



Universidad Autónoma de Querétaro
 Facultad de Ingeniería, División de Investigación y Posgrado
 Maestría en Ciencias de Ingeniería, Instrumentación y Control Automático



Diseño e implementación de un controlador de movimiento para un servo sistema.

Tesis

Que como parte de los requisitos para obtener el grado de
 Maestro en Ciencias en Instrumentación y Control Automático

Presenta:

Ing. Carlos Miguel Torres Hernández

Dirigido por:

Dr. Juvenal Rodríguez Reséndiz

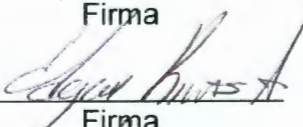
Dr. Juvenal Rodríguez Reséndiz
 Presidente


 Firma

MC. Miguel Ángel Martínez Prado
 Secretario


 Firma

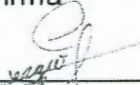
Dr. Edgar Alejandro Rivas Araiza
 Vocal


 Firma

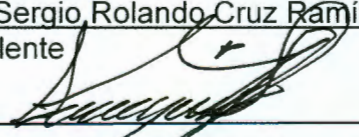
Dr. Roberto Augusto Gómez Loenzo
 Suplente


 Firma

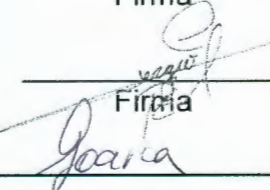
Dr. Sergio Rolando Cruz Ramírez
 Suplente


 Firma

Dr. Aurelio Domínguez González
 Director de Facultad de Ingeniería



Dra. Ma. Guadalupe Flavia Loarca Piña
 Directora de Investigación y Posgrado


 Firma

Centro Universitario
 Querétaro, Qro.
 Noviembre 2016

México

RESUMEN

Los sistemas tecnológicos para el sector industrial han tenido un crecimiento importante en los últimos años, destacando el área de control de movimiento, que mediante la integración de diferentes conocimientos de software y hardware son capaces de incrementar la productividad y eficiencia, así como la reducción de costos. El objetivo del presente trabajo es el diseño de un sistema de control de movimiento utilizando dos de las tecnologías más importantes en el área, los FPGA y los DSP, para el control de motores de DC, se trabajó una metodología en la cual se detalla el desarrollo de hardware y software. Uno de los principales objetivos es el desarrollo de una plataforma electrónica competente para este proyecto. De igual manera se describe una metodología para el diseño, simulación y validación de los algoritmos de control utilizados en una mesa de coordenadas utilizando el software Matlab y Simulink.

Palabras clave: FPGA, DSP, Controlador de movimiento, Motor de DC, PCB

SUMMARY

Technological systems for the industrial sector have had a significant growth in recent years, especially the movement control area, which through the integration of different software and hardware knowledge are able to increase productivity and efficiency, as well as the reduction of prices. The objective of the present work is the design of a movement control system using two of the most important technologies in the area, the FPGA and the DSP, for the control of DC motors, a methodology was worked out which details the development Hardware and software. One of the main objectives is the development of a competent electronic platform for this project. A methodology for the design, simulation and validation of control algorithms used in a coordinate table using Matlab and Simulink software is also described.

(Key words: FPGA, DSP, Motion Controller, DC Motor, PCB)

A mi familia, porque gracias a su cariño, guía y apoyo he llegado a realizar un objetivo más en mis estudios profesionales, que constituyen el legado más valioso que pudiera recibir.

AGRADECIMIENTOS

A Dios, por acompañarme todos los días en mi camino.

A mi mamá, quien siempre ve por mí y lo da todo por nosotros, porque sin ella no estaría donde estoy en estos momentos. Te quiero mamá.

A mi papá, por ser el apoyo y ejemplo en mi vida. Gracias por tu ejemplo trabajador que siempre me motiva a seguir adelante, este es mi futuro, el que tú me ayudaste a construir.

A mi hermana, por ser mi guía y ejemplo de profesionalismo, una gran amiga y mi enfermera. Gracias hermanita.

A mi novia Aída, a quien agradezco todo su apoyo, confianza y amor por todo este tiempo, porque este trabajo también tiene tu esfuerzo y tu comprensión. Gracias por siempre motivarme a seguir y no dejarme caer.

A mi familia, Yolanda Hernández y Enrique Franco quienes siempre ven por mí como su hijo y a quienes quiero mucho. A mi hermano, Oscar por ser un gran amigo y ejemplo de perseverancia y pasión por las cosas que uno ama.

A mi asesor Juvenal, por su confianza, apoyo y gran amistad que construimos en este tiempo, gracias por sus consejos y palabras, gracias por siempre tener un tiempo para mí, gracias por alentarme a seguir creciendo.

A mis sinodales y maestros porque siempre me apoyaron en los momentos que lo necesitaba, por sus palabras y correcciones para terminar este trabajo.

A mis maestros que me marcaron en cada etapa de mi formación. Porque ellos fueron la base de lo que hoy soy como profesionalista.

Gracias a todos mis compañeros con quien compartí este trabajo porque siempre es mejor las cosas cuando las puedes compartir con gente como ustedes.

A la universidad por darme la oportunidad de seguir desarrollándome y de poder aportar algo a la sociedad. “Educo en la verdad y el honor”.

A CONACyT por confiar en proyectos que son el presente de nuestro país.

A todas las personas que durante este tiempo pudimos coincidir y que de una u otra forma fueron parte importante de este trabajo.

TABLA DE CONTENIDOS

Resumen

Summary

Dedicatorias

Agradecimientos

Índice

Índice de Tablas

Índice de Figuras

Abreviaciones

1. INTRODUCCIÓN	12
1.1 JUSTIFICACIÓN	13
1.2 PLANTEAMIENTO DEL PROBLEMA.....	14
1.3 DESCRIPCIÓN DEL PROYECTO.....	16
1.4 OBJETIVOS.....	17
1.4.1 <i>Objetivo General</i>	17
1.4.2 <i>Objetivos Particulares</i>	17
2. REVISIÓN DE LA LITERATURA.....	18
2.1 ANTECEDENTES.....	18
2.1.1 <i>Antecedentes UAQ</i>	18
2.1.2 <i>Antecedentes Generales</i>	19
3. METODOLOGÍA	22
3.1 MARCO TEÓRICO	22
3.1.1 <i>Servomecanismo</i>	22
3.1.2 <i>Servodrivers</i>	22
3.1.3 <i>Control de movimiento</i>	22
3.1.4 <i>Cálculo de Trayectoria</i>	23
3.1.5 <i>Componentes de un sistema de control de Movimiento</i>	25
3.1.6 <i>Decodificador Incremental (Encoder)</i>	26

3.2	METODOLOGÍA.....	28
3.2.2	<i>Modelado e identificación del sistema.....</i>	<i>29</i>
3.2.3	<i>Modelo del motor con la corriente como salida.....</i>	<i>33</i>
3.2.4	<i>Parámetros del motor.....</i>	<i>34</i>
3.2.5	<i>Diseño de controlador.....</i>	<i>37</i>
3.2.6	<i>Generador Perfil Trapezoidal.....</i>	<i>41</i>
3.2.7	<i>Diseño y fabricación de PCB.....</i>	<i>43</i>
3.2.8	<i>Arquitectura de la plataforma.....</i>	<i>48</i>
3.2.9	<i>Descripción de un sistema digital basado en DSP y FPGA.....</i>	<i>51</i>
4.	RESULTADOS.....	55
5.	CONCLUSIONES.....	68
6.	REFERENCIAS.....	70
7.	APÉNDICE.....	74
7.1	DIAGRAMAS ESQUEMÁTICOS.....	75
7.2	SCRIPTS MATLAB.....	91

ÍNDICE DE TABLAS

Tabla 1. Parámetros del motor de DC

Tabla 2. Metodología propuesta para el desarrollo del proyecto.

Tabla 3. Conexión de encoder a plataforma.

ÍNDICE DE FIGURAS

GRAFICA DE PERFIL DE VELOCIDAD TRAPEZOIDAL	24
COMPONENTES DE UN SISTEMA DE CONTROL DE MOVIMIENTO.....	26
ENCODER DE MOTOR DE DC, RESOLUCIÓN DE 500 CPR A 5V Y 200MA.....	27
SEÑALES DE CUADRATURA DE ENCODER, CANALES A Y B	27
MODELO DEL MOTOR DE IMANES PERMANENTES DE DC.	30
DIAGRAMA DE BLOQUES DE UN MOTOR DC.	33
DIAGRAMA A BLOQUES DE MODELO DE MOTOR DC EN MODO CORRIENTE.	34
MODELO DE MOTOR DC CON ENTRADA DE VOLTAJE Y CORRIENTE COMO SALIDA.	34
MODELO DE MOTOR DC EN SOFTWARE MATLAB/SIMULINK.	35
RESPUESTA DEL MOTOR DE DC A UNA ENTRADA ESCALÓN	36
RESPUESTA DE LA CORRIENTE AL APLICAR EL VOLTAJE NOMINAL AL MOTOR.....	37
MODELO DE MOTOR CON CONTROLADOR PD, SIMULACIÓN EN SOFTWARE SIMULINK.....	40
CONTROLADOR PD IMPLEMENTADO EN SIMULINK.....	40
RESPUESTA DEL MOTOR DE DC A UNA ENTRADA ESCALÓN CON UN CONTROLADOR PD.	41
FLUJO DE DISEÑO DE PCB DE PLATAFORMA ELECTRÓNICA	45
DISEÑO CAD DE PLATAFORMA CREADA EN EAGLE	47
FABRICACIÓN DE DISEÑO DE CIRCUITO IMPRESO EN BAQUELITA.....	47
TARJETA CON COMPONENTES MONTADOS Y SOLDADOS VISTA SUPERIOR (IZQUIERDA) VISTA INFERIOR (DERECHA)...	47
ARQUITECTURA PROPUESTA DE LA PLATAFORMA ELECTRÓNICA.....	48
DIAGRAMA DE FLUJO PROGRAMA DEL DSP	52
DIAGRAMA DE FLUJO DE COMUNICACIÓN DE PC.....	53
DIAGRAMA DE FLUJO PROGRAMA FPGA	54
PLATAFORMA ELECTRÓNICA FINAL CON COMPONENTES SOLDADOS VISTA INFERIOR.....	55
PLATAFORMA ELECTRÓNICA FINAL CON COMPONENTES SOLDADOS VISTA SUPERIOR.	56
MICROSCOPIO DIGITAL RESOLUCIÓN 640 X 480, 5X ZOOM DIGITAL.	56

REVISION DE MONTAJE DE COMPONENTES CON MICROSCOPIO DIGITAL A) CRISTAL Y CAPACITOR; B) PINES DE DSP; C) PINES DE FTDI; D) RESISTENCIA Y LED E) PINES FPGA.....	57
MESA DE COORDENADAS COMO BANCO DE PRUEBAS CON IDENTIFICACIÓN DE EJE "X" Y "Y".	57
MOTOR DC PARA EJE X DE MESA DE COORDENADAS.....	58
MOTOR DC PARA EJE Y DE MESA DE COORDENADAS.....	58
CONEXIÓN DE ENCODER CON DB15.	59
SERVODRIVERS MODELO MAXON MOTOR ADS PARA EJE "X" Y "Y"	60
CONEXIÓN DE SISTEMA CON LA PLATAFORMA ELECTRÓNICA.	60
SIMULACIÓN DE SISTEMA CON BLOQUE DE CONTROL (P, PI, PD Y PID).	61
RESPUESTA DEL SISTEMA CON CONTROLADOR PD TEÓRICO (IZQUIERDA) CON PLATAFORMA (DERECHA).....	61
RESPUESTA DE CORRIENTE DEL SISTEMA CON CONTROLADOR PD.	62
RESPUESTA TEÓRICA DE UN CONTROLADOR DE TIPO PI ANTE UN ENTRADA DE TIPO ESCALÓN DE 500 RPM.	63
VALORES TEÓRICOS DE LAS GANANCIAS PARA CONTROLADOR PI.....	63
RESPUESTA DEL SISTEMA CON CONTROLADOR PI ANTE UNA ENTRADA ESCALÓN DE 500RPM.....	64
RESPUESTA TEÓRICA DE UN CONTROLADOR DE TIPO PID ANTE UN ENTRADA DE TIPO ESCALÓN DE 500 RPM.....	64
VALORES TEÓRICOS DE LAS GANANCIAS PARA CONTROLADOR PID.	65
RESPUESTA DEL SISTEMA CON CONTROLADOR PID ANTE UNA ENTRADA ESCALÓN DE 500RPM	65
RESPUESTA DEL PERFIL TRAPEZOIDAL CON UNA VELOCIDAD DE 13000 PULSOS/S, Y UNA POSICIÓN FINAL DE 40000 PPR.....	66
RESPUESTA DEL PERFIL TRAPEZOIDAL CON UNA VELOCIDAD DE 13000 PULSOS/S, Y UNA POSICIÓN FINAL DE 80000 PPR.....	67
RESPUESTA DEL PERFIL TRAPEZOIDAL CON UNA VELOCIDAD DE 13000 PULSOS/S, Y UNA POSICIÓN FINAL DE 40000 PPR.....	68

ABREVIACIONES

ABET - Accreditation Board for Engineering and Technology

ADC – Analog Digital Convert

ASICA - Application Specific Integrated Circuit

CNC- Computer Numerical Control

DSP - Digital Signal Processor

FPGA – Field Programmable Gate Array

GPIO - General Propose Inputs Outputs

LCD – Liquid Crystal Display

LED - Light-Emitting Diode

PCB – Printed Circuit Board

PWM – Pulse Width Modulation

1. INTRODUCCIÓN

En la actualidad los servomecanismos se usan de manera extensa en la industria, en máquinas herramientas de control numérico y principalmente en robots, siendo este último un elemento indispensable en una gran parte de los procesos de manufactura, impulsados principalmente por el sector automotriz; mientras nuevas tecnologías y aplicaciones robóticas industriales surgen, mientras que los robots industriales que llevan mucho tiempo realizando labores enfrentan problemas en los procesos automatizados, por lo que cada vez grandes industrias se deshacen de máquinas en desuso, en tanto que pequeñas y medianas empresas no tienen acceso a esta tecnología, principalmente debido a altos costos tales como accesorios, tarjetas de control, interfaz humano-máquina y que cada fabricante utiliza protocolos propios para realizar la comunicación entre los dispositivos (Bracarense 1999).

Los sistemas de control de movimiento juegan un papel importante en el control de los diferentes tipos de servosistemas. La mayoría de sistemas de control de movimiento se basan en un Procesador de Señales Digitales, soluciones en un chip como la tecnología FPGA, también con chips de uso dedicado o Application Specific Integrated Circuit (ASIC), o con base en un microprocesador o un microcontrolador. Un sistema de control de movimiento necesita muchas funciones con el fin de realizar de forma rápida y precisa las complejas tareas necesarias en el control de robots industriales y sistemas de automatización (Chan *et al.*, 2007).

Bajo las circunstancias mencionadas, es necesario desarrollar controladores electrónicos con un alto poder de cómputo y, también contar con sistemas de comunicación internos y externos de muy alta velocidad (Bueno *et al.*, 2009). Un FPGA proporciona una solución entre un ASIC y un procesador de propósito general (Gu *et al.*, 2010). Por otra parte los DSP ofrecen una solución para el procesamiento matemático que se requiere en este tipo de controladores.

1.1 Justificación

Los controladores de movimiento disponibles en el mercado tienen un alto costo, su arquitectura es cerrada; por tanto, no es posible integrar módulos personalizados; su *firmware* está protegido, es decir, no es posible modificar sus algoritmos a fin de ajustar su comportamiento a las necesidades del proceso. También, sus frecuencias de operación son relativamente bajas y pueden no satisfacer las tasas de procesamiento y transferencia de datos requeridas.

Más adelante se mencionan alternativas a nivel de prototipo reportadas en la literatura las cuales están basadas en PC y sistemas embebidos. Este tipo de sistemas son altamente flexibles, modulares y escalables; también reducen ampliamente el tiempo de desarrollo. El problema asociado a estos sistemas es que no cumplen con los requerimientos de tiempo real.

Los sistemas basados en lógica reconfigurable, aunque presentan un tiempo de desarrollo mucho más largo, satisfacen requerimientos de velocidad de procesamiento y transferencia de datos. Lo anterior debido a que realizan operaciones de forma concurrente. Existen trabajos de control de movimiento implementados en FPGA o en DSP que reportan excelentes resultados; sin embargo, estos proyectos no trascienden más allá de pruebas con servomecanismos a nivel laboratorio y su implementación se lleva a cabo en tarjetas de desarrollo comerciales las cuales, por ser de propósito general, no aprovechan al máximo las prestaciones de estos dispositivos.

Es necesario desarrollar una tarjeta de control de movimiento especializada para el análisis de sistemas mecánicos que cumpla con los requerimientos de velocidad de procesamiento y desempeño a fin de reducir los costos y tiempos de implementación. Con la cual se pueda poner en operación procesos industriales dando como resultado, el aumento de la calidad de los productos a costos competitivos.

La integración de las dos principales tecnologías digitales (lógica reconfigurable y sistemas embebidos) para el control de movimiento puede conducir a la fabricación de sistemas de control de movimiento de alto desempeño lo suficientemente flexibles para cubrir las necesidades de los bancos de prueba para productos de consumo en general.

1.2 Planteamiento del Problema

Una de las características del mundo contemporáneo es el acelerado cambio tecnológico, y de forma particular en el ámbito de las tecnologías de la información y las comunicaciones. Esto se traduce en mejores y mayores oportunidades en los diversos campos del conocimiento.

El impacto de la tecnología en las empresas ha sido cada vez más importantes en los últimos años en que cada vez se han ido incorporando nuevas tecnologías y se ha producido un gran avance dentro de la propia tecnología. Cada vez las tecnologías facilitan las tareas de las empresas y producen innovación, lo que hace que las empresas que no se sumen a estas tecnologías se vayan quedando atrás.

Una empresa que incorpora nuevas tecnologías es más competitiva, cuenta con mayores recursos de producción y competitividad, mientras que las que no invierten en tecnología se van quedando atrás, bien porque no tienen las mejoras que las demás empresas o su capacidad productiva es mejor, esto depende mucho de la empresa de la que se trata, pero está claro que una empresa que va incorporando nuevas tecnologías se encuentra en avance constante.

Existen dos problemas relacionados con la importancia de la tecnología en la empresa: El primero es que la tecnología se está renovando constantemente de manera que la inversión cada vez es mayor y existen muchas empresas que no

son capaces de ir avanzando a su ritmo por su capacidad económica, y el segundo se refiere a la dependencia total que existe de la tecnología en muchas empresas de manera que si se produce un fallo la empresa se encuentra totalmente desamparada y en muchas ocasiones sin poder continuar hasta que esto sea solventado.

Es muy importante desarrollar alternativas para las pequeñas y medianas empresas debido a que en la actualidad los fabricantes de dispositivos de control de alto rendimiento presentan un bajo nivel de personalización y costos muy elevados provocados por los gastos de importación y servicios de soporte técnico. Así mismo muchos de los servomecanismos y controladores que se utilizan en empresas nacionales son productos de origen extranjero, que necesitan de usuarios especializados que cuenten con la capacitación adecuada para el manejo del equipo.

En México se está compitiendo con países de gran desarrollo dentro de los mercados internacionales, por lo que es necesario tomar acciones para fortalecer al país con soluciones eficientes, confiables y seguras, pero que sean accesibles con respecto a los productos ya existentes en el mercado. De esta manera estaríamos impactando en un amplio porcentaje de las pequeñas y medianas empresas en México.

El sector de la robótica, al contrario que la máquina herramienta y otros segmentos de equipamiento industrial, no ha adoptado ningún estándar significativo en el área de control, por lo que cada fabricante ofrece su propia tecnología; lo que dificulta la integración de los robots en los entornos de producción. Un robot industrial está diseñado para soportar una larga vida de trabajo; sin embargo, los controladores quedan obsoletos más rápidamente que los robots que controlan. Cuando un fabricante lanza al mercado una nueva generación de controladores, las mejoras e innovaciones del último modelo no suelen estar disponibles para las versiones previas. Incluso, es frecuente que las mejoras introducidas en el lenguajes de programación no puedan ser utilizadas en

sistema ya en funcionamiento; esto debido a la arquitectura cerrada de los controladores que no permiten una reconfiguración o modularidad de sus componentes. Por otro lado, la disponibilidad de recambios en tarjetas y componentes electrónicos por parte del fabricante suele limitarse a un periodo de tiempo mucho menor que la vida útil del equipo, debido a la rápida obsolescencia tecnológica de los circuitos integrados. Estos factores dificultan enormemente incorporar los avances tecnológicos en equipos ya instalados y que han supuesto habitualmente, una fuerte inversión económica (Gonzalez et. al., 2004).

Con base a lo anterior se debe cubrir las demandas de las empresas a través de una propuesta de un controlador de movimiento que brinde la posibilidad de incrementar la calidad de los procesos a su control, además de poder ofrecer al sector industrial un controlador confiable, de buena relación costo- beneficio, flexible y modular para evitar la obsolescencia tecnológica.

1.3 Descripción del proyecto

El resultado de esta investigación es congruente con las necesidades del sector industrial y está encaminado a la solución de los problemas de control de movimiento para robots, maquinaria CNC y de cualquier servo mecanismo en general.

El controlador de movimiento, resultado de este proyecto, será utilizado para la implementación en una mesa coordinada; sin embargo, este sistema puede ser utilizado como plataforma para otros trabajos de investigación relacionados con las áreas de control, sistemas digitales y servo mecanismos.

La plataforma a desarrollar impulsará una herramienta con la que se pueda trabajar de manera eficiente en los cursos afines con los cuales se pretende formar y dotar al alumno con las capacidades mencionadas en ABET (organización no gubernamental, sin ánimo de lucro, dedicada a la acreditación de programas de educación universitaria o terciaria en disciplinas de ciencias aplicadas, ciencias de la computación, ingeniería y tecnología):

- Capacidad de aplicar los principios de matemáticas, ciencia e ingeniería.
- Capacidad para identificar, formular y resolver problemas de ingeniería.
- El reconocimiento de la necesidad y la capacidad de participar en el aprendizaje de por vida.
- Capacidad para utilizar las técnicas, habilidades y herramientas modernas de ingeniería necesarias para la práctica de la ingeniería.

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar e implementar una tarjeta de control de movimiento con base en una arquitectura modular que cumpla con los requerimientos de velocidad de procesamiento y desempeño para un servo sistema.

1.4.2 Objetivos Particulares

- Diseñar y fabricar una tarjeta electrónica que incluya un dispositivo de lógica reconfigurable y un procesador digital de señales junto con los componentes electrónicos para su funcionamiento y programación.
- Controlar independientemente cada actuador (motor) del servo sistema mediante servodrives, para entender su modo de funcionamiento que después se implementará en la tarjeta de control.
- Describir e implementar un sistema digital basado en FPGA para el control de movimiento de motores síncronos de imán permanente y una interfaz de comunicación implementada en un controlador digital de señales.
- Desarrollar un banco de pruebas para el depurado de la tarjeta electrónica.

2. REVISIÓN DE LA LITERATURA

2.1 Antecedentes

2.1.1 Antecedentes UAQ

En lo referente a los trabajos realizados por los alumnos e investigadores de la facultad de Ingeniería de la UAQ, se puede mencionar que existen trabajos referentes a los sistemas de control de movimiento. Entre las investigaciones encontradas destacan el presentado por Osornio (2007) quien desarrolló un controlador para maquinaria de alta velocidad basado en FPGA, el presentado por Roquillo (2002) quien desarrolló un controlador digital para servomotores sin escobillas, implementando un algoritmo de control usando un DSP en el cual maneja el concepto de generación de trayectorias en hardware mediante perfiles trapezoidales en velocidad; la investigación de Osornio (2004) quien diseñó un controlador de 3 ejes para máquinas herramienta basado en FPGA. El trabajo de Fortino (2012) quien realizó un controlador estándar de movimiento multieje con base en FPGA.

Un trabajo de particular interés es el descrito por Osornio-Ríos en (2008), en el cual se describe la implementación de un controlador de movimiento para una máquina CNC de alta velocidad. Los resultados confirman la ventaja de utilizar los dispositivos de lógica reconfigurable sobre los DSPs. Sin embargo, la dinámica de una máquina con estas características es menor comparada con un robot industrial.

Galindo (2001) presenta un desarrollo de una tarjeta electrónica basada en los controladores de arquitectura abierta, el cual fue implementado en un procesador digital de señales, con dicha tarjeta realizó experimentaciones sobre servomecanismos con motores de corriente continua. Por otra parte, Loenzo (2009) propone el desarrollo de un sistema CNC de maquinado con base en un proceso estandarizado de desarrollo de software, enfocándose en definir de una

manera precisa los requerimientos del sistema CNC, de tal forma que se tenga un sistema de arquitectura abierta que se adapte al proceso de desarrollo del fabricante.

2.1.2 Antecedentes Generales

Hoy en día la robótica industrial, trata fundamentalmente de dotar flexibilidad a los procesos productivos manteniendo al mismo tiempo la productividad que se consigue con una máquina automática especializada (robot), por esta razón, actualmente se observa una tendencia a la integración de la robótica industrial en las técnicas de automatización y manufactura de las industrias (Santos, 2012).

