

CASE 2019

Libro de Trabajos

Modalidades Artículo, Foro Tecnológico y Resumen

Congreso Argentino
de
Sistemas Embebidos

17 al 19 de Julio de 2019

Facultad de Ciencias Exactas Ingeniería y Agrimensura de
la Universidad Nacional de Rosario, Santa Fe, Argentina



ISBN 978-987-46297-6-0



Congreso Argentino de Sistemas Embebidos CASE 2019 : libro de trabajos en modalidad artículo, foro tecnológico y resumen / Antonelli, Maximiliano / Zacchigna, Federico G. / Blet, Nora / Sosa, José Ignacio / Pistarelli, Marcelo / De Micco, Luciana / Brengi, Diego Javier / Lipovetzky, José / Lutenberg, Ariel / Garcia Inza, Mariano. - 1a ed ilustrada. - Ciudad Autónoma de Buenos Aires : ACSE - Asociación Civil para la investigación, Promoción y Desarrollo de Sistemas Eléctricos Embebidos, 2019.
Libro digital, PDF

Archivo Digital: descarga
ISBN 978-987-46297-6-0

1. Circuitos Electrónicos. 2. Software. 3. Hardware. I. Antonelli, Maximiliano.
CDD 004

Fecha de catalogación: 1/7/2019

Libro de Trabajos
Modalidades Artículo, Foro Tecnológico y Póster
Congreso Argentino de Sistemas Embebidos - CASE 2019

Editores:

Antonelli, Maximiliano	UNMDP/ICyTE
Blet, Nora	UNR
Brengi, Diego	INTI/UNLaM/FIUBA
De Micco, Luciana	UNMDP/ICyTE/CONICET
Garcia Inza, Mariano	FIUBA
Lipovetzky, José	IB/CNEA/CONICET
Pistarelli, Marcelo	UNR
Sosa, José Ignacio	UNR
Zacchigna, Federico	FIUBA

Copyright © 2019

Asociación civil para la investigación, promoción y desarrollo de los sistemas electrónicos embebidos.



Se otorga permiso para copiar y redistribuir este libro de trabajos, siempre que se mantengan los mensajes de copyright y autoría de la obra y sus partes.

Prefacio

El diseño de sistemas embebidos es un motor clave de la industria y del desarrollo científico y tecnológico, y es un campo que en los últimos años ha crecido notablemente en la Argentina, tanto en la academia como en la industria.

El SASE busca fomentar esta temática realizando las siguientes actividades:

- CASE: Congreso Argentino de Sistemas Embebidos, presentación de trabajos científicos.
- Workshops: talleres prácticos en la modalidad *hands-on*.
- Tutoriales: charlas de capacitación.
- Plenarias: conferencias y debates abiertos.
- Concurso de proyectos tecnológicos: sobre trabajos finales y materias de grado.
- Concurso de emprendimientos tecnológicos: destinado a promover emprendimientos electrónicos con viabilidad económica.
- Programa de equipamiento para universidades: para transferir a las universidades las donaciones de los auspiciantes.
- Becas de viaje y alojamiento: ayudas económicas para viaje y estadía a estudiantes de grado, estudiantes de posgrado, docentes e investigadores de Argentina y Latinoamérica.

El SASE 2019 se desarrollará del 17 al 19 de julio en la Facultad de Ciencias Exactas Ingeniería y Agrimensura (FCEIA, <https://web.fceia.unr.edu.ar/es/>), de la Universidad Nacional de Rosario (UNR), sede Av. Pellegrini 250, Rosario. En dicha sede, donde se encuentra el edificio histórico de la FCEIA (1929), tienen su asiento el Decanato, Consejo Directivo, dependencias administrativas, algunos Institutos, Laboratorios y las Escuelas de Formación Básica, Agrimensura, Ingeniería Industrial, Ciencias Exactas y Naturales y, la Escuela de Posgrado y Educación Continua. Algunas de las carreras de la FCEIA se dictan en el Centro Universitario Rosario (CUR), donde también funcionan otras Facultades de la UNR. En el CUR, la FCEIA cuenta con numerosos edificios: las Escuelas de Ingeniería Electrónica y Civil, Mecánica y Eléctrica además, diversos Laboratorios, Centros e Institutos y, el

Reactor nuclear RA-4, desarrollan sus actividades en ese sitio, destacándose el Centro Científico Tecnológico Conicet Rosario (<https://www.rosario-conicet.gov.ar/>) donde, más de 1000 personas trabajan reunidas e interrelacionadas en 13 institutos y grupos de diferentes facultades

Los objetivos que persigue el CASE son:

- Ofrecer un lugar de encuentro para investigadores y becarios de todo el país, fomentando la colaboración.
- Difundir en el medio académico los adelantos científicos y tecnológicos producidos a nivel mundial.
- Propiciar la presentación y discusión de trabajos de investigación desarrollados en Argentina.
- Estimular en los estudiantes universitarios avanzados el interés por la investigación en el área de los sistemas embebidos.
- Difundir los proyectos de investigación mediante el desarrollo de un sitio web.
- Coordinar y actualizar los contenidos de sistemas embebidos de los programas de grado y posgrado de las universidades argentinas.

Las áreas temáticas del CASE se organizan de la siguiente manera: Arquitecturas de Procesadores, Bioingeniería, DSPs, FPGAs, HDLs y ASICs, Implementación de Sistemas Embebidos, Protocolos y Comunicaciones, Robótica, RTOS, Software Embebido, Linux Embebido y Comunicaciones Inalámbricas. Dentro de cada una de estas áreas se permiten las modalidades Artículo, Foro Tecnológico y Resumen, según el tipo de trabajo.

Los trabajos presentados al CASE fueron sometidos a un proceso de revisión por pares y posterior corrección. De este modo fueron seleccionados 12 trabajos en la modalidad Artículo, 23 en modalidad Foro Tecnológico y 12 en la modalidad Resumen.

Esta publicación se encuentra también disponible en forma *online* en la página web: www.sase.com.ar/case/

Esperamos que los trabajos recopilados en esta memoria sean de su interés y contamos con su participación en futuras ediciones del evento.

Atentamente,

Comité Organizador CASE

Auspiciantes Diamond

- Indusys
- IncluIT
- Synopsys Chile

Auspiciantes Platinum

- ClariPhy Argentina
- EMakers
- Fundación Fulgor (Desarrollo y formación en áreas estratégicas para el futuro del país)

Auspiciantes Gold

- Collabora
- Mayer
- Electrocomponentes
- IEA
- Semak

Auspiciantes Silver

- Fourtech Soluciones Integrales
- Digi tech
- Colegio de Ingenieros Especialistas de la Provincia de Santa Fe

Instituciones Auspiciantes

Institución Organizadora

- UTN-FRC (Universidad Tecnológica Nacional – Facultad Regional Córdoba)

Instituciones Coorganizadoras

- RUSE (Red Universitaria de Sistemas Embebidos)
- ACSE (Asociación Civil para la Investigación, Promoción y Desarrollo de los Sistemas Electrónicos Embebidos)

Instituciones Auspiciantes

- ANPCyT (Agencia Nacional de Promoción Científica y Tecnológica)
- MinCyT (Ministerio de Ciencia, Tecnología e Innovación Productiva)
- CONICET (Consejo Nacional de Investigaciones Científicas y Técnicas)
- Fundación Sadosky (Investigación y Desarrollo en TIC)
- ISOC (Internet Society)
- FIUBA (Facultad de Ingeniería Universidad de Buenos Aires)

Instituciones Acompañantes

- AADECA (Asociación Argentina de Control Automático)
- ADIMRA (Asociación de Industriales Metalúrgicos de la República Argentina)
- CAI (Centro Argentino de Ingenieros)
- CAME (Confederación Argentina de la Mediana Empresa)
- CAMOCA (Cámara Argentina de Máquinas de Oficina, Comerciales y Afines)
- CASEL (Cámara Argentina de Seguridad Electrónica)
- CEIL (Cámara de Empresas Informáticas del Litoral)
- CESSI (Cámara de Empresas de Software y Servicios Informáticos)
- CIIECA (Cámara de Industrias Informáticas, Electrónicas y de Comunicaciones del Centro de Argentina)
- CNEA (Comisión Nacional de Energía Atómica)
- CONFEDI (Consejo Federal de Decanos de Ingeniería)
- INTI (Instituto Nacional de Tecnología Industrial)
- MinCyT (Ministerio de Ciencia, Tecnología e Innovación Productiva)

Universidades Auspiciantes

- UBA: Universidad de Buenos Aires
- UNAJ: Universidad Nacional Arturo Jauretche
- UNC: Universidad Nacional de Córdoba
- UNCA: Universidad Nacional de Catamarca
- UNCOMA: Universidad Nacional del Comahue
- UNCUYO: Universidad Nacional de Cuyo
- UNER: Univerisdad Nacional de Entre Ríos
- UNICEN: Universidad Nacional del Centro de la Provincia de Buenos Aires
- UNLAM: Universidad Nacional de La Matanza
- UNLP: Universidad Nacional de La Plata
- UNLu: Universidad Nacional de Luján
- UNNE: Universidad Nacional del Nordeste
- UNNOBA: Universidad Nacional del Noroeste de la Provincia de Buenos Aires
- UNM: Universidad Nacional de Misiones
- UNMDP: Universidad Nacional de Mar del Plata
- UNPA: Universidad Nacional de la Patagonia Austral
- UNPSJB: Universidad Nacional de la Patagonia San Juan Bosco
- UNQ: Universidad Nacional de Quilmes
- UNR: Universidad Nacional de Rosario
- UNRC: Universidad Nacional de Río Cuarto
- UNS: Universidad Nacional del Sur
- UNSA: Universidad Nacional de Salta
- UNSJ: Universidad Nacional de San Juan
- UNSL: Universidad Nacional de San Luis
- UNSM: Universidad Nacional de San Martín
- UNT: Universidad Nacional de Tucumán
- UNTF: Universidad Nacional de Tres de Febrero
- UTN-FRA: Universidad Tecnológica Nacional - Facultad Regional Avellaneda
- UTN-FRBA: Universidad Tecnológica Nacional - Facultad Regional Buenos Aires
- UTN-FRBB: Universidad Tecnológica Nacional - Facultad Regional Bahía Blanca
- UTN-FRD: Universidad Tecnológica Nacional - Facultad Regional Delta
- UTN-FRH: Universidad Tecnológica Nacional - Facultad Regional Haedo
- UTN-FRLR: Universidad Tecnológica Nacional - Facultad Regional La Rioja
- UTN-FRM: Universidad Tecnológica Nacional - Facultad Regional Mendoza
- UTN-FRN: Universidad Tecnológica Nacional - Facultad Regional Neuquén
- UTN-FRP: Universidad Tecnológica Nacional - Facultad Regional Paraná
- UTN-FRRE: Universidad Tecnológica Nacional - Facultad Regional Resistencia
- UTN-FRRG: Universidad Tecnológica Nacional - Facultad Regional Rio Grande

- UTN-FRSF: Universidad Tecnológica Nacional – Facultad Regional San Francisco
- UTN-FRSN: Universidad Tecnológica Nacional – Facultad Regional San Nicolás
- UTN-FRVT: Universidad Tecnológica Nacional – Facultad Regional Venado Tuerto
- UTN-FRVM: Universidad Tecnológica Nacional – Facultad Regional Villa Mercedes
- UTN-FRT: Universidad Tecnológica Nacional – Facultad Regional Tucumán
- UADE: Universidad Argentina de la Empresa
- CAECE: Universidad CAECE
- FASTA: Fraternidad de Agrupaciones Santo Tomás de Aquino
- ITBA: Instituto Tecnológico de Buenos Aires
- CRUC-IUA: Centro Regional Universitario Córdoba - IUA
- UBP: Universidad Blas Pascal
- UCC: Universidad Católica de Córdoba
- UCU: Universidad Católica de Uruguay
- UCSE: Universidad Católica de Santiago del Estero
- UM: Universidad de Mendoza
- UREP: Universidad de la República del Uruguay

Coordinación General SASE

- Ing. Ignacio Sosa (UNR)

Comité Local CASE

- Mg. Nora Blet (UNR)

Comité Organizador CASE

- Dr. José Lipovetzky (IB/CNEA/CONICET)
- Dra. Luciana De Micco (UNMDP/ICyTE/CONICET)
- Mg. Diego Brengi (INTI/UNLaM/FIUBA)
- Dr. Maximiliano Antonelli (UNMDP/ICyTE)
- Ing. Federico Zacchigna (FIUBA)
- Dr. Ariel Lutenberg (FIUBA)
- Dr. Mariano Garcia Inza (FIUBA)

Chairs temáticos

- Bioingeniería: Ing. Juan Manuel Reta (UNER)
- DSP: Dra. Luciana De Micco (UNMDP/ICyTE/CONICET)
- Linux Embebido: Ing. Alejandro Furfaro (UTN-FRBA)
- Software Embebido: Dr. Maximiliano Antonelli (UNMDP/ICyTE)
- FPGA, HDL y ASIC: Ing. Salvador Tropea (INTI/UTN-FRBA)
- Implementación de Sistemas Embebidos: Dr. Gustavo Sutter (UAM) y
Mg. Cristian Sisterna (UNSJ)
- Arquitectura de procesadores: Ing. Alejandro Furfaro (UTN-FRBA)
- Comunicaciones y protocolos: Ing. Federico Zacchigna (FIUBA)
- Robótica: Claudio Verrastro (CNEA)
- RTOS: Dr. Ricardo Cayssials (UNS)
- Comunicaciones inalámbricas: Ing. Federico Zacchigna (FIUBA)

Chair de vinculación

- Dra. María Liz Crespo (ICTP)

Revisores

Alessandrini, Gustavo
Antonelli, Maximiliano
Arias, Ricardo
Belussi, Cristian
Brenzi, Diego
Briff, Pablo
Burgos, Enrique Sergio
Carbonetto, Sebastián
Cayssials, Ricardo
Comas, Edgardo
Corti, Rosa
Costa, Diego Esteban
De Micco, Luciana
Dondo, Julio
Ferreyra, Pablo Alejandro
Filomena, Eduardo
García Inza, Mariano
Gavinowich, Gabriel
Grimblatt, Victor
Hidalgo, Roberto
Leiva, Lucas

Lipovetzky, José
Lozano, Clevis
Lutenberg, Ariel
Martínez, Roberto
Martos, Pedro
Melo, Rodrigo Alejandro
Monte, Gustavo
Orallo, Carlos Martín
Ovilla-Martinez, Brisbane
Perez-Paina, Gonzalo
Perez, Santiago
Petraschin, Pablo Antonio
Reta, Juan Manuel
Sisterna, Cristian
Taffernaberry, Carlos
Todorovich, Elías
Tropea, Salvador
Uriz, Alejandro J.
Verrastro, Claudio
Zacchigna, Federico G.
Zaradnik, Ignacio

ÍNDICE DE TRABAJOS EN MODALIDAD ARTÍCULO

Título	Autores	Area	Pág
Sistema de adquisición de señales respiratorias para asistencia al monitoreo y diagnóstico	Gonzalo Fontanella, Gustavo Meschino, Mariel Azul Gonzalez, Marcelo Nicolas Guzman, Estefany Gabriela Cujano Ayala and Lucia Isabel Passoni	BIOINGENIERÍA	2
Mediciones en Celdas 4G LTE con Tecnología SDR	Diego Abel Gutierrez, Francisco Gimenez, Carlos Alberto Zerbini and Guillermo Riva	DSP	5
Implementación de Compresor IZ sobre FPGA	Lucas Leiva, Martín Vázquez, Oscar Goñi, Marcelo Tosini and José Noguera	FPGAS, HDLS Y ASICS	8
Algoritmo de ordenamiento híbrido implementado por Síntesis de Alto Nivel	Mariano Acosta, Maximiliano Antonelli and Luciana De Micco	FPGAS, HDLS Y ASICS	11
Hardware de a bordo basado en un procesador ARM para plataformas nanosatelitales	Adrian Stacul, Agustín Gadea, Ramiro Puga, Ariel Dalmas Di Giovanni, Martín Eugenio Morales and Sergio Saluzzi	IMPLEMENTACIÓN DE S.E.	14
Generación automática de modelos y código para sistemas electrónicos de enclavamiento ferroviario	Ramiro Adrián Ghignone, Facundo Santiago Larosa, Hernán Mendes Gouveia, Leandro Chang, Martín Nicolás Menéndez and Ariel Lutenberg	IMPLEMENTACIÓN DE S.E.	17
Sistema de supervisión de la seguridad del material ferroviario utilizando patrones de diseño	Ivan Mariano Di Vito, Pablo Gomez, Ariel Lutenberg and Adrian Laiuppa	IMPLEMENTACIÓN DE S.E.	20
Sistema de ensayos de relés ferroviarios de seguridad basado en computación en la nube	Gustavo Ramoscelli, Adrián Laiuppa and Ariel Lutenberg	IMPLEMENTACIÓN DE S.E.	23
Diseño de un sistema modular para el monitoreo y gestión de sistemas ferroviarios	Ariel Lutenberg, Adrián Laiuppa, Martin Amado, Bruno Palacios, Gustavo Ramoscelli, Lucas Dórdolo, Alejandro Permingeat, Carlos Mancón, Leandro Francucci, Dario Baliña, Pablo Martín Gomez and Santiago Germino	IMPLEMENTACIÓN DE S.E.	26
Extendiendo la cobertura del servicio SMS GSM en áreas rurales mediante tecnología a LPWAN	Guillermo Riva, Carlos Zerbini, Francisco Gimenez and Diego Gutierrez	IMPLEMENTACIÓN DE S.E.	29
Algoritmo para la mejora de las prestaciones de los flujos de datos utilizando Redes Centradas en Contenidos	Ignacio Navarrete, Diego Dujovne and Guillermo Riva	PROTOCOLOS Y COMUNICACIONES	32
Evaluación de Desempeño de un Patrón de Software para Multiprocesadores Asimétricos en Sistemas Embebidos	Pedro Martos and Alejandra Garrido	SOFTWARE EMBEBIDO	35

ÍNDICE DE TRABAJOS EN MODALIDAD FORO TECNOLÓGICO

Título	Autores	Area	Pág
Módulo vocalizador para balanza digital basado en EDU_CIAA_NXP	Eduardo Filomena and Diana Carolina Vertiz del Valle	BIOINGENIERÍA	39
Procesamiento de señal visualizado sobre un espectrograma.	Martin Ezequiel Paz, Guillermo Friedrich and Christian Luis Galasso	DSP	42
Implementación de un Vúmetro de 8 Bandas con la EDU-CIAA	Maximiliano Alexandroff, Tomás Rosales and José Juárez	DSP	45
Diseño de un sistema crítico de enclavamiento ferroviario implementado mediante FPGA	Martin Nicolás Menéndez, Facundo Santiago Larosa, Nicolas Alvarez, Ramiro Adrián Ghignone and Ariel Lutenberg	FPGAS, HDLS y ASICs	48
Adquisición de mediciones inerciales y de GNSS para sistemas de navegación integrada	Elián Hanisch, Germán Scillone, Santiago Rodríguez and Javier Garcia	FPGAS, HDLS y ASICs	51
Implementación de una Plataforma de Desarrollo CANSAT Multipropósito	Andrés Fernando Torti, Leonardo Agustín Anchino, Emmanuel Mario Dovis and Emanuel Bernardi	IMPLEMENTACIÓN DE S.E.	54
Automatización de un banco de ensayos de generadores eléctricos para aplicaciones en eólica de baja potencia	Juan Agotegaray, Andrea Pinzón and Nicolás Emanuel Lera	IMPLEMENTACIÓN DE S.E.	57
Implementación del control de un atenuador programable de RF mediante EDU-CIAA	Lucas Agustín Camino, Lihuen León Figueroa, Santiago Vicente Corrao and Gerardo Enrique Sager	IMPLEMENTACIÓN DE S.E.	60
Diseño y construcción de un Tacómetro Digital mediante el uso de una placa Arduino	Victor Claudio Juarez	IMPLEMENTACIÓN DE S.E.	63
Tecnologías Peltier para Control de Temperatura Aplicado a Sistemas de Refrigeración Láctea	Marco Antonio Palacios Curay, Wickson Janniro Castillo Vincas and Roosebelt Yonatan Carmen Aguirre	IMPLEMENTACIÓN DE S.E.	66
Control de motores DC sin escobillas basado en un microcontrolador de doble núcleo asimétrico	Ramiro Adrián Ghignone, Federico Joaquín Calá, Sharon Michelle Domanico, Julian Guido Giampetruzzii and Cristian Gabriel Juárez	IMPLEMENTACIÓN DE S.E.	68
Sistema de monitoreo de calidad de aire y ruido ambiental	Ignacio Zaradnik, Daniel Lupi, Agustín Agüero, Christian Behar, Leandro Lanzillotti and Vazquez Matias	IMPLEMENTACIÓN DE S.E.	71
eUCCvm: Plataforma educativa de programación y compilación cruzada universal de sistemas embebidos	Cristian Federico Perez-Monte, Gustavo Mercado, Carlos Taffernaberry, Ana Laura Diedrichs, Mario Sebastian Tobar, Enzo Crespillo, Raul Moralejo, Gabriel Caballero and Rodrigo Gonzalez	IMPLEMENTACIÓN DE S.E.	74
Implementación de un Adaptador de Video por Software en Microcontrolador de Doble Núcleo	Santiago Germino	IMPLEMENTACIÓN DE S.E.	77
Diseño e implementación de una arquitectura IoT en un sistema contador de pasajeros	Yoel Yamil López, Pablo Martín Gomez and Ariel Lutenberg	IMPLEMENTACIÓN DE S.E.	80
Implementación de un sistema de estimación de flujo vehicular y velocidad basado en procesamiento de video.	Felix Matias Taborda, Sergio Johan Comas and Enrique Sergio Burgos	LINUX EMBEBIDO	83
Sistema analógico de bajo costo para la adquisición y visualización de datos para aerogeneradores basado en el chip ESP8266	Maximiliano Ozono, Juan Agotegaray and Andrea Pinzon	PROTOCOLOS Y COMUNICACIONES	86
Desarrollo de dispositivos electrónicos para entrenamiento deportivo	Ramiro Romero Dapozo, Laura Andrea Fava, Diego Vilches Antao and Javier Francisco Díaz	PROTOCOLOS Y COMUNICACIONES	89
Análisis del desempeño de sensor embebido basado en etiqueta RFID pasiva	Pablo Garrone, Martín S. Baudino and Diego Vicente	PROTOCOLOS Y COMUNICACIONES	92
Plataforma flexible para nodos sensores inalámbricos	Javier G. Belmonte, Néstor Marcón, Marcelo Pistarelli, Rosa Corti, Roberto Martínez and Juan Pablo Vecchio	PROTOCOLOS Y COMUNICACIONES	95
Robot Móvil Omnidireccional Experimental con Interfaz de Comandos Táctil y Vocal	Melisa Kuzman, Juan Manuel López, Walter Andrés Gemin, Graciela Fernández, Raul Rivera and Miguel Revuelta	ROBÓTICA	98
Sistema de integración de datos para vehículo terrestre no tripulado	Jesús Ezequiel Benavidez, Enrique Sergio Burgos and Leonardo Giovanini	ROBÓTICA	100
Dispositivo para Posicionamiento Utilizando GPS y Acelerómetros en Tiempo Real de Bajo Costo	Francisco E. Paez, Jose Manuel Urriza and Rodrigo Tolosa	RTOS	103

ÍNDICE DE TRABAJOS EN MODALIDAD RESUMEN

Título	Autores	Area	Pág
Aplicando LSP en planificador para sistemas dinámicamente reconfigurables.	Alejandro Nuñez Manquez, Julio Dondo, Carlos Federico Sosa Paez, Martín Murdocca and Mario Marcelo Berón	ARQ. DE PROCESADORES	107
Clasificación de señales acústicas subacuáticas mediante el empleo de redes neuronales	Maria Celeste Cebedio and David Mario Petruzzi	BIOINGENIERÍA	110
Impresora basada en Interfaz Cerebro-Computadora	German Castro, Santiago Mac Mullen, Juan Manuel Geria and Nahuel Gonzalez	BIOINGENIERÍA	112
Comparación de resultados entre el uso de Nexys-2 y Pynq-Z1 para controlar una adquisidora de datos	Tomás Di Fiore, Hector Lacomí, Mario Lavorato, Nicolas Urbano Pintos and Cristian Donato	DSP	115
Determinación de eficiencia en la ejecución de algoritmos de procesamiento de imágenes con múltiples procesadores en FPGA	Jorge Osio, Walter Aroztegui, Antonio Quijano and Jose Rapallini	FPGAs, HDLS y ASICs	117
Sistema de Inspección Visual de Baldosas Cerámicas basado en SoC FPGA	Tomás Medina, Martín Vázquez, Oscar Goñi, Marcelo Tosini and Lucas Leiva	FPGAs, HDLS y ASICs	119
Implementación de Filtros Digitales SAL/CAL para la Medición de Voltaje Eficaz de la Red Eléctrica.	Marco Antonio Palacios Curay, Roosevelt Yonatan Carmen Aguirre, Wickson Janniro Castillo Vincas and Antenor Segundo Aliaga Zegarra	IMPLEMENTACIÓN DE S.E.	121
Implementación alternativa de sensores AM2302 en un firmware cooperativo	Ariel Dalmas Di Giovanni, Sergio Saluzzi, Martín Eugenio Morales, Adrian Stacul, Sebastián Alvarez and Daniel Pastafiglia	PROTOCOLOS Y COMUNICACIONES	123
Vulnerabilidad de los dispositivos utilizados en aplicaciones de IoT	Oswaldo Marianetti, Ernesto Chediack and Pablo Godoy	PROTOCOLOS Y COMUNICACIONES	126
Evaluación del Desempeño de una Placa Galileo para la Detección de Rostros en un Robot Autónomo	Miguel Revuelta, Graciela Fernández, Raul Rivera, Walter Andrés Gemin, Melisa Kuzman and Juan Manuel López	ROBÓTICA	128
Navegación autónoma en interiores mediante marcadores para Vehículos Aéreos No Tripulados	Juan Pablo Mercol, Manuel Alfonso and Cesar Rodriguez	ROBÓTICA	130
Actualización de un robot industrial de 6 grados de libertad	German Ghigi, Santiago Rojas and Luis Javier Vazquez	ROBÓTICA	133

ARTÍCULOS

Sistema de adquisición de señales respiratorias para asistencia al monitoreo y diagnóstico

Gonzalo Roberto Fontanella
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
Universidad Nacional de Mar del Plata
Mar del Plata, Argentina
g.fonta@fi.mdp.edu.ar

Mariela Azul González
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
Universidad Nacional de Mar del Plata
Mar del Plata, Argentina
mazulgonzalez@fi.mdp.edu.ar

Estefany Gabriela Cujano Ayala
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
Universidad Nacional de Mar del Plata
Mar del Plata, Argentina
ecujano@fi.mdp.edu.ar

Gustavo Javier Meschino
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
Universidad Nacional de Mar del Plata
Mar del Plata, Argentina
gmeschin@fi.mdp.edu.ar

Marcelo Nicolás Guzmán
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
CONICET
Mar del Plata, Argentina
marcelo.guzman@fi.mdp.edu.ar

Lucía Isabel Passoni
Instituto de Investigaciones Científicas
y Tecnológicas en Electrónica (ICYTE)
Universidad Nacional de Mar del Plata
Mar del Plata, Argentina
lpassoni@fi.mdp.edu.ar

Abstract— Las mediciones de parámetros respiratorios son un requerimiento de interés en el área de anestesiología y neumología. El incesante avance de los microcontroladores hizo que las posibilidades de diseñar dispositivos biomédicos de relativo bajo costo se ampliaron de forma considerable. Estos dispositivos son de gran importancia en el monitoreo de parámetros fisiológicos de interés a través del sensado y la consecuente adquisición de señales biomédicas. En este trabajo se presenta el diseño de un equipo de adquisición y procesamiento de parámetros respiratorios. Sus especificaciones surgen de los requerimientos de un médico anesthesiólogo. Puede ser utilizado como monitor constante, por ejemplo, en un paciente bajo anestesia, o bien para observar los cambios en parámetros respiratorios al realizar diferentes maniobras que el profesional médico puede solicitar al paciente. El equipo desarrollado permite el almacenamiento de las señales adquiridas, con fines de procesamiento offline, además de online, lo que puede proveer valioso material de investigación del efecto de maniobras o estados respiratorios en los parámetros medidos. Las señales también pueden ser mostradas en tiempo real, lo que hace del dispositivo una valiosa herramienta para el monitoreo.

Keywords—Sistema respiratorio, presión, caudal, volumen, monitoreo, diagnóstico

I. INTRODUCCIÓN

Las mediciones de parámetros respiratorios son un requerimiento de interés en el área de anestesiología y neumología. En particular, las señales de flujo de aire y el volumen pulmonar pueden utilizarse tanto para diferenciar los trastornos pulmonares obstructivos de los restrictivos, como para medir y caracterizar la gravedad de los estados patológicos permitiendo además el seguimiento de estos. Los trastornos obstructivos se caracterizan por la reducción del flujo de aire, mientras que los trastornos restrictivos se caracterizan por la reducción del volumen pulmonar.

Las evaluaciones del flujo inspiratorio y el volumen se realizan típicamente mediante un espirómetro, donde los pacientes espiran tan completamente como sea posible y después, inspiran con fuerza hasta el límite de su capacidad.

En las evaluaciones del flujo espiratorio, los pacientes realizan una inspiración tan profunda como sea posible, con sus labios cerrados alrededor de una boquilla, y espiran con tanta fuerza y de una manera tan completa como puedan en un dispositivo que registra el volumen espirado [1].

Los equipos actuales miden solo el flujo de aire e integran en el tiempo para estimar el volumen espirado. Suelen constituir dispositivos de costo significativo para las instituciones médicas. Otros equipos realizan las mediciones de estas señales de forma indirecta a partir de otras señales [2].

El incesante avance de los microcontroladores hizo que las posibilidades de diseñar dispositivos biomédicos de relativo bajo costo se ampliaron de forma considerable. Estos dispositivos son de gran importancia en el monitoreo de parámetros fisiológicos de interés a través del sensado y la consecuente adquisición de señales biomédicas [3].

En este trabajo se presenta el diseño de un equipo de adquisición de parámetros respiratorios. Sus especificaciones surgen de los requerimientos de un médico anesthesiólogo. Puede ser utilizado como monitor continuo, por ejemplo, en un paciente bajo anestesia, o bien para observar los cambios en parámetros respiratorios al realizar maniobras que el profesional médico puede solicitar al paciente.

Se utiliza un sensor de flujo, un sensor de presión y un sensor de temperatura acoplados convenientemente de forma no invasiva para poder ubicarse a la salida de la vía aérea. Además, mediante integración numérica embebida, se obtiene la señal de volumen inspirado y espirado.

Para ello, en el mismo sistema de adquisición de señales respiratorias se agregan sensores de presión y temperatura. El conjunto completo de sensores permite al profesional médico realizar un abordaje integral del estado del sistema respiratorio.

II. MATERIALES Y MÉTODOS

A. Equipo desarrollado

El equipo desarrollado es capaz de medir el caudal volumétrico respiratorio y, a través de procesamiento embebido, delegado en el microcontrolador, determinar el volumen pulmonar de aire.

Se instrumentó la medición de las variables requeridas a través de un microcontrolador PIC16F877A programado en lenguaje de alto nivel C, mediante el compilador CCS, versión PCWHD.

Se utilizan 3 sensores para medir las condiciones del flujo aéreo de la vía, tal como se ve en la Fig. 3.

El caudal volumétrico se mide mediante el sensor de la marca Sensirion (Staefa, Suiza) SFM300. Este sensor permite

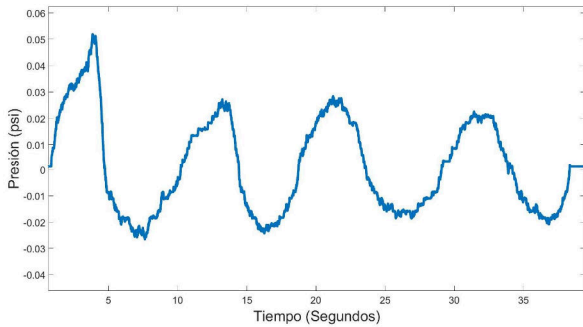


Fig. 1. Señal de presión adquirida [psi].

la medición del caudal con una precisión de 1.5%, frecuencia de muestreo de hasta 2 kHz y rango de medición de ± 200 litros por minuto. Transfiere la información mediante una interfaz I2C. Este dispositivo entrega un valor digital de 14 bits de resolución. A caudal nulo, entrega el valor 32000. A través de la siguiente expresión, implementada en el microcontrolador, se determina el caudal actual de aire:

$$\text{caudal} \left[\frac{l}{\text{min}} \right] = \frac{\text{valor digital recibido} - \text{offset}}{\text{factor de escala}} \quad (1)$$

donde, *offset* es el valor entregado por el sensor a caudal nulo (32000) y el *factor de escala* es una constante que depende del gas en cuestión, en este caso, la atmósfera (140).

Para la determinación de la presión en la entrada de la vía aérea, se utiliza un sensor Honeywell (Charlotte, Estados Unidos). Este constituye un manómetro alimentado con la misma alimentación del microcontrolador (5V). Cuando la presión absoluta entre sus boquillas es igual, entrega un valor de mitad de escala (2.5V) y su variación es lineal tanto para diferencias positivas o negativas, siendo su rango ± 1 psi.

En cuanto a la temperatura, se utilizó un sensor analógico LM35 (Texas Instruments, Dallas, Estados Unidos), el cual entrega una tensión de 10 mV/°C.

Teniendo disponibles estas 3 mediciones (Caudal, Temperatura y Presión), se procedió al desarrollo de un software que, embebido en el microcontrolador, adquiere dichas señales, cada una con su naturaleza: analógica en el caso de la temperatura y la presión, y digital en el caso del caudal.

Luego de la adquisición se calcula el valor de volumen de aire inspirado/espirdo mediante la integración de la señal de caudal respiratorio. Para realizar la integración, se utiliza un método numérico que consiste en la acumulación de los valores que resultan de multiplicar el periodo de muestreo con el valor de la señal de caudal.

Habiéndose procesado la señal de caudal y adquirido las de temperatura y presión, se procede al envío de la terna mediante el puerto serie del microcontrolador. Esta se adquirió en la PC para su representación mediante una interfaz USB-UART. Adicionalmente, se diseñó un método de visualización mediante una placa de video adicional compuesta por una pantalla de LCD gráfica de 320x240 píxeles manejada por un microcontrolador 18F46K22 como driver de video.

III. RESULTADOS

Se muestran las señales obtenidas, previamente almacenadas y cargadas en MATLAB. En la Fig. 1 se

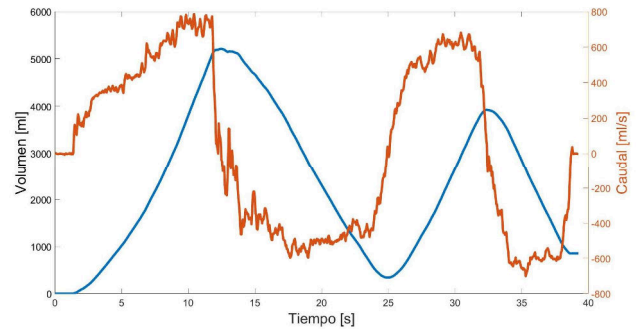


Fig. 2. Señal de caudal obtenida en la adquisición, en [ml/s] y señal de volumen calculada por integración en el microcontrolador [ml].

visualiza la señal de presión. En la Fig. 2 puede observarse el gráfico de la señal de caudal obtenida por el sensor, superpuesta a la señal de volumen calculada por integración de la anterior por el mismo microcontrolador.

Todas las señales pueden ser mostradas en forma sincrónica en la pantalla LCD. A su vez, como los datos se almacenan en formato genérico en una computadora, las señales pueden visualizarse y reprocesarse posteriormente.

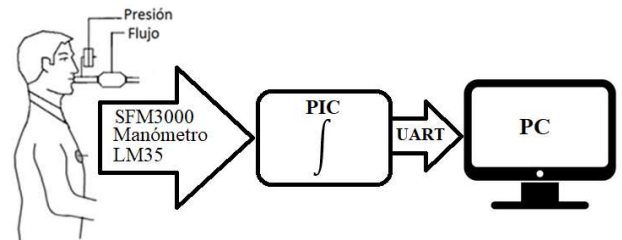


Fig. 3. Diagrama en bloques de la operación del adquisidor.

IV. CONCLUSIONES

Se diseñó un dispositivo de bajo costo, basado en un microcontrolador, para la adquisición y cálculo de diversas señales respiratorias, a través de sensores disponibles en el mercado.

El equipo desarrollado permite el almacenamiento de las señales adquiridas, con fines de procesamiento offline, lo que puede proveer valioso material de investigación del efecto de maniobras o estados respiratorios en los parámetros medidos.

Además, las señales pueden ser mostradas en tiempo real, lo que hace del dispositivo una valiosa herramienta para el monitoreo.

En el futuro inmediato se trabajará sobre aspectos de la sensibilidad y resolución de los sensores, con el fin de obtener señales cardiogénicas, lo que constituirá una relevante línea de investigación.

Posteriores procesamientos podrían considerar el cálculo de otros parámetros de interés, como la frecuencia respiratoria instantánea, valores medios, máximos y mínimos de los ciclos respiratorios, morfología del ciclo promedio, entre muchos otros.

AGRADECIMIENTO

Los autores agradecen al Dr. Gerardo Tusman, del Hospital Privado de Comunidad, de la ciudad de Mar del Plata, Argentina, por su interés en el presente trabajo y su invaluable colaboración.

REFERENCIAS

- [1] J. E. (John E. Hall, *Guyton y Hall. Tratado de fisiología médica*. Elsevier Health Sciences Spain - T, 2016.
- [2] P. H. Charlton *et al.*, "Extraction of respiratory signals from the electrocardiogram and photoplethysmogram: technical and physiological determinants," *Physiol. Meas.*, vol. 38, no. 5, pp. 669–690, May 2017.
- [3] K. van Loon, B. van Zaane, E. J. Bosch, C. J. Kalkman, and L. M. Peelen, "Non-Invasive Continuous Respiratory Monitoring on General Hospital Wards: A Systematic Review.," *PLoS One*, vol. 10, no. 12, p. e0144626, 2015.

Mediciones en Celdas 4G LTE con Tecnología SDR

Diego Gutierrez, Francisco Gimenez, Carlos Zerbini, Guillermo Riva

GInTEA, Universidad Tecnológica Nacional, Facultad Regional Córdoba (UTN-FRC)

diegocba9210@gmail.com, frangimz@gmail.com, czerbini@frc.utn.edu.ar, griva@frc.utn.edu.ar

Resumen—En este trabajo se presentan alternativas de medición sobre señales de telefonía celular 4G LTE mediante herramientas Software Defined Radio (SDR) de código abierto. Por un lado, se explora el estado del arte, describiendo los alcances técnicos de las herramientas, las capas sobre las cuales se destacan, y sus campos de aplicación. Por otro lado, se analizan las mediciones más comunes sobre celdas LTE, y se plantea un caso de aplicación concreto. Sobre esta base, se demuestra el enorme potencial de esta tecnología para realizar mediciones específicas, que actualmente requieren instrumentos dedicados de elevado costo, con aplicaciones que van desde la enseñanza e investigación, hasta el monitoreo de la salud de las redes móviles en servicio.

Index Terms—LTE, SDR, Open Source, eNB

I. INTRODUCCIÓN

La tecnología SDR permite implementar múltiples sistemas de comunicaciones sobre un mismo hardware, entre ellos las redes de telefonía móvil de cuarta generación (Long Term Evolution, LTE) [1]. En particular, poseen potencial para implementar mediciones que actualmente se presentan sólo en instrumentos específicos de muy alto costo. Si bien existe una gran variedad de proyectos que permiten extraer información de estas redes, no existe trabajo que procese esta información para realizar mediciones. El presente trabajo pretende demostrar cómo las herramientas SDR se pueden aplicar en casos concretos de medición.

II. PROYECTOS DE CÓDIGO ABIERTO

A continuación se comparan brevemente las principales herramientas existentes, respecto a la información que ellas extraen de la red. Sus principales características se resumen en el Cuadro I. Por mayores detalles sobre el sistema LTE, referirse a [1].

- *OpenLTE*. El principal objetivo de este proyecto es crear una estación base (eNB) básica [2]. Brinda asimismo scripts para simulación y configuración de tarjetas SIM. No incluye herramientas para medición de la red, si bien reporta todos los procedimientos que ejecuta. Actualmente se encuentra discontinuado.
- *gr-LTE*. Desarrollado como una librería de GnuRadio [3], tiene el objetivo de sincronizarse con una señal LTE, en tiempo real o previamente generada, y decodificar el Master Information Block (MIB) contenido en el Physical Broadcast Channel (PBCH). Asimismo, se documentan de manera completa los procesos de sincronización y decodificación.

- *srsLTE*. Se basa en OpenLTE y agrega la librería *libLTE* [4] [5], que le permite realizar mediciones y decodificaciones de utilidad para nuestros fines. Los datos obtenidos corresponden al MIB y el System Information Block Type 1 (SIB1), en canales PBCH y Physical Downlink Shared Channel (PDSCH) respectivamente, parámetros de potencia, y relación señal-ruido (SNR), entre otros. Este proyecto se encuentra en constante evolución.
- *Open Air Interface (OAI)*. Este proyecto es el más completo, con una gran comunidad trabajando activamente [6]. Como *srsLTE*, tiene la capacidad de realizar mediciones e implementar una red LTE completa, aunque su instalación y configuración es mucho más compleja.

III. MEDICIÓN SOBRE SEÑALES LTE

A fin de abordar casos prácticos de medición mediante las herramientas mencionadas, el primer aporte consiste en compilar e investigar las prestaciones de instrumentos comerciales típicos. Nos concentramos en el caso de mediciones *de capa física en transmisión de Downlink, mediante instrumentos de campo*, tomando como casos de referencia los equipos Anritsu BTS/Cell/Spectrum Master [7] y Rohde & Schwarz FSH Spectrum Analyzer [8]. Las mediciones de interés son:

1. Mediciones de calidad de radiofrecuencia (RF)
 - a) Potencia y ancho de banda de canal
 - b) Potencias en ON y OFF (sólo para frames TDD)
 - c) Emisiones fuera de banda
 - Relación de fuga de canal adyacente
 - Máscara de emisión espectral
 - d) Piso de ruido en recepción: interferencia en UL
2. Mediciones de calidad de modulación
 - a) Magnitud de Error Vectorial (EVM) pico y RMS
 - Según canal o señal: PBCH (control), PDSCH (datos), PCFICH, PSS, SSS
 - Según modulación: QPSK, 16QAM, 64QAM
 - b) Potencia de señales de soporte
 - Señales de sincronismo: PSS y SSS
 - Potencia en la señal de referencia (RS)
 - c) Error o corrimiento de frecuencia
 - d) Error de alineación de tiempo

Características	OpenLTE	gr-LTE	srsLTE	Open Air Interface
Lanzamiento / Últ. Update	2012 / 2017	2013 / 2013	2015 / 2019	2016 / 2019
Lenguaje	C++	C++, Python	C/C++	C
Dificultad de instalación	media	media	baja	alta
Legibilidad de código	muy buena	buena	excelente	buena
Implementa UE/eNB/EPC	× / ✓ / ×	× / × / ×	✓ / ✓ / ✓	✓ / ✓ / ✓
Mediciones	×	✓	✓	✓
Licencia	AGPLv3	GPLv3	AGPLv3	OAI 5G y Apache V2.0
Soporte de Hardware	OsmoSDR	General	USRP, LimeSDR, BladeRF	USRP, LimeSDR, BladeRF

Cuadro I: Proyectos de LTE de código abierto para aplicaciones con SDR

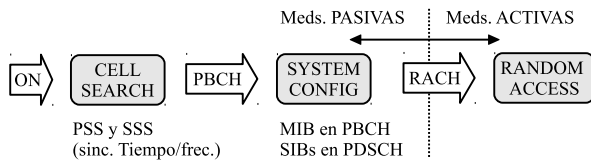


Figura 1: Procesamiento necesario para mediciones en LTE

IV. CASO DE APLICACIÓN

LTE, en su canal de DL, trabaja con un esquema de *grilla tiempo-frecuencia* [1], con anchos de banda y modulaciones variables. Las mediciones pueden ser *pasivas*, donde se observa la red, o *activas*, donde el instrumento debe integrarse a la red e interactuar con el eNB. En este trabajo nos ocupamos sólo de las pasivas, tomando como caso ejemplo la *medición de EVM de un canal dado* (caso 2a) que es la principal medición de calidad de señal. El procedimiento general necesario, ilustrado en la Fig. 1, es el siguiente.

En primer término, el instrumento realiza el procedimiento de *cell search* para sincronizarse con la red y obtener las señales de sincronismo primario y secundario (PSS y SSS) con el ID físico de las celdas. Posteriormente, se recupera el PBCH conteniendo el MIB con información esencial del sistema, como el ancho de banda utilizado, número de antenas y número de secuencia de frame.

En segunda instancia, se recupera la información de scheduling del SIB1, transportado en el PDSCH, a fin de conocer la periodicidad de los 31 tipos de SIBs necesarios para otras etapas del procesamiento. Luego, el instrumento puede buscar el SIB Type 2 (SIB2), que le permitirá realizar el procedimiento de *random access* mediante el Primary Random Access Channel (PRACH) para abordar mediciones activas.

Asimismo, para realizar estos pasos, se debe saber en qué zonas del espectro o Resource Blocks (RBs) se encuentran los SIBs. Para ello, el Physical Downlink Control Channel (PDCCH), ubicado al comienzo de cada subframe, informa a todos los UEs acerca de la posición de los SIBs en la grilla tiempo-frecuencia. En este contexto, se asigna a los UEs un identificador especial predefinido, el *System Information-Radio Network Temporary Identifier (SI-RNTI)*.

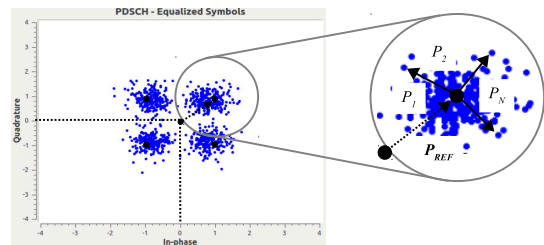
En la Fig. 2 se muestra un fragmento de los resultados obtenidos. El hardware utilizado es la placa SDR *Ettus USRP B200* (70 MHz-6 GHz, BW 56 MHz, ADC de 12 bits) sintoni-

```

Searching for cell...
*Found Cell_id: 177 CP: Normal , DetectRatio=100% PSR=23.32, Power=-15.6 dBm
Found Cell_id: 0 CP: Normal , DetectRatio=0% PSR=0.00, Power=-inf dBm
Found Cell_id: 0 CP: Normal , DetectRatio=0% PSR=0.00, Power=-inf dBm
Decoding PBCH for cell 177 (N_id_2=0)
-- Asking for clock rate 15.360000 MHz...
-- Actually got clock rate 15.360000 MHz.
-- Performing timer loopback test... pass
Setting sampling rate 15.36 MHz
Decoded MIB. SFN: 0, offset: 1 FrameCnt: 0, State: 1
Decoded SIB1. Payload: [48 5c 88 69] [28 9a] [07 6f d0 08] [18 31 60 81 04 4c 23] c9 52
PLMN Id LAC/TAC Cell Id Cell Selection Scheduling

```

(a)



(b)

Figura 2: (a) Búsqueda de celdas y decodificación de SIB1, (b) Constelación PDSCH y vectores para el cálculo de EVM.

zada en 2135,7 MHz correspondiente al DL E-UTRA Radio Frequency Channel (EARFCN) 2225, con ancho de banda en tiempo real 5 MHz (25 Resource Blocks). En primer término, el software realiza una búsqueda de las celdas circundantes, seleccionando la que posee *Señal de sincronismo (SS) con dominancia de +10 dB*. Una vez extraído el canal de interés de la grilla, se procede a su demodulación, que puede ser BPSK, QPSK, 16QAM o 64QAM. Se dispone así de la información transportada en la MIB y en la SIB1, mostrada en la Fig. 2(a), que se decodifica obteniendo campos de interés. En la Fig. 2(b), en tanto, se muestra la constelación QPSK para el caso del canal PDSCH, a partir de la cual se calculan los errores relativos $P_1/P_{REF}, \dots, P_N/P_{REF}$. Finalmente, calculando su valor RMS se determina el EVM de este canal.

V. CONCLUSIÓN

En base a la experiencia realizada, comprobamos que las mediciones sobre redes móviles, actualmente sólo realizables mediante instrumentos dedicados de elevado costo, pueden llevarse a cabo efectivamente combinando y adaptando herramientas SDR de código abierto. Estas técnicas permiten a la academia realizar investigación sobre redes móviles con recursos moderados, así como estudiar y optimizar la salud de estas redes.

REFERENCIAS

- [1] R. A. Comes, F. B. Álvarez y otros, "LTE: Nuevas Tendencias en Comunicaciones Móviles," Fundación Vodafone España, 2010.
- [2] "OpenLTE - An open source 3GPP LTE implementation," [Online:] <http://openlte.sourceforge.net/>
- [3] J. Demel, S. Koslowski, and Friedrich K. Jondral, "A LTE Receiver Framework Using GNU Radio," *Journal of Signal Processing Systems*, 78(3):313–320, 2015.
- [4] P. Sutton, "Examining 4G performance at MWC with AirScope," *Mobile World Congress 2017*.
- [5] V. Marojevic, D. Chheda, R. M. Rao, R. Nealy, J. Park and J. H. Reed, "Software-Defined LTE Evolution Testbed Enabling Rapid Prototyping and Controlled Experimentation," *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA, pp. 1-6, 2017.
- [6] F. García Espigares, "Prototipo de una estación base 4G usando Open Air Interface", 2017
- [7] *Anritsu BTS Master® MT8220T User Guide*, Anritsu Company, 2018.
- [8] *R&S® FSH Handheld Spectrum Analyzer Operating Manual*, Rohde & Schwarz Test and Measurement Division, 2015.

Implementación de Compresor IZ sobre FPGA

Lucas Leiva y Martín Vázquez
LabSET, Dpto. de Computación y Sistemas
UNICEN
Dpto. de Ciencia y Tecnología
UNTREF
Tandil, Argentina
{lleiva,mvazquez}@exa.unicen.edu.ar

Oscar Goñi, Marcelo Tosini y José Noguera
LabSET, Dpto. de Computación y Sistemas
UNICEN
Tandil, Argentina
{oegoni,mtosini}@exa.unicen.edu.ar;
jnoquera@alumnos.exa.unicen.edu.ar

Abstract—En el campo de investigación de la compresión de información, el objetivo principal es lograr la mayor reducción posible del volumen de datos. Sin embargo, los algoritmos más utilizados poseen una latencia considerable y resultan poco prácticos para aplicaciones del mundo real. Cuando se trata de imágenes, la situación resulta más compleja si se tiene en cuenta que las resoluciones de las mismas aumentan constantemente. En este sentido, resulta propicio el desarrollo de compresores dedicados que se ajusten a fuertes restricciones de tiempo, espacio y potencia. En este trabajo, se presenta el desarrollo de una arquitectura VHDL del algoritmo de compresión de imágenes sin pérdida ImageZero. Este algoritmo fue seleccionado por sus condiciones de baja latencia tanto en la compresión como en la descompresión. El trabajo presenta los resultados de síntesis para la arquitectura propuesta, con una frecuencia máxima de operación independiente de la resolución de la imagen de entrada.

Index Terms—Compresión de Imágenes, IP Core, ImageZero, FPGA

I. INTRODUCCIÓN

Los algoritmos de compresión de imágenes representan una imagen en el menor número de bits posible sin perder el contenido de información esencial. Los métodos de compresión pueden clasificarse en: sin pérdida (*lossless*) y con pérdida (*lossy*) [1]. En los métodos sin pérdida, la imagen comprimida debe ser una réplica exacta de la imagen original, y su principal aplicación es en la medicina [2]. Estos métodos también pueden encontrarse en el almacenamiento y transmisión de imágenes capturadas por satélites [3], [4] y aplicaciones de monitoreo remoto [5], como la vigilancia de incendios forestales [6].

En el benchmark [7] se evalúan diferentes compresores usando un conjunto de imágenes comn. Los resultados señalan que el algoritmo *ImageZero* posee una tasa de compresión competitiva frente a otras propuestas y un rendimiento superior, tanto en la compresión como en la descompresión, superando a PNG [8] o JPEG-LS [9].

El algoritmo de compresión *ImageZero* (IZ) [10] pertenece al conjunto de compresores de imágenes sin pérdida y está diseñado para ser utilizado en fotos naturales en espacio de colores RGB de 24 bits. El algoritmo utiliza un predictor de 3 píxeles, y los datos son codificados teniendo en cuenta un

valor de contexto único, utilizado para codificar los píxeles a partir de tablas estáticas de Huffman.

En este trabajo se aborda la implementación de una arquitectura para FPGA del algoritmo IZ. Se establece como límite de este trabajo sólo la compresión de datos (o *encoder*), asumiendo que esta etapa resulta más propicia de incluir en sistemas embebidos.

La sección 2 del trabajo presenta la arquitectura propuesta, y en la sección 3 sus resultados experimentales en cuanto a ocupación de área y estimación de tiempos. Finalmente, se exponen las conclusiones y líneas futuras de trabajo.

II. ARQUITECTURA PROPUESTA

El algoritmo IZ comienza con la escritura del encabezado del archivo resultante, que se compone de la concatenación del número de bits necesarios para almacenar alto y ancho de la imagen, codificado en 4bits (b), el alto de la imagen -1 en b bits (h), y el ancho de la imagen-1 (w) en b bits. Asumiendo que la arquitectura del compresor propuesta será implementada en una plataforma preestablecida (con resoluciones fijas), se considera que estos valores serán constantes, y sólo implica la escritura/transmisión de un valor constante al inicio de cada imagen.

IZ considera que cada píxel es representado en 24 bits (componentes *Red*, *Green* y *Blue*), utilizando 8 en bits en cada uno de ellos. El algoritmo está basado en un filtro de ventana (de dimensión 2×2). Para cada píxel $p[i, j]$ (denominado *pix*), el algoritmo predice el valor del mismo (pp) a partir de los valores de $p[i-1, j]$ (denominado y), $p[i, j-1]$ (también llamado x), y $p[i-1, j-1]$ (denotado como xy), con i en $\{0, 1, \dots, h-1\}$ y j en $\{0, 1, \dots, w-1\}$. En la arquitectura propuesta se consideró el predictor *Predictor3avgplane*.

La arquitectura propuesta para esta sección del algoritmo se presenta en la Fig. 1, en donde un píxel de la imagen $pix[i, j]$ (con i en $0, 1, \dots, h-1$ y j en $0, 1, \dots, w-1$) ingresa como la señal pix_in de 24 bits (R, G y B). Se observa que para procesar el píxel actual, se deben tener almacenados los $W+1$ píxeles anteriores. El valor a procesar, (pix) se corresponde con el valor almacenado en $reg[1]$.

Condiciones particulares se tienen en cuenta en los bordes de la imagen. Nótese que para $i > 0$ y $j > 0$, cuando $mode = 3$, el predictor se corresponde con el $reg[2]$ ($pix[i, j-1]$),

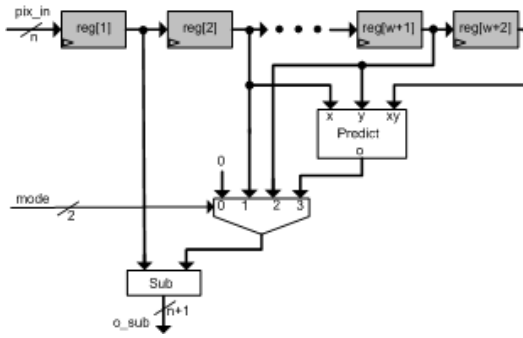


Fig. 1. Circuito *Process_Pixel*.

$reg[w + 1]$ ($pix[i - 1, j]$) y $reg[w + 2]$ ($pix[i - 1, j - 1]$). En el caso que $i = j = 0$ ($mode = 0$), el valor será 0; en el caso que $j > 0$ e $i = 0$ ($mode = 1$), el predictor es $reg[2]$; y en el caso que $j = 0$ e $i > 0$ ($mode = 2$), será $reg[w + 1]$.

El algoritmo continúa con la sustracción del valor correspondiente entre $pix[i, j]$ y $pp[i, j]$, dando lugar a $o_sub[i, j]$, y aplica una (*forward transform*) a estos valores obtenidos.

Los valores de la transformación y $o_sub_G[i, j]$ son llevados a una representación sin signo, que son utilizados para la actualización del valor del contexto. Este valor es el número de bits necesarios para codificar la mayor diferencia de componente entre el píxel predicho y el píxel real, y tiene en cuenta el valor nl del píxel anterior. Siendo nl el número de bits mínimo necesario para almacenar los componentes $o_sub[i, j]$, el valor de contexto se actualiza como se muestra en la Fig. 2 (reg_cx).

En la propuesta, esto implica un desplazamiento del valor almacenado en el registro reg_cx a izquierda de 4 posiciones y su concatenación con nl . El valor de nl se establece a partir de la selección del mayor valor de las codificación de prioridad resultante tanto para $o_sub_R[i, j]$, $o_sub_G[i, j]$ y $o_sub_B[i, j]$.

IZ utiliza códigos de Huffman canónicos fijos con hasta 6 bits de profundidad. Estos valores son almacenados en tablas estáticas que contienen los valores de los códigos ($dbits$), y sus dimensiones ($dcount$). En la arquitectura, estas tablas son implementadas como ROMs con valores constantes, y se presenta como el componente *Tables* en la arquitectura.

La arquitectura final propuesta para la implementación del *encoder IZ* se presenta en la Fig. 2. El componente *Control* genera las señales de control que se corresponden con la habilitación de los registros y la señal *mode*, la cuál vale 0 para el primer píxel, vale 1 para la primer fila, vale 2 para la primer columna, y 3 en el resto de los casos.

La salida de la arquitectura comprende la concatenación de los $dcount[reg_cx]$ bits obtenidos de la tabla $dbits[reg_cx]$, con los nl bits de $o_sub_G[i, j]$, $o_sub_R[i, j]$ y $o_sub_B[i, j]$. Esta funcionalidad se encuentra implementada en el componente *Gen Out* de la Fig. 2.

III. RESULTADOS EXPERIMENTALES

La arquitectura propuesta para la compresión IZ se describió en VHDL y se encuentra disponible en [11]. La arquitectura se validó para diferentes dimensiones de imágenes usando

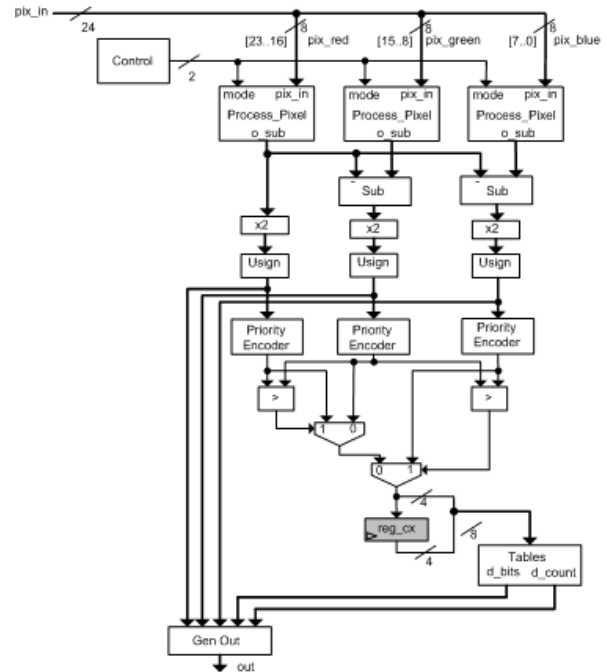


Fig. 2. Arquitectura de *Encoder IZ* propuesta.

testbenches, y comparando resultados con los ofrecidos por la librería.

Se implementaron diferentes arquitecturas en Xilinx Virtex 7 con speed grade 3 [12] (7v585tffg1157-3), considerando resoluciones de imágenes de: 256x256, 512x512, 1024x1024 usando Xilinx Vivado v2018.1. En todos los casos el tiempo de ciclo fue de 9.02 ns, y la latencia fue: 0.6 ms, 2.37 ms y 9.46 respectivamente. El número de LUTs usada en lógica: 286, 288 y 291, y como memoria: 192, 400 y 800.

El tiempo de ciclo es el mismo para todas las implementaciones, debido a que la ruta crítica es independiente de la dimensión de la imagen. Por otro lado, la ocupación de LUTs para lógica es similar a todas las configuraciones, pero incrementa en la medida que aumenta el tamaño de la imagen (parámetro w). Esto se produce porque el componente *Process_Pixel* debe registrar los $w + 1$ píxeles anteriores para poder procesar correctamente.

IV. CONCLUSIONES Y TRABAJOS FUTURO

En este trabajo se presenta una arquitectura para la compresión de imágenes del algoritmo IZ en VHDL. El diseño fue validado, y se presentan los resultados de síntesis tanto en tiempo como en área para diferentes dimensiones de imágenes de entrada.

Si bien las características del algoritmo permiten implementarlo en otras tecnologías (como microcontroladores), la arquitectura seleccionada ofrece simpleza respecto a otras alternativas, y se presume una significativa reducción de consumo de potencia. Se proyecta profundizar el análisis del consumo como continuación del trabajo.

En el caso que se desee incrementar la performance, se propone realizar una segmentación de la arquitectura. Por otra parte, el área de ocupación circuito puede ser reducida mediante la utilización de FIFO/BRAM del dispositivo destino. En consecuencia, esto disminuirá la ocupación de LUTs utilizadas como memorias.

REFERENCIAS

- [1] Madhuri Joshi, Mehul Raval, Yogesh Dandawate, Kalyani R. Joshi, and Shilpa Metkar. *Image and Video Compression: Fundamentals, Techniques and Applications*. 11 2014.
- [2] M Ferni Ukrit, A Umamageswari, and GR Suresh. A survey on lossless compression for medical images. *International Journal of Computer Applications*, 31(8):47–50, 2011.
- [3] Yadava Deigant, Vora Akshat, Hosangadi Raunak, Paliwal Pranjal, and Jain Avi. A proposed method for lossless image compression in nano-satellite systems. In *2017 IEEE Aerospace Conference*, pages 1–11. IEEE, 2017.
- [4] Bormin Huang. *Satellite data compression*. Springer Science & Business Media, 2011.
- [5] Bogdan Rusyn, Oleksiy Lutsyk, Yuriy Lysak, Adolf Lukenyuk, and Lubomyk Pohreliuk. Lossless image compression in the remote sensing applications. In *2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP)*, pages 195–198. IEEE, 2016.
- [6] Behçet Uğur Töreyn. Smoke detection in compressed video. In *Applications of Digital Image Processing XLI*, volume 10752, page 1075232. International Society for Optics and Photonics, 2018.
- [7] Alex Rhatushnyak. Lossless photo compression benchmark, June 2018.
- [8] Greg Roelofs and Richard Koman. *PNG: the definitive guide*. O’Reilly & Associates, Inc., 1999.
- [9] Shantanu D Rane and Guillermo Sapiro. Evaluation of jpeg-ls, the new lossless and controlled-lossy still image compression standard, for compression of high-resolution elevation data. *IEEE Transactions on Geoscience and Remote Sensing*, 39(10):2298–2306, 2001.
- [10] Christoph Feck. Fast lossless color image compression library, November 2013.
- [11] LabSET. Vhdl implementation of imagezero encoder, December 2018.
- [12] Xilinx. Virtex-7 FPGAs Product Advantage, January 2019.

Algoritmo de ordenamiento híbrido implementado por Síntesis de Alto Nivel

Luciana De Micco, M. Acosta y Maximiliano Antonelli
 Instituto de Investigaciones Científicas y Tecnológicas en Electrónica (ICYTE)
 Facultad de Ingeniería, Universidad Nacional de Mar del Plata (FIUNMdP)
 Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
 Buenos Aires, Argentina.

Resumen—En este trabajo se propone un algoritmo de ordenamiento de datos híbrido el cual ejecuta instrucciones en serie y en paralelo. Se presenta la implementación del sistema en un SoC (System on Chip) ZedBoard de Xilinx, el mismo recibe un vector de 1024 elementos y entrega el conjunto de índices de los 32 elementos mayores ordenados. Se analiza la complejidad del algoritmo en forma genérica y se evalúan los tiempos y recursos requeridos.

Keywords—Algoritmo ordenamiento, ZedBoard, System on Chip.

I. INTRODUCCIÓN

Muchos algoritmos requieren realizar ordenamiento de datos, por ello disponer de un método de ordenamiento eficiente es muy preciado ya que permite acelerar procesos complejos. De los algoritmos de ordenamiento existentes la mayoría trabajan en forma secuencial, como el método de burbujeo [1], árbol cartesiano [2], ordenamiento por casilleros [3], etc. El avance de la tecnología permite ejecutar instrucciones en forma secuencial y concurrente en un mismo dispositivo, lo que abre nuevas posibilidades y permite desarrollar algoritmos que combinen estos procesamientos a partir de los existentes. Por supuesto, existe siempre un compromiso entre velocidad y consumo de recursos. La implementación de algoritmos de ordenamientos en FPGA (field-programmable gate array) ha tenido un creciente interés en los últimos años [4], [5], [6], [7], [8], [9]. En este trabajo se combina procesamiento secuencial y concurrente a fin de optimizar el ordenamiento de datos, tanto en velocidad como en recursos empleados.

II. ALGORITMO PROPUESTO

El núcleo del algoritmo emplea la red de ordenamiento en paralelo que se puede ver en la Fig. 1. En la misma se puede apreciar que los elementos se almacenan en un registro de tamaño M y en cada ciclo de reloj se efectúan comparaciones de manera concurrente. En forma genérica esta red recibe M elementos y entrega ordenados en forma descendente los L mayores. El valor de M y de L determinan la cantidad de niveles de comparadores necesarios, lo que limitará la frecuencia máxima de operación (f_{CLK}). Si tenemos N datos a ordenar, se realizarán inicialmente $\frac{N}{M}$ comparaciones, y se obtendrán los $\frac{N}{M} \times L$ elementos mayores ordenados, que corresponderán a los grupos de L elementos mayores de cada comparación. Luego se realizan $\frac{N \times L}{M}$ nuevas comparaciones, obteniéndose $\frac{N \times L}{M} \times L$ elementos mayores. Así siguiendo hasta que se cumpla la desigualdad $N \times \left(\frac{L}{M}\right)^\alpha \leq M$.

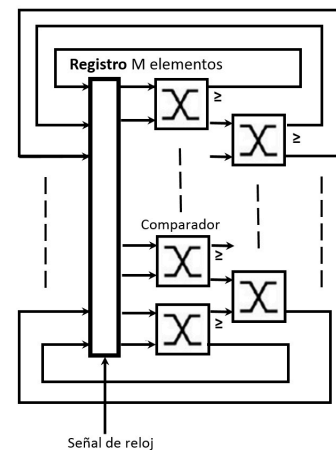


Figura 1. Interior del bloque Red de Ordenamiento (Paralelo).

$$\text{Despejando obtenemos, } \alpha = \frac{\log(M)}{\log\left(N \times \frac{L}{M}\right)}.$$

Si M , N y L tienen la misma base, la cantidad de iteraciones resulta ser $N_{iter} = \sum_{i=1}^{\alpha} \frac{N \times L^{(i-1)}}{M^i}$.

Sacando factor común N se puede ver que la cantidad de iteraciones crece en forma lineal con la cantidad de datos a ordenar. Esto es, el orden de complejidad del algoritmo es lineal $O(n)$.

III. IMPLEMENTACIÓN EN VIVADO HLS

Con el fin de describir funcionalmente este sistema, se muestra su diagrama en bloques en la Fig. 2. Primero, el diseño recibe un vector de $N = 1024$ elementos enteros sin signo de 32 bits mediante una interfaz AXI (Advanced eXtensible Interface) Stream [10]. Luego, se procede a agregar el índice correspondiente (posición del elemento en el vector) como cabecera de 10 bits a cada dato. Seguidamente, el vector ahora de 42 bits por 1024 se almacena en una memoria RAM. La máquina de estado, que controla la secuencia de pasos desde la recepción, procesamiento y transmisión, envía en bloques de $M = 128$ los datos a una red de ordenamiento en paralelo (Fig. 1). El elemento comparador siempre devuelve el dato de mayor magnitud por su terminal \geq . Si ambos son iguales, entonces no realiza el intercambio. Como puede observarse, existen dos niveles: 64 comparadores sucedidos por otros 62. A su vez, el sistema está realimentado, por lo que luego de cada etapa de ordenamiento el resultado se vuelve a guardar

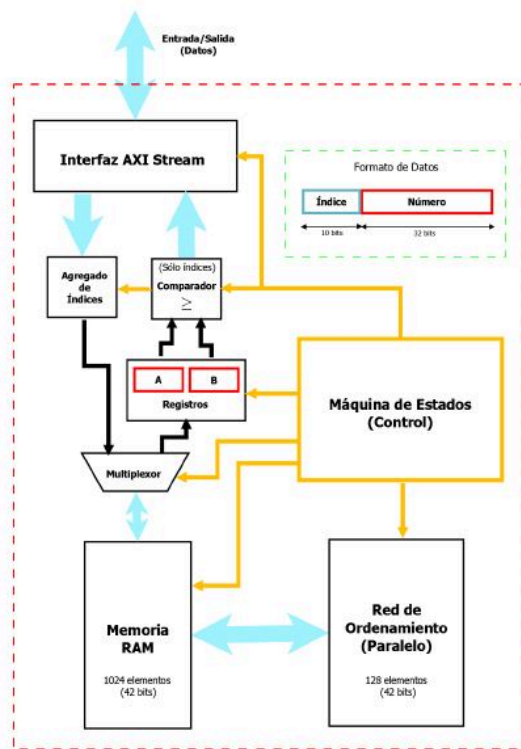


Figura 2. Diagrama en Bloques para el IP de ordenamiento.

en el registro. Iterando 64 veces, el segmento siempre queda ordenado de mayor a menor. Sin embargo, en cada proceso de ordenamiento de un bloque de 128 datos, sólo se almacenan los $L = 32$ mayores y se descarta el resto. Es decir, después de utilizar la red 8 veces, se tendrá 256 elementos restantes. Posteriormente, se vuelve a dividir al remanente en dos partes iguales para ordenarlas. Como resultado, se obtiene los 64 elementos mayores divididos en dos grupos ordenados de 32. Gráficamente, este procedimiento descrito se muestra en la Fig. 3.

