

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE GUAYAQUIL

CARRERA:

INGENIERÍA ELECTRÓNICA

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO ELECTRÓNICO

TÍTULO:

REPOTENCIACIÓN DE LA PLATAFORMA INERCIAL DEL LABORATORIO DE CONTROL AUTOMÁTICO UTILIZANDO CONTROLADORES PID Y FUZZY A TRAVÉS DE MATLAB/SIMULINK

AUTORES:

JARA LOAIZA WILLIAM ANDRÉS CHALACÁN AGUAYO VICENTE DANIEL

TUTOR:

MSC. VICENTE PEÑARANDA

GUAYAQUIL – ECUADOR

2019

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Nosotros, WILLIAM ANDRÉS JARA LOAIZA y VICENTE DANIEL CHALACÁN AGUAYO, estudiantes de Ingeniería Electrónica de la Universidad Politécnica Salesiana, certificamos que los conceptos desarrollados, análisis realizado, y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

Guayaquil, septiembre del 2019

William Andrés Jara Loaiza

C.I.: 0926210634

Vicente Daniel Chalacán Aguayo

C.I.: 0940433519

CERTIFICADO DE CESIÓN DE DERECHOS

A través del presente certificado, se ceden los derechos de propiedad intelectual correspondiente a este trabajo, a la Universidad Politécnica Salesiana, según lo establecido por la ley de la propiedad intelectual y por su normatividad institucional vigente.

Guayaquil, septiembre del 2019

William Andrés Jara Loaiza

C.I.: 0926210634

Vicente Daniel Chalacán Aguayo

C.I.: 0940433519

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo Msc. Vicente Avelino Peñaranda Idrovo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación "REPOTENCIACIÓN DE LA PLATAFORMA INERCIAL DEL LABORATORIO DE CONTROL AUTOMÁTICO UTILIZANDO CONTROLADORES PID Y FUZZY A TRAVÉS DE MATLAB/SIMULINK" realizado por los estudiantes, WILLIAM ANDRÉS JARA LOAIZA con cédula de identidad N.º 0926210634 y VICENTE DANIEL CHALACÁN AGUAYO con cédula de identidad N.º 0940433519, obteniendo un producto que cumple con los objetivos del diseño de aprobación, informe final y demás requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Guayaquil, septiembre del 2019

Msc. Vicente Avelino Peñaranda Idrovo

DIRECTOR DEL PROYECTO

DEDICATORIA

Dedico este proyecto de titulación a mis padres Vicente Chalacán y Lola Aguayo, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí y convertirme en lo que soy. Ha sido el orgullo y el privilegio de ser su hijo, son los mejores padres. A mis hermanas Natalia Chalacán, Maria Chalacán y Nadia Chalacán por estar siempre presentes, acompañándome y por el apoyo moral, que me brindaron a lo largo de esta etapa de mi vida. A todas las personas que nos han apoyado y han hecho que el trabajo se realice con éxito en especial a aquellos que nos abrieron las puertas y compartieron sus conocimientos.

VICENTE DANIEL CHALACÁN AGUAYO

DEDICATORIA

Dedico este proyecto de titulación a mis padres Willan Amado Jara Leon y Blanca Piedad Loaiza Pacheco por ser la mayor motivación de avanzar y crecer tanto profesionalmente, como en la vida personal. A mi hermana Lady Briggitte Jara Loaiza para que este trabajo sirva como influencia y motivación en ella, que llegue a tal sitial y mucho más en su futura carrera profesional.

A cada una de las personas que dieron su ayuda, amigos, conocidos, profesores, compañeros de trabajo, ya que con esto se logró concluir el arduo trabajo que implicó la investigación y construcción de todo el proyecto.

WILLIAM ANDRÉS JARA LOAIZA

AGRADECIMIENTO

Agradecemos principalmente a Dios por darnos la vida y a nuestros padres por darnos su apoyo incondicional. A todas y cada una de las personas que nos dieron su granito de arena con ideas, tiempo y trabajo para que este proyecto sea una realidad y logremos conseguir nuestro título profesional.

Al Msc. Vicente Peñaranda por su guía oportuna y concisa. Al Msc. Víctor Huilcapi por su acertada dirección en nuestra carrera.

VICENTE DANIEL CHALACÁN AGUAYO

WILLIAM ANDRÉS JARA LOAIZA

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE PROYECTO	TEMA DE PROYECTO DE TITULACIÓN
2019	CHALACÁN AGUAYO VICENTE DANIEL JARA LOAIZA WILLIAM ANDRÉS	MSC. VICENTE PEÑARANDA	"REPOTENCIACIÓN DE LA PLATAFORMA INERCIAL DEL LABORATORIO DE CONTROL AUTOMÁTICO UTILIZANDO CONTROLADORES PID Y FUZZY A TRAVÉS DE MATLAB / SIMULINK"

El presente trabajo de titulación tiene como objetivo realizar una guía de prácticas del funcionamiento de la plataforma inercial de dos grados de libertad utilizando controladores PID y FUZZY, con ayuda del entorno de programación Simulink, que funciona sobre el software Matlab. La plataforma se encuentra ubicada en el laboratorio de control automático del bloque 'E' de la Universidad Politécnica Salesiana sede Guayaquil campus Centenario y se enfoca en reforzar el aprendizaje práctico de los controladores antes mencionados. Para esto se tiene como referencia la documentación con el tema "*Diseño e implementación de una plataforma inercial de dos grados de libertad y evaluación del modelado matemático a través de Matlab/Simulink*" realizada en el año 2015. En el mencionado documento se encuentra el diseño, construcción y modelado de la plataforma inercial de dos grados.

A partir del análisis ejecutado sobre el diseño propuesto en el año 2015, se puede describir la plataforma con los siguientes elementos: motores reductores DC con encoder, pantalla táctil resistiva, mecanismo con distribución de pesos para el movimiento del plato donde se aloja la pantalla táctil, acelerómetro analógico, microcontroladores, fuentes de poder, tarjeta electrónica puente H para control de los motores, tarjeta de circuito impreso o PCB para conexión de todos los elementos y una tarjeta de adquisición de datos PCI-1711 como medio de comunicación entre la plataforma inercial y el software Matlab/Simulink. Considerando la existencia de nuevos recursos electrónicos, se efectúa una reingeniería a la propuesta del 2015 que condujo a las siguientes mejoras: los dos motores reductores DC con encoder, son reemplazados por dos servomotores como actuadores en el sistema, por ende, la tarjeta electrónica puente H queda sin uso. La pantalla táctil resistiva se reemplaza

por el mismo modelo, aunque con un mejor terminal de conexión. Esta pantalla en conjunto con una placa arduino NANO son utilizados como sensor para obtener las coordenadas del cuerpo útil (esfera) en el plato.

El mecanismo para el plato, se lo rediseña agregando un pilar base en el centro con un rodamiento tipo rotula, el cual permite libre movimiento. En conjunto con esto, se agregan dos brazos articulados que transforman el movimiento angular de los servomotores a movimiento lineal para el eje X y el eje Y. Se decide cambiar las tarjetas PCB y PCI-1711 por una placa arduino DUE que, al ser un elemento muy usado en la electrónica actual, disminuye el costo del proyecto, facilita el control del sistema y la comunicación con los elementos (sensor y actuadores) se vuelve más sencilla. Además, se instaló una etapa eléctrica de potencia para el encendido y apagado del sistema con, un botón de paro, un botón de marcha y un botón tipo hongo para el paro de emergencia. La comunicación entre la plataforma inercial de dos grados de libertad y el computador se realiza mediante la placa arduino DUE que permite el control del movimiento en los servomotores y la lectura de las coordenadas de la pantalla táctil mediante Matlab/Simulink. La comunicación entre el arduino DUE

Para el diseño de los controladores es fundamental trabajar en el modelado matemático de la planta, la identificación se realiza empleando la herramienta "Sistema de identificación" (systemIdentification o ident) propia del software Matlab. Gracias a la adquisición de datos de la plataforma y procesamiento de dichos datos, se obtiene las funciones de transferencia que nos indica y describe el funcionamiento de todo el sistema. Se obtiene las funciones de transferencia por separado en cada eje, facilitando la identificación y por ende el control de la planta. De esta forma se tiene un sistema MIMO (**M**ultiple-Input **M**ultiple-**O**utput) general en el que su control incluye la función de transferencia para el eje X y una segunda función de transferencia para el eje Y, por motivos que se detallan en este trabajo, el sistema general se separa en dos sistemas SISO (**S**ingle-Input **S**ingle-**O**utput) uno por cada eje que trabajan en conjunto para el equilibrio de la esfera.

Palabras Clave: Controladores, PID, FUZZY, Plataforma Inercial, Grados de Libertad, Software Matlab, Entorno de programación Simulink, Arduino, Pantalla táctil resistiva, Servomotor, tarjeta PCB, tarjeta PCI-1711, arduino DUE, arduino NANO, SISO, MIMO.

ABSTRACT

YEAR	STUDENTS	PRJ. DIRECTOR	SUBJECT	
2019	CHALACÁN AGUAYO VICENTE DANIEL JARA LOAIZA WILLIAM ANDRÉS	MSC. VICENTE PEÑARANDA	"REPOTENTIATION OF THE INERCIAL PLATFORM OF THE AUTOMATIC CONTROL LABORATORY USING PID AND FUZZY CONTROLLERS THROUGH MATLAB / SIMULINK"	

This paper aims to develop a practice guide for the operation of the inertial platform of two degrees of freedom using PID and FUZZY controllers, with the help of the Simulink programming environment, which works on Matlab software. The platform is located in the automatic control laboratory of the 'E' block of the Universidad Politecnica Salesiana in Guayaquil and focuses on strengthening the practical learning of the aforementioned controllers. For this we have as reference the documentation with the theme "Design and implementation of an inertial platform of two degrees of freedom and evaluation of mathematical modeling through Matlab / Simulink" carried out in 2015. In the mentioned document is the design, construction and modeling of the inertial platform with two degrees of freedom.

From the analysis carried out on the design proposed in 2015, the platform can be described with the following elements: DC reduction motors with encoder, resistive touch screen, mechanism with weight distribution for the movement of the plate where the touch screen is housed, analog accelerometer, microcontrollers, power sources, electronic bridge H card for motor control, printed circuit board or PCB for connection of all elements and a PCI-1711 data acquisition card as a means of communication between the inertial platform and Matlab / Simulink software. Considering the existence of new electronic resources, a reengineering was carried out on the 2015 proposal which led to the following improvements: the two DC reduction motors with encoders, are replaced by two servomotors as actuators in the system, therefore, the H bridge electronic card is left unused. The resistive touch screen is replaced by the same model, although with a better connection terminal. This screen together with an Arduino NANO plate are used as a sensor to obtain the coordinates of the useful body (sphere) on the plate.

The mechanism for the plate is redesigned by adding a base pillar in the center with a ball-type bearing, which allows free movement. Along with this, two articulated arms are added that transform the angular movement of the servomotors to linear motion for the X axis and the Y axis. It was decided to change the PCB and PCI-1711 cards for an Arduino DUE board which, being an element widely used in current electronics, it reduces the cost of the project, facilitates system control and communication with the elements (sensor and actuators) becomes easier. In addition, an electric power stage was installed for switching the system on and off with a stop button, a start button and a mushroom button for emergency stop. The communication between the inertial platform of two degrees of freedom and the computer is carried out by means of the Arduino DUE board that allows the control of the movement in the servo motors and the reading of the coordinates of the touch screen by means of Matlab / Simulink. The communication between the Arduino DUE and the touch screen is done with software developed on the NANO Arduino board.

For the design of the controllers it is essential to work on the mathematical modeling of the plant, the identification is carried out using the "System Identification" toolbox (system Identification or ident) of the Matlab software. Thanks to the acquisition of data from the platform and the processing of said data, the transfer functions that describe the operation of the entire system are obtained. The transfer functions are obtained separately on each axis, facilitating the identification and therefore the control of the plant. In this way, there is a general MIMO (Multiple-Input Multiple-Output) system in which its control includes the transfer function for the X axis and a second transfer function for the Y axis, for reasons detailed in this work, the general system is separated into two SISO systems (Single-Input Single-Output) one for each axis that work together to balance the sphere.

Keywords: Controllers, PID, FUZZY, Inertial Platform, Degrees of Freedom, Matlab Software, Simulink programming environment, Arduino, Resistive touch screen, Servomotor, PCB card, PCI-1711 card, arduino DUE, arduino NANO, SISO, MIMO.

ÍNDICE GENERAL

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍAII
CERTIFICADO DE CESIÓN DE DERECHOSIII
CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN IV
DEDICATORIA V
AGRADECIMIENTO
RESUMENVIII
ABSTRACTX
ÍNDICE GENERAL XII
ABREVIATURAS Y SIMBOLOGÍAXV
ÍNDICE DE TABLASXVI
ÍNDICE DE FIGURASXVII
ÍNDICE DE ANEXOS XXVI
INTRODUCCIÓN1
1. EL PROBLEMA2
1.1 Descripción del problema2
1.2 Antecedentes2
1.3 Importancia y alcance2
1.4 Delimitación
1.4.1 Temporal
1.4.2 Espacial
1.4.3 Académico
1.5 Objetivos4
1.5.1 Objetivo General4
1.5.2 Objetivos Específicos4
2. ESTADO DEL ARTE
2.1 Sistemas inerciales
2.2 Plataforma Inercial
2.2.1 Grados de Libertad5
2.3 Servomotores
2.3.1 Partes de un servomotor
2.4 Panel Táctil Resistivo8
2.4.1 Panel táctil resistivo de 4 cables9
2.5 Tarjeta de circuito impreso o PCB10
2.5.1 Relé o Relevador11
2.5.2 Borneras y Espadines11

3.3.2	2 Servomotores	49
3.3.3	3 Arduino DUE	52
3.3.4	Identificación de la Planta	53
3.3.5	6 Identificación de la Planta en el eje X	59
3.3.6	Identificación de la Planta en el eje Y	65
3.4	Diseño del Algoritmo PID	71
3.4.′	Diseño del Algoritmo PID en el eje X	72
3.4.2	2 Diseño del Algoritmo PID en el eje Y	78
3.5	Diseño del Algoritmo FUZZY	85
3.5.1	Diseño del Algoritmo FUZZY en el eje X	87
3.5.2	2 Diseño del Algoritmo FUZZY en el eje Y	92
4. GUÍA	S DE PRÁCTICAS DE LABORATORIO	98
4.1	PRÁCTICA # 1	98
4.2	PRÁCTICA # 2	107
4.3	PRÁCTICA # 3	112
4.4	PRÁCTICA # 4	117
4.5	PRÁCTICA # 5	127
4.6	PRÁCTICA # 6	137
4.7	PRÁCTICA # 7	149
4.8	PRÁCTICA # 8	162
4.9	PRÁCTICA # 9	175
RESULT	ADOS	192
ANÁLIS	S DE RESULTADO	199
CONCL	JSIONES	201
RECOM	ENDACIONES	202
REFERE	ENCIAS BIBLIOGRÁFICAS	203
ANEXO	5	207
ANEX	O 1: TABLAS DE DATOS GENERADOS DE LA SEÑAL PRBS	207
ANEX	O 2: CRONOGRAMA DE ACTIVIDADES	212
ANEX	O 3: PRESUPUESTO DEL PROYECTO DE TITULACIÓN	213
ANEX	O 4: FACTURAS DE EQUIPOS Y ELEMENTOS VARIOS	213
ANEX	O 5: DATASHEET PANTALLA TACTIL RESISTIVA	215

ABREVIATURAS Y SIMBOLOGÍA

PID	Tipo de controlado Proporcional Integral Derivativo
FUZZY	Tipo de controlado basado en lógica difusa
SISO	Single-Input Single-Output
MIMO	Multiple-Input Multiple-Output
PCB	Printed Circuit Board
PCI-1711	Tarjeta Multifunción para el bus PCI
2-GDL	2-Grados de Libertad
NANO	Modelo de Placa Arduino para desarrollos de tamaño compacto.
DUE	Modelo de Placa Arduino para desarrollos de tamaño compacto.
PWM	Pulse Width Modulation
LED	Light Emitting Diode
ATMEGA328	Chip microcontrolador creado por Atmel
ATSAM3X8E	Chip microcontrolador creado por Atmel
CPU	Central Processing Unit
Memoria SRAM	Static Random Access Memory
DMA	Direct Memory Access
SAM3X	Microcontrolador basado en memoria flash
Memoria ROM	Read Only Memory
MCU	Microcontroller Unit
UART	Universal Asynchronous Receiver-Transmitter
TTL	Transistor to Transistor Logic
ATMEGA16U2	Microchip de alto rendimiento y bajo consumo.
ARDUINO IDE	Software de código abierto de Arduino XV

ÍNDICE DE TABLAS

55
56
89
198
209
216

ÍNDICE DE FIGURAS

Figura 2. 1: Grados de libertad en la visión de una persona.	6
Figura 2. 2: Servomotor DC y Servomotor AC.	6
Figura 2. 3: Variación de ángulos en el servomotor deacuerdo a su señal PWM.	7
Figura 2. 4: Partes de un servomotor DC	8
Figura 2. 5: Conector del servomotor con sus 3 cables, Vcc, GND y Señal	8
Figura 2. 6: Pantalla táctil de vidrio o touch panel, compuesta de varias capas	9
Figura 2. 7: Distribución de pines de conexión para el panel táctil resistivo	9
Figura 2. 8: Arreglo de resistencias en el panel táctil resistivo.	.10
Figura 2. 9: Tarjeta de circuito impreso.	.11
Figura 2. 10: Mecanismo de un relé y relé tipo electromagnético	.11
Figura 2. 11: Borneras, espadines hembra y espadines macho	.12
Figura 2. 12: Circuito RC y su gráfica en función de la frecuencia.	.12
Figura 2. 13: Porcentajes diferentes del ciclo de trabajo en la señal PWM	.13
Figura 2. 14: Ciclo de trabajo del 90% en la señal PWM	.14
Figura 2. 15: Ciclo de trabajo del 20% en la señal PWM	.15
Figura 2. 16: Control de brillo en un diodo LED con la señal PWM.	.15
Figura 2. 17: Tarjeta Arduino NANO.	.16
Figura 2. 18: Diagrama pines del Arduino NANO	.17
Figura 2. 19: Puertos de programación en Arduino DUE.	.18
Figura 2. 20: Diagrama pines del Arduino DUE	.19
Figura 2. 21: Menús y opciones del software arduino IDE.	.21
Figura 2. 22: Logo de software Matlab/Simulink	.22
Figura 2. 23: Sistema de control en lazo cerrado con control PID	.23
Figura 2. 24: Efectos en la señal modificando parámetros Kp, Kd, Ki	.25
Figura 2. 25: Sistema de control difuso Mamdani.	.26
Figura 2. 26: Funciones de membresía más usadas en los controladores difusos	.27
Figura 2. 27: Lógica de decisiones de un controlador difuso	.27
Figura 2. 28: Bloque "PID" en un sistema de lazo cerrado de Simulink	.28
Figura 2. 29: Paramentos del bloque "PID Controller"	.29
Figura 2. 30: Herramienta de sintonización de PID en el bloque "PID Controller".	.29
Figura 2. 31: Ejecución del comando "fuzzy" en la ventana de comandos Matlab.	.30
Figura 2. 32: Ventana para editar las funciones de membresía.	.30
Figura 2. 33: Ventana para editar el conjunto de reglas	.31
Figura 2. 34: Ventanas para probar y visualizar el efecto del controlador difuso	.31

Figura 2. 35: Bioque FUZZY en un sistema de lazo cerrado de Simulink	32
Figura 2. 36: Paramento "FIS name" del bloque FUZZY Logic Controller	32
Figura 3. 1: Plataforma Inercial 2-GDL, vista frontal	33
Figura 3. 2: Plataforma Inercial 2-GDL, acoples	33
Figura 3. 3: Tornillos de acoples dañados	34
Figura 3. 4: Diagnóstico de los motores reductores dc con encoder	34
Figura 3. 5: Diagnóstico de pantalla táctil resistiva	35
Figura 3. 6: Conector pantalla táctil resistiva	35
Figura 3. 7: Acelerómetro, vista de los pines	36
Figura 3. 9: Tarjeta PCI	36
Figura 3. 10: Tarjeta SparkFun.	37
Figura 3. 11: Pistas de la tarjeta PCB, vista posterior	37
Figura 3. 12: Tarjeta PCB, vista superior.	38
Figura 3. 13: Fuente de 12vdc/5Amp	38
Figura 3. 14: Fuente de 12vdc/30Amp	39
Figura 3. 15: Diseño de la Proforma Inercial 2-GDL, vista isométrica	39
Figura 3. 16: Medida en centímetros de la Proforma Inercial 2-GDL	40
Figura 3. 17: Ubicación de los servomotores con sus brazos y el pilar base	40
Figura 3. 18: Servomotor marca Fan Model	41
Figure 2 10: Deptalla Táctil Posistiva	41
Figura 3. 19. Faritalia Taclii Nesistiva.	
Figura 3. 20: Tarjeta Arduino DUE.	42
Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico	42 42
Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL.	42 42 42 43
 Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. 	42 42 43 44
 Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. 	42 42 43 44 44
 Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. 	42 42 43 44 44 44
 Figura 3. 19. Paritalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. 	42 42 43 44 44 44 45
 Figura 3. 19. Paritalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. 	42 42 43 44 44 44 45 46
 Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. 	42 42 43 44 44 44 45 46 48
 Figura 3. 19. Paritalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. Figura 3. 29: Diseño del filtro Pasa bajo. 	42 42 43 44 44 44 45 46 48 48
 Figura 3. 19. Pantalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. 	42 42 43 44 44 44 45 46 48 48 49
 Figura 3. 19. Pantalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. 	42 42 43 44 44 44 45 46 48 48 48 49 50
 Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. Figura 3. 32: Librería de Arduino para Simulink. 	42 42 43 44 44 44 45 46 48 48 48 49 50 51
 Figura 3. 19. Pantalia ractil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 28: Microcontrolador Arduino NANO. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. Figura 3. 32: Librería de Arduino para Simulink. Figura 3. 33: Bloques de la librería de Arduino instalada en Matlab- 	42 42 43 44 44 44 45 46 48 48 48 48 49 50 51
 Figura 3. 29: Faintalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. Figura 3. 32: Librería de Arduino para Simulink. Figura 3. 33: Bloques de la librería de Arduino instalada en Matlab Figura 3. 34: Característica del bloque Standard Servo Write. 	42 42 43 44 44 44 45 46 48 48 48 48 48 49 50 51 51
 Figura 3. 19. Paritalia Factil Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. Figura 3. 32: Librería de Arduino para Simulink. Figura 3. 33: Bloques de la librería de Arduino instalada en Matlab Figura 3. 34: Característica del bloque Standard Servo Write. Figura 3. 35: Tarjeta PBC para la actuación de la Plataforma Inercial de 2-GDL. 	42 42 43 44 44 44 45 46 48 48 48 48 48 49 50 51 51 52 53
 Figura 3. 19. Palitalia Factili Resistiva. Figura 3. 20: Tarjeta Arduino DUE. Figura 3. 21: Ubicación de los botones del control físico. Figura 3. 22: Rediseño de la Proforma Inercial 2-GDL. Figura 3. 23: Diagrama de bloques del sistema de control en el Eje X. Figura 3. 24: Diagrama de bloques del sistema de control en el Eje Y. Figura 3. 25: Diagrama de bloques del sistema de control. Figura 3. 26: Dimensiones de la pantalla táctil. Figura 3. 27: Diagrama de conexiones para la obtención de la posición. Figura 3. 29: Diseño del filtro Pasa bajo. Figura 3. 30: Señal PWM filtrado con Pasa Bajo. Figura 3. 31: Servomotor marca Fan Model. Figura 3. 32: Librería de Arduino para Simulink. Figura 3. 33: Bloques de la librería de Arduino instalada en Matlab- Figura 3. 34: Característica del bloque Standard Servo Write. Figura 3. 35: Tarjeta PBC para la actuación de la Plataforma Inercial de 2-GDL. Figura 3. 36: Señal PRBS Servomotor X. 	42 42 43 44 44 44 45 46 48 48 48 48 49 50 51 51 51 52 53 56

Figura 3. 37: Señal PRBS Servomotor Y.	.57
Figura 3. 38: Diagrama de bloques de eje X utilizando los datos del sensor	.57
Figura 3. 39: Filtrado Kalman	.58
Figura 3. 40: Algoritmo de Control Conocido para el Eje X	.59
Figura 3. 41: Ident, sentencia escrita en el comando de Matlab	.60
Figura 3. 42: Estructura del Toolbox Ident.	.60
Figura 3. 43: Ingreso de datos en el Toolbox Ident en el eje X	.61
Figura 3. 44: Importación de datos-ident en el eje X	.61
Figura 3. 45: Procesamiento de señal-Rango en eje X	.62
Figura 3. 46: Selección del rango de operación para la identificación de la señal.	.62
Figura 3. 47: Rango para realizar la identificación del servomotor X	.63
Figura 3. 48: Procesamiento de identificación de la planta utilizando ident	.63
Figura 3. 49: Gráfico de identificación con su respectivo porcentaje de error	.64
Figura 3. 50: Algoritmo del Control Conocido para el Eje Y	.65
Figura 3. 51: Ingreso de datos en el Toolbox Ident en el eje Y	.67
Figura 3. 52: Importación de datos-ident	.67
Figura 3. 53: Procesamiento de señal-Rango	.68
Figura 3. 54: Rango de operación para la identificación de la señal	.68
Figura 3. 55: Rango seleccionado para la identificación del servomotor Y	.69
Figura 3. 56: Procesamiento de la identificación de la planta en el eje Y	.69
Figura 3. 57: Identificación con su respectivo porcentaje de error para el eje Y	.70
Figura 3. 58: Diagrama de control lógico PID en la plataforma inercial de 2-GDL.	.71
Figura 3. 59: Respuesta a una señal escalón del eje X	.72
Figura 3. 60: Valores de la función de transferencia.	.73
Figura 3. 61: Planta del Eje X en Simulink.	.73
Figura 3. 62: Constantes de PID eje de las X.	.74
Figura 3. 63: Respuesta al escalando en lazo cerrado para el eje X	.74
Figura 3. 64: Bloques para el eje de X, con el algoritmo de control PID	.75
Figura 3. 65: Parámetros de control PID en tiempo Real para el eje de las X	.76
Figura 3. 66: Diseño del Algoritmo PID en el eje X.	.77
Figura 3. 67: Respuesta a una señal escalón para el eje Y	.78
Figura 3. 68: Planta del Eje X en Simulink.	.79
Figura 3. 69: Constantes de PID eje de las Y.	.79
Figura 3. 70: Respuesta al escalón en circuito de lazo cerrado para el eje X	.80
Figura 3. 71 Algoritmo de control PID para el eje Y	.81
Figura 3. 72: Parámetros de control PID en tiempo Real para el eje Y	.82
Figura 3. 73: Diseño del Algoritmo PID en el eje Y XIX	.83

Figura 3. 74: Algoritmo de control PID.	84
Figura 3. 75: Señal de Control del PID la Plataforma Inercial de 2-GDL	85
Figura 3. 76: Toolbox Fuzzy de Matlab	85
Figura 3. 77: Métodos de creación de algoritmo FUZZY	86
Figura 3. 78: FUZZY y variables para el eje Y y X	87
Figura 3. 79: Rango de operación de la señal de ERROR del eje X	87
Figura 3. 80: Rango de operación de la señal de D-ERROR del eje X	88
Figura 3. 81: Rango de operación de la señal de CONTROL del eje X	88
Figura 3. 82: Reglas de correspondencia de control FUZZY del eje X	89
Figura 3. 83: Parámetros del Bloque FUZZY en Simulink	90
Figura 3. 84: Diagrama de control lógico FUZZY para el eje de las X	91
Figura 3. 85: Rango de operación de la señal de ERROR del eje Y	92
Figura 3. 86: Rango de operación de la señal de D-ERROR del eje Y	92
Figura 3. 87: Rango de operación de la señal de CONTROL del eje Y	93
Figura 3. 88: Reglas de correspondencia de control FUZZY del eje Y	93
Figura 3. 89: Parámetros del Bloque FUZZY en Simulink	94
Figura 3. 90: Diagrama de control lógico FUZZY para el eje de las Y	95
Figura 3. 91: Algoritmo de control FUZZY	96
Figura 3. 92: Gráficas del Funcionamiento de control FUZZY	97
Figura 3. 93: Movimiento de la esfera sobre pantalla táctil	97
Figura 4. 1: Menú de complementos para Matlab.	99
Figura 4. 2: Paquete de Simulink para Hardware Arduino	99
Figura 4. 3: Credenciales para cuenta de MathWorks	99
Figura 4. 4: Paquete de Simulink para hardware Arduino.	100
Figura 4. 5: Componentes para instalación completa de Arduino IDE	100
Figura 4. 6: Instalación de para comunicar las placas Arduino	100
Figura 4. 7: Conector IEC y luz piloto indicadora de conexión.	101
Figura 4. 8: Activación de botón marcha y luz piloto ON/OFF	101
Figura 4. 9: Conexión de cable USB en la planta y el computador	101
Figura 4. 10: Reconocimiento conexión de una placa Arduino	102
Figura 4. 11: Modelo en blanco de Simulink.	102
Figura 4. 12: Bloque "Analog Input" del paquete de soporte para Arduino	102
Figura 4. 13: Lectura de coordenadas con bloques de entradas analógicas	103
Figura 4. 14: Parámetros de configuración Arduino DUE	103
Figura 4. 15: Selección del modo de simulación.	103
Figura 4. 16: Botón de ejecución en Simulink.	104
Figura 4. 17: Práctica #1 ejecutada en tiempo real.	104
$\mathbf{V}\mathbf{V}$	

Figura 4. 18: Lectura de posición superior derecha en la pantalla	105
Figura 4. 19: Lectura de posición inferior derecha en la pantalla.	105
Figura 4. 20: Lectura de posición inferior izquierda en la pantalla	105
Figura 4. 21: Lectura de posición superior izquierda en la pantalla	106
Figura 4. 22: Simulink modelo en blanco y Librería para Arduino	108
Figura 4. 23: Bloque "Constant" en la librería de Simulink.	108
Figura 4. 24: Modelo para escritura de ángulos al motor-X	108
Figura 4. 25: Bloque "Constant" en la librería de Simulink.	108
Figura 4. 26: Parámetros del Bloque "Constant".	109
Figura 4. 27: Ángulos para motor-X con límites de saturación.	109
Figura 4. 28: Parámetros del Bloque "Sine Wave".	110
Figura 4. 29: Práctica #2 ejecutada en tiempo real	110
Figura 4. 30: Valor de 150º enviado al servomotor	110
Figura 4. 31: Posición del plato a 150º en el eje-X	111
Figura 4. 32: Valor de 90º enviado al servomotor.	111
Figura 4. 33: Posición del plato a 90º en el eje-X	111
Figura 4. 34: Simulink modelo en blanco y Librería para Arduino	113
Figura 4. 35: Bloque "Constant" en la librería de Simulink.	113
Figura 4. 36: Modelo para escritura de ángulos al motor-Y	113
Figura 4. 37: Bloque "Constant" en la librería de Simulink.	113
Figura 4. 38: Parámetros del Bloque "Constant"	114
Figura 4. 39: Ángulos para el motor-Y con límites de saturación.	114
Figura 4. 40: Parámetros del Bloque "Sine Wave"	114
Figura 4. 41: Práctica #3 ejecutada en tiempo real.	115
Figura 4. 42: Valor de 80º enviado al servomotor.	115
Figura 4. 43: Posición del plato a 80º en el eje-Y	115
Figura 4. 44: Valor de 20º enviado al servomotor.	116
Figura 4. 45: Posición del plato a 20º en el eje-Y	116
Figura 4. 46: Simulink modelo en blanco y Librería para Arduino	118
Figura 4. 47: Bloque "Data conver X" con la variable "doublé"	118
Figura 4. 48: Bloque "Constant" en la librería de Simulink.	119
Figura 4. 49: Parámetros del Bloque "Constant"	119
Figura 4. 50: Bloque de saturación para el servomotor X	119
Figura 4. 51: Bloque "Kalman Filter" en la librería de Simulink	120
Figura 4. 52: Parametrización para el bloque "Kalman Filter"	120
Figura 4. 53: Filtrado de la señal de la pantalla táctil resistiva	121
Figura 4. 54: Bloque "Discrete PID Controller".	121
1/1/1	

Figura 4. 55: Bloque "Constant" para la referencia del controlador PID	122
Figura 4. 56: Algoritmo de control PID para el eje X	122
Figura 4. 57: Parametrización del bloque PID para el eje X	123
Figura 4. 58: Bloque "Scope" para el eje X.	123
Figura 4. 59: Algoritmo de control PID para el eje X con el bloque "SCOPE"	124
Figura 4. 60: Variable Ts en la ventada de comando de Matlab	124
Figura 4. 61: Algoritmo de control PID para el eje X en tiempo real	124
Figura 4. 62: Posición del objeto en la referencia 550 para el eje X	125
Figura 4. 63: Bloque "SCOPE" utilizado en sistema SISO para el eje X	125
Figura 4. 64: Simulink modelo en blanco y Librería para Arduino	128
Figura 4. 65: Bloque "Data conver Y" con la variable "doublé"	128
Figura 4. 66: Bloque "Constant" en la librería de Simulink	129
Figura 4. 67: Parámetros del Bloque "Constant"	129
Figura 4. 68: Bloque de saturación para el servomotor Y	129
Figura 4. 69: Bloque "Kalman Filter" en la librería de Simulink	130
Figura 4. 70: Parametrización para el bloque "Kalman Filter"	130
Figura 4. 71: Filtrado de la señal de la pantalla táctil resistiva	131
Figura 4. 72: Bloque "Discrete PID Controller".	131
Figura 4. 73: Bloque "Constan" para la referencia del controlador PID	132
Figura 4. 74: Algoritmo de control PID para el eje Y	132
Figura 4. 75: Parametrización del bloque PID para el eje Y	133
Figura 4. 76: Bloque "Scope" para el eje Y	133
Figura 4. 77: Algoritmo de control PID para el eje Y con el bloque "SCOPE"	'134
Figura 4. 78: Variable Ts en la ventada de comando de Matlab	134
Figura 4. 79: Algoritmo de control PID para el eje Y en tiempo real	134
Figura 4. 80: Posición del objeto en la referencia 550 para el eje delas Y	135
Figura 4. 81: Bloque "SCOPE" utilizado en sistema SISO para el eje Y	135
Figura 4. 82: Simulink modelo en blanco y Librería para Arduino	138
Figura 4. 83: Bloque "Data conver" con la variable "doublé"	139
Figura 4. 84: Bloque "Constant" en la librería de Simulink	139
Figura 4. 85: Parámetros del Bloque "Constant"	139
Figura 4. 86: Bloque de saturación para los servomotores X y Y	140
Figura 4. 87: Bloque Kalman Filter en la librería de Simulink	140
Figura 4. 88: Parametrización para el bloque "Kalman Filter"	141
Figura 4. 89: Filtrado de las señales de la pantalla táctil resistiva	141
Figura 4. 90: Bloque "Discrete PID Controller"	141
Figura 4. 91: Bloque "Constant" para la referencia del controlador PID XXII	142

Figura 4. 92: Algoritmo de control PID para la plataforma inercial de 2-GDL142
Figura 4. 93: Parametrización del bloque PID para el eje X y Y143
Figura 4. 94: Bloque "Sine Wave" para la creación de la señal circular143
Figura 4. 95: Parametrización de los bloques "Sine Wave"144
Figura 4. 96: Algoritmo de control PID para la plataforma inercial de 2-GDL144
Figura 4. 97: Bloques "Scope" para el eje X y Y145
Figura 4. 99: Variable Ts en la ventada de comando de Matlab145
Figura 4. 100: Algoritmo de control PID para el eje de las X y Y en tiempo real145
Figura 4. 101: Posición del objeto en la referencia 550 para el eje de las X y Y146
Figura 4. 102: Bloque "SCOPE" para el eje de las X y Y146
Figura 4. 103: Algoritmo de control PID para el seguimiento de trayectoria147
Figura 4. 104: Grafica del seguimiento de la esfera147
Figura 4. 105: Bloque "SCOPE" para el eje de las X y Y148
Figura 4. 106: Simulink modelo en blanco y Librería para Arduino150
Figura 4. 107: Bloque "Data conver X" con la variable "doublé"150
Figura 4. 108: Bloque "Constant" en la librería de Simulink
Figura 4. 109: Parámetros del Bloque "Constant"151
Figura 4. 110: Bloque de saturación para el servomotor X151
Figura 4. 111: Bloque Kalman Filter en la librería de Simulink152
Figura 4. 112: Parametrización para el bloque "Kalman Filter"152
Figura 4. 113: Filtrado de la señal de la pantalla táctil resistiva153
Figura 4. 114: Bloque "FUZZY Logic Controller"153
Figura 4. 115: Bloque "Constan" para la referencia del controlador FUZZY154
Figura 4. 116: Algoritmo de control PID para el eje X
Figura 4. 117: Ventana de comando para llamar al Toolbox Fuzzy155
Figura 4. 118: Método Mamdani para la creación del algoritmo FUZZY155
Figura 4. 119: FUZZY para el eje X156
Figura 4. 120: Funciones de membresía para las variables de entrada156
Figura 4. 121: Funciones de membresía para las variables de entrada157
Figura 4. 122: Reglas de correspondencia para el controlador FUZZY del eje X.157
Figura 4. 123: Parametrizar el Bloque FUZZY en Simulink158
Figura 4. 124: Controlador FUZZY para el eje X158
Figura 4. 125: Bloque "Scope" para el eje X159
Figura 4. 126: Algoritmo de control FUZZY para el eje X159
Figura 4. 127: Variable Ts en la ventada de comando de Matlab159
Figura 4. 128: Algoritmo de control FUZZY para el eje X en tiempo real160
Figura 4. 129: Posición del objeto en la referencia 550 para el eje X160 XXIII

Figura 4. 130:	Bloque "SCOPE" utilizado en sistema SISO para el eje X	161
Figura 4. 131:	Simulink modelo en blanco y Librería para Arduino	163
Figura 4. 132:	Bloque "Data conver Y" con la variable "doublé"	163
Figura 4. 133:	Bloque "Constant" en la librería de Simulink.	164
Figura 4. 134:	Parámetros del Bloque "Constant".	164
Figura 4. 135:	Bloque de saturación para el servomotor Y	164
Figura 4. 136:	Bloque Kalman Filter en la librería de Simulink	165
Figura 4. 137:	Parametrización para el bloque "Kalman Filter"	165
Figura 4. 138:	Filtrado de la señal de la pantalla táctil resistiva	166
Figura 4. 139:	Bloque "FUZZY Logic Controller".	166
Figura 4. 140:	Bloque "Constan" para la referencia del controlador FUZZY	167
Figura 4. 141:	Algoritmo de control PID para el eje Y	167
Figura 4. 142:	Ventana de comando para llamar al Toolbox Fuzzy	168
Figura 4. 143:	Método Mamdani para la creación del algoritmo FUZZY	168
Figura 4. 144:	FUZZY para el eje Y.	169
Figura 4. 145:	Funciones de membresía para las variables de entrada	169
Figura 4. 146:	Funciones de membresía para las variables de entrada	170
Figura 4. 147:	Reglas de correspondencia para el controlador FUZZY del eje Y.	170
Figura 4. 148:	Parametrizar el Bloque FUZZY en Simulink	171
Figura 4. 149:	Controlador FUZZY para el eje Y	171
Figura 4. 150:	Bloque "Scope" para el eje Y.	172
Figura 4. 151:	Algoritmo de control FUZZY para el eje Y	172
Figura 4. 152:	Variable Ts en la ventada de comando de Matlab	172
Figura 4. 153:	Algoritmo de control FUZZY para el eje Y en tiempo real	173
Figura 4. 154:	Posición del objeto en la referencia 550 para el eje Y	173
Figura 4. 155:	Bloque "SCOPE" utilizado en sistema SISO para el eje Y	174
Figura 4. 156:	Simulink modelo en blanco y Librería para Arduino	176
Figura 4. 157:	Bloque "Data conver" con la variable "doublé	177
Figura 4. 158:	Bloque "Constant" en la librería de Simulink	177
Figura 4. 159:	Parámetros del Bloque "Constant"	177
Figura 4. 160:	Bloque de saturación para los servomotores X y Y	178
Figura 4. 161:	Bloque Kalman Filter en la librería de Simulink	178
Figura 4. 162:	Parametrización para el bloque "Kalman Filter"	179
Figura 4. 163:	Filtrado de las señales de la pantalla táctil resistiva	179
Figura 4. 164:	Bloque "FUZZY Logic Controller".	180
Figura 4. 165:	Bloque "Constant" para la referencia del controlador PID	180
Figura 4. 166:	Algoritmo de control FUZZY	181
	XXIV	

Figura 4. 167: Ventana de comando para llamar al Toolbox Fuzzy	.181
Figura 4. 168: Método Mamdani para la creación del algoritmo FUZZY	.182
Figura 4. 169: FUZZY para el eje X y Y.	.182
Figura 4. 170: Funciones de membresía para FuzzyX	.183
Figura 4. 171: Funciones de membresía para FuzzyY	.183
Figura 4. 172: Funciones de membresía para las variables de entrada	.184
Figura 4. 173: Funciones de membresía para FuzzyY	.184
Figura 4. 174: Reglas de correspondencia para el controlador FUZZY	.185
Figura 4. 175: Parametrizar los Bloque FUZZY en Simulink.	.185
Figura 4. 176: Bloque FUZZY en Simulink	.186
Figura 4. 177: Bloque "Sine Wave" para la creación de la señal circular	.186
Figura 4. 178: Parametrización de los bloques "Sine Wave"	.187
Figura 4. 179: Algoritmo de control FUZZY con el seguimiento de trayectoria	.187
Figura 4. 180: Bloques "Scope" para el eje X y Y	.188
Figura 4. 181: Algoritmo de control FUZZY para el eje X y Y	.188
Figura 4. 182: Variable Ts en la ventada de comando de Matlab	.188
Figura 4. 183: Algoritmo de control FUZZY para el eje de las X y Y	.189
Figura 4. 184: Posición del objeto en la referencia 550 para el eje de las X y Y.	.189
Figura 4. 185: Bloque "SCOPE" para el eje de las X y Y	.190
Figura 4. 186: Algoritmo de control FUZZY para el seguimiento de trayectoria	.190
Figura 4. 187: Grafica del seguimiento de la elíptica	.190
Figura 4. 188: Bloque "SCOPE" para el eje de las X y Y	.191
Figura 5. 1: Plataforma Inercial de 2-GDL	.192
Figura 5. 2: Tarjeta PCB instalada.	.192
Figura 5. 3: Panel de control.	.193
Figura 5. 4: Análisis de la respuesta en tiempo real del PID en el eje X	.194
Figura 5. 5: Análisis de la respuesta en tiempo real del PID en el eje Y	.194
Figura 5. 6: Algoritmo de control PID en tiempo real.	.195
Figura 5. 7: Análisis de la respuesta en tiempo real del FUZZY en el eje X	.195
Figura 5. 8: Análisis de la respuesta en tiempo real del FUZZY en el eje Y	.196
Figura 5. 9: Algoritmo de control FUZZY en tiempo real.	.196
Figura 5. 10: Seguimiento de trayectoria con el controlador PID	.197
Figura 5. 11: Seguimiento de trayectoria con el controlador FUZZY	.197
Figura 6. 1: III jornada científica de electrónica y automatización 2019	.199
Figura 6. 2: Análisis de resultados.	.199
Figura 6. 3: Formato de encuesta.	.200

ÍNDICE DE ANEXOS

ANEXO 1: TABLAS DE DATOS GENERADOS DE LA SEÑAL PRBS	207
ANEXO 2: CRONOGRAMA DE ACTIVIDADES	212
ANEXO 3: PRESUPUESTO DEL PROYECTO DE TITULACIÓN	213
ANEXO 4: FACTURAS DE EQUIPOS Y ELEMENTOS VARIOS	213
ANEXO 5: DATASHEET PANTALLA TACTIL RESISTIVA	215

INTRODUCCIÓN

Hoy en día existen varias aplicaciones industriales que tienen el movimiento como principal condición de su entorno, estas pueden operar sobre, vehículos aéreos tripulados y no tripulados, vehículos terrestres, armamento militar inteligente, robots humanoides e inclusive transbordadores si se quiere hablar de naves espaciales, dichas aplicaciones trabajan con los llamados sistemas inerciales montados como dispositivos o módulos de control. Estos dispositivos también conocidos como plataformas inerciales son colocados en el sistema electrónico que en esencia miden la posición espacial y velocidad a la que se desplaza el cuerpo.

