Ingeniería y Automatización

Vargas-Soto José Emilio
Aceves-Fernández Marco Antonio
Orozco-Ramírez Jorge Enrique

Ingeniería y Automatización

1ª Edición

Vargas-Soto José Emilio
Aceves-Fernández Marco Antonio
Orozco-Ramírez Jorge Enrique

“Ingeniería y Automatización”
Vargas-Soto José Emilio
Aceves-Fernández Marco Antonio
Orozco-Ramírez Jorge Enrique

© Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
México.

Internet: http://www.mecamex.net

1ª Edición, 12 de Octubre del 2018

ISBN 978-607-9394-15-8

Derechos reservados.
Esta obra es propiedad intelectual de sus autores y los derechos de publicación han sido legalmente transferidos a la editorial. Las opiniones y la información que se muestran en los capítulos del libro son exclusivas de los autores de cada capítulo y no representan la postura de la Asociación Mexicana Mecatrónica A.C. Prohibida su reproducción total o parcial por cualquier medio sin la autorización por escrito del propietario de los derechos del copyright.

Impreso en México – Printed in Mexico

.

Prólogo

La Asociación Mexicana de Mecatrónica A.C. tiene como misión impulsar el conocimiento y el desarrollo de la Ingeniería Mecatrónica a través de vincular empresas e industrias con instituciones de educación superior, esto con el propósito de buscar elementos de colaboración para difundir, actualizar y/o desarrollar innovaciones tecnológicas orientadas a generar beneficios en la sociedad. En este sentido, la presente obra intelectual se ha preparado durante varios meses a fin de celebrar el talento y la creatividad de entusiastas de la Ingeniería Mecatrónica.

En este libro, el lector encontrará información técnica relevante sobre diversos proyectos de robótica y mecatrónica en diferentes áreas del conocimiento, como son: manufactura automatizada, educación y docencia, energías alternativas y renovables, procesamiento de señales, modelación y simulación de máquinas mecatrónica, sistemas embebidos y desarrollo de prototipos, entre otras.

Confío en que el trabajo aquí presentado sea de utilidad a quienes de alguna forma se encuentran estudiando, investigado o bien realizando un trabajo profesional en donde los robots y las máquinas mecatrónicas propician beneficios en la sociedad.

Lic. Jorge Enrique Orozco Ramírez
Presidente
Asociación Mexicana de Mecatrónica A.C.

ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Índice

Capítulo 1 1
Co-Simulación de un Omni-Phantom para Control
Verde-Lampon Saraí, Gudiño-Márquez Jorge W., Gudiño-Lau Jorge,
Alcalá-Rodríguez Janeth, Charre-Ibarra Saida y Durán-Fonseca Miguel.

Capítulo 2 15
Análisis de Esfuerzos en Dientes de Engranes Rectos,
Mediante Metodologías AGMA y CAE
Hernández Corona Francisco, Mora Santos Carlos Alberto,
Bedolla Hernández Marcos, Flores Lara Vicente y Bedolla Hernández Jorge.

Capítulo 3 26
Descomposición de Trayectoria Óptima para Robots
Móviles Generada por PSO y Descriptores de Forma
Méndez Guzmán Hugo Antonio, Flores Muñoz Víctor Hugo
y Solano Blancarte Miguel Ángel.

Capítulo 4 45
Propuesta de Robot Serial de Servicio para
Tender Camas de Hospital
Lasso Perdomo Cristiam Javier, Sastoke González Stephany,
Peña Medina Erik y Neftali Elorza Lopez.

Capítulo 5 59
Análisis y Diseño de Control para la Corrección de
Ángulo Pitch en Pala de un Aerogenerador de Baja Potencia
Sánchez Luna David, Aguilar Anastacio José Carlos
y Muñoz Flores Luis David.

Capítulo 6 77
Diseño de un Horno para Laminar Vidrio de Luz Inteligente
Mora Mora Omar, Mora Santos Carlos Alberto,
Hernández Corona Francisco, Bedolla Hernández Jorge
y Cuatecontzi Xochitiotzi Desiderio.

Capítulo 7 87
Prototipo de Lectura y Escritura en Código Braille
y Bastón para Ciegos
Ortega Blanco Christian Aarón, Corral Quintanar Álvaro Martín,
Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen
y Platt Borbón Martina Elisa.

Índice

Capítulo 8 101
Interfaz de Usuario y Sistema de Navegación para un Vehículo Eléctrico para Discapacitados
Delgado Arcega Rodrigo Oliver, Robles Julio Cesar y Gómez Becerra Fabio Abel.

Capítulo 9 111
Automatización de Grippers del Laboratorio de Robótica
López Luna Alma Rosa, Leon Martínez Martin Francisco, Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen y Platt Borbón Martina Elisa.

Capítulo 10 121
Diseño y Construcción de un Prototipo Gripper Industrial
Amavizca Ramírez Miguel, Verdugo Neudert Oswaldo Alfonso, Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen y Platt Borbón Martina Elisa.

Capítulo 11 136
Desarrollo de una Interfaz para la Generación de MegaDatos del Desempeño de Automóviles
Sánchez Cuapio Iván Jesús.

Capítulo 12 147
Filtro Electrostático de Alta Tensión para la Eliminación de Partículas en un Quirófano con Alarma
García Juárez Ricardo Aarón, Pérez Pamatz Iván, Nicolás Máximo Ángel Eduardo, Nicolás Máximo Santiago y Ramírez Rodríguez Eduardo.

Capítulo 13 159
Implementación de Algoritmos para el Monitoreo de Puentes en Tiempo Real
Martínez Trujano Luis Álvaro, Hernández Figueroa Jorge Alberto, Gasca Zamora Héctor Miguel, Quintana Rodríguez Juan Antonio y Hernández Guzmán Andrés.

Capítulo 14 176
Diseño y Validación de un Prototipo Robótico Continuum Multisección
Castro Riveros Fabian Camilo y Perilla Castaño Victor Mario.

ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Índice

Capítulo 15 188
Diseño de un Robot Planar para Asistencia en Terapias de Rehabilitación
Chávez-Olivares César, Zacarías-Mena Juan Antonio, Cervantes-Herrera Alejandro, Salinas-Martínez Jorge y Arámbula-Miranda Luis Enrique.

Capítulo 16 205
Diseño y Plan de Manufactura de una Plataforma Experimental para Control: Péndulo Invertido Inercial
Hernández Gallardo Julián Alejandro, Pacheco Villegas Jesús Eduardo, Désiga Orenday Omar, Talavera Otero Jorge y Cuevas Cortés Carlos.

Capítulo 17 224
Instrumentación y Control para Tiempos de Inmersión y Flujo Fotónico en Acondicionamiento in vitro de Plantas Mediante Biorreactores de Inmersión Temporal
Durán Lugo Juan M., Quej Cosgaya Hector M. y Valladares Castellanos Manuel A.

Capítulo 18 235
Diseño de un Robot Subacuático para la Rehabilitación de Tobillo: Mecanismo y Control de Temperatura
Madrigal López Oscar, Désiga Orenday Omar, Talavera Otero Jorge y Guzmán Valdivia Cesar H.

Capítulo 19 245
Measurments of the Friction Coefficient for the Vehicles Ball joints, Using a Mechatronic Measure Instrument
Wozniak Marek, Siczek Krzysztof, Ozuna Gustavo y Tapia Pedro.

Capítulo 20 255
Diseño y Construcción de un Prototipo de Manipulador Robótico de 4 Grados de Libertad para Fines Educativos
Carmona Ruiz Deisy, Carrasco López Rosaura, Rebeca Itai Cruz Pacheco, Petrilli Barceló Alberto Elias y Lugo González Esther.

Capítulo 21 272
Control de Velocidad de un Robot Rueda con Locomoción Inercial por Péndulo
Herrera Cordero Mario Enrique, Arias Montiel Manuel y Lugo González Esther.

ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Índice

Capítulo 22 287
Verificación e Implementación en Hardware del Código de Redundancia Cíclica
Soto Hernández Enrique, Ramírez Villa Goretti, Cházaro Zaharías Adriana C., Traslosheros Michel Alberto y González Chávez Miguel A.

Capítulo 23 300
Propuesta de Modelo en una Microred Eléctrica Basado en un Sistema Híbrido ante una Falla Eléctrica Detectada
Méndez Navarro Alejandro, Rodríguez Hernández Jose Jorge, Ávalos García Jorge Hugo y Villalobos Guerra Arnoldo Ulises.

Capítulo 24 308
Generación de Consignas con Perfil Quintico para el Control del Robot UR5
Noé Alfredo Martínez Sánchez, Erik Peña Medina, Patricio Martínez Zamudio, Octavio Diaz Hernández y José Antonio Sousa Jiménez.

Capítulo 25 322
Aplicación de Sistemas Embebidos e IoT para el Monitoreo de Estanques Acuícolas en Eldorado, Sinaloa
Nava Pintor Jesus Antonio, Guerrero Osuna Héctor Alonso, Ornelas Vargas Gerardo, Solís Sánchez Luis Octavio y Miranda Castañeda Rodrigo.

Capítulo 26 336
Plataforma de Bajo Consumo de Energía para la Capacitación en Sistemas Satelitales
González Chávez Miguel Alejandro, Ramírez Villa Goretti, Traslosheros Michel Alberto y Soto Hernández Enrique.

Capítulo 27 352
Algoritmo de División Entera para Aplicaciones en Hardware Difuso
César Barrón Romero y Antonio Hernández Zavala.

Capítulo 28 362
Prototipo de un Clasificador de Piezas Basado en un Brazo Robótico con Sistema de Transporte y Paletizado
Hernández Mosqueda Carlos, Pérez Chimal Rosa Janette, Luna Puente Rafael, Prieto Figueroa Araceli y Manjarrez Mendoza Javier.

ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Índice

Capítulo 29 372
Rocker-Bogie del Espacio a la Tierra
Gutiérrez Castillo Rodrigo Uzziel y José Reyes Rosales.

Capítulo 30 383
Optimización de la Detección de Altitud/Orientación Utilizando un Sensor Monocular para la Coordinación de dos Drones
Arias Méndez Nelson Edgardo, Jonatán Muñoz Álvarez, Escareño Castro Juan Antonio, Palomino Merino Amparo Dora y María Aurora Diozcora Vargas Treviño.

Capítulo 31 393
Control Simultáneo de Fuerza y Posición para Robot Manipulador en Contacto con una Superficie Desconocida
Domínguez Fuentes Juan Ramón y García Valdovinos Luis Govinda.

Capítulo 32 403
Robótica Competitiva en RUNIBOT Colombia, Aprendizaje Significativo
Merchán Vaquiro Carlos Arturo y Burgos Hernández María Cristina.

Capítulo 33 414
Incubadora con Agitación Orbital
Téllez Cabrera Ricardo Alberto, Carbajal Cruz Alain Israel, Juárez Aparicio Jesús Albín, Cruz Almaraz Viridiana y Gaspar Antonio Lizbeth.

Capítulo 34 424
Desarrollo de un Escáner 3D Basado en Visión Estereoscópica
Cornejo Guerra Paulo Sergio, Traslosheros Michel Alberto, Chagolla Gaona Hernando, Castellanos Galindo Joaquín y Martínez Garrido David.

Capítulo 35 434
Diseño e Implementación de un Sistema de Control para Pruebas de Soldadura de Plasma
Salas-Llanes Luis Arturo, Ramírez-Torres Flor, Cerón-Franco Aureliano, Montijo-Valenzuela Eliel Eduardo y Rodríguez Sánchez Saúl.

Capítulo 36 443
Aplicación Móvil para el Control de Asistencia a Eventos, utilizando Tecnología RFID
Torres Frausto David Antonio, Villegas Téllez Rodrigo, Lozano Vázquez Ángel Gerardo y Serrato Mares Lía Alejandra.

Índice

Capítulo 37 456
Estación Meteorológica de Bajo Costo Implementada con PSoC5LP y Comunicación por Bluetooth
Julián R. Camargo L. y César A. Perdomo Ch.

Capítulo 38 467
Sistema de Prueba de Sensores Analógicos y Digitales Utilizando un PSoC5LP
César A. Perdomo Ch. y Julián R. Camargo L.

Capítulo 39 481
Control PID a Motor sin Escobillas
Ortega Orta Aldo Augusto, Ley Arteaga Kevin Alfonso, Sosa Sales Jorge, Gómez Guzmán César Alberto y Llamas Negrete Juan Carlos.

Ingeniería y Automatización, Capítulo 1, pp. 1–14.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018

Co-Simulación de un Omni-Phantom para Control

Verde-Lampon, Saraí, Gudiño-Márquez, Jorge W., Gudiño-Lau, Jorge,
Alcalá-Rodríguez, Janeth, Charre-Ibarra, Saida, Durán-Fonseca, Miguel.

Facultad de Ingeniería Electromecánica.
Universidad de Colima
Km. 20 carretera Manzanillo-Barra de Navidad
Manzanillo, Colima
E-mail: jgudino1@ucol.mx, jglau@ucol.mx, janethalcala@ucol.mx,
scharre@ucol.mx, mduran@ucol.mx

Resumen

El presente trabajo presenta la co-simulación como una herramienta de apoyo para simulación y control de robots manipuladores, con el objetivo de validar los mecanismos y algoritmos de control antes de llevarlos a la validación experimental. Se diseña una plataforma de co-simulación que tiene un robot manipulador con el apoyo de los softwares Solidwork, Simmechanics y Matlab. El empleo de Solidwork permite hacer un análisis y validación del modelo cinemático y dinámico del robot Omni Phantom y para resolver el problema de control del robot se utiliza Matlab y Simmechanics. Además, se describe como se aplicada la co-simulación para hacer un ambiente virtual, y como se unen los eslabones y las articulaciones hasta forma el robot manipulador Omni Phantom, por lo anterior, la co-simulación ayuda al diseño de robots y al control, ya que se necesitan validar los algoritmos de control en co-simulación antes de llevarlos a la validación experimental y no dañar el equipo experimental.

Palabras clave: Cosimulación, diseño mecánico, omni phantom, modelado de robots y control.

1. Introducción

Actualmente la co-simulación ha tenido mucho interés en el área educativa y por parte de los investigadores, ya que es una herramienta que permite obtener resultados de manera más fácil y exactos, sin necesidad de tener el sistema dinámico de forma física, en este trabajo de investigación se redactan los avances y desarrollos más sobresalientes obtenidos hasta el momento en la literatura.

En 2007 se introduce un método de co-simulación propuesto para resolver con precisión la dinámica de un robot TPT 3 paralelo, la ecuación de la dinámica de los sistemas de acoplamiento es dispuesta por la ecuación de Lagrange, sin embargo, resulta muy complejo y difícil deducir una solución numérica, por lo que el método de la co-simulación con ADAMS y ANSYS es propuesto, la dinámica se cumple en el análisis de ADAMS y el esfuerzo en ANSYS [1]. Esto permite observar mejor los resultados gracias a la co-simulación, esta técnica presentada es muy cercana a la realidad por lo que los resultados de la co-simulación son más auténticos y pueden reflejar una visión más real de la dinámica característica del robot delta 3-TPT, en la figura 1 se muestra el robot TPT.

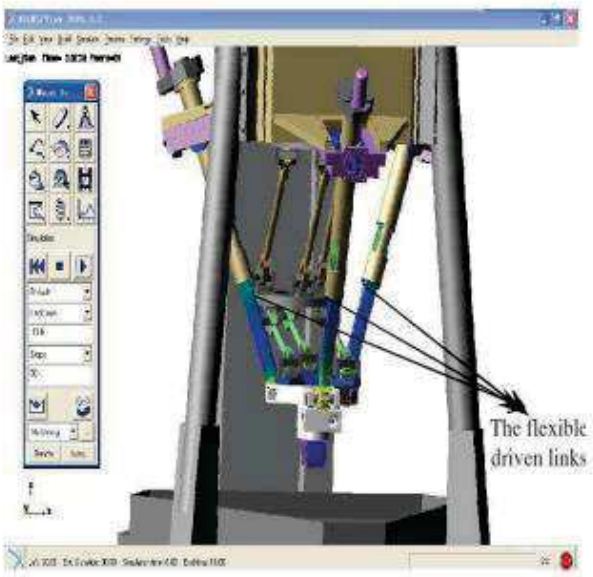


Figura 1. Robot Delta 3-TPT.

En 2011 se presenta el modelado dinámico y la simulación de un robot industrial staubli© TX40 véase en la figura 2, utilizando MATLAB / ADAMS en Co-simulación donde se propone precisar un simulador para desarrollar enfoques para la simulación cinemática, dinámica y análisis de control. El modelado del robot se desarrolla tan exacto como uno real por medio de ADAMS, software de modelado dinámico, y vinculado con MATLAB para los estudios de movimiento [2].

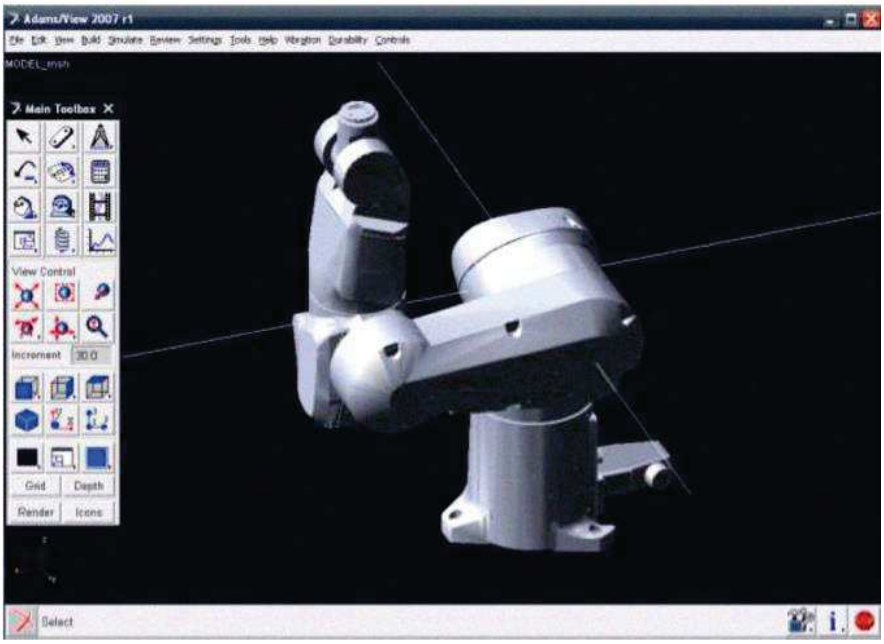


Figura 2. Robot Staubli TX40.

En 2012 se desarrollan varios modelos MBS de tractores, implementos agrícolas y cosechadoras combinadas. Los modelos son construidos en el software SIMPACK® y utilizan el Modelo de Llantas de Hohenheim desarrollado en Matlab/Simulink. En [3] muestra los beneficios que tiene el implementar el software SimMechanics, en la figura 3 se muestra la co-simulación.

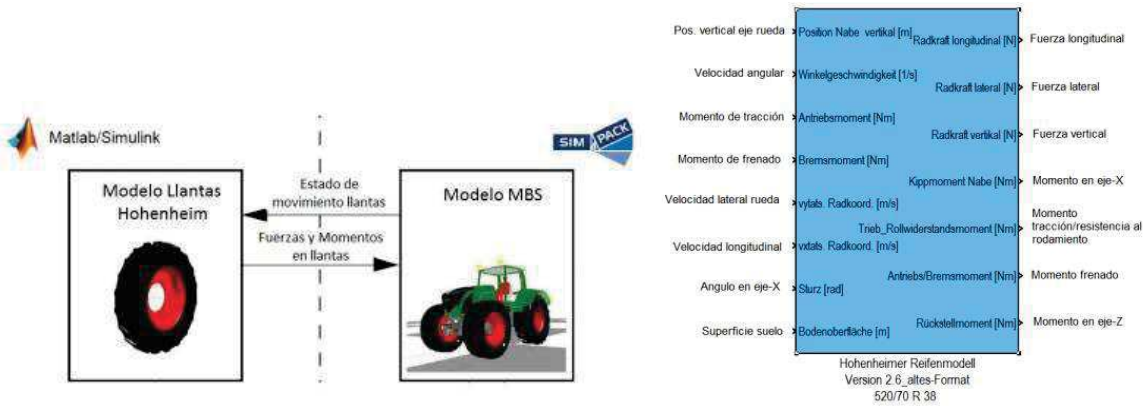


Figura 3. Co-simulación de la llanta de tractor.

En ese mismo año, se realiza un proyecto para el control de posición de un robot industrial a través de la Co-Simulación [4]. Se realiza un modelado virtual de las piezas mecánicas que componen el Robot del caso de estudio (Robot GRYPHON de 5 grados de libertad) en los entornos virtuales CATIA Y SOLIDWORK. También se realiza una simulación de la dinámica del robot en el software ADAMS y se obtiene un sistema con todos datos del prototipo. Seguidamente con ayuda de MATLAB se programa un planificador de trayectorias con el cual se prueban las estrategias de control de posición del modelo y se hace la co-simulación ADAMS /MATLAB del sistema, como se muestra en la figura 4.

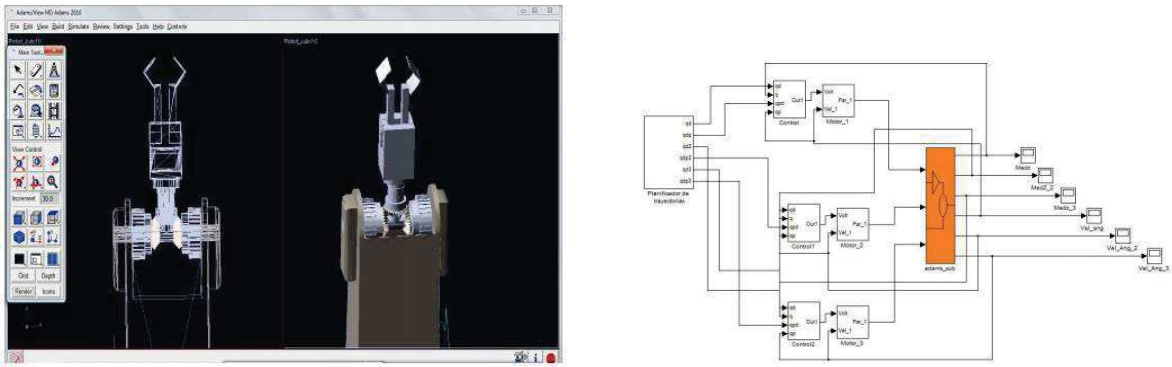


Figura 4. Co-simulación de un sistema de engranes de un robot.

En 2014 se lleva a cabo un proyecto de diseño y co-simulación de un exoesqueleto implementando el uso de los siguientes softwares. SolidWorks® que es aplicado en diseños asistidos por computadora evalúa la parte mecánica y Matlab® resuelve el sistema de control del exoesqueleto. Esto permite conseguir un diseño personalizado, que simula los movimientos de una marcha completa cubriendo las restricciones cinemáticas para lograr un movimiento natural y las limitaciones del usuario cuando tienen algún problema para caminar [5]. En la figura 5 se muestra el trabajo presentado, que es la integración de varios programas de cómputo para el diseño y simulación de prototipos virtuales.

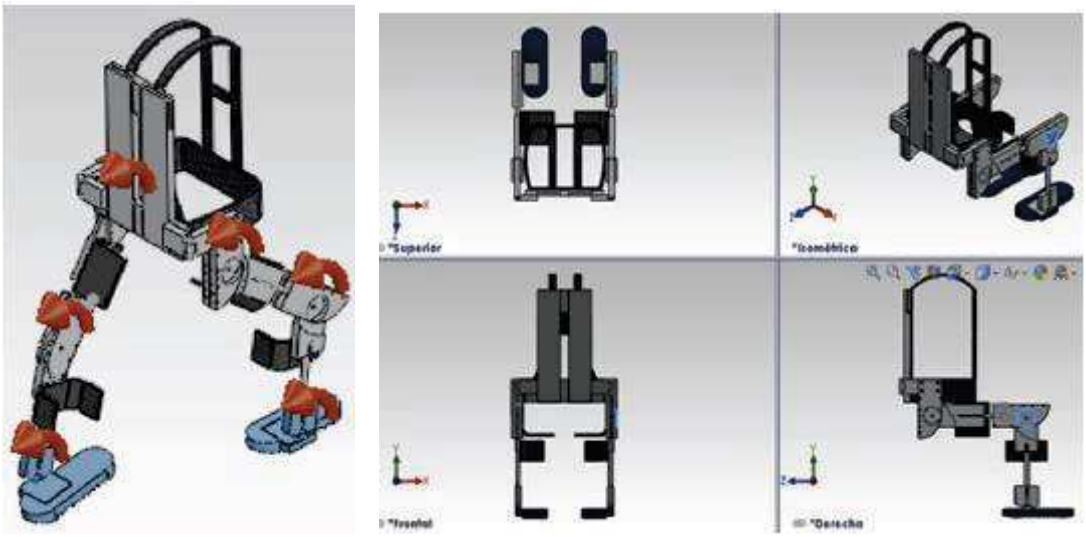


Figura 5. Movimiento de los componentes del exoesqueleto.

En 2016, la co-simulación se hace necesaria para conocer métodos de análisis, resolución e interpretación del modelo en estudio. Sudharsan, J. & Karunamoorthy, L. proponen un método donde se desarrolla una ecuación cinemática de un robot antropomórfico, como se observa en la Figura 6, posteriormente es modelada en Pro-E una solución de software para diseño mecánico de alto nivel 3D invocado en la herramienta de software ADAMS para su posterior análisis. Un algoritmo de planificación cubico es derivado matemáticamente para accionar el movimiento y simularlo utilizando el entorno de MATLAB [6].

Con la ayuda de MATLAB y ADAMS el modelo es accionado mediante el algoritmo de planificación de ruta escrito en MATLAB. La entrada al brazo robótico es una trayectoria deseada, que confirma la eficiencia del algoritmo de control de planificación y de camino. Los resultados permiten analizar el comportamiento dinámico y proporciona una visión clara de los parámetros como el par, aceleración y velocidad.

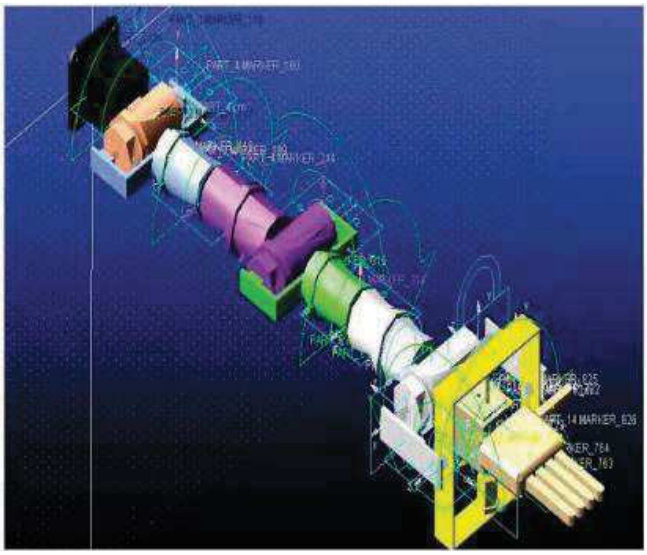


Figura 6. Brazo Robótico Antropomórfico de 8 Grados de Libertad

En ese mismo año, se diseña un robot industrial KUKA KR 5 sixx 850 con el apoyo de los softwares Solidwork y Matlab -SimMechanics [7]. Este trabajo presenta el desarrollo de un software que une al menos dos programa (cosimulación) para el control de posición de un robot manipulador industrial de seis grados de libertad (g.d.l.) Kuka KR5 sixxR850. Uno de los primero pasos es realizar el modelo del robot en forma virtual, es decir hacer el diseño mecánico, como son los eslabones y articulaciones en Solidword. Como segundo paso, consiste en convertir en modelo del robot manipulador realizado en Solidword a SimMechanics/Matlab, donde se le asignan los sensores (encoders) y actuadores (motores) a cada una de las articulaciones, asi como las restricciones del robot, en la figura 7 se muestra la co-simulación del robot.

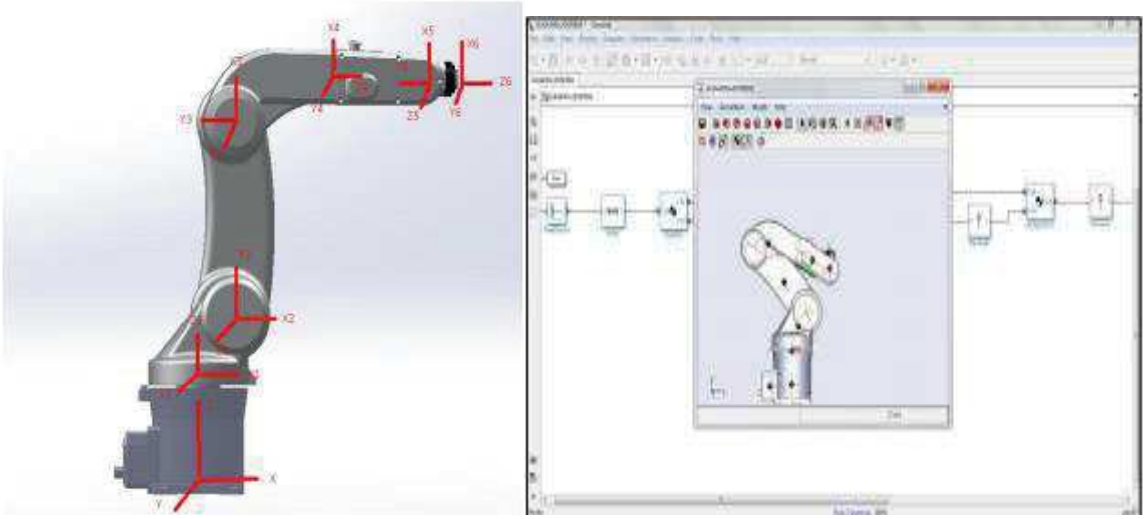


Figura 7. Co-simulación en SimMechanics del robot Kuka KR-5

La co-simulación también es llamada simulación cooperativa por el apoyo de varios software, es una metodología de simulación que permite a los componentes individuales para simular en ambientes virtuales y que se ejecutan simultáneamente, por lo que se tiene intercambio de información de manera cooperativa.

Este trabajo presenta el desarrollo de un software que une dos programas programa (co-simulación) para el control de posición de un robot manipulador como es dispostivo Omni Phantom de seis grados de libertad (g.d.l.), tres actuados y tres subactuados.

Uno de los primeros pasos es realizar el modelo del robot en forma virtual, es decir hacer el diseño mecánico, como son los eslabones y articulaciones en Solidworks. Como segundo paso, consiste en convertir en modelo del robot manipulador realizado en Solidworks a SimMechanics, donde se le asignan los sensores (encoders) y actuadores (motores) a cada una de las articulaciones, asi como las restricciones del robot. Y como parte final se hace un control de posición para manipular el robot en Matlab/Simulink donde se hace la co-simulación.

2. Software y Hardware

En esta sección, se describe los dos softwares empleados en la elaboración de este trabajo de investigación, con el propósito de relacionarse con la funcionalidad de cada software y su aplicación en la co-simulación. Asimismo, se describe el dispositivo háptico utilizado para la cosimulación.

Esto es con el propósito de señalar la variedad de diferentes herramientas informáticas (SolidWorks, SimMechanics, Matlab), con los cuales es posible acceder al diseño y análisis de modelos en estudio.

2.1 Matlab.

Es un lenguaje de programación de alto nivel y un entorno interactivo para el cálculo técnico, e incluye funciones para el desarrollo de algoritmos, análisis de datos, cálculo numérico, y visualización. Simulink: es un entorno de diagramas de bloques entendible para la simulación de sistemas y el diseño basado en modelos. Permite a los ingenieros simular y analizar una amplia gama de sistemas que incluyen controles, señales y procesamiento de imágenes, comunicaciones, y sistemas físicos multidominio [8].

2.2 Solidworks.

Es un software de modelado mecánico en 3D, asistido por computadora, cuenta con una gran gama de herramientas de diseño para dibujo, piezas y ensamblajes, Solidworks incluye herramientas de gestión de proyectos, productividad, análisis y simulación que lo convierten en uno de los softwares más competitivos en el mercado [9].

2.3 SimMechanics.

Es una herramienta que se utiliza para modelar y simular sistemas mecánicos en el ambiente de Simulink, además de poder realizar estudios y análisis. El estudio de los sistemas mecánicos es realizado desde la interfaz de Simulink. SimMechanics permite trabajar con diagramas de bloques para simular el movimiento de sistemas mecánicos y medir el movimiento generado por la actuación mecánica. Los bloques des SimMechanics se unen de manera normal a los bloques de Simulink mediante los bloques de sensores y actuadores [10].

2.4 Dispositivo háptico.

El dispositivo háptico Geomagic® Touch TM (OMNI PHANToM®) de la figura 8 proporciona una auténtica navegación tridimensional y re-troalimentación de fuerza e integran el sentido del tacto en los sistemas de modelado 3D, así como en aplicaciones comerciales y de investigación. Este dispositivo háptico de 3D Systems puede medir de forma precisa la posición espacial 3D (a lo largo de los ejes X, Y y Z) y la orientación (giro, inclinación y dirección) del lápiz de mano. Utiliza motores para crear las fuerzas de retorno en la mano del usuario para simular el tacto y la inter-acción con objetos virtuales, es decir, proporciona una retroalimentaciónde fuerza de 3 grados de libertad (DOF) [11].

En la tabla 1 se muestran las especificaciones del dispositivo háptico OMNI PHANToM.

Tabla 1. Especificaciones del dispositivo Geomatic Touch

Área de trabajo	160An × 120Al × 70P mm
Rango de movimiento	Movimiento de la mano con giro de la muñeca
Resolución nominal	0.055mm
Fuerza máxima	3.3N
Retroalimentación de fuerza	x,y,z
Interfaz Puerto	Ethernet conforme a RJ45

Figura 8. Geomagic® TouchTM (OMNI PHANToM®)

2.5 Equipo de cómputo

La figura 9 se muestra el equipo de cómputo utilizado para el desarrollo de este proyecto, es una computadora de escritorio cuyas características se muestran en la tabla 2.

Figura 9. Computadora de escritorio.

Tabla 5.3. Especificaciones del equipo de computo

Marca	Marca HP
Modelo	Modelo xw6600 Workstation
Procesador Intel	Procesador Intel ® Xeon ® CPU E5420 @2.50 GHz
Memoria instalada (RAM)	8.00 GB
Tipo de sistema	Sistema operativo de 64 bits
Edición de Windows	Windows 7 Ultimate

3. Desarrollo y resultados experimentales

3.1 Elaboración del modelo dinámico en Solidworks.

El sistema mecánico ha sido completamente realizado en el software de diseño SolidWorks, con las medidas originales del modelo tomando en cuenta los principios de diseño de máquinas como simetría. El software cuenta con una gran cantidad de herramientas para el diseño y modelado de piezas mecánicas en 3D, en la figura 10 se muestra el modelo.

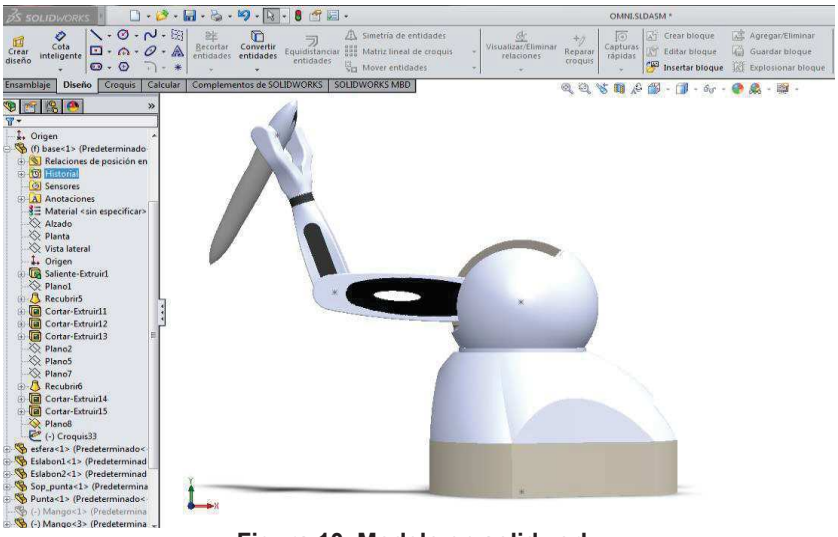


Figura 10. Modelo en solidwork.

La elaboración de cada una la pieza permite generar un modelo cercano a la realidad en la figura 11 se muestra la vista de conjunto del modelo en solidworks, que muestra las piezas desacopladas entre las cuales incluye pernos, eslabone de unión, eslabones principales de transmisión de movimiento como se puede observar.

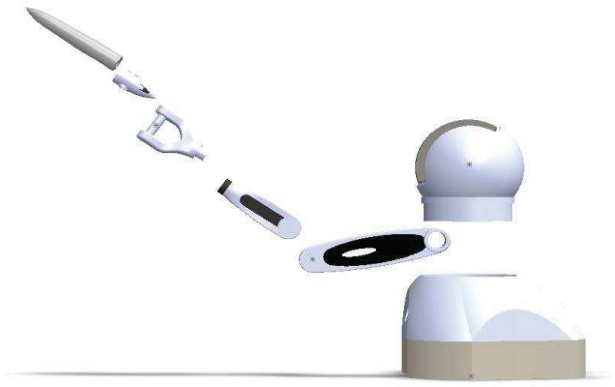


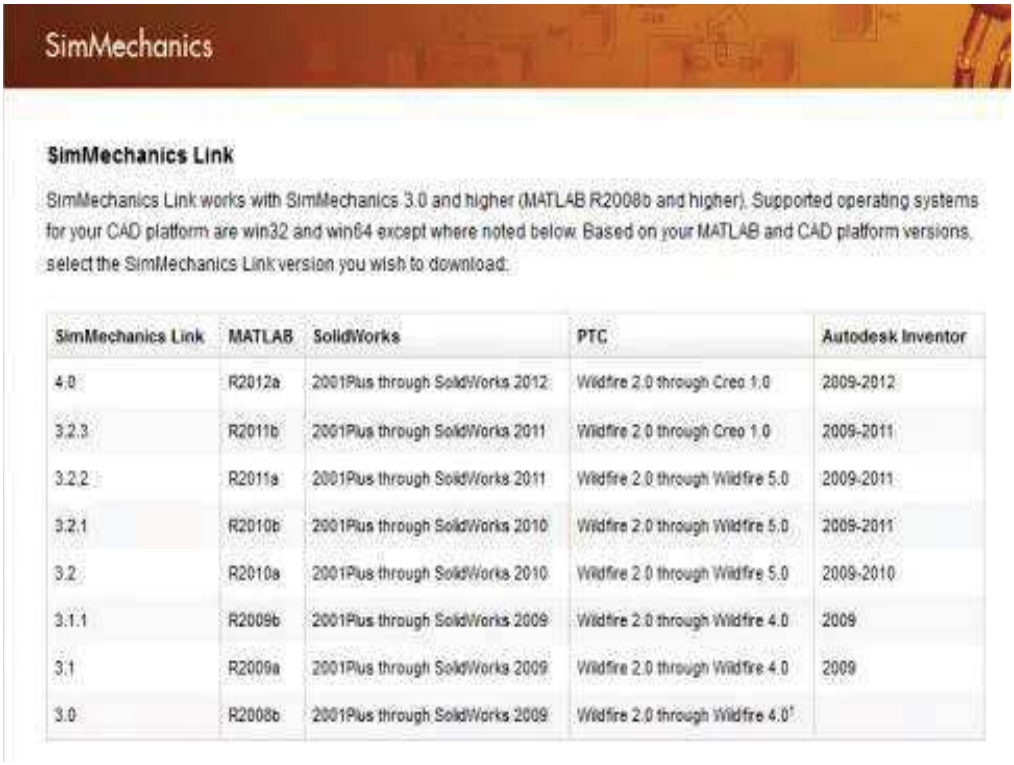
Figura 11. Vista explosionada del modelo en solidworks

3.2 Importación del modelo dinámico en Solidworks a Adams

Para realizar la importación, hay que descargar los 2 archivos de instalación de la página de descargas de Matworks correspondiente a la versión de Matlab instalada, la versión de SolidWorks instalada y la arquitectura de Windows en su PC. El link es mostrado a continuación.

https://la.mathworks.com/products/simmechanics/download_smlink.html

En la figura 12 muestra la ventana con los programas a descargar, dependiendo de la versión que se tenga instalado de Matlab.



SimMechanics

SimMechanics Link

SimMechanics Link works with SimMechanics 3.0 and higher (MATLAB R2008b and higher). Supported operating systems for your CAD platform are win32 and win64 except where noted below. Based on your MATLAB and CAD platform versions, select the SimMechanics Link version you wish to download:

SimMechanics Link	MATLAB	SolidWorks	PTC	Autodesk Inventor
4.0	R2012a	2001Plus through SolidWorks 2012	Wildfire 2.0 through Creo 1.0	2009-2012
3.2.3	R2011b	2001Plus through SolidWorks 2011	Wildfire 2.0 through Creo 1.0	2009-2011
3.2.2	R2011a	2001Plus through SolidWorks 2011	Wildfire 2.0 through Wildfire 5.0	2009-2011
3.2.1	R2010b	2001Plus through SolidWorks 2010	Wildfire 2.0 through Wildfire 5.0	2009-2011
3.2	R2010a	2001Plus through SolidWorks 2010	Wildfire 2.0 through Wildfire 5.0	2009-2010
3.1.1	R2009b	2001Plus through SolidWorks 2009	Wildfire 2.0 through Wildfire 4.0	2009
3.1	R2009a	2001Plus through SolidWorks 2009	Wildfire 2.0 through Wildfire 4.0	2009
3.0	R2008b	2001Plus through SolidWorks 2009	Wildfire 2.0 through Wildfire 4.0	

Figura 12. Ventana de Descarga de Simmechanics

Una vez que se tiene el archivo Zip descargado, escribir desde el command window de Matlab se escribe el siguiente comando

instala_ad don('smlink.r2015b.win64.zinc')

Ahora, ir a SolidWorks y en complementos, marcar SimMechanics Link, como se observa en la figura 13.

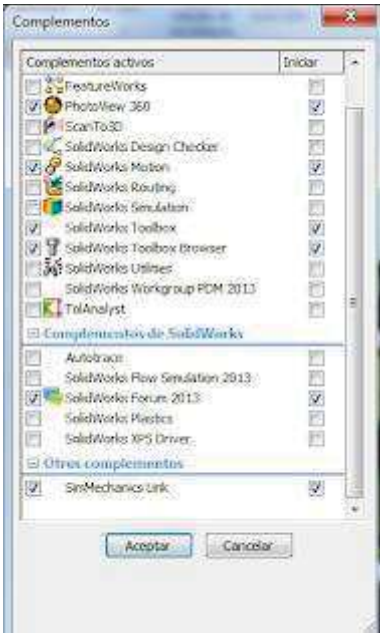


Figura 13. Activación de SimMechanics

3.3 Importación del modelo dinámico de Matlab a Simmechanics

Abrir la ventana de Matlab e se ingresa el siguiente comando para registrar la aplicación.

molinos_lindel

Para poder exportar el ensamble, primero hay que registrar Matlab como un servidor de automatización. Esto se hace ingresando los siguientes comandos.

regmatlabserver

enableservice('AutomationServer',true)

Para tener la importación final, abrir SolidWorks y aparece un menú para exportar el ensambles a *.xml. (Este menú solo aparece en ensambles), como se muestra en la figura 14

Figura 14 . Importar

Se puede exportar a primera generación o segunda generación el ensamble deseado. Por último, en Matlab ingresar el siguiente comando para que convierta el archivo *.xml y se visualice en Simulink, para ellos escribir los siguientes comandos:

mech_import('Omni Phantom .xml') % Primera generación

smimport('Omni Phantom .xml') % Segunda generación

y la importación en matlab queda como se muestra en la figura 15.

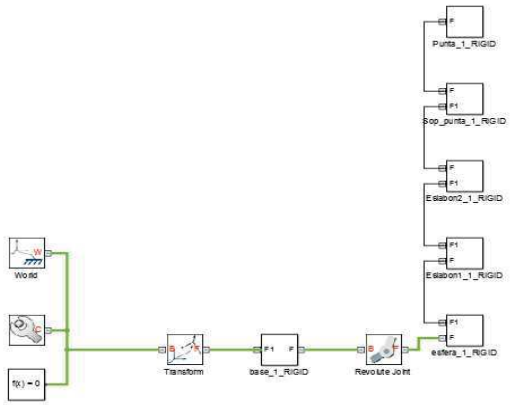


Figura 15. Diagrama de Bloques en SimMechanics

Y el entorno en SimMechanics que como se muestra en la figura 16, como se puede observar aparecen los sistemas de coordenadas de cada articulación.

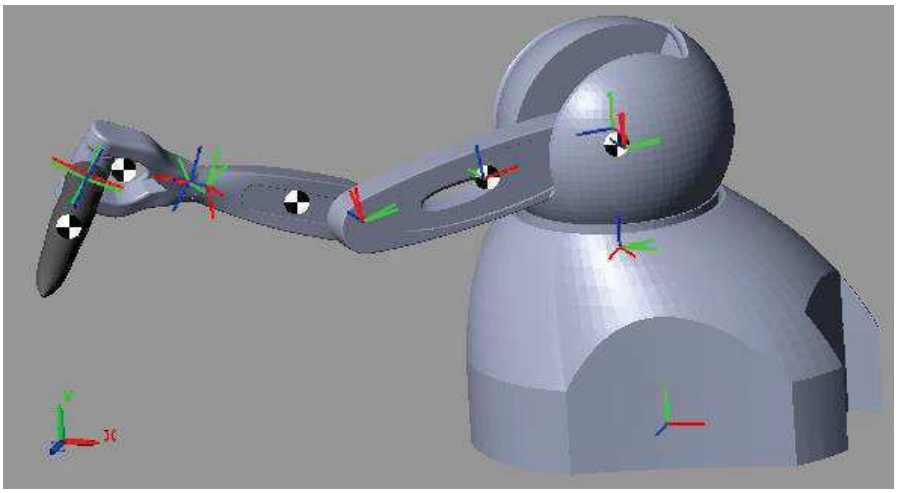


Figura 16. Modelo Dinámico en SimMechanics

3.4 Equipo experimental y resultados

En la figura 17, se muestra el equipo experimental que se utiliza para la co-simulación, como se observa esta compuesta y el dispositivo háptico Omni Phantom.

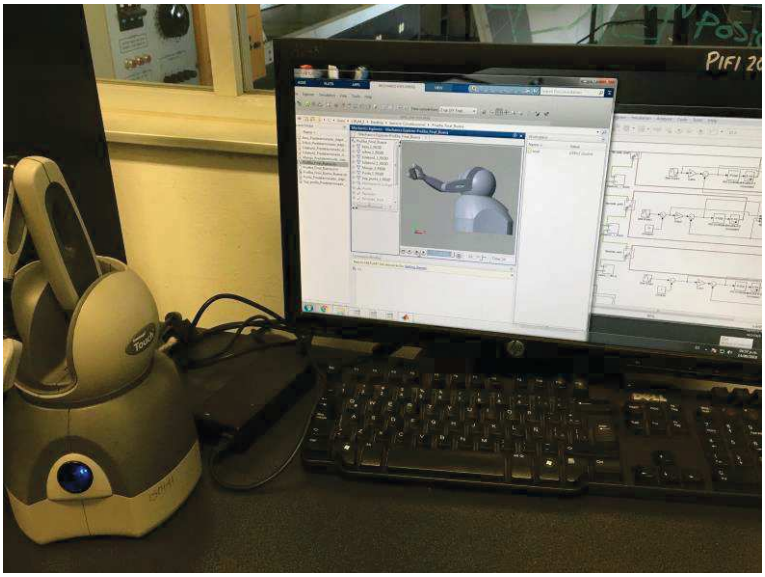


Figura 17. Equipo experimental

El diagrama de bloques en Matlab y la ventana de simulación de SimMechanics trabajando en conjunto para establecer un co-simulación y permitir el análisis del modelo como se observa en la figura 18.

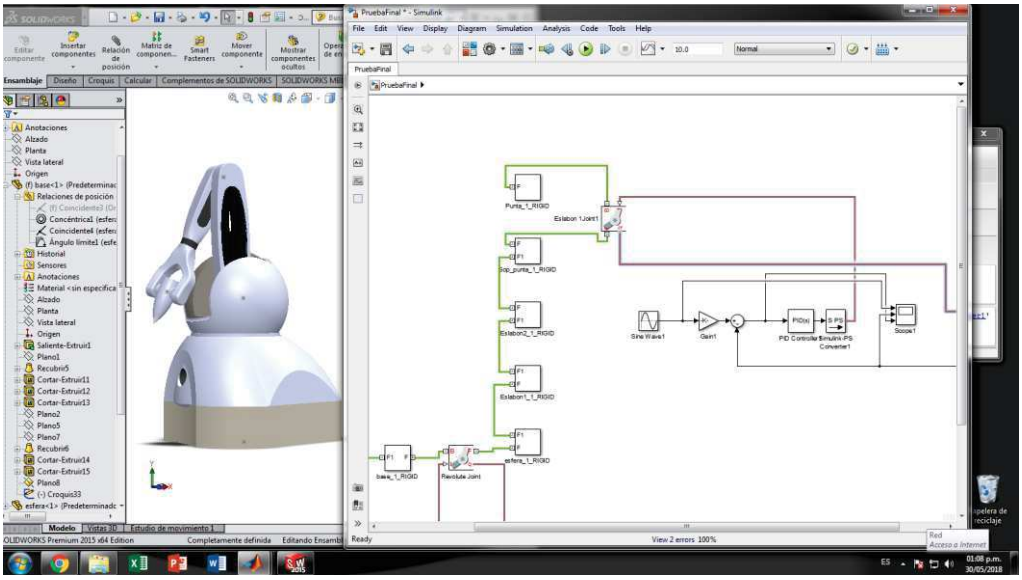


Figura 18. Co-simulación SimMechanics y Matlab

Para validar el modelo en co-simulación funcione correctamente, se utiliza un control lineal PID para verificar que el diseño en 3D esté funcionando correctamente. para el control del proceso dinámico. Su objetivo es permitir que se acerque lo más posible al punto de referencia y el error tienda a cero. En la figura 19 se muestra el diagrama de bloques en Matlab y la ventana de simulación de SimMechanics trabajando en conjunto.

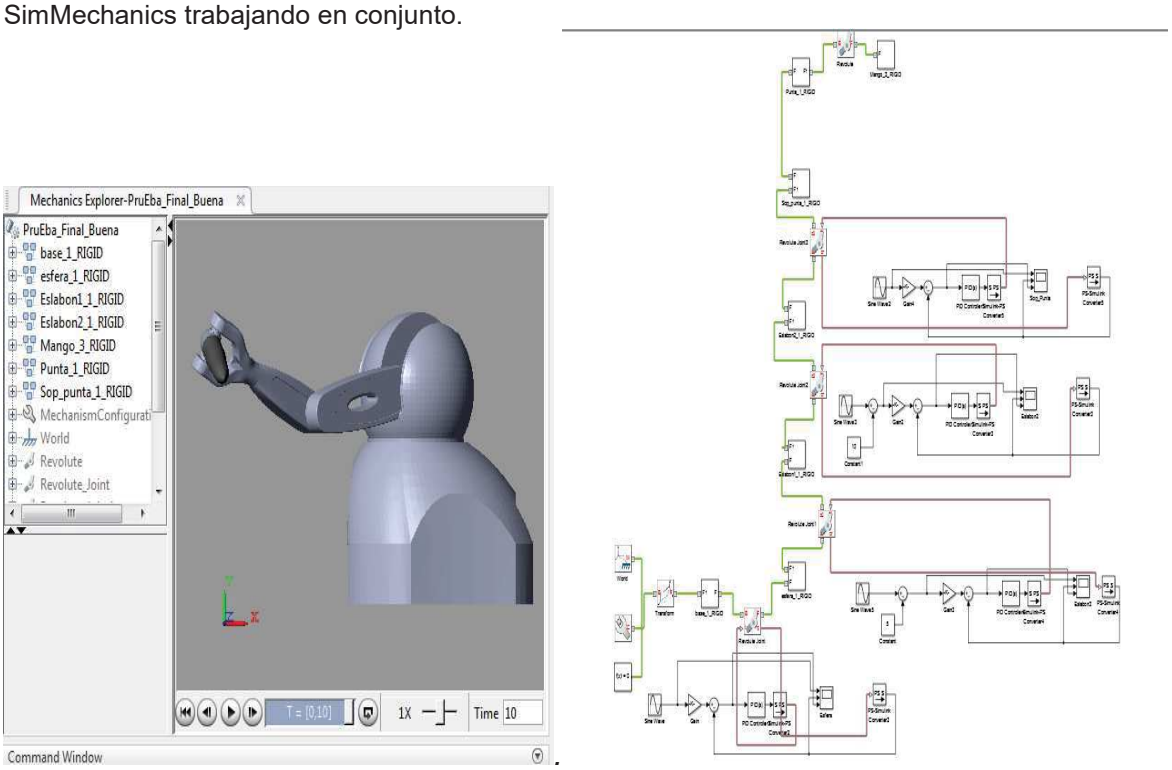


Figura 19. Co-simulación Matlab/SimMechanics

Como señal de referencia se introduce la señal seno con una amplitud de 180 grados a cada una de las articulaciones, y en la figura 20 se puede observar que las señales son similares, la señal

roja es la deseada y la señal real es azul. La señal del error angular se muestra en la figura 21, los errores son muy pequeños, por lo que se concluye que el sistema co-simulado es diseñado correctamente, y se puede introducir cualquier algoritmo de control antes de llevarlo a la validación experimental.

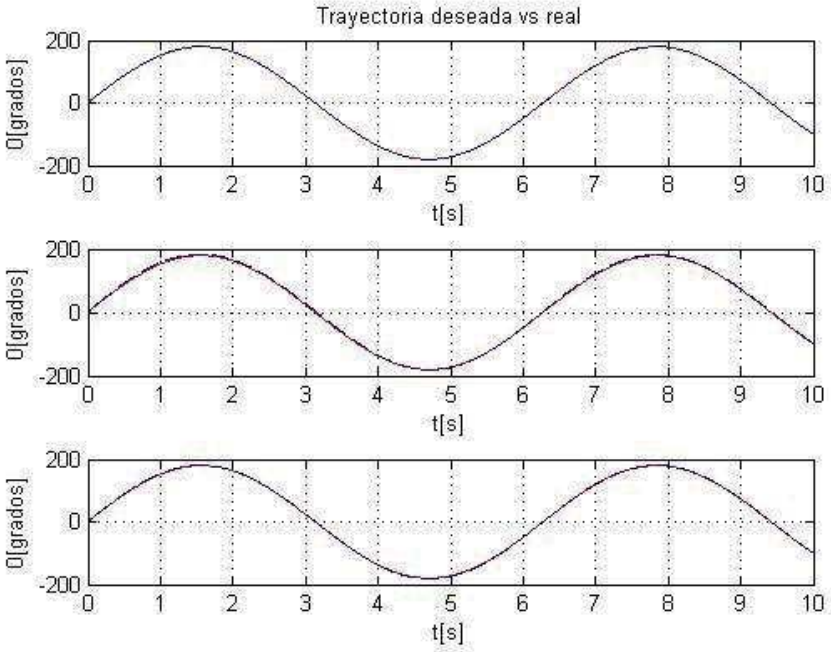


Figura 20. Señal deseada vs señal real.

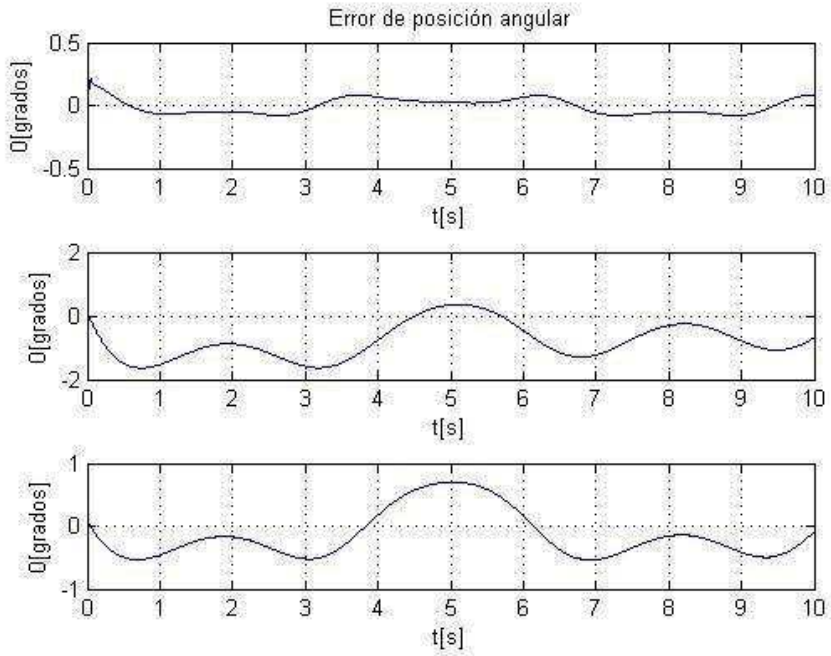


Figura 21. Señal del error.

Conclusiones

Este trabajo muestra el diseño de un robot manipulador virtual a través de la co-simulación entre los softwares SolidWorks y Matlab, en el primer software se diseña el modelo completo del robot a medidas reales otorgadas por el dispositivo háptico Omni Phantom, y en el segundo se utiliza para importar el modelo del robot, en SimMechanics se hace toda la parte virtual del robot, es decir se asignan los sensores y actuadores; y en Matlab se hace toda la parte de la co-simulación y control. Para validar que el modelo del robot manipulador en co-simulación funciona correctamente se emplea un control lineal PID con resultados satisfactorios. Este trabajo ayuda al diseño de robots y al control, ya que muchas veces se necesita validar los algoritmos de control en co-simulación antes de llevarlos a la validación experimental y con ello no dañar el equipo experimental. Por lo que se concluye que la co-simulación funciona correctamente.

Referencias

[1] Romaiguere, Anton JL, Roth M, Casini L, Roll JP. “Kinaesthesia activates both motor and parietal cortical areas in humans: a parametric fMRI study. Cognit”. Brain Res. 16, 74-82, 2007.
[2] E. Lugo, P. A. Co-simulación del Diseño Biomecánico para un Exoesqueleto Robótico del Miembro Inferior. Obtenido de Scielo: http://www.scielo.org. mx/scielo.php?script=sci_arttext&pid=S0188-95322014000200004. 2014.
[3] Arévalo Montaña, César Andrés. “Desarrollo de un Modelo de Simulación Multicuerpo del Conjunto Tractor Implemento en el Software Simmechanics”. Tesis de la Universidad Nacional de Colombia. 2012.
[4] Pérez Sandoval, Mary Paola. “Co-Simulación Adams/Matlab para el Control de Posición del Robot Gryphon”. Tesis de la Universidad Pontificia Bolivariana. 2012.
[5] E. Lugo, P. A. Co-simulacion del Diseño Biomecánico para un Exoesqueleto Robótico del Miembro Inferior. Obtenido de Scielo: http://www.scielo.org. mx/scielo.php?script =sci_arttext&pid=S0188-95322014000200004. 2014.
[6] Sudharsan, J. & Karunamoorthy, L. “Path Planning and co-simulation control of 8 dof anthropomorphic robotic arm”. 2016.
[7] Gudiño-Lau, Jorge, Chávez-Montejano, Fidel, Méndez, Alan y Hernández, Adrian. “Sistema de Co-Simulación de un Robot Industrial para Control”. Revista de Aplicación Científica y Técnica. 2016.
[8] A. Gilat, MATLAB. Hoboken, NJ: Wiley. 2015.
[9] S. González. Solidworks practico II. Marcombo. 2012.
[10] Ramírez Ramos, Osvaldo. Tesis “Simulación en SimMechanics de un Sistema de control difuso para el Robot UDLAP”. 2008.
[11] 3DSystems. Geomagic ® touch device guide. 2013.

Ingeniería y Automatización, Capítulo 2, pp. 15 - 25.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Análisis de Esfuerzos en Dientes de Engranes Rectos, Mediante Metodologías AGMA y CAE.

Hernández Corona Francisco, Mora Santos Carlos Alberto, Bedolla Hernández Marcos, Flores Lara Vicente y Bedolla Hernández Jorge

Tecnológico Nacional de México / Instituto Tecnológico de Apizaco
Av. Instituto Tecnológico S/N, cp. 90300, Apizaco, México.
francisco_hc2002@hotmail.com

RESUMEN

Antes de que aparecieran los paquetes de diseño, los diseñadores solo contaban con su ingenio y un gran equipo de delineantes que trasladaban a un papel sus ideas con una cierta severidad. Probablemente, por este motivo, los primeros paquetes de diseño surgieron como una necesidad a estos dibujantes, con la ventaja de facilitar el uso, edición y rapidez.

Conforme el hardware evolucionaba, los programas eran mas rápidos y las bases de datos de mayor tamaño, en este momento surgieron problemas de insatisfacción en los usuarios, ya que un buen programa de dibujo no bastaba y era necesario un sistema que diseñara desde el dibujo hasta la pieza terminada, siguiendo ciertas reglas de diseño.

*En la actualidad el manejo de software **CAD-CAE** para el desarrollo del diseño y análisis de sistemas mecánicos es de uso frecuente en el sector industrial, este permite calcular como va a comportarse la pieza en la realidad, este avance tecnológico permite predecir el comportamiento de elementos y sistemas mecánicos y surge la pregunta ¿Qué confiabilidad se puede tener en cuanto a sus resultados? Con base en esto, es conveniente considerar el conocimiento, la habilidad y la capacidad del diseñador para el desarrollo de este tipo de trabajo.*

*Usualmente se debe trabajar con el método de elementos finito (**MEF**), el cual es un método numérico que permite la aproximación de soluciones de ecuaciones muy complejas. Así mismo, este método hace necesario mallar la pieza en pequeños elementos y el calculo que se efectúa sirve para determinar las interacciones entre estos elementos y solucionarlos para la obtención de resultados.*

*En el presente trabajo de investigación se propone una metodología para el análisis de falla por esfuerzo flexionante en los dientes de engranes rectos para una maquina trituradora de **PET**, por medio de Software de simulación, y el cual se comparará con los resultados obtenidos bajo las normas **AGMA** de la (American Gear Manufacturers Association).*

*Una parte importante del diseño del sistema de transmisión, son los engranajes que interactúan para mover los ejes donde se encuentran montadas las cuchillas que efectuaran el corte en el **PET**, en los cuales se desarrollara un análisis y simulación de los esfuerzos que sufren los dientes rectos de los engranes y determinar si son adecuados para la operación de dicha máquina.*

Palabras clave: CAD, CAE, AGMA, MEF, confiabilidad, interacción, análisis, flexionante, engranes.

Ingeniería y Automatización, Capítulo 2, pp. 15 - 25.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

1. Introducción

La Ingeniería Asistida por Computadora es una herramienta utilizada en el análisis, simulación, validación y optimización de sistemas mecánicos que comúnmente se desarrollan en tareas de ingeniería. Estas incluyen la robustez y el rendimiento de componentes y ensamblajes.

La Ingeniería Asistida por Ordenador o **CAE** (Computer Aided Engineering) proporciona un paso más en los sistemas **CAD** tradicionales, ya que además del diseño del modelo, también permite integrar sus propiedades, condiciones a las que está sometido el material del cual está constituido el elemento. De esta forma, las herramientas **CAE** existentes permiten calcular cómo va a comportarse la pieza o la estructura en la realidad, en aspectos tan diversos como: Esfuerzos, Deformaciones, Factor de seguridad, Características térmicas, Vibraciones, etc.

Para ello es necesario pasar de la geometría creada en un entorno **CAD** al sistema **CAE**.

En la actualidad en el área de Ingeniería Mecánica es frecuente el uso de CAE, sin embargo su aplicación en los sistemas de transmisión no se ha extendido debido a la complejidad en la estructura física de los mismos y a la necesidad de contar con conocimientos en el uso de Diseño Asistido por Computadora (CAD), esto ha hecho que en la mayor parte de los casos el diseñador continúe utilizando las fórmulas básicas en Ingeniería Mecánica complementándolas con factores de corrección recomendados por normas estandarizadas reconocidas internacionalmente, no obstante para seleccionar correctamente estos factores dependen del criterio y experiencia del diseñador.

En este trabajo de investigación se propone una metodología simple y sencilla para analizar un engrane de dientes rectos sometido a esfuerzo flexionante aplicando la metodología **CAE** y realizando un comparativo con los resultados obtenidos por las normas internacionales **AGMA**, se da inició con el análisis de un engrane recto convencional aplicando la fórmula de Lewis complementándola con el cálculo y selección de factores de corrección de **AGMA**, a continuación se desarrolla un modelo de la pieza en SolidWorks y posteriormente con el complemento SolidWorks Simulation se muestra como introducir valores y ejecutar un estudio activo para calcular el esfuerzo flexionante e identificar el valor máximo.

Para validar esta metodología se efectuan otros estudios con diferentes modelos, estableciendo una comparacion de resultados entre ambos métodos y presentando resultados de los mismos.

2. Metodología empleada en la determinación del esfuerzo flexionante.

Para este trabajo de investigación se considera un sistema de engranes de dientes rectos con la información mostrada en la **Tabla 1**.

Tabla 1. Datos nominales.

Dato	Unidad
Motor eléctrico	25 hp
Velocidad de rotación del motor (n_1)	1775 rpm
Velocidad de rotación de máquina trituradora (n_2)	1400 rpm
No. Dientes engrane eje primario	42 dientes
No. Dientes engrane eje secundario	42 dientes
Paso diametral (P_d)	7 pulg
Diámetro de paso (D_p)	6 pulg
Angulo de presión	20°
Diámetro del eje	2 pulg

2.1 Metodología aplicando normas AGMA

El número de calidad **AGMA** considerado para este caso Qv=8, se determina en función del nivel de exactitud en la transmisión, la cual va de 8 a 12 para engranajes de precisión.

En la **Tabla 2**, se muestran los datos complementarios calculados bajo la norma 1012F90 de AGMA (1990), [1].

Tabla 2. Datos complementarios bajo la norma AGMA.

Dato complementario	Valor calculado
Diámetro de paso del engrane (D_p)	6 pulg
Velocidad en la línea de paso (V)	2199 ft/min
Ancho de cara del diente (F)	2 pulg
Carga tangencial (W_t)	375.15 lb.f
Carga radial (W_r)	136.54 lb.f
Carga total (W)	399.22 lb.f

Las fórmulas empleadas para determinar los valores de los datos complementarios en el modelado y simulación del engrane, de la **Tabla 2**, se muestran a continuación:

$$D_p = \frac{N_p}{P_d} \tag{1}$$

$$V = \frac{\pi * D_p * n_2}{12} \tag{2}$$

$$W_t = \frac{33000 * H}{V} \tag{3}$$

$$W_r = W_t * tan\emptyset \tag{4}$$

$$W = \frac{W_t}{cos\emptyset} \tag{5}$$

La fórmula para calcular el esfuerzo flexionante en el diente de un engrane está basada en la ecuación de Lewis partiendo del principio de que un diente de engrane recto se puede estudiar como una viga en voladizo, pero se han agregado factores adicionales K que son modificadores empíricos para tomar en consideración diversas condiciones de operación. De acuerdo con la norma 2001-D04 AGMA (2004), [2].:

$$\sigma_b = W_t K_o K_v K_s \frac{P_d}{F} \frac{K_m K_B}{J} \tag{6}$$

Para determinar los factores de corrección, se hace uso de la norma 2001-D04 AGMA (2004). Así mismo, el factor geométrico a la flexión bajo la norma 908-B89 AGMA (1990), [3]. En la **Tabla 3**, se muestran los valores correspondientes.

Tabla 3. Factores de corrección AGMA.

Factor de corrección	Valor
Factor de distribución de carga (K_m)	1.2
Factor dinámico (K_v)	1.378
Factor de tamaño (K_s)	1
Factor de espesor de borde (K_B)	1
Factor de sobrecarga (K_o)	1.25
Factor geométrico de resistencia a la flexión (J)	0.401

Sustituyendo los valores de los Factores de corrección obtenidos en la formula (6), se calcula el esfuerzo de flexionante de acuerdo con la metodología **AGMA**:

$$\sigma_b = 6.785 \text{ ksi}$$

2.2 Metodología aplicando CAD-CAE.

La ecuación de Lewis fue introducida para estimar el esfuerzo de flexión en dientes de engranes en la que interviene la forma de los mismos. Esta ecuación aun sigue siendo la base de la mayoría de los diseños de engranes [4].

En la **Figura 1a**, se muestra un voladizo con dimensiones de su sección transversal, longitud y una carga, uniformemente distribuida a lo largo del ancho de la cara [4].

En la **Figura 1b**, se muestra que el esfuerzo máximo en un diente de engrane ocurre en el punto **"a"** [4].

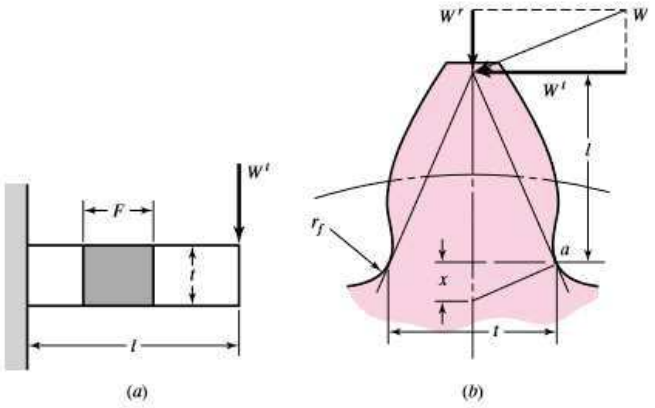


Figura1. Diente de un engrane relacionado con una viga en voladizo [4].

En la **Figura 2**, se representan los dientes de dos engranes en contacto o engranados. Para representar las cargas tangencial y radial que actúan en dichos dientes por medio del software, se hace necesario contemplar un punto de contacto que permita la interacción entre estos y este se encuentra considerado en el círculo de paso, que es donde se tiene el contacto máximo entre los dientes [5].

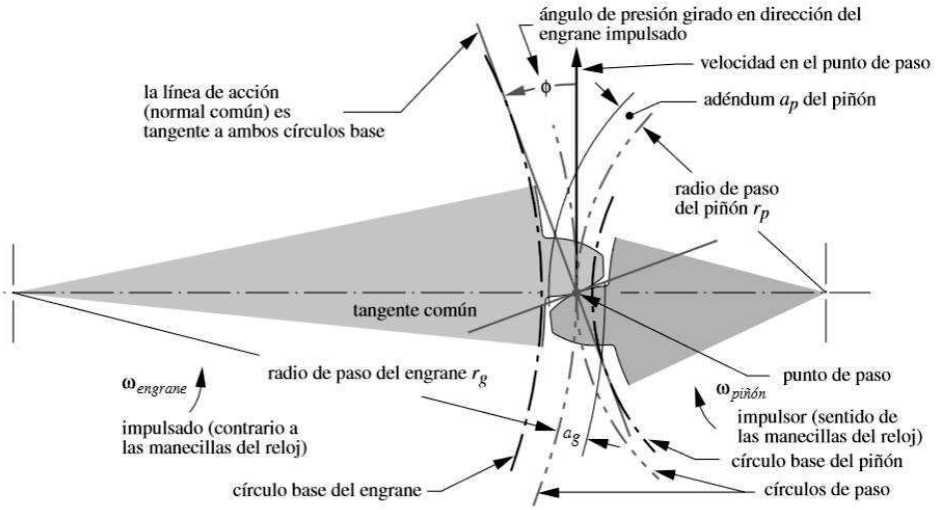


Figura 2. Geometría de contacto entre dientes de engranes [5].

Primeramente, mediante el CAD, se desarrolla el modelado de un engrane de dientes rectos en 3D, con la forma de involuta del diente, mostrado en la **Figura 3**.

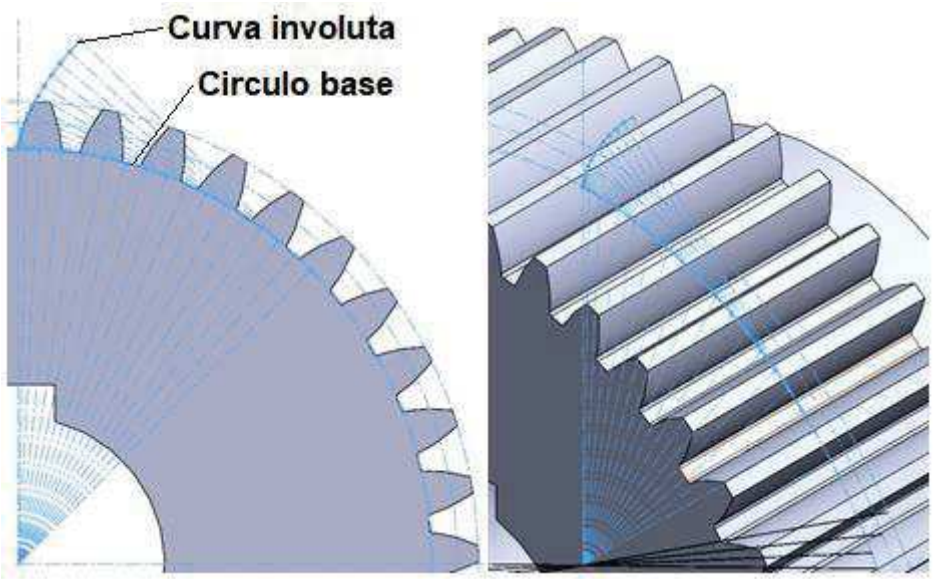


Figura 3. Perfil de involuta

En la **Figura 4**, se muestra el modelo 3D del engrane. Mediante el Software *SolidWorks from Dassault Systemes*, dicho elemento se modela con base en datos nominales y datos complementarios presentados en **Tabla 2**.

El siguiente paso es definir el material para efectuar el correspondiente análisis y simulación en software, en este caso se considera un acero recocido 4340 que es un acero al Cr-Ni-Mo recocido de alta templabilidad adecuado para flechas y engranes de alta ductilidad y resistencia al choque [6], ver **Figura 5.**

Figura 4. Modelo engrane 3D

Figura 5. Material engrane

Considerando una restricción de geometría fija en el centro del engrane, donde se ubica el eje y cuñero, como se muestra en la **Figura 6**, siendo que de esta forma se simula la aplicación de la carga que debe proporcionar el engrane durante su operación.

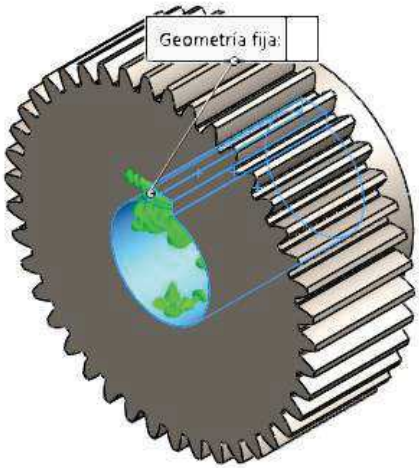


Figura 6. Restricción geometría fija

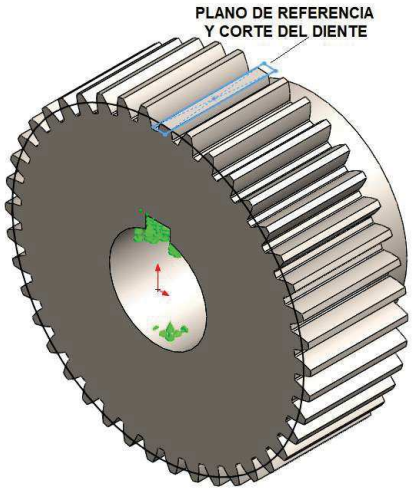


Figura 7. Plano de referencia y corte del diente

La identificación del punto de acción de las cargas tangencial y radial, respectivamente en el software, se efectúa considerando que dicho punto se encuentra donde ocurre el contacto máximo entre dos dientes, el cual será en el círculo de paso, como se muestra en la **Figura 7**. Así mismo, se efectúa un corte sobre la superficie del diente a la altura del diámetro de paso.

En la **Figura 8**, se muestra la representación de las cargas tangencial y radial, con sus respectivos valores, los cuales se identifican en la **Tabla 2**.

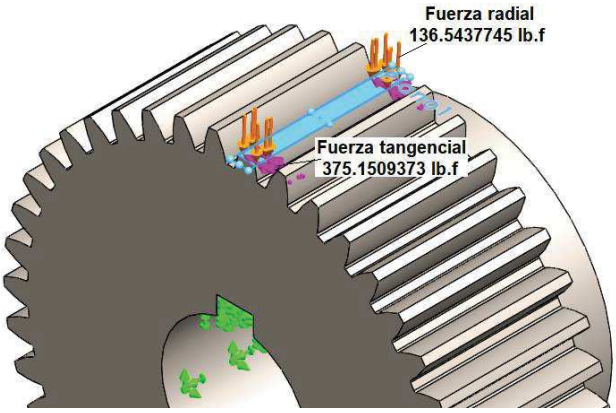


Figura 8. Carga radial y tangencial sobre el plano de corte del diente

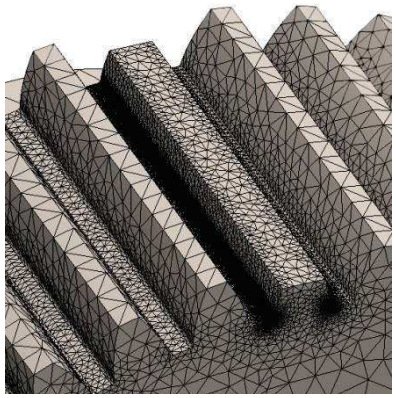


Figura 9. Densidad de malla en el engrane.

El mallado es una parte fundamental para la simulación y este consiste discretizar el modelo 3D al dividirlo en pequeñas porciones que simplifican al modelo, obteniendo un conjunto de nodos que forman una malla y que permite obtener soluciones aproximadas a una amplia variedad de problemas de mecánica en el continuo. La **Figura 9**, muestra una densidad de malla fina con valores de tamaño estándar para el elemento de 0.03 in y con tamaño fino de 0.001 in para las caras donde se presenta el esfuerzo flexionante con base en la dirección de las cargas.

.

2.2.1 Análisis estático.

El análisis estático de una pieza o ensamblaje permite conocer como se deforma el solidos bajo la acción de condiciones de carga definidas y como esta se transmite a través del mismo. El análisis permite determinar los desplazamientos, las deformaciones unitarias, las tensiones y las fuerzas de reacción que se producen sobre el modelo estudiado. Además, permite conocer si el modelo puede llegar a romper y en caso de hacerlo, en qué punto se iniciaría la fractura. A continuación, se definen algunas de las principales propiedades mecánicas que se debe tener en cuenta en un análisis estático [7]:

Tensión de Von Mises. - Es la tensión equivalente que actúa sobre los elementos diferenciales del modelo y que es comparada con la tensión limite elástica del material. Permite conocer si la tensión esta dentro de la zona elástica y optima o por el contrario se encuentran en la zona plástica no deseada [7].

Limite elástico. - Es la máxima tensión que puede soportar un material sin que se produzcan deformaciones permanentes [7].

Tensión limite. - Valor de la tensión para el cual el material no puede soportar mas esfuerzos y se produce la rotura [7].

Con la finalidad de determinar los resultados que proporciona el análisis por el Método de los Elementos Finitos, se desarrolla la simulación de este engrane en el software, como puede observarse en la **Figura 10**, la base del diente muestra el lugar donde se producen los máximos esfuerzos debido a la acción de las cargas a las cuales se encuentran sometidos cada uno de los dientes. Así mismo, el valor obtenido del esfuerzo del esfuerzo flexionante máximo es:

$$\sigma_b = 6.704 \text{ ksi}$$

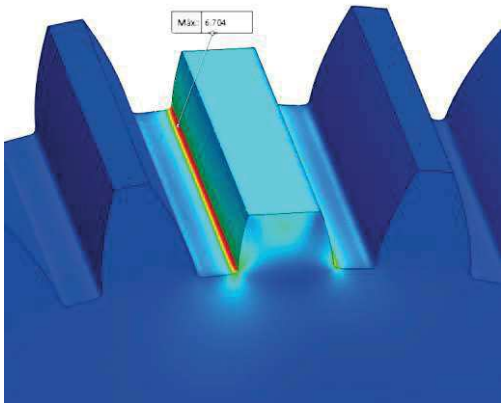


Figura 10. Máximo esfuerzo flexionante en el diente

Los resultados obtenidos muestran una amplia similitud entre la norma **AGMA** y la metodología **CAE,** por lo que el uso CAE representa una alternativa para el análisis y solución de engranes.

3. RESULTADOS

Con la finalidad de evidenciar los resultados obtenidos al efectuar los cálculos y simulación correspondiente, bajo la metodología **AGMA** y Software de simulación, y efectuar la comparación pertinente, se realizan otros estudios con diferentes datos, los cuales se muestran en la **Tabla 4**.

Dado que se trata de una máquina trituradora de **PET**, se hace necesario que el numero de dientes del piñón y del engrane sean los mismos, ya que se trata de una transmisión y los engranes deberán girar a la misma velocidad, pero en sentido contrario para provocar la trituración del PET.

Tabla 4. Resultados de estudio realizados a diferentes engranes rectos, en función del esfuerzo flexionante, bajo la norma AGMA y CAE.

	ESTUDIOS REALIZADOS			
DATOS	**1**	**2**	**3**	**4**
Potencia en hp	20	25	25	20
Velocidad rpm	1200	1400	1800	1200
P_d	6	7	5	6
D_p	6	6	6	3
N_p	36	42	30	18
N_g	36	42	30	18
Φ	20°	20°	20°	20°
Diámetro eje (in)	1	2	1 7/8	1
Q_v	8	8	7	8
K_m	1.32	1.2	1.35	1.33
K_v	1.352	1.378	1.487	1.265
K_s	1	1	1	1
K_B	1	1	1	1
k_o	1	1.25	1.6	1
J	0.413	0.401	0.376	0.316
σ_b (ksi) AGMA	**4.54**	**6.785**	**8.014**	**11.180**
σ_b (ksi) CAE	**4.44**	**6.704**	**7.969**	**11.121**
Porcentaje error	**2.2%**	**1.19%**	**0.56%**	**0.52%**

La **Tabla 4**, muestra los resultados obtenidos del esfuerzo flexionante en el diente de un engrane recto, tomando en consideración la variación de datos para cada estudio, y desarrollar el cálculo mediante la fórmula (6). Así mismo se efectúa la simulación **CAE**, haciendo uso de los factores de corrección **AGMA**, los cuales dependen principalmente del criterio del diseñador, bajo esta misma consideración el modelado del engrane en 3D se efectúa con la mayor precisión posible y esto se logra mediante el perfil de involuta.

En el proceso de mallado, el software propone por default un tamaño de los elementos; por lo que se hace necesario manipular el tamaño de los elementos de tal forma que este pueda proporcionar un mallado fino, el punto de interés, como lo muestra la Figura 8, donde se localiza el esfuerzo máximo, además de tener un mallado basado en curvatura se proporciona un mallado todavía mas fino para una solución las precisa.

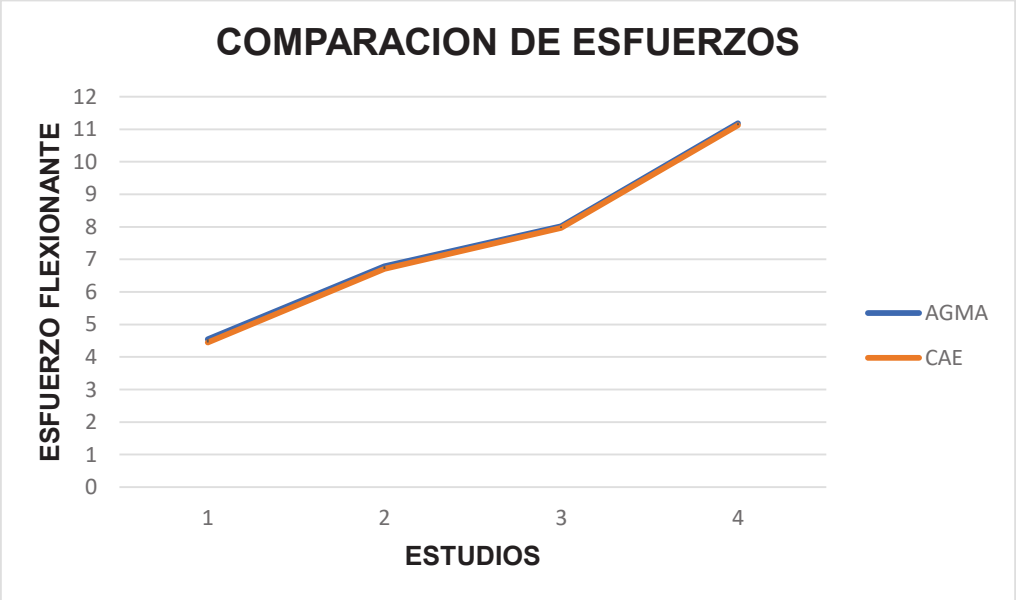
Calcular el porcentaje de error permite comparar que tan exacto es un valor estimado. El porcentaje de error permite conocer la diferencia entre el valor aproximado y el verdadero como porcentaje del valor exacto y ayuda a determinar qué tan cerca está el valor estimado del valor real. La fórmula (8), proporciona, la forma en la que se puede calcular dicho valor:

$$\%\ error = \frac{Valor\ exacto - Valor\ aproximado}{Valor\ exacto} * 100 \tag{8}$$

En la **Tabla 4**, se presenta la determinación del porcentaje de error al comparar las 2 metodologías empleadas en este trabajo de investigación, mismos que al visualizarlos son tolerable en cuanto a sus valores.

Como puede observarse en la **Grafica 1**, se presenta una comparación entre los esfuerzos flexionantes obtenidos en el diente de un engrane recto mediante la metodología **AGMA** y **CAE**, debido a que la tendencia es similar en cada uno de los estudios, se opta por presentar resultados de cuatro estudios realizados con diferentes datos.

De lo cual se puede deducir que la confiabilidad en el uso de la tecnología de los sistemas **CAD-CAE** es adecuada en el análisis y simulación de engranes. Por lo que se hace necesario que el diseñador cuente con la experiencia suficiente para desarrollar, analizar y simular este tipo de elementos, dado que el software proporciona una alternativa de solución y el diseñador deberá aplicar su criterio para tomar la mejor decisión en la solución del diseño.



Grafica 1. Comparación entre esfuerzos flexionantes.

En la **Figura 11**, se muestra el sistema de transmisión mediante engranes, que deberán ser ensamblados en conjunto con los ejes, los cuales a su vez tienen montadas las cuchillas de corte que trituraran el PET, dentro de la cámara de molienda de la máquina trituradora de **PET**.

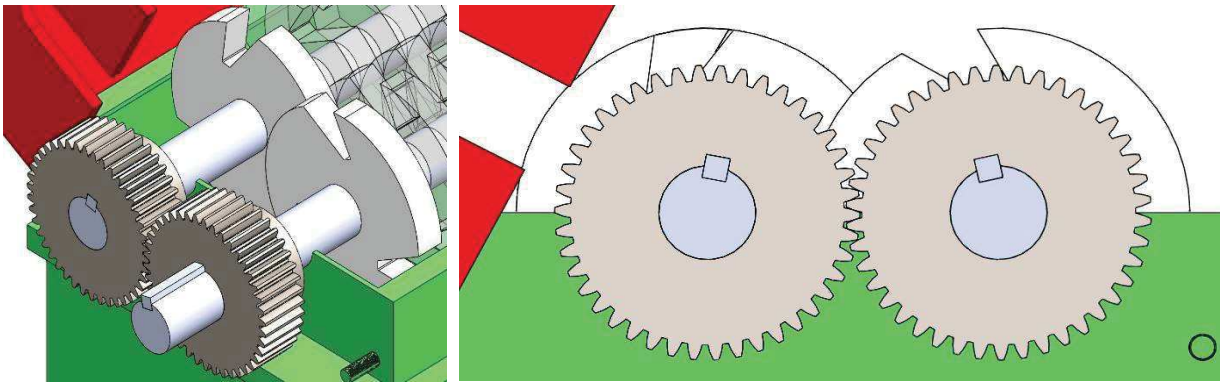


Figura 11. Sistema de transmisión por engranes.

4. Conclusiones

De acuerdo con los resultados obtenidos, se puede observar que la aplicación de la metodología **CAE** proporciona resultados aceptables, ya que al compararla con la metodología **AGMA**, esta presenta un margen de error mínimo y tolerable para su estudio, por lo que es confiable en el momento de efectuar su análisis.

El **MEF** o método del elemento finito es confiable siempre que el diseñador tenga los fundamentos teóricos y conocimientos adecuados en su manejo y en la aplicación de simulación con software de diseño de alto nivel. Si la pieza o elemento mecánico se modela de forma incorrecta por desconocimiento, el software proporcionara resultados erróneos, y si no se tiene el criterio para tomar una decisión conveniente, esto puede propiciar que el elemento tenga consecuencias en su operación.

De igual forma las normas **AGMA** deben ser utilizadas preponderantemente por un diseñador con experiencia debido a los criterios que se deben tomar en cuenta para la determinación de los factores de corrección.

Es importante recalcar que el software de simulación muestra cómo se comportaría el sistema de transmisión de engranes sometido a dichas cargas y presenta resultados físicos que simulan la realidad, pero cada diseñador debe decidir bajo sus propios criterios y utilizar las herramientas que considere adecuadas ya que como tal el software realiza los cálculos y desarrolla lo que se le pide, pero este no toma decisiones finales.

Bajo estas consideraciones, se puede concluir proponiendo este sistema de transmisión por medio de engranes para una máquina trituradora de PET, ya que, con base en los resultados obtenidos y las metodologías empleadas, los resultados son adecuados.

Referencias

[1] American Gear Manufacturers Association. Norma 1012-F90. Gear Nomenclature, Definitions Of Terms with Symbols AGMA 1990.
[2] ANSI/AGMA 2101-D04. “Fundamental Rating Factors and Calculations Methods for Involute Spur and Helical Gear Teeth”.
[3] 908-B89 8 Geometry Factors for Determining the Pitting Resistance and Bending Strength of Spur, Helical and Herringbone Gear Teeth
[4] Budynas R. G., Nisbett J. K., *“Shigley’s Mechanical Engineering Design”*, Mc Graw Hill, United States of America, 9ª Edición, 2006.
[5] Norton R. L., *“Diseño de maquinas Un enfoque integrado”*, PEARSON, México, 4ª Edición, 2011.
[6] Mott R., *“Diseño de Elementos de Máquinas”,* PEARSON, México, 4ª Edición, 2006.
[7] Gómez S., *“SolidWorks Simulation”* Alfaomega, Ra-Ma, España, 1ª Edición 2010.

Ingeniería y Automatización, Capítulo 3, pp. 26 - 44.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Descomposición de Trayectoria Óptima para Robots Móviles Generada por PSO y Descriptores de Forma

Méndez Guzmán Hugo Antonio, Flores Muñoz Víctor Hugo
y Solano Blancarte Miguel Ángel

*Departamento de Ingeniería Robótica. Universidad Politécnica del Bicentenario.
Carretera Silao - Romita Km. 2. Col. San Juan de los Durán C.P. 36283
hmendezg@upbicentenario.edu.mx

Resumen

Este trabajo muestra el comportamiento del algoritmo Particle Swarm Optimization (PSO) modelando el escenario de un Robot Móvil con la presencia de obstáculos virtuales de geometrías regulares y no regulares. La imagen es adquirida por una webcam y procesadas a través de openCV describiendo el entorno de un Robot Móvil en camino. A través de procesamiento de imágenes se adquieren la posición inicial y final de la Trayectoria, además de modelar el escenario y la función costo dependiendo del descriptor de forma elegido. Los puntos intermedios de la trayectoria generados por el algoritmo PSO son usados como puntos de control para que un Robot Móvil de Tracción Diferencial ejecute la trayectoria descomponiendo su trayecto en movimientos rotativos y lineales. Finalmente se muestra una comparativa de las trayectorias óptimas generadas con los descriptores cónicas, regiones y Y-axis partiendo de una misma trayectoria aleatoria.

Palabras clave: Robótica Móvil, Descriptores de Forma, Optimización, Algoritmos Heurísticos, Particle Swarm Optimization.

1. Introducción

La evasión de obstáculos en la trayectoria de un robot móvil es una de las tareas cotidianas para que estos efectúen algún trabajo específico ya sea transporte de materiales, reconstrucción de escenarios, búsqueda de objetos y demás.

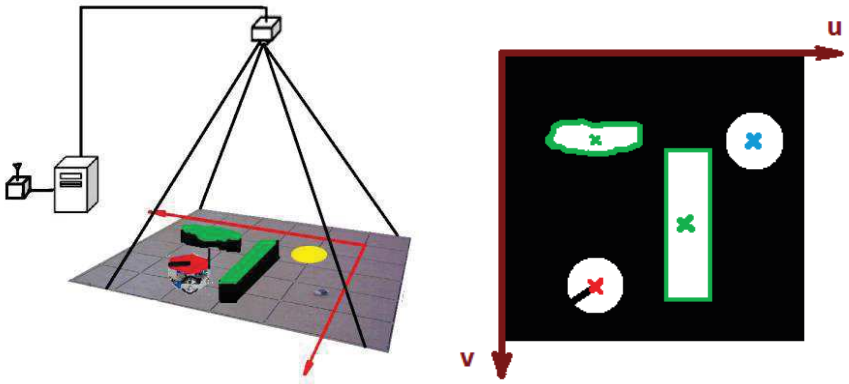
Una alternativa para que los robots móviles puedan ejecutar sus tareas es bajo la supervisión de su entorno a través de visión usando cámaras de vigilancia colocadas en la parte superior. Sin embargo, para poder efectuar un movimiento preciso, en el menor tiempo posible y bajo condiciones de seguridad tanto para los objetos de su entorno como para personal circulante en el área, es necesario contemplar además de la evasión, la búsqueda de una trayectoria optima que genere la ruta más adecuada en cuanto a la longitud recorrida, tipo de trayectorias a seguir dependiendo de la morfología del robot hasta tolerancias geométricas del algoritmo con respecto al cuerpo del robot.

Entre las técnicas más eficientes para la búsqueda óptima de soluciones se encuentra la programación dinámica y programación lineal, sin embargo el tiempo de cómputo muchas veces es demasiado alto. Otro enfoque es la aplicación de algoritmos heurísticos cuya velocidad de convergencia hacia un valor óptimo global es mucho mejor, además de poder enfrentar problemas altamente complejos, problemas no lineales hasta la búsqueda de soluciones en espacios multidimensionales [1]. El PSO es un algoritmo heurístico y se basa en comportamiento social, donde la mejor solución a un problema se puede representar como un punto en un espacio *n*-dimensional, tal que la búsqueda de la solución óptima se rige en el estudio de un problema por múltiples agentes resistiéndose a mantenerse en un resultado óptimo local y llegando a una solución óptima a través de la evaluación de la experiencia individual y la experiencia colectiva de un grupo de agentes [2].

Este trabajo presenta las soluciones óptimas generadas por el algoritmo PSO bajo distintas perspectivas en el modelado del escenario. Una vez que se define una forma de modelar el escenario, se genera una ruta óptima evadiendo los obstáculos en el entorno, cubriendo la menor distancia en el recorrido y con la menor curvatura posible con el fin de que la trayectoria pueda ser descompuesta en movimientos simples de rotación y movimiento lineal para ser ejecutado por un robot móvil de tracción diferencial.

2. Metodología

Partiendo de tener un escenario supervisado por una cámara aérea mostrado en la figura 1a, donde un robot móvil tiene que navegar de un punto a otro con la consigna de incursionar ejecutando la trayectoria más corta y evadiendo los obstáculos de su entorno, la metodología para solución de este problema comprende desde la adquisición de la imagen y extracción de características que describan su entorno (figura 1b) hasta el cálculo de la trayectoria óptima y descomposición de su movimiento en movimientos simples para el robot. La figura 2 muestra la metodología desarrollada en este trabajo.



(a) (b)

Figura 1. Imagen Representativa del Problema de Estudio, (a) Escenario supervisado por cámara superior, (b) Descripción del Escenario por Procesamiento de Imágenes.

Figura 2. Diagrama de Flujo solución del Problema.

Partiendo de que la imagen del escenario es adquirida desde una vista superior, es necesario localizar donde se encuentra el robot, objetivo y los obstáculos. Para el escenario se eligen 3 colores distintos para cada tipo de objeto, tal que a través de segmentación se identifique la clase (Bloque Segmentación BGR).

Una vez segmentada la imagen se describe la forma de cada clase de objeto con el fin de conocer donde está ubicado actualmente en el caso del robot y el objetivo, y el área que cubre en el caso de los obstáculos (Bloque Obtención de Descriptores).

Dada la descripción de cada objeto se modela la forma en que una trayectoria interactúa con el entorno, tal que se pueda penalizar matemáticamente los comportamientos indeseados en el camino (Bloque Modelado del Escenario).

Conociendo si una ruta es adecuada o no para llegar de un punto a otro, se procede a la búsqueda de soluciones considerando múltiples trayectorias. Aquí es donde entra el algoritmo PSO, cuya consigna es encontrar puntos clave en el escenario, tal que el camino formado por ellos y que pudiera ser curvilíneo se encuentre sin obstrucciones.

Finalmente teniendo los puntos clave para generar el camino, se puede descomponer la trayectoria curvilínea en movimientos básicos para el robot como rotación pura y traslación pura, cuya estrategia de control a nivel eléctrico es sumamente simple.

A continuación se detallan cada uno de los bloques que conforman la metodología de solución.

3. Segmentación de Objetos por Modelo BGR

Los desarrolladores de OpenCV almacenan la información de color en el modelo BGR cuando se adquiere una imagen. La interpretación de una imagen en este modelo es igual al RGB, con la diferencia del orden de las capas de color. La segmentación consiste en representar un color en un rango mínimo y máximo de búsqueda para los tres colores, generando una máscara binaria para dicha información, tal que un objeto localizado sobre este modelo posteriormente pueda ser expresado por las características de un descriptor de contorno.

El modelo BGR al igual que el RGB expresa un color por la combinación de los tres colores primarios, tal que la menor y mayor intensidad de un color se da en un rango de 0 a 255. El color negro está conformado por tres ‘0’ mientras el blanco tres ‘255’. La figura 3 muestra la formación de colores secundarios a través de la combinación de primarios en el orden del modelo BGR.

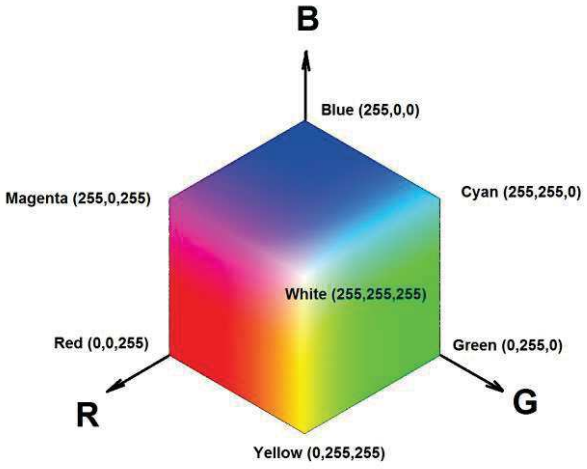
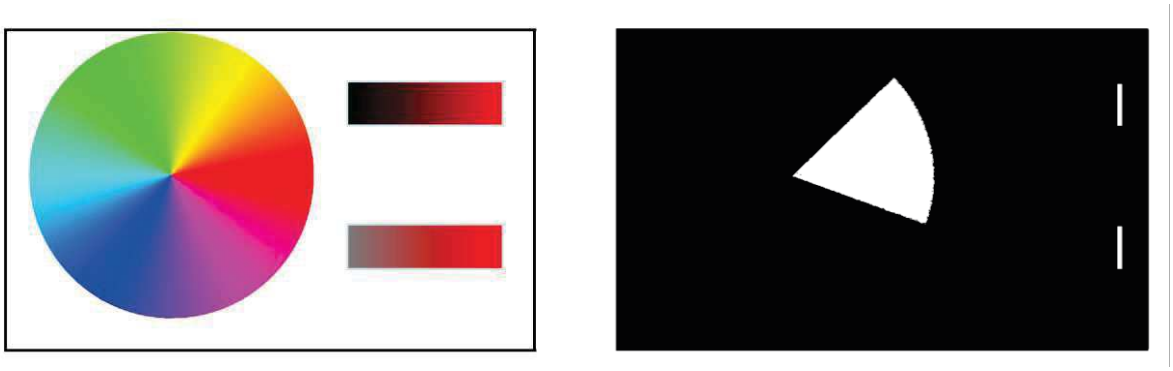


Figura 3. Representación de Colores en el Modelo BGR.

Una vez elegido un rango de color para las tres capas, se compara la imagen de color con los rangos elegidos, tal que sí cada pixel de la imagen se encuentra en el rango se expresará como un '1' o de lo contrario un '0' formando una imagen binaria representativa del objeto localizado. En la figura 4b se aprecia la generación de una imagen binaria a partir de la segmentación de la figura 4a sobre un rango BGR entre (0, 0, 252) y (73, 192, 255).



(a) (b)

Figura 4. Segmentación BGR, (a) Imagen original, (b) Imagen binaria resultado de la segmentación BGR.

4. Descriptores de Forma

Una vez que se ha localizado un objeto a través de segmentación es necesario describir su forma para relacionarlo con el objetivo de la generación de trayectoria para el robot móvil. Una manera de describir un objeto es a través de descriptores simples como Área, Perímetro y Rectangularidad. Otra manera es describirlos por su contorno ya sea identificando regiones (representación matricial), representaciones geométricas (convex hull), aproximaciones poligonales o formas invariantes (cónicas) [3].

4.1 Momentos

Los momentos de una imagen hacen referencia a propiedades geométricas de un objeto tales como área, perímetro, centroide, ángulo de rotación, etc. Entre las principales características que pueden describir la pertenencia de un punto a una forma geométrica se encuentran el área y el centroide, ya que a partir de estos se puede establecer una medida de que tan cerca se encuentra un punto en una región.

Los momentos de orden $p+q$ de una imagen $\mathrm{I}(\mathrm{u},\mathrm{v})$ son:

$$M_{p,q} = \sum_{u=1}^{N}\sum_{v=1}^{M} u^{p} v^{q} I(u,v) \tag{1}$$

donde su evaluación en una imagen binaria para p y q en cero genera el descriptor de área.

$$M_{0,0} = \sum_{u=1}^{N}\sum_{v=1}^{M} I(u,v) \tag{2}$$

En el caso de medición de centroide, el centroide de un objeto está definido como el punto de un objeto que tiene la misma cantidad de masa en cualquier dirección [4], que con respecto a los momentos está definido por:

$$\bar{u} = \frac{M_{1,0}}{M_{0,0}} \tag{3}$$

$$\bar{v} = \frac{M_{0,1}}{M_{0,0}} \tag{4}$$

4.2 Aproximación de Contornos

A través de procesamiento de imagen se puede obtener aproximaciones del contorno de una imagen a través de aproximaciones en base a líneas enlazadas. Un ejemplo de ello es el algoritmo de Douglas-Peucker, el cual se basa en la distancia entre puntos y líneas [5], donde una aproximación debe contener un subconjunto de los puntos de los datos originales y todos los puntos de datos originales deben estar dentro de una distancia de aproximación definida.

Dada una poli-línea P y una tolerancia ε como se representa en la figura 5a. El algoritmo de Douglas-Peucker tiene una estructura que comienza con el segmento de línea simple que une el primer v_1 y último v_n vértice de la poli-línea original (figura 5b). Los vértices restantes se prueban para ver si están cerca de su segmento aproximado. Si hay vértices más allá de una tolerancia ε especificada lejos del segmento, entonces el vértice v_i más alejado se agrega a la poli-línea simplificada previamente. Esto crea una nueva aproximación para la poli-línea original (figura xc). Recursivamente este proceso continúa para cada segmento de línea aproximado (figuras 5d) hasta que todos los vértices de la poli-línea original satisfagan la condición de proximidad (figura 5f).

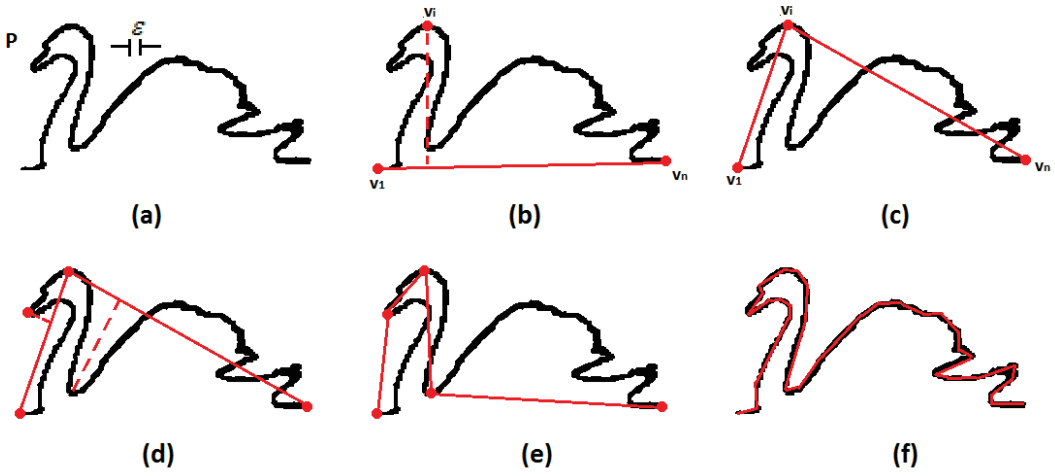


Figura 5. Algoritmo Douglas-Peucker, (a) Contorno a localizar, (b) Localización de vértice inicial v_1 y final v_n, (c) Búsqueda del siguiente vértice más alejada de ε, (d) Agregación de vértices con tolerancia ε, (e) Nueva aproximación de contorno, (f) Contorno final.

4.3 Convex Hull

Un contorno tipo Convex Hull está definido como el polígono más pequeño que contiene los puntos que forman una figura geométrica, ya sea que se encuentren contenidos en el borde de la figura geométrica o en su interior [6].

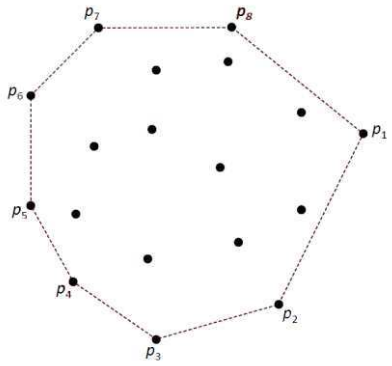


Figura 6. Contorno tipo Convex Hull.

4.4 Minimum Enclosing Circle

Este método consiste en encontrar el mínimo círculo que contenga todos los puntos de una figura. Uno de los métodos para su obtención es el algoritmo Sven Skyum, el cuál parte de un contorno Convex Hull para realizar la búsqueda de los vértices de un triángulo circunscrito [7]. El algoritmo consiste en realizar la comparativa de tres puntos consecutivos, tal que si el ángulo formado entre las líneas que lo forman genera un ángulo mayor a 90°, el punto intermedio se descarta (figura 7a). El procedimiento se repite para cada conjunto de tres puntos consecutivos, tal que si el ángulo formado entre ellos es menor o igual a 90° el punto se guarda como vértice (figuras 7a, 7c, 7d). Una vez terminando de analizar los n puntos, se termina con la obtención del circuncentro de los tres puntos que permanecieron, dando a lugar a un contorno circular que contiene a la figura geométrica en análisis (figura 7f).

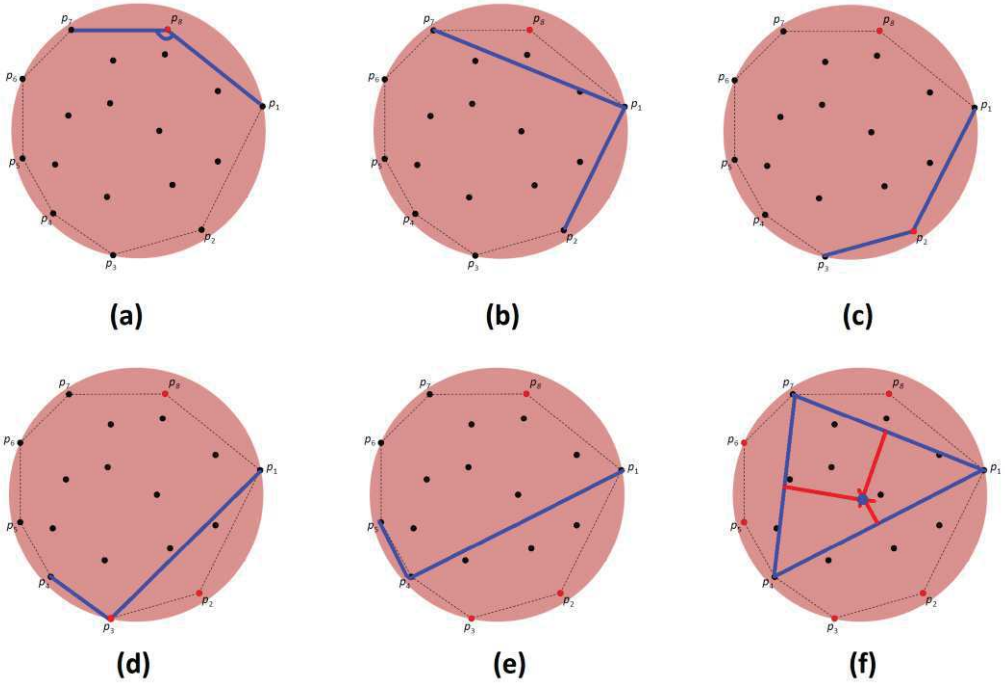


Figura 7. Algoritmo Sven Skyum para búsqueda del Mínimum Enclosing Circle, (a) Verificación de puntos consecutivos, (b) Descarte de punto intermedio y verificación de puntos consecutivos restantes, (c) Verificación de siguientes puntos consecutivos, (d) Descarte de punto intermedio y verificación de siguientes puntos consecutivos, (e) Descarte de punto intermedio y verificación de siguientes puntos consecutivos, (f) Ubicación de circuncentro con puntos no descartados.

Ingeniería y Automatización, Capítulo 3, pp. 26 - 44.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

4.5 Bounding Rectangle

Un Bounding Rectangle es el mínimo rectángulo que contiene a una forma geométrica y se puede obtener a través de un contorno descrito, tal que la línea horizontal inferior se relaciona a las coordenadas del punto de borde más bajo en la imagen; la línea horizontal inferior se relaciona a las coordenadas del punto de borde más alto en la imagen; las líneas vertical izquierda y vertical derecha se relacionan con los puntos de borda más alejados a la izquierda y derecha respectivamente. Las características que ofrece el Bounding Rectangle son las coordenadas (u,v) correspondientes a la esquina superior izquierda y (w,h) el ancho y alto en pixeles del rectángulo.

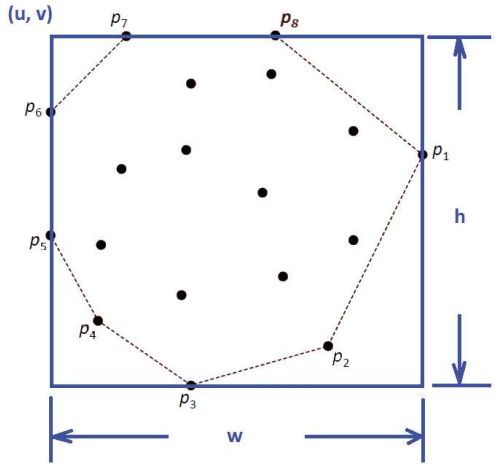


Figura 8. Obtención de un Bounding Rectangle a partir de un descriptor de contorno.

4.6 Fitting an Ellipse

Un método para obtener los parámetros de una elipse que contenga la mayor cantidad de puntos de una forma geométrica es el cálculo a partir de la obtención de un Bounding Rectangle. El método consiste en realizar un escaneo de izquierda a derecha y de arriba abajo para detectar los pixeles que están sobre las fronteras del Bounding Rectangle (*etop, eright, eleft, ebottom*), de tal manera que los valores de frontera encontrados (figura 9) forman parte de una de posibles elipse y que pueden ser descartados o actualizados comparando su centro con el centro del Bounding Rectangle [8].

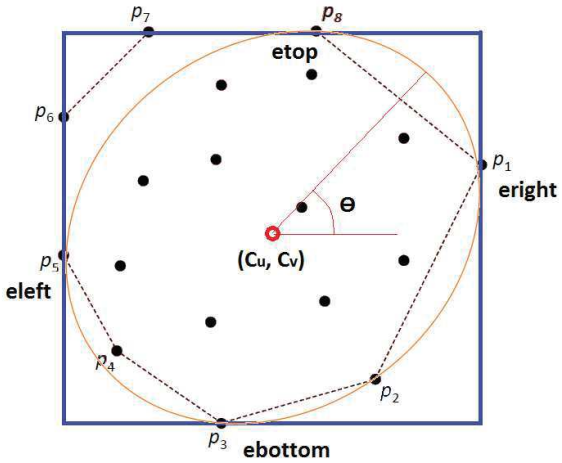


Figura 9. Obtención de una Elipse ajustada a la mayor cantidad de puntos de una figura.

Este trabajo se centra en describir la forma en que una trayectoria interactúa con los objetos del entorno usando los descriptores de centroide y Minimum Enclosing Circle; Area; Bounding Rectangle y Convex Hull.

5. Modelado del Escenario

El modelado del escenario con respecto al objetivo de generar una trayectoria con evasión de obstáculos, consiste en describir matemáticamente sí una trayectoria pasa por el rango de cobertura de un obstáculo, asignando una penalización para dicho evento. La penalización es asignada con respecto a las características que ofrece el descriptor de contorno elegido.

5.1 Secciones cónicas

El modelado a través de secciones cónicas consiste en encerrar cada obstáculo de la imagen en una región circular a través del Minimum Enclosing Circle y un Centroide, tal que la penalización se asigna con respecto a si la trayectoria pasa por el radio de cobertura de cada obstáculo, esto es mediante la comparación de la distancia del centroide a los puntos intermedios de la trayectoria T y el radio del obstáculo $Robs$, tal y como se muestra en la figura 10.

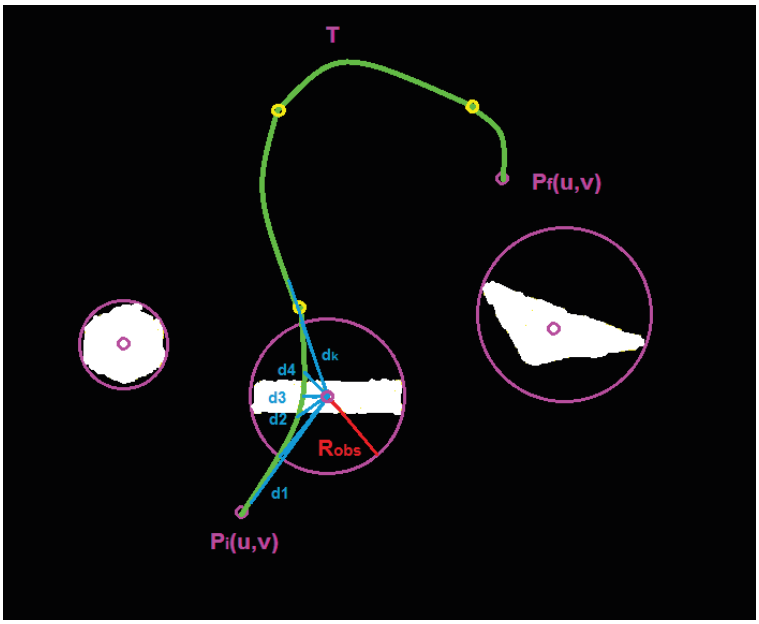


Figura 10. Trayectoria sobre una imagen segmentada por modelo BGR y descrita por Minimun Enclosing Circle y Centroide.

Para lo cual la penalización puede estar dada por:

$$d_{i,j} = \sqrt{\left(T_{u,i} - C_{u,j}\right)^2 + \left(T_{v,i} - C_{v,j}\right)^2} \tag{5}$$

$$P_{i,j} = \begin{cases} 1 - \dfrac{d_{i,j}}{Rrob + Robs_j} & sí \quad d_{i,j} \le Rrob + Robs_j \\ 0 & \text{de otro modo} \end{cases} \tag{6}$$

$$Violation = \frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N} P_{i,j} \tag{7}$$

Donde $d_{i,j}$ representa la distancia entre el i-ésimo punto intermedio de la trayectoria T y el centroide del *j-ésimo* obstáculo; $Rrob$ representa el radio del robot móvil que cubrirá la trayectoria T; $Robs_j$ representa el radio de cobertura del j-ésimo obstaculo; y por tanto el puntaje de la penalización es la suma de las medias de las distancias entre los puntos de la trayectoria y el centroide del obstáculo que se encuentra en la cobertura circular dada por el Minimum Enclosing Circle.

5.2 Regiones

A partir de las regiones R que conforman los obstáculos en un escenario generado por una imagen binaria pre-procesada en BGR, se puede dar una penalización a las trayectorias T cuyos puntos intermedios pasen por las regiones R tal y como se ejemplifica en la figura 11.

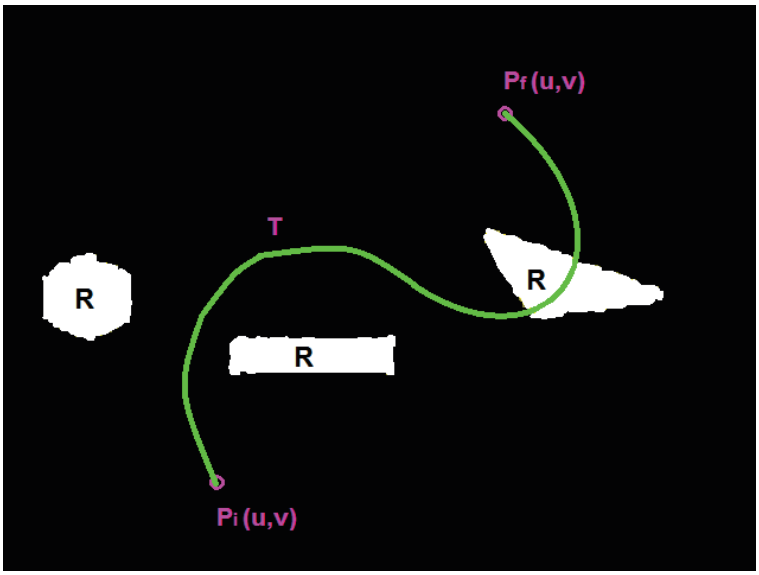


Figura 11. Trayectoria sobre una imagen segmentada por modelo BGR y descrita por Regiones Binarias.

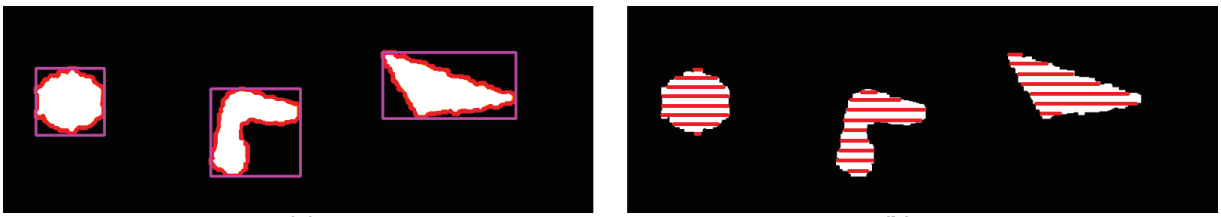
La función de penalización se puede dar por la suma de pixeles en blanco en la localización (u,v) de los N puntos intermedios de la trayectoria T, tal que:

$$P(i) = \begin{cases} 1 & \text{sí } T_i(u,v) \subset R \\ 0 & \text{de otra manera} \end{cases} \tag{8}$$

$$Violation = \sum_{i=1}^{N} P(i) \tag{9}$$

5.3 Eje Y

A través de la unión de los descriptores Convex Hull y Bounding Rectangle (figura 12a) se puede codificar la región de estudio en un conjunto de listas por renglón en donde cada lista tiene las coordenadas x donde la línea sale y entra de la región (figura 12b).



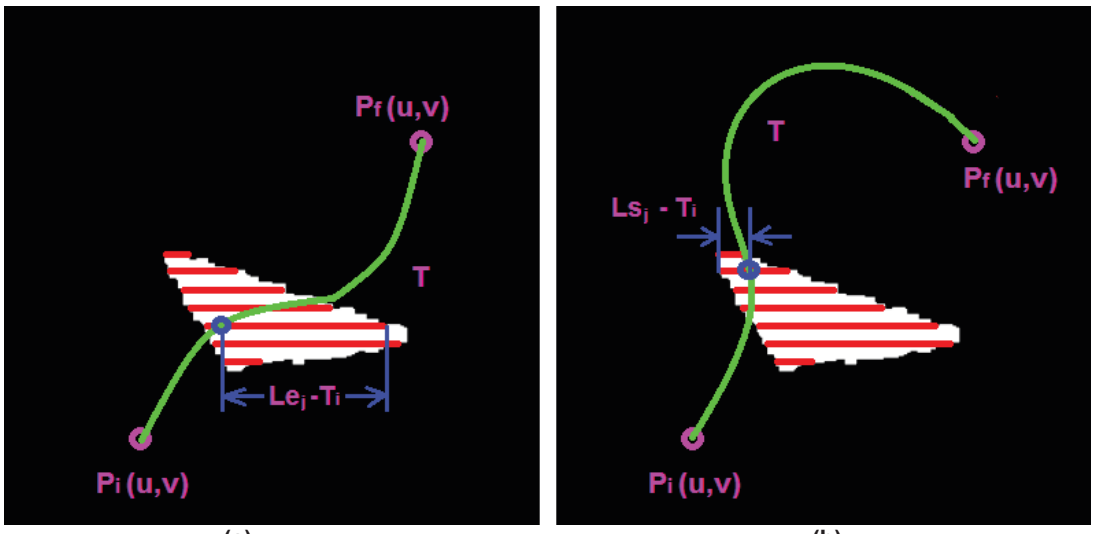
(a) (b)

Figura 12. Descriptor Eje Y, (a) Ubicación de contorno y bounding rectangle en imagen segmentada, (b) Descomposición de objetos en listas Eje Y, a partir de (a).

La penalización se puede dar por la distancia en que cada uno de los M puntos de la trayectoria T se aleja del borde del objeto, tal que:

$$P(i,j)=\begin{cases}|Le_j - T_i| & \text{si } T_i - T_{i-1} \geq 0 \quad \text{y} \quad T_i \subset L_j \quad \text{para } j = 1,2,3,\ldots,N \\ |Ls_j - T_i| & \text{si } T_i - T_{i-1} < 0 \quad \text{y} \quad T_i \subset L_j \\ 0 & \text{de otro modo}\end{cases} \tag{10}$$

$$Violation = \sum_{i=1}^{M} P(i,j) \tag{11}$$



(a) (b)

Figura 13. Penalización respecto a la dirección de la trayectoria, (a) Trayectoria entrante al obstáculo, (b) Trayectoria saliente al obstáculo.

Este criterio genera una mayor penalización cuando la dirección de la trayectoria en el eje x se encuentra entrando al objeto (figura 13a), y de manera contraria menor penalización cuando la trayectoria está saliendo de él (figura 13b).

6. Algoritmo PSO

El algoritmo básico de Enjambre de Partículas (PSO), consiste en que n partículas, donde cada partícula es un vector de datos que representa una solución potencial para un problema. Las partículas rigen su comportamiento en base a tres principios:

1. Conservar su inercia al memorizar la dirección de vuelo anterior.
2. Cambiar su condición de acuerdo a la mejor solución óptima.
3. Cambiar su condición de acuerdo a la mejor solución óptima del enjambre.

El algoritmo puede ser representado de forma sintética como sigue:

```
Inicializar_Partículas();                    % Generar posiciones aleatorias para
                                             % cada partícula, considerar
                                             % velocidades iniciales en cero y
                                             % obtener la mejor posición global a
                                             % partir de ellas
Para i=1 hasta número_iteraciones
        Para cada partícula p en Enjambre Hacer
                fp = f(p);                   % Calcular la función aptitud.
                Sí fp es mejor que f(pBest)  % Comparación entre la aptitud actual
                                             % y la mejor aptitud hasta el momento.
                        pBest = p;           % Actualizar mejor posición individual.
                fin Sí
        fin  Para
        gBest = la mejor p en Enjambre;      % Comparar y actualizar mejor posición
                                             % global entre la mejor posición global
                                             % actual y las mejores posiciones hasta el
                                             % momento em enjambre
        Para cada partícula p en Enjambre Hacer
                Calcular nueva velocidad individual
                Calcular nueva posición individual
        fin Para
fin Para
```

Para actualizar las velocidades y posiciones de cada partícula se adoptan las ecuaciones:

$$V_j^i(t+1) = w(t)V_j^i(t) + c_1 r\left[pbest_j^i(t) - P_j^i(t)\right] + c_2 r\left[gbest_j(t) - P_j^i(t)\right] \tag{12}$$

$$P_j^i(t+1) = P_j^i(t) + V_j^i(t) \tag{13}$$

$$w(t+1) = \alpha w(t) \quad , \quad 0 < \alpha < 1 \tag{14}$$

Donde $V_j^i(t)$, $P_j^i(t)$ representan la velocidad y posición de la partícula en el instante t; $pbest_j^i(t)$ representa la mejor posición individual de cada partícula hasta el instante t; $gbest_j(t)$ representa la mejor posición entre todas las partículas hasta el instante t; c_1 y c_2 representan la ponderación atribuida al mejor resultado personal de cada partícula y al mejor resultado entre todos las partículas; r es un valor aleatorio entre 0 y 1 que permite una tendencia de zig-zag en la búsqueda de una solución y previene la convergencia prematura sobre un punto no optimo; $w(t)$ representa la habilidad

de búsqueda de la partícula, cuando $w(t)$ es grande permite la dispersión de las partículas en un gran espacio de búsqueda, encontrando rápidamente la solución más óptima, conforme pasa el tiempo puede ser reducida en forma descendente para realizar la búsqueda de forma más minuciosa [9].

6.1 Consideraciones de Velocidad y Factor de Convergencia

Para acelerar la convergencia del PSO se le agrega un factor de convergencia χ propuesto y analizado por Clerc y Kennedy [10], tal como se muestra en la ecuación 15.

$$\chi = \frac{2\kappa}{\left|2-\phi-\sqrt{\phi^2-4\phi}\right|} \quad , \quad 0 \le \kappa \le 1 \quad , \quad \phi_1+\phi_1 \ge 4 \tag{15}$$

Donde κ es una constante arbitraria [0,1] y regula la velocidad de convergencia del algoritmo; las constantes c_1 , c_2 y $w(t)$ del PSO original son calculadas a partir de las variable χ y ϕ, tal que:

$$w(t) = \chi \tag{16}$$

$$c_1 = \chi\phi_1 \tag{17}$$

$$c_2 = \chi\phi_2 \tag{18}$$

Adicionalmente al factor de convergencia se le puede agregar un límite de velocidad antes de actualizar las partículas, generando un rango de posiciones posibles que se encuentren entre las fronteras admitidas para la posición de las mismas. En nuestro caso los límites de velocidad admitidas están dadas por:

$$V_{i,\max} = \beta\left(P_{i,\max} - P_{i,\min}\right) , \quad 0 < \beta \le 1 \tag{19}$$

$$V_{i,\min} = -V_{i,\max} \tag{20}$$

donde

$$V_j^i(t+1) = \min\left[V_j^i(t+1), V_{i,\max}\right] \tag{21}$$

$$V_j^i(t+1) = \max\left[V_j^i(t+1), V_{i,\min}\right] \tag{22}$$

6.2 Elección de la Función Aptitud

El problema de la trayectoria se puede definir como aquella curva que vaya de un punto inicial $P_i(u,v)$ a un punto final $P_f(u,v)$ evadiendo M obstáculos en su trayecto y proporcionando la ruta más corta.

Para lograr la trayectoria más corta se puede medir la distancia del recorrido de N puntos intermedios a través de:

$$L = \sqrt{(u_i - u_{i-1})^2 + (v_i - v_{i-1})^2}\ ; \quad i = 2,3,4,\ldots\ \ldots, N \tag{23}$$

$$F = L * (1 + beta * Violation) \tag{24}$$

Donde u_i y v_i representan un punto de la trayectoria solución y L la suma de las distancias euclideanas entre dos puntos [11].

Como se aprecia en la ecuación (24) la función aptitud depende tanto de la longitud de la trayectoria como de la función de penalización del descriptor de contorno.

Ahora bien, si se considera la movilidad de un robot de tracción diferencial se puede agregar una restricción adicional, tal que los puntos intermedios que conforman la ruta tengan la menor curvatura posible para que el sistema de control del robot móvil realice el menor esfuerzo para llegar de un punto a otro. Sí se considera que la diferencia entre los puntos de L es una derivada discreta, se puede asumir que un descriptor de curvatura está dada por la gradiente de L (ecuación 25).

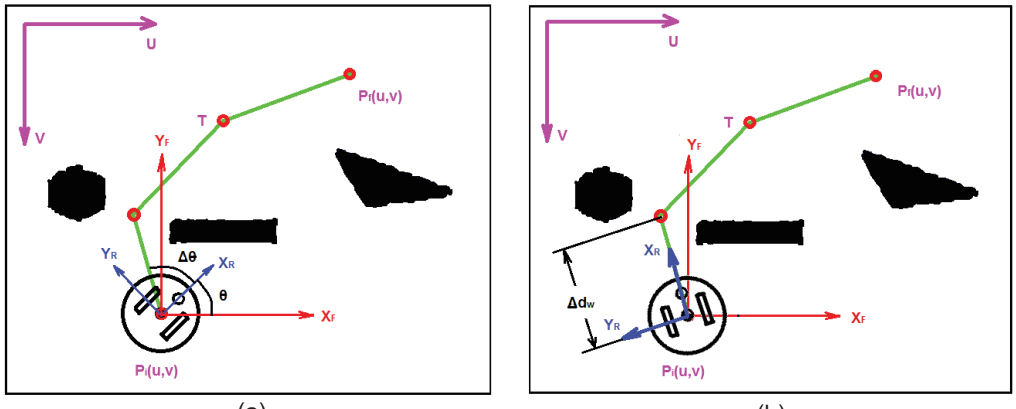
$$\nabla L = \sqrt{(u_i - 2u_{i-1} + u_{i-2})^2 + (v_i - 2v_{i-1} + v_{i-2})^2}\ ; \quad i = 3,4,5,\ldots\ \ldots, N \tag{25}$$

$$F = L * (1 + beta * Violation) + gamma * \nabla L \tag{26}$$

Entonces la nueva función aptitud puede estar dada por la ecuación (26), orientando el resultado al trayecto más corto que evada todos los obstáculos y con la menor curvatura posible entre puntos intermedios.

7. Descomposición de Trayectoria en Movimientos Rotatorios y Lineales

Una vez obtenida la trayectoria más corta y con menor curvatura libre de obstáculos, cada punto intermedio de T forma a ser un punto al que debe llegar el robot, tal que para llegar $P_i(u,v)$ hasta $P_f(u,v)$, se toma como punto inicial el punto donde se encuentra actualmente el robot y como punto final el siguiente punto intermedio, generando primero una rotación $\Delta\theta$ (figura 14a) y posteriormente en un desplazamiento lineal Δd_W en dirección de X_R (figura 14b). El procedimiento se repite hasta llegar a $P_f(u,v)$.



(a) (b)

Figura 14. Descomposición de Trayectoria, (a) Movimiento rotatorio, (b) Movimiento lineal.

7.1 Consideraciones Cinemáticas

Suponiendo que se tiene un controlador para las llantas izquierda y derecha de un ´robot de tracción diferencial que se mueve en el eje de coordenadas de la imagen y tomando como eje de referencia global (X_F, Y_F) con origen en $P_i(u,v)$. Las relaciones cinemáticas están dada por:

$$\xi_F = R^{-1}\xi_R \tag{27}$$

$$R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{28}$$

$$\xi_P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \xi_F \tag{29}$$

Donde ξ_R, ξ_F, ξ_P son las posiciones representativas en el eje de referencia del Robot, eje global y eje de las imágenes respectivamente, y por lo cual una velocidad expresada en el eje de las imágenes puede ser transformada al eje de referencia del robot. Así mismo partiendo de estas relaciones se puede expresar la relación de velocidades y a su vez su descomposición en velocidades individuales para los neumáticos [12].

$$\dot{\xi}_F = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = f\left(l, r, \theta, \dot{\varphi}_1, \dot{\varphi}_2\right) \tag{30}$$

$$\dot{\xi}_F = R^{-1}\begin{bmatrix} \frac{r\dot{\varphi}_1}{2} + \frac{r\dot{\varphi}_2}{2} \\ 0 \\ \frac{r\dot{\varphi}_1}{2l} + \frac{r\dot{\varphi}_2}{2l} \end{bmatrix} \tag{31}$$

Con:

r - radio de los neumáticos.

l - distancia del centro de masa al eje de los neumáticos.

$\dot{\varphi}_1$ - velocidad angular del neumático izquierdo.

$\dot{\varphi}_2$ - velocidad angular del neumático derecho.

8. Resultados

Para el modelado del escenario se tomaron en cuenta dos situaciones, la primera tomando en cuenta la trayectoria más corta y un segundo tomando en cuenta la trayectoria más corta con menor grado de curvatura, ambas sin considerar el tamaño del robot en el escenario. En las tablas 1 y 2 se muestra la evolución del algoritmo para ambos casos, comparando los resultados para el modelado usando cónicas, regiones y codificación Eje-Y. Para ambas situaciones mostró menor costo la codificación Eje-Y, sin embargo el costo en tiempo de cómputo en la codificación es enorme a comparación de los otros dos.

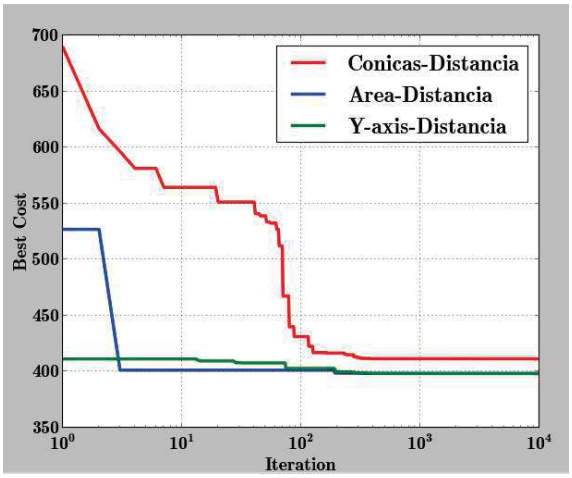
Tabla 1. Evolución del PSO para trayectoria más corta.

Modelado	Iteraciones	Costo (px)	Violación (ua)	Distancia (px)	Curvatura (px)
Secciones Cónicas (a)	100	431.249623839	0.00762939446383	642.928570971	60.3279945018
	500	411.449621394	0.0302797270293	415.26795419	17.8052976145
	1,000	411.437294407	0.0	411.437294416	19.4381432569
	5,000	411.437294406	2.22044604925e-18	411.437294406	19.4381432439
	10,000	411.437294406	2.22044604925e-18	411.437294406	19.4381432439
Regiones (b)	100	401.405012768	3570	492.832546851	46.2815471915
	500	398.420669458	255	398.417646087	11.958948207
	1,000	398.419294088	0	398.419294527	11.9506374913
	5,000	398.412416593	0	398.412416593	11.8382062743
	10,000	398.412416593	0	398.412416593	11.8382062743
Eje Y (c)	100	402.898143652	637.457263256	596.998105804	24.2996098706
	500	398.285406724	0.99107777974	398.245284193	8.38439562835
	1,000	398.279686377	0.913204323382	398.279686377	8.24393185342
	5,000	398.262031325	0	398.262031325	7.80278283413
	10,000	398.262030986	0	398.262030986	7.80278310321

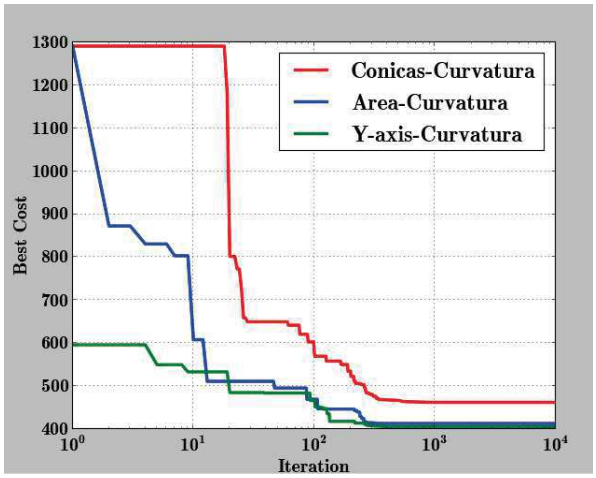
Tabla 2. Evolución del PSO para la trayectoria más corta y menor grado de curvatura.

Modelado	Iteraciones	Costo (px)	Violación (ua)	Distancia (px)	Curvatura (px)
Secciones Cónicas (d)	100	579.466519947	0.00616224696587	644.173149865	35.7303579377
	500	467.165914484	0.0	419.456089174	4.7936969267
	1,000	462.614192386	0.0	418.777127705	4.38444616228
	5,000	462.585460493	0.0	418.801056099	4.37844043946
	10,000	462.585460493	0.0	418.801056099	4.37844043946
Regiones (e)	100	469.835399533	0	508.844863098	30.9600925843
	500	413.44495164	255	398.707186034	1.47197203143
	1,000	413.441871124	0	398.712020374	1.47298507502
	5,000	413.441871124	0	398.712020374	1.47298507502
	10,000	413.441871124	0	398.712020374	1.47298507502
Eje Y (f)	100	466.636566978	384.298540176	815.936330995	31.6500524774
	500	405.927962129	0.0405078239244	398.659008848	0.816017806858
	1,000	405.927530326	8.059424772e-08	398.532787291	0.739474300709
	5,000	405.927530321	0	398.532786528	0.739474379349
	10,000	405.927530321	0	398.532786528	0.739474379349

En la figura 15a se muestra la evolución del costo en el algoritmo PSO para el cómputo de la trayectoria más corta y la figura 15b para el computo de la trayectoria más corta y con menor grado de curvatura.



(a)



(b)

Figura 15. Comparativa entre modelos, (a) Evolución del PSO para la trayectoria más corta, (b) Evolución del PSO para la trayectoria más corta y menor grado de curvatura.

La figura 16 muestra la trayectoria inicial de la cual parten ambos modelos para los tres casos considerando tanto obstáculo de geometría regular como irregular.

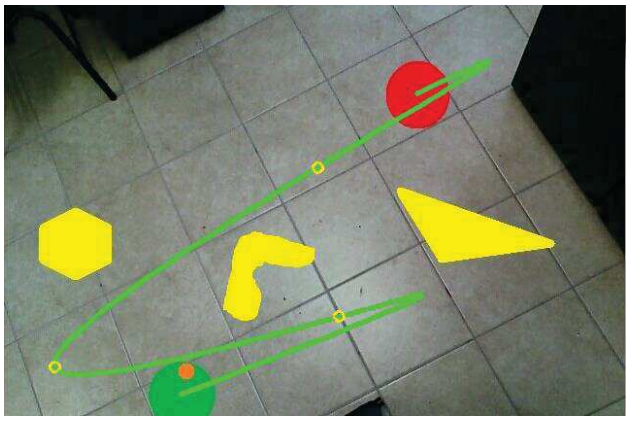


Figura 16. Mejor Trayectoria generada en la inicialización del PSO.

Las figuras 17a, 17b y 17c muestran la trayectoria final para el caso de la ruta más corta, mientras las figuras 17d, 17e y 17f muestran la trayectoria final para el caso de la ruta más corta con menor grado de curvatura. Adicionalmente en las figuras 17g, 17h y 17i se muestra trayectorias finales considerando tanto la trayectoria más corta como el menor grado de curvatura además de considerar una tolerancia de paso para la geometría del robot móvil. En todos los casos la trayectoria final es mostrada tras 10,000 iteraciones.

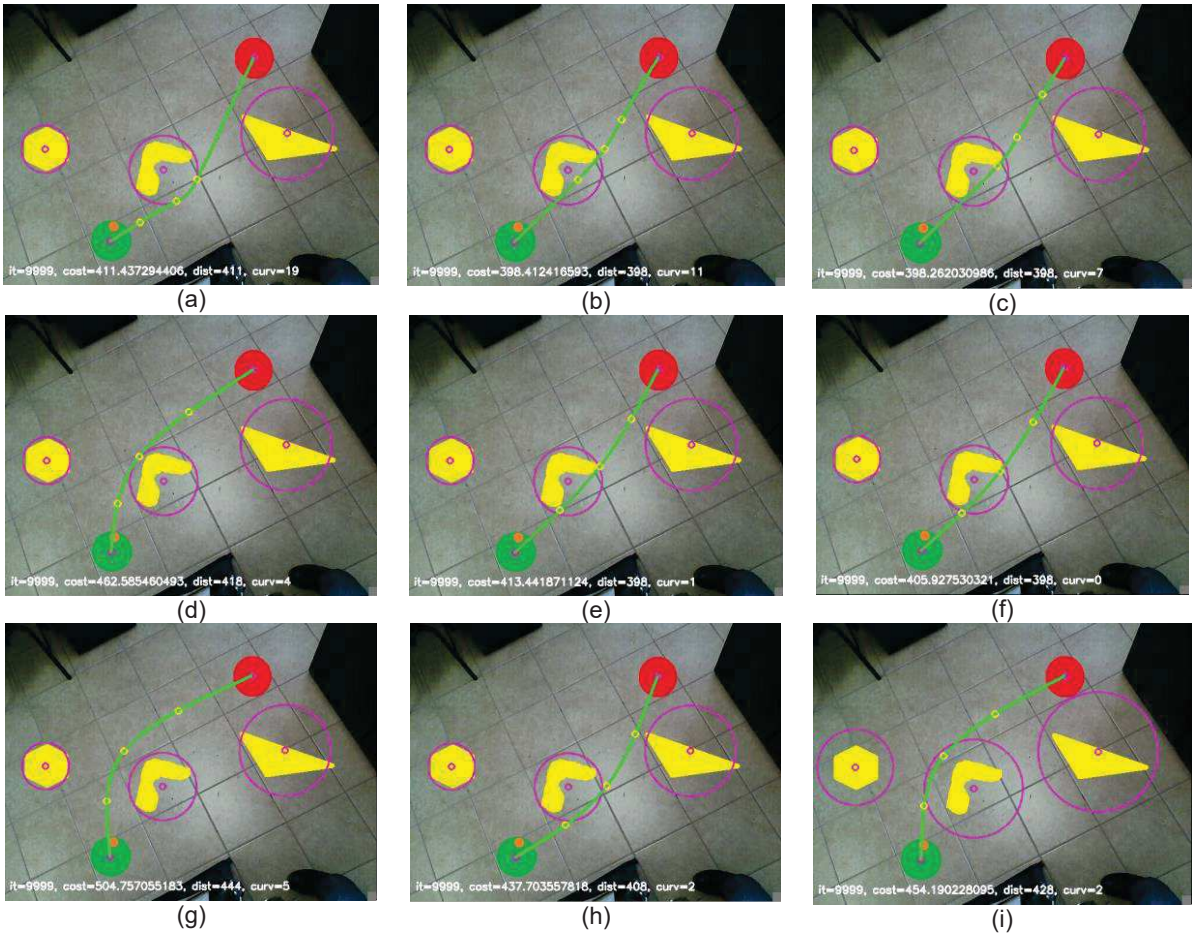
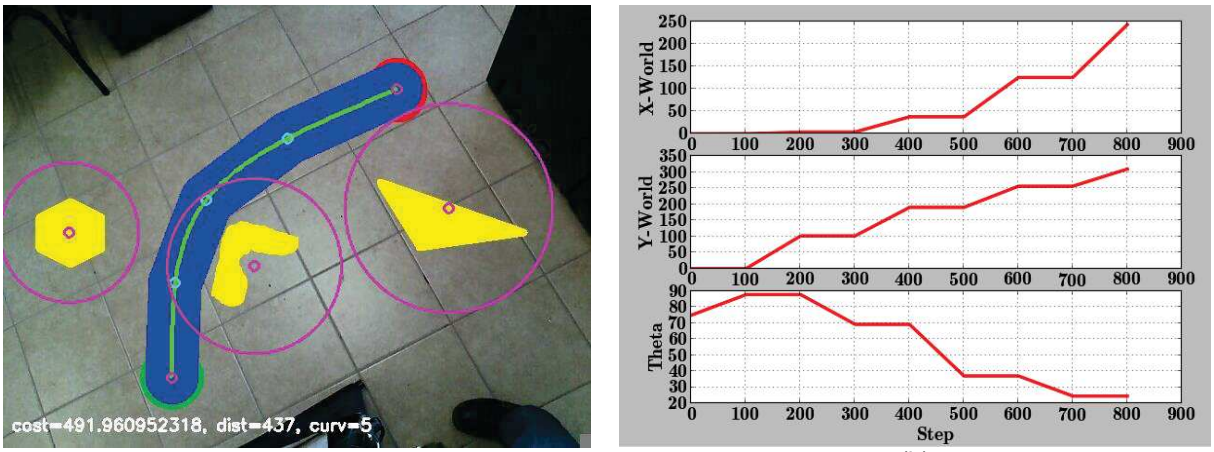


Figura 17. Mejor Trayectoria Obtenida con obstáculos ficticios, (a) Trayectoria más corta modelado por cónicas, (b) Trayectoria más corta modelado por regiones, (c) Trayectoria más corta modelado por codificación Eje-Y, (d) Trayectoria más corta con menor grado de curvatura modelado por cónicas, (e) Trayectoria más corta con menor grado de curvatura modelado por regiones, (f) Trayectoria más corta con menor grado de curvatura modelado por codificación Eje-Y, (g) Trayectoria más corta con menor grado de curvatura y tolerancia para el robot modelado por cónicas, (h) Trayectoria más corta con menor grado de curvatura y tolerancia para el robot modelado por regiones, (i) Trayectoria más corta con menor grado de curvatura y tolerancia para el robot modelado por codificación Eje-Y.

Para la ejecución de una trayectoria se eligió la modelación por regiones considerando el criterio de menor distancia, menor grado de curvatura y una tolerancia para el cuerpo del robot. El modelado por regiones se eligió puesto que ofrece una solución intermedia en cuanto a la minimización del costo y el tiempo de procesamiento es mínimo con respecto a la codificación Eje-Y.

El algoritmo PSO fue implementado para una población de 20 partículas, 3 puntos intermedios, $\kappa = 0.8$, $\phi_1 = 2.05$ y $\phi_2 = 2.05$.

La figura 18a muestra la simulación de la ejecución de la trayectoria de un robot móvil, descomponiendo los puntos intermedios originados por el PSO. Por otra parte la figura 18b muestra por iteraciones los movimientos graduales del robot y su descomposición en movimientos rotatorios y lineales en un eje de coordenadas global (X_F, Y_F).



(a) (b)

Figura 18. Ejecución de trayectoria óptima, (a) Evolución del movimiento del robot móvil, (b) Descomposición de trayectoria en movimientos rotativos y lineales.

9. Conclusiones

Este trabajo mostró el comportamiento del algoritmo PSO modelando el escenario de un Robot Móvil con la presencia de obstáculos virtuales de geometrías regulares y no regulares. La imagen del escenario fue adquirida por una webcam convencional y procesada en openCV describiendo el entorno de un robot móvil, modelando el escenario por cónicas, regiones y Eje-Y.

El modelado del escenario por codificación Eje-Y presentó menor costo con respecto al modelado por regiones y por cónicas, sin embargo el tiempo de cómputo en la codificación fue muy elevado dada la búsqueda de orígenes y finales por renglón para cada objeto en una imagen binaria. El menor costo en la modelación fue por Eje-Y atribuido a la penalización adicional cuando una trayectoria va entrando o va saliendo hacia una región obstáculo. El modelado por regiones aunque no resultó el mejor, su cómputo es muy simple y presentó alternativas aceptables y más cerradas en comparación con cónicas.

Los puntos intermedios de la trayectoria fueron generados por el algoritmo PSO y usados como puntos de control para que un Robot Móvil de Tracción Diferencial ejecutara la trayectoria descomponiendo su trayecto en movimientos rotativos y lineales. Finalmente se mostró una comparativa de las trayectorias óptimas generadas con los descriptores cónicas, regiones y Y-axis partiendo de una misma trayectoria aleatoria.

Referencias

[1] Madic M., Markovik D., Radovanovic M. “Comparison of Meta-Heuristic Algorithms for Solving Machining Optimization Problems”, FACTA Universitatis, Mechanical Engineering, Vol. 11, N° 1, 2013, pp. 29-44.

[2] Kokash N. “An introduction to heuristic algorithms”, Department of Informatics and Telecommunications, University of Trento, Italy.
[3] Morse B. “Shape Description (Contours)”, Brigham Young University, 2000. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/boundary-rep-desc.pdf
[4] Hu M. “Visual Pattern Recognition by Moment Invariants”, IRE Transactions on Information Theory, 1962.
[5] Wu S., da Silva A., Márquez M. “Sufficiency Conditions for Non-Self-Intersections”, 2006.
[6] Talasila J. “Convex Hull”, Indiana State University, 2014. http://cs.indstate.edu/~jtalasila/convexhull.pdf
[7] Sven Skyum’s Algorithm for Computing the Smallest Enclosing Circle, Department of Computer Science, Aarhus University, 2014. http://www.cs.au.dk/~gerth/slides/sven14.pdf
[8] Chang C. “Detecting Ellipses via Bounding Boxes”, Asian Journal of Health and Information Sciences, Vol.1, No.1, pp.73-84, 2006.
[9] Bai Q. “Analysis of Particle Swarm Optimization Algorithm”, College of Computer Science and Technology Inner Mongolia University for Nationalities, China, 2010.
[10] Clerc M., Kennedy J. “The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space”, IEEE Transactions on Evolutionary Computation, Vol. 6, No. 1, 2002.
[11] Malathi M., Ramar K., Paramasivam C. “Optimal Path Planning for Mobile Robots Using Particle Swarm Optimization and Dijkstra Algorithm with Performance Comparison”, Middle-East Journal of Scientific Research 24, pp: 312-320, 2016.
[12] Siegwart R., Nourbakhsh I. “Introduction to Autonomous Mobile Robots”, The MIT Press, USA, pp: 48-52, 2004

Ingeniería y Automatización, Capítulo 4, pp. 45 - 58.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Propuesta de Robot Serial de Servicio para Tender Camas de Hospital

Lasso Perdomo Cristiam Javier ([1]), Sastoke González Stephany ([1]),
Peña Medina Erik ([2]) y Neftali Elorza Lopez ([3])

Universidad Tecnológica de Pereira, Colombia ([1]), Facultad de Ingeniería, México ([2]), Facultad de Estudios Superiores Aragón ([3])
cristiam.lasso@utp.edu.co
ssastoke@utp.edu.co
epm01@live.com.mx
foxnek@comunidad.unam.mx

Resumen

El presente trabajo muestra el planteamiento para determinar los parámetros para que un robot serial destienda la cama de un hospital. El planteamiento consta de modelar un robot serial de cinco grados de libertad, posteriormente se establecieron los parámetros de las trayectorias para que el robot realice la tarea propuesta. Con la información obtenida se establecieron restricciones para que el robot realice la tarea. Para verificar el robot puede realizar la tarea se realizó una simulación y se observó el comportamiento de las juntas del robot.

Palabras clave: Manipulador serial, robot hospitalario, plegado sábanas.

1. Introducción

Los robots son sistemas mecatrónicos, con cierto grado de autonomía, y el impacto que ha tenido la robótica desde sus inicios se ha convertido en la rama del saber con características propias que ha cautivado el interés de muchas de las otras líneas, que se embarcan en la búsqueda de nuevas alternativas a la solución de sus problemas [1]. Una planificación de trayectorias normalmente tiene tres tipos de resultados: rutas, caminos y trayectorias. Una ruta es un conjunto ordenado de configuraciones que deben ser alcanzadas por el robot. Por camino se entiende la discretización de una función continua que interpola las configuraciones definidas en una ruta. Finalmente, cuando se habla de trayectoria temporal (por omisión simplemente trayectoria) se está haciendo referencia a un camino que tiene asociado un perfil cinemático; es decir, a cada configuración perteneciente al camino se le asocia una velocidad [2].

En 1980 comienzan los experimentos quirúrgicos en neurocirugía y ortopedia, en 1985 el robot PUMA 560 introduce una aguja en un cerebro mediante un guía de TAC [3], en 1986 comienza las investigaciones para implante robotizado en prótesis de cadera en el Centro de Investigación de Thomas Wattson de IBM, estos y algunos avances más, esto para nombrar algunos de los primeros avances
Clasificación: a partir del 2007, aparecen una diversidad de robots cirujanos, pueden ser: robots servo-asistentes (AESOP y otros brazos asistentes), asistentes coordinadores (Hermes), efectores semiautónomos (Robodoc, Caspar, Acrobot, Probot, PAKI) y los robots tele manejados (da Vinci) [4].

Para el estudio de los parámetros y determinación de trayectoria, se ha utilizado para la generación de gráficos con el uso de por herramientas computacionales.

En el campo de la medicina, las aplicaciones en el campo asistencial y de rehabilitación que han surgido han mostrado un papel fundamental y uno de los resultados más relevantes es la unión de dos áreas, que tienen un objetivo en común, al beneficio humano, con la unión de la ingeniería y la

medicina estas dos áreas han logrado grandes avances, que dan como resultado diversos robots al servicio de los pacientes.

2. **Clasificación**

En cuanto a la robótica médica se puede referir lo siguiente:

- Robótica Asistencial: El objetivo de los robots asistenciales, es mejorar la calidad de vida de las personas con capacidades diferentes o enfermas, brindándoles mayor autonomía y seguridad. Por lo general, los robots destinados a este fin, están diseñados para entornos específicos (hospitalarios, residencia el primer robot de esta clasificación es MANUS [4].

- Robótica en Rehabilitación: La rehabilitación es el conjunto de procedimientos dirigidos a ayudar a una persona a alcanzar el más completo potencial físico, psicológico, social, vocacional y educacional compatible con su deficiencia fisiológica o anatómica y limitaciones medioambientales, Monark 871E Trainer Rehab [5].

Otra de las clasificaciones que le puede atribuir a este tipo de robots hospitalarios, en acorde con su funcionalidad traslado, asistencia fija, locomoción del usuario.

- Robots de locomoción del usuario: Este tipo de los robots asiste, buscan mejorar la calidad de la recuperación del paciente con la facilidad de otorgar mayor seguridad y una estandarización a la hora de brindar un procedimiento de recuperación, este robot tiene la particularidad de que son visualmente parecidos a la imagen humanoide brindando así una mejor adaptación y aceptación con el usuario un claro ejemplo, es lo que está trabajando la empresa RF Curl, HW Kroto y RE Smalley, J McCarthy del Stanford Artificial Intelligence Laboratorio pública una computadora con manos (capacidad de manipulación), ojos (cámara para movimiento) y oídos (reconocimiento de órdenes verbales) [6].

- Robots de asistencia fija: Estos buscan mejorar la precisión es utilizado como brazo robótico en la mayoría de los casos caracterizado por su geometría tal como la empresa Y Wang funda en 1989 la empresa Computer Motion inc. especializada en la construcción de robots quirúrgicos; En 1992 se presenta en Inglaterra Robodoc, el primer asistente mecánico de la cirugía de artroplastia de cadera y rodilla [7], otro de los ejemplos más conocidos es el Robot da Vinci que es el avance más significativo hasta ahora conseguido con los robots de este tipo [8].

- Robots de servicio: La robótica de servicios de traslado, tiene como límites naturales hasta el momento, los de los interiores de los edificios e instalaciones privadas debido a las normativas que rigen, la propia complejidad técnica del entorno del transporte, su objetivo principal es el ahorro de recursos, y la optimización de los tiempos [9].

2.1 *Planteamiento del problema*

Existen en la actualidad varios problemas frecuentes en los hospitales, tales como la cantidad excesiva de pacientes saturando el sistema, falta de personal capacitado, y el desperdicio de tiempo en tareas repetitivas pero indispensables para el completo funcionamiento del hospital. Lo cual representa en todos los casos un exceso de costos.

2.2 Clasificación de las camas

Las camas se clasifican según su uso, tomando en cuenta ubicación dentro del hospital y procedimiento que le antecede:

- Cama cerrada (sin paciente).
- Cama abierta (con paciente).

- Cama ocupada (con paciente encamado).
- Cama posanestésica (paciente post-operado) [10].

Se ha decidido tomar las camas cerradas como área de trabajo, puesto que es el mayor número de camas existentes en los hospitales generando así un mayor impacto.

Al analizar el problema se enfoca a la manipulabilidad de las sábanas, se realiza una interpretación de los diversos factores que interactúan en esta problemática tomando en cuenta estudios realizados para acotar de una manera apropiada y apegada a la realidad tal como es norma Oficial Mexicana NOM-197-SSA1-2000, que establece los requisitos mínimos de infraestructura y equipamiento de hospitales y consultorios de atención médica especializada [11], adicionalmente se revisa la experiencia de trabajo anterior y considerando parámetros de la primera configuración del robot que se encuentra en los laboratorios de la UNAM, llamado por los autores SOL.

Desperdicios originales	Ejemplos de desperdicios en salud (NHSIII, 2007)
Transporte	Transporte: • Personal que camina hasta el otro extremo de un área para recoger notas • Almacenes centrales para artículos de uso común, en lugar de localizarlos en almacenes de artículos de mayor frecuencia de uso
Inventario	Inventario • Exceso de existencias en almacenes que no están siendo utilizadas • Pacientes en espera de ser dados de alta • Listas de espera
Movimiento	Movimiento • Movimiento innecesario de personal buscando papeles • No contar con el equipo básico en cada examen/habitación
En espera (retardo)	En espera • Pacientes, resultados, preceptos y medicamentos • Médicos para dar alta a los pacientes
Sobreproducción	Sobreproducción • Solicitud de pruebas diagnósticas innecesarias
Durante el procesamiento	Durante el proceso de atención • Duplicación de la información • Pedir información sobre los pacientes varias veces
Defectos	Corrección • Reingreso de pacientes por fallas en la atención • Repetición de pruebas ya que la información no era correcta

Figura 1. Ejemplos de pérdidas en salud. [9]

La cama es de especial importancia para toda persona enferma; si está hospitalizada, la identifica como algo enteramente suyo. No hay duda, en efecto, que buena parte del bienestar del paciente está en la cama, sobre todo si tiene que permanecer en ella largo tiempo.

Figura 2. Cama hospitalaria Fuente. [12]

2.2 Modelado de la cama

En la figura 3 y figura 4 se puede visualizar el área de trabajo con el que interactúa el robot, la cama. El modelado CAD de ésta, figura 3, se realizó por medio de la herramienta computacional SolidWorks, con la finalidad de poder construir un modelo a escala, 20 a 3, para la interacción física entre el robot y su espacio de trabajo. En la figura 4 se observa un modelo simplificado, un paralelogramo, donde se ubicarán las trayectorias que el robot debe recorrer.

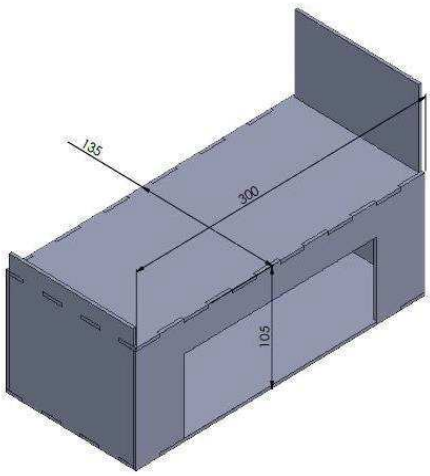


Figura 3. Cama hospitalaria sencilla Fuente autores sistema de CAD.

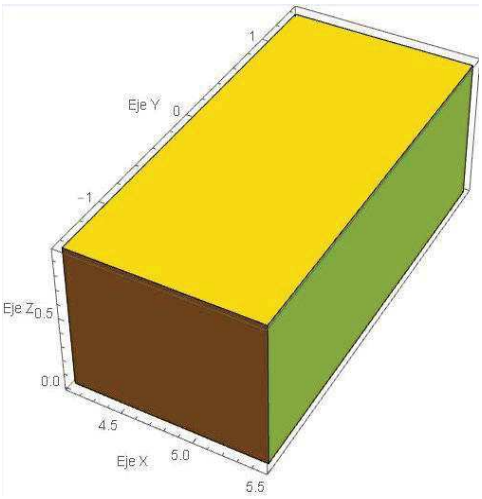


Figura 4. Representación de la cama en la simulación.

3. Modelado del robot

En este apartado se va a analizar y dar a conocer el planteamiento del modelado de la cinemática directa e inversa, las trayectorias y la comprobación de las mismas mediante un sistema de modelado.

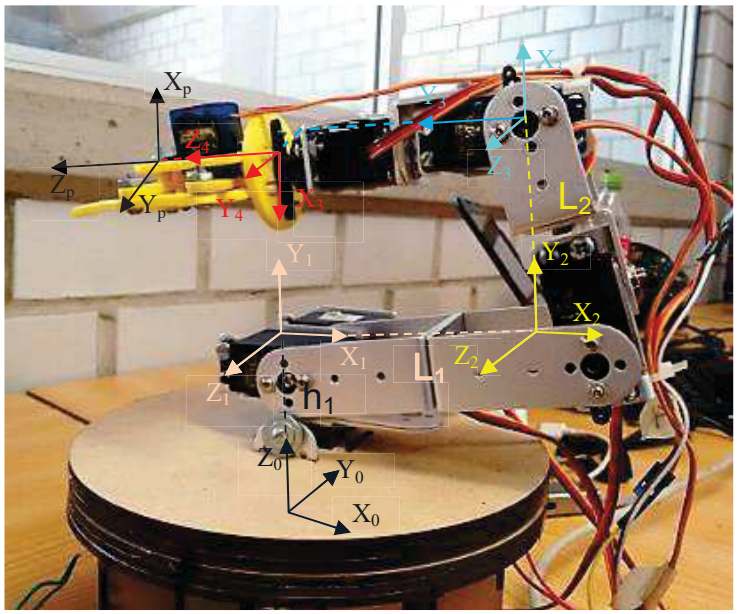


Figura 5. Configuración inicial del robot, Home.

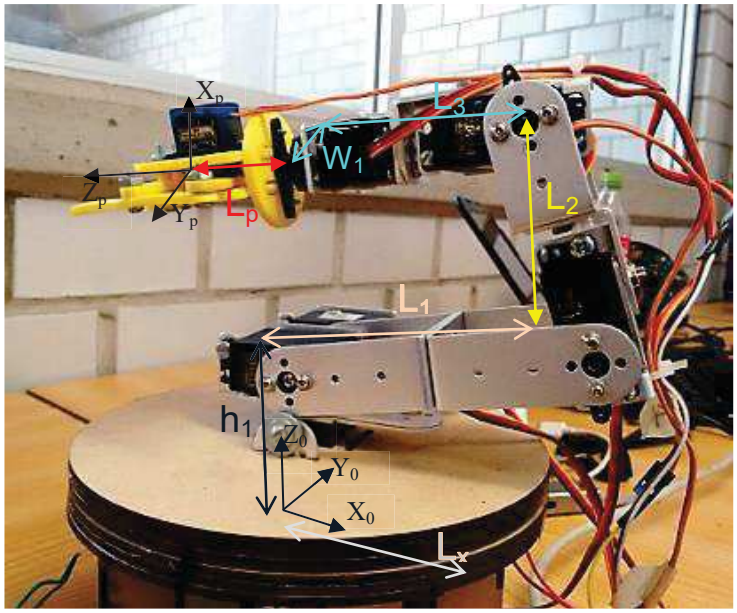


Figura 6. Parámetros del Robot.

3.1 Cinemática Directa

Con el fin de mostrar la coordinación del manipulador, primero se modela el robot. Tomando en cuenta las consideraciones tomadas de [14].

- Los eslabones del manipulador serial son rígidos.
- El número de sistemas de coordenadas será igual a n+1, donde n=GDL, Grados de libertad.
- Se ubica un sistema de coordenadas en la base fija del manipulador, éste lleva la numeración 0.
- Además, el sistema inercial de referencia será externo al robot, denominado O.
- El eje Z de cada uno de los sistemas de coordenadas coincide con el eje de giro de cada una de las articulaciones. El eje Y se establece cumpliendo la regla de la mano derecha.
- El último sistema de coordenadas se ubica en el extremo del último eslabón del manipulador.

Tabla 1 Parámetros del robot 5R utilizando D-H

i	α_i	b_i	d_i	Θ_i
0	0	0	L_x	π
1	π/2	h_1	0	Θ_1
2	0	0	L_1	Θ_2
3	0	L_2	0	Θ_3
4	-π/2	0	$\sqrt{L_3{}^2 + w_1{}^2}$	Θ_4 + π/2
5	0	0	L_p	Θ_5

El planteamiento del vector $\vec{X}$ de posición del robot es:

$$\vec{X} = \overrightarrow{a_0} + Q_0\overrightarrow{a_1} + Q_0Q_1\overrightarrow{a_2} + Q_0Q_1Q_2\overrightarrow{a_3} + Q_0Q_1Q_2\,Q_3\overrightarrow{a_4} + Q_0Q_1Q_2\,Q_3Q_4\overrightarrow{a_5} \quad (1)$$

Dónde:

$$\overrightarrow{a_i} = \begin{bmatrix} d_i \cos(\theta_i) \\ d_i \sin(\theta_i) \\ b_i \end{bmatrix} \quad (2)$$

$$Q_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \quad (3)$$

Para este caso, el vector $\overrightarrow{a_0}$ representa la configuración existente entre el sistema inercial y el sistema 0, ubicado en la base del robot; con coordenadas $\overrightarrow{a_0} = [x_e \quad y_e \quad h]^T$. El vector $\overrightarrow{a_0}$ sirve para realizar el emplazamiento del robot al asignar valores de x_e, y_e y h, dentro del espacio de trabajo [1]. Para el modelo dichos valores corresponden a $x_e =$ y $y_e = h = 0$. Para modelar al robot se sustituyen los parámetros anteriores en las transformaciones para describir la posición de HOME del manipulador. Finalmente, el vector que describe el modelo del robot es:

Vector de posición:

$$p_{P,O} = \begin{bmatrix} \frac{1}{2}\begin{pmatrix} 2\mathrm{Tr}_1 - \mathrm{L}_1\mathrm{Cos}[\theta_1 - \theta_2] - \mathrm{L}_1\mathrm{Cos}[\theta_1 + \theta_2] + \mathrm{L}_3\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + \\ \mathrm{L}_P\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + \mathrm{L}_3\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] + \mathrm{L}_P\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] \\ -2\mathrm{w}_1\mathrm{Sin}[\theta_1] - \mathrm{L}_2\mathrm{Sin}[\theta_1 - \theta_2 - \theta_3] + \mathrm{L}_2\mathrm{Sin}[\theta_1 + \theta_2 + \theta_3] \end{pmatrix} \\ \mathrm{w}_1\mathrm{Cos}[\theta_1] + \mathrm{Sin}[\theta_1](-\mathrm{L}_1\mathrm{Cos}[\theta 2] + (\mathrm{L}_3 + \mathrm{L}_P)\mathrm{Cos}[\theta_2 + \theta_3 + \theta_4] + \mathrm{L}_2\mathrm{Sin}[\theta_2 + \theta_3]) \\ \mathrm{h}_1 + \mathrm{L}_2\mathrm{Cos}[\theta_2 + \theta_3] + \mathrm{L}_1\mathrm{Sin}[\theta 2] - \mathrm{L}_3\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] - \mathrm{L}_P\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] \end{bmatrix} \quad (4)$$

Una vez obtenido el vector de posición, la forma más fácil para comprobar si éste ha sido correctamente modelado es asignar a las entradas, o los ángulos de las juntas, un valor igual a (0). De esta manera la proyección de las longitudes sobre el sistema inercial, deben corresponder con la configuración asignada desde un comienzo, la cual se denomina HOME.

$$\theta_1 = \theta_2 = \theta_3 = \theta_4 = \theta_5 = 0 \quad (5)$$

$$\mathrm{p}_{\mathrm{P,O}} = \begin{bmatrix} \mathrm{Tr}_1 - \mathrm{L}_1 + \mathrm{L}_3 + \mathrm{L}_\mathrm{P} \\ \mathrm{w}_1 \\ \mathrm{h}_1 + \mathrm{L}_2 \end{bmatrix} \quad (6)$$

Ya comprobado el modelo de la cinemática directa, a continuación, se realiza la cinemática inversa, mediante la definición del vector de configuración del robot (7).

$$\mathrm{pos}_{\xi \mathrm{P},0} = \begin{bmatrix} \frac{1}{2}\begin{pmatrix} 2\mathrm{Tr}_1 - \mathrm{L}_1\mathrm{Cos}[\theta_1 - \theta_2] - \mathrm{L}_1\mathrm{Cos}[\theta_1 + \theta_2] + \mathrm{L}_3\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + \\ \mathrm{L}_P\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + \mathrm{L}_3\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] + \mathrm{L}_P\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] \\ -2\mathrm{w}_1\mathrm{Sin}[\theta_1] - \mathrm{L}_2\mathrm{Sin}[\theta_1 - \theta_2 - \theta_3] + \mathrm{L}_2\mathrm{Sin}[\theta_1 + \theta_2 + \theta_3] \end{pmatrix} \\ \mathrm{w}_1\mathrm{Cos}[\theta_1] + \mathrm{Sin}[\theta_1](-\mathrm{L}_1\mathrm{Cos}[\theta 2] + (\mathrm{L}_3 + \mathrm{L}_P)\mathrm{Cos}[\theta_2 + \theta_3 + \theta_4] + \mathrm{L}_2\mathrm{Sin}[\theta_2 + \theta_3]) \\ \mathrm{h}_1 + \mathrm{L}_2\mathrm{Cos}[\theta_2 + \theta_3] + \mathrm{L}_1\mathrm{Sin}[\theta 2] - \mathrm{L}_3\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] - \mathrm{L}_P\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] \\ \mathrm{Cos}[\theta_1]\mathrm{Cos}[\theta_5]\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] + \mathrm{Sin}[\theta_1]\mathrm{Sin}[\theta_5] \\ -\mathrm{Cos}[\theta_1]\mathrm{Cos}[\theta_5] - \mathrm{Sin}[\theta_1]\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4]\mathrm{Sin}[\theta_5] \\ -\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] \end{bmatrix} \quad (7)$$

El vector (7) está expresado en términos de las variables de configuración de la tarea, este mismo vector puede ser expresado en términos de las variables que describen la tarea con respecto al sistema inercial (8).

$$\mathrm{pos}_{\xi P,O} = \begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \\ \mathrm{Cos}[\theta_{yp}] \\ \mathrm{Cos}[\theta_{zp}] \end{bmatrix} \quad (8)$$

Una vez definidos los dos vectores, se igualan 0 y se obtiene el espacio nulo de donde se obtendrán las soluciones de posición para cada junta.

$$C_q = \mathrm{pos}_{\xi P,O} - \mathrm{pos}_{\xi P,0} = 0 \quad (9)$$

$$C_q = \begin{bmatrix} \begin{pmatrix} X_p + \frac{1}{2}(-2\mathrm{Tr}_1 + L_1\mathrm{Cos}[\theta_1 - \theta_2] + L_1\mathrm{Cos}[\theta_1 + \theta_2] - L_3\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] \\ -L_P\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + L_3\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] - L_P\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] \\ +2w_1\mathrm{Sin}[\theta_1] + L_2\mathrm{Sin}[\theta_1 - \theta_2 - \theta_3] - L_2\mathrm{Sin}[\theta_1 + \theta_2 + \theta_3] \end{pmatrix} \\ Y_p - w_1\mathrm{Cos}[\theta_1] + \mathrm{Sin}[\theta_1](-L_1\mathrm{Cos}[\theta 2] + (L_3 + L_P)\mathrm{Cos}[\theta_2 + \theta_3 + \theta_4] + L_2\mathrm{Sin}[\theta_2 + \theta_3]) \\ Z_p - h_1 - L_2\mathrm{Cos}[\theta_2 + \theta_3] - L_1\mathrm{Sin}[\theta 2] + L_3\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] + L_P\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] \\ 1 - \mathrm{Cos}[\theta_1]\mathrm{Cos}[\theta_5]\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] - \mathrm{Sin}[\theta_1]\mathrm{Sin}[\theta_5] \\ \mathrm{Cos}[\theta_1]\mathrm{Cos}[\theta_5] + \mathrm{Cos}[\theta_{yp}] + \mathrm{Sin}[\theta 1]\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4]\mathrm{Sin}[\theta_5] \\ \mathrm{Cos}[\theta_{zp}] + \mathrm{Sin}\theta_2 + \theta_3 + \theta_4] \end{bmatrix} \quad (10)$$

Debido a que el sistema no converge a una única solución por la redundancia del sistema, fue necesario asumir dos condiciones adicionales. La primera de ellas fue que el 5 grado de libertad iba ser igual a 0, es decir no rotará , ya que el efector final no necesita girar cuando está recogiendo la sabana; Por otro lado, se asumió un valor para el 4 grado de libertad, θ_4, el cual fue determinado experimentalmente a través de la manipulación directa entre el robot y la cama modelo, estas determinaciones, se pueden asumir, debido a que solo es necesario tres variables para describir un cuerpo en el espacio.

$$\theta_4 = -\frac{\pi}{6}; \qquad \theta_5 = 0 \quad (11)$$

De esta manera, el sistema que resolverá las posiciones de las juntas para cada articulación estará determinada solamente por las posiciones de los puntos que conforman la trayectoria que debe seguir el robot.

$$\begin{gathered} \mathbf{X_p} = \frac{1}{2}(2\mathrm{Tr}_1 - L_1\mathrm{Cos}[\theta_1 - \theta_2] - L_1\mathrm{Cos}[\theta_1 + \theta_2] + L_3\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] \\ +L_P\mathrm{Cos}[\theta_1 - \theta_2 - \theta_3 - \theta_4] + L_3\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] + L_P\mathrm{Cos}[\theta_1 + \theta_2 + \theta_3 + \theta_4] \\ -2w_1\mathrm{Sin}[\theta_1] - L_2\mathrm{Sin}[\theta_1 - \theta_2 - \theta_3] + L_2\mathrm{Sin}[\theta_1 + \theta_2 + \theta_3] \\ \mathbf{Y_p} = w_1\mathrm{Cos}[\theta_1] + \mathrm{Sin}[\theta_1](-L_1\mathrm{Cos}[\theta 2] + (L_3 + L_P)\mathrm{Cos}[\theta_2 + \theta_3 + \theta_4] + L_2\mathrm{Sin}[\theta_2 + \theta_3]) \\ \mathbf{Z_p} = h_1 + L_2\mathrm{Cos}[\theta_2 + \theta_3] + L_1\mathrm{Sin}[\theta 2] - L_3\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] - L_P\mathrm{Sin}[\theta_2 + \theta_3 + \theta_4] \end{gathered} \quad (12)$$

3.2 Determinación de las trayectorias

Las funciones utilizadas para describir la trayectoria de un robot deben ser continuas y derivables para todos sus puntos. Algunos métodos de interpolación por polinomios crean polinomios cuyo grado depende de la cantidad de puntos utilizados para definir la curva. Las interpolaciones polinómicas más comunes son las propuestas por Lagrange, Neville y Newton. En cada una de las interpolaciones mencionadas se genera un polinomio de grado igual n-1, donde n es la cantidad de puntos usados para crear la función. [16]

Las trayectorias partirán de la esquina cercana donde se encuentra ubicado el robot SOL, y de vuelta al centro, esto se repite con cada una de las esquinas, plegando de esta manera la sabana. Las trayectorias propuestas para el desarrollo de la tarea parten del planteamiento de la ecuación de una semi-elipse se genera mediante funciones paramétricas, donde θ será la variable que irá cambiando los valores de x y y, la cual tiene un rango de 0 a π puesto que sólo es necesario media elipse. Además, se encuentra contenida en el plano yz y está desfasa para iniciar en el origen arbitrario y posteriormente, mediante transformaciones se rotará de forma perpendicular en las cuatro esquinas de la cama.

$$y = Acos(\theta) + d \quad (13)$$

$$z = Bsin(\theta) \quad (14)$$

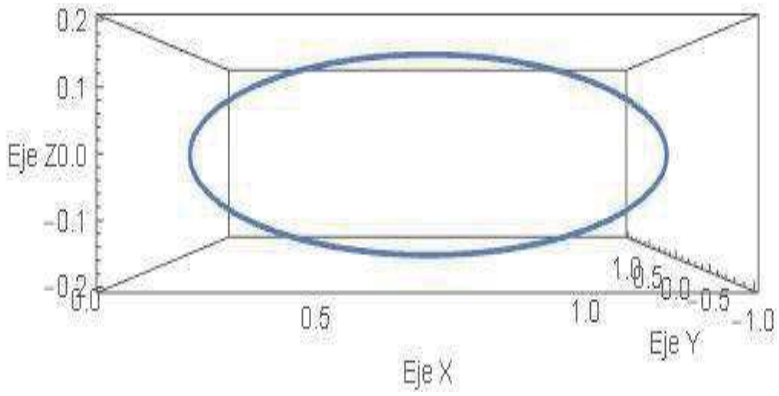


Figura 7. Elipse de referencia Curva para las trayectorias.

Para obtener los parámetros A, B y d se establecieron tres puntos 3, el inicial (0,0) el final (1.44,0) de la trayectoria y uno intermedio (0.72,0.2). El primero es el origen del sistema, el segundo es la mitad de la longitud del lado más largo de la cama, y el último será la mitad de la elipse junto con una altura que se estimó como la necesaria para poder plegar la sabana. Así las constantes que se obtuvieron son: A = -0.72; B=0.2; d=0.72. Una vez obtenida la función se procede a graficar la curva para comprobar que ésta se obtuvo correctamente, como se observa en la figura 7.

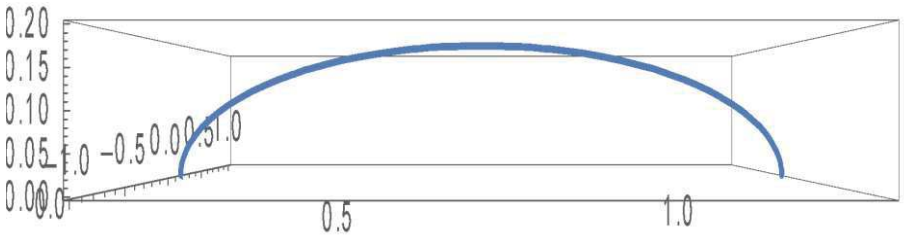


Figura 8. Gráfica de la Elipse generada para las trayectorias.

A partir de allí, la curva se debe transformar para poder adaptarse al espacio de trabajo, la cama del hospital. Para ello se utilizarán las mismas transformaciones homogéneas con las que se modeló el robot de la siguiente manera: una transformación de traslación en x (X_C), una transformación de traslación en z, Z_C, y una transformación de rotación en $z\,\theta_r$. Las dos primeras serán compartidas por todas las curvas, mientras que la última varia de acuerdo a la curva, dónde las constantes serán las siguientes:

$$X_C = 4.8\,;\; Z_C = 0.96; \quad (15)$$

$$\theta_{r1} = \frac{2\pi}{15};\ \theta_{r2} = -\frac{2\pi}{15};\ \theta_{r3} = -\frac{13\pi}{15};\theta_{r4} = \frac{13\pi}{15} \quad (16)$$

A continuación, se observa, en la figura 8, la trayectoria 1 una vez aplicadas las transformaciones a la curva de la elipse.

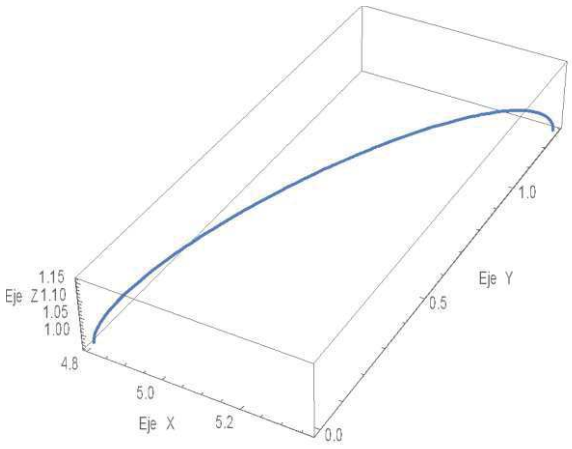


Figura 9. Elipse en el espacio.

La manera en que el robot realice la tarea depende de la resolución de los actuadores y la capacidad de procesamiento del robot. De igual forma, la cantidad de puntos depende del valor de los incrementos al evaluar la función [16], es decir que para este caso se usaron 20 puntos, ya que con estos son suficientes, y por otro lado no se requiere una gran capacidad computacional. Cuando ya se tienen los puntos de todos los segmentos de curva es necesario pegarlos en un arreglo y con ello se genera la curva completa que describe las trayectorias. Con todo lo anterior se puede proceder a la simulación del robot y las trayectorias.

3.2 Simulación

Una vez comprobadas que las trayectorias se han creado, para poder continuar con el proceso de modelado y simulación se debe confirmar que estas se encuentren dentro del espacio de la cama y van del centro a cada esquina. Lo cual se puede comprobar en la figura 9. De allí se procede con la generación de los eslabones del robot, figura 10, y la comprobación de los movimientos de cada articulación. Una vez comprobados los movimientos, se junta al robot y la cama para verificar nuevamente que ambos se encuentran en las posiciones de configuración inicial.

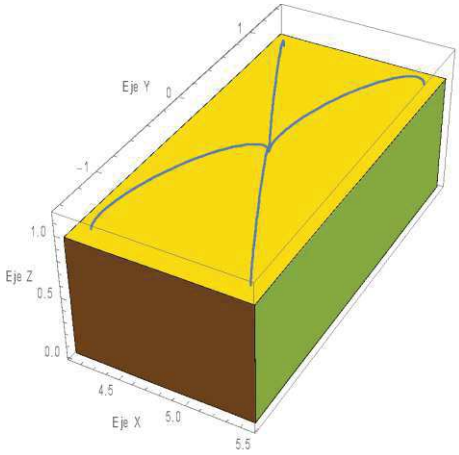


Figura 10. Cama con trayectorias generadas.

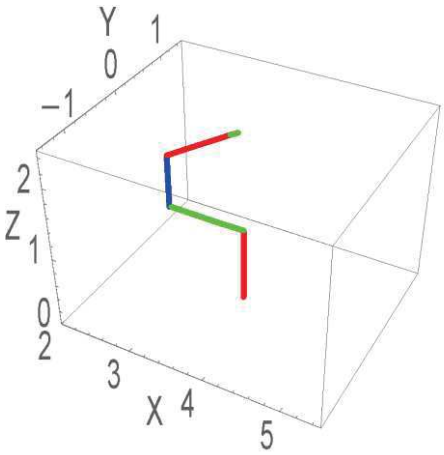


Figura 11. Comprobación del modelado del robot.

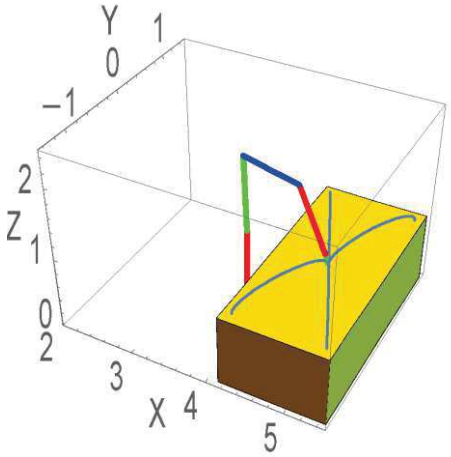


Figura 12. Posicionamiento de Robot y cama.

Por último, a través de la simulación computacional se encuentran las configuraciones a través del cálculo de los ángulos de las juntas rotacionales. Con la información anterior se determinó que el robot no presentó ninguna singularidad al recorrer las trayectorias.

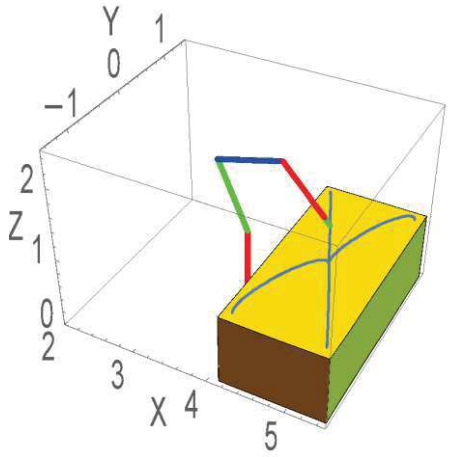


Figura 13. Simulación de la primera trayectoria.

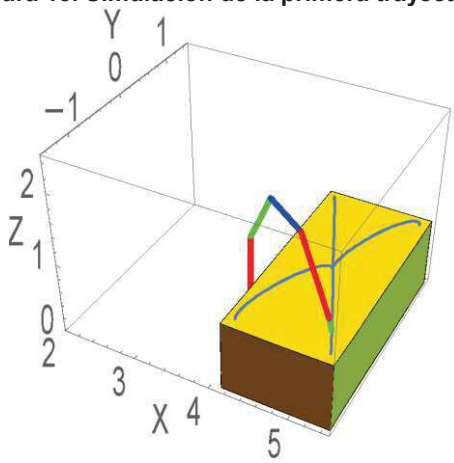


Figura 14. Simulación de la tercera trayectoria.