Una vez que se obtienen los dos conjuntos de 32 datos, se procede a cargar los registros A y B con el primer dato de cada grupo. Ambos números son comparados y se transmite a la salida el índice (los primeros 10 bits) del elemento mayor o igual. El registro que contenía el valor cuyo índice fue transmitido es reemplazado por uno nuevo perteneciente al mismo conjunto que el anterior. De esta manera, luego de repetir este proceso 32 veces, se envía la ubicación de los 32 números mayores. Se utilizaron las directivas disponibles para la optimización de la síntesis (pragmas). Se utilizó HLS INTERFACE axis register para indicar que se quiere realizar una comunicación de datos mediante el estándar de comunicación AXI Stream y en particular HLS ARRAY PARTITION para permitir la lectura en paralelo de las memorias. No obstante, se hace uso de la sentencia `#pragma HLS INLINE recursive` para que la función Comp (que realiza las comparaciones) no se sintetice como un bloque aparte sino como varios elementos dentro de la red. Los lazos for llamados sort 1 y sort 2 tienen incluidos `#pragma HLS UNROLL` lo que indica al sintetizador que implemente todas las comparaciones en forma paralela.

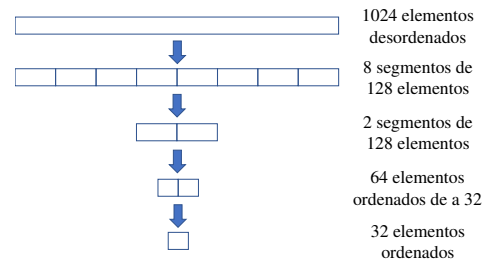


Figura 3. Procedimiento de ordenamiento.

Performance Estimates

Timing (ns)

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	7.78	1.25

Latency (clock cycles)

Summary

Latency		Interval		Type
min	max	min	max	
3107	5879	3108	5880	none

Utilization Estimates

Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	-	0	12217
FIFO	-	-	-	-
Instance	-	-	0	340
Memory	3	-	0	0
Multiplexer	-	-	-	16707
Register	-	-	16517	-
Total	3	0	16517	29264
Available	280	220	106400	53200
Utilization (%)	1	0	15	55

Figura 4. Resultado de timing y recursos utilizados.

IV. SÍNTESIS

Los resultados del análisis temporal se muestran en la Fig. 4. El período máximo de señal de reloj es 7,78 ns y el retardo, medido en ciclos, varía entre 3108 (31,08 μs) hasta 5880 (55,80 μs). Si el vector esta completamente ordenado (mejor caso) se tarda el tiempo mínimo. En caso contrario, el máximo. La red de ordenamiento utiliza el 55 % de las LUTs, 3 bloques de memoria BRAM y ningún elemento DSP (Fig. 4).

V. CONCLUSIONES

Se implementó un algoritmo de ordenamiento híbrido que combina procesamiento secuencial y concurrente en un dispositivo SoC. Se utilizó síntesis de alto nivel como técnica de descripción de hardware, con uso de directivas para optimización de la implementación mediante la declaración de pragmas en el código a sintetizar. Se reportaron los recursos requeridos y el análisis de tiempo del sistema. El trabajo futuro será realizar comparaciones con otros algoritmos de ordenamiento implementados en la misma plataforma para analizar recursos y tiempo de ejecución.

AGRADECIMIENTOS

Este trabajo fue parcialmente financiado por el Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), UNMDP Argentina e ICTP.

REFERENCIAS

- [1] O. Astrachan, "Bubble sort: an archaeological algorithmic analysis," in *ACM SIGCSE Bulletin*, vol. 35, no. 1. ACM, 2003, pp. 1–5.
- [2] E. D. Demaine, G. M. Landau, and O. Weimann, "On cartesian trees and range minimum queries," in *International Colloquium on Automata, Languages, and Programming*. Springer, 2009, pp. 341–353.
- [3] B. S. Chlebus, "A parallel bucket sort," *Information processing letters*, vol. 27, no. 2, pp. 57–61, 1988.
- [4] V. Sklyarov, I. Skliarova, and A. Sudnitson, "Fpga-based accelerators for parallel data sort," *Applied Computer Systems*, vol. 16, no. 1, pp. 53–63, 2014.
- [5] S. Dong, X. Wang, and X. Wang, "A novel high-speed parallel scheme for data sorting algorithm based on fpga," in *2009 2nd International Congress on Image and Signal Processing*. IEEE, 2009, pp. 1–4.
- [6] F. A. Alquaied, A. I. Almudaifer, and M. A. AlShaya, "A novel high-speed parallel sorting algorithm based on fpga," in *2011 Saudi International Electronics, Communications and Photonics Conference (SIEPCP)*. IEEE, 2011, pp. 1–4.
- [7] D. Mihhailov, V. Sklyarov, I. Skliarova, and A. Sudnitson, "Parallel fpga-based implementation of recursive sorting algorithms," in *2010 International Conference on Reconfigurable Computing and FPGAs*. IEEE, 2010, pp. 121–126.
- [8] R. Mueller, J. Teubner, and G. Alonso, "Sorting networks on fpgas," *The VLDB Journal?The International Journal on Very Large Data Bases*, vol. 21, no. 1, pp. 1–23, 2012.
- [9] A. Széll, "Parallel sorting algorithms in fpga," in *13TH PHD MINI-SYMPOSIUM*, 2006, p. 8.
- [10] A. Community. (1995) Introduction to axi protocol: Understanding the axi interface. [Online]. Available: <https://community.arm.com/developer/ip-products/system/b/soc-design-blog/posts/introduction-to-axi-protocol-understanding-the-axi-interface>

Hardware de a bordo basado en un procesador ARM para plataformas nanosatelitales

1^o Adrián Stacul

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
astacul@citedef.gov.ar

2^{do} Agustín Gadea

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
gadea.agustin@gmail.com

3^o Ramiro Puga

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
ramiropuga.93@gmail.com

4^{to} Ariel Dalmas Di Giovanni

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
adigiovanni@citedef.gov.ar

5^{to} Martín Morales

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
mmorales@citedef.gov.ar

6^{to} Sergio Saluzzi

Laboratorio de Técnicas Digitales
CITEDEF

Buenos Aires, Argentina
ssaluzzi@citedef.gov.ar

Resumen—El Instituto de Investigaciones Científicas y Técnicas para la Defensa de Argentina (CITEDEF) está desarrollando un módulo de hardware para el procesamiento de señales basado en un procesador ARM Cortex-M4 de la firma STMicroelectronics. El microcontrolador (MCU) tiene la capacidad de funcionar a una frecuencia de reloj máxima de 180 MHz, integra una unidad de punto flotante (FPU) y se agregó una SDRAM de 8MB para asignación dinámica de datos. Este hardware tiene la capacidad de poder alojar algoritmos para poder determinar la actitud de la plataforma nanosatelital, ya que gestiona un conjunto de sensores inerciales a tal fin. El módulo está diseñado para ser compatible con el estándar Cubesat, posee un diseño flexible y puede administrar copias de seguridad en tarjetas de memoria microSD de hasta 32 GB de capacidad.

Palabras clave—Hardware de procesamiento, Procesadores ARM, Nanosatélites, CubeSat

I. INTRODUCCIÓN

En el Laboratorio de Técnicas Digitales del Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF) se está desarrollando un módulo de procesamiento para ser aplicado en plataformas nano-satelitales. Este diseño debe cumplir con el requerimiento de ser capaz de interconectarse con un sensor solar (actualmente en desarrollo) y procesar diferentes tipos de algoritmos de navegación (actualmente en desarrollo). Este desarrollo es necesario para poder incorporar un sistema de determinación y control de actitud (Attitude Determination and Control System - ADCS).

El núcleo de procesamiento es un microcontrolador (MCU) con arquitectura ARM [1]. Para su selección, se consideraron aspectos de velocidad, flexibilidad y confiabilidad. Debido a que este módulo de hardware integrará parte del ADCS, el procesador utilizado debe ser lo suficientemente potente para poder realizar cálculos complejos (e.g., filtro de Kalman)

Proyecto financiado por el Ministerio de Defensa de Argentina (MINDEF) y ejecutándose en el Instituto de Investigaciones Científicas y Técnicas para la Defensa de Argentina (CITEDEF)

utilizando datos de punto flotante. Simultáneamente, debe ser capaz de poder manejar un gran volumen de datos, incluyendo la telemetría (tramas de datos) y el estado de salud del hardware. El firmware está configurado para que el MCU se encuentre en modo de bajo consumo la mayor parte del tiempo sin comprometer su eficiencia.

Se diseñaron y desarrollaron dos demostradores tecnológicos (prototipos) y luego de varias pruebas se implementó el circuito impreso final. Este diseño y desarrollo pretende ser compatible con las dimensiones impuestas por el estándar CubeSat [2] y hasta la fecha se han realizado algunos ensayos operativos del módulo: térmicos, vibratorios y de integridad de los datos. Aún debe estudiarse como lograr espacializarlo y realizar pruebas más específicas.

II. DESCRIPCIÓN GENERAL Y DISEÑO DEL SISTEMA

El sistema fue diseñado para tener las siguientes características y capacidades de interconexión:

- Una interconexión de subsistemas por medio de un bus central de datos [3]. Este interactúa con un EPS para el suministro y supervisión de energía, con un transceptor UHF para enviar tramas de datos a una estación en tierra, con una plaqueta de navegación y control y con puertos adicionales para aplicaciones de usuario y futuras expansiones.
- Un puerto UART [4] para depuración del sistema, enviando información en tiempo real del estado general del sistema y de todos los datos de los sensores internos.
- Una serie de sensores integrados en un único bus SPI cuya información se destinará a cálculos específicos de navegación. Estos son: una unidad medición de parámetros inerciales de 9 ejes (Invensense MPU-9250 [5] que integra un giroscopio, un acelerómetro y un magnetómetro), un sensor de temperatura (Texas Instruments TMP122

[6]) con 2°C de precisión en un rango de medición desde -55°C a 125°C, un acelerómetro lineal de 3 ejes de alto rendimiento y ultrabaja potencia (STMicroelectronics AIS328DQ [7]) con una escala dinámica seleccionable por el usuario capaz de medir aceleraciones con una frecuencia de muestreo hasta 1kHz.

- Para el almacenamiento local existe una memoria microSD que registra en tiempo real todos los datos del hardware. Además se cuenta con una interfaz SDIO y una memoria DRAM sincrónica de 64Mb (ISSI IS42S16400J [8]) para obtener un óptimo rendimiento y lograr transferencias de alta velocidad. Esto último es muy útil para tener la capacidad de implementar un sistema operativo en tiempo real (RTOS) en una actualización futura del sistema.

A. Diseño del Firmware

El firmware que se ejecuta dentro del MCU fue diseñado con el requisito de conocer de manera precisa el estado del hardware completo en todo momento. El tiempo de adquisición de todos los sensores internos y del estado del hardware están acotados dentro de un intervalo fijo, y se deja otro intervalo de tiempo libre para que el usuario pueda ensayar sus propios cálculos de navegación y de control. Para poder lograr esto fue necesario realizar un estudio de como disminuir los tiempos de adquisición de los sensores y los tiempos de lectura y escritura de la memoria [9] [10]. Periódicamente, el sistema se actualiza cada 100 ms mediante la implementación de un temporizador principal (heartbeat) [11], de allí se utilizan 10 ms para adquirir todos los sensores internos y realizar las operaciones de lectura y escritura de memoria quedando libre un intervalo de tiempo de 90ms para los otros cálculos.

Para la supervisión externa, la MCU tiene un IWDG el cual toma su base de tiempo del oscilador LSI, con el fin de lograr independizarse del MCU. El temporizador interno del IWDG debe reiniciarse cada 4 segundos, si se alcanza el tiempo de espera, el IWDG reiniciará la MCU y se recupera del estado problemático.

III. RESULTADOS

Se han medido las características principales del hardware desarrollado y los resultados se muestran en la Tabla I y la Tabla II.

TABLA I
MEDICIONES DE CORRIENTE

Descripción	Medición	Observaciones
Modo Sleep	$I_{media} = 25mA$	Periféricos en stand-by
Adquisición de sensores internos	$I_{media} = 34,3mA$	Con período de 100 ms
Consumo pico en formateo de MicroSD	$I_{pk} = 77,8mA$	Escribiendo todos los datos

IV. CONCLUSIONES

En este artículo se presentó el diseño, desarrollo e implementación de un sistema de hardware de procesamiento de

TABLA II
MEDICIONES DE TIEMPO

Descripción	Medición	Observaciones
Adquisición de sensores	1,51ms 22,8kBytes/s	Con reloj SPI de 185 kHz
Comunicación UART	8ms	Con Baudrate de 230400bps
Escritura en MicroSD	$T_{medio} = 7ms$ $T_{max} = 14ms$	Con sector de 512 bytes ^a
Escritura en MicroSD	$T_{medio} = 18ms$ $T_{max} = 24ms$	Con sector de 4096 bytes

^a Con esta configuración se pueden producir latencias significativas.

TABLA III
VALORES DEL HARDWARE

Parámetro	Mín	Tip	Máx	Unid.
Alimentación	4.4	5	20 ^a	V
Consumo	25	45	77.8	mA
Clock del procesador			180	MHz
Temperatura	-40		105	°C
Rango de aceleración	±2		±8 ^b	g
Rango del giróscopo	±250		±2000 ^c	°/s
Rango de magnetómetro		4800		uT
ODR		10		Hz
Clock del SPI		185 ^d	1000	kHz
Velocidad del SDIO		1	48 ^e	MHz
Velocidad de la UART			230400	bps
Tamaño de la SDRAM			8	MB

^a No se garantiza su correcto funcionamiento.

^b Escala configurable en ±2g, ±4g and ±8g .

^c Escala configurable en ±250°/s, ±500°/s, ±1000°/s y ±2000°/s.

^d Depende del clock de procesador.

^e No se garantiza su correcto funcionamiento.

alto rendimiento para poder ser aplicado a plataformas nano-satelitales. El módulo utiliza un procesador con arquitectura ARM con capacidad de velocidad de funcionamiento de hasta 180 MHz. Se desarrolló un software visual de prueba adicional para monitorear y detectar fallas del MCU o de algún sensor interno. El concepto de modularización y la disponibilidad de varios periféricos diferentes han permitido reutilizar y actualizar el diseño de manera sencilla. Este sistema pudo ser caracterizado y se obtuvieron resultados comparables con otros sistemas comerciales.

AGRADECIMIENTOS

Los autores desean agradecer al Ing. Daniel Pastafiglia, Jefe del Laboratorio de Técnicas Digitales, quien es el principal responsable de conseguir el subsidio y realizar la gestión proyectual. Al Ing. Edgardo Comas por su desempeño como Jefe del Departamento de Electrónica Aplicada del CITEDEF. Además, se quiere agradecer a Sebastián Alvarez y Gerardo García quienes han colaborado de manera extraordinaria en temas de software y metalmeccánica. Finalmente, se agradecen a las autoridades de CITEDEF y al Ministerio de Defensa de Argentina (MINDEF) que hacen posible este tipo de proyectos por medio subsidios y apoyo financiero.

REFERENCIAS

- [1] STM. (2018) STMicroelectronics. [Online]. Available: www.st.com
- [2] CubeSat. (2012) CubeSat Design Specification. [Online]. Available: www.cubesat.org/resources/
- [3] PC/104 Consortium. (2018) Hardware Specifications. [Online]. Available: pc104.org/hardware-specifications/pc104/
- [4] Future Technology Devices International - FTDI Chip. (2018) FT232R USB UART IC Datasheet. [Online]. Available: www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf
- [5] InvenSense Inc. (2016) MPU-9250 Product Specification. [Online]. Available: www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf
- [6] Texas Instruments. (2008) TMP122 - 1.5°C Accurate programmable digital temperature sensors with SPI interface. [Online]. Available: www.st.com/resource/en/datasheet/ais328dq.pdf
- [7] STMicroelectronics. (2012) AIS328DQ - High-performance ultra low-power 3-axis accelerometer with digital output for automotive applications. [Online]. Available: www.st.com/resource/en/datasheet/ais328dq.pdf
- [8] Texas Instruments. (2014) IS42S16400J - 1 Meg Bits x 16 Bits x 4 Banks (64-MBIT) synchronous dynamic RAM. [Online]. Available: www.issi.com/WW/pdf/42-45S16400J.pdf
- [9] G. Mayer, *Design and global analysis of spacecraft attitude control systems*. National Aeronautics and Space Administration - NASA, 1971. [Online]. Available: <https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19710009016.pdf>
- [10] P. Crouch, "Spacecraft attitude control and stabilization: Applications of geometric control theory to rigid body models," *IEEE Transactions on Automatic Control*, vol. 29, no. 4, pp. 321–331, April 1984.
- [11] B. Menges, C. Guadamos, and H. Pernicka, "Dynamic modeling of micro-satellite spartnik's attitude," *American Institute of Aeronautics and Astronautics*, 1997.

Generación automática de modelos y código para sistemas electrónicos de enclavamiento ferroviario

R. Ghignone, F. Larosa, H. Mendes Gouveia, L. Chang
CONICET-GICSAFe
Grupo de Aplicaciones en Sistemas Embebidos
Departamento de Electrónica, UTN-FRH
Haedo, Buenos Aires, Argentina
embebidos@frh.utn.edu.ar

A. Lutenberg, M. Menéndez
CONICET-GICSAFe
Laboratorio de Sistemas Embebidos
Facultad de Ingeniería - UBA
Buenos Aires, Argentina
lse@fi.uba.ar

Abstract—Este trabajo presenta el diseño y desarrollo de una herramienta para la simulación, implementación y verificación de sistemas de enclavamiento ferroviarios. Debido a la variedad de casos y al elevado nivel de seguridad que deben cumplir estos sistemas, se necesita una solución flexible, capaz de interpretar requisitos e implementarlos adecuadamente. La herramienta comprende un generador automático de código para transformar las tablas de uso estándar que describen la lógica del enclavamiento en modelos en lenguaje C, Python o VHDL. Luego, estos modelos se enlazan con una biblioteca de clases que permite simular y visualizar su comportamiento o implementarlo sobre un sistema embebido basado en un microcontrolador o FPGA, e incluso construir interfaces gráficas de operación basadas en el mismo modelo.

Índice de términos—Enclavamiento ferroviario, generación automática de código, programación orientada a objetos, seguridad funcional, sistemas críticos, FPGA.

I. INTRODUCCIÓN

Hoy en día, la red argentina de trenes es uno de los medios de transporte más usados en el país. Cada año se movilizan en Argentina más de 400 millones de pasajeros en ferrocarril [1]. Además, el país cuenta con una de las diez redes ferroviarias más extensas del mundo, con más de 36.000 km de vías férreas [2]. Sin embargo, en la mayor parte de estas vías se emplean sistemas de señalamiento mecánicos o electromecánicos (basados en relés y lógica cableada). Estos son sistemas costosos que no aprovechan los avances tecnológicos y muchas veces son difíciles de mantener o reponer.

Esto se contrapone con las tendencias en la industria y en otros países con similar extensión ferroviaria, tanto desarrollados como emergentes, donde se han implementado nuevas soluciones de enclavamiento electrónico [3-13]. Frente a este contexto, se propone el desarrollo local de un camino alternativo para implementar este tipo de sistemas. Esto permitiría reducir el costo de actualización e impulsar el desarrollo de la industria nacional.

Cabe destacar que este proyecto para el desarrollo local de sistemas de enclavamiento electrónico es de interés para Trenes Argentinos, y dicha empresa participa activamente en todos los procesos de diseño y toma de decisiones, asegurando así que el resultado sea de real utilidad para la red ferroviaria argentina.

II. SISTEMAS DE ENCLAVAMIENTO

A. Definiciones

Un sistema de enclavamiento ferroviario tiene por función garantizar el accionamiento seguro de los dispositivos de señalamiento y de maniobra de los trenes. Entre dichos dispositivos de señalamiento se pueden distinguir [14]:

- Circuitos de vía: detectan la posición del tren.
- Señales: indican órdenes de marcha al conductor.
- Cambios de vía: determinan la circulación del tren en una bifurcación.
- Pasos a nivel: permiten el paso peatonal o vehicular.

B. Tecnologías de implementación

La Fig. 1 muestra un modelo en capas propuesto para el estudio de sistemas de enclavamiento a partir del análisis de sistemas comerciales [3-13]. Se pueden distinguir tres capas:

1. **Logística:** contiene la interfaz gráfica, eléctrica y/o mecánica que permite al operador de señalamiento controlar y monitorear el sistema.
2. **Lógica:** contiene la unidad central que procesa la lógica vital del sistema. Debe cumplir un nivel de seguridad funcional SIL 4 (del inglés *Safety Integrity Level*) [15].
3. **Infraestructura:** contiene la interfaz entre la unidad central y los dispositivos de señalamiento y maniobra en campo (señales, cambios, circuitos de vía, pasos a nivel).

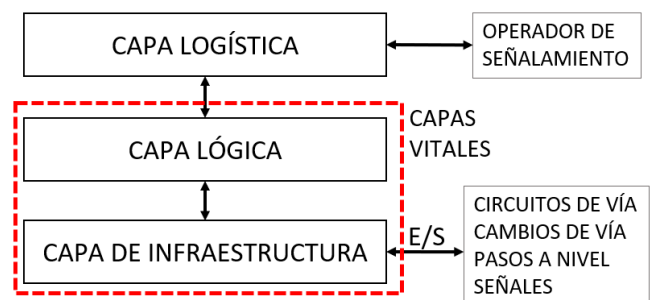


Fig. 1 – Modelo en capas de un sistema de enclavamiento completo

Si bien los sistemas de enclavamiento electrónico emplean en su mayoría microprocesadores para la capa lógica [3-13], un enfoque interesante es el uso de FPGA (del inglés *Field Programmable Gate Array*) propuesto por [16-17], que tiene varias ventajas respecto a las alternativas comerciales basadas en microprocesadores, entre las cuales se pueden listar:

- Mayor facilidad para escalar y agregar funcionalidades (por ejemplo, conectividad o redundancia).
- Menor costo y dificultad de verificación que si se usaran componentes discretos.
- El uso de un lenguaje de descripción de hardware provee independencia respecto al modelo de FPGA utilizada.

III. HERRAMIENTA DE GENERACIÓN DE ENCLAVAMIENTOS

A. Problemática y solución propuesta

Comúnmente, un enclavamiento queda especificado a través de un documento denominado “tabla de control”, que indica las condiciones de seguridad que deben cumplirse para activar cada ruta o señal [18]. A partir de la tabla se realiza el modelado del sistema, mediante alguna herramienta formal de simulación o chequeo de modelos, y se verifica el cumplimiento estricto de la lógica de seguridad para todos los casos posibles [19].

Todos los sistemas de enclavamiento son diferentes entre sí en cuanto a topología y complejidad, por lo que repetir el proceso de especificar, simular, implementar y verificar cada caso mediante herramientas *ad-hoc* es muy costoso en términos de tiempo.

Por ese motivo, para este proyecto se diseñó e implementó una herramienta denominada RAILIB (del inglés *RAilway Interlocking LIBrary*), que resuelve este proceso de forma flexible, acelerando la implementación de enclavamientos electrónicos.

Para esto, la herramienta consta de dos partes, que se indican a continuación y serán desarrolladas en las subsecciones posteriores:

1. Conjunto de bibliotecas de objetos de enclavamiento
2. Generador automático de código

B. Bibliotecas de Objetos de Enclavamiento

Las bibliotecas de objetos de enclavamiento usan el enfoque de la programación orientada a objetos [20] para describir e instanciar, mediante plantillas o clases, todos los bloques que componen un enclavamiento [21].

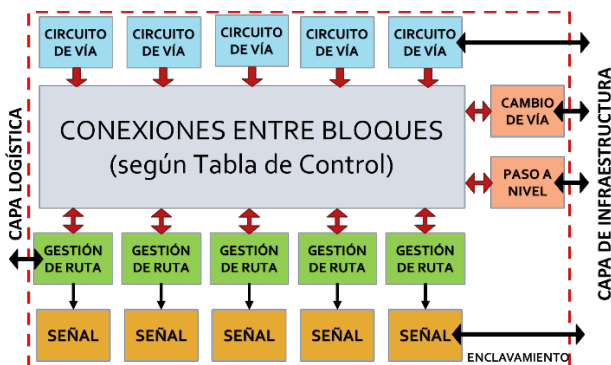


Fig. 2 – Diagrama en bloques de enclavamiento

La Fig. 2 muestra cómo se interrelacionan estos elementos según la tabla de control que da origen al código. Cada bloque u objeto internamente implementa la lógica vital que controla el cumplimiento de las condiciones de seguridad. De esta forma, el comportamiento global del sistema queda determinado por la interrelación entre sus bloques constitutivos.

C. Generador Automático de Código.

Como se muestra en la Fig. 3, el generador automático de código puede leer una tabla de control y generar el código asociado, en el lenguaje correspondiente a la plataforma final de implementación (C, Python, VHDL). El programa generado, a su vez, se complementa con las bibliotecas de objetos, que son estáticas.

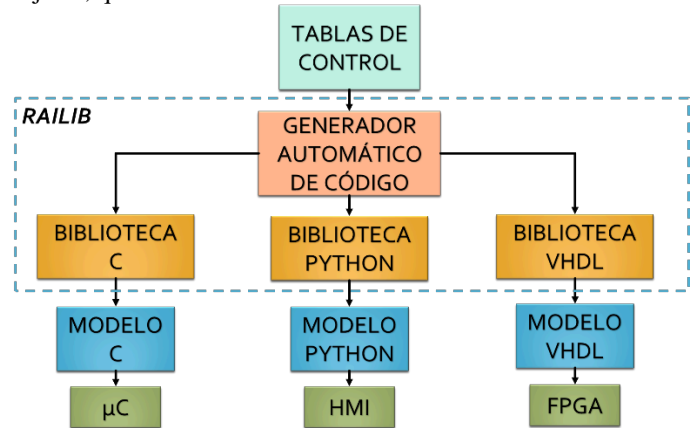


Fig. 3 – Esquema de uso de la herramienta RAILIB

IV. RESULTADOS

A. Ensayos realizados

La herramienta fue puesta a prueba simulando casos típicos de enclavamiento ferroviario, por ejemplo, vías dobles con cambios y pasos a nivel hasta playas de maniobra y vías de sobrepaso. Se comprobó el funcionamiento del generador automático de código y las bibliotecas de objetos, ambos escritos tanto en lenguaje C como Python. También se está desarrollando la biblioteca de objetos en VHDL, sobre la cual se han realizado hasta ahora sólo pruebas preliminares.

V. CONCLUSIONES

El enfoque propuesto en este trabajo aporta varias ventajas respecto al modelado *ad-hoc* de enclavamientos:

1. *Parte de un documento estándar* de requisitos (tabla de control) que asegura que la solución generada cumple las especificaciones del cliente.
2. *Reduce el tiempo de desarrollo*, ya que automatiza la implementación y provee una interfaz de abstracción con llamadas y funciones simplificadas.
3. *Reduce el esfuerzo de certificación*, pues en lugar de recertificar cada nuevo diseño solo se requiere certificar la biblioteca de bloques y la herramienta de generación automática de código [22].
4. *Facilita el agregado de nuevas funcionalidades*, dado que solo se requiere extender la clase asociada a un objeto dado y, si es necesario, modificar el generador de código.

VI. REFERENCIAS

- [1] Estadísticas ferroviarias relevadas por la Comisión Nacional de Regulación del Transporte (CNRT): <https://www.argentina.gob.ar/cnrt/estadisticas-ferroviarias>
- [2] Mapa comparativo de líneas ferroviarias existentes en distintos países: <https://www.indexmundi.com/map/?v=113&l=es>
- [3] Trackguard Westrace Mk II. (2014) Flexible safety processor. [Online]. Disponible: <https://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/en/rail-solutions/rail-automation/electronic-interlockings/trackguard-westrace-mk2-en.pdf>
- [4] Trackguard WESTLOCK. (2013). Cost-effective, highly available and reliable. [Online]. Disponible: <https://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/en/rail-solutions/rail-automation/electronic-interlockings/trackguard-westlock-en.pdf>
- [5] Alstoms Smartlock equipment at the heart of modular signalling project in the UK (2011). [Online]. Disponible: <https://www.alstom.com/press-releases-news/2011/3/Alstom-Smartlock>
- [6] Computer based interlocking (2019). [Online]. Disponible: <http://sts.hitachirail.com/en/products-services/business-segments/computer-based-interlocking>
- [7] Systémy pro kolejovou dopravu (2019). Elektronické Středlo TYP ESA® 33 [Online]. Disponible: <https://www.azd.cz/admin-data/storage/get/198->
- [8] Route Control Systems (2019). LockTrac 6111 ESTW L90 [Online]. Disponible: <https://www.thalesgroup.com/en/route-control-systems>
- [9] EBI LOCK (2014). Computer-Based Interlocking System [Online]. Disponible: http://www.infrasig.net/media/88880/pub-00074_ebi_lock_bro_prof_print_english.pdf
- [10] Bombardier Transportation (2015). Rail Control Solutions (RCS) [Online]. Disponible: http://www.irse.org/knowledge/publicdocuments/Bombardier%201%20IRSE_BT_RCS_Introduction_%20November%202015%20final.pdf
- [11] Railway Signaling Solutions (2019). Interlocking Equipment [Online]. Disponible: <https://www.kyosan.co.jp/english/product/signal03.html>
- [12] Kyosan history (2019). [Online]. Disponible: <https://www.kyosan.co.jp/english/company/history02.html>
- [13] Hima, Rail references (2019). COTS Rail Applications [Online]. Disponible: <https://www.hima.com/en/industries-solutions/cots-rail-references>
- [14] Reglamento Interno Técnico Operativo (RITO) de los Ferrocarriles del Estado Argentino.
- [15] International Electrotechnical Commission, “IEC 61508-1 : Functional safety of electrical/electronic/programmable electronic safety-related devices – part 1 : General requirements”, section 7.6.2.9 (Safety Integrity Level)
- [16] R. Dobias, H. Kubatova, “FPGA Based Design of the Railway’s Interlocking Equipments”
- [17] P. Kawalek, M. Rzyzko, “Modern methods in railway interlocking algorithms design”, ACM Journal of Microprocessor & Microsystems, Volume 44, Issue C, 2016
- [18] Institution of Railway Signal Engineers, “Study Guide for Module 3 – Signalling Principles”, Chapter 14 : Interlocking Control Tables.
- [19] K. Winter, N.J. Robinson, “Modelling Large Railway Interlockings and Model Checking Smaller Ones”.
- [20] M. Weisfeld, “The Object-Oriented Thought Process”, Addison-Wesley, 2009, 3rd edition.
- [21] F. van Dijk, W. Fokink, G. Kolk, P. van de Ven, B. van Vlijmen, “EURIS, a Specification Method for Distributed Interlockings”
- [22] HIMA, “TÜV-Tested and Approved BMS Function Blocks for Safety and Reliable Control of Fossil Fuel Burners”.

Sistema de supervisión de la seguridad del material ferroviario utilizando patrones de diseño

Ivan Mariano Di Vito, Pablo Gomez, Ariel Lutenberg
CONICET - GICSAFe
Laboratorio Sistemas Embebidos
Facultad de Ingeniería - UBA
Buenos Aires - Argentina
lse@fi.uba.ar

Adrian Laiuppa
CONICET - GICSAFe
Departamento de Electrónica
UTN-FRBB
Bahía Blanca, Argentina
alaiuppa@frbb.utn.edu.ar

Resumen—En las formaciones ferroviarias los sistemas de seguridad actúan cortando la tracción y aplicando el freno de emergencia. Las formaciones así detenidas generan problemas de logística en el sistema ferroviario. En este trabajo se presenta el diseño de un sistema que ante esta situación permite mover a la formación en un modo limitado y controlado hasta un taller para que sea reparada, o a un lugar adecuado para que desciendan los pasajeros. Dada la naturaleza crítica de este sistema el mismo fue diseñado siguiendo recomendaciones de la norma UNE-EN 50126, como el uso de patrones de diseños de hardware utilizados en otras industrias, como la aeroespacial, aeronáutica o automotriz.

Index Terms—sistemas críticos, patrones de diseño, sistemas ferroviarios, UNE-EN 50126

I. INTRODUCCIÓN

En el sistema ferroviario de la República Argentina un número considerable de locomotoras poseen sistemas de seguridad como el que se ilustra en la Fig. 1. En ellos una serie de subsistemas controlan la ausencia de errores en ciertas funciones específicas. Ejemplo de estos son el subsistema que monitorea el correcto funcionamiento de las puertas, el sistema de hombre vivo y el monitoreo con cámaras de seguridad. Cada uno de estos subsistemas informan su estado a un sistema central. Ante un error en cualquiera de esos subsistemas, el sistema central activa un mecanismo de seguridad por el cual detiene a la formación cortando la tracción y/o activando el freno de emergencia.

Esta forma de responder a los fallos implica que una sola formación con un problema menor puede causar la congestión de todo el sistema ferroviario. Sin embargo, en muchos casos el conductor podría llevar la formación a un lugar en donde no afecte la circulación y, en el caso de un servicio de pasajeros que además sea adecuado para que estos desciendan de la formación. La única manera en la que el sistema actual permite realizar esta operación es a través de la anulación total de los sistemas de seguridad. Este modo de funcionamiento se conoce como modo aislado total. Esto soluciona el problema logístico pero compromete seriamente la seguridad.

El sistema embebido diseñado se llama SAL/T (Sistema de Aislamiento Limitado/Total). En la Fig. 1 se puede ver como interactúa con los sistemas de seguridad de la formación. En una situación de falla de un subsistema, luego de la

activación del sistema de seguridad, este equipo permite al maquinista activar un modo de funcionamiento limitado del material rodante conocido como modo aislado limitado. En esta situación el SAL/T toma el control de las señales de corte de tracción y freno de emergencia mediante los relés K_2 y K_4 permitiendo al conductor mover la formación siempre que no se supere una velocidad máxima y que exista una señal de permiso de circulación emitida por la central operativa del ramal. Si estas condiciones no se cumplen el equipo utiliza los relés K_1 y K_3 para reducir la velocidad de la formación o detenerla, de manera controlada, según el caso.

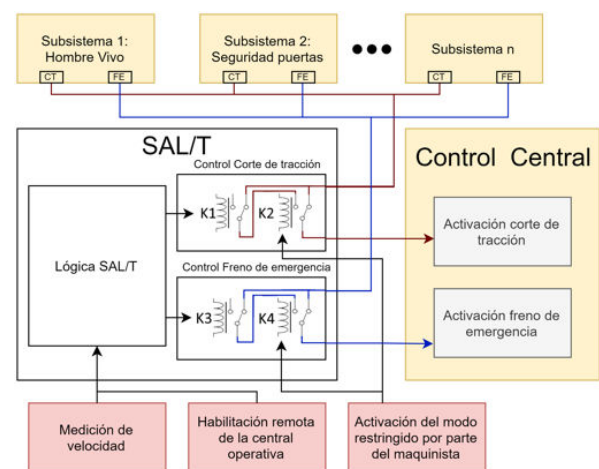


Figura 1. Sistema de seguridad del material rodante con SAL/T incorporado.

El SAL/T está diseñado para ser instalado en la locomotora, sobre el pupitre, para que el conductor pueda activar el modo aislado limitado si fuera necesario. Mientras este activado el SAL/T muestra la velocidad medida por el material rodante así como su estado a través de indicadores LED. Otras funciones que debe cumplir es reportar a la central operativa el estado de la formación. Esto incluye una indicación de modo aislado activado/desactivado y la velocidad a la que circula el material rodante.

II. DISEÑO DEL SISTEMA

Para el diseño del sistema se siguieron las primeras cinco fases planteadas en la norma UNE-EN 50126 [3]. Estas fases consisten en implementar y documentar una serie de pasos que incluyen una definición conceptual del proyecto, una definición de las políticas RAMS (Fiabilidad, Disponibilidad, Mantenibilidad y Seguridad), un análisis de riesgos y por último la definición de los requisitos del sistema y los subsistemas que lo componen.

Existen una serie de desafíos asociados al diseño del SAL/T. El mismo debe cumplir ciertas funciones de forma segura pero sin descuidar la disponibilidad del sistema. Para esto se decidió aplicar en los subsistemas críticos patrones de diseño de hardware que mejoren las características deseadas [4].

Es fundamental para el correcto funcionamiento del sistema tener una medición confiable de la velocidad del material rodante. Un error en esta medición puede generar que el sistema permita una condición insegura. Por esto se decidió usar el patrón de diseño de *hardware dúplex heterogéneo* [6]. El mismo consiste en utilizar más de un circuito de hardware con idéntico objetivo de forma tal de evitar errores de modo común. El diseño contempla el uso de tres fuentes de información independientes: la referencia de velocidad medida por el propio equipo de seguridad del material rodante, un contador de pulsos conectado a un tacómetro y un sistema de GPS. El SAL/T utiliza estas tres fuentes de información para buscar inconsistencias en las mediciones y así evitar condiciones inseguras.

Otro desafío en el diseño es la actuación segura sobre las señales críticas del material rodante. Dado que las mismas controlan el corte de tracción y el freno de emergencia, un error en su operación conlleva un gran riesgo. Para mitigar esto se decidió aplicar el patrón de diseño *canal simple protegido* [6]. El mismo consiste en que un sistema que actúa sobre una señal, en simultáneo mida el efecto de su accionar con el objetivo de comprobar que se logró el efecto deseado.

El tercer desafío es la necesidad de contar con una conexión inalámbrica entre el material rodante y la central operativa de manera fiable. Pueden generarse problemas logísticos si un equipo, debido a un error de conexión, no recibe la señal de permiso que habilita el modo aislado. Es un requisito del sistema que esta conexión se realice a través de la red celular con la tecnología GPRS. Por esto se decidió utilizar nuevamente el patrón de diseño de *hardware dúplex heterogéneo* [6]. En este caso se implementaron dos conexiones en paralelo con diferentes proveedores de servicio para minimizar el riesgo a que existan zonas sin cobertura.

III. IMPLEMENTACIÓN DE HARDWARE

En la etapa de diseño de hardware se definió que el sistema cuente con dos placas. La placa principal contiene todas las funciones de lógica, comunicación y medición y accionamiento seguro sobre las señales del material rodante. La placa secundaria se encarga de la interfaz hombre máquina (HMI). En la Fig. 2 se pueden ver ambas placas.

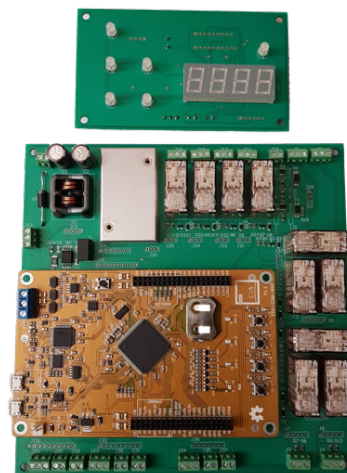


Figura 2. Placas electrónicas que componen el SAL/T.

El diseño de los subsistemas planteados se realizó siguiendo las especificaciones determinadas en las etapas de diseño. Para la actuación sobre las señales críticas se decidió utilizar relés de seguridad TE Connectivity V23047A1024A501. Los mismos, además de estar certificados para aplicaciones críticas, constan de dos pares de contactos vinculados mecánicamente. Esto habilita aplicar el patrón de *canal simple protegido* [6] al utilizar un par de contactos para accionar sobre la señal crítica y el otro para medir si efectivamente el circuito está funcionando correctamente. Esto habilita que el firmware reaccione ante errores en el hardware protegiendo la integridad del sistema. En las señales particularmente críticas como el corte de tracción y frenado de emergencia este circuito de relés está redundado siguiendo el patrón de diseño de *hardware dúplex homogéneo* [6]. Este consiste en duplicar un circuito con una función crítica de forma tal que la misma se sigue cumpliendo aunque uno de los dos circuitos falle.

Para cumplir con la especificación de conexión GPRS redundada se utilizaron dos módulos SIM 808. El mismo utiliza un chip GSM para conectarse a la red celular. Esto permite utilizar el servicio de dos empresas distintas mejorando la disponibilidad de la conexión. A la vez cada uno de estos módulos cuentan con funcionalidad GPS brindando redundancia en la medición de la velocidad del sistema.

IV. CONCLUSIÓN

En esta trabajo se presentan las principales características del diseño de un sistema de supervisión de la seguridad del material ferroviario. El mismo se basa en las etapas planteadas en la norma UNE-EN 50126 y en la aplicación de patrones de diseño de hardware para mejorar las características RAMS del sistema. Aplicar esta metodología lleva a un diseño que permite su uso en sistemas críticos como los asociados a la industria ferroviaria.

AGRADECIMIENTOS

Este trabajo fue financiado parcialmente mediante el subsidio UBA-PDE N°8 2018.

REFERENCIAS

- [1] Humberto Luiz Valdivia de Matos, Adilson Marques da Cunha, Luiz Alberto Vieira Dias. "Using design patterns for safety assessment of integrated modular avionics". 2014.
- [2] Tiago Amorim, Helmut Martin, Zhendong Ma. "Systematic Pattern Approach for Safety and Security Co-engineering in the Automotive Domain". 2017.
- [3] UNE-EN 50126-1. "Aplicaciones ferroviarias. Especificación y demostración de la fiabilidad, la disponibilidad, la mantenibilidad y la seguridad (RAMS). Parte 1: Requisitos básicos y procesos genéricos". 2005.
- [4] Bruce Powel Douglass. "Real-Time Design Patterns: Robust Scalable Architecture for Real-Time Systems". Addison Wesley. 2002.
- [5] Christopher Preschern, Nermin Kajtazovic, Christian Kreiner. "Building a Safety Architecture Pattern System". 2015.
- [6] Ashraf Armoush. "Design Patterns for Safety-Critical Embedded Systems". RWTH Aachen University, 2010.
- [7] Dulcinea Penha, Gereon Weiss, Alexander Stante. "Pattern-Based Approach for Designing Fail-operational Safety-Critical Embedded Systems". 2015.
- [8] Christopher Preschern, Nermin Kajtazovic, Christian Kreiner. "Catalog of Safety Tactics in the light of the IEC 61508 Safety Lifecycle". 2013.

Sistema de ensayos de relés ferroviarios de seguridad basado en computación en la nube

G. Ramoscelli
CONICET - GICSAFe
Depto. de Ing. Eléctrica y
de Computadoras - UNS
Bahía Blanca, Argentina
ramoscel@uns.edu.ar

A. Laiuppa, B. Palacios, M. Amado
CONICET - GICSAFe
Departamento de Electrónica
UTN-FRBB
Bahía Blanca, Argentina
alaiuppa@frbb.utn.edu.ar

L. Francucci, D. Balaña
CONICET - GICSAFe
Investigación & Desarrollo
Vortex
Mar del Plata, Argentina
info@vortexmakes.com

A. Permingeat, C. Mancón
CONICET - GICSAFe
Investigación & Desarrollo
Detecap / VSATMotion
Mar del Plata, Argentina
ap@detecap.com

A. Lutenberg
CONICET - GICSAFe
Laboratorio de Sistemas Embebidos
Facultad de Ingeniería - UBA
Bahía Blanca, Argentina
lse@fi.uba.ar

Resumen—En este trabajo se presentan mejoras en la arquitectura de un sistema automático para validar el ciclo de vida de relés ferroviarios de seguridad mediante la utilización de computación en la nube. Estos cambios simplifican en el desarrollo y disminuyen el mantenimiento del sistema. También se muestra el desarrollo un componente de *software* para cargar, visualizar y realizar el monitoreo de una máquina de estados finita. Este componente puede reusarse en otros proyectos similares.

Para mejorar la fiabilidad del sistema, se utilizaron herramientas visuales de alto nivel para generar parte del código del *firmware*. Esta forma de trabajo disminuye el riesgo de errores de código escrito *ad-hoc*. Finalmente, para mejorar la disponibilidad de las comunicaciones, se agregaron módulos GPRS.

Index Terms—Patrones de diseño, generación automática de código, computación en la nube, programación visual

I. CONTEXTO Y MOTIVACIÓN

El sistema ferroviario de la Argentina depende mayormente de componentes electromecánicos. Algunos de los sistemas críticos ferroviarios son los pasos a nivel con barreras automáticas y los cambios de vías. En estos sistemas un elemento importante son los relés que forman parte de la lógica de accionamiento y control. Para alcanzar los altos niveles de fiabilidad necesarios, los relés deben estar certificados bajo normas estrictas, como UNE-EN 50578.

En el CASE 2018 nuestro grupo presentó el trabajo “Sistema automático para ensayos de ciclo de vida de relés ferroviarios de seguridad” [1]. Allí se muestra el diseño y desarrollo de un sistema que funcionará de forma ininterrumpida durante varios meses o incluso un año. Teniendo en cuenta que un relé certificado cuesta más de US\$ 1.000 (dólares estadounidenses), que cada año el sistema ferroviario argentino compra de miles de relés de este tipo y que se consiguen únicamente por importación, sería muy conveniente el desarrollo de una industria nacional que fabrique estos relés.

Debido a la importancia de la certificación de los relés ferroviarios, es conveniente desarrollar un sistema automático de ensayos de relés. Por esto la empresa estatal Trenes Argentinos le encomendó el desarrollo de este sistema al grupo de investigación CONICET-GICSAFe.

El diseño del sistema automático de ensayo de relés se basa en la norma UNE-EN 50578. Según esta norma, la vida útil mínima de un relé de seguridad es de diez millones ($10 \cdot 10^6$) de ciclos sin carga. Para asegurar esto, la norma establece valores máximos y mínimos en las variaciones de los valores iniciales de la corriente de excitación, la corriente de caída y el factor K. Teniendo esto en cuenta, se diseñó el sistema de ensayo para medir estos valores eléctricos.

II. CAMBIOS EN LA ARQUITECTURA

La arquitectura presentada en [1] fué reemplazada por una nueva arquitectura, que se muestra en la figura 1. Se observa el reemplazo en el nivel de monitoreo de la aplicación de computadora por un conjunto de aplicaciones en la nube [2]. Otro cambio importante mostrado en la figura es el agregado de módulos GPRS para mejorar la disponibilidad de las comunicaciones entre nivel de excitación y registro, y el nivel de monitoreo. Este módulo se conecta al puerto serie de la placa CIAA y se usa en caso de falla de la red *ethernet*.

III. USO DE COMPUTACIÓN EN LA NUBE

En este trabajo el término *nube* se refiere a un conjunto de servidores físicos junto con los recursos de *software* necesarios para facilitar la programación y el desarrollo de aplicaciones. El acceso a estos servidores sólo es posible desde *Internet*. Por otro lado, *computación en la nube* [2] se refiere al estilo y al flujo de trabajo usados en el desarrollo de las aplicaciones que aprovechan los recursos de la *nube*.

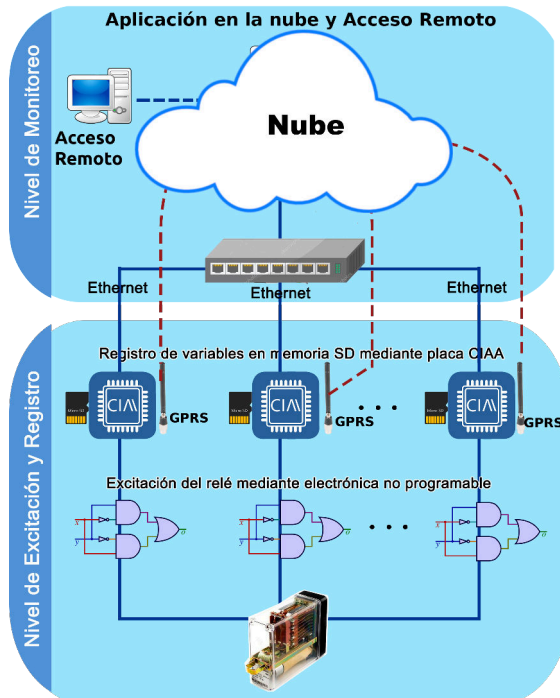


Figura 1. Nueva arquitectura del sistema.

En este proyecto los servidores y los recursos de *software* son los que posee Trenes Argentinos. Los recursos utilizados son: Servicio MQTT [3], Base de Datos *PostgreSQL* [4], herramienta de programación *Node-RED* [5] y un servicio *Web* con un intérprete de lenguaje PHP [6].

En varios proyectos del CONICET-GICSAFe se hace uso de máquinas de estado. Por esto se desarrolló un componente de *software* para almacenar modelos de máquinas de estados finitas en la nube a partir de los datos que ingresa el usuario. El subsistema visual de este componente muestra el diagrama de estados al momento de la carga de los datos.

La función de transición $\delta : Q \times \Sigma \rightarrow Q$ correspondiente a una máquina de estados finita se deduce a partir de los datos ingresados por el usuario, donde Q es el conjunto de estados y el alfabeto de entrada Σ son las transiciones de estado. Al terminar la carga de los datos, se genera una representación interna de la máquina de estados para monitoreo y detección de fallas o anomalías.

La representación interna que se usa para el monitoreo de la lógica no programable es una lista de adyacencia. Hasta ahora ha sido suficiente el uso de listas de adyacencias para representar las máquinas de estados finitas usadas en los proyectos en curso. Sin embargo si surgen casos en los que esta representación no alcanza, se emplearán lógicas temporales para esta representación interna [7].

Desde que arranca el sistema y hasta la conclusión de los ensayos, la aplicación en la nube supervisa que los cambios de estado respeten la función de transición δ . En caso de producirse una anomalía, enviará avisos por correo electrónico. Esta mejora evita la escritura de código *ad-hoc* para realizar esta

tarea, disminuyendo la posibilidad de errores de programación.

Además de la máquina de estados, la aplicación también supervisa que las medidas eléctricas sensadas del relé no se desvíen de los intervalos de valores fijados. Si hubiera alguna desviación, se dará aviso por correo electrónico. Toda esta información se muestra en tiempo real en la computadora o el dispositivo conectado en forma remota.

Otro cambio en el sistema de monitoreo es el acceso multiusuario mediante identificación y contraseña. El administrador puede asignar distintos roles a cada usuario. Para desarrollar la parte de *backend* de la aplicación se optó por el entorno *Laravel* [9]. Para el *frontend* se usan varias librerías, como *Vue.js* [10], junto con patrones de diseño *web* adaptativo para lograr una visualización óptima desde una computadora o desde un dispositivo móvil [11].

IV. MEJORAS EN EL *Firmware* DE LA CIAA

En el la codificación del *firmware* de la CIAA [8] se usó un desarrollo guiado por pruebas [12] y el entorno *RKH*. Se usó el entorno de la compañía *Yakindu* [14] y un plugin de resiente desarrollo [13] para generar automáticamente el código del *firmware* a partir de diagramas de máquinas de estados en UML, evitando errores de escritura de código.

Otras mejoras son la posibilidad de recibir comandos desde el servidor remoto mediante mensajes MQTT, el uso del sistema operativo *FreeRTOS* y la generación de los comando AT para el manejo del módulo GPRS. En la figura 2 se muestra este módulo acoplado al puerto serie de cada una de las tres placas CIAA en el momento del montaje final. Cada módulo de excitación y registro posee triple redundancia para alcanzar los niveles de disponibilidad y mantenibilidad esperados. Esto se observa en las figuras 1 y 2.

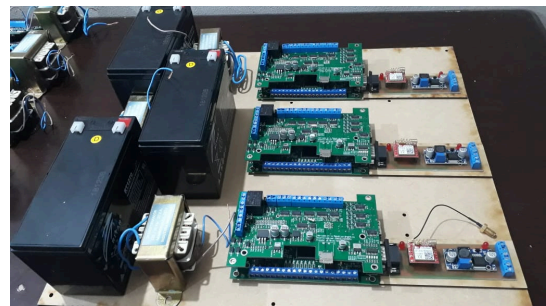


Figura 2. Montaje de la etapa de excitación y registro con el módulo GPRS.

V. CONCLUSIONES

Los cambios mostrados significaron una nueva arquitectura del sistema que permitió disminuir los costos asociados al diseño, desarrollo y mantenimiento. Además se presentaron mejoras en el desarrollo del *firmware*. Estas mejoras aprovechan una gama de herramientas visuales y de recursos basados en la nube que optimizaron el flujo del desarrollo respecto al trabajo previamente publicado en [1].

REFERENCIAS

- [1] (2018) A. Laiuppa, M. Amado, J. M. Cruz, F. Larrosa, E. Irrazabal, I. Sambrana, S. H. Gallina, M. A. Gomez Lopez, A. Lutenberg, 'Sistema automático para ensayos de ciclo de vida de relés ferroviarios de seguridad', Libro de Trabajos del CASE2018, Congreso Argentino de Sistemas Embebidos, Córdoba, Argentina. (2018). [Online] Disponible: <http://www.sase.com.ar/case18/>
- [2] ¿Qué es informática en la nube?. [Online]. Disponible: <https://aws.amazon.com/es/what-is-cloud-computing/>
- [3] ¿Qué es MQTT?. [Online]. Disponible: <https://mqtt.org/faq>
- [4] PostgreSQL. [Online]. Disponible: <https://es.wikipedia.org/wiki/PostgreSQL>
- [5] Node-RED. [Online]. Disponible: <https://en.wikipedia.org/wiki/Node-RED>
- [6] ¿Qué es PHP?. [Online]. Disponible: <https://www.php.net/manual/es/intro-what-is.php>
- [7] (2018) E. Bartocci, Y .Falcone, A. Francalanza, G. Reger. 'Introduction to Runtime Verification'. [Online]. Disponible: https://www.researchgate.net/publication/323082969_Introduction_to_Runtime_Verification
- [8] (2018) Proyecto CIAA. [Online]. Disponible: http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:hardware:ciaa_nxp:ciaa_nxp_inicio
- [9] Laravel. [Online]. Disponible: <https://es.wikipedia.org/wiki/Laravel>
- [10] Vue.js. [Online]. Disponible: <https://en.wikipedia.org/wiki/Vue.js>
- [11] Diseño web adaptable. [Online]. Disponible: https://es.wikipedia.org/wiki/Dise~no_web_adaptable
- [12] Desarrollo guiado por pruebas (2019). [Online]. Disponible: https://es.wikipedia.org/wiki/Desarrollo_guiado_por_pruebas
- [13] RKH (2019). [Online]. Disponible: <https://www.vortexmakes.com/que-es/>
- [14] Yakindu (2019). [Online]. Disponible: <https://www.itemis.com/en/yakindu/state-machine/>
- [15] Integración continua. [Online]. Disponible: https://es.wikipedia.org/wiki/Integracion_continua

Diseño de un sistema modular para el monitoreo y gestión de sistemas ferroviarios

A. Laiuppa, B. Palacios, M. Amado
CONICET - GICSAFe
Departamento de Electrónica
UTN-FRBB
Bahía Blanca, Argentina
alaiuppa@frbb.utn.edu.ar

G. Ramoscelli
CONICET - GICSAFe
Depto. de Ingeniería Eléctrica
y de Computadoras - UNS
Bahía Blanca, Argentina
ramoscel@uns.edu.ar

L. Dórdolo
CONICET - GICSAFe
Laboratorio de Señalamiento
Línea Mitre, Trenes Argentinos
Buenos Aires, Argentina
lucas.dordolo@trenesargentinos.gob.ar

A. Permingeat, C. Mancón
CONICET - GICSAFe
Investigación & Desarrollo
Detecap / VSATMotion
Mar del Plata, Argentina
ap@detecap.com

L. Francucci, D. Baliña
CONICET - GICSAFe
Investigación & Desarrollo
Vortex
Mar del Plata, Argentina
info@vortexmakes.com

A. Lutenberg, P. Gomez, S. Germino
CONICET - GICSAFe
Laboratorio Sistemas Embebidos
Facultad de Ingeniería - UBA
Buenos Aires, Argentina
lse@fi.uba.ar

Resumen— En este trabajo se presenta un sistema para implementar soluciones a diversos problemas de aplicación ferroviaria, que surgen a partir de los requisitos del principal operador ferroviario de la República Argentina.

Keywords— Railway systems; Internet of Things; TDD

I. MOTIVACIÓN Y CONTEXTO

Este trabajo es la ampliación de un trabajo previo, referido a un sistema de monitoreo de barreras ferroviarias automáticas [1] realizado por nuestro grupo CONICET-GICSAFe [2] para la empresa Trenes Argentinos [3]. A partir del uso del sistema el cliente propuso el agregado de nuevas funcionalidades y nuevos proyectos, con requisitos similares a los del sistema de monitoreo remoto. En consecuencia se optó por rediseñar la arquitectura del sistema, empleando una estructura modular para que un conjunto acotado de módulos puedan ser reutilizados para resolver distintos problemas. En este trabajo se presentan los diversos proyectos y la solución adoptada.

II. DESCRIPCIÓN DE LOS PROBLEMAS A RESOLVER

El sistema de monitoreo remoto de barreras ferroviarias automáticas [1] reporta el estado de ciertas señales al centro de control ferroviario para que en caso de falla la barrera sea reparada inmediatamente. En la sección IV de [1] se explica que el sistema a partir del análisis de la información que obtiene puede detectar situaciones en las que un tren desocupa la vía y otro tren la ocupa inmediatamente. En estos casos el brazo sube, a pesar de que hay un tren aproximándose, y al llegar a la posición vertical baja rápidamente. En consecuencia el brazo al bajar podría ser destruido por el avance de un vehículo que hubiera comenzado a cruzar las vías. Dado que no es posible alterar el funcionamiento del sistema de la barrera para evitar que esto ocurra, el cliente solicitó incluir la

activación de una señal que indique que otro tren se aproxima, para advertir a conductores y peatones, y así evitar accidentes. Esto implica incorporar salidas de contacto seco al sistema.

El cliente también requirió que el sistema incorpore la capacidad de activar una cámara registradora certificada. Esta cámara debe activarse a partir de que el brazo de la barrera baja y por un intervalo máximo configurable en alrededor de diez minutos, de modo que si la barrera está funcionando adecuadamente registre a los vehículos que cruzan con la barrera baja. Así se labran infracciones de tránsito, con el objeto de disminuir los accidentes. Para implementar esta funcionalidad es necesario incluir otra salida de contacto seco.

Por otra parte, Trenes Argentinos propuso la realización de otro proyecto ligado a la visualización en tiempo real del sistema de señalamiento ferroviario. En la Fig. 1 se muestra un diagrama del ramal Tigre de la línea Mitre. Este tiene decenas de circuitos de vía, máquinas de cambio, señales, etc. La información del estado de esos elementos está disponible en siete cabinas (Fig. 1). Para adquirir esa información es necesario contar con muchas más entradas digitales que en [1]. Además las cabinas cuentan con conectividad vía Ethernet, por lo que se prefiere utilizar ese enlace para la comunicación.

En la Tabla I se presenta un resumen de las características de los sistemas a implementar. Se observa que existen entre ellos muchas similitudes, a partir de lo cual se propone el diseño que se presenta en la sección III de este trabajo.



Fig. 1. Esquema del ramal Tigre de la línea Mitre. Se indica la ubicación de las estaciones, las cabinas de señalamiento y su área de alcance.

TABLA I. CARACTERÍSTICAS DEL MONITOREO DE BARRERAS AUTOMÁTICAS Y DEL MONITOREO DEL SISTEMA SEÑALAMIENTO EN LAS CABINAS

Elemento	Monitoreo de barreras	Monitoreo en cabinas
Entradas digitales	Entre 8 y 12 señales	Entre 60 y 120 señales
Entradas analógicas	4 tensiones y 4 corrientes	4 tensiones y 4 corrientes
Salidas digitales	2 contactos secos	Ninguna
Comunicaciones	GSM con redundancia mediante GSM	Ethernet con redundancia mediante GSM
Alimentación	110/220 VAC + batería interna propia	220 VAC + batería interna propia

III. ARQUITECTURA PROPUESTA

En la Fig. 2 se presenta un diagrama de la arquitectura modular propuesta. Se observa que la misma está compuesta por un módulo de alimentación, una serie de módulos de entradas y salidas digitales, y un módulo central de procesamiento y comunicaciones, que incluye además a las entradas analógicas.

El módulo de alimentación tiene una entrada de 110/220 VAC, a partir de la cual carga una batería propia de 12 VDC y 7 Ah, mediante la cual el sistema logra una autonomía de al menos 48 horas.

Los módulos de entradas digitales se implementan en base a un registro de desplazamientos. Así mediante su conexión en cascada se puede aumentar en múltiplos de 16 la cantidad de entradas del sistema. Los módulos de salidas digitales se implementan también mediante registros de desplazamiento, pero en este caso provistos de 4 salidas a relé en cada módulo.

El microcontrolador del módulo central (FB 4.1) es un STM32F407VG, elegido por su memoria, capacidad de procesamiento y documentación disponible. El convertor DC/DC (FB 4.2) está basado en un LM2596, de bajo costo, alta disponibilidad y fiabilidad. La comunicación vía Ethernet se implementa con un nodemcu basado en esp8266, mediante el cual se establece una conexión Wi-Fi (FB 4.3) con un router Wi-Fi/Ethernet externo. Se eligió al esp8266 debido a su bajo costo, su buen desempeño, la documentación disponible y su extendido uso. El sistema consta además de dos módulos GSM, en un caso un SIM800 (FB 4.4) y en el otro un A6 (FB 4.5), debido a su gran difusión y bajo costo, y para tener diversidad en el hardware. Se incluye además una socket micro SD (FB 4.6) para almacenar eventos aún cuando no se disponga de conectividad, y un módulo RS-485 full duplex aislado (FB 4.7) para conectarse con otros equipos.

IV. DESARROLLO DE FIRMWARE Y MEJORAS INTRODUCIDAS

Para lograr un firmware fiable en un tiempo de desarrollo acotado, se usó *Test Driven Development* [4] junto con modelos de software, en especial máquinas de estados, cuya codificación se realizó automáticamente mediante el plugin RKH Generator [5] de Yakindu [6], usando un estándar de

codificación preestablecido. Gran parte del firmware fue diseñado de modo tal que sea multiplataforma, lo que permitió el desarrollo concurrente del hardware y el firmware.

Al sistema presentado en [1] se agregó la posibilidad de recibir parámetros de configuración y comandos desde el servidor remoto mediante mensajes MQTT. Se incorporó además al trabajo presentado en [1] la comunicación sobre TCP/IP mediante Ethernet, utilizando el sistema operativo de tiempo real FreeRTOS [7].

Se está trabajando además en la incorporación de análisis de cobertura de código, el uso de más de un analizador estático de código y la automatización de estas tareas mediante la integración continua.

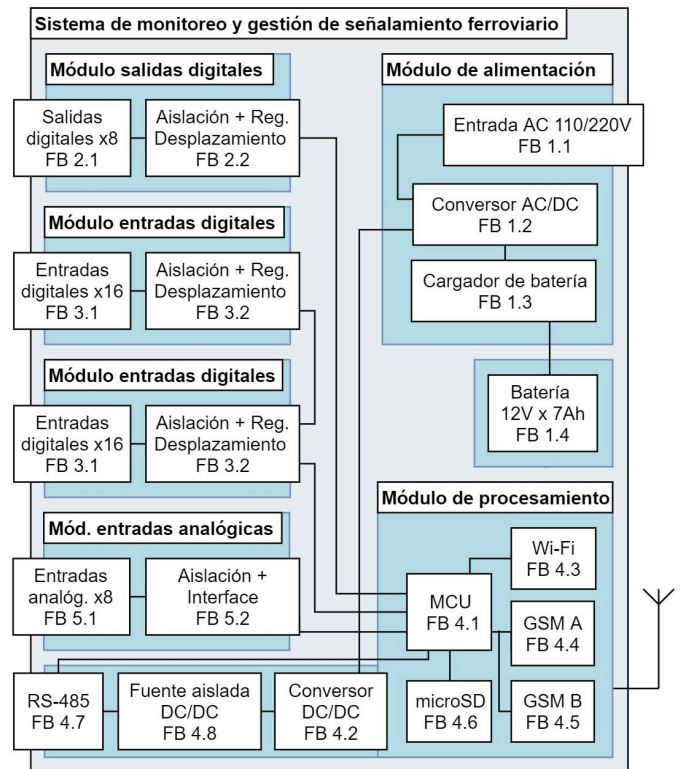


Fig. 2. Diagrama de la arquitectura modular propuesta. Se incluyen a modo ilustrativo todos los módulos de la arquitectura.

V. CONCLUSIONES

En este trabajo se presentó una arquitectura modular para el monitoreo y gestión de sistemas ferroviarios, junto con los requisitos y criterios a partir de los cuales se tomaron las decisiones de diseño correspondientes. Además se presentaron las herramientas utilizadas para el desarrollo del firmware y las mejoras introducidas respecto al trabajo previamente publicado en [1].

AGRADECIMIENTOS

Este trabajo fue financiado parcialmente mediante el subsidio UBA-PDE N°26.

REFERENCIAS

- [1] A. Laiuppa, M. Amado, S. H. Gallina, E. Irrazabal, I. Sambrana, H. Ferrari, D. Baliña, L. Francucci, M. A. Gomez Lopez, J. M. Cruz, P. Gomez, A. Lutenberg, 'Sistema de monitoreo remoto de barreras ferroviarias automáticas', Libro de Trabajos del CASE2018, Congreso Argentino de Sistemas Embebidos, Córdoba, Argentina. (2018).
- [2] CONICET-GICSAFe (2019). [Online]. Disponible: <http://www.conicet-gicsafe.com.ar>
- [3] Trenes Argentinos (2019). [Online]. Disponible: <https://www.argentina.gob.ar/transporte/trenes-argentinos>
- [4] 'Test-driven development concepts, taxonomy, and future direction', D. Janzen ; H. Saiedian, Computer, Vol. 35, Issue 9, pp. 43-50 (2005).
- [5] RKH (2019). [Online]. Disponible: <https://www.vortexmakes.com/que-es/>
- [6] Yakindu (2019). [Online]. Disponible: <https://www.itemis.com/en/yakindu/state-machine/>
- [7] FreeRTOS (2019). [Online]. Disponible: <https://www.freertos.org/>

Extendiendo la Cobertura del Servicio SMS GSM en Áreas Rurales mediante Tecnología LPWAN

Francisco Gimenez, Carlos Zerbini, Guillermo Riva

GInTEA, Universidad Tecnológica Nacional, Facultad Regional Córdoba (UTN-FRC)

frangimz@gmail.com, czerbini@frc.utn.edu.ar, griva@frc.utn.edu.ar

Resumen—Los operadores de telefonía móvil proveen servicios de comunicación en centros urbanos y a lo largo de rutas y su entorno. En el último caso, la cobertura puede alcanzar algunos kilómetros desde las rutas campo adentro según la tecnología disponible. En este sentido, los habitantes de zonas rurales tienen en su mayoría limitaciones o falta de servicio de telefonía móvil. Este trabajo propone una solución para extender la cobertura del servicio de mensaje corto basada en tecnologías de largo alcance y bajo consumo. La implementación considera la incorporación de dos equipos de comunicaciones llamados *nodo GSM* y *nodo remoto*, e involucra diferentes tecnologías de comunicaciones inalámbricas. A nuestro entender, esta es la primera implementación y evaluación de un sistema que permite extender la cobertura del servicio de mensajes de telefonía móvil en zonas rurales, basado en tecnología de radio de largo alcance y bajo consumo. Varias pruebas de comunicación fueron implementadas para evaluar la performance de la solución propuesta. Los resultados demuestran las ventajas de agregar tecnologías de radio de largo alcance y bajo consumo a los sistemas de telefonía móvil para extender el área de cobertura del servicio de mensajes.

Index Terms—IoT, LPWAN, LoRa, GSM, SMS, BLE

I. INTRODUCCIÓN

Este trabajo propone una solución para hacer frente al problema de la limitada o falta de cobertura de servicio de telefonía móvil en zonas rurales o lugares alejados de centros urbanos [1]. La propuesta se basa en el uso de la red GSM (Global System for Mobile Communication), debido a su amplia disponibilidad y al reducido costo del hardware comparado con los de cuarta generación (4G).

Para hacer frente a los problemas de conectividad, en los últimos años se han propuesto muchas soluciones para extender la cobertura de GSM. Soluciones en el despliegue de celdas con tecnologías de Software Defined Radio (SDR) [2], [3], que pueden utilizar estrategias como Community Cellular Networks (CCNs) y Cognitive Radio (CR) [4], [5], han sido propuestos en los últimos años. Por otro lado, se ofrecen soluciones para tener servicio GSM, basados en sistemas comerciales de amplificadores (boosters) o repetidores.

