

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería Electrónica



**Automatización del proceso de verificación y calibración de
equipos de medición de resistencias de alto y bajo valor**

**Proyecto de graduación para optar por el título de Ingeniero
Electrónico con el grado académico de Licenciatura**

Instituto Costarricense de Electricidad (ICE)
Departamento de desarrollo y ejecución de proyectos
Plantel Colima de Tibás

Víctor Ureña Nassar

Cartago, 27 de noviembre del 2012

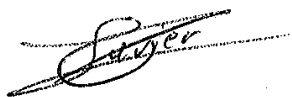
ESCUELA DE INGENIERIA ELECTRONICA

PROYECTO DE GRADUACIÓN

TRIBUNAL EVALUADOR

Proyecto de Graduación defendido ante el presente Tribunal Evaluador como requisito para optar por el título de Ingeniero en Electrónica con el grado académico de Licenciatura, del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal



Ing. Javier Pérez Rodríguez

Profesor lector



Ing. Carlos Badilla Corrales

Profesor asesor

Los miembros de este Tribunal dan fe de que el presente trabajo de graduación ha sido aprobado y cumple con las normas establecidas por la Escuela de Ingeniería Electrónica

Cartago, 27 de noviembre de 2012



Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado.

Cartago, viernes 27 de noviembre de 2012



Víctor Ureña Nassar

Cédula: 2 627 343

Resumen

El Laboratorio de Metrología (LAMETRO) nació en diciembre del 2004 como parte del “Proyecto de Metrología de la Dirección Técnica Desarrollo y Ejecución de Proyectos”, forma parte del Instituto Costarricense de Electricidad. Perteneció al Departamento de desarrollo y ejecución de proyectos ubicado en el Plantel de Colima de Tibás.

Este laboratorio tiene como objetivo brindar los servicios de verificación y calibración de equipos relacionados con mediciones en fibra óptica para telecomunicaciones, así como los servicios de calibración de magnitudes físicas relacionadas con la electricidad. El presente proyecto se desarrolló en relación al servicio de calibración de variables eléctricas, específicamente con medidores de resistencia de alto y bajo valor.

El problema que existe en el laboratorio es que no cuenta con un sistema automatizado para la verificación y calibración de los tipos de patrones de resistencia de alta y baja exactitud. Además existe una carencia de un mecanismo control en tiempo de real de las mediciones en el laboratorio y de los factores que afectan las pruebas en el sistema.

Para alcanzar una solución óptima de este problema se proyectó implementar una planta que permita variar las magnitudes de los patrones de resistencia para las pruebas de medidores de resistencia, así como establecer el debido modelo de control para dicho sistema que permita reducir el tiempo necesario para ejecutar cada verificación y calibración. Al mismo tiempo, con esto, se pretendió contar con un mayor control sobre los factores externos que afectan el proceso de calibración.

Palabras claves: Laboratorio de Metrología, calibración, variables eléctricas, medidores de resistencia, factores externos, microprocesador.

Abstract

The Metrology Laboratory (LAMETRO) was born in December 2004 as part of the "Project Metrology Technical Management Development & Execution", and it is part of the Instituto Costarricense de Electricidad (ICE). It belongs to the Department of Development and implementation of Projects located in the campus of Colima, Tibás.

This laboratory aims to provide verification services and related equipment calibration measurements in optic fiber, telecommunications and calibration services of physical quantities related to electricity. This project was developed in relation to the service of electrical calibration, specifically resistance meters of high and low value.

The problem in this laboratory is that they don't have an automated system for the verification and calibration of the types of resistance patterns of high and low accuracy. Furthermore there is a lack of a mechanism of real time control of the laboratory measurements and factors affecting the system tests.

To achieve an optimal solution to this problem, is to implement a plan designed to allow varying magnitudes of resistance patterns for testing resistance meters and ensure an appropriate control model for the system to reduce the time required to execute each verification and calibration. At the same time, it is intended to have greater control over external factors affecting the calibration process.

Keywords: Metrology Laboratory, calibration, electrical variables, resistance meters, external factors, microprocessor.

Tabla de contenido

1	CAPÍTULO 1: INTRODUCCIÓN	11
1.1	DESCRIPCIÓN DE LA EMPRESA.....	11
1.1.1	DESCRIPCIÓN GENERAL	11
1.1.2	DESCRIPCIÓN DEL DEPARTAMENTO	12
1.2	DEFINICIÓN DEL PROBLEMA	15
1.3	SÍNTESIS DEL PROBLEMA	16
1.4	ENFOQUE DE LA SOLUCIÓN	16
2	CAPITULO 2: META Y OBJETIVOS.....	18
2.1	META	18
2.2	OBJETIVO GENERAL.....	18
2.3	OBJETIVOS ESPECÍFICOS	19
3	CAPITULO 3: MARCO TEÓRICO	20
3.1	INVESTIGACIÓN SOBRE PATRONES DE RESISTENCIA DE ALTO Y BAJO VALOR	20
3.1.1	CAJA DE DÉCADAS DE ALTA RESISTENCIA, HRRS- 5kV	21
3.1.2	CAJA DE DÉCADAS DE BAJA RESISTENCIA, HARS-X	24
3.2	SERVOMOTORES	26
3.3	SENSORES DE PROXIMIDAD INFRARROJOS	28
3.4	MICROPROCESADOR	30
3.4.1	CARACTERÍSTICAS	31
3.5	TIPOS DE CONTROL AUTOMÁTICO	32
4	PROCEDIMIENTO METODOLÓGICO	35
4.1	USOS DE RECURSOS	37
5	DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN	38
5.1	DESCRIPCIÓN DE LA PLANTA Y FUNCIONAMIENTO	38
5.2	INTERFACE	43
5.2.1	INTERFACE DE PC EN EL LABORATORIO.	43
5.3	INTERFACE WEB.....	49
5.3.1	CODIFICACIÓN DE DATOS.	50
5.3.2	INTERFAZ EN JAVA CON COMUNICACIÓN POR PUERTO SERIAL.	52
5.1.2.1	CONDICIONES DE SEGURIDAD Y USO DE LA INTERFAZ:	52
5.3.3	COMUNICACIÓN DEL PUERTO SERIAL EN JAVA.	53
5.4	UNIDAD DE CONTROL Y COMUNICACIÓN	54
5.4.1	MÓDULO EUSART.....	54
5.4.2	MÓDULOS DE TIMER:.....	55
5.4.3	MÓDULO DE ADC:	56

5.4.4	MÓDULO DEL LCD	57
5.4.5	SISTEMA DE PWM	58
5.4.6	MÓDULO DE ETHERNET	60
5.4.7	CONTROL DE LA PLANTA.....	62
5.4.8	FLUJO DEL PROGRAMA	68
5.5	AMPLIFICACIÓN DE LA SEÑAL DE PWM.	76
5.5.1	PINZA BIG GRIPPER.....	76
5.5.2	CABLE STP REFORZADO	77
5.5.3	SEÑAL DEL SENSOR.....	78
6	<u>PRESUPUESTO.....</u>	<u>79</u>
7	<u>CONCLUSIONES.....</u>	<u>80</u>
8	<u>RECOMENDACIONES.....</u>	<u>81</u>
9	<u>GLOSARIO.....</u>	<u>82</u>
10	<u>BIBLIOGRAFÍA.....</u>	<u>84</u>
11	<u>ANEXOS</u>	<u>85</u>
11.1	PASO DEL PROGRAMA EN FORMATO DE CÓDIGO A FORMATO EJECUTABLE.	85
11.2	ESQUEMA DE CONEXIÓN A LOS PINES DEL KIT DE DESARROLLO.	86
11.3	OBTENCIÓN EMPÍRICA DEL MODELO DEL SERVOMOTOR PARA LA PLANTA.	88

Índice de Figuras

Figura 1. Caja de décadas de alta resistencia, HRRS- 5kV	13
Figura 2. Caja de décadas de resistencias, Test Equipment Depot 279301	13
Figura 3. Diagrama de bloques propuesto para implementar el “Sistema de automatización del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor”	17
Figura 4. Configuración interna del HRRS-5kV.	23
Figura 5. Conexión óptima de los terminales.	25
Figura 6. Ejemplo de señal de control para un servomotor.	27
Figura 7. Circuito de control del servomotor.	28
Figura 8. Sensor infrarrojo Sharp GP2Y0A21YK0F.	29
Figura 9. Triángulos formados por el emisor, objeto y receptor	29
Figura 10. Diagrama de bloques del sensor Sharp GP2Y0A21YK0F	30
Figura 11. y Figura 12. Microprocesador PIC Maxi Web.	32
Figura 13. Diagrama de bloques de lazo abierto.	32
Figura 14. Diagrama de bloques de lazo cerrado.	33
Figura 15. Planta de calibración y verificación de equipos de medición de resistencias de alto y bajo valor.	38
Figura 16. Detalle de pinza con conexión a servomotores.	39
Figura 17. Detalle de riel plástico de la planta y contrapeso.	40
Figura 18. Detalle de sensor de proximidad.	40
Figura 19. Detalle microprocesador y etapa de amplificación.	41
Figura 20. Y Figura 21. Estructuras auxiliares para las cajas de décadas y conmutadores.	42
Figura 22. Diagrama de bloques propuesto para implementar el sistema.	43
Figura 23. Icono del software en el escritorio de la PC.	44
Figura 24. Menú selección de datos inicial.	44
Figura 25. Selección de equipo a utilizar.	45
Figura 26. Selección de conmutador individual.	46
Figura 27. Ingresar tiempo de prueba a un conmutador seleccionado.	46
Figura 28. Ingresar tiempo de prueba para todos los conmutadores.	47
Figura 29. Ingresar diferentes tiempos de prueba para los conmutadores.	47
Figura 30. Selección de puerto serial a utilizar.	48
Figura 31. Acceso a la interface vía Ethernet.	49
Figura 32. Acceso a la interface vía Ethernet.	50
Figura 33. Flujo del proceso para generar la señal de PWM.	58
Figura 34. Diagrama de tiempos de la señal de PWM generada con el método de timer.	58

Figura 35. Diagrama de la asociación de las funciones de TCPIP Stack [5].....	60
Figura 36. Diagrama de flujo servomotor.	62
Figura 37. Posición de salida del sistema y escalón de entrada del sistema.....	64
Figura 38. Señal de salida simulada contra datos medidos experimentalmente. .	65
Figura 39. Diagrama de polos y ceros del sistema.....	66
Figura 40. Diagrama de flujo para la extracción de los datos enviados.	71
Figura 41. Diagrama de flujo del programa del microcontrolador.....	75
Figura 42. Configuración del transistor para aumentar la tensión del PWM.	76
Figura 43. Pinza Big Gripper.	77
Figura 44. Cable STP reforzado.....	78
Figura 45. Comportamiento de la señal del sensor.	78
Figura 46. Puertos de conexión del kitt de desarrollo. [2].....	86
Figura 47. Numeración de los pines del puerto X1-X6. [2]	87
Figura 48. Numeración de los pines del puerto de extensión. [2].....	88
Figura 49. Importación de datos al workspace de Matlab en forma de vectores. ..	88
Figura 50. Ventana para importar datos al ident.	89
Figura 51. Selección de dos rangos del total de datos.....	90
Figura 52. Ventana del ident con los modelos obtenidos.	90
Figura 53. Comparación de modelos con Model output.	91

Índice de Tablas

Tabla 1. Especificaciones Caja de décadas de alta resistencia, HRRS- 5kV	24
Tabla 2. Especificaciones Caja de décadas de baja resistencia, HARS-X	25
Tabla 3. Estructura y comportamiento de envío de órdenes.	51
Tabla 4. Respuestas obtenidas del servomotor.	63
Tabla 5. Condiciones para el sistema de control.....	66
Tabla 6. Tabla del presupuesto del proyecto contando todos los gastos y recursos que se tiene en el laboratorio, la totalidad se encuentra en colones.....	79

1 Capítulo 1: Introducción

1.1 Descripción del Instituto Costarricense de Electricidad (ICE)

1.1.1 Descripción general

El Instituto Costarricense de Electricidad (ICE) fue creado como institución autónoma el 8 de Abril de 1949, por el Decreto - Ley No. 449, definido con el nombre de “Ley de Creación del Instituto Costarricense de Electricidad”, de la Junta Fundadora de la Segunda República. Representando el final de una larga lucha llevada por los costarricenses con el fin de solucionar el problema de la escasez de energía eléctrica y pérdida de la soberanía nacional en el ámbito de la explotación de los recursos hidroeléctricos, ya que éstos eran desarrollados por empresas privadas.

En sus inicios al ser la Institución encargada de generar fuentes para la energía eléctrica, se le designaron entre sus principales funciones las siguientes: construcción de plantas eléctricas y sus respectivas redes de distribución, motivar el desarrollo del país en el uso de la energía eléctrica como fuente de fuerza motriz, procurar el uso racional de los recursos naturales, los procedimientos técnicos, administrativos y financieros deben ser modelos de eficiencia capaces de que la Institución tenga un buen funcionamiento.

En 1963 por medio de la Ley No. 3226, la Asamblea Legislativa le asignó al ICE un nuevo objetivo, el cual fue: el establecimiento, mejoramiento, extensión y operación de los servicios de comunicaciones telefónicas, radiotelegráficas y radiotelefónicas en el territorio nacional. De esta manera en 1966 se instaló las primeras centrales telefónicas automáticas, dando origen a las telecomunicaciones.

El ICE ha evolucionado en el campo de la energía eléctrica y las telecomunicaciones, para mantenerse al tanto en las nuevas tecnologías que ofrece el mercado mundial. Es por esto que la Institución ha evolucionado como un grupo de empresas estatales conformadas por: ICE, Radiográfica Costarricense S.A. y la Compañía Nacional de Fuerza y Luz. Estas empresas buscan la modernización mediante diversos proyectos utilizando tecnología de punta con el fin de satisfacer al cliente.

1.1.2 Descripción del Laboratorio de Metrología (LAMETRO)

El Laboratorio de Metrología (LAMETRO) del Departamento de desarrollo y ejecución de proyectos ubicado en el Plantel de Colima de Tibás, forma parte del Instituto Costarricense de Electricidad dentro de la Gerencia de Telecomunicaciones, División Banda Ancha y de la Dirección Desarrollo Infraestructura de Banda Ancha. Nació en diciembre del 2004 como parte del “Proyecto de Metrología de la Dirección Técnica Desarrollo y Ejecución de Proyectos” con el fin de brindar los servicios de verificación y calibración de equipos relacionados con mediciones en fibra óptica para telecomunicaciones, así como los servicios de calibración de magnitudes físicas relacionadas con la electricidad.

La política de calidad del Laboratorio de Metrología consiste en “mantener patrones metrológicos en electricidad, magnetismo e infocomunicaciones, con trazabilidad al Sistema Internacional de Unidades de Medida (SI) y dar servicios de calibración que cumplan con una relación de incertidumbre que satisfaga las necesidades del ICE y de los clientes del Laboratorio”.^[1] Entre las principales necesidades del ICE y de los clientes del laboratorio se encuentran la calibración de equipos Medidores de potencia óptica, Atenuadores ópticos, Fuentes ópticas, Analizadores de espectro óptico (OSA) y Reflectómetros ópticos en el dominio del tiempo (OTDR's).

LAMETRO se encuentra acreditado ante el Ente Costarricense de Acreditación (ECA) desde el 21 de setiembre del 2011, lo que respalda la calidad de los servicios brindados por este laboratorio. Esta acreditación respalda la competencia de las actividades técnicas y de gestión del laboratorio según normativa INTE-ISO/IEC 17025:2005. Se utiliza un sistema de informes de calibración los cuales cuentan con una validez y reconocimiento a nivel nacional e internacional.

Entre los servicios de calibración que se ofrecen para equipos de medición en variables eléctricas se encuentran: multímetros digitales, amperímetros, analizadores de redes telefónicas y medidores de resistencia de alto y bajo valor.

La calibración de este último tipo de equipos se realiza por medio de patrones de resistencias de alta exactitud, los cuales tienen magnitudes en el orden de 1 M Ω a 1,1 k Ω (denominada caja de décadas de baja resistencia) y entre 1 M Ω hasta 111 G Ω (denominada caja de décadas de alta resistencia). En la figura 1 se muestra la caja de décadas de alta resistencia HRRS-5kV de la marca IET LABS.^[2]



Figura 1. Caja de décadas de alta resistencia, HRRS- 5kV

En la figura 2 se muestra la caja de décadas de resistencia HARS-X SERIES. ^[3]



Figura 2. Caja de décadas de resistencias, Test Equipment Depot 279301

En el laboratorio estos patrones de resistencia son verificados por medio de un multímetro de alta exactitud.

Para realizar esta verificación se toma una determinada cantidad de mediciones en cada magnitud que el equipo permita. Los equipos están compuestos por 5 conmutadores que representan los factores multiplicativos de la magnitud de la resistencia, con 11 posiciones cada uno, los cuales van desde el 0 hasta el 10.

La verificación de la precisión de estas décadas la realiza una persona encargada de variar las magnitudes de los conmutadores aproximadamente entre

cada 5 y 7 minutos, por lo que para verificar la totalidad de cada patrón se tarda aproximadamente 2 días.

Para calibrar en forma correcta los equipos, se debe garantizar que el porcentaje de humedad sea menor al 50%, así como que la temperatura en el laboratorio se encuentre en un valor nominal de 23° C. Las mediciones en el laboratorio no pueden verse afectadas por campos magnéticos mientras se realizan, razón por la cual las instalaciones se encuentran totalmente aisladas contra este tipo de campos.

Para un futuro el “Laboratorio de Metrología (LAMETRO)” pretende implementar en estos patrones calibraciones internas para diferentes voltajes, con valores desde los 50 V hasta los 1000 V.

Para cada una de estas tensiones se debe verificar las diferentes magnitudes de resistencias que los equipos permiten. Por la complejidad de este proceso estas pruebas deberán ser más rigurosas, ejecutar una mayor cantidad de mediciones, de mayor atención y cuidado que demandan un mayor tiempo, lo que produciría que para una correcta verificación y calibración de los sistemas el tiempo se prolongue de 8 a 15 días.

1.2 Definición del problema

El Laboratorio de Metrología (LAMETRO) del Instituto Costarricense de Electricidad cuenta con diferentes sistemas para brindar los servicios de calibración y asesoramiento en el uso de equipos de medición de variables eléctricas.

Estos servicios se brindan para la institución así como para empresas privadas, bajo una serie de normas que aseguran una alta calidad en las características deseadas.

Para el caso de los medidores de resistencia de alto y bajo valor, el equipo de calibración consiste en tres tipos de patrones de resistencia de alta exactitud.

En la actualidad, la verificación de los tipos de patrones de resistencias de alta exactitud se realiza manualmente, por lo que el problema principal es que se produce una serie de desventajas tales como que se necesita un operario para que se dedique completamente a variar las magnitudes de los conmutadores de los patrones, aproximadamente cada 5 a 7 minutos, lo que conlleva a un total de 2 días de su jornada semanal para realizarle las pruebas a un solo patrón.

Asimismo, en ocasiones ocurre que las pruebas se deben pausar, repetir, retroceder, cambiar tiempos de medición, entre otras opciones, debido a factores externos que afectan la correcta comprobación de las magnitudes esperadas y, al realizarse estas pruebas manualmente, se debe considerar el factor humano como un factor el cual aumenta las posibilidades de error al momento de realizar la verificación deseada.

El sistema de calibración debe mantener las normas establecidas para realizar una correcta verificación y calibración de los equipos. En el laboratorio se cuenta con dispositivos específicos para cada variable por lo que se debe llevar un control de estas características pero no se cuenta con un sistema que permita el control en tiempo real cuando las pruebas se vean afectadas por estos factores.

1.3 Síntesis del problema

El Laboratorio de Metrología (LAMETRO) no cuenta actualmente con un sistema automatizado para la verificación y calibración de los tipos de patrones de resistencia de alta y baja exactitud. Además existe una carencia de un mecanismo control en tiempo de real de las mediciones en el laboratorio y de los factores que afectan las pruebas en el sistema.

1.4 Enfoque de la solución

El Laboratorio de Metrología tiene proyectado automatizar la verificación y calibración de sus patrones. Para alcanzar una solución óptima de este problema se proyectó implementar una planta que permita variar las magnitudes de los patrones de resistencia para las pruebas de medidores de resistencia, así como establecer el debido modelo de control para dicho sistema que permita reducir el tiempo necesario para ejecutar cada verificación y calibración. Al mismo tiempo, con esto, se pretendió contar con un mayor control sobre los factores externos que afectan el proceso de calibración.

Para llevar a cabo la automatización del sistema se realizó una investigación sobre los patrones de resistencia de alto y bajo valor utilizados; esto con el fin de entender su funcionamiento, además de tomar en cuenta los factores por los que se ven afectados estos patrones a la hora de su funcionamiento y de este modo considerar estos datos para así plantear la automatización de los equipos.

Una vez dominadas las características de los patrones de resistencia, se deberá realizar una investigación sobre que tipo de planta es más eficiente para cumplir con los requisitos deseados y por medio del uso de servomotores; tomando además en cuenta componentes electrónicos y de diseño estructural para obtener la mayor eficiencia posible, así como el mejor acople a los conmutadores de los patrones.

Asimismo, se realizó una investigación para desarrollar la implementación de un sistema efectivo de automatización, tomando en cuenta todas las variables por las cuales el sistema se ve afectado.

Además se implementó un sistema que permite mover los conmutadores, de modo que el hardware utilizado no produzca ningún campo magnético para no

introducir errores en la medición, por lo que se realizó un análisis de los componentes utilizados con respecto a este tema.

