Proc. of the 2003 ACM SIGMOD International Conference on Management of Data*, San Diego, CA, 9-12 June 2003, pp. 289-300.
- [6] Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation, 04 February 2004. (<http://www.w3.org/TR/REC-xml/>)
- [7] R. Bourret, "XML and database", July 2004. (<http://www.rpbouret.com/xml/XMLAndDatabases.htm>)
- [8] J. H. Yoon, Hong T. Ju, James W. Hong, "Development of SNMP-XML translator and gateway for XML-based integrated network management", *International Journal of Network Management*, Vol 13, pp. 259- 276, 2003
- [9] OMG XML Metadata Interchange (XMI) Specification, Version 1.2, Object Management Group (OMG), Inc., January 2002. (<http://www.omg.org/docs/formal/02-01-01.pdf>.)
- [10] Gustavsson, L., (2007). "Dynamically Adaptive Intelligent Agents in Driving Simulator Environments".
- [11] Kearney, J. (1999) "Scenario Languages for Driving Simulation," *Proc. Driving Simulation Conf.*, 1999, pp. 123–133 .
- [12] Nagel, K., (2005), Multi-agent transportation simulations, draft-in-progress for a book.
- [13] Wainer, G., N., Giambasi, (1998). "Specification, modelling and simulation of timed Cell-DEVS models". Technical Report 97-007, Departamento de Computación, FCEN/UBA.
- [14] López-Neri, E., López-Mellado, L. E. and Ramírez-Treviño, A. (2005). "Mobil Agent Systems based urban traffic microsimulation". Master thesis. Cinvestav. Unidad-Guadalajara. Spanish.
- [15] Avatech Solutions, (2008)., "AutoCAD Civil 3D 2008 LandXML Drawing Support", White paper, Civil, Survey and mapping solutions. (<http://www.avatech.com/solutions/infrastructure/infrastructure-whitepapers.aspx>)
- [16] Miller, P., R. Styles, and T. Heath. (2008)., "Open Data Commons, a license for open data". Workshop on Linked data on the web, Beijing, China.
- [17] Nelson, A., A. Sherbinin, and F., Pozzi, (2003)., "Towards development of a high quality public domain global roads database". In *Data Science Journal*, Volume 5, 20 November 2006. pg. 223-265.
- [18] R. Lake, "Markup Languages and Google Earth Enable the GeoWeb," Issue Date: November – 2005

SCAM: SISTEMA DE CONTROL DE ACCESO AL MACROCENTRO DE CÓMPUTO

Italia Estrada Cota, Mónica A. Carreño León, Elvia E. Aispuro Félix, J. Andrés Sandoval Bringas, Jaime Suarez Villavicencio, Martín Agúndez Amador, Mario Josué Rubio Durán*

Profesores- Investigadores del Departamento de Sistemas Computacionales

*Alumno de Licenciatura en Computación

Universidad Autónoma de Baja California Sur

La Paz B.C.S., México

[iestrada, mcarreno, aispuro.sandoval, jsuarez.magundez, *mrubio}@uabcs.mx](mailto:{iestrada, mcarreno, aispuro.sandoval, jsuarez.magundez, *mrubio}@uabcs.mx)

RESUMEN

SCAM (Sistema de Control de Acceso al Macrocentro) es un sistema informático que se desarrolló para apoyar las actividades propias del macrocentro de cómputo de la Universidad Autónoma de Baja California Sur (UABCS). SCAM es un sistema que controla el acceso de usuarios al macrocentro, asigna equipos de cómputo, a través de la utilización de sockets permite que el sistema sea capaz de realizar diferentes tareas como: bloquear, desbloquear, apagar, reiniciar los equipos de cómputo así como tomar fotos de la pantalla del equipo. El desarrollo de SCAM se apoyó en las técnicas, herramientas y modelos que proporciona la ingeniería de software para lograr un software de calidad.

Palabras clave: Sistema, sistema informático ingeniería de software.

I. INTRODUCCIÓN

La Universidad Autónoma de Baja California Sur (UABCS), dentro de su infraestructura cuenta con un macrocentro de cómputo, el cual viene a satisfacer la demanda del alumnado, personal académico y administrativo, apoyándolo en sus actividades cotidianas.

El macrocentro de cómputo de la UABCS, actualmente cuenta con 102 equipos de cómputo para el uso de la comunidad universitaria, este macrocentro de cómputo está estructurado por

zonas específicas de software además cuenta con servicios digitales e internet.

La UABCS no contaba con un sistema [1,2] dedicado a controlar el acceso al macrocentro, así como la asignación de equipo, por lo cual fue necesario desarrollar un software especializado y dedicado a realizar estas funciones específicas.

A lo largo de este artículo se reporta el desarrollo del sistema SCAM utilizando las técnicas, herramientas y modelos de la ingeniería de software.

II. SOFTWARE ACTUALES

Actualmente existen en el mercado diversas herramientas comerciales de software especializado para la administración de “cafés internet”,(i.e. CybertPlanet [3], Smartlaunch [4], CyberPro[5], ICafe[6]), cada uno de ellos con sus ventajas y desventajas, sin embargo desarrollados cada uno de manera general para que un centro de cómputo se adopte a ellos y no viceversa, además de tener costos considerables; por lo cual se desarrollo un software específico para dar solución a las necesidades propias del macrocentro de cómputo de la institución.

III. CONCEPTOS BÁSICOS

INGENIERÍA DE SOFTWARE

Actualmente los sistemas de información están cambiando la manera de operar de las organizaciones. A través de su uso se logran importantes mejoras: automatizan procesos, facilitan la manipulación de información para el proceso de toma de decisiones, facilitan el logro de ventajas competitivas a través de su implantación dentro de las empresas, entre otras [7].

Según *Jane & Kenneth C. Laudon* un sistema de información se define: "conjunto de componentes interrelacionados que permiten capturar, procesar, almacenar y distribuir la información para apoyar la toma de decisiones y el control en una institución" [8]. Estos sistemas básicamente realizan los cuatro pasos: entrada, almacenamiento, procesamiento y salida de información.

Hoy en día, para el desarrollo de sistemas de información, se toma en cuenta la *Ingeniería de Software*, ya que ofrece métodos y técnicas para desarrollar y mantener software de calidad [9].

La ingeniería del software tiene diversas definiciones; una definición desarrollada por la IEEE es: "La ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software" [10].

Con independencia del área de aplicación, tamaño o complejidad del proyecto, cualquier sistema se encontrará al menos en una de las siguientes fases genéricas: definición, desarrollo y mantenimiento [9]. Esto es:

- *Definición.*- Se centra sobre el qué; esta fase intenta identificar: que información va a ser procesada, que función y rendimiento desea, que compartimiento, que criterios de validación, entre otros.
- *Desarrollo.*- Se centra en el cómo. Esta fase define como son las estructuras de datos, como

se traduce el diseño en una codificación, como son las interfaces, entre otros.

- *Mantenimiento.*- Se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software, entre otros.

La ingeniería del software permite desarrollar sistemas de información de calidad, para ello actualmente existen dos enfoques: estructurado y orientado a objetos [11]. En el enfoque estructurado, la descomposición del problema es fundamentalmente por funciones o procesos, lo cual origina una división jerárquica de procesos constituidos por subprocesos; por otro lado el enfoque orientado a objetos busca ante todo descomponer un espacio de problema por objetos.

La ingeniería de software orientada a objetos permite crear software modelando el mundo de forma tal que ayuda a entenderlo, controlarlo y gobernarlo de una mejor manera. [9].

IV. DESARROLLO DE SCAM

INTRODUCCIÓN

Para el desarrollo de SCAM, primeramente se planteó una idea general para la administración y control del macrocentro de cómputo, está se ilustra en la figura 1, los puntos principales de funcionamiento para SCAM son:

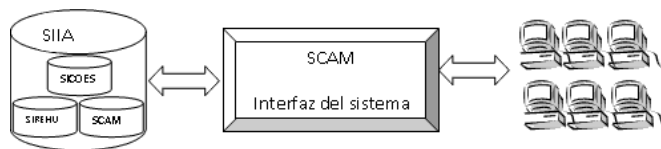


Fig.1 Esquema general del SCAM

1. Identificación de la información necesaria que existe dentro del SIIA (Sistema Integral de Información Administrativa) que permita *crear la bases de datos* de SCAM, para que alumnos, personal docente y administrativo de

la institución, puedan acceder al macrocentro de cómputo.

2. *Interfaz de usuario* del sistema SCAM, que permitan que el sistema sea fácil de usar, para realizar las tareas pertinentes a la administración y control de un macrocentro de cómputo.

El desarrollo de SCAM se realizó utilizando el ciclo de vida clásico [12] y la ingeniería de software orientada a objetos.

REQUERIMIENTOS

La primera fase del ciclo de vida para el desarrollo de SCAM es *definir* claramente los *requerimientos* del sistema, entre las tareas necesarias para realizar son:

- Administrar los procesos de asignación, control y monitoreo de los equipos de cómputo.
- Llevar registros de tiempo de uso de los equipos por parte de los usuarios.
- Monitorear lo que el usuario este realizando (tomar fotos de la pantalla).
- Manejar reportes de software y hardware de los equipos.
- Generar estadísticas de uso de los equipos, software más utilizado, horas pico de utilización del macrocentro.
- Bloquear, desbloquear, apagar y reiniciar los equipos.
- Registrar y controlar todo el equipo de computo (hardware y software) del macrocentro.
- Manejar toda la información necesaria de los usuarios (alumnos, académicos, administrativos e invitados) para su ingreso y uso del equipo del macrocentro, esta información se encuentra en el SIIA.
- Llevar un historial por equipo de cómputo, para su mantenimiento.
- Llevar una bitácora de operación del sistema.
- Administrar las zonas específicas; el equipo de cómputo destinado a cada zona y su software.
- Tener seguridad en el sistema.

ANÁLISIS

Definidos los requerimientos de SCAM, la siguiente fase es que se *analicen* cada uno de los requerimientos para determinar: *¿Qué es lo que se requiere para SCAM?* . Los requisitos fundamentales del sistema, se ilustran en la figura 2, está muestra el contexto de SCAM mediante un diagrama de casos de usos, el cual muestra un conjunto de casos de usos, sus actores y relaciones. Los actores que interactúan en SCAM son:

- *Administrador*, es la persona responsable del funcionamiento del sistema.
- *Encargado*, es la persona que maneja el sistema.
- *Usuario del Macrocentro*, es la persona que accesa al macrocentro.
- *SiReHu*, es el sistema que cuenta con la base de datos del personal académico y administrativo.
- *SiCOEs*, es el sistema que cuenta con la base de datos de todos los alumnos inscritos de la institución.

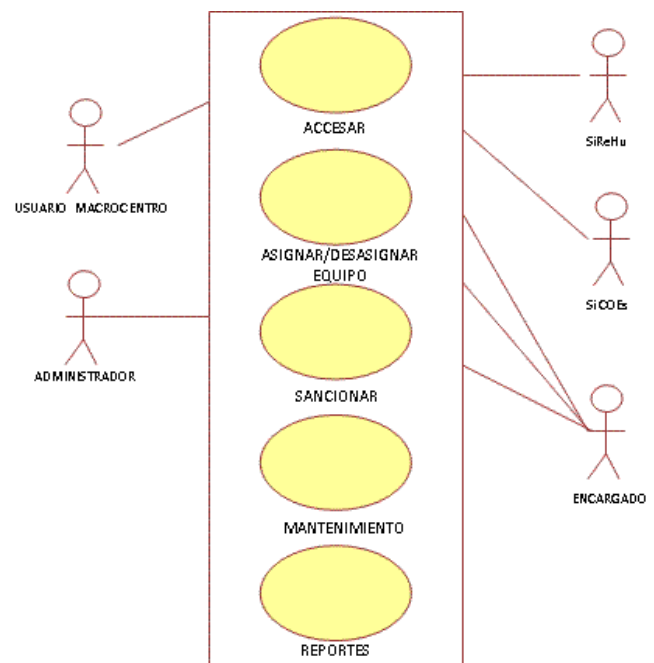


Fig.2 Modelado del contexto de SCAM (caso de uso)

Dentro de los casos de uso de SCAM están:

- *Caso de uso Accesar:* Permite la entrada al macrocentro, para ello el usuario (académico, administrativo, alumno u caso especial) debe estar registrado como parte de la institución.
- *Caso de uso Asignar o desasignar equipo:* Permite asignar/ desasignar un equipo de cómputo a un usuario previamente identificado, se controla el tiempo de entrada y salida, y la computadora en uso.
- *Caso de uso Sancionar:* Permite registrar sanciones a los usuarios, estas podrán repercutir en la próxima asignación de equipo al usuario dependiendo del tipo de sanción.
- *Caso de uso Mantenimiento:* Permite contralar y administrar las computadoras, zonas del macrocentro y hardware.
- *Caso de uso Reportes:* Permite generar diversos reportes del sistema.

DISEÑO

Para continuar con las fases de desarrollo del ciclo de vida de SCAM, fue necesario definir el *¿Cómo iba a ser el funcionamiento del sistema?*, para ello primeramente se realizaron *los diagramas de secuencia* con los objetos y actores que participan; a continuación se muestra en la figura 3 un diagrama de secuencia correspondiente a la asignación de un equipo de cómputo a un usuario, ya que esto es la tarea principal de SCAM.

El diagrama de secuencia (figura 3) muestra un escenario posible al momento en que un usuario accesa al macrocentro de cómputo identificándose; el encargado a través del sistema verifica si es usuario, disponibilidad en la zona que desea entrar, así como si el usuario no tiene sanciones. El encargado por último asigna un equipo de cómputo a un usuario en una fecha y tiempo determinado.

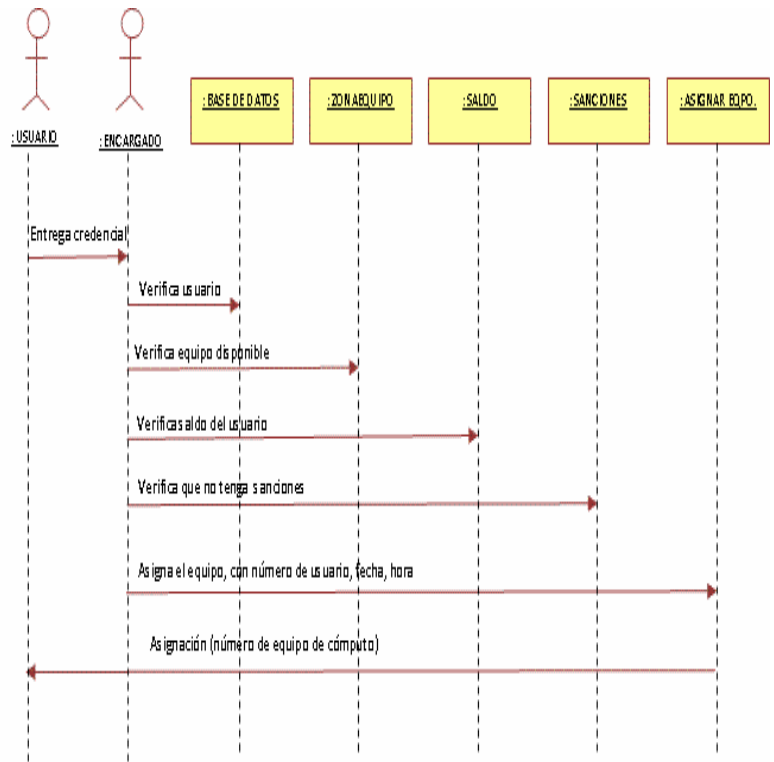


Fig.3 Interacción mostrada correspondiente a la asignación de un equipo mediante un diagrama de secuencia

Contando con todos los diagramas de secuencia del sistema, se procedió al *diseño de la base de datos* que permite llevar un control y manejo de toda la información referente a los usuarios, equipo de cómputo, hardware y software.

La base de datos se encuentra en Oracle 8i, (ya que el sistema forma parte del SIIA) y está trabaja bajo el modelo cliente- servidor mediante la comunicación soportada con TCP/IP; el acceso a los datos es ah-doc.

La figura 4 muestra la base de datos del sistema, a continuación se describen algunas de las tablas que forman parte de la bases de datos de SCAM:

- *Seguridad del sistema* mediante las tablas: privilegios de encargados, tipos de encargados, encargados del sistema y bitácora de operación.
- *Manejo de zonas y equipos* mediante las tablas: zonas, equipos por zonas, software, software

zona, equipo pc, estados de equipos e historial, equipo por zona.

- *Manejo de usuarios del macrocentro* mediante las tablas: alumnos (vista obtenida de SiCOEs), empleados (vista obtenida de SiREHu), usuarios especiales y tipos de usuarios especiales. Cabe mencionar que se realizan actualizaciones de usuarios en un periodo de seis meses.
- *Historial de equipo por zona* en la que la tabla contiene información de las asignaciones de equipos a los usuarios (quién, qué equipo de cómputo usó, cuanto tiempo, cuando inició, cuando terminó, quien se la asignó, entre otras).

En el *diseño de interfaces de usuario*, se tomaron en cuenta las personas que utilizarían el sistema, la tarea a realizar, así como el entorno; a cada interface se le midió su complejidad, además se aseguró que las interfaces fueran: consistentes, fáciles de aprender, flexibles, robustas, entre otras.

DESARROLLO, IMPLEMENTACIÓN Y PRUEBAS

Posteriormente se procedió con la fase de la implementación del sistema SCAM este fue desarrollado bajo el lenguaje de programación Delphi 7.0, y el manejador de base de datos Oracle 8i (debido a que el sistema forma parte del SIIA y son las plataformas de desarrollo de la institución), este proceso duró aproximadamente 6 meses.

Posteriormente se en la fase de pruebas se realizaron las *pruebas de caja blanca* las cuales aseguraron que todas las sentencias y condiciones se ejecutaran al menos una vez; y las *pruebas de caja negra* permitieron verificar el comportamiento adecuado de los módulos de SCAM, en respuesta a las acciones o entradas originadas por el usuario.

Terminada la fase de prueba, se procedió a dar la *capacitación* al personal que operaría el sistema,

para su posterior implantación en el macrocentro de cómputo de la institución.

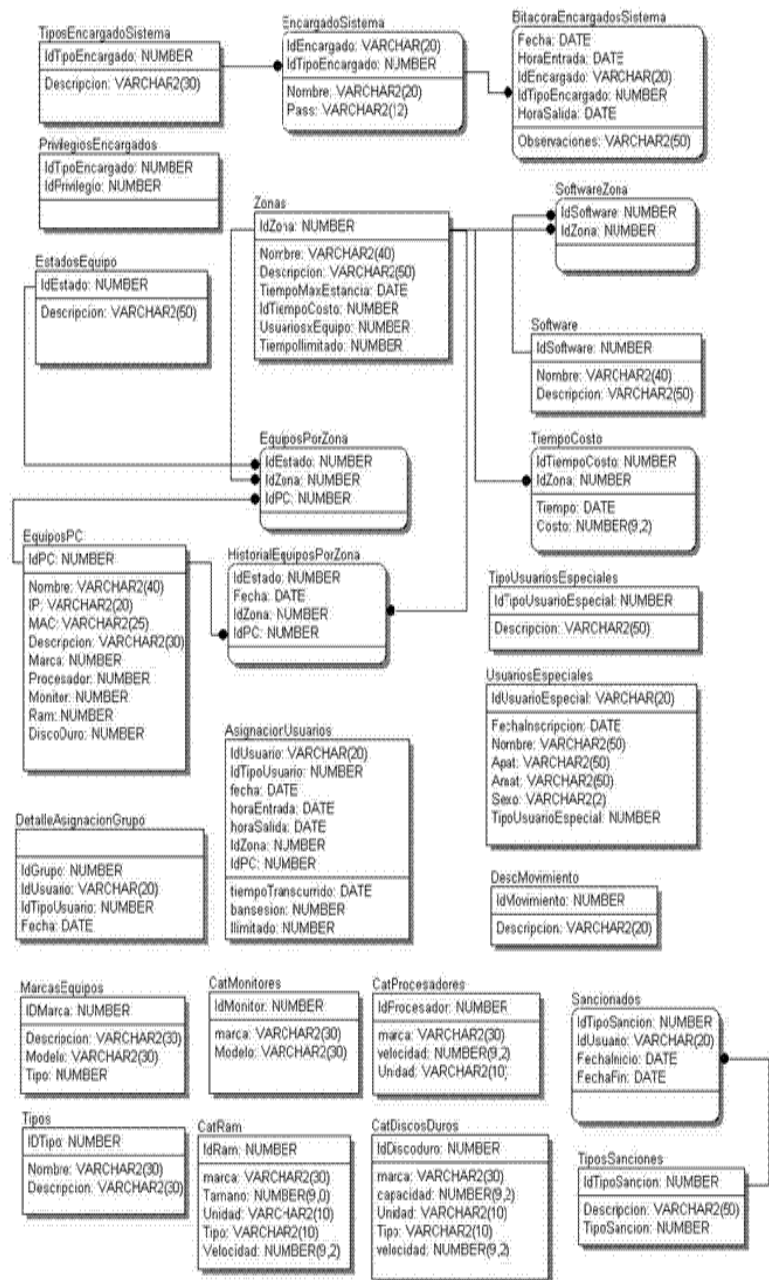


Fig.4 Modelo entidad – relación de SCAM

A lo largo de este apartado, se mostró el desarrollo de SCAM por medio de las fases que comprenden el ciclo de vida y la importancia de la ingeniería de software para la construcción de software de calidad y además se mostró como el desarrollo de este sistema viene a satisfacer las necesidades de

administración y control requeridas para este macrocentro de cómputo en específico.

V. BREVE RECORRIDO POR SCAM

Para conocer el funcionamiento de SCAM a continuación se describen algunas interfaces del sistema; la figura 5 muestra la pantalla principal de SCAM, en ella se puede observar básicamente tres módulos:

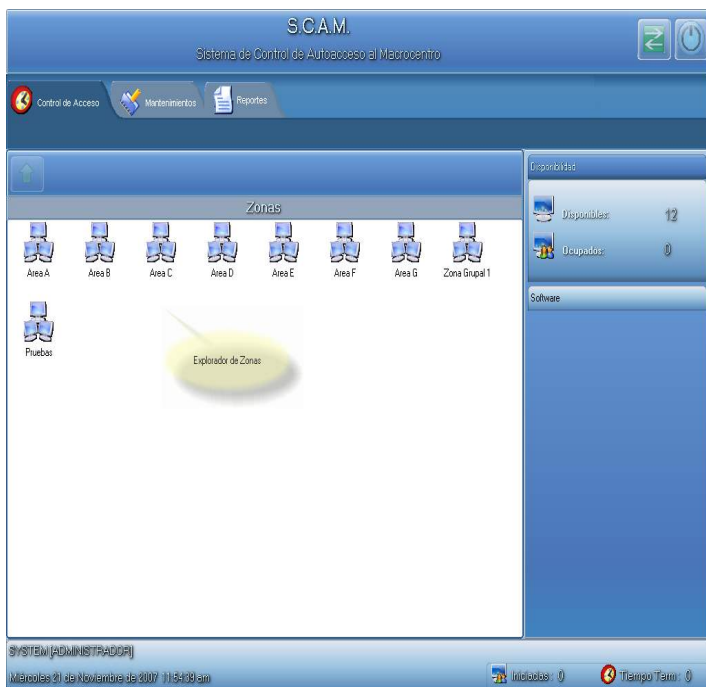


Fig.5 Pantalla principal de SCAM

- **Módulo de Control de acceso**, este permite visualizar todas las zonas de macrocentro de cómputo, su disponibilidad en equipo, así como su software instalado (ver figura 5). En este módulo es donde se realiza la asignación/designación de equipo de cómputo a un usuario.
- **Módulo de Mantenimientos**, (ver figura 6) este permite tener una administración y control de los catálogos de zonas, equipo de cómputo, hardware, software, usuarios y usuarios especiales.

Cada uno de los catálogos tiene sus tareas específicas, por ejemplo: en el *catálogo de zonas* se registran zonas (i.e. lenguajes de programación, diseño, paquetería) y éstas se pueden consultar, actualizar, eliminar; además de esto aquí se asigna el software y los equipos correspondientes a esta zona, para esto tanto hardware como software deben estar previamente registrados, además se permite cambiar equipos de zona a zona (ver figura 7), actualizar software, definir tiempo máximo en la zona, costo de la zona, entre otros.

- **Módulo de Reportes**: en este se generan todos los reportes solicitados por el encargado del macrocentro de cómputo, para llevar una mejor administración y ayuda en la toma de decisiones para el mejoramiento del macrocentro.

La tarea principal de SCAM es la asignación de equipo de cómputo, está se realiza de acuerdo al tipo de usuario (maestro, alumno, personal administrativo), tipo de software a utilizar (paquetería, lenguajes de programación, software especializado, diseño, entre otros), zona a la que desea entrar y tiempo requerido. A continuación se ejemplifica la asignación de un equipo de cómputo (ver figura 8):

- En el *módulo de control de acceso* se muestran las zonas existentes en el macrocentro de cómputo; se selecciona una zona y se verifica la disponibilidad de equipo de cómputo.
- Luego se entra a la zona y al equipo de cómputo seleccionado se le presiona el clic derecho para asignar el equipo.
- Posteriormente aparece una pantalla en la cual se ingresa o se pasa la credencial del usuario, en pantalla aparece su foto y sus datos (carrera, turno, campus, fecha de ingreso, entre otros), luego se le da el botón asignar.

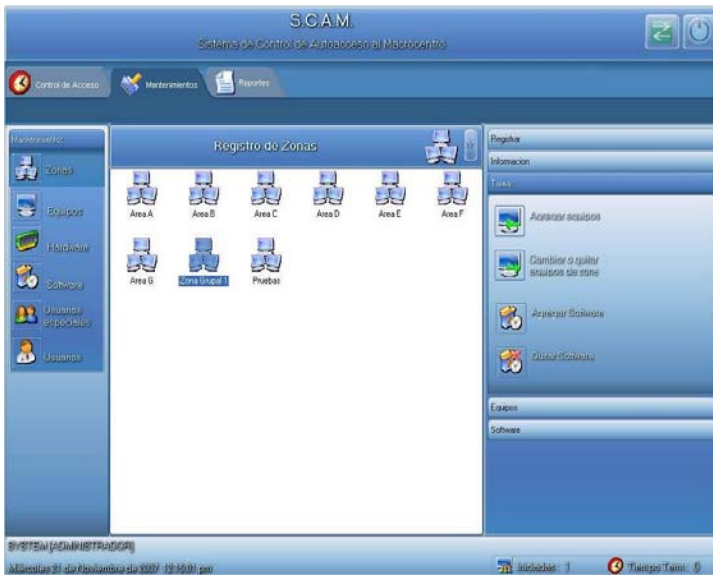


Fig.6 Pantalla Mantenimientos de SCAM

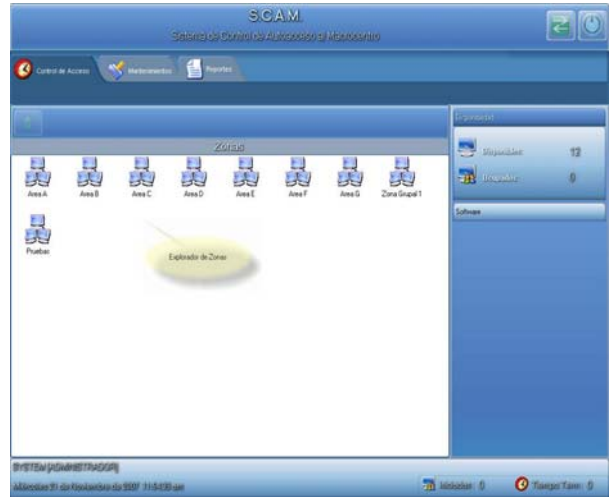


Fig.7 Pantalla Mantenimientos de SCAM que permite cambiar equipos de una zona a otra zona



Fig.8 Pantallas correspondientes a la asignación de un equipo de cómputo

VI. CONCLUSIONES

SCAM es un software específicamente desarrollado para el macrocentro de cómputo de la UABCS. SCAM presenta una interfaz amigable al usuario lo que hace que sea fácil de usar para cualquier persona encargada, al estar en operación se encontró que este sistema es rápido al momento de asignar equipo; permite obtener fácilmente información a través de sus reportes de manera oportuna y precisa, además llevar un buen control de la información del macrocentro en cuanto a los equipos de cómputo, hardware, software, y su utilización.

El desarrollo de SCAM eficienta los procesos del macrocentro de cómputo de la institución; y permite concentrar la información para su procesamiento, consulta, verificación y apoyo en la toma de decisiones en cuanto a las acciones de mejoras del macrocentro de cómputo de la institución.

VII. TRABAJOS FUTUROS

Se propone como trabajo futuro la incorporación de un módulo de punto de venta ya que actualmente el macrocentro cuenta con un área dedicada a la venta de accesorios y servicios digitales.

VIII. REFERENCIAS

- [1] K.C. Laudon y J.P. Laudon. *Sistemas de información gerencial organización y tecnología de la empresa conectada en red*. Ed. Prentice Hall. Sexta ed. México.2002
- [2] J.A. Senn. *Análisis y diseño de sistemas de información*. Ed. Mc Graw Hill. 2da. Ed. Mexico.2003
- [3] Sitio Web de CyberPlanet, URL <http://www.cyberplanetsoft.com/>
- [4] Sitio Web de Smartlunch, URL <http://www.smartlaunch.net/webpages/>
- [5] Sitio Web de CyberPro, URL <http://www.datahousecompany.com.ar/cibercafe.html>
- [6] Sitio Web de ICafe, URL <http://www.mitsoftware.com/icafe.asp>
- [7] Daniel Cohen, Enrique Asín. *Sistemas de Información para los Negocios*. Ed. Mc Graw Hill. 2da. Ed. Mexico.2000
- [8] Jane P. Laudon y Kenneth C. Laudon. *Administración de los Sistemas de Información*. Ed. Prentice Hall. México.1996
- [9] Roger S. Pressman. *Ingeniería del Software. Un Enfoque Práctico*. Ed. Mc Graw Hill. 5ta. ed. México.2002
- [10] IEEE. *IEEE Standards Collection: Software Engineering 610.12-1990*. IEEE 1993
- [11] Craig Larman. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. Ed. Prentice Hall. 1ra.ed. México.1999
- [12] Kenneth E. Kendall & Julie E. Kendall. *Análisis y diseño de sistemas*. Ed. Prentice Hall. 6ta.ed. México.2005

REPLICAS DE INFORMACIÓN UTILIZANDO AGENTES MÓVILES

Israel Guzmán Pérez, Luis Enrique Colmenares Guillen
Benemérita Universidad Autónoma de Puebla (BUAP) Facultad de Computación (FCC)
(222) 2-29 55 00, México, Puebla
ig89852005@hotmail.com, lecolme@cs.buap.mx

RESUMEN

El uso de datos no estructurados es ampliamente usado por diversas Instituciones y Empresas, por lo tanto es necesario tener copia de todos esos datos. Para resolver esto se utiliza la replicación y así tener un manejo distribuido de los datos y no dependa de una arquitectura cliente-servidor. La replicación de datos puede ser solucionado con agentes móviles.

En éste trabajo, se realiza una aplicación donde se pueden actualizar replicas de datos no estructurados a partir del uso de la tecnología de JAVA en conjunto con la plataforma de JADE para la creación de un sistema multi-agentes. Los agentes son capaces de desplazarse a través de distintas redes y distintos hosts.

Palabras clave: agente móvil, JAVA, JADE, replicación de información, sistema operativo.

I. INTRODUCCIÓN

La motivación de este trabajo es hacer un agente capaz de desplazarse por distintas redes y distintos hosts, es decir, que el agente se desplace a través de sistemas heterogéneos con el objetivo de replicar datos, estos datos pueden ser no estructurados (archivos de word, pdf, excel, etc.) hasta incluso datos estructurados (bases de datos).

Lo primero que se debe tener en cuenta, son los siguientes criterios:

1. Los dos tipos de agentes móviles existentes y el tipo de agente móvil que se va a utilizar.
2. Se debe considerar el tipo de migración que va a usar el agente, se discutirán los dos tipos de migración.
3. Se debe considerar el grado de tolerancia que tendrá el agente móvil ante los posibles fallos en la red.
4. Se considera que el agente debe de ser reactivo, es decir, capaz de responder a cambios del entorno.
5. El agente tiene que ser autónomo, sin embargo, éste debe proporcionar un informe de todas las actualizaciones que se han realizado.

II. TRABAJO RELACIONADO

Los trabajos relacionados con la replicación de datos no estructurados pertenecen principalmente a empresas de TIC (Tecnología de información y Comunicación), entre ellas podemos mencionar a Solution Centre Ltd[3], Software Greenhouse[4], entre otras. Las empresas además de la replicación ofrecen mecanismos de búsqueda capaces de recuperar está información. Con la finalidad que los usuarios que contratan estos servicios aumenta su productividad. La mayoría de estas empresas

generan herramientas dependientes de la plataforma, es decir, del Sistema Operativo.

Para replicar información estructurada (data-marts data-warehouse o bases de datos) en la mayoría de motores de bases de datos incorporan tecnología de agentes móviles por ejemplo: Microsoft SQL Server 2008[5], MySQL[6], entre otros. Lo interesante, es que la replicación se realiza de modo interno, es decir, se replica una base de mysql en otra de mysql. Aunque existen herramientas que ayudan a convertir información de una base de mysql a una de oracle por ejemplo: SQL Developer Migration Workbench[7]. Además existen herramientas para convertir de Oracle a MySQL, Oracle Generic Connectivity and Oracle Transparent Gateway[8]. El agente que se está desarrollando puede ser programado para realizar replicación de un motor de base de datos a otro, por ejemplo, replicar de mysql a oracle y viceversa; actualmente el agente que proponemos solo puede replicar de MySQL a MySQL.

Además existen trabajos que proponen protocolos de control de replicación y además se enfocan en un punto muy importante de la replicación que es la consistencia de los datos entre ellos podemos mencionar a **MARP**[9] (Mobile Agent Enabled Replication Protocols).

También hoy en día se están integrando servicios Web con sistemas multi-agentes con el fin de mejorar los servicios en cuanto a tiempo de respuesta, además evitar la caída de los servicios a través de la replicación (mirrors) y hacer un mejor QoS (Quality of Service), entre estos trabajos se puede mencionar A Simulation Analysis of Multiagent-Based Dynamic Request Placement Techniques[10].

III. ANÁLISIS DEL AGENTE MÓVIL

En esta sección se describen las características principales de los agentes móviles: tipo de migración y tipos de agentes móviles.

Los dos tipos de migración de un agente más utilizados, son los siguientes:

1. Single-hop agent[1]. El agente realiza sólo una migración al host destino, de manera que vuelve al host origen una vez realizadas las tareas asignadas.
2. Multiple-hop agent[1]. El agente migra de host en host ejecutándose en cada uno de ellos. Por tanto realiza muchos saltos antes de volver al host de origen.

Además, se tienen dos tipos de agentes móviles los que tienen una movilidad de tipo:

1. Débil[1], es decir, el agente migra de máquina en máquina pero no se envía información del estado de la ejecución en las máquinas anteriores. Por tanto el agente siempre se ejecuta desde el mismo punto o estado inicial.
2. Fuerte[1], es decir, el agente migra de máquina en máquina pero en éste caso además del código se envía el estado de la ejecución, el proceso mientras migra va cambiando de estado, de manera que la ejecución en una determinada máquina puede continuar desde el punto donde se quedó el anterior.

IV. PROPUESTA PARA LA IMPLEMENTACIÓN DEL AGENTE MÓVIL

En la propuesta es conveniente usar un agente que utilice migración multiple-hop ya que realiza la misma acción en cada uno de los hosts, el inconveniente, es que puede surgir un fallo en la red, entonces el agente puede perderse y se desconoce si se realizó la replicación de información.

Para prevenir éste tipo de fallo se puede tener un agente monitor, el cual esté monitoreando al agente móvil y éste agente monitor, informe de algún posible fallo del agente móvil (ver Fig. 1).

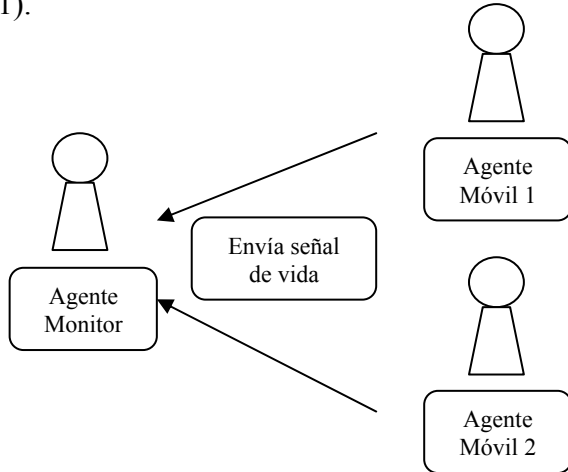


Figura 1. Mensajes del Agente Móvil al Agente Monitor.

Puesto que el agente debe actualizar datos en distintos hosts, es necesario que éste guarde el estado de su ejecución para así conocer los hosts en los que ya ha estado. Además, el agente lleva consigo todos los datos que debe actualizar, por lo tanto, es conveniente el uso de un agente con movilidad fuerte.

Otra característica del agente, es que sea reactivo, ya que si se planea que funcione a través hosts heterogéneos (ver Fig. 2), el agente debe de ser capaz de detectar el tipo de sistema operativo, para que de acuerdo a éste sea capaz de realizar las acciones necesarias sobre ficheros, ya que se podría necesitar de alguna llamada al sistema operativo para realizar cierta acción sobre un fichero; sobre el acceso a las base de datos, se pueden usar funciones de JAVA para acceso a base de datos por medio de JDBC.

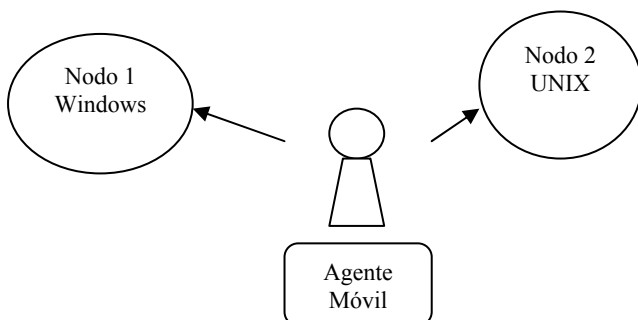


Figura 2. El agente móvil se puede desplazar a través de distintos hosts.