En muchos casos, solamente es necesario un sistema que permita enviar un mensaje de alerta o notificaciones ante algún determinado evento. En este sentido, es posible que el uso de estas nuevas tecnologías LPWAN tales como LoRa, Weightless, Sigfox, etc, combinadas con GSM para lograr establecer comunicaciones por SMS en zonas rurales.

El escenario analizado consiste en un usuario sin servicio de telefonía móvil que quiere enviar mensajes de texto de tipo SMS (Short Message Service). Para hacer esto posible, se utiliza una aplicación en el teléfono el cual actúa como interfaz para que el usuario pueda generar el mensaje. Luego, esta aplicación establece un enlace de corta distancia (por ejemplo a través de Bluetooth) entre el teléfono y un nodo denominado *nodo remoto*. Este último envía el mensaje en forma remota usando comunicación del tipo LPWAN (Low Power Wide Area Network) basada en tecnología LoRa (Long Range), enlazándolo con otro nodo distante denominado *nodo GSM*, localizado en una zona con servicio de telefonía móvil. Finalmente, el *nodo GSM* retransmite el mensaje al destino a través de la red de telefonía (Fig. 1), de manera totalmente transparente para el usuario.

Si bien existen equipos con conectividad LoRa que permiten el envío de datos a un servidor a través de distintos sistemas de comunicación (gateway), el sistema planteado en este trabajo provee de una interfaz completa y transparente al usuario para enviar mensajes de texto a un destinatario sin ninguna intervención adicional y con ventajas como bajo costo, bajo consumo y largo alcance. Por eso podemos afirmar que según nuestro conocimiento y con las tecnologías actuales, este trabajo es la primer implementación y evaluación de un sistema que permite extender el servicio SMS a zonas alejadas.

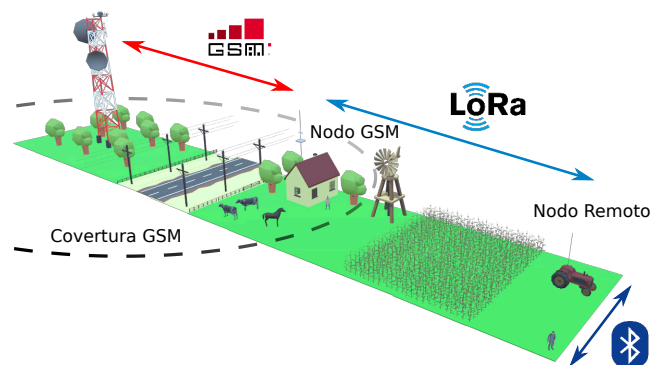


Figura 1: Esquema general de la solución propuesta.

II. DESARROLLO E IMPLEMENTACIÓN

La solución propuesta para extender la cobertura del servicio SMS combina tanto hardware como software, y tres sistemas de comunicación: GSM, LoRa y BLE. La secuencia de envío de un mensaje desde un usuario remoto es realizado a través de un smartphone conectado por Bluetooth, esta conformada por un *nodo remoto* como un puente entre Bluetooth y LoRa para enviar los mensajes a grandes distancias, un *nodo GSM* que actúa como un Gateway entre el enlace LoRa y la red GSM. Finalmente, el Gateway es el encargado de enviar los mensajes al usuario final. A continuación se describen todas las partes desarrolladas.

II-A. Interfaz de usuario remoto

El usuario interactúa con el sistema a través de una interfaz gráfica en el smartphone, en el que define el número de teléfono de destino y el contenido del mensaje. Mediante la aplicación *BLE Scanner* de Android, se establece el enlace entre el teléfono y el nodo remoto. La estructura del mensaje de BLE está compuesto por 10 caracteres que representan el número de teléfono de destino, y por 150 caracteres disponibles para escribir el mensaje [6]. Se implementó la estructura del mensaje para no superar los 160 caracteres de 7 bits, a fin de evitar el uso de mensajes concatenados [7]. Para evitar errores se utiliza un control de sintaxis de dicha estructura de entrada y de esta manera garantizar la transferencia exitosa del mensaje.

II-B. Nodo remoto

El *nodo remoto* está compuesto por un módulo *PyCom LoPy* con su placa de expansión y antena [8]. Este módulo se programa en MicroPython, una implementación simplificada del lenguaje Python 3, optimizada para ejecutarse en microcontroladores [9].

El *nodo remoto* establece una interfaz, por un lado, con el teléfono del usuario a través de BLE, y por otro lado con el *nodo GSM* a través de LoRa. Si se requiere de mayor cobertura, se puede incrementar la ganancia y altura de la antena, o se pueden utilizar múltiples saltos de comunicación para alcanzar al *nodo GSM*.

Para la implementación con BLE, se tuvo en cuenta que el tamaño máximo del mensaje y del buffer es de 20 bytes o caracteres [10], por lo tanto, el módulo *LoPy* recibe paquetes de datos de 20 bytes y los reenvía al *nodo GSM* vía LoRa.

Para la comunicación LoRa se utilizó el modo *LoRa-MAC*, un spreading factor (SF) de 11, y un ancho de banda (BW) de 125 kHz, con lo cual se obtiene una sensibilidad de recepción de -135 dBm a una velocidad de 440 bps [11]. No se realizaron pruebas con un valor de SF=12, debido a que el tiempo de transmisión es elevado, se presentan problemas en la recepción, y que en ciertas regiones no está regulada su utilización [12].

II-C. Nodo GSM

El *nodo GSM* es instalado en zonas con cobertura, y está compuesto por dos módulos, un *LoPy* de *Pycom*, y otro *GSM*

Posición	Señal a h_{r1} [dBm]	Señal a h_{r2} [dBm]	Distancia [m]
1	-81	-64	108
2	-88	-65	211
3	-122	-111	1720
4	sin señal	-115	2492

h_{r1} a nivel del suelo
 h_{r2} a 2 metros de altura

Tabla I: Mediciones de niveles de señal LoRa.

que cumple la función de Modem, ambos conectados entre sí a través de una interfaz serial SPI bidireccional, actuando el *LoPy* como maestro y el módulo *GSM* como esclavo. La función de este nodo es la de recibir los datos del nodo remoto a través de LoRa, procesarlos y transmitirlos a través del modem *GSM* a la red móvil. Cada uno de los bloques implementa en código las funcionalidades requeridas para verificar el estado de la red, y establecer una comunicación estable.

II-D. Usuario final

El usuario final recibe el mensaje enviado desde el teléfono remoto, a través del nodo remoto y del nodo *GSM*.

Además del mensaje, se agrega información sobre el nivel de señal con la que recibe el nodo *GSM* desde el nodo remoto a través de la comunicación LoRa. Esto es útil para realizar las mediciones que se verán en la sección III.

III. MEDICIONES DE CAMPO

Para evaluar la propuesta, se realizaron diferentes ensayos en el campus de la UTN Córdoba. El nodo *GSM* fue colocado a una altura de 5 metros, y el nodo remoto fue posicionado a diferentes distancias, a dos diferentes alturas, a fin de evaluar el efecto del despejamiento en la comunicación. Las mediciones obtenidas durante los ensayos se muestran en la Tabla I.

Se puede ver en que medida la señal es atenuada cuando la distancia entre ambos nodos aumenta, y que las distancias de comunicación pueden ser extendidas más allá de los ensayos realizados, ya que se cuenta con márgenes de sensibilidad elevados. Es decir, a 2500 metros de distancia se tienen niveles de señal del orden de -115 dBm, cuando la sensibilidad en la configuración de radio utilizada es de -136 dBm, o sea, con un margen de enlace de 21 dB.

IV. CONCLUSIONES

En este trabajo se describe el desarrollo e implementación de un sistema para extender el servicio SMS de *GSM* a través de comunicación LoRa. Se demuestra el beneficio de contar con estaciones que mediante la incorporación de tecnologías LPWAN como LoRa permiten extender en gran medida el área de cobertura del servicio de mensajes en zonas sin infraestructura de comunicaciones, con sistemas embebidos disponibles en el mercado local. Como trabajo futuro, se plantea el desarrollo del mecanismo de comunicación inverso para recibir mensajes en el teléfono remoto, y la mejora en la aplicación Android para el manejo de los mismos.

REFERENCIAS

- [1] G. Cruz, G. Touchard, "Enabling rural coverage regulatory and policy recommendations to foster mobile broadband coverage in developing countries," GSMA, 2018.
- [2] A. Anand, V. Pejovic, D. L. Johnson, E. M. Belding, "VillageCell: Cost Effective Cellular Connectivity in Rural Areas," In Proc. of the Fifth International Conference on Information and Communication Technologies and Development - ICTD 12, 2012.
- [3] K. Aggrawal, K. Vachhani, "Reconfigurable cellular GSM network using USRP B200 and OpenBTS for disaster-hit regions," 2017 IEEE 13th Malaysia International Conference on Communications (MICC), Johor Bahru, 2017, pp. 141-146, 2017.
- [4] K. Heimerl, K. Ali, J. Blumenstock, "Expanding Rural Cellular Networks with Virtual Coverage," 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI '13), pp. 283-296, 2013.
- [5] S. Hasan, K. Heimerl, K. Harrison, K. Ali, S. Roberts, A. Sahai, E. Brewer, "GSM Whitespaces: An Opportunity for Rural Cellular Service," 2013.
- [6] "Digital cellular telecommunications system (Phase 2+); Alphabets and language-specific information (GSM 03.38)," Tech. note, ETSI, 3GPP TS 23.038 version 10.0.0 Release 10, 2011.
- [7] "Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS)," Tech. note, ETSI, 3GPP TS 23.040 version 9.3.0 Release 9, 2010
- [8] Fred de Haro, Bettina Rubek Slater, "LoPy Specsheat." Pycom, www.pycom.io/wp-content/uploads/2018/08/lopy-specsheet.pdf. Último acceso 15 de Julio, 2019.
- [9] D. George, "The MicroPython language." MicroPython, <http://docs.micropython.org/en/latest/reference/index.html>. Último acceso 15 de Julio, 2019.
- [10] J. Tosi, F. Taffoni, M. Santacatterina, R. Sannino, D. Formica, "Performance Evaluation of Bluetooth Low Energy: A Systematic Review," MDPI Sensors Journal, 2017.
- [11] "AN1200.22 LoRa Modulation Basics," Technical Note, Semtech, 2015.
- [12] "LoRaWAN 1.1 Regional Parameters," LoRa Alliance, Revision: B, 2018.

Algoritmo para la mejora de las prestaciones de los flujos de datos utilizando Redes Centradas en Contenidos

Ignacio Navarrete¹, Diego Dujovne¹, y Guillermo Riva²

¹Escuela de Informática y Telecomunicaciones, Universidad Diego Portales, Santiago, Chile,

²GInTEA, National Technological University, Cordoba Regional Faculty, Córdoba, Argentina,

Resumen—Las Redes Centradas en Contenidos constituyen un nuevo paradigma como evolución de la arquitectura actual de Internet, utilizando la generalización de las redes P2P en un sistema productor-consumidor mediante la publicación de intereses. Un interés se genera a partir de un consumidor y debe llegar al productor, enviando el bloque de datos solicitado. Las propuestas actuales de Redes Centradas en Contenidos suponen el uso de nodos fijos, sin considerar la movilidad, por ejemplo, en redes de telefonía celular. El principal problema para un dispositivo móvil surge cuando publica un interés en un nodo de infraestructura y la respuesta llega a este mismo nodo cuando el móvil ya se ha trasladado al siguiente. El objetivo de este trabajo es proponer un algoritmo para mejorar la performance de la transmisión de flujos de datos en presencia de nodos móviles de usuarios finales, tales como teléfonos celulares. La solución considera llegar a las futuras posiciones por donde se desplace el consumidor móvil, a través de un módulo predictivo de la movilidad de consumidores. La mejora observada con respecto a la inexistencia del módulo, supone un aumento del 20 % del Caché Hit Rate.

Index Terms—information-centric networking, ICN, streaming

I. INTRODUCCIÓN

Las Redes Centradas en los Contenidos en la actualidad definen aquel sistema o conjunto de sistemas que utilizan tanto la carga de los paquetes de datos como la información sobre los archivos para direccionar y hacer más eficiente la transmisión de archivos a través de Internet. La idea se basa en la generalización de las comunicaciones Peer-to-Peer, desde la red periférica a toda la red de comunicación. Un ejemplo de este sistema es el proyecto CCNx [1], que planteó las bases de este nuevo paradigma para una Internet centrada en contenidos. La estructura de una red orientada a contenidos define la existencia de entidades productoras de contenidos y consumidoras de contenidos, que se encuentran diseminadas en una red de computadores y enrutadores. Los Consumidores generan intereses que difunden en la red para ubicar el contenido y poder recibirlo. El camino que recorren los paquetes de datos es el camino inverso al interés generado por el consumidor. Los elementos intermedios actúan como almacenamiento temporal (con capacidad limitada) de los datos, de manera de poder disponer de ellos acercando los

datos al destino para beneficiar a otros Consumidores que verían así reducido el retardo y beneficiando a la red, y de esta manera se podría reducir el tráfico intermedio. Los sistemas de uso comercial denominados Caché Networks, como por ejemplo, Akamai y CenturyLink son implementaciones parciales dedicadas a reducir el tráfico hacia el exterior de los proveedores de servicio de internet (ISPs) mediante el almacenamiento temporal estadístico de los datos solicitados por los usuarios. Uno de los problemas de las Redes Centradas en Contenidos es la gestión de la movilidad, debido a que uno de los axiomas fundamentales es el desconocimiento (a priori) de la topología de la red, siendo la única información disponible de enrutamiento la lista de nodos vecinos. Existe un conjunto de soluciones propuestas [2]–[5], pero en general, exigen de la red mayor información geográfica, de enrutamiento o de capas inferiores. Los autores en [6], [7] y [8] avanzaron en el planteamiento y análisis del problema, estableciendo pautas para su solución; siendo el presente el cuarto trabajo en esta línea, con el objeto de obtener una implementación de un prototipo funcional.

II. ALGORITMO PREDICTIVO

El caso de uso más común en la telefonía celular consiste en un conjunto de nodos de infraestructura de acceso fijos (radiobases) a los que acceden los nodos móviles (teléfonos celulares) para realizar los requerimientos de flujos de datos (flujos desde el productor al consumidor, por ejemplo, video). En este caso, el nodo móvil se desplaza con cierta velocidad entre una radiobase y la siguiente.

El algoritmo propuesto consiste en un agente que reside en los nodos de infraestructura de acceso fijos (radiobases) donde detecta el primer interés para un contenido determinado y predice el desplazamiento del nodo móvil, para luego hacer que los próximos nodos de infraestructura en la línea de desplazamiento generen un interés similar para el mismo contenido al productor. El productor, al recibir los intereses de los nodos intermedios, envía el dato en el camino inverso solicitado. De esta forma los próximos nodos de infraestructura de acceso dispondrán del contenido en su caché disponible cuando el nodo móvil alcance los futuros nodos en su recorrido. Para lograr esto, el NA que recibe primero el interés lo reenvía

ignacio.navarrete@mail.udp.cl
diego.dujovne@mail.udp.cl
griva@frc.utm.edu.ar

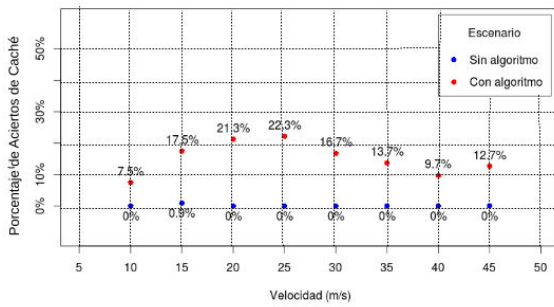


Figura 1. Tasa de aciertos de caché vs Velocidad aplicando el algoritmo

a al NA siguiente para que este lo solicite al productor, encadenando las solicitudes.

III. ESCENARIO Y RESULTADOS DE ACIERTOS DE CACHÉ

Este trabajo se basa en una simulación realizada con ns3, utilizando el módulo NDN. El escenario propuesto representa el caso de uso de un conjunto de vehículos en una autopista o un conjunto de usuarios en un tren. Este escenario consta de un nodo móvil consumidor, que se desplaza en línea recta y se enlaza con uno de los Nodos de Acceso (NA) (al costado del camino o de la vía) a medida que avanza. Los NA a su vez reciben y envían los intereses del consumidor hacia el servidor donde se encuentra el contenido solicitado por el nodo móvil productor. Un acierto de caché ocurre cuando una aplicación solicita un dato. Primero los NA buscan los datos en su ubicación de memoria más cercana, que en este caso sería su propio Caché. Si los datos solicitados están almacenados allí, se considera un acierto de caché. Esta métrica permite conocer cuántos intereses fueron recibidos por parte de los nodos móviles y así observar qué tan exitoso sería en un caso real.

En la Figura 1 se presenta el porcentaje de aciertos de caché para los escenarios sin algoritmo y con algoritmo, en función de la velocidad del móvil. Se observa que en el escenario sin algoritmo, la tasa de aciertos es siempre cero en una arquitectura de red realista, comparada con los casos estudiados en la bibliografía, donde se utilizan mallas uniformes. En términos cuantitativos, se observa una tasa de acierto de caché de más de un 20% para el caso de 20 metros por segundo.

En la Figura 2, se muestran los resultados para el escenario con algoritmo donde se observa la mejora hasta para cuatro nodos móviles, después que comenzó el desplazamiento el primer nodo móvil. Se destaca que el cuarto móvil tiene 5 veces más probabilidad de acierto de caché que el primero a bajas velocidades. En una transmisión en streaming de un evento masivo (por ejemplo, un concierto o un evento deportivo) esta característica genera un efecto colaborativo, reduciendo el tráfico de red de manera significativa, al no necesitar reenviar los bloques de datos solicitados por el nodo móvil. Esto lleva concluir que en la topología de NA lineales,

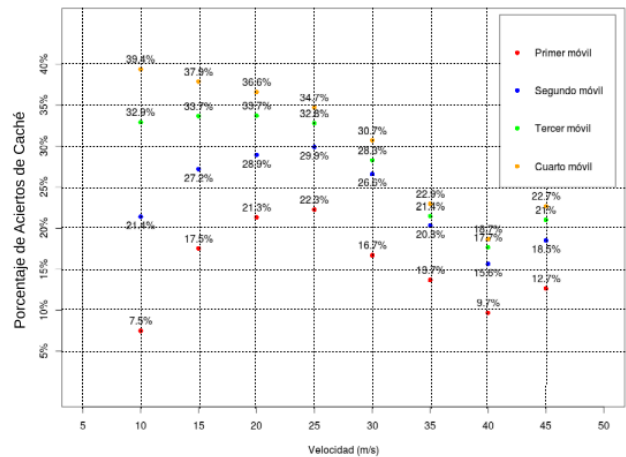


Figura 2. Tasa de aciertos de caché vs Velocidad con algoritmo para cuatro móviles.

se puede mejorar el acceso al contenido de los nodos móviles. Además, se aprecia que llegando a la velocidad de 30 metros por segundo o 108 km/h, los aciertos en la topología bajan, esto debido a que ya los intereses solicitados no están siendo satisfechos gradualmente, debido a la velocidad con que se desplaza un consumidor. La leve alza posterior a esta velocidad se atribuye a la influencia de los bloques siguientes, que fueron solicitados con una estimación de velocidad mayor.

IV. CONCLUSIÓN

En este trabajo se propone y valida con simulaciones un algoritmo para mejorar el comportamiento de las Redes Orientadas a Contenidos en condiciones de movilidad. A futuro se plantea realizar mejoras al algoritmo para el caso de usuarios en medios colectivos de transporte.

REFERENCIAS

- [1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM, 2009, pp. 1–12.
- [2] D.-h. Kim, J.-h. Kim, Y.-s. Kim, H.-s. Yoon, and I. Yeom, "Mobility support in content centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 13–18.
- [3] Y. Luo, J. Eymann, and A. Timm-Giel, "Mobility support for content centric networking," *Telecommunication Systems*, pp. 1–18, 2014.
- [4] X. Vasilakos, V. A. Siris, G. C. Polyzos, and M. Pomonis, "Proactive selective neighbor caching for enhancing mobility support in information-centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 61–66.
- [5] G. Tyson, N. Sastry, I. Rimac, R. Cuevas, and A. Mauthe, "A survey of mobility in information-centric networks: challenges and research directions," in *Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications*. ACM, 2012, pp. 1–6.
- [6] F. Silva, "Predicción de solicitudes móviles en redes orientadas a contenidos," Diciembre 2012, memoria de Título para Ingeniería Civil en Informática y Telecomunicaciones.
- [7] F. Silva and D. Dujovne, "Predictive mobility applied to content centric networks," in *Communications (LATINCOM), 2013 IEEE Latin-America Conference on*. IEEE, 2013, pp. 1–6.

- [8] C. S. Matus, "Estudio y aplicación de algoritmos para movilidad en redes orientadas al contenido," Septiembre 2014, memoria de Título para Ingeniería Civil en Informática y Telecomunicaciones.

Evaluación de Desempeño de un Patrón de Software para Multiprocesadores Asimétricos en Sistemas Embebidos

Pedro Ignacio Martos

GPSIC & LSE – Facultad de Ingeniería
Universidad de Buenos Aires
Ciudad Autónoma de Buenos Aires, Argentina
pmartos@fi.uba.ar / pimartos@gmail.com

Alejandra Garrido

LIFIA – Facultad de Informática
Universidad Nacional de La Plata & CONICET
La Plata, Prov.de Buenos Aires, Argentina
garrido@lifia.info.unlp.edu.ar

Resumen—Dentro de los sistemas embebidos hay una variante de sistemas multiprocesador (Multicore System on Chip devices – MSoC devices) donde no todos los procesadores son iguales. Estos SoCs se denominan “Dispositivos Multiprocesadores Asimétricos” (Asymmetric Multi Processing Devices – AMP Devices). Para tomar ventaja de las posibilidades que estos dispositivos ofrecen en sistemas embebidos, se han definido patrones de diseño específicos. No obstante ello, en la literatura hay pocos resultados experimentales que muestren los beneficios de aplicar estos patrones. En este trabajo se realiza la medición de desempeño del patrón “Ejecución Optimizada” (Optimized Execution), aplicado a un dispositivo AMP y usando la suite de desempeño “CoreMark”. Los resultados muestran que el uso de este patrón de diseño mejora el desempeño del sistema.

Palabras Clave— *Patrones de Sistemas Embebidos; Patrones para Multiprocesamiento Asimétrico; Software Embebido.*

I. INTRODUCCIÓN

Los sistemas embebidos (Embedded Systems – E.S), en oposición a los sistemas de propósito general, son sistemas desarrollados con un propósito específico. En algunos casos el usuario final del sistema puede configurarlo (e incluso programarlo), pero el sistema no cambia su propósito. Estos sistemas se denominan “embebidos” porque son parte de un sistema más grande en un dispositivo con una funcionalidad específica [1].

Un Multiprocesador Asimétrico (AMP), es un procesador en el que los elementos de cómputo (“núcleos”) tienen diferentes características. Estas diferencias van desde el mismo tipo de procesador ejecutándose a diferentes velocidades de reloj, hasta arquitecturas completamente distintas. Estos procesadores son estudiados porque se encontró que proveen una mejora en la potencia de cómputo [2], y también en las relaciones (desempeño)/(consumo) y (desempeño)/(área de silicio) [3], por lo que son atractivos para la implementación de E.S. No obstante ello, el diseño de un E.S. que pueda tomar ventaja de la mejora en desempeño de los AMP es difícil. Por ejemplo, la asimetría en desempeño puede afectar adversamente el comportamiento de distintas cargas de cómputo en servidores comerciales y hacer que el sistema sea menos escalable [2].

Con el objetivo de ayudar a los desarrolladores a tomar buenas decisiones de arquitectura de software al implementar sistemas AMP y para tomar ventaja de sus características, hemos trabajado en la especificación de un lenguaje de patrones para sistemas embebidos AMP [4]. Un lenguaje de patrones es una colección interconectada de patrones de diseño o buenas prácticas en un dominio específico [5]. Cada patrón de diseño del lenguaje identifica una solución recurrente a un problema en un contexto específico [6]. De esta manera, un patrón contiene una pequeña porción de conocimiento de diseño y arquitectura para resolver un problema en un cierto contexto y una vez resuelto, deja al sistema en un nuevo contexto, donde hay nuevos problemas a resolver por otros patrones del lenguaje [7].

En otros trabajos hemos realizado mediciones de desempeño de algunos patrones del lenguaje propuesto para sistemas AMP [4]. En particular hemos demostrado que una configuración AMP tiene un desempeño mejor a lo esperado al aplicar el patrón “Mini-Me” también cuando la asimetría de la configuración AMP se debe a procesadores con diferente hardware y con cargas específicas de E.S.[8]

En este trabajo se realiza la medición de desempeño del patrón “Optimized Execution” para evaluar la mejora que produce al aplicarlo. Este patrón, el cual se describe en detalle en la Sección 2, propone que, cuando se tiene un sistema AMP cuyos núcleos tienen distinto ISA (Instruction Set Architecture – ISA), y un ISA es un subconjunto del otro, se debe tomar ventaja de las instrucciones específicas del ISA más grande, a fin de mejorar el tiempo de ejecución de las tareas. En particular se intenta responder a la siguiente pregunta: “¿Que tan importante es la diferencia de desempeño cuando se usa el ISA específico de cada núcleo en una configuración AMP donde el ISA de un tipo de núcleo es un subconjunto del ISA del otro tipo de núcleo?”

Esta pregunta es importante porque el uso del mismo ISA en núcleos como los descriptos anteriormente tiene una ventaja importante: el sistema puede implementarse con todo el software usando el mismo ISA y la asimetría solo estaría en el hardware, con los resultados descriptos en [8]. Pero, si la diferencia en desempeño es significativa, se necesita un nuevo patrón de software para contemplar esta situación. Para contestar esta pregunta se utilizó una plataforma de cómputo AMP con dos núcleos: un núcleo Cortex M4 y un núcleo Cortex M0, en el que el ISA del núcleo Cortex M0 es un

subconjunto del ISA del núcleo Cortex M4. Asimismo se uso una suite de desempeño bien conocida para este estudio: CoreMark [9].

El resto de este trabajo se estructura de la siguiente forma: la Sección 2 presenta los trabajos relacionados, la Sección 3 describe la suite de desempeño y detalles de la implementación, la Sección 4 muestra los resultados experimentales, y la sección 5 presenta las conclusiones y trabajos a futuro.

II. TRABAJOS RELACIONADOS

Los procesadores AMP se estudian como dispositivos de cómputo por las ventajas que ofrecen. En particular su desempeño es mejor a lo esperado. Como se explicó en la introducción, una limitación del trabajo previo [8] para nuestro propósito es que en el sistema utilizado todos los núcleos tienen el mismo ISA. Nosotros estamos interesados en evaluar el desempeño cuando el ISA de los procesadores es diferente.

Otros trabajos sobre lenguajes de patrones relacionados con el dominio de los E.S. son los patrones para software tolerante a fallas de Hanmer [7] y los patrones para sistemas embebidos de White [10]. Uno de los patrones en el lenguaje de patrones para AMP en E.S es “Optimized Execution” [4], el cual se usa en el contexto en el que un sistema embebido utiliza un procesador AMP donde los núcleos tienen diferentes ISA, y el ISA de un tipo de núcleo, denominado “A”, es un subconjunto del ISA del otro tipo de núcleo, denominado “B”. Este patrón se aplica al problema de cuando se tienen tareas con requerimientos de tiempo real estricto, cuando hay diferentes ISA disponibles, y un ISA es un subconjunto del otro, es necesario tomar ventaja de las instrucciones específicas del ISA más grande, para mejorar el tiempo de ejecución de las tareas. Por otra parte este Patrón puede considerarse como opuesto al Antipatrón “Level Down” [4], el cual plantea que el uso del mismo ISA en procesadores asimétricos produce una merma en el desempeño total del sistema. Nuestra intención en este trabajo es evaluar el desempeño de la arquitectura propuesta por el patrón Optimized Execution en el contexto de los E.S.

III. MEDICIÓN DE DESEMPEÑO

La plataforma de hardware utilizada fue la Edu-CIAA, la cual utiliza el procesador de dos núcleos LPC 4337 de NXP, el cual tiene un núcleo complejo (ARM Cortex M4@204MHz) y un núcleo simple (ARM Cortex M0@204MHz) [11]. El ISA del núcleo Cortex M0 (Armv6-M) es un subconjunto del ISA del núcleo Cortex M4 (Armv7-M), lo que hace a esta plataforma apropiada para medir una implementación del patrón Optimized Execution. La suite de desempeño CoreMark fue obtenida del sitio web de EEMBC; compilada con el nivel de optimización -O3; y configurada para ejecutarse desde memoria RAM en configuración “Release”.

CoreMark es una suite que mide el desempeño de microcontroladores (MCUs) y unidades centrales de procesamiento (CPUs) utilizadas en sistemas embebidos. CoreMark implementa los siguientes algoritmos: manejo de

listas (búsqueda y ordenamiento), manipulación de matrices (operaciones comunes con matrices), máquinas de estado (determinar si una secuencia de entrada contiene números válidos), y chequeo de redundancia cíclica (CRC – cyclic redundancy check). Está diseñada para ejecutarse en dispositivos que van desde microcontroladores de 8-bit hasta microprocesadores de 64-bit. Para asegurar que el compilador no pueda pre-computar los resultados en tiempo de compilación, cada operación de la suite de desempeño genera un valor que solo puede determinarse en tiempo de ejecución. Asimismo todo el código usado en la parte temporizada es parte de la suite en sí misma (no se realizan llamadas a librerías).

IV. RESULTADOS EXPERIMENTALES

La Tabla 1 muestra los resultados CoreMark absolutos de la plataforma de hardware. Asimismo se muestran los resultados CoreMark para el procesador Cortex M4 LPC 54102[12] para validar nuestra medición respecto a la medición CoreMark realizada por NXP. La Tabla 2 muestra los resultados CoreMark/MHz, los cuales se utilizan para comparar mediciones CoreMark realizadas con procesadores ejecutándose a diferentes velocidades.

TABLA 1: RESULTADOS COREMARK ABSOLUTOS

Configuración	CoreMark
LPC 4337 Armv7-M	488
LPC 4337 Armv6-M	390
LPC 54102 Armv7-M	499

TABLA 2: RESULTADOS COREMARK/MHz

Configuración	CoreMark/MHz
LPC 4337 Armv7-M	2.39
LPC 4337 Armv6-M	1.91
LPC 54102 Armv7-M	2.45

De la Tabla 1 se ve que hay un 25% de diferencia en desempeño usando el ISA Armv6-M, respecto de usar el ISA Armv7-M. Este resultado muestra que el ISA utilizado tiene gran impacto en el desempeño del sistema.

De la Tabla 2 se observa que la medición de NXP es similar a nuestra medición. La diferencia puede explicarse debido a que el procesador LPC 54102 es más nuevo y seguramente mejorado respecto al procesador LPC 4337 usado en nuestras mediciones, por lo que consideramos que nuestras mediciones son válidas y nuestra implementación de la suite CoreMark es correcta.

V. CONCLUSIONES Y TRABAJO A FUTURO

En este trabajo se realizó la medición de desempeño del patrón de software Optimized Execution, utilizando una suite de desempeño orientada a tareas específicas de E.S. Nuestra pregunta probó ser muy importante y a través de mediciones experimentales se encontró que se puede esperar una diferencia de desempeño de alrededor de 1.25 veces.

Los trabajos a futuro incluyen realizar mediciones de desempeño de otros patrones del lenguaje, extenderlo y descubrir nuevos patrones específicos del dominio.

REFERENCIAS

- [1] Steve Heath, *Embedded Systems Design*, 2nd Ed. Elsevier, (2002)
- [2] Balakrishnan, Rajwar, Upton, Lai, The impact of performance asymmetry in emerging multicore architectures, *Proceedings of the 32nd International Symposium on Computer Architecture (ISCA '05)*. IEEE. (2005)
- [3] Fedorova, Saez,Shelepov, Prieto, Maximizing power efficiency with asymmetric multicore systems, *Communications of the ACM*, Vol 52 Issue 12 (2009)
- [4] P. Martos, *Architectural Patterns for Asymmetric Multiprocessing Devices on Embedded Systems*, *Proceedings of the 11th Latin American Conference on Pattern Languages of Programs (SugarLoaf PLoP '16)*. Hillside Group (2016)
- [5] R. Hanmer, *Pattern-Oriented Software Architecture For Dummies*, John Wiley & Sons, 2013
- [6] Gamma, Helm, Johnson, Vlissides. *Design Patterns*. Addison-Wesley, 1995.
- [7] Robert S. Hanmer, *Patterns for fault tolerant software*, Wiley Series in Software Design Patterns. John Wiley & Sons (2007)
- [8] P.Martos & A.Garrido, *Software Patterns for Asymmetric Multiprocessing devices on Embedded Systems: a Performance Assessment*", *Eight Argentine Symposium and Conference on Embedded Systems (CASE)*. IEEE (2017)
- [9] Shay Gal-On and Markus Levy, *Exploring CoreMark™ – A Benchmark Maximizing Simplicity and Efficacy*. EEMBC (2009)
- [10] E. White, "Making Embedded Systems: Design Patterns for Great Software", O'Reilly Media (2011)
- [11] "LPC 4337",
<https://www.nxp.com/products/processors-and-microcontrollers/arm-based-processors-and-mcus/lpc-cortex-m-mcus/lpc4300-cortex-m4-m0/32-bit-arm-cortex-m4-m0-mcu-up-to-1-mb-flash-and-136-kb-sram-ethernet-two-high-speed-usb-lcd-emc:LPC4337JBD144>,
(obtenido en Mayo,2019)
- [12] NXP Semiconductors, "AN 11607 LPC 5410x CoreMark Cortex M4 Porting Guide" (2015)

FORO TECNOLÓGICO

Módulo vocalizador para balanza digital

basado en EDU_CIAA_NXP

Diana C. Vertiz del Valle
Universidad Nacional de Entre Ríos
Laboratorio de Laboratorio de Prototipado Electrónico y 3D
Oro Verde, Entre Ríos - Argentina
jesuisdianacarol@gmail.com

Eduardo Filomena
Universidad Nacional de Entre Ríos
Laboratorio de Laboratorio de Prototipado Electrónico y 3D
Oro Verde, Entre Ríos - Argentina
efilomena@ingenieria.uner.edu.ar

Resumen—Este artículo presenta el diseño e implementación de un módulo de reproducción de voz para conectar a una balanza digital comercial y permitir la operación de la misma a personas con dificultad en la visión. El módulo lee los comandos digitales que envía la balanza al estabilizar la medida, y reproduce vocalmente el peso. Este proyecto, ha sido realizado a raíz del pedido de una escuela para personas con discapacidad visual, para facilitar a sus alumnos el trabajo en los talleres de panadería.

Palabras clave—vocalizador, modulo parlante, conversión texto a voz.

I. INTRODUCCIÓN

La Convención Internacional sobre los Derechos de las Personas con Discapacidad, reconoce el derecho al trabajo como derecho humano fundamental y exige a los Estados a salvaguardar y promover el ejercicio de ese derecho, adoptando las medidas pertinentes [1]. La inclusión de las personas con discapacidad al ámbito laboral, representa un reto tanto desde lo humano, lo social, como incluso desde el punto de vista tecnológico. La adaptación de instrumentos o dispositivos de uso comercial y cotidiano para su utilización por parte de personas con capacidades visuales diferentes es un desafío a la ingeniería que siempre es reconocido y valorado por los usuarios de esas tecnologías.

En este caso, como solicitud del personal de la Escuela Integral N° 1 para personas con discapacidad visual "Helen Keller" de la ciudad de Paraná, surge la idea de adaptar una balanza digital de uso comercial para su utilización por parte de los alumnos que realizan labores en los talleres de panadería de esta institución.

II. DISEÑO

La idea original, fue la de disponer de una balanza comercial que posea un puerto de comunicaciones mediante el cual se transmita la información del peso registrado y diseñar un módulo que reproduzca vocalmente esta información.

El modelo de balanza digital se seleccionó por varios aspectos distintivos: calidad, precio y disponibilidad del protocolo del puerto de comunicaciones. Con todo esto, se llegó a la selección de la balanza digital marca Kretz modelo ECO NOVEL II [2] de la cual el fabricante suministró toda la información necesaria para tener acceso al puerto de comunicaciones, en este caso, un puerto serie RS232.

Como alternativas para generar la vocalización, surgieron tres posibilidades que se estudiaron en

profundidad: el uso de circuitos integrados sintetizadores de voz, los algoritmos de software sintetizadores de voz, y la disposición de un banco de voces y fonemas previamente grabado y almacenado, que se reproducen en forma secuencial y ordenada según el texto que se quiere reproducir.

La alternativa de uso de circuitos integrados como el WTS701 [3] o de módulos como el TTS-06 [4] fue descartada por el elevado costo de cualquiera de estas alternativas. Además, al momento de tomar la decisión, todas las opciones disponibles eran solamente para el idioma inglés, lo que dificultaba seriamente la interpretación del habla puesto que la entonación no resulta para nada natural.

La opción de un algoritmo que sintetice por software las señales de voz fue descartada debido a que, la complejidad de estos algoritmos era tal, que se necesitaban procesadores y placas de gran capacidad de procesamiento y costo para poder implementarlos [5] [6] [7] [8].

La tercera opción fue la alternativa seleccionada como la más indicada para este proyecto: disponer de un banco de voces y fonemas almacenados en una memoria y un algoritmo que, según el código del dato recibido por el puerto de comunicaciones, lee y reproduce estas voces para vocalizar el peso correspondiente. La reproducción de las señales de voz se realiza mediante un conversor digital analógico (DA), un amplificador de audio y un parlante.

III. DESARROLLO

El diagrama del sistema puede observarse en la Fig. 1.

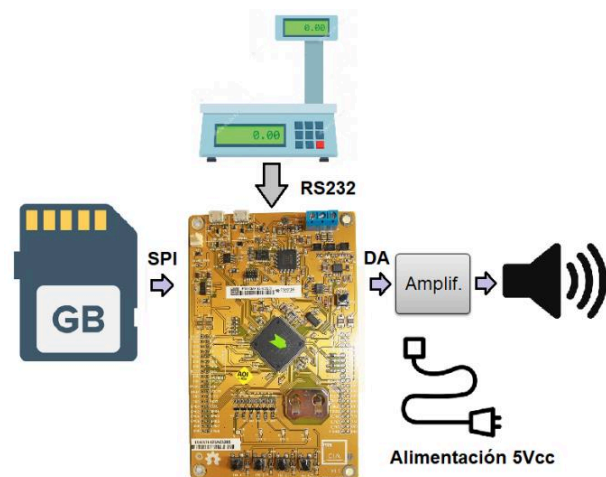


Fig. 1. Diagrama de la solución propuesta

Como sistema de almacenamiento del banco de voces se utilizó una memoria SD debido a su bajo costo, facilidad de lectura y escritura, y fundamentalmente por la posibilidad de montar sobre ella un sistema de archivos de manera tal de poder leer y escribir los archivos fácilmente desde una computadora personal. La posibilidad de desmontar esta memoria y montarla en una computadora, permite la mejora, ampliación y mantenimiento del banco de voces con mucha facilidad.

Con el fin de acortar los plazos del desarrollo del prototipo, se optó por el uso de una placa de desarrollo ya completamente montada y funcional que contenga todos los periféricos necesarios: un puerto serie RS232 para conectar la balanza, un puerto SPI para la conexión de la memoria SD y una salida analógica mediante conversor DA. Es por esto que se empleó la placa EDU_CIAA_NXP [10] que es una versión educativa del proyecto CIAA [9] y de la cual se disponía mucho conocimiento de su uso y programación. Por otro lado, esta placa, cuenta con un procesador –el LPC4337– cuya velocidad de operación y cantidad de memoria RAM (136KB), es suficiente para poder realizar la lectura de la memoria y el armado de las frases en memoria a una velocidad tal que no se perciban cortes ni saltos en la señal generada.

Esta señal de salida generada por el conversor DA es magnificada por un amplificador de audio y reproducida mediante un parlante.

A. Banco de voces

Para el almacenamiento del banco de voces se utilizó una memoria SD de tipo SDHC de 1Gb, con un sistema de archivos FAT32 en el que se graban todas las voces necesarias para poder replicar vocalmente la información del peso registrado por la balanza digital. El banco fue inicialmente descargado del sitio YouTube [11], remuestreado a 8 kHz, transformado a monoaural y almacenado en un formato sin compresión para facilitar la lectura y reproducción. Este banco está formado por los vocablos ‘uno’, ‘un’, ‘dieci’, ‘veinte’, ‘veinti’, ‘kilogramo’, ‘gramo’, etc. cada uno en un archivo diferente. Algunas de las voces faltantes fueron directamente grabadas mediante el software “Audacity” y almacenadas en el mismo formato. Este banco ocupó un total de 340 kbytes.

B. Firmware

La balanza digital, al presentar una lectura de peso estable, envía por el puerto de comunicaciones RS232, un mensaje en código ASCII conteniendo el peso registrado. El formato de transmisión se muestra en la Fig. 2, donde (BOT) es un indicador de inicio de la transmisión.

```

2 , XX.XXX , CR
|       |       |
(BOT) (PESO NETO) (CARACTER ASCII 13)

```

Fig. 2. Formato de transmisión.

El firmware desarrollado en la EDU_CIAA_NXP recibe este mensaje, lo decodifica y arma en memoria toda la frase para ser verbalizada, y luego genera mediante el conversor DA la señal de audio correspondiente. Seis vectores de 8kbytes cada uno localizados en memoria RAM almacenan temporalmente la información del peso separado en

kilogramos, centenas de gramos, decenas de gramos y gramos junto a las unidades de peso “kilogramo” y “gramo”.

C. Amplificador de audio

Para las pruebas de funcionamiento, se utilizó un amplificador comercial con entrada de audio y control de volumen.

IV. RESULTADOS

Una fotografía del prototipo funcional se puede ver en la Fig. 3.



Fig. 3. Fotografía del prototipo funcional

Al momento de realizar las pruebas no se disponía de la balanza, por lo que se implementó una aplicación que, según el peso seleccionado, generaba el mensaje correspondiente según el protocolo y lo enviaba por puerto serie. Puede apreciarse el funcionamiento del sistema en estas condiciones en [11].

V. DISCUSIÓN

Si bien las voces generadas por este sistema resultan claras, fáciles de comprender y con una entonación natural del idioma español, mejoraría mucho la vocalización, si se pudiera conseguir un banco de voces completamente registrado por una sola voz y de un profesional como puede ser un locutor. El uso de la EDU_CIAA_NXP, facilitó y aceleró el desarrollo pero el sistema resulta voluminoso, costoso y poco compacto. Un avance, sería diseñar una sola placa que contenga el zócalo de la memoria SD, el procesador y el amplificador de audio. El procesador empleado resultó sobredimensionado, por lo que se podría cambiar por un modelo más pequeño, económico y de menor consumo.

VI. CONCLUSIONES

Los objetivos de este trabajo fueron ampliamente cumplidos, el prototipo responde a los comandos digitales generando vocalizaciones claras, sin cortes y con una entonación aceptable. El método de disponer de un banco de voces y fonemas almacenados y leerlos en función del peso recibido de la balanza para así vocalizar el mensaje, demostró ser eficiente, consumir pocos recursos tanto del procesador como de la memoria y la calidad del audio generado resultó más que satisfactoria.

Como trabajo a futuro, se propone mejorar el banco de voces almacenado en la tarjeta SD y reemplazar la EDU_CIAA_NXP por una placa diseñada ad hoc, con un micro más ajustado a los requerimientos reales del sistema.

REFERENCIAS

- [1] Convención sobre los derechos de las personas con discapacidad Naciones Unidas Enable. Consultado:06-03-2009.
- [2] Kretz (2019, Mayo, 20), Balanza Novel ECO 2 [Online] http://www.kretz.com.ar/contenidos/2016/11/24/Editorial_2815.php
- [3] Winbond (2019, Mayo, 15), Single Chp Text to Speech Processor [Online] <https://datasheet.octopart.com/WTS701EM/T-Winbond-datasheet-122861.pdf>
- [4] TextSpeak (2019, Abril, 12), ADA ATM TTS-06 [Online] <https://www.textspeak.com/product/adaatm-tts-07/>
- [5] Lemmetty, S., 1999. Review of Speech Synthesis Technology. Masters Dissertation, Helsinki University Of Technology
- [6] Dutoit, T., 1993. High quality text-to-speech synthesis of the French language. Doctoral dissertation, Faculte Polytechnique de Mons.
- [7] Suendermann, D., Höge, H., and Black, A., 2010. Challenges in Speech Synthesis. Chen, F., Jokinen, K., (eds.), Speech Technology, Springer Science + Business Media LLC.
- [8] Allen, J., Hunnicutt, M. S., Klatt D., 1987. From Text to Speech: The MITalk system. Cambridge University Press.
- [9] Proyecto CIAA (2019, Mayo, 15), EDU-CIAA-NXP [Online] <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>
- [10] NXP (2019, Mayo, 23) LPC4337 [Online] <https://urlzs.com/enV5U>
- [11] YOUTUBE Video de módulo para balanza para ciegos <https://youtu.be/IPz-e7X9T7E>

Procesamiento de señal visualizado sobre un espectrograma.

Paz, Martín E.¹, Friedrich, Guillermo.², Galasso, Christian L.³

^{1,2}Cátedra de Técnicas Digitales III, ³Cátedra de Teoría de Circuitos II,
Departamento de Ingeniería Electrónica
Facultad Regional Bahía Blanca
Universidad Tecnológica Nacional.
Bahía Blanca - Argentina

¹pazmartin35@gmail.com, ²gfried@frbb.utn.edu.ar, ³christian_galasso81@yahoo.com.ar

Resumen—En el presente trabajo se describe la implementación de un espectrograma digital sobre un sistema embebido, para la visualización de los efectos de aplicar filtros digitales y FFT sobre una señal. Se implementa el uso de un sistema operativo FreeRTOS como administrador de recursos del microcontrolador, y para el correcto funcionamiento del equipo.

Palabras claves — sistema embebido; espectrograma; filtro digital; FFT; sistema operativo.

I. INTRODUCCION

El espectrograma es una herramienta de representación que se utiliza para el análisis de una señal eléctrica, de comunicación o audiovisual. Consiste en tomar una determinada cantidad de muestras por medio de una ventana temporal, para luego realizar el cálculo del contenido espectral de las muestras por medio de una FFT (*Fast Fourier Transformation – Transformada Rápida de Fourier*) [1], representando los resultados en una gráfica de tres dimensiones (frecuencia, tiempo y amplitud). Seguidamente, la ventana temporal se desplaza a lo largo del tiempo de la señal y se toman una nueva cantidad de muestras, repitiendo el ciclo de cálculo y graficando los resultados. La suma de las representaciones de la FFT obtenidas, aportan información en el dominio de la frecuencia sobre la variación de la energía y la frecuencia de la señal en función del tiempo.

La FFT es un algoritmo eficiente que permite reducir el costo computacional al realizar un análisis en el dominio frecuencial de una señal mediante DFT (*Discrete Fourier Transform – Transformada Discreta de Fourier*).

En este proyecto también se utilizan filtros digitales de tipo IIR (*Infinite Impulse Response – Respuesta Infinita al Impulso*) y FIR (*Finite Impulse Response – Respuesta Finita al Impulso*) [2] pudiendo elegir entre: filtro pasa bajo o pasa alto de tipo IIR o FIR, a fin de comparar su funcionamiento, graficando los resultados a través del espectrograma.

II. IMPLEMENTACIÓN

El microcontrolador utilizado es el STM32F407VG, el cual trabaja a 168 MHz. Los periféricos del microcontrolador utilizados son: conversores ADC de 12 bits y velocidad de conversión de hasta 1 Mhz., FSMC (*Flexible Static Memory*

Controller – Control de Memoria Estática Flexible) para el control de una pantalla TFT de 7 pulgadas, I2C para la configuración y comunicación con el controlador de la pantalla táctil capacitiva, y SPI para el almacenamiento de las capturas de pantalla en una memoria SD en formato bmp.

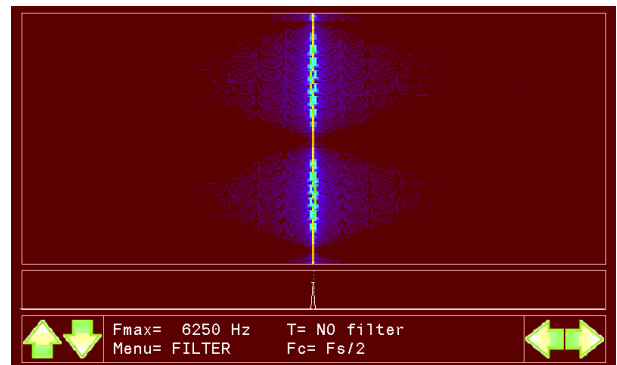


Fig. 1 – Captura de pantalla del espectrograma.

La figura 1 muestra una captura de pantalla de la GUI (*Graphical User Interface – Interfaz Gráfica de Usuario*), la cual se puede separar en tres ventanas: ventana inferior – donde se encuentra el menú de usuario junto con los botones de selección, ventana central – donde se grafica la FFT de la señal de entrada (analyzer de espectro), y ventana superior – donde se registra el histograma de la FFT (función principal del Espectrómetro).

La librería utilizada para el algoritmo de FFT [3] pertenece a la firma STMicroelectronics. Este algoritmo se encarga de convertir la señal muestreada en tiempo a frecuencia permitiendo visualizar su componente espectral. Una variable importante al configurar este algoritmo es la cantidad de datos que se van a utilizar, ya que de esto depende la resolución del espectro.

```
arm_rfft_init_f32(&S,      &S_CFFT,      FFT_LENGTH,
ifftFlag, doBitReverse);

arm_rfft_f32(&S, FFT_DATA_IN, FFT_CMPLX_DATA);

arm_cmplx_mag_f32(FFT_CMPLX_DATA,      FFT_MAG_DATA,
FFT_LENGTH);
```

Debido a la naturaleza del algoritmo para realizar la transformada, los datos en el arreglo FFT_MAG_DATA

(resultados de la conversión) se encuentran reflejados, siendo útiles $FFT_LENGTH/2$. El algoritmo se configuró para trabajar con FFT_LENGTH igual a 512 datos, y frecuencia de muestreo configurable.

R es la resolución de la FFT, F_S es la frecuencia de muestreo y F_N es la frecuencia de Nyquist, que aparece como F_{MAX} en la figura 1 indicando la máxima frecuencia que se puede visualizar en el espectrograma, y puede ser modificada por el usuario dentro de una gama de valores establecidos. Como los datos útiles son $FFT_LENGTH/2$, la resolución se puede calcular de la siguiente manera:

$$R = \frac{F_S}{FFT_LENGTH} = \frac{F_S}{512} \quad (1)$$

En la ventana superior, el inicio del historial es en la parte inferior de la ventana, avanzando hacia la parte superior repitiendo el proceso y sobre escribiendo los datos anteriores. El desplazamiento del eje temporal, para graficar una nueva trama de datos, se realiza con un período de 30 ms otorgando un historial de 10 segundos del comportamiento de la señal de entrada al instrumento.

El filtro digital IIR se diseñó a partir de una configuración de tipo *Butterworth*, resultando en un filtro de orden 5, mientras que el FIR se realizó con un diseño de enventanado *Rectangular* obteniéndose un filtro de orden 23. Como el filtro FIR presentó una notable oscilación antes y después de la zona de corte, se procedió a cambiar la ventana por una de *Hamming* resultando en un filtro de orden 79.

FreeRTOS [4] es el sistema operativo encargado de administrar los recursos del microcontrolador. Está configurado para trabajar en tiempo real y para crear 3 tareas con distintos niveles de prioridad (0, 1 y 2), dándole mayor prioridad al menú de usuario para un mejor funcionamiento del instrumento.

III. ENSAYOS REALIZADOS

Se realizaron ensayos con diferentes tipos de señales, tales como: senoidal de tono puro con frecuencia fija y frecuencia variable, y amplitud modulada con modulación al 30% y 90%. El instrumento utilizado para generar señal es un Keysight 33500B, configurando la señal senoidal con 1V de pico para todas las señales utilizadas.

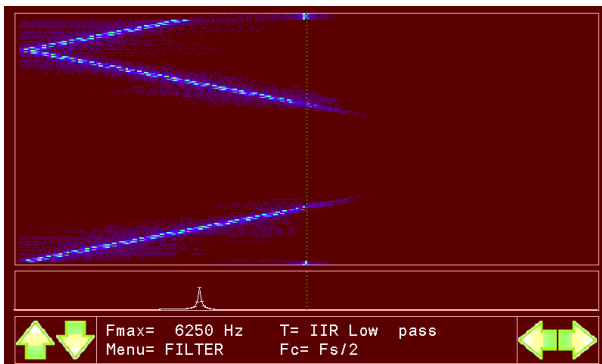


Fig. 2 – Captura de pantalla del espectrograma

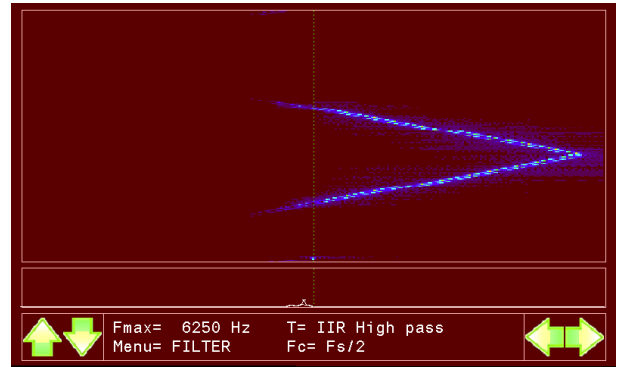


Fig. 3 – Captura de pantalla del espectrograma

En la figura 2 y 3 se observa el espectrograma de una señal senoidal que atraviesa dos filtros digitales tipo IIR: pasa bajo y pasa alto, aplicados a la mitad de la frecuencia de muestreo. La señal de entrada es senoidal con incremento y decremento lineal de su frecuencia desde 100 Hz hasta 6 kHz.

IV. CONCLUSIONES

Cuando la señal de entrada posee una frecuencia igual a un múltiplo de la resolución de la FFT, se puede apreciar un único pico en el analizador de espectro, y una línea fina y continua en el espectrograma. Cuando no coinciden las frecuencias, se produce un derramamiento o leakage de la señal, resultando en picos extendidos a las señales vecinas con amplitud reducida a la real. Este fenómeno se puede apreciar en la figura 1, donde la frecuencia de la señal oscila entre 3120 Hz y 3130 Hz. En el espectrograma se aprecian tramos del historial donde se visualiza una línea fina y continua, periodo donde la señal está a una frecuencia de 3125 Hz, y tramos del historial donde la línea es gruesa, de menor amplitud, y que se extiende a los datos vecinos del arreglo, que en este caso son los extremos de la oscilación.

Una opción para mejorar el fenómeno de derramamiento es aumentar el valor de FFT_LENGTH , permitiendo una mejor discriminación de la señal en el analizador de espectro, a costa de aumentar el tiempo de cálculo para el procesamiento de la señal, y en un aumento de la memoria requerida.

Modificar la frecuencia de muestreo, manteniendo FFT_LENGTH constante, y visualizando una parte del arreglo en la pantalla.

Otra opción es la aplicación de un bloque de enventanado, es decir multiplicar los valores de la señal de entrada por una función ventana para disminuir en derramamiento espectral, utilizando ventanas Hamming, Hanning o Blackman, para luego realizar la FFT.

V. REFERENCIAS

- [1]. Samir S. Soliman, “Señales y Sistemas continuos y discretos”, segunda edición, Madrid 1999, ISBN: 84-8322-154-3.
- [2]. Steven W. Smith, “The Scientist and Engineer's Guide to Digital Signal Processing”, Segunda edicion, California 1999, ISBN 0-9660176-6-8.
- [3]. DSP library. Disponible en:
http://www.st.com/content/ccc/resource/technical/document/user_manual/b1/01/3b/ab/75/ef/4d/91/CD00144074.pdf/files/CD00144074.pdf/jcr:content/translations/en.CD00144074.pdf
- [4]. FreeRTOS open source licensing. 22 de diciembre de 2017. – Disponible en
<https://www.freertos.org/a00114.html>

Implementación de un Vúmetro de 8 Bandas con la EDU-CIAA

1.° Maximiliano Alexandroff
Universidad Nacional de La Plata
Facultad de Ingeniería
La Plata, Buenos Aires
malexandroff@linti.unlp.edu.ar

2.° Tomás Rosales
Universidad Nacional de La Plata
Facultad de Ingeniería
La Plata, Buenos Aires
tomasjoaquinrosales@gmail.com

3.° José Juárez
Universidad Nacional de La Plata
Facultad de Ingeniería
La Plata, Buenos Aires
UNQ Dpto. CyT
Bernal, Buenos Aires
jjuarez@ing.unlp.edu.ar

Resumen—En el presente trabajo se describe la implementación de un vúmetro de 8 bandas utilizando la plataforma EDU-CIAA. El objetivo principal es el procesamiento digital de una señal de audio en tiempo real, su análisis espectral y la visualización del resultado en una matriz de LEDs de 8x8. Se presenta el diseño de hardware para acondicionar la señal analógica y para controlar la matriz, se describe el desarrollo de software y presentan los resultados obtenidos.

Palabras claves—vúmetro, audio, EDU-CIAA, DSP, diseño de PCB, FFT.

I. INTRODUCCIÓN

En los últimos años con el avance de la tecnología se ha producido una migración cada vez mayor desde el procesamiento analógico hacia el procesamiento digital. En consecuencia, para realizar el procesamiento digital es necesario realizar varias tareas, desde el acondicionamiento de señales hacia el microcontrolador, la conversión analógico-digital, el procesamiento de la misma y su posterior visualización. El presente artículo muestra la implementación de un vúmetro de 8 bandas sobre la placa de desarrollo EDU-CIAA [1] en el cual se realiza el procesamiento digital de audio en tiempo real, proyecto propuesto por la materia *Taller de Proyecto I* de la carrera de Ingeniería en Computación de la Facultad de Ingeniería de la UNLP.

II. ETAPA DE ANÁLISIS

El proceso de desarrollo del sistema actual, comenzó con una exploración de los componentes necesarios. Luego, el diseño y fabricación de la placa de expansión o “Poncho” que cumple las funcionalidades principales del proyecto. Las cuales son: recibir y acondicionar la señal de audio para enviarla a la placa EDU-CIAA, brindar una interfaz al usuario mediante el uso de teclado y posteriormente al análisis de la señal de audio, presentar por medio de la matriz de leds 8x8 la potencia promedio de dicha señal en cada banda. Por último, se realizó el desarrollo de software el cual incluye: muestreo de la señal entrante, análisis espectral, comunicación SPI, ingreso de teclado y planificación de tareas compartidas.

Las 8 bandas que se eligieron para realizar el presente proyecto se muestran en el *Cuadro I*.

Cuadro I
SELECCIÓN DE BANDAS

Banda	Rango [Hz]	Banda	Rango [Hz]
1	0 a 160	2	160 a 300
3	300 a 600	4	600 a 1200
5	1200 a 2400	6	2400 a 5000
7	5000 a 10000	8	10000 a 16000

Para realizar el análisis en el dominio de frecuencia se optó por utilizar la transformada rápida de Fourier (*FFT, Fast Fourier Transform*) [2][3] que es un algoritmo optimizado para facilitar el cálculo computacional.

En la Figura 1, se ilustra un esquema sintetizado del proceso que se utilizó para llevar a cabo la implementación del sistema de procesamiento de audio en tiempo real.

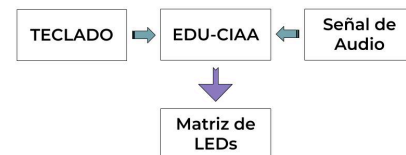


Figura 1. Diagrama de bloques del sistema.

III. ACONDICIONAMIENTO DE LA SEÑAL

Antes de muestrear y procesar, fue necesario conocer las características de la señal de entrada al sistema. Como entrada, se dispuso de una señal audio provista por un celular, tablet o PC que está estandarizada para trabajar con tensiones entre $\pm 1V$, lo cual fue confirmado midiendo con un osciloscopio, su señal a máximo volumen. Por otro lado, para el muestreo de la señal se utilizó el conversor analógico-digital (ADC) de 10 bits de la EDU-CIAA, el cual posee un rango de tensiones entre 0V y +3,3V. Como primer paso, se aplicó un offset a la señal entrante, para que la misma trabaje en tensiones positivas. Esto se logró mediante la utilización de un divisor de tensión desacoplado en continua, seguido por un filtro anti-aliasing cuya frecuencia de corte es de 16kHz.

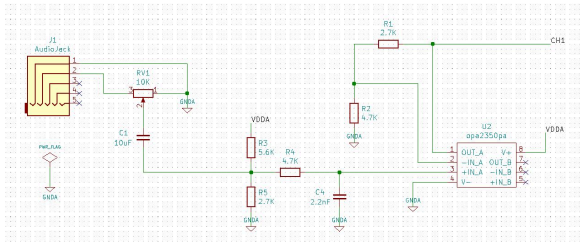


Figura 2. Esquemático - Entrada de Audio.

Luego, para utilizar el rango completo de entrada del ADC de la EDU-CIAA, se utilizó un amplificador operacional rail to rail (OPA2350) utilizando un circuito no-inversor para obtener una ganancia de $G = 1,5745$. Con lo cual, como resultado se obtuvo una señal que variaba aproximadamente entre un valor mínimo de $+0,12V$ y un valor máximo de $+3,27V$, centrada en $+1,695V$. El circuito de acondicionamiento de la señal se muestra en la Figura 2.

IV. DESARROLLO DE PONCHO

Para este sistema, se diseñó un circuito impreso con los siguientes componentes: audio jack de entrada, resistencias, capacitores, amplificador operacional OPA2350, controlador de matriz de led MAX7219, matriz de leds de 8×8 y, por último, los terminales adecuados para los pines de la placa EDU-CIAA. Luego de la etapa de diseño, se pasó a la etapa de fabricación del PCB, en la cual, se realizó la impresión del diseño sobre una placa de cobre para luego realizar la soldadura de los componentes sobre la placa. Para esta etapa se utilizó el paquete de software libre de KiCad [4] y el modelo de poncho [7] mediano (planilla PCB con las interfaces de conexión de la EDU-CIAA) como base para el diseño del PCB.

V. DESARROLLO DE FIRMWARE

Previamente a comenzar a escribir el código, se realizó un estudio de la arquitectura de la placa de desarrollo a utilizar y de los módulos que se comunican con la misma.

V-A. Filtrado Digital

Para realizar el procesamiento digital de audio en tiempo real en la EDU-CIAA se optó, como se mencionó anteriormente, por utilizar la FFT en vez de utilizar 8 filtros digitales mediante la convolución. El motivo de esta decisión, fue la optimización de tiempo de ejecución del firmware y poder realizar el procesamiento en tiempo real, ya que, la FFT es más eficiente que la utilización de filtros digitales. Para realizar el muestreo, se escogió una frecuencia de muestreo de 32.768 Hz para cumplir con el teorema de Nyquist-Shannon y un tamaño de muestra de 2048 elementos debido al espacio reducido de memoria. Con estos parámetros, se calculó la resolución de la FFT mediante la siguiente ecuación:

$$Resolucion = \frac{Fs}{N} \quad (1)$$

Esto significa que, se cuenta con una resolución de 16 Hz . De esta manera, el filtrado digital se realizó tomando porciones del buffer de 2048 elementos. A modo de ejemplo, si se toman los primeros 10 elementos del mismo, se obtiene el rango de frecuencias de 0 Hz a 160 Hz . Mediante este análisis, se pudo realizar la división del espectro de frecuencia en 8 bandas y realizar el cálculo de potencia para cada una en particular y expresarlo mediante una escala de leds.

V-B. Cálculo de Potencia

Una vez realizado el filtrado de la señal, se calculó la potencia de cada banda mediante el uso de la librería **CMSIS-DSP** [5]. Para esto, se procesó la FFT calculada por la función: `void arm_cmplx_rfft_fast_f32()`. Ya que, retornaba un arreglo de números complejos. Por lo tanto, se calculó la magnitud de ese arreglo mediante la función: `void arm_cmplx_mag_f32()`. Este arreglo de magnitudes solo presentaba la mitad de la FFT, pero, como se estaba procesando una señal real, la magnitud de la transformada era simétrica. De esta manera, se utilizó la función: `void arm_power_f32()`. La cual, realizaba el cálculo de la energía de la señal. Finalmente, aplicando la relación de Parseval, y multiplicando por dos, debido a la simetría, se obtuvo la potencia.

VI. RESULTADO

A lo largo del proyecto, se desarrollaron diferentes ensayos llevados a cabo en el laboratorio ATEI de la Facultad de Ingeniería. Se comenzó con un ensayo de prueba sobre el Firmware v2 [6] que provee el proyecto CIAA, el cual, hace utilización del conversor analógico digital y se comprobó la correcta configuración del ADC y periféricos. Luego, se le realizaron pruebas al poncho para verificar que la señal estaba bien acondicionada antes de conectarse al ADC de la placa. Esto se realizó utilizando el osciloscopio y observando en diferentes niveles de la señal de entrada, esto es: señal de entrada, señal luego de pasar por el divisor de tensión, señal después de ser amplificada y por último la señal resultante al pasar por el circuito antialiasing. Posteriormente, se realizaron ensayos de volumen y determinación de la escala de visualización junto con ensayos de tonos senoidales puros barriendo diferentes frecuencias. Por último, se comprobó el funcionamiento utilizando la señal proveniente del reproductor de música de un smartphone.

En la Figura 3, se puede observar el sistema en funcionamiento.

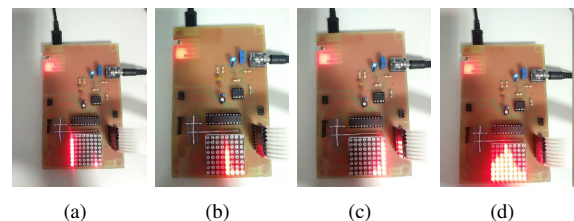


Figura 3. Tono de (a) 80Hz (b) 1800 Hz (c) 15000Hz (d) Música

REFERENCIAS

- [1] (2018, Nov.) CIAA website. [Online]. Disponible en internet: <http://www.proyecto-ciaa.com.ar/>
- [2] A.V. Oppenheim, A. S. Willsky, S.H. Nawab, Señales y Sistemas, 2ª edición, Pearson Educación, 1998.
- [3] M.J. Roberts, Señales y Sistemas, McGraw - Hill Interamericana, 2005.
- [4] (2018, Nov.) KiCad website. [Online]. Disponible en internet: <http://kicad-pcb.org/>
- [5] (2018, Nov.) Keil website. [Online]. Disponible en internet: <https://www.keil.com/pack/doc/CMSIS/DSP/html/index.html>
- [6] (2018, Nov.) Firmware v2 website. [Online]. Disponible en internet: https://github.com/ciaa/firmware_v2
- [7] (2018, Nov.) CIAA GitHub. [Online]. Disponible en internet: <https://github.com/ciaa/Ponchos/tree/master/modelos>

Diseño de un sistema crítico de enclavamiento ferroviario implementado mediante FPGA

M. Menéndez, A. Lutenberg, N. Alvarez
CONICET – GICSAFe
Laboratorio de Sistemas Embebidos
Facultad de Ingeniería, UBA
Buenos Aires, Argentina
lse@fi.uba.ar

F. Larosa, R. Ghignone
CONICET – GICSAFe
Grupo de Aplicaciones en Sistemas Embebidos
Departamento de Electrónica, UTN-FRH
Haedo, Buenos Aires, Argentina
embebidos@frh.utn.edu.ar

Resumen— Un enclavamiento es un sistema ferroviario que controla en forma automática que los cambios de vías se produzcan en forma segura, evitando choques de trenes y descarrilamientos. En este artículo se presenta el análisis del estado del arte realizado sobre diversos equipos de enclavamiento comerciales. A partir de este análisis se propone el diseño de la arquitectura de un sistema de enclavamiento y su implementación en tecnología FPGA, aplicando técnicas apropiadas para lograr el nivel de seguridad requerido.

Keywords— *FPGA; Interlocking; Railway; Critical systems*

I. MOTIVACIÓN Y CONTEXTO

Este trabajo nace de un pedido de Trenes Argentinos[1], que requirió el diseño de un sistema de enclavamiento electrónico. El mismo tiene por función garantizar el accionamiento seguro de los dispositivos de señalamiento y de maniobra de los trenes asociados a las maniobras de cambio de vía. En Argentina los enclavamientos tienen entre 40 y 100 años de antigüedad, por lo que muchos han agotado su vida útil y deben ser reemplazados.

Con el objetivo de realizar el diseño de un sistema de enclavamiento se realizó una extensa búsqueda bibliográfica. Sin embargo, no se encontraron artículos que detallen el estado del arte en cuanto a las tecnologías que se utilizan en la industria, excepto por [2] que abordaba un análisis superficial de los sistemas de enclavamientos de la India y dada la fecha de publicación, la tecnología empleada resulta obsoleta hoy en día. Es por eso que se realizó en primer lugar un análisis del estado del arte para determinar los puntos de partida con vistas a realizar una implementación propia. A partir del relevamiento realizado sobre una decena de empresas [3-13] se encontró que las mismas se encuentran en su mayoría en Europa (73%), Norteamérica y parte de Asia. No hay empresas entre las relevadas que diseñen, fabriquen o comercialicen sistemas de enclavamiento en el resto de América, África u Oceanía.

Las empresas analizadas recurren en la totalidad de los casos a el uso de electrónica programable dedicada, ya sea con

computadoras o microcontroladores específicos, haciendo uso de la redundancia de software y hardware[14].