Cuando se trata de sistemas de control de movimiento en el contexto de la industria local, frecuentemente se habla de empresas de origen extranjero como Galil Motion Controls (2011), Delta Tau (2011), Baldor, por mencionar sólo algunas. Estas y otras empresas tienen en su catálogo de productos soluciones de control movimiento completas; sin embargo, su principal limitante es que se trata de productos altamente genéricos y de tipo caja negra, es decir, no es posible realizar modificación alguna de su operación; sin mencionar su alto costo.

En este sentido, existen numerosos trabajos en la literatura científica que solucionan parcialmente esta problemática. La implementación de estas soluciones se puede catalogar por tres diferentes tecnologías utilizadas: micro-controladores, procesadores digitales de señales y dispositivos de lógica reconfigurable.

Las soluciones basadas en micro-controladores están fuertemente limitadas en cuanto a su frecuencia de operación. Sobh (2011) desarrolló un controlador para un robot manipulador de tres grados de libertad con un micro controlador MC68HC11. En él, implementa un controlador proporcional integral derivativo (PID); sin embargo, el periodo de muestreo obtenido es de aproximadamente 150Hz. Los resultados de este trabajo reportan un desempeño aceptable para la

dinámica del robot utilizado; sin embargo, un robot con una dinámica más compleja requiere de frecuencias mucho más altas.

Las soluciones basadas en DSPs son a menudo más eficaces. Masten (2000) describe las ventajas de utilizar un DSP sobre un micro-controlador cuando se trata de la implementación de algoritmos de control.

Lim (1996) describe la implementación de un controlador digital de movimiento usando una plataforma con base en un DSP. Este tipo de tecnología permite la implementación rápida de algoritmos de control dado que su implementación se realiza por software. Lim reporta una frecuencia de control de al menos 500us utilizando un dispositivo de Texas Instruments con número de parte TMS320C30.

Circuitos integrados de aplicación específica para el control de movimiento como el descrito por Grout and Dorey (2005) poseen una frecuencia de operación alta y generalmente no requieren de software adicional; sin embargo, estas soluciones requieren componentes adicionales de hardware como una interfaz de comunicación.

A diferencia de las plataformas antes mencionadas, un FPGA puede llevar a cabo procesamiento de señales de forma concurrente. Esto significa que un controlador implementado en esta plataforma puede operar en conjunto con otros componentes sin modificar su frecuencia de trabajo. Goetz (2004) reporta la implementación de un controlador de par en un servo-mecanismo y logra una frecuencia de control de hasta 5kHz.

Una solución mucho más sofisticada se describe en el trabajo de Astarloa (2009) implementa un decodificador de cuadratura, un controlador PID y un modulador de ancho de pulso (PWM por sus siglas en inglés) para actuar servo motores. Se incluyen cuatro instancias de cada uno de estos componentes junto con un procesador empotrado dentro del FPGA.

Oliveira y Guenther (2008) diseñaron un controlador con una estructura modular, cuya arquitectura de comunicación es en capas, la cual incluye una capa de tareas, una de integración, una de comunicación, una interfaz y una física, dichas capas e incluyendo el software fueron desarrollados bajo una arquitectura abierta.

Dentro del trabajo realizado por Xing, Jia y Yanqianga (2011) se propone el uso de una interfaz SERCOS (Serial Real-time Communication System), sobre la tecnología de bus EtherCAT, en el cual estructura el protocolo de comunicación en un modo maestro-esclavo, en el cual el dispositivo maestro utiliza un estándar utilizando un FPGA para enviar y recibir tramas de información, el modelo integra máquinas de estados para realizar la sincronización, la comunicación de datos de proceso y acceso a través de servicio, este modelo de estructura Ethernet muestra comportamiento en tiempo real. Hongyu, Yong y Na (2010), presenta un sistema distribuido de control digital de movimiento basado en SERCOS, por el cual transmite a través de fibra óptica, proporcionando al sistema inmunidad al ruido y alta velocidad, este sistema se compone de una PC y la tarjeta de interfaz SERCOS. El equipo esclavo comprende un DSP para controlar un servomotor.

La estructura modular de software que se adopta para el desarrollo de la arquitectura de controlador de movimiento libre es propuesta por Liu, Wang and Fu (2008), donde realiza un análisis de la estructura de sistema universal de Centro de Control Numérico.

Zbigniew *et al.* (2000) proponen la integración de DSP con un FPGA como solución a controladores eléctricos de alto desempeño.

3. METODOLOGÍA

3.1 Marco Teórico

3.1.1 Servomecanismo

De acuerdo con la Real Academia Española un servomecanismo se puede definir como:

“Sistema electromecánico que se regula por sí mismo al detectar el error o la diferencia entre su propia actuación real y la deseada”.

3.1.2 Servodrivers

Un *servodrive* es una herramienta de alta tecnología que integra en un solo dispositivo interfaces de potencia y control para servomecanismos.

Las principales características que aporta un sistema con *servodrives* son:

- Tiempo de posicionamiento
- Mayor precisión
- Mejor confiabilidad
- Mayor repetitividad
- Movimientos coordinados

3.1.3 Control de movimiento

Un controlador de movimiento actúa como el cerebro del sistema de control de movimientos y calcula cada movimiento comandado de la trayectoria. Debido a que esta tarea es vital, a menudo se realiza en un procesador digital de señales (DSP) sobre la misma placa para prevenir interferencias. El controlador de movimientos utiliza las trayectorias que calcula para determinar el comando de

torque adecuado para enviar al amplificador del motor y producir el movimiento efectivo.

El controlador de movimientos también debe cerrar el lazo de control. Debido a que esto requiere un alto nivel de determinismo y es vital para una operación consistente, ese lazo se cierra normalmente en la misma placa. Junto con el cierre del lazo de control, el controlador de movimientos también administra el control supervisarlo mediante el monitoreo de los límites y las paradas de emergencia para asegurar una operación segura. Dirigiendo cada una de estas operaciones para que ocurran en la placa o en un sistema operativo de tiempo real asegura una gran confiabilidad, determinismo, estabilidad y la seguridad necesaria para crear un sistema de control de movimientos que trabaje de manera correcta.

3.1.4 Cálculo de Trayectoria

La trayectoria de movimiento describe la señal de salida de la placa controladora o el comando hacia el manejador/amplificador, resultando en una acción de motor/movimiento que sigue el perfil. El control de movimientos típico calcula los segmentos de la trayectoria del perfil de movimientos en base a los valores de parámetros que se programan. El controlador de movimientos usa la posición final buscada, la máxima velocidad a obtenerse y los valores de aceleración dados para determinar cuánto tiempo invertirán en los tres segmentos primarios de movimiento, los que incluyen aceleración, velocidad constante y desaceleración.

Los perfiles de velocidad se utilizan para trazar las geometrías deseadas, estos perfiles son creados especificando parámetros del movimiento como lo son la máxima velocidad, máxima aceleración y la máxima desaceleración. Estos parámetros son utilizados para generar perfiles trapezoidales o curvas "s". El generado de trayectorias en un controlador de movimiento asegura que estas condiciones nunca son violadas mientras se realiza un movimiento. La salida del

generador de trayectorias son los valores de referencia que alimentan al lazo de control, que actúan en base a las necesidades del perfil deseado.

Para el segmento de aceleración de un perfil trapezoidal típico, el movimiento comienza a partir de una posición de parada, o del movimiento previo, y sigue una rampa de aceleración predeterminada hasta que la velocidad alcanza la velocidad buscada para el movimiento.

3.1.4.1 Perfil Trapezoidal

Cuando un cuerpo realiza un desplazamiento de un punto “A” a un punto “B”, saliendo del estado del reposo y llegando nuevamente a este, primeramente debe acelerar hasta llegar a una velocidad máxima, hacer el recorrido a esta velocidad, llegando a una posición en la cual debe comenzar a desacelerar, hasta alcanzar el reposo en el punto “B”. La gráfica de este desplazamiento se conoce como perfil de trayectoria como la mostrada en la fig 3-1

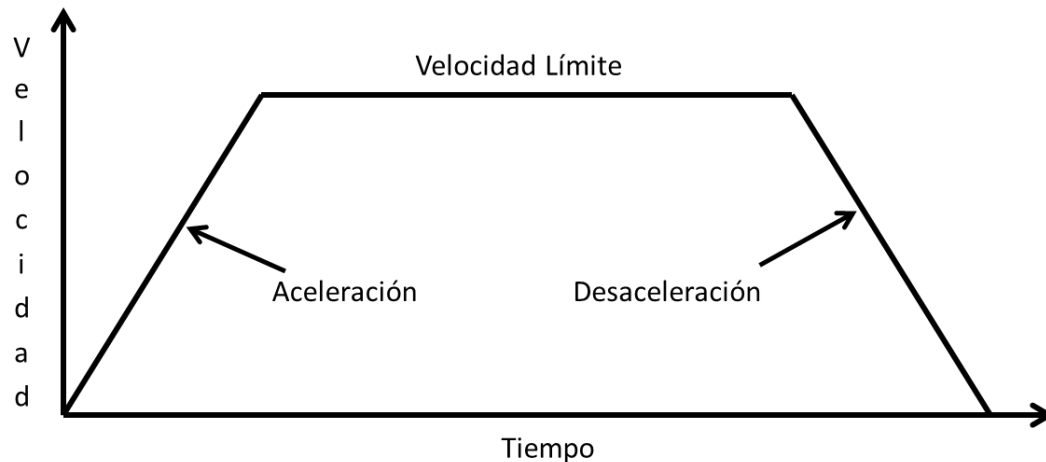


Figura 3-1. Gráfica de perfil de velocidad trapezoidal

Este perfil de movimiento puede ser calculado utilizando las siguientes ecuaciones de cinemática para una aceleración constante:

$$v = v_o + at \quad (1)$$

$$x = x_o + \frac{1}{2}(v_o + v)t \quad (2)$$

$$x = x_o + v_o t + \frac{1}{2}at^2 \quad (3)$$

$$v^2 = v_o^2 + 2ax \quad (4)$$

En donde x es la posición o distancia, v la velocidad, a la aceleración y t el tiempo.

3.1.5 Componentes de un sistema de control de Movimiento

Software de aplicación – Se utiliza un software de aplicación para indicar posiciones deseadas y perfiles de control de movimiento. En este caso se utiliza el software Matlab para calcular los perfiles de movimiento necesarios a partir de los parámetros del sistema.

Controlador de movimiento – El controlador de movimiento actúa como cerebro del sistema. Toma los perfiles de las posiciones y movimientos deseados y crea las trayectorias que deberán seguir los motores. Después entrega una señal a los servomotores.

Amplificador o drive – Los drives toman los comandos del controlador y generan la corriente necesaria para dirigir o girar el motor.

Motor – Los motores convierten la energía eléctrica en energía mecánica y producen el torque requerido para moverse a la posición deseada.

Elementos mecánicos – Los motores están diseñados para proporcionar torque a algunos dispositivos mecánicos. Éstos incluyen deslizadores lineales, brazos robóticos y actuadores especiales.

Dispositivo de retroalimentación o sensor de posición – Algunas aplicaciones de control de movimiento no requieren ningún dispositivo de retroalimentación de posición, pero para los servomotores sí es vital. El dispositivo de retroalimentación, generalmente un encoder de cuadratura, detecta la posición del motor y reporta el resultado al controlador, y de esa manera cierra el lazo con el controlador de movimiento.

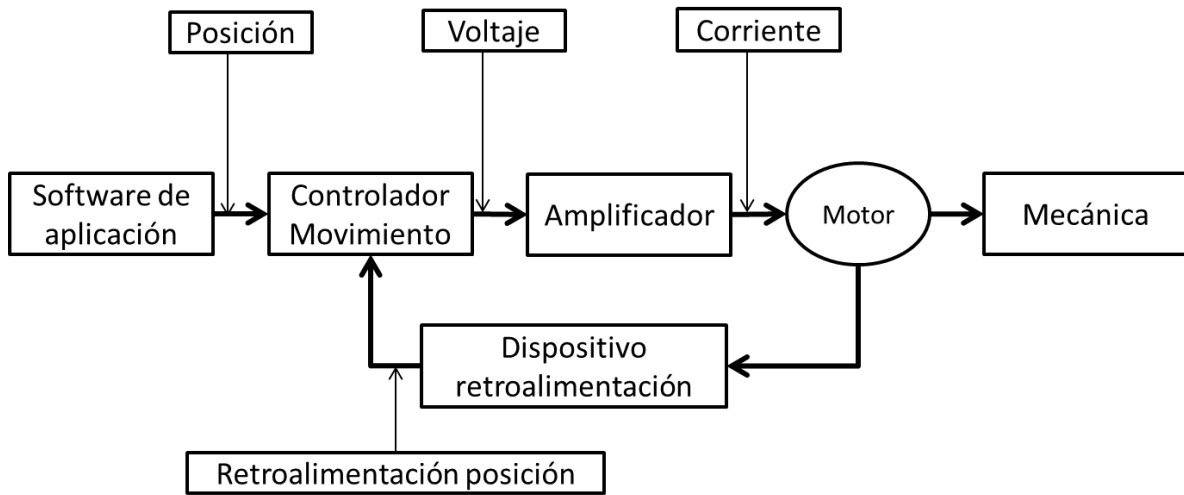


Figura 3-2. Componentes de un sistema de control de movimiento

3.1.6 Decodificador Incremental (Encoder)

Un encoder es un dispositivo electrónico que es capaz de convertir un desplazamiento rotacional en una señal digital sin necesidad de un convertidor analógico digital. La medida de desplazamiento se realiza contando las interrupciones que ocasiona un disco ranurado a un haz de luz. (Baturone, 2016).

Los codificadores o encoders rotatorios son fabricados en dos formas básicas: el codificador absoluto en donde, una sola palabra digital corresponde a la posición

absoluta de un giro del eje, y el codificador incremental que produce pulsos cuando el eje gira, permitiendo la medición en relación al desplazamiento del eje.

Los encoders ópticos son ampliamente utilizados en aplicaciones donde se requiere la medición de la posición lineal o angular, la velocidad y dirección del movimiento. Estos no solo se utilizan como parte de las maquinas computarizadas sino también en muchos dispositivos de precisión de control de movimiento en procesos industriales para asegurar la calidad y seguridad de los equipos (Shetty *et al.*, 2010). El sistema utiliza 2 encoders incrementales con una resolución de 500 cuentas por revolución (CPR) y opera a un voltaje DC 5V a una corriente máxima de 200mA.



Figura 3-3. Encoder de motor de DC, resolución de 500 CPR a 5V y 200mA.

El encoder incremental proporciona normalmente dos formas de onda cuadradas desfasadas en 90 grados eléctricos, los cuales se nombran “canal A y canal B”. Con la lectura de un solo canal se dispone de la información correspondiente a la velocidad del eje, mientras que si simultáneamente se capta la señal B es posible discriminar el sentido de rotación en base a la secuencia de datos que producen ambas señales (SILGE Electrónica, 2011).

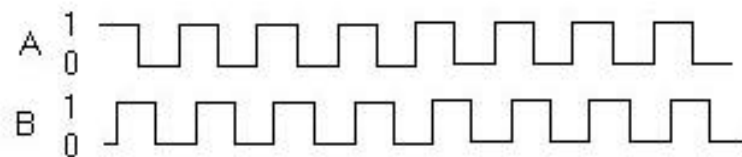


Figura 3-4. Señales de cuadratura de encoder, canales A y B

3.2 Metodología

3.2.1.1 *El método experimental*

El experimento dentro de los métodos empíricos resulta el más complejo y eficaz; este surge como resultado del desarrollo de la técnica y del conocimiento humano, como consecuencia del esfuerzo que realiza el hombre por penetrar en lo desconocido a través de su actividad transformadora.

El experimento es el método empírico de estudio de un objeto, en el cual el investigador crea las condiciones necesarias o adecua las existentes, para el esclarecimiento de las propiedades y relaciones del objeto, que son de utilidad en la investigación (González Río, 1997).

El objetivo del experimento puede ser: esclarecer determinadas leyes, relaciones o detectar en el objeto una determinada propiedad; para verificar una hipótesis, una teoría, un modelo. Un mismo experimento puede llevarse a cabo con variados fines. (Castillo, 2014)

El proyecto se hizo con base en los siguientes puntos de la metodología en donde se incluyen de manera general desde el estudio teórico, donde se aborda el estado del arte, el conocimiento del sistema; hasta el análisis de los resultados obtenidos, del cual estarán fundamentadas nuestras conclusiones.

3.2.1.2 *Documentación teórica.*

Este punto se abordó durante todo el periodo de la investigación, al inicio del proyecto se consultaron los trabajos existentes en la literatura con respecto al tema de FPGA, DSP y las características técnicas requeridas para el control de movimiento y que se presentan de manera detallada en el capítulo 3 de este documento. Durante el tiempo que duró la investigación se consultó el estado del arte, esto permitía tener una visión clara sobre el aporte que los resultados de la investigación nos podían dar.

3.2.2 Modelado e identificación del sistema

En este apartado se identificará el sistema de manera analítica a través de las ecuaciones diferenciales que lo modelan; además, se utiliza una identificación experimental con el método de la entrada de tipo escalón al sistema para medir su respuesta. Ambos resultados se comparan para conocer mejor los parámetros y propiedades del sistema.

3.2.2.1 *Modelo matemático*

Para desarrollar el modelo matemático nos centraremos en los motores DC de imanes permanentes con escobillas, se describen las ecuaciones diferenciales que describen su dinámica, su función de transferencia en el dominio de Laplace y su diagrama en bloques para su modelado y posterior análisis en Simulink complemento del software Matlab.

El modelado matemático del motor de corriente continua requiere de dos ecuaciones, una ecuación mecánica y otra eléctrica. Estas ecuaciones están acopladas y se basan en las leyes de la dinámica y de Kirchhoff, respectivamente. Por una parte, la ecuación mecánica modela el movimiento del rotor, y por otra parte la ecuación eléctrica modela el comportamiento del circuito eléctrico del inducido.

En la Fig. 3-5 se muestra el diagrama eléctrico de la armadura y la representación mecánica de las partes móviles del motor.

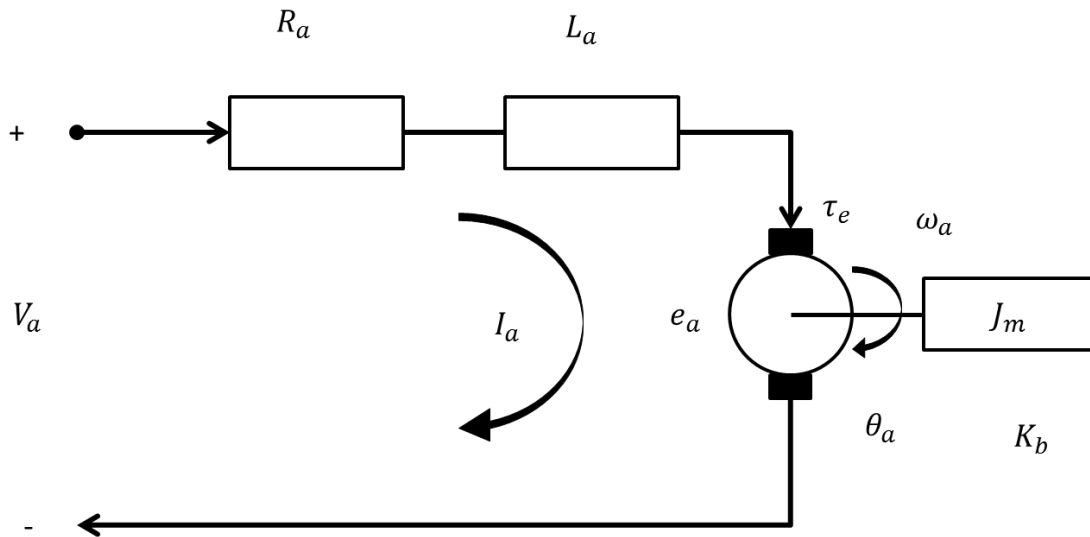


Figura 3-5. Modelo del motor de imanes permanentes de DC.

Donde nuestras variables son:

Voltaje de armadura (V_a)

Corriente en la armadura (I_a)

Resistencia eléctrica (R_a)

Inductancia eléctrica (L_a)

Voltaje inducido (e_a)

Torque eléctrico (τ_e)

Velocidad del rotor (ω_a)

Posición del rotor (θ_a)

Inercia del motor (J_m)

Fricción del sistema (k_b)

3.2.2.2 Ecuaciones Eléctricas

Utilizando las leyes de Kirchoff en el diagrama eléctrico obtenemos la ecuación diferencial:

$$V_a - V_{Ra} - V_{La} - e_a = 0 \quad (5)$$

Sabiendo que:

$$V_{Ra} = i_a R_a \quad (6)$$

$$V_{La} = L_a \frac{di_a}{dt} \quad (7)$$

$$e_a = k_e \omega_a \quad (8)$$

Con k_e como la constante de la fuerza contra electromotriz, sustituyendo (6), (7) y (8) en (5) y despejando la derivada, obtenemos la ecuación diferencial de la corriente en la armadura.

$$\frac{di_a}{dt} = -\frac{R_a}{L_a} i_a - \frac{k_e}{L_a} \omega_a + \frac{V_a}{L_a} \quad (9)$$

3.2.2.3 Ecuaciones mecánicas

Utilizando la segunda ley de Newton, en la que la suma de las fuerzas en un sistema debe ser igual a cero se obtiene:

$$T_e - T_{\dot{\omega}} - T_{\omega} - T_L = 0 \quad (10)$$

Donde

Torque electromagnético (T_e)

Torque debido a la aceleración del rotor ($T_{\dot{\omega}}$)

Torque debido a la velocidad del rotor (T_{ω})

Torque debido a la carga (T_L)

Debido a que el torque electromagnético es directamente proporcional a la corriente en la armadura y a una constante de torque k_T podemos obtener:

$$T_e = k_T i_a \quad (11)$$

$$T_{\dot{\omega}} = J_m \frac{d\omega_a}{dt} \quad (12)$$

$$T_{\omega} = k_b \omega_a \quad (13)$$

Sustituyendo (11), (12) y (13) en (10) y despejando nuestra derivada obtenemos la ecuación diferencial de la velocidad de la armadura:

$$\frac{d\omega_a}{dt} = \frac{k_T}{J_m} i_a - \frac{k_b}{J_m} \omega_a - \frac{T_L}{J_m} \quad (14)$$

Aplicando la transformada de Laplace a nuestras ecuaciones (9) y (14), considerando las condiciones iniciales igual a cero tendremos:

$$sI_a(s) = -\frac{R_a}{L_a} I_a(s) - \frac{K_e}{L_a} \Omega_a(s) + \frac{V_a(s)}{L_a} \quad (15)$$

$$s\Omega_a(s) = \frac{k_T}{J_m} I_a(s) - \frac{k_b}{J_m} \Omega_a(s) - \frac{T_L(s)}{J_m} \quad (16)$$

En la Figura 3-6 se muestra la representación en bloques de las ecuaciones (15) y (16), se agregó un bloque de integración a la salida de la velocidad para conocer posición, de esta manera se tiene un modelo en el que se exponen las variables para el control de movimiento.

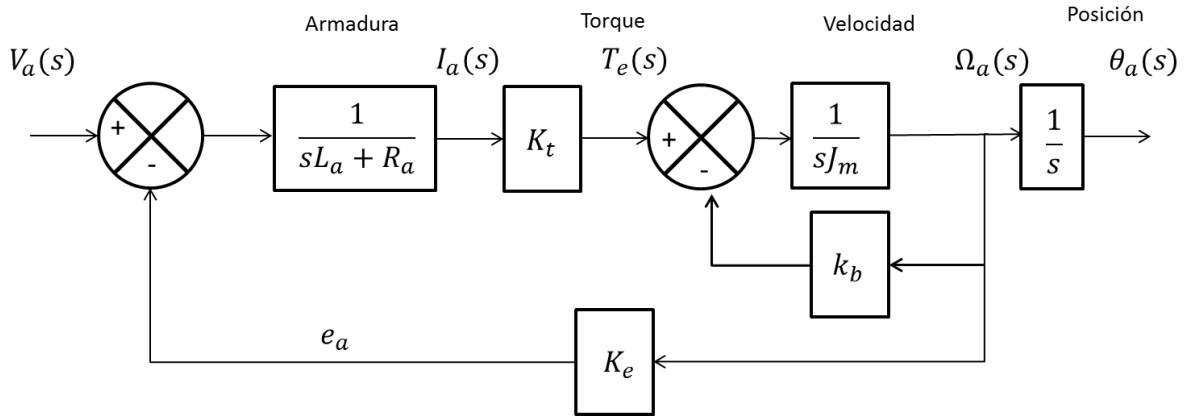


Figura 3-6. Diagrama de bloques de un motor DC.

Sin considerar el integrador para la posición, la función de transferencia resultante es:

$$\frac{\Omega_a}{V_a} = \frac{k_T}{J_m s^2 + (J_m R_a + k_b) s + (k_T k_e)} \quad (17)$$

3.2.3 Modelo del motor con la corriente como salida

Una de las ventajas adicionales del control en modo corriente incluye la limitación inherente de corriente al fijar el comando en el amplificador de error. Partiendo de la fig. 3-6 podemos manipular el sistema para obtener el modelo del motor con el voltaje como entrada y la corriente como salida.