Una vez diseñada la planta y su respectivo control se diseñó una interfaz para internet y el operario en el laboratorio, la cual permita un control en tiempo real del proceso y le permita al usuario tener completo control del sistema para así modificar el orden y el tiempo en las verificaciones y calibraciones llevadas a cabo. La interfaz diseñada para internet es compatible con la red interna del ICE, así como contemplar un sistema de seguridad en el que solo los empleados del laboratorio tengan acceso.

En la Figura 3 se muestra el diagrama de bloques del “Sistema de automatización del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor” que se implementó en el presente Proyecto de Graduación.

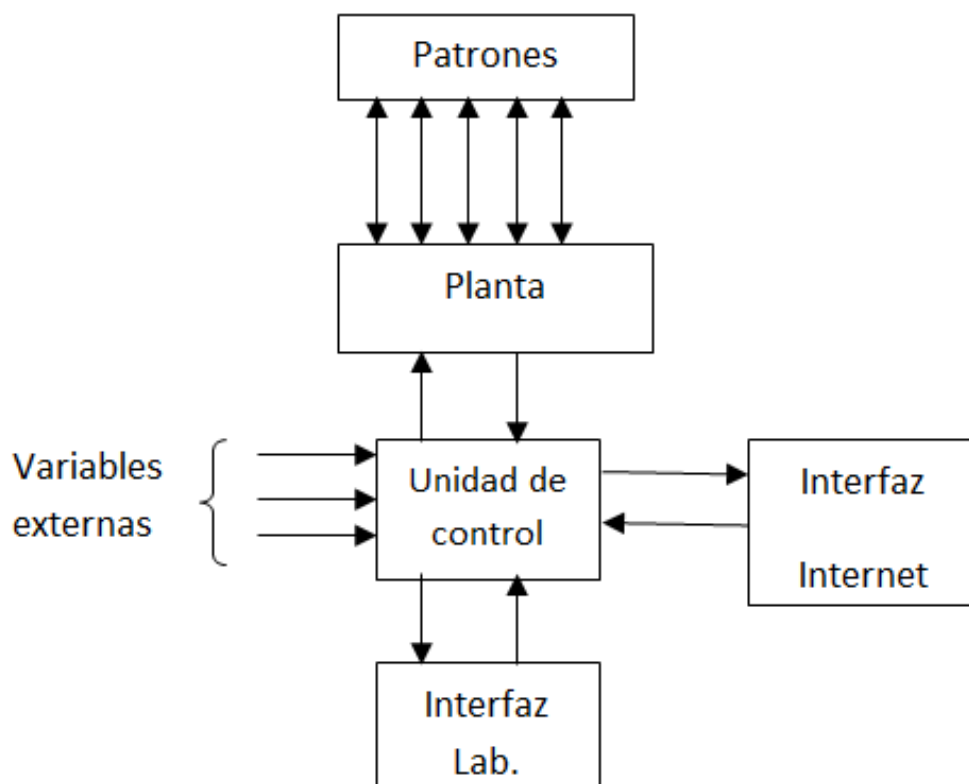


Figura 3. Diagrama de bloques propuesto para implementar el “Sistema de automatización del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor”

2 Capítulo 2: Meta y objetivos

2.1 Meta

Implementar un sistema de control y automatización para variar las magnitudes de los patrones de resistencia de alto y bajo valor, utilizados para la verificación y calibración de equipos medidores de resistencia y asegurar un alto nivel de precisión en el proceso de medición.

2.2 Objetivo general

Realizar un sistema capaz de verifique y calibre automáticamente los patrones de resistencia de los equipos utilizados para hacer pruebas a dispositivos medidores de resistencias. La automatización de este proceso debe tomar en cuenta los factores externos que afectan la operación de este sistema.

2.3 Objetivos específicos

1. Diseñar y construir una planta que permita variar las magnitudes de los patrones de resistencia para cada prueba.
2. Implementar la planta de variación de magnitudes con un sistema de control automatizado.
3. Desarrollar el hardware de la planta de tal forma que el campo magnético producido no afecte las verificaciones y calibraciones.
4. Diseñar una interfaz para el usuario, en lenguaje de alto nivel, que le permita controlar el proceso en tiempo real vía internet.
5. Diseñar una interfaz para el usuario del laboratorio, en lenguaje de alto nivel, que le permita controlar el proceso en tiempo real en el lugar donde se encuentra el equipo por calibrar.

3 Capítulo 3: Marco teórico

3.1 *Investigación sobre patrones de resistencia de alto y bajo valor*^[2,3]

Un patrón de medición es una representación física de una medición. De esta forma una unidad se realiza con referencia a un respectivo patrón físico o a un respectivo fenómeno natural compuesto por constantes físicas.

Para los patrones de resistencia se debe tomar en cuenta que en el Sistema Internacional de unidades el valor absoluto del ohm esta definido en términos de las unidades de longitud, masa y tiempo. Un patrón de resistencias esta compuesto por una bobina de alambre, la cual debe proporcionar una elevada resistividad eléctrica y un bajo coeficiente de temperatura-resistencia. Esta bobina resistiva debe colocarse en una estructura que la proteja de cualquier cambio en las condiciones atmosféricas, normalmente se utiliza una doble capa protectora en las paredes de la estructura.

Cuando el alambre que se utiliza en la bobina resistiva mantiene un valor aproximadamente constante en una amplia escala de temperaturas, se puede definir una expresión para determinar el valor exacto de la resistencia a cualquier temperatura. Esta expresión es la siguiente:

$$R_t = R_{25^{\circ}\text{C}} + \alpha(t - 25) + \beta(t - 25)^2 \quad (1)$$

Donde:

R_t = resistencia a la temperatura de ambiente.

$R_{25^{\circ}}$ = resistencia a 25°C.

α, β = coeficientes de temperatura.

Las aplicaciones incluyen la calibración de medidores de resistencia, la comprobación de sensores electroquímicos y biomédica. Estos instrumentos son útiles donde se requieren altas resistencias, tal como en la prueba de alta impedancia amplificadores y los aislamiento de los circuitos de baja potencia.

En el Laboratorio de Metrología se trabaja con dos tipos de patrones de resistencia:

3.1.1 Caja de décadas de alta resistencia, HRRS- 5kV

El HRRS-5kV es un patrón de resistencias de alta exactitud, forma parte de la familia de instrumentos utilizados para la verificación y calibración de equipos, con estándares de trabajo que van hasta los 10 TΩ. Esta caja de décadas brinda una gran variedad de altos rangos y un excelente desempeño en la exactitud de los valores de resistencias. Este equipo se muestra en la Figura 1.

Las magnitudes seleccionadas en este patrón de resistencias están disponibles para su uso en altas tensiones, sin alterar su coeficiente de precisión, estabilidad, temperatura y su coeficiente de tensión.

Esta serie emplea tecnología de punta para brindar la mayor exactitud a la hora de seleccionar la magnitud de resistencia seleccionada sin sacrificar propiedades eléctricas. Está diseñada con la implementación de resistencias especiales, interruptores y bornes de conexión para funcionar con altas tensiones suministrando una degradación mínima en la precisión sin alterar otras propiedades eléctricas.

Las uniones de las paredes del sistema están aisladas con Kel-F de alta resistencia, el cual no permite que el equipo absorbe humedad y de esta manera asegurar la baja corriente de fuga así como que no se vea afectado por factores externos. El Kel-F es un polímero policlorotrifluoroetileno el cual ofrece una combinación única de propiedades físicas y mecánicas, no inflamabilidad, resistencia química, cerca de cero absorción de humedad y altas propiedades eléctricas.

Cuando se utiliza el conmutador de menor magnitud entre las posiciones de 0 – 9, el valor de la resistencia se lee directamente del panel. Para contemplarle al usuario mayor rango y flexibilidad, en cada conmutador se ofrecen 10 posiciones. La posición 10 es igual a la posición 1 del siguiente conmutador según su rango de superioridad con respecto a la magnitud.

Para obtener una precisión óptima, la caja de décadas se debe mantener a una temperatura nominal de 23°C. Si existiera una variación de esta se le debe dar un tiempo de 2 horas para la estabilización de la temperatura especificada, ya que no deben existir variantes de temperatura en el momento de uso de los equipos para evitar datos erróneos. El porcentaje de humedad debe mantenerse menor al 50% en el laboratorio.

Al realizar la calibración del HRRS-5kV se debe realizar las siguientes acciones:

1. Confirmar que la resistencia de fuga entre el puesto de conexión y la tierra de la caja es $>10^6$ veces la máxima unidad de resistencia, si esta magnitud no se obtiene se debe limpiar el área alrededor de los puestos de conexión.
2. Se debe confirmar el cero de la unidad de resistencia.
3. Determinar los límites superiores e inferiores admisibles para cada ajuste de resistencia
4. Confirmar que la magnitud de la resistencia está comprendida entre estos límites después de haber revisado el cero de la unidad.

En la Figura 4 se observa la configuración interna de la caja de resistencias de alto valor. Sus especificaciones se muestran en la Tabla 1.

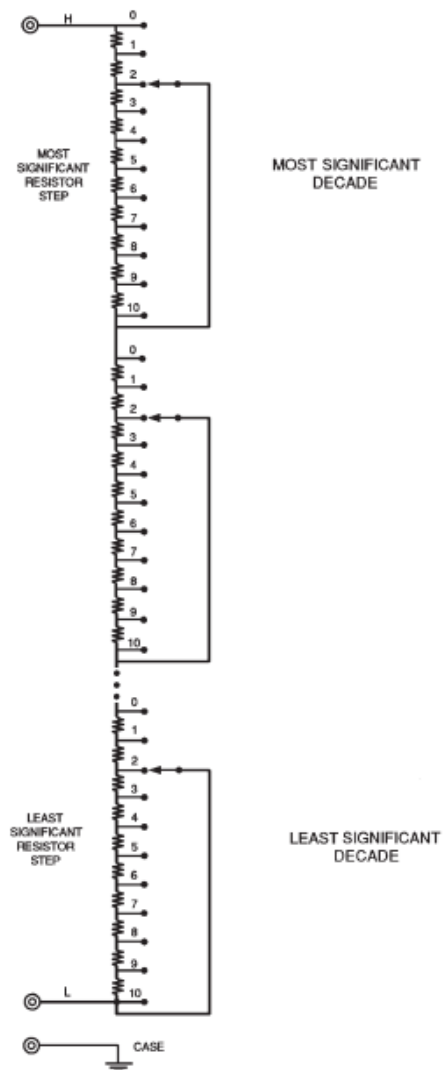


Figura 4.Configuración interna del HRRS-5kV.

Tabla 1. Especificaciones Caja de décadas de alta resistencia, HRRS- 5kV

Resistencia por paso	Decada Resist. Total	Precisión			Max voltaje por paso (V)	Maximo Voltaje (V)	Coeficiente de Temperatura ($\pm$ ppm/ $^{\circ}$ C)	Coeficiente de tensión ($\pm$ ppm/V)	Estabilidad ($\pm$ ppm/yr)	Resistencia Tipo
		Q	B	F						
10 Ω	100 Ω	0.01%	0.03%	0.1%	2.5	25	15	0	10	Bobinado
100 Ω	1 k Ω	0.01%	0.03%	0.1%	8	80	5	0	10	Bobinado
1 k Ω	10 k Ω	0.01%	0.03%	0.1%	23	230	5	0	10	Bobinado
10 k Ω	100 k Ω	0.01%	0.03%	0.1%	70	700	5	0	10	Bobinado
100 k Ω	1 M Ω	0.01%	0.03%	0.1%	230	2300	5	0	10	Bobinado
1 M Ω	10 M Ω	0.01%	0.03%	0.1%	1000	5000	15	0.2	25	Película del alto voltaje
10 M Ω	100 M Ω	0.03%	0.1%	1%	5000	5000	15	0.2	50	Película del alto voltaje
100 M Ω	1 G Ω	0.1%	0.2%	1%	5000	5000	25	1.5	100	Película del alto voltaje
1 G Ω	10 G Ω	0.2%	0.5%	1%	5000	5000	25	5	500	Película del alto voltaje
10 G Ω	100 G Ω	0.5%	1%	1%	5000	5000	25	5	500	Película del alto voltaje
100 G Ω	1 T Ω	0.5%	1%	1%	5000	5000	100	5	500	Película del alto voltaje
1 T Ω	10 T Ω	3%	5%	10%	5000	5000	300	5	500	Película del alto voltaje

3.1.2 Caja de décadas de baja resistencia, HARS-X

Para la calibración y verificación de equipos de baja resistencia en Lametro se utiliza el patrón de resistencia HARS-X. La cual es una fuente de resistencia de precisión con excelentes características de estabilidad, coeficiente de temperatura, coeficiente de potencia y su respuesta en frecuencia.

La serie HARS-X fue creada con un diseño especial para mantener el cero de la resistencia en menos de un 1m Ω por década. Los contactos de cada conmutador están compuestos por una aleación de plata, la resistencia de cada contacto están bañadas en oro y cobre telúrico permitiendo que sean mas estables y minimizando los efectos de la fuerza electromagnética térmica que artificialmente refleja un cambio en las mediciones de resistencia en CD. El resto de los conductores que forman parte del instrumento al igual que todas las soldaduras empleadas no están compuestos por metales o cruces similares que podrían generar problemas con la fuerza electromagnética térmica.

Esta caja de décadas permite valores de resistencia desde 1m Ω hasta un máximo de 111M Ω . Las especificaciones de la caja de resistencia de bajo valor se muestran en la Tabla 2.

Tabla 2. Especificaciones de la caja de décadas de baja resistencia, HARS-X

Resistencia por paso	Resistencia Total	Estabilidad ($\pm$ ppm/año)	Estabilidad a largo plazo ($\pm$ ppm/3 año)	Coefficiente Temperatura ($\pm$ ppm/ $^{\circ}$ C)	Máxima Potencia (W/paso)	Máxima Corriente (por década)	Máximo Voltaje (por paso)
1 m Ω	10 m Ω	100	700	50	0.025	5 A	5 mV
10 m Ω	100 m Ω	50	350	20	0.2	4 A	40 mV
100 m Ω	1 Ω	30	50	20	0.25	1.6 A	0.16 V
1 Ω	10 Ω	10	25	20	0.6	0.8 A	0.8 V
10 Ω	100 Ω	10	25	15	0.6	0.25 A	2.5 V
100 Ω	1 k Ω	10	25	5	0.6	80 mA	8 V
1 k Ω	10 k Ω	10	25	5	0.5	23 mA	23 V
10 k Ω	100 k Ω	10	25	5	0.5	7 mA	70 V
100 k Ω	1 M Ω	10	25	5	0.5*	2.3* mA	230 V*
1 M Ω	10 M Ω	10	25	10	0.5*	0.7* mA	700 V*
10 M Ω	100 M Ω	25	40	10	0.1*	0.1* mA	1000 V*

*Sujeto a un máximo de 2000 V.

En este patrón de resistencias la magnitud puede variar desde 1m Ω por paso hasta 10 M Ω por paso. El sistema que compone a cada conmutador esta encerrado en una lata de aluminio con el fin de que esto funcione como un escudo para evitar variaciones externas. Cada unidad de estas está conformada por una baja inductancia resistiva en serie, con un alto rendimiento compuesta por plata y diferentes aleaciones. En la Figura 2 se muestra la caja de resistencias de bajo valor.

Con el fin de hacer un uso correcto de este equipo se debe tener cuidado con las conexiones eléctricas de los terminales, con el fin de mantener la resistencia de contacto al mínimo. Estas terminales aceptan conexiones tipo banana, pinzas de cocodrilo y cable sin cobertura plástica. La conexión correcta se muestra en la Figura 5.

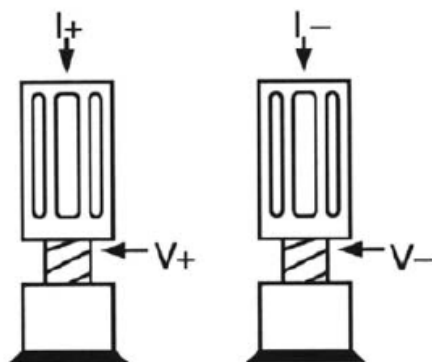


Figura 5. Conexión óptima de los terminales.

Al igual con la caja de resistencia de alto valor, en la de bajo valor cuando se utiliza el conmutador de menor magnitud entre las posiciones de 0 – 9, el valor de la resistencia se lee directamente del panel y la posición 10 es igual a la posición 1 del siguiente conmutador según su rango de superioridad con respecto a la magnitud. Para obtener la máxima precisión la temperatura en el laboratorio debe mantenerse en 23°C y un porcentaje de humedad del 50%, si alguna de estas variables se viera modificada se debe dar un tiempo de estabilización de 2 horas a partir de que estos valores se mantengan constantes.

Para la calibración de este equipo se debe seguir los siguientes pasos:

1. Confirmar el valor del cero de la resistencia.
2. Determinar los límites superiores e inferiores de cada resistencia. Para este equipo los límites se encuentran entre $[R \pm (0.0001 R + 2 \text{ m}\Omega)]$.
3. Confirmar que las magnitudes de las resistencias a utilizar se encuentren entre estos límites.
4. Si alguna magnitud queda fuera de estos límites, se debe revisar el interruptor porque puede requerir un remplazo.

3.2 Servomotores

Un servomotor es un dispositivo semejante a un motor de corriente continua el cual presenta un eje de rendimiento controlado, conformado por un motor CD, una caja reductora y un circuito de control. El servo tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación y mantener constante dicha posición, esto mediante una señal codificada. Cuando esta señal cambia la posición del servomotor va a variar según la señal recibida.

Para obtener el control de posicionamiento de este dispositivo se utiliza un circuito de control proporcional. La señal para este control debe ser cuadrada definida PWM, de esta manera el ancho del pulso de la señal indica el ángulo de posición del servomotor, ya que la ubicación depende de la duración del nivel alto de la señal. Conforme mayor sea el ancho del pulso mayor será el ángulo seleccionado para el dispositivo, en el caso contrario en el que el ancho del pulso disminuya el ángulo también disminuirá. Lo anterior se ejemplifica en la Figura 6.

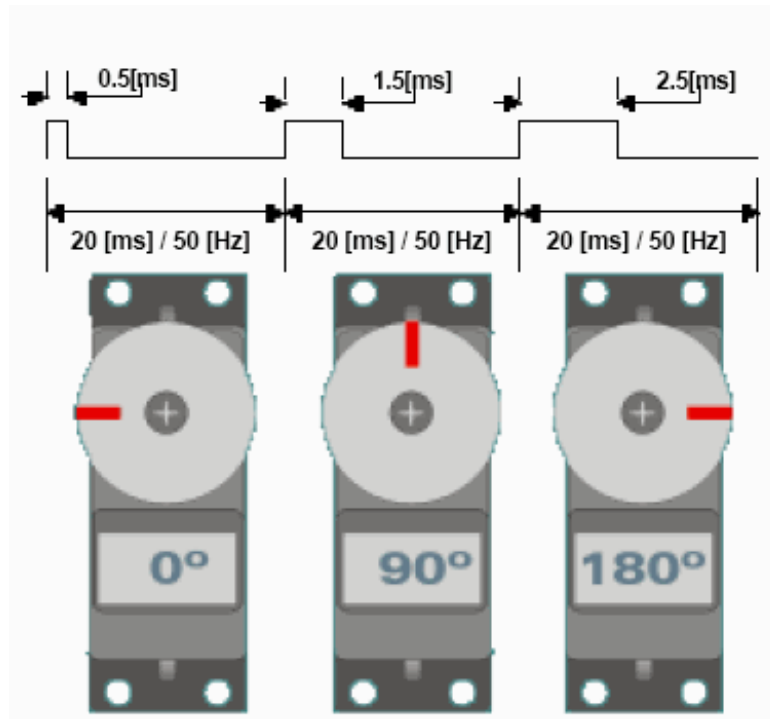


Figura 6. Ejemplo de señal de control para un servomotor.

Para esta clase de dispositivos existe una relación entre el tiempo en alto que se mantiene la señal y el ángulo de posición:

$$t = 1 + \frac{\varnothing}{180} \quad (2)$$

Donde las unidades de t deben ser en milisegundos y las de $\varnothing$ en grados. Se debe tomar en cuenta que el servomotor tiene un límite de giro por lo que no se puede superar el tiempo o ángulo de operación.

Si se desea mantener el servomotor en una posición constante se debe enviar continuamente la señal de la posición deseada, de manera que el sistema de control se mantenga en operación. Al accionar esta función el servomotor no se verá afectado por fuerzas externas mientras estas sean menores a las especificadas en la hoja de datos del dispositivo.

El circuito de control el cual está implementado en un servomotor se muestra en la Figura 7, el cual esta compuesto por un convertidor de ancho de pulso a voltaje para convertir la señal del PWM a un valor analógico de voltaje, un amplificador de error el cual indica la relación que existe entre la posición deseada y la posición real del dispositivo. Este último valor se obtiene por medio de un

potenciómetro configurado en retroalimentación el cual esta conectado a la caja reductora, de esta manera el potenciómetro varía de igual forma que el motor modificando así la señal de entrada en el amplificador.