En el presente documento se realiza la repotenciación de la plataforma inercial que se encuentra ubicada en el laboratorio de control automático del bloque 'E' de la Universidad Politécnica Salesiana sede Guayaquil campus Centenario, la cual cuenta con dos grados de libertad.

En búsqueda de tener un módulo didáctico con elementos de bajo costo, a la plataforma inercial de 2-GDL se le realiza una repotenciación general que implica el rediseño de su estructura mecánica utilizando acero inoxidable y cambio de sus dispositivos de control utilizando nuevas tarjetas electrónicas. Esta repotenciación se inicia con el diagnóstico de sus partes mecánicas y electrónicas, ya que la plataforma cuenta con elementos que necesitan mantenimiento preventivo o correctivo.

Mediante el diagnóstico de los elementos de la plataforma inercial de 2-GDL, se tiene una idea contundente sobre qué elementos electrónicos se deben reemplazar, sea que se encuentren defectuosos o descontinuados. Y que elementos mecánicos se deben rediseñar para afinar el movimiento de los ejes en la plataforma.

Tomando en cuenta que los estudiantes deben reforzar sus conocimientos prácticos del control automático se debe tener una guía práctica fundamentada con los controladores PID y Fuzzy. Además de reforzar los conocimientos prácticos, esta guía es la evaluación final para la plataforma inercial de 2-GDL.

1. EL PROBLEMA

1.1 Descripción del problema

Con el ánimo de realizar un aporte a la carrera de ingeniería electrónica, se determina la necesidad de brindar a los estudiantes la posibilidad de desarrollar en forma práctica lo aprendido en las materias relacionadas a la teoría de control, por esto se plantea repotenciar el laboratorio de control automático del bloque 'E' de la Universidad Politécnica Salesiana sede Guayaquil campus Centenario, con la mejora de la plataforma inercial de dos grados de libertad, que se encuentra en dicho lugar.

1.2 Antecedentes

Debido a que existen muchos principios de medición inercial y avances en la electrónica, se tiene la necesidad del estudio practico en un sistema inercial, para ello, los estudiantes de la carrera de ingeniería electrónica de la Universidad Politécnica Salesiana sede Guayaquil campus Centenario, tienen a disposición la plataforma inercial de dos grados de libertad, en el laboratorio de control automático del bloque 'E'.

En vista que esta plataforma representa un gran aporte para los estudiantes, se propone este proyecto para repotenciar dicho equipo en sus partes mecánicas y electrónicas mejorando la cinética y comunicación con el software Matlab. Además, no se cuenta con una guía de prácticas, la cual es de vital importancia para evaluar dicha repotenciación.

1.3 Importancia y alcance

El control automático ha desempeñado un papel vital en el avance de la ingeniería y la ciencia, incluso se ha convertido en una parte importante y cuasi obligatorio en los sistemas electrónicos de vehículos, terrestres, aéreos, espaciales, robots inteligentes, procesos modernos de fábricas (transporte, llenado y empaquetado del producto), en cualquier operación industrial que requiera el control de temperatura, presión, humedad, flujo, etc. Por esto es fundamental que la mayoría de los ingenieros y científicos estén familiarizados con la teoría y la práctica del control automático. (Ogata, 2010)

En el laboratorio de control automático existen los equipos necesarios para reforzar el conocimiento adquirido en las materias de teoría de control. Uno de ellos es la plataforma inercial de 2-GDL, que con su repotenciación y desarrollo de una guía práctica fundamentada con el diseño y evaluación de controladores PID y Fuzzy, son la ayuda que los futuros ingenieros electrónicos necesitan para elegir el mejor control al momento de encontrarse frente a situaciones reales dentro de su respectivo campo laboral.

1.4 Delimitación

1.4.1 Temporal

El trabajo de titulación es realizado entre los años 2017 al 2019.

1.4.2 Espacial

El trabajo de titulación está destinado para su uso en el laboratorio de control automático que está ubicado en el tercer piso del bloque 'E' de la Universidad Politécnica Salesiana sede Guayaquil campus Centenario con dirección Chambers #277 entre Laura Vicuña y Robles.

1.4.3 Académico

La repotenciación de este proyecto se enfoca en el desarrollo práctico de lo aprendido en las materias relacionadas con la teoría de control. La plataforma inercial de 2-GDL está delimitada académicamente para los estudiantes de la carrera de Ingeniería Electrónica. Se realiza una guía que consta de 9 prácticas didácticas para evaluar el funcionamiento de la plataforma inercial de 2-GDL. Además, los futuros estudiantes egresados pueden realizar estudios de investigación para utilizar otros controladores con el fin de evaluar su rendimiento y facilidad de aplicación.

1.5 Objetivos

1.5.1 Objetivo General

Repotenciar la plataforma inercial de 2-GDL para su óptimo desempeño y realizar una guía de prácticas para las materias de Teoría de Control I, II y III utilizando los controladores PID y FUZZY.

1.5.2 Objetivos Específicos

- Diagnosticar el estado de las partes mecánicas y electrónicas de la plataforma inercial para su correcto funcionamiento.
- Reemplazar los elementos defectuosos o descontinuados que se encuentren en la plataforma de acuerdo al diagnóstico inicial.
- Definir el modelo del sistema, empleando herramientas de identificación con la ayuda del Software Matlab (Ident).
- Implementar una guía de prácticas utilizando los controladores PID y FUZZY aplicados en la plataforma inercial.

2. ESTADO DEL ARTE

2.1 Sistemas inerciales

El grupo de cuerpos respecto de otro sistema los cuales se puede describir el movimiento, se llama *"sistema de referencia"*, y los sistemas con el objetivo de que la ley de inercia se verifique en ellos se llaman *"sistemas inerciales"*. Es sencillo ver que cualquier sistema que se mueva constantemente en relación con otro grupo de cuerpos es también llamado sistema inercial. Al considerar el libre movimiento de un cuerpo no se diferencia entre varios sistemas inerciales. La experiencia demuestra que todas las leyes de la mecánica son equivalentes en cada sistema inercial, y esto se conoce como *el principio de la relatividad de Galileo*. (Roederer, 2002)

2.2 Plataforma Inercial

La estabilización de las plataformas inerciales ha tenido mucho interés durante bastante tiempo en aplicaciones aeronáuticas y en navegación marítimas. Regularmente, este tipo de plataformas están situadas en un eje que tiene una específica orientación variable con respecto a la tierra. Estas variaciones de orientación del eje son detectadas por sensores giroscópicos situados hacia un extremo de la plataforma. Entre algunos modelos de ejemplo se tiene, el control automático de aeronaves, el dispositivo que controla la dirección de un torpedo, el equilibrio de la mira para rastreo visual, sistemas con movimiento planetario traslacional, el control de robots humanoides y entre otros diversos dispositivos. En cada uno de estos casos, el ajuste inercial requiere una realimentación de un sensor giroscópico situado en la plataforma móvil, que por regla general proporciona la velocidad de un cuerpo con respecto al marco inercial. (Rubio, Ortega, & Gordillo, 2010)

2.2.1 Grados de Libertad

El concepto de grados de libertad o GDL (degrees of freedom o DOF en inglés) es pobremente explicado en textos de investigación, esto puede advertir que se trata de un tema difícil de comprender y de esta manera ser pasado por alto la posterior revisión del estudiante. El significado de grados de libertad se puede comprender desde un punto de vista geométrico, algebraico e inclusive intuitivo. La geometría nos detalla a los grados de libertad como espacios e hiperespacios de libertad a través de los cuales una medida de resumen puede moverse y tomar diferentes valores. El punto de vista algebraico los describe como el número de ecuaciones que se crean usando los datos. Ambos puntos de vista se relacionan y ayudan a entender con mayor fondo el concepto de grados de libertad.

El número de grados de libertad se comprende mejor si es visto como el número de dimensiones espaciales en los que un punto es libre de moverse. Cada relación que se establece hace que se tengan que modificar los grados de libertad si va a ser usado en cálculos futuros. (Cruz-Oré & Luis, 2013)



Figura 2. 1: Grados de libertad en la visión de una persona. (Idis, 1996)

2.3 Servomotores

El servomotor es un dispositivo electromecánico que básicamente contiene un motor eléctrico, un juego de engranes, potenciómetro y una tarjeta de control, todo dentro de una carcasa plástica o de metal. Un servo tiene la cualidad de ser controlado en posición y capaz de ubicarse en cualquier ángulo dentro de un rango de operación generalmente de 180º pero puede ser fácilmente modificado para tener un giro libre de 360º. Estos suelen ser utilizados en robótica, automatismo y modelismo (vehículos por radio-control) debido a su gran precisión en el posicionamiento. (Mecafenix, 2017)



Figura 2. 2: Servomotor DC y Servomotor AC. (Direct Industry, 2019)

Los servomotores funcionan por medio de modulación de ancho de pulso (PWM). Todos los servos disponen de tres cables, dos para alimentación Vcc y Gnd (4.8 hasta 7.4 [V]) y un tercero para aplicar el tren de pulsos de control, que hace que el circuito de control diferencial interno ponga el servo en la posición indicada. La frecuencia usada para mandar la secuencia de pulsos al servomotor es de 50 Hz esto significa que cada ciclo dura 20 ms, las duraciones de cada pulso se interpretan como comandos de posicionamiento del motor, mientras que los espacios entre cada pulso son despreciados. (Mecafenix, 2017)



Figura 2. 3: Variación de ángulos en el servomotor deacuerdo a su señal PWM. (Mecafenix, 2017)

2.3.1 Partes de un servomotor

En general, los servos suelen estar compuestos por 4 elementos fundamentales:

• Motor de corriente continua (DC)

Es el componente que le entrega movilidad al servo. Cuando se aplica una diferencia de potencial a sus dos terminales, este motor gira en un sentido a su velocidad máxima. Si el potencial aplicado sus dos terminales es inverso, el sentido de giro también se invierte.

• Engranajes reductores

El conjunto de engranajes se encarga de reducir la alta velocidad de giro del motor para acrecentar su capacidad de torque o par-motor.

• Sensor de desplazamiento

Normalmente es un potenciómetro ubicado en el eje de salida que se utiliza para conocer la posición angular del servomotor.

• Circuito de control

Es la placa electrónica que efectúa una táctica de control para la posición por realimentación. Para ello, este circuito compara la señal de entrada de referencia (posición deseada) con la posición actual medida por el potenciómetro. La diferencia entre la posición actual y la deseada es amplificada y utilizada para mover el motor en la dirección necesaria para reducir el error. (Mecafenix, 2017)



Figura 2. 4: Partes de un servomotor DC. (Mecafenix, 2017)

El servomotor tiene 3 cables, Vcc, Gnd y la señal. Los colores son los siguientes:

- Rojo Vcc (4.8 hasta 7.4 [V])
- Marrón Gnd
- Naranja Señal de control (PWM)



Figura 2. 5: Conector del servomotor con sus 3 cables, Vcc, GND y Señal.

2.4 Panel Táctil Resistivo

Las pantallas táctiles dan una posición precisa del punto que se presiona sobre ellas. Según el proceso utilizado en su montaje, hay varios tipos. A pesar de existir las pantallas táctiles por IR, también se tiene hoy en día las capacitivas, de onda acústica superficial (SAW) y resistivas. Las más utilizadas y que se presenta en este proyecto de titulación son los de tipo resistivo. Su destacado funcionamiento junto con un costo muy competitivo y un manejo simple las convierte en piezas significativamente importantes y de fácil acceso para todo tipo de trabajo que conlleve la necesidad de obtener una posición. (Echeverría, Angulo, & Angulo, 2008)



Figura 2. 6: Pantalla táctil de vidrio o touch panel, compuesta de varias capas. (Alcívar Cevallos & Parrales Duarte, 2015)

2.4.1 Panel táctil resistivo de 4 cables

Las pantallas táctiles resistivas se forman de dos capas de un material conductor (óxido de indio y estaño) que tienen una resistencia contra el flujo eléctrico que es proporcional a la longitud de este material conductor. Una de las capas se usa para obtener la posición del punto en el eje X y la otra en el eje Y. Como se muestra en la siguiente figura, una de las capas tiene dos barras del material conductor en los lados paralelos asociadas con los pines X+ y X-, mientras que la otra capa tiene otras dos barras en la parte superior e inferior de la pantalla, que están asociadas con los pines Y + e Y -. (Echeverría et al., 2008)



Figura 2. 7: Distribución de pines de conexión para el panel táctil resistivo. (Echeverría et al., 2008)

A medida que el toque o el "contacto" en la pantalla se hace hacia un lado o el otro en sentido horizontal, el voltaje incrementa o disminuye en X+. El voltaje en Y+ de igual forma incrementa o disminuye a medida que el "contacto" se hace más abajo o arriba en el sentido vertical. Dicho esto, cuando se conecta una diferencia de potencial entre Y+ e Y-, el pin X+ da un voltaje correspondiente a la posición del "contacto" en sentido horizontal. Lo equivalente ocurre con el pin Y+, en sentido vertical, cuando hay una diferencia de potencial conectado entre X+ y X-. (Echeverría et al., 2008)



Figura 2. 8: Arreglo de resistencias en el panel táctil resistivo.

2.5 Tarjeta de circuito impreso o PCB

Una tarjeta de circuito impreso o PCB en inglés (Printed Circuit Board), también llamada baquelita o baquela, es una placa o lámina que se usa para conectar los distintos elementos que conforman el circuito y es el núcleo de la mayoría de dispositivos en la tecnología como un teléfono móvil, una CPU, un robot, el sistema electrónico de un vehículo.

Los circuitos impresos son muy utilizados en el diseño electrónico, mecatrónica, eléctrica, producción y manufactura de investigaciones innovadoras. Hoy en día todos los productos electrónicos tienen dentro de sí, este tipo de tarjetas con diferentes formas, características, tamaño, componentes y colores. (Aparicio, 2010)


Figura 2. 9: Tarjeta de circuito impreso. (Castillo, 2017)

2.5.1 Relé o Relevador

Es un dispositivo eléctrico que funciona como un interruptor, abriendo y cerrando la entrada del flujo eléctrico, sin embargo, funciona eléctricamente. El relé permite que los contactos se abran o cierren por medio de un electroimán, es la razón por la que también se les llama relevador o relés electromagnéticos. En la figura esclarece su funcionamiento.(Harper, 2005)



Figura 2. 10: Mecanismo de un relé y relé tipo electromagnético. (Área Tecnología, 2016)

2.5.2 Borneras y Espadines

La Bornera de dos Pines es un componente básico que no debe faltar a la hora de diseñar tus circuitos impresos, este componente permite conectar una alimentación externa con la sujeción rápida que posee en sus entradas delanteras, las cuales están hechas de metal para la conductividad. La bornera es compatible no solo con circuitos impresos sino también con protoboards. (Castillo, 2017)

Una tira de pines o espadines hembra como su nombre lo indica es una tira que posee los pines de acuerdo a la necesidad tal y como se ve en la figura ya sean de tres o cuatro pines, estas tiras son muy útiles cuando se quiere realizar conexiones en placas impresas, la ventaja de estas tiras es que se pueden soldar a una placa y se puede conectar a otro dispositivo con ayuda de jumpers machos. (Castillo, 2017)



Figura 2. 11: Borneras, espadines hembra y espadines macho. (Castillo, 2017)

2.5.3 Filtro RC Pasa bajo

Son aquellos que aplican muy poca atenuación a la frecuencia que es más baja que una específica, llamada la frecuencia de corte. Las frecuencias que sobrepasan la frecuencia de corte se filtran y son atenuadas fuertemente. Dado que la reactancia capacitiva disminuye con la frecuencia, el circuito RC mostrado discrimina estas altas frecuencias. (Ibarra & López, 2007)



Figura 2. 12: Circuito RC y su gráfica en función de la frecuencia. (Olmo Nave, 2017)

2.6 Modulación por ancho de pulso o PWM

Modulación por ancho de pulso o PWM en inglés (Pulse Width Modulation). Es un tipo de señal de voltaje que se usa en la electrónica con una amplia gama de aplicaciones y para distintas tareas. En el entorno se encuentran varios dispositivos que utilizan PWM para realizar alguna actividad. (Tomasi, 2003)

2.6.1 Cómo funciona la modulación por ancho de pulso

La PWM está formada por una señal de onda cuadrada que generalmente no tiene una relación similar entre el tiempo en que es alta y el tiempo en que es baja. En la figura se ve una señal que cambia entre 5 voltios y 0 voltios. La señal fluctúa cierto tiempo entre estos dos valores de voltaje. Durante un tiempo específico, la señal está en alto (5 voltios) y para otro período de tiempo está en bajo, el segundo valor de voltaje (0 voltios). (Rashid, 2004)



Figura 2. 13: Porcentajes diferentes del ciclo de trabajo en la señal PWM. (Gómez, 2017)

El tiempo que la señal está en alto (5 voltios) se lo denomina (Ton), mientras que el tiempo que la señal está en bajo se lo denomina (Toff). El tiempo completo sumado de alto y bajo es el periodo de la señal (T). Y como en cualquier señal periódica, el inverso del período (1 / T), es la frecuencia de esta. La PWM funciona cambiando su voltaje entre estos dos valores conocidos, por ejemplo, Vcc y GND en periodos de tiempo específicos y con una frecuencia fija. (Rashid, 2004)

2.6.2 Ciclo de trabajo o Duty Cycle

La variación la señal PWM comprende diferentes tiempos de encendido y apagado, es decir, Ton y Toff. Al cambiar el valor de un PWM, realmente se están ajustando estos tiempos. Uno de los atributos más significativos de una señal PWM es su ciclo de trabajo o Duty Cycle, en inglés, ya que este es el que fluctúa en un PWM. El ciclo de trabajo no es otra cosa que la relación entre el tiempo que está alto y el periodo o tiempo total de la señal PWM. (Rashid, 2004)

$$(1) \qquad DC = \frac{T_{on}}{T}$$

Cuanto mayor sea el Duty Cycle, más extendido será la señal de voltaje alto, sin cambiar el período. Por lo tanto, si el período ni la suma de Ton y Toff cambian, el tiempo de voltaje bajo disminuye cuando el tiempo de voltaje alto aumenta. Esta es la razón por la que se llama modulación por ancho de pulso, dado que realmente es el ancho del pulso el cual se modifica. (Rashid, 2004)

Normalmente, el valor del ciclo de trabajo se expresa en porcentaje y para eso solo se necesita multiplicar por 100 la ecuación (1).

$$(2) \qquad DC = \frac{T_{on}}{T} * 100$$

Si el ciclo de trabajo es del 90%, como en la siguiente imagen, la señal se encuentra en un estado alto durante el 90% del período. PWM de 90%.



Figura 2. 14: Ciclo de trabajo del 90% en la señal PWM. (Gómez, 2017)

Si el ciclo de trabajo es del 20%, como en la siguiente imagen, la señal será del 20% del tiempo en estado alto y el 80% del tiempo en estado bajo. PWM de 20%.



Figura 2. 15: Ciclo de trabajo del 20% en la señal PWM. (Gómez, 2017)

2.6.3 Usos para la señal PWM

Con lo antes mencionado se sabe que la PWM es una señal que fluctúa el ancho de su pulso y se basa en el parámetro que se llama, ciclo de trabajo. Lo que no se ha discutido aún es para lo que nos sirve la modulación por ancho de pulso en la práctica. Lo cierto es que cambiando el ciclo de trabajo de una señal PWM, lo que se hace es cambiar su voltaje promedio y esta es la clave por la cual se utiliza en muchas ocasiones el PWM. Cuando una señal de voltaje promedio atraviesa ciertos elementos electrónicos, puede hacer que cambie su conducta. Por ejemplo, LEDs, motores de corriente continua o ventiladores, incluidos altavoces y timbres. (GOILAV & LOI, 2016)



Figura 2. 16: Control de brillo en un diodo LED con la señal PWM. (Gómez, 2017)

En el caso que se tenga un LED conectado con un microcontrolador, se puede cambiar el brillo con el que se enciende el LED variando la señal PWM a la que está sometido. Otra opción puede ser controlar la velocidad de un motor de CC, por ejemplo, aquellas PC que utilizan ventiladores, a pesar del hecho que tiende a ser utilizado con cualquier motor de CC. Cuando el ciclo de trabajo varía, la velocidad

cambia a medida que cambia el voltaje promedio en el bobinado del motor. Con un ciclo de trabajo del 100%, el motor girará a toda velocidad. Al disminuir el ciclo de trabajo, se reducirá la velocidad. Otra gran ventaja es enviar una señal PWM que cambia su ciclo de trabajo a un altavoz o timbre para crear sonidos y canciones de manera efectiva y rápida. (GOILAV & LOI, 2016)

2.7 HARDWARE Y SOFTWARE - ARDUINO

Arduino es una plataforma para desarrollo de código abierto que depende de equipos y programación sencillos de utilizar. Las tarjetas de Arduino pueden leer en sus entradas, luz de un sensor, la pulsación de un botón o hasta un mensaje de Twitter y transformarlo en una señal de salida para: activar un motor, encender un LED, publicar algo en la web. Se puede indicar a la tarjeta lo que haga enviando instrucciones al microcontrolador. Para hacer esto, se usa el lenguaje de programación Arduino (basado en C++) y el software de Arduino, que es un entorno de desarrollo integrado o IDE en inglés (Integrated Development Environment). (Artero, 2013)

2.7.1 ARDUINO NANO

El hardware Arduino Nano utiliza el mismo microcontrolador del Arduino Uno, que es el ATmega328, sin embargo, en su versión SMD que disminuye su tamaño en gran medida. Tiene 14 pines de entrada/salida, 6 de ellos son PWM y tiene 8 entradas analógicas. (Artero, 2013)



Figura 2. 17: Tarjeta Arduino NANO. (El Arduino, 2019)

Características técnicas del Arduino NANO:

- Microcontrolador: ATMega328
- Voltaje de operación: 5V
- Voltaje de alimentación (Recomendado): 7-12V
- I/O Digitales: 14 (6 son PWM)
- Entradas Analógicas: 8
- Memoria Flash: 32KB
- EEPROM: 1KB
- Velocidad del reloj: 16MHz
- Dimensiones: 0.73" x 1.70"



Figura 2. 18: Diagrama pines del Arduino NANO. (El Arduino, 2019)

2.7.2 ARDUINO DUE

El hardware Arduino DUE utiliza el microcontrolador Atmel SAM3X8E ARM Cortex-M3. Es el primer Arduino basado en un microcontrolador ARM de 32 bits y tiene 54 pines de entrada/salida digital (de todos estos, 12 se pueden utilizar como salidas PWM), 12 entradas análogas, 4 UARTs (hardware), un reloj de 84MHz, USB OTG, 2 DACs, 2 TWI, conector para alimentación de voltaje, cabezal SPI, cabezal JTAG, botón de borrado y botón de reset. (Millahual, 2017)



Figura 2. 19: Puertos de programación en Arduino DUE. (Delgado Crespo, 2016)

Advertencia: En contraste con otras placas, la Arduino DUE funciona con 3.3V. El máximo voltaje que los pines de entrada/salida pueden tolerar es 3.3V, proporcionarles un voltaje mayor, como 5V puede dañar la placa. Compatible con las Shields que funcionan con 3.3V. (Millahual, 2017)

La Arduino DUE supera las placas con microcontroladores de 8 bits ya que tiene un núcleo ARM de 32 bits. Las diferencias más marcadas son:

- Permite hacer operaciones en datos de 32 bits en cada ciclo de reloj.
- Reloj para la CPU de 84Mhz
- Memoria SRAM de 96 KB
- Memoria Flash de programa de 512 KB
- Controlador DMA, libera la CPU de acceso a memoria a las multitareas intensivas

Características técnicas del Arduino DUE:

- Microcontrolador: AT91SAM3X8E
- Voltaje de Operación: 3.3V
- Voltaje de Entrada (recomendado): 7-12V
- Voltaje de Entrada (límites): 6-20V
- Pines de E/S Digital: 54 (de los cuales 12 proveen salida PWM)
- Pines de Entrada Análoga: 12
- Pines de Salida Análoga: 2 (DAC)
- Corriente DC Total de Salida en todas las líneas de E/S: 130 mA
- Corriente DC para pin 3.3V: 800 mA
- Corriente DC para pin 5V: 800 mA

- Memoria Flash: 512 KB toda disponible para las aplicaciones de usuario
- SRAM: 96 KB (dos bancos: 64KB y 32KB)
- Velocidad de Reloj: 84 MHz
- Dimensiones: 2.10" x 3.99"



Figura 2. 20: Diagrama pines del Arduino DUE. (Naylamp, 2018)

• Entradas y Salidas

Entradas / Salidas digitales de los pines 0 al 53. Cada uno de los 54 pines digitales en el DUE se puede usar como entrada o salida, usando las funciones pinMode(), digitalWrite() y digitalRead(). Funcionan a 3,3 voltios. Cada pin puede proporcionar (fuente) una corriente de 3 mA o 15 mA, dependiendo del pin, o recibir (lectura) una corriente de 6 mA o 9 mA, dependiendo del pin. También tienen una resistencia interna de pull-up (desconectada por defecto) de 100 KOhm. Los pines del 2 al 13 proporcionan una salida PWM de 8 bits con la función analogWrite(), la resolución del PWM se puede cambiar con la función analogWriteResolution(). Las entradas analógicas de los pines A0 al A11. Cada una puede proporcionar 12 bits de resolución (es decir, 4096 valores diferentes). De forma predeterminada, la resolución de las lecturas se establece en 10 bits. Es posible cambiar la resolución del ADC con analogReadResolution(). Los pines de las entradas analógicas de DUE miden desde GND hasta un valor máximo de 3.3V. La aplicación de más de 3.3 V en los pines del DUE dañará el chip SAM3X. Los pines DAC1 y DAC2 proporcionan verdaderas salidas analógicas con una resolución de 12 bits (4096 valores) con la función analogWrite(). Tenga en cuenta que el rango de salida DAC es en realidad de 0.55 V a 2.75 V solamente. (Millahual, 2017)

Comunicación

El Arduino DUE tiene una serie de instalaciones para comunicarse con una computadora, otro Arduino u otros microcontroladores y diferentes dispositivos como, cámaras, tabletas, teléfonos, etc.

El puerto de programación está conectado a un ATmega16U2, que proporciona un puerto COM virtual al software en una computadora conectada. En los pines RX0 y TX0 proporciona comunicación serie a USB para programar la placa a través del microcontrolador ATmega16U2. El software Arduino IDE incluye un monitor en serie que permite enviar datos textuales simples desde y hacia la placa. Los LED de RX y TX en la placa parpadearán cuando los datos se transmitan a través del chip ATmega16U2 y la conexión USB a la computadora .

El puerto USB nativo está conectado al SAM3X. Permite la comunicación serial (CDC) a través de USB. Esto proporciona una conexión en serie al Monitor Serial u otras aplicaciones en su computadora. El puerto USB nativo también puede actuar como un host USB para periféricos conectados como ratones, teclados y teléfonos inteligentes. (Millahual, 2017)

• Protección de Sobrecorrientes USB

El Arduino DUE tiene un polyfusible reiniciable que protege los puertos USB de su computadora de cortocircuitos y sobrecorriente. Aunque la mayoría de las computadoras brindan su propia protección interna, el fusible proporciona una capa adicional de protección. Si se aplican más de 500 mA al puerto USB, el fusible romperá automáticamente la conexión hasta que se elimine el cortocircuito o la sobrecarga. (Millahual, 2017)

2.7.3 Software de arduino

El entorno de desarrollo integrado Arduino o IDE, es un programa para PC compuesto por herramientas de programación. Muy bien puede estar dedicado solo a un lenguaje de programación o tiende a utilizarse para varios lenguajes. Se ha empaquetado como un programa de aplicación; es decir, se compone de un administrador de código, un compilador, un depurador y un constructor gráfico de interfaz de usuario (GUI). Además, Arduino incorpora las herramientas para cargar el programa compilado en la memoria flash del hardware. (Artero, 2013)



Figura 2. 21: Menús y opciones del software arduino IDE.

Es importante conocer cada uno de los menús y opciones que tiene, pero los más importantes por ahora son:

- Botones de Verificar y Subir
- Botón Monitor Serie
- Consola de Error
- Menú herramientas Placa y Puerto
- Menú de pestañas
- Puerto y placa seleccionada
- Menú preferencias
- Proyecto/Sketch

2.8 Matlab

MATLAB o "laboratorio de matrices" de la abreviatura en inglés (MATrix LABoratory), es un software matemático que sirve como herramienta, ofrece un entorno de desarrollo integrado (IDE) con su propio lenguaje de programación (lenguaje M). Es accesible para las plataformas Windows, Mac OS, Unix y Linux. El paquete MATLAB tiene dos instrumentos adicionales que amplían sus aspectos más destacados, en particular Simulink (etapa de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, las capacidades de MATLAB se pueden ampliar con herramientas extras o "toolboxes"; y las de Simulink con los paquetes de bloques o "blocksets". (MathWorks, 2019)

Es un producto generalmente utilizado en universidades y enfocados en trabajos de investigación. En los últimos tiempos, la cantidad de aspectos destacados se ha ampliado, por ejemplo, la programación de procesadores de señales digitales directamente o la creación de código VHDL. (MathWorks, 2019)



Figura 2. 22: Logo de software Matlab/Simulink. (MathWorks, 2019)

Características principales:

- Lenguaje de alto nivel para cálculos científicos y de ingeniería.
- Entorno de escritorio para exploración iterativa, diseño y solución de problemas.
- Gráficas y herramientas para visualizar datos y crear diagramas personalizados.
- Aplicaciones para clasificar datos, analizar señales y ajustar sistemas de control.
- Toolboxes con amplia variedad de aplicaciones científicas y de ingeniería.
- Herramientas para crear aplicaciones con interfaces de usuario personalizadas
- Interfaces para C/C++, Java, .NET, Python, SQL, Hadoop y Microsoft Excel
- Opciones de implementación free para compartir programas de MATLAB con los usuarios finales.

2.9 Simulink

En la práctica de la ingeniería de control, el modelado es una parte significativa. Hoy en día, con el uso de PC y destacadas programaciones, se pueden simular sistemas genuinos muy impredecibles y su conducta puede anticiparse en el mundo real como también supervisarlos. Utilizando el software MATLAB / SIMULINK, se logra realizar simulación de estimaciones eléctricas de algunos parámetros eléctricos de circuitos eléctricos sencillos o complejos al hacer modelos virtuales en SIMULINK. Además de SIMULINK se puede hacer uso de varias bibliotecas de bloques entre ellas "Simulink Support Package for Arduino Hardware" que se utiliza para la comunicación entre SIMULINK y las placas ARDUINO, con esto se logra recibir y transmitir información entre hardware y software. (Matos, Rodríguez, & Toledo, 2018)

2.10 Control PID

Un regulador o controlador PID es un algoritmo de control que permite manejar un sistema de lazo cerrado para lograr el estado de salida ideal. El controlador PID está formado por tres componentes que proporcionan una operación, proporcional, integral y derivativa. Estos tres componentes son los que dan el nombre al controlador PID. (Cajo & Agila, 2015)



Figura 2. 23: Sistema de control en lazo cerrado con control PID.

2.10.1 Signo de referencia y señal de error

La señal r(t) se conoce como referencia y muestra el estado que desea obtenerse en la salida del sistema y(t). En un sistema de control de posición, la referencia r(t) será la posición deseada y su salida y(t) será la posición real del sistema controlado. Como se puede encontrar en la figura anterior, la entrada al controlador PID es la señal de error e(t). Esta señal le muestra al controlador la diferencia entre el estado que se desea o la referencia r(t) y la condición real del sistema estimado por el sensor, la señal h(t). Ante la posibilidad de que la señal de error sea muy significativa, implica que la condición del sistema está muy lejos del estado de referencia deseado. En el caso de que el error sea pequeño, implica que el sistema ha alcanzado el estado deseado. (Pardo, 2013)

2.10.2 Control Proporcional

Como su nombre lo indica, este parámetro de control corresponde a una proporción de la señal de error e(t). Dentro del parámetro proporcional se multiplica la señal de error por la constante **Kp**. Con esto se intenta disminuir el error del sistema. En el momento que el error es muy grande, la acción de control es muy grande y, en general, minimiza este error. (Sung, Lee, & Lee, 2009)

Aumentando el parámetro proporcional Kp lo acompañan los siguientes efectos:

- Incrementa la velocidad de reacción del sistema.
- Reduce el error del sistema en régimen permanente.
- Incrementar la inseguridad del sistema.

2.10.3 Control Derivativo

Como su nombre lo indica, este parámetro de control corresponde a la derivada proporcional de la señal de error e(t). En el control anterior, cuando la posición se encuentra por debajo, la acción de control proporcional intenta incrementar la posición de manera confiable. El inconveniente viene al considerar las inercias. En un punto donde el sistema se mueva rápidamente hacia el punto de referencia, el sistema pasará de largo debido a su inercia, esto crea sobrepulsos y oscilaciones. Para mantener la estabilidad, el controlador debe saber la velocidad para así tener la opción de detenerlo antes de tiempo y amortiguarse. (Sung et al., 2009)

Aumentando el parámetro derivativo Kd lo acompañan los siguientes efectos:

- Incrementa la estabilidad del sistema controlado.
- Reduce un poco la velocidad del sistema.
- El error en régimen permanente continuará como antes.