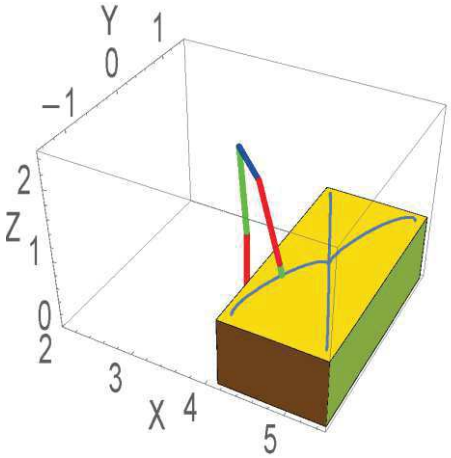


Figura 15. Simulación de la cuarta trayectoria.

4. Conclusiones

Mediante la investigación planteada se encontró que el robot propuesto para la tarea de plegado de sábanas desarrolla su función sin presentar ninguna singularidad al ejecutar las trayectorias. De igual forma, las trayectorias propuestas se ajustaron a los requisitos que inicialmente se habían planteado, el robot mantiene su efector final con un ángulo aproximado de 45°, gracias a la restricción cinemática en la 4 articulación.

La simulación permite probar diferentes trayectorias para manipular las sabanas, estas dependerán del diseño de la herramienta que utilicen y de las condiciones sanitarias del paciente. Por lo anterior, es necesario plantear las trayectorias con base a las características del tipo de cama hospitalaria y del tipo de padecimiento del paciente.

Se propuso una clasificación para los robots en ambientes hospitalarios, en vista de que en los estudios revisados no se encontraron metodologías que involucran la clasificación específica de los robots según su designación de actividad para ambientes hospitalarios. La relación con la clasificación se limita a la disposición de los elementos con que está construido y no a su actividad final.

Para futuros trabajos es necesario comprobar si la longitud de los eslabones es la adecuada para el desarrollo de la tarea, es por ello que se plantea tener en cuenta este parámetro. Igualmente se tiene la necesidad de explorar el diseño de un efector final especial para la tarea de tomar la sábana. Finalmente, ya que la investigación pretende establecer el destendido y tendido de la cama, es necesario continuar con el desarrollo del modelo dinámico y proponer los nuevos criterios para cumplir con la tarea mencionada, y que, una vez obtenido el modelo, se pueda proceder a realizar las pruebas físicas con el prototipo para poder nuevamente validar la información obtenida.

5. Agradecimientos.

Se agradece la colaboración del M.I. Erik Peña Medina y Dr. Víctor Javier González Villela, a los compañeros de la materia de robótica de semestres anteriores que trabajaron en la construcción del robot SOL, y al departamento de Ingeniería Mecatrónica de la Facultad de ingeniería de la UNAM y a la DGAPA por el apoyo al proyecto Papiit IN118117, por la realización del presente proyecto.

Referencias

[1] Peña E., Víctor J. González-Villela, Mejía H. Cortés J, y Díaz Hernández. (2014), DETERMINACIÓN DEL LUGAR GEOMÉTRICO DE UNA TRAYECTORIA PARA EL DISEÑO DE UN ROBOT SERIAL RRR, memorias del xx congreso internacional anual de la Somim 24 al 26 de septiembre, Juriquilla, México, 2014.

[2] Latombe, J. C. (1990). Robot Motion Planning (the Kluwer international series in engineering and computer science) K.luwer Academic Publishe, Estados Unidos de Norte America, 8 Ed, 2004.

[3] Armstrong, B., Khatib, O., & Burdick, J. (1986, April). The explicit dynamic model and inertial parameters of the PUMA 560 arm. In Robotics and Automation. Proceedings. 1986 IEEE International Conference on (Vol. 3, pp. 510-518). IEEE

[4] Galeano D. Robótica Médica, Universidad Católica Nuestra Señora de la Asunción, Paraguay, Ed *22*, 2012.

[5] Klimstra, M. D., Thomas, E., Stoloff, R. H., Ferris, D. P., & Zehr, E. P, Neuromechanical considerations for incorporating rhythmic arm movement in the rehabilitation of walking. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *19*(2), 026102. 2009.

[6] McCarthy, J. (1964). Computer Control of a Machine for exploring Mars, STANFORD ARTIFICIAL INTELLIGENCE PROJECT. *COMPUTER*, Vol. 14, 1964

[7] Moreno, M., Valenzuela, C., Quiroz, C. D., Pachecho, C., & Rojano-Rodríguez, M. *"Cirugía robótica*". Gaceta Médica de México, 150(s3), 293-297. 2014.

[8] Sánchez, F. M., Jimenez, P., Millán, F., Salvador, J., Monllau, V., Palou, J., & Villavicencio, H. *"Historia de la robótica: de Arquitas de Tarento al Robot da Vinci (Parte II)"*. Actas Urológicas Españolas, 31(3), 185-196. 2007.

[9] Niechwiadowicz, K., & Khan, Z. *"Robot based logistics system for hospitals-survey"*. IDT Workshop on interesting results in computer science and engineering. 2008.

[10] Potter, P. A., Perry, A. G., Stockert, P. A., & Hall, A. M. "*Fundamentos de enfermería".* Mosby. España. 4 Ed. 1996.

[11] NORMA Oficial Mexicana NOM-197-SSA1-2000, *"Que establece los requisitos mínimos de infraestructura y equipamiento de hospitales y consultorios de atención médica especializada."* Diario Oficial de la Federación. México. 14 de octubre de 2001.

[12] Franco, S. *"Técnica de tendidos de cama".* Recuperado de: enfermeriahoy.com/técnica-de-tendidos-de-cama/

[13] Navarro, N. P., Robles, S. C., & Paulsen, K. F. *"Diseño y fabricación de un brazo robot de cinco grados de libertad articulado verticalmente: modelado cinemático y dinámico"*. Octavo congreso Iberoamericano de Ingeniería Mecánica. Federación Iberoamericana de Ingeniería Mecánica. Cusco. 2007.

Ingeniería y Automatización, Capítulo 5, pp. 59 - 76.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Análisis y Diseño de Control para la Corrección de Ángulo Pitch en Pala de un Aerogenerador de Baja Potencia

Sánchez Luna David, Aguilar Anastacio José Carlos y Muñoz Flores Luis David

Tecnológico de Estudios Superiores de Ecatepec
Posgrado de Eficiencia Energética y Energías Renovables
Av. Tecnológico S/N, Valle de Anáhuac, Ecatepec de Morelos, Estado de México. C.P.55210.
Tel.50002330. Mail: dslunaok@hotmail.com

Resumen

En años recientes el incremento de la contaminación, los precios de los combustibles, la continua expansión de la mancha urbana y como consecuencia la gran demanda de energía eléctrica en el mundo, son motivos que despiertan el interés en la búsqueda de nuevas alternativas para el aprovechamiento de las fuentes renovables, de manera eficiente y sin daño alguno para los sistemas de conversión de energía a partir del viento a energía eléctrica. En el presente trabajo se reporta el análisis y diseño de un controlador con amplificadores operacionales y un motor de CD, para el ajuste del ángulo pitch (ángulo de paso), de una pala en aerogeneradores de baja potencia. Actualmente los aerogeneradores de baja potencia son aquellos que tienen un área de barrido de 78.54m^2 como máximo, con una longitud de pala de 1 a 5m aproximadamente. La mayoría de los aerogeneradores de baja potencia no cuentan con control de ángulo de pitch en sus palas, esto ocasiona que las rachas de viento superiores a las permitidas en estos sistemas ocasionan fracturas o fisuras en las palas, además de sobrecarga, por lo que su eficiencia y tiempo de vida tienen a limitarse. Se plantea desarrollar la función de transferencia de un modelo simple de un servomotor para la variación del ángulo pitch a partir de la velocidad del viento. Los resultados del análisis mostrarán el ángulo adecuado para una mayor eficiencia y el ángulo de protección para la pala en velocidades de viento extremos.

Palabras clave: Eólico, Controlador, Servomotor, Función de transferencia, Velocidad de viento, Angulo Pitch.

Ingeniería y Automatización, Capítulo 5, pp. 59 - 76.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

1. Introducción

La energía eólica es la energía cinética de una masa de aire en movimiento que se origina por la insolación, la diferencia de temperaturas y presión en ciertas zonas del planeta, principalmente en el ecuador la radiación directa ocasiona un incremento de temperatura y disminución de la presión, mientras que en los polos la radiación solar es inferior ocasionando una temperatura menor y mayor presión, esta diferencia de presiones origina la producción de corrientes de aire en todo el mundo. [1] Durante años se ha aprovechado este tipo de energía en molinos de viento para la molienda de granos y el bombeo de agua principalmente, hoy en día una de las mayores aplicaciones del aprovechamiento de la energía cinética del viento es la producción de energía eléctrica. En la última década, la capacidad de potencia instalada de las grandes turbinas eólicas tuvo una tasa de crecimiento anual de aproximadamente 30%, mientras que las pequeñas turbinas eólicas fue solo del 9% [2].

En la actualidad las tecnologías aplicadas para la generación eléctrica han revolucionado a tal grado que se encuentran disponibles en una gran variedad de tipos y tamaños desde los 10 kW hasta los 6 MW aproximadamente, la aplicación de los generadores estará en función de la capacidad de generación y el tipo de carga. En general existen generadores eólicos de dos tipos de eje, *horizontal* y *vertical*, donde el eje es parte principal del generador en él se encuentran montadas las palas que tienen interacción con la velocidad del viento y al mismo tiempo transmite el movimiento mecánico al sistema de generación donde puede montarse una, dos o hasta tres palas según la velocidad del viento. Los generadores con mayor demanda de instalación son de eje tipo horizontal.

La instalación de aerogeneradores de baja y mediana potencia se encuentra en un continuo crecimiento en nuestro país y en el mundo, al mismo tiempo se tiene un incremento en los daños que sufren las palas como fisuras, perdida de verticalidad o rompimiento de la misma, producto de las altas velocidades del viento que se pueden alcanzar en algunas estaciones del año y la colocación en el rotor a un ángulo permanente con respecto a su eje de simetría a lo largo de la pala, lo que tiende a disminuir su tiempo de vida y el rendimiento del generador.

En este artículo se propone el análisis y diseño de control proporcional de un servomotor para realizar el movimiento angular en un rango de 0 a 45° de una pala para generadores de baja y mediana potencia, implementando un motor de CD como servomotor, amplificadores operacionales como controlador y un potenciómetro como sensor de posición, la alimentación del controlador se realizará a partir de un panel solar de 15 watts como fuente de alimentación y la conexión de la fuente a los servomotores de las palas se llevará a cabo a través de un sistema de anillos rozantes, donde se tomará como variable de control la velocidad del viento.

La metodología se realiza con el análisis del ángulo pitch y el desarrollo del controlador a partir de amplificadores operacionales en configuración proporcional, se genera un modelo de la pala con el servomotor incluido para una posible construcción, se efectúa la comprobación del modelo del controlador en Matlab.

Los resultados esperados tienen como objetivo mantener el ángulo adecuado para la protección de la pala cuando la velocidad del viento exceda lo estimado en ciertos generadores y de esta manera incrementar el rendimiento del generador y la vida útil.

Ingeniería y Automatización, Capítulo 5, pp. 59 - 76.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2. Antecedentes

Existen evidencias que los primeros molinos de viento eran aerogeneradores de eje vertical. En la Edad Media, numerosos molinos de viento horizontales se instalaron en Europa: Holanda, Grecia y España. Los primeros aerogeneradores capaces de producir electricidad fueron desarrollados en Dinamarca por Paul La Cour en 1891. Los hermanos Jacobs en América en 1922, produjeron turbinas eólicas de carga de baterías. Después de la segunda guerra mundial, el primer aerogenerador de corriente alterna fue diseñado por Johannes Juul en Dinamarca, el cual poseía dos láminas delgadas de fibra de vidrio montadas al final de la torre con un rotor de posición variable, un generador asíncrono e incluía además de un freno aerodinámico.[3]

En los años 80s, la construcción de parques eólicos tienen una capacidad inicial de 1500 MW con aerogeneradores de capacidad aproximada de 50 kW, a partir de los años 90s, a nivel mundial se presenta un mayor apoyo financiero para el desarrollo de parques eólicos y como consecuencia la modernización de nuevas tecnologías de conversión.

Las pequeñas turbinas eólicas generan menor cantidad de energía; la potencia tiene una disminución en relación con el cuadrado del radio de la cuchilla. En la definición de la Comisión Electrotécnica Internacional 61400-2, una pequeña turbina eólica tiene un área de barrido máximo de 200 m^2. El área de barrido es aproximadamente igual a ocho metros de radio de la cuchilla. Según la Asociación Canadiense de Energía Eólica (CANWEA), una turbina mediana genera entre uno y 300 kW, la generación menor a 100 kW son considerados sistemas de baja potencia. [4]

Actualmente las políticas medioambientales en todo el mundo pretenden reducir los gases de efecto invernadero y el interés por utilizar la energía eólica va en aumento. Según el “GLOBAL WIND REPORT 2015”, reporta un incremento considerable en capacidad instalada de energía eólica representativa en todo el mundo.

México tiene el compromiso de limitar la generación eléctrica por fuentes fósiles al 65% (de un 80% actual) para el año 2024. Para el año 2020 el 12% de la energía eléctrica mundial será abastecida por energía eólica. En el 2010, los 84 GW creados por energía eólica, han evitado 126 millones de toneladas de emisiones de CO_2, equivalente al 30% de los autos de los Estados Unidos de América (60 millones). [5]

2.1 Velocidad y energía del Viento.

Una turbina eólica captura la energía cinética del viento en un rotor, el cual esta mecánicamente acoplado a un generador eléctrico. [6]

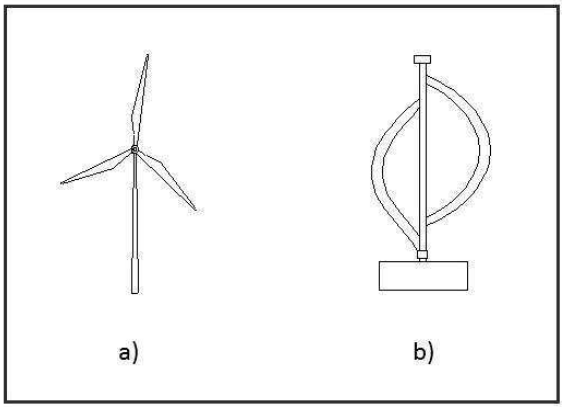


Figura 1. Tipos de generadores eólicos: a) Eje horizontal, b) Eje vertical

Los generadores eólicos son instalados a cierta altura para garantizar una mayor captura de energía cinética del viento. Desde el punto de vista operacional, existen básicamente dos tipos de aerogeneradores: de eje *horizontal* y *vertical*, como se puede observar en la figura 1, la tendencia de instalación de aerogeneradores se ha reflejado para tipo horizontal.

2.2 Relaciones de velocidad y potencia

La energía cinética (**Ec**) de la masa (**m**) de aire que se mueve a una velocidad (**v**), se presenta en la ecuación (1)

$$Ec = \frac{1}{2} m v^2 \quad \text{.....(1)}$$

La potencia (**P**) del movimiento del aire es el caudal de energía cinética sobre segundo, ecuación (2)

$$P = \frac{1}{2} \frac{(\rho A v) v^2}{s} \quad \text{.....2)}$$

La potencia mecánica que transmite el viento al generador se presenta en la ecuación (3), expresada en watts por metro cuadrado de barrido de las palas giratorias

$$P = \frac{1}{2} \rho A v^3 \quad \text{.....(3)}$$

Donde:
P = Potencia mecánica en el aire en movimiento (Watts)
ρ = Densidad del aire (kg/m^3)
A = Área barrida por las palas del rotor (m^2)
V = Velocidad del aire (m/s)

2.3 Perfil de Velocidad del Viento.

La porción de la atmosfera de los perfiles de la velocidad del viento, es la parte inferior de la capa límite (Atmospheric Boundary Layer, ABL), la cual es denominada capa atmosférica superficial (Atmospheric Surface Layer, ASL) de altura variable entre 60 y 100 metros sobre la superficie del suelo, agua o hielo. En esta capa superficial de la atmosfera existe alta interacción de flujo de calor, humedad y de momento entre la atmosfera y la superficie los cuales son estudiados dentro del campo específico de la micrometeorología y microclimatología. [7]

En la capa atmosférica superficial se conoce que la velocidad del viento tiende a incrementarse a mayor altura y que la superficie terrestre ejerce una acción de fricción o de retardo sobre la velocidad. La representación de este comportamiento de velocidad del viento con la altura se conoce como perfil vertical del viento. En los perfiles del viento las superficies más irregulares o rugosas tendrán mayor influencia sobre la velocidad del viento que las superficies lisas.

La velocidad del viento se define por la expresión (4)

$$V_Z = V_{ref}\left(\frac{Z}{Z_{ref}}\right)^{\alpha} \quad \text{.....(4)}$$

Donde:

Vz = Velocidad del Viento a estimar a una altura Z sobre el nivel del suelo.
Vref = Velocidad de referencia, velocidad del viento observada a una altura prestablecida.
Zre = Altura de referencia (generalmente 10m).
α= exponente de rugosidad (wind shear exponent). Variación entre (0 – 0.40) según el tipo de rugosidad del terreno: agua 0.13; grama 0.14 a 0.16; cultivos y arbustos 0.20; bosques 0.25 y zonas urbanas 0.40.

2.4 Actuador.

Dispositivo que transforma algún tipo de energía a un desplazamiento mecánico rotacional o lineal, este es el encargado de transmitir el movimiento en el posicionamiento de la "pala" del aerogenerador al ángulo de choque con el viento adecuado.

2.5 Servomotor

Actuador especial que responde a una señal de entrada para ajustar su posición o velocidad, es decir obedece a quien envía la señal y efectúa la orden, es un sistema de lazo cerrado que monitorea continuamente la posición de la flecha del motor, por medio de un potenciómetro conectado en la misma flecha a través de un conjunto de engranes, la señal producida en el potenciómetro producto del desplazamiento angular, es enviada al comparador donde se compara con el voltaje de referencia de un circuito externo como señal de control. El resultado de la comparación puede tomar dos posibles valores positivo o negativo.

2.6 Función de transferencia

En la teoría de control la función de transferencia se utiliza para el análisis del comportamiento de sistemas en diferentes procesos de nuestro entorno de forma dinámica o estacionaria. La operación dinámica depende del tiempo, a medida que el tiempo varía cambian las condiciones que describen el comportamiento de las variables de manera adecuada se plantean ecuaciones que representan el proceso funcional.

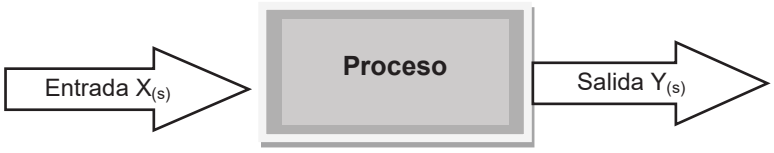


Figura 2. Función de Transferencia.

Los sistemas de control se ocupan del control de variables específicas, las variables controladas deben relacionarse con las variables de control, esta relación está representada por la función de transferencia de las variables de entrada y salida. [8]

3. Metodología.

El diseño del controlador tiene por objetivo principal ajustar el ángulo pitch de la pala como tarea específica en función de la velocidad del viento, los requisitos que se deben de cumplir sobre el sistema de control se presentan como especificaciones de comportamiento. En la figura 3, se observa la metodología a desarrollar en una secuencia ordenada.

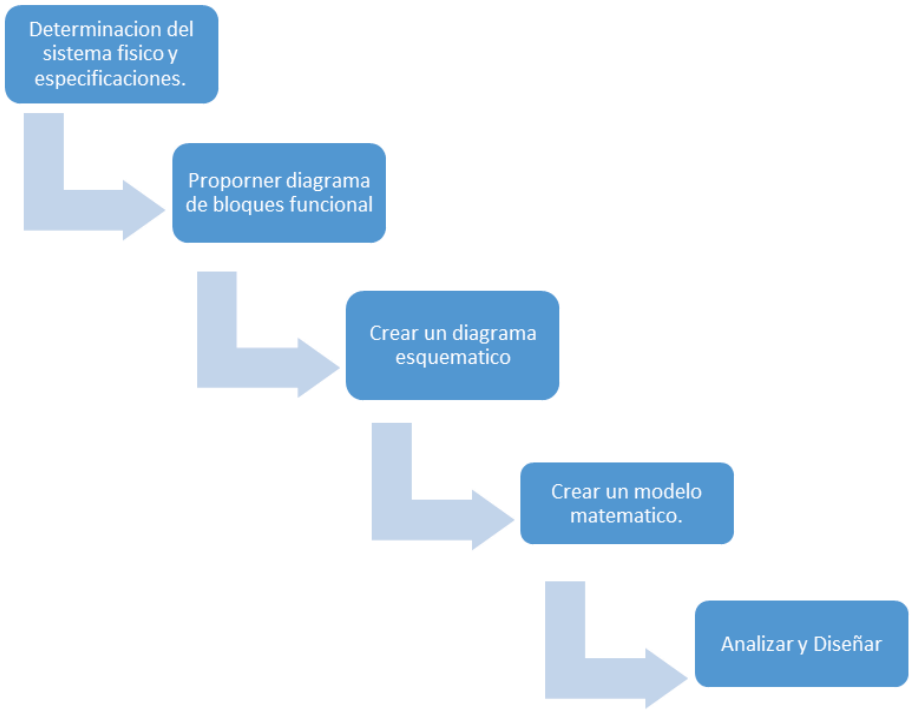


Figura 3. Proceso de diseño de un sistema de control.

Normalmente las especificaciones de un sistema de control se deben de conocer antes del proceso de diseño, en este caso las especificaciones estarán determinadas por la relación existente entre la velocidad del viento (máxima o mínima), el ángulo de giro de las palas (ángulo pitch) de un aerogenerador de eje horizontal.

Para el diseño del control, se debe de obtener un modelo matemático con ciertos parámetros importantes que se ajustan en la verificación del comportamiento del sistema, una vez obtenido el modelo matemático satisfactorio se diseña un prototipo para la verificación de datos y estabilidad absoluta del sistema en lazo cerrado, el diseño del prototipo se podrá ajustar hasta que el sistema cumpla las especificaciones adecuadas, una vez realizado los ajustes se analiza cada prueba se incorpora resultados de este análisis en la prueba siguiente. Por último se debe conseguir que el sistema final cumpla las especificaciones de comportamiento, fiable y económico.

Ingeniería y Automatización, Capítulo 5, pp. 59 - 76.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

4. Desarrollo.

Para el ajuste del ángulo pitch se implementa el diseño del servomotor, la colocación se realiza en la raíz de la pala para realizar el giró longitudinalmente en ambas direcciones. Actualmente los aerogeneradores de baja potencia no cuentan con este sistema de control por lo que las rachas de viento superiores ocasionan grandes daños por fracturas mecánicas o por sobrecarga en el sistema de control. [9]

4.1 Especificaciones del sistema Físico.

De acuerdo a los parámetros de control que se desean regular para un aerogenerador de baja potencia, se realizó el cálculo del perfil de velocidad de viento para establecer los límites de velocidad del caso de estudio, para calcular el perfil vertical del viento se tomó los siguientes datos del observatorio meteorológico central de la ciudad de México, en la Figura 4, se observa la dirección y velocidad del viento en Km/h, para el día miercoles 16 de mayo de 2018, muestras en cada 15 min.

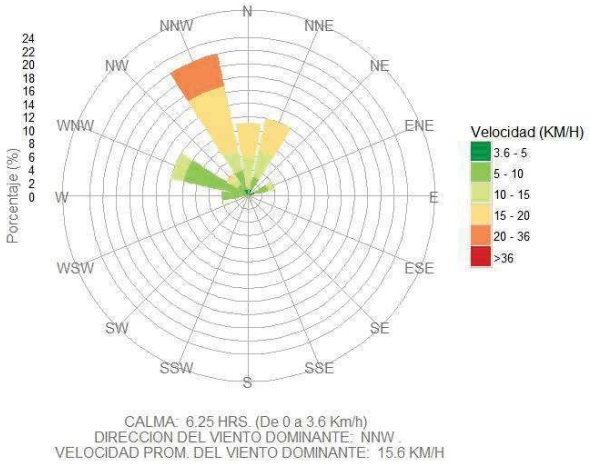


Figura 4. Rosa de los Vientos.

Bajo estas condiciones se determinaron los parámetros de velocidad establecidos como:

Velocidad de viento Máxima	40.5	km/hora	**11.25**	**m/s**
Velocidad de viento Promedio	15.6	km/hora	**4.333**	**m/s**

Dirección del Viento Máximo: NNW
Dirección del Viento Dominante: NNW

La velocidad promedio nos permite realizar el cálculo del perfil del viento (Ecuación 4), se estima colocar a una altura de 5 m, bajo esta condición el ángulo pitch debe proporcionar mayor potencia posible, el movimiento angular en el servomotor ajustará un ángulo de 0° en la pala cuando se alcance la máxima velocidad del viento. En la tabla 1, se muestra las velocidades promedio que puede alcanzar el viento en la ciudad de México en base a la altura.

Velocidad Promedio de Referencia: 4.33 m/s	
Altura	V_z
2	4.33
3	5.09
4	5.71
5	**6.25**
6	6.72
7	7.15
8	7.54
9	7.90
10	8.24
15	9.69

Tabla 1. Calculo de la velocidad del viento en base a la altura con respecto al suelo.

En la figura 5, se observa la gráfica de velocidad del viento en función de la altura, utilizada en la colocación de aerogeneradores a cierta altura de acuerdo a la potencia de generación.

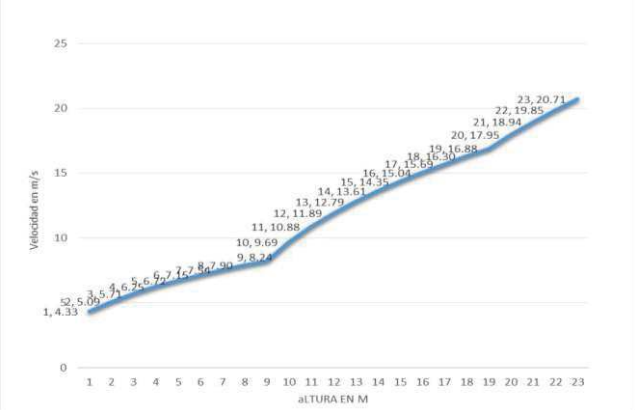


Figura 5. Perfil vertical del viento (Velocidad vs Altura)

La velocidad promedio calculada según la altura se encuentra en el rango de 2.33m/s a 6.25m/s, por lo que la variación de velocidad en el comparador proporcionará un cierto voltaje al ser comparado, el resultado generará en el servomotor un movimiento angular a un valor determinado.

En el caso específico de las dimensiones del aerogenerador de baja potencia, el diámetro de las palas en el rotor se encuentran en un intervalo de 1 a 5m, la potencia mecánica (Ecuación 3) para los generadores de baja potencia se muestra en la tabla 2.

Diámetro del Rotor (m)	Potencia Mecánica (Kw)	Densidad del aire (kg/m^3)	Área (m^2)	Velocidad (m/s)	Coeficiente aerodinámico de la pala (0.4 - 0.5)
1	0.20	1.22	0.78	6.25	0.45
1.3	**0.35**	1.25	1.40	6.25	0.45
2	0.81	1.22	3.14	6.25	0.45
3	1.82	1.22	7.06	6.25	0.45
4	3.24	1.22	12.56	6.25	0.45

Tabla 2. Calculo de Potencia respecto al diámetro del rotor.

El diámetro del rotor seleccionado conforme al diseño de la pala es de 1.336 m, con respecto a esta dimensión se puede estimar que el aerogenerador es de baja potencia y tendrá una producción efectiva de 0.35 kW.

Para el diseño de las palas del aerogenerador se debe seleccionar un perfil especifico donde el viento tiene una acción sobre el mismo que conforma la pala, donde la “cuerda” (línea que une el borde de fuga con el borde de ataque), forma un ángulo de inclinación “Ω” también llamado ángulo de ataque normal a la dirección del viento como se muestra en la figura 6.

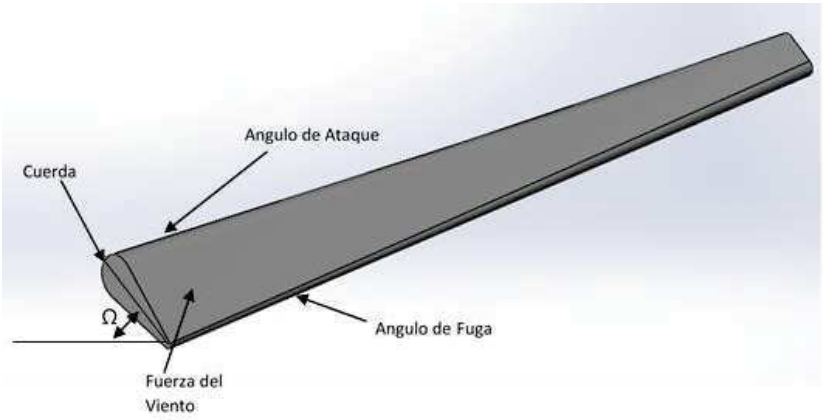


Figura 6. Acción del Viento sobre la pala del aerogenerador.

Las fuerzas originadas por las sobre presiones y depresiones que se presentan en el perfil de las palas del aerogenerador tienen una fuerza resultante “R”, que está en una posición media entre la vertical y la horizontal, respecto a la dirección del viento y aplicada en el centro aerodinámico, que depende del ángulo Ω como se muestra en la figura 7. Esta fuerza resultante R se descompone en dos fuerzas, la fuerza de sustentación (FS) la cual es perpendicular a la dirección del viento y la fuerza de arrastre (FA) que es paralela a la dirección del mismo las cuales vienen dadas por las Ecuaciones (5) y (6). [10]

$$F_s = \frac{1}{2}\left(C_s \rho A v^2\right) \quad \text{.....::(5)}$$

$$F_A = \frac{1}{2}\left(C_A \rho A v^2\right) \quad \text{.....::(6)}$$

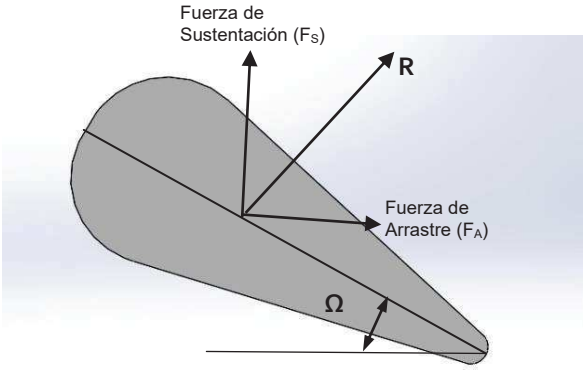


Figura 7: Perfil Aerodinámico de la pala del aerogenerador.

Donde C_s y C_A, son denominados como coeficientes de sustentación y de arrastre respectivamente, que están determinados por el tipo de perfil aerodinámico, las palas del rotor de una turbina eólica normalmente no están estáticas debido a que giran sobre un eje, esto origina una velocidad del aire respecto de la pala diferente a la de la velocidad del viento. El ángulo formado entre la cuerda y el plano de rotación del rotor se denomina ***ángulo de pitch "θ"***. La fuerza **R** puede descomponerse en dos fuerzas. Una fuerza motora **Fm** situada en el plano de rotación y en una fuerza axial **Fa**, paralela al eje de giro del rotor como se muestra en la figura 8.

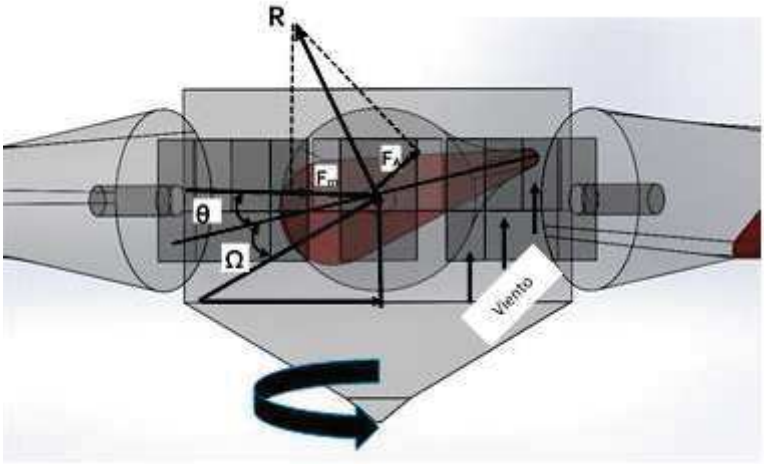


Figura 8. Ángulos y componentes del perfil aerodinámico de la pala.

Los valores de estas dos fuerzas vienen dados por:

$$F_m = F_s sen(\Omega+\theta) - F_A \cos(\Omega+\theta) \quad \text{.....::(7)}$$

$$F_m = F_s sen(\Omega+\theta) - F_A \cos(\Omega+\theta) \quad \text{.....::(8)}$$

Existe un máximo para la relación entre la fuerza de sustentación y la fuerza de arrastre y este máximo se consigue para un valor dado del ángulo de ataque Ω. Es decir, para el elemento de pala considerado existe un ángulo de ataque que es óptimo.

La velocidad del viento varía con la altura, cuando una pala se encuentra en posición perpendicular, la velocidad en la punta de la pala difiere de la velocidad en la raíz de la misma. Suponiendo que la velocidad del viento V es constante a lo largo de la pala, habrá una distancia r al eje del rotor y la velocidad de giro ω es independiente de la distancia. Por consiguiente, el viento relativo en la pala origina que el ángulo de ataque varíe a lo largo de la misma.

El rendimiento en una pala que ha sido diseñada para una determinada velocidad del viento V cambiará cuando lo haga dicha velocidad o lo haga la velocidad de giro del rotor ya que cambia el ángulo de ataque, al cambiar el ángulo de pitch de la corriente de aire incidente Figura 9.

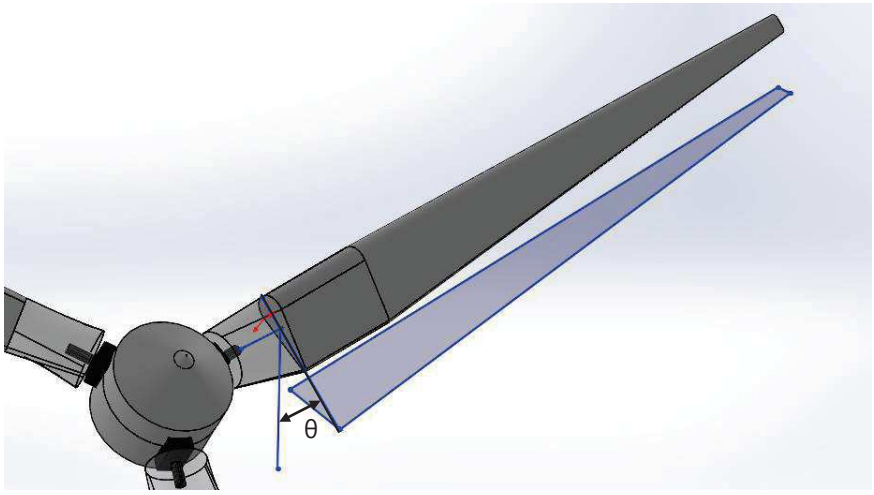


Figura 9: Proyección del Angulo pitch.

Para que la pala proporcione la potencia máxima es necesario que cada segmento de la misma trabaje con un ángulo de ataque óptimo. Para ello es necesario que la pala se pueda construir con diferentes ángulos pitch, para compensar la variación en el ángulo de ataque de la corriente, el cual podrá ser regulado con un servomotor, como se muestra en la figura 10.

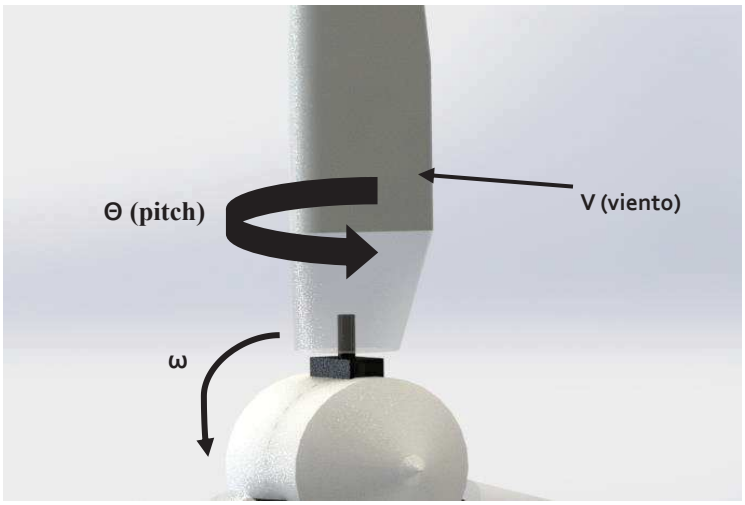


Figura 10: Angulo de paso de pala (pitch).

4.2 Control

Un modelo matemático de un sistema dinámico se define como un conjunto de ecuaciones que representan la dinámica del sistema con precisión [11]. Aunque un modelo matemático no es único para un sistema determinado, ya que un sistema puede representarse de formas diferentes, se puede tener varios modelos matemáticos dependiendo de cada perspectiva.

A continuación se muestra en la figura 11, un diagrama a bloques de un sistema de control el cual se lleva a cabo por medio de un motor de CC, será regulado por un sensor, que detecta la velocidad del viento y a su vez permite el desplazamiento angular en la flecha del mismo, hasta alcanzar los parámetros idóneos para mantener una eficiencia durante la generación.

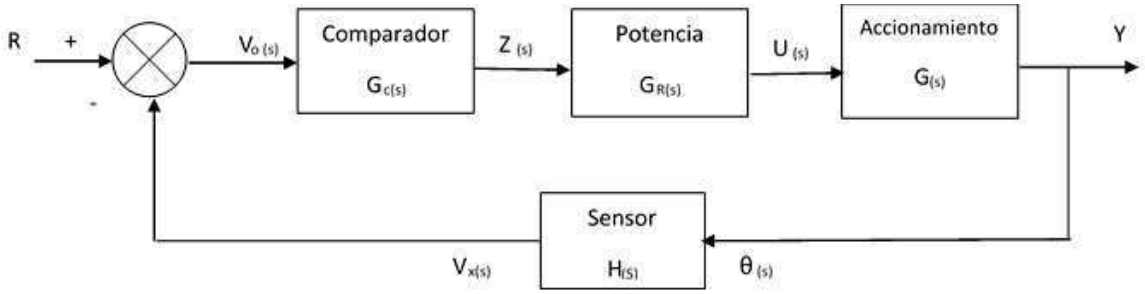


Figura 11. Diagrama de bloques del Control.

La relación entre el desplazamiento Y(s) y el voltaje de entrada Vo(s) está representado en el diagrama de bloques, Por lo tanto, la función de transferencia que relaciona la entrada de referencia $R_{(s)}$ y la salida con la variable controlada $Y_{(s)}$ se le llama ecuación de lazo cerrado, ecuación (9).

$$\frac{Y_S}{R_S} = \frac{G_{(S)}G_{a(s)}G_{c(s)}R_{(s)}}{1+G_{(s)}G_{a(s)}G_{c(s)}R_{(s)}H_{(s)}} \quad \text{.....(9)}$$

Cuando se realiza la comparación, el resultado de la comparación puede tomar dos posibles valores positivo o negativo de acuerdo a la ecuación (10).

$$V\text{err} = G\,(+V - -V) \text{.....(10)}$$

Donde **G** es la ganancia interna del propio comparador, +V es el voltaje de entrada no inversora entrada de la señal, y –V es el voltaje de entrada inversora voltaje del potenciómetro, $\mathbf{V_{err}}$ es el resultado de la comparación también llamado error de comparación, sí éste es positivo el motor dará un giro en un cierto sentido, de lo contrario sí el resultado es negativo el sentido de giro es opuesto al anterior.

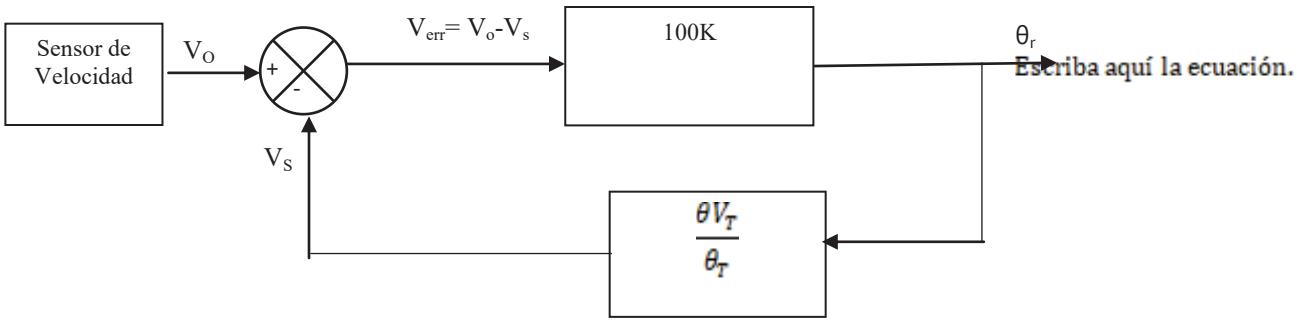


Figura 12. Diagrama de bloques equivalente.

La figura 12, representa el modelo aplicado al ángulo de desplazamiento de la flecha que acciona la pala del aerogenerador también llamado ángulo de paso (pitch) "**θ**", donde se muestra el diagrama equivalente. Por lo tanto, realizando la sustitución obtenemos que el desplazamiento angular ejercido en el motor para el movimiento de la pala se tienen:

$$\frac{\theta_r}{V_{err}} = 100K \quad \text{..... (11)}$$

$$V_{err} = \frac{\theta_r}{100k} \quad \text{.....(12)}$$

Donde:

$$V_{err} = V_o - V_s \quad \text{..... (13)}$$

$$\frac{V_T}{\theta_T} = \frac{V_r}{\theta_r}; \qquad V_s = \frac{\theta_r V_T}{\theta_T} \quad \text{..... (14)}$$

Sustituyendo

$$\frac{\theta_r}{100K} = V_o - \frac{\theta_r V_T}{\theta_T} \quad \text{.....(15)}$$

Entonces:

$$\frac{\theta_r}{100K} + \frac{\theta_r V_T}{\theta_T} = V_o \quad \text{.....(16)}$$

$$\theta_r \left(\frac{1}{100K} + \frac{V_T}{\theta_T} \right) = V_o \quad \text{.....(17)}$$

$$\theta_r \left(\frac{\theta_T + 100KV_T}{\theta_T (100K)} \right) = V_o \quad \text{.....(18)}$$

$$\theta_r = \frac{V_o \theta_T 100K}{\theta_T + 100KV_T} \quad \text{.....(19)}$$

Utilizando la Variable Compleja

$$\frac{\delta^2 \theta_r}{\delta t} \frac{1}{100} + \frac{\theta_r V_T}{\theta_T} = V_{o(S)} \quad \text{.....(20)}$$

$$V_{O(S)} = \theta_{r(s)} \left(\frac{S}{100} + \frac{V_T}{\theta_T} \right) \quad \text{.....(21)}$$

$$\theta_{r(S)} = \frac{V_{o(s)}}{\frac{S}{100} + \frac{20}{\pi}} \quad \text{.....(22)}$$

$$\theta_{r(S)} = V_o \left[\frac{1}{S\left(\frac{S}{100} + \frac{20}{\pi} \right)} \right] \quad \text{.....(23)}$$

$$\theta_{r(S)} = 100V_o \left[\frac{1}{S\left(S + \frac{2000}{\pi} \right)} \right] \quad \text{.....(24)}$$

Utilizando la transformada Inversa de Laplace, para obtener la función del tiempo se tiene:

$$\theta_{r(t)} = \frac{1}{20} \pi V_o \left[1 - e^{-\frac{2000(t)}{\pi}} \right] \quad \text{.....(25)}$$

5. Resultados

Una vez obtenida la relación entre el desplazamiento angular efectuado por la flecha del servomotor (ángulo pitch) en relación al voltaje de entrada del comparador, se sustituyen los valores de la ecuación 25, se establecen los siguientes resultados indicados en la Tabla 3:

Velocidad del Viento	Voltaje Sensor equivalente a la velocidad	Desplazamiento	
(m/s)	(Vo)	Θr (radianes)	Grados
9	5.00	0.78	45
8	4.44	0.70	40
7	3.89	0.61	35
6	3.33	0.52	30
5	2.78	0.44	25
4.5	2.50	0.39	22.5
4	2.22	0.35	20
3	1.67	0.26	15
2	1.11	0.17	10
1.8	1	0.16	9

Tabla 3. Desplazamiento en relación a la Velocidad del Viento

El desplazamiento máximo en la pala del aerogenerador es de 45º y un desplazamiento mínimo de 0º, el objetivo es mantener una velocidad promedio en el aerogenerador por lo que la orientación estará determinada en base a la velocidad detectada por el sensor, como se muestra en la figura 13 donde la respuesta está determinada por el voltaje.

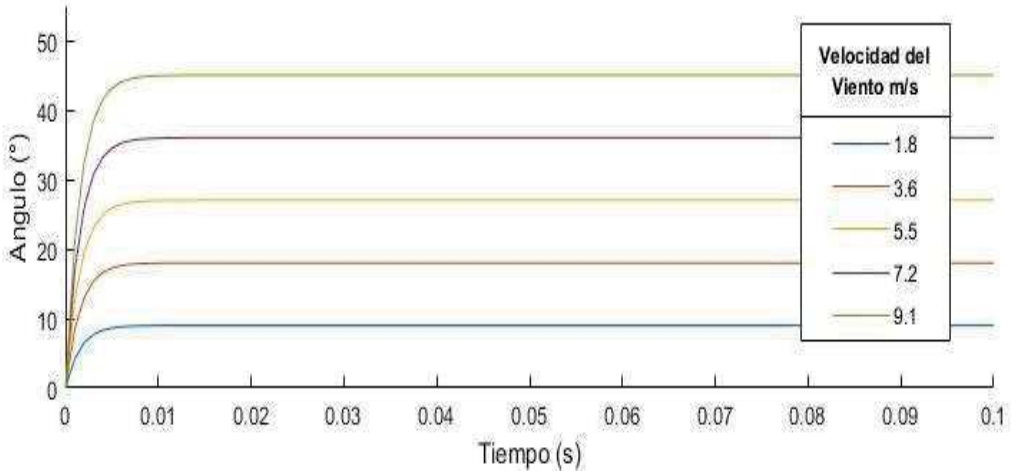


Figura 13. Respuesta del ángulo de desplazamiento en función de la velocidad del viento

El movimiento de las palas estará orientado a una mayor área de contacto con el viento, cuando la velocidad del mismo tenga una variación, como se muestra en la figura 14 (a y b).

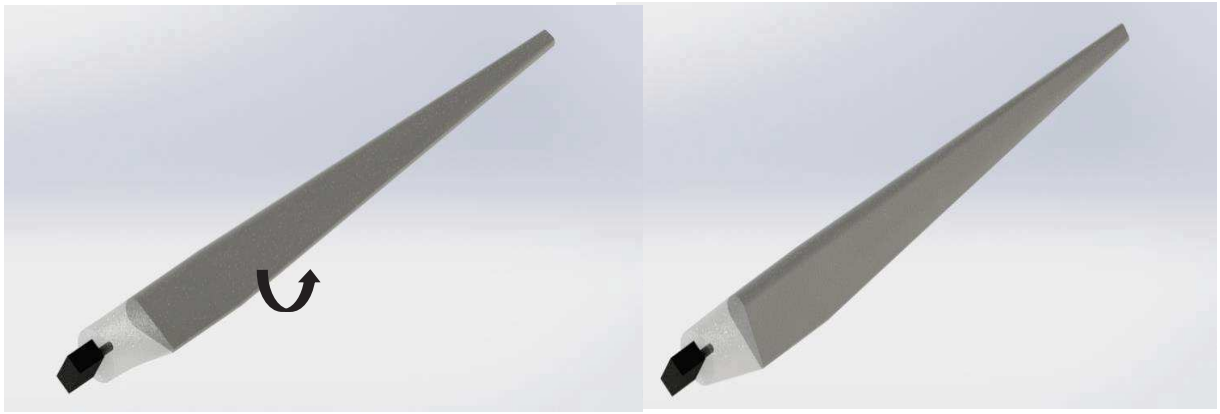


Figura 14. Ángulo de Palas; a) desplazamiento 0°; b) desplazamiento 22.5°

Los movimientos que se transmiten al servomotor ocurren bajo un movimiento angular hasta cierta potencia eléctrica suministrada por las diferentes velocidades del viento. La figura 15, nos permite observar el acomplamiento que existe entre cada sevomotor y el eje de rotacion.

La fuerza longitudinal ejercida por el aire en el rotor de la turbina eolica debe reducirse. Para lograr una eficiencia adecuada, la mayoría de la energía eólica debe ser transformada por las palas en fuerza de elevación en lugar de fuerza de arrastre (fuerza longitudinal).

Para diferentes casos de velocidad del viento, los resultados de la simulación de variacion del angulo de paso (pitch) mejora en términos de potencia de salida de manera que se puede aprovechar más energía eólica en diferentes condiciones de funcionamiento.

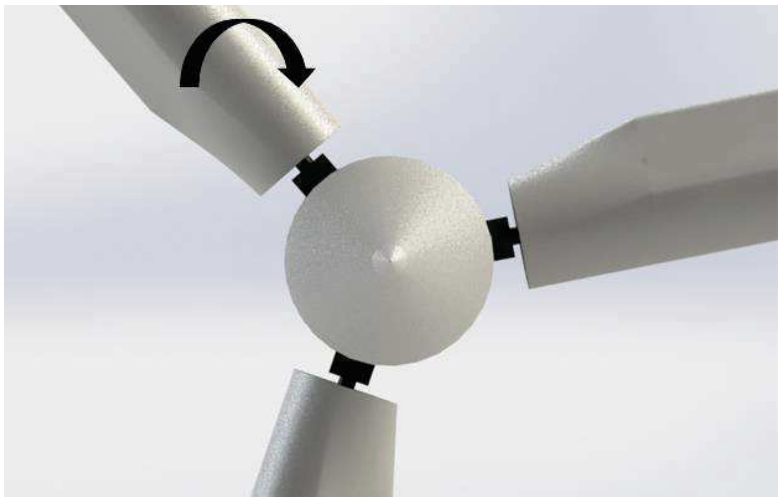


Figura 15. Acoplamiento pala-rotor.

El sistema de control del ángulo pitch tiene dos objetivos por un lado busca maximizar la energía eólica capturada y convertida en energía eléctrica y por otro lado, minimizar la fatiga y la carga mecánica, el mecanismo del servomotor acoplado se muestra en la figura 16, en la figura se muestra la instalación del servomotor y el potenciómetro como sensor de posición.

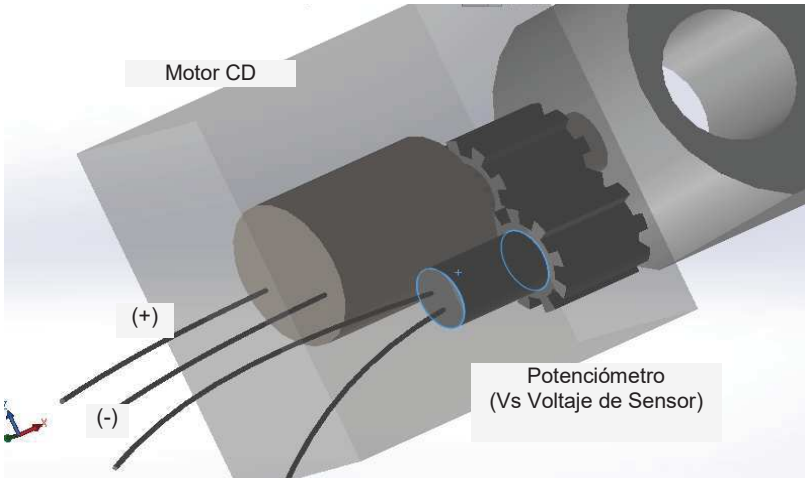


Figura 16: Servomotor.

Ingeniería y Automatización, Capítulo 5, pp. 59 - 76.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

El circuito equivalente propuesto, se muestra en la figura 17.

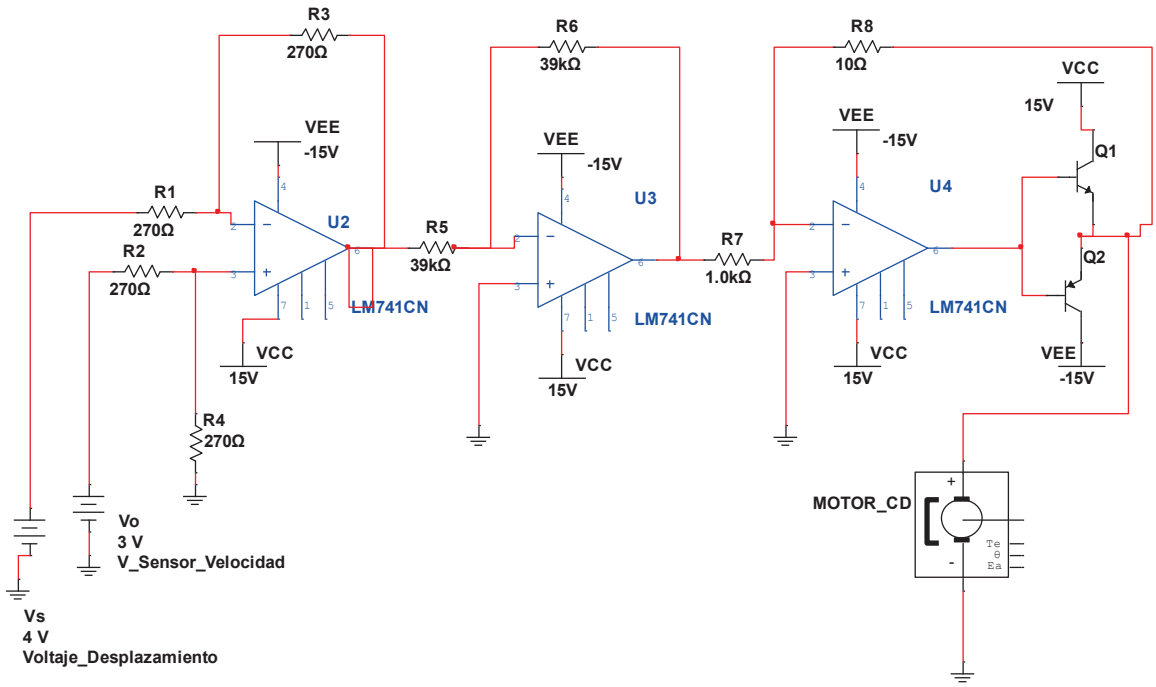


Figura 17. Diagrama equivalente del circuito físico.

6. Conclusiones

Bajo el planteamiento se ha podido calcular los parámetros que permiten establecer la función de transferencia de un servomotor acoplado a las palas de un aerogenerador para manipular el ángulo pitch y con ello simular la respuesta del sistema ante distintas variaciones de velocidad.

El modelo desarrollado puede aplicarse en el diseño para conocer previamente el comportamiento de un servomotor acoplado a una pala de un aerogenerador y así poder ajustar el ángulo pitch de la misma para elevar la superficie de contacto y con ello poder elevar la potencia del sistema. Cuando el desplazamiento angular de la flecha del motor se define en función de la magnitud de la diferencia del error es decir, sí la diferencia es pequeña el desplazamiento angular será pequeño, sí la diferencia es mayor entonces el desplazamiento angular de la flecha del motor será mayor. En el caso de la resultante de la salida del comparador se suministra a la etapa de potencia para proporcionar la corriente y voltaje necesario al motor, el servomotor se comporta como un sistema de lazo cerrado con retroalimentación. El desarrollo del análisis dinámico del servomotor se realiza considerando los elementos que componen el sistema de retroalimentación y las características del motor. Las variables importantes que se consideraron en este modelo son: Voltaje de entrada y el ángulo de desplazamiento (θ) final como respuesta del motor, a un voltaje de alimentación al servomotor como constante (Vo) de 5volts.

El desarrollo del prototipo se encuentra en la etapa de diseño para la prueba física y verificación de datos con el modelo matemático del controlador, donde se someterá a diferentes pruebas en un túnel de viento y obtener resultados de forma experimental.

7. Referencias

[1] Gilbert M., Renewable and Efficient Electric Power Systems, Published by John Wiley & Sons, Inc., Hoboken, New Jersey, 2004

[2] GWEC. Global cumulative and annual offshore wind capacity 2011-2016; 2017, [Fecha de consulta: 04 de julio de 2018] disponible desde <http://gwec.net/global-figures/graphs/>

[3] ("Danish Wind Industry Association." [Online]. Available: http://www.windpower.org/en. [Acceso: 01-Mar-2018]).

[4] S. Xu, R. Shao, L. Chang and C. Church, "Energy Cost Estimation of Small Wind Power Systems An Integrated Approach," IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 3, no. 4, 2015, pp. 945-956.

[5] SENER. Secretaria de Energía México. Ley para el Aprovechamiento de Energías Limpias y el Financiamiento de la Transición Energética.

[6] González J., Pérez R., Santos A., Gil M., Centrales de energías renovables: generación eléctrica con energías renovables, Pearson Educación, S.A., Madrid, 2009.

[7] Guevara D. Cuantificación del perfil del viento hasta 100 m de altura desde la superficie y su incidencia en la climatología eólica. Terra Nueva Etapa [en linea] 2013, XXIX (Julio-Diciembre): [Fecha de consulta: 20 de agosto de 2018] Disponible en:<http://www.redalyc.org/articulo.oa?id=72130181006> ISSN 1012-7089.

[8] Norman, S. (2011). Control systems engineering. USA: John Wiley.

[9] Abad G., López J., Rodríguez M., Marroyo L. and Iwansk G., Doubly Fed Induction Machine: Modeling and Control for Wind Energy Generation, Wiley, 2011.

[10] Dyachuk M., Rossander A., Goude H., "Measurements of the aerodynamic normal forces on a 12-kW straight-bladed vertical Axis wind turbine, Energies 8 (no. 8) (2015) 8482e8496.

[11] Ogata, K. (2010). Ingeniería de control moderno. España: Prentice Hall.

[12] Bolton W. (2008). Mechatronics, electronic control systems in mechanical and electrical engineering. Canada: Prentice Hall.

[13] Wood D, (2011). Small Wind Turbines Analysis, Design, and Application, Springer

[14] Burton T., Sharpe D, Jenkins N, Bossanyi E, (2001), Wind Energy Handbook, Wiley.

[15] Mukund R. (2006) Wind and Solar Power Systems, Design, Analysis, and Operation.

Ingeniería y Automatización, Capítulo 6, pp. 77 - 86.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño de un Horno para Laminar Vidrio de Luz Inteligente

Mora Mora Omar✉, Mora Santos Carlos Alberto, Hernández Corona Francisco, Bedolla Hernández Jorge y Cuatecontzi Xochitiotzi Desiderio

Tecnológico Nacional de México, Instituto Tecnológico de Apizaco
Avenida Instituto Tecnológico S/N C.P. 90300, Apizaco, Tlaxcala
✉omm82@hotmail.com

Resumen

En el presente artículo se expone el diseño de un horno para laminar vidrio de luz inteligente; es decir, un horno de tipo horizontal con etapas independientes para realizar procesos de termofusión, entre placas de vidrio plano, láminas de materiales plásticos de PVB y láminas de LED. El diseño esta basado en la metodología de desarrollo de productos y se desarrolla las dos primeras fases de dicha metodología: en la primera fase se obtendrá la solución a una necesidad planteada mediante el despliegue de la casa de la calidad QFD para poder determinar los bloques de la constitución de la máquina y en donde se establece mediante una ponderación porcentual los puntos a considerar al momento de realizar los bloques involucrados, en el desarrollo de la segunda fase se describen los bloques resultantes que involucra una aplicación de la ingeniería para poder determinar el tipo de sistema a desarrollar, para finalmente obtener los resultados de la obtención de los primeros croquis y diseños CAD que satisfagan el óptimo desarrollo cumpliendo la necesidad y cubrir las metas de diseño planteadas en las tablas de desarrollo y así obtener un primer modelo de operación.

Palabras clave: La casa de la calidad QFD aplicada a desarrollo de productos, Diseño CAD

1. Introducción

El vidrio laminado incorpora capas intercaladas de materiales plásticos como él *Butiral de Polivinilo* (PVB), Cuando el cristal es laminado recibe el nombre de vidrio de seguridad. Estas láminas son del tipo transparentes y suelen incluir materiales que dan un alto impacto de diseño como lo será incluir elementos como diodos (*Light Emitting Diode)* LED del tipo *(Surface-Mount-Device)* SMD y así recibir el nombre de vidrio de luz inteligente Figura 1 [1].

El vidrio laminado sirve para reducir el peligro de trozos de vidrio y asegura que las personas no se caigan por aperturas inesperadas en superficies flotantes. Así como evitar el ruido, y bloquear los rayos de luz *ultra violeta* (UV), y En vehículos y edificios cumplir con lo establecido por la Norma Oficial Mexicana NOM-146-SCFI-2001, Productos de vidrio-Vidrio de seguridad usado en la construcción-Especificaciones y métodos de prueba [2].

Ingeniería y Automatización, Capítulo 6, pp. 77 - 86.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

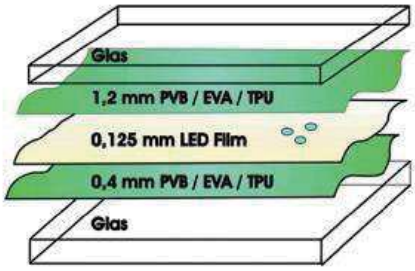


Figura 1. Especificación Típica de Vidrio Laminado con Película con Led Integrados

Existen dos procedimientos de laminado el primero es el *procedimiento por laminación* en el que el intercalario es una película sólida que se coloca entre las hojas del vidrio y se somete a un calentamiento y a una presión para obtener el producto acabado y el segundo es *laminación por fundición* en el que el intercalario se obtiene vertiendo un líquido entre las hojas de vidrio, que posteriormente es químicamente curado para obtener el producto final [3].

Existen hornos en el mercado que realizan este fin común, pero estos suelen ser de un costo muy elevado y trabajan bajo el principio de autoclave al vacío que generan una presión de 12 a 14 kg/cm^2 está al contraerse generan paredes irregulares lo que conlleva a realizar un mal laminado por aglomerados de adherencia.

En esta investigación se trabaja con el primer método, las láminas de PVB son compatibles a la fusión con las láminas de LED SMD y su adherencia con el cristal es a una temperatura no mayor a 120°C, resultando este método como seguro a la compresión.

Para lograr esto se Diseña un horno del tipo horizontal para poder realizar el proceso de termofusión de un vidrio de luz inteligente en donde se busca por medio de la compresión la expulsión de los aglomerados y el cual debe alcanzar una temperatura que garantice la adhesión máxima entre los mismos, dicho sistema está constituido por sistemas independientes, obtenidos de los resultados del despliegue de la casa de la calidad QFD.

2. Metodología de Diseño

Dado que se tiene un problema de diseño, se utilizará la metodología clásica del diseño mecánico cuya tarea fundamental es: “crear lo que no existe”. En el presente artículo, se describe la primera y segunda fase de la metodología desarrollo de productos figura 2 [4].

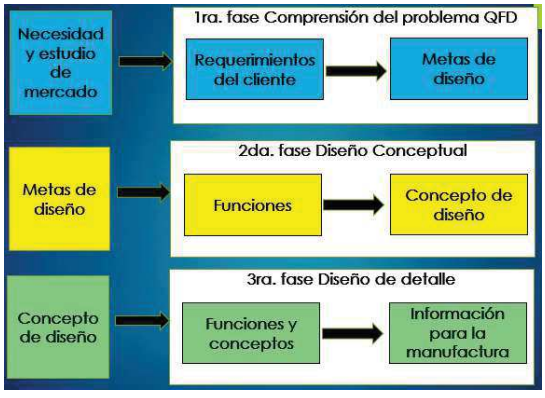


Figura 2. Metodología de Diseño “Desarrollo de Productos”

2.1 Comprensión del Problema QFD

Lo que se necesita: *"Una máquina que logre realizar la unión térmica de un acetato de PVB (Butiral de Polivinilo) y una lámina de LED (Light Emitting Diode) del tipo SMD (Surface-Mount-Device) entre dos vidrios planos con la finalidad de obtener un vidrio de luz inteligente"*

Para limitar las tareas de diseño de cada bloque se utilizan varias estrategias mentales, siendo la más común los cuestionamientos: ¿En que no puedo o no debo profundizar? ¿Qué componentes puedo conseguir ya fabricados? ¿Qué puedo aproximar y que debo diseñar? El responder correctamente a estas cuestiones permite ahorrar tiempo y esfuerzo al momento de diseñar y se ve reflejado económicamente en la fabricación (solo si ya está validado el diseño). Las respuestas a estos cuestionamientos constituyen los criterios y consideraciones de diseño, estos se exponen en el diseño conceptual.

La Casa de la Calidad, Análisis de necesidades y expectativas o **QFD** (Quality Function Deployment) es la metodología que se usó en este artículo, para **crear productos que se adapten a las necesidades del usuario**, figura 3.

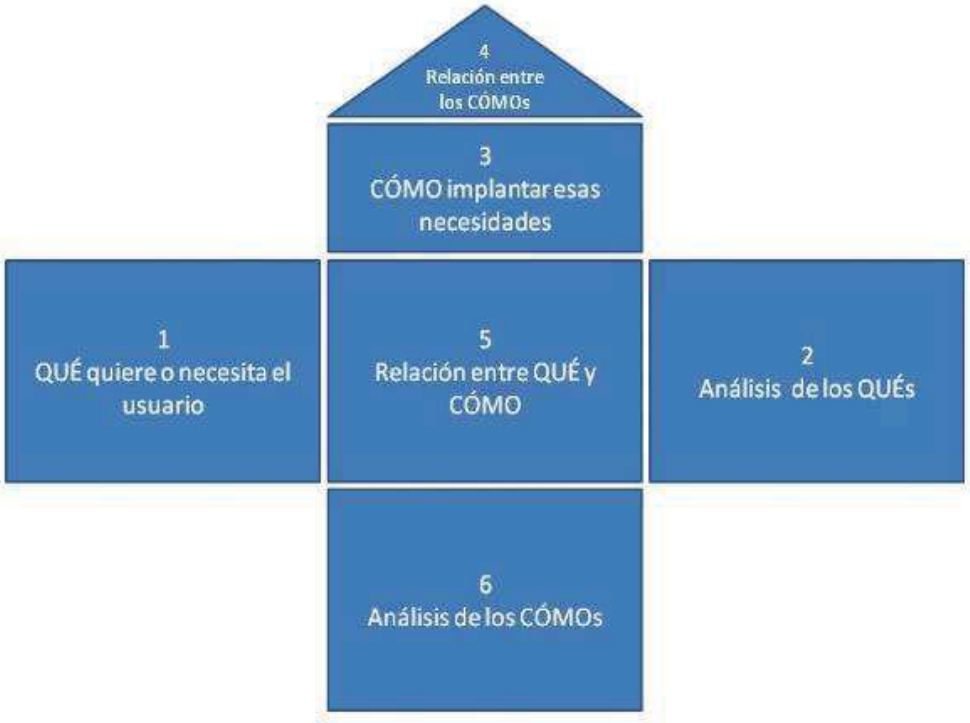


Figura 3. Estructura del QFD

De esta forma, con esta metodología se puede determinar qué características debe añadirse al producto. También permitirá saber cuáles son las características no necesarias que aportan un sobrecosto al producto sin ser apreciadas por el usuario y dará una visión de cómo está nuestro producto frente a la competencia para poder decidir cuáles son los aspectos prioritarios por mejorar, tal y como se muestra ejecutado el despliegue del QFD en la figura 4 [5].

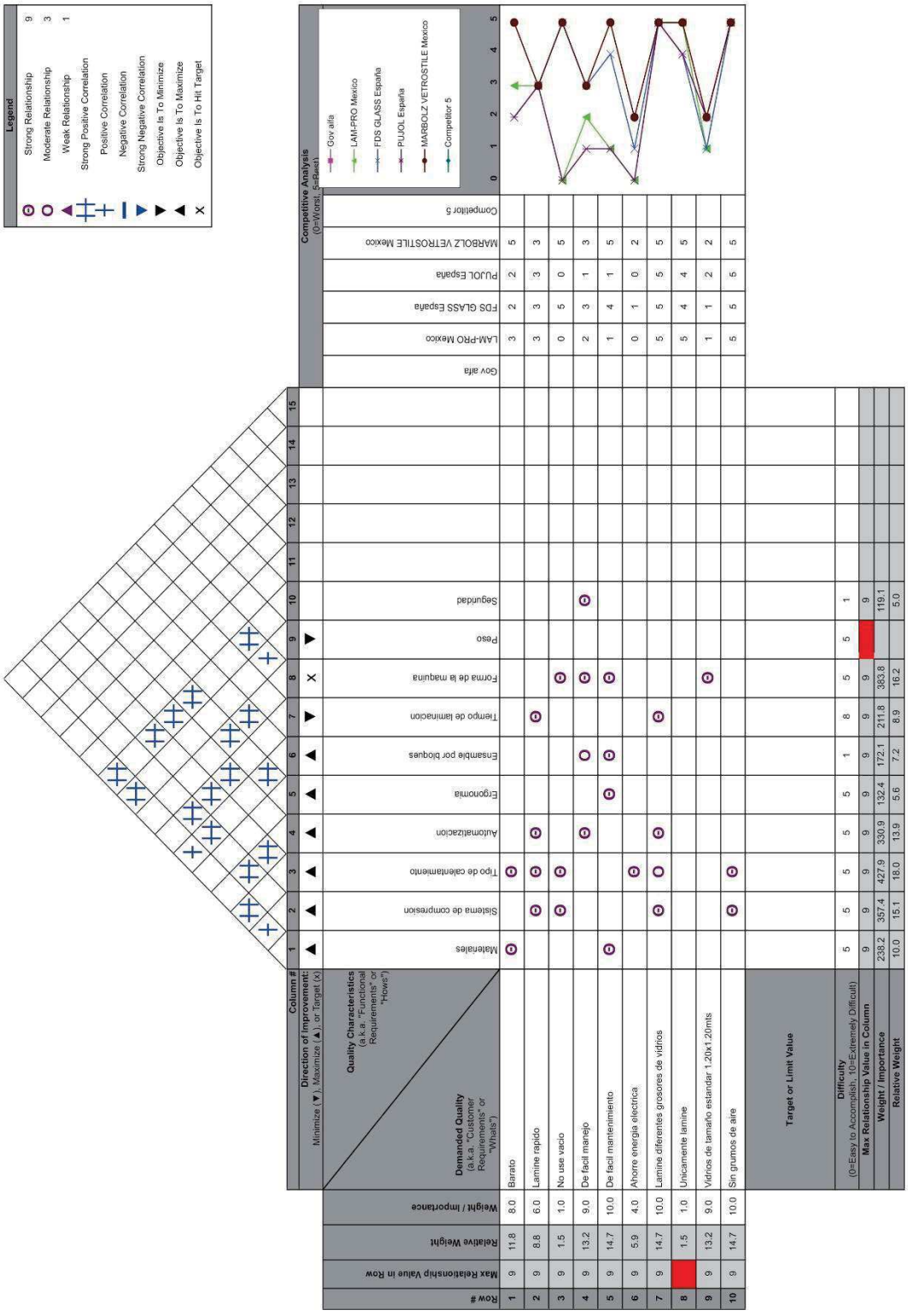


Figura 4. Despliegue de la Casa de la Calidad QFD

Ya con los datos claros obtenidos del análisis del QFD, se procede a discretizar cada concepto en sistemas o bloques que en conjunto realicen la función completa que el concepto describe. Los bloques se categorizan en la forma en que estos serán diseñados o por el tipo de tarea que representa su solución. Además, es necesario visualizar en que se convierte el bloque al ser solucionado pudiendo ser un sistema mecánico, un sistema eléctrico, un manual de usuario o una capacitación por mencionar algunos. Se enlistan los bloques de cada concepto en la tabla 1.

Tabla 1. Tabla de Bloques y sus Respectivas Tareas

TABLA DE BLOQUES POR CONCEPTO			
CONCEPTO	**BLOQUE**	**TIPO DE TAREA**	**RESULTADO**
MAQUINA	MECANISMO DE PRESIÓN	DISEÑO MECÁNICO	SISTEMA MECÁNICO
	ESTRUCTURA	DISEÑO MECÁNICO	SISTEMA MECÁNICO
	PLANCHAS TÉRMICAS A Y B	DISEÑO MECÁNICO	SISTEMA MECÁNICO/ELÉCTRICO
	SISTEMA TÉRMICO	DISEÑO MECÁNICO/ELÉCTRICO	SISTEMA MECANICO/ELÉCTRICO
	SISTEMA DE CONTROL	DISEÑO ELECTRÓNICO	SISTEMA ELECTRÓNICO
	SISTEMA ELÉCTRICO	DISEÑO ELÉCTRICO	SISTEMA ELÉCTRICO
LAMINACIÓN (UNIÓN TÉRMICA)	PROCESO DE LAMINADO	INVESTIGACIÓN Y PRUEBAS	MANUAL DE USUARIO, MODIFICACIONES A OTROS BLOQUES
	METODOLOGÍA	DISEÑO DE FÁCIL PROCESO DE LAMINADO Y PRUEBAS	MANUAL DE USUARIO Y CAPACITACIÓN
	RECOMENDACIONES	DISEÑO DE FÁCIL PROCESO DE LAMINADO Y PRUEBAS	MANUAL DE USUARIO Y CAPACITACIÓN
VIDRIO	TABLA DE MODELOS	INVESTIGACIÓN Y PRUEBAS	MANUAL DE USUARIO
	RECOMENDACIONES	PRUEBAS	MANUAL DE USUARIO
PLACAS DE ACETATO (PVB Y LED SMD)	SELECCIÓN DE ACETATOS PVB Y LED SMD	INVESTIGACIÓN Y PRUEBAS	FICHAS TÉCNICAS DE LOS ACETATOS
	LAMINADO CORRECTO DE LOS COMPONENTES	PRUEBAS	MANUAL DE USUARIO Y CAPACITACIÓN

2.2 Diseño Conceptual

Consiste en ejecutar posibles soluciones de manera general, sin entrar en detalles, que logren satisfacer las metas de diseño. Se describe el proceso de diseño conceptual de cada uno de los bloques que constituyen la visión global de la **máquina**. Cada bloque será descrito brevemente y se resumirá en una tabla sus consideraciones de diseño y las primeras decisiones de diseño CAD.

Mecanismo de Presión

Dado que se está hablando de una máquina semiautomática, un motor en conjunto con un sistema reductor de velocidad es el encargado de aportar la potencia al sistema. La fuerza que desarrolla debe ser trasmitida y multiplicada por un mecanismo que logre la presión correcta en los vidrios planos y los acetatos. La posición de descanso tiende a ser con la plancha térmica hacia arriba, mientras que la de trabajo tiende a ser con la plancha térmica hacia abajo con el fin de usar la gravedad a favor. El mecanismo ejercerá una presión gradual hasta que el tiempo de planchado se cumpla tabla 2.

Tabla 2. Consideraciones de Diseño del Mecanismo de Presión

BLOQUE: MECANISMO DE PRESIÓN

CONSIDERACIONES Y CRITERIOS DE DISEÑO DE:	
INGENIERÍA	**USUARIO**
SIMPLE Y ECONÓMICO	FÁCIL DE USAR
ROBUSTO Y DE LARGA VIDA	CÓMODO PARA REPETICIONES
PRECISO	NO APARATOSO
POCO JUEGO RADIAL	POCO IMPACTO VISUAL
RESISTENTE AL DESGASTE	MINIMIZAR RIESGOS
RESISTENTE A ALTA TEMPERATURA (120ºC-250ºC)	
PRESIÓN DE 2-6Bar	
BAJO Y FÁCIL MANTENIMIENTO	
CARRERA DE 15cm LINEAL	
SIN MOVIMIENTO SEMICIRCULAR	

Estructura

La estructura de la máquina será la encargada de sostener y proteger todas sus partes funcionales y móviles, como son el mecanismo de presión, las planchas térmicas A y B y el sistema de control. Se puede entender que la estructura será el esqueleto de la máquina y es una pieza fundamental en la apariencia global de la misma tabla 3.

Tabla 3. Consideraciones de Diseño de la Estructura

BLOQUE: ESTRUCTURA	
CONSIDERACIONES Y CRITERIOS DE DISEÑO DE:	
INGENIERÍA	**USUARIO**
SOPORTAR MECANISMO CON MENOR JUEGO POSIBLE	DISEÑO ESTÉTICO
DE CIERRE PARALELO	PROTECCIÓN DE PARTES ELÉCTRICAS
SIN MODIFICAR SU ALTURA DE TRABAJO (2mts)	PROTECCIÓN DE SUPERFICIES CALIENTES
PESO MENOR A 500kg.	SEÑALAMIENTOS NECESARIOS
PATAS EMPOTRADAS A PISO	FÁCIL DESARME PARA MANTENIMIENTO. ELÉCTRICO
PESO DISTRIBUIDO HACIA ATRÁS	COLORES SOBRIOS (NEGRO.GRIS, BLANCO)
GRASERAS PARA LUBRICACIÓN	
GUARDAS PARA POLVO Y TEMPERATURA	
ENSAMBLAJE POR MÓDULOS	
VOLUMEN APROXIMADO 1.80mtsX2. 50mts.X1.80mts.	
PINTURA PARA ALTA TEMPERATURA	

Planchas Térmicas A y B

Las planchas térmicas A y B, son el componente que entra en contacto con los vidrios y los acetatos para proveer la presión y calor necesario para que los acetatos se activen tabla 4.

Tabla 4. Consideraciones de Diseño de las Planchas Térmicas A y B

BLOQUE: PLANCHAS TERMICAS A Y B
CONSIDERACIONES Y CRITERIOS DE DISEÑO DE:

INGENIERÍA	USUARIO
DE MATERIAL ALUMINIO	DISMINUIR RIEGOS
EVITAR FUNDICIÓN Y MOLDES	PROTECCIÓN DE SUPERFICIES CALIENTES
AMINORAR CONTACTOS QUE PROVOQUEN DAÑOS AL LAMINADO	COMODO DE TRABAJAR
GEOMETRÍA ADECUADA A LA FORMA	NO CONFIGURABLE
ÁREA DE LAMINADO (1. 20mts.X 1.20mts.)	ÁREA DE IMPRESIÓN DESEADA
EVITAR PIEZAS INTERCAMBIABLES	
SELECCIÓN DE TIPO DE RESISTENCIA	
POTENCIA DE RESISTENCIA	
MATERIALES INOXIDABLES (COSTOSO)	
INTERFACE A MECANISMO	
TAMAÑO CERCANO A LA AREA DE LAMINADO DESEADA	
CABLEADO PARA ALTA TEMPERATURA	
UBICACIÓN DEL SENSOR DE TEMPERATURA	
MODELO TÉRMICO DE LAS PLANCHAS	
UBICACIÓN DE RESISTENCIAS	
PROTECCIÓN DE CONEXIÓN ELÉCTRICA Y CABLEADO	
AISLANTE TÉRMICO	
TEMPERATURA AMBIENTE	
FUGAS DE CALOR	
PASTA TÉRMICA (TERMOPOSOS)	

Sistema de Potencia Eléctrica y de Control

Los circuitos eléctricos de potencia y control son los encargados de proveer la energía al sistema para su funcionamiento y mantener las variables de trabajo dentro de los parámetros de funcionamiento o de especificación de usuario tabla 5.

Tabla 5. Consideraciones de Diseño del Sistema Eléctrico

BLOQUE: SISTEMA DE POTENCIA ELÉCTRICA Y DE CONTROL	
CONSIDERACIONES Y CRITERIOS DE DISEÑO DE:	
INGENIERÍA	USUARIO
VOLTAJES IGUAL O MENOR A 110V	INTERFASE CÓMODA
CONTROLADOR DE TEMPERATURA	ESTÉTICA
ELEMENTO DE PROTECCIÓN	MANEJO SENCILLO
CABLEADO PARA ALTA TEMPERATURA	MINIMIZAR RIEGOS
CABLES CERCA DE PARTES MÓVILES ARMADAS Y ROBUSTAS	CABLEADO Y CIRCUITO OCULTO
CONEXIONES ROBUSTAS PARA ALTA TEMPERATURA	
“STRAIN RELIEFS” Y ACCESORIOS PARA CIRCUITO	
LIMIT SWITCH PARA TIMER (O SENSOR DE PRESENCIA)	
FÁCIL CONEXIÓN Y MANTENIMIENTO	
PROGRAMACIÓN RÁPIDA	
DIAGRAMA	

3. Resultados

Se obtiene el diagrama de flujo de la máquina que indicara la secuencia de operación de la siguiente manera: Para comenzar, se debe encender la laminadora. A continuación, se debe introducir los parámetros de trabajo de temperatura, presión y tiempo deseados en la pantalla táctil.

Una vez introducidos los parámetros, se calentará la laminadora hasta que alcance la temperatura deseada. Al alcanzar la temperatura deseada, se introduce la muestra de acetatos y vidrios planos, se colocan las dos planchas térmicas A y B en posición vertical y comienza el proceso de laminación como se describe en siguiente diagrama de flujo [6].

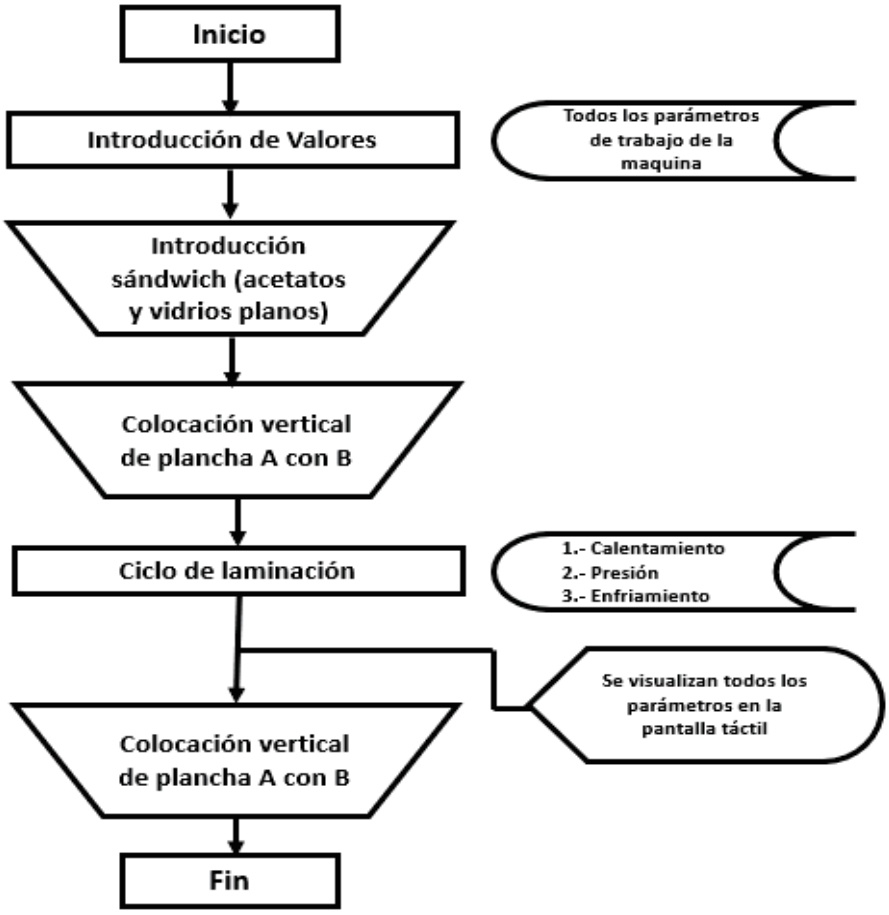


Figura 5. Diagrama de Flujo de Operación de la Maquina Laminadora

Los resultados obtenidos del desarrollo de las dos primeras fases de la metodología de desarrollo de productos es la generación de una máquina compuesta por bloques o sistemas independientes, que son desarrollados y solucionados con la integración de normas y consideraciones de diseño de ingeniería bajo el criterio de una primera solución, que serán ejecutadas en diseño CAD que satisfaga la necesidad del sistema y en conjunto la del cliente, tal y como se muestra en la solución de cada bloque de la máquina figura 6 [7].

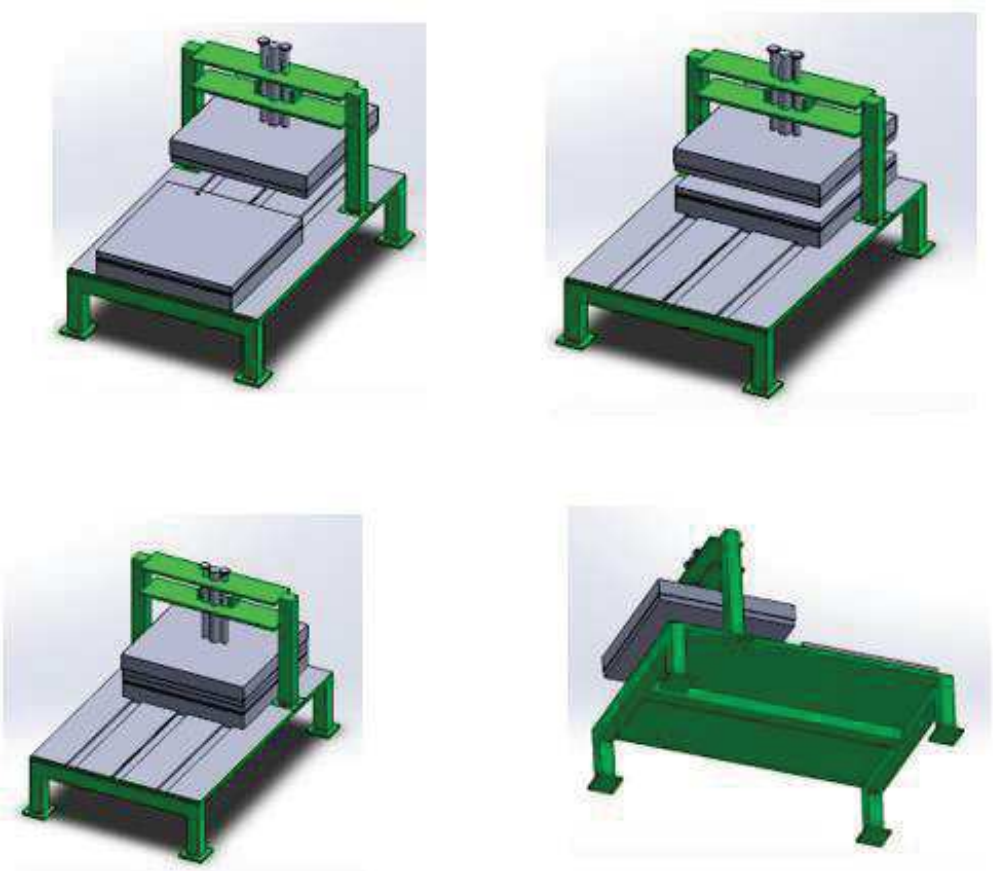


Figura 6. Imágenes de Diseño CAD, que Constituyen el Ensamble Final de las Etapas de la Maquina

4. Conclusión

En el presente artículo se desarrolló el diseño CAD de una maquina laminadora de vidrio de luz inteligente, a partir de una necesidad propuesta por el cliente y que es abordada mediante la metodología de investigación de desarrollo de productos, cuya aportación es a aplicar métodos de soluciones por ponderación como lo es el QFD lo que determina los parámetros de diseño y las metas de diseño a cubrir, la cuales se identificaron por bloques con la finalidad de seccionar las etapas que constituye la máquina, y así dar la satisfacción de producto del cliente.

Referencias

[1] Jean M., “Histoire des Polymeres”, Ligne, Septiembre 2017

http://www.societechimiquedefrance.fr/IMG/pdf/histoire_industrielle_intro.vfx2_sav.pdf,

Francia, 2000.

[2] Diario Oficial de la Federación, NOM-146-SCFI “Vidrios de Seguridad”, Septiembre 2017

http://dof.gob.mx/nota_detalle.php?codigo=765879&fecha=09/05/2001, México, 2014

[3] VIDRESIF “Vidrio Laminado” Girona España, Tutorial, 2014.

[4] Ramos J., “Diseño Mecánico vol. 2”, IPN ESIME SEPI, México, Segunda Edición, 2003.

[5] Jimeno J., “Despliegue de la Función de Calidad QFD”, Febrero 2018 https://www.pdcahome.com/1932/qfd-despliegue-calidad/ , México, 2012.

[6] Giesecke F., Mitchell A., Spencer H., Hill I., Dygdon J., Novak J., “Dibujo y Comunicación Grafica”, Pearson/Prentice Hall, México, Tercera Edición, 2006.

[7] García C., “SolidWorks Dibujo y Diseño Mecanico”, I.T.C.J., México, Tutorial, 2001

Ingeniería y Automatización, Capítulo 7, pp. 87 - 100.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Prototipo de Lectura y Escritura en Código Braille y Bastón para Ciegos

Ortega Blanco Christian Aarón, Corral Quintanar Álvaro Martín, Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen y Platt Borbón Martina Elisa

Universidad de Sonora, División de Ciencias Exactas y Naturales, Departamento de Investigación en Física y División de Ingeniería y Departamento de Ingeniería Industrial.
christomajuro@hotmail.com

Resumen

El objetivo de la investigación es la construcción de dos prototipos de bajo costo, el primero es para lectura y escritura en código Braille, el cual lo pueden manipular tanto personas que pueden ver para capacitarse en aprender el código, así como personas débiles visuales y ciegos; el otro es un prototipo para bastón de ciegos. El dispositivo para lectura y escritura en braille utiliza un teclado numérico USB para escribir, y para leer utiliza el sentido del tacto con seis actuadores. Utiliza bocinas externas de la computadora para escuchar los tonos de voz audibles, para su desarrollo se emplea una computadora de bajo costo la RaspBerry Pi modelo B y dispositivos magnéticos para activar las señales a los actuadores para generar el código braille, la programación es en Python bajo la plataforma de Raspbyan que es una versión de Linux para el dispositivo electrónico. Por otra parte, la necesidad de bastones hoy en día es mucha y generamos un producto económico y durable, después de evaluar varias opciones de materiales.

Palabras clave: Braille, Linux, Python, RaspBerry

1. Introducción

El objetivo de la investigación es la construcción de dos prototipos, el primero es un dispositivo para lectura y escritura en código Braille, el cual lo pueden manipular personas que pueden ver para capacitarse en aprender el código, así como personas débiles visuales y ciegos y el otro es un prototipo para bastón de ciegos que sea económica su fabricación, La ceguera o debilidad visual es considerada por organismos internacionales como la segunda discapacidad más inhabilitante, afecta a 467 mil personas en México. Pese al subregistro de casos que señalan investigadores y especialistas, también es considerada como la segunda causa de discapacidad en nuestro país.

Si bien la población más afectada son adultos y ancianos, 17.2 por ciento de quienes padecen discapacidad visual en México son menores de 30 años; 33 por ciento tiene entre 30 y 59 años de edad, mientras que 48.8 por ciento es mayor de 60 años, ya que las causas principales son edad avanzada y enfermedades, con 33 por ciento de los casos, respectivamente; 12.4 por ciento por accidentes y 11.2 por males congénitos.

De acuerdo con información del Instituto Nacional de Estadística, Geografía e Informática [1], 58.4 por ciento no cuenta con acceso a los servicios de salud, mientras que sólo 21.1 por ciento de la población rural afectada es derechohabiente de alguna institución de salud, contra un promedio de 51.4 por ciento en las zonas urbanas.

Existen diversos dispositivos para lectura-escritura del código braille como ejemplo tenemos:
La pantalla de bolsillo Braille Focus 14 Blue cuenta con un teclado de 8 puntos altamente receptivo y un cómodo diseño de control con conectividad USB y Bluetooth. Con comodidad,

portabilidad y funciones avanzadas, lea los mensajes de texto, navegue por la pantalla de su teléfono inteligente e ingrese texto, todo sin sacar el teléfono de su cinturón ni sacarlo de su bolso. [2]

La principal desventaja que presenta este sistema es el costo tan elevado (alrededor de 995 dólares que equivalen a $18,435.00 pesos m.n.), además de los gastos de importación e impuestos que se generen. En la Figura 1 puede visualizarse el teclado Focus Blue Braille Display.

Figura 1. Focus Blue Braille display.

Otro sistema que existe en el mercado es el El Braille´n speak o braille hablado es uno de los aparatos con más aceptación entre las personas con discapacidad visual. Es un aparato manejable, portátil, de poco peso (medio kilo) y tamaño reducido (como un libro de bolsillo) que se utiliza para almacenar y procesar información. [3]

Además, tiene distintas funciones de lectura y escritura, y permite comunicarse con otros sistemas a través de su unidad de disco. La entrada de datos se realiza mediante teclado braille y la salida de información es a través de voz sintética. Consta de un editor de textos para manejar y tratar la información, agenda, calendario (fecha y hora), cronómetro, diccionario, calculadora, etc. Se puede conectar a otros dispositivos como impresora braille o tinta. Se maneja a través de «macros» o acciones ejecutadas a través de una simple pulsación de tecla o instrucción. Ver figura 2.

Figura 2. Braille´n Speak.

Otro sistema más completo es el Braille lite 2000 que es una combinación de ambos: táctil y se escucha, sin embargo, se encuentra descontinuado. En la figura 3 se presenta una imagen de dicho sistema. [4]

Figura 3. Braille lite 2000.

Para todas las opciones el alto costo hace que resulten inaccesibles para las personas de bajos recursos económicos. Este trabajo presenta una opción del desarrollo de un prototipo similar al sistema Braille lite 2000, de bajo costo, que sea táctil y que emita en tonos de voz audibles lo representado en el código Braille.

2. Métodos

2.1 Dispositivo de Lectura y escritura braille

El Braille [5] es un sistema de lectura y escritura táctil pensado para personas ciegas. [6] Se conoce también como cecografía. Fue ideado por el francés Louis Braille a mediados del siglo XIX, que se quedó ciego debido a un accidente durante su niñez mientras jugaba en el taller de su padre. Cuando tenía 13 años, el director de la Escuela de Ciegos y Sordos de París donde estudiaba el joven Braille le pidió que probara un sistema de lecto-escritura táctil inventado por un militar llamado Charles Barbier de la Serre para transmitir órdenes a puestos de avanzada sin tener necesidad de delatar la posición durante las noches. Louis Braille descubrió al cabo de un tiempo que el sistema era válido y lo reinventó utilizando un sistema de 8 puntos. Al cabo de unos años lo simplificó dejándolo en el sistema universalmente conocido y adoptado de 6 puntos.

En las figuras 1 y 2 se muestra el código Braille. Y sus dimensiones

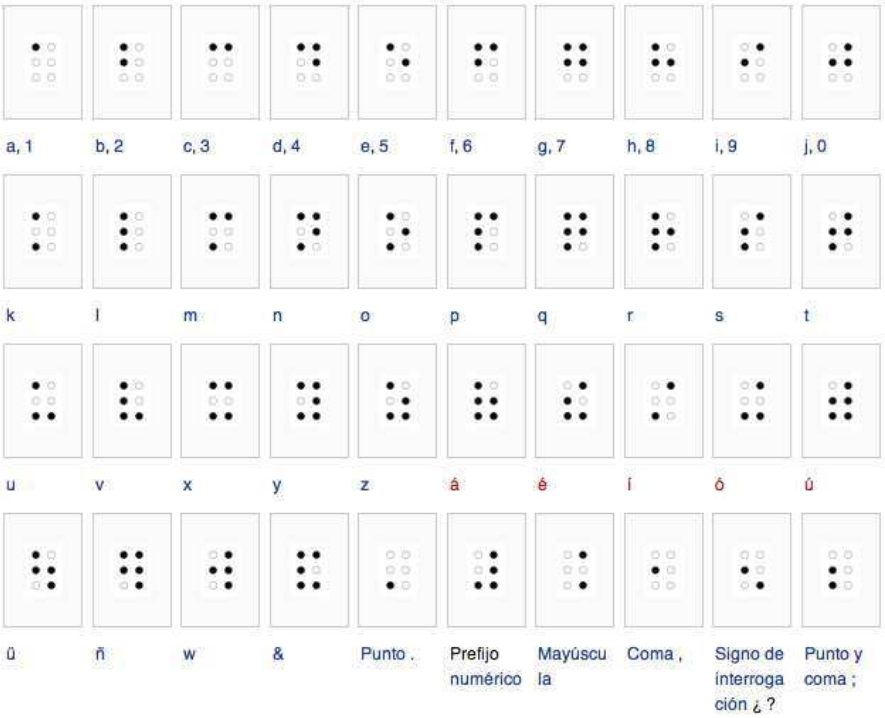


Figura 1. Código Braille de 6 puntos.

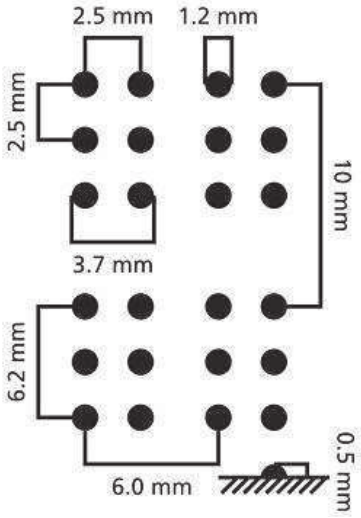


Figura 2. Dimensiones del código braille.

Se utilizará tecnología: mecánica, electrónica y de sistemas de información que a fin de cuentas es mecatrónica el dispositivo para lectura y escritura en código braille se utilizara una RaspBerry Pi [7], un teclado numérico y dispositivos magnéticos para activar las señales a los actuadores para generar el código braille, [8] la programación es en Python bajo la plataforma de Raspbyan que es una versión de Linux para el dispositivo electrónico. Como podemos observar en la figura 3 se inicializa la RaspBerry, la interface con los motores vibradores para generar el código Braille.

Figura 3. Inicializando el Prototipo.

En la figura 4 se le requirió a una persona débil visual que probara el equipo, se le enviaba información a los actuadores y la persona tenía que leer el código enviado a través del tacto con sus dedos.

Figura 4. Probando el prototipo con una persona débil visual.

El diagrama a bloques lo podemos observar en la figura 5

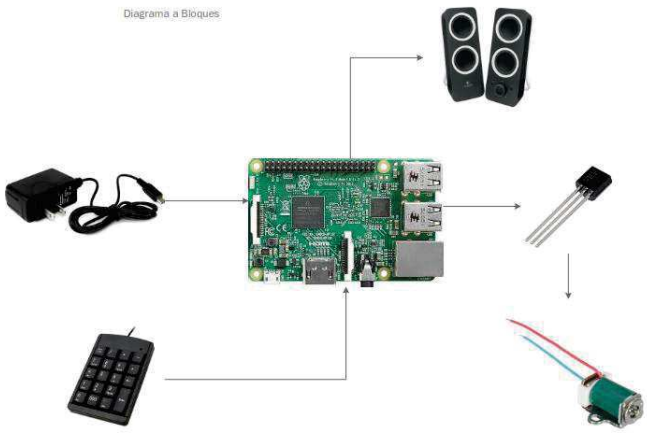


Figura 5. Diagrama a bloques.

En la figura 6 se presenta el Diagrama de Flujo empleado para el funcionamiento de lectura y escritura del teclado.

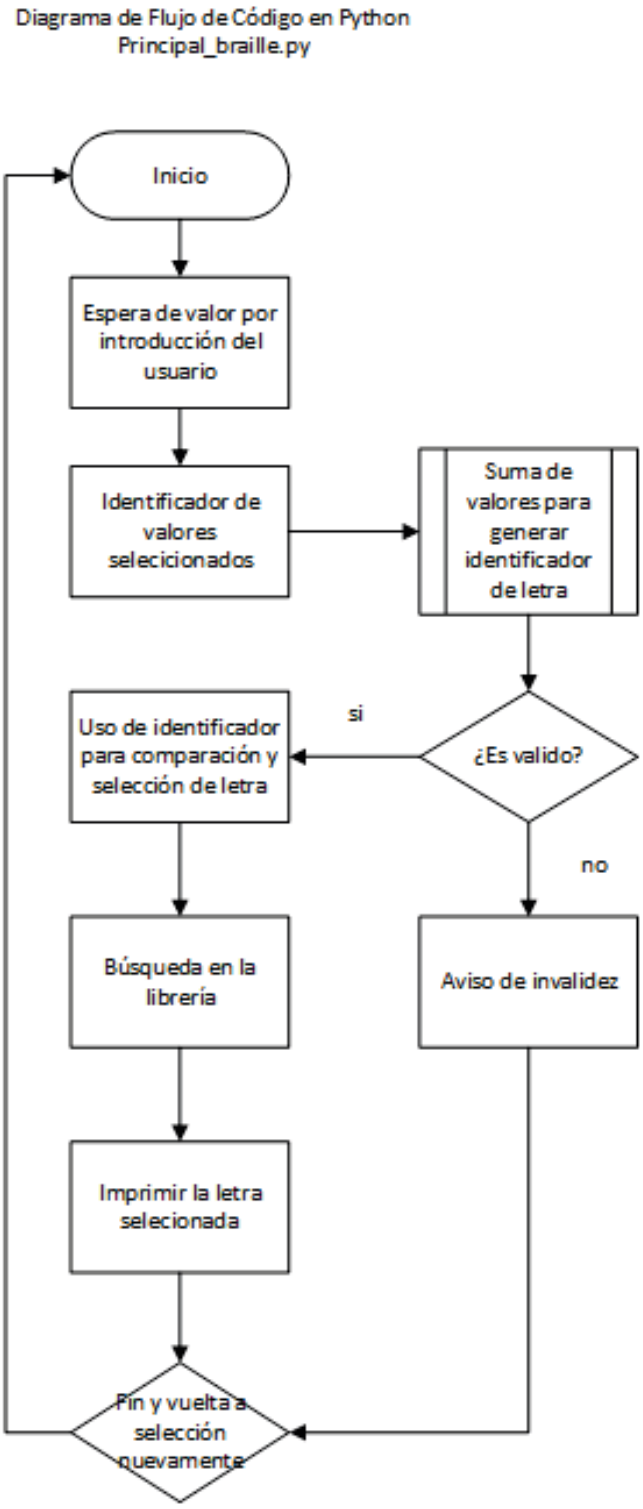


Figura 6. Diagrama de flujo del prototipo de teclado Braille.

Detectando el valor de Z1 descubre que letra es y enciende los motores 1, 2, 3, 4, 5, 6 en su caso e imprime la letra en el display se observa en tabla 1.

Tabla 1. De acuerdo al código generado con las teclas se encienden los motores y se envía al display la letra seleccionada

Z1	Encender motor	letra
55	1	a
107	1,2	b
111	1,4	c
164	1,4,5	d
109	1,5	e
163	1,2,5	F
216	1,2,4,5	G
160	1,2,5	H
108	2,4	I
161	2,4,5	J
104	1,3	K
156	1,2,3	L
160	1,3,4	M
213	1,3,4,5	N
266	1,2,4,5,6	ene
157	1,3,5	O
212	1,2,3,4	P
265	1,2,3,4,5	Q
209	1.2.3.5	R
157	2,3,4	S
210	2,3,4,5	T
154	1,2,3,4	U
206	1,2,3,6	V
211	2,4,5,6	W
210	1,3,4,6	X
263	1,3,4,5,6	Y
207	1,3,5,6	Z

2.2 Los bastones para ciegos

El bastón supone para el ciego una prolongación de su cuerpo. Esta herramienta táctil, fabricada con diversos materiales, guía los pasos del invidente, haciéndole notar sobre qué tipo de superficie está caminando y proporcionándole asimismo información acerca de su ubicación. El contacto del puntero con el suelo remite unas vibraciones al ciego, que son más intensas cuanto más dura es la superficie sobre la que se encuentra. En el mercado, existen actualmente distintos modelos de bastones, cada uno de los cuales presenta un tipo diferente de agarre.

Ejemplos:
1.- Bastón para invidentes retráctil de la empresa total medic. Con un costo de $486.00 pesos m.n. mas envío. [9]

2.- Bastón plegable para invidente [10] con un costo de $259.00 pesos m.n.

Para la creación del prototipo se utilizaron materiales muy diversos en forma de tubo como: Aluminio, metálico cromado, metálico Conduit, PVC Hidráulico Blanco y PVC para uso Eléctrico de color verde.

Otro requisito que fueran plegables, en cuatro segmentos: la punta, dos intermedios y un final con mango o empuñadura. Unidos mediante una cuerda elástica por su interior y coples para fijar dos segmentos contiguos.

Para usuarios adultos y niños con medidas de 1.20 Mt Para adultos y 80 Cm para niños, deben de medir aproximadamente a la altura de su pecho al piso. En la figura 8 podemos apreciar el proceso de fabricación de los bastones.

Figura 8. Proceso de fabricación

En la figura 9 se puede observar varios tipos de bastones para ciegos de diferente material como: aluminio, metálico cromado, PVC blanco hidráulico, PVC verde para instalaciones eléctricas y Conduit también para instalaciones eléctricas. En dos versiones para adultos y para niños.

Figura 9. Prototipos con diferente material.

Después en la figura 10 se observa cómo se pesan los bastones en una báscula electrónica

Figura 10. Se pesaron todos los prototipos.

Y posteriormente en la figura 11 se le solicitó a un grupo de personas débiles visuales y ciegos que probaran cada uno de los bastones y nos dieran su opinión para evaluar los distintos materiales con los que fueron fabricados, de una muestra de 50 personas se hizo el análisis y se registró en la Tabla 2, considerando que el valor 1 es el nivel más alto y el 5 es el valor más bajo.

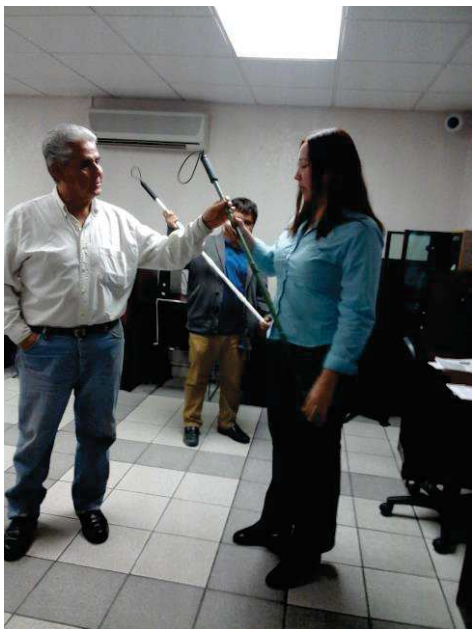


Figura 11. Se solicitó la opinión de los usuarios

Como podemos observar en la tabla 2 se probaron 5 materiales distintos todos son tubos, con sus coples y punta con regatón o con material de Nylamid, con empuñadura de bicicleta, todos unidos con una cuerda elástica.

Tabla 2 Comparativo de bastones para ciegos.

Material	Costo m.n.	Peso gramos	Opinión de usuarios
Aluminio	$60.70	190.70	1
Tubo cromado	$79.05	380.30	2
PVC blanco	$58.20	380.30	3
PVC verde	$63.49	205.30	4
Conduit	$50.80	610.90	5

3. Resultados

.
Se obtuvo el prototipo para lectura/escritura de braille en figura 12, con un costo aproximado de $3,000.00 pesos m.n. bastante más accesible que los demás. Es una pequeña computadora con acceso a internet por cable Ethernet, WiFi, Bluetooth, salida HDMI para monitor, 4 puertos USB para conectar teclado numérico, mouse, etc. Salida de audio para bocinas externas, se puede conectar a la corriente AC con un cargador de teléfono celular. 6 actuadores para la matriz de braille, expandibles a más.

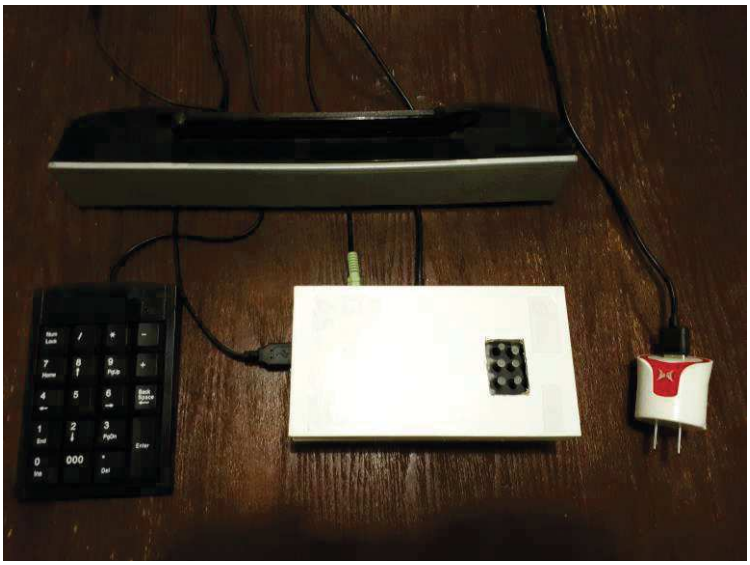


Figura 12. Prototipo de lectura/escritura Braille

Se eligió el bastón de aluminio por el peso muy ligero, firmeza para que no se doblen con el uso y opinión de usuarios en la figura 13 con un costo de $60.70 m.n. de materiales.

Figura 13. Bastón de aluminio

Para la fabricación de los bastones para ciegos, se está en proceso de constitución de una entidad persona moral sin fines de lucro, en el cual se han involucrado los maestros asesores. La organización tendrá como objeto social la fabricación de bastones para ciegos y débiles visual y su posterior distribución gratuita a aquellos a quienes no les sea posible su compra por dificultades económicas. La organización estará financiada por empresas y personas físicas, así como organismos gubernamentales mediante donaciones y aportaciones que apoyen esta problemática social.

La organización pretende solicitar autorización a la Secretaría de Hacienda y Crédito Público para emitir facturas deducibles de impuestos que permitan incentivar a la población a efectuar donaciones voluntarias. La distribución de bastones hará posible que los ciegos y débiles visuales que no están en posibilidades económicas de adquirir un bastón, puedan desplazarse fuera de sus hogares de manera confiable, mejorando de manera sustantiva su calidad de vida. Además, se tiene contemplado incorporar dentro de la organización, personas con esta discapacidad que estén en posibilidades de realizar actividades productivas, generando un ingreso que apoye la economía familiar.

Por otra parte, el dispositivo para escribir y leer braille es muy útil para personas que pueden ver porque se capacitan en el aprendizaje con el código Braille, en una versión posterior al momento de teclear o leer las letras la micro computadora por medio de las bocinas conectadas reproducirán un audio, por ejemplo: está usted escribiendo la letra “A”, o está usted leyendo la letra “A” y así para todas letras, números y caracteres, se ampliarán a 4 matrices de 6 actuadores para poder leer varios caracteres a la vez.

4. Conclusiones

Se presenta el desarrollo de un prototipo de teclado para lectura y escritura en código Braille empleando tecnología mecánica, electrónica y de sistemas de información. Se empleó una RaspBerry Pi, un teclado numérico y dispositivos magnéticos, obteniendo con éxito la interfaz para el código Braille. Logrando el objetivo de bajo costo.

También se desarrollaron prototipos de bastones más ligeros y que a su vez cumplen con las especificaciones requeridas para el uso de débiles visuales y ciegos. Logrando el objetivo de bajo costo.

Referencias

[1] Instituto Nacional de Estadística y Geografía (INEGI) consultado 14/06/2018 Disponible en: http://www.inegi.org.mx/

[2] Página del producto: **Focus Blue Braille display**, consultado 6/08/2018 Disponible en: https://www.freedomscientific.com/Products/Blindness/Focus14BasicBrailleDisplay

[3] Página del producto: Braille´**n Speak** consultado 6/08/2018 Disponible en: http://www.ite.educacion.es/formacion/materiales/129/cd/unidad_10/m10_otras_herramientas.htm

[4] Página del producto: **Braille lite 2000** consultado 6/08/2018 Disponible en: http://www.dbt.org.au/Internet%202011/AcceTech/Braille%20Lite%202000.htm

[5] Álvarez Morales, Miguel. (2010). Luis Braille. La luz en los dedos (Biografía Joven). Editorial Casals, ISBN 978842184555.

[6] Martínez, Rogelio. (2009). Braille el acceso de los ciegos al conocimiento, Editorial MAD,ISBN 9788467620801

[7] RaspBerry Pi Modelo B. consultado 14/06/2018 Disponible en: https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf

[8] Upton Eben, Halfacree Gareth . RaspBerry Pi Guía del Usuario. Consultado 14/06/2018 Disponible en: https://dlscrib.com/queue/raspberry-pi-gu-iacute-a-del-usuario-2da-ed-en-espa-ntilde-ol_58966c216454a7601db1e8dc_pdf?queue_id=59a47f06dc0d602b60568edc

[9] Página del producto: Bastón para invidente consultado 6/08/2018 Disponible en https://totalmedic.com.mx/productos/baston-para-invidente-con-rondada.html?gclid=Cj0KCQjw45_bBRD_ARIsAJ6wUXQPEBrBAr-y20VdfmN5ZRKfodbbQn_54L1prhG-5hDMhSOz_av_LWoaAlVaEALw_wcB

[10] Página del producto: Bastón plegable para invidente consultado 6/08/2018 Disponible en https://articulo.mercadolibre.com.mx/MLM-630333233-baston-plegable-para-invidente-_JM

Ingeniería y Automatización, Capítulo 8, pp. 101 - 110.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Interfaz de Usuario y Sistema de Navegación para un Vehículo Eléctrico para Discapacitados

Delgado Arcega Rodrigo Oliver, Robles Julio Cesar y Gómez Becerra Fabio Abel

Instituto Tecnológico Mario Molina Pasquel y Henríquez, rodrigo.delgado@vallarta.tecmm.edu.mx, julio.robles@vallarta.tecmm.edu.mx, fabio.gomez@vallarta.tecmm.edu.mx

Resumen

El presente trabajo tiene la finalidad de enlazar una interfaz natural de usuario y un sistema de navegación autónomo a una silla de ruedas convencional con dos modalidades de control: manual y automático. Se tiene en cuenta que no todas las personas tienen el mismo tipo o grado de discapacidad, por esta razón se busca implementar una interfaz de usuario que conlleva diferentes formas de interacción usuario – máquina, con las que el usuario podrá elegir el modo manual que permite órdenes directas de control de movimiento de la silla y el modo automático, que solo necesita el punto o lugar al que se necesita ir y mediante trayectorias predefinidas, el sistema de navegación autónoma ejecutará los movimientos necesarios para que la silla cumpla con la ruta deseada.

Palabras clave: control, vehículo eléctrico, control, autónomo, lazo de control.

Introducción

La Organización Mundial de la Salud (OMS), en un informe sobre las personas con discapacidad, menciona que más de mil millones de personas viven en todo el mundo con alguna forma de discapacidad, 200 millones de ellas presentan dificultades en cuanto a su movimiento. Estas personas tienen muchas limitaciones de accesibilidad a muchos de los servicios que una persona sin discapacidad podría considerar muy necesarios u obvios.

La discapacidad motora es la dificultad que presentan algunas personas para participar en actividades propias de la vida cotidiana, que surge como consecuencia de la interacción entre una dificultad específica para manipular objetos o acceder a diferentes espacios [1]. En los últimos años la tecnología ha tenido un vertiginoso avance en sus diversas ramas de estudio, respecto a este proyecto la inteligencia artificial y la robótica, han dado a la humanidad la realización de tareas de una forma eficaz y autónoma.

Una gran sección de la robótica e inteligencia artificial aplicada a la ayuda para discapacitados está orientada al desarrollo de las sillas de ruedas inteligentes [2] . Estas innovaciones pretenden elevar la calidad de vida de las personas de tal modo que el usuario de la silla de ruedas se evite la necesidad de estar controlando cada avance y giro de la silla durante el recorrido, permitiéndole simplemente sentarse y relajarse mientras se desplaza hacia el lugar que el usuario manda.

Vehículos para discapacitados

Las sillas de ruedas autónomas apoyan de manera significativa al usuario, permitiendo la evasión de obstáculos, en algunos casos navegación autónoma, entre otros. Las sillas de ruedas autónomas son sillas convencionales que se han equipado con sistemas informáticos y sensores; los sensores captan el entorno y el estado de la silla, éstos mandan señales al controlador y éste los procesa para la ejecución de tareas especificadas en la programación [3].

Si bien este proyecto no utiliza exactamente dicho concepto, “silla de ruedas autónoma” sus funcionalidades aplican de manera muy directa a él, ya que se instrumentará una silla de ruedas convencional con un sistema informático y sensorial, por lo que dicho proyecto finalmente tendrá un producto el cual pueda ser considerado así, como una silla de ruedas autónoma, sin embargo, esta constará con dos modos de operación, manual y automático. Los robots móviles, en este caso sillas de ruedas, pueden basarse en diferentes diseños de plataformas, que se diferencian por los diversos sistemas de tracción que utilizan. El sistema de tracción diferencial es el que será usado para el prototipo ya que es uno de los más usados; además, para motorizar una silla de ruedas convencional es el sistema de tracción más fácil de implementar. Un sistema de tracción diferencial utiliza motores independientes para cada rueda, pero situados en el mismo eje, además, utiliza ruedas locas o puntos de sostenimiento para proporcionar estabilidad a la plataforma [4]. han propuesto un diagrama de un sistema de motorización de una silla de ruedas de un sistema de motorización de una silla de ruedas convencional (ver Figura 1).

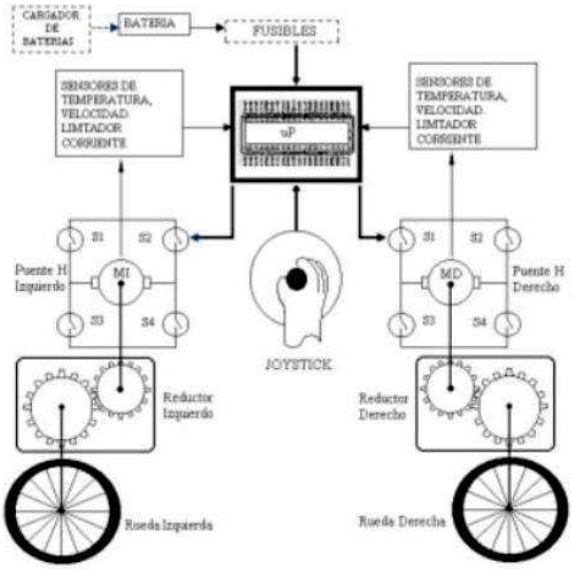


Figura 1. Diagrama de motorización de una silla de ruedas

Según el diagrama, la motorización de una silla de ruedas conlleva un sistema de control que recibe mandos desde un joystick manual y en consecuencia manipula los motores para llevar a cabo la acción. El sistema de tracción que utilizan en dicho diagrama es el diferencial [5].

Las prestaciones deseables para una silla de ruedas autónoma son las siguientes:
1. Actuación fácil y de ser posible automática sobre los movimientos de la silla, obedeciendo en todo caso los deseos del usuario.
2. Si fuera posible, desplazarse entre dos localizaciones especificadas por el usuario sin necesidad de intervenciones posteriores o correcciones de ruta.
3. Desplazamiento a velocidades normales, tomando como velocidad máxima 5-6 Km/H y como mínima 1 Km/H.
4. Seguridad: evitar accidentes cualesquiera, sea por choque contra elementos fijos u otros móviles o por caídas.

Por otro lado, se debe hacer un estudio descriptivo de la problemática de las personas con discapacidades motoras para el desarrollo de un proyecto de control de una silla de ruedas [6], esto con el fin de tener una mayor cobertura de las necesidades de los posibles usuarios de la silla y sugiere la metodología siguiente:

1. Identificar las necesidades del objeto de estudio.
2. Idear y realizar las opciones de solución a las necesidades de los posibles usuarios.
3. Estudiar la tecnología de los dispositivos móviles y a la interconexión vía Bluetooth entre el dispositivo y el controlador (hardware).
4. Adaptar los accesorios a la silla de ruedas (sensores, baterías, motores, entre otros).
5. Desarrollar el programa de control.

Otro de los puntos importantes del proyecto, es otorgar al usuario, en modo manual pero más que nada en modo automático, la oportunidad de que el vehículo evite impactar objetos estacionales (muros, jardineras, entre otros) y objetos temporales (como pueden ser personas, objetos varios, entre otros).
Los robots móviles que tienen como una de sus funciones detectar obstáculos [7], usan una serie de sensores para obtener la información necesaria del entorno que les rodea para poder interactuar con él.

El funcionamiento básico de los sensores ultrasónicos como medidores de distancia, cuenta con un receptor que emite un pulso de ultrasonido, el cual rebota sobre un determinado objeto y la reflexión de ese pulso es detectada por un receptor de ultrasonidos.
Existen factores inherentes tanto a los ultrasonidos como al mundo real, que influyen de una forma determinante en las medidas realizadas. Por tanto, es necesario un conocimiento de las diversas fuentes de incertidumbre que afectan a las medidas para poder tratarlas de forma

Para llevar a cabo todos estos procesos, la robótica á recurrido a los sistemas de control, que son aquellos dedicados a obtener la salida deseada de un sistema o un proceso. Se clasifican principalmente en sistemas de control en lazo abierto y lazo cerrado (Figura 2).

Figura 2. Diagrama de un sistema de control de lazo cerrado.

Como ya se ha dicho anteriormente, la tracción del prototipo de silla de ruedas tendrá tracción diferencia, que a continuación se mostrarán los aspectos básicos que deben considerarse para el modelado de este sistema. Se comienza a partir de un diagrama en un plano xy con un vehículo de donde se toma el centro de masa (Figura 3).

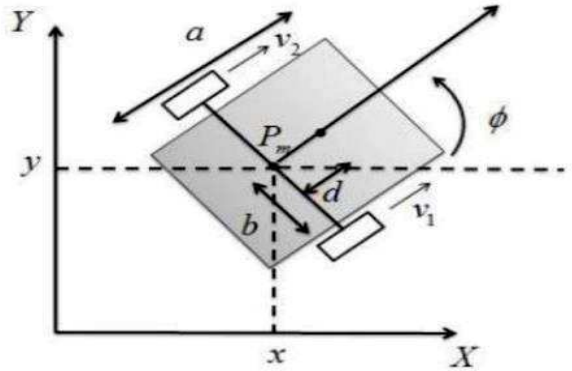


Figura 3.Esquema generalizado de un robot móvil diferencial.

Propuesta del modelo cinemático:

$$\dot{x} = v * \cos(\theta) \quad (1)$$
$$\dot{y} = v * sen(\theta) \quad (2)$$
$$\theta = w \quad (3)$$

El modelo cinemático está dado por las ecuaciones anteriores, las variables de interés son v y w ya que estas son las entradas del sistema. “No obstante, la velocidad lineal del robot y el cambio de dirección del vehículo se obtienen por el cambio de velocidades angulares izquierda y derecha, wi y wd, correspondientes a la velocidad de las ruedas [1]. Entonces, es posible modificar las ecuaciones para encontrar la relación entre v y w y las velocidades angulares wi y wd. En esta configuración se consideran únicamente las dos llantas delanteras del robot móvil y se supone que sobre cada una actúa una fuerza Fi y Fd que proporcionan las velocidades lineales del punto de apoyo de las ruedas, vi y vd, respectivamente. Nótese que el sistema es no lineal, al contener funciones senos y cosenos en el modelo cinemático, por lo que es necesaria una linealización para desacoplamiento entrada – salida. A este tipo de linealización se le conoce como linealización entrada-estado. Un sistema no lineal de grado relativo r caracterizado en términos de campos vectoriales y derivadas direccionales de Lie esta dado de la forma:

$$\dot{q}=f(q)+g(q)u \quad (4)$$
$$y=h(q) \quad (5)$$

Donde “h” es la salida del sistema, “$\dot{q}$” es el vector de estados y “u” es la entrada de control. El procedimiento de linealización exacta comienza derivando respecto al tiempo a “y” sucesivamente hasta aparecer al menos una entrada de control. Una vez explicita la o las salidas de control la derivada de más alto orden será entonces el grado relativo del sistema con lo que debe cumplirse que

$$y_r = L_f{}_r h(q) + L_g L_f{}_{r-1} h(q) u \quad (6)$$

Con

$$L_g L_f{}_{r-1} h(q) \neq 0 \quad (7)$$

Al no depender las r-1 primeras derivadas temporales del control u se debe cumplir

$$y = h(q) \quad (8)$$
$$\dot{y} = L_f h(q) \quad (9)$$

Cumpliéndose también

$$L_g h(q)=0 \quad (10)$$
$$L_g L_f h(q)=0 \quad (11)$$
$$\vdots$$
$$L_g L_f{}_{r-2} h(q) = 0 \quad (12)$$

La silla inteligente cuenta con sensores que captan el entorno, equipos informáticos que procesan la información y junto con el reconocimiento de voz, permiten al usuario ordenar verbalmente el lugar al que se desea ir y la silla ejecuta autónomamente los movimientos suficientes para llegar al punto deseado. Este artículo servirá como referencia para este proyecto ya que se cuenta con dos características similares: la interacción hombre máquina mediante comandos de voz y el guiado autónomo de la silla. En la cual el diseño en el que se basan reduce en un 50% el precio de sillas motorizadas comerciales además de conservar sus funcionalidades clásicas: plegable y tracción manual.

Como resultado se obtuvo que el sistema de control, los sensores y la alarma sonora funcionan correctamente y dentro de las expectativas. Aunque este artículo de la Universidad de Entre Ríos utiliza un control manual mediante un Joystick, para el proyecto se tomara en cuenta ya que nos servirá como base para uno de los objetivos: la motorización de una silla de ruedas convencional. Como referencia nacional se encontró un artículo científico que lleva el título “Control de una silla de ruedas por medio de un dispositivo móvil con Sistema Operativo Android.”. En el que José Mancilla, Héctor Crespo y Néstor Morales, estudiantes y docentes del Instituto Tecnológico de Tuxtla Gutiérrez presentan un dispositivo electrónico adaptado a una silla de ruedas para motorizarla y controlarla por medio de un dispositivo móvil con SO Android. Como resultado se tuvo que los motores de corriente continua respondieron correctamente a la manipulación de la interfaz gráfica del dispositivo con SO Android. De esta investigación se rescatará la parte de la creación de la interfaz gráfica para dispositivos con SO Android además de la metodología que se utilizó para la problemática que tienen las personas con discapacidades motoras para su desplazamiento.

Prototipo

Debido a que tenemos como finalidad un prototipo experimental se buscó la forma de construir una silla de ruedas de bajo costo que fuera acorde con las necesidades del proyecto y con los materiales con los que cuenta para su realización. Entre los materiales que se utilizaron se tiene tubería de PVC (cedula 80), accesorios de PVC (conexiónes 90 y T cedula 40), triplay 18mm, tornillería y maquinas herramienta como taladro y atornillador. Mediante el uso de software de diseño CAD se diseñó un prototipo virtual (Figura 4), para obtener así la cantidad de materiales necesarios y para realizar los cortes y ensambles a la hora de construir físicamente el prototipo.

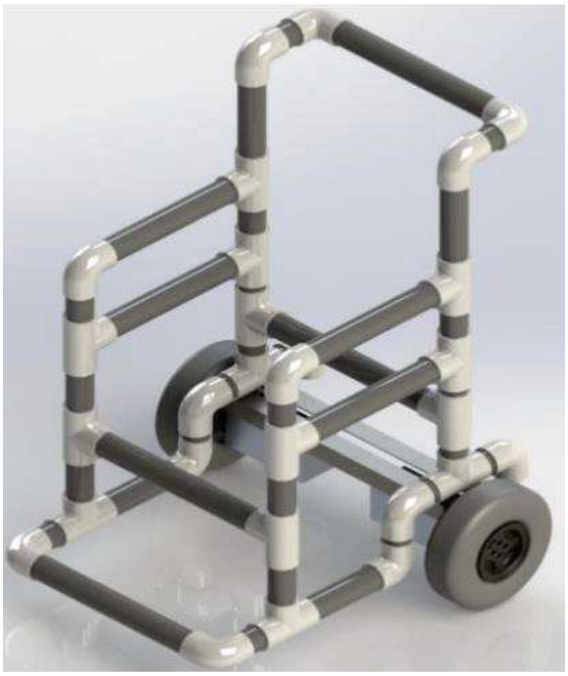


Figura 4. Prototipo digital de vehículo eléctrico.

La instrumentación de la silla se divide en 5 partes: alimentación, actuadores, sensores, controlador y comunicación.

La parte de energía se divide en 2, alimentación para la parte electrónica y alimentación para los motores; para la parte electrónica se usa una batería LIPO de 11.1V a 2200 mAh y para los motores 2 baterías conectadas en serie de 12V a 4 Ah, que dan como resultado 24V a 4Ah.

Como actuadores tenemos 2 motor reductores de 20 Kgcm que funcionan con 24volts a 1 A. para carga máxima. Debido a que no se tiene su tabla de características, estas se desconocen y solo se tienen datos aproximados de ellos.

Es el dispositivo central del sistema, recibe señales de los sensores y de los dispositivos secundarios para ejecutar órdenes a los actuadores del sistema, además de comunicarse con la interfaz de usuario.

Es la placa donde se conectan los sensores y actuadores del sistema, así como las placas de desarrollo.

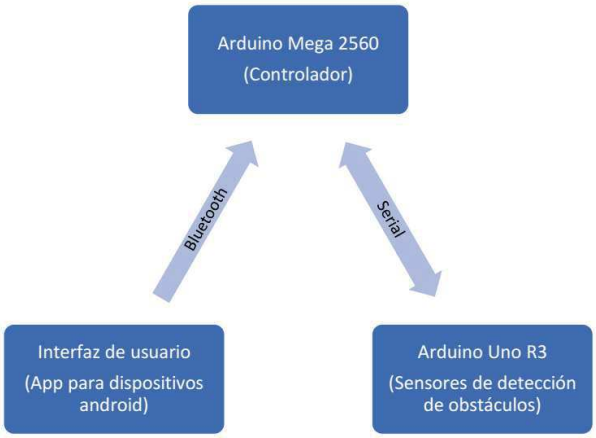


Figura 5. Esquema de comunicación del sistema.

Modelado cinemático

Las ecuaciones de movimiento de un robot diferencial han sido un tema bastante estudiado a lo largo de los años, en base a las ecuaciones 4, 5 y 6 en el apartado del fundamento teórico [1], muestra que las ecuaciones de movimiento son las siguientes:

$$\dot{x} = v * \cos(\theta) \tag{13}$$

$$\dot{y} = v * sen(\theta) \tag{14}$$

$$\dot{\theta} = w \tag{15}$$

Aún con estas ecuaciones, se necesita desglosar la velocidad *v* por lo que se propone las siguientes ecuaciones de movimiento para un robot móvil diferencial:

$$\dot{x} = \frac{r(w_d + w_i)}{2} * \cos(\varphi) \tag{16}$$

$$\dot{y} = \frac{r(w_d + w_i)}{2} * sen(\varphi) \tag{18}$$

$$\dot{\varphi} = \frac{r(w_d - w_i)}{2l} \tag{19}$$

Propuesta de un controlador para él robot móvil diferencial.

Una vez obtenidas las ecuaciones de movimiento de un robot móvil tracción diferencial el siguiente paso es proponer una técnica de control que permita llevar a cabo el seguimiento de trayectorias deseadas por el usuario, ver ecuación 20.

$$\frac{d}{dt}\begin{bmatrix} I_a \\ \omega \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k}{L_a} \\ \frac{k}{J} & -\frac{B}{J} \end{bmatrix}\begin{bmatrix} I_a \\ \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix} u \tag{20}$$

La ley de control está dada por

$$u = \left[\frac{JL_a}{k}\right] u_{aux} + \left[\frac{BL_a + JR_a}{k}\right]\dot{F} + \left[\frac{BR_a}{k} + k\right] F \tag{21}$$

$$u_{aux} = \ddot{F}_d - \gamma_2\left(\dot{F} - \dot{F}_d\right) - \gamma_1(F - F_d) - \gamma_0 \int (F - F_d) \tag{22}$$

Resultados

En cuanto a la instrumentación de la silla, se instalaron los dispositivos suficientes para realizar pruebas experimentales necesarias para poder definir los resultados obtenidos (Ilustración 8 Prototipo físico instrumentado.); uno de los puntos a destacar, es que el prototipo únicamente funcionó sin tripulante, debido a que el torque máximo que otorgan los motores no soportaría tal fuerza, por esa razón, el joystick de mando fue colocado en la parte trasera de arriba de la silla como se ve en la figura 9, para facilitar las pruebas del prototipo

La detección de obstáculos realizada desde el Arduino Uno R3 con el uso de sensores ultrasónicos, los detecta ya sea estáticos o en movimiento, a una distancia promedio de 1.5m, enviando una señal al dispositivo central (Arduino MEGA 2560) el cual se encarga de detener los motores antes de impactarse con el obstáculo, reanudando su movimiento al recibir la señal de que el camino está libre, cambiar de trayectoria o cambiar de mando autónomo a manual. Cabe mencionar que la detección de obstáculos funciona únicamente en modo autónomo.

Un punto bastante importante para la seguridad y el confort del usuario, es el arranque y el paro suave de ésta, en el que se obtuvieron buenos resultados, a continuación, se muestran dos gráficas de la simulación en software del método basado en el seguimiento de una trayectoria de velocidad utilizando una curva de Bézier de 3er grado (Figura 6 y Figura 7):

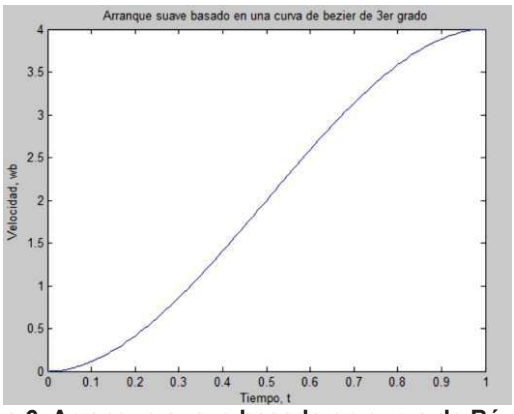


Figura 6. Arranque suave basado en curva de Bézier.

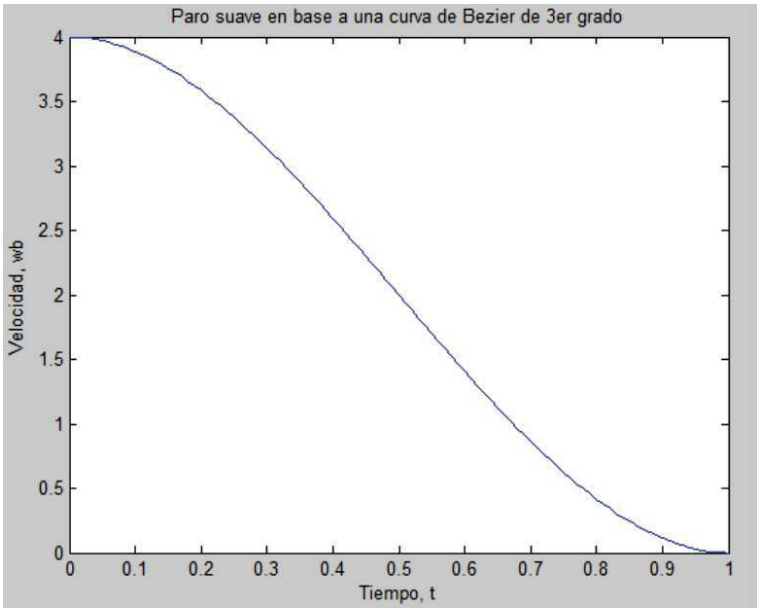


Figura 7. Paro suave basado en curva de Bezier.

El controlador diseñado para este sistema tiene un alto grado de complejidad para ser ejecutado en microcontroladores de gama baja y probablemente media, y al no contar con dispositivos de mejor calidad, se imposibilitó la implementación de este controlador en la silla de ruedas, pero, se obtuvieron buenos resultados en simulación en el software MatLab Simulink. Para trabajos futuros, se recomienda la utilización de microcontroladores más potentes para implementar el control propuesto, con el que seguramente se obtendrán mejores resultados.

Mediante un sistema de control a lazo cerrado, se produjo una implementación básica para el seguimiento de trayectorias (en este caso, una línea física), agregando un controlador PD buscando una mayor fiabilidad y seguridad para un seguimiento de línea suave y preciso, obteniendo resultados bastante buenos, en el que el usuario ordena el destino al cual desea ir (punto B o punto C) y la silla de forma autónoma se traslada hasta cumplir la orden, controlando las velocidades de los motores sin que el usuario intervenga directamente.

La aplicación diseñada, ver Figura 8, para dispositivos con SO Android fue desarrollada de manera simple de tal forma que sea suficiente para esta etapa experimental del proyecto, pero no resulta una aplicación bastante intuitiva y eficaz si lo que se busca es llevarlo al público o comercializar.

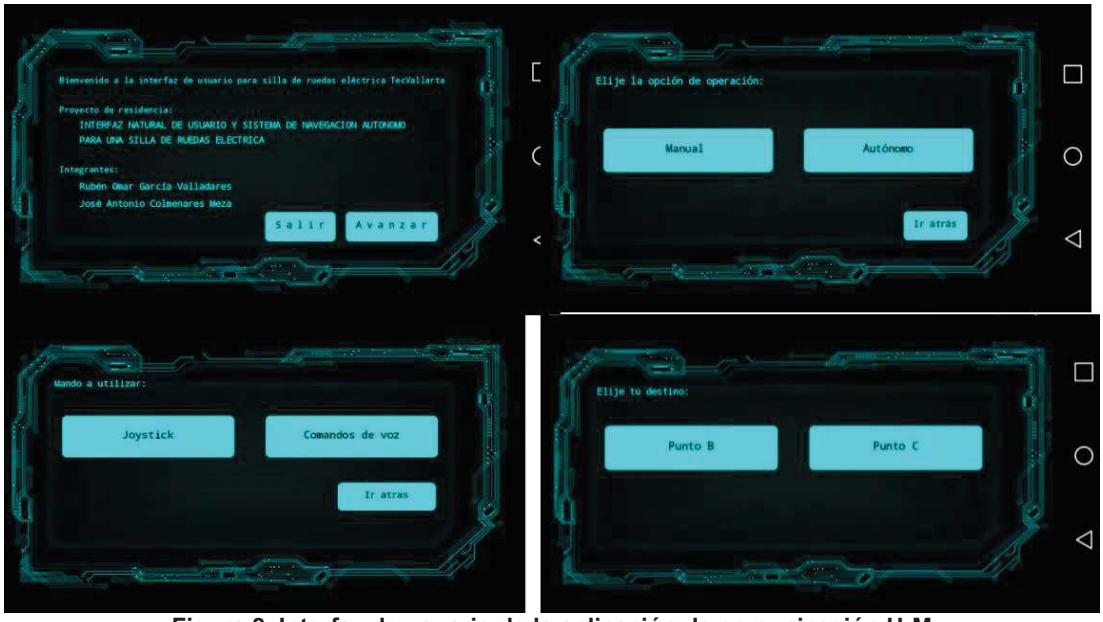


Figura 8. Interfaz de usuario de la aplicación de comunicación H-M

La comunicación entre la aplicación y el dispositivo central se logró con bastante fiabilidad y con seguridad, enviando con éxito los datos necesarios para que la silla realice sus movimientos en función de lo que el usuario ordene desde la aplicación.

El prototipo queda funcionando y a disposición de la academia de electromecánica para trabajos futuros, ,con el fin de conseguir mejoras en los aspectos estéticos y funcionales; tales como los métodos de control, alimentación de la parte electrónica y de potencia de la silla, las formas de interacción usuario – máquina, etc.

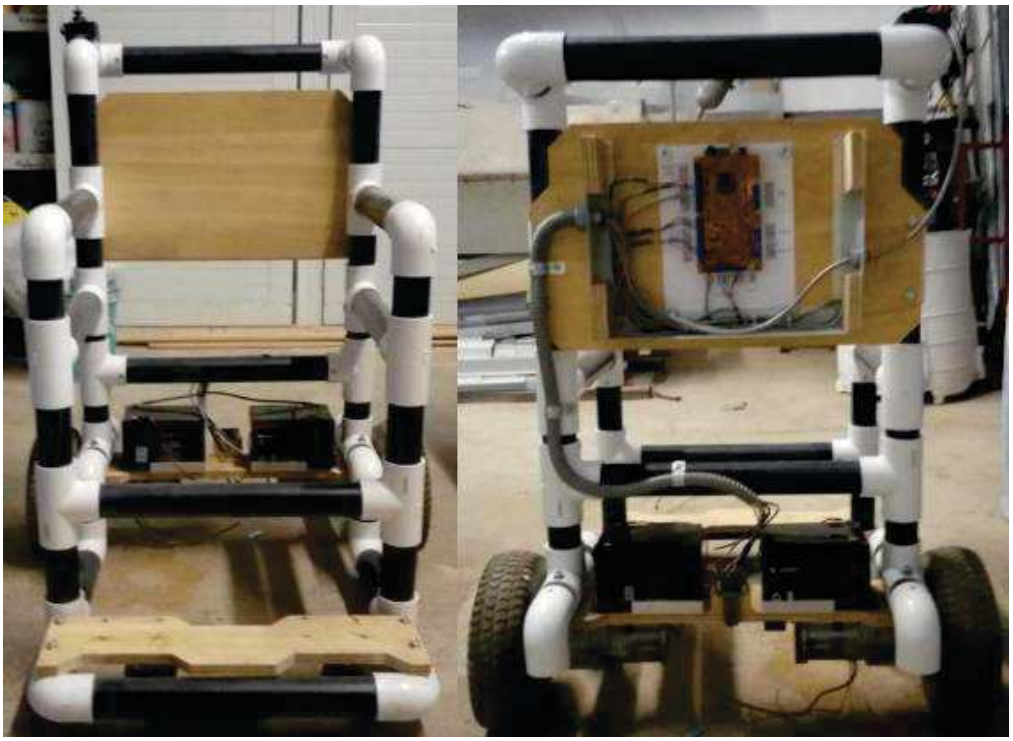


Figura 9. Prototipo de real vehículo eléctrico.

Referencias

[1] E. Aguilar, *Plataforma para la interacción con dispositivos de escelografía.,* Ciudad Real, Castilla, 2013.

[2] M. J. M. L. M. L. Alcubierre J, *Silla de ruedas inteligente controlada por voz,* Zaragoza, 2005.

[3] B. O, *Robotica manipuladores y robots moviles.,* 2001.

[4] C. J. C. A. Brutti C, *Motorizacion de silla de ruedas convencionales,* 2004.

[5] C. A, *Planeacion de trayectorias y control por planitud diferencial,* Cuernavaca: CENIDET, 2013.

[6] G. J, *Sistema de posicionamiento y autolocalización para sillas de ruedas autonomas,* Alcala, Alcala , 2001.

[7] K. O, *Ingeniería de control moderno,* 1998.

[8] M. J. A, *Control de una silla de ruedas por medio de un dispositivo con sistema operativo,* 2012.

[9] M. R. Ramos E., *Modelado, simulación y construcción de un robot movil de ruedas tipo diferencia.,* 2012.

[10] R. E. Sanfelio, *Tecnicas de detección de obstaculos y seguimiento de personas usando fusion Lidar y otros sensores.,* Catalunya, Catalunya, 2013.

[11] S. C., *Interfaz Ntural de usuario,* ResearchGate 1, 2015.

[12] T. G, *Research on Autonomous Robotic Wheelchairs in Europe,* 2001.

Ingeniería y Automatización, Capítulo 9, pp. 111 - 120.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Automatización de Grippers del Laboratorio de Robótica

López Luna Alma Rosa, Leon Martínez Martin Francisco, Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen y Platt Borbón Martina Elisa

Universidad de Sonora, División de Ciencias Exactas y Naturales, Departamento de Investigación en Física y División de Ingeniería y Departamento de Ingeniería Industrial.
leonm.94@gmail.com

Resumen

Se realizó un proyecto con el fin de poder controlar el movimiento de los grippers que se encuentran en los robots del laboratorio de robótica en la Universidad de Sonora mediante el desarrollo de una página Web. Esto con el fin de tener otro tipo de control aparte del accionamiento manual y por PLC. Este control se realizó con el dispositivo RaspberryPi3, con el objetivo de que el control sea inalámbrico y cualquier dispositivo con capacidades web pueda acceder al control de los grippers. Como medida de seguridad se le adicionó como requisito control de usuario con contraseña.

Palabras clave: Automatización, Grippers, Raspberry, WebIOPi.

1. Introducción

La Raspberry Pi es una minicomputadora que fue creada específicamente para facilitar el aprendizaje tecnológico. Tiene muchos componentes para proyectos basados en computadora, como puertos USB, un puerto Ethernet, una ranura para tarjeta SD, puertos de antena WiFi y más.

En [1] se aprovechan las capacidades de la Raspberry Pi para elaborar un proyecto domótico, en este proyecto se aprovecharon sus capacidades de acceso y control para el desarrollo de una interfaz web basada en WebIOPi para poder manejar la apertura y el cerrado de los grippers de dos robots industriales de 6 grados de libertad en el laboratorio de Robótica de la Universidad de Sonora de una manera más sencilla mediante el diseño de una página web.

Se inicio por realizar las configuraciones necesarias a nuestra Raspberry Pi como se ve en [2], junto con las conexiones entre nuestra Raspberry Pi y las electroválvulas que controlan a los grippers que fueron conectadas mediante un módulo de relevadores de canales independientes protegidos para asegurar la integridad de nuestros dispositivos. Se realizaron pruebas de accionamiento manual muy sencillas como configurar los pines a utilizar y manualmente mediante comandos colocarlos en estados altos o bajos para asegurar su funcionalidad.

Una vez concluida la etapa de pruebas se procedió a la configuración de nuestra página web para el accionamiento de los grippers mediante botones de tipo “press and hold” junto con la configuración de usuario y contraseña.

Finalmente se logró el diseño e instalación de la interfaz y se procedió a la mejora estética de la página web.

2. Desarrollo

2.1 Configuración inicial.

1. Se instaló el sistema operativo a la Raspberry Pi, se optó por Raspbian Jesse por su versatilidad y simplicidad de uso.

2. Se actualizaron los componentes de la Raspberry mediante el uso de los comandos*:

•sudo apt-get update
•sudo apt-get upgrade
•sudo reboot (Reinicia la Raspberry Pi para que hagan efecto las actualizaciones)

*Todos los comandos son introducidos individualmente en la Terminal de Raspbian.

3. Se instaló WebIOPi [Ptak, 2018] que es la que nos ayudará a manejar las comunicaciones entre la página web y la Raspberry Pi. Para esto se descarga el comprimido del WebIOPi introduciendo el siguiente comando en la terminal:

•wget http://sourceforge.net/projects/webiopi/files/WebIOPi-0.7.1.tar.gz

Cuando la descarga esté hecha se procede a descomprimirlo:
•tar xvzf WebIOPi-0.7.1.tar.gz
•cd WebIOPi-0.7.1/

Se procede a instalar un patch de la WebIOPi.
•wget https://raw.githubusercontent.com/doublebind/raspi/master/webiopi-pi2bplus.patch
•patch -p1 -i webiopi-pi2bplus.patch

Ejecutamos la instalación de la configuración de la WebIOPi
•sudo ./setup.sh

Se reinicia la Raspberry Pi
•sudo reboot

Una vez instalado se procedió a realizar las pruebas necesarias para asegurar el funcionamiento de nuestro nuevo servidor.

2.2 Configuración de la aplicación.

Una vez concluida las pruebas se empezó la elaboración de la aplicación web para nuestra Raspberry Pi, primero comenzamos con la creación de la carpeta la cual contendrá nuestra página web y después las subcarpetas que le darán la funcionalidad y lo estético a esta.

WebIOPI [3] nos permite comunicar la pagina web con la Raspberyy Pi de una manera sencilla mediante el uso de librerias.

A continuación, una breve explicación de lo que hace el código de Javascript en la carpeta de Scripts.

webiopi().ready(function(): Esto instruye a nuestro sistema para crear esta función y ejecutarla cuando el webiopi esté listo.

webiopi().setFunction(17,"out"); Esto nos ayuda a decirle al servicio WebIOPi que configure GPIO17 como salida. Tenemos ocho botones aquí, se podrían agregar o quitar botones dependiendo la aplicación.

webiopi().digitalWrite(17,1); Esto indica que al momento de inicializar los botones se pongan en un estado alto, todo esto debido al funcionamiento por lógica inversa de los relés.

var content, button; Instruye a nuestro sistema a crear una variable de nombre content y button.

content = $("#content"); La variable content se usará en nuestro html y css. Entonces, cuando nos referimos a #content, el framework de WebIOPi creara todo lo relacionado con él.

button = webiopi().createButton("relay 1", "R1-cerrar", mousedown1, mouseup1); WebIOPi puede crear diferentes tipos de botones. El fragmento de código anterior nos ayuda a decirle al servicio WebIOPi que cree un botón de tipo press and hold con la etiqueta relay 1 y el nombre “R1-cerrar” que ejecuta una función cuando el botón se presiona (mousedown1) y otra para cuando no (mouseup1). En este caso estamos indicando que al presionarlo se ponga en estado bajo (0) para activar el relé y al no presionarlo se ponga en estado alto (1) para desactivar el relé.

content.append(button); Agregue este código a cualquier otro código para el botón creado en el archivo html o en otro lugar. Se pueden crear más botones y todos tendrán las mismas propiedades que este botón.

function mousedown1(){webiopi().digitalWrite(5,0);}: Función que se manda llamar al momento de presionar el botón.

function mouseup1(){webiopi().digitalWrite(5,1);}: Función que se manda llamar al no presionar el botón.

Una vez concluida con la configuración de la funcionalidad de nuestra aplicación se procedió a editar los parámetros estéticos de esta mediante la implementación de código CSS dentro de la carpeta con el respectivo nombre. A continuación, una breve descripción de lo que realizamos.

Establecemos el color de fondo como #ffffff que es blanco, posteriormente agregamos una imagen de fondo ubicada en la carpeta de imágenes de la Raspberry asignada a nuestro proyecto, nos aseguramos de que la imagen no se repite estableciendo la propiedad background-repeat en no-repeat, y de igual manera el código indica al CSS que centralice el fondo. Se establecen también el tamaño de fondo, la fuente y el color. Después la configuración de los botones para que sean más estéticos y para que cambien sus visuales dependiendo del estado.

Finalmente procedemos a anexar todo mediante la realización del index en código HTML, es lo que une todos los elementos de la página, este código manda a llamar las variables content creadas previamente. Habilita que nuestra página web tenga una versión móvil. De igual manera manda llamar tanto los scripts del JavaScript, como los visuales del código CSS. Da las configuraciones finales de los visuales. Inserta el nombre y título de la página y el escudo de la universidad.

2.3 Instalación eléctrica y electrónica.

Una vez concluido el desarrollo de la aplicación web y realizadas las pruebas correspondientes se procedió a integrar el proyecto.

Se configuraron algunos puertos GPIO de nuestra Raspberry como salidas y se conectaron a las entradas del modulo de relevadores como se puede ver en las figuras 1 y 2.

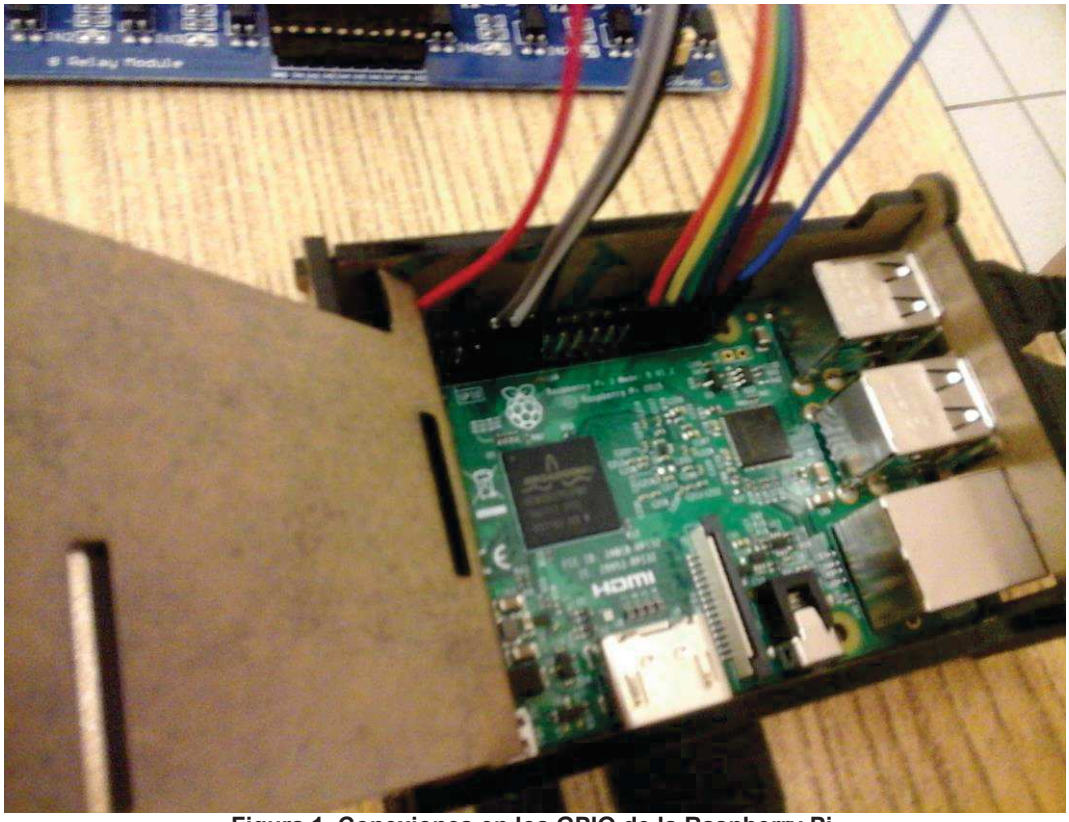


Figura 1. Conexiones en los GPIO de la Raspberry Pi.

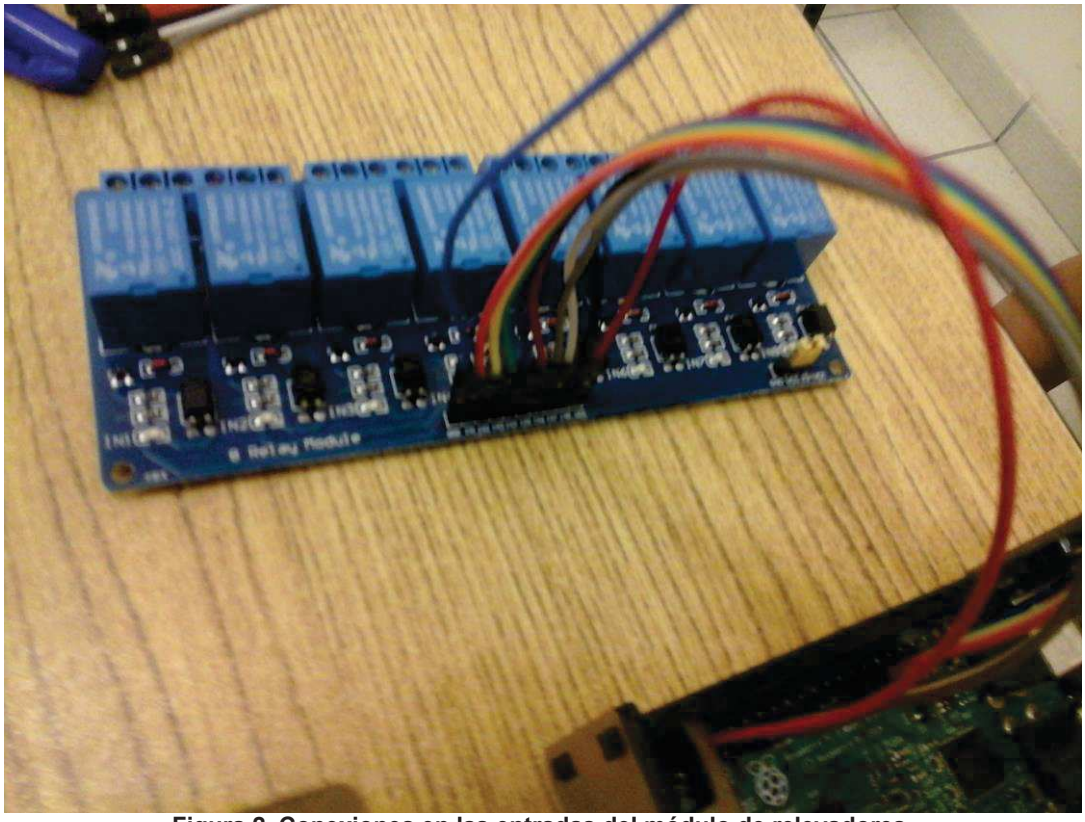


Figura 2. Conexiones en las entradas del módulo de relevadores.

Ingeniería y Automatización, Capítulo 9, pp. 111 - 120.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Una vez realizadas estas conexiones se procedió a realizar las conexiones de las salidas en el modulo de relevadores hacia cada electroválvula como se puede apreciar en la imagen 3.

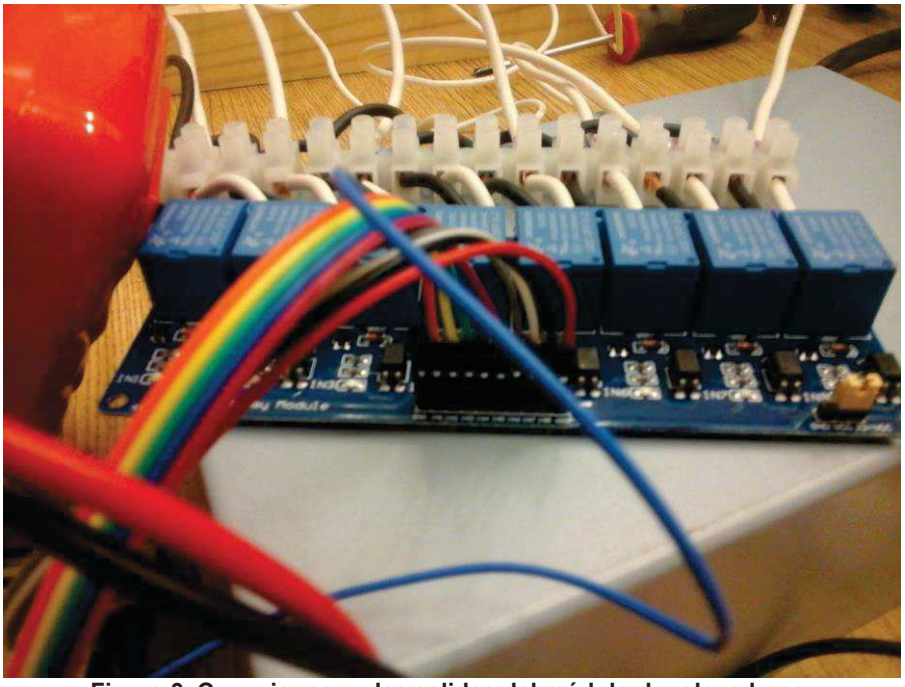


Figura 3. Conexiones en las salidas del módulo de relevadores.

3. Resultados

En la figura 4 se puede observar una captura de pantalla de como se ve actualmente la página web al acceder desde una computadora. Se observan los 8 botones que controlan los dispositivos.

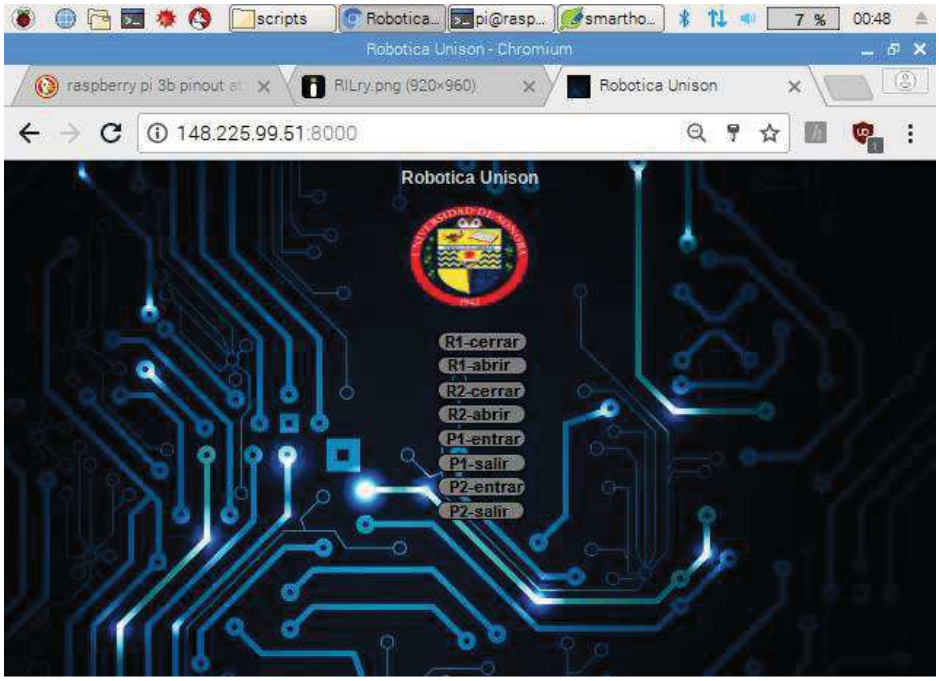


Figura 4. Ilustración de página Web para el control de los gripper.

Configuración final WebIOPi

Una vez terminada nuestra página web y comprobada su funcionalidad se edita la configuración inicial del WebIOPi para que esta mande llamar a nuestra página web en vez de la predeterminada.

Es posible tanto cambiar la configuración de usuario y contraseña como quitarlas por completo.

- sudo webiopi-passwd

La página web se pone en funcionamiento y se termina con códigos específicos.

Para consultar el estado de conexión.

- sudo /etc/init.d/webiopi status

Se puede configurar la Raspberry Pi para que siempre que se encienda se inicialice automáticamente nuestra página web.

- sudo update-rc.d webiopi defaults

Y para hacer que deje de iniciarse.

- sudo update-rc.d webiopi remove

Finalmente, para acceder a la página, es necesario tener encendida la Raspberry. Dependiendo de la conexión a internet que tenga la Raspberry; ya sea con Wifi o directamente de un Ethernet, es necesario acceder a la terminal de la Raspberry de la siguiente manera:

- sudo ifconfig

Aparecerá el IP actual de la Raspberry el cual se introduce a la barra de direcciones exactamente como aparezca, por ejemplo, como se muestra en la figura 4:

- 148.225.99.51:8000

(Observe que se añadió un: 8000, el puerto que está configurado en la Raspberry)

E introduciendo el usuario y contraseña que se empleo o si bien el default, se obtiene acceso a la aplicación web.

En la figura 5 podemos observar el equipo utilizado: Raspberry y relevadores, que convierten 5 volts corriente directa a 110 volts corriente alterna, para accionar las electroválvulas neumáticas que a su vez salen o entran los pistones para hacer que abran o cierren sus efectores finales (pinzas), todo esto para que desde cualquier dispositivo móvil se puedan controlar los gripper, ya sea con un teléfono celular, Tablet o computadora portátil vía WiFi. También se puede tener un control

directamente en la Raspberry con un Monitor HDMI, teclado y mouse USB. O bien, con equipo de cómputo conectado por Ethernet cableado a la red local.

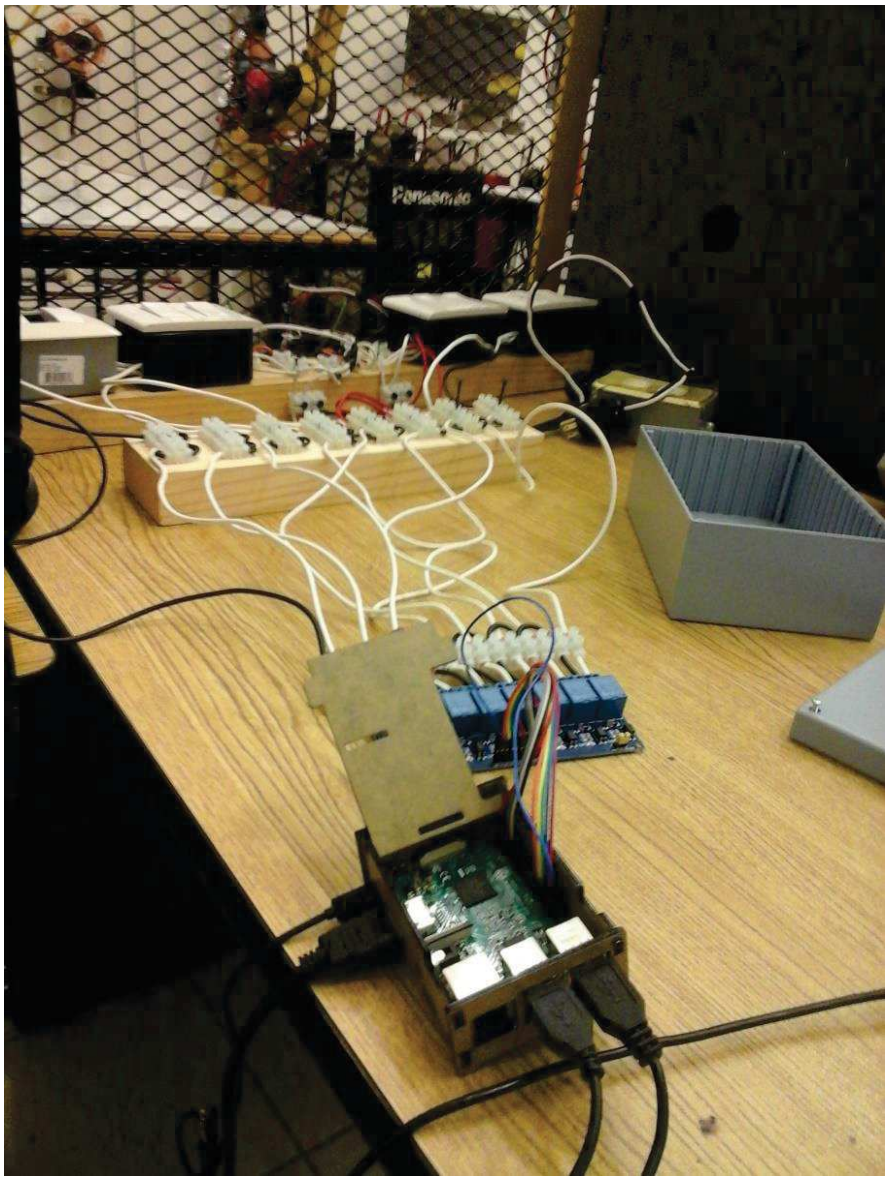


Figura 5: Fotografía del equipo utilizado.

Una vez concluidas las intalaciones y que se realizaron mas pruebas sobre los equipos, se procedió a colocar los componentes dentro de un gabinete procurando colocarlos de una manera óptima como se muestra en la figura 6.

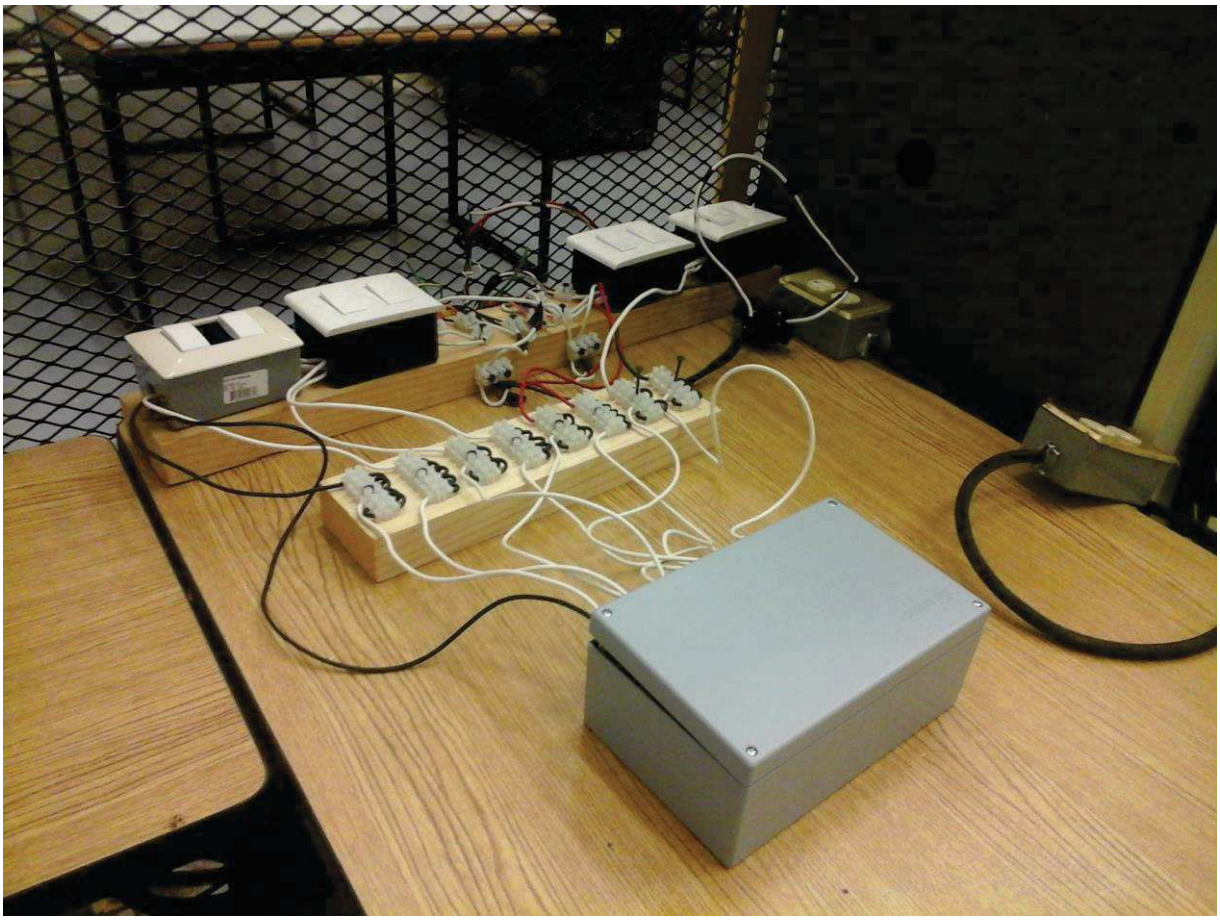


Figura 6. Gabinete con la raspberry y Relevadores.

En la figura 6 podemos percibir el Robot 1 con su gripper sujetando un objeto.

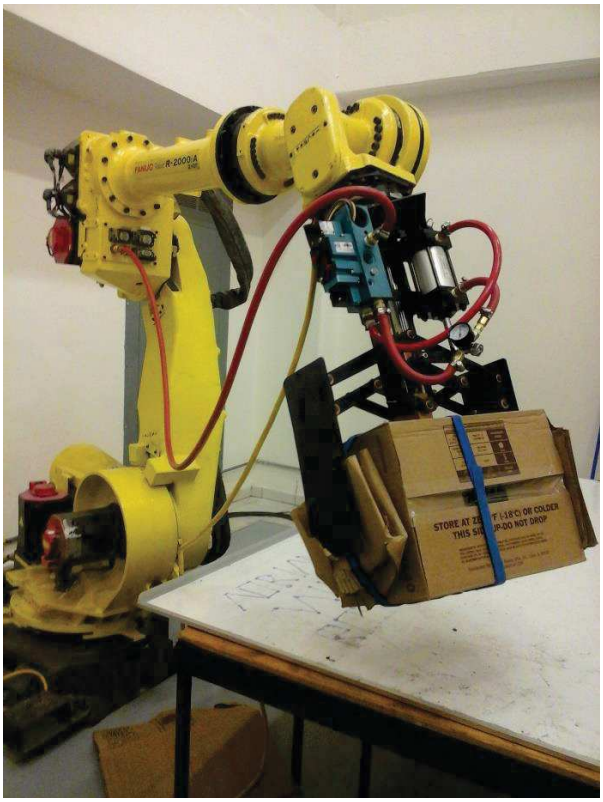


Figura 6. Fotografía del Robot 1 con gripper.

En la figura 7 podemos apreciar el Robot 2 con su gripper sujetando un objeto.

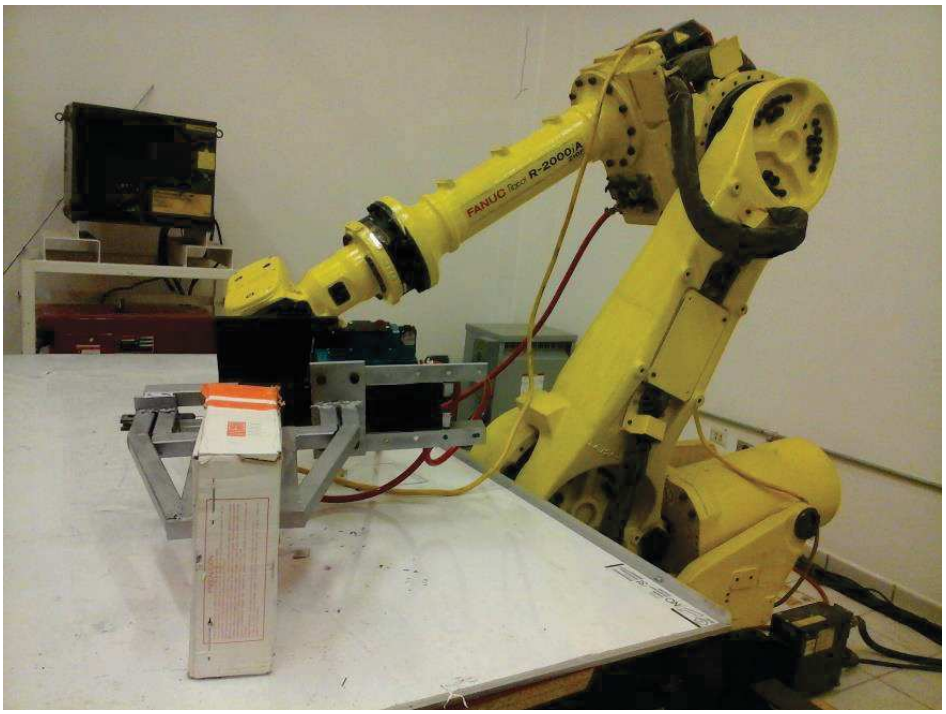


Figura 7. Fotografía del Robot 2 con gripper.

4. Conclusiones

La elaboración del proyecto permitió ampliar el conocimiento entorno al desarrollo de aplicaciones web mediante la implementación Jquery, códigos JavaScript, HTML y CSS, junto con los conocimientos básicos necesarios de Linux para poner en funcionamiento tanto la Raspberry Pi configurada con Raspbian como el WebIOPi. Los conocimientos adquiridos con este proyecto son de amplia importancia ya que son aplicables a la industria y es muy fácil escalar el proyecto junto con las posibles aplicaciones desarrolladas en un entorno del Internet de las Cosas (IoT).

Las capacidades para agregar más componentes se aprovecharon a posterioridad para incluir aplicaciones de domótica y control inteligente en el laboratorio de Robótica de la Universidad de Sonora. De esta manera se obtuvo control automático e inalámbrico de la iluminación, monitoreo de temperatura, gas, actividad en el laboratorio, control de entradas y salidas. La Raspberry Pi nos da un sinfín de opciones que escoger de acuerdo con nuestras necesidades y presupuesto junto con la dirección que le queramos dar.

Posibles trabajos futuros que podrían nacer del proyecto antes mencionado podrían ser mejorar la ciberseguridad de la aplicación, ya que, si bien el IoT tiene muchas ventajas, una desventaja de este es que al tener todo conectado a la red, uno se vuelve más vulnerable a los ataques cibernéticos por lo que se buscaría volver mas seguros todos los dispositivos conectados.

Otras opciones para trabajar a futuro podrían ser la inclusión de Big Data e inteligencia artificial (IA) y/o machine learning como dispositivos controladores, de esta manera el dispositivo al estar obteniendo un flujo constante de información que se va a ir almacenando, podría programársele algoritmos que le enseñen a aprender a tomar decisiones prediciendo el comportamiento de los usuarios, de esta manera no solo se vuelve mas eficiente el control del laboratorio y/o salones si no que puede representar ahorros considerables en energía a la Universidad, al por ejemplo controlar el comportamiento de los aires acondicionados, computadoras e iluminaria.

También podemos enfocarnos en un entorno mas industrial y enfocarnos en el Internet de las cosas Industrial (IIoT) o en la industria 4.0. Que ambas, toman e incluyen aspectos del Big Data, machine learning y la IA. De esta manera al agregar gran cantidad de sensores a las maquinas, conectarlos a la red e ir almacenando la información de estos, se pueden obtener muchos aspectos importantes dentro de una fábrica, como lo son la eficiencia, como va la producción, controles estadísticos, monitoreo del funcionamiento de maquinaria para predecir el mejor tiempo para darle mantenimiento o detectar cuando está fallando, las maquinas pueden detectar cuanto queda de materia prima (Control del almacén) y realizar pedidos antes de que esta se agote para no tener tiempos muertos.

En conclusión podemos decir que el desarrollo de aplicaciones web y la interconexión entre dispositivos nos brinda un sinfín de opciones de donde elegir, junto con nuevos retos a enfrentar.

Referencias

[1] Odunlade, E. (2017). IoT Based Raspberry Pi Home Automation Project. Retrieved June 14, 2018, from https://circuitdigest.com/microcontroller-projects/iot-Raspberry-pi-home-automation
[2] Perles, A. Empezar con la Raspberry Pi. Universidad Politécnica de Valencia. 2017
[3] PTAK, E. (n.d.). CSS, Installation, Javascript Retrieved June 14, 2018, from http://webiopi.trouch.com/CSS.html,
http://webiopi.trouch.com/INSTALL.html
http://webiopi.trouch.com/JAVASCRIPT.html

Ingeniería y Automatización, Capítulo 10, pp. 121 - 135.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño y Construcción de un Prototipo Gripper Industrial

Amavizca Ramírez Miguel, Verdugo Neudert Oswaldo Alfonso, Castillo Ortega Rafael, Acosta Enríquez Milka del Carmen y Platt Borbón Martina Elisa

Universidad de Sonora, División de Ingeniería, Departamento de Ingeniería Industrial y División de Ciencias Exactas y Naturales, Departamento de Investigación en Física.
miguelamavizca2@gmail.com

Resumen

El objetivo de la investigación, es el diseño y la construcción de un prototipo gripper industrial de bajo costo y reciclar dispositivos ya existentes como los actuadores y las electroválvulas neumáticas, en base a la necesidad de una herramienta de trabajo para un robot industrial y con el propósito de hacerlo didáctico, se elaboró un gripper o garra mecánica, este con el fin de tener una mejor experiencia didáctica en el laboratorio de robótica de la Universidad De Sonora, campus Hermosillo. Se detalla paso a paso cada uno de los procedimientos a seguir para su desarrollo, desde el diseño a través del software de "SolidWorks", los estudios de resistencia contra fuerzas, a través de "SolidWorks SIMULATION" y el maquinado a través de una CNC mediante el software "CAMWorks", logrando el objetivo de diseñar y construir el prototipo, obteniendo un sistema funcional.

Palabras clave: CAMWorks, CNC, Gripper, SolidWorks.

1. Introducción

En la industria la implementación de robots juega un papel muy importante para la asistencia en diversos procesos, y existen cientos de ellos diferentes, desde rotar una pieza, hasta cambiar de posición y ubicación dentro de una línea de producción. Y de este gran avance surge una necesidad inminente, elaborar el dispositivo adecuado para realizar la tarea indicada en la línea de producción, acorde a las dimensiones del producto que se manipulara. El '"gripper" es este dispositivo que da solución a esta necesidad, siendo diseñado especialmente para esta tarea, como por ejemplo el levantamiento de una caja, tubo o artículo irregular, etc. Diseñado a la forma exacta de este, y es ahí donde el trabajo de la ingeniería entra, el diseño y elección de materiales adecuados que puedan soportar la tarea a realizar. Así como también, encontrar los materiales y componentes necesarios acorde a nuestro presupuesto.

A través de este artículo se presenta paso a paso el diseño y construcción de un gripper para el uso de un robot industrial, por parte de estudiantes de la Universidad de Sonora, cada uno de los problemas encontrados a través de su desarrollo y las soluciones para seguir con el trabajo del mismo hasta llegar a un producto terminado y funcional, para elaborarlo se basó en el libro "Gripper in motion" en el cual detalla la funcionalidad de un gripper además de dar sugerencias en el diseño del mismo, así como los tipos de actuadores que se podían usar. [1]

2. Gripper mecánico accionado electroneumáticamente

2.1 Diseño

Este proyecto se creó con el objetivo de que los alumnos de la Universidad de Sonora en la asignatura de Robótica Industrial pudieran utilizar los robots de una manera más eficiente, debido a que en el laboratorio no se contaba con un mecanismo, el cual permitiera realizar alguna actividad en específico, como levantar objetos y moverlos. Se limitaba especialmente a mover los 6 ejes del robot y programarlo. De ahí continuó la búsqueda, para crear un modelo que cubra las necesidades del laboratorio. Para la creación del Gripper se tomó en cuenta el modelo del software Handling PRO de ROBOGUIDE [2] y se adecuó a las necesidades del laboratorio y los materiales que se usarían.
La Figura 1 muestra el softwar**e** Handling PRO de ROBOGUIDE.

Figura 1. Programa FANUC de simulación de robots

Por consiguiente, se dispuso a diseñar un prototipo con la ayuda de SolidWorks [3], y al crearlo con un modelo lo más parecido al de Handling PRO, se decidió disminuir las proporciones, y cambiar un poco el mecanismo, se agregó un actuador neumático y una válvula electroneumática, aquí se tomó la decisión de colocar el pistón por un lado del gripper en lugar del centro, con ello, se redujo la extensión del mismo, logrando desempeñar la misma tarea de levantar objetos. El actuador como la válvula fueron facilitados por la Universidad de Sonora en diferentes tamaños y capacidades, y se optó por elegir el más pequeño, favoreciendo con uno de los objetivos principales, disminuir el tamaño del prototipo. En la Figura 2 podemos apreciar el gripper neumático.

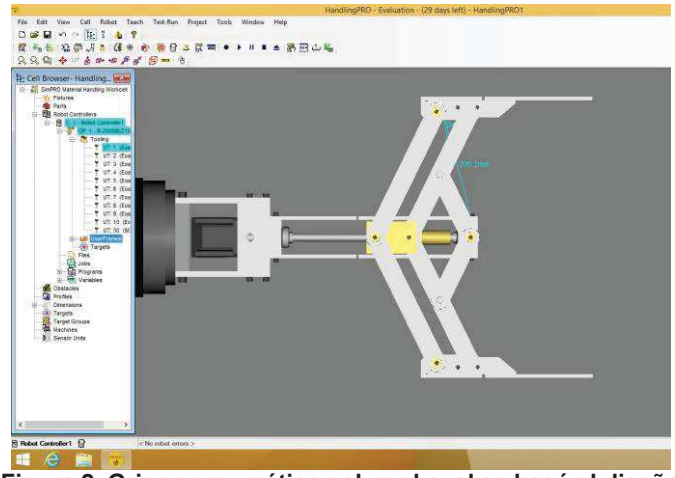


Figura 2. Gripper neumático sobre el cual se basó el diseño

La electroválvula utilizada es una 5/2 y el modelo es: MAC 6333D-511-PM-111DA, esta es de las válvulas más sencillas y se escogió por su facilidad a la hora de accionar, cuenta con salidas una para abrir el gripper y la otra para cerrarlo, esto es accionado mediante dos interruptores de un polo un tiro eléctrico. Un actuador neumático TRD Manufacturing, Inc. de 2 ½” x 2 ½”, S/N 06-09803 (L5), este es accionado mediante la electroválvula anteriormente citada.

En la parte del diseño y después de hacerse algunas pruebas mediante el software de elemento finito, de las cuales se detallará más adelante en este mismo artículo, se determinó que los materiales más óptimos por su costo y funcionalidad serían los siguientes:

- Solera de 1”x0.25” de acero ASTM**-36,** Este para las barras de movimiento.
- Placas de Acero de ¼” de grosor ASTM-A36, estas para el diseño de la base y paletas.
- Balero de 3/8” en diámetro exterior y 1/4” de diámetro interior.
- Correderas de cajón para la transmisión del movimiento lineal del pistón.
- PTR 2” x 3 de ¼ de grosor para las guías centrales.
- Tornillos de 1/4” de diferentes tamaños.
- Tuercas de seguridad.
- Guasas planas.

Tomando en cuenta el modelo de Handling PRO y las adecuaciones que se agregaron, se obtuvo un primer modelo, el cual se muestra en la Figura 3.