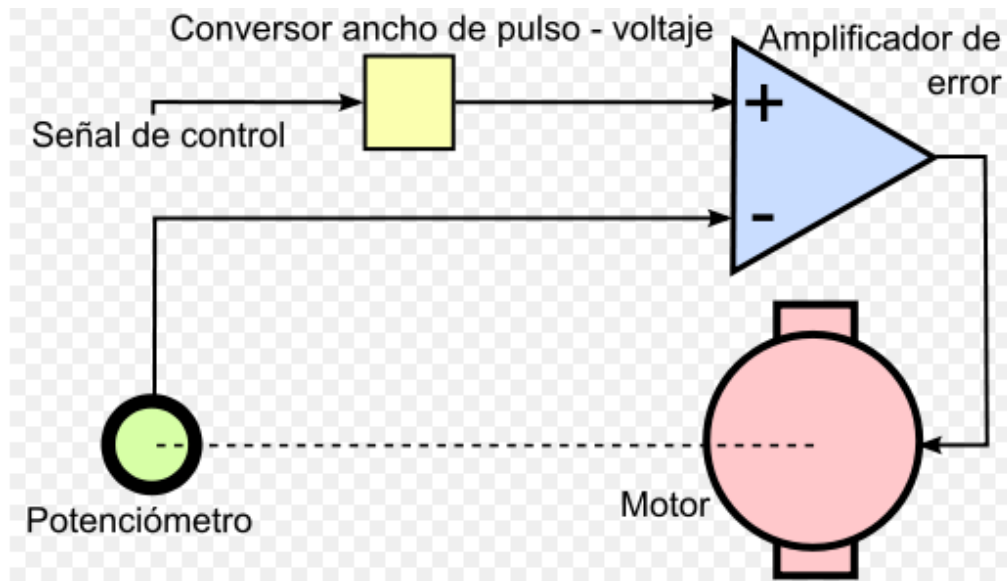


Figura 7. Circuito de control del servomotor.

3.3 Sensores de proximidad infrarrojos [7]

Dispositivos electrónicos que utilizan el principio físico de reflexión para obtener la capacidad de medir la radiación electromagnética infrarroja de los objetos en su margen de vista. Para este proyecto el Laboratorio de Metrología brindó dos sensores tipo Sharp GP2Y0A21YK0F, los cuales se muestran en la Figura 8.



Figura 8. Sensor infrarrojo Sharp GP2Y0A21YK0F.

Su funcionamiento consiste en emitir un pulso de luz IR desde el emisor, por lo que se puede presentar dos opciones las cuales son: que no exista objeto cerca o que golpee un objeto. Si se diera el hecho de que ningún objeto se encuentre en el campo de emisión del sensor nunca se reflejará la luz por lo que su respuesta indicará esta opción. En el caso del que la luz golpee algún objeto, esta se reflejara y se devolverá al detector creando un triángulo entre el emisor, el objeto y el receptor, tal como se muestra en la Figura 9.

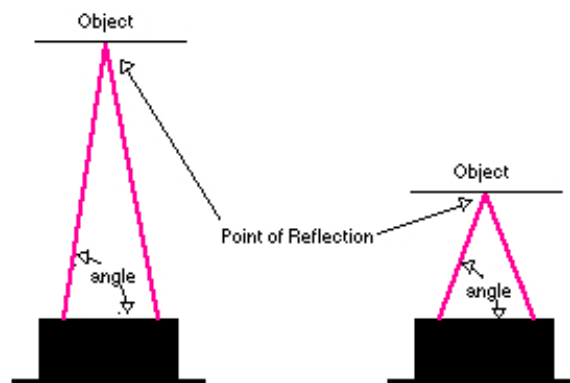


Figura 9. Triángulos formados por el emisor, objeto y receptor

Estos triángulos se encuentran en función de la distancia al objeto, esta distancia se calcula a partir del ángulo de la luz reflejada que llega al receptor. Ya que el receptor es un objetivo de precisión que transmite la luz reflejada sobre la matriz CCD lineal, esta matriz captura la luz y la convierte en una señal eléctrica analógica.

Entre las principales especificaciones del sensor infrarrojo Sharp GP2Y0A21YK0F se encuentran que la distancia máxima de medición debe ser 80cm, la distancia mínima debe de ser de 10cm, el voltaje de alimentación debe estar entre 4, 5 a 5,5 V, la salida de tensión analógica es inversamente proporcional a la distancia y la tensión de salida de 3,1 V en 10cm a 0,3 V a 80cm. El tiempo de respuesta del sensor es de 38 ± 10 ms. En la Figura 10 se muestra el diagrama de bloques del sensor infrarrojo.

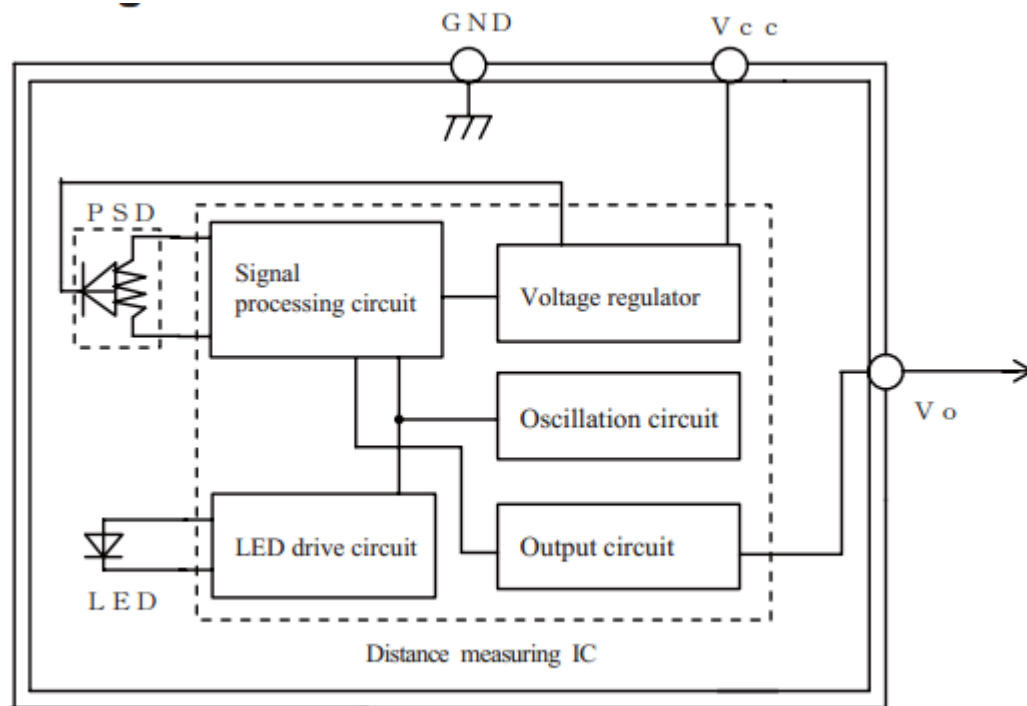


Figura 10. Diagrama de bloques del sensor Sharp GP2Y0A21YK0F

3.4 Microprocesador^[8]

Este microprocesador permite desarrollar de manera efectiva aplicaciones con conectividad Ethernet. Posee componentes necesarios y de alta importancia como relays que se pueden controlar a través de la interfaz web, una pantalla LCD, entradas analógicas para la conexión a los sensores, entradas digitales optoaisladas, potenciómetro, sensor de temperatura y conector Ethernet. Este PIC cuenta con una memoria flash de 128 KB de programación y también hay un 1 Mbit (128 KB) de memoria flash para almacenamiento de datos. Posee un convertidor CD/DC que permite que esta tarjeta sea flexible respecto a la fuente de alimentación externa, que puede variar entre 9-25V DC o 7-18V AC gama. El

punto de rectificadores, que se presenta en la entrada de CA de este microprocesador, permite el uso de adaptadores de corriente continua, desapareciendo de esta forma los problemas de suministro de polaridad.

3.4.1 Características

- Microcontrolador integrado con Ethernet MAC y PHY y 128 KB de espacio de programación Flash.
- 1 Mbit de memoria de almacenamiento en página web.
- Conector para programar ICSP / ICD y depurador con PIC-ICD2 y PIC-ICD2-POCKET.
- Sensor de temperatura.
- Potenciómetro conectado a la entrada analógica.
- Dos relays 10A/250Vca.
- Cuatro entradas digitales optoaisladas.
- Doce entradas analógicas en el bloque de terminales.
- Dos botones para asignar funciones.
- LCD16x2 con luz de fondo.
- Interfaz RS232
- Interfaz Ethernet
- Servidor de web completo y soporte de protocolo TCP-IP por medio de Microchip TCP-IP stack.
- Bloque de terminales para la fuente de alimentación funciona con 9-25 VCC.
- Cabecera de extensión para conectar con otros microprocesadores.
- Dimensiones: 120x108 mm.

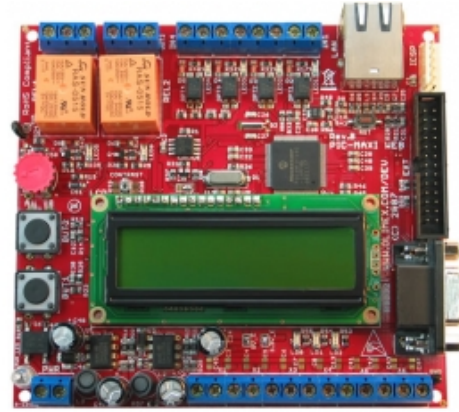


Figura 11. y Figura 12. Microprocesador PIC Maxi Web.

3.5 Tipos de control automático [12]

Un sistema de control automático es un conjunto de componentes relacionados entre si de manera que ejecuten una acción por si mismo. Se pueden clasificar en dos tipos:

- Sistema de lazo abierto:

Es aquel sistema en el cual la señal de salida no brinda ninguna información sobre la señal de entrada. Estos sistemas deben de ser muy bien calibrados para lograr una exactitud óptima de tal forma que se logre una relación entre la señal de salida y la señal de entrada.

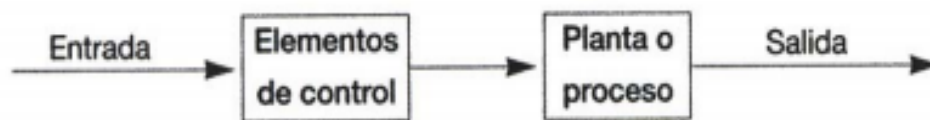


Figura 13. Diagrama de bloques de lazo abierto.

- Sistema de lazo cerrado:

Sistema en el cual existe una retroalimentación de la señal de salida, por lo que esta señal influye en la entrada. Debido a esto la señal de entrada se va a ver modificada cada vez que se altere la señal de salida, de tal forma que la acción del controlador se establece en función de ambas señales. En la Figura 14 se muestra un diagrama de bloques representando un sistema de lazo cerrado y su respectiva función de transferencia se muestra en la ecuación 3.

$$\frac{F(s)}{Y(s)} = \frac{G_p(s) \cdot G_v(s)}{1 + G_p(s) \cdot G_v(s) \cdot H(s)} \quad (3)$$

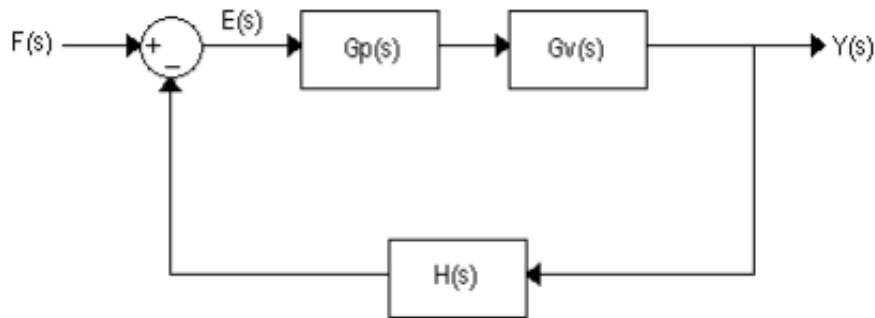


Figura 14. Diagrama de bloques de lazo cerrado.

En la Figura 14 la señal de entrada se ve representada por $F(s)$, la señal de salida es $Y(s)$, $H(s)$ es la función de transferencia de la retroalimentación, $G_v(s)$ representa la función de transferencia de la planta y $G_p(s)$ es la función de transferencia del control diseñado.

Para controlar un sistema es posible implementar un compensador o regulador que cuenta con componentes (Proporcional, Integral y Derivativo), los cuales proporcionan diferentes características para la realización del control de la planta; cada una de ellas se detalla a continuación: [10]

Proporcional: La parte proporcional consiste en el producto entre la señal de error y la constante proporcional, para que lograr que el error en estado estacionario sea casi nulo. [6] De este modo, cualquiera que sea el mecanismo real y la forma de la potencia de operación, el controlador proporcional es, en esencia, un amplificador con una ganancia ajustable.

El termino proporcional esta dado por la siguiente ecuación:

Parte proporcional = $K_p e(t)$

Donde K_p es la constante proporcional.

Integral: con el control proporcional de una planta, cuya función de transferencia no posee un integrador, hay un error en estado estable, o desplazamiento (offset), en la respuesta para una entrada escalón. Tal offset se elimina si se incluye la acción de control integral en el controlador.

En el control integral de una planta, la señal de control, es en todo momento el área bajo la curva de la señal de error hasta tal momento.

El termino integral esta dado por:

$$\text{Parte integral} = K_i \int_0^t e(t) dt$$

Donde K_i es la constante integral.

Derivativo: Cuando una acción de control derivativa se agrega a un controlador proporcional, aporta un medio de obtener un controlador con alta sensibilidad. Una ventaja de usar una acción de control derivativa es que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Por tanto, el control derivativo prevé el error, inicia una acción correctiva oportuna y tiende a aumentar la estabilidad del sistema.

Debido a que el control derivativo opera sobre la velocidad de cambio del error, y no sobre el error mismo, este control nunca se usa solo. Siempre se emplea junto con una acción de control proporcional o proporcional-integral

El término derivativo se expresa de la siguiente forma:

$$\text{Parte derivativa} = K_d \frac{d e(t)}{dt}$$

Donde K_d es la constante derivativa.

4 Procedimiento metodológico

Para el alcance de los objetivos del proyecto en el desarrollo de la solución se deberán realizar actividades, tal como se explica a continuación:

1. Diseñar e implementar una planta para variar las magnitudes de los patrones de resistencia para cada prueba.

Actividades:

- Analizar los manuales de los equipos a utilizar en el laboratorio para analizar su funcionamiento y los factores por los que se ven afectados.
- Realizar una investigación con el fin de elegir, luego de un análisis convincente, que tipo de planta es más eficiente para cumplir con los requisitos deseados.
- Investigar sobre los componentes a utilizar, en busca de cuales harían el sistema más eficiente y que cumplan con las condiciones requeridas.
- Diseñar la planta, realizar el análisis de cómo se pueden acoplar los componentes elegidos y el tipo de planta seleccionada para obtener un óptimo funcionamiento.
- Realizado los análisis anteriores se procede a la implementación de la planta.

2. Implementar un sistema de control automatizado para la planta implementada.

Actividades:

- Investigar sobre el tipo de controlador más eficiente para la planta según sus características.
- Implementar el controlador en el microcontrolador.

3. Desarrollar un sistema en el cual al mover los conmutadores, el hardware utilizado no produzca ningún campo magnético para no introducir errores en la medición.

Actividades:

- Analizar los componentes a utilizar para no tener error en las mediciones realizadas.
- Implementar un sistema en el cual mientras se estén realizando las pruebas no se requiera estar variando los conmutadores y de esta forma asegurarse que la planta no produzca ningún campo que afecte a las mediciones realizadas.

4. Diseñar una interfaz para internet y para el usuario en el laboratorio, la cual permita un control en tiempo real del proceso.

Actividades:

- Realizar un estudio para elegir un diseño adecuado para la interface, la cual sea amigable y fácil de usar para el usuario.
- Analizar las características que se deben tomar en cuenta en el hardware para implementar la interface.
- Implementar la interface para el laboratorio.
- Investigar requisitos para realizar una página en internet con todos los derechos reservados para que los usuarios del laboratorio puedan modificar y monitorear el sistema desde la web.
- Implementar la interface para página web.
- Realizar pruebas en el laboratorio con todo los equipos funcionando
- Verificar resultados.

4.1 Usos de recursos

Entre los recursos que se necesitarán para implementar el proyecto se encuentran los siguientes:

- patrones de resistencia de alto y bajo valor con alta exactitud
- 2 multímetros de alta exactitud
- 1 osciloscopio
- 2 computadoras
- Laboratorio aislado completamente de campos magnéticos
- Tarjeta de desarrollo SBC65EC
- 1 Microcontrolador PIC18F6627 de Microchip
- 5 servomotores
- Dominio web y componentes electrónicos según el diseño a implementar para el control.

5 Descripción detallada de la solución

5.1 Descripción de la planta y funcionamiento

En solución a los problemas y necesidades encontrados en el proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor, se construyó una planta que permite calibrar el equipo de manera automática con la única intervención del operario mediante un software que será descrito más adelante.

Esta planta está constituida por una estructura rectangular con cuatro soportes a la superficie, los cuales brindan estabilidad al sistema y son un medio de soporte para todos los demás elementos y componentes de la planta. Está constituida en un 100% en madera sólida, material que contribuye a la estabilidad de la estructura debido a su rigidez, resistencia mecánica y estabilidad dimensional. El tipo de acabado que se le da a la planta, además de contribuir con un valor estético agregado, contribuye a la disminución de la fricción que podría generarse entre el material huloso de las ruedas y la madera. En la Figura 15 se muestra la planta diseñada.

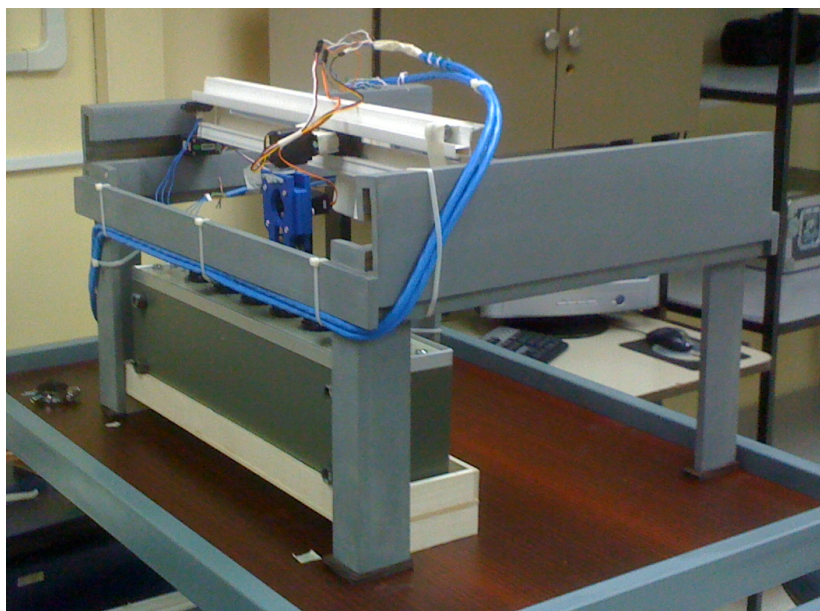


Figura 15. Planta de calibración y verificación de equipos de medición de resistencias de alto y bajo valor.

Esta estructura rectangular planteada posee en dos de sus laterales opuestos, rieles que envuelven un 90% de la superficie de las ruedas dirigidas por los servomotores. Estos rieles permiten una trayectoria controlada de estas ruedas, que generando un movimiento libre de obstáculos sobre el eje y del sistema. Sobre estos servomotores que se trasladan sobre el eje y, se encuentra otra estructura de rieles, la cual está compuesta de plástico en su totalidad, esto con el fin de generar un mínimo de peso con la estructura y de este modo evitar un mayor desgaste, o un menor desempeño en los servos por exceso de carga. Sobre estos rieles plásticos se traslada, en sentido del eje x, otra rueda conectada a otro servomotor, del cual se conecta una pinza, que mediante movimientos prensiles y giratorios repetitivos, es la responsable y encargada de variar las magnitudes de los conmutadores; la cual es manejada a su vez por dos servomotores: uno que genera el movimiento prensil y otro que permite el movimiento de rotación, esto se muestra en la Figura 16.

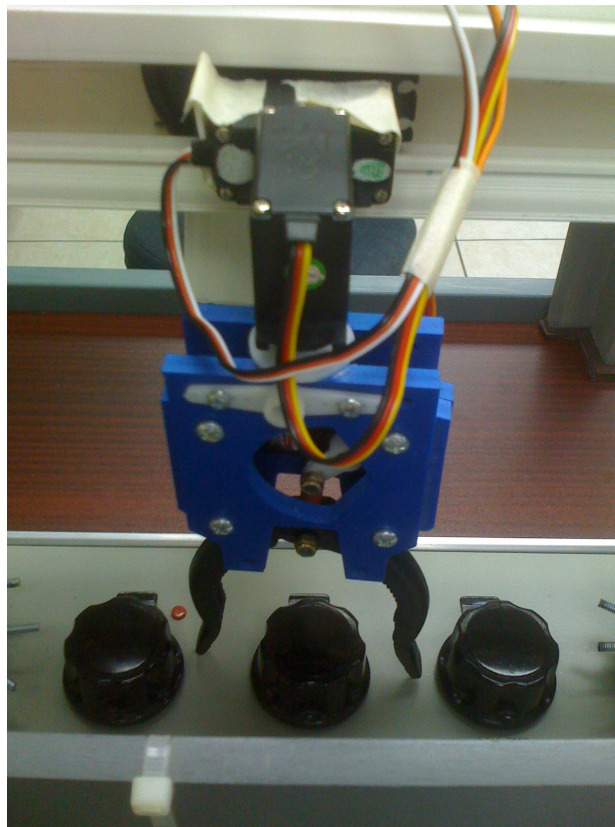


Figura 16. Detalle de pinza con conexión a servomotores.