A. PLATAFORMA JADE

Para la implementación del agente se usa la plataforma de Java Agent DEvelopment Framework JADE[1][2]. Esta plataforma simplifica la implementación de sistemas multi-agente a través de un Middleware que usan las especificaciones de Foundation for Intelligent Physical Agents, FIPA[1]. Dentro de los estándares de FIPA, actualmente la comunicación entre agentes se hace a través de ACL (Agent Communication Language), este estándar sustituye a KQML[1].

Además, JADE ofrece al agente, la funcionalidad de poder migrar o clonar a distintos hosts ubicados en distintas redes. Esta funcionalidad, se realiza con su ontología. Es decir, esta ontología contiene todos los conceptos y acciones necesarias para la movilidad de los agentes y la clonación de agentes.

En la versión de JADE 3.5 la movilidad del agente tiene soporte intra-plataforma e inter-plataforma.

El agente móvil puede navegar a través de diferentes contenedores pero está confinado a una sola plataforma de JADE, es decir intra-plataforma.

B. IMPLEMENTACIÓN DEL AGENTE MÓVIL

A partir de algunas propiedades de la plataforma JADE (Inciso A) y las características del agente, se propone la creación de dos agentes:

- ✓ Un agente monitor, es responsable de vigilar al agente móvil. Este agente monitor, informará cuando el agente móvil quede extraviado debido a la caída de alguno de los hosts y además, éste agente tomará las medidas necesarias para crear a un nuevo agente móvil si es necesario.
- ✓ Un agente móvil, es responsable de la actualización de los datos en los distintos hosts. Se recuerda que estos datos pueden ser ficheros. Además, el agente móvil debe de interactuar con el sistema operativo, es decir, que el agente necesite ocupar servicios para el manejo de ficheros.

Se considera que el agente móvil tiene que informar su estado de actividad al agente monitor. Para saber, que el agente móvil aún está funcionando. El agente móvil y el agente monitor tendrán que ser sociables entre sí, es decir, un agente sociable es aquel que puede interactuar con otros agentes.

Por lo anterior, el agente monitor tiene el siguiente funcionamiento (ver Fig. 3):

- El agente monitor espera información por parte del usuario, es decir, espera a que el usuario le proporcione un fichero para actualizar.
- El agente monitor, espera las notificaciones del agente móvil para saber donde se llevó a cabo la actualización.
- Se crean clones del agente móvil, para que cada clon posea toda la información proporcionada por el usuario, y así cada clon pueda realizar las acciones necesarias para la replicación de información.

- El ciclo de vida del agente monitor no termina a menos que el usuario le indique al agente monitor terminar su ejecución.

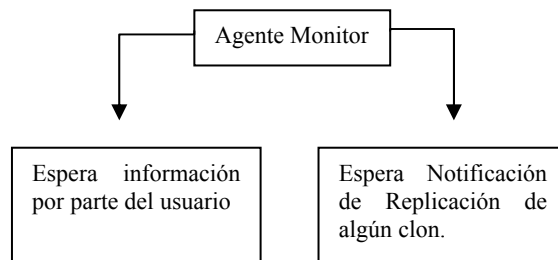


Figura 3. Behavior del Agente Monitor

El ciclo de vida del agente móvil será el siguiente (ver Fig. 4):

- El agente móvil es un clon creado por el agente monitor.
- El agente móvil reúne toda la información necesaria para replicar los ficheros o las bases de datos.
- Una vez hecho lo anterior el agente verifica los hosts a los cuales puede moverse.
- Una vez que el agente comienza a moverse y llega a su destino, comienza a buscar el archivo o base de datos, si es que lo encuentra actualiza el fichero o base de datos y si no crea una copia de ese fichero o base de datos.
- Terminada la replicación de los datos, entonces el agente envía un mensaje de notificación al agente monitor.
- A continuación el agente móvil vuelve a buscar los hosts a los cuales puede moverse y repite el procedimiento hasta que haya visitado todos los hosts.
- Cuando el agente terminó de visitar todos los hosts, termina su ciclo de vida.

En la replicación de la información no importa el Sistema Operativo, ya que el agente es capaz de buscar y reemplazar ficheros a través de distintas arquitecturas, además el agente móvil puede transportar archivos sin importar el tipo o el tamaño, el agente sólo está limitado por las características físicas de la computadora y el ancho de banda; la base de datos que se desea replicar debe contener un campo que especifique si está modificado, así el agente solo toma los datos modificados recientemente.

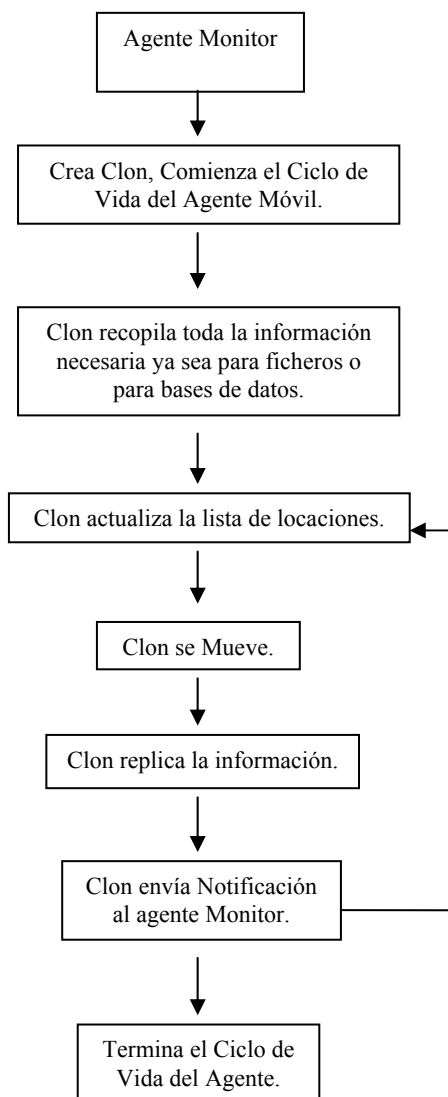


Figura 4. Ciclo de vida de los agentes

V. CONCLUSIÓN

Los agentes son una buena solución para la replicación de información ya que tienen tolerancia a fallos en la red, además la cantidad de información que puede llevar un agente consigo no tiene límite excepto las propias características de hardware de las computadoras como memoria, almacenamiento y ancho de banda, además de que se puede comenzar con un sistema simple de agentes y este sistema puede escalar a un gran sistema multi-agentes (ver sección VI).

VI. TRABAJO A FUTURO

Algunos retos que se tienen para mejorar la propuesta en la replicación de información, son los siguientes:

- Crear un behavior[1] en donde el agente se le proporcione una lista de ficheros o una lista de bases de datos para que el agente los monitoree y cada cierto tiempo se encargue de replicar esos ficheros en los distintos hosts a los cuales puede desplazarse. En éste caso el usuario ya no tendría que decirle al agente cuándo replicarlos sino que el agente sería capaz de tomar la decisión por sí mismo.
- Además se puede crear un mejor algoritmo para la detección de fallas en la red, para que el agente pueda desplazarse sin ningún problema y se garantice con fiabilidad alta que realmente se lleva a cabo la replicación de información.

Una vez que se lleven a cabo los dos puntos anteriores, el trabajo se puede extender a un sistema multi-agentes más complejo. Así, a partir de cualquiera de los hosts se pueda realizar la replicación de información; la replicación se da, si existen los ficheros origen

en una maquina y a partir de esa máquina se comienzan a replicar los ficheros (ver Fig. 5).

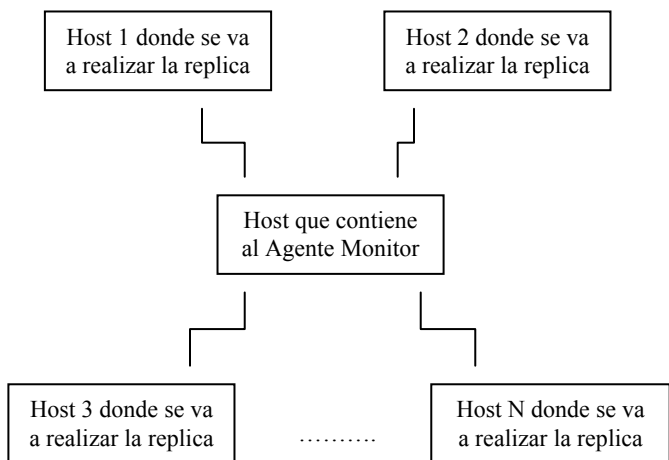


Figura 5. Esquema de la Forma en que se realiza la replicación.

Si se desea crear un sistema donde se pueda comenzar la replicación desde cualquiera de los hosts (ver Fig. 6). Se puede tener un problema de concurrencia. Es decir, se puede dar el caso que desde distintos hosts se intente replicar el mismo fichero, entonces puede haber dos ó más agentes intentando modificar el mismo fichero en la misma computadora al mismo tiempo, por lo tanto, se propone lo siguiente:

- En cada máquina se encuentre un agente monitor el cual se va a encargar de controlar el acceso a los ficheros o a las bases de datos.
- El agente monitor debe utilizar técnicas de concurrencia.
- Cuando un agente móvil llegue a un hosts antes de modificar algún fichero tiene que solicitar permiso al agente monitor local para acceder a ese fichero.

Otra mejora que se propone es que el agente pueda replicar bases de datos, de manera heterogénea, es decir, no importa el motor de bases de datos que se esté usando (MySQL, ORACLE, SQLServer), el agente debe de ser capaz de replicar la base de datos de un motor a otro, esto se lograría a través de JDBC, es decir, si existe un controlador JDBC para ese motor de base de datos, el agente se puede conectar y acceder a dicha base y así tomar todos los datos a replicar.

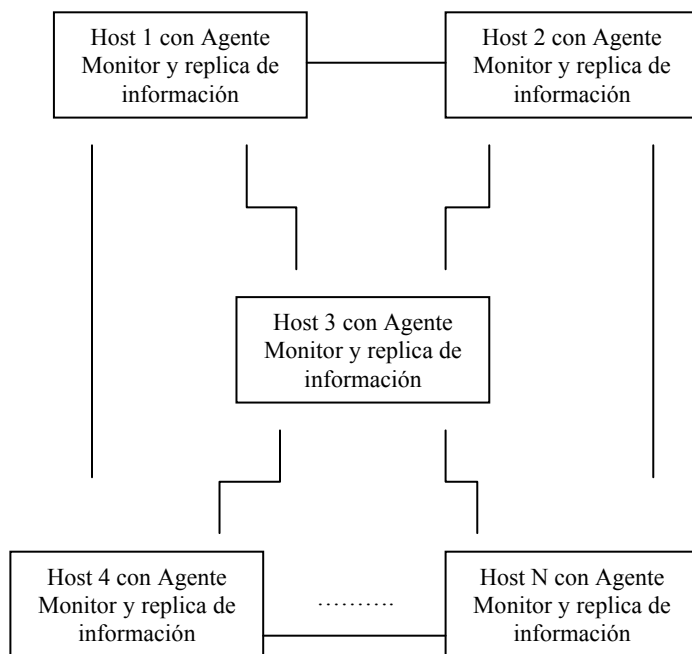


Figura 6. Esquema de la Forma en que se realiza la replicación de acuerdo al trabajo a futuro.

VII. RECONOCIMIENTOS

A la Facultad en Ciencias de la Computación (FCC) de la Benemérita Universidad Autónoma de Puebla (BUAP).

VIII. REFERENCIAS

- [1] Developing Multi-Agent Systems with JADE Fabio Bellifemine, Giovanni Caire, Dominic Greenwood Editorial WILEY
- [2] <http://jade.tilab.com>, 2008
- [3] <http://www.solutioncentre.co.uk/products.php?productid=114>, 2008
- [4] <http://www.swgreenhouse.com/Productos/H-i-Spins/Facilidades.html>, 2008
- [5] [http://msdn.microsoft.com/es-es/library/ms152477\(SQL.100\).aspx](http://msdn.microsoft.com/es-es/library/ms152477(SQL.100).aspx), 2008
- [6] <http://dev.mysql.com/doc/refman/5.0/es/replication.html>, 2008
- [7] http://www.oracle.com/technology/tech/migration/workbench/index_sqldev_omwb.html, 2008
- [8] http://www.oracle.com/technology/products/oracle9i/datasheets/gateways/gateway_rel2ds.html, 2008
- [9] Achieving Replication Consistency Using Cooperating Mobile Agents; *Giannong Cao, Alvin T.S. Chan, Jie Wu*
- [10] A simulation Analysis of Multi-Agent Based Dynamic Request Placement Techniques; Felix F. Ramos y Francisco Velasco

APÉNDICE A: PANTALLAS DE LA PROPUESTA

En la Fig. 1, la pantalla de la parte superior izquierda corresponde a la GUI (Graphical User Interface) del agente móvil, en la parte inferior derecha se tiene la GUI de JADE. En la Fig. 2, se puede ver la ejecución del agente móvil cuando replica un archivo. En la Fig. 3, se puede apreciar la información que envía el agente móvil desde su nueva locación.

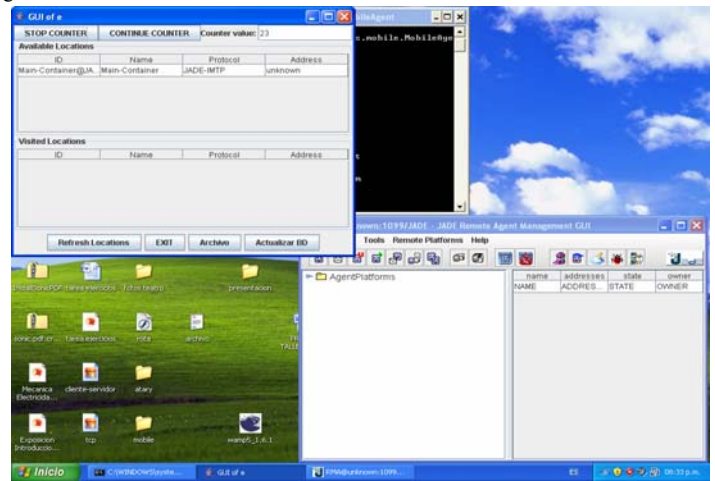


Figura 1: GUI del Agente Móvil

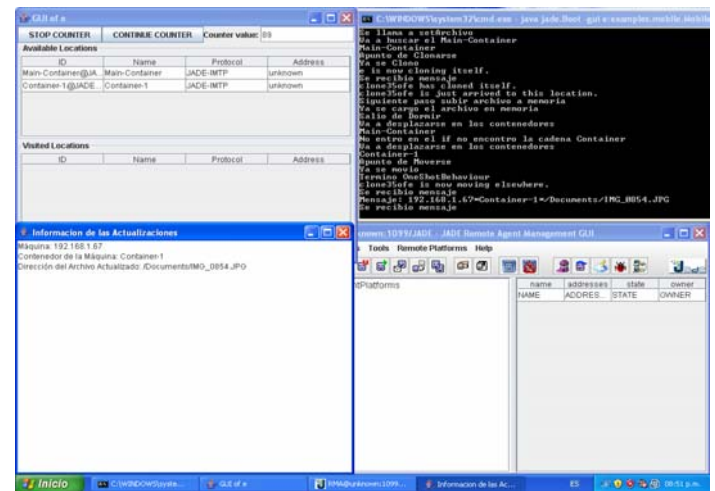


Figura 2: Ejecución del Agente Móvil

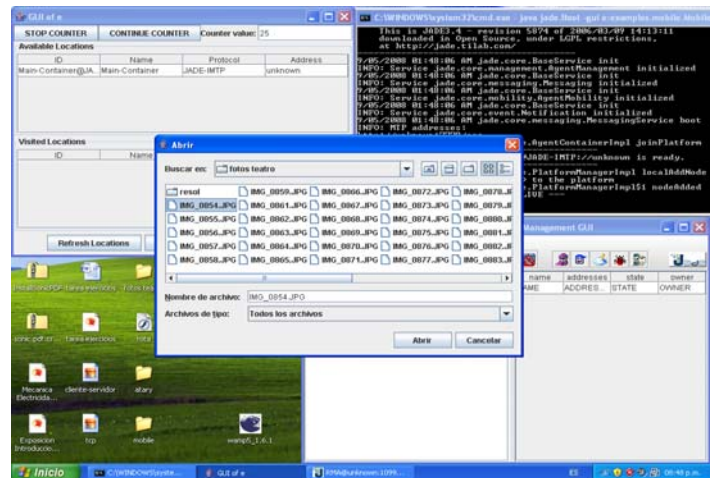


Figura 3: Mensajes del Agente Móvil al Agente Monitor.

APLICACIÓN DE REDES DE PETRI PARA EL DISEÑO DE CONTROL LÓGICO

Bernardo Haro M.†, Karen Hernández R.‡

*†Universidad Autónoma de Guadalajara
Departamento de electrónica
Av. Patria 1201, Zapopan Jalisco. CP 45110
bharo@uag.mx*

*‡Universidad de Guadalajara
Cucsur-Teleinformática
Av. Independencia Nacional 151
Autlán, Jalisco. CP48900
karenhr@cucsur.udg.mx*

RESUMEN

Este artículo muestra el ciclo de diseño de un controlador lógico utilizando Redes de Petri Interpretadas. Se emplean técnicas ya existentes durante el ciclo de diseño introduciendo dos nuevas partes, que son la metodología de modelado de la especificación y el diseño del controlador. El controlador diseñado es implementado en un controlador lógico programable (PLC).

Palabras clave: Automatización, Redes de Petri, Sistemas de eventos discretos

I. INTRODUCCIÓN

Este trabajo presenta un caso práctico de cómo usar las Redes de Petri Interpretadas (RPI) para diseñar un controlador e implementarlo en un Controlador Lógico programable (PLC por sus siglas en inglés). La ventaja de contar con la metodología que aquí se presenta radica en que todo el ciclo de diseño, modelado, especificación, diseño del controlador e implementación del controlador en un PLC se formaliza con las RPI y la etapa de depuración de los programas implementados es eliminada, ahorrándose dinero y tiempo, ya que esta etapa es la que resulta más cara.

El flujo de diseño que aquí se sigue es:

1. Obtener un modelo en RPI del sistema que se necesita controlar. En el caso aquí presentado se utilizó la metodología citada en [2].
2. Obtener un modelo en RPI de la especificación. La metodología que aquí se presenta es propia, pero está basada fuertemente en la metodología del punto previo.
3. Diseño del control. La metodología aquí propuesta está basada en las técnicas de control supervisor y el control por regulación citadas en [7], [10].
4. Implementación del controlador en PLC. La metodología de traducción de RPI a diagramas de escalera (PLC) es la propuesta en [8].

Este trabajo está organizado como sigue. La sección II resume los conceptos relacionados con las RPI. La sección III presenta el sistema que se controla así como el comportamiento requerido del mismo. Usando este ejemplo se muestra la metodología de modelado del sistema y se presenta la metodología de modelado de la especificación. La sección IV presenta cómo construir el controlador y cómo traducirlo a diagramas de escalera. La sección V presenta los resultados prácticos para implementar el controlador en el

sistema real y finalmente se presentan las conclusiones y trabajo futuro.

II. REDES DE PETRI INTERPRETADAS

Esta sección presenta a las RPI [1], [5] como una extensión de las Redes de Petri [3], [4], [9]. Las RPI han sido ampliamente usadas en automatización y especificación de programas, principalmente donde existe paralelismo. Formalmente las RPI están definidas como sigue.

Definición 1 Una Red de Petri Interpretada (RPI) denotada por (Q, M_0) , es una cuádrupla $Q=(N, \Sigma, \lambda, \varphi)$ donde

- $N = (P, T, I, O, M_0)$ es una RP compuesta de lugares (conjunto P), transiciones (conjunto T), arcos de entrada a las transiciones (función I) y arcos de salida de las transiciones (función O); M_0 es el marcado inicial de la RP.
- $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_r\}$ es el alfabeto de entrada a la red, donde σ_i es un símbolo de entrada.
- $\lambda: T \rightarrow \Sigma \cup \{\varepsilon\}$ es una función de etiquetado de transiciones con las siguientes restricciones: $\forall t_j, t_k \in T, j \neq k$ si $I(p_i, t_j) = I(p_i, t_k) \neq 0$ y ambas $\lambda(t_j), \lambda(t_k) \neq \varepsilon$, entonces $\lambda(t_j) \neq \lambda(t_k)$, donde ε representa un evento interno del sistema y p_i un lugar de $\lambda(t_k)$.
- $\varphi: \mathbf{R}(Q, M_0) \rightarrow \mathbf{Z}^{+q}$ es una función lineal de salida representada por una matriz de $q \times n$, donde $\mathbf{R}(Q, M_0)$ es el conjunto de los posibles estados que alcanza la RPI y q es el total de salidas. El vector de salida $\varphi(M_k)$ de la red es un vector de q elementos.

Definición 2 Si $\lambda(t_i) \neq \varepsilon$, entonces la transición t_i es llamada controlable, de otra forma es llamada no-controlable. Un lugar $p_i \in P$ es llamado medible si el i -ésimo vector columna de φ no es nulo (i.e. $\varphi(\bullet, i) \neq 0$), de otra forma es llamado no-medible. U

Una transición $t_j \in T$ de una RPI está habilitada en el marcado M_k si $\forall p_i \in P, M_k(p_i) \geq I(p_i, t_j)$. Si $\lambda(t_j) = a_i \neq \varepsilon$ está presente y t_j está habilitada, entonces t_j debe dispararse. Si $\lambda(t_j) = \varepsilon$ y t_j está habilitada, entonces t_j puede ser disparada. Cuando una transición habilitada t_j es disparada en un

marcado M_k , entonces un nuevo marcado M_{k+1} es alcanzado. Este hecho se representa como:

$$M_k \xrightarrow{t_k} M_{k+1}$$

Una secuencia de disparos de una RPI (Q, M_0) es una secuencia $\sigma = t_i t_j \dots t_k \dots$ tal que $M_0 \xrightarrow{t_i} M_0 \xrightarrow{t_j} \dots M_w \xrightarrow{t_k} \dots$. El vector de Parikh $\vec{\sigma}: T \rightarrow \mathbf{Z}^+$ de la secuencia de disparo σ mapea cada transición $t \in T$ al número de ocurrencias de t en σ . Entonces el marcado alcanzado desde M_0 al disparar σ se puede calcular con la ecuación de estados de una RPI, la cual se escribe como:

$$\begin{aligned} M_{k+1} &= M_0 + C v_k \\ y_k &= \varphi M_k \end{aligned} \quad (1)$$

donde C es la matriz de incidencia definida como en una RP [3] [9] y $v_k = \vec{\sigma}$.

El conjunto de todas las secuencias de disparo es llamado el lenguaje de disparo y formalmente se representa como $\mathcal{L}(Q, M_0) = \{\sigma \mid \sigma = t_i t_j \dots t_k \text{ y } M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots M_w \xrightarrow{t_k} \dots\}$. Dos lenguajes más pueden ser definidos.

Definición 3 Sea (Q, M_0) una RPI. El lenguaje de entrada de (Q, M_0) es $\mathcal{L}_{in}(Q, M_0) = \{\lambda(t_i) \lambda(t_j) \dots \lambda(t_k) \mid t_i t_j \dots t_k \in \mathcal{L}(Q, M_0)\}$. El lenguaje de salida de (Q, M_0) es:

$$\mathcal{L}_{out}(Q, M_0) = \{\varphi M_0, \varphi M_1, \dots, \varphi M_w \mid$$

$$M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots \xrightarrow{t_k} M_w \text{ y } t_i t_j \dots t_k \in \mathcal{L}(Q, M_0)\}.$$

Gráficamente los lugares se representan por círculos y las transiciones por barras. Estos dos nodos se conectan entre sí respetando las funciones I y O de la RP. Los lugares medibles de una RPI se representan con círculos transparentes, mientras que los no-medibles con círculos opacos. Las transiciones controlables son dibujadas con rectángulos transparentes, mientras que las no-controlables con rectángulos opacos.

III. DESCRIPCIÓN DEL SISTEMA Y METODOLOGÍAS DE MODELADO

El sistema que se requiere controlar consta de:

- Dos puertas de bronce, una superior y otra inferior, denominadas P_s y P_i respectivamente.
- Dos motores, uno superior y otro inferior llamados M_s y M_i .
- Una botonera que consta de dos botones (del tipo "push button"), llamados B_s y B_i .
- Cuatro sensores del tipo límite de carrera (interruptores), llamados $Ss1$, $Ss2$, $Si1$ e $Si2$.

El funcionamiento del sistema es el siguiente:

- Los motores son independientes, cada uno puede girar en el sentido horario y antihorario.
- El motor M_s está conectado por cables de acero a la puerta P_s . Al girar M_s en sentido horario abre P_s , mientras que en sentido antihorario la cierra.
- El motor M_i está conectado por cables de acero a la puerta P_i y al girar M_i en sentido horario abre P_i , mientras que en sentido antihorario la cierra.

La figura 1 muestra al sistema, donde a la derecha se observan las dos puertas (superior e inferior) y a la izquierda, donde se observan los motores y poleas usadas para mover las puertas. La botonera, no está conectada.



Figura 1. Fotografía del sistema. A la derecha se observan las dos puertas y a la izquierda la parte posterior mostrando los motores.

La figura 2 muestra esquemáticamente al sistema. Se ven los sensores $Ss1$ y $Ss2$ para detectar la puerta superior completamente cerrada o abierta, respectivamente. También se muestran los sensores $Si1$ y $Si2$ para detectar que la puerta inferior está

completamente cerrada o abierta, respectivamente. Además se observa la disposición de los motores.

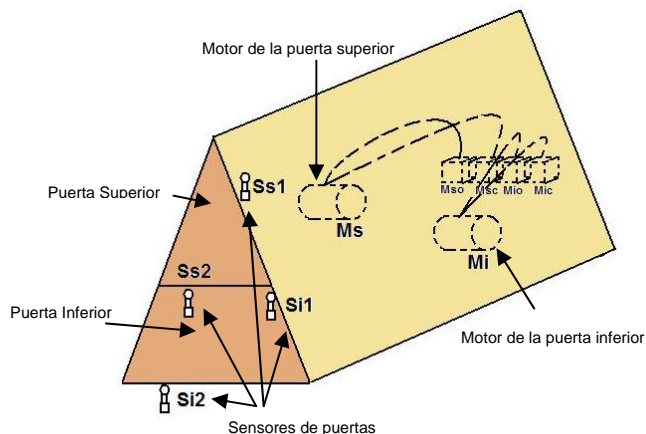


Figura 2. Diagrama esquemático del sistema.

El comportamiento requerido (o especificación) para el sistema es el siguiente:

1. Si las dos puertas están abiertas al iniciar el sistema entonces:
 - Se debe cerrar P_s , y ya que esté cerrada entonces cerrar P_i .
2. Si sólo una puerta está abierta al iniciar el sistema entonces:
 - Se debe mandar cerrar la puerta abierta.
3. Por restricciones espaciales las puertas no deben abrirse simultáneamente.
4. Cuando las dos puertas están cerradas y se presiona el botón B_s , entonces el motor M_s debe girar en el sentido contrario al último giro dado. Si es la primera vez que se presiona B_s , entonces M_s sólo podrá girar en el sentido antihorario.
5. Si se presiona el botón B_i , entonces el motor M_i debe girar en el sentido contrario al último giro dado. Si se está inicializando el sistema, entonces M_i debe girar en el sentido antihorario.

Para modelar al sistema se sigue la metodología propuesta en [2], donde indica:

1. Identificar cada uno de los componentes del sistema (que no sean sensores) y obtener las variables de estado que describen el comportamiento de los componentes.

Como se observa se tienen seis componentes, Ps , Pi , Ms , Mi , Bs y Bi . El comportamiento de los componentes Ps , Pi , Bs y Bi se describe por las variables de estado "posición", ya que de cada uno de ellos sólo importa su posición (puerta abierta o cerrada, botón abierto o cerrado). Para Ms y Mi se utiliza la variable "operación" (motor girando en sentido horario, motor girando en sentido anti-horario o en espera).

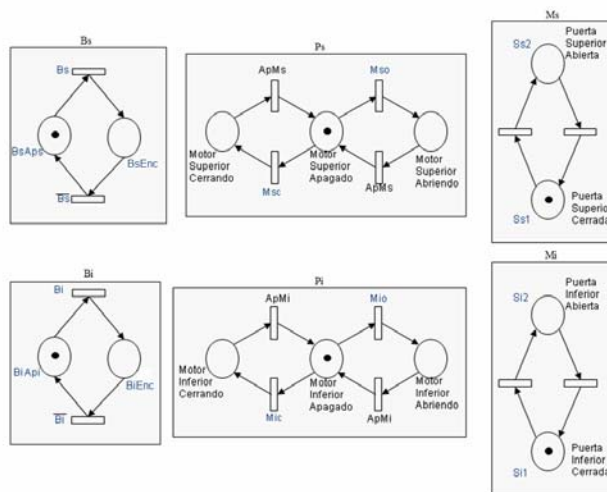


Figura 3. Modelo en RPI de cada una de las variables del sistema.

2. Representar cada variable de estado en Redes de Petri del tipo máquinas de estado binarias.

Llamaremos $posicionPs$, $posicionPi$, $posicionBs$, $posicionBi$ a la posición de Ps , Pi , Bs , Bi respectivamente. Los valores posibles para posición Ps es "puerta superior abierta" o "puerta superior cerrada" y se puede pasar de "puerta superior abierta" a "puerta superior cerrada" en un paso. Por tanto, en el modelo de RPI hay un lugar para la "puerta superior abierta" y otro para "puerta superior cerrada" y se conecta "puerta superior abierta" con "puerta superior cerrada" con una transición (porque se puede pasar de "puerta superior abierta" a "puerta superior cerrada" en un solo paso). También se conecta "puerta superior cerrada" con "puerta superior abierta" por medio de una transición. El modelo de todas las variables se presenta en la figura 3.

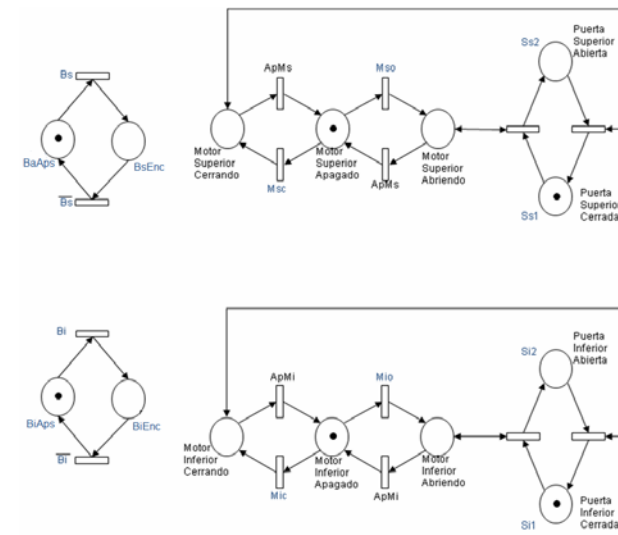


Figura 4. Modelo global del sistema. Note que las botoneras están independientes, mientras que los motores abren las puertas por medio de las poleas existentes.

3. Unir todos los modelos anteriores utilizando el producto síncrono y causal [2]. El modelo global del sistema se presenta en la figura 4.

El siguiente paso es encontrar el modelo de la especificación. La técnica consiste en modelar el comportamiento requerido de cada uno de los componentes del sistema utilizando únicamente los estados encontrados durante el modelado y después incluir las restricciones requeridas, utilizando el diseño de supervisores presentado en [11].

Una vez que ha sido obtenido el modelo, se procede a su análisis. Puede ocurrir que el sistema ejecute únicamente lo que se requiere. En este caso ya no hay necesidad de controlador. Sin embargo son pocas las veces en que esto ocurre por lo tanto, se debe diseñar el controlador.

1. Para el sistema bajo estudio se encontró el comportamiento de las seis componentes: Ps , Pi , Ms , Mi , Bs y Bi . De acuerdo a la especificación, los motores podrán girar en cualquiera de los dos sentidos o quedarse en reposo. Las puertas podrán estar abiertas o cerradas y cuando se presione un

combinaciones pueden resultar en condiciones de arranque inoperables.

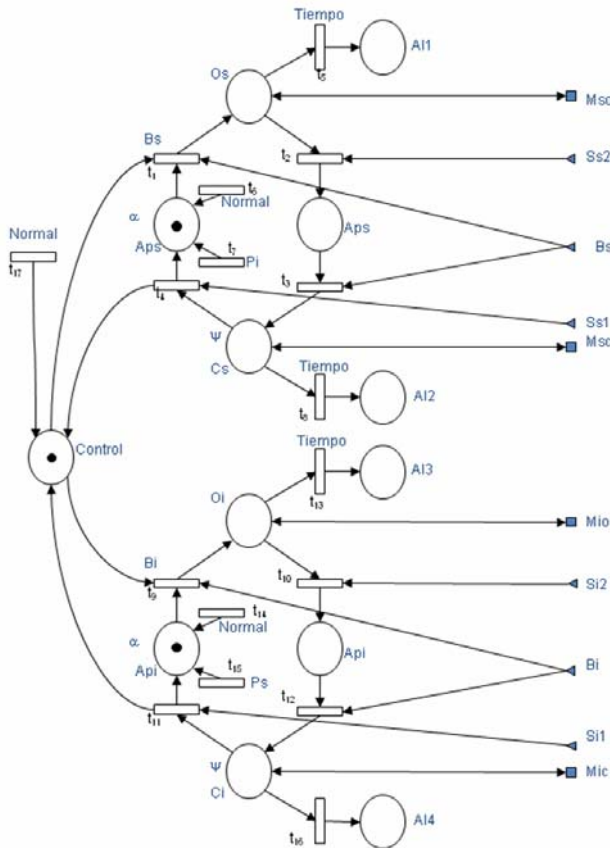


Figura 6. Modelo del controlador. Note que además se han incluido transiciones de alarma. Si el motor está encendido por cierto tiempo y la puerta no alcanza su estado especificado, entonces se dispara la alarma.

Si la condición de arranque resulta inoperable, el sistema debe detenerse y no realizar secuencia alguna. En otras ocasiones una combinación de los sensores deberá iniciar una secuencia de arranque. La tabla 1 muestra las condiciones de arranque para el ejemplo del trabajo, se presentan casos de error, de uso de secuencias de arranque y de marcar condiciones iniciales dentro del comportamiento normal. Recordar que: el sensor Ss_1 detecta cuándo la puerta superior está cerrada, el sensor Ss_2 detecta cuándo la puerta superior está abierta, el sensor Si_1 detecta cuándo la puerta inferior está cerrada y el sensor Si_2 detecta cuándo la puerta inferior está abierta.

Ss1	Ss2	Si1	Si2	Marcado	Significado
0	0	0	0		Error
0	0	0	1	Scs	Cerrar la puerta superior y luego la inferior
0	0	1	0	Cs,αApi	Cerrar la Puerta Superior
0	0	1	1		Error
0	1	0	0		Error
0	1	0	1		Error
0	1	1	0	Cs,αApi	Cerrar la Puerta Superior
0	1	1	1		Error
1	0	0	0	αAps,Ci	Cerrar la Puerta Inferior
1	0	0	1	αAps,Ci	Cerrar la Puerta Inferior
1	0	1	0	αAps,αApi,Control	Condición Normal
1	0	1	1		Error
1	1	0	0		Error
1	1	0	1		Error
1	1	1	0		Error
1	1	1	1		Error

Tabla 1 de condiciones iniciales.

Como ejemplo de lectura de la tabla tome la condición $[0 \ 0 \ 0 \ 1]$, la cual indica que la puerta superior está entre abierta (ni abierta ni cerrada) y la puerta inferior está abierta. Para que las puertas no choquen, se debe cerrar primero la puerta superior y luego la puerta inferior, es decir se debe seguir esa secuencia de arranque (orden del cerrado de puertas). Otra condición en la tabla es $[0 \ 0 \ 1 \ 0]$, indicando que la puerta superior está entre abierta y la inferior cerrada. Por lo tanto se debe cerrar la puerta superior. Note que en este caso no hay secuencia, por tanto sólo se deben poner las marcas en el lugar indicado del controlador para que se tenga el efecto requerido. En este caso se pone marca en Cs para que se cierre la puerta superior y αApi para que la puerta inferior permanezca cerrada. Otra condición en la tabla es $[1 \ 0 \ 1 \ 0]$, esta condición indica que las dos puertas cerradas, por lo tanto deben marcarse los lugares αAps, αApi para que ambas puertas permanezcan cerradas, y se marca el lugar llamado control para que permita el exclusión mutua (sólo puede trabajar una puerta a la vez). La figura 7 muestra la secuencia de

arranque. Note que primero manda cerrar la puerta superior al energizar el motor superior (señal *Msc*) y después la puerta inferior al energizar el motor inferior (señal *Mic*). El cambio de energizar *Ms* a energizar *Mi* se da cuando la transición detecta que ya se cerró el motor superior (señal *SsI*). Cuando se detecta que ya se cerró la puerta inferior (señal *SiI*) entonces se pasa a condición normal para que trabaje el controlador de la figura 6.

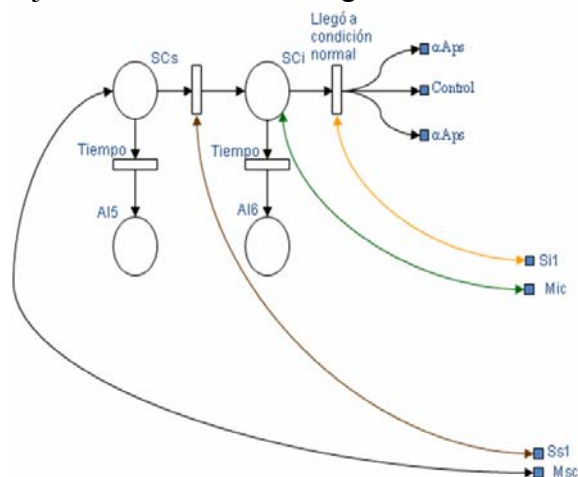


Figura 7. Secuencia de arranque para la condición: [0 0 0 1] de la tabla de condiciones iniciales.