2.10.4 Control Integral

Este parámetro de control como su nombre lo muestra, corresponde a la integral proporcional de la señal de error e(t). La integral se puede ver como la sumatoria o colección de la señal de error. A largo plazo, se agregan pequeños errores para hacer que el parámetro integral se esté incrementando. Con esto, es posible reducir el error del sistema en régimen permanente. El inconveniente de utilizar el parámetro integral es que agrega una inercia específica al sistema y por lo tanto lo hace cada vez más inestable. (Sung et al., 2009)

Aumentando el parámetro integral Ki tiene lo acompañan los siguientes efectos:

- Disminuye el error del sistema en régimen permanente.
- Incrementar la inestabilidad del sistema.
- Incrementa un poco la velocidad del sistema.



Figura 2. 24: Efectos en la señal modificando parámetros Kp, Kd, Ki. (Pardo, 2013)

2.11 Control FUZZY

Es muy valioso conocer rápidamente cómo se ha desarrollado la teoría control y sus aplicaciones en el área de la lógica difusa. Es una rama de la inteligencia artificial o IA por sus siglas en inglés, que permite a una PC computar datos verdaderos en una escala entre lo verdadero y falso. La lógica difusa ahora es muy normal encontrarla en varias áreas de la innovación tecnológica, tanto en hardware de control, ciencia, autonomía aplicada, etc.

Su objetivo principal es hacer que un sistema dependa de la conducta y el pensar humano, la experiencia que tenga el humano en el trabajo que se desea controlar, es la clave. El sistema toma ese conjunto de conocimientos ya sea de, electrónica, mecánica, industrial, computación, y con esto se establece las reglas para crear la propuesta de control. (Huilcapi Subía, 2015)

2.11.1 Etapas del control FUZZY

El control difuso o FUZZY se compone de tres etapas bien marcadas:

- Fuzzificación
- Base de conocimiento o Conjunto de reglas
- Defuzzificación



Figura 2. 25: Sistema de control difuso Mamdani.

2.11.2 Interfaz de fuzzificación

Aquí se toman las estimaciones nítidas en tiempo real y se relegan valores difusos en un rango entre cero y uno, es decir, valores que tienen un nivel específico de incertidumbre lingüística, este procedimiento se conoce como fuzzificación. Para definir los grados de pertenencia de las estimaciones nítidas a valores difusos se tiene varias funciones de membresía y estas son las más usadas (Huilcapi Subía, 2015):

- Función singleton
- Función triangular
- Función trapezoidal
- Función gaussiana
- Función sigmoidea





2.11.3 Base de conocimiento o Conjunto de reglas y Lógica de decisiones

En la base de conocimientos se establece el grupo de reglas lingüísticas, estas se calibran con la experimentación del sistema. El grado de pertenencia de las variables lingüística se evalúa en el grupo de reglas de inferencia. Se establecen los límites del controlador que en nuestro caso son los límites del movimiento de la plataforma inercial de 2-GDL. El núcleo de un controlador difuso depende de la lógica que utiliza para decidir. Como consecuencia de utilizar la lógica de decisiones resulta en conjuntos recortados difusos o conjuntos escalados. (Huilcapi Subía, 2015)



Figura 2. 27: Lógica de decisiones de un controlador difuso.

2.11.4 Interfaz de defuzzificación

Aquí se toma las estimaciones difusas del fuzzificador y las transforma en valores nítidos, este procedimiento se conoce como defuzzificación, por lo que el sistema comprende las órdenes y desarrollará las actividades que el humano experto haya preparado. La defuzzificación es un proceso más complejo que la fuzzificación porque existen varios métodos matemáticos para realizarlo y presentar las conclusiones basadas en los conjuntos difusos. Algunos de los principales son: el método del centro de gravedad, método del promedio de los centros, método de máximos, entre otros. (Huilcapi Subía, 2015)

2.12 Diseño de Controladores

Los controladores son la base para el manejo y monitoreo de sistemas automatizados, estos van mejorándose día a día con la investigación, el objetivo es tener nuevas maneras de controlar y llegar a mejores resultados de implementación y eficiencia. Matlab presenta varias herramientas para el diseño de controladores utilizando tiempo continuo y discreto, así como herramientas para el diseño de controladores utilizando lógica difusa.

2.12.1 Diseño de Controladores P, PI, PID

Para diseñar un algoritmo de control analógico o digital PID, usando de variables de estado o algún otro método dentro de la teoría de control clásico. En Simulink se puede encontrar las herramientas como bloques dentro de su librería >> Continuo, también dentro de esta librería se tiene bloques para la identificación de plantas y sistemas.



Figura 2. 28: Bloque "PID" en un sistema de lazo cerrado de Simulink.

Como se puede ver dentro del bloque Controlador PID se tiene: qué tipo de controlador se desea utilizar, si se desea trabajar con tiempo continuo o discreto, tiempo de muestreo, cambiar los valores de los parámetros de control y hasta definir una saturación para la salida de dicho controlador.

Controller: PID • Form: Parallel Time domain: O Continuous-time Filter method: Forward Filter method: Forward	
O Continuous-time Integrator method: Forward Filter method: Forward	
Discrete-time	Euler Euler
Controller parameters Source: internal Compensator form	nula
Proportional (P):	
Integral (I):	
Derivative (D): 0 $P + I \cdot T_c \frac{1}{1} + D$	N
	$1 + N \cdot T_{e} - \frac{1}{1}$

Figura 2. 29: Paramentos del bloque "PID Controller".

Este bloque además contiene una herramienta para la sintonización automática de los valores P, I, D, donde se puede variar el tiempo de respuesta y comportamiento transitorio como desee. Se observa una gráfica con línea entrecortada de la respuesta a un escalón y otra gráfica con la sintonización que esta herramienta logra.



Figura 2. 30: Herramienta de sintonización de PID en el bloque "PID Controller".

2.12.2 Diseño de Controladores FUZZY

Para diseñar un algoritmo de control difuso en Matlab/Simulink se debe ingresar el comando "fuzzy" en la ventana de comandos de Matlab con lo cual se abrirá el "Diseñador de Lógica Difusa". Aquí se puede crear las variables de entrada y salida, asignar las funciones de membresía, crear el grupo de reglas, entre otros detalles para el controlador. El método de defuzzificación que se usa por defecto es el de centro de área.





En el diseñador de lógica difusa se tiene una gama de editores, uno de estos es el de las funciones de membresía donde se puede agregar, nuevas variables, nombre, tipo, parámetros, rangos y editar la visualización de las gráficas.



Figura 2. 32: Ventana para editar las funciones de membresía.

Posteriormente de haber creado las variables de entrada y salida con sus respectivas funciones de membresía, se debe crear el conjunto de reglas que asocian las variables, como ya se ha mencionado anteriormente. El trabajo de este conjunto de reglas será óptimo tanto como el diseñador del controlador conozca el sistema.



Figura 2. 33: Ventana para editar el conjunto de reglas.

Después de realizar la creación de entradas, salidas y reglas mencionadas anteriormente, ya el controlador difuso está diseñado en su totalidad, ahora se puede hacer una prueba del sistema ingresando en View >> Rules para verificar los valores de salida, modificando los valores de las variables de entrada. Para ver la gráfica total se ingresa en View >> Surface.



Figura 2. 34: Ventanas para probar y visualizar el efecto del controlador difuso.

Ahora que se tiene el controlador creado y probado, se va a exportar el "archivo.fis" que se lo llama desde el bloque del controlador de lógica difusa y se usa en el control de la planta en cuestión. A continuación, se muestra una planta con su bloque controlador de lógica difusa.



Figura 2. 35: Bloque "FUZZY" en un sistema de lazo cerrado de Simulink.

Se indica la forma adecuada de llamar el "archivo.fis" desde el bloque de control. Es importante mencionar que el "archivo.fis" debe estar en la misma carpeta del "archivo.slx" de Simulink para un correcto funcionamiento del control.

Block Parameters: Fuzzy Logic Controller	×
Fuzzy Inference System	
Specify the Fuzzy Inference System (FIS) as either a structure or file.	a
Parameters FIS name: (For a file, use quotes and file extension, e.g., 'tipper.fis'.)	
'Tanque'	:
OK Cancel Help App	ly

Figura 2. 36: Paramento "FIS name" del bloque FUZZY Logic Controller.

3. MARCO METODOLÓGICO

El proyecto se inicia realizando el diagnóstico de las partes mecánicas y electrónicas de la Plataforma de 2-GDL, con el objetivo de repotenciar dichas partes para la implementación de las guías prácticas empleando el controlador PID y FUZZY.



Figura 3. 1: Plataforma Inercial 2-GDL, vista frontal.

3.1 Diagnóstico de la Plataforma de 2-GDL

Para la ejecución del diagnóstico se energiza la plataforma inercial de 2-GDL y se comprueba el funcionamiento de sus partes electrónicas y mecánicas lo cual se detalla a continuación.

3.1.1 Estructura Mecánica

La estructura mecánica de la plataforma inercial de 2-GDL cuenta con deterioró en los tornillos de los acoples utilizados para los motores 1 y 2.



Figura 3. 2: Plataforma Inercial 2-GDL, acoples. 33



Figura 3. 3: Tornillos de acoples dañados.

3.1.2 Motores Reductores DC con Encoder

En la plataforma de 2-GDL cuenta con dos motores reductores DC con encoder marca Pololu, a continuación, se muestra el siguiente resultado:

- Los motores reductores cuentan con daño en los engranes ya que los tornillos que lo sujetan obstruyen el movimiento de ellos.
- En la evaluación de los motores reductores se notó, que para el movimiento se debe usar un controlador Puente H. La tarjeta en mención utiliza la señal PWM para el control de los motores.
- Se detecta que los encoder poseen un funcionamiento correcto.



Figura 3. 4: Diagnóstico de los motores reductores dc con encoder.

3.1.3 Pantalla táctil Resistiva

La Pantalla táctil resistiva marca NKK Switches posee comunicación RS232 con un controlador que solo funciona en Windows XP, al ser este un sistema operativo descontinuado se necesita crear un nuevo controlador que se comunique con las nuevas versiones del software Matlab/Simulink.

Los pines del conector de comunicación presentan mal estado como se observa en la figura 3.6 ocasionando que la adquisición de datos no sea el correcto.



Figura 3. 5: Diagnóstico de pantalla táctil resistiva.



Figura 3. 6: Conector pantalla táctil resistiva.

3.1.4 Acelerómetro

El Acelerómetro marca Pololu no presentó novedades en el diagnóstico y se corroboró su funcionamiento apropiado.



Figura 3. 7: Acelerómetro, vista de los pines.

3.1.5 Tarjeta PCI

La tarjeta PCI con ayuda de su ficha técnica se evidenció su apropiado desempeño.



Figura 3. 8: Tarjeta PCI.

3.1.6 SparkFun

Al realizar el diagnóstico de la tarjeta SparkFun se utilizó la hoja de datos y como resultado se comprobó su óptimo funcionamiento.



Figura 3. 9: Tarjeta SparkFun.

3.1.7 PCB (Tarjeta de Circuito Impreso)

La tarjeta PCB (Tarjeta de Circuito Impreso) no consta con su hoja de datos ni el manual de conexiones. Para la ejecución del diagnóstico se realizó la puesta en marcha de la plataforma inercial de 2-GDL la cual tiene movimiento de sus brazos. Se concluye que la tarjeta PCB cuenta con un correcto funcionamiento.



Figura 3. 10: Pistas de la tarjeta PCB, vista posterior.



Figura 3. 11: Tarjeta PCB, vista superior.

3.1.8 Fuentes de Voltaje

La plataforma inercial de 2-GDL utiliza dos fuentes de poder:

• Fuente de 12VDC/5Amp, alimenta las tarjetas electrónicas y como se muestra en la figura 3.13 el estado de la fuente es correcto, su voltaje de salida es de 12 voltios con una corriente de 5 amperios



Figura 3. 12: Fuente de 12vdc/5Amp.

• Fuente de 12VDC/30Amp, alimenta los motores reductores y como se muestra en la figura 3.14 el estado de la fuente es correcto, su voltaje de salida es de 12 voltios con un corriente de 30 amperios



Figura 3. 13: Fuente de 12vdc/30Amp.

3.2 Ejecución de la Repotenciación

Las investigaciones científicas que estudian los movimientos de las plataformas inerciales se enfocan con un diseño particular de su estructura, las cuales ayudan al movimiento de sus brazos y equilibrio de la esfera. Parte de la repotenciación es el diseño de la estructura mecánica y a continuación se muestra en la figura 3.15 el nuevo diseño de la plataforma inercial de 2-GDL.



Figura 3. 14: Diseño de la Proforma Inercial 2-GDL, vista isométrica-

Las medidas de las diferentes partes de la estructura mecánica de la plataforma inercial de 2-GDL se detallan en la figura 3.16.



Figura 3. 15: Medidas en centímetros de la Proforma Inercial 2-GDL.

El material que se utiliza para la base de la plataforma inercial de 2-GDL es de acero inoxidable con dobleces a sus lados y soldadura en sus esquinas. En figura 3.17 se observan la nueva ubicación de los servomotores con sus respectivos brazos y la instalación del pilar base que cuenta con un rodamiento tipo rotula en el centro que permite el libre movimiento de los brazos de los servomotores.



Figura 3. 16: Ubicación de los servomotores con sus brazos y el pilar base.

Se reemplaza los motores reductores DC con encoder por los servomotores marca FAN MODEL modelo FS-20W. La nueva ubicación de los servomotores se observa en la figura 3.17 con sus respectivos brazos para transmisión del movimiento. Se realiza la importación de la pantalla táctil resistiva marca NKK Switches ya que la actual tiene daño en los pines del conector de comunicación.



Figura 3. 17: Servomotor marca Fan Model.

Para la creación del nuevo controlador de la pantalla táctil se utiliza un microcontrolador marca Arduino modelo NANO.



Figura 3. 18: Pantalla Táctil Resistiva.

Se selecciona el microcontrolador marca Arduino modelo DUE para la comunicación entre la pantalla táctil resistiva y los servomotores ya que cuenta con una librería académica desarrollada en el software Matlab/Simulink.



Figura 3. 19: Tarjeta Arduino DUE.

Para un mejor control físico de la marcha, paro y el botón de paro de emergencia se realizó la instalación de los siguientes elementos:

- Botón tipo hongo para el paro de emergencia de color rojo
- Botón tipo pulsador para el paro de color rojo.
- Botón tipo pulsador para la marcha de color verde.
- Luz piloto color verde para mostrar que la plataforma esta puesta en marcha.
- Luz piloto color rojo para mostrar que la plataforma está encendida.

El paro de emergencia cuando está enclavado tiene como funcionamiento la desenergización de la plataforma de 2-GDL por lo tanto la luz piloto color rojo está apagado. En la figura 3.21 se muestra la ubicación de los botones de control.



Figura 3. 20: Ubicación de los botones del control físico.

A continuación, en la figura 3.22 se muestra el resultado obtenido al realizar la repotenciación de sus partes mecánicas y electrónicas ya antes mencionadas.



Figura 3. 21: Rediseño de la Proforma Inercial 2-GDL.

La plataforma inercial de 2-GDL utiliza los siguientes elementos para su control: una esfera de acero con 4cm de diámetro y 92.3 gramos de peso, una pantalla táctil de 15 pulgadas resistiva de 4 cables, una placa arduino NANO como controlador de la pantalla, dos servomotores de 20 kgf, una placa arduino DUE como medio de comunicación, y el software Matlab/Simulink donde se diseña y analiza los controladores PID y FUZZY.

3.3 Diseño e Implementación de los Algoritmos de Control PID y Fuzzy

Para la creación del diseño del control automático es primordial realizar la identificación del sistema MIMO existente en la plataforma inercial de 2-GDL, el cual implica entender todas las variables de entrada y las variables de salida al poner en marcha nuestra plataforma. El sistema MIMO por cada grado de libertad que tiene la plataforma cuenta con un sistema SISO.

A continuación, se muestra el diagrama de bloque de cada sistema SISO.



Figura 3. 22: Diagrama de bloques del sistema de control en el Eje X.



Figura 3. 23: Diagrama de bloques del sistema de control en el Eje Y.



Figura 3. 24: Diagrama de bloques del sistema de control.

3.3.1 Pantalla Táctil Resistiva y controlador

El sensor de nuestra plataforma inercial de 2-GDL es la pantalla táctil resistiva, las características principales son:



Figura 3. 25: Dimensiones de la pantalla táctil.

- Dimensión 15 Pulgadas.
- Voltaje de operación: 5 Voltios DC.
- 4 pines de trabajo:
 - PIN 1, posición XLE
 - o PIN 2, posición YUP
 - o PIN 3, posición XRI
 - PIN 4, posición YLO

En la figura 3.26 se observa las dimensiones y ubicaciones de los pines de la pantalla táctil.

Para la lectura de la posición Y se energiza con 5 voltios DC el pin 2 y GND el pin 4, la posición se la adquiere en el pin 3 en voltios. Para la lectura de la posición X se energiza con 5 voltios DC el pin 1 y GND el pin 3 la posición se la adquiere en el pin 2 en voltios, el fabricante recomienda que el pin restante sea conectado a GND y en la figura 3.27 se muestra lo antes mencionado.



Figura 3. 26: Diagrama de conexiones para la obtención de la posición.

Para obtener las posiciones del eje X y del eje Y en tiempo real se debe realizar una programación que se ejecute en un ciclo infinito de veces alternando la energización de los pines y no dañar la pantalla táctil. Dicho programa se desarrolló en el software Arduino y se ejecuta en el Arduino NANO. A continuación, se expone el código del programa.

#define YU A0
#define XR A1
#define YD A2
#define XL A3
#define WPx 9
#define WPy 10
int X = 0; int XReal = 0;
int Y = 0; int YReal = 0;

void setup()
{pinMode(LED_BUILTIN, OUTPUT);}

```
void loop()
{ pinMode(XR,INPUT);
    pinMode(XL,INPUT);
    digitalWrite(XL,LOW);
    pinMode(YU,OUTPUT);
```
```
pinMode(YD,OUTPUT);
digitalWrite(YU,HIGH);
digitalWrite(YD,LOW);
delay(5); X = analogRead(XR);
XReal = map(X, 0, 1023, 0, 185);
analogWrite(WPx,XReal);
/*Serial.print("X = ");
Serial.print(X);
Serial.print(", ");*/
pinMode(YU,INPUT);
pinMode(YD,INPUT);
digitalWrite(YU,LOW);
pinMode(XR,OUTPUT);
pinMode(XL,OUTPUT);
digitalWrite(XL,HIGH);
digitalWrite(XR,LOW);
delay(5);
Y = analogRead(YD);
YReal = map(Y, 0, 1023, 0, 185);
analogWrite(WPy,YReal);
/*Serial.print("Y = ");
Serial.println(Y);
Serial.flush();*/}
```

Los pines para la adquisición de datos utilizados en el arduino NANO son los siguientes:

Pines de lectura:

- PIN1 de la pantalla táctil-Pin Analógico A0 Arduino NANO
- PIN2 de la pantalla táctil-Pin Analógico A1 Arduino NANO
- PIN3 de la pantalla táctil-Pin Analógico A2 Arduino NANO
- PIN4 de la pantalla táctil-Pin Analógico A3 Arduino NANO

Pines de salida que se envía la señal PWM del eje X y del eje Y:

- Pin Digital 9 para la posición X
- Pin Digital 10 para la posición Y

	ŲS	B	
	ê0		
	NIO		
	NP.		
	NO		
	Ing		
	ЧĦ		
	Created Michael Hay CC 3/04/6	By: puldu	

Figura 3. 27: Microcontrolador Arduino NANO

Con la ayuda de dos filtros pasa bajo se realiza la conversión de las señales PWM tanto de la posición X y de la posición Y a señales analógicas. Cabe recalcar que una salida PWM es una señal Digital.

Para la construcción de los filtros Pasa Bajo se utiliza los siguientes elementos:

- 1 Resistencia
- 1 Capacitor



Figura 3. 28: Diseño del filtro Pasa bajo.

Se realiza el cálculo para la obtención de los valores de resistencia y capacitancia necesarias para un filtro que ayude a convertir la señal digital analógica. Los valores de los elementos electrónicos para el diseño del, filtro pasa bajo son:

- R1= 1KΩ
- C1= 2.2uF

Cabe recalcar que se debe diseñar dos filtros pasa bajo tanto para el eje X y para el eje Y, ambos filtros están diseñados con los mismos valores de resistencia y capacitancia. El valor de voltaje máximo de la señal de PWM es de 5 Voltios DC, con este valor de voltaje se realiza los cálculos del filtro que permita que la señal PWM se convierta a una señal analógica.



Figura 3. 29: Señal PWM filtrado con Pasa Bajo.

Para evaluar el funcionamiento del filtro diseñado se utiliza la ayuda del software Matlab y el Arduino NANO y se muestra la gráfica obtenida en la figura. Dicho filtro será incorporado a la tarjeta PCB.

La característica de la señal son las siguientes:

- Los valores de frecuencia de corte obtenido son 7.2343 Hz
- El voltaje promedio con rizado es de 2.5 voltios
- El rizado 0.1136 Voltios

3.3.2 Servomotores

Los servomotores ejecutan los movimientos de la plataforma inercial de 2-GDL tanto en los ejes X y Y. Los servomotores marca Fan Model son utilizados para la plataforma inercial de 2-GDL.



Figura 3. 30: Servomotor marca Fan Model.

La característica de dichos servomotores se muestra a continuación:

- Torque (4.8v/6.0v/7.4V): 16.0 kg-cm (306 oz/in) | 18.0 kg-cm (333 oz/in) | 20.0 kg-cm (347 oz/in)
- Velocidad (4.8v/6.0v/7.4V): 0.18 sec/60° | 0.16 sec/60° | 0.14 sec/60°
- Rango de voltaje operativo: 4.8 ~ 7.4 voltios DC
- Corriente sin carga: <500mA
- Spline: 25T(6mm de diámetro)
- Tipo de motor: Motor sin núcleo
- Tipo de rodamiento: Rodamiento de bolas x 2
- Tipo de engrane: Aleación de acero
- Temperatura: -20°C~60°C
- Longitud del cable conector: 320mm/12.6in
- Carcasa: plástico y aluminio
- Peso: 65 g (2.3 oz)
- Dimensiones: 40 x 20.5 x 40 mm (1.57 x 0.78 x 1.57 in)
- Ángulo de límite: 180°± 10°
- Rango de funcionamiento: 90°(1000~2000 µsec)
- Método de control: PWM
- Rango del ancho de pulso(0°~180°): 500~2500 µsec
- Relación de trabajo: 0.5ms~2.5ms
- Periodo de pulso: 20ms
- Ancho de banda muerto: 4 µsec

Para el control de los movimientos de los servomotores se utiliza la comunicación entre el Arduino DUE y Matlab/Simulink, es necesario la instalación de la librería de Arduino para Matlab/Simulink como se muestra en la figura 3.32.



Simulink Support Package for Arduino Hardware by MathWorks Simulink Team STAFF Run models on Arduino boards. Simulink[®] Support Package for Arduino[®] Hardware enables you to create and run Simulink models on Arduino boards. The support package includes a library of Simulink blocks for configuring and Hardware Support

Figura 3. 31: Librería de Arduino para Simulink.

La figura 3.33 muestra la librería de Arduino instalado en Simulink. Estos bloques fueron utilizados para el diseño de los diferentes algoritmos de control que se estudia en este trabajo de grado.



Figura 3. 32: Bloques de la librería de Arduino instalada en Matlab-

El bloque que se utiliza para el movimiento de los servomotores es el Standard Servo Write que se expone en la figura 3.34 de la librería de Arduino en Simulink. El bloque Standard Servo Write envía la posición angular al pin de la tarjeta arduino especificado en el bloque.

	🔁 Block Parameters: Standard Servo Write 🛛 🗙		
	Arduino Standard Servo Write (mask) (link)		
	Set the shaft position of the standard servo motor.		
	The expected range of input is between 0 and 180. The value of 0 positions the shaft at 0 degrees, while the value of 180 positions the shaft at 180 degrees.		
	Enter the number of the digital pin. Do not assign the same pin number to multiple blocks within a model.		
ARDUINO	View pin map		
Pin 9	Parameters		
indard Servo Write	Pin number:		
	OK Cancel Help Apply		

Figura 3. 33: Característica del bloque Standard Servo Write.

A continuación, se muestra los pines seleccionado para la comunicación de los servomotores en el arduino DUE:

- Pin Digital 9 para el servomotor X.
- Pin Digital 10 para el servomotor Y.

El bloque Standard Servo Write trabaja conectando un bloque constante para el envío de la posición angular de los servomotores, el valor que se envía está en el rango de 0 a 180.

3.3.3 Arduino DUE

La tarjeta Arduino DUE es la interfaz para la adquisición de datos de la lectura de la pantalla y él envió de la señal de control a los servomotores.

A continuación, se describe los pines a utilizar.

Puertos Analógicos para la adquisición de datos de la pantalla táctil:

- Pin Analógico A0 del Arduino DUE.
- Pin Analógico A1 del Arduino DUE.

Puerto Digitales para el envío de posicionamiento angular de los servomotores:

- Pin Digital 9 para el servomotor X.
- Pin Digital 10 para el servomotor Y.



De esta manera mostramos el diseño final de nuestra tarjeta PBC en la figura 3.35.

Figura 3. 34: Tarjeta PBC para la actuación de la Plataforma Inercial de 2-GDL.

3.3.4 Identificación de la Planta

La plataforma Inercial de 2 GDL, tiene como entrada del sistema el sensor de la pantalla táctil resistiva y la salida es el ángulo de los servomotores el cual afecta a la posición de la pantalla táctil en los ejes X y Y, dicho ángulo es obtenido mediante la posición real de la bola que se mueve en la pantalla táctil.

Para la identificación de la planta se debe crear una variable conocida que simula la posición de la bola en la pantalla la cual realiza el movimiento de los ángulos de los servomotores. Dicha variable debe ser una señal que permita simular la posición de la bola. La señal tiene como nombre PRBS y su realización en el software Matlab.

La señal PRBS, que posee el trabajo de titulación, tiene como característica fundamental el rango de operación de los servomotores, estos a su vez trabajan en grados (0 a 180).

Cabe recalcar que cada servomotor tiene un rango de operación distinta por su ubicación como se muestra en la figura 3.17.

A continuación, se muestra las características de la señal PRBS por cada servomotor:

Servomotor X, su rango de operación es de 90° a 150° donde su centro de equilibrio es de 120°, considerando que en este rango de operación se debe tener claro la protección al momento de identificar la planta.

La señal PRBS que se envía al servomotor X con lo antes mencionado es la siguiente. (Cárdenas Ibáñez & Peñaranda Idrovo, 2019)

disp('CREACION DE UNA SEÑAL PRBS'); clear PRBS teta=120; %valor nominal de la referencia umax=5; %valor máximo positivo de la variación porcentual de la referencia. umin=5; %valor mínimo positivo de la variación porcentual de la referencia. PRBS = idinput(500 , 'PRBS' , [0 0.75] , [teta-umin*teta/100 teta+umax*teta/100]); L=length(PRBS); senal=double([(0:L-1)',PRBS]);

Servomotor Y, su rango de operación es de 20° a 80° donde su centro de equilibrio es de 50° considerando que en este rango de operación se debe tener claro la protección al momento de identificar la planta.

La señal PRBS que se envía al servomotor Y con lo antes mencionado es la siguiente.

disp('CREACION DE UNA SEÑAL PRBS');

clear PRBS

teta=50; %valor nominal de la referencia

umax=5; %valor máximo positivo de la variación porcentual de la referencia.

umin=5; %valor mínimo positivo de la variación porcentual de la referencia.

PRBS = idinput(500, 'PRBS', [0 0.75], [teta-umin*teta/100 teta+umax*teta/100]); L=length(PRBS);

senal=double([(0:L-1)',PRBS]);

Se expone el valor de la señal de PBRS, generada para el servomotor en X y el servomotor Y en los tiempos detallados en las siguientes tablas.

Mot	or X
Tiempo	Grados
1	114
2	114
3	114
4	114
5	114
6	114
7	114
8	114
9	114
10	126
11	126
12	126
13	126
14	114
15	114
16	114
17	114
18	126
19	114
20	114
21	114
22	126
23	126
24	126
25	126
26	114
27	126
28	114
29	114
30	126
31	126
32	114

33	114
34	126
35	114
36	114
37	126
38	114
39	114
40	114
41	114
42	126
43	114
44	126
45	126
46	126
47	126
48	114
49	114
50	114
51	126
52	126
53	114
54	114
55	126
56	126
57	126
58	126
59	114
60	126
61	126
62	114
63	126
64	126
65	126
66	114

67	126
68	114
69	126
70	114
71	114
72	114
73	126
74	114
75	126
76	114
77	114
78	114
79	114
80	126
81	126
82	114
83	126
84	126
85	114
86	126
87	114
88	114
89	114
90	126
91	126
92	114
93	114
94	114
95	126
96	126
97	126
98	126
99	126
100	126

 Tabla 1: Datos generados de la señal PRBS-Motor X.

Motor Y				
Tiempo	Grados			
1	47,5			
2	47,5			
3	47,5			
4	47,5			
5	47,5			
6	47,5			
7	47,5			
8	47,5			

9	47,5
10	52,5
11	52,5
12	52,5
13	52,5
14	47,5
15	47,5
16	47,5
17	47,5
18	52,5

19	47,5
20	47,5
21	47,5
22	52,5
23	52,5
24	52,5
25	52,5
26	47,5
27	52,5
28	47,5

29	47,5	53	47,5	77	47,5
30	52,5	54	47,5	78	47,5
31	52,5	55	52,5	79	47,5
32	47,5	56	52,5	80	52,5
33	47,5	57	52,5	81	52,5
34	52,5	58	52,5	82	47,5
35	47,5	59	47,5	83	52,5
36	47,5	60	52,5	84	52,5
37	52,5	61	52,5	85	47,5
38	47,5	62	47,5	86	52,5
39	47,5	63	52,5	87	47,5
40	47,5	64	52,5	88	47,5
41	47,5	65	52,5	89	47,5
42	52,5	66	47,5	90	52,5
43	47,5	67	52,5	91	52,5
44	52,5	68	47,5	92	47,5
45	52,5	69	52,5	93	47,5
46	52,5	70	47,5	94	47,5
47	52,5	71	47,5	95	52,5
48	47,5	72	47,5	96	52,5
49	47,5	73	52,5	97	52,5
50	47,5	74	47,5	98	52,5
51	52,5	75	52,5	99	52,5
52	52,5	76	47,5	100	52,5

Tabla 2: Datos generados de la señal PRBS-Motor Y.

Las señales PRBS tienen una variación del 5% desde su centro de equilibrio, a continuación, se muestran la señal obtenida tanto para el servomotor X y el servomotor Y.



Figura 3. 35: Señal PRBS Servomotor X.



Figura 3. 36: Señal PRBS Servomotor Y.

La señal PRBS diseñada en Matlab se importa al Simulink, al realizar la identificación la señal PRBS se envía a la tarjeta Arduino DUE para la actuación de los servomotores. Se realiza el algoritmo de lectura y escritura dentro del entorno de Simulink. La identificación de la planta se realiza en lazo cerrado como se muestra en la figura 3.38 ya que es una planta inestable, aquello obliga a utilizar un control conocido y aplicar luego la señal PRBS antes descrita.



Figura 3. 37: Diagrama de bloques de eje X utilizando los datos del sensor.

Para la escritura de los valores de posición al servomotor en el eje X y en el eje Y se debe trabajar con el rango de 0 a 1023 Bits, que obliga a realizar una conversión de datos a la salida de la lectura de la pantalla táctil como se expone en la figura 3.38, ya que el valor que se obtiene está en el rango de operación de 0 a 254 bits.

Tal como se muestra en la figura 3.38 utilizamos un filtro Kalman para filtrar la señal aún luego de la conversión de la señal PWM a la señal analógica. El filtro Kalman desarrollado en el entorno de Simulink se utiliza calculando el estado de un sistema dinámico a partir de una serie de mediciones incompletas y o ruidosas.

Este bloque puede usar el estado estimado previamente para predecir el estado actual. También puede usar la medición actual y el estado predicho para estimar el valor del estado actual. Para este caso se utiliza una medida estimada de salida como se muestra en la siguiente figura 3.39.



Figura 3. 38: Filtrado Kalman.

3.3.5 Identificación de la Planta en el eje X

Para la obtención de los controladores conocidos tanto para el eje X y el eje Y se utiliza el método científico de prueba y error estableciendo una hipótesis y haciendo ensayos de cómo responde cada atributo del PID.

En la Figura 3.40 que se presenta a continuación se observa los valores de PID del control conocido, obtenido al realizar el método científico de prueba y error.

PID Controlle	er					
This block im eset, and sig	plements gnal track	continuous- and discrete-tim ing. You can tune the PID ga	e PID control algorit ns automatically usi	thms and includes advanced featu ing the 'Tune' button (requires S	res such as anti-windup, e Simulink Control Design).	external
ntroller: PI	D		•	Form: Parallel		
Fime domain	n:			Discrete-time settings		
	us-time			Integrator method:	Forward Euler	•
	ab cirric			Filter method:	Forward Euler	•
Discrete-t	time			Sample time (-1 for inherited):	Ts	:
Main PID	Advance	d Data Types State At	tributos			
Controller pa	arameters		and deco			
Source:		internal		▼ □	Compensator formula	
Proportional	l (P):	0.35		:		
Integral (I):		-0.004				
Derivative (I	D):	0.0471			$P + I \cdot T = \frac{1}{1} + D = \frac{N}{1}$	1
		Use filtered derivative			z = 1 1+N-7	$\frac{1}{z-1}$
Filter coeffic	cient (N):	100		:		
				Tune		
Initial condit	tions					
Source:	internal					•
Integrator:	0					:
Filter:	0					:
xternal reset	t: none					•
-						

Figura 3. 39: Algoritmo de Control Conocido para el Eje X.

A continuación, se explica la función de transferencia en tiempo discreto del PID obtenido por el método científico de prueba y error utilizando el software Matlab.

P=0.35; %Proporcional I=-0.004; %Integral D=0.0471; %Derivativo Ts=0.01; %Tiempo de Muestreo

KP=P-(P*I*Ts/2); % Equivalencias de constantes de sintonía continuas KI=P*Ts*I; % a discretas

KD=(P*D)/Ts;

pidz=tf([(KP+KI+KD) -(KP+2*KD) KD],[1 -1 0],Ts) %PID discreto Forma posicional pidz =

1.998 z^2 - 3.647 z + 1.648 -----z^2 - z

Sample time: 0.01 seconds Discrete-time transfer function.

Tal como se explica en la figura 3.38, se expone los datos de salida a través de la pantalla táctil, los mencionados datos son enviados al Workspace con los datos de entrada que se están obteniendo luego del control conocido, los valores de entrada y salida tienen como nombre InX y OutX respectivamente.

Con los valores obtenidos se realiza la identificación utilizando la herramienta del software Matlab llamado ident como se muestra en la siguiente figura 3.41.







Figura 3. 41: Estructura del Toolbox Ident.

El Toolbox Ident como se observa en la figura 3.42tiene como característica al momento de ingresar los datos escribirle el nombre de las variables de entrada y las variables de salida. La identificación se realiza en el dominio del tiempo como se muestra en la figura 3.43.

承 Import Data	_		×		
Data Format for Signals					
Time-Domain Si	Time-Domain Signals 🗸 🗸				
Worksp	ace Vari	iable			
Input:	InX				
Output:	OutX				
Data I	nformati	ion			
Data name:	Мо	torX			
Starting time:	0				
Sample time:	Sample time: 0.01				
More					
Import		Reset			
Close		Help			

Figura 3. 42: Ingreso de datos en el Toolbox Ident en el eje X.

A continuación, se muestra la importación de los datos del Workspace y con ayuda de ident se realiza el proceso de la identificación del sistema SISO.

承 System Identification - Untit	tled		i	-		×
<u>F</u> ile <u>O</u> ptions <u>W</u> indow <u>H</u> e	elp					
MotorX	Operations		Import models			
Data Views	To To Univer	Model outp	Model Views ut Transient resp	N	Ionlinear /	ARX
Data spectra	Trash	Model resid	s Frequency resp Zeros and poles Noise spectrum	H	lamm-Wie	ner
Data s	set MotorX inserted. Double clic	ck on icon (right i	mouse) for text information.			

Figura 3. 43: Importación de datos-ident en el eje X.

Para un mejor procesamiento de la señal se seleccionan los datos para la identificación como se muestra en la figura 3.44, de esta manera se elige cierto rango de los datos importados anteriormente.



Se estable la opción de procesamiento de selección de rango.

Figura 3. 44: Procesamiento de señal-Rango en eje X.

A continuación, se muestra el rango seleccionado para la identificación.



Figura 3. 45: Selección del rango de operación para la identificación de la señal.



Figura 3. 46: Rango para realizar la identificación del servomotor X.

En la figura 3.46 se expone el rango de operación seleccionado para la identificación y diseño de la función de transferencia de la planta.

En la figura 3.48 que se describe a continuación se explica el procesamiento de la señal utilizando 2 polos y un cero.

7	Progress	7.45	1,00-100	2.50	15.6	2	
6	21.0/22	1.45	1.230+03	2.58	15.0	2	
0	10.1004	5.05	2.09e+03	3.12	10.4	2	
9 10	10.1501	5.21	4.99e+03	3.79	16.0	2	
11	10 5241	5.54	1 376±04	9.02	16.6	2	
12	9 70709	7.06	2 140+04	5.04	16.0	2	
12	7 20750	2.00	2.140+04	0.00	16.4	2	
14	6 24769	9.02	A 75e±04	10	15.4	2	
15	5 33601	10.2	6 85e±04	12	14.6	2	
16	4 62139	11.5	9 72e+04	14 1	13.4	2	
17	4.02100	12.9	1 36e+05	16.5	11.9	2	
18	3.66708	14.7	1.87e+05	19	9,96	2	
19	3,38483	16.7	2.54e+05	21.5	7.7	2	
20	3.21063	19.1	3.38e+05	23.8	5,15	2	
19 20 timatin	3.38483 3.21063 ng parameter c	14.7 16.7 19.1 covariance.	2.54e+05 3.38e+05	21.5 23.8	7.7 5.15	2	
Jue.							

Figura 3. 47: Procesamiento de identificación de la planta utilizando ident.

Este paso se lo realiza hasta hallar el mejor porcentaje de estimación de la señal, obteniendo la función de transferencia estimada del sistema SISO en el eje X.

La figura 3.49, se evidencia que el procesamiento de la identificación toolbox ident y nos muestra las funciones de transferencia con sus respectivos porcentajes de error a los valores ingresados anteriormente.



Best Fits

tf3: 80.22 tf2: 43.26 tf1: 13.44

Figura 3. 48: Gráfico de identificación con su respectivo porcentaje de error.

Luego se escoge la función de transferencia con mejor porcentaje y se exporta al Workspace para la utilización de la misma, se realiza la debida transformación de tiempo continuo a discreto utilizando el software Matlab.

La función de transferencia que se utiliza es de 2 polos y 1 cero y tiene como nombre la variable tf3 y cuenta con un porcentaje de estimación de 80.22%.

tf3 =

From input "u1" to output "y1":

1.966 s + 1.232

s^2 + 16.11 s + 0.9911

Name: tf3

Continuous-time identified transfer function.