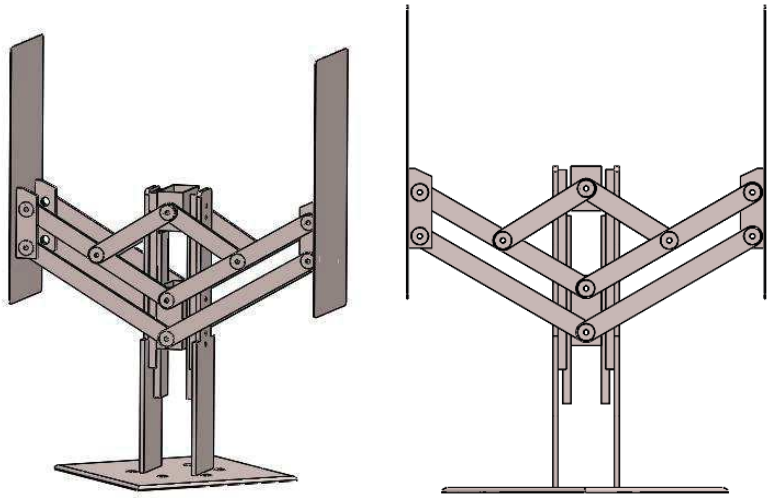


Figura 3. Diseño base del gripper.

2.2 Estudio de elemento finito (utilería incluida en el software SolidWorks)

Se decidió hacer este tipo de estudios, para analizar previamente si este resistiría las fuerzas a las cuales será sometido y gracias a ello detectar puntos de fallo del gripper y puntos en los cuales se puede mejorar el prototipo. Para el estudio de elemento finito, al ser estudios muy complicados y tardados se tomó la decisión de hacer el estudio únicamente sobre el elemento que iba a sufrir más fatiga, primero se hizo un estudio en general rápido, con los esfuerzos que sufrirá normalmente (peso del mismo, peso del objeto a manipular y la fuerza del pistón sobre piezas del mecanismo) y se obtuvo que el elemento que más fatiga tendría son las paletas de agarre, por lo cual se decidió aislar y obtener un resultado más a detallado de la deformación de esta.

Los resultados obtenidos mediante el estudio serán presentados en la sección de resultados. pero de manera general podemos decir que el estudio previo del mismo se obtuvo de forma prematura que el diseño sufriría una deformación a la hora de tomar lo elementos en las paletas, para solucionar esta falla en el diseño se decidió comprar una placa de acero de mayor grosor, esto añadiendo más rigidez a la misma y así evitando una deformación. La Figura 4 muestra las fuerzas que se sometieron al modelo.

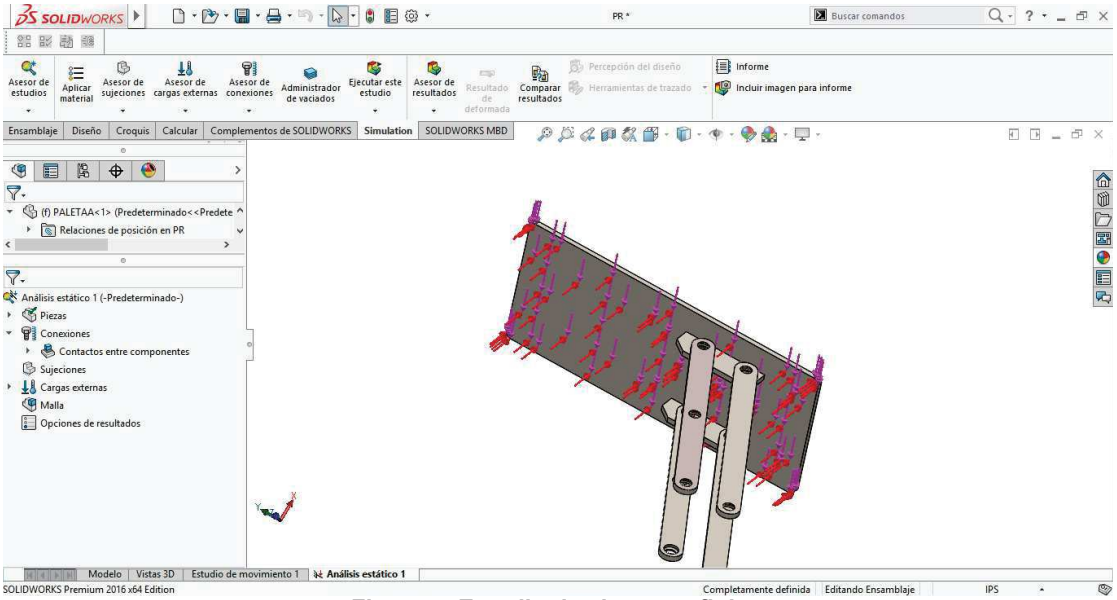


Figura 4. Estudio de elemento finito.

Nuestro gripper está diseñado para levantar elementos ligeros como cajas, por lo tanto, se tomó la decisión de hacer la simulación de levantar un peso de 15 kg, además de eso se consideró el peso del mismo y la presión que ejerce el pistón 35 psi.

2.3 CAMWorks (Utilería que se incluye en SolidWorks)

Para la elaboración de los componentes y para tenerlos con mayor precisión se utilizó el software CAMWorks, ya teniendo el diseño en SolidWorks, se utilizó CAMWorks para hacer las simulaciones de maquinado. El software nos arrojó muy buenos resultados pudiendo elaborar cualquier geometría de una manera más sencilla, siendo la mayoría materiales muy sencillos de maquinar, con la excepción la placa de ¼ de grosor, la cual al ser tan gruesa es necesario hacer muchas pasadas antes de poder hacer los cortes necesarios, lo cual se soluciona con el uso de herramientas adecuadas para la manipulación del acero y una refrigeración óptima. Las figuras 5,6 y 7 dan a conocer las simulaciones realizadas en CAMWorks para posteriormente ser maquinadas.

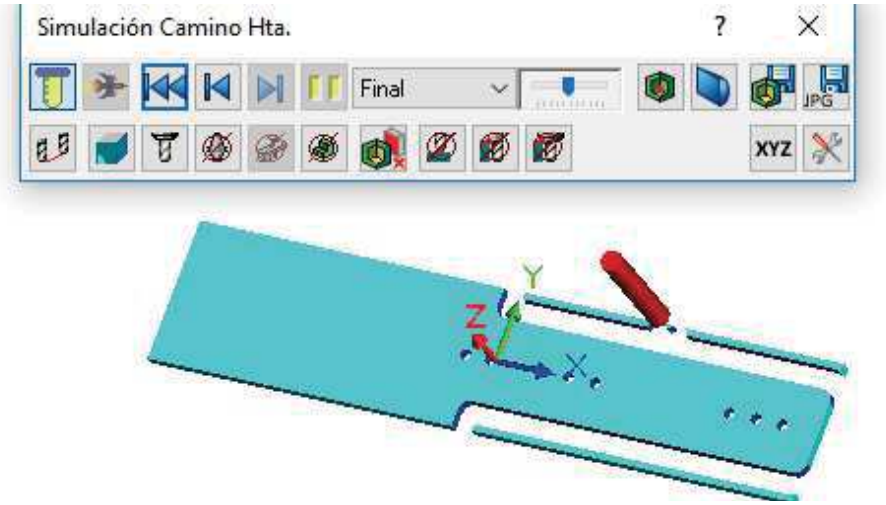


Figura 5. Simulación de CAMWorks de placas laterales.

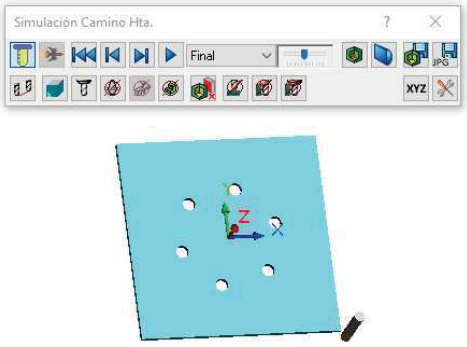


Figura 6. Simulación de CAMWorks de placa base.

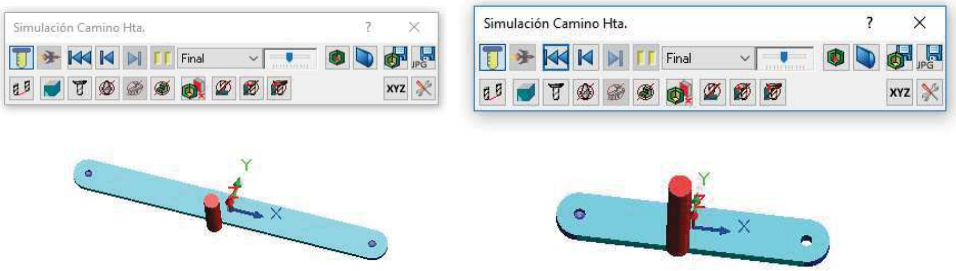


Figura 7. Simulación en CAMWorks de las barras del movimiento.

El equipo para el cual fue diseñado es un “Centro de maquinado CNC milltronics RW15 Centurión 7” [4] el cual cuenta con 3 ejes de movimiento, pudiendo alcanzar hasta las 8000 RPM en el husillo lo cual nos es suficiente para la manipulación de nuestros materiales. Además de esto como el espacio dentro del centro de maquinado es limitado se tienen que hacer cortes previos a los materiales para poder introducir la pieza de stock y maquinaria.

Es importante citar que se consideró hacer los cortes de la placa con un equipo de corte de plasma manual, pero al no contar con las instalaciones no fue factible, aunque hubiera sido la forma más óptima de la elaboración de la misma al tener la potencia de cortar la placa de acero con mayor facilidad, aunque las perforaciones quedan más precisas con la CNC.

2.4 Diseño terminado

Finalmente se llegó a un producto que se puede apreciar en la Figura 8, mostrando la vista isométrica.

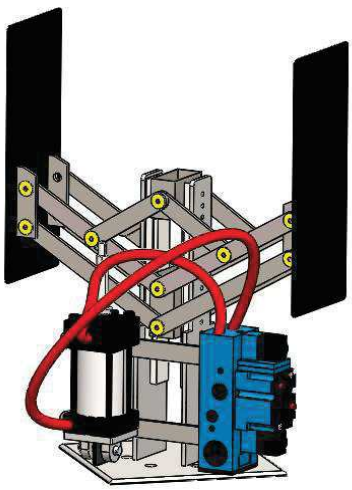


Figura 8. Vista isométrica.

Ingeniería y Automatización, Capítulo 10, pp. 121 - 135.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Las figuras 9,10 y 11 muestran las vistas frontal, lateral y superior respectivamente, dando una mejor idea de la forma y dimensiones del prototipo.

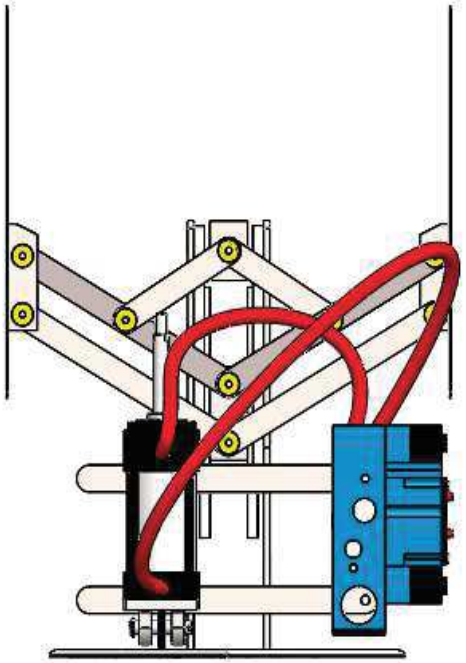


Figura 9. Vista frontal.

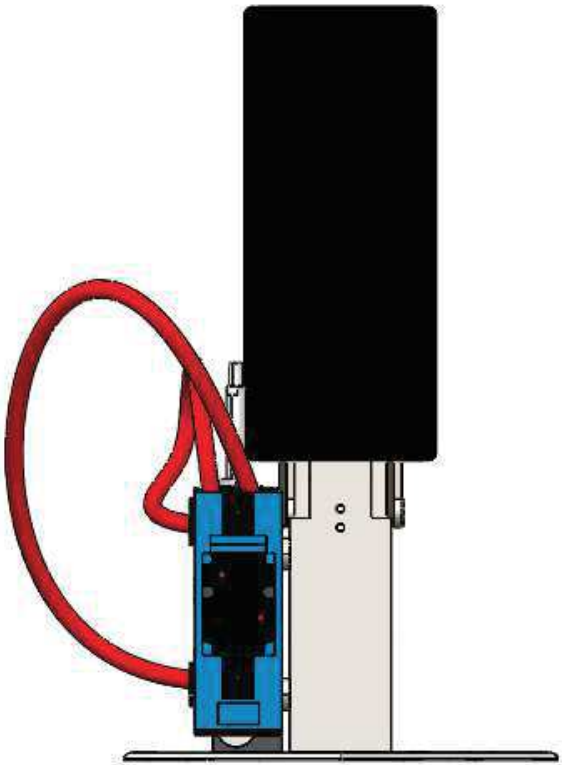


Figura 10. Vista lateral.

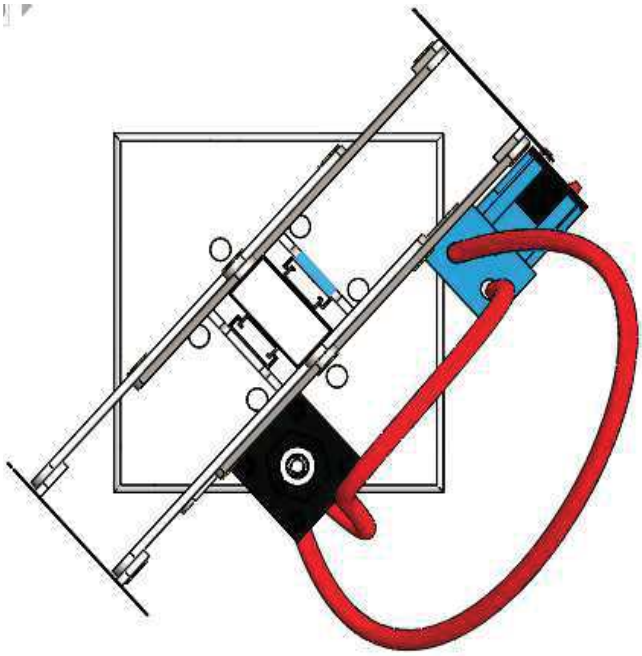


Figura 11. Vista superior

Las propiedades físicas que arrojó SolidWorks se dan a conocer en la Figura 12, en las cuales se muestra la masa, el volumen, el área de superficie, así como también el centro de masa, momento de inercia entre otros. Los resultados que se arrojan son muy semejantes a los reales y el error es casi nulo, lo cual permite tomar acción y llevarlo a un siguiente nivel.

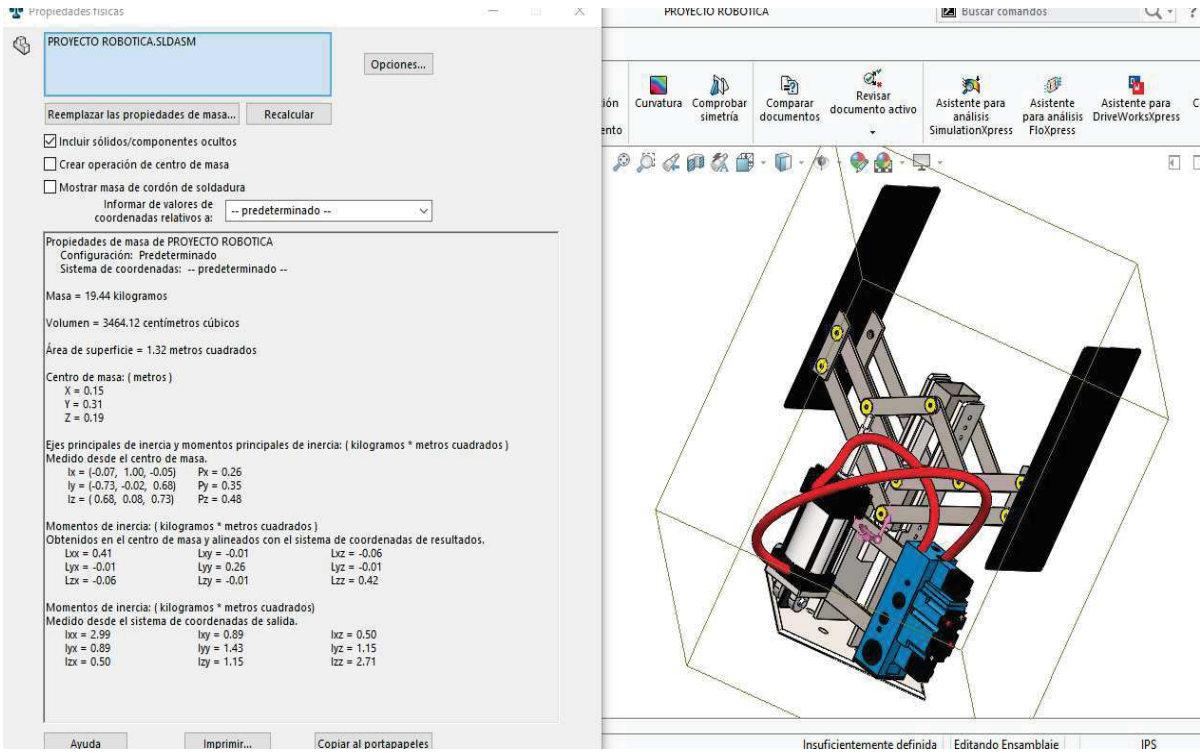


Figura 12. Propiedades físicas.

2.5 Fabricación

Una vez terminado nuestro asistido por computadora, se pasó a hacerlo realidad, desde un principio se pensó utilizar el centro de maquinado CNC para realizar los cortes al metal, pero no fue posible hacerlo de esa manera, debido a indisposiciones del laboratorio, y lo caro que eran las herramientas necesarias para su elaboración, por lo cual se recurrió a los instrumentos convencionales, para los cortes generales se usó un soplete, véase en la figura 10, el cual deja materiales imperfectos y sobrados. Debido a lo anterior fue necesario darle un acabado con el esmeril, como lo muestra la figura 11, los agujeros se hicieron en un taladro de banco. Figura 12.

Figura 10. Corte con soplete.

Figura 11. Imperfecciones con esmeril.

Figura 12. Maquinado de las placas laterales.

El maquinado de las placas laterales fue muy sencillo una vez ya cortadas solo fue necesario hacer los agujeros con el taladro de banco, el único inconveniente que se encontró, fue que se tuvo que hacer uso de varios tamaños de brocas de forma ascendente para poder penetrar en la placa, las barras fueron muy sencillas de perforar al ser soleras en las cuales solo se usaron dos tamaños de broca.

El tiempo de elaboración fue algo extendido, teniendo un total de 8 barras largas las cuales se necesitó un tiempo aproximado de 1 hora y media cada una, las 4 barras cortas 1 hora cada una, además el resto de los elementos tomaban entre 4 y 5 horas al tener más cortes y algunos de los cuales se tienen que hacer con mayor cuidado ya que al hacerlos de forma incorrecta podría provocar problemas de funcionamiento.

Figura 13. Maquinado de la placa base.

2.6 Gripper ensamblado

Una vez elaboradas las piezas solo se ensamblaron todos los componentes mediante tornillos y tuercas de seguridad y agregando los baleros donde fuese necesario para mejorar el movimiento del mecanismo; Se montó el pistón y la electroválvula también con el uso de tornillería, esto nos permite ensamblar y desensamblar de manera sencilla, aunque reduce la rigidez a comparación de la soldadura, pero esto nos ayuda a seguir haciendo pruebas y seguir mejorando el diseño con facilidad. Como podemos apreciar en las Figuras 14 el mecanismo de barras, la figura 15 el ensamblaje y la figura 16 el gripper ensamblado.

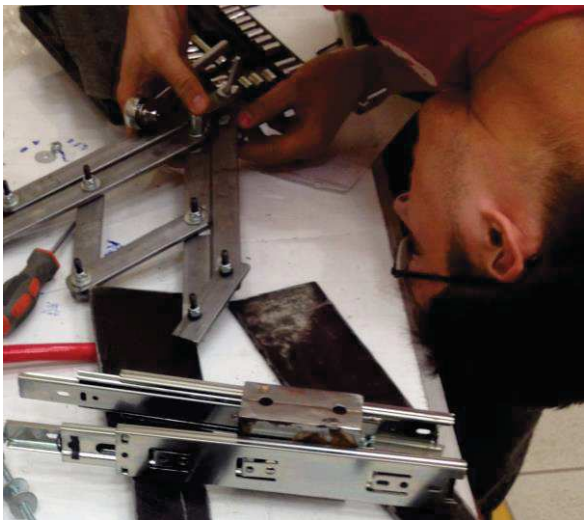


Figura 14. Mecanismo de barras.

Figura 15. Ensamblaje de gripper.

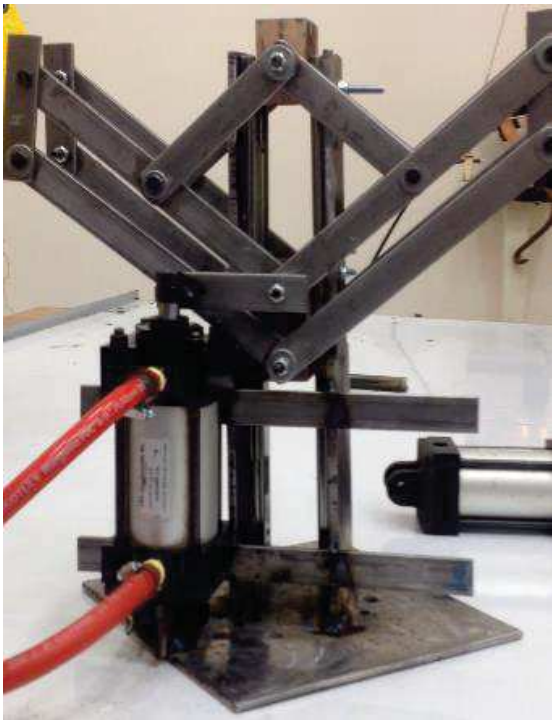


Figura 16. Gripper ensamblado.

Una vez ya montado en el robot se hicieron pruebas de movimiento sobre el robot, todo funcionaba tal cual el software nos arrojaba las predicciones, se sometió a las mismas pruebas de peso que se hicieron un reaccionó de una manera favorable, actualmente es utilizado por los alumnos para hacer prácticas en el laboratorio y también se sigue trabajando el diseño para hacerlo más eficiente y poder manipular mayores pesos sin sufrir tanta fatiga.

Ingeniería y Automatización, Capítulo 10, pp. 121 - 135.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Resultados

3.1 Resultados del estudio de elemento finito

Las figuras 17,18 y 19 muestran los primeros resultados del estudio, dando a conocer el comportamiento del prototipo de acuerdo a los esfuerzos y cargas a aplicadas, mostrando los esfuerzos, el factor de seguridad y los desplazamientos respectivamente.

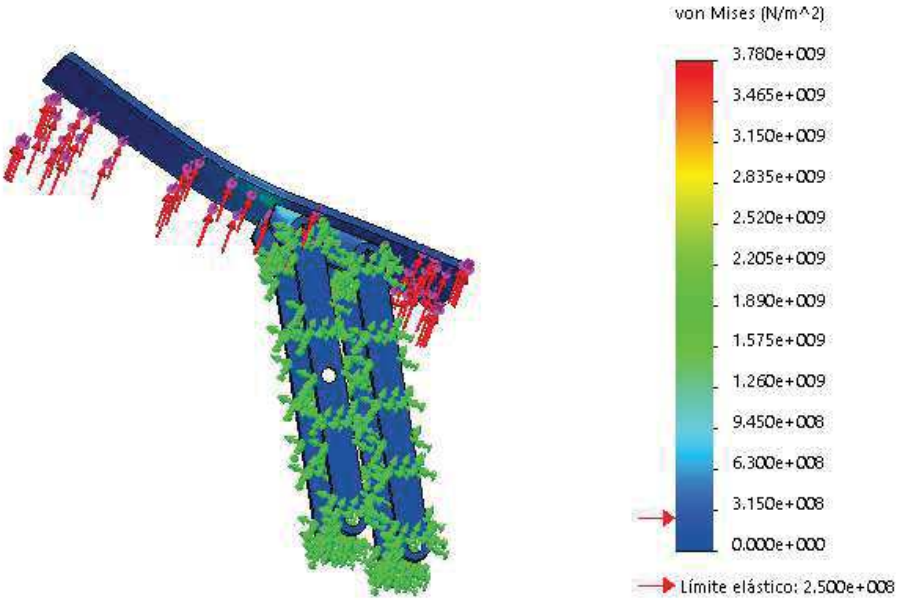


Figura 17. Esfuerzos en N/m^2

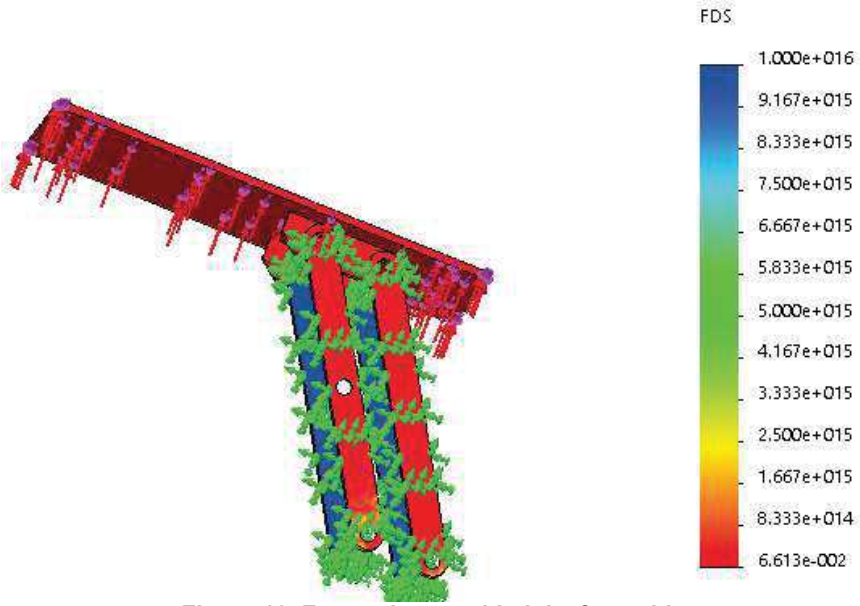


Figura 18. Factor de seguridad desfavorable.

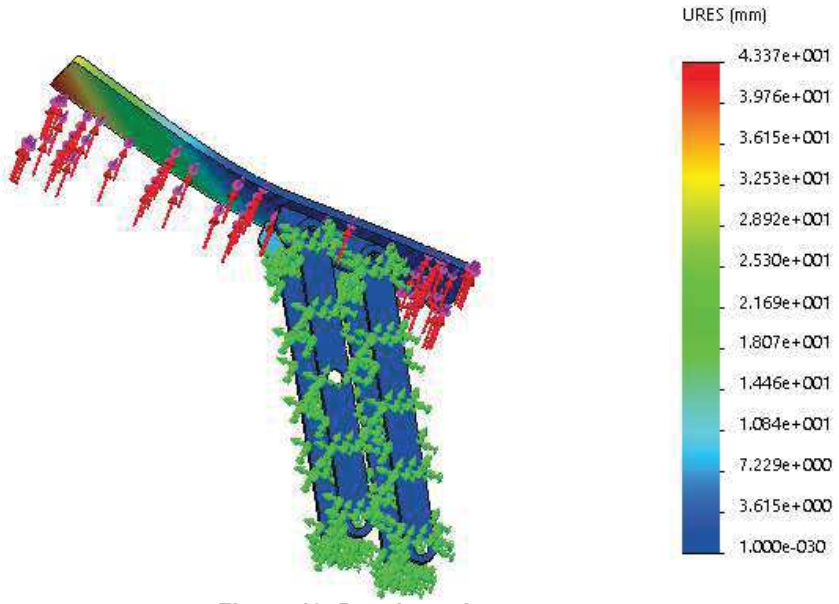


Figura 19. Desplazamientos en mm.

Gracias a los resultados que se obtuvieron con el Estudio de elemento finito se llegó a la conclusión de aumentar el grosor de las agarraderas, para que no sufriera alguna deformación permanente, ya que el factor de seguridad se mostró desfavorable. Y como observamos en los desplazamientos, hay un lugar donde su máximo desplazamiento es de 4.337 mm y esto no es conveniente.

3.2 Nuevo estudio de elemento finitos, con las correcciones pertinentes.

El segundo análisis da indicios de los mismos comportamientos anteriormente señalados, pero con las agarraderas más gruesas, como se muestran en las figuras 20,21 y 22.

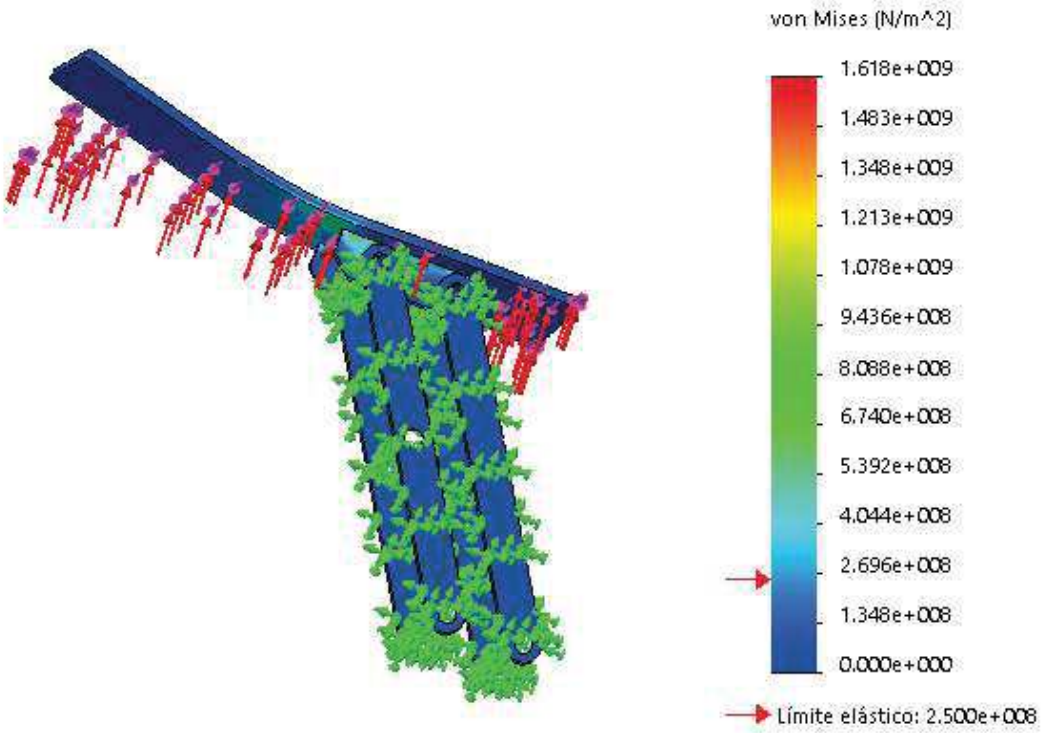


Figura 20. Esfuerzos en N/m^2

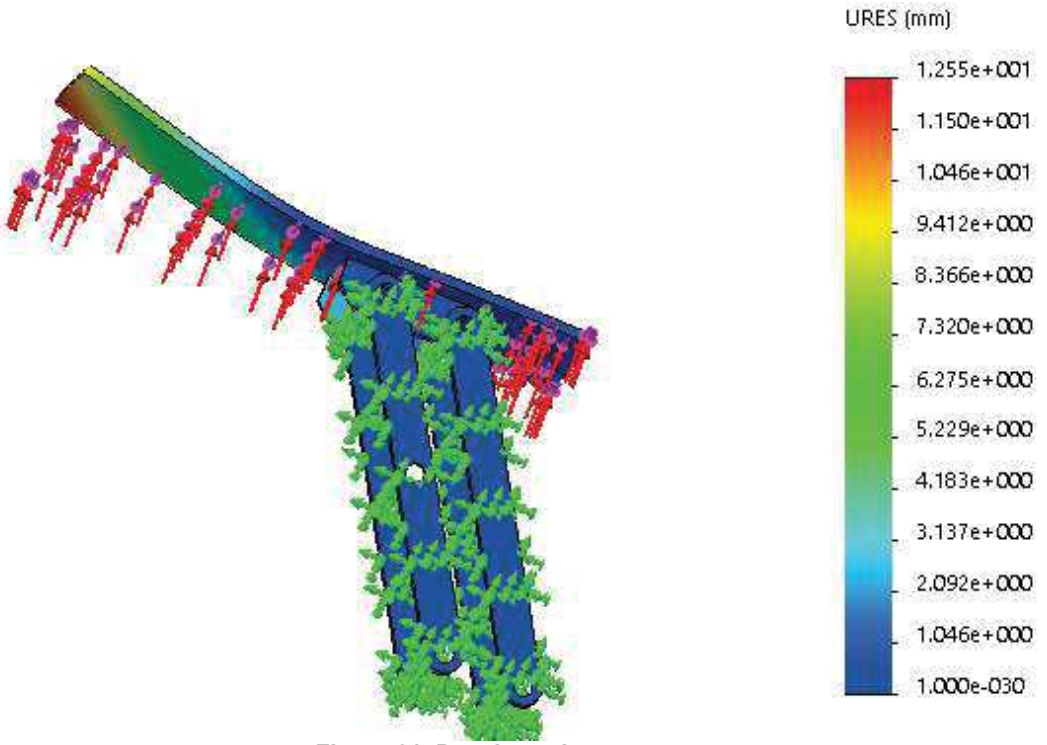


Figura 21. Desplazamientos en mm.

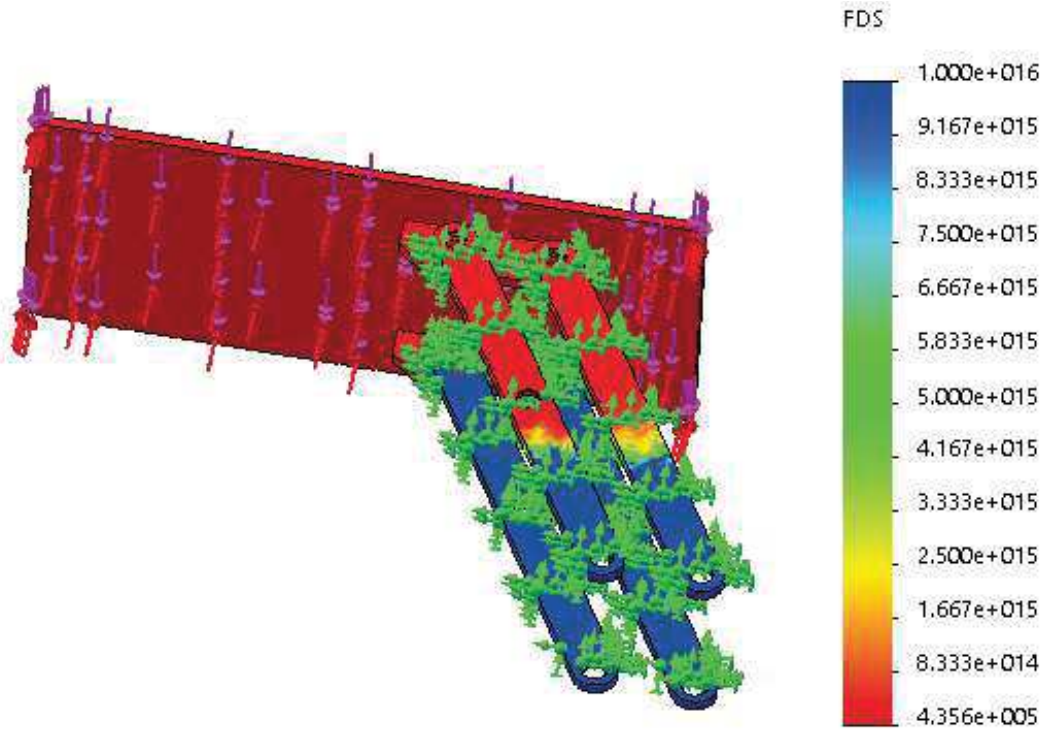


Figura 22. Factor de seguridad favorable.

Gracias a que se aumentó el grosor de las agarraderas del Gripper, con el nuevo estudio de elemento finitos, los resultados dieron un giro muy favorable para el proyecto, reduciendo los esfuerzos en los puntos críticos, así como también, se disminuyeron los desplazamientos de manera muy significativa y se aumentó el factor de seguridad.

3.3 Resultado final del Prototipo de Gripper Industrial

La figura 21 muestra el prototipo ya terminado y ensamblado listo para ser usado.

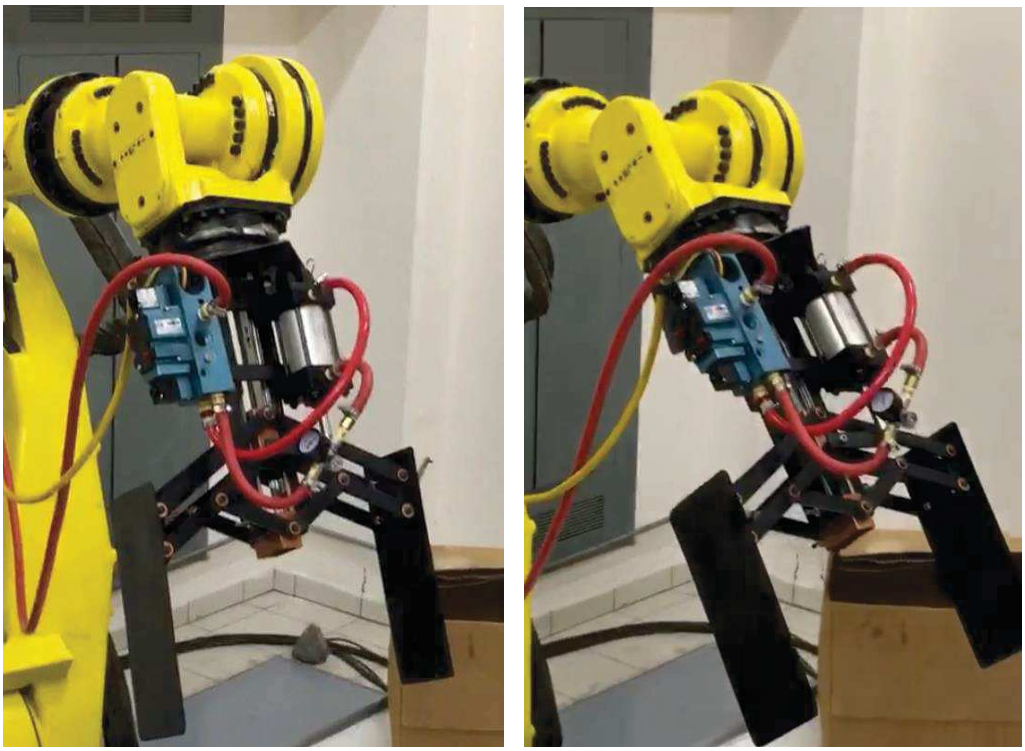


Figura 21. Resultado final neto.

4. Conclusiones

Como se ve reflejado a través de todas las pruebas y resultados, estos fueron satisfactorios al poder utilizar el proyecto ya montado y funcional, claro que mediante se fue avanzando se resaltan varios puntos de mejoría dentro del mismo, desde el diseño hasta los materiales mismos, siendo uno de los casos el material usado para las paletas que en un principio al ser tan delgadas con el uso rápidamente sufrieron una deformación, el cual fue corregido con una paleta de mayor grosor.

El análisis del mismo no acaba aun, en el corto plazo se está evaluando reemplazar el riel central para hacer un movimiento más fluido y no forzar tanto el soporte central, además también se está trabajando en una base más sólida para el pistón y la electroválvula que están sobre el gripper, para así obtener menor vibración a la hora de la manipulación.

El desarrollo de los mismo será propuesto como un proyecto cada semestre para que los alumnos puedan adentrarse en el mundo de la robótica, además del desarrollo industrial de maquinaria, y se puede conseguir grippers para diferentes usos no solo el de tomar cajas, si no el de poder manipular cualquier tipo de geometría y así enriquecer cada vez más el aprendizaje del mismo con diferentes simulaciones de líneas de producción y diferentes problemas que resolver.

Referencias

[1] Wolf Andreas, Steinmann Ralf, Schunk Henrik. "*Grippers in Motion",* Editorial Springer Verlag Berlin Heidelberg 2005, ISBN 3-540-25657-1.
[2] Página del producto: softwar**e** Handling PRO de ROBOGUIDE, consultado 6/08/2018 Disponible en: http://www.fanucrobotics.com.mx/Productos/Software-vision/simulation-software.aspx
[3] Página del producto: softwar**e** Solidworks, consultado 6/08/2018 Disponible en: https://www.solidworks.com/es
[4] MILLTRONICS MANUFACTURING COMPANY, INC. *"RW15"*, 9 páginas, 01 de septiembre del 2014.

Ingeniería y Automatización, Capítulo 11, pp. 136 - 146.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Desarrollo de una Interfaz para la Generación de MegaDatos del Desempeño de Automóviles

Sánchez Cuapio Iván Jesús
Universidad Tecnológica De Tlaxcala
ivan.sanchez@uttlaxcala.edu.mx

Resumen

Actualmente se ha estado potenciando el uso de las tecnologías de comunicación con el fin de tener conocimiento del desempeño de todo lo que gira alrededor de las personas, el internet de las cosas "IoT", ha tenido un auge incremental, de tal modo que los datos generados cotidianamente pueden ser consultados desde dispositivos con acceso a la red y así mismo generar bases de datos que permitan almacenar tendencias precisas de los parámetros registrados.

Para los automóviles el uso del "IoT", presenta una posibilidad única para el monitoreo y conocimiento del desempeño de estos medios de transporte, de este modo se ha ido acrecentando la tendencia de brindar a los automóviles de esa conectividad hacia la red de tal modo que se pueda ingresar a la nube esta información para determinar un diagnóstico de los mismos. Los automóviles de última generación tienen la posibilidad de generar está información, más sin en cambio el 80% de los automóviles que ya se encuentran en circulación no consideran esta posibilidad y no podrían desarrollarla.

El diseño e implementación de una interfaz que permita obtener la información del desempeño de un automóvil, brinda la posibilidad de tener en un medio accesible como lo es una página web el registro y almacenamiento de estos parámetros, que permitirían a un automóvil con el protocolo OBDII tener la posibilidad de integrarse a la alimentación de MegaDatos que involucrarían a los automóviles de una región.

La información obtenida de estos MegaDatos se considera una herramienta con posibilidades de ser interpretadas en diferentes ámbitos de la comunidad, hablemos de la posibilidad de que un automóvil con un reciente ajuste, pueda ser monitoreado con el fin de saber si una falla ha sido corregida de manera correcta, así mismo en el tema ambiental es pertinente saber cómo se comportan en tiempo real los automóviles en consideración de la emisión de contaminantes al medio ambiente, identificar las marcas y tiempos de desgaste de los automóviles considerando los recorridos y tiempo de vida de los elementos de ajuste.

Palabras clave: MegaDatos, Automovil, Microcontrolador, Desempeño.

1. Introducción

El complemento cada vez más notable entre las telecomunicaciones y las tecnologías de un automóvil, ha logrado que cada compañía genere en sus nuevos modelos una conectividad al mundo de manera específica y confiable. El intercambio de esta información mantiene aspectos personales de entretenimiento o del mismo modo estos datos contienen de manera precisa el manejo y comportamiento del sistema que administra y controla a los vehículos. La generación de MegaDatos de las flotillas con esa capacidad, ha tenido la mira de ser explotados en años cercanos, considerando así una evolución de los procesos de manufactura y mantenimiento predictivo de los mismos. Con toda la información generada en tiempo real de los parámetros de funcionamiento de todos los automóviles, es posible generar tomas de decisión que involucren la mejora en muchos de los procedimientos que pertenecen a este ramo del transporte.

El procesamiento de la base de datos obtenida después de algunos años de muestreo, logrará considerar la implementación de metodologías que permitan realizar tomas de decisión que reflejen mejoras en los procedimientos tales como control de tráfico, contingencias ambientales, mantenimientos predictivos y estadísticas financieras. La culminación del uso de los MegaDatos lleva una tendencia de aplicación para la siguiente década, donde bajo este panorama aún tendremos la interacción con modelos de autos que ahora son recientes y que continuaran en declive de su funcionamiento y sin la capacidad de ser monitoreados bajo esos panoramas. En este momento el trabajo con las “IoT” es una inicialización a este rubro del tratamiento de la información, y la integración de tecnologías con esta capacidad es un ámbito muy propio del cual se puede obtener trabajo futuro.

Mientras que los modelos recientes de automóviles contienen la conectividad adecuada a manera de entretenimiento, es necesario que se establezcan capacidades de monitoreo pero del aspecto técnico propio de los automóviles, esta interacción entre el usuario y el automóvil requiere de dispositivos capaces de generar esa comunicación. La propuesta de este trabajo es generar sistemas embebidos que logren interactuar y obtener los parámetros de un automóvil en tiempo real, para ser llevados a un monitoreo y almacenamiento de los mismos en la denominada nube. El primer paso en la generación de MegaDatos de vehículos es el trabajo individual propio de la integración de sistemas de este tipo y que por ahora son ajenos al sistema electrónico automotriz.

Un sistema embebido es la compactación de circuitos con propósito específico que permiten realizar una tarea y que son formados por algunos subsistemas que favorecen al cumplimiento de la misma. El uso de microcontroladores es la opción inicial para la generar prototipos con estas características, debido a que su arquitectura permite el manejo de protocolos de comunicación para generar el vínculo del sistema electrónico del vehículo y la capacidad de integrar sistemas que permitan llevar esta información al internet.

Ingeniería y Automatización, Capítulo 11, pp. 136 - 146.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2. Integración de la Interfaz y la Generación de MegaDatos

El desarrollo de una interfaz que permita realizar una comunicación con el ordenador del automóvil, requiere interactuar con los parámetros y recursos necesarios involucrados en el monitoreo del desempeño considerando, por tanto, al diagnóstico a bordo de los vehículos como la herramienta óptima para utilizar.

2.1 Diagnóstico De Automóviles

En la actualidad, la mayoría de dispositivos que se utilizan están compuestos por algún dispositivo electrónico que controle o procese información para determinar las acciones propias y la industria automotriz no es la excepción. Los automóviles modernos tienen sistemas inteligentes de control y monitoreo que ofrecen mayor seguridad y confort al conductor y pasajeros.

Todos los elementos de medición y registro, están conectados a un sistema central que integra la información del funcionamiento y estado actual de los sistemas de control del vehículo. Este sistema no es abierto a entregar la información de manera abierta manteniendo solo esa información a sistemas que puedan interpretarla. Actualmente, este sistema central cuenta con un conector de comunicación que permite establecer un enlace entre el sistema central y sistemas de diagnóstico que leen el estado actual de los componentes de cada uno de los sistemas de control, con los cuales el automóvil este equipado. Logrando así, entregar la información requerida para una interpretación adecuada y así mismo se realicen las acciones correspondientes. Existen diferentes herramientas de diagnóstico que son adaptadas a un diverso número de automóviles de distintas marcas y modelos.

A mediados de la década de los noventas, después de casi seis años de investigación entre SAE (Society of Automotive Engineer), CARB y EPA, la nueva generación sistemas de diagnóstico fue lanzada con el nombre de OBDII (Sistema de diagnóstico abordo versión dos) y a partir del 1 enero de 1996, todos los vehículos vendidos tendrían que estar equipados con OBDII. El estándar OBDII incorporó un conector de diagnóstico, así como la ubicación del conector dentro de la cabina del automóvil. El estándar OBDII definió las partes del motor que obligadamente debían ser monitoreadas y bajo qué parámetros. Si uno de los sensores detectaba cualquier mal funcionamiento el sistema lo indicaría en el tablero del automóvil por medio de una luz indicadora de funcionamiento anormal o Malfunction Indicator Light (MIL). Los distintos tipos de fallos se estandarizaron de tal forma que ayudaría a reparar la falla en menos tiempo y de forma correcta.

OBDII estandarizó los protocolos de comunicación con la Unidad de Control del Motor o ECU (Engine Control Unit.). La estandarización de protocolo de comunicación facilitó el diagnóstico de los automóviles, ya que no se necesita de herramienta de diagnóstico propia del fabricante. OBDII tiene como fin controlar de una manera más rigurosa la emisión de gases y la vez diagnosticar el funcionamiento anormal de automóviles para cumplir con las reglas establecidas por las instituciones ambientales.

Se ha establecido entonces, que la base primordial para el desarrollo del proyecto es el análisis y entendimiento del sistema OBDII, que monitorea una gran variedad de condiciones y sensores mientras el automóvil es conducido. Cuando el sistema detecta un problema en el sistema de control de emisiones enciende la luz MIL, y a su vez, el sistema almacena un código de falla en la memoria de la computadora. El código tiene un formato en el cual indica en qué parte del automóvil ocurrió la falla y bajo qué condiciones. [1]

El sistema ODBII cuenta con un conector estándar llamado Data Link Connector (DLC) o J1962. Este conector cuenta con 16 pines, los cuales tienen asignados una función específica para cada protocolo. Como se muestra en la figura 1.

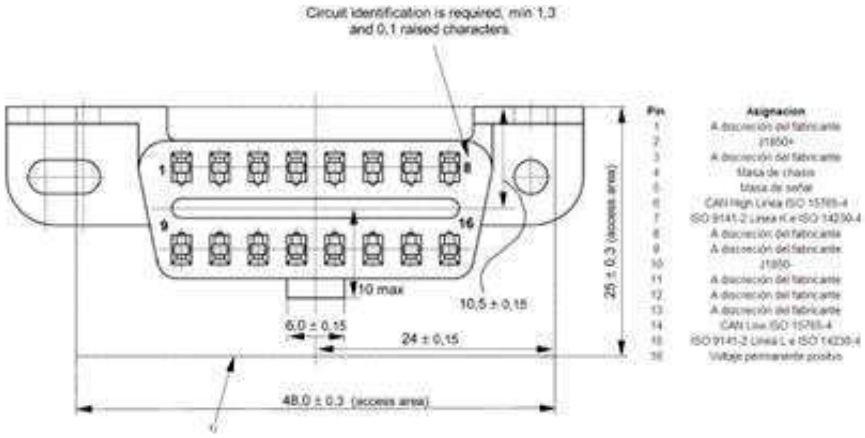


Figura 1. Data Link Connector.

Luego de establecer comunicación con la ECU, se tiene la opción de poder entrar a los 9 modos de trabajo. Cada modo de trabajo tiene información acerca de monitoreo de sensores en tiempo real, registro de las condiciones al ocurrir una falla, información del automóvil, borrado de códigos de falla, resultados de pruebas de sensores de oxígeno, resultado de pruebas de sistema u otros componentes y control de operación de sistema de diagnóstico abordo.[1]

2.2 Protocolos De Comunicación

Es un conjunto de reglas y normas establecidas que permiten una comunicación exitosa entre dos o más dispositivos para el intercambio de información. Si se quiere acceder a la ECU para diagnosticar el automóvil se hace indispensable una herramienta de diagnóstico con el protocolo de comunicación que resida dentro del automóvil. Los tipos de protocolo se originan de dos organizaciones: ISO y SAE.

Los protocolos estandarizados por ISO se dividen en tres:

- ISO 15765-4 (CAN,Controller Area Network),
- ISO 91412,
- ISO 14230-4 (KWP2000, Keyword Protocol 2000).

Los protocolos estandarizados por SAE se dividen en dos:

- SAE J1939 y
- SAE J1850 (PWM (Pulse Width Modulation) y VPW (Variable Pulse Modulation)

Cada protocolo de comunicación tiene asignado uno o varios pines. Según la marca del automóvil y el protocolo con el cual este equipado, así será cantidad de pines que se utilicen del conector DLC-OBDII. El conector DLC-OBDII macho se utiliza para las herramientas de diagnóstico.

La asignación de pines para DLC-OBDII es en espejo. Los 10 modos de operación para el estándar OBDII que permiten acceder a los distintas partes de la ECU. Los modos de operación y su descripción se muestran en la tabla 1.

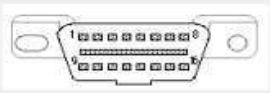
Tabla 1. Modos de Trabajo OBDII.

Modo (hex)	Descripción
0x01	Mostrar datos actuales
0x02	Mostrar datos de Freeze Frame
0x03	Mostrar códigos de diagnóstico almacenados
0x04	Borrar códigos de diagnóstico y valores almacenados
0x05	Resultados de pruebas de monitoreo del sensor de oxígeno (no aplica para CAN)
0x06	Resultados de pruebas, otro componente de monitoreo / sistema (resultados de pruebas, monitoreo del sensor de oxígeno, solo CAN)
0x07	Mostrar códigos de diagnóstico pendientes (detectados durante el ciclo de conducción actual o pasado)
0x08	Control de funcionamiento del componente de a bordo o sistema
0x09	Solicitud de información del vehículo
0x0A	Códigos de diagnóstico permanentes. (Códigos de diagnóstico borrados).

Al acceder a cada modo, la información de respuesta usará código hexadecimal para interpretar la información establecida por el estándar y obtener el valor deseado.[1]

Estos datos se conocen como PID (Parameter ID). PID es un código de solicitud de información que se envía a la ECU según el modo OBDII. Cada modo en OBDII tiene PID estandarizados. Dependiendo del modo y del PID, la respuesta se interpreta de distinta forma. Figura 2.

Connector pinout:



PS: It should be noted that the remaining pins may be used by the manufacturer as needed.

Pin no.	Description
1	
2	J1850 BUS+ (SAE)
3	
4	Chassis ground
5	Signal ground
6	CAN High
7	K line (ISO)
8	
9	
10	J1850 BUS- (SAE)
11	
12	
13	
14	CAN Low
15	L line (ISO)
16	Battery +

Figura 2. Pines de Conexión.

2.3 Adquisición De Datos De Diagnóstico De Los Automóviles

El punto desde ahora es generar un sistema que integre una interfaz que sea capaz de lograr una comunicación continua con la ECU del automóvil y tener en tiempo real el comportamiento de los distintos sensores que reflejaran el desempeño completo del mismo. Sin perder de vista que se debe complementar con el ingreso de datos a la nube, generando una base de datos en un web server.

La arquitectura de esta interfaz se muestra en la figura 3.

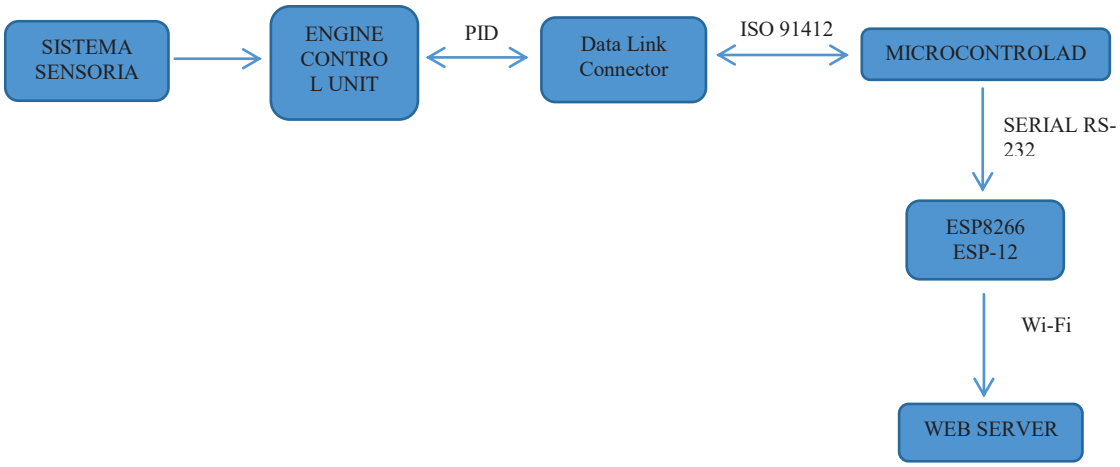


Figura 3. Arquitectura de Interfaz.

En esta estructura se observa el flujo de datos y los modos de comunicación que se presentan entre cada uno de los elementos de está interfaz. Describiendo la interacción desde la obtención de comportamiento del automóvil hasta su puesta en un Web Server para su publicación en Internet.

2.4 Interfaz De Comunicación

El desarrollo de este trabajo inicia en el conector OBDII en comunicación con el microcontrolador por medio de un puerto serial, utilizando el protocolo RS-232 como método de comunicación. Esta adaptación es realizada con el circuito integrado ELM327, que puede ajustarse al protocolo de trabajo del conector y poder dejarlo en niveles de 5v, justos para ser manipulados por un microcontrolador. Figura4.

Figura 4. ELM327.

Esta conexión se realiza a partir del uso del microcontrolador de Microchip PIC18F87J50 (Figura 5) el cual contiene en memoria RAM el parámetro ID para determinar el monitoreo en tiempo real del comportamiento del automóvil el modo 0x01 del OBDII, considerando igual la desconcatenación de los datos en hexadecimal y su conversión a números enteros los cuales serán llevados el módulo ESP8266.

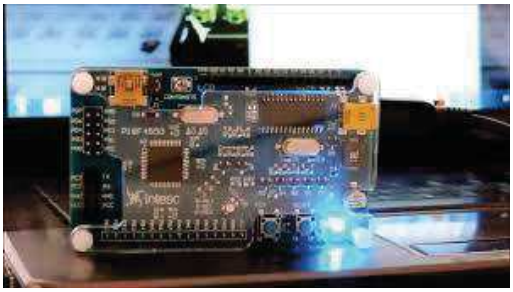


Figura 5. PIC 18F87J5 MIUVA.

```
#include <18F87J50.h>
#use delay(clock=48MHz,crystal=8MHz,USB_FULL)
#include <usb_cdc.h>
#use rs232(baud=9600, xmit=pin_c6, rcv=pin_c7, bits=8, parity=N)
char dato[], datoi[];

#int_RDA
RDA_isr()
  {
   dato[0]=GETC();
   dato[1]=GETC();
   dato[2]=GETC();
   dato[3]=GETC();
   dato[4]=GETC();
   dato[5]=GETC();
   lcd_putc("\fCODIGO RECIBIDO");

printf(lcd_putc,"%c%c%c%c%c%c",dato[0],dato[1],dato[2],dato[3],dato
[4],dato[5]);
  }
do{
      if(usb_cdc_kbhit())
        {
          datoi[0]=usb_cdc_getc();
          datoi[1]=usb_cdc_getc();
          datoi[2]=usb_cdc_getc();
          datoi[3]=usb_cdc_getc();
          datoi[4]=usb_cdc_getc();
          datoi[5]=usb_cdc_getc();*/
          lcd_gotoxy(1,1);
          lcd_putc("\fCODIGO  ID ENVIADO");
          lcd_gotoxy(5,2);
printf(lcd_putc,"%c%c%c%c%c%c",datoi[0],datoi[1],datoi[2],datoi[3],da
toi[4],datoi[5]);
          printf("%c%c%c",datoi[0],datoi[1],datoi[2]);*/
          delay_ms(1000);
          lcd_putc("\f");
        }
       // printf(usb_cdc_putc,"CODIGO EN HEXADECIMAL");
       /*lcd_putc("HOLA");
       fputs("Codigo desconcatenado");*/
       delay_ms(200);
       lcd_gotoxy(4,1);
       lcd_putc("niveles de oxigeno:%d", oxi);
       lcd_gotoxy(3,2);  /// COLUMNA, FILA
       lcd_putc("dato almacenado en web server");
         }
  while(true);
```

El módulo ESP8266 ESP-12 ha sido desarrollado de tal manera se pueda llevar información de un medio físico a un servidor web, considerando una segunda manipulación de puerto serial en el microcontrolador, siendo este configurado de tal modo que pueda llevarse en conexión con el servidor web. Figura 6.

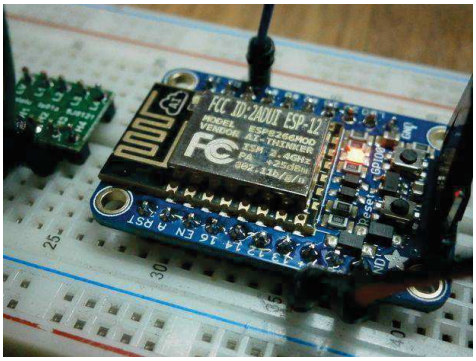


Figura 6. ESP8266 ESP-12.

El código desarrollado en el compilador CCS, nos da la posibilidad de generar la configuración adecuada para realizar su ingreso a una red de área local, que con su conexión a internet puede lograrse el desarrollo del WebServer.

```
#include <18F87J50.h>
#use delay(clock=48MHz,crystal=8MHz,USB_FULL)
#use rs232(baud=115200,xmit=pin_C6, rcv=pin_C7, wi-fi)
char const name[]="xxxx", pass[]="xxxxxxxxxxx";
char const server[]="xxx.xxx.xxx.xx", port[]="xxxx";

void set_wifi_mode()
{
//printf(lcd_putc,"\f ");
//lcd_gotoxy(1,1);
//printf(lcd_putc,"Setup wifi");
printf("AT+CWMODE_DEF=3");
esp_crlf();
}
void connect_ap()
{
//printf(lcd_putc,"\f ");
//lcd_gotoxy(1,1);
//printf(lcd_putc,"Wifi: %s",name);
// lcd_gotoxy(1,2);
// printf(lcd_putc,"Connecting...");
printf("AT+CWJAP_DEF=\"%s\",\"%s\"",name,pass);
esp_crlf();
}
void esp_connect_mode()
{
printf("AT+CIPMUX=0");
esp_crlf();
}
/*********************************************************
**********/
void connect_to_server()
{
printf("AT+CIPSTART=\"TCP\",\"%s\",%s\r\n",server,port);
delay_ms(1000);
}
/*********************************************************
**********/
void send_data()
{
connect_to_server();
char str[24]="";
unsigned int8 num;
num=sprintf(str,"t,%2.0f,h,%2.0f,",temp,humid);
printf("AT+CIPSEND=%u\r\n",num);delay_ms(500);
printf("%s\r\n",str);
}
```

2.5 Generación De MegaDatos

La parte medular del sistema recae en el microcontrolador, que resulta ser el intermediario entre la ECU y la Web Server, la disposición de un sistema basado en microcontrolador se justifica por la portabilidad del mismo, considerándose como un dispositivo adecuado al montaje sobre el sistema electrónico del automóvil, sin considerar una invasión. Así mismo el dispositivo ESP8266 ESP-12 considera una conexión inalámbrica que puede ser obtenida a partir de una tarjeta SIM con conexión a la red celular o de manera que se pudiera conectar a una red local de un dispositivo móvil o del mismo sistema de conectividad del automóvil. A partir de ello el alojamiento de la información en una base de datos puede del mismo modo tener una diversidad de posibilidades.

El servidor realizado es desarrollado bajo la naturaleza de programación de C#, donde se establece la recepción de los datos del módulo de comunicación y se hace un almacenamiento de la misma. La esencia básica de esta metodología es la configuración de un Web Socket para la recepción y actualización de datos. Vale la pena mencionar que este servidor se ajusta a RFC 6455 por lo que solo manejará las conexiones de Chrome versión 16, Firefox 11, IE 10 y superiores. Y la estructura básica se determina como:

```
class Server {
   public static void Main() {
      TcpListener server = new
TcpListener(IPAddress.Parse("xxx.xxx.xxx.xxx"), xxx);
      server.Start();
      Console.WriteLine("El server se ha iniciado en
xxx.xxx.xxx.xxx:xxx.{0}Esperando una conexión...",
Environment.NewLine);
      TcpClient client = server.AcceptTcpClient();
      Console.WriteLine("Un cliente conectado.");
   }
}
TcpClient client = server.AcceptTcpClient();
Console.WriteLine("Un cliente conectado.");
NetworkStream stream = client.GetStream();
while (true) {
   while (!stream.DataAvailable);
   Byte[] bytes = new Byte[client.Available];

   stream.Read(bytes, 0, bytes.Length);
}
using System.Text;
using System.Text.RegularExpressions;

Byte[] bytes = new Byte[client.Available];

stream.Read(bytes, 0, bytes.Length);

String data = Encoding.UTF8.GetString(bytes);

if (new Regex("^GET").IsMatch(data)) {
} else {

}
```

Hasta este punto se debe volver a considerar que el propósito de la generación de MegaDatos es el almacenamiento de una cantidad de información que después pueda ser analizada pero que tenga la posibilidad de tenerse a disposición, así que la estructura de este almacenamiento en el mismo ambiente de C# para una alojamiento SQL que se muestra como:

```
Customer newCust = new Customer();
newCust.CompanyName = "****";
newCust.CustomerID = "****";

db.Customers.InsertOnSubmit(newCust);
Console.WriteLine("\nCustomers matching CA before insert");
foreach (var c in db.Customers.Where(cust =>
cust.CustomerID.Contains("CA")))
{
   Console.WriteLine("{}, {}, {}", c.CustomerID,
c.CompanyName, c.Orders.Count);
}
```

Considerando entonces que el intervalo de muestreo desde que es enviado el código ID en el automóvil hasta su publicación y almacenamiento en el Web Server se establece por cada segundo.

3. Resultados

El desarrollo de la interfaz fue establecido en un periodo de cuatro meses y su conexión al servidor en un mes más, a partir de este punto en instalaciones de la Universidad Tecnológica de Tlaxcala se han hecho las pruebas y mediciones en un automóvil propio de la casa de estudios, de un JETTA GLI 2017, en el cual se ha realizado el monitoreo de los niveles de emisión de Óxidos de Nitrógeno (NOx), que es uno de los parámetros medibles en el nuevo modelo de verificación vehicular en la Megalópolis. El prototipo montado se muestra en la figura 7:

Figura 7. Prototipo Montado Jetta GLI 2017 - UTT.

De las imágenes se puede observar que al dispositivo puede ser colocado en la caja de fusibles del automóvil sin requerir de ninguna otra adaptación al vehículo, considerando el registro de los parámetros.

Del mismo modo se presenta un diseño inicial de la página web del sitio que aloja la información, y de primera vista se observa en la figura 8.

Figura 7. Aplicación WebServer.

Así, mismo el sistema tuvo que realizar mediciones considerando márgenes de error, debido al ajuste de los parámetros de comunicación, como la velocidad de transmisión de baudios en el puerto serial, sin embargo se tuvieron horas prolongadas de trabajo y el gráfico que muestra la tendencia del desempeño del sensor de NOx se muestra a continuación. Figura 8.

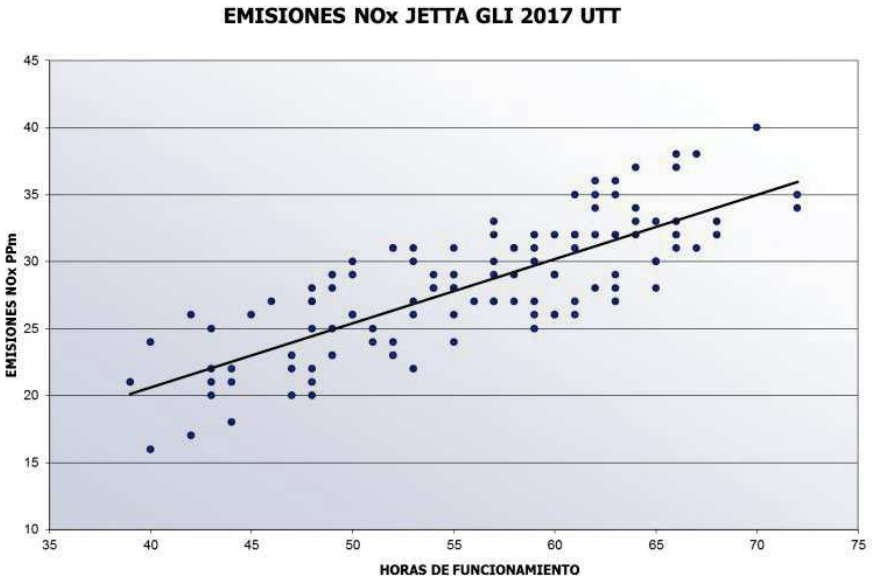


Figura 8. Generación y Análisis de Información.

4. Conclusiones

El trabajo con el Internet de las cosas ha tomado un auge incremental y su aplicación a generado la expectativa de que hacer con esa información puesta en la nube, en muchos de los casos es posible que la información sea tomada y desechada de inmediato puesto que solo representa un impacto al momento de ser evaluada, por ejemplo el nivel de gas de una casa habitación, solo permite conocer el estado actual del mismo; en un segundo término encontramos el inicio de un historial de información, por ejemplo la glucosa medida en una persona puede generar un historial médico que a partir del tiempo muestre el comportamiento a un tratamiento. Así la generación de los Megadatos, llega al análisis de grandes cantidades de información que es almacenada para generar gráficos de comportamientos y tomas de decisión que generen beneficios.

Este proyecto ayuda a desarrollar una herramienta que permite la adquisición de datos provenientes del sensor de Óxidos de Nitrógeno que emite un automóvil, los cuales son tomados para ser almacenados y lograr la generación de MegaDatos que posteriormente pueden ser analizados.

La tendencia de este trabajo al enfocarse a ese sensor es analizar de manera objetiva el sistema de un automóvil para determinar su comportamiento y cumplimiento con los nuevos parámetros de verificación vehicular en la Megalopolis.

Trabajo Futuro

Trabajar con un parámetro único es una de las limitantes de este proyecto y sin duda una de las fuentes de trabajo futuro, la expansión de registro de datos de los diferentes componentes que generan contaminación deben ser registrados y de igual modo analizarse, para determinar el desgaste de los componentes que intervienen en la emisión de sustancias toxicas. Por otra parte se permitirá medir su tiempo de vida de estos elementos, por lo que se podrá determinar un cambio en la planeación de mantenimiento preventivo de los vehículos buscando dar un ahorro a los propietarios con la identificación propia de los elementos que deben sustituirse.

Referencias

[1] Bosh R. *“Manual de la Técnica del Automovil”*,. Bosh, 4ta Ed, pag. 576, 2015.

Ingeniería y Automatización, Capítulo 12, pp. 147 - 158.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Filtro Electrostático de Alta Tensión para la Eliminación de Partículas en un Quirófano con Alarma

García Juárez Ricardo Aarón, Pérez Pamatz Iván, Nicolás Máximo Ángel Eduardo, Nicolás Máximo Santiago y Ramírez Rodríguez Eduardo

Instituto Tecnológico Superior de Pátzcuaro

Resumen.

En vista del alto nivel de contaminación y a que los hospitales se les considera como un foco de infección debido al tratamiento de enfermos, se ha empezado a buscar medidas que contrarresten este problema que puede afectar a la población en general, una solución de este problema es la implementación de filtros que eliminen las partículas ya sean infecciosas o que afecten al medio ambiente, para ello es la implementación de un filtro electrostático que atraiga las partículas por su campo electromagnético y que las elimine por la alta tensión que este maneja, dando una limpieza al quirófano manteniendo en buen estado los equipos médicos y proporcionado una alarma en caso de que haya partículas o gases que puedan afectar al quirófano. Con la finalidad de entender cómo se puede implementar en un quirófano se logró la realización de un prototipo que permite limpiar el aire y mantener una oxigenación mayor del 90%

1. Introducción.

Según la Organización Mundial de la Salud (OMS) los hospitales se pueden considerar como un foco de infección, esto debido al resguardo de enfermos y esto a su vez también de la acumulación de agentes patógenos [1]. La OMS señala que la tasa de infección entre los enfermos hospitalizados no debe ser mayor al siete por ciento, porque una tasa elevada atribuible a infecciones intrahospitalarias prolonga la hospitalización de 5 a 10 días en promedio [2]. En México, señala la OMS, se calculan 450,000 casos de infección relacionada con la atención sanitaria que causan 32 muertes por cada 100,000 habitantes por año.[1]

La eliminación electrostática de las moléculas se dio a conocer en los primeros años del siglo xx. Desde esa época su aplicación en las industrias ha ido creciendo. En la actualidad es uno de los procesos de eliminación de moléculas más empleados en las industrias [1]. El principio de funcionamiento electrostático de alta tensión fue dado por el ingeniero eléctrico Nikola Tesla debido a sus investigaciones de la creación de un campo electrostático que brindara energía inalámbrica, pero al necesitar un campo eléctrico demasiado grande creaba rayos de alta tensión con lo cual atraía las partículas y las eliminaba. El físico y químico estadounidense Frederick Gardner Cottrell tomo parte del trabajo de Nikola Tesla y en 1906 desarrolló el primer precipitador electrostático comercial (solo atrae partículas), exitoso, que fue instalado en una planta manufacturera de ácido cerca de Pinole, California. [3]

Hasta el momento se han aplicado filtros electrostáticos en diferentes áreas de la industria pero no en el área de la salud más específicamente en los quirófanos de hospitales ya que es donde por la limpieza se generan gases oxidantes de alcohol lo que daña todo el equipo médico, además que todos los patógenos viajan por los conductos de aire y desembocan en el filtro de micro mallas que hay en el quirófano con el fin de no expandirse en todo el hospital. Al aplicar un filtro electrostático en el quirófano se asegura un área libre de oxidantes que afectan al equipo médico dándole durabilidad, además se eliminarían los patógenos que viajan en los conductos de aire evitando infecciones a los pacientes y al personal del hospital [1]

Ingeniería y Automatización, Capítulo 12, pp. 147 - 158.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2. Funcionamiento y estructura.

El principio de ionización es la base del funcionamiento de un precipitador electrostático, es decir, las partículas que pasan a través de las placas se ven afectadas por un campo eléctrico, lo que hace que estas se carguen eléctricamente. Los gases pasan por el interior de una cámara, lugar donde se crea el campo electrostático formado por electrodos de descarga (negativo) y electrodos colectores (positivo); ambos electrodos son de placas de cobre. Cuando pasa el gas por las placas, este se carga negativamente debido al principio de ionización, y es atraído a los electrodos colectores debido a su diferente carga eléctrica esto se ilustra en la siguiente figura. [4]

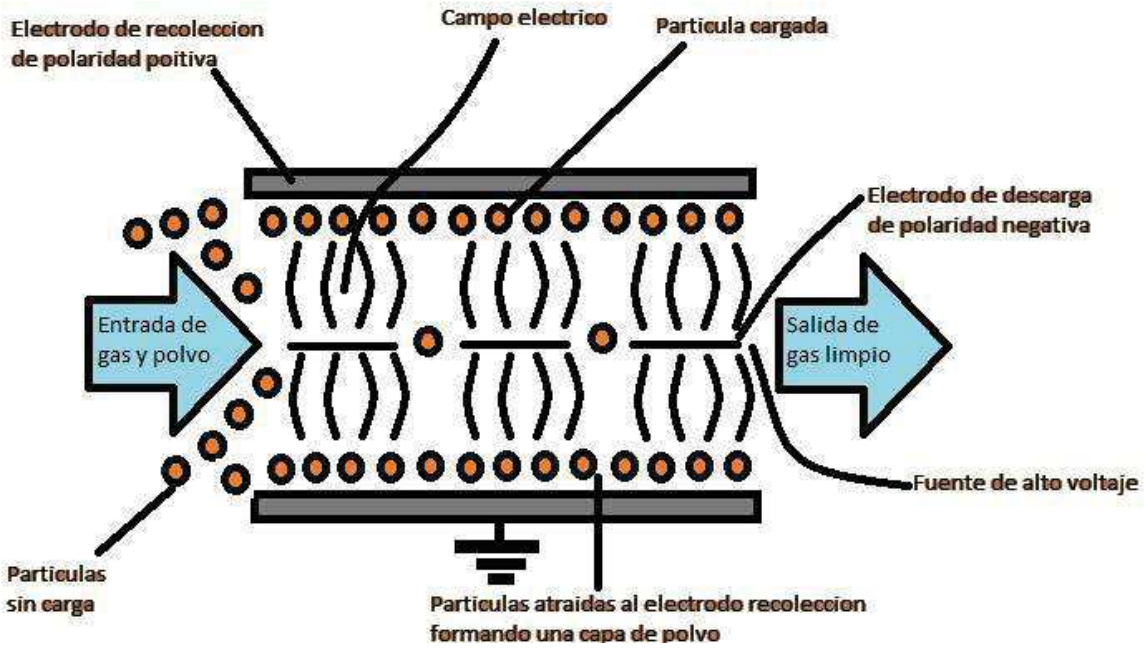


Figura 1. Funcionamiento del filtro electrostático

2.1 implementación en el quirófano.

La conducción de aire dentro de un quirófano es de manera circular, de tal forma que el aire es absorbido y pasa por conductos que desembocan en un filtro de micro mallas en el cual se atrapan los agentes patógenos así también como partículas de polvo, el inconveniente de este proceso es que ese mismo aire que paso por el filtro vuelve a entrar al quirófano de tal manera que no es un aire nuevo completamente limpio.

Un inconveniente es la esterilización del quirófano ya que este proceso libera oxidantes que dañan los instrumentos médicos el cual no puede ser eliminado por el filtro de micro mallas.

Ingeniería y Automatización, Capítulo 12, pp. 147 - 158.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

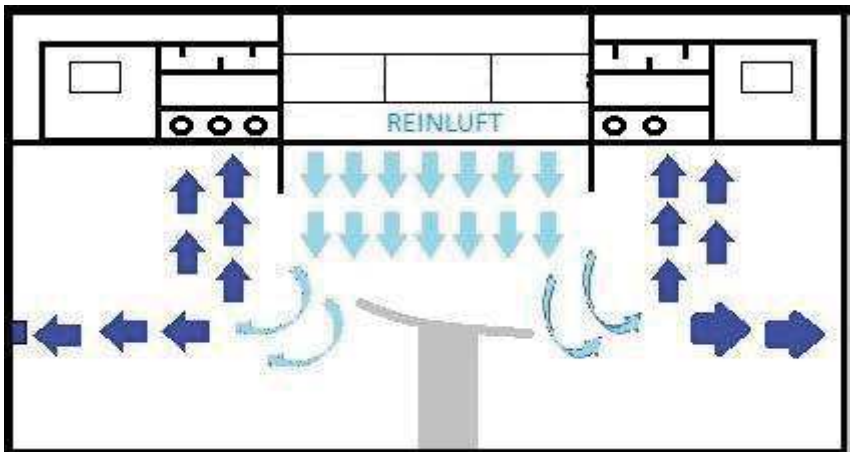


Figura 2. Filtro convencional de quirófano

Para la implementación del filtro electrostático en el quirófano se sustituirá el filtro de micro mallas por el filtro electrostático de alta tensión por donde pasara el aire después pasara por otro filtro de mallas y finalmente se liberara el aire filtrado hacia el exterior, para la oxigenación se agregó una bomba de aire que suministra el oxígeno al quirófano, para saber la calidad de aire que se encuentra en el quirófano se agregaron 2 sensores para su nivel los cuales al descender la calidad de aire activaran una alarma que indique su baja el diagrama de la implementación se ve a continuación.

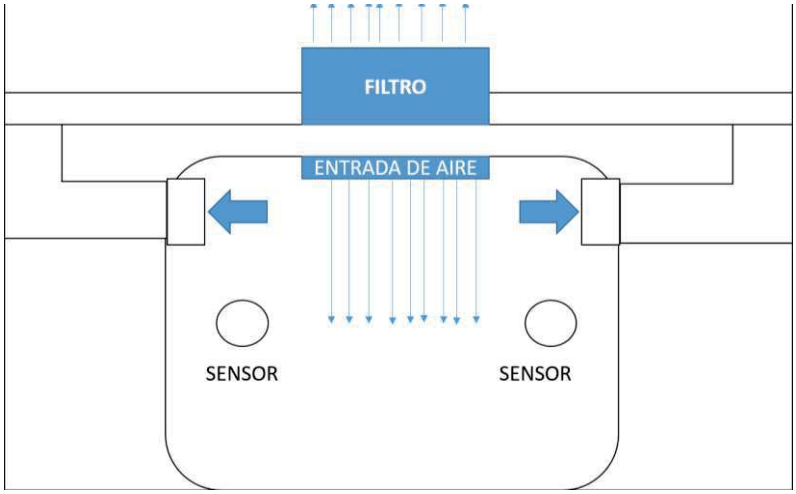


Figura 3. Implementación del filtro electrostático

La acción del filtro está controlado manualmente en el cuarto de control esto debido a que solo es necesario cuando hay una operación en el quirófano y después de la esterilización.

2.2 Diseño del filtro electrostático de alta tensión

Para la elaboración del filtro se debe de tener los siguientes cálculos.

La velocidad de deriva *we* es el resultado del balance entre la fuerza electrostática, gracias a la carga *Fe*, y la fuerza de arrastre, *Fd*, ejercida por el aire debido a la relación del movimiento entre el aire y las partículas: [4]

$$E = \frac{\Delta V}{d} \quad (1)$$

$$Fe = qE \quad (2)$$

$$Fd = \frac{3\pi\,\mu f\; dp\, We}{Cc} \quad (3)$$

$$Fe = Fd \;\rightarrow qE = \frac{3\pi\,\mu f\; dp\, We}{Cc} \rightarrow We = \frac{Cc\; qE}{3\pi\,\mu f\; dp} \quad (4)$$

Donde:
Q = carga de cada partícula
E = campo eléctrico
C_c= factor Cunningham
μf = viscosidad del fluido = 1.81 x 10–5 kg/ms
por aire a temperatura ambiente
dP = diámetro de la partícula

La carga *q* adquirida por la molécula, que es 1.6 x 10–19 C (coulomb). Por ejemplo, la carga de una partícula que adquiere 10 electrones sería: q = 10 x 1.6 x 10–19 C = 1.92 x 10–18 C

El número de electrones adquiridos depende de la intensidad de la corona que se genera cerca de los electrones y, a la vez, es proporcional al campo eléctrico. De esta forma, es proporcional a lo que hace que la fuerza eléctrica sea proporcional a E^2.

Esta es la fórmula para determinar la carga adquirida por una partícula:

$$q = \pi d_p^2 \varepsilon_0 \frac{3\varepsilon}{2+\varepsilon} E_{ch} \quad (5)$$

Donde
ε0 = 8,854 x 10–12 C2/Vm2. = permitividad del vacío
ε = 3.7 = constante dieléctrica de la partícula respecto al vacío
Ech = carga de la intensidad del campo en (V/m)

El rendimiento o eficacia es definido como el porcentaje de eliminación de partículas. Se puede hallar con la siguiente fórmula:

$$n = \frac{particulas\ removidas}{particulas\ de\ entrada} = \frac{C_{in} - C_{out}}{C_{in}} = 1 - \frac{C_{out}}{C_{in}} = 1 - \exp\left(-\frac{WeL}{U\,d}\right) \quad (6)$$

Dado que la velocidad de flujo U es el flujo volumétrico Q dividido por la sección de área transversal *Wd, también* se puede definir la eficacia así:

$$n = 1 - exp\left(-\frac{WeWL}{Q}\right) = 1 - \exp\left(-\frac{WeA}{Q}\right) \quad (7)$$

Donde el área de las placas es el producto del largo por el ancho: *A=WL.*

Las anteriores fórmulas nos permitieron determinar qué capacidad tiene el filtro electrostático depende del tamaño de las placas, la separación entre ellas y el voltaje, que debe ser seleccionado de acuerdo con el tamaño y tipo de las partículas.

2.3 Alarma

El sistema de alarma es agregado para la indicación de un nivel bajo de oxigenación con lo que da indicios de que hay contaminación en el quirófano. Para esta alarma se utilizaron sensores de gas (mq-x) los cuales tiene la capacidad de censar CO_2, nitrógeno, alcohol, monóxido de carbono, gas LP, formaldehido y dióxido de cloro.

Se utilizaron dos uno para medir únicamente el nivel de oxigenación que hay en el quirófano y otro para la medición de los gases ya mencionados, los valores se muestran en una LCD, la activación de la alarma depende de estos sensores los cuales están conectados a una placa arduino (anexo 1 código de los sensores).

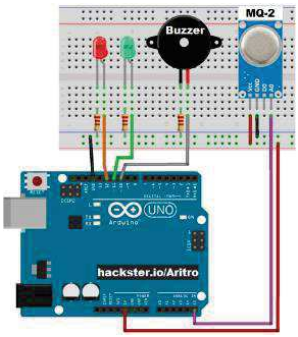


Figura 4. Conexión de la alarma

2.4 Ventiladores y placas del filtro electrostático.

El sistema de ventilación en los quirófanos (figura 2) está basado en la absorción del aire por ello antes de que el aire llegue al filtro se agregó un ventilador más fuerte para captar todo el aire del conducto y pasarlo por el filtro.

Las placas del filtro son metálicas de forma ovalada esto debido a que si se encuentra en forma rectangular la energía se concentra en las puntas por lo que las partículas que pasen por el centro no se eliminaran. De la manera ovalada la energía se dispersa en toda la placa creando el campo electrostático más efectivo (el filtro debe de estar casi en la parte superior de la salida del aire esto debido a que el campo eléctrico generado pueda dañar algún componente).

2.5 Componentes.

Para la elaboración del proyecto se realizó un modelo con los siguientes componentes:

- Flyback
- Sensor MQ-x
- Ventiladores de 12v
- Ventilador de corriente alterna
- Malla
- LCD
- Parlante
- Bomba de aire
- Placa Arduino UNO
- Led´s

Ingeniería y Automatización, Capítulo 12, pp. 147 - 158.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Figura 5: Flyback

El flyback es utilizado para generar alta tensión debido a su fácil uso, por ello se puede utilizar un circuito de alta tensión donde se puede modificar la frecuencia debido al uso de un transistor de lata ganancia.

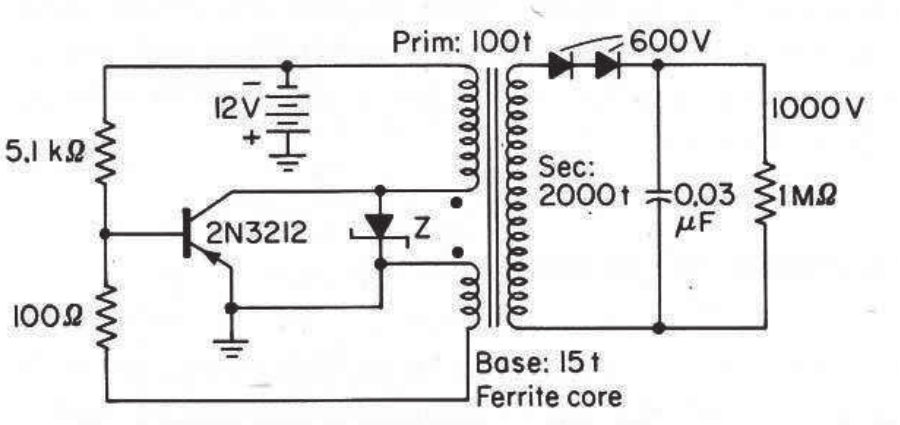


Figura 6: Circuito de alta tensión

El anterior circuito es de interés debido a que dependiendo de la frecuencia a la que se calibre será la distancia a la que el rayo de alta tensión pueda sobre salir (aunque en el proyecto no afecta debido al tamaña de este, es esencial tener en cuenta, debido a que teniendo una baja frecuencia mayor será la distancia del rayo y a una distancia menor a una alta frecuencia).

El sensor mq-x es implementado debido a que como ya se mencionó tiene la capacidad de detectar varios gases que son de interés en el quirófano. El sensor tiene un retardo de 2 a 5 minutos de respuesta esto se debe a su modo de funcionamiento ya que primero se tiene que llenar una pequeña sección del sensor con lo cual después ya empieza a mandar la señal en dado caso de que haya una detecciones algún gas.

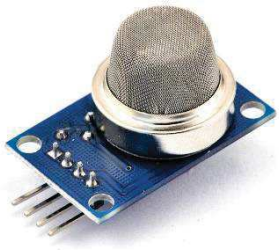


Figura 7: Sensor mq-x

Para el control del proyecto se utilizó una placa Arduino UNO, esto debido a la compatibilidad del sensor facilitando la detección de los gases en el quirófano, otro factor que en ventaja que nos brindó Arduino son los puerto que tiene con los cueles se pueden controlar los ventiladores (utilizando relevadores), la alarma y la LCD.

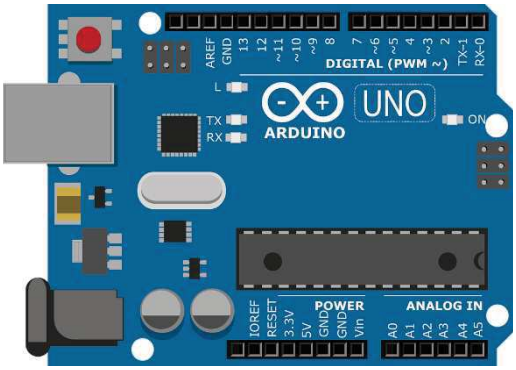


Figura 8: Placa Arduino UNO

2.6 Tiempo de funcionamiento.

Su función principal como se abordó al inicio es la eliminación de partículas mediante alta tensión, por lo que es necesario la creación de esta, y aunque el flyback nos brinda esta alta tensión, el embobinado y los componentes del circuito empiezan a calentarse conforme para el tiempo lo cual en cierto momento puede llegar a descomponerse algún componente o el flyback por lo que se agregó un disipador de calor, esto da beneficio ya que delimita la temperatura hasta cierto grado en el componente.

Por el calor generado en los componentes se puede ver que la duración de este filtro ronda las 3 a 4 horas (con un sistema de enfriamiento para el transformador de alta tensión no tendría establecido un límite establecido de funcionamiento).

El transistor usado como es de alta ganancia tiene la capacidad de aguantar esa temperatura pero para facilitar su modo de empleo se agregó un disipador de calor en conjunto con pasta térmica.

2.7 Oxigenación.

El proceso de oxigenación es necesario en el quirófano ya que el modo de filtración no es de manera circular no retroalimenta por así decirlo al quirófano, para ello se agregó una bomba de aire que oxigenara al quirófano teniendo de esta manera aire limpio de forma continua y contante, permitiendo así liberar el aire que pasa por el filtro de manera libre y sin partículas o contaminación que afecte al medio ambiente.

Figura 9: bomba de aire implementada al prototipo

3. Resultados.

El presente prototipo se realizó tratando de simular un cuarto cerrado como si fuera un quirófano de tal manera que se pudiera comprobar el funcionamiento del filtro electrostático realizando pruebas de gases y partículas teniendo en cuenta la calidad de oxigeno que hay dentro del cuarto.

Figura 10: prototipo

El prototipo consiste en el diagrama ya descrito (figura 3), para ello el diseño del prototipo fue realizado de tal manera que permitiera un encierro total del aire que está dentro, la puerta se realizó de manera deslizable para evitar la creación de corriente s de aire, de esta manera se aseguró que no hubiera alguna interacción externa con lo interno.

En el caso del conducto de ventilación se pusieron dos ventiladores (ventiladores de 12V DC) en los extremos para poder absorber el aire dentro del quirófano, de esta manera se conduce a otro ventilador (ventilador de 127 AC) que permite enviar todo el aire absorbido al filtro electrostático.

Figura 11: ventilador

Dentro del prototipo se agregaron dos sensores de gas uno para determinar la calidad de oxígeno y otro para la detección de otros gases, estos están conectados a la placa Arduino que a su vez al ser accionados el arduino se encarga de la alarma.

Figura 12: sensores de gas

En la parte superior del prototipo es donde se ubica un ventilador, las placas del filtro y una malla para la retención de partículas más grandes.

Figura 13: conducto del filtro

Teniendo todo el prototipo se realizaron pruebas con los siguientes compuestos:

- Alcohol
- CO_2
- Gas L.P.
- Cloruro
- Formaldehido
- Macropartículas (residuos de la quema de una hoja de papel)

Algunos de estos compuestos son empleados para la limpieza y esterilización del quirófano por ello es necesario saber la respuesta del sistema, algunos otros representan partículas que pueden estar presentes en un hospital y su salida del este puede representar una contaminación al medio ambiente.

Tabla 1: respuesta del filtro

Compuestos	Tiempo de respuesta del sensor	Calidad de aire	Tiempo de respuesta de reducción de contaminación	Calidad de aire final
Alcohol	3 minutos	63%	4-5 minutos	92%
CO_2	3 minutos	57%	7 minutos	93.6%
Gas L.P.	2 minutos	36%	6 minutos	83%
Cloruro	3 minutos	59%	5 minutos	91%
Formaldehido	3-4 minutos	57%	6-7 minutos	93%
Macropartículas	4-5 minutos	77%	8 minutos	93%

Los resultados expresados en la tabla nos indican que los sensores tiene un retardo promedio de 3.2 minutos de respuesta, la calidad del aire baja alrededor de 58.16% al detectar contaminación y el promedio de respuesta para la reducción de esta es de 6. 16 minutos llegando a una calidad mayor igual a 90% de calidad de aire el cual era el objetivo de este filtro.

Tabla 2: respuesta del filtro a la salida

compuestos	Calidad de aire final
Alcohol	86%
CO_2	84%
Gas L.P.	76%
Cloruro	86%
Formaldehido	89%
Macropartículas	84%

En el caso de la salida se puede ver que el promedio de calidad del aire es de 84.16%, si lo comparamos con el nivel normal de aire exterior es de alrededor de 82% a 88% por lo cual está dentro del rango de esta manera se están cumpliendo dos propósitos, tener una calidad de aire estable en el quirófano y evitar un grado de contaminación ambiental grave.

4. Conclusiones.

El diseño y la elaboración del prototipo dieron los resultados deseados, esto se debe a que los resultados están dentro de los parámetros establecidos y dentro de los objetivos planteados. Al realizar las pruebas se podía notar que las variaciones tenían un pequeño marguen de error pero aun así están dentro de los parámetros. El prototipo fue implementado satisfactoria mente, teniendo ventajas de que el prototipo necesito materiales con fácil acceso y de un costo no muy elevado.

El prototipo se puede considerar como un sistema de instrumentación por su funcionamiento y respuesta. Se pudo dar cuenta de que este prototipo tiene una funcionabilidad notable ya que el campo electromagnético generado tiene la capacidad de atraer las partículas deseadas y aun en la prueba de macropartículas se esperaba que la malla las detuviera pero el campo generado fue quien detuvo estas partículas de esta manera se pudo percatar de la gran potencia que tiene el campo generado que a su vez indica que el rayo de alta tensión está separando las partículas para que estas sean liberadas de manera individual.

5. Anexos

Código elaborado para la placa Arduino UNO:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(8, 9, 10, 11, 12, 13);
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  lcd.begin(20, 4);
  pinMode(6,OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
  analogWrite(6,0);
  int sensor1=analogRead(A0);
  int sensor2=analogRead(A1);


  float calidad1=100-(sensor1/10.23);
  float calidad2=100-(sensor2/10.23);

  lcd.setCursor(0,0);
  lcd.print("Nivel O2: ");
```

```
            lcd.print(calidad1);
            lcd.print("%");
            lcd.setCursor(0,1);
            lcd.print("Nivel OG: ");
            lcd.print(calidad2);
            lcd.print("%");
            lcd.setCursor(0,4);
            lcd.print("CALIDAD ESTABLE");

            if(calidad1<=60 or calidad2<=60){
              lcd.setCursor(0,0);
              lcd.print("Nivel O2: ");
              lcd.println(calidad1);
              lcd.setCursor(0,1);
              lcd.print("Nivel OG: ");
              lcd.println(calidad2);
              lcd.setCursor(0,4);
              lcd.print("ALARMA CALIDAD BAJA");
              for(int g; g<=255;g++){
                analogWrite(6,g);
                delay(3);
              }