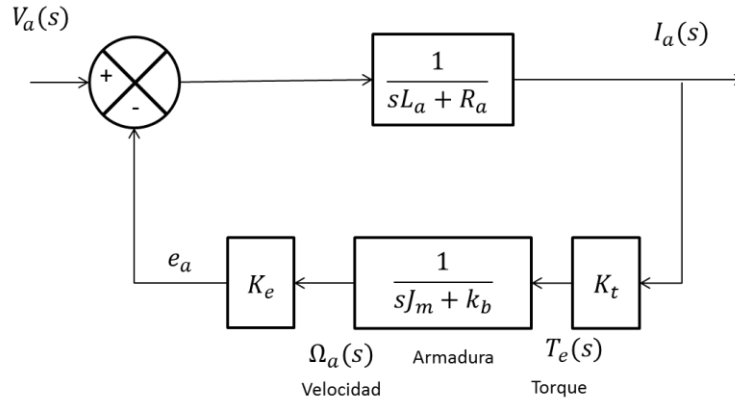


Figura 3-7. Diagrama a bloques de modelo del motor DC en modo corriente.

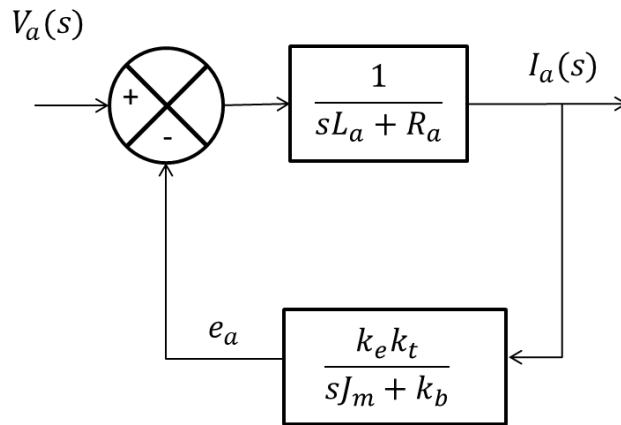


Figura 3-8. Modelo de motor DC con entrada de voltaje y corriente como salida.

3.2.4 Parámetros del motor

Tabla 1. Parámetros del motor de DC

Parámetros	Unidad	Variable	Motor DC
Resistencia	Ω	R_a	0.309
Inductancia	mH	L_a	83.9
Constante de torque	Nm/A	k_t	0.02
Constante de voltaje	V/(rad/s)	k_e	0.0549

Inercia del rotor	Kgm^2	J_m	$13.8 \cdot 10^{-6}$
Constante de amortiguamiento	Kgm^2/s	K_b	$17.48 \cdot 10^{-6}$
Voltaje nominal	V	VDD	20
Corriente nominal	A	I_{\max}	59
Velocidad nominal sin carga	RPM	ω_{\max}	3500
Resolución Encoder	Pulsos por rev	P_{pr}	400

Una vez conocidos los parámetros del motor, estos se introducen en el software Matlab/Simulink, para realizar el análisis del modelo.

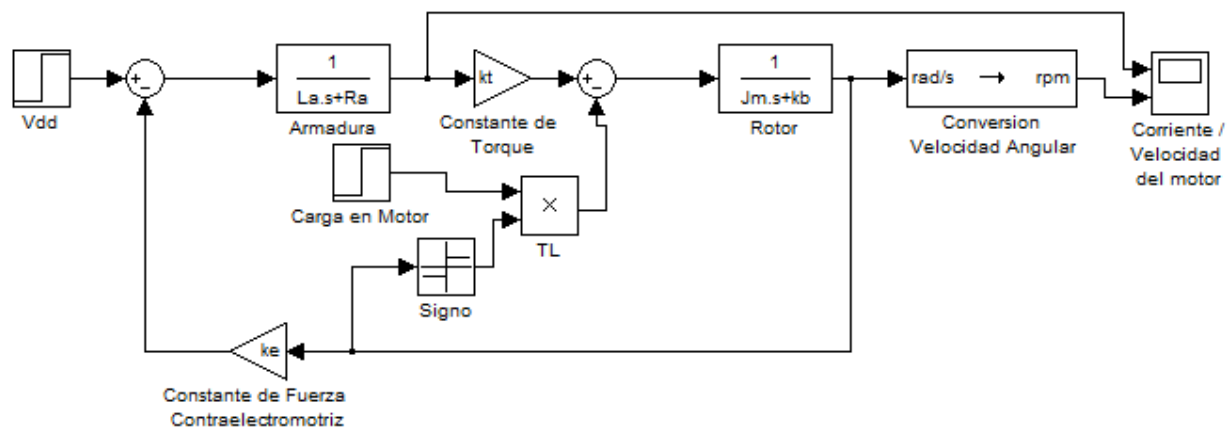


Figura 3-9. Modelo de motor DC en software Matlab/Simulink.

El modelo es validado aplicando una entrada escalón con el valor del voltaje nominal del motor y con $T_L = 0$ para no considerar una carga en el mismo.

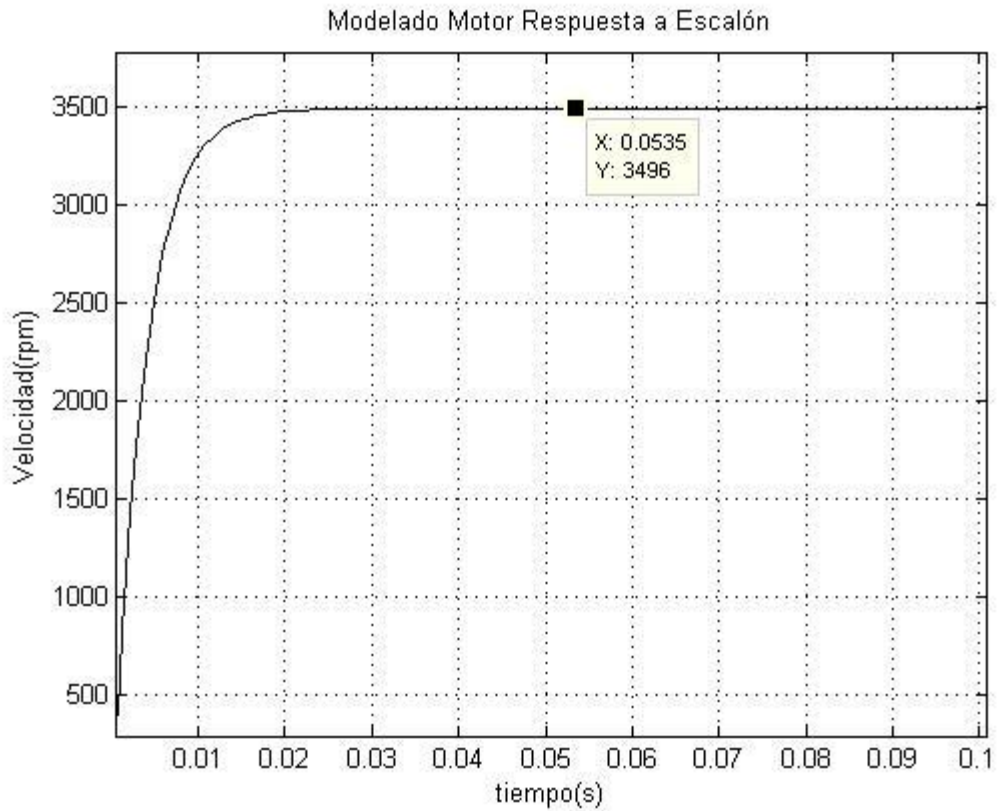


Figura 3-10. Respuesta del motor de DC a una entrada escalón

En la Fig 3-10 se observa la respuesta del modelo a la entrada escalón, obteniendo un valor máximo de 3496 rpm, obteniendo un error del 0.1% en comparación con los parámetros dados en la hoja de datos del fabricante. (Tabla 1)

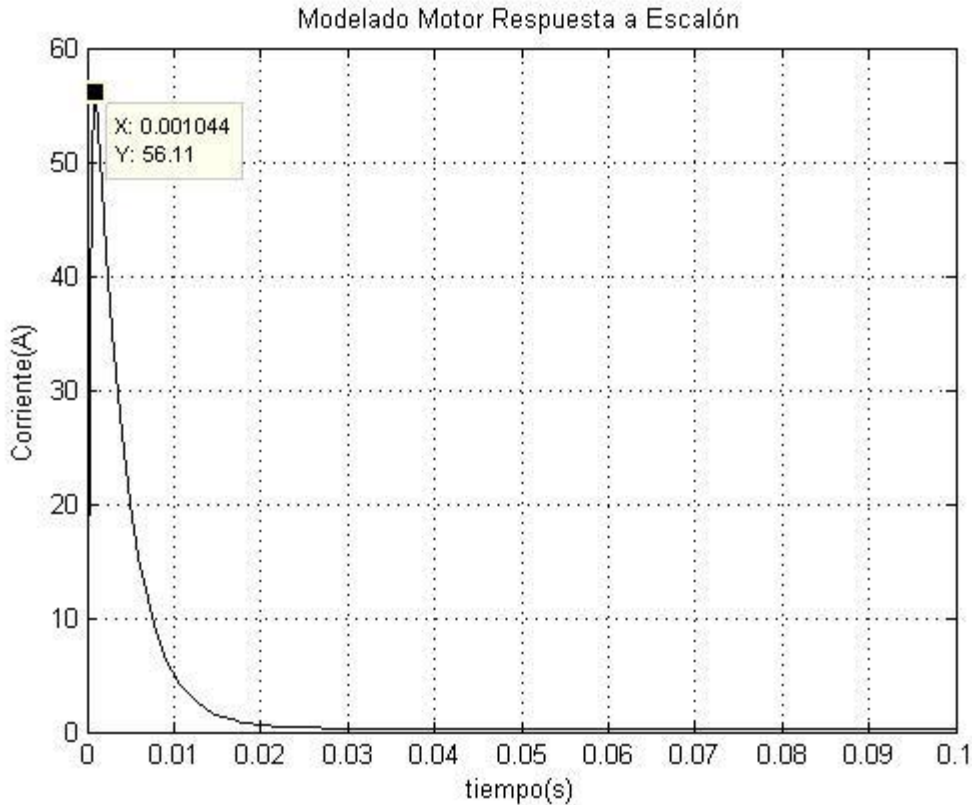


Figura 3-11. Respuesta de la corriente al aplicar el voltaje nominal al motor.

En la Fig 3-1 se puede observar el comportamiento de la corriente al arranque del motor. En la simulación se obtiene un valor de 56.11A. En comparación con los parámetros de la hoja de datos del motor se obtiene un error de 4.8%.

3.2.5 Diseño de controlador

Un control automático compara el valor efectivo de salida de una planta con el valor deseado, determina la desviación y produce una señal de control que reduce la desviación a cero o a un valor pequeño. La forma en que el control produce la señal recibe el nombre de “acción de control”.

Seguidamente veremos las acciones de control utilizadas en los controles automáticos industriales. Veremos también las formas de compensar los servosistemas, utilizando diversas técnicas.

A menudo se emplean especificaciones de diseño para describir qué debe hacer el sistema y cómo hacerlo. Estas especificaciones son únicas para cada aplicación individual y con frecuencia incluyen especificaciones como estabilidad relativa, respuesta transitoria y características de respuesta en frecuencia.

El diseño de sistemas de control se puede realizar ya sea en el dominio del tiempo o en el dominio de la frecuencia. Por ejemplo, la precisión en estado estable a menudo se especifica con respecto a una entrada escalón, una rampa o una parábola, y el diseño para cumplir ciertos requisitos es más conveniente realizarlo en el dominio del tiempo. Otras especificaciones como el sobre impulso máximo, tiempo de subida o tiempo de estabilización, están definidas para una entrada escalón unitario, y por lo tanto se emplean para el diseño en el dominio del tiempo.

De acuerdo con su acción de control, los controles típicos industriales se pueden clasificar en (Bolton, 2001):

- Controles de dos posiciones o de sí-no
- Control Proporcional (P)
- Control Integral (I)
- Control Proporcional e integral (PI)
- Control Proporcional y derivativo (PD)
- Control Proporcional, integral y derivativo (PID)

En el proyecto se seleccionó el control Proporcional y derivativo, se invita a revisar el teorema de valor final para este tipo de controladores y que se dé cuenta de porque con un controlador de este tipo se puede asegurar que el error se reducirá a cero en estado estable para un sistema de tipo 1 como lo menciona Bolton (2001) y como se puede ver en la figura 57.

La acción de control proporcional y derivativo queda definida por la siguiente ecuación:

$$m(t) = Kp * e(t) + Kp T_D \frac{de(t)}{dt} \quad (18)$$

Y la función de transferencia después de aplicar la transformada de Laplace es:

$$\frac{M(s)}{E(s)} = Kp(1 + T_D s) \quad (19)$$

Donde Kp es la ganancia proporcional y T_D es el tiempo derivativo. La acción de control derivativa, a veces llamada control de velocidad, es aquella en que la salida de control es proporcional a la velocidad de variación de la señal de error actuante. El tiempo derivativo es el intervalo de tiempo en que la acción de velocidad se adelanta al efecto de acción proporcional.

Se diseñó un controlador por medio de la metodología propuesta en la cual se van a realizar los cálculos por medio del software: MATLAB R2011a

Se realizó un *script* en donde se realizan los pasos de la metodología (Anexo7-2)

- Cálculo de Polos
- Cálculo de Polos en lazo cerrado
- Cálculo de Ángulos
- Cálculo de Z
- Cálculo de T_d
- Cálculo de Norma de controlador
- Cálculo de la planta
- Cálculo de Ganancia Kp

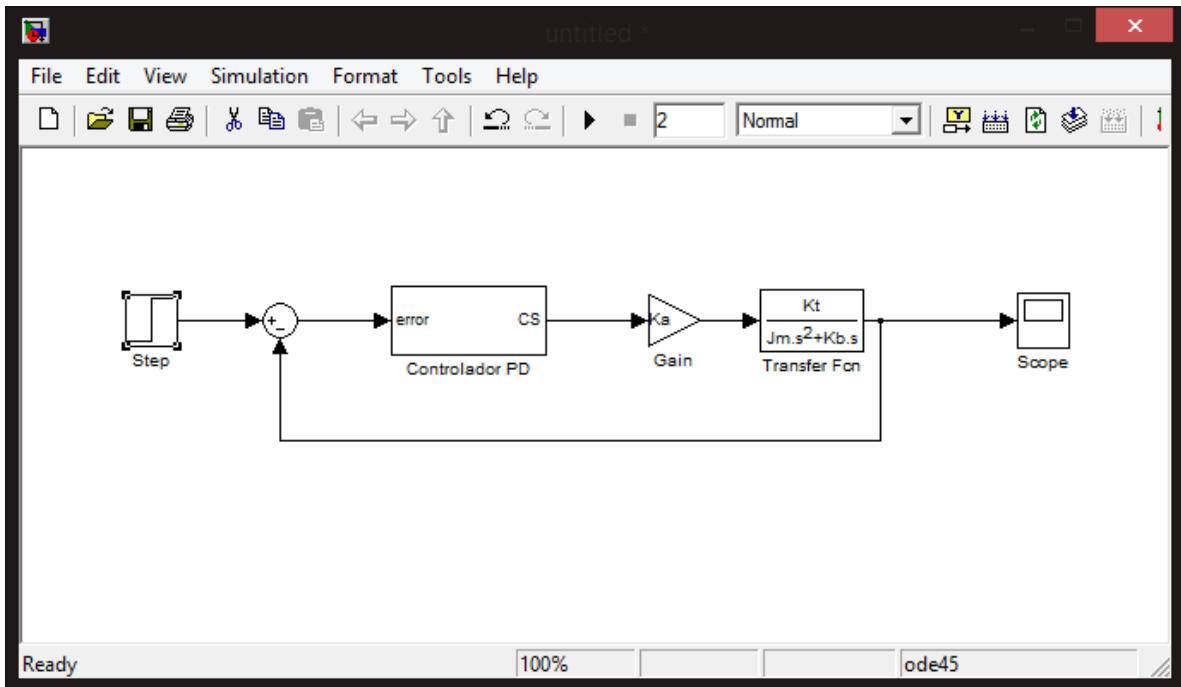


Figura 3-12. Modelo de motor con controlador PD, simulación en software Simulink.

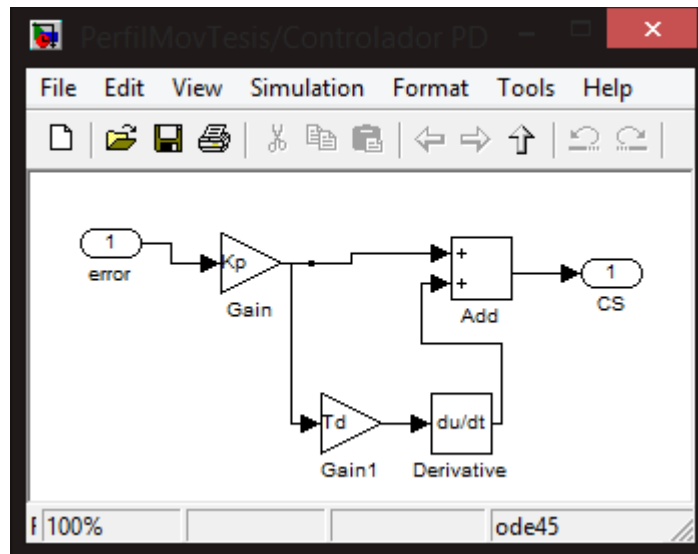


Figura 3-13. Controlador PD implementado en Simulink.

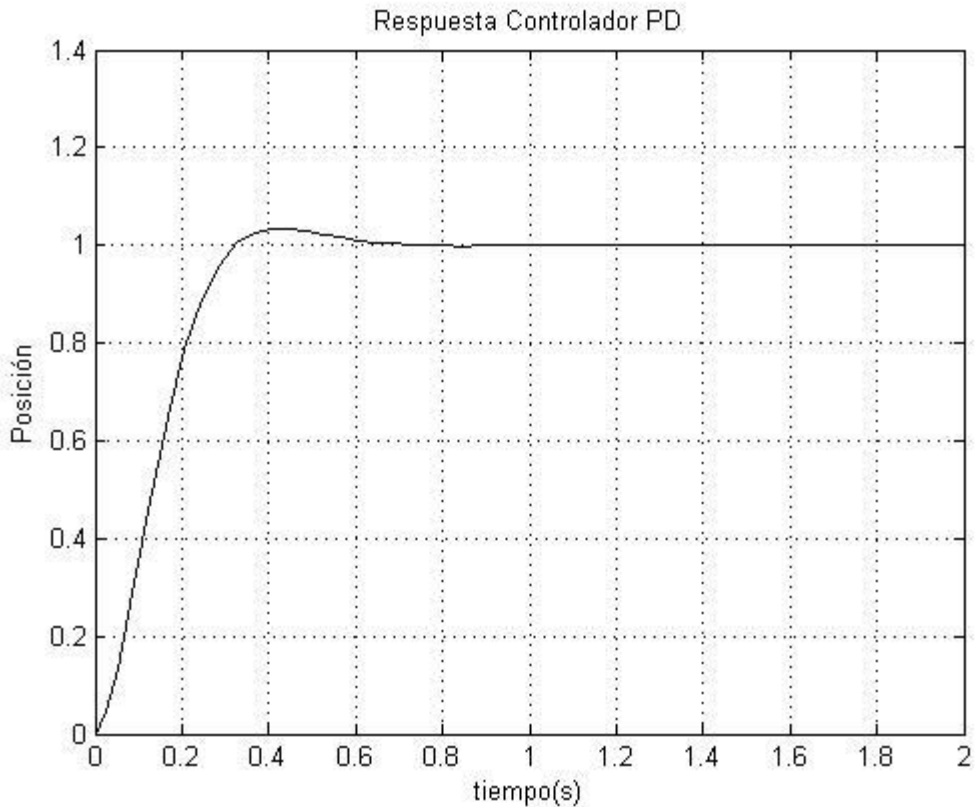


Figura 3-14. Respuesta del motor de DC a una entrada escalón con un controlador PD.

El *script* generado, realiza un reporte de los cálculos realizados en un block de notas, el cual contiene los datos del motor, la función de transferencia del motor, los cálculos que se realizaron paso a paso según la metodología.

3.2.6 Generador Perfil Trapezoidal

La trayectoria de movimiento caracteriza el control de la tarjeta del controlador de movimiento o la señal de comando de salida al amplificador; y como resultado un motor o una acción de movimiento siguen ese perfil. El controlador de movimiento típico calcula los segmentos de la trayectoria del perfil de movimiento en base a los valores de parámetros que usted programe. El controlador de movimiento utiliza los valores de la posición especificada deseada, la velocidad máxima que

se desea alcanzar y la aceleración que usted proporciona para determinar cuánto tiempo empleará en los tres segmentos de movimiento principales (los cuales incluyen aceleración, velocidad constante y desaceleración).

Para el segmento de aceleración de un perfil típico trapezoidal, el movimiento comienza desde una posición de detenimiento o desde un movimiento previo y sigue una rampa de aceleración indicada hasta que la velocidad alcanza la velocidad deseada para el movimiento.

La sintonización del controlador se realizará por el método de frecuencia de corte, en donde por medio de la siguiente metodología podemos realizar la sintonización.

Teniendo los datos de entrada: Margen de fase (θ_m), frecuencia de corte (ω_c) y tiempo de muestreo (T_s) se puede calcular la ganancia de la planta.

$$A = |G(j\omega)| \quad (20)$$

$$\alpha = \text{Arg}(G(j\omega)) \quad (21)$$

Posteriormente se calcula el ángulo β

$$\beta = \theta_m - 175^\circ - \alpha + \frac{180^\circ \omega_c T_s}{2\pi} \quad (22)$$

Y con el cálculo de los ángulos α y β se puede calcular los parámetros proporcional y derivativo del controlador:

$$P = \frac{1}{A} \cos \beta \quad (23)$$

$$D = \frac{1}{\omega_c A} \sin \beta \quad (24)$$

Se realizó un script para sintonizar un controlador por medio de la metodología propuesta y encontrar los valores requeridos para el perfil de velocidad, se van a realizar los cálculos por medio del software: MATLAB R2011a

Pasos de la metodología:

- Cálculo de aceleración
- Cálculo de par requerido
- Cálculo de corriente
- Cálculo de potencia

Para la sintonización:

- Cálculo de ganancia de la planta
- Cálculo de ángulos beta y alfa
- Cálculo de la parámetros

3.2.7 Diseño y fabricación de PCB.

Se desarrolló la plataforma de control de movimiento basándonos en la misma metodología de desarrollo de Software, pero esta vez adaptada a la parte del desarrollo de la tarjeta electrónica:

•*Análisis*: En esta parte se analizaron otras tarjetas similares que se puedan utilizar para proponer un nuevo diseño. Se obtuvo la información de diseños previos tanto de DSP como de FPGA, así como las ventajas y desventajas de tarjetas comerciales como la basys2 y el kit Delfino de Texas Instruments.

•*Diseño*: Se reunió la información de los componentes que se requieren para el desarrollo de la plataforma. Los módulos que se proponen aparte de los componentes principales son entradas digitales de forma de interruptores y de tipo botón, entradas analógicas: dos potenciómetros, salidas digitales: ocho diodos emisores de luz independientes, así como los pines dedicados para conectar una pantalla LCD de formato 16x2.

•*Discusión sobre tecnología a implementar:* En esta parte se define sobre las marcas y modelos de los componentes principales para el uso del control de movimiento, conviene justificar el uso de cada componente. Se analizaron los componentes utilizados en la tarjeta como lo es el DSP de Texas Instruments TMS320F28335 y para el FPGA modelo Spartan XC3S250E de Xilinx.

•*Implementación:* En esta sección de la metodología se realiza el diseño CAD de las pistas, el acomodo de los componentes en la PCB y todo el diseño en la computadora, una vez terminada esta parte se manda maquilar la tarjeta y se montan los componentes sobre la misma.

•*Depuración:* se evalúa el funcionamiento de la tarjeta haciendo pruebas a cada parte de la tarjeta para verificar que todo haya sido trabajado correctamente.

3.2.7.1 *Diseño CAD*

El gran impacto que han producido las herramientas CAD (*Computer Aided Design*) sobre el diseño de circuitos electrónicos ha producido que un gran número de compañías dedicadas a la fabricación de software hayan sacado al mercado programas dedicados a esta rama.

La primera tarea a realizar dentro del diseño de hardware, es realizar una descripción de lo que se pretende desarrollar. Las computadoras ofrecen hoy en día herramientas especiales para la creación y verificación de los diseños. Como es el caso de EagleSoft que permite realizar diseños esquemáticos, para posteriormente pasar los diagramas a diseños de circuitos impresos.

La fig. 3-15 muestra el flujo del diseño de la PCB (Printed Circuit Board) de este proyecto.