V. RESULTADOS PRÁCTICOS

En este apartado, la tabla de condiciones iniciales, la secuencia de arranque y el controlador deben ser traducidos a diagramas de escalera.

Cada condición de la tabla de condiciones iniciales se traduce como un peldaño en el diagrama de escalera. Un cero en la señal del sensor indica que debe traducirse como un contactor normalmente cerrado, mientras que un uno indica que debe usarse un contactor normalmente abierto. Los lugares que se deben marcar, según la tabla de condiciones iniciales, se ponen como una bobina que enclava la señal que representa el lugar.

Para traducir la RPI del controlador a un diagrama de escalera, se procede con la traducción transición a transición. Cada peldaño del diagrama de escalera representa una transición en la RPI de controlador. Los lugares de entrada y las señales asociadas a la transición se transforman en contactores normalmente abiertos. Las bobinas del

peldaño enclavan los lugares de salida de la transición y limpian (ponen a valor falso) los lugares de entrada de la transición.

La RPI de la secuencia de arranque se traduce de igual forma a la del controlador.

Para poder realizar las pruebas al sistema se conectaron las entradas y salidas del PLC a los sensores y actuadores respectivamente. La figura 8 ilustra el diagrama de escalera obtenido.

Previamente se hicieron las conexiones y cableados de los 4 sensores de límite de carrera, la botonera y los 4 relevadores que permiten encender los motores.

Al realizarse las pruebas al sistema con el PLC conectado se pudo comprobar que éste cumple con las especificaciones y está libre de errores.

VI. CONCLUSIONES Y TRABAJO FUTURO

Se mostró que el uso de herramientas formales (como las RPI) permiten obtener controladores correctos (libre de errores) y automatizan todo el ciclo de diseño de controladores. Por lo tanto la etapa de depuración de programas es eliminada. Actualmente se está trabajando en aplicar estas ideas a redes de alto nivel.

VII. REFERENCIAS

- [1] L. Aguirre, A. Ramírez, O. Begovich. *Design of asymptotic observers for discrete event systems. Proceedings of the IASTED International Conference on Intelligent Systems and Control. Santa Barbara, pp. 188-193. October 1999.*
- [2] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel and E. López-Mellado. *On-line Fault Diagnosis of Discrete Event Systems. A Petri Net Based Approach. Transactions on Automation science engineering 2007, Vol. 4-1, pp 31-39*
- [3] Jörg Desel and Javier Esparza. *Free choice Petri nets, Cambridge University Press, 1995.*

[4] F. DiCesare, G. Harhalakis, J. M. Proth, M. Silva and F.B. Vernadat. *Practice of Petri Nets in Manufacturing*. Chapman & Hall Ed. Great Britain, 1993.

[5] M. E. Meda, A. Ramírez and A. Malo. *Identification in discrete event systems*. IEEE International Conference Systems, Man and Cybernetics, pp. 740-745 October 1998.

[6] T. Murata. *Petri nets: Properties, analysis and applications*. Proc. IEEE, Vol. 77, no 4 pp. 541-580, April, 1989.

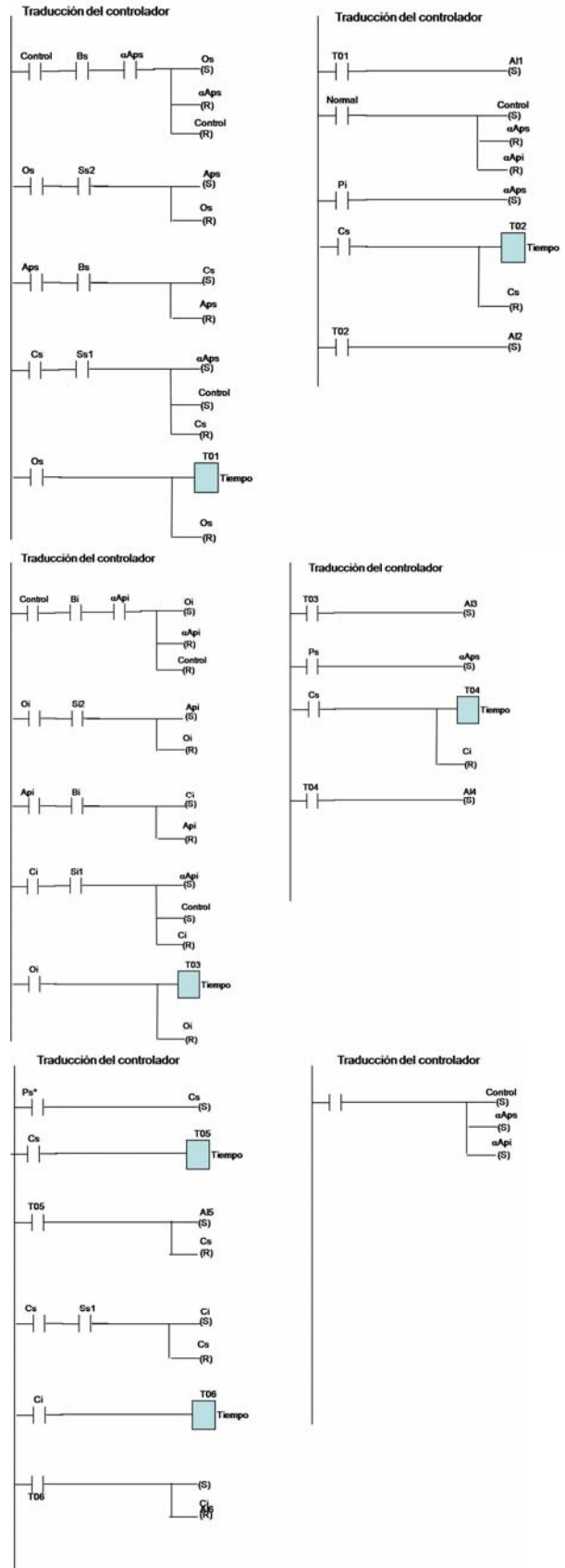
[7] A. Santoyo, O. Begovich, A. Ramírez, *Model Tracking in DES Using Interpreted Petri Nets*, Proceeding of the International Symposium on Robotics and Automation: Saltillo, México, pp. 359-363, 1998.

[8] A. Santoyo, I. Jiménez-Ochoa, A. Ramírez Treviño. *A Complete Cycle for Controller Design in Discrete Event System Proceedings of the IEEE conf. On Systems, man and cybernetics*, pp 2688-2693, Arizona, EEUU 2001.

[9] Silva : M. Silva. *Las Redes de Petri: en la automática y la informática*. Editorial AC, Madrid, España, 1985.

[10] G. Ramírez-Prado, A. Santoyo, A. Ramírez-Treviño and O. Begovich, *Regulation problem in discrete event systems using interpreted Petri nets*. Proceedings of the IEEE con. on Systems, Man and Cybernetics 2000, pp 2174-2179.

[11] J. O. Moody, P. J. Antsaklis, *Supervisory Control of Discrete Event Systems Using Petri Nets*. Ed. Kluwer Academic Publishers, Estados Unidos, 1998.



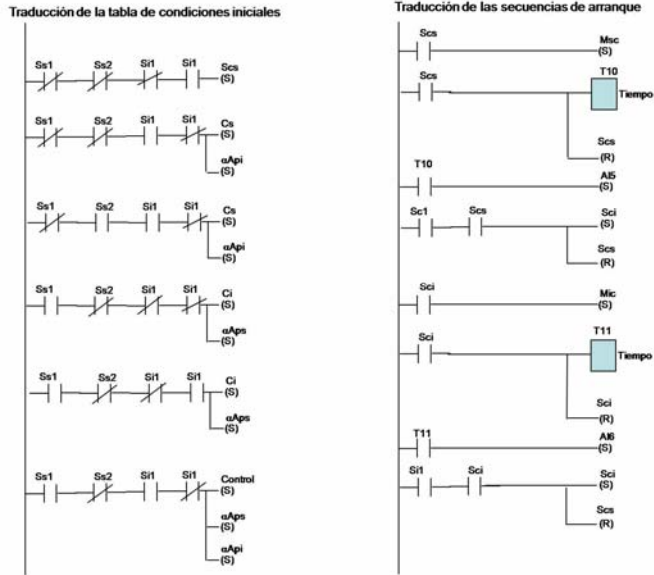


Figura 8. El diagrama de escalera del controlador

DISEÑO DE UNA HERRAMIENTA DE SIMULACIÓN DE SISTEMAS INDUSTRIALES BASADA EN REDES DE PETRI *

Santoyo-Sanchez Alejandra¹, Guzmán-Hernández Héctor Miguel¹,
Alcalá-García Bruno Daniel¹, Aguirre-Salas Luis Isidro², y
De Jesús-Velasquez Carlos Alberto³.

¹Departamento de Computación, División de Electrónica y Computación, CUCEI
Universidad de Guadalajara
Av. Revolución No. 1500 Modulo "O", Col. Olímpica, CP. 44430, Guadalajara, Jalisco, México
Tel. (33) 36398038, correos electrónicos: alejandra.santoyo@cucei.udg.mx,
hemi85@gmail.com, bruno.alcala@gmail.com.

²Departamento de Ingenierías, División de Desarrollo Regional, Centro Universitario de la Costa Sur (CUCSUR)
Universidad de Guadalajara
Av. Independencia Nacional No. 151, CP 48900, Autlán de Navarro, Jalisco, México
Tel. (317) 3825010, correo electrónico: laguirre@cucsur.udg.mx

³Compatibility Validation
Intel Guadalajara Design Center
Parque Industrial Tecnológico II, Periférico Sur 7980 Edificio 4E, Col. Santa Maria Tequepexpan, C.P. 45600,
Tlaquepaque, Jalisco, México
Tel. (33) 3540-6215, correo electrónico: carlos.de.jesus@intel.com

RESUMEN

Este artículo, se enfoca en el diseño de una nueva herramienta para el modelado y simulación de sistemas industriales (SSI) basada en Redes de Petri. En primer lugar, se presenta un breve estudio sobre las herramientas para el modelado y simulación de Sistemas de Eventos Discretos. En segundo lugar, la arquitectura de la nueva herramienta denominada para fines generales SSI; la cuál está diseñada para alcanzar 1) flexibilidad, en que se basa la plataforma estándar de la industria, 2) extensibilidad, ya que puede ser ampliado por las funciones programadas por el usuario, y pueden tener acceso a funciones matemáticas y las bibliotecas, y 3) lenguaje de programación y sistemas operativos neutral, como los modelos se pueden definir en XML, y EXCEL.

Palabras clave: Sistemas Industriales, Sistemas de Eventos Discretos, Redes de Petri, Herramienta.

I. INTRODUCCIÓN

En la actualidad, es un hecho que la industria de procesos está tomando pasos seguros para

consolidar su automatización utilizando las herramientas más modernas disponibles en el mercado. Uno de los aspectos más importantes durante el diseño formal de automatizaciones con Controladores Lógicos Programables (en inglés Programmable Logic Controllers PLC's) consiste en la elaboración y utilización de herramientas que auxilien durante su diseño e implementación.

Las tareas involucradas con el diseño de automatizaciones son: diseño de la especificación o tareas requeridas, calendarización de actividades (scheduling), asignación de recursos, optimización de recursos y tiempos, movimientos y balanceos de cargas, entre otros. Generalmente, estas tareas son resueltas usando técnicas heurísticas a través de herramientas convencionales como PERT, GERT, CPM, adaptadas a esquemas de producción fijos. Lo que provoca que el diseño de automatizaciones sea algo complejo y poco exacto.

Ante este panorama, este artículo se enfoca en el caso de la simulación de sistemas industriales, y el desarrollo de una nueva herramienta en este sentido. En este artículo se describe el desarrollo de una nueva herramienta denominada SSI basada

* Supported by project PROMEP/103.5/07/2636

en Redes de Petri que está diseñada para facilitar el modelado, simulación, ejecución y evaluación de sistemas de eventos discretos.

Este documento está estructurado de la siguiente manera: En la siguiente sección (sección 2), se presenta una breve encuesta relativa a herramientas informáticas para la simulación de sistemas industriales y de eventos discretos. La sección 3 describe las principales cuestiones sobre la arquitectura de SSI. Mientras que en la sección 4 se analizan los pasos que están incluidos en la simulación, y en la sección 5 se presenta un ejemplo de aplicación del simulador. Finalmente en la sección 6 se presentan las conclusiones y trabajo futuro.

II. REVISIÓN DE LA LITERATURA

Un Sistema Industrial (SI) es un sistema automatizado de fabricación que consta de máquinas de control numérico capaz de realizar múltiples funciones y un sistema automatizado de manejo de materiales, todos controlados por un sistema informático [4] y [9]. Un aspecto importante para la descripción de los SI (modelo de empresa) es la especificación del producto, definición de la ruta a seguir para su fabricación y los recursos activos (máquinas) con que cuenta su planta de producción.

En el caso de la simulación de sistemas industriales diversas herramientas de software se utilizan para acelerar la creación y simulación de modelos computacionales. Esta sección presenta un breve estudio sobre estas herramientas de software enfocadas en el proceso de modelado de los SI para su representación y análisis, en el área de planificación y control de la producción.

Existen en general diversas técnicas para el modelado de SI cada uno captura diferentes aspectos, y cada uno tiene ventajas y desventajas, entre las cuales se encuentran las siguientes.

1. *Máquinas deterministas*. Describen el SI como una secuencia fija de actividades bien definidas para convertir los insumos en productos con el fin de lograr objetivos claros. Para esta perspectiva, un modelado de producción estática (componentes e interacciones) es suficiente, con técnicas como la definición de métodos integrados (IDEF0, IDEF3),

diagramas de actividad (RAD, DFD), PERT, GERT, CPM [1, 2].

2. *Sistemas dinámicos complejos*. Este segundo punto de vista se centra en el comportamiento dinámico, interactivo y características de los planes de procesos. Para lo cual se emplean formalismos matemáticos, una de las ventajas de los modelos formales es que se pueden verificar matemáticamente, además de que se pueden verificar las propiedades de los modelos [5]. En esta perspectiva de modelado se encuentran técnicas como Redes de Petri, y programación matemática (generalmente programación lineal) [3] [6] [7].

Sin embargo, en la vida real los planes de procesos comprenden muchos aspectos ilustrados en la tabla 1, y, por lo tanto, es evidente que no existe la técnica de modelado que puede abarcar toda esta variedad de características que constituyen un SI. Por otra parte, los simuladores surgen de la incapacidad de evaluar más de una fracción del inmenso rango de opciones disponibles para la planeación, programación de operaciones y ejecución [8].

PERSPECTIVA	TÉCNICA DE MODELADO	TIPOS DE ANÁLISIS
MÁQUINAS DETERMINISTAS	IDEF	Observación, Monitoreo, Simulación
	RAD, DFD	Observación, Análisis de desempeño.
SISTEMAS DINÁMICOS COMPLEJOS	Redes de Petri	Observación, Validación, Verificación, Análisis de desempeño, Simulación.
	Modelos matemáticos	Análisis de desempeño, Simulación.

Tabla 1. Perspectivas y técnicas de modelado.

En especial, las herramientas existentes pueden clasificarse en diversos enfoques de acuerdo con los siguientes criterios.

1. *Considerando los recursos usados para la formalización*. En este criterio se fijan las características del PLC, como: el lenguaje a utilizar, representación de información adicional, etc.

2. *Basado en la complejidad de las estructuras.* Se enfoca en la complejidad del algoritmo de solución, la arquitectura de la solución, etc.
3. *Elaborando una descripción a través de modelos formales.* Este criterio, se encarga de analizar y determinar cuales métodos deben usarse para generar una representación a través de un modelo formal.
4. *Traducción de modelos formales en programas para PLC's.* Cómo se traducen al PLC, porque cada PLC tiene su lenguaje.

En el mercado existen algunos simuladores de PLC's disponibles en línea tales como: Simulador SimuPLC3.1.0 [10], S7-200 [11], PC-Simu [12], ZenSoftware V3 [13] que pueden simular el proceso mas no automatizar los procesos industriales, sus características principales se muestran en la tabla 2, desafortunadamente los responsables de estos simuladores no proporcionan su técnica de modelado.

NOMBRE	ALGUNAS CARACTERÍSTICAS
SimuPLC3.1.0	Permite la simulación del funcionamiento de las instrucciones básicas (STL).
S7-200	Para CPUs 212, 214, 215, 216, 221, 222, 224, 226
PC-Simu	Permite simular los elementos que contiene una industria como: interruptores, pulsadores, hidroneumáticos, etc.
ZenSoftware V3	Programar todos los modelos V1 y los modelos de 20E/S a través de modelos en diagrama de circuito eléctrico. Además se pueden simular y verificar los programas desarrollados.

Tabla 2. Algunos Simuladores de PLC's.

Hay principalmente dos tipos de herramientas disponibles para la simulación de sistemas: herramientas de simulación visual (por ejemplo Amplie [14], LabView [15], Simulink [16]), y Lenguajes de modelado (por ejemplo, SIMSCRIPT [17]). Desafortunadamente, estas herramientas tienen como inconvenientes, entre otros, que funcionan para ciertos sistemas operativos, y al cambiar estos el software no puede usarse; el lenguaje en el cuál deben especificarse los modelos es complejo; los resultados obtenidos no pueden exportarse para ser analizados en otro software; es imposible agregar nuevas funciones matemáticas. Por lo que la herramienta que se propone en este artículo debe evitar estos inconvenientes, para que tenga una vida útil más larga y que los usuarios del sistema puedan agregar sus propias funciones.

III. ARQUITECTURA PARA SSI

La arquitectura del SSI debe satisfacer los siguientes aspectos.

1. *Estándar de la industria:* La herramienta debe utilizar estándares de la industria, a fin de que los modelos creados por la herramienta tengan una vida útil más larga. Debido a que la técnica de modelado incluye a las Redes de Petri, se debe usar el estándar ISO 15909-2 para el lenguaje de especificación de las Redes de Petri.
2. *El acceso a funciones matemáticas y de biblioteca:* Obviamente, una herramienta de software no puede ser completa, en el sentido que ofrece todas las funciones matemáticas que se necesitan para el modelado y simulación de los sistemas. De este modo, la herramienta debe ser capaz de acceder a bibliotecas adicionales y funciones para hacer complejo análisis matemático.
3. *Lenguaje de programación y el sistema operativo neutral:* De este modo, deberá ser capaz de aceptar y/o crear en un formato estándar de intercambio las entradas y salidas de datos, tales como XML. Además de ser capaz de ejecutarse en cualquier sistema operativo.

La Figura 1 muestra la arquitectura del SSI, note que contiene cuatro tipos de ficheros. El primero,

PUF se utiliza para indicar los diferentes marcos de trabajo que tiene el simulador: modo administrador del sistema, modo ingeniero industrial, y modo científico. Los otros tres tipos de ficheros se utilizan para la simulación: SIF para proporcionar los datos tal y como describen los procesos los ingenieros industriales, MLL en donde se pueden definir funciones lógicas y matemáticas, y los archivos de definición para Redes de Petri (PNF). Cada uno de estos archivos será manipulado por el SSI, que tiene una serie de módulos como el módulo de Redes de Petri para conectar los diferentes subsistemas, módulo temporizador para cada tiempo de transición, etc.

La arquitectura propuesta para el SSI mostrado en la figura 1 permite además la separación de los detalles estáticos y dinámicos del sistema modelado. Los detalles estáticos son fijos y no cambian durante las simulaciones, por lo que estos detalles deben mantenerse separados en los archivos fuente. Mientras que los detalles dinámicos varían, cambiando en cada simulación; por lo que los datos obtenidos en la simulación deben mantenerse en un archivo común (el archivo de simulación correspondiente al sistema).

Por otra parte los diferentes archivos de origen (como el archivo principal de simulación, los archivos de definición de Redes de Petri, y los archivos de definición de funciones) pueden tener

acceso e intercambio de parámetros globales a través de un paquete. Así que el diseño contiene un archivo de información global para intercambio, a fin de conjuntar los valores que pueden ser leídos e intercambiados entre los distintos ficheros del SSI.

Otra característica importante de la arquitectura para el SSI consiste en que se pueden importar y exportar funciones de archivos de Microsoft Excel, así se pueden analizar resultados, o introducir datos al sistema que por lo general los ingenieros industriales han elaborado en Microsoft Excel.

IV. SIMULACIÓN CON SSI

La metodología para realizar simulaciones con SSI se enfoca en dos modos: modo industrial, y modo de Redes de Petri; y en tres pasos: definir el modelo del sistema a simular, definir el conjunto de variables de intercambio, generar los archivos de simulación.

En ambos modos de trabajo (modo industrial y modo de Redes de Petri) se pueden crear modelos, analizar modelos y simular modelos. En SSI cuando se crean o adicionan modelos para simulación se sigue la siguiente metodología.

- *Definir el comportamiento estático del sistema.* Consiste en definir el modelo matemático del sistema a simular. En el caso

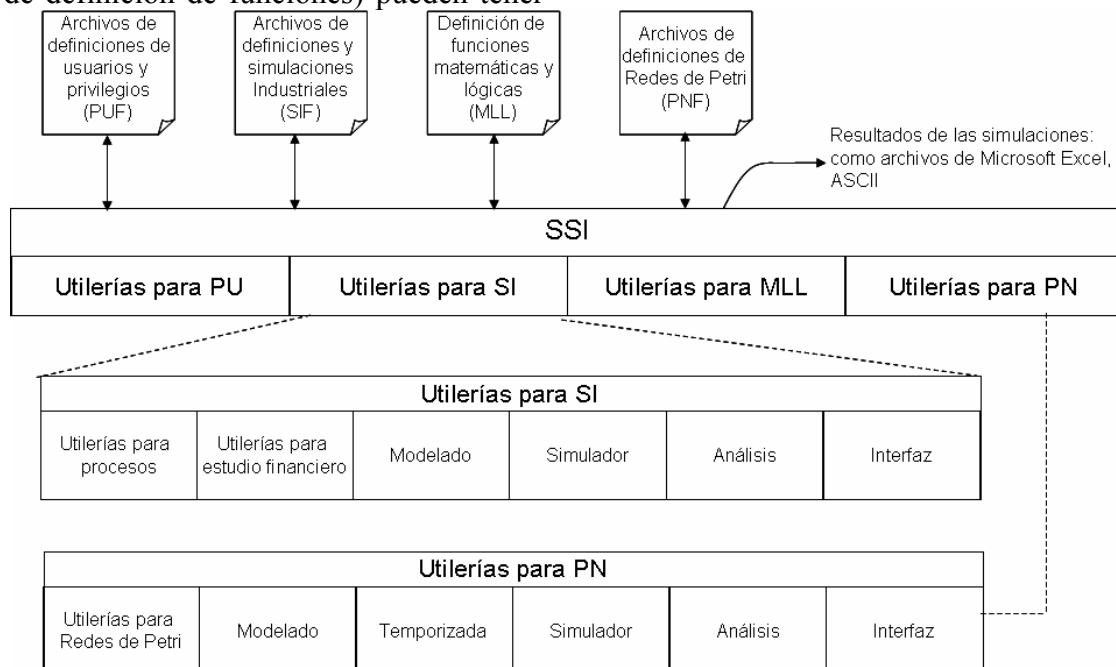


Figura 1. Arquitectura del SSI.

del modo industrial, se parten de especificaciones de secuencias de procesos. Mientras que en el caso del modo en Redes de Petri consiste en modelar cada parte del sistema (con lugares y transiciones).

- *Definir el comportamiento dinámico del sistema.* En esta etapa el simulador se enfoca en capturar las condiciones de funcionamiento del sistema. En ambos casos se trata de modelar las condiciones iniciales (marcado inicial), los tiempos de duración de las actividades (tiempo de retardo de las transiciones), etc.
- *Programación sobre los modelos y ejecución de la simulación.* A través de archivos se deben elegir el tipo de simulación a realizar, la información a intercambiar, y después de realizada la simulación los resultados deben registrarse en un archivo.

V. EJEMPLO DE SIMULACIÓN CON SSI

Para ilustrar cómo realiza las simulaciones el SSI consideramos el caso en el cuál el sistema será modelado a través de la herramienta para posteriormente efectuar algún tipo de análisis, ya que en otro caso la información que se captura en el simulador debería cargarse al sistema en un archivo de Excel.

El modelado del sistema parte de una descripción informal denominada curso-grama, el cuál contiene la información de las secuencias de procesos y las herramientas utilizadas en forma de una descripción funcional y gráfica. Los elementos básicos en un curso-grama son: transporte, maquina, operador, almacén y supervisión; los cuales se describen en la tabla 3.

El modelado en el simulador SSI parte de la definición estática del sistema. En este caso del curso-grama, cuando este va a ser generado dentro del simulador SSI, el usuario debe indicar una descripción de cada elemento (acción ilustrada en la figura 2).

Una vez que cada elemento es descrito, se procede a capturar la información dinámica del sistema, en este caso las secuencias de operación. Para lo cuál se utilizan dos vistas, el cronograma y

el curso-grama, ambas vistas capturan la secuencia de mensajes entre los componentes. En dónde el cronograma nos muestra la representación gráfica de la secuencia de mensajes en forma similar a un diagrama de secuencias (ilustrado en la figura 3). Mientras que el curso-grama representa la información en forma de tabla (ilustrado en la tabla 4).



Figura 2. Ventana de especificación para los elementos del sistema a modelar.

ELEMENTO	FUNCIÓN	REPRESENTACIÓN GRÁFICA
Operador	Se encarga de cargar las piezas	
Transporte	Transportar las piezas a los distintos componentes	
Maquina	Realiza la operación configurada	
Almacén	Guarda el producto terminado o semi-terminado	
Supervisión	Revisa que el producto este correcto	

Tabla 3. Elementos básicos en un curso-grama analítico y su representación en SSI.

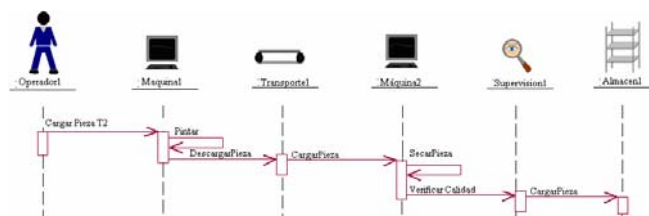


Figura 3. Vista de cronograma.

SECUENCIA DE ACTIVIDADES	RESPONSABLE				
	OPER1	MAQ1	MAQ2	ALM1	SUP1
Cargar Pieza T2	1				
Pintar Pieza		1			
Descargar Pieza		1			

Tabla 4. Vista de curso-grama.

Finalmente, se realiza la programación sobre los modelos y la ejecución de la simulación. En este caso cada uno de los modelos anteriores se selecciona como información para intercambiar con el modelado de especificaciones en redes de Petri, como se ilustra en la figura 4 y se describen sus elementos en la figura 5.

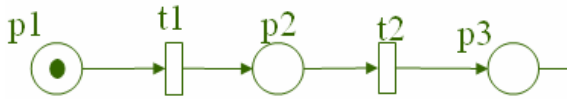


Figura 4. Modelado como red de Petri.

p1	Oper1Ready	t1	CargarPieza t2
p2	Maq1Ready	t2	PintarPieza
p3	TralReady	t3	DescargarPieza

Figura 5. Descripción de algunos elementos de red de Petri de la figura 4.

Una vez elaborado el modelo, se procede a hacer la simulación dependiendo del análisis que se requiera. En este momento nos encontramos generando los algoritmos de simulación.

VI. CONCLUSIONES Y TRABAJO FUTURO

En este artículo se propone una arquitectura general para el diseño y desarrollo de una nueva herramienta para el modelado y simulación de sistemas industriales (SSI) basada en Redes de Petri. La arquitectura está diseñada para alcanzar 1) flexibilidad, debido a que se basa en estándares de la industria, 2) extensibilidad, ya que puede ser ampliado por las funciones programadas por el usuario, y pueden tener acceso a funciones matemáticas y las bibliotecas, y 3) lenguaje de programación y sistemas operativos neutral, como los modelos se pueden definir en XML, y Microsoft Excel.

Debido a que el diseño y operación de sistemas industriales es un ámbito de aplicación amplio y complejo. Entre las ventajas del uso del simulador se encuentra que puede generar una representación a través de un modelo formal basado en Redes de Petri. Los modelos en Redes de Petri a través de diversos algoritmos ayudarán a la reducción de errores introducidos en un sistema, además de proporcionar técnicas para el modelado, análisis, optimización, control, simulación entre otras tareas.

Como trabajo futuro se está trabajando en extensiones de la herramienta sobre los tipos de análisis y simulaciones, entre las que se encuentran: monitoreo de sistemas, control, cadenas logísticas y diagnóstico.

VII. REFERENCIAS

- [1] G. Doumeingts, B. Vallespir, D. Chen, "Methodologies for designing CIM Systems: A survey", Elsevier Science Computers in Industry, pp. 263-280, 1995.
- [2] G. Kateel, M. Kamath, D. Pratt, "An overview of CIM enterprise modeling methodologies", Proceeding of Winter Simulation Conference, pp. 1000-1007, 1996.
- [3] J. Billington, S. Christensen, K. Hee, E. Kindler, O. Kummer, L. Petrucci, R. Post, C. Stehno, and M. Weber, "The Petri Net Markup Language: Concepts, Technology, and Tools", Proceedings of the 24 th International Conference on Application and Theory of Petri Nets, Springer LNCS 2679, pp. 483-505, 2003.
- [4] K. Yeong-Dae, L. Geun-Cheol, L. Seung-Kil, "Tool requirements planning in a flexible manufacturing system with an automatic tool transporter", IEEE Transactions on Robotics and Automation, Vol. 17, pp. 795-804, 2001.
- [5] M. Koubarakis, and D. Plexousakis, "A formal framework for business process modelling and design", Inf. Systems, Vol. 27, pp. 299-319, 2002.

- [6] M. Weber, and E. Kindler, “*The Petri Net Markup Language. Petri Net Technologies for Modeling Communication Based Systems*”, Springer LNCS 2472, pp. 124-144, 2003.
- [7] S. Escudero, J. A. Rojas-Ramírez, “*Reprogramación de operaciones en procesos de producción: Complejidad y alternativas de resolución*”, Memorias del 5º. Congreso Nacional de Ingeniería Electromecánica y de Sistemas, pp. 974-982, 2000.
- [8] S. Kanai, T. Kishinami, and T. Tomura, “*Object-oriented Graphical Specification and Seamless Design Procedure for Manufacturing Cell Control Software Development*”, IEEE Proceedings International Conference on Robotics & Automation, pp. 401-407, 2000.
- [9] W.X. Xun, W. Lihui, R. Yiming, “*STEP_NC and function blocks for interoperable manufacturing*”, IEEE Transactions on Automation Science and Engineering, Vol. 3, pages 297-308, 2006.
- [10] *Autómatas Programables: Programas de simulación SimuPLC3.1.0*, en <http://www.andrebarbosa.eti.br/SimuPLC%20Eng/Help%20HTML/index.html>
- [11] *Autómatas Programables: Programas de simulación S7-200*, en <http://personales.ya.com/canalPLC/interest.htm>
- [12] *Autómatas Programables: Programas de simulación PC-Simu*, en <http://personales.ya.com/canalPLC/favorite.htm>
- [13] *Autómatas Programables: Programas de simulación ZenSoftware V3*, en <https://www.software.omron.es/software/index.html>
- [14] *Extended* (2008), en <http://www.imaginetthatinc.com/>
- [15] *LabView 8.5* (2007), en <http://www.ni.com>
- [16] *MATLAB* (2008a), en <http://www.mathworks.com>
- [17] *SIMSCRIPTII* (2006), en <http://www.simsript.com/>

DIAGNÓSTICO DE FALTAS DE CONTROL EN RPI

Hernández Flores Elías, Lizette Rubio Gómez.

Centro de Investigación y Estudios Avanzados.
Unidad Guadalajara,
Av. Científica 1145, Col. El Bajío, 45010 Zapopan, Jal. México.
{hernande,lrubio}@gdl.cinvestav.mx

RESUMEN

Se propone un método basado en la relación de dependencia lineal entre las transiciones de una RPI (con el comportamiento de una máquina de estados), para determinar la propiedad de diagnosticabilidad de un SED. El tipo de faltas que se tratan son las faltas de control. Se dan condiciones suficientes para que un SED sea diagnosticable cuando ocurren este tipo de faltas.

Palabras clave: Diagnóstico, Redes de Petri Interpretadas, Sistemas de eventos discretos.

I. INTRODUCCIÓN

El hombre está diseñando sistemas cada vez más grandes y más complejos, teniendo como consecuencia una mayor dificultad para administrarlos, y por consiguiente de verificar su correcto funcionamiento. El diagnóstico de faltas es un tema de investigación importante ya que ayuda a que los sistemas funcionen adecuadamente. Evitando situaciones riesgosas, que podrían tener como consecuencia pérdidas humanas y económicas. De aquí la importancia de un sistema de diagnóstico de faltas eficiente.

Un sistema de diagnóstico proporciona información sobre algunos estados del sistema. Con dicha información se pueden tomar decisiones sobre qué hacer en situaciones riesgosas, ya sea llevar el sistema a un estado seguro o iniciar el sistema de recuperación de faltas, evitando así una posible catástrofe. Este tema ha sido abordado desde diferentes enfoques y con diferentes herramientas. En este artículo se trata el diagnóstico basado en modelos de Redes de Petri Interpretadas (RPI). Algunos de los trabajos

realizados en esta área de investigación se mencionan a continuación. En [3] se define formalmente la diagnosticabilidad de un sistema de eventos discretos (SED) en términos de su lenguaje de salida representado por un autómata finito. Se dan condiciones necesarias y suficientes para que un sistema sea diagnosticable en base a la existencia de ciclos F_i -Indeterminados en el modelo. Si existen ciclos F_i -Indeterminados en el modelo entonces éste no es diagnosticable. En [1] se trata el problema de diagnóstico en línea basado en modelos de RPI. El modelo incluye tanto el funcionamiento normal como el de falta. Se tratan tres tipos de faltas: *faltas permanentes*, *faltas de control*, y *faltas intermitentes*. Además se propone una caracterización de la propiedad de diagnosticabilidad y un algoritmo polinomial para verificar la propiedad, así como un esquema de diagnóstico en línea.

En este trabajo se propone una caracterización diferente de RPI diagnosticables cuyo funcionamiento anormal contiene *faltas de control*. Ésta se basa en la dependencia lineal de algunas de las columnas de la matriz de incidencia C , lo que tiene como ventaja es que este método se puede usar en un rango más amplio de sistemas como son los sistemas continuos y los sistemas híbridos [4]. El resto de este artículo se organiza de la siguiente manera. En la sección II se hace una breve revisión de los conceptos básicos sobre Redes de Petri Interpretadas. En la sección III se define diagnosticabilidad en términos de Redes de Petri y se propone una caracterización de RPI diagnosticables.

II. REDES DE PETRI INTERPRETADAS

A continuación se dan algunos conceptos básicos sobre Redes de Petri (RP) y diagnóstico de faltas.

Definición 1. Una estructura G de una RP es un grafo bipartito representado por la 4-tupla $G = (P, T, I, O)$ donde:

- $P = \{p_1, p_2, \dots, p_n\}$ es un conjunto finito de vértices llamados lugares.
- $T = \{t_1, t_2, \dots, t_m\}$ es un conjunto finito de vértices llamados transiciones.
- $I: P \times T \rightarrow \mathbb{Z}^+$ es una función que representa el peso de los arcos que van de los lugares a las transiciones.
- $O: T \times P \rightarrow \mathbb{Z}^+$ es una función que representa el peso de los arcos que van de las transiciones a los lugares.

Donde \mathbb{Z}^+ es el conjunto de enteros no negativos. Gráficamente los lugares son representados por círculos, las transiciones por rectángulos y los arcos por flechas. El símbolo $\bullet t_j$ denota el conjunto de lugares p_i tales que $I(p_i, t_j) \neq 0$ y $t_j \bullet$ denota el conjunto de lugares p_i tal que $O(p_i, t_j) \neq 0$, de forma similar $\bullet p_i$ denota el conjunto de transiciones t_j tales que $O(p_i, t_j) \neq 0$, y $p_i \bullet$ el conjunto de transiciones t_j tales que $I(p_i, t_j) \neq 0$.

La matriz de pre-incidencia de G es $C^- = [c_{ij}^-]$, donde $[c_{ij}^-] = I(p_i, t_j)$; la matriz de incidencia posterior de G es $C^+ = [c_{ij}^+]$, donde $[c_{ij}^+] = O(p_i, t_j)$; la matriz de incidencia de G es $C = C^+ - C^-$.

Definición 2. La función de marcado $M: P \rightarrow \mathbb{Z}^+$ representa el número de marcas (dibujadas como puntos) en cada lugar de G .

El marcado generalmente se representa como un vector de n entradas, donde n es el número de lugares.

Definición 3. Una Red de Petri se compone del par (G, M_0) , donde G es la estructura de la RP y M_0 es la distribución de marcas inicial.

Ejemplo 1. En la figura 1 se muestra una RP en la que $P = \{p_1, p_2, p_3, p_4, p_5\}$, $T = \{t_1, t_2, t_3, t_4, t_5\}$, y las funciones $I: P \times T \rightarrow \mathbb{Z}^+$, $O: P \times T \rightarrow \mathbb{Z}^+$

están representadas por la matriz de incidencia $C = C^+ - C^-$ de la ecuación (1).

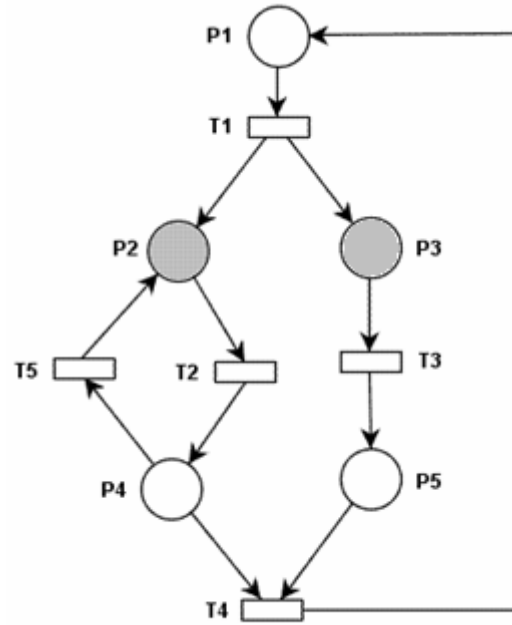


Figura 1. Red de Petri.