              }

    }
```

Referencias.

[1]. Fernández Sandoval, E. (2012). Mejoras en la eficiencia de los colectores de polvo tipo Jet Pulse y precipitador electrostático (tesis de grado). Piura (Perú): Universidad de Piura, Facultad de Ingeniería, p. 81.

[2]. Sokhi, R. S. (2005). Air pollution control engineering. En L. Wang, N. Pereira y Y. Hung (eds.) *Handbook* of environmental engineering (vol. 1). New York: Humana Press.

[3]. Turner, J., Lawless, P., Yamamoto, T., Coy, D. y Research Triangle Institute (1999). Precipitadores electrostáticos. United States: Environmental Protection Agency, p. 9.

[4]. Rodríguez, J. F., Alarcón, U. F., García Bedoya, O. y Benavides, G. (2017). Diseño y prototipo de un precipitador electrostático.

Ingeniería y Automatización, Capítulo 13, pp. 159 - 175.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Implementación de Algoritmos para el Monitoreo de Puentes en Tiempo Real

Martínez Trujano Luis Álvaro, Hernández Figueroa Jorge Alberto, Gasca Zamora Héctor Miguel, Quintana Rodríguez Juan Antonio y Hernández Guzmán Andrés

Instituto Mexicano del Transporte
Km 12+000, Carretera Estatal N° 431 "El Colorado-Galindo", Parque Tecnológico San Fandila, Mpio. Pedro Escobedo, Querétaro, C.P. 76703, Tel. (442) 216 9777
amartinez@imt.mx

Resumen

Los puentes están expuestos durante su vida útil a numerosos factores que afectan su funcionalidad, tales como terremotos, huracanes, interacciones viento-estructura o deterioro de sus elementos por fatiga, por lo tanto, se vuelve prioritario valorar su condición bajo estándares de servicio y seguridad. La importancia de la seguridad de los usuarios y los bienes transportados hace necesario desarrollar estrategias para la conservación y mantenimiento de la infraestructura vial. El Instituto Mexicano del Transporte se ha preocupado por establecer estrategias para detectar oportunamente, el daño estructural y alargar la vida útil de los puentes más importantes en nuestro país mediante la instrumentación y el monitoreo permanente de los parámetros estructurales críticos en los puentes. El demandante procesamiento de grandes volúmenes de datos durante el monitoreo, exige el desarrollo de algoritmos que detecten en tiempo real eventos inusuales, además de pronosticar mediante la tendencia del comportamiento a largo plazo, futuros daños en la estructura, para actuar de manera oportuna y conservar la integridad tanto de los puentes como de sus usuarios. Utilizando la plataforma de desarrollo LabVIEW se han implementado algoritmos para ayudar en el monitoreo de los puentes más importantes de México, comenzando con el puente Río Papaloapan. Con la aplicación de los algoritmos se han caracterizado situaciones especiales que afectan el desempeño normal de los puentes, tales como vehículos con peso mayor al promedio de los que circulan por el puente, actividades de reencarpetamiento de la calzada del puente, cierres viales de los cuerpo del puente, falla y reemplazo de elementos de anclaje de los tirantes, redistribución de cargas de tensión en tirantes aledaños, ciclos de tensión-compresión por día y años en las vigas principales debido a efectos térmicos (fenómenos de estacionalidad verano-inverno), así como tendencias globales del comportamiento de la estructura. Por ello, es importante preservar una estrategia integral que conjugue el monitoreo de cada puente para generar una base de información lo suficientemente robusta con la cual se pueda predecir el comportamiento de la estructura ante diversas condiciones.

Palabras clave: Algoritmos, monitoreo remoto, instrumentación, sensores FBG, LabVIEW, CeMPEI.

1. Introducción

El Centro de Monitoreo de Puentes y Estructuras Inteligentes (CeMPEI), creado por el Instituto Mexicano del Transporte (IMT), tiene el objetivo de evaluar la estructura de los puentes mexicanos instrumentados para la Secretaría de Comunicaciones y Transportes (SCT), para alcanzar este objetivo es necesario analizar en tiempo real la información de los sensores instalados para la detección oportuna de eventos o tendencias inusuales que se manifiesten en un comportamiento estructural atípico o degeneren la condición estructural de los puentes [1].

Actualmente, en el alcance del CeMPEI se incluyen los puentes Tipo: El Caracol, Papaloapan, Nautla, Raudal y Buenavista; así como los puentes Especiales: Chiapas, Río Papaloapan, El Carrizo y Mezcala.

El puente Chiapas fue instrumentado en 2008 por el Instituto de Ingeniería de la UNAM (Universidad Autónoma de México) mediante una red de sensores de deformación tipo FBG, interrogador FBG y una computadora.

El puente Río Papaloapan fue el primer puente instrumentado por el IMT, en el año 2012. La instrumentación se dividió en dos etapas. En la primera etapa se instaló el sistema eléctrico, para lo cual se colocaron paneles fotovoltaicos, así como baterías de ciclado profundo. La segunda etapa consistió en la instalación del sistema de adquisición de datos, ello comprendió la colocación de 62 sensores de fibra óptica con tecnología FBG, un interrogador-multiplexor FBG, dos cámaras de video, una estación sísmica y una estación climatológica.

En 2012 se instrumentaron a través de la UNAM, cinco puentes Tipo: El Caracol, Papaloapan, Nautla, Raudal y Buenavista. La instrumentación se realizó en dos etapas. La primera consistió en la instalación de sensores de deformación tipo extensómetro. En la segunda etapa se instaló el sistema de adquisición de datos [2].

El puente Mezcala se instrumentó entre 2015 y 2016 por el IMT, con sensores FBG: 36 acelerómetros, 16 inclinómetros, 10 extensómetros de concreto, 48 extensómetros de acero, 11 sensores de temperatura, un medidor de desplazamiento; adicionalmente se instalaron 2 cámaras de video, una estación climatológica, un interrogador, un multiplexor y una unidad de control. Para el sistema eléctrico, se instalaron paneles solares, y baterías [3].

El último puente instrumentado por el IMT es el puente El Carrizo, a finales del primer trimestre del 2018. Se colocaron sensores FBG: 16 extensómetros distribuidos en ocho dovelas, dos por cada una, además de cuatro inclinómetros y cuatro sensores de temperatura.

Uno de los retos del CeMPEI es el análisis en tiempo real de las señales provenientes de los sensores instalados, ya que implica el manejo de grandes volúmenes de datos a los cuales se les debe dar una interpretación física para poder evaluar la condición estructural de los puentes.

2. Diseño y desarrollo de algoritmos para el monitoreo de puentes

El paradigma clásico del monitoreo estructural consiste en una cadena de procedimientos y algoritmos, partiendo de la instalación de sensores en las estructuras, con los que se realiza la medición de algunas variables físicas como aceleración, desplazamiento y/o inclinación, por nombrar algunas. Estos sensores vienen acompañados de sistemas de adquisición y procesamiento de datos que capturan las respuestas dinámicas de las variables físicas y permiten analizarlas mediante programas computacionales especialmente diseñados para detectar patrones y cambios en el comportamiento normal de las señales [4].

Un algoritmo se define como el método que se requiere para resolver algún problema [5]. De manera general, un algoritmo debe estar constituido por los siguientes elementos característicos: información de entrada, número finito de instrucciones bien definidas e información de salida (solución).

El diseño de los algoritmos para una aplicación de monitoreo depende de la cantidad y la complejidad de los datos de medición, la confiabilidad real del conjunto de sensores y del sistema de adquisición de datos. Las características de la adquisición de datos del sistema de monitoreo a menudo introducen ciertas particularidades en los datos, éstas pueden afectar aún más la disponibilidad de los datos recolectados para su posterior análisis. Por lo tanto, se desarrolló un conjunto de algoritmos con un enfoque dedicado a garantizar la calidad de los datos, en donde, son

implementados en múltiples niveles. En la figura 1 se observa la jerarquía de los algoritmos utilizados en el CeMPEI para el monitoreo de puentes.

A continuación, se describe de manera general cada uno de los algoritmos desarrollados en el monitoreo de puentes.

1 Comunicación: Es el conjunto de algoritmos que tienen como objetivos, decodificar los datos, así como establecer y mantener el flujo de información entre los sensores instalados en el puente con los algoritmos de manejo de información.

2 Detección: Estos algoritmos son capaces de extraer información puntual del monitoreo, ya sea para la creación de una base de información histórica o bien, para la identificación de parámetros estructurales que se encuentren fuera de los límites de operación normal del puente. Los algoritmos de detección son los precursores de los algoritmos de almacenamiento de información, de generación de reporte y de envío de reporte.

3 Manejo de información: Son algoritmos que tienen la función de asegurar la calidad de los datos adquiridos, así como recolectar los datos de interés cuando se presente un evento atípico, además, validan automáticamente el tiempo y valores dentro de rangos de medición esperados, buscando eliminar lecturas falsas de las señales de los sensores.

4 Visualización: Estos algoritmos permiten desplegar en gráficas de tiempo real, las amplitudes registradas en cada uno de los sensores instalados para que el personal encargado del monitoreo observe y relacione los valores mostrados con el comportamiento físico de la estructura.

5 Almacenamiento de información: Se encargan de comprimir y guardar los datos relevantes del monitoreo, como archivos de datos y video, en el ordenador principal. Lo anterior, para generar una base de información que permita comprender la naturaleza de los fenómenos de interés.

6 Generación de reporte: Su función es recabar los datos de relevancia, instantes después de registrarse un evento atípico, con ello, generar una gráfica en archivo de Microsoft Excel que incluye la información del sensor que registró los valores fuera del límite de operación normal.

7 Envío de información: Este algoritmo permite transferir los datos de interés al personal encargado del monitoreo, a través de correo electrónico. Esto, con el fin de conocer la condición actual de la estructura.

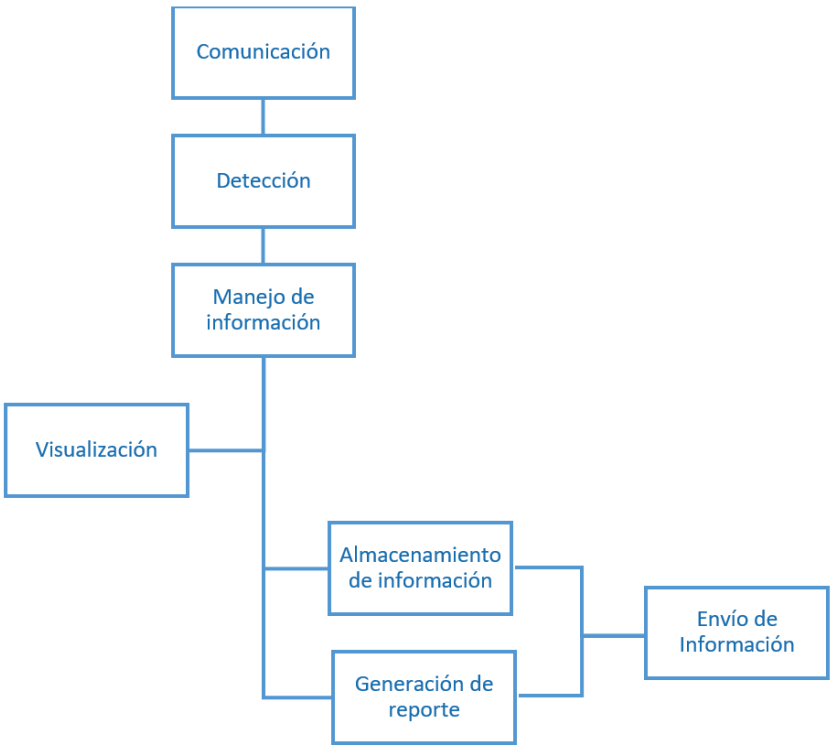


Figura 1. Algoritmos implementados en el monitoreo remoto de puentes.

El conjunto de algoritmos genera información relevante en tiempo real para crear una base histórica que permita detectar en cualquier instante de tiempo algún evento o tendencia inusual en los parámetros estructurales monitoreados. Con ello, realizar las actividades pertinentes para preservar el puente en condiciones aceptables de operación.

3. Algoritmos para detección

Con los extensómetros se realiza la medición, de la frecuencia natural y de los modos de vibración del puente, la detección del eje neutro para vigas tipo cajón, la medición del valor de referencia de deformación y amplitud de deformación del puente, así como la velocidad y dirección de los vehículos que circulan por el puente. El uso de inclinómetros y sensores de temperatura permiten conocer los valores de referencia de inclinación y temperatura, respectivamente. Por último, con los acelerómetros colocados en los tirantes se detecta la frecuencia natural de vibración para el cálculo de la tensión axial. Por tal razón, los algoritmos de detección se convierten en la columna vertebral del monitoreo de estructuras.

A continuación, se explican dos de los parámetros estructurales monitoreados por el CeMPEI, que evidencian la integridad del puente al utilizar extensómetros: el valor de referencia de deformación y las amplitudes de deformación por la carga viva (tránsito vehicular), además de la tensión axial del tirante, al emplear acelerómetros en los tirantes. Con ello, en el momento en que alguno de estos parámetros se encuentre fuera de los límites de operación normal, definido para cada uno de ellos, se detectará y realizaran las acciones pertinentes para atender la situación.

3.1 Detección del valor de referencia de deformación

Las variaciones del valor de referencia de deformación pueden ser causadas por tres fenómenos:

1. Efecto térmico: En el análisis de las estructuras se toman en cuenta los esfuerzos, las deformaciones y los desplazamientos que en ella se originan por las variaciones de temperatura en sus elementos estructurales [6]. Este tipo de cambios, presentan un comportamiento cuasi-estáticos, en general están asociados a pequeñas variaciones del valor de referencia de deformación durante el transcurso del día. A largo plazo, se comporta como una variación periódica y predecible en el periodo anual.

2. Mantenimiento: Se detectan modificaciones en el valor de referencia de deformación al realizar actividades de mantenimiento a elementos estructurales y no estructurales; tales como modificaciones en la carpeta asfáltica en la calzada, guarniciones, banquetas, parapetos y elementos de instalaciones de servicio público.

3. Daño estructural: Se presenta como un cambio permanente en el valor de referencia de deformación, se perciben como cambios instantáneos de gran magnitud.

El algoritmo diseñado para detectar el valor de referencia de deformación se muestra en la figura 2. De manera general, el algoritmo adquiere el valor instantáneo de deformación, el cual se almacena temporalmente en un búfer de datos, en seguida se calcula el promedio de los datos del búfer por cada nuevo valor que se almacena. Al transcurrir el tiempo, se incrementa la cantidad de datos almacenados consiguiendo un valor confiable para la referencia de deformación. Lo anterior, se replica para cada uno de los extensómetros instalados en el puente.

Figura 2. Algoritmo de detección del valor de referencia de deformación.

3.2 Detección de deformación por carga viva en el puente

En el caso de los extensómetros, además del valor de referencia de deformación se obtiene el valor de deformación por carga viva asociada al flujo vehicular. Este tipo de cargas vivas son fuerzas variables generadas por el tránsito de camiones, autobuses, automóviles, equipos para construcción y toda aquella carga que circule por la calzada del puente. En las vigas del puente el efecto de la carga viva se aísla en dos fenómenos: tensión y compresión en la zona instrumentada.

Por cada valor de deformación registrado, este algoritmo, ver figura 3, extrae el valor de referencia de deformación de manera instantánea, ejecutando el algoritmo mostrado en la figura 2. Por lo que los valores de amplitudes de deformación están exclusivamente asociados al efecto dinámico de la carga viva que transita en ese momento. En seguida, se aplica un filtro a la señal de deformación, para remover las señales de alta frecuencia que no aportan información relevante, y con ello, obtener valores confiables en las amplitudes de deformación. Finalmente, tanto para las cargas vivas a tensión como las cargas vivas a compresión, la señal filtrada se compara con los límites de deformación por carga viva y con los límites normales de operación por carga viva. Con el límite de deformación por carga viva se detectan todos los valores de deformación por carga viva que transitan

en la zona de la viga instrumentada. El límite normal de operación por carga viva se emplea para detectar únicamente las amplitudes de deformación atípicos (eventos extraordinarios) que se encuentren afuera del 98.8% del percentil de los datos históricos registrados para ese sensor. Así, el algoritmo es capaz de detectar de manera simultánea las amplitudes de deformación en cada sensor, por los efectos de las cargas vivas que transitan.

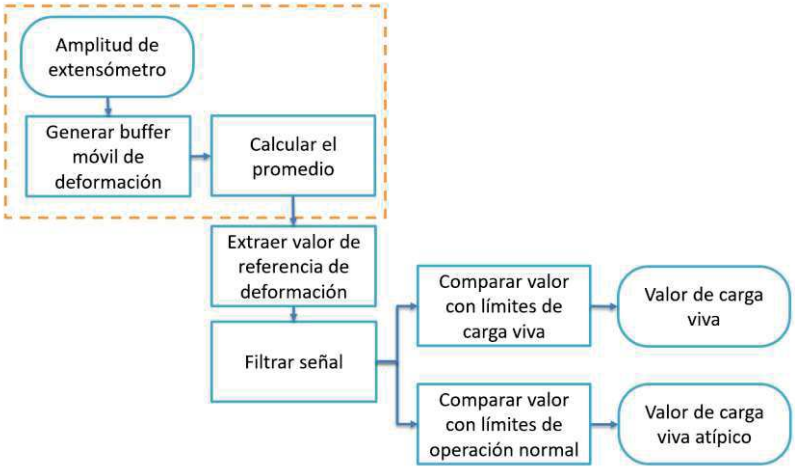


Figura 3. Algoritmo para detección de carga viva.

3.3 Detección de tensión axial en cables para puente atirantado

El algoritmo adquiere el valor instantáneo de aceleración, el cual se almacena temporalmente en un búfer de datos, en seguida se calcula el promedio de los datos del búfer por cada nuevo valor que se almacena. Al transcurrir el tiempo, se incrementa la cantidad de datos almacenados consiguiendo un valor confiable para extraer el offset de la señal de entrada y con ello, eliminar las frecuencias bajas asociadas al efecto térmico en el acelerómetro. Lo anterior, se realiza para cada acelerómetro instalado en el centro del tirante.

Posteriormente, se aplica un filtro pasa baja para eliminar el ruido que tenga la señal y sólo visualizar las frecuencias relativas a la vibración libre del tirante. Ya que los acelerómetros registran los datos en múltiplos o submúltiplos de la aceleración de la gravedad (g, equivalente a 9.81 m/s^2). Para obtener un valor de tensión asociado a cada cable de un puente atirantado, se requiere un análisis dinámico en el dominio de la frecuencia, por lo que se emplea la Densidad Espectral de Potencia, PSD por sus siglas en inglés, para pasar del dominio del tiempo al dominio de la frecuencia [7].

Lo siguiente es determinar la frecuencia fundamental de vibración, así como sus armónicos. En general, la frecuencia encontrada en cada uno de los tirantes instrumentados corresponde al primer modo de vibración, debido a que el sensor se ubica en el punto medio de la longitud del tirante, el instrumento es más sensible a registrar la frecuencia correspondiente al primer modo, así como los modos impares de vibración.

Adicionalmente, se utiliza un modelo lineal (ecuación 1) para obtener la tensión axial a la que está sometido el tirante a partir de la frecuencia fundamental de vibración.

$$f_n = \frac{n}{2l}\sqrt{\frac{T}{m}} \tag{1}$$

Donde *n*, es el número asociado al modo de vibración, *l* es la longitud del cable, *T* es la tensión del tirante y *m* es la masa por unidad de longitud del cable. Por lo tanto, la tensión del tirante se obtiene de la ecuación 1, la cual puede ser calculada en función del primer modo o como el promedio de dos o más modos de vibrar del tirante [8].

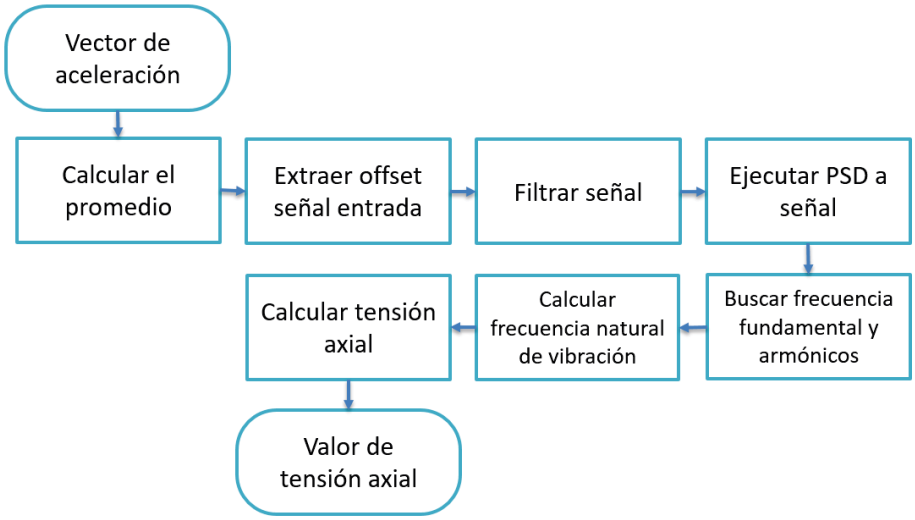


Figura 4. Algoritmo para detección de tensión axial en cables.

4. Aplicación de algoritmos para detección en el monitoreo del puente Río Papaloapan

4.1 Plataforma para implementación de algoritmos: LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un sistema de desarrollo basado en programación gráfica orientado a desarrollar aplicaciones para instrumentación que integra una serie de librerías para comunicación con instrumentos electrónicos como GPIB, RS232 ó RS485 con tarjetas de adquisición de datos, sistemas de adquisición y acondicionamiento como VXI o SCXI, comunicaciones en redes TCP/IP, UDP, o en los estándares de software COM, OLE, DDE, DLL o ActiveX para Windows, así como AppleEvents para MacOS o PIPE para UNIX [9].

Las principales características de los Instrumentos Virtuales (VIs por sus siglas en inglés), se describen a continuación: Los VIs contienen una interface interactiva de usuario, la cual se llama panel frontal, ya que simula el panel de un instrumento físico. Se pueden introducir datos usando el teclado o el *mouse* y tener una visualización de los resultados en la pantalla de la computadora. El Panel Frontal es la interface hombre-máquina de un VI. Los VIs reciben instrucciones de un diagrama de bloques construido en lenguaje G, el cual suministra una solución gráfica a un problema de programación. El diagrama de bloques es el código fuente de un VI. Usan una estructura hereditaria y modular que permite realizar programas por niveles o hacer programas con otros programas o subprogramas. Un VI contenido en otro VI es denominado subVI. Todo VI se puede convertir en subVI sin ningún tipo de cambio en su estructura. Con estas características LabVIEW permite dividir un programa en una serie de tareas las cuales son divisibles nuevamente hasta que una aplicación complicada se convierte en una serie de sub-tareas simples. Es por ello que se le considera programación modular.

Las principales ventajas en el uso de LabVIEW son: su conectividad con instrumentos, capacidad de adquisición de datos, flujo de datos natural basado en una interface gráfica de programación, escalabilidad, además del complemento de funciones pre-configuradas.

LabVIEW implementa análisis y rutinas matemáticas que trabajan en conjunto de forma nativa con funciones de adquisición de datos y capacidades de despliegue de información, de forma que pueden ser construidos dentro de cualquier aplicación. Adicionalmente, LabVIEW ofrece rutinas de análisis para ejecuciones de punto por punto, estas rutinas están diseñadas específicamente para cumplir las necesidades de análisis en línea en aplicaciones de tiempo real.

4.2 Puente Río Papaloapan

La estructura en donde se evaluaron los algoritmos de detección es el puente Río Papaloapan, el cual está ubicado en el kilómetro 85 + 980 de la autopista la Tinaja-Acayucan, en el estado de Veracruz, fue construido en el año de 1994 y se puso en servicio en el año de 1995, es del tipo atirantado con un claro de 203 m y una longitud total de 407 m. El puente tiene 112 cables distribuidos en 8 semi-arpas con 14 cables cada una, ver figura 5.

Figura 5. Puente Río Papaloapan.

El puente Río Papaloapan tiene instalados sensores y sistemas de adquisición de datos basados en tecnología de fibra óptica, en específico rejillas de Bragg, ver figura 6. A esta tecnología se le conoce como FBG (*Fiber Bragg Grating* por sus siglas en inglés). Las principales características de los sistemas basados en fibra óptica son:

- Inmunidad al ruido electromagnético
- Bajo consumo energético
- Capacidad de instalar varios sensores en un único lazo
- Larga estabilidad de medición a lo largo del tiempo.

A continuación, se presenta en la tabla 1, información relevante tanto de los equipos de fibra óptica como de los sensores elegidos para la instrumentación del puente.

Tabla 1. Equipos y sensores de fibra óptica instalados en el puente Río Papaloapan.

Cantidad	Nombre	Descripción	Marca	Modelo

1	Interrogador	Equipo de adquisición de datos de fibra óptica de 4 canales con frecuencia de adquisición máxima de 1000 Hz.	Micron Optics	sm-130
1	Multiplexor	Equipo que multiplexa los canales del interrogador para conseguir 16 canales.	Micron Optics	sm-041
1	Sensor de desplazamiento	Sensor empleado para medir desplazamientos.	Micron Optics	os5100
5	Sensores de temperatura	Sensor empleado para medir cambios de temperatura.	Micron Optics	os4350
8	Inclinómetros	Sensor empleado para medir inclinaciones.	FBG TECH	T1-310
24	Acelerómetros	Sensor empleado para medir vibraciones.	Micron Optics	os7100
24	Extensómetros	Sensor empleado para medir deformaciones unitarias en concreto.	Micron Optics	os3600

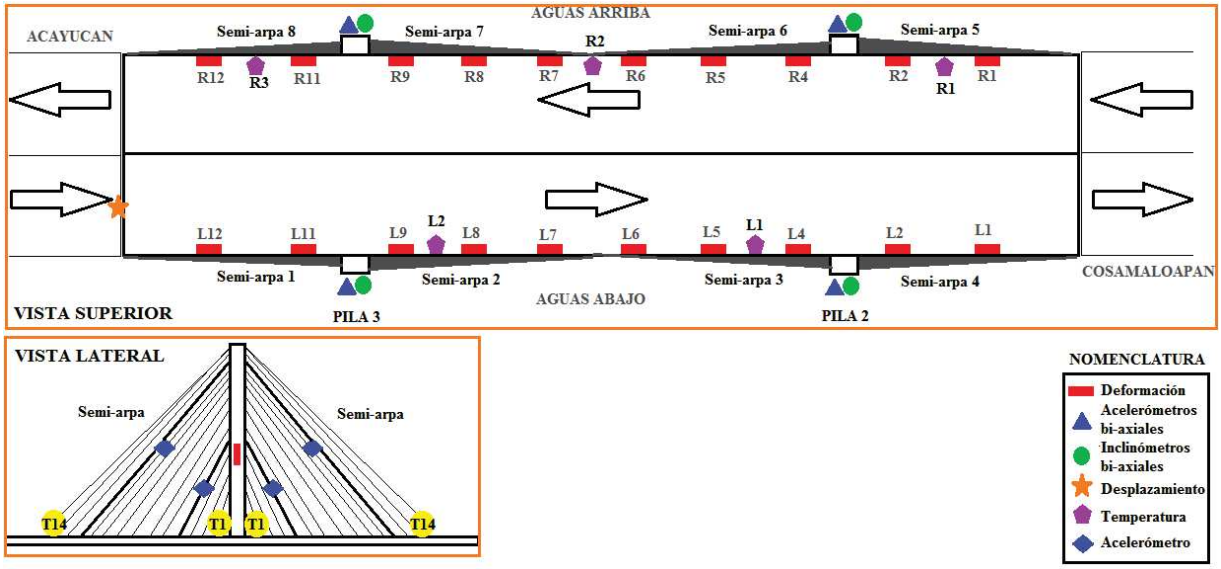


Figura 6. Diagrama de instrumentación puente Río Papaloapan.

4.3 Detección del valor de referencia de deformación

En el caso del puente Río Papaloapan, de los 62 sensores instalados, 24 corresponden a extensómetros, para los cuales se calculan sus respectivos valores de referencia de deformación. Con el promedio de 120 segundos de información en el búfer de datos se obtiene un valor confiable de la referencia de deformación, equivalente a 7500 datos de adquisición continua ya que el sistema registra datos cada 0.016 segundos. El valor de referencia de deformación obtenido, representa 1/720 valores que se generan en el transcurso de 24 h. En la figura 7, se muestra en color gris el conjunto de valores del búfer móvil de deformación correspondientes al extensómetro L9 a las 8:32 a.m. del 22 de abril de 2018. Así, el valor de referencia de deformación para ese instante de tiempo es -45.10 με, representado con una línea horizontal segmentada en color azul.

Calculando los valores de referencia de deformación cada 2 min, se da seguimiento al comportamiento del sensor y, a su vez, se determina el fenómeno que provoca dichos cambios.

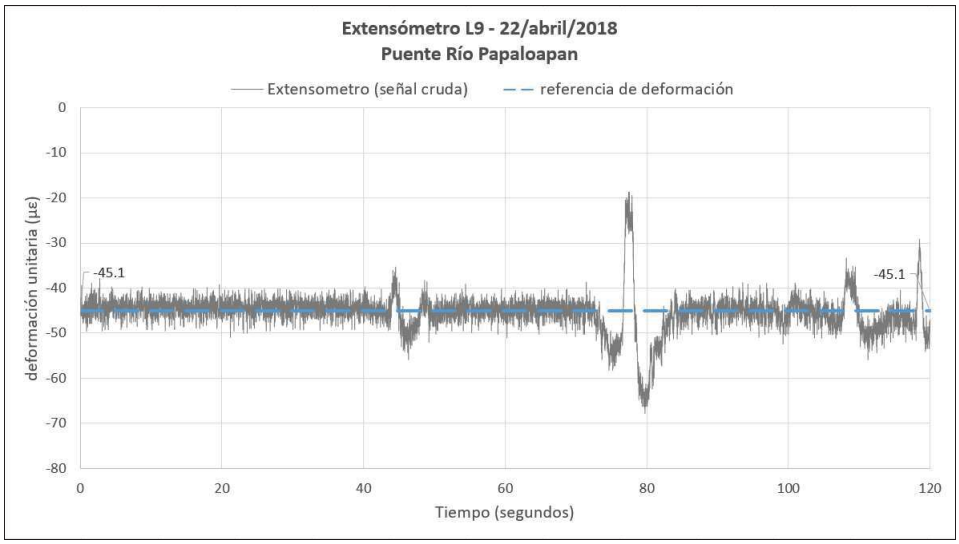


Figura 7. Referencia de deformación del extensómetro L9.

La figura 8 muestra los 720 valores de referencia de deformación del extensómetro L9 para el día 22 de abril de 2018. Se observa en los cambios de referencia de deformación que el comportamiento se debe únicamente al efecto térmico. En donde, la zona instrumentada de la viga principal se comprime, al incrementarse la temperatura por efecto del sol, alcanzando una compresión máxima en el día, posteriormente se presentan esfuerzos a tensión, conforme disminuye la temperatura. Al final, se efectúa un ciclo térmico por cada día.

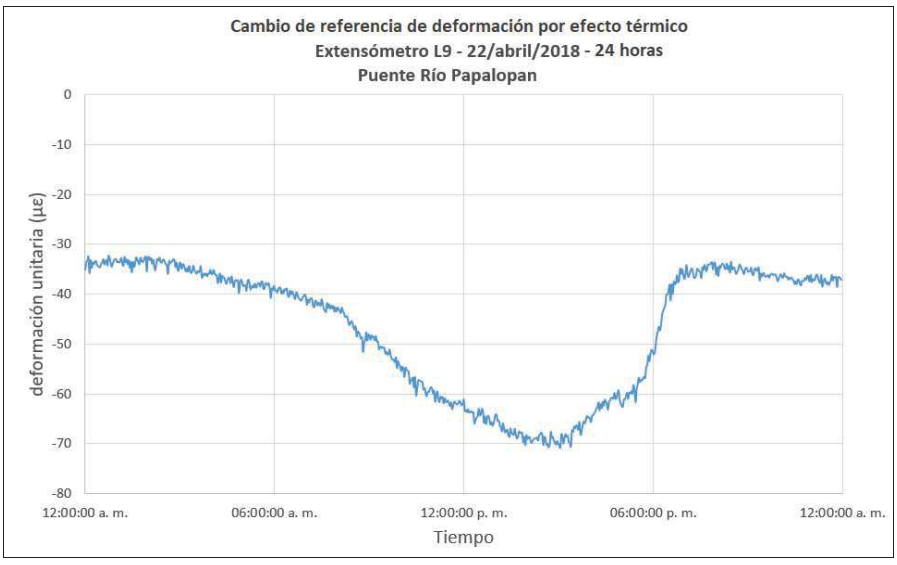


Figura 8. Comportamiento de la referencia de deformación del extensómetro L9 en el día 22/abril/ 2018.

4.4 Detección de deformación por carga viva

Continuando con el caso del puente Río Papaloapan, en la figura 9 se observan dos minutos de datos continuos de deformación en el extensómetro L9, al cual se sustrajo el valor de referencia de deformación para detectar las amplitudes absolutas por efectos de las cargas vivas.

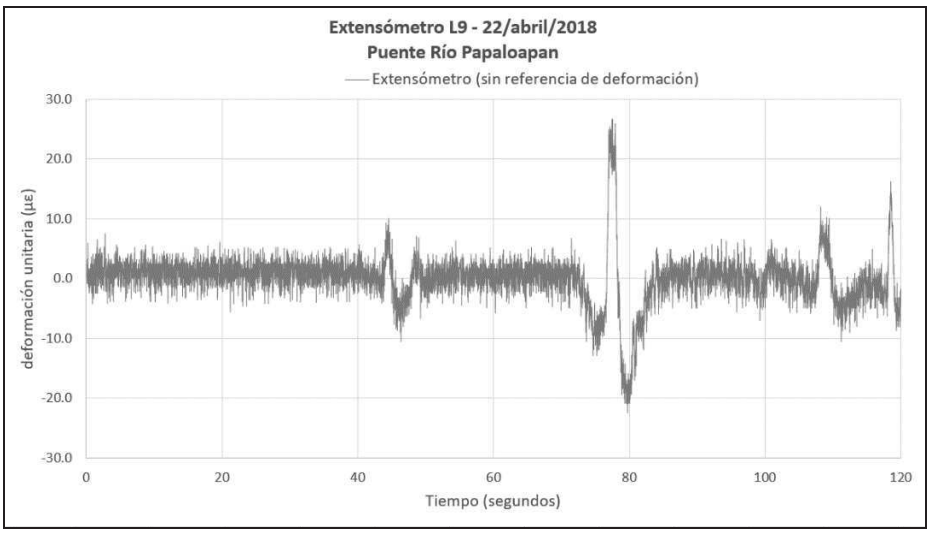


Figura 9. Deformación en el extensómetro L9 sin el valor de referencia de deformación.

Ahora, se comparan los valores de deformación con los límites de detección de carga viva, tal como se muestra en la figura 10. En donde, la deformación del extensómetro L9, graficado en gris, se le aplicó un filtro de señal. El límite de deformación a tensión, línea segmentada color verde claro, permite detectar cuatro amplitudes, siendo la máxima de 23.40 με. En el caso del límite de deformación a compresión, línea segmentada color verde obscuro, se detectan cinco valores de amplitud, en donde la máxima es de -18.90 με.

También, el algoritmo detecta cuando la amplitud registrada se encuentra fuera del límite de operación normal. En la figura 10, se muestra un evento extraordinario de deformación a tensión por carga viva de 23.40 με que se localiza fuera del límite de operación normal, línea segmentada color naranja, siendo el límite a tensión 20.00 με para el extensómetro L9. En el caso de la deformación a compresión por carga viva la máxima amplitud está por debajo del límite de operación normal, -20.00 με.

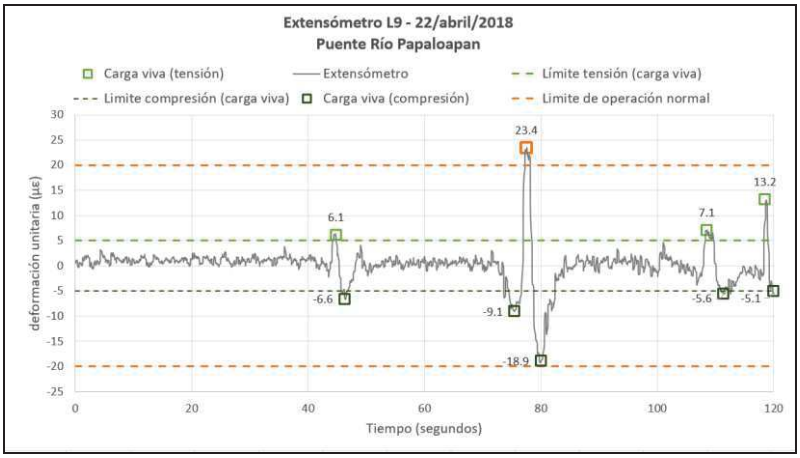


Figura 10. Detección de amplitudes de deformación por carga viva.

4.5 Detección de tensión axial en tirante

En el puente se instrumentaron los tirantes N° 4 y N° 11 de cada una de las ocho semi-arpas, siendo un total de 16 acelerómetros instalados al centro de cada cable. Para cada sensor, se agrupan los datos continuos de aceleración por periodos de 2 min, en seguida se remueve el offset por deriva térmica a la señal de aceleración, ver figura 11, para posteriormente aplicar una función de Densidad Espectral de Potencia, así procesar la señal en el dominio de la frecuencia.

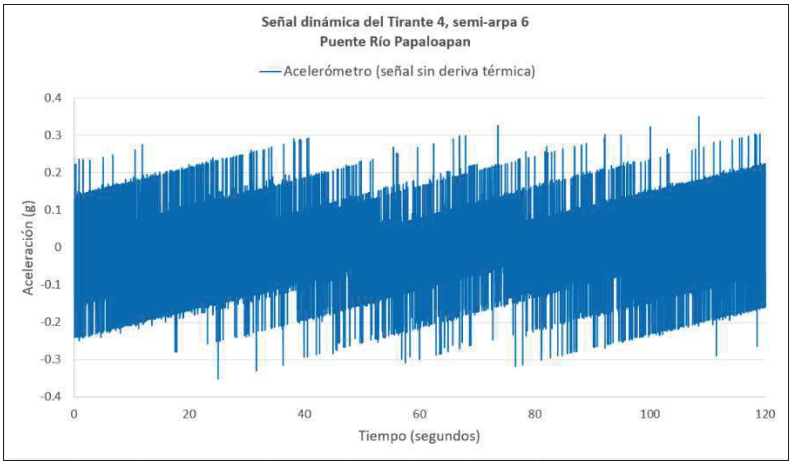


Figura 11. Detección de amplitudes de deformación por carga viva.

Con base en el estudio experimental realizado en 2003, para determinar la frecuencia natural de vibración de los 112 tirantes del puente Río Papaloapan [8], se crean: un umbral de detección de frecuencia para cada tirante instrumentado (tres primeros modos de vibración) y un umbral de amplitud basado en 4 σ del promedio de amplitudes del PSD. Con ello, el algoritmo detecta la frecuencia fundamental del tirante. En la figura 12 se muestra la detección de frecuencia del tirante 4 de la semi-arpa 6 correspondiente a 2 minutos continuos, se observa el primer modo de vibración en 3.45 Hz y el segundo modo en 6.93 Hz, así como sus respectivos umbrales de detección, par de líneas en color morado para la frecuencia y línea en color azul para el umbral de amplitud.

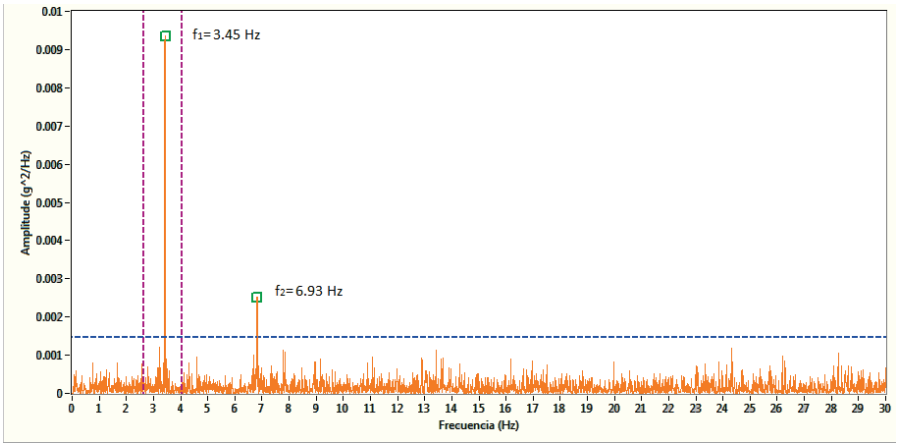


Figura 12. Detección de frecuencia de vibración del tirante.

Por último, el algoritmo obtiene el valor de la tensión axial del tirante 4 de la semi-arpa 6, al despejar *T* de la ecuación 1, ya son conocidos tanto la frecuencia del tirante como la longitud y masa por unidad de longitud del elemento. Aplicando la fórmula, se obtiene una tensión de 170.649 Toneladas,

representando únicamente la tensión de dos minutos continuos de monitoreo. En la figura 13 se muestran los 720 valores de la tensión del tirante 4 de la semi-arpa 6, en el día 22 de abril de 2018. Se observa que el efecto térmico no es significativo en el comportamiento de la tensión durante el periodo de un día. Los incrementos de tensión se deben principalmente al paso de vehículos con mayor peso que el promedio del flujo vehicular, siendo 164.34 t. Adicional, se observan en líneas amarillas la tensión mínima (153 t) y máxima de proyecto (189 t), y con línea color roja la tensión máxima de diseño (226.37 t) para este tirante.

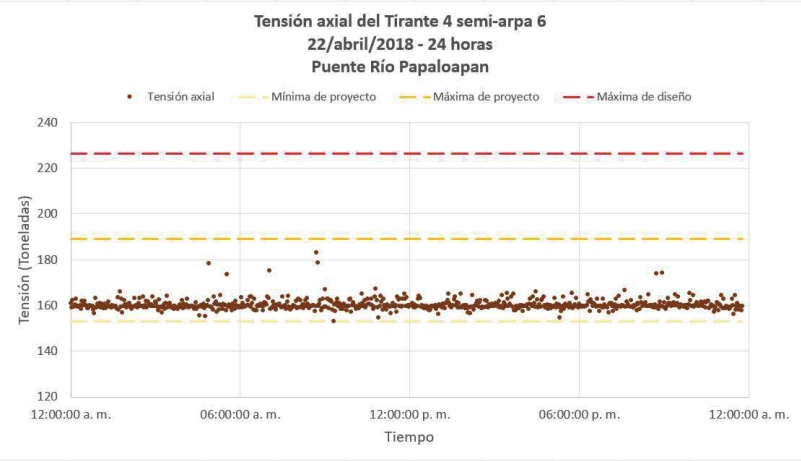


Figura 13. Comportamiento de la tensión en el cable 4 semi-arpa 6 en el día 22 de abril de 2018.

5. Resultados

5.1 Detección de referencia de deformación

La aplicación de los algoritmos para detección de la referencia de deformación ha proporcionado al monitoreo de puentes del CeMPEI, una herramienta potente para identificar de manera inmediata eventos como: variaciones por efectos térmicos diarios, fallas en elementos estructurales, destensado y tensado de tirantes; a mediano plazo: actividades de re-encarpetado de la calzada y ciclos térmico por estacionalidad, y a largo plazo: comportamiento multianual de los ciclos por temperatura.

La figura 14 indica el comportamiento histórico del valor de la referencia de deformación del extensómetro R9, durante los últimos cinco años de monitoreo. De manera inmediata se observa alta sensibilidad de la estructura, al tensado y destensado del tirante 2 de la semi-arpa 7. En verano, cuando el valor de la temperatura se eleva por encima del valor medio anual, la estructura sufre un fenómeno de tensión; comprimiéndose en invierno. Es notable, en el comportamiento anual, una ligera tendencia a efectos de compresión en la zona instrumentada.

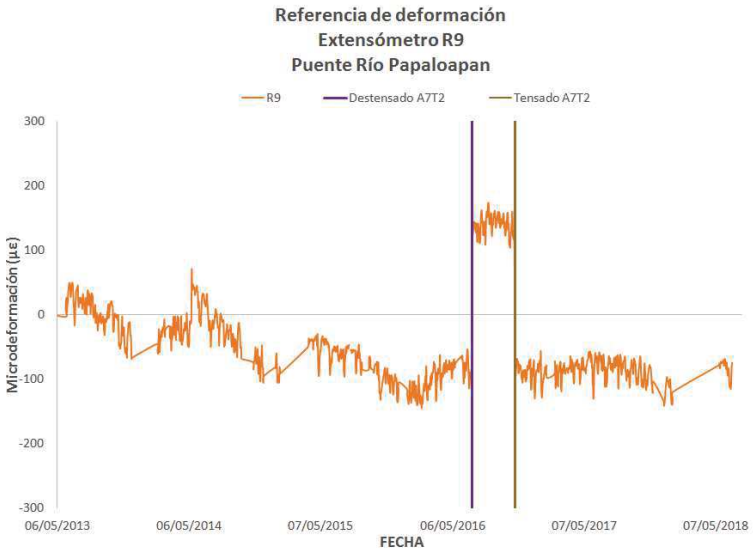


Figura 14. Comportamiento multi-anual de la referencia de deformación del extensómetro R9.

5.2 Detección de deformación por carga viva

En el caso de los algoritmos para detección de carga viva, de manera inmediata se han identificados: deformaciones a tensión y compresión por día, eventos extraordinarios (vehículos con sobrecarga); a mediano plazo: variaciones de la media por carga viva asociados a fallas de elementos estructurales, ciclos por estacionalidad y actividades de rehabilitación, y a largo plazo: comportamiento multianual de los ciclos por temperatura.

La figura 15 muestra el comportamiento histórico de los últimos cinco años de monitoreo del valor de la media de deformación por carga viva, efectos a tensión en la zona instrumentada, del extensómetro R6. Cada valor de deformación por carga viva corresponde al promedio de 15 días continuos de deformaciones por carga a tensión, ajustados a la función de distribución estadística Inversa de Gauss. Es evidente el comportamiento cíclico debido a las temporadas de verano e invierno, comportándose con amplitudes mayores de deformación por carga viva en épocas de menor temperatura. Adicional, se percibe alta sensibilidad a actividades de reencarpetamiento de la calzada del puente, así como actividades de desvió de tráfico debido a los cierres temporales de un cuerpo del puente.

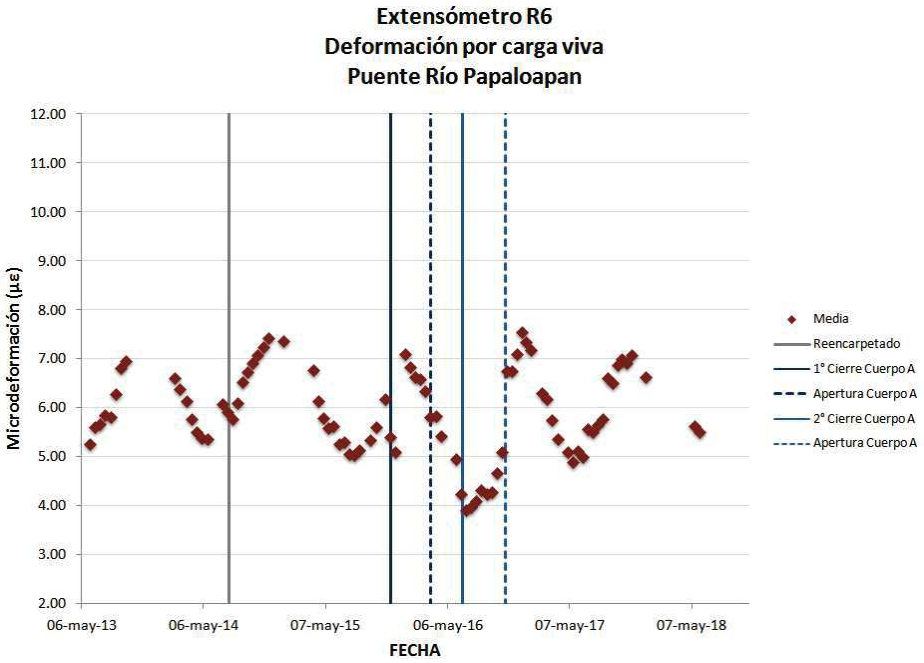


Figura 15. Comportamiento multi-anual de deformación por carga viva del extensómetro R6.

5.3 Detección de tensión axial en tirantes

El algoritmo de detección de tensión axial en tirantes ha permitido identificar de manera inmediata: vehículos con mayor carga que el promedio de vehículos, trabajos de mantenimiento en el puente. A mediano plazo: se han observado incrementos de tensión por reencarpetamiento en el puente, así como redistribución de cargas en tirantes aledaños. A largo plazo: incrementos anuales en la tensión de los tirantes, debidos al incremento del flujo vehicular que transita por el puente.

En la figura 16 se observa el comportamiento histórico del valor de la media de la tensión del tirante 4 de la semi-arpa 6, que comprende los últimos cinco años de monitoreo. Cada valor de tensión axial corresponde al promedio de 15 días continuos de tensión, ajustados a la función de distribución estadística Log-logistic. Como en caso del valor de deformación por carga viva, el tirante es sensible a actividades de reencarpetado de la calzada del puente, así como a fallas en elementos estructurales próximos, tales como la ruptura del elemento superior de anclaje del tirante 1 de la semi-arpa 5, además de las actividades relacionadas para su reemplazo.

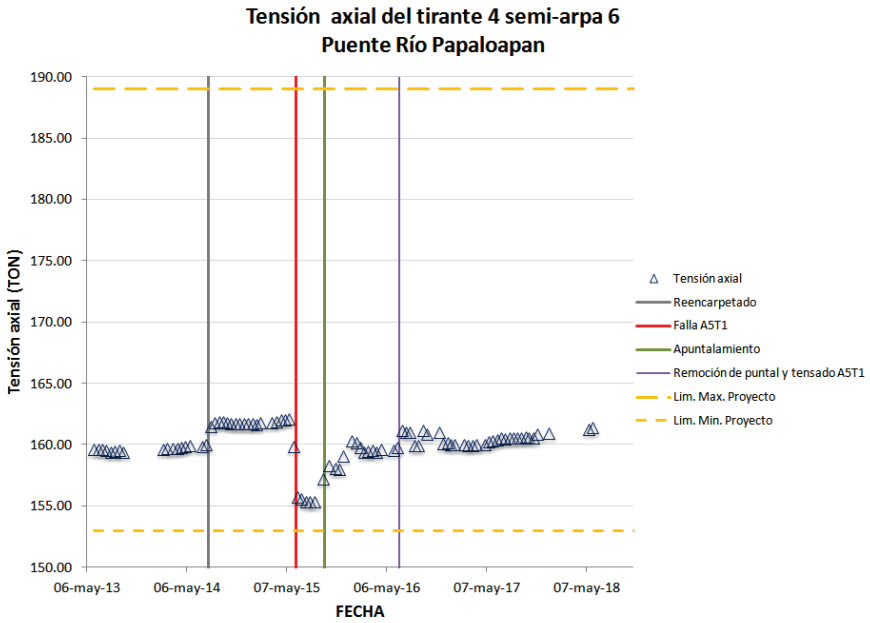


Figura 16. Comportamiento multi-anual de tensión axial del tirante 4 semi-arpa 6.

6. Conclusiones

Los resultados han permitido generar una base de información robusta en los puentes instrumentado, con ello se discriminan fenómenos de carácter térmico de los eventos dinámicos y fallas estructurales.

La capacidad de procesamiento de las computadoras actuales permite que los algoritmos sean ejecutados en tiempo real.

La optimización de los algoritmos actuales permitirá desarrollar e implementar nuevos algoritmos como: detección del tiempo de ocupación, discriminación del sentido de los vehículos y pesaje instantáneo de vehículos.

Referencias

[1] Martínez L. A., Quintana J. A., Arroyo B., Carrión F., Crespo S., Hernández J., Bonilla V., Gasca H., Hernández A. "*Sistema Inteligente y protocolos de actuación para monitoreo remoto de puentes",* IMT-Publicación Técnica N° 464. Sanfandila, Qro. 2016.

[2] Arenas L. M., Gómez R. "*Comparación de un modelo matemático simple de un puente para reproducir respuestas debidas a incrementos de deformación producto de cargas móviles, mediante datos obtenidos de instrumentación con fibra óptica*", XVII Congreso Nacional de Ingeniería Estructural. León, Gto. 2010.

[3] Quintana J. A., Crespo S., Hernández J., Gasca H. y Martínez L. A. *"Implementación del sistema de monitoreo del Puente Mezcala",* IMT-Nota N° 165. Sanfandila, Qro. 2017.

[4] Lagos G., "*Implementación de un algoritmo de monitoreo estructural basado en objetos simbólicos y clasificación por agrupamiento"*, Tesis de maestría. Universidad de Chile, 2017.

[5] Sedgewick R., "*Algoritmos en C++"*, Ed. Adisson-Wesley Publishing Company Inc., Massachusetts, EUA 1995.

[6] Normativa SCT, "*N-PRY-CAR-6-01-003/01, Proyecto de puentes y estructuras"*, Capítulo 003 Cargas y Acciones. 2001.

[7] Óskarsson, K. U. "*Structural health modeling of the Ölfusá Suspension Bridge – Damage detection and monitoring aspects",* Master´s thesis. Faculty of Civil and Environmental Engineering, University of Iceland, 2012.

[8] Carrión, F. J., Hernández J. R. y Terán, J., "*Estudio experimental de un puente atirantado",* Publicación Técnica del IMT. Querétaro, México, No. 255, pp. 21, 34, 54-55, 2005.

[9] National Instruments, "*White paper 3566: LabVIEW para medición y análisis de datos".* EUA, Abril 2014.

Ingeniería y Automatización, Capítulo 14, pp. 176 - 187.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño y Validación de un Prototipo Robótico Continuum Multisección

Castro Riveros Fabian Camilo y Perilla Castaño Victor Mario

Corporación Tecnológica Industrial Colombiana
Facultad de Mecatrónica
Grupo de Investigación INNOVATEC
Área de robótica y control

Resumen

Este documento presenta el diseño y validación de un robot continuum multisección, el cual se elaboró con el propósito de trabajar sobre entornos desconocidos y pueda tener múltiples grados de libertad. El prototipo consta de 13 eslabones conectados por medio de 6 guayas (tendones), dividiendo al robot en dos secciones para posicionar, la sección 1 (efector final) consta de 3 tendones separados entre sí por resortes y la sección 2 consta de 3 tendones de igual manera separados por resortes, estos se utilizan para controlar la posición y orientación, además dicha sección es atravesada por los 3 tendones de la sección 1 para dar mayor rigidez y estabilidad al posicionamiento. Se implemento un modelo de cinemática directa e inversa de curvatura constante, con el fin de validar el comportamiento del robot dentro de su espacio de trabajo. Los resultados obtenidos nos muestran que el robot se logra posicionar en las coordenadas establecidas, pero generan un desfase por el diseño mecánico y no se implementó un control en lazo cerrado que elimine el error de posicionamiento.

Palabras clave: Robot Continuum, Cinemática de Robots, Posicionamiento.

1. Introducción

Las estructuras de brazo Robótico con múltiples de grados de libertad y alta flexibilidad se suelen llamar brazos continuum[1] . Estos robots presentan una estructura similar a contrapartes biológicas como serpientes, trompas de elefante y brazos de pulpo. Las increíbles capacidades de locomoción, manipulación y destreza de dichos animales han inspirado a los investigadores a trabajar para recrear sus capacidades en dispositivos electromecánicos[2] .

Los brazos bio-asistidos están siendo utilizados en el campo de la medicina sobre todo en prótesis adecuando sistemas cerebro - maquina a través de interfaces. También las posibles aplicaciones de los robots continuum incluyen la navegación a través de entornos congestionados e impredecibles en los que un robot continuum puede usarse para la exploración subterránea o subacuática, un robot continuum con sus habilidades únicas puede llegar a lugares que generalmente son inaccesibles para robots de enlaces rígidos y hostiles para seres humanos.

En el diseño del robot continuum se utilizaron herramientas computacionales con el fin de validar cargas de acuerdo a los materiales seleccionados y poder predecir posibles fallas en los mecanismos, facilitando la construcción y su control, esta técnica llamada diseño basado en modelos nos permite tratar diversos diseños mecánicos, eléctricos y de control con el fin de optimizar el comportamiento del sistema y utilizar un entorno de simulación para poner a prueba todo el envolvente operativo antes de su fabricación.

Posicionar un robot continuum es el paso posterior al diseño, dicho posicionamiento se puede hacer de acuerdo con los modelos matemáticos de cinemática, que consiste en el uso de ecuaciones

para el cálculo de la posición, mediante la utilización de tendones y la cinemática inversa que nos ayuda a partir de la posición final del brazo, calcular mediante la elongación de los tendones su nueva posición. Estos modelos serán utilizados en la elaboración del robot para controlar los movimientos y los grados de libertad del brazo.

2. Mecanismos y Diseño

El robot continuum está dividido en dos secciones móviles de 40 cm cada una y la sección de posicionamiento que mide 30 cm

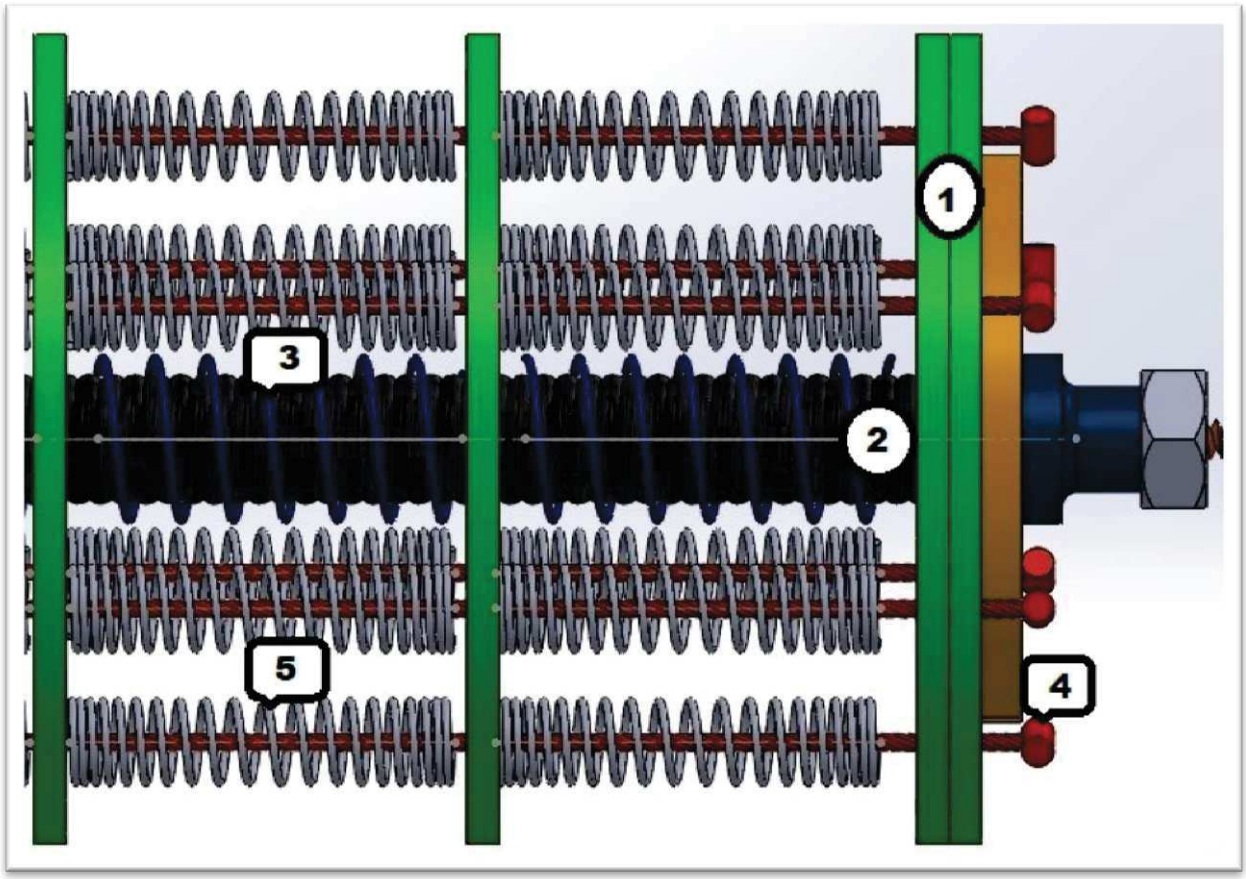


Figura 1. Sección del brazo robótico

La sección móvil del robot está dividida en 2 partes flexibles las cuales están compuestas por (ver tabla 1). Mediante el uso de la estructura flexible se proporciona un menor costo de fabricación, reduciendo el peso del brazo robótico, aumentando la velocidad en el movimiento. En nuestro diseño, la mecánica completa consiste en 13 secciones que están conectados a través de tendones (guayas) y resortes que permiten tener una mejora en la flexibilidad comparándolo con otro tipo de brazos robóticos.

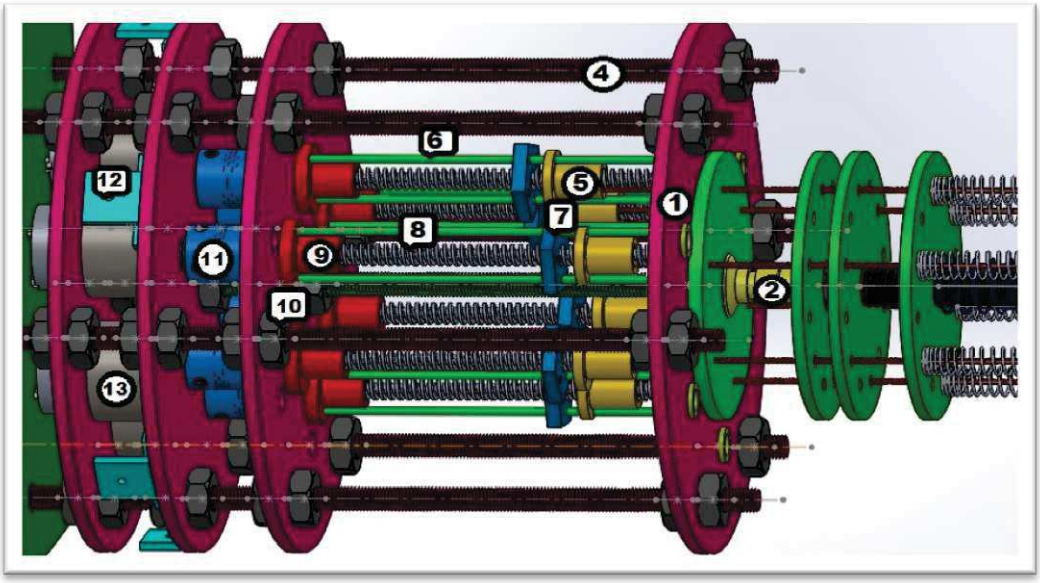


Figura 2. Sección de posicionamiento

El mecanismo de posicionamiento del robot continuum, como se muestra en la (Figura 2), se compone principalmente de 6 motores, tornillos sin fin, tensores, separadores y acrílicos

3. Cinemática

El diagrama de bloques del análisis cinemático para el módulo de articulación (j). La cinemática de avance se presenta en las subsecciones $f_{especifico}$ y $f_{independiente}$ y la cinemática inversa es cubierta en las subsecciones $f^{-1}_{especifico}$ y $f^{-1}_{independiente}$ para el brazo robótico de n-módulos. (Ver figura 3.)

El espacio de trabajo global [G] se adjunta a la plataforma fija del primer módulo o modulo cartesiano. Cada conjunto de ángulos comprende 2 componentes que reflejan la orientación relativa del disco base con el movimiento de la plataforma por cada módulo. La plataforma móvil de un módulo se considera como el disco base del siguiente modulo.

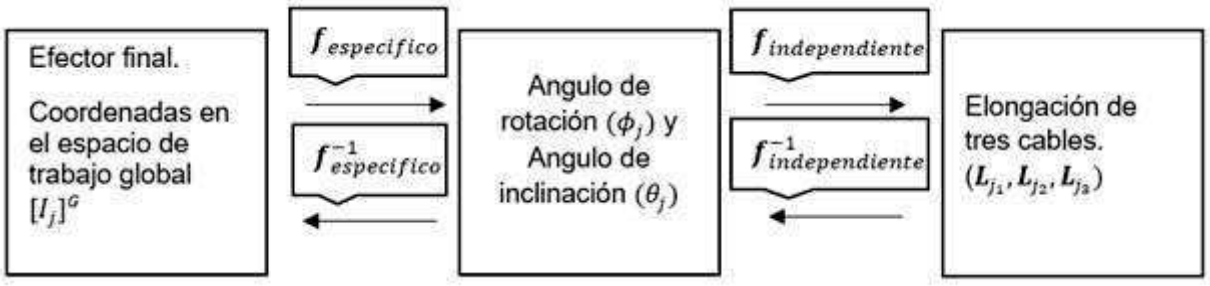


Figura 3. Diagrama de espacios de trabajo de un robot manipulador continuum

En la figura 3 se puede ver los espacios de trabajo de un robot continuum, se generan tres espacios, que son: espacio cartesiano donde encontramos las coordenadas en X, Y, Z de posicionamiento, espacio de articulación donde implementamos los ángulos de rotación e inclinación y espacio longitudinal donde aplicamos la elongación de los tres cables.

3.1 Matriz de orientación y rotación

R(Φ,θ) = $\boldsymbol{R}_z(\Phi)\ \boldsymbol{R}_y(\theta)\ \boldsymbol{R}_z(-\theta)$

$$= \begin{pmatrix} C\phi^2\ C\theta + S\phi^2 & S\phi\ C\phi\ C\theta & C\phi\ S\theta \\ S\phi\ C\phi\ C\theta & S\phi^2\ C\theta + C\phi^2 & S\phi\ S\theta \\ -C\phi\ S\theta & -S\phi\ S\theta & C\theta \end{pmatrix} \tag{1}$$

$$S\phi = \sin\phi\,, C\phi = \cos\phi, \qquad S\theta = \sin\theta, C\theta = \cos\theta$$

3.2 Desde los ángulos de orientación para encontrar las coordenadas del efector final en $[\boldsymbol{G}] = \boldsymbol{G}$ en cada seccion

Para cualquier desplazamiento del efector final, la flexión que forma la columna vertebral se supone que es circular con una longitud de arco constante L_0, si el angulo de inclinación de la sección es θ_j, entonces el arco del radio es $R_j = \frac{L_0}{\theta_j}$. Desde la variación en los dos puntos finales del arco son geométricamente perpendiculares a las plataformas respectivas, la configuración flexible de la columna puede ser considerada como una articulación universal con su centro en k_j y la longitud del brazo definido por:

$$L_j = \overrightarrow{|k_j I_j|} = \overrightarrow{|k_j I_{j-1}|} = \left(\frac{L_0}{\theta_j}\right) \tan\left(\frac{\theta_j}{2}\right) \tag{2}$$

$$\boldsymbol{L_j} = \left(\frac{\boldsymbol{L_0}}{\boldsymbol{\theta_j}}\right) \boldsymbol{tan}\ \left(\frac{\boldsymbol{\theta_j}}{\boldsymbol{2}}\right) \tag{3}$$

$$\begin{matrix}[G] \\ [j]_3\end{matrix}\boldsymbol{T} = \begin{matrix}[1]_1 \\ [1]_2\end{matrix}\boldsymbol{Q}\begin{matrix}[1]_2 \\ [1]_3\end{matrix}\boldsymbol{R}\ [\ \prod_{k=2}^{k=j}(\ \begin{matrix}[k-1]_3 \\ [k]_2\end{matrix}\boldsymbol{Q}\begin{matrix}[k]_2 \\ [k]_3\end{matrix}\boldsymbol{R})] \tag{4}$$

Por el cual:

$\begin{matrix}[1]_2 \\ [1]_3\end{matrix}\boldsymbol{R}$; $\begin{matrix}[k]_2 \\ [k]_3\end{matrix}\boldsymbol{R}$ reflejar la matriz de rotación de la ecuación (1) y ecuación (3) respectivamente, obtenidos mediante la sustitución del conjunto de ángulos de orientación $(\phi_1, \theta_1); (\phi_k, \theta_k)$ en la ecuación (1).

$$\begin{matrix}[1]_2 \\ [1]_3\end{matrix}\boldsymbol{R} = \begin{bmatrix}\boldsymbol{R}(\phi_1, \theta_1) & 0 \\ 0 & 1\end{bmatrix};\ \begin{matrix}[k]_2 \\ [k]_3\end{matrix}\boldsymbol{R} = \begin{bmatrix}\boldsymbol{R}(\phi_k, \theta_k) & 0 \\ 0 & 1\end{bmatrix} \tag{5}$$

$\begin{matrix}[k-1]_3 \\ [k]_2\end{matrix}\boldsymbol{Q}$; $\begin{matrix}[1]_1 \\ [1]_2\end{matrix}\boldsymbol{Q}$ se consulta la ecuación de la matriz entre los marcos, los valores están representados por la siguiente matriz.

$$\begin{matrix}[j-2]_3 \\ [j-1]_2\end{matrix}\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{I}_{3*3} & \begin{pmatrix}0 \\ 0 \\ \boldsymbol{L}_{j-1} + \boldsymbol{L}_{j-2}\end{pmatrix} \\ 0 & 1\end{bmatrix};\ \begin{matrix}[1]_1 \\ [1]_2\end{matrix}\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{I}_{3*3} & \begin{pmatrix}0 \\ 0 \\ L_1\end{pmatrix} \\ 0 & 1\end{bmatrix} \tag{6}$$

Con los correspondientes ángulos de inclinación de cada ecuación y podemos obtener esos valores de la longitud de los tensores.

La coordenada del efector final $\boldsymbol{I}_j$ de $\boldsymbol{j}^{th}$ de la ecuación, se define como $[\boldsymbol{I}_j]^{[j]_3} = \left(0\ 0\ \boldsymbol{L}_j\right)$ en el espacio de trabajo $[j]_3$, del mismo modo, sus coordenadas en el espacio de trabajo global esta dado por $[\boldsymbol{I}_j]^G$ y expresado en la siguiente formula.

$$\begin{pmatrix}[I_j]^G\\1\end{pmatrix}_{4x1} = {}^{[1]_1}_{[j]_3}\boldsymbol{T}\begin{bmatrix}[\boldsymbol{I_j}]^{[j]_3}\\\mathbf{1}\end{bmatrix}_{4x1} \tag{7}$$

El objetivo es encontrar $[I_j]^G$ y se puede lograr evaluando ${}^{[1]_1}_{[j]_3}\boldsymbol{T}$ de (4).

3.3 Determinación de la orientación del ángulo para cada módulo basado en la coordenada del efector final en el espacio de trabajo global

Aquí, el propósito es encontrar el ángulo de inclinación (θ) y el ángulo de rotación (Φ) de cada ecuación con su posición:
Coordenadas del efector final I_j de todas las ecuaciones en el espacio de trabajo $(\boldsymbol{X}_{I_j}, \boldsymbol{Y}_{I_j}, \boldsymbol{Z}_{I_j})$ son dados.
Considerando la ecuación (1) y su efector final, la transformación de la matriz del espacio de trabajo $[1]_3$ al espacio de trabajo global es

$$\begin{pmatrix}\boldsymbol{X}_{I_j}\\\boldsymbol{Y}_{I_j}\\\boldsymbol{Z}_{I_j}\end{pmatrix} = \begin{pmatrix}L_1 & \cos\phi_1 & \sin\theta_1\\L_1 & \sin\phi_1 & \sin\theta_1\\L_1 & \cos\phi_1 & +L_1\end{pmatrix} \tag{8}$$

Igualando ambos lados

$$\therefore \phi_1 = tan^{-1}\left(\frac{Y_{I_1}}{Z_{I_1}}\right);\ \theta_1 = cos^{-1}\left(\frac{-A+1}{A+1}\right);\ con\ A = (\frac{Y_{I_1}}{Z_{I_1}}sin\phi_1)^2 \tag{9}$$

Por lo tanto, la magnitud del brazo de la articulación universal imaginada es:
De la ecuación (4) → $\boldsymbol{L}_1 = \left(\frac{L_0}{\theta_1}\right)\tan\left(\frac{\theta_1}{2}\right)$

Usando el método inductivo en matemáticas asumiendo todos los conjuntos: $(\Phi_k, \theta_k, \boldsymbol{L}_k)$ con k=1.. (j – 1) están determinados, vamos a encontrar esos valores $\left(\Phi_j, \theta_j, \boldsymbol{L}_j\right)$.

Usando ${}^{[1]_1}_{[j]_3}\boldsymbol{T}$ de (4); coordenadas del efector final $\boldsymbol{I_j}$ de $\boldsymbol{j}^{th}$ modulo en el cuadro $[\boldsymbol{j}]_3$: $[\boldsymbol{I_j}]^{[j]_3} = \left(\mathbf{0}\ \mathbf{0}\ \boldsymbol{L_j}\right)$, coordenada del efector final y el espacio de trabajo global $[\boldsymbol{G}]$: $[\boldsymbol{I_j}]^{\boldsymbol{G}} = (\boldsymbol{X_{I_j}}\ \boldsymbol{Y_{I_j}}\ \boldsymbol{Z_{I_j}})$ y ecuación (7)

$$\therefore \begin{bmatrix}X_{I_j}\\Y_{I_j}\\Z_{I_j}\\1\end{bmatrix} = \begin{bmatrix}\boldsymbol{R}_{(\boldsymbol{\phi_1},\boldsymbol{\theta_1})} & \begin{pmatrix}0\\0\\L_1\end{pmatrix}\\0 & 1\end{bmatrix}\prod_{k=j-1}^{k=j}\begin{bmatrix}\boldsymbol{R}_{(\phi_k,\theta_k)} & \begin{pmatrix}0\\0\\L_{k-1}+L_k\end{pmatrix}\\0 & 1\end{bmatrix}\begin{bmatrix}0\\0\\L_j\\1\end{bmatrix} \tag{10}$$

Multiplicando a ambos lados por la matriz inversa de los que están en la derecha, lado de la mano, obtenemos un conjunto de ángulos de orientación del módulo (j)

→ $\phi_j = \tan^{-1}\left(\frac{b_j}{a_j}\right);\ \theta_j = \cos^{-1}\left\{\frac{B+1}{B+1}\right\}$ con $B = \left(\frac{b_j}{j}\sin\phi_j\right)^2$ en el cual a_j, b_j, c_j se obtienen calculando

$$\begin{bmatrix}a_j\\b_j\\c_j\\1\end{bmatrix} = \prod_{k=j-1}^{k=2}\begin{bmatrix}R_{(\phi_k,\theta_k)} & \begin{pmatrix}0\\0\\L_{k-1}+L_k\end{pmatrix}\\0 & 1\end{bmatrix}^{-1} * \begin{bmatrix}R_{(\phi_1,\theta_1)} & \begin{pmatrix}0\\0\\L_1\end{pmatrix}\\0 & 1\end{bmatrix}^{-1}\begin{bmatrix}X_{I_j}\\Y_{I_j}\\Z_{I_j}\\1\end{bmatrix} - \begin{bmatrix}0\\0\\L_{j-1}\\1\end{bmatrix} \tag{11}$$

$R_{(\phi_k,\theta_k)}, R_{(\phi_1,\theta_1)}$ son matriz de rotación para el modulo k y 1 respectivamente.

$L_{k-1}; L_{k-2}; L_1; L_{j-1}$ son de la ecuación (2) con sustitución de los ángulos de inclinación correspondientes para cada sección.

4. Cinemática inversa

La expresión de la transformación homogénea viene dada por la ecuación (14) y obtenida en función del vector (12) y la matriz de rotación (13) en el vector $\boldsymbol{t} \in \boldsymbol{R^3} : \boldsymbol{t} = \boldsymbol{t(s)}$, donde s es un parámetro variable que representa la longitud al cruzar la curva.

$$\boldsymbol{t}\,(\boldsymbol{s}) = [\boldsymbol{x(s)}, \boldsymbol{y(s)}, \boldsymbol{z(s)}] \quad (12)$$

$$\boldsymbol{Rot}\,(\boldsymbol{s}) = \left[\boldsymbol{T}_{(s)}\ \boldsymbol{N1}_{(s)}\ \boldsymbol{N2}_{(s)}\right] \quad (13)$$

$$\boldsymbol{{}^{k}_{k+1}T(s)} = \begin{pmatrix} \boldsymbol{Rot}\,(\boldsymbol{s}) & \boldsymbol{t(s)} \\ \boldsymbol{0} & \boldsymbol{1} \end{pmatrix} \quad (14)$$

4.1 Relación entre la longitud del cable y los ángulos de orientación

4.1.1 Dada la rotación y los ángulos de inclinación, encontrar la longitud del cable para cada espacio de trabajo.

Es necesario evaluar las longitudes de los cables en función de un conjunto de ángulos orientados. La figura 4, muestra la configuración del modulo j^{th} con tres cables. Los puntos B_{j_1} y P_{j_1} (i=1,2,3) son los puntos finales de i^{th} con un resorte o columna vertebral, que une el disco base con la plataforma móvil. La longitud del vector $\overrightarrow{B_{j1}P_{ji}}$ (i=1,2,3) representa la longitud del cable I_{ji} de i^{th} cable.

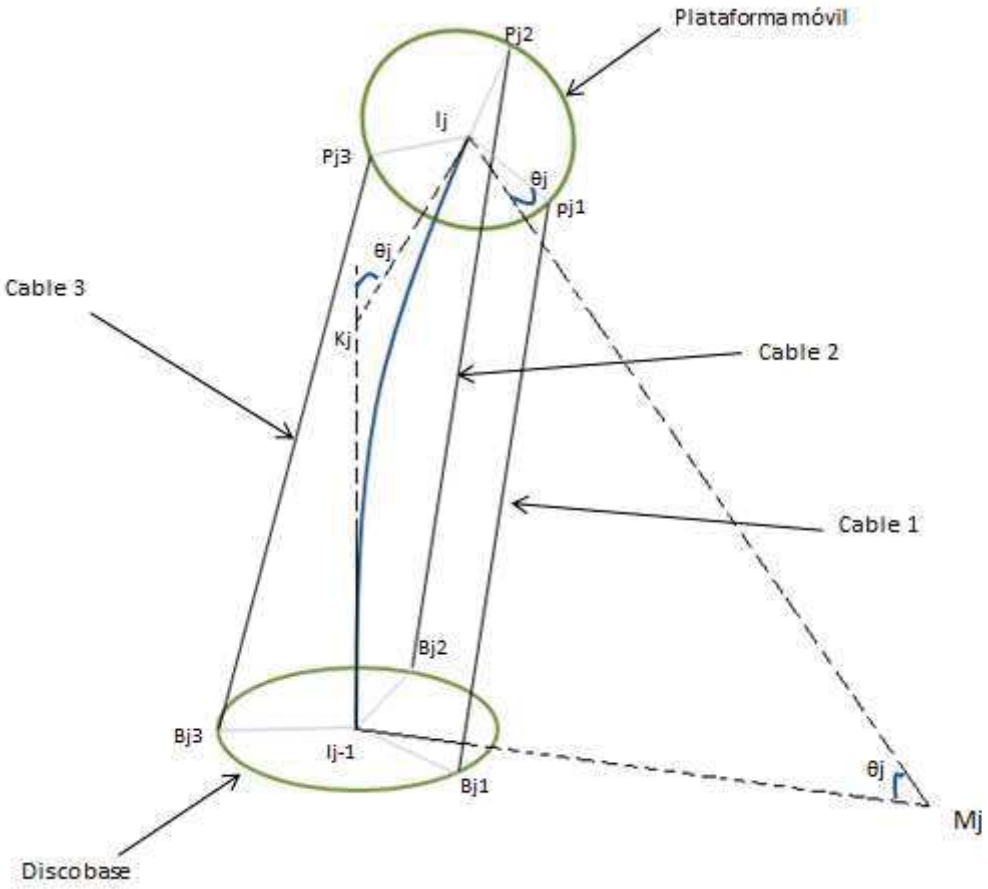


Figura 4. Configuración del modulo $(j)^{th}$

Una ecuación de cierre de bucle para el cable se puede escribir como: $\overrightarrow{P_{J1}B_{J1}} = \overrightarrow{P_{J1}I_J} + \overrightarrow{I_JK_J} + \overrightarrow{K_JI_{J-1}} + \overrightarrow{I_{J-1}B_{J1}}$ todos los vectores están representados en el cuadro $\{j\}_1$ es decir, con respecto a la plataforma base del modulo j^{th} por lo tanto:

$$\overrightarrow{P_{J1}B_{J1}} = \begin{pmatrix} I_{3*3} & R_{(\phi_j,\theta_j)} \end{pmatrix} \begin{pmatrix} \text{rcos}(j\pi - 2\pi(i-1)/3) \\ \text{rsin}(j\pi - 2\pi(i-1)/3) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ L_j \end{pmatrix} \tag{15}$$

R es el radio de la plataforma L_j, se visualiza el brazo del módulo de la articulación universal, por lo tanto, la longitud del tendón se determina tomando la magnitud de $\overrightarrow{P_{J1}B_{J1}}$.

4.1.2 Encontrar los ángulos de orientación basados en longitudes de cable.

Geométricamente, los tres tendones dentro de un módulo conjunto, son paralelos:

$$P_{j1}B_{j1} // P_{j2}B_{j2} // P_{j3}B_{j3} // I_{j-1}I_j \tag{16}$$

Entonces el ángulo θ_j, se obtiene a partir de la siguiente ecuación, en la que L_0 es la longitud de la columna vertebral:

$$\sin\left(\frac{\theta_j}{2}\right) = \frac{\theta_j(I_{j1}+I_{j2}+I_{j3}}{12L_0} \tag{17}$$

El cable 1 en el módulo (j) está conectado en 2 puntos B_{j1} y P_{j1}. Diferenciamos el cable 1 de otros cables en el mismo modulo, la ubicación del punto B_{j1} debe ser determinado. La ubicación del punto B_{j1} estaría en el disco base del modulo (j). Si j es impar, la ubicación del punto B_{j1} esta en el punto 2; de lo contrario esta en el punto 1. Ver Figura 5

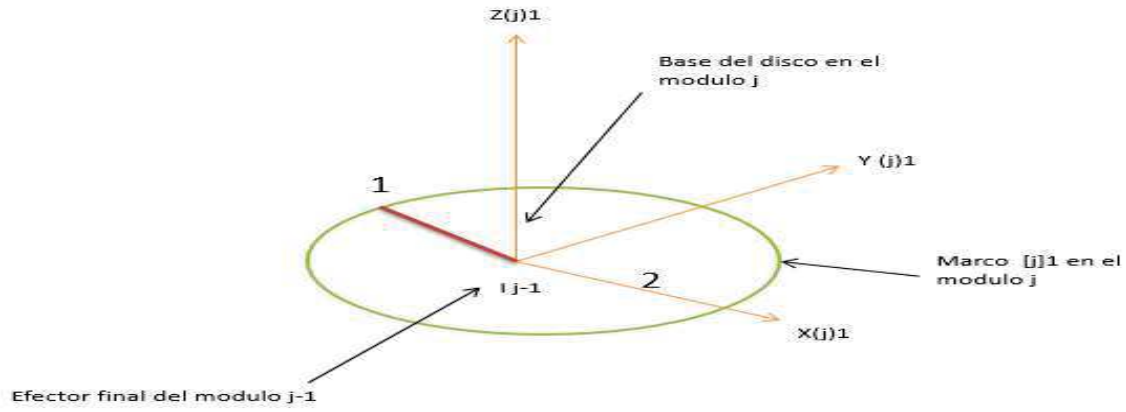


Figura 5. Posición del punto B_{j1} en el modulo $(j)^{th}$

La relación entre la longitud del cable 1y el conjunto de los ángulos de orientación en el modulo j^{th} se obtiene de la ecuación (15):

$$I_{j1}{}^2 = \left[(-1)^{j+1} r\left(1 - C_\phi{}^2 * C_\theta - S_\phi{}^2\right) + L_j C_\phi S_\theta\right]^2 + \left[(-1)^{j+1} r\left(S_\phi C_\theta C_\phi - S_\phi C_\phi\right) + L_j C_\phi S_\theta\right]^2 + \left[(-1)^{j+1} r S_\theta C_\phi + L_j\,(1 + C_\theta)\right]^2 \tag{18}$$

Donde $S_\phi = \sin\phi$, $C_\phi = \cos\phi_j$, $S_\theta = \sin\theta_j$, $C_\theta = \cos\theta_j$, r es el radio de la plataforma, L_j se visualiza el brazo del módulo de la articulación universal j^{th}.

Al reorganizar términos, se obtiene una ecuación cuadrática:

$$AC_\phi{}^2 + BC_\phi + C = 0$$
$$A = 2r^2(C_\theta - 1), B = (-1)^{j+1} * 4L_j r S_\theta \; y \; C = I_{j1}{}^2 - 2L_j{}^2 - 2L_j{}^2(1 + C_\theta)$$
$$\therefore \phi_j = \cos^{-1}\left(\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}\right)$$

5. Resultados

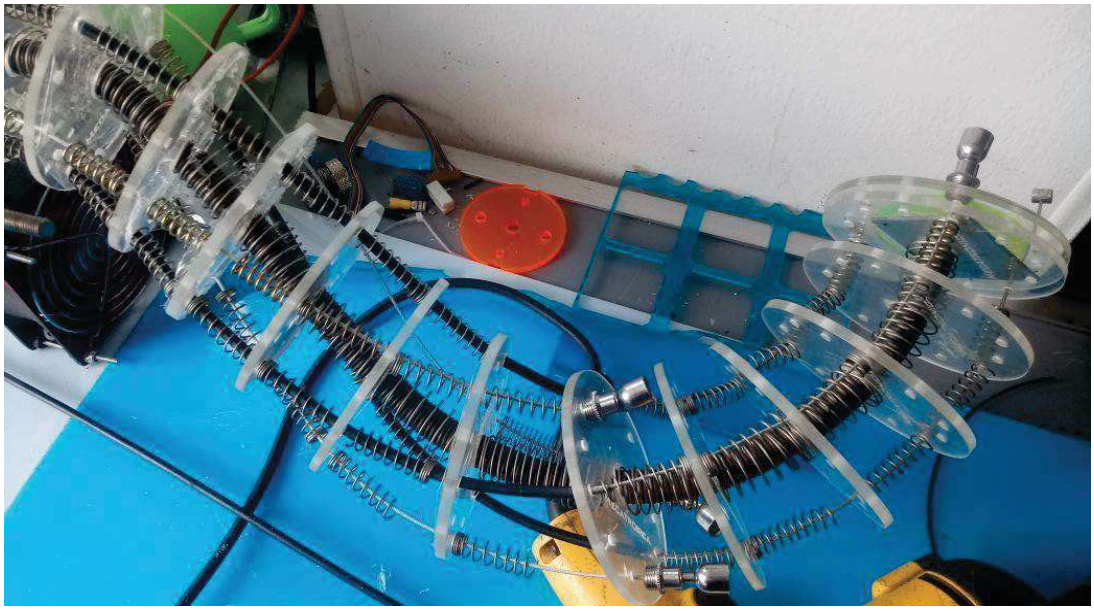


Figura 6. Secciones móviles del brazo robótico.

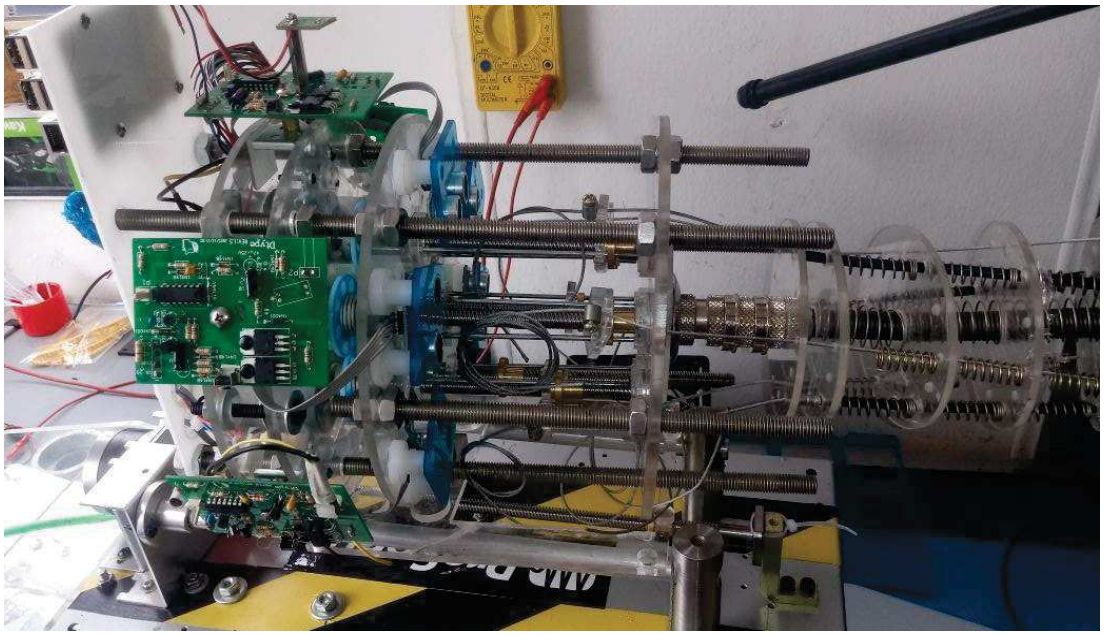


Figura 7. Sección de posicionamiento.

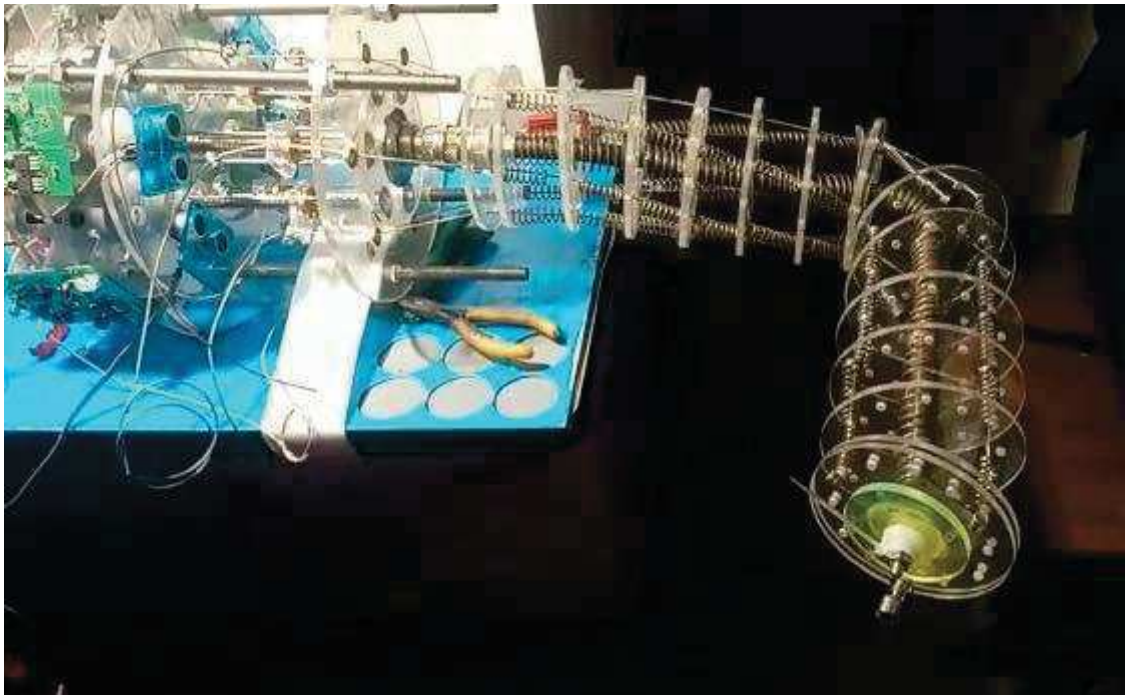


Figura 6. Brazo robótico.

5.1 Tablas.

En las tablas 1 y 2, tenemos el listado de los materiales utilizados en la elaboración del brazo robótico y su descripción.

Tabla 1. Sección del Brazo Robótico

#	Nombre	Descripción
1	Discos en Acrílico	Acrílico transparente, de espesor 5 mm y un diámetro de 100 mm. Cuenta con 8 perforaciones en los extremos de circunferencia de diámetro 4 mm y una perforación en el centro de 14 mm.
2	Columna Vertebral	Resorte de extensión de 14 mm. Tiene un diámetro interno de 12 mm, diámetro externo 14 mm, diámetro del alambre 4 mm, longitud de 800 mm y está fabricado en acero para resorte de uso general.
3	Soporte Vertebral	Resorte de alambre. material Aisi 1045 Acero estirado al frio con una longitud de 7 cm.
4	Tendones (guayas)	Cables de acero de diámetro 1.6 mm.
5	Separadores	Resorte de alambre. material Aisi 1045 Acero estirado al frio con una longitud de 7 cm.

Tabla 2. Posicionamiento del brazo

#	Nombre	Descripción
1	Discos en Acrílico	Acrílico transparente, de espesor 5 mm y un diámetro de 100 mm. Cuenta con 8 perforaciones en los extremos de circunferencia de diámetro 4 mm y una perforación en el centro de 14 mm.
2	Acoplador	Rodamiento circular

En la tabla 3. Podemos observar el comportamiento del brazo robótico, desarrollando el control en lazo abierto y brindando una mayor movilidad en la sección del efector final, con el fin de determinar qué tan preciso es el mecanismo de control y que tan alto es el porcentaje de error, para de esta manera elaborar un control más optimo que se adapte de manera eficiente al movimiento y posicionamiento del brazo.

Tabla 3. Resultados de las pruebas realizadas sin sensores

ENSAYO	POSICION DESEADA EN mm				POSICION REAL EN mm			
#	X	Y	Z		X	Y	Z	Angulo
1	400	0	400		400	0	350	120
2	300	100	100		300	50	50	135
3	380	0	200		380	0	120	145
4	400	400	0		400	370	0	110
5	200	200	100		180	200	80	160
6	400	0	-400		400	0	-350	-120
7	330	50	-300		320	50	-290	-135

En la tabla 4.

Tabla4. Resultados del error absoluto con respecto a la tabla 3.

Error Absoluto en %			
muestra	X	Y	Z
1	0	0	12.5
2	0	50	50
3	0	0	40
4	0	7.5	0
5	10	0	20
6	0	0	12.5
7	3.03	0	3.33
total	1.86	8.21	19.76

6. Conclusiones

Teniendo como base los resultados obtenidos en las pruebas realizadas con el prototipo, aplicando un control en lazo abierto (ver Tabla 3) y calculando el porcentaje de error absoluto (ver Tabla 4), podemos decir que el control del prototipo se debe optimizar, ya que el margen de error es bastante alto y por lo tanto las trayectorias no son las adecuadas ni las deseadas.

Se puede observar que el diseño del modelo CAD, se asemeja al modelo del prototipo real, aunque en el modelamiento del CAD no se tuvo en cuenta los sensores ultrasónicos, ya que el brazo robótico se controla directamente con los motores. Después de realizar los ensayos y obtener los resultados (ver tablas 3 y 4), se decidió incluir los sensores ultrasónicos para obtener un posicionamiento mas aproximado al real y un mejor control de posicionamiento del prototipo.

Como trabajo a futuro, se puede desarrollar un algoritmo para la planeación y seguimiento de trayectorias que involucren algoritmos de control, esto con el fin de garantizar el posicionamiento del robot con la curvatura deseada.

Referencias

[1] T. Zheng, D. T. Branson, E. Guglielmino, and D. G. Caldwell, "*A 3D dynamic model for continuum robots inspired by an octopus arm,*" Proc. - IEEE Int. Conf. Robot. Autom., pp. 3652–3657, 2011.

[2] R. Webster and B. Jones, "*Design and kinematic modeling of constantcurvature continuum robots: A review,*" Int. J. Rob. Res., vol. 29, no. 13, pp. 1661–1683, 2010.

[3] C. Escande et al. "*Modelling of Multisection Bionic Manipulator: application to RobotinoXT*". In: IEEE International Conference on Robotics and Biomimetics (ROBIO) (Dec. 2011), pp. 92–97.

[4] M. S. Espinoza et al. "*Inverse kinematics of a 10 dof modular hyper-redundant robot resorting to exhaustive and error-optimization methods: A comparative study*". In: Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian. IEEE. 2012, pp. 125–130.

[5] R. Kang et al. "*Kinematic model and inverse control for continuum manipulators*". In: Control and Automation (ICCA), 2013 10th IEEE International Conference on. IEEE. 2013, pp. 1615–1620.

[6] M. Rolf and J. J. Steil. "*Constant curvature continuum kinematics as fast approximate model for the Bionic Handling Assistant*". In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. IEEE. 2012, pp. 3440–3446.

[7] G.Z. Lum, S.K. Mustafa, H.R. Lim, W.B. Lim, G.L. Yang, and S.H. Yeo,"*Design and motion control of a cable-driven dexterous robotic arm*", *Sustainable Utilization and Development in Engineering and Technology (STUDENT), 2010 IEEE Conference,* pp. 106, Malaysia, 2010.

[8] S. Sahin, and L. Notash, "*Kinematics, workspace and stiffness analysis of wire-actuated parallel manipulators*", *World Congress in Mechanism and Machine Science*, pp. 1868–1872, Tianjin, China, 2004.

[9] C.B. Pham, S.H. Yeo, and G.L. Yang, "*Tension analysis of cable-driven parallel mechanisms*", *IEEE Conference on Intelligent Robots and Systems*, pp. 257-262, Singapore, 2005.

[10] I.A. Gravagne, *"Asymptotic regulation of a one-section continuum manipulator", IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 2779-2784, Las Vegas, USA, 2003.

[11] L. Yu et al. *"Kinematics Simulation and Analysis of 3-RPS parallel robot on SimMechanics"*. In: Information and Automation (ICIA), 2010 IEEE International Conference on. IEEE. 2010, pp. 2363–2367.

[12] C. Escande et al. *"Modelling of Multisection Bionic Manipulator: application to RobotinoXT*". In: IEEE International Conference on Robotics and Biomimetics (ROBIO) (Dec. 2011), pp. 92–97.

[13] D. B. Roppenecker et al. "*Multi arm snake-like robot kinematics*". In: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE. 2013, pp. 5040–5045.

[14] A. J. Hanson and H. Ma. "*Parallel transport approach to curve framing*". In: Indiana University, Techreports-TR425 11 (1995), pp. 3–7.

Ingeniería y Automatización, Capítulo 15, pp. 188 - 204.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño de un Robot Planar para Asistencia en Terapias de Rehabilitación

Chávez-Olivares César, Zacarías-Mena Juan Antonio, Cervantes-Herrera Alejandro, Salinas-Martínez Jorge y Arámbula-Miranda Luis Enrique

Departamento de Ingeniería Robótica, Universidad Autónoma de Aguascalientes
cesar.chavez@edu.uaa.mx

Resumen

En el presente trabajo se describe el desarrollo de una plataforma robótica planar de arquitectura abierta, diseñada para la implementación de esquemas de control de interacción, con las características deseables para un robot de rehabilitación de extremidades superiores. El sistema robótico tiene la configuración de paralelogramo, con servomotores fijos en la base y las flechas acopladas directamente a los eslabones, es decir, con transmisión directa. Las juntas pasivas consisten en flechas acopladas con baleros. Se presenta la validación del diseño mecánico usando valores numéricos obtenidos del modelo CAD y las hojas de datos de los elementos empleados para su construcción. Además, se muestra el funcionamiento de la plataforma con la implementación de un esquema de control no lineal conocido en la literatura.

Palabras clave: Robot, Arquitectura abierta, Rehabilitación robótica

1. Introducción

La Federación Internacional de Robótica (IFR, por sus siglas en inglés) clasifica a los robots en dos ramas con respecto a sus funciones: robots industriales y robots de servicio [1]. Dentro de la categoría de robots de servicio se encuentran los robots para medicina, área donde destacan los robots para cirugías y rehabilitación asistida, de hecho, la rehabilitación robótica ha tenido un amplio interés por los investigadores desde finales de los 90's, llevando los manipuladores a los hospitales para su evaluación clínica, específicamente los dispositivos robóticos para la neurorehabilitación de pacientes que han sufrido un accidente cerebro-vascular (ACV) [2].

Un ACV, ictus, infarto cerebral o apoplejía, ocurre cuando una arteria se obstruye produciendo interrupción o pérdida repentina de flujo sanguíneo cerebral, o bien, es el resultado de la ruptura de un vaso, dando lugar a un derrame o una hemorragia. Este tipo de accidentes son, de acuerdo con la Organización mundial de la Salud (WHO, por sus siglas en inglés), de las principales causas de muerte en el mundo y la primera causa de discapacidad en los adultos [3][4]. Los sobrevivientes de un ACV usualmente quedan con alguna discapacidad, principalmente del tipo motriz en los miembros superiores y pérdida de destreza con las manos, ambas parcialmente recuperables si se sigue adecuadamente una terapia de neurorehabilitación [5].

La neurorehabilitación es una disciplina médica integrada por un equipo de terapeutas profesionales que incluye terapia física, terapia de lenguaje y terapia ocupacional. La rehabilitación neurológica explota la neuroplasticidad del cerebro, utilizando la práctica y repetición de ejercicios en patrones de movimiento para que las neuronas sobrevivientes aprendan a controlar nuevamente los miembros. En la rehabilitación tradicional se combina el ejercicio pasivo y el ejercicio activo. La rehabilitación robótica, que complementa y mejora el proceso de rehabilitación, ha demostrado ser efectiva cuando el paciente se compromete con el seguimiento de la terapia [6][7][8]. La terapia con asistencia de robots tiene ciertas ventajas, como el incremento de repeticiones que el terapeuta puede

suministrar, la constancia del esfuerzo en cada repetición, el registro de resultados para evaluar los avances del paciente y la motivación, gracias a otras tecnologías involucradas, como la realidad virtual y los videojuegos [9].

En robótica de rehabilitación para extremidades se pueden clasificar dos tipos de dispositivos: los robots tipo brazo manipulador que funcionan con un efector final y los de tipo exoesqueleto. Quizá el más famoso de los manipuladores con efector final es el MIT Manus, comercializado bajo el nombre de InMotion ARM robot, el cual cuenta con numerosas pruebas clínicas, a través de las cuales se ha confirmado que hay mejoras en la capacidad de movimiento de las extremidades después de un ACV [6][7]. Entre los robots tipo exoesqueleto, el primero y único comercialmente disponible para rehabilitación de miembros superiores es el ArmeoPower, basado en el robot ARMinIII y sacado a la venta a finales del 2011 por Hocoma [10]. A finales del 2016 se publicaron resultados de un estudio clínico usando este dispositivo y se ha confirmado que existen mejoras motrices en los pacientes [11].

La robótica de rehabilitación sigue en constante crecimiento, de forma que con los esfuerzos de investigadores de alrededor del mundo se desarrollan nuevas plataformas para ponerlas al servicio de clínicas de rehabilitación, también se experimenta con nuevos esquemas de control de interacción, así como su integración con aplicaciones interactivas para mantener motivado al paciente. Por lo tanto, resulta de interés el contar en nuestra institución con una plataforma robótica de este tipo para la experimentación con nuevos esquemas de control de interacción y el desarrollo de interfaces visuales interactivas; sin dejar de mencionar que se espera ponerla al servicio de los fisioterapeutas de nuestra institución.

En este trabajo se presenta la propuesta y el desarrollo de una plataforma robótica planar de arquitectura abierta. En la sección 2 se presentan los detalles del diseño del manipulador, en la sección 3 los modelos cinemático y dinámico. Por último, en la sección 4 se presentan los resultados alcanzados en esta etapa del proyecto y la evaluación del diseño mecánico.

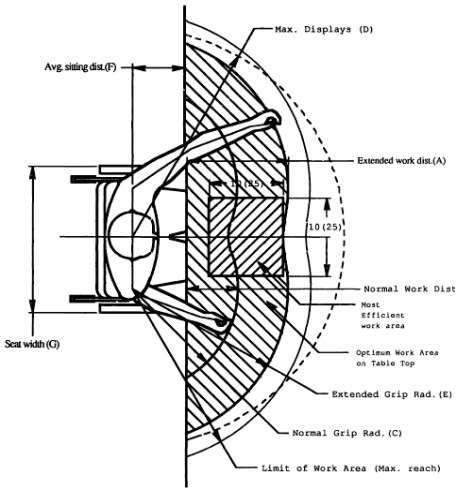


Figura 1. Espacio de trabajo de una persona en una tarea de rehabilitación, dimensiones en pulgadas. Fuente [12].

2. Diseño del robot RRCCI

El manipulador robótico presentado en este documento fue diseñado con las características deseables para un robot de rehabilitación de extremidades superiores: con un espacio de trabajo que permitiera la implementación de ejercicios para rehabilitación de hombro y codo, es decir, de aproximadamente 38 x 45 cm. Comparado con un robot industrial, un manipulador de este tipo debe presentar transparencia, baja fricción, inercia y masa generalizadas o aparentes en el efector final, así como una capacidad de fuerza de empuje similar al de una mujer débil con el codo flexionado al frente

del torso, es decir, entre 30 y 50 N (ver Figura 1). Debe presentar elementos de seguridad en su operación y docilidad en la interacción humano-robot [12]. El manipulador nombrado “RRCCI” por las siglas de Robot de Rehabilitación del Centro de Ciencias de la Ingeniería, está compuesto por los sistemas mecánico, eléctrico/electrónico y una interfaz gráfica de usuario. Estos sistemas se detallan a continuación.

2.1 Sistema mecánico

El diseño mecánico del manipulador se realizó, como ya se mencionó, atendiendo a las características que debe tener un robot de rehabilitación. Se revisaron diferentes opciones para robots planares, como el robot tipo SCARA, el cartesiano, el tipo pantógrafo y el paralelogramo, siendo este último el seleccionado por sus características dinámicas y de espacio de trabajo. El robot tipo paralelogramo tiene la ventaja de que los motores se pueden colocar en la base, de forma que las flechas quedan acopladas directamente con los eslabones, es decir, el robot puede ensamblarse con transmisión directa. Esta característica permite baja inercia y baja fricción en el efector final, además de la propiedad conocida como transparencia (back-driveability), es decir, que el usuario perciba fuerzas nulas en el efector final cuando no se inyectan torques en las articulaciones [13].

Como se puede observar en el diseño del manipulador renderizado de la Figura 2, los motores con sus flechas concéntricas están sujetos a la base, unidos a placas de acero de ½ pulgada. Estas placas están soldadas a una estructura manufacturada en PTR de 2 pulgadas y reforzadas por medio de venas a 45° de inclinación. En la parte inferior tiene también una placa de acero que cumple con dos funciones: permite montar los servoamplificadores y tarjetas electrónicas de control, a la vez que mueve el centro de masa de la estructura hacia abajo y hace contrapeso con los motores, lo que incrementa la estabilidad mecánica de la estructura cuando el manipulador se encuentra en funcionamiento.

Figura 2. Diseño mecánico del manipulador

Los eslabones están manufacturados en tubular de aluminio 6063 de 1.5 pulgadas de diámetro y cajas de aluminio 6061 en los extremos, sujetas por medio de opresores. Con las cajas y otros elementos que se describen a continuación, se forman las articulaciones o uniones, de las cuales se pueden distinguir dos tipos: las activas y las pasivas. Las articulaciones pasivas consisten en flechas fabricadas en acero cold rolled, acopladas a las cajas con baleros de bolas modelo 6202. En la Figura

3 se muestra una articulación pasiva con las cajas transparentes, donde se aprecian la flecha de acero cold rolled y los baleros, así como el tubular de aluminio.

Figura 3. Articulación pasiva

Desde un punto de vista técnico, las articulaciones activas están formadas por la parte fija (estator) y la parte móvil del motor (la flecha o rotor). El acoplamiento entre las flechas de los motores y las cajas de los eslabones se realizó con un cople ranurado de forma cilíndrica. Sobre las ranuras se coloca un cople tipo dona, manufacturado en acero cold rolled, el cual puede ajustar la presión de agarre por medio de un tornillo. El cilindro y la caja se sujetan entre sí por medio de opresores. En la Figura 4 se aprecia el acoplamiento entre la flecha y un eslabón.

a) Acoplamiento de la flecha al cilindro

b) Acoplamiento del cilindro y la caja del eslabón

Figura 4. Acoplamiento con la articulación activa

Los eslabones del manipulador se muestran en la Figura 5, éstos forman un mecanismo de paralelogramo. Este mecanismo se caracteriza por que hay dos parejas de barras que son paralelas entre sí, lo cual requiere respetar algunas relaciones de distancia que hay entre cada eje de rotación, como se explica enseguida. De acuerdo con la nomenclatura de la Figura 5, el eslabón 1 y el eslabón 4 son paralelos y tienen la misma distancia entre sus ejes de rotación, es decir, una distancia de 0.35 m. La otra pareja de barras paralelas está conformada por el eslabón 2 y el eslabón 3, en este caso la distancia entre los ejes de rotación es de 0.19 m. Además, la longitud total del eslabón 2, medida desde el eje de rotación hasta el centro del poste usado como efector final, es de 0.45 m.

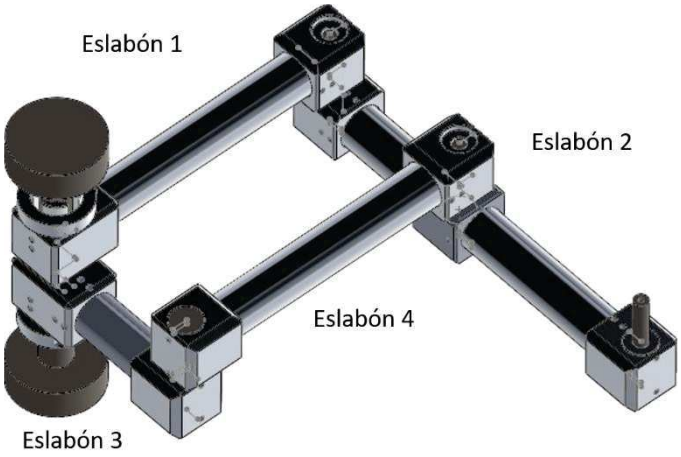


Figura 5. Mecanismo de paralelogramo

2.2 Sistema eléctrico y electrónico

Para explicar cómo está conformado el sistema electrónico, es importante mencionar la tecnología de los actuadores empleados y sus sistemas de control. Los actuadores son servomotores Brushless modelo SVM-220B de la compañía Sureservo [14], los cuales son energizados y controlados por medio de servoamplificadores modelo SVA-2300 de la misma compañía [15], es decir, se trata del servosistema completo.

Los servomotores cuentan con un encoder de cuadratura de 2,500 líneas, el cual, en modo 4x entrega 10,000 pulsos por revolución, proporcionando las dos señales de fase y la señal de índice. Este motor puede suministrar un par continuo de 9.4 Nm y un par máximo de 23.5 Nm, lograr una velocidad nominal de 2000 rpm y una velocidad máxima de 3000 rpm, además, tiene integrado un freno electromagnético con un par de 32 Nm cuando se encuentra desenergizado. El fabricante comercializa este actuador como un motor de media inercia, la cual es de 2.78 x 10^{-3} kg m^2 y tiene un peso aproximado de 19 Kg.

Los servoamplificadores, además de las entradas y salidas para conexión al motor, cuentan con entradas y salidas para desarrollo de aplicaciones. Por ejemplo, tienen salidas para la lectura externa del encoder de cuadratura (en modo diferencial o complementario), 8 entradas y 5 salidas digitales programables. El servoamplificador permite controlar al motor en modo posición, velocidad o torque. En modo torque, se controla por una interfaz analógica de ±10V, con resolución de 10 bit debido al ADC interno. El voltaje de referencia de torque puede generarse de diferentes formas, por ejemplo, con una tarjeta de adquisición para computadora personal, o un sistema embebido con un puerto DAC.

La instalación trifásica se realizó siguiendo la configuración recomendada en el manual del fabricante, es decir, con una caja de fusibles para la etapa de potencia y otra para la etapa de alimentación lógica y de control del servoamplificador. Cuenta con un contactor electromagnético, el cual se enclava con lógica de relevadores para energizar el sistema y se desenclava con botones de apagado y dos botones de seguridad tipo hongo. Uno de estos se encuentra en la caja de controles y otro es móvil, para que se encuentre cerca del operador en todo momento. En la caja de controles también se encuentra un switch para energizar y liberar los frenos electromagnéticos.

Para eliminar el ruido tanto en las señales de los encoders, como en la señal analógica del torque de referencia o control, se diseñó una tarjeta de aislamiento. Para los encoders se usó un optoacoplador de alta velocidad HCPL-2211, y para las señales analógicas se usó un amplificador de aislamiento ISO124P, alimentado con una fuente integrada de aislamiento, modelo DCP010512DBP.

Para la aplicación de la interacción humano-robot en la terapia de rehabilitación, se requiere de un sensor de fuerza con la capacidad de medir por lo menos fuerzas bidireccionales en un plano. Por falta de presupuesto, por el momento no se tiene un sensor de fuerza comercial, ya que la mayoría son de 6 ejes y son costosos. En los laboratorios de la institución se está diseñando un sensor específico para esta aplicación, por lo cual, aún no se tienen resultados de control con retroalimentación de fuerza.

Los experimentos desarrollados hasta el momento han sido sobre control de posición articular. Estos algoritmos han sido implementados en un microcontrolador Atmel AT91SAM3X8E en una interrupción de timer a 4 ms. Se trata de un microcontrolador ARM Cortex M3 de 32 bits a 84 Mhz, el cual tiene dos canales DAC de 12 bit que entregan una señal de voltaje en el rango de 0.56 a 2.76 V. Para que exista compatibilidad en los niveles de voltaje del servoamplificador, se diseñó una tarjeta de acondicionamiento de amplitud y offset, basada en amplificadores operacionales LT1013 y resistencias del 1% tolerancia. Para decodificar la lectura del encoder de cuadratura se utilizan interrupciones externas en puertos I/O de propósito general. Actualmente se analiza la posibilidad de migrar a un microcontrolador más poderoso en caso de que se requiera mayor poder de cómputo para la implementación de técnicas más avanzadas de control.

En la Figura 6 se muestra el sistema eléctrico y electrónico del manipulador. En la zona izquierda aparecen los servoamplificadores, alimentados por el sistema trifásico de la parte superior. En la zona central se encuentran las tarjetas de aislamiento, una fuente a la derecha, y en la zona inferior derecha aparece el sistema embebido y la tarjeta de acondicionamiento de señal analógica.

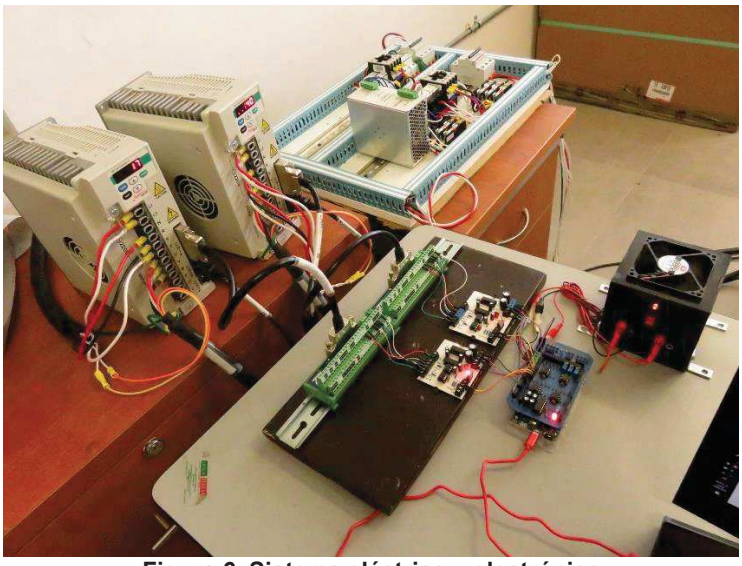


Figura 6. Sistema eléctrico y electrónico

2.3 Interfaz gráfica

Con propósitos de demostración y puesta a punto del sistema de control, se desarrolló una interfaz gráfica en LabVIEW [16]. Esta interfaz, mostrada en la Figura 7, tiene la función de enviar comandos de referencia hacia el sistema embebido para el control de los actuadores, así como de capturar los resultados de los experimentos. Permite visualizar los resultados de forma gráfica en tiempo real para su posterior almacenamiento y análisis. Es importante aclarar que el algoritmo de control se ejecuta en el microcontrolador en una interrupción de timer, como se mencionó anteriormente. La computadora personal y el sistema embebido se comunican por protocolo UART a 230,400 bps, por medio de un convertidor USB/serial.

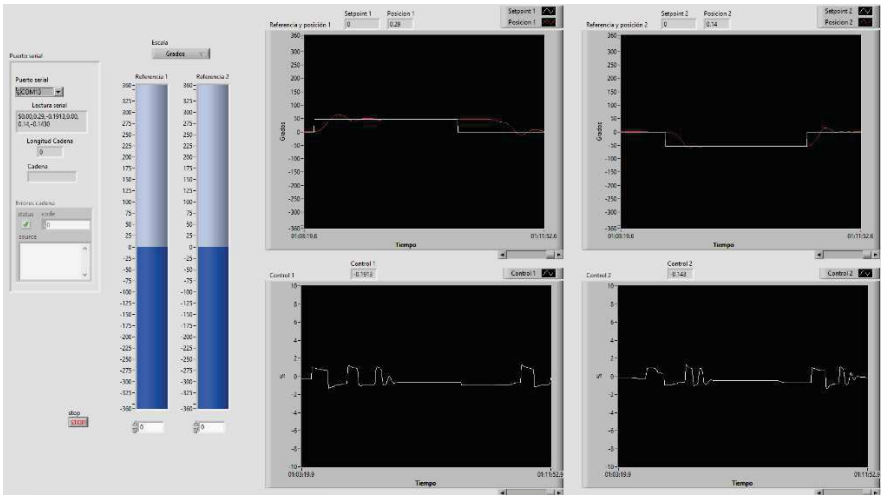


Figura 7. Interfaz gráfica en LabVIEW para mostrar resultados en tiempo real.

En la Figura 8 se muestra un esquema general de la arquitectura funcional del sistema robótico. Para la implementación de la terapia de rehabilitación se contemplan dos interfaces gráficas, la del usuario y la del operador. Para el usuario generalmente se desarrollan aplicaciones interactivas, mientras que, en otra aplicación, el operador puede monitorear variables o parámetros de interés en tiempo real, administrar el sistema y almacenar los resultados.

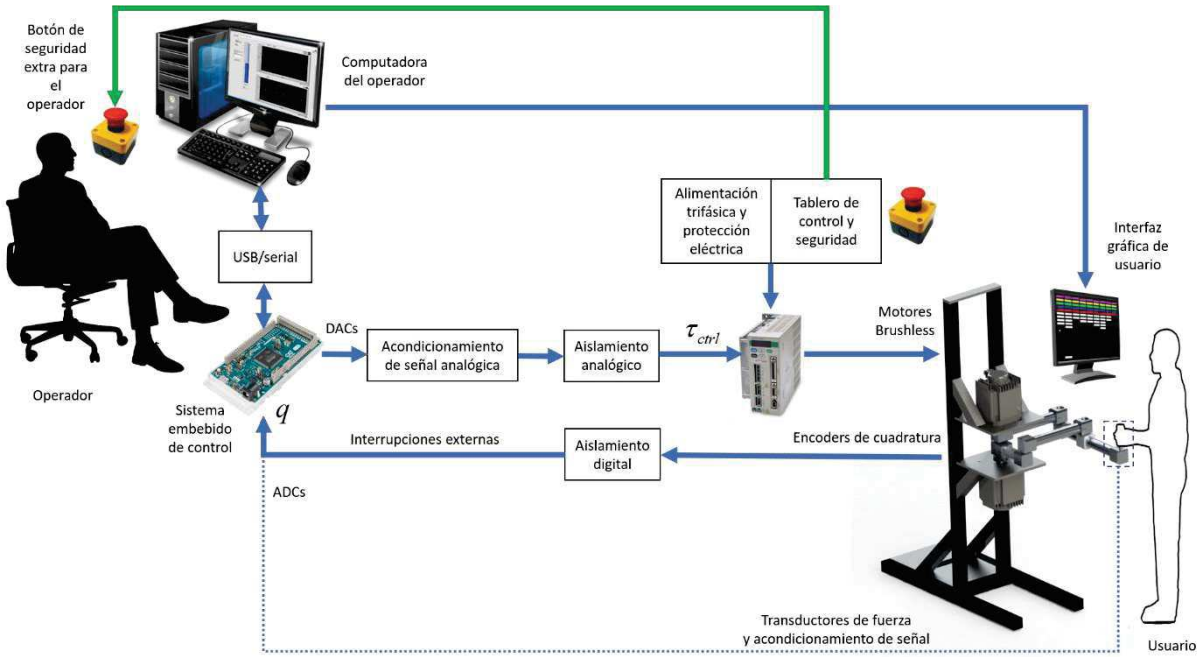


Figura 8. Diagrama funcional del sistema robótico

3. Modelado del manipulador

Los modelos desarrollados para este manipulador corresponden al cinemático directo y el dinámico. Con el modelo cinemático es posible establecer las dimensiones adecuadas para alcanzar el espacio de trabajo requerido para una tarea de rehabilitación, así como calcular la capacidad de fuerza en cualquier dirección, para diferentes puntos de operación. Por otro lado, el modelo dinámico

permite estimar la inercia, masa y fricción percibida por el usuario en el efector final, con el objetivo de evaluar parámetros del diseño mecánico.

3.1 Modelo cinemático

Para desarrollar el modelo cinemático, considere el diagrama de la Figura 9. El origen y dirección para las variables de junta fue establecido de acuerdo con el movimiento de los eslabones en la estructura física del manipulador, observe que tienen dirección contraria ya que las flechas alineadas son opuestas entre sí.

La relación entre los sistemas $x_0y_0z_0$ y $x_1y_1z_1$ está dada por,

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \underbrace{\text{Trans}_x\left(L_1\sin\left(q_1\right)\right)\text{Trans}_y\left(L_1\cos\left(q_1\right)\right)}_{T_{0/1}} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \tag{1}$$

además, para llegar desde el sistema $x_1y_1z_1$ al sistema $x_fy_fz_f$, se puede establecer la siguiente relación,

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \underbrace{\text{Rot}_z\left(q_2\right)\text{Trans}_x\left(L_1\right)}_{T_{1/f}} \begin{bmatrix} x_f \\ y_f \\ z_f \\ 1 \end{bmatrix} \tag{2}$$

de forma que, la matriz de cinemática directa se calcula postmultiplicando la matriz $T_{1/f}$ a la matriz $T_{0/1}$, es decir,

$$\begin{aligned} T_{0/f} &= T_{0/1}T_{1/f} \\ &= \begin{pmatrix} \cos\left(q_2\right) & -\sin\left(q_2\right) & 0 & L_1\sin\left(q_1\right)+L_2\cos\left(q_2\right) \\ \sin\left(q_2\right) & \cos\left(q_2\right) & 0 & L_1\cos\left(q_1\right)+L_2\sin\left(q_2\right) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{3}$$

Con el modelo cinemático directo se calcula el Jacobiano o matriz Jacobiana del manipulador, en este caso es

$$J\left(q\right) = \begin{bmatrix} L_1\cos\left(q_1\right) & -L_2\sin\left(q_2\right) \\ -L_1\sin\left(q_1\right) & L_2\cos\left(q_2\right) \end{bmatrix} \in \mathbb{R}^{2\times 2} \tag{4}$$

El Jacobiano permite conocer las configuraciones singulares del mecanismo, las cuales se presentan cuando $\det\left(J\right)=0$. Tal condición aparece cuando $q_1+q_2=\pm\frac{\pi}{2}$, es decir, cuando los eslabones se encuentran alineados. Si se observa detalladamente la Figura 5, se concluye que físicamente esto no es posible, ya que debido a la forma en que está ensamblado el mecanismo, ocurre una colisión entre los eslabones antes de alinearse en la condición positiva. Al observar la

Figura 2, se puede determinar que tampoco es físicamente posible la condición negativa, ya que ocurre una colisión con la estructura antes de que los eslabones queden alineados.

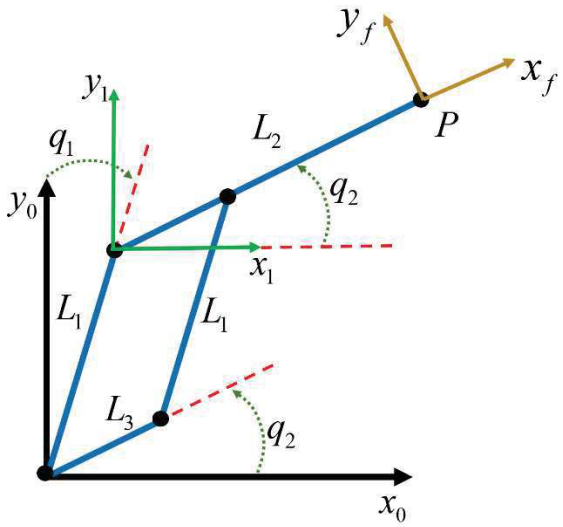
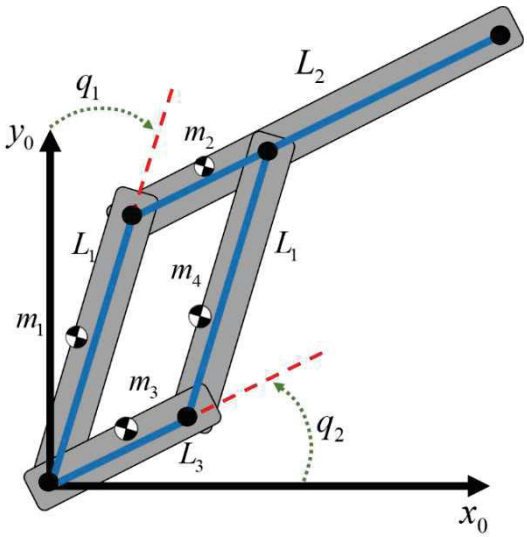


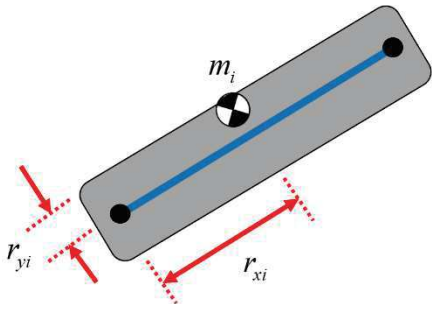
Figura 9. Diagrama cinemático del manipulador

3.2 Modelo dinámico

Para desarrollar el modelo dinámico del manipulador considere el diagrama de la Figura 10 y las ecuaciones de movimiento de Euler-Lagrange [17]. Observe que se plantea que el centro de masa m_i del eslabón i-ésimo se encuentra fuera del eje principal de simetría por defectos de fabricación, es decir, tiene 2 componentes: r_{xi} para la distancia medida a lo largo del eslabón y r_{yi} medida en el eje perpendicular a lo largo y también perpendicular con el eje de rotación del eslabón. Los momentos de inercia alineados al eje de rotación, y calculados en el centro de masa, se denotan como I_{zzi}. La fricción es modelada de la forma tradicional, es decir, viscosa y de Colulomb, con coeficientes b_j y f_{cj}, respectivamente, con $j=1,2$. Éstos corresponden a la fricción vista en las articulaciones activas.



a) Mecanismo completo



b) Ubicación del centro de masa en un eslabón

Figura 10. Diagrama con los centros de masa de los eslabones del manipulador

Debido a que el robot se mueve en un plano horizontal, la energía potencial es constante, por lo tanto, no hay componentes de gravedad en el modelo dinámico, es decir, tiene la siguiente forma:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + f(\dot{q}) = \tau - J^T f_e \tag{5}$$

donde, $q, \dot{q}, \ddot{q} \in \mathbb{R}^2$ son los vectores de posición, velocidad y aceleración articular, respectivamente. $M(q) \in \mathbb{R}^{2\times 2}$ es la matriz de inercia, $C(q,\dot{q}) \in \mathbb{R}^{2\times 2}$ es la matriz de fuerzas centrípetas y de Coriolis, $f(\dot{q}) \in \mathbb{R}^2$ es el vector de pares de fricción viscosa y de Coulomb, τ es el vector de pares de control y f_e es el vector de fuerzas externas aplicadas en el efector final del manipulador.

La energía cinética del manipulador se calcula sumando las energías cinéticas individuales de los eslabones. Con la energía cinética se puede factorizar la matriz de inercia, usando la forma cuadrática con respecto al vector de velocidades articulares,

$$\begin{aligned} K(q,\dot{q}) &= \sum_{i=1}^{n} K_i(q,\dot{q}) \\ &= \frac{1}{2}\dot{q}^T M(q)\dot{q} \end{aligned} \tag{6}$$

donde la energía del eslabón i-ésimo está dado por $K_i(q,\dot{q}) = K_{i,tras}(q,\dot{q}) + K_{i,rot}(q,\dot{q})$, es decir, cada energía cinética individual se calcula con la suma de la energía de traslación de los centros de masa y la energía cinética de rotación de los cuerpos sólidos.

Para este manipulador, se obtienen las siguientes energías cinéticas individuales,

$$\begin{aligned} K_1(q,\dot{q}) &= \frac{1}{2}\dot{q}_1^2\left[I_{zz1} + m_1\left(r_{x1}^2 + r_{y1}^2\right)\right] \\ K_2(q,\dot{q}) &= \frac{1}{2}\dot{q}_1^2\left[m_2 L_1^2\right] + \frac{1}{2}\dot{q}_1\dot{q}_2\left[-2L_1 m_2\left(r_{x2}\sin(q_1+q_2) + r_{y2}\cos(q_1+q_2)\right)\right] \\ &\quad + \frac{1}{2}\dot{q}_2^2\left[I_{zz2} + m_2\left(r_{x2}^2 + r_{y2}^2\right)\right] \\ K_3(q,\dot{q}) &= \frac{1}{2}\dot{q}_2^2\left[I_{zz3} + m_3\left(r_{x3}^2 + r_{y3}^2\right)\right] \\ K_4(q,\dot{q}) &= \frac{1}{2}\dot{q}_1^2\left[I_{zz4} + m_4\left(r_{x4}^2 + r_{y4}^2\right)\right] + \frac{1}{2}\dot{q}_1\dot{q}_2\left[2L_3 m_4\left(r_{y4}\cos(q_1+q_2) - r_{x4}\sin(q_1+q_2)\right)\right] \\ &\quad + \frac{1}{2}\dot{q}_2^2\left[m_4 L_3^2\right] \end{aligned} \tag{7}$$

Para simplificar los cálculos, considere los siguientes parámetros agrupados θ_i,

$$\begin{aligned} \theta_1 &= I_{zz1} + I_{zz4} + m_1(r_{x1}^2 + r_{y1}^2) + m_2 L_1^2 + m_4(r_{x4}^2 + r_{y4}^2) \\ \theta_2 &= -L_1 m_2 r_{x2} - L_3 m_4 r_{x4} \\ \theta_3 &= -L_1 m_2 r_{y2} + L_3 m_4 r_{y4} \\ \theta_4 &= I_{zz2} + I_{zz3} + m_2(r_{x2}^2 + r_{y2}^2) + m_3(r_{x3}^2 + r_{y3}^2) + m_4 L_3^2 \\ \theta_5 &= b_1 \\ \theta_6 &= f_{c1} \end{aligned}$$

$$\theta_7 = b_2$$
$$\theta_8 = f_{c2} \tag{8}$$

Sustituyendo las energías cinéticas individuales de (7) en la expresión (6) y factorizando en su forma cuadrática, se obtiene la matriz de inercia en términos de los parámetros definidos en (8),

$$M(q) = \begin{bmatrix} \theta_1 & \theta_2 \sin(q_1+q_2) + \theta_3 \cos(q_1+q_2) \\ \theta_2 \sin(q_1+q_2) + \theta_3 \cos(q_1+q_2) & \theta_4 \end{bmatrix} \tag{9}$$

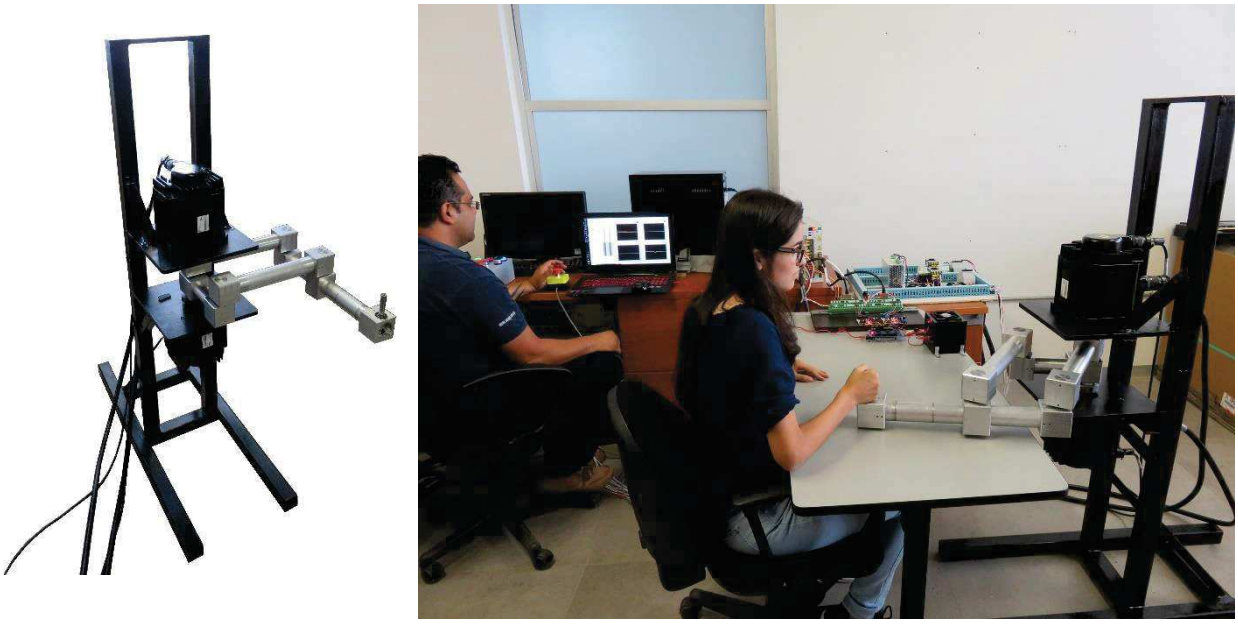
Para encontrar los elementos de la matriz de Coriolis $C(q,\dot{q})$ se usan los símbolos de Christoffel, la cual está dada por,

$$C(q,\dot{q}) = \begin{bmatrix} 0 & \dot{q}_2\left(\theta_2 \cos(q_1+q_2) - \theta_3 \sin(q_1+q_2)\right) \\ \dot{q}_1\left(\theta_2 \cos(q_1+q_2) - \theta_3 \sin(q_1+q_2)\right) & 0 \end{bmatrix} \tag{10}$$

Además, el vector de fricción $f(\dot{q})$ modelado de la forma tradicional, está dado por,

$$f(\dot{q}) = \begin{bmatrix} \theta_5 \dot{q}_1 + \theta_6 \operatorname{sgn}(\dot{q}_1) \\ \theta_7 \dot{q}_2 + \theta_8 \operatorname{sgn}(\dot{q}_2) \end{bmatrix} \tag{11}$$

Contar con el modelo de forma explícita y sus parámetros permite calcular algunos parámetros para evaluar el diseño mecánico.



a) Manipulador construido

b) Montaje simulando la terapia de rehabilitación con paciente y operador

Figura 11. Sistema robótico instalado en el laboratorio

4. Resultados

En esta sección se muestran y analizan los resultados obtenidos en el diseño del manipulador. En la Figura 11a se muestra una fotografía del manipulador construido y en la Figura 11b se muestra un montaje del sistema robótico simulando la terapia de rehabilitación con una persona sana y el

operador monitoreando los resultados en tiempo real. Observe que el operador tiene el botón de paro de emergencia cerca de él, por motivos de seguridad. Este botón es móvil, por lo que lo puede traer en las manos, aunque no esté cerca de la computadora de monitoreo.

4.1 Espacio de trabajo

Para encontrar las dimensiones de los eslabones de forma que permitieran un espacio de trabajo adecuado para las tareas de rehabilitación (aproximadamente 38 x 45 cm), y que, a su vez, se cumplan las características necesarias para la tarea de rehabilitación, relacionadas con la fuerza, inercia y fricción, se emplearon herramientas computacionales, específicamente Matlab y SolidWorks, con un procedimiento iterativo basado en la experiencia. Las dimensiones de los eslabones, mostradas en la Figura 9 son las siguientes,

$$L_1 = 0.35 \text{ m}$$
$$L_2 = 0.45 \text{ m}$$
$$L_3 = 0.19 \text{ m}$$

Se realizó un estudio de colisiones en SolidWorks para determinar las condiciones de operación para evitar la colisión entre eslabones y la base. Estas condiciones son,

$$q_1 > -38°$$
$$q_2 > -124° \tag{12}$$

además, para evitar colisiones entre eslabones, debe cumplirse la siguiente condición de operación,

$$-72° < q_1 + q_2 < 72° \tag{13}$$

Al considerar las restricciones cinemáticas dadas por (12) y (13), se obtiene el espacio de trabajo dibujado en la Figura 12. Observe que en esta figura aparecen dos rectángulos de 38 x 45 cm, candidatos para desarrollar las tareas de interacción. En los análisis posteriores se considera el rectángulo que no presenta inclinación.

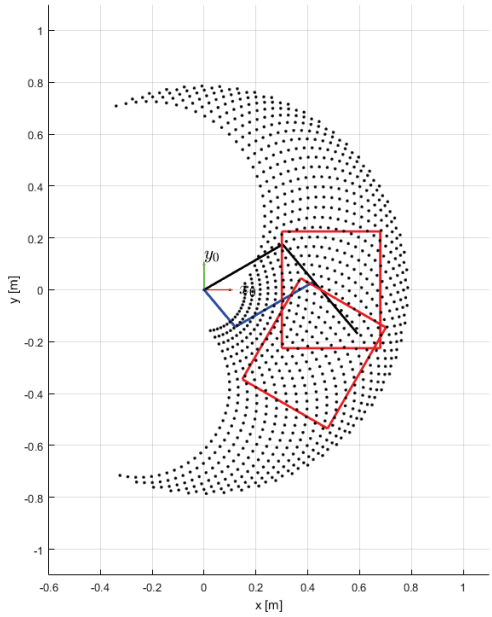


Figura 12. Espacio de trabajo del manipulador

4.2 Parámetros de desempeño

Como se mencionó anteriormente, los actuadores suministran 9.4 Nm de par nominal y 23.5 Nm de par máximo. Tomando en consideración estos valores, es posible calcular la fuerza que puede ejercer el efector final en cualquier dirección, empleando la relación Jacobiano transpuesto $\tau = J^T F$. Al analizar las posibles configuraciones en el espacio de trabajo propuesto, se determinó que con el torque nominal se obtiene una fuerza máxima de 15 N en cualquier dirección y hasta 67 N en algunas direcciones. Con el valor del torque pico se obtiene una fuerza máxima de 38 N en cualquier dirección y 167 N algunas direcciones, lo cual cumple de forma muy cercana con los requerimientos de diseño.

En la gráfica de la Figura 13 se muestran los elipsoides de fuerza en el espacio de trabajo seleccionado. El eje mayor (rojo) representa la capacidad máxima de fuerza en esa dirección para ese punto de operación, mientras que el eje menor (azul) representa la mínima cantidad de fuerza para el mismo punto de operación. Los elipsoides de fuerza se calculan con la siguiente desigualdad, por medio de la descomposición en valores singulares [18],

$$F\left[J(q)J(q)^T\right]F \leq 1 \tag{14}$$

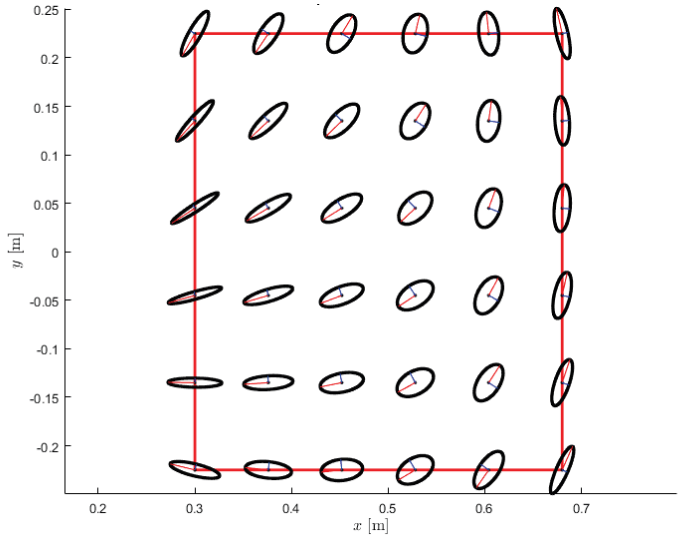


Figura 13. Elipsoides de fuerza en el espacio de trabajo seleccionado

La expresión para inercia generalizada para este manipulador es, en efecto, la expresión para masa generalizada, considerando que el Jacobiano que aparece es el de la velocidad lineal. Ésta permite calcular la masa percibida por el usuario en el efector final y está dada por [19],

$$\Lambda(q) = \left(J(q)M^{-1}(q)J(q)^T\right)^{-1} \tag{15}$$

de forma que el elipsoide de inercia se grafica con la siguiente desigualdad,

$$u^T \Lambda u \leq 1 \tag{16}$$

Para realizar los cálculos, se tomaron los parámetros obtenidos del modelo en SolidWorks y la hoja de datos del actuador y los baleros. En la Tabla 1 se muestran los valores numéricos de los parámetros agrupados.

Tabla 1. Valores numéricos de los parámetros agrupados del modelo dinámico

Parámetro	Valor	Unidades
θ_1	0.437749	Nm s^2/rad
θ_2	-0.305427	Nm s^2/rad
θ_3	0.000000	Nm s^2/rad
θ_4	0.517612	Nm s^2/rad
θ_5	0.005000	Nm s/rad
θ_6	0.001000	Nm
θ_7	0.005000	Nm s/rad
θ_8	0.001000	Nm

Al analizar los resultados en Matlab, se determinó que la masa máxima efectiva percibida en el efector final, en cualquier dirección es de 203 g y hasta de 405 g en algunas direcciones, lo cual es un resultado relativamente bajo. Con estos parámetros se prevé que las personas con poca fuerza muscular (pacientes) serán capaces de mover el manipulador. En la gráfica de la Figura 14 se muestran los elipsoides de inercia. Al igual que en el caso anterior, el eje mayor representa la mayor masa percibida en esa dirección para ese punto de operación, mientras que el eje azul representa la menor.

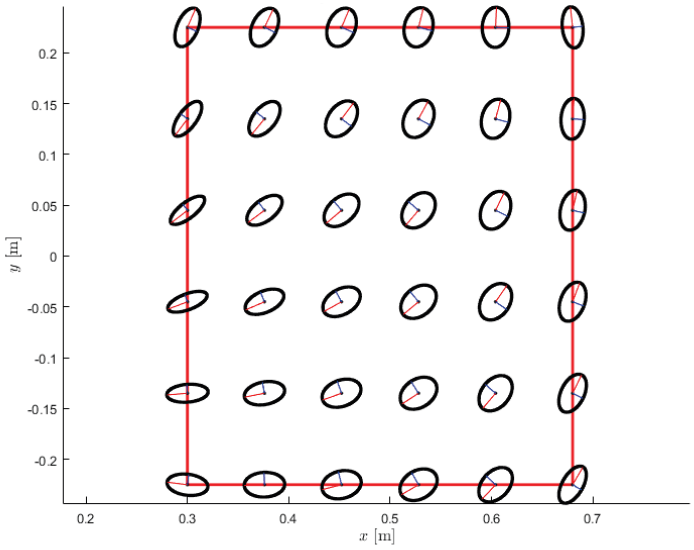


Figura 14. Elipsoides de inercia en el espacio de trabajo seleccionado

La expresión para la fricción generalizada permite calcular la fricción percibida en el efector final, la cual debe ser baja para que responda con alta capacidad de fuerza a velocidades bajas. La fricción viscosa reflejada en el efector final es la siguiente [13],

$$\Omega(q) = \left(J(q) B^{-1} J(q)^T \right)^{-1} \tag{17}$$

donde B es una matriz diagonal que contiene los coeficientes de fricción viscosa. El elipsoide de fricción se calcula con la siguiente desigualdad,

$$u^T \Omega u \leq 1 \tag{18}$$

Al analizar los resultados en Matlab, se determinó que la fricción viscosa máxima percibida en el efector final es de 0.0188 Ns/m para cualquier dirección y 0.0828 Ns/m para algunas direcciones, la cual es baja. En la gráfica de la Figura 15 se muestran los elipsoides de fricción viscosa. Al igual que en el caso anterior, el eje mayor representa la mayor fricción percibida en esa dirección para ese punto de operación, mientras que el eje azul representa la menor.

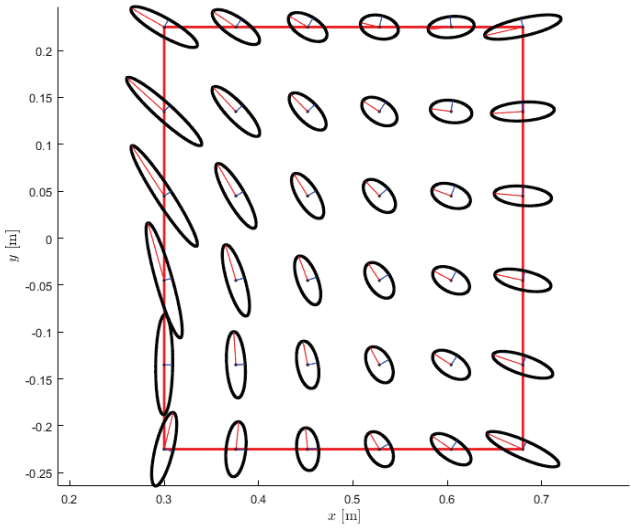


Figura 15. Elipsoides de fricción viscosa en el espacio de trabajo seleccionado

Analizando los resultados obtenidos, se determina que el diseño cumple con los requerimientos de espacio de trabajo, características de fuerza, inercia y fricción reflejadas en el efector final para una tarea de rehabilitación.

Para poner en marcha el manipulador, se implementó un controlador de posiciones articulares con acciones acotadas, específicamente el controlador tanh, muy conocido en la literatura de control de manipuladores [20]. El controlador tiene la siguiente forma,

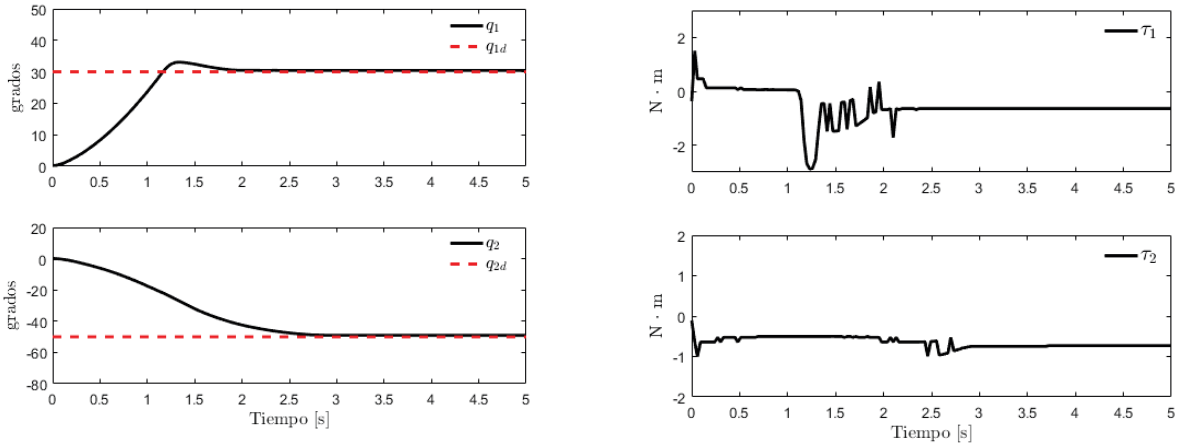
$$\tau = K_p \tanh\left(\tilde{q}\right) - K_d \tanh\left(\dot{q}\right) \tag{19}$$

donde $\tilde{q} = q_d - q\left(t\right)$.

En el experimento el robot parte de la posición de casa, es decir $q_1 = 0$ y $q_2 = 0$, además se le proporcionaron las referencias $q_{d1} = 30°$ y $q_{d2} = -50°$. Se usaron las siguientes ganancias de sintonía,

$$K_p = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.0 \end{bmatrix}$$
$$K_d = \begin{bmatrix} 1.45 & 0 \\ 0 & 0.5 \end{bmatrix} \tag{20}$$

En la Figura 16 se muestran los resultados. En la gráfica del lado izquierdo aparece la referencia contra la posición, mientras que en la gráfica del lado derecho se muestran los torques de control aplicados. Observe que los pares de control no llegan a sus límites de saturación.



a) Referencia contra posición b) Pares de control

Figura 16. Resultados del experimento de control de posición

5. Conclusiones

En este trabajo se presentó el diseño de un manipulador planar de arquitectura abierta, que servirá como una plataforma de experimentación para nuevos esquemas de control de interacción. El manipulador presenta las características dinámicas deseables para un robot de rehabilitación de extremidades superiores, específicamente para hombro y codo. El diseño ha sido validado empleando valores numéricos obtenidos del diseño en SolidWorks y las hojas de datos los dispositivos empleados. Es importante mencionar que no se han implementado esquemas de control de interacción que requieran la retroalimentación de la fuerza de contacto robot-entorno, ya que por el momento no se cuenta con un sensor comercial para este fin, principalmente por la falta de presupuesto, en su lugar, en los laboratorios de la institución se está diseñando un sensor de fuerza bidireccional para un plano. Por el momento, para demostración del funcionamiento de la plataforma experimental, se implementó un controlador de posición articular ampliamente conocido en la literatura.

Se desarrolló una interfaz en LabVIEW para conocer la respuesta del sistema en tiempo real de forma gráfica, con la posibilidad de almacenar los resultados para su análisis posterior. Los algoritmos de control corren en un sistema embebido a 4 ms, basado en un microprocesador ARM Cortex M3 de Atmel, sin embargo, se analiza la posibilidad de migrar a un microcontrolador de mejor desempeño.

Agradecimientos

Este trabajo fue financiado con el proyecto PRODEP UAA-PTC-178. Se reconoce y se agradece públicamente a los estudiantes por su asistencia técnica en la manufactura de partes y tarjetas electrónicas: Enrique Sánchez, Gerardo García, Diana Martínez, Fernando Barba, César Alcayde y Jorge Eudave, así como a los técnicos Ed Palomino y Victor Morán.

Referencias

[1] International Federation of Robotics, "*Introduction into Service Robots*". World Robotics 2016 Edition.
[2] Proietti T., Crocher V., Roby-Brami A., Jarrasse N., "*Upper-Limb Robotic Exoskeletons for Neurorehabilitation: A Review on Control Strategies*". Reviews in Biomedical Engineering. IEEE. Vol. 9. p.p. 4-14, 2016.
[3] World Health Organization. "*Global status report on noncommunicable diseases 2014*". Geneva, Switzerland. 2014

[4] World Health Organization. Sitio web: http://www.who.int [Último acceso: Agosto 2018].
[5] Richards C., Malouin F., Nadeau S., "*Stroke rehabilitation: Clinical picture, assessment, and therapeutic challenge*", Sensorimotor Rehabilitation: At the Crossroads of Basic and Clinical Sciences. Elsevier, Vol. 218, pp. 253–280. 2015.
[6] Volpe B. "*Robot training enhanced motor outcome in patients with stroke maintained over 3 years*", Neurology. Vol 53, No. 8, p.p. 1874-1876, 1999.
[7] Volpe B., Lynch D., Rykman-Berland A., Ferraro M., Galgano M., Hogan N., Krebs H., "*Intensive sensorimotor arm training mediated by therapist or robot improves hemiparesis in patients with chronic stroke*". Neurorehabilitation and Neural Repair. Vol 22, No. 3, p.p. 305-310, 2008.
[8] Mazzoleni S., Turchetti G., Palla I., Posteraro F., Dario P., "*Acceptability of robotic technology in neuro-rehabilitation: Preliminary results on chronic stroke patients*". Computer Methods and Programs in Biomedicine, Vol. 116, No. 2, p.p. 116 -122, 2014.
[9] Chong L., Zoltán R., Imre H., Linhong J., "*Influence of complementing a robotic upper limb rehabilitation system with video games on the engagement of the participants: a study focusing on muscle activities*". International Journal of Rehabilitation Research. Vol. 37, No. 4, p.p. 334-342, 2014.
[10] Riener R., "*Transferring ARMin to the clinics and industry*". Topics Spinal Cord Injury Rehabilitation. Vol. 17, No. 1, p.p. 54-59, 2011.
[11] Calabro R., Russo M., Naro A., Milardi D., Balleta T., Leo A., Filoni S., Bramanti P., "*Who May Benefit From Armeo Power Treatment? A Neurophysiological Approach to Predict Neurorehabilitation Outcomes*". PM&R , Vol. 8, No. 10 , p.p. 971 – 978, 2016.
[12] Charnnarong J., "*The design of an intelligent machine for upper-limb physical therapy*". Ph.D. dissertation, Massachusetts Institute of Technology. Dept. of Mechanical Engineering, Agosto 1991
[13] Zollo L., Salerno A., Vespignani M., "*Dynamic characterization and interaction control of the CBM-Motus Robot for upper limb rehabilitation*". International Journal of Advanced Robotic Systems. Vol. 10, 374:2013.
[14] SureServo, "*SVM 220 & 220 B Motor specifications*". En línea: http://www.sureservo.com/svm_220.htm [Último acceso: Agosto 2018]
[15] SureServo, "*SVA 2300 Driver specifications*". En línea: http://www.sureservo.com/sva_2300.htm [Último acceso: Agosto 2018]
[16] National Instruments "*LabVIEW*". En línea: http://www.ni.com/es-mx/shop/labview.html [Último acceso: Agosto de 2018]
[17] Kelly R., Santibañez V., Loria A. "*Control of Robot Manipulators in Joint Space*", Springer, 2005.
[18] Cassadio M., Sanguineti V., Morasso P., Arrichiello V., "*Braccio di Ferro: a new haptic workstation for neuromotor rehabilitation*". Technology and health care: official journal of the European Society for Engineering and Medicine, Vol 14, No 3, p.p. 123-142, 2006.
[19] Khatib O., "*Inertial Properties in Robotic Manipulation: An Object-Level Framework*", International Journal of Robotics Research, Vol. 14, No. 1, p.p. 19-36, 1995.
[20] Reyes F. "*Robótica: control de robots manipuladores*". Alfaomega 2011.

Ingeniería y Automatización, Capítulo 16, pp. 205 - 223.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño y Plan de Manufactura de una Plataforma Experimental para Control: Péndulo Invertido Inercial

Hernández Gallardo Julián Alejandro, Pacheco Villegas Jesús Eduardo, Désiga Orenday Omar, Talavera Otero Jorge y Cuevas Cortés Carlos

Instituto Politécnico Nacional, Unidad Profesional Interdisciplinaria de Ingeniería campus Zacatecas.

jhernandezg1313@alumno.ipn.mx, jpachecov1300@alumno.ipn.mx, odesigao@ipn.mx, jtalaveraot@ipn.mx, ccuevas@ipn.mx

Resumen

En este documento se presenta el diseño mecánico de una plataforma experimental del sistema subactuado Péndulo Invertido Inercial. Se estudió la dinámica no lineal del sistema por medio del método de Euler-LaGrange para el diseño del control, y por el método recursivo de Newton-Euler para el análisis mecánico. Se dedujeron requerimientos de torque para el actuador a través del diseño, simulación y sintonización de un controlador híbrido, compuesto por un controlador lineal y uno no lineal. Se realizó una selección de los componentes electrónicos, mecánicos y materiales que se utilizarán para la construcción del sistema, así como un modelo CAD detallado.

Palabras clave: Péndulo Invertido Inercial, Newton-Euler, Euler-LaGrange

1. Introducción

Se conoce como péndulo a un cuerpo que puede oscilar suspendido de un punto por un hilo o una varilla [1]. El Péndulo Invertido es un sistema dinámico inestable en lazo abierto, comúnmente usado como referencia para investigación en control desde los años 50 (modelado matemático, puntos de equilibrio, análisis de estabilidad, entre otras) por su dinámica simple, lo que permite probar distintas estrategias de control. De este sistema dinámico existen diferentes configuraciones que dependen de la forma en que actúa (grados de libertad) y resultado a esto, aumenta o disminuye su complejidad de controlabilidad [1].

Los sistemas pendulares, son un tipo de mecanismos subactuados, los cuales emulan el movimiento característico oscilatorio de un péndulo simple, entre los que destacan: el péndulo invertido simple, el péndulo invertido lineal, el péndulo de Furuta o invertido rotacional, el péndulo invertido con rueda inercial (IWP, por sus siglas en inglés), el acrobot, entro otros.

El Péndulo Invertido Inercial, originalmente nombrado como Reaction Wheel Pendulum (RWP), fue introducido por Mark W. Spong [2], posteriormente K. J. Åström en colaboración con Spong en [3] lo renombraron como Inertia Wheel Pendulum (IWP), nombre con el que ahora se le identifica. Desde un punto de vista mecánico, el Péndulo Invertido Inercial es un péndulo simple con una rueda o volante rotatorio en su extremo. La rueda está sujeta al eje de un motor de DC y el torque generado del acoplamiento entre estos elementos al eje del péndulo puede ser usado para controlar el movimiento del sistema. Se puede pensar en este sistema como un péndulo simple con una inercia controlada por un par [3].

2. Métodos y materiales

Para dar inicio al proceso de diseño, se dividió el trabajo en áreas y etapas, y con esto determinar un diagrama de proceso mostrado en la figura 1. El desarrollo del proyecto se realizó bajo la metodología del diseño mecatrónico, este lleva al diseño más óptimo:

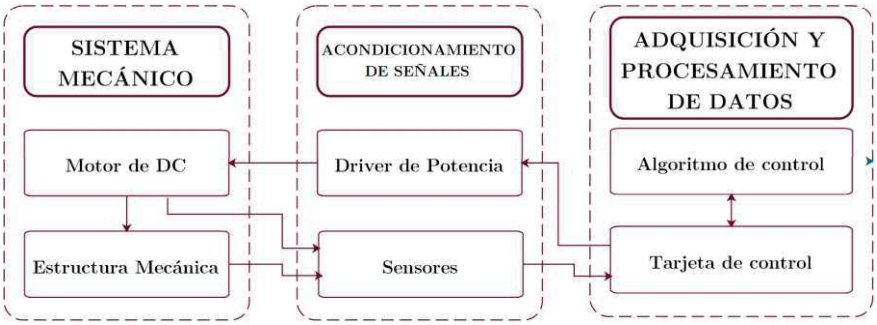


Figura 1. Áreas del proceso de diseño.

Para diseñar estas áreas, se siguió un proceso concurrente, puesto que la finalización del diseño de un área esté relacionada con el inicio de alguna otra.

2.1 Modelo dinámico por Euler-LaGrange

La obtención del modelo dinámico se basó en el procedimiento propuesto por M. W. Spong [2], C. A. Merlo Zapata [4] y D. Alonso [5]. La figura 2 muestra el diagrama esquemático para la convención de ángulos.

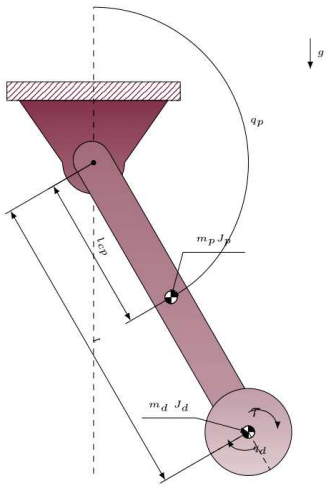


Figura 2. Representación esquemática del sistema.

El método de Euler-LaGrange dice que se obtendrá una ecuación para cada coordenada generalizada del sistema, el Péndulo Invertido Inercial posee dos, las cuales se representan por las siguientes ecuaciones

$$\left(m_p l_{cp}^2 + m_d l^2 + J_p + J_d\right)\ddot{q}_p + J_d \ddot{q}_d - \left(m_p l_{cp} + m_d l\right) g \sin\left(q_p\right) = 0 \tag{1}$$

$$J_d\left(\ddot{q}_p + \ddot{q}_d\right) = \tau(t) \tag{2}$$

Donde:

- $g \equiv$ constante de la aceleración de gravedad
- $m_p \equiv$ masa del péndulo
- $J_p \equiv$ momento de inercia del péndulo
- $l_{cp} \equiv$ distancia del centro de gravedad del péndulo al origen
- $l \equiv$ distancia del centro de gravedad del disco al origen
- $m_d \equiv$ masa del disco
- $J_d \equiv$ momento de inercia del disco
- $q_p \equiv$ posición angular del péndulo respecto al origen
- $q_d \equiv$ posición angular del disco respecto al péndulo
- $\tau(t) \equiv$ fuerza generalizada aplicada sobre el sistema

Con el modelo obtenido anteriormente, se concluyó que:

- La representación en el espacio de estados del sistema linealizado requiere sólo tres variables de estado: q_p, $\dot{q}_p$ y $\dot{q}_d$.
- Dado que es un sistema conservativo, se puede estudiar el sistema a través de su energía para proponer una estrategia de control no lineal.

2.2 Modelo dinámico por Newton-Euler

El objetivo de trasladar las fuerzas y torques del sistema representado como cadena cinemática desde el disco hasta la base, para posteriormente utilizar las ecuaciones en el diseño mecánico, el procedimiento se basó en el trabajo desarrollado por Warren White [6]. En la figura 3, se muestra el diagrama de cuerpo libre del péndulo y del disco respectivamente.

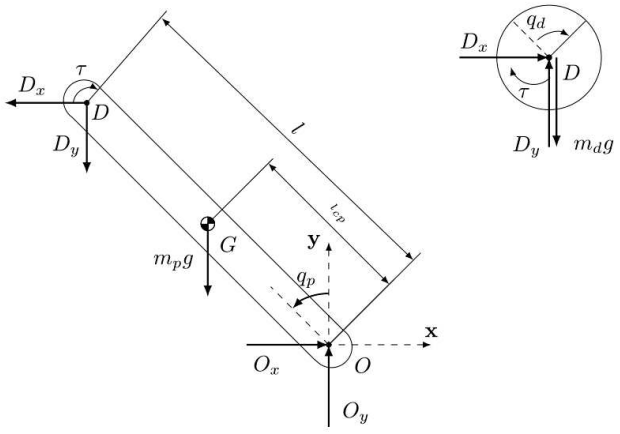


Figura 3. Diagrama de Cuerpo Libre del Péndulo Invertido Inercial.

Después de analizar los diagramas de cuerpo libre, se deben obtener las ecuaciones que representen las fuerzas presentes en el punto O (O_x, O_y), las ecuaciones se obtuvieron utilizando la segunda ley de Newton y la ecuación de rotación de Euler, las cuales se expresan en las (3) y (4).

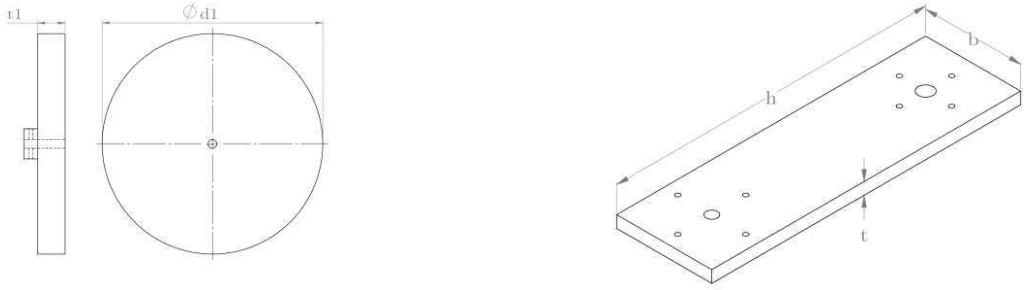
$$O_x = m_p l_{cp} \cos\left(q_p\right)\ddot{q}_p - m_d \ddot{q}_p l \cos\left(q_p\right) + \dot{q}_p^2 l \sin\left(q_p\right) \quad (3)$$

$$O_y = m_p l_{cp} \sin\left(q_p\right)\ddot{q}_p - m_d \ddot{q}_p l \sin\left(q_p\right) + m_d \dot{q}_p^2 l \cos\left(q_p\right) \quad (4)$$

Se concluyó que las reacciones en el punto O dependen directamente del momento y la fuerza centrípeta que se generan por la aceleración, velocidad y posición del péndulo ($\ddot{q}_p$, $\dot{q}_p$ y q_p).

2.3 Parametrización del modelo

El primer paso de esta etapa de diseño fue particularizar los parámetros del modelo dinámico, haciendo uso de las propiedades del material seleccionado (Aluminio 6061 T6) y el asistente de cálculo de masas de SOLIDWORKS®. Cabe mencionar, que se eligieron geometrías básicas con dimensiones promedio para el péndulo y el disco de inercia, obtenidos de [4], [7], [8].



a) Disco de inercia. **b) Péndulo.**

Figura 4. Geometrías básicas del péndulo y disco de inercia.

2.4 Control de Estabilización

El primer paso para el diseño del controlador es la tarea de estabilizar al péndulo en su punto de equilibrio no estable, para esto a través de la teoría de control moderno ya conocida se busca proponer un controlador que mantenga al péndulo en el punto de equilibrio inestable ($q_p = 0$). Al iniciar con el diseño, lo primero y más importante es eliminar el término no lineal ($\sin\left(q_p\right)$), de la ecuación de movimiento de q_p , ecuación (1). Un primer enfoque puede ser el realizar una expansión en serie de Taylor del seno en torno a 0. Se puede demostrar que esta linealización es aceptable en un rango de ± 0.1818 rad (aproximadamente $\pm 10.417°$) con un margen de error $\left|q_p - \sin\left(q_p\right)\right| < 1\times10^{-3}$.

Para los posteriores cálculos de optó por reescribir las ecuaciones (1) y (2) en su forma matricial, la cual se muestra a continuación:

$$\begin{bmatrix} m_p l_{cp}^2 + m_d l^2 + J_p + J_d & J_d \\ J_d & J_d \end{bmatrix}\begin{bmatrix} \ddot{q}_p \\ \ddot{q}_d \end{bmatrix} + \begin{bmatrix} -\left(m_p l_{cp} + m_d l\right) g \sin\left(q_p\right) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau(t) \end{bmatrix} \quad (5)$$

Se observó que la matriz es invertible ya que su determinante es diferente de 0. Resolviendo para $\ddot{q}$ se obtuvo que cuando el péndulo se encuentra en $q_p = 0$, la velocidad del disco necesariamente tiene que ser constante. Posteriormente se definió el modelo lineal aproximado en el espacio de estados, del cual se demuestra su controlabilidad mediante una retroalimentación completa de los estados, de la forma:

$$\tau = -\mathbf{K}z \quad (6)$$

Ingeniería y Automatización, Capítulo 16, pp. 205 - 223.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Se aplicó el método de Ackerman para la colocación de polos [9], por lo que se escribió el sistema en su forma canónica controlable. Inicialmente se seleccionaron arbitrariamente los siguientes conjuntos de polos:

$$s_{1,2} = -5 \pm 5\sqrt{2}i \qquad s_3 = -\sqrt{2} \tag{7}$$

Aplicando el método anteriormente descrito se obtuvieron valores para el vector $\mathbf{K}$. Por último, se realizó la simulación del controlador diseñado, la figura 5 muestra el diagrama de bloques del controlador.

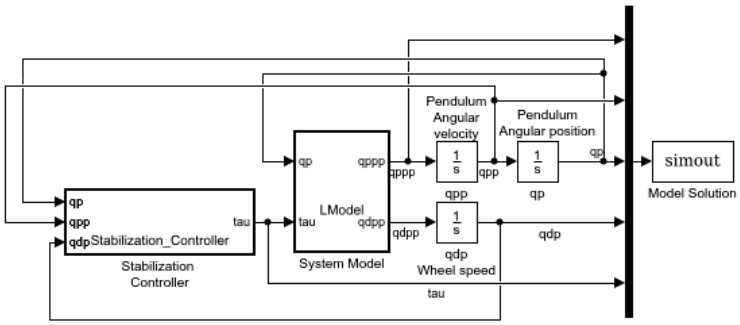
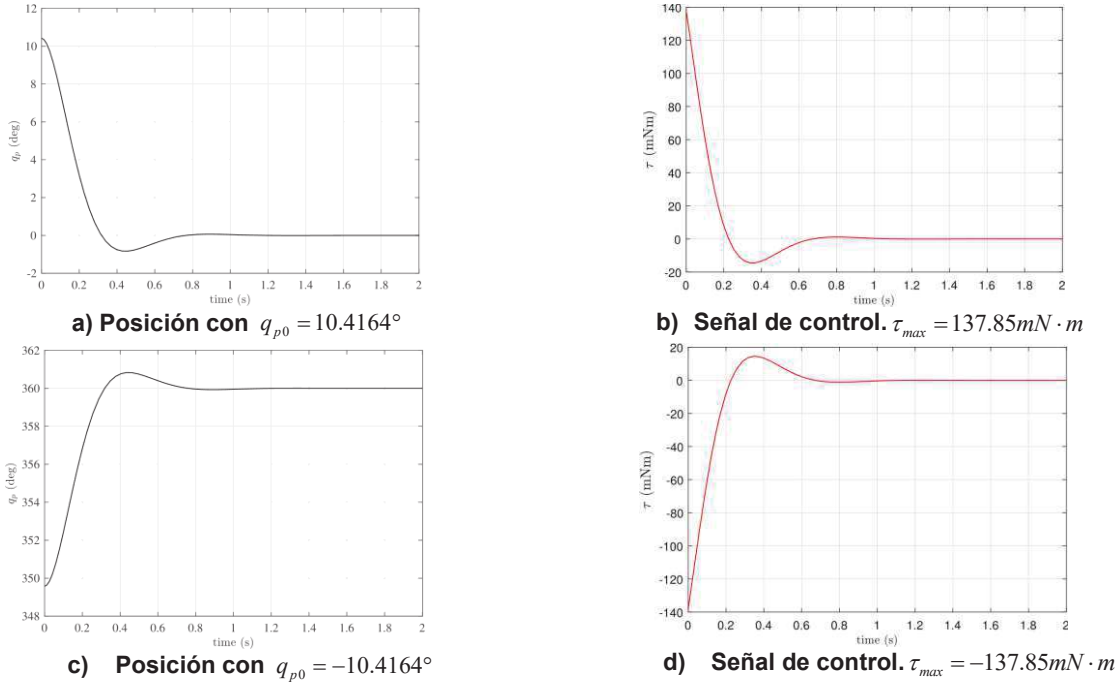


Figura 5. Diagrama de bloques del controlador de estabilización

Para probar el comportamiento del controlador se hicieron dos pruebas sobre el límite en el cuál la linealización es válida (± 0.1818 rad), las cuales se muestran en la figura 6.



a) Posición con $q_{p0} = 10.4164°$

b) Señal de control. $\tau_{max} = 137.85mN \cdot m$

c) Posición con $q_{p0} = -10.4164°$

d) Señal de control. $\tau_{max} = -137.85mN \cdot m$

Figura 6. Respuesta del controlador de estabilización.

2.5 Control Swing up

La siguiente tarea es el diseño del controlador, que consiste en tomar el péndulo a partir de su posición de equilibrio estable y llevarlo a un rango cercano al punto de equilibrio inestable, a esta tarea de control se le conoce como balanceo (*Swing-up*). La mayor dificultad reside en la no linealidad del sistema, ya que no existe una linealización en el intervalo completo como para proponer una estrategia de control lineal, con esto se descarta el uso de estrategias de control como PID o alguna otra en el espacio de estados, por lo que es necesario pensar en la estrategia del sistema de control a través de otra estrategia.

Se aborda la estrategia de control aplicada en [2], [3], [4], [7] y [8] que consiste en la regulación de la energía a través de la deducción de una condición de pasividad de la energía del sistema y del uso de funciones de saturación.

Un sistema pasivo no puede entregar más energía de la que está recibiendo, es decir un sistema no lineal que es pasivo se puede estabilizar haciendo una realimentación negativa de la salida $u = -k\tau$, con $k > 0$.

Por medio de una función de Lyapunov definida positiva, K. J. Åström [3] e I. Fantoni [9] presentan una demostración de la pasividad del sistema. Tomando el desarrollo de [3] y [9] se despejó $\ddot{q}_d$ de la ecuación (2) y se sustituyó en la ecuación (1), obteniendo:

$$\left(m_p l_{cp}^2 + m_d l^2 + J_p\right)\ddot{q}_p + J_d \ddot{q}_d - \left(m_p l_{cp} + m_d l\right) g \sin\left(q_p\right) = -\tau(t) \tag{8}$$

La ecuación (8) permite analizar el sistema descrito por las ecuaciones (1) y (2) como un péndulo simple, cuestión que simplifica el análisis. En la figura 7 se contrastan la energía de un péndulo y otro.

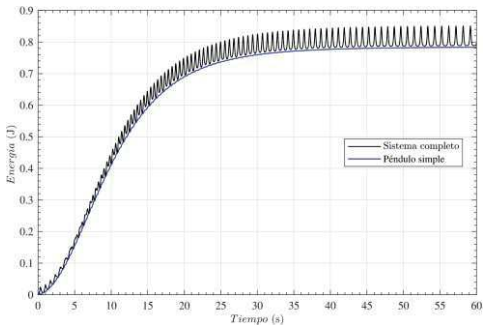


Figura 7. Energía del sistema

De la ecuación (8) se puede demostrar que la energía total del sistema es:

$$E = \frac{1}{2}\left(m_p l_{cp}^2 + m_d l^2 + J_p\right)\dot{q}_p^2 + \left(m_p l_{cp} + m_d l\right) g \left(1 + \cos\left(q_p\right)\right) \tag{9}$$

Y la propiedad pasiva del sistema es:

$$\frac{dE}{dt} = \left(m_p l_{cp}^2 + m_d l^2 + J_p\right)\dot{q}_p \ddot{q}_p - \left(m_p l_{cp} + m_d l\right) g \dot{q}_p \sin\left(q_p\right) = -\dot{q}_p \tau(t) \tag{10}$$

En [8] se evaluó la energía en el punto $q_{p0} = 2k\pi$ con $k = 0, 1, 2, \ldots, n$, se obtuvo la condición de pasividad de la energía, ecuación (11), lo que servirá como valor de referencia para la estrategia de control.

$$E_o = 2\left(m_p l_{cp} + m_d l\right) g \tag{11}$$

La entrada de control que R. Silva Ortigoza propone en [8] toma la forma de:

$$-\tau = -k_d \left(E - E_o\right) sat\left(\dot{q}_p\right) \tag{12}$$

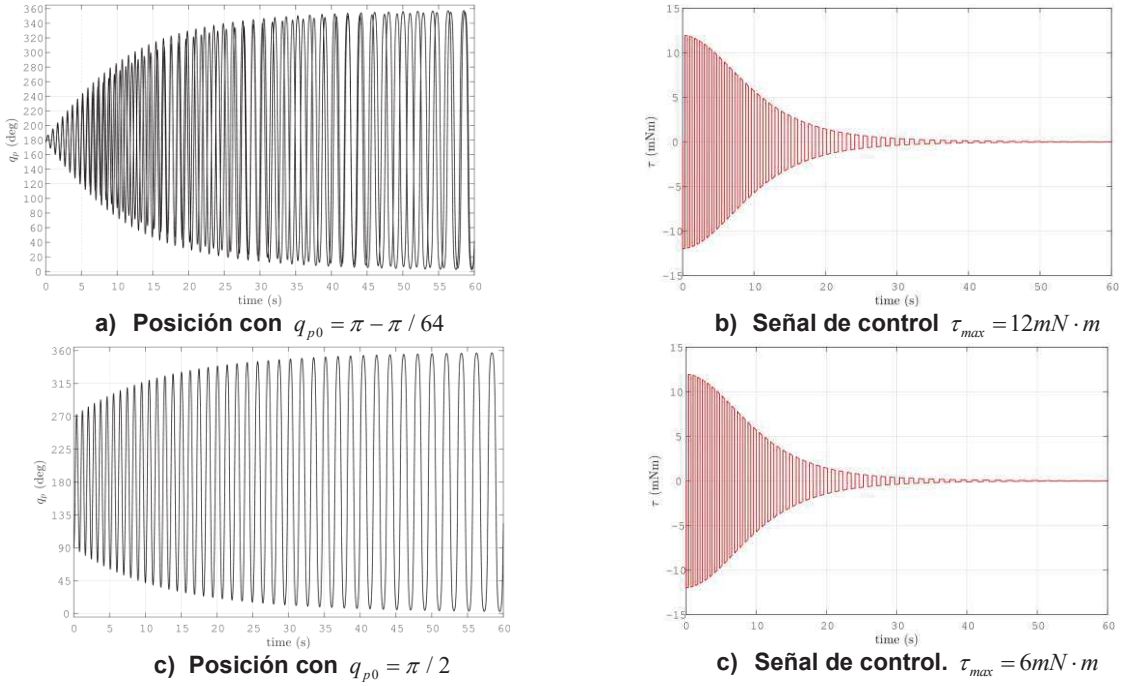
$$sat\left(\dot{q}_p\right) = q\begin{cases} d & si \quad \dot{q}_p > d \\ d & si \quad -\dot{q}_p < -d \\ \dot{q}_p & si \quad \left|\dot{q}_p\right| \le d \end{cases} \tag{13}$$

$$d = \frac{V_{max} k_{vm}}{k_d R E_o} = \frac{k_{mi} i_{max}}{k_d E_o} \tag{14}$$

Donde:

- $V_{max} \equiv$ voltaje máximo del motor
- $i_{max} \equiv$ corriente máxima del motor
- $k_{mv} \equiv$ constante de torque-voltaje del motor
- $k_{mi} \equiv$ constante de torque-corriente del motor
- $R \equiv$ resistencia del motor
- $k_d \equiv$ constante arbitraria para regular la respuesta del controlador

Tomando dicha entrada se programó el bloque de "*Swing_up*". Para la simulación del controlador se hizo uso del diagrama mostrado en la figura 5, donde solo se sustituyó el bloque correspondiente al control "*Stabillization_Controller*" por el de "*Swing_up*". Se realizaron dos pruebas, las cuales se muestran en la figura 8.



a) Posición con $q_{p0} = \pi - \pi / 64$

b) Señal de control $\tau_{max} = 12mN \cdot m$

c) Posición con $q_{p0} = \pi / 2$

c) Señal de control. $\tau_{max} = 6mN \cdot m$

Figura 8. Respuesta del controlador de Swing up.

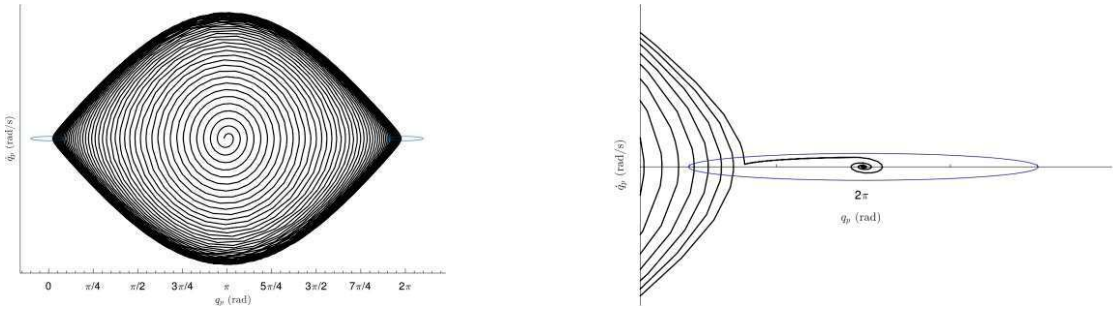
Ingeniería y Automatización, Capítulo 16, pp. 205 - 223.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2.6 Control Híbrido

La finalidad del control híbrido es acoplar los dos controladores diseñados previamente de la manera más suave, es decir, buscando una transición no abrupta. Los enfoques a abordar para este diseño serán los propuestos en [7] y [8]. Para el desarrollo del presente trabajo, se tomará el criterio descrito por la ecuación (15):

$$\tau(t)\begin{cases} -k_d\left(E-E_o\right)sat\left(\dot{q}_p\right) & si \quad \left(q_p-\hat{x}_1\right)^2+\dot{q}_p^2 \geq \delta \\ -\mathbf{K}z & si \quad \left(q_p-\hat{x}_1\right)^2+\dot{q}_p^2 < \delta \end{cases} \tag{15}$$

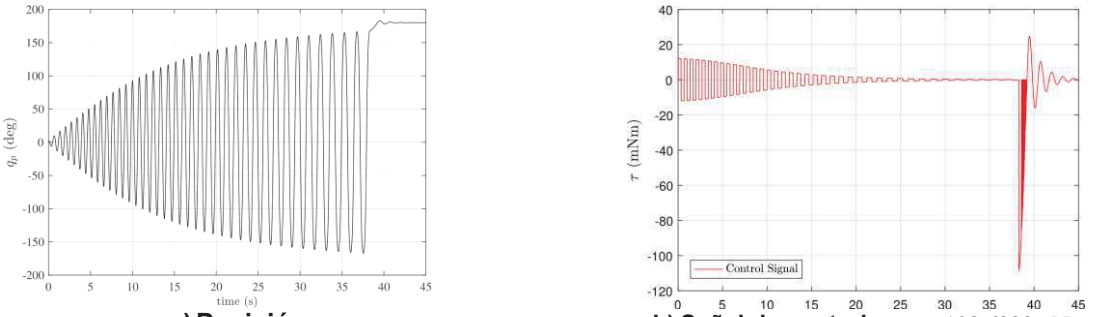
Representando en forma gráfica la ecuación (15) se obtuvo la figura 9.



a) Espacio de fase del balanceo. **b) Región de convergencia.**

Figura 9. Representación del criterio en el espacio de fase.

Para la simulación del controlador híbrido se realizó al igual que los anteriores, con el diagrama de la figura 5, solo se sustituye el de "*Stabillization_Controller"* por el de "*Hybrid_Controller".* Se realizaron varias pruebas para diferentes valores de δ y $\mathbf{K}$, el resultado más satisfactorio fue el que se hizo con valores de $\delta = 0.5$ y $\mathbf{K} = \begin{bmatrix} -0.5084 & -0.0062 & -0.0000534 \end{bmatrix}$, el cual se muestra en la figura 10.



a) Posición. **b) Señal de control.** $\tau_{max} = 108.6332\text{mN}\cdot\text{m}$

Figura 10. Respuesta del control híbrido.

2.7 Selección del motor

A partir de las simulaciones que se realizaron en la sección anterior, se determinó que el torque máximo constante requerido por el controlador es de $14.5145mN\times m$ presentado durante la

estabilización, también se determinó que el torque pico máximo al momento en el que el péndulo se encuentra en el punto crítico superior o en su parte de equilibrio inestable ($q_p \approx 0$), se necesita un torque de $108.6332 mN \cdot m$.

Al realizar la búsqueda para obtener el actuador, se encontró al fabricante *Pittman*®, el cual proporciona todos los datos técnicos del actuador. Se identificó en la serie Pittman 9000, diversos modelos que satisfacen los requerimientos anteriormente expuestos, por lo que se toma la decisión de elegir el modelo de motor 9234S007-R1-SP con un costo de $ 219.32 dlls. La tabla 1 enlista algunos de los parámetros eléctricos y mecánicos más importantes del motor. Cabe mencionar que el motor posee su propio encoder, el cual proporciona 500 pulsos por revolución para la estimación de la velocidad y aceleración de la rueda.

Tabla 1. Parámetros importantes del motor 9234S007-R1-SP.

Parámetro	Unidad	Magnitud
Voltaje	V	24
Torque	$mN \cdot m$	43.07
Velocidad	rpm	5040
Corriente	A	1.5
Torque pico	$mN \cdot m$	289.52
Corriente pico	A	8.1
Masa	kg	0.340

2.8 Selección del driver para el motor

De acuerdo a la Tabla 1, el motor consume una potencia continua de 23 W por lo que se elige un driver que consiste en un arreglo de semiconductores tipo MOSFET que entrega 12 A de corriente continua y 43 A de pico de corriente, con una alimentación que puede variar de 5.5 V a 27.5 V. Además, soporta señales de PWM de hasta 20 kHz. Tomando en cuenta las características anteriormente mencionadas el driver deberá cumplir con la potencia que consume el motor para su correcto funcionamiento. El driver seleccionado es el BTS7960.

2.9 Selección de los sensores

Para la medición de la posición del péndulo (posición angular) y posición del disco, las características deseables para la elección de un sensor son las siguientes:

- Alta resolución: más de una muestra por grado recorrido.
- Alta repetitividad: que el sensor no tienda a variar su medida tras varias sesiones o ciclos de trabajo.
- Rapidez: debe de permitir muestrear la posición angular del sistema a altas frecuencias.
- Rango de medida entre 0 y 360°.

Por lo tanto, los encoder son los sensores que mejor se adaptan a las características expuestas anteriormente. Una pequeña variación en la medición puede representar un error alto en la dinámica del controlador.

Anteriormente se mencionó que el motor posee su propio encoder el cual servirá para medir posición, velocidad y aceleración de la rueda. Para la posición y velocidad del péndulo se seleccionó un encoder digital de tipo absoluto, con 10 bits de resolución, el modelo del encoder es EMS22A.

2.10 Selección de la tarjeta de control

La técnica que se selecciona para la adquisición y procesamiento de datos, es implementar el control con un sistema de adquisición de datos, el cual involucra una tarjeta, sensores, actuadores y una computadora para el procesamiento de la información.

La tarjeta seleccionada es una tarjeta NI myRIO 1900 de la empresa National Instruments®, se decide dicha tarjeta dada su compatibilidad de conexión con el software LabVIEW®, además de cumplir con el número mínimo de señales y los protocolos de comunicación necesarios para el acondicionamiento de todas las señales involucradas.

2.11 Diagramas electrónicos

En la figura 11 y figura 12 se presenta la distribución de entradas y salidas de la tarjeta de control del sistema de adquisición y procesamiento de datos seleccionado.

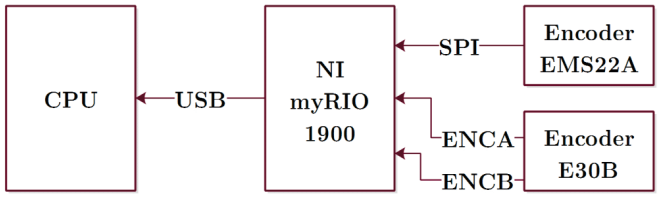


Figura 11. Diagrama de bloques de entradas del sistema.

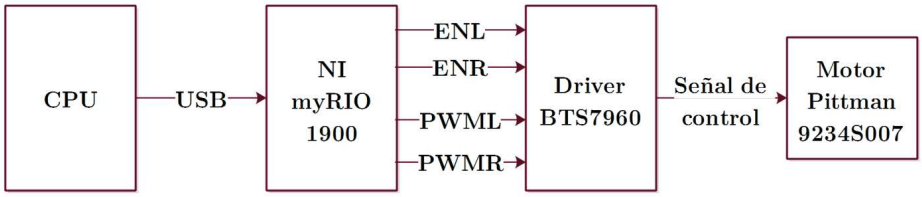


Figura 12. Diagrama de bloques de salidas del sistema.

En la figura 13 se muestra como serán las conexiones eléctricas del sistema, nótese que esta etapa corresponde al acondicionamiento de señales entre el sistema de adquisición, sensores y actuador.

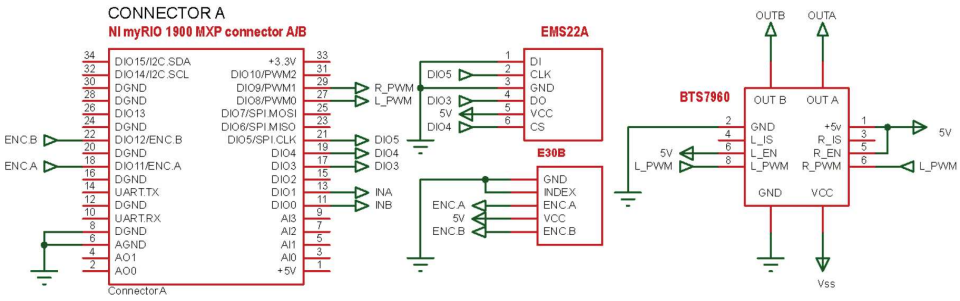


Figura 13. Conexiones eléctricas.

2.12 Cálculo del eje

Utilizando un análisis estático, en la figura 14 se muestra el diagrama de cuerpo libre del eje, se supone un eje simple de longitud L y diámetro d el cual está soportado por rodamientos en sus extremos (puntos A y B), a una distancia l_o del soporte más a la izquierda (A), el eje se une con un péndulo de masa m. La masa del péndulo se interpreta como una carga de magnitud O en dirección negativa respecto al eje y, a su vez las reacciones en los rodamientos se toman en dirección y positiva con magnitud en los dos planos (YZ y XZ) respectivamente.

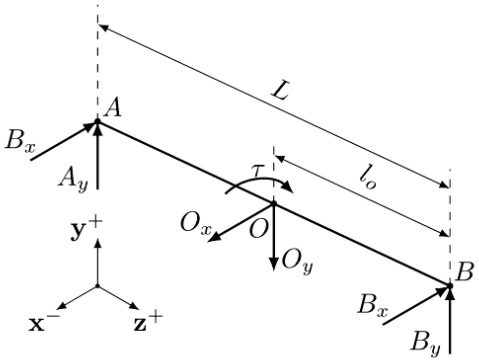
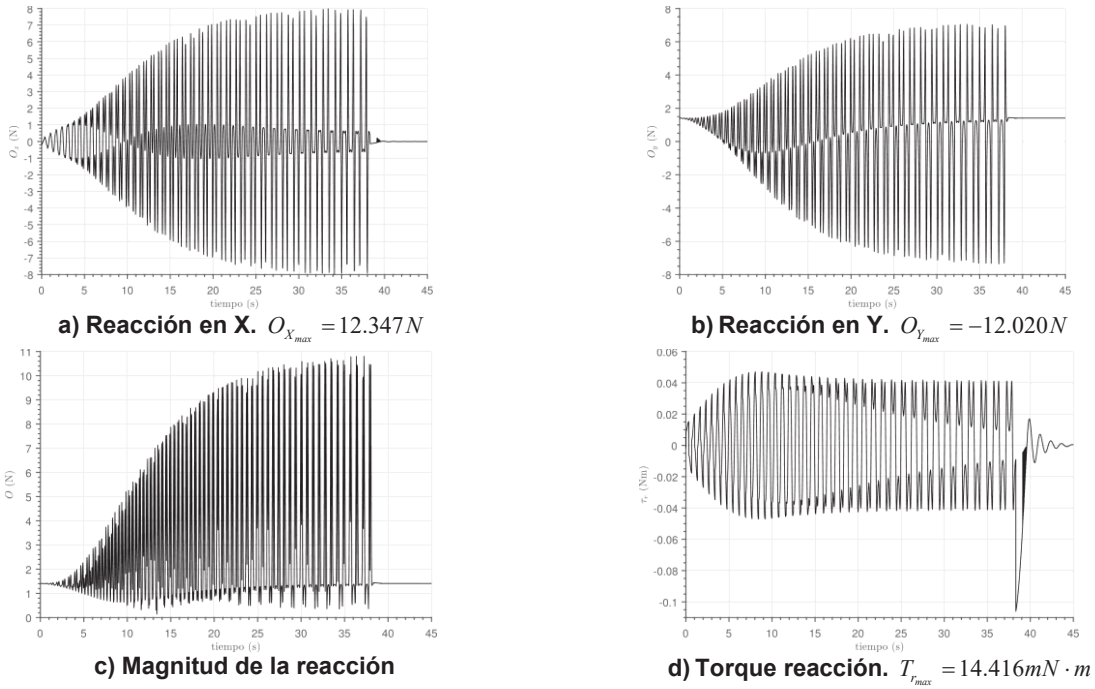


Figura 14. Diagrama de cuerpo libre del eje.

Por sumatoria de fuerzas y sumatoria de momentos en el plano YZ y XZ se obtienen las expresiones que representan las fuerzas presentes en los puntos A y B, tomando las distancias como $L = 5in$ y $l_o = L/2$, se obtuvieron las expresiones dependientes de las fuerzas presentes en el punto O, donde sus componentes se expresan en la ecuaciones (3) y (4), para obtener sus valores máximos se realizaron algunas simulaciones y se obtuvieron los valores mostrados en la figura 15:



a) Reacción en X. $O_{X_{max}} = 12.347N$

b) Reacción en Y. $O_{Y_{max}} = -12.020N$

c) Magnitud de la reacción

d) Torque reacción. $T_{r_{max}} = 14.416mN \cdot m$

Figura 15. Simulación de las cargas dinámicas

Después se realizaron los diagramas de cuerpo libre, cortante y momento para cada plano (YZ y XZ) respectivamente, para así obtener los momentos máximos en cada plano y así calcular el momento máximo aplicado al eje, el cual nos dio un valor de $M_{max} = 0.547N \cdot m$.

El material que se eligió para el eje es un Acero Inoxidable Recocido, de la Tabla A-22 del libro de R. G. Budynas [11] se obtuvo la propiedad mecánica $S_y = 241MPa$.

En [11] se presenta el esfuerzo de Von Misses haciendo uso de las componentes xyz del esfuerzo tridimensional, en la figura 14 se observó que al eje se le aplica un esfuerzo plano, por lo que se utilizará la ecuación (16):

$$\sigma' = \sqrt{\sigma_x^2 + \sigma_y^2 - \sigma_x \sigma_y + 3\tau_{zx}^2} \tag{16}$$

Identificando los esfuerzos se tienen:

$$\sigma' = \frac{S_y}{n}, \quad \sigma_x = \frac{M_{xz_{\max}} c}{I}, \quad \sigma_y = \frac{M_{yz_{\max}} c}{I}, \quad \tau_{zx} = \frac{T_r c}{J} \tag{17}$$

Donde:

$$I = \frac{\pi}{4} c^4, \quad J = \frac{\pi}{2} c^4 \tag{18}$$

Sustituyendo las expresiones de (18) en las expresiones de (17), se obtiene una expresión de la cual se despeja c (radio del eje) y así obtener el diámetro mínimo del eje.

$$c = \sqrt[3]{\frac{2n}{S_y \pi}\sqrt{2\left(M_{xz_{\max}}^2 + M_{yz_{\max}}^2 - M_{xz_{\max}} M_{yz_{\max}}\right) + 3T_r^2}} \tag{19}$$

Utilizando un factor de seguridad mínimo $n = 1$ se obtuvo un diámetro mínimo de $d = 2.717mm$.

Por facilidad de adquisición de elementos, se eligió un eje lineal de 8 mm de Acero Inoxidable Recocido al ser una medida comercial. Se recalculó el factor de seguridad para un eje lineal de 8 mm, por lo que de la ecuación (19) se despejó el factor de seguridad. y se sustituyeron valores numéricos, el resultado arrojó un factor de seguridad de $n = 25.51$.

2.13 Cálculo del rodamiento

Del Catálogo General de Rodamientos de SKF® se seleccionó un rodamiento rígido de una hilera de bolas. El modelo que se eligió fue un SKF 608, el cual tiene un diámetro interno $d = 8mm$, y con propiedades de capacidad de carga básica estática $C = 3.45kN$, capacidad de carga dinámica $C_o = 1.37kN$. En el diseño del eje se observa que solo existen cargas radiales, por lo que el factor de seguridad se define por:

$$S_o = \frac{C_o}{P_o} \tag{20}$$

Donde:

$$P_o = 0.6F_r + 0.5F_a \tag{21}$$

Al existir solo cargas radiales y si $P_o < F_r \rightarrow P_o = F_r$ esto lleva a:

$$S_o = \frac{C_o}{P_o} = \frac{C_o}{F_r} = \frac{3.45kN}{6.010N} = 574.04 \tag{22}$$

Como se obtuvo un valor mucho mayor a uno para el factor de seguridad se concluye que el rodamiento funcionará correctamente en forma estática y dinámica.

3. Resultados

3.1 Descripción de la estructura mecánica

La estructura mecánica que se utiliza en este trabajo es similar a la utilizada por S. H. Park en [7]. Esta consiste en una base fija conectada con dos soportes laterales paralelos por medio de elementos de uniones no permanentes (tornillos) de forma perpendicular. A través de los soportes laterales cruza un eje que sirve para la rotación del péndulo (q_p), el cual se une al péndulo mediante un cople. En el otro extremo del péndulo se fija el motor seleccionado, en el eje del motor se une el disco de inercia, conformando así el grado de libertad actuado q_d . La estructura propuesta se muestra en la figura 16 y sus componentes se enlistan en la Tabla 2.

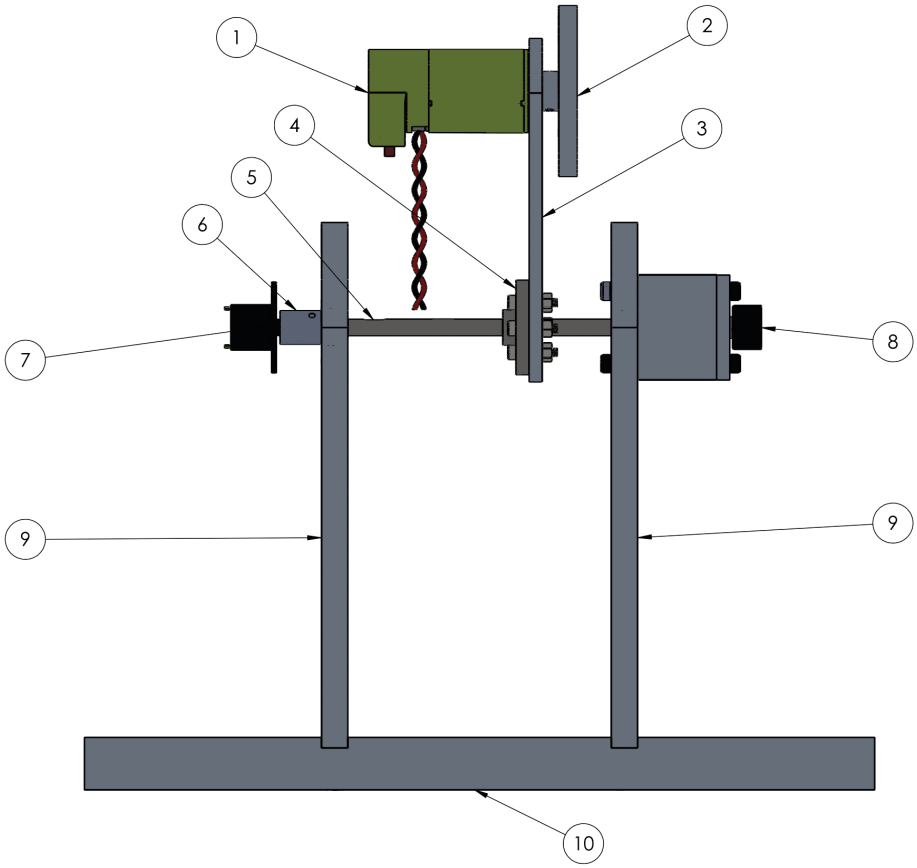


Figura 16. Diagrama de elementos que componen el Péndulo Invertido Inercial.

Tabla 1. Parámetros importantes del motor 9234S007-R1-SP.

Número	Componente
1	Motor de CD/Encoder de cuadratura
2	Disco de Inercia
3	Péndulo
4	Cople Péndulo – Eje
5	Eje
6	Cople Eje-Slip ring
7	Slip ring
8	Encoder
9	Soportes laterales
10	Base Fija

3.1.1 Soporte lateral

Para los soportes laterales, se propone la geometría mostrada en la figura 17, en la cual se muestran dos detalles. El detalle A corresponde a barrenos roscados, los cuales servirán para unir los soportes laterales con la base. Finalmente, el detalle B corresponde a una caja circular en la cual se colocarán los rodamientos y serán fijados por un ajuste a presión.

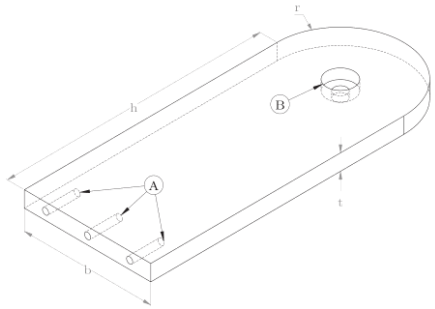


Figura 17. Soporte Lateral.

3.1.2 Base fija

En la figura 18, se muestra la geometría propuesta para la base, en la cual se muestran dos detalles. El detalle A corresponde a una caja cuadrada en la cual se asentarán los soportes laterales. Finalmente, el detalle B corresponde a barrenos roscados para la tornillería que une la base y los soportes laterales.

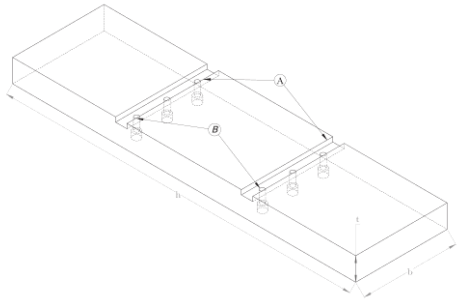


Figura 18. Base fija.

3.1.3 Péndulo

El péndulo propuesto tiene la geometría mostrada en la figura 19, en la que se muestran tres detalles: el detalle A corresponde a un barreno pasado a través del cual pasa el eje del motor, el detalle B corresponde a un barreno pasado a través del cual pasa el eje de rotación del péndulo. El detalle C corresponde a ocho barrenos pasados y roscados, donde 4 barrenos servirán para acoplar el péndulo a su eje de rotación mediante la pieza bosquejada en la figura 24, y 4 barrenos para fijar el motor al péndulo.

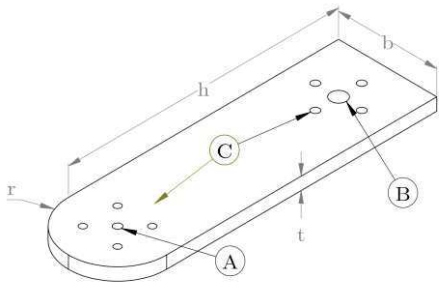


Figura 19. Péndulo.

3.1.4 Disco de inercia

En la figura 20 se muestra la geometría adoptada para el disco de inercia. El detalle A, corresponde a un barreno a través del cual pase el eje del motor y el detalle B corresponde a barrenos roscados para opresores los cuales tendrán la función de fijar el eje del motor con el disco de inercia.

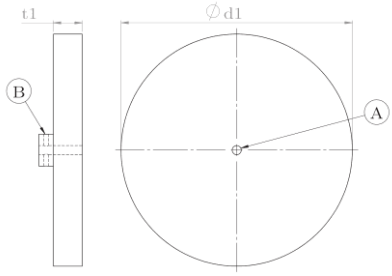


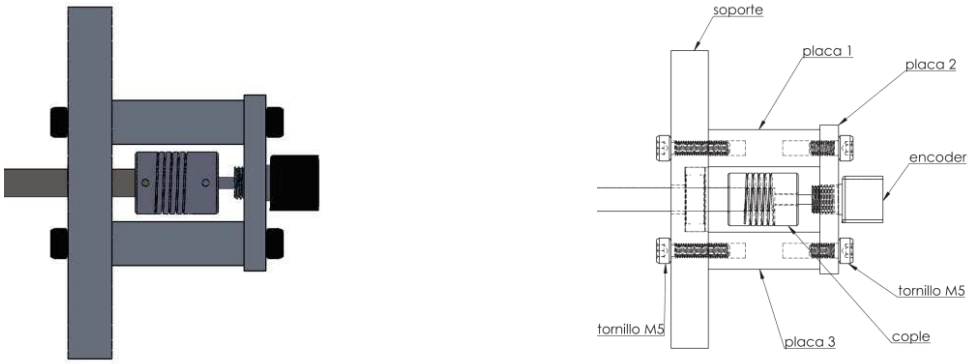
Figura 20. Disco de inercia.

3.2 Detalles de ensamblaje

La sujeción de la base y los soportes laterales se hace por medio de tornillos M5x0.8 de 20 mm de largo, en la base se añade una caja para la cabeza de los tornillos además de machuelear los soportes, en la figura 21 se presenta la unión utilizada.

Figura 21. Vista detallada de la base.

El encoder para medir la posición q_p se fija a través de un soporte formado por tres placas de aluminio, la fijación se hace mediante barrenos en las placas y machueleados de las mismas. La sujeción de las placas al soporte lateral se hace por medio de tornillos M5X0.8 de 12 mm de largo. Se hace uso de un cople flexible utilizado para unir el encoder y el eje de rotación del péndulo, por medio de opresores M4x0.8. En la figura 22 se muestra el esquemático de las uniones.



a) Vista sombreada con aristas **b) Vista líneas ocultas visibles**

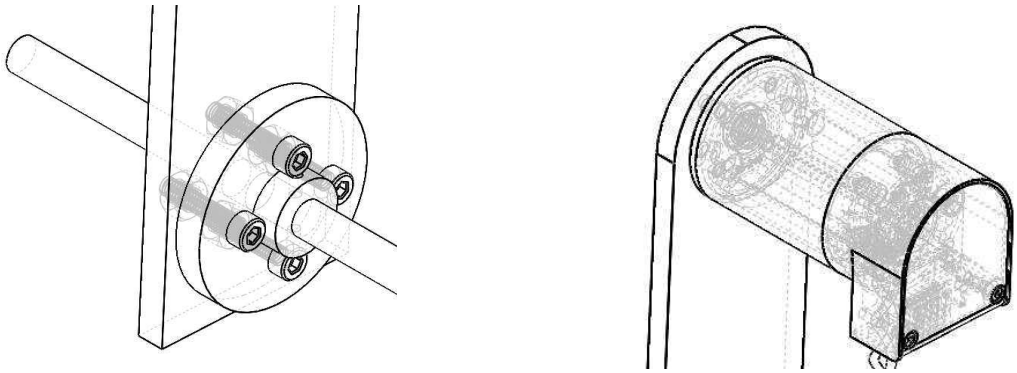
Figura 22. Acoplamiento del encoder a la estructura.

Se utiliza un rotor (slip-ring) para las conexiones eléctricas, este es unido al eje de rotación mediante un cople de aluminio y opresores M4x0.8, en la figura 23 se muestra un detalle del montaje del componente y la estrategia tomada para pasar los cables a través del rodamiento.

a) Vista sombreada con aristas **b) Vista líneas ocultas visibles**

Figura 23. Acoplamiento del slip ring a la estructura.

Al diseño del péndulo se le agregan los barrenos necesarios para fijar el cople y así unirlo al eje de rotación, además de los barrenos para la fijación del motor con el péndulo, figura 24.



a) Unión péndulo-cople-eje. **b) Unión péndulo-motor.**

Figura 24. Acoplamiento el péndulo con eje y con motor.

El eje de rotación mencionado en la sección 2.12 tiene un diámetro de 8 mm y una longitud de 200 mm. Al eje se le añade un barreno de 4mm con una profundidad de 38.1 mm sobre la cara de sección transversal y una caja perpendicular del mismo diámetro y de longitud de 25mm, con la finalidad pasar los cables de sensores y motor hacia el slip ring (figura 23 y figura 25), y así evitar que se enrollen en el eje de rotación.

Figura 25. Eje con barrenos.

Con lo anteriormente mencionado en los detalles de ensamble, en la figura 26 se muestra una vista explosionado del dibujo detallado en 3D, así como de sus líneas de ensamble.

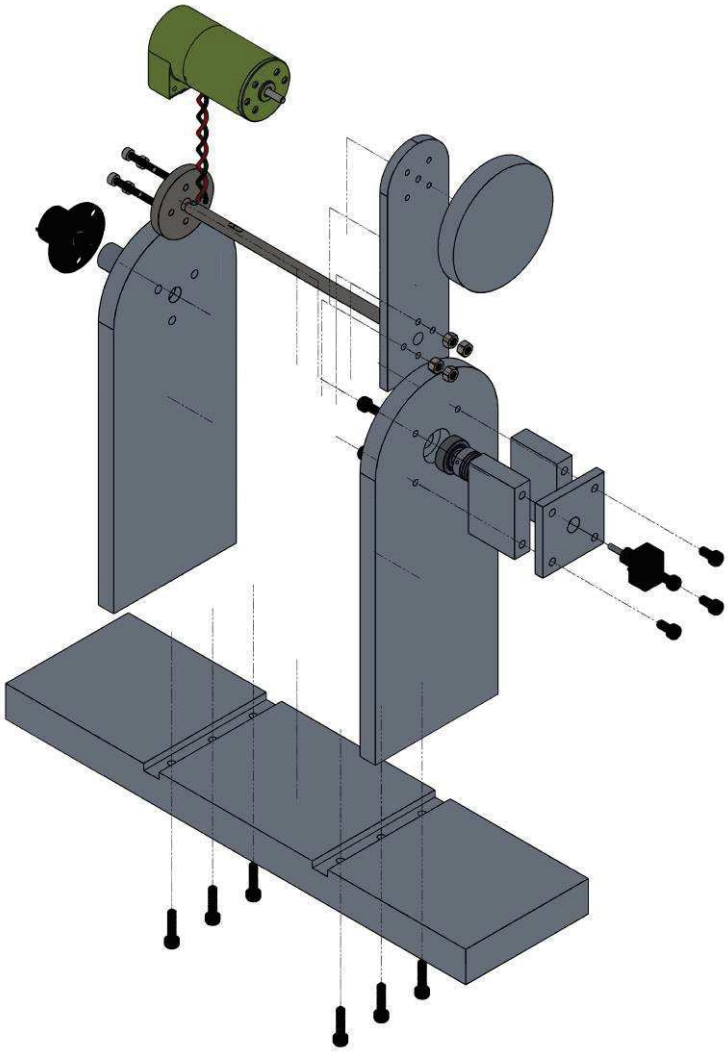


Figura 26. Vista explosionada del Péndulo Invertido Inercial

4. Conclusiones

La metodología propuesta buscó responder la pregunta: ¿Qué actuador se necesita para controlar esta configuración del sistema?, a través de un enfoque mecatrónico. La razón para tomar este enfoque es que, en el Estado del Arte del Péndulo Invertido Inercial, una amplia mayoría opta por analizar única y exclusivamente estrategias de control, mientras que son contadas las publicaciones que abordan algo referente al diseño mecánico y/o electrónico. Si bien, no se puede dejar de lado el control, en el desarrollo del trabajo sólo se hace referencia a estrategias ya probadas y se hace hincapié en el diseño total del dispositivo (mecánica y eléctricamente).

El siguiente paso es el diseño de un plan de manufactura para la construcción del prototipo.

El campo de estudio de este tipo de plataformas es muy amplio, ya que día con día se desarrollan diferentes técnicas de control aplicables a los péndulos invertidos. Un proyecto a futuro sería deducir y estudiar un modelo dinámico del sistema Péndulo Invertido Inercial, donde se consideren las fuerzas de fricción. Esto con la intención de obtener una predicción del comportamiento del sistema más cercana a la real, éstas perturbaciones se obtendrían mediante experimentos como lo proponen M. W. Spong en [2] y C. Aguilar en [12].

Otro proyecto sería estudiar otras problemáticas de control como seguimiento de trayectorias, generación de oscilaciones, control de posición y de velocidad entre otras.

Referencias

[1] Åström K., Furuta K., “Swinging up a pendulum by energy control”, Automatica, Volumen 36, 287-295, 2000.
[2] Spong M., Coke P., Lozano R., “Nonlinear control of the reaction wheel pendulum”, Automatica, Volumen 37, 1845-1851, 2001.
[3] Åström K., Block D., Spong M., “The Reaction Wheel Pendulum”, Morgan & Claypool Publishers, EUA, 1° edición, 2007.
[4] Merlo Zapata C., “Diseño, construcción, modelado y control en tiempo real del péndulo con rueda inercial”, Tésis de maestría, CDMX, 2015.
[5] Alonso D., Robbio F., Paolini E., Moiola J., “Modelling an inertia wheel pendulum benchmark”, Mathematical and Computer Modelling of Dynamical Systems, Volumen 11, 255-272, 2007.
[6] White W., “Dynamic sistems & controls”, Researchgate, 2010.
[7] Park S., Lee H., Lim S., Kim J., “Robust position control of a reaction wheel inverted pendulum”, Journal of Korean Institute of Intelligent Systems, Volumen 26, 127-134, 2016
[8] Silva Ortigoza R., Carrillo Serrano R., Hernandez V., “Control Automático: Teoría de Diseño, Construcción de Prototipos, Modelado, Identficación y Pruebas Experimentales”, CDMX, 2013.
[9] Fantoni I., Lozano R., “Non-linear Control for Underactuated Mechanical Systems”, Springer, Londres, 1° edición, 2002.
[10] Ogata K., “Ingeniería de Control”, Prentice Hall, España, 5° edicion, 2010.
[11] Budynas R., Nisbett J., “Shigley's Mechanical Engineering Design”, McGraw Hill, EUA, 9° edición, 2010.
[12] Moreno Valenzuela J., Aguilar Avelar C., “Motion control of underactuated mechanichal systems”, Springer, Volumen 88, 2017.

Ingeniería y Automatización, Capítulo 17, pp. 224 - 234.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Instrumentación y Control para Tiempos de Inmersión y Flujo Fotónico en Acondicionamiento in vitro de Plantas Mediante Biorreactores de Inmersión Temporal

Durán Lugo Juan M., Quej Cosgaya Hector M. y Valladares Castellanos Manuel A.

Universidad Autónoma de Campeche – Facultad de Ingeniería
Av. Agustín Melgar s/n entre Juan de la Barrera y Calle 20, Col. Buenavista
San Fco. de Campeche, Campeche. Tel. (981) 8119800 Ext. 3030104
jumduran@uacam.mx

Resumen

La micropropagación de plantas se fundamenta en el desarrollo in vitro de tejidos vegetales por biorreactores, regulando tiempos de inmersión en medios de cultivo, longitudes de onda e intensidad del Flujo Fotónico Fotosintético (PPF). El proyecto se centra en regular y caracterizar el PPF mediante un controlador difuso retroalimentado integrado a un sistema embebido sustentable y de ahorro energético automatizado para la reproducción masiva de plantas en un ambiente controlado, integrando diferentes etapas a una Interfaz Gráfica de Usuario (GUI) y un Sistema Embebido, para el control de longitudes de onda (tonalidad) y de tiempos de inmersión del proceso en tiempo real. Usando regresión por mínimos cuadrados, se linealiza el sensor desarrollado con respecto a un luxómetro digital de alta precisión, seguidamente se genera un algoritmo de conversión (luxes – PPF). Se obtiene un factor de medición del sensor de -+.5% de error máximo, con respecto al controlador difuso un +- 1% del valor deseado con respecto al valor medido en índice de PPF incidente en el desarrollo y reproducción de plantas.

Palabras clave: Instrumentación virtual, control difuso, flujo fotónico, micropropagación, sistema embebido, biorreactores, in vitro.

1. Introducción

La creciente población que azota al mercado en el consumo de alimentos puede causar un desabasto en pocos años, por ello día a día aparecen nuevas alternativas para satisfacer dicha demanda, siendo una de ellas la reproducción masiva de plantas por micropropagación. La técnica anteriormente mencionada, se lleva a cabo de manera in vitro (dentro de recipientes de vidrio) en un Sistema de Inmersión Temporal (SIT), con un ambiente esterilizado y hermético, sin embargo dichos sistemas implementan tecnologías laboriosas y costosas [10]

Los SIT en particular, están conformados por Biorreactores de Inmersión Temporal (BIT) en donde se desarrollan los procesos biológicos importantes de las plantas, proporcionando un ambiente controlado y permitiendo el crecimiento eficaz hasta su cultivo [17]. El sistema está constituido por dos vasos transparentes de vidrio (llamados también vasos gemelos), esterilizados y sellados herméticamente previamente, se interconectan por medio de tubos de silicona para la inyección de presión (Fig. 1A) con un compresor de aire, el medio líquido contenido en uno de los recipientes de vidrio se desplaza hacia el que contiene las plantas (Fig. 1B), con la finalidad de brindarle los nutrientes necesarios para el desarrollo de esta, es importante señalar que se requiere definir los tiempos de inmersión de las plantas según su naturaleza (Fig. 1C), para posteriormente realizar el proceso inverso (Fig. 1D) y retornando el medio de cultivo al reciente de vidrio original [2].

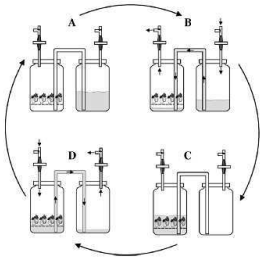


Figura 1. Proceso de inmersión temporal [2].

La luz en las plantas juega un papel esencial, ya que con ello realizan el proceso de fotosíntesis, absorbiendo energía en un rango espectral visible de la luz. La Radiación Fotosintéticamente Activa (PPFD, PAR o RFA) comprende las longitudes de onda de la luminosidad de entre 400nm hasta 700 nm, siendo ese intervalo el que aprovechan las plantas en la fotosíntesis [15]. También es importante enfatizar que los generada por Diodos Emisores de luz por sus siglas en inglés LED, se han implementado en diversas áreas de investigación, como síntesis de clorofila, fotosíntesis, foto-biorreactor de algas, foto morfogénesis, etc. [14].

Las plantas en la oscuridad carecen de fotosíntesis, sin embargo no quiere decir que no respiren; por tal motivo requieren un valor mínimo de irradiación para lo cual la tasa de consumo de CO2 y la de desprendimiento por respiración sean iguales Es en este punto la tasa de asimilación de CO2 fotosintético es 0 y se le llama punto de compensación de luz, a partir de este valor mínimo de irradiación, cuanto más aumente la irradiancia más aumenta la fotosíntesis hasta alcanzar un punto de saturación consecuencia de la limitación impuesta por la velocidad de las reacciones del carbono [2]. Para lograr una correcta valoración de la luz aprovechable por diferentes plantas, se requiere controlar y asegurar el cumplimiento de la cantidad de fotones incidentes.

La luz al ser concebida como una corriente de partículas de luz, llamados fotones, se puede calcular la intensidad de una fuente de luz artificial de acuerdo número de fotones que golpean una hoja por segundo. La unidad para este cálculo es la Densidad de Flujo Fotónico Fotosintético (PPFD) dada en µmol.m-2s-1.

Actualmente para fines de micropropagación de plantas, se requiere regular la intensidad de PPFD y cerciorarse de que el valor deseado se cumpla, por lo que se plantea la implementación de un Sistema de Control Difuso de Lazo Cerrado mediante un Controlador Proporcional Integral Derivativo.

2. Metodología

2.1 Sistema lumínico

Siendo uno de los objetivos el uso eficiente de energía, además de particularizar está en fines de control, la tecnología LED resulta ser la adecuada para la implementación en el SIT por los beneficios siguientes:

1. La velocidad de respuesta (0.2 µs), lo cual permite cambiar su intensidad por medio de la modulación por ancho de pulso (PWM) [14], la cual se controla por medio de una tarjeta de desarrollo Arduino ATMEGA 2560 (figura 2) el cual cuenta con las suficientes salidas de PWM para este fin [3].
2. No irradian calor, tienen alta eficacia y duran hasta 50.000 horas de vida, por lo tanto, ahorra energía eléctrica [11]; además que hoy en día, esta tecnología está sustituyendo a las existentes como luces de halógeno, lámparas incandescentes, etc., [5].
3. Evaluar los efectos fotomorfogénicos de la calidad de luz con distintas cantidades de moles (PAR) en las plantas [4], especialmente con el nivel cinco, el cual se puede variar a voluntad.

De acuerdo a las dimensiones por nivel planteadas en el diseño de la consola, y con la finalidad de aprovechar al máximo los espacios, fueron situados en la parte superior de cada nivel 10 mts de tiras LED, tomando en cuenta el grosor de cada tira de 0.9 cm para cubrir el total de 10 mts se tuvo una separación entre cada tira de 4 cm (con un total de nueve separaciones) para cubrir el total del ancho de cada nivel y, de esa manera, lograr una iluminación uniforme.

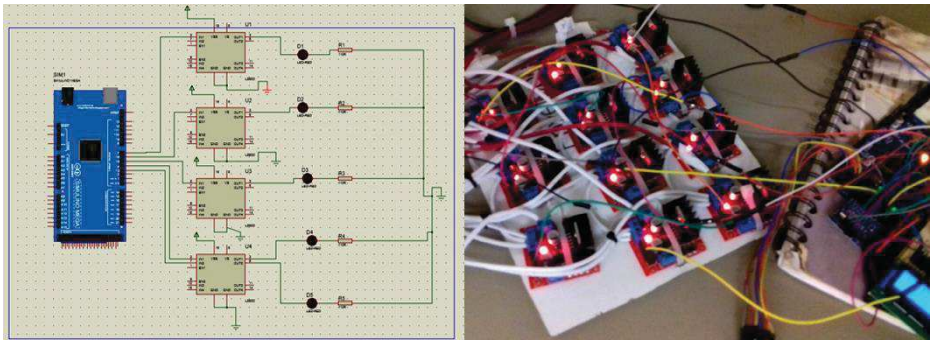


Figura 2. Etapa de potencia y Atmega 2560.

2.2 Sistema neumático

Para la adecuación del sistema neumático se definieron los siguientes elementos: 1 Compresor, 1 Unidad de Mantenimiento, 10 Reguladores de Presión Neumática (2 por nivel) y 2 Electroválvulas neumáticas. A continuación se mencionan las características principales de cada elemento:

El compresor conceptualizado es alimentado por 220 VRMS (Voltaje Eficaz) de corriente alterna, con un depósito de 24 L a un caudal de 50 L/min; teniendo como presión máxima de carga a 8 bar (800 kPa), por lo que este elemento suministra la energía neumática para desplazar los medios de cultivo entre vasos gemelos. Las Unidades de Mantenimiento son las encargadas de eliminar las impurezas causadas por aceites, agua, polvo, etc. generadas en el compresor y por completo dañinas en el SIT.

Los reguladores de presión disminuyen la presión que llega a cada nivel donde se sitúan los biorreactores, se utilizaron diez reguladores en total, cinco para las inmersiones de todos los niveles y cinco para las emersiones y/o aireaciones; de la misma manera se ajustó por nivel una presión de 1 bar, así pues, se considera que por cada 6.89 bar existe una pérdida de 0.68 bar (Almachi y Johanna, 2015) por lo cual la presión que se reguló viene dada por una regla de tres simple: (6.85 bar)/(1 bar) = (0.68 bar)/(X bar) de la cual resulta para X=0.1 bar, con lo cual la presión que se ajustó fue de 1.1 bar.

Las electroválvulas neumáticas son activadas con una señal eléctrica de 24 VCD a 40 mA de corriente, cinco vías, dos posiciones y retorno por muelle; las cuales para su conmutación se implementó un sistema de control por medio de un semiconductor de potencia (L298N), los cuales pueden operar hasta 50 V de corriente directa disipando un máximo de 25W (ST, 2000). Fueron requeridas dos electroválvulas para el correcto funcionamiento, una para las inmersiones y la otra para las aireaciones; cabe mencionar que los ciclos de inmersión se sincronizan con un reloj conectado a la placa de desarrollo cuyo modelo es el DS1307 con comunicación I2C que opera a 5V [12].

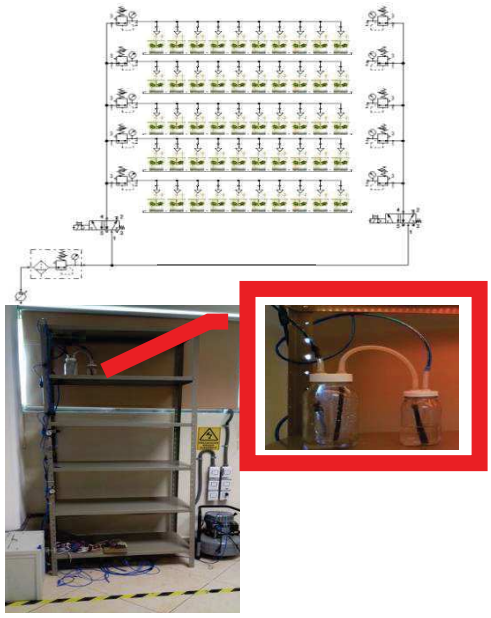


Figura 3. Esquema neumático de los biorreactores.

La etapa anterior fue acoplada por tuberías de 4mm, y conectores tipo “T”, finalmente por biorreactor es necesario instalar un filtro de 0.2 µm, para con ello evitar la entrada de contaminantes biológicos, como se puede observar en la Figura 3.

2.3 Sistema eléctrico

El sistema encargado de alimentar a los dispositivos del SIT se definieron según la caracterización del consumo de energía detallado en la Tabla 1, componentes como el ATMEGA 2560, DS1307 o la pantalla LCD se describen más adelante, sin embargo se desprecian al tener un consumo significativamente bajo de corriente eléctrica.

Tabla 1. Distribución de energía eléctrica.

Elemento	Cantidad	Voltaje requerido por elemento (V)	Potencia necesaria por elemento (W)	Potencias subtotales (W)
Semiconductor de potencia L298N	15	12-30	0.48	7.2
Tira de 300 led monocolor de 5 mts	8	12	24	192
Tira de 300 led multicolor de 5 mts	2	12	36	72
Electroválvulas	2	24	0.96	1.92
			Potencia total consumida	273.12 Watts

Para la prevención sobrecargar en un rango superior al 80%, para el suministro alimentación se utilizaron dos fuentes de energía con 12V a 15A marca Commax modelo FC1215, esto con la finalidad de poder activar las electroválvulas sin requerir otro circuito, alcanzando un total de 360W.

2.4 Sistema eléctrico

Para el diseño ideal de la consola en primera instancia se definieron las dimensiones específicas de los vasos gemelos para el sistema de biorreactores de inmersión temporal, los cuales

cuentan con las siguientes medidas: altura de 7cm, diámetro de base de 9cm y diámetro de boquilla de 6.5cm.

Mediante el programa SolidWorks M.R. se dibujó y renderizó la base con los cinco niveles requeridos para realizar la experimentación con cinco variedades de longitudes de onda, con las especificaciones siguientes: altura de 220cm, largo de 100 cm, ancho de 45cm, distancia por nivel de 40 cm, atendiendo la Norma Mexicana NOM-093-SSA-1994 (Higiene y Sanidad Industrial), definiendo como altura mínima de 15 cm del suelo al primer nivel del estante, por lo que se concluyó en una altura de 20 cm. Al final se obtuvo el SIT mostrado en la Figura 4.

Figura 4. Sistema de Inmersión Temporal.

2.5 Tarjeta de Adquisición de Datos (DAQ) e Instrumento Virtual (VI)

Debido a los requerimientos previamente estudiados del prototipo, se implementó un algoritmo de control capaz de cambiar la intensidad de luz en cada nivel, determinar escalas del 0 al 100% la intensidad de cada color de los LED´s RGB del ultimo nivel (generando de esta manera diferentes longitudes de onda), modificar los datos de las inmersiones, así como consultar y modificar la hora.

Como interfaz externa de usuario se acopló un teclado matricial 4x4, conectado a los pines del 46 al 53 para la comunicación entre el usuario y la tarjeta de desarrollo, por lo cual, en coordinación del programa, se cumplen con las rutas de control que se muestran a continuación en donde se puede ver las opciones del programa en su totalidad, así como los límites de operación en cuanto a las opciones que podemos ingresar en cada una de las rutas.

Para el control del SIT mediante un sistema embebido, se designaron los botones alfanuméricos de las siguientes formas:

- Teclas numéricas: sus valores correspondientes en valor numérico.
- Tecla C: Opción de regresar.
- Tecla D: Opción de Acceder o aceptar.
- Tecla #: Opción de Borrar.

Finalmente para la visualización del usuario se implementó una pantalla LCD, donde el usuario puede visualizar las operaciones que se llevan a cabo, comunicándose por medio del teclado matricial

hacia el sistema, a través de la tarjeta de desarrollo; la conexión que se utilizó para ello fueron los pines 38 al 45 (Fig. 5).

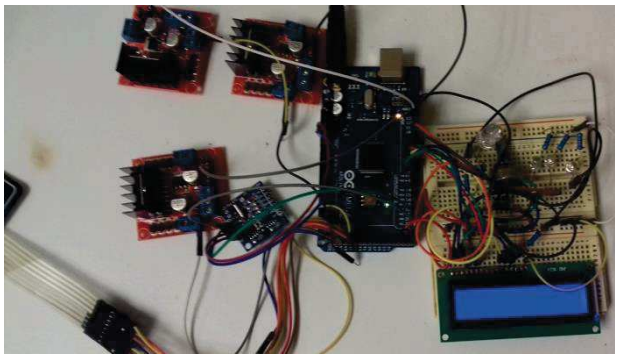


Figura 5. Conexión del sistema embebido con etapa de potencia.

2.6 Tarjeta de Adquisición de Datos (DAQ) e Instrumento Virtual (VI)

En el contexto a que el proyecto se está llevando a cabo desde su conceptualización, hasta brindarle al usuario seguridad de operación, en la que podamos integrar los diferentes componentes de las fases (semiconductores de potencia y el sistema embebido), se tomó la decisión de dividir el embalaje en dos: contenedor de potencia y contenedor de mando; al dividir el embalaje de componentes se logra practicidad, ergonomía, eficiencia y seguridad. Para lograr una impresión en 3D del embalaje a dimensiones precisas de diseño, se dibujó en el software SolidWorks de la empresa Dassault Systems, generando los códigos necesarios para la impresión (Fig. 6).

Figura 6. Carátula impresa en 3D para el control que forma parte del sistema embebido.

2.7 Tarjeta de Adquisición de Datos (DAQ) e Instrumento Virtual (VI)

El algoritmo de programación del ATMEGA2560, se realizó la integración del Firmware MakerHub que emula una DAQ en el Arduino a una velocidad de 9600 baudios a una resolución de 10 bits (1024 combinaciones) resultando en una lectura mínima del sensor de 4.88mV. Para el Panel Frontal del Instrumento Virtual, se obtuvo una interfaz colorida y de fácil comprensión en la que se puede manipular de forma intuitiva las variables del sistema, logrando observar y controlar en lazo abierto en tiempo real las tonalidades de luz para cada nivel, así mismo se diseñó una Base de Datos que almacena los datos en tablas de Excel, mostrando en gráficas de fácil comprensión, el consumo eléctrico que se obtiene (Figura 7).

Figura 7. Interfaz de Usuario en LabView.

Mediante el software de desarrollo de National Instruments, LabView, se generó la Interfaz Gráfica de Usuario (GUI) intuitiva y el Diagrama de Bloques necesario para control y monitoreo de intensidad lumínica, tiempos e intervalos de inmersión / emersión. Es importante mencionar que tanto el control como monitoreo es en tiempo real (definido por la frecuencia de operación del DAQ), lo que genera un valor agregado al sistema por su capacidad de monitorear y controlar en cualquier instante el proceso de Micropropagación in vitro con Inmersión Temporal.

Para dicho sistema se trabajó con el Arduino MEGA conectando sus puertos a los periféricos, configurando del pin 7 al 13 como PWM para el control de las tiras led y los pines 20 al 21 para el reloj externo DS1307 que sincroniza las inmersiones y la hora del sistema.

2.8 Control Difuso para Sistema Lumínico

Para poder medir la intensidad lumínica aislando la luz ambiental, fue necesario diseñar e imprimir en 3D una estructura (figura 8) para encapsular una fotoresistencia variable (LDR) y un diodo tipo LED, que recibe la señal PWM del Arduino, misma que es alimentada a las diferentes tiras LED.

Figura 8. Encapsulado sensor

Para lograr controlar la intensidad lumínica, se decidió implementar un controlador difuso, el cual está compuesto de cuatro partes principales (figura 9): interfaz de disuficación, base de conocimientos, lógica de decisiones e interfaz de desfusificación.

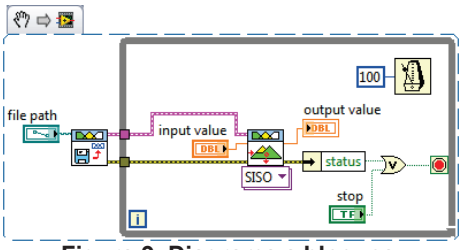


Figura 9. Diagrama a bloques.

Teniendo en cuenta la intensidad lumínica requerida así como la real medida por el LDR, se determina la diferencia entre estas dos como el error. Conociendo el error, mediante el desarrollo de las funciones de pertenencia y las reglas de inferencia difusa que se muestran en la figura 10, se busca reducir significativamente el error anteriormente mencionado.

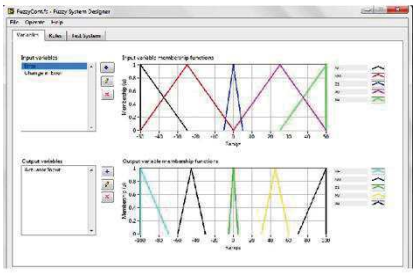


Figura 10. Funciones de pertenencia

3. Resultados

La experimentación lumínica se centró en las variaciones según el color por lo que los tratamientos fueron al 100% de la capacidad luminosa en cada piso, se acondicionó el sistema neumático a 1 bar de presión de salida, como se había planteado anteriormente, tanto para inmersión como para la aireación ya que son las cantidades recomendadas en las especificaciones de los sistemas no automatizados, con los cuales los explantes no presentan daño alguno. En la Fig. 11 se observan diversas tonalidades de luz aplicado en el Agave Tequilana.

Figura 11. Sistema de inmersión temporal automatizado.

Por su parte el análisis de costos de implementación del proyecto, se reduce significativamente el costo de inversión como se observa en la Tabla 2.

Tabla 2. Costos del prototipo automatizado (MXN)

	Articulo	*Precio unitario*	*Cantidad*	*Subtotales*	*Modelo*
Eléctrico/Elec	Tiras Led blanco, rojo, azul.	$420	6 pza	$2520	MODLED-150/BCO
	Tiras led RGB	$795	2 pza	$1,590	MODLED-300/RGB
	Driver Puente H	$93	13 pza	$1,209	L298N
	Arduino	$430	1 pza	$430	MEGA
	Fuente de alimentación	$204	2 pza	$408	12V, 15ª
	Teclado Matricial	$29	1 pza	$29	4x4

trónico	LCD	$44	1 pza	$44	1602ª
	Cables	$13	20 mt	$260	18 AWG
	Timer	$33	1 pza	$33	RTC1302
	Sensor de humedad y temperatura	$45	1 pza	$45	DTH11
	Sensor de luz	$29	1 pza	$29	LDR
Neumático	Compresor	$1,550	1 pza	$1,550	TC-20
	Regulador de presión	$300	5 pza	$1,500	LR-D-MIDI
	Unidad de Mantenimiento	$1,099	1 pza	$1,099	Lrf-d-maxi
	Mangueras	$33	15 mt	$495	Sms
Electroneumático	Electroválvulas	$680	2 pza	$1,360	EMC V5221-08
	Anaquel	$399	1 pza	$399	SV
	TOTAL			$13,000	

En la Tabla 3 se pueden observar los costos de los diferentes sistemas existentes (prototipo automatizado, sistema semi-automatizado y sistema de biorreactores comercial de la marca INFORS HT). Como se estima el sistema de biorreactores comercial por su alto costo, está diseñado específicamente para laboratorios con instrumentos de precisión, pero limitados únicamente al estudio de especie de planta a la vez, dejando de un lado la capacidad de realizar estudios de forma simultánea, como se plantea en el prototipo propuesta.

Tabla 3. Comparativa de precios.

TIPO DEL SISTEMA	PRECIO (MXN)
AUTOMATIZADO	$13,000.00
SIN AUTOMATIZAR	$14,244.00
COMERCIAL (LABFORS 5 LUX)	$514,237.28

El sistema automatizado opera con un compresor de 0.66 kW de potencia, las electroválvulas utilizan la presión necesaria en las inmersiones, lo cual disminuye su arranque a tres veces por día. El tanque permanece encendido 1 minuto por arranque hasta llenarse, lo cual se traduce en una energía necesaria al día de:

E=(0.66 kW)(3/60 hrs)=0.033 kWh, al bimestre esto significa un gasto de 1.98 kWh en energía.

Finalmente, en lo que respecta al Controlador Difuso se obtuvo un porcentaje del +-1% con un tiempo de establecimiento de 2.5 segundos (Figura 12).

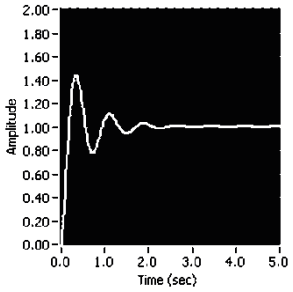


Figura 12. Señal de salida controlada para intensidad lumínica

Como se mencionó al principio del articulo el esquema de control se basa en Luxes, sin embargo para el caso de las plantas, lo que se requiere controlar y medir son PPFD (Densidad de Flujo Fotónico Fotosintético), por tal motivo es necesario aplicar una equivalencia en la interfaz gráfica

de usuario de acuerdo a la siguiente Luxes x factor de conversión, siendo el factor de conversión para el caso de los LED 0.0165, quedando de tal manera la ecuación de la siguiente manera: 0.01165 x luxes. Como detalle importante es necesario mencionar que el sol irradia un total de 108000 luxes ó 2000 PPFD, por lo que el Sistema de Inmersión Temporal debe suministrar mínimo ese porcentaje en irradiancia LED.

4. Conclusiones

Es necesario impulsar el trabajo multidisciplinario que conlleve a la investigación y resolución de problemas enfocados a la producción agroindustrial, vinculando diversas disciplinas para generar nuevas tecnologías y múltiples oportunidades de aprendizaje.
El uso de una plataforma de desarrollo LabVIEW permite versatilidad, portabilidad y apertura a diversos hardwares (DAQ´s), facilitándole al diseñador la generación del diagrama de bloques y construcción de la GUI que cubriera con las necesidades planteadas desde la contextualización del proyecto.

El uso de este tipo de tecnologías gráficas nos permite una mejor presentación con respecto a los sistemas comerciales, ya que nos ofrece la opción de personalización de las herramientas desarrolladas, así como una mejor eficiencia de recursos y tiempo con base a los requerimientos del sistema, capaz de brindar una amplia flexibilidad y actualización a requerimientos futuros proyectos. La inclusión de dispositivos y drivers periféricos al sistema embebido permitieron su utilización por personal no altamente calificado, destacando la practicidad y reducción de costos que estos conllevan, planteando una mejora en la calidad del producto final y mejorando significativamente la presentación en el acabado final del prototipo.

Finalmente, se logra un equipo 100% funcional para experimentos en plantas. De igual manera se pretende realizar mejoras en el controlador y al ser autoajustable no importa las variaciones en el sistema ya que se regula de forma automática, sin embargo el para las diferentes áreas que iluminan las tiras LED, se requiere de otra fuzzificación de parámetros medidos en las LDR, ya que la irradiancia suele variar en cada área y resultaría complejo implementar un controlador por cada LDR y cada tira LED correspondiente.

Referencias

[1] Almansa, E.; Chica, R.; Espín, A. y Lao, M. (2008). *“Modelo sostenible de ajardinamiento de espacios interiores sin radiación luminosa natural: iluminación con LEDs alimentados por energía solar fotovoltaica”.* Actas de Horticultura nº 52. Innovación y futuro en la jardinería. I Simposio Iberoamericano- IV Jornadas Ibéricas de Horticultura Ornamental. Pontevedra, España.

[2] Aragón, C; Escalona, M; Capote, I; Cejas, I; Rodríguez, R; Sandoval, J; Roels, S; Debergh, P y González-Olmedo, J (2006). “Aspectos metabólicos del crecimiento y desarrollo de las plántulas de plátano (CEMSA ¾) micropropagadas en biorreactores de inmersión temporal (BIT)”, *Cultivos Tropicales*, vol. 27, no. 1, pp. 39-44, ISSN 0258-5936.

[3] Atmel corporation (2000). “Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V”. San José, CA. Recuperado de http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf [Consultado: 18/01/2018].

[4] Casierra-Posada, F.; Peña-Olmos, J.E.; Ultrichs, Ch.: *“CRECIMIENTO Y EFICIENCIA FOTOQUÍMICA DEL FOTOSISTEMA II EN PLANTAS DE FRESA (Fragaria sp.) AFECTADAS POR LA CALIDAD DE LA LUZ: IMPLICACIONES AGRONÓMICAS”*. Revista U.D.C.A Actualidad & Divulgación Científica 14 (2): pp: 43 – 53.

[5] Chica, R.; Lao, M. y Almasa, E. (2016). “Evaluación de LED’s para uso agronómico”. I Simposio nacional de ingeniería hortícola. Automatización y TICs en la agricultura. España.

[6] Coughlin, R. y Driscoll, F. (1993). “Amplificadores operacionales y circuitos integrados lineales”. 4ta Edición. México. Prentice-Hall Hispanoamericana.

[7] Cruz, P. P., & Herrera, A. (2011). Inteligencia artificial con aplicaciones a la ingeniería (Vol. 1). Marcombo.

[8] Floyd, T. (2008). “Dispositivos electrónicos” 8va edición. México. Pearson Educación.

[9] Lajara Vizcaino, J. y Pelegri Sebastia J. (2012). “Labview: entorno gráfico de programación”. 2da edición. Marcombo. Isbn: 9788426718686.

[10] Lugo Espinosa, O; Arellano Ostoa, G; Hernández Cote, D. (2017). *Automatización de un sistema de inmersión temporal con base en plataformas abiertas de hardware y software*. Terra Latinoamericana, vol. 35, núm. 3, julio-septiembre, pp. 269-277 Sociedad Mexicana de la Ciencia del Suelo, A.C. Chapingo, México.

[11] Macías Ferro, H.; Ramos Gonzalías, Y.; Ulianov López, Y. (2012). “Estudio de los beneficios de cambio de bombillas de sodio de alta presión por diodos emisores de luz de alto brillo”. *El Hombre y la Máquina*, núm. 39, mayo-agosto, pp. 12-18. Cali, Colombia.

[12] Maxim Integrated. 2015. DS18B20 Programmable Resolution 1-Wire Digital Thermometer. https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf. [Consultado: 21/01/2018].}

[13] Orellano, J; Dellagiovanna, Imanishi, A; Mazzone, V: E. Faed, N (2016). “SISTEMA DE INMERSIÓN TEMPORAL: AUTOMATIZACIÓN DE PROPAGACIÓN *IN VITRO* DE PLANTAS UTILIZANDO HERRAMIENTAS LIBRES” en autoría del congreso Semana del Control Automático - 25o Congreso Argentino de Control Automático. Buenos Aires, Argentina.

[14] Posada Contreras, J. (2005). “Modulación por ancho de pulso (PWM) y modulación vectorial (SVM). Una introducción a las técnicas de modulación”. El Hombre y la Máquina, núm. 25, julio-diciembre, pp. 70-83. Cali, Colombia

[15] Righini, R y Grossi Gallegos, H (2005). *Analisis de la correlación entre la radiación fotosintéticamente activa y la radiación.* Avances en Energías Renovables y Medio Ambiente Vol. 9. Argentina. ISSN 0329-5184.

[16] Rodríguez Arévalo, A; Cabrera Llanos, A y Valencia Flores, J (2003). “Diseño y construcción de los instrumentos de medición para un biorreactor prototipo”. *Revista Mexicana de ingenieria biomedica.* Vol. XXIV. Núm. 1. pp. 55-70.

[17] Rodríguez Castillo, N. A. (2017). Evaluación de la respuesta ecofisiológica de plantas de granadilla (Passiflora ligularis Juss) bajo dos condiciones ambientales en el departamento del Huila (Colombia) (Doctoral dissertation, Universidad Nacional de Colombia-Sede Bogotá).

Ingeniería y Automatización, Capítulo 18, pp. 235 - 244.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño de un Robot Subacuático para la Rehabilitación de Tobillo: Mecanismo y Control de Temperatura

Madrigal López Oscar [1], Désiga Orenday Omar [1],
Talavera Otero Jorge [1] y Guzmán Valdivia Cesar H. [2, *]

[1] Instituto Politécnico Nacional
Unidad Profesional Interdisciplinaria en Ingeniería campus Zacatecas

[2] Universidad Politécnica de Zacatecas
Departamento de Ingeniería Mecatrónica

omadrigall1000alumno@ipn.mx, odesigao@ipn.mx, jtalaveraot@ipn.mx

[*] Autor de correspondencia: cesar.guzman@upz.edu.mx

Resumen

La articulación del tobillo es una de las estructuras más comúnmente lesionada en actividades deportivas. Una lesión de tobillo si no se atiende a tiempo puede generar dolor crónico. El presente artículo tiene como finalidad mostrar el análisis cinemático del mecanismo, así como el control de temperatura del agua de un robot de un grado de libertad subacuático para la rehabilitación de tobillo para pacientes en etapa de recuperación para los movimientos de plantarflexión y dorsiflexión.

Palabras clave: Diseño, control, robot subacuático, rehabilitación de tobillo.

1. Introducción

La rehabilitación, en general, es un proceso de reincorporación física, mental y social de duración limitada que tiene como propósito integrar a una persona a sus actividades diarias [1]. Las lesiones de tobillo son las más comunes que se atienden a diario. En general, el incremento demográfico y la falta de recursos económicos hace difícil que se brinde una rehabilitación de calidad a los pacientes después de haber sufrido alguna lesión de tobillo [2]. Por ejemplo, los centros de rehabilitación públicos cuentan con poco personal para atender a la gran cantidad de personas que requieren de una rehabilitación y por lo tanto no se les brinda la atención adecuada. Para dar solución a esta problemática es necesario el desarrollo de tecnologías que permitan asistir a los terapeutas en la rehabilitación del paciente. Actualmente, existen máquinas de movimiento pasivo continuo (CPM) que las utilizan en los hospitales privados para rehabilitar el tobillo. Sin embargo, su alto costo no permite que sean utilizadas por muchas personas por lo cual esta tecnología no es muy accesible.

Recientemente, una alternativa propuesta en la literatura es la de utilizar la robótica de rehabilitación para ayudar en la recuperación de los pacientes. Hoy en día se han desarrollado una gran cantidad de dispositivos para la rehabilitación de tobillo. Solo por citar un ejemplo, el sistema *TobiBot* fue desarrollado por Guzmán-Valdivia et al. en la Universidad Politécnica de Zacatecas donde proponen un diseño de un sistema interactivo de rehabilitación para el tobillo que cumple con los movimientos de plantarflexión y dorsiflexión [6]. El artículo aborda el diseño del mecanismo de un grado de libertad con movimientos de plantar flexión y dorsiflexión tomando en cuenta los grados de movimiento que nos permite tener el tobillo en este tipo de movimientos. La investigación en México y en el mundo se ha centrado en tecnologías de asistencia para rehabilitación pasiva y activa con el objetivo de mejorar el proceso de rehabilitación física. Pero no se ha abordado una nueva alternativa que es combinando los beneficios del agua caliente en los músculos afectados. En este artículo se

describe el diseño de un robot subacuático para la rehabilitación de tobillo enfocado en el mecanismo y el control de temperatura.

2. Fundamento teórico

2.1 Biomecánica del pie

La estructura musculo esquelética del tobillo es una de las más complejas del cuerpo humano debido a que transmite fuerzas y torques durante el andar [1]. En la figura 1 se muestran los rangos de movimiento del tobillo con sus ángulos máximos permisibles. El rango de movimiento del tobillo para movimientos de plantarflexión y dorsiflexión es de 30° como mínimo y 70° como máximo tomando en cuenta el arco que se realiza durante la marcha [3].

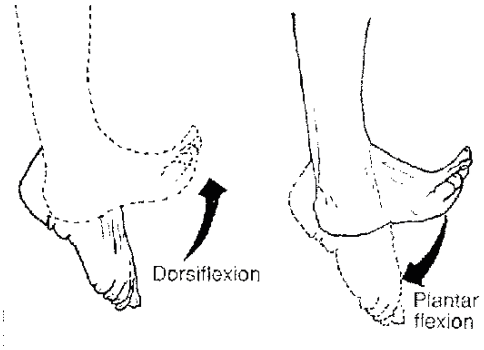


Figura 1. Movimientos biomecánicos del tobillo [3].

Para cuando el pie este en una posición como en la figura 2 los tres ejes son perpendiculares entre sí y los identificamos de esta manera:

- El eje transversal (X) está incluido en el plano frontal y condiciona los movimientos de flexo-extensión del pie, que se realizan en el plano sagital.

- El eje longitudinal de la pierna (Y) es vertical y condiciona los movimientos de aducción-abducción del pie, que se efectúan en el plano transversal y que son factibles con la rotación axial de la rodilla flexionada.

- El eje longitudinal del pie(Z) es horizontal y pertenece al plano sagital.

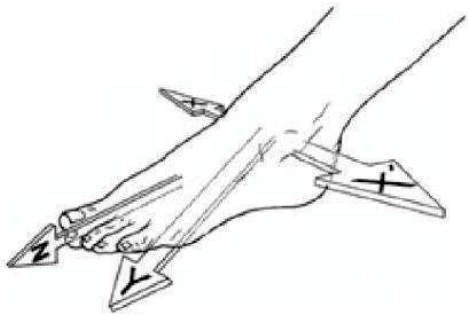


Figura 2. Coordenadas del pie [3].

2.2 Fisioterapia bajo el agua

La rehabilitación en el agua es uno de los tratamientos más utilizados para el esguince de tobillo, por sus beneficios que nos puede ofrecer el agua a una temperatura entre 38° y 40°C. La terapia acuática acelera los tratamientos convencionales, cambiando las inmovilizaciones prolongadas por periodos cortos, estos procesos de tratamientos tempranos benefician a los deportistas de elite. Cabe mencionar que se pueden emular a las vasodilataciones de las compresas por calor, a la liberación de endorfinas que producen la electroestimulación percutánea. Es decir, que este tipo de rehabilitación puede ayudar a acelerar la rehabilitación [5]. La rehabilitación de tobillo en agua se divide en 3 diferentes fases:

- La fase 1 tiene una duración entre 1 a 3 semanas, está caracterizada por la presencia de dolor, inflamación, perdida de la movilidad y edema.

- La fase 2 se caracteriza por una reducción significativa en los síntomas y mejorar de la función: menos dolor, movimientos más amplios y contracción completa.

- La fase 3 evita una vuelta prematura a la práctica de los síntomas como el edema, rango articular incompleto, y disminución de la fuerza.

Existen dos tipos de rehabilitación, activa y pasiva. En la rehabilitación activa los pacientes son capaces de realizar los movimientos de la terapia y en la rehabilitación pasiva el fisioterapeuta tiene que realizar los movimientos al paciente. La rehabilitación más utilizada, según la fisioterapeuta María de la Luz Longoria, es la del movimiento de extensión y flexión que consiste en el movimiento de arriba y abajo del pie para generar movilidad y fuerza en los músculos que cubren el tobillo. El diseño que se presenta en el documento está enfocado a la fase 2 donde los ejercicios realizados se realizan en agua a una temperatura de 37° a 40°C [1].

3. Diseño del mecanismo

Para el diseño del mecanismo se consideraron los ángulos máximos permisibles para el movimiento del tobillo, es decir, el movimiento de plantarflexión y movimiento de dorsiflexión. La figura 3 muestra el bosquejo del mecanismo propuesto.

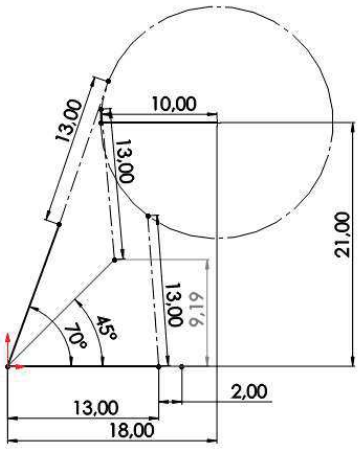


Figura 3. Bosquejo del mecanismo.

Las distancias que se consideraron para las longitudes de los eslabones fueron tomadas a partir de diferentes factores. Como se observa en la figura 4, para el primer eslabón (l_1) se tomó en cuenta los percentiles máximos y mínimos del pie para poder hacer ajustes dependiendo del tamaño del pie del paciente, para el siguiente eslabón (l_2) se consideró la distancia del contenedor del agua para poder tener el sistema electrónico fuera del sistema de rehabilitación para poder evitar accidentes, también en la figura 4 se puede observar que una parte del mecanismo se encuentra dentro del agua, por lo que este diseño permite que el motor que se encuentra en el pivote fijo (b) equilibre el peso del pie en l_1 con el contrapeso (P), además se observa el mecanismo en la primera posición, es decir, el movimiento de plantarflexión donde se realiza un movimiento hacia abajo del pie como se observa en la figura 1.

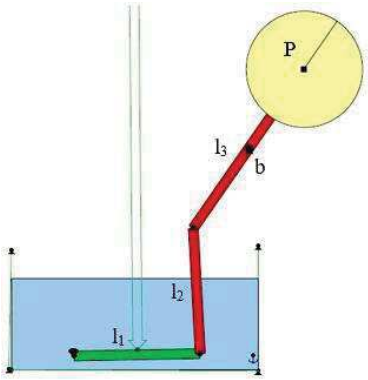


Figura 4. Simulación del mecanismo.

Para el movimiento de dorsiflexión el mecanismo se tendrá que mover 70°, en la figura 5 se observa como el mecanismo empezará en un movimiento de plantarflexión y el punto neutro del pie estará inclinado a un ángulo de 45° debido a que al sentarse el pie se coloca en esta posición de forma neutral.

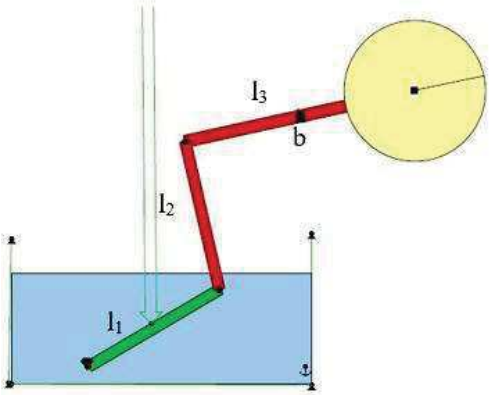


Figura 5. Mecanismo en movimiento.

Después de observar el comportamiento del mecanismo, se realizó el modelo matemático basado en la literatura [4].

$$X_1 - l_1 \cos(\Theta) = 0 \quad (1)$$

$$X_1^2 + Y_1^2 - l_1^2 = 0 \quad (2)$$

$$(X_2 - X_1)^2 + (Y_2 - Y_1)^2 - l_1^2 = 0 \quad (3)$$

$$(X_b - X_2)^2 + (Y_b - Y_2)^2 - l_3^2 = 0 \quad (4)$$

Donde:

- $l_1 \equiv Eslabón\ 1$
- $l_2 \equiv Eslabón\ 2$
- $l_3 \equiv Eslabón\ 3$
- $b \equiv Pivote\ fijo$
- $X_1 \equiv Coordenada\ de\ l_1\ en\ x$
- $Y_1 \equiv Coordenada\ de\ l_1 en\ Y$
- $X_2 \equiv Coordenadas\ de\ l_2\ en\ X$
- $Y_2 \equiv Coordenadas\ de\ l_2\ en\ Y$
- $X_b \equiv Coordenadas\ de\ b\ en\ X$
- $Y_b \equiv Coordenadas\ de\ b\ en\ Y$
- $P \equiv Contrapeso$

Para verificar el modelo matemático planteado es necesario realizar simulaciones de posición y velocidad que se podrán encontrar en los resultados más adelante.

4. Control de temperatura

La figura 6 muestra el sistema de calentamiento del agua en base a resistencias eléctricas.

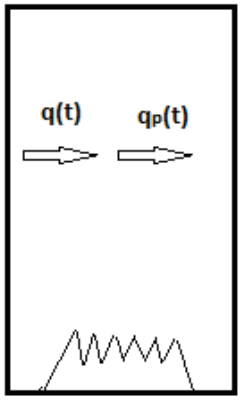


Figura 6. Sistema de calentado de agua.

Las ecuaciones que describen el intercambio de calor del sistema son:

$$\Delta q = C\ T_{des} \quad (5)$$

$$q(t) - q_p(t) = m_{H2O} C_{e_{H2O}} \frac{d\, T_{des}(t)}{dt} \tag{6}$$

$$q_p(t) = \frac{T_{des}(t)}{R_t} \tag{7}$$

Aplicando la transformada de Laplace se obtiene:

$$q(s) - q_p(s) = m_{H2O} C_{e_{H2O}}\, Temp_{H2O}(s) \tag{8}$$

$$q_p(s) = \frac{T_{des}(s)}{R_t} \tag{9}$$

Donde:

- $T_{des} \equiv Temperatura\ deseada$
- $m_{H2o} \equiv Masa\ del\ agua$
- $C_{H2O} \equiv Capacitancia\ temica$
- $R_t \equiv Resistencia\ Térmica$

Las ecuaciones obtenidas se describen por el siguiente diagrama de bloques que se muestra en la figura 7.

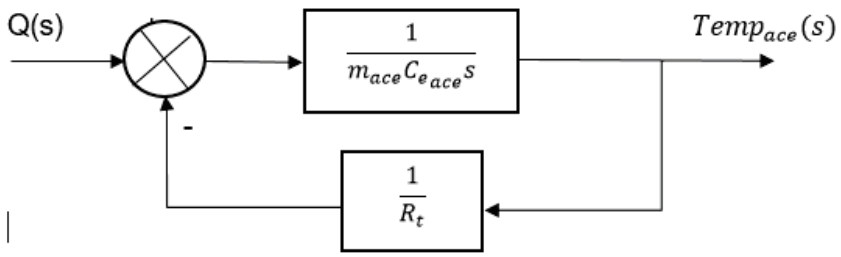


Figura 7. Diagrama a bloques del sistema.

Tomando en cuenta la figura 7, se plantea la siguiente función de transferencia en lazo cerrado.

$$\frac{T_{des}(s)}{q(s)} = \frac{R_t}{1 + m_{H2O} C_{e_{H2O}} R_t s} \tag{10}$$

De modo que:

$$T_{des}(s) = \frac{R_t}{1 + m_{H2O} C_{e_{H2O}} R_t s}\, q(s) \tag{11}$$

Debido a que es necesario controlar la temperatura del agua se propone un controlador PID. En los controladores PID el proporcional (P) puede controlar cualquier planta estable, pero tiene limitaciones en su desempeño y un error en el régimen permanente, el controlador integral (I) se

obtiene a partir de una salida en el proporcional al error acumulado, lo que significa que es un modo de controlar lento y el controlador derivativo (D) hace más rápida la respuesta del control.

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int e(\tau)d(\tau) + K_p\, T_d \frac{de(t)}{dt} \tag{12}$$

Tomando en cuenta la figura 7, el sistema puede reducirse como se muestra en la figura 8.

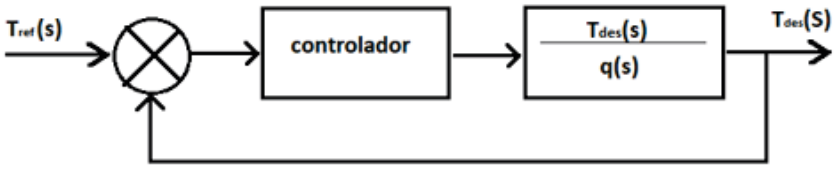


Figura 8. Control de temperatura

Donde:

- $T_i \equiv Temperatura\ inicial$
- $T_d \equiv Temperatura\ deseada$
- $T_{des} \equiv Temperatura\ deseada$
- $K_p \equiv Ganacia\ proporcional$

5. Resultados

5.1 Respuesta de posición y velocidad del mecanismo

La figura 9 muestra la respuesta de posición del mecanismo. Se observa que la posición del pie alcanza el rango completo para realizar la rehabilitación del tobillo ya que el rango de movimiento del mecanismo se encuentra dentro del rango del ángulo del pie para realizar los movimientos básicos de 30° en dorsiflexión y 70° en plantarflexión, es decir un rango de 100°.

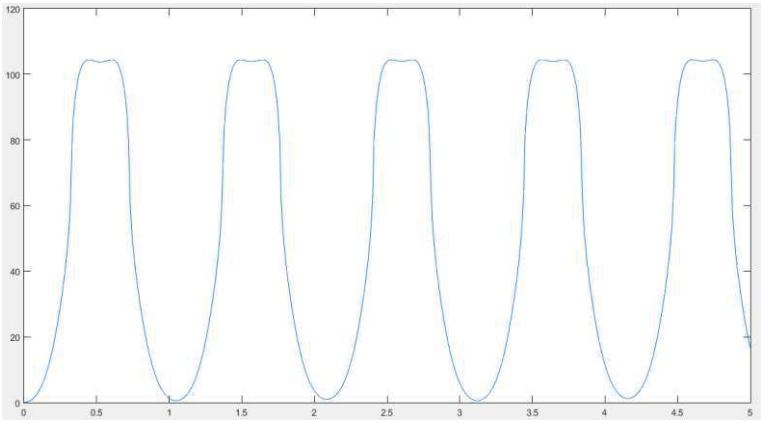


Figura 9. Simulación de la posición del mecanismo.

La figura 10 muestra la velocidad en l_1, con esto se observa que es posible variar la velocidad del mecanismo para realizar la rehabilitación a distintas velocidades según lo indique el fisioterapeuta.

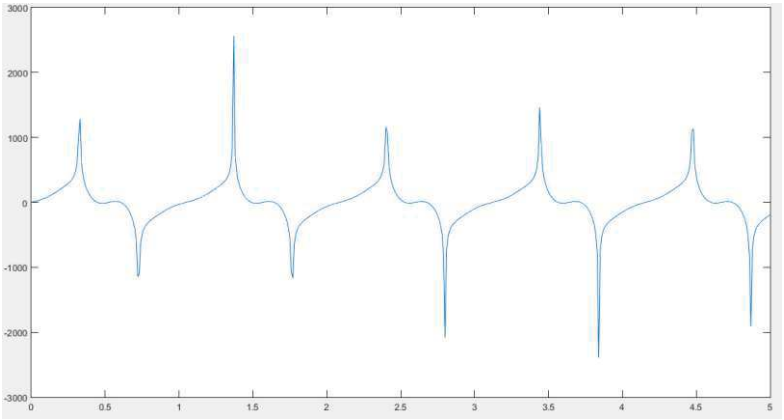


Figura 10. Simulación de la velocidad del mecanismo.

Con estas simulaciones el diseño del mecanismo fue validado. Al realizar el análisis del mecanismo cumple con los movimientos de la biomecánica del tobillo para brindar los movimientos de rehabilitación en plantarflexión y dorsiflexión. Estos resultados apuntan a la formalización del diseño en un software CAD para su realización en 3D. Este es el primer paso para el diseño final de un prototipo funcional el cual pueda incluir la dinámica de la pierna del paciente.

5.2 Respuestas del control PID para el sistema de temperatura

Al tener diseñado el control PID se prosiguió por comprobar los cálculos y comparar el comportamiento antes y después de implementar el control PID. Al simular el sistema de calentamiento de agua a una temperatura dada sin control PID, se puede ver en la figura 11 como el sistema se calienta de forma rápida, la temperatura se eleva rebasando la temperatura deseada lo cual podría producir quemaduras en el paciente, por lo que es necesario implementar el control PID para poder controlar los parámetros y reducir los posibles accidentes en los pacientes.

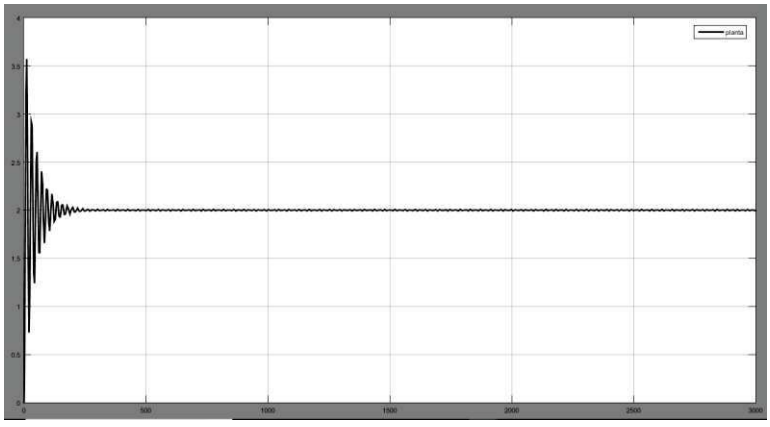


Figura 11. Comportamiento de la temperatura sin control PID.

Después de haber observado el comportamiento del sistema de temperatura sin ningún control se realizó una segunda simulación con un control PID con las ganancias calculadas anteriormente y en la figura 12 como resultado se puede observar el comportamiento de forma controlada de manera que este no rebasa la temperatura deseada a comparación del sistema de la figura 11. También se puede observar que el sistema con control PID a comparación del sistema sin el controlador se comporta de manera más lenta, por lo tanto, al tener los resultados de nuestro sistema con los dos comportamientos podemos elegir el sistema con el control PID debido a que nuestro principal objetivo es la integridad del paciente y no afecta el tiempo de calentamiento.

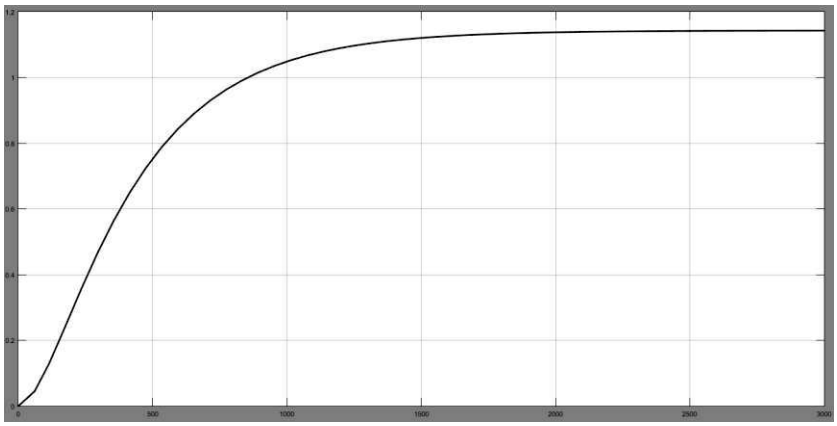


Figura 12. Comportamiento de la Temperatura con control PID.

6. Conclusiones

En este artículo se presentó el análisis del mecanismo, así como el control de temperatura de un robot de un grado de libertad subacuático para la rehabilitación de tobillo, la elección de las dimensiones de los eslabones que permiten al sistema realizar el movimiento de plantarflexión y dorsiflexión llegando a los ángulos permisibles del pie, así como la protección del paciente debido a que se aisló por completa la parte electrónica del mecanismo que se encuentra dentro del agua. El control PID de temperatura permite controlar la temperatura del agua en el rango sugerido para la realización de la terapia. Como trabajo futuro es obtener el modelo dinámico en extenso para analizar la respuesta del motor en condiciones de carga, se tiene contemplado continuar con el desarrollo de este trabajo hasta concluir con la fabricación del prototipo funcional.

Referencias

[1] L. Nagua, "Diseño y construcción de un prototipo automático para rehabilitación pasiva de lesión por esguince de tobillo", *Tesis de Ingeniería Mecatrónica,* 2015.

[2] C. Racu and I. Doroftei, "New Concepts of Ankle Rehabilitation Devices—Part I: Theoretical Aspects", *"Gheorghe Asachi" Technical University of Iasi, Iași, Romania*, vol. 57, p. 223, 2018.

[3] S. Sánchez, J. O., R. Navarro Navarro, R. Navarro García, E. Brito Ojeda and J. Ruiz Caballero, "Bases Biomecánicas del Tobillo", *Canarias Médica y Quirúrgica*, vol. 8, no. 24, 2011.

[4] R. Norton, *Design of machinery*, 4th ed. New York: McGraw-Hill, 2012.

[5] M. Pistolesi, "Rehabilitación del esguince de tobillo con hidroterapia", *ISDe Sports Magazine*, vol. 5, no. 17, 2013.

[6] C. H. Guzmán Valdivia, J. Carrera Escobedo, A. Blanco Ortega, M. Salazar and F. Gómez Bacerra, "Diseño y control de un sistema interactivo para la rehabilitación de tobillo: TobiBot", *INGENIERÍA MECÁNICA TECNOLOGÍA Y DESARROLLO*, vol. 6, no. 1, pp. 255-264, 2018.

[7]M. Sandoval, "Rehabilitación de tobillo", Zacatecas, 2017.

.

Ingeniería y Automatización, Capítulo 19, pp. 245 - 254.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Measurments of the Friction Coefficient for the Vehicles Ball joints, Using a Mechatronic Measure Instrument

Politechnika Łódzka Poland[1], Universidad de Sonora[2]
Wozniak Marek[1], Siczek Krzysztof[1], Ozuna Gustavo[2] y Tapia Pedro[2]
marek.wozniak.1@p.lodz.pl gozuna@industrial.uson.mx

1. Introduction

The set of appearance in the contact area among two bodies in the rest, or moving towards themselves is called friction. As a result of these appearance's resistance of motion are arises. A lot of kinds of the friction can occur. It is possible to differentiate variety of friction conditions types in contact between mating components or inside their materials (figure 1).

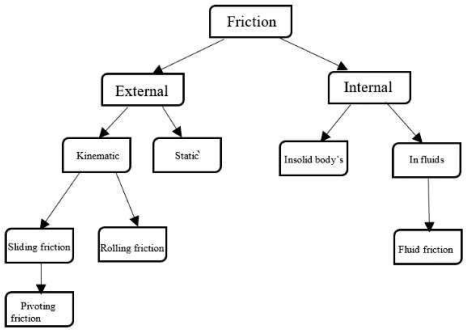


Fig. 1 Variety of friction [8].

Fluid friction is more favorable due lower resistance to motion, so machine constructor try to use it to replace the dry friction. Because co-operating surfaces are not ideally smooth, there are usually conditions of technically dry or boundary friction on the top of surface asperities and surface cavities are fill by lubricant, in the case of such rough contact surfaces the mixed friction occurs. Even, when the amount of applied lubricant seems to be high (figure 2), the unfavorable conditions of motion like small slip speed, reverse motion together with uneven distribution of grease result in generation of mixed friction conditions in contact zone. Particularly, such situation exists in the case of ball joint.

Weiss presented in [7] the results which suggested that the ball joint and friction parameters depended from a complicated variety of nonlinear periodic solutions exists. It is shown that the previously often used approach of computing parameter values where stable sliding turns unstable can not even serve as a first rough indicator for the existence of friction induced oscillations. Instead, existence areas for stable/unstable sliding and co-existing stable limit cycles give a far better impression of the system characteristics, and can even give hints on measures that can help to eliminate friction whirl vibrations from ball joints.

Fig. 2 Lubrication of ball joint with grease

Falah described et al [5] the system contains a coil springs, shock absorbers, wheels and linkages that connect the car to each of the wheels and allow damping their relative motion.

A vehicle ball joint is tangentially loaded as a result of the vertical, lateral and horizontal forces acting on the suspension system when the car operates in long term conditions, under which, the ball joint material does not exceed its elastic limit and these varying loading effects directly influences the performance of the car linkages.

A ball joint is a movable connection enabling a rotational oscillation motion of one connected element relative to the other element. The round ball-joint passes through the axle which, at the moment when the steering wheel of the vehicle is turned by the driver, takes up the wheel turning point. In that, it is possible to deflect the angle and transfer the lateral and longitudinal forces. Since in the time of work ball joints performs swing-rotational movement, they are lubricated by plastic grease supplied by grease nipple or by graphite grease in case when the ball joint pin is knead in the ball joint body by machinery. Figure 3 shows build and description of main components of ball joint.

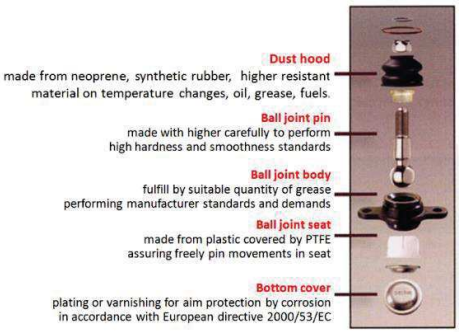


Fig. 3 Build and description of the ball joint elements [1].

Ossa described et al [6] described the analysis and investigation of the causes of the sudden failure of a MacPherson strut suspension system ball joint. The axis of the ball joint element showed a complete fracture which occurred midway between the top and bottom section changes of the element. The failed ball joint element along with a ball joint element not previously used, in order to show the height at which the fracture occurred in the element. The main goal of the paper was to determined failures caused by defective materials, overload or deficient design of the components.

When polishing the ball joints, we minimize friction. The grease is distributed on the ball joint pin and inside the ball joint body it lubricates partly inner surface of the body and partly the Teflon seat, which also ensures smooth operation and faster reaction time of ball joint. The dust cover was made mainly of neoprene (CR), it is characterized by protection against temperature changes in fuel and oil conditions as well as protection against changing weather conditions. When a nylon insert is used, it prevents corrosion of the pins and nuts inside them, and prevents the cap from unscrewing. The failure of ball joints can result from the following fact: when water, sand or other foreign matter gets inside, it causes faster wear of the binding elements in the joints balls which are the effect of blasting the rubber cover [2]

Ingeniería y Automatización, Capítulo 19, pp. 245 - 254.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2. Material and methods

2.1. Material samples.

The friction coefficient between ball joint pin and its seat was determined using the bearing balls made of steel 100Cr6 with diameter 8mm and material samples cut from the seat. The material sample was glued with the steel holder (figure 4) allowing proper mounting of sample in the tester. Before investigations on the tester samples and the analyzed steel balls were cleaned using alcohol and dried in the stream of sprayed air. Material samples coming from cars: Honda Accord (year of production 2009, mileage 195000km), Peugeot 207 (year of production 2008, mileage 105000km), Renault Master (year of production 2006, mileage 280000km). All of the ball joints are mounted originally in mentioned cars. Described cars didn't pass the inspection on the diagnostic station for ball joints on the control arm reason.

Fig. 4 Material sample cut from the ball joint seat and glued with the steel holder

2.2. Test bench for determination friction coefficient

Figure. 5 shows the test bench for determination of the friction coefficient by the string twist angle measurement.

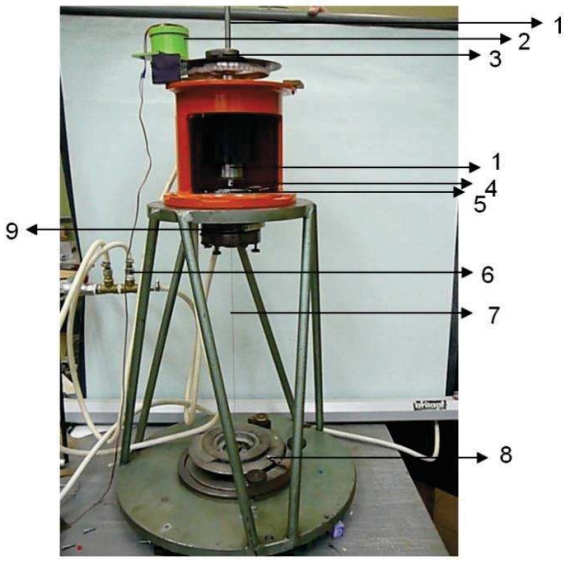


Fig. 5 Test bench for the friction moment and friction coefficient for the frictional pair ball joint – spherical cap; 1 – shaft, 2 – electric motor, 3 – weights, 4 – ball joint, 5 – tested ball joint, 6 –compressed air supply, 7 – string,8 – disc to read the angle of torsion, 9 – aerostatic table [3].

The measuring station consists of two cylindrical and axial cylindrical bearings. The drive is supplied from an electric motor (2) which is transferred between the belt gear on the shaft (1), which ends with a ball joint. By adding weights (3) on the shaft (1) load of the tested friction pair is changed. At the aerostatic table (9) we place the ball joints tested (5) connected by measuring the string (7). The

other end of the string is attached to the table (8) thanks to which we can read from the scale the compensation angle plotted on the table (8).

The tests were carried out on the test bench by measuring the torsion angle of the string as a function of charging. After mounting the ball joint and measuring the ball on the test bench, the electric motor was started. The measuring shaft with the ball rotated at a constant speed - 36 rpm. Then the test bench was loaded with a step of 2 N in the range of 7.06 ÷ 16.96N. Then the test bench was lightened in reverse. Weights were added for 500 cycles.

3.3. The pivoting friction

The pivoting friction occurs during kinetic sliding of two surfaces along one axis (figure 6).

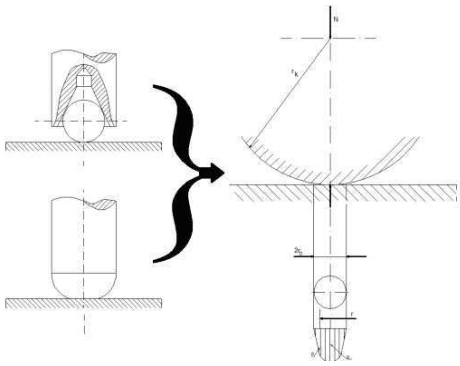


Fig. 6 Model of the bearing type spherical cap on plate; p - Hertz pressure, p_{max} - maximum Hertz pressure.

Particularly, as a result of continuous rotation or gyroscopic movement around the normal point at the contact point between ball and assumed plane, pivoting friction is created, which is analyzed in the article.

Indeed, there is the Hertz contact between mating surfaces of ball and inner sphere, but as the radius of inner sphere is much higher than the radius of the ball, the analyzed contact can be approximated using ball-plane type contact. For such contact, the contact pressure p can be determined from the formula (1):

$$p = p_{max}\sqrt{1-\left(\frac{r}{r_o}\right)^2} \quad (1)$$

The maximum contact pressure p_{max} can be obtained from the formula (2)

$$p_{max} = \frac{1}{\pi}\sqrt{\frac{6N\frac{1}{r_k^2}}{\left(\frac{1-\nu_1^2}{E_1}+\frac{1-\nu_2^2}{E_2}\right)^2}} \quad (2)$$

The radius r_o of contact zone can be obtained from the formula (3)

$$r_o = \Theta\sqrt[3]{\frac{3}{4}N\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right)} = \Theta\sqrt[3]{\frac{3}{4}N\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right)} \quad (3)$$

$$= \Theta\sqrt[3]{\frac{3}{4}N\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right)}$$

where: υ - Poisson ratio, E - Young's modulus, N - axial force, r_o - contact radius, r_k - sphere radius.

The torque of friction M_t between analyzed ball and plain can be calculated from the formula (4)

Ingeniería y Automatización, Capítulo 19, pp. 245 - 254.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

$$M_t = \int_0^{r_o} 2\cdot\pi\cdot\mu\cdot p\cdot r^2\,dr = \int_0^{r_o} 2\cdot\pi\cdot\mu\cdot p_{max}\cdot\sqrt{1-\left(\frac{r}{r_o}\right)^2}\cdot r^2\,dr =$$

$$= 2\cdot\pi\cdot\mu\cdot p_{max}\cdot\int_0^{r_o}\sqrt{1-\left(\frac{r}{r_o}\right)^2}\cdot r^2\,dr = 2\cdot\pi\cdot\mu\cdot p_{max}\cdot 7{,}1633\cdot r_o^3 = \quad (4)$$

$$= 2\cdot\pi\cdot\mu\cdot p_{max}\cdot 7{,}1633\cdot(0{,}721)^3\cdot N\cdot 2\cdot r_k\cdot\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right) =$$

$$= 33{,}722\cdot\mu\cdot p_{max}\cdot N\cdot r_k\cdot\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right)$$

The friction coefficient between bearing ball and plane can be obtained from the formula (5)

$$\mu = \frac{M_t\left(\frac{1}{r_k}\right)}{33{,}722\,p_{max}N\left(\frac{1-\nu_k^2}{E_k}+\frac{1-\nu_p^2}{E_p}\right)} \quad (5)$$

The measured value of the torque of friction Mt can be determined from the formula (6)

$M_t = \alpha_{aver}\cdot k_{string}$ (6)

where: α_{aver} - average angle of string torsion, k_{string} – torsional spring constant

2.4. Calibration of the strings

The torsional spring constant was measured on the special unit. In order to eliminate external vibrations from the environment, string patterns are placed on the table on special measuring devices for the aerostatic bearing (Fig. 7, 8) . Aerostatic fed bearing pressure of 10 bars. In order to obtain the actual value of the angle of torsion measurements made over ten series of successively burdening string weights G (figure 7) with a mass of about 2 g, 4 g, 6 g and 8 g (accurate mass values are given in the tables of measurement data). The arm of the torque generated by force G from weights was 50 mm. In this way it was obtained data to calculate the torque values and to obtain characteristics $M_t(\alpha)$ for the studied strings. The string constant was calculated as the slope of the related characteristic curve $M_t(\alpha)$.

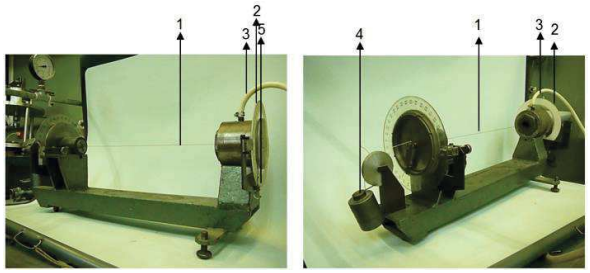


Fig. 7 The scheme of test bench for string calibration: 1 – string, 2 - disk for read torsional string angle, 3 – aerostatic bearing, 4 – weight for string load, 5 – the place for weights position [8].

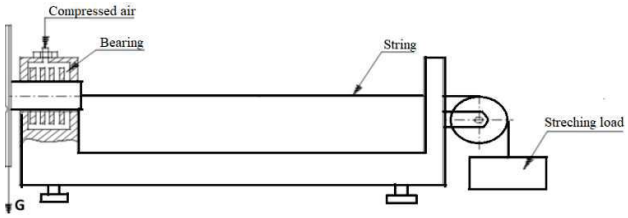


Fig. 8 The scheme of test bench for string calibration, where G is a load making string turns [8].

2.5. Model for contact between ball pin and its seat

To calculate the friction torque between real ball joint pin and its seat the FEM model was elaborated (figure 9a). The axial symmetry along the Y axis was included therein. The model contain spherical part of the pin and simplified geometry of the seat. The outer spherical surface of bearing was assumed to be fixed. The top surface of the pin part was loaded by the pressure related to the longitudinal force loading real ball joint. Between ball surface and inner spherical surface of the seat the contact elements and initial interference was introduced. The material models: of steel for the ball and of PEEK for the seat, respectively, were introduced. The generated grid of finite elements was shown in the figure 9b.

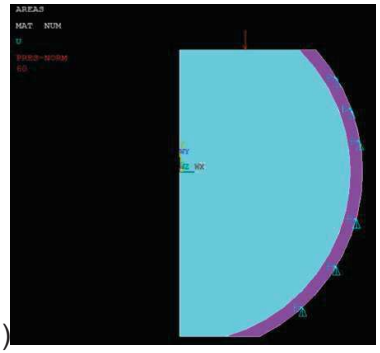
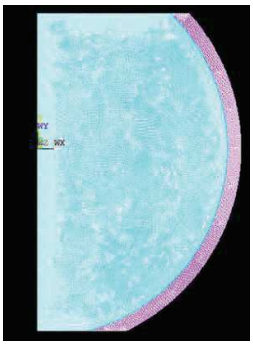
a)  b) 

Fig. 9 a) The FEM model of ball pin – seat set and boundary conditions, b) grid of finite elements.

The friction torque M_{tbj} in the ball joint can be determined from the formula (7)

$$M_{tbj} = \mu \cdot 2\pi \cdot \sum_{i=1}^{i=N} p_i \cdot l_i \cdot r_i^2 \quad (7)$$

where: p_i – average contact pressure for *i*-th contact finite element, l_i – length of the *i*-th contact finite element, r_i – distance between centroid of the *i*-th contact finite element and the Y axis in the model, *N* – number of contact finite elements.

3. Results

3.1. Characteristics for torsional springs

The obtained characteristic $M_t(\alpha)$were presented in the figures 10 and 11. After analyzing such figures, we can see that the actual waveforms are slightly different from the trend line, because the lines are opened by the smallest average square characteristics defined. The difference in these dependence may result from inaccurate reading of the angle from the measuring instruments and the end of the coicidation measurement on the scale. Readings can be obtained using digital measuring instruments.

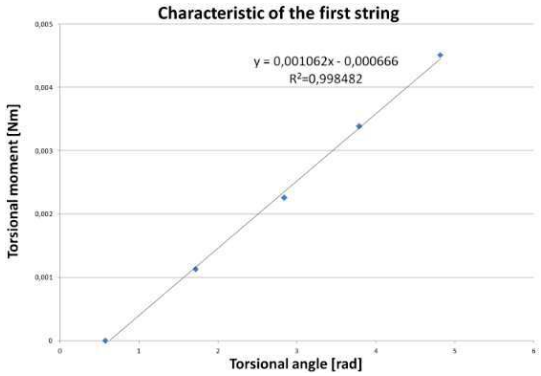


Fig. 10 Characteristic of the first string.

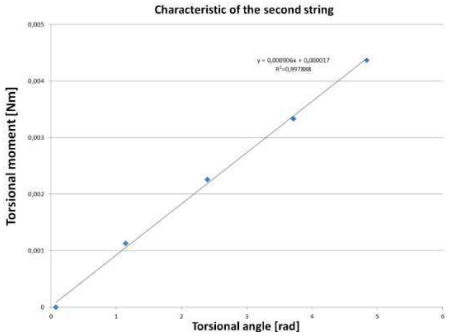


Fig. 11 Characteristic of the second string.

3.2. Friction coefficient obtained from the tester

Friction coefficients obtained from the tester for the seat material samples coming from cars: Honda Accord (year of production 2009, mileage 195000km), Peugeot 207 (year of production 2008, mileage 105000km), Renault Master (year of production 2006, mileage 280000km) were presented in the Figure 12. This figure shows also approximation equations for the cars listed. Table 1 shows the values of the coefficient of friction. It can be seen, that friction coefficient decreases nonlinearly with the increase of the contact zone loading.

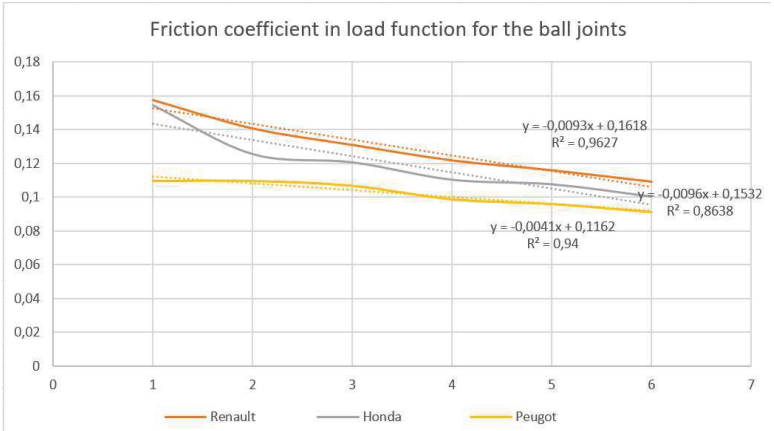


Fig. 12 Friction coefficient in load function for Honda Accord, Peugeot 207 and Renault Master

Tab.1 Values of friction coefficient and P_{max} for selected cars.

Load [N]	Pmax [Pa]	u- friction coeficcient		
		Renault Master	Honda Accord	Peugeot 207
7,04	1,06E+09	0,15748	0,15438	0,10965
9,15	1,15E+09	0,14067	0,12539	0,10965
11,14	1,22E+09	0,13091	0,12051	0,10678
12,85	1,29E+09	0,12176	0,11016	0,09871
14,9	1,35E+09	0,11593	0,10746	0,09592
16,96	1,39E+09	0,10914	0,10018	0,09109

3.3. Contact pressures and stresses from FEM model

The obtained values of contact pressure between the ball part of pin and the seat, for the lack of loading force was presented in the figure 13. Obtained values were below 6.6 MPa. The related values of von Misses stresses were shown in the figure 14. Obtained values were below 7.8 MPa. Obtained values of contact pressure and von Misses stresses resulted from the existing interference of 0,015% between ball surface and inner surface of the seat. It was also obtained gradients of contact pressure and von Misses stresses for the interference of 0,15%, which were almost the same in the shape as those for the interference of 0.015%, however, obtained respective values were almost ten times higher than the first ones.

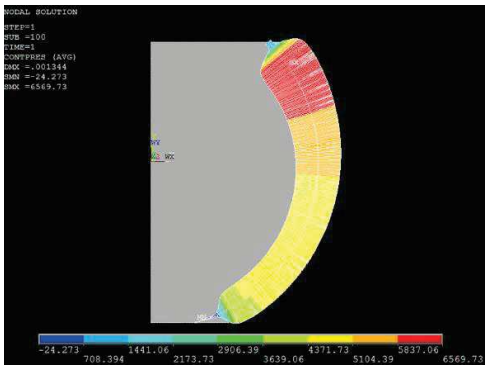


Fig. 13 Contact pressure for the case without loading of contact zone

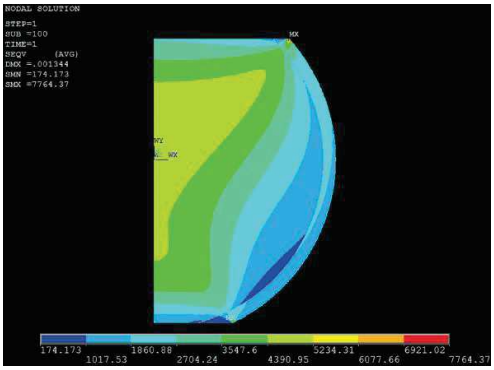


Fig. 14 von Misses stresses for the case without loading of contact zone

The obtained values of contact pressure between the ball part of pin and the seat, for the case of loading force equal 3.5 kN was presented in the figure 15. Obtained values were below 7.9 MPa. The related values of von Misses stresses were shown in the figure 16. Obtained values were below 8.1 MPa. Obtained values of stresses resulted from the existing interference of 0,015% between ball surface and inner surface of the seat and from the applied vertical loading. The effect of the loading on the obtained maximal values of stresses seems to be small in comparison to the analogical effect of the initial interface between ball surface and inner surface of the seat. Although, obtained distributions of contact pressure and von Misses stresses are different.

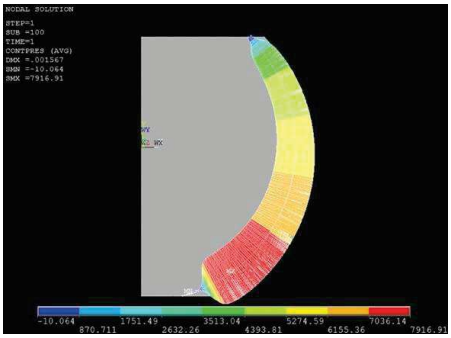


Fig. 15 Contact pressure for the case without loading of contact zone

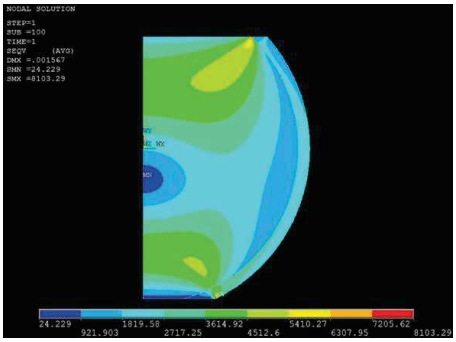


Fig. 16 von Misses stresses for the case without loading of contact zone

Friction coefficient values in contact between the ball part of pin and the seat were predicted using extrapolation of curves obtained from the tester for the contact bearing ball-sample from the ball joint seat. The obtained waveforms were presented in the figure 17. Table 2 presents values of friction torque M_t calculated from the formula (7), using friction coefficient values from the Fig. 17 and obtained contact pressure values from the FEM model, for selected cars and interference between ball surface and inner surface of seat in the respective ball joint. Obtained values of the friction torques were in range (26,4-59,4). For the case of the Peugeot 207 obtained values were the lowest.

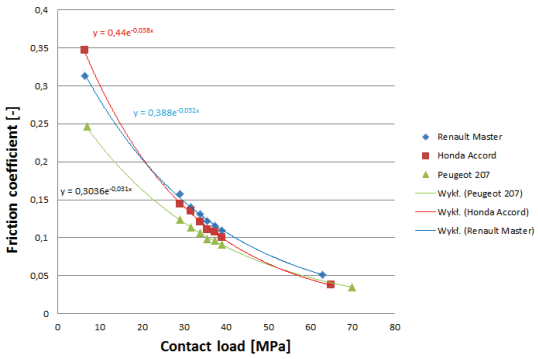


Fig. 17 Predicted values of the friction coefficient for seat materials coming from three vehicles

Tab.2 Values of friction torque M_t for selected cars and interference between ball surface and inner surface of seat in the ball joint.

Load [N]	Interference [%]	M_t - friction torque [Nm]		
		Renault Master	Honda Accord	Peugeot 207
0	0,015	33,7	36,9	26,4
0	0,15	39,9	43,6	31,2
3500	0,015	59,4	49,9	47,0

7. Conclusions

- It can be noticed that the values of the coefficient of friction decrease exponentially with the increase of contact loading.

- The values of the coefficient of friction varies in range 0.09-0.15 for three studied material samples cut from three different ball joint seats. The smallest values are obtained for the case of Peugeot 207 case.

- Using analytical linear models to approximate the dependence of the friction coefficient on loading results in differences between values of the friction coefficient from the experiment and from such models below 0,4% in whole load range.
- Obtained values of the contact pressure between ball and material sample investigated on the tester can resulted in local plastic deformations in contact zone.
- Obtained stresses in the ball joint resulted mainly from the effect of initial interference between ball surface and inner surface of the seat. Loading of the ball joint by the vertical force influenced such stresses in lesser extent.
- The tenfold increase in interference between ball surface and inner surface of seat caused a tenfold increasing of contact pressures and von Misses stresses in the analysed ball joints.

4. References

1. Wozniak M., Ozuna G., De La Fuente P., Jozwiak P., Pawelski Z.: *Test bench with AFM and STM modules for wear researches passengers car suspensions elements*. International Virtual Journal for Science, Technics and Innovations for the Industry MTM. Year VI, Issue 4/2012, pp.9-11, Bulgaria, 27-29.06.2012,
2. Wozniak M., Kubiak P., Pluciennik P., Slowik P., Pijanowski K., Siczek K., Ozuna G.: *Define of Wear Degree of Selected Car Suspension Elements*. Fisita 2016 World Automotive Congress, 26-30 September 2016, Busan, Korea
3. Wozniak M., Ozuna G., De La Fuente P., Jozwiak P., Pawelski Z.: C*omparision of researches of friction coefficient in concentrated contact for the stress: steel-steel and steel-magnesium alloys*. International Virtual Journal for Science, Technics and Innovations for the Industry MTM. Year VI, Issue 6/2013, pp.51-54, Bulgaria, 1-2.07.2013,
4. *Geren N., Osman O. A., Melih B. "Parametric design of automotive ball joint based on variable design methodology using knowledge and feature-based computer assisted 3D modelling" Engineering Applications of Artificial Intelligence, Volume 66, November 2017, Pages 87-103,*
5. A.H. Falah, M.A. Alfares, and A.H. Elkholy, *Failure investigation of a tie rod end of an automobile steering System*, Mechanical Engineering Department, Kuwait University, Safat 13060, Kuwait, 2006
6. Ossa E.A., Palacio C.C., Paniagua M.A.: Failure analysis of a car suspension system ball joint. Engineering Failure Analysis 18, 2011, pp.1388–1394.
7. Weiss C., Morlock M.M., Hoffmann P.N.: Friction induced dynamics of ball joints: Instability and post bifurcation behavior. European Journal of Mechanics - A/Solids Volume 45**,** May–June 2014, Pages 161-173,
8. Wozniak M., Ozuna G., De La Fuente P., Jozwiak P., Pawelski Z.: C*omparision of friction coefficient for selected car suspension elements.* International Virtual Journal for Science, Technics and Innovations for the Industry MTM. Year VIII, Issue 4/2014, pp.23-25, ISSN 1313-0226,
9. Löfgrena H.B.: *A solution to the hydrodynamic lubrication of a circular point contact sliding over a flat surface with cavitation.* Theoretical & Applied Mechanics Letters 2, 032004, 2012,
10. Pati P.B., Kharade M. V.: *Finite Element Analysis and Experimental Validation of Lower Control Arm.* International journal of Engineering Development and Research, Volume 4, Issue 2, 2016, pp. 1914-1922,
11. Kang B.B.: *Theoretical model of ball joint squeak*. Journal of Sound and Vibration, Volume 330, Issue 22, 24 October 2011, pp.5490-5499.

Ingeniería y Automatización, Capítulo 20, pp. 255 - 271.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño y Construcción de un Prototipo de Manipulador Robótico de 4 Grados de Libertad para Fines Educativos

Carmona Ruiz Deisy[1], Carrasco López Rosaura[1], Rebeca Itai Cruz Pacheco[1], Petrilli Barceló Alberto Elias[2] y Lugo González Esther[1]

[1]Instituto de Electrónica y Mecatrónica, Universidad Tecnológica de la Mixteca
[2]División de Posgrado- Universidad Tecnológica de la Mixteca
Huajuapan de León, Oaxaca, México

Resumen

*En este trabajo se presenta*n *los aspectos principales del diseño y construcción de un manipulador robótico de 4 grados de libertad. El prototipo se realizó siguiendo la metodología del diseño mecatrónico y tiene como propósito fines educativos. Se presenta la propuesta de diseño del manipulador robótico con base en los requerimientos de las asignaturas de Robótica I y diseño mecatrónico. Respecto a la parte del control de movimientos, se utilizó Matlab para resolver la cinemática directa a partir de las matrices obtenidas por Denavit-Hartenberg y con el método geométrico se obtuvo la cinemática inversa. Con la herramienta GUIDE se creó la interfaz para hacer la interacción del usuario con el Robot. Se utilizó el software Solidworks® para hacer el diseño del robot y Robotics tool para simular las trayectorias. Como resultados se presentan la secuencia de operaciones de diseño, el prototipo del manipulador robótico y la implementación del control para dibujar figuras geométricas.*

Palabras clave: Robot, cinemática inversa, cinemática directa, diseño, figuras geométricas.

1. Introducción

En las universidades o escuelas en donde se imparte la asignatura de robótica, no siempre se cuenta con manipuladores para hacer pruebas o implementar controles, por lo que es necesario construir los prototipos didácticos como se muestra en [1][2][3]. Para el diseño y manufactura existen diversas metodologías y formas de construcción, pero debido a la necesidad por tener prototipos para implementar los conocimientos y validar resultados, en los últimos años se ha incrementado el uso de proyectos impresos en 3D [4][5], por la facilidad que presenta para realizar prototipos rápidos. Para el caso del diseño de robots de bajo costo, lo que ha revolucionado es realizar proyectos con impresión en 3D y utilizar tarjetas como Arduino para una fácil y rápida programación. Estudiantes de distintas universidades como en [7][8], lo han empleado en sus diseños y prototipos. Por ejemplo el proyecto creado por un estudiante de la universidad Carlos III, permite el desarrollo del modelo cinemático, el control y la simulación mediante un ordenador de un brazo robótico imprimible [4].
Con base en algunas de estas propuestas y lo solicitado por los profesores que imparten las asignaturas de Robótica y Diseño Mecatrónico en la Universidad Tecnológica de la Mixteca, se realizó el diseño y manufactura de un manipulador robótico con 4 grados de libertad. Para la construcción e implementación del control se utilizó Matlab para resolver los cálculos de la cinemática directa e inversa y realizar la comunicación por el puerto UART. Con el entorno visual GUIDE se realizó la interfaz para la comunicación del usuario con el manipulador a través del Arduino UNO y con el programa Robotics Tool se realizó la simulación de las trayectorias.

2. Metodología

La metodología que se usó para diseñar, construir y probar el funcionamiento del manipulador robótico de 4 GDL, es la que se basa en el diseño mecatrónico [9].

Con base en esta, el trabajo se dividió en tres etapas. En la primera se define el prototipo, su forma y su aplicación. Para realizar el concepto de diseño, se consideraron variables como: dimensiones, alcancé, carga máxima, peso y área de trabajo. Para obtener un prototipo de forma rápida, ahorrando tiempo y dinero, se utilizó un software CAD para el diseño de cada pieza y para el ensamble final, además de ser la base para las simulaciones de movimiento, las cuales se hicieron con el software Robotics toolbox de Peter Corke [11]. La segunda etapa está dada por el proceso de fabricación de las piezas, las cuales se diseñaron en el CAD Solidworks®. Para hacer los cortes y dar el formado correspondiente a las piezas, se usó el software VISIO CAD y se maquinaron con la CNC. En la tercera etapa se realiza la implementación del sistema de control de trayectoria y una interfaz, utilizando una tarjeta (en este caso Arduino®), para garantizar el funcionamiento de los servomotores del brazo robótico.

3. Desarrollo

El problema general de los brazos manipuladores es realizar una tarea específica sin perturbaciones o movimientos bruscos, en este caso una trayectoria predeterminada (figuras geométricas) con movimientos suaves y más precisos en sus articulaciones. Para esto, es necesario construir un prototipo que tenga peso ligero, uso de motores con el par suficiente para generar las trayectorias y tenga un control fácil de manipular.

Las variables para realizar los movimientos son los ángulos de las articulaciones que toma al manipulador llegar a una posición determinada y la corriente que se les suministra a los motores. Las variables de entrada que se ocupan son los puntos que conforman la trayectoria que debe seguir el manipulador y las variables de salida son los ángulos que debe tomar el manipulador para alcanzar un punto en específico.

3.1 Definición del diseño del manipulador robótico.

La metodología básica para obtener los requerimientos del cliente en este caso es la de QFD y PDS, con el objetivo de tener mayor acceso a las prioridades y necesidades para el desarrollo de la máquina, base en el diseño CAD y construcción. Con base en QFD, los requerimientos básicos son:

- Rapidez de producción
- Adaptable a Materiales comerciales
- Fácil manejo
- Robustez
- Fácil Mantenimiento
- Diseño Electrónico
- Diseño Mecánico
- Control Eficiente
- Dimensiones Óptimas:
  -El brazo estirado debe tener unas dimensiones de 0 a 40 cm.
  -El peso aproximado que aguantan los motores es de 15KG/cm.

Así mismo, en la tabla 1 se muestra la información generada del diseño y resumida por la metodología PDS.

Tabla 1. Tabla de PDS (Product Design Specification) del manipulador robótico.

GENERAL	PARTICULAR	DESCRIPCION
RENDIMIENTO	Función	El manipulador robótico tiene como objetivo dibujar figuras geométricas.
	Metodología	Metodología de Denavit-Hartenberg para el cálculo de la cinemática directa. La cinemática inversa se resuelve por el método geométrico y la trayectoria por método punto a punto.
	Apariencia	La interfaz debe captar la atención del usuario.
	Eficiente	La comunicación debe ser eficiente para que el movimiento del ancho de pulso para los servos se haga cuando se indique.
	Peso	Es importante que sea ligero para que el servomotor sea capaz de rotar la articulación del manipulador, este no debe ser mayor a 15 Kg.
	Dimensiones	El robot debe tener un largo de 40 cm en posición vertical extendido
MANUFACTURA	Procesos	Ensamblaje por tornillos
		Fresado
	Materiales	Ligero
		Fácil de fresarlo
		Resistente
		Económico
		Materiales comerciales
	Uniones	Las uniones deben tener la menor fricción posible y estar fijas para que el robot se mueva adecuadamente
CONTROL Y SOFWARE	Matlab®	Se utilizaron las herramientas GUIDE, la base de programación y Robotics Tool.
	Arduino UNO®	La comunicación por puerto serial entre la interfaz y el robot se realizó a través del Arduino UNO el cual generaba el ancho de pulso para los servomotores
	Simulaciones	Matlab®
		SolidWorks®
		Robotics Toolbox™

Al definir el tipo de manipulador robótico que se desea diseñar, considerando los requerimientos, debe desarrollarse la cinemática directa e inversa, estudios de área de trabajo y singularidades [10].

Para resolver la cinemática, es necesario tener la información básica del robot como es: manipulador de 4 GDL, de dimensiones menores a 40 cm, peso máximo de 15 Kg/cm., generador de trayectorias sencillas, fácil de programar y tener una interfaz que permita fácil interacción.

Con el Software SolidWorks® se realizó el diseño CAD de cada eslabón derivado de los requerimientos. En la figura 1 se muestra el ensamblaje final realizado con el Software SolidWorks® con una vista explosionada de las piezas que fueron maquinadas, con los servomotores y el efector final. En la tabla 2 se describen las partes que lo forman. En la figura 2 se muestra la vista lateral del manipulador robótico y en la figura 3 se observa el robot en posición extendida.

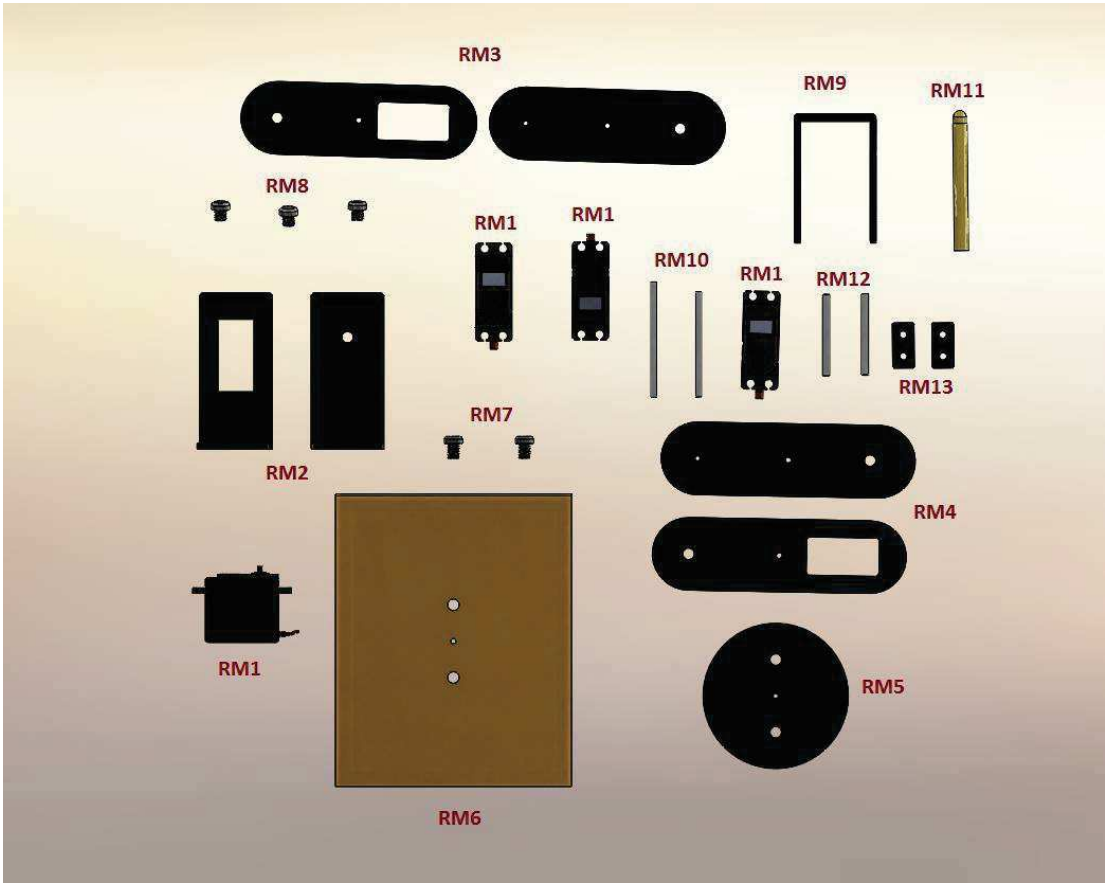


Figura 1. Vista Explosionada del manipulador robótico

Tabla 2. Componentes del manipulador.

Código	Descripción	Cantidad
RM1	Servo motores MG995.	4
RM2	Base del manipulador	2
RM3	Antebrazo	2
RM4	Brazo	2
RM5	Base redonda	1
RM6	Soporte Rectangular	1
RM7	Tornillos de Base 1/4	2
RM8	Tornillos de Unión 3/16	3
RM9	Efector Final	1
RM10	Ejes	2
RM11	Plumón	1
RM12	Eje del plumón	2
RM13	Soporte del plumón	2

Figura 2. Diseño de Manipulador robótico en vista lateral.

3.1.1. Cinemática Directa

En este proyecto se calculó la cinemática directa para determina la posición y orientación del efector final, usando los valores de las variables de unión. Con la metodología de Denavit-Hartenberg (DH) se estableció la ubicación de los sistemas de referencia de los eslabones.

El manipulador cuenta con 5 eslabones y 4 articulaciones y para su análisis se estableció el sistema de coordenadas con Zo apuntando en la dirección del eje de actuación de la articulación 1. Para visualizar se muestra la figura 3 indicando el sistema coordenado del manipulador robótico y en la tabla 3 se tienen los parámetros de Denavit- Hartenberg.

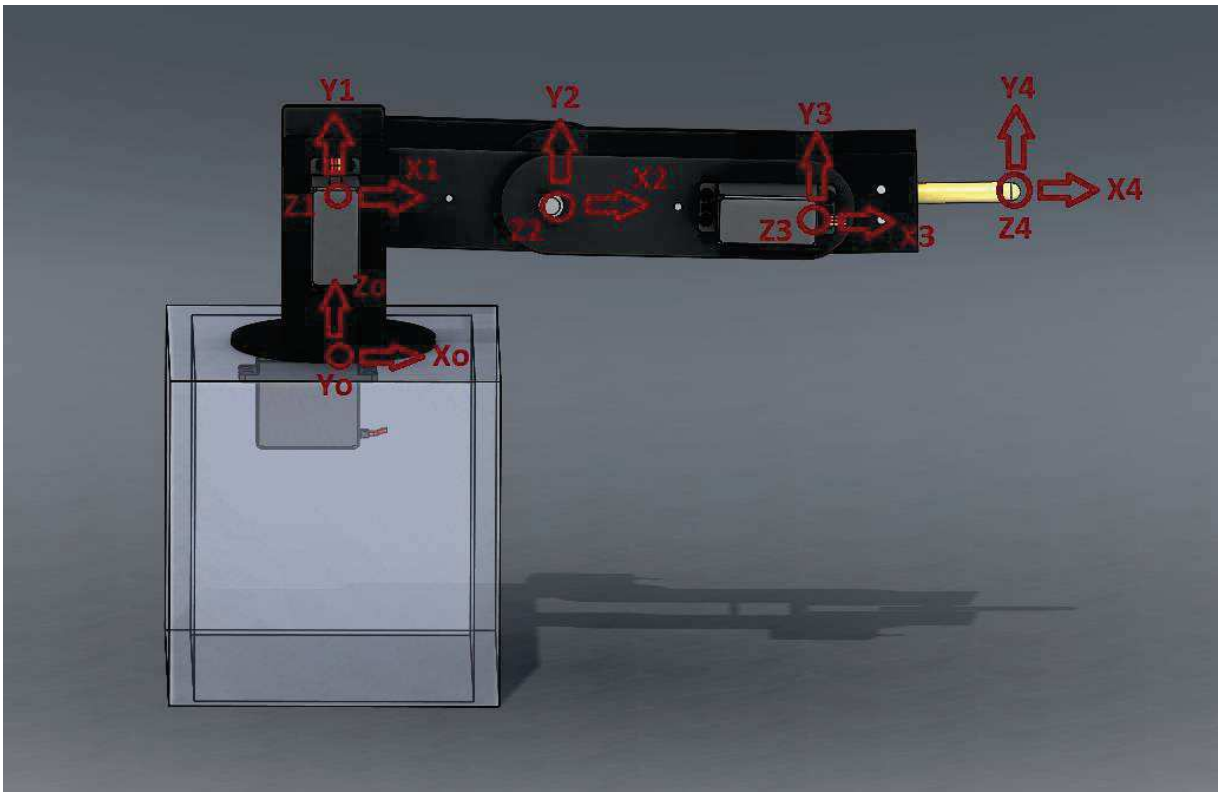


Figura 3. Manipulador robótico, con brazo extendido.

Tabla 3. Parámetros Denaviy-Hartenberg

	q_i	α_i	d_i	θ_i
1	0	-90	6.1	θ_1^*
2	9.5	180	0	θ_2^*
3	8.9	180	0	θ_3^*
4	12.2	0	0	θ_4^*

Se sustituyen los parámetros de D-H en la matriz de transformación y se obtiene la matriz para cada articulación. En las ecuaciones de la 1 a la 4, se muestran las matrices de cada articulación.

$$A_1^0 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & 0 \\ S\theta_1 & 0 & -C\theta_1 & 0 \\ 0 & 0 & 1 & 6.1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$A_2^1 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & 9.7C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & 9.7S\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$A_3^2 = \begin{bmatrix} C\theta_3 & -S\theta_3 & 0 & 8.9C\theta_3 \\ S\theta_3 & C\theta_3 & 0 & 8.9S\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$A_4^3 = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & 12.2C\theta_4 \\ S\theta_4 & C\theta_4 & 0 & 12.2S\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

La Matriz de Transformación final ecuación 5 se resuelve con un programa realizado en Matlab®.

$$T_n^0 = A_1 A_2 A_3 \dots A_n = A_n^0 \quad (5)$$

El resultado de esta matriz es:

$$\boldsymbol{A_4^0} = \begin{bmatrix} C\theta_1C\theta_2C\theta_3C\theta_4 & C\theta_1C\theta_2C\theta_3C\theta_4 & & \\ -C\theta_1S\theta_2S\theta_3C\theta_4 & -C\theta_1S\theta_2S\theta_3C\theta_4 & S\theta_1 & 12.2C\theta_1C\theta_2C\theta_3C\theta_4 - 12.2C\theta_1S\theta_2S\theta_3C\theta_4 \\ -C\theta_1C\theta_2S\theta_3S\theta_4 & -C\theta_1C\theta_2S\theta_3S\theta_4 & & 8.9C\theta_1C\theta_2C\theta_3 - 8.9C\theta_1S\theta_2S\theta_3 + 9.7C\theta_1C\theta_2 \\ -C\theta_1S\theta_2C\theta_3S\theta_4 & -C\theta_1S\theta_2C\theta_3S\theta_4 & & \\ S\theta_1C\theta_2C\theta_3C\theta_4 & -S\theta_1C\theta_2C\theta_3S\theta_4 & & \\ -S\theta_1S\theta_2S\theta_3C\theta_4 & -S\theta_1S\theta_2S\theta_3S\theta_4 & -C\theta_1 & 12.2S\theta_1C\theta_2C\theta_3C\theta_4 - 12.2S\theta_1S\theta_2S\theta_3C\theta_4 \\ S\theta_1C\theta_2S\theta_3S\theta_4 & -C\theta_1C\theta_2S\theta_3S\theta_4 & & 12.2S\theta_1C\theta_2S\theta_3S\theta_4 - 12.2S\theta_1S\theta_2C\theta_3S\theta_4 \\ -S\theta_1S\theta_2C\theta_3S\theta_4 & -S\theta_1S\theta_2C\theta_3C\theta_4 & & 8.9S\theta_1C\theta_2C\theta_3 - 8.9S\theta_1S\theta_2S\theta_3 + 9.7S\theta_1C\theta_2 \\ 0 & 0 & 1 & 6.1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

3.1.2 Cinemática inversa

Para la cinemática Inversa existen varios métodos de solución y el geométrico es uno de ellos. Con el análisis geométrico en el plano *XYZ* se obtienen las ecuaciones del 7 al 16, considerando que el último eslabón siempre se mantendrá de forma vertical y perpendicular al eje *x* para facilitar el análisis.

$$m_1 = \sqrt{x^2 + y^2} \quad (7)$$

$$\theta_1 = \tan^{-1}\frac{y}{x} \quad (8)$$

$$m_2 = d_4 + z - d_1 \quad (9)$$

$$m_3 = \sqrt{{m_1}^2 + {m_2}^2} = \sqrt{x^2 + y^2 + (d_4 + z - d_1)^2} \quad (10)$$

$$\beta_1 = \cos^{-1}\left(\frac{{d_2}^2 + {d_3}^2 - {m_3}^2}{2d_2d_3}\right) = 180° - \theta_3 \quad (11)$$

Despejando la ecuación 11 se tiene la ecuación 12 que corresponde el ángulo de la tercera articulación.

$$\theta_3 = 180° - \cos^{-1}\left(\frac{{d_2}^2+{d_3}^2-{m_3}^2}{2d_2d_3}\right) = 180° - \cos^{-1}\left(\frac{{d_2}^2+{d_3}^2-\left(x^2+y^2+(d_4+z-d_1)^2\right)}{2d_2d_3}\right) \quad (12)$$

$$\beta_2 = \tan\frac{m_2}{m_1} = \tan^{-1}\left(\frac{d_4+z-d_1}{\sqrt{x^2+y^2}}\right) \quad (13)$$

$$\beta_3 = \cos^{-1}\left(\frac{{d_2}^2+{m_3}^2-{d_3}^2}{2d_2d_3}\right) = \cos^{-1}\left(\frac{{d_2}^2+x^2+y^2+(d_4+z-d_1)^2-{d_3}^2}{2d_2d_3}\right) \quad (14)$$

Sustituyendo las ecuaciones 13 y 14 en la ecuación 15 se tiene el ángulo de la segunda articulación:

$$\theta_2 = 180° - \beta_2 - \beta_3 = 180 - \tan^{-1}\left(\frac{d_4+z-d_1}{\sqrt{x^2+y^2}}\right) - \cos^{-1}\left(\frac{{d_2}^2+x^2+y^2+(d_4+z-d_1)^2-{d_3}^2}{2d_2d_3}\right) \quad (15)$$

La ecuación 16 fue establecida debido a la condición del último eslabón y muestra la ecuación que representa el ángulo de la cuarta articulación.

$$\theta_4 = 270° - \theta_2 - \theta_3 \quad (16)$$

Con las ecuaciones 8, 12,15 y 16 se obtiene el valor de los ángulos de cada articulación.
La cinemática inversa presenta ciertas restricciones para el robot manipulador, las cuales son:

- θ_1, θ_2, θ_3 y θ_4 , solo pueden girar en un intervalo de 0° a 180°.
- "Y" siempre se encuentra en el eje negativo.
- θ_2 siempre debe ser mayor a 90.
- θ_4 está establecido a la orientación del último eslabón mientras este se encuentre perpendicular al eje X.
- θ_3 y θ_4 no pueden ser 90 al mismo tiempo.
- Las coordenadas de desplazamiento permitidas son:
  9 <= X <= 9; 9 <= Y <= 0; 0 <= Z <= 9.

3.2 Fabricación del manipulador robótico.

Se utilizó acrílico como el material para la elaboración de los eslabones, por la facilidad de manufactura y el peso ligero que tiene. Se usaron tornillos para unir el aspa del servo al eslabón. Los tornillos que se usaron fueron de 1/4 “ para las uniones entre los eslabones y para los ejes de unión de 1/8 “ y 100 mm de largo con sus respectivos arandelas y tuercas, también se ocuparon tornillos de 3/16 “ para la unión en la base.
La parte de manufactura se realizó con la secuencia de la tabla 4 y el ensamble como se muestra en la figura 4 (diagrama de pescado):

Tabla 4. Tabla de la secuencia de manufactura.

Código	Proceso de Manufactura		Orden
	Fresado	Taladrado	
RM1			
RM2	X		2
RM3	X		3
RM4	X		4
RM5	X		5
RM6		X	1
RM7			
RM8			
RM9	X		6
RM10			
RM11			
RM12			
RM13	X		7

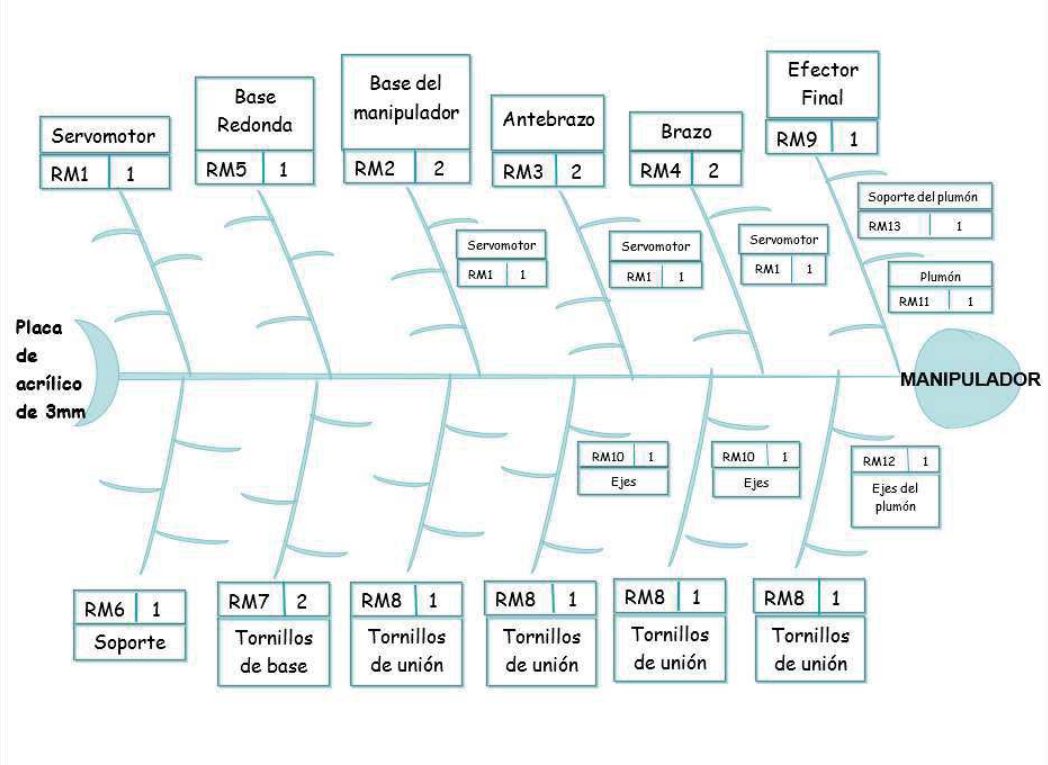


Figura 4. Diagrama de pescado del ensamble.

Para generar el movimiento rotacional de las articulaciones, se utilizaron servomotores tipo MG995 para posicionarse entre un rango de 0° a 180° a través de un modulador de ancho de pulso. Los servomotores tienen un margen de error, el cual fue aumentando con el peso, para atenuar esto, se interpolaron los ángulos, para que la posición fuera lo más exacta posible.

En la figura 5 se muestra una vista lateral del prototipo final y en la figura 6 se muestra de forma extendida

Figura 5. Ensamblaje final, vista lateral.

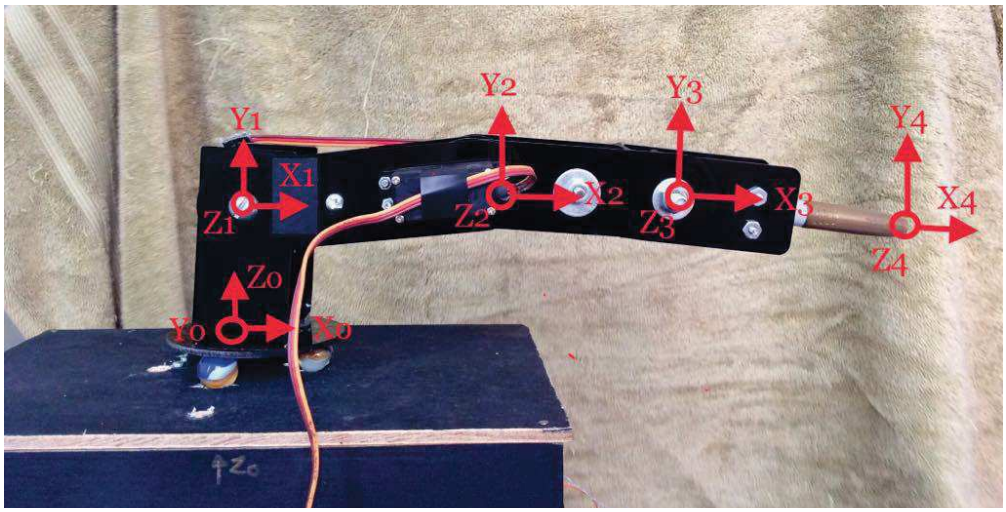


Figura 6. Ensamblaje final en vista extendida.

3.3 Sistema electrónico y de control.

Para realizar el control del ancho de pulso para los servomotores, se utilizó una tarjeta Arduino UNO® y para la alimentación de los servos se empleó una fuente de 12v a 5A. En la figura 7 se muestran las conexiones realizadas. Se utilizó PROTEUS® para generar el circuito de PCB, en la que se incluyen las entradas de voltaje de la fuente, la conexión de la tarjeta Arduino UNO® y la conexión de los servomotores. En la figura 8 se puede ver el circuito en PCB correspondiente.

Para controlar el manipulador robótico se programó el Arduino® modulando el ancho de pulso. Para el cálculo de la cinemática directa, inversa y la trayectoria. Se programó en Matlab® y con GUIDE se realizó la interfaz para la interacción entre el usuario y el manipulador, a través de la comunicación entre Matlab® y Arduino®. En la figura 9 se muestra la interfaz y el manipulador en funcionamiento.

En la figura 10 se muestra el algoritmo para controlar el manipulador, el cual inicia realizando la comunicación serial. En la interfaz se tienen 3 opciones: realizar la cinemática directa, la cinemática inversa o la trayectoria (en este caso cuadrado o circulo).
Si se desea la cinemática directa se ingresas el valor de “teta1”, “teta2”, “teta3” y “teta4” (que son los ángulos para la posición), posteriormente al activar el programa, se empiezan la interpolación de los ángulos para que el robot sea más aproximado y se envían los nuevos ángulos a través de comunicación serial a la tarjeta Arduino®. Después se calcula la matriz de transformación y muestra en la pantalla las coordenadas *X, Y* y *Z*. Si se decide salir, entonces se presiona el botón “Salir” y no se seguirá en la comunicación y el puerto se cierra. En el caso de que no se le oprima el botón se pueden realizar más opciones para el control del manipulador.
Al elegir la cinemática inversa, se ingresarán las coordenadas *X, Y* y *Z*. Posteriormente se calcula la cinemática inversa para obtener el valor de los ángulos, se interpolan y son enviados por puerto serial a la tarjeta Arduino®.
Con la opción de trayectoria se puede elegir el cuadrado o el circulo (que son las figuras predeterminadas) y el algoritmo correspondiente empezará a realizarse generando los vectores de las coordenadas en *X, Y* y *Z*, los cuales se evaluaran en las ecuaciones de la cinemática inversa, dando como resultado el cálculo de los ángulos tetas; estos se interpolarán y serán transmitidos a la tarjeta Arduino®.

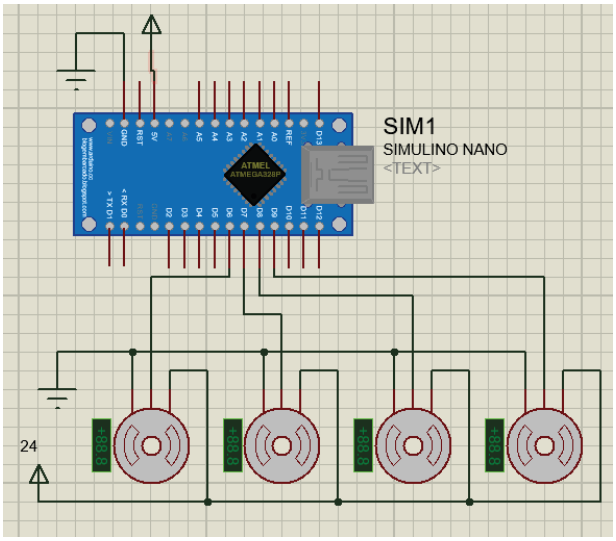


Figura 7. Diseño del circuito.

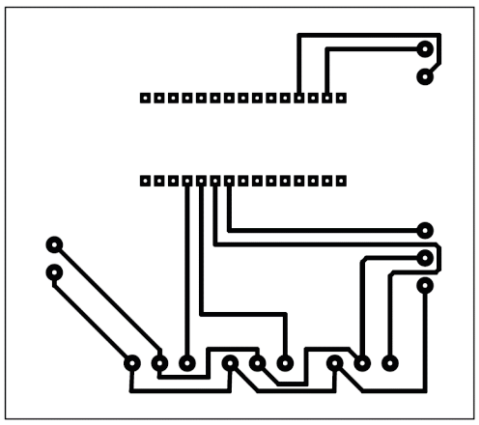


Figura 8. Diseño del circuito impreso.

Figura 9. Interfaz y el robot en funcionamiento.

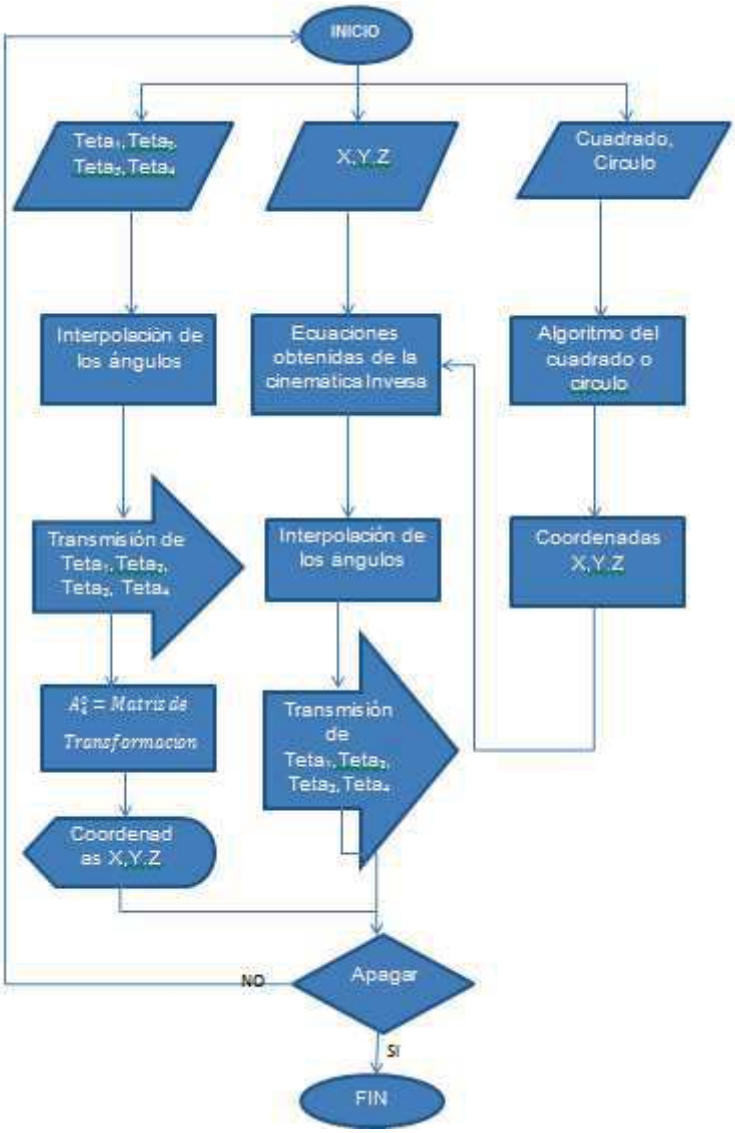


Figura 10. Diseño de flujo del control del Manipulador robótico

Cómo se puede observar en la figura 11, la interfaz puede realizar la cinemática directa e inversa, así como dibujar dos figuras geométricas (cuadrado o circulo). En la cinemática directa se ingresa el valor de los ángulos para la posición que se quieren rotar en cada articulación del manipulador. Posteriormente se activa con la instrucción “play”, mostrando en la interfaz los valores de las coordenadas. Para la cinemática inversa se ingresan los valores para las coordenadas, indicando hasta donde llegará el efector final y en la interfaz se mostraran los valores de los ángulos. En la trayectoria se tiene la opción de círculo o cuadrado; al oprimir “empezar” el robot inicia el dibujo. Al presionar “Salir” se cierra el puerto serial.

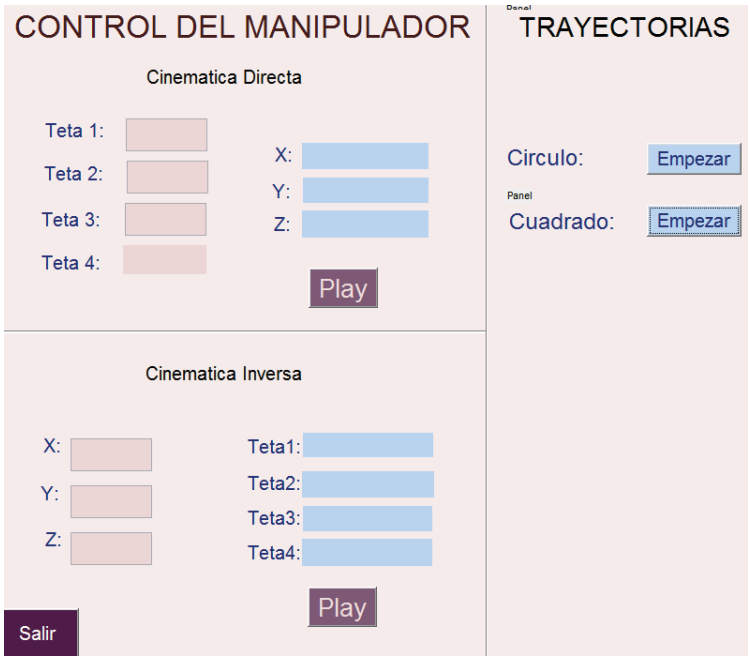


Figura 11. Interfaz del Manipulador robótico.

3.4 Generación de trayectorias

Para realizar la trayectoria primero se graficó el polígono correspondiente a cada figura geométrica en Matlab® y se obtuvo un algoritmo capaz de obtener las posiciones necesarias para trazarlo. En la figura 12 se presenta un extracto de cómo se construye el algoritmo que se utilizó para dibujar el círculo. Para ello se utilizaron dos vectores, uno para las posiciones de *X* y otro para las posiciones de *Y*. Con ciclos “*for*” se fue asignando el valor de cada posición por cada grado del círculo, una vez obtenidas todas las posiciones, se evaluaron en la cinemática inversa, manteniendo el valor de *Z=5*, dando como resultados el valor de los ángulos para cada articulación, las cuales fueron interpoladas y enviadas por puerto serial a la tarjeta Arduino®.

