	GESTIÓN DE SERVICIOS ACADÉMICOS Y BIBLIOTECARIOS		CÓDIGO	FO-GS-15	
			VERSIÓN	02	
	ESQUEMA HOJA DE RESUMEN			FECHA	03/04/2017
				PÁGINA	1 de 1
ELABORÓ		REVISÓ		APROBÓ	
Jefe División de Biblioteca		Equipo Operativo de Calidad		Líder de Calidad	

RESUMEN TRABAJO DE GRADO

AUTOR(ES): Ronaldo Sánchez Leal y Camila Guadalupe Camperos Amaya

NOMBRE(S) Ronaldo APELLIDOS: Sánchez Leal

NOMBRE(S) Camila Guadalupe APELLIDOS: Camperos Amaya

FACULTAD: Ingeniería

PLAN DE ESTUDIOS: Ingeniería Electromecánica

DIRECTOR:

NOMBRE(S): Cristián Leonardo APELLIDOS: Tarazona Celis

TÍTULO DEL TRABAJO (TESIS): VALIDACIÓN DE TÉCNICAS DE CONTROL Y SINTONIZACIÓN USANDO COMO PROCESO UN MOTOR DC

La siguiente propuesta se basa en usar un motor DC como sistema o proceso a controlar para realizar la validación de las distintas técnicas de control convencionales, partiendo de la obtención de un modelo matemático a partir de diferentes técnicas, para su posterior validación a través de la simulación.

Una vez obtenido el respectivo modelo del motor DC a utilizar se procede a determinar el BIAS del proceso para el punto de referencia seleccionado, con los datos obtenidos se determinan la constantes del controlador utilizando diferentes técnicas de sintonización para el control convencional (P,PI,PID) por medio del desarrollo de distintos programas basados en Arduino, se observará una comparación entre cada técnica de control convencional y se determinará en base a parámetros de diseño la mejor opción a utilizar.

PALABRAS CLAVES: (ESCRIBIR MÁXIMO 5)

-Parametrización. –Motor DC. –Técnicas de Control. –Sintonización. –Validación.

CARACTERÍSTICAS:

PÁGINAS: 89

PLANOS: NO

CD ROOM: NO

ILUSTRACIONES: 54.

VALIDACIÓN DE LAS TÉCNICAS DE CONTROL Y SINTONIZACIÓN
USANDO COMO PROCESO UN MOTOR DC

RONALDO SANCHEZ LEAL

CAMILA GUADALUPE CAMPEROS AMAYA

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

FACULTAD DE INGENIERÍA

PROGRAMA ACADÉMICO INGENIERÍA ELECTROMECAÁNICA

SAN JOSÉ DE CÚCUTA

2023

VALIDACIÓN DE TÉCNICAS DE CONTROL Y SINTONIZACIÓN USANDO
COMO PROCESO UN MOTOR DC

Proyecto de grado

Requisito para optar al título de Ingeniero Electromecánico

Director

MSc. Cristián Leonardo Tarazona Celis

Codirector

PhD. Jaime Antonio González Castellanos

Presentado A:

Comité Curricular de Ingeniería Electromecánica

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

FACULTAD DE INGENIERÍA

PROGRAMA ACADÉMICO INGENIERÍA ELECTROMECAÁNICA

SAN JOSÉ DE CÚCUTA

2023



**ACTA DE SUSTENTACIÓN PROYECTO DE GRADO
MODALIDAD TRABAJO INVESTIGATIVO**

FECHA: 7 de noviembre de 2023

HORA: 8:00 a.m.

LUGAR: Sala 4 del Cread Ufps

PLAN DE ESTUDIOS: INGENIERÍA ELECTROMECAÁNICA

TITULO DEL TRABAJO DE GRADO: "VALIDACIÓN DE LAS TÉCNICAS DE CONTROL Y SINTONIZACIÓN USANDO COMO PROCESO UN MOTOR DC"

JURADOS: MSc: GLORIA ESMERALDA SANDOVAL MARTINEZ
Esp: ALEXANDRA GALVIS MONTAGUT

DIRECTOR: MSc: CRISTIAN LEONARDO TARAZONA CELIS
CODIRECTOR PhD: JAIME ANTONIO CASTELLANOS GONZÁLEZ

APROBADA

NOMBRE DEL ESTUDIANTE:	CÓDIGO	CALIFICACION
RONALDO SÁNCHEZ LEAL	1091018	4.4
CAMILA GUADALUPE CAMPEROS AMAYA	1091342	4.4

FIRMA DE LOS JURADOS:

VOBO. COORDINADOR COMITÉ CURRICULAR

Magorline Ch.

TABLA DE CONTENIDO

INTRODUCCIÓN	9
1. PLANTEAMIENTO DEL PROBLEMA	10
3. OBJETIVOS	12
3.1 OBJETIVO GENERAL	12
3.2 OBJETIVOS ESPECÍFICOS	12
6. MARCO REFERENCIAL	15
6.1 ANTECEDENTES	15
6.1.1 <i>Internacional</i>	15
6.1.2 <i>Nacional</i>	15
6.2 MARCO TEÓRICO.....	17
6.2.1 <i>Motor DC</i>	17
6.2.2 <i>Modelo matemático de un motor DC</i>	18
6.2.3 <i>Técnicas de control</i>	24
6.2.3.1 Técnicas de control convencional.	24
6.2.3.1.1 <i>Acción de control proporcional.</i>	24
6.2.3.1.2 <i>Acción de control proporcional integradora (PI).</i>	27
6.2.3.1.3 <i>Acción de control proporcional integradora derivativa (PID).</i>	29
6.2.4 <i>Encoder.</i>	31
6.2.5 <i>Arduino MEGA.</i>	32
6.3 MARCO LEGAL	34
7. DISEÑO METODOLÓGICO	35
7.1 ACTIVIDADES Y METODOLOGÍAS	35
8. DESARROLLO DE LA INVESTIGACIÓN	39
8.1 MODELO MATEMÁTICO DE UN MOTOR DC.....	39

8.2	CONSTANTES DEL CONTROLADOR MEDIANTE DIFERENTES TÉCNICAS DE SINTONIZACIÓN	45
8.3	SOFTWARE BASADO EN ARDUINO PARA IMPLEMENTAR LAS DISTINTAS TÉCNICAS DE CONTROL CONVENCIONALES	67
8.4	INTERFAZ GRÁFICA PARA LA SUPERVISIÓN DE LA VELOCIDAD DEL MOTOR DC.....	70
8.5	VALIDACIÓN DE LAS TÉCNICAS DE CONTROL CONVENCIONALES	72
9.	CONCLUSIONES.....	79
10.	RECOMENDACIONES	81
11.	REFERENCIAS	82
	ANEXOS	87

LISTA TABLAS

<i>TABLA 1. NORMAS APLICADAS A MOTORES DC.....</i>	<i>34</i>
<i>TABLA 2. RESULTADOS OBTENIDOS PARA HALLAR LA CONSTANTE ELECTROMOTRIZ.....</i>	<i>39</i>
<i>TABLA 3. RESULTADOS OBTENIDOS PARA HALLAR EL COEFICIENTE DE FRICCIÓN B.....</i>	<i>42</i>
<i>TABLA 4. VARIABLES DE ZIEGLER Y NICHOLS - MÉTODO 1.</i>	<i>48</i>

LISTA DE FIGURAS

<i>ILUSTRACIÓN 1. MODELO ELECTROMECÁNICO DE UN MOTOR DC.</i>	18
<i>ILUSTRACIÓN 2. DIAGRAMA DE BLOQUES DE UN PROCESO QUE UTILIZA UNA ACCIÓN DE CONTROL PROPORCIONAL.</i>	25
<i>ILUSTRACIÓN 3. DIAGRAMA DE BLOQUES DE UN PROCESO QUE UTILIZA UNA ACCIÓN DE CONTROL PROPORCIONAL (CONTROL P) JUNTO CON UN BIAS.</i>	26
<i>ILUSTRACIÓN 4. GRAFICA DE ERROR VS. TIEMPO (E vs. T).</i>	28
<i>ILUSTRACIÓN 5. ECUACIÓN DE UN CONTROLADOR PID.</i>	30
<i>ILUSTRACIÓN 6. DIAGRAMA DE BLOQUES DE UN SISTEMA DE CONTROL EN LAZO CERRADO CON CONTROL PID.</i>	30
<i>ILUSTRACIÓN 7. CONSTRUCCIÓN BÁSICA DE UN ENCODER ROTATIVO INCREMENTAL CON TECNOLOGÍA ÓPTICA.</i>	32
<i>ILUSTRACIÓN 8. DIAGRAMA DE PINES DEL ARDUINO MEGA 2560.</i>	33
<i>ILUSTRACIÓN 9. TIPO DE RESPUESTA OTORGADA POR UN SISTEMA DE PRIMER O SEGUNDO ORDEN.</i>	46
<i>ILUSTRACIÓN 10. CURVA DE REACCIÓN DEL MOTOR DC.</i>	47
<i>ILUSTRACIÓN 11. GRAFICA DE UN SISTEMA DE PRIMER ORDEN CON RETARDO.</i>	48
<i>ILUSTRACIÓN 12. MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS.</i>	50
<i>ILUSTRACIÓN 13. GRAFICA DEL MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS.</i>	51
<i>ILUSTRACIÓN 14. MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL PROPORCIONAL.</i>	51
<i>ILUSTRACIÓN 15. GRAFICA DEL MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL PROPORCIONAL.</i>	52
<i>ILUSTRACIÓN 16. MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL.</i>	52
<i>ILUSTRACIÓN 17. GRAFICA DEL MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL.</i>	53
<i>ILUSTRACIÓN 18. MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL DERIVATIVO.</i>	53
<i>ILUSTRACIÓN 19. GRAFICA DEL MODELO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL DERIVATIVO.</i>	54
<i>ILUSTRACIÓN 20. DIAGRAMA DE BLOQUES DEL MODELO MATEMÁTICO DE LA PLANTA.</i>	55
<i>ILUSTRACIÓN 21. GRAFICA DEL DIAGRAMA DE BLOQUES DEL MODELO MATEMÁTICO DE LA PLANTA.</i>	55
<i>ILUSTRACIÓN 22. DIAGRAMA DE BLOQUES DEL MODELO MATEMÁTICO DE LA PLANTA IMPLEMENTANDO EL CONTROL PROPORCIONAL.</i>	56
<i>ILUSTRACIÓN 23. PARÁMETROS DEL CONTROLADOR PROPORCIONAL OTORGADOS POR EL PID TUNER.</i>	56
<i>ILUSTRACIÓN 24. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR PROPORCIONAL.</i>	57
<i>ILUSTRACIÓN 25. DIAGRAMA DE BLOQUES DEL MODELO MATEMÁTICO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL.</i>	57
<i>ILUSTRACIÓN 26. PARÁMETROS DEL CONTROLADOR INTEGRAL OTORGADOS POR EL PID TUNER.</i>	58
<i>ILUSTRACIÓN 27. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR INTEGRAL.</i>	58
<i>ILUSTRACIÓN 28. DIAGRAMA DE BLOQUES DEL MODELO MATEMÁTICO DE LA PLANTA IMPLEMENTANDO EL CONTROL INTEGRAL DERIVATIVO.</i>	59
<i>ILUSTRACIÓN 29. PARÁMETROS DEL CONTROLADOR INTEGRAL DERIVATIVO OTORGADOS POR EL PID TUNER.</i>	59
<i>ILUSTRACIÓN 30. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR INTEGRAL DERIVATIVO.</i>	60
<i>ILUSTRACIÓN 31. CÓDIGO IMPLEMENTADO EN MATLAB PARA LA OBTENCIÓN DE LA TRANSFORMADA Z DEL SISTEMA.</i>	60
<i>ILUSTRACIÓN 32. RESPUESTA GRAFICA DE LA DISCRETIZACIÓN DE LA PLANTA.</i>	61
<i>ILUSTRACIÓN 33. MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS DISCRETIZADA.</i>	62
<i>ILUSTRACIÓN 34. GRAFICA DEL MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS DISCRETIZADA.</i>	62
<i>ILUSTRACIÓN 35. MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS DISCRETIZADA IMPLEMENTANDO EL CONTROL PROPORCIONAL.</i>	63
<i>ILUSTRACIÓN 36. PARÁMETROS DEL CONTROLADOR PROPORCIONAL OTORGADOS POR EL PID TUNER.</i>	63
<i>ILUSTRACIÓN 37. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR PROPORCIONAL.</i>	64
<i>ILUSTRACIÓN 38. MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS DISCRETIZADA IMPLEMENTANDO EL CONTROL INTEGRAL.</i>	64
<i>ILUSTRACIÓN 39. PARÁMETROS DEL CONTROLADOR INTEGRAL OTORGADOS POR EL PID TUNER.</i>	65
<i>ILUSTRACIÓN 40. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR INTEGRAL.</i>	65
<i>ILUSTRACIÓN 41. MODELO DE LA PLANTA POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS DISCRETIZADA IMPLEMENTANDO EL CONTROL INTEGRAL DERIVATIVO.</i>	66
<i>ILUSTRACIÓN 42. PARÁMETROS DEL CONTROLADOR INTEGRAL DERIVATIVO OTORGADOS POR EL PID TUNER.</i>	66
<i>ILUSTRACIÓN 43. GRAFICA DE RESPUESTA DEL PID TUNER PARA EL CONTROLADOR INTEGRAL DERIVATIVO.</i>	67
<i>ILUSTRACIÓN 44. DIAGRAMA DE FLUJO DEL CÓDIGO IMPLEMENTADO EN ARDUINO PARA EL CONTROLADOR PROPORCIONAL.</i>	68