Parameterization:

Number of poles: 2 Number of zeros: 1 Number of free coefficients: 4 Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.

Status:

Estimated using TFEST on time domain data "MotorXv". Fit to estimation data: 80.22% (stability enforced) FPE: 0.942.7, MSE: 0.8342

3.3.6 Identificación de la Planta en el eje Y

De la misma forma como se identificó la planta en el eje X se lo realiza a continuación para el eje Y. Es necesario la obtención del control conocido el cual se muestra en la siguiente figura.

PID Controll	er							
This block in reset, and si	nplements gnal tracki	continuous- and d ng. You can tune t	screte-time PID control a he PID gains automatica	algorithms an Illy using the '	d includes advanced Tune' button (requ	featur ires Si	es such as anti-windu mulink Control Desigr	p, external I).
ontroller: P	ID			▼ Form:	Parallel			
Time domai	n:			Discr	ete-time settings			
Continuous-time Discrete-time		Integrator method: Filter method: Sample time (-1 for inherited):			Forward Euler			
				ted):	Ts	:		
Main PI	Advance	d Data Types	State Attributes					
Controller p	arameters	a butta types	State Attabates					
Source:		internal			¥	Ξ (compensator formula	
Proportiona	I (P):	-0.01573						
Integral (I)	:	-0.02502			1			
Derivative (D):	-0.01397			:		$P + I \cdot T_{i} = \frac{1}{1} + D$	N
		Use filtered der	ivative				* z -1 1+1	$V \cdot T_s \frac{1}{z-1}$
Filter coeffi	cient (N):	100			:			
					Tune			
Initial cond	itions							
Source:	internal							+
Integrator:	0							:
Filter:	0							:
xternal rese	t: none							+
-								

Figura 3. 49: Algoritmo del Control Conocido para el Eje Y.

Se expone la función de transferencia en tiempo discreto del PID obtenido por el método científico de prueba y error utilizando el software Matlab.

P=-0.01579; %Proporcional I=-0.02502; %Integral D=-0.01397; %Derivativo Ts=0.01; %Tiempo de Muestreo

KP=P-(P*I*Ts/2); % Equivalencias de constantes de sintonía continuas KI=P*Ts*I; % a discretas KD=(P*D)/Ts; pidz=tf([(KP+KI+KD) -(KP+2*KD) KD],[1 -1 0],Ts) %PID discreto Forma posicional

pidz =

Sample time: 0.01 seconds Discrete-time transfer function.

Como se explica en la figura 3.48 se obtienen los datos de salida a través de la pantalla táctil, dicho datos son enviados al Workspace con los datos de entrada que se están obteniendo luego de los controles conocidos, los valores de entrada y salida tiene como nombre InY y OutY respectivamente.

Con los valores obtenidos se efectúa el reconocimiento utilizando la herramienta de identificación del software Matlab llamado ident como se obtiene en la figura 3.50

El Toolbox Ident que muestra la figura 3.51 tiene como característica que, al momento de ingresar los datos se debe escribir el nombre de las variables de entradas estos datos son ingresado en el dominio del tiempo.

承 Import Data	_		×						
Data Format for Signals									
Time-Domain Signals 🗸 🗸									
Workspace Variable									
Input:	InY								
Output:	OutY								
Data I	nformat	ion							
Data name:	Мо	torY							
Starting time:	0								
Sample time:	0.0	1							
		More							
Import		Reset							
Close		Help							

Figura 3. 50: Ingreso de datos en el Toolbox Ident en el eje Y.

De esta forma continua la identificación de la planta, se importan los datos del Workspace y con la ayuda de Ident se realiza el proceso de la identificación de la planta como se muestra en la figura 3.52.



Figura 3. 51: Importación de datos-ident.

Para un mejor procesamiento de la señal se seleccion.an los datos para el reconocimiento, de este modo se escoge cierto rango de los datos importados anteriormente.

Por lo cual, se define la opción de procesamiento de selección de rango como se observa en la figura 3.52



Figura 3. 52: Procesamiento de señal-Rango.

A continuación, se expone el rango seleccionado para la identificación.



Figura 3. 53: Rango de operación para la identificación de la señal.



Figura 3. 54: Rango seleccionado para la identificación del servomotor Y.

En la figura 3.55 explica el rango de operación seleccionado para la identificación y diseño de la función de transferencia de la planta.

En la figura 3.56 que se expone a continuación, se ubica el procesamiento de la señal utilizando 2 polos y un cero.



Figura 3. 55: Procesamiento de la identificación de la planta en el eje Y.

Este paso se realiza hasta hallar le mejor porcentaje de estimación de la señal obteniendo la función de transferencia estimada de la planta.

El procesamiento permite identificar las siguientes funciones de transferencia con sus respectivos porcentajes de error a los valores ingresados anteriormente.



Figura 3. 56: Identificación con su respectivo porcentaje de error para el eje Y.

Luego se selecciona la función de transferencia con mejor porcentaje y se exporta al Workspace para la utilización de esta, se realiza la debida transformación de tiempo continuo a discreto utilizando Matlab.

La función de transferencia que se selecciona es de 2 polos y 1 cero y tiene el nombre la variable tf3, la misma que tiene un porcentaje de estimación de 85.69%.

tf3 =

From input "u1" to output "y1":

1.742 s + 2.942 ----s^2 + 20.11 s + 2.991

Name: tf3

Continuous-time identified transfer function.

Parameterization:

Number of poles: 2 Number of zeros: 1

Number of free coefficients: 4 Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.

Status: Estimated using TFEST on time domain data "MotorYv". Fit to estimation data: 85.69% (stability enforced) FPE: 0.8116.3, MSE: 0.6272

Con ayuda de Simulink se evalúa las funciones de transferencia obtenida en lazo cerrado para obtención del algoritmo PID para el eje X y el eje Y.

3.4 Diseño del Algoritmo PID

Para realizar el diseño del algoritmo PID es fundamental definir en qué tiempo se efectúa el control, para ello, la identificación fue obtenida con un Ts=0.01, partiendo de eso se diseña el control en tiempo discreto.

La función de transferencia de los sistemas SISO se transforma de tiempo continuo a tiempo discreto con ayuda del Software Matlab. Para diseño del algoritmo PID se muestra a continuación la figura 3.58.





3.4.1 Diseño del Algoritmo PID en el eje X

• Transformación de Continuo a Discreto

Con ayuda del software Matlab se realiza la transformación

%MOTOR X

numX=[1.966 1.232]; denX=[1 16.11 0.9911]; tfX=tf(numX,denX); step(tfX) G_sx=tf(numX,denX) G_zx=c2d(G_sx,0.01)

Se muestra el resultado obtenido en la figura 3.59.



Figura 3. 58: Respuesta a una señal escalón del eje X.

Función de transferencia en tiempo discreto.

0.01822 z - 0.0181 ----z^2 - 1.851 z + 0.8512

Sample time: 0.01 seconds Discrete-time transfer function.

 Determinación de los parámetros de PID usando el modelo de la planta del eje X en Simulink.

Main Data 1	ypes State	ator must be greater than or equal to the order of the numerator.
Data		
	Source	Value
Numerator:	Dialog	▼ [0.01822 -0.0181]
Denominator:	Dialog	▼ [1 -1.851 0.8512]
Initial states:	Dialog	• 0
Input processin	ng: Elements a skipping divide	as channels (sample based) • e by leading denominator coefficient (a0)
ample time (-1	for inherited):	
		1
-1		
-1		
1		

Figura 3. 59: Valores de la función de transferencia.

Se realiza el diseño de la planta del eje X para la obtención de los parámetros Kp, Ki y Kd, el diagrama es en lazo cerrado como se muestra en la figura 3.61.





Utilizando el Toolbox PID tuner se obtiene el parámetro constante de PID para que trabaje de tal forma como se espera que la planta responda en el eje X.

En la siguiente figura se describe la respuesta sin modificar los valores de constante Kp, Ki, Kd en un tiempo de 10.0 segundo de ejecución de la planta del eje de las X.

Controller Parameters	
	Tuned
P	-0.0158
I	-0.0390
D	-0.055
N	100
Performance and Robustnes	S
Performance and Robustnes	s Tuned
Performance and Robustnes Rise time	s Tuned 1.54 seconds
Performance and Robustnes Rise time Settling time	s Tuned 1.54 seconds 6.61 seconds
Performance and Robustnes Rise time Settling time Overshoot	s Tuned 1.54 seconds 6.61 seconds 8.08 %
Performance and Robustnes Rise time Settling time Overshoot Peak	s Tuned 1.54 seconds 6.61 seconds 8.08 % 1.08
Performance and Robustnes Rise time Settling time Overshoot Peak Gain margin	s Tuned 1.54 seconds 6.61 seconds 8.08 % 1.08 29.9 dB @ Inf rad/s
Performance and Robustnes Rise time Settling time Overshoot Peak Gain margin Phase margin	s Tuned 1.54 seconds 6.61 seconds 8.08 % 1.08 29.9 dB @ Inf rad/s 63.4 deg @ 0.909 rad/s

Figura 3. 61: Constantes de PID eje de las X.

Con la ayuda del Toolbox para encontrar el PID se hallan los datos que se describen en la figura 3.62.

Los parámetros ubicados presentan la respuesta a la función escalón de la función de transferencia como se indica en la figura 3.63.



Figura 3. 62: Respuesta al escalando en lazo cerrado para el eje X.

Con la obtención de los parámetros de Kp, Ki y Kd del PID del eje de las X en lazo cerrado se procede a la evaluación de dichos parámetros en la plataforma inercial de 2-GDL.

Se realiza el diseño del algoritmo de la planta de 2 GDL en el Simulink considerando las entradas y salidas antes mencionadas. Para el desarrollo de la misma se emplea el diseño de la figura 3.38 que se realizó las siguientes modificaciones que se describen en la figura 3.64.



Figura 3. 63: Bloques para el eje de X, con el algoritmo de control PID.

Al realizar las pruebas con la planta real se realizó ajustes en los valores PID los cuales se muestra a continuación en la figura 3.65.

Dia di Danam	D:-	anata DID X						
DID Controller	ieters: Dis	Crete PID A						
PID Controller								
This block imp windup, extern Simulink Contr	nal reset rol Desig	continuous- and d , and signal trackin n).	iscrete-time PID contr ig. You can tune the P	ol algor 'ID gain	ithms and ir s automatic	ally using the '	ed features s Tune' butto	uch as anti- on (requires
Controller: PID)		•	Form:	Parallel			
Time domain:				Disc	rete-time se	ttings		
	-timo			Integ	grator meth	od:	Forward Eu	ler 🔻
Continuous	sume			Filter	method:		Forward Eu	ler 👻
Oiscrete-tir	me			Sam	ple time (-1	for inherited):	Ts	
Main PID	Advance rameters	d Data Types	State Attributes					
Source:		internal			•	Comper	isator formula	1
Proportional ((P):	-0.0200			:			
Integral (I):		-0.0290						
Derivative (D);	-0.0200			:	$P \pm I.T$	$1 \rightarrow D$	Ν
	,	Use filtered der	ivative			1 1 1 1	$s \overline{z-1} + D \overline{1+}$	$N \cdot T_s \frac{1}{z-1}$
Filter coefficie	ent (N):	100			:			~ 1
					Tune			
Initial condition	ons							
Source: i	nternal							•
Integrator: 0)							:
Filter: 0)							:
External reset:	none							•
Iqnore reset	t when li	nearizing						>
0				Г	OK	Cancel	Help	Apply

Figura 3. 64: Parámetros de control PID en tiempo Real para el eje de las X.

Con la ayuda de Matlab, se muestra a continuación la función de transferencia en tiempo discreto del controlador PID de la plataforma inercial de 2-GDL en tiempo real del eje de las X

PID discreto en forma posicional.

pidz =

0.02 z^2 - 0.06 z + 0.04

z^2 - z

Sample time: 0.01 seconds Discrete-time transfer function.

En la figura 3.66 se muestra el algoritmo de control PID en el eje de las X.



Figura 3. 65: Diseño del Algoritmo PID en el eje X. 77

3.4.2 Diseño del Algoritmo PID en el eje Y

 Transformación de tiempo continuo a discreto de la función de transferencia del sistema SISO del eje Y.

Con ayuda del software Matlab se realiza la transformación

```
%MOTOR Y
numY=[1.742 2.942];
denY=[1 20.11 2.9911];
tfY=tf(numY,denY);
step(tfY)
G_sy=tf(numY,denY)
G_zy=c2d(G_sy,0.01)
```

Resultado obtenido a la transformación se muestra en la figura 3.67



Figura 3. 66: Respuesta a una señal escalón para el eje Y.

Función de transferencia en tiempo discreto.

G_zy =

0.01592 z - 0.01565 -----z^2 - 1.818 z + 0.8178

Sample time: 0.01 seconds

Discrete-time transfer function.

Se realiza el diseño de la planta del eje Y para la obtención de los parámetros Kp, Ki y Kd, el diagrama es en laso cerrado como se muestra en la figura 3.68.



Figura 3. 67: Planta del Eje Y en Simulink.

Utilizando el Toolbox PID tuner se obtiene los parámetros constantes de PID para que trabaje de tal forma como se espera que la planta responda en el eje Y. En la siguiente figura se describe la respuesta sin modificar los valores de constante Kp, Ki, Kd en un tiempo de 10.0 segundo de ejecución de la planta del eje de las Y

Controller Parameters		
	Tuned	
P	-0.0351	
I	-0.0248	
D	-0.02781	
N	100	
Performance and Robustnes	5	
Performance and Robustnes	s Tuned	
Performance and Robustnes	s Tuned 0.599 seconds	
Performance and Robustnes Rise time Settling time	s Tuned 0.599 seconds 8.6 seconds	
Performance and Robustnes Rise time Settling time Overshoot	s Tuned 0.599 seconds 8.6 seconds 8.53 %	
Performance and Robustnes Rise time Settling time Overshoot Peak	s Tuned 0.599 seconds 8.6 seconds 8.53 % 1.09	
Performance and Robustnes Rise time Settling time Overshoot Peak Gain margin	s Tuned 0.599 seconds 8.6 seconds 8.53 % 1.09 16.8 dB @ 11.3 rad/s	
Performance and Robustnes Rise time Settling time Overshoot Peak Gain margin Phase margin	s Tuned 0.599 seconds 8.6 seconds 8.53 % 1.09 16.8 dB @ 11.3 rad/s 61 deg @ 1.96 rad/s	

Figura 3. 68: Constantes de PID eje de las Y.

Con la ayuda del Toolbox para encontrar el PID se hallan los datos que se describen en la figura 3.69.

Los parámetros ubicados presentan la respuesta a la función escalón de la función de transferencia como se indica en la figura 3.70.



Figura 3. 69: Respuesta al escalón en circuito de lazo cerrado para el eje X.

Con la obtención de los parámetros de Kp, Ki y Kd del PID del eje de las Y en lazo cerrado se procede a la evaluación de dichos parámetros en la plataforma inercial de 2-GDL.

Se realiza el diseño del algoritmo de la planta de 2-GDL en el Simulink considerando las entradas y salidas antes mencionadas. Para el desarrollo de la misma se emplea el diseño de la figura 3.38 donde se muestran las siguientes modificaciones para el eje de las Y que se presentan en la figura 3.71.



Figura 3. 70 Algoritmo de control PID para el eje Y.

Al realizar las pruebas con la planta real se realiza ajustes en los valores PID los cuales se describen a continuación.

Block Parameters: Di	iscrete PID Y					×
PID Controller						
This block implements windup, external rese Simulink Control Desig	s continuous- and di t, and signal trackin gn).	iscrete-time PID contr g. You can tune the F	rol algorithms and i PID gains automatio	ncludes advance ally using the 'T	ed features such as Fune' button (requ	anti- Jires
Controller: PID		•	Form: Parallel			
Time domain:			Discrete-time se			
O Continuous-time O Discrete-time			Integrator meth	Forward Euler Forward Euler Ts		
			Filter method:			
			Sample time (-1			
Main PID Advance Controller parameter	ed Data Types	State Attributes				
Source:	internal		•	<u>Compens</u>	sator formula	
Proportional (P):	-0.02200]		
Integral (I):	-0.01751]		
Derivative (D):	-0.02215			$P + I \cdot T_s$	$\frac{1}{1} + D - \frac{N}{1}$	_
	Use filtered der	ivative			$z - 1 = 1 + N \cdot T_s - \frac{1}{2}$	2-1
Filter coefficient (N):	100]		
			Tune]		
Initial conditions						
Source: internal						•
Integrator: 0						:
Filter: 0						:
External reset: none						•
Iqnore reset when I	linearizing					,
0						

Figura 3. 71: Parámetros de control PID en tiempo Real para el eje Y.

Con la ayuda de Matlab se muestra la función de transferencia en tiempo discreto del controlador PID de la plataforma inercial de 2-GDL.

PID discreto en forma posicional.

pidz =

0.02673 z^2 - 0.07546 z + 0.04873

z^2 - z

Sample time: 0.01 seconds

Discrete-time transfer function.

En la figura 3.73 se muestra el algoritmo de control PID en el eje de las Y


Figura 3. 72: Diseño del Algoritmo PID en el eje Y. 83

Diseño del algoritmo de control PID tanto para el eje X y para el eje Y se presenta el diagrama de bloques del sistema de control automático de la plataforma inercial de 2-GDL que se expone en la figura 3.74.



Figura 3. 73: Algoritmo de control PID.

×	▼ Sn/ConY - □ ×									
File Tools View Simulation Help	File Tools View Simulation Help									
◎ · ◎ ◎ \$ · Q · A · A · # Ø ·	◎ · ◎ ◎ ≫ · <u>♡</u> · ፼ · @ ` @ · @ ` @ ` @ ` @ ` @ ` @ ` @ ` @ ` @ `									
There appears to be gaps between data points on the visual. Turn on Markers to see the actual data points.	There appears to be gaps between data points on the visual. Turn on Markers to see the actual data points.									
L Das Gener X Kaines Filer X	Kainaa Filer Y									
20										
	and Anna Anna Anna Anna Anna Anna Anna A									
 Ma-Mattananthalanantha-Jahanan-Amtatananunahanantah tantual 	A REAL WARD STATE HARD THE STATE STATE AND									
a.										
31										
20										
5 50 162 153 266 273 30										
Reads Panels hand (Meals) 7.510.52	Pands Date D									

Figura 3. 74: Señal de Control del PID la Plataforma Inercial de 2-GDL.

3.5 Diseño del Algoritmo FUZZY

Para el diseño del algoritmo Fuzzy en la plataforma inercial de 2-GDL se utiliza la aplicación Matlab con ayuda del Toolbox Fuzzy.

Para continuar con el diseño es necesario entender la operación de la planta y en base a esa información se establecen las reglas de membresía y de correspondencia.

fuzzy	s for <u>Getting Started</u> .					
承 Fuzzy Logic Desig	ner: Untitled			-		×
File Edit View						
	<mark>/ </mark> [Untitled	٦	\square	\mathbb{N}	
input1		(mam dani)	FIS Tune:		tput1	
Input1	Untitied	(mamdani)	FIS Type:	ou	tput1	
Input1	Untitled	(mamdani)	FIS Type: rent Variable	ou	tput1	
FIS Name: And method Or method	Untitled min max	(mamdani)	FIS Type: rent Variable ne e	ou mam inpu	dani trut1	
FIS Name: And method Or method Implication	Untitled min max min	(mamdani) Cur V Nar Typ Rar	FIS Type: rent Variable ne e uge	ou mam input input [0 1]	tput1	
Input1 FIS Name: And method Or method Implication Aggregation	Untitled min max max	(mamdani)	FIS Type: rent Variable ne e ge	ou mam input input [0 1]	dani	

Figura 3. 75: Toolbox Fuzzy de Matlab.

Tal como se explica en la figura 3.76, para llamar al Toolbox es necesario escribir la palabra fuzzy en la ventana de comando de Matlab. El Toolbox Fuzzy como se presenta en la figura 3.77, tiene todas las características para el diseño del algoritmo de control.

Se utiliza el método Mamdani para la creación del algoritmo de control, por lo cual se presenta los diferentes métodos existentes para la creación de algoritmos.



Figura 3. 76: Métodos de creación de algoritmo FUZZY.

Luego de elegir el método de fuzzificación se debe definir las variables de entrada y de salida, para el desarrollo del algoritmo de control FUZZY de la plataforma inercial de 2-GDL

Se definió las siguientes variables de entradas:

- Error, representa la retroalimentación de error de la plataforma inercial de 2-GDL
- **D-Error**, representa la derivada de error.

La variable de Salida:

• **Control**, representa la señal de control hacia los servomotores que entrega la posición angular del bloque FUZZY.

Nombre de bloque Fuzzy por cada eje:

- FuzzyX, en el entorno de Simulink al utilizar el bloque FUZZY es necesario nombrar al bloque para la definición, para este caso el FUZZY del eje x se llama FuzzyX.
- **FuzzyY**, en el entorne de Simulink al emplear el bloque FUZZY se requiere nombrar al bloque para la definición, para este caso el FUZZY del eje y se llama FuzzyY.

En la figura 3.78 se muestra los bloques FUZZY utilizando el método Maldini con sus respectivas variables de entradas y salidas.



Figura 3. 77: FUZZY y variables para el eje Y y X.

Para editar las funciones de membresía de control tanto para las señales de entrada Error y D-Error y la variable de salida de Control es necesario hacer énfasis en reconocer cómo es el funcionamiento de operación de la planta inercial de 2-GDL.

3.5.1 Diseño del Algoritmo FUZZY en el eje X

El diseño del algoritmo FUZZY para el eje X cuenta con funciones gaussianas para las variables de entrada y de salida como se expone a continuación.

Variables de entrada:

 Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000 como se explica en la figura 3.79.



Figura 3. 78: Rango de operación de la señal de ERROR del eje X.

 D-Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000 como se describe en la figura 3.80.



Figura 3. 79: Rango de operación de la señal de D-ERROR del eje X.

Variable de Salida:

Control, tiene 5 funciones de membresía gaussianas que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 107.5 a 127.5, este rango es el movimiento del servomotor en el eje de X y se explica en la figura 3.81.



Figura 3. 80: Rango de operación de la señal de CONTROL del eje X.

La señal de CONTROL no se realiza ninguna escala a la salida del FUZZY ya que el rango de operación de la señal gaussiana es el rango de operación del movimiento del servomotor en el eje de la X.

Para la definición de las reglas de correspondencia que indican la acción a tomar, las cuales se muestran en la tabla 3 que tiene la relación de las variables de entras y salidas del controlador FUZZY del eje X.

		D-ERROR							
		NEG	CERO	POS					
ш	NEG	PG	Ρ	Ρ					
RRC	CERO	Ρ	Z	N					
R	POS	N	Ν	NG					

 Tabla 3: Reglas de correspondencia

Con las reglas de correspondencia expuestas en la tabla 3, se ingresan los valores en el editor del Toolbox FUZZY, el cual será exportado al Workspace para luego ser importado al Simulink. Se observa en la figura 3.82 el ingreso de valores en el editor de reglas.

承 Rule Editor: FuzzyX	_		×
File Edit View Options			
If (ERROR is NEG) and (D-ERROR is NEG) then (CONTROL is PG) (1) Z. If (ERROR is NEG) and (D-ERROR is CERO) then (CONTROL is P) (1) J. If (ERROR is NEG) and (D-ERROR is POS) then (CONTROL is P) (1) J. If (ERROR IS CERO) and (D-ERROR is NEG) then (CONTROL is P) (1) J. If (ERROR is CERO) and (D-ERROR is POS) then (CONTROL is Z) (1) J. If (ERROR is CERO) and (D-ERROR is POS) then (CONTROL is N) (1) J. If (ERROR is POS) and (D-ERROR is CEG) then (CONTROL is N) (1) J. If (ERROR is POS) and (D-ERROR is CEG) then (CONTROL is N) (1) J. If (ERROR is POS) and (D-ERROR is CERO) then (CONTROL is N) (1) J. If (ERROR is POS) and (D-ERROR is POS) then (CONTROL is N) (1) J. If (ERROR is POS) and (D-ERROR is POS) then (CONTROL is N) (1)			~
If and D-ERROR is D-ERROR is NEG CERO POS none v not not	The C NG N Z P PG non	n ONTROL e not	is
Connection Weight: or and 1 Delete rule Add rule Change rule		Clos	>>

Figura 3. 81: Reglas de correspondencia de control FUZZY del eje X.

Con la creación del contralor FUZZY para el eje X realizado en el Toolbox FUZZY se debe diseñar el diagrama de bloques en Simulink.

Es importante recordar que al diseñar el controlador FUZZY se define dos variables de entradas las cuales son ERROR y D-ERRO para la medición de la velocidad y el cambio de error se implementan estas dos variables de entradas respectivamente. En la variable de entrada de la derivada del error se coloca un bloque derivativo en tiempo discreto para cumplir con lo antes mencionado.

Block Parameters: Fuzzy Controller X							
Fuzzy Inference System							
Specify the Fuzzy Inference System (FIS) as either a structure or a file.							
Parameters							
FIS name: (For a file, use quotes and file extension, e.g., 'tipper.fis'.)							
'FuzzyX'	:						
<u>OK</u> <u>Cancel</u> <u>H</u> elp <u>Apply</u>	/						

Figura 3. 82: Parámetros del Bloque FUZZY en Simulink.

A continuación, se muestra en la figura 3.83 el diagrama de bloques para el control automático utilizando el bloque FUZZY.



Figura 3. 83: Diagrama de control lógico FUZZY para el eje de las X.

3.5.2 Diseño del Algoritmo FUZZY en el eje Y

El diseño del algoritmo FUZZY para el eje Y contará con funciones gaussianas para las variables de entrada y de salida como explica a continuación Variables de entradas:

 Error, tiene 3 funciones de membresía gaussiana, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000 como se explica en la figura 3.85.



Figura 3. 84: Rango de operación de la señal de ERROR del eje Y.

 D-Error, tiene 3 funciones de membresía gaussiana, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000 como se describe en la figura 3.86.



Figura 3. 85: Rango de operación de la señal de D-ERROR del eje Y.

 Control, tiene 5 funciones de membresía gaussiana que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 10.5 a 90.5, este rango es el movimiento del servomotor en el eje de Y, se visualiza en la figura 3.87.



Figura 3. 86: Rango de operación de la señal de CONTROL del eje Y.



Figura 3. 87: Reglas de correspondencia de control FUZZY del eje Y

La señal de CONTROL no se realiza ninguna escala a la salida del FUZZY ya que el rango de operación de la señal gaussiana es el rango de operación del movimiento del servomotor en el eje de la Y.

Para la definición de las reglas de correspondencia que indican la acción a tomar las cuales se describe en la tabla 3, se tiene la relación de las variables de entras y salidas del controlador FUZZY del eje Y. Se muestra en la figura 3.88 las reglas de correspondencia del eje Y

Con el diseño del contralor FUZZY para el eje Y creado en el Toolbox Fuzzy se establece el diagrama de bloques en Simulink, en la figura 3.89 se observa la creación del bloque FUZZY en Simulink.

Es importante recordar que al diseñar el controlador FUZZY se definió dos variables de entradas las cuales son ERROR y D-ERROR, error y la derivada de error respectivamente.

Para la medición de la velocidad y el cambio de error se implementa las dos variables de entradas antes mencionadas.

En la variable de entrada de la derivada del error se instala un bloque derivativo en tiempo discreto para cumplir con lo antes mencionado.

Block Parameters: Fuzzy Controller Y	×
Fuzzy Inference System	
Specify the Fuzzy Inference System (FIS) as either a structure or a file.	
Parameters FIS name: (For a file, use quotes and file extension, e.g., 'tipper.fis'.)	
'FuzzyY'	:
<u>O</u> K <u>Cancel H</u> elp <u>Apple</u>	y

Figura 3. 88: Parámetros del Bloque FUZZY en Simulink.

A continuación, se muestra en la figura 3.90 el diagrama de bloques para el control automático donde se emplea el bloque FUZZY para el eje Y.



Figura 3. 89: Diagrama de control lógico FUZZY para el eje de las Y.

Luego de crear los controladores FUZZY individualmente, a continuación se realiza el diagrama de bloques que se muestra en la figura 3.91 para la plataforma inercial de 2-GDL tanto para el eje X y para el eje Y.



Figura 3. 90: Algoritmo de control FUZZY.



Figura 3. 91: Gráficas del Funcionamiento de control FUZZY



Figura 3. 92: Movimiento de la esfera sobre pantalla táctil

4. GUÍAS DE PRÁCTICAS DE LABORATORIO

4.1 **PRÁCTICA # 1**

SALESIANA ECUADOR		Gl	JIA DE PRÁCTICA DE LABORATORIO					
CARRERA: Ingeniería Electr			onica	ASIGNATURA: Teoría de Control I				
NRO. PRÁCTICA:	1	TÍTULO DE LA PRÁCTICA: LECTURA DE UBICACIÓN DE LA ESFERA EN LA PANTALLA DE LA PLATAFORM INERCIAL DE 2-GDL CON EL SOFTWARE MATLAB/SIMULINK.						
OBJETIVOS:								
Conocer la conexión de la pantalla ubicada en la plataforma inercial de 2-GDL								
Instalar el paque	ete	de sopor	te de S	imulink para el hardware Arduino.				
Instalar drivers	dels	software	Arduin	o (IDE).				
Crear un progra	ma	en Simu	ılink pa	ra observar los valores de las coordenadas				
donde se encue	ntra	a la esfer	a.					
INSTRUCCIONES	 1. 2. 3. 4. 5. 6. 	Dispon Matlab/ Aliment 110VA0 Confirm emerge Encenc Conect comput Visualiz	er de (Simulir tar la p C. nar qu encia no der la p ar la p tador ut zar las l	un computador con los programas nk y Arduino IDE. planta "Plataforma inercial de 2-GDL" con e la botonera tipo hongo o paro de o esté accionada. lanta presionando el botón de marcha. planta "Plataforma inercial de 2-GDL" al cilizando el cable USB macho-macho. uces indicadoras de encendido en la planta.				

ACTIVIDADES POR DESARROLLAR

- 1. Instalar el paquete de soporte de Simulink para el hardware Arduino.
- 2. Instalar drivers de Arduino IDE.
- **3.** Guía de pasos para energizar, encender y comunicar la planta "Plataforma inercial de 2-GDL".
- **4.** Creación del programa para lectura de los valores de coordenadas en la pantalla táctil resistiva de la planta "Plataforma inercial de 2-GDL".
- 5. Ejecución del programa de lectura de las coordenadas de posición en la pantalla.

<u>Paso 1.</u> Instalar el paquete de soporte de Simulink para el hardware Arduino.

En la pantalla principal del software Matlab dirigirse a HOME >> Add-Ons >> Get Hardware Support Packages.



Figura 4. 1: Menú de complementos para Matlab.

Buscar el paquete "Simulink Support Package for Arduino Hardware" e iniciar con la instalación.





Cuando se vaya a iniciar el proceso de instalación se debe ingresar las credenciales de una cuenta de MathWorks. En el caso de no contar con una cuenta, la creación de ella es gratis y muy sencilla en la página de MathWorks. https://www.mathworks.com/mwaccount/register

on's MATLAB license, use your
~
~
Cancel Create
Can

Con la cuenta ingresada, seguir los pasos hasta finalizar la instalación, aceptando términos y condiciones más otros terceros complementos. Una vez descargado e instalado el paquete se procede a instalar los drivers que incluye el software Arduino IDE.

Download and Installation Progress								
🤣 Downloading Support Packages 100%								
Ownloading Third-Party Packages 0%								
Installing Support Packages								
Installing Third-Party Packages								
Configuring your installation								

Figura 4. 4: Paquete de Simulink para hardware Arduino.

Paso 2. Instalar drivers de Arduino IDE.

Este software se lo descargar de forma gratuita directamente de la página oficial de arduino. <u>https://www.arduino.cc/en/Main/Software</u>. En el proceso de instalación del software Arduino IDE es fundamental que se incluya la instalación de todos los complementos para la comunicación del

hardware Arduino con el computador y por ende con el software Matlab/Simulink.

Arduino Setup: Installation Options Check the components you want to install and uncheck the component you don't want to install. Click Next to continue.								
Select components to install:	Install Arduir Install USB d Create Start Create Desk Associate .ir	no software Iriver t Menu shortcu top shortcut no files	t					
Space required: 482.3MB								
Cancel Nullsoft Insta	all System v3.0	< Back	Nex	t >				

Figura 4. 5: Componentes para instalación completa de Arduino IDE.

Este software de Arduino contiene los drivers necesarios para utilizar la placa Arduino DUE en el software Matlab/Simulink que sirve para comunicar la plataforma inercial y el computador.



<u>Paso 3.</u> Guía de pasos para energizar, encender y comunicar la planta "Plataforma inercial de 2-GDL".

Conectar el cable de poder en el costado izquierdo de la plataforma como se ve en la ilustración, esta conexión cuenta con un fusible de 110V/2A como protección de sobre voltajes. Inmediatamente, si el paro de emergencia no se encuentra pulsado, se encenderá la luz piloto roja lo que indica que la planta está alimentada con 110V.



Figura 4. 7: Conector IEC y luz piloto indicadora de conexión.

Se presiona el botón de marcha lo que encenderá la luz piloto verde que indica que la planta se ha encendido.



Figura 4. 8: Activación de botón marcha y luz piloto ON/OFF.

Se conecta la planta con el computador utilizando un cable USB macho-macho para iniciar la comunicación con el hardware de Arduino.



Figura 4. 9: Conexión de cable USB en la planta y el computador.

Al estar conectados ambos dispositivos, se muestra en la pantalla principal de Matlab un mensaje que indica, el dispositivo se encuentra listo para usarlo con el paquete de soporte de Simulink para Arduino, caso contrario repetir paso 1.



<u>Paso 4.</u> Creación del programa para lectura de los valores de coordenadas en la pantalla táctil resistiva de la planta "Plataforma inercial de 2-GDL".

Ahora que se puede utilizar el paquete de soporte de Simulink para Arduino se procede a verificar que los componentes reciban y envíen datos sin ningún problema. Se procede a abrir un modelo en blanco en la herramienta Simulink.

Pa P	RACTICA	01 - S	imulink													-		×
File	Edit V	iew	Display I	Diagram	Sim	ulation	Anal	ysis Co	de To	ools He	нp							
3	• 🛅 •		$\Leftrightarrow \Rightarrow$		22 4	ĝ.	•	ų,	4) 🗈 (- 12	10.0]	Normal	•	@ •	₩•
PRA	CTICA_01																	
۲	PRACT	ICA_0																•
Đ,																		
≓																		
A																		
2																		
8																		
Ē1																		
>>																		
Ready										100%						Vari	iableStep	Auto

Figura 4. 11: Modelo en blanco de Simulink.

Para abrir la librería de Simulink, dirigirse a View>>Library Browser. Aquí se encuentra el paquete de Simulink para Arduino previamente instalado, ahora para la lectura de la pantalla táctil resistiva se necesita las entradas analógicas del arduino DUE.



Figura 4. 12: Bloque "Analog Input" del paquete de soporte para Arduino.

Se crea un modelo en Simulink usando los bloques de entrada analógica para lectura de las coordenadas en la pantalla. Se inserta dos bloques a la pantalla principal designando el Pin 0 y Pin1 para posición de X y Y respectivamente.





Antes de ejecutar el programa se debe seleccionar el hardware a utilizar en los parámetros de configuración de Simulink, elegir la placa 'Arduino DUE'. Se aplica los parámetros por defecto para la selección del puerto COM en automático y se crea una variable para el tiempo de muestreo en la ventana de comandos. Ts = 0.01. Se Aplican y aceptan los cambios.



<u>Paso 5.</u> Ejecución del programa de lectura de las coordenadas de posición en la pantalla.

Se procede a ejecutar el programa con todas las configuraciones para la correcta conexión con el hardware Arduino DUE.



Figura 4. 16: Botón de ejecución en Simulink.

Aquí se debe observa la lectura de las coordenadas en el eje X y eje Y en un bloque 'display'. El bloque representa el voltaje como un valor digital (0-1023, mínimo a máximo).



Figura 4. 17: Práctica #1 ejecutada en tiempo real.

Estamos listo para realizar las pruebas de adquisición de datos de la pantalla táctil resistiva.

Los valores obtenidos están el rango de 0 a 1023 y se observa de acuerdo a la ubicación de la esfera sobre la pantalla táctil.

Se debe tener una esfera de un peso de 92 a 96 gramos para un correcto funcionamiento.

RESULTADO(S) OBTENIDO(S)

Con el programa donde se obtiene los valores de la posición de la esfera en la pantalla táctil resistiva, se puede mover manualmente la esfera para observar los cambios que produce su movimiento en la lectura de Simulink.



Figura 4. 18: Lectura de posición superior derecha en la pantalla.



Figura 4. 19: Lectura de posición inferior derecha en la pantalla.





4.2 PRÁCTICA # 2

	UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR			JIA DE PRÁCTICA DE LABORATORIO					
CARRERA: Ingeni	ería	Electró	nica	ASIGNATURA: Teoría de Control I					
NRO. PRÁCTICA:	2	TÍTULO DE LA PRÁCTICA: EXPERIMENTACIÓN Y PRUEBAS DEL SERVOMOTOR EN EL EJE X.							
OBJETIVOS:									
Conocer la con	exió	n del m	otor-Χ ι	ubicado en la plataforma inercial de 2-GDL.					
 Identificar el movimiento en el eje-X del plato donde se encuentra la pantalla táctil resistiva. 									
 Crear un progr motor-X. 	ama	en Sin	nulink p	ara enviar los valores de ángulos hacia el					
 Generar una s táctil resistiva u 	eñal Ibica	sinuso Ida en e	idal par el plato.	a el movimiento en el eje-X de la pantalla					
	1.	Alimen 110VA	itar la p .C.	olanta "Plataforma inercial de 2-GDL" con					
	2.	Confirm	mar qu encia no	e la botonera tipo hongo o paro de					
	3.	Encen	der la p	anta presionando el botón de marcha.					
INSTRUCCIONES	4.	Verifica	ar que s	e cuente con paquetes y drivers necesarios					
		para la	. comun	icación de la planta (PRÁCTICA #1).					
	5.	Conec	tar la	olanta "Plataforma inercial de 2-GDL" al					
	computador utilizando el cable USB macho-macho.								
	6.	6. Visualizar las luces indicadoras de encendido en la planta.							
ACTIVIDADES POR DESARROLLAR									
ubicado en la pla	gran nta "	na para Platafoi	ia esci rma ine	itura de los valores angulares del motor-X rcial de 2-GDL".					

- **2.** Agregar bloque de saturación para limitar el movimiento angular del plato.
- **3.** Envío de los valores de ángulos hacia el motor-X usando el bloque de escritura para servomotores.
- **4.** Agregar una señal sinusoidal con los valores de ángulos para un movimiento continuo del plato.

<u>Paso 1.</u> Creación del programa para la escritura de los valores angulares del motor-X ubicado en la planta "Plataforma inercial de 2-GDL".

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos desde la librería de Simulink.



Figura 4. 22: Simulink modelo en blanco y Librería para Arduino.