La redundancia en sistemas críticos se define por medio de la abreviatura $NóM$, donde M representa a la cantidad de módulos de medición o decisión que posee el sistema y N la cantidad de dichos módulos que deben funcionar correctamente para que el sistema opere normalmente[15]. De la investigación realizada surge que 66% de las empresas considera que una redundancia $2ó62$ o $2ó63$ es suficiente para alcanzar los niveles de seguridad requeridos[2-4,7-13]. Sólo una pequeña porción de las mismas utilizan redundancias menos estrictas ($1ó62$)[5] o más exigentes ($2ó64$)[6]. En consecuencia se puede afirmar que es razonable utilizar redundancia $2ó62$ o $2ó63$ en un diseño propio.

Adicionalmente, se llegó a la conclusión de que una arquitectura de tres capas (Fig. 1) es representativa de la mayoría de los casos estudiados. La comunicación entre capas se implementa por medio de protocolos tales como Profibus o Ethernet industrial, aunque también existen protocolos cerrados definidos por cada empresa[7]. Se considera entonces que es conveniente utilizar Ethernet industrial dada la gran variedad de implementaciones que existen, lo cual redundará en una mayor flexibilidad del diseño.

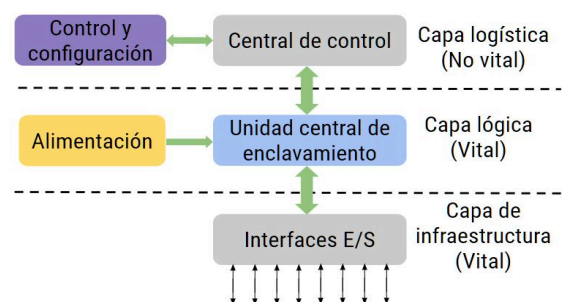


Figura 1. Capas del sistema de enclavamiento

II. ARQUITECTURA PROPUESTA

De acuerdo a los resultados obtenidos en el relevamiento realizado se definieron las funciones de cada módulo (Tabla 1) con el objeto de servir de base para un diseño propio.

TABLA I. ARQUITECTURA DE CAPAS DEL SISTEMA DE ENCLAVAMIENTO

Capa	Tipo	Incluye	Descripción
Lógica	Vital	Alimentación y Unidad central de enclavamiento	Procesamiento de las señales y enclavamiento de estados.
Infraestructura	Vital	Módulo de adquisición	Lectura de las señales con redundancia para evitar valores erróneos.
Logística	No vital	Interfaz gráfica y terminal de comandos	Presenta estado del sistema, ingreso de peticiones de ruta y cambios de modo.

La capa lógica se implementará sobre una FPGA con redundancias 2ó3[15] y para la capa logística se diseñará una interfaz en Python. Por otro lado, en esta instancia del trabajo, la capa de infraestructura se simulará por medio de señales sintéticas. A futuro se migrará esta última a un módulo de lectura/escritura en una estación ferroviaria real, luego de superar los ensayos.

La arquitectura se describe en VHDL (Fig. 2) y se implementa bajo un modelo de desarrollo guiado por pruebas (del inglés, *test-driven development*, TDD)[18]. Es decir, cada módulo creado debe ser antecedido por una serie de ensayos orientado a su validación utilizando la herramienta COCOTB[19]. De esta forma se construye un sistema bajo el método de programación *bottom-up* para luego integrar los módulos funcionales progresivamente.

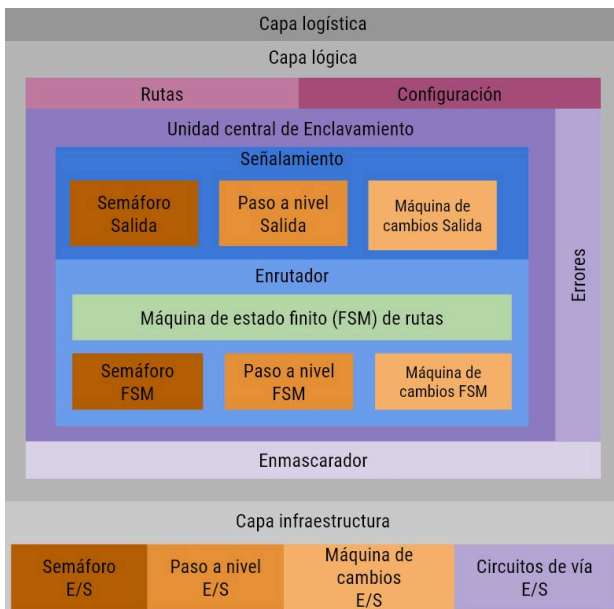


Figura 2. Modelo de arquitectura en capas

III. UNIDAD CENTRAL DE ENCLAVAMIENTO

Dentro de la capa lógica se destaca la unidad central de enclavamiento (Fig. 3) por ser el núcleo del procesamiento y la sección del sistema que se implementa íntegramente en FPGA. La misma se divide en dos niveles de trabajo con diferentes funciones: la zona de control y la zona de comando.

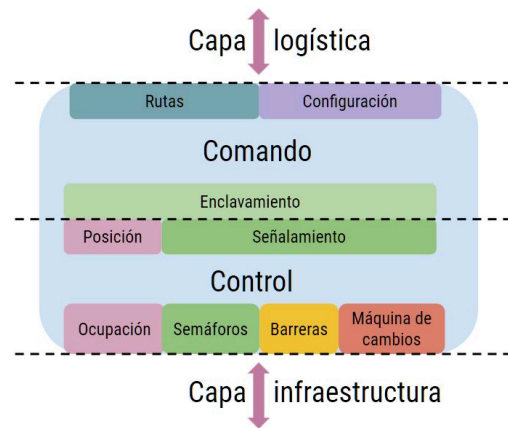


Figura 3. Unidad central de enclavamiento

Zona de comando: procesa las señales que vienen de la capa de logística y corrobora que el modo pedido pueda ser ejecutado o las rutas requeridas puedan ser aplicadas. En caso contrario, las señales se mantendrán bloqueadas o según la jerga ferroviaria, enclavadas, hasta que la situación que lo impide se extinga o cancele. Además se envía el estado actual del sistema a la interfaz por medio de la comunicación a la capa logística.

Zona de control: procesa las señales que ingresan a la capa de infraestructura según el modo de funcionamiento, rutas requeridas y el estado de las señales. A su vez, se envían las señales procesadas a la capa de infraestructura para su escritura.

IV. CONCLUSIONES

La investigación realizada brindó información relevante para realizar el diseño del sistema de enclavamiento presentado en la sección III. Todas las capas se encuentran ya definidas y se están implementando bajo metodología TDD. Además se está trabajando en el uso de la herramienta COCOTB para automatizar las pruebas de VHDL.

REFERENCIAS

- [1] Trenes Argentinos (2019). [Online]. Disponible:
<https://www.argentina.gob.ar/transporte/trenes-argentinos>
- [2] Analysis of Railway Interlocking Systems (2019). [Online]. Disponible:
https://www.academia.edu/15034386/Analysis_of_Railway_Interlocking_Systems
- [3] Trackguard Westrace Mk II. (2014) Flexible safety processor.[Online]. Disponible:
<https://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/en/rail-solutions/rail-automation/electronic-interlockings/trackguard-westrace-mk2-en.pdf>
- [4] Trackguard WESTLOCK. (2013). Cost-effective, highly available and reliable. [Online]. Disponible:
<https://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/en/rail-solutions/rail-automation/electronic-interlockings/trackguard-westlock-en.pdf>
- [5] Alstoms Smartlock equipment at the heart of modular signalling project in the UK (2011).[Online]. Disponible:
<https://www.alstom.com/press-releases-news/2011/3/Alstom-Smartlock>
- [6] Computer based interlocking (2019).[Online]. Disponible:
<http://sts.hitachirail.com/en/products-services/business-segments/computer-based-interlocking>
- [7] Systémy pro kolejovou dopravu (2019). Elektronické Stavedlo TYP ESA® 33[Online]. Disponible:
<https://www.azd.cz/admin-data/storage/get/198->
- [8] Route Control Systems (2019). LockTrac 6111 ESTW L90 [Online]. Disponible: <https://www.thalesgroup.com/en/route-control-systems>
- [9] EBI LOCK (2014). Computer-Based Interlocking System [Online]. Disponible:
http://www.infrasing.net/media/88880/pub-00074_ebi_lock_bro_prof_print_english.pdf
- [10] Bombardier Transportation (2015). Rail Control Solutions (RCS) [Online]. Disponible:
http://www.irse.org/knowledge/publicdocuments/Bombardier%2001%20IRSE_BT_RCS_Introduction_%20November%202015%20final.pdf
- [11] Railway Signaling Solutions (2019). Interlocking Equipment [Online]. Disponible: <https://www.kyosan.co.jp/english/product/signal03.html>
- [12] Kyosan history (2019). [Online]. Disponible:
<https://www.kyosan.co.jp/english/company/history02.html>
- [13] Hima, Rail references (2019). COTS Rail Applications [Online]. Disponible:
<https://www.hima.com/en/industries-solutions/cots-rail-references>
- [14] Shimamura, K., Yamaguchi, S., Kanekawa, N., Miyazaki, N., Yamada, H., Takahashi, Y., Hirotsu, T., Tomobe, K., Satoh, K., Hotta, T. and Fujita, R. (2002). A fail-safe microprocessor using dual synthesizable processor cores. *AP-ASIC'99. First IEEE Asia Pacific Conference on ASICs (Cat. No.99EX360)*.
- [15] R. Dobias and H. Kubatova, "FPGA based design of the railways interlocking equipments," *Euromicro Symposium on Digital System Design, 2004. DSD 2004.*, 2004.
- [16] Formal Specification of Interlockings using Statecharts, Italy, Università degli Studi di Firenze.
- [17] Software Design based on Architecture Conformance, Japon, NEC Corporation.
- [18] Test-driven development concepts, taxonomy, and future direction (2005). D. Janzen, H. Saiedian. [IEEE]
- [19] COroutine based COsimulation TestBench. (2014). COCOTB. [Online]. Disponible: <https://cocotb.readthedocs.io/en/latest/introduction.html>

Adquisición de mediciones inerciales y de GNSS para sistemas de navegación integrada

Elián R. Hanisch[†], Germán Scillone, Santiago Rodríguez, Javier G. García[‡]

UIDET Sistemas Electrónicos de Navegación y Telecomunicaciones (SENYT), Facultad de Ingeniería de la UNLP

[†]elian.hanisch@ing.unlp.edu.ar [‡]jgarcia@ing.unlp.edu.ar

Resumen—La integración de un sistema de navegación inercial (INS) con un sistema satelital de navegación global (GNSS) es una técnica muy potente para obtener una solución de navegación superior a la que se obtendría de cada sistema si se utilizaran aisladamente. En este trabajo se describe el diseño de un sistema de adquisición de mediciones inerciales y de GNSS para el posterior desarrollo de un sistema de navegación integrado INS/GNSS.

Index Terms—INS, GNSS, IMU, navegación

I. INTRODUCCIÓN

Los sistemas satelitales de navegación global (GNSS), tales como GPS, GLONASS o Galileo, son ampliamente utilizados gracias a que proveen una solución de posición confiable y con algunos metros de error; pero tienen desventajas tales como una baja tasa de datos, pueden ser obstruidos por edificación, sufrir radio interferencias y, para el caso de receptores de una sola antena, no brindan información sobre orientación. Los sistemas de navegación inerciales (INS) en contrapartida tienen una alta tasa de datos, son inmunes a las interferencias, poco error a corto plazo y pueden determinar orientación; tiene como desventajas requerir inicialización con una posición conocida y acumular errores a largo plazo. Como las desventajas y ventajas de los GNSS e INS son complementarios se puede obtener una solución de navegación superior al combinarse ambos sistemas en un sistema integrado.

II. DESCRIPCIÓN DEL SISTEMA

El sistema de adquisición consiste en una unidad de medición inercial (IMU) ADIS16305 [1] para adquirir las mediciones inerciales, un receptor GNSS de doble antena desarrollado en el grupo de trabajo [2] y una placa de desarrollo ZYBO de Diligent [3] para almacenar y sincronizar las mediciones.



Fig. 1. Diagrama del sistema.

La placa de desarrollo ZYBO tiene como características relevantes un SoC Zynq Z-7010 (que incluye 2 núcleos Cortex-A9 a 650 MHz y un FPGA Artix-7), 512 MB de memoria DDR3 y soporte para memoria MicroSD.

El receptor GNSS que está desarrollado es compatible con el formato CubeSat [4], dispone de 2 antenas y es capaz de operar en las bandas L1 y L2 del sistema GPS. El receptor transmite sus datos a la placa de desarrollo a través de una UART a 115,2 kbps y tiene una señal de Pulso Por Segundo (PPS), que está sincronizada con el reloj del sistema GPS e indica el momento de las mediciones, que se usa para sincronizar en el tiempo la información de navegación.

La IMU utilizada posee una terna de acelerómetros y un giroscopio orientado verticalmente y es capaz de generar mediciones a una tasa de 819,2 Hz, que se transmiten a través de un SPI con un *clock* de 1,5 MHz. Una señal *Data Ready* (DR) que se genera cada vez que termina una conversión se utiliza para sincronizar en el tiempo las mediciones.

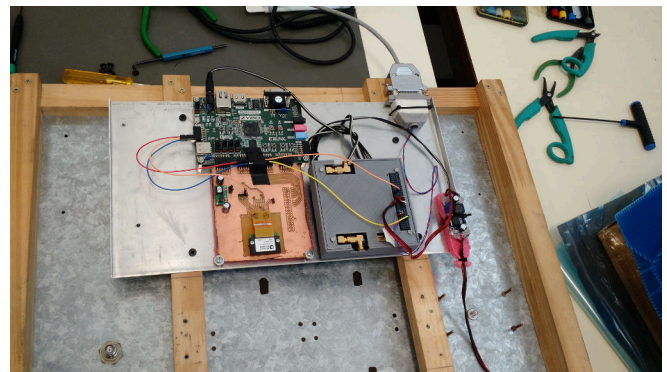


Fig. 2. Fotografía del sistema armado. Debajo de la placa ZYBO se encuentra la placa con la IMU y a la derecha el receptor GNSS. En la parte inferior en la placa galvanizada se encuentran los 2 conectores donde se ubican las antenas, aún sin conectar al receptor.

El FPGA del SoC Zynq solo implementa la comunicación SPI con la IMU, siendo el resto del sistema programado en C y ejecutado en uno de los núcleos Cortex-A9. Este atiende 3 interrupciones: uno para manejar los mensajes que el receptor envía por la UART, y los otros 2 son interrupciones GPIO para detectar los flancos de subida de las señales DR y PPS.

Cuando se inicia la adquisición la placa ZYBO espera la llegada de un flanco de PPS que el receptor solo envía después de adquirir suficientes satélites para tener una solución de posición. Una vez ocurrido esto por cada pulso de DR la placa ZYBO se comunica con la IMU para guardar 8 campos de 16 bits donde los más importantes son los 3 acelerómetros,

el giróscopo, la temperatura y la tensión de alimentación. El receptor GNSS envía por la UART distintos mensajes de información, pero solamente se almacenan los mensajes de navegación y observables. Los mensajes de navegación incluyen la solución de posición y velocidad entre otros datos; los observables incluyen información de cada señal de satélite que se recibe en cada antena. Por cada pulso de PPS se reciben y guardan 2 mensajes de navegación (uno por cada banda L1 y L2) y uno de observables.

La captura de las mediciones inerciales se realiza sin perder muestras. En el caso de los mensajes de GNSS, el sistema podría descartar mensajes silenciosamente en el caso de que estos llenen la cola de la UART, pero esto sería una ocurrencia extraordinaria.

Para sincronizar los datos se utiliza un *timer* global de 64 bits (con un *clock* de 325 MHz); por cada flanco de DR y PPS se toma una captura del valor del *timer* global y se guarda junto a los datos de la IMU y del receptor.

Los datos generados se almacenan inicialmente en la memoria RAM de la placa ZYBO y una vez finalizada la adquisición en una tarjeta SD; la memoria RAM disponible permite adquirir datos durante al menos 2 horas antes de tener que guardarlos a la tarjeta SD.

III. RESULTADOS

En la Fig. 3 se muestra un ensayo realizado en la playa de estacionamiento de un centro comercial.



Fig. 3. Trayecto realizado, según la solución de posición de L1.

Los resultados de las mediciones de GPS resultaron ser satisfactorios excepto en el breve lapso donde hay un desvío de la trayectoria de entre 60 m y 100 m aproximadamente; que podría deberse fundamentalmente a interferencia de múltiples trayectorias o multicamino.

Otro problema que afectó la calidad de la solución de navegación fue que el receptor tuvo pocos satélites adquiridos y esto se debió a 2 razones: 1) porque el receptor funcionaba con una lista reducida de satélites para reducir procesamiento y 2) porque el receptor se instaló junto a las antenas y expuesto al medio ambiente, lo que afectaba la estabilidad de los osciladores del equipo. Esto era algo que se esperaba y se trató de mitigar utilizando una caja fabricada con una impresora 3D para proteger el receptor.

Las mediciones inerciales obtenidas (Fig. 4) resultan ser consistentes con la trayectoria realizada.

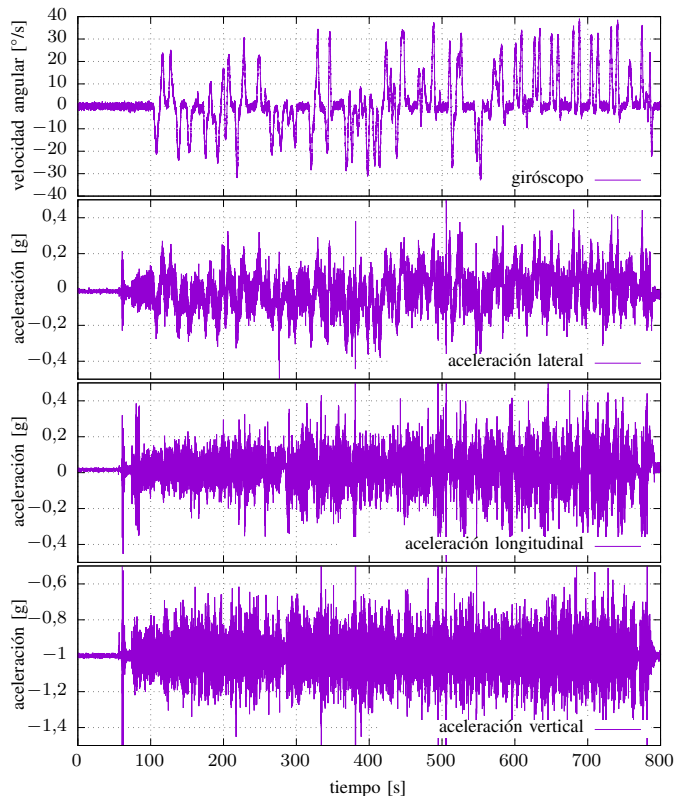


Fig. 4. Mediciones de la IMU.

IV. CONCLUSIÓN Y TRABAJO FUTURO

Se logró el objetivo principal de implementar un adquisidor de mediciones inerciales y de GNSS que estuvieran sincronizadas en el tiempo.

El trabajo futuro consistirá en el cálculo de una solución de navegación integrada implementando un filtro de Kalman [5] junto con un análisis de los errores involucrados.

Como la IMU utilizada no dispone de una terna completa de giróscopos, no es apta para estimar la orientación del vehículo a no ser que sus mediciones se combinen con otros sensores. En el caso de vehículos terrestres este tipo de IMU se puede combinar con un sensor de velocidad [6], en lugar de eso se quiere explorar la posibilidad de aprovechar que el receptor GNSS dispone de 2 antenas (lo que permite determinar 2 ejes de la orientación del vehículo, guiñada y cabeceo) y determinar orientación en el sistema integrado.

REFERENCIAS

- [1] ADIS16305 Precision Four Degrees of Freedom Sensor <https://www.analog.com/media/en/technical-documentation/data-sheets/ADIS16305.pdf>
- [2] S. Rodríguez, R. L. L. Valle, G. L. Puga, J. G. García, y P. A. Roncagliolo, "Design of a dual-antenna and dual-band GPS receiver for CubeSats", en 2017 IEEE URUCON, 2017, pp. 1-4.
- [3] ZYBO FPGA Board Reference Manual <https://reference.digilentinc.com/reference/programmable-logic/zybo/reference-manual>

- [4] W.A. Jordi Puig-Suari, Clark Turner, "Development of the standard cubesat deployer and a cubesat class picosatellite", 2001 IEEE Aerospace Conference, 2001.
- [5] P. D. Groves, "Principles of GNSS, inertial, and multisensor integrated navigation systems". Artech house, 2013.
- [6] U. Iqbal, A. F. Okou, y A. Noureldin, "An integrated reduced inertial sensor system — RISS / GPS for land vehicle", en 2008 IEEE/ION Position, Location and Navigation Symposium, 2008, pp. 1014-1021.

Implementación de una Plataforma de Desarrollo CANSAT Multipropósito

Leonardo Anchino[†], Andrés Torti, Emmanuel Dovis, Emanuel Bernardi
Grupo de Investigación en Robótica y Control (GIROC)
Universidad Tecnológica Nacional,
Facultad Regional San Francisco (UTN-FRSFCO)
San Francisco, Córdoba, Argentina
[†]anchinoleonardo@gmail.com

Resumen—El presente artículo describe los aspectos técnicos y procedimentales involucrados en la implementación de un nanosatélite no orbital tipo CanSat. Este dispositivo integra componentes mecánicos, electrónicos y elementos de software, desarrollados específicamente para fomentar la divulgación científico-tecnológica, y el despertar de vocaciones científicas. Por último, es importante remarcar que la presente plataforma se empleará como instrumento de desarrollo en la enseñanza de tecnologías espaciales.

Keywords—Educación, nanosatélite, electrónica, firmware, tecnologías espaciales

I. INTRODUCCIÓN

En la actualidad, el ingreso y posterior desgranamiento observado en las carreras de grado, con orientación tecnológica, se ha convertido en un importante obstáculo para el desarrollo de las actividades científicas y técnicas del país. Si bien no es tarea sencilla diagnosticar las razones que ocasionan dicha problemática, es pertinente identificar como causas a la falta de vocación, al desconocimiento y al desinterés en afrontar carreras altamente demandantes [1, 2].

Por tal motivo, desde la presente propuesta, enmarcada en el PID-UTN “Desarrollo de una plataforma educativa basada en proyectos CANSAT” (CCUTNSF0005414), que además constituye parte del proyecto final de grado de dos de los autores [3], se pretende exponer y argumentar las técnicas empleadas en el desarrollo e implementación de un nanosatélite no orbital tipo CanSat [4], que se utilizará como instrumento de divulgación de las tecnologías espaciales.

En base al entorno de aplicación del dispositivo, se consideró necesario cumplir con los siguientes requerimientos:

1. construir una plataforma lista para usar, fácil de replicar, y de diseño libre y abierto.
2. contemplar la capacidad de expansión a través de módulos de simple conexión y configuración que le permitan al usuario, con su funcionalidad base, obtener un CanSat completamente funcional.
3. cumplir con la normativa impuesta por la Agencia Espacial Europea (ESA) [5] que determina las dimensiones, el peso y las velocidades terminales.

Desde el punto de vista de su utilización, el objetivo principal de la plataforma es la realización de misiones de forma similar a las de un satélite orbital. Para ello, el CanSat

es elevado, a través de un globo cautivo o un cohete, a una altura máxima de 1000 m (o 150 m desde un cuadricóptero), para luego descender a velocidad controlada por medio de su paracaídas integrado. Durante la misión, ascenso y descenso, el satélite envía datos a tierra, que luego se analizan para cumplir con el objetivo de la misión, previamente propuesto.

Por último, es de destacar que la plataforma, capturada en la Fig. 1, forma parte de la propuesta de UNISEC-Argentina [6] para fomentar la enseñanza de tecnologías espaciales, y se plantea utilizarla en el dictado del CANSAT Workshop 2019.



Figura 1: Satélite CanSat

II. DESCRIPCIÓN DE LAS HERRAMIENTAS

La plataforma presentada consta de tres partes fundamentales, a saber:

- CanSat: dispositivo a ‘lanzar’ que incluye un sistema de protección interno de tensiones y corrientes, dos actuadores (liberación y paracaídas), un módulo de comunicaciones, una baliza electrónica y diversos sensores para la medición de variables físicas, como lo son la temperatura, la humedad, la presión, la aceleración, el campo magnético, entre otras.
- Estación terrena: módulo compuesto de un transceptor de radiofrecuencia que realiza el enlace entre la terminal de telemetría (en tierra) y el satélite CanSat (en vuelo).
- Software de telemetría: administra y visualiza todos los parámetros del satélite permitiendo la realización de misiones y la creación de reportes.

Durante el diseño de la plataforma se consideró importante la implementación a través de herramientas de software libre y abierto, de modo que su utilización facilite a los posibles usuarios replicarla y expandir su funcionamiento.

II-A. Desarrollo de hardware

En el desarrollo de los esquemas y los correspondientes paneles de circuito impreso, tanto del dispositivo CanSat como de la estación terrena, se empleó el software de diseño asistido por computadora *KiCad* [7]. Dicha herramienta, de acceso libre y gratuito, cuenta con una gran cantidad de librerías pre-diseñadas, o sencillamente realizables a través del software gratuito *Library Loader* de la empresa *SamacSys* [8], reduciendo su complejidad y el tiempo de desarrollo involucrado.

II-B. Desarrollo de firmware

La unidad de control del satélite CanSat está compuesta por el microcontrolador ESP32, de la empresa *Espressif* [9], por lo que se utilizó el framework de desarrollo ESP-IDF, provisto por la misma. Este se compone de la totalidad de los drivers de periféricos y provee las herramientas necesarias para compilar y grabar el firmware en el microcontrolador. Además, dicha herramienta se basa en una versión modificada de FreeRTOS [10], que nos permite aprovechar ambos núcleos.

II-C. Herramientas de desarrollo del software de telemetría

En el desarrollo de la propuesta se estableció que la aplicación sea fácil de utilizar, intuitiva, multiplataforma, multi-lenguaje y responsiva. Teniendo esto en cuenta, se utilizaron las siguientes herramientas:

1) *Template Vuestic Admin* [11]: plantilla de administración responsiva creada en *Vue.js* por la empresa *Epicmax*, bajo la licencia MIT [12]. Esta, cuenta con soporte para la inclusión de nuevos idiomas, además del español e inglés.

2) *Electron* [13]: framework de código abierto utilizado para la creación de aplicaciones de escritorio con HTML, JS y CSS. Permite crear ejecutables de la aplicación para los sistemas operativos Mac OS, Windows y Linux.

3) *Vue CLI Plugin Electron Builder* [14]: plugin que permite convertir la aplicación realizada en *Vue.js* a *Electron* manteniendo su funcionalidad.

III. IMPLEMENTACIÓN DEL HARDWARE

La plataforma de desarrollo se divide en dos dispositivos.

III-A. Cansat

Como se observa en la Fig. 2, el satélite CanSat se compone de múltiples PCBs apilados, e interconectados entre sí. Se los designa de arriba hacia abajo, y se los describe:

1) *PCB_A*: en este módulo se alojan los pines de conexión de la batería LiPo, de una celda, que luego se distribuyen en el dispositivo. Además, incluye la conexión de los servos de liberación de globo y paracaídas, el GPS para obtener su localización y altitud, el interruptor de encendido y apagado general del sistema, y el circuito transceptor que establece el enlace de radiofrecuencia con la estación terrena.

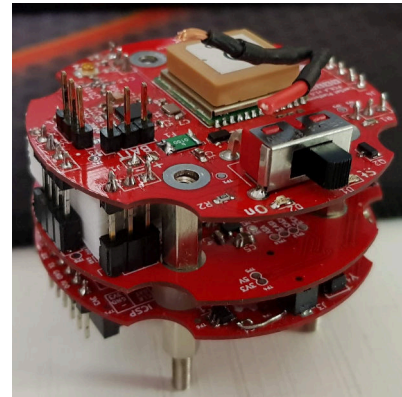


Figura 2: Estructura del CanSat.

2) *PCB_B*: aquí se gestiona la carga y descarga de energía en la batería. Esto es, se construyen las tensiones de alimentación requeridas por el sistema, incluso las tensiones auxiliares que utilizan los circuitos de expansión. Es de destacar que las últimas están protegidas frente a cortocircuitos, sobrecorrientes y sobre-tensiones, de modo que un fallo sobre las mismas no afecte el funcionamiento general, asegurando así la integridad del satélite. Las tensiones generadas son: +5 V para los servos de liberación de globo y paracaídas, +3.3 V para el sistema supervisor, y de manera diferenciada +5 V y +3.3 V para los circuitos de expansión.

3) *PCB_C*: consta principalmente del sistema de control y supervisión del satélite. Lo integra un microcontrolador ESP32, junto a los sensores de presión atmosférica, de temperatura y humedad, e inerciales. Entre sus funciones se destacan la activación del paracaídas, el control de las fuentes de alimentación y mantener la comunicación con la estación terrena. Esto asegura la integridad del satélite, y permite que los módulos de expansión no tengan que cumplir con dichas tareas, simplificando su diseño. Además, en esta se encuentran el conector de carga de la batería y el de expansión. Este último, permite ampliar el funcionamiento base del satélite, de forma segura y sencilla. Para ejemplificar el uso del conector de expansión, se desarrolló un módulo compatible con *Arduino UNO* y otro para prototipos, que contiene un sector pre-perforado para crear circuitos simples rápidamente.

III-B. Estación terrena

Esta se compone de una antena acoplada a un transceptor de radiofrecuencia, con sus elementos de polarización y filtrado, que trabaja a 915 MHz y se conecta al terminal de telemetría para establecer comunicación con el satélite CanSat.

IV. CONCLUSIONES Y TRABAJO FUTURO

Luego de haber ensayado la plataforma a través de múltiples lanzamientos, de hasta 150 m de altura, se ha observado que los resultados alcanzados cumplen con los objetivos inicialmente propuestos. En resumen, se ha construido una plataforma de desarrollo, lista para usar, basada en un satélite CanSat, cuya operación es sencilla, y de diseño libre y abierto.

Las pruebas realizadas expusieron unos pocos aspectos de hardware a mejorar en la siguiente versión, que se encuentra en desarrollo. Por último, se planea ensayar la plataforma a su altura máxima de diseño (1000 m) que no fue alcanzada por falta de medios de elevación.

REFERENCIAS

- [1] UTN-BA. *Esfuerzos en todo el país para promover el ingreso a la UTN*. URL: <https://www.frba.utn.edu.ar/esfuerzos-en-todo-el-pais-para-promover-el-ingreso-a-la-utn/> (visitado 02-05-2019).
- [2] ITBA. *Apostar a las ingenierías, las carreras del futuro*. URL: <https://www.itba.edu.ar/la-universidad/prensa/apostar-a-las-ingenierias-las-carreras-del-futuro/> (visitado 02-05-2019).
- [3] Leonardo Anchino y Andrés Torti. «Desarrollo de satélite CanSat para utilizar como plataforma educativa y fomentar el interés en ingenierías». Proyecto final de carrera. Universidad Tecnológica Nacional, mar. de 2019.
- [4] European Space Agency. *What is a CanSat?* Feb. de 2019. URL: https://www.esa.int/Education/CanSat/What_is_a_CanSat (visitado 02-05-2019).
- [5] European Space Agency. *2019 CanSat Guidelines*. 2019.
- [6] UNISEC Global. *UNISEC Argentina Point of Contact*. URL: <http://www.unisec-global.org/reyna.html> (visitado 02-05-2019).
- [7] *KiCad EDA*. URL: <http://kicad-pcb.org/> (visitado 02-05-2019).
- [8] SamacSys. *Library Loader*. URL: <https://www.samacsys.com/library-loader/> (visitado 02-05-2019).
- [9] Espressif. *ESP32 Overview*. URL: <https://www.espressif.com/en/products/hardware/esp32/overview> (visitado 02-05-2019).
- [10] *FreeRTOS*. URL: <https://www.freertos.org/> (visitado 02-05-2019).
- [11] Epicmax. *Vuestic Admin Dashboard*. 2019. URL: <https://github.com/epicmaxco/vuestic-admin> (visitado 02-05-2019).
- [12] Epicmax. *Vue.js*. URL: <https://vuejs.org/> (visitado 02-05-2019).
- [13] *Electron*. 2019. URL: <https://electronjs.org/> (visitado 02-05-2019).
- [14] Noah Klayman. *Vue CLI Plugin Electron Builder*. 2019. URL: <https://nklayman.github.io/vue-cli-plugin-electron-builder/> (visitado 02-05-2019).

Automatización de un banco de ensayos de generadores eléctricos para aplicaciones en energía eólica de baja potencia

Juan Carlos Agotegaray
Instituto de Industria
Universidad Nacional de
General Sarmiento
Los Polvorines, Argentina
jagotega@campus.ungs.edu.ar

Andrea Pinzón
Instituto de Industria
Universidad Nacional de
General Sarmiento
Los Polvorines, Argentina
apinzon@campus.ungs.edu.ar

Nicolás Emanuel Lera
Instituto de Industria
Universidad Nacional de
General Sarmiento
Los Polvorines, Argentina
nlera@hotmail.com

Abstract—En este trabajo se presenta el desarrollo de un banco de ensayos automatizado de generadores eléctricos con aplicación en eólica de baja potencia y uso en zonas urbanas. La implementación de este banco permite ensayar el generador bajo condiciones de funcionamiento semejantes a los que se sometería como aerogenerador acoplado a una turbina eólica en entornos urbanos, a partir de lo cual, es posible determinar los parámetros específicos que permitan caracterizarlo.

Keywords—Ensayos, Automatización, Aerogenerador, Generator, Adquisición, Control

I. INTRODUCCION

En la Universidad Nacional de General Sarmiento se lleva a cabo el proyecto investigación “Generadores Eólicos Urbanos”, el cual tiene como objetivo el estudio y desarrollo de aerogeneradores de eje vertical con implementación en zonas urbanas. Se construyó un aerogenerador a partir de una turbina Darrieus H acoplada a un generador eléctrico sincrónico de flujo axial con imanes permanentes, el cual se diseñó y construyó teniendo en cuenta los parámetros de funcionamiento de la turbina eólica. Para ensayar el generador eléctrico se lo instaló en un banco [1] y se acopló a un motor trifásico controlado por un variador de frecuencia con el fin de poder modificar la velocidad de rotación (RPM) del conjunto. Las variables relevantes del sistema, que en un principio se realizaban de forma manual, son: RPM, corriente y tensión.

A partir de las mediciones, se planteó la necesidad de mejorar el banco de ensayos utilizando un controlador programable (PLC) que pueda registrar datos y controlar el variador de frecuencia. Con esto, se logró caracterizar el generador eléctrico y dispone de una herramienta para realizar ensayos para el estudio de futuros desarrollos.

II. AEROGENERADOR

A. Turbina Eólica Darrieus H

Para captar la energía cinética contenida en el viento se utiliza una turbina eólica, que para el caso, es de eje vertical, es decir, su eje de rotación es perpendicular a la dirección de viento. Este tipo de aerogeneradores tienen ventajas como la posibilidad de funcionar con vientos que provengan desde cualquier dirección y con cierto nivel de turbulencia. Este comportamiento se da normalmente en zonas urbanas, donde la rugosidad superficial hace que los vientos presenten características aleatorias. La turbina eólica Darrieus H está formada por palas de perfil normalmente biconvexo, unidos rígidamente entre sí, que giran alrededor de un eje vertical

[2]. En el proyecto, la turbina fue construida con tres palas fabricadas a partir de un perfil NACA 0018, con una cuerda de 200 milímetros y 1 metro de largo, definiendo un diámetro total de 1,1 metros, la misma se observa en la figura 1. El ángulo pitch β que se forma entre la dirección de rotación y la cuerda de las palas es de 0° .



Fig. 1. Aerogenerador Darrieus H.

B. Generador Eléctrico de Flujo Axial

Existen distintos tipos de generadores eléctricos que se acoplan a las turbinas eólicas para poder producir energía eléctrica. Los aerogeneradores de baja potencia se implementan normalmente en zonas aisladas de la red eléctrica como una fuente de generación, como los sistemas fotovoltaicos o los grupos electrógenos. Por esta razón, se recurren a generadores de diseño robustos y con mantenimiento mínimo, entre distintos tipos, se destaca el generador sincrónico de flujo axial con imanes permanentes. Este tipo de generador posee rotores con polos magnéticos cuyas líneas de campo atraviesan axialmente un estator de bobinas donde se inducen la corriente eléctrica. Tienen la particularidad de poder tener un gran número de polos magnéticos en poco espacio, lo que es ideal para aplicaciones de baja velocidad. En el marco del proyecto, se construyó un generador eléctrico de este tipo, tomando como base el diseño implementado en el aerogenerador de autoconstrucción Piggot [4]. El generador consta de dos rotores con 12 imanes permanentes cada uno, haciendo un total de 24, y 9 bobinas para conseguir una configuración trifásica compuesta por 3 bobinas por fase.

III. BANCO DE ENSAYOS

El estudio del generador eléctrico se realizó con el fin de determinar su rendimiento, para lo cual se lo consideró, al generador y al sistema de rectificación, como una caja negra sobre la cual se realizan mediciones para conocer la energía mecánica que ingresa y la energía eléctrica que sale. El banco de ensayos debía permitir realizar mediciones eléctricas y mecánicas sobre el generador eléctrico mientras este fuera accionado de forma directa por un motor trifásico Siemens 1LE0102-OEC4 de 1,1 kW. Con esto se buscó simular las condiciones de funcionamiento a distintas velocidades de rotación que le transmite la turbina eólica al generador. Para controlar el funcionamiento del motor trifásico, se empleó un variador de frecuencia Siemens V20 de 1,5 kW, el cual puede recibir las órdenes de funcionamiento a través de un panel de parametrización y operador MOP (*motorized potentiometer*). El motor trifásico y el generador eléctrico se montaron en una estructura metálica especialmente construida para soportar y fijar ambos equipos, y mediante un acople fabricado en una impresora 3D se vincularon permitiendo la trasmisión de potencia mecánica.

En un principio, las mediciones eléctricas de tensión y corriente en continua, y mecánicas de RPM se realizaban con instrumentos manuales. Se configuraba el variador de frecuencia para que el motor gire a una determinada velocidad y se realizaban las mediciones.

A. Integración de PLC y módulo analógico

En el proyecto se decidió construir un tablero de control en el cual se instaló el variador de frecuencia y un PLC Siemens S7 200 CPU 222 con un módulo EM 235 para procesar señales analógicas. Con este equipamiento se mejoró el sistema de medición del banco de ensayos al tener la posibilidad de registrar señales digitales y analógicas provenientes de los sensores instalados sobre el generador eléctrico a analizar, así como también, la posibilidad de controlar el funcionamiento del variador de frecuencia enviándole órdenes mediante señales digitales del PLC y la consigna de velocidad con una señal analógica. El banco de ensayos de generadores eléctricos y el tablero de control se observan en la figura 2.

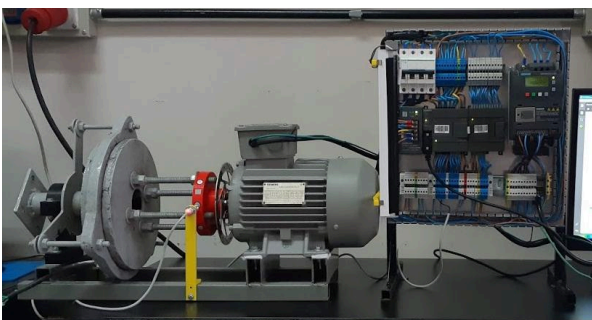


Fig. 2. Banco de ensayos de generadores electricos.

B. Parámetros a medir en el banco de ensayos

La medición que se realizó de parámetros mecánicos fue de velocidad de rotación (RPM). Se espera a futuro poder medir también el torque que trasmite el motor trifásico al generador eléctrico, y de esta manera, calcular la potencia mecánica que ingresa al generador eléctrico. Las mediciones eléctricas que se realizaron son la tensión y corriente a la salida del rectificador en continua, con lo cual se determinó

la potencia eléctrica. De esta manera se espera realizar un balance energético y determinar el rendimiento del conjunto generador-rectificador.

C. Metodología del ensayo

En el ensayo del generador eléctrico se consideró el rango de RPM en el que funciona la turbina eólica Darrieus H, el cual comienza en 100 RPM aproximadamente cuando la tensión del mismo supera los 24 V en continua de generación, para que circule corriente hacia el banco de baterías. Para analizar puntos intermedios, se realizaron mediciones a 100, 150, 200, 250 y 300 RPM, es decir, en cinco condiciones de funcionamiento para analizar el comportamiento mecánico y eléctrico.

IV. RESULTADOS

A partir de las mediciones realizadas sobre el banco de ensayos se construyeron los gráficos de la figura 3 y 4, en las cuales se muestran la tensión y potencia en continua, respectivamente, en función de la velocidad del generador eléctrico.

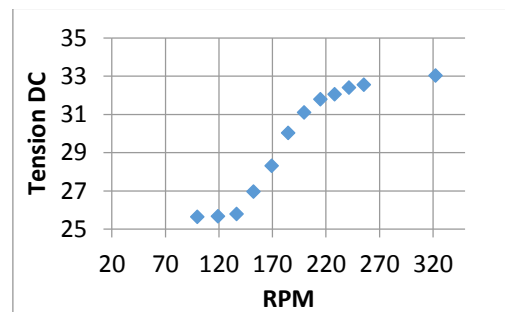


Fig. 3. Curva de tension en DC en funcion de las RPM.

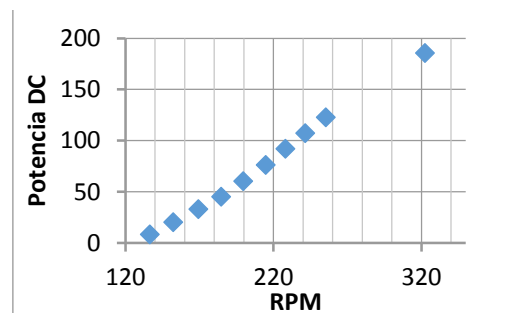


Fig. 4. Curva de potencia en DC en funcion de las RPM

En el análisis de los gráficos se observa que el generador eléctrico comienza a entregar energía eléctrica al banco de baterías de 24 V a partir de 140 RPM, y crece de forma lineal con el aumento de la velocidad hasta alcanzar aproximadamente 180W en continua a 300 RPM.

V. CONCLUSIONES

A partir de la construcción de un banco de ensayos y la automatización del mismo, se pudo realizar la caracterización del funcionamiento de un generador eléctrico de flujo axial para ser implementado en un aerogenerador de baja potencia. Con los resultados del ensayo se pudo determinar cuál será la velocidad de arranque del generador eléctrico para producir energía y cuál será la velocidad máxima que soporte antes de que se activen los sistemas de protección del aerogenerador.

REFERENCIAS

- [1] Agotegaray, J. C., Pinzon Montes, A. L., & Iratchet Prado, M. S. (2017). Prototipo de aerogenerador de eje vertical Giromill. CLADI 2017. Congreso Latinoamericano de Ingeniería .
- [2] Gourieres, D. L. (1983). Energía Eólica. Teoría, concepción y cálculo práctico de las instalaciones. Masson S.A.
- [3] Latoufis, K.C., Messinis, G.M., Kotsampopoulos P.C & Hatzargyriou N.D.(2012). Axial flux permanent magnet generator design for low cost manufacturing of small wind turbinesGenerator axial. Wind Engineering Volume 36, no. 4, 2012 pp 411-442
- [4] Piggot, Hugh (2003). How to build a wind generator - the axial flux alternator windmill plans. Link: https://www.scoraigwind.com/pirate/Hugh_Piggott_Axial-flow_PMG_wind_turbine_May_2003.pdf

Implementación del control de un atenuador programable de RF mediante EDU-CIAA

Camino, Lucas Agustín
Facultad de Ingeniería UNLP
La Plata, Buenos Aires, Argentina
lucascamino95@gmail.com

Figueroa, Lihuen León
Facultad de Ingeniería UNLP
La Plata, Buenos Aires, Argentina
lihuenfigueroa@gmail.com

Corrao, Santiago, Vicente
Facultad de Ingeniería UNLP
La Plata, Buenos Aires, Argentina
santiagocorrao96@gmail.com

Sager, Gerardo Enrique
GridComD Facultad de Ingeniería UNLP
La Plata, Buenos Aires, Argentina
ger@ing.unlp.edu.ar

Resumen—El presente documento describe la implementación de un sistema de control basado en la EDU-CIAA para un atenuador programable. El mismo formará parte de un sistema de mediciones de RF que permitirá la calibración y ensayo de transmisores destinados a sistemas de recolección de datos satelitales con frecuencias de hasta 2GHz.

Palabras clave— instrumentación, atenuador programable, EDU-CIAA

I. INTRODUCCIÓN

A partir de la necesidad de implementar un sistema de mediciones destinado a calibrar y ensayar equipos transmisores de RF, el cual necesita un atenuador variable, y la existencia en el Laboratorio de uno de ellos de marca Aeroflex modelo 3200-1, que permite variar la atenuación entre 0 y 127 dB, se decidió implementar el control del mismo mediante una EDU-CIAA complementada con una placa mezzanina habitualmente llamada “poncho” [1] por sus usuarios. Dentro de la línea de productos que ofrece Aeroflex, existen modelos con interfaz TTL, sin embargo, el modelo del que se dispone necesita niveles de tensión de 12V para su comando. Se decidió que el diseño del “poncho”, incluyera el hardware necesario para accionar el atenuador, a partir de los niveles TTL provistos por la EDU-CIAA, mediante el uso de opto acopladores lo que significa una mejora en la confiabilidad y robustez del instrumento.

II. OBJETIVO

El objetivo de este trabajo es implementar un sistema destinado a controlar la operación del atenuador programable. El control estará comandado manualmente mediante un teclado y un display o bien remotamente a través de interfaces USB, RS232 y/o RS485. Esto permitirá acoplar el conjunto a un sistema de mediciones automatizado.

III. DISEÑO DE HARDWARE

A. Selección del microcontrolador[2]

El controlador principal del sistema fue la placa EDU-CIAA NXP. La elección se debió a factores como la robustez de la placa, la diversidad de interfaces serie que posee (se utilizaron la USB, la RS232 y la RS485), así como los numerosos puertos de propósito general. Asimismo se simplifica el diseño de hardware al estar prevista la interfaz con una placa de circuito impreso adicional, habitualmente denominada “poncho” por los usuarios de la EDU-CIAA. No se puede dejar de destacar la existencia de abundante documentación en cuanto al hardware[3] y al software del proyecto CIAA.

B. Diseño del “Poncho”

Fueron necesarios el diseño y la construcción de un poncho específico para la conexión y protección de los distintos componentes y elementos, así como para el acondicionamiento de las señales de tensión (Fig. 1).

Un requerimiento importante del proyecto es la tensión de 12V necesaria para accionar el atenuador, en efecto, al tratarse de un componente costoso y delicado se decidió separar el control del mismo de la EDU-CIAA por medio de opto acopladores. La tensión de alimentación de ésta última es de 5V y las salidas de los puertos proveen un nivel lógico alto de 3,3V. Se optó por establecer la tensión de alimentación externa en 15V de continua y utilizar sendos reguladores analógicos de 12V y 5V para proveer las tensiones necesarias.

Desde el punto de vista funcional, podemos dividir la placa en distintos sectores:

- 1) *Display y teclado*: se accede a los mismos a través de zócalos conectados a los puertos de entrada/ salida correspondientes[4].
- 2) *Interfaz RS232*: se utiliza un circuito HIN2320 para proveer los niveles necesarios para la comunicación a partir de los niveles LV-TTL que provee la EDU-CIAA.
- 3) *Interfaz USB*: Se utiliza la interfaz auxiliar de debug, a la que se accede a través de la UART0 de la EDU-CIAA.
- 4) *Interfaz RS485*: Se utiliza la interfaz provista por la EDU-CIAA.

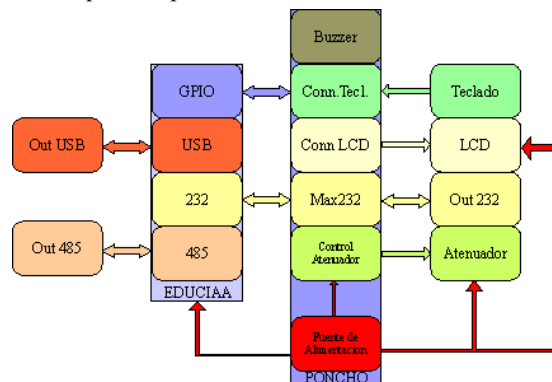


Fig. 1. Diagrama en bloques del sistema. Poncho integrado.

- 5) *Salida hacia el atenuador* [5]: se acondicionan las señales que comandan el atenuador, mediante el uso de optoacopladores, ya que tiene la ventaja de aislar completamente la EDU-CIAA del atenuador, evitando que corrientes no deseadas puedan dañar la placa controladora. Para compatibilizar la corriente que es capaz de entregar la EDU-CIAA en sus puertos de salida con la corriente necesaria para accionar las entradas del atenuador, se diseñó un circuito auxiliar que permite que un transistor operando como llave,

proporciona la corriente necesaria al optoacoplador sin cargar excesivamente las salidas de la EDU-CIAA. Dado que el dispositivo necesita 8 entradas que deben ser manejadas en forma independiente, se replicó el circuito otras tantas veces.

IV. DISEÑO DEL SOFTWARE

A los efectos del diseño de software se utilizó la biblioteca sAPI, desarrollada dentro del proyecto "Firmware_v2" [6]. Esta biblioteca nos permite tener una abstracción cómoda para el entendimiento y la programación de la placa y los periféricos utilizados.

Un aspecto importante que merece ser mencionado en el presente documento es el diseño, modelado, y elaboración del software implementado. El mismo fue desarrollado siguiendo una arquitectura *event-triggered* [7], que, generalmente implica el uso de múltiples interrupciones, cada una asociada a eventos periódicos específicos (como desborde de timers), o a eventos aperiódicos (como puede ser la entrada o salida de datos al sistema), lo que se ajusta perfectamente a éste caso en particular.

A su vez, se adoptó un modelo de máquinas de estado finitas; una principal, que maneja display, teclado y control del atenuador, y una secundaria, independiente de la primera, que se ocupa de configurar y manejar las interfaces de comunicación serie.

De esta manera, podría entenderse el software como constituido por capas (Fig. 2), encontrando así en la capa superior la máquina de estados principal que controla al atenuador, muestra el valor de atenuación en el display LCD y atiende las entradas de teclado. En una capa inferior, encontramos la segunda máquina de estados mencionada anteriormente, que se encarga de controlar el flujo de información con las interfaces de comunicación serie, esto es, tanto validar el comando recibido, como procesarlo adecuadamente y devolver la correspondiente respuesta; ésta es quien se vale de los drivers para comunicarse con los distintos periféricos involucrados en las tareas mencionadas anteriormente.

El lenguaje de programación utilizado para la codificación del algoritmo fue C, dentro del entorno de desarrollo Eclipse, en una de sus versiones adaptadas específicamente para el trabajo con la CIAA. Afortunadamente, contamos con un Firmware previamente desarrollado, que es de público acceso, y completamente gratuito, que nos proporciona facilidades a la hora de desarrollar el software dentro del entorno de programación.

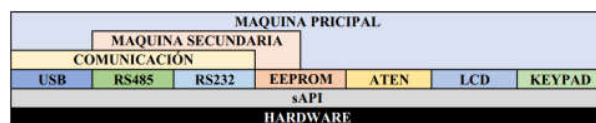


Fig. 2. Diseño de software por capas.

A la hora de la comunicación con las distintas interfaces serie, trabajando a modo *half-duplex*, decidimos seguir un formato estándar fijo, tanto para los comandos que envía el sistema como para los que recibe, con el objetivo de no solo facilitar el algoritmo, sino también para simplificar la posible comunicación con otros sistemas. Dicho formato, es el siguiente: "*nombreComando,parámetro.*"; en donde *nombreComando* hace referencia a qué parte del sistema se quiere configurar o conocer, mientras que en *parámetro* se encuentra un valor numérico o un operador que será de vital importancia para el intérprete de comandos. A continuación,

algunos ejemplos de los posibles comandos: con "*ATEN,100.*" se establece el valor de la atenuación en 100 dB ; en cambio, con "*BR232,?*" consultamos el *baudrate* al que está trabajando la interfaz serie RS232.

Por otra parte, las interfaces de comunicación serie poseen, a nivel de software, varios modos de trabajo, correspondientes a tres estados distintos: *PRIMARIA*, *SECUNDARIA*, *APAGADA*. Por aquella interfaz que se encuentre en el primero de ellos, el sistema será capaz tanto de enviar datos, como de recibir; si es, en cambio, *SECUNDARIA*, éste sólo podría recibirlos. Finalmente, si la interfaz se encuentra *APAGADA*, directamente se encuentra deshabilitada tanto la transmisión como la recepción de datos. Con respecto a las tasas de transmisión de cada interfaz, cabe aclarar que también son configurables por comando, o bien localmente por el teclado. Los parámetros de configuración, a su vez, son almacenados en memoria no volátil (EEPROM) para facilitar la operación del instrumento.

V. RESULTADOS

Todos los ensayos realizados resultaron satisfactorios, a través de ellos se pudo validar implementación del hardware y del firmware de manera que el prototipo es funcional y se encuentra actualmente en uso (Fig. 3).

VI. CONCLUSIONES

Como mejora posible del equipo, que actualmente está diseñado para uso en condiciones controladas de laboratorio, se podrían realizar algunas modificaciones, ya sea para su elaboración como producto o bien para su utilización portátil. En ese caso se podría reemplazar la EDU-CIAA por un hardware desarrollado en base a un procesador similar que permita la utilización de las mismas interfaces.

También para el caso de utilización como instrumento portátil, alimentado a baterías, podrían utilizarse algunas de las características de bajo consumo que provee el microcontrolador, para aumentar la duración de las mismas.

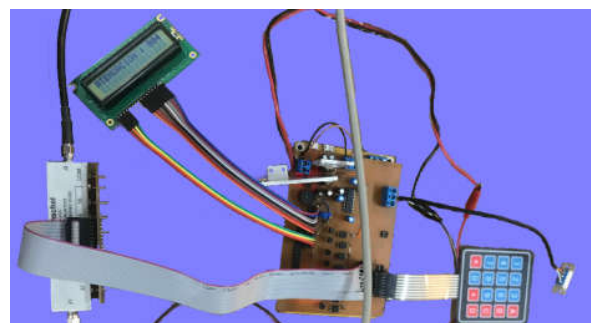


Fig. 3. Sistema en funcionamiento.

REFERENCIAS

- [1] Modelos de Ponchos ya existentes, <https://github.com/ciaa/Ponchos>
- [2] Manual del microcontrolador lpc4337, <https://www.alldatasheet.es/datasheet-pdf/pdf/466540/NXP/LPC4337.html>
- [3] Esquemáticos EDU-CIAA, <http://www.proyecto-ciaa.com.ar/devwiki/doku.php>
- [4] Mapeo de Puertos de Firmware, https://github.com/ciaa/firmware_v2/blob/master/sapi_examples/documentation/sAPI%20mapeo%20de%20perifericos%20EDU-CIAA-NXP.pdf

- [5] *Manual del Atenuador Aeroflex/Weinschel 3200 series*, web:
<https://www.apitech.com/products/rf-solutions/rf-components/attenuators/programmable-attenuator/electromechanical-relay-switched/3200-programmable-attenuator/>
- [6] *Documentación del Firmware utilizado*,
https://github.com/ciaa/firmware_v2/blob/master/sapi_examples/documentation/sAPI_reference.pdf
- [7] Pont, M.J., “*The Engineering of Reliable Embedded Systems: LPC 1769 edition*”, SafeTTY Systems. April 2015.

Diseño y construcción de un tacómetro digital mediante el uso de una placa Arduino

Victor C. Juarez, *Student Member IEEE*, Estudiante de Ingeniería Eléctrica
Universidad Tecnológica Nacional Facultad Regional de Tucumán, victorcjuarez@ieee.org

Abstract—Esta investigación apunta al diseño y construcción de dicho instrumental priorizando el bajo error en la medición y facilidad de uso. Una placa Arduino será usada para leer y traducir la señal entregada por un taco generador

Index Terms—Maquina Generalizada, Tacómetro, Arduino.

I. INTRODUCCIÓN

LA medición de velocidad es un parámetro a tener en cuenta cuando hablamos de máquinas eléctricas rotativas. Históricamente, esta medición se hace mediante la toma de una señal eléctrica generada por un taco generador solidario al eje de la máquina. En otros casos, se realiza mediante un sistema mecánico. Numerosos avances tecnológicos llevaron a reemplazar estos tacómetros analógicos por digitales. Entre ellos, se encuentran mejoras en la precisión y exactitud, en virtud de la utilización, de los micro controladores. Estos tacómetros digitales son capaces de tomar lectura directamente del eje de la máquina con menor error, volviendo innecesario el uso de tacos generadores para la medición de velocidad. En el apartado de los micro controladores, Arduino demuestra ser una plataforma idónea para la adquisición de datos. La carencia de un instrumento práctico a la hora de medir velocidad llevó al diseño y desarrollo de un sistema que combina tecnologías de tiempos distintos ya que, para efectuar la medición, se toma la señal generada por el taco generador, la cual es acondicionada y traducida por la placa Arduino.

II. METODOLOGÍA

A. Taco generador

Al tener acoplado la máquina eléctrica un dispositivo que posee una señal que varía tanto en amplitud como periodo, se debieron efectuar dos ensayos con el fin de determinar sus curvas características de tensión y frecuencia respecto a la velocidad. Ver figura 1 y 2. [1]

Como es observado, existe una linealidad respecto a la tensión y frecuencia. En consecuencia, es posible medir independientemente para obtener la velocidad.

B. Motor de CC

La unidad motriz es un motor de CC en conexión derivación. Montado sobre la bancada de una máquina generalizada [2][3]. Esta configuración es adecuada para aplicaciones en donde se requiere depender de un rango apreciable de velocidades por medio del control de campo.[4]

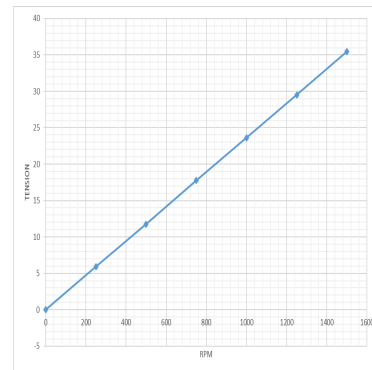


Fig. 1. Tensión vs RPM

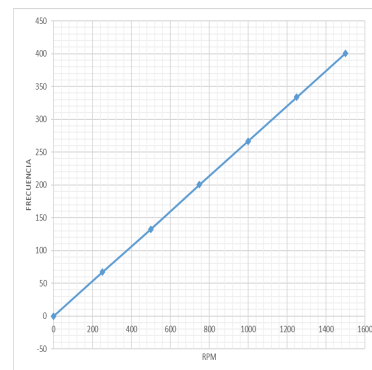


Fig. 2. Frecuencia vs RPM

C. Instrumento patrón

El instrumento es un tacómetro óptico SKF TKRS-20. Dicho dispositivo permite medir las revoluciones en el eje del motor de CC sin contacto.[5]

III. DISEÑO DEL SENSOR

La señal entregada por el taco generador tiene dos parámetros que varían respecto a la velocidad angular. Esto genera un problema a la hora de efectuar la lectura de señal. Para solucionarlo se realizan dos mediciones usando los sensores, obtenidos los resultados son contrastados posteriormente mediante un instrumento patrón.

A. Sensor de Tensión

Consiste en un módulo transformador de voltaje alterno ZMPT101B [6] que permite medir hasta 240V. El mismo es

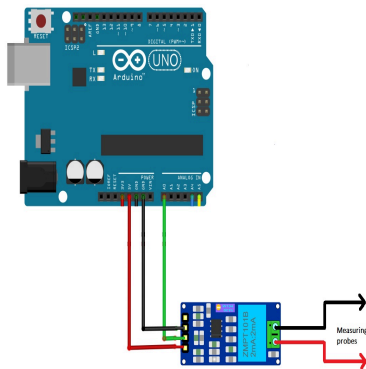


Fig. 3. Esquema de conexión del Sensor de tensión.

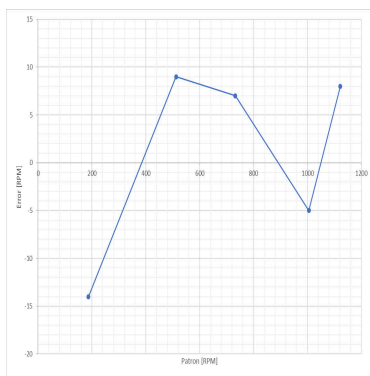


Fig. 4. Error del Sensor de Tensión

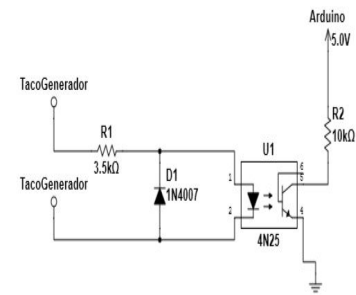


Fig. 5. Circuito del Sensor de Frecuencia

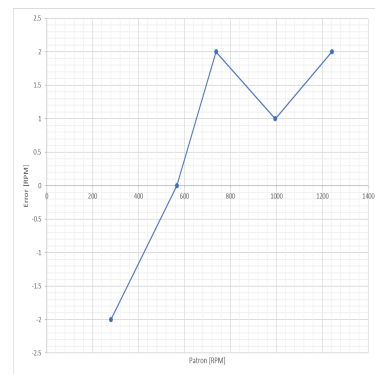


Fig. 6. Error del Sensor de Frecuencia

conectado a la placa Arduino [7]. Esquema de conexión Fig. 3.

Para obtener los distintos puntos de calibración se realizan variaciones en las revoluciones desde un mínimo al valor máximo, obteniendo una serie de puntos en los cuales se aprecia el error del sensor de tensión respecto a la medición patrón.

En el análisis de los datos obtenidos en el ensayo, se aprecia un error significativo a bajas revoluciones con respecto al instrumento patrón. Esto se debe a las señales inestables generadas por la máquina a bajas revoluciones, lo cual dificulta la lectura de tensión. Ver Fig. 4.

B. Sensor de Frecuencia

El sensor consiste en un sistema de censado por cero, el cual a través de un opto acoplador [8] envía un pulso cada vez que la señal eléctrica del taco generador pasa por cero. Dichos pulsos son capturados por la placa Arduino, la cual los contabiliza en un tiempo de 1 [s] y devuelve el valor de la frecuencia eléctrica de la señal [9]. Ver esquema Fig. 5.

Para el contraste, se utiliza el mismo sistema que para el sensor de tensión, obteniendo así los distintos puntos de error del dispositivo. Ver Fig. 6.

Observando los datos del ensayo, es posible apreciar menos error cuando es comparado al sensor de tensión.

IV. CONCLUSIONES

Al comparar los errores de ambos dispositivos se eligió el sensor de frecuencia sobre el de tensión al obtener una medición más exacta y estable, de la señal producida por el taco generador. Ver Fig. 7.

El inconveniente que se encontró en este dispositivo fue que a bajas revoluciones (aproximadamente menor a 200 [RPM]), el dispositivo pierde lectura de los datos. Este problema se corrigió modificando el circuito mediante un selector para bajas revoluciones. Ver Fig. 8.

Armado el dispositivo, se pasó a realizar un ensayo de contraste para comparar las diferencias entre el instrumento

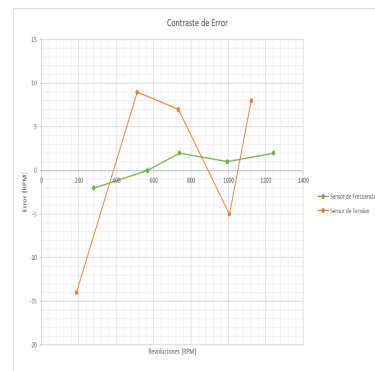


Fig. 7. Comparación de errores

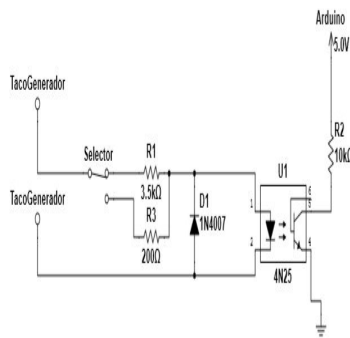


Fig. 8. Circuito final

patrón y el tacómetro digital, el error obtenido en los distintos puntos de medición no supera el 1% de error, cumpliendo los objetivos planteados.

RECONOCIMIENTOS

La realización de esta investigación fue posible, en primer lugar, a la cooperación brindada por el Ing. Pablo Duberti, que gracias a sus aportes técnicos y apoyo se logró hacer este proyecto. También se desea agradecer el total apoyo del laboratorio de Máquinas Eléctricas y su cuerpo docente.

REFERENCES

- [1] Luis Marin y Edgar Caraballo (25 de mayo 2007). Mediciones Industriales. [Online]. Available: <http://miluisyedgar.blogspot.com/2007/05/sensor-tacogenerador.html>
- [2] Ingeniería Eléctrica Experimental Luis A. Salazar. por Salazar, Luis A. Editor: Barcelona Marcombo 1973.
- [3] "Máquinas generalizadas", Máquinas Eléctricas, Departamento Eléctrica, Universidad Nacional de Tucumán.
- [4] Ing. Miguel Pozueta (2017). Máquinas de corriente continua. [Online]. Available: <https://personales.unican.es/rodrigma/PDFs/Maquinas%20cc.pdf>
- [5] Catalogo SKF TKRS 20.[Online]. Available: http://www.reliabilitydirectstore.com/v/vsfiles/RDIStorePDF/TKRS20_User.pdf
- [6] Datasheet ZMPT101B.[Online]. Available: <http://5nr0rxhmqijik.leadongcdn.com/attachment/kjilKBmoioSRqlkjoipSR7ww7fgzb73m/ZMPT101B-specification.pdf>
- [7] Datasheet Arduino UNO.[Online]. Available: https://www.fecegypt.com/uploads/dataSheet/1522237550_arduino%20uno%20r3.pdf
- [8] Datasheet 4N25.[Online]. Available: <http://www.vishay.com/docs/83725/4n25.pdf>
- [9] Nick Gammon (22 de marzo 2019). attachInterrupt(). [Online]. Available: <https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt/>

Tecnologías Peltier para Control de Temperatura Aplicado a Sistemas de Refrigeración Láctea.

Castillo Vinces, Wickson Janniro
Facultad de Ingeniería Electrónica y
Telecomunicaciones
Universidad Nacional de Piura
Piura, Perú
jannirocv@gmail.com

Palacios Curay, Marco Antonio
Facultad de Ingeniería Mecánica-
Eléctrica
Universidad de Piura
Piura, Perú
marco9309@gmail.com

Roosebelt Yonatan, Carmen Aguirre
Facultad de Ingeniería Electrónica y
Telecomunicaciones
Universidad Nacional de Piura
Piura, Perú
rjonatansa@outlook.es

Resumen— La geografía extensa y accidentada de los andes peruanos representa una limitación considerable en la cadena de transporte de la leche. Además, las altas temperaturas durante el transporte incrementan la degradación de sus propiedades nutricionales. En este contexto, la implementación de una etapa de transporte con temperatura controlada, es necesaria para conservar sus propiedades nutricionales en su traslado hacia la planta de procesamiento. En este proyecto se desarrolló un sistema de refrigeración de estado sólido modular que permite la conservación de la temperatura interna de los recintos lácteos basándose en tecnologías Peltier. Este sistema se basa en la inducción controlada de corriente en las celdas, para generar un diferencial de temperatura en las zonas adyacentes al sistema de recirculación, ocasionando un descenso controlado de la temperatura y manteniendo los estándares de calidad del producto a lo largo de todo su recorrido.

Palabras Claves—Celdas peltier, sistema de refrigeración, control de temperatura, control on/off, optoacoplador.

I. INTRODUCCIÓN

Actualmente, la producción de leche se realiza de manera artesanal, es decir, el ordeño es manual y el producto es almacenado en tanques de acero inoxidable, proporcionando una barrera contra la contaminación externa. Sin embargo, son inadecuados para mantener la leche a su temperatura de conservación 4°C [1][2].

La refrigeración de estado sólido es relativamente nueva. Sin embargo, a nivel internacional se han desarrollado aplicaciones industriales como los micro-refrigeradores termoeléctricos y los generadores de agua atmosférica [3][4].

El efecto Peltier consiste en la creación de una diferencia térmica a partir de una diferencia de potencial eléctrico [5]. Asimismo, la dirección de la transferencia de calor es controlada por la polaridad de la corriente. Es decir, al invertir la polaridad cambiará la dirección de transferencia del calor absorbido o producido [6].

II. DESCRIPCIÓN DEL SISTEMA

A. Sistema de Refrigeración y Recirculación

El sistema de enfriamiento consta de un tanque Cornelius de 19 litros, una bomba peristáltica y un módulo de enfriamiento termoeléctrico Peltier (LA-160-24).

La bomba peristáltica trabaja con un flujo de 2.5 litros por minuto y posee un requerimiento energético de 24W. Asimismo, el módulo de enfriamiento termoeléctrico tiene un área de enfriamiento de 272 cm² y posee un requerimiento energético de 160W y 7A.

El proceso inicia cuando la bomba peristáltica, impulsa la leche desde el tubo de succión y lo redirección hacia las celdas de enfriamiento para retornar por el tubo de inyección. Este ciclo (Figura 1) se repite hasta que el líquido alcanza la temperatura del rango de funcionamiento.