<i>ILUSTRACIÓN 45. DIAGRAMA DE FLUJO DEL CÓDIGO IMPLEMENTADO EN ARDUINO PARA EL CONTROLADOR INTEGRAL ...</i>	69
<i>ILUSTRACIÓN 46. DIAGRAMA DE FLUJO DEL CÓDIGO IMPLEMENTADO EN ARDUINO PARA EL CONTROLADOR INTEGRAL DERIVATIVO.</i>	69
<i>ILUSTRACIÓN 47. PANTALLA DE PROGRAMACIÓN EN LABVIEW.....</i>	71
<i>ILUSTRACIÓN 48. INTERFAZ GRÁFICA DISEÑADA EN LABVIEW.....</i>	71
<i>ILUSTRACIÓN 49. RESPUESTA DEL SOFTWARE DE ARDUINO RESPECTO AL CONTROLADOR PROPORCIONAL POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS.</i>	73
<i>ILUSTRACIÓN 50. RESPUESTA DEL SOFTWARE DE ARDUINO RESPECTO AL CONTROLADOR PROPORCIONAL POR MEDIO DEL MODELO MATEMÁTICO.</i>	74
<i>ILUSTRACIÓN 51. RESPUESTA DEL SOFTWARE DE ARDUINO RESPECTO A CONTROLADOR INTEGRAL POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS.</i>	75
<i>ILUSTRACIÓN 52. RESPUESTA DEL SOFTWARE DE ARDUINO RESPECTO AL CONTROLADOR INTEGRAL POR MEDIO DEL MODELO MATEMÁTICO.</i>	76
<i>ILUSTRACIÓN 53. RESPUESTA DEL SOFTWARE DE ARDUINO RESPECTO AL CONTROLADOR INTEGRAL DERIVATIVO POR EL MÉTODO 1 DE ZIEGLER Y NICHOLS.</i>	77
<i>ILUSTRACIÓN 54. RESPUESTA DEL SOFTWARE ARDUINO RESPECTO AL CONTROLADOR INTEGRAL DERIVATIVO POR MEDIO DEL MODELO MATEMÁTICO</i>	78

LISTA DE ANEXOS

ANEXO A. <i>DIAGRAMA DE CONEXIONES IMPLEMENTADO PARA LA OBTENCIÓN DE LOS PARÁMETROS REQUERIDOS PARA DETERMINAR LA CONSTANTE CONTRA ELECTROMOTRIZ (K_A)</i>	87
ANEXO B. <i>CIRCUITO IMPLEMENTADO PARA HALLAR LA VARIABLE T_M DEL MOTOR DC</i>	87
ANEXO C. <i>RESPUESTA OBTENIDA DE LA SEÑAL PWM DE 9,99V MEDIANTE EL CIRCUITO IMPLEMENTADO EN EL ANEXO B</i>	88
ANEXO D. <i>RESPUESTA OBTENIDA DE LA SEÑAL PWM DE 5,19 V MEDIANTE EL CIRCUITO IMPLEMENTADO EN EL ANEXO B</i>	88
ANEXO E. <i>DIAGRAMA DE CONEXIONES PARA OBTENCIÓN Y DETERMINACIÓN DEL COEFICIENTE DE FRICCIÓN (B)</i>	89
ANEXO F. <i>DIAGRAMA DE CONEXIONES IMPLEMENTADO PARA LA OBTENCIÓN DE LOS PARÁMETROS REQUERIDOS PARA DETERMINAR EL TORQUE DE FRICCIÓN</i>	89

Introducción

La problemática a la cual se le pretende dar solución en este trabajo de investigación es cómo aplicar las distintas técnicas de control convencional (control P, PI, PID), cuando es pertinente aplicarlas e inquirir una solución para las respectivas desventajas que tendría cada técnica de control convencional.

Se efectuarán diferentes maneras de la parametrización de un motor DC, no solo la manera convencional, la cual es por medio de un encoder usando la velocidad angular del eje del motor.

Se realizarán programas basados en Arduino que implementarán las distintas técnicas de control convencional (control P, PI, PD, PID), en los programas ya mencionados se explicará el proceso paso a paso con el fin de entregar una mayor comprensión de los mismos.

A cada técnica de control convencional que se implemente en el proceso del motor DC se va llevar a cabo una sintonización para poder obtener los valores adecuados, los cuales serán aplicados a la técnica de control implementada en el proceso.

Por último, se llevará a cabo la respectiva validación de las distintas técnicas de control convencionales (P, PI, PID), con el fin de realizar una comparación de los resultados obtenidos en el laboratorio con los resultados teóricos adquiridos en el modelo matemático.

1. Planteamiento del Problema

La caracterización del comportamiento ante diferentes técnicas de control convencional de las máquinas eléctricas en la formación del ingeniero electromecánico es fundamental, actualmente en el laboratorio de máquinas eléctricas de la Universidad Francisco de Paula Santander no se cuenta con un módulo que permita validar el funcionamiento de los motores de corrientes continua ante sistemas de control proporcional, proporcional integral y proporcional integral derivativo.

La construcción de un módulo didáctico en donde el estudiante pueda realizar prácticas de laboratorio que le permitan implementar diferentes estrategias de control, lo llevarán a adquirir destrezas en predecir el comportamiento de la velocidad del motor en régimen transitorio y régimen permanente.

El comportamiento que se predice desde el modelo teórico y a nivel de simulación en muchas ocasiones difiere de las características reales en cuanto a funcionamiento de los diferentes procesos en estudio y mediante el desarrollo de una serie de prácticas de laboratorio en el área de control de máquinas eléctricas permitirán validar y definir esas brechas existentes entre lo teórico y lo práctico.

¿Qué estrategia de control convencional de motores DC presenta un mejor comportamiento en régimen transitorio y permanente respecto al control de velocidad?

2. Motivación y Justificación

En la industria uno de los principales problemas son los de validación de modelos de procesos y sintonización de controladores, en este caso se trabajará con las técnicas de validación de control convencional en un motor DC, esta investigación permitirá a los estudiantes de ingeniería electromecánica de la universidad Francisco de Paula Santander tener un sistema en el cual puedan observar el comportamiento dinámico de la velocidad del motor ante diferentes técnicas de control convencional (control P, PI, PID).

Por otra parte, se describen diferentes técnicas que permiten obtener el modelo matemático del motor DC, desde el punto de vista circuital y por medio de prueba de motor lanzado, describiendo las diferentes técnicas que se utilizarán para la determinación de cada uno de los parámetros.

3. Objetivos

3.1 Objetivo General

Validar las técnicas de control convencionales (control P.PI.PID) usando como proceso un motor DC.

3.2 Objetivos Específicos

Obtener el modelo matemático de un motor DC utilizando diferentes técnicas.

Determinar las constantes del controlador utilizando diferentes técnicas de sintonización teniendo como variable controlada la velocidad del motor.

Desarrollar un software basado en Arduino que permita implementar las distintas técnicas de control convencionales en un motor DC.

Desarrollar una interfaz gráfica para la supervisión del comportamiento de la velocidad del motor

Validar las técnicas de control convencionales (P, PI, PID) y comparar los resultados obtenidos en experimentación en laboratorio con los resultados obtenidos teóricamente para la selección partiendo de parámetros de diseño.

4. Alcances

Es una investigación descriptiva la “validación de técnicas de control convencionales (P,PI, PID) usando como proceso a controlar las variables de un motor DC” porque se buscara especificar las propiedades y las características de cada técnica de control convencional (como aplicar esa técnica de control convencional correctamente, los posibles problemas que puedan ocurrir al momento de aplicar esa técnica de control convencional y como solucionar esos problemas) también se buscará como hacer la correcta sintonización del controlador dependiendo de cada técnica de control convencional aplicada

La sintonización es “sintonizar un sistema de control realimentado significa regular parámetros en el controlador para lograr implementar un control robusto en el proceso”

5. Limitaciones y Delimitaciones

5.1 Limitaciones

Una limitación en la realización de este proyecto es que los resultados obtenidos dependerán de los equipos que se usarán para el desarrollo de la investigación, en este caso serán los que se encuentran en los laboratorios generales de la Universidad Francisco de Paula Santander.

Las pruebas se realizarán con un motor DC con encoder y la estrategia que se presentará para la obtención del modelo estar sujeta a que las características del sensor de velocidad sean las mismas.

También se tendrán como limitación las horas para utilizar el espacio de trabajo el cual se encuentra en los laboratorios de electricidad y electrónica de la universidad francisco de paula Santander debido a los horarios laborales ya asignados en cada aula.

5.2 Delimitaciones

Este proyecto se pretende ejecutar durante el primer semestre y segundo semestre del año 2023 a partir de la aprobación del anteproyecto.

Este proyecto se realizará en el área geográfica de la ciudad de San José de Cúcuta, departamento Norte de Santander.

6. Marco Referencial

6.1 Antecedentes

6.1.1 *Internacional*

En el ámbito internacional I. Cerón Morales, J. O. Vázquez Espinoza y E. Aquino Díaz realizo en México en la Universidad Politécnica de Puebla una investigación denominada Modelo matemático del motor de corriente directa. El objetivo principal de esta investigación fue hallar el modelo matemático (las funciones de transferencia) y los parámetros de un motor DC de imán permanente (PMDC)

Esta investigación es de gran importancia debido a que nos proporciona, el diagrama que representa al modelo electromecánico del motor DC de imán permanente, las ecuaciones matemáticas que representan al modelo electromecánico del motor DC de imán permanente, las funciones de transferencia que representan al motor DC de imán permanente y se grafican los resultados obtenidos experimentalmente de las funciones de las funciones de transferencia (Ceron Morales , Vázquez Espinoza, & Aquino Díaz, 2017)

6.1.2 *Nacional*

En el ámbito nacional Juan Alberto Sabogal Linares y Gilberto Andrés Vargas Ortega, realizaron en Bucaramanga, Santander, Colombia en la Universidad Pontificia Bolivariana seccional Bucaramanga una investigación denominada Control automático de un motor DC mediante procesamiento de imágenes. El objetivo principal de esa investigación es diseñar un sistema de control de velocidad de un motor DC usando dispositivos electrónicos con información obtenida a través de un software procesamiento de imágenes

Esa investigación sustituye al tradicional encoder como sensor de velocidad y distancia por un disco con dos puntos en el eje del rotor, un punto central en el disco que sirve de referencia y otro punto que ira girando a medida que gira el rotor, una cámara web tomara imágenes del disco en cada momento y por medio del software Matlab instalado en el pc junto con la cámara web se procesan las imágenes para saber la velocidad angular y distancia angular que tiene el eje del motor, en ese momento el computador usara la comunicación serial RS232 para comunicarse con MAX232 que a su vez se comunicara con un microcontrolador PIC16F877A usando comunicación TTL ,el microcontrolador PIC efectuara la estrategia de control dando como resultado un control PWM que va a controlar a un CI puente H modelo L293D que a su vez controlara al motor DC (Sabogal Linares & Vargas Ortega, 2013)

Esta investigación es de gran importancia para el desarrollo de este proyecto debido a que esta investigación muestra el diagrama de conexiones del proyecto y el diagrama de bloques del proyecto

En el ámbito nacional Jhonatan Bautista Contreras y Jesús Alirio Guevara Núñez realizaron en San José de Cúcuta, Norte de Santander, Colombia en la Universidad Francisco de Paula Santander una investigación denominada Modulo didáctico para el entrenamiento del control de velocidad para motores eléctricos. El objetivo principal de esa investigación es crear un módulo didáctico para poder enseñar cómo controlar motores AC y motores DC, el proceso de implementación de ese modulo didáctico seria menos costoso que módulos didácticos con funciones similares de empresas de marca como Schneider y De Lorenzo.