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (1)$$

$$C = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$

El marcado inicial está dado por $M_0 = [1 \ 0 \ 0 \ 0 \ 0]^T$.

Una transición $t_j \in T$ está habilitada en un marcado M si y sólo si $\forall p_i \in \bullet t_j: M(p_i) \geq C^-(p_i, t_j)$.

Una transición $t_j \in T$ habilitada en un marcado M_k , puede ser disparada alcanzando un nuevo marcado M_{k+1} , el cual puede ser calculado con la ecuación de estado de una RP que se muestra a continuación.

$$M_{k+1} = M_k + C \cdot \vec{v}_j$$

Donde \vec{v}_j es un vector de m entradas, cuyas componentes se definen como: $\vec{v}_j(i) = 0$, para $i \neq j$, $\vec{v}_j(j) = 1$, y m es el número de transiciones.

Lo anterior también se denota por $M_k \xrightarrow{t_k} M_{k+1}$.

Definición 4. Una *secuencia de disparo* de (G, M_0) es una secuencia de transiciones $\sigma = t_1 t_2 \dots t_k \dots$ tal que $M_0 \xrightarrow{t_1} M_1 \xrightarrow{t_2} \dots M_w \xrightarrow{t_w} \dots$.

Definición 5. Sea $\sigma = t_1 t_2 t_3 \dots$ una secuencia de disparo. El *vector de Parikh* $\vec{\sigma}: T \rightarrow (\mathbb{Z}^+)^m$ donde $\vec{\sigma}$ mapea a cada transición $t \in T$ en el número de ocurrencias de t en σ .

Definición 6. Un sistema de RP (G, M_0) es una *máquina de estados* (ME) si $\forall t_i \in T, | \bullet t_i | = 1 = | t_i \bullet |$.

Definición 7. Sea C la matriz de incidencia de (G, M_0) . Un T-semiflujo X_i de (G, M_0) es una solución de valores racionales semi-positivos de la ecuación $CX_i = 0$.

Definición 8. Sea X_i un T-semiflujo. P_i es un vector de tamaño n tal que $\forall t_k \in X_i, p_j \in \bullet t_k, P_i(p_j) = 1$.

Definición 9. Una RPI [2] es una 4-tupla $Q = (N, \Sigma, \lambda, \varphi)$ donde:

- $N = (G, M_0)$ es una RP como se definió antes.
- $\Sigma = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ es el alfabeto de entrada de la RPI.
- $\lambda: T \rightarrow \Sigma U\{\epsilon\}$ es la función de etiquetado de transiciones, con la siguiente restricción, $\forall t_j, t_k \in T, j \neq k$, si $\forall p_i I(p_i, t_j) = I(p_i, t_k) \neq 0$, $\lambda(t_j) \neq \epsilon$, $\lambda(t_k) \neq \epsilon$ entonces $\lambda(t_j) \neq \lambda(t_k)$. En este caso ϵ representa un evento interno del sistema.

- $\varphi: R(Q, M_0) \rightarrow (\mathbb{Z}^+)^q$ es la función de salida, que asocia a cada marcado en $R(Q, M_0) \rightarrow (\mathbb{Z}^+)^q$ un vector de salida. Donde q es el número de salidas (sensores disponibles) y $R(Q, M_0)$ es el conjunto de marcados alcanzables desde el marcado inicial disparando solo transiciones habilitadas.

Una transición habilitada t_j , etiquetada con $\lambda(t_k) \neq \epsilon$ (transición manipulable) debe dispararse cuando $\lambda(t_k)$ es activado. Una transición

habilitada t_j , etiquetada con $\lambda(t_k) = \epsilon$ (transición no manipulable) puede dispararse si está habilitada.

Este trabajo trata sólo con casos donde φ es lineal; por lo que se puede representar como una matriz de tamaño $q \times n$. Cada columna de esta matriz es un vector elemental o nulo. Si el símbolo de salida i está presente siempre que $M(p_j) \geq 1$, entonces $\varphi(i, j) = 1$, de lo contrario $\varphi(i, j) = 0$.

Ejemplo 2. Retomando el ejemplo 1 con la matriz φ mostrada en la ecuación (2) y $\forall t \in T, \lambda(t) = \epsilon$. Vemos que todas las columnas de la matriz φC de la ecuación (3) son diferentes del vector nulo, por lo que todas las transiciones de esta RPI son medibles. Además puesto que para todas las transiciones $\lambda(t) = \epsilon$ el sistema es autónomo.

$$\varphi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (2)$$

$$\varphi C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 1 & 0 \\ 1 & -1 & -1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$

$$\varphi C = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & -2 & -1 \end{bmatrix} \quad (3)$$

III. DIAGNOSTICABILIDAD

En esta sección se dan condiciones estructurales suficientes, basadas en la dependencia lineal de las columnas de C para que una RPI, que es una ME, sea diagnosticable con respecto a las faltas de control.

Definición 10. Falta de control. Una falta de control se representa por el disparo de una transición t_i que existe en el comportamiento normal del sistema, pero que debería ser evitado por el sistema de control, se considera que el disparo de t_i alcanza estados dentro del funcionamiento normal del sistema.

Definición 11. Se denota por T^f al conjunto de transiciones de falta de control.

Definición 12. (Q^N, M_0^N) define la subred que no considera las transiciones en T^f .

Definición 13. Un lugar p_i está en *conflicto estructural* si $|p_i \bullet| > 1$.

Definición 14. P^c es el conjunto de lugares en conflicto estructural en (Q^N, M_0^N) .

Definición 15. Se denota por T^c al conjunto de transiciones que pertenecen a P^c . Exceptuando las faltas de control.

Definición 16. Una transición es *evento-detectable* si y sólo si su disparo puede ser detectado a partir de la salida y la entrada del modelo de RPI. En [5] se propone una caracterización para este tipo de transiciones.

En la definición 17 se describen los pasos para construir la matriz de incidencia C' a partir de C . C' es una matriz C con nuevos lugares. La adición de los nuevos lugares toma en cuenta la matriz φC por lo que se puede deducir fácilmente de la construcción de C' que si una columna de C es linealmente dependiente de otras, y estas transiciones producen la misma observación, entonces estas columnas seguirán siendo linealmente dependientes en C' .

$$C' = \begin{bmatrix} C \\ p_{n+1} \\ \vdots \end{bmatrix}$$

Definición 17. Matriz de incidencia C' . Es una copia de la matriz C tal que

- Para todo par de transiciones t_i, t_j donde $\varphi C(\bullet, t_i) = \varphi C(\bullet, t_j) \neq 0$ se hace $\sigma(p_i, t_i) = 1, \sigma(p_i, t_j) = 1$ donde p_i es un nuevo lugar agregado a la RPI. (t_i puede ser igual a t_j).

Definición 18. Se denota por P^a al conjunto de lugares que se agregan a (Q, M_0) según la definición 17.

El ejemplo 3 muestra una RPI que se ha modificado según lo especificado en la definición 17.

Ejemplo 3. Considere la RPI que se muestra en la figura 2, cuya matriz de φ y matriz φC se muestran en las ecuaciones (4) y (5) respectivamente. Donde $T^f = \{t_{12}, t_{13}\}$.

$$\varphi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$\varphi C = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix} \quad (5)$$

Aplicando la definición 17 obtenemos la RP que se muestra en la figura 3.

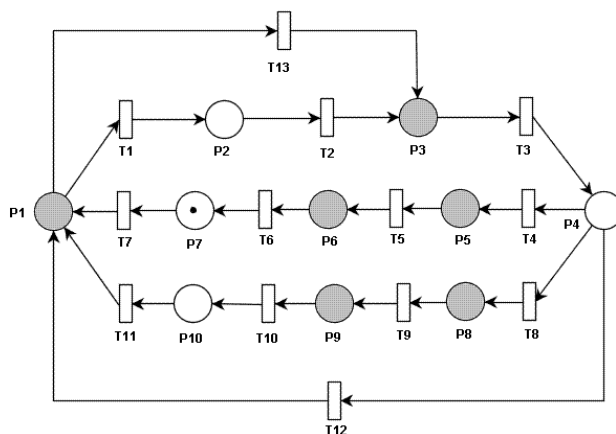


Figura 2. Red de Petri Interpretada.

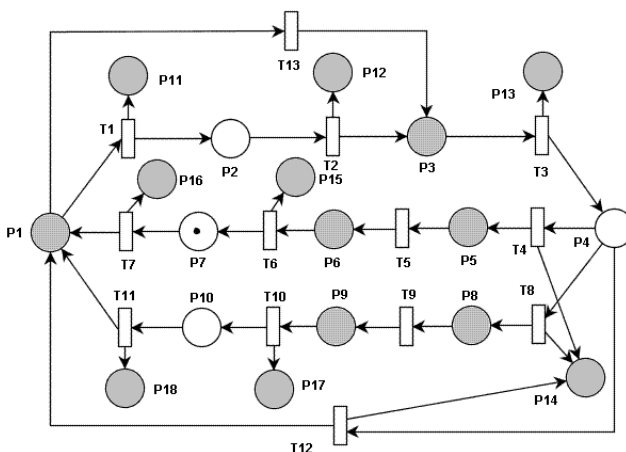


Figura 3. Transformación según definición 17. (C')

La ecuación 6 muestra la matriz C' que se obtiene al aplicar la transformación de la definición 17.

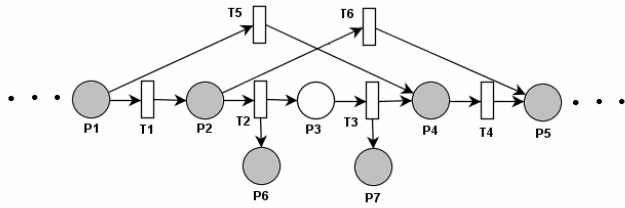


Figura 4. Partición de faltas.



(6)

A continuación se define diagnosticabilidad en términos de RPI.

Definición 19. Sea (Q, M_0) una RPI. (Q, M_0) es diagnosticable si en $k < \infty$ eventos después de la ocurrencia de una falta de control, es posible determinar que ésta ocurrió.

En este trabajo se considera que la partición Π_i sobre T^f [6] contiene a t_i, t_j en la misma partición cuando se presenta el caso de la figura 4. Es decir, si dos transiciones producen el mismo efecto en las observaciones entonces éstas pertenecen a la misma partición. Considere por ejemplo t_2, t_6 como faltas, estas producen la misma observación, por lo que pertenecen a la misma partición.

El siguiente teorema establece que dada una RPI (Q, M_0) que es una ME, con la matriz de incidencia C :

- Si la columna que representa la transición de falta no es una combinación lineal de otras columnas de la matriz C' .
- No existe en C' algún T-semiflujo.
- Toda transición en conflicto estructural es evento-detectable.

Entonces (Q, M_0) es diagnosticable.

Teorema 1. Sea una RPI (Q, M_0) que es una ME viva y segura que contiene faltas de control. Si

1. $\forall t_f \in T^f$ no existe un conjunto de columnas que sea combinación lineal de $C'(\bullet, t_f)$.
2. $\nexists X_i$ tal que $C'X_i = 0$.
3. $\forall t_i \in T^d$, t_i es evento-detectable.

Entonces (Q, M_0) es diagnosticable.

Demostración: Puesto que (Q^N, M_0^N) es viva entonces existe una secuencia de disparo σ tal que $M_0 \xrightarrow{\sigma} M_k$ donde M_k es un marcado donde t_f está habilitada y por lo tanto puede ocurrir. En el marcado M_k existen al menos dos secuencias de transiciones σ_f, σ_r tal que $M_k \xrightarrow{\sigma_f} M_s$ y $M_k \xrightarrow{\sigma_r} M_s$ donde M_s es un marcado en el cual $M(t_f, \bullet) = 1$ y $\sigma_f = t_f$ por lo que $C(\bullet, t_f)$ y $C(\bullet, \sigma_r)$ son linealmente dependientes. Ahora se demostrará que considerando la matriz C' , si no existe una secuencia de transiciones σ_r tal que $C'(\bullet, t_f)$ y $C'(\bullet, \sigma_r)$ son linealmente dependientes entonces t_f es diagnosticable. En el marcado M_k se pueden presentar tres casos.

Si sólo existe una transición $t_j \in \sigma_r$ tal que $\varphi C(\bullet, t_j) \neq 0$ y $\varphi C(\bullet, t_j) - \varphi C(\bullet, \sigma_f)$ entonces no es posible distinguir entre el disparo de σ_f y σ_r . En este caso la matriz C' cumple con $O(p_i, t_j) = 1$ y $O(p_i, t_f) = 1$ donde $p_i \in P^a$, por lo que $C'(\bullet, \sigma_f)$ y $C'(\bullet, \sigma_r)$ seguirán siendo linealmente dependientes.

Cuando para toda transición $t_j \in \sigma_r$ se cumple que $\varphi C(\bullet, t_j) = 0$. En este caso también $\varphi C(\bullet, \sigma_f) = 0$ por lo que no se puede distinguir el disparo de σ_f y σ_r . Puesto que $\forall t_j \in \sigma_r$, $\varphi C(\bullet, t_j) = 0$ entonces no se cumple que $O(p_i, t_j) = 1$ para toda t_j igualmente para t_f por lo que $C'(\bullet, \sigma_f)$ y $C'(\bullet, \sigma_r)$ seguirán siendo linealmente dependientes.

Cuando existe más de una transición $t_j \in \sigma_r$ tal que $\varphi C(\bullet, t_j) \neq 0$ entonces es posible distinguir entre el disparo de σ_f y σ_r . En este caso $\forall t_j \in \sigma_r$ y $\varphi C(\bullet, t_j) \neq 0$ existe $O(p_i, t_j) = 1$ donde $p_i \in P^a$ por lo que $C'(\bullet, \sigma_f)$ y $C'(\bullet, \sigma_r)$ son linealmente independientes.

Considerando la condición dos. Debido a que no existen T-semiflujos en C' , entonces por la forma en que se construyó C' a partir de C , sabemos que la matriz C no contiene T-semiflujos no observables. Por lo que no existen secuencias de eventos no observables de longitud infinita. En particular no existe un secuencia de disparos no observables infinita a partir de M_2 , por lo que en un número finito de pasos se deberá obtener una observación.

Tomando en cuenta la condición tres. Para cada situación de elección que se presente en (Q^N, M_0^N) se sabe cuál fue la transición disparada dado que $\forall t_i \in T^o$, t_i es evento-detectable. Por lo que, para una elección que lleva a un estado que habilite una falta t_{f_i} , t_{f_i} se distingue de otra falta t_{f_j} a la cual se llega disparando otra transición de elección. □

Enseguida se propone un algoritmo para verificar la propiedad de diagnosticabilidad de una RPI que es una ME.

Algoritmo 1. Diagnosticabilidad.

Entrada: C' que se obtiene de (Q, M_0) según definición 17.

Salida: Diagnosticable.

Paso 1. Resolver el PPL siguiente.

$$PPL_1 = \begin{cases} \min & \bar{\sigma} \\ s.a. & C' \bar{\sigma} = C' \bar{\sigma} \\ & \bar{\sigma} > 0 \\ & \bar{\sigma}(t_f) = 0 \end{cases}$$

Si PPL_1 tiene solución entonces $C'(\cdot, t_f)$ es una combinación lineal de $C' \bar{\sigma}$ por lo que éstos se confunden y no es posible diagnosticar el disparo de t_f , de lo contrario continuar.

Paso 2. Resolver el PPL siguiente.

$$PPL_2 = \begin{cases} \min & X_r \\ s.a. & C' X_r = 0 \end{cases}$$

Si PPL_2 no tiene solución entonces no existe una secuencia de disparos que llevan de una falta a un T-semiflujo no observable.

Paso 3. Verificar que para toda $\forall t_i \in T^o$, t_i es evento-detectable [5].

Si el PPL_1 no tiene solución para toda transición de falta y si PPL_2 no tiene solución entonces (Q, M_0) es diagnosticable.

Ejemplo 4. Continuando con el ejemplo 3. Se considera la secuencia $\bar{\sigma}_1 = t_1 t_2$ donde se puede ver que $C' \bar{\sigma}_1 = C' t_{12}$. De acuerdo al PPL_1 se tiene que $C' \bar{\sigma}_1 \neq C' t_{12}$, por lo que el paso uno del algoritmo no tiene solución para t_{12} . En el caso de t_{12} se consideran las secuencias $\bar{\sigma}_2 = t_4 t_5 t_6 t_7$ y $\bar{\sigma}_3 = t_8 t_9 t_{10} t_{11}$ se tiene que $C' \bar{\sigma}_2 \neq C' t_{12}$ y $C' \bar{\sigma}_3 \neq C' t_{12}$ y en consecuencia tampoco tiene solución para t_{12} . En el paso dos, se ve que no existen T-semiflujos en C' . Lo que nos indica que no existen T-semiflujos no observables en C .

Dado que ningún PPL tiene solución y las transiciones de elección son evento-detectable se tiene que el sistema es diagnosticable.

IV. CONCLUSIONES

Este trabajo propone una caracterización y un método para verificar la propiedad de diagnosticabilidad en RPI, cuyo comportamiento es el de las máquinas de estado. Esta caracterización tiene la ventaja de que se puede extender a sistemas híbridos (lineal y discretos). El algoritmo propuesto aquí, tiene una complejidad polinomial. Algunos aspectos pendientes sobre este trabajo son: extender el resultado a una mayor clase de RPI y proponer un esquema diagnosticador.

V. REFERENCIAS

[1] Elvia Ruiz-Beltrán, "Esquemas de diagnostico de faltas en sistemas de eventos discretos", Cinvestav GDL, 2007.

[2] M. E. Meda, A. Ramírez-Treviño and A. Malo. "Identification in discrete event systems". Proceedings of the IEEE Conference on Systems, Man & Cybernetics, pp. 740- 745, 1998.

[3] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis.

- “Diagnosability of discrete-event systems”*.
IEEE Trans. Autom. Control, vol. 40, no 9,
pp. 1555-1575, Sep. 1995.
- [4] M. Massoumnia, G. C. Verghese, and A. S. Willsky. *“Failure Detection and Identification”*. IEEE Trans. Autom. Control, vol. 34, no 3, pp. 316-321, Mar. 1989.
- [5] I. Rivera-Rangel, A. Ramirez-Treviño, L.I. Aguirre-Salas and J. Ruiz-León. *“Geometrical characterization of Observability in Interpreted Petri Nets”*. Kybernetika, vol. 41, 553-574, 2005.
- [6] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis. *“Diagnosability of discrete event systems.”* IEEE Transactions on Automatic and Control, vol 4, no. 9, pp. 1555-1575, 1995.

CÁLCULO DISTRIBUIDO DEL GRAFO DE ALCANZABILIDAD DE UNA RED DE PETRI

M. Alcaraz-Mejía, R. Campos-Rodríguez, C. Estrada-Gutiérrez

Departamento de Ciencias Tecnológicas, División de Desarrollo Bio-Tecnológico
 Centro Universitario de la Ciénega, Universidad de Guadalajara
 Av. Universidad 1115, Col. Linda Vista, C.P. 47810, Ocotlán, Jalisco, México
 Tel. +52 (392) 9259400, correo electrónico: {mildreth.alcaraz, raul.campos}@cuci.udg.mx,

RESUMEN

En este artículo se presenta un algoritmo distribuido para el cálculo del grafo de alcanzabilidad de una Red de Petri. Se utiliza la plataforma de invocación de métodos remotos de Java con un enfoque Cliente/Servidor. Bajo este esquema, se crea un conjunto de Servidores que realizan expansiones parciales sobre el grafo de alcanzabilidad. El Cliente se encarga de asignar las tareas a los servidores utilizando un método FIFO. Con los resultados parciales calculados por los Servidores, el Cliente construye la solución global.

Palabras clave: Redes de Petri, Grafo de Alcanzabilidad, RMI, Java, Algoritmo Distribuido.

I. INTRODUCCIÓN

El análisis de propiedades de un Sistema de Eventos Discreto (SED) modelado con una Red de Petri (RP) a menudo requiere de la obtención de su grafo de alcanzabilidad. Se conoce que la construcción de dicho grafo es un problema NP, lo que se complica aún más conforme la red se vuelve más compleja [1][3].

Una forma eficiente de abordar este problema consiste en hacer una expansión en amplitud por nodo [2]. Para aplicar esta técnica de expansión en amplitud, se puede utilizar un esquema Cliente/Servidor para realizar cálculos distribuidos y aprovechar el paralelismo proporcionado por este esquema [4].

Una de las plataformas más conocidas para crear aplicaciones distribuidas es la tecnología RMI de Java (Java Remote Method Invocation) [5]. Esta técnica de Invocación de Métodos Remotos permite crear esquemas Cliente/Servidor con

facilidad. En la Figura 1 se muestra un modelo conceptual del esquema Cliente/Servidor con RMI de Java.

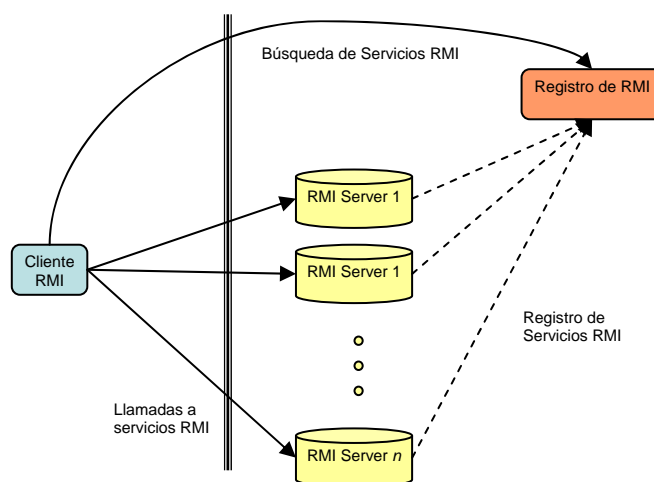


Figura 1. Esquema Cliente/Servidor con RMI de Java.

El algoritmo distribuido para el cálculo del grafo de alcanzabilidad presentado en este artículo, utiliza la tecnología RMI de Java. La Figura muestra el escenario de aplicación. Existen n servidores que se encargan de realizar una expansión parcial por nodo del grafo de alcanzabilidad. Un cliente distribuye las tareas a los servidores y de reunir los resultados parciales para formar la solución global.

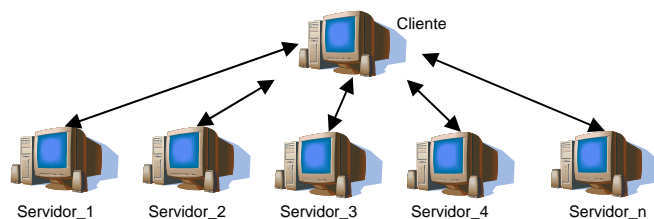


Figura 2. Cliente – n Servidores

El resto de este documento se organiza de la siguiente manera: la sección **¡Error! No se encuentra el origen de la referencia.** hace una revisión de las Redes de Petri con énfasis en el Grafo de Alcanzabilidad. La sección **¡Error! No se encuentra el origen de la referencia.** explica el algoritmo para el cálculo distribuido del grafo de alcanzabilidad, así como su implementación con RMI de Java. Finalmente, la sección **¡Error! No se encuentra el origen de la referencia.** presenta las conclusiones de este trabajo, así como trabajo futuro.

II. GRAFO DE ALCANZABILIDAD DE UNA RED DE PETRI

Las Redes de Petri son una metodología conocida para el modelado, análisis y simulación de SED's. Formalmente, una RP se define de la manera siguiente.

Definición 1. Una Red de Petri (RP) es un grafo bipartito (G, M_0) tal que M_0 es la distribución inicial de marcas, y $G = \{P, T, I, O\}$ [6]:

- $P = \{p_1, p_2, \dots, p_n\}$ y $T = \{t_1, t_2, \dots, t_m\}$ son conjuntos finitos de nodos llamados lugares y transiciones, respectivamente.
- $I(O): P \times T \rightarrow \mathbb{N} \cup \{0\}$ es una función de entradas (salidas) que representa el peso de los arcos que van de los lugares a las transiciones (de las transiciones a los lugares), donde \mathbb{N} es el conjunto de los naturales. Estas funciones, $I(O)$, se representan mediante una matriz, conocida como matriz de entradas (salidas).

En una RP, una transición t_j está habilitada en un marcado M_k si $\forall p \in P, M_k(p_i) \geq I(p_i, t_j)$. Una transición habilitada t_j puede ser disparada alcanzando un nuevo marcado M_{k+1} el cual puede ser calculado utilizando la siguiente Ecuación de Estados:

$$M_j = M_i + C \cdot \bar{v}, \quad (1)$$

donde $v(j) = 1$, y $v(i) = 0, \forall i \neq j$.

El análisis de propiedades en una RP, como vivacidad, ausencia de bloqueos, acotamiento, entre otras, es una etapa común en el proceso de

diseño de SED's. El problema de comprobar algunas de estas propiedades a menudo requiere verificar la alcanzabilidad de ciertos estados [1], [6]. La definición formal del conjunto de alcanzabilidad para un RP se presenta a continuación.

Definición 2. Sea (G, M_0) una RP. El conjunto de alcanzabilidad $R(G, M_0) = \{M_i | M_i \text{ puede ser alcanzado desde } M_0 \text{ a través del disparo de una secuencia disparable } \sigma \text{ utilizando la Ecuación 1}\}$.

El conjunto de alcanzabilidad se representa por un grafo dirigido definido de la siguiente manera.

Definición 3. Sea (G, M_0) una RP. Sea $R(G, M_0)$ el conjunto de alcanzabilidad de (G, M_0) . El grafo de alcanzabilidad es un grafo tal que el nodo inicial está formado por el estado M_0 , y para todo par de nodos conectados $M_i \xrightarrow{t_a} M_j$ se cumple que $M_j = M_i + C \cdot \bar{t}_a$ para la transición t_a habilitada.

El algoritmo para el cálculo del grafo de alcanzabilidad se puede implementar como un algoritmo de búsqueda en amplitud [8], donde el nodo inicial está formado por el marcado inicial, mientras que la búsqueda en amplitud genera todos los marcados alcanzados desde el marcado inicial. Aplicando este procedimiento de manera recursiva a los nuevos nodos, se obtiene el grafo de alcanzabilidad. A continuación se presenta en pseudocódigo el algoritmo de construcción del Grafo de Alcanzabilidad por expansión en amplitud.

Algoritmo 1. Grafo_de_Alcanzabilidad;

Entradas: I : Matriz de Entradas; O : Matriz de Salidas, M : Marcado Inicial.

1. Inicializar la Cola con el estado inicial M ;
2. Hacer {
 - a. Sacar al primer marcado en la Cola M_k ;
 - b. Para cada transición t_i que está habilitada en el marcado M_k entonces {
 - i. Calcular su estado siguiente M_j con la ecuación de estados (1);
 - ii. Hacer $M_k - t_i > M_j$;
 - iii. Si M_j es nuevo marcado

entonces Agregar M_j a la Cola;
} Mientras la cola no esté vacía;

3. Fin;

En la Figura 3 se muestra una RP para el problema de 5-filósofos comensales, un problema clásico en la teoría de la computación propuesto por Edsger Dijkstra para representar el problema de la sincronización de procesos en un sistema operativo [7]. En la Figura 4 se muestra el grafo de alcanzabilidad para la RP de los 5-filósofos comensales presentada en la Figura 3.

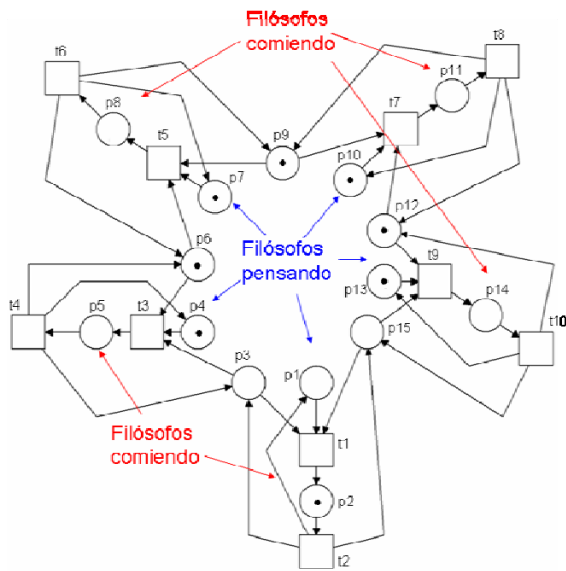


Figura 3. RP para el problema de 5-filósofos comensales

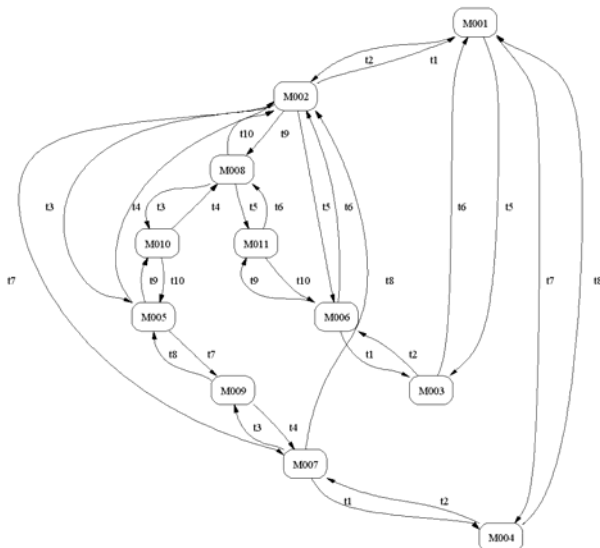


Figura 4. Grafo de Alcanzabilidad

Al iterar el Algoritmo 1 se obtienen las transiciones habilitadas para el marcado inicial M0001 en la RP de la Figura 3, estas son t2, t5 y t7, obteniendo los marcados M002, M003, M004, respectivamente. Puesto que ninguno de estos marcados ha sido generado previamente, entonces se meten a la Cola={M002, M003, M004}, En la segunda iteración se saca de la Cola al primer marcado M002, y se generan las transiciones habilitadas para este marcado, que resultan ser t1, t3, t5, t7 y t9, obteniendo los marcados M001, M005, M006, M007, M008. Puesto que el marcado M001 ya ha sido generado previamente, entonces no se agrega a la Cola, quedando esta como Cola={M003, M004, M005, M006, M007, M008}. En la siguiente iteración se saca de la Cola al primer marcado M003, y se obtienen las transiciones habilitadas t2 y t6, generando los marcados siguientes M006 y M001. Puesto que ambos marcados ya han sido generados previamente, la Cola no aumenta sus estados. Este procedimiento se repite para cada elemento en la Cola. El Grafo de Alcanzabilidad en forma de lista se muestra en la Figura 5.

```

M001: 0(01010101101101100)---t2--> M002: 1(10110110110110110)
      ---t5--> M003: 1(0101000010101100)
      ---t7--> M004: 1(0101011000101010)
M002: 1(10110110110110110)---t1--> M001: 0(01010110110110110)
      ---t3--> M005: 2(10001010110110110)
      ---t5--> M006: 2(10111000101011011)
      ---t7--> M007: 2(1011011100010101)
      ---t9--> M008: 2(101101101100010)
M003: 1(0101000010101100)---t2--> M006: 2(1011000010101101)
      ---t6--> M001: 0(01010110110110110)
M004: 1(0101011000101010)---t2--> M007: 2(1011011100010101)
      ---t6--> M001: 0(01010110110110110)
M005: 2(10001010110110110)---t4--> M002: 1(10110110110110110)
      ---t7--> M009: 3(100010100010101)
      ---t9--> M010: 3(100010101100010)
M006: 2(10111000101011011)---t1--> M003: 1(0101000010101100)
      ---t6--> M002: 1(10110110110110110)
      ---t9--> M011: 3(1011100010100010)
M007: 2(1011011000101010)---t1--> M004: 1(0101011100010101)
      ---t3--> M009: 3(100010100010101)
      ---t8--> M002: 1(10110110110110110)
M008: 2(101101101101100010)---t3--> M010: 3(100010101100010)
      ---t5--> M011: 3(1011100010100010)
      ---t10--> M002: 1(10110110110110110)
M009: 3(100010100010101)---t4--> M007: 2(1011011100010101)
      ---t8--> M005: 2(1000101011011011)
M010: 3(100010101100010)---t4--> M008: 2(101101101100010)
      ---t10--> M005: 2(1000101011011011)
M011: 3(1011100010100010)---t6--> M008: 2(101101101100010)
      ---t10--> M006: 2(1011000101011011)
    
```

Figura 5. Grafo de Alcanzabilidad en formato de texto

III. CÁLCULO DISTRIBUIDO DEL GRAFO DE ALCANZABILIDAD

Una forma eficiente de calcular el grafo de alcanzabilidad en un RP es distribuir el cálculo de tal manera que se pueda aprovechar el paralelismo presente en un esquema Cliente/Servidor.

En este artículo se presenta un algoritmo distribuido para el cálculo del grafo de alcanzabilidad de una RP implementado en la plataforma RMI de Java. El grafo se calcula mediante una expansión en amplitud por nodo.

En este escenario, cada Servidor proporciona servicios los cuáles son registrados en el *Registro de RMI*. Entonces, el Cliente puede buscar dichos servicios en el Registro de RMI.

En la solución propuesta en este trabajo, todos los Servidores proporcionan los mismos servicios:

- 1) Registro de Matriz de Entradas y Matriz de Salidas.
- 2) Cálculo de marcados siguientes a partir de un marcado dado.

Con el Registro de las Matrices de Entradas y Salidas los servidores mantienen una copia local de la estructura de la red. Esto permite ahorrar ancho de banda al momento de solicitar a los servidores el cálculo de los estados siguientes, ya que solo es necesario enviar el marcado a partir del cual se requiere calcular los estados siguientes. Cada servidor utiliza su copia local de las matrices para encontrar las transiciones habilitadas y calcular los marcados siguientes.

El Cliente realiza las siguientes funciones:

- 1) Busca los servidores que proporcionan los servicios.
- 2) Mantiene una cola de los marcados por expandir.
- 3) Asigna continuamente el siguiente marcado en la cola de marcados por expandir al primer Servidor libre.

- 4) Construye el grafo de alcanzabilidad global con los resultados calculados por cada servidor.

A continuación se presenta el algoritmo en pseudocódigo para el Cliente.

Algoritmo 2. *Cliente_Grafo_de_Alcanzabilidad;*
Entradas: *I: Matriz de Entradas; O: Matriz de Salidas, M: Marcado Inicial, S: Servidores*

1. Enlazarse a los servidores y registrar con ellos las matrices *I, O*.
2. Inicializar la Cola con el marcado inicial *M*;
3. Hacer {
 - a. Sacar el siguiente marcado M_k de la Cola;
 - b. Crear un hilo H_i para solicitar al siguiente servidor disponible el *Cálculo_Estados_Siguientes*, desde M_k ;
 - i. Dentro de H_i , cada servidor retorna una estructura *S* con las transiciones habilitadas y sus estados alcanzados de la forma: $S_i = M_k - t_i - > M$;
 - ii. H_i guarda la información S_i ;
 - iii. Para todo estado siguiente M_j , Si es nuevo marcado entonces H_i lo agrega a la Cola;}
- } Repetir mientras la cola no esté vacía;
4. Fin;

El algoritmo en pseudocódigo para el Servidor se muestra a continuación.

Algoritmo 2. *Cálculo_Estados_Siguientes;*
Entradas: *I: Matriz de Entradas; O: Matriz de Salidas, M_k ;*

1. Calcular el conjunto E_k de transiciones habilitadas en M_k ;
2. Para cada transición $t_i \in E_k$
 - c. Calcular su estado siguiente M_j con la ecuación de estados (1);
3. Guardar la estructura $S_i = M_k - t_i - > M_j$;
4. Retornar *S*;

La Tabla 1 muestra una corrida del Algoritmo 2. El Cliente realizó la asignación de tareas conforme a la Cola de tareas.

Tabla 1. Asignación de tareas

1)	Cola	M001			
	Asignada a	Server_1			
2)	Cola	M002	M003	M004	
	Asignada a	Server_2	Server_3	Server_4	
3)	Cola	M005	M006	M007	M008
	Asignada a	Server_1	Server_2	Server_3	Server_4
4)	Cola	M009	M010	M011	
	Asignada a	Server_1	Server_2	Server_3	

En las Figuras 6, 7, 8 y 9 se muestran los resultados calculados por los servidores Server_1, Server_2, Server_3, y Server_4, respectivamente, correspondientes al paso 3) en la Tabla 1. Se puede observar claramente, la parte del Grafo de Alcanzabilidad que cada servidor ha calculado.

M001: 0(0 1 0 1 0 1 1 0 1 1 0 1 1 0 0 1 0 1 0 0)	---t2-->	M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 0 1)
	---t5-->	M003: 1(0 1 0 1 0 0 0 1 0 1 0 1 1 0 0 0)
	---t7-->	M004: 1(0 1 0 1 0 1 1 1 0 0 0 1 0 1 0 0 0)
M005: 2(1 0 0 0 1 0 1 0 1 1 0 1 1 0 1 1 0 1)	---t4-->	M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1)
	---t7-->	M009: 3(1 0 0 0 1 0 1 0 0 0 1 0 1 0 1 0 1)
	---t9-->	M010: 3(1 0 0 0 1 0 1 0 1 1 0 0 0 1 0 0 1 0)
M009: 3(1 0 0 0 1 0 1 0 0 0 1 0 1 0 1 0 1)	---t4-->	M007: 2(1 0 1 1 0 1 1 0 0 0 1 0 1 0 1 0 1)
	---t8-->	M005: 2(1 0 0 0 1 0 1 0 1 1 0 1 1 0 1 0 1)

Figura 1. Resultados calculados por el Server_1.