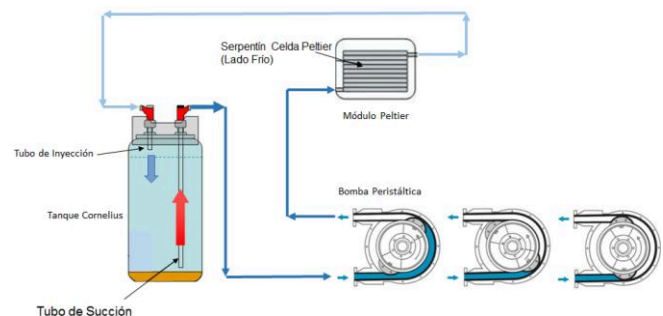


Fig. 1. Esquema de funcionamiento del sistema de Refrigeración y Recirculación.

El consumo total de potencia del sistema es 184W. Para satisfacer esta demanda, se ha dispuesto de un arreglo de 2 baterías de 115 Ah y 12VDC colocadas en serie para obtener una tensión de 24 VDC y suministrar el sistema.

B. Sistema Control y Monitoreo

Este sistema cuenta con sensores RTD's (PT100) y sensores de Corriente (ACS712) para el monitoreo de las variables de control. Asimismo, se implementó un control On/Off con el sistema integrado CY8CKIT-059 PSoC 5LP para accionar el Sistema de Enfriamiento.

Este sistema embebido cuenta con un circuito de conmutación opto-acoplado que envía una señal de control en función de la temperatura medida por la RTD (PT100), permitiendo la activación del sistema de enfriamiento cuando la temperatura del sistema es mayor a la temperatura máxima de transporte (8°C), y desactivándolo cuando es menor a la temperatura mínima (4°C) del rango establecido.

III. IMPLEMENTACIÓN DEL SISTEMA CONTROL

La electrónica del sistema de refrigeración ha sido distribuida en un tablero eléctrico, integrando los siguientes elementos:

1) *La Fuente de Alimentación*: Se implementó una fuente de alimentación DC-DC de 24V a 4.9V, basándose en el regulador 7805.

2) *Tarjeta de Control*: Se habilitaron 4 conexiones RTD's para realizar las mediciones de temperatura e implementar el control automático del sistema de refrigeración (Figura 2).

Este trabajo ha sido financiado por el Fondo de Investigación y Desarrollo para la Competitividad del Ministerio de Producción del Perú, a través del proyecto 333-FIDECOM-PNIPC-PIPEI-2014.

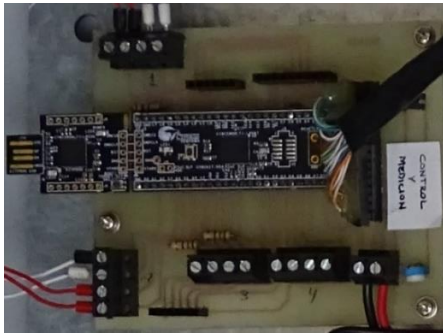


Fig. 2. Sistema de Control PSoC 5LP On/Off.

3) *Tarjeta de Potencia:* Para controlar el voltaje y la alta corriente (10 Amp.) requerida por la carga de los actuadores del sistema, se implementó una tarjeta de potencia que optocopla las señales de control de baja corriente con las de alta potencia de las cargas.

Finalmente, se integraron todos los componentes en un tablero de control (Figura 3).

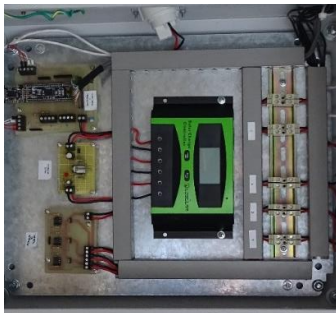


Fig. 3. Tablero de Control del Sistema de Refrigeración.

IV. RESULTADOS

El sistema desarrollado, en su estado actual (Figura 4), permite mantener las condiciones óptimas de refrigeración con una autonomía de 6 horas aproximadamente. Proporcionando a los productores de leche el tiempo suficiente para trasladar su producto en óptimas condiciones hasta la planta de procesamiento.



Fig. 4. Sistema Final de Refrigeración de Estado Sólido Modular.

En la primera prueba se realizó un recorrido de 12 kilómetros, desde el pueblo de Olleros hasta la planta pasteurizadora, logrando una temperatura mínima de 9.57°C (Figura 5) y en la segunda prueba se realizó un recorrido de 7.5 kilómetros, desde el pueblo de Rodeo Pampa hasta la planta pasteurizadora, logrando una temperatura mínima de 10.09°C (Figura 6).

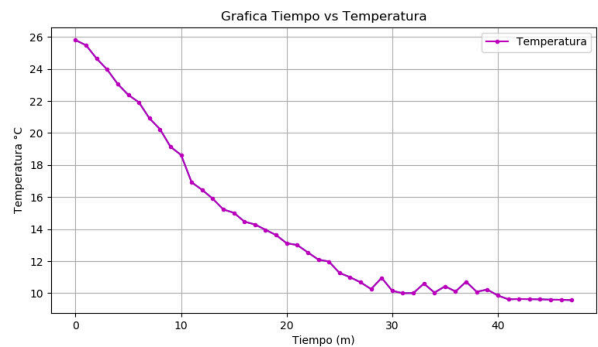


Fig. 5. Variación de Temperatura en la Primera Prueba.

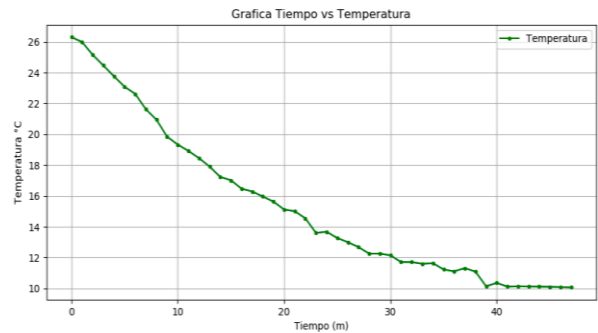


Fig. 6. Variación de Temperatura en la Segunda Prueba.

Durante los primeros 20 minutos, en ambas pruebas, se ha logrado disminuir en 45% la temperatura del sistema, respecto a la temperatura ambiente. Luego de ese periodo, el comportamiento de la temperatura es asintótico, manteniendo su temperatura durante todo su recorrido.

V. CONCLUSIONES

La implementación de sistemas de refrigeración por tecnologías Peltier, representa una alternativa adecuada para el transporte en frío de productos lácteos en zonas de geografía accidentada, ya que reduce la temperatura de la leche hasta 10°C respecto a su transporte tradicional en tanques Cornelius, donde las temperaturas oscilan entre 25 °C y 35 °C.

La versatilidad del sistema de refrigeración de estado sólido le permite adaptarse a cualquier modalidad de transporte, incluso mochilas. Esto nos permite plantear un futuro desarrollo para el transporte de vacunas hacia zonas inaccesibles para el mercado farmacéutico tradicional.

REFERENCIAS

- [1] A. Poghossian, H. Geissler and M. J. Schöning, "Rapid methods and sensors for milk quality monitoring and spoilage detection," *Biosensors and Bioelectronics*, 2019.
- [2] D. Tian; C. Li, "Risk assessment of raw milk quality and safety index system based on primary component analysis," *Sustainable Computing: Informatics and Systems*, Vol. 21, pp. 47-55, 2019.
- [3] D. Sun, G. Liu, L. Shen, H. Chen, Y. Yao, S. Jin, "Modeling of high power light-emitting diode package integrated with micro-thermoelectric cooler under various interfacial and size effects," *Energy Conversion and Management*, Vol. 179, pp. 81-90, 2019.
- [4] A.H. Shourideh; W.B. Ajram; J.A. Lami; S. Haggag; A. Mansouri, "A comprehensive study of an atmospheric water generator using Peltier effect," *Energy Conversion and Management*, Vol. 6, pp. 14-26, 2018.
- [5] G. Casano, S. Piva, "Experimental investigation of a Peltier cells cooling system for a Switch-Mode Power Supply," *Microelectronics Reliability*, Vol. 79, pp. 426-432, 2017.
- [6] C. Lundgaard; O. Sigmund, "Design of segmented thermoelectric Peltier coolers by topology optimization," *Applied Energy*, Vol. 239, pp. 1003-1013, 2019.

Control de motores DC sin escobillas basado en un microcontrolador de doble núcleo asimétrico

R. Ghignone, F. Calá, S. Domanico, J. Giampetruzzi, C. Juárez
Universidad Tecnológica Nacional
Facultad Regional Haedo
Haedo, Buenos Aires, Argentina
embebidos@frh.utn.edu.ar

Resumen—El presente documento describe el proceso de diseño, desarrollo y verificación de un sistema de potencia para motores de corriente continua sin escobillas, especialmente diseñado para su uso en vehículos eléctricos, considerando el rápido crecimiento de las tecnologías de movilidad eléctrica. El diseño propuesto implementa funcionalidades adicionales clave para este tipo de aplicaciones, como frenado regenerativo y telemetría mediante una aplicación móvil. El prototipo construido fue verificado mediante ensayos de laboratorio, y constituye el Proyecto Final de los autores del presente trabajo para la carrera de Ingeniería Electrónica.

Índice de términos—BLDC, freno regenerativo, vehículo eléctrico, electrónica de potencia, ARM Cortex, multicore, CIAA.

I. INTRODUCCIÓN

En los últimos años, los efectos del cambio climático a nivel mundial han motivado a investigar, desarrollar y financiar nuevas tecnologías que reduzcan las emisiones de efecto invernadero [1]. Por ejemplo, las actividades relacionadas al transporte representan el 32% del consumo de energía global, y los Vehículos Eléctricos (VE) aparecen como la solución de mayor crecimiento para reducir las emisiones derivadas de la quema de combustibles fósiles en vehículos convencionales [2].

Como consecuencia del intenso trabajo en investigación y desarrollo de VE, surgieron varias tecnologías destinadas a mejorar su eficiencia o usabilidad, por ejemplo, los sistemas de frenado regenerativo o el monitoreo y control a través de aplicaciones móviles [3]. Bajo estas consideraciones, se desarrolló un controlador para motores de corriente continua sin escobillas (BLDC, del inglés *brushless DC*) orientado a su uso en vehículos eléctricos de baja potencia. El diseño propuesto soporta tanto frenado regenerativo como monitoreo remoto. Para su control se utilizó un microcontrolador LPC4337, que combina una alta relación desempeño/costo gracias a su arquitectura de doble núcleo.

II. VISTA GENERAL DEL PROTOTIPO

A. Módulos del proyecto

La Fig. 1 muestra un diagrama en bloques del prototipo en el cual podemos identificar un conjunto de módulos principales, que desarrollaremos con detalle más adelante.

- **Convertor DC/DC bidireccional:** regula el flujo bidireccional de potencia entre la batería y el motor.
- **Convertor DC/AC bidireccional:** genera una señal alterna trifásica a partir de la tensión de batería y rectifica la tensión provista por el motor durante el frenado. Está conectado con la etapa anterior mediante el *bus DC*.
- **Unidad de control:** mide tensiones y corrientes en el sistema, genera las señales de conmutación para la electrónica de potencia, y se comunica con el usuario.

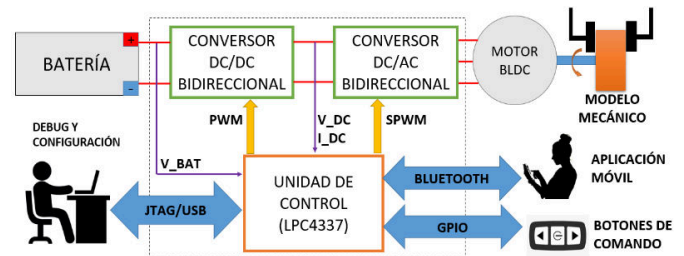


Fig. 1 – Arquitectura general del prototipo

- **Motor BLDC:** para este prototipo se utilizó un motor BLDC A2212/13T [4] tipo *sensorless*, el cual puede entregar hasta 150 W de potencia mecánica.
- **Batería:** este modelo emplea una batería de polímero de litio (LiPo) de 2200 mAh, con una tensión de 11,1V y una capacidad de descarga de 30C.
- **Interfaz de usuario:** la unidad de control se conecta de forma serial con un módulo Bluetooth HC-06 [5], que se comunica con la aplicación móvil desarrollada para dispositivos Android mediante la herramienta *Android Studio* [6].

III. ETAPAS DE POTENCIA

A. Convertor DC/AC Bidireccional

El motor BLDC requiere para su funcionamiento de una alimentación trifásica [7] o de un tren de pulsos sincronizado con la posición del rotor [8]. A fines de mantener un diseño simple y a la vez capaz de utilizar ambas técnicas se implementó un inversor trifásico básico que se muestra en la Fig. 2. El mismo emplea modulación SPWM [9] (del inglés *Sine Pulse Width Modulation*) y tiene dos modos de trabajo:

- **Modo motor:** los transistores son conmutados por la unidad de control y alimentan al motor.
- **Modo regenerativo:** los transistores permanecen apagados, y los diodos en antiparalelo rectifican la fuerza contraelectromotriz inducida en el motor.

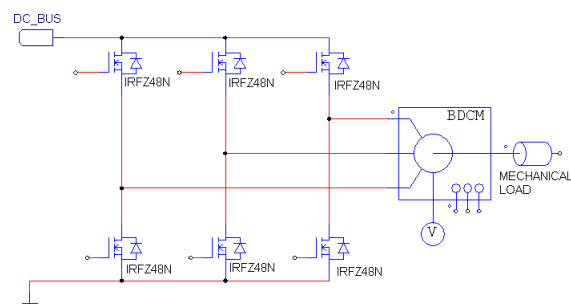


Fig. 2 – Esquema del convertor DC/AC bidireccional

B. Conversor DC/DC bidireccional

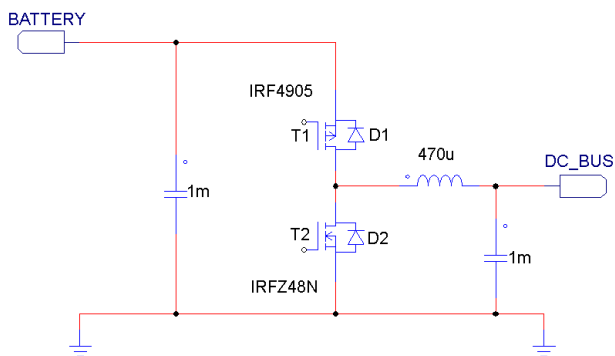


Fig. 3 – Esquema del conversor DC/DC bidireccional

Como se mencionó en la sección I, el conversor DC/DC tiene una doble función: debe reducir la tensión de batería para proveer un nivel de tensión en el bus DC adecuado para el inversor, pero también debe aumentar la tensión entregada por el motor y rectificada por el conversor DC/AC para recargar la batería durante el frenado regenerativo.

Ambos objetivos se alcanzan mediante la estructura semipunte propuesta en la Fig. 3, que muestra un conversor DC/DC bidireccional capaz de implementar una topología Buck o Boost, según el flujo de potencia y el estado de sus transistores [10].

Al igual que el circuito DC/AC, este módulo también tiene dos modos de trabajo, según el sentido del flujo de potencia:

- **Modo motor:** T2 permanece apagado, mientras que T1 y D2 forman un conversor Buck que reduce la tensión de la batería al valor necesario en el bus DC.
- **Modo regenerativo:** T1 permanece apagado, mientras que T2 y D1 forman un conversor Boost que aumentan la tensión del bus DC para cargar la batería.

IV. UNIDAD DE CONTROL

A. Placa EDU-CIAA-NXP y microcontrolador LPC4337

Para la unidad de control se empleó una placa EDU-CIAA-NXP [11], basada en un microcontrolador LPC4337 [12], el cual contiene dos núcleos ARM Cortex de 32 bits, así como un gran número de periféricos (USART, I2C, SPI, ADC, DAC, Ethernet, SCT, GPIO, etc.) que permiten su uso en un amplio espectro de aplicaciones.

B. Arquitectura de Firmware

Se aprovecharon tanto el núcleo ARM Cortex M4F [13] como el M0 [14] del microcontrolador para repartir la carga de trabajo y mejorar el desempeño de la unidad de control. El firmware de control fue escrito en C, y las secciones críticas se optimizaron reescribiéndolas en lenguaje ensamblador y usando las instrucciones SIMD (del inglés *Single Instruction Multiple Data*) disponibles en el M4F. La Fig. 4 muestra cómo se dividen las tareas y periféricos entre los núcleos del microcontrolador.

V. RESULTADOS

El prototipo fue ensayado mediante una maqueta mecánica para distintos regímenes de trabajo, verificando que cumple con el comportamiento requerido. La relación tensión-frecuencia impuesta por el lazo de control del convertidor DC/DC se ajusta a la recta configurada en la unidad de control, y las señales de tensión y corriente generadas concuerdan con las simulaciones realizadas.

Además, la optimización del controlador digital mediante instrucciones SIMD redujo su tiempo de procesamiento a menos del 30% de su valor original [15]. Esto permite dedicar tiempo del núcleo a otras tareas o dejarlo en un estado inactivo y reducir su consumo energético.

VI. CONCLUSIONES

A lo largo de este trabajo se describe el desarrollo de una unidad completa de control y potencia para motores BLDC en vehículos eléctricos. El diseño partió de un estudio de las tendencias actuales y la definición de una arquitectura capaz de cumplir con las especificaciones surgidas de ese análisis.

De acuerdo a los requerimientos de flujo de potencia entre la batería y el motor, se desarrollaron dos convertidores bidireccionales de potencia, los cuales fueron verificados exitosamente. El uso de la arquitectura doble núcleo del LPC4337 para implementar las tareas de control del sistema, impactó favorablemente en la arquitectura del sistema propuesto, permitiendo una implementación más eficiente.

Se podría aprovechar la misma arquitectura del sistema, separando interfaces físicas y procesamiento de señales en cada núcleo, para implementar estrategias de control más complejas, por ejemplo, de control orientado al campo [16].

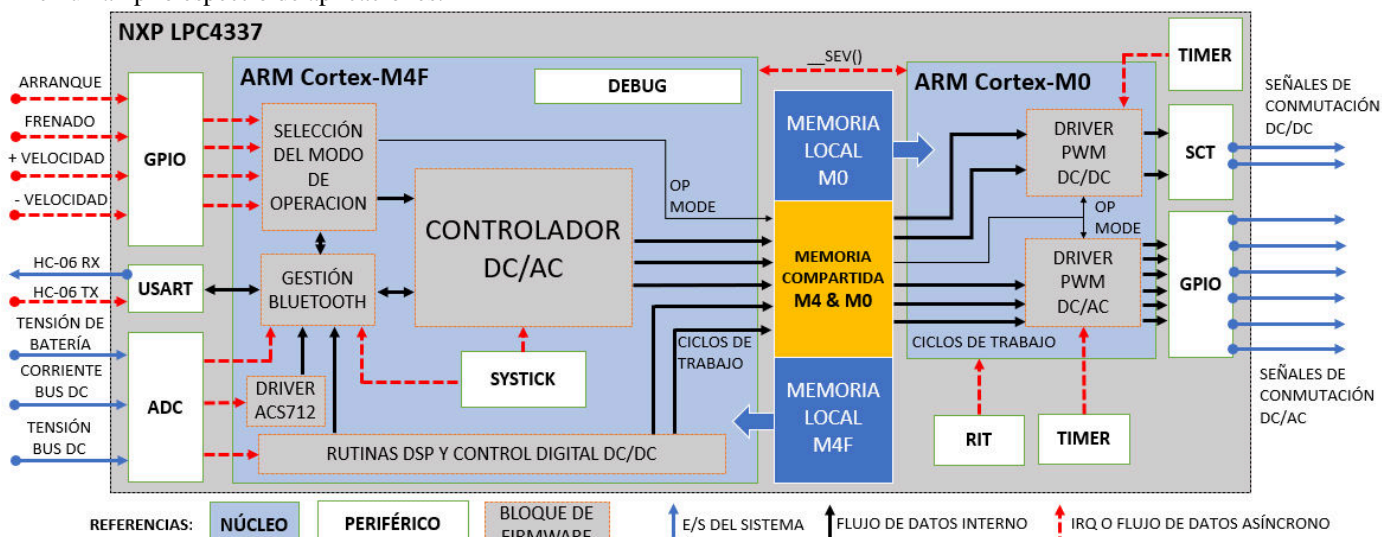


Fig. 4 – Arquitectura de firmware y uso de hardware del LPC4337

VII. REFERENCIAS

- [1] International Energy Agency, “*World Energy Investment 2017*” (executive summary), 2017
- [2] Renewable Energy Policy Network for the 21th Century (REN21), “*Renewables 2018 Global Status Report*”, 2018
- [3] J. Cody et al, “*Regenerative Braking in an Electric Vehicle*”, University of South Australia, 2009.
- [4] A2212/13T Technical Data
- [5] Guangzhou HC Information Technology Co. Ltd., “*HC-06 Product Data Sheet*”, Rev. 2.0
- [6] Android Studio : <https://developer.android.com/studio/>
- [7] P.C. Krause, O. Wasynczuk, S. D. Sudhoff, “*Anaylsis of Electric Machinery and Drive Systems*” (IEEE Series on Power Engineering), 2nd edition, Wiley & Sons Inc, 2002, chapter 6.
- [8] W. Brown, “*Brushless DC Motor Control Made Easy*” (AN857 Application Note), Microchip Technology Inc., 2002
- [9] H.M. Rashid, “*Power Electronics Circuits, Devices and Applications*”, 2nd edition, 1995, Prentice Hall, chapter 10
- [10] A. Agatep, M. Ung, “*Design and simulation of V2G Bidirectional Inverter and DC-DC Converter*”, Electrical Engineering Department, California Polytechnic State University, 2011
- [11] Proyecto CIAA: http://www.proyecto-ciaa.com.ar/index_en.html
- [12] “*UM10503 LPC43xx ARM Cortex-M4/M0 dual-core microcontroller User Manual*”, NXP Semiconductors, Rev 1.2, 2012
- [13] “*Cortex-M4 Technical Reference Manual*”, ARM Ltd., rev. r0p0, 2010
- [14] “*Cortex-M0 Technical Reference Manual*”, ARM Ltd., rev. r0p0, 2010
- [15] R. Ghignone, I. Castellucci Vidal, J. Giampetruzzi, F. Larosa, “*Implementación y optimización de una biblioteca embebida para receptor GPS*”, Universidad Tecnológica Nacional, Facultad Regional Haedo, 2016.
- [16] E. Klintberg, “*Comparison of Control Approaches for Permanent Magnet Motors*” (Master of Science Thesis), Chalmers University of Technology, Sweden, 2013.

Sistema de monitoreo de calidad de aire y ruido ambiental

Agüero, Agustín¹; Behar, Christian¹; Lanzillotti, Leandro¹; Vázquez, Matías¹; Zaradnik, Ignacio¹; Lupi, Daniel^{1,2}.

¹Departamento de Ingeniería e Investigación Tecnológica Universidad Nacional de la Matanza. Buenos Aires, Argentina.

²Instituto Nacional de Tecnología Industrial. Buenos Aires, Argentina.

agustinaguero.aa@gmail.com; behar.christian@yahoo.com.ar; lanzillottilea@yahoo.com.ar; mati.pity@gmail.com; izaradnik@unlam.edu.ar

Abstract—Se presenta el desarrollo de un sistema de monitoreo de calidad de aire y ruido ambiental, junto el uso de Computación en la Nube (Cloud Computing) como interfaz de visualización de la información. Se comienza con una introducción a la temática de ciudades inteligentes y su importancia para el futuro. A continuación, se presenta la descripción general del sistema: sus componentes, breve una explicación del funcionamiento, la estructura del firmware desarrollado y se finaliza con descripción del uso del Portal de uno de los tantos proveedores de computación en la nube.

Keywords- Ciudades inteligentes, Calidad de aire, Ruido ambiental, Computación en la Nube.

I. INTRODUCCIÓN

Según la Unión Internacional de Telecomunicaciones (ITU por sus siglas en inglés), las ciudades modernas son el hogar del 50% de la población mundial y se espera que para el 2050 el porcentaje sea del 66%. Actualmente las ciudades modernas sean las responsables del 80% de la producción económica mundial y del 70% del consumo de energía mundial y las emisiones de gases de efecto invernadero [1], por lo tanto, es necesario pensar en políticas que permitan la sustentabilidad de las ciudades. Con esto en mente, la ITU creó el grupo de estudio 20 – Internet de las Cosas, ciudades inteligentes y comunidades [2] y la Unión para ciudades sustentable inteligentes (U4SSC por sus siglas en inglés) [3]. Estos grupos tienen como objetivos finales mejorar la calidad de vida de los habitantes, la eficiencia de las operaciones y los servicios urbanos, y su competitividad, mientras se asegura que esta se encuentre de acuerdo a las necesidades de las generaciones presentes y futuras en lo que respecta a aspectos económicos, sociales, medioambientales, así como también culturales.

En este contexto, un sistema de monitoreo de calidad de aire y ruido ambiental es un elemento fundamental. A nivel mundial se encuentran un gran número de desarrollos e implementaciones de sistemas de este tipo, como ejemplos podemos nombrar: la red de monitoreo de la ciudad de Buenos Aires [4], el SIMAT (Sistema de Monitoreo Atmosférico de la Ciudad de México) [5], el proyecto “World Air Quality Index”

[6], el proyecto “Air Quality Egg” [7], entre otros. El presente trabajo es parte de lo realizado en el marco del proyecto “Internet de las cosas y sus aplicaciones en ciudades inteligentes”, por miembros del Laboratorio de Inteligencia Ambiental del Departamento de Ingeniería e Investigación Tecnológica de la Universidad Nacional de la Matanza, quienes vienen trabajando desde hace algunos años en aplicaciones de Internet de las Cosas (IoT) [8][9][10][11].

II. DESARROLLO

A. Descripción general

El sistema de monitoreo de calidad de aire y ruido ambiental está constituido por: un microcontrolador de 32 bits (STM32F411 [14]), un módulo celular de tecnología 3G (UL865 [15]), un sensor de CO2 (MH-Z19 [16]), un sensor de temperatura y humedad (HTU21D [17]), sensor de material particulado (PMS 5003 [18]), sensor de presión (BMP280 [19]) y el sensor de ruido ambiental (basado en micrófono electret y un circuito analógico). La figura N°1 presenta una imagen de este.



Figura N°1. Sistema experimental.

B. Firmware

El microcontrolador es el encargado de efectuar la lectura de los sensores, procesar y analizar los datos obtenidos y transmitirlos, a través del módulo celular, a un servidor en la nube. El procesado, dependiendo del sensor, implica verificar la integridad de los datos, calcular el valor de la magnitud física y corregir las variaciones de los estos debido a la temperatura, entre otras cosas. Mientras que el análisis verifica, en forma local, si los valores obtenidos excedan los límites preestablecidos. Si los límites son excedidos, se transmitirá de forma inmediata un aviso al Portal y se activará una indicación en forma local. Caso contrario los datos serán transmitidos periódicamente.

Como parte del firmware se emplearon: rutinas para el manejo de los puertos serie, tanto para la comunicación con el modulo celular como para con los sensores de CO2 y de material particulado; rutinas para el cálculo de la concentración de CO2 (1) y de verificación de la integridad de los datos de en la comunicación son el sensor (2) ; rutinas para el manejo de la interfaz I2C, tanto para la comunicación con el sensor de temperatura y humedad como para el de presión; rutinas para el cálculo de la temperatura (3) y humedad (4) y la compensación de esta última en función de la temperatura (5)(6); rutina para la lectura del conversor analógico digital asociado al sensor de ruido y rutina para conversión de datos a ASCII para el envío al Portal. Los sensores de material particulado y presión no requieren el calculo de su magnitud ya que la relación entre esta y el valor medido es directa y solo implica una conversión a ASCII.

$$\text{CO2} = \text{HighLevelByte} * 256 + \text{LowLevelByte} \quad (1)$$

$$\text{Checksum} = (\text{invert}(\text{Byte } 1 + \dots + 7)) + 1 \quad (2)$$

$$\text{Temp} = -46,85 + 175,72 * (\text{Dato} / 2^{16}) \quad (3)$$

$$\text{HRactual} = -6 + 125 * (\text{Dato} / 2^{16}) \quad (4)$$

$$\text{HRcomp} = \text{HRacual} + f(t) \quad (5)$$

$$f(t) = -0,15 * (25 - \text{Temp}) \quad (6)$$

C. Portal (Computación en la nube)

Para el desarrollo del Portal se empleó como proveedor de Computación en la Nube a la empresa Telit, la cual ofrece una Plataforma como un Servicio (PaaS - Platform as a Service) para aplicaciones de Internet de las Cosas (IoT) llamada Telit IoT Portal [12]. El Telit IoT Portal posee entre los mecanismos de transporte los protocolos MQTT, COaP y HTTP [13], pero gracias al uso del módulo UL865 no fue necesario la implementación de ninguno de ellos, ya que el módulo permite el envío de datos a través de simples comandos AT. La figura N°2 presenta los gráficos de los sensores de CO2 y de nivel de ruido.

III. CONCLUSIONES

Se logro el desarrollo de un sistema de monitoreo calidad de aire y ruido ambiental. Con el cual, se consiguió interiorizarse en los aspectos mas relevantes de la aplicación y

sus principales dificultades para la implementación de un modelo comercializable. Entre estos podemos nombrar: magnitudes a medir, sus rangos y la exactitud asociada; procedimientos de calibración e instrumentos necesarios; aspectos mecánicos relacionados con la instalación del equipo y su uso en exteriores.

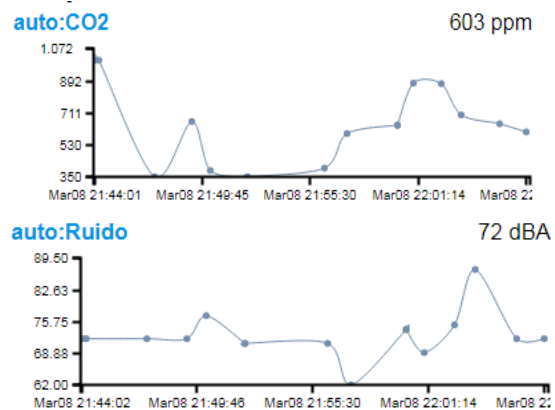


Figura N°2. Gráfica de los sensores de CO2 y nivel de ruido.

En lo que respecta al uso de Computación en la Nube, se logró: comprender distintos aspectos asociados a la Computación en la Nube (tipos de servicios, protocolos relacionados, etc.), el uso efectivo del Portal y la generación de una interfaz de visualización de datos y alarmas.

Como parte de las futuras mejoras del proyecto se plantean: la incorporación de un modulo de posicionamiento global (GNSS) para georeferenciar los datos; el uso de sensores de mayor rango y exactitud, así como la incorporación de sensores de óxidos de nitrógeno; implementación de un anemómetro, y el desarrollo de un tablero de datos (dashboard) en el Portal en la nube.

REFERENCIAS

- [1] Sekhar N. Kondepudi (2016) "Shaping smarter and more sustainable cities. Striving for sustainable development goals". Extraída el 09/03/2018 desde: <https://www.itu.int/es/publications/Pages/publications.aspx?lang=en&media=electronic&parent=T-TUT-SSCIOT-2016-1>.
- [2] ITU (2019). "CE20: Internet de las cosas (IoT) y Ciudades y Comunidades Inteligentes (C+CI)". Extraída el 25/05/2019 desde: <https://www.itu.int/es/ITU-T/studygroups/2013-2016/20/Pages/default.aspx>
- [3] ITU (2019). "United 4 Smart Sustainable Cities". Extraída el 25/05/2019 desde: <https://www.itu.int/en/ITU-T/ssc/united/Pages/default.aspx>
- [4] "Red de Monitoreo de Aire y Ruido de la Ciudad de Buenos Aires" (n. d.). Extraída el 25/05/2019 desde http://www.buenosaires.gob.ar/areas/med_ambiente/apra/calidad_amb/re_d_monitoreo/texto.php?menu_id=32434
- [5] "Sistema de Monitoreo Atmosférico de la Ciudad de México" (n. d.). Extraída el 23/10/2017 desde <http://www.aire.cdmx.gob.mx/default.php>
- [6] "World Air Quality Index" (n. d.). Extraída el 25/05/2019 desde <http://aqicn.org/contact/es/>
- [7] "Air quality egg device" (n. d.). Extraída el 25/05/2019 desde <https://airqualityegg.wickeddevice.com/>

- [8] Brengi; Canziani; Gomez; Gwirc; Lupi; Moltoni; Nassipián; Slawiski; Zaradnik, “Sistema inalámbrico de microestaciones meteorológicas para aplicaciones agropecuarias”. Congreso Argentino de Sistemas Embebidos 2013.
- [9] Canziani; Gomez; Lupi; Nassipián; Slawiski; Turconi; Zaradnik, “Plataforma de conexión de Redes Eléctricas Inteligentes a Internet de las Cosas” en el Congreso Argentino de Sistemas Embebidos 2014.
- [10] Bernis; Turconi; Benacerraf; Dominguez; Lupi; Zaradnik; Rzepa, “Sistema de seguimiento de dosimetría personal”. VII congreso de microelectrónica aplicada 2016.
- [11] Lupi; Zaradnik; Turconi; Dominguez; Rodriguez; Kumvich, “Utilización de tecnología RFID/NFC para el desarrollo de un sistema de llenado selectivo de envases. Industria 4.0. en la 21° Edición del Workshop de Investigadores en Ciencias de la Computación (WICC2108)
- [12] Telit (2019). “Telit IoT Portal: A Cloud-Based Platform as a Service for IoT”. Extraída el 25/05/2019 desde <https://www.telit.com/m2m-iot-products/iot-platforms/telit-iot-portal/>
- [13] Telit (2019). “IoT Portal API Reference Guide”. Extraída el 25/05/2019 desde <http://help.devicewise.com/display/ARG/IoT+Portal+API+Reference+Guide>
- [14] ST (2019). “STM32F411xC STM32F411xE”. Extraída el 25/05/2019 desde <https://www.st.com/resource/en/datasheet/stm32f411ce.pdf>
- [15] Telit (2019). “UL865 Series”. Extraída el 25/05/2019 desde https://y1cj3stn5fbwhv73k0ipk1eg-wpengine.netdna-ssl.com/wp-content/uploads/2017/09/Telit_UL865-Series_Datasheet_2.pdf
- [16] Winsen (2019). “Intelligent Infrared CO2 Module (Model: MH-Z19)”. Extraída el 25/05/2019 desde <https://www.winsen-sensor.com/d/files/PDF/Infrared%20Gas%20Sensor/NDIR%20CO2%20SENSOR/MH-Z19%20CO2%20Ver1.0.pdf>
- [17] TE Connectivity (2019). “HTU21D(F) RH/T SENSOR IC”. Extraída el 25/05/2019 desde https://www.te.com/commerce/DocumentDelivery/DDEController?Action=showdoc&DocId=Data+Sheet%7FHPC199_6%7FA6%7Fpdf%7FEnglish%7FENG_DS_HPC199_6_A6.pdf%7FCAT-HSC0004
- [18] Plantower (2019), “Digital universal particle concentration sensor. PMS5003 series data manual”. Extraída el 25/05/2019 desde http://www.aqmd.gov/docs/default-source/aq-spec/resources-page/plantower-pms5003-manual_v2-3.pdf
- [19] Bosch Sensortec (2019). “BMP280 Digital Pressure Sensor”. Extraída el 25/05/2019 desde <https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST-BMP280-DS001.pdf>

eUCCvm: Plataforma educativa de programación y compilación cruzada universal de sistemas embebidos

Cristian F. Perez-Monte*, Gustavo Mercado*, Carlos Taffernaberry*, Ana Laura Diedrichs*[†], Mario Sebastian Tobar*, Enzo Crespillo*, Raúl Moralejo*, Gabriel Caballero*, Rodrigo Gonzalez*

* GridTICs, Universidad Tecnológica Nacional, Facultad Regional Mendoza, Argentina

Email: {cdlt,gmercado,carlos.taffe,ana.diedrichs,mtobar,enzocrespillo,rmoralejo,gcaballero,rodralez}@gridtics.frm.utn.edu.ar

[†] DHARMA, Universidad Tecnológica Nacional, Facultad Regional Mendoza, Argentina

Email: {ana.diedrichs}@frm.utn.edu.ar

Resumen—Este documento presenta una plataforma educativa para la programación universal de sistemas embebidos para uso educacional en carreras de ingeniería en electrónica, computación y sistemas de información, facilitando la compilación y puesta en funcionamiento del código en diversas arquitecturas.

Index Terms—eUCCvm, sistemas embebidos, EDU-CIAA, arduino, pic

I. INTRODUCCIÓN

En este trabajo presentamos un sistema educativo denominado eUCCvm (*Embedded use Universal Cross Compiler Virtual Machine*) para desarrollo, compilación y programación de sistemas embebidos usados en las cátedras Informática II [1], Técnicas Digitales II [2] y III [3] de la Universidad Tecnológica Nacional, Facultad Regional Mendoza. Está realizado sobre una máquina virtual [4] basada en el sistema operativo Linux Ubuntu 18.04 LTS y puede utilizarse en distintos sistemas operativos de máquinas reales gracias al uso de VirtualBox [5]. La plataforma eUCCvm es pública para uso educativo y puede ser descargada desde [6].

En la sección II se presentan los fundamentos y objetivos para su implementación. En la sección III se exponen las capacidades y limitaciones de la plataforma, las arquitecturas y módulos soportados. En la sección IV se exponen los resultados obtenidos y conclusiones. Finalmente, en la sección V se comentan futuras actualizaciones

II. FUNDAMENTOS Y OBJETIVOS

Las condiciones políticas y económicas de Argentina en los últimos 30 años han impedido el fácil acceso a tecnologías de sistemas embebidos producidas en el exterior más allá de la creciente producción y abaratamiento de los mismos. Proyectos como EDU-CIAA [7] [8] han impulsado el acceso a nivel educacional para el desarrollo de proyectos pero el relativo alto costo de esta placa hace que no sea accesible en el ámbito local. Además, al momento no existían plataformas integradas de fácil acceso que permitan tanto el uso de EDU-CIAA como

de otros sistemas embebidos para su implementación rápida en el ambiente académico.

Por ello, se ha propuesto una plataforma que integre ambientes de programación para la EDU-CIAA, las tradicionales arquitecturas PIC de Microchip [9], otras más recientes como las disponibles para programación desde el IDE de Arduino [10] y otras que puedan aparecer en un futuro.

La creación de la herramienta en las cátedras de Informática II, Técnicas Digitales II y III tiene como objetivo fundamental la formación educativa universitaria del alumnado de carreras de ingeniería en electrónica para resolver problemas en su ámbito laboral y realizarlo con tecnologías apropiadas que permitan su adaptación a un mundo tecnológico cambiante. Además se espera fortalecer su criterio para el análisis y evaluación de desempeño de la solución obtenida.

Los objetivos propuestos para el desarrollo de la plataforma son:

- Disponer de un entorno utilizable en computadoras de bajos recursos computacionales (procesador y cantidad de memoria RAM) tales como las netbooks disponibles del Plan Conectar Igualdad [11] y con conexiones de Internet deficientes, como las disponibles en la provincia de Mendoza [12].
- Fortalecer en el alumno los conocimientos básicos para el uso de GNU/Linux y su interfaz de línea de comandos.
- Fortalecer el conocimiento en el uso de herramientas para el desarrollo, construcción y compilación de programas, y en la grabación de los objetos binarios generados en sistemas embebidos reales.
- Utilizar tecnologías avanzadas con capacidad de adaptación a nuevas tecnologías del futuro.
- Permitir integrar diferentes tecnologías modulares para la resolución de problemas complejos.
- Brindar capacidad de desarrollo y actualización de forma iterativa e incremental.

- Simplificar el acceso del alumno al entorno de desarrollo y maximizar el tiempo de clase dedicado a la programación.

III. CAPACIDADES, LIMITACIONES Y SOPORTE DE ARQUITECTURAS

III-A. Capacidades y Limitaciones

El sistema desarrollado agrega las siguientes capacidades al sistema operativo:

- Escritura de código. Permite la generación de código en diferentes lenguajes de programación, especialmente lenguaje ensamblador [13] y C [14].
- Modificación de códigos existentes. Interpretación del funcionamiento de códigos de ejemplos existentes y su modificación para adaptarlos a la solución de nuevos problemas propuestos.
- Compilación. Generación del código máquina para diferentes arquitecturas de hardware.
- Grabación de los programas compilados en dispositivos embebidos reales.

El sistema está limitado al uso de herramientas de entorno de línea de comandos (CLI) para permitir un entorno liviano y transportable, impidiendo la ejecución de herramientas que requieran de modo gráfico (GUI). Para la escritura y modificación de código se utilizan editores estándares disponibles para linux. Por el momento el sistema está limitado a su uso en programación, compilación y grabación en hardware real, no así a simulación o *debugging*.

III-B. Soporte de Arquitecturas y módulos

Existe una gran cantidad de arquitecturas soportadas en la actualidad, permitiendo su actualización a futuras arquitecturas.

Las arquitecturas de sistemas embebidos soportadas para compilación, tanto en ejecución *bare metal* como para su ejecución en GNU/Linux (para el caso de las más potentes), son:

- Microchip PIC [9].
- AVR (usados en Arduino. [15] NANO, UNO y MEGA).
- Xtensa LX106 (usados por ESP8266 [16] y ESP32).
- ARM Cortex M [17] [18] (utilizados en Blue pill, EDU CIAA o Arduino DUE).
- ARM 11 de 32 bits [19] (utilizados en Raspberry PI 1).
- ARM Cortex A, A7 de 32 bits [20] [21] (utilizados en Raspberry PI 2) y A53 de 64 bits [22] [23] (utilizados en Raspberry PI 3).

Soporta además arquitecturas más avanzadas para su ejecución en Linux, tales como las siguientes arquitecturas de microprocesadores:

- x86 estándar y x86 (amd64/emt64).
- MIPS.
- SPARC.
- ALPHA.
- 68k.
- POWERPC.

Además, agrega ejemplos para la utilización de módulos, que ofrecen funcionalidades a las arquitecturas antes mencionadas, tales como:

- Módulos de GPIO, sensado de variables analógicas y visualización numérica de 7 segmentos.
- Pantallas LCD y OLED de alta resolución.
- Módulos de conectividad: Ethernet, de radio GPRS, WiFi o Bluetooth.
- Módulos de CCDs.
- Almacenamiento con microSD.
- Módulos de Posicionamiento global.

El sistema dispone además de ejemplos sencillos para la implementación de diferentes tecnologías electrónicas de entrada-salida y comunicaciones tales como:

- Uso de Entradas salidas digitales de propósito general.
- Uso de Entradas y Salidas analógicas.
- Protocolo de comunicaciones SPI. [24] e I2C.
- Protocolos de comunicaciones paralelos.

Finalmente están integradas herramientas para la grabación de la compilación en diversas arquitecturas para sistemas embebidos tales como:

- Grabación por puerto USB de AVR Arduino Uno y Mega.
- ARM Blue Pill, Arduino Due y EDU-CIAA.
- ESP8266 y ESP32.
- Microchip Pic con PICKit 2 [25].

Todos los ejemplos están disponibles dentro del sistema, ordenados en carpetas, por arquitectura y módulos usados.

IV. RESULTADOS Y CONCLUSIONES

Las primeras experiencias en el uso de la plataforma fueron por parte de Técnicas Digitales III en el año 2017. Desde finales del año 2018, la plataforma está siendo utilizada también en Técnicas Digitales II con resultados satisfactorios mediante el uso de métricas de facilidad de instalación y uso. Los alumnos disponen de una plataforma lista para usar, permitiéndoles aprovechar de forma óptima el tiempo de clases, realizando las actividades sin requerir procesos de descarga o instalación. Durante el año 2018 se disponía de actividades y trabajos prácticos para el uso de EDU-CIAA mientras que para el año 2019 se disponen también de actividades para Microchip PIC y kits de Arduino.

V. TRABAJOS FUTUROS

Se propone, en el transcurso del año 2019, implementar el uso del sistema también en la cátedra Informática II. A partir del año próximo la plataforma se actualizará agregando prácticas y actividades que permitan utilizar más arquitecturas y uso de más módulos de comunicación, visualización y entrada salida.

AGRADECIMIENTOS

Se agradece el soporte de grupo GridTICs perteneciente a UTN - FRM y a los docentes de las cátedras Técnicas Digitales II, III e Informática II que han facilitado diferentes módulos para el desarrollo de la plataforma. El resto del hardware necesario para el desarrollo de la plataforma ha sido obtenido por financiamiento particular de autores del presente trabajo.

REFERENCIAS

- [1] C. F. Perez-Monte and A. L. Diedrichs, "Programa informática ii 2019," Jul 2019. [Online]. Available: <https://doi.org/10.6084/m9.figshare.8862392>
- [2] C. F. Perez-Monte and A. L. Diedrichs, "Programa técnicas digitales ii 2019," Jul 2019. [Online]. Available: <https://doi.org/10.6084/m9.figshare.8862377>
- [3] C. F. Perez-Monte and A. L. Diedrichs, "Programa técnicas digitales iii 2019," Jul 2019. [Online]. Available: <https://doi.org/10.6084/m9.figshare.8862380>
- [4] R. P. Goldberg, "Survey of virtual machine research," *Computer*, vol. 7, no. 6, pp. 34–45, June 1974.
- [5] P. Li, "Selecting and using virtualization solutions: our experiences with VMware and VirtualBox," *Journal of Computing Sciences in Colleges*, vol. 25, no. 3, pp. 11–17, 2010.
- [6] C. F. Perez-Monte and A. L. Diedrichs, "eUCCvm: Plataforma educativa de programación y compilación cruzada universal de sistemas embebidos," May 2019. [Online]. Available: <https://doi.org/10.6084/m9.figshare.8185169>
- [7] "EDU-CIAA-NXP," Marzo 2017. [Online]. Available: <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo-edu-ciaa:edu-ciaa-nxp>
- [8] M. Dell'Oso, L. C. Lanzarini, and P. Ridolfi, "Prototipo funcional de un sistema de detección de caídas basado en la plataforma CIAA," in *XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016)*, 2016.
- [9] J. Sanchez and M. P. Canton, *Microcontroller programming: the microchip PIC*. CRC press, 2006.
- [10] Y. A. Badamasi, "The working principle of an Arduino," in *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*. IEEE, 2014, pp. 1–4.
- [11] S. L. M. y Ana Marotias y Sheila Amado, "Inclusión digital en la educación pública argentina. el programa conectar igualdad," *Revista Educación y Pedagogía*, vol. 24, no. 62, pp. 205–218, 2013. [Online]. Available: <http://bit.ly/2wm5Ke3>
- [12] "Mendoza, entre las provincias con peor conexión a Internet," Mayo 2019. [Online]. Available: <https://www.elsol.com.ar/mendoza-una-de-las-provincias-con-internet-mas-lento-del-pais>
- [13] R. Hyde, *The art of assembly language*. No Starch Press, 2003.
- [14] D. M. Ritchie, "The development of the C language," *ACM Sigplan Notices*, vol. 28, no. 3, pp. 201–208, 1993.
- [15] A. D'Ausilio, "Arduino: A low-cost multipurpose lab equipment," *Behavior research methods*, vol. 44, no. 2, pp. 305–313, 2012.
- [16] G. Prachchhak, C. Bhatt, and J. Thik, "Data logging and visualization using bolt IoT," in *International Conference on Advanced Computing Networking and Informatics*. Springer, 2019, pp. 155–164.
- [17] E. Blem, J. Menon, and K. Sankaralingam, "A detailed analysis of contemporary ARM and x86 architectures," *UW-Madison Technical Report*, 2013.
- [18] J. Yiu, *The definitive guide to the ARM Cortex-M3*. Newnes, 2009.
- [19] "BCM2835 chipset, Raspberry Pi documentation." [Online]. Available: <http://bit.ly/2woaRkW>
- [20] "BCM2836 chipset, Raspberry Pi documentation." [Online]. Available: <http://bit.ly/2HDmLH9>
- [21] M. E. Soper, "Raspberry Pi system anatomy," in *Expanding Your Raspberry Pi*. Springer, 2017, pp. 1–15.
- [22] "BCM2837 chipset, Raspberry Pi documentation." [Online]. Available: <http://bit.ly/2K76gET>
- [23] J. Ivković and B. Radulović, "The advantages of using Raspberry Pi 3 compared to Raspberry Pi 2. SoC computers for sensor system support," in *International Conference on Applied Internet and Information Technologies*, 2016, pp. 88–94.
- [24] F. Leens, "An introduction to I2C and SPI protocols," *IEEE Instrumentation & Measurement Magazine*, vol. 12, no. 1, pp. 8–13, 2009.
- [25] M. Meriac, "Heart of darkness-exploring the uncharted backwaters of hid iclass (tm) security," in *27th Chaos Communication Congress*, 2010.

Implementación de un Adaptador de Video por Software en Microcontrolador de Doble Núcleo

Santiago Germino
Laboratorio de Sistemas Embebidos
Facultad de Ingeniería - UBA
Buenos Aires, Argentina
sgermino@retro-ciaa.com

Resumen—En este artículo se revisan las consideraciones y técnicas utilizadas para el diseño e implementación por software de un adaptador de video de alta prestación en un kit de desarrollo con microcontrolador de doble núcleo, dedicando un núcleo exclusivamente para tal fin mientras el segundo se utiliza para ejecutar la aplicación.

Palabras clave—computación gráfica, adaptador de video, doble núcleo.

I. INTRODUCCIÓN

Un adaptador de video contiene memoria de video con información de al menos una pantalla completa o cuadro de animación, capacidad de generar una señal de video y un puerto para conectar dicha señal a un monitor compatible [1]. La implementación por software demanda un uso intensivo de recursos. La solución óptima es dedicar un núcleo en un sistema multi-núcleo.

El kit de desarrollo educativo EDU-CIAA-NXP [2] integra un microcontrolador de 204 MHz con un núcleo ARM Cortex-M0, un núcleo ARM Cortex-M4, 136 KiB de SRAM, 1 MiB de Flash compartida [3] y puertos de expansión.

El proyecto RETRO-CIAA es una placa de expansión o «poncho» que suma conectividad y un nuevo firmware en donde se implementó un adaptador de video de alta prestación corriendo en el núcleo Cortex-M0 [4]. Este documento revisa la técnica en base a las decisiones de diseño del proyecto.

II. DISEÑO E IMPLEMENTACIÓN

A continuación se enumeran los elementos que componen esta implementación y se comentan los criterios de diseño.

II-A. Memoria de video

El *framebuffer* es un segmento contiguo en memoria que contiene los pixeles (mínima unidad de una imagen digital) que componen el cuadro de animación a visualizar [1]. Modificar la memoria del cuadro visible produce inestabilidad en la imagen en pantalla. Modificar solo en los períodos inactivos de la señal de video reduce la capacidad de dibujo en un 95%. En consecuencia se decidió utilizar un esquema superior denominado *double-buffering* [1]: dos buffers iguales, uno visible y otro oculto en proceso de actualización. Los buffers intercambian roles toda vez que la señal de video termina de emitir un cuadro de animación.

II-B. Resolución

La resolución de una imagen define su cantidad de pixeles en ancho y alto. En esta implementación cada uno de los dos *framebuffers* tienen una resolución de 256x144 pixeles.

II-C. Formato de pixel

El formato refiere a la cantidad total de bits por pixel y al reparto de esos bits entre los componentes rojo, verde y azul que describen color mediante la intensidad de primarios aditivos. En este caso se decidió utilizar un formato de pixel de 8 bit repartidos en 3 bit para rojo, 3 bit para verde y 2 bit para azul, lo que configura un formato RGB 3:3:2 tal como se observa en la Figura 1. Con 8 bit se obtiene una combinación total de $2^8 = 256$ colores disponibles.

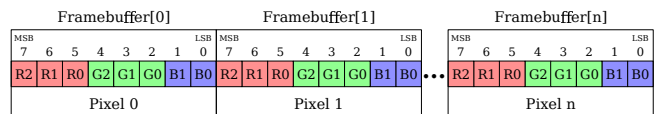


Figura 1: Formato de pixel y pixeles en el framebuffer.

La cantidad de SRAM necesaria para implementar el adaptador de video por software es de $((256 \times 144) \times 2 \times 8) / 8 = 72$ KiB

II-D. Raster scan

El proceso conocido como *raster scan* [1] define el temporizado de los elementos de una señal de video. La señal recorre la pantalla de izquierda a derecha -cada recorrido es una línea de pixeles o *scanline*- y de arriba hacia abajo -donde un barrido completo de todas las *scanlines* completa un cuadro de imagen-. Al final de cada línea se genera una demora que se denomina *horizontal blanking interval*. Del mismo modo, al terminar cada cuadro se genera una demora denominada *vertical blanking interval*. Los *blanking intervals* son áreas inactivas en donde no se dibujan pixeles. En la Figura 2 se observa esta dinámica.

El intervalo de *blanking* horizontal se divide en tres partes: *front porch*, pulso de sincronismo horizontal (HSYNC) y *back porch*. El intervalo de *blanking* vertical se divide en *front porch*, pulso de sincronismo vertical (VSYNC) y *back porch*.

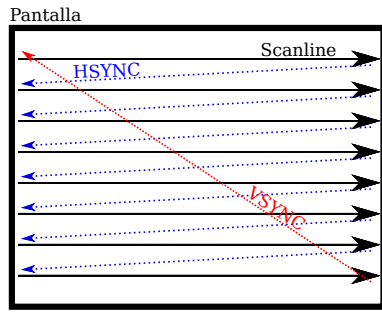


Figura 2: Escaneo raster de una señal de video.

II-E. Señal de video

Se generó un modo de video compatible con el estándar de televisión de pantalla ancha y alta definición debido a que la tecnología CRT ha sido reemplazada por pantallas LCD HDTV. El modo elegido es 720p@60Hz o un área activa de 1280x720 pixeles y frecuencia de actualización de pantalla de 60 Hz o 60 cuadros por segundo [5]. En la Figura 3 se observa un diagrama completo de los elementos que componen la señal de video.

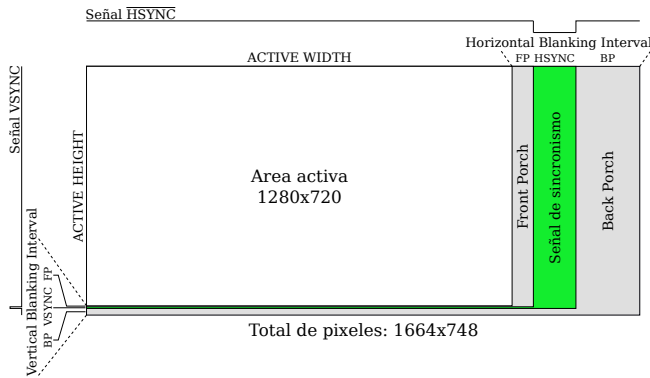


Figura 3: Elementos de la señal de video.

Para generar el temporizado de la señal se utilizó el estándar VESA CVT [6]. Se obtuvo una frecuencia de sincronismo vertical o VSYNC de 59,86 Hz y una frecuencia de sincronismo horizontal o HSYNC de 44,77 kHz. La cantidad de píxeles total de la señal sumando los intervalos de *blanking* es de 1664x748. El reloj de pixel se calcula con los píxeles visibles, los inactivos o no visibles (áreas de *blanking*) y la frecuencia de actualización de la pantalla de este modo: $PCLK = (1664 \times 748) \times 59,86 \text{ Hz} = 74,50 \text{ MHz}$.

Se eligió la interfaz eléctrica VGA por el bajo costo de implementación del hardware necesario. La intensidad individual de los componentes de color rojo, verde y azul se define en un rango de tensión de 0 a 0,7 Vpp a 75 Ω de impedancia. Las señales de sincronismo horizontal y vertical son TTL [7], aunque aquí se generan con 3,3 V. La señal de componentes se implementa mediante un DAC ad-hoc simple como se observa en la Figura 4. Esta es una solución similar a la implementada en los kits de desarrollo FPGA Nexys de Digilent Inc [8].

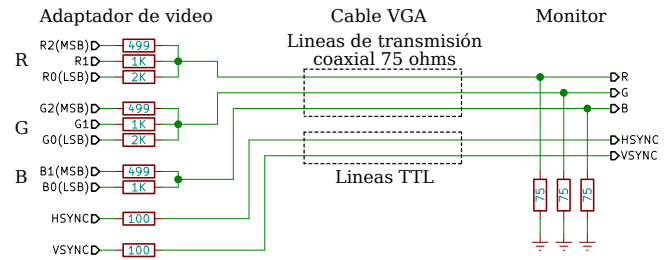


Figura 4: Interfaz eléctrica de la señal de video.

II-F. Puerto de video

El puerto expone las señales de video al exterior del dispositivo. En este caso se utiliza el puerto estándar VGA con conector DB-15HD hembra [7].

II-G. Escalado

Por una decisión de diseño en base a las restricciones de memoria, la resolución del *framebuffer* es menor a la señal de video: contiene exactamente 5 veces menos píxeles tanto en horizontal como en vertical.

El escalado se realiza «al vuelo» mientras se genera la señal de video en la dirección del escaneo raster. Por cada pixel del *framebuffer* se emiten 5 píxeles de la señal de video hasta completar una línea. Luego se emiten 4 copias de la misma línea, completando un total de 5. El resultado a la salida es un *framebuffer* escalado con un factor de 5 en ambas dimensiones.

III. RESULTADOS OBTENIDOS

La señal estándar de alta definición es compatible con televisores LCD HDTV y adaptadores VGA a HDMI. La imagen resultante es similar a la producida por las consolas de videojuegos de los años '80s. La fluidez, nitidez y estabilidad de imagen es sobresaliente para un sistema de este tipo [9]. El núcleo Cortex-M4 queda totalmente libre para ejecutar la aplicación.

IV. CONCLUSIONES

La técnica revisada posibilita: 1. Agregar valor a un proyecto existente implementando una salida de video por software y sumando al hardware solo algunas resistencias y un conector para la señal. 2. Bajar costos utilizando un microcontrolador sin soporte de fábrica para video. 3. Cubrir una necesidad específica como puede ser escalar un *framebuffer* «al vuelo», utilizar cualquier formato de pixel, generar una señal de video no estándar, etc.

Si bien esta técnica no es nueva, su uso junto a un *framebuffer* doble es poco común. Y generar una señal de alta definición escalando un *framebuffer* de baja resolución es un enfoque novedoso. Estas decisiones de diseño fueron requerimientos del proyecto RETRO-CIAA. Otros proyectos pueden adaptar el uso de memoria, resolución, interfaz y modo de video según requerimientos particulares que modifiquen el balance de los recursos disponibles.

REFERENCIAS

- [1] Foley, van Dam, Feiner, Hughes, «Computer Graphics, Principles and practice - Second edition in C», Addison Wesley, 1997.
- [2] Proyecto CIAA, «Computadora Industria Abierta Argentina - EDU-CIAA-NXP», <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>.
- [3] NXP Semiconductors, «LPC43xx/LPC43Sxx ARM Cortex-M4/M0 multi-core microcontroller», version 2.3, July 2017.
- [4] Santiago Germino, «RETRO-CIAA: Consola de videojuegos basada en EDU-CIAA-NXP», <http://laboratorios.fi.uba.ar/lse/tesis/LSE-FIUBA-Trabajo-Final-CESE-Santiago-Germino-2018.pdf>
- [5] The Advanced Television Systems Committee, Inc., «ATSC Digital Television Standard», A/53 Parts 1 - 6, 2007.
- [6] Video Electronics Standards Association, «VESA Coordinated Video Timings (CVT) Standard», version 1.2, February 2013.
- [7] Robert L. Myers, «Display Interfaces - Fundamentals and standards», Wiley-SID, 2002.
- [8] Digilent Inc., «Nexys A7: FPGA Trainer Board Recommended for ECE Curriculum» <https://store.digilentinc.com/nexys-a7-fpga-trainer-board-recommended-for-ece-curriculum>.
- [9] Santiago Germino, «RETRO-CIAA: Consola de videojuegos basada en EDU-CIAA-NXP - Demostración Defensa TF CESE», https://youtu.be/tJM6_TdOuKQ.

Diseño e implementación de una arquitectura IoT en un sistema contador de pasajeros

Yoel Yamil López, Pablo Gomez, Ariel Lutenberg
CONICET - GICSAFe
Laboratorio Sistemas Embebidos
Facultad de Ingeniería - UBA
Buenos Aires - Argentina
lse@fi.uba.ar

Resumen— En este artículo se presentan avances en el diseño de un sistema embebido para la contabilización de pasajeros y la implementación de una arquitectura modular desarrollada bajo el paradigma de Internet de las Cosas (IoT). Este proyecto es la continuación de un desarrollo previo realizado para Trenes Argentinos, el principal operador ferroviario de la República Argentina. Se muestran resultados experimentales y se discuten los detalles de la implementación.

Keywords— *beacons WiFi; OTA; ESP8266; probe request; Node Red; MQTT; JSON; Machine Learning.*

I. MOTIVACIÓN Y CONTEXTO

El sistema de transporte metropolitano de pasajeros en Buenos Aires genera todos los días una gran cantidad de datos a través del Sistema Único de Boleto Electrónico (SUBE). Esta información no cubre lo que sucede con el pasajero una vez que ingresa a la estación. El sistema contador de pasajeros se encarga de suministrar esos datos. El proyecto es desarrollado por CONICET-GICSAFe [2] para la empresa Trenes Argentinos [3].

El sistema que inicialmente contaba con módulos contadores de pasajeros redundados en una única ubicación de un coche [1], actualmente maneja seis módulos contadores a lo largo de seis coches de una formación. Esta situación condujo al diseño de una arquitectura que permite administrar los datos adquiridos de una forma más flexible y eficiente en términos de capacidad de procesamiento y de almacenamiento de la información.

II. DESCRIPCIÓN DE LOS PROBLEMAS A RESOLVER

La necesidad de gestionar más módulos y, a su vez, procesar y almacenar de forma más eficiente los datos adquiridos, disparó la decisión de rediseñar e implementar una arquitectura modular desarrollada bajo el paradigma de Internet de las Cosas (IoT) [4]. Dentro de las principales mejoras y beneficios de esta nueva arquitectura se pueden mencionar: i) la utilización de protocolo OTA [5] para actualizar el firmware en forma remota, ii) la incorporación de procesos de tratamiento y recolección de datos que facilitan el análisis de los mismos y iii) un modelo de datos para poder almacenar y consultar la información procesada.

En la sección III se detallan cada uno de los componentes que forman parte de la arquitectura propuesta, en la sección IV las pruebas realizadas junto con los resultados obtenidos, en la sección V las conclusiones a las que se han arribado y en la sección VI los avances que tendrá el próximo *release*.

III. ARQUITECTURA PROPUESTA

En el marco del presente trabajo, la arquitectura del sistema está estratificada en capas tal como se ilustra en la Fig.1. A continuación se describen cada una de estas capas.

A. Capa de Procesamiento Cloud

La capa de procesamiento cloud se encuentra constituida por módulos ESP8266-01 [6] que se encargan de detectar los *probe request* o *beacons WiFi* (en adelante balizas) que emiten los dispositivos con conectividad WiFi, por ejemplo un teléfono celular.

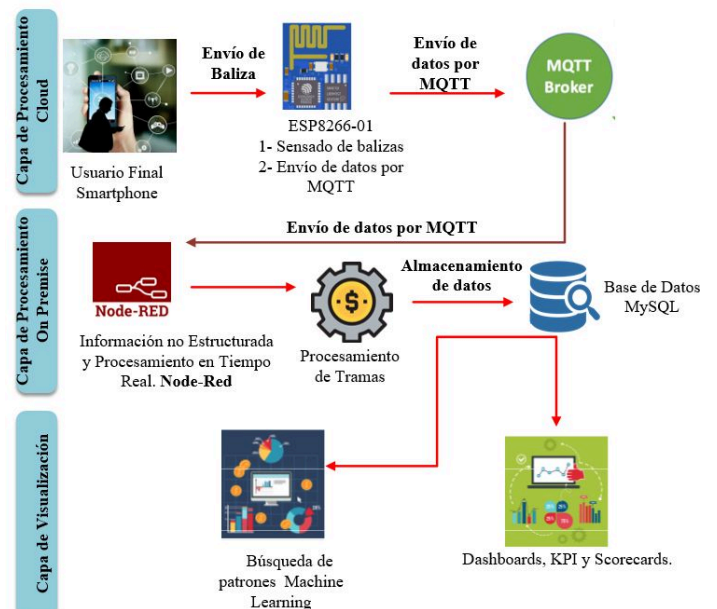


Fig.1 Arquitectura modular del sistema contador de pasajeros.

El firmware implementado utiliza estas balizas para generar y actualizar un listado con la siguiente información:

- MAC del dispositivo que genera la baliza,
- potencia con la que se recibe,
- tiempo transcurrido desde la última baliza recibida del dispositivo, y
- cantidad de balizas recibidas del dispositivo.

A partir de estos datos el firmware depura y procesa este listado. Se asume que cada MAC del listado depurado corresponde a una persona en el rango de alcance del módulo [1]. Es importante comentar que en la solución presentada no se está utilizando ningún sensor adicional para contar pasajeros.

En el trabajo presentado en [1] la información recopilada de las balizas era procesada localmente por cada módulo. En esta nueva versión el procesamiento se efectúa globalmente en el servidor a través de un listado actualizado de las balizas que envía cada uno de los módulos en formato JSON [8] utilizando el protocolo MQTT [7].

B. Capa de Procesamiento On Premise

La capa de procesamiento on premise (Fig.1. A) se encarga de realizar el procesamiento, recolección y almacenamiento de la información enviada por los módulos en la base de datos del sistema. Está constituida por una aplicación desarrollada en Node Red y un motor de base de datos relacional MySQL.

El Node Red [9] es una herramienta para comunicar hardware y servicios de una forma rápida y sencilla, y simplifica la programación del lado del servidor.

MySQL [10] es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL[11]).

Las funcionalidades principales del Node Red son:

- Recoger y procesar las tramas de datos en formato JSON de cada uno de los contadores instalados en la formación ferroviaria.
- Generar un listado global de balizas con los datos enviados por cada contador verificando los solapamientos para evitar contar al mismo dispositivo más de una vez.
- Generar un listado con tiempo de ingreso y egreso de cada uno de los dispositivos. Comparando esta información con la ubicación de la formación es posible determinar la estación de ingreso y egreso de cada pasajero.

C. Capa de Visualización

En la capa de visualización se desarrollan tableros para explotar la información alojada en la base de datos a partir de la generación de distintos indicadores en función de lo que se desee medir, por ejemplo, la cantidad de celulares detectados y cantidad de pasajeros.

IV. RESULTADOS OBTENIDOS

Se instaló el sistema en una formación de la línea Tigre - Retiro con el objetivo de analizar su desempeño.

En la Fig. 2 se presenta una de las mediciones obtenidas a lo largo de un viaje entre cabeceras partiendo de Tigre a las 5:16 a.m. y llegando a Retiro a las 6:21 a.m. En la gráfica se aprecia la cantidad de MACs detectadas y los pasajeros contabilizados luego de la depuración y procesamiento de la información recibida por los módulos. Por ejemplo, una de las condiciones utilizadas para considerar que una MAC es un pasajero se basa en contabilizarla por lo menos 3 veces durante un determinado lapso de tiempo.

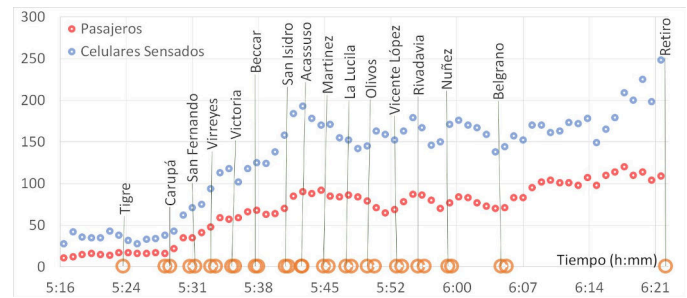


Fig. 2 MACs detectadas y pasajeros contabilizados a lo largo de un viaje entre cabeceras.

Es interesante notar que a partir de la estación Belgrano los pasajeros contabilizados se mantienen en aproximadamente 100. Esto ocurre a pesar del incremento en la cantidad de MACs detectadas. Esto es una muestra de que el sistema logra discriminar celulares detectados y pasajeros ya que en este trayecto no hay ingreso ni egreso de pasajeros.

V. CONCLUSIONES

Los resultados obtenidos permitieron confirmar que los cambios introducidos en el diseño del sistema fueron apropiados. La arquitectura implementada permitió mejorar el análisis y facilitó la disponibilización y la explotación de la información procedente de los contadores. A su vez, la actualización remota del firmware permitió realizar diferentes ajustes a la lógica para obtener un valor cada vez más exacto en la estimación de la cantidad de pasajeros.

VI. PRÓXIMOS PASOS

Actualmente, se está desarrollando un modelo estadístico que tiene como objetivo mejorar el proceso de estimación de la cantidad de pasajeros utilizando técnicas de Machine Learning[12]. Además, se está trabajando en la inclusión de FreeRTOS[13] sobre el firmware y en la incorporación de hardware (Raspberry Pi 3 Model B+[14]) que facilite el procesamiento de la información de forma local (en la misma formación).