```
%Dibujando el primer cuadrante del
círculo del 0° a 90°
for(i=0:90)
    alfa1=i*pi/180;    %Angulo    en
radianes
    vecX(i+1)=-6+(3*cos(alfa1));
%vector    para    almacenar    las
posiciones en X
    vecY(i+1)=-6-(3*sin(alfa1));
%vector    para    almacenar    las
posiciones en Y
end
```

Figura 12. Algoritmo para realizar un círculo

Para visualizar las trayectorias, se utilizó el simulador Robotic toolbox en la figura 13 se muestra el robot durante la simulación y los resultados de las trayectorias son mostrados en la figura 14.

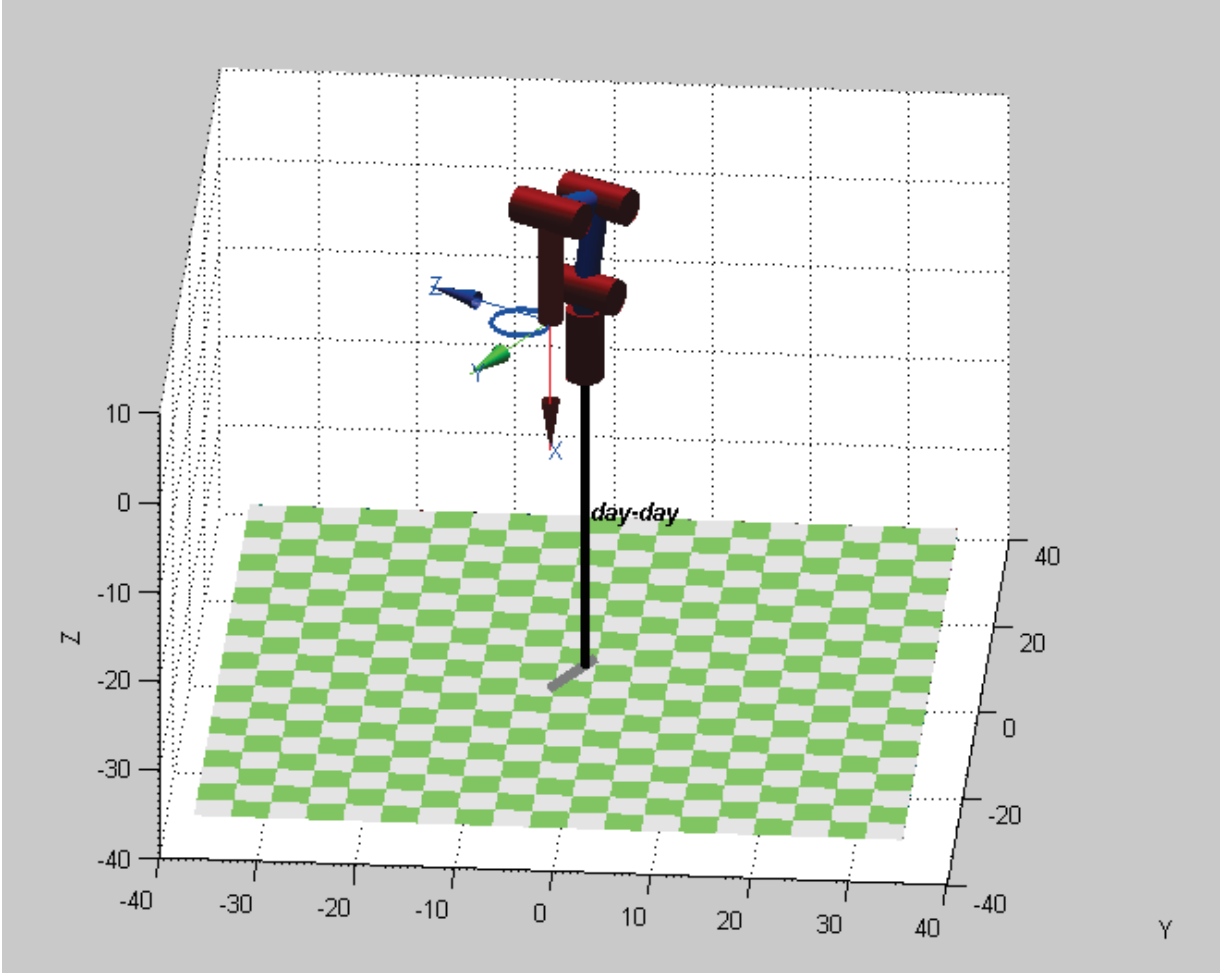


Figura 12. Simulación en Robotic Toolbox del manipulador robótico en funcionamiento

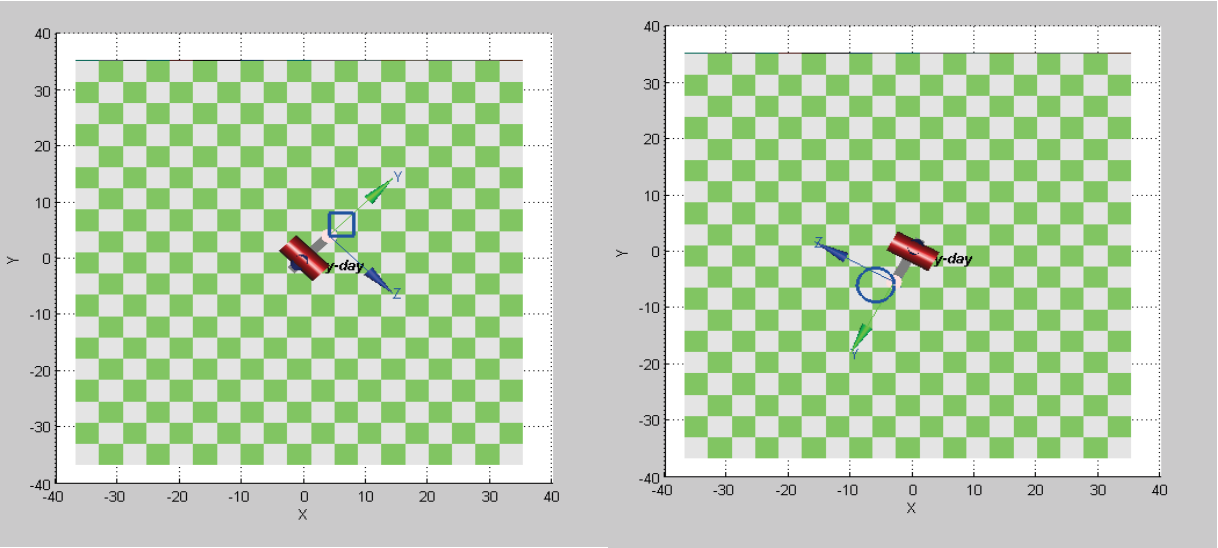


Figura 14. Trayectorias del simulador Robotic Toolbox

Utilizando estos algoritmos e implementándolos en el manipulador, en la figura 15 se puede observar como dibuja el círculo el prototipo del manipulador robótico.

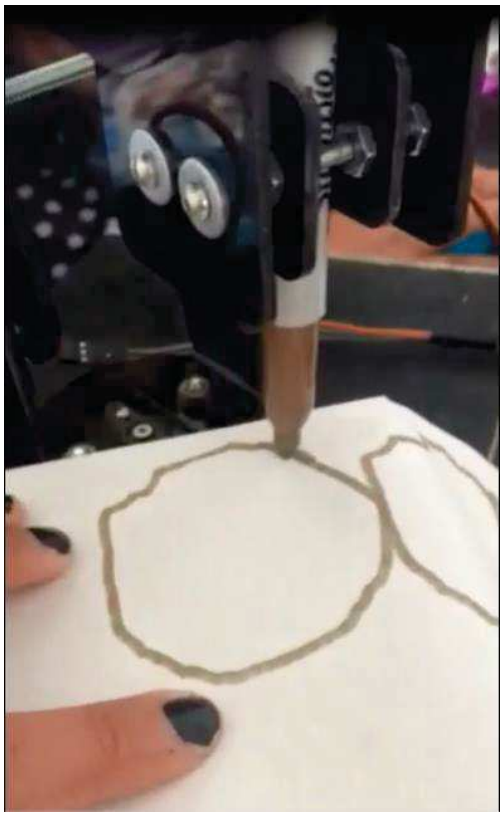


Figura 15. Trayectoria del Manipulador dibujando un círculo.

4 Resultados

En la tabla 5 se muestran los requerimientos establecidos para el diseño del manipulador robótico y se justifica como se cumplió con cada uno de ellos.

Tabla 5. Justificación de los requerimientos logrados.

Requerimientos	Logrado	Justificación
Rapidez de producción	X	La construcción y ensamblaje no tuvo dificultad y los eslabones de acrílico fueron hechos por una Fresadora CNC.
Adaptable a materiales comerciales	X	Se utilizaron servomotores MG995, los eslabones fueron hechos de acrílico, las transmisiones y ejes se solucionaron utilizando pernos y tornillos de medidas comerciales y para el control de los servos se utilizó Arduino UNO®.
Fácil manejo	X	La interfaz en GUIDE de Matlab® facilitó la comunicación entre el usuario y el robot manipulador.
Robustez	X	
Fácil Mantenimiento	X	El ensamblaje de cada componente se dio a través de uniones con tornillos y pernos, que resulta fácil desensamblar al sustituir un componente
Diseño Electrónico	X	Para que las conexiones electrónicas fueran más eficientes se realizó el circuito impreso en una placa PCB.
Diseño Mecánico	X	El diseño CAD se realizó en Solidworks® y los eslabones se cortaron con la fresadora CNC.
Control Eficiente	X	El control se realizó con comunicación serial entre el Arduino® y Matlab®. Se realizaron muchas pruebas para comprobar que la comunicación en el robot manipulador sea eficiente.
Dimensiones Optimas	X	Las dimensiones se acoplaron al peso y al torque que los motores podían dar, en este caso el manipulador quedo de un largo de 32 cm cuando está totalmente estirado y con un peso sin la caja es de 256.46 gramos y con la caja es de 507.57 gramos.

Con los datos de la tabla 5, se observa todo el proceso que se llevó acabo, tanto para el diseño, como para la manufactura y la programación, de una manera más simplificada. En la figura 15 el resultado del dibujo del manipulador es un circulo, con un pequeño margen de error, ya que los servomotores utilizados no son muy precisos (debido al costo y al acceso del material). Si se desea realizar una mejor trayectoria, se recomienda cambiar el tipo de servomotor y utilizar un control PID para manipular mejor las variables, como la ganancia y el amortiguamiento.

5 Conclusión

Se demostró que, para construir un prototipo de manera satisfactoria, es necesario seguir una metodología, ya que con esta se localizan de manera más efectiva y rápida los errores o los detalles de diseño que se pueden optimizar. Por otro lado, se validó que los métodos matemáticos utilizados para calcular la cinemática directa e inversa fueron los correctos, ya que se realizó la comprobación matemática en Matlab® y se implementaron en el prototipo simulado y en el real del manipulador robótico.

Para lograr que la posición del efector final fuera lo más exacto posible se interpolaron los grados de los servomotores, evitando que el error de fabricación y el peso del robot causara un desfase en la posición final.

Los programas empleados fueron suficientes para obtener un prototipo básico que puede ser utilizado en cualquier escuela o universidad, debido a que son los más empleados y sugeridos por los profesores para realizar este tipo de proyecto.

Referencias

[1] I. A. G. Luís Andueza, “Diseño de un manipulador robótico con tres grados de libertad para fines educativos,” *Rev. Cienc. e Ing.*, vol. 30, no. ISSN: 1316-7081, pp. 3–14, 2009.

[2] y S. N. P. . Felipe, Sánchez Cristo Andrés Alfredo, Portilla Flores Edgar Andrea, “Diseño Mecatrónico de un Robot Manipulador Móvil,” *10º Congr. Nac. Mecatrónica* ., vol. 1, pp. 157–162, 2011.

[3] R. M. J. y S. R. R. Martínez A. Gloria M., Jáquez O. Sonia A., “Diseño propio y Construcción de un Brazo Robótico de 5 GDL .,” *Rev. Ing. ELÉCTRICA, ELECTRÓNICA Y Comput.*, vol. 4, pp. 9–15, 2008.

[4] A. A. Romero, “Diseño, impresión, montaje y control de un manipulador robótico,” Universidad Carlos III de MAdrid, 2012.

[5] E. A. R. G. Eliseo de Jesús Cortés Torres and C. A. P. Cortés, “Desarrollo de un robot paralelo 5R para la impresión en 3D,” *IV Congr. Int. 12 Ing. MECATRÓNICA Y Autom. - CIIMA 2015 /*, pp. 12–22, 2015.

[6] A. S. Garrido, “Robot pintor artístico,” UNIVERSIDAD DE ALCALÁ, 2014.

[7] J. L. R. Arias and A. R. Fonseca, “Modelamiento matemático de la cinemática directa e inversa de un robot manipulador de tres grados de libertad,” *Ing. Solidar.*, vol. 8, no. 15, pp. 46–52, 2012.

[8] D. J. C. R. Zambrana, “Modelo cinemático y control de un brazo robótico imprimible,” Universidad Carlos III de Madrid, 2013.

[9] M. A. González-Palacios, “Procedimientos de diseño en mecatrónica Procedures of Design in Mechatronics,” *Ing. Investig. y Tecnol.*, vol. 1, pp. 209–222, 2011.

[10] I. Cerón-morales, “Ejemplo de aplicación de cinemática directa e inversa a un manipulador de 2 grados de libertad .,” no. September, pp. 1–3, 2016.

[11] Corke,P, *“Robot Manipulator Capability in MATLAB: A Tutorial on Using the Robotics System Toolbox [Tutorial]”.* IEEE Robotics & Automation Magazine, 24(3), 2017.

Ingeniería y Automatización, Capítulo 21, pp. 272 - 286.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Control de Velocidad de un Robot Rueda con Locomoción Inercial por Péndulo

Herrera Cordero Mario Enrique*, Arias Montiel Manuel y Lugo González Esther

Universidad Tecnológica de la Mixteca, Instituto de Electrónica y Mecatrónica, Carretera Acatlima km 2.5, Huajuapan de león, Oaxaca, México.
*herrera.enrique09@gmail.com

Resumen

.
Los robots móviles de una rueda se han convertido en una alternativa a los robots de varias ruedas debido a su tamaño compacto y flexibilidad de movimiento. Estas características dan a los robots de una sola rueda aplicaciones potenciales tales como búsqueda y rescate, exploración y transporte de cargas en espacios estrechos. En este trabajo, se desarrolla el modelo dinámico de un robot de una sola rueda con locomoción inercial por péndulo y se diseña el control de movimiento mediante la técnica del lugar geométrico de las raíces. Tanto el modelo dinámico como el desempeño del controlador son validados mediante simulaciones numéricas, y los resultados obtenidos muestran la viabilidad para su implementación práctica.

Palabras clave: Robot de una rueda, modelo dinámico, lugar geométrico de las raíces.

1. Introducción

La mayoría de los robots móviles con ruedas, usan tres o más ruedas debido a los aspectos de estabilidad. Sin embargo, en los últimos años, se han llevado a cabo múltiples esfuerzos para estabilizar los robots de dos ruedas con el fin de lograr una mayor versatilidad para operar en entornos no estructurados, complejos y dinámicos [1]. Como alternativa para reducir el tamaño y el espacio requerido para operar, han surgido nuevas configuraciones de los robots móviles, por ejemplo, ballbots [2], [3], robots esféricos [4], [5] y robots de una rueda o monociclo [6], [7], [8], [9], [10].

Dentro de la familia de los robots de una sola rueda están los robots monociclo que están equipados con una sola rueda y tienen tamaños considerablemente compactos. Además, los robots monociclo son más flexibles que otros robots con ruedas, esto es debido a que están en contacto con el suelo en un solo punto. Por lo tanto, los robots monociclo pueden alcanzar un radio de rotación muy pequeño. Por lo que, estos robots tienen aplicaciones potenciales, como el movimiento libre y el transporte de carga en espacios estrechos [8]. Sin embargo, una de las cuestiones más importantes de los robots monociclo es el problema del equilibrio lateral. De modo que, los métodos existentes de control de balance lateral se basan en ruedas de inercia horizontal [11,12], ruedas de inercia vertical [13] y giroscopios de alta velocidad [7]. Por lo tanto, el equilibrar la estabilidad de manera autónoma es relativamente difícil. En las investigaciones recientes sobre los robots monociclo, los tres métodos se aplican en los prototipos para resolver el problema del equilibrio lateral.

Lee et al. [6], [7], [8] han trabajado sobre la dinámica y el control de diferentes configuraciones de robots monociclo. En [6], propusieron un modelo dinámico desacoplado e implementaron dos leyes de control independientes para los ejes *pitch* y *roll*. La estrategia de control se prueba experimentalmente solo para una trayectoria en línea recta. En [7], los autores presentan un péndulo estabilizado giroscópicamente (GSP) como una alternativa al modelo del robot rueda con péndulo de reacción convencional en donde el control de equilibrio es lateral, es decir, hay otro péndulo en el

robot rueda. Los problemas relacionados con las técnicas de control de GSP incluyen incertidumbres paramétricas, acoplamientos no lineales entre el péndulo y el giro-rotor y vibraciones mecánicas causadas por el giro de alta velocidad del giro-rotor, de modo que estos esquemas complican las topologías de los algoritmos de control y por lo tanto, son computacionalmente complejos. En [8] se presenta un nuevo algoritmo de control basado en la compensación de fuerza centrífuga para equilibrar robots monociclo, lo cual es presentado con el fin de mejorar la robustez del sistema. Mediante la simulación y experimentos con el prototipo del mecanismo, se verificó la validez de la estructura para el control del equilibrio y la robustez de los algoritmos de control propuestos. Sin embargo, la estructura del robot se modificó añadiendo elementos mecánicos, por lo tanto, el robot es menos compacto que las versiones anteriores.

Otros autores han desarrollado investigaciones sobre otras configuraciones de robots de una sola rueda, Park y Jung [9] propusieron un enfoque mecatrónico práctico para el desarrollo de un robot de una rueda llamado GYROBO. Este enfoque permite resolver algunos problemas sobre la integración del sistema e implementar controladores lineales para ambos ángulos de balanceo y guiñada. En [10], Lee y Jung analizan la estabilidad de modo vertical del GYROBO y prueban el rendimiento del control de equilibrio de GYROBO empíricamente para verificar la comparación entre el ancho de banda de estabilidad calculado y el rango de desplazamiento determinado. Ellos aplican control solo para el equilibrio o GYROBO.

En este trabajo, se presentan el diseño, el modelo dinámico acoplado y el control de velocidad de una novedosa configuración de robot rueda con locomoción inercial por péndulo. La transmisión de movimiento se basa en engranajes planetarios que resulta en una configuración compacta y mecánicamente simple, por lo que es viable obtener las ecuaciones dinámicas del robot mediante el método de Euler-Lagrange, así como también aplicar una técnica para el diseño del control mediante el lugar geométrico de las raíces (LGR). El modelo matemático del robot y el desempeño del controlador se validan mediante simulaciones numéricas. Los resultados obtenidos muestran la viabilidad para la implementación del sistema propuesto de manera experimental.

2. Diseño del robot rueda

El diseño del robot se propone basado en el principio BCO (Bary Center Offset), que consiste en modificar el centro de masa del robot para producir el movimiento deseado [14]. Este cambio en la distribución de masa se logra mediante el movimiento de un péndulo acoplado a un micromotor de CD y dos mecanismos de piñón y corona interna, ambas coronas están fijas a dos discos que sirven como base al péndulo inercial como se aprecia en la Figura 1. Una descripción más detallada del diseño y el principio de funcionamiento del robot rueda se presentan en [15].

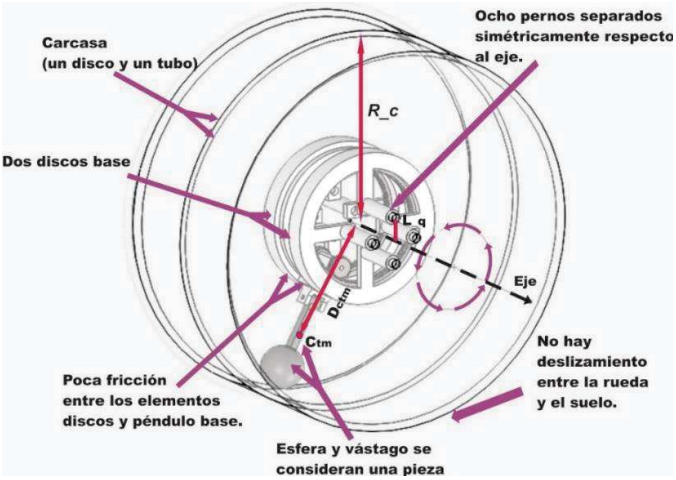


Figura 1. Prototipo del robot rueda

Ingeniería y Automatización, Capítulo 21, pp. 272 - 286.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Modelo dinámico del robot rueda

La obtención del modelo dinámico del robot rueda se realiza con base en los elementos estructurales mostrados en la Figura 1 y bajo las siguientes consideraciones:

1. No hay deslizamiento entre la carcasa del robot rueda y el suelo.
2. El péndulo está formado por un vástago y una esfera, y estos elementos son considerados como una sola pieza.
3. Se consideran 8 pernos distribuidos simétricamente con respecto al eje de rotación.
4. La fricción entre los discos base y el péndulo es despreciable.

Dadas las consideraciones anteriores, es posible realizar el diagrama de cuerpo libre del sistema, como se muestra en la Figura 2.

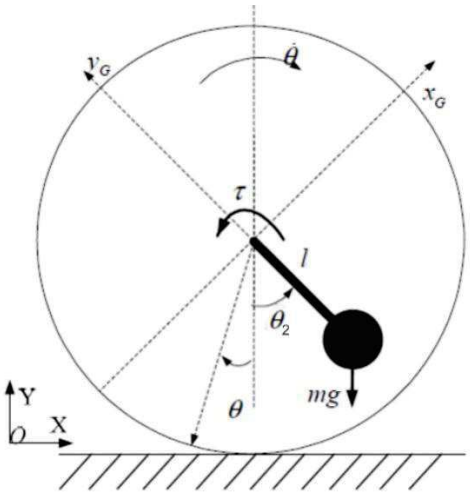


Figura 2. Diagrama de cuerpo libre.

donde:

θ = Desplazamiento angular del robot.
τ = Par torsional de entrada proporcionado por el motor.
$\theta_2 = \theta - R * \alpha$ = Ángulo de elevación del péndulo.
α = Desplazamiento angular del piñón.
R = Razón de transmisión del piñón y la corona.

El robot se desplazará de manera lineal a lo largo del eje X con una velocidad $\dot{X}$ dada por

$$\dot{X} = R_c\dot{\theta} \tag{1}$$

donde R_c es el radio de la carcasa y $\dot{\theta}$ es la velocidad angular de la carcasa del robot.

Para la obtención del modelo dinámico del robot, es necesario realizar el análisis tanto de energía cinética y potencial del robot, el cual se propone de la siguiente manera:

1. Analizar las energías de los elementos relacionados solo con θ.
2. Analizar las energías de los elementos relacionados con α y θ.

3.1 Elementos relacionados con theta

Los elementos que están relacionados con θ son la carcasa, los dos discos base, los ocho pernos y las dos coronas (ver Figura 1), de modo que sus energías tanto cinética y potencial están dadas por las siguientes expresiones:

$$T_1 = \frac{1}{2}\left(m_c R_c^2 + m_d R_c^2 + m_{cor} R_c^2 + I_c + I_{cor} + 8\left(m_{per}\left(L_q^2 + R_c^2\right) + I_{per}\right)\right)\dot{\theta}^2 \quad (2)$$

$$V_1 = m_c g R_c + m_d g(R_c - R_d) + m_{cor} g(R_c - R_{cor}) + 8 m_{per} g L_q \quad (3)$$

donde m_c es la masa de la carcasa, m_d es la masa de los dos discos base, m_{cor} es la masa de una corona, m_{per} es la masa de un perno, e I_c, I_d, I_{cor} y I_{per} denotan las respectivas inercias de los mismos elementos.

3.2 Elementos relacionados con α y θ

Los elementos que dependen de las variables α y θ son el péndulo y su base, que conforman el péndulo compuesto con centro de masa en C_{tm}, dado por las coordenadas X_{ctm} y Y_{ctm} (ver Figura 3). De modo que las energías cinética y potencial de estos elementos se expresan como

$$T_2 = \frac{1}{2} m_{comp} R_c \dot{\theta}^2 + m_{comp} \dot{X} D_{ctm} \cos(\theta_2)\,\dot{\theta}_2 + \frac{1}{2} m_{comp} D_{ctm}^2 \dot{\theta}_2^2 + \frac{1}{2} I_{ctm} \dot{\theta}_2^2 \quad (4)$$

$$V_2 = m_{comp} g D_{ctm}(1 - \cos(\theta_2)) \quad (5)$$

donde m_{comp} es masa del péndulo compuesto, D_{ctm} es la distancia del eje de rotación al centro de masa del elemento compuesto, *X* es el desplazamiento lineal de la rueda y θ_2 es el ángulo del péndulo con respecto al eje vertical.

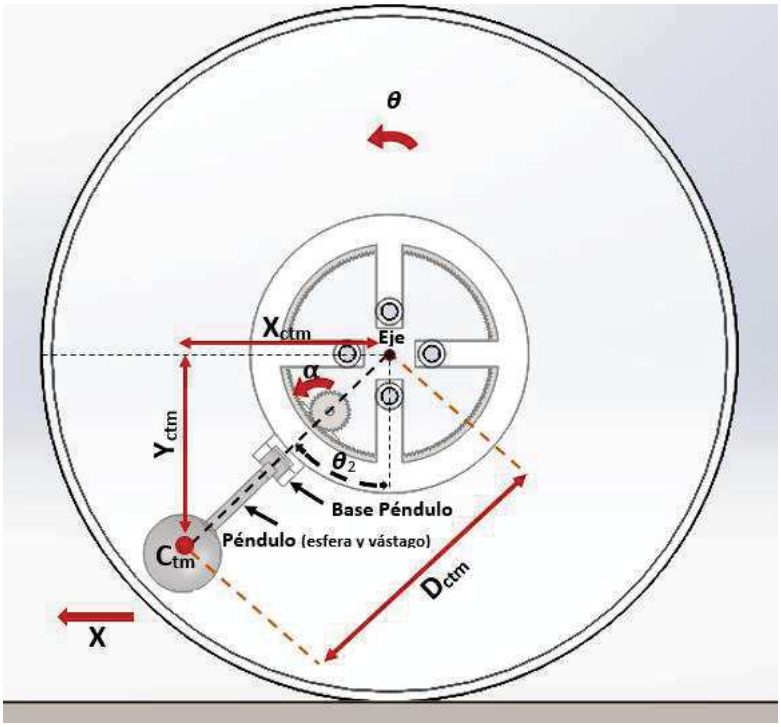


Figura 3. Ubicación de centro de masa del péndulo compuesto.

3.3 Modelo matemático del robot

Por tanto, las energías cinética y potencial totales del sistema de obtienen mediante la suma de las ecuaciones (2) y (4) y (3) y (5), respectivamente, obteniéndose

$$T_{total} = \frac{1}{2}\left(m_c R_c^2 + m_d R_c^2 + m_{cor} R_c^2 + I_c + I_{cor} + 8\left(m_{per}\left(L_q^2 + R_c^2\right) + I_{per}\right) + m_{comp} R_c\right)\dot{\theta}^2 + m_{comp} R_c D_{ctm} \cos(\theta_2)\dot{\theta}\,\dot{\theta}_2 + \frac{1}{2} m_{comp} D_{ctm}^2 \dot{\theta}_2^2 + \frac{1}{2} I_{ctm} \dot{\theta}_2^2 \quad (6)$$

$$V_{total} = m_c g R_c + m_d g(R_c - R_d) + m_{cor} g(R_c - R_{cor}) + 8 m_{per} g L_q + m_{comp} g D_{ctm}(1 - \cos(\theta_2)) \quad (7)$$

Mediante las ecuaciones (6) y (7) y aplicando la metodología de Euler – Lagrange se obtiene el modelo matemático del sistema dado por las siguientes expresiones

$$\ddot{\theta} = \left[m_{comp} D_{ctm} R_c \sin(b\alpha - \theta)\left(b\dot{\alpha} + \dot{\theta}\right)^2 + m_{comp} g D_{ctm} \sin(b\alpha - \theta) - \left[m_{comp} D_{ctm} R_c \cos(b\alpha - \theta) - I_{comp} - m_{comp} D_{ctm}^2\right] b\ddot{\alpha} - c\dot{\theta}\right]\frac{1}{W} \quad (8)$$

$$\ddot{\alpha} = \left[\frac{I_{parTorsional}}{b} - \left[m_{comp} D_{ctm} R_c \cos(b * \alpha - \theta) - I_p - m_{comp} D_{ctm}^2\right]\ddot{\theta} - m_{comp} g D_{ctm} \sin(b * \alpha - \theta)\right]\frac{1}{\left[I_{comp} + m_{comp} D_{ctm}^2\right] b} \quad (9)$$

donde $b = R$ y

$W = \left(m_c R_c^2 + m_d R_c^2 + m_{cor} R_c^2 + I_c + I_{cor} + 8\left(m_{per}\left(L_q^2 + R_c^2\right) + I_{per}\right) m_{comp} R_c\right) - 2 m_{comp} D_{ctm} R_c \cos(b\alpha - \theta) + I_{comp} + m_{comp} D_{ctm}^2$.

Con las ecuaciones (8) y (9) definidas, es posible llevar a cabo simulaciones numéricas que permitan conocer el comportamiento del robot, para lo cual es necesario definir los valores de las propiedades físicas de cada elemento como se muestra en la Tabla 1.

Tabla 1. Propiedades físicas de los elementos

Propiedad física	Masa (gr)	Inercia (gr*cm^2)	Radio (cm)
Carcasa	583.44	73884.32	13.5
Discos base c/u	70.42	1548.73	-
Perno c/u	1.87	0.35	1.64
Péndulo compuesto c/u	122.36	586.611	10.3441
Piñón c/u	-	-	0.75
Corona c/u	4.199	7112.644	3.975

Considerando una entrada de par torsional constante de 2N*mm, se resolvieron numéricamente las ecuaciones (8) y (9) y los resultados se compararon con simulaciones realizadas en SolidWorks.

Estas comparaciones se muestran en las Figuras 4 y 5 para el desplazamiento lineal del centro de masa y la velocidad angular del robot rueda.

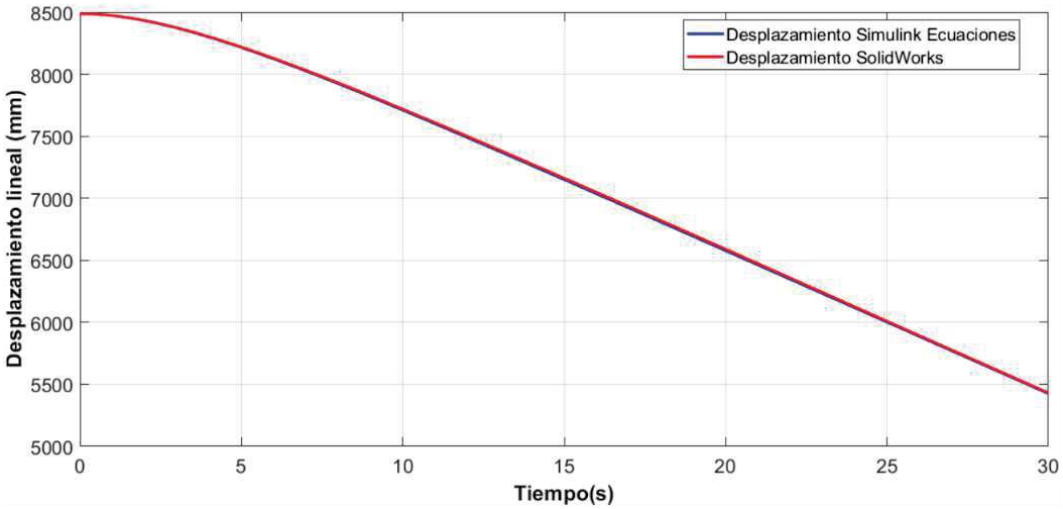


Figura 4. Comparación entre SolidWorks y Matlab del desplazamiento del robot.

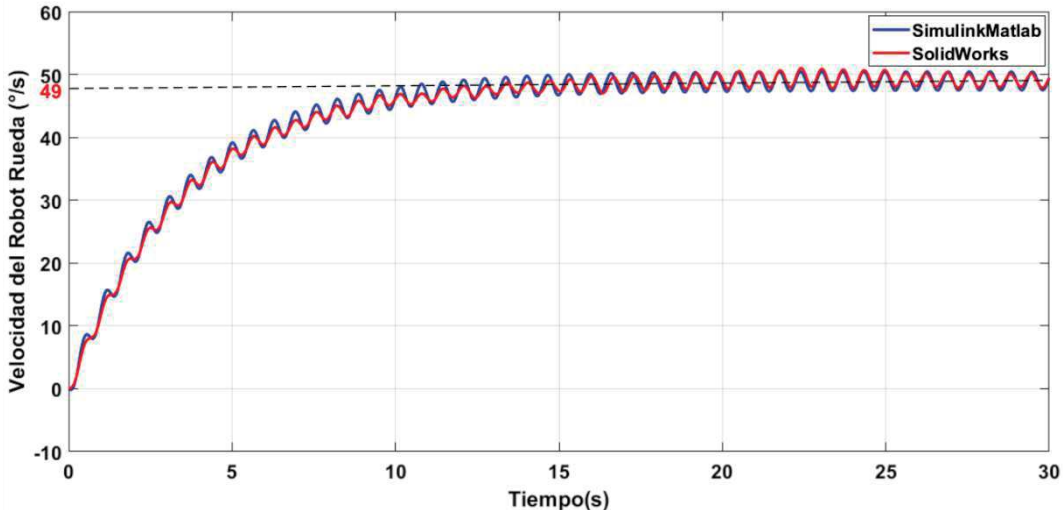


Figura 5. Comparación entre SolidWorks y Matlab de la velocidad angular del robot.

De los resultados obtenidos, se puede observar que las respuestas son prácticamente idénticas, por lo que se puede concluir que las ecuaciones obtenidas describen la dinámica del robot de forma precisa. Sin embargo, es importante señalar que la respuesta de la planta en cuanto a velocidad es sumamente lenta ante una entrada de par torsional constante, por lo que es necesario hacer que el sistema sea más óptimo, es decir que adquiera la velocidad deseada en el menor tiempo posible, de modo que es necesario implementar una ley de control que permita dar solución a este problema.

4. Linealización, función de transferencia y control de velocidad.

Partiendo de las ecuaciones (8) y (9), es posible realizar la linealización del modelo del sistema alrededor de un punto de operación, en este caso, cuando $\theta_2 = 0$ o $\theta = b * \alpha$, es decir el péndulo del robot rueda se encuentra verticalmente hacia abajo. Se cumple además que la entrada de equilibrio es

0 (el sistema se mantiene con velocidad constante), y puesto que la velocidad es constante, $\ddot{\theta} = 0$, $\ddot{\alpha} = 0$, por lo tanto, se definen las variables de estado qcomo $x_1 = \theta_{eq}$, $x_2 = \dot{\theta}_{eq}$, $x_3 = \alpha_{eq}$, $x_4 = \dot{\alpha}_{eq}$ y $U_{eq} = 0$.

Una vez que se tienen las condiciones de equilibrio, es posible linealizar mediante la expansión en series de Taylor alrededor del punto de equilibrio o punto de operación $\theta_2 = 0$, una forma de visualizar tal expansión es mediante las siguientes expresiones

$$\begin{bmatrix} \dot{x}_{1\delta} \\ \dot{x}_{2\delta} \\ \dot{x}_{3\delta} \\ \dot{x}_{4\delta} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{f_1(\alpha,\theta)}{\partial x_1} & \frac{f_1(\alpha,\theta)}{\partial x_2} & \frac{f_1(\alpha,\theta)}{\partial x_3} & \frac{f_1(\alpha,\theta)}{\partial x_4} \\ \frac{f_2(\alpha,\theta)}{\partial x_1} & \frac{f_2(\alpha,\theta)}{\partial x_2} & \frac{f_2(\alpha,\theta)}{\partial x_3} & \frac{f_2(\alpha,\theta)}{\partial x_4} \\ \frac{f_3(\alpha,\theta)}{\partial x_1} & \frac{f_3(\alpha,\theta)}{\partial x_2} & \frac{f_3(\alpha,\theta)}{\partial x_3} & \frac{f_3(\alpha,\theta)}{\partial x_4} \\ \frac{f_4(\alpha,\theta)}{\partial x_1} & \frac{f_4(\alpha,\theta)}{\partial x_2} & \frac{f_4(\alpha,\theta)}{\partial x_3} & \frac{f_4(\alpha,\theta)}{\partial x_4} \end{bmatrix}}_{\textbf{Matriz A}} \begin{bmatrix} x_{1\delta} \\ x_{2\delta} \\ x_{3\delta} \\ x_{4\delta} \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{f_1(\alpha,\theta)}{\partial I_{n2}} \\ \frac{f_2(\alpha,\theta)}{\partial I_{n2}} \\ \frac{f_3(\alpha,\theta)}{\partial I_{n2}} \\ \frac{f_4(\alpha,\theta)}{\partial I_{n2}} \end{bmatrix}}_{\textbf{Matriz B}} [I_{n2\delta}] \tag{10}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{x_1}{\partial x_1} & \frac{x_1}{\partial x_2} & \frac{x_1}{\partial x_3} & \frac{x_1}{\partial x_4} \\ \frac{x_2}{\partial x_1} & \frac{x_2}{\partial x_2} & \frac{x_2}{\partial x_3} & \frac{x_2}{\partial x_4} \\ \frac{x_3}{\partial x_1} & \frac{x_3}{\partial x_2} & \frac{x_3}{\partial x_3} & \frac{x_3}{\partial x_4} \\ \frac{x_4}{\partial x_1} & \frac{x_4}{\partial x_2} & \frac{x_4}{\partial x_3} & \frac{x_4}{\partial x_4} \end{bmatrix}}_{\textbf{Matriz C}} \begin{bmatrix} x_{1\delta} \\ x_{2\delta} \\ x_{3\delta} \\ x_{4\delta} \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{x_1}{\partial I_{n2}} \\ \frac{x_2}{\partial I_{n2}} \\ \frac{x_3}{\partial I_{n2}} \\ \frac{x_4}{\partial I_{n2}} \end{bmatrix}}_{\textbf{Matriz D}} [I_{n2\delta}] \tag{11}$$

Donde cada $f_n(\alpha, \theta)$ representa cada variable de estado independiente de otra variable de estado de mayor o igual orden, siendo esto así, se procede a aplicar las ecuaciones (10) y (11) con las consideraciones del punto de operación de las funciones $f_1 = x_2$, $f_2 = \dot{x}_2$, $f_3 = x_4$ y $f_4 = \dot{x}_4$, dando como resultado las ecuaciones matriciales (12) y (13).

$$\begin{bmatrix} \dot{x}_{1\delta} \\ \dot{x}_{2\delta} \\ \dot{x}_{3\delta} \\ \dot{x}_{4\delta} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ -3.7951 & -0.3034 & 0.7161 & 0 \\ 0 & 0 & 0 & 1 \\ 486.0901 & 0.4006 & -91.7151 & 0 \end{bmatrix}}_{\textbf{Matriz A}} \begin{bmatrix} x_{1\delta} \\ x_{2\delta} \\ x_{3\delta} \\ x_{4\delta} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ -32.3079 \\ 0 \\ 2.0577e+04 \end{bmatrix}}_{\textbf{Matriz B}} [T_{in}] \tag{12}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\textbf{Matriz C}} \begin{bmatrix} x_{1\delta} \\ x_{2\delta} \\ x_{3\delta} \\ x_{4\delta} \end{bmatrix} + \underbrace{[0]}_{\textbf{Matriz D}} [T_{in}] \tag{13}$$

Una vez obtenidos los resultados anteriores, se pueden identificar las matrices de estados que definen al sistema linealizado alrededor del punto de operación ($\theta_2 = 0$), siendo esto así, es posible simular las ecuaciones matriciales en Matlab Simulink, obteniendo los resultados mostrados en la Figura 6.

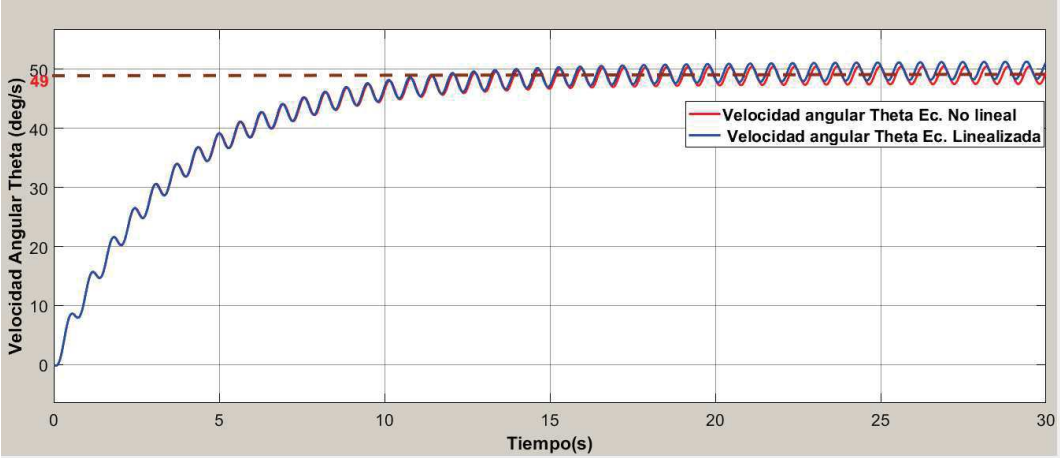


Figura 6. Comparación de las respuestas entre las ecuaciones no lineales y lineales.

Como se puede apreciar por la Figura 6, por comparación de la respuesta con el sistema no lineal original, el sistema fue correctamente linealizado, de modo que, con base en las matrices de estados obtenidas se puede obtener la función de transferencia dada por

$$G(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + D \tag{14}$$

Al aplicar la ecuación (14), se obtienen las funciones de transferencia que propiamente dan como respuesta las salidas del sistema en función a la entrada de par torsional.

$$\mathrm{G(s)} = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix} = \begin{bmatrix} \frac{-32.3079s^2+1.1772*10^4}{s^4+0.3034s^3+95.5102s^2+27.5410s} \\ \frac{-32.3079s^2+1.1772*10^4}{s^3+0.3034s^2+95.5102s+27.5410} \\ \frac{2.0577*10^4s^2+6.2302*10^3s+62387}{s^4+0.3034s^3+95.5102s^2+27.5410s} \\ \frac{2.0577*10^4s^2+6.2302*10^3s+62387}{s^3+0.3034s^2+95.5102s+27.5410} \end{bmatrix} \tag{15}$$

De la ecuación (15), se tiene que la función de transferencia Y_2 es la función que relaciona la entrada de par torsional con la velocidad del sistema

$$G(s) = \frac{-32.3079s^2+1.1772*10^4}{s^3+0.3034s^2+95.5102s+27.5410} \tag{16}$$

Se hará uso de ella para aplicar control a la velocidad del robot, de modo que, es de importancia validar dicha función de transferencia mediante resultados numéricos de simulación en donde se comparará la respuesta de las ecuaciones no lineales con la función de transferencia. Dicha comparación se puede apreciar en la Figura 7, en donde se puede observar que la respuesta obtenida

mediante la función de transferencia sigue prácticamente sin error a la respuesta del sistema no lineal, de modo que se puede diseñar el control de velocidad del robot mediante LGR a partir de la función de transferencia (16).

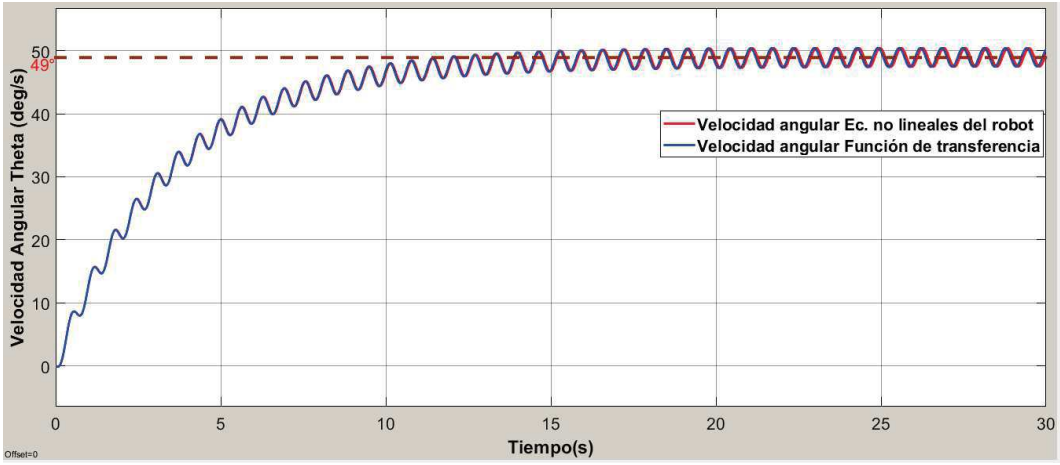


Figura 7. Comparación de las respuestas de velocidad angular entre la función de transferencia y las ecuaciones no lineales.

Para diseñar el controlador adecuado para el robot, es necesario analizar la ubicación de polos y ceros del sistema tanto en lazo abierto como en lazo cerrado, es decir, observar si el sistema es estable, críticamente estable o inestable, así como también, saber si es amortiguado o subamortiguado. Por lo que la distribución de los polos y ceros del sistema en lazo abierto se muestra en la Figura 8.

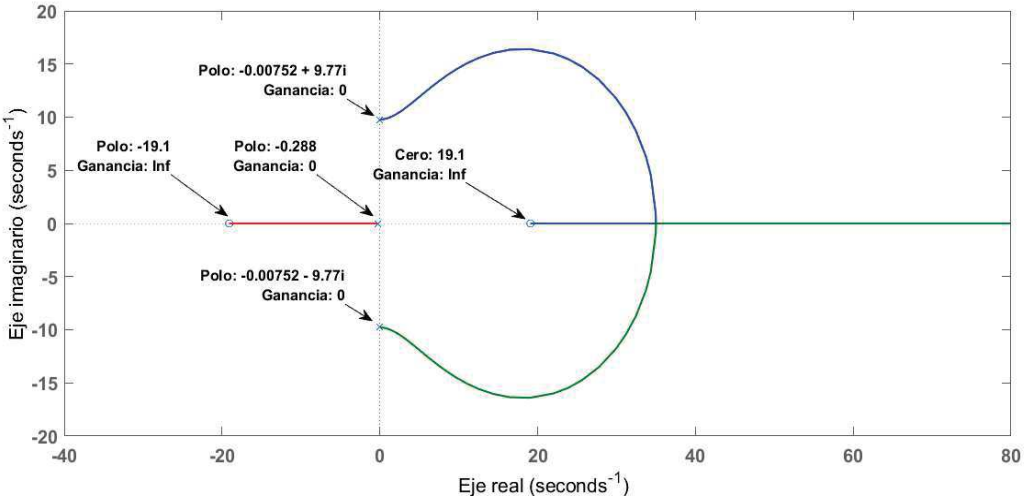


Figura 8. Lugar geométrico de las raíces en L.A

Como se puede apreciar en la Figura 8, el sistema es estable en presencia de todos los polos en el semiplano izquierdo, en donde además se tiene que el par de polos dominantes están en $S = -0.00752 \pm 9.77i$, sin embargo, para el control es necesario cerrar el lazo, es decir, tener el sistema en lazo cerrado (LC), en donde se compara la salida con una entrada de referencia, como se muestra en el diagrama a bloques de la Figura 9 y se representa en la ecuación (17).

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

$$= \frac{-32.31s^2 + 11772}{s^3 - 32s^2 + 95.51s\ 11800} \tag{17}$$

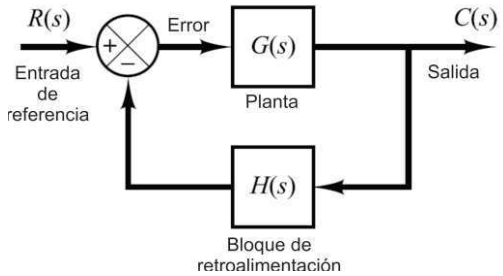


Figura 9. Diagrama a bloques de la planta en lazo cerrado.

Al tener la función de transferencia en lazo cerrado, es posible trazar el lugar geométrico de las raíces del sistema como se muestra en la Figura 10, dicha gráfica revela que hay un par de polos conjugados en le semiplano derecho en $s = 23.4396799 \pm 15.60853\text{i}$ y un polo en el semiplano izquierdo en $s = -14.8793599$, de modo que, el sistema es inestable, aunque haya un ajuste la ganancia, el sistema seguirá siendo inestable.

Como los polos conjugados son los que dominan al sistema, es necesario saber sus propiedades como el factor de amortiguamiento y la frecuencia de oscilación, las cuales son 0.83234 y 28.1605 rad/s, respectivamente.

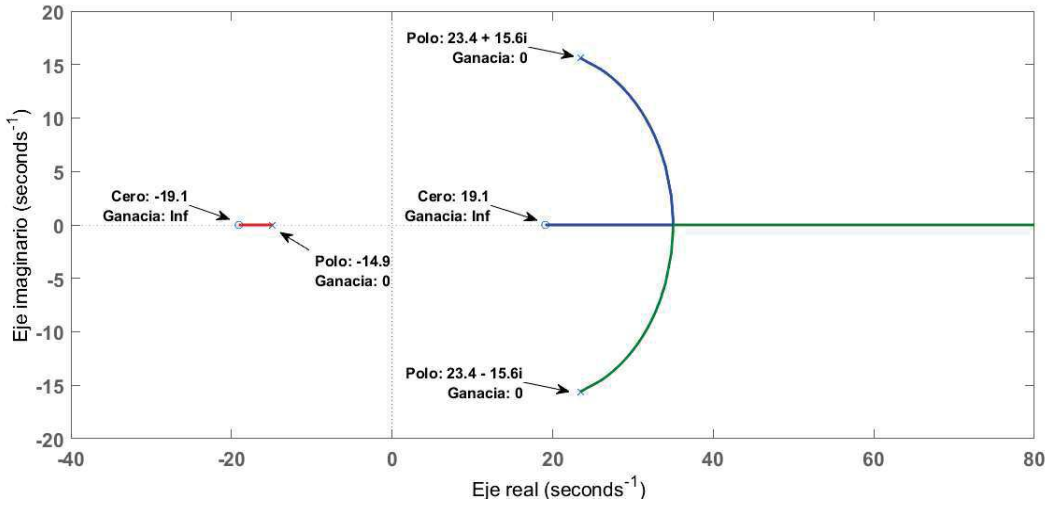


Figura 10. Lugar geométrico de las raíces en L.C

Considerando lo anterior, es posible aplicar la técnica de control del lugar geométrico de las raíces, en donde se imponen las condiciones del sistema como el coeficiente de amortiguamiento $\zeta = 0.5$ y la frecuencia de oscilación $\omega_n = 1rad/s$, por lo que, los polos dominantes están definidos por $s_{1,2} = \zeta\omega_n \pm \omega_n\sqrt{1-\zeta^2}i$, siendo esto así, los polos deseados son $s_{1,2} = -0.5 \pm 0.866\text{i}$. Ahora bien, una vez que se conocen los polos dominantes es de importancia ver la manera en la que se agregarán, es por ello que se hará un análisis del LGR tanto en lazo abierto como en lazo cerrado.

En el caso del LGR en lazo abierto (ver Figura 8), se tiene que los polos dominantes están en el semiplano izquierdo, es decir la planta es estable, sin embargo, cuando se cierra el lazo, el LGR (ver Figura 10) indica que la planta se vuelve inestable, y eso es debido a que los polos dominantes pasan al semiplano derecho, en donde un polo dominante busca al cero del semiplano derecho por lo que se cancela con él, y el otro polo dominante se va al infinito. Siendo esto así, se tiene un criterio de la causa de la inestabilidad, esto prácticamente proviene de los polos conjugados en el semiplano derecho y el cero ($s = -19.088454$) en el semiplano izquierdo, ya que cancela el único polo presente.

Por lo expuesto anteriormente, se tiene que es de importancia cancelar ese par de polos conjugados y el cero de la planta en lazo cerrado, es decir agregar un compensador directo (G(s)supr) con la planta, por lo que la función de transferencia queda dada por

$$G(s) * G(s)_{supr} = \frac{-32.3079s^2 + 1.1772 * 10^4}{s^3 + 0.3034s^2 + 95.5102s + 27.541} * \left(\frac{s^2 + 0.01503027s + 95.49204}{(s + 19.088454)(s)}\right) = \frac{-32.3079(s-19.088454)}{(s+0.288)(s)} \quad (18)$$

Ahora bien, es importante notar que la planta queda simplificada, sin embargo, se tiene que compensar el cambio en la función de transferencia al agregar un compensador de adelanto que cumpla con los requerimientos del funcionamiento de la planta, de modo que el compensador tiene que compensar el ángulo de deficiencia dado por los polos conjugados deseados en lazo cedrrado en la planta en lazo abierto. La ecuación (19) muestra el ángulo a compensar.

$$Ángulo_{comp} = 180° + AngCeros_{s=-0.5+0.866i} - AngPolos_{s=-0.5+0.866i} = 180 - 2.53138° - 103.755° - 120° = -46.28638° \quad (19)$$

Conociendo el ángulo a compensar, se puede proceder mediante el método grafico a calcular el compensador de adelanto, como se muestra en la Figura 11, en donde el polo está dado en -1.65520103 y el cero en -0.604129, siendo esto así, la función de transferencia del sistema compensado en lazo cerrado queda dado por la ecuación (20).

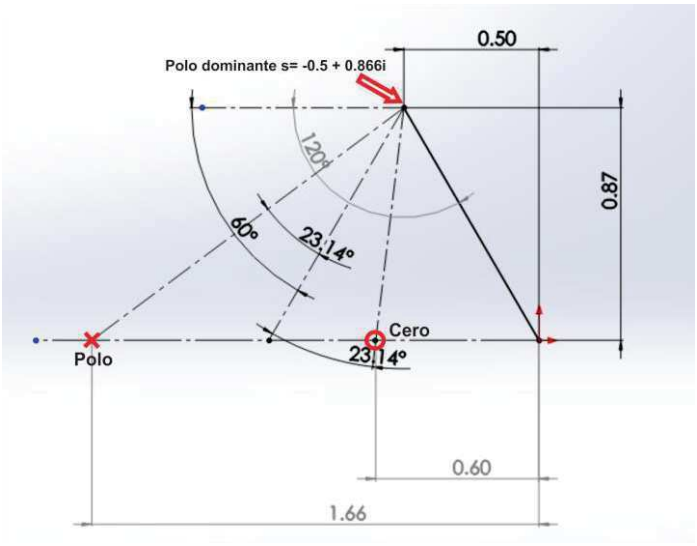


Figura 11. Ubicación de polos y ceros.

$$G_T = G(s)_s * G(s)_{comp} = \frac{-32.3079(s-19.088454)}{(s+0.288)(s)} * \frac{K(s+0.604129)}{(s+1.65520103)} \quad (20)$$

Por lo que solo falta calcular la ganancia K del sistema para que tenga los polos dominantes en el lugar deseado, siendo así, se debe de cumplir la condición de magnitud y ángulo, en donde la magnitud debe ser 1 y el ángulo de 180° en el polo deseado. La condición de magnitud se puede encontrar al sustituir el polo dominante deseado $s_1 = -0.5 + 0.866\mathrm{i}$ en la función de transferencia dada por la ecuación (20) e igualar a -1, por lo que la ganancia requerida es $K = 2.32967 * 10^{-3}$, así que la función de transferencia queda denotada por

$$G_T = \frac{-32.3079(s-19.088454)}{(s+0.288)(s)} * \frac{2.32967*10^{-3}(s+0.604129)}{(s+1.65520103)} \quad (21)$$

Para probar si el diseño cumple con las especificaciones, se procede a cerrar el lazo (ver Figura 12) y se grafica su lugar geométrico de las raíces, como se muestra en la Figura 13.

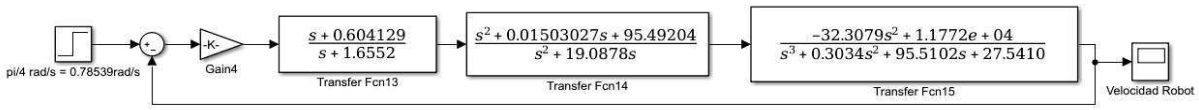
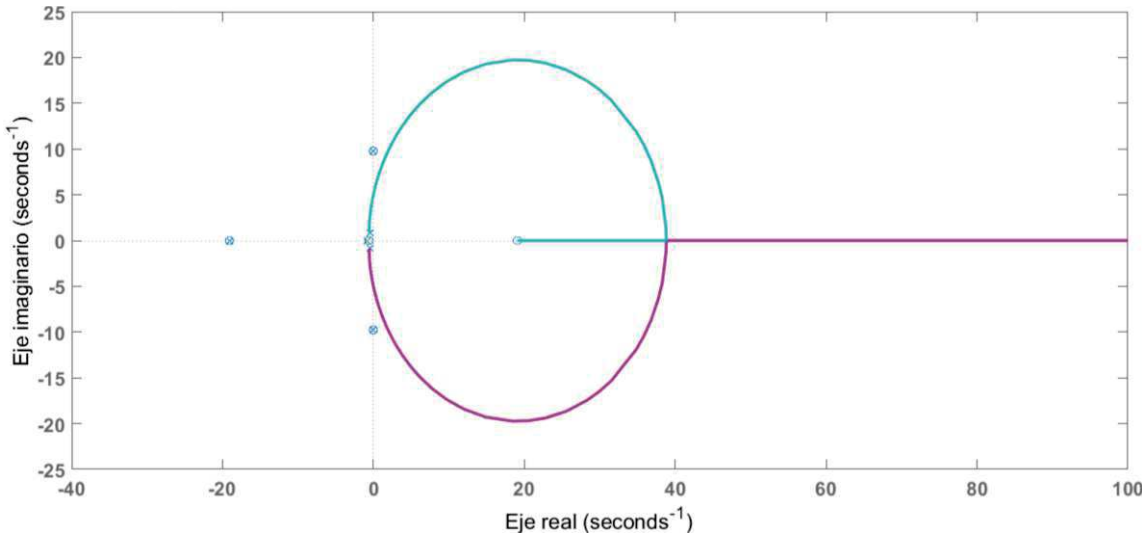
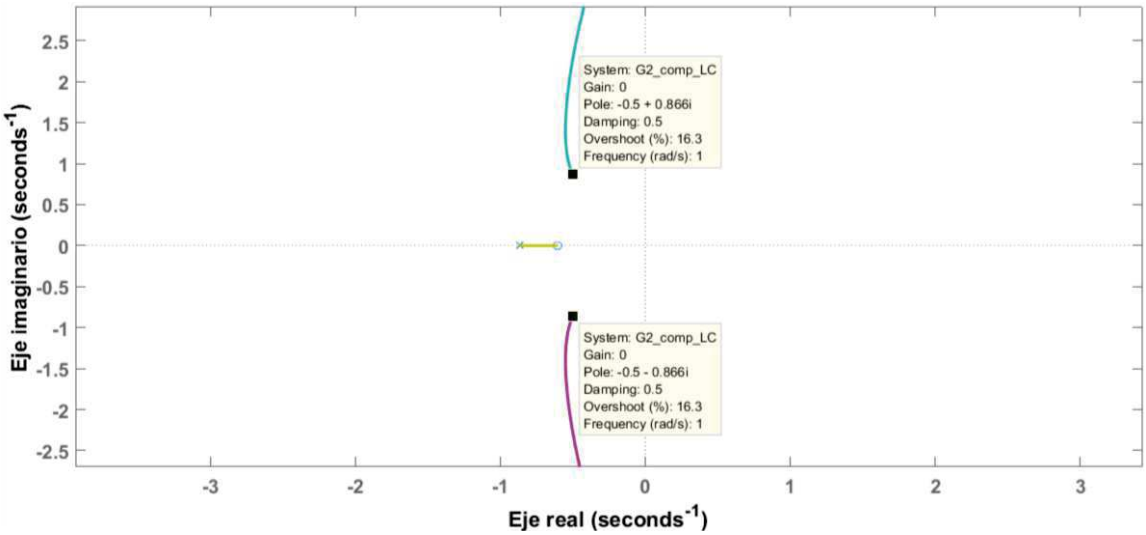


Figura 12. Diagrama a bloques del sistema en lazo cerrado compensado.



A)



B)

Figura 13. Lugar geométrico de las raíces en L.A de la planta compensada.

Ahora bien, como se puede apreciar en la Figura 13, el sistema cumple con las características deseadas, ya que se ubicaron correctamente los polos dominantes, por lo que ahora se simulará numéricamente el sistema para observar el comportamiento del robot. Dicho comportamiento se puede visualizar en la Figura 14, en donde se puede observar que el control tiene una gran eficiencia, basada en los parámetros de sobretiro (1.25) y tiempo de asentamiento (5% de tolerancia del valor final de asentamiento). En cuanto al tiempo de asentamiento, que es prácticamente 205% más rápido que el sistema no controlado con par torsional constante.

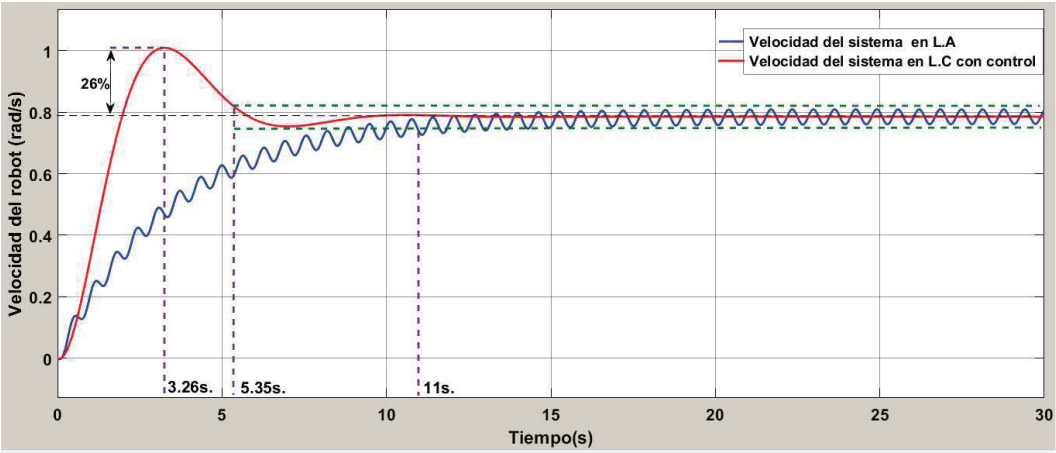


Figura 14. Comparación de las respuestas de velocidad de la planta compensada y no compesada.

Es importante mencionar que la respuesta del sistema podría mejorarse aún más, modificando los requerimientos del control, sin embargo, se tienen que considerar las características del actuador para realizar dicho control, por lo que hay que observar qué par es requerido para que el sistema pueda comportase de la manera deseada. Dicho par requerido se muestra en la Figura 15 en donde se tiene que el par máximo es de 5.325N*mm.

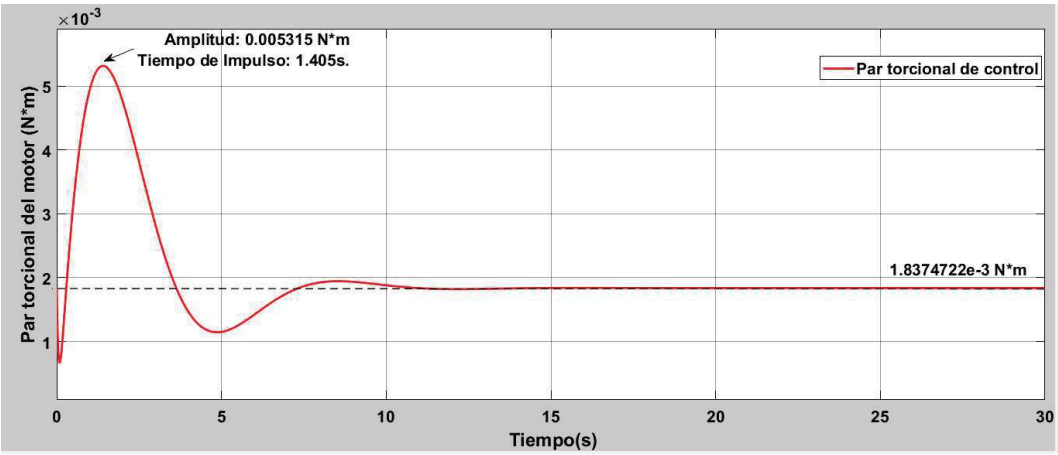


Figura 15. Par torsional requerido para el control de la planta.

Por todo lo anterior expuesto, el diseño del controlador cumple con los criterios de desempeño planteados como especificaciones, ya que polos y ceros de la planta se tienen que cancelar mediante un compensador, y así lograr que el sistema sea estable en lazo cerrado, siendo esto así, no es necesario diseñar otro controlador para mejorar la respuesta.

5. Conclusiones

En este trabajo se presentó el desarrollo de un controlador basado en el análisis del lugar geométrico de las raíces (LGR) para un robot rueda con locomoción por péndulo inercial. Para esto, se presentó el desarrollo y la validación del modelo del sistema, el cual fue obtenido usando el formalismo de Euler-Lagrange. Al aplicar la metodología del LGR al sistema, se obtuvieron resultados satisfactorios para el desempeño del controlador, el cual cumple con las especificaciones establecidas para estabilizar al sistema en lazo cerrado. Además, se consideró la limitante física del actuador. Por todo lo anterior, se puede afirmar que es viable implementar experimentalmente el controlador propuesto. En este sentido, actualmente se está terminando de construir el primer prototipo físico de robot rueda con péndulo inercial en la Universidad Tecnológica de la Mixteca, con el cual se podrán evaluar el diseño, el modelo dinámico y el controlador a partir de resultados experimentales.

Referencias

[1] Nozaki, K., Murakami, T. *"A motion control of two-wheels driven mobile manipulator for human robot cooperative transportation"*. In: 35th Annual Conference of IEEE on Industrial Electronics 2009, IECON'09, pp. 1574–1579. IEEE Press, New York, 2009. doi:10.1109/IECON.2009.5414734.

[2] Nagarajan, U., Kantor, G., Hollis, R. *"The ballbot: An omnidirectional balancing mobile robot"*. *The International Journal of Robotics Research*. SAGE Journals, Vol. 33, No. 6, pp. 917–930, 2014. doi:https://doi.org/10.1177/0278364913509126.

[3] Garcia, R.A., Arias, M. *"Linear controllers for the NXT ballbot with parameter variations using linear matrix inequalities". IEEE Control Systems Magazine.* IEEE, Vol. 36, No. 3, pp. 121–136, 2016. doi:10.1109/MCS.2016.2535915.

[4] DeJong, B.P., Karadogan, E., Yelamarthi, K., Hasbany, J. *"Design and analysis of a four-pendulum omnidirectional spherical robot". Journal of Inteligent & Robotic Systems.* Springer, Vol. 86, No. 1, pp. 3–15, 2017. doi:https://doi.org/10.1007/s10846-016-0414-4.

[5] Muralidharan, V., Mahindrakar, A.R. *"Geometric controllability and stabilization of spherical robot dynamics". IEEE Transactions on Automatic Control*. IEEE, Vol. 60, No. 10, pp. 2762–2767, 2015. doi:10.1109/TAC.2015.2404512.

[6] Lee, J., Han, S., Lee, J. *"Decoupled dynamic control for pitch and roll axes of the unicycle robot". IEEE Transactions on Industrial Electronics*. IEEE, Vol. 60, No. 9, pp. 3814–3822, 2013. doi:10.1109/TIE.2012.2208431.

[7] Zhu, Y., Gao, Y., Xu, C., Zhao, J., Jin, H., Lee, J. *"Adaptive control of a gyroscopically stabilized pendulum and its application to a single-wheel pendulum robot". IEEE/ASME Transactions on Mechatronics*. IEEE, Vol. 20, No. 5, pp. 2095–2106, 2015. doi:10.1109/TMECH.2014.2363090.

[8] Jin, H., Wang, T., Lee, J. *"Unicycle robot stabilized by the effect of gyroscopic precession and its control realization based on centrifugal force compensation". IEEE/ASME Transactions on Mechatronics*. IEEE, Vol. 21, No. 6, pp. 2737–2745, 2016. doi:10.1109/TMECH.2016.2590020.

[9] Park, J.H., Jung, S. *"Development and control of a single-wheel robot: Practical mechatronics approach". Mechatronics*. Elsevier, Vol. 23, No. 6, pp. 594–606, 2013. doi:https://doi.org/10.1016/j.mechatronics.2013.05.010.

[10] Lee, S.D., Jung, S. *"Experimental verification of stability region of balancing a single-wheel robot: An inverted stick model approach"*. In: Industrial Electronics Society, IECON 2015 - 41st Annual Conference of the IEEE, pp. 4556–4561. IEEE Press, New York, 2015. doi:10.1109/IECON.2015.7392810.

[11] Schoonwinkel, A., *"Desing and test of a computer stabilized unicycle"*. Tesis de Doctorado, Stanford University, Dec. 1987.

[12] Vos, W. D., Von Flotow, A. H., *"Dynamics and nonlinear adaptive control of an autonomous unicycle: Theory and experiment"*, in Proceedings of the 29th IEEE Conference on Decision and Control, pp. 182-187, IEEE, 1990. doi: 10.1109/CDC.1990.203573.

[13] Jin, H, Hwang, J, Lee. J., *"A balancing control strategy for a one-wheel pendulum robot based on dynamic model decomposition: Simulations and experiments"*, *IEEE/ASME Transactions on Mechatronics*. IEEE, Vol. 16, No. 4, pp. 763-768, 2011. doi: 10.1109/TMECH.2010.2054102.

[14] Chase, R., Pandya, A. *"A review of active mechanical driving principles of spherical Robots". Robotics*. MDPI, Vol. 1, No. 1, pp. 3–23, 2012. doi:10.3390/robotics1010003.

[15] Herrera-Cordero, M.E, Arias-Montiel, M., Lugo-González, E. *"Design and Dynamic Modeling of a Novel Single-Wheel Pendulum Robot"*. In: Mechanism Design for Robotics, Marco Ceccarelli and Alessandro Gasparetto (Eds.), pp. 353-360. Springer, 2019. doi: https://doi.org/10.1007/978-3-030-00365-4_42.

Ingeniería y Automatización, Capítulo 22, pp. 287 - 299.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Verificación e Implementación en Hardware del Código de Redundancia Cíclica

Soto Hernández Enrique*, Ramírez Villa Goretti, Cházaro Zaharías Adriana C., Traslosheros Michel Alberto y González Chávez Miguel A.

Universidad Aeronáutica en Querétaro.
Carretera Estatal 200 Querétaro - Tequisquiapan No. 22154. Col. Parque Aeroespacial de Querétaro. Colón, Querétaro. México. C.P. 76278
*Correo electrónico de contacto: enrique.soto.hdez@gmail.com

Resumen

Enfocados a la formación de recursos humanos para el desarrollo de sistemas satelitales, mediante la adquisición de habilidades en el uso de dispositivos electrónicos, se presentará la simulación y verificación de hardware personalizado en una plataforma lógica programable (FPGA). El artículo se centrará en la integración en hardware de un módulo de gestión de transferencia de datos mediante la codificación y control de errores con el código de redundancia cíclica (CRC) USB-CRC5, implementando la tarjeta de desarrollo DE0-Nano FPGA Cyclone IV para el sintetizado en una computadora básica basada en el procesador Nios II.

Palabras clave: CRC, FPGA, Verilog, SoPC.

1. Introducción

Con el interés de desarrollar capacidades en el ámbito espacial se propuso el proyecto *"Sistema de bajo consumo de energía para el desarrollo y entrenamiento de sistemas satelitales"* [1], cuyo objetivo se centra en el desarrollo de habilidades para el uso de dispositivos electrónicos enfocados al sector aeroespacial. A partir de dicho enfoque se pretende implementar a futuro un sistema de procesamiento de señales en dispositivos FPGA grado aeroespacial, aplicando normativa para aplicaciones de control crítico.

El enfoque de procesamiento de los FPGA (arreglo de compuertas lógicas programables) permite realizar operaciones aritméticas y lógicas y el almacenamiento o transferencia de datos binarios mediante la combinación de un conjunto reducido de componentes lógicos. "Se puede pensar en el proceso de conexión de los diversos componentes [lógicos] para obtener la configuración deseada como si se tratase de una forma de programación" [2]. El programa resultante es hardware y se denomina programa cableado. Este procedimiento se ha vuelto popular con la ayuda de dispositivos FPGA y el uso de Lenguajes Descritos por Hardware (HDLs), simplificando la configuración y verificación de hardware específico. Este último (proceso de verificación) es alrededor del cual se centrará el proyecto descrito en este documento, a través del desarrollo de un código de redundancia cíclica (CRC) aplicado al proceso de transmisión de datos en un FPGA.

El código de redundancia cíclica presenta un método general para la detección de errores en comunicación digitales; el cálculo de CRC se realiza con base en los datos a enviar dentro del paquete por lo que mantiene una dependencia con los mismos. Mediante la vinculación y aplicación de los conceptos desarrollados en el marco teórico, el presente artículo expondrá el procedimiento del diseño del algoritmo USB-CRC5 (código de redundancia cíclica con generador de grado 5) y su implementación de hardware en una plataforma FPGA. El diseño del módulo de gestión de transferencia de datos, mediante la codificación y control de errores en palabras binarias, se

incorporará a una computadora basada en el procesador Nios II sintetizada en la tarjeta FPGA DE0-Nano.

2. Marco teórico

El marco teórico de este documento, enfocado al estudio de procesamiento de señales en una plataforma FPGA, se estructurará a partir de la mención de definiciones y la interpretación de conceptos fundamentales. La información se proveerá en tres partes, comenzando por una visión general de la computadora, seguido del planteamiento de la matemática aplicada para el código de redundancia cíclica (CRC) y finalizando con el concepto SoPC.

2.1 Visión general

2.1.1 Organización y arquitectura

Al momento de definir la computadora, será relevante conocer la relación entre los conceptos *organización y arquitectura*, ya que no se podría abordar la definición del sistema implementado en el dispositivo FPGA desde una perspectiva arquitectónica sin hacer referencia a la organización del hardware o remitirse a las unidades funcionales sin dar lugar a especificaciones de arquitectura.

Stallings [2] hace mención de la arquitectura de la computadora y la define como aquellos atributos que tienen un impacto directo en la ejecución lógica de un programa. Coloquialmente, nos referiremos a la arquitectura como aquellos atributos de un sistema visibles al programador. La organización de una computadora se refiere a las unidades funcionales y su interconexión, dando lugar a especificaciones arquitectónicas. A partir de los conceptos mencionados, se definen los siguientes atributos con características:

- **Atributos arquitectónicos:** conjunto de instrucciones, número de bits usados para la representación de datos, mecanismos de E/S e interfaces entre el computador y los periféricos.

- **Atributos organizacionales:** detalles del hardware transparente al programador, tales como señales de control e interrupciones.

2.1.2 Programa cableado

Anteriormente, se mencionó la definición de programa cableado como procedimiento dirigido a la configuración de hardware mediante lenguajes de programación.

Considérese ahora la siguiente alternativa: se construye una configuración de uso general de funciones lógicas y aritméticas. Este hardware ejecutará funciones diferentes según las señales de control captadas. Para el caso del hardware específico, Figura 1 a), el sistema produce resultados a partir de la captación de datos. Con el hardware de uso general, Figura 1 b), el sistema acepta datos y señales de control produciendo resultados. Así, en lugar de reconfigurar el hardware para procedimientos específicos, el programador necesitará proporcionar un nuevo conjunto de señales de control.

Con el objetivo de definir la arquitectura y organización de un sistema integrado en un dispositivo FPGA, se vuelve necesario el uso de hardware específico y hardware de uso general.

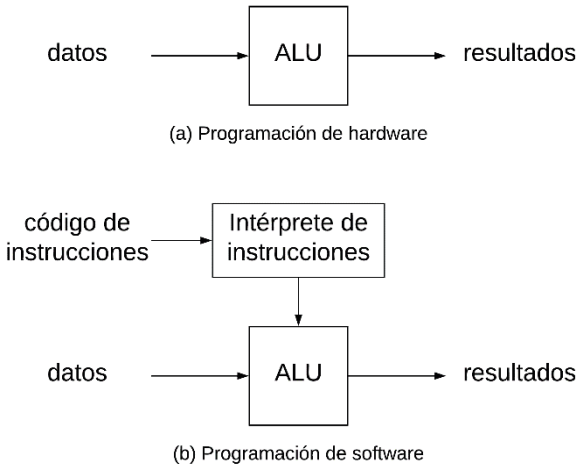


Figura 1. (a) hardware específico y (b) hardware de uso general.

2.2 Componentes de la computadora

Los diseños actuales de computadores se basan en el concepto desarrollado por John von Neumann. A este diseño se le conoce con el nombre de arquitectura de von Neumann y se basa en tres criterios:

- almacenamiento de datos e instrucciones en una sola memoria de lectura-escritura;
- contenidos en memoria direccionados indicando su posición; y
- la ejecución se produce siguiendo una secuencia de instrucciones a la vez.

La Figura 1 a) muestra dos conceptos esenciales del sistema: un intérprete de instrucciones y un módulo de uso general para funciones aritméticas y lógicas. Estos dos elementos constituyen la Unidad Central de Procesamiento (CPU).

Los datos y las instrucciones de entrada deben ser captados, requiriendo de alguna clase de módulo de entrada para desarrollar la captación de las señales. Análogamente, se precisa un medio para proporcionar los resultados, el módulo de salida. Estos módulos se conocen con el nombre de componentes de Entrada/Salida (E/S).

Por último, las operaciones con datos pueden necesitar acceder a más de un operador y en una secuencia determinada. Para ello, se precisa un sitio para el almacenamiento temporal, tanto para las instrucciones como para los datos. Este módulo se denominará memoria o memoria principal.

2.3 Código de redundancia cíclica

Los códigos de redundancia cíclica (CRCs) tienen un amplio uso de aplicaciones en comunicación de datos y dispositivos de almacenamiento. Primero, se presentan los polinomios binarios que facilitarán la definición de los CRCs.

El polinomio generador se representa como sigue:

$$g(x) = g_m x^m + g_{m-1} x^{m-1} + \cdots + g_0 \; ; \tag{1}$$

Mensaje a codificar:

$$u(x) = u_k x^k + u_{k-1} x^{k-1} + \cdots + u_0 \; ; \tag{2}$$

Palabra codificada:

$$v(x) = u(x)\, x^{n-k} + s(x) \; ; \tag{3}$$

Donde,

$$s(x) = R_{g(x)}[u(x)\, x^{n-k}] \text{ , y} \tag{4}$$

$$n = k + m. \tag{5}$$

Esto se puede escribir como se muestra en la Figura 2.

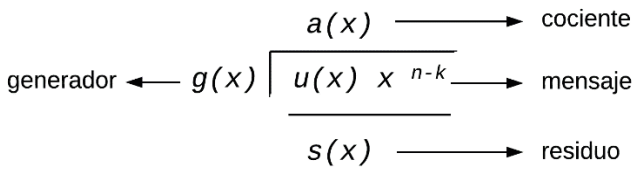


Figura 2. Relación entre polinomios.

CRC es un código de bloque de comprobación, utilizado para la detección de errores en el que la palabra $u(x)$, ec. 2, debe dividirse por un número predeterminado denominado palabra generadora o $g(x)$, ec. 1, obteniendo el residuo $s(x)$, ec. 4, con el fin de obtener la palabra codificada $v(x)$, ec. 3.

Los códigos CRC son nombrados así ya que el código de verificación es redundante, no agrega información, y el algoritmo se basa en códigos cíclicos.

2.4 Sistema-en-Chip-Programable

Se denomina SoPC (Sistema-en-Chip-Programable, *System-On-a-Programmable Chip*) a todo sistema basado en unidades centrales de procesamiento, memoria y otros periféricos incluidos en un dispositivo lógico programable (FPGA).

Un dispositivo FPGA contiene celdas lógicas e interconexiones que se pueden configurar para realizar una función específica. Las funcionalidades de hardware deseadas se describen en código HDL (Lenguaje Descrito por Hardware) que es luego sintetizado e implementado por las celdas lógicas.

Por lo tanto, es posible adaptar el procesador, seleccionar los periféricos de E/S necesarios, crear una interfaz de E/S personalizada y desarrollar módulos de hardware específico para tareas de cálculo intensivo. “El sistema integrado basado en SoPC proporciona una nueva dimensión de flexibilidad, tanto el hardware como el software pueden ser personalizados para satisfacer necesidades específicas” [3].

Ingeniería y Automatización, Capítulo 22, pp. 287 - 299.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Metodología

El presente proyecto contempló la configuración del dispositivo FPGA DE0-Nano Cyclone IV de Altera para el sintetizado de un módulo de gestión de transferencia de datos mediante la codificación y control de errores con el código de redundancia cíclica USB-CRC5; dirigiendo una investigación hacia la configuración de hardware mediante Verilog HDL [3], la sintetización de un computador basado en el soft-processor Nios II/e [3-6] y la simulación e implementación de CRCs [7 y 8].

3.1 Simulación del USB-CRC5

Mediante el lenguaje de programación GNU Octave, software libre con sintaxis ampliamente compatible con MATLAB, se efectuó la simulación para la obtención del residuo $s(x)$, ec. 4, mediante la programación del siguiente pseudocódigo,

$$\begin{aligned}&\text{rutina } CRC\ (N_{in}, M_{in})\\ &\quad M_{out} = M_{in}\\ &\quad \text{for } (i = 0; i < N; i++)\\ &\qquad M_{out} = CRC_{serial}(N_{in}, M_{out})\\ &\text{return } M_{out}\end{aligned} \tag{6}$$

El pseudocódigo presentado cuenta con la aplicación del método de recursividad lineal, en donde CRC_{serial} es representado por el LFSR (registro de desplazamiento con retroalimentación lineal) en la Figura 3, donde u_i representa el mensaje a ser codificado, ec. 2, y $i = k, k-1, \dots, 0$.

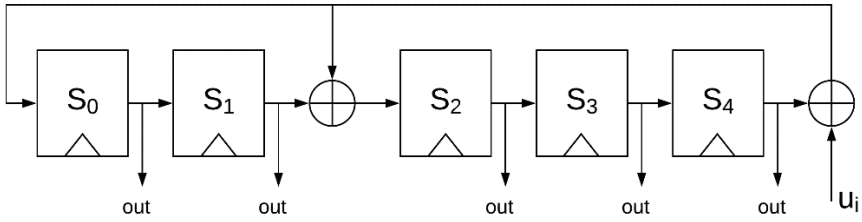


Figura 3. LFSR para la obtención del residuo $s(x)$ mediante el polinomio generador $g(x) = x^5 + x^2 + 1$.

Del LFSR presentado se deducen las relaciones, presentadas en la ec. 7 para la obtención del residuo $s(x)$. El algoritmo 1 presenta el código para la verificación del algoritmo USB-CRC5.

$$\begin{aligned}s_0 &= s_4 \oplus u_i\\ s_1 &= s_0\\ s_2 &= s_1 \oplus (s_4 \oplus u_i)\\ s_3 &= s_2\\ s_4 &= s_3\end{aligned} \tag{7}$$

```
function [s] = crc_iteration(idle, u)

  s = idle;

  for i = 0:3
    s = crc5_serial(s, u(4-i));
  end

endfunction
```

```
function [s] = crc5_serial(s_old, u)

  s(1)      = xor(s_old(5), u);
  s(2)      = s_old(1);
  s(3)      = xor(s_old(2), xor(s_old(5), u));
  s(4:5)    = s_old(3:4);

endfunction
```

```
clc; clear;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                            USB-CRC5                                        %
%              Verificación de Hardware Serial                               %
%                            Sep. 2018                                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%   [LSB-MSB]
u = [1 0 1 1];

%% Inicialización del Hardware
s = [0 0 0 0 0];

%% Solución por Recursividad Lineal del LFSR
s = crc_iteration(s, u);
v = [s, u];

disp('mensaje:'), disp(u)
disp('residuo:'), disp(s)
disp('palabra codificada:'), disp(v)
```

Algoritmo 1. Simulación del algoritmo USB-CRC5.

Por último, se ingresó el mensaje con valor $0Dh$ obteniendo la siguiente respuesta del algoritmo programado (Figura 4):

Figura 4. Respuesta de la simulación del USB-CRC5.

La simulación proporcionó la identificación de atributos arquitectónicos y la traducción del algoritmo a Verilog HDL.

3.2 Definición de requerimientos

El módulo de gestión de trasferencia de datos permitirá mostrar en la pantalla de la computadora de usuario la palabra codificada $v(x)$, ec. 3, calculada por el dispositivo FPGA a partir de la entrada del mensaje $u(x)$, ec. 2, en función del polinomio generador $g(x) = x^5 + x^2 + 1$ (Figura 3). El sistema debe cumplir con las siguientes especificaciones:

Requisitos funcionales

i. El mensaje $u(x)$ debe ser proporcionado por el usuario en formato hexadecimal.

ii. Se podrá mostrar en pantalla el valor del residuo $s(x)$ calculado, en formato hexadecimal, esto con el fin de verificar la salida del módulo de codificación.

iii. Se mostrará en pantalla la palabra codificada $v(x)$ en formato hexadecimal.

Requisitos no funcionales

- *Hardware*. El sistema, basado en la incorporación del procesador Nios II/e, se debe implementar en la plataforma de desarrollo DE0-Nano, equipo existente en el laboratorio de Sistemas Embebidos en la UNAQ.

- *Software*. Implementar la herramienta Quartus II 13.1 (64-bit) Web Edition para gestionar el diseño e implementación del sistema en la plataforma FPGA.

3.3 Organización y arquitectura del sistema

A partir de la simulación del algoritmo USB-CRC5 (algoritmo 1) se identificaron atributos para su integración en una plataforma FPGA. Debido a que el algoritmo y la organización del sistema serán incorporados en el dispositivo mediante hardware digital, y comandado por software, se establecieron entradas y salidas por módulos. A continuación, se muestran los atributos del sistema (Tablas 1-3).

Tabla 1. Atributos arquitectónicos del hardware específico CRC5.

Nombre	Tipo de Señal	Ancho de palabra [bits]	Descripción
clk	entrada	1	Señal externa de reloj para la sincronización de datos, programada a 115.2 kHz.
rst	entrada	1	Señal de control para el reinicio del sistema, ‘1’ para mantener al circuito inactivo y ‘0’ para iniciar el procesamiento de residuo $s(x)$.
done	registro de salida	1	Señal de control para determinar la finalización del cálculo de la palabra $s(x)$, ‘0’ para estado no finalizado y ‘1’ para indicar finalización del cálculo.
u	entrada	4	Entrada de la palabra $u(x)$.
s	registro de salida	5	Salida del residuo $s(x)$.

Tabla 2. Atributos arquitectónicos del SoPC.

Nombre	Tipo de Señal	Ancho de palabra [bits]	Descripción
clk	entrada	1	Señal externa de reloj, programada a 50 MHz.
reset	entrada	1	Señal de control para el reinicio del SoPC; conexión al push-button externo de la tarjeta para efectuar restablecimientos del sistema completos.
dram_…	entradas, salidas y bidireccionales		Múltiples conexiones para el direccionamiento del dispositivo SDRAM.
crc_rst	salida	1	Señal de control para el control del hardware CRC5.
crc_done	entrada	1	Señal de control, aviso de finalización de cálculo para la palabra *s(x)*.
crc_u	salida	4	Salida de la palabra $u(x)$.
crc_s	entrada	5	Entrada del residuo $s(x)$.

Tabla 3. Atributos arquitectónicos del PLL.

Nombre	Tipo de Señal	Ancho de palabra [bits]	Descripción
inclk0	entrada	1	Señal externa de reloj a 50 MHz procedente del osciloscopio.
c0	entrada	1	Señal de referencia configurada a 50 MHz y desplazada -3 ns para la SDRAM.
c1	entradas, salidas y bidireccionales	1	Señal de referencia configurada a 50 MHz, sin desplazamiento, para el SoPC.
c2	salida	1	Señal de referencia configurada a 115.2 kHz, sin desplazamiento, para el hardware CRC5.

La organización del sistema se basa en la incorporación de los siguientes módulos, de los cuales el funcionamiento es transparente al programador:

- Señal externa de reloj (señal de referencia del sistema).
- Memoria RAM interna (almacenamiento de la información de estado del sistema).
- Puerto de comunicación serial JTAG (permitirá comunicar al sistema implementado en el dispositivo con una computadora personal para la programación y depuración del diseño).
- Temporizador del sistema (periférico que permitirá temporizar eventos en el procesador).
- Módulos de E/S (múltiples módulos para las entradas y salidas de señales de control y palabras dirigidas al hardware del CRC).

3.4 Construcción y generación del SoPC

Altera provee el software Quartus II como soporte de sus dispositivos lógicos programables. Esta herramienta de desarrollo se referencia como “herramienta de diseño totalmente integrada” [5], ya que abarca todas las aplicaciones necesarias para realizar un proyecto en FPGA, sin requerir ninguna otra aplicación adicional.

El proceso de diseño del SoPC, plasmado en la Figura 5, involucra el diseño de hardware y del software asociado. El hardware del SoPC se realizó integrando los componentes de biblioteca conforme a los atributos de organización del sistema. Simultáneamente, el desarrollo del software se realizó acorde a la arquitectura de hardware del sistema.

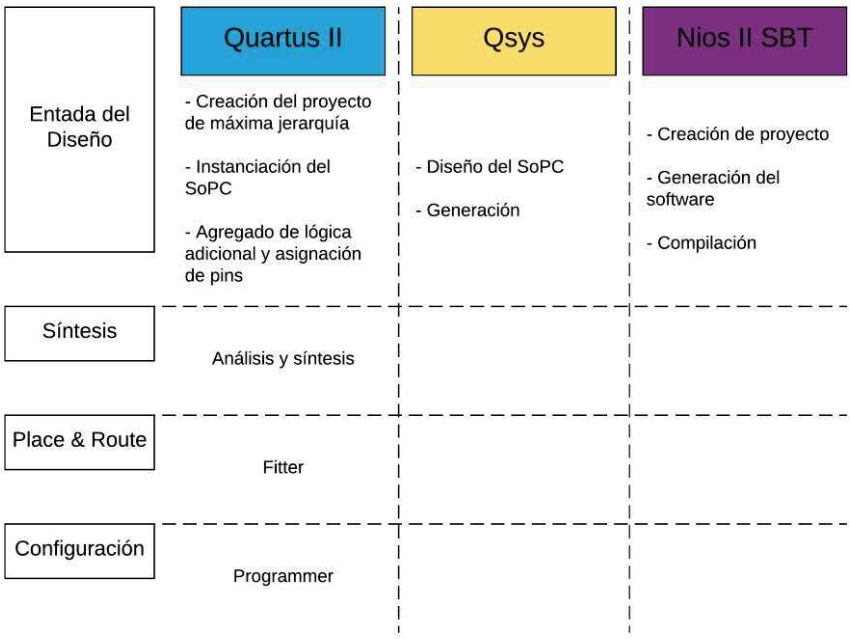


Figura 5. Flujo de diseño del SoPC de Altera [5].

La Figura 5 muestra las herramientas propuestas para el flujo de diseño del sistema. La descripción de sistemas embebidos se desarrolló con la herramienta Qsys y el software se realizó mediante la herramienta Nios II SBT Eclipse.

Mediante la simulación del pseudocódigo (ec. 6) y la definición de requerimientos, se tradujo el algoritmo 1 al lenguaje de programación Verilog. Comenzando por la declaración de las siguientes funciones:

- *crc5_serial* para la incorporación de las relaciones deducidas del LFSR presentado en la Figura 3; y

- *crc_iteration* para la solución de $s(x)$ mediante el método de recursividad lineal.

Finalmente se declaró la máquina de estados finitos para el control del sistema mediante la entrada y salida crc_rst y crc_done, respectivamente.

```
1. module CRC5
2. (
3.     input      wire            clk     ,
4.     input      wire            rst     ,
5.     output     reg             done    ,
6.     input      wire    [3:0]   u       ,
7.     output     reg     [4:0]   s
8. );
9.
10./* Implementación Serial de LFSR */
11.    function [4:0] crc5_serial;
12.        input   [4:0]   s_old;
13.        input           u;
14.    begin
15.        crc5_serial[0]     = (s_old[4] ^ u);
16.        crc5_serial[1]     = s_old[0];
17.        crc5_serial[2]     = s_old[1] ^ (s_old[4] ^ u);
18.        crc5_serial[4:3]   = s_old[3:2];
19.    end
20.    endfunction
21./* Solución por Recursividad Lineal */
22.    function [4:0] crc_iteration;
23.        Input   [4:0]   idle;
24.        input   [3:0]   u;
25.        integer i;
26.
27.    begin
28.        crc_iteration = idle;
29.        for(i = 0; i < 4; i = i + 1)
30.            crc_iteration = crc5_serial(crc_iteration, u[3-i]);
31.
32.    end
33.    endfunction
34.
35.    reg [3:0] r_u_R= 4'd0;
36.    reg [3:0] r_u  = 4'd0;
37.
38.    always @(posedge clk, posedge rst)
39.     if (rst)
40.        begin
41.            s       <= 5'h00;
42.            done    <= 1'b0;
43.        end
44.     else
45.        begin
46.            r_u_R   <= u;
47.            r_u     <= r_u_R;
48.            s       <= crc_iteration(s, r_u);
49.            done    <= 1'b1;
50.        end
51.endmodule
```

Algoritmo 2. Implementación en hardware del algoritmo USB-CRC5.

Ingeniería y Automatización, Capítulo 22, pp. 287 - 299.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

4. Resultados

El código mostrado a continuación, algoritmo 3, es la función descrita por software (lenguaje de programación C) encargada de presentar la solución de las ecuaciones 3 y 4 (sección 2.3) mediante el control del comportamiento del hardware USB-CRC5. El algoritmo sigue la siguiente secuencia de instrucciones:

i. Envío del mensaje $u(x)$ a la dirección CRC_U (atributo de salida crc_u en la Tabla 2).

ii. Inicialización del hardware USB-CRC5 mediante el envío del dato '0' en la dirección CRC_RST (atributo de salida crc_rst en la Tabla 2).

iii. La función se encontrará en espera hasta recibir la señal de control CRC_DONE (atributo de entrada crc_done en la Tabla 2) con el valor '1' para indicar que el cálculo del valor del residuo $s(x)$ ha finalizado.

iv. Lectura de la dirección CRC_S (atributo de entrada crc_s en la Tabla 2) para obtener el valor del residuo $s(x)$.

```
1.  void USB_CRC5(int u, int *s){
2.
3.      IOWR(CRC_U      , 0, u);
4.      IOWR(CRC_RST    , 0, 0x00);
5.
6.      while (IORD(CRC_DONE, 0) == 0);
7.          *s = IORD(CRC_S, 0);
8.
9.      IOWR(CRC_RST    , 0, 0x01);
10. }
```

Algoritmo 3. Función USB_CRC5 para el comando de hardware.