A este subsistema conformado por riel plástico, servo, rueda y pinza, se le adhiere en su lado opuesto un contrapeso, lo cual permite un mejor desplazamiento de la rueda y aporta equilibrio al subsistema.

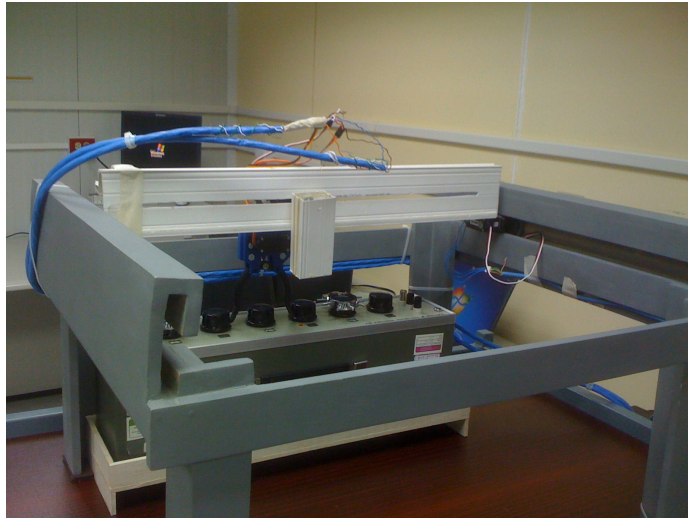


Figura 17. Detalle de riel plástico de la planta y contrapeso.

Este sistema de rieles antes descrito, genera una amplia gama de posibilidades de desplazamiento e innumerables posiciones en los ejes coordenados x y y ; lo cual permite una localización más detallada y exacta de los conmutadores que debe calibrar. En un extremo de cada uno de los tres rieles de la planta, se colocaron sensores de proximidad infrarrojos, los cuales detectan que tan próximas o lejanas se encuentran las ruedas de ellos, y así poder limitar su trayectoria sobre sus respectivos ejes de desplazamiento (Figura 18).



Figura 18. Detalle de sensor de proximidad.

Los servomotores y sensores, deben ir conectados al microprocesador, para recibir señales de control. La salida de los sensores, está conectada al microprocesador, sin embargo, los servos primero se conectan a una etapa de amplificación, ya que los amplificadores trabajan a 5V y la salida máxima de tensión del microprocesador es de 3,1V. La conexión realizada entre los servos y esta etapa de amplificación se da mediante un cable de STP reforzado de cuatro pares. En la Figura 19 se puede observar la estructura diseñada para almacenar la etapa de amplificación.

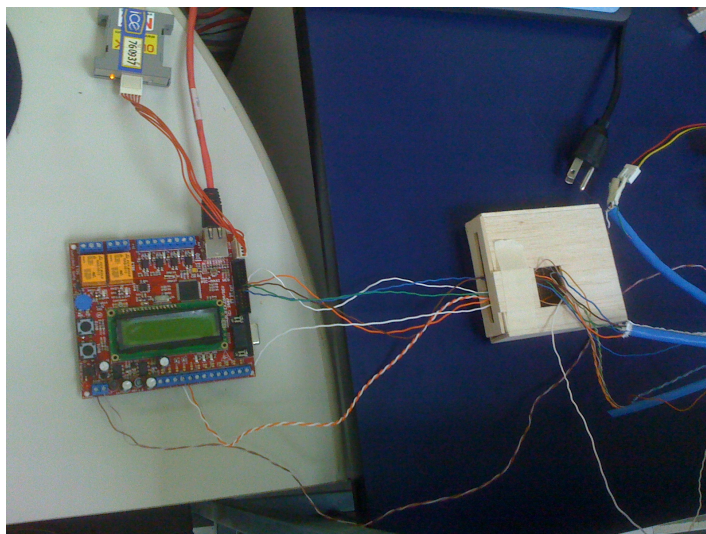


Figura 19. Detalle microprocesador y etapa de amplificación.

Para las cajas de décadas que debe manipular la planta, se crearon dos estructuras auxiliares: una de madera, de forma rectangular, que soporta la caja de décadas y le da la altura necesaria para que la pinza pueda calibrar los conmutadores sin problema alguno. La otra estructura, es un anillo metálico ajustable al diámetro de los conmutadores, el cual, a su alrededor posee unas pequeñas barras que permiten un mejor agarre y acción giratoria por parte de la pinza. Ambas estructuras, se muestran en las Figuras 20 y 21, son necesarias para el mejor funcionamiento de la planta y una mejor ejecución del proceso de calibración que se desea realizar.

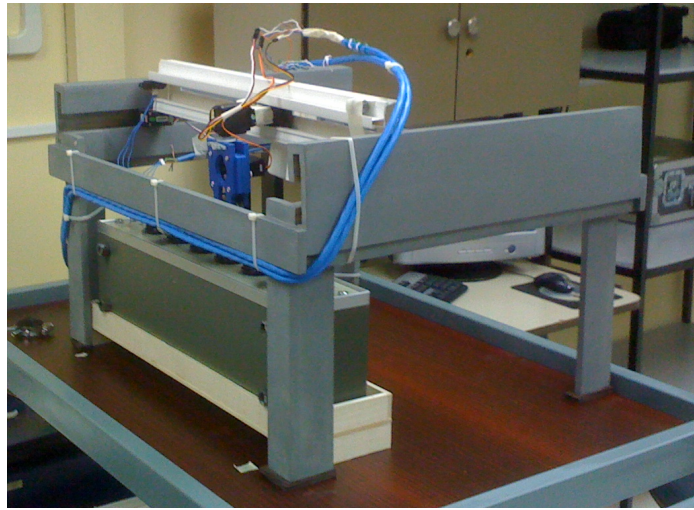


Figura 20. Y Figura 21. Estructuras auxiliares para las cajas de décadas y conmutadores.

Para la solución de este proyecto se implementó el diagrama de bloques mostrado en la Figura 22, que definen las etapas del desarrollo de la automatización del proceso de verificación y calibración del sistema. Para esto es necesario elaborar una interfaz de PC para ser utilizada en el Laboratorio, y que además se pueda acceder desde cualquier PC conectada a la red de LAMETRO.

El bloque del microprocesador Olimex Maxi Web está conformado por los módulos de Ethernet y EUSART los cuales son los encargados de la comunicación del microcontrolador con las PCs por los diferentes medios. También esta compuesto por los módulos PWM y ADC utilizados en funciones de control de los dispositivos que componen este proyecto. Además se debe incluir

el módulo LCD y LEDs el cual es utilizado para indicarle al usuario los diferentes procesos que se están llevando a cabo con las características de los mismos. La etapa de amplificación, servomotores y sensores son bloques periféricos explicados posteriormente en este informe.

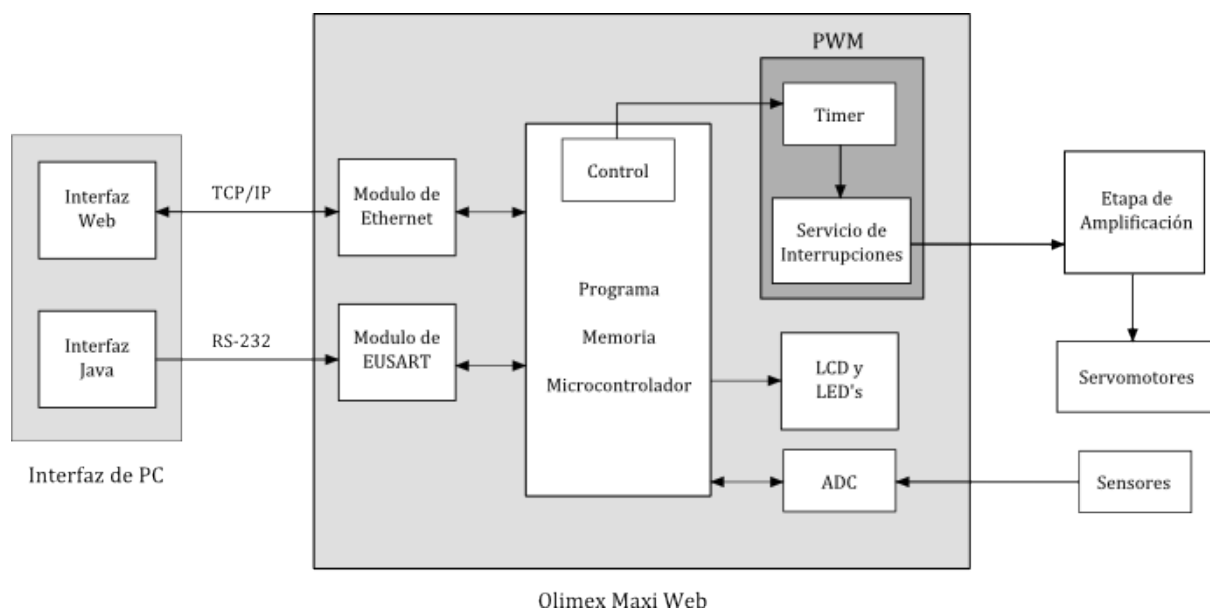


Figura 22. Diagrama de bloques propuesto para implementar el sistema.

5.2 Interface

Para el control del sistema automatizado del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor, se diseñaron y crearon dos interfaces de fácil uso, que le permiten al usuario obtener los datos necesarios y deseados de una manera rápida y eficiente, desde ubicaciones distintas.

5.2.1 Interface de PC en el laboratorio.

Una de estas interfaces puede ser accesada desde el laboratorio mismo, ya que esta se encuentra en el equipo anexo a la planta. Dicha interface fue programada en lenguaje Java y se creó un archivo ejecutable desde el escritorio de la computadora, para que el usuario del laboratorio no tenga problemas al ubicarlo, el logo de este archivo .exe se muestra en la Figura 23. Al abrir este software se despliega una interface con varias opciones; que el usuario irá

eligiendo, dependiendo de los datos que desee obtener. Tal como se muestra en la Figura 24.



Figura 23. Icono del software en el escritorio de la PC.



Figura 24. Menú selección de datos inicial.

Como primera opción, deberá ser seleccionado el equipo a utilizar, donde se despliegan las cuatro cajas de décadas disponibles para la calibración, en la Figura 25 se observa un ejemplo seleccionando la caja de décadas iET 5k.



Figura 25. Selección de equipo a utilizar.

Seguidamente se encuentra la opción para elegir el o los conmutadores que se desean calibrar, donde se despliega un menú secundario en el que se puede desde: elegir un solo conmutador de los seis existentes en cada caja de décadas (Figura 26.), todos los conmutadores (Figura 28.) y todos los conmutadores con diferente tiempo para cada uno (Figura 29.). Si esta opción es seleccionada, se desplegará otro menú, donde se podrá introducir individualmente, cada uno de los tiempos para cada conmutador realizando el proceso en cada magnitud con su respectivo tiempo. Si la opción elegida es alguna de las dos primeramente mencionadas (un conmutador individual, o todos los conmutadores a la vez), debe introducirse el tiempo requerido en el espacio llamado: tiempo de prueba de cada conmutador (minutos), (Figura 27.). Indicándole al sistema de esta manera el tiempo de prueba deseado en cada magnitud del conmutador.



Seleccione el equipo a utilizar

- iET 1k
- iET 5k
- YOKOGAWA 2793
- iET Decada 4

Conmutador : 2

Tiempo de prueba de cada conmutador : 2

Seleccione el puerto : Diferentes T.

Inicio Reset

Figura 26. Selección de conmutador individual.



Seleccione el equipo a utilizar

- iET 1k
- iET 5k
- YOKOGAWA 2793
- iET Decada 4

Conmutador : 2

Tiempo de prueba de cada conmutador (minutos) : 2

Seleccione el puerto : COM1

Inicio Reset

Figura 27. Ingresar tiempo de prueba a un conmutador seleccionado.

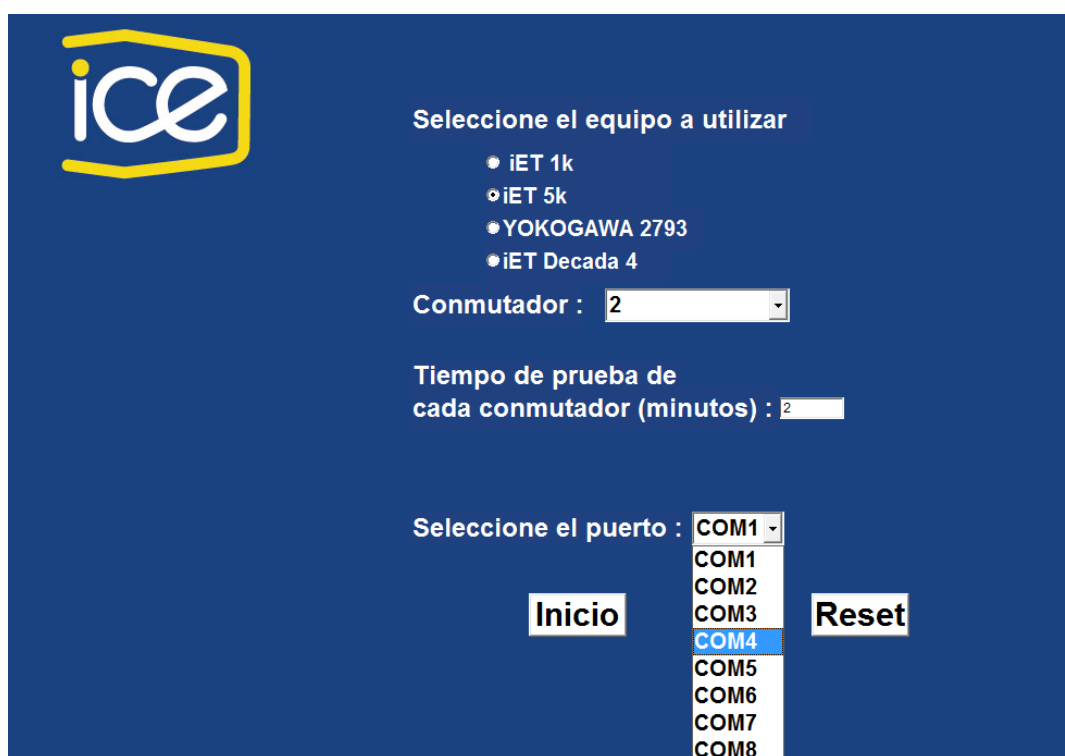
The screenshot shows the 'ICE - LAMETRO' application window. On the left is the 'ice' logo. The main area has a dark blue background with white text. It includes a section 'Seleccione el equipo a utilizar' with four radio buttons: 'iET 1k', 'iET 5k', 'YOKOGAWA 2793', and 'iET Decada 4'. Below this is a dropdown menu for 'Conmutador' set to 'Todos'. Then, there is a label 'Tiempo de prueba de cada conmutador (minutos):' followed by a text input field containing the number '0'. Further down is another dropdown menu for 'Seleccione el puerto' set to 'COM1'. At the bottom are two buttons: 'Inicio' and 'Reset'.

Figura 28. Ingresar tiempo de prueba para todos los conmutadores.

This screenshot shows the same 'ICE - LAMETRO' application window, but with different settings. The 'Conmutador' dropdown is now set to 'Diferentes T.'. The 'Tiempo cada conmutador (minutos):' section features a digital time entry interface with two rows of six boxes each. The top row of boxes contains the numbers 1, 2, 3, 4, 5, and 6. The bottom row of boxes contains the numbers 3, 1, 2, 0, 0, and 0. The 'Inicio' and 'Reset' buttons remain at the bottom.

Figura 29. Ingresar diferentes tiempos de prueba para los conmutadores.

La opción siguiente se muestra en la Figura 30., es la selección del puerto serial a utilizar, donde se muestra un submenú con las ocho opciones correspondientes. Cuando todos estos datos han sido llenados y/o seleccionados se presionará el botón de “Inicio”, el cual permitirá el arranque del proceso de calibración, o también se muestra el botón de “Reset” que permite borrar la información accesada y volver al punto de inicio de la interfaz. Si el usuario desea iniciar el proceso sin haber completado alguna de las casillas de datos el sistema no permitirá el inicio del proceso. Al seleccionar “Inicio”, aparecerán los botones con las opciones de “Detener”, el cual permite pausar el proceso sin perder dato alguno y el botón “Reanudar”, el cual permite continuar con el proceso luego de haber sido pausado o detenido.



The screenshot shows the ICE software interface with a dark blue background. In the top left corner is the ICE logo, which consists of the letters 'ice' in white lowercase font inside a yellow-outlined hexagon. The main content area contains the following elements:

- Seleccione el equipo a utilizar**: A section with four radio button options:
 - iET 1k
 - iET 5k
 - YOKOGAWA 2793
 - iET Decada 4
- Conmutador :** A dropdown menu currently showing the value '2'.
- Tiempo de prueba de cada conmutador (minutos) :** A text input field containing the number '2'.
- Seleccione el puerto :** A dropdown menu currently showing 'COM1'. A sub-menu is open, displaying a list of serial ports: COM1, COM2, COM3, COM4 (highlighted in blue), COM5, COM6, COM7, and COM8.
- Inicio**: A white button with black text.
- Reset**: A white button with black text.

Figura 30. Selección de puerto serial a utilizar.

5.3 Interface Web

La otra interfaz construida, consta de una que fue diseñada para su uso desde cualquier parte del edificio del Laboratorio LAMETRO, por medio de internet. Esta interface puede ser utilizada por cualquier equipo conectado a la red del laboratorio y por usuarios portadores de una contraseña, ya que esta interfaz consta de un sistema que funciona y permite el acceso únicamente mediante un código de seguridad, tal como se muestra en la Figura 31.

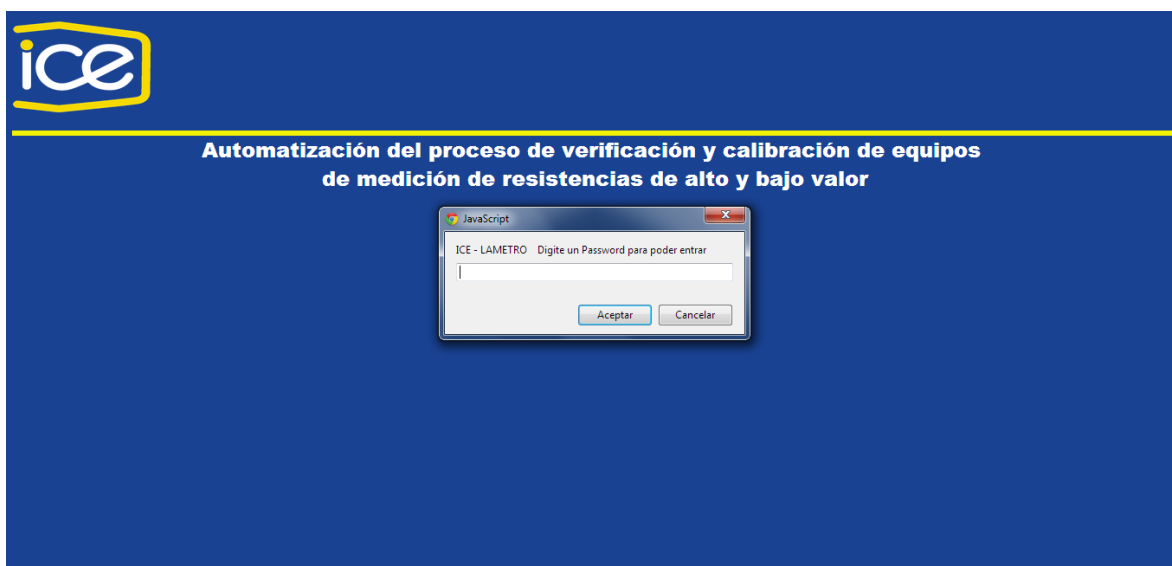


Figura 31. Acceso a la interface vía Ethernet.

Este sistema de seguridad por contraseña le fue implementado a esta interfaz ya que el hecho de que se encuentre en la red, la pone en peligro de ser manipulada por cualquier particular. En cuanto a la información y datos que se deben introducir o que se visualizan en esta interface, la única diferencia con respecto a la que se encuentra anexada a la planta, es que esta interfaz para uso en la red, mantiene un control de la temperatura interior del laboratorio, que debe permanecer en una temperatura ideal de trabajo de 23 °C, esto debido a que al estar lejos del ambiente del laboratorio, el operario no cuenta con la facilidad poder monitorear de manera directa los termómetros que se encuentran dentro de la instalación. Si la temperatura ideal del laboratorio variara de manera que se pusiera en riesgo el proceso y/o equipo, el sistema emite una alerta que pone al

usuario al tanto de manera inmediata. Además le permite al usuario activar el uso de la interface diseñada para la computadora que se encuentra continua a la planta, únicamente desde esta página se puede permitir esta función.

The screenshot shows a web interface titled "LAMETRO" with the subtitle "Automatización del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor". The interface includes the following elements:

- Logo:** "ice" logo in the top left corner.
- Form Fields:**
 - "Seleccione el equipo a utilizar : IET 1k" (dropdown menu)
 - "Seleccione conmutador : 1" (dropdown menu)
 - "Tiempo de prueba de cada conmutador (minutos):" (text input field)
 - "Tiempo diferente para cada conmutador (minutos) : 1 2 3 4 5 6" (six text input fields for individual times)
- Buttons:**
 - "Inicio" (Start)
 - "Reset"
 - "Detener" (Stop)
 - "Reanudar" (Resume)
 - "Usar puerto Com" (Use COM port)

Figura 32. Acceso a la interface vía Ethernet.