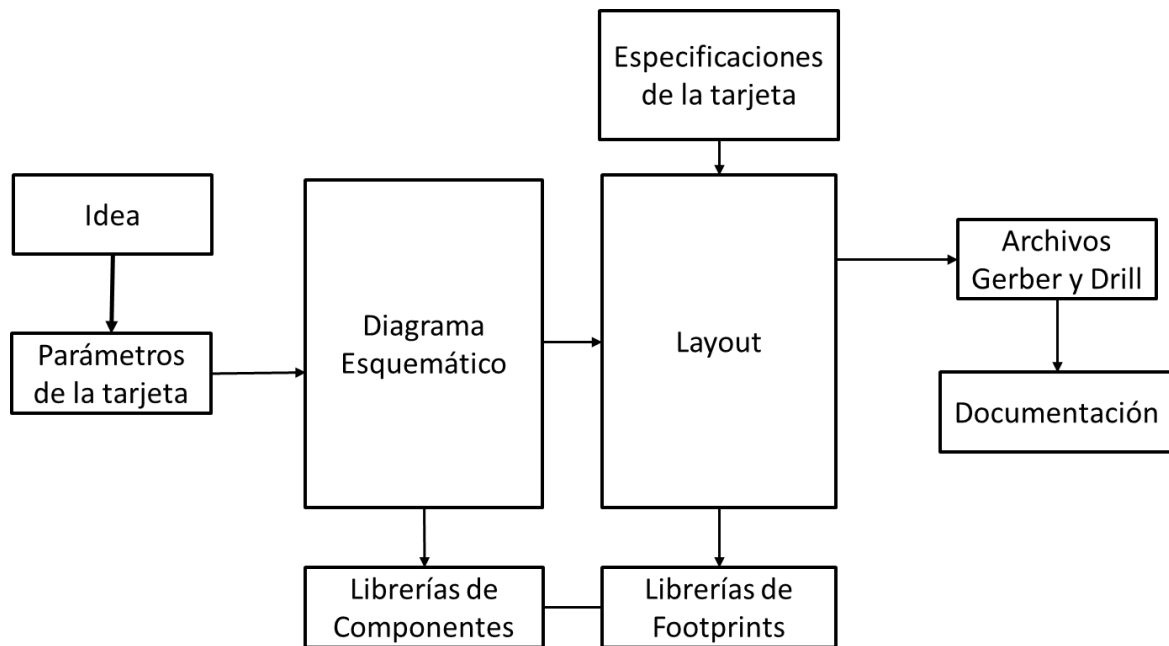


Figura 3-15. Flujo de diseño de PCB de plataforma electrónica

3.2.7.2 *Parámetros de diseño*

Al empezar el diseño de una PCB hay que tener en cuenta los requerimientos funcionales impuestos por el sistema, por el usuario final, por el aspecto económico comercial del proyecto. Pero además hay una serie de factores limitados por las reglas de diseño como: requerimientos electrónicos (voltajes, ganancias, impedancias), tamaño, componentes, ensamble; y por las facilidades de que dispone el fabricante como son: maquinaria, herramientas, entre otras; que se deben tener en cuenta para el diseño de la PCB.

Conocido el diagrama esquemático, se procede a la distribución de los componentes del circuito, colocando los elementos con la idea de reducir los puntos de cruce de las interconexiones.

Todos estos requisitos son tomados en cuenta como guía para realizar el diseño de la tarjeta:

- a) Estudio de diagrama esquemático y de la lista de componentes.
 - Parámetros dimensionales de la tarjeta y componentes.
 - Parámetros eléctricos y mecánicos.
- b) Asignar posiciones a todos los elementos fijos del circuito.
- c) Buscar una distribución geométrica uniforme de los componentes.
- d) Fijar los tamaños de pistas, agujeros y conexiones de los componentes en la tarjeta. Considerar los requerimientos eléctricos, mecánicos y ambientales como son:
 - Resistencia y propiedades dieléctricas para componentes de frecuencia o impedancia mayor.
 - Caídas de tensión y elevación de temperatura de los conductores y componentes.
 - Condiciones ambientales de funcionamiento y almacenamiento (humedad, polvo, temperatura, vibraciones, etc).
 - Distribución uniforme del peso de los componentes en la superficie de la tarjeta.
- e) Tener en cuenta en la colocación de los componentes, la separación necesaria de los mismos para hacer factible la utilización de las herramientas necesarias para la operación de ensamble. Los componentes no deben interferirse físicamente, de acuerdo con los requisitos eléctricos, térmicos y de reparación.

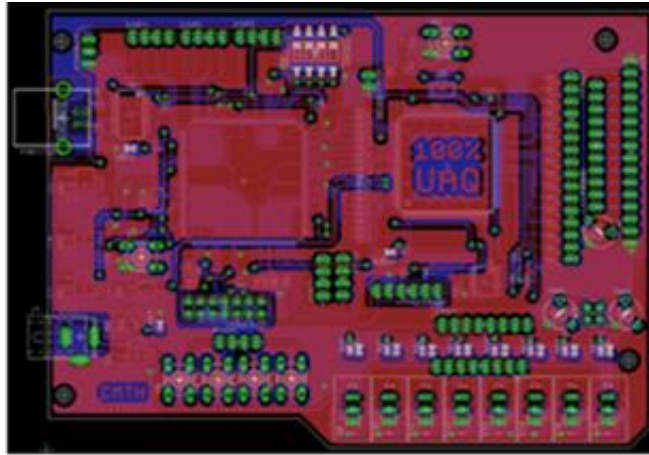


Figura 3-16. Diseño CAD de la plataforma creada en Eagle

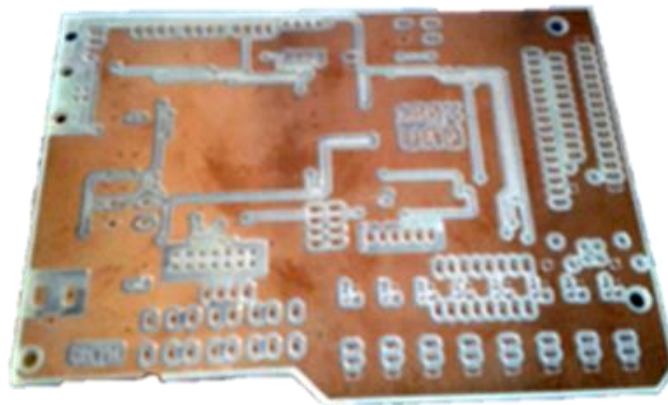


Figura 3-17. Fabricación del diseño de circuito impreso en baquelita.

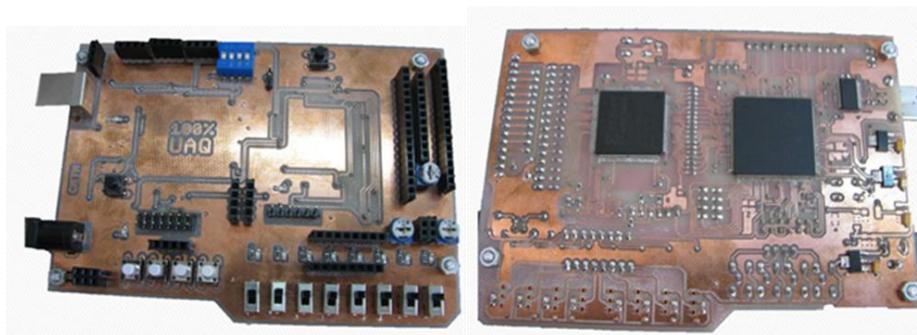


Figura 3-18. Tarjeta con componentes montados y soldados vista superior (izquierda) vista inferior (derecha).

Evolución de la plataforma según la metodología propuesta (Diseño en Eagle, tarjeta maquilada, montaje de componentes)

3.2.8 Arquitectura de la plataforma

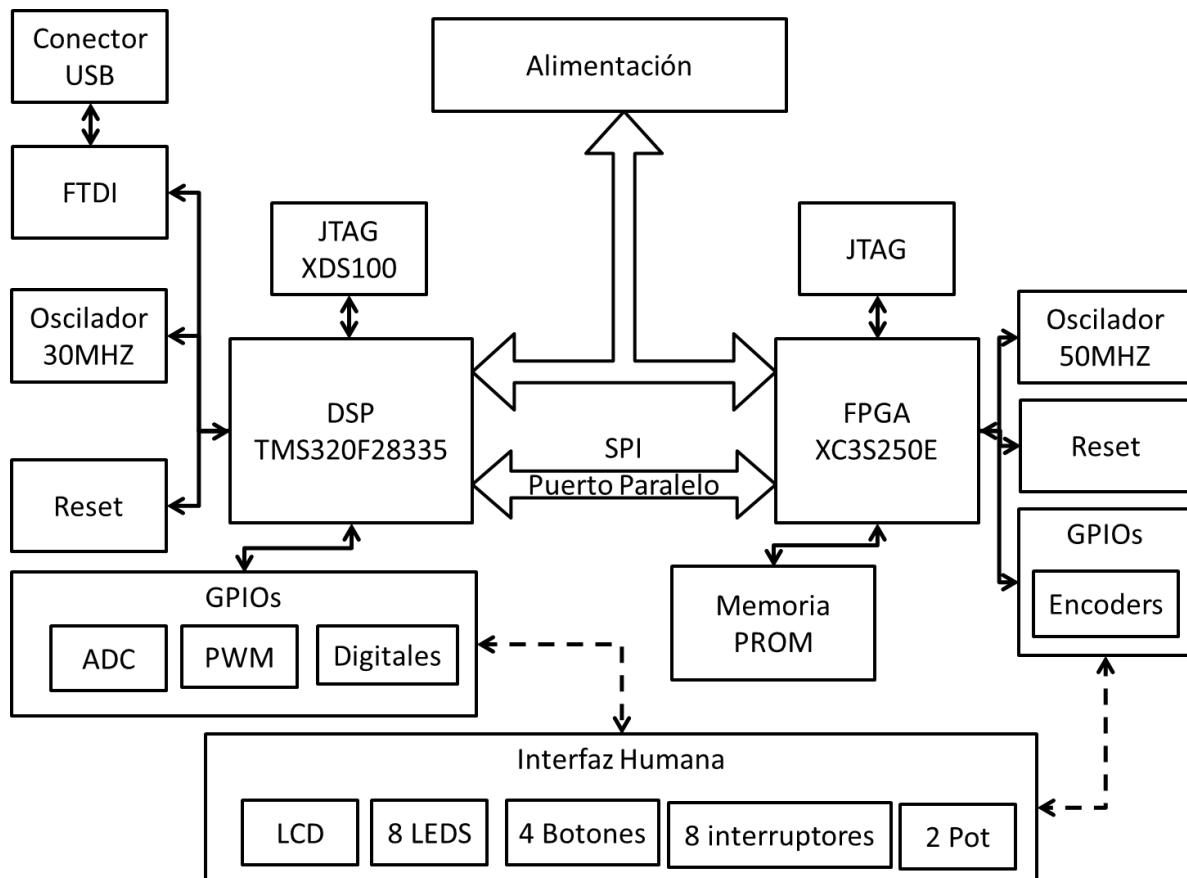


Figura 3-19. Arquitectura propuesta de la plataforma electrónica

Los algoritmos de control utilizados en la actualidad requieren alta potencia de cálculo para que el procesamiento de señales más eficiente, una respuesta más rápida a las interrupciones y el acceso a las ranuras de memoria. Los sistemas como microcontroladores han comenzado a estar limitada a las exigencias actuales de los sistemas de control, por nombrar algunas aplicaciones, las

transformadas rápidas de Fourier, PWM, las estimaciones de las variables de estado, protocolos de comunicación con un rendimiento satisfactorio, entre otros.

La combinación de las ventajas de un DSP y un alto rendimiento, bajo consumo de energía, y una buena relación coste-beneficio ha permitido que esta tecnología sea más común en las soluciones de control. El DSP TMS32028335 tiene características y funciones tales como:

- Voltaje GPIO a 3.3V
- Frecuencia de operación de hasta 150 MHz
- 32x32 -16x16 MAC MAC dual
- 6 canales DMA
- Flash 256K de memoria x16
- 63 GPIO
- E / S dedicada a un PWM, codificadores y puertos seriales.

Las aplicaciones no siempre son iguales, por lo que tienen una tecnología digital que puede adaptarse a diferentes funciones es de gran importancia. Anteriormente, los sistemas tradicionales como el GAL se integró a los dispositivos para diferentes aplicaciones debido a que los dispositivos ASIC eran relativamente caros. Una implementación alternativa para este problema fue que cubre las necesidades de estructura digital en las gamas: medio y complejo. La tecnología FPGA con bajo consumo de energía, alta capacidad y la lógica reprogramable de procesamiento de alta velocidad hace posible los sistemas de hardware con mayor facilidad, por lo que el FPGA pronto se convirtió en una solución ideal para los negocios. La FPGA XC3S250E tiene características como:

- Voltaje GPIO a 3.3V
- 250K compuertas en el sistema
- 12 multiplicadores dedicados
- 172 E / S
- 216K de RAM

La plataforma tiene la TMS320F28335 DSP, que puede ser programado a través de un emulador. El XDS100 emulador de Texas Instruments es una interfaz USB JTAG muy bajo costo, que es compatible con la programación

Code Composer Studio se utiliza principalmente para programar la memoria de lectura / escritura, lectura, programas de registros de carga, funcionamiento programas, igualmente ayuda a realizar la depuración de código.

El DSP ha dedicado para el protocolo de comunicación en serie a través del dispositivo FTDI, que puede enviar y recibir datos en el PC interfaz es de gran aporte para una interfaz más visual a través de este pin del puerto. Dentro de la plataforma, una opción permite para el DSP puede utilizar la plataforma FTDI, o se puede utilizar el pin de comunicación en serie con otro dispositivo.

Era el diseño de la plataforma para aprovechar GPIO viene DSP dedicada, como ADC, PWM, y la entrada y pines de salida de propósito general. En la misma manera, la FPGA tiene entrada y salida de pines para el uso general. Se utilizan estos pines para conectar diferentes dispositivos, y algunos de los más utilizados son incorporado en la plataforma, se usan interfaces para la conexión de LCD de tipo 16x2, tiene salidas digitales de plataforma de tipo LED, forma de entradas digitales y de push botón del tipo de interruptor y la plataforma cuenta con entradas analógicas de tipo potenciómetro.

El XC3S250E FPGA se puede programar a través de un USB interfaz JTAG, que es compatible con el software ISE de Xilinx y se usa principalmente para cargar el programa en el dispositivo. El dispositivo tiene una memoria PROM que sirve para guardar el programa, y se lleva a cabo incluso cuando se ha quitado la fuente de alimentación de poder.

3.2.9 Descripción de un sistema digital basado en DSP y FPGA.

Se desarrolla el programa de control de movimiento basados en la siguiente metodología:

Tabla 2. Metodología propuesta para el desarrollo del proyecto.

Análisis	<ul style="list-style-type: none">• En esta parte se reúne toda la información en relación a ¿para qué se va a utilizar?, ¿Quién lo va a utilizar?, ¿Qué variables son las más importantes para el usuario final?
Diseño	<ul style="list-style-type: none">• Se toman los requisitos obtenidos de la parte de análisis para realizar el diseño.
Discusión	<ul style="list-style-type: none">• En esta parte se define sobre que plataforma va a funcionar el programa.
Implementación	<ul style="list-style-type: none">• En esta sección de la metodología se escribe el código en el lenguaje de programación seleccionada, esta parte del proyecto debe llevar solo un pequeño tiempo del proyecto.
Depuración	<ul style="list-style-type: none">• Se evalúa el proyecto acompañado del usuario final para ver las posibles mejoras.

Para la sección de implementación del software se utilizarón los siguientes diagramas flujo, donde se puede ver claramente la aportación de cada uno de los dispositivos involucrados en la plataforma, estos diagramas los dividimos en 3 partes, DSP, PC y FPGA.

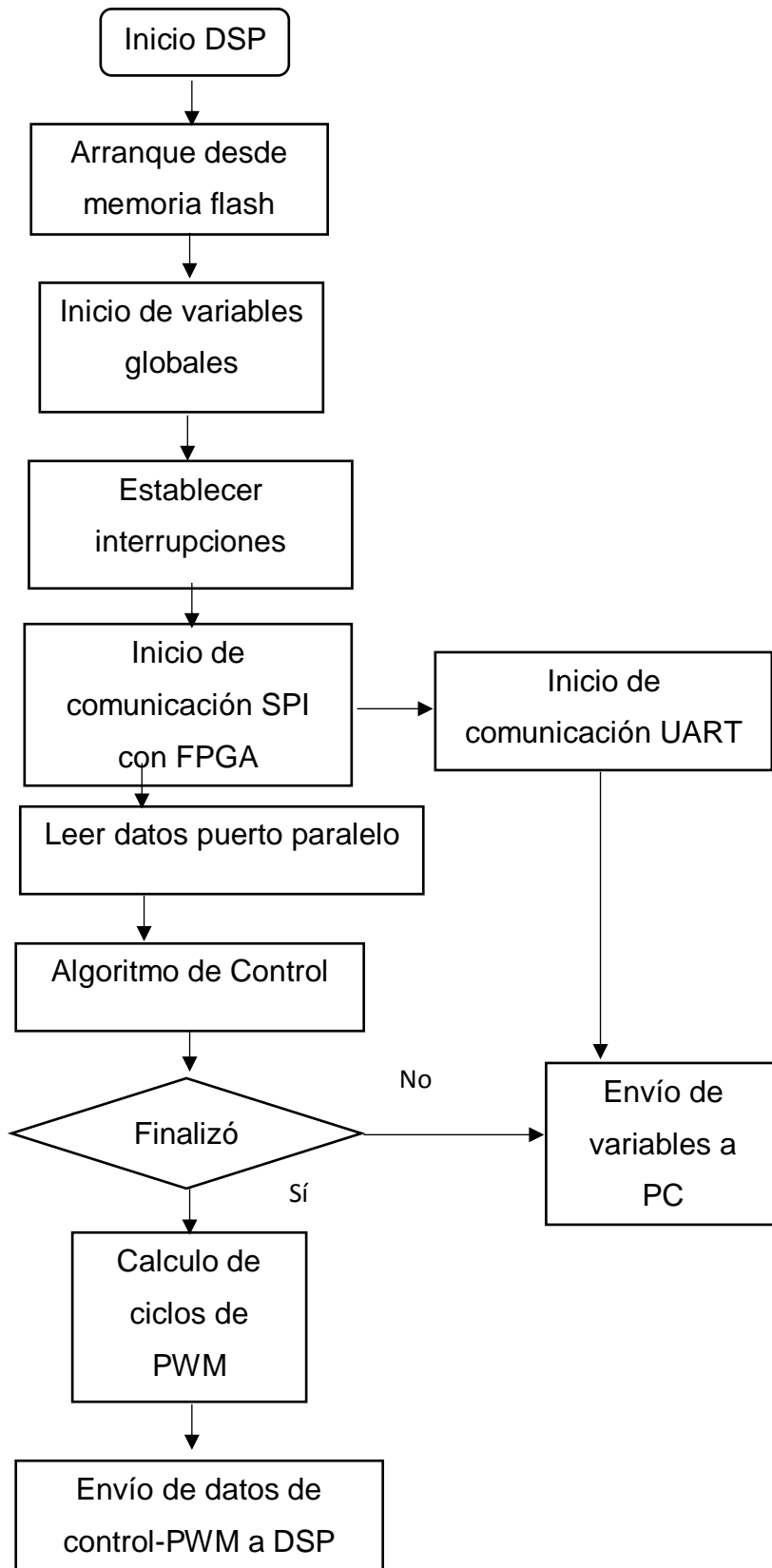


Figura 3-20. Diagrama de flujo del programa del DSP

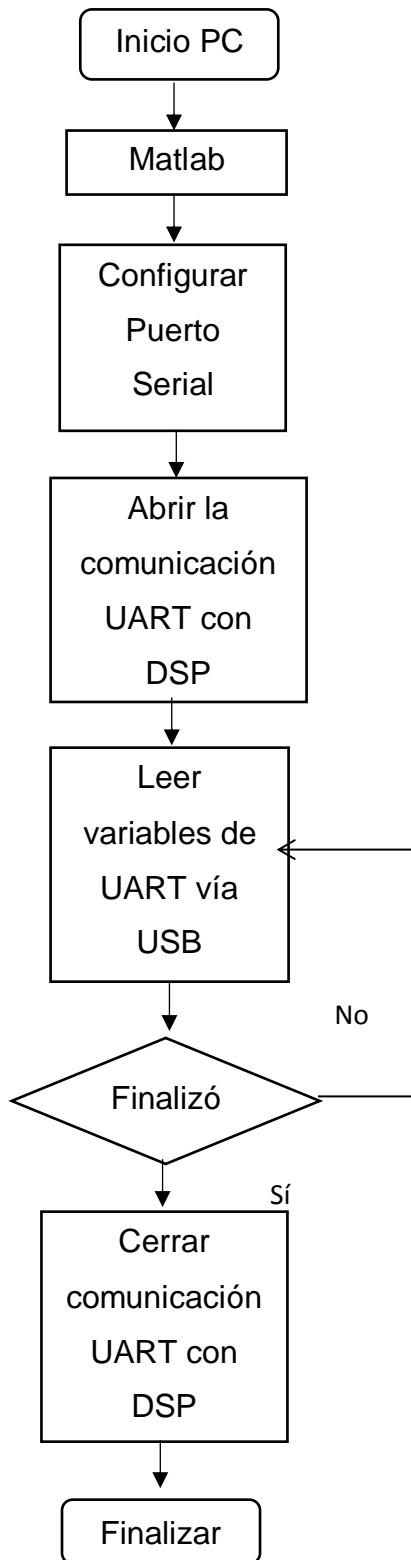


Figura 3-21. Diagrama de flujo de comunicación de PC con DSP

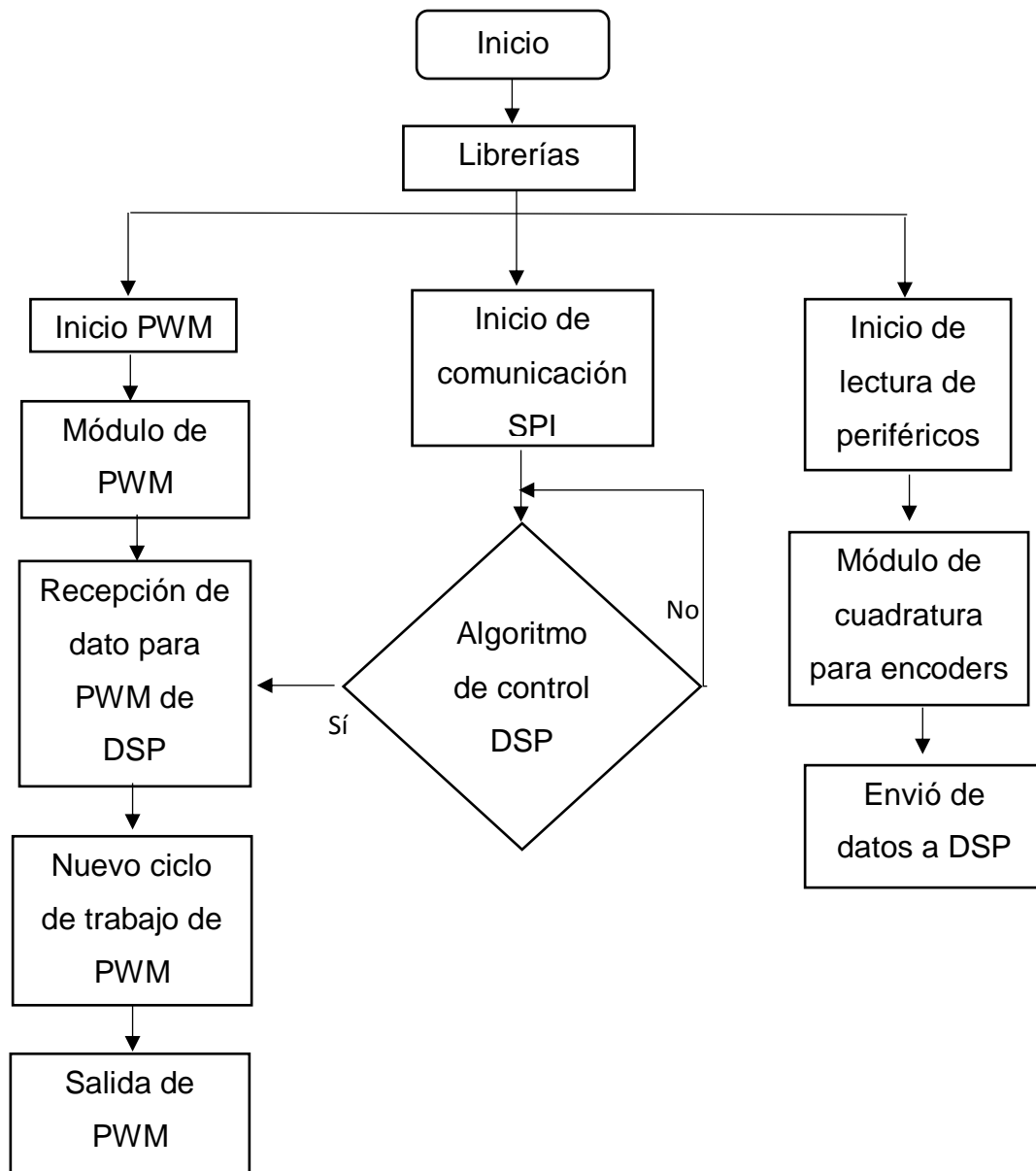


Figura 3-22. Diagrama de flujo programa FPGA

Los programas se implementaron en la plataforma desarrolla a través de los diferentes dispositivos de programación siendo que a través de los mismos cables de JTAG se logró hacer el depurado de estos programas.