```
1.  int main(){
2.      alt_putstr("-------------------------------------\r\n");
3.
4.      int u = 0x0D;
5.      int s, v;
6.
7.      USB_CRC5(u, &s);
8.      v = (u << 5) + s;
9.
10.     printf("Mensaje:            %02Xh\r\n", u);
11.     printf("Residuo:            %02Xh\r\n", s);
12.     printf("Palabra codificada: %03Xh\r\n", v);
13.
14.     alt_putstr("-------------------------------------\r\n\n");
15.     return 0;
16. }
```

Algoritmo 4. Código fuente, declaración de mensaje y cálculo de palabra codificada.

Ingresando el dato $0Dh$ para el mensaje especificado por el usuario, mostrado en el algoritmo 4 perteneciente al código fuente del sofwtare, se obtiene el resultado mostrado en la Figura 6.

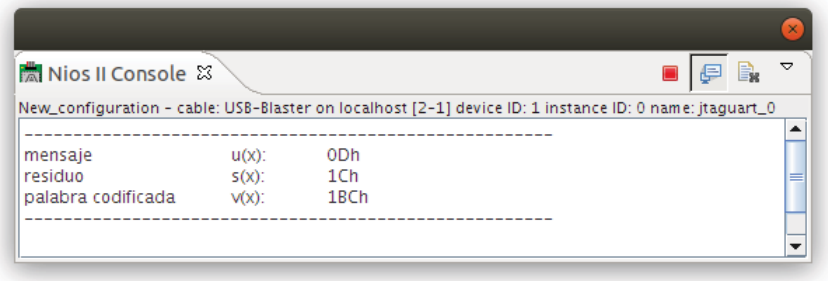


Figura 6. Respuesta del sistema.

La ecuación 3 puede ser traducida al lenguaje de programación C como se muestra en el algoritmo 3, para el caso específico en donde el grado del polinomio mensaje cuenta con un valor $k = 3$ y el polinomio generador $m = 5$. Por lo tanto, de la ecuación 5 se tiene $n = 5$.

$$\mathrm{v} = (\mathrm{u} \ll 5) + \mathrm{s}$$
$$v(x) = u(x)\, x^5 + s(x) \tag{8}$$

5. Conclusiones

Luego de realizar la integración del SoPC a la plataforma DE0-Nano y ejecutar las instrucciones para comandar el hardware diseñado, se concluye con la validez del sistema mediante la comparación de los datos presentados en la simulación e implementación, Figura 4 y 6 respectivamente.

Del pseudocódigo presentado en la ecuación 6, N corresponde al grado del polinomio $u(x)$, ec. 2. De esto identificamos que el algoritmo se restringe a resolver $s(x)$ para un mensaje con menor grado que el polinomio generador (esto es $k < m$). La metodología y el algoritmo implementados cuentan con la capacidad de adaptar diferentes CRCs (p. ej. CRC-16 y CRC-32) al sistema mediante la modificación del LFSR en la función CRC_{serial}, ec. 6, y los atributos arquitectónicos de manera que se adapten a la definición de requerimientos.

Como aplicación a trabajos futuros, se tiene contemplado emplear el código CRC-16, mostrado en la Figura 7, y su adaptación con las relaciones mostradas en la ecuación 9, con el fin de gestionar palabras de mensajes con longitud inferior a 16-bits.

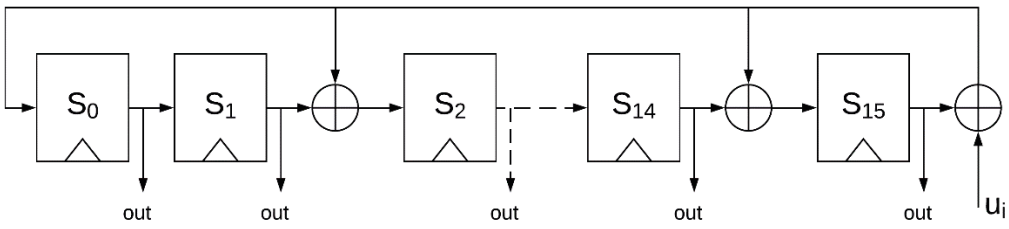


Figura 7. LFSR para la obtención del residuo s(x) mediante el polinomio CRC-16 $g(x) = x^{16} + x^{15} + x^2 + 1$.

$$\begin{aligned} s_0 &= s_{15} \oplus u_i \\ s_1 &= s_0 \\ s_2 &= s_1 \oplus (s_{15} \oplus u_i) \\ s_{3\to 14} &= s_{2\to 13} \\ s_{15} &= s_{14} \oplus (s_{15} \oplus u_i) \end{aligned} \tag{9}$$

Incursionando en la formación de recursos para el desarrollo de sistemas satelitales, se ha impactado directamente a la fase de capacitación del grupo de trabajo sobre el uso de dispositivos electrónicos enfocados al desarrollo aeroespacial; aportaciones dirigidas al proyecto de *"Sistema de bajo consumo de energía para el desarrollo y entrenamiento de sistemas satelitales"* [1], con clave 275809 perteneciente a la Convocatoria 2016 AEM-CONACYT.

Referencias

[1] González M. A., Traslosheros A., Ramírez G., *"Plataforma de bajo consumo de energía para la capacitación en sistemas satelitales, Primer encuentro académico: Hacia el desarrollo espacial de México"*, CONACYT-AEM, Querétaro, 2017.
[2] Stallings W., *"Organización y arquitectura de computadores"*, Pearson, España, 4ta edición, 2006.
[3] Chu P. *"Embedded SoPC Design with Nios II Processor and Verilog Examples"*, Wiley, EUA, 3ra edición, 2012.
[4] Kilts S. *"Advanced FPGA Design: Architecture, Implementation, and Optimization"*, Wiley, EUA, 1era edición, 2007.
[5] Cayssials R. *"Sistemas Embebidos en FPGA"*, Alfaomega, Argentina, 1era edición, 2014.
[6] Nurmi J. *"Processor Design: System-On-Chip Computing for ASCIs and FPGAs",* Springer, Países Bajos, 2007.
[7] Ramabadran T. y Gaitonde S. *"A Tutorial on CRC Computations"*, IEEE, EUA, 1ra edición, 1988.
[8] Lin S. y Costello D. *"Error Control Coding: Fundamentals and Applications"*, Prentice-Hall, EUA, 1983.

Ingeniería y Automatización, Capítulo 23, pp. 300 - 307.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Propuesta de Modelo en una Microred Eléctrica Basado en un Sistema Híbrido ante una Falla Eléctrica Detectada

Méndez Navarro Alejandro, Rodríguez Hernández Jose Jorge, Ávalos García Jorge Hugo y Villalobos Guerra Arnoldo Ulises

Instituto Tecnológico de Estudios Superiores de Zamora
amendez@teczamora.mx

Resumen

Las microredes son diseñadas para trabajar en dos modos de operación, conectadas a la red eléctrica principal y de forma autónoma. Son usadas para mejorar la eficiencia, seguridad y calidad de la energía eléctrica. El consumidor tiene la capacidad de administrar y gestionar la energía eléctrica. Son consideradas como sistemas de potencia distribuidos ya que son integradas por fuentes renovables, fuentes convencionales y sistemas de almacenamiento de energías, por consecuencia el flujo de potencia en una microred no es de forma unidireccional. El enfoque propuesto en el presente trabajo es basado en observadores de estado a partir de la medición de voltajes y corrientes en la microred. Dichas mediciones se realizan por dispositivos que incluyen medidores inteligentes, redes de sensores inalámbricos y unidad de medición de fasores.

Palabras clave: Sistemas híbridos, microredes eléctricas, observadores de estado, detección de fallas.

1. Introducción

Debido al rápido crecimiento en las últimas décadas en términos de tecnologías de la información y comunicación, monitoreo y medición, control y automatización, almacenamiento de energía eléctrica y a la gran demanda de confiabilidad, seguridad y disponibilidad en los sistemas de potencia eléctrica, éstos son testigos de la aparición de las microredes eléctricas inteligentes. Una microred eléctrica inteligente tiene la capacidad de formar un sistema distribuido de potencia la cual está integrada por fuentes de energías renovables (eólica, paneles fotovoltaicos, celdas de combustiones, entre otros), fuentes de energías convencionales (generadores diésel y gas natural) y dispositivos de almacenamiento de energía eléctrica; estas fuentes de energía suministran potencia eléctrica en forma local dentro de una región de cobertura establecida. Las fuentes de energías convencionales son usadas cuando las fuentes de energías renovables no son capaces de suministrar la potencia eléctrica que demandan las cargas conectadas en una microred. El objetivo principal de las microredes es formar un sistema eficiente y confiable capaz de entregar la potencia eléctrica incluso en casos de fallos en la red principal [1-3].

Las microredes eléctricas son diseñadas para trabajar en dos modos de operación, conectada a la red eléctrica principal y autónoma. En el caso de presentarse una falla o alguna situación de emergencia en una red eléctrica principal, una microred es capaz de seguir operando en modo autónomo y se garantiza el suministro de potencia eléctrica a las cargas conectadas en la microred, en el momento que la falla en la red eléctrica principal es restaurada la microred puede conectarse a la red principal [4].

Como una microred está integrada por distintos generadores distribuidos, el flujo de potencia eléctrica se puede presentar de forma bidireccional bajo distintas condiciones en los modos de operación. Por otro lado, los niveles de corriente que se presentan en fallas ocasionadas por los cortos circuitos son de magnitudes distintas en el modo conectado y autónomo. Es por ello que es

necesario diseñar los sistemas de protección adaptables que sean capaces de trabajar con seguridad en ambos modos de operación [5-6].

Una red eléctrica opera bajo condiciones balanceadas, cuando el sistema llega a ser no balanceada es debido a alguna falla en algún punto específico en el sistema. Estas fallas se pueden presentar por un sin número de razones, por ejemplo la caída de un árbol sobre las líneas de transmisión o distribución, terremotos que dañen las líneas de transmisión o distribución, aislamiento por avería en algún componente conectado a la red, y demás. Las fallas se clasifican es simétricas y asimétricas. Las fallas simétricas se tienen de dos tipos, las tres fases en cortocircuito o las tres fases a tierra. Por otro lado, fallas asimétricas se caracterizan por ser fallas de línea a tierra, línea a línea, doble línea a tierra. Además existen otros tipos de fallas debido a condiciones como el malfuncionamiento en los actuadores y sensores, variaciones anormales de parámetros del proceso y fallas en equipos debido a cambios estructurales, por mencionar algunos [7].

El objetivo del presente trabajo, es el presentar una propuesta de la dinámica de los microredes eléctricas inteligentes a partir de un enfoque basado en sistemas híbridos ante la detección de una falla eléctrica detectada.

2. Marco Teórico

Los sistemas híbridos lineales (LHS, por sus siglas en inglés *Linear Hybrid Systems*) se caracterizan por interactuar entre sistemas dinámicos continuos y discretos. La naturaleza de estos sistemas resulta de la conjunción entre controladores con decisiones lógicas o controladores embebidos con los procesos físicos, por lo tanto su análisis requiere la combinación de las áreas de la ciencia de la computación e ingeniería en control [1]. La conjunción de las palabras sistemas híbridos usado en el contexto anterior fue usado por primera vez por H. S. Witsenhausen en 1966 en [2]. Witsenhausen reporta el comportamiento dinámico de los sistemas en función de entradas continuas a tramos considerando el estado continuo sin saltos y la conmutación entre sistemas lineales ocurre sólo cuando el estado continuo satisface las condiciones dadas por cada tipo de conmutación. Los sistemas híbridos lineales pueden ser utilizados para representar distintos tipos sistemas de ingeniería tales como sistemas mecánicos [3], sistemas de control con histéresis [4], sistemas de control de nivel de agua [5], electrónica de potencia [6], sistemas biológicos y químicos [1], autoservicio en medios de transporte como bicicletas públicas [7], en aplicaciones de la industria automotriz: control en motor y en tren motriz [8], etc. Las razones para estudiar sistemas híbridos lineales pueden ser muy diversas, según [9] existen tres grandes aplicaciones en las que tienen una gran influencia los sistemas híbridos lineales: 1) diseño de sistemas tecnológicos, 2) lógica en la toma de decisiones y las acciones de control embebido con la combinación de procesos físicos continuos y 3) procesos físicos que presentan un comportamiento no suaves.

Un sistema híbrido lineal $\sum_{\sigma}$ es una colección de sistemas lineales $\Upsilon = \{\sum_1, \cdots \sum_m\}$ donde cada $\sum_i$ para $i = 1,\ldots,\mathrm{m}$ evoluciona en el espacio $\Im = \mathbb{R}^n$ y conmutadas por una señal σ tomadas valores en $\{1,\ldots,m\}$ que determina la evolución del sistema. La ecuación de estado $\sum_{\sigma}$ tiene la siguiente forma

$$\begin{aligned} \dot{x}(\tau) &= A_{\sigma} x(\tau), \qquad x(\tau_0) = x_0 \\ y(\tau) &= C_{\sigma} x(\tau) \\ \sigma &\in 1,\ldots,m. \end{aligned} \tag{1}$$

donde $y \in \mathbb{R}^r$ es la señal de la salida. La evolución lineando cuando $\sigma = i$ es representado por $\sum_i (A_i, C_i)$ o simplemente $\sum_i$ donde las matrices A_i y C_i son de dimensión apropiada.

En el siguiente ejemplo se muestra la interacción entre los estados continuos y discretos en un LHS. La regulación de la temperatura, ver figura 1, $T(t)$ en una habitación con un sistema de calefacción opera con dos estados discretos: encendido y apagado $q(\tau)=\{encendido, apagado\}$. En cada estado discreto, la temperatura $T(\tau)$ en una habitación evoluciona en el tiempo τ y es descrita por dos sistemas lineales distintos Σ_1 y Σ_2; en consecuencia, por dos ecuaciones diferenciales distintas $\dot{T}(\tau)=f_1(\cdot)$ y $\dot{T}(\tau)=f_2(\cdot)$.

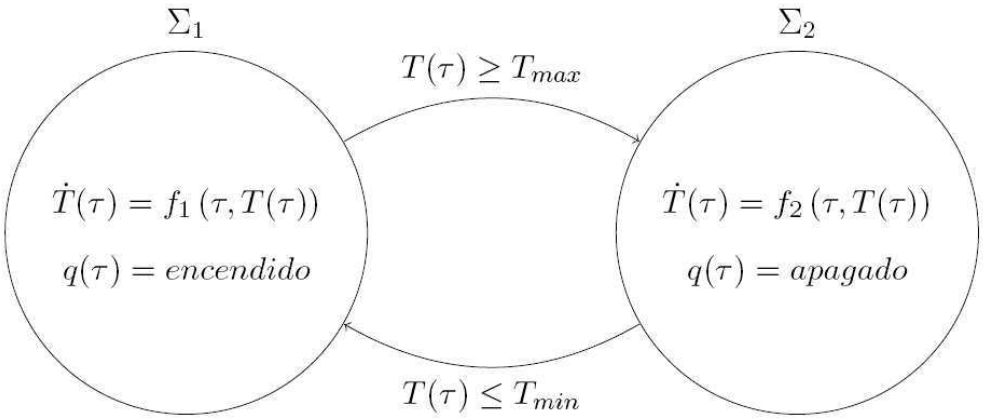


Figura 1. Sistema de temperatura modelado por un sistema híbrido lineal.

Este sistema tiene un estado híbrido $[q(\tau), T(\tau)]$, el estado discreto solo puede tomar dos estados $q(\tau)=\{encendido, apagado\}$ y el estado continuo $T(\tau)$ puede tomar valores reales, $T(\tau)\in\mathbb{R}$. Los cambios en el estado discreto son determinados por el estado continuo: si $T(\tau)=T_{\max}$ se conmuta al estado $apagado$; por otro lado si $T(\tau)=T_{\min}$ se conmuta al estado $encendido$, como se muestra en la figura 2.

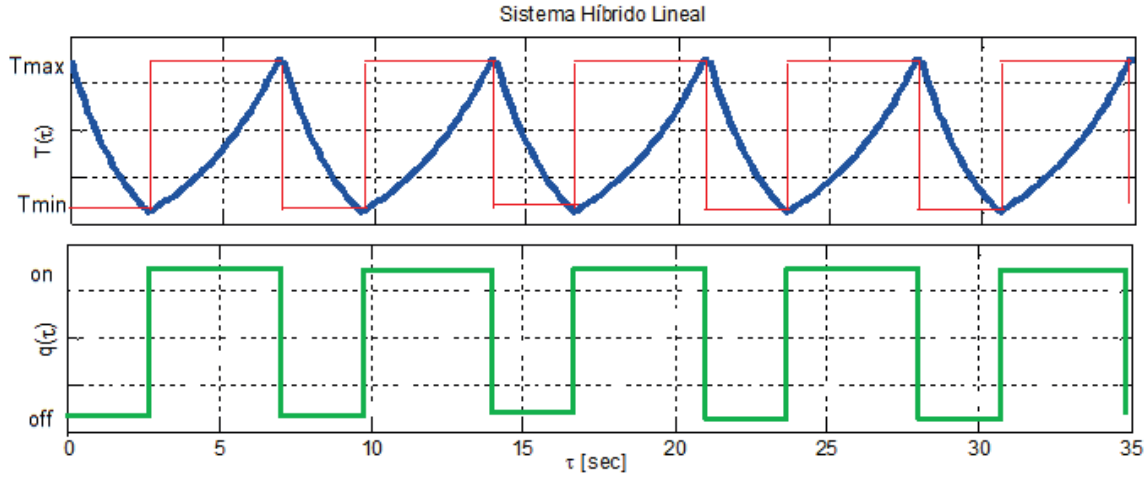


Figura 2. Dinámica del estado híbrido en el sistema de calefacción en una habitación.

Existen diferentes versiones de modelos de sistemas híbridos lineales que se distinguen respecto a su comportamiento y fenómenos [10], por mencionar algunos: autómata híbrido, sistemas híbridos conmutados (sistemas de tiempo continuo conmutados por eventos discretos [11]), sistemas

definidos a trozos, sistemas dinámicos lógicos mixtos, sistemas complementarios, autómatas temporizados [1], dichos modelos tienen distintos campos de aplicación. Como los sistemas híbridos lineales interactúan con dinámica en tiempo continuo (modelado por ejemplo por ecuaciones diferenciales) y dinámica de eventos discretos (modelado por ejemplo con autómatas y redes de Petri), las propiedades cualitativas en las que se han concentrado los estudios de los LHS es el modelado, estabilidad, controlabilidad, identificación y observabilidad. Por otro lado, también se estudia la conmutación entre los sistemas lineales continuos que es llevada a cabo por una señal de conmutación, que se puede clasificar en dependiente del estado continuo, dependiente del tiempo, autónomo (en el que se desconoce la regla que define la señal de conmutación) y controlado.

La red eléctrica convencional presenta cuatro componentes principales, plantas de generación, red de transmisión, red de distribución y control de energía eléctrica [13]. Por naturaleza, la red eléctrica convencional es un sistema con estructura electromecánica, con flujo de energía eléctrica unidireccional, de generación centralizada, tiene instalados pocos sensores, restauración, manipulación y control de forma manual, además, los consumidores tienen una pequeña y nula participación en la gestión y administración de la energía eléctrica [14]. La red convencional suministra la energía eléctrica desde plantas de generación, a un gran número de usuarios. Las principales plantas de generación son nucleares, geotérmicas, petróleo, hidroeléctrica, eólicas y plantas de carbón, las características de estas es que son centralizadas.

En contraste con la red eléctrica convencional, en una red inteligente su estructura es digital, soporta dos líneas de comunicación de electricidad e información (el suministro de energía eléctrica es distribuido y automatizado), la generación es descentralizada, se supervisa, controla y restaura de forma automática, además, los consumidores participan en la administración y gestión de la energía eléctrica [14]. La red inteligente se usa con mayor eficiencia y calidad en el suministro de energía eléctrica, mantener la sostenibilidad ambiental, flexibilidad en gestión de demanda eléctrica, presenta confiabilidad y seguridad que permite que la red sea observable, controlable y plenamente integrada a partir de numerosos sensores, es decir, proporciona la capacidad en forma segura de integrar fuentes de energía renovables [2,15], vehículos eléctricos y generadores distribuidos dentro de la red, figura 3.

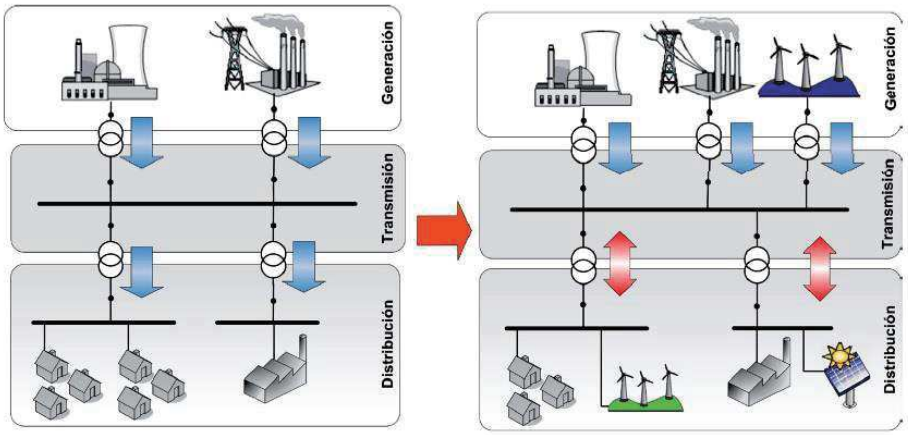


Figura 3. Red eléctrica convencional, b) Red eléctrica inteligente [16].

Una red eléctrica integrada de generadores distribuidos (fuentes de energías renovables, hidráulica, solar térmica, biomasa, solar, eólica, geotérmica, marica, entre otras) así como dispositivos de almacenamiento de energía los cuales suministran la demanda de energía eléctrica en forma local dentro de una región de cobertura establecida, se llama microred [3]. Si se presenta una falla o alguna situación de emergencia en una red eléctrica principal, una microred es capaz de seguir operando en modo autónoma y se garantiza el suministro de potencia eléctrica a las cargas conectadas en la microred, en el momento que la falla en la red eléctrica principal es restaurada la microred puede conectarse a la red eléctrica principal.

3. Metodología y Resultados

Una microred eléctrica opera con dos modos de operación, es decir, con dos estados discretos: conectado y autónomo $q(\tau)=\{conectado, autonomo\}$. En cada estado discreto, el voltaje entre el alambre y el plano de tierra $v(z,\tau)$ y la corriente que circula a través de él $i(z,\tau)$ evoluciona en el tiempo τ y desplazamiento de onda z, pueden tomar valores reales, $v,i\in\mathbb{R}$. Es descrita por cuatro sistemas no lineales distintos $\sum_1$, $\sum_2$, $\sum_3$ y $\sum_4$; en consecuencia por cuatro ecuaciones diferenciales no lineales distintas.

El sistema híbrido no lineal $\sum_1$ representa el sistema operando en modo conectado a la red principal con generadores distribuidos conectados a la microred, $\sum_3$ modo conectado en generadores distribuidos totalmente desconectados, $\sum_2$ representa el sistema operando en modo autónomo con generadores distribuidos totalmente conectados y $\sum_4$ representa el sistema operando en modo autónomo con generadores distribuidos desconectados, como se aprecia en la figura 3.

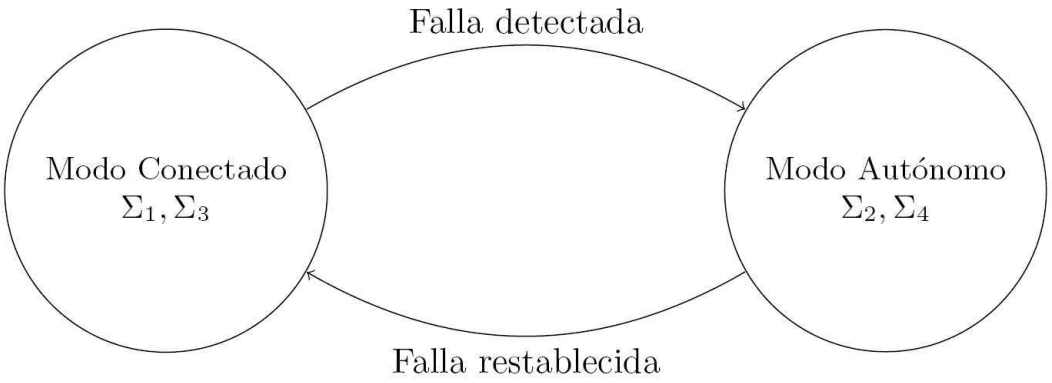


Figura 3. Modo de operación de una microred eléctrica inteligente.

Los cambios en el estado discreto son determinados por la detección de la falla en el sistema principal: si se detecta una falla cuando la microred se encuentra en modo conectado se conmuta al estado $q(\tau)=autonomo$; por otro lado, si se restablece la falla se conmuta al estado $q(\tau)=conectado$.

Propuesta de modelo en una microred eléctrica basado en un sistema híbrido ante una falla eléctrica detectada representa un sistema híbrido $\sum_\sigma$ con una colección de sistemas cuatro lineales no lineales $\Upsilon=\{\sum_1,\sum_2,\sum_3,\sum_4\}$ donde cada $\sum_i$ para $i=1,\ldots,4$ evoluciona en el espacio $\Im=\mathbb{R}^n$ y conmutadas por una señal σ tomadas valores en $\{conectada, autonomo\}$ que determina la evolución del sistema. La ecuación de estado $\sum_\sigma$ tiene la siguiente forma

$$\begin{aligned}\dot{x}&=A_\sigma(z,\tau)x+B_\sigma(z,\tau)y\\ y&=C_\sigma(z,\tau)x(\tau)+D_\sigma(z,\tau)y\\ \sigma&\in\{conectado, autonomo\}.\end{aligned}\qquad (2)$$

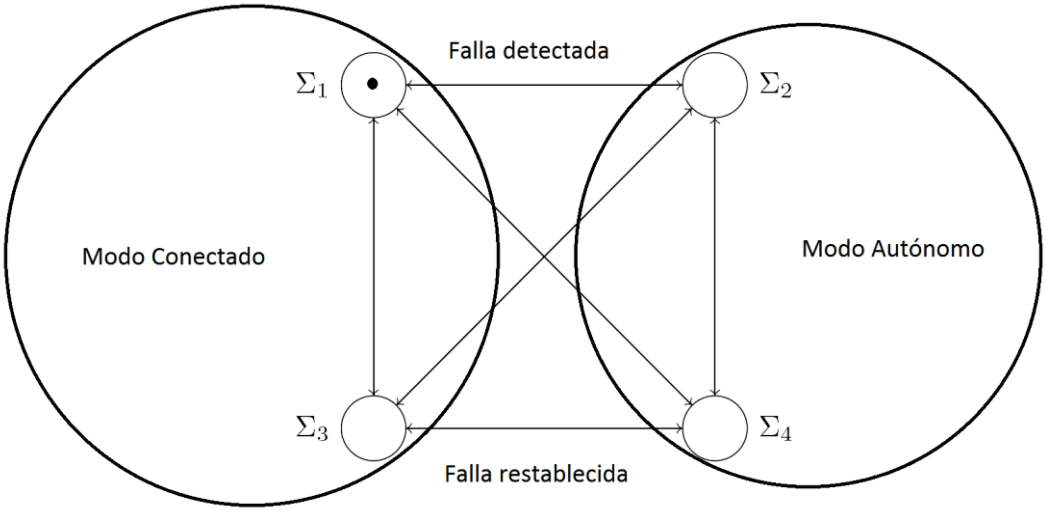


Figura 4. Modo de operación de una microred eléctrica inteligente.

donde $y = \begin{bmatrix} v(x,\tau) \\ i(x,\tau) \end{bmatrix} \in \mathbb{R}^r$ es la señal de la salida y es la misma salida en todo el sistema, la dinámica cambian dependiendo del modo que se encuentre operando la microred eléctrica. La evolución del sistema cuando $\sigma = i$ es representado por $\sum_i (A_i, B_i, C_i, D_i, E_i, F_i)$ o simplemente $\sum_i$ donde las matrices A_i, B_i, C_i, D_i, E_i y F_i son de dimensión apropiada. En tabla 1 se muestra la dinámica de una microred eléctrica basado en un sistema híbrido ante una falla eléctrica detectada.

Tabla 1. Dinámica de una microred eléctrica basado en un sistema híbrido ante una falla eléctrica detectada.

$\sum_i$		Generadores distribuidos conectados		Generadores distribuidos desconectados	
		$v(z,\tau)$	$i(z,\tau)$	$v(z,\tau)$	$i(z,\tau)$
$q(\tau)$	Conectado	$\sum_1$: $\dot{x} = A_\sigma(z,\tau)x + E_\sigma(z,\tau)y$ $y = C_\sigma(z,\tau)x(\tau) + F_\sigma(z,\tau)y$ $\sigma \in conectado.$		$\sum_2$: $\dot{x} = A_\sigma(z,\tau)x + B_\sigma(z,\tau)y$ $y = C_\sigma(z,\tau)x(\tau) + D_\sigma(z,\tau)y$ $\sigma \in conectado.$	
	Autónomo	$\sum_3$: $\dot{x} = A_\sigma(z,\tau)x + E_\sigma(z,\tau)y$ $y = C_\sigma(z,\tau)x(\tau) + F_\sigma(z,\tau)y$ $\sigma \in autonomo.$		$\sum_4$: $\dot{x} = A_\sigma(z,\tau)x + B_\sigma(z,\tau)y$ $y = C_\sigma(z,\tau)x(\tau) + D_\sigma(z,\tau)y$ $\sigma \in autonomo.$	

Si la microred eléctrica se encuentra operando en modo conectada a la red principal con generadores distribuidos conectados a la microred, figura 4, la dinámica de la salida $y = \begin{bmatrix} v(x,\tau) \\ i(x,\tau) \end{bmatrix}$ se encuentra representada por $\sum_1$ como

$$\begin{aligned} &\sum\nolimits_1 : \\ &\dot{x} = A_\sigma(z,\tau)x + E_\sigma(z,\tau)y \\ &y = C_\sigma(z,\tau)x(\tau) + F_\sigma(z,\tau)y \\ &\sigma \in conectado. \end{aligned} \tag{3}$$

donde las matrices $A_\sigma, C_\sigma, E_\sigma$ y F_σ son de dimensión apropiada.

La metodología presentada como propuesta para la dinámica de una microred eléctrica basado en un sistema híbrido ante una falla eléctrica detectada toma ventaja en el conocimiento de la señal de conmutación modelada por los dos estados que guarda el sistema, cuando dicho sistema opera en modo conectado y cuando opera de modo autónomo. La ventaja principal es que se pueden separar los estados observables y los estados no observables.

4. Conclusiones

En la actualidad, no existe un método universal para el modelado de las microredes eléctricas inteligentes ya que estos sistemas presentan no linealidades, incertidumbres o perturbaciones. Este modelo basado en sistemas híbridos permite desarrollar un método robusto basado en con enfoque de observadores para la detección e identificación de la falla presente en la microred. Sin embargo, aunque se sabe que la dinámica del estado continuo es una dinámica no lineal, se tiene un modelo general de la microred eléctrica. Donde se pueden estudiar las propiedades cualitativas en las que se han concentrado los estudios como es el modelado, estabilidad, controlabilidad, identificación y observabilidad. Por otro lado, también se estudia la conmutación entre los sistemas continuos (la detección de fallas).

Referencias

[1] J. Hare, X. Shi, S. Gupta, and A. Bazzi, \Fault diagnostics in smart micro-grids: A survey," Renewable and Sustainable Energy Reviews, vol. 60, pp. 1114{1124, 2016.
[2] I. Colak, S. Sagiroglu, G. Fulli, M. Yesilbudak, and C.-F. Covrig, \A survey on the critical issues in smart grid technologies," Renewable and Sustainable Energy Reviews, vol. 54, pp. 396{405, 2016.
[3] F. Li, W. Qiao, H. Sun, H. Wan, J. Wang, Y. Xia, Z. Xu, and P. Zhang, \Smart transmission grid: Vision and framework," IEEE transactions on Smart Grid, vol. 1, no. 2, pp. 168{177, 2010.
[4] M. Esreraig and J. Mitra, \An observer-based protection system for microgrids," in 2011 IEEE Power and Energy Society General Meeting, pp. 1{7, IEEE, 2011.
[5] B. Li, Y. Li, Z. Bo, and A. Klimek, \Design of protection and control scheme for microgrid systems," in Universities Power Engineering Conference (UPEC), 2009 Proceedings of the 44th International, pp. 1{5, IEEE, 2009.
[6] E. Sortomme, S. Venkata, and J. Mitra, \Microgrid protection using communication-assisted digital relays," IEEE Transactions on Power Delivery, vol. 25, no. 4, pp. 2789{2796, 2010.
[7] J. Zhang, Robust observer based fault diagnosis for nonlinear systems. PhD thesis, ResearchSpace@ Auckland, 2013.
[8] G. Buigues, A. Dy_sko, V. Valverde, I. Zamora, and E. Fern_andez, \Microgrid protection: Technical challenges and existing techniques," in International Conference on Renewable Energies and Power Quality (ICREPQ?13), 2013.

[9] E. Sortomme, G. Mapes, B. Foster, and S. Venkata, \Fault analysis and protection of a microgrid," in Power Symposium, 2008. NAPS'08. 40th North American, pp. 1{6, IEEE, 2008.
[10] H. Al-Nasseri, M. Redfern, and R. O'Gorman, \Protecting micro-grid systems containing solid-state converter generation," in 2005 international conference on future power systems, pp. 5{pp, IEEE, 2005.
[11] F. Z. Peng, Y. W. Li, and L. M. Tolbert, \Control and protection of power electronics interfaced distributed generation systems in a customer-driven microgrid," in 2009 IEEE Power & Energy Society General Meeting, pp. 1{8, IEEE, 2009.
[12] H. Nikkhajoei and R. H. Lasseter, \Microgrid protection," in Power Engineering Society General Mee-ting, 2007. IEEE, pp. 1{6, IEEE, 2007.
[13] X. Fang, S. Misra, G. Xue, and D. Yang, \Smart grid?the new and improved power grid: A survey," IEEE communications surveys & tutorials, vol. 14, no. 4, pp. 944{980, 2012.
[14] H. Farhangi, \The path of the smart grid," IEEE power and energy magazine, vol. 8, no. 1, pp. 18{28,2010.
[15] E. Ancillotti, R. Bruno, and M. Conti, \The role of communication systems in smart grids: Architectures, technical solutions and research challenges," Computer Communications, vol. 36, no.
[16] ANDRADE, Carlos Andrés Díaz; HERNÁNDEZ, Juan Carlos. Smart Grid: Las TICs y la modernización de las redes de energía eléctrica–Estado del Arte. Sistemas & Telemática, 2011, vol. 9, no 18, p. 53-81.

Ingeniería y Automatización, Capítulo 24, pp. 308 - 321.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Generación de Consignas con Perfil Quintico para el Control del Robot UR5

Noé Alfredo Martínez Sánchez[1], Erik Peña Medina[1], Patricio Martínez Zamudio[2], Octavio Diaz Hernández[1] y José Antonio Sousa Jiménez[2]

[1]Universidad Nacional Autónoma de México, Av. Universidad #3000, Ciudad de México, C.P. 04510, CDMX.
[2]Facultad de Estudios Superiores Aragón, Universidad Nacional Autónoma de México, Av. Hacienda de Rancho Seco S/N, Impulsora Popular Avícola, 57130 Nezahualcóyotl, Méx.
Correo de contacto: mecatronicanoe@gmail.com

Resumen

El requerimiento mínimo para un robot es la capacidad de moverse de una postura inicial a una postura final. La transición debe estar caracterizada por una ley de movimiento para que los actuadores generen las fuerzas en el espacio de juntas que no violen los límites de saturación y no excite la resonancia del sistema. En este trabajo se presenta la generación de consignas para el control del robot UR5 (Universal Robots) utilizando un perfil quintico. El modelado cinemático directo del robot UR5 se obtiene utilizando Denavit-Hartenberg, el modelo cinemático inverso se resuelve por método numérico, el modelo diferencial se obtiene con el método de propagación de velocidades y el modelo dinámico con la formulación de Lagrange. Se presenta el perfil quintico en el espacio de trabajo del robot, se propone una trayectoria y lo que se obtiene con el perfil se mapea al espacio de juntas mediante las funciones inversas del modelo cinemático. Con las consignas en el espacio de juntas obtenidas se evalúa la dinámica del robot UR5 a través del control dinámico par-calculado, la simulación realizada en Matlab muestra un movimiento suave en el robot. Se muestra que el uso del perfil quintico y de las funciones inversas de la cinemática para la generación de trayectorias de un robot es útil para planear movimientos suaves.

Palabras clave: Cinemática, Cinemática diferencial, Perfil Quintico, Dinámica, Control Par-Calculado, Universal Robots.

1. Introducción

Hay un gran incremento en el uso de robots a nivel mundial, de estos los manipuladores seriales son usados en muchos sistemas industriales y de investigación[1]. Por un lado, en las aplicaciones industriales se buscan aspectos de funcionalidad como repetibilidad, rapidez y capacidad de carga, mientras que el ámbito científico se concentra en resolver aspectos teóricos como el modelado dinámico, planeación de movimiento, esquemas de control novedosos, entre otros[2]. Uno de los aspectos más relevantes a responder por los investigadores es la planeación de la trayectoria durante la realización de una tarea por el robot. La planeación de la trayectoria se encarga de describir el movimiento deseado de un manipulador en espacio multidimensional.

Actualmente, el movimiento de un robot es típicamente especificado en el espacio operacional en términos de la trayectoria usando un punto inicial y punto final del efector final, mientras que los puntos intermedios y tiempo de ejecución a lo largo de la trayectoria, son determinados automáticamente y normalmente se usan modelos lineales para obtener dichos puntos intermedios. En este artículo se presenta un enfoque de planeación de trayectorias suaves para tareas de tomar y colocar (*pick and place*) en robots seriales usando curvas *spline*, lo cual resulta en un movimiento más natural y que puede mejorar la dinámica y por tanto el gasto de energía.

La trayectoria en un robot se define como un conjunto de puntos (en el espacio articular o en el espacio de trabajo) sobre los que se ha definido una ley temporal [3]. Este conjunto de puntos da como resultado una señal de referencia para el controlador. Existen dos esquemas para generar trayectorias: en el espacio articular y en el espacio cartesiano. En la primera la trayectoria se describe en términos de funciones de ángulos de articulación mientras la segunda se describe en términos de funciones que calculan la posición y orientación cartesianas. Los esquemas en el espacio de articulación son generalmente los más fáciles de calcular y, como no hay una correspondencia continua entre el espacio de articulación y el espacio cartesiano, esencialmente no hay ningún problema con las singularidades del mecanismo. En [4] consideran el problema de control de trayectorias de un robot serial, para eso usan un interpolador cubico en el espacio de juntas. Realizan una optimización considerando restricciones de torque y velocidad. En [5] realizan una planeación de trayectorias para un robot industrial utilizando *splines*, proponen un *spline* cubico para la optimización del tiempo y suprimen la vibración que se produce del movimiento del robot. En [6] proponen un nuevo algoritmo basado en una curva *spline* de cuaternion para realizar la interpolación de la orientación del efector final del robot, con esto satisfacen los movimientos suaves requeridos en aplicaciones industriales. En [7] presentan un enfoque para planeación de trayectorias libre de colisiones para dos robots que comparten el mismo espacio de trabajo. El enfoque utiliza un *spline* que solo cambia la trayectoria local alrededor del área de colisión sin cambiar la tarea de forma global. En [8] describen el un algoritmo de control de aprendizaje que mejora la precisión de un robot industrial. Estiman la dinámica del robot y obtienen una matriz de ganancias del controlador, disminuyendo el error de seguimiento mediante iteraciones. En [9] plantean un método de planeación que genera trayectorias continuas y acotadas para el *jerk* en el espacio de juntas. El algoritmo permite definir independientemente la posición, velocidad, aceleración y *jerk*.

Ya que el control de un robot es llevado a cabo en el espacio de juntas, las funciones inversas de la cinemática son usadas para reconstruir la secuencia de tiempo de las variables de juntas que corresponden a la secuencia en el espacio operacional. En este trabajo se utiliza un perfil quintico con condiciones iniciales y finales cero para describir la trayectoria del efector final en el espacio de juntas. Después con las funciones inversas de la cinemática, esto es: cinemática inversa, velocidad inversa y aceleración inversa, se obtiene la descripción de la trayectoria en el espacio de juntas. Con lo anterior se generan las consignas de control para el robot UR5.

El resto del artículo está organizado como sigue: en la sección 2 se presenta el modelado cinemático directo, inverso y la cinemática diferencial del robot UR5. En la sección 3 se presenta el perfil quintico y se realiza una simulación donde se muestran las gráficas de la trayectoria en el espacio operacional y el espacio de juntas. En la sección 4 se presenta el modelo dinámico en el espacio de juntas obtenido por la formulación de Lagrange y también se presenta la simulación del control Par-calculado utilizando las referencias obtenidas en el apartado anterior. Finalmente, en la sección 5 se muestran las conclusiones de este trabajo.

2. Modelo cinemático del robot UR5

2.1. Modelo cinemático directo

En la figura 1 se muestra el robot UR5 de 6 grados de libertad construido por Universal Robots[10]. Dentro de las características más relevantes del robot UR5 es la agilidad debido a su poco peso y velocidad. También, se caracteriza por tener los tres últimos ejes no coincidentes en un punto. Por lo tanto, las 6 juntas del robot producen un cambio en la posición y orientación del efector final. Esto hace que el análisis cinemático sea más complicado con respecto a la mayoría de los robots con los últimos tres ejes coincidentes en un punto.

La cinemática describe el movimiento del robot sin considerar las fuerzas y momentos causados por el movimiento. La posición y orientación del efector final se expresa con el vector:

Ingeniería y Automatización, Capítulo 24, pp. 308 - 321.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

$$x = \left(x_e, y_e, z_e, \alpha_e, \beta_e, \gamma_e\right)^T \tag{1}$$

donde $x \in R^6$ es el vector de posiciones y orientaciones cartesianas con x_e, y_e, z_e expresando posición y $\alpha_e, \beta_e, \gamma_e$ la orientación del efector final.

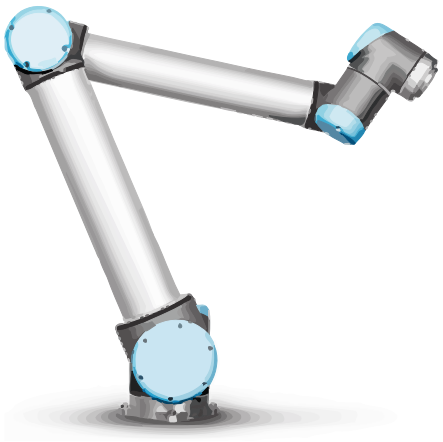


Figura 1. Robot UR5

Las coordenadas articulares del robot son:

$$q = \left(q_1, q_2, q_3, q_4, q_5, q_6\right)^T \tag{2}$$

La relación entre las coordenadas articulares y cartesianas del robot manipulador define propiamente el modelo cinemático directo.

$$x = f(q) \tag{3}$$

donde $f(q) \in R^6$. En la figura 2 se muestra un esquema del robot que contiene la localización de los sistemas de referencias para cada junta. La colocación de los sistemas de referencia se realizó de acuerdo al método Denavit-Hartenberg [11]. En la tabla 1 se muestran los parámetros del robot:

Tabla 1.- Parámetros Denavit-Hartenberg robot UR5

i	α_{i-1}	a_{i-1}	d_i	q_i
${}^0A_1$	0	0	l_1	q_1
${}^1A_2$	$\pi/2$	0	l_2	$q_2 + \pi/2$
${}^2A_3$	0	l_3	$-l_2$	q_3
${}^3A_4$	0	l_4	l_2	$q_4 - \pi/2$
${}^4A_5$	$-\pi/2$	0	l_5	q_5
${}^5A_6$	$\pi/2$	0	0	q_6
${}^6A_7$	0	0	l_6	0

La cinemática directa se obtendrá por la multiplicación sucesiva de todas las matrices Denavit-Hartenberg de la tabla 1. Ahora con la definición de la matriz homogénea[12], la matriz de transformación que expresa la ecuación 3 es:

$$ {}^{0}T_{n} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

$$ n_x = -c_6(s_1 s_5 - c_1 c_2 c_3 c_4 c_5 + c_1 c_2 c_5 s_3 s_4 + c_1 c_3 c_5 s_2 s_4 + c_1 c_4 c_5 s_2 s_3) - c_1 s_6 s_{234} $$

$$ n_y = -c_6(c_2 c_5 s_1 s_3 s_4 - c_2 c_3 c_4 c_5 s_1 - c_1 s_5 + c_3 c_5 s_1 s_2 s_4 + c_4 c_5 s_1 s_2 s_3) - s_1 s_6 s_{234} $$

$$ n_z = \frac{s_{2346}}{2} - \frac{s_{234_6}}{2} + c_5 c_6 s_{234} $$

$$ o_x = s_6(s_1 s_5 - c_1 c_2 c_3 c_4 c_5 + c_1 c_2 c_5 s_3 s_4 + c_1 c_3 c_5 s_2 s_4 + c_1 c_4 c_5 s_2 s_3) - c_1 c_6 s_{234} $$

$$ o_y = s_6(c_2 c_5 s_1 s_3 s_4 - c_2 c_3 c_4 c_5 s_1 - c_1 s_5 + c_3 c_5 s_1 s_2 s_4 + c_4 c_5 s_1 s_2 s_3) - c_6 s_1 s_{234} $$

$$ o_z = \frac{c_{2346}}{2} + \frac{c_{234_6}}{2} - c_5 s_6 s_{234} $$

$$ a_x = c_5 s_1 + c_1 c_2 c_3 c_4 s_5 - c_1 c_2 s_3 s_4 s_5 - c_1 c_3 s_2 s_4 s_5 - c_1 c_4 s_2 s_3 s_5 $$

$$ a_y = c_2 c_3 c_4 s_1 s_5 - c_1 c_5 - c_2 s_1 s_3 s_4 s_5 - c_3 s_1 s_2 s_4 s_5 - c_4 s_1 s_2 s_3 s_5 $$

$$ a_z = \frac{c_{234_5}}{2} - \frac{c_{2345}}{2} $$

$$ p_x = l_2 s_1 - c_1 l_3 s_2 + c_5 l_6 s_1 - c_1 c_2 l_4 s_3 - c_1 c_3 l_4 s_2 - c_1 c_2 c_3 l_5 s_4 - c_1 c_2 c_4 l_5 s_3 - c_1 c_3 c_4 l_5 s_2 + c_1 l_5 s_2 s_3 s_4 + c_1 c_2 c_3 c_4 l_6 s_5 $$
$$ -c_1 c_2 l_6 s_3 s_4 s_5 - c_1 c_3 l_6 s_2 s_4 s_5 - c_1 c_4 l_6 s_2 s_3 s_5 $$

$$ p_y = l_5 s_1 s_2 s_3 s_4 - c_1 c_5 l_6 - l_3 s_1 s_2 - c_2 l_4 s_1 s_3 - c_3 l_4 s_1 s_2 - c_2 c_3 l_5 s_1 s_4 - c_2 c_4 l_5 s_1 s_3 - c_3 c_4 l_5 s_1 s_2 - c_1 l_2 + c_2 c_3 c_4 l_6 s_1 s_5 $$
$$ -c_2 l_6 s_1 s_3 s_4 s_5 - c_3 l_6 s_2 s_2 s_4 s_5 - c_4 l_6 s_1 s_2 s_3 s_5 $$

$$ p_z = l_1 + c_2 l_3 - \left(\frac{l_6 c_{2345}}{2} + l_4 c_{23} + \frac{l_6 c_{234_5}}{2} \right) + l_5 c_{234} $$

Ingeniería y Automatización, Capítulo 24, pp. 308 - 321.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

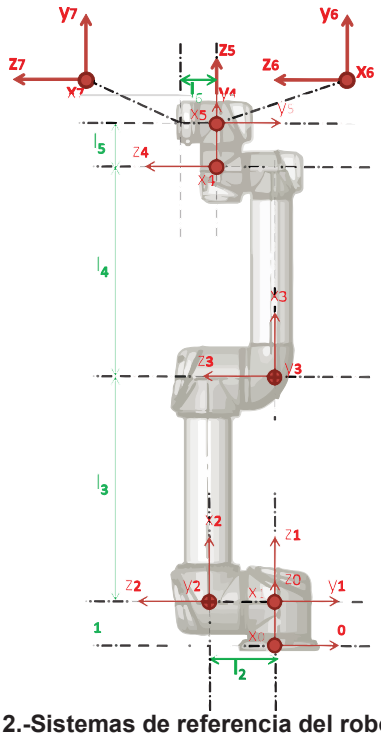


Figura 2.-Sistemas de referencia del robot UR5

2.2. Modelo cinemático inverso

El objetivo del problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $q=(q_1,q_2,q_3,q_4,q_5,q_6)^T$ para que el efector final se posicione y oriente según una determinada localización espacial $x=(x_e,y_e,z_e,\alpha_e,\beta_e,\gamma_e)^T$.

$$q=f(q)^{-1}x \tag{5}$$

Existen diversas soluciones para la cinemática inversa, por ejemplo: métodos geométricos, a partir de la matriz de transformación homogénea, desacoplo cinemático, métodos iterativos[13]. La cinemática inversa por método geométrico para el robot UR5 ha sido formulada y ampliamente utilizada[14, 15]. Dado que la ecuación 4 está compuesta por ecuaciones no-lineales en este trabajo se utiliza un método numérico para solucionar la cinemática inversa al ser de fácil implementación en Matlab. Para formar el sistema de ecuaciones a resolver primero se describirá la orientación y posición del efector final. La especificación de la orientación del efector final se realiza con los ángulos RPY (Roll-Pitch-Yaw)[3]. En este caso, los tres ángulos son α,β,γ y representan la orientación del efector final con respecto al marco inercial. La composición de las tres matrices de rotación viene dada por:

$$R_\phi=R_z(\gamma)R_y(\beta)R_x(\alpha)=\begin{bmatrix} c_\alpha c_\beta & c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta c_\gamma + s_\alpha s\gamma \\ s_\alpha c_\beta & s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta c_\gamma - c_\alpha s\gamma \\ -s_\beta & c_\beta s_\gamma & c_\beta c_\gamma \end{bmatrix} \tag{6}$$

Una vez que se tiene la descripción de la orientación, se agrega la posición del efector final con respecto al sistema absoluto y esta se iguala con la ecuación 4:

$$ {}^{0}A_1\,{}^{1}A_2\,{}^{2}A_3\,{}^{3}A_4\,{}^{4}A_5\,{}^{5}A_6\,{}^{6}A_7 = transl(x_e, y_e, z_e) R_z(\gamma_e) R_y(\beta_e) R_x(\alpha_e) \tag{7} $$

De la ecuación 7 se tienen 12 ecuaciones no lineales y 6 variables a resolver, esto quiere decir que existen varias soluciones para la cinemática inversa. En este trabajo se utilizan las ecuaciones (1,4), (2,4), (3,4) y la diagonal de la submatriz de rotación de la ecuación 7 para resolver la cinemática inversa.

2.3. Cinemática diferencial

Para encontrar el modelo cinemático en velocidades del manipulador móvil, se utiliza la técnica de propagación de velocidades[11], ${}^{i+1}v_{i+1} = {}^{i+1}_{i}R({}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1})$ para la velocidad lineal y ${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega_i + \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1}$ para la velocidad angular. Como el robot está anclado al piso, la velocidad linear y angular de entrada del sistema inercial son cero, es decir: ${}^{0}v_0 = (0,0,0)^T$ y ${}^{0}\omega_0 = (0,0,0)^T$. Los valores de ${}^{i+1}_{i}R$, ${}^{i}P_{i+1}$ son obtenidos de la tabla 1 y ${}^{i+1}\hat{Z}_{i+1} = (0,0,1)^T$.

La última propagación de velocidades es la ${}^{7}v_7$, ${}^{7}\omega_7$, esta representa la velocidad del efector final vista desde el efector final (ver anexo 1). Para obtener el jacobiano del robot referenciado al sistema inercial, se multiplica la última propagación por la transformación ${}^{0}R_7 = {}^{i}R_{i+1}, ..., {}^{n-1}R_n$ formada por las submatrices de rotación de la tabla 1, ${}^{0}v_7 = {}^{0}R_7\,{}^{7}v_7$, ${}^{0}\omega_7 = {}^{0}R_7\,{}^{7}\omega_7$. Con esto la velocidad del efector final queda referida al sistema inercial, importante para la definición de la trayectoria. Ahora el jacobiano se obtiene como:

$$ \begin{bmatrix} {}^{0}v_7 \\ {}^{0}\omega_7 \end{bmatrix} = {}^{0}\upsilon_7 = {}^{0}J_7(q)\dot{q} \tag{8} $$

Entonces la cinemática diferencial directa queda expresada como:

$$ \dot{x} = J(q)\dot{q} \tag{9} $$

Por simplicidad ${}^{0}J_7 = J$, la ecuación 9 permite encontrar la velocidad del efector final $\dot{x}$ dadas las velocidades de las juntas $\dot{q}$. La relación inversa de la ecuación 9 es la siguiente:

$$ \dot{q} = J(q)^{-1}\dot{x} \tag{10} $$

Esto siempre y cuando la matriz jacobiana J se pueda invertir, en otro caso se utiliza la pseudo-inversa[3]. Derivando la ecuación 9 se obtiene:

$$ \ddot{x} = \dot{J}(q)\dot{q} + J(q)\ddot{q} \tag{11} $$

Esta ecuación permite conocer la aceleración $\ddot{x}$ del efector final, dadas las posiciones q, velocidades $\dot{q}$ y aceleraciones $\ddot{q}$ de las juntas. La relación inversa se obtiene sustituyendo la ecuación 10 en la ecuación 11 y despejando $\ddot{q}$.

$$ \ddot{q} = J(q)^{-1}\ddot{x} - J(q)^{-1}\dot{J}(q)J(q)^{-1}\dot{x} \tag{12} $$

Ingeniería y Automatización, Capítulo 24, pp. 308 - 321.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Diseño de trayectoria con perfil quintico

El movimiento de un cuerpo en el espacio consiste en dos partes: una trayectoria lineal o curva en el espacio que sigue un punto del cuerpo (en el centro de gravedad o en el efector final del manipulador) y la orientación angular del cuerpo. El perfil quintico es conocido en la literatura de robótica, en este apartado solo se muestra el resultado del perfil quintico. Para definir la trayectoria cartesiana del efector final se utilizó un perfil quintico con condiciones de velocidad y aceleración inicial y final igual a cero, para la posición se tiene:

$$R(t) = p_i + \left(10\frac{t}{tf}^3 - 15\frac{t}{tf}^4 + 6\frac{t}{tf}^5 \right)\left(p_f - p_i \right) \tag{13}$$

$$V(t) = \left(30\frac{t^2}{tf^3} - 60\frac{t^3}{tf^4} + 30\frac{t^4}{tf^5} \right)\left(p_f - p_i \right) \tag{14}$$

$$A(t) = \left(60\frac{t}{tf^3} - 180\frac{t^2}{tf^4} + 120\frac{t^3}{tf^5} \right)\left(p_f - p_i \right) \tag{15}$$

donde p_i, p_f definen posiciones iniciales y finales.

Para la orientación se sigue un procedimiento similar, en el entendido de que, para este caso, solo se desea pasar de valores iniciales a finales, para la posición, velocidad y aceleración angular de la plataforma móvil, ya que no se requiere cumplir con una trayectoria particular en el espacio. Esto conducirá a las siguientes ecuaciones:

$$\phi(t) = \beta_i + \left(10\frac{t}{tf}^3 - 15\frac{t}{tf}^4 + 6\frac{t}{tf}^5 \right)\left(\phi_f - \phi_i \right) \tag{16}$$

$$\dot{\phi}(t) = \left(30\frac{t^2}{tf^3} - 60\frac{t^3}{tf^4} + 30\frac{t^4}{tf^5} \right)\left(\phi_f - \phi_i \right) \tag{17}$$

$$\ddot{\phi}(t) = \left(60\frac{t}{tf^3} - 180\frac{t^2}{tf^4} + 120\frac{t^3}{tf^5} \right)\left(\phi_f - \phi_i \right) \tag{18}$$

Donde el vector $\phi = (\alpha, \beta, \gamma)$. De la misma manera $\phi_i = (\alpha_i, \beta_i, \gamma_i)$ y $\phi_f = (\alpha_f, \beta_f, \gamma_f)$, que se refieren a los valores iniciales y finales.

3.1. Simulación perfil quintico

Con el fin de comprobar el funcionamiento del perfil quintico se propone una trayectoria a seguir. La trayectoria está compuesta por tres líneas las cuales forman un triángulo (figura 5). Primera línea con el punto inicial en $p_{i1} = \begin{pmatrix} -0.2 & 0.4 & 0.4 \end{pmatrix}$ y punto final $p_{f1} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \end{pmatrix}$ con la orientación inicial $\phi_{i1} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$ y final $\phi_{f1} = \begin{pmatrix} -0.2618 & -0.1745 & -0.0873 \end{pmatrix}$. Segunda línea con el punto inicial en $p_{i2} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \end{pmatrix}$ y punto final $p_{f2} = \begin{pmatrix} 0.0 & 0.4 & 0.7464 \end{pmatrix}$ con la orientación inicial $\phi_{i2} = \begin{pmatrix} -0.2618 & -0.1745 & -0.0873 \end{pmatrix}$ y final $\phi_{f2} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$. Tercera línea con el punto inicial en $p_{i3} = \begin{pmatrix} 0 & 0.4 & 0.7464 \end{pmatrix}$ y punto final $p_{f3} = \begin{pmatrix} -0.2 & 0.4 & 0.4 \end{pmatrix}$ con la orientación inicial $\phi_{i3} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$ y final $\phi_{f3} = \begin{pmatrix} 0.2618 & 0.1745 & 0.0873 \end{pmatrix}$. Cada línea se recorrerá en un tiempo de 5 segundos, de manera que el triángulo se realiza en 15 segundos. Sustituyendo estos puntos en las ecuaciones 13-18 se

generan la descripción de la trayectoria en el espacio cartesiano (figura 3). El resultado anterior se sustituye en las ecuaciones 5, 10 y 12 obteniéndose la descripción de la trayectoria en el espacio de juntas (figura 4).

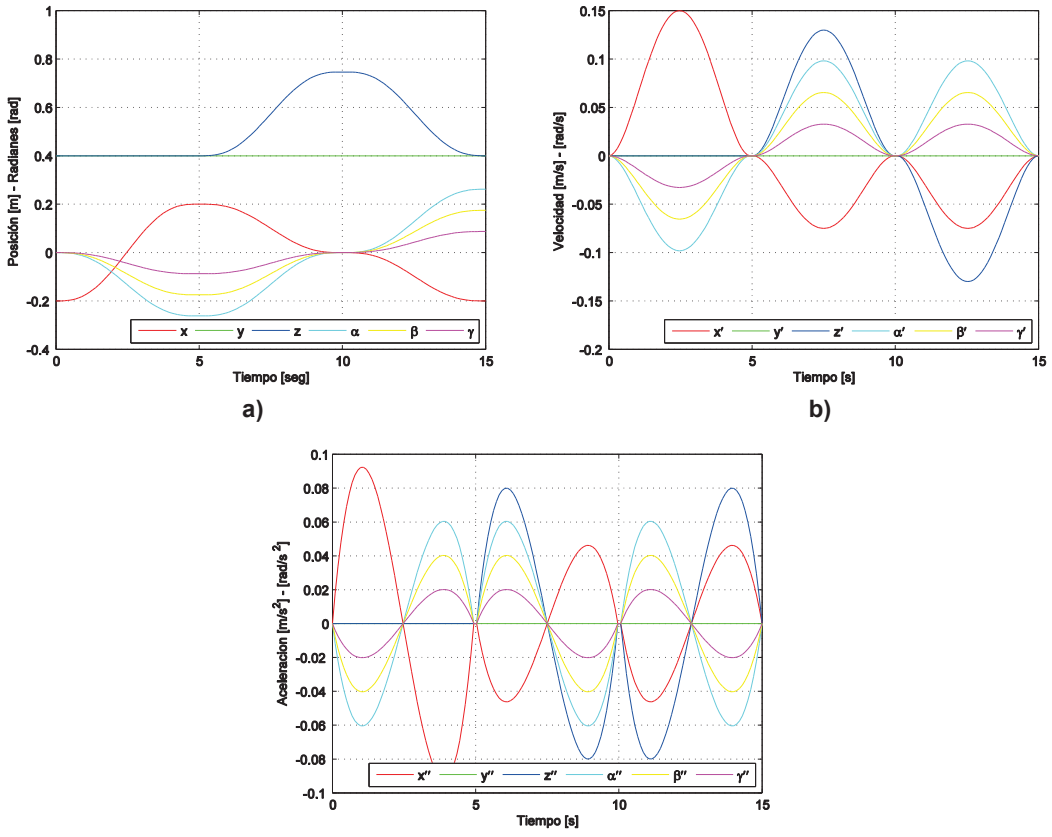
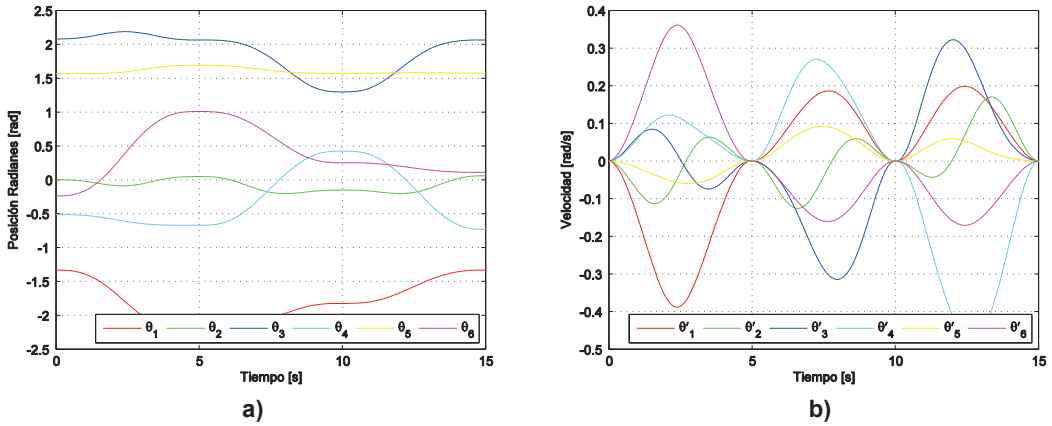
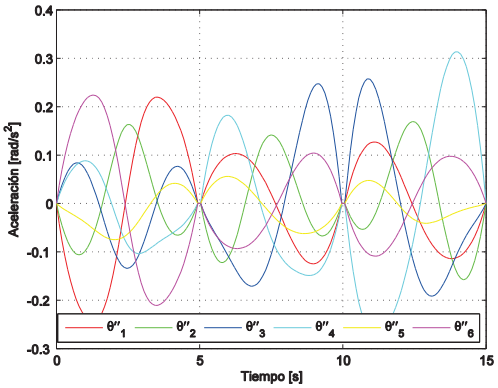


Figura 3. a) Posiciones (x_e, y_e, z_e) y orientaciones $(\alpha_e, \beta_e, \gamma_e)$ cartesianas del efector final, b) Velocidades lineales $(\dot{x}_e, \dot{y}_e, \dot{z}_e)$ y velocidades angulares $(\dot{\alpha}_e, \dot{\beta}_e, \dot{\gamma}_e)$ cartesianas del efector final c) Aceleraciones lineales $(\ddot{x}_e, \ddot{y}_e, \ddot{z}_e)$ y aceleraciones angulares $(\ddot{\alpha}_e, \ddot{\beta}_e, \ddot{\gamma}_e)$ cartesianas del efector final.





c)

Figura 4. a) Posiciones articulares $q=\left(q_1,q_2,q_3,q_4,q_5,q_6\right)^T$ **, b) Velocidades articulares** $\dot{q}=\left(\dot{q}_1,\dot{q}_2,\dot{q}_3,\dot{q}_4,\dot{q}_5,\dot{q}_6\right)^T$ **c) Aceleraciones articulares** $\ddot{q}=\left(\ddot{q}_1,\ddot{q}_2,\ddot{q}_3,\ddot{q}_4,\ddot{q}_5,\ddot{q}_6\right)^T$ **.**

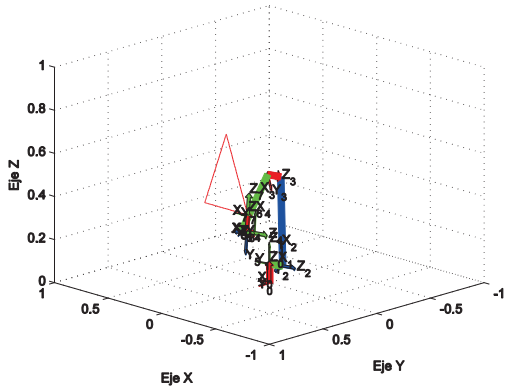


Figura 5. Trayectoria trazada por el robot.

4. Modelo dinámico y control del robot UR5

Considérese el robot manipulador que se compone de n eslabones mostrado en la figura 2. La energía total E de un robot manipulador de n GDL es la suma de sus energías cinética K y potencial U :

$$E(q,\dot{q})=K(q,\dot{q})+U(q) \tag{19}$$

donde $q=[q_1,...,q_n]^T$.

La función lagrangiana $L(q,\dot{q})$ de un robot manipulador de n GDL es la diferencia entre su energía cinética K y su energía potencial U ;

$$L(q,\dot{q})=K(q,\dot{q})-U(q) \tag{20}$$

Las ecuaciones de movimiento de Lagrange de un robot serial es comúnmente escrita en la forma [16]:

$$\frac{d}{dt}\left[\frac{\partial L(q,\dot{q})}{\partial \dot{q}_i}\right]-\frac{\partial L(q,\dot{q})}{\partial q_i}=\tau_i; i=1,...,n \tag{21}$$

donde las variables q_i constituyen el conjunto de coordenadas generalizadas del sistema, τ_i son las fuerzas generalizadas de la junta i (i.e., torque para las juntas rotacionales y fuerza para las juntas prismáticas) y representa el torque que puede potencialmente ser aplicado por los motores del robot UR5.

4.1. Ecuación dinámica de movimiento en el espacio de juntas

La ecuación dinámica de movimiento en el espacio de juntas del robot UR5 tiene la siguiente forma:

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)+f(\dot{q})=\tau \tag{22}$$

donde $M(q)\in R^{q\times q}$ es la matriz de inercia, $C(q,\dot{q})\dot{q}\in R^{q\times 1}$ es el vector de fuerzas de coriolis y centrípetas, $g(q)\in R^{q\times 1}$ es el vector de fuerzas gravitacionales, $\tau\in R^{q\times 1}$ es el vector de entrada. Las propiedades de la ecuación dinámica pueden resumirse como:
Propiedad 1. $M(q)$ es simétrica y positiva definida.
Propiedad 2. $\dot{M}(q)-2C(q,\dot{q})$ es antisimetrica.

4.2. Control Par-calculado

El problema de control de movimiento de robots manipuladores puede formularse en los siguientes términos. Considérese una ecuación dinámica de un robot de n g.d.l. (ecuación 22). Dado un conjunto de funciones vectoriales acotadas q_d, $\dot{q}_d$ y $\ddot{q}_d$ referidas como posiciones, velocidades y aceleraciones articulares deseadas, se trata de determinar una función vectorial τ, de tal forma que las posiciones q asociadas a las coordenadas articulares del robot sigan con precisión a q_d [17]. En términos más formales, el objetivo de control consiste en determinar τ, de tal forma que:

$$\lim_{t\to\infty}\tilde{q}(t)=0 \tag{23}$$

donde $\tilde{q}\in R^n$ denota el vector de errores articulares, simplemente denominado error de posición, y definido como

$$\tilde{q}(t)=q_d(t)-q(t) \tag{24}$$

Considerando la definición anterior, el vector $\dot{\tilde{q}}(t)=\dot{q}_d(t)-\dot{q}(t)$ denotara el error de velocidad. Si el objetivo de control se verifica, significara que las articulaciones del robot manipulador siguen asintóticamente la trayectoria de movimiento deseado. El cálculo del vector τ involucra generalmente una función vectorial no lineal de q, $\dot{q}$ y $\ddot{q}$, esta función se denomina “Ley de control”. Genéricamente, el controlador puede expresarse como:

$$\tau=\tau(q,\dot{q},\ddot{q},q_d,\dot{q}_d,\ddot{q}_d,M(q),C(q,\dot{q}),g(q)) \tag{25}$$

El control Par-Calculado es una de las estructuras de control de movimiento basabas en el modelo del robot a ser controlado, es decir, se hace uso explícito del conocimiento del modelo dinámico [18]. La ecuación correspondiente al control Par-Calculado viene dada por:

$$\tau=M(q)^{-1}\left[\ddot{q}+K_p e(t)+K_v\dot{e}(t)\right]+C(q,\dot{q})\dot{q}+G(q)+f(\dot{q}) \tag{26}$$

Donde K_p, K_v son matrices definidas positivas y son las ganancias del controlador, $e(t) \in R^n$ denota el vector de errores de posiciones articulares, simplemente denominado error de posición $e(t) = q_d(t) - q(t)$, el vector $\dot{e}(t) = \dot{q}_d(t) - \dot{q}(t)$ denotara el error de velocidad. La ley de control Par-Calculado linealiza y desacopla la dinámica del robot, las no linealidades como Coriolis y términos centrífugos así como los términos de gravedad se pueden compensar agregando estas fuerzas en el controlador. Los parámetros del robot para esta simulación se presentan en el Anexo 2. En la figura 6 a se muestran los errores de posición de las juntas articulares, esta muestra que existe un error de 0.1 radianes en las juntas 4, 5 y 6. Por otro lado el error de posición de las de primeras juntas es muy pequeño menos de 0.5 radianes. La figura 6 b muestra los errores de velocidad de las juntas, la figura 6 c muestra los torques requeridos para que cada junta realice el movimiento deseado. La figura 6 d muestra el seguimiento en coordenadas cartesianas de la trayectoria, se aprecia que el error es mínimo, el efector final sigue a la referencia.

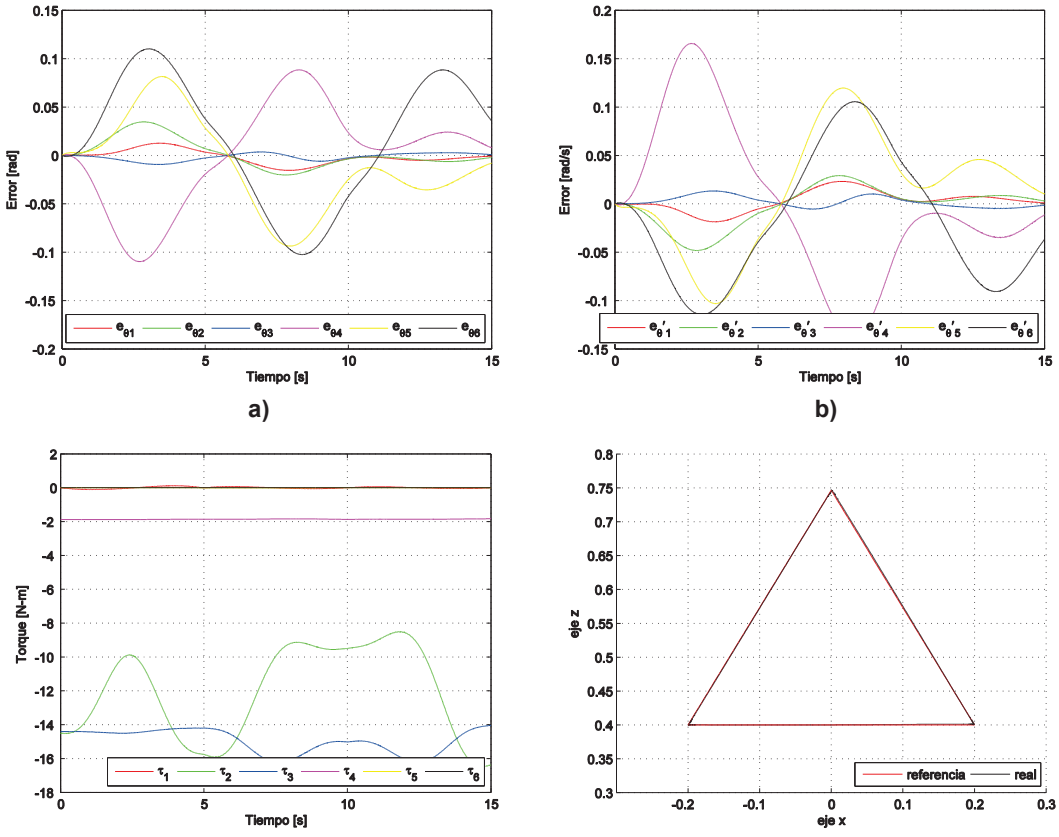


Figura 6. a) Errores de posición articular, b) Errores de velocidad articular, c) Torques, d) referencia (triangulo rojo) vs real (triangulo negro).

5. Conclusiones

En este trabajo se presentó la generación de consignas para el control del robot UR5 utilizando un perfil quintico. Lo anterior fue posible utilizando los modelos cinemáticos inversos del robot, con estos se generaron las consignas para el control dinámico. La generación de consignas para el control de robots es importante ya que debe tenerse en consideración la dinámica del robot. Se deben de cumplir ciertas condiciones tales como: tener en cuenta las limitaciones de los motores del robot, las singularidades, movimientos suaves de inicio y final de la trayectoria. Con el perfil

quintico se mostró que el movimiento resultante del robot es suave y que un control general como el par calculado no tiene problemas con las consignas generadas.

Anexo 1. Propagación de velocidades

$$ {}^{1}v_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},\ {}^{1}\omega_1 = \begin{pmatrix} 0 \\ 0 \\ \dot{q}_1 \end{pmatrix},\ {}^{2}v_2 = \begin{pmatrix} -l_2\dot{q}_1 s_2 \\ -l_2\dot{q}_1 c_2 \\ 0 \end{pmatrix},\ {}^{2}v_2 = \begin{pmatrix} \dot{q}_1 c_2 \\ -\dot{q}_1 s_2 \\ \dot{q}_2 \end{pmatrix},\ {}^{3}v_3 = \begin{pmatrix} l_3\dot{q}_2 s_3 \\ l_3\dot{q}_2 c_3 \\ l_3\dot{q}_1 s_2 \end{pmatrix},\ {}^{3}\omega_3 = \begin{pmatrix} \dot{q}_1 c_{23} \\ -\dot{q}_1 s_{23} \\ \dot{q}_2 + \dot{q}_3 \end{pmatrix} $$

$$ {}^{4}v_4 = \begin{pmatrix} l_2\dot{q}_1 c_{234} - l_4\dot{q}_3 c_4 - l_4\dot{q}_2 c_4 - l_3\dot{q}_2 c_{23} \\ l_4\dot{q}_2 s_4 + l_4\dot{q}_3 s_4 - l_2\dot{q}_1 s_{234} + l_3\dot{q}_2 s_{34} \\ l_3\dot{q}_1 s_2 + l_4\dot{q}_1 s_{23} \end{pmatrix},\ {}^{4}\omega_4 = \begin{pmatrix} \dot{q}_1 s_{234} \\ \dot{q}_1 c_{234} \\ \dot{q}_2 + \dot{q}_3 + \dot{q}_4 \end{pmatrix}, $$

$$ {}^{5}v_5 = \begin{pmatrix} -c_5(l_5\dot{q}_2 + l_5\dot{q}_3 + l_5\dot{q}_4 + l_4\dot{q}_2 c_4 + l_4\dot{q}_3 c_4 - l_2\dot{q}_1 c_{234} + l_3\dot{q}_2 c_{34}) - \dot{q}_1 s_5(l_4 s_{23} + l_3 s_2 + s_{234}) \\ s_5(l_5\dot{q}_2 + l_5\dot{q}_3 + l_5\dot{q}_4 + l_4\dot{q}_2 c_4 + l_4\dot{q}_3 c_4 - l_2\dot{q}_1 c_{234} + l_3\dot{q}_2 c_{34}) - \dot{q}_1 c_5(l_4 s_{23} + l_3 s_2 + s_{234}) \\ l_4\dot{q}_2 s_4 + l_4\dot{q}_3 s_4 - l_2\dot{q}_1 s_{234} + l_3\dot{q}_2 s_{34} \end{pmatrix}, $$

$$ {}^{5}\omega_5 = \begin{pmatrix} \frac{\dot{q}_1 s_{2345}}{2} - \dot{q}_2 s_5 - \dot{q}_3 s_5 - \dot{q}_4 s_5 + \frac{\dot{q}_1 s_{234_5}}{2} \\ \frac{\dot{q}_1 c_{2345}}{2} - \dot{q}_2 c_5 - \dot{q}_3 c_5 - \dot{q}_4 c_5 + \frac{\dot{q}_1 c_{234_5}}{2} \\ \dot{q}_5 + \dot{q}_1 + c_{234} \end{pmatrix} $$

$$ {}^{6}v_6 = \begin{pmatrix} s_6\left(l_4\dot{q}_2 s_4 + l_4\dot{q}_3 s_4 - l_2\dot{q}_1 s_{234} + l_3\dot{q}_2 s_{34}\right) - c_6\left(c_5\left(l_5\dot{q}_2 + l_5\dot{q}_3 + l_5\dot{q}_4 + l_4\dot{q}_2 c_4 + l_4\dot{q}_3 c_4 - l_2\dot{q}_1 c_{234} - l_3\dot{q}_2 c_{34}\right) + \right. \\ c_6\left(l_4\dot{q}_2 s_4 + l_4\dot{q}_3 s_4 - l_2\dot{q}_1 s_{234} + l_3\dot{q}_2 s_{34}\right) + s_6\left(c_5\left(l_5\dot{q}_2 + l_5\dot{q}_3 + l_5\dot{q}_4 + l_4\dot{q}_2 c_4 + l_4\dot{q}_3 c_4 - l_2\dot{q}_1 c_{234} - l_3\dot{q}_2 c_{34}\right) + \\ \dot{q}_1 c_5\left(l_4 s_{23} + l_3 s_2 + l_{\ 5} s_{234}\right) - s_5\left(l_5\dot{q}_2 l_5\dot{q}_3 + l_5\dot{q}_4 + l_4\dot{q}_2 c_4 + l_4\dot{q}_3 c_4 - l_2\dot{q}_1 c_{234} + \right. \\ \left. \dot{q}_1 s_5\left(l_4 s_{23} + l_3 s_2 + l_5 s_{234}\right)\right) \\ \left. \dot{q}_1 s_5\left(l_4 s_{23} + l_3 s_2 + l_5 s_{234}\right)\right) \\ l_3\dot{q}_2 c_{34}\Big) \end{pmatrix} $$

$$ {}^{6}\omega_6 = \begin{pmatrix} \dot{q}_5 s_6 - c_6\dot{q}_2 s_5 - c_6\dot{q}_3 s_5 - c_6\dot{q}_4 s_5 + c_3\dot{q}_1 s_6 c_{24} - \dot{q}_1 s_3 s_6 s_{24} + c_3 c_5 c_4\dot{q}_1 s_{24} + c_5 c_6\dot{q}_1 s_3 c_{24} \\ \dot{q}_5 c_6 + s_6\dot{q}_2 s_5 + s_6\dot{q}_3 s_5 + s_6\dot{q}_4 s_5 + c_3\dot{q}_1 c_6 c_{24} - \dot{q}_1 c_6 s_3 s_{24} - c_3 c_5 c_4\dot{q}_1 s_{24} - c_5 s_6\dot{q}_1 s_3 c_{24} \\ \dot{q}_6 + c_5\dot{q}_2 + c_5\dot{q}_3 + c_5\dot{q}_4 + c_3\dot{q}_1 s_5 s_{24} + \dot{q}_1 s_3 s_5 c_{24} \end{pmatrix} $$

$$
{}^{7}v_7 = \begin{pmatrix}
c_6 l_6 \dot{q}_5 - c_5 c_6 l_5 \dot{q}_2 - c_5 c_6 l_5 \dot{q}_3 - c_5 c_6 l_5 \dot{q}_4 + l_6 \dot{q}_2 s_5 s_6 + l_6 \dot{q}_3 s_5 s_6 + l_6 \dot{q}_4 s_5 s_6 + l_4 \dot{q}_2 s_6 s_4 + l_4 \dot{q}_3 s_6 s_4 + c_3 c_6 l_6 \dot{q}_1 c_{23} - c_3 l_2 \dot{q}_1 s_6 s_{24} - \\
c_5 l_5 \dot{q}_2 s_6 - l_6 \dot{q}_5 s_6 + c_5 l_5 \dot{q}_3 s_6 + c_6 l_6 \dot{q}_2 s_5 + c_5 l_5 \dot{q}_4 s_6 + c_6 l_6 \dot{q}_3 s_5 + c_6 l_6 \dot{q}_4 s_5 + c_6 l_4 \dot{q}_2 s_4 + c_4 l_4 \dot{q}_3 s_4 + l_3 \dot{q}_1 s_5 s_6 s_2 - c_3 c_6 l_2 \dot{q}_1 s_{24} - \\
c_5 l_3 \dot{q}_1 s_2 - l_5 \dot{q}_3 s_5 - l_5 \dot{q}_4 s_5 - l_5 \dot{q}_2 s_5 - l_4 \dot{q}_2 s_5 c_4 - l_4 \dot{q}_3 s_5 c_4 + l_3 \dot{q}_2 s_3 s_5 s_4 + c_3 c_5 l_5 \dot{q}_2 s_{24} + c_3 l_2 \dot{q}_1 s_5 c_{24} + c_5 l_5 \dot{q}_1 s_3 c_{24} - \\
c_6 l_6 \dot{q}_1 s_3 s_{24} - l_2 \dot{q}_1 s_3 s_6 c_{24} - c_5 c_6 l_4 \dot{q}_2 c_4 - c_5 c_6 l_4 \dot{q}_3 c_4 - c_6 l_3 \dot{q}_1 s_5 s_2 + c_3 l_3 \dot{q}_2 s_6 s_4 + l_3 \dot{q}_2 s_6 c_4 + c_3 c_5 l_2 \dot{q}_1 c_{24} - c_5 c_6 l_2 \dot{q}_1 s_3 s_{24} - c_3 c_6 l_5 \dot{q}_1 s_5 s_{24} - \\
c_6 l_2 \dot{q}_1 s_3 c_{24} - c_3 l_6 \dot{q}_1 s_6 c_{24} + l_6 \dot{q}_1 s_3 s_6 s_{24} + c_3 c_6 l_3 \dot{q}_2 s_4 + c_6 l_2 \dot{q}_2 s_3 c_4 + c_5 l_4 \dot{q}_2 s_6 c_4 + c_5 l_4 \dot{q}_3 s_6 c_4 - c_3 c_5 c_6 l_6 \dot{q}_1 s_{24} - c_3 c_5 l_2 \dot{q}_1 s_6 c_{24} - c_5 c_6 l_6 \dot{q}_1 s_3 c_{24} + \\
l_2 \dot{q}_1 s_3 s_5 s_{24} + c_3 c_5 l_4 \dot{q}_1 s_2 + c_5 l_4 \dot{q}_1 s_3 c_2 - c_3 l_3 \dot{q}_2 s_5 c_4 \\
c_3 c_5 l_6 \dot{q}_1 s_6 s_{24} - c_6 l_5 \dot{q}_1 s_3 s_5 c_{24} - c_5 l_6 \dot{q}_1 s_3 s_6 c_{24} - c_3 c_5 c_6 l_3 \dot{q}_2 c_4 - c_3 c_6 l_4 \dot{q}_1 s_5 s_2 + c_5 c_6 l_3 \dot{q}_2 s_3 s_4 - c_6 l_4 \dot{q}_1 s_3 s_5 c_2 \\
c_5 l_2 \dot{q}_1 s_3 s_6 s_{24} + c_3 l_5 \dot{q}_1 s_5 s_{24} + l_5 \dot{q}_1 s_3 s_5 s_6 c_{24} + c_3 c_5 l_3 \dot{q}_2 s_6 c_4 + c_3 l_4 \dot{q}_1 s_5 s_6 s_2 - c_5 l_3 \dot{q}_2 s_3 s_6 s_4 + l_4 \dot{q}_1 s_3 s_5 s_6 c_2
\end{pmatrix}
$$

$$
{}^{7}\omega_7 = \begin{pmatrix}
\dot{q}_5 s_6 - c_6 \dot{q}_2 s_5 - c_6 \dot{q}_3 s_5 - c_6 \dot{q}_4 s_5 + c_3 \dot{q}_1 s_6 c_{24} - \dot{q}_1 s_3 s_6 s_{24} + c_3 c_5 c_6 \dot{q}_1 s_{24} + c_5 c_6 \dot{q}_1 s_3 c_{24} \\
\dot{q}_5 c_6 + s_6 \dot{q}_2 s_5 + s_6 \dot{q}_3 s_5 + s_6 \dot{q}_4 s_5 + c_3 \dot{q}_1 c_6 c_{24} - \dot{q}_1 c_6 s_3 s_{24} - c_3 c_5 s_6 \dot{q}_1 s_{24} - c_5 s_6 \dot{q}_1 s_3 c_{24} \\
\dot{q}_6 + c_5 \dot{q}_2 + c_5 \dot{q}_3 + c_5 \dot{q}_4 + c_3 \dot{q}_1 s_5 s_{24} + \dot{q}_1 s_3 s_5 c_{24}
\end{pmatrix}
$$

Anexo 2. Tabla de parámetros del robot UR5.

Parametro	valor
l_1	$0.089m$
l_2	$0.10m$
l_3	$0.42m$
l_4	$0.39m$
l_5	$0.095m$
l_6	$0.82m$
m_1	$3.7kg$
m_2	$8.393kg$
m_3	$2.275kg$
m_4	$1.214kg$
m_5	$1.219kg$
m_6	$0.187kg$
I_1	$diag(0.0014,\ 0.0015,\ 0.0014)kgm^2$
I_2	$diag(0.0035,\ 0.0035,\ 0.0035)kgm^2$
I_3	$diag(0.0088,\ 9.4792e-04,\ 0.0088)kgm^2$
I_4	$diag(0.0041,\ 5.0583e-04,\ 0.0041)kgm^2$
I_1	$diag(4.8316e-04,\ 5.0792e-04,\ 4.8316e-04)kgm^2$
I_1	$diag(6.5467e-05,\ 7.8292e-05,\ 6.5467e-05)kgm^2$

Referencias

[1] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full*, et al.*, "The grand challenges of Science Robotics," *Science Robotics,* vol. 3, 2018.

[2] C. T. Landi, V. Villani, F. Ferraguti, L. Sabattini, C. Secchi, and C. Fantuzzi, "Relieving operators' workload: Towards affective robotics in industrial scenarios," *Mechatronics,* vol. 54, pp. 144-154, 2018/10/01/ 2018.

[3] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, "Robotics Modelling, Planning and Control," *Springer-Verlag,* 2009.

[4] A. de Luca, L. Lanari, and G. Oriolo, "A sensitivity approach to optimal spline robot trajectories," *Automatica,* vol. 27, pp. 535-539, 1991/05/01/ 1991.
[5] W. Aribowo and K. Terashima, "An implementation of cubic spline optimization with floating via points and vibration suppression for trajectory planning of industrial robot arm," in *SICE Annual Conference (SICE), 2013 Proceedings of*, 2013, pp. 288-293.
[6] X. Niu and T. Wang, "C2-continuous orientation trajectory planning for robot based on spline quaternion curve," *Assembly Automation,* vol. 38, pp. 282-290, 2018.
[7] Y. Chen and L. Li, "Collision-free trajectory planning for dual-robot systems using B-splines," *International Journal of Advanced Robotic Systems,* vol. 14, p. 1729881417728021, 2017.
[8] H. Ozaki, K. Hirano, M. Iwamura, C.-J. Lin, and T. Shimogawa, "Improvement of trajectory tracking for industrial robot arms by learning control with B-spline," in *Assembly and Task Planning, 2003. Proceedings of the IEEE International Symposium on*, 2003, pp. 264-269.
[9] C. D. Porawagama and S. R. Munasinghe, "Reduced jerk joint space trajectory planning method using 5-3-5 spline for robot manipulators," in *7th International Conference on Information and Automation for Sustainability*, 2014, pp. 1-6.
[10] www.universal-robots.com.
[11] J. J. Craig, *Introduction to Robotics: Mechanics & Control*: Addison-Wesley Publ., 1986.
[12] R. C. G. K. S. Fu, C. S. G. Lee, "ROBOTICS: Control, Sensing, Vision, and Intelligence," *McGraw-Hill,* 1987.
[13] A. Barrientos and B. Carlos, *Fundamentos de robótica, 2a Ed*: McGraw-Hill Interamericana de España S.L., 2007.
[14] P. Kebria, S. Al-Wais, H. Abdi, and S. Nahavandi, *Kinematic and dynamic modelling of UR5 manipulator*, 2016.
[15] R. S. Andersen, "Kinematics of a UR5," *Available at: http://rasmusan.blog.aau.dk/files/ur5_kinematics.pdf,* vol. Aalborg University, 2018.
[16] D. Guida and C. M. Pappalardo, "Forward and inverse dynamics of nonholonomic mechanical systems," *Meccanica,* vol. 49, pp. 1547-1559, July 01 2014.
[17] R. Kelly, V. S. Davila, V. Santibañez, and A. Loria, *Control of Robot Manipulators in Joint Space*: Springer-Verlag London Limited, 2005.
[18] W. Carlos Canudas de, B. Georges, and S. Bruno, *Theory of Robot Control*: Springer-Verlag New York, Inc., 1996.

Ingeniería y Automatización, Capítulo 25, pp. 322 - 335.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Aplicación de Sistemas Embebidos e IoT para el Monitoreo de Estanques Acuícolas en Eldorado, Sinaloa

Nava Pintor Jesús Antonio[1], Guerrero Osuna Héctor Alonso[1], Ornelas Vargas Gerardo[1], Solís Sánchez Luis Octavio[1] y Castañeda Miranda Rodrigo[1]

Posgrado en Ingeniería y Tecnología Aplicada
[1] Universidad Autónoma de Zacatecas
Jardín Juárez #147, Centro Histórico C.P. 98000 Zacatecas, Zacatecas, México
antonio.nava9022@gmail.com

Resumen

En la actualidad el estado de Sinaloa se encuentra entre los primeros lugares a nivel nacional en producción de camarón, así como de otras actividades acuícolas, gracias a esto, muchos productores han podido encontrar una actividad sustentable y sostenible. Es por esta razón que los acuicultores buscan alternativas que les ayude a mejorar sus procesos de producción. En Sinaloa como en muchos lugares alrededor del mundo el proceso de producción de camarón se realiza mediante personal capacitado que toma muestras químicas del agua o con dispositivos electrónicos capaces de medir la calidad del agua, sin embargo, las ineficiencias del personal responsable de tomar las mediciones de las variables importantes para el buen desarrollo del camarón producen muchas pérdidas en la producción, que se traduce a perdida de dinero. Por ese motivo se describe el diseño, desarrollo e implementación de un sistema de monitoreo remoto con conectividad al internet de las cosas que permita al usuario disponer de mediciones confiables de calidad del agua en estanques dedicados a la producción de camarón mediante una interfaz gráfica en la que el productor acuícola podrá visualizar con más exactitud los datos obtenidos, reduciendo los fallos humanos, costos de contratación de personal y aumentando la cantidad de camarón producido.

Palabras clave: Acuicultura, LoRa, calidad del agua, siembra de camarón, red de sensores inalámbricos, red de área amplia, sistemas embebidos, monitoreo, Internet de las cosas (IoT), Industria 4.0.

1. Introducción

Actualmente, los sectores productivos primarios constituyen una parte esencial en el ámbito estratégico para el desarrollo económico del país [1], hoy en día es usual que maquinaria especializada para estos sectores sean utilizados para obtener mejoras en la producción, sin embargo, con el surgimiento de una nueva revolución industrial, la llamada “Industria 4.0”, se vuelve necesario aplicar innovación en sectores primarios [2]. La industria 4.0 es un concepto que busca la interconexión de los sistemas físicos (sensores, actuadores, etc.) con tecnologías de la información, con el fin de obtener una mejora en los servicios, posibilita también la virtualización de la información, auto diagnóstico, auto ajuste y la optimización de los procesos de producción, siendo clave para la mejora de las condiciones de trabajo laborales y en la obtención de mejores productos [3].

En los sectores productivos primarios se está buscando introducir el concepto de industria 4.0, sin embargo, debido a los diferentes problemas que este sector implica, como la lejanía de la zona de producción, las limitaciones energéticas, las fallas en las comunicaciones, el entorno ambiental, costos de producción, entre otros factores, han dificultado su implementación [4]. En el área de la acuicultura, específicamente en la siembra de camarón, el monitoreo de los parámetros de calidad del agua es de vital importancia para asegurar el buen desarrollo del producto, un deficiente monitoreo puede llevar al deterioro en las condiciones de calidad del agua, que afectan la salud del camarón de tal forma que se puede llegar a comprometer considerablemente la cosecha. Los parámetros más críticos a monitorear cuando se trata de cultivos acuícolas, son el oxígeno disuelto y la temperatura, mientas más alta sea la temperatura en el agua la solubilidad de oxígeno disuelto tiende a disminuir [5] y en condiciones en donde el oxígeno disuelto es nulo, el camarón puede morir, sin embargo, si los niveles de oxígeno

disuelto son bajos, los camarones tienden a tener un crecimiento lento y los hace propensos a adquirir enfermedades fácilmente [6]. Además, la cantidad de alimento que consume el camarón depende mucho del oxígeno disuelto, cuando los niveles de oxígeno disuelto no son los óptimos el camarón no se alimenta, es por eso que muchos acuicultores no lo proporcionan cuando los niveles de oxígeno son bajos. Contrariamente, en estanques donde se administra oxigeno continuamente, las especies marinas pueden alimentarse a cualquier hora del día, esto puede disminuir considerablemente perdidas económicas por sobrealimentación y evitar la contaminación de los estanques [7], sin embargo, los costos de inversión en mano de obra y productos químicos son mayores.

Mantener unos niveles óptimos en los parámetros de calidad del agua resulta ser un problema internacional, en Bangka Island, indonesia, uno de los principales problemas es la localización geográfica de los estanques, que, al estar en áreas remotas, resulta difícil llevar un monitoreo preciso [8], además el uso de recurso humano no es suficiente para realizar un buen monitoreo, los procedimientos químicos, se vuelve un proceso muy largo y caro, que va desde la contratación de recurso humano que tome las muestras, hasta las personas que realizan las pruebas químicas, lo que puede perjudicar en tiempo de respuesta de ser necesario[9]. Otra manera de tomar medidas es mediante dispositivos de mano, sin embargo, esta opción se presta a realizar mediciones fuera de tiempo o a la mala práctica por parte de los trabajadores [10].

En Eldorado, Sinaloa las mediciones de calidad del agua se llevan a cabo manualmente mediante la utilización de herramientas especializadas de medición, específicamente de la marca YSI el cual tiene un costo de aproximadamente 30,000 pesos mexicanos, las mediciones se realizan en la parte media del estanque utilizando lanchas de transporte, el equipo de medición tarda de 10 a 15 minutos para estabilizarse y poder realizar medidas precisas, por lo que el personal tiene que esperar este tiempo antes de tomar la medición, el equipo registra la medida y posteriormente el trabajador tiene que salir nuevamente del estanque y repetir el mismo procedimiento en cada uno de los estanques siguientes. Actualmente la empresa cuenta con cerca de 10 estanques dedicados a la cosecha de camarón repartidos en alrededor de 10 kilómetros cerca de la costa de la playa, el lugar de cosecha se encuentra aproximadamente a 20 Km alejadas de las oficinas de operación, por lo que el personal se tiene que desplazar por aproximadamente 30 minutos de viaje para tomar las mediciones, véase la figura 1, estas mediciones deben realizarse al menos 3 veces al día debido a los cambios que sufren los estanques de oxígeno disuelto, la medición más critica a realizar es durante la noche, cuando la fotosíntesis cesa las plantas en el estanque como el fitoplancton dejan de producir oxígeno [11], los camarones y otros organismos vivos siguen consumiéndolo, por lo que, si no se realiza una medida precisa y se toman precauciones, el producto puede verse afectado a la mañana siguiente.

Figura 1. Trayectoria desde las oficinas centrales hasta la zona de cosecha.

Este proceso de monitoreo genera unas condiciones de trabajo adversas para el personal responsable de ejecutarlo, que para minimizar su carga de trabajo tiende a realizar mal las mediciones, omitir o incluso falsificar medidas. Esto ocasiona que el productor no tenga la información adecuada y por ende no se ejecuten las acciones pertinentes de ser necesarias cuando exista una anomalía en los estanques. Además, la inversión que el acuicultor tiene que hacer para la compra de los equipos de medición, la contratación de trabajadores, vehículos de transporte, combustibles, etc. es considerable y no se pueden permitir perdidas en la producción debido a la omisión o fallo de los trabajadores en la zona de cosecha.

Analizando la información aquí presentada, encontramos que para un monitoreo óptimo de los estanques de cosecha de camarón, es necesario que el sistema cuente con los siguientes requerimientos:

- Monitoreo Continuo: La información debe estar disponible en intervalos de tiempo cortos, para que el acuicultor tenga conocimiento del estado actual de sus estanques. La información debe ser mostrada de forma amigable para que el acuicultor la entienda fácilmente.

- Cobertura de la zona de cosecha de Camarón: El sistema de monitoreo inalámbrico debe cubrir los estanques pertenecientes a los acuicultores, en total son 10 estanques dedicados a la cosecha de camarón.

- Suministro de Energía: Al ser una zona alejada del pueblo, la zona de cosecha no cuenta con instalaciones eléctricas, por lo que el uso de aparatos electrónicos que dependan de una fuente de suministro de energía constante se vuelve una tarea difícil, sin embargo, el sistema debe contar con energía eléctrica para poder desempeñar su trabajo.

- Conexión a Internet: El sistema está basado en el concepto de Internet de las Cosas (IoT), para subir la información de los nodos de monitoreo es necesario tener acceso a internet.

- Análisis de Información: La plataforma debe permitir el análisis y gestión de los datos recabados para implementar mejoras como control automático o análisis predictivo.

Para cubrir estas necesidades, se propone implementar un sistema de monitoreo autónomo inalámbrico que le permita al acuicultor visualizar continuamente las condiciones actuales de calidad del agua de sus estanques, el sistema utiliza la tecnología inalámbrica de bajo consumo energético y largo alcance LoRa que cubre sin problemas el área de cosecha. El equipo se conecta a internet gracias al uso de la red celular 3G, la visualización de los datos se hace mediante una plataforma en línea dedicada al Internet de las Cosas (www.resiot.oi), en donde se almacena y puede ser revisada, analizada y gestionada (Big Data). El sistema usa los recursos naturales existentes en las granjas para que el suministro de energía sea renovable, además, con esto se busca introducir el concepto de industria 4.0 en acuicultura. El sistema propuesto establece una base para mejorar los procesos de producción, fomentar el crecimiento y buen desarrollo del producto, y deja abierta la posibilidad a futuras investigaciones y aplicaciones en el área.

El documento está organizado de la siguiente manera, la segunda sección “Trabajos Relacionados” habla de los trabajos relacionados y que influyeron en el desarrollo de este proyecto. La tercera sección “Materiales y Métodos” describe el desarrollo del sistema, su instalación y las pruebas de funcionamiento en un estanque acuícola en el estado de Sinaloa. La tercera sección “Resultados”, muestra los resultados obtenidos del desarrollo e implementación del sistema. Por último, la cuarta sección “Conclusiones” se reflexiona sobre la implementación del sistema, así como de la posibilidad a trabajos futuros.

2. Trabajos Relacionados

En esta sección se presentan algunos de los trabajos que se han implementado para su uso en los sectores primarios y los cuales fueron influencia para el desarrollo de nuestro trabajo.

N. Murali [12] desarrolló una red de sensores y actuadores para controlar las redes de distribución de agua, la red consiste de nodos con válvulas eléctricas con comunicación de radiofrecuencia LoRa, monitorean el nivel del agua y la información es recopilada por una raspberry Pi que actúa como puerta de enlace para subir la información, y como control de las válvulas, se diseñó un software para la visualización de la información y el manejo manual de las válvulas.

Liu Y. [13] implementaron un sistema para combatir la contaminación en el agua mediante sensores de temperatura, pH, turbidez y conductividad los cuales se comunican vía LoRa para la transmisión de datos hacia un Gateway que sube la información a una plataforma IoT en donde puede ser visualizada la información en tiempo real.

Maulana Y.[14] describe el diseño de un sistema de monitoreo de calidad del agua en cultivos de camarón, el sistema está diseñado para monitorear específicamente el oxígeno disuelto, pH y la temperatura en los estanques, se diseñó un módulo de sensor el cual recolecta la información y la envía inalámbricamente mediante el protocolo Zegbee a un módulo “datalogger” el cual cuenta con capacidad de GPRS/GSM por el cual manda la información a una página web también diseñada y que está conectada a una base de datos por MySQL.

Cao-Hoang T.[15] Propone el desarrollo de un sistema rentable para el monitoreo de los datos del suelo, el cual está conformado por un dispositivo que monitorea los datos del suelo en un intervalo de una hora y transmite la información a la nube mediante el protocolo GPRS, la información puede ser visualizada en una página web llamada heroku.com, el sistema está pensado para sus uso en la agricultura.

Payero J.[16] propone un sistema de monitoreo de humedad del suela con conectividad al internet de las cosas de bajo coste, para promover la aplicación de la tecnología en la agricultura, se desarrolló un dispositivo con sensor de humedad, el cual fue instalado en una granja de trigo, los datos recopilados son enviados vía WiFi a una plataforma del internet de las cosas ThingSpeak.com para alojar y visualizar los datos, el sistema recopilo datos exitosamente y se pretende promover el uso del sistema para la programación de riego entre los productores comerciales.

3. Materiales y Métodos

3.1 Desarrollo del Sistema.

El sistema propuesto está conformando por un nodo Gateway, nodo de monitoreo y la plataforma en línea para el Internet de las Cosas (IoT). Los nodos de monitoreo son instalados en los estanques de cosecha de camarón, en donde recolectan datos de temperatura y oxígeno disuelto, los datos recabados son enviados por radiofrecuencia al nodo Gateway que sube dicha información a la red mediante el protocolo 3G, los datos pueden ser observados en la plataforma www.resiot.io para su monitoreo continuo, la figura 2 proporciona el marco general del sistema propuesto.

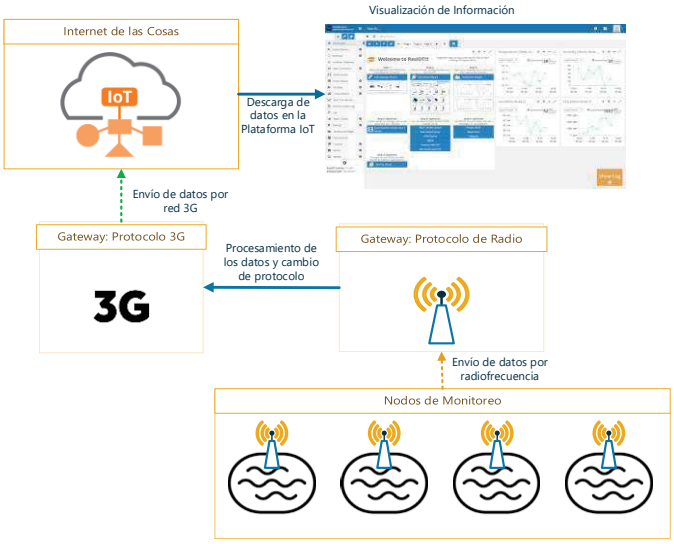


Figura 2. Marco General del Sistema Propuesto.

3.2 Nodo de Monitoreo

El objetivo del nodo de monitoreo es tomar las medidas de calidad del agua, específicamente la temperatura y el oxígeno disuelto en los estanques acuícolas. Para cumplir con su objetivo el nodo contiene un atmega328p, el cual es un microcontrolador de 8 bits que soporta comunicaciones UART, I2C y SPI que son las necesarias para los periféricos a utilizar. Los periféricos esta conformados por los sensores de temperatura y oxígeno disuelto que pueden ser conectados vía UART o I2C, sin embargo, se optó por utilizar el protocolo I2C con el fin de utilizar la misma línea de transmisión y recepción. También se cuenta con un reloj de tiempo real (RTC), el cual, como característica principal, contiene un pin de interrupción que puede ser activado por una alarma de tiempo, el RTC también utiliza el protocolo I2C, y por último se tiene el módulo de comunicación inalámbrica LoRa, el cual es un dispositivo de bajo consumo energético y largo alcance que utiliza el protocolo SPI para el control de registros. En las figuras 3, 4 y 5 se puede apreciar la interconexión de los periféricos con el microcontrolador, el diseño y la placa de circuito impreso resultante.

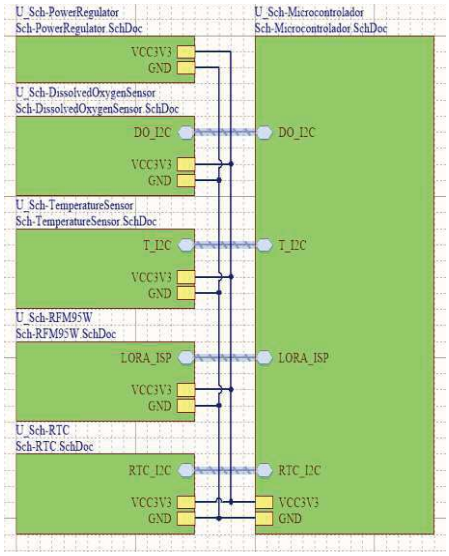


Figura 3. Interconexión de los periféricos del nodo de monitoreo.

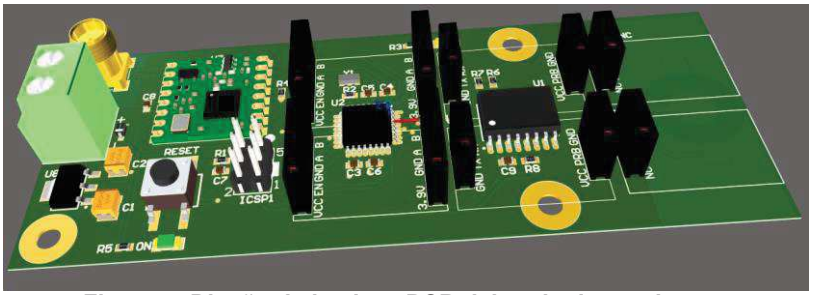


Figura 4. Diseño de la placa PCB del nodo de monitoreo.

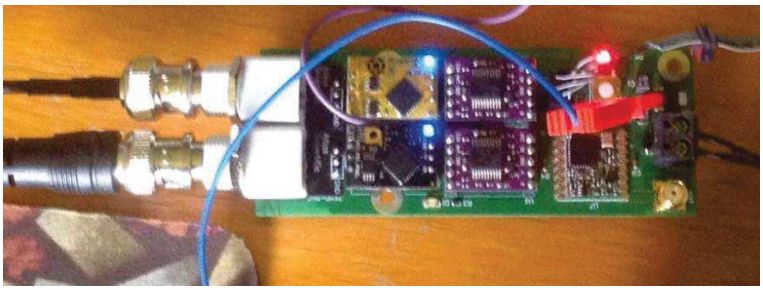


Figura 5. Resultado de la placa PCB para el nodo de monitoreo.

El nodo de monitoreo utiliza al RTC contenido para activar una alarma cada 5 minutos mediante el pin de interrupción, el cual le indica al microcontrolador cuando debe despertar y realizar la lectura de los sensores, estas lecturas se realizan 5 veces y se saca un promedio para obtener mejores datos de información, posteriormente el microcontrolador envía los datos recopilados mediante el módulo de radiofrecuencia LoRA hacia el nodo Gateway, y finalmente el microcontrolador configura una nueva alarma y el nodo de monitoreo pasa a un estado de “dormido” para obtener un ahorro en el consumo energético, en la figura 6 puede apreciarse el diagrama de flujo del funcionamiento del nodo de monitoreo.

Figura 6. Diagrama de flujo del nodo de monitoreo

El suministro de energía eléctrica para el nodo de monitoreo se realiza mediante un panel solar y una batería en donde es almacenada la energía, el controlador administra la recarga de la batería y la salida del voltaje hacia el circuito electrónico. Cuando el nodo está en estado “despierto”, obtiene un consumo de 168.4 mA y cuando entra en modo “dormido” baja a 50mA. Para cubrir las demandas energéticas del nodo de monitoreo se determinó el uso de un panel solar de 25 watts y una batería de 7Ah que asegura su funcionamiento durante 24 horas continuas. La figura 7 muestra la instalación del nodo, la batería y el controlador de carga en el gabinete contenedor, el panel se sueldo a la parte superior del gabinete.

Figura 7. Instalación del equipo electrónico en el gabinete (izquierda) y el panel solar (derecha).

3.3 Nodo Gateway

El nodo Gateway como su nombre lo indica, es una puerta de enlace entre los nodos de monitoreo y el interne de las cosas, tiene la tarea de recopilar la información de los nodos de monitoreo utilizando comunicación inalámbrica LoRa y subirlas a la red. El Gateway soporta dos protocolos de conexión a internet, ethernet o mediante la red celular 3G, sin embargo, debido a las limitaciones eléctricas de la zona, no se cuenta con una red de internet cercana, por lo que lo más factible es utilizar es la red 3G, la figura 8 muestra el Gateway a utilizar. el Gateway es configurado para que la información que le llegue sea direccionada a la plataforma de internet de las cosas Resiot.io en donde serán visualizados los datos.

Figura 8. Gateway de la marca multitech.

El Gateway tiene un rango de voltaje de operación de 9VCD hasta 32VCD, y tiene un consumo energético de 600mA, el Gateway debe mantenerse prendido todo el día para estar al tanto de la llegada de información. Para cubrir los requerimientos energéticos del Gateway se agregaron un panel solar de 80 watts, una batería recargable de 12V 12 Ah, y un controlador de carga, con este equipo, el Gateway puede estar encendido por casi 24 horas solo con el uso de la batería.

Figura 9. Instalación del equipo electrónico y Gateway (Derecha) y el panel solar (izquierda).

3.4 Plataforma de IoT

ResIoT es una plataforma especializada para proyectos de IoT e industria 4.0. Permite la gestión del Gateway, canales de transmisión, conexiones MQTT, reglas para el control de reinicio, y su localización. ResIoT permite también la gestión de millones de dispositivos en una red y proporciona la posibilidad de usar "dashboards" que son de utilidad para la visualización de la información, cuenta también con gráficos lineales, botones, imágenes y mapas. Permite utilizar el protocolo MQTT con el cual se puede comunicar maquina a máquina cuando sea necesario y cuenta con un lenguaje script basado en LUA 5.1 con el cual se puede desarrollar automatización, gestión automática de datos, análisis predictivo, etc. Tienen un apartado dedicado a las alertas y notificaciones, que posibilita enviar correos electrónicos o mostrarlo directamente en la interfaz de ResIoT. Además, cuenta con una aplicación móvil con la cual se puede realizar las mismas acciones que estando en una computadora.

Para permitir la recepción de información, primero se tiene que añadir las llaves de sesión del nodo de monitoreo. Se utilizan dos llaves que son únicas para cada dispositivo LoRa, la NwKSkey es usada para garantizar la integridad del mensaje desde el dispositivo al servidor, y la AppSkey es utilizada para el cifrado de la encriptación AES-128. Para el dispositivo de monitoreo se utilizó un método de unión ABP (Activation By Personalization), en el cual las dos llaves son colocadas manualmente por el usuario, de esta forma se busca tener un control en los dispositivos que son agregados a la lista de recepción del Gateway.

Para la visualización de la información se desarrolló un script LUA, este se ejecuta cuando la información llega al servidor de ResIoT, el script toma la información y la direcciona al dashboard especificado para su visualización, ResIoT se actualiza cada minuto por lo que el grafico mostrara la información cuando transcurra este tiempo. el acuicultor solo tiene que entrar a la URL administrada para poder visualizar la información. La figura 10 muestra la sección grafica de la plataforma ResIot.io.

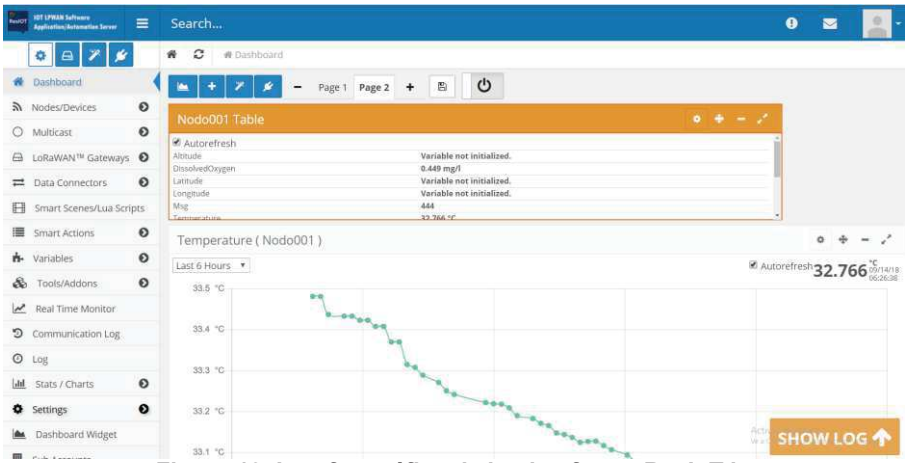


Figura 10. Interfaz gráfica de la plataforma ResloT.io

Además, existe un apartado en la plataforma en donde se guarda el historial de los datos de los últimos 5 días de cualquier nodo, 5 días es el máximo permitido por ResloT debido a que es una versión gratuita, así como solo permite agregar 15 dispositivos. Sin embargo, esto por a hora no supone un problema ya que el acuicultor cuenta con 10 estanques únicamente, por lo que podemos cubrir el resto sin tener que cambiar el tipo de cuenta, además la información se puede descargar y guardar localmente.

3.5 Instalación del Equipo

Las pruebas se realizaron en Eldorado Sinaloa, en la zona de cosecha marcada con circulo amarillo en la figura 11, se eligió uno de los estanques para realizar las pruebas de funcionamiento debido a que cuenta con una lancha ya instalada en el agua y a que se encuentra cerca de una pequeña choza perteneciente a los acuicultores.

Figura 11. Zona de Cosecha cerca de las playas de ponce en Eldorado, Sinaloa. El circulo amarillo delimita los estanques pertenecientes a los acuicultores en donde se implementa el sistema.

Dicha choza sirvió como estructura para la instalación del nodo Gateway. Para la instalación del nodo de monitoreo se utilizó la lancha para colocar el nodo en un poste en la mitad del estanque. La figura 12 muestra la instalación del equipo del sistema de monitoreo.

Figura 12. Instalación de los dispositivos del sistema en tierra (izquierda) y en el estanque acuícola (derecha).

3.6 Pruebas de Funcionamiento

Prueba de Monitoreo: La prueba consiste en mantener un constante monitoreo de oxígeno disuelto y temperatura en un estanque en donde se realiza la cosecha de camarón, al mismo tiempo se toman medidas a diferentes horas con un equipo de mano registrando los datos, para posteriormente compararlos con los de nuestro sistema, véase la figura 13.

Figura 13. Registro de muestras realizadas manualmente para su comparación.

Pruebas de Alcance: Esta prueba se realizó desde la base de operaciones de la empresa, ubicada en el pueblo de Eldorado, Culiacán, Sinaloa a una distancia cercana a los 10 Km, el objetivo es enviar un dato desde las oficinas hasta la zona de cosecha y posteriormente visualizar dicho dato en la plataforma en línea ResIoT. En la figura 14 el circulo representa el área de cobertura de LoRa de aproximadamente 15 Km, podemos observar que la oficina se encuentra dentro del área, por lo que la información debería llegar al Gateway sin problema.

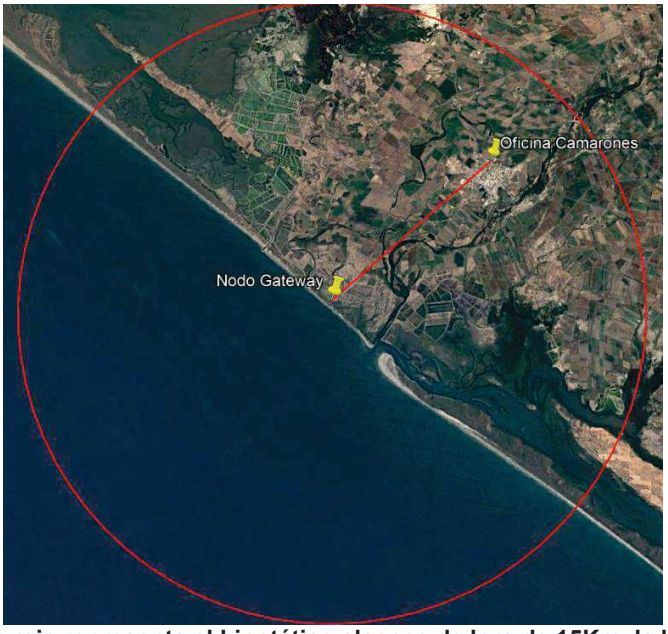


Figura 14. El circulo en rojo representa el hipotético alcance de lora de 15Km, la línea roja representa la distancia de 10Km que existe entre las oficinas de camarones y los estanques acuícolas en donde se encuentra el Gateway.

Pruebas de Suministro de Energía y Conexión de Internet: El suministro de energía está planeado para que los equipos no se apaguen, esto puede visualizarse mediante la plataforma de IoT ResIoT, el Gateway cada minuto envía un mensaje “Gateway alive” via red celular a la plataforma, esto significa que el dispositivo esta encendido, si el nodo Gateway recibe información continuamente significa que el nodo de monitoreo también esta encendido, esta prueba consiste en estar al tanto de que el sistema este continuamente conectado, esto se podrá visualizar en los “dasborads” de ResIoT. Con esto también podremos visualizar la estabilidad de la conexión de internet por medio de la red celular en la zona.

4. Resultados

El sistema de monitoreo autónomo inalámbrico que permite al acuicultor visualizar continuamente las condiciones actuales de calidad del agua de sus estanques, mediante la plataforma de IoT Resiot.io ha sido implementado, los datos medidos fueron comparados con los de un dispositivo de mano, la diferencia entre los datos medidos en el sensor de temperatura es de 0.5 °C y el de oxígeno disuelto es de 0.5 mg/l por arriba del nuestro, sin embargo según los acuicultores esto no supone un problema grave, la información también puede ser descargada en un archivo .csv para su análisis offline, en la figuras 15 y 16 se puede visualizar la información del sensor de temperatura y oxígeno disuelto respectivamente.

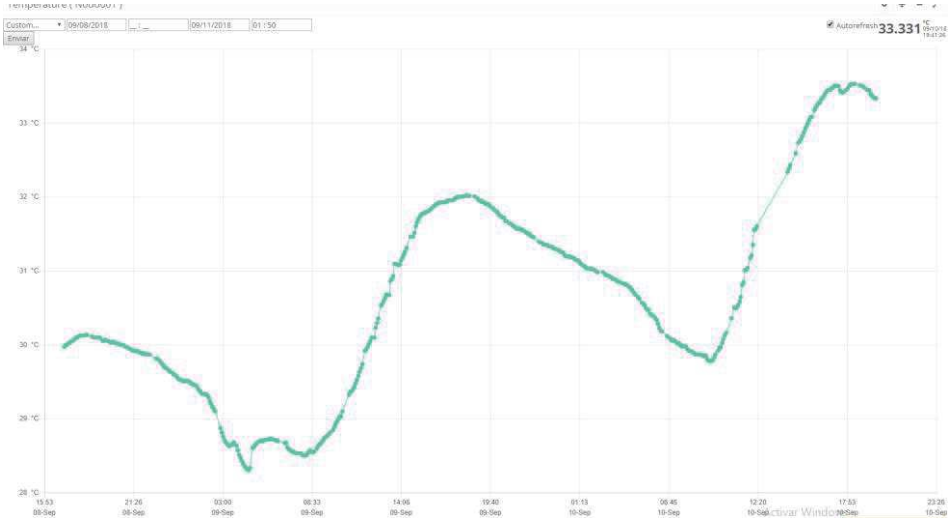


Figura 15. Dashboard correspondiente al sensor de temperatura.

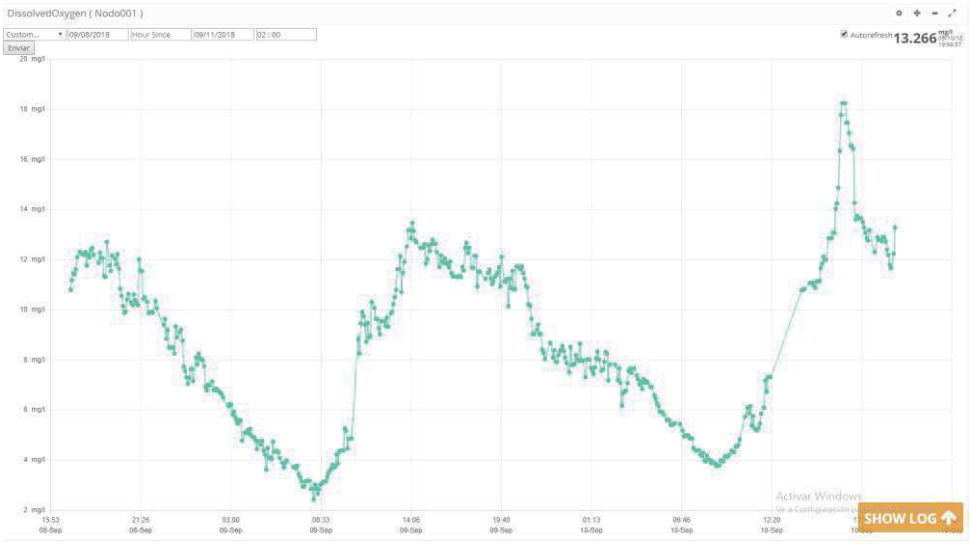


Figura 16. Dashboard correspondiente al sensor de oxígeno disuelto.

El resultado de la prueba de alcance fue positivo, se enviaron datos desde las oficinas centrales hasta la zona de cosecha, la informacion se visualizó satisfactoriamente en la plataforma ResIoT.io, por lo que la capacidad de cobertura el LoRa puede alcanzar cubre sin problemas la zona de la cosecha. La energia del gateway no se ha interrumpido y manda constantemente los mensajes de “gateway alive”, ademas recibe la informacion y la muestra en la plataforma, esto indica que el gateway y el nodo de monitoreo no se han apagado en ningun momento, vease la figura 17.

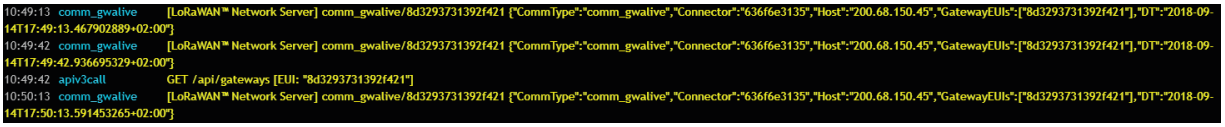


Figura 17. Mensajes recibidos por el Gateway, entre ellos el de “Gwalive”, mensaje recibido cuando el Gateway esta encendido.

Sin embargo, la conexión de la red celular ha presentado problemas, aunque pocas veces, existen desconexiones de la red, la desconexión más larga ha sido de 2 horas (véase la figura 18), sin embargo, estas desconexiones no pasan regularmente. Cuando la señal celular se restablece, el Gateway se vuelve a conectar automáticamente para realizar sus funciones.

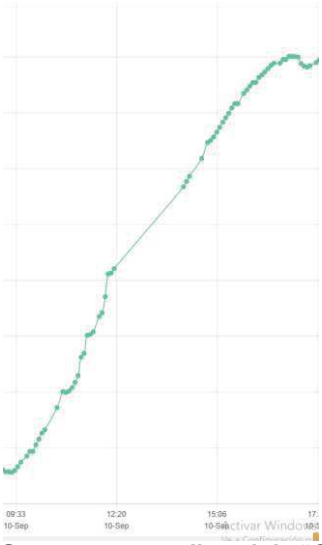


Figura 18. Mensajes recibidos por el Gateway, entre ellos el de "Gwalive", mensaje recibido cuando el Gateway esta encendido.

5. Conclusiones

En este papel, se describe la implementación de un sistema de monitoreo autónomo inalámbrico, el nodo de monitoreo realiza continuamente el monitoreo de temperatura y oxígeno disuelto, en uno de sus estanques de camarón y envía los datos al nodo Gateway. El Gateway sube la información a la plataforma ResIoT.io. Los acuicultores son capaces de observar la información desde sus dispositivos móviles, así como por sus computadoras, de esta forma pueden ver el estado actual de su estanque, el sistema utiliza el suministro de energía renovable mediante el uso de paneles solares y baterías recargables. La implementación del sistema nos permito visualizar el gran potencial de desarrollo posible en el área de la acuicultura, se busca desarrollar mejoras para posteriormente aplicarlas, algunas de las mejoras que se están planeando, son la aplicación de control de los alimentadores, disminuir el consumo energético de los equipos, establecer una red de internet más confiable, se busca el desarrollo de hardware con posibilidad de calibración de los sensores y reprogramación del dispositivo utilizando la plataforma ResIoT.io, así como mejorar la precisión en los sensores. El sistema aún está siendo implementado y siendo de utilidad para los acuicultores.

Referencias

[1] Hinojosa B. "Mexico Alimentaria 2018, Food Show", SAGARPA, 2018.

[2] Barreiro P. "Lo Que Hemos Aprendido De La (Re)Evolución De Las 4.0: Agricultura De Transporte E Industria Alimentaria", IX Congreso CyTA/CESIA, E.T.S. de Ingeniería Agronómica, Alimentaria y de Biosistemas (UPM), Madrid, España

[3] Cruz M. "Las tecnologías IoT dentro de la industria conectada: internet of thing", EOI Esc.Organiz. Industrial, 2015

[4] Zarazaga J. “Agricultura 4.0: Las tecnologías de la industria 4.0 aplicadas al campo”, https://www.geoslab.com/es/blog/agricultura-40-las-tecnologias-de-la-industria-40-aplicadas-al-campo,2017.

[5] Boyd C. “Water quality in ponds for aquaculture. Alabama Agricul- tural Extension”,Auburn University,1990

[6] C. Boyd., “Consideraciones sobre la calidad del agua y del suelo en cultivos de camarón,” *Dep. Fish. Allied*, pp. 1–30, 2001

[7] White P., “Environmental consequences of poor feed quality and feed management,” *On-farm Feed. Feed Manag. Aquac.*, no. C, pp. 553–564, 2013.

[8] Maulana Y.,“Online monitoring of shrimp aquaculture in Bangka Island using wireless sensor network,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 8, no. 2, pp. 358–364, 2018.

[9] Rao S, “Design of low-cost autonomous water quality monitoring system,” *Proc. 2013 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2013*, pp. 14–19, 2013

[10] Glasgow H., “Real-time remote monitoring of water quality: A review of current applications, and advancements in sensor, telemetry, and computing technologies,” *J. Exp. Mar. Bio. Ecol.*, vol. 300, no. 1–2, pp. 409–448, 2004.

[11] Boyd C., “Water Quality for Pond Aquaculture,” *International Center Experimental Station*. p. 37, 1998.

[12] N. Murali, “IoT ENABLED MONITORING AND CONTROL OF WATER DISTRIBUTION NETWORK,” pp. 1–8, 2018.

[13] Liu Y., “A solar powered long range real-time water quality monitoring system by LoRaWAN,” *2018 27th Wirel. Opt. Commun. Conf. WOCC 2018*, pp. 1–2, 2018.

[14] Maulana Y.,“Online monitoring of shrimp aquaculture in Bangka Island using wireless sensor network,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 8, no. 2, pp. 358–364, 2018.

[15] Cao-Hoang T., “Design of a Cost Effective Soil Monitoring System to Support Agricultural Activities for Smallholder,” *J. Inf. Commun. Technol. Digit. Converg.*, vol. 2, no. 2, pp. 1–5, 2017.

[16] Payero J, “Development of a Low-Cost Internet-of-Things (IoT) System for Monitoring Soil Water Potential Using Watermark 200SS Sensors,” *Adv. Internet Things*, vol. 07, no. 03, pp. 71–86, 2017.

Ingeniería y Automatización, Capítulo 26, pp. 336 - 351.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Plataforma de Bajo Consumo de Energía para la Capacitación en Sistemas Satelitales

González Chávez Miguel Alejandro,
Ramírez Villa Goretti,
Traslosheros Michel Alberto,
Soto Hernández Enrique,

Universidad Aeronáutica de Querétaro.
Posgrado.
Colon, Querétaro, México.

a.g@live.com.mx
goretti.rv@gmail.com.
alberto.traslosheros@unaq.edu.mx
enrique.soto.hdez@gmail.com

Resumen

Con el propósito de generar nuevas líneas de conocimiento que impulsen el desarrollo del sector espacial nacional, la Universidad Aeronáutica en Querétaro (UNAQ) se propone la formación profesionales e investigadores especializados en sistemas de comunicación satelital a través de un plataforma educativa para el análisis y procesamiento de señales satelitales con elementos electrónicos de grado espacial, un módulo de software didáctico y el diseño de prácticas y capacitación continua del personal. En este artículo se presenta la arquitectura propuesta para la plataforma educativa, así como los procedimientos para la capacitación de los estudiantes mediante esta.

Palabras clave: plataforma educativa, tecnología aeroespacial, FPGA, VHDL, análisis de señales.

1. Introducción

El presente trabajo está siendo desarrollado dentro del entorno académico del área aeroespacial con el objetivo de capacitar al estudiante en el manejo y programación de dispositivos electrónicos de grado espacial, así como el análisis de sus señales correspondientes; lo que permitirá sentar las bases para la construcción de un Flat-Sat institucional, el cual específicamente, muestre de manera sencilla los principios de funcionamiento de cada subsistema y las consideraciones importantes a tener en cuenta durante el proceso de construcción.

Lo anterior, bajo el control de un dispositivo FPGA (Field Programmable Gate Array), los cuales nos dan flexibilidad en el diseño de sistemas digitales con fines específicos, pudiendo procesar datos digitales de acuerdo con las necesidades específicas de la aplicación. Estos dispositivos no solo han proporcionado herramientas rápidas de creación de prototipos e implementación para la industria, sino que también los convierten en un buen candidato para investigar su aplicación hacia una plataforma satelital para la educación en ingeniería, misma que necesita consistentemente plataformas de implementación que sean fáciles de usar y se puedan aplicar para la demostración de varios conceptos.

Por lo tanto, esta implementación tiene como objetivo permitir a los estudiantes utilizar los componentes como el convertidor analógico a digital (ADC) y el convertidor digital a analógico (DAC), el transceptor, entre algunos otros, mismos que permitirán diseñar un módulo de los varios subsistemas que requiere un Fla-Sat. Todos estos en interconexión con las entradas a un FPGA, lo que permite a los educadores y estudiantes llevar a cabo experimentos de laboratorio para aprender a utilizar esta tecnología con la idea fundamental de que en un futuro sea utilizada para el análisis de señales de tipo satelital.

Puntualizando lo anteriormente descrito y sabiendo que el estado de Querétaro es uno de los semilleros nacionales de desarrollo de la industria aeroespacial, la disponibilidad de recursos humanos de alto nivel se hace particularmente importante cuando se hacen expertos en el manejo de hardware y software de este sector. Por ello que surge la necesidad de desarrollar mecanismos que faciliten y contribuyan a la formación de personal capacitado de la industria aeroespacial en México.

2. Antecedentes

En cualquier proceso aeroespacial se analizan y controlan una serie de variables que determinan la operatividad de todo el sistema. Desde la perspectiva didáctica, a nivel universitario, es necesario disponer de entornos de aprendizaje donde se pueda comprobar la funcionalidad de pequeños sistemas que complementados con entornos de simulación ofrezcan una aproximación al mundo real y en los que se pueda llevar a cabo una estrategia de probada eficacia como es la de aprender haciendo uso del diseño e implementación del entorno.

Un ejemplo de lo anterior, es el micro satélite desarrollado en el Instituto de Sistemas Espaciales de la Universidad de Stuttgart´s, en Alemania en el año 2017, el cual tuvo como objetivo lanzar el micro satélite con una computadora basada en un FPGA para realizar diferentes observaciones científicas. Los desarrolladores de este proyecto, basaron la elección de un FPGA, debido a la imperiosa necesidad de desarrollar el trabajo con nuevas tecnologías, la necesidad de dispositivos de alto rendimiento, gran velocidad de procesamiento y la facilidad programación.

Los compiladores de software a hardware permitieron generar directamente la configuración lógica del FPGA, lo que hizo que fuera capaz de combinar la flexibilidad del software con el alto rendimiento de hardware. El uso de un FPGA para este proyecto implicó que los algoritmos de control y navegación se implementarán como un hardware en lugar de ejecutarse como un software. Este diseño proporcionó un amplio paralelismo y tiempos muy precisos, al permitir la sincronización precisa entre rutinas,[1].

Otro interesante ejemplo con aplicación muy similar de los FPGA, fue el desarrollado por la Universidad de Colorado en los Estados Unidos de América, entre los años 2015 y 2016, en donde se deseaba explorar la superficie de Júpiter, por lo cual esta universidad propuso desarrollar herramientas de observación y comunicación de bajo costo, con las cuales se podrían recopilar datos mediante sondas que ellos llamaron “Ball Aerospace”.

El desarrollo se centró en la recopilación de datos, el desarrollo de software en un FPGA y la integración de sistemas. Se seleccionaron dos instrumentos científicos relevantes, un magnetómetro triaxial y un contador Geiger, y se integraron con un paquete de aviónica, que se centra en el manejo de datos y comandos basados en FPGA, el cual está integrado con un transceptor para comunicarse por RF a una estación terrestre, simulando un orbitador de Europa.

El prototipo recolecta datos por un total de 100 horas y transmite esos datos en periodos de 8 minutos, lo que demuestra la viabilidad de utilizar sondas pequeñas y rentables para explorar la superficie de Europa u otros cuerpos planetarios, [2].

Ahora bien, centrándonos en el proyecto en curso, la plataforma educativa para la capacitación en sistemas satelitales el dispositivo base es un FPGA que como es sabido y se explicará más

adelante, es un componente digital usado para el procesamiento de señales, al igual que los microprocesadores, de los cuales podemos destacar su versatilidad, costo y tamaño reducido, mientras que, de los FPGA se destaca su capacidad de procesado en paralelo, que le permite tener un buen rendimiento en velocidad al realizar diversas operaciones de forma simultánea, [3].

Se ha resaltado que los sistemas lógicos programables son dispositivos digitales que incluyen una gran cantidad de componentes lógico diseminados en un circuito integrado sin una función determinada. Luego, bajo el planteo de un proyecto y herramientas adecuadas se interconectan para obtener una funcionalidad específica. Existe una gran variedad de estos dispositivos, de los cuales sobresalen dos tipos principales que son los CPLD o Complex Programable Logic Devices y las FPGA o Field Programmable Gate Array.

En un circuito lógico se puede implementar cualquier tipo de función digital, desde sistemas combinacionales sencillos hasta complejos autómatas secuenciales. Ambos, CPLDs y FPGAs incluyen un gran número de elementos lógicos programables o mejor dicho configurables. La densidad de puertas es mayor en las FPGAs que en los CPLDs. La diferencia entre ellos es la arquitectura, ya que los CPLDs tienen una organización rígida y poco flexible, consistente en sumas de productos programables asociados a algunos sistemas secuenciales elementales. Los FPGAs, por su parte están constituidas por un gran número de pequeños bloques digitales, dónde se producen operaciones lógicas simples vinculadas a bloques secuenciales básicos, todos ellos interconectados formando complejas estructuras lógicas, lo que facilita una interconexión entre estos bloques y otorga a los FPGA una gran flexibilidad,[4].

Por lo tanto, son sistemas altamente integrados que suelen ocupar muy poco espacio y tener un consumo de potencia muy reducido. Una vez desarrollados los FPGA (Field Programmable Gate Array) fue posible reducir aún más los sistemas embebidos, lo que permitió llegar a los sistemas SoC(system on chip, o sistema en chip), es decir todo un sistema compuesto de procesador, memoria, y circuitos de apoyo personalizados, pueden entrar en un solo microchip, [5].

Los sistemas embebidos son sistemas informáticos de uso específico, construidos dentro de un dispositivo mayor, donde la mayoría de los componentes se encuentran incluidos en la placa base. Un sistema embebido consiste en un sistema con microprocesador cuyo hardware y software están específicamente diseñados y optimizados, para resolver un problema concreto eficientemente. Muchas veces un sistema embebido es un componente de un sistema mucho más grande.

Los sistemas embebidos simples, cuentan con un microprocesador, memoria, unos pocos periféricos de entrada/salida y un programa dedicado a una aplicación concreta, almacenado permanentemente en la memoria para cumplir con los requisitos de la aplicación concreta.

Básicamente los sistemas embebidos están formados por un microprocesador y un software que se ejecute sobre este. El software requiere un lugar donde almacenarse para luego ser ejecutado por el procesador, el cual puede ser en memoria RAM o ROM o incluso dentro del mismo chip del procesador, los cuales comúnmente cuentan con registros de 8 o 16 bits. Además de que contará con salidas y entradas útiles para comunicarse.

El programa destinado a gobernar una aplicación determinada es el único residente de la memoria, sus líneas de Entrada/Salida soportan la conexión de los actuadores del dispositivo a controlar y recursos complementarios disponibles. Estas son las únicas características que tienen en común los sistemas embebidos ya que todas lo demás será diferente para cada sistema embebido en particular.

Los FPGA ofrecen una plataforma que soporta núcleos de procesamiento embebidos (Co-procesadores, periféricos o microprocesadores), aunado a esto incorporan componentes digitales (memoria, multiplicadores, transceptores y muchos más) en un solo chip. La posibilidad de realizar diseños de alta densidad han hecho del FPGA una herramienta idónea y flexible para programar arquitecturas digitales personalizadas, lo que la hace al mismo tiempo viable para desarrollar varias

aplicaciones, por ejemplo, procesamiento de señales, audio, video, multimedia, comunicaciones, cómputo de alto desempeño, adquisición y procesamiento de señales, entre otras.

La integración de componentes digitales, procesadores e interfaces de comunicación en un solo dispositivo han hecho viable elegir al FPGA para el desarrollo de la presente plataforma educativa. Además de contar con la disponibilidad de herramientas de software para diseñar, simular, sintetizar, probar y programar al FPGA, lo que permite a los ingenieros y programadores elegir la metodología que mejor se ajuste a su estilo de trabajo e intereses; aunado a esto, el migrar entre ambientes de desarrollo de fabricantes de FPGAs es relativamente simple ya que el proceso de diseño es similar,[6].

El lenguaje por medio del cual será posible la programación de los FPGA, es conocido como un lenguaje de descripción de hardware o HDL (Hardware Description Language), el cual es un lenguaje de especificación que permite construir modelos a través de la descripción de los sistemas sobre los cuales se aplica.

Estos se utilizan para definir la estructura y funcionalidad de dispositivos electrónicos digitales ya que VHDL resulta ser el acrónimo que representa la unión de VHSIC y HDL, siendo la primera inicial representativa de Very High Speed Integrated Circuit y el segundo de Hardware Description Language.

Por tanto, VHDL permite la descripción formal de un circuito electrónico digital, y posibilita su análisis automático, simulación y en algunos casos hasta la síntesis. Estos lenguajes de descripción de hardware se parecen mucho a lenguajes de programación de procesadores. Estos consisten en una descripción textual con expresiones, declaraciones y estructuras de control, el cual fue definido por la IEEE a pedido del Departamento de Defensa de Estados Unidos para solucionar el grave problema de especificación de los sistemas electrónicos digitales y lograr equivalencia entre los diferentes fabricantes.

Para los diseños complejos como en la programación de software los programas son generalmente jerárquicos y VHDL permite establecer “capas graduales”, por lo que se establece un entorno adecuado de trabajo definiendo los diversos módulos que integran el sistema y sus interfaces, dejando para etapas posteriores la realización de los detalles.

Una vez realizado el código en el lenguaje para cada módulo, interfaces y el resto de las especificaciones de los circuitos, se compila el programa adecuando la sintaxis y revisando compatibilidad con otros módulos, produciendo así toda la información necesaria para la simulación.

La simulación por su parte permite establecer la excitación adecuada a cada módulo y verificar la respuesta, de manera tal que comprueba el funcionamiento del circuito y comprueba si se comporta como era lo esperado, donde se verifican dos características:

Funcionalidad arquetipo: Se analiza el comportamiento lógico como un circuito ideal, es decir considerando tiempo de propagación cero.

Funcionalidad real: Se verifica el comportamiento colocando los circuitos reales con sus correspondientes tiempos de propagación.

Completada la fase de verificación se procede a la finalización del diseño.

1.- Síntesis: convierte la descripción en VHDL en un conjunto de componentes que pueden ser realizados en la tecnología seleccionada.

2.- Capacidad: Se ajusta la necesidad del proyecto a los dispositivos configurables seleccionados.

3.- Revisión Final: Implica una comprobación temporal con los circuitos seleccionados y con los que se producirá el proyecto.

El lenguaje VHDL fue organizado dentro del paradigma de la programación estructurada, por lo que permite escribir programas en forma sencilla y eficiente. El planteo de un programa tiene dos partes bien diferenciadas: Primero la declaración del conexiones externas y luego la funcionalidad,[4].

3. Metodología

Este proyecto está dirigido a principiantes en el estudio de la tecnología espacial, particularmente a aquellos interesados en el análisis, diseño, programación y construcción de satélites pequeños. Por lo que para el desarrollo de los subsistemas de esta plataforma educativa se inició trabajando con dispositivos de grado industrial para desarrollar los protocolos de comunicación, la configuración y la documentos para la capacitación y entrenamiento, haciendo énfasis en aquellas necesarias para el control y el aseguramiento de la calidad y de los estándares del sector aeroespacial, de manera tal que se establezcan las bases para migrar a dispositivos de grado espacial.

El siguiente cuadro muestra la metodología utilizada para el desarrollo de la primera etapa y las actividades que involucran su aplicación. Hoy en día nos encontramos finalizando la primera etapa de este proyecto, formulando y desarrollando la capacitación y los manuales que darán la pauta de una capacitación continua. Así mismo se mencionan las actividades que son requeridas para finalizar el proyecto de manera exitosa, mismas que serán desarrolladas en la segunda etapa del proyecto.

METODOLOGÍA	ACTIVIDADES PRINCIPALES	ETAPAS
Observación de las necesidades del sector e interpretación de la problemática a abordar.	Desarrollo de la propuesta del proyecto.	Preliminar
Propuesta de solución y planificación de la investigación.		
Formulación teórica	Revisión del estado del arte de los sistemas embebidos, microcontroladores y normativa aplicados a satélites.	Primera etapa
Selección de herramientas y materiales.	Selección de dispositivos de grado espacial.	
Diseño de la investigación y del desarrollo tecnológico.	Formulación de un plan de capacitación y desarrollo de material didáctico.	
Desarrollo de capacidades y recursos humanos.	Capacitación del grupo de trabajo del proyecto y personal de las instituciones participantes. Desarrollo de manuales de cada curso impartido. Diseño de prácticas y sistemas didácticos para la transferencia de tecnología y desarrollo de recursos humanos.	Segunda etapa
Desarrollo de la solución (plataforma educativa)	Diseño de desarrollo y fabricación de módulo de procesamiento de señales para un satélite pequeño.	
	Diseño y desarrollo de módulos de software didáctico.	
Documentación de información y generación de conclusiones.	Desarrollo de un plan de trabajo para la capacitación continua de personal y alumnos (posterior al desarrollo del proyecto)	
	Diseño de un plan para la actualización continua.	

Tabla 1. Metodología y actividades relacionadas.

.

4. Arquitectura propuesta

La arquitectura propuesta para la plataforma educativa, la cual se ilustra en el diagrama a bloques de la figura 1, está integrado por varios dispositivos, los cuales se describen brevemente en los siguientes párrafos. En primer lugar, el control de la plataforma lo tiene un FPGA de grado espacial tolerante a la radiación, de la marca MicroSemi, que ofrece 151,824 elementos lógicos en un empaquetado cerámico con 1,657 pines.

La placa de desarrollo del FPGA, incluye dos memorias de 1 GB DDR 3 (double data rate) y dos memorias flash SPI de 1 GB. La placa también tiene varios periféricos estándar y avanzados, como el conector PCIe (x4), dos conectores FMC (Flexible Microstrip Connectors) para usar varias tarjetas secundarias estándar, USB, circuito interconectado Philips (I2C), puerto Ethernet gigabit, interfaz de periféricos seriales (SPI) y UART. Un circuito amplificador operacional de alta precisión en la placa, que ayuda a medir el consumo de energía del núcleo del dispositivo, Ver figura 2. Además

cuenta con un programador de FlashPro, que está integrado en la placa, lo que permite la programación RTG4 FPGA a través de la interfaz JTAG,[7].

.

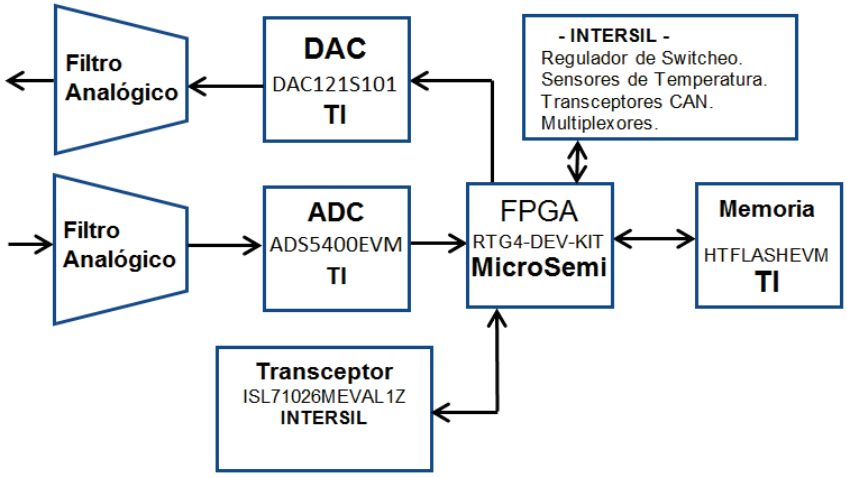


Figura 1. Bloques funcionales propuestos.

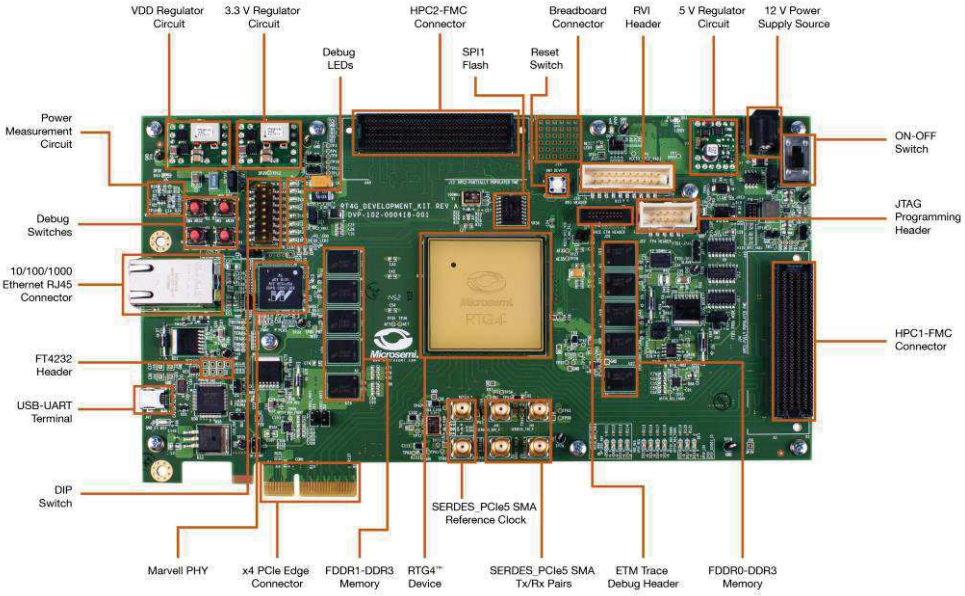


Figura 2. RTG4 FPGA DE MicroSemi.

El ADC por su parte, es el ADS5400EVM de la marca “Texas Instruments”, este módulo de evaluación está diseñado para la conversión de señales de hasta 2 GHz de frecuencia de entrada y muestreo de hasta 1GSPS, tiene un excelente rendimiento de bajo ruido y un rango dinámico libre de espurias, tiene una resolución de 12 bits y un rango de temperatura de -40 a 85 °C, en un empaquetado de HTQFP-100, con lo cual en combinación del empaquetado y el consumo moderado del ADS5400 permite la operación sin un disipador de calor externo, ver figura 3, [8].

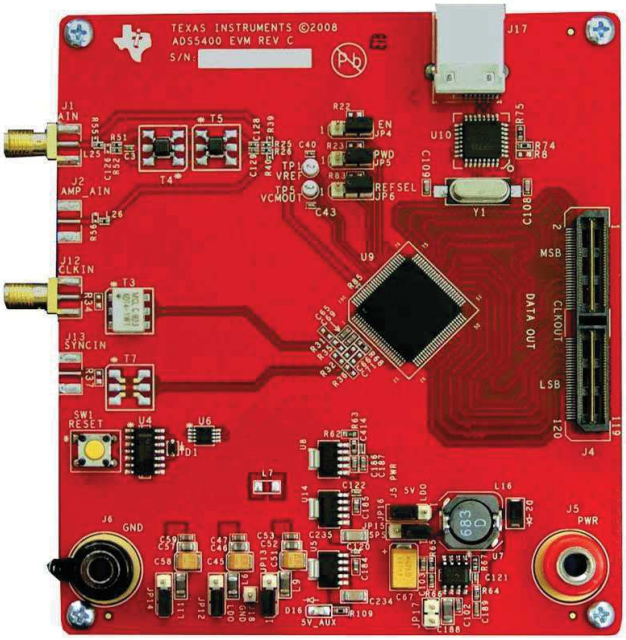


Figura 3. ADS5400.

El dispositivo DAC121S101 es un convertidor digital a analógico (DAC) también de la marca “Texas Instruments” de salida de voltaje de 12 bits, con un suministro de 2.7 a 5.5 V y consume solo 177uA de corriente a 3.6V. Opera a velocidades de reloj de hasta 30MHz y es compatible con interfaces estándar SPI, QSPI, MICROWIRE y DSP. El bajo consumo de energía de los DAC121S101 lo convierten en una excelente opción para su uso en equipos que funcionan con baterías, ver figura 4,[9].

Figura 4. DAC121S101EVM.

El ISL71026M es un transceptor CAN de 3.3V que es compatible con el estándar ISO11898-2 para su uso en sistemas de comunicación en serie de la red de área de controlador (CAN). El transceptor realiza funciones de transmisión y recepción entre el controlador CAN y el bus diferencial CAN. Puede transmitir y recibir a velocidades de bus de hasta 1Mbps. Fue diseñado específicamente para operar en ambientes hostiles como el espacio, ver figura 5, [10].

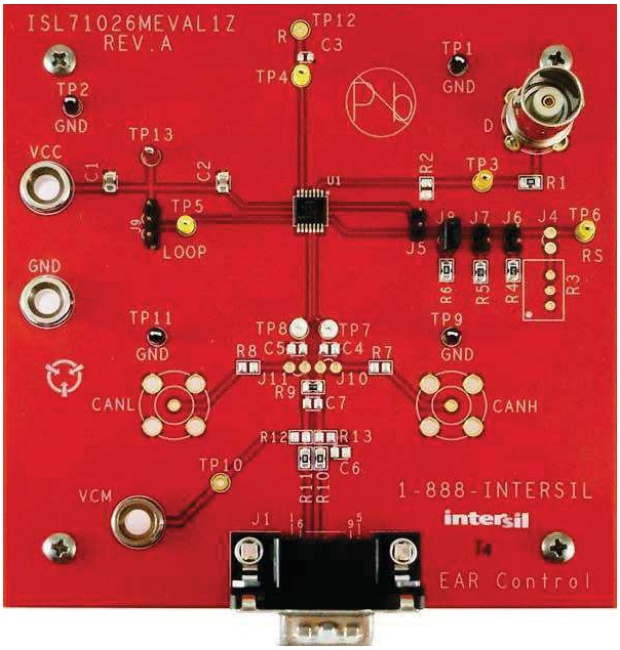


Figura 5. ISL71026MEVAL1Z-Transceptor.

El módulo HTFLASHEVM está diseñado para la verificación del SM28VLT32 en entornos extremos, la cual es una memoria flash de 32 megabit diseñada para almacenar código de programa y / o datos de adquisición. El amplio rango de temperaturas que soporta este módulo lo hace ideal para aplicaciones que necesitan funcionar de manera confiable en entornos hostiles, ver figura 6, [11].

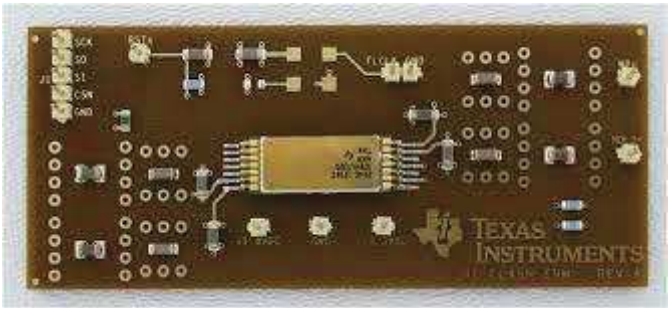


Figura 6. HTFLASHEVM - Memoria Flash.

La memoria SM28VLT32 cuenta con una interfaz de periféricos seriales (SPI), lo que permite mayor confiabilidad, fácil ensamblaje en estas aplicaciones y para leer/escribir/borrar datos en la memoria flash. La memoria está dividida en sectores, para proporcionar el direccionamiento de palabras de 2-M de 16 bits cada una. Por lo tanto, la dirección de memoria tiene 21 bits de ancho,[12].

Las características descritas, en los párrafos anteriores, sobre los dispositivos seleccionados, nos permiten sentar las bases de una plataforma educativa en sistemas satelitales, y aun cuando no todos los dispositivos de grado espacial, si nos permiten capacitar a los alumnos en el manejo de estos, su configuración, su programación y en el análisis de señales.

Es por ello que es de suma importancia, para la capacitación en sistemas satelitales, la amplia documentación que se está generando con los rudimentos necesarios, para la comunicación del FPGA con las diferentes tarjetas de desarrollo de la plataforma, así como para la configuración física y

lógica de los mismos. Un claro ejemplo es el de la comunicación I2C, la cuál fue programada y se están desarrollando pruebas en el laboratorio de sistemas embebidos de la universidad, como lo podemos observar en la figura 7, la cual muestra imágenes del osciloscopio con el "handshake" o "establecimiento de comunicación" del FPGA con otro dispositivo mediante el protocolo de comunicación I2C; trabajo que fue desarrollado por el mismo autor del artículo citado [13] y que también pertenece a este proyecto en curso.

En la figura 8, se muestra el sintetizado de una computadora básica incluyendo el módulo personalizado Bus I2C en lenguaje Verilog HDL. En la figura 9, se muestra el control de la recepción/transmisión de señales del módulo personalizado del bus I2C en lenguaje C.

Altera provee el software Quartus II como soporte de sus dispositivos lógicos programables. Esta herramienta de desarrollo se referencia como "herramienta de diseño totalmente integrada" [14], ya que abarca todas las aplicaciones necesarias para realizar un proyecto en FPGA, sin requerir ninguna otra aplicación adicional. El proceso de diseño del SoPC, involucra el diseño de hardware y del software asociado. El hardware del SoPC se realizó integrando los componentes de biblioteca conforme a los atributos de organización del sistema. Simultáneamente, el desarrollo del software se realizó acorde a la arquitectura de hardware del sistema. La figura 10, muestra las herramientas propuestas para el flujo de diseño del sistema. La descripción de sistemas embebidos se desarrolló con la herramienta Qsys y el software se realizó mediante la herramienta Nios II SBT Eclipse.

Dicho lo anterior, se resalta el hecho de que cada uno de estos documentos que se formulan servirán como manuales y bibliografía básica del personal bajo capacitación, por lo que será necesario su dominio y entendimiento antes de avanzar a las diferentes etapas de prácticas de laboratorio.

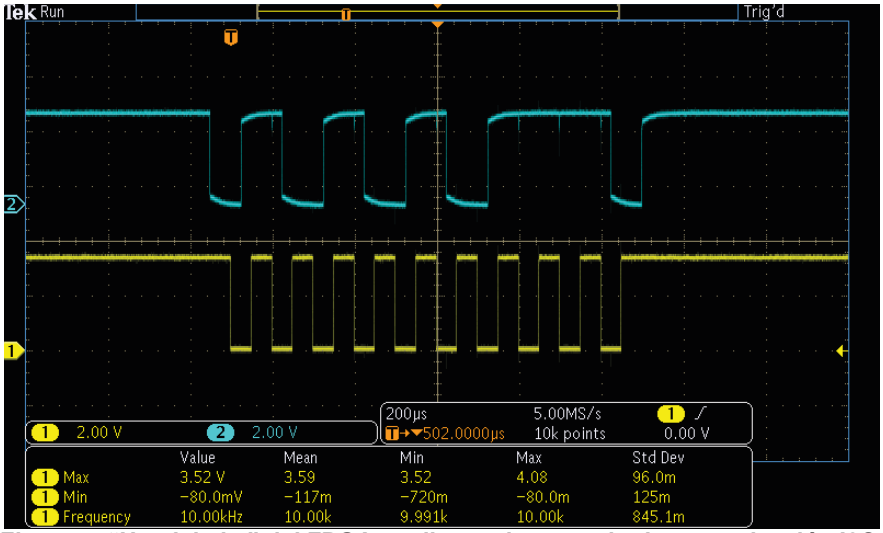


Figura 7. "Handshake" del FPGA mediante el protocolo de comunicación I2C.

Ingeniería y Automatización, Capítulo 26, pp. 336 - 351.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

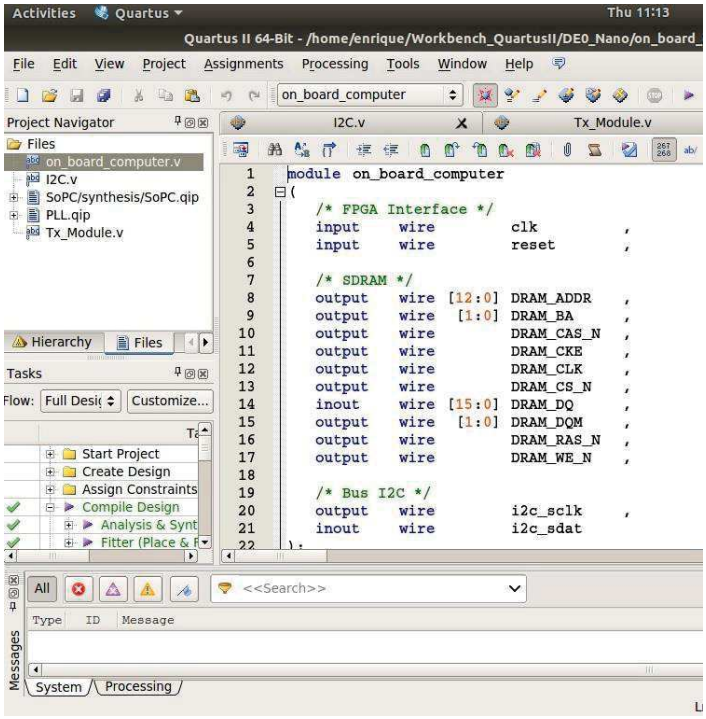


Figura 8. Sintetizado de una computadora básica y Bus I2C.

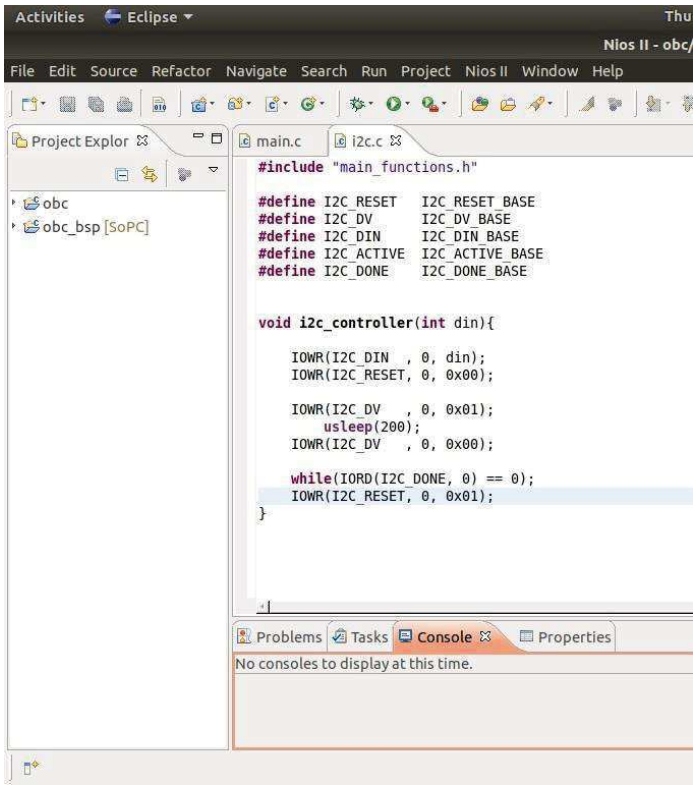


Figura 9. Control de la recepción/transmisión del bus I2C.

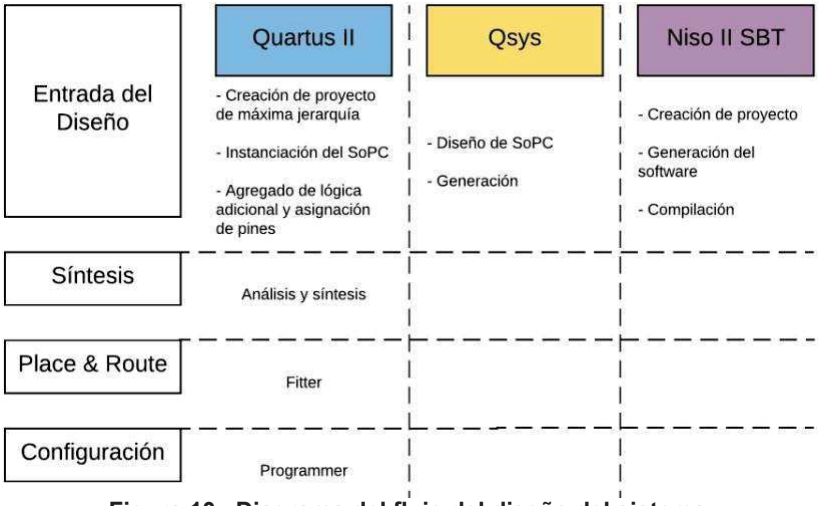


Figura 10. Diagrama del flujo del diseño del sistema.

4.1 Capacitación y prácticas.

La industria aeroespacial en México y en el mundo presenta retos y oportunidades de investigación a los cuales se les deben invertir tiempo y recursos para dar cabida al desarrollo tecnológico. Hoy en día la Universidad Aeronáutica en Querétaro no cuenta con un curso de entrenamiento en dispositivos de grado espacial que permita a los estudiantes tener una guía e incursionar en el desarrollo de hardware y software y así explotar los beneficios de dichos conocimientos en un ambiente laboral o de investigación, por lo que el desarrollo de este proyecto mediante actividades teóricas y prácticas permitirá satisfacer esas áreas de oportunidad.

Hoy en día se cuenta con un grupo de trabajo multidisciplinario, el cual ha estado recibiendo capacitación relativa a las áreas e intereses que circundan el proyecto actual, entre los cuales destaca el de la tarjeta de desarrollo “NanoBoard 3000” con un FPGA de la marca Altera, Cyclone III, el cual es programado mediante el software de la marca Altium. Esto ha permitido formar un amplio equipo de trabajo, que a su vez ha impartido capacitaciones a pequeños grupos de estudiantes y detonado el interés y desarrollo de prácticas como herramienta de enseñanza/aprendizaje, las cuales pueden ser incluidas en los programas educativos de la universidad.

Lo anterior, nos ha permitido experimentar y entender que para implementar una enseñanza productiva y perdurable, es necesario aplicar un método eficaz y una estrategia de evaluación adecuada. En este tipo de aprendizaje se establece una metodología en la que el estudiante además de recibir los contenidos verbalmente, descubre por sí mismo los conceptos y sus relaciones. Así acomoda y adapta a su esquema de conocimientos lo visto en clase, perfeccionando las destrezas de investigación, del método inductivo de aprendizaje y la solución de problemas. Si bien este método está orientado para el aprendizaje de la metodología científica y realización de experimentos de laboratorio y aplicable al estudio y análisis de su estructura de hardware y proceso funcional del software.

Para el desarrollo de estas prácticas se tomó en consideración que los alumnos cuenten ya con conocimientos previos, tanto teóricos como experimentales en las áreas de sistemas digitales, conocimientos que serán la base sobre la cual se inicie este proceso de enseñanza. Así mismo, el estudiante debe conocer los procedimientos de observación, búsqueda, control, verificación de resultados y medición de variables para facilitar el desarrollo y avance de las prácticas de acuerdo con lo planeado en cada sesión.

En el proceso de capacitación en sistemas satelitales el instructor deberá guiar y supervisar el correcto desarrollo de las prácticas dando indicaciones de los pasos a seguir en cada punto de estas. El usuario deberá mostrar al instructor los resultados obtenidos en cada punto antes de pasar al punto siguiente.

Los objetivos y metodología deben ser atrayente y entusiasmar a los alumnos para que se involucren en el aprendizaje por descubrimiento, mismo que debe ser restringido y bien delimitado al tratar cada una de las prácticas como temas individuales.

Cada una de las prácticas guiadas de este entrenamiento, están formuladas de la siguiente manera:

- ✓ Carátula.
- ✓ Introducción.
- ✓ Marco teórico.
- ✓ Descripción de la práctica.
- ✓ Listado de material.
- ✓ Código base en lenguaje VHDL.
- ✓ Diagrama esquemático propuesto.
- ✓ Sección de evaluación al estudiante.
- ✓ Guía básica para la programación de la tarjeta de desarrollo (FPGA).
- ✓ Referencias.

La caratula describe de manera breve la práctica, los objetivos y/o actividades a desarrollar en la misma, además de que se convierte en una herramienta de evaluación que el instructor podrá tener como evidencia para respaldar el historial académico de sus estudiantes, lo cual le da formalidad y presentación a la capacitación, ver figura 7.

UNAQ

Prácticas de laboratorio.
Agosto de 2018

UNIVERSIDAD AERONÁUTICA DE QUERÉTARO.
DEPARTAMENTO DE POSGRADO.

FORMATO DE PRÁCTICAS DE LABORATORIO.

Fecha: / /

CARRERA.	CUATRIMESTRE.	MATERIA.

PRACTICA #	DURACIÓN:
5.-Maquina de estados - Dado.	30 min.

1.-INTRODUCCIÓN.

Se denomina maquina de estados a un modelo de comportamiento de un sistema con entradas y salidas, en donde las salidas dependen no sólo de las señales de entradas actuales sino también de las anteriores.

Las maquinas de estados se definen como un conjunto de estados que sirve de intermediario en esta relación de entradas y salidas, haciendo que el historial de señales de entrada determine, para cada instante, un estado para la máquina, de forma tal que la salida depende únicamente del estado y las entradas actuales.

2.-OBJETIVOS.

Con la finalidad de familiarizarse con la plataforma educativa y el manejo del FPGA, el estudiante pondrá en práctica la programación en VHDL implementando estructuras comunes dentro del lenguaje.

FORMULÓ.	REVISÓ.	APROBÓ.	AUTORIZÓ.
Ing. M. Alejandro González Chávez.	Ing. Goretti Ramírez Villa.	Dr. Alberto Trasiosheros Michel.	Ing. Ángel Ramírez Vázquez.

Figura 7. Ejemplo del formato de prácticas.

En la introducción centramos al lector, sea el estudiante o un revisor, en las ideas y conceptos principales para las cuales estamos desarrollando esta práctica, siendo puntuales, descriptivos y concisos.

El marco teórico, nos permite sentar las bases de la teoría que respalda el desarrollo de cada práctica, es una sección breve pero completa, para encaminar al estudiante a conocer los criterios, principios, teoremas o leyes bajo las cuales el desarrollo de la práctica está siendo propuesta, además de que servirá como respaldo para temas cubiertos en clases por el instructor o materias anteriores.

La descripción de la práctica, nos permite detallar la información sobre los procedimientos y requisitos que deberá cubrir el alumno con respecto a las actividades solicitadas como medio de aprendizaje en el desarrollo de la práctica, de manera tal, que el estudiante pueda consultarlos y en conjunto con el marco teórico entender, razonar y aprender de manera correcta.

El listado del material, dará al alumno y al instructor la información y certeza de que cada práctica se hará con los aditamentos necesarios para las actividades o retos descritos para la práctica, evitando pérdida de tiempo en explicaciones y conteos innecesarios dentro del periodo de evaluación.

El código base de lenguaje VHDL, podrá ser o no añadido a este documento, eso dependerá de la práctica a trabajar, ya que en etapas avanzadas el alumno será quien proponga soluciones (de manera parcial o total) a los requerimientos de la práctica mediante programación en el lenguaje VHDL, etapa que será evaluada (si así se requiere) a consideración del instructor.

El diagrama esquemático propuesto, es la representación gráfica de la solución dada al problema o ejercicio propuesto en el lenguaje VHDL, este refleja (a manera de diagrama de bloques) las conexiones físicas hechas en la plataforma de aprendizaje y que de forma didáctica los estudiantes podrán corroborar de manera física con osciloscopios, multímetros o como corresponda. Los esquemáticos podrán ser o no añadidos a este documento, eso dependerá de la práctica a trabajar, ya que en etapas avanzadas el alumno será quien proponga soluciones (de manera parcial o total) a los requerimientos de la práctica, etapa que será evaluada (si así se requiere) a consideración del instructor.

La sección de evaluación al estudiante, servirá como herramienta para que el instructor pueda corroborar y calificar el rendimiento y aprovechamiento del estudiante, el conocimiento alcanzado y de esta manera hacerlo acreedor a una calificación con base en sus respuestas. Cada práctica tendrá un numero diferente de preguntas, ejercicios o retos que el estudiante deberá cubrir, así como, al final, dar una retroalimentación al instructor de las conclusiones que esta actividad le deja en su camino a aprender a utilizar el hardware y software de la plataforma satelital.

La guía básica es una descripción detallada de los procedimientos de programación de la plataforma del FPGA, permitirá al estudiante comprender y memorizar los pasos base para una correcta programación de la plataforma. Esta será una constante en cada documento de práctica, pero está ubicada intencionalmente al final del documento para dejar a elección del interesado si es necesario imprimirla o no, ya que la parte medular del documento es la práctica a desarrollar, ver figura 8.

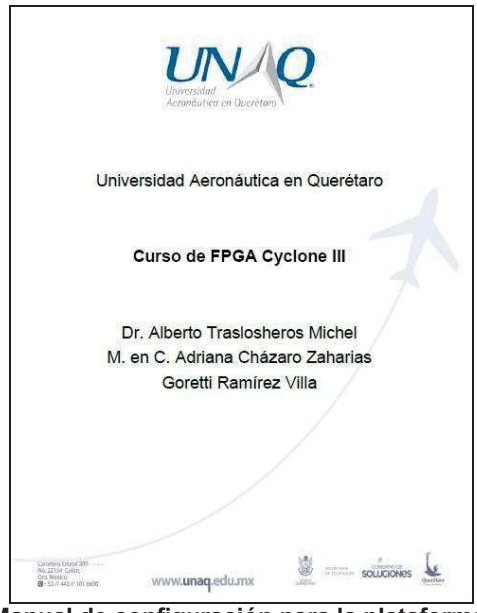


Figura 8. Manual de configuración para la plataforma del FPGA.

Las referencias bibliográficas son el origen del marco teórico y servirán al estudiante para completar su preparación y conocer los tópicos de manera más profunda, con respecto a lo expuesto por el instructor.

De esta manera, cada una de las etapas del formato de prácticas permitirá al instructor capacitar adecuadamente a los estudiantes, iniciando con una amplia y adecuada explicación teórica del tema a tratar, de los dispositivos de la plataforma a ocupar y del código en VHDL adecuado para la práctica en turno. También, como ya se dijo, servirá como herramienta de evaluación e historial, siendo por tanto una herramienta muy completa para la capacitación de sistema satelitales.

Para los estudiantes, el documento de prácticas es una herramienta teórica y práctica, pero aunado a esto, será importante que estudien y trabajen con la documentación de cada uno de los manuales desarrollados para la configuración y puesta a punto de los dispositivos que conforman la plataforma. La parte teórica deberán complementarla con la bibliografía de apoyo que encontrarán en las referencias de cada documento, ver figura 9.

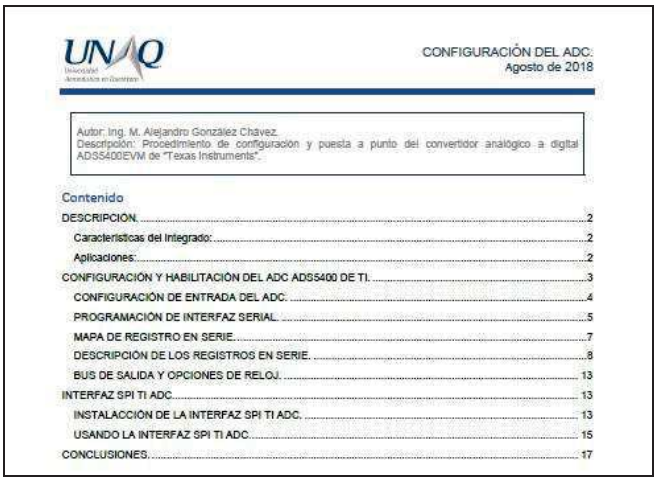


Figura 9. Manual de configuración del ADC.

Además de lo anteriormente descrito y gracias a la capacitación continua ha sido posible el desarrollado de otro tipo de materiales de estudio como son los siguientes:

- ✓ Estado del arte del proyecto – Investigación de los conceptos teóricos y desarrollos que servirán de base para este proyecto.
- ✓ Apuntes de electrónica digital – Notas para introducirse al diseño digital y uso de FPGAs.
- ✓ Implementación del soft-core Nios II en la plataforma DE0/DE0-Nano.
- ✓ Introducción a norma DO-178C – Análisis de las actividades del proceso de ciclo de vida del software contenidas en el estándar DO-178C.

5. Conclusiones

Puntualizando que este proyecto está enfocado para un área estudiantil, se encuentra aún en construcción, sin embargo como se mostró en la sección de metodología, nos encontramos concluyendo la primera etapa del mismo. Se destaca que las prácticas desarrolladas para el entrenamiento en sistemas satelitales centrarán su estructura y tópicos en brindarle al estudiante los conocimientos básicos para la capacitación tanto en el manejo de los diferentes dispositivos electrónicos de la arquitectura propuesta, así como en crear habilidades de programación en lenguaje VHDL.

Es importante decir que toda la documentación que está siendo generada formará parte de un compendio de documentos para la capacitación y entrenamiento en la plataforma educativa y sentará las bases para la construcción de un Flat-Sat, con el cual los estudiantes puedan realizar diversas actividades, como serán, el análisis de señales de uso satelital, prácticas de laboratorio, diseño y pruebas para satélites, entre otras varias. La arquitectura será totalmente escalable y permitirá, debido a la diversidad de protocolos de comunicación, que a futuro se puedan anexar y/o combinar con otras tecnologías y módulos de evaluación con el fin de proponer soluciones a diversos problemas del ámbito aeroespacial.

Además, estas herramientas pueden emplearse como recurso de apoyo a la docencia en la impartición de clases de teoría, aprovechándose del realismo y dinamismo de las experiencias prácticas y de la simulación por computadora. Igualmente pueden utilizarse como valioso apoyo en el laboratorio, donde los alumnos pueden ejercitar sus conocimientos manejando ellos mismos las distintas posibilidades que les brindan y favoreciendo así el aprovechamiento de las clases y la asimilación de la materia.

Se debe destacar también el carácter multidisciplinario de esta plataforma ya que se puede aplicar a numerosos problemas en áreas diversas. Por ejemplo, señales de voz o de audio, imágenes, señales de dispositivos de control, etc. Se pueden también aplicar con distintos enfoques dentro de cada área: para diagnóstico y reconocimiento de señales, para clasificación, para eliminación de ruido y perturbaciones, detección de errores, prevención, etc.

Por lo tanto, el estudiante tendrá una serie de herramientas de programación muy sencillas de manejar, donde puede observar de una manera gráfica lo que está haciendo. Además, la utilización de que esta plataforma le permitirá trabajar en temas de procesamiento abstrayendo toda la problemática de la programación, de la electrónica y de la matemática que hay detrás. El programa se encargará de realizar los cálculos y llamar a las subrutinas de simulación necesarias y además visualizará de forma gráfica la forma de las señales.

La realización de las prácticas con esta plataforma y su software se puede estructurar en varias fases generales, cada una de ellas con distintos objetivos, la metodología propuesta, junto con la supervisión del instructor, proporciona un modo de aprendizaje rápido y eficiente, sobre todo para establecer relaciones entre conceptos vistos en teoría y los experimentos. Y por si fuera poco, permite

la repetición, por parte del estudiante, de las prácticas hasta conseguir el resultado que busca o entender un determinado comportamiento, y le ayuda a avanzar en su estudio de una forma natural, motivándole a preguntar el por qué y cómo del funcionamiento de los distintos tipos de señales, del programa y el conocimiento que se puede extraer de su procesamiento.

Referencias

[1] M. Yasir, “Simulation-Based Testing of Embedded Attitude Control Algorithms of an FPGA based Micro Satellite,” p. 8, 2017.
[2] D. Combs *et al.*, “Project ELSA: Europa Lander for Science Acquisition,” p. 11, 2016.
[3] F. A. Silva Gómez, “Metodología de codiseño hardware-software para procesamiento de señales radar en sistemas embebidos,” Pontificia Universidad Javeriana, 2017.
[4] H. Morales, S. Pérez, D. Argüello, and S. Molina, “Implementación y análisis de prestaciones de un procesador elemental didáctico con VHDL y FPGA.,” no. September, p. 9, 2015.
[5] I. J. Sánchez Cuapio, “Evaluación e implementación de algoritmos de adquisición de una señal GPS, en un dispositivo reconfigurable FPGA,” 2009.
[6] J. I. Guzman Mercado, “Plataforma FPGA reconfigurable para implantar el sistema de control de apuntamiento del satélite SATEDU,” Universidad Nacional Autónoma de México, 2014.
[7] Microsemi, “RTG4 FPGA Development Kit User Guide.” MicroSemi, p. 58, 2017.
[8] T. Instruments., “ADS5400EVM User’s Guide,” no. October. Texas Instruments, 2009.
[9] T. Instruments, “DAC121S101Q1 Digital to Analog Converter.” p. 37, 2015.
[10] Intersil, “ISL71026MEVAL1Z User’s Manual: Evaluation Board,” no. May. Intersil, p. 21, 2017.
[11] Texas Instruments, “HTFLASHEVM User ’ s Guide,” no. November. p. 21, 2012.
[12] Texas Instruments, “SM28VLT32-HT 32-MBIT HIGH-TEMPERATURE FLASH MEMORY WITH SERIAL PERIPHERAL INTERFACE (SPI) BUS,” no. March. 2013.
[13] E. Soto Hernandez, “Verificación e Implementación en hardware del código de redundancia ciclica.,” 2018.
[14] R. Cayssials, *Sistemas Embebidos en FPGA*, AlfaOmega. Argentina, 2014.

Ingeniería y Automatización, Capítulo 27, pp. 352 - 361.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Algoritmo de División Entera para Aplicaciones en Hardware Difuso

César Barrón Romero y Antonio Hernández Zavala

Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Instituto Politécnico Nacional,
Cerro Blanco 141, Col. Colinas del Cimatario. Querétaro, Qro. C.P. 76090.
cbarronr1500@alumno.ipn.mx, anhernandezz@ipn.mx

Resumen

La operación aritmética de división es muy utilizada en varios campos de la ingeniería, esta puede ser llevada a cabo de distintas formas, sin embargo los algoritmos secuenciales en base-2 son generalmente utilizados tanto en software como hardware. Recientemente la utilización de arreglos de compuertas programables en campo (FPGA's) en disciplinas como el computo suave en específico el hardware difuso, requieren de bloques lógicos simples pero con gran potencialidad de procesamiento y altas velocidades de operación. El objetivo de este trabajo es el diseño y desarrollo de un algoritmo de división con operandos enteros, cuyas características sean alta velocidad de procesamiento y bajo consumo de recursos computacionales, que permita ser utilizado en la arquitectura de un procesador difuso que cual sea basado en el procesamiento mediante niveles.
.

Palabras clave: Algoritmo, División, FPGA, Velocidad, Recursos Computacionales.

1. Introducción

La operación de división es la más compleja de las cuatro operaciones aritméticas básicas, este cálculo realizado en hardware consume recursos computacionales e incluso es más lenta que la operación de multiplicación. En los algoritmos de división por recurrencia, el cociente es generado un bit a la vez comenzando por el bit más significativo, es decir se realiza de manera secuencial. El cálculo de la operación de división entera entre un dividendo z de tamaño 2k-bit por un divisor d de dimensión k-bit es llevado a cabo en k ciclos de reloj utilizando corrimientos y restas como se muestra en la figura 1. Sin embargo cuando el tamaño de los operandos es grande, el tiempo o ciclos de reloj para realizar el cálculo llega a ser excesivo.

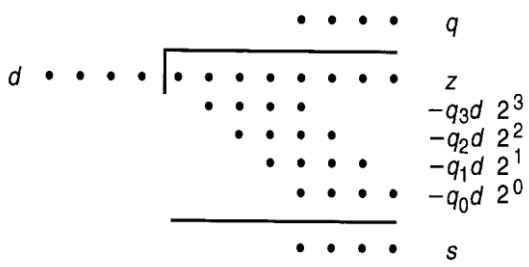


Figura 1. División de un número de 8-bits entre un número de 4-bits.

Cada hilera de puntos en la matriz de bits sustraída es o todos 0 (si la correspondiente $q_i = 0$) o igual que d (si $q_i = 1$), los términos sustraídos se forman uno a la vez, se resta el termino adecuado (0 o una versión con corrimiento adecuado para d) del resto parcial, que se inicializa del valor de dividendo z, y se procede hasta que todos los *k* bits del cociente q se hayan determinado, sin tomar en cuenta el residuo final es denominado s [1][2]. Este proceso se presenta como en la ecuación (1). Dos algoritmos clásicos secuenciales de división en base-2 son: la división con restauración y sin restauración [3][4]. Inclusive se han realizado comparativos tomando en cuenta los recursos computacionales entre los algoritmos más utilizados en base-2 desarrollados en FPGA como en [5] o algoritmos más eficientes [6][7].

Para la implementación en hardware o software, se utiliza la ecuación de recurrencia que describe el proceso anterior bit a bit.

$$s^{(j)} = 2s^{(j-1)} - q_{k-j}d2^k \text{ , con } \quad s^{(0)} = z \quad \text{y} \quad s^{(k)} = 2^k s \tag{1}$$

←Corrimiento →

← Sustraer ⟶

Hoy en día la inclusión de los arreglos de compuertas programables en campo (FPGA's) en el diseño y desarrollo de hardware difuso para acciones de control de procesos y toma de decisiones, permiten el procesamiento concurrente, reconfigurabilidad y la versatilidad en estas arquitecturas. Sin embargo este tipo de controladores incluyen dentro de su estructura algun bloque lógico que realice la operación de división, la cual limita el procesamiento concurrente del FPGA, pero sobre todo ralentiza el diseño debido a la operación secuencial bit a bit de este tipo de algoritmos y aumenta considerablemente tiempo de procesamiento y los recursos computacionales.

Los diseños basados en procesamiento paralelo, generalmente son preferidos debido a sus altas velocidades de procesamiento, por lo cual la propuesta de un algoritmo de división entera surge de la necesidad de incluir este bloque lógico, para que realice el cálculo aritmético en una arquitectura nueva de hardware difuso, debido a que los algoritmos tradicionales existentes no cumplían con las caracteristicas deseables tanto en el consumo de recursos computacionales como en tiempo de respuesta al diseño de hardware difuso en cuestión.

2. Metodología

La metodología para realizar el algoritmo de divisor propuesto basado en FPGA a 8-bits, se puede observar de manera general en la figura 2. En la cual se nuestra que el algoritmo se divide esencialmente en los siguiente bloques de proceso simples.

1. Inicializar:

   El dividendo z de 8-bits, el divisor d de 8-bits y la variable n de 8-bits

2. Generar 7 desplazamientos al mismo tiempo:

$r_7 = R(n+7) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_6 = R(n+6) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_5 = R(n+5) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_4 = R(n+4) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_3 = R(n+3) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_2 = R(n+2) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_1 = R(n+1) \leftarrow z(n);\ n = 0, \ldots, 7$
$r_0 = R(n) \leftarrow z(n);\ n = 0, \ldots, 7$

3. Comparar el dividendo z con todos los términos r_n obtenidos:

Si $z \geq r_7$ Entonces
$w_7 = 2^7$

De lo contrario
$w_7 = 0$

Entonces si
Si $z \geq r_6$ Entonces
$w_6 = 2^6$
De lo contrario
$w_6 = 0$

Si $z \geq r_5$ Entonces
$w_5 = 2^5$
De lo contrario
$w_5 = 0$

Si $z \geq r_4$ Entonces
$w_4 = 2^4$
De lo contrario
$w_4 = 0$

Si $z \geq r_3$ Entonces
$w_3 = 2^3$
De lo contrario
$w_3 = 0$

Si $z \geq r_2$ Entonces
$w_2 = 2^2$
De lo contrario
$w_2 = 0$

Si $z \geq r_1$ Entonces
$w_1 = 2^1$
De lo contrario
$w_1 = 0$

Si $z \geq r_0$ Entonces
$w_0 = 2^0$

De lo contrario
$w_0 = 0$
Fin

4. Sumar todos los pesos w_n para obtener el cociente q
$q = w_7 + w_6 + w_5 + w_4 + w_3 + w_2 + w_1 + w_0$

Fin.

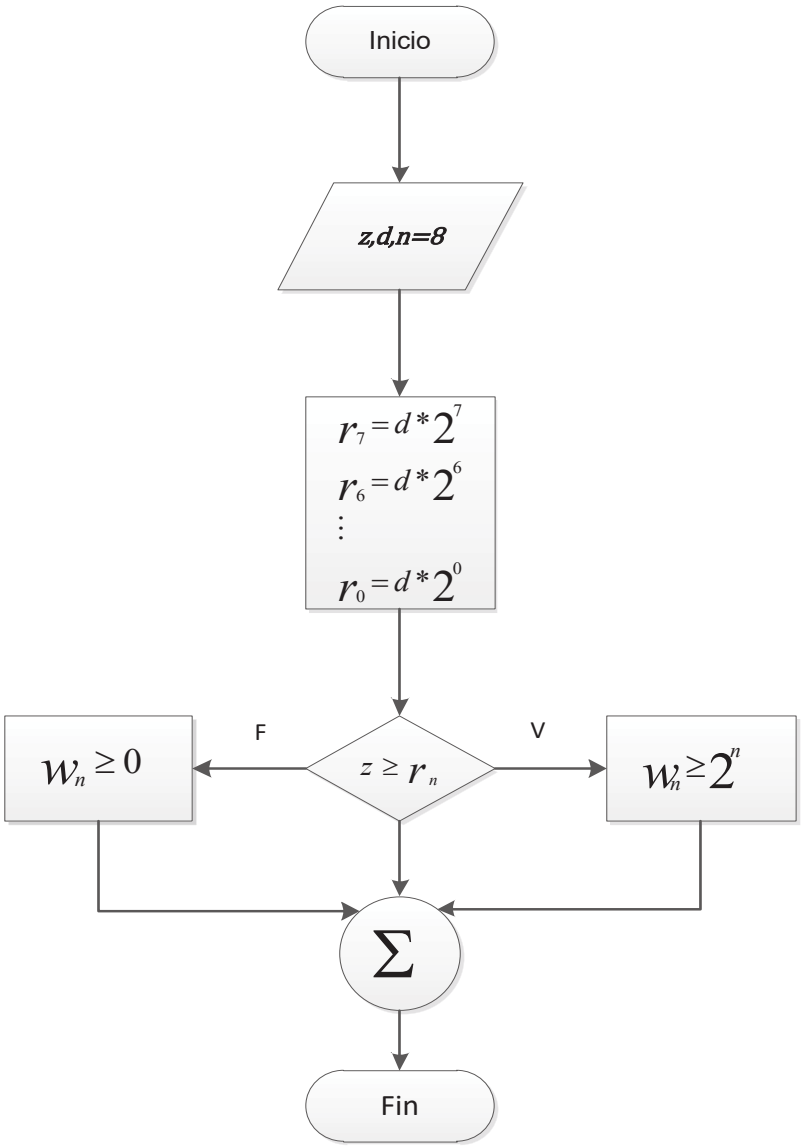


Figura 2. Algoritmo de división propuesto.

Ingeniería y Automatización, Capítulo 27, pp. 352 - 361.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Resultados

Como resultado de la investigación y del análisis de los algoritmos de división entera binaria, permitió diseñar y desarrollar un algoritmo paralelo de 8-bit adecuado para utilizarse en arquitecturas de hardware difuso que así se requieran. En la figura 3, se muestra la arquitectura funcional del divisor obtenido. Esencialmente consta de tres bloques: el primer bloque GNR_D realiza desplazamientos hacia la izquierda del dato d desde 0 a 2^{n-1}, el bloque CMP_U compara la entrada z con cada uno de los desplazamientos y activa en '1' sus salidas de G7 a G0 si la entrada es mayor o igual, y permanece en '0'si la entrada es menor, finalmente el bloque SUM realiza la sumatoria de un vector de pesos generados desde 2^0 hasta 2^{n-1}.

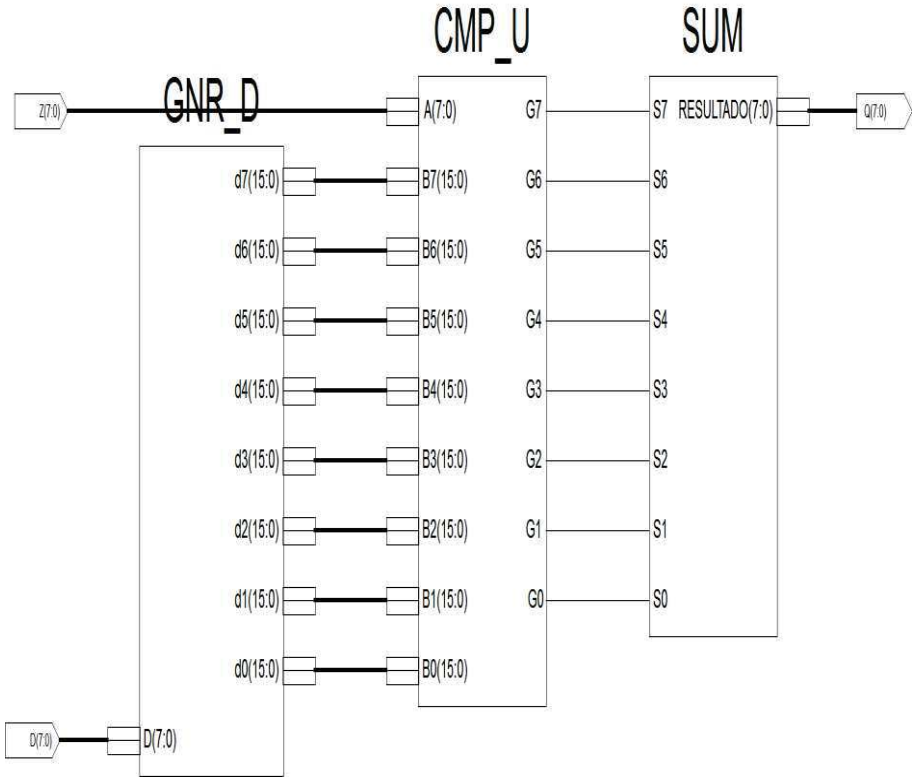


Figura 3. Arquitectura funcional de divisor propuesto.

3.1 Simulación del divisor a 8-bits.

El circuito fue simulado en el entorno de desarrollo ISE Design Suite 14.2 ingresando cuatro valores aleatorios de 8-bits para el dividendo z y el divisor d y obteniendo el valor en formato entero para el cociente q como se muestra en la Tabla 1. En la figura 4 se puede observar la simulación del algoritmo de división en donde se puede apreciar que el cálculo de la operación solo dura el tiempo de propagación a través del bloque.

Tabla 1. Calculo de división a 8-bits.

Dividendo z	Divisor d	Cociente q
247	5	49
132	16	8
199	43	4
77	10	7

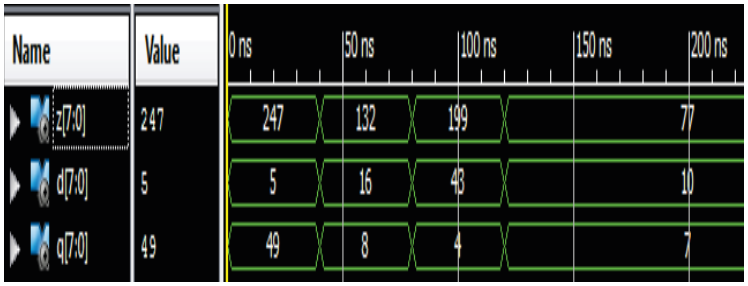


Figura 4. Simulación del algoritmo de división.

Finalmente en la tabla 2 se compara el algoritmo propuesto contra algunos divisores tanto en recursos computacionales como ciclos de reloj o en tiempo de respuesta.

Tabla 2. Comparativo entre algoritmos de división.

Implementacion Artix-7 FPGA	Número de Slice Registers	Número de Slice LUTs	Número de occupied Slices	Clock Cycles/tiempo
Division corrimiento-resta (DCR) [3]	296	159	64	4
Division Binaria Entera (DBE) [8]	41	85	52	8
Algoritmo de recurrencia [9]*	--	3595	--	154 ns
Algoritmo propuesto	26	97	45	70.018 ns

*Algoritmo de división a 16-bits.

3.2 Diseño y desarrollo del circuito a 16 bits.

Adicionalmente a la simulación el algoritmo fue escalado para trabajar con operandos de una mayor longitud 16-bits, para realizar la operación de división en bloque de hardware difuso propuesto[10], cuyo objetivo era reducir el número de ciclos de reloj 50 ciclos para realizar esta operación a una ejecución el paralelo. En la figura 5 se muestra parte de la arquitectura de hardware difuso en la cual el bloque de división propuesto (circulo azul) propuesto puede ser utilizado un tamaño de bit mayor.

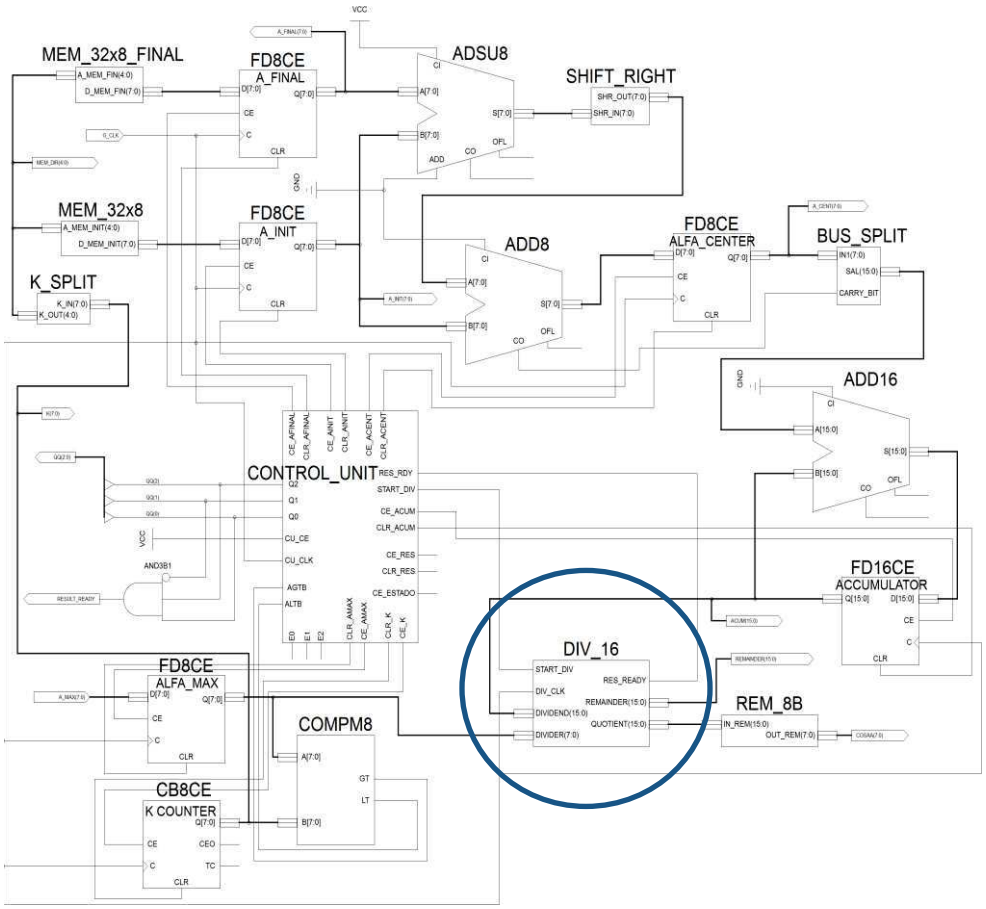


Figura 5. Arquitectura de Hardware difuso a mejorar

El circuito propuesto adaptado el cual reemplazara al circuito etiquetado como DIV_16 de la figura anterior utiliza solo 2 entradas las cuales están etiquetadas como DIV (divisor) y DIVD (dividendo) con un tamaño de 16 bits y la salida denominada RESULTADO con un tamaño de 8 bits. Esta arquitectura sigue la metodología del bloque de fuzzificación propuesto para pendientes racionales. El circuito de división a 16 bits se muestra en la figura 6.

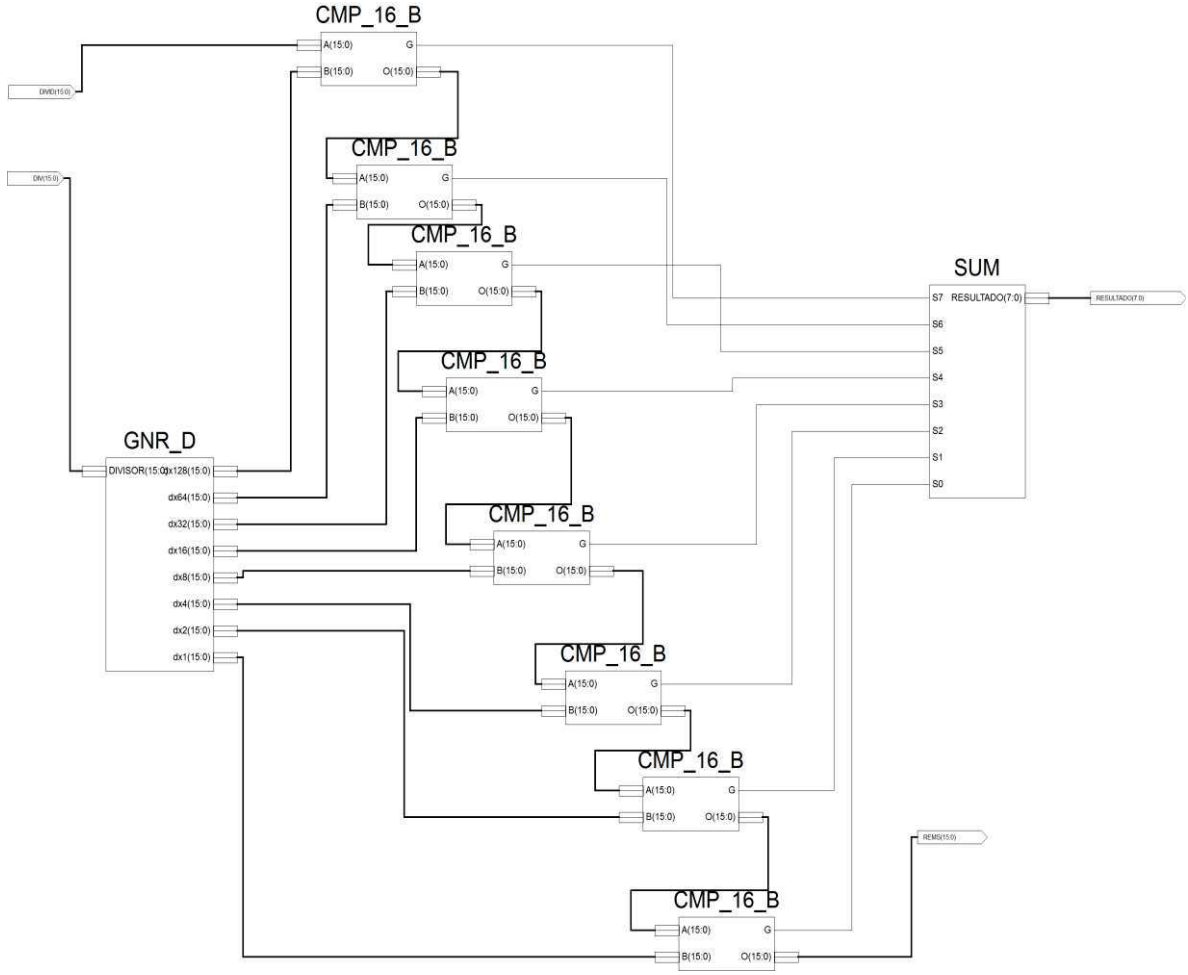


Figura 6. Arquitectura de fuzzificador propuesto para división a 16 bits.

3.3 Simulación de divisor a 16-bits propuesto para circuito COSSA

Para el caso del bloque de división para 16-bits para el hardware difuso se realiza la simulación de un bloque de división propuesto, el cual reemplazara al bloque actual. Para la simulación solo es necesario ingresar los parámetros de divisor div y el dividendo $divid$ para obtener el resultado del cociente como se presentan en la tabla 3. La figura 7 muestra la simulación para 5 operaciones de división, se puede notar que el tiempo de respuesta para concluir el cálculo es solo en tiempo que tarda en propagarse la señal a través del circuito lógico.

Tabla 2. Calculo de división a 16-bits.

Dividendo $divid$	Divisor div	resultado
3402	226	49
1034	7	147
432	9	47
129	2	64
2100	11	190

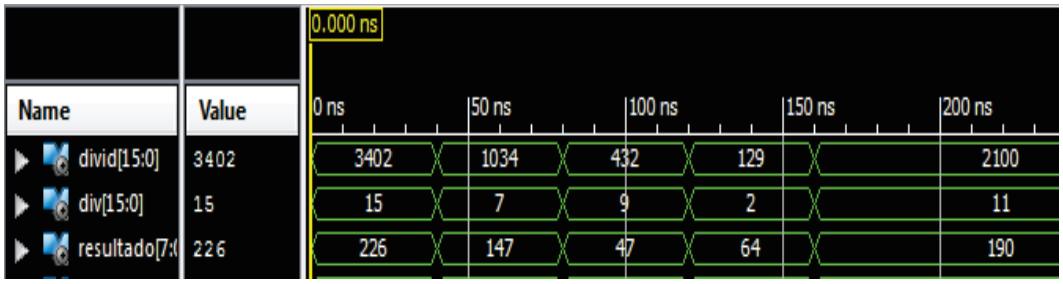


Figura 7. Simulación de divisor propuesto para circuito COSAA.

4. Conclusiones

El consumo de recursos del algoritmo de división propuesto fue comparado contra los divisores DCR y el DBE de [3] y [8] respectivamente. El algoritmo propuesto utiliza 91 % menos registros de slices que el DCR, y 36 % menos que el DBE. Para el consumo de Slice LUT, el algoritmo propuesto, incrementa el consumo in 13 % comparado con el DBE, y reduce su consumo en 53 % comparado con el circuito DCR. Referente al número de slices ocupadas, el algoritmo propuesto, reduce su número en un 20 % contra el DBE y 47 % en comparación con el DCR.

La característica más importante del circuito propuesto es que realiza la operación de división entera en paralelo obteniendo el resultado tan pronto como la señal se propaga a través del circuito en 70.018 ns.

En conclusión, este algoritmo diseñado y desarrollado para ser incluido en una arquitectura de hardware difuso basada en niveles-α mostro utilizar igual o menor cantidad de recursos computacionales que algoritmos secuenciales desarrollados previamente en base-2, pero su potencialidad radica en el tiempo de procesamiento el cual puede realizar el proceso de división como lo muestra la tabla 1.

Debido que este algoritmo fue diseñado en FPGA puede ser utilizado en algunas otras arquitecturas basadas en 8-bits que requieran este bloque lógico debido a su versatilidad. Por otra parte este algoritmo puede ser escalado a la utilización de operandos con una mayor cantidad de bits, es decir una aplicación de 16-bits como lo muestra la tabla 3.

Referencias

[1] P. Behrooz, *Arquitectura de Computadoras de los Microprocesadores a las Supercomputadoras*. Mexico: McGraw-Hill, 2007.
[2] J. L. H. David A. Patterson, *Computer Organization and Desig*, 3th ed. Unites States of America: Morgan Kaufmann, 2005.
[3] P. Behrooz, *Computer Arithmetic*. Nueva York: Oxford University Press, 1999.
[4] M. E. and T. Lang, *Digital Arithmetic*. Kaufhmann, Morgan, 2004.
[5] G. Sutter, J. Deschamps, G. Bioul, and E. Boemo, “Power Aware Dividers in FPGA,” pp. 574–584, 2004.
[6] A. E. Bashagha, “Area-efficient nonrestoring radix-2 k division,” vol. 15, pp. 367–381, 2005.
[7] K. Yen, *Design and realization of low power CMOS dividers*. 2012.
[8] K. C. Chang, *Digital Systmes Desig with VHDL ans Synthesis: An Integrated Approach*. U.S.A: IEEE Computer Society, 1999.
[9] M. Baesler, “2013 - Baesler - Analysis of fast radix-10 digit recurrence algorithms for fixed-point and floating-point dividers on FPGAs.” .
[10] H. A. and C. Oscar, “Centre of Slice Area Average Defuzzifier For Digital Fuzzy Systems and Its Hardware Implementation,” vol. 0, pp. 1–28, 2012.

Ingeniería y Automatización, Capítulo 28, pp. 362 - 371.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Prototipo de un Clasificador de Piezas Basado en un Brazo Robótico con Sistema de Transporte y Paletizado

Hernández Mosqueda Carlos, Pérez Chimal Rosa Janette, Luna Puente Rafael, Prieto Figueroa Araceli y Manjarrez Mendoza Javier

Universidad Tecnológica de Salamanca, Departamento de Mecatrónica, CP 36766, Salamanca, Gto. México.
chernandez@utsalamanca.edu.mx

Resumen

En la Universidad Tecnológica de Salamanca se realizan diversos esfuerzos para proveer a los estudiantes de habilidades prácticas en el ámbito industrial. Es por ello que este proyecto se enfoca específicamente en el diseño de un sistema de transporte de materiales y un sistema de paletizado para la inclusión en un proceso desarrollado por el Robot Mitsubishi RV-7F-D. Donde los estudiantes sean capases de realizar la programación de cada uno de los elementos empleando diversas secuencias, con la finalidad de practicar la programación usando el lenguaje MELFA-BASIC V, así como la manipulación del Robot Mitsubishi RV-7F-D para guardar las coordenadas mediante las cuales el robot accederá según su programación previa. Como resultado específico se obtienen el diseño en solidworks del sistema de transporte junto con la programación en lenguaje C bajo la plataforma Arduino y la etapa de potencia para la conexión con el robot. Respecto, al sistema de paletizado se obtiene el diseño en solidworks aunado con la programación en lenguaje C bajo la plataforma CCS compiller para su implementación con un microcontrolador PIC 17F877A y la etapa de potencia.

Palabras clave: Sistema de Transporte, Sistema de Paletizado, Robot RV-7F-D, *solidworks, Arduino, PIC 17F877A*

1. Introducción

La industria transnacional es altamente dinámica, la constante evolución de tecnologías y la ambición por optimizar los procesos productivos ha generado la necesidad de buscar soluciones que se adapten a los requerimientos actuales. Por esta razón, el poder contar con equipo para la práctica e implementación de proyectos de automatización da la oportunidad de ofrecer respuestas competitivas a esas necesidades.

En la Universidad Tecnológica de Salamanca se cuenta con un brazo robótico modelo RV-7F-D al cual se le ha acondicionado un sistema de banda transportadora (un elemento básico en la industria) y un sistema de paletizado. El sistema en conjunto puede ser utilizado de manera versátil para diferentes aplicaciones industriales. La ventaja de este sistema es que puede ser reprogramado en función a la necesidad del usuario ya que la banda transportadora puede ser reprogramada al igual que el sistema de paletizado y por su puesto del brazo. Lo anterior permite este ajuste a medida de la aplicación que se busque.

Los robots representan una opción rentable y ampliamente aceptada en toda la industria donde se requiere de una solución económica para aplicaciones “Pick and Place” rápida y altamente

precisas. Los sistemas de paletizado son altamente utilizados por su versatilidad para la paletización de productos pero se cuenta con producción de distintos formatos con pocas modificaciones.

El paletizado es actualmente utilizado en las industrias que necesitan paletizar diferentes productos o productos similares pro con distintas características en una sola línea de producción, el sistema propuesto es flexible y ofrece solución a estos procesos por ejemplo, de una línea de envasado de distintas medidas.

La metodología para poder desarrollar el presente proyecto consiste en analizar los tipos de sistemas de paletizado para comprender de esta manera como operan y poder adecuar un sistema para implementarlo al brazo.

Los sistemas de paletizado también son importantes en la rama industrial, paletizar está definido como el agrupamiento de elementos sobre una superficie (paleta, pallet, tarima etc), esto debido a que para muchas empresas es importante manipular el lugar donde se encuentran sus productos, a esta área se le denomina logística y la manipulación física de su mercancía es su principal función. Debido a la gran cantidad de mercancía que se maneja, muchas de las veces no es costeable mover del punto de origen del producto al punto final por lo que la mejor manera y más económica es agrupar. Por lo que el mover la mayor cantidad de elementos o componentes al mismo tiempo implica una reducción de costos.

Existen diversas clasificaciones de las bandas transportadoras, aplicadas a la industria, unas de ellas son por su tamaño, materiales utilizados y aplicaciones específicas. El funcionamiento de las bandas transportadoras está basado en una cinta la cual es arrastrada por fricción entre dos tambores, uno de los tambores es el que está conectado a un motor y el cual permite el control de velocidad y posición de la cinta. El otro tambor gira libremente y tiene como finalidad el de servir de retorno de la cinta. Algunos de los tipos de bandas son de tornillo, de suelo móvil y de rodillos, como ya se mencionó estos dependerán de la aplicación que se desee, por ejemplo para la industria alimenticia donde es necesario una banda “limpia”, las cuales son comúnmente utilizadas de aluminio, la cual permite la reducción de grasas y residuos.

Es conocido que los sistemas automatizados se describen como sistemas que ofrecen ahorro de tiempo y de reducción de riesgos, sin embargo, los costos iniciales son grandes por lo que el diseño del sistema de automatización no da pie a errores. El contar con un prototipo como el que se presenta, también permite la optimización del sistema y la realización de pruebas preliminares del diseño final.

2. Sistema de transporte y Sistema de paletizado

Los sistemas de transporte de materiales y paletizado son de gran importancia en la industria por lo que este artículo se centra en el diseño y control de los mismos.

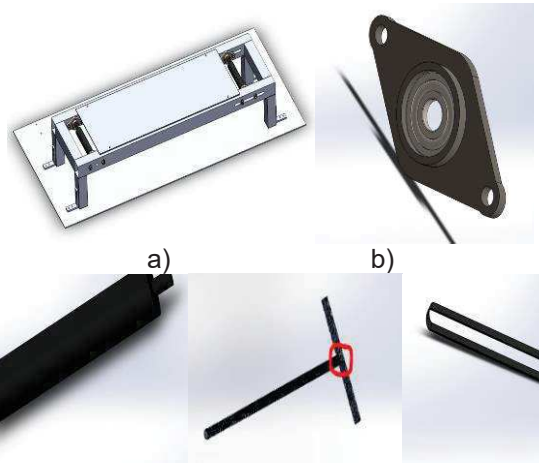
2.1 Sistema de transporte

La banda transportadora es uno de los sistemas de transporte más utilizado en la industria, permite transportar piezas y producto a granel, es arrastrada por fricción por uno de los dos tambores, que es accionado por un motor. El otro tambor gira libremente y tiene como función el de servir de retorno a la banda. Entre los dos tambores la banda (Fig. 1) es soportada por rodillos. [1]

Figura 1 Banda transportadora comercial

Como primer paso se realizó el diseño de la banda transportadora en el software de diseño Solidworks, para esto se realizaron las piezas que integran la banda transportadora (ver Figura 2).

.



a) b) c) d) e)

Figura 2 Partes que componen el sistema de transporte, a) base principal, b) chumaceras, c) rodillos, d) tensores, e) banda

Una vez concluidas cada una de las piezas se procedió a la realización del ensamble (Fig. 3) el cual consiste en la asignación de restricciones geométricas entre los elementos.

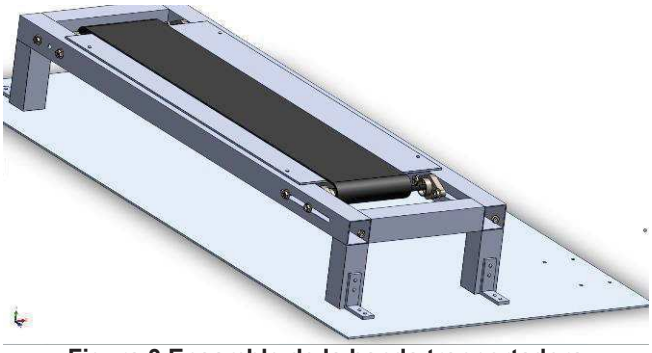


Figura 3 Ensamble de la banda tranportadora

La banda transportadora se montó en la una placa de manera que ésta realiza la función de soportar el peso de la banda mientras el robot Mitsubishi realiza sus movimientos, se empotra en la meza del brazo robótico, cabe mencionar que la banda transportadora debe estar ajustada a las medidas del robot para que estos puedan trabajar de manera conjunta sin percances o accidentes.

Se realizaron las conexiones del módulo de relevadores alimentando con 5V y su respectiva GND del mismo Arduino, se utilizó 1 relevador para el control del motor, los otros 3 sensores se usaron para enviar las señales de 24 volts al robot.

El ordenamiento de los sensores es muy importante ya que estos se encargan de enviar la señal al robot para tomar la pieza dependiendo el tamaño de ésta. Para este proceso los sensores son montados en una base y son activados enviando las señales a los relevadores en función del tamaño de la pieza.

La importancia de la banda transportadora recae en la velocidad de los motores los cuales van a hacer llegar a los sensores las piezas que se deseen ordenas, además de la sincronización con el robot el cual de no estar correctamente alineados y coordinados podría causar errores en todo el sistema.

2.2 Control del sistema

Para esta etapa se empleó una tarjeta Arduino que es una plataforma de electrónica de código abierto basada en hardware y software fácil de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlo en una salida, activar un motor, encender un LED y publicar algo en línea. Puede decirle a su tablero qué hacer enviando un conjunto de instrucciones al microcontrolador en el tablero. Para hacerlo, utiliza el lenguaje de programación Arduino (basado en el cableado) y el software Arduino (IDE), basado en el procesamiento. [2]

El entorno de desarrollo integrado Arduino - o el software Arduino (IDE) - contiene un editor de texto para escribir código, un área de mensajes, una consola de texto, una barra de herramientas con botones para funciones comunes y una serie de menús. Se conecta al hardware Arduino y Genuino para cargar programas y comunicarse con ellos [2].

Para la realización del programa (Fig. 4) se consideraron las siguientes condiciones:

- Debe ser un sistema que controle el movimiento de la banda hasta que la pieza este en posición de inspección.
- En la posición de inspección el sistema se deberá discriminar entre piezas de tres tamaños diferentes y enviar una señal de 24V que permita la activación al Robot RV-7F-D para realizar su secuencia.