El módulo didáctico de esa investigación consiste en un maletín que contiene un variador de velocidad, un potenciómetro, una fuente DC, un control PWM, una protección magneto térmica para proteger al variador de velocidad, un motor AC y un motor DC (Bautista Contreras & Guevara Lidueñez, 2021)

Esta investigación es de gran importancia para el desarrollo de este proyecto debido a que su marco teórico explica muchos conceptos importantes del control PWM y de los motores DC

6.2 Marco Teórico

6.2.1 Motor DC

Los motores de Corriente Directa o motor DC (correspondiente a las iniciales en inglés “direct current”) es también conocido como motor de Corriente Continua o motor CC, son muy utilizados en diseños de ingeniería debido a las características de torque-velocidad que poseen con diferentes configuraciones eléctricas o mecánicas

Una gran ventaja de los motores de CD se debe a que es posible controlarlos con suavidad y en la mayoría de los casos son reversibles, responden rápidamente gracias a que cuentan con una gran razón de torque a la inercia del rotor

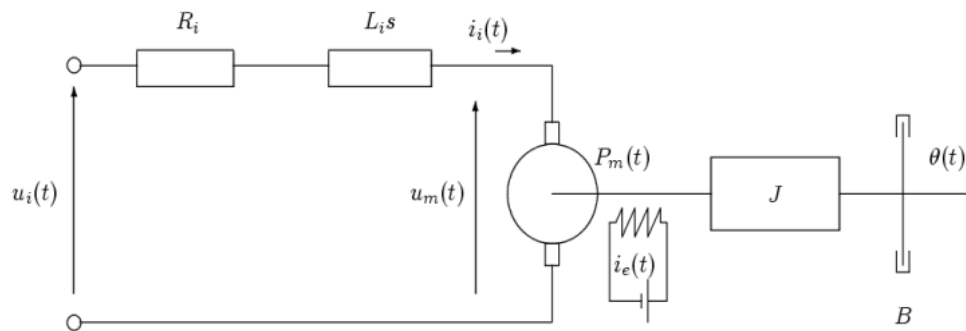
Otra ventaja es la implementación del frenado dinámico, donde la energía generada por el motor se alimenta a un resistor disipador y el frenado regenerativo donde la energía generada por el motor retroalimenta al suministro de potencia CD, esto es muy utilizado en aplicaciones donde se deseen frenados rápidos y de gran eficiencia.

(MecatrónicaLATAM, 2021)

6.2.2 Modelo matemático de un motor DC

Ilustración 1.

Modelo Electromecánico de un motor DC.



El modelo matemático de un motor DC se compone de dos partes, una parte eléctrica y una parte mecánica. (Modelado de un motor CC, s.f.) (Ceron Morales , Vázquez Espinoza, & Aquino Díaz, 2017)

Parte eléctrica

Se hace un LVK

$$-V + VR + VL + EA = 0$$

$$VL = V - VR - EA$$

$$VR = R \times I$$

$$VL = L \times \frac{dI}{dt}$$

$$L \times \frac{dI}{dt} = V - R \times I - EA$$

$$EA(t) = V(t) - R \times I(t) - L \times \frac{dI(t)}{dt} \quad (\text{Ecu. 1})$$

Donde:

L=Inductancia de la bobina de la armadura

R=Resistencia del cable conductor de la bobina de la armadura

I=Corriente de la armadura

EA=Fuerza contra electromotriz

La parte mecánica se presenta de la siguiente manera:

$$\text{Entrada} = \text{Salida}$$

$$Tm = J \times a + B \times \omega$$

$$a = \frac{d\omega}{dt}$$

$$Tm = J \times \frac{d\omega}{dt} + B \times \omega$$

$$Tm(t) = J \times \frac{d\omega(t)}{dt} + B \times \omega(t)$$

Dónde:

Tm=Torque generado por el motor

ω =Velocidad angular

Velocidad angular del eje del motor

J=Momento de inercia

Momento de inercia equivalente al motor y a la carga que se coloca sobre el motor

B =Coeficiente de fricción

Se asumen que existen dos relaciones, una entre la fuerza contra electromotriz y la velocidad angular (EA, ω) y otra entre el torque y la corriente de la armadura (T_m, I)

Se asume que existe una relación proporcional K_a , entre el voltaje de inducido en la armadura o fuerza contra electromotriz y la velocidad angular del eje del motor (EA, ω)

$$EA = K_a \times \omega$$

$$EA(t) = K_a \times \omega(t)$$

Se establece la siguiente relación electromecánica K_m que establece que el torque mecánico es proporcional a la corriente eléctrica (T_m, I_m)

$$T_m = K_m \times I$$

$$T_m(t) = K_m \times I(t)$$

Se halla la transformada de Laplace de cada una de las ecuaciones obtenidas

$$L \times \frac{dI(t)}{dt} = V(t) - R \times I(t) - EA(t)$$

$$L \times S \times I(s) = V(s) - R \times I(s) - EA(s)$$

$$T_m(t) = J \times \frac{d\omega(t)}{dt} + B \times \omega(t)$$

$$T_m(s) = J \times S \times \omega(s) + B \times \omega(s)$$

$$EA(t) = K_a \times \omega(t)$$

$$EA(s) = Ka \times \omega(s)$$

$$Tm(t) = Km \times I(t)$$

$$Tm(s) = Km \times I(s)$$

Se halla la función de transferencia de torque vs voltaje (Tm/V)

$$L \times S \times I(s) = V(s) - R \times I(s) - EA(s) \quad (\text{Ecu.2})$$

$$Tm(s) = J \times S \times \omega(s) + B \times \omega(s) \quad (\text{Ecu.3})$$

$$Tm(s) = Km \times I(s) \quad (\text{Ecu.4})$$

$$EA(s) = Ka \times \omega(s) \quad (\text{Ecu.5})$$

Se reemplaza en la ecuación (2) las ecuaciones (4) y (5)

$$L \times S \times I(s) = V(s) - R \times I(s) - EA(s)$$

$$Tm(s) = Km \times I(s)$$

$$\frac{Tm(s)}{Km} = I(s)$$

$$EA(s) = Ka \times \omega(s)$$

$$L \times S \times \frac{Tm(s)}{Km} = V(s) - R \times \frac{Tm(s)}{Km} - Ka \times \omega(s) \quad (\text{Ecu.6})$$

Se despeja ω en la ecuación (3)

$$Tm(s) = J \times S \times \omega(s) + B \times \omega(s)$$

$$Tm(s) = (J \times S + B) \times \omega(s)$$

$$\frac{Tm(s)}{(J \times S + B)} = \omega(s) \quad (\text{Ecu.7})$$

Se reemplaza la ecuación (6) en la ecuación (5)

$$L \times S \times \frac{Tm(s)}{Km} = V(s) - R \times \frac{Tm(s)}{Km} - Ka \times \omega(s)$$

$$\frac{Tm(s)}{(J \times S + B)} = \omega(s)$$

$$L \times S \times \frac{Tm(s)}{Km} = V(s) - R \times \frac{Tm(s)}{Km} - Ka \times \frac{Tm(s)}{(J \times S + B)}$$

$$\frac{L \times S \times Tm(s)}{Km} = V(s) - \frac{R \times Tm(s)}{Km} - \frac{Ka \times Tm(s)}{(J \times S + B)}$$

$$\frac{L \times S \times Tm(s)}{Km} + \frac{R \times Tm(s)}{Km} + \frac{Ka \times Tm(s)}{(J \times S + B)} = V(s)$$

$$\left[\frac{L \times S}{Km} + \frac{R}{Km} + \frac{Ka}{(J \times S + B)} \right] \times Tm(s) = V(s)$$

$$\left[\frac{(L \times S + R)}{Km} + \frac{Ka}{(J \times S + B)} \right] = Tm(s) = V(s)$$

$$\left[\frac{(L \times S + R) \times (J \times S + B) + Km \times Ka}{Km \times (J \times S + B)} \right] \times Tm(s) = V(s)$$

$$\left[\frac{L \times S \times J \times S + L \times S \times B + R \times J \times S + R \times B + Km \times Ka}{Km \times J \times S + Km \times B} \right] \times Tm(s) = V(s)$$

$$\left[\frac{L \times J \times S \times S + L \times B \times S + R \times J \times S + R \times B + Km \times Ka}{Km \times J \times S + Km \times B} \right] \times Tm(s) = V(s)$$

$$\left[\frac{L \times J \times S^2 + (L \times B + R \times J) \times S + R \times B + Km \times Ka}{Km \times (J \times S + B)} \right] \times Tm(s) = V(s)$$

$$Tm(s) = \frac{V(s)}{\left[\frac{L \times J \times S^2 + (L \times B + R \times J) \times S + R \times B + Km \times Ka}{Km \times (J \times S + B)} \right]}$$

$$\frac{Tm(s)}{V(s)} = \frac{1}{\left[\frac{L \times J \times S^2 + (L \times B + R \times J) \times S + R \times B + Km \times Ka}{Km \times (J \times S + B)} \right]}$$

$$\frac{Tm(s)}{V(s)} = \frac{\frac{1}{1}}{\left[\frac{L \times J \times S^2 + (L \times B + R \times J) \times S + R \times B + Km \times Ka}{Km \times (J \times S + B)} \right]}$$

$$\frac{Tm(s)}{V(s)} = \frac{Km \times (J \times S + B)}{L \times J \times S^2 + (L \times B + R \times J) \times S + R \times B + Km \times Ka}$$

Gracias al procedimiento anterior se obtienen las funciones de transferencia a utilizar:

Función de transferencia Torque-voltaje (Tm/V)

$$\frac{Tm(s)}{V(s)} = \frac{Km \times (J \times S + B)}{L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times Ka} \quad (Ecu.8)$$

Función de transferencia Fuerza contra electromotriz-voltaje (EA/V)

$$\frac{EA(s)}{V(s)} = \frac{Km \times Ka}{L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times Ka} \quad (Ecu.9)$$

Función de transferencia corriente-voltaje (I/V)

$$\frac{I(s)}{V(s)} = \frac{J \times S + B}{L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times Ka} \quad (Ecu.10)$$

Función de transferencia velocidad angular-voltaje (ω /V)

$$\frac{\omega(s)}{V(s)} = \frac{Km}{L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times ka} \quad (\text{Ecu.11})$$

Función de transferencia posición angular –voltaje (Θ/V)

$$\frac{\theta(s)}{V(s)} = \frac{Km}{S \times [L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times Ka]} \quad (\text{Ecu.12})$$

El numerador de la función de transferencia que representa al comportamiento del motor DC en base al modelo matemático indica que el modelo matemático obtenido del motor DC es un sistema de segundo orden

$$D(s) = L \times J \times S^2 + (R \times J + L \times B) \times S + R \times B + Km \times ka$$

6.2.3 Técnicas de control

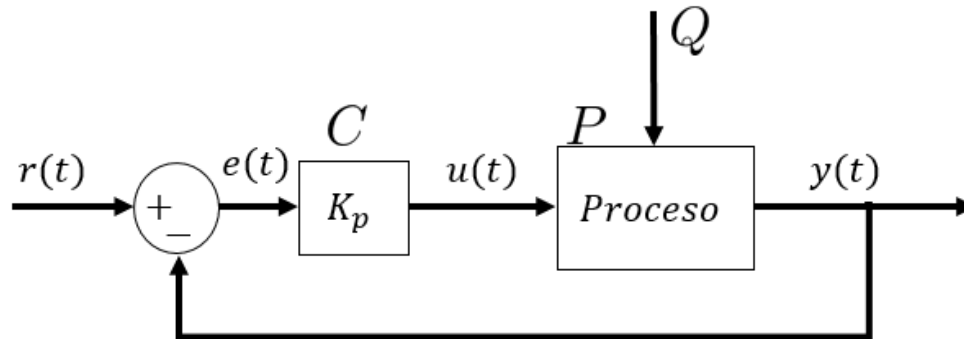
6.2.3.1 Técnicas de control convencional.

6.2.3.1.1 Acción de control proporcional.

La acción de control proporcional o control proporcional (Control P) sirve para poder hacer que el elemento final de control (efc) trabaje de una manera más suave, el control proporcional cuando el error $e(t)$ es muy grande hace que el elemento final de control trabaje más y cuando el error $e(t)$ es muy pequeño hace que el elemento final de control trabaje menos. (coggle, s.f.)

Ilustración 2.