Con un bloque de valor constante se puede asignar el ángulo que se envía como escritura para el servomotor.



Figura 4. 23: Bloque "Constant" en la librería de Simulink.

Se crea un modelo en Simulink utilizando, el bloque de escritura estándar para servos y el bloque constante, donde el valor constante ingresa directamente al bloque del servomotor. El pin designado para el motor-X es el número 9.



Figura 4. 24: Modelo para escritura de ángulos al motor-X.

Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Agregando el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Saturation

Figura 4. 25: Bloque "Constant" en la librería de Simulink.

El motor-X trabaja de 90º a 150º estos son los valores que se configuran para el bloque de saturación.

	Block Parameters: Saturation X 150 - 90				
	Saturation				
	Limit input signal to the upper and lower saturation values.				
	Main Cignal Attributor				
	150				
	Lower limit:				
	90				
	< > ×				
	OK Cancel Help Apply				
Fig	ura 4 26: Parámetros del Bloque "Constant"				
i iy	ara Ti ZV. I arametros del Dioque Constant.				
Paso 3. Envío de lo	os valores de ángulos hacia el motor-X usando el bloque				
de escritura para s	ervomotores.				
Antes de ejecutar,	se debe seleccionar el hardware a utilizar, la placa Arduino				
DUE, más el tiempo	b de muestreo en el Workspace de Ts = 0.01 . Se aplican y				
aceptan los cambios	s. El modo de simulación es externo y un tiempo infinito como				
parada de la simulación. Se ejecuta el programa.					
	02 - Simulink − □ X				
File Edit	rew Display Diagram Simulation Analysis Code Tools Help				
2 🗁 -					
PRACTICA_02					
	PLATAFORMA				
<u>C3</u>	INERCIAL				
	ARDUINO				
(@)	Ángulo Motor X Saturation X 150 - 90 MOTOR EJE X				
	~				
Ready	150% FixedStepAuto				
Figura 4. 27: Ángulos para motor-X con límites de saturación.					
Paso 4. Agregar u	na señal sinusoidal con los valores de ángulos para un				
movimiento contin	uo del plato.				
Dene evidencien					
Para evidenciar un r	novimiento oscilatorio del plato, se agrega un bloque de onda				
sinusoidal. Con los s	siguientes parametros:				
م المناظمة المع					
Amplitua: 30 Bioo: 400					
DidS. 120					

- Frecuencia: 6.28 rad/sec
- Tiempo de muestreo: 0.01 sec

A continuación se muestra la configuración del bloque en mención con ayuda del bloque "Sine Wade" que se encuentra en la Liberia de arduino en la librería de arduino

Block Parameters: Sine Wave Function						
Time (t): Use simulation time						
Amplitude:						
30						
Bias:						
120						
Frequency (rad/sec):						
6.28						
Phase (rad):						
0						
Sample time:						
Ts						
Figura 4. 28: Parametros del Bloque "Sine Wave".						
Nuevamente se ejecuta el programa con la onda sinusoidal para comprobar el movimiento del plato.						
File Edit View Display Diagram Simulation Analysis Code Tools Help						
100 ▼ Image: Second s						
Function X 150 - 90 MOTOR EJE X						
©≣						
Running the model on 'Arduino Due' View diagnostics 150% T=11.630 auto(FixedStepDiscrete)						
i igura 4. 23. Fractica #2 ejecutada en tiempo real.						
 RESULTADO(S) OBTENIDO(S) Se comprueba la posición del plato, cuando se envía un valor de ángulo de 150°. 						
PRACTICA_02 - Simulink - 🗆 🗙						
File Edit View Display Diagram Simulation Analysis Code Tools Help						
▶ ▼ ■ ↓ ● ■ ● ● ■ ● ● ■ ● ● ● ■ ● </td						
PRACTICA_02						
PLAIAFORMA INERCIAL						
Ángulo Motor X Saturation Pin 9 X 150 - 90 MOTOR E IE X						
Figura 4 30: Valor de 150% enviado al servomotor						



4.3 PRÁCTICA # 3

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR			GUIA DE PRÁCTICA DE LABORATORIO					
CARRERA: Ingeniería Electrónio			nica	ASIGNATURA: Teoría de Control I				
NRO. PRÁCTICA:	3	TÍTULO DE LA PRÁCTICA: EXPERIMENTACIÓN Y PRUEBAS DEL SERVOMOTOR EN EL EJE Y.						
OBJETIVOS:								
• Conocer la conexión del motor-Y ubicado en la plataforma inercial de 2-GDL.								
 Identificar el movimiento en el eje-Y del plato donde se encuentra la pantalla táctil resistiva. 								
 Crear un programa en Simulink para enviar los valores de ángulos hacia el motor-Y. 								
• Generar una señal sinusoidal para el movimiento en el eje-Y de la pantalla								
táctil resistiva ubicada en el plato.								
	 Alimentar la planta "Plataforma inercial de 2-GDL" con 110VAC. 							
	2.	Confirmar que la botonera tipo hongo o paro de						
	2	emergencia no este accionada.						
INSTRUCCIONES	J.	Verificar que se queste con pequetes y drivers pessestion						
		para la		icación de la planta (PRÁCTICA #1).				
	5.	Conectar la planta "Plataforma inercial de 2-GDI" a						
	_	compu	tador ut	ilizando el cable USB macho-macho.				
	6.	Visualizar las luces indicadoras de encendido en la plant						
ACTIVIDADES POR DESARROLLAR								
1. Creación del programa para la escritura de los valores angulares del motor-Y								
ubicado en la pla	ubicado en la planta "Plataforma inercial de 2-GDL".							

- 2. Agregar bloque de saturación para limitar el movimiento angular del plato.
- **3.** Envío de los valores de ángulos hacia el motor-Y usando el bloque de escritura para servomotores.
- **4.** Agregar una señal sinusoidal con los valores de ángulos para un movimiento continuo del plato.

<u>Paso 1.</u> Creación del programa para la escritura de los valores angulares del motor-Y ubicado en la planta "Plataforma inercial de 2-GDL".

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos desde la librería de Simulink.



Figura 4. 34: Simulink modelo en blanco y Librería para Arduino.

Con un bloque de valor constante se puede asignar el ángulo que se envía como escritura para el servomotor.



Figura 4. 35: Bloque "Constant" en la librería de Simulink.

Se crea un modelo en Simulink utilizando, el bloque de escritura estándar para servos y el bloque constante, donde el valor constante ingresa directamente al bloque del servomotor. El pin designado para el motor-Y es el número 10.



Figura 4. 36: Modelo para escritura de ángulos al motor-Y.

Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Se agrega el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Saturation

Figura 4. 37: Bloque "Constant" en la librería de Simulink.

El motor-Y trabaja de 20° a 80° estos son los valores que se configuran para el bloque de saturación.

	Plack Parameter: Sturation V 90, 20
	Saturation
	Limit input signal to the upper and lower saturation values.
	Main Cional Attributes
	Upper limit:
	80
	Lower limit:
	20
	<
	OK Cancel Heln Anniv
Figure	ance nep reprint
Figur	a 4. 36: Parametros del Bioque Constant.
Paso 3. Envio de los	s valores de angulos nacia el motor-y usando el bioque
de escritura para sei	rvomotores.
A (1 ·)	
Antes de ejecutar, se	ebe seleccionar el hardware a utilizar, la placa Arduino
DUE, mas el tiempo	de muestreo en el workspace de TS = 0.01. Se aplican y
aceptan los campios.	El modo de simulación es externo y un tiempo infinito como
parada de la simulació	on. Se ejecuta el programa.
	INERCIAL
	ARDUINO
	Angulo Motor Y Saturation Y 80 - 20 MOTOR FUE Y
Figura 4. 39:	Ángulos para el motor-Y con límites de saturación.
Paso 4. Agregar una	a señal sinusoidal con los valores de ángulos para un
movimiento continue	o del plato.
Para evidenciar un mo	ovimiento oscilatorio del plato, se agrega un bloque de onda
sinusoidal. Con los sig	guientes parámetros:
 Amplitud: 30 	
 Bias: 50 	
• Frecuencia: 6.	28 rad/sec
 Tiempo de mu 	estreo: 0.01 sec
·	
	Block Parameters: Sine Wave Function1
	Amplitude:
	JU İ
	50
	Frances (rad/col)
	riequency (rau/sec).
	6.28 [
	6.28 i Phase (rad): i
	6.28 Phase (rad): 0 Sample time:
	6.28 i Phase (rad): i 0 i Sample time: i Ts i
	6.28 Phase (rad): 0 Sample time: Ts OK
Figure	6.28 Phase (rad): 0 Sample time: Ts OK Cancel Help Apply A 40: Parámetros del Plaque "Sino Waye"



• Se comprueba la posición del plato, cuando se envía un valor de ángulo de 90°.







Figura 4. 45: Posición del plato a 20º en el eje-Y.

CONCLUSIONES:

- Se obtiene una comunicación exitosa con el motor-Y de plataforma inercial de 2-GDL con el software Matlab/Simulink.
- El bloque de escritura para el servomotor recibe valores directos de ángulos sin ningún tipo de conversión de datos, por lo que facilita el control del mismo.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.

4.4 PRÁCTICA # 4

UNIVERSIDAD POLITECNICA SALESIANA ECUADOR		GL	JIA DE PRÁCTICA DE LABORATORIO				
CARRERA: Ingeniería Electrónica			nica	ASIGNATURA: Teoría de Control II			
NRO. PRÁCTICA:	4	TÍTULO DE LA PRÁCTICA: CREACIÓN DEL CONTROL PID PARA EL EJE X.					
OBJETIVOS:							
• Creación de un controlador PID para la estabilización del objeto sobre la							
pantalla táctil resistiva.							
Utilizar el bloque PID y sus atributos.							
• Transformar la señal obtenida de la pantalla táctil resistiva a variable doble.							
Realizar filtrado de la señal obtenida de la pantalla táctil resistiva.							
• Crear un programa en Simulink para enviar la señal de control del PID en el							
eje X a la plata	form	a inerci	al de 2-	GDL.			
	1.	Alimentar la planta "Plataforma inercial de 2-GDL" con 110VAC.					
	2.	Confirr emerge	nar qu encia no	e la botonera tipo hongo o paro de o esté accionada.			
	3.	Encen	der la p	lanta presionando el botón de marcha.			
INSTRUCCIONES	4.	Verifica	ar que s	e cuente con paquetes y drivers necesarios			
		para la	comun	icación de la planta (PRÁCTICA #1).			
	5.	Conec	tar la	planta "Plataforma inercial de 2-GDL" al			
		compu	tador u	ilizando el cable USB macho-macho.			
	6.	Visuali	zar las l	uces indicadoras de encendido en la planta.			
ACTIVIDADES POR DESARROLLAR							
 Creación del algoritmo de control PID para el eje X de la panta "Plataforma inercial de 2-GDL" para la estabilización de la esfera en la referencia. 							
2. Agregar bloque de saturación para limitar el movimiento angular del plato.							

- **3.** Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.
- **4.** Agregar el bloque PID discreto.
- 5. Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

<u>Paso 1.</u> Creación del algoritmo de control PID para el eje X de la panta "Plataforma inercial de 2-GDL" para la estabilización del objeto en la referencia.

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos y el bloque "Analog Input" desde la librería de Simulink.



Figura 4. 46: Simulink modelo en blanco y Librería para Arduino.

Se procede añadir el bloque DATA CONVERT a la salida del bloque de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico.


Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Se agrega el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Saturation

Figura 4. 48: Bloque "Constant" en la librería de Simulink.

El motor-Y trabaja de 90° a 150° estos son los valores que se configuran para el bloque de saturación.

	Block Parameters: Saturation X 1	150 - 90 ×	
	Saturation		<u>^</u>
	Limit input signal to the upper a	nd lower saturation values.	
	Main Signal Attributes		
	Upper limit:		
	150		
	Lower limit:		
	90		
	<	>	Ť
	OK Cancel	Help Apply	1
			_
I	Figura 4. 49: Parámetros	del Bloque "Constan	t".
munetra el re	sultado de agregar el blog	ue de "Saturation"	
	sultado de agregar el bloq	ue de "Saturation"	- 0
		ue de "Saturation"	- 0
weter- indust i for yee performer performer of the second		ue de "Saturation"	- 0
muestra el re	sultado de agregar el bloq	ue de "Saturation"	- 0
Interest of the second	sultado de agregar el bloq	jue de "Saturation"	- 0
wetter imulat i for your puter bagen greaten gelyn i for your puter bagen gelyn i for you	sultado de agregar el bloq	ue de "Saturation"	~ *
witter: Smuth	ARDUINO	ue de "Saturation"	
muestra el re	ARDUINO	ue de "Saturation"	D Data
muestra el re	ARDUINO Pin 9	ue de "Saturation"	D double Data Conver X
muestra el re	ARDUINO Pin 9 Servo X Write	que de "Saturation" ARDUING Pin 0 Analog Input X	D double Data Conver X
muestra el re	ARDUINO Pin 9 Servo X Write	que de "Saturation" ARDUING Pin 0 Analog Input X	D double Data Conver X
muestra el re	ARDUINO Pin 9 Servo X Write	que de "Saturation" ARDUING Pin 0 Analog Input X	Deta Conver X
muestra el re	ARDUINO Pin 9 Servo X Write	que de "Saturation" ARDUING Pin 0 Analog Input X	D double Data Conver X

<u>Paso 3.</u> Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.

La señal que se obtenida del bloque Analog Input que nos muestran los valores en voltaje del objeto sobre la pantalla táctil resistiva se la debe filtrar para reducir la mayor cantidad de ruido.

Para realizar lo antes mencionado se debe añadir el bloque "Kalman Filter" que se lo encuentra en la librería de Simulink como se muestra a continuación.



Figura 4. 51: Bloque "Kalman Filter" en la librería de Simulink.

Para la utilización del filtro se debe modificar los parámetros del bloque "Kalman Filter" para un correcto uso del bloque antes mencionado.

Estimate the state of measurements. This can also use the cu	of a dynamic system from a s s block can use the previousl rent measurement and the p	series of incomplete and/or y estimated state to predict predicted state to estimate t	noisy the current state. It he current state	
All filters have the s noise covariance, b state, measuremen	ame state transition matrix, it their state, measurement, ; enable, and MSE signals, e	measurement matrix, initial enable, and MSE signals are ach column corresponds to	conditions, and e unique. Within the a filter.	
Parameters				
Number of filters:	1		:	
Enable filters:	Always		•	
Initial condition for State transition ma Process noise cova Measurement mate Measurement noise Outputs	estimated error covariance: trix: [1] riance: 0.005*eye(1 x source: Specify via matrix: [1] e covariance: eye(1)	10*eye(1)	1 1 1 1 1 1	
Outputs	d massurament <7 est>		rement <7 prd>	
Output estimate Output estimate	d state <x_est></x_est>	Output predicted measure Output predicted state	<x_prd></x_prd>	
Output MSE of a	stimated state <mse_est></mse_est>	Output MSE of predicted		
		<u>OK</u> <u>C</u> ancel <u>I</u>	<u>i</u> elp <u>Apply</u>	





Paso 4. Agregar el bloque PID discreto.

A continuación, se debe añadir el bloque PID discreto, adicional se crea un bloque constante indicando la referencia. Posterior se cierra el circuito creando el controlador PID para el eje X en tiempo discreto, todos los bloques antes mencionados se encuentran en la librería de Simulink.



121

Se añade el bloque constante para la referencia y se cierra el lazo con la salida del bloque "Kalman Filter".



Figura 4. 55: Bloque "Constant" para la referencia del controlador PID.

A continuación, se muestra el algoritmo de control PID para el eje X y se modifican los parámetros del PID.



We want the there there the the the the the the the the the th		😼 Block Parameters: D	iscrete PID X				>	×
<form></form>		PID Controller						^
with the second seco		Simulink Control Desi	s continuous- and discrete it, and signal tracking. You gn).	time PID control alg can tune the PID ga	iorithms and includes ad ains automatically using t	vanced features such the 'Tune' button (r	as anti- requires	
<form></form>		Controller: PID		▼ F	orm: Parallel		•	
Windersteiner		Time domain:			Discrete-time settings	Forward Eule	er 👻	
Since the state of the state database Proportion []		Continuous-time Discrete-time			Filter method:	Forward Eule	er 🔻	
The state of th		O Discrete time			Sample time (-1 for inhe	erited): Ts	:	
Sector State St		Main PID Advance Controller parameter	ed Data Types Stat 's	e Attributes				
Put starting (b) is intermediated in the starting is intermedia		Source:	internal		• 8	Compensator formul	a	
<pre>www.ie (i) with the density is the filled density is the fill</pre>		Proportional (P):	-0.02					
Winder Of Control 		Integral (I):	-0.0290			1	N	
<pre>interdence (b): interest servases</pre>		Derivative (D):	-0.02		E	$P + I \cdot T_s \frac{1}{z-1} + D \frac{1}{1}$	+ N.T. 1	
The contract (b) 100 Image: contract (c) 100 Image: contract			Use filtered derivative			1.	z-1	
Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualizarán las diferentes señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualización de las señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualización de las señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualización de las señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope'' se visualización de las señales para resultados, el bloque en mención se lo encuentra en la librería Image: Scope services de los señales para resultados de las señales para resula distados de las señales para resultados de las señal		Filter coefficient (N):	100					
gura 4. 57: Parametrización del bloque PID para el eje X. gura 4. 57: Parametrización del bloque PID para el eje X. egar el bloque "Scope" se visualizarán las diferentes señales para s resultados, el bloque en mención se lo encuentra en la librería se resultados, el bloque en mención se lo encuentra en la librería					Tune			
<pre>winder of control of the second of the</pre>		Initial conditions						
Provide and the network of the ne		Source: internal					-	
re: interview		Integrator: 0					:	
The second se		Filter: 0					:	
gura 4. 57: Parametrización del bloque PID para el eje X. gura 4. 57: Parametrización del bloque PID para el eje X. egar el bloque "SCOPE" para la visualización de las señal a del bloque "Scope" se visualizarán las diferentes señales para se resultados, el bloque en mención se lo encuentra en la librería se formado de las señales para la visualización de las señales para se resultados, el bloque en mención se lo encuentra en la librería		External scorety in						
<text><text><text><text></text></text></text></text>		External reset: none	linendelen				•	
<text><text><text><text></text></text></text></text>		Ignore reset when	iinearizing					
<text></text>		rnable zero-crossin	ig detection					¥
gura 4. 57: Parametrización del bloque PID para el eje X. egar el bloque "SCOPE" para la visualización de las señal a del bloque "Scope" se visualizarán las diferentes señales para a resultados, el bloque en mención se lo encuentra en la librería Servinisk Libray Brovser Servinisk Libray Brovser 		0			ОК	Cancel Help	Apply	
s resultados, el bloque en mención se lo encuentra en la librería	Agreç as.	gar el blo	oque "SC	OPE"	para la v	visualiza	ación	de las seña
 Simulink Library Browser Scope Dashboard Scope Scope Dashboard Scope Dashboard Scope Scope Dashboard Scope Dashboard Scope Scope Dashboard Scope Scope Scope Dashboard Scope Scope <l< th=""><th>Agreç s. /uda (</th><th>gar el blo</th><th>oque "SC e "Scope"</th><th>COPE"</th><th>para la v Jalizarán</th><th>visualiza las dife</th><th>ación rentes</th><th>de las señal</th></l<>	Agreç s. /uda (gar el bl o	oque "SC e "Scope"	COPE"	para la v Jalizarán	visualiz a las dife	ación rentes	de las señal
 Simulink Library Browser Scope Dashboard Discrete Lockup Tables Discrete Logic and Bit Operations Lockup Tables Signal Attributes Sources User-Defined Functions Additional Math Discrete Actrospace Biockset Additional Math A Discrete Actrospace Biockset Additional Math B Discrete Communications System Toolbox Communications System Toolbox OPS System Toolbox HDL Support Embedded Coder Furre Locit Toolbox 	Agreç as. ayuda o de los	gar el blo del bloqu resultado	oque "SC e "Scope" s, el bloqu	COPE" se visu ue en m	para la v ualizarán iención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Scope Image: Init of the second s	greç da o los	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu	COPE" se visu ue en m	para la v ualizarán iención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Search Results: Scope <	greç ıda o los	gar el bloqui del bloqui resultado	oque "SC e "Scope" s, el bloqu	COPE" se visu ue en m	para la v ualizarán iención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Simulink - 8 (simulink/Commonly Used Blocks/Scope) Simulink Commonly Used Blocks Continuous Dashboard Discontinuities Discontinuities Discrete Logic and Bit Operations Model Verification Signal Routing Sinks Sources User-Defined Functions > Additional Math & Discrete Acrospace Blockset Acrospace Blockset Acrospace Blockset Communications System Toolbox Control System Toolbox Obser System Toolbox DSP System Toolbox	greç Ida d Ios	gar el bloqui del bloqui resultado	oque "SC e "Scope" is, el bloqu	Se visu se visu ue en m	para la v ualizarán iención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
 ✓ Simulink Commonly Used Blocks Continuous Dashboard Discontinuities Discrete Logic and Bit Operations Model Verification Model Verifica	uda (los	gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser	Se visu ue en m	para la v ualizarán iención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Commonly Used Blocks Continuous Dashboard Discontinuities Discrete Logic and Bit Operations Lockup Tables Math Operations Model Verification Model-Wide Utilities Ports & Subsystems Signal Attributes Signal Routing Sinks Sources User-Defined Functions > Additional Math & Discrete > Additional System Toolbox Communications System Toolbox > DSP System Toolbox HDL Support > Discrete Toolbox > DSP System Toolbox HDL Support > Embedded Coder > furzu Lodi Trodivov > DSP System Toolbox HDL Support > furzu Lodi Trodivov	greg Ida o Ios	gar el bloque del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser	Se visu ue en m	para la v ualizarán bención se	visualiza las dife e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Dashboard Discrete Discrete Logic and Bit Operations Lobup Tables Math Operations Math Operations Model Verification Model-Wide Utilities From Ports & Subsystems Signal Attributes Signal Attributes From Signal Attributes Goto Tag Sinks Sources User-Defined Functions Goto Tag A Aerospace Blockset Image: Sources A Aerospace Blockset Image: Sources Communications System Toolbox Scope Observer Ysion System Toolbox Scope DSP System Toolbox Scope DSP System Toolbox Support DSP System Toolbox Windify and factors DSP System Toolbox Support Dimeded Coder Support Support Support Storm Low System Toolbox Support Dimeded Coder Support Support Support </td <td>greg da o los</td> <td>gar el blo del bloque resultado</td> <td>oque "SC e "Scope" is, el bloqu ary Browser</td> <td>COPE" se visu ue en m</td> <td>para la v ualizarán hención se</td> <td>visualiza las difer e lo enci</td> <td>ación rentes uentra</td> <td>de las señal s señales para a en la librería</td>	greg da o los	gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser	COPE" se visu ue en m	para la v ualizarán hención se	visualiza las difer e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Discrete Logic and Bit Operations Lockup Tables Model Wide Utilities Ports & Subsystems Signal Attributes Signal Attributes Signal Routing Sinks Sources User-Defined Functions Additional Math & Discrete A Aerospace Blockset Additional Math & Discrete A Aerospace Blockset Communications System Toolbox Communications System Toolbox Obst Acquisition roolbox Obst System Toolbox Dest Acquisition roolbox Obst System Toolbox Dobs HDL Support Embedded Coder Simra Lowin Toolbox Discrete HTOOLS Discrete HTOOLS	greç da (los	gar el bloqué del bloqué resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found)	COPE" se visu ue en m	para la v ualizarán iención se	visualiza las difei e lo enci	ación rentes uentra	de las señal s señales para a en la librería
Logic and Bit Operations Lokup Tables Math Operations Model Verification Model Verification Model Verification Signal Attributes Signal Routing Sinks Sources User-Defined Functions (A) Additional Math & Discrete Acrospace Blockset Additional Math & Discrete Scope Additional Math & System Toolbox Discrete Communications System Toolbox Scope Does Psystem Toolbox Scope Data Acquisition Toolbox Suport Disp System Toolbox Suport Disp System Toolbox Suport Prizeru Indie Toolboy Three-axis Gyroscope • Disp System Toolbox Suport • Firzeru Indie Toolboy Strema Coolbox - 2	greg da d los	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found)	COPE" se visu ue en m	para la v ualizarán iención se ⊧= ② k - 8 (simulink/Commo	visualiza las difei e lo enci -	ación rentes uentra	de las señal s señales para a en la librería
Lockup Tables Math Operations Model Verification Model Verification Model Verification Model Verification Model Verification Model Verification Model Verification Model Verification Model Verification Model Verification Signal Routing Sinks Sources User-Defined Functions > Additional Math & Discrete > Additional Math	l reç la c os	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found)	COPE"	para la v ualizarán iención se ⊧ ②	visualiza las difei e lo enci -	ación rentes uentra	de las seña s señales para a en la librería
 Houst Operations Model Verification Model Verification Model Verification Model Verification Signal Attributes Signal Attributes Signal Routing Sinks Sources User-Defined Functions Additional Math & Discrete A Additional System Toolbox Communications System Toolbox Communications System Toolbox Dose /li>	la d	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks	COPE"	para la v ualizarán iención se ∗ 8 (simulink/Commo scope	visualiza las difei e lo enci niy Used Blocks/Sco Dashboard Sco	ación rentes uentra	de las señal s señales para a en la librería
Model-Wide Utilities From Goto Ports & Subsystems Signal Attributes Image: Computer Name Image: Computer Name Sinks Sources Image: Computer Name Image: Computer Name Image: Computer Name A Additional Math & Discrete A Additional Math & Discrete Image: Computer Name Image: Computer Name Image: Computer Name A Additional Math & Discrete A Acrossition System Toolbox Image: Computer Name Image: Computer Name Image: Computer Name Communications System Toolbox Computer Vision System Toolbox Image: Computer Name Image: Computer Name Image: Computer Name DSP System Toolbox DSP System Toolbox Image: Computer Name Image: Computer Name Image: Computer Name Image: Computer Name DSP System Toolbox DSP System Toolbox Image: Computer Name		gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser (21 Blocks found) 'Used Blocks ittes BR Operations bles server	COPE"	para la v ualizarán hención se	visualiza las dife e lo enci - ny Used Blocks/Sco Dashboard Sco V rai	ación rentes uentra	de las seña s señales para a en la librería
Ports & Subsystems Signal Attributes Signal Authoutes Signal Routing Sinks Sources User-Defined Functions > Additional Math & Discrete > Auditional Math & Discrete > Autorison System Toolbox > Communications System Toolbox > Communications System Toolbox > Communications System Toolbox > Communications System Toolbox > DSP System Toolbox HDL Suport > Firmu Indie Trollovy < Firmu Indie Trollovy < Support System Toolbox - 2 ✓		gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser (21 Blocks found) Used Blocks s itties Bit Operations bles ations fication	COPE" se visu ue en m	para la v ualizarán hención se -> ② k - 8 (simulink/Commo scope	visualiza las difei e lo enci niy Used Blocks/Sco Dashboard Sco V(A)	ación rentes uentra	de las señal s señales para a en la librería
Signal Routing Sinks Sources User-Defined Functions > Additional Math & Discrete > Acrospace Blockset > Audio System Toolbox > Communications System Toolbox Computer Vision System Toolbox Computer Vision System Toolbox Data Acquisition Toolbox > DSP System Toolbox HDL Support > Furzu Louic Toolbox > Furzu Louic Toolbox	la o os	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks s ittes Bit Operations bles ations ification	COPE"	para la v ualizarán lención se * 2 k - 8 (smulink/Commo Scope	visualiza las difei e lo enci ny Used Blocks/Sco Dashboard Sco Vai Goto	ación rentes uentra	de las señal s señales para a en la librería
Sinks Sources User-Defined Functions > Additional Math & Discrete > Audio System Toolbox > Communications System Toolbox HDL Sup > Computer Vision System Toolbox > Computer Vision System Toolbox Data Acquisition Toolbox > DSP System Toolbox HDL Support > Embedded Coder > Furzu Indie Toolbox > Embedded Coder > Furzu Indie Toolbox > Embedded Coder > Furzu Indie Toolbox	line (da (os	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) 'Used Blocks s itites Bit Operations bles Bit Operations bles bles bles bles bles bles bles ble	COPE"	para la v ualizarán iención se ⊧= ② k - 8 (simulink/Commo scope [A] > From	visualiza las difei e lo enci niy Used Blocks/Sco Dashboard Sco V(A) Goto		de las señal s señales para a en la librería
Sources User-Defined Functions > Additional Math & Discrete > Aerospace Blockset > Auditional Math & Discrete > Aurospace Blockset > Communications System Toolbox > Communications System Toolbox HDL Suj > Computer Vision System Toolbox Data Acquisition Toolbox > DSP System Toolbox > DSP System Toolbox HDL Support > Embedded Coder > Furzu Louis Toolbox > DSP System Toolbox HDL Support > Embedded Coder > Furzu Louis Toolbox > DSP System Toolbox HDL Support > Embedded Coder > Furzu Louis Toolbox > DSP System Toolbox HDL Support > Furzu Louis Toolbox	reç la (os	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) 'Used Blocks titles BR Operations bles attions fication fication bioystems bibystems bibystems	COPE"	para la v ualizarán iención se	visualiza las dife e lo enci ny Used Blocks/Sco Dashboard Sco (1) Goto	ación rentes uentra	de las señal s señales para a en la librería
∪ Ger -Verinder Functions >> Additional Math & Discrete >> Additional Math & Discrete >> Auditional Math & Discrete >> Autional Math & Discrete >> Auditional Math & Discrete >> Communications System Toolbox >> Communications System Toolbox >> Communications System Toolbox >> DSP System Toolbox >> Einzu Louis Toolbox >		gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser (21 Blocks found) (21	COPE"	para la v ualizarán hención se -> 2 k - 8 (simulink/Commo k - 8 (simulink)	visualiza las difei e lo enci niy Used Blocks/Sco Dashboard Sco $\sqrt{[A]}$ Goto	ación rentes uentra	de las señal s señales para a en la librería
 Aerospace Blockset Audio System Toolbox Communications System Toolbox Computer Vision System Toolbox Computer Vision System Toolbox Computer Vision System Toolbox Data Acquisition Toolbox DSP System Toolbox HDL Support Embedded Coder Furro Louis Toolbox Serve Louis HDL Support Furro Louis Toolbox Serve Louis HDL Support Furror Louis Toolbox Serve Louis HDL Support Furro Louis Toolbox Serve Louis HDL Support Furro Louis Toolbox Serve Louis HDL Support Furro Louis Toolbox Serve Louis Toolbox Serve Louis HDL Support Furro Louis Toolbox Serve Louis Toolbox	a (gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks i ities BR Operations bles ations frication le Utilities bisystems ibiotes tring	COPE"	para la v ualizarán lención se → ② k - 8 (simulink/Common Scope (A) From (A) Goto Tag	visualiza las difei e lo enci ny Used Blocks/Son Dashboard Son V(A) Goto	ación rentes uentra	de las señal s señales para a en la librería
 Audio System Toolbox Communications System Toolbox Computer Vision System Toolbox Computer Vision System Toolbox Computer Vision System Toolbox Data Acquisition Toolbox DSP System Toolbox HDL Support Embedded Coder Furzu Louic Toolbox Struct Louic To	eg a o s	gar el blo del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks (21 Blocks found) Used Blocks Blc Operations bles ations fication fication biotes titing ed Functions	COPE"	para la v ualizarán jención se ⊧= ② k - 8 (simulink/Commo ∑ scope [A] From (A) Goto Tag Visibility	visualiza las difei e lo enci ny Used Blocky/Sco Dashboard Sco VII Goto	ación rentes uentra	de las señal s señales para a en la librería
 Communications System Toolbox HDL Sup Computer Vision System Toolbox HDL Sup Computer Vision System Toolbox DSP System Toolbox HDL Support Scope XY Graph Aerospace Blockset - 1 (rad/s) (gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser (21 Blocks found) Used Blocks (21 Blocks found) Used Blocks Bit Operations bles dities Bit Operations bles dities bles tittes bles tittes bles dities dities bles dities bles dities dities bles dities dities dities bles dities bles dities ditte	COPE"	para la v Jalizarán Jención se → ② k - 8 (simulink/Commo Scope (A) From (A) Goto Tag	visualiza las dife e lo enci ny Used Blocks/Sco Dashboard Sco Cato Cato Floating Floating	ación rentes uentra	de las señal s señales para a en la librería
 > Computer Vision System Toolbox Control System Toolbox Data Acquisition Toolbox > DSP System Toolbox > DSP System Toolbox HDL Support > Embedded Coder > furrou I nail- Toolbov > Three-axis Gyroscope > DSP System Toolbox - 2 		gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ay Browser (21 Blocks found) 'Used Blocks f itties Bit Operations bies ations fication le Utilities bisystems ibutes ting ed Functions Math & Discrete cket Toolbox	COPE"	para la v ualizarán hención se -> 2 k - 8 (simulink/Commo k - 8 (simulink/Commo k - 8 (simulink/Commo k - 8 (simulink/Commo k - 9 (simulink) - 9 (simulink/Commo k - 9 (simulink) -	visualiza las difer e lo encl ny Used Blocks/Scor Dashboard Sco V[A] Goto Poating Scope	ación rentes uentra	de las señal s señales para a en la librería
Control System Toolbox >		gar el blo del bloqué resultado	oque "SC e "Scope" is, el bloqu ay Browser (21 Blocks found) Used Blocks dittes Bit Operations bles ations fication le Utilites bisystems bisystems bisystems bisystems tring de Functions Math & Discrete ciceet Toolbox	COPE"	para la v ualizarán iención se · 2 ·	visualiza las difei e lo encl ny Used Blocks/Sco Dashboard Sco Coto Coto Floating Scope	ación rentes uentra	de las señal s señales para a en la librería
> DSP System Toolbox > > DSP System Toolbox HDL Support > > Embedded Coder > < Firzzu Louie Toolbox		gar el blo del bloqué resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks titles Bit Operations bles ations le Utilités biss ations le Utilités biss ations le Utilités biss ations le Utilités biss ations fifation le Utilités bissystems biotes tring	Sope"	para la v ualizarán jención se ↓ 2 k - 8 (simulink/Commo ↓ scope (A) From (A) Goto Tag Visibility ↓ Scope	visualiza las difei e lo enci ny Used Blocks/Sco Dashboard Sco (A) Goto Floating Scope X Graph	ación rentes uentra	de las señal s señales para a en la librería
DSP System Toolbox HDL Support Embedded Coder Furzy Louis Toolbox DSP System Toolbox - 2	reç la cos	gar el ble del bloque resultado	oque "SC e "Scope" s, el bloqu ary Browser (21 Blocks found) Used Blocks (21 Blocks found) Used Blocks Bit Operations bles ations fication le Utilités bibystems titug ed Functions Math & Discrete ckset Toolbox ns System Toolbox Hol S ns System Toolbox Hol S ns System Toolbox No n System Toolbox No n System Toolbox No n Toolbox	Sorter Structure	para la v ualizarán iención se → ② k - 8 (simulink/Commo K - 8 (simulink/Commo Scope (A) Scope Sco	visualiza las difei e lo enci ny Used Blocky/Sco Dashboard Sco VIA Goto Floating Scope XY Graph	ación rentes uentra	de las señal s señales para a en la librería
> Embedded Loder V > Firsty Lodie Toolbov V DSP System Toolbox - 2	da (los	gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ary Browser (21 Blocks found) 'Used Blocks (21 Blocks found) 'Used Blocks bles titles BR Operations bles ations fraction fractions bles bles ations fractions bles titles BR Operations bles titles BR Operations bles titles BR Operations bles ations fractions fractions fractions bles titles BR Operations bles titles BR Operations bles titles titles BR Operations bles titles BR Operations bles titles BR Operations bles titles BR Operations bles titles BR Operations bles titles BR Operations titles BR Operations titles titles BR Operations titles titles BR Operations bles tit	Sun Aerospu	para la v ualizarán hención se * 2 k - 8 (simulink/Commo k - 8 (simulink/Commo (A) Scope (A) From (A) Scope Scope Scope Scope Scope Scope Scope Scope Scope Scope Scope	visualiza las difei e lo enci ny Used Blocks/Sco Dashboard Sco Cat Goto Floating Scope XY Graph	ación rentes uentra	de las señal s señales para a en la librería
✓ DSP System Toolbox - 2 ✓	reç la o os	gar el blo del bloque resultado	oque "SC e "Scope" is, el bloqu ay Browser (21 Blocks found) (21 Blocks found	Sul Sul Sul Sul COPE" Sul Construction of the subscription of the subscript	para la v ualizarán hención se → ② k - 8 (simulink/Commo k - 8 (simulink/Commo (A) From (A) Scope ace Blockset - 1 w _{mess} (ræd/s)	visualiza las difer e lo encl ny Used Blocks/Scor Dashboard Sco V[A] Goto V[A] Goto Poating Scope XY Graph	ación rentes uentra	de las señal s señales para a en la librería
	a (gar el blo del bloque resultado	oque "Scope" e "Scope" is, el bloqu ary Browser (21 Blocks found) Used Blocks s titles Bit Operations bles ations fication le Utilites bles ations fication le Utilites bleystems blots ations fication s System Toolbox ns System Toolbox HD 1 5 no System Toolbox HD 1 5	Sul Visu Sul Visu Sul Acrospi Sul Acrospi Sul Acrospi Sul Acrospi Sul Acrospi Sul Three	para la v ualizarán hención se → ② k - 8 (simulink/Commo ↓ Scope (A) From (A) Scope Scop	visualiza las difei e lo encl 	ación rentes uentra	de las señal s señales para a en la librería

Г

123

Se procede a configurar el bloque para visualizar las señales del control aplicado al eje X.





Para proceder a poner en marcha nuestro algoritmo, se debe configurar el parámetro "sample time" de todos los bloques de simulink con la variable del tiempo de muestreo (Ts) creado en Matlab. El tiempo de muestreo Ts es de 0.01 y se lo debe de escribir en la ventana de comando de Matlab como se muestra a continuación.

Сс	ommand Window						
N	ew to MATLAB? See resources for Getting Started.						
	>> Ts=0.01						
	Ts =						
	0.0100						
fx	>>						

Figura 4. 60: Variable Ts en la ventada de comando de Matlab.

RESULTADO(S) OBTENIDO(S)

Se comprueba la posición del objeto sobre el plato, se encuentra en la posición deseada de manera visual, la referencia para el eje de las X es de 550.





Figura 4. 62: Posición del objeto en la referencia 550 para el eje X.

• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control PID para el eje X.

Donde la señal de color amarillo es la referencia del sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema



CONCLUSIONES:

- Se obtiene una comunicación exitosa con el motor-X de plataforma inercial de 2-GDL con el software Matlab/Simulink utilizando el algoritmo de control PID
- Se configura el bloque PID para el eje X y se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul que se muestra en la figura 4.63.