M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1)	---t1-->	M001: 0(0 1 0 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0)
	---t3-->	M005: 2(1 0 0 0 1 0 1 0 1 1 0 1 1 0 1 1 0 1)
	---t5-->	M006: 2(1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 1 0 1)
	---t7-->	M007: 2(1 0 1 1 0 1 1 0 0 0 1 0 1 0 1 0 1)
	---t9-->	M008: 2(1 0 1 1 0 1 1 0 1 1 0 0 0 1 0 0)
M006: 2(1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 1 0 1)	---t1-->	M003: 1(0 1 0 1 0 0 0 1 0 1 0 1 1 0 0 0)
	---t6-->	M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1)
	---t9-->	M011: 3(1 0 1 1 0 0 0 1 0 1 0 0 0 1 0 0)
M010: 3(1 0 0 0 1 0 1 0 1 1 0 0 0 1 0 1 0 1)	---t4-->	M008: 2(1 0 1 1 0 1 1 0 1 1 0 0 0 1 0 0)
	---t10-->	M005: 2(1 0 0 0 1 0 1 0 1 1 0 1 1 0 1 0 1)

Figura 2. Resultados calculados por el Server_2

M003: 1(0 1 0 1 0 0 0 1 0 1 0 1 1 0 0 1 0 0)	---t2-->	M006: 2(1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 1 0 1)
	---t6-->	M001: 0(0 1 0 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0)
M007: 2(1 0 1 1 0 1 1 0 0 0 1 0 1 0 1 0 1 0)	---t1-->	M004: 1(0 1 0 1 0 1 1 0 0 0 1 0 1 0 1 0 0)
	---t3-->	M009: 3(1 0 0 0 1 0 1 0 0 0 1 0 1 0 1 0 1)
	---t6-->	M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1)
M011: 3(1 0 1 1 0 0 0 1 0 1 0 0 0 1 0 1 0 1)	---t6-->	M008: 2(1 0 1 1 0 1 1 0 1 1 0 0 0 1 0 0)
	---t10-->	M006: 2(1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 1 0 1)

Figura 3. Resultados calculados por el Server_3

M004: 1(0 1 0 1 0 1 1 0 1 1 0 0 0 1 0 1 0 0)	---t2-->	M007: 2(1 0 1 1 0 1 1 0 0 0 0 1 0 1 0 1)
	---t8-->	M001: 0(0 1 0 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0)
M008: 2(1 0 1 1 0 1 1 0 1 1 0 0 0 1 0 1 0 0)	---t3-->	M010: 3(1 0 0 0 1 0 1 0 1 1 0 0 0 1 0 0)
	---t5-->	M011: 3(1 0 1 1 0 0 0 1 0 1 0 0 0 1 0 0)
	---t10-->	M002: 1(1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 0 1)

Figura 4. Resultados calculados por el Server_4

La Figura 10 muestra el diagrama de Clases de la Aplicación para el Cálculo Distribuido del Grafo de Alcanzabilidad propuesto en este trabajo.

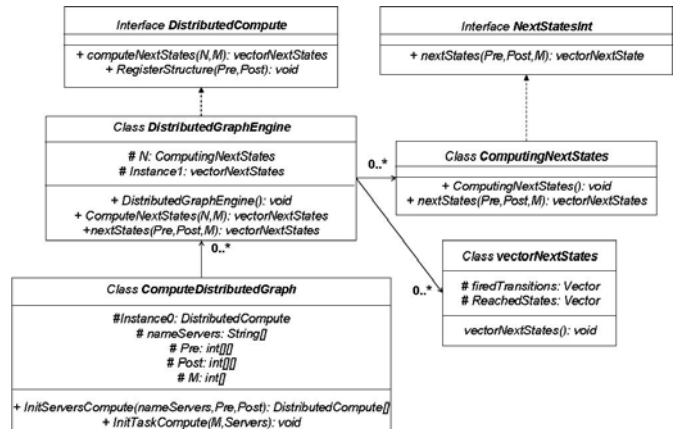


Figura 5. Diagrama de Clases de la Aplicación

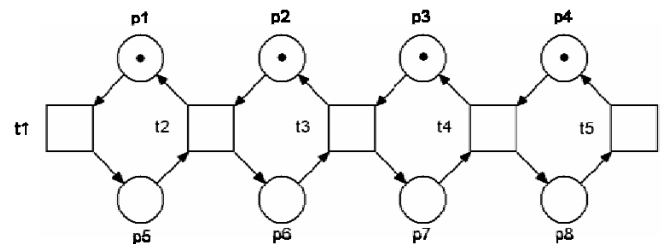


Figura 6. Red de Petri Ejemplo1.

La clase ComputeDistributedGraph implementa el Algoritmo 2. Como primer paso, localiza a los servidores y solicita el registro de las matrices de entradas y salidas. Enseguida, solicita el cálculo de los estados siguientes y almacena los resultados devueltos por los servidores. La clase DistributedGraphEngine realiza los procesos especificados en el Algoritmo 3. Este realiza el registro de las matrices de entradas y salidas una sola vez al inicio de la ejecución de la aplicación, puesto que es información que no cambia y se utiliza durante todos los cálculos. Adicionalmente, calcula las transiciones habilitadas a partir de un marcado determinado y evalúa la ecuación de estados para cada una de ellas invocando al método Calculo_Estados_Siguientes. Como resultado de

esta acción, se retornan las transiciones habilitadas y sus respectivos marcados alcanzados.

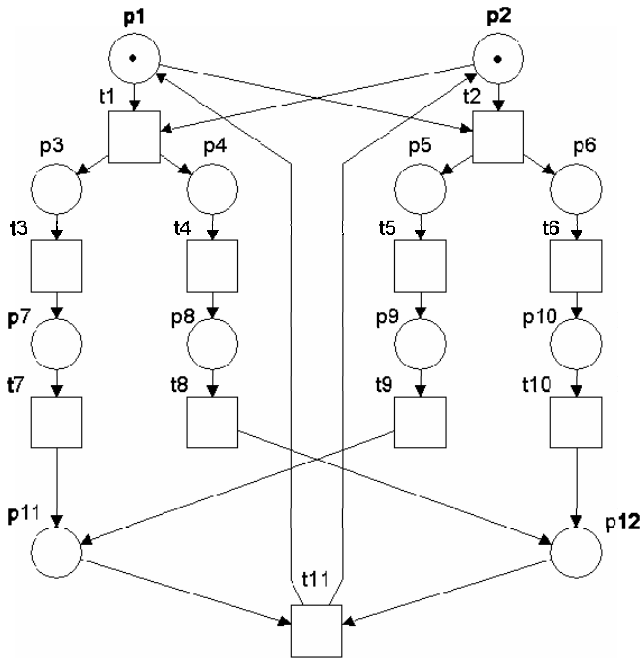


Figura 7. Red de Petri Ejemplo2.

IV. RESULTADOS EXPERIMENTALES

Con la finalidad de obtener comparativas entre los tiempos de cálculo del grafo de alcanzabilidad de una implementación centralizada con una distribuida se diseñó una red en estrella utilizando un switch y 6 PC's, donde una PC es el Cliente y las otras 5 PC's son los servidores, tal como se muestra en la Figura 13.

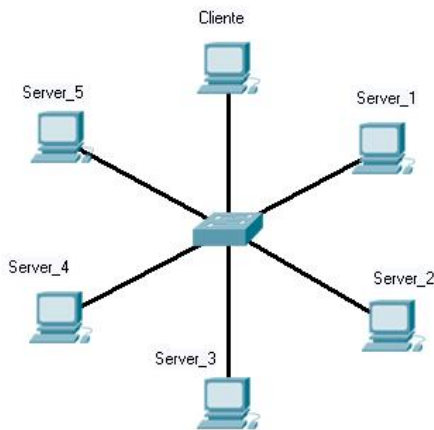


Figura 8. Topología de la Red

Las características de los componentes de la red son los siguientes: el switch utilizado es un Cisco Catalyst Express 500 Series; el cable utilizado para la conexión del switch con las PC's es UTP Cat 5e de 3 metros con estándar T568B; y las PC's utilizadas son todas HP Pavilion Media Center m7350la. Por lo tanto, la velocidad de conexión en la red es de 100Mbps.

En la Tabla 2 se muestra un análisis cualitativo comparativo entre el cálculo centralizado y cálculo distribuido para la RP Ejemplo1 que se muestra en la Figura 6 considerando diferentes marcados iniciales. En la columna *Marcas por lugar* se indica la cantidad de marcas para los lugares p1, p2, p3, p4, representando el marcado inicial de la RP Ejemplo1. Se puede observar que cuando el marcado inicial es $M0=[1,1,1,1,0,0,0,0]^T$ el tiempo de cálculo es menor en la implementación centralizada que en la distribuida, esto se debe al tiempo de comunicación en la red. A partir del marcado inicial $M0=[3,3,3,3,0,0,0,0]^T$ el tiempo de cálculo en la implementación distribuida es menor, lo cual representa una mejora de esta implementación sobre la centralizada. Se observa claramente que existe una disminución considerable en tiempo de cálculo entre las dos implementaciones conforme el número de marcas aumenta.

Tabla 2. Tabla comparativa para la RP Ejemplo1.

Marcas por lugar	Tiempo Centralizado (seg)	Tiempo Distribuido (seg)	Estados Generados
1	0.5731	0.6224	16
3	4.0923	0.8823	256
5	19.4278	2.4469	1296
7	56.7556	6.7834	4096
9	146.8442	20.0179	10000

Para la RP Ejemplo2 que se muestra en la Figura 7, la situación es similar. En la tabla comparativa que se presenta en la Tabla 3, se puede observar que para esta RP el tiempo de cálculo en la implementación distribuida para la obtención del grafo de alcanzabilidad es menor incluso desde el marcado inicial $M0=[1,1,0,0,0,0,0,0,0,0,0]^T$.

Tabla 3. Tabla comparativa para la RP Ejemplo2.

Marcas por lugar	Tiempo Centralizado (seg)	Tiempo Distribuido (seg)	Estados Generados
1	0.5237	0.4834	18
3	4.1923	2.7388	1444
5	239.9225	93.845	22180

V. CONCLUSIONES

En este artículo se presenta un algoritmo para el cálculo distribuido del grafo de alcanzabilidad de una Red de Petri. Se utiliza una técnica de expansión en amplitud por nodos para la construcción del grafo. Se emplea la plataforma RMI de Java para su implementación. Se utiliza una topología de red tipo estrella con 6 PC's, un cliente y cinco servidores para implementar el algoritmo distribuido propuesto y comparar los resultados obtenidos con la implementación de un algoritmo centralizado. Los servidores realizan el cálculo de los marcados alcanzados por el disparo de una sola transición a partir de un marcado dado. Un cliente se encarga de distribuir las tareas a los servidores utilizando un método similar al Round-Robin. Con los resultados parciales calculados por los servidores, el cliente construye la solución total. La implementación del algoritmo distribuido presenta una disminución considerable en el tiempo de cálculo con respecto a la implementación centralizada.

Como trabajo futuro, se pretende utilizar una técnica similar para el cálculo distribuido de invariantes mínimos en una Red de Petri.

VI. REFERENCIAS

- [1] James Lyle Peterson, "*Petri Net Theory and the Modeling of Systems*", Prentice-Hall 1981.
- [2] Paritosh K. Pandya, Jaikumar Radhakrishnan, "*FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science*", 2003 Springer.
- [3] Branislav Rován, Peter Vojtáš, "*Mathematical Foundations of Computer Science*", 2003 Springer.
- [4] Alex Berson, "*Client/server Architecture*", 1996 McGraw-Hill.
- [5] William Grosso, "*Java RMI: Designing & Building Distributed Applications*", 2002 O'Reilly.
- [6] Jörg Desel, Javier Esparza, "*Free Choice Petri Nets*", 1995 Cambridge University Press.
- [7] Ida M. Flynn, Ann McIver Mchoes, "*Sistemas operativos*", 2001 Thomson.
- [8] Dieter Jungnickel, "*Graphs, Networks and Algorithms*", Algorithms and Computation in Mathematics Vol. 5, 2007 Springer.

MODEL SYNTHESIS OF CONCURRENT DISCRETE EVENT SYSTEMS USING INTERPRETED PETRI NETS

M.E. Meda-Campaña

Departamento de Sistemas de Información
 Universidad de Guadalajara
 Periférico Nte. 799, Los Belenes, Módulo L-308, Zapopan, Jal. 45100, México
 Tel. (33) 3770-3352, correo electrónico: emeda@cucea.udg.mx

ABSTRACT

In this paper is defined the asymptotic identification problem for DES and is introduced a procedure to detect concurrent transitions. This approach mainly differs from others identification approaches in the way of how the model of the system is computed and also in the class of computed model describing the system. This novel identification approach proposes to compute an IPN model describing the behavior of the unknown DES. The construction of this model, is incrementally performed by computing a new model from new on-line measurements of the system outputs. One advantage of this approach is that it is not needed to have a priori information of the system (commonly called a positive sample) because the inference mechanism will operate on-line processing the output signals.

Keywords: Discrete Event Systems, Petri Nets, System identification

I. INTRODUCTION

This paper addresses the identification problem for concurrent Discrete Event Systems (DES). The identification process is devoted to obtain a mathematical model from the observation of the system behavior, which is described by the evolution of its input and output signals. The class of systems dealt in this work are those in which the entire state of the DES cannot be determined from the measurement of its outputs, this is because this class of systems represents more realistic DES since not always is possible to have completely instrumented a system with sensors. However this

class of DES must exhibit the event-detectability property.

Given the well-known features of Petri Nets (PN), in this paper they will be used as the mathematical formalism to describe DES.

The adopted identification approach is a passive one in which the input signals of the system are not manipulated; the strategies proposed are translated into procedures executed on-line, building the models progressively as new information of system behavior is detected. These models represent the observed behavior of the system and they are approaching asymptotically to the actual model of the system. Every sequence of events computed from the observed system behavior is analyzed and if it provides new information about the system then the computed model is updated.

The model synthesis procedure performs mainly two tasks: the computation of the measurable part of the system and the inference of the non-measurable part of the system.

The first task is made directly from the observation of the output signals of the system, while the second task, rather difficult, derived a more detailed study about the non-measurable dependencies computed into a model. The proposed algorithm to updated the non-measurable places is of lineal complexity in the number of the transitions computed and the transition sequences detected. Novel concepts and algorithms for the identification of DES modeled by binary IPN have been reported in [9][10][11].

The practical interest in addressing this problem is the modeling of legacy systems (whose functioning is partially known), and in reverse engineering of

digital systems. Identification methods could be also applied in verification of digital/software systems or used as a learning mechanism of intelligent systems.

A similar problem has been addressed in the area of grammatical learning. Several works have proposed methods to obtain finite automata (FA) for subclasses of regular languages [2][5], as well as Petri nets (PN) [3][6].

This article is organized as follows. In section II, is introduced the basic definitions of IPN and also the basics of the asymptotic identification approach while in section III is presented the identification approach for concurrent DES. Finally, the concluding remarks and discussion of future work follow in section IV.

II. BACKGROUND

In this section, is presented the basic concepts of Interpreted PN (IPN) [7], an extension to the PN that can represent input and output DES signals. Good references about PN are [4][13][15][16]. In this section also is introduced the basics of the asymptotic identification approach [9][10][11].

BASIC CONCEPTS OF IPN

Definition 1: An Interpreted Petri Net is the 5-tuple $Q = (N, \Sigma, \Phi, \lambda, \Pi)$ where N is a PN with initial marking M_0 , $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_r\}$ is a finite set of elements σ_i called input symbols, $\Phi = \{\phi_1, \phi_2, \dots, \phi_p\}$ is a finite set of elements ϕ_i called output symbols, $\lambda : T \rightarrow \Sigma \cup \{\varepsilon\}$ is a labeling transition function, where ε is the null symbol and $\varphi : R(N, M_0) \rightarrow \{\Phi \cup \{\varepsilon\}\}^q$ is an output function; where $R(N, M_0)$ is the set of reachable markings and q is the number of sensors associated to places in Q .

In an IPN, if a transition t_j is enabled and the input signal $\lambda(t_j) = a_i \neq \varepsilon$ is present then t_j must fire else if $\lambda(t_j) = \varepsilon$ then t_j can be fired.

Definition 2: A place $p_i \in P$ is said to be measurable if it has a sensor signal assigned, and non measurable otherwise. Non-measurable places are depicted as dark circles.

This paper focuses on the case where $\Phi = Z^+$ (the non-negative integer numbers), and $\varphi : R(N) \rightarrow \{Z^+\}^q$ is a linear function that can be represented by a $q \times n$ matrix $\varphi = [\varphi_{ij}]$, where the i -th row vector φ_i of φ is the transpose of the elemental vector e_j ($e_j[i \neq j] = 0, e_j[j] = 1$), if p_j is the i -th measurable place ($\varphi_i = e_j^T$), according to the order given by the place labeling.

The state equation of an IPN is completed as:

$$\begin{aligned} M_{k+1} &= M_k + Cv_k \\ y_k &= \varphi M_k \end{aligned} \quad (1)$$

where $y_k = \varphi M_k$ is a $q \times 1$ vector called output symbol.

Notice that an output symbol is the marking vector of the measurable places in the marking M_k .

The incidence matrix C of an IPN Q can be decomposed as $C = \begin{bmatrix} \varphi C \\ \gamma C \end{bmatrix}$, where φC and γC

represent the measurable and non measurable places of Q respectively, and γ is a linear function defined in a similar way than φ function but considering the non measurable places instead of the measurable places.

Definition 3: The characteristic vector or Parikh vector of a transition sequence σ is a vector $\vec{\sigma} \in \{Z^+\}^m$, where $m = |T|$. $\vec{\sigma}(i)$ represents the number of firings of t_i in σ .

Definition 4: Let Q be an IPN. Two transitions t_i and t_j of Q form a dependency $p_k = [t_i, t_j]$ if $\exists p_k$ such that $p_k \sqsubset t_i \bullet$ and $p_k \sqsubset \bullet t_j$. If p_k is a non-measurable place, then $[t_i, t_j]$ will be called a non-measurable dependency (**NDep**), otherwise will be called a measurable dependency (**MDep**).

BASICS OF ON-LINE IDENTIFICATION

The model Q of a DES S_f is unknown in principle; the hypothesis of Q known is held just for proving the convergence of the proposed identification technique. In the remaining of this paper will be referred to a system S_f as its representation Q in IPN terms, and will be called system model or just

system.

The asymptotic identification approach [9][10][11], consists in compute on-line an IPN model Q_i describing the behavior of an unknown system Q from the measure of its output symbols. Every time a cyclic behavior is detected, the previous computed model Q_{i-1} is updated such that the new (updated) model Q_i acquires more detail of the system than Q_{i-1} . Hence, an IPN model sequence $Q_0, Q_1, \dots, Q_{n-1}, Q_n$ is computed as Q evolves, such that inequality $f(Q, Q_i) \leq f(Q, Q_{i-1})$ is fulfilled, where f is a similitude function defined in [11].

Next definition states how to determine that a cycle in the system has occurred from the observation of its output symbols.

Definition 5: Let Q be an IPN. If $w_o = \varphi(M_0) \dots \varphi(M_j)$ is an output sequence generated by Q ; and $\sigma = t_1 \dots t_k$ is the firing sequence detected when w_o is measured, then σ is an m-word if $\varphi(M_0) = \varphi(M_j)$.

Notice however that these cycles not always are t-semiflows of the system. This fact will lead to make wrong conjectures about how the non-measurable places are connected in the unknown system model Q .

Every transition t_i in the firing sequence σ of previous definition is computed from any two consecutive output symbols $\varphi(M_i)$ and $\varphi(M_{i-1})$ as:

$$\varphi C_Q(\bullet, t_i) = \varphi(M_i) - \varphi(M_{i-1}) \quad (2)$$

When a m-word w_i is computed all measurable dependencies related with its transitions are computed, hence the φC submatrix of a system model Q is computed correctly when all transitions of Q are fired. However the computation of the non-measurable places (rows of matrix γC) is not as straight as the computation of the measurable places (rows of matrix φC). Some non-measurable places must be inferred from several evolutions of the system. The non measurable places are computed according to: 1) preserve the firing order of the transitions in the current m-word and 2) preserve the order in which the m-words have been

computed.

The basic identification step consists in compute a sequence with the transitions of the current computed m-word adding non measurable places between any two consecutive transitions in order to form a t-component in the model; this is stated in next criterion.

Criterion 1: Let Q be a system model, Q_n be the computed model for Q , $w_n = \dots t_i t_j \dots$ be the current computed m-word from the observed behavior of Q and t_i and t_j be any two consecutive transitions in w_n . If there exists not a dependency $[t_i, t_j]$ in Q_n then the non-measurable place $p_x = [t_i, t_j]$ can be computed to preserve the firing of t_i before t_j in Q_n .

For each NDep $[t_i, t_j] = p_k$, a vector $u_k = [v_1, \dots, v_i \dots v_j \dots v_r]$ is computed; where r is the number of detected transitions and $v_i = 1, v_j = -1$ and $v_x = 0 \quad \square x \neq i, j$.

To compute the incidence matrix C of Q , the columns $\varphi C_{Q_i}(\bullet, t_i)$ and u_{ij} vectors are arranged as follows:

$$C = \begin{bmatrix} \varphi C(\bullet, t_1) & \dots & \varphi C(\bullet, t_r) \\ & & u_j \\ & & \vdots \\ & & u_k \end{bmatrix}$$

Notice that the u_i vectors are the rows of the matrix γC of the computed model Q_i .

According to previous evolutions of the system and the new m-word computed, the non-measurable places can be updated as follows:

- If there exists a NDep $[t_i, t_j] = p_k$ and it is needed to form another NDep $[t_x, t_y]$ using the same non measurable place p_k , then an arc is added from t_x to p_k and another arc is added from p_k to t_y .
- If a NDep $[t_i, t_j] = p_k$ must be removed and p_k belongs to another NDep, then the input and the output arcs of p_k related with t_i and t_j are removed; only in the case when p_k belongs to a single NDep, the place p_k can be removed.
- If p_i and p_j are two non-measurable places to be merged, then a new non-measurable place p_k is computed such that $\bullet p_k = \bullet p_i \cup \bullet p_j$ and $p_k \bullet = p_i \bullet$

$\cup p_j$. In the incidence matrix C_{Q_i} the rows i and j are added forming the row k , and the rows i and j are removed from C_{Q_i} .

Identification considering just the output information of the system

Due to the problem of identification of DES is relatively a new research area, here in this work will be considered that each input to the system is reflected in the output, it means that even the system is not completely instrumented the information provided by the sensors is enough to detect any change of state occurred in the system Q .

Hence, a model Q_i will be built using only the output information of Q . For this case the identification procedure will require that all columns of φC can be detected and uniquely determined from its output information. Hence in this work will be considered a particular case of the event detectability property.

Definition 6 Let Q be an IPN. Q is event-detectable by the output if every transition firing can be uniquely determined by the knowledge of its output signals.

The characterization of the event-detectability property considering only the output information of the system is that all the columns of φC matrix be non null and different from each other. This is stated in the next proposition.

Proposition 1 Let Q be an IPN. Q is event-detectable by the output i :

1. $\forall t_i \in T$ it is fulfilled that $\varphi C \vec{t}_i \neq \vec{0}$ and
2. $\forall t_i \neq t_j \in T$ it holds that $\varphi C \vec{t}_i \neq \varphi C \vec{t}_j$

The first condition of proposition 1 states that if two transitions have the same input signal assigned then their columns in φC matrix must be different to can distinguish them, while the second condition refers to the fact that any change of marking allows to determine the occurrence of a transition.

This property makes possible that every transition can be detected using equation 2 and hence allows computing the measurable part of a DES.

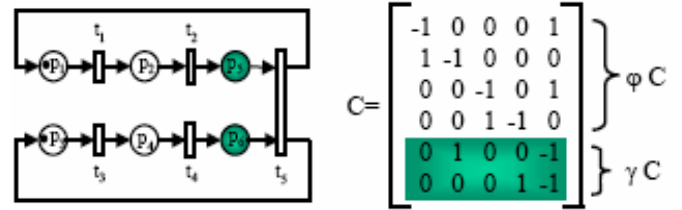


Figure 1: System model Q and its incidence matrix.

Example. Consider the IPN Q depicted on figure 1, the system to be identified. The non-measurable part of Q is computed using equation 2 as:

$$\begin{bmatrix} \varphi M_1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \xrightarrow{t_1} \begin{bmatrix} \varphi M_2 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \xrightarrow{t_2} \begin{bmatrix} \varphi M_3 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{t_3} \begin{bmatrix} \varphi M_4 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{t_4} \begin{bmatrix} \varphi M_5 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{t_5} \begin{bmatrix} \varphi M_1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

where all the elements $\varphi(M_i) - \varphi(M_{i-1})$ are the columns of the matrix φC_Q .

Remark 1 Using this approach it is not possible to detect the event $t_j \parallel t_k$ (the simultaneous firing of t_j and t_k); thus the detection of a multiple change in the entries of the output symbol is assumed to be caused by the firing of one transition.

Notice that each column i of φC_Q represents only the input and output measurable places of the transition t_i .

III. MODEL SYNTHESIS OF CONCURRENT SYSTEMS

In previous works [9][10][11][12], were presented procedures to identify specific behaviors of DES, now in this work; is presented how to identify concurrent DES.

The main characteristic of a concurrent DES is that

some activities can be executed concurrently. Concurrent activities are those that can occur independently during the execution of a cycle in a DES. This characteristic can be easily expressed in terms of Petri nets. In general, two transitions are said to be concurrent if one transition may fire before, after or in parallel with other transition, this occurrence is considered in a determined t-component.

The transitions t_1 and t_3 of the IPN depicted on figure 2 are an example of concurrent transitions.

It is possible that in any m-word $w_i = \dots t_i t_j \dots$ two concurrent transitions t_i and t_j occur consecutively, for example in the m-word $w_1 = t_1 t_3 t_2 t_4 t_5$ of the same IPN (figure 2) t_1 and t_3 occurs consecutively.

At the beginning of the identification procedure it is not possible to know if any two consecutive transitions are concurrent transitions or not, then if there exists not a dependency between any two consecutive transitions t_i and t_j in the current m-word $w_n = \dots t_i t_j \dots$ and there exists not evidence that t_j has occurred before t_i in another m-word, then to constrain the firing of t_i before t_j it is computed a NDep $p_k = [t_i, t_j]$ using criterion 1. If t_i and t_j are concurrent transitions then $p_k = [t_i, t_j]$ is a wrong NDep, i.e. this NDep does not belong to the system model to be identified. This wrong NDep will be removed when another m-word $w_m = \dots t_j \dots t_i \dots$ be computed in which t_j occurs before t_i . Next it is presented the definition of concurrent transitions, the needed properties to detect such transitions and the procedures to update a previous model when concurrent transitions are detected.

Definition 7 Let Q be an IPN and X_k be a t-component of Q , two transitions t_p and t_q of X_k are concurrent transitions if there exists two transition sequences $\sigma_i \neq \sigma_j \in \mathcal{L}(X_k)$, such that $\sigma_i = \dots t_p t_q \dots$ and $\sigma_j = \dots t_q t_p \dots$. If there exists two transition sequences $\sigma_x = \dots t_p \dots t_q \dots$ and $\sigma_y = \dots t_p \dots t_q \dots$, t_p and t_q are considered pseudoconcurrent.

Since the two sequences $\sigma_i \neq \sigma_j$ of previous definition are sequences generated by the same t-component they fulfill that $\vec{\sigma}_i = \vec{\sigma}_j$ (i.e. they have the same characteristic vector).

The set of all concurrent transitions in an IPN Q

will be denoted as $Concurrent(Q)$. Two concurrent transitions t_i and t_j are denoted as $c[t_i, t_j]$ that is equivalent to $c[t_j, t_i]$.

Let t_i and t_j be any two consecutive transitions in the current m-word w_n and T be the set of all computed m-words.

To detect if t_i and t_j are concurrent transitions or not, it is searched if there exists an m-word $w_k \neq w_n$ such that $\vec{w}_k = \vec{w}_n$ in which t_j occurs before t_i . The existence of the m-word $w_k = \dots t_j \dots t_k \dots$ implies that t_i and t_j are concurrent transitions (by definition 7). However the absence of $w_k = \dots t_j \dots t_k \dots$ in T does not implies that t_i and t_j are not concurrent, it only indicates that it has not been already computed an m-word in which t_j occurs before t_i .

Example. Consider the system model Q depicted on figure 2, the transitions t_1 and t_3 are concurrent transitions, however when Q has just generated the m-words $w_1 = t_1 t_3 t_2 t_4 t_5$ and $w_2 = t_1 t_2 t_3 t_4 t_5$ it is not possible to determine that t_1 and t_3 are concurrent transitions since from the observed behavior of Q , t_1 occurs always before than t_3 .

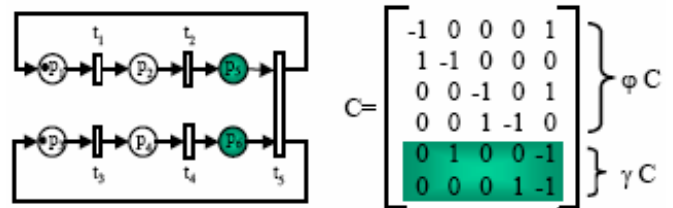


Figure 2: System model and its incidence matrix.

Proposition 2 Let Q be a system model, $w_n = \dots t_i t_j \dots$ be the current computed m-word from the last output word produced by a t-component X_k of Q and T be the set of all computed m-words from Q . If there exist an m-word $w_k = \dots t_j \dots t_i \dots$ in T such that $w_k \neq w_n$ and $\vec{w}_k = \vec{w}_n$, then t_i and t_j are concurrent transitions.

Proof. By hypothesis if w_k exists in T implies that w_k and w_n belong to the same t-component, from definition 7 t_i and t_j are concurrent transitions due to Q had produced two m-words $w_k = \dots t_j \dots t_i \dots$ and $w_n = \dots t_i t_j \dots$ in which the transitions t_i and t_j occurs in different order. ■

Corollary 1 Let Q be a system model, Q_i be the computed model for Q and $w_n = \dots t_i \dots t_x \dots$ be the current m-word computed from the last output word generated by the t-semiflow X_k of Q . If there exists a NDep $[t_x, t_i]$ in Q_i , then t_i and t_x are concurrent transitions.

Proof. By criterion 1 a NDep is built to form a sequence between any two consecutive transitions of a computed m-word, hence if there exists a NDep $[t_x, t_i]$ in a t-component of the computed model Q_i implies that from the observed behavior of Q was already computed a m-word $w' = \dots t_x t_i \dots$ in which t_x occurs before t_i , by proposition 2 t_i and t_x are concurrent transitions since in the current m-word $w_n = \dots t_i \dots t_x \dots$, t_i occurs before t_x . ■

Let Q be an IPN, Q_i be the computed model for Q and t_i and t_x be any two concurrent transitions detected from the current computed m-word $w_n = \dots t_i \dots t_x \dots$. If there exists a NDep $[t_x, t_i]$ in $Dep(Q_i)$, then the NDep $[t_x, t_i]$ is removed from $Dep(Q_i)$.

Notice that after remove a NDep $[t_i, t_j]$ in the computed model, could occur that t_i or t_j remain as a sink or as a source transition respectively such that $t_i \bullet = \square$ or $\bullet t_j = \square$. Assume that the NDep $[t_i, t_j] = p_k$ is removed from $Dep^u(Q_n)$: the transition t_i could become a sink transition when the only NDep formed with t_i is using the non measurable place p_k and t_i has not another output place. Similarly, t_j could become a source transition when the only NDep formed with t_j is using the non-measurable place p_k and t_j has not another input place.

Will be considered that a transition remain as a sink or as a source transition after the evaluation of all transitions in the current m-word w_n due to it is possible that in w_n t_i has another successor transition different from t_j or t_j has another predecessor transition different from t_i , computing a new NDep.

Example. Consider the system model Q depicted on figure 3.a. The model Q_1 depicted on figure 3.b. is computed when the first m-word $w_1 = t_1 t_2 t_3 t_4 t_5$ is detected. A NDep $[t_2, t_3]$ is computed (among

others) using criterion 1 due to t_2 and t_3 are consecutive transitions in w_1 and there exist not any dependency between these two transitions. Assume now that $w_2 = t_3 t_1 t_2 t_4 t_5$ is the next m-word computed. Then the transitions t_2 and t_3 are determined to be concurrent transitions because t_2 occurs before t_3 in w_1 (proposition 2 and corollary 1). Since there exists a NDep $[t_2, t_3]$ in $Dep(Q_2)$, it is removed to allow the firing of t_3 before t_2 as stated in w_2 (criterion 1).

After remove the NDep $[t_2, t_3]$ from $Dep(Q_2)$, t_2 remains as a sink transition and t_3 remains as a source transition; this is illustrated on figure 3.c.

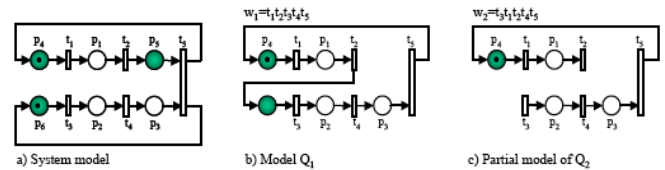


Figure 3: System model and its computed models.

However, the existence of sink or source transitions after removing a wrong NDep will depend on the computed m-words, i.e. in the observed behavior of the system, and also in how the non measurable places are allocated into the net. This fact is illustrated in the next two examples.

Example. Consider again the system model Q depicted on figure 3.a and its first model Q_1 depicted on figure 3.b computed when the m-word $w_1 = t_1 t_2 t_3 t_4 t_5$ was detected. If the second m-word computed is $w_2' = t_3 t_4 t_1 t_2 t_5$ (instead of $w_2 = t_3 t_1 t_2 t_4 t_5$ like in previous example) then the computed model Q_2' is equal to Q , because even the NDep $[t_2, t_3]$ is removed; the NDep $[t_2, t_5]$ is computed due to t_2 and t_5 are consecutive transitions in w_n and hence t_2 does not appear as a sink transition after evaluates w_2' . The NDep $[t_5, t_3]$ is computed since t_3 and t_5 are the first and the last transitions of w_n and as was stated previously they are considered as consecutive transitions. Hence, t_3 does not remain as a source transition.

Example. Now, consider the system model Q depicted on figure 4 which has the same structure as the IPN depicted on figure 3, however they have a different interpretation since the number and

allocation of the non measurable places in these two IPN is different. The model Q_1 depicted on figure 4.b is computed when the m-word $w_1 = t_1t_2t_3t_4t_5$ is detected. When the m-word $w_2 = t_3t_1t_2t_4t_5$ is computed it is removed the NDep $[t_2, t_3]$ since in this m-word t_3 occurs before t_2 , notice that t_3 remains as a source transition as shown in the partial model Q_2 depicted on figure 4.c. In this example the transition t_2 does not remain as a sink transition as in the previous example.

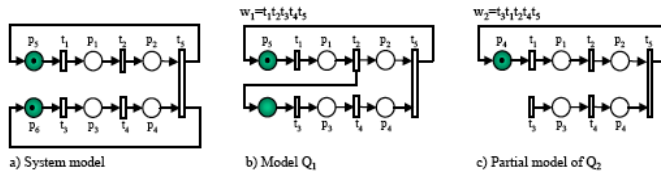


Figure 4: System model and its computed models.

In order to rebuild the t-component associated to w_n , when a transition t_i remains as a sink or as a source transition it is needed to compute an NDep, this new NDep will be computed considering the predecessor or the successor of such a transition in the current m-word w_n , if t_i is a source or a sink transition respectively. This NDep is computed using the following criterion.

Criterion 2 Let w_n be the current computed m-word and Q_i be the computed model for a system model Q ,

Case 1: If t_i remains as a sink transition in Q_i then an NDep $[t_i, t_x]$ is added to $Dep(Q_i)$ such that t_x is the successor transition of t_i in w_n . If t_i and t_x are concurrent transitions then t_x is now taken as the next successor of t_i in w_n .

Case 2: If t_j remains as a source transition in Q_i then an NDep $[t_y, t_j]$ is added to $Dep(Q_i)$ such that t_y is the predecessor transition of t_j in w_n . If t_y and t_j are concurrent transitions then t_y is now taken as the previous predecessor of t_j in w_n .

Proposition 4 Let Q be a system model, Q_i be the computed model for Q , w_n be the current computed m-word and t_i be a sink transition or t_j be a source transition. If the criterion 2 is used to connect a sink or a source transition in Q_i , then it is fulfilled that the observed behavior of Q is preserved in Q_i .

Proof. Case 1: Since t_i is a sink transition implies that $t_i \bullet = \square$, in order to compute an output non-measurable place it is searched a successor t_x of t_i in w_n . Since t_i and t_x are consecutive transitions in w_n then it is possible to compute the NDep $[t_i, t_x]$, however if t_i and t_x are concurrent transitions this NDep cannot be computed because there exists another m-word in which t_x occurred before t_i ; then the next successor of t_i in w_n it is chosen to form the NDep $[t_i, t_x]$. Hence when t_i is connected with a successor, the behavior of Q_i correspond to the observed behavior of Q since t_i has occurred before t_x in all the m-words computed.

Case 2: Since t_j is a source transition implies that $\bullet t_j = \square$, in order to compute an input non-measurable place it is searched a predecessor t_y of t_j in w_n . Since t_y and t_j are consecutive transitions in w_n then it is possible to compute the NDep $[t_y, t_j]$, however if t_y and t_j are concurrent transitions this NDep cannot be computed because there exists another m-word in which t_j occurred before t_y , then the previous predecessor of t_j in w_n is chosen to form the NDep $[t_y, t_j]$. Hence, when t_j is connected with a predecessor t_y the behavior of Q_i correspond to the observed behavior of Q since t_j had occurred after t_y in all the m-words computed. ■

Example. Consider the partial model Q_2 depicted on figure 3.c, after remove the NDep $[t_2, t_3]$ from $Dep(Q_2)$, t_2 remains as a sink transition and t_3 remains as a source transition. So t_2 need to be reconnected with its successor in $w_2 = t_3t_1t_2t_4t_5$ (criterion 2, case 1), the successor transition of t_2 in w_2 is t_4 , since t_2 and t_4 have not been detected to be concurrent transitions because neither in $w_1 = t_1t_2t_3t_4t_5$ nor in $w_2 = t_3t_1t_2t_4t_5$, t_4 occurs before t_2 (thus $[t_4, t_2]$ does not belong to $Concurrent(Q_2)$), then the NDep $[t_2, t_4]$ could be added to $Dep(Q_2)$ as presented in figure 5.a. Now the transition t_3 need to be reconnected with its predecessor (criterion 2 case 2) in $w_2 = t_3t_1t_2t_4t_5$, since t_3 is the first transition in w_2 then its predecessor is the last transition of w_2 due to w_2 is assumed to be a t-semiflow, as t_5 and t_3 have not been detected to be concurrent transitions (proposition 2 and corollary 1), then the NDep $[t_5, t_3]$ could be added to $Dep(Q_2)$ as presented in figure 5.b. Notice that although Q_2 is

not equal to Q it can generate the m -words w_1 and w_2 .