4. RESULTADOS

El resultado de esta investigación es congruente con las necesidades dentro de la Universidad y está encaminado a la solución de los problemas de control de movimiento para robots, maquinaria CNC y de cualquier servo mecanismo en general.

El controlador de movimiento, resultado de este proyecto, puede ser utilizado como plataforma para otros trabajos de investigación relacionados con las áreas de control, sistemas digitales y servo mecanismos.

Al termino de este proyecto se cuenta con una plataforma digital que integra las tecnologías DSP y FPGA, junto con otros módulos de interfaz; que es capaz de controlar la mesa de coordenadas de 2 ejes mediante un controlador de movimiento y servodrivens. En las fig. 4-1 y 4-2 se muestra la tarjeta finalizada con todos sus componentes ensamblados vista de la cara inferior y superior respectivamente.

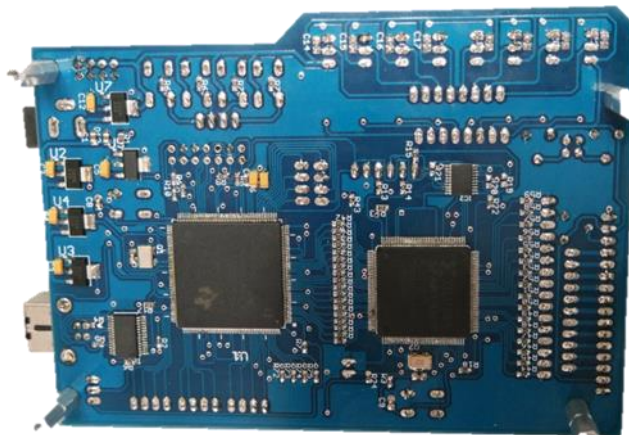


Figura 4-1. Plataforma electrónica final con componentes soldados vista inferior.

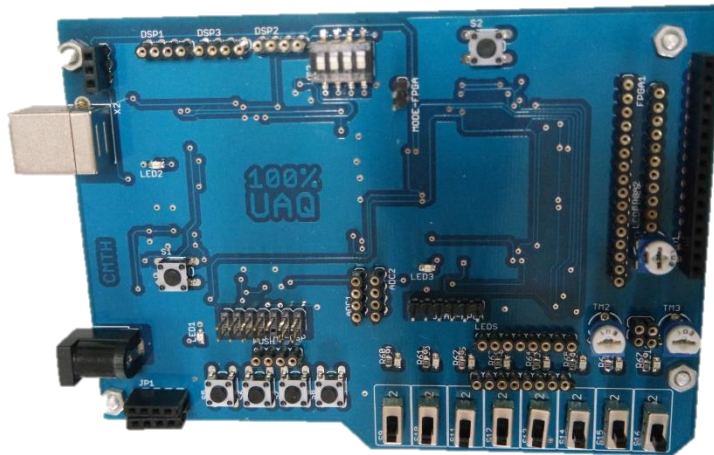


Figura 4-2. Plataforma electrónica final con componentes soldados vista superior.

Al manejar tarjetas electrónicas con componentes de montaje superficial se debe tener especial cuidado al momento de montar los elementos en la PCB, para este paso se utilizó un microscopio digital con sensor CMOS, con resolución de 640 x 480 conexión USB 2.0 y 5x de zoom digital como el que se muestra en la fig. 4-3



Figura 4-3. Microscopio digital resolución 640 x 480, 5x zoom digital.

Es recomendable revisar cada componente, sin embargo para fines prácticos muchos de los elementos son fáciles de soldar, otros por el contrario debemos tener especial cuidado como son los DSP y FPGA, dispositivos que se revisaron minuciosamente con la ayuda del microscopio, como se muestran en las figuras 4-4a, 4-4b, 4-4c, 4-4d, 4-4e las revisiones de cada pin soldado a la tarjeta se pueden ver claramente.

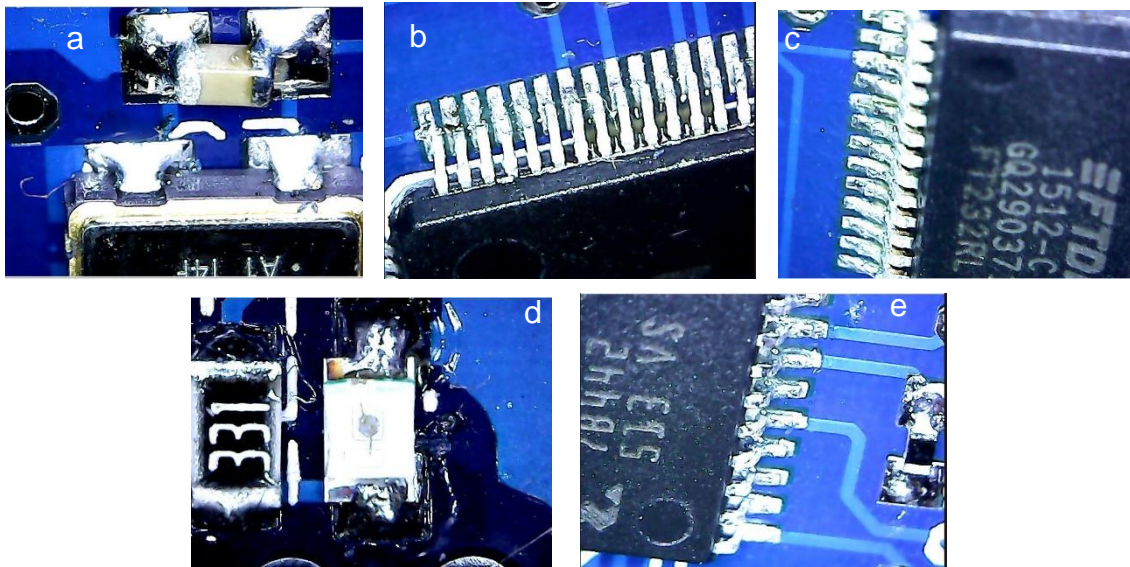


Figura 4-4. Revision de montaje de componentes con microscopio digital a) cristal y capacitor; b) pines de DSP; c) pines de FTDI; d) resistencia y led e) pines FPGA

En la fig.4-5 se puede ver la mesa de coordenadas que se utilizó como banco de pruebas de la tarjeta, vienen especificados los ejes “x” y “y”. La mesa de coordenadas cuenta con un gabinete en donde se encuentran los *servodrivers* para cada eje.

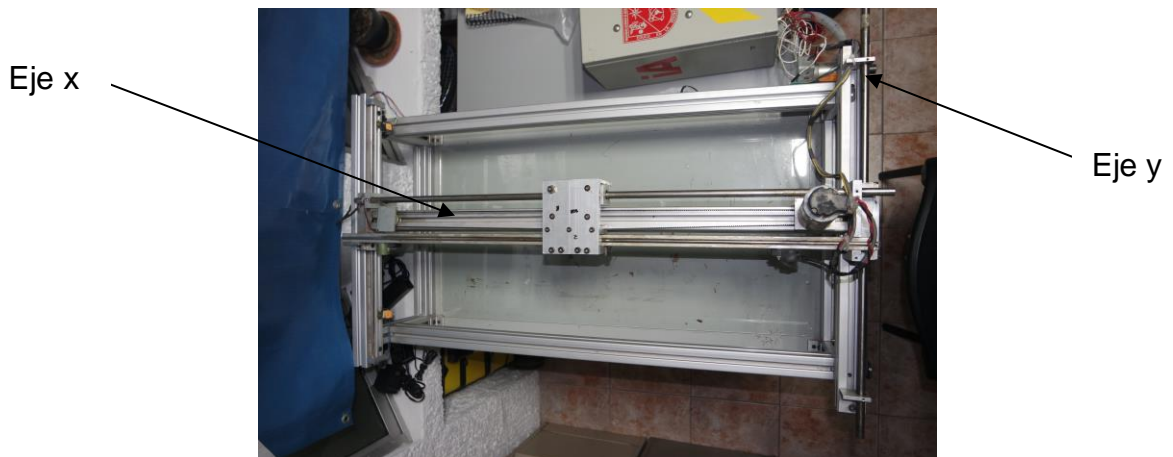


Figura 4-5. Mesa de coordenadas como banco de pruebas con identificación de eje “x” y “y”.

En la figuras 4-6 y 4-7 se muestran los motores de los ejes “x” y “y” respectivamente, en las imágenes también se puede apreciar que la unión para transmitir movimiento es por medio de bandas y engranes en cada eje.



Figura 4-6. Motor DC para eje x de mesa de coordenadas.

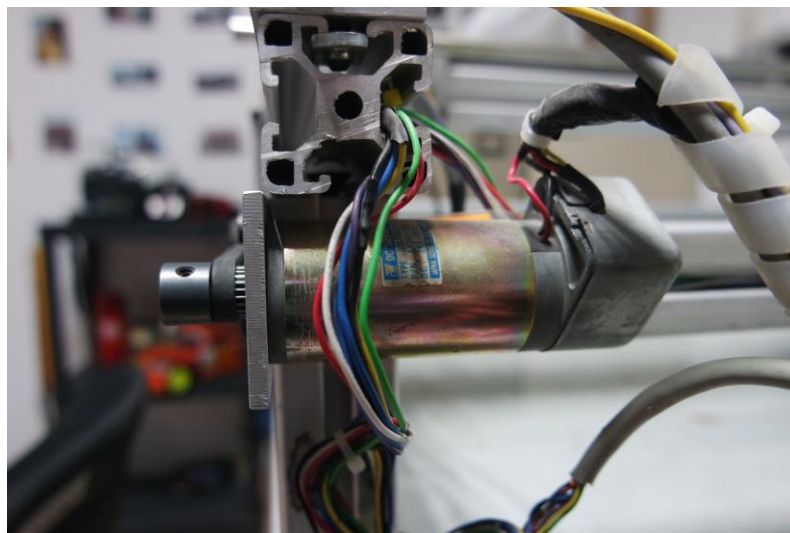


Figura 4-7. Motor DC para eje y de mesa de coordenadas.

Cada motor cuenta con su sensor de tipo encoder, los cuales están cableados hasta el gabinete principal y tienen al final un conector de tipo DB15 para facilitar su conexión a las diferentes plataformas. Donde los pines que se utilizan del conector son los que podemos ver en la tabla 3.

Tabla 3. Conexión de encoder a plataforma.

Pin Conector DB15	Función
Pin 2	Canal A encoder
Pin 3	Canal B encoder
Pin 5	Conexión a tierra
Pin 15	Alimentación a 5 volts

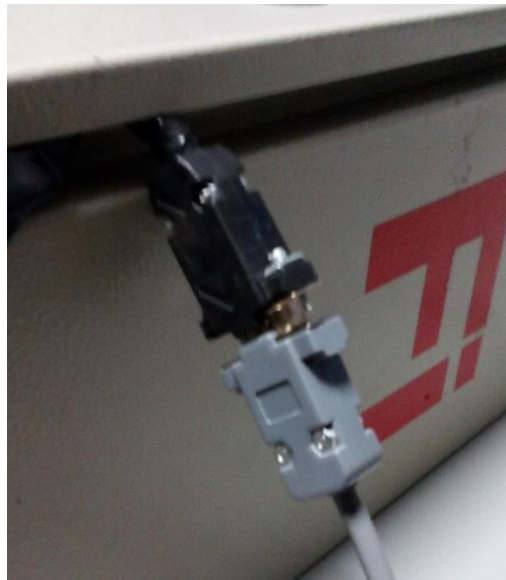


Figura 4-8. Conexión de encoder con DB15.

El gabinete principal cuenta con servodrivens modelo Maxon Motor ADS 50/10 50V/10 A 201583, en la fig.4-9 se pueden ver los servodrivens y su conexión a los motores de cada eje.



Figura 4-9. Servodrivers modelo Maxon Motor ADS para ejes "x" y "y".

Una vez realizadas las conexiones eléctricas y electrónicas del sistema fue posible empezar con las pruebas de los ejes mediante el uso de la tarjeta descrita en este proyecto. Donde con las conexiones de los encoders en los pines de uso general de la tarjeta se pudo entablar el control del sistema.

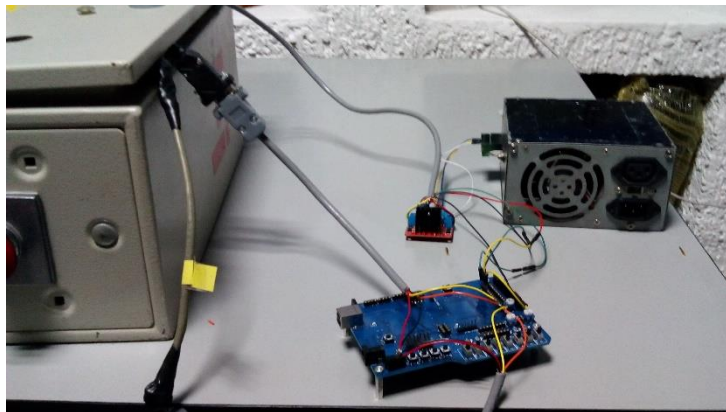


Figura 4-10. Conexión de sistema con la plataforma electrónica.

Se realizaron pruebas con los controladores, comparándolos con las simulaciones que se realizaron en el software de Matlab-Simulink. Se realizaron diferentes pruebas para verificar el funcionamiento de los controladores. En la fig.4-12 se puede ver un controlador de tipo PD ante un entrada de tipo escalón de 500 revoluciones por minuto y su comparación con la implementación de la tarjeta.

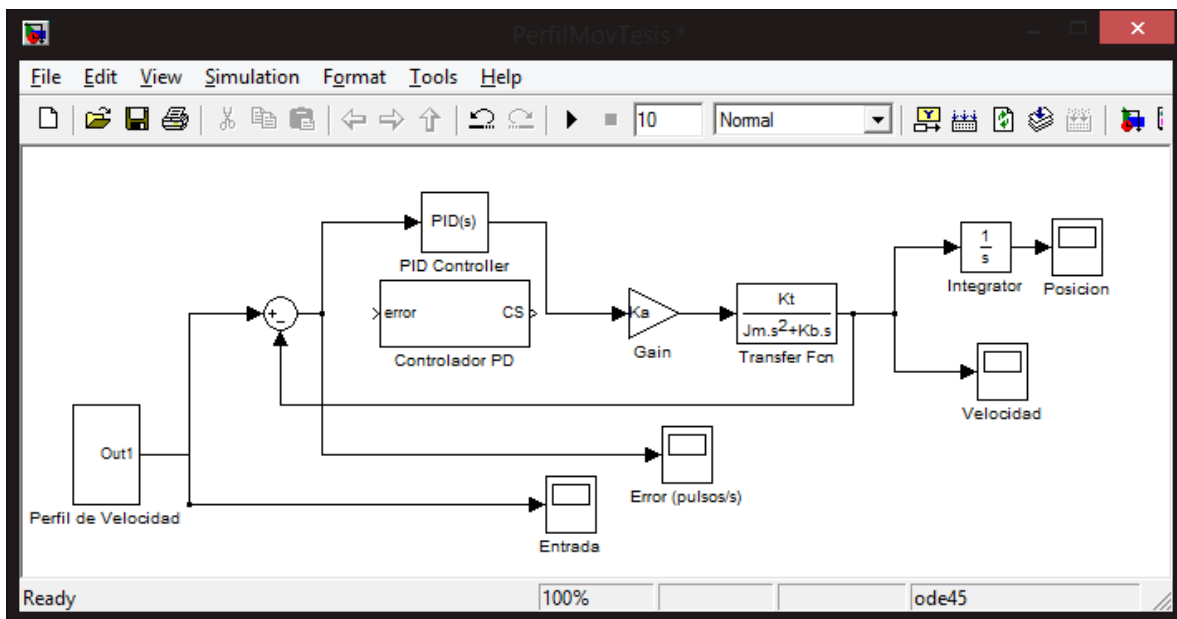


Figura 4-11. Simulación de sistema con bloque de control (P, PI, PD y PID).

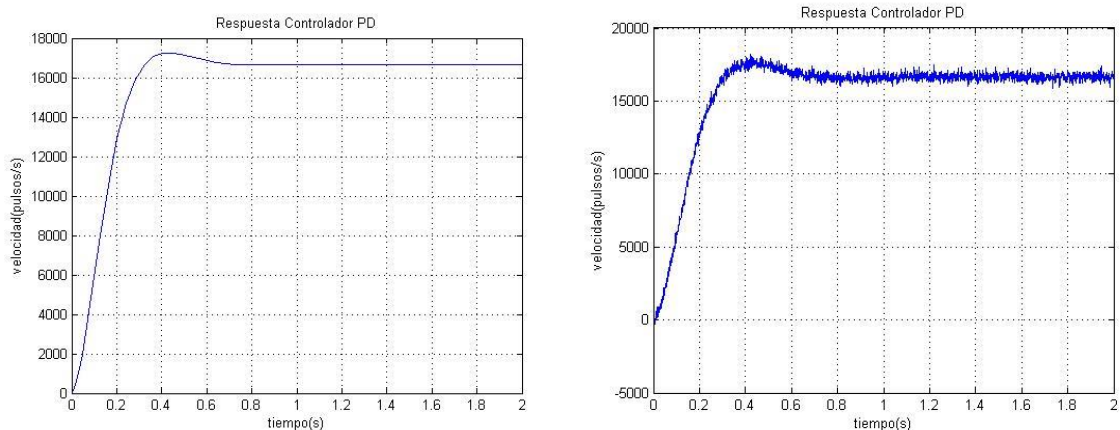


Figura 4-12. Respuesta del sistema con controlador PD teórico (izquierda) con plataforma (derecha).

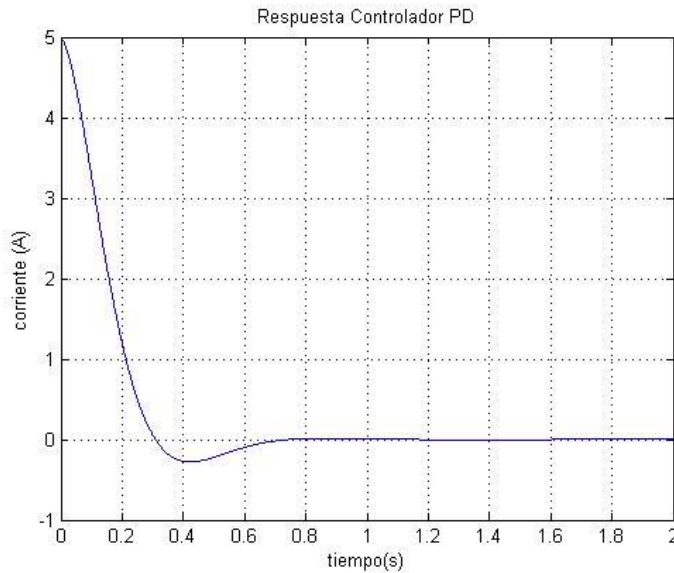


Figura 4-13. Respuesta de corriente del sistema con controlador PD.

Como se puede observar los resultados obtenidos de la metodología ofrece una representación de los modelos de control donde utilizando las herramientas de Matlab/Simulink podemos observar el comportamiento del sistema bajo diferentes condiciones. Existen diferentes tipologías para el control, las cuales se basan en el control de tipo PID. En la fig 4-14 se observa la respuesta teórica de un controlador de tipo PI ante una entrada de tipo escalón de 500 revoluciones por minuto con cambios de referencia a 1seg y a 3 seg.

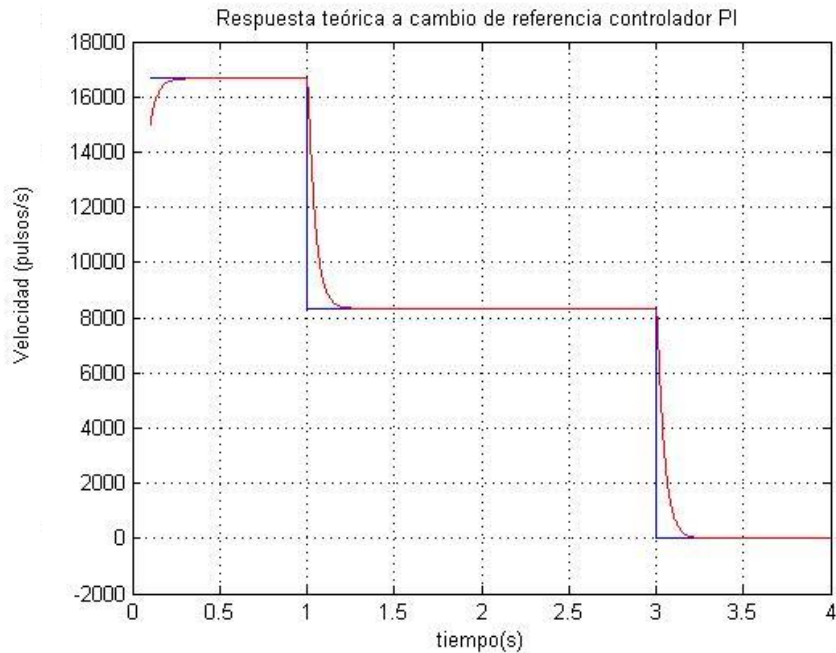


Figura 4-14. Respuesta teórica de un controlador de tipo PI ante un entrada de tipo escalón de 500 rpm.

En la fig. 4-15 se muestra los valores teóricos de las ganancias calculadas en el software Simulink.

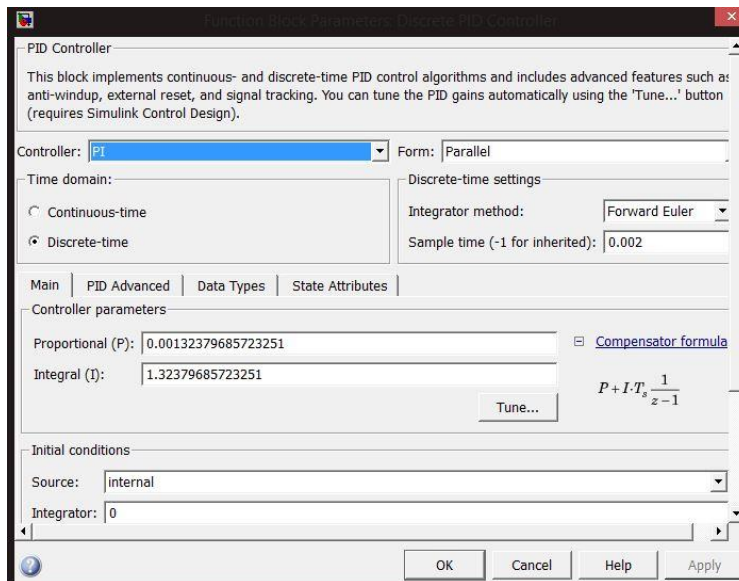


Figura 4-15. Valores teóricos de las ganancias para controlador PI.

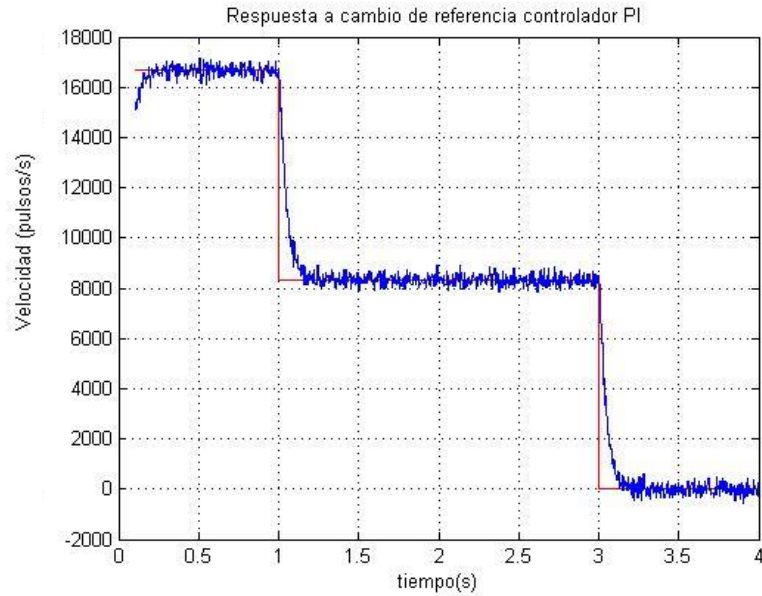


Figura 4-16. Respuesta del sistema con controlador PI ante una entrada escalón de 500rpm.

En la fig 4-17 se observa la respuesta teórica de un controlador de tipo PID ante una entrada de tipo escalón de 500 revoluciones por minuto con cambios de referencia a 1seg y a 3 seg.