Diagrama de bloques de un proceso que utiliza una acción de control proporcional.



$$u(t) = Kp \times (r(t) - y(t))$$

$$e(t) = r(t) - y(t)$$

El objetivo del controlador (C) es por medio de la técnica de control aplicada volver la señal de error a cero, es decir, que el valor de la variable medida del proceso $y(t)$ sea igual al valor deseado de la variable medida del proceso o Setpoint $r(t)$ ($e=0$)

En el controlador proporcional (control P) la salida del controlador $u(t)$ se define como:

$$u(t) = Kp \times e(t)$$

$$u(t) = Kp \times (r(t) - y(t))$$

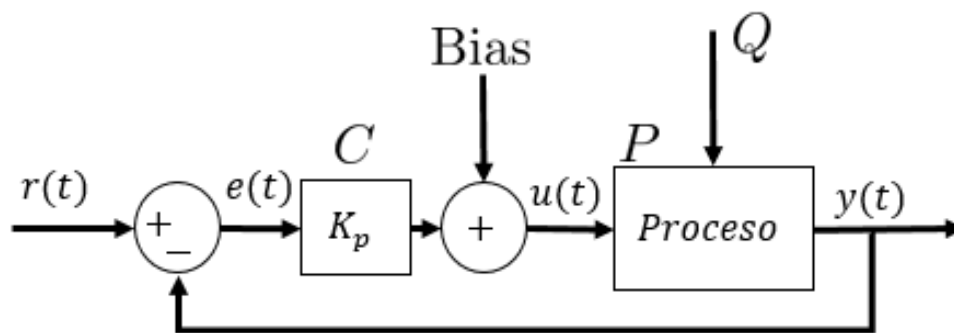
Donde Kp es la ganancia proporcional que tendrá la salida del controlador $u(t)$ y determina que tan rápido queremos que se elimine el error ($e(t)=0$), es decir, que la señal de salida del controlador $y(t)$ sea igual al valor del Setpoint $r(t)$

Error en estado estacionario

Cuando solo se usa un control proporcional (control P) siempre se va a tener un error en estado estacionario $ess(t)$, eso significa que la salida del proceso o variable controlada $y(t)$ nunca va a alcanzar al Setpoint $r(t)$, se debe agregar un Biaspoint o un Bias al proceso para poder eliminar el error ($e(t)=0$) más específicamente eliminar al error en estado estacionario ($ess(t)=0$) (Giraldo, Control PID- Acción de control proporcional, 2023)

Ilustración 3.

Diagrama de bloques de un proceso que utiliza una acción de control proporcional (Control P) junto con un Bias.



$$u(t) = [Kp \times e(t)] + Bias$$

$$e(t) = r(t) - y(t)$$

$$u(t) = [Kp \times (r(t) - y(t))] + Bias$$

Ventajas y desventajas del control proporcional

Ventajas

- Aumenta la velocidad del proceso al agregarse una ganancia proporcional

Kp

- El error en estado estacionario $e(t)$ puede eliminarse agregando un Biaspoint o Bias a la señal del controlador cuando el proceso alcance un error igual a cero ($e(t)=0$), el Biaspoint debe agregarse manualmente

Desventajas

- Si se pone una ganancia proporcional K_p demasiado alta el proceso puede tener oscilaciones, eso significa que el elemento final de control tendría cambios de conmutación súbitos (encendidos y apagados excesivamente rápidos) lo que ocasionaría un desgaste prematuro del elemento final de control
- El Biaspoint se debe agregar manualmente cuando el proceso alcance un error igual a cero ($e(t)=0$) y el Biaspoint puede variar según el nivel del Setpoint $r(t)$, se puede agregar el Biaspoint automáticamente agregando al proceso con control proporcional (control P) una acción de control integradora (control I) (Bolívar, 2023)

6.2.3.1.2 Acción de control proporcional integradora (PI).

En la acción de control proporcional (control P) para poder eliminar el error en estado estacionario $e(t)$ del proceso, se debe adicionar manualmente un Biaspoint cuando el error en el proceso sea cero ($e(t)=0$) y el valor del Biaspoint varía según el valor del Setpoint del proceso

La acción de control integradora (control I) ajusta el valor del Biaspoint automáticamente

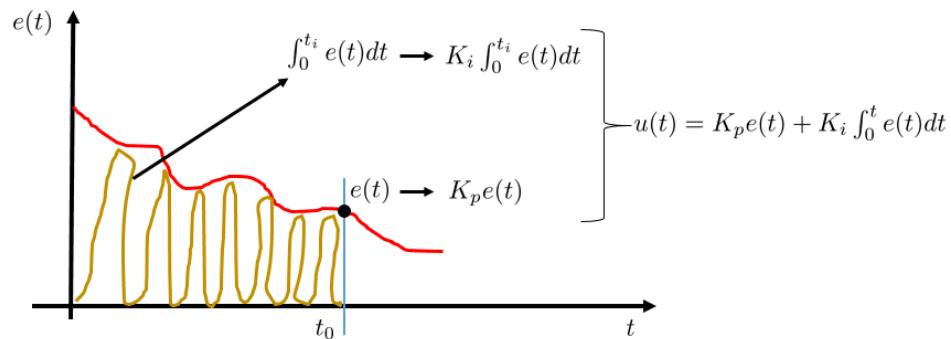
$u(t) = \text{acción de control proporcional} + \text{acción de control integral}$

$$u(t) = K_p \times e(t) + K_i \int_0^t e(t) \times dt$$

Ilustración 4.

Gráfica de Error vs. Tiempo (E vs. T).

Control PI – Proporcional, Integral



En la gráfica se explica el funcionamiento de un proceso con un control proporcional y un control integral (control PI)

La acción integradora (control I) toma los datos del error pasado $e(t)$ y los va integrando en el tiempo, es decir, la acción integradora va acumulando al valor del error pasado

Cada vez que se integra, la salida del proceso $y(t)$ se va aproximando al Setpoint $r(t)$, dependiendo del proceso esto puede tardar poco tiempo o mucho tiempo, se adiciona al control integral (control I) una ganancia integrativa K_i para acelerar o atenuar la respuesta del control integrativo

$$u(t) = K_i \times \int_0^t e(t) dt$$

$u(t)$ =salida del controlador o variable manipulada o acción de control del controlador

K_i =Ganancia integrativa

$$e(t) = \text{Error } (e(t) = y(t) - r(t))$$

El control integral (control I) va a estar rampeando (hacer que tenga forma de señal rampa) a la señal de salida $y(t)$ y no se detendrá el control integral hasta que el error sea de cero ($e(t)=0$) (que la señal de salida $y(t)$ alcance al Setpoint $r(t)$) incluyendo al error en estado estacionario ($ess=0$). (Giraldo, Acción de Control Integral , 2023)

Ventajas y desventajas de control proporcional integral

Ventajas

- Eliminar el error en estado estacionario
- Mejorar el desempeño en el régimen del estado estacionario
- Mejorar la estabilidad relativa o marginal
- Reducir el ancho de banda

Desventajas

- El costo por $ess = 0$ radica en un efecto negativo sobre la velocidad de respuesta, reduciéndola y comprometiendo la estabilidad del sistema.
- Filtrar las componentes de alta frecuencia, pero no mejora el desempeño en régimen transitorio. (Bolívar, 2023)

6.2.3.1.3 Acción de control proporcional integradora derivativa (PID).

Su funcionamiento es lo contrario a los controladores PI, ya que reduce el sobre impulso y el tiempo de estabilización, pero tiende a aumentar la estabilidad del sistema mejorando la respuesta del mismo.

Ilustración 5.

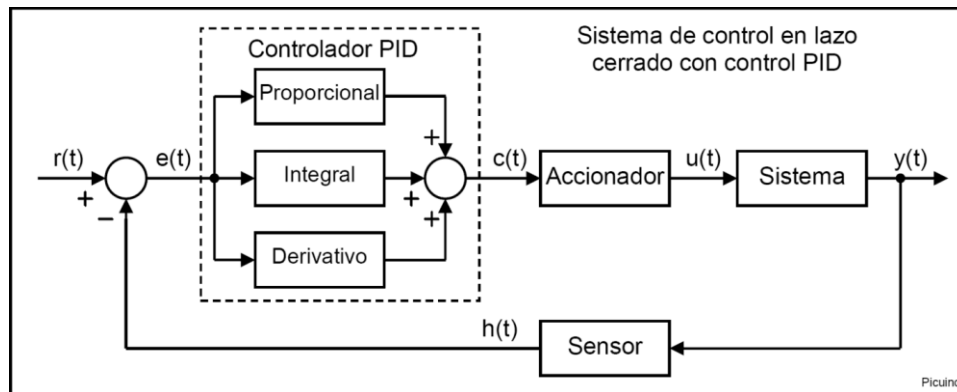
Ecuación de un controlador PID.

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt}$$

Donde K_p es la ganancia proporcional, T_i es el tiempo integral y T_d es el tiempo derivativo.

Ilustración 6.

Diagrama de bloques de un sistema de control en lazo cerrado con control PID.



Resumiendo, se puede decir que el control proporcional actúa sobre el tamaño del error, el control integral rige el tiempo para corregir el error y el control derivativo le brinda la rapidez a la actuación. (Giraldo, Acción de Control Derivativo, 2023)

Ventajas y desventajas de los controladores PID

Ventajas

- Rapidez ante el cambio de la señal de referencia en la entrada (por la componente derivativa).

- Incremento en la señal de entrada para que el error sea cero (por la componente integrativa).
- La reducción en el tiempo de establecimiento (por la componente proporcional).

Desventajas

- El termino derivativo podría amplificar cualquier ruido presente en la variable controlada. (Contaval, 2016)

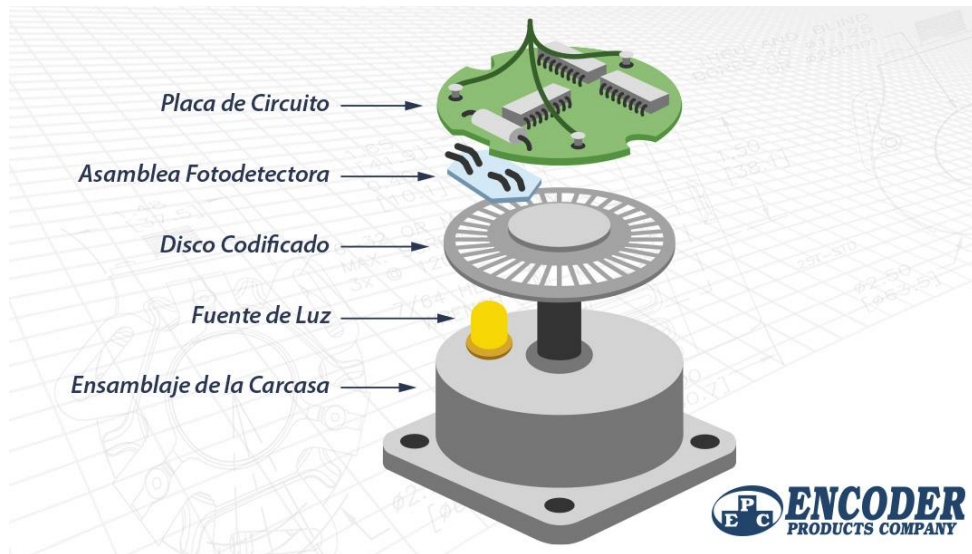
6.2.4 Encoder.

Un Encoder, también denominado generador de pulsos, es un elemento electromecánico, es decir relaciona la energía eléctrica con la mecánica, que transforma el movimiento, es decir energía mecánica, en una señal eléctrica, que pueden ser pulsos, ondas, entre otras. Esta señal eléctrica puede ser leída por algún tipo de dispositivo con el fin de controlar este movimiento.

Como mencionamos anteriormente, los Encoder son dispositivos utilizados para transformar movimiento en señales eléctricas. Esto hace que sean muy utilizados a la hora de tener que controlar movimientos de máquinas; debido a que el encoder envía información sobre el movimiento del dispositivo a controlar, un PLC (controlador) analiza esa señal y en función del análisis envía una señal a la máquina para que realice cierta función. (R, 2021) (Encoder Products Company, s.f.)

Ilustración 7.

Construcción básica de un Encoder rotativo incremental con tecnología óptica.



6.2.5 Arduino MEGA.

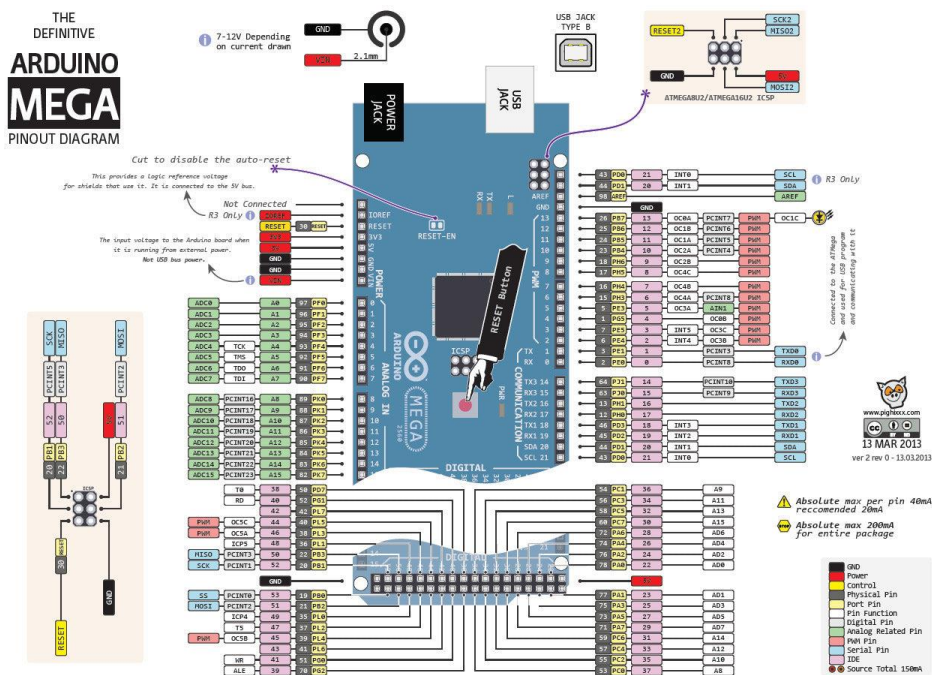
Arduino Mega es una tarjeta de desarrollo open-source construida con un microcontrolador modelo Atmega2560 que posee pines de entradas y salidas (E/S), analógicas y digitales. Esta tarjeta es programada en un entorno de desarrollo que implementa el lenguaje Processing/Wiring. Arduino puede utilizarse en el desarrollo de objetos interactivos autónomos o puede comunicarse a un PC a través del puerto serial (conversión con USB) utilizando lenguajes como Flash, Processing, MaxMSP, etc.