4.5 PRÁCTICA # 5

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR		GI	JIA DE PRÁCTICA DE LABORATORIO			
CARRERA: Ingenier	ría Ele	ctrónica	ASIGNATURA: Teoría de Control II			
	5 TÍ P	TÍTULO DE LA PRÁCTICA: CREACIÓN DEL CONTI PID PARA EL EJE Y.				
OBJETIVOS:						
Creación de un	contro	olador PID	para la estabilización del objeto sobre la			
pantalla táctil res	sistiva.					
Utilizar el bloque	PID y	sus atribut	os.			
Transformar la se	eñal o	otenida de	a pantalla táctil resistiva a variable doble.			
Realizar filtrado	de la s	eñal obteni	da de la pantalla táctil resistiva.			
Crear un program	ma en	Simulink p	ara enviar la señal de control del PID en el			
eje Y a la platafo	orma ir	ercial de 2-	GDL.			
INSTRUCCIONES	 Alimentar la planta "Plataforma inercial de 2-GDL" of 110VAC. Confirmar que la botonera tipo hongo o paro emergencia no esté accionada. Encender la planta presionando el botón de marcha. 					
	4. ve na	rilical que s	acción de la planta (PRÁCTICA #1)			
	 5. Conectar la planta "Plataforma inercial de 2 computador utilizando el cable USB macho-mac 					
	6. Vis	. Visualizar las luces indicadoras de encendido en la p				
		IDADES P	OR DESARROLLAR			
 Creación del algor inercial de 2-GDL" Agregar bloque de 	ritmo d para la satura	le control l a estabilizad	PID para el eje Y de la panta "Plataforma ción de la esfera en la referencia.			

- **3.** Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.
- **4.** Agregar el bloque PID discreto.
- **5.** Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

<u>Paso 1.</u> Creación del algoritmo de control PID para el eje Y de la panta "Plataforma inercial de 2-GDL" para la estabilización del objeto en la referencia.

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos y el bloque "Analog Input" desde la librería de Simulink.



Figura 4. 64: Simulink modelo en blanco y Librería para Arduino.

Se procede añadir el bloque DATA CONVERT a la salida del bloque de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico.



Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Se agrega el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Figura 4. 66: Bloque "Constant" en la librería de Simulink.

El motor-Y trabaja de 20º a 80º estos son los valores que se configuran para el bloque de saturación.

🎦 Block	Parameters: Satura	tion Y 80	- 20		X
Saturati	on				^
Limit inp	out signal to the u	pper and	lower saturat	tion values.	
Main	Signal Attributes	;			
Upper lin	nit:				
80					
Lower lin	nit:				
20					U
<				>	Ť
0	OK C	ancel	Help	Apply	

Figura 4. 67: Parámetros del Bloque "Constant".

Se muestra el resultado de agregar el bloque de "Saturation"



Paso 3. Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.

La señal que se obtenida del bloque Analog Input que nos muestran los valores en voltaje del objeto sobre la pantalla táctil resistiva se la debe filtrar para reducir la mayor cantidad de ruido.

Para realizar lo antes mencionado se debe añadir el bloque "Kalman Filter" que se lo encuentra en la librería de Simulink como se muestra a continuación.



Figura 4. 69: Bloque "Kalman Filter" en la librería de Simulink.

Para la utilización del filtro se debe modificar los parámetros del bloque "Kalman Filter" para un correcto uso del bloque antes mencionado.

Se muestra a continuación los bloques agregados al Simulink hasta el presente paso paras el diseño del algoritmo de control PID para el eje Y.



Figura 4. 71: Filtrado de la señal de la pantalla táctil resistiva.

Paso 4. Agregar el bloque PID discreto.

A continuación, se debe añadir el bloque PID discreto, adicional se crea un bloque constante indicando la referencia. Posterior se cierra el circuito creando el controlador PID para el eje Y en tiempo discreto, todos los bloques antes mencionados se encuentran en la librería de Simulink.







Figura 4. 73: Bloque "Constan" para la referencia del controlador PID.

A continuación, se muestra el algoritmo de control PID para el eje Y y se modifican los parámetros del PID.



Se procede a modificar	los valores del b	loque "Discr	ete PID".	
Pal Block Parameters	Discrete PID Y			<
PID Controller			,	^
This block implemen windup, external re Simulink Control De	nts continuous- and discrete-time PID co set, and signal tracking. You can tune th sign).	ntrol algorithms and includes a PID gains automatically using	idvanced features such as anti- g the 'Tune' button (requires	
Controller: PID		▼ Form: Parallel		
Time domain:		Discrete-time settings		
O Continuous-time		Integrator method:	Forward Euler	
Discrete-time		Filter method:	Forward Euler	
		Sample time (-1 for inner	nted): Is	
Main PID Advar Controller paramet	nced Data Types State Attributes ers			
Source:	internal		ompensator formula	
Proportional (P):	-0.02200	:		
Integral (I):	-0.01751	:		
Derivative (D):	-0.02215	: P	$+I \cdot T_8 - \frac{1}{1} + D - \frac{N}{1}$	
	Use filtered derivative		$\frac{z-1}{1+N}\cdot T_s \frac{1}{z-1}$	
Filter coefficient (N	I): 100	:		
		Tune		
Initial conditions				
Source: intern	al		•	
Integrator: 0			:	
Filter: 0			:	
External reset: non	e			
☐ Ianore reset whe	n linearizing			~
<			>	
		<u>O</u> K <u>C</u> anc	cel <u>H</u> elp <u>A</u> pply	
Figura 4. 75:	Parametrizació	n del bloque	PID para el	eje Y.
<u>Paso 5.</u> Agregar el bl obtenidas.	oque "SCOPE"	para la vis	sualización	de las señales
Con la ayuda del bloqu análisis de los resultado Simuliak	e "Scope" se vis os, el bloque en	sualizarán la mención se	as diferentes lo encuentra	señales para el en la librería de
	nan/ Browcer		— П X	
		3		
Scope		• = 🥑		
Search Results: S <<>>Page 1 of	cope 1 (21 Blocks found)			
Simulink	∧ ▼ Sim	ulink - 8 (simulink/Commonly U	Ised Blocks/Scope)	
Common	Ily Used Blocks			
Dashboa	rd		\sim	
Discontir Discrete	nuities	Econo	Dachhoard Scono	
Logic an	d Bit Operations	Scope	Dashboard Scope	
Lookup Math Op	erations	[A]	× [A]	
Model Ve	erification			
Model-W Ports &	ide Utilities Subsystems	From	Goto	
Signal At	tributes			
Signal Ro	outing	{A}		
Sinks		Cata Tan	<u> </u>	
User-Def	ined Functions	Visibility	Scope	
> Addition	al Math & Discrete			
> Audio System	n Toolbox		30	
> Communicati > Communicati	ions System Toolbox ions System Toolbox HDL Su	Econo	YX Craph	
> Computer Vis	sion System Toolbox	ospace Blockset - 1	AT Graph	
Data Acquisi	tion Toolbox	4(4)		
> DSP System > DSP System	Toolbox Toolbox HDL Support	ω _{meas} (rad/s)		
> Embedded C	oder	ree-axis Gyroscope		
S Fuzzy Logic	> DSF	System Toolbox - 2	~	
Figu	ra 4. 76: Bloque	"Scope" par	a el eie Y.	

Se procede a configurar el bloque para visualizar las señales del control aplicado al eje Y.





Para proceder a poner en marcha nuestro algoritmo, se debe configurar el parámetro "sample time" de todos los bloques de simulink con la variable del tiempo de muestreo (Ts) creado en Matlab. El tiempo de muestreo Ts es de 0.01 y se lo debe de escribir en la ventana de comando de Matlab como se muestra a continuación.



Figura 4. 78: Variable Ts en la ventada de comando de Matlab.

RESULTADO(S) OBTENIDO(S)

Se comprueba la posición del objeto sobre el plato, se encuentra en la posición deseada de manera visual, la referencia para el eje Y es de 550.





Figura 4. 80: Posición del objeto en la referencia 550 para el eje delas Y.

• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control PID para el eje Y.

Donde la señal de color amarillo es la referencia del sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema



CONCLUSIONES:

- Se obtiene una comunicación exitosa con el motor-Y de plataforma inercial de 2-GDL con el software Matlab/Simulink utilizando el algoritmo de control PID
- Se configura el bloque PID para el eje Y y se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul.

4.6 PRÁCTICA # 6

GUIA DE PRÁCTICA DE LABORATORIO							
CARRERA: Ingen	CARRERA: Ingeniería Electrónica ASIGNATURA: Teoría de Contro						
NRO. PRÁCTICA:	6	TÍTULO DE LA PRÁCTICA: CREACIÓN DEL CONTRO PID PARA LA PLATAFORMA INERCIAL DE 2-GDL.					
OBJETIVOS:							
 Crear un contr táctil resistiva. 	olad	or PID	para la	estabilización del objeto sobre la pantalla			
Utilizar el bloqu	le Pl	ID y sus	atribut	os para ambos ejes de libertad.			
Transformar la para los ejes X	señ y Y.	al obte	nida de	la pantalla táctil resistiva a variable doble			
 Realizar filtrado X y Y. 	o de	la seña	l obteni	da de la pantalla táctil resistiva para los ejes			
Crear un progra	ama	en Sim	ulink pa	ra enviar la señal de control del PID para la			
plataforma iner	cial	de 2-G[DL.				
Diseñar una tra	ayec	toria cire	cular pa	ra que la esfera lo siga indefinidamente.			
	1.	Alimen 110VA	itar la _l .C.	olanta "Plataforma inercial de 2-GDL" con			
	2.	Confirr emerge	mar qu encia n	e la botonera tipo hongo o paro de o esté accionada.			
INSTRUCCIONES	3.	Encen	der la p	lanta presionando el botón de marcha.			
	4.	Verifica	ar que s	e cuente con paquetes y drivers necesarios			
	_	para la	comur	icación de la planta (PRÁCTICA #1).			
	5.	Conec	tar la Itador u	planta "Plataforma inercial de 2-GDL" al			
	6. Visualizar las luces indicadoras de encendido en la r						
	A		ADES P	OR DESARROLLAR			
1. Creación del algo	oritm	no de co	ontrol P	ID para la panta "Plataforma inercial de 2-			
GDL" para la estabilización de la esfera en la referencia que se establece en el							

eje X y Y.

- Agregar los bloques de saturación para limitar el movimiento angular del plato para los ejes X y Y.
- **3.** Agregar los bloques de filtrado de la señal obtenida de la pantalla táctil resistiva de los ejes X y Y por el Arduino DUE.
- 4. Agregar los bloques PID discreto.
- 5. Agrega dos bloques para el diseño de la señal circular de la trayectoria que la esfera sigue indefinidamente
- 6. Agregar los bloques "SCOPE" para la visualización de las señales obtenidas.

<u>Paso 1.</u> Creación del algoritmo de control PID para la panta "Plataforma inercial de 2-GDL" para la estabilización de la esfera en la referencia que se establece en el eje X y Y.

Se procede a abrir un modelo en blanco en Simulink, se agrega dos bloques de escritura estándar para servos y dos bloques "Analog Input" desde la librería de Simulink.



Figura 4. 82: Simulink modelo en blanco y Librería para Arduino.

Se procede añadir los bloques DATA CONVERT a la salida de los bloques de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico.

A continuación de muestra los bloques creados en Simulink para el proceso de la práctica en mención.

a particul density train particul and a br>β(μ) (μ) (μ) (μ) (μ) (μ) (μ) (μ) (μ) (μ)	- 0 ×
Pin 9 Servo X Write	Pin 0 Analog Input X
ARDUINO Pin 10 Servo Y Write	ARDUINO Pin 1 Analog Input Y
Figura 4. 83: Bloque "Data	a conver" con la variable "doublé".
Paso 2. Agregar los bloques de angular del plato para los ejes X y Y	saturación para limitar el movimiento Y.
Se agrega dos bloques de saturaci máximos y mínimos que los servomot	ión que sirve para limitar con los rangos tores se deben mover.
>	
5	aturation
Figura 4. 84: Bloque "Cor	nstant" en la librería de Simulink.
El motor-X trabaja de 90 A 150 y el m valores que se configuran para el bloc	otor-Y trabaja de 20º a 80º estos son los que de saturación.
Block Parameters: Saturation Y 80 - 20	X Block Parameters: Saturation X 150 - 90 X
Saturation	Saturation Imple input signal to the upper and lower saturation values
Main Cianal Attributes	Main Signal Attributes
Upper limit:	Upper limit:
80	150
Lower limit:	Lower limit:
20	→ v [90
	atros del Bloque "Constant"
riguia 4. 65: Parame	
Se muestra el resultado de agregar el Simulink.	l bloque de "Saturation" en el entorno de



	Block Parameters: Kalman Filter X X
	Kalman Filter
	Estimate the state of a dynamic system from a series of incomplete and/or noisy measurements. This block can use the previously estimated state to predict the current state. It can also use the current measurement and the predicted state to estimate the current state value.
	All filters have the same state transition matrix, measurement matrix, initial conditions, and noise covariance, but their state, measurement, enable, and MES supplicable are unique. Within the state, measurement, enable, and MES signals, each column corresponds to a filter.
	Parameters
	Number of filters: 1
	Enable filters: Always
	Initial condition for estimated state:
	Initial condition for estimated error covariance: 10*eye(1)
	State transition matrix: [1]
	Process noise covariance: 0.005*eye(1)
	Measurement matrix source: Specify via dialog •
	Measurement matrix:
	Measurement noise covariance: eye(1)
	Outputs
	□ Output estimated measurement <z_est> □ Output predicted measurement <z_prd></z_prd></z_est>
	Output estimated state <x_est> Output predicted state <x_prd></x_prd></x_est>
	Output MSE of estimated state <mse_est> Output MSE of predicted state <mse_prd></mse_prd></mse_est>
	QK Cancel Help Apply
Figura / 99.	Parametrización para el bloque "Kalman Filter"
i iyula 4. 00.	r arametrizacion para el pioque maiman i inel .
Se muestra a continuac paso paras el diseño de 2-GDL.	ción los bloques agregados al Simulink hasta el presente el algoritmo de control PID para la plataforma inercial de

Sutterl'-Souled	a gadata (anta Junta Junta			- 0 >
5 · 🖙 · 🗃 🗢 🗠 🖇 🚳 ·		• @ • 🚔 •		
Saturation X 150 - 90	ARDUINO Pin 9 Servo X Write	ARDUINO Pin 0 Analog Input X	► double ► Z Data Conver X	Kalman Z_est Filter Z_est Kalman Filter X
Saturation Y 80 - 20	ARDUINO Pin 10 Servo Y Write	ARDUINO Pin 1 Analog Input Y	→double → Z Data Conver Y	Kalman Z_est Filter X Kalman Filter Y

Figura 4. 89: Filtrado de las señales de la pantalla táctil resistiva.

Paso 4. Agregar los bloques PID discreto.

A continuación, se debe añadir dos bloques PID discreto, adicional se crea un bloque constante indicando la referencia por cada eje de liberta.

Posterior se cierra el circuito creando el controlador PID para el eje X y Y en tiempo discreto, todos los bloques antes mencionados se encuentran en la librería de Simulink.



Figura 4. 90: Bloque "Discrete PID Controller".

Se añade los bloques constantes para las referencias y se cierra el lazo con la salida de los bloques "Kalman Filter" del eje X y Y.



Figura 4. 91: Bloque "Constant" para la referencia del controlador PID.

A continuación, se muestra el algoritmo de control PID para la plataforma inercial de 2-GDL y se modifican los parámetros del PID.



Figura 4. 92: Algoritmo de control PID para la plataforma inercial de 2-GDL.

Se procede a modificar los valores del bloque "Discrete PID".

D. Carabarllan		X Block Parameters: Discrete PID Y X
D Controller		PID Controller
is block implements continuous- and discrete-time PID o ndup, external reset, and signal tracking. You can tune t nulink Control Design).	ontrol algorithms and includes advanced features such as anti- he PID gains automatically using the 'Tune' button (requires	This block implements continuous- and discrete-time PID control algorithms and includes advanced features such as anti- windup, external reset, and signal tracking. You can tune the PID gains automatically using the "Tune" button (requires Simulitic Control Osegin).
troller: PID	Form: Parallel	Controller: PID Form: Parallel
me domain:	Discrete-time settings	Time domain: Discrete-time settings
Continuous-time	Integrator method: Forward Euler	Continuous-time Integrator method: Forward Euler
Diseasta kima	Filter method: Forward Euler	Filter method: Forward Euler
Discrete-time	Sample time (-1 for inherited): Ts	Oiscrete-time Sample time (-1 for inherited): Ts
in PID Advanced Data Types State Attribute		Main DID Advanced Date Turns Clade Attributes
ntroller parameters		Main PID Advanced Data Types State Attributes
urce: internal	E Compensator formula	
oportional (P): -0.02	1	
egral (I): -0.0290		Proportional (P): -0.02200
rivative (D): -0.02	$P+I\cdot T$, $\frac{1}{N}+D$	Integral (I): -0.01751
Use filtered derivative	$z = 1$ $1 + N \cdot T_{s} = \frac{1}{z}$	Derivative (D): -0.02215 $P + I \cdot T_s \frac{1}{z-1} + D \frac{N}{1-1}$
er coefficient (N): 100		
	Tune	Filter coefficient (N): 100
		Tune
internal internal		- Initial conditions
arce: internal		Source: internal
egrator: U		Integrator: 0
er: 0		Filter: 0
rnal reset: none		•
nore reset when linearizing		External reset: none
hable zero-crossing detection		Ignore reset when linearizing Section 2.2 Sec
	OK Const Halt	
	2000 2000 2000 2000 2000 2000 200	
ie la esfera sigu	le indefinidamente	diseno de la senal circular de la trayector
Je la esfera sigu ara el seguimient	o de la trayectoria c	ircular de la esfera se crea 2 bloques "Sine
ara el seguimient ara el seguimient	o de la trayectoria c X y Y que se encuel	ircular de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos
ara el seguimient ave" para el eje 2	o de la trayectoria d X y Y que se encuel es con un desfase d	ircular de la senal circular de la trayector htra en la librería de Simulink, se crean do de $\pi/2$ para la creación de la señal circula
ara el seguimient ave" para el eje 2 ñales sinusoidal	o de la trayectoria d X y Y que se encuel es con un desfase o	ircular de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sind ntra en la librería de Simulink, se crean do de π/2 para la creación de la señal circula
ara el seguimient ave" para el eje i ñales sinusoidal le esfera sigue s	o de la trayectoria d X y Y que se encuel es con un desfase o obre la pantalla fácil	ircular de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sind ntra en la librería de Simulink, se crean do de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 1 ñales sinusoidal le esfera sigue s	o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fáci	ircular de la esfera se crea 2 bloques "Sind ntra en la librería de Simulink, se crean do de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje ñales sinusoidal le esfera sigue s	o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fácil	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje males sinusoidal le esfera sigue s	o de la trayectoria o X y Y que se encue es con un desfase o obre la pantalla fácil	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de $\pi/2$ para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 males sinusoidal le esfera sigue s	o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fácil	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de $\pi/2$ para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	o de la trayectoria o X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje i eñales sinusoidal ue esfera sigue s	o de la trayectoria d o de la trayectoria d X y Y que se encuel es con un desfase d obre la pantalla fácil Simulink Library Browser	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser ch Results: sine wwe >> Page 1 of 1 (5 Blocks found)	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje i males sinusoidal le esfera sigue s	o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser Simulink Library Browser Simulink Library Browser Simulink Library Browser	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 íñales sinusoidal le esfera sigue s	o de la trayectoria d X y Y que se encuel es con un desfase d obre la pantalla fácil Simulink Library Browser	ircular de la esfera se crea 2 bloques "Sine ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	os bioques para el c le indefinidamente o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ine wave >> Page 1 of 1 (5 Blocks found) Simulink Commonly Used Blocks Continuos Desenboard Discontinuities	ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje males sinusoidal le esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser Simulink Library Browser Simulink Library Browser Simulink Library Browser Simulink Library Browser Simulink Commonly Used Blocks Continuous Dashboard Discontinuities Discrete	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 íñales sinusoidal le esfera sigue s	o de la trayectoria d X y Y que se encuel es con un desfase d obre la pantalla fácil Simulink Library Browser is ine wave A E E C simulink Simulink Commonly Used Blocks Continuous Discontinuities Discontinuities Discrete Logic and BR Operations	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	os bioques para el c le indefinidamente o de la trayectoria c X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ine wave >> Page 1 of 1 (5 Blocks found) Simulink Commonly Used Blocks Continuos Deshoard Discontinuities Discrete Lockup Tables	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ie la esfera sigu ara el seguimient ave" para el eje ñales sinusoidal le esfera sigue s	o de la trayectoria o X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser Simulink Simuli	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 3 ñales sinusoidal le esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulínk Library Browser Simulínk Library Browser Simulínk Library Browser Simulínk Library Browser Simulínk Commonly Used Blocks Continuous Dashboard Discontinuties Discrete Logic and Bit Operations Logicand Bit Operations Logicand Bit Operations Model Weiffication Model Wide Utilities	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sind ntra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	os bioques para ero le indefinidamente o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ne wave > Page 1 of 1 (5 Blocks found) Simulink Continuous Dashboard Discontinutiles Discrete Logic and BR Operations Lookup Tables Math Operations Model Wiffication Model-Wide Utilities Ports & Subsystems	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine intra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ie la esfera sigu ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	os bioques para ero le indefinidamente o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ine wave > Page 1 of 1 (5 Blocks found) Simulink Commonly Used Blocks Continuous Discrete Logic and BR Operations Lockup Tables Medt Verification Model-Wide Utilities Ports & Subsystems Signal Attributes	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ie la esfera sigu ara el seguimient ave" para el eje ñales sinusoidal le esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ne wave	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	o de la trayectoria co X y Y que se encuer es con un desfase o obre la pantalla fáci simulink Library Browser is ne wave > Page 1 of 1 (5 Blocks found) Simulink Commonly Used Blocks Continuous Dashboard Discontinutises Discrete Logic and Bit Operations Lockup Tables Math Operations Model Verification Model-Wide Utilities Ports & Subsystems Signal Attributes Signal Attributes Signal Attributes Signal Attributes Signal Attributes Signal Attributes	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine that an la librería de Simulink, se crean dos de π/2 para la creación de la señal circular resistiva.
ara el seguimient ave" para el eje 2 ñales sinusoidal le esfera sigue s	So bioques para ero le indefinidamente o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje i ñales sinusoidal le esfera sigue s	Sources Simulink Library Browser Simulink Commonly Used Blocks Continuous Dashboard Discorte Lockup Tables Math Operations Lockup Tables Math Operations Model Werfication Model Werficatio	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine intra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 males sinusoidal le esfera sigue s	Sources Simulink Library Browser Simulink Commonly Used Blocks Continuous Dashboard Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Discontinuous Signal Attributes Signal Autributes Signal Autributes	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine that en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	S bioques para ero le indefinidamente o de la trayectoria co X y Y que se encuer es con un desfase o obre la pantalla fácil Simulink Library Browser	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje males sinusoidal le esfera sigue s	So bioques para ero le indefinidamente o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser Imulink Commonly Used Blocks Continuous Discrete Logic and BR Operations Lockup Tables Math Operations Model Weffication Model Weffication Signal Routing Sinks Sources User-Defined Functions - Additional Math & Discrete Aerospace Blockset Addo System Toolbox Communications System Toolbox	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine intra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	Sources Simulink Library Browser Simulink Library Browser Signal Routing Sinks Sources User-Defined Functions Additional Math & Discrete Arcospace Blockset Audio System Toolbox HOL Su Communications System Toolbox HOL Su	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine that a en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient 'ave" para el eje 2 eñales sinusoidal ue esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ne wave > Page 1 of 1 (5 Blocks found) Simulink Commonly Used Blocks Continuos Discrete Logic and Bit Operations Lockup Tables Med Verification Model-Wide Utilities Discrete Logic and Bit Operations Lockup Tables Med Verification Model-Wide Utilities Signal Attributes Signal Attrib	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient (ave" para el eje 2 eñales sinusoidal ue esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase obre la pantalla fácil Simulink Library Browser Simulink Library Browser Simulink Library Browser Simulink Library Browser New Y Que se encuel es con un desfase obre la pantalla fácil Simulink Library Browser New Y Que se encuel simulink Commonly Used Blocks Continuous Dashboard Discrete Logic and Bit Operations Lockup Tables Meth Operations Model Verification Model Verification Signal Routing Sinks Signal Routing Sinks Communications System Toolbox Communications System Toolbox Communication System Toolbox Communication System Toolbox	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	Sources Lockup Tables Meth Operations Lockup Tables Model Verification Model Verifications Signal Attributes Signal At	Alseno de la senal circular de la trayector ircular de la esfera se crea 2 bloques "Sine that a en la librería de Simulink, se crean dos de π/2 para la creación de la señal circular resistiva.
ara el seguimient ave" para el eje 2 eñales sinusoidal ue esfera sigue s	o de la trayectoria co X y Y que se encuel es con un desfase o obre la pantalla fácil Simulink Library Browser is ne wave > Page 1 of 1 (5 Blocks found) Simulink Commonity Used Blocks Continuos Discrete Logic and Bit Operations Lockup Tables Math Operations Model Wide Utilities Ports & Subsystems Signal Attributes Signal Math & Discrete Arcospace Blockset Audio System Toolbox Communications System Toolbox DSP System Toolbox DSP System Toolbox	ircular de la esfera se crea 2 bloques "Sine htra en la librería de Simulink, se crean dos de π/2 para la creación de la señal circula resistiva.

Figura 4. 94: Bloque "Sine Wave" para la creación de la señal circular

Se parametriza los dos bloques "Sine Wave" para la creación de la señal circular que se envían como referencia a la entrada del bloque PID para el eje X y Y.

📓 Block Parameters: Sine Wave 🛛 🗙	Block Parameters: Sine Wave1 X
Sine Wave	Sine Wave
Output a sine wave:	Output a sine wave:
O(t) = Amp*Sin(Freq*t+Phase) + Bias	O(t) = Amp*Sin(Freq*t+Phase) + Bias
Sine type determines the computational technique used. The parameters in the two types are related through:	Sine type determines the computational technique used. The parameters in the two types are related through:
Samples per period = 2*pi / (Frequency * Sample time)	Samples per period = 2*pi / (Frequency * Sample time)
Number of offset samples = Phase * Samples per period / (2*pi)	Number of offset samples = Phase * Samples per period / (2*pi)
Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.	Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.
Parameters	Parameters
Sine type: Time based	Sine type: Time based
Time (t): Use simulation time	Time (t): Use simulation time
Amplitude:	Amplitude:
100	100
Bias:	Bias:
550	525
Frequency (rad/sec):	Frequency (rad/sec):
6.2831853	6.2831853
Phase (rad):	Phase (rad):
0	1.5707963267948966192313216916398
Sample time:	Sample time:
Ts	Ts
☑ Interpret vector parameters as 1-D	☑ Interpret vector parameters as 1-D
OK Cancel Help Apply	OK Cancel Help Apply

Figura 4. 95: Parametrización de los bloques "Sine Wave"

A continuación, se muestra el algoritmo de control PID para la plataforma inercial de 2-GDL.



Figura 4. 96: Algoritmo de control PID para la plataforma inercial de 2-GDL.

Paso 6. Agregar dos bloques "SCOPE" para la visualización de las señales obtenidas.

Con la ayuda de dos bloques "Scope" se visualizarán las diferentes señales para el análisis de los resultados, el bloque en mención se lo encuentra en la librería de Simulink.







• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control PID para el eje de las X y Y.

Donde la señal de color amarillo es la referencia del sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema.



Figura 4. 101: Bloque "SCOPE" para el eje de las X y Y.

Se comprueba el seguimiento de la esfera de la trayectoria circular sobre el plato, a continuación, se muestra la evidencia del seguimiento de la trayectoria.



Figura 4. 102: Algoritmo de control PID para el seguimiento de trayectoria.



Figura 4. 103: Grafica del seguimiento de la esfera.

• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control PID para el seguimiento de la esfera de la señal generada.

Donde la señal de color amarillo es la señal circular generada que ingresa al sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema.

Se muestra el correcto funcionamiento del controlador siguiendo la señal sinusoidal



Figura 4. 104: Bloque "SCOPE" para el eje de las X y Y.

CONCLUSIONES:

- Se obtiene una comunicación exitosa con la planta didáctica y el software Matlab/Simulink.
- Se configura el bloque PID para el eje de las X y Y, se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul.
- Se realiza con éxitos el seguimiento de la esfera de la señal circular.

4.7 PRÁCTICA # 7

			GUIA DE PRÁCTICA DE LABORATORIO	
CARRERA: Ingen	iería	Electrónica	ASIGNATURA: Teoría de Control II	
NRO. PRÁCTICA:	7	TÍTULO DE FUZZY PAF	FÍTULO DE LA PRÁCTICA: CREACIÓN DEL CONTRO FUZZY PARA EL EJE X.	
OBJETIVOS:				
Creación de ur	n cor	ntrolador FUZ	ZZY para la estabilización del objeto sobre la	
pantalla táctil re	esist	iva.		
Utilizar el bloqu	le Fl	JZZY y sus a	tributos.	
Transformar la	seña	al obtenida de	e la pantalla táctil resistiva a variable doble.	
Realizar filtrade	o de	la señal obtei	nida de la pantalla táctil resistiva.	
Crear un progr	ama	en Simulink	para enviar la señal de control del FUZZY en	
el eje X a la pla	atafo	rma inercial d	le 2-GDL.	
	1.	Alimentar la 110VAC.	planta "Plataforma inercial de 2-GDL" con	
	2.	Confirmar o emergencia	que la botonera tipo hongo o paro de no esté accionada.	
	3.	Encender la	planta presionando el botón de marcha.	
INSTRUCCIONES	4.	Verificar que	e se cuente con paquetes y drivers necesarios	
		para la comu	unicación de la planta (PRÁCTICA #1).	
	5.	Conectar la	planta "Plataforma inercial de 2-GDL" al	
		computador	utilizando el cable USB macho-macho.	
	6.	Visualizar las	s luces indicadoras de encendido en la planta.	
	A		POR DESARROLLAR	
1. Creación del algo	oritm	o de control l	FUZZY para el eje X de la panta "Plataforma	
inercial de 2-GDL	." pa	ra la estabiliz	ación de la esfera en la referencia.	
2. Agregar bloque d	le sa	turación para	limitar el movimiento angular del plato.	

- **3.** Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.
- 4. Agregar el bloque FUZZY Logic Controller.
- 5. Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

<u>Paso 1.</u> Creación del algoritmo de control FUZZY para el eje X de la panta "Plataforma inercial de 2-GDL" para la estabilización del objeto en la referencia.

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos y el bloque "Analog Input" desde la librería de Simulink.



Figura 4. 105: Simulink modelo en blanco y Librería para Arduino.

Se procede añadir el bloque DATA CONVERT a la salida del bloque de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico,

Esos bloques son exportados a Simulink



Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Se agrega el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Figura 4. 107: Bloque "Constant" en la librería de Simulink.

El motor-Y trabaja de 90° a 150° estos son los valores que se configuran para el bloque de saturación.

Pa Block	Parameters: Saturatio	on X 150 - 9	90		×
Saturati	on				^
Limit in	out signal to the upp	per and lo	wer satur	ation values	
Main	Signal Attributes				
Upper lir	nit:				
150					
Lower lin	nit:				
90					Ξ.,
<				>	
0	OK Can	cel	Help	Apply	

Figura 4. 108: Parámetros del Bloque "Constant".

Se muestra el resultado de agregar el bloque de "Saturation" en el ambiente de Simulink para observar el proceso de la práctica que se esta realizando.



<u>Paso 3.</u> Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.

La señal que se obtenida del bloque Analog Input que nos muestran los valores en voltaje del objeto sobre la pantalla táctil resistiva se la debe filtrar para reducir la mayor cantidad de ruido.

Para realizar lo antes mencionado se debe añadir el bloque "Kalman Filter" que se lo encuentra en la librería de Simulink como se muestra a continuación.



Figura 4. 110: Bloque Kalman Filter en la librería de Simulink.

Para la utilización del filtro se debe modificar los parámetros del bloque "Kalman Filter" para un correcto uso del bloque antes mencionado.

Kalman Filter	unamic sustan from a cavica of incomplete and/or acia:	
measurements. This blog	ck can use the previously estimated state to predict the current state	e. It
value.	measurement and the predicted state to estimate the current state	
All filters have the same	state transition matrix, measurement matrix, initial conditions, and	
noise covariance, but the	eir state, measurement, enable, and MSE signals are unique. Within t able, and MSE signals, each column corresponds to a filter.	the
Deservations		
Parameters		
Number of filters: 1		
Enable filters: Alwa	ays	-
Initial condition for estir	nated state: zeros([1])	
Initial condition for estir	nated error covariance: 10*eye(1)	
State transition matrix:	[1]	
Process noise covarianc	e: 0.005*eye(1)	
Measurement matrix so	urce: Specify via dialog	•
Measurement mate	ix: [1]	
Measurement noise cov	ariance: eve(1)	
Fieldsurement holse cov	ununce. eye(1)	
Outputs		
✓ Output estimated me	asurement <z_est> U Output predicted measurement <z_prd></z_prd></z_est>	•
Output estimated sta	te <x_est></x_est>	
Output MSE of estimation	ated state <mse_est> Output MSE of predicted state <mse_prd< td=""><td>d></td></mse_prd<></mse_est>	d>
0	OK Cancel Help Appl	slv
•		, y




Para cumplir con la utilización del control FUZZY se debe crear las variables a utilizar en el bloque de antes mencionado, se escribe en la ventana de comando la palabra "fuzzy" para llamar el toolbox FUZZY.



Figura 4. 116: Ventana de comando para llamar al Toolbox Fuzzy.

Para la creación del algoritmo de control se utiliza el método Mamdani.



Figura 4. 117: Método Mamdani para la creación del algoritmo FUZZY.

Se crean dos variables de entrada y una variable de salida.

Variables de entradas:

Error, representa la retroalimentación de error de la plataforma inercial de 2-GDL **D-Error**, representa la derivada de error.

La variable de Salida:

Control, representa la señal de control hacia los servomotores que entrega la posición angular del bloque fuzzy.

Se define el nombre del bloque FUZZY para el eje X.

FuzzyX, en el entorno de Simulink al utilizar el bloque fuzzy es necesario nombrar al bloque para la definición, para este caso el fuzzy del eje x se llama FuzzyX.



Se muestra la creación de las variables antes mencionados.

Figura 4. 118: FUZZY para el eje X.

Se debe editar las funciones de membresía de control para las variables de entradas y la variable de salida y se utilizan funciones gaussianas para el diseño del algoritmo FUZZY para el eje X.

Variables de entrada:

Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.

D-Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.





Variable de Salida:

Control, tiene 5 funciones de membresía gaussianas que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 107.5 a 127.5, este rango es el movimiento del servomotor en el eje de X.



Figura 4. 120: Funciones de membresía para las variables de entrada.

La señal de CONTROL no se realiza ninguna escala a la salida del fuzzy ya que el rango de operación de la señal gaussiana es el rango de operación del movimiento del servomotor en el eje de la X.

Se procede a agregar las reglas en el Toolbox FUZZY de correspondencia para el correcto funcionamiento del controlador.

承 Rule Editor: F	uzzyX	-		×
File Edit Viev	v Options			
1. If (ERROR is NE	G) and (D-ERROR is NEG) then (CONTROL is PG) (1)			<u>^</u>
2. If (ERROR is NE	G) and (D-ERROR is CERO) then (CONTROL is P) (1)			
3. If (ERROR is NE	G) and (D-ERROR is POS) then (CONTROL is P) (1)			
4. If (ERROR is CE	RO) and (D-ERROR is NEG) then (CONTROL is P) (1)			
5. If (ERROR is CE	RO) and (D-ERROR is CERO) then (CONTROL is Z) (1)			
7 If (ERROR is PO	S) and (D-ERROR is NEG) then (CONTROL IS N) (1)			
8. If (ERROR is PO	S) and (D-ERROR is CERO) then (CONTROL is N) (1)			
9. If (ERROR is PO	S) and (D-ERROR is POS) then (CONTROL is NG) (1)			
				~
If	and	The		
ERROR is	D-ERROR is	0	ONTROL	. is
NEG	NEG	NG		
CERO	CERO	N		
POS	POS	Z		
none	none	P		
		PG		
· ·		Inon	e .	•
not	not		not	
- Connection -	Weight			
Oar				
 and 	1 Delete rule Add rule Change rule		<<	>>
FIG Name Francisk				_
TIS Name: FUZZVX	Help		Clos	20

Figura 4. 121: Reglas de correspondencia para el controlador FUZZY del eje X

Es importante recordar que al diseñar el controlador Fuzzy se define dos variables de entradas las cuales son ERROR y D-ERRO para la medición de la velocidad y el cambio de error se implementan estas dos variables de entradas respectivamente. En la variable de entrada de la derivada del error se coloca un bloque derivativo en tiempo discreto para cumplir con lo antes mencionado.

ice System		
uzzy Inference Syste	em (FIS) as eithe	r a structure or a
e quotes and file ex	tension, e.g., 'tip	per.fis'.)
	ice System uzzy Inference Syste e quotes and file ex	ice System uzzy Inference System (FIS) as eithe e quotes and file extension, e.g., 'tip

Figura 4. 122: Parametrizar el Bloque FUZZY en Simulink.

A continuación, se muestra el algoritmo de control FUZZY para el eje X con los bloques para cumplir con los parámetros creados.



Figura 4. 123: Controlador FUZZY para el eje X.

Paso 5. Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

Con la ayuda del bloque "Scope" se visualizarán las diferentes señales para el análisis de los resultados, el bloque en mención se lo encuentra en la librería de Simulink.



RESULTADO(S) OBTENIDO(S)

• Se comprueba la posición del objeto sobre el plato, se encuentra en la posición deseada de manera visual, la referencia para el eje de las X es de 550.







Figura 4. 128: Posición del objeto en la referencia 550 para el eje X.

• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control FUZZY para el eje X.