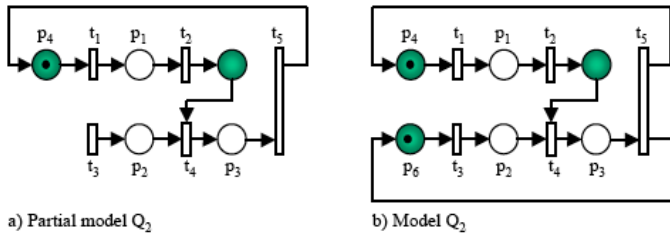


Figure 5: Computed models.

The remaining sink or source transitions will be connected after the current m -word is evaluated by pairs of consecutive transitions detecting new NDep or detecting concurrent transitions.

Remark 2 Notice that if a new NDep $[t_x, t_y] = p_k$ is computed such that t_x is the last transition of an m -word then p_k must be marked.

So, in previous example when the NDep $[t_5, t_3] = p_6$ is computed (figure 5.b) it is added one token to the place p_6 because the transition t_3 cannot be fired in Q_2 if p_6 is not marked.

Next algorithm states how to remove wrong computed NDep, constraining the firing order of any two consecutive concurrent transitions and reconnecting the transitions remaining as a sink or as a source transition. This procedure allows identifying concurrent systems.

Procedure to update a wrong single NDep

Input: The $Dep(Q_n) = Dep(Q_{n-1})$ set, $Concurrent(Q_n) = Concurrent(Q_{n-1})$ and the current computed m -word $w_n = t_m \cdots t_n$
 Output: The updated set $Dep(Q_n)$

1. Let t_i and t_j be any two consecutive transitions in $w_n = t_m \cdots t_n$
 - a. If there exists an NDep $[t_x, t_i]$ or $[t_y, t_j]$ (where $t_x \neq t_n$ and $t_y \neq t_n$) such that t_x or t_y occurs after t_i or t_j in the m -word $w_n = t_m \cdots t_n$ then remove $[t_x, t_i]$ and/or $[t_y, t_j]$ from $Dep^u(Q_n)$ and add it to $Concurrent(Q_n)$. (Corollary 1, Criterion 2)

- b. If $[t_i, t_j] \notin Dep(Q_n)$ and $[t_j, t_i] \notin Concurrent(Q_n)$ then
 - i. If there exist an NDep $[t_j, t_i] \in Dep^u(Q_n)$ then remove $[t_j, t_i]$ from $Dep^u(Q_n)$ and add it to $Concurrent(Q_n)$. (Corollary 1, Criterion 2).
 - ii. If an m -word $w' = \cdots t_j \cdots t_i \cdots$ in which t_j occurred before t_i has been already computed, then add $[t_j, t_i]$ to $Concurrent(Q_n)$. (Proposition 2).
 - iii. If there exists not an MDep $[t_i, t_j]$ in $Dep(Q_n)$ then add the NDep $[t_i, t_j]$ to $Dep(Q_n)$. (Criterion 1).

2. After remove a wrong NDep any transition t_a could remain as a source or as a sink transition (Criterion 2)
 - a. If t_a remains as a source transition then add an NDep $[t_b, t_a]$ to $Dep^u(Q_n)$ such that t_b precedes t_a in the m -word w_n . If t_a is the first transition of $w_n = t_m \cdots t_n$ (i.e. $t_a = t_m$) then t_b will be the last transition of $w_n = t_m \cdots t_n$.
 - b. If t_a remains as a sink transition then add a NDep $[t_a, t_b]$ to $Dep^u(Q_n)$ such that t_b follows t_a in the m -word w_n . If t_a is the last transition of $w_n = t_m \cdots t_n$ (i.e. $t_a = t_n$) then t_b will be the first transition of $w_n = t_m \cdots t_n$ ($t_b = t_m$).
 - c. To form the NDep $[t_b, t_a]$ or $[t_a, t_b]$ we use the step 1 where $t_i = t_b$ and $t_j = t_a$ if t_a is a source transition, else in the case that t_a be a sink transition then $t_i = t_a$ and $t_j = t_b$. If $[t_i, t_j]$ belongs to $Concurrent(Q_n)$ then repeat the step 2 such that $t_b = \bullet(t_b)$ if t_a is a source transition, or $t_b = (t_b)\bullet$ if t_a is a sink transition until $[t_a, t_b]$ does not belong to $Concurrent(Q_n)$.

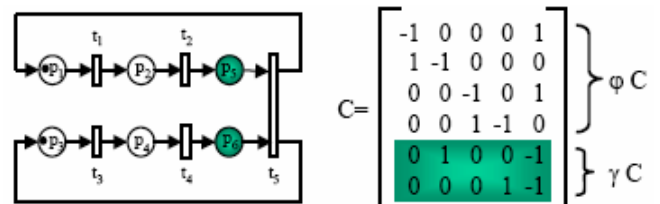


Figure 6. System model and its incidence matrix.

Example. Consider the system model Q depicted

on figure 6. Let $w_1=t_1t_3t_2t_4t_5$ be the first m-word computed. The measurable part of Q , computed when w_1 is detected is depicted on figure 7.a. To preserve the observed behavior of the system the NDeps $[t_1, t_3]$, $[t_3, t_2]$, $[t_2, t_4]$ and $[t_4, t_5]$ are computed according to criterion 1. This is illustrated in step 1.b.iii of the Algorithm; the computed model Q_1 is depicted on figure 7.b.

Consider $w_2=t_3t_4t_1t_2t_5$ the next m-word computed, the updating of the computed model is as follows: Evaluating the consecutive transitions t_3 and t_4 ; since there exist the NDep $[t_1, t_3]$ and $[t_2, t_4]$ in the previous computed model Q_1 and t_3 occurs before t_1 and t_4 occurs before t_2 in the m-word w_2 , then t_1 and t_3 , and t_2 and t_4 are concurrent transitions; hence the NDep $[t_1, t_3]$ and $[t_2, t_4]$ must be removed from $Dep(Q_2)$ (corollary 1, criterion 2) and added to $Concurrent(Q_2)$; this procedure is described in step 1.a. The NDep $[t_3,t_4]$ is not added to the new model Q_2 since there exists the MDep $[t_3, t_4]$ in $Dep(Q_2)$ (step 1.b.iii). The following consecutive transitions in w_2 are t_4 and t_1 but they are concurrent transitions because in w_1 , t_1 occurs before t_4 , this verification is made in step 1.b.ii (based on proposition 2) and then the pair $[t_1,t_4]$ is added to $Concurrent(Q_2)$. Since there are not dependencies constraining the firing of t_2 before t_1 and in w_1 , t_1 occurs before t_2 , where t_1 and t_2 are the following consecutive transitions in w_2 , then they are not concurrent transitions however the NDep $[t_1, t_2]$ is not added to $Dep(Q_2)$ because there exists the MDep $[t_1, t_2]$ in $Dep(Q_2)$ (step 1.b.iii). The next computation concerns to the NDep $[t_2, t_5]$ since t_2 and t_5 are consecutive transitions in w_2 and they are not concurrent transitions; this procedure is specified in step 1.b.iii of the previous algorithm.

The computed model after evaluate w_2 is depicted on figure 7.c. Notice that this model describes the observed behavior of the system; however in order to compute a model for the system Q , it is necessary to compute from future measurements, a m-word in which t_2 occurs before t_3 for example the m-word $w_3=t_1t_2t_3t_4t_5$ should induce removing the wrong NDep $[t_3, t_2]$.

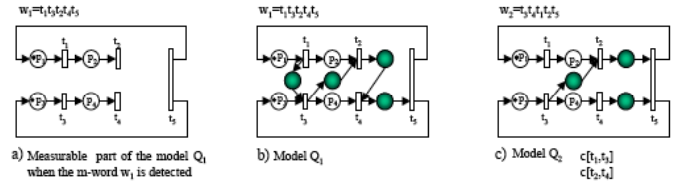


Figure 7: Iterations of the algorithm 3.

IV. CONCLUSIONS AND FUTURE WORK

In this work was presented an on-line identification approach. This identification approach consists in build a mathematical model just from the knowledge of the output symbols of the system. Once a new information of the system is detected a new model is computed. However the main objective of this work was to present a procedure to identify concurrent systems. The complexity of this procedure is how to detect concurrent transitions.

Once concurrent transitions are detected the wrong non-measurable places are removed. As future work is left the identification procedure using also the input signals of the system.

V. REFERENCES

- [1] L. Aguirre, A. Ramírez & O. Begovich. *Design of Asymptotic Observers for Discrete Event Systems*. Proceedings of the IASTED International Conference on Intelligent Systems and Control. Santa Barbara Ca., USA 1999.
- [2] D. Angluin. *Inductive Inference of Formal Languages from Positive Data*. Information and Control. Vol. 45, pp. 117-135. 1980.
- [3] P. Darondeau and L. Bernardinello, *Polynomial algorithms for the synthesis of bounded nets*. Proceedings of CAAP'95, Springer Verlag LNCS 915 (364-378), 1995.
- [4] Javier Esparza and Jörg Desel. *Free Choice Petri Nets*. Cambridge University Press 1995.
- [5] E. M. Gold. *Complexity of Automaton Identification from Given Data*. Information and Control, Vol. 37, pp. 302-320. 1978.

- [6] K. Hiraishi. *Construction of Safe Petri Nets by Presenting Firing Sequences*. Lectures Notes in Computer Sciences, 616, pp. 244-262. 1992.
- [7] M.E.Meda & A. Ramírez. *Identification in Discrete Event Systems*. IEEE-Systems Man and Cybernetics Conference, San Diego Ca., USA, pp. 747-745. 1998.
- [8] M.E. Meda-Campaña, A. Ramírez-Treviño and E. López-Mellado. *Dynamical local properties for estimation and control of discrete event systems modeled by interpreted Petri*. 2000 IEEE International Conference on Systems, Man and Cybernetics. Nashville TN, USA. October 2000.
- [9] M.E. Meda-Campaña, A. Ramírez-Treviño and E. López-Mellado. *Asymptotic Identification for DES*. IEEE International Conference on Decision and Control. Sydney, Australia. December 2000.
- [10] M.E. Meda-Campaña and E. López-Mellado. *A passive method for on-line identification of Discrete Event Systems*. IEEE International Conference on Decision and Control. Orlando Fl. USA. December 2001
- [11] M.E. Meda-Campaña and E. López-Mellado. *Incremental synthesis of Petri net models for identification of discrete event systems*. IEEE International Conference on Decision and Control. Las Vegas, NV. USA. December 2002.
- [12] M.E. Meda-Campaña and E. López-Mellado. *Required transition sequence to identify a DES*. IEEE International Conference on Decision and Control. Maui, Hawaii. December 2003.
- [13] T. Murata. *Synthesis of Decision-free Concurrent Systems for Prescribed Resources and Performance*. IEEE Trans. on Software Engineering, Vol. SE-6, No. 6, 525-530, 1980.
- [14] A. Ramírez-Treviño, I. Rivera-Rangel and E. López-Mellado. *Observer Design for Discrete Event Systems Modeled by Interpreted Petri Nets*. Proceedings of the IEEE International Conference on Robotics and Automation, 2000.
- [15] J. Peterson. *Petri net theory and the modeling of systems*. Prentice hall inc., Englewood Cliffs, New Jersey 07632
- [16] Manuel Silva. *Las Redes de Petri: en la Automática y la Informática*. Editorial AC. 1985

DIAGNÓSTICO DE FALLAS USANDO REDES DE PETRI INTERPRETADAS: APROXIMACIÓN ESTRUCTURAL *

Santoyo-Sanchez Alejandra¹, Ruiz-Beltrán Elvia², y De Jesús-Velasquez Carlos Alberto³

¹Departamento de Computación, División de Electrónica y Computación, CUCEI
Universidad de Guadalajara
Av. Revolución No. 1500 Modulo "O", Col. Olímpica, CP. 44430, Guadalajara, Jalisco, México
Tel. (33) 36398038, correo electrónico: alejandra.santoyo@cucei.udg.mx

²Departamento de Sistemas y Computación
Instituto Tecnológico de Aguascalientes
Av. A. López Mateos Ote. 1801, Fracc. Bona Gens, C.P 20256, Aguascalientes, Aguascalientes, México
Tel. (33) 36398038, correo electrónico: eruiz@ita.mx

³Compatibility Validation
Intel Guadalajara Design Center
Parque Industrial Tecnológico II, Periférico Sur 7980 Edificio 4E, Col. Santa Maria Tequepexpan, C.P. 45600,
Tlaquepaque, Jalisco, México
Tel. (33) 3540-6215, correo electrónico: carlos.de.jesus@intel.com

RESUMEN

En este trabajo se presenta una técnica de diagnóstico basado en Redes de Petri Interpretadas. Para detectar y ubicar la falla el comportamiento normal y anormal del sistema, como dos modelos en Redes de Petri Interpretadas denominadas modelo diagnosticador y de sistema respectivamente. Considerando ambos modelos se define una nueva Red de Petri Interpretada cuya finalidad es detectar cuando ocurren las fallas, llamada de error. Utilizando la estructura de esta nueva red se localiza el estado de falla, el componente de falla y el tipo de falla. Se incluyen algoritmos de implementación de diagnosticadores y se ilustra su funcionamiento con un ejemplo.

Palabras clave: Detección de fallas, diagnóstico y pronóstico, Estimación de estado, Redes de Petri, Red de Petri de error.

I. INTRODUCCIÓN

En general, las tareas de monitoreo, detección, ubicación y restablecimiento ante fallas en Sistemas de Eventos Discretos (SED) se han abordado a través de diversos enfoques y representaciones, entre las cuales se encuentran los autómatas y las Redes de Petri (RP). Aunque los

autómatas son adecuados para describir SED, su aplicación se limita a los sistemas de pequeño tamaño, ya que los modelos explícitamente deben tener en cuenta todos los estados posibles del sistema, lo que resulta en modelos bastante grandes.

El diagnóstico de fallas utilizando RP se ha analizado considerando varios métodos. En [14] se propone una técnica para la detección de fallos en línea mediante el monitoreo de los p-semiflujos, mientras que en [6] se propone la detección y aislamiento de fallas a través de sistemas redundantes; ambas caracterizaciones enfrentan el problema de la explosión de estados.

Para evitar este inconveniente, en [3] se propone una técnica de diagnóstico basada en una caracterización estructural sobre la propiedad de diagnosticabilidad, similar a la propiedad de observabilidad propuesta en [15], [1], [12], cuya caracterización estructural se sintetizó en [4] como un algoritmo polinomial bajo un esquema de detección y localización de fallas en línea. Sin embargo las caracterizaciones sobre diagnóstico sólo se han enfocado en detectar una falla a la vez.

Con la finalidad de solucionar este problema en

* Supported by project PROMEP/103.5/07/2636

este trabajo se implementa una técnica para llevar a cabo el diagnóstico de fallas en redes eléctricas basada en RPI's.

Este artículo está conformado por dos partes fundamentales. En la primera parte, se presentan la notación y conceptos utilizados para implementar el diagnosticador como una RPI. En la segunda se identifican y localizan los componentes del sistema en estado de fallo a través de la Red de Petri del error. Extiendiendo los resultados presentados en [2], [3] y [7]. Una vez detectada y localizada la falla se procede a emitir el diagnóstico, indicando la situación que llevo al estado de alerta, y cuales protecciones fueron activadas. Finalmente, se ilustra la detección, localización de fallas y diagnóstico considerando una pequeña red eléctrica del Sistema Eléctrico Nacional de México, correspondiente a algunas poblaciones del país.

II. CONCEPTOS BÁSICOS

Las RP son un formalismo para modelado y análisis de propiedades de SED. Un lector interesado en RP's puede consultar [13]. En este trabajo se usan las Redes de Petri Interpretadas (RPI) [16], las cuales son una extensión de las RP, debido a que permiten asociar señales de entrada y salida a los modelos en RP. Formalmente se define como sigue.

Definición 1: Una red de Petri Interpretada (RPI) es una 4-tupla $Q = (N', \Sigma, \lambda, \varphi)$ donde $N' = (N, M_0)$ es un sistema de RP, donde $N = (P, T, I, O)$ es la estructura de la Red de Petri y $M_0 : P \rightarrow Z^+$ es el marcado inicial; $\Sigma = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ es un alfabeto de entrada, donde α_i es un símbolo de entrada; $\lambda : T \rightarrow \Sigma \cup \{\varepsilon\}$ es una función de etiquetado, la cual asigna un símbolo de entrada a cada transición de la red, donde ε representa un evento interno del sistema; $\varphi : R(Q, M_0) \rightarrow (Z^+)^q$ es una función de salida, la cual asigna a cada marcado en $R(Q, M_0)$ un vector de salida en el sistema. En este caso q representa el número de salidas. Donde $R(N, M_0)$ representa el conjunto de alcanzabilidad de una red de Petri, es decir, el conjunto de todos los posibles

marcados que se pueden alcanzar desde M_0 disparando sólo transiciones habilitadas.

Cuando una transición t_j dispara en un marcado M_k , entonces un nuevo marcado M_{k+1} es alcanzado. Esto se calcula como:

$$\begin{aligned} M_{k+1} &= M_k + C\bar{\sigma}_k \\ y_k &= \varphi(M_k) \end{aligned} \quad \text{Ec. 1.}$$

donde C y $\bar{\sigma}_k$ son definidos como en una RP, y $y_k \in (Z^+)^q$ es la k -ésima salida de la RPI.

La función de etiquetado λ representa la asignación de las señales de los actuadores del sistema (el alfabeto de entrada Σ) al conjunto de transiciones de la red. Similarmente, el conjunto de vectores de salida $(Z^+)^q$ representa el estado de las señales de los sensores del sistema. En este contexto, es posible identificar dos tipos de transiciones: transiciones controlables

$(T_M = \{t_j \in T \mid \lambda(t_j) \neq \varepsilon\})$ e incontrolables

$(T_{NM} = \{t_j \in T \mid \lambda(t_j) = \varepsilon\})$, además de dos tipos de

lugares: medibles ($P_M = \{p_j \in P \mid \exists j \text{ tal que}$

$\varphi(i, j) \neq 0\}$) y no medibles ($P_{NM} = \{p_j \in P \mid \exists j \text{ tal que}$

$\varphi(i, j) = 0\}$). Para distinguirlos gráficamente, trabajo

los lugares medibles se representan mediante

círculos claros; mientras que los lugares no

medibles se representan con círculos oscuros.

Similarmente, las transiciones manipulables se

representan con barras claras, mientras que las

transiciones no manipulables se representan

mediante barras oscuras.

En una RPI una transición $t_j \in T$ está habilitada

en el marcado M_k si $\forall p_i \in P, M_k(p_i) \geq I(p_i, t_j)$; si

t_j es controlable, entonces es necesario que la

acción de entrada $\lambda(t_j) = \alpha_i \neq \varepsilon$ esté presente.

Cuando la transición t_j es incontrolable, entonces

cuando está habilitada, puede o no disparar.

Notas: Para enfatizar el hecho de que existe un

marcado inicial en una sistema en RPI, de ahora en

adelante se denotará como (Q, M_0) en lugar de

$Q = (N', \Sigma, \lambda, \varphi)$. Se considera el caso cuando la

función φ es la matriz identidad. En un Sistema en

Red de Petri, un autolazo es una relación donde $c_{i,j} = O(p_i, t_j) - I(p_i, t_j) = 0$, $O(p_i, t_j) \neq 0$, e $I(p_i, t_j) \neq 0$.

Una secuencia de disparo de una RPI (Q, M_0) es una secuencia de transiciones $\sigma = t_i t_j \dots t_k \dots$ tal que $M_j \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots M_w \xrightarrow{t_k} \dots M_k$. El tamaño de $|\sigma|$ de σ es el número de transiciones en σ . El prefijo de σ se define como $\bar{\sigma} = \{w' | \exists v \text{ tal que } w'v = \sigma\}$. El hecho de alcanzar M_k desde M_j por el disparo de la secuencia σ se denota por $M_j \xrightarrow{\sigma} M_k$. El lenguaje de disparos de (Q, M_0) es el conjunto $\mathcal{L}(Q, M_0) = \{\sigma | \sigma = t_i t_j \dots t_k \wedge M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots \xrightarrow{t_k} M_w\}$, mientras que el vector de Parikh $\bar{\sigma}: T \rightarrow (Z^+)^m$ de σ relaciona cada transición $t_x \in T$ al un número de ocurrencias de t_x en σ . $L_{in}(Q, M_0) = \{\lambda(t_i)\lambda(t_j)\dots\lambda(t_k) \text{ con } t_i t_j \dots t_k \in L(Q, M_0)\}$ es el lenguaje de entrada, y $L_{out}(Q, M_0) = \{\varphi(M_0)\varphi(M_1)\dots\varphi(M_w)\dots | M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots M_w \xrightarrow{t_k} \dots \text{ con } t_i t_j \dots t_k \in L(Q, M_0)\}$ es el lenguaje de salida.

Adicionalmente, dado un lenguaje L , el conjunto de infijos de L es $L_{mid} = \{w | \exists v, z \text{ tal que } vwz \in L \text{ donde } v \text{ y } z \text{ pueden ser la cadena vacía. En forma similar, el conjunto de prefijos de un lenguaje } L \text{ es } \bar{L} = \{z | z \in w \text{ and } w \in L\}$, donde L puede ser cualquiera de los lenguajes previos (lenguajes de entrada o salida).

En este trabajo se relacionan las secuencias de símbolos de entrada-salida con las secuencias de disparo de transiciones, y con las secuencias de marcados generadas. Estos conceptos son usados en el estudio de la propiedad de diagnosticabilidad.

Una secuencia de símbolos de entrada-salida de (Q, M_0) es una secuencia $w = (\alpha_0, \gamma_0) (\alpha_1, \gamma_1) \dots (\alpha_n, \gamma_n)$, donde $\alpha_j \in \Sigma \cup \{\varepsilon\}$ y α_{i+1} son las entradas actuales en la RPI y la salida a cambiado de y_i a y_{i+1} . Se asume que $\alpha_0 = \varepsilon$, $\gamma_0 = \varphi(M_0)$ cuando $(\alpha_{i+1}, \gamma_{i+1})$ pertenece a la secuencia entonces (α_i, γ_i) también pertenece a la secuencia, donde $y_{i+1} \neq y_i$ es el primer cambio a la salida debido a la

ocurrencia de y_i . El conjunto de todas las secuencias de símbolos de entrada-salida de (Q, M_0) es representado por $\Lambda(Q, M_0) = \{w | w \text{ es una secuencia de símbolos de entrada-salida}\}$, mientras que $\Lambda^k(Q, M_0) = \{w | w \in \Lambda(Q, M_0) \text{ y } |w| \geq k\}$ es el conjunto de secuencias de entrada-salida de tamaño mayor o igual a k , y $\Lambda_B(Q, M_0) = \{w | w \in \Lambda(Q, M_0) \text{ y } \exists \sigma \in \Omega(w) \text{ tal que } M_0 \xrightarrow{\sigma} M_j \text{ y } M_j \text{ no habilita transiciones o donde } M_j \xrightarrow{t_i} \text{ entonces } C(\bullet, i) = \bar{0}\}$. Donde $\Omega(w) = \{\sigma | \sigma \in L(Q, M_0) \text{ y el disparo de } \sigma \text{ produce } w\}$, es decir, $\Omega(w)$ es el conjunto de todas las posibles secuencias de disparo de transiciones que pueden generar la palabra w .

Otros conceptos utilizados son los siguientes. El conjunto de predecesores $\bullet x = \{y | \exists \text{ un arco de } y \text{ a } x\}$; el conjunto de sucesores $x \bullet = \{y | \exists \text{ un arco de } x \text{ a } y\}$ donde $x, y \in T \cup P$. Un p-invariante Y de una RPI es una solución real de la ecuación $Y^T C = \bar{0}$. El soporte de un p-invariante Y_i es el conjunto $\|Y_i\| = \{p_j | Y_i(p_j) \neq 0\}$. Similarmente, un t-invariante X de una RPI es una solución real de la ecuación $CX = \bar{0}$. El soporte de un t-invariante X_i es el conjunto $\|X_i\| = \{t_j | X_i(t_j) \neq 0\}$.

III. DIAGNOSTICABILIDAD USANDO RPI

La diagnosticabilidad es la propiedad de detectar y localizar la ocurrencia de falla en el sistema a través de la información disponible como las entradas al sistema, salidas y su estructura. Esta sección presenta una revisión sobre la diagnosticabilidad y una caracterización para RPI diagnosticable. Para profundizar en estos temas consulte [3] y [4].

Definición 2: Una RPI (Q, M_0) es entrada-salida diagnosticable en $k < \infty$ pasos, si para cualquier marcado $M_f \in F$ este se puede distinguir de cualquier otro $M_k \in R(Q, M_0)$ usando cualquier palabra $w \in \Lambda^k(Q, M_f) \cup \Lambda_B(Q, M_f)$.

Note que la definición 2 implica que después de la ocurrencia de una palabra de entrada-salida con tamaño k , es posible distinguir cualquier marcado

de falla. Entonces para establecer la diagnosticabilidad en una RPI es necesario detectar la ocurrencia de cada evento del sistema, es decir, el disparo de cada transición desde M_0 a M_f , definida como sigue.

Definición 3: Una RPI (Q, M_0) es evento detectable si el disparo de cualquier transición $t \in T$ en un marcado $M_k \in R(Q, M_0)$ puede ser determinado únicamente a través de la información de los símbolos de entrada $\alpha = \lambda(t)$ y las señales de salida $\varphi(M_k)$, donde $\varphi C(\bullet, k)$ es la columna de φC correspondiente a la transición t_k .

Lema 1: Una RPI (Q, M_0) es evento detectable si y solo si $\forall i \in [1, 2, 3, \dots, m], \varphi C(\bullet, i) \neq \bar{0}$ y $\forall j \neq k \in [1, 2, 3, \dots, m]$ tal que $\varphi C(\bullet, j) = \varphi C(\bullet, k), \lambda(t_j) \neq \lambda(t_k)$ donde $\varphi C(\bullet, k)$ es la columna de φC correspondiente a la transición t_k .

Demostración. Se encuentra en [1]. ■

Note que esta caracterización estructural puede probarse en tiempo polinomial. Una vez que se ha determinado la ocurrencia de la secuencia de entrada-salida es necesario caracterizar la diagnosticabilidad en RPI. En este caso se usa el teorema 1, propuesto en [3], el cuál se basa en la idea de que el modelo del diagnosticador (Q^N, M_0^N) está inmersa en la RPI del sistema (Q^D, M_0^D) (vea la figura 3). Así todas las secuencias de entrada que corresponden al comportamiento normal son aceptadas por ambos modelos, mientras que las de falla sólo aparecen en la RPI del sistema como transiciones no controlables $t_k \in -T^F$, marcando un lugar de falla. Entonces una RPI es diagnosticable si es posible conocer cuando un lugar de falla es marcado y determinar cuál es el lugar marcado.

Teorema 1: Sea (Q^D, M_0^D) una RPI binaria, viva, fuertemente conectada y evento detectable. Sea $\{X_1, \dots, X_r\}$ el conjunto de todos los T-psemiflujos de (Q^D, M_0^D) . Si $\forall p_i^N \in P^N, (p_i^N) \bullet \cap T^F \neq \{\}$ se mantienen las siguientes condiciones:

1. $\forall r, \exists^j X_r(j) \geq 1$, donde $t_j \in (p_i^N) - T^F$;
2. $\forall t_k \in (p_i^N) - T^F, \bullet(t_k) = \{(p_i^N)\}$ y $\lambda(t_k) \neq \varepsilon$.

Entonces la RPI (Q^D, M_0^D) es diagnosticable entrada-salida.

Demostración. Puede encontrar la demostración en [3]. ■

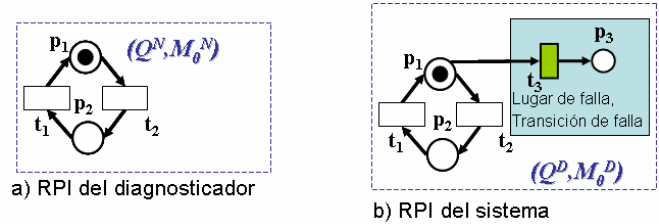


Fig. 3. Par modelo sistema-diagnosticador
 (Q^D, Q^N) .

IV. DISEÑO DEL DIAGNOSTICADOR

La clave con la detección y localización de fallas consiste en identificar el comportamiento anormal en los sistemas, y localizar la causa o los recursos que están trabajando mal, donde un comportamiento anormal no forma parte de la definición de comportamiento del sistema.

IV.1. ESQUEMA DEL DIAGNOSTICADOR

La Figura 4 ilustra la estructura general y conceptual del diagnosticador de fallas del sistema; el cuál esta compuesto por seis elementos:

1. Modelo del diagnosticador (Q^N, M_0^N) , una RPI que representa el comportamiento normal del sistema.
2. Modelo del sistema (Q^D, M_0^D) , una RPI que contiene el comportamiento normal y anormal del sistema, así el modelo del diagnosticador esta inmerso en la RPI del sistema.
3. Bloque detector de disparos del sistema, detecta y determina cuales transiciones han sido disparadas en el sistema.
4. Modelo de Error, es una RP que compara el comportamiento entre (Q^D, M_0^D) y (Q^N, M_0^N) , la cuál se detalla en la definición 2.
5. Algoritmo de detección de fallas, detecta y localiza la falla a través del modelo de error en RPI, además de indicar el estado de falla.
6. Algoritmo de Diagnóstico, indica cuál componente fallo y el tipo de falla que presentó.

IV.2. MODELO DEL ERROR

Definición 4: [5] La RPI de Error entre (Q_1, M_0^1) y (Q_2, M_0^2) se define como:

$$E(Q_1, Q_2) = \begin{cases} E_{k+1} = E_k + [C^1 - C^2] \omega_k \\ \mu_k = y_k^1 - y_k^2 \end{cases} \quad \text{Ec. 2.}$$

donde $|P^1| = |P^2|$, E_{k+1} es el marcado de error, ω_k es el vector de disparo y μ_k es la salida definida como:

$$E_{k+1} = M_{k+1}^1 - M_{k+1}^2 \quad \text{Ec. 3.}$$

$$\omega_k = \begin{bmatrix} \bar{\sigma}_k^1 \\ \bar{\sigma}_k^2 \end{bmatrix} \quad \text{Ec. 4.}$$

$$\mu_k = y_k^1 - y_k^2 = \varphi \bullet M_k^1 - \varphi \bullet M_k^2 \quad \text{Ec. 5.}$$

La figura 5c muestra la estructura de la RPI de error entre las RPI's de las figuras 5a y 5b, note que la RPI de error definida en la ecuación 2 representa otra RPI donde los lugares son los de (Q_1, M_0^1) o (Q_2, M_0^2) y las transiciones son la unión de las transiciones de (Q_1, M_0^1) y (Q_2, M_0^2) , donde su matriz de incidencia es definida por la matriz a bloque $[C^1 - C^2]$ por lo que en la RPI de error las relaciones de (Q_2, M_0^2) están invertidas. El marcado, sin embrago puede ser negativo ya que es definido como la diferencia de dos marcados independientes. Cuando la salida de la RPI de error

es $\bar{0}$ significa que no existe diferencia entre las salidas de (Q_1, M_0^1) y (Q_2, M_0^2) , en otro caso la información de la RPI de error puede usarse para determinar dónde existe la diferencia entre (Q_1, M_0^1) y (Q_2, M_0^2) .

El siguiente teorema caracteriza el diagnóstico basado en el modelo de RPI de error.

IV.3. DETECCIÓN DE FALLAS

Teorema 2. Sea (Q^D, Q^N) un par sistema-diagnosticador representado con las ecuaciones de estado 6 y 7 respectivamente. Donde el modelo del sistema es una copia del modelo diagnosticador al que se le han agregado en forma de trampas las fallas de cada elemento del sistema (véase la Figura 3), y cada falla se considera no manipulable.

$$Q^D = \begin{cases} M_{k+1}^D & = M_k^D + C^D \bar{\sigma}_k^D \\ y_k^D & = \varphi(M_k^D) \end{cases} \quad \text{Ec. 6.}$$

$$Q^N = \begin{cases} M_{k+1}^N & = M_k^N + C^N \bar{\sigma}_k^N \\ y_k^N & = \varphi(M_k^N) \end{cases} \quad \text{Ec. 7.}$$

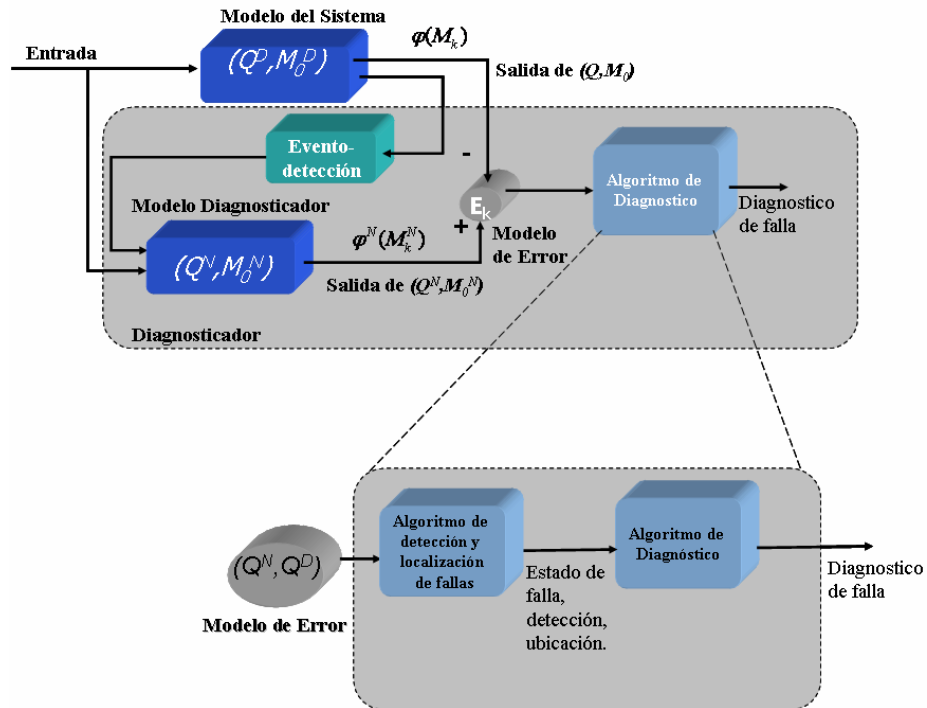
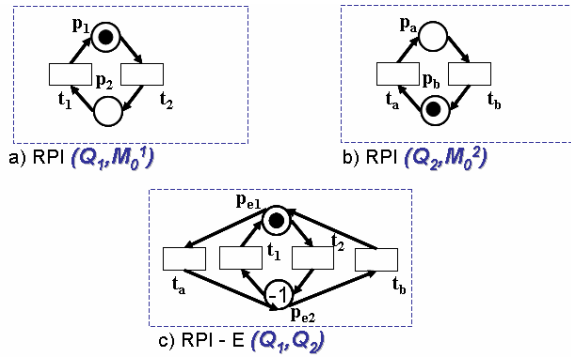


Figura 4. Esquema del diagnosticador.


Figura 5. Representación de la RPI de error.

Si (Q^D, Q^N) es diagnosticable entrada-salida [3], y la ecuación de estado de la RPI-E tiene la forma:

$$E(Q^N, Q^D) = \begin{cases} E_{k+1} = E_k + \begin{bmatrix} C^N & -C^D \\ \bar{0} & -C^D \end{bmatrix} \omega_k \\ \mu_k = y_k^N - y_k^D \end{cases} \quad \text{Ec. 8.}$$

donde E_{k+1} , el vector de disparo ω_k y la salida μ_k se definen en las ecuaciones 9, 10 y 11 respectivamente.

$$E_{k+1} = \begin{bmatrix} M_{k+1}^N \\ \bar{0} \end{bmatrix} - M_{k+1}^D \quad \text{Ec. 9.}$$

$$E_{k+1} = \begin{bmatrix} M_{k+1}^N \\ \bar{0} \end{bmatrix} - \begin{bmatrix} M_{k+1}^N \\ M_{k+1}^F \end{bmatrix}$$

$$\omega_k = \begin{bmatrix} \bar{\sigma}_k^N \\ \bar{\sigma}_k^D \end{bmatrix} \quad \text{Ec. 10.}$$

$$\mu_k = y_k^N - y_k^D = \varphi \bullet M_k^N - \varphi \bullet M_k^D \quad \text{Ec. 11.}$$

Entonces se puede detectar y localizar la falla del sistema.

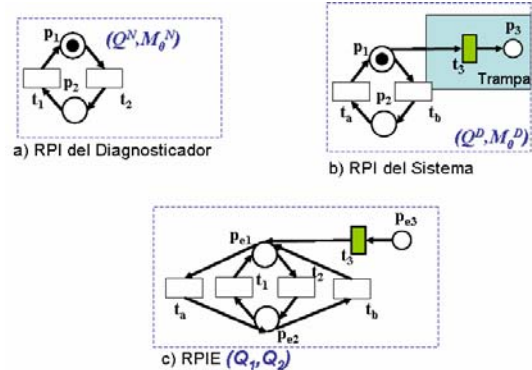
Demostración. Por hipótesis el sistema (Q^D, Q^N) es diagnosticable entrada-salida, entonces el disparo de cualquier transición puede ser detectada por el cambio entre las salidas que ocurran en el sistema. Existen dos casos posibles.