El Arduino Mega tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16MHz, conexión USB, jack de alimentación, conector ICSP y botón de reset. Arduino Mega incorpora todo lo necesario para que el microcontrolador trabaje; simplemente conéctalo a tu PC por medio de un cable USB o con una fuente de

alimentación externa (9 hasta 12VDC). El Arduino Mega es compatible con la mayoría de los shields diseñados para Arduino Duemilanove, diecimila o UNO. (MCI Electronics , s.f.) (pighixx, 2013)

Ilustración 8.

Diagrama de pines del Arduino MEGA 2560.



6.3 Marco Legal

Tabla 1.

Normas aplicadas a motores Dc.

Norma	Código	Descripción
ISO	2372/2373	Vibración mecánica de máquinas con velocidades de operación entre 10 y 200 rev/s. La ISO 2373 constituye una adaptación especial de la ISO 2372 para motores eléctricos, y se aplica a motores de corriente alterna trifásica y a motores de corriente continua con alturas de eje entre 80 y 400 mm.
ISO	3945	Medida y evaluación de la severidad de vibración en grandes máquinas rotativas, in situ con velocidades de operación entre 10 y 200 rev/s.
ISO	10816	Vibración mecánica. - Evaluación de la vibración en una máquina mediante medidas en partes no rotativas. Reemplaza la ISO 2373 e ISO 3945.
ISO	7919	Vibración mecánica de máquinas no alternativas - Medidas en ejes rotativos y evaluación.
VDI	2056	Evaluación de vibraciones mecánicas de maquinaria rotativa. Basada en la ISO 2372. Reemplazada por la ISO 10816.
ANSI	2526	Vibración mecánica. - Evaluación de la vibración en una máquina mediante medidas en partes no rotativas. Este estándar es idéntico y ha sido reproducido a partir de la ISO 10816-1:1995.
IEC	34-14:1997	Vibraciones mecánicas de determinadas máquinas con altura de eje igual o superior a 56 mm. Medición, evaluación y límites de la intensidad de vibración.
AENOR	UNE EN 60034-14:1997	Vibraciones mecánicas de determinadas máquinas con altura de eje igual o superior a 56 mm. Medición, evaluación y límites de la intensidad de vibración. Basada en la IEC 34-14, acaba de salir la revisión del 2004, denominada por AENOR UNE EN 60034-14:2004 y correspondiente a la norma IEC 60034-14:2003
BSI	7823	Vibración mecánica. - Evaluación de la vibración en una máquina mediante medidas en partes no rotativas. Basada en la ISO 10816-1:1995.
API	670	Monitorización en tiempo real y la protección de maquinaria crítica.

(Monte, 2023) (Torres Niño & Rivas Rengifo, 1996)

7. Diseño Metodológico

A continuación, se describe la metodología a utilizar en el proyecto para alcanzar cada uno de los objetivos estipulados:

7.1 Actividades y Metodologías

Objetivo 1. Obtener el modelo matemático de un motor DC utilizando diferentes técnicas.

Actividades.

- Hallar los valores de los parámetros que conforman el modelo matemático del motor DC (R, L, J, B, Km, Ka)

Metodología

Se interpreta el modelo matemático que representa al motor DC, a partir de ese modelo matemático se hallaron las funciones de transferencia que representan al comportamiento de las diversas variables de salida

- Recopilar información referente al tema de estudio

Metodología

Se realizó la investigación referente a las diferentes técnicas utilizadas para la parametrización empleadas en un motor DC.

- Mediante prueba de motor lanzado determinar un modelo aproximado del motor DC.

Metodología

se energizó el motor con el 70% de su voltaje nominal y posteriormente se aumentó al 100% su voltaje nominal, mediante la curva de reacción se obtuvieron los parámetros aproximados.

Objetivo 2. Determinar las constantes del controlador utilizando diferentes técnicas de sintonización teniendo como variable controlada la velocidad del motor.

Actividades.

- Implementar en simulink la función de transferencia del motor y el bloque del controlador para obtener las diferentes constantes de las estrategias de control tradicional.

Metodología

Utilizaron la técnica de Nichols y Ziegler como primera estrategia de sintonización para obtener las constantes, mediante el tuner de Matlab se sintonizó el controlador para que cumpla parámetros específicos de la respuesta transitoria.

- Validar mediante simulación el comportamiento de los controladores a diseñar.

Metodología

Ajustaron los parámetros de las constantes en el bloque PID en simulink y verificaron los resultados obtenidos con respecto a los parámetros preestablecidos.

Objetivo 3. Desarrollar un software basado en Arduino que nos permita implementar las distintas técnicas de control convencionales en un motor DC

Actividades.

- Mediante un diagrama de flujo se representará la lógica de programación para la implementación de las estrategias de control.

Metodología

En base a los pasos a seguir para la implementación en un sistema de control tradicional en un dispositivo programable se desarrolló el diagrama de flujo de señal a utilizar.

- Traducir a código Arduino el diagrama de flujo realizado anteriormente.

Metodología

A partir del diagrama de flujo de señal se desarrolló un algoritmo de programación y posterior a ello se verificó la operatividad del mismo.

Objetivo 4. Desarrollar una interfaz gráfica para la supervisión del comportamiento de la velocidad del motor.

Actividades

- Mediante LabVIEW se supervisará la velocidad que el dispositivo programable transmitirá mediante el protocolo de comunicaciones al ordenador.

Metodología

Determinaron el protocolo de comunicaciones a utilizar, partiendo del mismo se implementó en el dispositivo programable la transmisión y en el ordenador la recepción de los datos.

Objetivo 5. Validar las distintas técnicas de control convencionales (P, PI, PID), comparar los resultados obtenidos mediante experimentación en laboratorio con los resultados obtenidos teóricamente.

Actividades.

- Implementar el algoritmo de programación para cada una de las estrategias de control, seguidamente verificar el comportamiento de la velocidad del motor en cada una de ellas.

Metodología

Implementaron una a una, cada una de las estrategias de control y mediante la interfaz HMI se analizó la dinámica del comportamiento de la velocidad del motor.

8. Desarrollo de la Investigación

8.1 Modelo Matemático de un Motor DC

Para hallar el modelo matemático del motor DC usado en el presente proyecto, al motor DC se le realizaron las siguientes pruebas:

Se realizó el despeje de la ecuación 4 para hallar la variable K_a la cual es una de las constantes que relaciona a la parte eléctrica con la parte mecánica del motor DC.

$$\frac{EA}{\omega} = K_a \quad (\text{Ecu.13})$$

Posterior a ello se realizó el despeje de la ecuación (1) para hallar el valor de la fuerza electromotriz (EA) la cual se calcula cuando el motor DC este en estado estable o régimen permanente y el valor del voltaje del inductor V_L cambia con el tiempo, esto significa que:

$$EA = V - R \times I \quad (\text{Ecu.14})$$

Se reemplazo la ecuación 13 en la ecuación 14 para despejar la variable K_a

$$\frac{V-R \times I}{\omega} = K_a \quad (\text{Ecu.15})$$

Tabla 2.

Resultados obtenidos para hallar la constante electromotriz.

V	I	EA=V- R*I	ω (rad/s)	RPM enconder	RPM medidor	$K_a=EA/\omega$
4,16	0,11	3,456	12,1967	116,47	116,66	0,2833
8,14	0,10	7,5	24,8489	237,29	237,30	0,3018
9,50	0,11	8,796	29,1560	278,42	278,47	0,3016
10,14	0,10	9,5	31,2598	298,51	298,67	0,3039
10,51	0,10	9,87	32,4819	310,18	310,34	0,3038
10,72	0,11	10,016	33,1218	316,29	316,44	0,3023

Se realizo la conversión de los valores obtenidos del programa realizado en Arduino para leer la velocidad del eje del motor DC en RPM a valores dados en ω (rad/s)

$$116,47 \frac{vueltas}{min\ u\ to} \times \frac{2\pi\ rad}{1\ vueltas} \times \frac{1\ minuto}{60\ segundos} = 12,1967 \frac{rad}{s}$$

$$\omega = 12,1967 \frac{rad}{s}$$

El procedimiento anterior fue reiterado en cada una de las muestras tomadas en el programa de Arduino.

en la ecuación 14 se reemplazaron los valores de V, I, ω proporcionado por la tabla 2. Resultados obtenidos para hallar la constante electromotriz, en el cual se tuvo en cuenta un valor de $R=6,4\Omega$, para así obtener los resultados de la fuerza contra electromotriz (EA)

por ejemplo:

$$EA = V - R \times I$$

$$EA = 4,16 - 6,4 \times 0,11$$

$$EA = 4,16 - 0,704$$

$$EA = 3,456v$$

Los resultados de la fuerza contra electromotriz (EA) se encuentran visualizados en la tabla 2. Resultados obtenidos para hallar la constante electromotriz.

en la ecuación 13 se reemplazaron los valores de EA y ω proporcionado por la tabla 2. Resultados obtenidos para hallar la constante electromotriz, para así obtener los resultados de la constante contra electromotriz (Ka)

por ejemplo:

$$Ka = \frac{EA}{\omega}$$

$$Ka = \frac{3,17}{12,1967}$$

$$Ka = 0,2833 \frac{v}{rad \times s}$$

Los resultados de la constante contra electromotriz (Ka) se encuentran visualizados en la tabla 2. Resultados obtenidos para hallar la constante electromotriz.

Se promedió el valor obtenido de constante contra electromotriz (Ka), para así obtener un valor único de Ka

$$\overline{Ka} = \frac{\sum Ka}{\#Ka}$$

$$\overline{Ka} = \frac{0,2833 + 0,3018 + 0,3016 + 0,3039 + 0,3038 + 0,3023}{6}$$

$$\overline{Ka} = \frac{1,7967}{6}$$

$$\overline{Ka} = 0,29945$$

$$Ka = 0,29945$$

$$Ka = 0,29945 \frac{v}{rad \times s}$$

Se halló el valor de Km el cual es la constante de torque, usaron la técnica paramétrica dimensional que consiste en usar expresiones que guardan una relación

paramétrica dimensional entre la constante de torque (K_m) y la constante contra electromotriz (K_a) cuya comprobación empírica ya ha sido sustentada.

$$K_m = K_a = 0,29945$$

$$K_m = 0,29945 \frac{N \times m}{A}$$

Tabla 3.

Resultados obtenidos para hallar el coeficiente de fricción B.

V	I	$T_m = K_m \cdot I$	ω	$B = (T_m - T_f) / \omega$	RPM
4,13	0,11	0,03293	12,993	0,000460247	117,45
6,75	0,09	0,02695	20,5983	0	196,70
8,19	0,11	0,03293	25,2374	0,000236949	241
8,96	0,11	0,03293	27,7350	0,000215612	264,85
9,51	0,13	0,03892	29,4262	0,00040678	281
9,69	0,11	0,03293	30,2116	0,000197937	288,50
10,13	0,11	0,03293	31,5709	0,000188464	301,48
10,32	0,11	0,03293	32,1175	0,000186191	306,70
10,48	0,12	0,03593	32,6202	0,000275289	311,50
10,70	0,11	0,03293	33,3396	0,000179366	318,37

Se realizó la conversión de los valores obtenidos del programa realizado en Arduino para leer la velocidad del eje del motor DC en RPM a valores dados en ω (rad/s)

$$117,45 \frac{\text{vueltas}}{\text{minuto}} \times \frac{2\pi \text{ rad}}{1 \text{ vueltas}} \times \frac{1 \text{ minuto}}{60 \text{ segundos}} = 12,993 \frac{\text{rad}}{\text{s}}$$

$$\omega = 12,993 \frac{\text{rad}}{\text{s}}$$

El procedimiento anterior fue reiterado en cada una de las muestras tomadas en el programa de Arduino.

Se halló la variable t_m , la cual es la constante de tiempo mecánica del motor DC

En la ecuación 4 se reemplazaron los valores de V , I y K_m entregados por la tabla 3. Resultados obtenidos para hallar el coeficiente de fricción B , para así obtener los resultados de la constante de tiempo mecánica (T_m).