Donde la señal de color amarillo es la referencia del sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema

Contra	IROL X		•			- ø ×
Eile Io	ools View Simulation Help					,
						Pos Final
						Señal Control
1000						
1	~					
500	⁷					
	5					
-	and the first of the second second second second	an a				
•	HILLING LA ANAL PRESIDE AND IN	وتقر فأنظرت وسرزيم بالإحدوار فالمطب ورأفض وسأرت مقربي	وتوعد ومطالبته ومستناعين رتقته سرط بنداني سابار	in altis bergeren den der der ditte verstiller.	na a participat, philipping and interaction of the hole of the strategic deg	. Na dan dan an kara glina paginal _m asar dapat daris
•	THE REPORT OF A DESCRIPTION OF A DESCRIP	i - ya konstruinen eta di gonia en de angele a su esta bibigatar.	yldana, de dynig, de para 1996), og er de men og på diftere op som på	ning a latan barg bagta ay sagi kapad da ang aning aning ta yang aning ang aning ang ang ang ang ang ang ang a	nini yadaliyat yatili ni sa shini a shini a shini yatili ya shini sh	n Magalan na katan katang Malangkang Katal _{Si} nan ana ding Katalan.
	Aldilla di Angela angene angela ang angela ang ang ang ang ang ang ang ang ang an	ى يەرىپىيە يەرىپىيە يەرىپەر يە يەرىپەر يەرىپەر	(daan, dadinia, dagan militi), yarda mango dalihan garanja	elo y aldar, barg ing yang di kenandakan pela yalahan, pegnatikan, ing pend	stad proticipy of settionics and disclosed states (d. de accepting	n Mayalana dina ana kaon di katapan jina (, anan atapit dan it
	A LEWERT AND A STREET PROCESSION AS A CONTRACT OF	, պետանցնությունը՝ այն չուն ու նշաց չուրելու է ու որ չուրեն հերձնու	ydd and ynh dynhy Hyna a Mithig wyr allannau gynd ffian gyddyrfr	nin yakata (, ay), nyayya i yaada ay a nyaihin yyyyyi i yaada	ndad y podsing od _v er Albunkson verditi store den skrist grif _v ele same pod g	ville des den och en efter som för i _{de} som de på den ef
-500	A TERRET AND A DESCRIPTION OF A	ng pangapang pang pang pang pang pang pa	y dan kanan daga daga sa dan kanan yang da	ning sakan bergapang depantakan pendalan pengakkan lapan k	and product a subman and transfording decompany	s lig al Sacint an Antag Shing ang ing _{an} an ang palaban
-500	A TEN DIA TA	n g hanning dan ang ang ang ang ang ang ang ang ang a	a da an	na ya ku	n ka mangan ka	n by dan ben a ben y bing og at genne op a ben
-500	NUMULA II A A A A A A A A A A A A A A A A A	neg transpirant pro di po da di da ggi neg po ggi neg populari di sotto.	a dan saya saya saya saya saya saya saya sa			
-500		ne fan en fan it ne fan it ne fan it ne fan gener fan gener fan	a dan serini senin sela sela serin serin serin da serin s			
-500		neg hene y hene en et ge a konst ne gene gene gene gene gene gene de gene gen	a dan parta para anti producence parta para parta p			
-1000		ng transpirang ang pang pang pang pang pang pang pa				
-500						
-500						

Figura 4. 129: Bloque "SCOPE" utilizado en sistema SISO para el eje X.

CONCLUSIONES:

- Se obtiene una comunicación exitosa con el motor-X de plataforma inercial de 2-GDL con el software Matlab/Simulink utilizando el algoritmo de control FUZZY
- Se configura dentro del Toolbox FUZZY las variables de entrada y de salida, las funciones de membresía y las reglas de correspondencia, se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul.

4.8 PRÁCTICA # 8

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR			GL	JIA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica				ASIGNATURA: Teoría de Control II
NRO. PRÁCTICA:	7	TÍTUL FUZZ	. o de i Y par <i>i</i>	LA PRÁCTICA: CREACIÓN DEL CONTROL A EL EJE Y.
OBJETIVOS:	1			
Creación de ur pantalla táctil re	n cor esist	ntrolado iva.	r FUZZ	Y para la estabilización del objeto sobre la
Utilizar el bloqu	ie Fl	JZZY y	sus atri	butos.
Transformar la	seña	al obten	ida de l	a pantalla táctil resistiva a variable doble.
Realizar filtrado	o de	la seña	l obteni	da de la pantalla táctil resistiva.
Crear un progra	ama	en Sim	ulink pa	ara enviar la señal de control del FUZZY en
el eje Y a la pla	atafo	rma ine	rcial de	2-GDL.
	1.	Alimen 110VA	tar la p C.	olanta "Plataforma inercial de 2-GDL" con
	2.	Confirm emerge	nar qu encia no	e la botonera tipo hongo o paro de o esté accionada.
INSTRUCCIONES	3.	Encend	der la p	lanta presionando el botón de marcha.
	4.	Verifica	ar que s	e cuente con paquetes y drivers necesarios
		para la	comun	icación de la planta (PRÁCTICA #1).
	5.	Conect	tar la _l	planta "Plataforma inercial de 2-GDL" al
		compu	tador u	tilizando el cable USB macho-macho.
	6.	Visuali	zar las l	uces indicadoras de encendido en la planta.
	AC	CTIVIDA	ADES P	OR DESARROLLAR
1. Creación del algo	oritm	o de co	ntrol Fl	JZZY para el eje Y de la panta "Plataforma
inercial de 2-GDL	." ра	ra la est	tabilizad	ción de la esfera en la referencia.

- 2. Agregar bloque de saturación para limitar el movimiento angular del plato.
- **3.** Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.
- 4. Agregar el bloque FUZZY Logic Controller.
- 5. Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

<u>Paso 1.</u> Creación del algoritmo de control FUZZY para el eje Y de la panta "Plataforma inercial de 2-GDL" para la estabilización del objeto en la referencia.

Se procede a abrir un modelo en blanco en Simulink, se agrega el bloque de escritura estándar para servos y el bloque "Analog Input" desde la librería de Simulink.



Figura 4. 130: Simulink modelo en blanco y Librería para Arduino.

Continuous Servo Write

ARDUINO

Digital Input

ARDUINO

Utilities

Wifi Shield

Se procede añadir el bloque DATA CONVERT a la salida del bloque de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico.



Figura 4. 131: Bloque "Data conver Y" con la variable "doublé".

Paso 2. Agregar bloque de saturación para limitar el movimiento angular del plato.

Se agrega el bloque de saturación que sirve para limitar con los rangos máximos y mínimos que el servomotor se debe mover.



Saturation

Figura 4. 132: Bloque "Constant" en la librería de Simulink.

El motor-Y trabaja de 20º a 80º estos son los valores que se configuran para el bloque de saturación.

Saturati	on	on Y 80 - 20	
Limit in	out signal to the upp	per and lower sat	uration values
Main	Signal Attributes		
Upper lir	nit:		
80			
Lower lir	nit:		
20			
<			3

Figura 4. 133: Parámetros del Bloque "Constant".

Se muestra el resultado de agregar el bloque de "Saturation"



Paso 3. Agregar el bloque de filtrado de la señal obtenida de la pantalla táctil resistiva por el Arduino DUE.

La señal que se obtenida del bloque Analog Input que nos muestran los valores en voltaje del objeto sobre la pantalla táctil resistiva se la debe filtrar para reducir la mayor cantidad de ruido.

Para realizar lo antes mencionado se debe añadir el bloque "Kalman Filter" que se lo encuentra en la librería de Simulink como se muestra a continuación.



Figura 4. 135: Bloque Kalman Filter en la librería de Simulink.

Para la utilización del filtro se debe modificar los parámetros del bloque "Kalman Filter" para un correcto uso del bloque antes mencionado.

Block Parameters: I	Kalman Filter X				×
Kalman Filter					
Estimate the state o measurements. This can also use the cur value. All filters have the sa	f a dynamic sy block can use rent measuren ame state tran	stem from a s the previously nent and the p sition matrix, i	eries of incomplete and v estimated state to pre- redicted state to estim- measurement matrix, ir	d/or noisy edict the current stat ate the current state nitial conditions, and	te. It
state, measurement	, enable, and I	MSE signals, ei	ach column correspond	is to a filter.	rule
Parameters					
Number of filters:	1				:
Enable filters:	Always				•
Initial condition for	estimated stat	ie:	zeros([1])		:
Initial condition for	estimated erro	or covariance:	10*eye(1)		:
State transition mat	trix:	[1]			:
Process noise covar	riance:	0.005*eye(1)		:
Measurement matri	ix source:	Specify via	dialog		-
Measurement	matrix:	[1]			:
Measurement noise	covariance:	eye(1)			:
Outputs					
Output estimated	d measuremer	it <z_est></z_est>	Output predicted m	neasurement <z_prd< td=""><td> ></td></z_prd<>	>
Output estimated	d state <x_est< td=""><td>></td><td>Output predicted st</td><td>ate <x_prd></x_prd></td><td></td></x_est<>	>	Output predicted st	ate <x_prd></x_prd>	
Output MSE of e	stimated state	<mse_est></mse_est>	Output MSE of pred	dicted state <mse_p< td=""><td>rd></td></mse_p<>	rd>
0		9	K <u>C</u> ancel	<u>H</u> elp <u>A</u> p	ply

Se muestra a continuación los bloques agregados al Simulink hasta el presente paso paras el diseño del algoritmo de control FUZZY para el eje y.



Figura 4. 137: Filtrado de la señal de la pantalla táctil resistiva.

Paso 4. Agregar el bloque FUZZY Logic Controller.

A continuación, se debe añadir el bloque FUZZY, adicional se crea un bloque constante indicando la referencia. Posterior se cierra el circuito creando el controlador FUZZY para el eje Y en tiempo discreto, todos los bloques antes mencionados se encuentran en la librería de Simulink.



Se añade el bloque constante para la referencia y se cierra el lazo con la salida del bloque "Kalman Filter".



167

palabra "fuzzy" para llamar el toolbox FUZZY.





Figura 4. 142: Método Mamdani para la creación del algoritmo FUZZY.

Se crean dos variables de entrada y una variable de salida.

Variables de entradas:

Error, representa la retroalimentación de error de la plataforma inercial de 2-GDL **D-Error**, representa la derivada de error.

La variable de Salida:

Control, representa la señal de control hacia los servomotores que entrega la posición angular del bloque fuzzy.

Se define el nombre del bloque FUZZY para el eje X.

FuzzyY, en el entorno de Simulink al utilizar el bloque fuzzy es necesario nombrar al bloque para la definición, para este caso el FUZZY del eje y se llama FuzzyY.

Se muestra la creación de las variables antes mencionados.



Figura 4. 143: FUZZY para el eje Y.

Se debe editar las funciones de membresía de control para las variables de entradas y la variable de salida y se utilizan funciones gaussianas para el diseño del algoritmo FUZZY para el eje Y.

Variables de entrada:

Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.

D-Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.





Variable de Salida:

Control, tiene 5 funciones de membresía gaussiana que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 12a 92, este rango es el movimiento del servomotor en el eje de Y.



Figura 4. 145: Funciones de membresía para las variables de entrada.

La señal de CONTROL no se realiza ninguna escala a la salida del fuzzy ya que el rango de operación de la señal gaussiana es el rango de operación del movimiento del servomotor en el eje de la Y

Se procede a agregar las reglas en el Toolbox FUZZY de correspondencia para el correcto funcionamiento del controlador.





Es importante recordar que al diseñar el controlador FUZZY se define dos variables de entradas las cuales son ERROR y D-ERRO para la medición de la velocidad y el cambio de error se implementan estas dos variables de entradas respectivamente. En la variable de entrada de la derivada del error se coloca un bloque derivativo en tiempo discreto para cumplir con lo antes mencionado.

🎦 Block Param	neters: Fuzzy Co	ntroller Y		
- Fuzzy Inferen	ce System			
Specify the Fu file.	zzy Inference	System (FIS) a	as either a struc	ture or a
Parameters				
FIS name: (For a file, use	e quotes and fi	ile extension, e	e.g., 'tipper.fis'.))
'FuzzyY'				
	OK	Cancol	Holp	Apply

Figura 4. 147: Parametrizar el Bloque FUZZY en Simulink.

A continuación, se muestra el algoritmo de control FUZZY para el eje Y con los bloques para cumplir con los parámetros creados.



Figura 4. 148: Controlador FUZZY para el eje Y.

<u>Paso 5.</u> Agregar el bloque "SCOPE" para la visualización de las señales obtenidas.

Con la ayuda del bloque "Scope" se visualizarán las diferentes señales para el análisis de los resultados, el bloque en mención se lo encuentra en la librería de Simulink.

Simulink Library Browser
< 🔶 Scope 🗸 🗸 🗸
Search Results: Scope <<>> Page 1 of 1 (21 Blocks found)
 Simulik Commonly Used Blocks Continuous Dashboard Discontinuities Discrete Loqic and Bit Operations Lookung Tables Math Operations Model Verification Signal Routing Sinkis Sources User-Defined Functions Addisional Math & Bicsrete Addisocket Audio System Toolbox Communication System Toolbox DSF System Toolbox HDL Support Embedded Coder Furzu India Toolbox

Se procede a configurar el bloque para visualizar las señales del control aplicado al eje Y.



Figura 4. 150: Algoritmo de control FUZZY para el eje Y.

Para proceder a poner en marcha nuestro algoritmo, se debe configurar el parámetro "sample time" de todos los bloques de simulink con la variable del tiempo de muestreo (Ts) creado en Matlab. El tiempo de muestreo Ts es de 0.01 y se lo debe de escribir en la ventana de comando de Matlab como se muestra a continuación.

Command Window	
New to MATLAB? See resources for Getting Start	ed.
>> Ts=0.01	
Ts =	
0.0100	
fx, >>>	





CONTROL Y File Tools View Simulation Help					- a ×
◎ · ○					T
1000					☐ Ref ☐ Pos Final ☐ Error ☐ Sofial Costrol ☐ Ditror
	lin whether the transmitted of the second	terrere die telefisie beingdele stelle sone è	na haran dagan di kasalah kana pina dadi ali ada ang ka	i yang manala kanangan kating disebut di dalam mengenti kalan da	
and the second	alon or for a for the form	e benned prijsken ne general de neder an de	nin ya postanji na poslava za pod plazava za plazava za poslava	engelene och an ann einen et blire at skille at vander stardt. St	nen fill som som som de side for at statistic hondra som
-000					
-100					
-1500					
50	100) 15	0 28	0 21	0

Figura 4. 154: Bloque "SCOPE" utilizado en sistema SISO para el eje Y.

CONCLUSIONES:

- Se obtiene una comunicación exitosa con el motor-Y de plataforma inercial de 2-GDL con el software Matlab/Simulink utilizando el algoritmo de control FUZZY
- Se configura dentro del Toolbox FUZZY las variables de entrada y de salida, las funciones de membresía y las reglas de correspondencia, se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul.

4.9 PRÁCTICA # 9

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR			Gl	JIA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica				ASIGNATURA: Teoría de Control III
NRO. PRÁCTICA:	9	TÍTUI FUZZ	.o de i Y par/	L A PRÁCTICA: CREACIÓN DEL CONTROL A LA PLATAFORMA INERCIAL DE 2-GDL.
OBJETIVOS:				
Crear un contro téctil registive	olado	or FUZZ	Y para	la estabilización del objeto sobre la pantalla
 Utilizar el blogu 	ıe Fl	JZZY v	sus atri	ibutos para ambos ejes de libertad.
 Transformar la para los ejes X 	señ y Y.	al obte	nida de	la pantalla táctil resistiva a variable doble
Realizar filtrade	o de	la seña	l obteni	da de la pantalla táctil resistiva para los ejes
ХуҮ.				
Crear un progra	ama	en Sim	ulink pa	ra enviar la señal de control del FUZZY para
la plataforma ir	nerci	al de 2-	GDL.	
Diseñar una tra	ayec	toria elí	ptica pa	ra que la esfera lo siga indefinidamente.
	1.	Alimer 110VA	itar la j .C.	olanta "Plataforma inercial de 2-GDL" con
	2.	Confirr emerg	mar qu encia ne	le la botonera tipo hongo o paro de o esté accionada.
INSTRUCCIONES	3.	Encen	der la p	lanta presionando el botón de marcha.
	4.	Verific	ar que s	e cuente con paquetes y drivers necesarios
		para la	a comur	icación de la planta (PRÁCTICA #1).
	5.	Conec	tar la	planta "Plataforma inercial de 2-GDL" al
		compu	itador u	tilizando el cable USB macho-macho.
	6.	Visuali	zar las	luces indicadoras de encendido en la planta.
	A		ADES P	OR DESARROLLAR
1. Creación del algoritmo de control FUZZY para la panta "Plataforma inercial de 2-GDL" para la establización de la esfera en la referencia que se establece en				

el eje X y Y.

- **2.** Agregar los bloques de saturación para limitar el movimiento angular del plato para los ejes X y Y.
- **3.** Agregar los bloques de filtrado de la señal obtenida de la pantalla táctil resistiva de los ejes X y Y por el Arduino DUE.
- **4.** Agregar el bloque FUZZY Logic Controller.
- 5. Agrega dos bloques para el diseño de la señal elíptica de la trayectoria que la esfera sigue indefinidamente
- 6. Agregar los bloques "SCOPE" para la visualización de las señales obtenidas.

Paso 1. Creación del algoritmo de control FUZZY para la panta "Plataforma inercial de 2-GDL" para la estabilización de la esfera en la referencia que se establece en el eje X y Y.

Se procede a abrir un modelo en blanco en Simulink, se agrega dos bloques de escritura estándar para servos y dos bloques "Analog Input" desde la librería de Simulink.



Figura 4. 155: Simulink modelo en blanco y Librería para Arduino.

Se procede añadir los bloques DATA CONVERT a la salida de los bloques de Analog Input para la transformación de la señal analógica de single a doble como se muestra en el siguiente gráfico.

A continuación se observa en el ambiente Simulink el proceso de la practica con los bloques antes mencionados.

\mathbb{R}_{2} of \mathbb{R}_{2} $0 \in \mathbb{Q}$ \mathbb{R}_{2} \mathbb{R}_{2} and \mathbb{R}_{2} $$	- • • ×
ARDUINO Pin 9 Servo X Write	ARDUINO Pin 0 Analog Input X
ARDUINO Pin 10 Servo Y Write	ARDUINO Pin 1 Analog Input Y
Figura 4. 156: Bloque "Data o	conver" con la variable "doublé.
Paso 2. Agregar los bloques de satur angular del plato para los ejes X y Y. Se agrega dos bloques de saturaciór máximos y mínimos que los servomotor	ación para limitar el movimiento n que sirve para limitar con los rangos es se deben mover.
×	$\not\leftarrow$
Figura 4. 157 : Bloque "Cons El motor-X trabaja de 90 A 150 y el moto valores que se configuran para el bloque	tant" en la librería de Simulink. or-Y trabaja de 20º a 80º estos son los e de saturación.
Block Parameters: Saturation Y 80 - 20	Block Parameters: Saturation X 150 - 90 X
Saturation Limit input signal to the upper and lower saturation values. Main Signal Attributes Upper limit: 80 Lower limit: 20 CK Cancel Help Apply	Saturation Limit input signal to the upper and lower saturation values. Main Signal Attributes Upper limit: 150 Lower limit: 90 OK Cancel Help Apply
Figura 4. 158: Parámetr	os del Bloque "Constant".
Se muestra el resultado de agregar el b Simulink para evidenciar el proceso de l	loque de "Saturation" en el ambiente de a práctica a realizar.



nuestra a contir o paras el diseñ 2-GDL. bloques que se 2-GDL a cualqui actica.	uación los o del algor utilizan so er movimie	bloques itmo de o on para la ento brus	agregade control FL as protecc co y logra	os al Sim JZZY para ciones de ar mitigar	ulink ha a la pla la plata los daŕ	asta el pre taforma in aforma ine ños a la p	eser nerc ercia lanta
nuestra a contir o paras el diseñ 2-GDL. bloques que se 2-GDL a cualqui	uación los o del algor utilizan so er movimie	bloques itmo de o on para la ento brus	agregade control FL as protecc	os al Sim JZZY para ciones de ar mitigar	ulink ha a la pla la plata los daŕ	asta el pre taforma il aforma ine ños a la p	eser nerc ercia lanta
muestra a contir o paras el diseñ 2-GDL.	uación los o del algor	itmo de o	agregade	os al Sim JZZY para	ulink ha a la pla	asta el pre taforma il	eser nerci
Figura 4.	161: Parar	netrizaci	ón para e	l bloque "	Kalmaı	n Filter".	
		acca state <mbc_e< th=""><th></th><th>cel <u>H</u>elp</th><th><u>Apply</u></th><th></th><th></th></mbc_e<>		cel <u>H</u> elp	<u>Apply</u>		
	Output estimated me Output estimated sta Output MSE of estima	easurement <z_est ite <x_est> ated state <msf @<="" td=""><td> Output predi Output predi St> Output MSF </td><td>cted measurement cted state <x_prd> of predicted state <</x_prd></td></msf></x_est></z_est 	 Output predi Output predi St> Output MSF 	cted measurement cted state <x_prd> of predicted state <</x_prd>	<z_prd></z_prd>		
	4easurement noise cov	ariance: eye(1)			:		
	Measurement matrix so	rix: [1]	y via dialOg		:		
F	Process noise covarianc	e: 0.005*e	eye(1)		:		
1	nitial condition for estir State transition matrix:	mated error covaria	ince: 10*eye(1)				
I	nitial condition for estir	nated state:	zeros([1]))	:		
E	Enable filters: Alwa	ays			•		
1	Number of filters: 1				E		
- F	arameters	able, and MSE signa	ais, each column com	esponds to a filter.			
A ni st	Il filters have the same bise covariance, but the ate, measurement, ena	state transition ma eir state, measuren able, and MSE signa	ntrix, measurement m ment, enable, and MSI als. each column corre	atrix, initial condition E signals are unique. esponds to a filter.	ns, and Within the		
m ca Vi	easurements. This bloo an also use the current alue.	ck can use the prev measurement and	riously estimated state the predicted state to	e to predict the curre o estimate the curre	ent state. It nt state		
	alman Filter stimate the state of a d	lynamic system fro	m a series of incompl	ete and/or noisy			

Paso 4. Agregar los bloques FUZZY Logic Controller

A continuación, se debe añadir dos bloques FUZZY Logic Controller, adicional se crea un bloque constante indicando la referencia por cada eje de liberta. Posterior se cierra el circuito creando el controlador FUZZY para el eje X y Y en tiempo discreto, todos los bloques antes mencionados se encuentran en la librería de Simulink.





Se añade los bloques constantes para las referencias y se cierra el lazo con la salida de los bloques "Kalman Filter" del eje X y Y.



A continuación, se muestra el algoritmo de control FUZZY para la plataforma inercial de 2-GDL.



Figura 4. 165: Algoritmo de control FUZZY.

Para cumplir con la utilización del control FUZZY se debe crear las variables a utilizar en los bloques antes mencionados, se escribe en la ventana de comando la palabra "fuzzy" para llamar el toolbox FUZZY.

to MATLAB? See resource	s for <u>Getting Started</u> .		
fuzzy			
📕 Fuzzy Logic Desig	ner: Untitled		- 🗆 ×
File Edit View			
		Untitled (mamdani)	
input1			output1
input1 FIS Name:	Untitled	FIS Type:	output1
FIS Name:	Untitled	FIS Type:	output1
FIS Name: And method Or method	Untitled min max	FIS Type:	e input1
Input1 FIS Name: And method Or method Implication	Untitled min max min	FIS Type:	outputt mamdani e input input m ut
FIS Name: FIS Name: And method Or method Implication Aggregation	Untitled min max min max	FIS Type: Current Variable Vame Type Range	outputt mamdani e input1 input [0 1]

Figura 4. 166: Ventana de comando para llamar al Toolbox Fuzzy.

Se debe realizar el proceso de las prácticas al detalle para cumplir con los objetivos y el correcto funcionamiento de la plataforma inercial con el controlador FUZZY.

Para la creación de los algoritmos de control se utiliza el método Mamdani.



Figura 4. 167: Método Mamdani para la creación del algoritmo FUZZY.

Se crean dos variables de entrada y una variable de salida para cada controlador FUZZY.

Variables de entradas:

Error, representa la retroalimentación de error de la plataforma inercial de 2-GDL **D-Error**, representa la derivada de error.

La variable de Salida:

Control, representa la señal de control hacia los servomotores que entrega la posición angular del bloque fuzzy.

Se define el nombre del bloque FUZZY para el eje X y Y.

FuzzyX, en el entorno de Simulink al utilizar el bloque fuzzy es necesario nombrar al bloque para la definición, para este caso el fuzzy del eje x se llama FuzzyX.

FuzzyY, en el entorno de Simulink al utilizar el bloque fuzzy es necesario nombrar al bloque para la definición, para este caso el FUZZY del eje x se llama FuzzyY.

Se muestra la creación de las variables antes mencionados.

Fuzzy Logic Designer:	FuzzyX			-		File Edit View				
ERROR		Fuz: (mam	zyX danī)			ERROF		Fuzz	tyY dani)	
D-ERROR	J			CONTRO	OL	FIS Name:	FuzzyY		FIS Type:	mamdani
FIS Name: Fu	zzyX		FIS Type:	mamdani	i	And method	min	~	Current Variable	
And method	min	×	Current Variable			Or method	max	~	Name	ERROR
Or method			Name	FRROR		Implication	min	~	Type	input
	max	~	Туре	input		Aggregation	max	\sim	Range	[-2000 2000]
Implication	min	~	Range	[-1000 1	0001	Defuzzification			Uala	01
Aggregation	max	~					Centroid	*		Close
Defuzzification	centroid	~	Help		Close	System "FuzzyY": 2	inputs, 1 output, and 9 rules			

Figura 4. 168: FUZZY para el eje X y Y.

Se debe editar las funciones de membresía de control para las variables de entradas y la variable de salida y se utilizan funciones gaussianas para el diseño

del algoritmo FUZZY para el eje X y Y.

Variables de entrada:

FuzzyX

Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.

D-Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.





FuzzyY

Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.

D-Error, tiene 3 funciones de membresía gaussianas, dicha funciones tienen los siguientes nombres, NEG (Negativo), CERO (Cero) y POS (Positivo), que operan en el rango de -1000 a 1000.



Variable de Salida para FuzzyX y FuzzyY:

FuzzyX

Control, tiene 5 funciones de membresía gaussianas que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 107.5 a 127.5, este rango es el movimiento del servomotor en el eje de X.





FuzzyY

Control, tiene 5 funciones de membresía gaussiana que se denominan NG (Negativo grande), N (Negativo), Z (Cero), P (Positivo) y PG (Positivo grande), que funcionan en el rango de 12 a 92, este rango es el movimiento del servomotor en el eje de Y.



Figura 4. 172: Funciones de membresía para FuzzyY.

La señal de CONTROL no se realiza ninguna escala a la salida del FUZZY ya que el rango de operación de la señal gaussiana es el rango de operación del movimiento del servomotor en el eje X y Y

Se procede a agregar las reglas en el Toolbox FUZZY de correspondencia para el correcto funcionamiento del controlador.

1. If (ERROR is NE	G) and (D-ERROR	is NEG) then ((CONTROL is PG)	(1)			
2. If (ERROR is NE	G) and (D-ERROR	is CERO) then	(CONTROL is P)	(1)			
3. If (ERROR is NE	G) and (D-ERROR	is POS) then (CONTROL is P) (1)			
4. If (ERROR is CE	RO) and (D-ERROP	R is NEG) then	(CONTROL is P)	(1)			
5. If (ERROR is CE	RO) and (D-ERROF	R is CERO) the	n (CONTROL is 2	.) (1)			
6. If (ERROR is CE	RO) and (D-ERROP	R is POS) then	(CONTROL is N)	(1)			
7. If (ERROR is PO	S) and (D-ERROR	is NEG) then (CONTROL IS N) (1)			
0. IT (ERROR IS PO	S) and (D-ERROR	is CERU) then	CONTROL IS NO	(1)			
5. II (ERROR IS PO	5) and (D-ERROR	is FUS) then (CONTROL IS NO.	0			
If	and				Then		
					1 II CII		
ERROR is	D-ERROR is	•			CO	NTROL	. is
ERROR is	D-ERROR is	A			CO	NTROL	. is
ERROR is	D-ERROR is NEG CERO	^			CO NG N	NTROL	. is
ERROR is NEG CERO POS	D-ERROR is NEG CERO POS	^			CO NG N Z	NTROL	. is
ERROR is NEG CERO POS none	D-ERROR is NEG CERO POS none	^			CO NG N Z P	NTROL	. is
ERROR is NEG CERO POS none	D-ERROR is NEG CERO POS none				CO NG N Z P PG	NTROL	. is
ERROR is NEG CERO POS none V	D-ERROR is NEG CERO POS none	×			CO NG N Z P PG none	NTROL	. is
ERROR is NEG ^ CERO POS none ~	D-ERROR is NEG CERO POS none	×			CO NG N Z P PG none	NTROL	is
ERROR is NEG CERO POS none v not	D-ERROR is NEG CERO POS none	~			CO NG N Z PG none	ot	is
ERROR is NEG CERO POS none not Connection	D-ERROR is NEG CERO POS none not Weight:	×			CO NG N Z P PG none	ot	is
ERROR is NEG CERO POS none not Connection or or	D-ERROR is NEG CERO POS none not Weight:	×			CO NG N Z P PG none	ntrol	is ,

Figura 4. 173: Reglas de correspondencia para el controlador FUZZY.

Es importante recordar que al diseñar los controladores Fuzzy se define dos variables de entradas las cuales son ERROR y D-ERRO respectivamente para la medición de la velocidad y el cambio de error se implementan estas dos variables de entradas respectivamente. En la variable de entrada de la derivada del error se coloca un bloque derivativo en tiempo discreto para cumplir con lo antes mencionado.

Block Parameters: Fuzzy Controller X X	Block Parameters: Fuzzy Controller Y X
Fuzzy Inference System	Fuzzy Inference System
Specify the Fuzzy Inference System (FIS) as either a structure or a file.	Specify the Fuzzy Inference System (FIS) as either a structure or a file.
Parameters	Parameters
FIS name: (For a file, use quotes and file extension, e.g., 'tipper.fis'.)	FIS name: (For a file, use quotes and file extension, e.g., 'tipper.fis'.)
'FuzzyX'	'FuzzyY'
<u>O</u> K <u>C</u> ancel <u>H</u> elp <u>Apply</u>	<u>O</u> K <u>Cancel Help</u> Apply

Figura 4. 174: Parametrizar los Bloque FUZZY en Simulink.

A continuación, se muestra el algoritmo de control FUZZY para el eje X y Y con los bloques para cumplir con los parámetros creados.



Figura 4. 176: Bloque "Sine Wave" para la creación de la señal circular

Se parametriza los dos bloques "Sine Wave" como se observa en la siguiente figura para la creación de la señal circular que se envían como referencia a la entrada del bloque FUZZY para el eje X y Y.

🛅 Block Parameters: Sine Wave X 🛛 🛛 🗡	- 🛅 Block Parameters: Sine Wave Y 🛛 🗙 -
Sine Wave	Sine Wave
Output a sine wave:	Output a sine wave:
O(t) = Amp*Sin(Freq*t+Phase) + Bias	O(t) = Amp*Sin(Freq*t+Phase) + Bias
Sine type determines the computational technique used. The parameters in the two types are related through:	 Sine type determines the computational technique used. The parameters in the two types are related through:
Samples per period = 2*pi / (Frequency * Sample time)	Samples per period = 2*pi / (Frequency * Sample time)
Number of offset samples = Phase * Samples per period / (2*pi)	. Number of offset samples = Phase * Samples per period / (2*pi)
Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.	Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.
Parameters	Parameters
Sine type: Time based -	Sine type: Time based
Time (t): Use simulation time	Time (t): Use simulation time
Amplitude:	Amplitude:
80	80
Bias:	Bias:
550	550
Frequency (rad/sec):	Frequency (rad/sec):
1.6	1.6
Phase (rad):	Phase (rad):
0	0.3
Sample time:	Sample time:
Ts	Ts
☑ Interpret vector parameters as 1-D	✓ Interpret vector parameters as 1-D
OK Cancel Help Apply	OK Cancel Help Apply

Figura 4. 177: Parametrización de los bloques "Sine Wave".

A continuación, se muestra el algoritmo de control FUZZY para la plataforma inercial de 2-GDL.



Figura 4. 178: Algoritmo de control FUZZY con el seguimiento de trayectoria

Paso 6. Agregar dos bloques "SCOPE" para la visualización de las señales obtenidas.

Con la ayuda de dos bloques "Scope" se visualizarán las diferentes señales para el análisis de los resultados, el bloque en mención se lo encuentra en la librería de Simulink,

Simulink Library Browser					
Social Artificial System Toblox Social Artificial System Toblox Communications System	📲 Simulink Library Browser		- 0	×	
Sector Require: Scope <->> Page 1 of 1 (21 Blocks found)	< 🔶 Scope 🗸 🕹 📩	• 🔚 • 🛥 🔞			
Simulark Commonly Used Blocks Continuus Continuus Discontinuities Discontinuities Discontinuities Disconte Lagic and Bit Operations Logic and Bit Operations Model Wetrification Model Wetrification Model Wetrification Model Wetrification Model Wetrification Model Wetrification Model Wetrification Model Wetrification Signal Arbitrutes Signal Arbitrutes Signal Arbitrutes Scope Visability Scope <	Search Results: Scope <<>> Page 1 of 1 (21 Blocks found)				
	Communications System Toolbox Communications System Toolbox Control System Control Control Sy	Simulink - 8 (simulink/Common) Scope (A) From (A) Goto Tag Viability Viability Scope Aerospace Blockset - 1 w (red(s) wmss(rad(s)) Three-axis Gyroscope DSP System Toolbox - 2	y Used Biodey/Scope)	<	

Figura 4. 179: Bloques "Scope" para el eje X y Y.

Se procede a configurar el bloque para visualizar las señales del control aplicado al eje X y Y.



Figura 4. 180: Algoritmo de control FUZZY para el eje X y Y.

Para proceder a poner en marcha nuestro algoritmo, se debe configurar el parámetro "sample time" de todos los bloques de simulink con la variable del tiempo de muestreo (Ts) creado en Matlab. El tiempo de muestreo Ts es de 0.01 y se lo debe de escribir en la ventana de comando de Matlab como se muestra a continuación.

Com	nand Window
New	to MATLAB? See resources for Getting Started.
>:	> Ts=0.01
T	3 =
	0.0100
fx >:	





Figura 4. 183: Posición del objeto en la referencia 550 para el eje de las X y Y.

 Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control FUZZY para el eje de las X y Y.
 Donde la señal de color amarillo es la referencia del sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema.




• Se visualiza que con ayuda del bloque "SCOPE" la señales para la operación del control FUZZY para el seguimiento de la esfera de la señal generada.

Donde la señal de color amarillo es la señal circular generada que ingresa al sistema, la señal de color azul es la salida del sistema, la señal de color marrón es el error del sistema



Figura 4. 187: Bloque "SCOPE" para el eje de las X y Y.

CONCLUSIONES:

- Se obtiene una comunicación exitosa con la planta didáctica y el software Matlab/Simulink.
- Se configura el bloque FUZZY para el eje de las X y Y, se obtiene que el objeto sobre el plato se ubique en la referencia del sistema de manera correcta.
- El bloque de saturación protege la planta para que no usen valores de ángulos que puedan ocasionar daños en la estructura mecánica.
- Se configura el bloque ""Kalman Filter" de manera correcta para que la señal obtenida de la pantalla táctil resistiva este sin ruido la cual observamos en la señal de color Azul.
- Se realiza con éxitos el seguimiento de la esfera de la trayectoria elíptica.

RESULTADOS

Al término del proyecto de titulación se evidencia un cambio en la estructura de la plataforma inercial de 2-GDL tanto en sus partes mecánicas y electrónicas, a continuación, se exponen los resultados obtenidos.



Figura 5. 1: Plataforma Inercial de 2-GDL.



Figura 5. 2: Tarjeta PCB instalada.

Para obtener un mejor movimiento de la plataforma inercial de 2-GDL se reubican los motores y se incorporan brazos articuladores para transmitir el movimiento al plato donde reposa la pantalla táctil resistiva. Los motores que se utilizan son servomotores, que con la ayuda de la tarjeta PCB y la pantalla táctil resistiva se cumple con la propuesta de realizar el proyecto de titulación con elementos de bajo costos.

Se instala un panel de control para que los estudiantes que realicen pruebas con la plataforma inercial de 2-GDL tengan un control físico. A continuación, mostramos el resultado obtenido del panel de control de la plataforma inercial de 2-GDL.



Figura 5. 3: Panel de control.

El panel de control cuenta con botones pulsadores para la marcha y paro, un botón tipo hongo como paro de emergencia y luces piloto como indicadores del encendido y la puesta en marcha. En el panel de control también se encuentra el puerto USB que se utiliza como medio de comunicación con la tarjeta PCB.

Con el fin de mostrar los resultados obtenidos al aplicar los controladores diseñados para la plataforma inercial de 2-GDL, se exporta al Workspace la variable de error de los controladores PID y FUZZY para obtener los porcentajes de error de los controladores diseñados.

Controlador PID

A continuación, con ayuda de Matlab y el valor exportado de Simulink se muestra el análisis de error del controlador PID.

Se exporta 100 registros de la variable de salida y se lo compara con la referencia y se obtiene los siguientes resultados.

Para el eje X se tiene un error del 6.58% y se adjunta la imagen del control aplicado donde la referencia es del color amarillo, la señal de salida es de color azul y la señal de color marrón es el error.



Figura 5. 4: Análisis de la respuesta en tiempo real del PID en el eje X.

Para el eje Y se tiene un error del 3.9% y se adjunta la imagen del control aplicado donde la referencia es del color amarillo, la señal de salida es de color azul y la señal de color marrón es el error.



Figura 5. 5: Análisis de la respuesta en tiempo real del PID en el eje Y.

Estos resultados obtenidos son gracias al algoritmo de control automático PID diseñado en tiempo discreto en Matlab/Simulink que se muestra a continuación.



Figura 5. 6: Algoritmo de control PID en tiempo real.

El resultado del error del controlador PID para la plataforma inercial de 2-GDL es de 5.24%.

Controlador FUZZY.

A continuación, con ayuda de Matlab y el valor exportado de Simulink se muestra el análisis de error del controlador FUZZY.

Se exporta 100 registros de la variable de salida y se lo compara con la referencia y se obtiene los siguientes resultados.

Para el eje X se tiene un error del 5.24% y se adjunta la imagen del control aplicado donde la referencia es del color amarillo, la señal de salida es de color azul y la señal de color marrón es el error.



Figura 5. 7: Análisis de la respuesta en tiempo real del FUZZY en el eje X.

Para el eje Y se tiene un error del 0.09% y se adjunta la imagen del control aplicado donde la referencia es del color amarillo, la señal de salida es de color azul y la señal de color marrón es el error



Figura 5. 8: Análisis de la respuesta en tiempo real del FUZZY en el eje Y.

Estos resultados obtenidos son gracias al algoritmo de control automático PID diseñado en tiempo discreto en Matlab/Simulink que se muestra a continuación.



Figura 5. 9: Algoritmo de control FUZZY en tiempo real.

El resultado del error del controlador PID para la plataforma inercial de 2-GDL es de 2.27%.

Los algoritmos de control ayudan a realizar que el objeto sobre la pantalla se ubique en la referencia establecida, con ayuda del diseño adicional se realiza que el objeto sobre la pantalla sigue una trayectoria definida.

Para el control PID se diseña el seguimiento de un ovalo, se muestra la figura obtenida.



Realizado para el control PID.

Figura 5. 10: Seguimiento de trayectoria con el controlador PID.



Realizado para el control FUZZY.

Figura 5. 11: Seguimiento de trayectoria con el controlador FUZZY.

Se procede a realizar el análisis de resultado con el cálculo del error RMS e índices de desempeño IAE e ISU con la obtención de los valores exportados anteriormente.

	RMSE	IAE	ISE
PID	51,2599	7,2555 x 10^5	1,8117 x 10^4
FUZZY	40,5092	4,4047 x 10^5	8,4824 x 10^3

Tabla 4: Comparativa del error RMS e índices de desempeño.