5.3.1 Codificación de datos

Para enviar las órdenes desde el programa al control se utilizó una codificación formada por dos partes:

- Un opcode: Se utiliza una letra mayúscula por cada opcode, y hay de dos tipos, las que representan una orden y las que representan un envío posterior de valores numéricos.
- Un valor numérico: Solo presente en los casos necesarios, representa una cantidad que puede indicar el tipo de máquina, el número de conmutador o la cantidad de tiempo de cada conmutador.

Este protocolo se explica en la tabla 3.

Tabla 3. Estructura y comportamiento de envío de órdenes.

Botón	Información a enviar	Valor en el puerto
Inicio	Tipo de máquina	Opcode A y número de la máquina
	Número de Conmutador	Opcode B y valor de conmutador
	Cantidad de Tiempo	Opcode C y valor de tiempo (esta trama de tiempo se puede enviar en varios caracteres por el puerto dependiendo del número digitado)
	Orden de Inicio	Opcode D
Pausa	Orden de Pausa	Opcode E
Reset	Orden de Reset	Opcode F
Reanudar	Orden de Reanudar	Opcode H
Usar puerto COM	Orden de Atender al puerto serial*	Opcode L

*Esta opción solo esta disponible en la interfaz Web.

La información de cantidad de tiempo varía si el número de conmutador es 8, lo q significa que se esta procesando un tiempo diferente para cada conmutador, por lo que la trama envía los valores de tiempo separados por un carácter especial, para el caso de envío por Ethernet corresponde a “N” y para el caso de envío por puerto serial corresponde a “%”. A continuación se presentan algunos ejemplos de envío de datos al presionar inicio.

Máquina 1, conmutador 2, tiempo 3 minutos. Dato enviado: A1B2C3D

Máquina 2, tiempos diferentes, tiempos en minutos 3, 5, 2, 6, 10, 1. Dato enviado: A2B8C3%5%2%6%10%1%D (puerto serie), A2B8C3N5N2N6N10N1ND (Ethernet).

5.3.2 Interfaz en Java con comunicación por puerto serial.

El programa presenta una interfaz para escoger las diferentes opciones de los procedimientos del control:

- El número de puerto serial al cual está asociado el sistema.
- El equipo a utilizar.
- La escogencia del conmutador.
- El tiempo de prueba para cada conmutador (ya sea que el tiempo sea el mismo para cada conmutador o que este varíe según el conmutador).
- Un grupo de botones los cuales dan la función de Inicio, Detener, Reanudar, y Reset.

5.1.2.1 Condiciones de seguridad y uso de la interfaz:

Primeramente hay que destacar que las instrucciones enviadas por el programa no son atendidas a menos que en la interfaz Web se le notifique al control para que atienda al puerto serie.

Luego hay condiciones necesarias para evitar un mal uso de la interfaz o el envío incorrecto de datos, estas radican en:

- Para que se dé el envío de datos al presionar el botón de inicio, el usuario debe llenar todos los campos de la interfaz, además debe seleccionar correctamente el puerto serial de conexión del sistema. Si el usuario no cumple con estas condiciones se presenta un mensaje específico para cada error.
- Los botones Detener y Reanudar no parecen hasta después de presionar el botón de inicio y se haya dado un correcto envío de datos, además los botones solo se habilitan uno a la vez; esto le da coherencia a los eventos que se presentan en el control y las opciones permitidas en la interfaz.

5.3.3 Comunicación del puerto serial en Java

Para la comunicación con el puerto serial se procedió a utilizar la librería gratuita “GiovynetDriver.jar” disponible en el sitio <http://www.giovynet.com/>.

Para utilizar el puerto serial se necesita extraer el paquete .jar en la misma carpeta donde se está compilando el programa de java, además de los archivos “libSerialPort.dll” y “libSOSerialPort.so” (también disponibles en el sitio web), estos archivos y librerías son los encargados de dar soporte a las funciones del puerto serial solicitadas en el código del programa. Luego en el encabezado del programa se debe incluir las librerías necesarias como se muestra a continuación:

```
import giovynet.nativelink.SerialPort;
import giovynet.serial.Baud;
import giovynet.serial.Com;
import giovynet.serial.Parameters;
```

El código del programa utiliza estas librerías en dos funciones:

- Función “EstadoPuerto()”: verifica el estado del puerto escogido por el usuario y muestra un mensaje de error por la escogencia incorrecta.
- Función “EscribirPuerto(int action)”: primeramente configura el puerto y lo abre, luego dependiendo del valor de “action” envía los datos según el botón presionado, la información enviada se explica en la sección de Codificación de Datos, y por último cierra el puerto serial.

A continuación se muestra una sección de código donde se configura el puerto serial, se abre y luego se envía un dato dependiendo del botón presionado:

```
String opcodeA = "A"; //String con el opdeco A
    /***Configuración del puerto.***/
Parameters parameters = new Parameters();
parameters.setPort(PuertoSelec);
parameters.setBaudRate(Baud._19200); //Configuración del puerto a 19200
baudios
    /***Apertura del puerto.***/
Com com = new Com(parameters);
    /***Envío de datos ***/
if (action == 1) { //La acción es inicio
    com.sendSingleData(opcodeA); //Instrucción que 53nvía el opcode A
por el puerto
```

El proceso para obtener un fichero ejecutable a partir del código fuente se explica en la sección de anexos. Además para la correcta ejecución del programa

ejecutable es necesario tener los ficheros “libSerialPort.dll” y “libSOSerialPort.so” en el mismo directorio donde se encuentra el ejecutable.

5.4 Unidad de control y comunicación

Esta constituida físicamente por la kitt de desarrollo Olimex PIC-MAXI-WEB, esta tarjeta utiliza el microprocesador PIC18F97J60, en la referencia 2 se presenta un manual donde se puede encontrar todas las especificaciones de este hardware. Este kitt se programo utilizando lenguaje y librerías C18, en el paquete de software MPLAB IDE, y la utilización del PICKit 3. En la referencia 3 se presenta un documento de Microchip con una compilación de las librerías presentes en C18.

Para el sistema se utilizaron diferentes módulos de la tarjeta de desarrollo, de los cuales se explican a continuación la función que desempeñan, los parámetros de inicialización, las librerías C18 asociadas, y algunos bloques de código importantes.

5.4.1 Módulo EUSART

Se encarga de la comunicación RS-232 por un conector DB-9, administra los buffer de recepción y transmisión de datos, esta configurado a 19200 Baudios, 1 bit de inicio, 1 bit de stop, sin paridad y sin generación de interrupciones. Para su uso se invoca las librerías “UART.C” y “UART.H” que contiene las funciones para lectura y escritura en el puerto serie, de las cuales se utiliza las funciones “DataRdyUART()” y “ReadUART()” que se explican más adelante.

En el uso del sistema, este módulo solo desempeña la función de recepción de datos, para lo cual se utiliza el siguiente bloque de código:

```
while(!DataRdyUART());  
    ArregloChart [CuentaBuffer] = ReadUART();
```

La función “DataRdyUART()” devuelve la información de la presencia de datos en el buffer de recepción, así el bucle se estará ejecutando hasta que el buffer de datos se vacíe. La función “ReadUART()” saca un dato del buffer y lo devuelve como resultado, de esta forma se almacena el dato recibido en un arreglo de caracteres. Cabe destacar que este sistema maneja la recepción de datos por software no por interrupciones, lo que permite un control en el flujo del programa para leer los datos solo cuando sea necesario.

Los datos recibidos son luego procesados para su interpretación en el resto del programa.

5.4.2 Módulos de Timer

Proporcionan una base de tiempo variable que depende de los parámetros de inicialización del módulo, el microcontrolador tiene 4 módulos de timer, de los cuales se utilizan:

- Timer 1: Este timer se utiliza en la librería “Tick.c” y “Tick.h” de C18, que permite tener una base de tiempo en intervalos mínimos de 1 s.
- Timer 2 y timer 4: Se utilizan para generar el PWM de control de los servomotores, se configuran para generar el mejor intervalo de tiempo necesario según el ciclo de trabajo, esto se explica mejor en la sección de Generación de PWM.

La librería “Tick.c” permite generar eventos de tiempo múltiplos de 1 segundo, así por ejemplo el siguiente código se utilizó para generar el conteo de tiempo de cada conmutador.

```
TickInit();           //Inicia del Timer1
StartTime = TickGet(); //Obtiene el valor actual del Timer1 y lo almacena
                      //en StartTime
while (n <= TiempoComutador) {
    if (TickGet() - StartTime >= 60*TICK_SECOND){
        n++;
    }
}
```

La constante “TICK_SECOND” está definida en la librería “Tick.h” y representa el valor del timer correspondiente a un segundo, así cuando el tiempo transcurrido “TickGet() - StartTime” sea mayor o igual a “60*TICK_SECOND” se habrá cumplido un minuto, la variable “n” aumenta en uno con lo que luego el bucle while termina cuando se cumplan los minutos correspondientes al tiempo asignado para el conmutador.

Para el control de los timer 2 y 4 se utiliza las librerías “TXopen.c” y “TXclose.c” donde X corresponde al número del timer, estas librerías contienen las funciones “OpenTimerX” que permite escoger los parámetros para la base de tiempo y la generación o no de interrupciones y luego inmediatamente inicia a correr el timer, la función “CloseTimerX” detiene el timer. En el siguiente código se presenta un ejemplo de la utilización de “OpenTimerX”

```

if (adelanteEjeX==0){
    OpenTimer4(TIMER_INT_ON & T4_PS_1_16 & T4_POST_1_13);
} else {
    OpenTimer4(TIMER_INT_ON & T2_PS_1_4 & T2_POST_1_14);
}

```

En este código se escoge si iniciar el timer en ciertos valores para obtener una diferente base de tiempo, el parámetro de entrada “TIMER_INT_ON” activa las interrupciones de timer, los otros dos parámetros corresponden al registro de preescala y de postescala del timer, todos los valores posibles de entrada se presentan en la referencia 3.

5.4.3 Módulo de ADC

Hace la conversión para las señales de los sensores de la entrada analógica a una representación digital. Utiliza los canales analógicos 7 y 8 que corresponden a los conectores X4 y X5 de la tarjeta de desarrollo para recibir la señal analógica. Para su control se usa las librerías “adcopen.c”, “adcclose.c”, “adcconv.c”, “adcbusy.c”, “adcread.c” y “adc.h”, que contienen las funciones necesarias para controlar el ADC, a continuación se presenta un ejemplo de código donde se hace la conversión de la señal de un sensor.

```

CloseADC();
OpenADC(ADC_FOSC_64 & ADC_RIGHT_JUST & ADC_12_TAD, ADC_CH7 &
ADC_INT_OFF & ADC_REF_VDD_VSS, 0);
ConvertADC();
while(BusyADC());
    resultado_s1 = ReadADC();
CloseADC();

```

Iniciando se llama a la función “CloseADC()” como medida de seguridad detener cualquier uso del ADC por parte de otros procesos, luego se procede a configurar el ADC con la función “OpenADC(parametros)” los parámetros se encuentran declarados en “adc.h” y todos se explican en la referencia 3, las configuraciones se presenta a continuación:

- “ADC_FOSC_64” genera un tiempo de conversión más lento posible.
- “ADC_RIGHT_JUST” se utilizar justificación derecha.
- “ADC_12_TAD” para promediar el ruido de la señal de los sensores se utiliza el mayor tiempo de adquisición.
- “ADC_CH7” se escoge el canal 7 como fuente de la señal analógica ha convertir.
- “ADC_INT_OFF” no se utilizan la interrupciones del ADC ya que el programa solicita el dato convertido en forma explicita.

- “ADC_REF_VDD_VSS” los sensores están alimentados con 5 V al igual que el kitt de desarrollo, además las tierras del sistema están todas unidas, por lo que corresponde utilizar para el ADC las mismas señales de 5 V y tierra, lo que se configura con este parámetro.

Luego el código llama a la función “ConvertADC()” que inicia la adquisición y conversión de datos; el posterior bucle while tiene la función de esperar a que la conversión termine y así luego pedir el resultado del ADC, para esto se utiliza la función “BusyADC()” que devuelve el estado de conversión del ADC, y función “ReadADC()” que devuelve el valor de la conversión.

5.4.4 Módulo del LCD

Permite el despliegue de letras en una pantalla de 32 caracteres en dos líneas, y se configura de forma que se pueda utilizar los 32 caracteres. Se controla mediante las librerías “LCDBlocking.c” y “LCDBlocking.h” que permiten iniciar y mostrar los datos en el LCD. El siguiente segmento de código se utiliza para mostrar un mensaje en el LCD.

```
LCDInit();
strcpypgm2ram((char*)LCDText, "Inicio");
LCDUpdate();
```

Primeramente se inicia el LCD mediante “LCDInit()” lo cual configura los pines de los puertos del microcontrolador asociados al LCD, y envía las secuencias que el LCD necesita para inicializar, esta función solo es necesaria ser llamada una vez en todo el programa, ya que el bloque de LCD solo requiere ser iniciado una vez. La instrucción “strcpypgm2ram((char*)LCDText, "Inicio")” copia en “LCDText” la palabra “Inicio” que es el mensaje a mostrar, para que luego la función “LCDUpdate()” utilice el valor presente en “LCDText” y lo muestre en la pantalla del LCD.

El kitt de desarrollo dispone de un LED para iluminar la pantalla LCD, este LED es utilizado en ocasiones para indicar un estado del sistema, así se utiliza de la forma:

- Cuando el sistema inicia el proceso la pantalla LCD se ilumina, al finalizar se apaga la pantalla.
- Cuando se está en pausa la luz de la pantalla se enciende intermitentemente, en intervalos de medio segundo, hasta que el sistema reanuda el proceso.

5.4.5 Sistema de PWM

El control de los servomotores se lleva a cabo mediante una señal de PWM, sin embargo para el correcto funcionamiento de los servomotores es necesario un periodo de la señal mayor a 20 ms, este periodo no puede ser generado por el módulo de PWM presente en el PIC18F97J60 que tiene el kitt de desarrollo.

Mediante el siguiente calculo se obtiene el mayor periodo posible del módulo de PWM^[4],

$$\begin{aligned}\text{PWM Period} &= [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2 \text{ Prescale Value}) \\ \text{PWM Period} &= [(255) + 1] \cdot 4 \cdot (1/41\text{MHz}) \cdot (16) \\ \text{PWM Period} &= 399.6 \text{ us}\end{aligned}$$

Para solucionar esto se recurre a implementar una base de tiempo variable mediante timer y con la utilización del servicio de interrupciones se hace cambiar el valor de un pin de salida cada cierta cantidad de ciclos del timer, esto se muestra en la Figura 33 y 34.

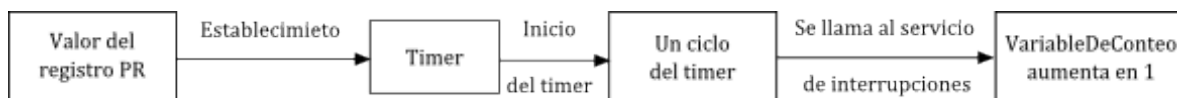


Figura 33. Flujo del proceso para generar la señal de PWM.

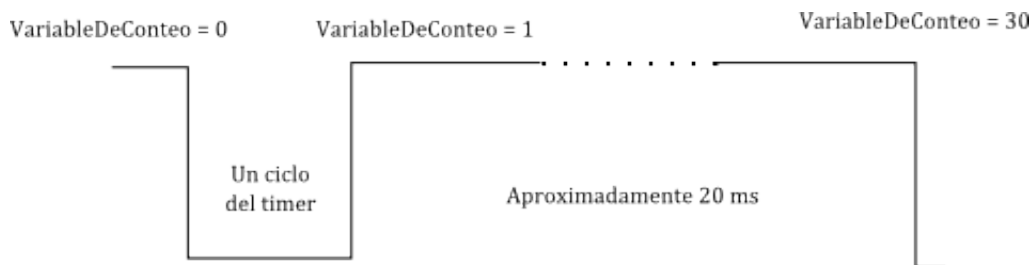


Figura 34. Diagrama de tiempos de la señal de PWM generada con el método de timer.

El valor del registro PR se establece en el Timer para variar al ciclo de tiempo del timer, cada ciclo llama al servicio de interrupciones el cual contiene una variable de conteo, dependiendo de esta variable se cambia el valor de un pin de salida del kitt de desarrollo.

El control del ciclo de tiempo se lleva a cabo por el establecimiento de un valor en el registro PRX del timerX, así mediante este registro, la preescala y la postescalar del timer se puede generar una variación del ciclo de trabajo del PWM adecuada para controlar los servomotores.

En la Figura 34 se debe destacar que esta señal es la invertida de la necesaria para controlar los servomotores, esto se debe a una etapa de amplificación que se explica en la sección: Amplificación de la señal de PWM.

A continuación se muestra una sección de código del servicio de interrupciones, correspondiente al PWM generado para el pin donde se conecta el servomotor de movimiento del bloque de la pinza.

```

    If (select_servo==2){           //Servomotor para el movimiento horizontal
        If (PIR3bits.TMR4IF){      //Se comprueba la fuente de la
interrupción, el timer 4
            PIR3bits.TMR4IF = 0;
            TiempoPWMT4 = TiempoPWMT4+1;
            if (TiempoPWMT4 <= 1){
                LATG = LATG & 0b11111110; //poner 0
            }
            if (TiempoPWMT4>1 && TiempoPWMT4<=36){
                LATG = LATG | 0b00000001; //poner 1
            }
            if (TiempoPWMT4>36){
                TiempoPWMT4=0;
            }
        }
    }

```

El primer if comprueba cual servomotor que se necesita controlar. El segundo if verifica que la fuente de interrupción es un timer. La siguiente instrucción reinicia la bandera del timer para evitar un nuevo llamado no solicitado de la interrupción. Los siguientes if se encargan de generar la señal de la Figura 34 mediante la revisión del valor de la variable de contero “TiempoPWMT4”. Las instrucciones “LATG = LATG & 0b11111110” y “LATG = LATG | 0b00000001”, son las encargadas de cambiar el valor del pin de salida del kitt de desarrollo.

5.4.6 Módulo de Ethernet

Encargado de la comunicación por red mediante el protocolo TCP/IP. Para controlarlo y configurarlo se utiliza las librerías proporcionadas por Microchip y adaptadas para el uso de esta tarjeta de desarrollo.

5.4.6.1 Protocolo de transferencia por Ethernet

La herramienta proporcionada por microchip llamada TCPIP Stack proporciona funciones de alto nivel en las librerías para el manejo del Ethernet, en la referencia 5 se encuentra la documentación dada por Microchip, así esta herramienta proporciona soporte para:

- Establecer la conexión con router o computadoras.
- Administrar los buffer de entrada y salida.
- La utilización de funciones como DHCP cliente o servidor, FTP, HTTP, TCP, otras. La Figura 35 muestra como el TCPIP Stack asocia las diferentes capas a las funciones que soporta.

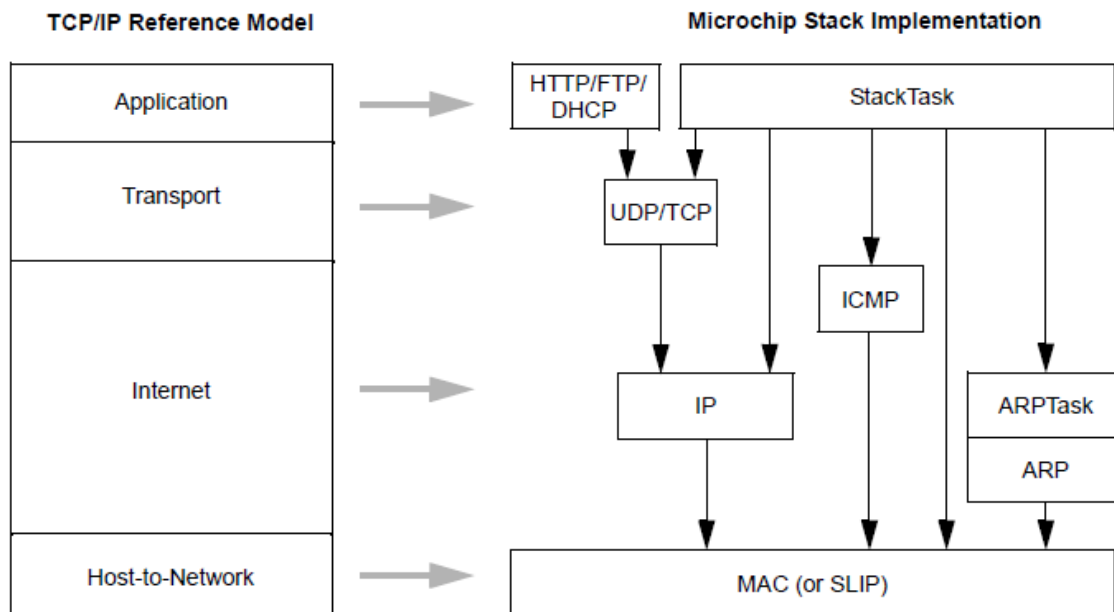


Figura 35. Diagrama de la asociación de las funciones de TCPIP Stack [5].

El fichero "TCPIPConfig.h" contiene los valores para configurar cuales sistemas se utilizaran en el programa, para este sistema se hace uso del HTTP Server y el MPFS, que se explican más adelante.

- Asignación de valores de red como la IP, la mascara de subred, la dirección MAC y DNS.