```
void loop() {

//inicia la programacion del sensor infrarrojo
int valor8=digitalRead(8);
int valor9=digitalRead(9);
int valor10=digitalRead(10);

      while (digitalRead(10) == LOW){
       digitalWrite(13, 0);
       digitalWrite(3, 0);
```

Figura 4 Plataforma de programación Arduino

2.3 Sistema de paletizado

El paletizado es la acción de disponer mercancía sobre un área llamada pallet para su facilitar el almacenaje.

Para la realización de este sistema, se parte del diseño de las piezas en el programa SolidWorks algunas de las cuales se ilustran en la figura 5.

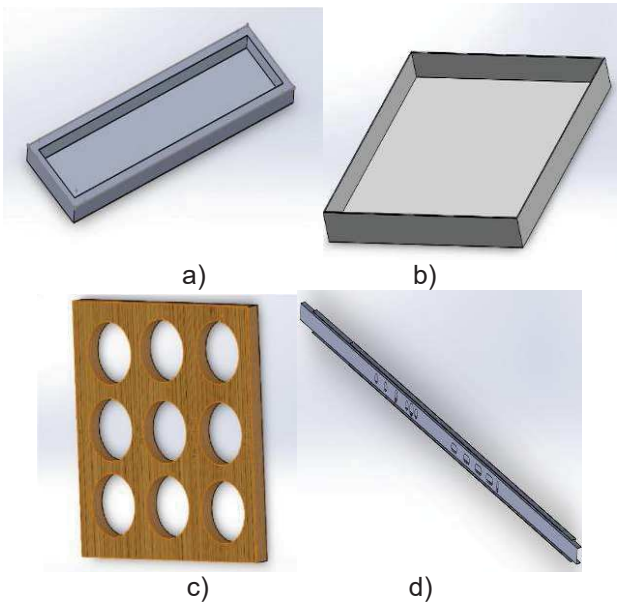


Figura 5 Piezas del sistema de paletizado. a) base del paletizado, b) bandeja de recepción de piezas, c) paleta de madera, d) corredera.

El sistema de paletizado (Fig 6) consiste en un mecanismo que permite el llenado y vaciado de tres pallets para la simulación de un sistema continuo de selección realizado por el Robot RV-7F-D.

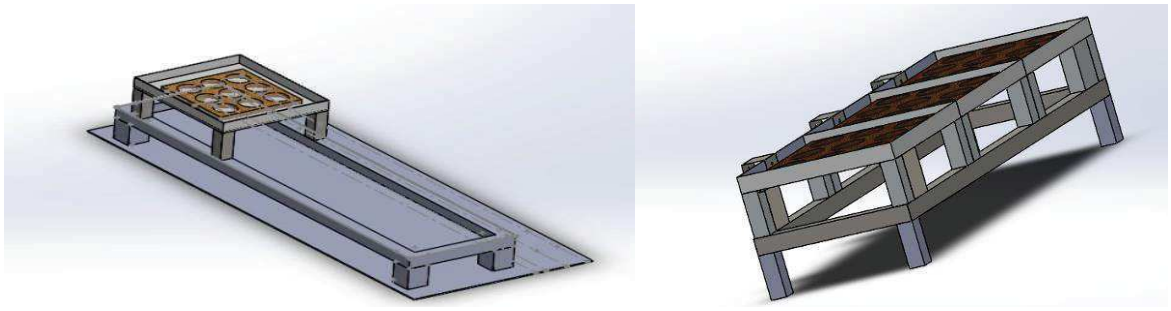


Figura 6 Diseño de sistema de paletizado. a) ensamble de primer modelo, b) ensamble del segundo modelo

Debido a la versatilidad que se le puede dar al prototipo, se diseñan dos sistemas de paletizado, el primero consiste en una paleta la cual se coloca sobre los soportes de la base, esta base del sistema se moverá de posición para acomodar las piezas de determinadas medidas.

El otro diseño busca lograr ser un soporte para la paleta o palé, en medio de esta bandeja se encuentra una trampilla en el cual una vez que esté listo la primera parte del paletizado, la trampilla tomara acción y dejara caer las fichas que se encuentran sobre ella, tomando en cuenta que la paleta

se quedara en la misma posición y sobre la bandeja. Para el soporte de la paleta se necesita de cuatro barras planas los cuales estarán soldados a los laterales en la parte baja para dar el soporte necesario a la paleta y no obstruya el paso de las piezas en la simulación.

Todos los elementos tanto de la corredora como de las paletas son diseñados y simulados en el programa de diseño para evitar la mayor cantidad de errores. En la parte inferior se le implementa el sistema mecánico para la paletización. Este sistema se activará cuando el sensor detecte que el robot cumple con su oficio, entonces se acciona la corredera para que esta pueda mover la trampilla de la bandeja y deje caer las fichas. Todo esto funcionando, gracias al motor tipo motorreductor y a los sensores finales de carrera, que son los encargados de dar la energía y la fuerza suficiente al sistema para que este sistema funcione correctamente.

2.4 Control del paletizado.

Para la realización del control se emplea el programa CCS Compiller [5] (Fig. 7) que corresponde al entorno de programación para microcontroladores Microchip también llamados PIC.
Para realizar la programación se consideraron los siguientes aspectos:

- Se debe controlar la apertura y cierre, para ello se cuenta con un sensor en cada pallet para indicar que se encuentran llenos.
- En el momento en que se llenan, se abren las trampas para liberar las piezas.
- Para el caso del pallet 2 solo se liberan las piezas hasta que el pallet 3 haya liberado sus piezas y antes que retorne a su posición.
- Se deberá enviar una señal de 24 V al Robot Mitsubishi RV-7F-D para indicar si es posible colocar más piezas o deberá esperar.

Una vez analizadas las consideraciones del programa se procedió a la realización (Fig.8)

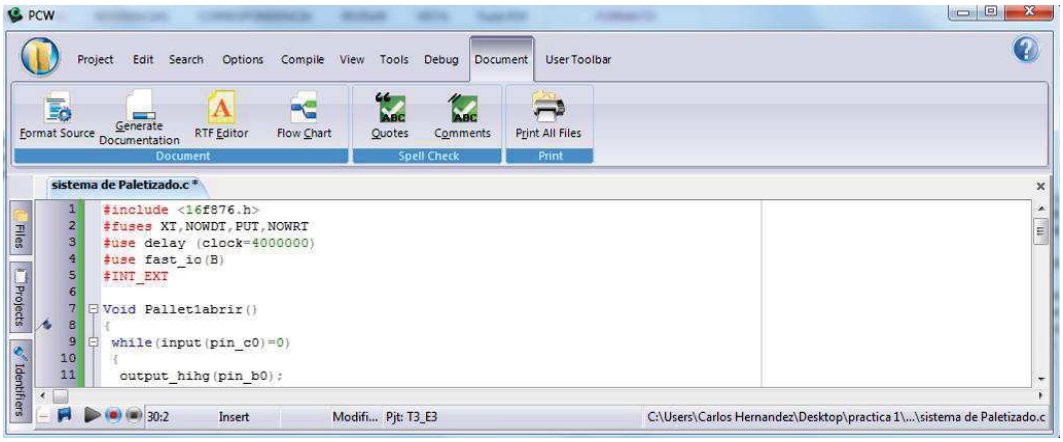


Figura 7 Plataforma de programación CCS COMPILLER

2.5 Integración con el Robot Mitsubishi RV-7F-D

Para la integración del sistema de transporte, el sistema de paletizado y el Robot Mitsubishi RV-7F-D se consideran tres etapas.

- Conexión física
- Programación de coordenadas del Robot Mitsubishi RV-7F-D
- Programación en MELFA Basic V

Conexión física: consiste en realizar la conexión entre la etapa de potencia del sistema de transporte y el sistema de paletizado, para esto se empleó un módulo de las entradas y salidas digitales la interface SysLink (fig. 8)

Figura 8 Interface SysLink

Para realizar la programación de coordenadas se accede a la interfaz de manipulación del Robot Mitsubishi RV-7F-D [6] llamada Teacing pendant (Fig. 9) en esta etapa se realizara la grabación de las coordenadas que accederá el robot para realizar la secuencia a programar.

Figura 9 Teaching pendant

Posteriormente se realizó la programación la cual debe considerar las siguientes características:

- Una vez que el sistema de transporte lleva las piezas a la posición de inspección envía una señal a la entrada M_IN(8) para piezas pequeñas, M_IN(9) para piezas medianas y M_IN(10) para piezas grandes.
- En función de la señal recibida el robot acomodara las piezas en el pallet correspondiente.
- Cuando la interfaz de M_IN(11), M_IN(12) y M_IN(13) son activadas es indicativo que los pallets de piezas pequeñas, medianas y grandes respectivamente se encuentran llenas.
- Una vez que esto ocurre el Robot espera a que se vacíen los pallets.
- Se repite la secuencia de manera indefinida.

Para la programación se empleó el software Cirus Programing (Fig. 10) de la empresa FESTO.

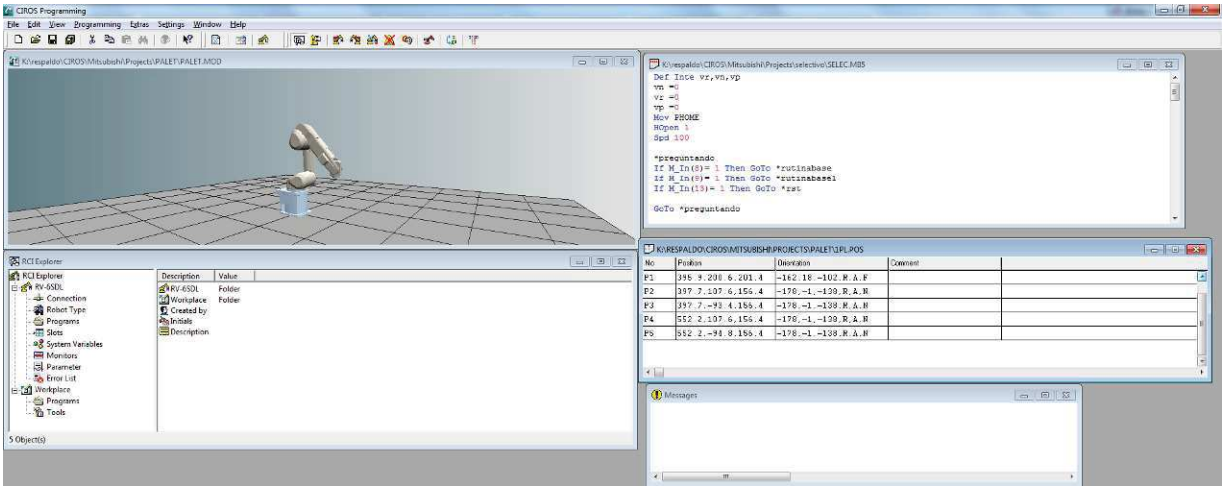


Figura 10 Cirus Programing

3. Resultados

La figura central del sistema automatizado reprogramable es el brazo robótico, anterior a este se encuentra la banda transportadora la cual es alineada al brazo, una vez alimentado el sistema de la banda transportadora se pueden colocar las piezas sobre la banda y esta comienza a girar, el motor es alimentado con 12V pero si se desea se puede colocar un motor de la potencia deseada para distintos trabajos.

El sistema continúa cuando uno de los sensores haya detectado una pieza y el tamaño del producto o material, estas piezas son colocadas de manera aleatoria para posteriormente ser ordenadas en las paletas, los señores envían una señal al Arduino activando uno de los relés del módulo de 4 líneas, la señal desactivara el motor temporalmente hasta que el robot recoja el material que está detectando los sensores, sí se precisa un cambio de banda basta con desatornillar los tonillos opresores de los rodillos, remover los 4 tornillos de la placa soporte de la banda y se podrá cambiar por una nueva.

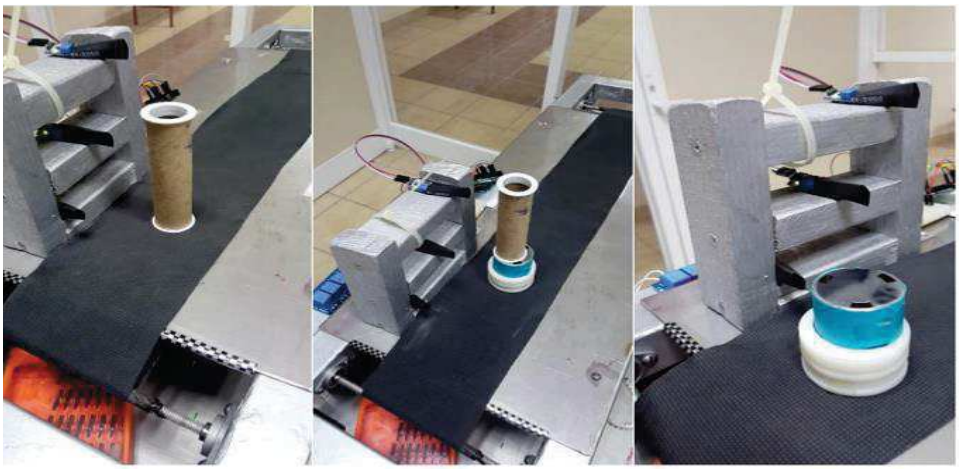


Figura 11 pruebas con objetos de distintos tamaños y formas

Una vez concluido este proceso, posterior al brazo se encuentra la estructura del paletizado (ver figura 12), cada paleta será para un determinado tamaño por lo que el robot deberá acomodarlas en una determinada paleta, esta misma paleta contendrá 6 agujeros, cuando la paleta se encuentre llena los sensores infrarrojos mandarán una alerta al brazo y a los motores; la alerta al robot será para

informarle que ya no llene esa paleta y se concentre en las otras dos, por su parte, la alerta al motor de la paleta le dará la orden de transportar la trampilla a un lado y al mismo tiempo que lo mueva, las fichas caerán y se vaciara la paleta, cuando finalmente ocurra esto la trampilla regresara a su posición original ayudado por el motor, esto ocurrirá con la otra paleta del extremo.

Figura 12 Sistema de paletizado

Cabe resaltar que la paleta de en medio no hará la función tal cual como las otras dos, ya que como la trampilla se mueve a los lados esta no se podrá mover hasta que la trampilla de al lado no se mueva, por ejemplo si en dado caso se llena primero la paleta de en medio esta tendrá que esperar hasta que la de al lado se llene primero y esta última se mueva de lugar, ahí será entonces cuando la de en medio tomará acción y se vaciará a los pocos segundos de que se halla vaciado la otra. Esto es todo el proceso a realizar en el sistema. El brazo funciona a través de un programa que se le implementó en sincronización con lavanda y el sistema de paletizado.

El sistema se detendrá cuando la banda transportadora no tenga más fichas para mover a su destino y el brazo robótico tenga ya informado que las tres paletas se llenaron con éxito, el sistema se volverá a iniciar cuando el usuario así lo quiera (ver figura 13).

Al realizar las pruebas con los sistemas de transporte, manipulación y el Robot Mitsubishi RV-7F-D se obtienen resultados favorables en la manipulación de las piezas hasta hacerlas llegar a la estación de inspección, una vez que las piezas se encontraron en esta estación fue necesario realizar diversos ajustes en las coordinas del Robot Mitsubishi RV-7F-D hasta lograr que las piezas se tomaran de forma adecuada para ser llevadas a la estación de paletizado, en esta estación se realizaron los ajustes necesarios para coordinar las trampas y asignarle las señales para que el robot continuar el proceso.

Figura 13 Prototipo de banda transportadora

4. Conclusiones

La adaptación de una banda trasportadora para proveer de objetos definidos al brazo robótico se implemento de manera eficaz, calibrada y orientada con gran precisión usando el *PIC 17F877A*. En lo que respecta a la programación del brazo robótico, se logró un exitoso posicionamiento de acuerdo a las coordenadas específicas de los objetos. Además, las maniobras realizadas por el mecanismo robótico se llevaron a cabo de manera precisa sin generar daño alguno. Finalmente, la clasificación de piezas mediante el uso de sensores infrarrojos se consiguió con el mínimo error, el cual se podría poner a prueba incrementando el número de piezas a clasificar.

Con la realización de este proyecto se logra la interconexión entre sistemas controlados por Arduino, PIC y el controlador del (Mitsubishi)

De esta forma se obtiene la sincronización de los tres sistemas con la finalidad que los estudiantes realicen prácticas y logren la programación y aprendizaje de:

- Programación en la plataforma Arduino
- Programación de microcontroladores en lenguaje C
- Manipulación del Robot Mitsubishi RV-7F-D
- Programación de coordenada
- Programación de secuencia del Robot

Como trabajo futuro se considera un sistema recolector de las piezas que las envíe nuevamente al sistema de transporte para lograr un sistema continuo.

Referencias

[1] QuimiNet.com. (23 de Noviembre de 2012). Obtenido de https://www.quiminet.com/articulos/las-bandas-transportadoras-y-su-usoen-la-industria-3346215.htm.
[2] Arduino. (s.f.). Obtenido de https://www.arduino.cc/en/Guide/Environment
[3] Fernando Reyes Cortés, J. C. (2015). *Arduino Aplicaciones En robótica, mecatrónica e ingenierías.* México: Alfaomega.
[4] Leonel G Corona Ramirez, G. S. (2015). *Sensores y actuadores aplicaciones con arduino.* Mexico D.F: Grupo Editorial Patria.
[5] Garcia, E. (2018). *Compilador C CCS y simulador Proteus .* ALFAOMEGA GRUPO EDITOR.
[6] Mitsubishi. (s.f.). Manual del Robot Mitsubishi RV-7F-D.
[7] Ojeda, L. T. (s.f.). arduino.cn. Obtenido de http://arduino.cl/que-es-arduino/
[8] QuimiNet.com. (23 de Noviembre de 2012). Obtenido de https://www.quiminet.com/articulos/las-bandas-transportadoras-y-su-usoen-la-industria-3346215.htm

Ingeniería y Automatización, Capítulo 29, pp. 372 - 382.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Rocker-Bogie del Espacio a la Tierra

Gutiérrez Castillo Rodrigo Uzziel y José Reyes Rosales

Instituto Tecnológico Superior de Atlixco, Calle Heliotropo, Unidad 8 Norte Nueva Xalpatlaco, Vista Hermosa, 74210 Atlixco, Pue.
rodriz_15@live.com.mx
jose.reyes@itsatlixco.edu.mx

Resumen

El proyecto descrito en este documento se refiere a la implementación de la robótica en situaciones de riesgo cuando un sismo ha dañado edificios y dentro de él se encuentran personas las cuales quedaron atrapadas. El robot cuenta con una suspensión del tipo Rocker-Bogie el cual es ampliamente usado en los rover enviados a realizar exploración en Marte por la NASA. En esta ocasión se implementa para poder ingresar a los inmuebles colapsados para buscar si hay alguien con vida y a la vez poder saber que tan dañado se encuentra el lugar de búsqueda y rescate.

Las pruebas realizadas al robot fueron bastante favorables ya que puede subir y bajar sin problema los escombros que se encuentran a su alrededor.

Palabras clave: Desastres naturales, Impresión 3D, Robot de ayuda y rescate, Robótica móvil, Rocker-Bogie, Rover.

1. Introducción

Hoy en día los robots están en todos lados, desde ensambladoras hasta el hogar. Actualmente hay un robot llamado Da Vinci el cual ayuda a los cirujanos a realizar su labor. También los vemos como mascotas mecánicas o en algunos casos como guías turísticos.

Sin embargo hay pocos de estos robots en ayuda y rescate. Un claro ejemplo es el desastre que causó el tsunami en la planta nuclear de Fukushima, Japón en 2011, para poder realizar varias operaciones e inspecciones en el área se tuvo que crear un robot móvil para poder ingresar ya que la radiación mataría a los humanos. [1]

Otro caso de emergencia donde se necesita la ayuda de los robots es cuando hay sismos ya que dañan los inmuebles y por ende es riesgoso entrar a asistir a las personas que se encuentran atrapadas. Un robot con cámaras puede ayudar a realizar un mapeo para que los rescatistas puedan planificar una ruta segura. Un ejemplo de esto fue el pasado 19 de Septiembre de 2017 cuando se presentó un sismo de magnitud 7.1 donde se calcula que 369 personas perdieron la vida, el colapso de sesenta edificios y al menos 8000 con daños de los cuales aproximadamente del 10% con daños graves [2], en este altercado acudieron personas a socorrer a las personas, sin embargo al haber réplicas las estructuras dañadas pueden colapsar y de esta manera perder más vidas humanas. Además al rescate también acuden perros los cuales ayudan a encontrar a personas y se sabe también que un robot llamado Xinamiqui Olinki acudió a asistir a varios rescatistas.

La robótica como se ha visto puede estar en situaciones de alto riesgo para los humanos. Sin embargo aún no hay suficientes robots que puedan asistir a tales eventos. Por esta razón se planeó

crear un robot móvil con suspensión Rocker-Bogie debido a su eficiencia y fácil operación a comparación de otros sistemas como los bípedos o hexápodos.

2. Suspensión Rocker-Bogie

El diseño de suspensión Rocker-Bogie es muy viable si se quiere crear un vehículo con gran estabilidad y capaz de escalar obstáculos. Gracias a estas cualidades del sistema se han implementado con gran éxito en los Mars Exploration Rover (MER). La estructura consiste en seis ruedas que hacen contacto con el suelo a pesar de que este se encuentre irregular, es decir cada rueda puede estar a una altura diferente de la otra sin perder estabilidad. Esto hace posible que un robot con ruedas pueda navegar en entornos de difícil acceso y con gran variedad de obstáculos, colinas, etc. Lo anterior lo puede apreciar en Fig. 1. [3]

Fig. 1 Robot con suspensión Rocker-Bogie en un terreno irregular. Obtenida de página web NASA

En un robot móvil se presentan dos suspensiones Rocker-Bogie, uno en la parte derecha del cuerpo y el otro a la izquierda. El sistema consiste en seis patas con llantas las cuales tienen un motor como actuador como se ve en la Fig. 1. El Bogie en cada extremo tiene dos patas que están unidas al Rocker mediante una junta, el Rocker sólo cuenta con una extremidad la cual está unida mediante una junta al cuerpo. Las partes que conforman el robot se muestran en la Fig. 2.

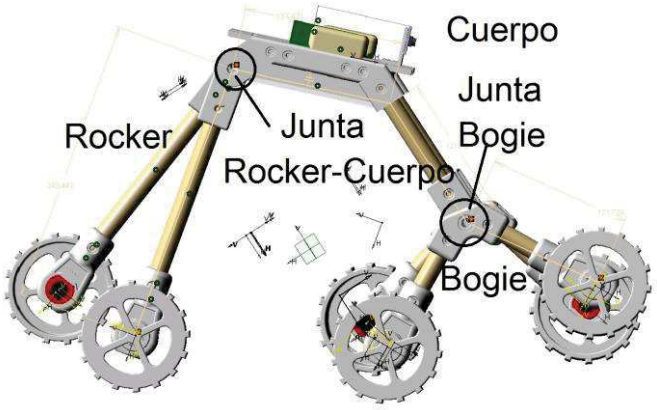


Fig. 2 Partes que conforman la suspensión Rocker-Bogie

El sistema de suspensión también cuenta con un diferencial que une a los Rocker para poder girar sin problemas.

3. Desastres naturales

Un desastre es cuando existe algún tipo de destrucción en un territorio el cual puede generar devastación del medio ambiente y pérdida de vidas humas. Un desastre natural se produce cuando la energía amenazante proviene de un fenómeno natural. Se dividen en tres tipos: [4]

- Meteorológicas: Producto de cambios atmosféricos o del clima.
  - Huracanes
  - Ciclones
  - Tifones
  - Tornados
  - Tormentas
  - Ondas térmicas
  - Sequías
- Topográficos y geotécnicos: Proveniente de la superficie terrestre.
  - Derrumbes
  - Avalanchas
  - Flujos
- Tectónicos o geológicos: Fuerzas internas de la tierra.
  - Terremotos
  - Erupciones volcánicas
  - Tsunamis

Existen registros verdaderamente alarmantes relacionados a los desastres naturales. En la Tabla 1 Se muestran varios de estos incidentes. [5]

Tabla 1. Desastres naturales en América Latina y el Caribe de 1972-1988.

Lugar	Año	TIPO	Víctimas
Managua	1972	Geológico	6000 muertos, 20000 heridos, derrumbe en casi todos los edificios públicos, comerciales
Centro América	1974	Meteorológico	7000 muertos, 15000 personas sin hogar
Guatemala	1976	Geológico	22800 muertos
República Dominicana	1979	Meteorológico	2100 muertos, 600000 quedaron sin hogar
Bolivia, Chile, Ecuador y Perú	1982-1983	Meteorológico	298000 muertos y heridos
Ciudad de México	1985	Geológico	Más de 10000 muertos, 30000 heridos o alteraciones síquicas, 33600 viviendas destruídas, 65000 con daños
Colombia	1985	Geológico	22800 muertos, 5200 heridos, 10000 personas quedaron sin hogar
San Salvador	1986	Geológico	1200 muertos, 10000 heridos, 500000 con daños en sus hogares o negocios
Ecuador	1987	Geológico	1000 muertos, 3000 viviendas destruídas, 12500 con daños
Nicaragua, Costa Rica, Panamá y El Salvador	1988	Meteorológico	310000 evacuados

Como se puede apreciar varios de los incidentes en el periodo de tiempo mostrado en la Tabla 1 varios son geológicos de los cuales bastantes edificaciones tienen daños de los que probablemente haya personas a las cuales se les puede socorrer. Sin embargo como las estructuras colapsadas no se pueden ingresar con facilidad hace que la búsqueda sea menos eficiente sin saber si aún hay cuerpos o no. En la Tabla 2 se muestran datos igual de alarmantes relacionados a las catástrofes producidas por fenómenos naturales pero en esta ocasión de Chile país del G20 que ha tenido la mayor pérdida económica en relación a su PIB (Producto Interno Bruto) de 1980-2011 el cual es mayor del 1.26% derivado de desastres naturales. Cabe mencionar que México ha tenido una pérdida poco mayor del 0.2% del PIB. [6]

Tabla 2. Desastres naturales en Chile 1900-2015.

Tipo	No. Incidentes	Pérdidas humanas	Heridos
Geológico	31	59543	76556
Geológico	8	110	
Geotécnico	5	407	186
Metereológico	29	1052	58

Los datos obtenidos de la tabla anterior fueron proporcionados por la EM-DAT (The International Disaster Database).

Sin olvidar la situación de México del 19 de Septiembre de 1985 y 2017 donde los sismos afectaron al centro del país. El primero ya mencionado en la Tabla 1 arrojó bastantes muertos en esa ocasión y el que se presentó en 2017 dejando menos víctimas pero sí bastantes edificios colapsados donde varias personas aún seguían con vida. Sin embargo no se pudo inspeccionar en su totalidad todas las zonas afectadas donde quizás todavía se encontraban sobrevivientes. Sin embargo en desastres naturales del tipo geológicos (específicamente terremotos) han sido varias las ocasiones que se han presentado en el país. En las Fig. 3 se aprecian los sismos más fuertes desde el año 2000 hasta el 2013 y en la Fig. 4 se observa un mapa de la República Mexicana con los sismos ocurridos en el 2017.

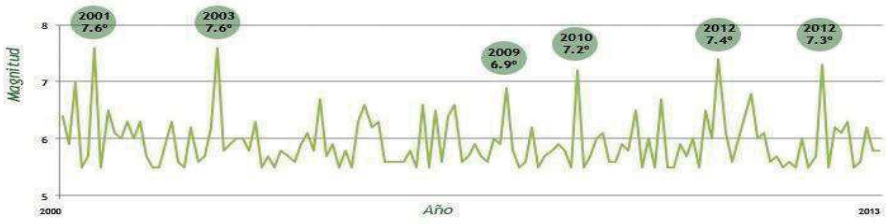


Fig. 3 Sismos más fuertes en México del año 2000 al 2013.
Obtenida de Estadísticas sobre Temblores en México.

Ingeniería y Automatización, Capítulo 29, pp. 372 - 382.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

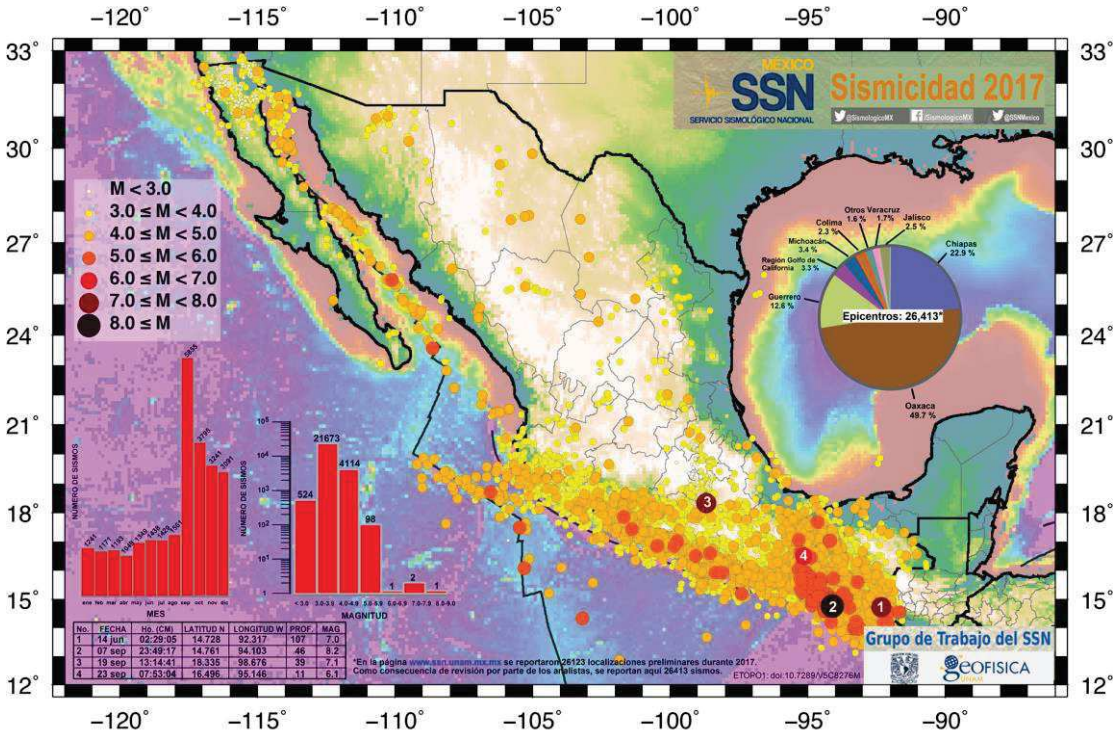


Fig. 4 Sismos presentes en México en 2017.
Obtenida de Servicio Sismológico Nacional.

Es por esta razón que se deben de implementar innovaciones tanto para prevenir como para actuar con rapidez. Afortunadamente ya se ha estado analizando oportunidades de innovación y emprendimiento. Hay instituciones que se están dedicando al desarrollo de productos y/o servicios tecnológicos para aminorar los daños ante una catástrofe como Understanding Risk, grupo internacional conformado por más de 6500 expertos y técnicos cuyas aportaciones tecnológicas han sido el uso de redes sociales, aplicación SMS Lapli, el uso de Drones, estaciones meteorológicas móviles, etc. [6]

En México también existen instituciones relevantes en temas de desastres naturales como lo es el FONDEN (Fondo de Desastres Naturales) que ayuda a prevenir algún altercado y reconstrucción. [7] Sin embargo en temas tecnológicos estamos algo atrasados como lo explicó el Dr. José Luis Morán López en la Cuarta presentación del Programa Cátedras CONACYT. Mencionó que se debe apostar por el desarrollo tecnológico y de la robótica para que en un futuro los robots rescatistas estén presentes en estos eventos. [8]

4. Estado del arte

La tecnología ha asistido al ser humano desde tiempos ancestrales, a brindado una mejor calidad de vida y un sin fin de bienes que hacen que hoy en día la vida sea más fácil. A pesar de las invenciones que se realizan día a día aún no se han logrado desarrollos tecnológicos capaces de erradicar los fenómenos naturales ya que es algo sumamente imposible, sin embargo sí se han creado sistemas que nos ayudan a prevenir los desastres naturales. Un ejemplo es en nuestro México que se han implementado innovaciones contra los terremotos como las citadas a continuación: [9]

- Escuchar los sonidos de la tierra: En Noviembre del 2017 Investigadores de la UNAM y de Japón colocaron sensores sobre la tierra y debajo del mar en la brecha de Guerrero para hacer el monitoreo de la actividad sísmica.
- Lo que dice la estructura de los edificios: Acelerómetros instalados en puntos estratégicos en los edificios para poder determinar la condición estructural después de algún altercado.
- Mantener la comunicación en la emergencia: Existe un protocolo de comunicación llamado Tetrapol inmune a interrupciones la cual está en proceso de ser digitalizada para ser operada.

También hay robots que han sido diseñados para ayuda y rescate en el mundo. Los tipos de robots de rescate se pueden clasificar según el medio al que se adentran y por su tamaño. Puede apreciar la Tabla 3. [10]

Tabla 3. Clasificación de los robots de rescate.

Ambiente		**Tamaño**	
Terrestre	UGV	Man-Package	Una persona lo puede llevar
Aéreos	UAV	Man-Portable	Vehículo ligero
Marinos	UUV USV	Maxi	Remolques especiales

Definiciones:

- UGV: Pequeños vehículos no tripulados (Unmanned Ground Vehicle)
- UAV: Vehículos Aéreos ni tripulados (Unmanned Aerial Vehicle)
- UUV: Vehículos no tripulados bajo el agua
- USV: Vehículos no tripulados sobre el agua

Varios de los prototipos diseñados y construidos para acceder a lugares de difícil acceso debido a algún derrumbe provocado por algún movimiento telúrico son presentados a continuación:

Robot hexápodo presentado en [10]. El autor de este robot aplica un sistema llamado C-Leg el cual consiste en patas en forma de C. Cada extremidad del hexápodo cuenta con una de ellas como se ve en la Fig. 5.

Fig. 5 Robot Hexápodo con extremidades del tipo C-LEG.

El mismo autor creó otra versión de robot hexápodo con extremidades del tipo C-LEG de manera más liviana [11]. Sin embargo sólo realizó el diseño mecánico más nunca lo manufacturó. El hexápodo se presenta en Fig. 6.

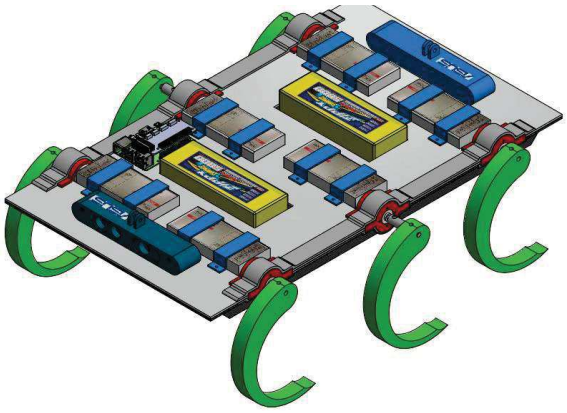


Fig. 6 Diseño mecánico de un robot Hexápodo liviano con mecanismos del tipo C-LEG.

El diseño propuesto por [12] y mostrado en Fig. 7 también es para búsqueda y rescate en estructuras colapsadas. El sistema robótico está conformado por seis extremidades colocadas en forma de rombo, además cuenta con varias articulaciones que ayudan a subir y bajar obstáculos. De la misma manera como el diseño anterior no se realizó su manufactura para comprobar su funcionamiento.

Fig. 7 Diseño mecánico de un robot móvil con varias articulaciones para la asistencia en zonas de desastre.

El robot presentado por [13] es una plataforma robótica móvil para búsqueda posterior a un sismo y a la vez apagar incendios de baja magnitud. El robot cuenta con sistema de amortiguamiento, su movilidad es del tipo oruga, cuenta con un brazo robotizado, sistema para apagar incendios menores y además transmisión de vídeo en tiempo real. Las pruebas de este robot han sido bastante buenas logrando su cometido e incluso el poder cargar un peso extra de aproximadamente 65Kg como se muestra en Fig. 8.

Fig. 8 Robot del tipo oruga con brazo mecánico.

Los robots de los autores anteriores son propuestas muy interesantes, sin embargo los dos primeros al ser robots hexápodos sin articulaciones hacen que su funcionamiento en un entorno donde hay escombros por todos lados no sea eficiente y posiblemente se queden atorados. El tercer robot mostrado se ve muy robusto, posiblemente pueda atravesar sin problemas los obstáculos que se le presenten sin embargo al no haberlo probado en un entorno real, es difícil saber si puede acudir a zonas dañadas a causa de algún fenómeno natural. Por último tenemos el robot oruga con un brazo mecánico el cuál ha sido comprobada su utilidad con fines de búsqueda y rescate e incluso ayudar a extinguir pequeños incendios pero el costo se ve muy elevado al contar con bastantes mecanismos y al ser realistas el brazo robot no puede ayudar a levantar escombros muy pesados o incluso pasar por lugares muy pequeños haciendo un poco deficiente el sistema. Por esta razón se ha desarrollado un robot de menor tamaño para poder entrar en lugares pequeños, es fabricado con manufactura aditiva o mejor conocida como impresión 3D. Las piezas pueden ser reemplazadas si alguna sufre de algún daño, además cuenta con vídeo-transmisión en tiempo real y es operado a distancia, también cuenta con un sistema de suspensión Rocker-Bogie lo cual hace que navegue con facilidad en terrenos irregulares y como se mostró en el capítulo 2 ha sido implementado en robots enviados a Marte. Dicho robot se muestra en la Fig. 9.

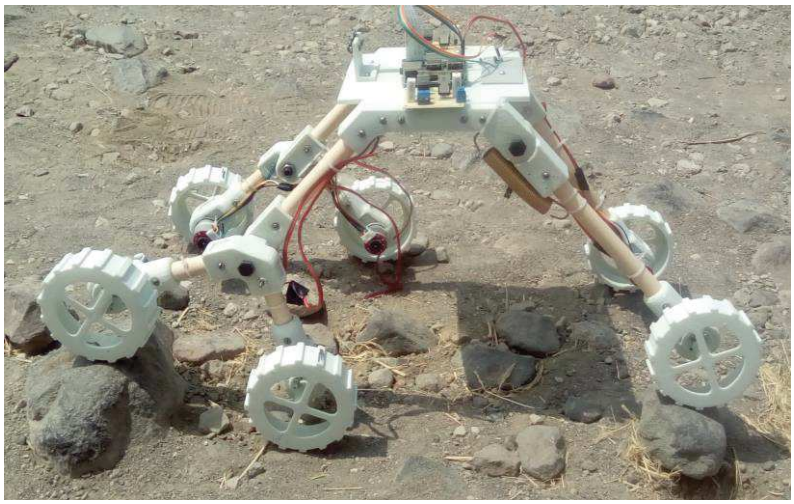


Fig. 9 Robot con suspensión Rocker-Bogie con piezas impresas en 3D.

5. Resultados

Como resultados del diseño de un robot móvil con suspensión del tipo Rocker-Bogie y vídeo transmisión en tiempo real manufacturado con impresión 3D, tenemos un prototipo eficiente, económico que no rebasa el precio de los $8000.00, con dimensiones de 286.5mm de altura, largo entre las llantas de los Bogie de 350mm y base de 577.6mm para que pueda entrar en lugares de difícil acceso.

Las evaluaciones del robot fueron realizadas con obstáculos simulando los escombros de una situación en la que un sismo sacudió el inmueble y en terrenos irregulares. La primera prueba arrojó que el robot con una eficiencia del 30% en los seis actuadores que tiene puede navegar en línea recta sin ningún tipo de obstáculos, pero cuando se trata de escalar obstáculos, este requiere de mayor energía aplicada en sus motores, mientras más inclinada sea un objeto al que se tiene que pasar, mayor será la energía suministrada en los motores. Estas pruebas se realizaron con tablas las cuales tenían un ángulo de inclinación diferente entre sí como se muestra en Fig. 10. Al realizar la prueba se comprobó que el robot no tenía ningún problema de estabilidad ya que todas las llantas siempre se encontraban tocando un punto fijo.

Fig. 10 Obstáculo improvisado con tablas formando Ángulos diferentes para las pruebas del robot.

La segunda prueba que se realizó en terreno irregular se comprobó que al sistema de suspensión Rocker-Bogie se le facilita navegar saltando piedras en un extremo del robot y del otro en un terreno plano. También se realizaron prueban en un pequeño montículo de tierra improvisada dónde tenía diferentes alturas mostrada en Fig. 11 y Fig. 12. No olvidemos que además el robot cuenta con vídeo-transmisión en tiempo real para ir monitoreando su trayecto y en aplicaciones de búsqueda y rescate se pueda evaluar el inmueble dañado y verificar sí hay o no personas con o sin vida.

Fig. 11 Montículo de tierra improvisada

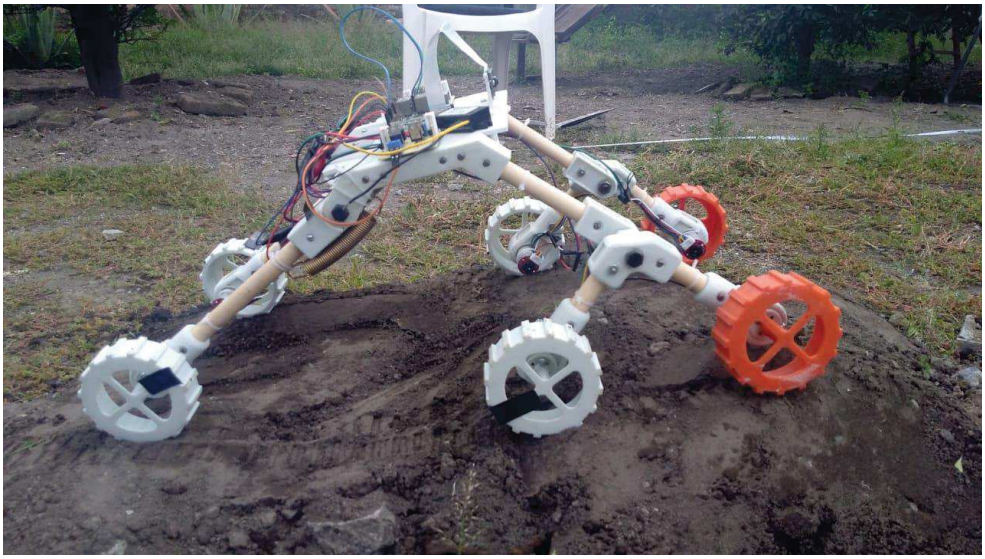


Fig. 12 Robot postrado en obstáculo improvisado con tierra

6. Conclusiones

En conclusión los avances tecnológicos y los robots están ayudando a la solución de problemas causados por algún fenómeno natural. Sin embargo aún hay bastante trabajo que realizar tanto como de prevención como de asistencia rápida a las personas que se encuentran atrapadas en los escombros de algún inmueble colapsado, es por esta razón que se tiene que innovar a como del lugar. El robot presentado en este artículo aún tiene mucho que mejorar como por ejemplo contar con una diferencial para poder dar giros y sea más fácil su movilidad.

Referencias

[1] Nagatani K., Kiribayashi S., Okada Y., Otake K., Yoshida K., Tadokoro S., Nishimura T., Yoshida T., Koyanagi E., Fukushima M., Kawatsuma S., "*Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots*", Journal of Field Robotics. J. Field Robotics, Vol. 30, No. 1, p. 63, 2012.

[2] Zúñiga F. "*Los sismos del 8 y 19 de septiembre de 2017 en México*", Enseñanza de las Ciencias de la Tierra. AEPECT, 25.3, p. 256, 2017.

[3] Harrington B., Voorhees C. "*The Challenges of Designing the Rocker-Bogie Suspension for the Mars Exploration Rover*", 37th Aerospace Mechanisms Symposium, Jet Propulsion Lab., California Inst. of Tech.; Pasadena, CA, United States, p. 185-195; NASA/CP-2004.

[4] Vargas J. "*Políticas públicas para la reducción de la vulnerabilidad frente a los desastres naturales y socio-naturales*". CEPAL ECLAC, Chile, Impreso en Naciones Unidas, 2002.

[5] Jovel J. "Revista de la CEPAL", Naciones Unidas Comisión económica para América Latina y el Caribe. No. 38, p. 161, 1989.

[6] Gobierno de Chile. "*CONSULTORÍA SOBRE DIMENSIONAMIENTO DEL MERCADO DE DESASTRES NATURALES: IMPACTO Y TAMAÑO EN CHILE Y EL MUNDO*". Idom, Versión 2.0, p. 154, 2017.

[7] Sin autor. "*El fondo de desastres naturales de México-Una reseña*". FONDEN, p. 12, 2012.

[8] Sánchez V. "*Los científicos ante el sismo*". Agencia Informativa Conacyt, p. 2, 2017.

[9] Peralta L. "*La tecnología en México contra los terremotos*", Obras Web. 2018.

[10] De León J., Barrientos A. "*Definición y análisis de los modos de marcha de un robot hexápodo para tareas de búsqueda y rescate*", INDUSTRIALES UPN, p. 137, 2015.

[11] De León J., Barrientos A. "*Diseño y simulación del Sistema de Locomoción de un Robot Hexápodo para Tareas de Búsqueda y Rescate*", Industriales UPN, p. 113, 2016.

[12] Cortes A., Rojas C. "*DISEÑO DE UN VEHÍCULO CONTROLADO REMOTAMENTE PARA APOYO EN LABORES DE BÚSQUEDA Y RESCATE EN ESTRUCTURAS COLAPSADAS*", ESCUELA TECNOLÓGICA INSTITUTO TÉCNICO CENTRAL INGENIERÍA EN DISEÑO DE MÁQUINAS Y PRODUCTOS INDUSTRIALES BOGOTÁ. BREC, VCR1.0, p. 178, 2014.

[13] Sánchez L. "*DESARROLLO DE UNA PLATAFORMA ROBÓTICA MÓVIL PARA BÚSQUEDA POSTERIOR A UN SISMO, Y MITIGACIÓN DE INCENDIOS DE BAJA ESCALA*", ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO, p. 112, 2017.

Ingeniería y Automatización, Capítulo 30, pp. 383 - 392.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Optimización de la Detección de Altitud/Orientación Utilizando un Sensor Monocular para la Coordinación de dos Drones

Arias Méndez Nelson Edgardo [(1)], Jonatán Muñoz Álvarez [(2)], Escareño Castro Juan Antonio [(2)], Palomino Merino Amparo Dora [(1)] y María Aurora Diozcora Vargas Treviño [(1)]

Benemérita Universidad Autónoma de Puebla [(1)].
Insititut Polytechnique des Sciences Avanceés [(2)]
arias.200910906@gmail.com

Resumen

*Este trabajo tiene como objetivo mejorar la detección de un sensor de visión (cámara) que es utilizado para coordinar dos drones, uno tiene la función de ser esclavo y el otro actúa como maestro. Para ello se implementarán filtros digitales en la detección de la orientación del vehículo mediante la perspectiva visual, se hará uso de la librería **OpenCv** en el entorno **Python**. Los filtros ayudarán a tener mejor posicionamiento del vehículo esclavo. Finalmente se analizarán las respuestas del sensor con los filtros y sin ellos, para finalmente hacer un consenso de vuelo de ambos drones y verificar sus respuestas graficando la posición angular (Roll, pitch y yaw) y la posición en el espacio del mundo real (x, y, z), donde **z** representa la distancia entre ambos drones, **y** la altitud y **x** la orientación lateral.*

Palabras clave: Robótica, visión.

1. Introducción

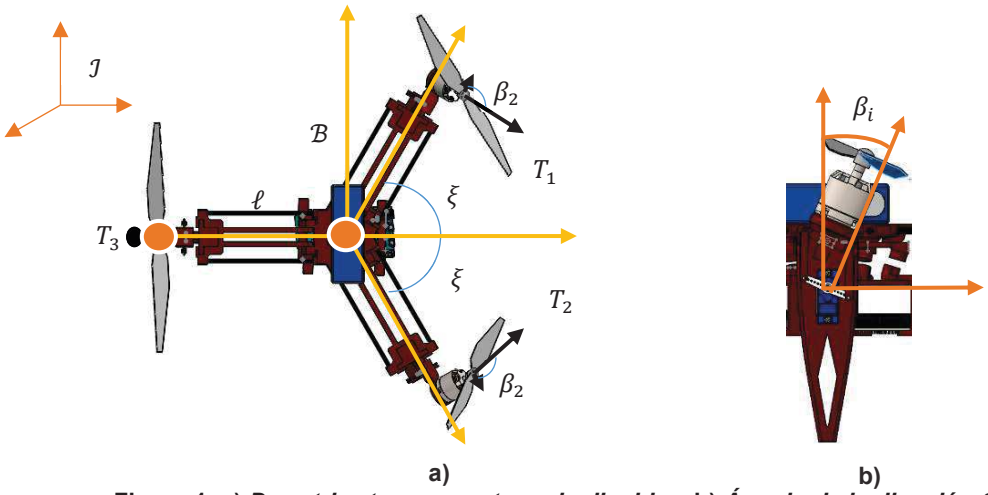
El progreso de los sistemas robóticos aéreos es más que remarcable involucrándose en diferentes aplicaciones. Por esta razón las configuraciones y las estrategias de control, evolucionan constantemente para hacer frente a los desafíos impuestos en las nuevas tareas. Los VANT's (Vehículos Aéreos No Tripulados) varían en talla y número de rotores, lo que permite tener un perfil adecuado de vehículo para una tarea específica. En los últimos años este perfil operacional ha evolucionado para abordar aplicaciones de manipulación aérea (aplicaciones de logística), enjambres de drones (aplicaciones de redes de sensores), solo por mencionar algunas. Con el fin de asegurar un buen desempeño de estas nuevas aplicaciones, se requieren de estrategias eficientes de control con las que se pueda realizar la tarea deseada.

Es por ello que en este trabajo se buscará realizar la manipulación de dos drones: un dron tri-rotor (maestro) y un dron cuadri-rotor (esclavo). Se puntualizarán sus modelos matemáticos de forma concisa en la sección 2 y 3. En la sección 4 se verá el modelo matemático de la cámara (sensor monocular) que detectará las coordenadas en el espacio y en el plano de la imagen, se optimizará la detección mediante el uso de filtros digitales y se observará su comportamiento. En la sección 5 se analizará la estrategia de control aplicada en cada dron, para finalmente implementarla y examinar su respuesta en la sección 6.

Ingeniería y Automatización, Capítulo 30, pp. 383 - 392.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

2. Modelo dinámico de un dron tri-rotor en configuración no holonómica.

Como se comentó con anterioridad, el dron tri-rotor tiene un movimiento de inclinación especial (Figura 1 a) y 1 b) que conlleva a tener un modelo matemático distinto al convencional en aeronaves (Ecuaciones (1) y (2) [5][6].



a) b)

Figura 1. a) ***Dron tri-rotor con motores inclinables,*** **b)** ***Ángulo de inclinación*** $\boldsymbol{\beta_i}$**,** $\boldsymbol{i = 1,2}$

$$I\left[\mathcal{W}_{\mathrm{n}}\ \ddot{\eta} + \frac{\partial W_n}{\partial \phi} + \frac{\partial W_n}{\partial \theta}\dot{\theta}\right] + \mathcal{W}_{\mathrm{n}}\dot{\eta} \times I\mathcal{W}_{\mathrm{n}}\dot{\eta} = \boldsymbol{\Gamma^B} \tag{1}$$

$$m\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} = R(0,0,\psi)\left(\boldsymbol{T_1^B} + \boldsymbol{T_2^B} + \boldsymbol{T_3^B}\right) - m\begin{bmatrix}0\\0\\g\end{bmatrix} \tag{2}$$

La ecuación 1 representa el sistema de rotación mientras que la ecuación 2 el sistema de traslación. Donde $\mathcal{W}_{\mathrm{n}}$ es la matriz que relaciona los ángulos de *Euler* en el cuerpo $\mathcal{B}$ hacia el sistema de referencia inercial $\mathcal{I}$, I es la matriz de inercias (que no es diagonal debido a que el modelo no es simétrico), η es el vector de posición angular $(\phi, \theta, \psi)^T$ y T_i^B son las fuerzas ejercidas por cada actuador. Como se observa en el sistema de traslación, la matriz de rotación $\mathcal{R}(*)$ solo tiene componentes en ψ para que el sistema se encuentre en configuración no holonómica. El diagrama del dron tri-rotor se muestra en la figura 1 a).

El vector $\boldsymbol{\Gamma^B}$ representa el producto cruz ($\boldsymbol{l} \times \boldsymbol{T_i^B}$) siendo $\boldsymbol{l}$ la distancia desde el centro de masa hasta cada uno de los tres extremos en donde se encuentran los motores, y $\boldsymbol{T_i}$ representa las componentes de fuerza (Ecuaciones 3,4 y 5).

$$\boldsymbol{T_1^B} = \begin{bmatrix} C_{(\xi)} & S_{(\xi)} & 0 \\ -S_{(\xi)} & C_{(\xi)} & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & C_{(\beta_1+\beta_2)} & -S_{(\beta_1+\beta_2)} \\ 0 & S_{(\beta_1+\beta_2)} & C_{(\beta_1+\beta_2)} \end{bmatrix}\begin{bmatrix}0\\0\\T_1\end{bmatrix} \tag{3}$$

$$\boldsymbol{T_2^B} = \begin{bmatrix} C_{(\xi)} & -S_{(\xi)} & 0 \\ S_{(\xi)} & C_{(\xi)} & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & C_{(\beta_1-\beta_2)} & -S_{(\beta_1-\beta_2)} \\ 0 & S_{(\beta_1-\beta_2)} & C_{(\beta_1-\beta_2)} \end{bmatrix}\begin{bmatrix}0\\0\\T_2\end{bmatrix} \tag{4}$$

$$\boldsymbol{T_3^B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_{(\beta_1)} & -S_{(\beta_1)} \\ 0 & S_{(\beta_1)} & C_{(\beta_1)} \end{bmatrix}\begin{bmatrix}0\\0\\T_3\end{bmatrix} \tag{5}$$

En las ecs. (3), (4) y (5) $C_{(*)}$ y $S_{(*)}$ denotan el coseno y seno del ángulo respectivamente. En cuanto al ángulo de inclinación β_1 está presente en los tres rotores y tiene la misma magnitud y sentido, este

define el movimiento en ***yaw*** mientras que β_2 es el ángulo que controla el desplazamiento en el eje **x** definido en la figura 1 a).

3. Modelo dinámico de un dron cuadri-rotor

El dron esclavo es un dron de cuatro hélices y en él se encuentra colocada la cámara con la cual se realizará el seguimiento. Las ecuaciones matemáticas de este vehículo se han estudiado ampliamente en la literatura, el método más usual es el de *Newton-Euler.* En el conjunto de ecs. (6) se presenta el modelo simplificado [6].

$$\begin{aligned} \dot{\xi} &= \boldsymbol{v} \\ m\dot{\boldsymbol{v}} &= R_{\mathcal{I}}\boldsymbol{F_{\mathcal{B}}} \\ \dot{R}_{\mathcal{I}} &= R_{\mathcal{I}}S(\omega) \\ J\dot{\omega} &= -\omega \times \boldsymbol{J}\omega + \boldsymbol{\tau_b} \end{aligned} \tag{6}$$

donde $\boldsymbol{v}$ representa la velocidad en el plano (x, y, z), $R_{\mathcal{I}}$ la matriz de rotación en el plano inercial, $\boldsymbol{F_b}$ y $\boldsymbol{\tau_b}$ son las fuerzas y pares externos aplicados al cuerpo respectivamente (ecs. 7) y finalmente $R_{\mathcal{I}}S(\omega)$ determinan la orientación del dron.

$$\tau_b = \begin{bmatrix} (f_1 + f_2) - (f_3 + f_4) \\ (f_1 + f_3) - (f_2 + f_4) \\ \sum_{i=1}^{4} \mathcal{T} M_i \end{bmatrix} = \begin{bmatrix} \ell b(\Omega_1^2 + \Omega_2^2) - (\Omega_3^2 + \Omega_4^2) \\ \ell b(\Omega_1^2 + \Omega_3^2) - (\Omega_2^2 + \Omega_4^2) \\ k_t((\Omega_1^2 + \Omega_2^2) - (\Omega_3^2 + \Omega_4^2)) \end{bmatrix} \tag{7}$$

En la ecs. (7) ℓ es la distancia desde centro de masa hacia cada extremo, $\Omega_{i=1,\dots,4}$ representa la velocidad de cada motor con respecto a su eje, b es el coeficiente de empuje aplicado a cada motor y k_t es una constante que relaciona el esfuerzo de torsión con el cuadrado de la velocidad de cada motor. En la figura 2 se muestra al dron esclavo.

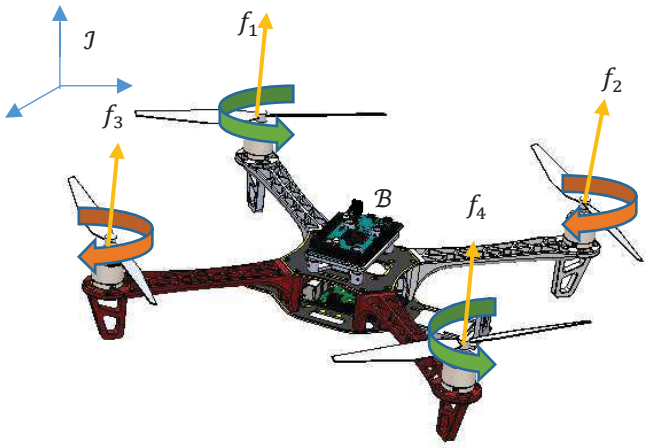


Figura 2. Dron cuadri-rotor

4. Sensor Monocular

Mediante una cámara empleada como sensor de posición se consigue la posición angular y orientación en el espacio del dron maestro (tri-rotor). La ec. (8) muestra el modelo ***pinhole*** y la ec. (9) su forma reducida***;*** este modelo sirve para formar una vista de escena proyectando los puntos 3D en el plano de la imagen utilizando una transformación de perspectiva [7][8].

$$s\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}f_x & 0 & C_x\\0 & f_y & C_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_1\\r_{21} & r_{22} & r_{23} & t_2\\r_{31} & r_{32} & r_{33} & t_3\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}$$

(8)

$$sm' = AR[t]M'$$

(9)

donde:

X, Y, Z : Son las coordenadas de un punto 3D en el espacio de coordenadas del mundo.
u, v : Son las coordenadas de la proyección del punto en pixeles.
A : Es la matriz de parámetros intrínsecos de la cámara.
(C_x, C_y): Es un punto principal que generalmente esta al centro de la imagen.
(f_x, f_y) : Son las distancias focales expresadas en unidades de pixeles.
$R[t]$: Es la matriz de parámetros extrínsecos
m' : Es el vector de coordenadas en el plano de la imagen
M' : Es el vector de coordenadas en el mundo real.

La detección del dron maestro se efectúa mediante un objeto cuadrado negro y de contorno blanco, de $18cm$ por lado utilizando una cámara con resolución de 256×256 pixeles, con la que se detectan sus vértices (figura 3 a)). El objeto será colocado en la parte trasera del dron tri-rotor, de modo que mantenga su ángulo de inclinación de avance y retroceso constante (debido a la configuración no holonómica); esto hará que no afecte a la medición de distancia y posición angular del dron esclavo.
El procesamiento de la información obtenida de la cámara se hace en el entorno ***Python*** utilizando la librería ***OpenCV*** mediante la función ***cv2.findContours()***.

Para obtener la distancia $\mathcal{D}$ (la cual representa la distancia entre el dron maestro y el dron esclavo) se caracteriza una función que determina la norma euclidiana $|\mathcal{N}|$ a partir del centro del objeto en la imagen (u_c, v_c), hacia uno de sus vértices con respecto a la distancia real medida (Figura 3 a)). Esta forma es válida si los ángulos son constantes o su variación es despreciable. Se debe considerar que, para el avance o retroceso del dron esclavo, éste debe modificar su ángulo de inclinación el cual se restringe a 5 grados para evitar grandes perturbaciones en la medición de la distancia.

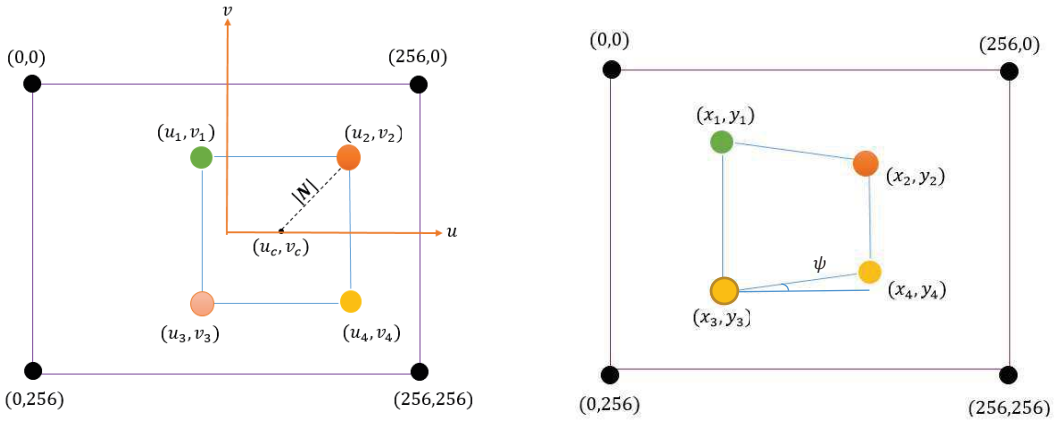


Figura 3. a) ***Proyección de la figura detectada en la cámara,*** **b)** ***Angulo de inclinación*** ψ

En la ec. (10) se puede observar que $\mathcal{D}$ es una función de $|\mathcal{N}|$.

$$\mathcal{D} = 0.026|\mathcal{N}|^2 - 4.2078|\mathcal{N}| + 227.88 \quad (10)$$

Para el posicionamiento angular solo se utilizará el ángulo ψ (figura 3 b)), ya que es el único que se necesita para el seguimiento en configuración no holonómica. Este ángulo se adquiere mediante el modelo de la cámara despejando la matriz $R[t]$, por lo que se utiliza la función ***cv2.SolvePnP,*** que retorna el vector de traslación y de rotación en el espacio del mundo real (el vector de traslación es despreciado ya que no es tan preciso en comparación con la obtención de la distancia a partir de la norma).
A continuación se implementa la función ***cv2.Rodriguez*** que convierte el vector de rotación en la matriz de rotación R (sin el vector de traslación), utilizando las ecs. (11) - (13), para transformar los coeficientes de la matriz a los ángulos de *Euler* (donde el subíndice v expresa el ángulo obtenido mediante la perspectiva de la cámara).
.

$$\psi_v = atan2\left(-r_{31}, \sqrt{r_{32}^2 + r_{33}^2}\right) \quad (11)$$

$$\phi_v = atan2(r_{21}, r_{11}) \quad (12)$$

$$\theta_v = atan2(r_{32}, r_{33}) \quad (13)$$

5. Filtrado de detección

Las señales del ángulo ψ_v y la distancia $\mathcal{D}$ presentan ruido conforme el objeto detectado se aleja, por lo que se utiliza un filtro complementario y un filtro pasa-bajas respectivamente para cada variable, mostrados en las ecs. (14) y (15).

$$\mathcal{D}_f = e^{\lambda h} D_{f-1} + \left(1 - e^{\lambda h}\right)\mathcal{D} \quad (14)$$

$$\psi_{vf} = K_1 \psi_{vf-1} + K_2 \psi_v \quad (15)$$

Donde D_f representa la distancia filtrada, ψ_{vf} el ángulo *yaw* filtrado, $\mathcal{D}_{f-1}$ y ψ_{vf-1} las variables filtradas en un muestreo anterior, λ es la constante que denota la frecuencia de corte del filtro, h el periodo de muestreo, K_1 y K_2 son las constantes que determinan el amortiguamiento de la señal del filtro complementario.

Se debe utilizar una constante λ adecuada para el periodo de muestreo de la cámara que es de 0.025 ms., la cual puede determinarse experimentalmente u obteniendo la frecuencia del ruido para determinar de manera exacta la frecuencia de corte.

6. Control y seguimiento

Para el control del avance y retroceso del dron maestro se eligió utilizar el controlador PD como se muestra en las ecs. ((16) - (21).

$$\mathcal{U}_{\phi 1} = \mathcal{K}p_{\phi 1} e_{\phi 1} - \mathcal{K}d_{\phi 1}\dot{\phi}1 \quad (16)$$

$$\mathcal{U}_{\theta 1} = \mathcal{K}p_{\theta 1} e_{\theta 1} - \mathcal{K}d_{\theta 1}\dot{\theta}1 \quad (17)$$

$$\mathcal{U}_{\psi 1} = \mathcal{K}p_{\psi 1} e_{\psi 1} - \mathcal{K}d_{\psi 1}\dot{\psi}1 \quad (18)$$

$$e_{\phi 1} = \phi_{d1} - \phi_1 \quad (19)$$

$$e_{\theta 1} = \theta_{d1} - \theta_1 \quad (20)$$

$$e_{\psi 1} = \psi_{d1} - \psi_1 \tag{21}$$

En las ecs. (16) y (17) $\theta_{d1} = 0$ y $\phi_{d1} = 0$ en todo instante de tiempo debido a la configuración no holonómica. La variable e_* denota el error de posición angular (ecs. (19) - (21)). Únicamente el ángulo ψ_{d1} es modificado y equivalente a $\beta_1 = \mathcal{U}_{\psi 1}$, mientras que el desplazamiento en x (figura 1 a)) está en función del ángulo de inclinación β_2; la suma de ambos ángulos se mantiene saturada en 30 grados. Las ecuaciones (22)- (24) son las variables de posición angular hacia los servomecanismos S_* que manipulan la inclinación

$$S_1 = \beta_1 + \beta_2 \tag{22}$$

$$S_2 = \beta_1 - \beta_2 \tag{23}$$

$$S_3 = \beta_1 \tag{24}$$

No existe una ecuación de control para el desplazamiento para β_2 ya que se manipula manualmente por medio de un control remoto.

En cuanto al control de seguimiento del dron esclavo se tienen las ecs. (25) - (36)

$$\mathcal{U}_{\phi 2} = \mathcal{K}p_{\phi 2} e_{\phi 2} - \mathcal{K}d_{\phi 2}\dot{\phi}2 \tag{25}$$

$$\mathcal{U}_{\theta 2} = \mathcal{K}p_{\theta 2} e_{\theta 2} - \mathcal{K}d_{\theta 2}\dot{\theta}2 \tag{26}$$

$$\mathcal{U}_{\psi 2} = \mathcal{K}p_{\psi 2} e_{\psi 2} - \mathcal{K}d_{\psi 2}\dot{\psi}2 \tag{27}$$

$$e_{\phi 2} = \phi_{d2} - \phi_2 \tag{28}$$

$$e_{\theta 2} = \theta_{d2} - \theta_2 \tag{29}$$

$$e_{\psi 2} = 0 - \psi_v \tag{30}$$

En esta caso ϕ_{d2} y θ_{d2} ecs. (31) y (32) llevan de forma implícita una ecuación de control, ya que la inclinación en ambos ángulos puede generar un avance/retroceso o un movimiento hacia los lados; mientras que el movimiento que controla la elevación está dado por el control en el ***trust (empuje)*** que es el mismo para todos los motores (ec. 33).

$$\phi_{d2} = K_{pD} e_D - K_{dD}\dot{D} \tag{31}$$

$$\theta_{d2} = K_{pu} e_u - K_{du}\dot{u} \tag{32}$$

$$T = K_{pv} e_v - K_{dv}\dot{v} \tag{33}$$

$$e_D = D_d - D \tag{34}$$

$$e_v = v_d - v_c \tag{35}$$

$$e_u = u_d - u_c \tag{36}$$

Siendo D_d la distancia deseada igual a 75 cm, mientras que u_d es la posición en pixeles $u_d = 127$ (que es el centro en el plano de la imagen) y u_c es la coordenada horizontal del centro de la figura

detectada. De igual manera v_d es la posición $v_d = 127$ y v_c la coordenada vertical del centro de la figura.

7. Resultados Experimentales

Las variables de posición angular de ambos drones son medidas con dos unidades de medida inercial, utilizando una tarjeta ***Arduino Due;*** las señales de los acelerómetros y giroscopios son combinadas mediante un filtro complementario y ambas conforman la posición angular filtrada. Las señales del procesamiento visual son enviadas mediante el puerto serie de una tarjeta ***Raspberry Pi 3*** (la cual realiza el procesamiento de imágenes) hacia la tarjeta ***Arduino***. Todas las señales son enviadas hacia la PC de forma inalámbrica mediante de un transceptor RF ***Xbee Pro S1,*** mientras que la conmutación de los motores se realiza mediante los ***ESC (Electronic Speed Controller)***. El diagrama de bloques del sistema maestro/esclavo se muestra en la figura 4.

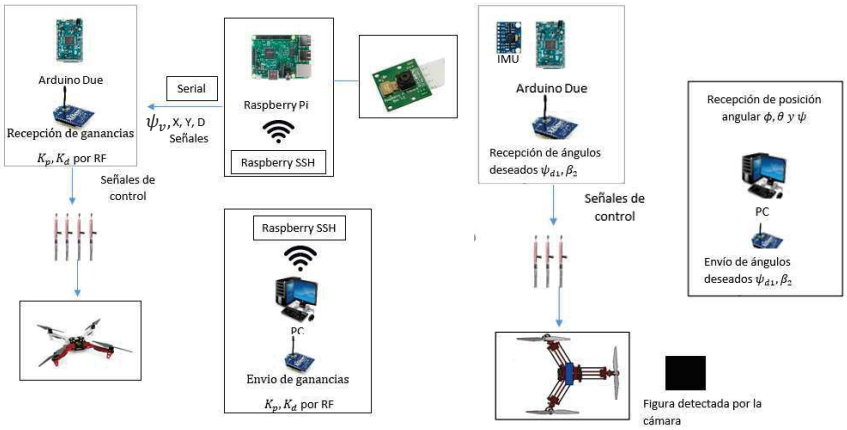


Figura 4. Diagrama de bloques del sistema maestro/esclavo

Ambos drones se inicializan separados uno del otro a una distancia aproximada de 1m, mientras que la cámara se posiciona en dirección hacia la figura detectada. El dron maestro debe elevarse junto con el esclavo colocando en el aire la figura, los dos reciben la misma señal de empuje (***trust***) hacia los motores para realizar el movimiento de elevación. El dron tri-rotor es manipulado manualmente por un control remoto, intentando mantenerlo en equilibrio, al mismo tiempo el dron cuadri-rotor se posiciona de forma automática en el centro de la figura detectada en el maestro (Figura 7). Como recomendación siempre es bueno tener buena iluminación en el entorno para tener una medición correcta, por lo que se utilizaron leds para iluminar el objeto detectado.

Figura 7 Plataforma experimental.

Ingeniería y Automatización, Capítulo 30, pp. 383 - 392.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

El posicionamiento angular en *roll* para ambos drones se muestra en la figura 8 a), mientras que para *pitch* y *yaw* en las figuras 8 b) y 8 c) respectivamente.

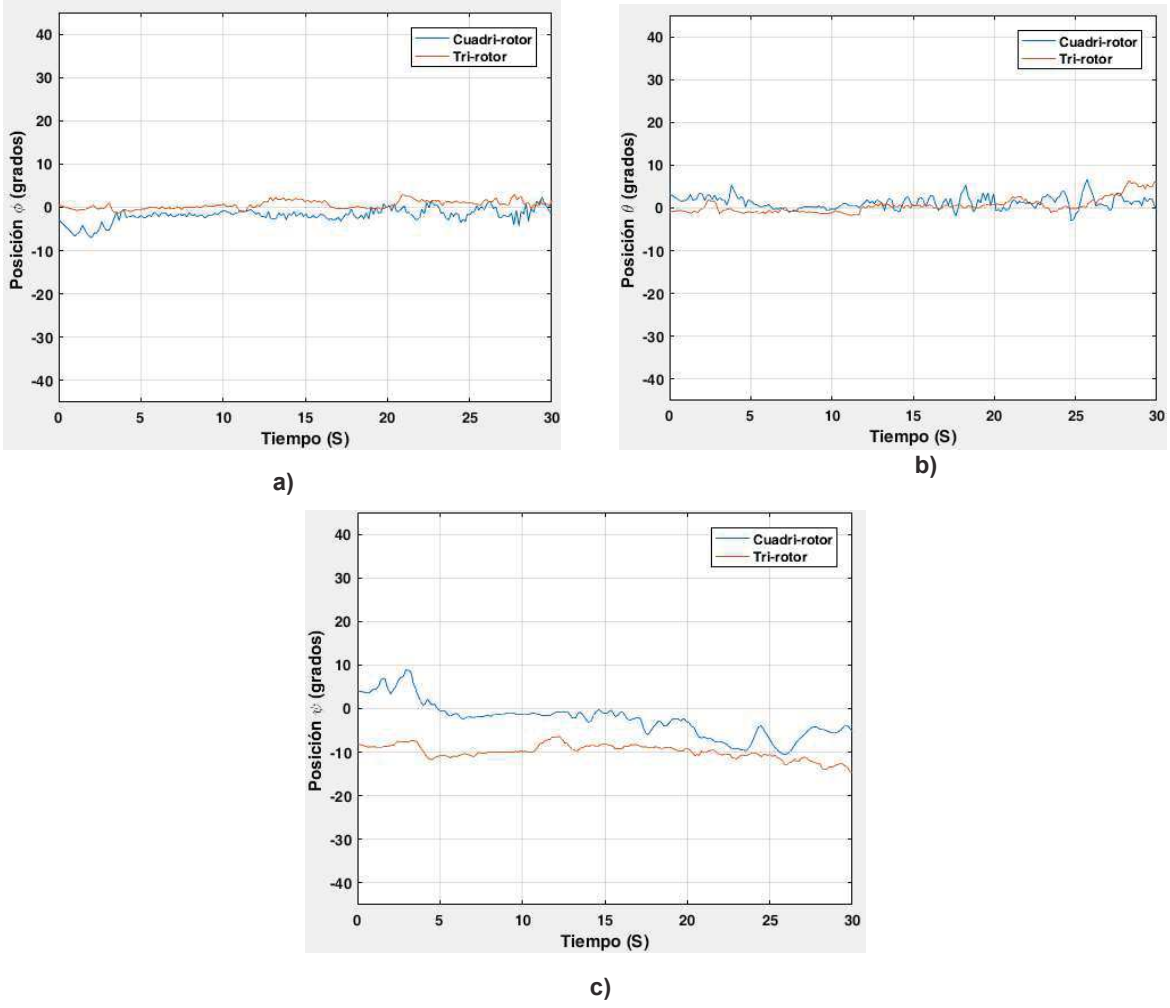


Figura 8 Posicionamiento angular a) *roll, b) pitch c) yaw*.

La respuesta del control de la distancia $\mathcal{D}$ del dron esclavo se observa en la figura 9, mientras que para el posicionamiento en pixeles x e y en las figuras 10 a) y 10 b).

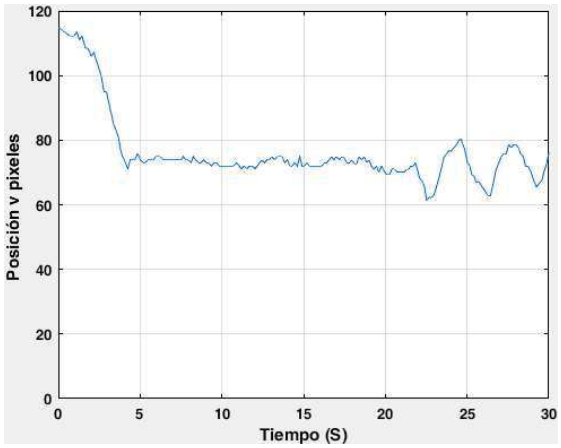


Figura 9 Posicionamiento en distancia entre ambos drones (Movimiento frontal).

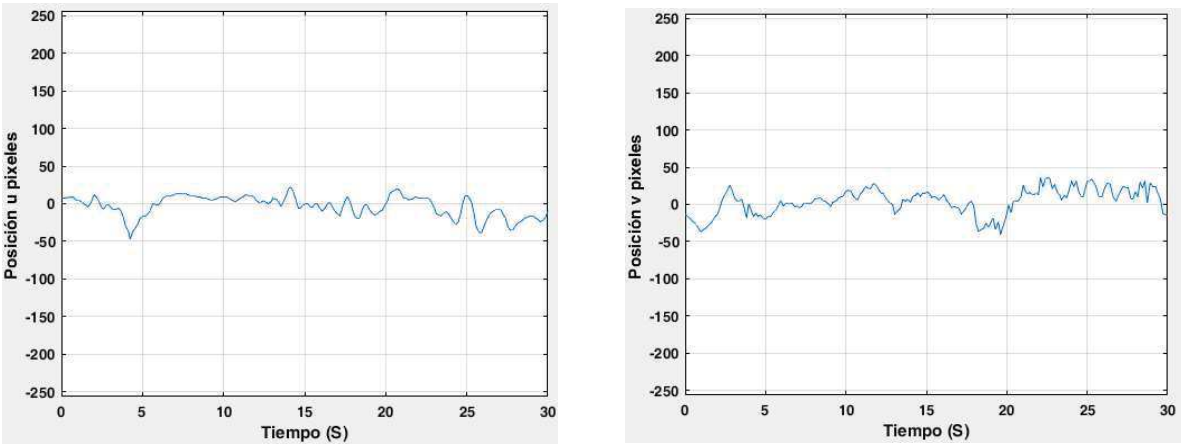


Figura 10 Posicionamiento del dron esclavo a) u pixeles (Movimiento lateral), b) v pixeles (Movimiento de elevación).

8. Conclusiones

Como se observa en las gráficas 10 a) y 10 b) de la sección de resultados, el posicionamiento de elevación y movimiento lateral del dron es realizado de tal forma en la que el dron pueda posicionarse detrás del maestro, ya que la relación de error en pixeles no es tan grande al convertirlo a centímetros, mientras que el posicionamiento en distancia que se observa en la figura 5 d) tiene grandes variaciones al inicio esto se debe al movimiento del dron esclavo que se da en el inicio para alcanzar la distancia deseada, también se observa un transitorio al final debido a que al finalizar el experimento se sostiene al dron para detenerlo. En cuanto al posicionamiento angular no se tienen grandes variaciones debido a que se intentó mantener al dron maestro en un solo punto, por lo que en trabajos futuros se planea realizar un control de trayectoria para tener mejores resultados en el comportamiento y también se planea realizar este control líder/seguidor con más de dos drones utilizando drones tri-rotor en configuración no holonómica para mejorar la respuesta de posicionamiento angular.

En cuanto al sensor monocular es posible la detección del objeto mejora considerablemente por el uso del filtro digital, que como se puede observar las variaciones de distancia son casi óptimas, sin

embargo, el posicionamiento angular en *yaw* es bastante ruidoso a pesar de haber sido filtrado esto es debido a la variación en la distancia. En trabajos futuros se planea mejorar aún más la detección con algún filtro de tipo predictivo.

Referencias

[1] A. Abdessameud and A. Tayebi, “Attitude synchronization of a group of spacecraft without velocity measurements” IEEE Transactions on Automatic Control, vol. 54, pp 2642-2648, Nov 2009.
[2] W. Ren, “Formation Keeping and attitude alignment fro multiple spacecraft through local interactions” Journal of guidance, Control, and Dynamics, vol. 30, no. 2, pp. 633-638, 2007.
[3] W. Ren and R. W. Beard, Distributed Consensus in Multiple-vehicle Cooperative Control, Theory and Applications, Springer 2008.
[4] C. X. and S. A. ., ISS-Based Robust Leader-Follower Trailing Control. Springer Verlag 2006.
[5] J. Escareño, Conception, modelisation et commande d’un drone convertible, Université Technologie de Compiègne 2008.
[6] G. Vianna Modelado y control de un robot helicóptero quadrotor, Sevilla 2007.
[7] Birchfield, S. and Tomasi, C. A pixel dissimilarity measure that is insensitive to image sampling. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1998.
[8] Z. Zhang. A Flexible New Technique for Camera Calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330-1334, 2000.

Ingeniería y Automatización, Capítulo 31, pp. 393 - 402.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Control Simultáneo de Fuerza y Posición para Robot Manipulador en Contacto con una Superficie Desconocida

Domínguez Fuentes Juan Ramón y García Valdovinos Luis Govinda

Centro de Ingeniería y Desarrollo Industrial (CIDESI),
Av. Playa Pie de la Cuesta No. 702. Desarrollo San Pablo. Querétaro, Qro. México

Resumen

En este trabajo se presenta una solución al problema de seguimiento de trayectorias del efector final de un robot manipulador sobre una superficie considerada altamente rígida desconocida, usando un control de fuerza y posición basado en el principio de ortogonalización. La solución propuesta parte de la aproximación de una función geométrica que describa a la superficie desconocida, por medio de una nube de puntos obtenida por un escáner 3D. Con la información de la función geométrica aproximada se planifica una trayectoria sobre la superficie y se define un perfil de fuerza deseado; por lo que, se realizó una simulación del seguimiento de una trayectoria con el perfil de fuerza definido sobre la superficie aproximada. Los resultados de la simulación son usados para validar la efectividad de la solución propuesta.

Palabras clave: Control fuerza/posición, robot manipulador, superficie desconocida, aproximación de superficies.

1. Introducción

Los robots industriales han sido usados en diversas aplicaciones, tales como pulido, soldadura y revestimiento. En estas aplicaciones, los robots industriales requieren de un control de posición del efector final y de la fuerza que aplica sobre la superficie [1]. Sin embargo, cuando está presente la interacción de un robot manipulador con una superficie, existe un conjunto de consideraciones para determinar el control y la trayectoria deseada necesarios para el desarrollo de la aplicación, las cuales son: la morfología del manipulador, la función geometría de la superficie y la rigidez de la superficie de contacto [2]. No obstante, es posible que las superficies con las que se interactúa tengan geometrías complejas y no se conozca una función matemática que las describa.

Si se considera una superficie altamente rígida (indeformable) y se tiene conocimiento de la morfología del manipulador, entonces el factor de la geometría del ambiente es la que puede variar dependiendo de la superficie. Por lo que, el problema es que no se puede garantizar el seguimiento de trayectorias sobre la superficie por el desconocimiento de una función geometría que describa a la superficie.

Existen trabajos relacionados a la implementación de controles de fuerza y posición en robots manipuladores que interactúan con superficies [3,4,5], en los cuales la función geométrica que describe a la superficie es desconocida. Sin embargo, estos trabajos son estrategias que se plantean para casos particulares enfocados a la geometría. Por lo tanto, generalizar el problema del conjunto de posibles superficies con geometrías simples hasta complejas se basa en el hecho de si es posible aproximar una función geométrica que describa a la superficie. Además, si la superficie que se encuentra en contacto con el efector final de un robot es altamente rígida, entonces es posible implementar un control de fuerza y posición en el robot que garantice la convergencia a cero de los errores de posición y fuerza.

En este trabajo se plantea el uso de un esquema de control simultaneo de fuerza y posición, basado en el principio de ortogonalización. Sin embargo, una restricción del esquema de control es conocer la función geométrica que describa a la superficie. Por lo que, se propone un método, basado en reconstrucción en 3D por medio de un escáner laser, que permita aproximar una función geométrica que describa la superficie.

2. Antecedentes

2.1 Modelo dinámico de un robot restringido en el efector final

Considere un robot manipulador de n grados de libertad cuando el efector final está en contacto con una superficie sin fricción y altamente rígida, donde la superficie esta descrita por una ecuación algebraica dada por

$$\varphi(x(q)) = 0 \tag{1}$$

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + B_0\dot{q} + g(q) = \tau + J_{\varphi+}^T(q)\lambda \tag{2}$$

donde $q \in \mathbb{R}^n$ es el vector de coordenadas articulares, $\dot{q} \in \mathbb{R}^n$ es el vector de velocidades articulares, $\ddot{q} \in \mathbb{R}^n$ es el vector de aceleraciones articulares, $H(q) \in \mathbb{R}^{nxn}$ es la matriz de inercia, $C(q,\dot{q}) \in \mathbb{R}^{nxn}$ es la matriz de coriolis y fuerzas centrípetas, $B_0 \in \mathbb{R}^{nxn}$ es la matriz de fricción viscosa, $g(q) \in \mathbb{R}^n$ es el vector de pares gravitacionales, $\tau \in \mathbb{R}^n$ es el vector que contiene a la ley de control, $\lambda \in \mathbb{R}$ es un escalar que representa a la fuerza que se produce en el punto de contacto, $J_{\varphi+}(q) \in \mathbb{R}^{1xn}$ es la proyección normalizada del jacobiano de $\varphi(x(q))$ denotado por

$$J_\varphi(q) = \frac{\partial\varphi(x(q))}{\partial x}\frac{\partial x}{\partial q} = J_\varphi(x) * J(q) \in \mathbb{R}^{1xn} \tag{3}$$

$$J_{\varphi+}(q) = \frac{J_\varphi(q)}{J_\varphi(q)J_\varphi^T(q)} \tag{4}$$

2.2 Principio de ortogonalización

El principio de ortogonalización [6] está basado en la propiedad física que establece que cuando se ejerce fuerza sobre una superficie altamente rígida (indeformable), y al mismo tiempo se desplaza sobre ella, el vector de velocidad generalizado está contenido en el plano tangente al punto de contacto, mientras la aplicación de la fuerza es perpendicular al plano, como se ilustra en la figura 1. La restricción es manipulada a fin de generar dos subespacios ortogonales; así pues, se utilizan para hacer la proyección ortogonal del vector articular $\dot{q}$ sobre el plano tangente $Q(q) \in \mathbb{R}^{nxn}$ ortogonal al vector $J_\varphi(q)$ y la proyección ortogonal de la integral de error de fuerza de contacto sobre el vector $J_\varphi(q)$.

El plano tangente está representado por

$$Q(q) = I_{nxn} - J_{\varphi+}^T(q)J_\varphi(q) \tag{5}$$

donde I_{nxn} es la matriz identidad. Dado que, $Q(q)$ y $J_\varphi^T(q)$ pertenecen a subespacios ortogonales, mientras el efector final del manipulador se mueve sobre la superficie se cumple lo siguiente

$$Q(q) * J_\varphi^T(q) = 0 \quad \Rightarrow \quad Q(q) * J_{\varphi+}^T(q) = 0 \tag{6}$$

$$J_\varphi(q) * \dot{q} = 0 \quad \Rightarrow \quad Q(q) * \dot{q} = 0 \tag{7}$$

Ingeniería y Automatización, Capítulo 31, pp. 393 - 402.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

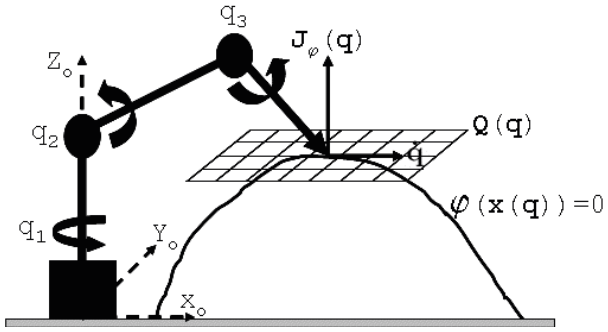


Figura 1. Descomposición geométrica en el punto de contacto

2.3 Control de fuerza y posición

En el seguimiento de una trayectoria sobre una superficie rígida aplicando un perfil de fuerza predefinido no se puede garantizar el seguimiento de la trayectoria por el desconocimiento de una función geometría que describa a la superficie. Por lo tanto, se propone un control PID deslizante ortogonal [7].

En primer lugar, el modelo dinámico del robot manipulador puede ser linealmente parametrizado en términos de una referencia nominal $\dot{q_r}, \ddot{q_r} \in \mathbb{R}^n$ como

$$H\ddot{q_r} + [B_0 + C]\dot{q_r} + g = Y_r\Theta \tag{8}$$

donde $Y_r(q, \dot{q}, \dot{q_r}, \ddot{q_r}) \in \mathbb{R}^{nxp}$ llamado regresor está compuesto de funciones no lineales conocidas y $\Theta \in \mathbb{R}^p$ representa parámetros constantes desconocidos. A continuación, restando en ambos miembros de (2) la parametrización lineal en términos de la referencia nominal, resultando la siguiente ecuación de error en lazo abierto

$$H\dot{S_r} = \tau + J_{\varphi+}^T(q)\lambda - [B_0 + C]S_r - Y_r\Theta \tag{9}$$

donde el error extendido está definido como

$$S_r = \dot{q} - \dot{q_r} \tag{10}$$

Permite diseñar una ley de control que produzca convergencia exponencial de posición y de fuerza. Por consiguiente, se propone la siguiente referencia nominal

$$\begin{aligned}\dot{q_r} = Q[\dot{q}_d - \alpha\Delta q + S_{dp} - \Upsilon_{p1}\int_{t_0}^{t} S_{qp}(\sigma)d\sigma - \Upsilon_{p2}\int_{t_0}^{t} sign(S_{qp}(\sigma))d\sigma] \\ + \beta J_\varphi^T[\Delta F - S_{dF} + -\Upsilon_{F1}\int_{t_0}^{t} S_{qF}(\sigma)d\sigma - \Upsilon_{F2}\int_{t_0}^{t} sign(S_{qF}(\sigma))d\sigma]\end{aligned} \tag{11}$$

donde $\Delta q = q - q_d$, $\Delta F = \int_{t_0}^{t} \Delta\lambda(\sigma)d\sigma = \int_{t_0}^{t}(\lambda - \lambda_d)(\sigma)d\sigma$. Además, $\alpha, \Upsilon_{p1}, \Upsilon_{p2} \in \mathbb{R}^{nxn}$ son matrices diagonales definidas positivas, y $\beta, \Upsilon_{F1}, \Upsilon_{F2} > 0$ son ganancias constantes. El subíndice *d* denota las variables de referencia deseada, los subíndices *p* y *F* indican subespacios referentes a posición y fuerza, respectivamente. Sustituyendo (11) en (10), se obtiene

$$S_r = QS_{vp} - \beta J_\varphi^T S_{vF} \tag{12}$$

donde S_{vp} y S_{vF} están definidos por

$$S_{vp} = S_{qp} + \Upsilon_{p1} \int_{t_0}^{t} S_{qp}(\sigma)d\sigma + \Upsilon_{p2} \int_{t_0}^{t} sign(S_{qp}(\sigma))d\sigma \quad (13)$$

$$S_{vF} = S_{qF} + \Upsilon_{F1} \int_{t_0}^{t} S_{qF}(\sigma)d\sigma + \Upsilon_{F2} \int_{t_0}^{t} sign(S_{qF}(\sigma))d\sigma \quad (14)$$

con

$S_{qp} = S_p - S_{dp}$	variedad extendida de posición
$S_p = \Delta\dot{q} + \alpha\Delta q$	superficie deslizante de posición
$S_{dp} = S_p(t_0)e^{-k_p t}$	sesgo de posición
$S_{qF} = S_F - S_{dF}$	variedad extendida de fuerza
$S_F = \Delta F$	superficie deslizante de fuerza
$S_{dF} = S_F(t_0)e^{-k_F t}$	sesgo de fuerza

donde $k_p, k_F > 0$ son constantes. En (12) los complementos Q y J_φ proyectan los errores de seguimiento de posición y velocidad, y de la integral de fuerza sobre subespacios ortogonales, respectivamente. Así pues, se propone la siguiente ley de control

$$\tau = -K_d S_r + J_{\varphi+}^T[-\lambda_d + \eta\Delta F + \Upsilon_{F1}S_{qF} + \Upsilon_{F2}\tanh(\rho S_{qF})] + J_{\varphi+}^T[\eta\Upsilon_{F1}\int_{t_0}^{t} S_{qF}(\sigma)d\sigma + \eta\Upsilon_{F2}\int_{t_0}^{t} sign(S_{qF}(\sigma))d\sigma] \quad (15)$$

donde $\eta, \rho > 0$ son constantes, $K_d \in \mathbb{R}^{nxn}$ es una matriz diagonal definida positiva y la función tangente hiperbólica es utilizada para aproximar la función signo.

2.4 Modelado de la fuerza de reacción

Con el fin de lograr que (1) se cumpla se propone un control que estabilice la restricción y sus derivadas en cero, tal que

$$\ddot{\varphi}(q) = J_\varphi \ddot{q} + \dot{J_\varphi}\dot{q} = u(t) \quad (16)$$

donde $u(t)$ garantiza que $\varphi(q) \to 0, \dot{\varphi}(q) \to 0$ y $\ddot{\varphi}(q) \to 0$.

Despejando la aceleración articular de (2), se obtiene

$$\ddot{q} = H(q)^{-1}[\tau + \frac{J_\varphi^T(q)}{J_\varphi(q)J_\varphi^T(q)}\lambda - C(q,\dot{q})\dot{q} - B_0\dot{q} - g(q)] \quad (17)$$

sustituyendo (17) en (16)

$$J_\varphi H(q)^{-1}[\tau + \frac{J_\varphi(q)}{J_\varphi(q)J_\varphi^T(q)}\lambda - C(q,\dot{q})\dot{q} - B_0\dot{q} - g(q)] + \dot{J_\varphi}\dot{q} = u(t) \quad (18)$$

La fuerza de reacción λ que define al Lagrangiano restringido de (18) es

$$\lambda = \frac{J_\varphi(q)J_\varphi^T(q)}{J_\varphi(q)H(q)^{-1}J_\varphi^T(q)}\left[u(t) - \dot{J_\varphi}\dot{q} + J_\varphi H(q)^{-1}[-C(q,\dot{q})\dot{q} - B_0\dot{q} - g(q)]\right] \quad (19)$$

La convergencia de las restricciones de posición y velocidad del área de contacto a cero no puede ser garantizado. Por lo que, se propone un filtro estable de segundo orden con entrada cero [8] definidos por

$$\ddot{\varphi}(q) = -2\omega_1\dot{\varphi}(q) - \omega_2^2\varphi(q) = -2\omega_1 J_{\varphi}(q)\dot{q} - \omega_2^2\varphi(q) \quad (20)$$

donde $\omega_1, \omega_2 > 0$ son ganancias seleccionadas apropiadamente, las cuales aseguran la convergencia rápida a cero de la restricción y su derivada.

3. Caso de estudio

La selección de la superficie con la que se pretende interactuar se basa en las suposiciones de la metodología de solución. Por lo que, se propuso una superficie, como se ilustra en la figura 2, que cumple con las siguientes características:

1) Superficie altamente rígida (indeformable)
2) La función geométrica que la describa es suave, al menos dos veces derivable

Figura 2. Fotografía de segmento de pala de aerogenerador

Además, se propuso un robot manipulador compuesto de articulaciones rotacionales de 5 grados de libertad actuado solamente en 3 con el fin de validar la propuesta, como se ilustra en la figura 3. Adicionalmente, en la tabla 1 se pueden observar los parámetros del robot.

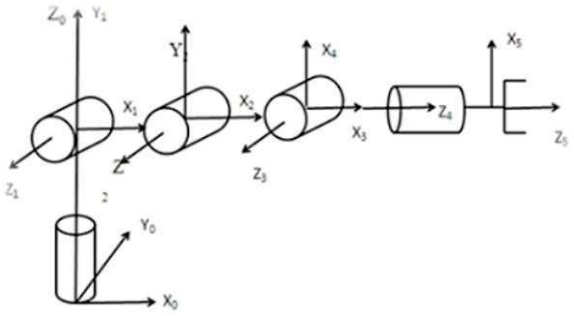


Figura 3. Diagrama de alambre del robot Catalyst-5

Ingeniería y Automatización, Capítulo 31, pp. 393 - 402.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Tabla 1. Parámetros del robot Catalyst-5.

Elemento	Variable	Valor	Unidad
Masa del eslabón 1	m_1	9.0	Kg
Masa del eslabón 2	m_2	2.0	Kg
Masa del eslabón 3	m_3	1.0	Kg
Masa del eslabón 4	m_4	0.4	Kg
Masa del eslabón 5	m_5	0.7	Kg
Longitud del eslabón 1	l_1	0.2504	m
Longitud del eslabón 2	l_2	0.2504	m
Longitud del eslabón 3	l_3	0.2504	m
Longitud del eslabón 4	l_4	0	m
Longitud del eslabón 5	l_5	0.1410	m
Aceleración de la gravedad	g	9.81	m/s^2
Carga máxima de peso	M_{max}	1.5	Kg

4. Simulación

4.1 Aproximación de la superficie

En primer lugar, es necesario escanear la superficie deseada, ya que se busca obtener una función geométrica que describa a la superficie, como la ilustrada en la figura 2. Por lo que, por medio de un escáner 3D se obtiene una nube de puntos que representa a la superficie con la que se deseada trabajar, como se ilustra en la figura 4 inciso a. Sin embargo, es necesario orientar la nube de puntos del espacio del escáner al espacio del robot manipulador.

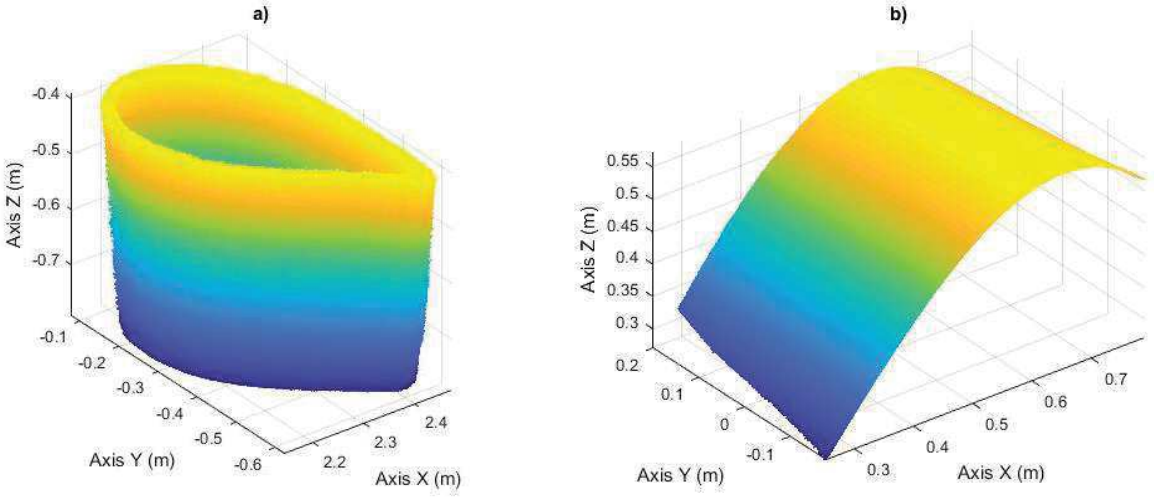


Figura 4. Gráfica de la superficie, a) Nube de puntos escaneada y b) Superficie aproximada

Con la información de la nube de puntos orientada con respecto al sistema de referencia del robot es posible aproximar una función geométrica. Por lo que, se propone (21)

$$z = a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2 + a_7xy^2 + a_8x^2y + a_9x^3 + a_{10}y^3 \quad (21)$$

donde $x, y, z \in \mathbb{R}$ son elementos que pertenecen al conjunto de la nube de puntos orientada al espacio de trabajo del robot y $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} \in \mathbb{R}$ son las constantes que definen a la superficie.

4.2 Planificación de la trayectoria deseada

Con el fin de validar la propuesta de solución, se planifica una trayectoria de posición descrita por (22), como se ilustra en la figura 5; y se propone un perfil de fuerza descrito por (23), como se ilustra en la figura 6.

$$\begin{aligned} x_d &= x_0 + \left(\frac{t}{t_b}\right) * (x_f - x_0) \\ y_d &= 0 \\ z_d &= a_1 + a_2x + a_4x^2 + a_9x^3 \end{aligned} \quad (22)$$

donde $x_d, y_d, z_d \in \mathbb{R}$ son las posiciones deseadas en cada eje cartesiano, $x_0 \in \mathbb{R}$ es la posición inicial de la trayectoria en el eje x, $x_f \in \mathbb{R}$ es la posición final de la trayectoria en el eje x, $t \in \mathbb{R}$ es el tiempo, $t_b \in \mathbb{R}$ es el tiempo base de la trayectoria.

$$\lambda_d = F + C * (\sin 2\pi\, t) \quad (23)$$

donde $\lambda_d > 0 \in \mathbb{R}$ es la fuerza aplicada deseada por el efector final del robot y $F, C \in \mathbb{R}$ son constantes.

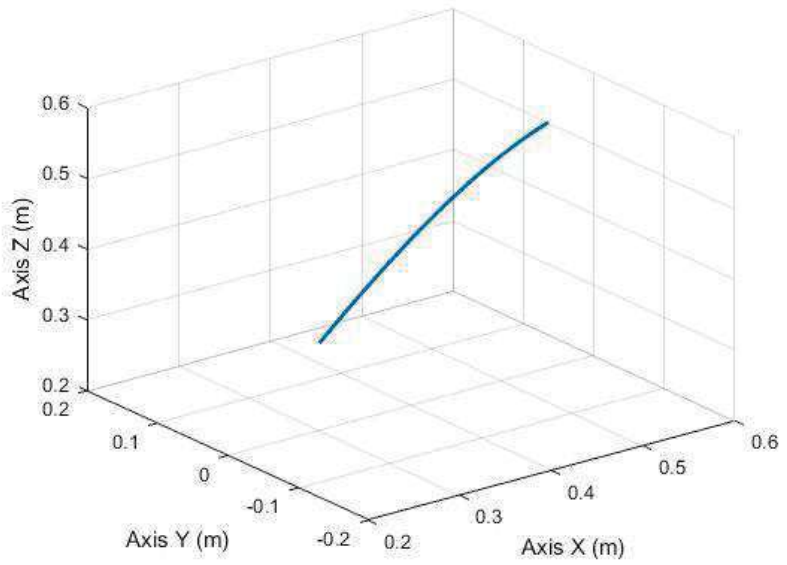


Figura 5. Gráfica de la trayectoria de posición deseada sobre la superficie

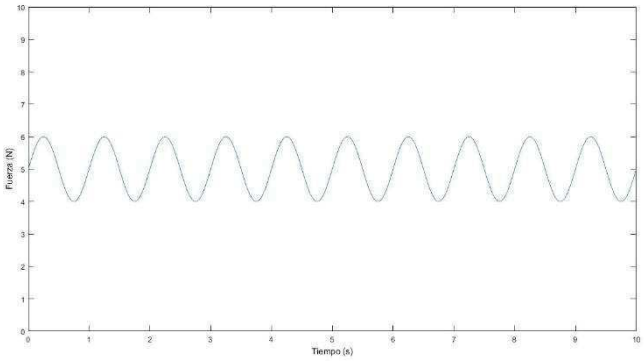


Figura 6. Gráfica del perfil de fuerza deseado sobre la superficie

4.3 Seguimiento de la trayectoria

Como se ilustra en la figura 7, se realizó el seguimiento de (22) y (23) considerando como condiciones iniciales $x_0 = 0.3\,m,\ y_0 = y_f = 0.0\,m,\ z_0 = 0.3236\,m,\ x_f = 0.55\,m,\ z_0 = 0.548\,m,\ F = 5,$ $C = 1$ y $t_b = 10\,s$. En la figura 8 se pueden observar los errores de posición en cada articulación durante el seguimiento de la trayectoria. Asimismo, en la figura 9 se puede observar el comportamiento de la fuerza aplicada con respecto al perfil de fuerza deseada.

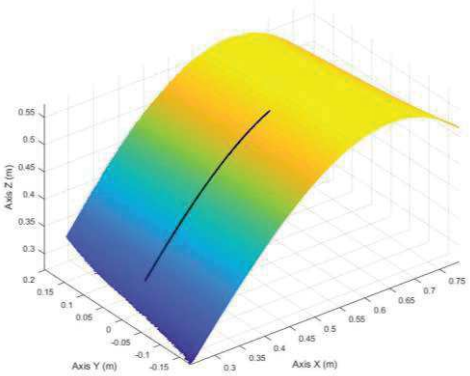


Figura 7. Gráfica del seguimiento de la trayectoria sobre la superficie

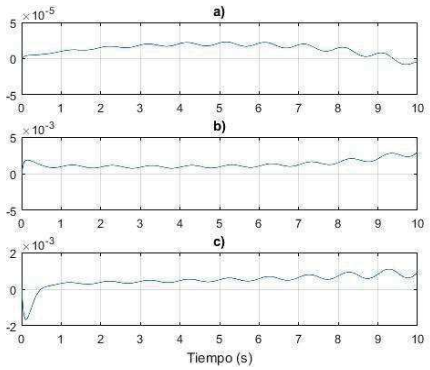


Figura 8. Gráficas del error de la trayectoria, a) Articulación 1, b) Articulación 2, c) Articulación 3

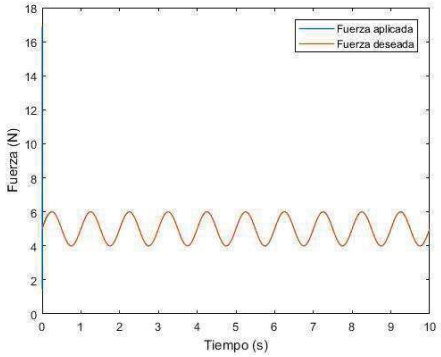


Figura 9. Gráfica del seguimiento del perfil de fuerza deseado

4.4 Implicaciones del principio de ortogonalización

Las restricciones del principio de ortogonalización son manipuladas a fin de generan los subespacios ortogonales $Q(q)$ y $J_{\varphi}(q)$. Por lo que, mientras el efector final del manipulador se mueve sobre la superficie se cumple (6) y (7). Como se observa en la figura 10, durante el seguimiento de la trayectoria se verificó que se cumplieran

a) $Q(q) * J_{\varphi}^{T}(q) = 0$
b) $Q(q) * J_{\varphi+}^{T}(q) = 0$
c) $J_{\varphi}(q) * \dot{q} = 0$
d) $Q(q) * \dot{q} = 0$

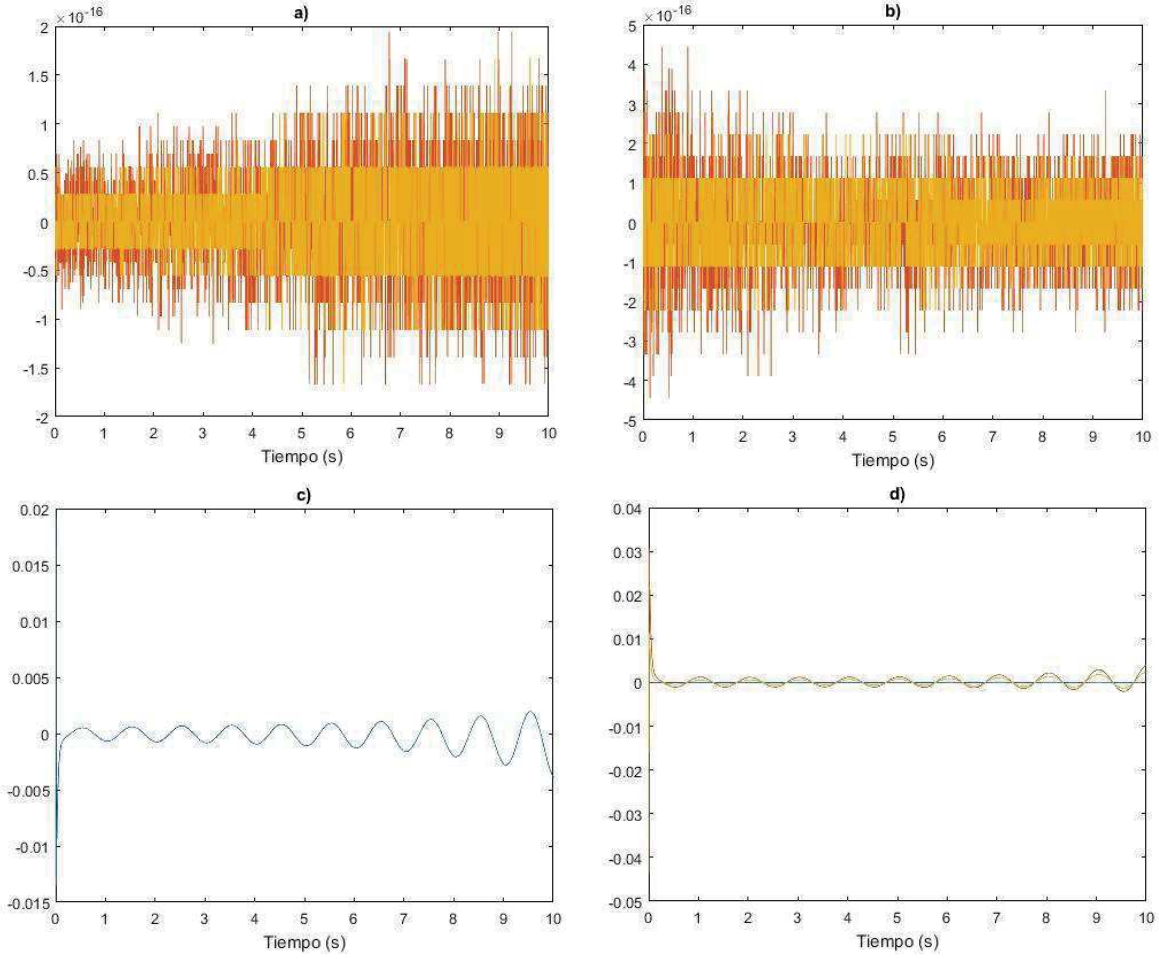


Figura 10. Gráficas de la comprobación de las propiedades del principio de ortogonalización, a) Prueba 1, b) Prueba 2, c) Prueba 3 y d) Prueba 4

5. Conclusiones

En esta investigación se propuso una metodología que garantice el seguimiento de trayectorias sobre una superficie desconocida a partir de la aproximación de una función geométrica que describa a la superficie. Esta metodología se apoya en la reconstrucción 3D por medio de un escáner para obtener una nube de puntos.

La manipulación de la nube de puntos permite aproximar una función geométrica, cuyo dominio está definido por la información que proporciona el conjunto que elementos que conforman a la nube de puntos. La función geométrica aproximada permite una mejor planificación de trayectorias paramétricas, con relación a las técnicas actuales en la planificación de trayectorias cuando se desconoce una función geométrica. Cabe resaltar que, el alcance de este artículo es un acercamiento al enfoque de aproximación de superficies. Se busca investigar diferentes técnicas con el fin de obtener una función geométrica óptima para el desarrollo de aplicaciones.

Referencias

[1] Vukobratovi M., Stoji R. *"Historical perspective of hybrid control in robotics: beginnings, evolution, criticism and trends"*, Elsevier, Belgrade, Serbia, 1994.
[2] Siciliano B., Sciavicco L. *"Robotics Modelling, Planning and Control",* Springer, España, 2009.
[3] Ouyang P.R., Acob J., Pano V. *"PD with sliding mode control for trajectory tracking of robotic system"*, Elsevier, Toronto, Canada, 2013.
[4] Baspinar C. *"On Robust Position/Force Control of Robot Manipulators with Constraint Uncertainties"*, IFAC, Wolfsburg, Germany, 2012.
[5] Mehdi H., Boubaker O., "*Robust Tracking Control for Constrained Robots"*, Elsevier, Tunis, Tunisia, 2012.
[6] Arimoto S., Liu Y. "*Principle of orthogonalization for hybrid control of robot arm"*, Elsevier, Tokyo, Japan, 1996.
[7] Pliego-Jimenez J., Arteaga M., "*Adaptive position/force control for robot manipulators in contact with a rigid surface with unknown parameters*", ECC, México, México, 2015
[8] Gudiño-Lau J, Arteaga M., "*Control de Robots con Movimiento Restringido: Modelado y Experimentación"*, AMCA, México, México, 2004.

Ingeniería y Automatización, Capítulo 32, pp. 403 - 413.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Robótica Competitiva en RUNIBOT Colombia, Aprendizaje Significativo

Merchán Vaquiro Carlos Arturo[1] y Burgos Hernández María Cristina[2]
Universidad UNIMINUTO- Universidad De La Baja California [1], Universidad De Cuauhtémoc[2],
ingmerchan@gmail.com, mariacburgoshernandez@gmail.com

Resumen

El presente artículo de reflexión, presenta los resultados estadísticos de las participaciones de los colegios y universidades de Colombia en las cinco versiones del evento organizado por la Red Universitaria Bogotá Robot RUNIBOT[1]. Por tanto, se analiza el comportamiento de las participaciones de los colegios distritales con respecto a los privados, así mismo de la cantidad de robots participantes por parte de las universidades y el impacto en los resultados de adicionar categorías en colegios, así como las nuevas categorías en universidades.

En tanto que la participación de los colegios es importante para estos eventos porque son estos estudiantes los futuros estudiantes de las universidades y es por esta razón que se debe evaluar las habilidades y competencias que desarrollan en la etapa de preparación de los estudiantes al evento, la posterior participación, finalizando con las experiencias significativas aprendidas derivadas de estas participaciones en la que los alumnos y docentes evalúan los aspectos estratégicos, tecnológicos y competitivos por mejorar para las próximas participaciones.

Dentro de las experiencias de los colegios se destaca la participación de los equipos conformados por los estudiantes, docentes y en algunos casos los padres de familia que trabajan en equipo fomentando en los jóvenes el aprendizaje significativo que obtienen como resultado de cada una de las experiencias en estos eventos con una adquisición de conocimiento integrado, coherente y estable que tiene sentido para los alumnos y que a futuro les representa ser personas competentes para los retos que les planteara la vida profesional.

Palabras clave: Experiencia significativa, RUNIBOT, Aprendizaje Significativo, Robótica competitiva.

1. Introducción

Actualmente la tecnología que se entiende como un producto de la ciencia y la ingeniería que envuelve un conjunto de instrumentos, métodos y técnicas que se encargan de la resolución del conflicto [1], es la que sigue abarcando cada vez más las diferentes áreas del conocimiento, incursionando incluso de forma vertiginosa en el área de la educación, en donde los gobiernos deben apostarle a políticas educativas que integren la tecnología como base y apoyo fundamental en los procesos académicos de aprendizaje, transformado la educación en beneficio de las oportunidades de desarrollo de los países, teniendo en cuenta las ventajas y desventajas que conlleva la apropiación de las nuevas tecnologías de la información y la comunicaciones para la educación, sobre todo en regiones como América Latina, en donde el uso desmedido e incontrolado de la tecnología hace que los estudiantes sean más dispersos y los procesos de aprendizaje no sean los adecuados y se presenten resultados como en las pruebas PISA, en donde los países latinoamericanos están lejos de los primeros puestos [2].

[1] RUNIBOT: Red Universitaria Bogotá Robot, conformada por las Universidades central, cooperativa, el bosque, católica, uniagraria y Uniminuto.

En consecuencia, el avance tecnológico ha desarrollado de manera acelerada la robótica, puesta a servicio en la industria y la sociedad, facilitando los trabajos que el hombre realizaba con mayor esfuerzo, tiempo y recursos, permitiendo llegar a tener una calidad de vida inimaginable hace siglos atrás. Por tanto, para llegar al desarrollo de estas tecnologías robóticas, se hace necesario que los países fomenten en sus políticas educativas, la educación en tecnología, en donde los estudiantes puedan tener acceso al conocimiento y manipulación de software y hardware especializado en desarrollos robóticos.

Por tanto, se hace considerable que instituciones o redes organizadas, realicen eventos de robótica competitiva en los que los estudiantes logren presentar desarrollos en hardware y software que permitan evidenciar los avances tecnológicos en las instituciones educativas colegios y universidades, y más aún, que permitan enfrentarse competitivamente con desarrollos tecnológicos de otras instituciones en los que se evidencien los avances de cada institución educativa. En este sentido, la robótica competitiva cobra relevancia en cuanto a la motivación que genera dentro de los jóvenes estudiantes de colegios y universidades que pueden ver en ella una opción de vida y que sienten que les aporta en habilidades blandas[2] y duras[3] para su posterior vida profesional.

Al respecto, las experiencias significativas de los colegios distritales en Bogotá, muestran como cada colegio con los pocos o muchos recursos asignados apuestan por dar a los estudiantes una calidad educativa en la que involucran la robótica competitiva como medio motivador para adquirir competencias en trabajo en equipo, colaborativo, gestión de tiempo y recursos, pensamiento lógico, creatividad, búsqueda de información, y las competencias académicas propias de la educación, influenciadas estas prácticas por la motivación de los docentes en la participación en eventos competitivos que dejen en los estudiantes experiencias de vida[3].

Es así, como en eventos como RUNIBOT Colombia se abre una oportunidad para que los colegios y universidades en Colombia presenten el desarrollo de diferentes tipos de tecnologías y sea un aprendizaje significativo al conocer las tecnologías que trabajan las demás instituciones educativas, incluso conocer los desarrollos tecnológicos que presentan instituciones internacionales. Por tanto, se concluye de acuerdo a los resultados de RUNIBOT Colombia, que la participación de colegios y universidades cada vez es mayor gracias a la motivación creada por toda la comunidad académica que se dedica a la robótica competitiva.

2. RUNIBOT Colombia

La red universitaria Bogotá robot RUNIBOT, fue creada en el año 2013 en un evento realizado en la universidad UNIMINUTO, cuando varias universidades motivadas por los temas de la robótica competitiva, llegaron allí para participar en este evento, surge la idea de organizar entre varias universidades de Bogotá un único evento que tuviera un gran impacto a nivel nacional e internacional, con el apoyo de todas las universidades. Por tanto, en el año 2013 se procedió a formalizar la RED de las universidades con la misión de realizar un Megaevento anual que como sede sea cada una de las diferentes universidades de la RED y que involucre las universidades y los colegios de Bogotá inicialmente.

.

Por tanto, desde el año 2014 se desarrolló el primer RUNIBOT en la universidad El Bosque, en el cual se diseñaron categorías para colegios y categorías para universidades, generando la motivación de las instituciones educativas en la participación de este tipo de eventos de robótica competitiva.

[2] Habilidad Blanda: es el resultado de una combinación de habilidades sociales, de comunicación, de forma de ser, de acercamiento a los demás, que hacen a una persona relacionarse y comunicarse de manera efectiva con otros.

[3] Habilidad Dura: son las destrezas técnicas que se adquieren y que son necesarias para realizar una determinada actividad.

2.1 Runibot 2014 – 1018 Colegios

En el año 2014, se realiza el primer RUNIBOT con sede en la Universidad el Bosque, donde se desarrolló en dos días el evento divididos en un día para competencias de colegios y el segundo día con competencias de universidades y aficionados. Por tanto, para el primer día de colegios se contó con 27 robots de colegios de los cuales el 80% fueron colegios distritales[4], evidenciando una mayor acogida por parte de los colegios que cuentan generalmente con pocos recursos tecnológicos porque no son utilizados adecuadamente. Así mismo, se contó con la colaboración de la SED[5] Bogotá en cuanto al transporte de los estudiantes de los colegios distritales, pero no se contó con la ayuda para pagar la inscripción de cada robot, las inscripciones fueron pagadas con recursos de las instituciones e incluso algunas fueron pagadas con recursos de los propios alumnos motivados por participar en el evento.

En el año 2015, se realiza el segundo RUNIBOT con sede en la Universidad Central, donde se desarrolló en dos días el evento divididos en un día para competencias de colegios y el segundo día con competencias de universidades y aficionados. Por tanto, para el primer día de colegios se contó con 32 robots de colegios de los cuales el 70% fueron colegios distritales, evidenciando una menor participación de los colegios distritales respecto a los privados. Pero se presenta un incremento del 18% de cantidad de robots de colegios participantes respecto al año 2014.

En el año 2016, se realiza el tercer RUNIBOT con sede en la Universidad Católica de Colombia, donde se desarrolló en tres días el evento, para este año debido al gran impacto del evento en los años anteriores si hizo necesario aumentar un día para el desarrollo del evento, divididos en un día para competencias de colegios y dos días con competencias de universidades y aficionados. Por tanto, para el primer día de colegios se contó con 65 robots de colegios de los cuales el 60% fueron colegios distritales, evidenciando menor participación de los colegios distritales respecto a los privados. Pero se presenta un incremento de más del 100% de cantidad de robots de colegios participantes respecto al año 2015, debido a la estrategia de agregar una nueva categoría para colegios como lo fue Sumo, en la que se presentan 21 robots que para comenzar es una buena cantidad.

En el año 2017, se realiza el cuarto RUNIBOT con sede en la Universidad Cooperativa de Colombia, donde se desarrolló en tres días el evento, divididos en un día para competencias de colegios y dos días con competencias de universidades y aficionados. Por tanto, para el primer día de colegios se contó con 76 robots de colegios de los cuales el 50% fueron colegios distritales, evidenciando igual participación de los colegios distritales respecto a los privados y se presenta un incremento del 14.7% de cantidad de robots de colegios participantes respecto al año 2016, mostrando una cifra considerable y en aumento en cada una de las participaciones en el evento.

En el año 2018, se realiza el quinto RUNIBOT con sede en la Universidad UNIMINUTO, donde se desarrolló en tres días el evento, divididos en un día para competencias de colegios y dos días con competencias de universidades y aficionados. Por tanto, para el primer día de colegios se contó con 106 robots de colegios de los cuales el 30% fueron colegios distritales, evidenciando una baja participación de los colegios distritales respecto a los privados y se presenta un incremento del 40% de cantidad de robots de colegios participantes respecto al año 2017, mostrando una alta participación de robots de colegios, pasando la meta de los 100 robots, cifra que no se ha considerado en ningún evento en Colombia con estudiantes de Colegio. Además, se estrena en esta quinta versión la categoría de futbol de colegios en la que se presentan 18 equipos que para comenzar no es una cifra insignificante.

Como se ilustra en la figura1, con cada versión de RUNIBOT, se muestra un avance significativo en cuanto a la participación de los robots de los colegios, lo que demuestra que el interés por las participaciones en este evento es alto por parte de los colegios y que las categorías nuevas

[4] Colegio Distrital: Colegio que funciona con recursos del gobierno.
[5] SED: secretaria de educación del distrito de Bogotá.

son llamativas para los estudiantes porque les permite tener nuevos desarrollos de los convencionales, además se presentó casos en los que un equipo de colegio con un solo lego participaba en seguidor de línea y posteriormente lo ajustaban físicamente y la programación para competir en Sumo, lo que ejemplifica que con una excelente motivación y con escasos recursos si es posible participar. Por tanto, también se evidencia en la figura 1, como se ha incrementado la participación de los colegios a través de cada versión de RUNIBOT, logrando pasar las metas establecidas en participación.

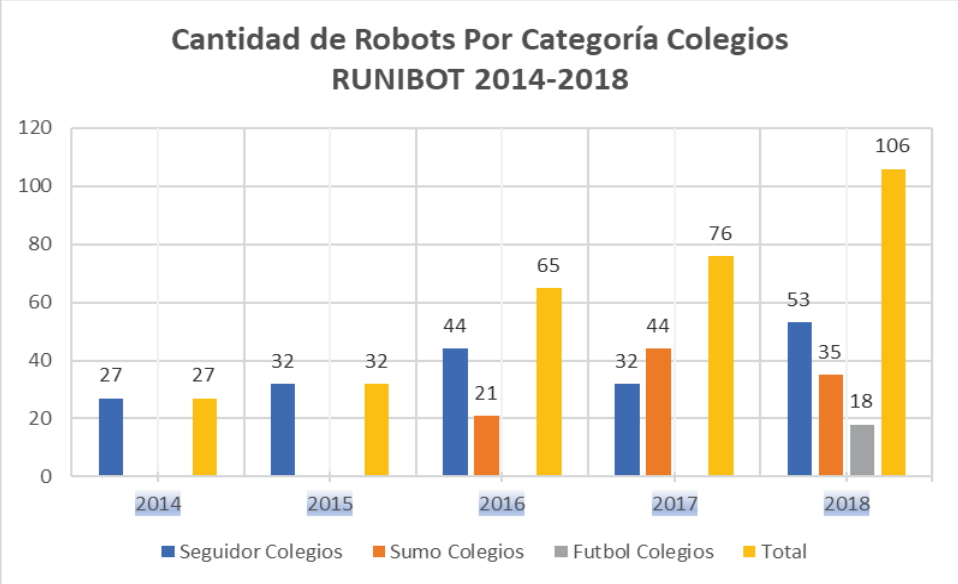


Figura 1. Cantidad Robots Participantes Colegios

En la Figura2. Se observa la competencia de colegios en las categorías actuales de Sumo autónomo, futbol y seguidor reto.

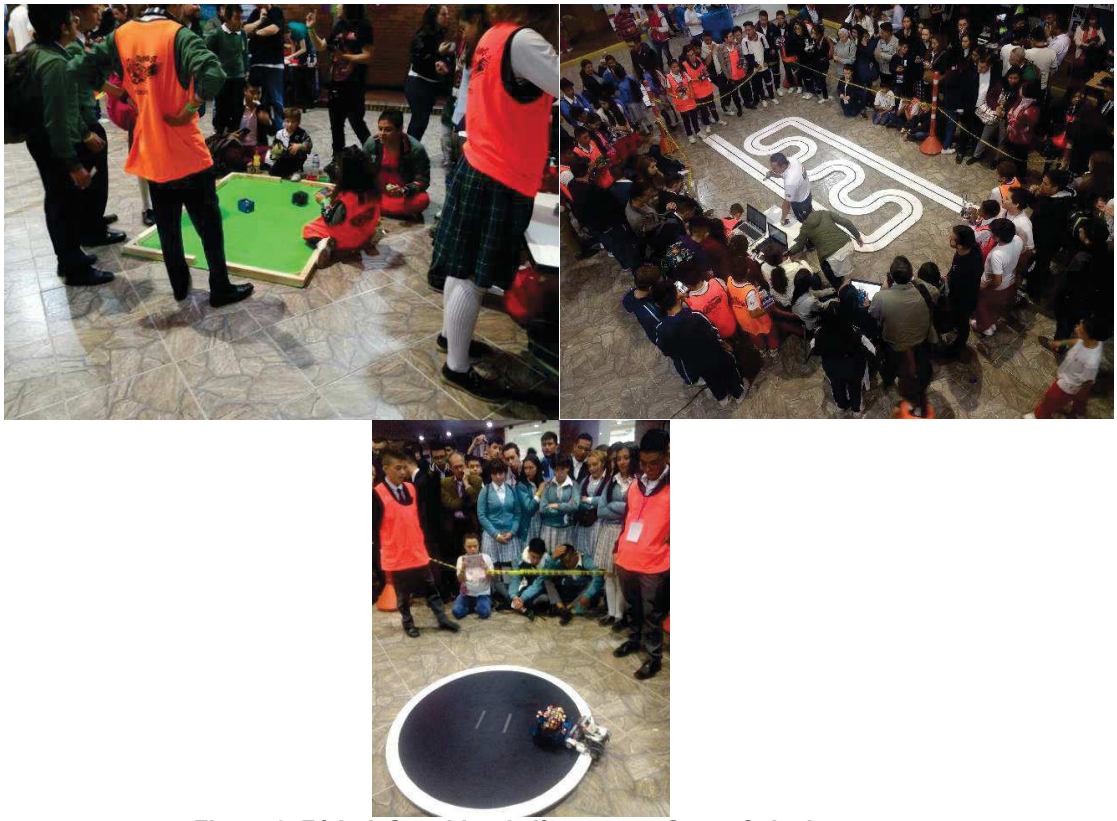


Figura 2. Fútbol, Seguidor de línea reto y Sumo Colegios.

2.2 Runibot 2014-2018 Universidades

Para el primer RUNIBOT 2014 con sede en la Universidad el Bosque, en el día de competencias de universidades y aficionados se presentaron a competencia 72 robots, de las cuales el 80% son representación de universidades de Bogotá, divididos en cuatro categorías, velocista, sumo autónomo, mini sumo y robótica social. En esta versión los robots seguidor de línea velocistas son los que tienen la mayor acogida y en la categoría de robótica social se tiene la menor cantidad debido a que en esta categoría se debe construir un robot que al estar en una pista determinada, tenga la capacidad de detectar latas de aluminio y latas plásticas, al seleccionarla debe recogerla y guardarla en un espacio destinado para ellas dentro del mismo robot y al final gana el robot que más envases recoja y haya seleccionado y depositado adecuadamente, como se observa en la figura3.

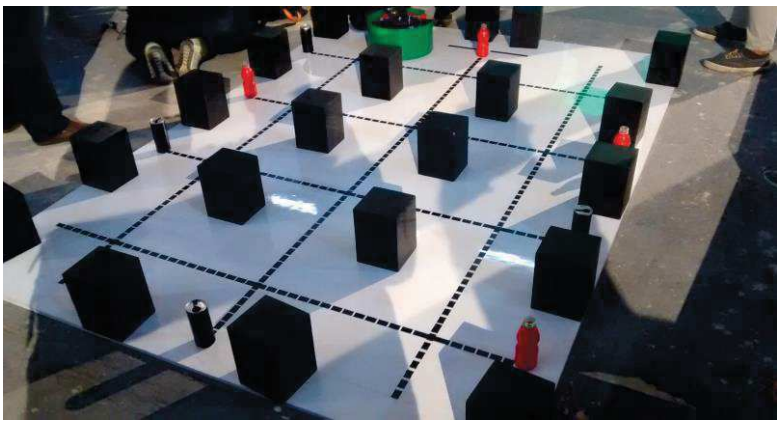


Figura 3. Pista de Robot Social (RUNIBOT 2014)

Para el segundo RUNIBOT 2015, con sede en la Universidad Central, en el segundo día que fue para las universidades y para los aficionados, se presentaron 192 robots en competencia representando un incremento del 167% en cantidad de robots, respecto al 2014. En consecuencia, la publicidad del evento se evidencia en la gran participación en el evento 2015, además, se adiciona la categoría desarrollo e innovación tecnológica, que consiste en el planteamiento de una solución a un problema social por parte de los estudiantes, en donde se premió la propuesta más innovadora.

Para el tercer RUNIBOT 2016, con sede en la Universidad Católica de Colombia, para la categoría de universidades y aficionados, se adiciono un día debido a la alta acogida del evento entre los participantes y a la adición de la categoría Sumo RC. Se presentaron 239 robots en competencia incrementando en un 25% la cantidad de robots, respecto al 2015. Por tanto, en esta versión se adicionó la categoría de sumo RC para entregar la certificación al all japan robot sumo tournament 2016, en esta categoría Sumo RC participaron 29 robots que es una buena cantidad para estar por primera vez. Adicionalmente, en esta versión del evento se contó con la participación de delegaciones internacionales de países como México, Ecuador, Perú y Japón.

Para el cuarto RUNIBOT 2017, con sede en la Universidad Cooperativa de Colombia, para la categoría de universidades y aficionados, se presentaron 177 robots en competencia decrementando en un 26% la participación del año 2016, aun así, en esta versión del evento se adicionó la categoría de futbol robótico y se eliminó la categoría de robot social por la baja acogida en los últimos eventos. Por tanto, En la nueva categoría de futbol robótico se presentaron 23 equipos, siendo una cantidad razonable y muy llamativa para la comunidad académica.

Para el quinto RUNIBOT 2018, con sede en la Universidad UNIMINUTO, para la categoría de universidades y aficionados, se presentaron 241 robots en competencia incrementando en un 37% la participación con respecto al año 2017. Para esta versión, se analizó la baja participación en el 2017 y como resultado de esta reflexión se optó por agregar categorías innovadoras y llamativas que motivaran a los estudiantes en el desarrollo de robots de competencia, es así, como se agregaron las categorías de Drones, robot batalla de 1lb y 3lb y microsumo, como resultado se incrementó la participación de los robotistas universitarios y aficionados logrando superar la cantidad de robots en competencia de las cuatro versiones anteriores. Es así, como en la categoría de drones se presentan 13 robots en competencia, en la categoría micro sumo se presentan 19 robots en competencia, en la categoría robot batalla de 1lb se presentan 7 robots a competencia y en robot batalla de 3 lb se presentan 6 robots en competencia, aclarando que estas dos últimas categorías son una lucha que puede ser destructiva entre los dos robots, por tal razón los estudiantes también se cuidan en la participación porque pueden salir totalmente afectado su robot y destruido en el peor de los casos.

Como se ilustra en la figura 4, se evidencia la evolución de la participación de las universidades y aficionados siendo las categorías de minisumo y seguidor de línea velocista las que más se destacan en todas versiones de RUNIBOT, por tanto, son las categorías que se deben seguir fortaleciendo para mantener la participación y las demás seguirlas evaluando para determinar que hace que sean atractivas para los estudiantes y la pertinencia de seguirlas manteniendo en las siguientes versiones de RUNIBOT. Así mismo, se destaca la participación de competidores internacionales en todas las categorías y las 16 ciudades colombianas que ya se han integrado al evento.

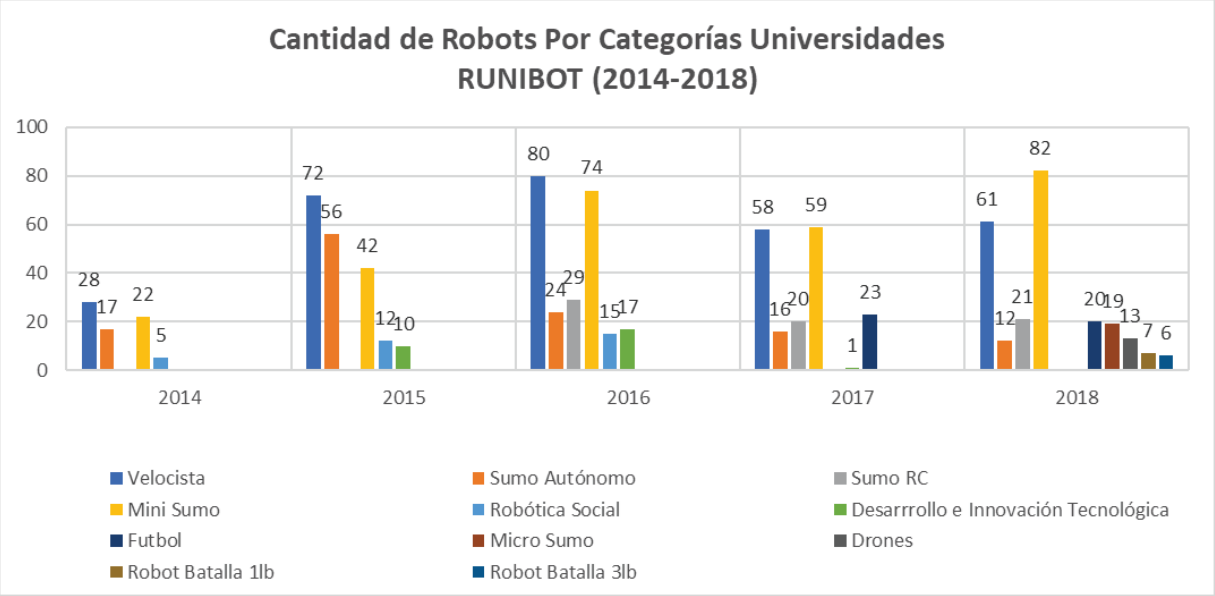


Figura 4. Cantidad de Robots por categorías universidades y aficionados.

En la figura 5 se observa dos de las actuales categorías de universidades, robot batalla y drones.

Figura 5. Categoría Robot Batalla y Drones Universidades y Aficionados 2018.

3. Impacto de las experiencias significativas en los colegios

La transformación cultural y tecnológica que enfrenta la escuela en el país, necesita de eventos como el organizado por la *Red Universitaria Bogotá Robot RUNIBOT* donde se desarrolle competencias robóticas actualizadas en las que se pueda reflexionar sobre los objetivos de las competencias, evaluar el desempeño y resultados, siendo críticos con los compromisos adquiridos en la enseñanza y aprendizaje de las tecnologías usadas, desarrollando habilidades y competencias duras y blandas que los estudiantes requerirán posteriormente para enfrentar las necesidades de la sociedad.

La ley general de educación 115 de 1994 considera (artículo 23- numeral 9) [4] al área de Tecnología e Informática como área fundamental y obligatoria por ello se imparte en las instituciones de carácter oficial y privada en los grados de secundaria y media. Las Orientaciones para la Educación en Tecnología, proponen definir el alcance, el sentido y la coherencia de las competencias requeridas, estas se formularon a partir de la interrelación que se da con otros campos como la técnica, la ciencia, la innovación, invención y descubrimiento, diseño, informática y ética. Es así, como

independiente de los campos que se designen como base para el trabajo en Tecnología se deben tener en cuenta los énfasis (la investigación, el manejo y la proyección de la información, los procesos interdisciplinarios, el trabajo en equipo, la autonomía y los valores), en los cuales se adquieren las habilidades, las capacidades y las competencias necesarias para acceder a mayores niveles de integración y comprensión del área de Tecnología [5].

En las instituciones educativas de Básica y Media demuestran interés ante los eventos competitivos como el organizado por la *Red Universitaria Bogotá Robot RUNIBOT* para innovar y orientar procesos de investigación en el área de la educación en Tecnología a través de un análisis sistemático del proceso enseñanza–aprendizaje, es por ello que se evidencia el aumento de la participación de la escuela a estos eventos, como lo muestra la tabla1 la participación de los colegios en todas las versiones del evento RUNIBOT.

Tabla 1. Participación de colegios en todas las versiones de RUNIBOT.

Categorías	2014	2015	2016	2017	2018
Seguidor Colegios	27	32	44	32	53
Sumo Colegios			21	44	35
Futbol Colegios					18
Total	27	32	65	76	106

Los eventos en robótica son una tentativa para hacer viable la transformación pedagógica de la escuela desde la enseñanza - aprendizaje de la comunidad académica[6], se conciben como herramientas que pueden ayudar a hacer pertinente la oferta de las instituciones educativas respecto al proyecto de vida de los estudiantes y contribuir a la construcción de proyectos educativos locales.

Desde las instituciones se proyectan a ser reconocidos por innovar, actualizar, investigar y reflexionar en torno a los principios de la robótica, donde se evidencia a la comunidad educativa una nueva manera de abordar la Tecnología. Por tanto, la robótica es una Tecnología multidisciplinar, hace uso de todos los recursos de vanguardia de otras ciencias afines, que soportan una parte de su estructura y se ha desarrollado como una perspectiva de acercamiento a la solución de problemas derivados de distintas áreas del conocimiento como: las Ciencias de Información, la Tecnología, Matemáticas, Ciencias Naturales, entre otras [6]. Así mismo, uno de los factores más interesantes es que la integración de diferentes áreas se da de manera natural porque en el desarrollo de la clase se necesita informarse sobre diferentes conocimientos para poder lograr un proyecto en el aula, es por ello la importancia de la participación de las comunidades educativas en los principios básicos de la robótica educativa teniendo en cuenta la manera natural de trabajar los énfasis propuestos para el área. Sin embargo, la presencia de Tecnologías en el aula de clase, busca proveer ambientes de aprendizaje interdisciplinarios donde los estudiantes adquieran habilidades para estructurar investigaciones y resolver problemas concretos, ayudando a las personas a desarrollar nuevas habilidades, conceptos y dar respuestas eficientes a los entornos cambiantes del mundo [7].

Dentro de las experiencias significativas como resultado de la participación de los estudiantes de colegios en los eventos de robótica, están el desarrollo de habilidades y competencias que hacen de los estudiantes personas que se desenvuelven adecuadamente evidenciando que han adquirido la capacidad de:

- Construir estrategias para la resolución de problemas, utilizando el método científico para probar y generar nuevas hipótesis,
- Usar vocabulario especializado y construir sus propias concepciones acerca del significado de cada objeto que manipulan.

[6] Comunidad Académica: Conformada por los estudiantes, profesores, padres de familia

- Seleccionar piezas de construcción como: ejes, engranajes, poleas, actuadores y sensores útiles según el diseño.
- Reconocer, clasificar y tomar decisiones sobre la conveniencia del uso de ciertas piezas estimando el tamaño y acople.
- Interiorizar conceptos tecnológicos tales como: diseño y construcción a través de modelos o prototipos que simulan objetos encontrados en el entorno.
- Aplicar los conceptos de programación en el control de un robot, reconocer la morfología de un robot.

Este tipo de competencias has dado como resultado la adquisición de habilidades que se desarrollan a partir del aprendizaje significativo que conduce a la creación de estructuras de conocimientos mediante la relación sustantiva entre la nueva información y las ideas previas de los estudiantes que se traduce en la implementación de sus robots y las estrategias que ellos utilizan para el desarrollo de la competencia, evidenciando que existe una amplia gama de estructuras de software y hardware que permite modelar de varias maneras los robots.

Cuando en estos eventos de robótica competitiva se dan cambios importantes en la estructura de conocimientos como resultado de la asimilación de la nueva información, gracias al intercambio de información tecnológica entre los competidores asistentes quienes muestran sus desarrollos en software y hardware de manera que los demás aprendan de estas nuevas tecnologías y las puedan desarrollar para las próximas competencias. Es así, como el aprendizaje significativo como resultado de las experiencias d ellos colegios, cobra relevancia porque implica un procesamiento muy activo de la información por aprender. Para que realmente sea significativo el aprendizaje, la nueva información debe relacionarse de modo no arbitrario y sustancial con lo que el alumno ya sabe. Cuando se habla de que haya relacionabilidad no arbitraria, se quiere decir que, si el material o contenido de aprendizaje en sí no es aleatorio y tiene la suficiente intencionalidad, habrá una manera de relacionarlo con las clases de ideas pertinentes que los seres humanos son capaces de aprender [8].

Los estudiantes después de vivir la experiencia de participar en un evento de robótica competitiva como RUNIBOT, guardan toda la información vista, escuchada, memorizada, grabada y entran a un modo de reflexión con todos estos datos adquiridos, de tal manera que al ser procesados se generen nuevos procesos mentales que desarrollen nuevas estrategias de hardware y software y los lleven a validar nuevamente en las experiencias prácticas. En consecuencia, el aprendizaje significativo es relevante académico en los eventos de robótica como se muestra en la figura 6.

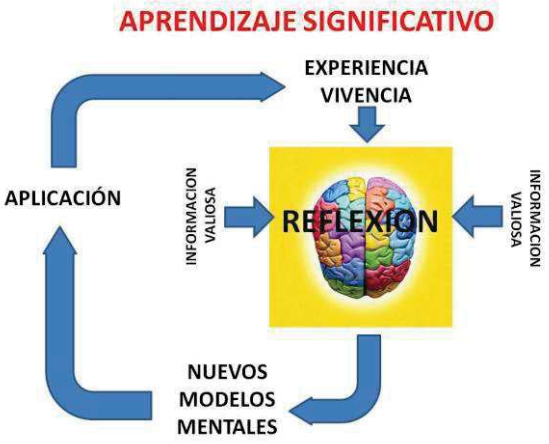


Figura 6. Etapas del aprendizaje significativo[7].

[7] Tomado de: https://sites.google.com/site/mateportafoliojosue/_/rsrc/1447970361127/noviembre-2015/aprendizaje-significativo/Aprendizaje-Significativo-1.jpg

Las características que enmarcan los resultados de las experiencias significativas en los estudiantes de los colegios, se presentan a través de como el aprendizaje se facilita cuando los contenidos al alumno se les presentan organizados de manera conveniente y siguen una secuencia lógica y psicológica apropiada, que se dan en el aprendizaje de la robótica. Por tanto, es conveniente delimitar las posibles intencionalidades y los contenidos de aprendizaje en una progresión continua que respete los niveles de inclusividad, abstracción y generalidad.

La experiencia de los colegios en las competencias de robótica, activa los conocimientos y experiencias previas en su estructura cognitiva, facilitando los procesos de aprendizaje de los estudiantes ayudándoles a que puedan detectar ideas, organizarlas e integrarlas significativamente, porque los conocimientos adquiridos significativamente, son estables para los estudiantes, menos vulnerables a ser olvidados y más fácil de transferirse [9].

4. Conclusiones

RUNIBOT es un ejemplo en Colombia de como se pueden unir esfuerzos para lograr metas en beneficio común, con visión de favorecer la academia en beneficio de la población estudiantil de los colegios y de la universidades que año a año acompañan estos megaeventos, y de cómo se construye comunidad y redes académicas internacionales que siguen en constante crecimiento, como se evidencia en los resultados de las diferentes versiones de RUNIBOT, incrementando la participación y motivación en la comunidad académica.

El especial seguimiento que se hace a los colegios es fundamental porque son los futuros estudiantes de las universidades y hay que mantenerlos motivados para que continúen en la línea tecnología y sea los que aporten soluciones a los problemas sociales que se encontraran en la vida profesional, generándole valores morales y éticos desde las competencias robóticas a las que asisten y competencias comunicativas al estar en contacto con otros equipos de otros colegios.

Las experiencias significativas de los estudiantes que participan en eventos de robótica, hacen que generen competencias y habilidades que son aplicables no solo en la participación de eventos de robótica, sino que también pueden aprenderlas para desarrollarlas en otros aspectos de la vida. Por tanto, esas experiencias, se transforman en un aprendizaje significativo en donde los alumnos toman lo mejor de las experiencias vividas en la competencia y lo transforman en nuevos modelos mentales que al colocarlos en la practica fomentan nuevos desarrollos de hardware y software.

El articulo muestra como desde el primer RUNIBOT 2014 realizado en la sede Universidad el Bosque se contó con una participación de 27 robots en pista por parte de los colegios, y en el RUNIBOT 2018 realizado en la sede Universidad UNIMINUTO se llegaron a 106 robots en competencia, presentando un incremento de 392% durante estas cinco versiones, lo cual afirma que la motivación por la participación en este tipo de eventos de robótica competitiva sigue incrementándose en las generaciones actuales, en beneficio de las universidades que a futuro tendrán a estos estudiantes de colegios en sus aulas.

Las universidades durante las cinco versiones oficiales de RUNIBOT, han presentado un incremento del 334% en la cantidad de robots en competencia, mostrando como en RUNIBOT 2014 se llegó a 72 robots en competencia y en RUNIBOT 2018 se llegó a 241 robots en competencia, lo que demuestra que al igual que en los colegios los universitarios han mostrado motivación e interés en la participación de estos eventos y que las nuevas categorías que se involucran tienen una muy buena acogida por parte de la comunidad académica robotista.

Finalmente, se hace una reflexión a las entidades que manejan la educación centralizadas y descentralizadas para que apoyen la participación de los colegios en estos eventos que como se muestra en este articulo la mayoría de colegios se inscriben y asisten con recursos propios, solamente en el primer RUNIBOT2014 se tuvo apoyo en el transporte de los competidores al evento.

Referencias

[1] "Tecnología" (s. f.). En: *Significados.com*. Disponible en: https://www.significados.com/tecnologia/: 1 de agosto de 2018.

[2] Sunkel, G., & Trucco, D. (2010). *Nuevas tecnologías de la información y la comunicación para la educación en América Latina: riesgos y oportunidades*. CEPAL.

[3] Quintero Cano, C. A. (2010). *Enfoque ciencia, tecnología y sociedad (CTS): perspectivas educativas para Colombia*. Zona próxima, (12).

[4] de Educación, L. G. Ley 115 de 1994 (1994). Bogotá, Colombia*:* Ediciones FECODE.

[5] Acevedo, G. D. R. (1998). *Ciencia, Tecnología y Sociedad: una mirada desde la Educación en Tecnología*. Revista Iberoamericana de Educación, 18, 107-143.

[6] Bravo Sánchez, F. Á., & Forero Guzmán, A. (2012). *La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales*. Teoría de la Educación. Educación y Cultura en la Sociedad de la Información, 13(2).

[7] Moreira, M. A. (2010). *Introducción a la tecnología educativa*. DIM: Didáctica, Innovación y Multimedia, (19), 1-78.

[8] Moreira, M. A. (1997). *Aprendizaje significativo: un concepto subyacente*. Actas del encuentro internacional sobre el aprendizaje significativo, *19*, 44.

[9] Odorico, A. (2004). *Marco teórico para una robótica pedagógica*. Revista Informática Educativa y Medios Audiovisuales, 1(3), 34-46.

Ingeniería y Automatización, Capítulo 33, pp. 414 - 423.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Incubadora con Agitación Orbital

Téllez Cabrera Ricardo Alberto, Carbajal Cruz Alain Israel, Juárez Aparicio Jesús Albín, Cruz Almaraz Viridiana y Gaspar Antonio Lizbeth

Universidad Tecnológica de Xicotepec de Juárez
inghenritrejo@gmail.com

Resumen

El articulo describe el desarrollo de un prototipo de código abierto de una incubadora con agitación orbital se tiene conocimiento que una incubadora sirve para mantener y reproducir cultivos microbiológicos o cultivos celulares, regulando factores de crecimiento, algunos de los factores que interviene son la temperatura, la humedad y la ventilación.

El prototipo consta de tres elementos: Un sistema mecánico el cual realiza el movimiento orbital, balanceando dinámicamente para evitar vibraciones. Un sistema térmico aplicando convección forzada, obteniendo una temperatura homogénea en la cámara de incubación con uso de ventiladores. Un sistema electrónico donde se programó un controlador de temperatura. Se realizó una interfaz humano-máquina para el ingreso de variables como el tiempo de trabajo de la incubadora, la velocidad y temperatura del equipo.

Las pruebas realizadas en el prototipo son de control de la velocidad de agitación y homogenización de la temperatura en su interior. El desarrollo de este primer prototipo se realizó con tecnología abierta y materiales de bajo costo obteniendo niveles similares de eficiencia comparadas con quipos industriales.

La realización de la incubadora fue con fines de poder complementar el aprendizaje de los alumnos del área de biotecnología al desarrollar sus prácticas con microorganismos llamados enzimas los cuales necesitan de ciertos parámetros para su reproducción.

1. Introducción

La filosofía de código abierto ha revolucionado y cambiado sustancialmente la forma de ver y crear cientos de soluciones a problemas cotidianos. Las ventajas que presenta la tecnología abierta se ven reflejada en el crecimiento de proyectos, accesibilidad, la dependencia tecnológica se minimiza, costos reducidos entre otros. En este ámbito los quipos de laboratorio no son la excepción y es por ello que se ha planteado el desarrollo del prototipo de la incubadora utilizando dichas ventajas.

Las incubadoras con agitación orbital se encuentran constituidas principalmente por la integración de dos equipos: el primero es una cámara diseñada para mantener una temperatura, atmosfera y humedad controladas con el objetivo de conservar organismos vivos en un entorno adecuado para su crecimiento. El segundo con un mecanismo de agitación orbital que se utiliza en los laboratorios para la homogenización y preparación de combinaciones de sustancias en función de la aplicación con el objetivo de regular la velocidad de funcionamiento.

Las incubadoras son equipos de usos continuo que sirven para mantener el desarrollo microbiológico progresivo de cultivos bacterianos especialmente. En los ambientes mencionados se regulan factores de crecimiento en condiciones óptimas como, por ejemplo, la temperatura, la humedad y la ventilación. Además, incluyen la capacidad de controlar temperaturas extremadamente bajas(microbiológicas) humedad y niveles de dióxido de carbono y agitación.

El presente artículo presenta una alternativa tecnológica de código abierto y abajo costo. Este prototipo de incubadora con agitación orbital ha sido realizado con el objetivo de contribuir a las investigaciones que se desarrollan en el área de Biotecnología.

La función principal de la incubadora será para la producción de enzimas las cuales son biomoléculas especializadas en la catálisis de las reacciones químicas que tiene lugar en la célula. Son muy eficaces como catalizador artificial conocido, y además son altamente específicos ya que cada uno de ellos induce la transformación de un solo tipo de sustancia y no de otras que se puedan encontrar en el medio de reacción.

Para su creación se necesita de ciertos parámetros los cuales son la temperatura en la que se reproducen y el número de revoluciones a las cuales deben de estar es por ello que el prototipo consta de tres elementos: Un sistema mecánico que realiza el movimiento orbital el cual será el encargado del funcionamiento del giro orbital, un sistema térmico que monitorea la temperatura en la cámara de incubación con uso de ventiladores los cuales se usaran para regular la temperatura dentro de la cámara, un sistema electrónico en el que interviene un controlador de temperatura del cual dependerá del tipo de microorganismo a reproducir, esto permitirá una interfaz humano-máquina para el ingreso de variables.

2. Objetivos

2.1 Objetivo general:

Construir una incubadora con agitación en un tamaño reducido (prototipo) para él área de Biotecnología de la Universidad Tecnológica de Xicotepec de Juárez, aplicando los conocimientos en control y automatización adquiridos en la carrera de Mecatrónica Área Automatización.

2.2 Objetivos Específicos:

- Fortalecer el aprendizaje de los alumnos al realizar sus prácticas.
- Ingreso de variables por el usuario.
- Reproducción de microorganismos.

3. Justificación

El proyecto del prototipo de incubadora con agitación se llevará a cabo para satisfacer la necesidad del área de biotecnología que no cuenta con este equipo para realizar sus prácticas.
Gracias a la elaboración de este equipo se espera que el área de biotecnología pueda producir enzimas que son necesarias para la realización de sus prácticas dando como resultado una mejor enseñanza a los alumnos, de igual manera se lograría tener una ganancia económica puesto que su venta es de alto costo.

Ingeniería y Automatización, Capítulo 33, pp. 414 - 423.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

4. Diseño del prototipo

4.1 Objetivo del diseño:

Aplicar los conocimientos adquiridos en el área de diseño, desarrollando piezas y ensamblándolas para darle una estructura al prototipo de la incubadora.

4.2 Diseño de estructura:

El diseño se llevó a cabo en un software llamado INVENTOR, este se desarrolló a partir de las especificaciones planteadas, a través del análisis se optó porque el diseño contara con tres fases para una mejor producción, dando como resultado el boceto siguiente.

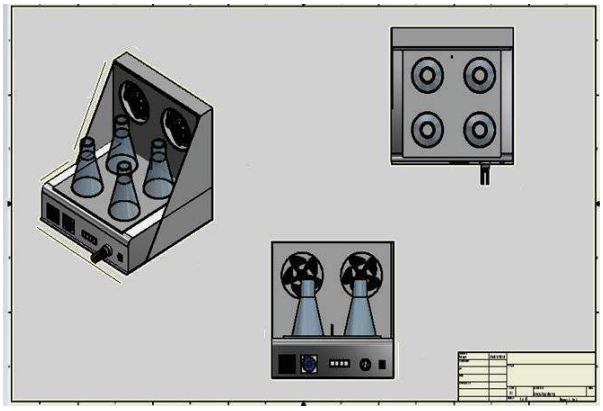


Figura 1. Boceto del diseño de incubadora con giro orbital

La figura 1 muestra el boceto de la incubadora en la cual se observa la tanto el lado frontal, el lado superior y el lado superior izquierdo.

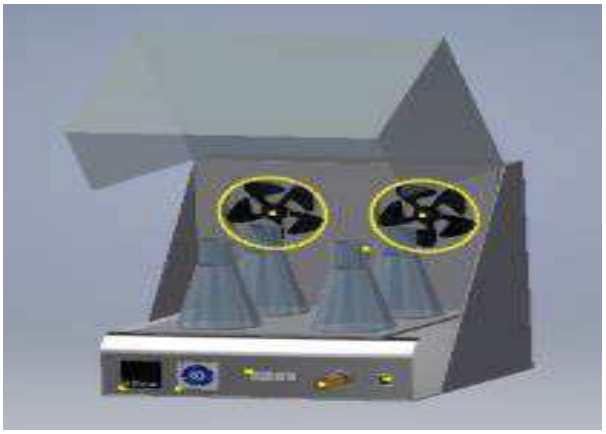


Figura 2. Diseño de incubadora de giro orbital

La figura 2 muestra el diseño final de la incubadora de giro orbital, la que contara con una tapa flexible para poder cerrar y abrir de acuerdo a la necesidad del usuario.

Ingeniería y Automatización, Capítulo 33, pp. 414 - 423.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

5. Construcción del prototipo

La construcción del proyecto se llevó a partir de tres fases importantes en la cuales se analizó de manera detallada cada fase dando como resultado las especificaciones siguientes:

5.1 Sistema mecánico:

El sistema refiere a los componentes mecánicos que permiten a la plataforma de la incubadora poseer un movimiento orbital. El sistema se compone de un motor de corriente directa el cual tendrá la función de hacer girar a un disco de aluminio donde irán colocados los matraces, en medio del disco ira un eje motriz ya que este se encargará de mantener firme el mecanismo.

5.2 Sistema electrónico y Control:

Este sistema consta de componentes electrónicos y eléctricos encargados de interpretar y ejecutar los comandos que sean ingresados por el operador. El sistema contiene de sensores de temperatura DTH22 los cuales son muy preciosos en cuestión de medición. Otro aspecto a controlar es la velocidad de agitación de la incubadora de la cual depende el motor es así que se incorporó un Arduino Mega que envía las señales en función de la velocidad ingresada por el usuario. Las señales enviadas para el funcionamiento del motor son recibidas y procesadas por un driver que se encuentra anexo a la fuente de alimentación dedicada al motor de 9 VDC.7

La incubadora con agitación posee una interfaz que facilita al usuario ingresar los datos y mantener un control de funcionamiento en tiempo real. Así mimo se implementó un botón de emergencia el cual bloquea todo el sistema de agitación y de incubación en caso de ocurrir accidentes. Finalmente se instaló un interruptor de dos posiciones para el encendido y apagado de todo el sistema del equipo.

5.3 Sistema térmico:

Su función principal es generar calor y además de transferir el mismo desde el origen (resistencia eléctrica) hacia la cámara de incubación. El calor el distribuido a través de dos ventiladores que se encargan de mantener una temperatura dentro de la misma. El principal objetivo es tener un ambiente controlado en todos los puntos donde se ubican las muestras o cultivos a incubar y agitar. El diseño de la cámara de incubación debe de ser completamente hermética, es decir que no debe de existir ninguna fuga o escape de calor hacia el exterior. Esto ayuda a que las muestras no se contaminen.

A partir de esto se ha defino un sistema térmico en dos cámaras separadas a través de una rejilla diseñada para el adecuado flujo de aire. La primera es una cámara de calentamiento en la cual se encuentra colocada una resistencia eléctrica de 800W, con suficiente potencia para calentar el volumen de aire contenido. En la frontera de las dos cámaras se encuentran ubicados dos ventiladores los mismo que cumplen con la distribución del aire caliente que circulará de la cámara de calentamiento a la de incubación. Los ventiladores se encargan de transferir el aire desde la cámara de incubación hacia la cámara de calentamiento y viceversa.

La segunda cámara de incubación es el lugar donde se encontrarán las muestras (enzimas) las cuales necesitan de un ambiente con una temperatura constante.

5.4 *Especificaciones de materiales*

Las siguiente Tabla 1 muestra las especificaciones de cada componente utilizando en la construcción de la incubadora, al igual que el costo de cada componente, teniendo como resultado el presupuesto total que se le invirtió al prototipo.

Tabla 1 Especificaciones y costos

INSUMO	ESPECIFICACIONES	COSTO
Motor DC 9 Kg	- Rango de voltaje: 3 o 12V - Voltaje nominal: 9V - Velocidad: 20160 RPM - Corriente: 680 mA - Corriente máxima: 3.8 A - Torque nominal: 600 g.cm - Potencia: 20W	$400.00
2 ventiladores.	- Medida de 20cm x 20cm - Voltaje de 220 – 240 VCA - Potencia: 60 W - Velocidad: 2500 – 2800 RPM	$200.00
Resistencia.	- Voltaje: 120V - Potencia: 800W	$120.00
4 sensores de Temperatura DHT22	- Alimentación de 3.3 a 6 V DC. - Rango de temperatura: -40 °C a 80 °C. - Precisión +- .5 °C. - Resolución: 0.1 °C.	$400.00
Chumacera de piso.	- Hecha de aluminio con espacio para 4 matraces y su propósito sostener estos.	$300.00
Banda tipo 3 V.	- Para relacionar el movimiento del motor con sus respectivos ejes.	$70.00
4 relé 5 Volts.	- Funciona a 5 Volts. - Capaz de controlar una salida de 120 Volts.	$130.00
Fuente de voltaje.	- Fuente de voltaje de 120 Volt a 12Voltos, capaz de controlar diferentes líneas de alimentación.	$200.00
Cable.	- Cable para Protoboard	$80.00
Estaño.	- Soldadura electrónica.	$60.00
Resistores.	- Resistores de múltiples valores.	$30.00
Transistores.	- Transistores PNP.	$60.00
5 PCB (Placa).	- Placa para montar el respectivo circuito.	$400.00
Arduino Mega.	- Controlador de pequeñas acciones estéticas (encendido de leds y detalles estéticos).	$350.00
Mano de obra.	- Elaboración de diagramas de conexión respectivos.	$1000.00
	TOTAL	**$5,070.00**

6. Área de pruebas

Durante el desarrollo del prototipo se analizo a fondo los factores y parámetros que intervienen en el proceso de producción de enzimas, esto se llevo a cabo en 3 pruebas como lo es:

Prueba de temperatura: en este se tomó en cuenta el punto máximo y mínimo que alcanza la temperatura para esto se posiciono dos sensores termopares, uno en la parte baja de la cámara térmica y otro en la parte media dando como resultado que la temperatura en ambo putos es constante, esto se observó gráficamente en un programa diseñado en LabView.

Prueba de velocidad: esta se llevó a cabo en la parte electrónica y control del sistema, gracias al programa estructurado en la plataforma de Arduino usando un control electrónico conocido como PWM que consiste en controlar el pulso eléctrico que es enviado al motor de esta manera se controla la velocidad del motor. De igual manera se realizó la prueba con la resistencia eléctrica de cómo controlar la activación y desactivación de la misma.

Prueba mecánica: consiste en la prueba de equilibrio que debe de tener el plato donde irán colocados los matraces, el movimiento orbital tuvo una complicación es por ello que se modificó el movimiento a circular puesto que los puntos de equilibrio no bastaban para controlar el peso.

7. Resultados

Mediante la investigación se construyó el primer prototipo de la incubadora con agitación y giro circular con resultados favorables tanto de circulación del aire al interior (homogenización), así como en el control de la temperatura y velocidad deseada. El resultado del prototipo se finalizó con el giro circular por el conflicto del punto de equilibrio que existe, puesto que el contrapeso no fue suficiente.

Mediante el software de LabView se observó el comportamiento de la temperatura mediante unas graficadas la cual representaban el punto máximo, estable y punto máximo de cómo se comportaba la temperatura.

Tomando en cuenta el rendimiento de equipos de laboratorio se realizó el control y la calibración del controlador de temperatura que se asemeja a los quipos comerciales, como es el caso de una incubadora con agitación el cual tiene parámetros similares de funcionamiento como lo es la temperatura y la velocidad, de igual manera se controló la posición de los sensores TH22.

Se ha desarrollado una interfaz hombre – máquina en la cual el usuario debe ingresar los parámetros deseados tales como: temperatura, velocidad y tiempo de funcionamiento.

8. Conclusiones

Se diseñó y construyó un prototipo de incubadora con agitación circular puesto que al realizar las pruebas correspondientes con el giro orbital el punto de equilibrio del plato no fue suficiente, se estructuro mediante un código abierto y de bajo costo. Un equipo comercial con control de temperatura y movimiento orbital tiene un costo elevado de aproximadamente $300,000.000 en el mercado actual, en el desarrollo de nuestro equipo se gastó alrededor de $5,100. Dicho ahorro proviene de dos fuentes: el uso de un microprocesador de Arduino Mega para el control del proceso de temperatura y visualización de las RPM. Segundo, el ahorro inherente que viene con la construcción de partes de madera y acrílico.

El sistema térmico es diseñado por convección forzada. Consiste principalmente por una cámara de calentamiento donde se encuentra la niquelina.

El trabajo realizado pretende contribuir al desarrollo de investigaciones dentro de la Universidad, el control de la incubadora se encuentra en fase de pruebas ya que el área de Biotecnología brinda información de sus requerimientos para la realización correcta del dispositivo, en las pruebas se monitorean variables de temperatura principalmente obteniendo información relevante que ayude a

controlar el proceso y a su vez para generar la interfaz del usuario. Las incubadoras de laboratorio se utilizan mucho para aplicaciones de biología, como, por ejemplo: para cultivos celulares y de tejidos, para estudios farmacéuticos y hematológicos, para estudios bioquímicos, procesamiento de alimentos y aireación de células, también se utiliza para estudios animales, estudios de solubilidad y de fermentación como también para cultivos de bacterias. El prototipo contribuye al desarrollo de la ciencia e investigación.

9. Viabilidad técnica

Un técnico superior universitario del área de mecatrónica área automatización cuenta con conocimientos que permiten diseñar, construir, automatizar, modernizar, implementar y dar mantenimiento a sistemas o procesos a través de la integración sinérgica de elementos mecánicos, electrónicos, de control y sistemas de software.

Al tener las bases de conocimiento antes propuestas se plantea que el proyecto de una incubadora sobre el cual se está trabajando contara con los requerimientos que se solicitan para llevar a cabo el funcionamiento correcto del mismo.

7.1 Factores de diseño.

Funcionalidad

- Control de temperatura
- Aislamiento
- Monitorización

7.2 Diseño

- Creación de un CAD

7.3 Factores de procesos

Insumos

- Motor
- Estructura
- Componentes electrónicos
- Sensores

Pruebas de ensayo

10. Requerimiento de recursos humanos y materiales

10.1 Diseñador

Formula y proyecta los objetos, es capaz de ubicarse y responder con precisión ante solicitudes de variada índole.

10.2 Instrumentista

Se requiere de un instrumentador para instalar, probar, diagnosticar y calibrar el equipamiento utilizado para el control y medición de variables, así como la elaboración de planos de instrumentación, documentar los diagramas eléctricos, electrónicos y mecánicos.

10.3 Programador

Elaborar, desarrollar, ensayar y mantener en buen estado los soportes lógicos y/o los programas informáticos, para cubrir las necesidades de los usuarios.

10.4 Ensamblado y Construcción

Ensambla, monta, modifica o mantiene componentes de equipos eléctricos, electromecánicos y electrónicos, como sistemas de grabación de imágenes y sonido y sistemas de comunicación, e instala y repara instalaciones eléctricas y sus cables, de acuerdo con las normas y prescripciones técnicas pertinentes.

11. Viabilidad financiera

Para recalcar la viabilidad de este proyecto se realizó una comparación entre una incubadora comercial contra una incubadora ensamblada**.**

11.1 Ventajas de una incubadora comercial

- Existen varios modelos de incubadora que se ajustan a varios requerimientos para laboratorio con cuerpo metálico que van desde 36 Lts hasta 110 Lts de capacidad.
- También pueden fabricar incubadoras para ofrecerle soluciones hechas a medida, según sus necesidades y requerimientos.
- Nuestra incubadora para laboratorio es diseñada pensando en la facilidad de manejo, precisión y seguridad. Por lo cual, requiere mínimo de mantenimiento por parte del usuario.

11.2 Desventajas

- Las incubadoras comerciales son bastante caras con un precio de 300 mil pesos mexicanos.
- Cuando se descompone algún instrumento de la incubadora es bastante difícil encontrar repuesto de la pieza.
- Los repuestos de las incubadoras son bastante caros.
- Una incubadora sencilla que solo se requiere para enzimas es bastante cara.
- Son bastante económicas en comparación de una incubadora comercial su precio viene siendo de unos 100 mil pesos mexicanos.
- Utiliza materiales bastante eficientes para su construcción.
- Si se descompone de algún instrumento ´puede mandarla a reparar con los fabricantes.
- Las piezas de la incubadora independiente son bastante fáciles de conseguir y baratas.
- Tienen la misma capacidad que una incubadora comercial a la hora de producir enzimas.

11.3 Desventajas de la incubadora independiente

- No tienen tanto espacio para almacenar enzimas
- La incubadora es algo pequeña

Ingeniería y Automatización, Capítulo 33, pp. 414 - 423.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

12. Viabilidad social

12.1 Encuestas de impacto o aceptación del proyecto:

Participe en nuestra encuesta

Cordial saludo.

Estimada comunidad el siguiente cuestionario tiene como objetivo determinar el nivel de aceptación del proyecto que tiene por nombre Incubadora con giro orbital con el fin de capacitar en el manejo de herramientas.

El presente es un cuestionario que contiene preguntas de selección múltiple cada una con sus respuestas correspondientes, donde deberán justificar algunas de ella. Les solicitamos responder con la mayor sinceridad, debido a que sus repuestas dependen los buenos resultados de este proyecto formativo.

Elija una sola respuesta de las preguntas que apareen a continuación, remarcando el recuadro.

1. ¿Considera importante el desarrollo del proyecto de la incubadora?

☐ Si

☐ No

☐ Justifica tu respuesta _____

2. Desarrollar un proyecto como la incubadora es:

☐ Necesario

☐ Innecesario

☐ Justifica tu respuesta _____

3. ¿Qué tan importante consideras el proyecto de la incubadora?

☐ Muy importante

☐ Poco importante

☐ No es importante

4. ¿Usted participaría en el proyecto?

☐ Si

☐ No

5. ¿Sabes cuál es l función del proyecto de la incubadora?

☐ Si

☐ No

6. Si la respuesta anterior es afirmativa.

☐ Justifica tu respuesta _____

7. ¿Conoces algún proyecto que se haya implementado en la institución anterior a este?

☐ Si

☐ No

☐ Cual _____

8. ¿Cuáles herramientas necesita aplicar en el desarrollo de actividades cotidianas del programa al cual usted pertenece?

☐ Utilizar el procesador de textos y hojas de cálculo para la recoolección de datos.

☐ Crear un blog donde se publiquen las actividades y novedades por dependencia.

☐ Consolidar una base de datos por cada dependencia.

☐ Diseñar y publicar una página web.

☐ Ninguna de las anteriores.

☐ Otra ¿Cuál? _____

9. ¿Nos darás sugerencias para el mejoramiento del proyecto?

☐ Si

☐ No

☐ ¿Cuales? _____

10. Anote sus datos por favor:
Fecha: _____
Nombre completo: _____
Ocupación: _____
Programa de Formación: _____

Les agradecemos por su tiempo empelado para responder este cuestionario. La información dada es muy confidencial y será utilizada para mejorar el proyecto de la incubadora.

Que tenga un buen día les desea:

Alumnos de la carrera de Mecatrónica área automatización 7 A
Universidad Tecnológica de Xicotepec de Juárez.

Referencias

[1] J.M. Pearce, *Open-Source Lab: How to Build Your Own Hardware and Reduce Research Cost*, Elsevier. 2013

[2] T. Oviedo y M. Fajardo, *Diseño y construcción de una cámara simular condiciones ambientales a 7000m.s.n.m*, Rev. Politécnica. 31(1), 2012, 46-51.

Ingeniería y Automatización, Capítulo 34, pp. 424 - 433.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Desarrollo de un Escáner 3D Basado en Visión Estereoscópica

Cornejo Guerra Paulo Sergio[1,2], Traslosheros Michel Alberto[2], Chagolla Gaona Hernando[1], Castellanos Galindo Joaquín[1] y Martínez Garrido David[1].

[1]Departamento de Ingeniería Eléctrica y Electrónica
Tecnológico Nacional de México/Instituto Tecnológico de Querétaro.
Av. Tecnológico S/N, Centro Histórico, Santiago de Querétaro, Querétaro, México, Tel: (442) 2274400
e- mail: paulo_guerra2000@yahoo.com, hchagollag@hotmail.com, jcastell_7700@yahoo.com.
[2]Universidad Aeronáutica en Querétaro.
Carretera Estatal Querétaro-Tequisquiapan 22154, Colón. Querétaro, México, Tel: (442) 1016600
e-mail: alberto.trasloshero@unaq.edu.mx

Resumen

Se describe el desarrollo y diseño de un prototipo de escáner 3D basado en un sistema de visión estereoscópica el cual será utilizado como equipo didáctico y de desarrollo ya que los modelos obtenidos con su uso podrán ser maquinados o fabricados. El sistema está constituido por un par de cámaras digitales, un emisor de puntos láser los cuales se inciden en el objeto a estudiar, las cámaras capturan pares de imágenes las cuales son procesadas digitalmente para extraer características o puntos de interés a partir de los puntos láser los cuales mediante triangulación se obtienen sus coordenadas tridimensionales logrando con ello una nube de puntos la cual al ser grabada en un archivo de extensión adecuada puede ser visualizada en plataformas como MeshLab, SolidWorks o Catia. El programa fue desarrollado utilizando la librería OpenCV.

Palabras clave: visión estereoscópica, puntos de interés, triangulación, correspondencia estéreo, estéreo rectificación, nube de puntos, procesamiento digital, disparidad, reconstrucción tridimensional.
.

1. Introducción

La visión artificial es una de las ramas de la ciencia y la tecnología que ha permitido el desarrollo de otras disciplinas como la robótica, inteligencia artificial, instrumentación espacial, equipo militar y sistemas industriales; específicamente un tópico de esta disciplina es la reconstrucción tridimensional la cual es esencial en áreas como la arquitectura, la realidad virtual la medicina y a escultura exige el diseño de sistemas más eficientes y pequeños que permitan obtener modelos tridimensionales cuyo error en la exactitud en comparación con el modelo real sea casi nulo [1-3]. El dispositivo capaz de obtener modelos tridimensionales mediante visión artificial se llama escáner 3D; un patrón de luz se hace incidir en el objeto a reconstruir cuya deformación al ser proyectado permite analizar digitalmente la geometría de la superficie del objeto, esa información se obtiene a partir de imágenes las cuales son capturadas mediante cámaras las cuales son principalmente de tipo digital. Un escáner está formado por una o más cámaras, cuando son dos se dice que el escáner es estereoscópico o está basado en un sistema de visión estereoscópica [1]. La visión estereoscópica es la forma en como los mamíferos superiores incluyendo los seres humanos ven, a partir de los ojos se obtienen imágenes del mundo exterior. La separación entre ellos hace que lo visto con el ojo derecho está ligeramente desplazado con respecto a lo visto con el izquierdo, esta diferencia en ambas imágenes se llama disparidad y permite determinar la profundidad o la distancia a la que se encuentran los cuerpos en el espacio.

Un sistema de visión estereoscópico funciona similarmente [1], con dos imágenes capturadas en el mismo instante por medio de dos cámaras separadas entre sí por una distancia dada es posible determinar la profundidad de los objetos que se encuentran los objetos en el espacio real. Esto significa que con dos imágenes, las cuales son bidimensionales (2D) se obtiene una representación

en tridimensional 3D de entorno. Esta técnica se llama reconstrucción tridimensional 3D. Una de los métodos fundamentales que permiten realizar la reconstrucción 3D se llama triangulación [5] con el cual se obtienen las coordenadas tridimensionales de un punto común en ambas imágenes a partir de la traslación (separación) entre las cámaras y su rotación; las variables que representan las características mencionadas se les conoce como parámetros intrínsecos y extrínsecos del sistema estereoscópico los cuales deben calcularse para hacer una reconstrucción correcta.

2. Sistema Estereoscópico. Análisis y Diseño

El proyecto consiste en el diseño de un escáner 3D basado en visión estereoscópica utilizando dos cámaras digitales el cual servirá como equipo didáctico y desarrollo en las áreas de manufactura de la UNAQ. A continuación se describen los elementos que lo constituyen.

a). Par de cámaras Logitech c920 con una resolución de 15 Mega pixeles. Las cuales están montadas sobre una regleta ajustable para controlar la posición y distancia de las cámaras entre sí. A su vez la regleta es colocada sobre un tripie el cual permite que el sistema estereoscópico pueda moverse verticalmente y horizontalmente.

b). Apuntador laser (fuente de luz estructurada). Dispositivo montado sobre una base que lo mantiene en posición fija. El láser tiene un cabezal que permite que el haz se divida en pequeños haces los cuales, al incidir en un objeto, se ven muchos puntos a diferencia de uno solo; esta característica es importante para obtener la nube de puntos como se explicara 2.6.

c).- Escenario. En él se coloca el objeto a escanear es de color blanco para evitar el ruido en las imágenes

d).- Computador. Permite la ejecución del programa diseñado para la reconstrucción 3D del objeto o material de trabajo así como la acción de control del movimiento de la base sobre la cual se coloca el objeto a escanear si la reconstrucción. En la figura 1 se muestra el sistema diseñado.

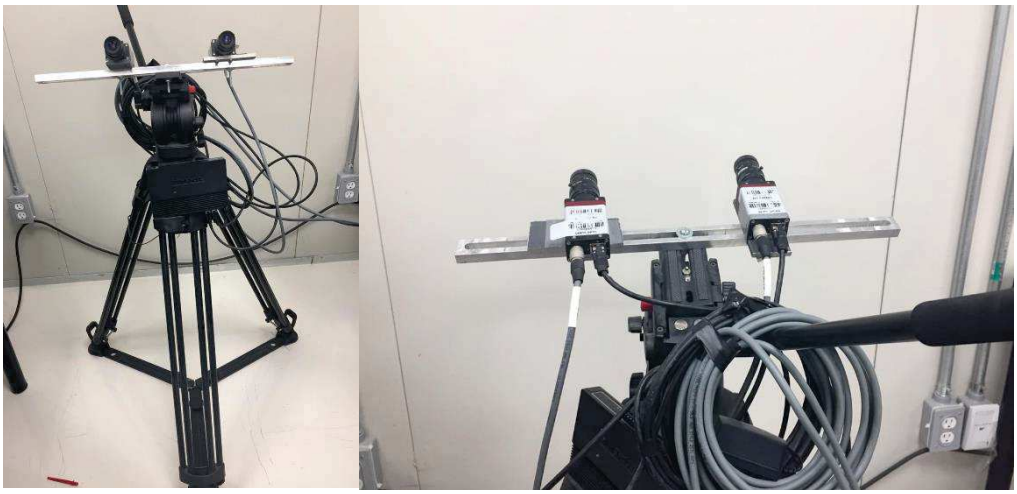


Figura 1. Sistema estereoscópico diseñado.

2.1. Etapas del proceso de captura, procesamiento y reconstrucción 3D

La forma de implementar la visión artificial es mediante la captura de pares de imágenes simultáneas una para cada una de las dos cámaras del sistema. Posteriormente se realizan etapas de acondicionamiento, extracción de características y procesamiento digital. Estas acciones se realizan mediante el diseño de software mediante un lenguaje de programación elegido, para el proyecto se decidió utilizar OpenCV que es un conjunto de librerías de uso libre de visión artificial, que es multiplataforma y basadas en C++, para la visualización del modelo tridimensional obtenido se utilizó MeshLab debido a que también es de uso libre, cabe mencionar que el programa desarrollado graba los resultados en archivos de diferente formato incluyendo aquellos que son visualizados en otros software como Solidworks o Catia. La figura 2 muestra un diagrama de flujo de las diferentes etapas que el programa desarrollado realiza los cuales que serán descritos.

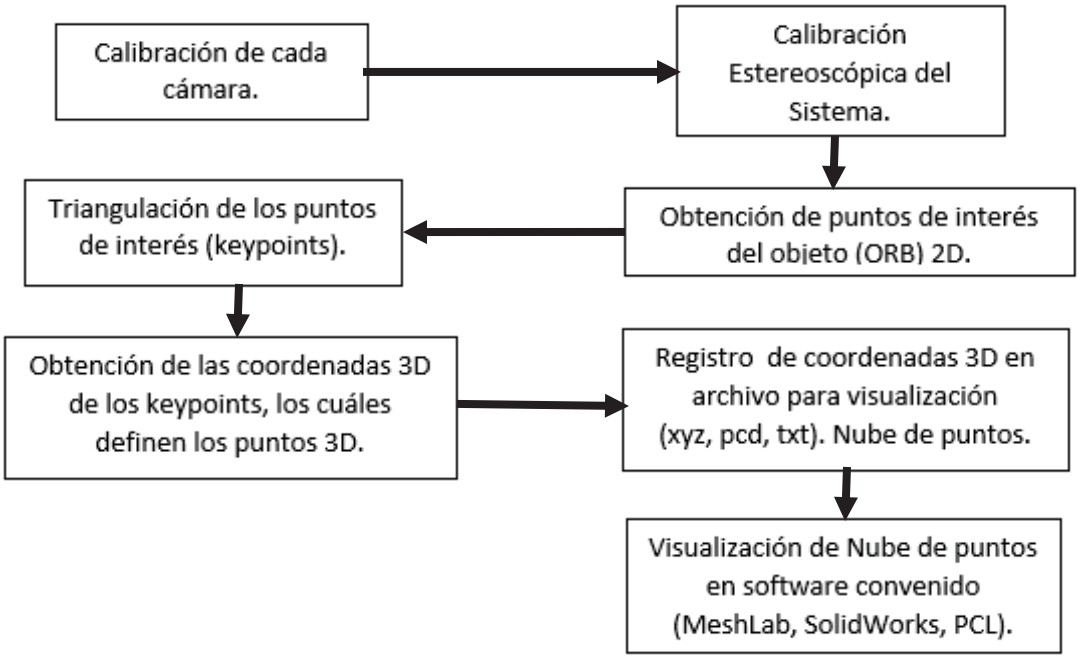


Figura 2. Diagrama de flujo en el cual se muestran los pasos realizados en el programa diseñado.

2.2 Calibración de las cámaras

La transformación de la luz en impulsos eléctricos se realiza en las cámaras digitales, estas tienen una matriz de sensores CCD (Charged Coupled Device) y cuyo tamaño determina su resolución como se muestra en la figura 3.

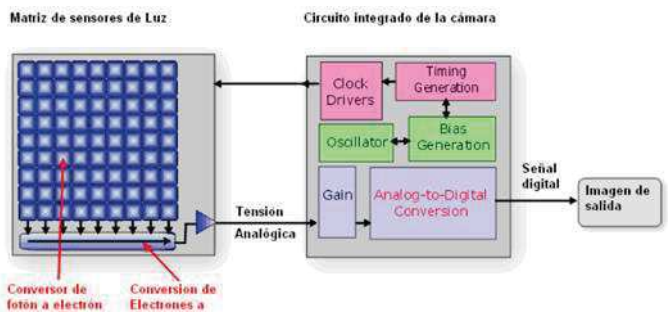


Figura 3. Disposición del sensor CCD en una cámara digital.

Ingeniería y Automatización, Capítulo 34, pp. 424 - 433.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

La unidad fundamental en las imágenes obtenidas por cámaras digitales se llama pixel. Si bien las cámaras digitales utilizan alta tecnología su funcionamiento es descrito con un modelo muy sencillo llamado modelo *pinole* en el cual el lente de la cámara actúa como un punto por el cual pasan los rayos de luz de la escena sin ser alterados en su dirección. Mediante este principio a un punto en el espacio 3D A(X,Y,Z) le corresponde a un punto del plano imagen de la cámara O(x,y) teniendo como resultado una transformación lineal de $R^3 \rightarrow R^2$ es decir del espacio 3D al plano 2D como se muestra en la figura 4.

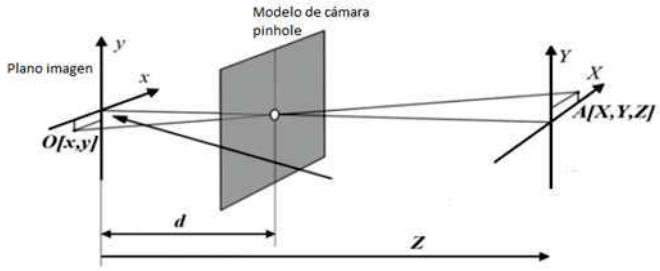


Figura 4. Modelo pinole para las cámaras utilizadas.

Las cámaras digitales pueden presentar defectos de fabricación sobre todo en los lentes, aún cuando sean de la misma marca y modelo es común que muestren diferencia en las imágenes que se pueden capturar con ellas ya que esas imperfecciones provocan distorsión (radial y tangencial) en las imágenes y que afectan su calidad, esta limitación puede producir modelos erróneos. Otros de los datos que son necesarios conocer, aparte de la distorsión, son las coordenadas del punto principal y la distancia focal que como se mencionó anteriormente son llamados parámetros intrínsecos ya que son propios de cada cámara. De igual forma es necesario conocer la posición (translación) o separación entre ambas cámaras y la orientación (rotación) de cada una con respecto al objeto en el espacio real, estos se llaman parámetros extrínsecos del sistema. Estos datos o valores importantes se obtienen mediante la calibración y estéreo calibración respectivamente. Existen varias técnicas para realizarlas:
:
a) Métodos basados en modelos 3D [Tsai 1987].
b) Métodos basados en patrones 2D [Zhang 2000]. Es menos precisa que l anterior.
c) Autocalibración [Harley 1994]. El planteamiento matemático es complejo.

En este trabajo se utilizó en método de Zhang [6], ya que está implementado en OpenCV, para realizar la calibración se realiza la captura de un número adecuado de pares de imágenes de un patrón de calibración que es un tablero de ajedrez, se recomienda un total de 15 o 20 pares de imágenes optando por esta última cantidad y en las cuales el patrón debe presentar posiciones diferentes para mejorar el cálculo de los parámetros en la figura 5 se muestra un par de imágenes del patrón utilizado.

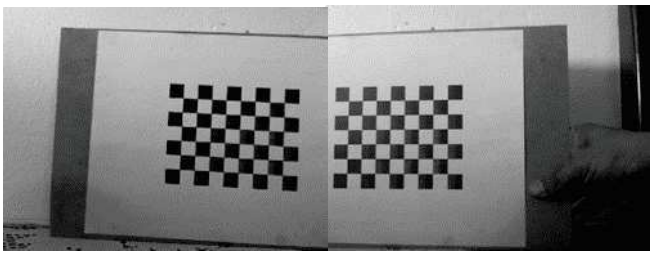


Figura 5. Ejemplo de par de imágenes estéreo capturadas para la calibración de las cámaras.

como se puede ver las imágenes están en escala de grises, es decir la asignación de un valor a cada pixel en color a un nivel digital en el rango 0-255 que corresponde al intervalo blanco-negro, la escala de grises facilita la implementación del procesamiento digital de las imágenes ya que son de menos bits (8 bits) que las imágenes en color las cuales se basan en 3 matrices (una para cada color primario) con ellas las funciones de suavizado, filtrado, segmentación y extracción de características es más eficiente.

A modo de comparación la calibración también se realizó en el Matlab el cual tiene un toolbox libre llamado Camera Calibration Toolbox y se concluyó que OpenCV es más rápido sin embargo Matlab tiene funciones gráficas que permiten realizar un análisis más minucioso del proceso. En ambos casos los resultados fueron adecuados dado el valor de error obtenido (error ≈ 0.2). Los parámetros intrínsecos y extrínsecos son desplegados en el computador ya sea en la ventana de trabajo de Matlab o en el símbolo de sistema en el caso de OpenCV para ello es necesario obtener el ejecutable del programa en un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés) en este caso se utilizó Visual Studio 2010. De igual forma los datos son registrados en archivos para su uso en las posteriores etapas del programa diseñado, en OpenCV los datos son grabados en archivos cuya extensión es .yml, los datos almacenados son visibles si se abren con WordPad y se despliegan en matrices y vectores, las más importante es una matriz A de dimensión 3x3 llamada matriz de parámetros intrínsecos la cual es descrita en la ecuación (1).

$$A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} \tag{1}$$

En (1) f_x y f_y son las distancias focales con respecto a los eje x y y, c_x y c_y son las coordenadas del punto principal, los valores r_{ij} y t_i para $i = j = 1,2,3$ en (2) son los parámetros de rotación de las cámaras con respecto a los ejes X, Y, Z y las componentes de translación del origen del sistema de coordenadas del objeto en el espacio al sistema de coordenadas de la cámara.

2.3 Estéreo calibración del sistema

Una vez que los parámetros intrínsecos han sido calculados se debe determinar la separación entre las cámaras así y su posición, mediante un proceso que recibe el nombre de estéreo calibración y consiste en obtener los parámetros extrínsecos del sistema. En OpenCV la estéreo calibración se realiza con el programa *stereocalib* [6-8], una vez que se hace ejecutable el programa solo hay que adicionar como argumento el nombre del archivo de extensión .txt donde se encuentra el listado de los pares de imágenes del patrón de calibración que también fue utilizado en la calibración individual, este archivo debe estar en la misma dirección donde se encuentra el ejecutable del programa. El resultado de la estéreo calibración es un archivo de extensión .yml donde se registran los parámetros extrínsecos del sistema en dos matrices una T de dimensión 3x1 llamada matriz de translación la cual contiene los datos referentes a la separación entre las cámaras y otra R de dimensión 3x3 llamada de rotación cuyos datos se refieren a la posición de las cámaras con respecto al sistema. Ambas matrices determinan una matriz 3x4 la cual se llama matriz de parámetros extrínsecos y que se muestra en la ecuación (2).

$$[R \;\; T] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \tag{2}$$

Esta matriz relaciona un punto (X, Y) en la imagen (2D) con su correspondiente punto (x, y, z) en el espacio real (3D) mediante la relación representada en la ecuación 3.

$$\lambda \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = A[\, R \; T \,] \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3}$$

Como se verá más adelante los puntos 2D reciben el nombre de puntos de interés o keypoints y deben estar en ambas imágenes para calcular sus coordenadas 3D de este modo obtendremos la nube de puntos que será la reconstrucción 3D del objeto. Como se mencionó, los puntos de interés deben estar presentes en un mismo par de imágenes esta propiedad se llama correspondencia estéreo, la búsqueda de estos puntos (búsqueda de correspondencias) es en general compleja dependiendo las características del par estéreo de imágenes y del objeto en el par que se desea modelar, en nuestro caso, la luz láser incidente en el objeto a escanear. Existen dos tipos de errores fundamentales en el proceso de búsqueda de correspondencias:

-Falsos positivos: Cuando se obtienen dos puntos cuya correspondencia es errónea.
-Falsos negativos: Cuando no se elige un punto de interés que es visible en el par de imágenes.

El primer caso es aceptable ya que es posible rechazarlos mediante la implementación de un algoritmo adecuado en el programa del sistema siendo esto una de las funciones principales de todo sistema de reconstrucción 3D. Una forma para reducir el número de falsos negativos es un procedimiento llamado rectificación estéreo que se explicará en 2.5.

En la figura 5 se muestra el resultado de la estéreo calibración del sistema realizada con Matlab solo para motivos de demostración.

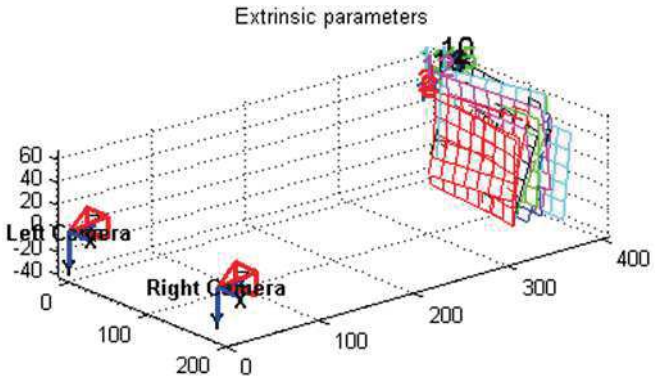


Figura 6. Resultado de la estéreo calibración del sistema.

2.4. Captura de pares de estéreo imágenes del objeto

El objeto o material que se reconstruirá tridimensionalmente debe colocarse en la base del escenario del escáner de modo que su posición se encuentre en el eje del sistema de visión y pueda ser captado por ambas cámaras. La iluminación debe ser adecuada (no muy intensa) para evitar el ruido y limitar la cantidad de falsos positivos y negativos presentes en las imágenes. Se decidió escanear 3 objetos para evaluar la capacidad del sistema en relación a captar diversos materiales. El haz láser multipunto es de color verde ya que mostró una mayor eficiencia que el rojo ya que es más visible para el ojo humano y se hace incidir sobre los objetos cuando se capturan el par de las imágenes mediante video, es decir el procesamiento de las imágenes es en tiempo real, cada imagen se pasa a escala de grises para su posterior suavizado (filtrado) ya que los puntos laser puede producir ruido al incidir en la superficie del material, la búsqueda de correspondencias es una de las partes más importantes del programa ya que se utilizaron varios procesos los cuales serán descritos en el apartado 2.6.

2.5. Rectificación estéreo

La posición del sistema estéreo debe ser tal que los ejes ópticos de las cámaras sean paralelos para lograr minimizar la existencia de falsos positivos, las cámaras están montadas sobre bases atornilladas las cuales asegura que permanezcan fijas y esa posición, sin embargo, generalmente esto no garantiza que los pares estéreo de imágenes estén alineados lo que complicaría la búsqueda de correspondencias. El problema se resuelve mediante la estéreo rectificación de cada par permitiendo que los puntos de interés (pixeles) en cada imagen del par estéreo estén proyectados en la misma fila mediante el cálculo de sus nuevas coordenadas, de este modo su búsqueda se reduce a solamente a las columnas en el plano de las imágenes. OpenCV tiene funciones que permiten realizar la rectificación de las imágenes mediante un algoritmo basado en el método de Bouguet [8] cuya finalidad consiste en determinar una matriz de rectificación R_{rect} con la cual se realiza la alineación de cada par de imágenes está definida definida por la relación (4):

$$R_{rect} = \begin{bmatrix} {\theta_1}^T \\ {\theta_2}^T \\ {\theta_3}^T \end{bmatrix} \tag{4}$$

Las componentes de esa matriz columna (vector) son vectores unitarios que son expresados por las fórmulas (5)

$$\theta_1 = \frac{T}{\|T\|}, \theta_2 = \frac{[-T_y T_x 0]^T}{\sqrt{T_x^2 + T_y^2}}, \theta_3 = \theta_1 x \theta_2 \tag{5}$$

Y son perpendiculares u ortogonales entre sí. Al realizar la calibración se calculan las matrices de rotación para cada cámara R_D (derecha) y R_I (izquierda), derecha e izquierda respectivamente, dadas por las fórmulas (6) y (7).

$$R_D = R_{rect} r_D \tag{6}$$

$$R_I = R_{rect} r_I \tag{7}$$

En OpenCV lo anterior se realizó al implementar la función “cvStereoRectify()”, por medio de la función “cvInitUndistortRectifyMap()” se realiza el mapeo de los pixeles de los pares estero a pares estéreo rectificados.

2.6. Correspondencia estéreo

Como se mencionó en 2.3 el problema de correspondencia es tal vez el proceso más complicado y que más costo computacional requiere en la reconstrucción 3D del modelo. Es necesario determinar la mayor cantidad de puntos de interés los cuales se conocen también como keypoints. En esta parte el programa se basó en el algoritmo ORB (Oriented FAST and Rotated BRIEF) el cual es un descriptor de características, es decir, un vector que contiene la ubicación de los keypoints y se basa en los métodos SIFT y SURF [9]. SIFT es un algoritmo de detección de keypoints el cual no varía a la escala o rotación del sistema, es decir, no importa si el sistema se mueve, se aleja, se acerca o gira siempre relacionará a los puntos característicos en una imagen con sus correspondientes en la otra, el método se basa en la implementación de la distribución normal o de Gauss de forma sucesiva en un par de imágenes para determinar los keypoints. SURF (Speeded Up Robust Features) es un algoritmo que tiene la misma finalidad que SIFT sin embargo los cálculos son más rápidos ya que en este última la implementación de la distribución gaussiana es reiterada lo cual hace que el proceso sea más lento, por otro lado, SURF se basa en la implementación de la matriz hessiana en las imágenes con la cual se establecen regiones donde se localizan los keypoints siendo mucho más rápido el procedimiento pero menos robusto. De este modo se puede decir que con SIFT la búsqueda de keypoints es más precisa pero más lenta mientras que con SURF el proceso es más rápido pero menos robusto. Ambos métodos se basan en los trabajos de David G. Lowe [10] y están patentados por lo tanto no se permite

su uso comercial si bien si puede utilizarse para fines de enseñanza como es el caso de este proyecto.

ORB se basa en considerar los cambios en las intensidades de los pixeles siendo mucho más rápido que los algoritmos mencionados, los puntos láser al ser incididos en el objeto son candidatos a ser considerados keypoints, en la figura 7 se muestra la captura de un par de imágenes de los tres objetos que se utilizaron como ejemplo en este trabajo. Los puntos láser se muestran como círculos en distintos colores y sólo aquellos que el algoritmo ORB detecta como keypoints son unidos con una línea recta.

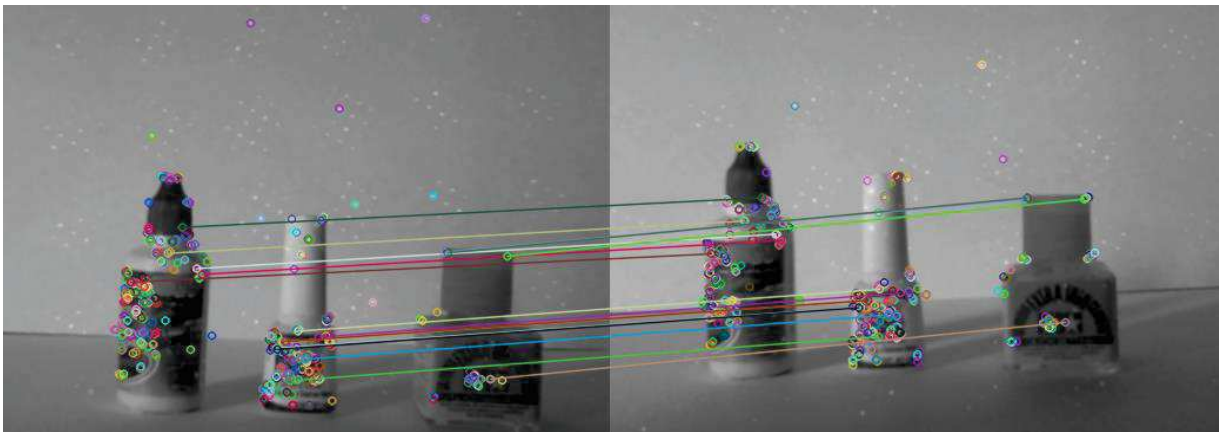


Figura 7. Determinación de keypoints o puntos correspondientes en un par de imágenes con el algoritmo basado en ORB.

2.7. Triangulación

Una vez que se han determinado los keypoints se determinan sus coordenadas en las imágenes, ese proceso es realizado en un procedimiento dentro del programa las cuales se utilizan para determinar sus coordenadas en el espacio tridimensional, todo el conjunto de las coordenadas tridimensionales obtenidas determinan la nube de puntos que es la base de la reconstrucción tridimensional del objeto. La triangulación se basa en el algoritmo de Hartley-Zisserman [7] basado en las ecuaciones 8 y 9

El siguiente paso es calcular las matrices de proyección de cada cámara para transformar los puntos del espacio tridimensional en puntos de sus planos imagen las cuales corresponden a las relaciones (8) y (9).

$$P_D = \begin{bmatrix} f & 0 & c_{xD} & T_x * f \\ 0 & f & c_{yD} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{8}$$

$$P_I = \begin{bmatrix} f & 0 & c_{xI} & 0 \\ 0 & f & c_{yI} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{9}$$

Donde la componente T_x es la distancia o desplazamiento horizontal entre las cámaras, f la distancia focal de las cámaras que hemos supuesto iguales y $c_{xD}, c_{yD}, c_{xI}, c_{yI}$, son las coordenadas de los puntos principales de las cámaras derecha e izquierda respectivamente.

3. Resultados

En la figura 8 se muestra el modelo 3D de los 3 objetos visualizados en MeshLab, al fondo se ve el panel del escenario el cual fue barrido también con los puntos láser: La nube de puntos corresponde a la toma frontal con las cámaras por ello se observan incompletas, se está trabajando para mejorar el

programa que al girar, ya sea el sistema o el objeto, se logre una nube de puntos del espacio total. Una etapa posterior del proyecto consiste en el mallado la nube de puntos para obtener un modelo 3D sólido del objeto. Se compararon los tamaños reales de los objetos con el modelo y son muy similares, una vez que el mallado se implemente se podrá obtener con mayor precisión un modelo para estimar el porcentaje de error.

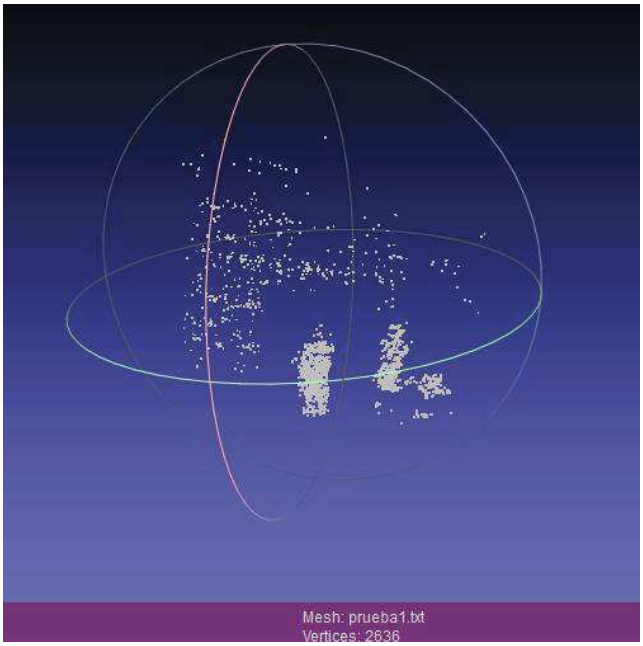


Figura 8. Modelo 3D del objeto visualizado con MeshLab.

4. Conclusiones

El método basado en ORB demostró ser método muy eficiente y rápido frente a las métodos SIFT y SURF los cuales fueron más lentos, al utilizarlos las imágenes en video se "congelaban" y con ello los cálculos tenían más errores. Anteriormente la luz estructurada utilizada fue una franja láser los resultados fueron convenientes pero el tiempo de escaneo y el cálculo de las coordenadas 3D del modelo fueron mucho más lentas.

Como trabajo futuro se pretende aumentar las capacidades del programa para que la visualización de la nube de puntos sea en tiempo real, se ha comenzado a investigar con PCL (Librería de Nube de Puntos por sus siglas en inglés) en cual es libre para tal meta.

5. Referencias

[1] David 3D scanner, [en línea], disponible: http://www.david-3d.com/en. Sitio visitado 2014.
[2] C. Davatzikos, "*Spatial Transformation and Registration of Brain Images Using Elkastically Deformable Model*s". Science Direct, May 1997, volume 66, Issue 2 pp. 207-222.
[3] R. Usamentiaga, J. Molleda, D. F. García. " *Fast and Robust Laser stripe extraction for 3D reconstruction in industrial enviroments, Machine Vision and Applications"*, January 2012, Volume 23 Issue 1 pp 179-196.
[4] Siggraph 2009 Asia 2009 Courses [en línea] Disponible: http://mesh.brown.edu/byo3d/

[5] "*Simple triangulation with OpenCV Harley & Zisserman*", More Than Technichal [en línea], disponible: http://www.morethantechnical.com/2012/01/04/simple-triangulation-with-opencv-from-harley-zisserman-w-code/
[6] OpenCv Camera Calibration and 3D reconstruction [en línea],disponible: http://docs.opencv.org/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html sitio visitado, 2017.
[7] Hartley R. y Zisserman A. "*Multiple View Geometry in Computer Vision*" Cambridge University Press. Second Edition 2003 pp. 405-458.
[8] Bradski G. y Kaehler A. "*Learning OpenCV: Computer Vision with the OpenCV library*". O´Reilly. First Edition September 2008. Pp. 378-458.
[9] Brahmbhatt S. "*Practical OpenCV*". Technology in action pp. 95-119.

[10] Holzer S., Shotton J. y Kohli P. "*Learning to Efficiently Reapetable Interest Points in Depth Data*" 12th European Conference on Computer Vision. Firenze, Italy, 2012. Poster Session 1.

Ingeniería y Automatización, Capítulo 35, pp. 434 - 442.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Diseño e Implementación de un Sistema de Control para Pruebas de Soldadura de Plasma

Salas-Llanes Luis Arturo, Ramírez-Torres Flor, Cerón-Franco Aureliano, Montijo-Valenzuela Eliel Eduardo, Rodríguez Sánchez Saúl
Departamento de metal-mecánica, Tecnológico Nacional de México/Instituto Tecnológico de Hermosillo, Ave. Tecnológico y Periférico Poniente S/N C.P. 83170 Colonia Sahuaro, Hermosillo Sonora, México, Parque Industrial Hermosillo, C.P. 83299.

Resumen

Implementar un sistema de control en línea de procesos mediante la integración de diversos equipos programables controlados por una interfaz sencilla para el operador y de fácil acceso para futuras modificaciones según lo requiera el equipo a probar, brindando capacidad de análisis y manejo de resultados para el personal competente en la toma de decisiones.

Palabras clave: Labview, PLC Teco, MakerHub, Arduino, CutMaster, Burn In, Tester, Hipot,

1. Introducción

El proyecto presenta una innovación tecnológica en el proceso de producción con una actualización de la prueba incorporando hardware y software de control con el objetivo de brindar una mejora en la etapa de “Burn In” en la linea de producción de CutMaster. Con la incorporación de nuevos equipos electrónicos innovando el proceso actual mediante una integración con una nueva interfaz hacia el personal y un nuevo manejo de resultados. Logrando, por medio de un sistema de control para pruebas de los equipos de soldadura de la línea CutMaster de la empresa Victor technologies, un aumento en la calidad de sus equipos al poder detectar equipos defectuosos en línea evitando que estos lleguen a sus clientes

Desarrollar e implementar módulos de pruebas mediante el diseño y la investigación de métodos electrónicos capaces de controlar diferentes magnitudes para la verificación del buen funcionamiento de una maquina cortadora de plasma en la línea de CutMaster

- Selección de componentes electrónicos óptimos para el proceso.
- Selección de software, programación y comunicación bidireccional entre componentes.
- Lectura y análisis de señales en parámetros de operación.
- Interfaz visual para el personal de línea de producción.
- Análisis de información generada en el proceso de prueba.
- Elaboración de informe de los parámetros relevantes de la prueba.

2. Metodología

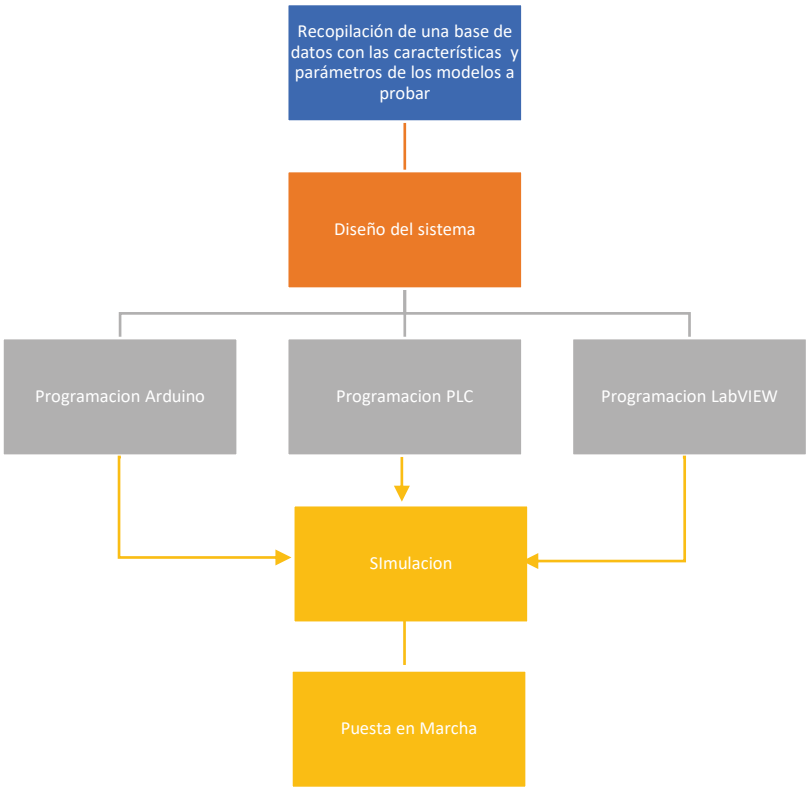


Figura 1. Diagrama de bloques del proceso

Las principales mejoras e innovaciones del siguiente proyecto están centradas en las etapas críticas y de mayor interés en el proceso, creando un mayor control en los estándares de calidad, modernización del equipo de pruebas, protección del equipo y personal de línea, adaptación para prueba en diversos equipos con sus parámetros específicos, verificación total en las etapas funcionales del equipo de corte de plasma en la línea de Cutmaster, capacidad de expansión y modificación para la incorporación de nuevos modelos en la línea de producción y capacidad de implementación en diversas líneas de producción, teniendo una base sólida y confiable en la etapa Burn In de los equipos fabricados.

Arduino es una plataforma de prototipos electrónicos de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Arduino puede interactuar con el entorno mediante la recepción de entradas desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos. El microcontrolador de la placa se programa usando el Arduino Programming Language (basado en Wiring) y el Arduino Development Environment (basado en Processing). Los proyectos de Arduino pueden ser autónomos o se pueden comunicar con software en ejecución en un ordenador (por ejemplo, con Flash, Processing, MaxMSP, entre otros). Las placas se pueden ensamblar a mano o encargarlas preensambladas; el software se puede descargar gratuitamente. Los diseños de referencia del hardware (archivos CAD) están disponibles bajo licencia open-source, por lo que eres libre de adaptarlas a tus necesidades.

El Controlador Lógico Programable marca Teco cuenta con una interfaz sencilla y software de licencia libre, con grandes ventajas al momento de programar ya que cuenta con un simulador integrado el cual permite comprobar el funcionamiento del programa. Además, soporta los lenguajes

de tipo escalera o bloques de funciones, brindando mayor facilidad al programador de elegir la sintaxis a convenir según sea el objetivo del programa.

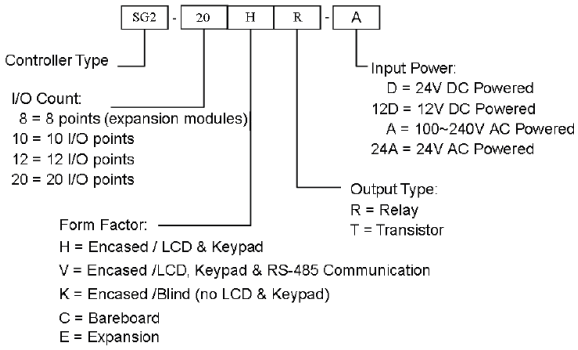


Figura 2. Numero de parte del PLC utilizado

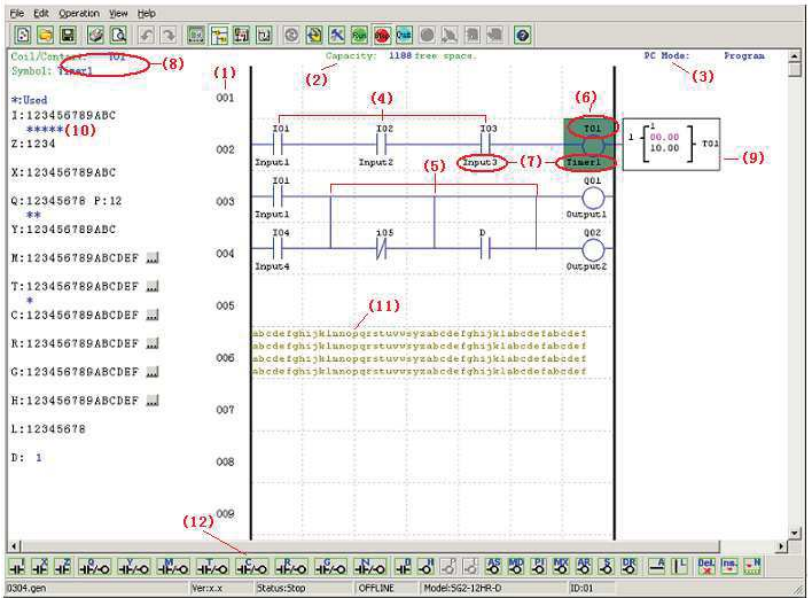


Figura 3. Interfaz de programación de PLC Teco

En el proyecto se utilizó programación escalera con instrucciones simples, como entradas, salidas y contacto auxiliar o banderas, cada uno de estos símbolos se les añade un número específico con el fin de distinguir distintos tipos de entradas, distintos tipos de salidas y distintos contactos de control. De esta manera I1, Q1, M1, representan contactos diferentes, de tal modo que cada uno de estos se pueden representar en diferentes estados según sea la necesidad en el programa. El estado de un contacto en programa escalera puede ser Normalmente abierto o Normalmente cerrado, mientras que las salidas son capaces de activar un contacto físico en el PLC, o cambiar el estado a Set o Reset de un contacto auxiliar en la programación.

También se utilizó el lenguaje de programación gráfica de National Instrument, LabVIEW. El cual provee de una mejor forma para solucionar los problemas que las alternativas tradicionales de bajo nivel y la prueba está en su facilidad de programación. Las claves diferenciadoras de la

programación en G son el código gráfico intuitivo que usted crea y las reglas de movimiento de los datos que gobiernan la ejecución que se combinan para ofrecer una experiencia de programación que expresa el pensamiento de los procesos de sus usuarios de forma más cercana que otros lenguajes. A pesar de que G es un lenguaje de alto nivel, usted puede lograr rendimientos comparables a los de los lenguajes como C gracias al compilador de LabVIEW.

3. Desarrollo

3.1 Diseño del sistema

El proyecto busca la modificación e innovación del proceso de prueba Burn In en los equipos CutMaster, para lo cual se tuvieron a disposición variedad de recursos tecnológicos, en base a equipos electrónicos y software, mediante los cuales se pudiera crear una integración capaz de satisfacer los parámetros establecidos en el proceso, de tal modo que se optó por una interfaz en labview para realizar toda la parte de control e interfaz hombre maquina (HMI), mientras que en un microcontrolador arduino se utilizó para la lectura de los sensores y el PLC para el manejo del alto voltaje.

3.2 Cotización y compra de materiales

En esta etapa se cotizaron varios dispositivos y materiales de diferentes marcas, posteriormente se seleccionaron los que se adaptaban mas a los requerimientos de la empresa y se procedió a la compra de estos. Para la etapa de acoplamiento se optó por un PLC y entre las diversas opciones estaban las marcas Allen Bradley, TECO y Mitsubishi. De las cuales se eligió la marca TECO con el modelo SG2-20HR-12D sobre las otras, ya que se presentaban diversos inconvenientes tal como en caso de tomar algún modelo Allen-Bradley el precio era mayor al permitido y a su vez era necesario adquirir la licencia del software lo cual era un gasto innecesario por parte de la empresa, por otro lado, la opción Mitsubishi se descartó por el poco conocimiento sobre estos PLC´s, las principales razones por las cuales se decidió por TECO SG2-20HR-12D fue que en la empresa ya se contaban con este modelo de PLC en otras áreas con resultados exitosos, contando con el conocimiento en el funcionamiento, programación y especificaciones, con un extra ya que el software es de licencia libre lo cual siempre es tomado en cuenta por parte de la empresa.

Debido a que la empresa ya contaba con el software NI Labview, fue un gran apoyo ya que se pudo facilitar enormemente la parte de la interfaz y control, a la vez que también estaba presente el dispositivo Arduino para la emisión y recepción de señales, así como el contactor para el voltaje de alimentación y cable para alta tensión. La requisición de material fue procesada por el departamento de compras de la planta y a su vez de anexaron diversos dispositivos extras tales como sensores de temperatura, diodos, Leds y transistores los cuales eran necesarios para el manejo de los diversos voltajes en las diferentes etapas.

3.3 Programa en NI Labview

La programación en NI Labview se llevó a cabo para cumplir la secuencia establecida por el diseñador y cumplir estándares del departamento de calidad, donde se determinó el número total de ciclos a realizar por la máquina y los tiempos entre las etapas; Quedando la secuencia de la manera siguiente:

Capacidad	Piloto	Transfer	Descanso

40%	10 Seg	35 Seg	25 Seg
80%	10 Seg	50 Seg	10 Seg

Tabla 1. Especificaciones de tiempos en cada etapa

El programa tiene establecido las rutinas según sea el caso, con las especificaciones en parámetros, tiempos, emisión y recepción de señales.

Para poder iniciar cualquier ciclo es necesario ingresar cierta información tal como el número de serie de la unidad a probar, verificación de las conexiones y funcionamiento del equipo en el encendido y comunicación con la interfaz. Ya una vez iniciada cualquiera de las rutinas, se captura la temperatura actual censada en el proceso y en la tabla mostrada en la parte central de la interfaz se van desplegando los datos generados tales como la temperatura, estado del transfer (On/Off), fecha y hora de la ejecución, así hasta terminar el ciclo y determinar si la maquina paso la prueba, anexando toda la información generada en un archivo de texto con nombre del número de serie de la unidad probada.

Figura 4. Interfaz hombre maquina diseñada

3.4 Programa PLC Teco SG2-20HR-12D

El programa realizado fue dedicado para el control y manejo de la alta potencia utilizada en la prueba, con la recepción de señales provenientes del programa en NI Labview mediante la interfaz Arduino, el PLC conmuta los contactos para la activación del banco de carga según sea el paso de la secuencia, con una resistencia de 8Ω en la etapa de piloto dejando circular la corriente nominal de la máquina que son alrededor de 20 amperes y 1Ω en la etapa de transfer donde la corriente máxima varía según el modelo de la máquina, que van desde 60 amperes hasta 120 amperes. Por esta razón el PLC fue utilizado para esta sección de la rutina, evitando problemas de comunicación causadas por el ruido generado por parte de la máquina y el banco de carga.

3.5 Ensamble

El ensamble se realizó de tal manera que se ahorrara espacio para poder incorporar el módulo de pruebas a la línea de producción sin mayor inconveniente de modo que se colocó en una de las caras de la protección, donde se puede observar el estado de los indicadores leds sin mayor inconveniente.

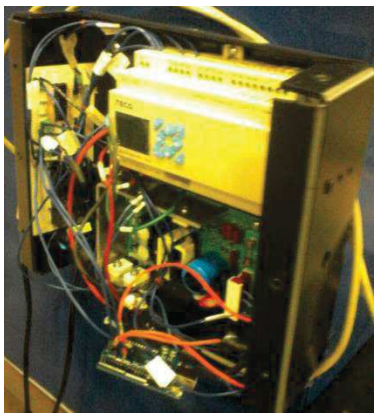


Figura 5. Ensamble en línea

3.6 Puesta en marcha

Ya una vez instalados los equipos en línea de tal manera que no se afectara al operador en la instalación y conexión de los equipos a probar se comenzaron con las pruebas, en las cuales se tomaron los datos en el consumo de corriente, voltajes utilizados, switches de seguridad y paro de emergencia, con la intención de corroborar el correcto funcionamiento de la prueba y la seguridad para el operador.

4. Resultados

Con una breve explicación a grandes rasgos del programa realizado en PLC Teco modelo SG2-20HR-12D, se verificó el estado y funcionamiento de todas las salidas y entradas presentes, los dispositivos periféricos y la respuesta del PLC, la comunicación con el programa de control NI Labview, de tal modo que se llevara a cabo la funcionalidad con especificaciones planteadas antes de la implementación en línea.

La primera etapa del prototipo se realizó simulando las entradas de los dispositivos periféricos en el PLC y poder comprobar el funcionamiento acorde a las especificaciones planteadas, realizando adecuaciones necesarias para mejora, tal como agregar una etapa extra de protección entre los dispositivos de control para evitar problemas en caso de una perturbación externa por los campos magnéticos o una contracorriente durante el funcionamiento de las maquinas en línea de producción. Se hicieron modificaciones en los programas del PLC, arduino y Labview, de modo que en caso de cualquier cambio imprevisto en alguna señal se parara automáticamente el proceso, alertando al operador que ha ocurrido un incidente en los equipos o error en la máquina sometida a prueba; tales

como los sensores de temperatura, los sensores en el display de la maquina en caso de mostrar error y el envío de señales extras al PLC mediante arduino para constatar que la comunicación este presente.

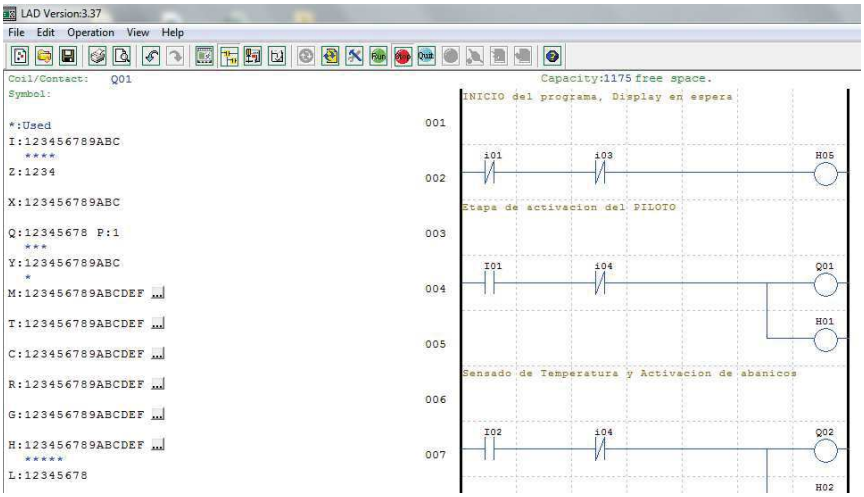


Figura 6. Sección del programa escalera utilizado

El uso de PLC tiene el principal objetivo de acoplar los dispositivos de control en las etapas de alta potencia aplicada durante el proceso, de modo que existiera un aislamiento entre los equipos utilizados, con toma de decisiones según el estado de las entradas presentes y activar las diversas etapas para estresar a la maquina bajo prueba, de este modo se tiene una mayor seguridad en caso de un desperfecto en los equipos. Para interacción con el personal en línea de producción y con la gran ventaja que se contaba con un display propio del PLC, se optó por desplegar diversos mensajes para mantener informado al operador en qué etapa se encontraba la rutina y poder verificar que se estuviera realizando de manera correcta con respecto al manejo de las corrientes y voltajes.

En la etapa de piloto el programa activa el banco de carga a 10Ω, dejando circular una corriente de 20 Amperes a través de la máquina, este modo se da cuando el “Trigger” de la antorcha es accionado sin hacer contacto con la superficie de trabajo, es la etapa anterior al “Arco de voltaje”. El arco de voltaje es posterior a la etapa del piloto, la cual se da cuando la maquina detecta que la antorcha está en contacto con la superficie de trabajo, es un breve tiempo antes de mandar la señal al banco de carga para dejar circular la corriente nominal.

La etapa de “Transfer” es en la cual la maquina deja circular la corriente nominal de funcionamiento, el PLC manda la señal al banco de carga el cual cambia su resistencia a 1Ω aumentando la corriente hasta 120 Amperes, la corriente varía según el modelo bajo prueba. Uno de los problemas durante la ejecución de la prueba es el aumento de temperatura la cual se debe mantener estable, la razón de cambio no debe ser mayor a 5℃ con respecto a la temperatura inicial, en caso de sobrepasarla el PLC mandará la señal de activar los abanicos para la circulación de aire, en caso de no reducirse dentro de un lapso de 1 minuto el PLC cortará la corriente a la máquina y detendrá la prueba como precaución

En caso de algún desperfecto en comunicación con la ausencia de señal por parte del arduino, banco de carga o falta de instalaciones previas al inicio de la prueba, el PLC no dejara iniciar la secuencia mostrando error el cual busca impedir llevar a cabo una situación insegura para el personal que realiza la prueba y los demás a su alrededor. La interfaz en Labview se realizó de manera sencilla

y práctica para el personal en línea de producción (el programa no puede ser mostrado en su totalidad por políticas de privacidad de la empresa), de forma que se puedan evitar problemas, de tal manera que solo sea necesario corroborar que las conexiones se hayan realizado correctamente y tan solamente dar un clic sobre la rutina dependiendo del modelo de equipo bajo prueba.

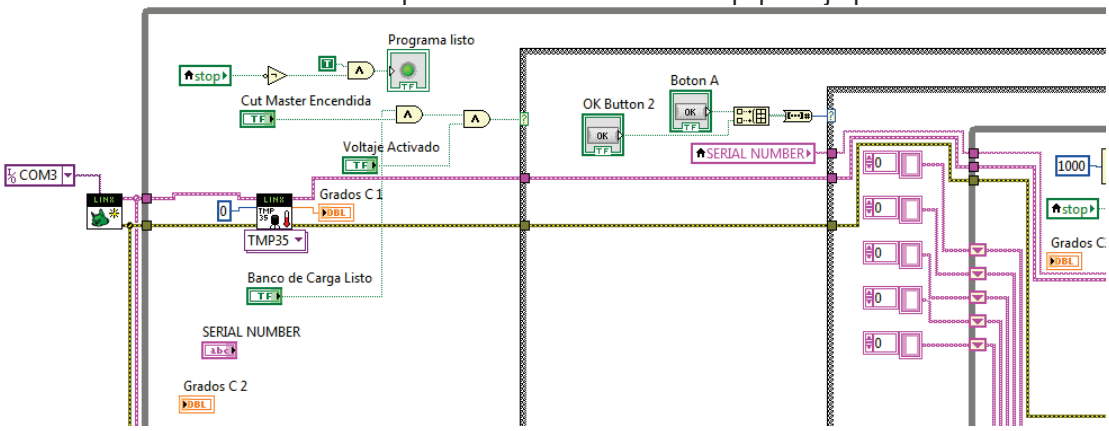


Figura 7. Sección del diagrama de bloques en LabView

El programa tiene una sintaxis sencilla de modo que sea fácil de comprender y modificar en caso de ser requerido, antes de poder ejecutarlo es necesario ingresar ciertos datos tales como el código de barras mediante un scanner y verificar los datos que pide en panel frontal. La etapa del análisis en la prueba es a partir de las señales de entrada proporcionadas por el arduino de tal modo que se verifican tiempos, la comparación de parámetros en la entrada, para así poder llevar a cabo una tabla con los datos obtenidos durante el proceso y generar un archivo .txt para cada máquina probada, teniendo un registro de fácil acceso y actualizado en todo momento.

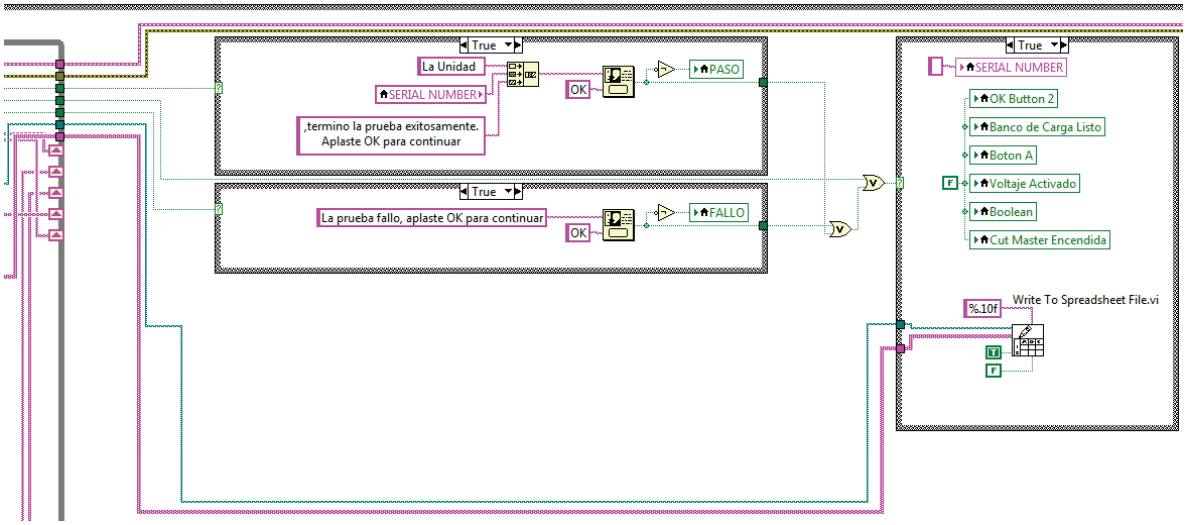


Figura 8. Diagrama de bloques de la etapa de análisis.

En caso de realizarse satisfactoriamente la prueba, la interfaz despliega un mensaje al operador; de lo contrario, si surge un error de la misma forma se desplegará alertando del mismo, teniendo que analizar la máquina para su corrección y comenzar la prueba nuevamente.

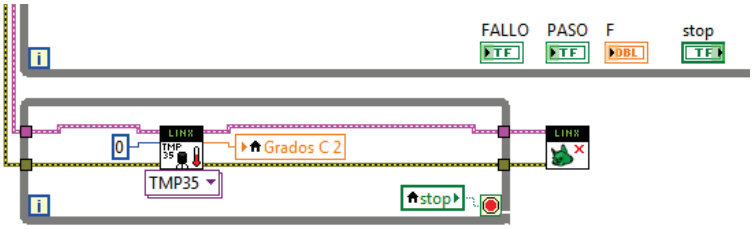


Figura 9. Diagrama de bloques de conexión de labVIEW con Arduino.

La temperatura es registrada en el inicio de la prueba y durante el proceso con el paralelismo en la ejecución de funciones que brinda Labview, de tal modo que se esté comparando por si ocurre algún incremento fuera de lo normal y de esta manera se pueda tomar acción preventiva antes de ocurrir un accidente.

5. Conclusiones

Como conclusión el proyecto se realizó de manera óptima, brindando la solución esperada para los departamentos involucrados, tales como el departamento de calidad el cual buscaba una manera fiable de tener un registro de cada equipo sometido a prueba y poder comprobar que se sometían a este proceso con los parámetros específicos que planteaban, el departamento de producción por otro lado, se vio beneficiado con la disponibilidad de las mesas de Burn In lo cual logra disminuir el cuello de botella producido esta etapa, con la reducción de tiempos muertos por causa de fallas en los equipos de control y el departamento de pruebas del cual surge la idea en el cambio de los equipos en dicha prueba, con el objetivo de poder tener una instalación más confiable a partir de la instalación de nuevos controladores realizando una integración de diversas tecnologías con el fin de brindar una solución óptima a la problemática presente en línea de producción.

La principal recomendación es tomar en cuenta la capacidad de aplicación a diversas líneas de producción dentro de la planta, teniendo un proceso más automático en estas etapas tan importantes para la empresa, con equipos relativamente de bajo costo se puede lograr un mayor control y registro de los eventos producidos en la etapa de Burn In, tomando acciones correctivas según sea requerido.

Referencias

[1] Mandado, E. Acevedo, J.M. & Fernández, C. (2009). Autómatas programables y sistemas de automatización. Marcombo.
[2] Innovación y cualificación. S.I. (2004). Autómatas programables.: Innovación y cualificación.
[3] Tienda de Robótica. (2012). Libro básico de Arduino: electrónica y programación.: Openlibra.
[4] José Rafael Lajara Vizcaíno, José Pelegrí Sebastiá. (2007). LabView: entorno gráfico de programación. : Marcombo.
[5] Ronald W. Larsen. (2009). LabVIEW para ingenieros.: PEARSON.
[6] Thomas L. Floyd. (2001). Principios de circuitos eléctricos.: PEARSON.

Ingeniería y Automatización, Capítulo 36, pp. 443 - 455.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Aplicación Móvil para el Control de Asistencia a Eventos, utilizando Tecnología RFID

Torres Frausto David Antonio, Villegas Téllez Rodrigo, Lozano Vázquez Ángel Gerardo y Serrato Mares Lía Alejandra

Instituto Tecnológico Superior de Irapuato
Coordinación de Ingeniería en Informática e Ingeniería en Sistemas Computacionales
datorres@itesi.edu.mx, rovillegas@itesi.edu.mx, anlozano@itesi.edu.mx, lia.serrato.m@gmail.com

Resumen

El presente proyecto consiste en el desarrollo de una aplicación móvil, que en conjunto con la tecnología RFID, contribuye a la reducción del tiempo de captura de datos, a la reducción de la probabilidad de inconsistencia en la información; además de minimizar el tiempo de migración del registro de asistencia de los eventos académicos dentro del Instituto Tecnológico Superior de Irapuato (ITESI). El proyecto se desarrolló siguiendo la metodología de desarrollo de software Mobile-D, utilizando el entorno de desarrollo Android Studio, en combinación con XAML para el diseño de las interfaces.

La aplicación, una vez instalada en el dispositivo móvil Android, se conecta a una base de datos local en SQLite; en donde se consulta la información leída por medio del lector RFID, y que se encuentra almacenada en tags de tarjetas plásticas. De ésta forma, se muestra la información académica del participante en la pantalla del dispositivo. Al finalizar el registro de los participantes, la aplicación móvil genera una lista en un documento de Microsoft Excel.

Se han realizado varias pruebas en el instituto con un prototipo de la aplicación móvil; en donde se tomó en consideración el proceso tradicional de control de asistencia para compararlo contra el proceso utilizando la aplicación. Los resultados obtenidos muestran una clara tendencia a la reducción del tiempo invertido en las diferentes actividades del proceso.

Palabras clave: Aplicación Móvil, Radiofrecuencia, Control de Asistencia

1. Introducción

El uso de las tecnologías de información se ha convertido en una parte importante de la vida de las personas que interactúan directa o indirectamente con ellas. En los últimos años, se ha convertido en una necesidad para la mayoría de los sectores; desde la industria, el transporte, el comercio, el sector educativo y hasta la manufactura. Gracias a los avances de la investigación y técnicas de fabricación en los dispositivos, computadoras y teléfonos inteligentes, se ha hecho posible la creación de dispositivos con mejores recursos, más ágiles y fáciles de manipular por el usuario.

Un importante avance es la tecnología RFID (*Radio Frequency Identification* o *Identificación por Radio Frecuencia*), la cual es conocida como el sustituto del código de barras. Esta tecnología es conocida actualmente gracias a tres factores importantes: la reducción de costos en la cadena de producción y logística; es una tecnología barata; y principalmente, a los usos que se le ha dado en diversos ámbitos de la vida moderna.

Dicha tecnología funciona a partir de electricidad que viaja por un cable a través de un radio definido por una antena a una frecuencia determinada, para que otros elementos; los "*tags*", lo escuchen.

Un tag es un dispositivo pequeño, que puede ser adherido o incorporado a un producto, animal o persona. Contienen antenas para permitirles recibir y responder a peticiones por radiofrecuencia desde un emisor/receptor RFID. Transmite y recibe ondas analógicas que posteriormente se transforman en cadenas de bits de información digital [1].

Otro avance muy importante son los *Smartphone* o *teléfonos inteligentes*. Estos, son teléfonos móviles en donde se pueden hacer las mismas funciones de una computadora, tales como: editar un documento, navegar por internet y hasta jugar. Son teléfonos que están diseñados para ejecutar múltiples aplicaciones simultáneamente. La mayor ventaja de estos teléfonos es su bajo costo, portabilidad, factibilidad de uso y funcionalidad [2].

En la actualidad ya se cuenta con gran variedad de soluciones para el control de asistencia a eventos que pretenden automatizar el proceso de tomar asistencia de participantes; y en algunos casos, generar documentos electrónicos para la mejor visualización de los asistentes y administrar así los eventos.

En su forma más básica, este proceso se registra en una lista física (hoja de papel), en donde se hace un proceso muy tardado; y en ciertas ocasiones, las hojas pueden llegar a perderse; o bien, el registro de los datos de los participantes suele tornarse confuso. El proceso de registro de asistencia a eventos académicos en el ITESI se lleva a cabo de la forma anteriormente descrita.

Al registrar la información de asistencia en papel, la probabilidad de inconsistencia de la información en muy alta. Por otro lado, los participantes de un evento emplean demasiado tiempo en registrar su información en las listas de asistencia. Dicho registro también puede llegar a confundirse con los datos de otro evento o participante del evento, ya sea que se trate de los datos de un alumno, un profesor, o un participante externo; derivando en ocasiones la pérdida de las listas de asistencia.

Otra causa del deficiente proceso de control de asistencia a eventos es que el responsable del departamento de Desarrollo Humano emplea demasiado tiempo para migrar la información de asistencia al sistema de información que maneja. Entre las consecuencias que provoca esta deficiencia, se encuentran la duplicidad de los datos registrados en las listas de asistencia y el aumento del uso de hojas de papel.

También, suele presentarse la asignación de puntos de desarrollo humano a los alumnos por parte del departamento, sin conocer si el alumno estuvo presente en el evento. Además, se presentan errores en los datos registrados en las listas de asistencia o información incompleta.

Ante esto, surge la idea de desarrollar una aplicación móvil para reducir el exceso de tiempo invertido en el registro a eventos y la probabilidad de inconsistencia al menos en un 70%, y a su vez reducir el tiempo empleado en la migración de la información de asistencia al menos en un 80%.

2. Justificación

El proyecto propuesto tiene como objetivo reducir el tiempo empleado en el proceso de control de asistencia a eventos; esto beneficiara por una parte a los alumnos, al ver reflejado en menor tiempo los puntos de desarrollo humano asignados a dicho evento, en el sistema que maneja el instituto.

Por otro lado, propone un proceso automatizado, el cual beneficiara al encargado del evento al hacer su trabajo más fácil y rápido.

Además, el proyecto propone mejorar el sistema de búsqueda de participantes. Esto beneficia a todos los departamentos que organicen un evento, al momento de asignar los puntos de desarrollo humano más rápido, fácil y efectivo, ahorrando tiempo en este proceso.

El proyecto tiene como alcance abarcar el control de asistencia dentro del ITESI, partiendo de los eventos que se realicen en la carrera de Ingeniera Informática. Por lo que se probará la aplicación móvil en la Semana de la Salud que se lleva a cabo en el instituto, para comprobar los resultados propuestos.

3. Desarrollo

Para el desarrollo del presente proyecto se utilizó la metodología Mobile-D; la cual fue creada basándose en un proyecto de la industria, contando con una participación importante de las empresas de TI finlandesas.

Mobile-D es una mezcla de técnicas de desarrollo de software, fue creada en un periodo de intenso crecimiento de las aplicaciones móviles. Mobile-D es un ciclo de desarrollo muy rápido y apto para equipos pequeños, esta propuesto para que los proyectos trabajados lleguen a su fin en 10 semanas mínimo [3].

Este ciclo de vida está apoyado en tecnologías como lo son: *eXtreme Progamming(XP)*, *Crystal methodologies* y *Rational Unified Process (RUP)*; en donde, se tomó de XP las prácticas de desarrollo, las metodologías de Crystal y el RUP es la base para el diseño completo del ciclo de vida.

El ciclo de vida está dividido en cinco fases: *Exploración*, *Inicialización*, *Producción*, *Estabilización* y *Prueba del Sistema*; todas las fases contienen mínimo tres días de desarrollo distintos en donde se planea, trabaja y libera el proyecto en iteraciones pequeñas, tal como se muestra en la Figura 1. Diagrama de la metodología Mobile-D [3].

Figura 1. Diagrama de la metodología Mobile-D

3.1 Exploración

- *Establecer los grupos de interesados:* dentro de esta actividad se definen los interesados del proyecto, para su mejor control se separan por grupos. En el proyecto se definieron cuatro grupos de interesados, los cuales van a interactuar con el proyecto, los cuales se muestran en la **¡Error! No se encuentra el origen de la referencia.**.

Tabla 1. Grupo de Interesados

Nombre	Rol
Lia Alejandra Serrato Mares Ángel Gerardo Lozano Vázquez	Desarrolladores del Proyecto
Rodrigo Villegas Téllez David Antonio Torres Frausto	Asesores del Proyecto
Erich Montero Rubio	Responsable del departamento de Desarrollo Humano de ITESI
Docentes y alumnos	Usuarios finales

- *Enfoque de objetivos:* es en donde se levantan los requerimientos del proyecto y se hace una investigación acerca de lo ya existente y todos los diferentes temas que conlleva el proyecto. Dicha actividad se divide en las siguientes sub-actividades:

    - *Investigación de Proyectos usando tecnología RFID.* Se realizó una investigación de proyectos que trabajen de forma similar al proyecto propuesto, en donde se identificaron los siguientes:

        - *Aplicación Android para el control automático de la asistencia a clase.* En el cual se propuso una aplicación móvil instalada en los teléfonos celulares de los profesores. Mediante esta aplicación, el receptor (App de los profesores) llamaba a los emisores (celulares con bluetooth de los estudiantes) para el pase de lista automático mediante la dirección MAC del dispositivo.

        - *Ticketea Checkpoint para Android.* Se trata de una aplicación pensada para facilitar la gestión de eventos y del control de acceso, fácil y cómodamente desde un dispositivo con sistema operativo Android. A través de la cámara del móvil se puede escanear el código QR de la entrada y validar contra los servidores de Ticketea [4].

        - *Tracesign.* Es un software de control de asistencia con tecnología NFC. El funcionamiento de esta aplicación comienza desde que se crea un nuevo evento y se asocia el listado de asistentes a este. Se pueden importar listas en Excel o integrar sistemas internos para importar datos y usuarios automáticamente. Después, cada usuario acredita su asistencia. Finalmente, se puede consultar vía web toda la información agregada de asistencia, materias o fechas e histórico por usuario.

        - *Sistema RFID de qronnos.* Permite controlar el acceso a los recintos y a las distintas salas pasando por arcos RFID. Los tags, identifican al congresista de una

manera única. El sistema y el software de gestión de inscripciones QCongresos MySQL, genera gran cantidad de informes relacionados con asistencia y tiempos de estancia en recintos y salas [5].

- *Investigación sobre las plataformas más usadas de desarrollo móvil.* Se realizó una investigación para conocer las plataformas de desarrollo de aplicaciones más utilizadas para proyectos como el propuesto, las cuales se muestran en la **¡Error! No se encuentra el origen de la referencia.**.

Tabla 2. Plataformas de desarrollo más usadas para aplicaciones móviles

Top	Plataforma
1	Android Studio
2	Titanium SDK
3	Genymotion
4	Rhonmobile
5	Ubertester

Con base en la investigación anterior, se llegó a la conclusión de que se utilizaría la plataforma de Android Studio.

- *Establecimiento del Proyecto:* se definen las herramientas a utilizar, para el desarrollo del proyecto.
  - *Selección del entorno de desarrollo:* Android Studio y XAML para el diseño de las interfaces de la aplicación.
  - *Selección de lector RFID y tarjetas tag a usar*:

Figura 2. Lector RFID USB y tag

3.2 Inicialización

- *Establecer los recursos físicos y técnicos:* dicha actividad comprende el medio ambiente de desarrollo del proyecto. Para el desarrollo del prototipo de la aplicación móvil se trabajó con una computadora con las características que se muestran en la **¡Error! No se encuentra el origen de la referencia.** y las características del teléfono móvil en la **¡Error! No se encuentra el origen de la referencia.**.

Tabla 3. Características del equipo de cómputo usado en el desarrollo del proyecto

Sistema Operativo	Windows 8.0 Pro
Fabricante	Samsung Electronics
Procesador	Intel® Celeron® CPU B820 @ 1.70GHz
Memoria Instalada (RAM)	10,0 GB
Arquitectura	X64, 64 bits

Tabla 4. Características del teléfono móvil que se usó en el prototipo

Marca	Motorola
Numero de modelo	Moto G XT1033 Dual 16GB
Versión de sistema operativo	Android 5.1 Lollipop
Arquitectura - Chipset	32bits - Qualcomm Snapdragon 400 MSM8226
Memoria Instalada (RAM)	10,0 GB
CPU Procesador / Núcleos	1.2Ghz Quad-Core ARM Cortex-A7
Memoria RAM	1GB LPDDR2
Memoria interna	16GB (13GB accesible al usuario)
Tamaño de la pantalla	4.5" pulgadas
USB	USB 2.0 Micro-B (Micro-USB)

- *Planear el diseño de la aplicación en papel con funcionamiento de cada elemento de la aplicación.* Se presentó un diseño inicial como wireframe en la aplicación, el cual consistía en un diseño simple, con colores referentes al ITESI, el logo de esta institución y un menú más amplio. Por otro lado, se diseñó la base de datos con la cual trabajara la aplicación móvil para gestionar los eventos, participantes y la asistencia.

- *Trabajar en la funcionalidad de las pantallas.* Aquí se describe cómo va a interactuar cada pantalla en relación con el usuario, para lanzar el diseño en papel con descripción de las pantallas de la aplicación. El diseño de la base de datos se muestra en la Figura 3. Diagrama relacional de la base de datos.

  El diseño inicial de la aplicación quedo como se muestra en la Figura 4. Diseño Inicial de la aplicación, pero este diseño se fue modificando con el paso del desarrollo del proyecto.

Ingeniería y Automatización, Capítulo 36, pp. 443 - 455.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

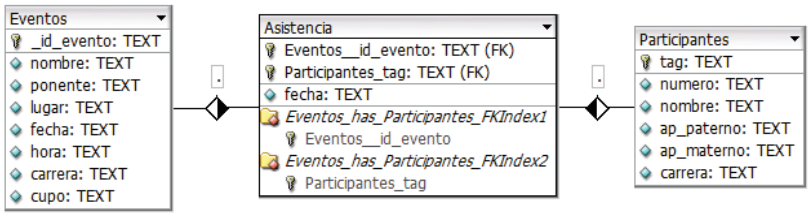
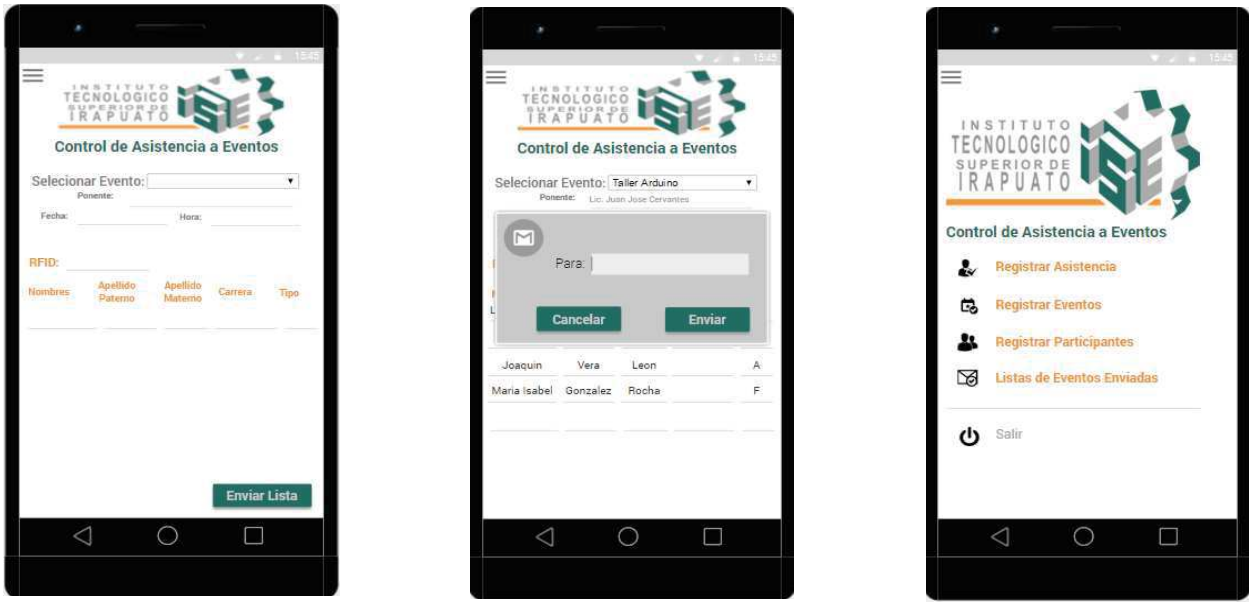


Figura 3. Diagrama relacional de la base de datos



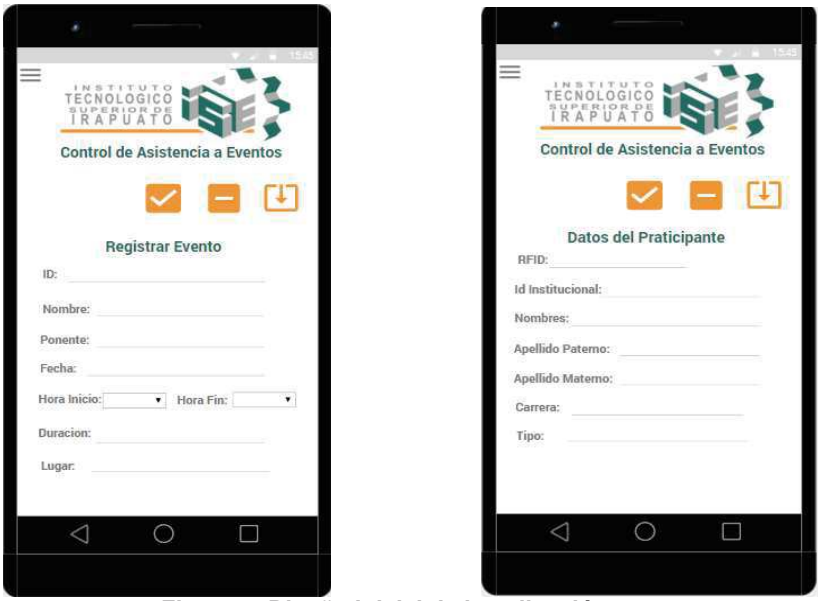


Figura 4. Diseño Inicial de la aplicación

3.3 Producción

La aplicación móvil consiste en siete módulos. A continuación, se describe el desarrollo de cada uno de ellos.

- **Iteración I**
  - *Módulo de Administración de Participantes.* En este módulo se pueden realizar los siguientes eventos:

    - *Alta de participantes manualmente.* El usuario final (docente) deberá ingresar el número de control, asignarle un número RFID, nombre, apellidos y carrera.

    - *Alta de participantes a partir de una lista en archivo de Excel.* El usuario final tendrá la opción de importar una lista de participantes, guardada en el dispositivo, con solo dar clic en el icono de "Excel".

    - *Baja de Participantes.* El usuario final podrá eliminar un participante, ingresando su número de control.

    - *Consulta de participantes.* El usuario podrá consultar los datos de un participante ingresando el número de control de dicho participante.

    - *Modificación de datos del participante.* El usuario final podrá modificar los datos de un participante corrigiendo dichos datos y dando clic en el icono "actualizar".

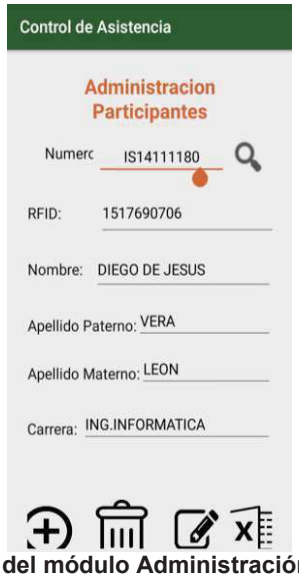


Figura 5. Pantalla del módulo Administración de Participantes

- **Iteración II**
  - *Módulo de Administración de Eventos. En este módulo se pueden realizar los siguientes eventos:*
    - *Alta de eventos. El usuario final (Docente) debe ingresar el Id del evento, así como el nombre del evento, el nombre del ponente que impartirá dicho evento, lugar, fecha, hora en la que se llevará cabo el evento, carrera que lo organiza y el cupo.*

    - *Baja de Eventos. El usuario final podrá eliminar un evento, ingresando el Id del evento.*

- *Consulta de eventos. El usuario podrá consultar los datos de un evento seleccionando el nombre del evento del spinner.*

- *Modificación de datos del evento. El usuario final podrá modificar los datos de un evento corrigiendo dichos datos y dando clic en el icono de "actualizar".*

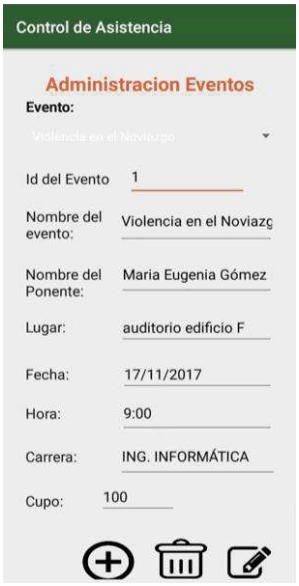


Figura 6. Pantalla del módulo de Administración de Eventos

- **Iteración III**
  - *Modulo Lista de Participantes.* En este módulo el usuario final podrá ver todos los participantes por carrera que se encuentran guardados en la base de datos, seleccionando el nombre de la carrera en el spinner, en donde a continuación muestra la lista de estos participantes.

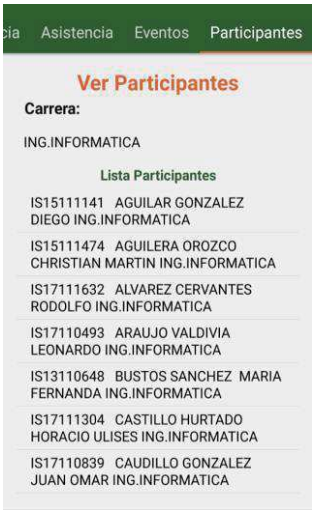


Figura 7. Pantalla del módulo Lista de Participantes

- **Iteración IV**
  - *Modulo Eventos.* En este módulo el usuario final puede consultar los datos generales del evento, seleccionado en el spinner el nombre del evento.

Figura 8. Pantalla del módulo Ver Eventos

- **Iteración V**
  - *Modulo Asistencia.* En este módulo se genera la lista de asistencia de los participantes que asistieron a un evento; Primeramente, se selecciona el evento previamente creado, para consultar datos importantes del evento; como lo son el ID del evento, el cupo y la carrera. Posteriormente, se posiciona en la caja de texto de RFID para comenzar con el pase de las tarjetas tag por el lector para registrar la asistencia.

Figura 9. Pantalla del módulo Generar Asistencia

- **Iteración VI**
  - *Modulo Lista de Asistencia.* En este módulo se genera automáticamente la lista de asistencia a un evento, en donde se ven datos de los asistentes; como lo son el número de control, nombre completo, y su carrera. El usuario final puede consultar las listas generadas por evento, seleccionando en el spinner el nombre del evento. Una vez que el usuario presione el botón de correo, se genera la lista de asistencia seleccionada en un archivo de Excel, en una carpeta que se crea en el teléfono móvil al instalar la aplicación.

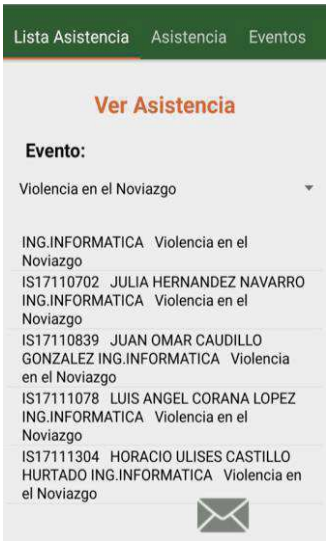


Figura 10. Pantalla del módulo Lista de Asistencia

- **Iteración VII**
  - *Modulo Enviar Lista de Asistencia.* Dentro de este módulo se puede agregar o no el destinatario, asunto y mensaje a enviar por correo en Gmail; y al dar clic en el botón de enviar, se abre una lista con las opciones para enviar la lista de asistencia (Gmail, WhatsApp, agregar a Dropbox o guardar en Drive).

Figura 11. Pantalla del módulo Enviar lista de Asistencia

3.4 Estabilización

- *Planeación.* Se plantea que como mejora de la aplicación, se incluya una fotografía del participante para mayor control de acceso al evento. Además, que la aplicación esté conectada en red para el control de acceso a eventos que se puedan llevar a cabo al mismo tiempo en diferentes lugares. Por último, se vio la posibilidad de controlar la estancia del participante en el evento.

- *Trabajo.* La aplicación se instaló en un dispositivo móvil para su posterior uso en pruebas reales. Además, se identificó cada comportamiento que se pueda llegar a observar de la aplicación. También, se les entrego una tarjeta RFID a 85 alumnos de la carrera de Ingeniería Informática de diferentes semestres.

- *Documentación.* Se localizaron errores, se documentaron y se resolvieron conforme el desarrollo de la aplicación, para tenerla lista en la fase de pruebas.

- *Lanzamiento.* Se presenta la aplicación terminada al responsable del Departamento de Desarrollo Humano para su evaluación.

3.5 Pruebas y Reparaciones

- *Pruebas del Sistema.* Las pruebas de la aplicación como prototipo se realizaron en la Semana de la Salud que organiza el ITESI, la cual se desarrolló del 13 al 18 de noviembre del 2017. Las pruebas se realizaron en dos ocasiones diferentes: una en la conferencia llamada “Estrés” en donde se registró la asistencia de 10 alumnos de la carrera de Ingeniería Informática de tercer semestre; y otra en la conferencia “Violencia en el Noviazgo” en donde se registró la asistencia de 26 alumnos de la misma carrera de primer semestre.

- *Trabajo.* Se detectaron defectos en el diseño de la aplicación, que se corrigieron al término de la primera prueba. Los cambios que se realizaron fueron:
  - Diseño de las interfaces.
  - Deshabilitar las cajas de texto de consulta en la pantalla de “Generar Asistencia”, para que no se corrompa la información del participante y posicionar el cursor en la caja de texto de registro de “RFID”.

- *Documentación.* Se documentó el desarrollo de la aplicación y el trabajo de investigación.

- *Lanzamiento.* Se demostró en ambiente real el funcionamiento del prototipo y se enviaron las listas de asistencia al responsable del Departamento de Desarrollo Humano para que dispusiera de ellas, para agregar a los participantes los puntos de desarrollo humano correspondientes por la asistencia a los eventos académicos.

4. Resultados

Con base en la entrevista realizada al responsable del Departamento de Desarrollo Humano y a las pruebas realizadas, se obtuvieron los resultados mostrados en la Tabla 5. Tabla de resultados obtenidos en las pruebas.

Los resultados arrojan que se redujo el tiempo de registro de los participantes a un evento en un 83% del tiempo. Además, el tiempo de migración de la información que requiere el encargado del Departamento de Desarrollo Humano, se redujo un 96% del tiempo.

Tabla 5. Tabla de resultados obtenidos en las pruebas

	Proceso Actual	Proceso usando el Prototipo
Tiempo de registro de información de asistencia	1 a 2 participantes en 1 min	15 participantes en 1 min
Probabilidad de inconsistencia de la información	80%	5%
Tiempo de migración de las listas de asistencia por carrera.	16 horas	0.5 horas
Tiempo de migración de las listas de asistencia de todas las carreras	208 horas	8 horas

5. Conclusiones

La aplicación móvil, junto con la tecnología RFID cumplió en su totalidad el objetivo plateado, comprobando la hipótesis, mostrando una mayor eficacia, eficiencia y promoviendo un mejor control, produciendo de esta manera un ahorro de tiempo notable para los participantes de los eventos, docentes encargados del evento y para el responsable del Departamento de Desarrollo Humano. Además, se propone como herramienta alternativa en el proceso de control de asistencia a eventos en una institución educativa, ya que es un prototipo de bajo costo, fácil de usar y con mayores ventajas en comparación a otros sistemas con el mismo fin. Cabe mencionar, que en el Instituto ya ha trabajado con esta tecnología aplicada a otros escenarios como el acceso a estacionamiento e inventario de mobiliario y préstamo de libros en la biblioteca.

Para el uso de la aplicación móvil se recomiendan los siguientes aspectos:

- Tener un dispositivo móvil con sistema operativo Android de 4.0 (Jellybean) en adelante.
- Tener instalada una aplicación que permita gestionar los archivos en el dispositivo móvil.
- Tener instalada una aplicación que permita visualizar los archivos de tipo “xls”. (Excel).

Se propone como trabajo futuro que la aplicación móvil esté conectada en red, para gestionar el control de acceso a los eventos que se realicen en diferentes carreras, al mismo tiempo. Además, se ve como mejora que la gestión de la información de eventos y participantes sea controlada por una aplicación web y que la aplicación móvil consuma esos datos.

Referencias

[1] Panda ID Soluciones. *“¿Qué es una etiqueta RFID?”*, *Sitio oficial Panda ID Soluciones*. 2017. Disponible en: http://pandaid.com/que-es-una-etiqueta-rfid/

[2] Goodwill Community Foundation. *“El smartphone o teléfono inteligente”,* 2018. Disponible en: https://edu.gcfglobal.org/es/informatica-basica/el-smartphone-o-telefono-inteligente/1/

[3] Jäälinoja J., Hanhineva A., Koskela J. *“Mobile-D/System Test & Fix”*, 2004. Disponible en: http://agile.vtt.fi/mobile-d_docs/systemtest&fix/mobiledpattern_systemtest&fix.pdf

[4] Ticketea. *“Ticketea Checkpoint”*, *Sitio oficial Ticketea*. 2018. Disponible en: https://www.ticketea.com/producto/checkpoint/

[5] SoftCongres. *“Control de Asistentes RFID radiofrecuencia”*, *Sitio oficial SoftCongres*. 2017. Disponible en: https://www.softcongres.com/index.php?idpagina=55&idioma=cas

Ingeniería y Automatización, Capítulo 37, pp. 456 - 466.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Estación Meteorológica de Bajo Costo Implementada con PSoC5LP y Comunicación por Bluetooth

Julián R. Camargo L.[1*], César A. Perdomo Ch.[2]
[1] Electronic Engineering Department, Engineering Faculty, Distrital Francisco José de Caldas University, Bogotá D.C., Colombia;
jcamargo@udistrital.edu.co
[2] Electronic Engineering Department, Engineering Faculty, Distrital Francisco José de Caldas University, Bogotá D.C., Colombia;
cperdomo@udistrital.edu.co

En este trabajo se presenta el diseño y la implementación del prototipo de una estación meteorológica de bajo costo, que permite la medición de variables meteorológicas mediante sensores de diversas características. Los datos medidos se presentan a través de una sencilla aplicación alojada en un Smartphone, adaptable a cualquier dispositivo móvil y con acceso público. El documento inicialmente presenta algunos fundamentos teóricos, luego características de cada uno de los sensores utilizados, en seguida detalles de manejo, configuración y programación de los bloques internos del PSoC, para finalizar mostrando el resultado final y las conclusiones.

1. Introduction

La meteorología y la climatología son ciencias que se dedican al estudio de los procesos que ocurren en la atmósfera, además de la interacción de estos procesos con otros sistemas. La medición de variables físicas que afectan los procesos atmosféricos y climáticos, es una pieza clave en los estudios e investigaciones que competen a estas ciencias. De este hecho surge la necesidad de diseñar y desarrollar los instrumentos de medición, es decir las estaciones meteorológicas, para obtener los datos necesarios que después de ser procesados por los expertos, se traducirán en información valiosísima, que puede ser utilizada para la prevención y atención de desastres, además de beneficiar al sector agrícola y ganadero y también ser de utilidad en los pronósticos y modelamiento del cambio climático.

En Colombia existe una gran cantidad de climas debido a su ubicación geográfica, esto permite la generación de procesos meteorológicos de micro-escala y meso-escala que afectan directamente las actividades socioeconómicas del país, como por ejemplo la agricultura y ganadería [1]. La necesidad de protegerse de los efectos de estos fenómenos y la intención de prevenir catástrofes, ha hecho que la población colombina demande más información sobre los procesos y fenómenos meteorológicos.

En la geografía colombiana el IDEAM (Instituto de Hidrología, Meteorología y Estudios Ambientales) es el encargado de obtener, analizar, estudiar, procesar y divulgar información básica sobre hidrología, hidrogeología y meteorología, cuenta con una red de estaciones de observación, medición y registro que se distribuye por todo el país [2], esta red está formada en su mayoría por estaciones que necesitan de una persona que registre los datos medidos, es decir que no son autónomas. En la actualidad un porcentaje muy bajo (12%) de estaciones meteorológicas en el país tiene la capacidad de transmitir su información en tiempo real, esto hace que la información que sirve como base para el modelamiento de los pronósticos del clima llegue con retrasos a la central, lo cual genera reportes tardíos y pronósticos no acertados [3].

Hoy en día muchas de las cosechas y cultivos son afectados enormemente por la escasa capacidad para prever y cubrir los riesgos asociados a los fenómenos climáticos anormales como granizadas, heladas, exceso de lluvias y fuertes vientos [4]. Solo para el año 2010 se presentaron pérdidas por 89.004 millones de pesos en los cultivos del país debido a estos fenómenos anormales en el clima [5]. Esto en gran parte se debe a la falta de información que obtienen los pequeños y medianos productores y a la falta de acceso a pronósticos adecuados. Por medio del diseño de una estación meteorológica modular autónoma de bajo costo que provea información en tiempo real y que se pueda adaptar a pequeños entornos para crear redes de estaciones, el productor o pequeño empresario tendrá la posibilidad de acceder a datos del clima sin retrasos que le permiten adoptar estrategias tempranas de prevención en sus cultivos para mitigar algunos de los posibles efectos debidos a cambios drásticos del clima.

Algunos trabajos realizados en algunos países de habla hispana muestran desarrollos de estaciones meteorológicas que pueden transmitir su información en tiempo real mediante algún sistema de comunicación, bien sea SMS, WiFi, Ethernet o Satelital [6]–[10], sin embargo, no se encuentra un sistema de bajo costo que permita su adaptación a diferentes condiciones de medición según los parámetros que el usuario desee.

2. Fundamentación teórica

2.1 Meteorología

La Meteorología es la ciencia encargada del estudio de la atmósfera, de sus propiedades y de los fenómenos que en ella tienen lugar, los llamados meteoros. El estudio de la atmósfera se basa en el conocimiento de una serie de magnitudes, o variables meteorológicas, como la temperatura, la presión atmosférica o la humedad, las cuales varían tanto en el espacio como en el tiempo.

Cuando se describen las condiciones atmosféricas en un momento y lugar concretos, se está hablando del tiempo atmosférico. El tiempo atmosférico es uno de los principales condicionantes de las actividades que el ser humano realiza, especialmente de aquellas que se realizan al aire libre, como la agricultura. A diario aparece información meteorológica en los medios de comunicación y, aunque a veces ésta es motivo de las conversaciones más triviales, se sabe que la comprensión del tiempo implica conocer un buen número de conceptos científicos, no todos ellos sencillos.

Desde tiempos inmemoriales, los hombres han admirado los fenómenos atmosféricos y han intentado explicar sus causas. Mientras no hubo instrumentos, ni grandes conocimientos científicos, la magia y la religión sirvieron de explicación a la mayor parte de los fenómenos meteorológicos. Pero hoy día, la meteorología es una ciencia tremendamente avanzada, basada en el conocimiento de la Física y en el uso de las más modernas tecnologías. Los meteorólogos son capaces, incluso, de predecir el tiempo hasta con una semana de antelación sin apenas fallar [11].

2.2. Tipos de estaciones meteorológicas

La OMM (Organización Mundial Meteorológica) clasifica a las estaciones meteorológicas basada en diferentes criterios [12].

Según su finalidad:

- Sinóptica
- Climatológica
- Agrícolas
- Especiales
- Aeronáuticas
- Satelitales

De acuerdo a la magnitud de las observaciones:
- Principales
- Ordinarias
- Auxiliares

Según el lugar de observación:
- Terrestre
- Aéreas
- Marítimas

2.3 Estaciones meteorológicas en Colombia

La red meteorológica del IDEAM está compuesta por un conjunto de estaciones de observación, medición y registro que se clasifican dependiendo de su función [2].

- Estación Pluviométrica (PM): estación meteorológica dotada de un pluviómetro o recipiente que permite medir la cantidad de lluvia caída durante un período determinado de tiempo.
- Estación Pluviográfica (PG): estación dotada de un pluviógrafo que registra en forma continua la precipitación en una gráfica que permite conocer la cantidad, duración, intensidad y periodo en que ha ocurrido la lluvia.
- Estación Climatológica Principal (CP): en ella se realizan observaciones de precipitación, temperatura y humedad del aire, viento, radiación y/o brillo solar, evaporación y fenómenos especiales. Se efectúan tres observaciones diarias: 07, 13 y 19 horas.
- Estación Climatológica Ordinaria (CO): en ellas existen al menos un pluviómetro, un pluviógrafo y un psicrómetro. Se efectúan tres observaciones diarias: 07, 13 y 19 horas.
- Estación Sinóptica Principal (SP): en ellas se efectúan observaciones de los principales elementos meteorológicos en horas convenidas internacionalmente. Los datos corresponden a nubosidad, dirección y velocidad del viento, presión atmosférica, temperatura del aire, tipo y altura de nubes, visibilidad, fenómenos especiales, humedad, precipitación, temperaturas extremas y secuencia de los fenómenos atmosféricos.
- Estación Sinóptica Suplementaria (SS): al igual que en la estación anterior, las observaciones se realizan a horas convenidas internacionalmente y los datos corresponden comúnmente a visibilidad, fenómenos especiales, tiempo atmosférico, nubosidad, estado del suelo, precipitación, temperatura del aire, humedad del aire y velocidad y dirección del viento.
- Estación Agrometeorológica (AM): se realizan observaciones meteorológicas que ayudan a determinar las relaciones entre el tiempo y el clima, por una parte, y el desarrollo de cultivos agrícolas y animales, por la otra. Incluye el mismo programa de observaciones de la estación CP, más registros de temperatura y eventualmente humedad, a varias profundidades (hasta un metro) y en la capa cercana al suelo (0, 10 y 20 cm sobre el suelo).
- Estación de Radiosonda (RS): tiene por finalidad la observación de temperatura, presión, humedad y viento en las capas altas de la atmósfera (tropósfera y baja estratósfera), mediante el rastreo, por medios electrónicos o de radar, de la trayectoria de un globo meteorológico que asciende libremente.
- Estación Meteorológica Especial: tiene como finalidad la observación de un fenómeno en particular; el IDEAM la usa específicamente para observación de heladas (temperaturas mínimas iguales o inferiores a 0°C).

3. Diseño y sensores utilizados

En la Figura 1 muestra el diagrama de bloques con la solución propuesta para el diseño de la estación meteorológica de bajo costo.

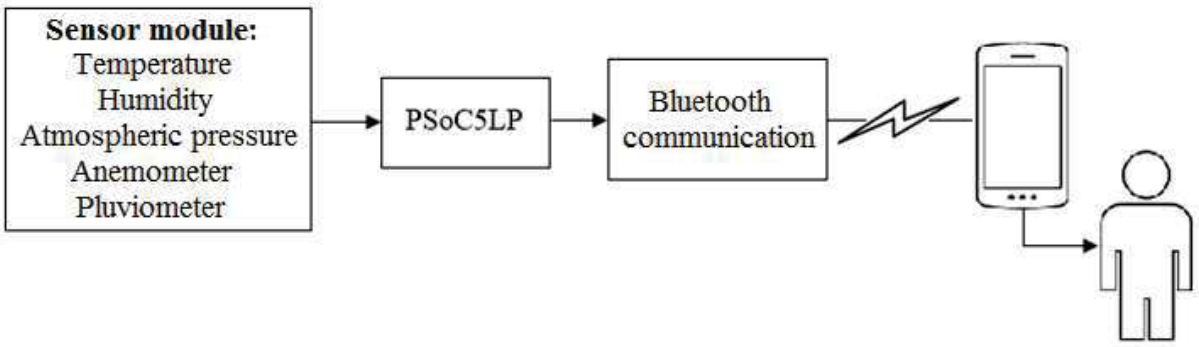


Figura 1. Diagrama de bloques con la solución propuesta

Humidity sensor HIH-4000

It is a sensor whose manufacturer is Honeywell, reviewing the user manual you get the response presented by this sensor in Figura 2.

As it is observed the output is linear, but it presents a variation in the slope of the line which depends on the ambient temperature on which the sensor is located. The Multifunction Environment Meter CEM-DT8820, which measures between 25% RH to 95% RH with 0.1% RH resolution and fast time response, was used to make an optimal characterization of the sensor response.

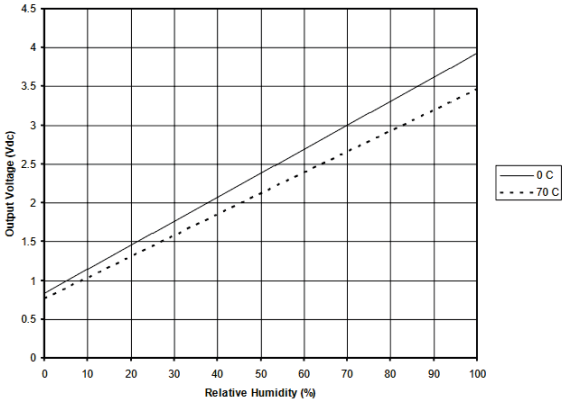


Figura 2. Humidity sensor response HIH-4000 [13]

Then it proceeded to perform the respective measurement of the ambient humidity present in the place where the characterization was being done. Once the measurement was stabilized, the measurement given by the standard equipment which marked 58% of relative humidity was noted, on the other hand, the output voltage of the sensor was measured, which gave a value of 2.27V. This would be a point of the line, but to be able to generate the respective equation, it is necessary to have 2 points. For the other point, the 0% humidity value was taken as reference, which is almost the same for the different temperatures and which according to the user manual is 0.826V.

Now, with the two points it can be find the equation of the line, but before using the equation inside the PSoC, the voltage values have to be passed to their value in bits. As the ADC was set to 10 bits and from 0 to 5V this gives a value of 4.8828mV per bit.

Anemómetro

Es un instrumento que se usa para medir la velocidad del viento en un determinado sitio geográfico, la unidad de medida se señala en metros por segundo (m/s). Existen diferentes tipos de anemómetros que se caracterizan por su precisión y por ende la tecnología con el que está construido, entre los más básicos se encuentra el anemómetro de molinete (Figura 3), que es una especie de molino con tres aspas que se encargan de poner resistencia al paso del viento para luego ser movidas y así registrar un movimiento, también existen el anemómetro ultrasónico, anemómetro de láser y anemómetro de cazoletas [9]. Para el díselo se utilizó anemómetro de molinete acoplado a un motor DC operando como un generador de voltaje cuyo valor generado es medido por el ADC del PSoC5LP.

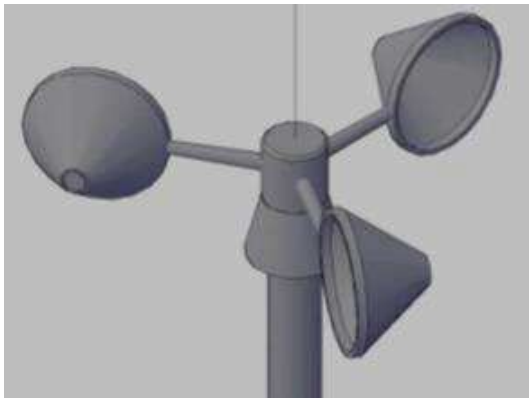


Figura 3. Anemómetro usado en la estación meteorológica

Pluviómetro

Un pluviómetro es un instrumento meteorológico el cual da una medida métrica de la lluvia o precipitación caída en un punto determinado y en un espacio temporal concreto. Con él se mide la lluvia o nieve caída, esta medición la hace en mm o en l/m2.

El pluviómetro es un instrumento sencillo de construir, sobretodo su versión totalizadora, y por eso es uno de los instrumentos meteorológicos más antiguos y gracias a los cuales nos permite tener series pluviométricas más largas.

Existen varios modelos de pluviómetros según su diseño y funcionalidad. La primera clasificación se da entre pluviómetros analógicos y los pluviómetros digitales. En segundo término se van a tener los modelos totalizadores y los modelos que van a medir intensidad de precipitación [14].

El sensor de lluvia o pluviómetro utilizado en la estación implementada es de doble cubeta basculante de tipo digital, cada 0.2794 mm de lluvia provoca un cierre de contacto momentáneo. El interruptor está conectado a los dos hilos conductores con cable RJ11 de salida.

Presión atmosférica y temperatura (BMP 280)

Un barómetro es un instrumento que mide la presión atmosférica, la presión atmosférica es el peso por unidad de superficie ejercida por la atmósfera y sus unidades son el hectoPascal (hPa) [15].

El sensor utilizado para medir la presión barométrica en la estación fue el BMP280 fabricado por Bosch Sensortec que se muestra en la Figura 4. Este sensor utiliza los protocolos de comunicación I^2C y SPI, además incorpora un sensor de temperatura que también se utilizó en la implementación de la estación meteorológica.

Figura 4. Sensor de presión BMP280 [16]

Las principales características de este sensor se resumen en la Tabla 1.

Tabla 1 Características de sensor de presión atmosférica y temperatura BMP280 [16].

Rango de medición Presión Temperatura	 300 hPa a 1100 hPa -40 a +85°C
Resolución de medición Presión Temperatura	 0.16 hPa 0.01 °C
Voltaje de alimentación	1.71 a 3.6V (VDD)
Resolución	16 -20 bits
Consumo de corriente (Max.)	4.2 µA
Filter options	Five bandwidths

4. METODOLOGÍA

Para el diseño del ecualizador digital, el proceso se dividió en varias etapas que facilitan este proceso en su análisis posterior.

Captura de señales analógicas

El bloque ADC_SAR está encargado de recibir los datos medidos por los sensores de velocidad de viento y humedad relativa, que son entradas de tipo analógico, a través del uso de dos pines del PSoC5LP. Se configuró el modulo con una resolución de12 bits, un rango de voltaje de desde Vss a Vdd, conversión única y reloj interno con frecuencia de 1MHz. Luego de recibir el dato y almacenarlo en una variable, se realizan las conversiones necesarias para cada medición para que el dato pueda ser enviado de manera correcta al Smartphone. En la Figura 5 puede verse una imagen de ADC y su conexión con los sensores.

Ingeniería y Automatización, Capítulo 37, pp. 456 - 466.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

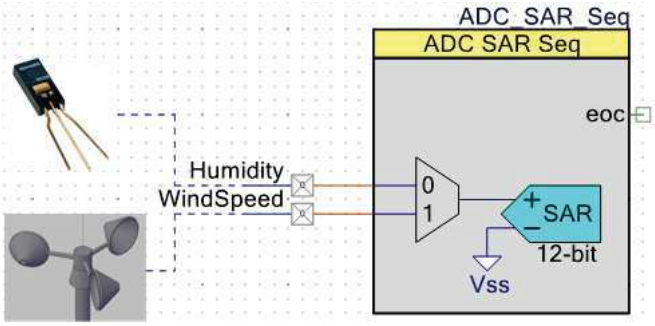


Figura 5. Conexión del bloque ADC a los sensores analógicos

El código utilizado en la PSoC5LP se muestra en la Figura 6**¡Error! No se encuentra el origen de la referencia.**, donde se muestran las operaciones realizadas para calcular el valor de la humedad relativa y la velocidad del viento.