Por ejemplo:

$$T_m = K_m \times I$$

$$T_m = 0,29945 \times 0,11$$

$$T_m = 0,03293N \times m$$

Los valores de la constante de tiempo mecánica se encuentran visualizados en la tabla

3. Resultados obtenidos para hallar el coeficiente de fricción B .

Se halló el torque de fricción (T_f) el cual es el torque que se opone al movimiento del motor, el motor debe tener un torque (T_m) superior al torque de fricción para que el eje del motor pueda empezar a moverse

$$T_f = K_m \times I_{arr}$$

(Ecu.16)

Al motor DC se aplicó un voltaje muy pequeño que se incrementa gradualmente poco a poco y se detuvo cuando el eje del motor empezó a moverse.

Se aumentó el voltaje del motor DC por medio del PWM, en pasos de 5 en 5 hasta que el eje del motor DC empezó a moverse y obtuvieron los siguientes resultados.

$$PWM = 35$$

$$V = 1,65$$

$$I=0.09$$

$$\text{RPM}=43,533$$

Se reemplazaron los valores que obtuvieron anteriormente en la ecuación 16 y así hallaron la constante de toque de fricción del motor.

$$Tf = Km \times Iarr$$

$$Tf = 0,29945 \times 0,09$$

$$Tf = 0,02695 \text{ N} \times \text{m}$$

Se halló el coeficiente de fricción (B), el cual se calcula cuando el motor DC este en estado estable o régimen permanente y el valor de la aceleración angular (a) cambia con el tiempo, esto significa que, se enciende el motor a diferentes niveles de voltaje para poder medir la corriente de la armadura (I) y la velocidad del eje del motor (ω):

$$J \times \frac{d\omega}{dt} = 0$$

$$Tm - Tf = 0 + B \times \omega$$

$$Tm - Tf = B \times \omega$$

$$\frac{Tm - Tf}{\omega} = B \tag{Ecu.17}$$

En la ecuación 17 se reemplazaron los valores de V, Tm, Tf y ω entregados por la tabla 3. Resultados obtenidos para hallar el coeficiente de fricción B, para así obtener los resultados del coeficiente de fricción (B).

Por ejemplo:

$$B = \frac{Tm - Tf}{\omega}$$

$$B = \frac{0,03293 - 0,02695}{12,993}$$

$$B = 0,000460247 \text{ N} \times \text{m} \times \text{s}$$

Los valores del coeficiente de fricción (B) se encuentran visualizados en la tabla 3.

Resultados obtenidos para hallar el coeficiente de fricción B.

Promediaron los valores obtenidos del coeficiente de fricción (B) para así obtener un valor único de B

$$\bar{B} = \frac{\sum B}{\#B}$$

$$\bar{B} = \frac{0,000460247 + 0 + 0,000236949 + 0,000215612 + 0,00040678 + 0,000197937 + 0,000188464 + 0,000186191 + 0,000275289 + 0,000179366}{10}$$

$$\bar{B} = \frac{0,002346835}{10}$$

$$\bar{B} = 0,000234683 \text{ N} \times \text{m} \times \text{s}$$

Nota: los diagramas de conexiones utilizados para la parametrización y sus respectivas respuestas se encontrarán presentes al finalizar el documento en la sección de anexos.

8.2 Constantes del Controlador Mediante Diferentes Técnicas de Sintonización

Se sintonizo el Controlador PID usando Ziegler-Nichols; Existen dos reglas de sintonía propuesta por los ingenieros Ziegler y Nichols en 1942 para poder determinar los parámetros del controlador PID tomando como base la respuesta transitoria del sistema.

Por lo tanto, este tipo de sintonía es especialmente útil, principalmente cuando NO SE CONOCE EL MODELO MATEMÁTICO DE LA PLANTA. Obviamente también puede ser aplicado si el modelo de la planta es conocido, sin embargo, podemos coincidir que existen estrategias más interesantes cuando el modelo es conocido.

Método 1 (Lazo Abierto)

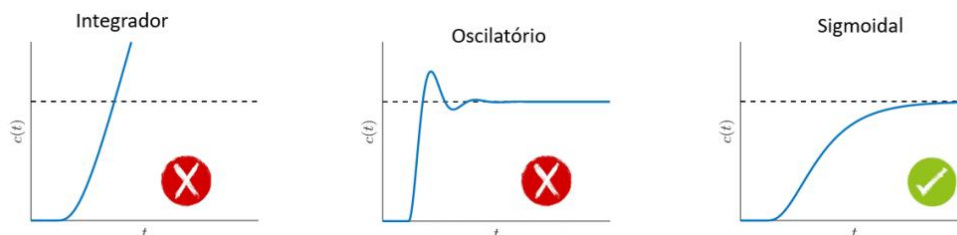
Este método de Ziegler Nichols se realiza con el sistema en **lazo abierto**, donde el controlador se coloca en modo manual para poder generar una variación del tipo escalón en la propia salida del controlador PID.

Esto quiere decir que la planta, en su entrada recibirá una entrada del tipo escalón (**generalmente escalón unitario**) y a partir de la respuesta de salida se obtienen los parámetros del PID. Este es el método de la curva de reacción de Ziegler-Nichols.

Sin embargo, para poder usar este método es necesario que el sistema tenga un comportamiento del **tipo sigmoïdal o forma de S**, esto quiere decir que la respuesta no puede tener sobre impulsos en lazo abierto, ni poseer una dinámica integradora que crezca constantemente con el tiempo. (Giraldo, Control Automático Educación , 2019)

Ilustración 9.

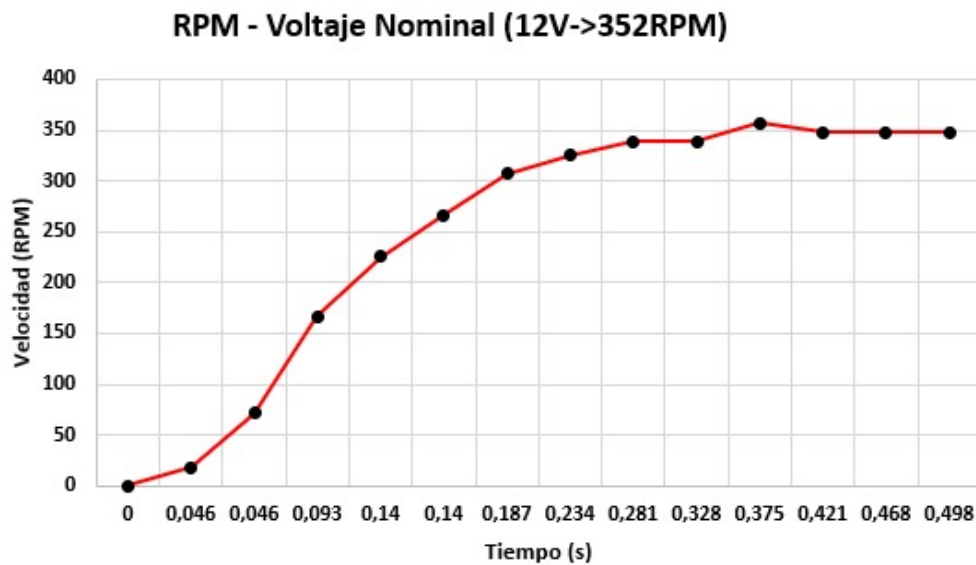
Tipo de respuesta otorgada por un sistema de primer o segundo orden.



Partiendo de la información anterior se concluyó que el sistema utilizado no podía ser trabajado con la técnica de Ziegler Nichols, debido a que la respuesta en el tiempo daba como resultado una curva oscilatoria, de este modo se procedió a trabajar mediante el método de curva de reacción y así obtener la curva correspondiente para implementar Ziegler Nichols teniendo como finalidad la obtención de las constantes del controlador.

Ilustración 10.

Curva de reacción del motor Dc.

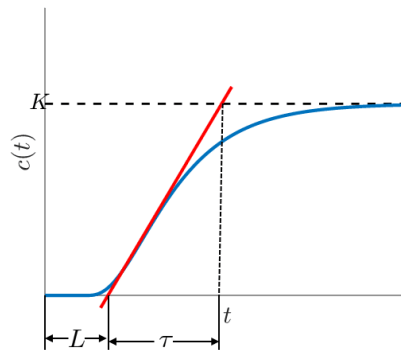


Este método como lo es mencionado anteriormente es utilizado en procesos donde no se tiene ningún modelo matemático por lo tanto se realizó omitiendo el modelo matemático ya presentado; para dar inicio a la solución se procede en una serie de pasos.

1. Se trazo una tangente en el punto de inflexión para hallar los parámetros de una función de transferencia de primer orden con retardo.

Ilustración 11.

Grafica de un sistema de primer orden con retardo.



$$Gp = \frac{350e^{-0.046s}}{0.094s+1} \quad (\text{Ecu.18})$$

2. Se aplico las ecuaciones de la tabla del método 1, para obtener los parámetros de los controladores

Tabla 4.

Variables de Ziegler y Nichols - método 1.

Controlador	Kp	Ti	Td
P	$\frac{\tau}{KL}$	∞	0
PI	$0.9 \frac{\tau}{KL}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{\tau}{KL}$	$2L$	$0.5L$

Teniendo en cuenta que los valores de K, L y T son los siguientes:

$$K=350$$

$$L= 0.046$$

$$T= 0.094$$

Para el control proporcional se despejó los valores de K, L y T en la ecuación entregada por la tabla 4. Variables de Ziegler y Nichols – método 1.

$$Kp = \frac{0.094}{(350)(0.046)} = 0.005838$$

En el control integral se despejaron los valores de K, L y T en las ecuaciones suministradas por la tabla 4. Variables de Ziegler y Nichols – método 1.

$$Kp = 0.9 \left(\frac{0.094}{(350)(0.046)} \right) = 0.005254$$

$$Ti = \frac{0.046}{0.3} = 0.313$$

El proceso para el control integral derivativo fue despejar los valores de K, L y T en las ecuaciones proporcionadas por la tabla 4. Variables de Ziegler y Nichols – método 1.

$$Kp = 1.2 \left(\frac{0.094}{(350)(0.046)} \right) = 0.007006$$

$$Ti = 2(0.046) = 0.092$$

$$Td = 0.5(0.046) = 0.023$$

3. Luego de obtener los valores necesarios de cada uno de los controladores se realizó el procedimiento para así adquirir las constantes Kp, Ki, Kd partiendo de las fórmulas halladas. (Teoría de Control, 2020)

$$Ki = \frac{Kp}{Ti} \tag{Ecu. 19}$$

$$Kd = KpTd \tag{Ecu.20}$$

Para el controlador integral se reemplazaron los valores de Kp y Ti en la ecuación 19 y se obtuvo como resultado el valor de Ki.

$$Ki = \frac{0.005254}{0.313} = 0.0016785$$

En el control integral derivativo se reemplazaron los valores de K_p , T_i y T_d en las ecuaciones 19 y 20, se obtuvo como resultado los valores de K_i y K_d .

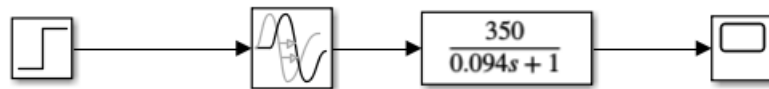
$$K_i = \frac{0.007006}{0.092} = 0.0076152$$

$$K_d = (0.007006)(0.023) = 0.0001611$$

Luego de que se obtuvieron todos los valores correspondientes de cada una de las constantes para su respectivo controlador se procedió a simular en el programa de SIMULINK en el cual se consiguieron como resultado las gráficas presentadas a continuación.

Ilustración 12.

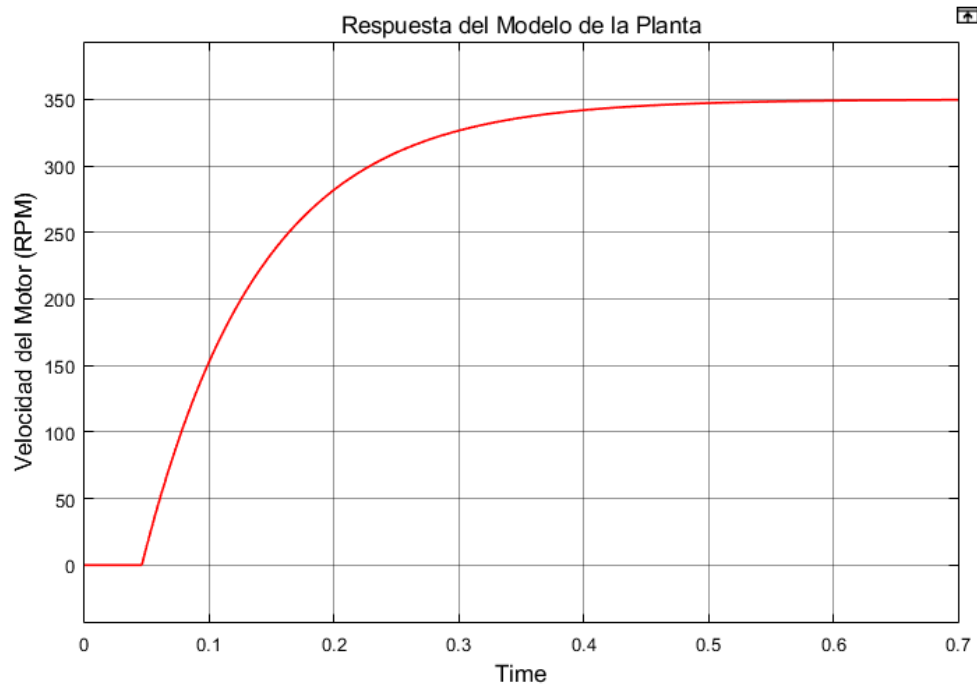
Modelo de la planta por el método 1 de Ziegler y Nichols.