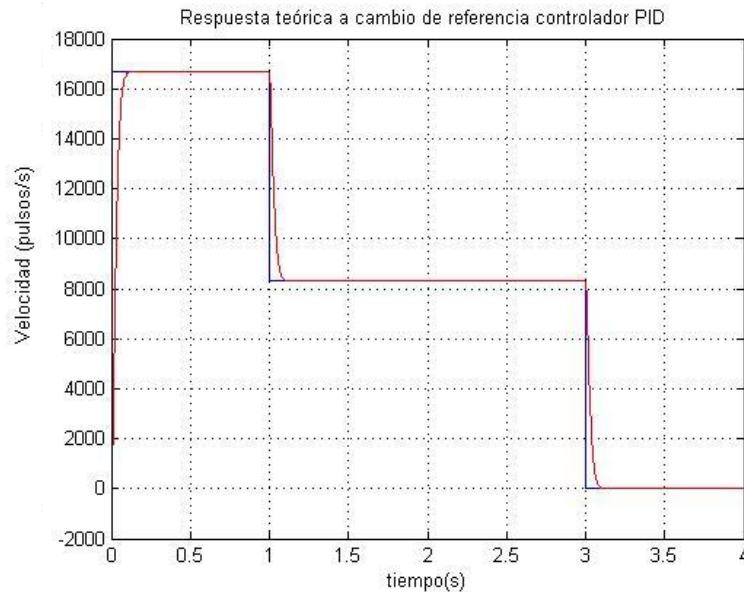


Figura 4-17. Respuesta teórica de un controlador de tipo PID ante un entrada de tipo escalón de 500 rpm.

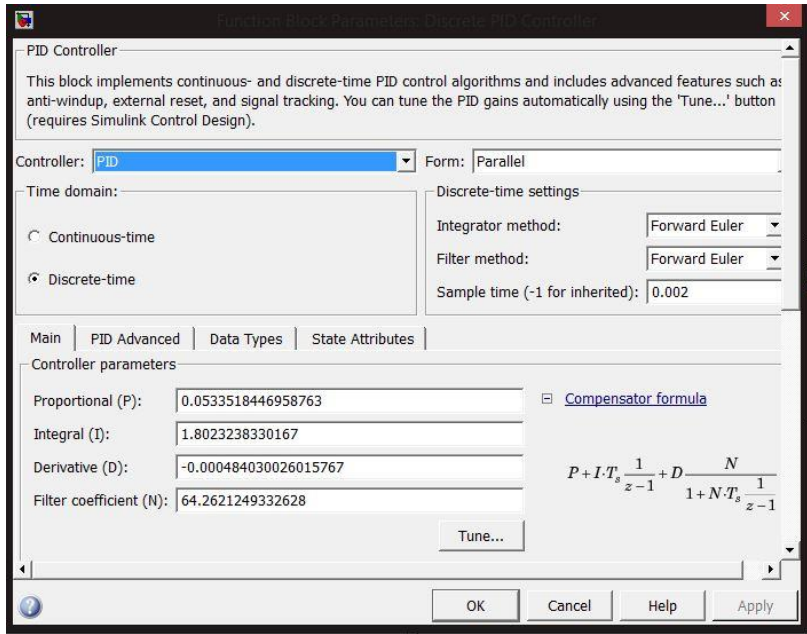


Figura 4-18. Valores teóricos de las ganancias para controlador PID.

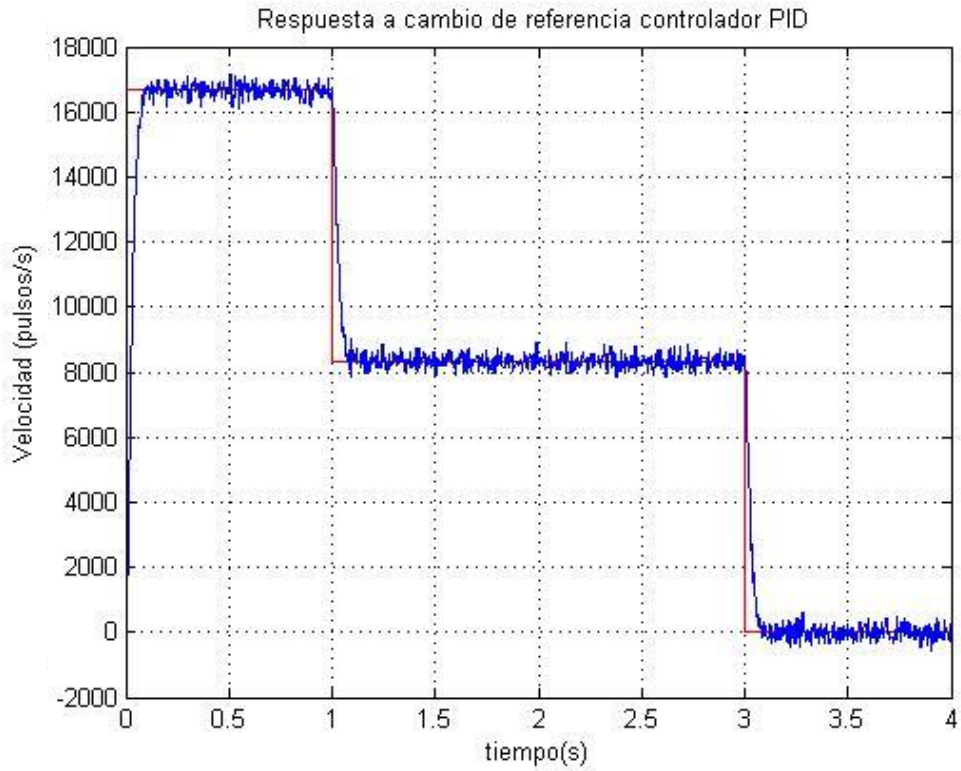


Figura 4-19. Respuesta del sistema con controlador PID ante una entrada escalón de 500rpm

Con la sintonización del controlador PD generado en Matlab se realizó un movimiento con un perfil trapezoidal con una velocidad de 13000 pulsos/s, una posición final de 40000 PPR, se muestran los resultados en la figura 4-20.

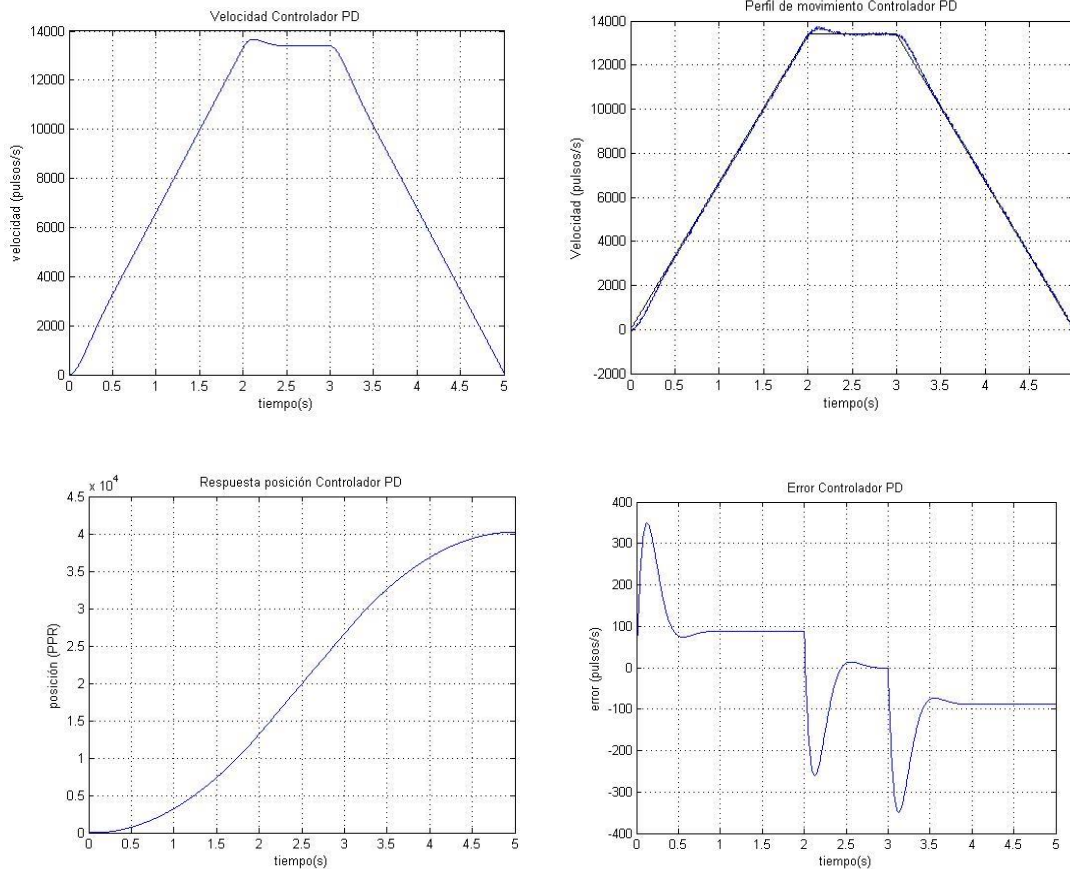


Figura 4-20. Respuesta del perfil trapezoidal con una velocidad de 13000 pulsos/s, y una posición final de 40000 ppr.

Con la sintonización del controlador PI generado en Matlab se realiza un movimiento con un perfil trapezoidal con una velocidad de 13000 pulsos/s, una posición final de 80000 PPR, se muestran los resultados obtenidos.

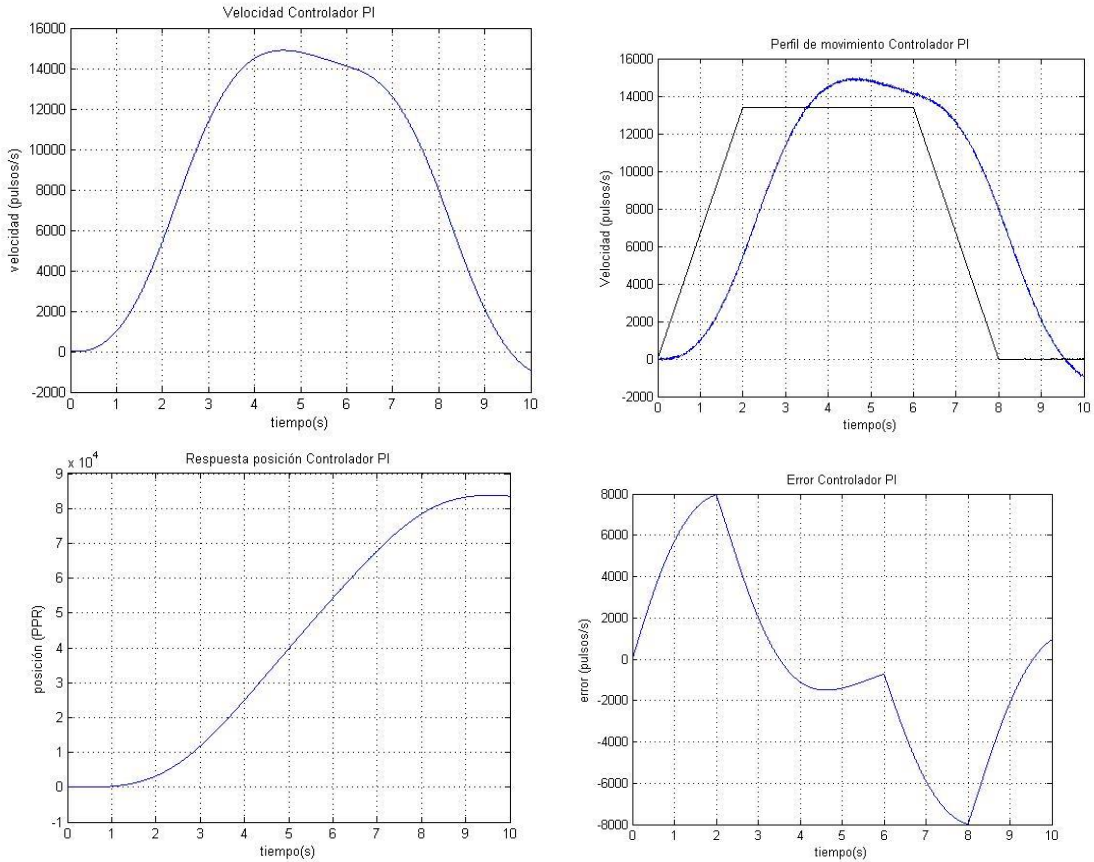


Figura 4-21. Respuesta del perfil trapezoidal con una velocidad de 13000 pulsos/s, y una posición final de 80000 ppr.

Con la sintonización del controlador PID generado en Matlab se realiza un movimiento con un perfil trapezoidal con una velocidad de 13000 pulsos/s, una posición final de 80000 PPR, se muestran los resultados obtenidos.

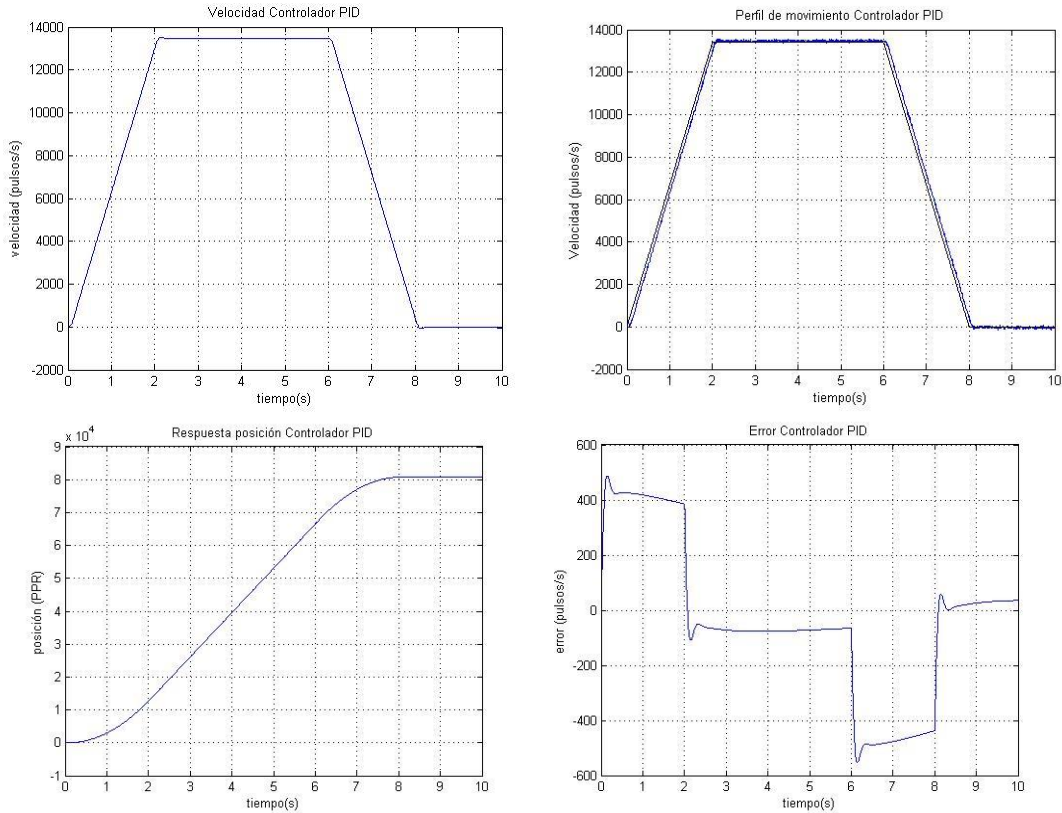


Figura 4-22. Respuesta del perfil trapezoidal con una velocidad de 13000 pulsos/s, y una posición final de 40000 ppr.

5. CONCLUSIONES

Al término de este proyecto se cuenta con una plataforma electrónica en la que se pueden implementar diferentes algoritmos, tanto en la parte de FPGA como en el DSP, esta plataforma permite la comunicación entre componentes electrónicos y los servomotores, la plataforma permite ser reconfigurable por su sistema integrado con el que cuenta, siendo una plataforma flexible para la implementación de software por parte del usuario. Básicamente el algoritmo de control es resuelto por el DSP mientras que el sistema de interfaces es a través del FPGA. La PC se utilizó para la depuración de los algoritmos.

La plataforma cuenta con diferentes módulos con los cuales, no solo se puede utilizar para fines de control, también puede ser utilizada para la enseñanza de

diferentes materias como electrónica digital, control digital, programación, sistemas embebidos, entre otras, al ser un plataforma didáctica queda como antecedente de desarrollo para futuras generaciones. La comunicación entre los dos dispositivos fue un factor importante para el desarrollo de la plataforma, ya que cuenta con canal de datos para intercambiar información del control de movimiento.

La metodología propuesta para el desarrollo del proyecto se llevó de manera satisfactoria pudiendo llevar a cabo la implementación del control de movimiento con una velocidad de procesamiento de 1ms en la plataforma electrónica. Dentro de la metodología una parte importante fue la retroalimentación que se tenía de los diseños tanto de software como de hardware, ya que estos daban paso a mejoras importantes que en primeras versiones no se habían considerado.

En un futuro la plataforma podrá ser utilizada para implementar diferentes algoritmos de control más sofisticados como lo son: redes neuronales, lógica difusa, control inteligente, entre otros. El éxito de un diseño de FPGA-DSP depende en gran medida de la experiencia del diseñador, no sólo en la implementación de los diseños en FPGA, sino también en la adaptación de algoritmos.

6. REFERENCIAS

Astarloa, A., Lázaro, J., Bidarte, U., Jiménez, J. y Zuloaga, A. 2009. "FPGA technology for multi-axis control systems". *Mechatronics*. Vol 19. Pp. 258-268.

Baturone, A. O. (2006). *Robótica: manipulados y robots móviles*. Marcombo.

Bueno, E.J., Hernández, Á., Rodríguez, F. J., Girón, C., Mateos, R. y Cóbreces, S. 2009. "A DSP- And FPGA-Based Industrial Control with High-Speed Communication Interfaces for Grid Converters Applied to Distributed Power Generation Systems", *IEEE Transactions on Industrial Electronics*, vol. 56, No.3.

Castillo, Mauricio (2014), *Guía para la formulación de proyectos de investigación 1ª ed.*, México, Cooperativa Editorial Magisterio, pp 14-16.

Chan, Y. F., Moallem, M. y Wang, W. 2007. "Design and implementation of modular FPGA Based PID controllers". *IEEE Transactions on Industrial Electronics*, vol. 54. No.4.

Delta Tau Data Systems. 2011. *PMAC2 PCI hardware reference manual*.

Galil Motion Controls Inc. 2011. *DMC-18x6 Manual Rev 1.0.e*.

Galindo J. C. 2001, *Diseño y construcción de una tarjeta de control de arquitectura abierta*. Universidad Autónoma de Querétaro.

Goetz J (2004) Implementation of complete AC servo control in a low cost FPGA and subsequent ASSP conversion. *IEEE Int Conf App Pow Electron* 1:565–570.

Gonzalez Río, M.J. (1997). *Metodología de la investigación, Técnicas de recolección de datos: Aguacera*.

González, J., Baeyens, E., Gayubo, F. Pérez J., Fraile, J. y García, F. 2004. "Desarrollo de un controlador abierto para un robot industrial tipo SCARA". *Revista Iberoamericana de Automática e Informática Industrial*. Vol 1. No.1.

Grout IA, Dorey AP (2005) Design and testing of a PI controller ASIC. *Microprocess Microsyst* 19:15–22.

Gu, G., Zhu, L., Xiong, Z. y Ding, H. 2010. “Design of a Distributed Multiaxis Motion Control System Using the IEEE-1394 Bus”. *IEEE Transactions on Industrial Electronics*. Vol 57. No. 12.

Hongyu, Shi, Feng Yong, and Chen Na (2010). “A Distributed Digital Motion Control 5 System

Lim SY, Dawson DM, Vedagarbha P (1996) Advanced motion control of mechatronic systems via a higher-speed DSP. *Mechatronics* 6:101–122.

Liu, Yuan, Yong-zhang Wang, and Hong-ya Fu. (2008). “An Open Architecture Motion Controller for CNC Machine Tools” 2008 2nd International Symposium on Systems and Control in Aerospace and Astronautics: 1-4.

Loenzo R.A. 2009, Sistema de control numérico computarizado de arquitectura abierta, modular y portátil. Universidad Autónoma de Querétaro.

Masten MK, Panahi I (2000) Digital signal processors for modern control systems. *Control Eng Pract* 5(4):449–458.

Mendoza, F. 2012. “Controlador Estándar de Movimiento Multieje con base en FPGA”. Tesis de Maestría. Universidad Autónoma de Querétaro.

Oliveira, Schneider De, and Raul Guenther. (2008). “Embedded open architecture robotic controller”: 308-315.

Osornio, R. 2004. “Diseño y Construcción de una tarjeta controladora de 3 ejes”. Tesis de Maestría. Universidad Autónoma de Querétaro.

Osornio, R. 2007. “Diseño de sistema de control para CNC de alta velocidad”. Tesis de Doctorado. Universidad Autónoma de Querétaro.

Osornio-Rios, Rene RT, Gilberto HR, Rodrigo CM (2008) The application of reconfigurable logic to high speed CNC milling machines controllers. Control Eng Pract 16:674–684.

Ronquillo, G. 2002. “Controlador Digital para Servomotores sin Escobillas de Corriente Directa”. Tesis de Maestría. Universidad Autónoma de Querétaro.

Santos, S. O. 2012. Control de movimiento para un robot (ROMAT 56). Querétaro, Qro.

Shetty, D. y Kolk, R. A., 2010, Mechatronics System Design, CENGAGE Learning, 2nd Edition, pp. 162-164.

SILGE Electrónica. (2011). Encoder Incremental, Descripción General. Buenos Aires: Eltra.

Sobh MS, Mohamed D, Thomas CH, Anil S, Raul M (2011) Concurrent design of a three-link manipulator prototype. Comput Electr Eng 27:445–458.

Texas Instruments. (1995). Code Composer Studio (CCS) Integrated Development Environment (IDE). February 2016, de Texas Instruments Sitio web: <http://www.ti.com/tool/ccstudio>

Texas Instruments. (2014). XDS100. February 2016, de Texas Instruments Wiki Sitio web: <http://processors.wiki.ti.com/index.php/XDS100>

Texas Instruments. (June 2007–Revised August 2015). TMS320F28335 Data Manual. February 2016, de Texas Instruments Sitio web: <http://www.ti.com.cn/cn/lit/ds/symlink/tms320f28335.pdf>

W. Bolton. (2001). Ingeniería de Control. México: Alfaomega.

Xilinx. (2013). ISE Design Suite. February 2016, de Xilinx all programmable Sitio web: <http://www.xilinx.com/products/design-tools/ise-design-suite.html>

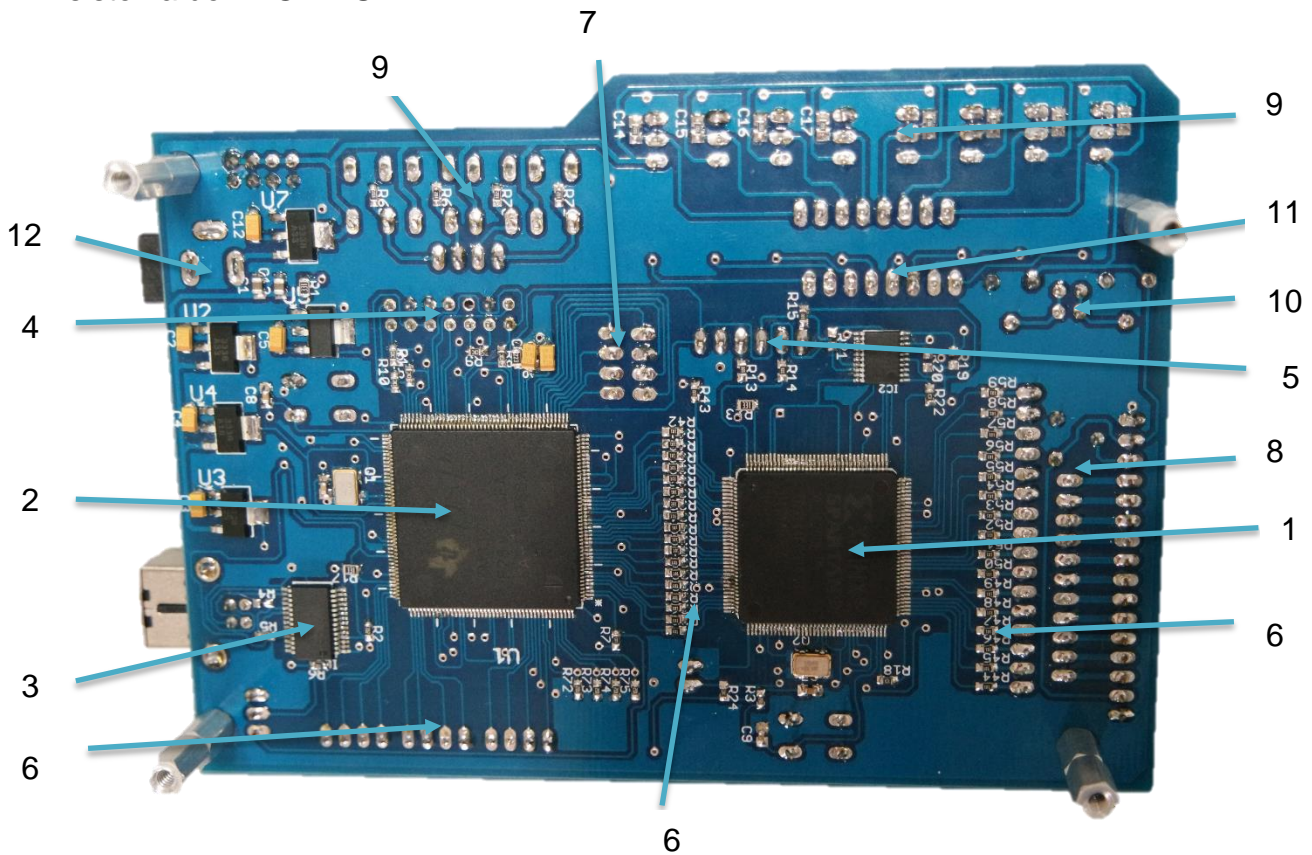
Xilinx. (July 19, 2013). Spartan-3E FPGA Family Data Sheet. February 2016, de Xilinx Sitio web: http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf

Xing, Hu, Huan Jia, and Liu Yanqianga. (2011). "Motion Control System Using SERCOS over EtherCAT" *Procedia Engineering* 24: 749-53.