```
ADC_SAR_Seq_StartConvert();
ADC_SAR_Seq_IsEndConversion(ADC_SAR_Seq_WAIT_FOR_RESULT);
anemo= ADC_SAR_Seq_GetResult16(0);
humed= ADC_SAR_Seq_GetResult16(1);
humed=humed-715;
humed=humed*0.03866;
anemo=(anemo*0.028);
```

Figura 6. Código implementado para la medición de humedad y velocidad del viento

Medición de cantidad de lluvia

El módulo Timer, en conjunto con un pin digital de entrada y sus respectivas interrupciones toman la medición del pluviómetro (sensor de caída de lluvia). Para esto, se toma el dato justo cuando el sensor envía el pulso, lo que genera la interrupción en el pin de entrada. Mientras esto ocurre, el módulo Timer está configurado con un periodo determinado de 3 segundos, y como se conoce previamente el volumen que tiene el instrumento el cual tiene una capacidad de 0.2794mm, se toma la medida del tiempo que este tarda en llenarse, obteniendo así una medición en términos de mm/s (Figura 7).

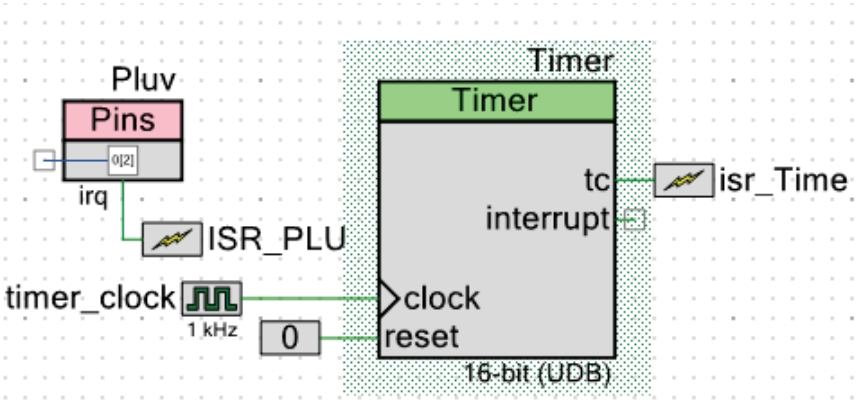


Figura 7. Conexión del bloque Timer al sensor de lluvia

Temperatura y presión atmosférica

El bloque I2C del PSoC5LP permite la lectura del dato obtenido por el sensor BMP280 tanto para la presión atmosférica como para la temperatura. Requiere de una serie de lecturas previas y

operaciones suministradas por el datasheet y que deben codificarse para que este sensor funcione de manera correcta. Lo primero que debe hacerse es indicar la dirección del sensor, definir los siete parámetros de calibración inicial para la medición, definir los registros correspondientes a presión y control e ID del mismo. En la Figura 8 se presenta la conexión del bloque I2C con el sensor BMP280.

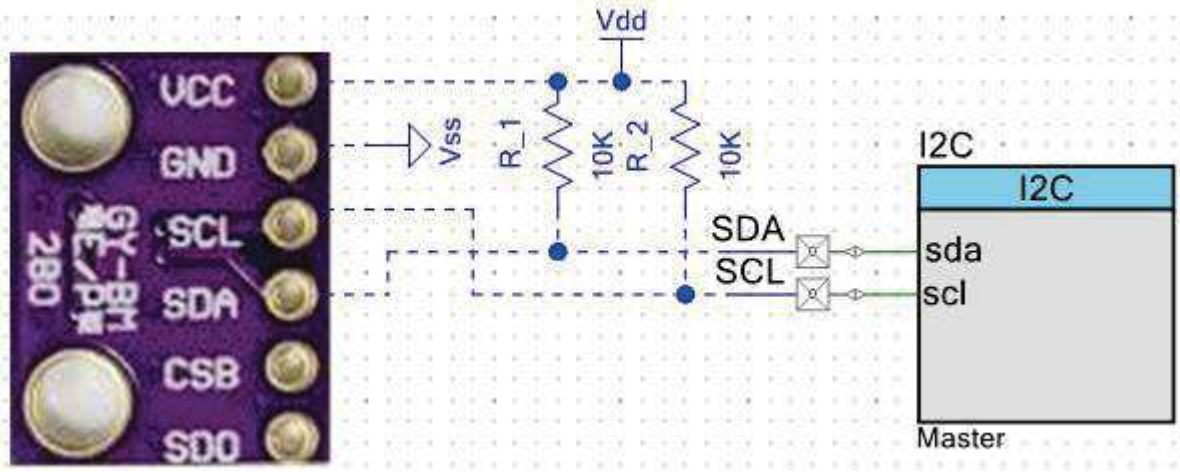


Figura 8. Conexión del bloque I2C con el sensor BMP280

Este sensor tiene desarrollada una librería en C que facilita la configuración del mismo, que es proporcionada por el fabricante, sin embargo, esta debe adecuarse al sistema embebido que se esté empleando. De esta forma, se crearon funciones encargadas de leer datos de los registros de diferentes tamaños, 8, 16, o 24. Con esta lectura de datos, es posible obtener los valores de calibración iniciales que trae por defecto dicho sensor y es posible inicializarlo para comenzar la medición. En la Figura 9 se presenta una parte del código implementado en el PSoC para la operación del BMP280, específicamente el código para la calibración del sensor.

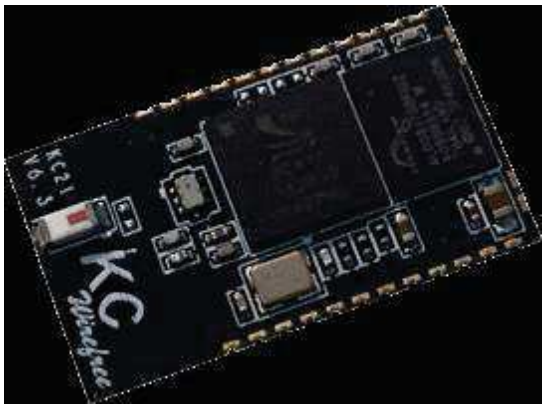
```
void Leer_Coeficientes_Calibracion(){
    datos_cal.dig_P1 = Leer_Datos_U16_LE(BME280_DIG_P1_REG);
    datos_cal.dig_P2 = (int16)Leer_Datos_U16_LE(BME280_DIG_P2_REG);
    datos_cal.dig_P3 = (int16)Leer_Datos_U16_LE(BME280_DIG_P3_REG);
    datos_cal.dig_P4 = (int16)Leer_Datos_U16_LE(BME280_DIG_P4_REG);
    datos_cal.dig_P5 = (int16)Leer_Datos_U16_LE(BME280_DIG_P5_REG);
    datos_cal.dig_P6 = (int16)Leer_Datos_U16_LE(BME280_DIG_P6_REG);
    datos_cal.dig_P7 = (int16)Leer_Datos_U16_LE(BME280_DIG_P7_REG);
    datos_cal.dig_P8 = (int16)Leer_Datos_U16_LE(BME280_DIG_P8_REG);
    datos_cal.dig_P9 = (int16)Leer_Datos_U16_LE(BME280_DIG_P9_REG);

}
```

Figura 9. Código para la calibración del sensor BMP280

Comunicación y aplicación en el Smartphone

El bloque UART es el encargado de realizar la comunicación entre el PSoC y un módulo bluetooth de referencia KC-21(Figurae 10). Como es necesario que la comunicación sea bidireccional, se configura el bloque en modo Full UART (Tx + Rx), a una velocidad igual a la velocidad que tiene programada el modulo bluetooth de 115200 bps. Para poder almacenar y manipular los datos obtenidos por el puerto serial, fue necesario habilitar la interrupción de recepción, que toma el dato recibido y lo almacena en un vector de 4 posiciones, y seguido

realiza la concatenación correspondiente para poder obtener el valor real que se envía desde la aplicación móvil, este valor se almacenara en una variable la cual tendrá como objetivo actualizar el valor de delay que va a tener el envío de datos del PSoC hacia el Smartphone.



Figurae 10. Módulo bluetooth KC-21

Para el desarrollo de la de la aplicación móvil para la estación meteorológica se usó una aplicación ya existente en el Play Store llamada Bluetooth Electronics. Esta aplicación permite programar de manera muy sencilla sub-aplicaciones internas que hagan uso del bluetooth del teléfono.

Inicialmente la aplicación desarrollada permite configurar la conexión entre el modulo bluetooth KC-21 y el Smartphone, una vez conectado el Smartphone al dispositivo bluetooth se puede avanzar al menú que contiene las diferentes ventanas de visualización de los sensores (Figura 11).

Figura 11. Ventanas de visualización de los sensores

5. Conclusiones

Se llegó un prototipo funcional de estación meteorológica modular, de bajo costo, que posee grandes ventajas respecto a los sistemas comerciales, ya que de acuerdo a las variables que se requieran medir, el sistema se puede adaptar para incorporar los sensores que sean necesarios.

De acuerdo a la investigación y lineamientos presentados por la OMM, se definieron las variables prioritarias para medición de las condiciones meteorológicas en el diseño del prototipo de estación meteorológica, además, con base en los requerimientos investigados, se seleccionaron los sensores adecuados para la recolección de datos y la metodología de medición recomendada por la OMM.
El diseño obtenido de estación meteorológica permite adaptar un amplio rango de módulos de comunicaciones para la trasmisión de datos, lo que ofrece una ventaja respecto a los sistemas comerciales, ya que de acuerdo al ambiente de medición y la distancia, se puede adaptar módulos de un precio menor y así reducir costos de implementación

6. Acknowledgments

The authors would like to thank to the Universidad Distrital Francisco José de Caldas and the LASER research group that supported the development and testing of the project.

7. Referencias

[1] M. G. Elizondo, E. Jurado, S. Gonzáles, Ó. Aguirre, J. Jiménes, and J. Navar, “Cambio climático mundial: origen y consecuencias,” *Cienc. uanl*, vol. VI, pp. 377–385, 2003.

[2] IDEAM. Catálogo Nacional de Estaciones del IDEAM. 2018. https://www.datos.gov.co/Ambiente-y-Desarrollo-Sostenible/Cat-logo-Nacional-de-Estaciones-del-IDEAM/hp9r-jxuu/data

[3] J. F. Boshell V., “Informe de Consultoría: Gestión De Información Agroclimática En Colombia,” *Programa Adapt. al Cambio Climático en la Región Andin.*, 2012.

[4] J. J. Perfetti, Á. Balcázar, A. Hernández, and J. Leibovich, *Políticas para el desarrollo de la agricultura en Colombia*, Primera ed. Bogotá D.C.: SAC y Fedesarrollo, 2013.

[5] Banco Interamericano de Desarrollo - BID and Comisión Económica para América Latina y el Caribe - Cepal, “Valoración de daños y pérdidas: Ola invernal en Colombia 2010-2011,” 2012.

[6] J. L. de Francisco, “Diseño de una estación meteorologica autónoma gestionada mediante mensajes SMS,” 2009.

[7] C. R. Mejía, “Sistema automático de monitoreo remoto de variables meteorológicas,” 2013.

[8] W. Robledo, H. Crespo, and G. Guillén, “Centro de monitoreo de parámetros meteorológicos basado en dispositivos móviles.,” *revistatecnologiadigital.com*, vol. 1, no. 1, pp. 11–26, 2011.

[9] C. E. Parada V. and I. A. Rivero C., “Diseño de una red de Estaciones Metereológicas de Bajo Costo.” Escuela de Ingeniería en Telecomunicaciones, Caracas, 2011.

[10] F. Rojas, “Diseño y construcción de un controlador para estación meteorológica con transmisión de datos a un computador,” 2008.

[11] R. M. Rodríguez, A Benito, A. Portela. Metereología y Climatología. 2004. Fundación española para la ciencia y la tecnología. https://cab.inta-csic.es/uploads/culturacientifica/adjuntos/20130121115236.pdf

[12] Oganización Mundial Metereológica (OMM), *Guía de Instrumentos y Métodos*, 6ta ed., vol. 3. 1996.

[13] Honeywell, Sensing. (2010). [Online]. Available: https://sensing.honeywell.com/honeywell-sensing-hih4000-series-product-sheet-009017-5-en.pdf

[14] Raigo instrumentos. Todo sobre pluviómetros. Agosto 2017. https://www.raig.com/noticias/todo-sobre-pluviometros

[15] F. Xavier and M. de Osés, *Meteorología aplicada a la navegación*. 2010.

[16] Bosch Sensortec .BMP280 Digital Pressure Sensor. Enero 2018. https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BMP280-DS001-19.pdf

Ingeniería y Automatización, Capítulo 38, pp. 467 - 480.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Sistema de Prueba de Sensores Analógicos y Digitales utilizando un PSoC5LP

César A. Perdomo Ch. *, Julián R. Camargo L.
Engineering Faculty, Universidad Distrital Francisco José de Caldas, Bogotá D.C., Colombia;
jcamargo@udistrital.edu.co, cperdomo@udistrital.edu.co

Abstract
Objectives: Diseñar e implementar un sistema de prueba de sensores analógicos y digitales haciendo uso del sistema programable PSoC (Programmable System On Chip) fabricado por la empresa Cypress Semiconductor desde el año 2002. **Methods/Analysis:** A cada sensor se le realiza una prueba de funcionamiento de acuerdo con sus características, medición de voltajes para los sensores analógicos o lectura de datos para el caso de los sensores digitales, esa prueba determinará si el sensor está o no operando adecuadamente, se deberá mostrar en la LCD el valor medido por el sensor y el resultado final de la prueba. **Findings:** Con este documento se busca mostrar la gran utilidad de la característica única de reconfiguración que tiene el sistema PSoC5LP. **Improvements:** El sistema diseñado e implementado puede mejorarse realizando algoritmos mas eficientes que generan una respuesta del sistema mucho más rápida.

Keywords: PSoC, brújula digital, sensor de efecto Hall, sensor de distancia, sensor de humedad

1. Introduction

Un sensor es un dispositivo eléctrico y/o mecánico que convierte magnitudes físicas (luz, magnetismo, presión, etc.) en valores medibles de dicha magnitud. Esto se realiza en tres fases:

- Un fenómeno físico a ser medido es captado por un sensor, y muestra en su salida una señal eléctrica dependiente del valor de la variable física.
- El sensor dispone de una circuitería que transforma y/o amplifica la señal eléctrica y ésta es modificada por un sistema de acondicionamiento de señal, para generar una salida que normalmente es voltaje.
- Cuando la señal de salida del sensor es digital, internamente la señal eléctrica pasa a un conversor A/D el cual transforma la señal de tensión continua en una señal discreta.

En este documento se muestra el diseño de un sistema de prueba de sensores analógicos y digitales haciendo uso de un PsoC5LP. El sistema incluye un teclado que permite seleccionar por medio de un menú mostrado en una LCD el sensor al cual se le va a realizar la prueba. A cada sensor se le realiza una prueba de funcionamiento de acuerdo con sus características, esa prueba determina si el sensor está o no operando adecuadamente, se muestra además en la LCD el valor medido por el sensor y el resultado final de la prueba.

Los sensores a los cuales se le puede realizar la prueba son los siguientes:
Sensor de Humedad. Ref. HIH-4000-002
Sensor efecto Hall analógico. Ref. A3112/A3212
Sensor efecto Hall digital. Ref. OH090U
Sensor de luz. Ref. GA1A2S100SS
Sensor de Distancia. Ref. GP2Y0A021YK
Sensor de presión. Ref. MPXV5004DP

Sensor de corriente. Ref. ACS714ELC
Brújula digital. CMPS10.

2. Fundamentos Teóricos

2.1 Humedad

La humedad es un fenómeno natural, que se presenta a nivel molecular y se encuentra básicamente relacionada con la cantidad de moléculas de agua presentes en una determinada sustancia, la cual puede estar en estado sólido o gaseoso.

Si bien el grado de concentración de agua en el ambiente, no influye mayormente sobre la vida normal de un ser humano (salvo en el confort), sabemos que sí resulta relevante para ciertos procesos, ya sean químicos, físicos o biológicos. Por ello, los científicos se han visto involucrados en el tema, dada la necesidad de desarrollar un conocimiento sustancial del fenómeno, con el fin de incorporarlo y relacionarlo con los procesos pertinentes, y de esa manera obtener cierto beneficio de ello.

Un sensor de Humedad lo que hace es medir la cantidad de vapor de agua contenida en el aire, en cualquier momento determinado, normalmente es menor que el necesario para saturar el aire. La humedad relativa es el porcentaje de la humedad de saturación, que se calcula normalmente en relación con la densidad de vapor de saturación[1].

$$Relative\ Humidity = \frac{actual\ vapor\ density}{saturation\ vapor\ density} x100\% \quad (1)$$

La unidad más común de densidad de vapor es el gm/m^3. Por ejemplo, si la densidad de vapor actual es de 10 g/m^3 a 20°C comparada con la densidad de vapor de saturación a esa temperatura de 17,3 g/m^3, entonces la humedad relativa es la mostrad en la ecuación (2).

$$R.H. = \frac{10 g/m^3}{17.3\ g/m^3} x100\% = 57.8\% \quad (2)$$

2.2 Efecto Hall

Los sensores de efecto Hall son aquellos que basan su funcionamiento en el principio de interacción entre campos magnéticos y cargas eléctricas, es decir, hacen variar la diferencia de potencial en un material conductor o semiconductor al ser atravesado por un campo magnético.

Las señales que proporcionan pueden ser tanto analógicas como digitales, por lo que se utilizan mucho como conmutadores binarios. Entre sus aplicaciones más importantes podemos encontrar finales de carrera, detectores de presencia, detectores de posición en cadenas de montaje, medición angular, sistemas ABS de vehículos, etc.

El efecto de Hall fue descubierto en el año 1879 y se manifiesta de manera que cuando entre dos puntos de un conductor existe la misma tensión, al colocarlo en un campo magnético exterior (B) aparece una diferencia de tensión (VH tensión de hall) que es proporcional a la corriente y a la inducción del campo magnético exterior (Figure 1) . Es por ello que son muy utilizados en medida de campos magnéticos y de corrientes eléctricas[2].

Ingeniería y Automatización, Capítulo 38, pp. 467 - 480.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

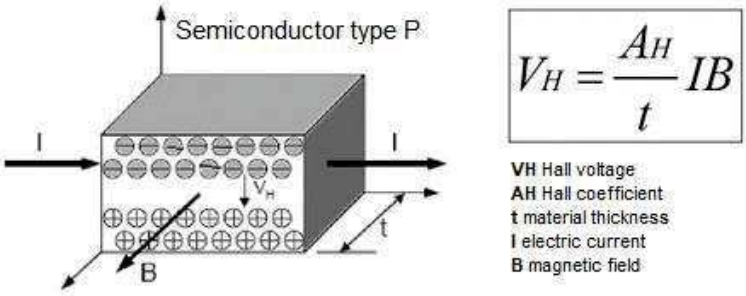


Figure 1. Efecto Hall[2]

Los sensores de Efecto Hall permiten medir la movilidad de una partícula cargada eléctricamente (electrones, lagunas, etc), los campos magnéticos (teslámetros), la intensidad de corrientes eléctricas (sensores de corriente de efecto Hall), permiten la elaboración de sensores o detectores de posición sin contacto, utilizados particularmente en el automóvil, para detectar la posición de un árbol giratorio (árbol de levas, caja de cambios, paliers, etc.), se encuentran también sensores de efecto Hall bajo las teclas de los teclados de los instrumentos de música modernos (órganos, órganos digitales, sintetizadores) evitando así el desgaste que sufren los contactos eléctricos tradicionales, se encuentran sensores de efecto Hall en el codificador de un motor de CD, los motores de Efecto Hall (HET) son aceleradores de plasma de gran eficacia.

2.3 Sensor de luz ambiental

El sensor de luz ambiental está compuesto por fotodiodos (Figure 2), el fotodiodo es un dispositivo electrónico que responde al cambio en la intensidad de luz, los fotodiodos generan una pequeña corriente proporcional al nivel de iluminación que reciben[2].

Los sensores de luz tienen aplicaciones en circuitos de atenuación automática, Automatización de equipos de oficina, Paneles táctiles, Equipos audiovisuales, Equipos para el hogar.

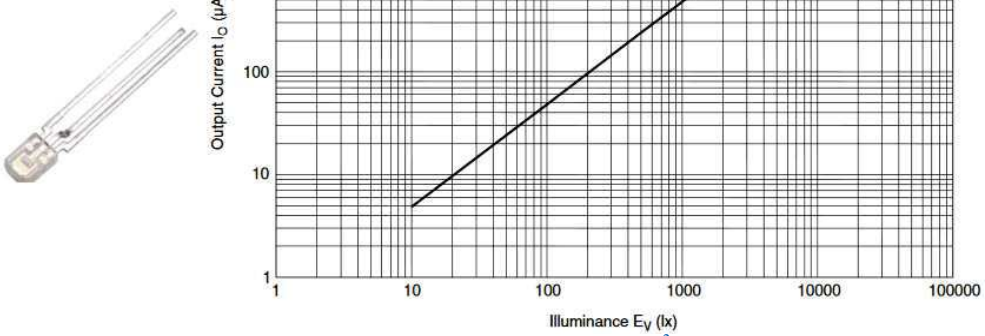


Figure 2. Sensor de luz ambiental[3]

2.4 Sensor medidor de distancias por infrarrojos

Los sensores de infrarrojos son dispositivos fotoeléctricos, emisores y receptores, cuyo espectro de trabajo se encuentra en la longitud de onda infrarroja, entre los 700 y 900 nanómetros. Generalmente asocian una modulación de emisión-recepción[2]. En la Figure 3 se observa un modelo del funcionamiento de un sensor medidor de distancia por infrarrojo.

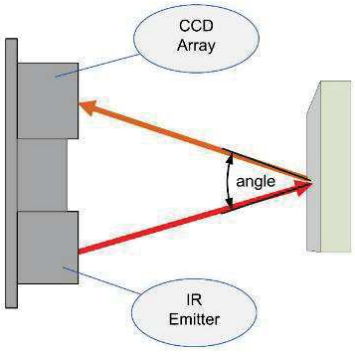


Figure 3. Modelo del funcionamiento del sensor IR[4]

2.5 Sensor de presión

Para medir la presión se utilizan sensores dotados de un elemento específico sensible a la presión, el cual proporciona una señal eléctrica al variar dicha presión, además de un transductor o elemento acondicionador que facilita la medida de la deformación producida[2].

El valor de la presión se mide en función de determinados parámetros, como por ejemplo la presión diferencial, la cual es: la presión medida que representa la diferencia entre dos presiones absolutas (P1-P2).

Aplicaciones de este tipo de sensores son: control de motobombas, robótica, indicadores de niveles, barómetros, altímetros, etc.

2.6 Brújula magnética de estado sólido**.**

El principal componente dentro de las brújulas de estado sólido es el sensor de campo magnético terrestre. Existe un buen número de clases de sensores de este tipo, pero debido a lo extenso y complejo de esta temática, este documento se limita a dar una breve descripción de dos clases de estos sensores, los MR. (Magneto-Resistiva) y los MI (Magneto-Inductiva).

Los sensores MR están basados en el efecto magneto-resistivo, que ocurre en metales ferro magnéticos de transición. En estos materiales, la resistencia eléctrica es una función del ángulo entre la corriente eléctrica y la dirección de magnetización. Entonces, si un campo magnético externo cambia la dirección de la magnetización, la resistencia eléctrica también, permitiendo a estos materiales ser usados como transductores en los sensores magnéticos[5].

Los sensores MI emplean un solenoide por cada eje, esta bobina sirve como elemento inductivo en un simple oscilador LR. La inductancia de esta bobina es influenciada por la componente del campo magnético paralelo al eje de la bobina. Entonces, la frecuencia de oscilación cambia con el campo magnético, lo que les permite también ser usados como transductores en los sensores magnéticos[6].

3. Desarrollo y programación

Para el desarrollo del proyecto se siguieron las recomendaciones dadas en temas de conectividad, por lo cual se configuró un único pin para hacer las lecturas de los sensores, en este pin se configuró un ADC a 10 bits. A continuación, se muestra en la Figure 4 la configuración del ADC[7].

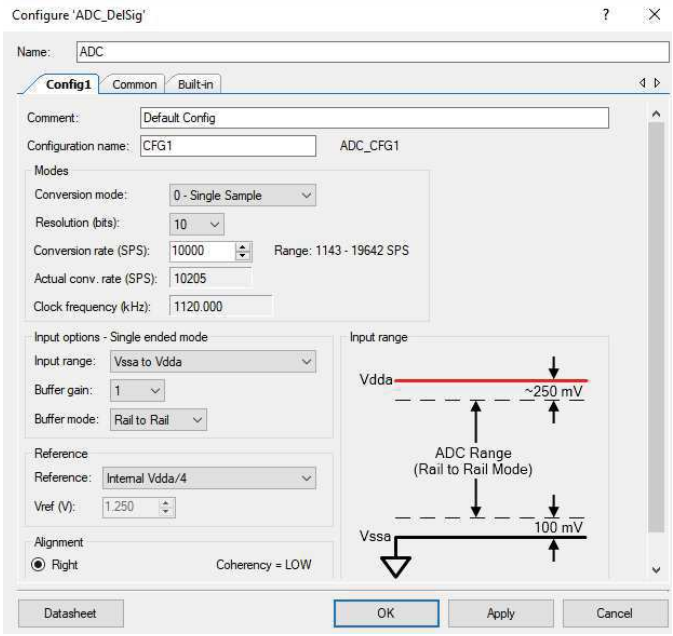


Figure 4. Ventana de configuración del ADC

Para logar un dato más estable de parte del ADC en la mayoría de los sensores se hace primero una toma de 60 muestras y sobre esta se halla el valor promedio que es el que se va a usar para establecer el valor de la unidad física medida. Ahora bien, en cuanto a la caracterización y tratamiento interno que se le dio a los sensores, hubo que realizar un trabajo especial en cada uno de ellos, por lo cual se procede a explicar algunos de ellos por separado.

3.1 Sensor de humedad HIH-4000

Es un sensor cuyo fabricante es Honeywell, revisando el manual de usuario se obtiene la respuesta que presenta este sensor en la Figure 5.

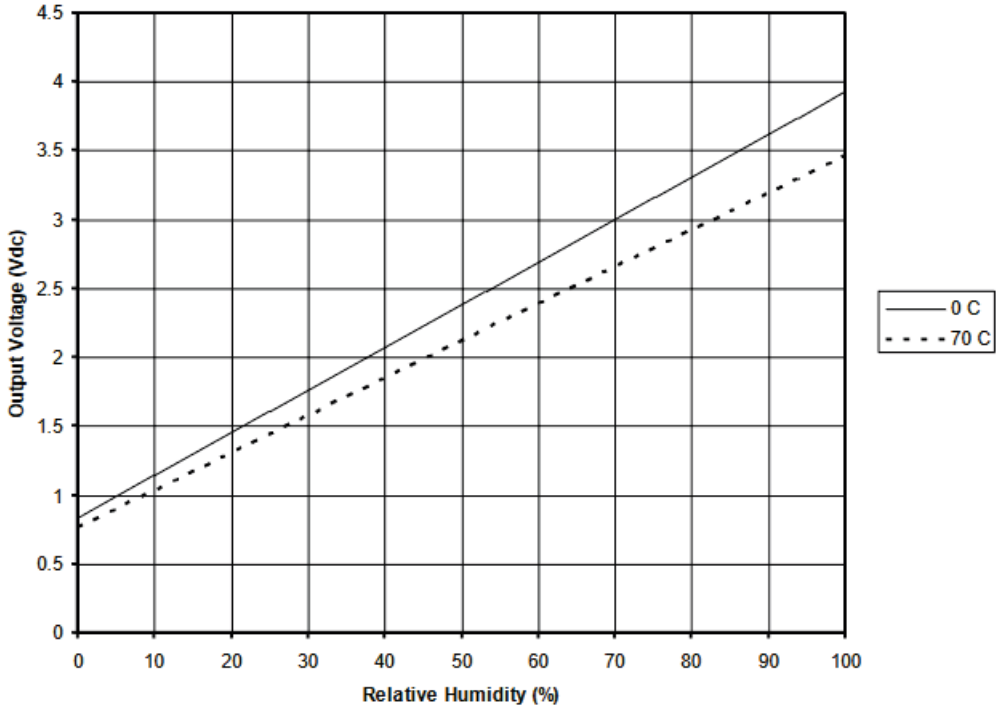


Figure 5.Respuesta del señor de humedad HIH-4000[8]

Como se observa la salida es lineal, pero presenta una variación en la pendiente de la recta la cual depende de la temperatura ambiente sobre la que se encuentre el sensor. Para poder hacer una caracterización óptima de la respuesta del sensor se utilizó como equipo de referencia y patrón de medida el Multifunction Environment Meter CEM-DT8820 que realiza mediciones entre del 25%RH to 95%RH with 0.1%RH resolution and fast time response.

Se procedió entonces a realizar la respectiva medición de la humedad ambiente presente el lugar en donde se estaba haciendo la caracterización. Una vez estabilizada la medición se tomó nota de la medida dada por el equipo patrón el cual marcaba 58% de humedad relativa, por otra parte, se midió el voltaje de salida del sensor el cual daba un valor de 2,27V. Este sería un punto de la recta, pero para poder generar la respectiva ecuación se precisa tener 2 puntos. Para el otro punto se tomó como referencia el valor en 0% de humedad el cual es casi el mismo para las diferentes temperaturas y que según el manual de usuario está en 0.826V.

Ahora bien, ya con los dos puntos se procede a hallar la ecuación de la recta, pero antes para poder usar la ecuación dentro del PSoC se tienen que pasar los valores de voltaje a su valor en bits. Como el ADC se configuró a 10 bits y de 0 a 5V esto da un valor de 4.8828 mV por bit, usando estos valores los puntos para hallar la ecuación se presentan en la Table 1.

Table 1. Valores para cálculo de ecuación característica del sensor

	Valor en Humedad Relativa (%) (eje y)	Valor en voltios	Valor en Bits (eje x)
Punto 1	0	0.826	169
Punto 2	58	2.27	466

Reemplazando los valores de la Table 1 en la ecuación característica de la línea recta se obtiene lo mostrado en la ecuación (3).

$$y = \frac{58}{297}(x - 169) \tag{3}$$

Debido a que el PSoC es más eficiente trabajando con números enteros que con decimales, no se simplifica y esta es la ecuación que se usará para hallar la humedad relativa (y) según el valor dado por el ADC al medir el sensor de humedad (x).

El código que se usó en el PSoC es mostrado en la Figure 6, en donde se resalta la operación que calcula el valor de la humedad relativa:

```
ADC_StartConvert();
ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
volt = ADC_Read16();
buffer[pos] = volt;
pos++;
if (pos == 60){
   LCD_ClearDisplay();
   LCD_Position(0,0);
   LCD_PrintString("Esperado: 60%");
   LCD_Position(1,0);
   LCD_PrintString("Medido: ");
   prom = 0;
   for (tmp = 0; tmp < 60; tmp++)
      prom += buffer[tmp];
   prom /= 60;

   prom  = prom - 169;
   prom *= 58;
   prom /= 297;

   LCD_PrintNumber(prom);
   LCD_PrintString("%");

   pos = 0;
   CyDelay(50);
}
```

Figure 6. Código implementado en el PSoC5LP para probar el sensor de temperatura

3.2 Sensor de Efecto Hall Analógico

Para este sensor también se utilizó el manual de usuario como referencia para hallar la ecuación que correspondiente a la respuesta dada por el sensor, los datos obtenidos puede verse en la Table 2. Para este caso no se contó con un equipo que pudiera servir como referencia para las medidas dadas, por lo cual únicamente se usó la información que se encontraba en el manual de usuario.

Table 2. Valores para cálculo de ecuación característica del sensor de efecto Hall

	Valor en Gauss (eje y)	Valor en voltios	Valor en Bits (eje x)
Punto 1	0	2.5	511
Punto 2	1000	4	819

Reemplazando los valores de la Table 2 en la ecuación característica de la línea recta se obtiene lo mostrado en la ecuación (4).

$$y = \frac{1000}{307}(x - 512) \tag{4}$$

El código que se usó en el PSoC es mostrado en la Figure 7, en donde se resalta la operación que calcula el valor medido por el sensor Efecto Hall analógico:

```
ADC_StartConvert();
ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
volt = ADC_Read16();
buffer[pos] = volt;
pos++;
if (pos == 60){
   LCD_ClearDisplay();
   LCD_Position(0,0);
   LCD_PrintString("Esperado: 22G");
   LCD_Position(1,0);
   LCD_PrintString("Medido: ");
   prom = 0;
   for (tmp = 0; tmp < 60; tmp++)
      prom += buffer[tmp];
   prom /= 60;

   gauss  = prom - 512;
   gauss *= 1000;
   gauss /= 307;

   LCD_PrintNumber(gauss);
   LCD_PrintString("G");
   pos = 0;
   CyDelay(50);
}
```

Figure 7. Código implementado en el PSoC5LP para probar el sensor de efecto Hall analógico

3.3 Sensor de luz GA1A2S100SS

A diferencia de los otros sensores hasta ahora trabajados, la respuesta de este sensor no es lineal sino logarítmica tal como lo muestra la Figure 2.

Teniendo esta gráfica como referencia se procedió a caracterizar el sensor. Para esta labor se utilizó como fuente de luz un bombillo de luz incandescente y para controlar la cantidad de luz incidente en el sensor se usó un tubo oscuro en donde en una punta estaba el sensor y por la otra entraba el bombillo y controlando la profundidad del bombillo se podía variar la luz incidente sobre el sensor. Adicionalmente para poder comprobar la lectura que genera el sensor de luz se usó el Multifunction Environment Meter CEM-DT8220 el cual también permite medir lúmenes.

Se plantea usar el método de rectas sucesivas, para lo cual modeló toda la curva como una suma de rectas pequeñas y de esta forma se logra hacer el cálculo más sencillo para el PSoC, los datos obtenidos pueden verse en la Table 3.

Table 3. Datos medidos por medidor CEM-DT8220

Lúmenes(y)	Bits (x)		Lúmenes(y)	Bits (x)
710	50		6340	450
1300	100		6810	500
1930	150		7540	550
2600	200		8220	600

3230	250		9800	650
3960	300		10530	700
4640	350		11620	750
5310	400		20000	770

Una vez obtenida la tabla la implementación del sistema que calcula el valor en lúmenes de cualquier valor medido se hizo ubicando primero el valor medido dentro de algunos de los valores en bits previamente dados y después se usan los datos de los valores de frontera anterior y posterior y usando la ecuación de la recta para dos puntos obtiene un valor bastante aproximado al valor real.

El código implementado puede verse en la Figure 8.

```
LumX[17]={0, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600,  650,  700,  750,  770};
//bits
lumY[17]={0,710,1300,1930,2600,3230,3960,4640,5310,6340,6810,7540,8280, 9800,10530,11620,20000};
//lumens

ADC_StartConvert();
ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
volt = ADC_Read16();
buffer[pos] = volt;
pos++;
if (pos == 60){
    LCD_ClearDisplay();
    LCD_Position(0,0);
    LCD_PrintString("Esperado: 2900L");
    LCD_Position(1,0);
    LCD_PrintString("Medido: ");
    prom = 0;
    for (tmp = 0; tmp < 60; tmp++)
        prom += buffer[tmp];
    prom /= 60;
    /* //ecuación real
    lumens  = prom * 0.00427374;
    lumens  = exp(lumens);
    lumens *= 692.841538;
    */
    if(prom > lumX[0] && prom <= lumX[13]){
        tmpLong = 0;
        for (tmp = 0; tmp < 16; tmp++){ //Recorre el arreglo
            if(prom >= lumX[tmp] && prom < lumX[tmp+1]){ //ubica el rango apropiado y calcula
                //y=y1+(y2-y1)[(x-x1)/(x2-x1)] operación a realizar
                tmpLong  = prom-lumX[tmp]; //x-x1
                tmpLong *= (lumY[tmp+1] - lumY[tmp]); //(y2-y1)*(x-x1)
                tmpLong /= (lumX[tmp+1] - lumX[tmp]); //(y2-y1)*(x-x1)/(x2-x1)
                tmpLong += lumY[tmp];                 //y1 + (y2-y1)*(x-x1)/(x2-x1)
            }
        }

        LCD_PrintNumber(tmpLong);

    } else {
        LCD_PutChar('0');
    }

    LCD_PrintString("L");

    //LCD_PrintNumber(prom);
    pos = 0;
    CyDelay(50);
}
```

Figure 8. Código implementado en el PSoC5LP para probar el sensor de luz

3.4 Sensor de Distancia

Este sensor al igual que el de luz no es lineal, sino que presenta una curva de forma logarítmica en su respuesta. Teniendo en cuenta la experiencia anterior se procedió a realizar el mismo procedimiento para la caracterización. Se usó como patrón de medida una cinta métrica en una mesa, la cual tenía marcado los valores en centímetros. Se puso el sensor en una punta y en la otra una tabla sobre la cual debería de rebotar el haz de luz. Para esta caracterización usó el PSoC para que nos arrojara el valor ya en bits. El resultado se puede observar en la Table 4.

Table 4. Datos de distancia medidos

Distancia en mm(y)	Bits (x)	Distancia en mm(y)	Bits (x)
150	556	500	241
200	506	550	219
250	446	600	203
300	391	650	190
350	339	700	173
400	229	750	166
450	265	800	159

De esta manera se caracteriza la curva como una sucesión de rectas haciendo mucho más eficiente el código para el PSoC. El resultado de la implementación se muestra en la Figure 9.

```
uint16 disX[14]={556,506,446,391,339,299,265,241,219,203,190,179,166,159};//bits
uint16 disY[14]={150,200,250,300,350,400,450,500,550,600,650,700,750,800};//mm

ADC_StartConvert();
ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
volt = ADC_Read16();
buffer[pos] = volt;
pos++;
if (pos == 60){
    LCD_ClearDisplay();
    LCD_Position(0,0);
    LCD_PrintString("Esperado: 300mm");
    LCD_Position(1,0);
    LCD_PrintString("Medido: ");
    prom = 0;
    for (tmp = 0; tmp < 60; tmp++)
        prom += buffer[tmp];
    prom /= 60;
    //LCD_PrintNumber(prom);

    if(prom < disX[0] && prom >= disX[13]){
        tmpLong = 0;
        for (tmp = 0; tmp < 13; tmp++){
            if(prom <= disX[tmp] && prom >= disX[tmp+1]){
                //y=y1+(y2-y1)[(x-x1)/(x2-x1)]
                tmpLong  = prom-disX[tmp]; //x-x1
                tmpLong *= (disY[tmp+1] - disY[tmp]); //(y2-y1)*(x-x1)
                tmpLong /= (disX[tmp+1] - disX[tmp]); //(y2-y1)*(x-x1)/(x2-x1)
                tmpLong += disY[tmp];                 //y1 + (y2-y1)*(x-x1)/(x2-x1)
            }
        }

        LCD_PrintNumber(tmpLong);

    } else {
        LCD_PutChar('0');
    }

    LCD_PrintString("mm");
    pos = 0;
    CyDelay(50);
}
```

Figure 9. Código implementado en el PSoC5LP para probar el sensor de distancia

3.5 Sensor de presión

Para este sensor se usó también la gráfica dada por el fabricante en el manual de usuario la cual se puede ver en la Figure 10.

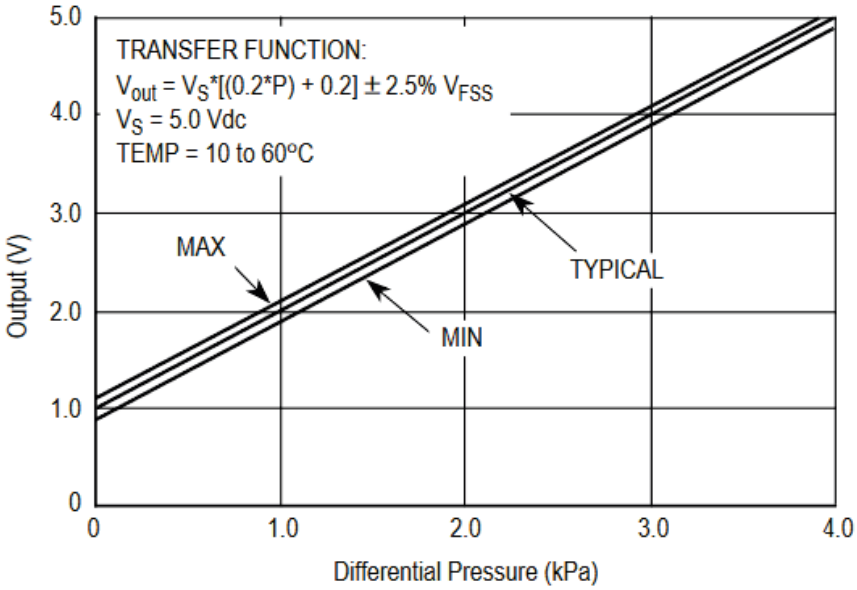


Figure 10. Respuesta del señor de presión MPXV5004DP[9]

Como se puede observar su respuesta es lineal y casi invariable. Para hallar la ecuación de la recta se usó el mismo procedimiento que en los sensores anteriores. Se determinaron dos puntos de referencia mostrado en la Table 5.

Table 5. Valores para cálculo de ecuación característica del sensor de presión

	Valor en pa (eje y)	Valor en voltios	Valor en Bits (eje x)
Punto 1	0	1	204
Punto 2	3000	4	819

Reemplazando los valores de la Table 5 en la ecuación característica de la línea recta se obtiene lo mostrado en la ecuación (5).

$$y = \frac{1500}{307}(x - 204) \quad (45$$

Se siguió el mismo procedimiento que en el caso de los sensores anteriores para la implementación del código en el PSoC5LP.

3.5 Sensor de compass o brújula electrónica CMPS10

A diferencia de los otros sensores este es un sensor completamente digital, por lo cual su implementación no podía ser igual que la de los otros sensores. Según el manual de usuario dado por el fabricante este sensor puede trabajar de 3 modos distintos: UART, I2C o PWM. Para este caso se usó el sensor en el modo UART, ya que su implementación es la más sencilla y da una mejor precisión que usando el método de PWM.

Siguiendo las recomendaciones del manual de usuario se configuro un bloque UART (puerto serial) en el PSoC5LP a 9600 baudios, 8 bits de datos y 2 bits de parada como se muestra la Figure 11.

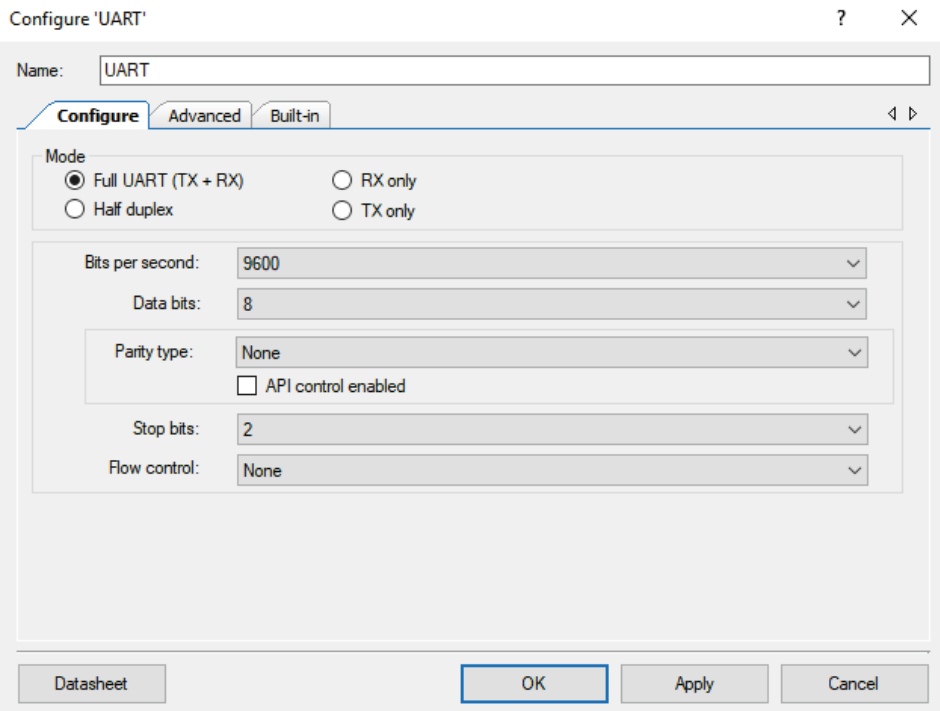


Figure 11. Configuración del bloque UART del PSoC5LP

Los comandos que soporta en el modo UART de este sensor se presenta en la Table 6.

Table 6. Comandos soportados por la brújula electrónica CMPS10 en el modo UART[10]

Command	Name	Bytes returned	Returned data description
0x11	GET VERSION	1	Software version
0x12	GET ANGLE 8 BIT	1	Angle as a single byte 0-255
0x13	GET ANGLE 16 BIT	2	Angle as two bytes, high byte first 0-3600
0x14	GET PITCH	1	Pitch angle +/- 0-85°
0x15	GET ROLL	1	Roll angle +/- 0-85°
0x21	GET MAG RAW	6	Raw magnetic data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x22	GET ACCEL RAW	6	Raw accelerometer data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x23	GET ALL	4	angle high, angle low (0-3600), pitch (+/- 0-85), roll (+/- 0-85)
0x31	CALIBRATE EN1	1	returns ok (0x55)
0x45	CALIBRATE EN2	1	returns ok (0x55)
0x5A	CALIBRATE EN3	1	returns ok (0x55)
0x5E	CALIBRATE	1	returns ok (0x55)
0x6A	RESTORE 1	1	returns ok (0x55)
0x7C	RESTORE 2	1	returns ok (0x55)
0x81	RESTORE 3	1	returns ok (0x55)
0xA0	BAUD 19200	1	returns ok (0x55)
0xA1	BAUD 38400	1	returns ok (0x55)

Para este caso se usó el comando 0x13 el cual devuelve el valor del ángulo medido como un numero de 16 bits. Hay que tener en cuenta que este ángulo si bien esta dado en grados el sistema lo envía con una precisión de 0.1 grados por lo cual un valor de 1786 debe interpretarse como 178.6°.

Teniendo esto claro en la implementación del código se debe enviar el comando 0x13 y esperar 2 bytes de respuesta, luego convertir estos dos bytes al valor en grados con sus respectivos decimales. Así bien la implementación del código se puede ver en la Figure 12.

```
LCD_ClearDisplay();
LCD_Position(0,0);
LCD_PrintString("Esperado: 180.0");
LCD_Position(1,0);
LCD_PrintString("Medido: ");
UART_ClearRxBuffer();
UART_PutChar(0x13);

while(UART_GetRxBufferSize() < 1); // espera los dos Bytes

volt  = UART_GetByte(); //guarda el dato
volt  = volt << 8;
volt += UART_GetByte();

//convierte y muestra el dato
tmp = volt/1000;
LCD_PutChar(tmp+0x30);
volt %= 1000;
tmp = volt / 100;
LCD_PutChar(tmp+0x30);
volt %= 100;
tmp = volt / 10;
LCD_PutChar(tmp+0x30);
LCD_PutChar('.');
tmp = volt % 10;
LCD_PutChar(tmp+0x30);
//LCD_PutChar('°');
```

Figure 12. Código implementado en el PSoC5LP para probar la brújula electrónica CMPS10

4. Conclusiones

Se pudo apreciar que en la actualidad los fabricantes de sensores se han esmerado en lograr entregar una salida lineal y muchas veces compensada en temperatura, lo que la hace muchos más fácil de trabajar en sistemas embebidos.

Los sensores que no son lineales se pueden utilizar técnicas como la de rectas sucesivas que son sencillas de implementar y a su vez son amigables para trabajar en los sistemas embebidos.

Para lograr obtener una mejor caracterización de los sensores es primordial poder tener un patrón de medida que genere un ambiente estable, en algunos sensores es fácil de lograr, pero otros tales como humedad y campo magnético, lograr ese ambiente controlado es bastante complejo.

El uso de los sensores es sin duda una parte importante en la mayoría de desarrollos electrónicos, pues son estos los que le van a permitir al sistema embebido poder saber qué es lo que está pasando a su alrededor y poder tomar las correspondientes acciones al respecto.

5. Acknowledgments

The authors would like to thank to the Universidad Distrital Francisco José de Caldas and the LASER research group that supported the development and testing of the project.

6. References

1. HyperPhysics, Humedad relativa, [Internet]. [Cited 2018 March 16]. Avaible from: http://hyperphysics.phy-astr.gsu.edu/hbasees/Kinetic/relhum.html
2. Serna A, Ros FA, Rico JC. Guía Práctica de Sensores. España: Creaciones Copyright, 2010.

3. Digikey, Sharp. GA1A2S100SS, OPIC™ Linear Output Ambient Light Sensor, Technical specification. [Internet]. 2007. [Cited 2018 March 16]. Avaible from: https://media.digikey.com/pdf/Data%20Sheets/Sharp%20PDFs/GA1A2S100LY,SS.pdf
4. Margolis M. Make an Arduino-Controlled Robot. 1005 Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly Media Inc., 2012.
5. Marek J, Trah H, Suzuki Y, Yokomori I, Queisser HJ. Sensors Applications, Sensors for Automotive Technology Volume 4. Wiley-VCH, 2003.
6. PNI Sensor Corporation. Application Note: Discrete Circuit for Magneto-Inductive Sensors, 2003.
7. Camargo JR, Perdomo CA. Introducción al PSoC5LP: Teoría y aplicaciones prácticas. Bogotá: Ed. Universidad Distrital Francisco José de Caldas; 2016.
8. Honeywell, Sensing. [Internet]. 2010. [Cited 2018 March 16]. Avaible from: https://sensing.honeywell.com/honeywell-sensing-hih4000-series-product-sheet-009017-5-en.pdf
9. NXP, Sensors, Pressure Sensors. [Internet]. 2015. [Cited 2018 March 17]. Avaible from: https://www.nxp.com/docs/en/data-sheet/MPXV5004G.pdf
10. Robot electronics, compass sensors. [Internet]. 2010. [Cited 2018 March 17]. Avaible from: https://www.robot-electronics.co.uk/htm/cmps10ser.htm

Ingeniería y Automatización, Capítulo 39, pp. 481 - 496.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

Control PID a Motor sin Escobillas

Ortega Orta Aldo Augusto, Ley Arteaga Kevin Alfonso, Sosa Sales Jorge, Gómez Guzmán César Alberto y Llamas Negrete Juan Carlos

Universidad Tecnológica de Nayarit
Carretera Federal 200 K.m 9 Xalisco, Nayarit. C.P. 63780.
cealgogu@hotmail.com

Resumen

Este documento muestra el modelado matemático de un motor brushless para aplicaciones de aeromodelismo y control. El sistema es modelado para un control clásico en base a la función de transferencia velocidad/voltaje, tomando como señales de control la posición del motor por giroscopio. El modelado se divide en análisis de circuito eléctrico y mecánico. Se presentan las simulaciones en el software de MATLAB, entorno Simulink para análisis de estabilidad y gráficas. Así como presentando el modelado de sensor para la obtención de la posición del motor.

Palabras clave: Control, motor brushless, modelado matemático, aeromodelismo, MATLAB, función de transferencia.

1. Introducción

Los motores brushless están compuestos por una parte móvil que es el rotor, es donde se encuentran los imanes permanentes, y una parte fija, denominada estator o carcasa, donde se encuentran los bobinados de hilo conductor.

Como su propio nombre indica, brushless quiere decir "sin escobillas". En este tipo de motor la corriente eléctrica pasa directamente por los bobinados del estator o carcasa, por lo tanto, aquí no son necesarios ni las escobillas ni el colector que se utilizan en los brushless. Esta corriente eléctrica genera un campo electromagnético que interacciona con el campo magnético creado por los imanes permanentes del rotor, haciendo que aparezca una fuerza que hace girar al rotor y por lo tanto al eje del motor. [1]

El propósito de este documento es proporcionar una idea clara del control de velocidad de un motor brushless por PID. Cabe mencionar que en este documento se agregan los pasos y protocolos necesarios para la realización del sistema de control. Se enfatiza a lo largo del documento conceptos y demostraciones de los modelos matemáticos elaborados.

El significado de modelo matemático en filosofía de la matemática y fundamentos de la matemática es, sin embargo, algo diferente. En concreto en esas áreas se trabajan con "modelos formales". Un modelo formal para una cierta teoría matemática es un conjunto sobre el que se han definido un conjunto de relaciones unarias y binarias, que satisface las proposiciones derivadas del conjunto de axiomas de la teoría. La rama de la matemática que se encarga de estudiar sistemáticamente las propiedades de los modelos es la teoría de modelos

Una vez obtenido un modelo matemático de un sistema, se usan diversos recursos analíticos, así como computadoras para estudiarlo y sintetizarlo y aplicar un sistema de control al sistema. Este puede variar según sus objetivos y tipo de sistema o planta [2].

Un sistema de control es aquél en el que la salida de este se controla para tener un valor específico o cambiarlo, según lo determina la entrada. De este modo, un control de temperatura en un

sistema de calefacción central en una casa, por ejemplo, puede tener en su entrada un termostato o panel en el que se fija la temperatura requerida y su salida es la temperatura real producida. Esta se ajusta mediante el sistema de control, de modo que se obtenga el valor fijado por la entrada al sistema. [3]

En la teoría de control, a menudo se usan las funciones de transferencia para caracterizar las relaciones de entrada-salida de componentes o de sistemas que se describen mediante ecuaciones diferenciales lineales invariantes en el tiempo. Se comenzará por definir la función de transferencia, para proseguir con el cálculo de la función de transferencia de un sistema de ecuaciones diferenciales. La función de transferencia de un sistema descrito mediante una ecuación diferencial lineal e invariante en el tiempo se define como el cociente entre la transformada de Laplace de la salida (función de respuesta) y la transformada de Laplace de la entrada (función de excitación) bajo la suposición de que todas las condiciones iniciales son cero. Para el motor, tomamos como salida la velocidad del motor, y como entrada e voltaje aplicado a su controlador. [4]

Es importante mencionar que para los cálculos y desarrollo matemático del presenta proyecto es necesario conocer y poder tratar ecuaciones de 1er y 2do orden, así como conocer sus características y cómo resolverlas.

Los sistemas que se representan mediante ecuaciones diferenciales cuyos coeficientes son funciones del tiempo, se denominan sistemas lineales variantes en el tiempo. Un ejemplo de un sistema de control variante en el tiempo es un sistema de control de naves espaciales. (La masa de una nave espacial cambia debido al consumo de combustible).

El controlador PID (Proporcional, Integrador y Derivativo) es el control utilizado debido a su fuerte funcionamiento en sistemas de control de velocidad de precisión. Este método PID es un controlador realimentado cuyo propósito es hacer que el error en estado estacionario, entre la señal de referencia y la señal de salida de la planta, sea cero de manera asintótica en el tiempo, lo que se logra mediante el uso de la acción integral. Además, el controlador tiene la capacidad de anticipar el futuro a través de la acción derivativa que tiene un efecto predictivo sobre la salida del proceso. Su uso extensivo en la industria es tal que el 95% de los lazos de control que existen en las aplicaciones industriales son del tipo PID, donde la mayoría son controladores PI, lo que muestra la preferencia del usuario en el uso de leyes de control muy simples.

Una vez elegido el controlador, la siguiente tarea es determinar los valores de los parámetros del mismo. Estos parámetros son típicamente coeficientes de una o más funciones de transferencia que componente al controlador. Al realizar el diseño ya sea en el dominio del tiempo o de la frecuencia, es importante establecer algunas guías básicas o reglas de diseño. Se debe tener en mente que el diseño en el domino del tiempo normalmente se basa fuertemente en el plano S y en el lugar geométrico de las raíces. En sistemas de control se suelen emplear plantas donde se pone a prueba el comportamiento de los sistemas analizados sin llevarlos a su aplicación directa. [5]

2. Modelado Matemático

El presente proyecto se realizó en una estructura de equilibrio, donde se acopló el motor. Los elementos utilizados fueron:

- Estructura mecánica
- Motor brushless
  - K (Torque)=0.053682 N*m
  - Km (Constante del motor) =0.033057 V/(Rad*s)
  - Ra (Resistencia)=4.188405 Ω
  - La (inductancia)=0.590001 H
  - B (Coeficiente de fricción) =0.000068 (N*m)/(Rad/seg)
  - J (Inercia) =0.000084 $Kg*m^2$
- Variador de velocidad de 30ª ESC300

- Arduino Mega
- Sensor MPU6050
- Batería LiPo 3S

En la estructura se mide el ángulo que forma la fuerza de empuje del motor a cierto voltaje aplicado. Siendo evidente la efectividad del uso de control PID. El control es embebido en una placa de Arduino y el sistema es alimentado por una batería externa. En este proyecto se utilizaron dos fuentes de alimentación para prevenir los ruidos en el sistema causado por los motores.

2.1 Modelado Matemático circuito electromecánico

El modelado del motor brushless realiza mediante el uso de leyes físicas, como el caso de la suma de tensiones eléctricas en una malla, o la suma de fuerzas que se igualan a una fuerza de excitación. El motor brushless se puede representar de la siguiente manera:

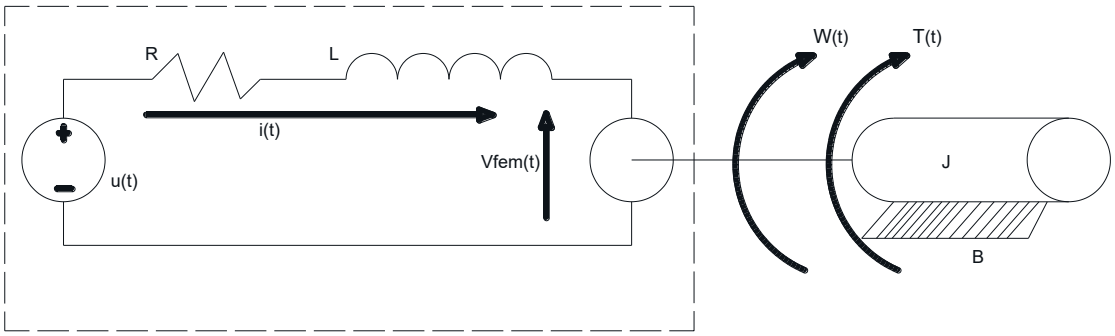


Figura 1 Circuito Electromecánico

Donde:

R: Resistencia óhmica de los embobinados del motor
L: Inductancias de los embobinados del rotor
J: Momento de inercia del motor
B: Coeficiente de fricción viscosa
U: Voltaje de entrada al sistema
W: Velocidad angular del rotor
T: Par del rotor
I: Corriente del sistema
Vfem: Voltaje del motor

Existen otros parámetros constantes que no se vislumbran en el diagrama anterior, estas son utilizadas para hacer vínculo entre la parte mecánica y la parte eléctrica del motor.
Dichos valores escalares son:
KT: Constante mecánica, considerando un flujo magnético constante (También puede ser nombrada como “Constante de campo”)
KE: Constante eléctrica, considerando un flujo magnético constante (También puede ser nombrada como “Constante de armadura”)
TI: Par de fricción o par inercial

La suma de las tensiones eléctricas que componen la malla eléctrica de la figura 1 se expresa en la ecuación 1:

$$v(t) = Ri(t) + L\frac{di}{dt} + Vfem \tag{1}$$

La suma de fuerzas que hacen girar al rotor respecto al eje simétrico, es igual a la suma de momentos que se oponen al movimiento del mismo, matemáticamente se expresa en la ecuación 2:

$$\tau(t) = B\omega(t) + J\frac{d\omega}{dt} \tag{2}$$

Como se mencionó, existen ciertas ecuaciones matemáticas que vinculan directamente la parte mecánica del sistema con la parte eléctrica del mismo; las ecuaciones 3 y 4 muestran la relación entre el sistema mecánico y eléctrico:

$$\tau(t) = \mathrm{KT} * i(t) \tag{3}$$

$$\mathrm{Vfem}(t) = \mathrm{K}E * \omega(t) \tag{4}$$

Se sustituye la ecuación 4 en la ecuación 1 para obtener la ecuación 5:

$$\mathrm{v}(t) = Ri(t) + L\frac{di}{\mathrm{dt}} + KE * \omega(t) \tag{5}$$

Se aplica la transformada de Laplace a la ecuación 5 para trabajar en función de la frecuencia obteniendo la ecuación 6:

$$\mathcal{L}\left\{v(t) = Ri(t) + L\frac{di}{dt} + KE * \omega(t)\right\} =$$

$$\frac{\mathrm{V}}{\mathrm{s}} = \mathrm{RI}s + \mathrm{L}s\mathrm{I}s + \mathrm{KE}\omega s \tag{6}$$

Se despeja la corriente de la ecuación 6 y se obtiene la ecuación 7:

$$\frac{\frac{\mathrm{V}}{\mathrm{s}} - \mathrm{K}E * \omega s}{R * Ls} = \mathrm{I}(s) \tag{7}$$

Se sustituye la ecuación 6 en la ecuación 3 y se pasa al dominio de la frecuencia para obtener la ecuación 8:

$$\mathcal{L}\left\{KT * i(t) = B\omega(t) + J\frac{d\omega}{dt}\right\} =$$

$$KTIs = B\omega s + Js\omega s \tag{8}$$

Se sustituye la ecuación 7 en 8 y se obtiene la ecuación 9:

$$\mathrm{KT}\frac{\frac{v}{s} - KE * \omega s}{R + Ls} = B\omega s + Js\omega s \tag{9}$$

Se despeja KT* V/s de la ecuación 9 lo que da resultado a la ecuación 10:

$$\mathrm{KT}\frac{\frac{v}{s} - KE * \omega s}{R + Ls} = B\omega s + Js\omega s \tag{10}$$

Se obtiene la relación que existe de la salida con respecto al voltaje de entrada expresado en la ecuación 11:

$$\frac{\omega(s)}{V(s)} = \frac{KT(s)}{(B*Js)(R+Ls)+KT*KE} \tag{11}$$

La ecuación 11 representa la función de transferencia del motor, esta indica la relación existente entre el voltaje que se suministra al motor y la velocidad que obtiene a la salida.

2.3 Modelado sensor giroscopio GY-521

El sensor giroscopio MPU6050 (mostrado en la figura 2), proporciona 6 grados de libertad, los cuales son representados en velocidad *x, y*, y *z*, y aceleración en las mismas coordenadas.
Para la obtención de los datos del sensor se hizo uso de la plataforma Simulink, donde ya se encuentran los bloques asignados para la obtención de los valores de velocidad y aceleración.
Este sensor es utilizado en el proyecto ya que es capaz de proporcionarnos los valores deseados de posición en grados de la estructura física del proyecto, por esta razón el sensor MPU6050 es utilizado en este proyecto.
En la figura 3 se muestran los bloques obtenidos de la librería arduino para simulink. La figura presenta el bloque para la obtención de velocidades del módulo de giroscopio en los ejes *x, y* y *z*, y el bloque para la obtención de aceleraciones en los mismos ejes respectivamente.

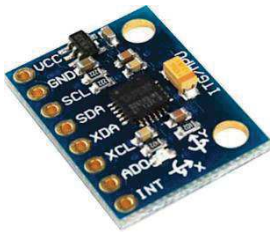


Figura 2 Giroscopio MPU6050

Figura 3 Bloques de sensor MPU6050

Se realiza la conexión del sensor y el arduino como se muestra en la figura 4; para la obtención de datos por parte del sensor. Una vez hecha la conexión se realiza la programación en simulink de los bloques para visualizar los valores, tal como se observa en la figura 5. Ya elaborados los diagramas de conexión y programación, se ejecuta y se obtienen las gráficas de la figura 6. Posteriormente se elabora la calibración del eje x en velocidad para tener la posición del ángulo, la programación se presenta en la figura 7 que representa los diagramas de bloque necesarios para la obtención del ángulo real en el eje X. La señal de salida del sensor se integra para la obtención de la posición, esta es multiplicada por una ganancia que nos reduce la señal en un rango de -90° a 90°.

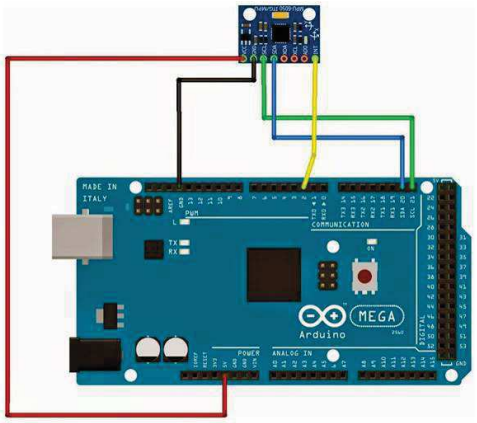


Figura 4 Conexión MPU6050 con Arduino

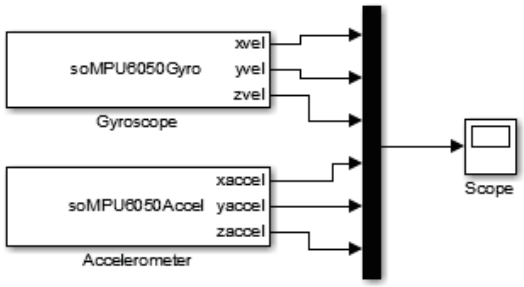


Figura 5 Programación de obtención de datos del sensor MPU6050

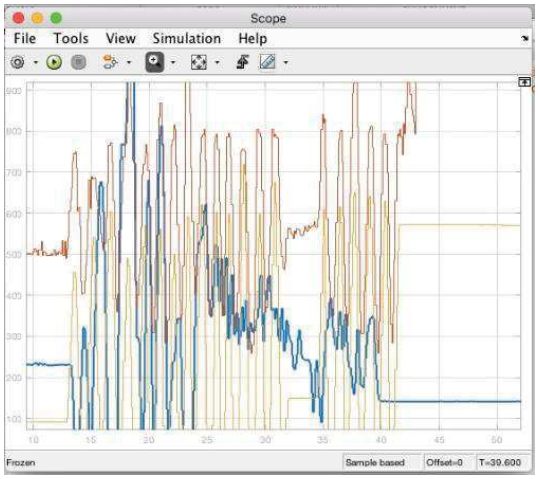


Figura 6 Graficas de sensor MPU6050

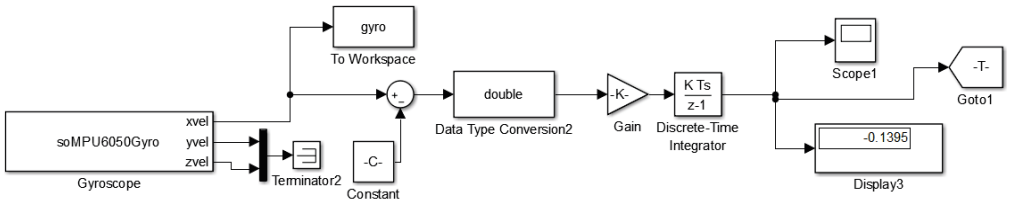


Figura 7 Ángulo de Velocidad en X

La programación de la figura 7 se realiza de acuerdo a la ecuación 12 proporcionada por el datasheet del sensor MPU6050 para obtener el ángulo de inclinación del dispositivo en el eje x. Se incluyen bloques de conversión numérica para trabajar con números enteros.

$$\theta(x) = \int vel\ (x)\ dt \tag{12}$$

La figura 8 es el osciloscopio que muestra la señal obtenida por parte de la programación de la figura 7. Las curvas de la figura 8 se obtienen haciendo girar el sensor en el eje x.

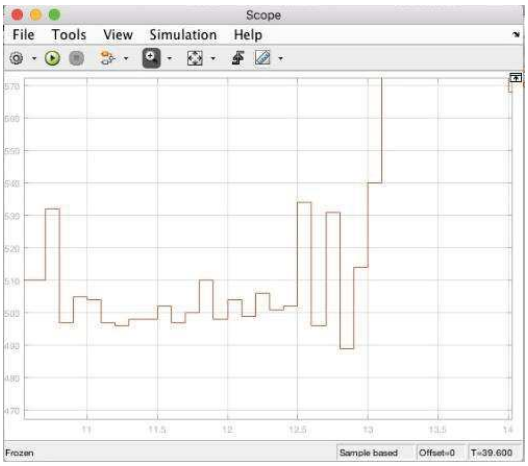


Figura 8 Gráfica de Posición en Velocidad X

Posteriormente se realiza la ecuación 13 en programación simulink obtenida en el datasheet del sensor MPU6050 para la obtención de la posición a partir de las lecturas proporcionadas por el giroscopio. La programación a bloques se presenta en la figura 9, donde se agregaron bloques de conversión numérica en los pines de salida de datos para trabajar con valores enteros.

$$\theta(x) = \tan^{-1}\frac{y\ accel}{z\ accel} \tag{13}$$

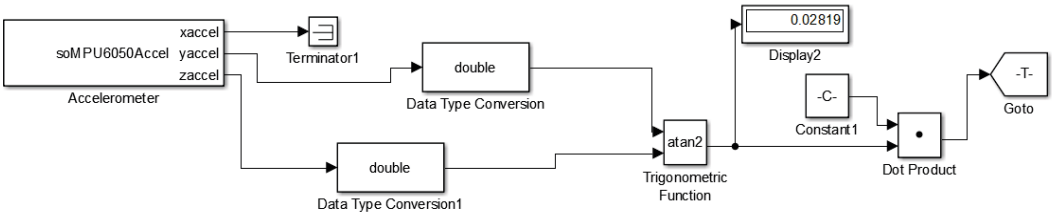


Figura 9 Ángulo por aceleración en YZ

La gráfica de la figura 10 muestra el comportamiento de los valores respecto a la posición del sensor. La curva de la gráfica se obtiene girando el sensor en el eje x. De acuerdo al datasheet del fabricante del sensor MPU6050, la ecuación 13 proporciona el ángulo de inclinación en el eje x.

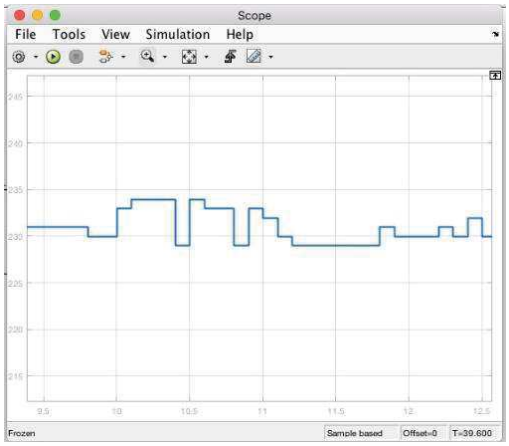


Figura 10 Gráfica de Posición en YZ

La figura 11 muestra el esquema de programación del filtro complementario que tiene como fin filtrar las medidas de posición obtenidas anteriormente en las figuras 7 y 9. La lógica de programación se desarrolla de acuerdo a la fórmula de la ecuación 14 obtenida del datasheet del sensor MPU6050.

*Valor del filtro complementario = (tm)*posición en x de giroscopio + (1-alfa) *posición en x de acelerómetro.* *(14)*

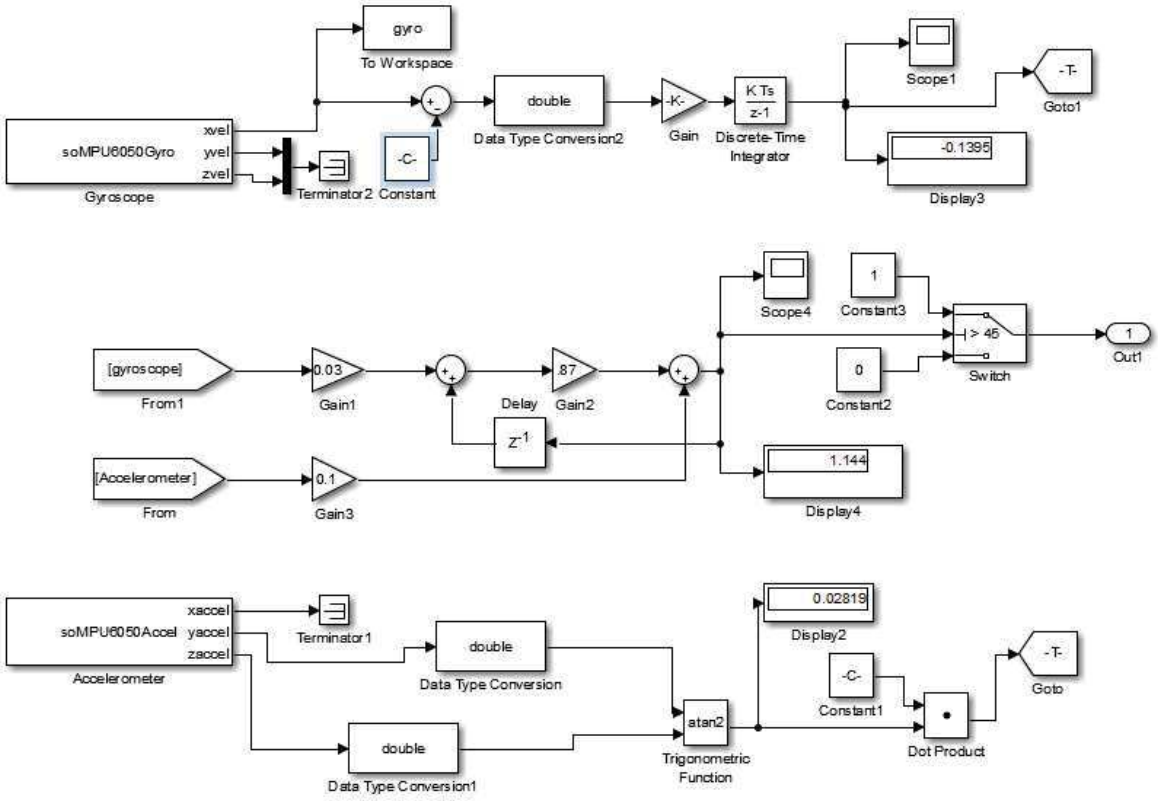


Figura 11 Filtro complementario

2.2 Conexión y Control de Velocidad Motor Brushless

Para el uso del motor brushless, se utilizó la plataforma Arduino. La conexión se lleva a cabo como muestra la figura 12. El pin naranja del regulador de velocidad del motor va en el pin 6 de Arduino. Y el pin café es conectado a la tierra de Arduino. Este esquema de conexión se aplica a la maqueta de tal manera que el motor quede suspendido en los extremos de la maqueta provocando una elevación de las partes móviles de la maqueta al someter a una aceleración al motor. Con estas conexiones se obtienen los ángulos y es posible accionar el motor.

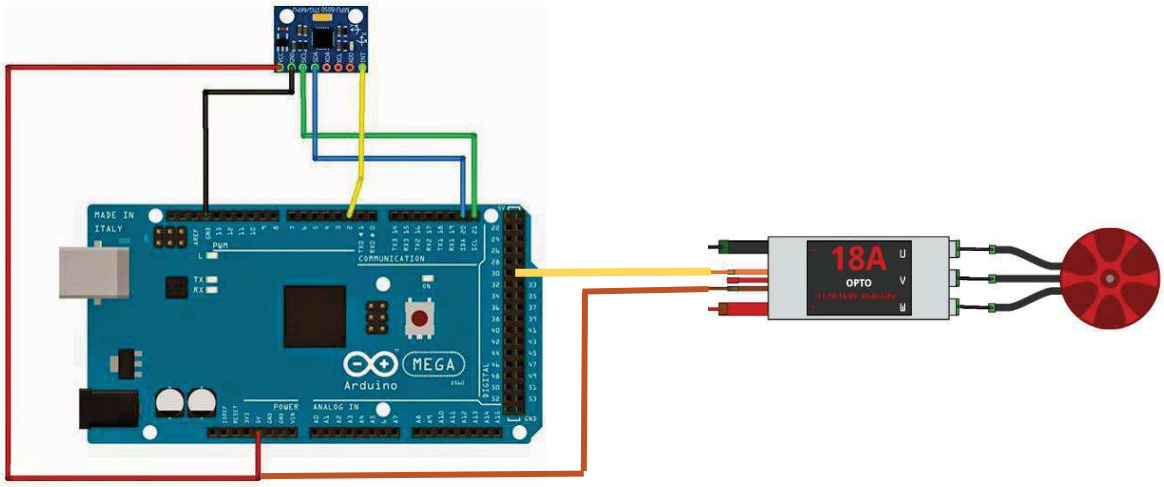


Figura 12 Conexión Motor

La velocidad del motor es regulada por señal PWM, originada mediante el bloque de programación Servomotor de la librería de Arduino en Simulink como muestra la figura 13. El valor de la constante es modificado por la slider, para aumentar y disminuir el ancho de pulso en la programación del bloque.

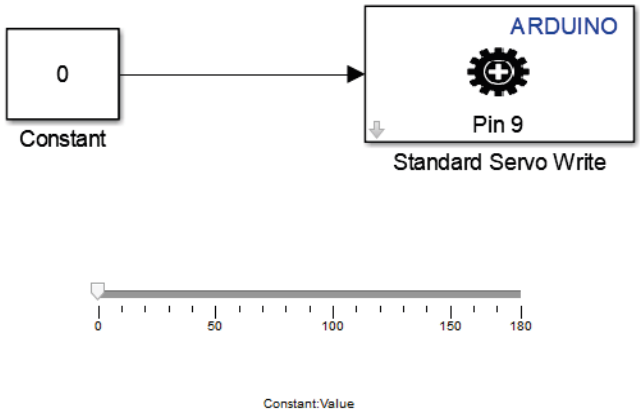


Figura 13 Programación de Velocidad

La frecuencia de operación del regulador de velocidad E300 es de 30Hz - 450Hz. El bloque de control de servomotor fue ajustado para la manipulación de un servomotor MG995 el cual trabaja a una frecuencia de 50Hz, como muestra la figura 14. Por lo tanto, este bloque permite trabajar con la frecuencia propuesta en el datasheet del regulador de velocidad E300 del motor.

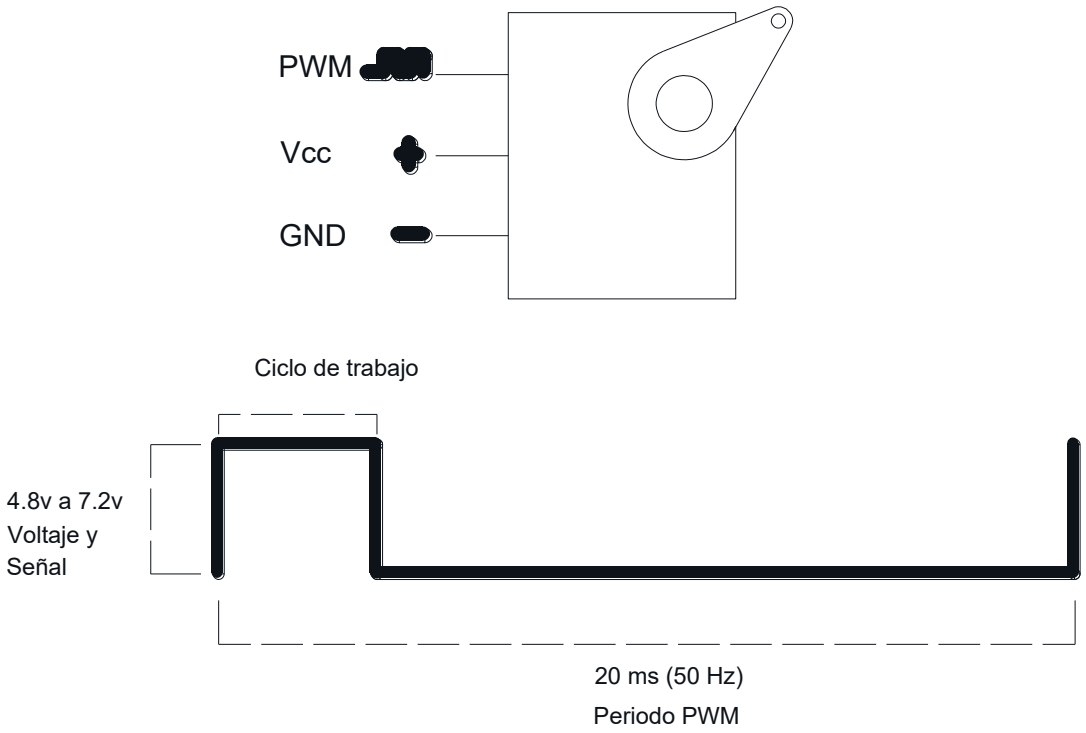


Figura 14 PWM de Servomotor MG995

La figura 14 explica el modo de operación del servomotor MG995, el cual trabaja a una frecuencia de 30 a 450Hz, por lo que se deduce que el ancho de pulso mínimo y máximo de operación para -90° y 90° es de 1 ms a 2 ms con un periodo de 20 ms contantes. Los pines de color naranja y café de la figura representan los mismos pines a conectar que los de la figura 12.

2.3 Simulación y obtención de los polos en Matlab

Para la función de transferencia de la ecuación 11 se asignan los valores de las constantes presentadas en el apartado 2.1, obteniendo la función de transferencia de la ecuación 12:

$$\frac{\omega(s)}{V(s)} = \frac{0.03306}{0.000049s^2 + 0.000391s + 0.002058} \tag{12}$$

Para el análisis de estabilidad se utiliza el método de Ruth-Hurwitz, donde por la posición de polos y ceros podemos determinar la estabilidad el sistema. Los polos y ceros del sistema se obtienen mediante la sintaxis `G= tf (num, den)` en Matlab, asignando los valores en el denominador y numerador en la ecuación. Posteriormente se grafica con `pzmap(G)` y obtenemos los polos y ceros mediante la sintaxis: `[polos_1, ceros_1]= pzmap(G).`

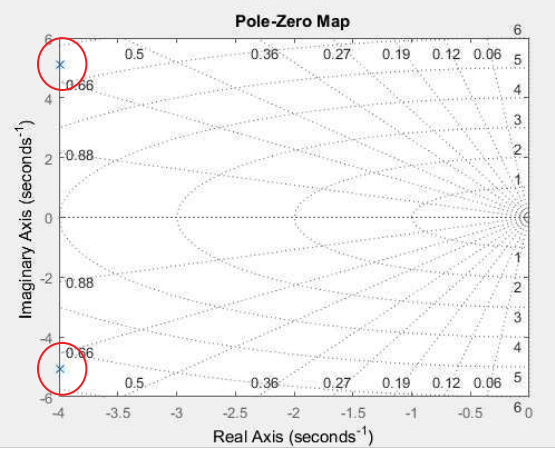


Figura 15 Polos de la FT

Según la figura 15, se observa que nuestro sistema es estable puesto que se encuentran dos polos conjugados en la parte negativa de la gráfica, y el criterio de Routh nos dice que, si nuestros polos se encuentran en el cuadrante negativo, nuestro sistema se denomina estable.

2.4 Estructura en 3D del sistema

En la figura 16 se presenta la estructura de la maqueta en el software inventor donde se muestran los motores utilizados, uno en cada extremo de una barra sujeta por su centro con la finalidad de que un motor compense a otro a la hora de que se produzca una alteración en la posición de reposo. Lo anterior logrado mediante el sistema de control creando un punto de equilibrio, el cuál debe ser mantenido por los motores de cada extremo. Estos deben girar a diferentes velocidades y en sentidos contrarios para obtener el equilibrio total de la barra que los sostiene.

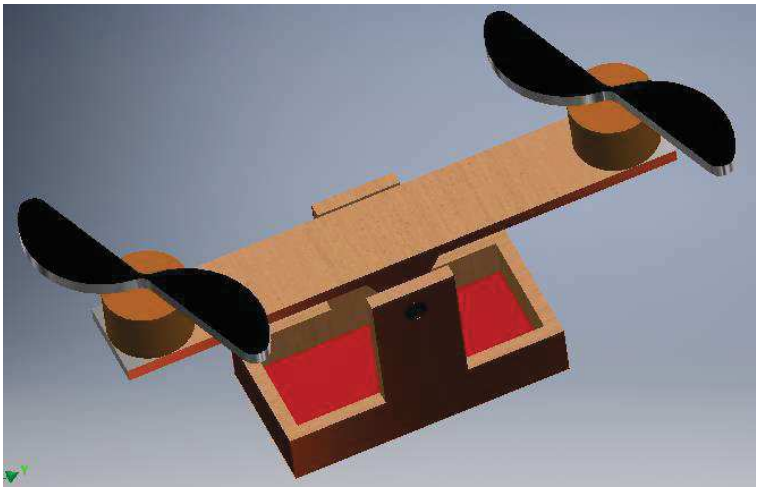


Figura 16 Emulación del sistema

Ingeniería y Automatización, Capítulo 39, pp. 481 - 496.
ISBN: 978-607-9394-15-8, Asociación Mexicana de Mecatrónica A.C., 2018.

3. Control PID

Para el cálculo de las ganancias se utilizó el método de Ziegler-Nichols. El cual permite conocer las ganancias Kp, Ti y Td del controlador PID. Se implementó el segundo método ya que el denominador contiene polos imaginarios conjugados. Además, el denominador cuenta con un exponente al cuadrado, lo cual indica que nuestro sistema es de segundo orden, por lo tanto, segundo método de Ziegler-Nichols.

El primer paso fue llevar a la inestabilidad la planta aumentando la ganancia proporcional el sistema hasta el punto en que el sistema oscile para obtener la ganancia crítica a partir de la gráfica.

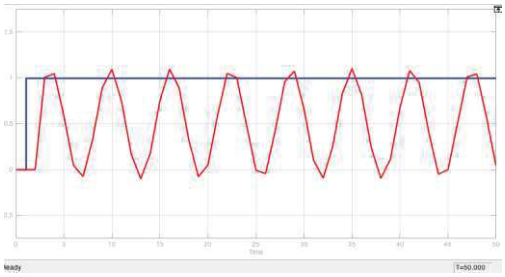


Figura 17 Gráfica de Kcr

Por medio de la gráfica de comportamiento que presenta la planta ante una ganancia que lo hace oscilar los resultados obtenidos Pcr es el tiempo que tarda en transcurrir un periodo en la onda. Obteniendo valores de Kcr=0.606 y de Pcr = 0.8

Ya que se conocen las ganancias Kcr y Pcr se procede a utilizar el segundo método de Ziegler-Nichols. Se utiliza el segundo método debido a que la planta presenta polos imaginarios conjugados. Tal hecho es justificado por los criterios de selección de método que indica Ziegler-Nichols.

Tabla 1 Ziegler-Nichols

Controlador	Kp	Ti	Td
P	0.5*Kcr	Infinito	0
PI	0.45*Kcr	Pcr/1.2	0
PID	0.6*Kcr	Pcr/2	0.125*Pcr

Al realizar las operaciones de la tabla 1, anteriormente mostrada, se obtienen valores de la tabla 2.

Tabla 2 Ganancias obtenidas por Ziegler-Nichols

Controlador	Kp	Ti	Td
P	0.30	0	0
PI	0.27	0.66	0
PID	0.364	0.417	0.1

Una vez obtenidos los valores de las ganancias, se desarrolla la simulación en simulink de cada uno de los controles aplicados a la planta.

En la figura 18 se muestra la respuesta del controlador PID ante una entrada escalón, dejando demostrado que nuestro sistema debería permanecer en la estabilidad pasado los 5 seg.

$$Kp=0.364 \quad Ki=0.417 \quad Kd=0.1$$

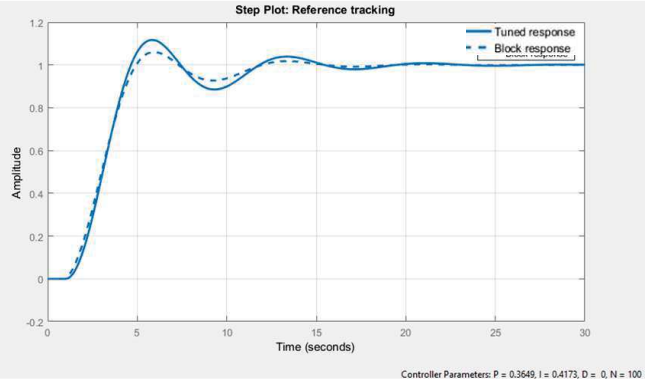


Figura 18 Gráfica de respuesta PID

4. Resultados

Las ganancias obtenidas por el modelado matemático de la planta fueron las mostradas en la tabla 3, y las ganancias aplicadas al sistema por calibración son las de la tabla 4.

Tabla 3 Ganancias obtenidas por Ziegler-Nichols

Controlador	Kp	Ki	Kd
PID	0.364	0.417	0.1

Tabla 4 Ganancias modificadas aplicadas al sistema

Controlador	Kp	Ki	Kd
PID	0.33	0.00265	0.3

Al incluir los valores en el controlador PID del programa en físico se notaron algunas irregularidades en la reacción ante las perturbaciones. Las oscilaciones eran constantes y tardaba aproximadamente 5 segundos en lograr la estabilidad. Por lo que se redujo la ganancia integral y se aumentó la ganancia derivativa para atenuar la señal y reducir los picos graficados que se muestran en la figura 19. Obteniendo la gráfica de la figura 20.

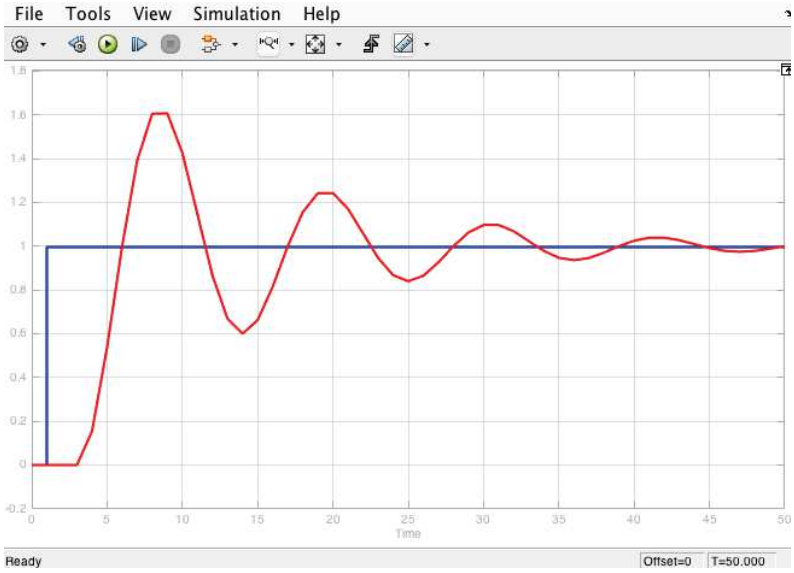


Figura 19 Gráfica de respuesta de la planta física

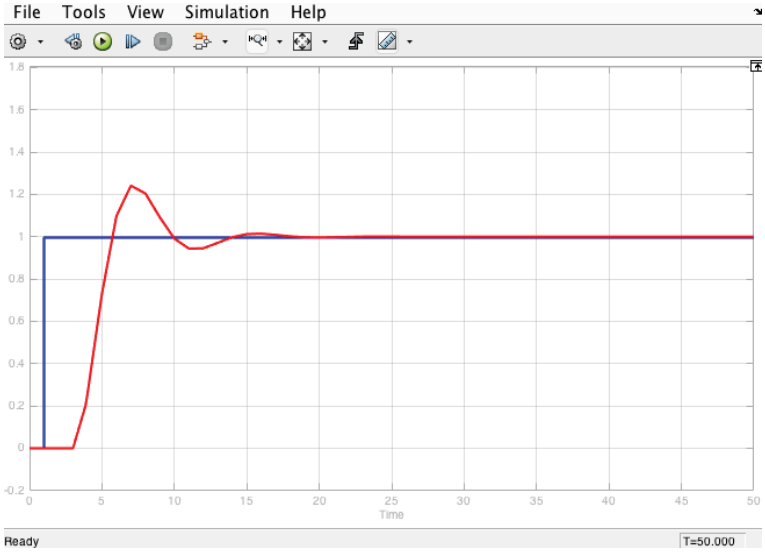


Figura 180 Grafica de comportamiento del sistema con PID calibrado

La figura muestra como el comportamiento del sistema predice la velocidad de subida y reduce la aceleración para llegar al setpoint.

Figura 19 Maqueta de pruebas

En la figura 21 se presenta la maqueta utilizada para la realización de las pruebas de velocidad de los motores brushless. Dicha estructura fue la presentada en la figura 16 de manera virtual.

5. Conclusiones

El comportamiento del motor con los valores dados es estable ante ciertas perturbaciones. El modelado matemático del motor permite conocer su comportamiento ante las entradas al sistema, teniendo como resultado de salida la velocidad en el rotor. Esto se logra mediante el modelado matemático por análisis de control clásico o función de transferencia. En la función de transferencia obtenida se sustituyen los valores del motor para obtener su comportamiento real. Con todo esto, se puede conocer cómo sería su conducta en el momento que este se someta a perturbaciones en el aire.

Ya teniendo el comportamiento de motor, se le ha aplicado diferentes perturbaciones, tales como vientos de diferentes velocidades (hasta 30 Km/h), teniendo buenos resultados con el control PID mostrado, está por aplicarse en móviles aerostáticos (como primera instancia con helio) a control remoto para sistema de vigilancia en la periferia de la Universidad Tecnológica Nayarit, contando con cámara de visión nocturna. Del sensor giroscopio se obtiene la posición del móvil para su corrección, cabe mencionar que las pruebas en campo se van a efectuar próximamente, ya que las pruebas en laboratorio ya están corregidas. Se han verificado precios del gas Helio y el gasto en energía, teniendo un gasto inferior en energía con el móvil que un drone convencional, este último solo tiene una duración de 10 a 20 minutos de energía (dependiendo de la batería, costo elevado), en el móvil aerostático solo se activan los motores para corrección, avance y giros.

Referencias

[1] Benjamin C, K. (1996). Sistemas de Control Automático. México: Prentice Hall.
[2] Bolton, W. (2001). Ingeniería de Control. México: ALFAOMEGA.
[3] Bueno, D. (2002). Motor brushless: Funcionamiento y Características. Sistemas automáticos.
[4] Fernando Reyes Cortés, J. C. (2013). Mecatrónica Control y Automatización. México: Alfaomega.
[5] Ogata, K. (2010). Ingeniería de Control Moderna. Madrid: Pearson.

Ingeniería y Automatización

Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
Se tiraron 1000 ejemplares, en formato digital pdf,
tipo internet, pp. 503, Tamaño: 30 Mb.

ISBN: 978-607-9394-15-8

ISBN:978-607-9394-15-8