Partiendo del diagrama de bloques implementado en la ilustración 12 se obtuvo como respuesta la siguiente gráfica, a partir de ella se pudo observar y comparar su comportamiento con la ilustración 10.

Ilustración 13.

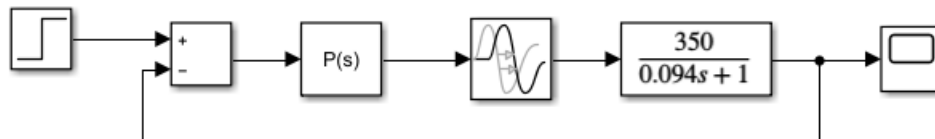
Grafica del modelo de la planta por el método 1 de Ziegler y Nichols.



Teniendo en cuenta la ilustración 13 y su similitud con la ilustración 10 se corroboro que el modelo de la planta es viable y funcional.

Ilustración 14.

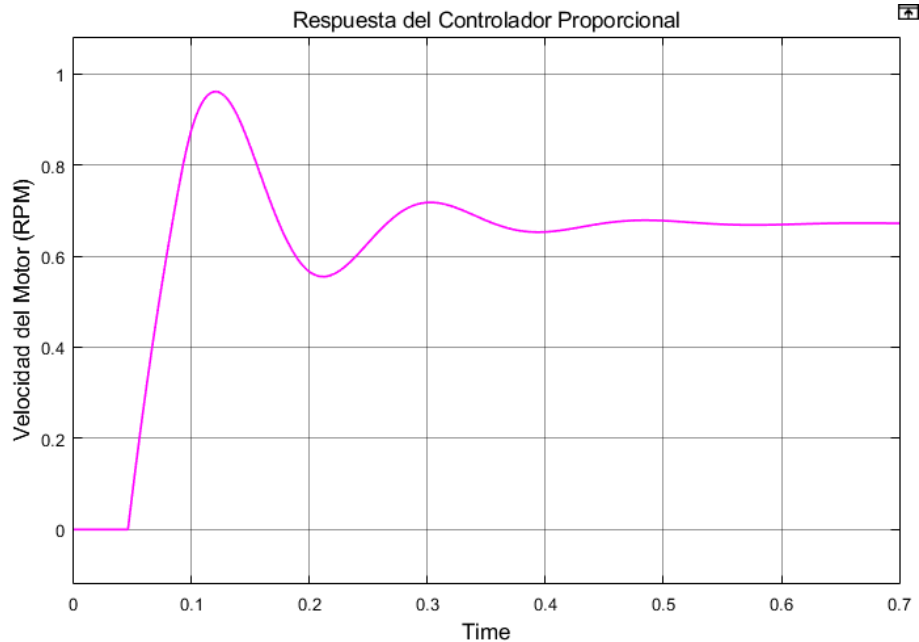
Modelo de la planta implementando el control proporcional.



Seguidamente se implementó el controlador proporcional con el valor de K_p que se halló con anterioridad, obteniendo como resultado la ilustración 14.

Ilustración 15.

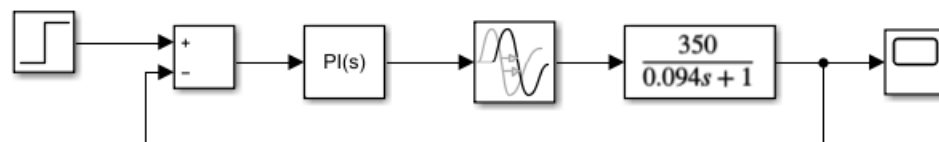
Grafica del modelo de la planta implementando el control proporcional.



En la ilustración 15 se observa un comportamiento con un tiempo pico 0.12, una tolerancia permitida del 2% al 5%, el cual quiere decir que el sistema de segundo de orden se encuentra dentro del régimen permanente y cuenta con un margen de sobrepaso del 0.3.

Ilustración 16.

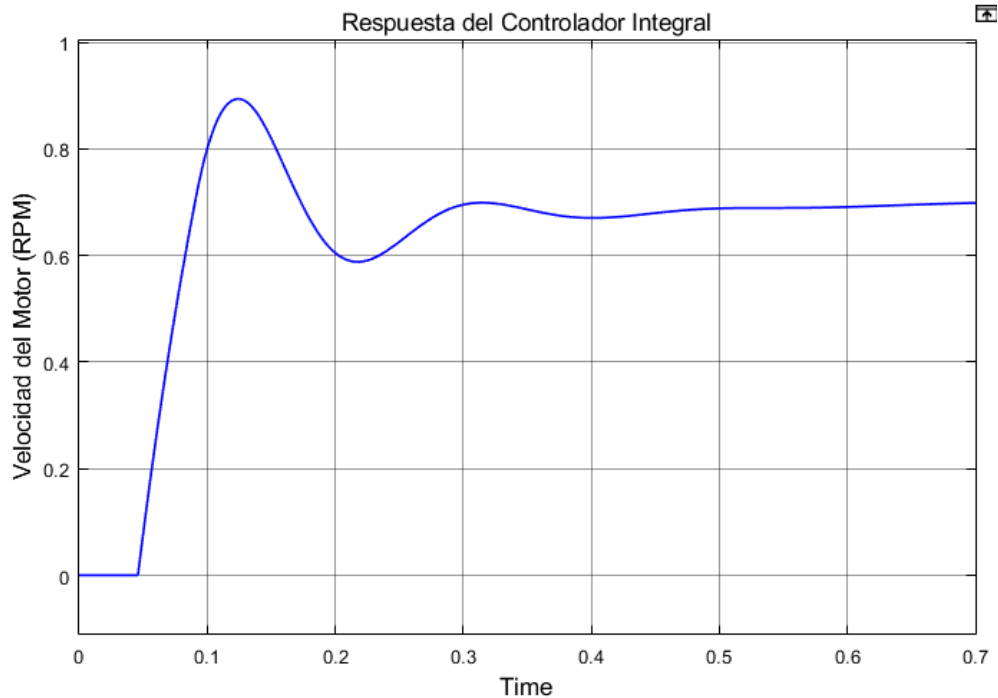
Modelo de la planta implementando el control integral.



Seguidamente se implementó el controlador integral con el valor de K_p y K_i que se halló con anterioridad, obteniendo como resultado la ilustración 16.

Ilustración 17.

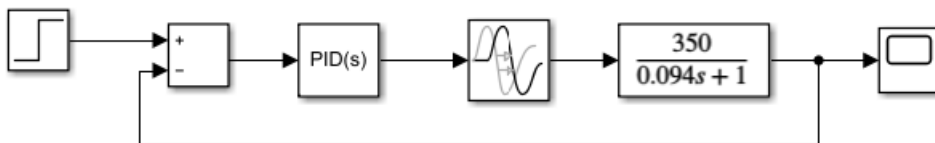
Grafica del modelo de la planta implementando el control integral.



En la ilustración 17 se observa un comportamiento con un tiempo pico 0.12, una tolerancia permitida del 2%, el cual quiere decir que el sistema de segundo de orden se encuentra dentro del régimen permanente y cuenta con un margen de sobrepaso del 0.2.

Ilustración 18.

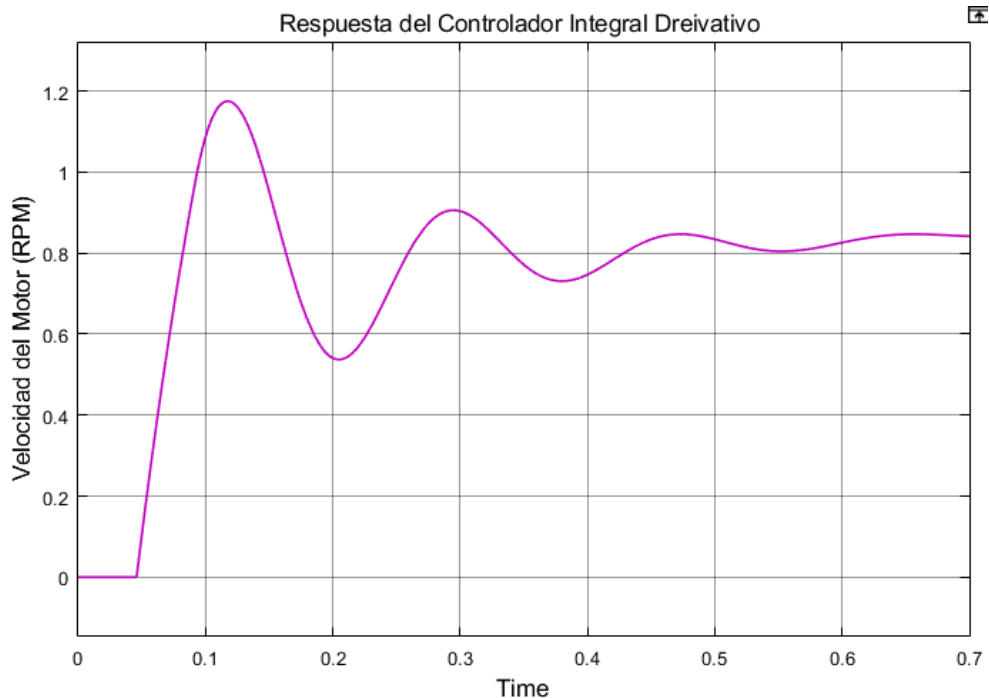
Modelo de la planta implementando el control integral derivativo.



Seguidamente se implementó el controlador integral derivativo con el valor de K_p , K_i y K_d que se halló con anterioridad, obteniendo como resultado la ilustración 18.

Ilustración 19.

Grafica del modelo de la planta implementando el control integral derivativo.

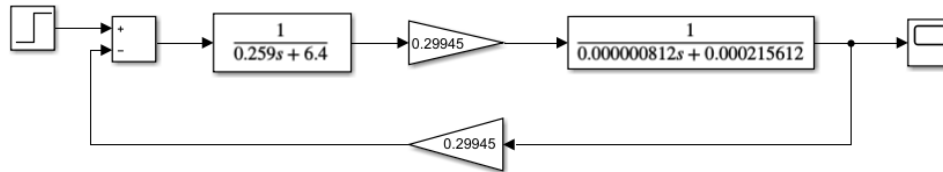


En la ilustración 19 se observa un comportamiento con un tiempo pico 0.12, una tolerancia permitida del 5%, el cual quiere decir que el sistema de segundo de orden se encuentra dentro del régimen permanente y cuenta con un margen de sobrepaso del 0.2.

Como método de sintonización también se recurrió al PID TUNER del software Matlab. Para obtener las constantes de cada uno de los controladores, el cual ajusta automáticamente las ganancias de un controlador PID para una planta SISO con el fin de lograr un equilibrio entre el rendimiento y la robustez, puede ser utilizado solo con una planta representada por un modelo LTI numérico, como lo es la función de transferencia o un modelo de espacio de estados. (MathWorks, 2023)

Ilustración 20.

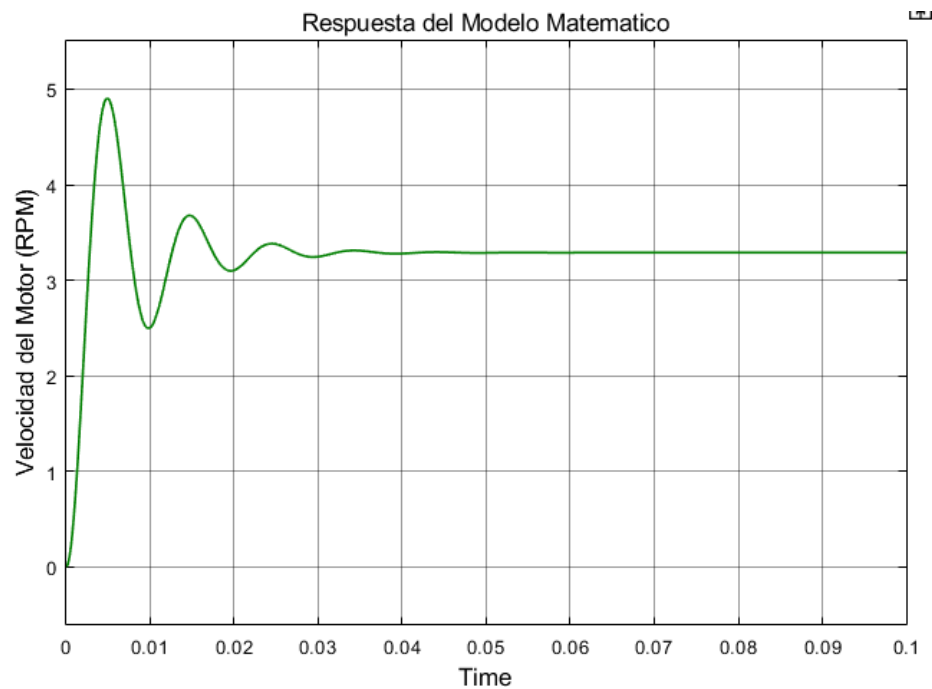
Diagrama de bloques del modelo matemático de la planta.