La configuración más adecuada para este trabajo corresponde a la utilización de una IP y mascara de subred estática, esto debido al tipo de red a la que se va a conectar la Ethernet, ya que el sistema de red solo presenta libres ciertas direcciones de red en una mascara de subred determinada, para lograr esto se debe deshabilitar las funciones de DHCP y asignar directamente en el programa los valores de IP y mascara de subred sobre las variables que para ello utiliza las librerías de Microchip, como lo presenta el siguiente código:

```
AppConfig.MyIPAddr.Val = (10ul) | (4ul)<<8ul | (24ul)<<16ul | (29ul)<<24ul;

AppConfig.MyMask.Val = (255ul) | (255ul)<<8ul | (252ul)<<16ul |
(0ul)<<24ul;
```

5.4.6.2 Utilización de las funciones HTTP Server y MPFS

La presentación de la interfaz web se alcanzó mediante estas dos funciones, así el módulo de HTTP Server permite la atención de peticiones de red para envío y recepción de datos, y el módulo MPFS o sistema de archivos de microchip administra todos los archivos que utiliza la pagina web.

Otra función que permite el HTTP Server es la ejecución de acciones, ya que la información que recibe se puede interpretar como comandos lo que se aprovechó como medio para contener las instrucciones que el sistema debe operar. También esta función permite el envío de variables del sistema a la interfaz web. En la siguiente sección de código se reciben los datos enviados por el Ethernet al presionar los diferentes botones de la interfaz web.

```
case CGI_CMD_LCDOUT: // ACTION=1
    if (argc > 2u) { // Text provided in argv[2]
        UnencodeURL(argv[2]); // Convert %20 to spaces, and other URL
        transformations
        if (strlen((char*)argv[2]) < 32u) {
            strcpy((char*)X_VICTOR, (char*)argv[2]);
        }
    }
}
```

El case verifica si se esta enviando datos. Entre los dos if aseguran que el argumento se encuentra correcto, y la función “strcpy((char*)X_VICTOR, (char*)argv[2]);” copia el argumento en el arreglo de string que se utiliza para verificar las instrucciones.

5.4.7 Control de la planta

El sistema a controlar esta formado por 5 servomotores los cuales tienen la función de ubicar la pinza para poder realizar el cambio de magnitud en los conmutadores de la caja de décadas, siendo estos los dispositivos a modelar. Los dos sensores de proximidad infrarrojos tienen la función de ubicar la posición de la pinza en todo momento. En la Figura 36 se muestra el diagrama de bloques de la configuración de cada servomotor.

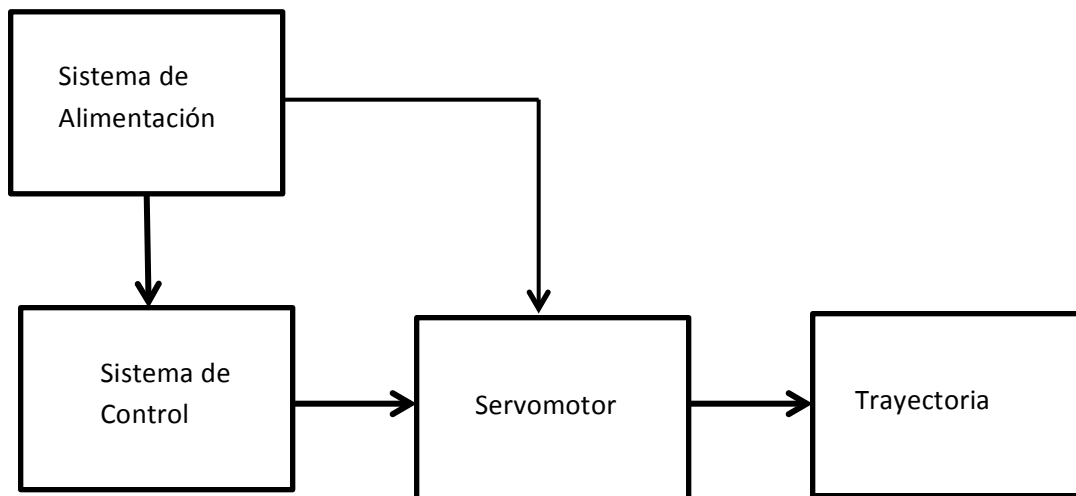


Figura 36. Diagrama de flujo servomotor.

El control de los servomotores se realizó por medio del modelado empírico al analizar las mediciones de las respuestas de los dispositivos. Estas mediciones se obtuvieron programando en el microprocesador una rutina en la cual le aplicara una entrada escalón que representaba un movimiento de 45° al servomotor.

Se utilizó un osciloscopio digital, el cual toma datos cada milisegundo, permitiendo hacer una tabla para tener acceso desde una computadora, se abrió la estructura del servomotor y se midió la salida del potenciómetro el cual internamente es utilizado como un transductor de posición. Por medio de este análisis se obtuvieron los datos de la Tabla 4.

Tabla 4.Respuestas obtenidas del servomotor.

Tiempo [s]	Salida	Entrada	Tiempo [s]	Salida	Entrada	Tiempo [s]	Salida	Entrada
0	1.1	-0.65281	0.06	1.2	-0.65281	0.122	1.26	-0.65281
0.002	1.18	-0.65281	0.062	1.2	-0.65281	0.124	1.26	-0.65281
0.004	1.16	-0.65281	0.064	1.18	-0.65281	0.126	1.22	-0.65281
0.006	1.18	-0.65281	0.066	1.22	-0.65281	0.128	1.26	-0.65281
0.008	1.16	-0.65281	0.068	1.18	-0.65281	0.13	1.22	-0.65281
0.01	1.18	-0.65281	0.07	1.18	-0.65281	0.132	1.22	-0.65281
0.012	1.18	-0.65281	0.072	1.16	-0.65281	0.134	1.24	-0.65281
0.014	1.18	-0.65281	0.074	1.18	-0.65281	0.136	1.26	-0.65281
0.016	1.22	-0.65281	0.076	1.18	-0.65281	0.138	1.22	-0.65281
0.018	1.26	-0.65281	0.078	1.16	-0.65281	0.14	1.26	-0.65281
0.02	1.26	-0.65281	0.08	1.16	-0.65281	0.142	1.30	0.65281
0.022	1.2	-0.65281	0.082	1.18	-0.65281	0.144	1.32	0.65281
0.024	1.18	-0.65281	0.084	1.26	-0.65281	0.146	1.30	0.65281
0.026	1.18	-0.65281	0.086	1.24	-0.65281	0.148	1.42	0.65281
0.028	1.16	-0.65281	0.088	1.26	-0.65281	0.15	1.64	0.65281
0.03	1.2	-0.65281	0.09	1.25	-0.65281	0.152	1.98	0.65281
0.032	1.22	-0.65281	0.092	1.26	-0.65281	0.154	1.97	0.65281
0.034	1.18	-0.65281	0.094	1.26	-0.65281	0.156	1.99	0.65281
0.036	1.16	-0.65281	0.096	1.22	-0.65281	0.158	2.00	0.65281
0.038	1.18	-0.65281	0.098	1.24	-0.65281	0.16	1.98	0.65281
0.04	1.18	-0.65281	0.1	1.26	-0.65281	0.162	1.98	0.65281
0.042	1.22	-0.65281	0.102	1.18	-0.65281	0.164	2.05	0.65281
0.044	1.22	-0.65281	0.104	1.16	-0.65281	0.166	1.96	0.65281
0.046	1.26	-0.65281	0.106	1.18	-0.65281	0.168	1.99	0.65281
0.048	1.22	-0.65281	0.108	1.22	-0.65281	0.17	1.98	0.65281
0.05	1.18	-0.65281	0.11	1.2	-0.65281	0.172	1.97	0.65281
0.052	1.22	-0.65281	0.112	1.22	-0.65281	0.174	1.95	0.65281
0.054	1.22	-0.65281	0.114	1.18	-0.65281	0.176	2.03	0.65281
0.056	1.16	-0.65281	0.116	1.22	-0.65281	0.178	1.98	0.65281
0.058	1.24	-0.65281	0.118	1.2	-0.65281	0.18	2.1	0.65281

Con los datos obtenidos se hizo uso del software matemático Matlab para poder identificar el sistema y comprobar los resultados obtenidos. Para esto se utilizó la función `ident`, herramienta de identificación de sistemas la cual construye modelos matemáticos de sistemas dinámicos de medición de entrada y salida de datos. Se puede utilizar en el dominio del tiempo y dominio de la frecuencia de datos para identificar las funciones de transferencia de tiempo continuo y discreto. En la Figura 37 se muestra el análisis de los datos medidos con esta herramienta.

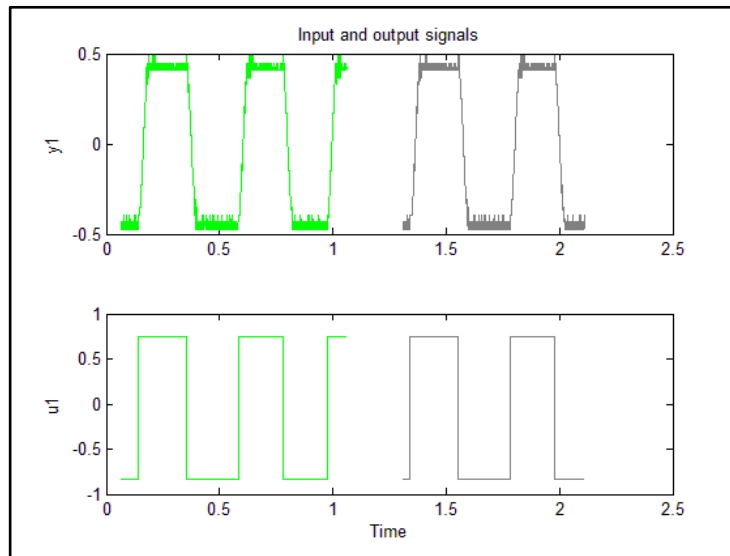


Figura 37. Posición de salida del sistema y escalón de entrada del sistema.

Con los datos obtenidos en la figura anterior se obtuvo una identificación del sistema dando una aproximación del 88.92%, la cual es aceptable para el modelo. En la Figura 38 se muestra dicha aproximación con la comparación entre la salida obtenida mediante la simulación del modelo empírico y los datos medidos experimentalmente. Con la ayuda de Matlab se procedió a calcular el modelo de la planta requerida para el sistema, seleccionando un modelo de segundo orden sin retraso, de esta manera se obtiene el modelo representado en la ecuación 4.

$$G_v(s) = \frac{8590}{(s+102)^2} \quad (4)$$

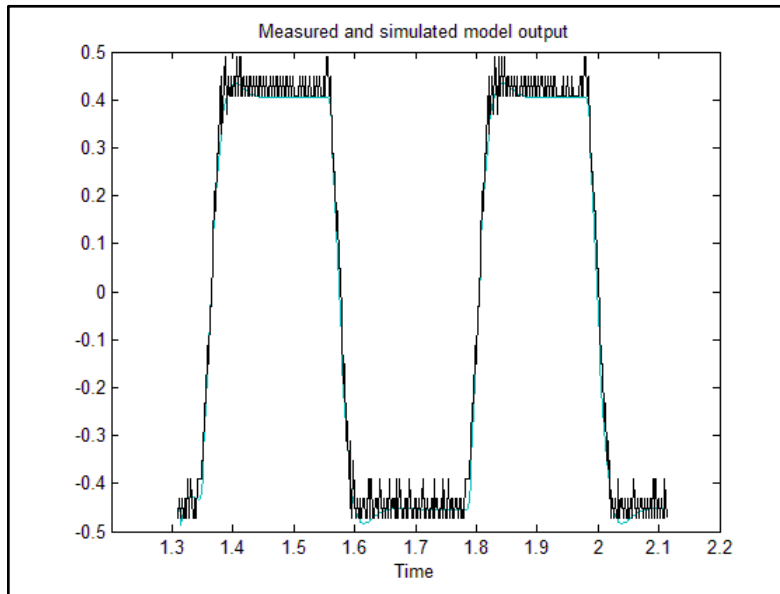


Figura 38. Señal de salida simulada contra datos medidos experimentalmente.

Para obtener el comportamiento óptimo del sistema compuesto por los servomotores se debe realizar el control de dichos dispositivos a partir del modelado obtenido en la ecuación 4, para esto se utilizó la herramienta rlocus de Matlab. Para el control se buscaba disminuir el error de estado estacionario así como controlar el tiempo de estabilización y el sobre impulso, por lo que se propuso utilizar un control PI.

Debido a la precisión que se requiere en la colocación de la pinza sobre los conmutadores se necesita que el error en estado estacionario sea cero y para la estabilidad de esta planta es necesario un sistema BIBO estable, el cual cumple esta condición si ante entradas de valor finito la respuesta no tiende a infinito.

Dado que entre las especificaciones técnicas del servomotor se encuentra que el tiempo para recorrer 1 rad es de 200 ms, se tomó este valor para estimar el tiempo de estabilización del sistema y considerando limitaciones de la planta se determinó que dicho tiempo debería ser superior a los 215 ms y menor a los 350 ms. El sobre impulso del sistema se determinó entre 1% y 5%. Estos valores se resumen en la Tabla 4.

Tabla 5. Condiciones para el sistema de control

Limitantes	Valor
Sobre impulso (M_p)	$1\% \leq M_p \leq 5\%$
Tiempo de estabilización (t_s)	$215 \text{ ms} \leq t_s \leq 350 \text{ ms}$
Error de estado estacionario (e_{ss})	$e_{ss} = 0$

Utilizando la función rlocus se introducen las especificaciones de la Tabla 4 y se ajusto la ganancia para obtener los polos del sistema dentro de la región desea. Obteniendo de esta manera la función de transferencia del control de la planta, la cual se muestra en la ecuación 5.

$$C(s) = \frac{0,0204(s+138,3)}{s} \quad (5)$$

Con esta herramienta de Matlab se puede obtener el lugar de las raíces a partir de su función de transferencia así como mostrar la posición de los polos y los ceros del controlador, en la Figura 39 se muestran dichas características del control de la planta diseñado en este proyecto.

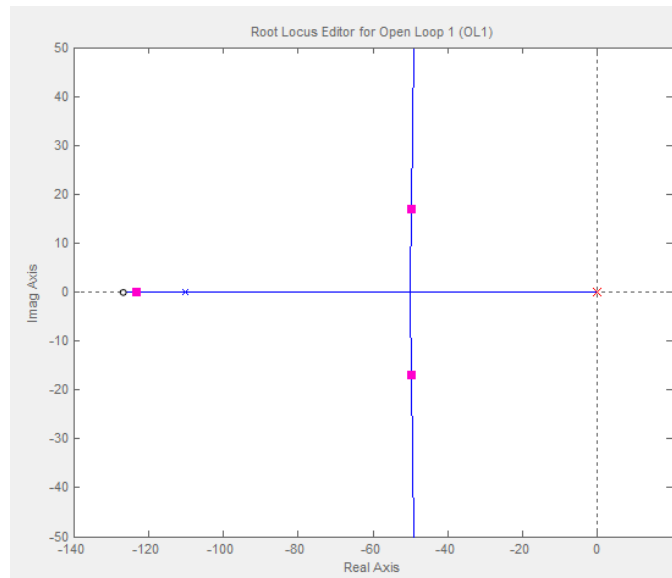


Figura 39. Diagrama de polos y ceros del sistema.

Para implementar esta función en una computadora es necesario discretizar el control obtenido anteriormente y de esta manera poderlo representar en sistemas discreto, para esto se utilizó el método de Tustin. Este método consiste en una aproximación de derivada como en una diferencia de adelanto (método de Euler)

$$px = \frac{dx(t)}{dt} \approx \frac{x(t+T) - x(t)}{T} = \frac{q-1}{qT} x(t) \quad (6)$$

Como una diferencia hacia atrás:

$$px = \frac{dx(t)}{dt} \approx \frac{x(t) - x(t-T)}{T} = \frac{q-1}{qT} x(t) \quad (7)$$

En transformadas significa remplazar:

$$s = \frac{z-1}{T} = \frac{z-1}{zT} \quad (8)$$

El cual corresponde a un desarrollo en serie truncado para el método de Euler

$$z = e^{sT} \approx 1 + sT \quad (9)$$

Para la discretización del sistema se utilizó la herramienta c2d de Matlab, cuya función es convertir un sistema continuo a un sistema discreto utilizando la operación de retención de orden cero. Se obtuvo la siguiente ecuación:

$$C(z) = \frac{0,0857*(z+0,0989)}{(z-1)} \quad (10)$$

Para implementar este sistema en el control programado en el microprocesador se debe obtener la ecuación de diferencia a partir de la función discretizada en la ecuación 10, la cual se muestra a continuación en la ecuación 11. Donde:

$$C(z) = \frac{Y(z)}{X(z)} \quad (11)$$

Por lo tanto se tiene como función final:

$$Y(z) = Y(z - 1) + 0,0857X(z) + 0,0428X(z - 1) \quad (12)$$

Se realizó un diseño Anti-Windup para evitar que el los servomotores continúen sumando el valor interno del control aunque el sistema físico no lo permita, ocasionando que el movimiento de los servomotores presente un tiempo de retraso para estabilizarse.

Para este diseño se utilizó un saturador, el cual fue implementado en el software del controlador del sistema mediante estructuras de control if – then-else. Obteniendo con esto que mientras los valores a la salida del controlador se mantuvieran dentro de los límites físicos de la planta el controlador continuara sumando, o restando según fuera el caso su posición. Si por otro lado, los valores de salida del controlador exceden los límites físicos de la planta se inicia la función para mantener los servomotores en el ámbito físico de trabajo de la planta.

5.4.8 Flujo del programa

En la figura 22 se muestra el proceso del programa del microcontrolador, a continuación se explican los bloques de procesos.

1. Inicio de módulos y variables: En esta sección se hace las inicialización de:
 - Registro de configuración: se configura el oscilador para que utilice el HSPLL y así obtener un reloj interno de 41 MHz, el watchdog timer se apaga, se deshabilita el Power On Reset, y se habilita las interrupciones globales de alta y baja prioridad.
 - Configuración de los puertos del microcontrolador: Se configuran los registros TrisX y PortX, para quedar correctamente conectados a los diferentes módulos existentes en el kit de desarrollo, se escoge cuales con señales digitales o analógicas y cuales pines son de entrada o de salida.
 - Declaración de variables: Se declaran todas las variables globales del sistema.
 - Declaración de prototipos de las funciones: Esto es necesario para la correcta compilación del código C18.
 - Inicialización de los servicios de TCPIP Strack: en esta sección se configuran los registros del módulo de Ethernet, se inicializan las variables del TCPIP Strack y se inician los servicios del servidor HTTP, se inicia el sistema de ficheros MPFS, se asignan los valores del IP, se establece la mascara de subred, y se establece la conexión de TCP.

- Inicio de los vectores de interrupción: Para hacer esto es necesario el siguiente código.

```
void InterruptHandlerHigh();

#pragma code InterruptVectorHigh = 0x08

void InterruptVectorHigh (void) {

    _asm

        goto InterruptHandlerHigh //jump to interrupt routine

    _endasm

}

#pragma code

#pragma interrupt InterruptHandlerHigh
```

En la primera línea se establece la función prototipo donde estará el código del servicio de interrupciones, en las otras líneas se establece que en la dirección 0x08 este un salto “goto InterruptHandlerHigh” a la función que contiene el servicio de interrupciones “InterruptHandlerHigh()”.

- Inicialización del módulo de LCD: para esto se llama a la instrucción “LCDInit()”. Esto se explica mejor en la sección de Módulo de LCD.
 - Inicialización del módulo de EUSART: para esto se le asigna valores a los registros de configuración del EUSART. Una explicación mayor de este módulo se encuentra en la sección de Módulo de EUSART.
 - Inicio de la posición de los servos: Este proceso acomoda la posición de giro de la pinza y abre la pinza, esto asegura una posición inicial para procesos posteriores en la planta.
2. Atención de eventos de Ethernet: en esta sección se verifican y atienden las peticiones de envío de datos de la pagina web y datos de las variables, también se reciben las ordenes a ejecutar por el sistema, para este procedimiento se utilizo el siguiente código.

```

while(1) {
    if (TickGet() - t >= TICK_SECOND/2) {
        t = TickGet();
        LED0_IO ^= 1;
    }
    StackTask();
    HTTPServer();
    CuentaBuffer=0;
    if ( strcmp(X_VICTOR[0],'L') == 0 ) {
        // llamada de función para atender al puerto serial
    } else if ( strcmp(X_VICTOR[0],'A')==0 || strcmp(X_VICTOR[0],'B')==0 ||
    strcmp(X_VICTOR[0],'C')==0 || strcmp(X_VICTOR[0],'D')==0) {
        // llamada de función para atender las instrucciones enviadas
por ethernet
    }
}

```

Todo el código está encerrado en un bucle infinito ya que la función “StackTask()” necesita estar chequeando los paquetes de entrada o salida al Ethernet periódicamente y así llamar a la función necesaria para atender los paquetes. Si hay un paquete de entrada este es atendido por la función “HTTPServer()” que en caso de ser un opcode se envía al arreglo de string “X_VICTOR[]”. Los siguientes bloques de if verifican cual opcode se recibió y así pasar el flujo del programa como lo muestra la Figura 40.

3. Atención del Puerto Serie: Recibe los datos del puerto serie y los almacena en un arreglo de string, esta recepción de datos se explica en la sección Módulo de EUSART.
4. Interpretación de comandos: Cada carácter recibido sea por Ethernet o por Puerto serie es almacenado en un arreglo de string, cuando la recepción de caracteres es completa, se procede a la inspección en busca de las órdenes a ejecutar y de los parámetros escogidos de la planta, como es el tiempo de cada conmutador, los valores numéricos son almacenados otras en variables del sistema para su posterior utilización. El siguiente diagrama muestra este proceso de inspección de los datos recibidos.