REFERENCIAS

- [1] 'Sistema contador de pasajeros en tiempo real mediante el procesamiento de beacons WiFi', M. Menendez, P. Gomez, A. Lutenberg, Libro de Trabajos del CASE2018, Congreso Argentino de Sistemas Embebidos, Córdoba, Argentina. (2018).
- [2] CONICET-GICSaFe (2019). [Online]. Disponible: <http://www.conicet-gicsafe.com.ar>
- [3] Trenes Argentinos (2019). [Online]. Disponible: <https://www.argentina.gob.ar/transporte/trenes-argentinos>
- [4] Paradigma IoT (2019). [Online]. Disponible: <https://geinfor.com/business/que-es-el-internet-de-las-cosas-iot/>
- [5] OTA (2019). [Online]. Disponible: https://en.wikipedia.org/wiki/Over-the-air_programming
- [6] Espressif. (2018). Catálogo de productos de la familia ESP8266. [Online]. Disponible: <https://www.espressif.com/en/products/hardware/modules>
- [7] MQTT (2019). [Online]. Disponible: <https://github.com/mqtt/mqtt.github.io/wiki>
- [8] JSON (2019). [Online]. Disponible: <https://www.json.org/>
- [9] Node Red (2019). [Online]. Disponible: <https://nodered.org/>
- [10] MySQL (2019). [Online]. Disponible: <https://www.mysql.com/>
- [11] SQL (2019). [Online]. Disponible: <https://es.wikipedia.org/wiki/SQL>
- [12] Machine Learning (2019). [Online]. Disponible: https://es.wikipedia.org/wiki/Aprendizaje_autom%C3%A1tico
- [13] FreeRTOS (2019). [Online]. Disponible: <https://www.freertos.org/>
- [14] Raspberry Pi (2019). [Online]. Disponible: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

Implementación de un sistema de estimación de flujo vehicular y velocidad basado en procesamiento de video.

S. E. Burgos
U.T.N. Facultad Regional Paraná
Paraná, Argentina
sergioburgos@frp.utn.edu.ar

S. J. Comas
U.T.N. Facultad Regional Paraná
Paraná, Argentina
sergiocomas@alu.frp.utn.edu.ar

F. M. Taborda
U.T.N. Facultad Regional Paraná
Paraná, Argentina
felixtaborda@alu.frp.utn.edu.ar

Resumen—En este trabajo se presenta la implementación de un sistema de procesamiento de video en tiempo real destinado a la estimación de flujo vehicular y la velocidad de los móviles. El trabajo se realizó utilizando flujos de video provenientes de cámaras de monitoreo no destinadas para este fin y la biblioteca de procesamiento de imágenes OpenCV.

Se presentan diferentes métodos existentes para el procesamiento deseado y el resultado obtenido en la implementación sobre un Raspberry Pi 3B.

Palabras clave—sistemas embebidos, opencv, optical flow, raspberry pi

I. INTRODUCCIÓN

Cada vez es más común encontrarnos con equipos electrónicos que realizan alguna forma de procesamiento de imágenes y video. Ya sean televisores inteligentes controlados por gestos o sistemas de vigilancia que detectan y/o identifican personas, en todos estos casos aparece la aplicación de diferentes técnicas de visión asistida por ordenador.

Este precedente y el hecho que cada vez son más las cámaras de video vigilancia instaladas en nuestras ciudades, han dado origen a este trabajo como resultado de una tesis de grado de Ingeniería Electrónica. El objetivo ha sido evaluar diferentes alternativas de procesamiento de video en tiempo real a fin de lograr un prototipo de sistema embebido capaz de determinar el flujo vehicular (cantidades de móviles que pasan por un lugar determinado) y la velocidad de los mismos.

A fin de evaluar el resultado logrado, se utilizó el prototipo para procesar fragmentos de videos reales, obtenidos por una cámara de vigilancia en el interior de un túnel donde circulan vehículos en ambas direcciones.

II. METODOLOGÍA

El desarrollo se inició evaluando diferentes métodos y algoritmos de procesamiento utilizando computadoras personales para realizar el procesamiento. Se optó por esta alternativa ya que algunos de los algoritmos evaluados requerían realizar tareas de entrenamiento laboriosas. Una vez identificados los algoritmos más convenientes, se procedería a adecuarlos para ejecutarse en un sistema embebido.

En el algoritmo de procesamiento del flujo de video utilizado puede observarse una clara división. En primer lugar

se tiene una etapa de adquisición de cuadros de video e identificación de las zonas de interés, que se corresponden con móviles dentro de la escena. En una segunda instancia, se realiza la detección de las entidades de interés y sus características a fin de clasificarlas, contarlas en función de su tipo y determinar la velocidad con la que se traslada (Figura 1).

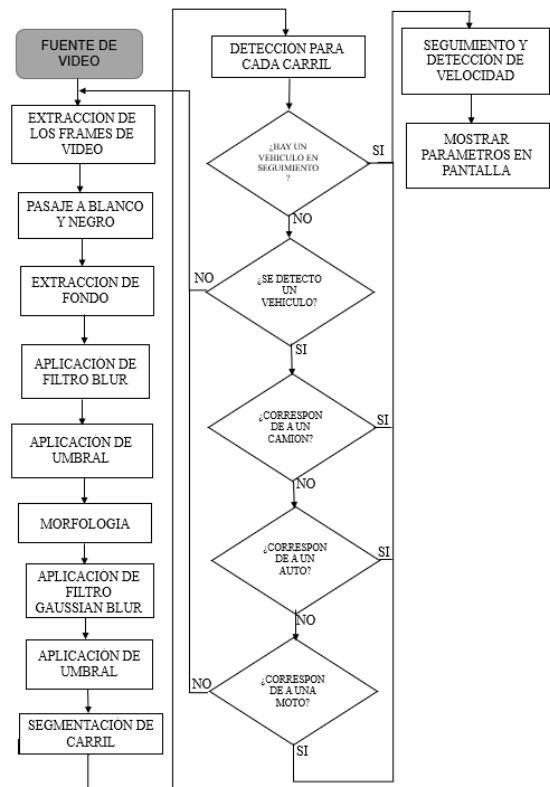


Figura 1. Diagrama en bloques del funcionamiento del código.

II-A. Identificación de zonas de interés

El problema inicial consiste en determinar, en un flujo de video continuo, un objeto móvil en una secuencia de cuadros de video. Considerando que cada imagen es tratada como una matriz numérica, los objetos en movimiento pueden detectarse como la diferencia entre cuadros consecutivos [1]. Si bien esta

técnica es teóricamente válida, se encontraron imprecisiones en la implementación debido a la presencia de ruido de diferente tipo. Para resolver esto se aplicaron diferentes filtros, entre ellos blur y detección de umbral a fin de lograr aislar el objeto de interés [2], [3].

II-B. Detección del móvil de interés y extracción de características

El problema en esta etapa es, dada una imagen con una superficie de interés contenida, determinar sus características. Es significativo identificar de que tipo de móvil se trata y la posición de su centro de masa dentro de la imagen completa. Los métodos considerados para este procesamiento fueron:

- Detector de características Haar-Like: esta técnica permite la detección de objetos a partir de clasificadores lineales en cascada. Según se avanza en el procesamiento, cada etapa descarta la posibilidad de encontrar un objeto de interés. Si se logra completar el procesamiento se tiene una entidad identificada. Este tipo de clasificador debe ser entrenado con los objetos que se desean identificar. [4]
- Redes neuronales: en esta dirección existen diferentes implementaciones. En nuestro caso utilizamos YOLO (*You Only Look Once*). Esta alternativa utiliza redes neuronales entrenadas para la detección de objetos. [5]
- Detector de contornos: esta alternativa, a diferencia de las anteriores, no identifica un objeto particular, sino que brinda detalles geométricos de un fragmento contenido en una imagen. A partir de la superficie, se podría inferir el tipo de móvil y realizar una clasificación posterior. [6]

Los dos primeros métodos mostraron gran precisión en la detección de objetos, sin embargo, su costo computacional los hicieron inviables en la implementación de un sistema embebido. La implementación se realizó utilizando detectores de contornos, con la información provista por ellos, se estimó la velocidad de los móviles.

La estimación de velocidad se realizó a partir de dos posiciones en los cuadros de video que se encontraban a una distancia conocida, luego midiendo el tiempo que tarda en pasar el centro de gravedad de un mismo móvil por ambas marcas el resultado es directo [7].

A fin de mejorar la precisión, luego de la separación de fondo y primer plano, se segmentó la imagen en dos partes utilizando máscaras. De este modo se procesó por separado el tráfico de cada carril.

III. RESULTADOS EXPERIMENTALES

A fin de cuantificar la exactitud en los resultados del procesamiento, se procesaron dos videos diferentes con las técnicas antes descriptas utilizando un Raspberry Pi 3, raspbian como sistema operativo e implementando con opencv 3.4 y gcc los algoritmos mencionados. Los videos utilizados fueron:

- Video 1: obtenido con una cámara de monitoreo de uso general. Dispuesta en el interior de un túnel, con una posición pre establecida, captores analógicos y digitalizado (resolución 854 x 480, duración 10 minutos).

- Video 2: obtenido a través de un celular con cámara digital, dispuesta de modo conveniente a 45 grados del piso, sobre un puente peatonal, en una avenida de doble mano, captando el flujo vehicular en ambos sentidos (resolución 640 x 480, duración 3 minutos y medio).

El resultado del procesamiento se presenta en la Tabla I, allí puede observarse una menor tasa de errores cuando la posición de la cámara se ubica de modo conveniente. No obstante, cabe mencionar que el resultado obtenido fue dependiente en gran medida de la hora a la que se realizó la grabación del Video 2, ya que según la posición del sol nos resultaba imposible realizar el procesamiento por no poder captar las imágenes o las sombras que se generaban resultando en una detección inexacta de los móviles, lo que es lógico por utilizar un detector de contornos para realizar el procesamiento. A raíz de esto, se experimentó posicionando la cámara a 90 grados, lográndose disminuir el error. En el caso del túnel, dado el ángulo de la cámara, era necesario aplicar mayor cantidad de filtros, lo cual aumentaba la carga de procesamiento, haciéndolo inviable en el sistema embebido.

Tabla I
RESULTADOS EXPERIMENTALES DE LA DETECCIÓN DE MÓVILES

Origen de datos	Vehículos	Resultado	Error (%)
Video 1	72	60	16,66
Video 2	58	52	10,30

En la Tabla II, se puede observar el rendimiento promedio de los distintos algoritmos de detección, usando la Raspberry Pi3. Cabe destacar que los videos fueron filmados a 25 FPS, los cuales eran nuestra meta a lograr en la velocidad de procesamiento. La diferencia de FPS del video 1 con respecto al video 2 viene dado por el aumento de filtrado, para mejorar las áreas de interés.

En cuanto a la estimación de velocidad, se tomo como referencia la velocidad real de un automóvil, que circulaba en un determinado momento de la captura. Podemos observar en la Tabla III, una buena aproximación de la misma, no obstante surge el problema que a menor velocidad los vehículos suelen estar muy próximos entre si, provocando que el cálculo de la velocidad sea completamente erróneo.

Se utilizó solamente el video 2 como referencia para contrastar la velocidad, dado que en el video 1 resulto complicado medir una distancia física dentro del túnel.

Tabla II
FPS OBTENIDOS SEGÚN EL ALGORITMO IMPLEMENTADO

Origen de datos	YOLO	Haar-Like	Detector de Contornos
Video 1	1	7	20
Video 2	1	8	24

Tabla III
ESTIMACIÓN DE VELOCIDAD PROMEDIO

Origen de datos	Velocidad Real	Velocidad Estimada
Video 2	50	56

Utilizando algoritmos de mayor carga computacional, fue posible detectar tanto el flujo de vehículos como su velocidad

procesando ambos videos con una computadora personal (4Gb de memoria, procesador AMD A6, sistema operativo Ubuntu 18.04). Al intentar realizar el mismo procesamiento utilizando un Raspberry PI, este era válido pero el tiempo precisado en el filtrado de los frames del video 1 eran superior al requerido. No obstante en el caso del ídeo 2, al requerir menor filtrado, se llegó a valores aceptables de tiempo de procesado.

En la figura 2 se presenta el prototipo logrado, constituido por una pantalla táctil y un raspberry pi que accedía al video a procesar desde un dispositivo de almacenamiento USB. Los resultados del procesamiento son transmitidos por un terminal serie, que era objetivo original, pero a fin mejorar la interfaz se presenta gráficamente el video procesado junto a etiquetas que indican el objeto detectado, su velocidad y la cantidad de móviles detectados.

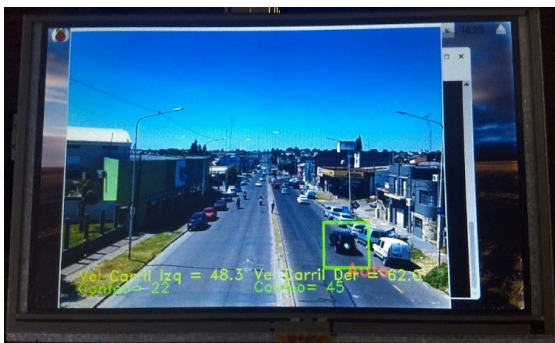


Figura 2. Detección de vehículo, clasificación y estimación de velocidad en Raspberry Pi3.

IV. CONCLUSIONES

Para realizar las pruebas se han capturado videos desde diferentes perspectivas y en distintas horas del día. Esto

ha provocando diversidad de resultados, de los cuales en algunos casos se ha conseguido contajes muy aproximados a la realidad, realizando estimaciones y clasificaciones bastantes acertadas. De otros, la dificultad del video ha hecho muy compleja esta tarea y se ha quedado lejos del propósito inicial.

Como mejora se puede disminuir la resolución de los frames, con el fin de aumentar la velocidad de procesamiento. Además se puede optar por bajar la tasa de cuadros del video y eliminar la visualización de los datos procesados, debido a que la complejidad del mismo puede provocar una disminución de la información a procesar.

Por otro lado se podrían llegar a implementar otras plataformas que sean mas acorde al procesamiento en cuestión, como por ejemplo el caso del Intel Neuronal Compute Stick.

Como conclusión final se puede decir que se llegó al objetivo deseado logrando detectar, clasificar y estimar la velocidad de los vehículos en la Raspberry Pi 3, teniendo en cuenta que dicho hardware se encontró al límite de sus capacidades de computo.

REFERENCIAS

- [1] A. Sobral, "Bgslibrary: An opencv c++ background subtraction library," in *IX Workshop de Visao Computacional*, vol. 2, no. 6, 2013, p. 7.
- [2] A. BENAVENT PLA, "Vehículo autónomo con visión artificial utilizando opencv," Ph.D. dissertation, 2018.
- [3] S.-C. S. Cheung and C. Kamath, "Robust background subtraction with foreground validation for urban traffic video," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 14, p. 726261, 2005.
- [4] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Proceedings. International Conference on Image Processing*, vol. 1. IEEE, 2002, pp. I-I.
- [5] Ó. Fernández Angulo *et al.*, "Sistema de detección de señales y vehículos mediante redes neuronales convolucionales," 2018.
- [6] A. Kaehler and G. Bradski, *Learning OpenCV 3: computer vision in C++ with the OpenCV library*. Reilly Media, Inc., 2016.
- [7] E. Patel and D. Shukla, "Comparison of optical flow algorithms for speed determination of moving objects," *International Journal of Computer Applications*, vol. 63, no. 5, 2013.

Sistema inalámbrico de bajo costo para la adquisición y visualización de datos para aerogeneradores basado en el chip ESP8266

Maximiliano Ozono
Instituto de Industria
Universidad Nacional de General
Sarmiento
Buenos Aires, Argentina
mozono@campus.ungs.edu.ar

Juan Carlos Agotegaray
Instituto de Industria
Universidad Nacional de General
Sarmiento
Buenos Aires, Argentina
jagotega@campus.ungs.edu.ar

Andrea Pinzón
Instituto de Industria
Universidad Nacional de General
Sarmiento
Buenos Aires, Argentina
apinzon@campus.ungs.edu.ar

Abstract—En este trabajo se presenta un sistema inalámbrico de adquisición y visualización de datos autónomo, aplicado a la medición de variables asociadas a la eficiencia de un aerogenerador de baja potencia. El desarrollo de este sistema permite la medición y visualización de los datos en tiempo real a través de un navegador web en cualquier dispositivo que soporte el lenguaje JavaScript, a bajo costo y prescindiendo de conectividad a Internet, a partir del estándar WiFi 802.11 que posee el chip ESP8266, junto a un stack TCP/IP y un microcontrolador programable en su interior. Asimismo, es posible recolectar la información generada y exportarla en un formato adecuado para su tratamiento en bases de datos. Esto permite obtener lecturas rápidas de las condiciones de entorno y funcionamiento del aerogenerador sin necesidad de operar directamente sobre el mismo, lo cual es particularmente útil en aquellos casos en los que se dificulte el acceso y la interacción física suponga un riesgo para el usuario.

Keywords—Aerogeneradores, Visualización, ESP8266, Wireless

I. INTRODUCCIÓN

En los últimos años, el surgimiento de lo que se conoce como Industria 4.0 [1] ha permitido la expansión y difusión de diversos sistemas tecnológicos. En particular, con el desarrollo de Internet y de la computación en nube, y también a partir de las optimizaciones pertinentes en materia de infraestructura, ahora es posible interconectar diversos sistemas ciber físicos [2] en el marco de un sistema integrado basado en lo que se conoce como Internet de las Cosas (IoT). Esto ha permitido la obtención de grandes volúmenes de datos a partir de dispositivos de sensado y medición de bajo costo, almacenados en servidores adhoc en la nube. Este fenómeno, inicialmente concebido en el marco de la industria manufacturera, también ha permitido la obtención de información con carácter sistémico en otros sectores tales como Energía y Servicios. En paralelo, la propagación casi masiva de hardware programable para prototipar soluciones de monitoreo y control de bajo costo orientado a conectar objetos a Internet, traccionado por una fuerte demanda de desarrolladores y hobbyistas concentrados en comunidades como la del proyecto Arduino, han contribuido a la emergencia de una oferta de sistemas tecnológicos fuera de las firmas que, tradicionalmente, han estado a la vanguardia de estos desarrollos. Ejemplo de este fenómeno ha sido el desarrollo del chip ESP8266, confeccionado inicialmente por la firma china Espressif para

productos de uso doméstico para dotarlos de conectividad WiFi, pero rápidamente implementados por diversos desarrolladores para el prototipado -y en algunos casos implementación- de soluciones IoT [3]. Al momento de visualizar y recolectar los datos a distancia, prácticamente todas las soluciones actuales suponen una interacción obligatoria entre el dispositivo en cuestión y un servidor alojado en la nube [4]. Pero en aquellos casos en los que, por cuestiones geográficas, es difícil o costoso lograr la conectividad a Internet, la mayoría de los sistemas de visualización y sensado de datos suponen una interacción física directa con el dispositivo (a menudo el usuario debe retirar la información almacenada en algún tipo de memoria alojada en el mismo, sucediendo algo similar con la visualización de los datos). Esto supone una tarea laboriosa en dispositivos alojados en lugares peligrosos o de difícil acceso, además de que la manipulación directa sobre el dispositivo puede implicar riesgos tanto para el usuario (que puede estar calificado para procesar los datos pero no para operar sobre el dispositivo) como para el dispositivo en sí.

Por otra parte, si sólo se desean registrar y visualizar los datos en tiempo real -sin necesidad de operar físicamente sobre el dispositivo-, entonces el uso de microcontroladores de bajo costo dotados de conectividad WiFi como el ESP8266 pueden considerarse teniendo en cuenta ciertas limitaciones, pero asegurando la portabilidad [5].

II. PRINCIPIO DE FUNCIONAMIENTO

En este caso particular, se ideó una solución basada en el chip ESP8266 orientada a la obtención y visualización de las RPM de un aerogenerador de baja potencia, junto a otros datos del clima, prescindiendo de la conectividad a Internet. La información obtenida es visualizable a partir de cualquier dispositivo móvil con comunicación WiFi, como un teléfono inteligente o una tablet. El desarrollo del proyecto se puede dividir en dos etapas: La de sensado y la de visualización de datos, que se describen a continuación.

A. Estación de sensado con punto de acceso WiFi.

El elemento central del proyecto lo constituye el chip con stack TCP/IP y conectividad WiFi ESP8266, en su versión NodeMCU, programable utilizando un puerto USB. A su vez, se utilizaron los siguientes sensores: sensor de temperatura y humedad relativa DHT11, sensor de presión

barométrica y temperatura BMP180 y sensor inductivo LJ12A3. Adicionalmente se agregó un reloj RTC DS1306 para registrar los datos con fecha y hora. El dispositivo fue configurado para generar un punto de acceso WiFi a partir del cual es posible acceder al servidor local montado en el sistema de archivos del ESP8266. Se diseñó una placa específica para el prototipo, el cual es alimentado por las baterías del aerogenerador, a través de un regulador de tensión. El dispositivo registra las RPM, temperatura, humedad relativa y presión barométrica, a la vez que realiza un cálculo estimativo de la altitud del dispositivo. Estas mediciones se realizan cada 10 minutos, promediando los valores obtenidos. Cada registro se guarda en un archivo de extensión CSV en el sector SPIFFS correspondiente al sistema de archivos. El tamaño máximo de archivo estimado para los datos obtenidos es de aproximadamente 2,5MB. Para una segunda etapa se espera poder sumar como variables de medición la velocidad del viento y la corriente en DC que produce el generador eléctrico, para calcular de esta forma la potencia extraída del viento y la potencia eléctrica generada.

B. Servidor web

Una parte importante del desarrollo del proyecto se sitúa en torno al servidor web local al que acceden los usuarios que se conectan a la red WiFi del dispositivo. Ante una nueva conexión, ingresando la IP local 192.168.4.1 en el navegador web que posea el dispositivo donde se visualizarán los datos, el servidor devuelve una página web y establece una conexión basada en el protocolo WebSockets entre ambos dispositivos. Esto permite una comunicación de alta velocidad, comparada a otros protocolos como TCP/IP. Al mismo tiempo se utilizó la librería Charts.js para el renderizado de gráficos a partir de los valores obtenidos, así como la opción de exportar los datos generados en CSV hacia el dispositivo que mide. Como se mencionó previamente, todo el servidor, incluyendo las librerías para renderizado, se alojan en el sistema de archivos del ESP8266.

III. AEROGENERADOR

Los aerogeneradores son máquinas que convierten la energía cinética contenida en el viento en energía eléctrica. En alta potencia se implementan en grandes parques eólicos para el abastecimiento del sistema nacional eléctrico. Mientras que en baja potencia existen dos aplicaciones principales: sistemas aislados, donde no existe red eléctrica y es muy costoso su instalación, por ejemplo zonas rurales; y sistemas *on grid*, es decir, equipos que generan energía eléctrica con la posibilidad de autoabastecerse o inyectarla a la red eléctrica, normalmente en zonas urbanas.

Para la caracterización de los aerogeneradores a través de curvas de potencia es necesario realizar mediciones de parámetros eléctricos, mecánicos y meteorológicos, ya que a partir de estos se podrá determinar los rendimientos de la turbina eólica y el generador eléctrico.

A. Turbina eólica

La turbina eólica implementada en el proyecto es de eje vertical, es decir, su eje de rotación es perpendicular al suelo. Posee tres palas fabricadas a partir de un perfil aerodinámico para transmitir torque al generador a partir de la energía cinética del viento.

B. Generador eléctrico

El generador eléctrico es sincrónico de flujo axial con imanes permanentes acoplado directamente a la turbina eólica sin el uso de sistema multiplicador. Está configurado como trifásico y la frecuencia de generación depende directamente de la velocidad de rotación, con lo cual, es variable y es necesaria rectificar a corriente continua para poder almacenar la energía eléctrica generada en baterías.

C. Curva de potencia

La Norma IEC 61400-12 Anexo H [6] trata la metodología que debe implementarse para la obtención de la curva de potencia de aerogeneradores de baja potencia con conexión a banco de baterías. Define al aerogenerador como una caja negra del cual se analiza la energía entrante (la energía potencial del viento), y la energía eléctrica en DC producida por el generador eléctrico. A partir de estas mediciones, se realiza los cálculos pertinentes para obtener la curva de potencia en función de la velocidad del viento, y el rendimiento del sistema.

D. Ensayos

En las pruebas realizadas, el dispositivo se colocó en una caja estanca. Se diseñó una pieza a medida con el fin de sostener el sensor inductivo que mide las RPM. Se midió la distancia efectiva desde la cual se obtiene la señal del punto de acceso generado por el dispositivo. La distancia máxima obtenida fue de 80 metros utilizando un teléfono móvil con vista directa. El alcance de la red puede ampliarse utilizando una antena externa, modificación prevista en las próximas versiones. Un esquema del sistema se muestra en la figura 1.

Se verificó que el dispositivo funciona y es posible obtener y almacenar los datos para realizar el estudio y caracterización del aerogenerador.

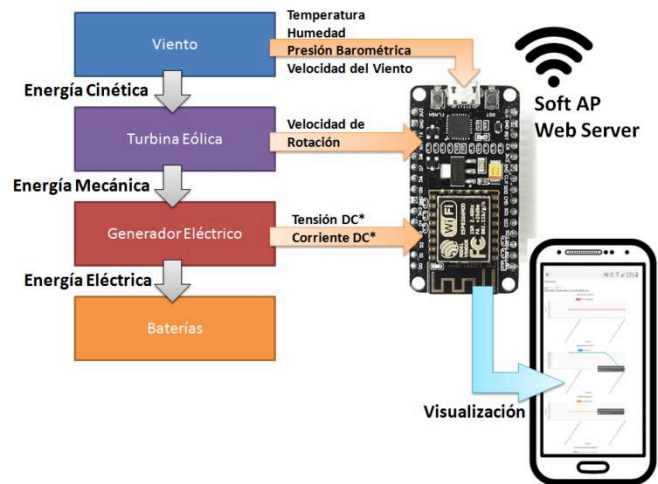


Fig. 1. Sistema inalámbrico de medición y visualización de datos basado en ESP8266.

IV. CONCLUSIONES

A partir del dispositivo presentado en este trabajo fue posible dar una solución de bajo costo a la necesidad del relevamiento y visualización de datos in situ, que permitan caracterizar el funcionamiento de un aerogenerador. La integración entre un sistema embebido y las posibilidades que ofrece el desarrollo de un servidor web, todo en un mismo dispositivo, representa una mejora respecto a la interacción con el usuario en comparación con soluciones pre-existentes.

- [1] Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & information systems engineering*, 6(4), 239-242.
- [2] Lee, E. A. (2008, May). Cyber physical systems: Design challenges. In *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)* (pp. 363-369). IEEE.
- [3] Srivastava, P., Bajaj, M., & Rana, A. S. (2018, March). IOT based controlling of hybrid energy system using ESP8266. In *2018 IEEMA Engineer Infinite Conference (eTechNXT)* (pp. 1-5). IEEE.
- [4] SriLakshmi, P., Das, N. L., & Manikumar, C. (2017). Low-cost wireless instrumentation for monitoring humidity, wind speed, and direction. *Instrumentation Science & Technology*, 45(5), 479-485.
- [5] Sarkar, I., Pal, B., Dutta, A., & Roy, S. (2018). Wi-Fi Based Portable Weather Station for Monitoring Temperature, Relative Humidity, Pressure, Precipitation, Wind Speed and Direction. In *International Conference on ICT for Sustainable Development (ICT4SD), India*.
- [6] International Electrotechnical Commission. Wind turbines –Part 12-1: Power performance measurements of electricity producing wind turbines. s.l.: IEC, 2005. IEC 61400-12-1:2005(E).

Desarrollo de dispositivos electrónicos para entrenamiento deportivo

Ramiro Romero Dopazo
*Laboratorio de Investigación en
Nuevas Tecnologías Informáticas*
La Plata, Argentina
ramiroromero94@gmail.com

Laura A. Fava
*Laboratorio de Investigación en
Nuevas Tecnologías Informáticas*
La Plata, Argentina
lfava@info.unlp.edu.ar

Diego Vilches Antao, Javier F. Díaz
*Laboratorio de Investigación en
Nuevas Tecnologías Informáticas*
La Plata, Argentina
dvilches, jdiaz@info.unlp.edu.ar

Resumen—En este trabajo se describe el proceso de desarrollo de un dispositivo electrónico, basado en sensores y LEDs, que se utilizan para hacer entrenamiento cognitivo de deportistas. El proceso tuvo tres iteraciones que incluyeron diferentes arquitecturas, dispositivos, protocolos y software.

Index Terms—sistemas embebidos, dispositivo electrónico, eda, printed circuit board, diseño de PCB, entrenamiento cognitivo-deportivo.

I. INTRODUCCIÓN

La ciencia está descubriendo que si bien toda la actividad física tiene un efecto positivo en el cerebro, el ejercicio que combina el uso de múltiples sentidos con movimientos corporales completos, desafía al cerebro a niveles más altos, que requieren una función cognitiva más compleja para tomar decisiones y ejecutar habilidades. En consecuencia, la práctica de estas actividades mejora la toma de decisiones y el rendimiento físico, sensorial y neurológico de los deportistas [1].

Conscientes de estas prácticas deportivas emergentes y de la evolución de las tecnologías, en el Laboratorio de Nuevas Tecnologías (LINTI) se comenzó a trabajar, entre otras cosas, en el desarrollo de dispositivos para deporte basados en sensores y LEDs para mejorar el entrenamiento y rendimiento de los deportistas de élite [2]. El dispositivo consiste en un conjunto de luces LEDs que se prenden según cierta configuración de rutinas y se apagan cuando el deportista pasa sobre ellas, lo cual es detectado por un sensor de proximidad. Las rutinas pueden usar grupos de dispositivos para lograr actividades más atractivas.

En este artículo, se sintetiza el proceso de desarrollo de estos dispositivos que involucró tres iteraciones, cada una arrojando un prototipo evolucionado. Para cada iteración se detalla la arquitectura, junto con el protocolo, software y hardware involucrado para la creación del producto.

II. DESARROLLO DE LOS PROTOTIPOS

El proceso de desarrollo del producto actual, comenzó con una exploración de componentes, luego diseño y desarrollo de un primer sistema (prototipo 2017) que sirvió para realizar pruebas de conceptos. En una segunda iteración se realizaron cambios al diseño de la placa y al software, principalmente para añadir nuevas funcionalidades (prototipo 2018) y por

último, en la iteración actual (prototipo 2019), se está realizando un nuevo diseño del sistema, se actualizaron algunos componentes y se está desarrollando una nueva versión del software. Para cualquiera de sus versiones, los prototipos incluyeron uno o varios dispositivos o nodos y un dispositivo controlador, encargado de manejar el estado de cada nodo.

Todas las iteraciones incluyen:

- El protocolo de comunicación nodo-controlador. Varía entre los prototipos.
- El software controlador de nodos. Varía entre los prototipos.
- El firmware del microcontrolador embebido en el hardware del nodo. Se encarga del control de las luces y de la lectura de los sensores en cada nodo, además de implementar el protocolo de comunicación.

III. FASES DE DESARROLLO DEL DISPOSITIVO ELECTRÓNICO

En todos los prototipos se utilizó el SoC¹ ESP8266EX de Espressif como microcontrolador para el hardware del nodo. El chip ESP8266EX combina un microcontrolador Tensilica Xtensa L106, un RISC de 32 bits corriendo a 80 Mhz, con hardware para la funcionalidad Wi-Fi [5].

III-A. Prototipo 2017 (Golfina)

Para este prototipo se diseñó un circuito impreso de capa simple con los siguientes componentes: Una placa *Adafruit Feather HUZZAH* que combina un módulo ESP12E con regulación de tensión, un circuito de cargado de baterías de polímero de litio y un puerto micro-USB por el cual se programa el firmware y se carga la batería. El módulo ESP12E le provee al chip memoria flash y una antena de radiofrecuencia para la comunicación WiFi, un sensor de proximidad IR-08H, un *circuito driver* para alimentar los LEDs y un interruptor *ON-OFF*.

En esta primera iteración se cubrió la funcionalidad básica del sistema: un dispositivo que se conecta a una red Wi-Fi, recibe comandos que «activan» encienden las luces y reportan eventos de proximidad para el posterior cómputo de estadísticas de los tiempos de reacción.

¹System on a Chip, en un sólo chip que integran el procesador, memoria, almacenamiento, y controladores de entrada/salida.

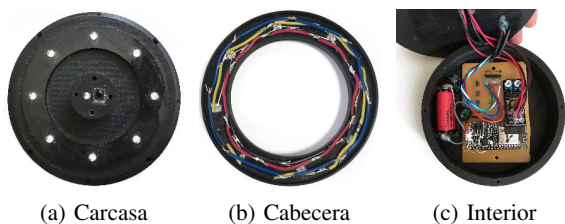


Figura 1: Prototipo 2017

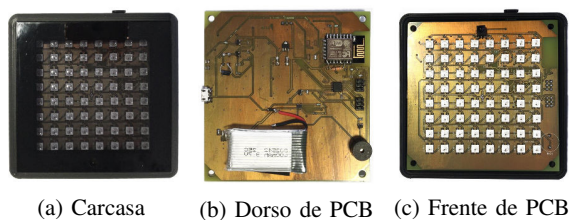


Figura 2: Prototipo 2018

La fabricación del circuito impreso se realizó mediante el método de transferencia de tóner, dado que se fabricaron únicamente seis unidades, y la simpleza del circuito permitía tener trazas con margen de error relativamente alto. El Firmware fue escrito en C, utilizando el SDK provisto por Espressif, que provee funcionalidad para controlar las conexiones TCP y los paquetes UDP.

Se diseñó e implementó un protocolo de aplicación binario específico para este sistema. Su unidad de dato es un paquete de longitud fija de 16 bytes. Estos paquetes se encapsulan en datagramas UDP o se transmiten a través de conexiones TCP, dependiendo de su tipo y función. La aplicación móvil de administración de las rutinas incluye el módulo controlador.

III-B. Prototipo 2018 (Prieta)

Para el prototipo 2018, se optó por diseñar una placa impresa doble capa que integre directamente el módulo ESP-12F, en vez de tener el módulo HUZAH, como es en el caso del prototipo 2017. Esta decisión de diseño permitió tener control sobre aspectos que antes eran manejados por el circuito dentro del módulo HUZAH. Además, se cambiaron los LEDs que previamente eran controlados como una sola unidad y se los reemplazaron por una matriz de LEDs RGB direccionables.

Se añadió un circuito de control de prendido y apagado del tipo «press ON, hold OFF», con posibilidad de autoapagado, se implementó un circuito para la medición (aproximada) de batería, se hizo uso extenso de componentes de montaje superficial, para mayor manufacturabilidad, se recreó el circuito para el sensado de proximidad (que previamente formaba parte del módulo IR-08H) de forma que la sensibilidad pueda ser ajustable y se utilice un sensor IR con filtro pasabanda centrado en 56 KHz sin control automático de ganancia y se añadió un buzzer para emitir pitidos.

El controlador fue movido de la aplicación Android a una Raspberry Pi 3B+. Se mantuvo la interfaz de usuario en el dispositivo móvil (celular o tablet) pero se delegó el control de los nodos al programa controlador residente en la Raspberry. Se hizo esto a causa de una limitación en Android que hacía

que resulte más difícil mantener procesos en segundo plano sin que el sistema operativo lo termine [7]. En el caso de iOS, habría resultado imposible ejecutar las rutinas desde el dispositivo móvil, ya que iOS tiene políticas estrictas de uso de CPU en segundo plano [8].

Para esta iteración se decidió utilizar MQTT [4], con el objeto de usar un protocolo bien establecido en IoT. Sin embargo, los *payloads* de los mensajes continuaron siendo estructuras binarias.

III-C. Prototipo 2019 (Laúd)

Para el prototipo actual, 2019, se conservará el hardware del nodo y se realizarán mejoras en cuanto al software del controlador y de la aplicación móvil. El controlador se desarrolla en lenguaje de programación Go y para la aplicación de administración se utiliza el *framework* React Native (para producir una versión para Android y otra para iOS).

En cuanto al protocolo de comunicación, se está evaluando volver a un protocolo propio, es decir, similar a la primera versión, un binario sobre TCP/UDP modificado, ya que el uso de MQTT, si bien es un estándar para IoT, no ha mejorado la implementación de la comunicación en ningún aspecto.

IV. RESULTADOS

En la primera iteración, se obtuvo como resultado un sistema de entrenamiento enteramente funcional formado por seis prototipos de nodos y una aplicación Android para controlarlos. Este sistema sirvió para hacer pruebas con equipos de fútbol y de hockey de clubes de La Plata. A partir de estos resultados, se comenzó con la segunda iteración, en la cual se obtuvieron dos prototipos con placas de mayor manufacturabilidad y un programa Java que se ejecuta en una Raspberry Pi y controla los nodos.

Se continúa trabajando en la iteración actual en base a la experiencia adquirida en los prototipos anteriores. El objetivo actual es realizar modificaciones mínimas a la placa del prototipo 2018, utilizando una implementación modificada del protocolo de la primera iteración, y agilizar el proceso de fabricación.

Cuadro I: Componentes y protocolos de cada iteración

	2017	2018	2019
Nodo	PCB capa simple	PCB capa doble	PCB capa doble
Controlador	Biblioteca Java usada en Android	Aplicación Standalone Java en Raspberry Pi	Go
Protocolo	Binario sobre TCP/UDP	Binario MQTT	Binario sobre TCP/UDP modificado
Software usuario	Aplicación Android	Aplicación Android	Aplicación React Native

REFERENCIAS

- [1] Cathi Lamberti, "Brain Training Enhancing Athletic Performance", SMARTfit Multisensory Fitness. <https://www.ibm.com/developerworks/ssa/library/iot-mqtt-why-good-for-iot/index.html>, Inc., Febrero 2018.
- [2] Fava, L., Vilches, D., Romero Dapozo, R, and Díaz, J.. "Tecnología aplicada al deporte de alto rendimiento". XX Workshop de Investigadores en Ciencias de la Computación, Corrientes, Argentina. ISBN 978-987-3619-27-4. Abril, 2018.
- [3] Roy T. Fielding, James Gettys and Jeffrey C. Mogul, Henrik Frystyk Nielsen, Larry Masinter, Paul J. Leach, Tim Berners-Lee, "Hypertext Transfer protocol – HTTP/1.1" Internet Request for Comments, no. 2616, Junio 1999.
- [4] Krajjak, S., and Tuwanut, P. "A survey on internet of things architecture, protocols, possible applications, security, privacy, real-world implementation and future trends". In Communication Technology (ICCT), 2015 IEEE 16th International Conference on (pp. 26-31). Octubre, 2015.
- [5] Espressif "ESP8266 Technical Reference", https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf. Accedido en abril, 2019.
- [6] Espressif "ESP8266 NON-OS SDK Reference ", https://www.espressif.com/sites/default/files/documentation/2c-esp8266_non_os_sdk_api_reference_en.pdf. Accedido en abril, 2019.
- [7] Google Developers. "Services Overview". <https://developer.android.com/guide/components/services>. Accedido en abril, 2019.
- [8] Apple. "App Programming Guide for iOS". <https://developer.apple.com/library/archive/documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/BackgroundExecution/BackgroundExecution.html>. Accedido en abril, 2019.

Análisis del desempeño de sensor embebido basado en etiqueta RFID pasiva

Pablo Garrone

Facultad de Ingeniería
Universidad Nacional de La Pampa
General Pico, La Pampa, Argentina
pdgarrone@gmail.com

Martín Baudino

Facultad de Ingeniería
Universidad Nacional de La Pampa
General Pico, La Pampa, Argentina
martinbaudino@gmail.com

Diego Vicente

Facultad de Ingeniería
Universidad Nacional de La Pampa
General Pico, La Pampa, Argentina
vicente@ing.unlpam.edu.ar

Abstract—Este trabajo presenta la arquitectura y algunos resultados obtenidos en el desarrollo de un sensor inalámbrico para medición de temperatura en cojinetes de bielas de máquinas alternativas. El sensor se basa en una etiqueta RFID que integra un transponder, un microcontrolador y un conversor analógico-digital. Además, se agregó un regulador de voltaje externo, con el fin de mejorar la capacidad de la etiqueta para recuperar y almacenar energía de la señal proveniente del lector. La evaluación de las mejoras obtenidas, a partir de esta incorporación, es el foco de este trabajo.

Keywords—sensor embebido, transferencia inalámbrica de potencia, RFID, bajo consumo

I. INTRODUCCIÓN

La Identificación por Radio Frecuencia (RFID) es una tecnología que utiliza ondas de radio para enviar y recibir información. Un sistema RFID típico consiste en un lector y una etiqueta que se comunican mediante esta técnica [1][2]. Los recientes avances en Transferencia Inalámbrica de Potencia (WPT) [3][4][5] han posibilitado la aparición de sensores inalámbricos basados en etiquetas RFID pasivas [6][7]. De esta manera se elimina la necesidad de conexiones cableadas y baterías.

Además, las etiquetas RFID pueden ser integradas junto a otros componentes electrónicos como conversores analógico-digitales (ADC), sensores de bajo consumo e incluso microcontroladores, convirtiéndose en módulos de sensado completamente integrados. En estos casos, la etiqueta RFID puede ser utilizada tanto como interfaz de comunicación para la transmisión de datos, como también para la captura de energía de radiofrecuencia (RF), alimentando los circuitos electrónicos que realizan la función de sensado [6][7].

Algunas aplicaciones para estos módulos RFID con sensores incorporados se pueden encontrar en implantes biomédicos [8], sensores embebidos en concreto [9] y monitoreo de condición estructural [10]. Siguiendo con esta tendencia, en [11] y [12] se presentó un prototipo basado en una etiqueta de sensado RFID para monitoreo de temperatura en cojinetes de bielas, el cual se muestra en la Fig. 1. En el mismo, la etiqueta sensora, solidaria a la biela, adquiere datos de temperatura desde las termocuplas adosadas a los cojinetes. El lector RFID, solidario a la carcasa de la máquina, provee de manera inalámbrica energía suficiente para la operación de la etiqueta y recibe datos provenientes de ésta. Sin embargo, en este prototipo original, la comunicación entre el lector y la etiqueta RFID solo era posible durante una pequeña fracción del ciclo de revolución del cigüeñal de la máquina.

Para solucionar este problema, se trabajó sobre dos aspectos del sistema. Por un lado, se rediseñó la antena del lector [12] y por otro, se realizaron modificaciones sobre el hardware de la etiqueta, cuyos resultados fundamentales se presentan en este trabajo.

Inicialmente se realiza una descripción de la arquitectura de la etiqueta completa y la secuencia de operación para adquisición de datos y comunicación con la estación base. Seguidamente se describen los resultados obtenidos a partir de la incorporación de un regulador de tensión externo a la etiqueta. A continuación, se compara la dinámica de las tensiones de alimentación de la etiqueta durante toda la secuencia de ejecución de una trama completa de comunicación, adquisición y procesamiento de datos. Finalmente, se presentan las conclusiones más importantes acerca de los resultados obtenidos.

II. ARQUITECTURA DE LA ETIQUETA PASIVA

En la Fig. 2 se muestra un diagrama de bloques de la interfaz RFID ez430-TMS37157 [13] con las adaptaciones introducidas. Al hardware original de la etiqueta se le agregó el circuito integrado (IC) ADS1018, un ADC utilizado para adquirir las señales de las termocuplas. Además, con el propósito de garantizar la disponibilidad de energía suficiente para las etapas de comunicación y procesamiento de información de los sensores, se incorporó el IC regulador de tensión externo adicional TPS71433 como se hizo en [14].

Como se puede ver en la Fig. 2, a diferencia del microcontrolador (μC), el ADC no se encuentra alimentado desde el pin VBAT, sino desde el terminal VBATI del IC TMS37157, luego de un interruptor electrónico. Esto se debe a que, si bien el ADS1018 es un ADC de bajo consumo [15], la potencia requerida por éste es significativamente mayor a la correspondiente al resto de la etiqueta. De esta manera, solo se activa cuando se envía un comando MSPAccess desde el lector a la etiqueta, posibilitando que la energía se acumule en el capacitor CBAT más rápidamente.

Cuando se requiere información de temperatura, el lector inicia la transferencia de energía hacia la etiqueta, que se acumula en CBAT, mientras el μC MSP430 permanece en

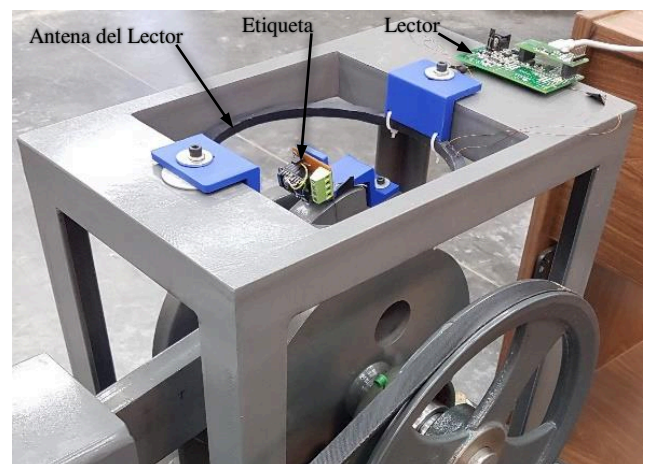


Fig. 1: Prototipo del sistema implementado.

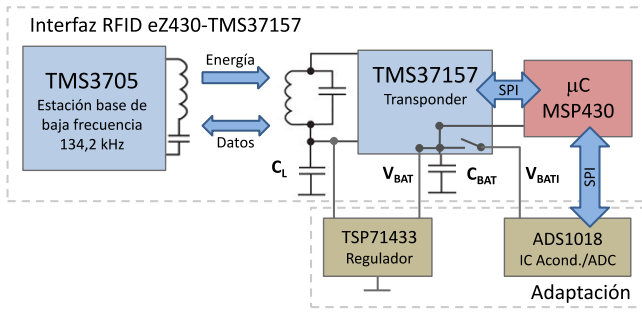


Fig. 2: Diagrama en bloques de la interfaz y adaptaciones en la etiqueta.

modo de bajo consumo. Luego, cuando el lector envía un comando MSPAccess, el μC es activado y el ADS1018 es energizado a través de VBATI, iniciándose la rutina de adquisición de la temperatura en cada uno de los cojinetes. Finalmente, el μC transfiere los datos de lectura obtenidos al transponder TMS37157 para que este los reenvíe al lector al concluir la trama de escritura.

III. ANÁLISIS DEL DESEMPEÑO DE LA ETIQUETA

Con la incorporación de un regulador de voltaje externo, se pueden destacar dos situaciones en las que se observan mejoras en la capacidad de recuperación y almacenamiento de energía RF enviada desde el lector. En el primer caso, cuando las antenas del lector y la etiqueta se encuentran a poca distancia, la potencia RF disponible en la etiqueta es mayor. En estas condiciones, el regulador externo facilita la acumulación de carga en CBAT a una tasa superior a la admitida por el regulador interno del TMS37157 [16]. En la Fig. 3.a se compara la evolución de la tensión de carga en CBAT (VBAT) a través de ambos reguladores en la situación antes descrita. Como se puede ver, el regulador interno limita la tasa de carga de CBAT, mientras que con el uso del regulador externo ésta aumenta en una proporción del 38% respecto a la anterior. Estas mediciones se realizaron alimentando el lector con una tensión de 14V, para un valor de CBAT de 10 μ F y utilizando la antena original de la interfaz ez430-TMS37157 [17], con la etiqueta ubicada de manera que la distancia entre los centros de las antenas sea de 15mm y con la orientación relativa más conveniente.

En el segundo caso, cuando la etiqueta se encuentra

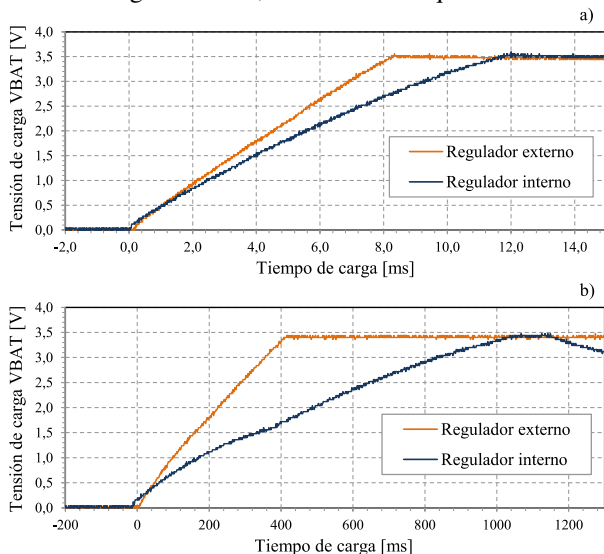


Fig. 3: Comparación de las curvas de carga con y sin regulador externo. a) Etiqueta y lector separados 15mm. b) Etiqueta y lector separados 75mm.

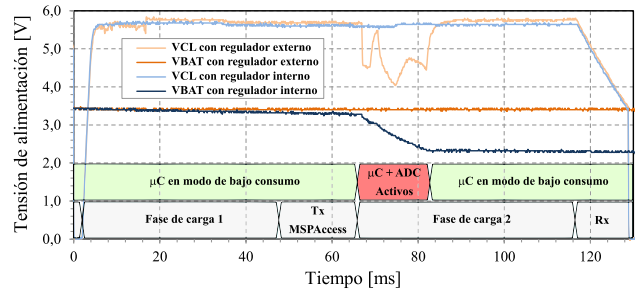


Fig. 4: Dinámica de tensión durante una secuencia completa de operación.

alejada del lector, la potencia RF disponible en la antena disminuye y se requiere mayor tiempo de carga para alcanzar la tensión mínima de funcionamiento. En la Fig. 3.b se puede ver que, en esta situación, mediante el regulador externo la tasa de acumulación de carga en CBAT aumenta 160% respecto a la correspondiente al regulador interno. Estos resultados se obtuvieron en las condiciones ya mencionadas pero con una distancia de 75mm entre los centros de las antenas. Se comprobó experimentalmente que gracias al incremento en la corriente de carga en CBAT, el rango de comunicación se incrementa un 26%.

Otra consecuencia importante de utilizar un regulador externo es la capacidad de recuperar energía en CBAT sin intervención de la unidad de control del TMS37157 [16]. En el sistema de carga original, solo es posible almacenar energía en CBAT si se envía inicialmente un comando de carga de batería y luego se mantiene activa la señal de RF, mientras que el regulador externo puede recuperar energía desde cualquier sección de la trama transmitida por el lector. En la Fig. 4 se compara la dinámica de las tensiones en CBAT y en CL durante la ejecución de una trama MSPAccess. Puede observarse que, utilizando el regulador externo, VBAT permanece constante durante toda la ejecución de la trama, mientras que con el regulador interno, esta tensión disminuye considerablemente durante la activación del μC y el ADC. Por otro lado, se puede ver que la tensión en el capacitor CL (VCL) presenta variaciones transitorias durante el período de mayor consumo cuando se utiliza el regulador externo. Sin embargo, en todo momento se mantiene por encima del mínimo requerido para la operación del transponder [16].

IV. CONCLUSIONES

Este trabajo presentó los principales avances en el desarrollo de un prototipo de sensor pasivo embebido para medición de temperatura en cojinetes de biela. Experimentalmente se comprobó que, con la incorporación del regulador de tensión externo la tasa de recuperación de energía de RF aumenta notablemente. Tanto en situaciones donde el acoplamiento entre las antenas del lector y la etiqueta es alto, como también cuando este es bajo, la tasa de almacenamiento de energía en la etiqueta es superior a la obtenida mediante el regulador interno del transponder. Esto no solo permite extender notablemente el rango de comunicación entre ambos dispositivos, sino también acumular la energía necesaria para adquirir, procesar y transmitir las temperaturas de ambos cojinetes en un solo paso. Finalmente, cabe destacar que se verificó sobre la máquina que la comunicación entre la etiqueta y el lector se mantiene durante todo el ciclo de revolución del sistema, aun cuando la velocidad de rotación del cigüeñal es alta.

REFERENCIAS

- [1] K. Domdouzis, B. Kumar, C. Anumba "Radio-Frequency Identification (RFID) applications: A brief introduction". *Advanced Engineering Informatics* 21 (2007) 350–355
- [2] K. Finkensteller. "The RFID Handbook, Fundamentals and applications in Contactless Smart Cards Radio Frequency Identification and Near-Field Communication". J. Wiley & Sons. 2010.
- [3] Surajit Das Barman, Ahmed Wasif Reza, Narendra Kumar, Md. Ershadul Karim, Abu Bakar Munir, "Wireless powering by magnetic resonant coupling: Recent trends in wireless power transfer system and its applications". *Renewable and Sustainable Energy Reviews*, vol. 51, November 2015, pp 1525–1552.
- [4] Stavros V. Georgakopoulos and Olutola Jonah, "Optimized wireless power transfer to RFID sensors via magnetic resonance". In *Proc. IEEE International Symposium on Antennas and Propagation*, Spokane, WA, USA, 3-8 July 2011. pp. 1421-1424.
- [5] Iker Mayordomo, Norbert van der Heyd, Tobias Drager and Josef Bernhard, "Development of a wireless power transmission system for embedded passive sensors using LF RFID technology". In *Proc. Fourth International EURASIP Workshop on RFID Technology*, Torino, Italy, 27-28 Sept. 2012. pp 65-71.
- [6] N. Cho et al., "A 5.1- μ W 0.3-mm² UHF RFID tag chip integrated with sensors for wireless environmental monitoring," in *Proc. IEEE Eur. Solid State Circuits Conf.*, Grenoble, France, Sep. 2005, pp. 279–282.
- [7] Alanson P. Sample, Daniel J. Yeager, Pauline S. Powledge, Alexander V. Mamishev, and Joshua R. Smith, "Design of an RFID-based battery-free programmable sensing platform". *IEEE Trans. on Instrumentation and Measurement*, vol. 57, no. 11, Nov 2008.
- [8] C. Occhiuzzi, M. Simiele, R. Lodato, G. Marrocco, "Feasibility, limitations and potentiality of UHF-RFID passive implants". In *Proc. IEEE International Conference on RFID-Technologies and Applications (RFID-TA)*, Nice, France, 5-7 Nov. 2012. pp 40-45.
- [9] Yongsheng Liu, Fangming Deng., Yigang He, Bing Li, Zhen Liang, and Shuangxi Zhou, "Novel concrete temperature monitoring method based on an embedded passive RFID sensor tag". *Sensor (Basel, Switzerland)* Vol 17, no. 7, 1463, July 2017.
- [10] Jun Zhang, Gui Yun Tian, Adi M. J. Marindra, Ali Imam Sunny and Ao Bo Zhao, "A review of passive RFID tag antenna-based sensors and systems for structural health monitoring". *Applications Sensors (Basel, Switzerland)* Vol 17, no. 2, 265, Feb 2017.
- [11] D. Vicente, M. Baudino, F. Mazzaferro y W. Fruccio, "Sensor RFID para monitoreo de temperatura en cojinetes de biela". VI Congreso Argentino de Ingeniería Mecánica VCAIM2018. San Miguel de Tucumán, República Argentina, 10 al 2 de Octubre de 2018.
- [12] D. Vicente, M. Baudino, J. Amigone and E. Berges, "Empirical RFID coil design considering environmental influences". *Argentine Conference on Automatic Control (AADECA)*. Buenos Aires, Argentina, 7-9 Nov. 2018.
- [13] eZ430-TMS37157 Development Tool. User's Guide. Texas instruments. 2010.
- [14] Andre Frantzke "A low-power battery-less wireless temperature and humidity sensor for the TI PaLFI device". Texas Instruments, App. Report SWRA395, Nov. 2011.
- [15] Texas Instruments, "ADS1018 Ultrasmall, Low-Power, SPI-Compatible, 12-Bit, Analog-to-Digital Converter With Internal Reference and Temperature Sensor", SBAS526C, Nov. 2012 [Revised Nov. 2015]
- [16] Texas Instruments, "Passive Low Frequency Interface Device With EEPROM and 134.2 kHz Transponder Interface", SWRS083A, Sep. 2009 [Revised Nov. 2009]
- [17] K. Aslanidis and J. Wyatt "LF HDX System Parameter Calculation for TMS37157 (PaLFI)". Texas Instruments, App. Report SCBA029, July 2010.

Plataforma flexible para nodos sensores inalámbricos.

Javier Belmonte¹
belmonte@fceia.unr.edu.ar

Néstor Marcón¹
nestormarcon@yahoo.com.ar

Marcelo Pistarelli¹
mpistare@fceia.unr.edu.ar

Rosa Corti¹
rcorti@fceia.unr.edu.ar

Roberto Martínez¹
romamar@fceia.unr.edu.ar

Juan Vecchio¹
juanpivec@gmail.com

¹Universidad Nacional de Rosario

Resumen—Las Redes Inalámbricas de Sensores Inteligentes tradicionales trabajan con datos escalares como temperatura o humedad, debido a que poseen fuertes restricciones de ancho de banda y consumo de energía. Sin embargo, en los últimos años, los avances tecnológicos han permitido un crecimiento significativo del uso de estas redes para procesamiento y transmisión de señales complejas como audio, video o vibraciones. En este trabajo se reporta el desarrollo de una plataforma flexible para nodos sensores capaces de medir, procesar, almacenar y transmitir inalámbricamente distintas variables físicas. El diseño de estos nodos permite su adecuación para el uso en aplicaciones que requieran de la adquisición y procesamiento de señales de complejidad diversa. Asimismo, los nodos permiten la transmisión inalámbrica de la información obtenida a partir del procesamiento local de las señales mencionadas.

Palabras clave —redes de sensores, estructuras civiles, vibraciones

I. INTRODUCCIÓN

Las Redes Inalámbricas de Sensores Inteligentes tradicionales (RISI) trabajan con datos escalares como temperatura, humedad o localización de objetos, debido a que poseen fuertes restricciones de ancho de banda y consumo de energía [1]. Sin embargo, los incrementos constantes en las escalas de integración, la aparición de protocolos de la más diversa índole, el aumento en la eficiencia de las baterías utilizadas en estas redes, las mejoras alcanzadas en los sensores y dispositivos de ultrabajo consumo, han permitido su evolución para poder incorporarlas en dominios que incluyen variables de mayor complejidad. En forma paralela a esta evolución se produce una explosión de los contenidos multimedia y sus técnicas de adquisición, procesamiento y transmisión. Pese a los grandes requerimientos de energía y ancho de banda normalmente necesarios para este tipo de datos, contrarios a las premisas de diseño de las RISI, en los últimos años han aparecido trabajos y tecnologías tendientes a definir un puente entre ambos universos, dando origen a las Redes Inalámbricas de Sensores Multimedia (RISM). Una RISM se define como una red de dispositivos inalámbricos que permiten la adquisición y transmisión de flujos de video, audio o imágenes, sumado al manejo de señales escalares habituales de las RISI [2]. En este sentido, se han llevado a cabo numerosos esfuerzos tendientes principalmente a adaptar el uso de la red para el transporte de estos flujos [3].

Es importante destacar, que en muchas aplicaciones no resulta de utilidad la transmisión de las señales complejas en sí mismas, sino datos extraídos u obtenidos de ellas en función de lo que efectivamente se requiera adquirir o medir. Incluso, aprovechar la naturaleza distribuida de las RISI para procesar los datos en los nodos sensores que realizan las

mediciones, eliminar redundancias en la información y optimizar las transmisiones [4]. Este último enfoque, fue el utilizado en el presente trabajo para abordar el desarrollo de aplicaciones en el dominio de la supervisión ambiental, sobre la base de técnicas de recolección y procesamiento local/distribuido para reducir en forma considerable la circulación de información en el sistema [5].

Un aspecto a considerar en esta área de trabajo es la dificultad existente para la definición de requerimientos típicos de las redes de sensores, debido a que se utilizan en aplicaciones y dominios extremadamente variados [6]. Los nodos sensores que las componen se deben modificar, o incluso rediseñar, atendiendo a factores particulares de cada aplicación y a las condiciones del ambiente donde se los emplaza [7]. Por lo tanto, la implementación final de estos dispositivos incluye, en la mayoría de los casos, componentes específicos para cubrir los requerimientos del problema, con el fin de obtener una solución eficiente que prolongue el tiempo de vida del sistema [8]. Sin embargo, esta situación impacta negativamente en términos de complejidad, costo y tiempo de desarrollo de las aplicaciones, motivo por el cual el diseño de plataformas reutilizables para redes de sensores ha recibido un particular interés en los últimos años [7]. Esta situación fue considerada por los autores desde el inicio del desarrollo, y jugó un rol fundamental en la selección de los componentes que se incluyen en la versión base de la plataforma y en las posibilidades de expansión y adaptación para otras versiones orientadas a aplicaciones que reúnan un conjunto de requerimientos comunes.

II. DISEÑO DE LA PLATAFORMA

A. Especificaciones generales

El modo de trabajo del nodo sensor consiste en la realización de mediciones de variables del medio, que se procesan y almacenan localmente. Periódicamente, estos datos son requeridos y descargados por un dispositivo de telefonía móvil. El último proceso se aborda combinando dos protocolos de comunicación inalámbrica, BLE y WiFi, con el objetivo de reducir el consumo de energía. El modo de trabajo elegido y el requerimiento de flexibilidad para la plataforma de los nodos sensores, se encaró diseñando una configuración base, capaz de cumplir con las siguientes especificaciones:

- Autonomía de batería para asegurar su funcionamiento por un mínimo de 30 días.
- Estanqueidad del dispositivo para poder ser emplazado en exteriores.
- Capacidad de procesamiento y almacenamiento local de datos.

- Acceso remoto de los datos almacenados localmente.
- Posibilidad de medición de magnitudes físicas diversas, soportando los correspondientes protocolos de comunicación on board de los dispositivos.

El diseño propuesto se puede personalizar para las aplicaciones en las que se lo utilice, a partir de la incorporación de componentes específicos necesarios para cada caso. Las distintas versiones de la plataforma base, se logran en virtud de la flexibilidad que le otorgan las características que se eligieron para su diseño.

B. Características del Hardware

El módulo EMW3239 de la empresa MXCHIP, se seleccionó como núcleo para el desarrollo debido a que integra un microcontrolador ARM de 32 bits, un chip WIFI/BLE y una memoria flash de 2MB. La Fig. 1 presenta el diagrama en bloques del módulo mencionado.

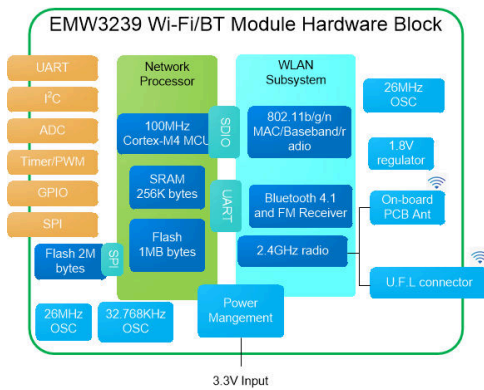


Fig. 1. Diagrama en bloques del módulo EMW3239

La plataforma diseñada cumple con las especificaciones anteriormente definidas a partir de las siguientes características:

- Disponibilidad de módulos de comunicaciones I2C, SPI y UART, que permiten conectar distintos tipos de sensores para cubrir requerimientos de diversas aplicaciones.
- Capacidad de procesamiento para soportar un protocolo de mayor nivel. El equipo de trabajo ha desarrollado CLUDITEM, un protocolo de encaminamiento jerárquico orientado a las aplicaciones de supervisión ambiental, que podrá ser incorporado a esta plataforma [9].
- Disponibilidad de cálculo en punto flotante y memoria RAM para reducir la grabación de datos a la FLASH externa.
- Disponibilidad de WiFi y BLE, posibilidad de agregar otros protocolos, como el 802.15.4 que brinda soporte a CLUDITEM y módulos GPRS.
- Modos de bajo consumo del controlador de WiFi/BLE para cumplir con la autonomía de funcionamiento requerida.

En el diseño y selección de los componentes se priorizó el ahorro de energía a fin de prolongar la vida útil del nodo cuya fuente principal de energía son baterías, como es habitual en las redes de sensores. Sin embargo, el diseño de la fuente de alimentación contempló la posibilidad de incluir

dispositivos sensores de consumo elevado y también la provisión de energía externa permanente o semipermanente. De esta manera se provee versatilidad a la provisión de energía de la plataforma para abordar aplicaciones de características diversas.

El firmware asociado se encuentra en etapa de desarrollo y se orienta a la definición de una estructura básica, que incluya la implementación de los servicios esenciales de administración del consumo, comunicaciones, datalogging, y modos de acceso para transferencia de la información registrada. El diseño contempla una fácil y rápida adaptación de estos módulos, así como la incorporación de otros específicos a la versión que se defina para la plataforma.

III. CASO DE ESTUDIO: MEDICIÓN DE VIBRACIONES EN PUENTES CARRETEROS

Los puentes carreteros, son estructuras civiles que se deterioran con el paso del tiempo como consecuencia de varios factores, entre ellos las vibraciones producidas por el tránsito vehicular. El uso de acelerómetros para medir estas vibraciones y su posterior análisis permiten evaluar el efecto sobre las estructuras. Este caso de estudio, una versión de la plataforma flexible presentada en este trabajo, está en etapa de desarrollo en un proyecto de fin de carrera.

El caso de estudio tiene por fin medir, procesar, almacenar y transmitir en forma inalámbrica variables físicas de interés, en particular vibraciones obtenidas en distintos puntos del puente. Los datos se recolectarán en forma periódica "off line", para lo cual se utilizarán las capacidades de datalogging de la plataforma, por un lado, y la flexibilidad en sus modos de comunicación, por otro. En esta aplicación, el nodo debe hacerse solidario al puente bajo estudio, por lo que las condiciones de montaje y estanqueidad del diseño son imprescindibles. Respecto de la medición de vibraciones en la estructura, estudios preliminares muestran que estarán comprendidas en el orden de milésimas de la aceleración de la gravedad (entre 0,02 m/s² y 0,3 m/s²) con un ancho de banda 25 Hz. En este sentido, se prevé incluir, en esta versión del nodo sensor, acelerómetros con una resolución de 16 bits. La Fig. 2 muestra el rendering de la placa diseñada.

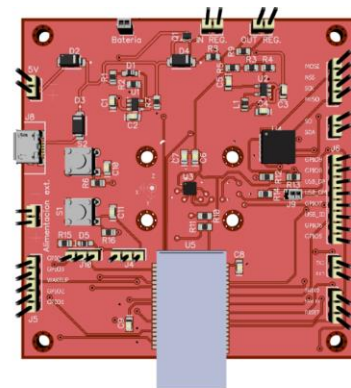


Fig. 2. Modelo 3D de la placa desarrollada

IV. TRABAJO FUTURO

La siguiente etapa es el emplazamiento de una red de quince nodos sensores de esta versión en un puente carretero en el área de la ciudad de Rosario, considerando la pertinencia de implementar el protocolo CLUDITEM, y extender su utilización para la adquisición de datos en otras estructuras civiles de características diferentes.

REFERENCIAS

- [1] M. Bhende, S. J. Wagh and A. Utpat , “A Quick Survey On Wireless Sensor Networks”, Fourth International Conference on Communication Systems and Network Technologies, 2014.
- [2] I.T. Almalkawi, M. Guerrero Zapata, J. N. Al-Karaki, J. Morillo-Pozo, “Wireless Multimedia Sensor Networks: Current Trends and Future Directions”, Sensors Journal , Issue 10, pp. 6662 – 6717, 2010.
- [3] G. Viscardi, M. Roveri, R. Camplani and C. Alippi, “Netbrick: A high-performance, low-power hardware platform for wireless and hybrid sensor networks“, IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2012), pp. 111-117, 2012.
- [4] M. Anupama Reddy, D. Anjaneyulu, D. Prasanth Varma and G. Rama Rao, “WSN in IOT Environment Interfacing with Smart Sensors using Arm7 with Zigbee for Industries”, International Journal of Engineering Research and Development, vol. 12, Issue, 8 pp. 54-60, 2016.
- [5] R. Corti, E. Giandoménico, J. Belmonte, R. Martínez, “Implementación de una Red de Sensores para aplicaciones de supervisión ambiental”, en los Anales de la VII Jornada de Ciencia y Tecnología de SECyT/UNR, Rosario 2013.
- [6] G. A. Al-Suhail, J. Mehdi, G. Nikolakopoulos. “A Practical Survey on Wireless Sensor Network Platforms”, Journal of Communications Technology, Electronics and Computer Science, Issue 13, 2017.
- [7] M. T. Lazarescu, "Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 3, no. 1, pp. 45-54, March 2013.
- [8] S. Bader, M. Krämer, B. Oelmann, “A Domain-Specific Platform for Research in Environmental Wireless Sensor Networks”, The Seventh International Conference on Sensor Technologies and Applications, pp 200 -207, 2013.
- [9] R. Corti, “Clustering dinámico para tiempo de encendido mínimo en redes inalámbricas de sensores (CLUDITEM)”, Tesis de Maestría, ISBN 978-950-34-0851-3, e-Book, 1a ed., La Plata: UNLP, 2012.

Robot Móvil Omnidireccional Experimental con Interfaz de Comandos Táctil y Vocal

Melisa Kuzman, Juan López, Juana Fernández, Raúl Rivera, Walter Gemin, Miguel Revuelta.

Universidad Nacional de Mar del Plata, Facultad de Ingeniería.
Depto. de Ingeniería Electrónica y Computación
Mar del Plata, Argentina
rrivera@fi.mdp.edu.ar

Resumen—Un vehículo robot omnidireccional, también conocido como holonómico, es un tipo de robot móvil con ruedas, cuya configuración le permite desplazarse en cualquier dirección sin la necesidad de alcanzar previamente una orientación específica.