Zbigniew Bielewicz, Leszek Debowski and Eugeniusz Lowiec. (2000). A DSP and FPGA based integrated controller development solutions for high performance electric drives.. *IEEE*, ., 679-684.

7. APÉNDICE

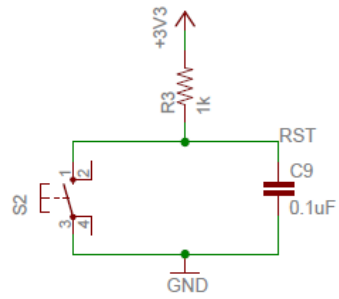
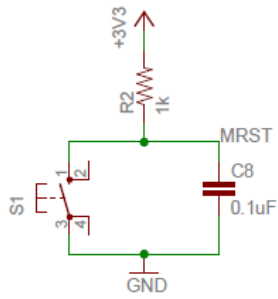
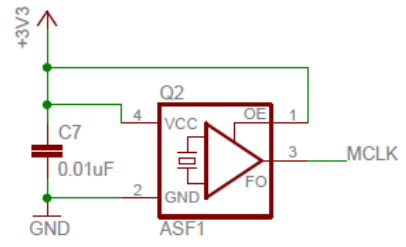
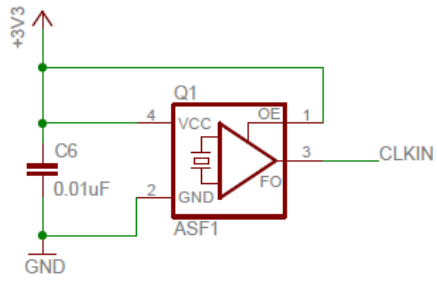
A continuación se presentan los módulos de la plataforma que acompañan a sistema de FPGA-DSP:

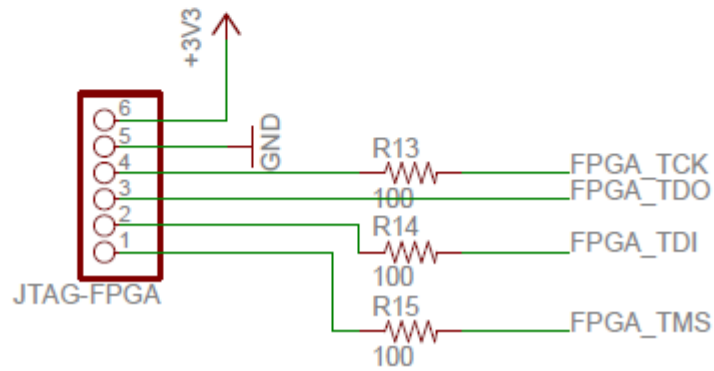
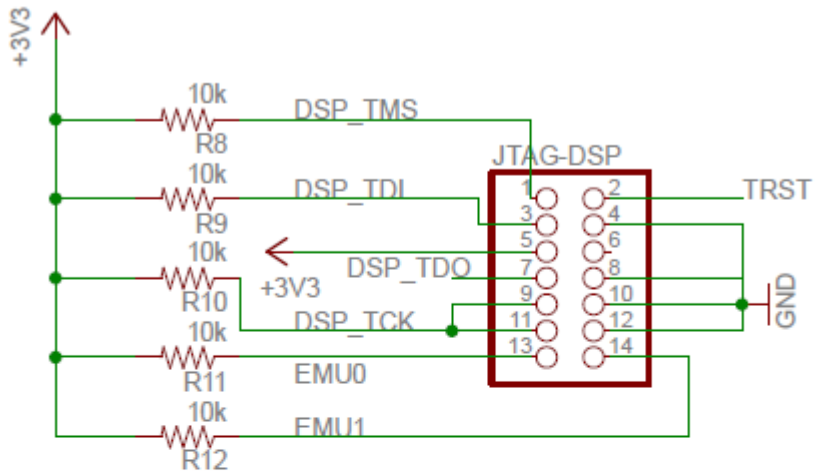


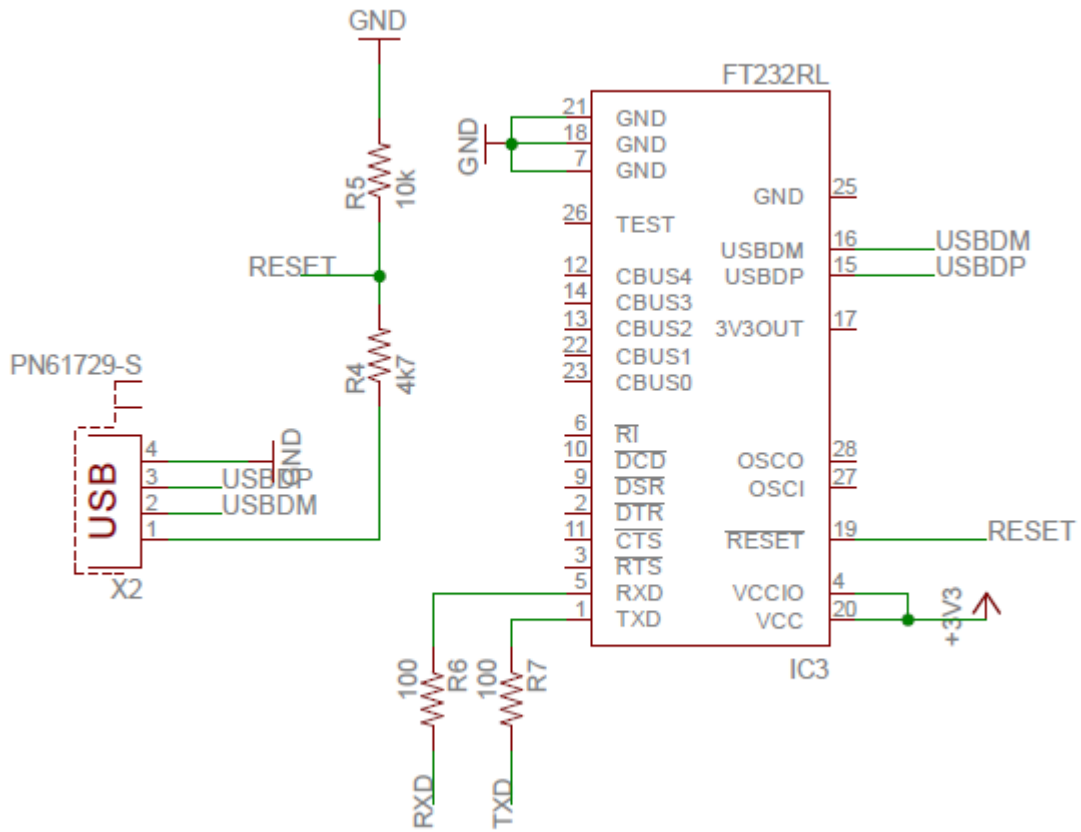
1. FPGA de Xilinx modelo Spartan XC3S250E.
2. DSP de Texas Instruments modelo TMS320F28335.
3. Comunicación Serial con FT232R de FTDI Chip, interfaz con conector USB tipo B.
4. JTAG-DSP, la programación y depuración del software se realiza a través de esta conexión por medio del emulador XDS100V2
5. JTAG-FPGA la programación y depuración del software se realiza a través de esta conexión por medio del cable JTAG HS2
6. I/O Propósito General la plataforma cuenta con 16 salidas digitales del PFGA y 12 salidas digitales y 6 entradas analógicas de parte del DSP, también cuentan con un bus de comunicación entre ambos dispositivos de 19 pines, para un total de 53 pines para utilizar.

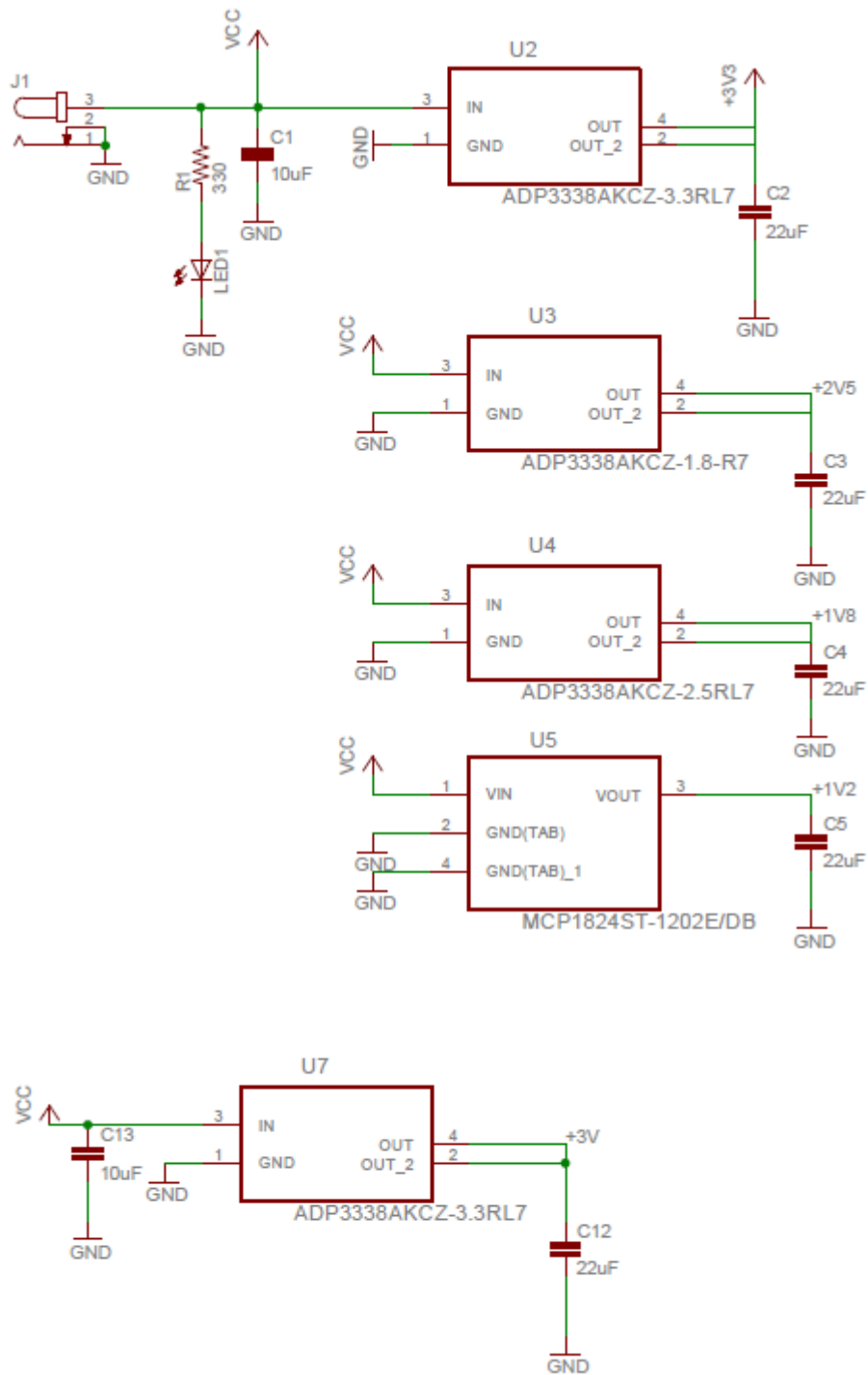
7. ADC, el DSP cuenta con seis entradas de tipo analógicas con una resolución de 12 bits, una entrada de voltaje entre 0 V y 3 V y una tasa de conversión de 80ns con un reloj de 25Mhz.
8. LCD, la plataforma cuenta con los pines dedicados para utilizar una pantalla de cristal líquido de tipo 16x2.
9. Entradas Digitales, dentro de la plataforma se tienen módulos digitales como son: 8 interruptores, 4 botones.
10. Entradas Analógicas, se cuenta con entradas de tipo potenciómetro los cuales entregan un valor de voltaje entre 0V y 3.3V.
11. Salidas Digitales, además del módulo de LCD con que cuenta la plataforma, se tienen 8 leds, que pueden sr utilizados por cualquiera de los dispositivos o módulos.
12. Alimentación de la plataforma, la tarjeta cuenta con un conector hembra de 2.1mm para su alimentación de voltaje, la cual se puede introducir voltaje entre 2.7V y 8V para su correcto funcionamiento. Para mayor referencia se anexan los diagramas esquemáticos de los reguladores de voltaje de la plataforma.

7.1 Diagramas Esquemáticos

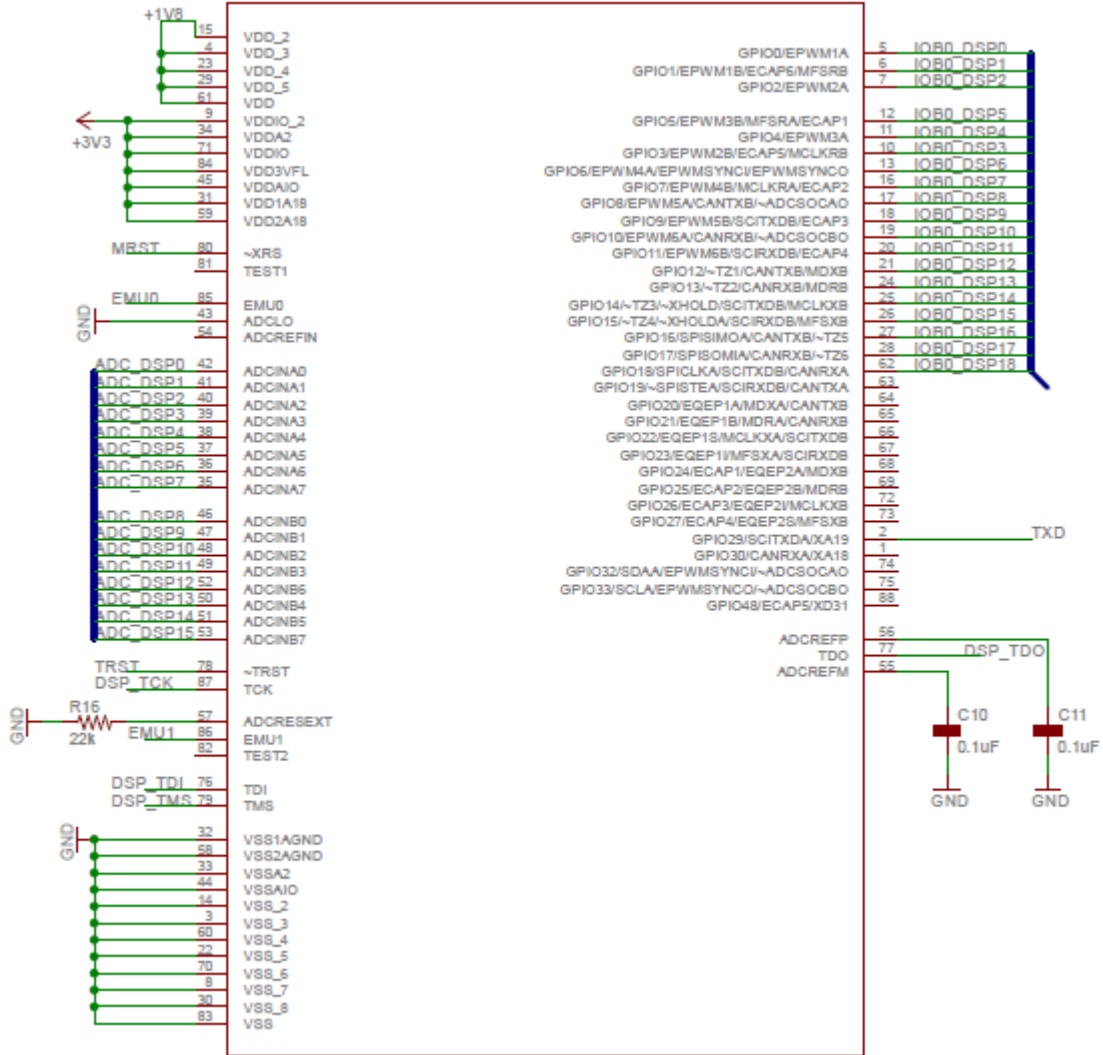






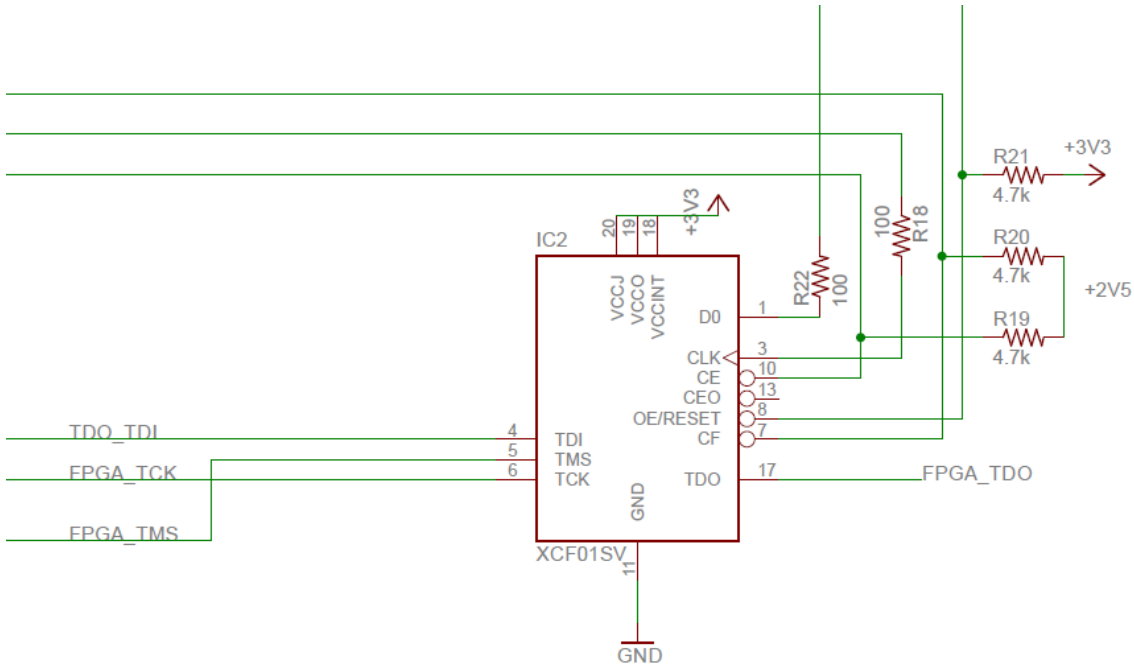
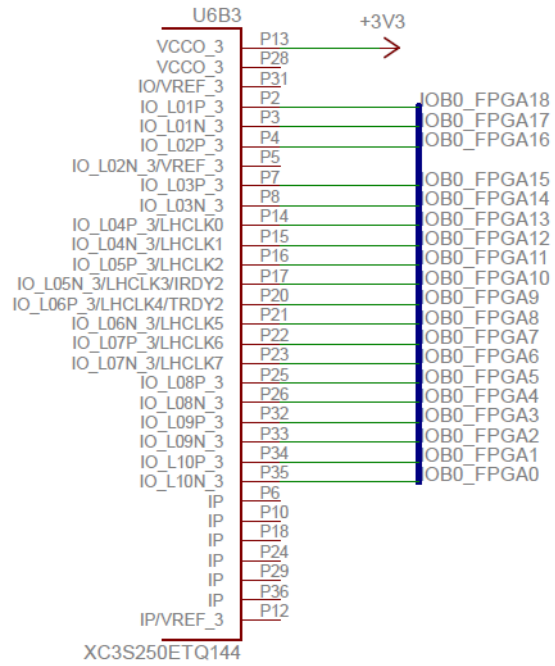
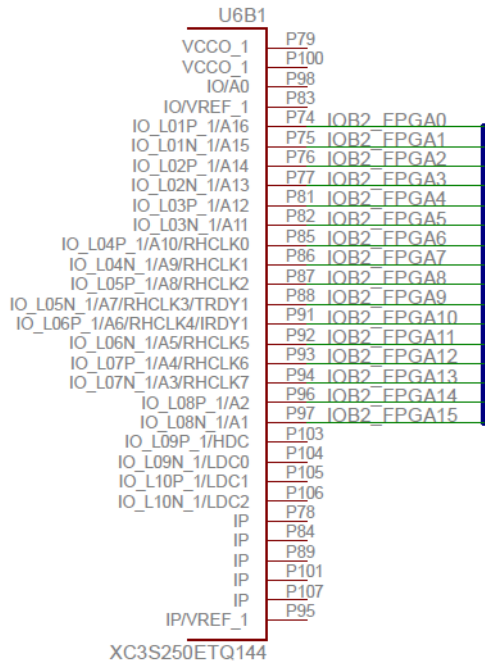


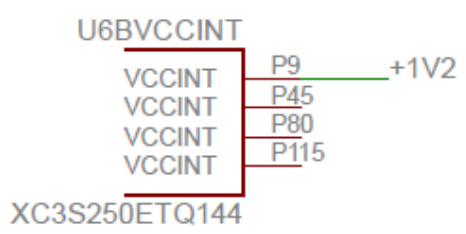
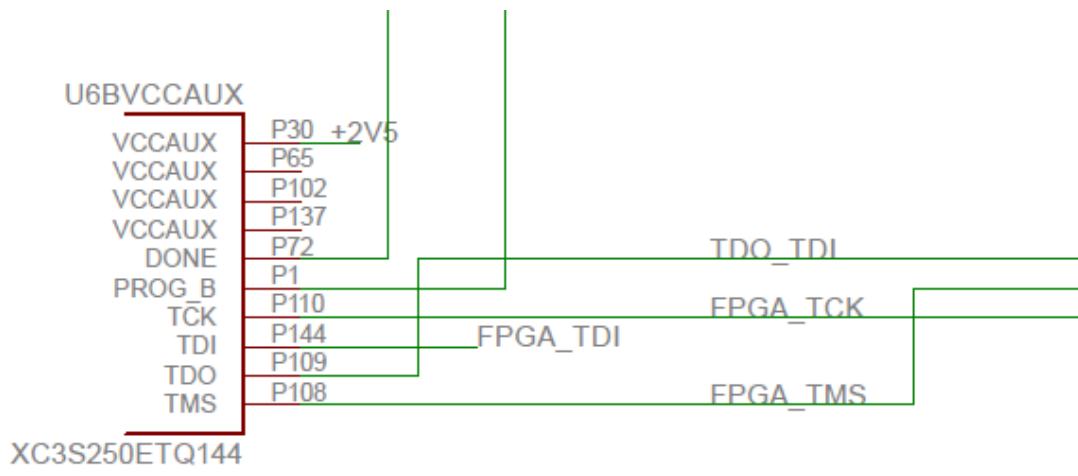
U1A