En la Tabla 4 se muestran los valores para la comparación entre el controlador FUZZY y PID. Se tiene la raíz del error cuadrático medio o RMSE por sus siglas en inglés, para obtener la precisión del sistema. Además, los índices de desempeño como son: el error integral absoluto o IAE por sus siglas en inglés y la desviación de señal de control cuadrado integral o ISE por sus siglas en inglés.

El RMSE se calcula con la siguiente ecuación:

$$RMSE = \sqrt{\left(\sum_{k=1}^{N} (r_k - y_k)^2\right) / N}$$

Los valores RMSE calculados para los controladores PID y FUZZY están llegando a ser aproximadamente 51 y 41 respectivamente, lo que no está mal. Estos valores nos indican que el controlador FUZZY es más preciso que el controlador PID aplicado a la plataforma inercial de 2-GDL. Un buen sistema debe tener un valor RMSE inferior a 180. (Lima, Cajo, Huilcapi, & Agila, 2017)

$$IAE = \sum_{k=1}^{N} |r_k - y_k|$$

Aplicando la ecuación de IAE se obtiene que cuando actúa el controlador PID en el sistema de la plataforma inercial de 2-GDL, el IAE de este controlador es 7,2555 x 103, pero cuando actúa el controlador FUZZY en el plato el IAE es 4,4047 x 103. Dados estos valores se determina que el controlador FUZZY tiene un mejor seguimiento de la referencia para posicionar la esfera en el plato. (Lima et al., 2017)

$$ISU = \sum_{k=0}^{N} (u(k) - u_{ss})^2$$

Aplicando la ecuación de ISU se obtiene valores aproximados de la energía que utiliza cada controlador, con esto puede comparar e indicar cual es el que emplea menor energía en cada ciclo de trabajo. Por lo cual se muestra que el controlador FUZZY obtiene una mejora de rendimiento en la energía utilizada por tener el menor valor de ambos controladores.

ANÁLISIS DE RESULTADO

Para realizar un análisis del proyecto de graduación se realiza una exposición a estudiantes y docentes en la III jornada científica de electrónica y automatización 2019.



Figura 6. 1: Ill jornada científica de electrónica y automatización 2019.

A continuación de la conferencia se realiza una encuesta al grupo de los estudiantes y docentes para medir el grado de aceptación e interés de este proyecto de titulación. Con ayuda de los resultados de la encuesta realizada en la conferencia, se evidencia que el proyecto de titulación tiene un gran interés en los asistentes que son estudiantes y docentes de diversas carreras de la Universidad Politécnica Salesiana, creando un gran impacto en el desarrollo de investigación con la ayuda de la plataforma inercial de 2-GDL.





UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR						
	TRABAJO DE TITULA	CION				
TEMA:	REPOTENCIACIÓN DE LA PLAT LABORATORIO DE CONTROL A CONTROLADORES PID Y FI	AFORM UTOMÁ UZZY	IA INEI ITICO I A TR	RCIAL UTILIZA XAVĖS	DEL NDO DE	
AUTORES:	MATLAB/SIMULINK. JARA LOAIZA WILLIAM ANDRÉS CHALACAN AGUAYO VICENTE DAN	IIEL				
TUTOR:	MSC. VICENTE PEÑARANDA					
Encuesta anón electrónica y a uso de la plata respectivament	ima de precepción a un grupo de asister utomatización 2019 en la Universidad P forma inercial de 2GDL utilizando 2 tipo te.	ite en la olitécnica de contr	III jornad a Salesia oladores	la científ ana acer PID y F	ica de ca del UZZY	
 Es estudiante la Universidad Politécnica Salesiana(Si, llenar a continuación los siguientes datos/No, saltarse esto paso para realizar la encuesta) Carrera: Ciclo que cursa el/la estudiante: 						
Género(masculo/femenino): Cómo calificar los siguientes aspectos de acuerdo a la siguiente ponderación, marcando la casilla correspondiente:						
1. =nada. 2. =muy poco. 3. =poco. 4. =regular. 5. =mucho.						
I						
		nada	muy poco	boco	regular	mucho
¿Cuanto conoce usites de los sitemas inerciales y sus aplicaciones?						
¿Cree usted que es de facil operal	Cree used que el nuevo diseno de la ser uccrucida dene una presentación atractiva? Cree usibid que es de facil operabilidad este modulo para los estudiantes?					
Cree uste di que esta planta ayuda a reforzar la parte teorica para los estudiantes? Domínio de los temas por parte de los expositores.						

Figura 6. 3: Formato de encuesta.

CONCLUSIONES

- Con ayuda del diagnóstico realizado a las partes mecánicas y electrónicas se logra reemplazar los elementos defectuosos o descontinuado de la plataforma inercial de 2-GDL.
- Con la ayuda del Software Matlab (ident) se obtiene el modelo del sistema por cada grado de libertad.
- Se realiza una guia de 9 practicas gracias al correcto funcionamiento de la plataforma inercial de 2-GDL utilizando los controladores PID y FUZZY
- Con los índices de desempeño se tiene que el controlador FUZZY tiene un mejor rendimiento en la energía utilizada y un mejor seguimiento de la referencia para posicionar la esfera en el plato, sobre el controlado PID aplicado al sistema de esta planta.
- Con ayuda de los resultados obtenidos al analizar los controladores aplicados a la plataforma inercial de 2-GDL desarrollados en este trabajo de titulación se determina que el controlador FUZZY en comparación al controlador PID y cuenta con un porcentaje de error RMS 40.5092 siendo este más preciso al compararlo con el controlador PID que tiene un porcentaje de error 51.2999.
- El controlador PID se puede utilizar a pesar de contar un porcentaje de error mayor que el controlador FUZZY, ya que las características de respuesta se encuentran dentro de los rangos aceptables para la utilización de un controlador.
- Se añade un PD al controlador FUZZY para lograr alcanzar los resultados deseados y se obtiene un controlador automático PD DIFUSO.
- Al diseñar el controlador FUZZY se determina que las reglas de correspondencia son iguales tanto para el algoritmo de FUZZY en el eje X y el eje Y, con dichas reglas el controlador opera correctamente.
- Gracias a la tarjeta Arduino DUE se logran diseñar sin ningún problema los controladores PID y FUZZY y se concluye que esta tarjeta tiene un excelente desempeño para la comunicación con el software Matlab/Simulink.
- Con la ayuda de la tarjeta PCB, los servomotores, la pantalla táctil resistiva y la repotenciación física de la plataforma inercial de 2-GDL. Se logra obtener una planta didáctica fácil de transportar y de bajo costo, donde se ejecutan los controladores PID y FUZZY en el software Matlab/Simulink de forma sencilla.
- Gracias a la característica de los elementos implementados en la plataforma inercial de 2-GDL tarjeta PCB en los siguientes trabajos de titulación se puede experimentar con diferentes algoritmos de control.

RECOMENDACIONES

- Antes de realizar pruebas con la plataforma inercial de 2-GDL verificar que el botón de paro de emergencia no esté pulsado.
- Utilizar una esfera que pese entre 92 a 96 gramos para un correcto funcionamiento de la plataforma inercial de 2-GDL.
- Se recomienda que la plataforma inercial de 2-GDL este en un lugar libre de humedad y polvo en exceso.
- Se debe realizar mantenimientos preventivos a la plataforma inercial de 2-GDL proporcionalmente se le de uso.
- Se recomienda utilizar un cable USB macho-macho de 3.0 con supresor de ruido para la comunicación entre la plataforma inercial de 2-GDL y la computadora utilizada.
- Utilizar procesadores core i7 para un correcto funcionamiento del controlador FUZZY.

REFERENCIAS BIBLIOGRÁFICAS

- Alcívar Cevallos, V. I., & Parrales Duarte, I. T. (2015). Diseño e implementación de una plataforma inercial de dos grados de libertad y evaluación del modelado matemático a través de matlab/simulink. Recuperado de http://dspace.ups.edu.ec/handle/123456789/10426
- Aparicio, M. P. (2010). *Diseño y desarrollo de circuitos impresos con KiCad*. RC Libros.
- Área Tecnología. (2016). RELES. Recuperado de https://www.areatecnologia.com/electricidad/rele.html

Artero, Ó. T. (2013). ARDUINO. Curso práctico de formación. RC Libros.

- Cajo, R., & Agila, W. (2015). Evaluation of Algorithms for Linear and Nonlinear PID Control for Twin Rotor MIMO System. 2015 Asia-Pacific Conference on Computer Aided System Engineering, 214-219. https://doi.org/10.1109/APCASE.2015.45
- Cárdenas Ibáñez, M. D. L. A., & Peñaranda Idrovo, V. (2019). Análisis comparativo de algoritmos de control pid, fuzzy y predictivo aplicado a sistemas seguidores de la posición del sol para la captación de energía solar usando paneles fotovoltaicos. Recuperado de

http://www.dspace.espol.edu.ec/handle/123456789/46062

Castillo, J. C. M. (2017). Electrónica. Novedad 2017. Editex.

Cruz-Oré, D. L., & Luis, J. (2013). ¿Qué significan los grados de libertad? *Revista Peruana de Epidemiología*, *17*(2). Recuperado de http://www.redalyc.org/resumen.oa?id=203129458002

Delgado Crespo, M. (2016). Arduino en español: Arduino DUE. Recuperado de Arduino en español website:

http://manueldelgadocrespo.blogspot.com/p/arduino-due.html

Direct Industry. (2019). Líder de mundo en servomotores y motores de pasos

integrados. JVLDirectIndustry. Recuperado de

http://trends.directindustry.es/jvl-s/project-22516-126937.html

- Echeverría, P. M., Angulo, I., & Angulo, J. M. (2008). Pantallas táctiles para mejorar la presentación y el funcionamiento de nuestros productos. 4.
- El Arduino. (2019, febrero 17). El Arduino NANO. Características de la placa Arduino NANO. Recuperado 4 de julio de 2019, de El Arduino website: https://www.elarduino.com/nano/
- GOILAV, N., & LOI, G. (2016). Arduino: Aprender a desarrollar para crear objetos inteligentes. Ediciones ENI.
- Gómez, E. (2017, diciembre 19). Que es PWM y para qué sirve. Recuperado de Rincón Ingenieril website: https://www.rinconingenieril.es/que-es-pwm-ypara-que-sirve/
- Harper, G. E. (2005). Protección de instalaciones eléctricas industriales y comerciales. Editorial Limusa.
- Huilcapi Subía, V. M. (2015). Diseño de un algoritmo de control inteligente utilizando lógica difusa comparado con un algoritmo PID, para su implementación en un sistema multivariable no lineal bola y plato. Recuperado de http://www.dspace.espol.edu.ec/handle/123456789/30110
- Ibarra, R., & López, M. S. (2007). *Principios de Teoría de Las Comunicaciones*. Editorial Limusa.
- Idis, D. (1996). 6DOF | IDIS. Recuperado de 6dof website: https://proyectoidis.org/6dof/
- Lima, B., Cajo, R., Huilcapi, V., & Agila, W. (2017). Modeling and comparative study of linear and nonlinear controllers for rotary inverted pendulum. *Journal of Physics: Conference Series*, 783, 012047. https://doi.org/10.1088/1742-6596/783/1/012047

MathWorks. (2019). MATLAB Product Description-MATLAB & Simulink.

Recuperado de MATLAB Product Description website:

https://www.mathworks.com/help/matlab/learn_matlab/productdescription.html

- Matos, J. F. B., Rodríguez, R. A. J., & Toledo, D. R. G. (2018). Matlab/Simulink:
 Modelos para simular mediciones eléctricas en circuitos r, rl y rlc
 monofásicos. *Roca: Revista Científico Educaciones de la provincia de Granma*, *14*(2 (abril-junio)), 102-115.
- Mecafenix, F. (2017, abril 18). Servomotor ¿Qué es y cómo funciona? Recuperado 24 de junio de 2019, de Ingeniería Mecafenix website: https://www.ingmecafenix.com/electricidad-industrial/servomotor/
- Millahual, C. P. (2017). Arduino De Cero a Experto: Proyectos Prácticos -Electrónica, hardware y programación. RedUsers.
- Naylamp, M. (2018). Arduino DUE R3. Recuperado 4 de julio de 2019, de Naylamp Mechatronics-Perú website: https://naylampmechatronics.com/arduinotarjetas/172-arduino-due-r3.html
- Ogata, K. (2010). Ingeniería de control moderna. Pearson Educación.
- Olmo Nave. (2017). Filter Circuits with Capacitors. Recuperado de http://hyperphysics.phy-astr.gsu.edu/hbasees/electric/filcap2.html#c3
- Pardo, C. (2013). Controlador PID-Control Automático-Picuino. Recuperado de https://www.picuino.com/es/arduprog/control-pid.html
- Rashid, M. H. (2004). *Electrónica de potencia: Circuitos, dispositivos y aplicaciones*. Pearson Educación.
- Roederer, J. G. (2002). *Mecánica Elemental*. Editorial Universitaria de Buenos Aires SEM.
- Rubio, F. R., Ortega, M. G., & Gordillo, F. (2010). Control de Posición e Inercial de
 Plataforma de Dos Grados de Libertad. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 7(4), 65-73.

https://doi.org/10.1016/S1697-7912(10)70061-3

Sung, S. W., Lee, J., & Lee, I.-B. (2009). *Process Identification and PID Control.* John Wiley & Sons.

Tomasi, W. (2003). Sistemas de comunicaciones electrónicas. Pearson Educación.

ANEXOS

ANEXO 1: TABLAS DE DATOS GENERADOS DE LA SEÑAL PRBS

Motor X		
Tiem	Grad	
1	114	
2	114	
3	114	
4	114	
5	114	
6	114	
7	114	
8	114	
0 0	11/	
10	126	
10	120	
10	120	
12	120	
13	126	
14	114	
15	114	
16	114	
17	114	
18	126	
19	114	
20	114	
21	114	
22	126	
23	126	
24	126	
25	126	
26	114	
27	126	
28	114	
20	114	
29	114	
30	120	
31	120	
32	114	
33	114	
34	126	
35	114	
36	114	
37	126	
38	114	
39	114	
40	114	
41	114	
42	126	
43	114	
44	126	
45	126	
46	126	
47	126	

48	114
49	114
50	114
51	126
51	120
52	120
53	114
54	114
55	126
56	126
57	126
58	126
59	114
60	126
61	126
62	114
63	126
64	126
65	126
66	114
00	114
67	120
68	114
69	126
70	114
71	114
72	114
73	126
74	114
75	126
76	114
77	114
78	114
79	114
80	126
00 01	120
01	114
02	114
83	120
84	126
85	114
86	126
87	114
88	114
89	114
90	126
91	126
92	114
93	114
94	114
94	126
06	120
30	120

97	126
98	126
99	126
100	126
101	114
102	114
102	114
104	126
105	114
106	114
107	114
108	126
100	114
110	126
111	126
112	114
112	11/
110	114
114	114
110	114
117	120
117	114
110	120
119	114
120	120
121	126
122	114
123	126
124	114
125	126
126	126
127	126
128	126
129	126
130	126
131	114
132	126
133	114
134	126
135	114
136	126
137	114
138	126
139	114
140	114
141	114
142	114
143	114
144	126
145	114

146	126
147	114
148	114
149	126
150	114
151	126
152	126
153	126
154	126
155	126
156	114
157	114
158	126
159	114
160	114
161	114
162	126
163	114
164	114
165	126
166	114
167	114
168	126
169	114
170	126
170	114
172	11/
172	126
174	120
175	120
176	120
170	120
178	11/
170	126
180	11/
100	114
101	114
102	114
103	11/
104	114
100	114
100	114
107	114
100	106
109	120
190	120
100	114
192	114
193	114
194	114

195	114		-
196	126		
197	126		-
198	114		-
199	114		-
200	126	_	-
201	114		-
202	126		-
202	126		-
200	114		-
205	114		-
200	126		-
207	114		-
208	126		-
200	11/	_	_
203	114		-
210	114		-
217	126		-
212	120		-
213	120		_
214	11/		_
215	114		_
210	114		_
217	11/		_
210	114		_
219	120		_
220	120	-	_
221	120	_	
222	114	_	
223	120	_	
224	114	_	
223	114	_	
220	114	_	
221	114	_	
220	114	_	
229	114	_	
230	114	_	
231	120	_	
232	114	_	
233	120	_	
234	120	_	
230	114	_	
230	120	_	
231	114	_	
230	114	_	
239	120	_	
240	120	-	_
241	144	-	_
242	114	-	
243	120	_	
244	114	_	
245	120	_	
246	120	_	
247	114		

248	114
249	126
250	126
251	126
252	114
253	114
254	126
255	126
256	120
250	120
201	120
200	120
259	126
260	126
261	114
262	114
263	126
264	126
265	114
266	114
267	126
268	126
269	114
270	126
270	11/
271	126
272	120
2/3	114
274	114
275	126
276	126
277	114
278	126
279	126
280	114
281	114
282	114
283	114
284	114
285	114
286	126
287	114
288	11/
200	126
209	120
290	114
291	126
292	126
293	114
294	126
295	126
296	114
297	126
298	126
299	114
300	114
2.2.0	· · · ·

301	126
302	114
303	114
304	114
305	114
306	11/
207	114
307	114
308	120
309	126
310	114
311	126
312	114
313	114
314	126
315	114
316	126
317	114
318	126
210	120
200	120
320	120
321	126
322	114
323	126
324	114
325	126
326	126
327	126
328	114
329	126
330	126
331	11/
222	114
<u> </u>	114
333	114
334	126
335	114
336	114
337	126
338	126
339	114
340	126
341	114
342	114
343	114
344	114
215	126
240	111
340	114
347	114
348	126
349	126
350	126
351	126
352	114
353	114

354	126
355	114
356	126
357	114
358	126
250	120
309	114
360	126
361	126
362	114
363	114
364	114
365	126
366	126
367	114
368	126
360	126
370	120
274	120
3/1	120
372	114
373	114
374	126
375	126
376	126
377	114
378	126
379	126
380	126
381	126
201	120
302	114
383	120
384	126
385	126
386	126
387	126
388	126
389	126
390	126
391	114
392	126
303	126
304	120
205	120
395	114
396	126
397	126
398	126
399	114
400	114
401	126
402	126
403	114
404	126
405	126
400	106
400	120

407	114	
408	114	
409	114	
410	126	
411	114	
412	126	
413	114	
414	126	
415	114	
416	114	
417	126	
418	114	
419	114	
420	126	
421	126	
422	126	
423	114	
424	114	
425	114	
426	126	
427	126	
428	126	
429	114	
430	126	

431	126
432	114
433	126
434	114
435	126
436	114
437	126
438	126
439	126
440	114
441	114
442	126
443	114
444	114
445	126
446	126
447	114
448	114
449	114
450	114
451	114
452	126
453	126
454	114

455	114
456	114
457	114
458	126
459	126
460	126
461	114
462	126
463	114
464	114
465	126
466	114
467	114
468	114
469	126
470	126
471	114
472	126
473	114
474	126
475	126
476	114
477	126
478	126

479	126
480	126
481	126
482	114
483	126
484	126
485	114
486	114
487	126
488	126
489	114
490	114
491	114
492	126
493	114
494	126
495	126
496	126
497	114
498	114
499	114
500	114

 Tabla 5: Datos generados de la señal PRBS-Motor X.

Mot	or Y
Tiempo	Grados
1	47,5
2	47,5
3	47,5
4	47,5
5	47,5
6	47,5
7	47,5
8	47,5
9	47,5
10	52,5
11	52,5
12	52,5
13	52,5
14	47,5
15	47,5
16	47,5
17	47,5
18	52,5
19	47,5
20	47,5
21	47,5
22	52,5
23	52,5

24	52,5
25	52,5
26	47,5
27	52,5
28	47,5
29	47,5
30	52,5
31	52,5
32	47,5
33	47,5
34	52,5
35	47,5
36	47,5
37	52,5
38	47,5
39	47,5
40	47,5
41	47,5
42	52,5
43	47,5
44	52,5
45	52,5
46	52,5
47	52,5
48	47,5

49	47,5
50	47,5
51	52,5
52	52,5
53	47,5
54	47,5
55	52,5
56	52,5
57	52,5
58	52,5
59	47,5
60	52,5
61	52,5
62	47,5
63	52,5
64	52,5
65	52,5
66	47,5
67	52,5
68	47,5
69	52,5
70	47,5
71	47,5
72	47,5
73	52,5

74	47,5
75	52,5
76	47,5
77	47,5
78	47,5
79	47,5
80	52,5
81	52,5
82	47,5
83	52,5
84	52,5
85	47,5
86	52,5
87	47,5
88	47,5
89	47,5
90	52,5
91	52,5
92	47,5
93	47,5
94	47,5
95	52,5
96	52,5
97	52,5
98	52,5

99	52,5	150	47,5		201	47,5	252	47,5
100	52,5	151	52,5		202	52,5	253	47,5
101	47,5	152	52,5		203	52,5	254	52,5
102	47,5	153	52,5		204	47,5	255	52,5
103	47,5	154	52,5		205	47,5	256	52,5
104	52,5	155	52,5		206	52,5	257	52,5
105	47,5	156	47,5		207	47,5	258	52,5
106	47,5	157	47,5		208	52,5	259	52,5
107	47,5	158	52,5		209	47,5	260	52,5
108	52,5	159	47,5		210	47,5	261	47,5
109	47,5	160	47,5		211	47,5	262	47,5
110	52,5	161	47,5		212	52,5	263	52,5
111	52,5	162	52,5		213	52,5	264	52,5
112	47,5	163	47,5		214	52,5	265	47,5
113	47,5	164	47,5		215	47,5	266	47,5
114	47,5	165	52,5		216	47,5	267	52,5
115	47,5	166	47,5		217	52,5	268	52,5
116	52,5	167	47,5		218	47,5	269	47,5
117	47,5	168	52,5		219	52,5	270	52,5
118	52,5	169	47,5		220	52,5	271	47,5
119	47,5	170	52,5		221	52,5	272	52,5
120	52,5	171	47,5		222	47,5	273	47,5
121	52,5	172	47,5		223	52,5	274	47,5
122	47,5	173	52,5		224	47,5	275	52,5
123	52,5	174	52,5		225	47,5	276	52,5
124	47,5	175	52,5		226	47,5	277	47,5
125	52,5	176	52,5		227	47,5	278	52,5
126	52,5	177	52,5		228	47,5	279	52,5
127	52,5	178	47,5		229	47,5	280	47,5
128	52,5	179	52,5		230	47,5	281	47,5
129	52,5	180	47,5		231	52,5	282	47,5
130	52,5	181	47,5		232	47,5	283	47,5
131	47,5	182	47,5		233	52,5	284	47,5
132	52,5	183	52,5		234	52,5	285	47,5
133	47,5	184	47,5		235	47,5	286	52,5
134	52,5	185	47,5		236	52,5	287	47,5
135	47,5	186	47,5		237	47,5	288	47,5
136	52,5	187	47,5		238	47,5	289	52,5
137	47,5	188	47,5		239	52,5	290	47,5
138	52,5	189	52,5		240	52,5	291	52,5
139	47,5	190	52,5		241	52,5	292	52,5
140	47,5	191	52,5		242	47,5	293	47,5
141	47,5	192	47,5		243	52,5	294	52,5
142	47,5	193	47,5		244	47,5	295	52,5
143	47,5	194	47,5		245	52,5	296	47,5
144	52,5	195	47,5		246	52,5	297	52,5
145	47,5	196	52,5		247	47,5	298	52,5
146	52,5	197	52,5		248	47,5	299	47,5
147	47,5	198	47,5		249	52,5	300	47,5
148	47,5	199	47,5	1	250	52,5	301	52,5
149	52,5	200	52,5]	251	52,5	302	47,5

303	47,5		354	52,5		405	52,5	456	47,5
304	47,5		355	47,5		406	52,5	457	47,5
305	47,5		356	52,5		407	47,5	458	52,5
306	47,5		357	47,5		408	47,5	459	52,5
307	47,5		358	52,5		409	47,5	460	52,5
308	52,5		359	47,5		410	52,5	461	47,5
309	52,5		360	52,5		411	47,5	462	52,5
310	47,5		361	52,5		412	52,5	463	47,5
311	52,5		362	47,5		413	47,5	464	47,5
312	47,5		363	47,5		414	52,5	465	52,5
313	47,5		364	47,5		415	47,5	466	47,5
314	52,5		365	52,5		416	47,5	467	47,5
315	47,5		366	52,5		417	52,5	468	47,5
316	52,5		367	47,5		418	47,5	469	52,5
317	47,5		368	52,5		419	47,5	470	52,5
318	52,5		369	52,5		420	52,5	471	47,5
319	52,5		370	52,5		421	52,5	472	52,5
320	52,5		371	52,5		422	52,5	473	47,5
321	52,5		372	47,5		423	47,5	474	52,5
322	47,5		373	47,5		424	47,5	475	52,5
323	52,5		374	52,5		425	47,5	476	47,5
324	47,5		375	52,5		426	52,5	477	52,5
325	52,5		376	52,5		427	52,5	478	52,5
326	52,5		377	47,5		428	52,5	479	52,5
327	52,5		378	52,5		429	47,5	480	52,5
328	47,5		379	52,5		430	52,5	481	52,5
329	52,5		380	52,5		431	52,5	482	47,5
330	52,5		381	52,5		432	47,5	483	52,5
331	47,5		382	47,5		433	52,5	484	52,5
332	47,5		383	52,5		434	47,5	485	47,5
333	47,5		384	52,5		435	52,5	486	47,5
334	52,5		385	52,5		436	47,5	487	52,5
335	47,5		386	52,5		437	52,5	488	52,5
336	47,5		387	52,5		438	52,5	489	47,5
337	52,5		388	52,5		439	52,5	490	47,5
338	52,5		389	52,5		440	47,5	491	47,5
339	47,5		390	52,5		441	47,5	492	52,5
340	52,5		391	47,5		442	52,5	493	47,5
341	47,5		392	52,5		443	47,5	494	52,5
342	47,5		393	52,5		444	47,5	495	52,5
343	47,5	1	394	52,5		445	52,5	496	52,5
344	47,5	1	395	47,5	1	446	52,5	497	47,5
345	52,5	1	396	52,5	1	447	47,5	498	47,5
346	47.5	1	397	52.5	1	448	47.5	499	47.5
347	47,5	1	398	52,5	1	449	47,5	500	47,5
348	52,5	1	399	47,5	1	450	47,5		,
349	52,5	1	400	47,5	1	451	47,5		
350	52.5	1	401	52.5	1	452	52.5		
351	52.5	1	402	52.5	1	453	52.5		
352	47.5	1	403	47.5	1	454	47.5		
353	47,5	1	404	52,5	1	455	47,5		

Tabla 6: Datos generados de la señal PRBS-Motor Y.

ANEXO 2: CRONOGRAMA DE ACTIVIDADES

Carbicol de Partes Image: Carbon de Partes <thimage: carbon="" de="" par<="" th=""><th>uescanicas y electronicas versión de Partes xperimentación oci uescanicas uoros a importación de uescanicas conseño de partes uescanicas uescanicas uescanicas uoros y pantalia uescanicas uoros y pantalia uescanicas uescanicas uescanicas<</th><th>Nombre del trabajo: Actividad</th><th>Repotenciamiento de la plataforma inercial del laboratorio de control automático utilizando controladores PID y fuzzy a través de MATLAB sep-17 oct-17 nov-17 dic-17 ene-18 feb-18 mar-18 mar-18 may-18 jun-18 jun-18 jun-18 cot-18 nov-18 dic-18 ene-19 feb-19 mar-19 jabr-19 may-19 jun-1</th></thimage:>	uescanicas y electronicas versión de Partes xperimentación oci uescanicas uoros a importación de uescanicas conseño de partes uescanicas uescanicas uescanicas uoros y pantalia uescanicas uoros y pantalia uescanicas uescanicas uescanicas<	Nombre del trabajo: Actividad	Repotenciamiento de la plataforma inercial del laboratorio de control automático utilizando controladores PID y fuzzy a través de MATLAB sep-17 oct-17 nov-17 dic-17 ene-18 feb-18 mar-18 mar-18 may-18 jun-18 jun-18 jun-18 cot-18 nov-18 dic-18 ene-19 feb-19 mar-19 jabr-19 may-19 jun-1
Superimetación con la Superimetación con la Superimetación de Image: Controlado de partas Image: Controlado de	gataforma inercial gatafor	Revision de Partes mecanicas y electronicas	
Igaladroma inercial Image: Image	intercial intercial <t< td=""><td>Experimentacion con la</td><td></td></t<>	Experimentacion con la	
Indices y Dantalla Implementation del Implementation <	uotores y partala unitala unitala <thunitala< th=""> unitala <thunitala< th=""></thunitala<></thunitala<>	Compro cimportoción do	
Rediseiro de pantes I	ediseño de partes ediseño de partes jedialicas i	motores y pantalla	
necanicas I	Inscaincians Image	Rediseño de partes	
Avariaje de motiviniento I </td <td>bintaje de motores y xperimentacion del xperimentacion del istema, empleando dent'e Invodelo del istema, empleando dent'e I</td> <td>mecanicas</td> <td></td>	bintaje de motores y xperimentacion del xperimentacion del istema, empleando dent'e Invodelo del istema, empleando dent'e I	mecanicas	
nuebas de movimiento I <thi< th=""> I I <thi< th=""></thi<></thi<>	nuebas de movimiento I <thi< th=""> I I <thi< th=""></thi<></thi<>	Montaje de motores y	
Experimentacion del Image: Controlador PID para la Jettaforma inercial Image: Controlador PID para la Vestigacion del sizema, empleando Image: Controlador PID para la Superimentacion del Image: Controlador PID para la Superimentacion del site Image: Controlador PID para la Jataforma inercial Image: Controlador PID para la J	xperimentacion del ontrolador PID para la lataforma inercial I	pruebas de movimiento	
Jatatorma inercial I <td>glatdorma inercial I</td> <td>Experimentacion del</td> <td></td>	glatdorma inercial I	Experimentacion del	
Idetarforma inercial Intercial	lataforma inercial effairi el modelo del erfairi el modelo del ent'en Matlab/Simulink istema, empleando dent'en Matlab/Simulink dent'en Matlab/Simulink unuello del vestigacion delogica unuello del itusa unuello del vostigacion delogica unuello del itusa unuello del vostigacion delogica unuello del del itusa unuello del del del vostigacion delogica unuello del	controlador PID para la	
Definir el modelo del istrema, empleando ident" en Matlab/Simulink renstigacion del ógica investigacion del ógica istrema, empleando istrema, empleando istrema, empleando </td <td>effinir el modelo del istema, empleando dent" en Mattab/Simulink istema, empleando istama, empleando <</td> <td>plataforma inercial</td> <td></td>	effinir el modelo del istema, empleando dent" en Mattab/Simulink istema, empleando istama, empleando <	plataforma inercial	
istema, empleando istema, empleando investigacion del ógica investigacion del ógica investigacion del ol	istema, empleando dent" en Matlab/Simulink I <td>Definir el modelo del</td> <td></td>	Definir el modelo del	
ident" en Matlab/Simulink I<	dent" en Mattab/Simulink I </td <td>sistema, empleando</td> <td></td>	sistema, empleando	
vestigacion de lógica vestigacion de lógica vestigacion de lógica vestigacion de lógica Xuperimentacion del vestigacion de lógica vestigacion de lógica vestigacion de lógica Xuperimentacion del vestigacion de lógica vestigacion de lógica vestigacion de lógica vestigacion de lógica Vestigacion de la guia de vestigacion de la guia de logica vestigacion de logica </td <td>vestigacion de lógica Image: Circular Sector Se</td> <td>"ident" en Matlab/Simulink</td> <td></td>	vestigacion de lógica Image: Circular Sector Se	"ident" en Matlab/Simulink	
iffusa Image: Second Secon	fusa rusa	Investigacion de lógica	
Experimentacion del Image: Controlador FUZZY para la controlador FUZZY para la Image: Controlador FUZZY para la lataforma inercial Image: Controlador FUZZY para la <t< td=""><td>xperimentacion del xperimentacion del controlador FUZZY para la lataforma inerciai lataforma inerciai I iseño de la guia de I racticas I iveño de la guia de I racticas I ruebas y verificacion de I guia de practicas I ocumentación del I guia de practicas I locumentación del I royecto I royecto I royecto I locumentación técnica I locumentación del I locumentación del I royecto I royecto I locumentación técnica I <tr< td=""><td>difusa</td><td></td></tr<></td></t<>	xperimentacion del xperimentacion del controlador FUZZY para la lataforma inerciai lataforma inerciai I iseño de la guia de I racticas I iveño de la guia de I racticas I ruebas y verificacion de I guia de practicas I ocumentación del I guia de practicas I locumentación del I royecto I royecto I royecto I locumentación técnica I locumentación del I locumentación del I royecto I royecto I locumentación técnica I <tr< td=""><td>difusa</td><td></td></tr<>	difusa	
controlador FUZZY para la	ontrolador FUZZY para la Intercial	Experimentacion del	
Idataforma inercial Image: Constraint of the constraint	Iztaforma inercial Intercial Iztaforma inercial Intercial Izseño de la guía de Intercial Izseño de la guía de Intercial Iztaforma inercial Intercial Izseño de la guía de Intercial Iztaforma inercial Intercial	controlador FUZZY para la	
Diseño de la guía de international de la guía de international de la guía de international de la guía de practicas en la guía	iseño de la guía de iseño de la guía de racticas iano de la guía de iracticas iano de la guía de ruebas y verificacion de iano de la guía de iracticas iano de la guía de iractica iano de la guía de	plataforma inercial	
rracticas	racticas Image: Constraint of the cons	Diseño de la guia de	
Pruebas y verificación de	ruebas y verificación de	practicas	
a guia de practicas	uguia de practicas	Pruebas y verificacion de	
Jocumentación del	vocumentación del	la guia de practicas	
	royecto	Documentación del	
	tevisión técnica	proyecto	

ÍTEM	DESCRIPCIÓN	VALOR
1	Servomotores Fan Model 20 Kg	\$ 47,98
2	Pantalla táctil resistiva 4 wire	\$ 60,99
3	Esferas de 1 pulg. y 1.5 pulg.	\$ 15,00
4	Rediseño de estructura mecánica	\$350,00
5	Soportes de teflón	\$ 30,00
6	Brazos metálicos para servomotores	\$ 20,00
7	Tarjeta de Circuito Impreso	\$ 80,00
8	Arduino DUE	\$ 45,00
9	Arduino NANO	\$ 15,00
10	Botoneras y luces piloto	\$ 15,00
11	Fuente de poder	\$ 8,00
12	Varios (movilización, material didáctico, papelería)	\$ 300,00
13	Costos de importación	\$ 40,81
14	Mano de obra	\$ 350,00
	ΤΟΤΑ	L \$1.377,78

ANEXO 3: PRESUPUESTO DEL PROYECTO DE TITULACIÓN

ANEXO 4: FACTURAS DE EQUIPOS Y ELEMENTOS VARIOS

SERVI O Fecha: J. Cliente: Q Dirección; R.U.C./C.I.	GUAYA CIO DE BL 6/03/20 ARA LOA CDLA LA	QUIL - E ISOUEDA 19 IZA WII CHALA 534	LIAM ANDRES	OBLIGADO A LLEVA FACT 001-002-00 0 0 6 DOCUMENTO CAT FECHA DE AUTORIZACI	URA URA 1490 EGORIZADO: NO ON: 18 FEBRERO 2019
CANT		D	DESCRIPCION	P IINIT	TOTAL
2	11225		LM358P	0,670	1.339
4	80		100 UF 16V 105	0 134	0.536
4	120		22 UF 16V	0.134	0.536
2	11904		RESISTENCIA DE ALAMBRE DE 1KOHM D	0.089	0.179
1	8553		JUMPER MACHO A MACHO 10CM 30 PIN	1.116	1.116
1	5800		PORTAFUSIBLE GRANDE 250V/10A AMER	0.446	0.446
3	7339	_	FUSIBLE AMERICANO 6X30MM 2AMP 250	0.179	0.536
****	******	**	Son 7 articulos despachados ****** *	**** * *****	*****
			VILCONSA S.A. CANZELADO		
FORMA	DE PAGO	То	tal: cinco/con 25/100 Dolares	SUB-TOTAL \$	
EFECTIVO		Son:	Dólares,		4.688
HNERO LECTRÓNICO			10	I.V.A. U 70	
CRÉDITO/DÉBITO				I.V.A. 12 %	0.563
RANSF, (CHEQUE)			Firma Autorizada Recibi Conforme	TOTAL \$	\$ 5.25

amazon.com

Final Details for Order #114-9904307-3602667 Print this page for your records.

Order Placed: August 10, 2018 Amazon.com order number: 114-9904307-3602667 Order Total: \$99.99

Shipped on August 12, 2018

Items Ordered

1 of: *NJYTouch 15inch 4 Wire Resistive Touch Screen Panel 322x247mm For 15"* \$60.99 Laptop LCD Screen Sold by: NJYTouch (seller profile)

Condition: New

Shipping Address: TMA01194 JORGE GARCIA 8276 NW SOUTH RIVER DR MEDLEY, FL 33166-7420 United States Item(s) Subtotal: \$60.99 Shipping & Handling: \$39.00 Total before tax: \$99.99 Sales Tax: \$0.00 -----Total for This Shipment:\$99.99

Price

Shipping Speed: Expedited Shipping

amazon.com

Final Details for Order #114-6536629-3054651 Print this page for your records.

Order Placed: August 10, 2018 Amazon.com order number: 114-6536629-3054651 Order Total: \$108.57

Shipped on August 16, 2018					
Items Ordered 2 of: Fan Model FS-20W Waterproof High Torque Metal Gear Standard Digital Servo 20KG/.14S for 1/8 1/10 scale RC Cars Sold by: Fan Model (<u>seller profile</u>)	Price \$23.99				
Condition: New 1 of: JINGRAYS 6mm Thick Curb Chain Necklace for Men Biker Punk Style, Male Stainless Steel Chain Link, 24 inches -Silver Sold by: JINGRAYS (<u>seller profile</u>)	\$9.99				
Condition: New 1 of: Tonnier Black Slim Stainless Steel Mesh Strap Mens Watch Quartz Watch for Men Golden Hands, Tonnier Sold by: tongnianfield (<u>seller profile</u>) Product question? <u>Ask Seller</u>	\$22.99				
Condition: New 1 of: <i>Tommy Hilfiger Men's Dore Passcase Billfold Wallet,British Tan,One Size</i> Sold by: Amazon.com Services, Inc Condition: New	\$25.80				
Shipping Address:Item(s) Subtotal:TMA01194 JORGE GARCIAShipping & Handling:8276 NW SOUTH RIVER DRFree Shipping:MEDLEY, FL 33166-7420Total before tax:United StatesSales Tax:	\$106.76 \$7.92 -\$7.92 \$106.76 \$1.81				
Shipping Speed: Total for This Shipment: FREE Shipping	\$108.57 				



ANEXO 5: DATASHEET PANTALLA TACTIL RESISTIVA