En comparación con las plataformas convencionales, estos vehículos poseen múltiples ventajas en términos de su movilidad en espacios estrechos o ambientes congestionados. Tienen la capacidad de realizar fácilmente tareas en entornos con obstáculos estáticos, dinámicos o áreas reducidas.

Este trabajo describe la construcción de un móvil omnidireccional simple equipado con 4 ruedas Mecanum, la electrónica de control y potencia de los motores, la programación del intérprete de comandos, desplazamiento y comunicación en la placa Arduino del vehículo y la interfaz de comandos inalámbrica en un dispositivo móvil Android.

Palabras clave—robot, omnidireccional, Mecanum, Interfaz, Android

I. INTRODUCCION

El principal aspecto constructivo de un móvil omnidireccional lo constituye el diseño de sus ruedas. Las utilizadas en el presente trabajo son las que se conocen como Mecanum.

El empleo de estas ruedas requiere una instalación en el chasis como la observada en la Fig. 1, donde se muestra la descomposición de fuerzas que producen un desplazamiento hacia adelante, como resultado de la cancelación de las componentes en x y el refuerzo de las componentes en y . Si se invierte el sentido de giro de las ruedas se invierten los vectores T_1 , T_2 , T_3 y T_4 y el móvil se desplaza hacia atrás.

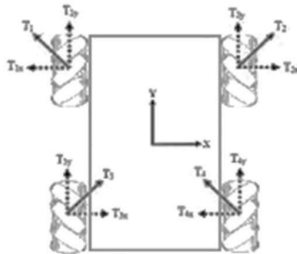


Fig. 1 Descomposición de fuerzas en un desplazamiento hacia adelante

Diferentes combinaciones de sentidos de giro y velocidades de rotación dan como resultado desplazamientos del chasis sin la necesidad de alcanzar previamente una orientación específica, Fig. 2.

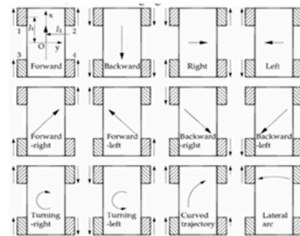


Fig 2. Combinaciones de sentidos de giro y velocidades y movimientos resultantes.

II. ELECTRÓNICA DE CONTROL Y POTENCIA

El circuito de potencia de excitación de los motores de CC en este trabajo es un puente H L298 conectado a la placa Arduino [3] que permite controlar el sentido de giro, velocidad y frenado del motor.

Las entradas de control del puente H utilizadas en este trabajo son los pares (IN) para la programación del sentido de giro, y las habilitaciones (EN) para la velocidad de los motores. La asignación de pines de la conexión entre los puentes H y la placa Arduino se muestra en la Fig.3.

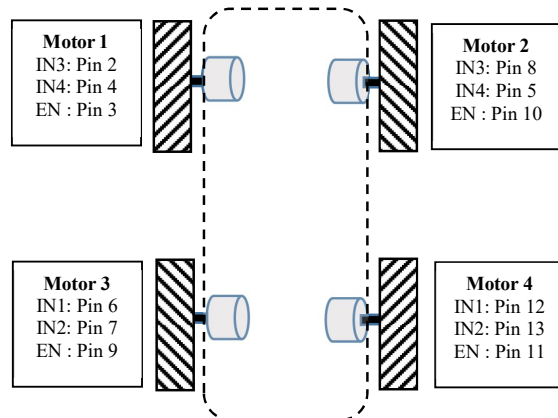


Fig. 3. Asignación de pines de los Puentes H y Arduino.

III. PROGRAMA DE MOVIMIENTO Y COMUNICACIÓN EN LA PLACA ARDUINO DEL VEHÍCULO

El firmware que controla el desplazamiento del vehículo es esencialmente un intérprete de comandos. El repertorio de los mismos se corresponde con los movimientos mostrados en la Fig. 2. Las subrutinas asociadas configuran los pines IN para el sentido de giro y los EN para la velocidad. La comunicación es bidireccional, mediante un módulo Bluetooth que además reporta el estado de la batería a partir de una lectura de la tensión en un canal analógico de la placa Arduino.

IV. INTERFAZ DE COMANDO EN EL DISPOSITIVO MÓVIL

La Fig. 4 muestra la interfaz gráfica desarrollada en una plataforma Android [4][5][6], compuesta por un control deslizante e indicación numérica que programa la velocidad, los botones de accionamiento de los comandos de desplazamiento para el móvil, la activación del micrófono para enviar órdenes verbales y la conexión y monitoreo de la comunicación Bluetooth e indicador del estado de la batería del móvil.



Fig. 4 Interfaz gráfica

Para el reconocimiento de voz (voz a texto) se utiliza la librería Voice Recognizer (VR) y para la respuesta audible (texto a voz) la librería Text To Speech (TTS). Una vez establecida la comunicación, los comandos pueden ejecutarse en forma indistinta (manual o vocal).

V. DISEÑO DEL PROTOTIPO EXPERIMENTAL

La Fig. 5 muestra el prototipo, construido en el laboratorio, en base a un chasis estándar, cuatro ruedas Mecanum, 4 motores DC, controlador basado en Arduino UNO, adaptador Bluetooth y un regulador de 8V de salida alimentado por un pack de 3 baterías LiPo de 3,7V/1,1A.

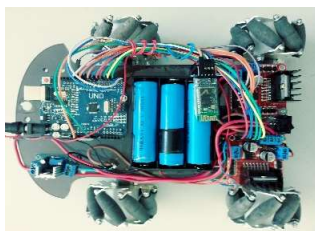


Fig. 5 Movil Omniwheel

Un aspecto importante en el control de velocidad de los motores es la frecuencia de la señal PWM, que está condicionada por el circuito eléctrico equivalente serie R_a - L_a del rotor del motor DC. En el momento del arranque la fuerza electromotriz e , proporcional al movimiento del rotor, es cero y la corriente i_a depende de los valores de L_a y R_a del

bobinado. La relación $\tau=L_a/R_a$ determina la constante de tiempo que limita la corriente de arranque y por lo tanto el torque inicial disponible[7][8]. La Fig. 7 muestra las respuestas para las relaciones Γ_{min}/τ de 5, 2.5 y 1 y un ciclo de trabajo Γ_{min}/T del 10%, donde T es el periodo de la señal PWM.

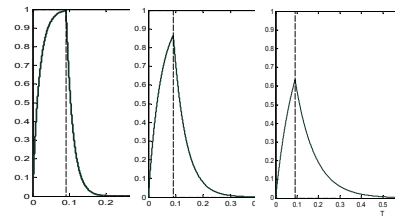


Fig. 6 currientes de arranque para Γ_{min}/τ , a) 5, b) 2.5, c) 1

En base a este criterio, y en función de un ajuste homogéneo de las 4 salidas PWM utilizadas en el control de los motores, se configuraron los timers de la placa Arduino para obtener frecuencias de 30,64 Hz.

VI. CONCLUSIONES

La construcción y experimentación con el modelo de robot omnidireccional descrito permitió identificar los aspectos constructivos y de programación relevantes en el diseño y la selección de sus componentes.

Se observó que es fundamental tener en cuenta el apareamiento de los motores, en lo que respecta a sus características eléctricas, para obtener el comportamiento esperado.

Las características de la interfaz Android desarrollada permite un control simple e intuitivo del móvil y un autodiagnóstico, en tiempo real, de sus estados de conexión y baterías.

REFERENCIAS

- [1] Florentina Adăscăliței, Ioan Doroftei, "Practical Applications for Mobile Robots based on Mecanum Wheels", Proc. of International Conf. On Innovations, Tech Products Develop.– MECAHITECH'11, vol. 3, 2011.
- [2] I. Güçlü, "Autonomous path-to-path movement for a vehicle with mecanum wheels", Master of Sc Thesis, Faculty of Mechanical, Maritime and Materials Engineering - Delft University of Technology, 2014
- [3] Arduino, "Arduino Uno," ed. <http://www.arduino.cc/>, 2019.
- [4] Android, <https://www.android.com/>.
- [5] Kevin Purdy, "The Complete Android Guide", Lulu.com, 2010.
- [6] Nicolas Gramlich, "Handbook - Android Programming", anddev.org, 2008.
- [7] Applications Notes, "AB-022 : PWM Frequency For Linear Motion Control", <https://www.precisionmicrodrives.com/content/ab-022-pwm-frequency-for-linear-motion-control>, Precision Microdrives, April 2019.
- [8] Motor DC Tutorial, "DC Motor Calculations", <https://www.micromo.com/technical-library/dc-motor-tutorials/motor-calculations>, FAULHABER Drive Systems, Abril 2019.

Sistema de integración de datos para vehículo terrestre no tripulado

J. E. Benavídez
U.T.N. - F.R.P.
jesusbenavidez@alu.frp.utn.edu.ar

E. S. Burgos
U.T.N. - F.R.P.
sergioburgos@frp.utn.edu.ar

L. Giovanini
SINC(i) Centro de investigación
lgiovanini@sinc.unl.edu.ar

Resumen—En este trabajo se presenta la implementación de un sistema de integración de datos utilizando datos de una unidad de medición inercial (IMU), datos provenientes de un sistema GPS y un algoritmo de fusión de datos. Se buscó un sistema con bajo costo. Este algoritmo tiene la tarea de fusionar la información proveniente de la IMU y generar un cuaternión que nos indica la orientación del vehículo.

computacional y capaz de integrar en trabajos de robótica móvil.

Keywords—Unidad de medición inercial, algoritmo de fusión de datos, cuaterniones, ángulos de Euler, GPS.

I. INTRODUCCIÓN

Las aplicaciones de la robótica móvil, cada vez más comunes, requieren del uso de un sistema de posicionamiento que les permita no sólo saber su posición (mediante coordenadas de GPS) sino también conocer la orientación del vehículo respecto a ciertos ejes de referencia.

En el desarrollo de algoritmos para el traslado de robots es fácil notar la necesidad de conocer dónde estamos y a dónde queremos llegar. Pero al profundizar en la secuencia de movimientos que se deben realizar para alcanzar el punto de destino, surge la necesidad de conocer como se encuentra orientado el robot respecto del punto de destino.

En el presente trabajo se describirán y compararán algoritmos cuyo desarrollo tuvo como objetivo cubrir las necesidades expuestas en el párrafo anterior.

II. CARACTERÍSTICAS DESEADAS

El sistema diseñado forma parte de desarrollos relacionados a robótica móvil, por lo que las características deseadas para este sistema debieron ser coherentes con las disponibles en los robots móviles sobre los que se trabajó. Estos tienen alimentación mediante baterías con regulación de tensión a 3.3[V] y 5[V], basan su control en un microcontrolador TM4C1233H6PM [1] de la firma Texas Instruments, cuyo núcleo es un ARM Cortex M4F.

Por lo tanto se requirió que el sistema tuviera las siguientes características:

- Alimentación de 3.3[V] o 5[V] para todos los módulos y sensores necesarios, incluyendo el módulo GPS y la Unidad de Medición Inercial.
- El microcontrolador mencionado cuenta con módulos de comunicación I2C, UART y SPI por lo tanto los

sensores y módulos utilizados deben poder comunicarse por algunos de estos protocolos.

- Bajo costo computacional, esto es necesario ya que el mismo microcontrolador que ejecuta estos algoritmos tiene, también, la tarea de ejecutar los algoritmos de control del vehículo.

III. POSICIÓN COORDENADA GEOGRÁFICA

Cualquier cuerpo puede ubicarse en el globo haciendo uso de algún sistema de posicionamiento. En este caso se decidió utilizar el sistema GPS. Para la obtención de los datos, a través del sistema mencionado, se utilizó el módulo GY-NEO6MV2.

El estándar NMEA-0183 [2] fue definido y es controlado por la *National Marine Electronics Association*. El formato establece flujos de datos representados en caracteres y cada línea de texto enviada contiene los datos y un comando que indica cómo interpretar estos datos.

Este formato en el que se encuentra presentada la información requiere de un algoritmo de procesamiento del flujo de datos para poder extraer la información relevante. A fin de minimizar el tiempo de procesamiento del flujo de datos se optó por utilizar el algoritmo y la metodología propuesta en [3].

IV. ORIENTACIÓN

Como ya se mencionó anteriormente con la localización de un cuerpo no es suficiente para analizar el movimiento que éste debe realizar para ir de un punto a otro, sino que hace falta también tener una idea de la orientación actual del cuerpo con respecto al punto donde desea llegar.

Si evaluamos la situación de movernos sobre la superficie de la tierra necesitaríamos el *bearing*, así se le denomina al ángulo horizontal que existe entre la dirección de un cuerpo y el punto donde se quiere llegar. Este ángulo puede ser calculado sabiendo las coordenadas del lugar donde se encuentra el vehículo y el punto donde se quiere llegar.

Para poder orientar el vehículo en la dirección del *bearing* es necesario conseguir la orientación magnética en la dirección del desplazamiento del vehículo (conocida como *Heading*), que se puede obtener mediante una brújula.

Sin embargo, en la situación hasta aquí evaluada no involucra la posibilidad de que los puntos de origen y de llegada se encuentren a diferentes alturas, lo que implicaría la necesidad de saber la información del ángulo que tiene el vehículo con respecto a la vertical.

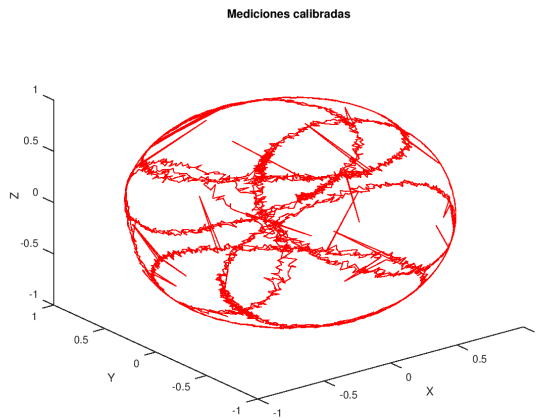


Figura 1: Gráfica de las mediciones calibradas.

Para poder obtener la información que se menciona en el párrafo anterior se utilizó la Unidad de Medición Inercial (IMU) LSM9DS1 [4] de la firma *ST Microelectronics*. Esta IMU cuenta con un acelerómetro, un giróscopo y un magnetómetro todos ellos con tres ejes de medición. La información del magnetómetro es utilizada para obtener la orientación magnética y ésta es mejorada con las velocidades angulares medidas con el giróscopo. Mientras que con el sensor acelerómetro se detecta la dirección de la gravedad, con esto se puede conocer la inclinación que se tiene respecto de la vertical.

Para lograr el proceso de fusión de datos de forma exitosa, se debió en un primer momento calibrar el magnetómetro y se utilizó un algoritmo de fusión de datos. Ambos procesos del desarrollo se detallan en las subsecciones siguientes.

IV-A. Calibración del magnetómetro

El magnetómetro es el sensor cuyo error en la medición toma la mayor importancia en el algoritmo. Los errores son en su mayoría errores del sensor excepto por aquellos que son debido a los efectos de *Hierro duro* y *Hierro blando*, estos efectos son distorsiones del campo magnético.

El efecto de *Hierro duro* es un campo magnético constante que se adiciona al campo magnético terrestre que se quiere medir, es producto de la magnetización remanente de materiales ferromagnéticos presentes en el ensamblaje del equipo. El efecto de *Hierro blando* es producto de la desviación de las líneas de campo magnético que producen los materiales ferromagnéticos presentes en la cercanía del sensor y su comportamiento es diferente dependiendo de la magnitud y de la dirección del campo magnético.

Los errores involucrados en la medición se pueden separar en errores del tipo matricial y errores aditivos. En la Expresión 1 podemos ver una forma reducida de cómo afectan a la medición los diferentes errores, éstos se encuentran agrupados en un coeficiente matricial K y un b que compensa los errores aditivos sufridos en la medición.

$$m = Km_i + b \quad (1)$$

En la Expresión 1 encontramos m que corresponde con la medición entregada por el sensor y la magnitud real del campo magnético m_i , que se encuentra despejada en la Expresión 2.

$$m_i = K^{-1}(m - b) \quad (2)$$

En la expresión 2 encontramos K^{-1} que corresponde con la inversa de la matriz K . Por lo tanto en el proceso de calibración se debe obtener el coeficiente K (o su inversa) y el vector b .

Para la calibración se utilizó el procedimiento presentado por *Deng Yang* [5]. En él se describe el procedimiento de obtención de datos y el algoritmo que se aplica a esos datos para obtener el coeficiente matricial del error K y el error aditivo b . Este algoritmo requiere, entre sus argumentos, la componente perpendicular (a la superficie terrestre) del campo magnético, éste valor se obtuvo de una tabla presente en [6].

Para observar cualitativamente la calibración conseguida con el K y b obtenidos, se realizó la gráfica de los valores calibrados.

En la Figura 1 se puede observar que las mediciones calibradas, mediante la utilización de la Expresión 2, forman una esfera centrada en el origen.

Una vez calibrado el magnetómetro, se presentó el problema de que el campo magnético terrestre presenta diferentes declinaciones dependiendo del lugar donde se mida, este error se resolvió mediante una simple sustracción entre el valor medido y la declinación del campo presente en la ciudad de Paraná. La declinación del campo se obtuvo de [7].

IV-B. Calibración del giróscopo

En sensores giróscopos de tecnología iNEMO, como el utilizado, el error más común es el error sistemático conocido como *Bias Error* por su nombre en inglés [5].

La corrección de este error se realiza de forma sencilla, ya que es un error aditivo permanente. El proceso consiste en tomar mediciones del giróscopo, con éste en estado de reposo y utilizar la media de estas mediciones como valor de corrección.

IV-C. Algoritmo de fusión de datos

Este algoritmo tiene la tarea de fusionar la información proveniente de la IMU y generar un cuaternión que nos indica la orientación del vehículo.

Para lograr la fusión se compararon los algoritmos de *Madgwick* [8] y de *Mahony* [9]. A continuación, se expone brevemente los principios de funcionamiento de ambos algoritmos.

1) *Madgwick*: Este algoritmo está basado en el método de optimización de Newton usando una formulación analítica del gradiente. Esto último es posible gracias a la representación del movimiento a través de cuaterniones.

2) *Mahony*: En este algoritmo el filtro se plantea aprovechando las propiedades presentes debido a la ortogonalidad de los frames de medición de actitud y de velocidades angulares. Utiliza un análisis de Lyapunov para asegurar la estabilidad.

Tabla I: Consumo de memoria del algoritmo de fusión de datos

Algoritmo	Consumo de memoria		
	code	ro data	rw data
Madgwick	4892	0	20
Mahony	3160	0	36

Tabla II: Tiempos de ejecución de los algoritmos

Algoritmo	Duración de rutinas	
	Ticks	Tiempo(μ s)
Madgwick	2600	52
Mahony	2500	50

V. RESULTADOS

El consumo de memoria de este algoritmo luego de su compilación, utilizando el compilador *ti-cgt-arm_18.1.3.LTS* se puede ver en la Tabla I.

Mientras que los tiempos de ejecución de ambos algoritmos se pueden encontrar comparados en la Tabla II. Estos tiempos fueron tomados con el procesador ejecutando instrucciones a 50[MHz].

En la aplicación final se decidió utilizar el algoritmo de *Mahony* ya que se adapta mejor a las características deseadas cumpliendo perfectamente con la tarea requerida.

REFERENCIAS

- [1] T. Instruments, *Tiva TM4C1233H6PM Microcontroller Data Sheet*, 2014.
- [2] N. M. E. Association, "Nmea 0183-standard for interfacing marine electronic devices." *NMEA*, 2002.
- [3] E. S. Burgos and L. Giovanini, "Implementación en lenguaje c de un algoritmo para procesamiento de flujos de datos nmea-0183," *Congreso Argentino de Sistemas Embebidos - CASE 2016*, 2016.
- [4] S. T. Microelectronics, *LSM9DS1 Datasheet*, www.st.com, 2014.
- [5] D. Yang, Z. You, B. Li, W. Duan, and B. Yuan, "Complete tri-axis magnetometer calibration with a gyro auxiliary," *Sensors 2017*, vol. 17(6), 2017.
- [6] N. Oceanic, A. Administration *et al.*, *The US/UK World Magnetic Model for 2015-2020*, 2015.
- [7] N. Oceanic and A. A. N. C. for Enviromental Information, "Magnetic declination estimated value," <https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml>.
- [8] S. H. O. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of imu and marg orientation using a gradient descent algorithm," *2011 IEEE International Conference on Rehabilitation Robotics*, 2011.
- [9] R. Mahony, T. Hamel, and J. M. Pfimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on automatic control*, vol. 53(5), 2008.

Dispositivo para Posicionamiento Utilizando GPS y Acelerómetros en Tiempo Real de Bajo Costo

Rodrigo Tolosa, Francisco E. Páez, José M. Urriza
Departamento de Informática - Facultad de Ingeniería
Universidad Nacional de la Patagonia San Juan Bosco
Sede Puerto Madryn
fpaez@unpata.edu.ar, josemurriza@gmail.com

Resumen—Este trabajo presenta un desarrollo que permite determinar la posición y dirección de un objeto, en condiciones y escenarios donde la señal GPS sea intermitente, o este ausente.

Index Terms—GPS, geolocalización, STR

I. INTRODUCCIÓN

El sistema GPS (o alternativos como Glonass) permiten a un receptor pasivo determinar su posición, mediante la triangulación de señales provistas por una red de satélites. Sin embargo, dentro de estructuras, o debajo del agua, puede ser imposible recibir correctamente estas señales, dada la atenuación que sufren. Como alternativa, el uso de acelerómetros y giroscopios permite desarrollar un sistema de posicionamiento en tiempo real.

Este trabajo presenta un desarrollo que determina la posición de un objeto, aún cuando la señal GPS sea intermitente o este ausente, utilizando un microcontrolador y sensores de bajo costo. Se complementa con una aplicación Android para visualizar el posicionamiento del cuerpo y los datos inerciales de los acelerómetros.

Trabajos similares pueden consultarse en [1]–[9].

II. PLATAFORMA DE DESARROLLO

Se utilizó el *kit* NXP FRDM-K64F (previamente Freescale), una plataforma de desarrollo de bajo costo, compatible con la disposición de pines de ArduinoTM R3. Cuenta con un microcontrolador ARM[®] CortexTM-M4 de 120 MHz, 1 MB de memoria flash y 256 kB de RAM, y cuenta con acelerómetro y magnetómetro de 3 ejes, y *slot* para tarjeta micro SD, entre otros periféricos integrados. Como módulo GPS se empleó el Ublox NEO-6M, y para la conectividad Bluetooth[®] un módulo HC-05.

III. IMPLEMENTACIÓN

Se configuró la precisión del acelerómetro interno de la placa en $\pm 8g$ (registro FXOS8700Q_XYZ_DATA_CFG) y la tasa de envío de datos en 200 ms (registro FXOS8700Q_CTRL_REG1). El módulo GPS se configuró para que envíe datos cada 200 ms, utilizando secuencias GPGGA (estándar NMEA¹), que proporcionan latitud, longitud y altitud en una misma trama. Esto requirió

¹<https://www.nmea.org/>

modificar la librería utilizada para la comunicación con el módulo.

La obtención y transmisión de los datos desde los sensores se realiza mediante cuatro tareas, todas con un periodo de 200 ms:

- **Tarea 1: Acelerómetro.** Adquiere los datos del acelerómetro integrado, y los almacena en memoria.
- **Tarea 2: GPS.** Recibe la señal de GPS y almacena los datos en memoria. Si no hay señal GPS, los valores de latitud, longitud y altitud son cero.
- **Tarea 3: Transmisión.** Transmite los datos de posicionamiento obtenidos por las tareas 1 y 2, al dispositivo receptor, mediante Bluetooth.
- **Tarea 4: Almacenamiento.** Realiza un respaldo de los datos generados por las tareas 1 y 2, en una tarjeta micro SD conectada al *slot* de la placa.

Las tareas son planificadas mediante *Rate Monotonic* (RM) [10], un algoritmo de planificación por prioridades fijas. En el mismo, se asigna la mayor prioridad a la tarea de menor período, y así sucesivamente. Cada tarea se caracteriza mediante su periodo ($T_i = 200$) y vencimiento relativo ($D_i = 200$), siguiendo el modelo propuesto en [10].

Como sistema operativo de tiempo real se utilizó FreeRTOS², de código abierto y bajos requerimientos de cómputo y memoria.

Los resultados se presentan de una manera amigable mediante una aplicación Android. Esta recibe los valores de latitud, longitud, altitud, y aceleración en X , Y y Z , mediante Bluetooth. Estos datos son visibles en todo momento, pudiéndose observar la posición como un punto en el mapa o como un recorrido.

IV. POSICIÓN MEDIANTE EL ACELERÓMETRO

La utilización de un sensor inercial para el cálculo de la posición introduce un error acumulativo en la medición, que produce un valor erróneo de la posición real del objeto. Para tratar este problema, se empleó un filtro Kalman [11], [12]. Este es un algoritmo recursivo, implementado en dos fases, una de predicción y otra de corrección.

²<http://www.freertos.org>

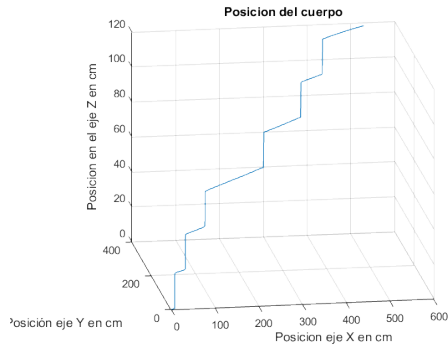


Figura 1. Registro del movimiento del dispositivo al subir una escalera.

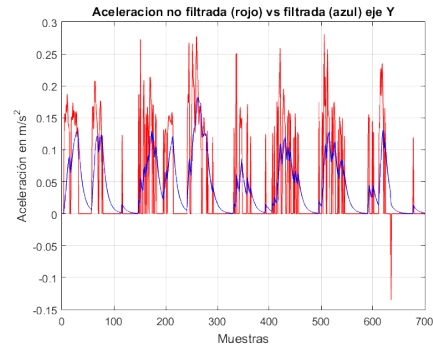


Figura 2. Filtrado de datos del movimiento del dispositivo al subir una escalera.

La fase de predicción es responsable de realizar una estimación a priori del estado del sistema (1) y la covarianza del error (2):

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (1)$$

$$\hat{P}_k^- = AP_{k-1}A^T + Q \quad (2)$$

La fase de corrección computa el nuevo estimado (3), incorporando una nueva medición (z_k) con los valores previos (\hat{x}_k^- y \hat{P}_k^-), y actualiza la covarianza del error (4). Su salida se provee a la fase de predicción, para mantener el ciclo de algoritmo.

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (3)$$

$$P_k = (I - K_k H)P_k^- \quad (4)$$

La aplicación del filtro a los datos obtenidos por el acelerómetro, logró mejorar considerablemente la precisión del cálculo de la posición.

V. RESULTADOS EXPERIMENTALES

Se realizaron múltiples pruebas, en diversos escenarios con y sin señal de GPS. Se presenta el resultado de registrar el movimiento al subir una escalera (Figuras 1 y 2), y la captura de un recorrido realizado en el exterior (Figura 3).

VI. CONCLUSIONES Y TRABAJOS FUTUROS

Se logró obtener la posición del dispositivo combinando datos del acelerómetro y el GPS, visualizando la posición mediante una aplicación Android, en un teléfono *smart*. Los datos del acelerómetro debieron ser filtrados para minimizar los errores producidos por la inexactitud de los mismos.

Como trabajos futuros, se planea realizar la aplicación del filtro Kalman desde el microcontrolador [13], [14], y comprobar su uso en escenarios que requieran registrar posición y sentido del movimiento, y donde la señal GPS sea esporádica o ausente.

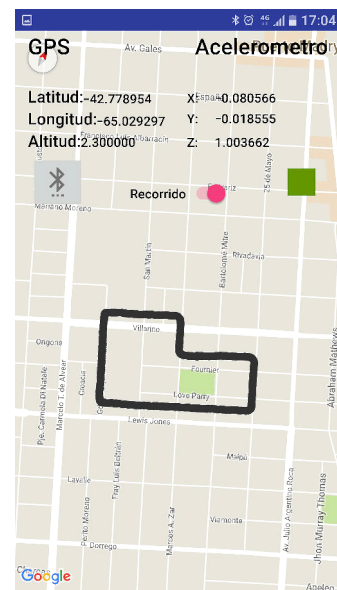


Figura 3. Captura de un recorrido utilizando la aplicación Android.

AGRADECIMIENTOS

Los autores quieren agradecer a los revisores por sus valiosos comentarios y aportes.

REFERENCIAS

- [1] J. Jeon, Y. Kong, Y. Nam, and K. Yim, "An indoor positioning system using bluetooth RSSI with an accelerometer and a barometer on a smartphone," in *10th International Conference on Broadband and Wireless Computing, Communication and Applications, BWCCA 2015, Krakow, Poland, November 4-6, 2015*, L. Barolli, F. Xhafa, M. R. Ogiela, and L. Ogiela, Eds. IEEE Computer Society, 2015, pp. 528–531. [Online]. Available: <https://doi.org/10.1109/BWCCA.2015.142>
- [2] Z. Kowalczyk and T. Merta, "Evaluation of position estimation based on accelerometer data," in *10th International Workshop on Robot Motion and Control, RoMoCo 2015, Poznań, Poland, July 6-8, 2015*. IEEE, 2015, pp. 246–251. [Online]. Available: <https://doi.org/10.1109/RoMoCo.2015.7219743>
- [3] J. Hua, Z. Shen, and S. Zhong, "We can track you if you take the metro: Tracking metro riders using accelerometers on smartphones," *IEEE Trans. Information Forensics and Security*, vol. 12, no. 2, pp. 286–297, 2017. [Online]. Available: <https://doi.org/10.1109/TIFS.2016.2611489>

- [4] R. Hostettler and P. M. Djuric, "Vehicle tracking based on fusion of magnetometer and accelerometer sensor measurements with particle filtering," *IEEE Trans. Vehicular Technology*, vol. 64, no. 11, pp. 4917–4928, 2015. [Online]. Available: <https://doi.org/10.1109/TVT.2014.2382644>
- [5] M. Sousa, A. Techmer, A. Steinhage, C. Lauterbach, and P. Lukowicz, "Human tracking and identification using a sensitive floor and wearable accelerometers," in *2013 IEEE International Conference on Pervasive Computing and Communications, PerCom 2013, San Diego, CA, USA, March 18-22, 2013*. IEEE Computer Society, 2013, pp. 166–171. [Online]. Available: <https://doi.org/10.1109/PerCom.2013.6526728>
- [6] H. Kimm and H. Kimm, "Improved track path method in real time by using gps and accelerometer," in *15th International Conference on Ubiquitous Computing and 8th Communications and International Symposium on Cyberspace Safety and Security, IUCC-CSS 2016, Granada, Spain, December 14-16, 2016*, J. G. Blas, J. Carretero, I. Ray, Q. Jin, and N. Georgalas, Eds. IEEE Computer Society, 2016, pp. 77–84. [Online]. Available: <https://doi.org/10.1109/IUCC-CSS.2016.019>
- [7] A. Serra, D. Carboni, and V. Marotto, "Indoor pedestrian navigation system using a modern smartphone," in *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services, ser. MobileHCI '10*. New York, NY, USA: ACM, 2010, pp. 397–398. [Online]. Available: <http://doi.acm.org/10.1145/1851600.1851683>
- [8] W. Chen, Z. Fu, R. Chen, Y. Chen, O. Andrei, T. Kroger, and J. Wang, "An integrated gps and multi-sensor pedestrian positioning system for 3d urban navigation," *2009 Joint Urban Remote Sensing Event*, pp. 1–6, 2009.
- [9] C.-W. Tan and S. Park, "Design of accelerometer-based inertial navigation systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, pp. 2520–2530, 2005.
- [10] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *J. ACM*, vol. 20, no. 1, pp. 46–61, Jan. 1973. [Online]. Available: <http://doi.acm.org/10.1145/321738.321743>
- [11] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1960. [Online]. Available: <http://dx.doi.org/10.1115/1.3662552>
- [12] M. Stakkeland, G. Prytz, W. E. Booij, and S. T. Pedersen, "Characterization of accelerometers using nonlinear kalman filters and position feedback," *IEEE Trans. Instrumentation and Measurement*, vol. 56, no. 6, pp. 2698–2704, 2007. [Online]. Available: <https://doi.org/10.1109/TIM.2007.908145>
- [13] A. Valade, P. Acco, P. Grabolosa, and J. Fourniols, "A Study about Kalman Filters Applied to Embedded Sensors," *Sensors*, vol. 17, no. 12, p. 2810, 2017. [Online]. Available: <https://doi.org/10.3390/s17122810>
- [14] G. Herman, A. Milshteyn, A. Lin, M. Garcia, C. Liu, D. Guillaume, K. Rad, , and H. Boussalis, "Implementation of Kalman Filter on PSoC-5 Microcontroller for Mobile Robot Localization," *Journal of Communication and Computer*, pp. 469–477, 2014.

RESUMEN

Aplicando LSP en planificador para sistemas dinámicamente reconfigurables.

Alejandro Nuñez Manquez
Departamento de Electrónica
Universidad Nacional de San Luis
San Luis, Argentina
janyo12@gmail.com

Julio Dondo Gazzano
Departamento de Electrónica
Universidad Nacional de San Luis
San Luis, Argentina
jdondo@gmail.com

Mario Berón
Departamento de Informática
Universidad Nacional de San Luis
San Luis, Argentina
mmberon@gmail.com

Carlos F. Sosa Paez
Departamento de Electrónica
Universidad Nacional de San Luis
San Luis, Argentina
cfsp@gmail.com

Martín Murdocca
Departamento de Electrónica
Universidad Nacional de San Luis
San Luis, Argentina
mmurdocc@gmail.com

Resumen—Las FPGA dinámicamente reconfigurables permiten modificar parte de su diseño sin detener el mismo de acuerdo a la necesidad de procesamiento. Para resolver el problema de planificación de reconfiguraciones posibles este trabajo propone, usando el método multicriterio LSP (Logic Scoring of Preference - Puntuación Lógica de Preferencias), una función que permita decidir sobre la ubicación óptima de un objeto hardware en un sistema con áreas dinámicamente reconfigurables.

Index Terms—LSP, FPGA, reconfiguración dinámica

I. INTRODUCCIÓN

El uso de la reconfiguración parcial dinámica [1] permite modificar un diseño basado en FPGAs en tiempo de ejecución, donde diferentes funcionalidades hardware (OH) comparten los recursos lógicos de áreas dinámicamente reconfigurables (ADR) [2]. Uno de los problemas aún sin resolver en este aspecto es la planificación de las distintas reconfiguraciones que se pueden realizar, teniendo en cuenta aspectos tales como prioridad de ejecución de los objetos hardware, áreas disponibles, recursos, etc. En este sentido, dado un Sistema Dinámicamente Reconfigurable (SDR) formado por varias ADR y varios OH, se propone el diseño de un algoritmo que permite decidir que OH se instancia en que ADR, aplicando el método multicriterio LSP.

II. MÉTODO MULTICRITERIO LSP

LSP es un método multicriterio cuantitativo para la creación de funciones de criterio y para sus aplicaciones en evaluación, optimización, comparación y selección de sistemas complejos en general [3].

LSP plantea los siguientes pasos generales:

1. Construir un Árbol de Criterios correspondiente a los elementos que serán evaluados.
2. Definir una función denominada Criterio Elemental, que normaliza las contribuciones por cada uno de los atributos del árbol de criterios.

3. Construir una Estructura de Agregación, que permitirá computar las contribuciones de los criterios en un único valor. Dicho valor indicará el grado de satisfacción general del elemento evaluado.
4. Llevar a cabo la evaluación y posterior selección de los elementos evaluados.

III. IMPLEMENTACIÓN DE LSP

En el caso de nuestro SDR se definen lo siguiente criterios:

III-A. Árbol de Criterios

El árbol de criterios se define a través de las ADR y de los objetos que las pueden ocupar.

III-A1. Características de las ADR: Las ADR poseen las siguientes características:

- Objetos que puede alojar (definidos en tiempo de diseño)
- Si está ocupada o no.
- Prioridad de ejecución del objeto que la ocupa.

III-B. Características de los Objetos Hardware

Los objetos hardware poseen las siguientes características:

- ADRs en donde puede alojarse.
- Prioridad de ejecución.

Su ubicación depende de qué ADR los puede alojar y las prioridades que poseen tanto el objeto alojado en la ADR y la prioridad del objeto que desea ocuparla.

III-C. Criterios Elementales

- **ADRx puede alojar el OH:** Toma valor 0 cuando no ó 100 cuando sí se puede alojar en la ADRx. Este es un parámetro esencial.
- **ADR ocupada:** 50 si ya hay un objeto instanciado en ella y 100 si está libre. Este es un parámetro deseable.
- **Diferencia de prioridad entre el objeto a alojar y el objeto ya alojado en la ADRx:** Cada OH posee un valor

de prioridad: baja(0), regular (25), media (50), alta(80), muy alta (100).

$$dif_p = pOHn - pOHv \quad (1)$$

donde **pOHn** es la prioridad del OH que se desea alojar y **pOHv** es la prioridad del OH ya alojado. Para valores negativos $diff_p = 0$. Este es un parámetro esencial.

III-D. Estructura de agregación

La estructura de agregación definida para obtener la preferencia global se muestra en la figura 1.

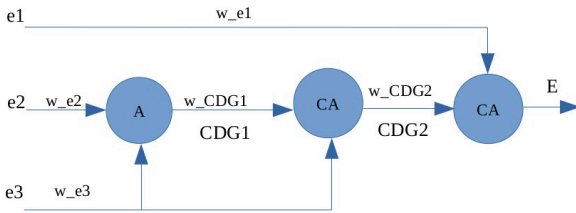


Figura 1. Estructura de agregación.

donde

- e1: ADRx puede alojar el OH
- e2: ADR ocupada
- e3: Diferencia de prioridad entre el objeto a alojar y el objeto ya alojado en la ADRx.
- CDGx: son funciones intermedias.
- E: es la salida de la función.
- wx: pesos de preferencias o funciones intermedias.

La función LSP que surge es la siguiente:

$$E = (w_{e1} * e_1^{r_{CA}} + w_{CDG2} * CDG_2^{r_{CA}}) \frac{1}{r_{CA}} \quad (2)$$

donde

$$CDG_2 = (w_{CDG1} * CDG_1^{r_{CA}} + w_{e3} * e_3^{r_{CA}}) \frac{1}{r_{CA}} \quad (3)$$

$$CDG_1 = (w_{e2} * e_2^{r_A} + w_{e3} * e_3^{r_A}) \frac{1}{r_A} \quad (4)$$

La selección de los pesos **wx** se hace en función de la incidencia de un criterio x con respecto a otro siendo ambos entradas de una misma función (por ejemplo la función CA). En este caso la condición es que la suma de todos los pesos de los criterios que llegan a una función sea igual a 1.

Los valores r_A y r_{CA} se seleccionan de acuerdo a la figura 2.

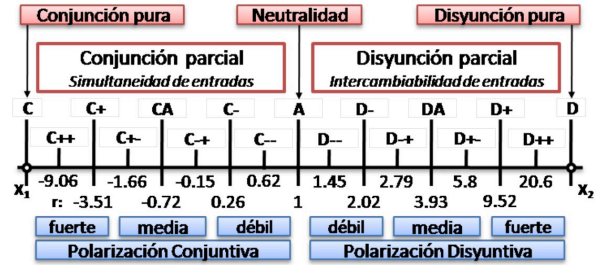


Figura 2. Funciones de Conjunción-Disjunción Generalizadas: Grados de Conjunción-Disjunción para dos operandos (e_1 y e_2) [4].

TABLA I
OBJETOS HARDWARE Y ADRS QUE LOS PUEDEN ALOJAR

Objetos Hardware	Áreas Dinámicamente Reconfigurables			
	Área 1	Área 2	Área 3	Área 4
Objeto 1	*	*		
Objeto 2	*		*	
Objeto 3	*			*
Objeto 4		*	*	
Objeto 5			*	*
Objeto 6		*		*

IV. RESULTADOS

El sistema propuesto para las pruebas posee 4 ADR y una biblioteca de 6 OH. Cada ADR puede alojar 3 diferentes OH, como se puede observar en la Tabla I

Se definieron pesos $w_x = 0.5$ y se usaron tres escenarios. Los resultados fueron los siguientes.

- La ADR **NO** puede alojar al OH ($e_1 = 0$); La ADR está libre ($e_2 = 100$); la diferencia entre prioridades de OH es 80.
 $E = 0$
- La ADR **SÍ** puede alojar al OH ($e_1 = 100$); La ADR está libre ($e_2 = 100$); la diferencia entre prioridades de OH es 80 (al estar libre la ADR la prioridad del objeto que la ocupa vale 0)
 $E = 91.831$
- La ADR **SÍ** puede alojar al OH ($e_1 = 100$); La ADR está ocupada por un objeto de mayor prioridad, ($e_2 = 50$); la diferencia entre prioridades de OH es 0.
 $E = 0$

El sistema fue probado utilizando una placa Zedboard de Xilinx y se utilizó el software dynDeT [5] desarrollado por la Universidad de Castilla-La Mancha, para el diseño del SDA. En este caso se definieron 4 áreas y seis OH. Las reconfiguraciones se realizaron utilizando diferentes configuraciones iniciales donde aplicando el algoritmo desarrollado se configuró el sistema. El algoritmo fue ejecutado en el procesador ARM de la FPGA. El programa desarrollado ejecuta el algoritmo y de acuerdo al resultado toma una de los bitstreams parciales almacenados en memoria. El tiempo transcurrido en el proceso de decisión fue de unos pocos milisegundos.

REFERENCIAS

- [1] Xilinx.com. (2019). Xilinx - Adaptable. Intelligent.. [online] Available at: <https://www.xilinx.com/> [Accessed 21 May 2019].
- [2] J. Dondo, J. Barba, F. Rincón, F. Moya, J. C. López, Dynamic objects: Supporting fast and easy run-time reconfiguration in FPGAs, *Journal of Systems Architecture*, Volume 59, Issue 1, 2013, ISSN 1383-7621.
- [3] A. Dasso, A. Funes, M. Peralta, C. Salgado. Una herramienta para la evaluación de sistemas. III Workshop de Investigadores en Ciencias de la Computación. 2001
- [4] J.J. Dujmovic. A Method for Evaluation and Selection of Complex Hardware and Software Systems. The 22nd Int'l Conference for the Resource Management and Performance Evaluation of Enterprise CS. CMG 96 Proceedings, 1:368–378, 1996.
- [5] Available at <https://github.com/juliancaba/dynDeT>

Clasificación de señales acústicas subacuáticas mediante el empleo de redes neuronales

Cebedio, María Celeste. Petruzzi, David Mario.
Lab. De Comunicaciones. ICYTE. Facultad de Ingeniería. UNMdP.
celestecebedio@fi.mdp.edu.ar. petruzzi@fi.mdp.edu.ar.

En la Argentina el mar se extiende en una superficie equivalente a casi el 50% de su territorio continental, lo que refleja el importante potencial, tanto para su conservación biológica, como para su aprovechamiento económico. Por este motivo, es imprescindible desarrollar investigaciones relacionadas con los recursos disponibles en el mar, costo de explotación, regulación, seguimiento y preservación [1].

En el territorio argentino, con la excepción de algunas localidades cuidadosamente estudiadas, el número de especies registradas para diferentes áreas de la costa se estima que es solo una pequeña fracción del número real de especies presentes [2]. Por lo tanto, el reconocimiento de las diferentes especies marinas y el monitoreo en tiempo real puede colaborar con la realización de estadísticas para estudios de población y preservación de especies. Diversos estudios han demostrado que la adquisición y estudio de datos sonoros marinos permiten un mayor entendimiento del medio. Algunos ejemplos incluyen la identificación de señales biológicas [3] y antropogénicas [4,5], las identificaciones vocales [6,7] y el monitoreo de las respuestas de la fauna marina al ruido antropogénico [8, 9].

A pesar de la necesidad de monitoreo automático del territorio marino, las herramientas existentes basadas en la acústica no están adaptadas para manejar las enormes cantidades de datos adquiridos [10]. Sin embargo, se han publicado estudios sobre la clasificación automática de datos acústicos más generales, basados en el uso de redes neuronales. En este sentido, se pueden citar diversos estudios que requieren reconocimiento y clasificación de patrones [11, 12], ya sea sobre sonidos ambientales, el habla, la música, cantos de pájaro, etc. Actualmente algunos pocos estudios sobre clasificación de peces son muy prometedores [10, 13, 14,15].

En este trabajo se pretende aplicar tecnologías de redes neuronales artificiales para clasificar sonidos que puedan estar presentes en los mares y océanos. Se propone la realización de un clasificador en tiempo real. Un diagrama en bloques del sistema propuesto se muestra en la figura 1.

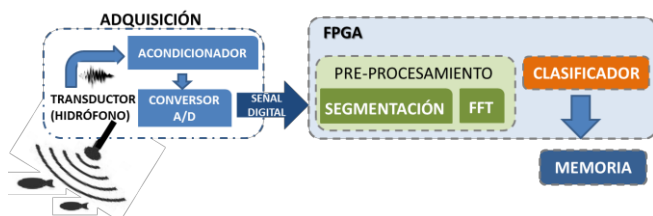


Figura 1. Diagrama en bloque del sistema de clasificación.

Etapa de adquisición. Es la encargada de levantar las muestras del medio y transformarla en un dato digital.

Pre-procesamiento. Los datos digitalizados se deben adecuar para la etapa posterior. Se necesita realizar una toma de muestras fija y normalizarlas (segmentación). Debido a los requerimientos del hardware en el que se implementa el clasificador es necesario obtener la misma

información, pero con menor cantidad de muestras. Para ello se obtiene el espectro de cada segmento adquirido. Esta etapa se implementa en una FPGA DE0-Nano de la empresa ALTERA.

Clasificador. La etapa de clasificación requiere una primera fase de diseño off-line y luego la fase de implementación.

a) **Fase de Diseño.** Para realizar la clasificación es necesario construir una red neuronal. Esta etapa se realiza netamente en software y utilizando lenguaje de programación de código abierto (Python).

- **Dataset.** La construcción de una red neuronal necesita de una gran cantidad de datos reales. Estos archivos de audio deben ser segmentados, normalizados y, posteriormente realizar una transformación espectral de los mismos. Con todos los datos pre-procesados, se debe formar una matriz numérica de una extensión fija. Debido a la enorme cantidad de datos que se manejan, se desarrolla un programa para automatizar al máximo este proceso. Del mismo modo, siguiendo el paradigma de código abierto, enfoque Open Science [16,17] y open Data [18], los datasets creados (con sus características bien definidas) y los programas involucrados serán de dominio público y estarán disponibles en un repositorio gratuito [19].

- **Construcción de la red neuronal.** Idealmente, un clasificador debería ser entrenado para que cuando se le presente una versión ligeramente distorsionada del patrón, pueda ser reconocida correctamente. En otras palabras, la red debería poder recuperar una señal "limpia", aún en medios ruidosos. Para poder reconocer y clasificar las señales acústicas inmersas en el océano, con todas las problemáticas que conlleva trabajar en este medio, se analizan las respuestas de dos tipos de redes neuronales muy útiles para el reconocimiento de patrones en medios ruidosos: la Red de Kohonen [20,21] y la Red Convolutiva 1D [22].

b) **Fase de Implementación.** La implementación pretende mapear la red neuronal en hardware digital. Para realizar esto en tiempo real se implementará sobre la misma FPGA que la etapa de pre-procesamiento. Dado que las redes neuronales están constituidas por decenas de neuronas y la descripción de esa arquitectura mediante hardware se expresa en una gran cantidad de recursos de la FPGA, es necesario considerar la red a implementar en términos de resultados y requerimientos de hardware para su implementación [23].

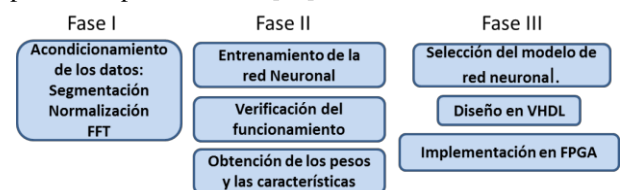


Figura 2. Fases del diseño e implementación del clasificador.

Almacenamiento en memoria. Se requiere de una memoria que guarde los datos para su posterior análisis.

REFERENCIAS

- [1] Gustavo Baruj-Sergio Drucaroff. “Estimaciones del potencial económico del océano en la Argentina”. 1a ed. Ciudad Autónoma de Buenos Aires: ciecti, 2018.
- [2] Bigatti G, Signorelli J, Marine invertebrate biodiversity from de Argentine Sea, South Western Atlantic. *ZooKeys* 791: 47-70. 2018
- [3] Gavrilov, A.N., McCauley, R.D. “Acoustic detection and longterm monitoring of pygmy blue whales over the continental slope in southwest Australia”. *J. Acoust. Soc. Am.* **134**(3), 2505–2513 (2013)
- [4] Duncan, A.J., McCauley, R.D. “Characterisation of an air-gun as a sound source for acoustic propagation studies”. In: UDT Pacific 2000 conference, Sydney, Australia, 7–9 Feb (2000)
- [5] Duncan, A.J., Gavrilov, A.N., Hu, T.: Using offshore seismic surveys as acoustic sources of opportunity for geoacoustic inversion. In: Proceedings of the Europe an Conference on Underwater Acoustics, Paris, June 29–July 4, pp. 255–260 (2008)
- [6] Ward, R., Parnum, I., Erbe, C., Salgado Kent, C.: Whistle characteristics of Indo-Pacific bottlenose dolphins (*Tursiops aduncus*) in Fremantle Inner Harbour, Western Australia (2016). doi:10.1007/s40857-015-0041-4
- [7] Recalde-Salas, A., Salgado Kent, C.P., Parsons, M.J.G., Marley, S.A., McCauley, R.D.: Non-song vocalizations of pygmy blue whales in Geographe Bay, Western Australia. *J. Acoust. Soc. Am.* **135**(5), EL213–EL218 (2014)
- [8] Dunlop, R.A., Noad, M.J., McCauley, R.D., Kniest, E., Paton, D., Cato, D.H.: The behavioural response of humpback whales (*Megaptera novaeangliae*) to a 20 cubic inch air gun. *Aqua. Mam.* **41**(4), 412–433 (2015). doi:10.1578/AM.41.4.2015.412
- [9] Paiva, E.G., SalgadoKent,C., Gagnon, M., McCauley, R., Finn, H.: Reduced detection of Indo-Pacific bottlenose dolphins (*Tursiops aduncus*) in an InnerHarbour channel during pile driving activities. *Aquat. Mamm.* **41**(4), 455–468 (2016)
- [10] Marielle Malfante, Jérôme I. Mars, Mauro Dalla Mura, and Cédric Gervaise. Automatic fish sounds classification. Article in The Journal of the Acoustical Society of America. May 2018
- [11] Duda, R., Hart, P., Stork, D., “pattern Classification. A Wiley – Inter science Publication, 2nd edition, USA. 655 pp. 2001
- [12] Challoner, Jack. “Artificial Intelligence”. Dorling Kindersley Limited, 80 Strand, London WC2R ORL. 72 pp. 2002
- [13] Noda, J. J., Travieso, C. M., and Sanchez-Rodriguez, D. “Automatic taxonomic classification of fish based on their acoustic signals,” *Appl. Sci.* 6(12), 443. (2016).
- [14] Vieira, M., Fonseca, P. J., Amorim, M. C. P., and Teixeira, C. J. “Call recognition and individual identification of fish vocalizations based on automatic speech recognition: An example with the Lusitanian toad fish,” *J. Acoust. Soc. Am.* 138(6), 3941–3950. (2015).
- [15] Sattar, F., Cullis-Suzuki, S., and Jin, F. “Acoustic analysis of big ocean data to monitor fish sounds,” *Ecol. Inform.* 34, 102–107. (2016).
- [16] M. Neilsen, "Doing science in the open", *Physics World*, vol. 22, no. 5, pp. 32, 2009.
- [17] A. Grand, K. Bultitude, C. Wilkinson, A.F. Winfield, "Muddying the waters or clearing the stream? Open Science as a communication medium", *Public Communication of Science and Technology New Delhi India*, pp. 6-9, December 2010.
- [18] Peter Murray-Rust. “Open Data in Science”. *Serials Review*, vol 34:1, 52-64, DOI: 10.1080/00987913.2008.10765152. 2008
- [19] <http://github.com/cebedio/dataset-audios-marinos>
- [20] Merlo, G & Fernández, V & Caram, L & Priegue, R & Martínez, R. (2019). “ Reconocimiento de la voz mediante una red neuronal de Kohonen”. CACIT 1997.
- [21] Luna-Ortega, Carlos & Mora-González, Miguel & Alvarez-Medina, Marco & Martíñez Romo, Julio César & J Luna Rosas, Francisco & E Gómez-Del Villar, Víctor. (2009). “Analysis of Kohonen's Neural Network with application to speech recognition”
- [22] Sajjad Abdoli I , Patrick Cardinal, Alessandro Lameiras Koerich. “End-to-End Environmental Sound Classification using a 1D Convolutional Neural Network”. Department of Software and IT Engineering, Ecole de Technologie Supérieure, Université du Québec, H3C 1K, Montreal, QC, Canada. Abril 2019. arXiv:1904.08990v1
- [23] Polanco Carrillo, Franky; Alvarez Gerónimo, Daniel y Moreno Vega, Valery. “Implementación mediante hardware de una Red Neuronal Artificial para Reconocimiento de Caracteres”. *EAC* [online]. 2015, vol.36, pp.95-106. ISSN 1815-5928.

Impresora basada en Interfaz Cerebro-Computadora

G. Castro, S. Mac Mullen, J. M. Geria, N. González

Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, Buenos Aires, Argentina

Contacto: gcastro@est.frba.utn.edu.ar

Resumen—Los productos de apoyo permiten disminuir las barreras que enfrentan las personas con discapacidad y adultos mayores y así favorecer su accesibilidad, comunicación y autonomía. En el presente trabajo se propuso como objetivo el diseño y evaluación de una interfaz cerebro-computadora aplicada a la estimulación de personas con movilidad reducida mediante una actividad lúdica como el dibujo. Se desarrolló una aplicación de escritorio multiplataforma (QT) capaz de vincular los potenciales evocados visuales del usuario (EEG) con un sistema embebido capaz de sostener un lápiz y dibujar a partir de las acciones recibidas. En esta primera instancia, se evaluaron sujetos con y sin entrenamiento previo pero sin discapacidad asociada. Se observó un mayor rendimiento en aquellos donde primaron los gestos faciales combinados con su concentración en comparación con el proceso de simple concentración y evocación.

Palabras Clave—BCI, Bluetooth, EEG, Estimulación.

I. INTRODUCCIÓN

Los productos de apoyo son aquellos dispositivos que permiten disminuir las barreras a las que se enfrenta una persona. Estos posibilitan una innegable mejora en la calidad de vida de las personas con “diversidad funcional” al brindar una real equiparación de oportunidades a través de su uso, posibilitando la inserción social y laboral. En los últimos años, se ha explorado la posibilidad de establecer un canal de comunicación entre el cerebro y el mundo exterior mediante la interpretación de las señales del cerebro (interfaz cerebro computadora - BCI). En este trabajo se propone el uso de un casco para EEG, pudiendo ser utilizado por personas sin grandes conocimientos técnicos, con el fin de que el usuario pueda dibujar a través de un sistema embebido (impresora en dos dimensiones) [3].

II. PRINCIPIO DE FUNCIONAMIENTO

El proyecto está orientado al control de una impresora 2D a partir de la utilización de potenciales evocados visuales. Para lograrlo, la señal atraviesa diferentes etapas como se ve en la figura (1).

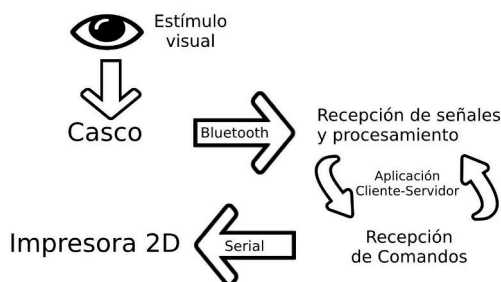


Figura 1: Diagrama en bloques

II-A. Flujo de datos

El proceso se inicia con la presentación de un estímulo visual mediante una aplicación desarrollada en Qt, la cual muestra una representación gráfica de los movimientos de una impresora. Se realiza un barrido automático destacando cada opción en forma secuencial de las alternativas presentadas para que el usuario pueda concentrarse fácilmente en la deseada.

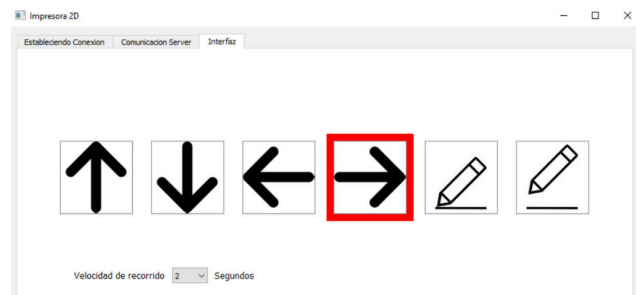


Figura 2: Aplicación Qt: Barrido Luminoso

II-B. Recepción de señales y procesamiento

Se busca asociar la imagen presentada con un determinado momento de concentración por parte del usuario.

Se utilizan las bibliotecas provistas por el casco EEG (Emotiv), las cuales permiten distinguir si la acción realizada por el usuario corresponde o no a una acción previamente entrenada.

II-C. Recepción y ejecución de comandos

La aplicación compara la respuesta recibida con la opción de movimiento ofrecida en ese momento y, de coincidir con una respuesta voluntaria, detiene el barrido en la interfaz visual destacando la opción elegida. Dicha opción se traduce a un comando asociado que será enviado por una interfaz serie hacia la impresora, la cual ya tiene estos comandos incorporados en su firmware. El proceso finaliza con la ejecución del comando por parte de la impresora.

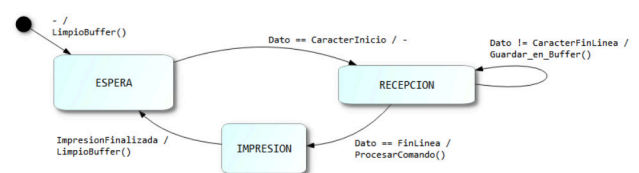


Figura 3: Diagrama en bloques

III. MATERIALES Y MÉTODOS

III-A. Registros de Potenciales Evocados Relacionados

Como interfaz de adquisición de la señal EEG se utilizó un casco modelo EPOC, de la marca EMOTIV. El entrenamiento consiste en solicitarle a la persona que se concentre en una situación específica durante el tiempo que se registra la señal de electroencefalografía.

Por medio de comunicación bluetooth, el casco EMOTIV enviará las señales a ser procesadas al cliente. Finalizada la etapa de procesamiento, la aplicación podrá discernir si se trata de una acción válida y poder así enviar la respuesta al servidor.

III-B. Accionamiento del dispositivo diseñado

Al iniciar la aplicación, se solicitará una conexión por bluetooth al casco así como también la configuración del perfil del usuario que lo utilice. Acto seguido se solicitará una conexión física al sistema embebido (impresora). A partir de este momento podemos pasar a la interfaz visual para empezar con la comunicación con la impresora.

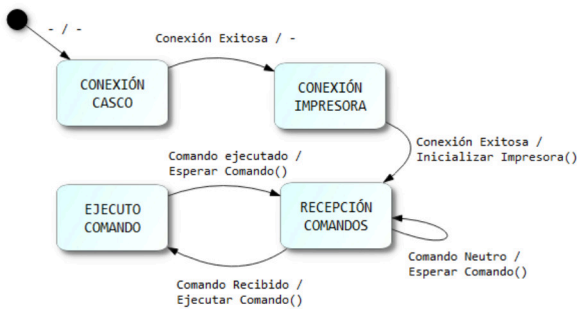


Figura 4: Diagrama de estados de la aplicación.

Una vez interpretada la respuesta en el servidor, este lo compara con la imagen en la que el usuario se concentró y detiene el barrido en dicha imagen para poder mantener la concentración. Cuando esto sucede, la aplicación envía un comando asociado al movimiento por puerto serie para ser ejecutada por la impresora.

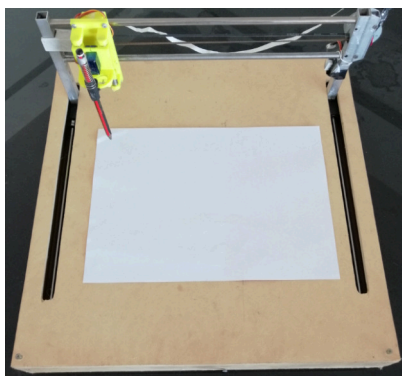


Figura 5: Impresora diseñada.

Esta cuenta con dos motores paso a paso para mover el cabezal sobre el plano de dibujo, y un servo motor para

levantar o bajar el mismo. Como microcontrolador para el firmware se utilizó un LPC1769 de NXP que cuenta con un ARM Cortex M3.

IV. RESULTADOS

Las pruebas para determinar la eficacia del proyecto se realizaron en un ambiente controlado dentro de la universidad, donde se sometieron distintos individuos a distintas pruebas.

El objetivo consiste en determinar el porcentaje de aciertos en las detecciones de las intenciones del individuo. Aquí discernimos en 3 posibles casos considerados:

- Detectado: Si el sistema responde dentro del tiempo de barrido al comando deseado.
- Falso Positivo: Si el sistema realiza una detección de un comando aleatorio en un estado de relajación del usuario.
- No Detectado: Si el sistema no responde dentro del tiempo de barrido al comando deseado.

Adicional a enviar un comando sólo con la concentración del usuario, se hicieron pruebas ayudando con algún gesto facial, como parpadeos o un simple movimiento de cejas.

Cuadro I Resultado de Pruebas Relativo

Método	Usuario	Porcentaje de aciertos
Parpadeo	Usuario 1	66.67 %
Parpadeo	Usuario 3	96.43 %
Parpadeo	Usuario 4	70.59 %
Movimiento de Ceja	Usuario 2	77.50 %
Concentración	Usuario 3	20.41 %
Concentración	Usuario 4	34.48 %

V. CONCLUSIONES

A partir de los resultados obtenidos, se observó que existen muchos factores que alteran el porcentaje de aciertos, entre los que destacamos el número de entrenamientos del usuario, el tiempo de barrido de la interfaz visual y el método de concentración elegido.

Se notó como en usuarios menos entrenados hacía falta un tiempo adicional para que pudieran concretar la concentración, esto ocasionaba que el comando ejecutado fuera el siguiente al deseado en el barrido. Esto fue considerado un error en la detección, aunque podría atribuirse a la falta de entrenamiento del usuario y/o un tiempo de barrido más rápido que el requerido.

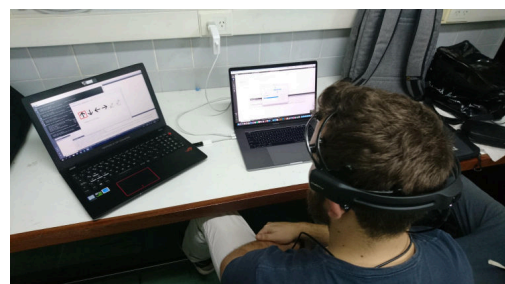


Figura 6: Adquisición de señal de EEG.

REFERENCIAS

- [1] Alvaro Morán García. “Diseño de interfaces cerebro-máquina controlados mediante registros de EEG”. Grupo de Neurocomputación biológica. Escuela Politécnica Superior Universidad Autónoma de Madrid, 2015.
- [2] F.Pose, N.González, M.Araujo, A. Ortiz, B. Niro, J. Balbi, L. Ramos. “Desarrollo de algoritmo de detección de eventos para comandar un teclado en pantalla utilizando una interfaz cerebro-computadora”. Co-naaisi. Argentina, 2017.
- [3] Claudia Nureibis Henríquez Muñoz. “Estudio de técnicas de análisis y clasificación de señales EEG en el contexto de sistemas BCI (Brain Computer Interface)”. Universidad Autónoma de Madrid. Escuela Politécnica Superior - Departamento de Ingeniería Informática, 2014.
- [4] Jeong-Hwan Lim, Han-Jeong Hwang, Chang-Hwan Im, “Eyes-closed SSVEP-based BCI for binary communication of individuals with impaired oculomotor function”, Hanyang University. International winter workshop on BCI 2013.
- [5] Nannan Zhang, Yadong Liu, Zongtan Zhou, “A SSVEP-BCI with random moving stimuli in simulation environment”, National University of Defense Technology. 2017. DOI 10.1109/IWW-BCI.2017.7858170.

Comparación de resultados entre el uso de Nexys-2 y Pynq-Z1 para controlar una adquisidora de datos

Di Fiore, Tomás; Lacomí, Héctor; Cristian Donato
División Radar Láser, Departamento Láser – CITEDEF
Grupo SyCE-FRH, UTN-FRH
Prov. de Buenos Aires

Urbano Pintos, Nicolás; Lavorato, Mario Blas
División Radar Láser, Departamento Láser – CITEDEF
Grupo Tama, UTN-FRH
Prov. de Buenos Aires

El objetivo de este trabajo es presentar los resultados obtenidos con las mejoras realizadas en el funcionamiento, implementación y verificación de una placa de adquisición de datos construida ad-hoc [1]. La placa de adquisición se encontraba operando con una placa Nexys-2 basado en el chip Spartan-3 [2] y para verificar su funcionamiento se implementó la lógica de control en lenguaje Verilog. Actualmente se utiliza para el funcionamiento de la adquisidora de datos, el kit de desarrollo PYNQ-Z1 [3] basado en el chip ZYNQ-7000 [4], que permite programar la lógica de control de los módulos Verilog en lenguaje Python. Como esta unidad adquisidora será utilizada en un sistema de adquisición de datos multipropósito, el nuevo sistema fue testeado en el laboratorio simulando distintos tipos de señales inyectándole diferentes niveles de ruido eléctrico (del tipo Gaussiano), simulando las señales reales que deberá adquirir.

La placa adquisidora utilizada en ambas experiencias posee un conversor analógico digital ADC10080 [5], el cual tiene una frecuencia máxima de muestreo de 80MS/s y 10 bits de resolución.

En los ensayos realizados con la placa Nexys 2 inyectando señales senoidales se obtenía como resultado un offset de aproximadamente 30mV, el cual se vuelve a observar con la PYNQ-Z1.

A diferencia del sistema anterior que utilizaba los 8 bits más significativos a modo de filtrado binario, con esta placa se utilizan los 10 bits que proporciona el ADC ya que por los resultados obtenidos la señal se visualiza con un ruido mucho menor que no perjudica a la lectura de la señal.

La frecuencia de muestreo es mayor a la utilizada con la Nexys 2, alcanzando 76.9MHz y obteniendo un ancho de banda de 14MHz, suficiente para detectar la información de interés en la señal de entrada. El ancho de banda fue calculado inyectando una señal rectangular de frecuencia 5KHz y una amplitud de 2Vpp de manera tal de adquirir el tiempo de conmutación entre estados, y luego aplicar una transformada de Fourier para observar el espectro. Además se hizo un barrido en frecuencia con señales senoidales y rectangulares para verificar los mismos datos.

El hecho de que la placa de desarrollo PYNQ-Z1 posea un chip en el cual se encuentran embebidos un microprocesador Cortex A9 y una FPGA Artix 7 permitió, a diferencia del sistema anterior, controlar, configurar y visualizar los datos a

través del microprocesador debido a la existencia de un bus de datos basado en el protocolo AXI, que comunica el mismo con la memoria RAM y los módulos de control desarrollados en verilog para FPGA. Esto lo convierte en un sistema compacto sin necesidad de contar con una PC para procesar los datos.

La cantidad de datos de almacenamiento se determinó en 16Kbytes que representa un tiempo de aproximadamente 213us por adquisición.

Para verificar el funcionamiento del sistema adquisidor se realizó la simulación de la señal senoidal sintetizada por un generador de funciones, de 1Vpp y frecuencia de 5MHz (Figura 1). Con el resultado de cada adquisición se representaron las muestras adquiridas a través del procesamiento de datos con Python desde el microprocesador.

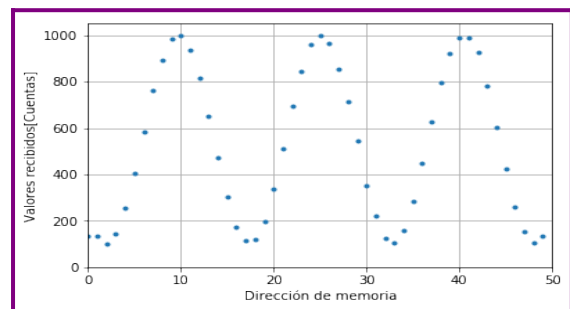


Figura 1: Adquisición de una señal senoidal de 1Vpp y 5MHz.

Conclusiones:

Se obtuvo una distorsión en la señal mucho menor al anterior debido al aumento de la frecuencia de muestreo, aumentando la relación señal a ruido.

A partir del procedimiento de este ensayo fue posible extender las funcionalidades del circuito realizado en la FPGA, habiendo agregado una señal de 'trigger', y teniendo la capacidad de muestrear a 76,9MHz, teniendo un ancho de banda mayor que el anterior.

El nuevo sistema propuesto donde se reemplaza la Nexys2 por la Pynq-Z1 como se puede observar, presenta resultados prometedores en cuanto a la calidad de la señal adquirida y mejoras notables en cuanto la relación Señal/Ruido y distorsión de las señales (Entrada vs. Salida).