Caso A. Cuando una secuencia de transiciones σ_k es disparada en (Q^D, M_0^D) y no contiene transiciones de fallo, entonces la diferencia de marcado E_k obtenida de la RPI-E $E(Q^N, Q^D)$ a

partir de la ecuación 8 es

$$E_k = \begin{bmatrix} M_k^N \\ \bar{0} \end{bmatrix} - \begin{bmatrix} M_k^N \\ M_k^F \end{bmatrix} = \begin{bmatrix} M_k^N \\ \bar{0} \end{bmatrix} - \begin{bmatrix} M_k^N \\ \bar{0} \end{bmatrix} = \bar{0} \quad \text{Ec. 12.}$$

es decir, cero como se muestra en la Figura 6c.


Fig. 6. RPI-E del par sistema-diagnosticador cuando no hay fallas.

Caso B. Cuando una secuencia de transiciones σ_k se dispara en (Q^D, M_0^D) y contiene al menos una transición de fallo $t_i \in T^F$, el cambio de estado por la ocurrencia de t_i sólo ocurrirá en el modelo del sistema, provocando que la marca del lugar normal $p_r \in \bullet(t_i)$, se coloque en el lugar de falla $p_j \in (t_i) \bullet$, ya que el diagnosticador sólo contiene el modelo del sistema sin fallas, por lo que se mantendría en el marcado anterior M_{k-1}^N . Entonces la diferencia de marcado E_k obtenida de la RPI-E $E(Q^N, Q^D)$ a partir de la ecuación 9 es:

$$E_k = \begin{bmatrix} M_{k-1}^N \\ \bar{0} \end{bmatrix} - \begin{bmatrix} M_k^N \\ M_k^F \end{bmatrix} \neq \bar{0} \quad \text{Ec. 14.}$$

obteniéndose el error ilustrado en la Figura 7c.

Entonces cuando la diferencia de marcados es diferente de cero, a través de la RPI-E es posible detectar que ha ocurrido una falla, los lugares de fallo se pueden obtener como $p_F = \{\forall p_i \mid E_{k+1}(p_i) = -1\}$. Mientras que los eventos de fallo se capturan en la RPI-E como los sucesores de los lugares de fallo, es decir: $t_F = \{\forall t_k \mid (p_F) \bullet\}$. La localización del componente

de falla se puede obtener como $Comp_F = \{\forall c \mid p_i \in (t_F)^\bullet, p_i \text{ es un lugar utilizado para representar el componente de } c, \text{ y } c \text{ es el nombre del componente del sistema en falla.}\}$

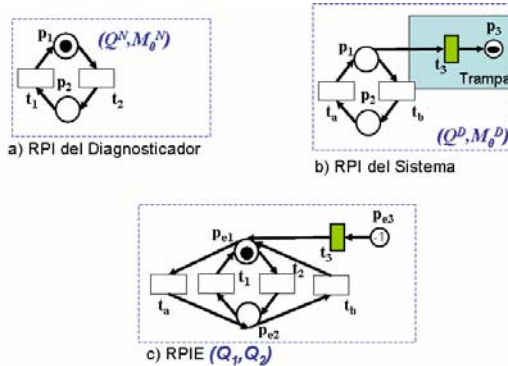


Fig. 7. RPI-E del par sistema-diagnosticador cuando hay fallas.

Por otra parte la RPI del sistema contiene la estructura de la RPI del diagnosticador, entonces toda secuencia de entrada correspondiente al comportamiento normal aceptada por diagnosticador será aceptada por el sistema. Esta propiedad es aprovechada para simplificar la representación gráfica de la RPI-E como se muestra en la figura 8, esta forma de representación será utilizada en el resto de este artículo. Note que la RPI-E simplificada contiene sólo los lugares de la RPI-E original, las transiciones de fallas y sus relaciones.

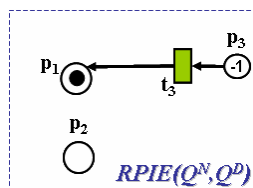


Fig. 8. RPI-E simplificada del par sistema-diagnosticador cuando hay fallas.

Utilizando el teorema 2 se propone el siguiente algoritmo para detectar y localizar el marcado de error.

Algoritmo 1: Detectar y localizar el marcado de error.

1. Defina la RPI-E como en las ecuaciones de estado 8, 9, 10 y 11.
2. Cuando $E_k \neq \emptyset$ entonces:
 - 2.1 Los lugares de falla son $p_F = \{\forall p_i \mid E_{k+1}(p_i) = -1\}$
 - 2.2 $\forall p_i \in p_F$ las transiciones de falla son $t_F = \{\forall t_k \mid (p_F)^\bullet\}$
 - 2.3 $Comp_F = \{\forall c \mid p_i \in (t_F)^\bullet, \text{ y } p_i \text{ es el lugar usado para representar el componente } c \text{ del sistema}\}$.
3. Retorne E_{k+1}, p_F, t_F y $Comp_F$.

El algoritmo 1 se basa en la idea de que si el error E_{k+1} es diferente de cero, entonces el comportamiento del diagnosticador no es consistente con el modelo del sistema, lo que implica que ha ocurrido un fallo. En seguida, se procede a identificar y clasificar el conjunto de posibles fallos en el algoritmo 2, lo que posteriormente permitirá establecer el diagnóstico de la falla. En este caso este algoritmo está especificado para hacer diagnóstico en redes eléctricas.

Algoritmo 2: Algoritmo de diagnóstico.

1. Defina el conjunto de fallas como $(Comp_F, \text{tipo_falla})$.
2. Determine el comportamiento de las protecciones.
 - 2.1 $ProteccD_{Comp_F} = \{\forall c \mid p_i \in M(p_i) = 1, \text{ donde } p_i \text{ es el lugar usado para representar la desconexión del componente } c \text{ por la falla en } Comp_F\}$.
 - 2.2 $ProteccI_{Comp_F} = \{\forall c \mid p_i \in M(p_i) = > 1, \text{ donde } p_i \text{ es el lugar usado para describir cómo se distribuye la energía eléctrica en la red, y } c \text{ indica el componente del sistema que está trabajando con estrés debido a la falla en } Comp_F\}$.
3. Diagnostique. Retorne los conjuntos de falla, $ProteccD_{Comp_F}$ y $ProteccI_{Comp_F}$.

V. CASO DE ESTUDIO

Considere el modelo del Sistema Eléctrico correspondiente a la región centro occidente de México, que incluye los estados de Jalisco, Michoacán y Guanajuato, el cual se muestra en la Figura 9. Note que la Red de estudio consta de dos generadores (*GU*) interconectados mediante líneas de transmisión (*LU_i*) a las cargas (*CU*), sensores, capacitores, interruptores, una subestación y transformadores (*TU_i*). Además de que los medidores se encuentran ubicados en puntos estratégicos para mejor control del sistema eléctrico.

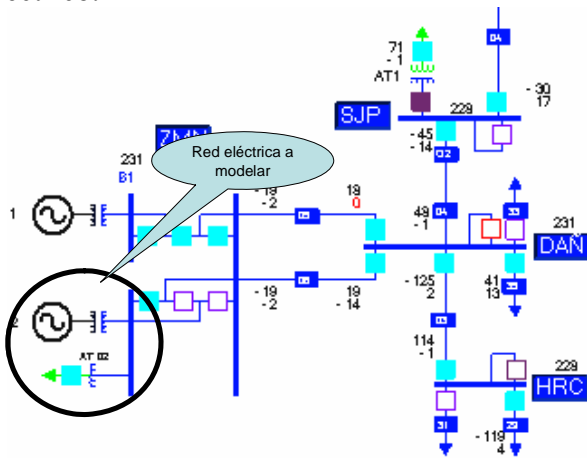


Fig. 9. Vista de la Red eléctrica del Sistema a diagnosticar.

Debido a la complejidad del modelo en RPI, en este artículo se ejemplifica el estudio de diagnosticabilidad a través del modelo en RPI del Sistema Eléctrico contenido dentro del círculo en la Figura 9, del cuál se obtuvo el modelo en red de Petri para el diagnosticador (Q^N, M_0^N) de la Figura 10-a.

Las funciones λ y ϕ son las siguientes:

1. $\lambda(t_1) = \lambda(t_3) = \lambda(t_5) = \lambda(t_7) = \lambda(t_9) = \lambda(t_{11}) = \lambda(t_{13}) = \lambda(t_{15}) = \lambda(t_{17}) = ON_{componente}$,
 $\lambda(t_2) = \lambda(t_4) = \lambda(t_6) = \lambda(t_8) = \lambda(t_{10}) = \lambda(t_{12}) = \lambda(t_{14}) = \lambda(t_{16}) = \lambda(t_{18}) = OFF_{componente}$,
 $\lambda(t_{19}) = \lambda(t_{20}) = \lambda(t_{21}) = \lambda(t_{22}) = \lambda(t_{23}) = \varepsilon$.

Por lo que todas las transiciones en el modelo del componente “DU” son no manipulables, y se asume que su disparo es instantáneo.

2. ϕ = Matriz Identidad, de 22×22 .

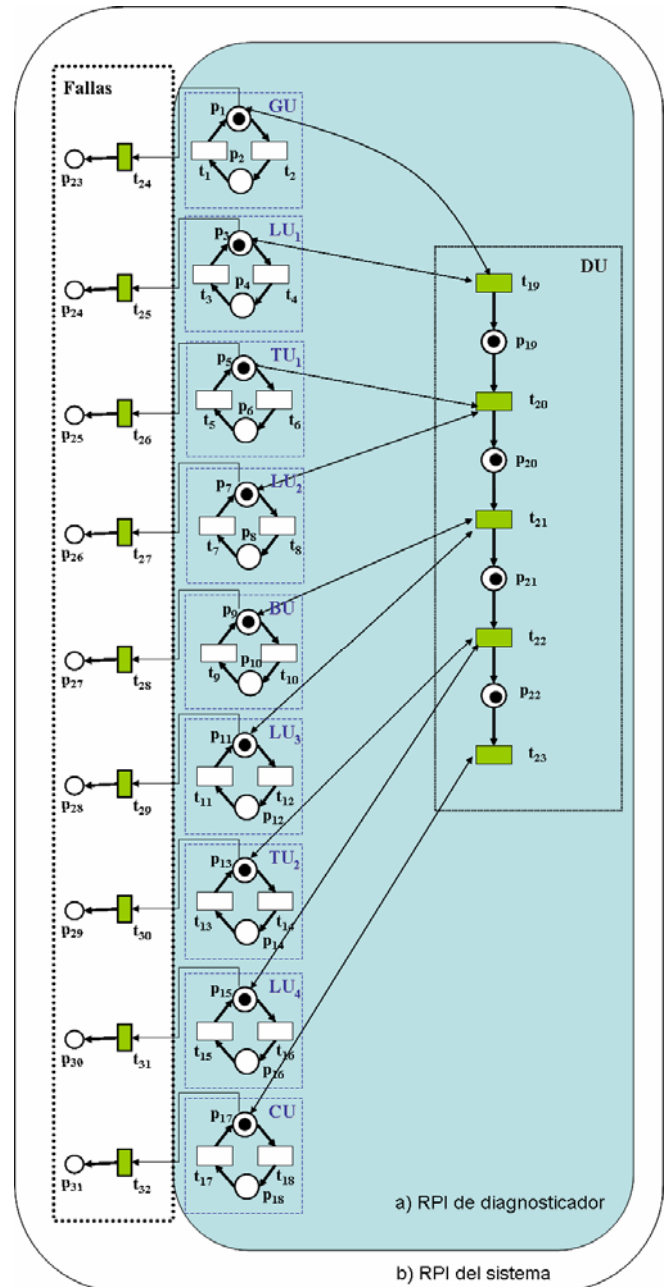


Fig. 10. Modelos en RPI del par diagnosticador y sistema para el Sistema Eléctrico.

Una vez elaborado el modelo del comportamiento normal del sistema (diagnosticador (Q^N, M_0^N)), se modela el comportamiento anormal del sistema y se relaciona con el comportamiento normal del mismo. Para el caso de estudio, los fallos considerados sólo están en relación con el voltaje, y se muestran en la tabla 1.

f_1 : Falla el generador por diferencia de voltaje
f_2 : Falla el transformador por diferencia de voltaje
f_3 : Falla en línea por diferencia de voltaje
f_4 : Falla en bus por diferencia de voltaje

Tabla 1. Fallos considerados para el caso de estudio.

Al considerar los fallos de la Tabla 1, el modelo del sistema para la zona uno se transforma en el modelo en RPI (Q^D, M_0^D) mostrado en la Figura 10-b, en donde los fallos del sistema son modelados como una variable de estado del componente que puede fallar. Para el modelo de la figura 10-b las fallas que representa cada transición se describen en la tabla 2.

t_{24} : Diferencia de voltaje en generador
t_{25}, t_{27} : Diferencia de voltaje en la línea de transmisión 1 y 2
t_{26} : Diferencia de voltaje en transformador 1
t_{28} : Diferencia de voltaje en bus
t_{29}, t_{31} : Diferencia de voltaje en línea de transmisión 3 y 4
t_{30} : Diferencia de voltaje en transformador 2
t_{32} : Diferencia de voltaje en carga (consumidor)

Tabla 2. Eventos de fallas considerados para el caso de estudio.

De acuerdo con [3] el modelo (Q^D, M_0^D), es evento-detectable, es decir, se pueden distinguir el disparo de cualquier par de transiciones. Una vez que se modelan el diagnosticador, y el sistema como RPI, se obtiene la RPIE simplificada mostrada en la Figura 11.

FUNCIONAMIENTO DEL DIAGNOSTICADOR.

Suponga que en el instante k la diferencia de marcado E_k obtenida de la RPI-E $E(Q^N, Q^D)$ es:

$$[E_k]^T = [00000000000100000000000000000000-1000]$$

en forma de lista $E_k = \{l_{11}, -l_{28}\}$. Por lo que el algoritmo de detección y localización de marcado (algoritmo 1) retorna lo siguiente:

1. Los lugares de fallo sólo contienen a $p_F = \{p_{28}\}$ ya que es el único lugar donde $E_k(p_{28}) = -1$.
2. Los eventos de fallo sólo incluyen a $t_F = \{t_{29}\}$ debido a que es el único sucesor del lugar de fallo p_{28} .
3. El componente de fallo como $Comp_F = \{LU_3\}$, es decir la línea de transmisión 3, porque p_{11} es un lugar utilizado para representar el comportamiento del componente de LU_3 .

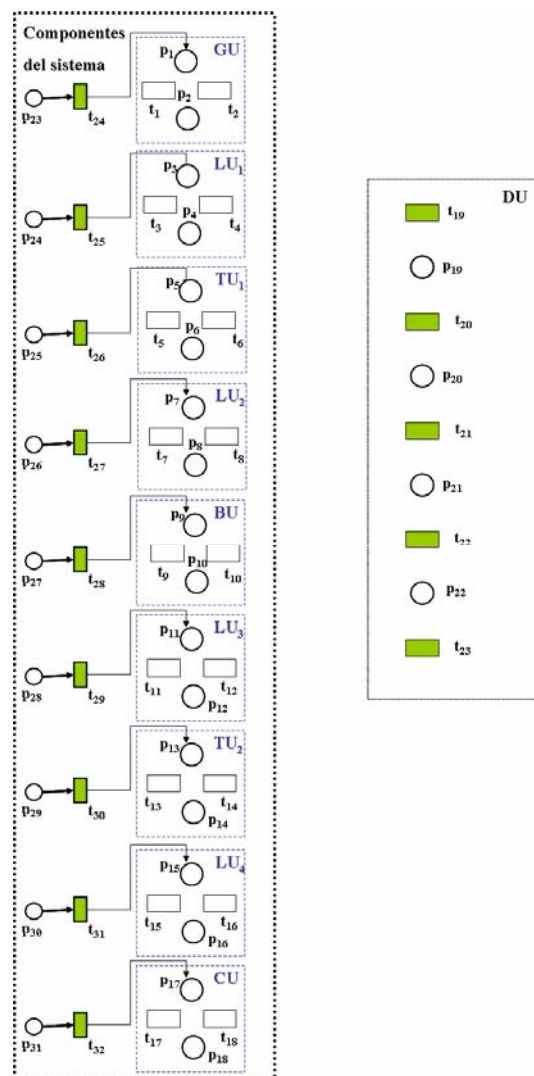


Fig. 11. RPI-E simplificada para el caso de estudio.

En seguida, se ejecuta el algoritmo de diagnóstico (algoritmo 2), emite lo siguiente.

1. Origen del disturbio;

Fecha de la falla	Día, hora
Nombre del elemento afectado	$Comp_F = \{LU_3\}$, es decir la línea de transmisión 3 [3]
Tipo de falla	f3: Falla en línea por diferencia de voltaje

2. Diagnóstico del comportamiento de protecciones. [4]

- a. Directo: $Protecc_{Comp_F} = \{\}$; aún no se ha desconectado el componente.
- b. Indirecto: $Protecc_{Comp_F} = \{BU\}$. En este caso, el fallo en la línea de transmisión 3 provoca que la transición t_{21} no pueda disparar, el conjunto de predecesores de $t_{21} = \{p_9, p_{11}, p_{20}\}$, en donde p_9 representa el buen funcionamiento del Bus (BU), p_{11} representa la falla de (LU3), y p_{20} el lugar de DU donde se acumulan marcas, es decir, en donde se esta acumulando la carga.

VI. CONCLUSIONES

En este trabajo se presentó una técnica de diagnóstico basado en Redes de Petri Interpretadas. Se basa en detectar y ubicar la falla a través del uso de la RPI de error formada a partir de los modelos diagnosticador y del sistema. Utilizando la estructura de esta nueva red se proponen algoritmos para localizar el estado de falla, el componente de falla y el tipo de falla. Actualmente, se investiga sobre las condiciones necesarias para la diagnosticabilidad cuando se considera más de un fallo por componente.

VII. REFERENCIAS

[1] A. Ramírez-Treviño, I. Rivera-Rangel, and E. López-Mellado, "Observability of discrete event systems modeled by interpreted petri nets", IEEE Trans. Robot. Autom., Vol. 19, No. 4, pp. 557-565, 2003.

[2] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel, and E. López-Mellado,

"Diagnosability of Discrete Event Systems. A Petri Net Based Approach", Proceedings of the IEEE International Conference on Robotic and Automation, pp. 541-546, 2004.

A. Ramírez-Treviño., E. Ruiz-Beltrán., I. Rivera-Rangel, and E. López-Mellado, "On-line Fault Diagnostic of Discrete Event Systems. A Petri Net Based Approach", IEEE Transactions on Automation Science and Engineering, Vol. 4, No. 1, pp. 31-39, 2007.

[4] A. Santoyo, I. Jiménez-Ochoa, and A. Ramírez-Treviño, "A complete cycle for controller design in Discrete Event System", IEEE Proceedings of System Man and Cybernetics, pp. 2688-2693, 2001.

[5] C. A. De Jesús, and A. Ramírez-Treviño, "Controller and Observer Synthesis in Discrete Event Systems Using Stability Concepts", Proceeding of the IEEE System Man and Cybernetics, pp. 664-668, 2001.

[6] C. N. Hadjicostis, and G. Verghese, "Structured Redundancy for Fault Tolerance in State Space models and Petri Nets", KYBERNETICA, Vol.35, No.1,pp.39-55, 1999.

[7] E. Ruiz-Beltrán, A. Ramírez-Treviño, E. López-Mellado, and M. Arámburo-Lizárraga, "A structural characterization of diagnosticable Petri net models", IEEE Conference on Automation Science and Engineering Scottsdale, pp. 1137-1142, 2007.

[12] I. Rivera-Rangel., A. Ramírez-Treviño, L.I. Aguirre-Salas, and J. Ruiz-León, "Geometrical characterization of Observability in Interpreted Petri Nets", KYBERNETIKA, Vol. 41, pp. 553-574, 2005.

[13] J. Desel, J. Esparza and C. J. van Rijsbergen, Free choice Petri nets, Cambridge University Press, 2005.

[14] J. Prock, "A New Technique for Fault Detection Using Petri Nets", Automatica, Vol. 27, pp. 239-245, 1991.

[15] L. Aguirre-Salas, O. Begovich, and A. Ramírez-Treviño, "Observability in interpreted petri nets using sequence invariants", Proceeding of the IEEE Control Decision Conference, pp. 3602-3607, 2002.

[16] M. E. Meda, and A. Ramírez, "Identification in discrete event systems", IEEE International Conference Systems, Man and Cybernetics, pp. 740-745, 1998.

Methodologies for obtaining reduced Petri net models to detect and locate faults

E. Ruiz-Beltrán*, A. Ramírez-Treviño**, E. López-Mellado**, and J. L. Orozco M*.

*Instituto Tecnológico de Aguascalientes, Aguascalientes, Aguascalientes., México.

** CINESTAV-IPN, Guadalajara, Jalisco., México

e-mail: er Ruiz@ita.mx, art@gdl.cinvestav.mx, drorozco@ita.mx

Abstract — This paper deals with fault detection and location of Discrete Event Systems (DES) modeled using Interpreted Petri nets (IPN). In this paper two efficient methods for obtaining reduced diagnosers IPN models are proposed: based on the DES model, a Petri net is synthesized depending on each methodology; each net consists of a single place (or more) and the same number of transitions that the system model has; the current marking of this place (places) is enough to determine and locate faults occurring within the DES.

Keywords — Discrete Event Systems, Interpreted Petri Nets, Fault detection and location.

I. INTRODUCTION

The analysis of the diagnosability property of a DES has been addressed through several approaches and methods, namely that based on artificial intelligence techniques and that based on discrete event models. Recently, finite automata (FA) and Petri nets (PN) have been widely used as modeling formalisms and formal tools for fault diagnosis.

In [1] M. Sampath et al. introduced the notion of diagnosability and proposed a method for designing an on-line diagnoser for determining the diagnosability property.

In [4], [5] and [7] Prock, Genc and Hadjicostis respectively proposed methods based on PN models to detect and isolate the faults presented in the system. In [4] the tokens residing in P-semiflows are monitored and, depending on the quantity of these tokens, faults into the system are

determined. In [5] the held approach analyzes the reachability graph to isolate faults, leading to NP-complete algorithms. The strategy presented in [7] consists in adding one more place to the system model, the marking of this place can be used to determine and isolate system faults; in this approach however, it is assumed that all places are measurable, that all transitions are controllable, and that system and observer can be synchronized each other. These conditions are hardly fulfilled in current systems.

Based on [7] and the structural characterization of diagnosability [9], we propose a novel diagnosis scheme that use simple and reduced IPN models for monitoring the DES behavior. The monitoring models play a similar role as used in [7], however they can operate using only partial information on the marking (measurable places), and their synchronization with the DES model are not longer needed.

This work is organized as follows: Section II provides basic definitions of IPN. In section III is presented a procedure to construct reduced an IPN model. Section IV presents the two methodologies in order to structure the diagnosis scheme and its operation for detecting and locating failures. Section V includes an example for illustrating the notions herein introduced. Finally conclusions are given.

II. BACKGROUND

This section presents the basic concepts and notation of *PN* and *IPN* used in this paper.

Definition 1: A Petri Net structure G is a bipartite digraph represented by the 4-tuple $G=(P,T,I,O)$ where:

- $P = \{p_1, p_2, \dots, p_n\}$ and $T = \{t_1, t_2, \dots, t_m\}$ are finite sets of vertices named places and transitions respectively,
- $I(O) : P \times T \rightarrow Z^+$ is a function representing the weighted arcs going from places to transitions (transitions to places); Z^+ is the set of nonnegative integers.

Pictorially, places are represented by circles, transitions are represented by rectangles, and arcs are depicted as arrows. The symbol $\bullet t_j (t_j \bullet)$ denotes the set of all places p_i such that $I(p_i, t_j) \neq 0$ ($O(p_i, t_j) \neq 0$). Analogously, $\bullet p_i (p_i \bullet)$ denotes the set of all transitions t_j such that $O(p_i, t_j) \neq 0$ ($I(p_i, t_j) \neq 0$).

The pre-incidence matrix of G is $C^- = [c_{ij}^-]$ where $c_{ij}^- = I(p_i, t_j)$; the post-incidence matrix of G is $C^+ = [c_{ij}^+]$, where $c_{ij}^+ = O(p_i, t_j)$; the incidence matrix of G is $C = C^+ - C^-$.

A marking function $M : P \rightarrow Z^+$ represents the number of tokens (depicted as dots) residing inside each place. The marking of a PN is usually expressed as an n -entry vector.

Definition 2: A Petri Net system or Petri Net (PN) is the pair $N=(G, M_0)$, where G is a PN structure and M_0 is an initial token distribution.

In a PN system, a transition t_j is enabled at marking M_k if $\forall p_i \in P, M_k(p_i) \geq I(p_i, t_j)$; an enabled transition t_j can be fired reaching a new marking M_{k+1} which can be computed as $M_{k+1} = M_k + Cv_k$, where $v_k(i)=0, i \neq j, v_k(j)=1$, this equation is called the PN state equation. The reachability set of a PN is the set of all possible reachable marking from M_0 firing only enabled transitions; this set is denoted by $R(G, M_0)$.

This work uses Interpreted Petri Nets (IPN) [8] an extension to PN that allow associating input and output signals to PN models.

Definition 3: An $IPN (Q, M_0)$ is an IPN Interpreted Petri Net structure $Q=(G, \Sigma, \lambda, \varphi)$ with an initial marking M_0 .

- G is a PN structure
- $\Sigma = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ is the alphabet of input symbols α_i .
- $\lambda : T \rightarrow \Sigma \cup \{\varepsilon\}$ is a labeling function of transitions with the following constraint: $\forall t_j, t_k \in T, j \neq k$, if $\forall p_i I(p_i, t_j) = I(p_i, t_k) \neq 0$ and both $\lambda(t_j) \neq \varepsilon, \lambda(t_k) \neq \varepsilon$, then $\lambda(t_j) \neq \lambda(t_k)$; ε represents a system internal event.
- $\varphi : R(Q, M_0) \rightarrow (Z^+)^q$ is an output function, that associates to each marking in $R(Q, M_0)$ q -entry output vector; q is the number of outputs.

In this work φ is a $q \times n$ matrix. Each column of φ is an elementary or null vector. If the output symbol i is present (turned on) every time that $M(p_j) \geq 1$, then $\varphi(i, j) = 1$, otherwise $\varphi(i, j) = 0$.

A transition $t_j \in T$ of an IPN fires it is enabled at marking M_k , and a) If $\lambda(t_j) = \alpha_i \neq \varepsilon$ is provided to the system, or b) If $\lambda(t_j) = \varepsilon$ and t_j is enabled then t_j can be fired. When an enabled transition t_j is fired in a marking M_k , then a new marking M_{k+1} is reached. This fact is represented as $M_k \xrightarrow{t_j} M_{k+1}$; M_{k+1} can be computed using the state equation:

$$M_{k+1} = M_k + \begin{matrix} y_k \\ Cv_k \end{matrix} = \varphi(M_k) \quad (1)$$

where C and v_k are defined as in PN and $y_k \in (Z^+)^q$ is the k -th output vector of the IPN .

According to functions λ and φ , transitions and places of an $IPN (Q, M_0)$ can be classified as follows.

Definition 4: If $\lambda(t_i) \neq \varepsilon$ the transition t_i is said to be manipulated. Otherwise it is nonmanipulated. A place $p_i \in P$ is said to be measurable if the i -th column vector of φ is not null, i.e. $\varphi(\bullet, i) \neq 0$. Otherwise it is nonmeasurable.

Definition 5: An $IPN (Q, M_0) = (N, \Sigma, \Phi, \lambda, D, \varphi)$ described by the state equation (1) is event-

detectable iff the firing of any pair of transition t_i , t_j of (Q, M_0) can be distinguished from each other by the observation of the sequences of input-output symbols.

The following lemma [8] gives a polynomial characterization of event-detectable IPN.

Lemma 6: Let $(Q, M_0) = (N, \Sigma, \Phi, \lambda, D, \varphi)$ be an IPN described by the state equation (1). (Q, M_0) is event-detectable iff all φC columns are not null and different from each other.

III. BUILDING IPN MODELS

We deal with IPN models representing normal and faulty events and states. In a DES an internal event representing a fault is associated to a transition that leads to a place that represents the failed operation. Below is included the procedure followed in the construction of an IPN model.

Procedure 7: Building an IPN model (Q, M_0)

1. Build an IPN model of the normal behavior of the system (Q^N, M_0^N) , i.e. when no failures are considered. The set of places of this model is named P^N (normal places) and the set of transitions of this model is named T^N (normal transitions). Define the initial marking M_0 .
2. Define a set of possible failure states F^S . For every state in F^S build another set of places, named P^F .
3. Connect (Q^N, M_0^N) to the places of P^F through new transitions representing the faults (T^F).
4. The matrix φ is extended; one row is added for each $p_i^F \in P^F$; where $\varphi(p_i^N, \bullet) = \varphi(p_j^F, \bullet)$ if place p_i^N is connected to p_j^F through a unique transition in T^F .

IV. DIAGNOSER DESIGN

We present an extension to the diagnoser proposed in [7].

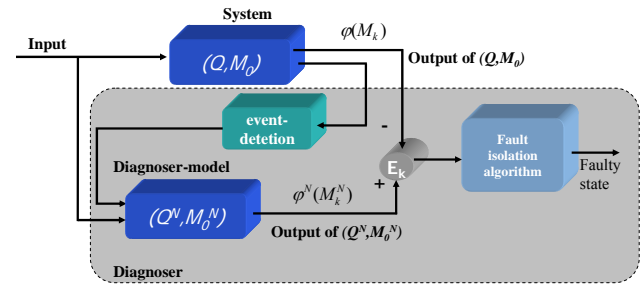


Fig. 1. Scheme for diagnosis

The features considered in this work are:

1. Some places are measurable (not all of them).
2. The diagnoser model is not synchronized with the system; rather the diagnoser model tracks the system.
3. A transition symbol is activated into the system when the transition is enabled or it was disabled by the firing of a faulty transition.
4. The diagnoser is robust and never fails.

These considerations are more realistic than those considered in all previous works since actual systems do not measure all the state variables; furthermore, true synchronization of the diagnoser with the system cannot be carried out in the general case and the diagnoser does not know the fault type within the system (it determines the fault type).

The diagnoser herein proposed consists of four modules: a diagnoser model (DM), an error computation algorithm, a fault isolation algorithm and an event-detection module. The diagnoser scheme is shown in figure 1; its components are defined below.

A. Diagnoser Model with one place

Definition 8: The proposed diagnoser model structure for the system normal behavior (Q^N, M_0^N) is an IPN (Q^d, M_0^d) where the set of places $P^d = \{p_d\}$ and the set of transitions is $T^d = T^N$, the incidence matrix C^d of (Q^d, M_0^d) is the following

$$C^d = B^T \varphi^N C^N \quad (2)$$

where C^N is the incidence matrix of (Q^N, M_0^N) , φ^N is the output function of (Q^N, M_0^N) and B is a $q \times 1$ non negative vector (q is the number of measurable places of (Q^N, M_0^N)) matrix with nonnegative entries.

The matrix B is computed as follows:

Algorithm 9: Building B

Inputs: C-incidence matrix of an IPN,
 q - number of measurable places in the IPN,
Outputs: The matrix B

1. The "base number" b should be computed. In this case $b = 2\max(\text{abs}(c_{ij})) + 1$, where c_{ij} is an element of incidence matrix C .
 2. Define a $q \times 1$ vector.
 3. $[b^0 \ b^1 \ \dots \ b^{q-1}]$
-

This procedure computes matrix B .

According to the way in which B was constructed, all columns of C^d will be different from zero and different from each other.

The initial marking of the diagnoser model structure is computed as:

$$M_0^d = B^T \varphi(M_0^N) \quad (3)$$

B. Error Computation of the DM with one place

Definition 10: Error computation. The k -th error is computed by the following equation:

$$e_k = M_k^d - B^T(\varphi M_k) \quad (4)$$

Notice that e_k is computed from the diagnoser-model output and not from the marking M_k . It means that the proposed diagnoser is using the system output and not internal system signals (those signals that are non measurable).

C. Diagnoser Model with more than one place

When there is a considerable quantity of tokens contained into the place of the diagnoser model, then a diagnoser model with more than one place is proposed in order to reduce the number of tokens contained in it.

For a diagnoser-model with more than one place, the incidence matrix C^d of (Q^d, M_0^d) is computed using the following equation:

$$C^d = B \varphi^N C^N \quad (5)$$

where B is a $l \times q$ matrix, where $l \leq q$.

The matrix B for this diagnoser model is obtained using the following algorithm.

Algorithm 11: Building matrix B

Inputs: C-incidence matrix of an IPN,
 l - number of places in the diagnoser-model,
 q - number of measurable places in the IPN,
Outputs: The matrix B

1. The "base number" b should be computed. In this case $b = 2\max(\text{abs}(c_{ij})) + 1$, where c_{ij} is an element of incidence matrix C .
2. Define a matrix B with $l \times q$, where $l \leq q$.

$$3. \quad B = \begin{bmatrix} b^0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & b^0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & b^0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & b^0 & b^1 & b^2 & \dots & b^{q-l} \end{bmatrix}$$

This procedure computes the matrix B .

According to the way in which B was constructed, all columns of C^d must be different from zero and different from each other.

The initial marking for this diagnoser model is computed as:

$$M_0^d = B\varphi(M_0^N) \quad (6)$$

D. Error Computation of the DM with one place

$$e_k = M_k^d - B(\varphi M_k) \quad (7)$$

E. Firing Rules of the diagnoser model with one or more places

If a transition $t_i \in T - (T^R \cup T^F)$ is fired in (Q, M_0) then it is fired in (Q^d, M_0^d) (it is possible since these transitions are event

The event-detection module determines which transition is fired into the system model and orders that this transition must be fired into the diagnoser model.

F. Fault Isolation Algorithm for the DM's

Definition 12: Fault isolation. When $e_k \neq 0$, an error is detected, then a faulty marking was reached. The mechanism used to find out the faulty marking is named fault isolation. This work proposes the following algorithm to accomplish this task.

Algorithm 13: Fault isolation

Inputs: M_k, M_k^d, e_k

Outputs: p (faulty place), M_f (faulty marking)

Constants: C^d is the IPN diagnoser structure incidence matrix

i = index of the column of C^d such that

$$C^d(1, i) = e_k$$

- $\forall p \in \cdot t_i, M_k(p) = 0$
- $\forall p \in t_i \cdot, M_k(p) = 0$
- $\forall p^F \in (\cdot t_i) \cap P^F, M_k(p^F) = 1$
- $M_f = M_k$
- Return (p, M_f)

Definition 14: Let (Q, M_0) be an input-output diagnosable IPN. The 3-tuple (N_B, e_k, A) , where

$N_B = (Q^d, M_0^d)$ is the diagnoser structure of (Q, M_0) ,

e_k is the error produced by error computation,

A is the algorithm Fault isolation,

is named an input-output diagnoser for (Q, M_0) .

We will prove that after the firing of a finite sequence, the input-output diagnoser for (Q, M_0) detects when a place $p_i \in P^F$ is marked, i.e. it isolates the faulty state.

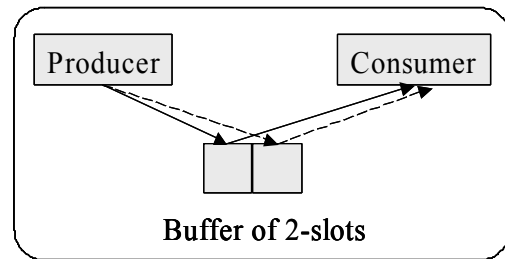


Fig. 2 Producer-Consumer with buffer of 2-slots scheme.

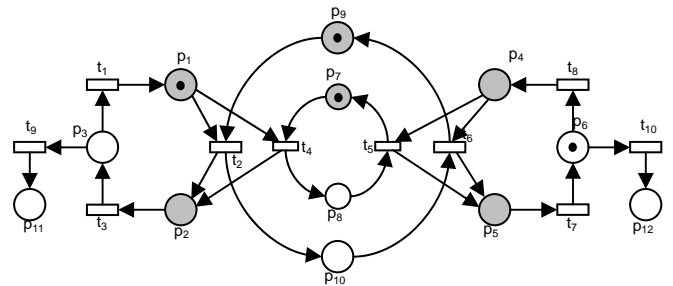


Fig. 3 IPN of the producer-consumer scheme

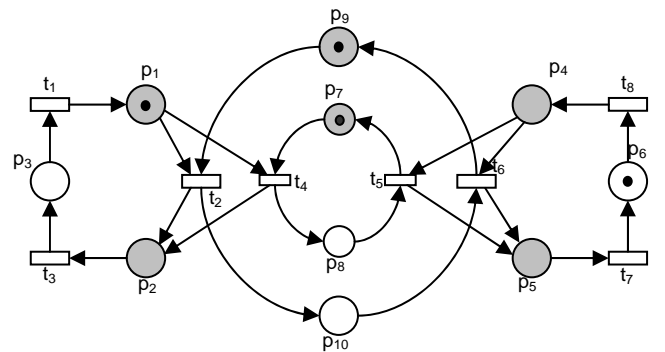


Fig. 4 Normal behavior of the IPN of figure 3.

V. EXAMPLE

Consider the producer-consumer scheme depicted in figure 2. The model consists of a producer unit (PU), a consumer unit (CU) and a buffer of 2-slots. The behavior of this system is the following. The producer unit PU creates and delivers products into the free buffer positions. The consumer unit CU retrieves products from the buffer when there is a product stored into a buffer slot. The producer unit PU could reach a faulty state from its producing state. Similarly, the consuming unit could reach a faulty state from its consuming state. Then the places p_1, p_2, p_3 represent the normal behavior of PU and p_{10} represents the faulty behavior. Places p_4, p_5, p_6 represent the normal behavior of the CU and p_{12} represents the faulty behavior. The places p_7, p_8, p_9 and p_{10} represent the 2- slots of the buffer.