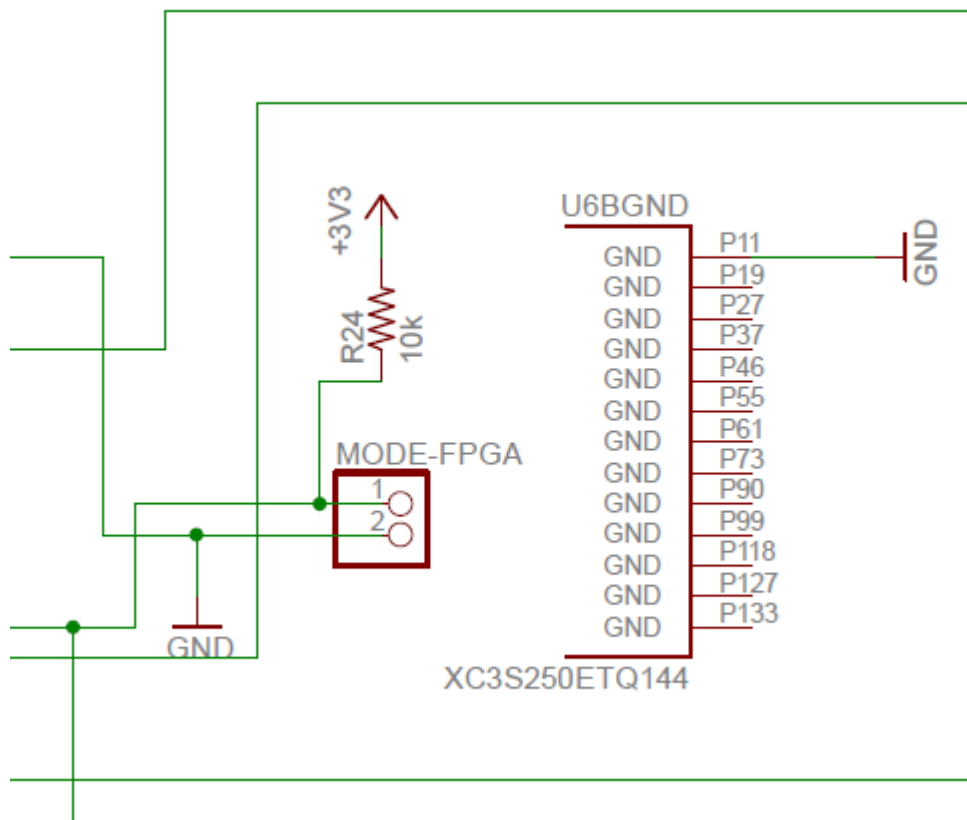


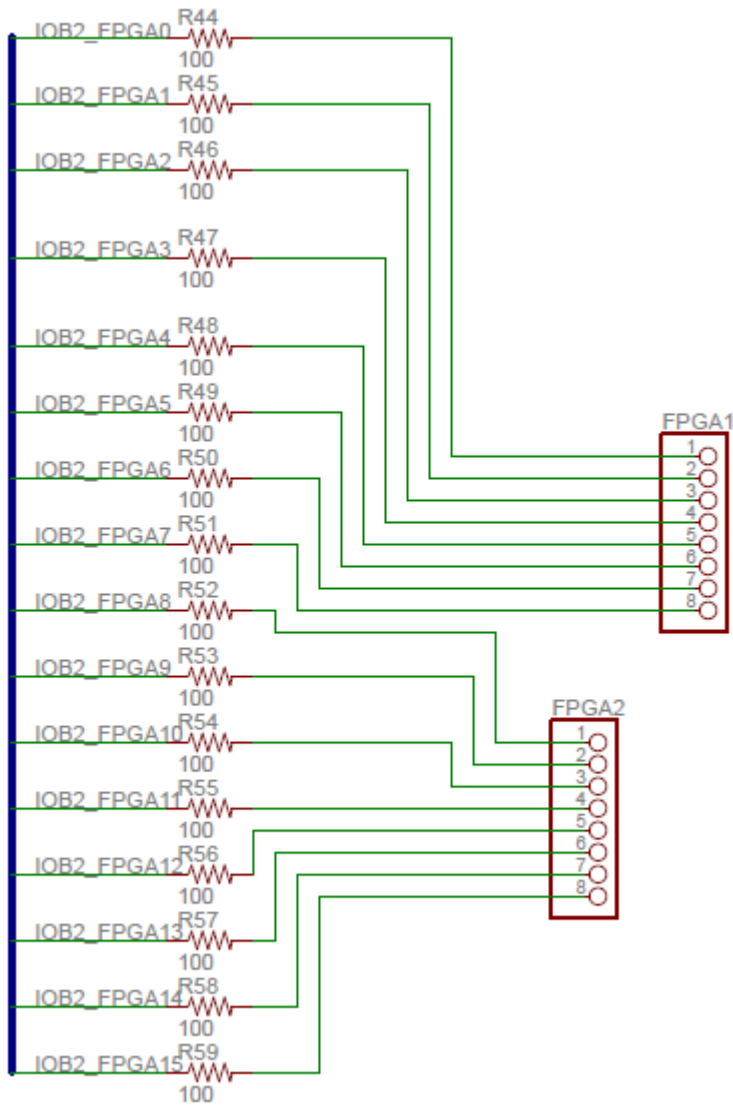
TMS320F28335PGFA

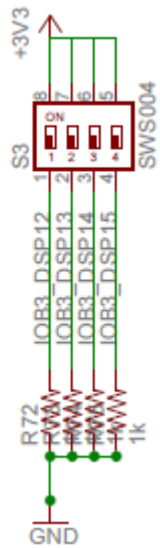
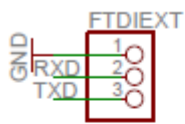
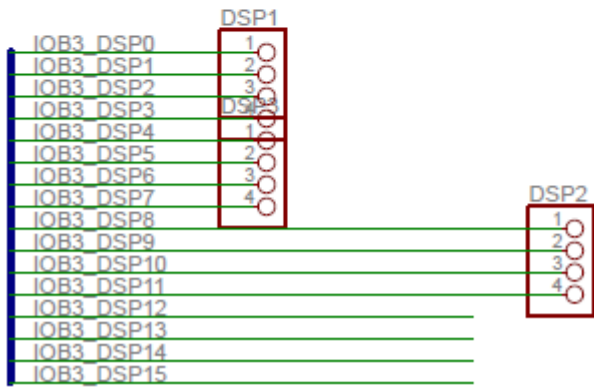
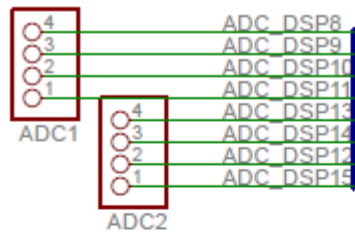


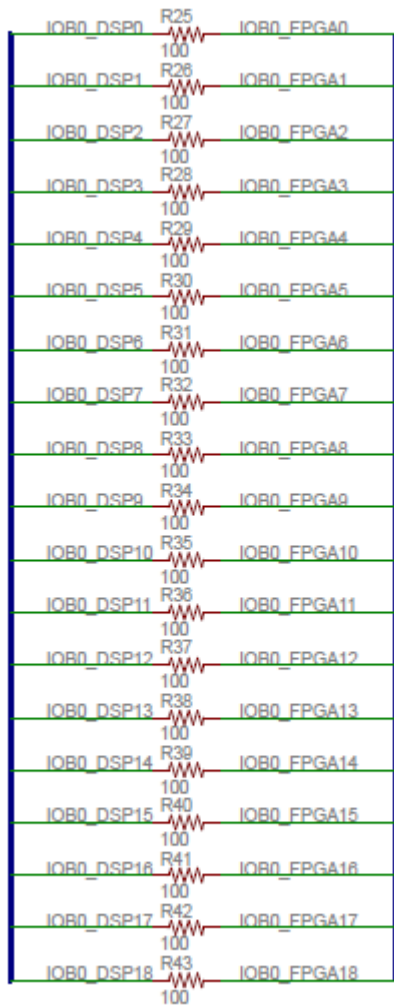




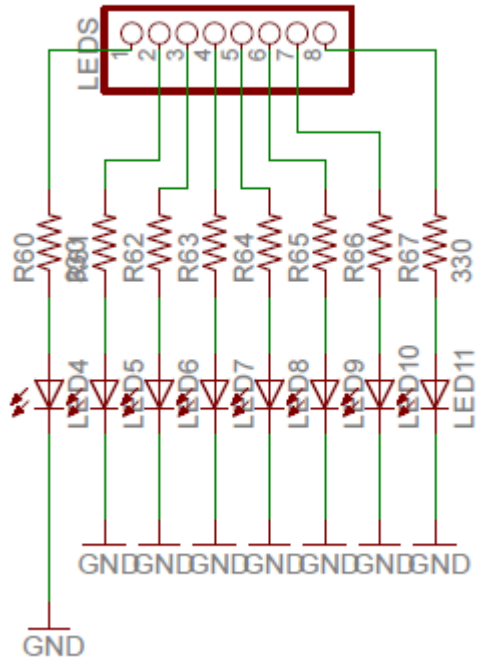




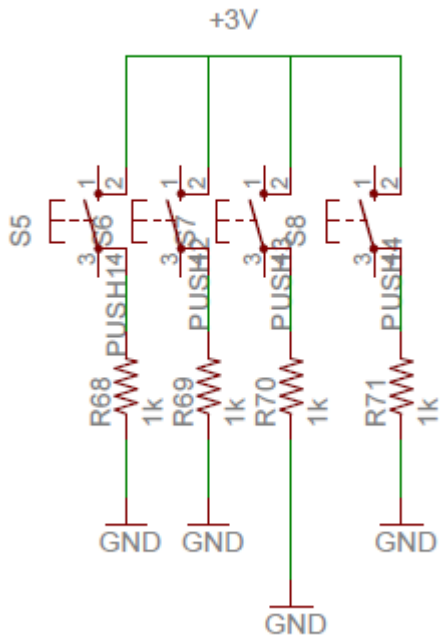


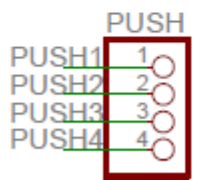
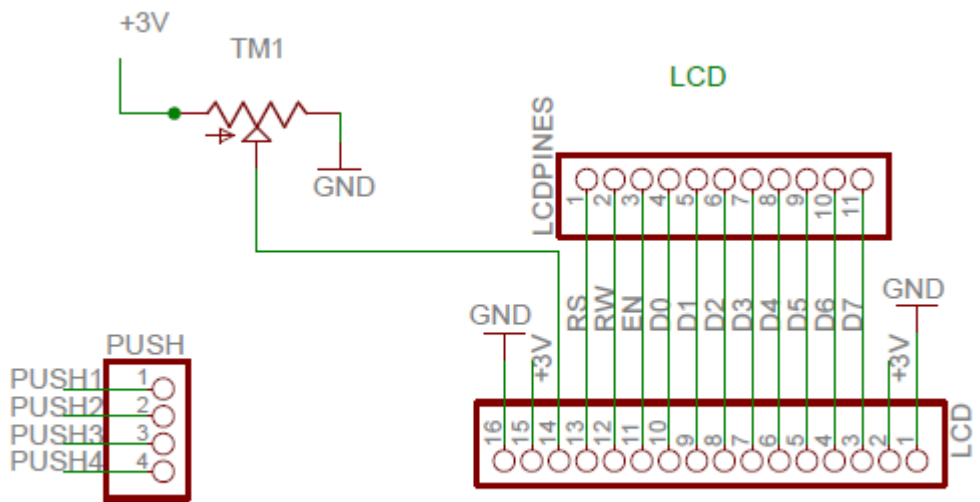


LEDS

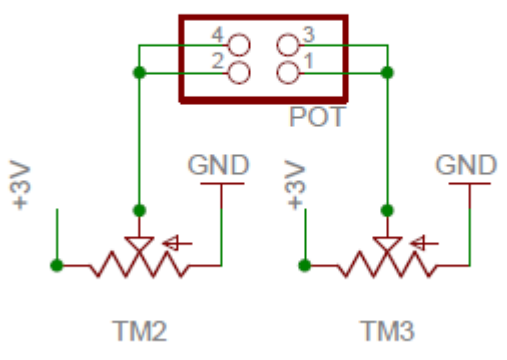


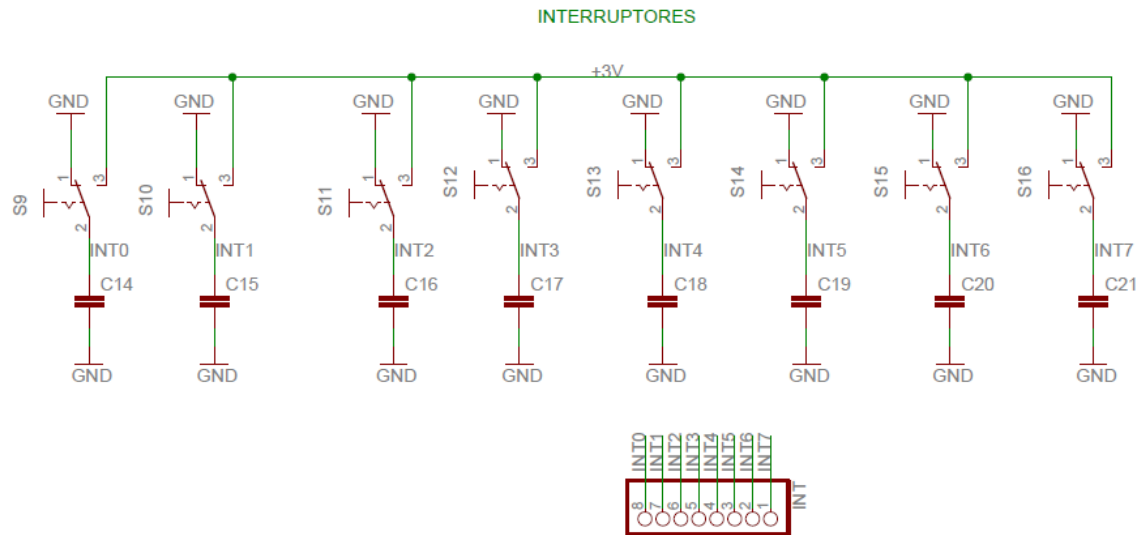
PUSH BOTON





POTENCIOMETROS





7.2 Scripts Matlab

Diseño de Controlador

```
clc
```

```
clear all
```

```
close all
```

```
%%
```

```
%Datos Generales
```

```
Kt=0.08; %N*m/A Constante de par
```

```
Jm=0.000025; %kg*m^2 Inercia del motor
```

```
Kb=0.00074; %(Propuesto) Constante de friccion viscosa
```

```
Ke=7.5; %Constante de fuerza contraelectromotriz
```

```
Ka=1; %constante de servo amplificador
```

```
Zeta=0.7; %Factor de amortiguamiento
```

```
Wn=12; %Frecuencia natural
```

```
%%
```

```
%Modelado del motor
```

```

L=0.0026;
R=1.2;

num=Kt;
Denf1=Jm*L;
Denf2=(Jm*R+Kb*L);
Denf3=(Kb*R+Kt*Ke);
Den=[Denf1 Denf2 Denf3];
Funcion_de_transferencia=tf(num,Den)

%%
%Paso 1
%%Primer Polo P1
P1=0
P2=-Kb/Jm
%%Polos en lazo cerrado
Polol_Re=-Zeta*Wn;
PololR_Im=Wn*sqrt(1-(Zeta^2));
Polol_Im=Wn*sqrt(1-(Zeta^2))*i;
Polol=Polol_Re+Polol_Im
Polol2=Polol_Re-Polol_Im

%%
%Paso 2
%%Angulo phi_1
Num=Wn*sqrt(1-(Zeta^2));
Phi_1= 180+atand((Num)/(-Zeta*Wn))

%Angulo de phi_2
Den=-Zeta*Wn-P2;

```

```

Phi_2=atand( (Num) / (Den) )

%%Angulo Phi_3
Phi_3=180-Phi_1-Phi_2
%%
%Paso 3
%%Calculo de Z
a=Num/tand(Phi_3);
Z1=-(a+Zeta*Wn);
S=Z1
%%
%Paso 4
%%Calculo de Ti
format long
Td=-1/S
format short
%%
%Paso 5
%%Calculo de Norma de controlador "Kp(Norma)"
Num1=sqrt( ((-Td*Zeta*Wn+1)^2)+((Td*Wn*sqrt(1-(Zeta^2)))^2));
Den1=1;
Norma1=Num1/Den1

%%Norma de la planta
Num2=Kt;
Den2=Wn*sqrt( (Jm^2) * (Zeta^2) * (Wn^2) -
2*Jm*Kb*Zeta*Wn+ (Kb^2) + (Jm^2) * (Wn^2) * (1- (Zeta^2)) );
Norma2=Num2/Den2

Den22=Wn*sqrt( (Jm^2) * (Wn^2) -2*Jm*Kb*Zeta*Wn+ (Kb^2) );
Norma22=Num2/Den22

```

```

%%
%Paso 6
%%calculo de KP
Kp=1/((Norma22)*(Normal))

```

Para tener un reporte de los resultados arrojados por MATLAB se realizó el siguiente código para que MATLAB hiciera un archivo de texto con los resultados:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Reporte de Controlador
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Open file
fid = fopen('Reporte_carlos.txt','wt');

% Description header
fprintf(fid, '-----
-\n');
fprintf(fid, '-- Reporte de Diseño de Controlador de Posición-
-\n');
fprintf(fid, '-- FI - UAQ                                     -
-\n');
fprintf(fid, '-- Servo Systems                               -
-\n');
fprintf(fid, '-- Carlos Miguel Torres Hdz                   -
-\n');
fprintf(fid, '-----
-\n');
fprintf(fid, '\n');

% Datos del Motor
fprintf(fid, '*Datos del Motor:\n');

```

```

fprintf(fid, '      Resistencia Eléctrica R: %f\n',R);
fprintf(fid, '      Inductancia L: %f\n',L);
fprintf(fid, '      Inercia del motor Jm: %f\n',Jm);
fprintf(fid, '      Constante de par Kt: %f\n',Kt);
fprintf(fid, '      Constante de fricción viscosa Kb:
%f\n',Kb);
fprintf(fid, '      Constante de fuerza contra electromotriz
Ke: %f\n',Ke);
fprintf(fid, '      Constante de Servoamplificador Ka:
%d\n',Ka);
fprintf(fid, '      Factor de amortiguamiento Zeta:
%f\n',Zeta);
fprintf(fid, '      Frecuencia Natural Wn: %d\n',Wn);

fprintf(fid, '\n');
fprintf(fid, '\n');

fprintf(fid, '*Función de Transferencia del Motor:\n\n');
fprintf(fid, '      %f\n',num);
fprintf(fid, 'G(s): -----
--\n');
fprintf(fid, '      %d + %d + %d\n',Denf1,Denf2,Denf3);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 1
fprintf(fid, '*Paso 1: Cálculo de Polos:\n');
fprintf(fid, '      P1: %f\n',P1);
fprintf(fid, '      P2: %f\n',P2);
fprintf(fid, '*Paso 1: Cálculo de Polos en lazo cerrado:\n');

```

```
fprintf(fid, '      Polo1: %f + %fi\n', Polo1_Re, Polo1R_Im);  
fprintf(fid, '      Polo2: %f - %fi\n', Polo1_Re, Polo1R_Im);
```

```
fprintf(fid, '\n');  
fprintf(fid, '\n');
```

`%Paso 2`

```
fprintf(fid, '*Paso 2: Cálculo de Ángulos:\n');  
fprintf(fid, '      Phi1: %d\n', Phi_1);  
fprintf(fid, '      Phi2: %d\n', Phi_2);  
fprintf(fid, '      Phi3: %d\n', Phi_3);
```

```
fprintf(fid, '\n');  
fprintf(fid, '\n');
```

`%Paso 3`

```
fprintf(fid, '*Paso 3: Cálculo de Z:\n');  
fprintf(fid, '      Z: %f\n', Z1);
```

```
fprintf(fid, '\n');  
fprintf(fid, '\n');
```

`%Paso 4`

```
fprintf(fid, '*Paso 4: Cálculo de Td:\n');  
fprintf(fid, '      Td: %f\n', Td);
```

```
fprintf(fid, '\n');  
fprintf(fid, '\n');
```

`%Paso 5`


```

fprintf(fid, '*Paso 5: Cálculo de Norma de controlador
"Kp(Norma)":\n');
fprintf(fid, '      Norma de controlador: %f\n', Norma1);
fprintf(fid, '      Norma de la planta: %f\n', Norma22);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 6
fprintf(fid, '*Paso 6: Cálculo de Kp:\n');
fprintf(fid, '      Kp: %f\n', Kp);

fprintf(fid, '\n');
fprintf(fid, '\n');

% Close file
fclose(fid);

```

Diseño de Perfil de Velocidad y sintonización de controlador de posición

```

clc
clear all
close all

%Datos
V=48; % [V]
d=0.35;% [m]
t=2.5; % [s]
Kb=0.00057;% [Nms/rad]
J=0.0028;% [Kgm^2]
Kt=0.02;% [Nm/A]
paso=0.04;% [m/rev]
Ka=1;% [A/V]
Tf=0.1073356;% [Kgm]
Rm=4.5;
DAC=0.00122

%%
%aceleracion

Sm=(t*(d/paso))/60 %[rpm]

```

```

tacc=t/3

acc=Sm/(9.55*tacc)

%%
%Torque
Tacc=J*acc+Tf
Tdec= -(J*acc)+Tf
Trun=Tf
Trms= sqrt(((Tacc^2)*tacc)+((Tdec^2)*tacc)+((Trun^2)*tacc))/(t)

%%
%Corriente
Iacc= Tacc/Kt
Irms=Trms/Kt

%%
%Potencia
Pdel=(Tacc*Sm*746)/63025
Pdiss=(Iacc^2)*Rm*1.5
Potencia=Pdel+Pdiss

%%
%Controlador
Wc=Sm
Kf=318;% Valor Propuesto
s=Wc;
tetam=60;
Ts=0.0015 %Valor Propuesto

G=DAC*Ka*Wc*10*Kf
Arg=(G/-s^2)

beta=tetam-175+180+((180*Wc*Ts)/(2*pi))

P=(1/-Arg)*cosd(beta)
D=(1/(Wc*-Arg))*sind(beta)

```

Para tener un reporte de los resultados arrojados por MATLAB se realizó el siguiente código para que MATLAB hiciera un archivo de texto con los resultados:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Graficas de perfil de velocidad
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure()
hold on
grid on

P1=[0 0];P2=[tacc Tacc];
P3=[tacc Tacc];P4=[2*tacc Trun ];

```

```

P5=[2*tacc Tdec];P6=[3*tacc 0];

plot([P1(1) P2(1)], [P1(2) P2(2)], 'r')
plot([P3(1) P4(1)], [P3(2) P4(2)], 'r')
plot([P5(1) P6(1)], [P5(2) P6(2)], 'b')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Reporte de Controlador
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Open file
fid = fopen('Reporte_carlos2.txt', 'wt');

% Description header
fprintf(fid, '-----\n');
fprintf(fid, '-- Reporte de Perfil de Velocidad de Motor y --\n');
fprintf(fid, '-- Sintonización de Controlador de Posición --\n');
fprintf(fid, '-- FI - UAQ --\n');
fprintf(fid, '-- Servo Systems --\n');
fprintf(fid, '-- Carlos Miguel Torres Hdz --\n');
fprintf(fid, '-----\n');
fprintf(fid, '\n');

% Datos del Motor
fprintf(fid, '*Datos:\n');
fprintf(fid, ' Constante de Friccion Kb: %f\n', Kb);
fprintf(fid, ' Momento de Inercia J: %f\n', J);
fprintf(fid, ' Constante de torque Kt: %f\n', Kt);
fprintf(fid, ' Paso del tornillo Paso: %f\n', paso);
fprintf(fid, ' Constante de Servoamplificador Ka: %d\n', Ka);
fprintf(fid, ' Distancia d: %f\n', d);
fprintf(fid, ' Tiempo t: %f\n', t);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 1
fprintf(fid, '*Paso 1: Cálculo de Aceleración:\n');
fprintf(fid, ' Velocidad Max Sm: %f\n', Sm);
fprintf(fid, ' Tiempo aceleracion: %f\n', tacc);
fprintf(fid, ' Tiempo funcionamiento: %f\n', tacc);
fprintf(fid, ' Tiempo desaceleración: %f\n', tacc);
fprintf(fid, ' Aceleracion: %f\n', acc);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 2
fprintf(fid, '*Paso 2: Cálculo de Par Requerido:\n');
fprintf(fid, ' Tacc: %f\n', Tacc);
fprintf(fid, ' Tdec: %f\n', Tdec);

```

```

fprintf(fid, '      Trun: %f\n',Trun);
fprintf(fid, '      Trms: %f\n',Trms);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 3
fprintf(fid, '*Paso 3: Cálculo de Corriente:\n');
fprintf(fid, '      Iacc: %f\n',Iacc);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 4
fprintf(fid, '*Paso 4: Cálculo de Potencia:\n');
fprintf(fid, '      Pdel: %f\n',Pdel);
fprintf(fid, '      Pdis: %f\n',Pdis);
fprintf(fid, '      Potencia: %f\n',Potencia);

fprintf(fid, '\n');
fprintf(fid, '\n');

fprintf(fid, '*Datos para Sintonia de Controlador:\n');
fprintf(fid, '      Constante Servoamplificador Ka: %f\n',Ka);
fprintf(fid, '      Constante encoder Kf: %f\n',Kf);
fprintf(fid, '      Constante de DAC DAC: %f\n',DAC);
fprintf(fid, '      Tiempo de muestreo Ts: %f\n',Ts);
fprintf(fid, '      Frecuencia de corte Wc: %f\n',Wc);
fprintf(fid, '      Margen de fase Tetam: %f\n',tetam);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 1
fprintf(fid, '*Paso 1: Cálculo de Argumento:\n');
fprintf(fid, '      Argumento Arg: %f\n',Arg);
fprintf(fid, '      beta: %f\n',beta);

fprintf(fid, '\n');
fprintf(fid, '\n');

%Paso 2
fprintf(fid, '*Paso 2: Cálculo de Parámetros:\n');
fprintf(fid, '      Proporcional: %f\n',P);
fprintf(fid, '      Derivativo: %f\n',D);

% Close file
fclose(fid);

```