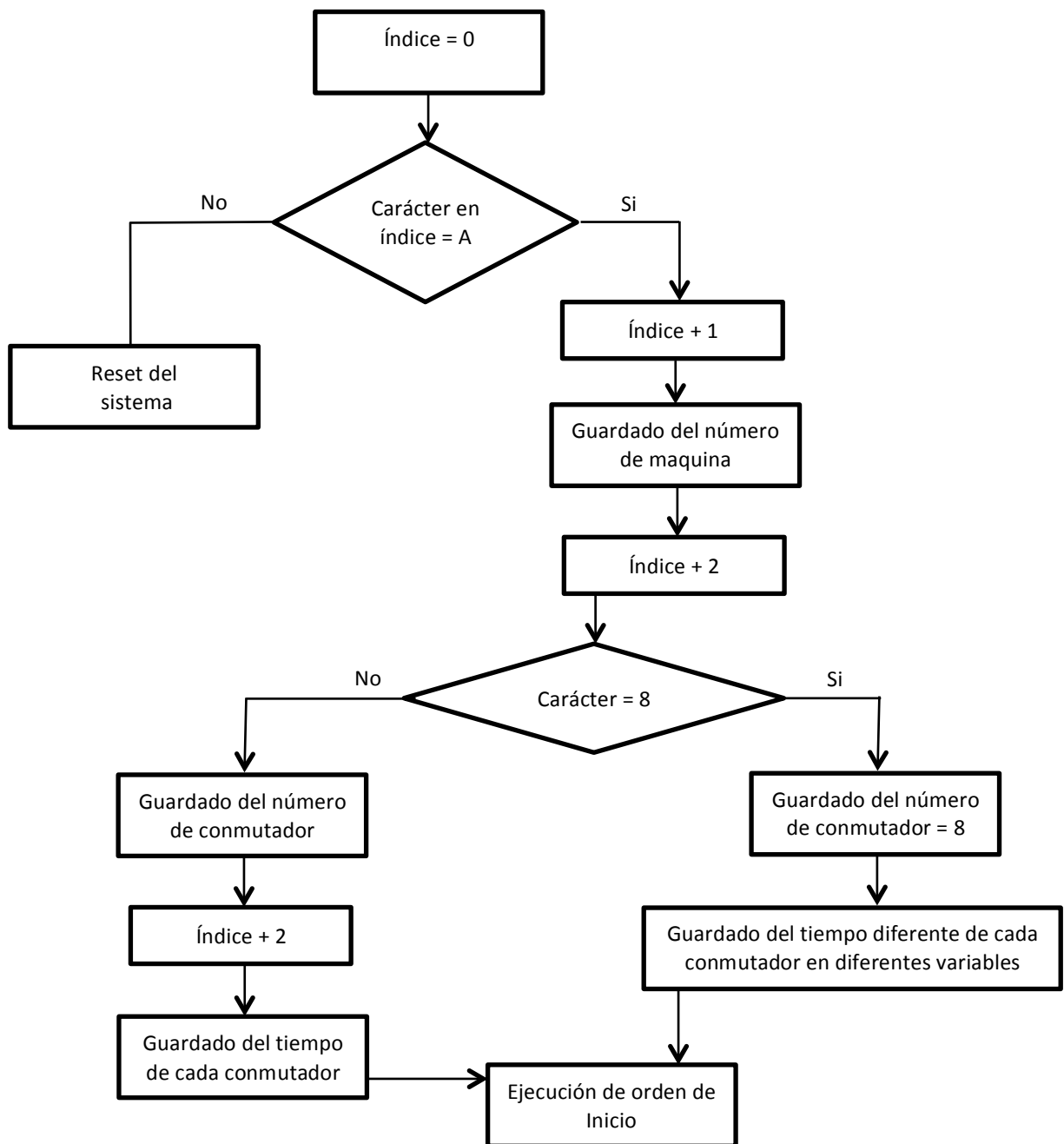


Figura 40. Diagrama de flujo para la extracción de los datos enviados.

5. Ejecución de órdenes para el control de la planta y atención de eventos de Ethernet o Puerto Serie: De acuerdo a las órdenes recibidas se ejecutan las señales de PWM de control de los servomotores, esto se explica mejor en la sección de control. Además periódicamente se atiende el Ethernet o el puerto serie en busca de las instrucciones de detener, reanudar o reset. Para lograr la atención periódica de los puertos, se creó la siguiente función, que es llamada repetidamente durante este bloque del programa:

```
static void AtencionRed(void){
    if (FinCComm == 20){
        for(cuenta3=0;cuenta3<50;cuenta3++){
            DelayMs(1000);
            StackTask();
            HTTPServer();
            if ( strcmp(X_VICTOR[0],'F') == 0 ){
                _asm
                    reset
                _endasm
            }
            if ( strcmp(X_VICTOR[0],'E') == 0){
                PausaRed();
                ReanudarRed();
            }
        }
    } else
    if (FinCComm == 1){
        if ( strcmp(X_VICTOR[0],'E') == 0){
            Pausa();
            Reanudar();
        }
    }
}
```

Aquí se utiliza la variable “FinCComm” para verificar si se tiene que atender al puerto Ethernet o al puerto serial.

En el caso de atender al puerto Ethernet (FinCComm = 20) se ejecuta un ciclo 50 veces con un delay de tiempo, así se da un periodo de tiempo para verificar el estado del puerto, también se verifica el opcode recibido para ejecutar ya sea el reset del sistema, la pausa o el reanudar.

Solo durante este bloque de ejecución de programa se utilizan las interrupciones del EUSART para la recepción de datos mediante el siguiente código, luego se compara el opcode para ejecutar la función correcta.

```

If (PIR1bits.RC1IF==1){
    PIR1bits.RC1IF = 0;
    if (RCREG == 'E'){
        X_VICTOR[0] = 'E';
    } else if (RCREG == 'H'){
        X_VICTOR[0] = 'H';
    } else if (RCREG == 'F'){
        _asm
            reset
        _endasm
    }
}

```

Este código atiende la interrupción por recepción de datos en el módulo de EUSART, verifica el valor de entrada y lo escribe en el arreglo de string de datos de entrada, para que luego en la función “AtencionRed()” se pregunte por el valor de entrada y se llame a la operación correcta.

6. Pausa: Detiene el proceso mediante apagar las señales de PWM y detener el avance del tiempo, a la vez que guarda el tiempo transcurrido, luego ejecuta el parpadeo del LED del LCD como señal para el usuario.
Para detener las señales de PWM, ya que esta son ejecutadas mediante los módulos de timer, lo que se necesita hacer es detener los timer utilizando la función “CloseTimer()”.

El siguiente código se utiliza en la función de pausa para generar el parpadeo del LED y verificar la llegada de la orden de Reanudar o reset, tal como se muestra en el flujo del programa de la Figura 41.

```

while ( strcmp(X_VICTOR[0], 'E') == 0 ){
    if (DataRdyUART()){ //para cuando se de Reanudar o Reset
        X_VICTOR[0] = ReadUART();
    }
    if ( strcmp(X_VICTOR[0], 'F') == 0 ){
        _asm
            reset //para la orden de reset
        _endasm
    }
}

```

```

        _endasm
    }
    If (TickGet() - StartTimeYo > TICK_SECOND){
        TickInit();
        StartTimeYo = TickGet();
        if ( LED3_IO == 0 ) { //parpadeo de la pantalla
            LED3_IO = 1;
        }else {
            LED3_IO = 0;
        }
    }
}

```

El bucle while mantiene el sistema en la pausa hasta que se reciba la orden de detener o reset, si se recibe el opcode F se ejecuta de una vez el reset, el bloque que esta dentro del tercer if se encarga de generar el parpadeo del LED con un periodo de un segundo, usando las funciones disponibles en la librería “Tick.c” como se explico en la sección Módulo de timer.

7. Reanudar: Reanuda el avance del tiempo desde el punto que se detuvo y enciende las señales de PWM mediante el reinicio de los timer.

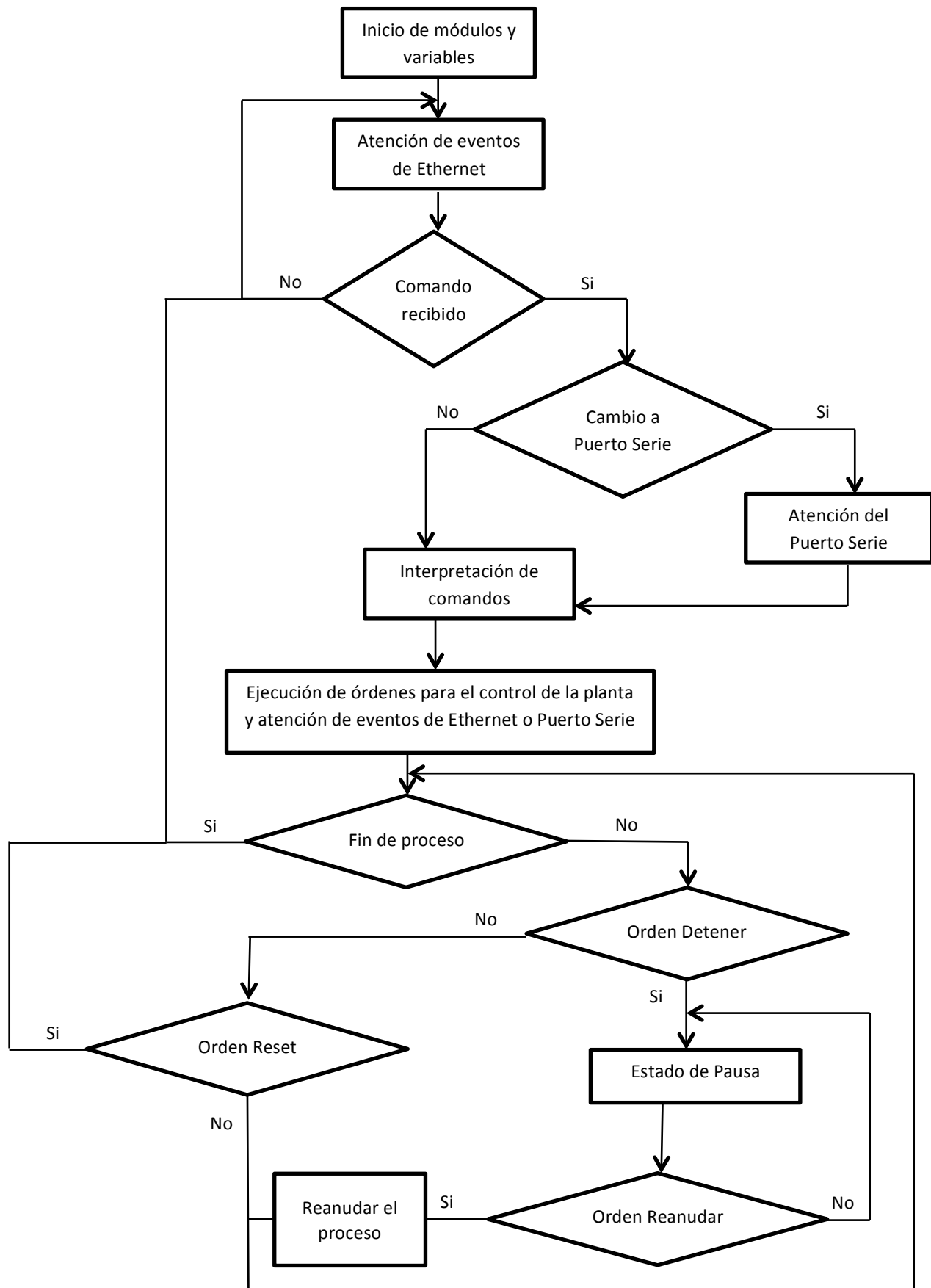


Figura 41. Diagrama de flujo del programa del microcontrolador.

5.5 Amplificación de la señal de PWM

Los servomotores necesitan una señal de control de 5 V, sin embargo la tarjeta de desarrollo da salidas de hasta 3.3 V, por lo que fue necesario utilizar una amplificación de la señal mediante la configuración de transistores, como lo muestra la Figura 42.

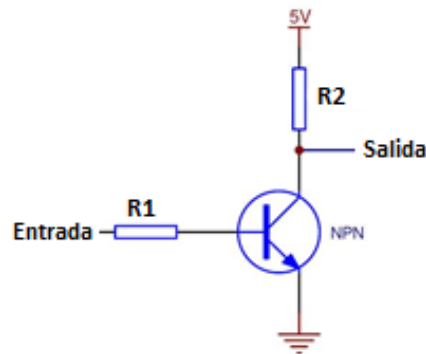


Figura 42. Configuración del transistor para aumentar la tensión del PWM.

La señal de Entrada proviene del kitt de desarrollo y la señal de salida se conecta a los servomotores en los pines de control. La inversión de la señal en esta configuración es compensada en la generación del PWM en el microcontrolador.

5.5.1 Pinza Big Gripper

Para este proyecto y la construcción de la planta para verificaciones, el Laboratorio LAMETRO donó la pinza Big Gripper. Esta pinza mide casi 3 pulgadas de longitud y tiene un ángulo de apertura que permite inclusive tomar con facilidad objetos del diámetro de una bola de tennis. Cuando tiene sus dos brazos abiertos, la distancia entre estos llega a ser de 2 ½ pulgadas. Esta pinza es controlada por un servomotor, que es la que le permite realizar su movimiento de apertura y cierre.

La estructura de la pinza está conformada de plástico PVC, que obtiene su forma mediante un mecanizado por medio de equipo CNC automatizado. Los brazos

están conformado de plástico de alto impacto y producidos por medio de un proceso de moldeo.

Las dimensiones de la pinza son: 6 pulgadas de largo, 2 $\frac{3}{4}$ pulgadas de ancho y 1 $\frac{1}{4}$ pulgadas de espesor. Su peso incluyendo el servomotor es de 3,7 onzas.

Esta pinza es utilizada en la planta como elemento primario de agarre y calibración de los conmutadores de las décadas, logrando la automatización del proceso de verificación y calibración de equipos de medición de resistencias de alto y bajo valor.



Figura 43. Pinza Big Gripper.

5.5.2 Cable STP reforzado

El cable STP utiliza una envoltura de protección, que ofrece un excelente apantallamiento, para así proteger los datos transmitidos de interferencias exteriores, lo que permite soportar mayores tasas de transmisión a distancias mayores.

El blindaje que posee este tipo de cable está diseñado para minimizar la radiación electromagnética y la diafonía. La longitud máxima de los cables de pares trenzados están limitados a 90 metros, ya sea para 10 o 100 Mbps.

En la planta construída, este cable se utiliza para enviar la señales de control y alimentación a los servomotores , y así evitar que los campos eléctricos y magnéticos, afecten la exactitud de la caja de décadas, en el momento de realizar las calibraciones y verificaciones.



Figura 44. Cable STP reforzado.

5.5.3 Señal del sensor

Según se muestra en al Figura 43 el sensor no es lineal, pero al observar la curva entre las distancias 20 cm a 80 cm, esta señal se pude aproximar a una línea recta, por lo cual no se prosede a utilizar un circuito acondicionador de señal en la implementación del sensor.

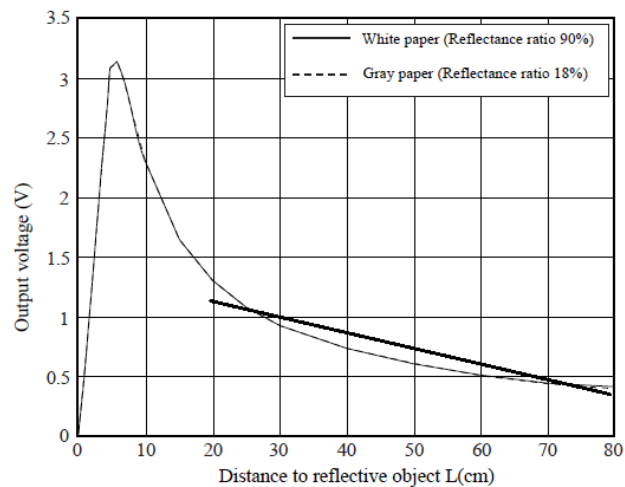


Figura 45. Comportamiento de la señal del sensor.

6 Presupuesto

Tabla 6. Tabla del presupuesto del proyecto contando todos los gastos y recursos que se tiene en el laboratorio, la totalidad se encuentra en colones.

Partidas	Valor
REMUNERACIONES	4300000
Servicios especiales	
SERVICIOS	1000000
SERVICIOS COMERCIALES Y FINANCIEROS	500000
SERVICIOS DE GESTION Y APOYO	0
Servicios médicos y de laboratorio	1200000
Servicios en ciencias económicas y sociales	900000
Servicio de desarrollo de sistemas informáticos	1000000
Gastos de viaje y transporte	100000
Materiales y suministros	300000
Materiales y productos de uso en mantenimiento	100000
Materiales y productos de cómputo	1000000
Accesorios	500000
Bienes Duraderos	15000000
Equipo sanitario, de laboratorio e investigación	3000000
Total	28900000

7 Conclusiones

- El servomotor viene diseñado con su propio control por lo que éste no presenta sobre impulso.
- El usar cables extensos para controlar los servomotores, produce ruido en los servos debido al ruido electromagnético producido por señales de otros servos.
- Se requiere diseñar un bloque Antiwindup para acotar la salida del control a valores físicamente posibles para la planta.
- El cable blindado disminuye el ruido electromagnético a un 90%.
- El modelo empírico de la planta es de segundo orden.
- Con el control automático diseñado para la planta mejoró el tiempo de estabilización y el error de estado estacionario.
- La planta diseñada produjo un sistema de operación más eficiente, preciso y redujo el tiempo de los procesos a 2 horas.
- El diseño amigable de las interfaces para el proceso garantizó la operación adecuada del sistema de calibración y verificación por parte del personal de LAMETRO.
- Los materiales utilizados en la planta, benefician el desempeño de la planta, ya que aportan estabilidad y rigidez necesarias para sostener a la planta y todos sus componentes en movimiento, a la misma vez que la estructura plástica permite un movimiento preciso sin agregar un peso estructural innecesario.
- La planta creada genera una interfase entre el usuario y el equipo a calibrar, que permite de manera simple, generar los movimientos requeridos para manipular el equipo deseado.

8 Recomendaciones

- La planta, para variar las magnitudes de los patrones de resistencias, debería de ser construida de un material más rígido, el cual de mayor estabilidad al sistema.
- Utilizar una pinza de un material de alta resistencia, con un sistema hidráulico entre sus brazos, que brinde una mayor presión sobre el conmutador.
- Diseñar un acople para los conmutadores, constituido por un sistema de engranajes, para reducir la fuerza que se debe aplicar para variar cada magnitud en el sistema.

9 Glosario

Caja reductora: sistema compuesto por engranajes, con la función de mantener la velocidad de salida cercano al valor deseado, transformando la velocidad de giro en torsión.

CNC (Control Numérico Computarizado): es una computadora programada, la cual controla la posición y velocidad de los motores que accionan los ejes de la máquina, por lo que pueden ejecutar trayectorias tridimensionales complejas.

Calibración: es un conjunto de operaciones que establecen, bajo condiciones especificadas, la relación entre los valores de magnitudes indicados por un instrumento o sistema de medición, o valores representados por una medida materializada o un material de referencia y los correspondientes valores aportados por patrones.

E.C.A.: Ente Costarricense de Acreditación, es la organización responsable de otorgar y emitir las acreditaciones en el país, contribuyendo a mejorar la calidad y la productividad de empresas e instituciones en sus productos, bienes y servicios. Permitiendo así cumplir los objetivos legítimos del Estado.

Estabilidad dimensional: Propiedad que tienen ciertos materiales que al ser sometidos a cambios de temperatura y humedad no pierden su forma y mantiene sus dimensiones originales.

Exactitud: es la capacidad de un instrumento de acercarse al valor de la magnitud real. La exactitud depende de los errores sistemáticos que intervienen en la medición, denotando la proximidad de una medida al verdadero valor y, en consecuencia, la validez de la medida.

Incertidumbre: una estimación unida al resultado de un ensayo que caracteriza el intervalo de valores dentro de los cuales se afirma que está el valor verdadero.

Mecanizado: este término se engloba bajo otro más amplio: el devastado, que es cualquier técnica de corte que desprende virutas del material como resultado del proceso.

Resistencia mecánica: La resistencia mecánica es la capacidad de los cuerpos para resistir las fuerzas aplicadas sin romperse.

Transductor: es un dispositivo capaz de transformar o convertir un determinado tipo de energía de entrada, en otra diferente a la salida. Es

comúnmente utilizado para obtener la información de entornos físicos y químicos y conseguir (a partir de esta información) señales o impulsos eléctricos o viceversa.

Trazabilidad: La propiedad del resultado de una medida o del valor de un estándar donde éste pueda estar relacionado con referencias especificadas, usualmente estándares nacionales o internacionales, a través de una cadena continúa de comparaciones todas con incertidumbres especificadas.