The IPN (obtained with procedure 7) depicted in figure 3 represents the behavior of the producer-consumer system. Since this IPN is input-output diagnosable [9], then a diagnoser can be built for this system. In this case we will use the structure presented in previous section. The normal behavior of this IPN is depicted in figure 4; its incidence matrix and output function are:

$$C^N = \begin{bmatrix} 1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix}; \varphi = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The base obtained to compute B is $b=2*1+1=3$; since we build B using algorithm 9. We obtain the following vector:

$$B^T = [1 \ 3 \ 9 \ 27]^T$$

Therefore C^d is:

$$C^d = [-1 \ 27 \ 1 \ 9 \ -9 \ -27 \ 3 \ -3]$$

Hence, its associated IPN is depicted in figure 5.

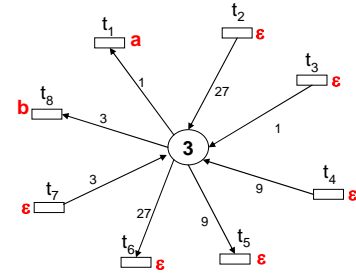


Fig. 5 The IPN diagnoser-model

The initial marking of the diagnoser is $M_0^d = [3]$. In order to show how the diagnoser works, assume that the following sequence $t_2 t_3$ is executed into the system, then this sequence is fired in the diagnoser. Thus the system output is $(\varphi(M_k))^T = [1 \ 1 \ 0 \ 1]$, and the marking of the IPN diagnoser is $M_k^d = [3 \ 1]$. Then $e_k = M_k^d - (B^T)(\varphi M_k) = [3 \ 1] - [3 \ 1] = 0$, thus the system is in a normal state. Now if the faulty transition t_9 is fired, then p_{11} is marked, however no change in the output system is detected. If the symbol of t_1 ($\lambda(t_1)=b$) is given as input to the IPN of the system model and IPN diagnoser, then the diagnoser evolves, and $M_{k+1}^d = [3 \ 0]$. Then $e_k = -1$ indicating the existence of an error. The fault isolation algorithm (algorithm 14) detects that the column 1 of C^d is equal to e_k , thus t_1 was not fired in the system. Then the same algorithm detects the faulty marking and determines that the faulty place p_{11} is marked.

If it is decided to compute a diagnoser model with more than one place ($l=3$), then the matrix B for this diagnoser will be the following:

$$B = \begin{bmatrix} b^0 & 0 & 0 & 0 \\ 0 & b^0 & 0 & 0 \\ 0 & 0 & b^0 & b^1 \end{bmatrix} \quad \text{or} \quad B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Thus C^d is:

$$C^d = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 3 & 0 & 1 & -1 & -3 & 0 & 0 \end{bmatrix}$$

The corresponded dignoser model is depicted in figure 6, where its initial marking is: $M_0 = [0 \ 1 \ 0]^T$.

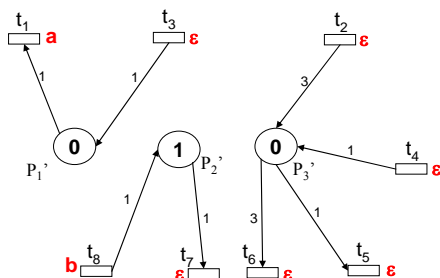


Fig. 6 The IPN diagnoser-model with three places.

It is easy to see that the diagnoser model almost uses 4 tokens into its places. Nevertheless, the diagnoser model (with one place) reaches almost 31 tokens. Thus, the number of tokens is considerably reduced in the diagnoser model with more than one place. But, it needs more places for monitoring the system than the diagnoser model with one place needs.

VI. CONCLUSION

We presented a diagnosis scheme allowing detecting and locating faults of partially observed DES. The diagnosability of the system implies the existence of a monitoring model; then two methods to conceive such models is proposed. Due to the simplicity of the monitoring IPN, the procedure for fault detection and isolation can be efficiently performed. Current research addresses the analysis of a methodology that reduces the potency that the base b is powered.

REFERENCES

[1] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. "Diagnosability of discrete event systems," *IEEE Trans. Automat. Contr.*, vol 4, no. 9, pp. 1555-1575, 1995.

[2] T. Murata. "Petri nets: Properties, analysis and applications," in *Proc. of the IEEE*, vol. 77, no. 4, pp. 541-580, 1989.

[3] Desel J. and J. Esparza. *Free Choice Petri Nets*, Cambridge University Press, 1995.

[4] J. Prock. "A New Technique for Fault Detection Using Petri Nets". *Automatica*, vol. 27, pp. 239-245, 1991.

[5] S. Genc, S. Lafortune. "Distributed Diagnosis of Discrete Event Systems Using Petri Nets," in *Proc. of the 24th ATPN*, 2003, pp. 316-336.

[6] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel, E. López-Mellado. "Diagnosability of Discrete Event Systems. A Petri Net Based Approach," in *Proc. of the IEEE ICRA*, 2004, pp. 541-546.

[7] E. Ruiz-Beltrán, E. López-Mellado, A. Ramírez-Treviño, "Fault Diagnosis based on Petri net reduced models", in the 3rd International Conference on Electrical and Electronics Engineering and XII Conference on Electrical Engineering, 2006, pp. 222-226, del 6-8 Sept., Veracruz, Veracruz México.

[8] A. Ramírez-Treviño, I. Rivera-Rangel, E. López-Mellado. "Observability of Discrete Event Systems Modeled by Interpreted Petri Nets, ", *IEEE Transactions on Robotics and Automation*, vol 19, no. 4, pp. 557-565, 2003.

[9] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel, E. López-Mellado. "On-line Fault Diagnosis of Discrete Event Systems. A Petri Net Based Approach", *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 1, pp. 31-39, 2007.

[10] E. Ruiz-Beltrán, I. Jiménez-Ochoa, A. Ramírez-Treviño, E. López-Mellado, M. Meda-Campaña. "Fault detection and location in DES modeled using Petri Nets," in *Proc. of the International Conference on Systems, Man and Cybernetics*, 2005, pp. 1645-1650.

FAULT DIAGNOSIS IN DISCRETE EVENT SYSTEMS USING DISTRIBUTED DIAGNOSERS

J. Arámburo-Lizárraga, A. Ramírez-Treviño, E. López-Mellado

CINVESTAV Unidad Guadalajara
Av. Científica 1145, Col. El Bajío. 45010 Zapopan, Jalisco, México
{jarambur, art, elopez}@gdl.cinvestav.mx

ABSTRACT

This paper deals with fault diagnosis in discrete event systems (*DES*). The approach herein presented uses Interpreted Petri Nets (*IPN*) to model both, the normal and faulty system behaviors. Based on this model, the property of diagnosability and on-line distributed diagnosers are presented. The approach herein presented exploits the *IPN* structure to determine when it is diagnosable; this approach leads to polynomial characterization of diagnosability. The diagnosability property is studied for the set of resulting models of the distributed Interpreted Petri Net. An architecture for reliable distributed on-line diagnosis is presented; it includes a set of redundant diagnosers that handle the corresponding reduced sub-model and other sub-models corresponding to the neighbour diagnosers.

Keywords: Petri nets, reliable distributed diagnosers, redundancy.

I. INTRODUCTION

Diagnosability property and fault detection schemes have been widely addressed on centralized approaches using the global model of the Discrete Event System (*DES*). Roughly speaking, diagnosability is the property of determining if using the system model is possible to detect and locate the faulty states in a finite number of steps.

Recently, fault diagnosis of *DES* has been addressed through a distributed approach allowing breaking down the complexity when dealing with

large and complex systems [1], [3], [4], [5], [6], [7], [9] and [10].

In [5] it is proposed a decentralized and modular approach to perform failure diagnosis based on Sampath's results [15]. In [4] and [10] the authors presented incremental algorithms to perform diagnosability analysis based on [15] in a distributed way; they consider systems whose components evolve by the occurrence of events; the parallel composition leads to a complete system model intractable. In [6] it is proposed a method that handles the reachability graph of the *PN* model in order to perform the analysis similarly to [15]; based on design considerations the model is partitioned into two labeled *PN* and it is proven that the distributed diagnosis is equivalent to the centralized diagnosis; later, [7] extends the results to systems modeled by several labeled *PN* that share places, and presents an algorithm to determine distributed diagnosis. In [11] it is studied the codiagnosability property, this property guarantee that any faults occurred in the system must be detected by at least one local diagnoser in a finite number of steps using the local information, besides, a notion of safe-codiagnosability is mentioned to capture the fact that the system has a safe specification while the system performance is tolerable. [2] Shows how to design low interaction distributed diagnosers reducing the communication among them.

This work considers the system modeled as an interpreted *PN* (*IPN*) allowing describing the system with partially observable states and events; the model includes the possible faults it may occur. We present a distributed architecture where every

distributed diagnoser uses the local information or communication among diagnosers for detecting and locating a system fault. The diagnosability property is preserved in the distributed architecture. Redundancy techniques could be applied to the distributed diagnosers to detect and locate a malfunction in the distributed diagnosers set.

The paper is organized as follows: section II provides basic definitions of *PN*, *IPN* and system modeling are presented. In section III the property of input-output diagnosability is defined and characterized. Section IV describes a method for model decomposition allowing interaction distributed diagnosers, also, it is presented a redundant scheme for reliable diagnosis applying redundancy to the distributed diagnosers.

II. BASIC NOTATIONS AND SYSTEM MODELING

II.1. PETRI NET BASICS

We consider systems modeled by Petri Nets and Interpreted Petri Nets. A Petri Net structure is a graph $G = (P, T, I, O)$ where: $P = \{p_1, p_2, \dots, p_n\}$ and $T = \{t_1, t_2, \dots, t_m\}$ are finite sets of nodes called respectively places and transitions, $I(O): P \times T \rightarrow$

\mathbb{Z}^+ is a function representing the weighted arcs

going from places to transitions (transitions to

places), where \mathbb{Z}^+ is the set of nonnegative integers.

The symbol $\bullet t_j$ ($t_j \bullet$) denotes the set of all places p_i such that $I(p_i, t_j) \neq 0$ ($O(p_i, t_j) \neq 0$). Analogously, $\bullet p_i$ ($p_i \bullet$) denotes the set of all transitions t_j such that $O(p_i, t_j) \neq 0$ ($I(p_i, t_j) \neq 0$) and the incidence matrix of G is $C=[c_{ij}]$, where $c_{ij} = O(p_i, t_j) - I(p_i, t_j)$.

A marking function $M: P \rightarrow \mathbb{Z}^+$ represents the

number of tokens (depicted as dots) residing inside each place. The marking of a *PN* is usually expressed as an n -entry vector.

A Petri Net system or Petri Net (*PN*) is the pair $N=(G, M_0)$, where G is a *PN* structure and M_0 is an initial token distribution. $R(G, M_0)$ is the set of all possible reachable markings from M_0 firing only enabled transitions. In a *PN* system, a transition t_j is enabled at marking M_k if $\forall p_i \in P, M_k(p_i) \geq I(p_i, t_j)$; an enabled transition t_j can be fired reaching a new marking M_{k+1} which can be computed as $M_{k+1} = M_k + C v_k$, where $v_k(i)=0, i \neq j, v_k(j)=1$.

Interpreted Petri Nets (*IPN*) [12] is an extension to *PN* that allow to associate input and output signals to *PN* models. An *IPN* (Q, M_0) is an Interpreted Petri Net structure where $Q = (G, \Sigma, \lambda, \varphi)$ with an initial marking M_0 , G is a *PN* structure, $\Sigma = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ is the input alphabet of the net, where α_i is an input symbol, $\lambda: T \rightarrow \Sigma \cup \{\varepsilon\}$ is a transition labelling function with the following constraint: $\forall t_j, t_k \in T, j \neq k$, if $\forall p_i I(p_i, t_j) = I(p_i, t_k) \neq 0$ and both $\lambda(t_j) \neq \varepsilon, \lambda(t_k) \neq \varepsilon$, then $\lambda(t_j) \neq \lambda(t_k)$; ε represents an

internal system event, and $\varphi: R(Q, M_0) \rightarrow (\mathbb{Z}^+)^q$ is

an output function that associates to each marking an output vector, where q is the number of outputs. In this work φ is a $q \times n$ matrix. If the output symbol i is present (turned on) every time that $M(p_j) \geq 1$, then $\varphi(i, j) = 1$, otherwise $\varphi(i, j) = 0$.

A transition $t_j \in T$ of an *IPN* is enabled at marking M_k if $\forall p_i \in P, M_k(p_i) \geq I(p_i, t_j)$. If t_j is enabled at marking M_k , and $\lambda(t_j)$ is present, then t_j can be fired reaching M_{k+1} , i.e., $M_k \xrightarrow{t_j} M_{k+1}$; M_{k+1} can be computed using the state equation:

$$\begin{aligned} M_{k+1} &= M_k + Cv_k \\ y_k &= \varphi(M_k) \end{aligned} \quad (1)$$

where C and v_k are defined as in PN and $y_k \in (\mathbb{Z}^+)^q$

is the k -th output vector of the IPN .

The sequence $\sigma = t_i t_j \dots t_k \dots$ is a firing transition sequence of an IPN (Q, M_0) if $M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots M_x \xrightarrow{t_k} \dots$. According to functions λ and φ , transitions and places of an IPN (Q, M_0) are classified. If $\lambda(t_i) \neq \varepsilon$ the transition t_i is said to be manipulated. Otherwise it is non-manipulated. A place $p_i \in P$ is said to be measurable if the i -th column vector of φ is not null, i.e. $\varphi(\bullet, i) \neq 0$. Otherwise it is non-measurable. The following concepts are useful in the study of the diagnosability property.

The set $\mathfrak{L}(Q, M_0) = \{ \sigma = t_i t_j \dots t_k \dots \wedge M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \dots M_x \xrightarrow{t_k} \dots \}$ of all firing transition sequences is called the firing language of (Q, M_0) .

A sequence of input-output symbols of (Q, M_0) is a sequence $\omega = (\alpha_0, y_0)(\alpha_1, y_1) \dots (\alpha_n, y_n)$, where $\alpha_j \in \Sigma \cup \{\varepsilon\}$. The symbol α_{i+1} is the current IPN input when the output changes from y_i to y_{i+1} . It is assumed that $\alpha_0 = \varepsilon$ and $y_0 = \varphi(M_0)$.

The firing transition sequence $\sigma \in \mathfrak{L}(Q, M_0)$ whose firing actually generates ω is denoted by σ_ω . The set of all possible firing transition sequences that could generate the word ω is defined as $\Omega(\omega) = \{ \sigma \mid \sigma \in \mathfrak{L}(Q, M_0) \wedge \text{the firing of } \sigma \text{ produces } \omega \}$.

The set $\Lambda(Q, M_0) = \{ \omega \mid \omega \text{ is a sequence of input-output symbols} \}$ denotes the set of all sequences of input-output symbols of (Q, M_0) and the set of all input-output sequences of length greater or equal

than k will be denoted by $\Lambda^k(Q, M_0)$, i.e. $\Lambda^k(Q, M_0)$

$= \{ \omega \in \Lambda(Q, M_0) \mid |\omega| \geq k \}$ where $k \in \mathbb{N}$.

The set $\Lambda_B(Q, M_0) = \{ \omega \in \Lambda(Q, M_0) \mid \sigma \in \Omega(\omega) \text{ such that } M_0 \xrightarrow{\sigma} M_j \text{ and } M_j \text{ enables no transition, or when } M_j \xrightarrow{t_i} \text{ then } c(\bullet, t_i) = \vec{0} \}$ denotes all input-output sequences leading to an ending marking in the IPN (markings enabling no transition or only self-loop transitions).

An IPN (Q, M_0) described by the state equation (1) is event-detectable iff the firing of any pair of transition t_i, t_j of (Q, M_0) can be distinguished from each other by the observation of the sequences of input-output symbols.

The following lemma [14] gives a polynomial characterisation of event-detectable IPN .

Lemma 1: A live IPN given by (Q, M_0) is event-detectable if and only if:

1. $\forall t_i, t_j \in T$ such that $\lambda(t_i) = \lambda(t_j)$ or $\lambda(t_i) = \varepsilon$ it holds that $\varphi C(\bullet, t_i) \neq \varphi C(\bullet, t_j)$ and
2. $\forall t_k \in T$ it holds that $\varphi C(\bullet, t_k) \neq 0$.

II.2. SYSTEM MODELING

We use the modeling methodology proposed in [13]. The methodology follows a modular bottom-up strategy. After identifying the system components, a set of state variables is assigned to every component, each state variable behavior is modeled by a PN model, herein named module. Then the set of modules are integrated into a single model according to the appropriate relationships achieved through two module composition operations. This methodology builds binary IPN modules to represent the behavior of each component of the identified SED and the relationships between them. The model captures the normal and faulty behavior of the individual components of the system. The sets of nodes are partitioned into faulty nodes (P^F , places coding

faulty states, and T^F , transitions leading to faulty states) and normal functioning nodes (P^N and T^N); so $P = P^F \cup P^N$ and $T = T^F \cup T^N$. p_i^N denotes a place in P^N . Since $P^N \subseteq P$ then p_i^N also belongs to (Q, M_0) . The set of risky places of (Q, M_0) is $P^R = \bullet T^F$. The post-risk transition set of (Q, M_0) is $T^R = P^{R\bullet} \cap T^N$. (Q^N, M_0^N) denotes the embedded normal behaviour of (Q, M_0) , i.e., (Q^N, M_0^N) is the subnet induced by normal nodes.

Example 1. Figure 1 presents an IPN model of a system. Notice that this model has four faulty states, represented by places $p_9, p_{10}, p_{16}, p_{17}$. Function λ is defined as $\lambda(t_1)=a, \lambda(t_4)=b, \lambda(t_5)=c, \lambda(t_8)=p, \lambda(t_9)=q, \lambda(t_{14})=r, \lambda(t_{15})=s$, for others transitions $\lambda(t_i)=\varepsilon$. Measurable places are $p_2, p_7, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}$, $P^R = \{p_7, p_8, p_{14}, p_{15}\}$, $T^R = \{t_8, t_9, t_{14}, t_{15}\}$, $T^F = \{t_6, t_7, t_{12}, t_{13}\}$ and $P^F = \{p_9, p_{10}, p_{16}, p_{17}\}$.

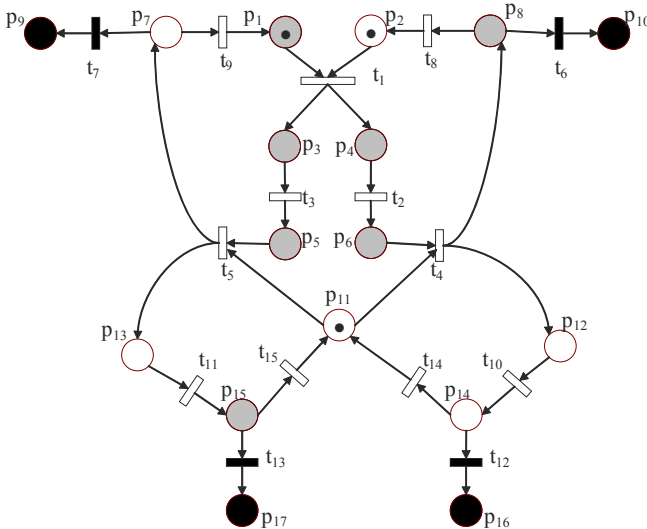


Fig. 1. Global model

III. CENTRALIZED DIAGNOSABILITY

The characterisation of input-output diagnosable IPN is based on the partition of $R(Q, M_0)$ into normal and faulty markings where all the faulty markings must be distinguishable from other reachable markings.

Definition 1: An IPN given by (Q, M_0) is said to be input - output diagnosable in $k < \infty$ steps if any marking $M_f \in F$ is distinguishable from any other

$M_k \in R(Q, M_0)$ using any word $\omega \in \Lambda^k(Q, M_f) \cup \Lambda_B(Q, M_f)$, where $F = \{M \mid \exists p_k \in P^F \text{ such that } M(p_k) > 0, M \in R(Q, M_0)\}$.

The following result provides sufficient structural conditions for determining the input-output diagnosability of an IPN model.

Theorem 1: Let (Q, M_0) be a binary IPN, such that (Q^N, M_0^N) is live, strongly connected and event detectable. Let $\{X_1, \dots, X_\tau\}$ be the set of all T-semiflows of (Q, M_0) . If $\forall p_i^N \in P^N, (p_i^N)^\bullet \cap T^F \neq \emptyset$ the following conditions hold:

1. $\forall i, \exists j X_j(j) \geq 1$, where $t_j \in (p_i^N)^\bullet - T^F$,
2. $\forall t_k \in (p_i^N)^\bullet - T^F, \bullet(t_k) = \{p_i^N\}$ and $\lambda(t_k) \neq \varepsilon$.

then the IPN (Q, M_0) is input-output diagnosable.

Proof: It is included in [13].

There exist different strategies for constructing the diagnoser-model, we presented distributed diagnosers which is very useful for large and complex system besides incorporate reliability to the system diagnosis process through redundancy techniques applied to the distributed diagnoser model.

IV. DISTRIBUTED DIAGNOSERS

iv.1. MODEL DISTRIBUTION

In order to build a distributed diagnoser, the IPN model (Q, M_0) can be conveniently decomposed into m interacting modules where different modules share nodes (transitions and/or places).

Definition 2. Let (Q, M_0) be an IPN. A module $\mu_k = (N_k, \Sigma_k, \lambda_k, \varphi_k)$ is an IPN subnet of the global model (Q, M_0) , where:

- $N_k = (T_k, P_k, I_k, O_k, I_k^C, O_k^C, M_{0k})$ where:
 - $T_k \subseteq T$,
 - $P_k = P_k^L \cup P_k^C$; $P_k^L \subseteq P$; P_k^C represents the communication places among modules; this set is a copy of some places

P_l^L that belongs to other modules, $l \neq k$. P_k^C is the minimal places that is required for the transitions of the module are event-detectable. $M(P_k^C) = M(P_l^L)$.

- $I_k^L(O_k)$: $P_k^L \times T_k \rightarrow Z^+$, s.t., $I_k(p_i, t_j) = I(p_i, t_j)$ ($O_k(p_i, t_j) = O(p_i, t_j)$), $\forall p_i \in P_k^L$ and $\forall t_j \in T_k$.
- $I_k^C(O_k^C)$: $P_k^C \times T_l \rightarrow Z^+$, s.t., $I_k^C(p_i, t_j) = I(p_i, t_j)$ ($O_k^C(p_i, t_j) = O(p_i, t_j)$), $\forall p_i \in P_k^C$ and $\forall t_j \in T_l$, $l \neq k$. $I_k^C(O_k^C)$ are the input(output) arcs to from $p_i \in P_k^C$ to transitions of other modules.
- $M_{0k} = M_0 |_{P_k}$
- $\Sigma_k = \{\alpha \in \Sigma \mid \exists t_i, t_i \in T_k, \lambda(t_i) = \alpha\}$
- $\lambda_k : T_k \rightarrow \Sigma_k \cup \{\varepsilon\}$, s.t. $\lambda_k(t_i) = \lambda(t_i)$ and $t_i \in T_k$
- $\varphi_k : R(\mu_k, M_{0k}) \rightarrow (Z^+)^{q_k}$, q_k is restricted to the outputs associated to P_k .

Definition 2. Let (Q, M_θ) be an IPN. A distribution DN_i of (Q, M_θ) is a finite set of m modules, i.e., $DN_i = \{\mu_1, \mu_2, \dots, \mu_m\}$. The distribution DN_i holds the following conditions:

1. $\bigcap_{k=1}^m P_k^L = \phi$; $\bigcup_{k=1}^m P_k^L = P$ (It is a partition over P_k^L).
2. $\bigcup_{k=1}^m T_k = T$ (It is not a partition)
- 3.

The set of communication places $P_{com} = \bigcup_{k=1}^m P_k^C$ of a distribution represents the measurable places of each module $\mu_k \in DN_i$ needed to guarantee the event-detectability property of μ_k .

Definition 3. Let DN_i be a distribution of (Q, M_θ) and $\sigma = t_i t_j \dots t_k \dots$ be a sequence of $\mathfrak{L}(Q, M_\theta)$. We define the natural projection PT_k of $\mathfrak{L}(Q, M_\theta)$ over the languages of the modules $\mu_k \in DN_i$ of the following way:

$$PT_k: \mathfrak{L}(Q, M_\theta) \rightarrow \mathfrak{L}(\mu_k, M_{0k})$$

$$\forall t_i t_j \dots t_s t_q \in \mathfrak{L}(Q, M_\theta)$$

$$PT_k(\varepsilon) = \varepsilon$$

$$PT_k(t_i t_j \dots t_s t_q) = \begin{cases} PT_k(t_i t_j \dots t_s) & \text{if } t_q \notin T_k \\ PT_k(t_i t_j \dots t_s) t_q & \text{if } t_q \in T_k \end{cases}$$

There exists the input-output symbol projection over the input and output module symbols. Let $\omega = (\alpha_0, y_0)(\alpha_1, y_1) \dots (\alpha_n, y_n)$ be a sequence of symbols of (Q, M_θ)

$P\Lambda_{\mu_k}(\omega) = ((P_{IN_k} \alpha_0, P_{OUT_k} y_0), (P_{IN_k} \alpha_1, P_{OUT_k} y_1) \dots (P_{IN_k} \alpha_n, P_{OUT_k} y_n))$ where:

$$P_{IN_k}(\alpha_i) = \begin{cases} \varepsilon & \text{if } \alpha_i \notin \Sigma_k \\ \alpha_i & \text{if } \alpha_i \in \Sigma_k \end{cases} \text{ and}$$

$$P_{OUT_k} \left(y_i = \begin{bmatrix} y_i(1) \\ \vdots \\ y_i(q) \end{bmatrix} \right) = \begin{bmatrix} y_i'(1) \\ \vdots \\ y_i'(q) \end{bmatrix}$$

where $y_i'(s) = 0$ if the measurable place does not belong to μ_k

where $y_i'(s) = y_i(s)$ otherwise.

Example 2. Consider the IPN model depicted in figure 1 (for the sake of simplicity, we use in the examples, the same names for places or transitions belonging to different modules). Figure 2 presents the distributed IPN, where three modules are considered ($m = 3$); $I_k^C(O_k^C)$ are represented by the dashed arcs. For example we can get the sets $T_1 \cap T_2 = \{t_1\}$ and $T_1 \cap T_3 = \{t_5\}$, $T_2 \cap T_3 = \{t_4\}$, $P_1^C = \{p_2, p_{11}, p_{13}\}$, $P_2^C = \{p_7\}$, $P_3^C = \{p_{11}, p_{12}\}$.

We are preserving the property of event detectability of the commons transitions using duplicated measurable places, which they establish the outputs that each module needs from others modules.

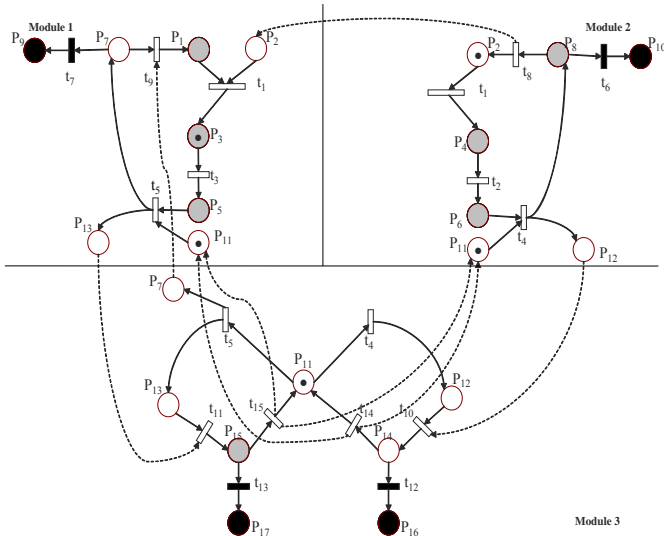


Fig. 2. Distributed Interpreted Petri Net

IV.2. DISTRIBUTED INPUT-OUTPUT DIAGNOSABILITY

The results of centralized diagnosability are applied to the modules.

A module is locally diagnosable if, for every local fault we can detect it only using local information, else it is conditionally diagnosable.

Definition 4. (Local Diagnosability) A module $\mu_n \in DN_i$ is said to be locally input-output diagnosable in $k < \infty$ steps if any faulty marking $M_{fn} \in R(\mu_n, M_{0n})$ is distinguishable from any other $M_{kn} \in R(\mu_n, M_{0n})$ using words ω_n , where $PA_{\mu_n}(\omega) = \omega_n$.

Definition 5. (Conditional Diagnosability) A module $\mu_n \in DN_i$ is said to be conditional input-output diagnosable in $k < \infty$ steps if any faulty marking $M_{fn} \in R(\mu_n, M_{0n})$ is distinguishable from any other $M_{kn} \in R(\mu_n, M_{0n})$ using words ω_n and ω_m , where $PA_{\mu_n}(\omega) = \omega_n$ and $PA_{\mu_m}(\omega) = \omega_m$, ω_m denotes the set of all input-output sequences that lead to a marking a duplicate place in μ_m , where $\mu_n \neq \mu_m$.

IV.3. REDUNDANT DIAGNOSER

Since distributed diagnosers leads to the use of several computers (CPU), then, redundancy can be

introduced in the diagnosers. For instance, the Triple Modular Redundancy (TMR) can be used in this case.

Assume that a distribution $DN_i = \{\mu_1, \mu_2, \dots, \mu_m\}$ was obtained and that it was distributed over m computers (see figure 3).

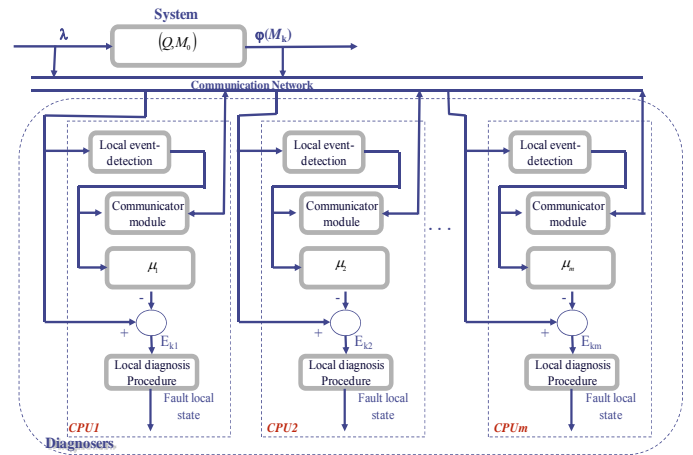


Fig. 3. On-Line Distributed Diagnosers

The service provided by a diagnoser is the detection and location of faults occurred within the system. If the diagnoser does not perform this function the dependability property is not guaranteed. Dependability is defined in [8] as the trustworthiness of a computer system such that reliance can justifiably be placed on the service it provides, and depending on the application, different attributes can be emphasized, namely reliability, availability, safety, and security. Reliability is the attribute most significant for fault tolerance; it deals with the continuity of service and reliability is strongly based on redundancy techniques such as duplication and Triple Modular Redundancy (TMR), which are widely used for designing dependable systems [16]. Then, a TRM scheme can be also applied to figure 3 (see figure 4), increasing the diagnoser reliability.

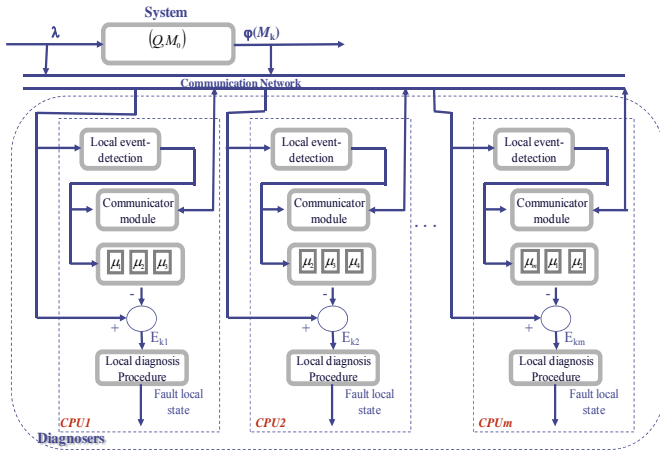


Fig. 4. On-Line Redundant Distributed Diagnoser

Example 3. Consider the DN_i distributed model depicted in figure 2, the figure 5 presents the local reduced models for distributed diagnosers where the place p_{11} is duplicated in module 2 for detecting the firing of t_4 . The communication between modules is represented by the dashed arcs. Reduced models are obtained as in [1] where we stated that the condition of event-detectability is needed only on $t_j \in \bullet P^R$ and $t_j \in P^R \bullet$, i.e. where faults can occur. A reduced model diagnoser contains the faulty nodes of the system normal behavior (Q^N, M_0^N) . Notice the diagnoser is a copy of the system. Now assume that the events represented by the sequence $t_1 t_2 t_3 t_4 t_{10}$ are executed into the system, then the sequence is projected in the distributed diagnoser models $PT_1(t_1 t_2 t_3 t_4 t_{10}) = \varepsilon$, $PT_2(t_1 t_2 t_3 t_4 t_{10}) = t_4$ and $PT_3(t_1 t_2 t_3 t_4 t_{10}) = t_{10}$. Thus both, the system and the distributed diagnoser models are marking the places p_8 and p_{14} . If the fault transition t_{12} is fired, then p_{16} is marked in the system (Q, M_0) and no system output change is detected. When the symbol of t_{14} , $\lambda(t_{14})$ is given as input to the system, then the diagnoser of module 3 evolves marking place p_{11} . Then there exist an output difference in the system and the diagnoser of module 3, thus the fault is isolated.

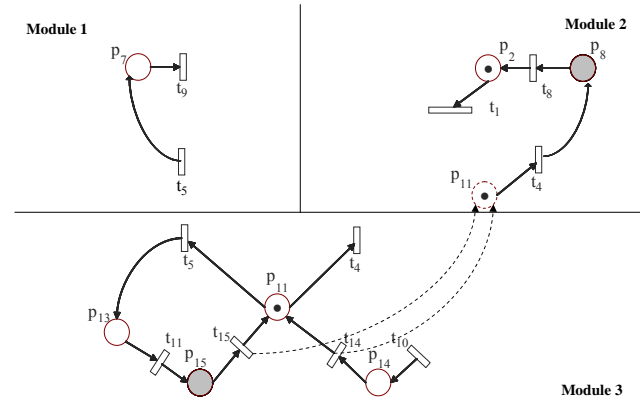


Fig. 5. Models for Distributed Diagnosers

V. CONCLUSION

This paper introduced the diagnosability property in *SED* modeled using *IPN*. It presented a structural characterization of this property using the T-semiflows of the *IPN*.

The approach herein presented exploits the *IPN* structure to determine when it is diagnosable; this approach leads to polynomial characterization of diagnosability. Based on the *SED* model, a distributed diagnoser architecture was presented which is very useful when the system could very large; leading to large diagnoser models thus the distributed diagnosers are designed to tackle this problem.

In this last case, a redundant diagnoser can be used for increasing the reliability of the distributed diagnosers.

VI. REFERENCES

[1] Arámburo-Lizárraga J., E. López-Mellado and A. Ramírez-Treviño. "Distributed Fault Diagnosis using Petri Net Reduced Models". Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics. pp. 702-707, October 2005.

[2] Arámburo-Lizárraga J., E. López-Mellado, and A. Ramírez-Treviño. "Design of Low

- Interaction Distributed Diagnoser for Discrete Event Systems*". Proc. of the 4th International Conference on Informatics in Control, Automation and Robotics. pp. 189-194. Angers, France. May 2007.
- [3] Benveniste A., S. Haar, E. Fabre and C. Jara. "*Distributed and Asynchronous Discrete Event Systems Diagnosis*". 42nd IEEE Conference on Decision and Control. 2003.
- [4] Contant O, S. Lafortune and D. Teneketzis. "*Diagnosis of modular discrete event systems*". 7th Int. Workshop on Discrete Event Systems Reims, France. September, 2004.
- [5] Debouk R, S. Lafortune and D. Teneketzis. "*Coordinated Decentralized Protocols for Failure Diagnosis of Discrete Event Systems*", Kluwer Academic Publishers, Discrete Event Systems: Theory and Applications, vol. 10, pp. 33-79, 2000.
- [6] Genc S. and S. Lafortune. "*Distributed Diagnosis of Discrete-Event Systems Using Petri Nets*". Proc. of the 24th. ATPN pp. 316 - 336, June, 2003.
- [7] Genc S. and S. Lafortune. "*A Distributed Algorithm for On-Line Diagnosis of Place-Bordered Nets*". 16th IFAC World Congress, Praha, Czech Republic, July 04-08, 2005.
- [8] Jalote P. "*Fault Tolerance in distributed systems*". Prentice Hall. 1994
- [9] Jiroveanu G. and R. K. Boel (2003). "A Distributed Approach for Fault Detection and Diagnosis based on Time Petri Nets". Proc. of CESA. Lille, France, July 2003.
- [10] Pencolé Y. "*Diagnosability analysis of distributed discrete event systems*". Proc. of the 15th International Workshop on Principles of Diagnosis. Carcassonne, France. June 2004.
- [11] Qiu W. and R. Kumar. "Decentralized Diagnosis of Event-Driven Systems for Safely Reacting to Failures". IFAC World Congress, Prague, July 2005.
- [12] Ramírez-Treviño A., I. Rivera-Rangel and E. López-Mellado (2003). "Observability of Discrete Event Systems Modeled by Interpreted Petri Nets". IEEE Transactions on Robotics and Automation, vol. 19, no. 4, pp. 557-565. August 2003.
- [13] Ramírez-Treviño A., E. Ruiz Beltrán, I. Rivera-Rangel, E. López-Mellado (2007). "On-line Fault Diagnosis of Discrete Event Systems. A Petri Net Based Approach". IEEE Transactions on Automation Science and Engineering. Vol. 4-1, pp. 31-39. January 2007.
- [14] Rivera-Rangel, I., A. Ramirez-Treviño, L.I. Aguirre-Salas and J. Ruiz-León. "*Geometrical characterization of Observability in Interpreted Petri Nets*". *Kybernetika*, vol. 41, pp. 553-574.
- [15] Sampath M., R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. "Diagnosability of discrete event systems", IEEE Transactions on Automatic Control, vol 40, no. 9, pp. 1555-1575, 1995.
- [16] Siewiorek, D.P. and R.S. Swarz. *Reliable computer systems: Design and Evaluation*, Digital Press. 1992.