REFERENCIAS

- [1] Implementación de una adquisidora de datos basada en el ADC10080 y controlada por FPGA - SASE 2018
- [2] <https://reference.digilentinc.com/reference/programmable-logic/nexys-2/start> - Nexys 2
- [3] <https://store.digilentinc.com/pynq-z1-python-productivity-for-zynq-7000-arm-fpga-soc/> - Python Productivity for Zynq
- [4] Zynq-7000 SoC - Technical Reference Manual - UG585 (v1.12.2) July 1, 2018 Xilinx
- [5] <http://www.ti.com/lit/ds/symlink/adc10080.pdf> - Conversor A/D Symlink - ADC10080.

Determinación de eficiencia en la ejecución de algoritmos de procesamiento de imágenes con múltiples procesadores en FPGA

Jorge R. Osio, Walter Aróztegui, Antonio A. Quijano,
José Rapallini
CeTAD – Depto. de Electrotecnia
Facultad de ingeniería - UNLP
La Plata, Argentina
jorge.osio@ing.unlp.edu.ar

Jorge R. Osio
Programa de Tecnologías de la información y la
comunicación (TIC) en aplicaciones de interés social
Instituto de Ingeniería - UNAJ
Florencio Varela, Argentina
josio@unaj.edu.ar

Resumen

En este trabajo se desarrolló un sistema para determinar la eficiencia en la ejecución de algoritmos de procesamiento digital de imágenes sobre dispositivos FPGAs, aprovechando las bondades que proveen en cuanto al cómputo paralelo.

Para el procesamiento, los algoritmos se aplicaron a imágenes médicas, ya que las necesidades de desempeño para el análisis en tiempo real de estas imágenes requieren plataformas de sistemas escalables tanto en software (CPUs) como en hardware.

La metodología desarrollada consiste en la implementación y ejecución de algoritmos de procesamiento de imágenes en un simple procesador y en dos procesadores para la determinación de la eficiencia en cuanto al tiempo de ejecución de los mismos. El método se aplica a los algoritmos de procesamiento de imágenes de tipo espacial basados en operadores de ventana; tales como el filtro de media, mediana, erosión, dilatación y la convolución espacial [1-3].

Implementación

El sistema se diseñó principalmente con dos microprocesadores Microblaze que cuentan con buses PLB y memoria compartida, en donde se almacenan los datos. Esto último, permitió obtener diferentes tiempos de procesamiento de las imágenes médicas, gracias a la posibilidad de acceso simultáneo a memoria.

Debido a la naturaleza de los datos paralelizables que tienen las imágenes, se utilizó el método de programa simple y múltiples datos (SPMD) [4-7], en donde cada procesador puede ejecutar el mismo programa, excepto por el código que se encarga del sincronismo, que se implementó mediante mailbox (paso de mensajes), entre los procesadores. El sincronismo es necesario debido a que el Procesador MB0, encargado de recibir los datos y almacenarlos en memoria, deberá informar al procesador MB1 cuando hay datos disponibles para procesar, a su vez, este último deberá informar cuando finalice el procesamiento para devolver la imagen resultado al host [8-11].

Las imágenes a procesar fueron leídas con el software Matlab y transmitidas desde una PC (Host) por puerto serie hacia la FPGA Virtex V, en donde se realizó todo el procesamiento y se obtuvo la imagen procesada para su posterior almacenamiento en la memoria compartida [12-14].

Resultados

Los tiempos de lectura, escritura y procesamiento se midieron para la mínima cantidad de filas, lo que permitió sacar conclusiones respecto a los tiempos individuales por fila y de la imagen completa.

Los tiempos de procesamiento con dos procesadores respecto al uso de un solo procesador para el filtro de mediana/erosión/dilatación fueron de 1870us y 3715us, para el filtro de media de 120us y 215us, por último para el de sobel de 3040us y 6055us respectivamente. Estos resultados permiten observar el beneficio obtenido al implementar un sistema multiprocesador, que usa memoria compartida con múltiples puertos, respecto a uno simple procesador. El análisis que se puede hacer en relación a los tiempos de los distintos algoritmos es que a medida que la complejidad del algoritmo se incrementa, el tiempo de lectura/escritura de memoria compartida se vuelve despreciable respecto al de procesamiento [15].

Conclusiones

En este trabajo se muestra la potencialidad de las plataformas FPGAs para implementar sistemas multi-core. La flexibilidad de estos sistemas permite evaluar diferentes diseños en la búsqueda de eficiencia al momento de procesar algoritmos de imágenes con características específicas (basados en operadores de ventana).

Mediante los resultados obtenidos se demostró que el sistema diseñado permite procesar de manera eficiente los algoritmos de procesamiento, aprovechando las características paralelizables que proveen los datos de este tipo de algoritmos. Los tiempos obtenidos en la ejecución de los filtros usando uno y dos procesadores fueron muy satisfactorios, debido a que cada procesador accede a datos ubicados en diferentes partes de la memoria simultáneamente.

Referencias

- [1] John L. Semmlow: *Biosignal and Biomedical Image Processing*, Marcel Dekker, New Jersey, 2004.
- [2] William K. Pratt: *Digital Image Processing*, Wiley, Los Altos California, 2004.
- [3] Dougherty, E.: *An Introduction to Morphological Image Processing*, SPIE, Bellingham, WA, 1992.
- [4] HUSSAIN, Z., *Digital image processing: practical applications of parallel processing techniques*, Ellis Horwood, Chichester-Inglaterra, 1991.
- [5] D.E. Culler, J.P. Singh. *Parallel Computer Architecture a Hardware/Software Approach*. Morgan Kaufman, 1999.
- [6] A. Grama, A. Gupta, G. Karypis, V. Kumar. *Introduction to Parallel Computing*. Segunda edición. Addison Wesley, 2003.
- [7] Parhami, Behrooz, *Introduction to Parallel Processing*. s.l. : Springer US, 2006. 1567-7974.
- [8] Jesman R., Martinez Vallina F., Sanjie J., *MicroBlaze Tutorial Creating a Simple Embedded System and Adding Custom Peripherals Using Xilinx EDK Software Tools*, Illinois Institute of technology, 2006.
- [9] *MicroBlaze Processor Reference Guide – Embedded Development kit EDK 13.2*, Xilinx, Junio de 2011
- [10] Vasanth Asokan, *Designing Multiprocessor Systems in Platform Studio*, Xilinx White Paper WP262, Noviembre de 2007.
- [11] Vasanth Asokan, *Dual Processor Reference Design Suite*, Application Note: embedded Processing, Octubre de 2008.
- [12] JORGE OSIO; JOSÉ RAPALLINI; ANTONIO ADRIÁN QUIJANO; JESÚS OCAMPO. *Implementación de un Algoritmo para procesamiento de imágenes en una FPGA*. Libro de memorias del Primer Congreso de Microelectrónica Aplicada. 2010. Artículo Completo. Congreso de microelectrónica Aplicada UEA 2010. Universidad Nacional de la Matanza; Universidad
- [13] Osio J., Aróztegui W, Rapallini J., Quijano A., Ocampo J., *Desarrollo de Algoritmos de Procesamiento de Imágenes Basados en Operadores de Ventana sobre una FPGA*. XVII Congreso de Iberchip. Bogotá: Universidad Nacional de Colombia. 2011 vol.1 n°1. p156 - 161. issn 2177-1286.
- [14] JORGE R. OSIO, WALTER ARÓZTEGUI, JOSÉ A. RAPALLINI, ANTONIO QUIJANO. *Procesamiento de Imágenes Médicas Sobre una FPGA para la detección de bordes*. Argentina. Rosario. 2012. Libro. Artículo Completo. Tercer Congreso de Microelectrónica Aplicada. Universidad Nacional de Rosario
- [15] [27] OSIO J., MONTEZANTI D., MORALES M. *Análisis de Eficiencia en Sistemas Paralelos*. Argentina. Ushuaia. 2014. Libro. Workshop. XVI WORKSHOP DE INVESTIGADORES EN CIENCIA DE LA COMPUTACIÓN. Universidad Nacional de Tierra del Fuego, Antártida e Islas del Atlántico Sur.

Sistema de Inspección Visual de Baldosas Cerámicas basado en SoC FPGA

Tomás Medina
Depto. De Computación y Sistemas
UNICEN
Tandil, Argentina
medinatomasariel@gmail.com

Martín Vázquez, Oscar Goñi, Marcelo Tosini y
Lucas Leiva
LabSET - INTIA
UNICEN
Tandil, Argentina
{mvazquez, oegoni, mtosini, lleiva}@exa.unicen.edu.ar

La calidad del producto final define el futuro de la industria y sólo una inspección objetiva y estricta puede prometer la mejor calidad del productos [1]. La incorporación de sistemas de inspección automática en la industria (Machine Vision) [2-4] tiene como objetivo recopilar la información necesaria para determinar si el producto se encuentra dentro de los parámetros correctos utilizando bases de conocimiento. Esta técnica permite la inspección en tiempo real sobre las líneas de producción, poseen una salida determinística, son sistemas robustos y pueden ser incorporados en ambientes hostiles para los humanos (radiación, temperatura, sustancias tóxicas, entre otros).

Generalmente las industrias incluyen en sus líneas de producción un alto grado de automatización [5-6], pero el control de calidad sigue siendo realizada por humanos. En este sentido, la capacidad de un ser humano para llevar a cabo este proceso depende de su entrenamiento, conocimiento y experiencia, pudiendo resultar además en clasificaciones erróneas de los productos sujetas al nivel de fatiga y la subjetividad [7].

Existen soluciones comerciales que permiten el prototipado rápido y la configuración/adaptación de sistemas de visión, tales como Matrox imaging library (MIL) y NI Vision Builder. También se encuentran disponibles librerías de código abierto, como OpenCV, SimpleCV, Visualization toolkit (VTK), Darwin, y ROVIS [8]. En este sentido, muchos de los desarrollos de sistemas de visión actuales están basados en el uso de OpenCV, que proporciona una infraestructura para construir aplicaciones de visión en tiempo real.

Por otra parte, como plataforma de implementación de sistemas industriales, se destacan los microcontroladores, los DSPs y los FPGAs. Los FPGA son ampliamente utilizados en aplicaciones industriales debido a su alta velocidad, flexibilidad, su paralelismo inherente, su buena relación costo-rendimiento y la gran variedad de recursos lógicos especializados disponibles. Como consecuencia, han sido extensamente analizados a lo largo de los años desde la perspectiva de la electrónica industrial [9]. En los últimos años, se ha expandido el uso de soluciones heterogéneas que combinan la naturaleza secuencial de los microprocesadores, con la paralela proveniente de la lógica programable (FPGA)

en un único dispositivo comúnmente denominado SoC FPGA. Asimismo, con el fin de promover la productividad de los diseñadores de sistemas embebidos y disminuir el tiempo de lanzamiento al mercado, los fabricantes han adoptado en sus herramientas capacidades de Síntesis de Alto Nivel (HLS).

El soporte brindado por la arquitectura heterogénea, permite realizar una compilación cruzada de las librerías de OpenCV para ejecutar sobre dispositivos SoC FPGAs, y utilizar las funciones en el dispositivo. Además, las herramientas de HLS incluidas brindan la posibilidad de implementar aceleradores de aquellas funciones que generan un cuello de botella en el sistema [10-11]. Por otra parte, Xilinx ha lanzado reVision Stack [12], que incluye una amplia gama de recursos de desarrollo para plataformas, algoritmos y desarrollo de aplicaciones. Esto incluye soporte para las redes neuronales más populares, incluidas AlexNet, GoogLeNet, SqueezeNet, SSD, FCN, CNN, y la posibilidad de integración con la librería xfOpenCV[13] con más de 50 funciones OpenCV equivalente optimizadas para ser implementadas en FPGAs de Xilinx.

El objetivo principal del proyecto es el desarrollo de un sistema de visión para la detección de defectos basado en el uso de un dispositivo SoC FPGA, para ser aplicado en el ámbito regional, orientado a la detección de fallas en la industria de pisos cerámicos. Como objetivo secundario se espera poder contribuir en el desarrollo de mecanismos y metodologías, que permitan explotar los beneficios ofrecidos de las librerías disponibles para el procesamiento de imágenes en conjunto con la tecnología propuesta como plataforma.

El conjunto de fallas a identificar comprende: detección de rayones, gotas, y fallas de material, verificación de dimensión, y comprobación de daños en esquinas y bordes. Se han desarrollado un conjunto de algoritmos en Matlab capaces de identificar el subconjunto de defectos detallado, y actualmente se está realizando la migración de estos algoritmos a OpenCV.

En la siguiente etapa los algoritmos serán implementados sobre el microprocesador con el que cuenta la plataforma. El resultado obtenido de la migración permitirá seleccionar los procesos críticos que forman un cuello de botella al procesamiento, y serán acelerados utilizando la lógica

programable con la que cuenta el dispositivo Zynq-7000. En esta etapa se espera aplicar las herramientas de síntesis de alto nivel que provee el fabricante (Vivado-HLS) y/o el desarrollo de cores IP (núcleos de propiedad intelectual) dedicados en lenguaje de descripción de hardware. Asimismo, se analizarán y propondrán alternativas a los algoritmos complejos, incursionando en optimizaciones que permitan obtener mejoras mediante la reducción de operaciones y/o espacial, y alcanzar el requerimiento de frecuencia de operación del sistema. La solución embebida lograda utilizará como entrada el conjunto de imágenes mencionado anteriormente.

En la fase final de la propuesta, el sistema será adaptado para permitir su ejecución utilizando como entrada una señal de video continua, y evaluado en términos de precisión en la detección y latencia. Se esperan lograr en esta etapa métricas que permitan cuantificar los resultados obtenidos del sistema en términos de tiempo de respuesta y tasa de error.

REFERENCIAS

- [1] D. T. Pham y R. J. Alcock, *Smart Inspection Systems: Techniques and applications of intelligent vision*, Academic Press, 2003.
- [2] E. R. Davies, *Machine Vision: Theory, algorithms, practicalities*, Fourth Edition, Academic Press, 2004.
- [3] U. S. Khan, "Automatic inspection system using Machine Vision", *Proceedings of the 34th Applied Imagery and Pattern Recognition Workshop*, pp. 217-222, 2005.
- [4] H. Golnabi, «Design and application of industrial Machine Vision systems», *Robotics and Computer-Integrated Manufacturing*, vol. 23, issue 6, pp. 630-637, December 2007.
- [5] Y. Meena y A. Mittal, «Blobs and Scratches Detection on Plain Ceramic Tile Surface», *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, issue 7, pp. 647-652, 2013.
- [6] H. Elbehiery, A. Hefnawy y M. Elewa, «Surface Defects Detection for Ceramic Tiles Using Image Processing and Morphological Techniques», *World Academy of Science, Engineering and Technology*, pp. 158-162, 2005.
- [7] S. Singh y M. Kaur, «Machine Vision System for Automated Visual Inspection of Tile's Surface Quality», *IOSR Journal of Engineering*, vol. 2, issue 3, pp. 429-432, 2012.
- [8] LIU, Zheng, et al. *Integrated Imaging and Vision Techniques for Industrial Inspection*. Springer, 2015.
- [9] Andina, Juan Jose Rodriguez, Eduardo De la Torre Arnanz, and Maria Dolores Valdes. *FPGAs: Fundamentals, Advanced Features, and Applications in Industrial Electronics*. CRC Press, 2017.
- [10] Bailey, Donald G. "The advantages and limitations of high level synthesis for FPGA based image processing." *Proceedings of the 9th International Conference on Distributed Smart Cameras*. ACM, 2015.
- [11] Qasaimeh, Murad, and Ehab Najeh Salahat. "Real-Time Image and Video Processing Using High-Level Synthesis (HLS)." *Computer Vision: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2018. 1004-1022.
- [12] reVISION Getting Started Guide, <http://www.wiki.xilinx.com/reVISION+Getting+Started+Guide>
- [13] Xilinx xfOpenCV Library, <https://github.com/Xilinx/xfopencv/blob/master/README.md>

Implementación de Filtros Digitales SAL/CAL para la Medición de Voltaje Eficaz de la Red Eléctrica.

R. Y. Carmen¹, M. A. Palacios², W. J. Castillo¹, A. S. Aliaga²

¹ Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad Nacional de Piura

² Facultad de Ingeniería Mecánica-Eléctrica, Universidad de Piura

Email: rjonatansa@outlook.es, marco9309@gmail.com, jannirocv@gmail.com, asaliagaz@hotmail.com

Resumen— El presente trabajo describe un sistema que permite medir el voltaje eficaz; también denominado RMS, de la red eléctrica, basado en la implementación de los filtros digitales SAL y CAL, los cuales son filtros del tipo respuesta finita al impulso (FIR), derivados de las funciones de Walsh. El sistema consta de un bloque de acondicionamiento de la señal de la red, con lo cual se logró obtener una señal exenta de ruidos y con niveles de tensión correctos para ser procesados por el conversor analógico digital (ADC) del microcontrolador PIC. La segunda etapa de adquisición de la señal, está basada en un microcontrolador PIC 12F1840 de Microchip, el cual cuenta con los módulos necesarios (Convertidor Analógico Digital (ADC), módulo de comunicación serie (UART) y Timer's) y la suficiente capacidad de memoria de datos y programa, lo que hizo posible la implementación de los filtros digitales y el algoritmo para el cálculo del valor de voltaje eficaz.

Palabras Claves— Voltaje RMS, Acondicionamiento de Señales, Procesamiento Digital de Señales, Filtros SAL y CAL.

I. INTRODUCCIÓN

Los voltajes y las intensidades en los sistemas de distribución de energía se expresan siempre en función de sus valores eficaces [1]. El valor eficaz es un parámetro que se utiliza para caracterizar formas de onda, definido como el valor de tensión continua que sería capaz de suministrar la misma potencia (a un elemento del circuito que disipa toda la potencia), que la potencia que le proporciona la forma de onda con dicho valor eficaz [2].

El procesamiento digital de señales, mediante la aplicación de filtros y algoritmos, han permitido grandes mejoras en los instrumentos de medición, destacando los de Fourier y las funciones de Walsh. Una ventaja de utilizar las funciones de Walsh para el filtrado de señales es que convierten las multiplicaciones por números irracionales de los filtros de Fourier en simples asignaciones de signos a los valores de las muestras.

A. Funciones de Walsh

Estas funciones son cuadradas de orden N, donde para formar un conjunto de N funciones es necesario que N sea potencia de 2 [4]. La función de Walsh $w_2(t)$, es la función tipo Seno (SAL) y $w_3(t)$ es la función tipo Coseno (CAL). En [5] se menciona que de la misma forma en que se puede calcular el par Seno-Coseno en el algoritmo de Fourier, dada su ortogonalidad, es que se pueden calcular los fasores del par SAL-CAL.

Para el cálculo de los filtros SAL y CAL. Según [6], se menciona lo siguiente: "Los 2 filtros usados son del tipo FIR, filtros de respuesta al impulso, los cuales son funciones de Walsh de un ciclo de duración". Además, se tiene la señal senoidal $v(t)$ en el dominio del tiempo como señal de entrada para el filtro digital. Discretizando, para $t = kT$, se expresa de la siguiente manera:

$$v(kT) = V_0 \sin(2\pi f k T + \theta) \quad (1)$$

Donde, T es el tiempo de muestreo, k es el número de muestras por ciclo ($k = 0,1,2,3$) y f es la frecuencia de la señal. Utilizando las 4 muestras de la señal en un periodo denominadas $V_k, V_{k+1}, V_{k+2}, V_{k+3}$ y utilizando lo mencionado en [5], los filtros sal y cal quedan determinados de la siguiente forma:

$$S(k) = V_k + V_{k+1} - V_{k+2} - V_{k+3} \quad (2)$$

$$C(k) = V_k - V_{k+1} - V_{k+2} + V_{k+3} \quad (3)$$

Reemplazando (1) en (2) y (3), para un $T = \frac{4}{f}$ y aplicando una reducción al primer cuadrante con simplificaciones trigonométricas de seno y coseno de suma de dos ángulos para $S(k)$ y $C(k)$, se obtiene:

$$S(k) = 2\sqrt{2}V_0 \left[\sin\left(\frac{\pi k}{2} + \theta + \frac{\pi}{4}\right) \right] \quad (4)$$

$$C(k) = -2\sqrt{2}V_0 \left[\cos\left(\frac{\pi k}{2} + \theta + \frac{\pi}{4}\right) \right] \quad (5)$$

Considerando $\varphi = \frac{\pi}{2}k + \frac{\pi}{4} + \theta$, y considerando la forma fasorial de V , se puede determinar el valor del voltaje RMS, mediante la siguiente ecuación:

$$\vec{V} = \frac{C}{2\sqrt{2}} + j\frac{S}{2\sqrt{2}} = |V_{rms}| = \frac{\sqrt{S^2 + C^2}}{4} \quad (6)$$

II. DESCRIPCIÓN DEL SISTEMA

El sistema de medición de voltaje RMS esta conformado por dos bloques. El primer bloque, esta dedicado al acondicionamiento de la señal, integrado por un transformador y un divisor de voltaje para reducir el voltaje de la red; un buffer, un filtro activo y un circuito offset utilizando un amplificador operacional (OPAMP). El segundo bloque de adquisición de la señal, está conformada por un microcontrolador para la implementación de los filtros SAL y CAL y el algoritmo de cálculo de valor RMS, como se muestra en la Figura 1.

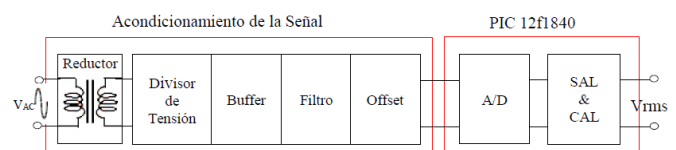


Fig. 1. Diagrama de Bloques del Sistema.

A. Etapa de Reducción

La etapa de reducción del sistema, utilizo un transformador reductor de 220VAC a 12 VAC. El factor de reducción del transformador para los valores experimentales ($V_s = 12.8v$ y $V_p = 232.3v$) concluyeron un $\alpha \approx 0.055$.

La adición del divisor de tensión tiene como finalidad obtener un valor de voltaje de 1.5 voltios rms, el cual en valor pico sería de 2.12 voltios. Determinándose valores de resistencia de 10 kΩ y 1.3 kΩ en la salida. Sin embargo, se usó una resistencia comercial de 1.2kΩ. Finalmente, la

ganancia de esta etapa mediante los valores previamente definidos, concluyeron un $G_{Div} = 0.107$.

B. Etapa de Buffer

Para evitar el acoplamiento de impedancias de la etapa de reducción y la etapa de filtrado, se implementó una etapa que permita desacoplar impedancias, formado por un circuito buffer o seguidor de voltaje con un OPAMP LM324.

Este circuito al ser un seguidor de voltaje tiene una ganancia de 1. Es decir, el voltaje de entrada en el Buffer es igual al voltaje de salida del Buffer ($G_{buffer} = 1$).

C. Etapa de Filtrado

La etapa de filtro consta de un filtro activo pasabaja de segundo orden Sallen-Key. Para este diseño se consideró una resistencia de retroalimentación de valor 30kΩ para compensar las corrientes de Bias. Realizando el análisis en frecuencia del filtro utilizando el diagrama de bode, se obtuvo una frecuencia de corte de 106.1Hz ($G_{Filtro} = 0.756$). Dada la frecuencia de muestreo del sistema de 240Hz (4 veces la frecuencia de la red eléctrica), de acuerdo al teorema de Nyquist, el filtro opera en los márgenes correctos.

D. Etapa de Offset

Esta es la última etapa consta de un amplificador diferencial, dado que la señal tratada tiene valores positivos y negativos. Por este motivo se realiza la etapa de offset para desplazar la señal en un valor 2.5v hacia arriba, con lo cual siempre es positiva. En esta etapa, se obtuvo un voltaje máximo superior de 4.32v y un voltaje máximo inferior de 0.680v, concluyendo en $G_{Offset} = 1.25$. Finalmente, se integraron cada una de las etapas del acondicionamiento de señal como se muestra en la Figura 2.

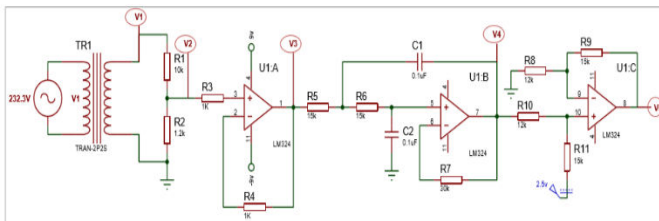


Fig. 2. Visualización de la etapa de acondicionamiento de señal.

III. ALGORITMO

La implementación del algoritmo, esta basado en la aplicación de las ecuaciones (2) y (3), para el cálculo de los filtros SAL y CAL, y la ecuación (6), para el cálculo del valor RMS, además de considerar las constantes del bloque de acondicionamiento de la señal, como del ADC. Para la generación del tiempo de muestreo de utilizo un timer, configurado para generar una interrupción cada 4.16 ms, equivalente a una frecuencia de 240 Hz.

IV. RESULTADOS

Se realizaron pruebas de medición en cada uno de los puntos de salida de voltaje indicados en la Figura 2 en las diferentes etapas del sistema, de ese modo tenemos los datos experimentales agrupados en la Tabla 1, correspondientes al voltaje del transformador, Buffer, Filtro, y Offset. Con estos valores se calculó las ganancias en cada etapa.

Tabla 1. Lectura de Voltajes y Ganancias del Sistema.

Nº	V ₂	V ₃	V ₄	V ₅	G _{buffer}	G _{filtro}	G _{offset}
1	12.79	1.393	1.048	1.308	0.1089	0.7523	1.2481
2	12.8	1.393	1.049	1.309	0.1088	0.7531	1.2479
3	12.82	1.394	1.05	1.31	0.1087	0.7532	1.2476
4	12.82	1.395	1.051	1.312	0.1088	0.7534	1.2483
5	12.84	1.397	1.053	1.312	0.1088	0.7538	1.2460
6	12.83	1.396	1.052	1.313	0.1088	0.7536	1.2481
7	12.84	1.396	1.052	1.313	0.1087	0.7536	1.2481
8	12.84	1.396	1.052	1.313	0.1087	0.7536	1.2481
9	12.84	1.398	1.053	1.315	0.1089	0.7532	1.2488
10	12.72	1.384	1.042	1.3	0.1088	0.7529	1.2476

Los valores mostrados en la Figura 3, corresponden al número de frame's (4 muestras), los valores de las 4 muestras tomadas durante el periodo actual, el valor del voltaje que ingresa al ADC y finalmente el valor de voltaje RMS. Asimismo, se calculó el valor promedio del Voltaje RMS mediante los 10 primeros valores, obteniéndose un $V_{rms Prom.} = 229.4$.

Frame	V1	V2	V3	V4	Vrms
174	872	625	154	397	1.8 Vrms=229.7
175	610	878	418	146	1.8 Vrms=229.7
176	257	787	770	240	1.8 Vrms=229.1
177	146	423	878	605	1.8 Vrms=229.5
178	434	142	588	882	1.8 Vrms=229.7
179	805	277	217	744	1.8 Vrms=228.9
180	848	682	178	340	1.8 Vrms=229.5
181	500	889	528	136	1.8 Vrms=229.8
182	170	670	856	358	1.8 Vrms=228.6
183	220	276	803	751	1.8 Vrms=229.5
184	582	142	439	881	1.8 Vrms=230.0
185	872	396	152	625	1.8 Vrms=228.8
186	775	783	252	239	1.8 Vrms=229.6

Fig. 3. Variables de Salida del Sistema-UART PICKIT2.

Para el contraste del diseño, se midió el voltaje de la red eléctrica con el multímetro Hyelec, obteniendo una lectura promedio de 229.6v, al cual se denominó voltaje Patrón. Finalmente, se determina el error porcentual considerando la magnitud de referencia de 232.3v y una magnitud del error de 0.2v, determinándose un error porcentual de 0.086%. Este es un resultado favorable, dada la resolución del ADC de 10 bits.

V. CONCLUSIONES

La utilización de los filtros SAL y CAL brindan resultados con alta precisión al evaluar variables eléctricas de corriente alterna (AC), aprovechando la ventaja de utilizar las funciones de Walsh para el filtrado de señales, mediante la simplificación de asignaciones de signos a los valores de las muestras.

La precisión en la medición requiere además de una frecuencia de muestreo estable, que mantenga la relación de ser 4 veces la frecuencia de la señal de entrada.

REFERENCIAS

- [1] S. E. Elias (s.f.). "Corriente Alterna", pag.52.
- [2] L. A. Huergo, "Corriente Alterna" Departamento de Telecomunicaciones, Cap 02.
- [3] F. A. Veiga Benini, "Transformada de Walsh-Hadamard para ingenieros como introducción a la transformada de Fourier," VI Semana de Matemática y Educación, Sao Paulo, 2017.
- [4] I. Díaz Verduzco, "Estudio comparativo de algoritmos de filtrado digital para protección de Líneas de transmisión," Tesis presentada para obtener el grado de Magister, Monterrey, Mexico, 1994.
- [5] A. S. Aliaga Zegarra, "Microprocessor-Based Time Overcurrent Relay: development simulation, design and testing," Tesis presentada para obtener el grado de Magister, Ohio, Estados Unidos, 1979.

Implementación alternativa de sensores AM2302 en un firmware cooperativo

1st Ariel Dalmas Di Giovanni
Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
adigiovanni@citedef.gob.ar

2nd Sergio Saluzzi
Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
ssaluzzi@citedef.gob.ar

3rd Martín Morales
Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
mmorales@citedef.gob.ar

4th Adrián Stacul

Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
astacul@citedef.gob.ar

5th Sebastián Alvarez

Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
salvarez@citedef.gob.ar

6th Daniel Pastafiglia

Laboratorio de Técnicas Digitales
CITEDEF

Villa Martelli, Pcia. Bs. As., Argentina
dpastafiglia@citedef.gob.ar

Resumen—El trabajo presenta una alternativa de implementación de sensores AM2302 sobre un microcontrolador de arquitectura Cortex- M4, con *firmware* cooperativo. Se presenta el desarrollo e implementación de una alternativa para administrar estos sensores de forma tal que se preserven los recursos de *hardware* involucrados.

Palabras clave—Sensores, microcontrolador, Cortex-M, cooperativo.

I. INTRODUCCIÓN

El sensor AM2302 [1] es un sensor de temperatura y humedad de sólo tres pines: alimentación, masa y una línea de datos, facilitando la conexión con una MCU. La línea de datos define una conexión punto a punto entre el sensor y la MCU que oficia de maestro. Esta forma de comunicación es similar, pero no totalmente compatible con el estándar 1-Wire® [2], sin embargo la problemática asociada a la administración del pin de datos por parte de la MCU es la misma en ambos casos. Implica que el maestro deba realizar una tarea de manejo de bits, lo que usualmente se realiza mediante la técnica conocida como “*Bit Banging*” [3] [4]. En sí es una tarea bastante ineficiente: se invierte mucho tiempo de la CPU en el manejo del pin, para poder cumplir con las restricciones temporales impuestas por el sensor, lo que condiciona la ejecución de otras tareas. Por estos motivos existen trabajos que buscan delegar en un *hardware* dedicado el problema, como por ejemplo, resolverlo mediante la utilización de una UART como se puede ver en [5]. Si bien la solución es válida, se utiliza un recurso limitado como ser una UART, sobre todo en los casos donde la MCU es pequeña, en donde no abundan estos periféricos. Otra opción comercialmente disponible es delegar el manejo de la comunicación en un maestro externo a la MCU vinculado por I²C, como el circuito integrado DS2484 [6]. En este caso

Trabajo realizado en el Laboratorio de Técnicas Digitales del Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF). Ministerio de Defensa.

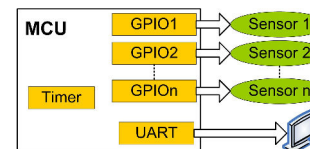


Figura 1. Diagrama conceptual para “N” sensores.

se encarece el costo final del producto, además del tiempo de desarrollo adicional para manejar el integrado.

El trabajo presenta una solución alternativa a las antes mencionadas para administrar la comunicación con uno o más sensores del tipo AM2302 utilizando un *firmware* cooperativo.

II. DESARROLLO E IMPLEMENTACIÓN

La comunicación con el sensor utiliza niveles y tiempos para señalar los distintos estados y datos, el maestro debe medir dichos tiempos tanto de los pulsos entrantes y salientes. Esta tarea se propone realizarla mediante un *timer* dedicado del microcontrolador (que opera como dispositivo maestro). Como los sensores a utilizar no soportan direccionamiento, se dedicó un pin GPIO para cada sensor, tal cual se muestra en el esquema de la Fig. 1. El *timer* es compartido por todos los sensores, resolviendo mediante el *firmware* el conteo de tiempo y el estado de cada sensor. Para simplificar el uso se desarrolló un *driver* de *firmware* para el sensor basado en un tipo de dato abstracto. En las siguientes subsecciones se presentan los distintos aspectos que conciernen al desarrollo del *hardware*, la estrategia de muestreo y del *firmware*.

A. Hardware

El *hardware* de evaluación se basó en un Kit NUCLEO-F411RE de STMicroelectronics [7], y dos sensores AM2302 cada uno conectado a un pin GPIO del MCU. El kit posee un microcontrolador Cortex-M4, con frecuencia de 48MHz de reloj.

El *timer* interno de la MCU se configura para que interrumpa en forma periódica, en modo auto recarga con un período de $10\mu s$. Como los sensores poseen un tiempo mínimo de encuesta de 2 segundos, el *timer* está operativo sólo durante los momentos en los que se están adquiriendo los datos de alguno de los sensores, luego estará apagado, disminuyendo así las interrupciones periódicas generadas a la CPU.

B. Estrategia de muestreo

El protocolo establece un mecanismo de señalización por parte del maestro para iniciar la comunicación. Luego, el maestro libera el *bus* colocándose en escucha, el sensor envía una respuesta y finalmente el sensor comienza a enviar los datos de la medición. Todo esto ocurre sobre una única línea de comunicación, por ende es necesario muestrear los pulsos de entrada. La especificación del sensor establece para los distintos tipos de pulsos un rango de tiempos admisibles. En este sentido, un dato se señala mediante un pulso en estado bajo de una duración determinada y según sea el próximo pulso, largo o corto corresponderá a un cero o a un uno lógico, respectivamente. De esta manera la estrategia utilizada consistió en que una vez detectado el inicio de la comunicación se muestreé a mitad de cada pulso leído y se contabilicé, así, el tiempo de espera máximo (*timeout*) establecido para el rango de tiempo admisible especificado por el fabricante. El tiempo más exigente corresponde a un valor de $20\mu s$, que con esta estrategia y la configuración del *timer* mencionada anteriormente queda adecuadamente determinado.

C. Firmware

El *firmware* se basa en un sistema cooperativo, del tipo *loop* infinito [8], es decir una ejecución secuencial de tareas, con la premisa que cada tarea no se apropie de la CPU realizando lazos bloqueantes innecesarios. El *driver* implementa la estrategia de muestreo antes mencionada, mediante una máquina de estados finitos y un tipo de dato abstracto. De esta manera es factible que el *driver* se ejecute en un contexto de un *firmware* cooperativo e internamente quede registrado el contexto (estado) actual de cada sensor. El *driver* denominado AM2302, posee:

- **Am2302_Init**: Función de inicialización.
- **Am2302_Proceso**: Función principal de ejecución permanente en el lazo principal.
- **Am2302_Leer**: Funciones de lectura de datos.
- **Am2302_BaseTiempoProceso**: Función para ejecución en el servicio de interrupciones del *Timer*.

Este conjunto de funciones permite abstraer al usuario del *driver* de la configuración del *timer*, el manejo del pin GPIO y otras inicializaciones asociadas al *hardware*. El usuario sólo debe verificar periódicamente cuándo finalizó una adquisición, y leer el dato de las mediciones, es decir, al usuario se lo abstrae del manejo de la interfaz física con el sensor.

El *driver* fue programado en Lenguaje C, utilizando el IDE *Atollic True STUDIO*[®], donde se consiguió manejar múltiples instancias del *driver*, a través de la definición de un tipo de dato abstracto.

III. RESULTADOS

La metodología de evaluación consistió en realizar una adquisición de cada sensor en forma periódica cada 30 segundos y enviar el reporte de los datos medidos a través de un puerto serie (UART) de la MCU. Estos datos fueron verificados mediante un *software* ad-hoc conectado a la MCU.

También, se realizaron mediciones temporales con un osciloscopio digital en puntos de medición específicos para verificar el *driver*, como ser: los puntos de muestreo de la señal de datos, el tiempo de adquisición promedio por sensor: 500ms, y la estabilidad del *timer*: $10\mu s \pm 1\mu s$.

El tiempo de adquisición promedio determina la ventana de actividad del *timer*, en esta ventana el microcontrolador recibirá interrupciones periódicas del *timer* cada $10\mu s$. Como la CPU posee una frecuencia de reloj de 48MHz, quedan disponibles 480 ciclos de reloj entre cada interrupción, sumamente útiles para mantener activas otras tareas de la MCU, sin perjuicio de la adquisición.

IV. CONCLUSIONES

Se logró el objetivo planteado, manejar más de un sensor AM2302 tal que conviva con un esquema de *firmware* cooperativo, y con una utilización mínima de periféricos: un *timer*. El *driver* desarrollado abstrae del manejo de la interfaz física: se encarga de configurar el *timer*, el pin GPIO, y gestiona la comunicación con el sensor, sin que el usuario tenga que estar involucrado en todos estos procesos. El usuario, sólo consulta si hay nuevos datos disponibles, y realiza la lectura de los mismos, que se convierten a las magnitudes físicas representativas.

La solución fue probada y verificada en *hardware*. Entendiéndola como una alternativa a las otras opciones mencionadas anteriormente: el “*Bit Banging*”, y la utilización de una UART. Como ventaja frente a esta última, la solución propuesta utiliza sólo un *timer*, como recurso de *hardware* adicional, preservando la UART para las función de comunicación para la que fue concebida. Como ventaja frente a la estrategia del “*Bit Banging*”, se mejora el rendimiento temporal de las otras tareas a atender por parte de la MCU, ejecutándose la tarea de administración y adquisición del sensor como una tarea más del *firmware* cooperativo.

Como trabajo futuro se pretende migrar esta solución a un Cortex-M0 que es una arquitectura más apta a la aplicación en términos de recursos tanto de *hardware* como de costos. Esto posibilitará mejorar el ahorro de energía del sistema, pudiendo utilizarse en aplicaciones como redes de sensores inteligentes, IoTs, entre otras.

AGRADECIMIENTOS

Se agradece: al Técnico Gerardo García por el apoyo diario durante la realización del trabajo. A la Dra. Claudia V. Vassena, Dra. Laura V. Harburguer, y Dra. V. Paula González por ser las motivadoras indirectas del trabajo.

REFERENCIAS

- [1] "Am2302 Product Manual". Aosong.
Available: <http://www.aosong.com/en/>.
- [2] Bernhard Linke, "Overview of 1-Wire Technology and Its Use," Tutorial 1796. Maxim Integrated. June 2008.
Available: <https://pdfserv.maximintegrated.com/en/an/AN1796.pdf>
- [3] A. Halder, R. Dasgupta, J. Chepada and N. Dash, "Implementation of MCU Invariant I2C Slave Driver Using Bit Banging," ICONS 2011 : The Sixth International Conference on Systems. IARIA 2011. ISBN:978-1-61208-114-4.
- [4] "Efficient Bit-Banged SPI Port for 8051-Compatible Microcontrollers," Application Note 3524. Maxim Integrated, June, 2005.
Available: <https://pdfserv.maximintegrated.com/en/an/AN3524.pdf>
- [5] "Using a UART to Implement a 1-Wire Bus Master," Tutorial 214. Maxim Integrated, 2002.
Available: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/214>
- [6] "DS2484," Datasheet, Maxim Integrated, July, 2015.
Available: <https://datasheets.maximintegrated.com/en/ds/DS2484.pdf>
- [7] "UM1724. User manual. STM Nucleo-64 board,". STMicroelectronics, noviembre, 2016.
- [8] G. Galeano, "Programación de sistemas embebidos en C," 1er ed. Alfaomega Grupo Editor: 2009, pp: 466-468.

Vulnerabilidad de los dispositivos utilizados en aplicaciones de IoT

Osvaldo Marianetti, Ernesto Chediak, Pablo Godoy
Ciencias de la Computación. Facultad de Ingeniería
Universidad Nacional de Cuyo Mendoza, Argentina
olmarianetti@gmail.com,
ernestochediack@gmail.com,
pablodgodoy@gmail.com

Osvaldo Marianetti, Daniel Fontana Instituto de
Microelectrónica. Facultad de Ingeniería Universidad
de Mendoza Mendoza, Argentina
osvaldo.marianetti@um.edu.ar,
daniel.fontana@um.edu.ar

Se pueden citar diferentes razones que permiten que los dispositivos que forman parte de un entorno de IoT sean vulnerables. Estas razones pueden originarse en diseños inadecuados, entornos no regulados o regulados ineficientemente, entre otras.

Organismos gubernamentales, por ejemplo, el departamento de Seguridad Nacional de Estados Unidos, ha planteado una serie de principios estratégicos para que sean tenidos en cuenta por desarrolladores, fabricantes, operadores y usuarios de aplicaciones IoT. Estos principios expresan necesidades tales como: incorporar la seguridad en la fase de diseño, actualizaciones de seguridad y gestión de vulnerabilidades de avanzada, construir sobre prácticas de seguridad probadas, compartir información relacionada a incidentes y vulnerabilidades y priorizar las medidas de seguridad según el impacto potencial.

El siguiente trabajo presenta un ejemplo desde el punto de vista del desarrollador. En dicho proyecto se usan dispositivos comerciales, que se emplean comúnmente en aplicaciones de IoT. Se implementó un equipo de control de condiciones ambientales. El producto final tiene por objetivo la medición de los siguientes parámetros ambientales: temperatura, humedad, iluminación, nivel de ruido, y presencia.

El desarrollo se realizó a partir de un módulo comercial de sensores empresa Global TE CONNECTIVITY, y el dispositivo seleccionado es un sensor AmbiMate MS4, con la opción de detección de eventos sonoros o medición de la intensidad de ruido ambiental. Estos sensores son ampliamente usados en aplicaciones tales como automatización de edificios, calidad de aire, manejo de energía, confort de espacios de trabajo y controles zonales ambientales. Los módulos AmbiMate poseen un core de 4 sensores que incluyen movimiento (PIR), luz, temperatura y humedad y opcionalmente están disponibles versiones con sensores de VOC, CO2 e intensidad de ruido (micrófono). Para la conectividad del sensor se utilizó un dispositivo ESP8266 ampliamente difundido en aplicaciones de IoT.

El funcionamiento del equipo responde a la consigna de medir los parámetros ambientales y transmitirlos via WIFI a

una plataforma en la Nube utilizando un protocolo UDP. El firmware embebido en el dispositivo, tiene una rutina principal que realiza una lectura de los sensores conectados por una interface I2C, realizando una actualización de los valores cada 30 segundos.

Por otra, parte se programó una subrutina embebida en el código que realiza la lectura del puerto analógico AN0, conectado a la salida analógica del micrófono disponible en la placa de sensores. La lectura del puerto analógico se realiza como tarea principal del dispositivo. Los valores de la conversión analógica a digital, también se envían vía WIFI con protocolo UDP, al mismo servidor de los sensores. En la Nube, se realizó una aplicación en el Back end de la plataforma (escrita en Python) del sistema de visualización de datos ambientales, que se encarga de tomar los datos provenientes del micrófono, armarlos y reproducirlos por los parlantes de una computadora conectados a la plataforma. La reproducción del sonido obtenido tiene un ancho de banda que varía en función de la calidad de conexión, sin embargo, dado que se logra una buena calidad de audio para la voz humana con anchos de banda del orden de 3Khz, en las pruebas realizadas sobre los sensores ambientales de muestra, se logró una perfecta escucha del sonido ambiente, del lugar en donde está instalado el sensor.

A los usuarios del sistema solo se les da acceso a la parte de la plataforma que presenta los parámetros ambientales, entre estos: intensidad de ondas sonoras presentes. Pero por otro lado, quienes desarrollaron el equipo pueden tener acceso pleno al sonido de la habitación en donde se encuentra instalado el sensor, pudiendo obtener información que no se debería poder acceder.

El experimento descripto tiene como objetivo demostrar que la utilización de determinados dispositivos comerciales requiere la incorporación de principios que cumplan con especificaciones de seguridad desde la fase de diseño, como así también, condiciones de flexibilidad para adaptarse a distintas situaciones de peligro que puedan presentarse respecto de la confidencialidad, la integridad y la disponibilidad de los datos del sistema.

BIBLIOGRAFÍA

- [1] G. IoT Hackers Handbook: An Ultimate Guide to Hacking the Internet of Things and Learning IoT Security 1.0 Edition. 2017- By Aditya Gupta. Packt Publishing Ltd. ISBN-13: 978-1974590124. ISBN-10: 1974590127.
- [2] The Five Technological Forces Disrupting Security: How Cloud, Social, Mobile, Big Data and IoT are Transforming Physical Security in the Digital Age 1st Edition. By Steve Van Till. 2018 Elsevier Inc. ISBN-13: 978-0128050958 ISBN-10: 0128050950
- [3] Aspectos de Seguridad en Internet de las Cosas. Alicia Castro, Eduardo Casanovas, Verónica Gil-Costa. XXIII Congreso Argentino de Ciencias de la Computación. La Plata. Argentina. Octubre 2017.
- [4] Restuccia, F., D'Oro, S., & Melodia, T. (2018). Securing the Internet of Things: New Perspectives and Research Challenges, 1(1), 1–14.<http://arxiv.org/abs/1803.05022>
- [5] Security and Identity for AWS IoT - AWS IoT - AWS Documentation<https://docs.aws.amazon.com/iot/latest/.../iot-security-identity.html>
- [6] IoT Security White Paper 2018 - Huaweihttps://www.huawei.com/.../iot/.../iot_security_white_paper_2018
- [7] IoT Sensor Connectivity and Processing with Ultra-Low Power, Small Form-Factor FPGAs Applications Run the Gamut from Sensor Buffers, Signal Aggregation to Embedded Camera, Audio and Display Solutions. A Lattice Semiconductor White Paper March 2018

Evaluación del Desempeño de una Placa Galileo para la Detección de Rostros en un Robot Autónomo

Miguel Revuelta, Juan López, Juana Fernández, Melisa Kuzman, Raúl Rivera, Walter Gemin.

Universidad Nacional de Mar del Plata, Facultad de Ingeniería,
Depto. de Ingeniería Electrónica y Computación
Mar del Plata, Argentina
mrevuelta@fi.mdp.edu.ar

Resumen— Este artículo describe el desarrollo de una aplicación basada en las librerías de Open CV, para la detección de rostros mediante un script Python, en una placa Galileo. Con el objetivo de evaluar su implementación en sistemas robóticos autónomos, se efectúa un análisis comparativo de los tiempos de detección de rostros en tiempo real, utilizando el mismo script y cámara web, en una PC y una placa autónoma Galileo.

Palabras claves—OpenCV; Intel Galileo; Tiempo Real, Detección de Rostros

I. INTRODUCCIÓN

La detección de un rostro es la primera etapa del reconocimiento facial, que permite a un sistema autónomo, por ejemplo, a un robot que interactúa con personas, relacionarse con el humano reconocido [1] [2] [3].

El uso de las librerías de Open CV [4] para el desarrollo de aplicaciones de detección de rostro en una PC está ampliamente documentado. En este artículo se describe su implementación, en una placa Intel Galileo [5], mediante una aplicación desarrollada en Python [6], para la detección de rostros con el propósito de incorporarla a un robot móvil autónomo.

Inicialmente el desarrollo de la aplicación en una PC permitió observar simultáneamente el streaming de video de la cámara web y el resultado de la detección de rostros para evaluar y depurar el script. Posteriormente, la aplicación se instaló en la placa Galileo que reconoce la región del rostro en la imagen y la guarda en un archivo jpg para su posterior procesamiento y/o reconocimiento.

II. DETECCION DE ROSTROS EN UNA PC

En esta aplicación, para permitir evaluar el script en tiempo real, se emplea el objeto denominado VideoCapture de la librería Open CV para la captura del streaming de video. Luego la función cap.read() retorna un argumento que indica si la lectura del frame fue correcta. Posteriormente la función cv2.imshow muestra en la pantalla de la computadora esta imagen con un recuadro sobre los rostros detectados.

La aplicación imprime el tamaño de los lados del recuadro en píxeles y las coordenadas del centro.

Para evaluar la eficiencia de la aplicación se emplea una imagen fuente conocida, que contiene rostros, denominada en este ejemplo 8-caras.jpg, perteneciente a la galería de fotos de amigos de Google. Se incluyen además librerías para calcular cuantas detecciones se efectúan por segundo y cuánto tiempo en milisegundos (en este caso 175 ms) dura un proceso de detección exitoso a partir de un frame.

III. SCRIPT EN LA PLACA GALILEO

Habitualmente, el script de una aplicación debe estar cargado en la memoria SD de la Galileo y cada cambio que se proponga requiere retirar la SD, modificar el script y reiniciar la placa. En este trabajo se empleó un esquema de control, basado en dos aplicaciones corriendo en la PC, WinSCP y Putty, que permiten editar y modificar el script sin quitar la memoria SD utilizando una conexión Ethernet.

En este caso, dado que no se dispone de monitor, el seguimiento del proceso de detección se imprime en la consola de comunicación (sesión de Putty), que resulta muy útil en la etapa de depuración del script.

En la aplicación prevista, sistema de detección y seguimiento de rostro para un robot autónomo, se configura uno de los puertos COM de la placa Galileo para transmitir esta información al resto del sistema.

Un programa de consola de comunicación en una PC, como muestra la Fig. 1, permite visualizar el resultado de la aplicación sobre la placa.

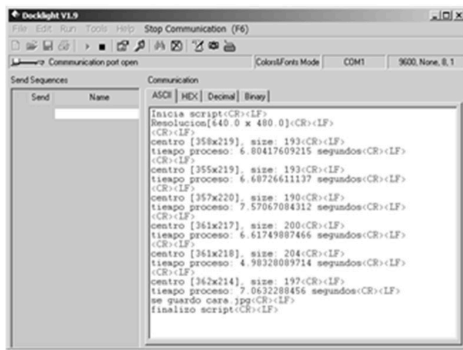


Fig. 1 Resultados generados por el script en la Placa Galileo

Para evaluar el desempeño de la placa Galileo, además de calcular el centro de coordenadas del recuadro sobre el rostro y el tamaño en ancho/alto del mismo, se imprime el tiempo requerido por el proceso de detección, que en este caso resultó de aproximadamente 7 segundos.

Con el objetivo de reducir el tiempo de procesamiento, se combinó un algoritmo que capture un frame, solamente cuando exista un cambio en el streaming y aplique el script descripto. La Fig. 2 muestra el resultado de este proceso, aplicado al frame de 8 caras mencionado, obteniendo una mejora sensible en el tiempo de detección de rostros que resultó de 2,37 segundos.

IV. CONCLUSIONES

En la placa Galileo el tiempo de proceso para detectar un rostro en un streaming insueme alrededor de 7 segundos, que se puede reducir sensiblemente cuando se aplica a un frame. Esto implica que si se emplea un preprocesamiento, que detecte modificaciones en la información del video, para habilitar la captura de un frame se puede aplicar el script descripto y detectar la presencia de uno o varios rostros en 2,37 segundos.

Esto permite establecer las reglas de comportamiento del robot para interactuar, prácticamente en tiempo real, con las personas y a la vez llevar un registro de aquellas con las que se relacionó.

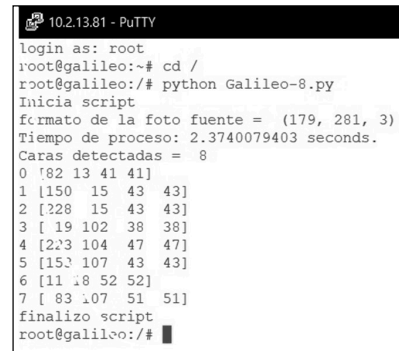


Fig. 2 Visualización de resultados del script con captura previa del frame

Cada vez que el script envíe por el puerto COM la información de la presencia de uno o varios rostros habilita al resto del sistema para la toma de decisiones en cada paso particular, como puede ser la interacción e identificación inmediata mediante el acceso a una base de datos.

La posterior aplicación de un proceso de reconocimiento, empleando una base de datos de rostros conocidos, puede ocurrir localmente (ejecutando el script de reconocimiento en la placa Galileo) o externamente, lo que requerirá conectividad a internet de la placa para enviar al sistema externo la imagen del rostro detectado.

Para una aplicación de detección más rápida, se podría desarrollar el script en C++ [7] y compilarlo para la placa Galileo en lugar de emplear el código interpretado utilizado en este trabajo.

REFERENCIAS

- [1] A. de la Escalera Hueso, "Visión por Computador. Fundamentos y Métodos", Pearson Educación, 2001.
- [2] J. Vélez Serrano, A. Moreno Diaz, A. Sánchez Calle, J. Sánchez Marín, "Visión por Computador", Ed. DYKINSON, Madrid España, 2003.
- [3] <http://www.visionporcomputador.es/libroVision/VisionPorComputador.pdf>
- [4] https://docs.opencv.org/master/db/d28/tutorial_cascade_classifier.html
- [5] <https://software.intel.com/es-es/iot/documentation>
- [6] https://opencv-python-trools.readthedocs.io/en/latest/py_tutorials/py_gui/py_video_display/py_video_display.html
- [7] A. Kaehler, G. Bradski, "Learning Open CV3: computer vision in C++ with the OpenCV Library", O Reilly Media Inc.,

Navegación autónoma en interiores mediante marcadores para Vehículos Aéreos No Tripulados

Juan Pablo Mercol
Gerente del laboratorio de innovación en NEORIS
Buenos Aires, Argentina
juan.mercol@neoris.com

Manuel Alfonso
Especialista en drones e imágenes en NEORIS
Buenos Aires, Argentina
manuel.alfonso@neoris.com

Hernán Zaldivar
Gerente del laboratorio de innovación en NEORIS
Rosario, Santa Fe, Argentina
hernan.zaldivar@neoris.com

César Rodríguez
Director del SAP Development Center Director en NEORIS
San Nicolás, Bs. As, Argentina
cesar.rodriguez@neoris.com

El avance tecnológico de los vehículos aéreos no tripulados (VANT) o “drones” ha devenido en numerosas aplicaciones en el ámbito profesional, donde la navegación autónoma surge como la más innovadora y disruptiva. En el caso de vuelos autónomos en exteriores, la posibilidad de definir la ubicación mediante un GPS incorporado facilita considerablemente la tarea para cierto tipo de misiones. Sin embargo, para vuelos en interiores sin GPS, la determinación de la ubicación debe realizarse por otros medios, lo cual dificulta enormemente la navegación. Debido a ello, los vuelos en interiores constituyen un importante tema de análisis. Existen diversas alternativas para el vuelo autónomo en interiores, como por ejemplo el uso de radiotransmisores para triangular la posición, el uso de mapeo y localización simultánea, odometría basada en sensores o imágenes, o el uso de marcadores y líneas guía.

Un vuelo autónomo propone automatizar las acciones de vuelo en base a la aplicación de conceptos de Inteligencia Artificial, relegando al piloto las tareas de supervisión e intervención únicamente ante eventuales fallas en la operación. Más aún, un área que todavía se encuentra en evolución, es la de vuelos en interiores donde la problemática principal reside en no poder utilizar el GPS para determinar la ubicación, imposibilitando vuelos autónomos basados en coordenadas de referencia.

Otra característica de los vuelos en interiores es la necesidad de mayor estabilidad del drone, en conjunto con una mayor conciencia del espacio circundante, de manera de evitar obstáculos. Esto generalmente es logrado añadiendo una serie de sensores, como pueden ser sensores de altitud (Barómetro, Lidars, ultrasónicos), sensores de estabilización axial como pueden ser cámaras cenitales que realizan el algoritmo conocido como “Optical Flow” [1]

En el presente trabajo, se realiza un vuelo autónomo utilizando la cámara como principal sensor, y el uso de marcadores (QR, ARUCO o líneas guía) para establecer la ubicación y acciones a realizar. El desarrollo se llevó a cabo en lenguaje Python por su simplicidad, utilizando las librerías OpenCv para procesar la imagen y pyParrot para comandar, mediante Wifi, un drone Parrot Bebop 2

De manera de obtener mayor versatilidad e integración con la industria, se decidió utilizar códigos QR como marcadores, se estudió la literatura actual y las distintas bibliotecas posibles a utilizar [2] decantando por la biblioteca PyZBar. Luego se realizaron diferentes pruebas para determinar la mínima y máxima distancia de manera de obtener un alto grado de detección en relación a los distintos tamaños de etiquetas y resoluciones del streaming de video. En dichas experiencias se varió: **a)** El tamaño de impresión,

b) Resolución streaming, **c)** Distancia de detección, **d)** Reflectividad en la impresión. **e)** Ángulo de detección.

El plan de vuelo consiste en la definición de una serie de “nodos” a visitar, descritos por un código identificatorio (QR), una dirección y distancia hacia el próximo nodo. Una vez iniciado el despegue, se comienza con el streaming de video. Desde la PC se procesa el mismo hasta detectar el primer marcador, que debe corresponderse con el primer nodo. En esta instancia se envían los comandos de movimiento para que el VANT se aproxime al próximo nodo. El ciclo se repite hasta que se llegue al último nodo, donde se ejecuta el comando de aterrizaje.

De manera de realizar “acciones” frente a cada marcador y probar el movimiento fino, se estableció que antes de avanzar al próximo nodo, el marcador debe centrarse dentro de una “región” determinada, por más de N cuadros. Para las primeras pruebas se utilizó el plan de vuelo descrito anteriormente conformado por cuatro marcadores dispuestos en forma de cuadrado, y un recorrido en sentido horario como se puede observar en la Fig. 1

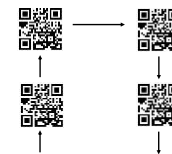


Fig. 1 Secuencia de vuelo

En las pruebas realizadas, se detectaron los siguientes inconvenientes: **a)** Interferencias electromagnéticas, que afectan al magnetómetro interno y desorientan al drone. **b)** Viento producido por el VANT: al volar en espacios reducidos produce turbulencia que provocan desviaciones en la trayectoria de vuelo. **c)** Deriva producida por falta de textura del piso, solucionándolo mediante líneas en el suelo.

Como estudios a futuro, se propone diseñar un plan de vuelo con mayor cantidad de nodos, como así también añadir estimación de posición, lo que redundaría en posibilidad de utilizar “waypoints” para el lazo de control. Además, se estima añadir corrección de postura frente a las irregularidades presentadas anteriormente.

Una de las vertientes a analizar es la posibilidad de contar con un drone no comercial, y ligado al mismo un sensor de posicionamiento para interiores. Una ventaja adicional del drone no comercial, es la posibilidad de contar con una computadora de a bordo para realizar el procesamiento, disminuyendo el tiempo de latencia.

REFERENCIAS

- [1] Horn, Berthold KP, and Brian G. Schunck. "Determining optical flow." *Artificial intelligence* 17.1-3 (1981): 185-203.
- [2] Peter Abeles (2019). Study of QR Code Scanning Performance in Different Environments.

A. Selección de tecnología

Primeramente, para la selección del VANT a utilizar se estudiaron las diferentes alternativas comerciales, optándose por los Drones Parrot Bebop 1 y Bebop 2. Debido al tamaño reducido, menor nivel de ruido y mejor estabilidad. Una segunda ventaja al elegir esos Drones, fue la posibilidad de contar con una librería de Python para comandarlos [6], que es un port del SDK oficial. La principal ventaja de este esquema es la velocidad y agilidad de desarrollo y la disminución de tiempos entre un cambio de código y las pruebas en el VANT. Finalmente, se diseñó un programa en Python que se conecta mediante Wifi al Bebop, y por este medio envía los comandos y recibe el estado de sensores y streaming de video el cual es procesado utilizando la conocida librería OpenCV [7].

B. Detección de marcadores (QR o Aruco)

De manera de obtener mayor versatilidad e integración con la industria, se decidió utilizar códigos QR como marcadores, se estudió la literatura actual y las distintas bibliotecas posibles a utilizar [8], decantando por la biblioteca PyZBar. Luego se realizaron diferentes experimentos para determinar la mínima y máxima distancia de manera de obtener un alto grado de detección en relación a los distintos tamaños de etiquetas y resoluciones del streaming de video. En dichas experiencias se varió: **a)** El tamaño de impresión del marcador. **b)** Resolución streaming **c)** Distancia de detección. **d)** Reflectividad en la impresión. **e)** Ángulo de detección.

C. Plan de vuelo

El plan de vuelo consistió en la definición de una serie de “nodos” a visitar, descritos por un código identificatorio (QR), una dirección y distancia hacia el próximo nodo

Una vez iniciado el despegue, se comienza con el streaming de video. Desde la PC se procesa el mismo hasta detectar el primer marcador, que debe corresponderse con el primer nodo. En esta instancia se envían los comandos de movimiento para que el VANT se aproxime al próximo nodo. El ciclo se repite hasta que se llegue al último nodo, donde se ejecuta el comando de aterrizaje.

De manera de realizar “acciones” frente a cada marcador y probar el movimiento fino, se estableció que antes de avanzar al próximo nodo, el marcador debe centrarse dentro de una “región” determinada, por más de N cuadros, como se observa en la Fig. 1



Fig. 1. Detección de QR, y distancia a recorrer

III. RESULTADOS

En la presente sección se exponen los resultados obtenidos en relación a las diferentes experiencias pre o post vuelo, como así también la experiencia resultante de la ejecución del plan de vuelo.

A. Detección de QR

En base a las pruebas realizadas se determinó la distancia mínima y máxima para cada tamaño de impresión del marcador y resolución utilizada (Tabla 1). Evaluando las diferentes posibilidades, se descartaron las impresiones de 3 y 5 cm, ya que necesitarían que el drone vuele demasiado cerca de los marcadores (y por ende objetos) para una correcta detección. Finalmente se decidió utilizar marcadores de 10 cm los cuales proveen una máxima distancia y seguridad para la detección

Tamaño del QR	480p	720p
10 cm	110 cm	130 cm
8 cm	75 cm	90 cm
5 cm	45 cm	50 cm

Tabla 1 Relación entre resolución, tamaño de QR y distancia máxima

En todas las pruebas se pudo obtener una excelente tasa de detección para ángulos de hasta 50 grados, desde la perpendicularidad (90°).

B. Vuelo

Para las primeras pruebas se utilizó el plan de vuelo descrito anteriormente conformado por cuatro marcadores equiespaciados dispuestos en forma de cuadrado, y un recorrido en sentido horario como se puede observar en la Fig. 2.

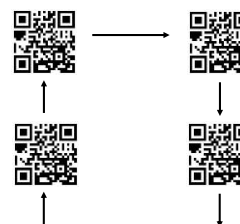


Fig. 2 Secuencia de vuelo

IV. CONCLUSIÓN

Se logró realizar un vuelo autónomo simple en interiores basado en marcadores de forma satisfactoria, pudiendo cumplir el objetivo del presente trabajo. En las pruebas realizadas, se detectaron los siguientes inconvenientes. **a)** Interferencias electromagnéticas, que afectan al magnetómetro interno y desorientan al drone. **b)** Viento producido por el VANT: al volar en espacios reducidos produce turbulencia que provocan desviaciones en la trayectoria de vuelo. **c)** Deriva producida por falta de textura del piso. Generalmente se pueden solucionar con líneas en el suelo.

Como estudios a futuro, se propone diseñar un plan de vuelo de mayor alcance (cantidad de nodos). Como así también añadir estimación de posición mediante alguna de las técnicas seleccionadas, lo que redundaría en posibilidad de utilizar “waypoints” para el lazo de control. Además, se estima añadir corrección de postura frente a las irregularidades presentadas anteriormente.

Una de las vertientes a analizar es la posibilidad de contar con un drone no comercial, y ligado al mismo un sensor de posicionamiento para interiores. Una ventaja adicional del drone no comercial, es la posibilidad de contar con una computadora de a bordo para realizar el procesamiento, disminuyendo el tiempo de latencia.

REFERENCIAS

- [1] ICAO Cir 328, Unmanned Aircraft Systems (UAS), Order Number: CIR328, ISBN 978-92-9231-751-5.
- [2] Artale, V & Ricciardello, Angela & Milazzo, C. (2013). Mathematical modeling of hexacopter. Applied Mathematical Sciences. 7. 4805 - 4811. 10.12988/ams.2013.37385.
- [3] Horn, Berthold KP, and Brian G. Schunck. "Determining optical flow." Artificial intelligence 17.1-3 (1981): 185-203.
- [4] Flávio de Melo, Leonimer & Mauricio Rosário, João & Franco da Silveira Junior, Almiro. (2013). Mobile Robot Indoor Autonomous Navigation with Position Estimation Using RF Signal Triangulation. Positioning. 04. 20-35. 10.4236/pos.2013.41004.
- [5] Yousif, K., Bab-Hadiashar, A. & Hoseinnezhad, R. Intell Ind Syst (2015) 1: 289. <https://doi.org/10.1007/s40903-015-0032-7>
- [6] A. McGovern, pyParrot library: <https://pyparrot.readthedocs.io>
- [7] Kari Pulli, Anatoly Baksheev, Kirill Korniyakov, and Victor Eruhimov. 2012. Realtime Computer Vision with OpenCV. Queue 10, 4, Pages 40 (April 2012), 17 pages.
- [8] Peter Abeles (2019). Study of QR Code Scanning Performance in Different Environments. <https://boofcv.org/index.php?title=Performance:QrCode>.

Actualización de un robot industrial de 6 grados de libertad

Con PID digitales embebidos en un uC ARM Cortex-M3

Luis Javier Vazquez¹

German Ghigi¹

Santiago Rojas¹

¹ Centro Universitario de Desarrollo en Automatización y Robótica (CUDAR),
Universidad Tecnológica Nacional - Facultad Regional Córdoba

El proyecto surge como propuesta para disponer de un sistema que pueda ser utilizado en el ámbito académico, con el objeto de contar con una plataforma robótica funcional, mediante la actualización de un brazo robot disponible.

Se cuenta con la estructura mecánica funcional de un brazo industrial de 6 grados libertad, sin módulos de control ni drivers de potencia para controlar los motores de corriente continua que posee. El brazo tiene 3 ejes principales y 3 secundarios pensados para darle precisión a la muñeca del robot.

El objetivo de esta actualización es que sea posible controlar cada eje del robot, de manera independiente, utilizando una PC, sin realizar un control de trayectoria espacial, tema que se desarrollará en trabajos futuros.

Para lograr este objetivo se debe desarrollar la electrónica de potencia y el módulo de control de cada eje, además de la comunicación del Robot-PC, tema a tratar en este resumen.

Todos los motores del robot cuentan con un encoder incremental y además dos de los principales cuentan con un taco generador. Se ha optado por descartar el taco generador, y utilizar el encoder como realimentación de velocidad y posición.

Para la actualización del sistema, se han desarrollado los drivers de cada motor DC, un puente-H con MOSFETs y protección por sobrecorriente. Además un módulo de control para cada motor, donde se utilizó un microcontrolador ARM Cortex-M3 de la empresa NXP LPC1768 para embeber el firmware de control y comunicación serie RS-485.

Se han utilizado herramientas FOSS [1] para el desarrollo de software. El entorno de programación es Eclipse con el plugin GNU MCU, que dispone de OpenOCD para el debug y el GNU toolchain para ARM Cortex-M. Además de utilizar GIT para el control de versiones del software y permitir el trabajo en equipo presencial y online.

Para el control se utiliza un controlador PID [2][3] discreto y anidado esto es, un control de velocidad dentro de uno de

posición. Estos dos se realimentan por medio del encoder correspondiente al motor que controlan. Se utiliza un control proporcional para el lazo de posición y para el lazo interno de velocidad se implementa un lazo proporcional integral con anti wind-up. Estos lazos serán sintonizados por diferentes procedimientos[4].

Para la comunicación entre Robot-PC se utiliza una interfaz RS-485. El camino elegido es disponer de un control central de comunicación que receptorá y comandará a cada motor individual, además de ser la interfaz con la PC. De esta manera desde la PC será posible enviar comandos de velocidad o posición a cada motor de manera independiente. La interfaz de la PC está escrita en Python, y permite la recepción de tramas enviadas por el control para debug y visualización de la respuesta temporal de los PID.

CONCLUSIONES

En la actualidad ya se han probado con éxito dos de los motores de la muñeca del robot y un motor del eje principal con el lazo de control propuesto, además de la comunicación con la interfaz de la PC. Es posible visualizar también las respuestas del PID de Velocidad y Posición en un plot 2D en la PC. A futuro se implementará el control en los 6 motores, y se mejorará la interfaz de la PC donde se podrá enviarle posición objetivo a cada eje del robot independientemente.

AGRADECIMIENTOS

Agradecemos al Ing. Picco Juan, Ing. Gutierrez Guillermo, Ing. Olmedo Sergio por la información brindada sobre el robot, también por su guía y consejo a la hora de resolver conflictos.

REFERENCIAS

- [1] FOSS. https://en.wikipedia.org/wiki/Free_and_open-source_software
- [2] K. Ogata, Ingeniería de control moderna, 5ta Ed. Prentice Hall, 2010, pp 567-806.
- [3] B.C Kuo, Sistemas de control automático, 7ma Ed. Prentice Hall, 1996
- [4] C. S. Mala & S. Ramachandran, "Design of PID Controller for Direction Control of Robotic Vehicle". Global Journal of Researches in Engineering, Vol. 14, Issue 3,2014

Congreso Argentino de Sistemas Embebidos
CASE 2019
Libro de trabajos en modalidad artículo, foro tecnológico y resumen

ISBN 978-987-46297-6-0