10 Bibliografía

- [1] Página oficial del ICE < www.grupoice.com >
- [2] IET LABS, INC. Manual para HRRS-5Kv SERIES, 2006.
- [3] TestEquipmentDepot. Manual para Decade Resistance Boxes < www.testequipmentdepot.com >
- [4] Microchip < www.microchip.com >
- [5] International Vocabulary of Basic and General Terms in Metrology of ISO
- [6] Guía ISO 3534-1
- [7] SHARP. GP2Y0A21YK0F. Hoja de datos. 01/12/2006. En línea: http://www.sharpsma.com/webfm_send/1489. Consultado el 9 de mayo de 2012.
- [8] OLIMEX. PIC-MAXI-WEB. Hoja de datos. Abril del 2009. En línea: <https://www.olimex.com/dev/pdf/PIC/PIC-MAXI-WEB.pdf>. Consultado el 23 de marzo de 2012.
- [9] Microchip. MPLAB C18 C COMPILER LIBRARIES. 2005. En línea: http://ww1.microchip.com/downloads/en/DeviceDoc/MPLAB_C18_Libraries_51297f.pdf. Consultado el 29 de marzo de 2012.
- [10] Microchip. PIC18F97J60 Family Data Sheet. 2011. En línea: <http://ww1.microchip.com/downloads/en/DeviceDoc/39762f.pdf>. Consultado el 29 de marzo de 2012.
- [11] Microchip. The Microchip TCP/IP Stack. 2008. En línea: <http://ww1.microchip.com/downloads/en/AppNotes/00833c.pdf>. Consultado el 9 de julio de 2012.
- [12] Ogata, K. (1987). Dinámica de sistemas (3ª ed.). México: Prentice Hall.
- [13] El Servomotor. En línea: <http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/servomotor.htm> Consultado el 6 de noviembre del 2012.
- [14] Manual Control Automático. En línea: http://www.ie.itcr.ac.cr/einteriano/control/Laboratorio/ProyectosCortos/ProyectoCorto1_I_2011.pdf. Consultado el 8 de febrero del 2012.

11 Anexos

11.1 Paso del programa en formato de código a formato ejecutable

Para compilar el código java se utilizó el programa de edición de texto TextPad, que se vincula a la plataforma de java jdk para compilar y mostrar la ejecución del código, pero este programa no genera un archivo ejecutable. Los pasos necesarios para obtener un fichero ejecutable del código del programa se presentan a continuación, estos pasos se dividen en dos procesos principales obtener un fichero .jar del fichero .java y luego utilizar el programa gratuito “JSmooth” para obtener el fichero .exe a partir el fichero .jar.

Primer proceso, este parte se ejecuta toda en la línea de comandos de Windows:

- Establezca el path donde se encuentra la carpeta bin del jdk de java, esto se hace mediante la siguiente línea:
`set path=%path%;C:\Archivos de programa\Java\jdk1.6.0_05\bin`

- En el directorio donde está el fichero .java ejecute el comando:
`javac NombreDelArchivo.java`

- Mediante un editor de texto cree el archivo manifest.mf y lo coloca en el directorio donde está el archivo .java, este archivo debe contener la siguiente información:

Manifest-Version: Indica la versión del archivo

Created-By: Autor del archivo.

Main-Class: Especifica la clase principal que será la primera en ejecutarse

Class-Path: Indica librerías extras que ocupará la aplicación, por ejemplo conectores JDBC para las bases de datos o librerías de terceros.

X-COMMENT: Aquí podrá colocar comentarios extras.

- Luego ejecute en línea de comandos lo siguiente:
`jar cvfm NombreDelArchivo.jar manifest.mf NombreDelArchivo.class otras_carpetas`
- Comprobar que el fichero funciona correctamente mediante el comando:
`java -jar NombreDelArchivo.jar`

Segundo proceso, este parte se ejecuta en el programa “JSmooth”:

- En la pestaña “Skeleton” seleccione la opción “Windowed Wrapper”
- En la pestaña “Executable” se escoge el nombre del fichero .exe y la dirección donde se guarda, además esta la opción de asociarle un icono al programa.
- En la pestaña “Application” primero se carga en la sección Classpath el .jar creado anteriormente, luego se escoge el “Main Class” que corresponde a la clase principal del programa que contiene el procedimiento “Main”.
- Ejecutar la opción compile del programa, con esto se crea el fichero ejecutable.

11.2 Esquema de conexión a los pines del kit de desarrollo

La Figura 46 muestra el kit de desarrollo y lo diferentes puertos de conexión, de donde se puede ver claramente conexión para el RS-232 y el ethernet.

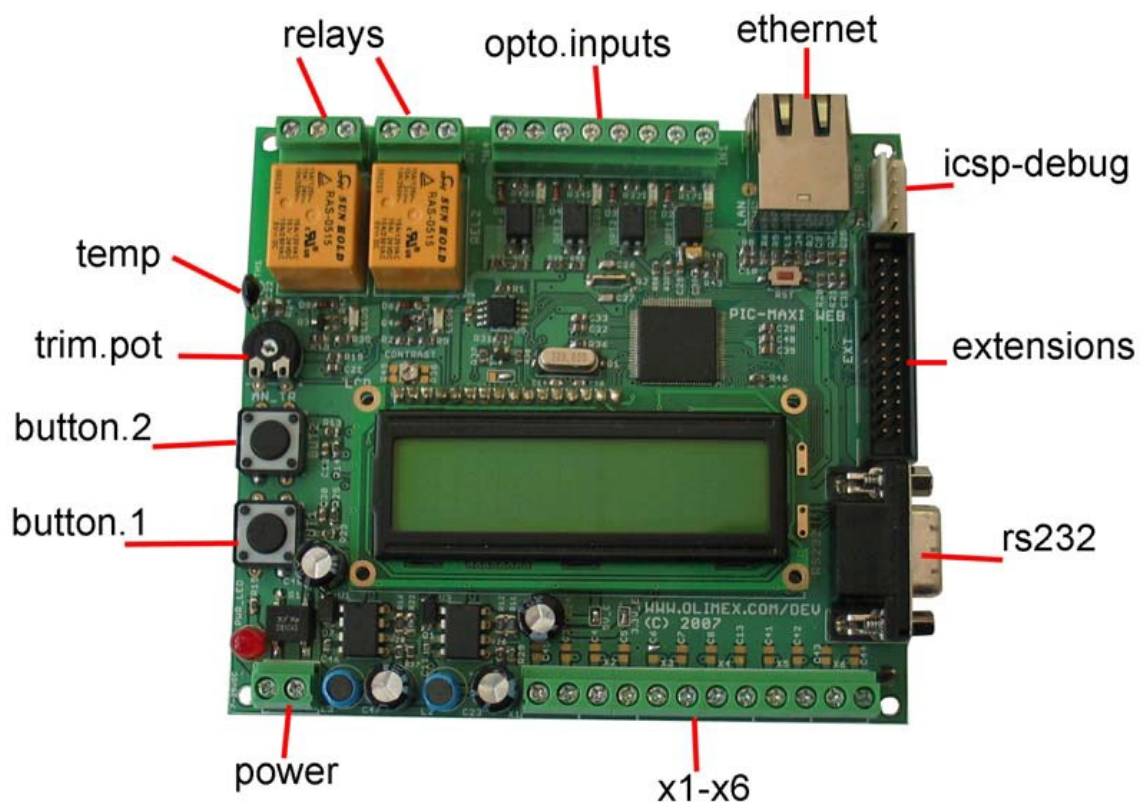


Figura 46. Puertos de conexión del kitt de desarrollo. [2]

En la Figura 47, se muestra el puerto X1-X6, donde se conecta los pines 4 y 5, para los sensores que establecen el movimiento en el eje x y en el eje y, correspondientemente.

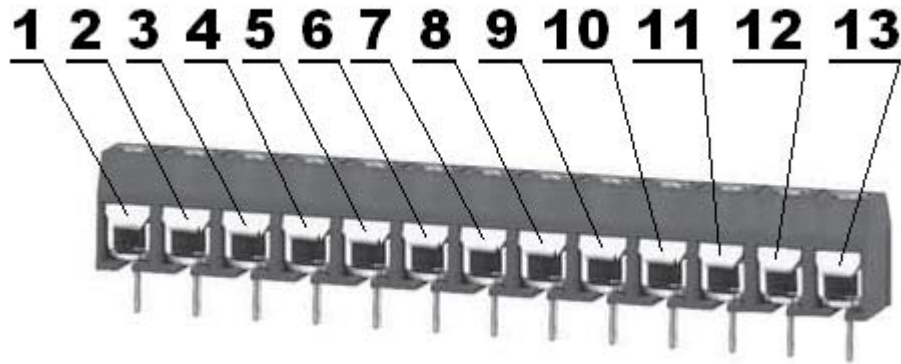


Figura 47. Numeración de los pines del puerto X1-X6. [2]

La Figura 48, se muestra la numeración del puerto de extensión, en este puerto se utiliza para los pines que tienen la señal de PWM, de la forma:

- Pin 12: Servomotor 1 para movimiento en el eje X.
- Pin 14: Servomotor 2 para movimiento en el eje X.
- Pin 16: Servomotor para movimiento en el eje Y.
- Pin 9: Servomotor para el giro de la pinza.
- Pin 11: Servomotor para el cierre de la pinza.

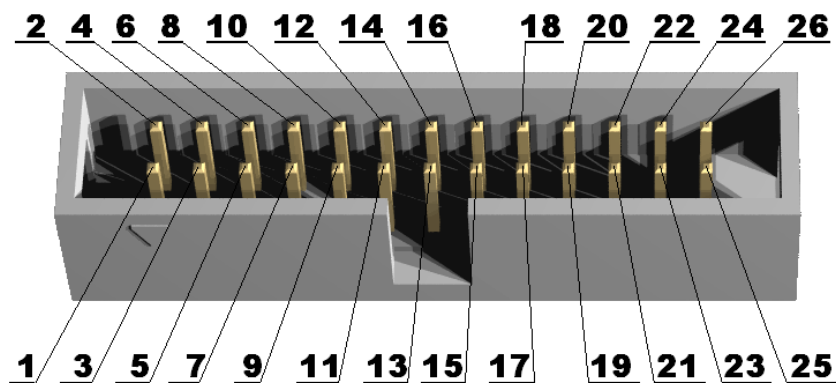


Figura 48. Numeración de los pines del puerto de extensión. [2]

11.3 *Obtención empírica del modelo del servomotor para la planta*[14]

Se utilizó el software Matlab para estimar, a partir de los resultados experimentales de entrada-salida del servo, el modelo o modelos óptimos para la planta.

Para esto se inicia Matlab y se ingresa en menú al File\Import Data..., se selecciona el archivo contenedor de los resultados experimentales, y luego oprima Open y luego Next, en la siguiente ventana seleccione Create vectors, como se muestra en la Figura 47 y luego oprima Finish. Los datos están ahora disponibles como vectores en el workspace de Matlab.

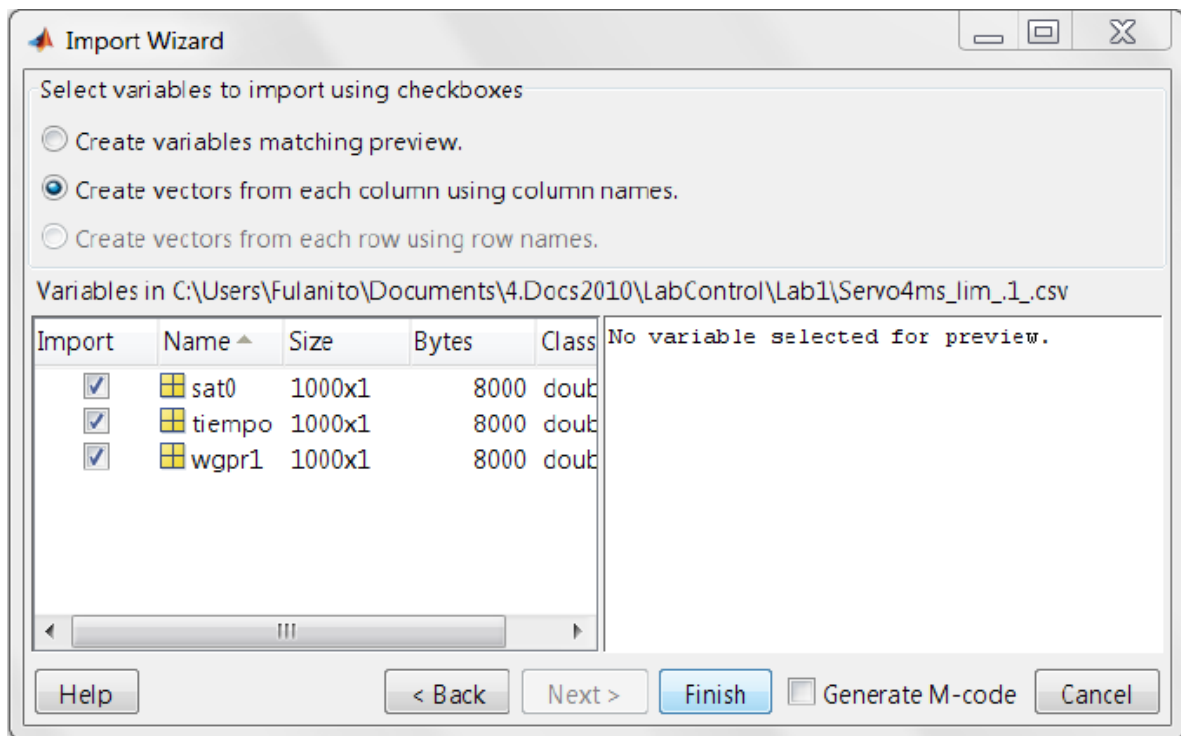


Figura 49. Importación de datos al workspace de Matlab en forma de vectores.

Luego en Matlab ejecute ident; espere unos segundos y se abrirá la ventana del ident, en la cual debe seleccionar Import data\Time domain data, llene los campos de la ventana con los datos correspondientes a su experimento: nombres de las variables de entrada y salida, tiempo de muestreo y otros como se

muestra en la Figura 48 y oprima Import. Una vez hecho lo anterior puede cerrar esa ventana.

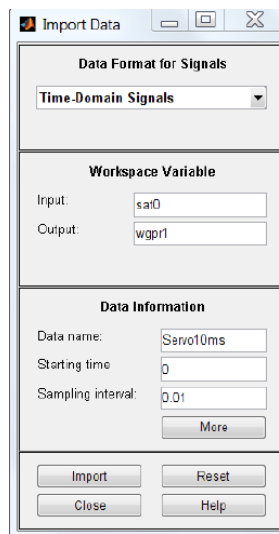


Figura 50. Ventana para importar datos al ident.

En la ventana del ident debe tener ahora un objeto. Este objeto contiene todos los datos del experimento, se debe seleccionar en dos partes, una para estimar el modelo y otra para validar los datos. Se utiliza para ello la operación Preprocess\Select range, que abrirá una ventana en la cual con el ratón se puede crear un rectángulo alrededor de los datos a seleccionar y luego oprimir Insert; repitiendo para cada rango a crear, como se muestra en la Figura 49; al terminar de insertar los segmentos seleccionados se debe cerrar esta ventana. Al finalizar se debe tener tres objetos en la ventana del ident.

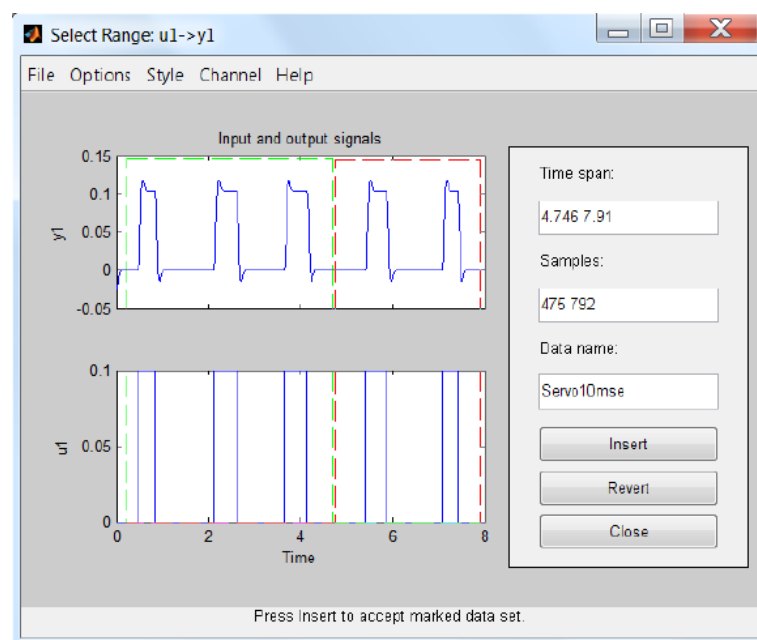


Figura 51. Selección de dos rangos del total de datos.

Se arrastra y se coloca el segundo objeto (color verde) en la posición llamada Working Data y el tercer objeto (color rojo) en la posición llamada Validation Data. Finalmente en la ventana del ident seleccionar Estimate\Process models... y en esta ventana, ya que el modelo no presenta un comportamiento subamortiguado, seleccionaremos la opción de 1 polos subamortiguados y se selecciona la opción de retardo (Delay) activada. Una vez hecho lo anterior se oprime el botón Estimate, el cual hará que se calcule un modelo de primer orden llamado PDU que aparecerá en la ventana del ident como se muestra en la Figura 50.

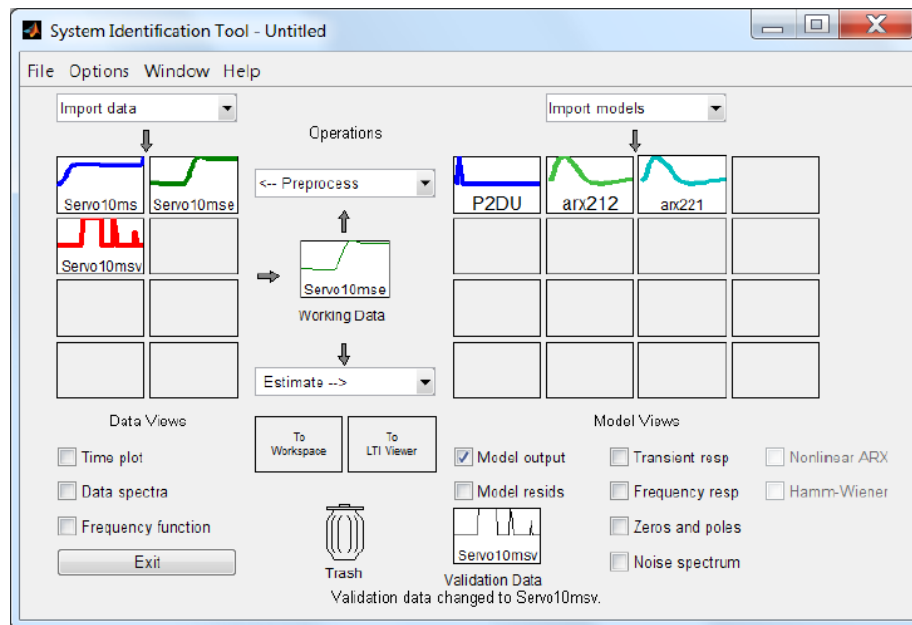


Figura 52. Ventana del ident con los modelos obtenidos.

Se puede también seleccionar otra forma de estimación para el modelo; por ejemplo Estimate\Linear parametric models, en cuya ventana se oprime Order selection para que calcule 10 modelos desde orden 1 a 10 y luego oprimir Estimate. En la ventana de resultados del estimador paramétrico ARX [1], [2] y [3], seleccionar el tercer modelo y el segundo modelo, cuyas varianzas de salida no explicada están sumamente reducidas respecto al primer modelo y luego oprimir Insert. El resultado se muestra en la figura 7, donde los modelos aparecen al lado del modelo PDU estimado antes.

Para validar los modelos obtenidos, se puede comparar su salida ante el estímulo original del experimento seleccionando la opción Model output que nos

presenta el porcentaje de aproximación entre el modelo seleccionado y la respuesta original como se muestra en la Figura 51.

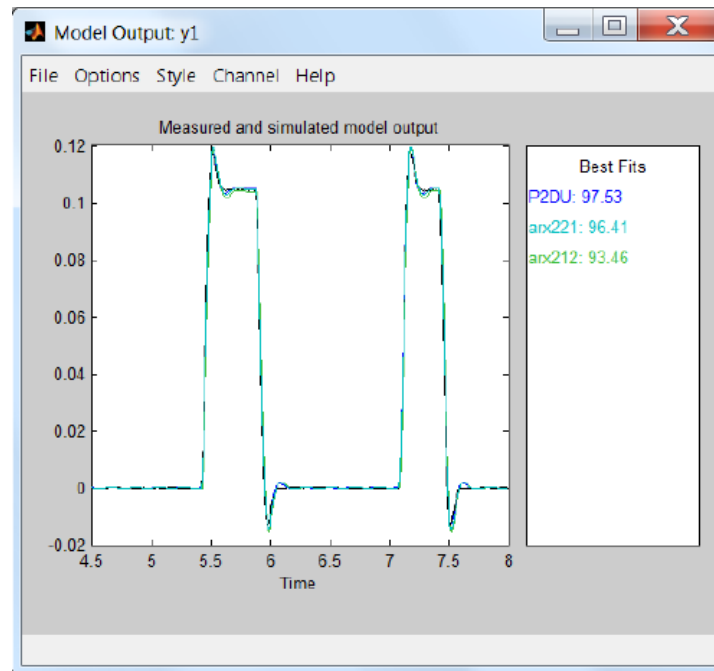


Figura 53. Comparación de modelos con Model output.

Para exportar el modelo seleccionado se arrastra hasta la posición To workspace en la ventana del ident. Luego en el workspace de Matlab, se puede extraer la función de transferencia de entrada-salida, con la sentencia: `modelo = zpk(P2DU,1);` donde "modelo" es el nombre asignado al resultado.