Después de obtenido el diagrama de bloques se implementó en el software de Matlab para así lograr una respuesta visual del comportamiento de la planta.

Ilustración 21.

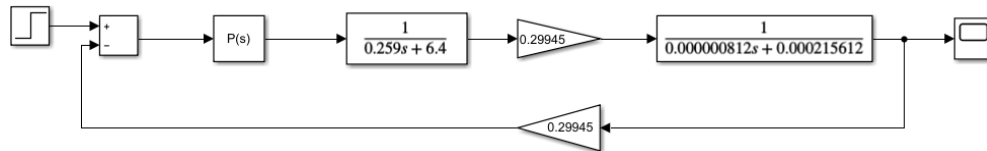
Grafica del diagrama de bloques del modelo matemático de la planta.



En la respuesta grafica dada en la ilustración 20 se muestra un comportamiento con un tiempo pico 0.05, una tolerancia permitida del 2%, el cual quiere decir que el sistema de segundo de orden se encuentra dentro del régimen permanente y cuenta con un margen de sobrepaso del 1.5.

Ilustración 22.

Diagrama de bloques del modelo matemático de la planta implementando el control proporcional.



Se empleo el diagrama de bloques y se anexo el control proporcional a la planta entregada por el modelo matemático como se observa en la ilustración 22.

Ilustración 23.

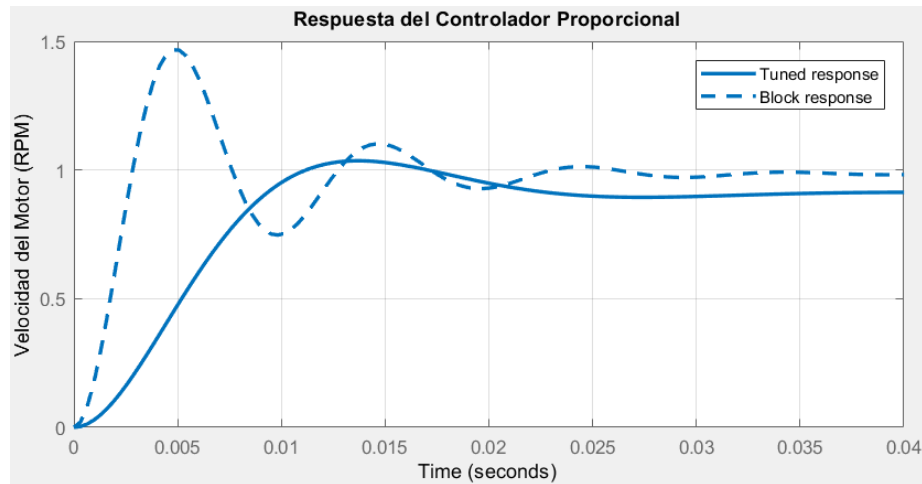
Parámetros del controlador proporcional otorgados por el PID TUNER.

Controller Parameters		
	Tuned	Block
P	0.15746	1
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a
Performance and Robustness		
	Tuned	Block
Rise time	0.00629 seconds	0.00187 seconds
Settling time	0.0213 seconds	0.0258 seconds
Overshoot	13.7 %	48.9 %
Peak	1.04	1.47
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	60 deg @ 200 rad/s	25.2 deg @ 626 rad/s
Closed-loop stability	Stable	Stable

Partiendo del diagrama de bloques visualizado en la ilustración 22, se ejecutó la aplicación de PID TUNER y se obtuvo como resultado los parámetros necesarios para el controlador proporcional vistos en la ilustración 23.

Ilustración 24.

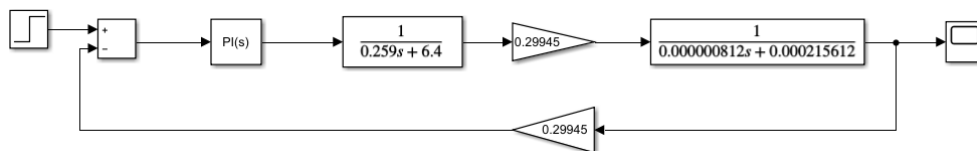
Grafica de respuesta del PID TUNER para el controlador proporcional.



Posterior a los parámetros visualizados en la ilustración 23 se obtuvo como respuesta la ilustración 24 la cual entrega un tiempo pico de 0.013, una tolerancia permitida del 2%, el cual quiere decir que el sistema de segundo de orden se encuentra dentro del régimen permanente y cuenta con un margen de sobrepaso del 0.12.

Ilustración 25.

Diagrama de bloques del modelo matemático de la planta implementando el control integral.



Se empleó el diagrama de bloques y se anexó el control integral a la planta entregada por el modelo matemático como se observa en la ilustración 25.

Ilustración 26.

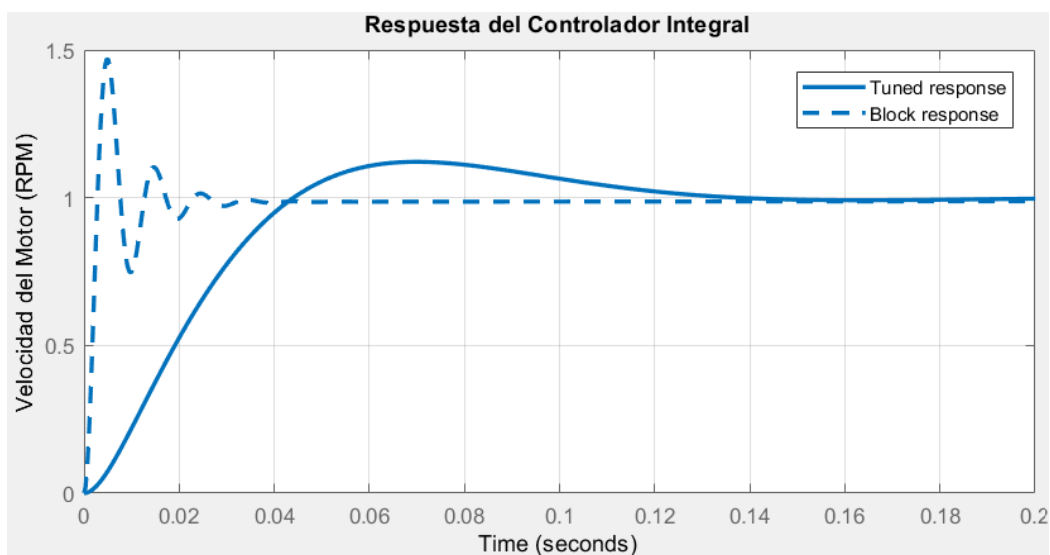
Parámetros del controlador integral otorgados por el PID TUNER.

Controller Parameters		
	Tuned	Block
P	0.020278	1
I	1.0615	1
D	n/a	n/a
N	n/a	n/a
Performance and Robustness		
	Tuned	Block
Rise time	0.0308 seconds	0.0019 seconds
Settling time	0.121 seconds	0.0313 seconds
Overshoot	12.1 %	46.9 %
Peak	1.12	1.47
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	60 deg @ 43.7 rad/s	25.1 deg @ 626 rad/s
Closed-loop stability	Stable	Stable

Partiendo del diagrama de bloques visualizado en la ilustración 25 se ejecutó la aplicación de PID TUNER y se obtuvo como resultado los parámetros necesarios para el controlador integral los cuales se pueden observar en la ilustración 26.

Ilustración 27.

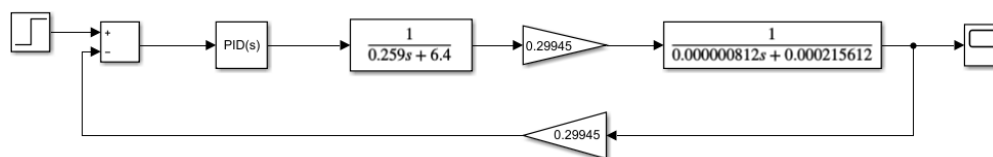
Grafica de respuesta del PID TUNER para el controlador integral.



Posterior a los parámetros visualizados en la ilustración 26 se obtuvo como respuesta la ilustración 27 la cual entrega un tiempo pico de 0.07, la curva de salida corresponde con un sistema de segundo orden subamortiguado y cuenta con un margen de sobrepaso del 0.12.

Ilustración 28.

Diagrama de bloques del modelo matemático de la planta implementando el control integral derivativo.



Se empleo el diagrama de bloques y se anexo el control integral derivativo a la planta entregada por el modelo matemático como se observa en la ilustración 28.

Ilustración 29.

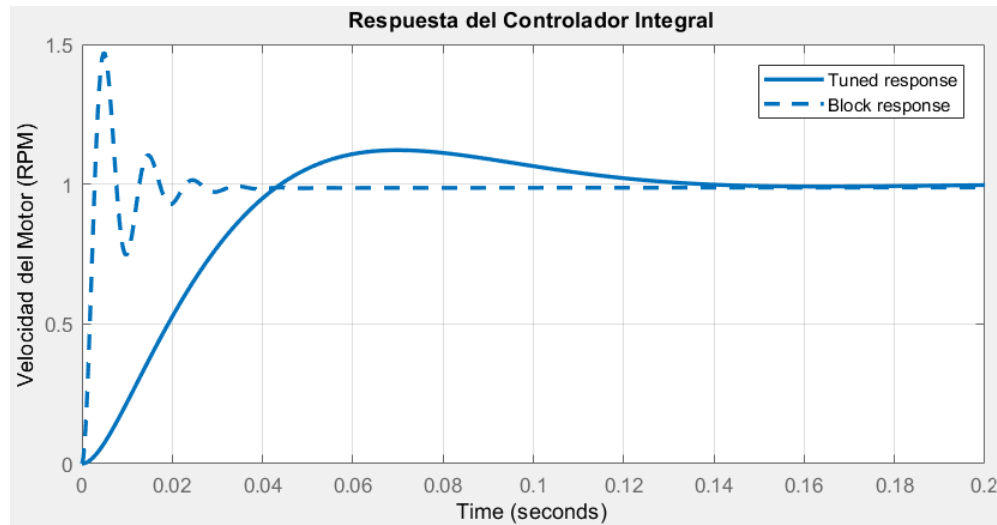
Parámetros del controlador integral derivativo otorgados por el PID TUNER.

Controller Parameters		
	Tuned	Block
P	0.023817	1
I	0.97466	1
D	3.3889e-05	0
N	4989.3557	100
Performance and Robustness		
	Tuned	Block
Rise time	0.0344 seconds	0.0019 seconds
Settling time	0.126 seconds	0.0313 seconds
Overshoot	7.2 %	46.9 %
Peak	1.07	1.47
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	69 deg @ 43.7 rad/s	25.1 deg @ 626 rad/s
Closed-loop stability	Stable	Stable

Partiendo del diagrama de bloques visualizado en la ilustración 28 se ejecutó la aplicación de PID TUNER y se obtuvo como resultado los parámetros necesarios para el controlador integral derivativo los cuales se pueden observar en la ilustración 29.

Ilustración 30.

Grafica de respuesta del PID TUNER para el controlador integral derivativo.



Posterior a los parámetros visualizados en la ilustración 29 se obtuvo como respuesta la ilustración 30 la cual entrega un tiempo pico de 0.07, la curva de salida corresponde con un sistema de segundo orden subamortiguado y cuenta con un margen de sobrepaso del 0.07.

Además, se discretizó la señal obtenida en el tiempo dada en la ecuación 18, para optimizar la señal recibida y para eliminar los posibles errores del espectro continuo, la discretización es el paso previo a su digitalización, proceso que agrega una determinada codificación a la señal muestreada, se realizó este proceso debido a que el sistema será implementado en un dispositivo programable. (Diaz, 2023) (Barco, 2023)

Ilustración 31.

Código implementado en Matlab para la obtención de la transformada Z del sistema.

```
clc
H = tf([350],[0.094 1], 'InputDelay',0.046);
Hd = c2d(H,0.005, 'foh')
step(H, '--', Hd, '-r')
```

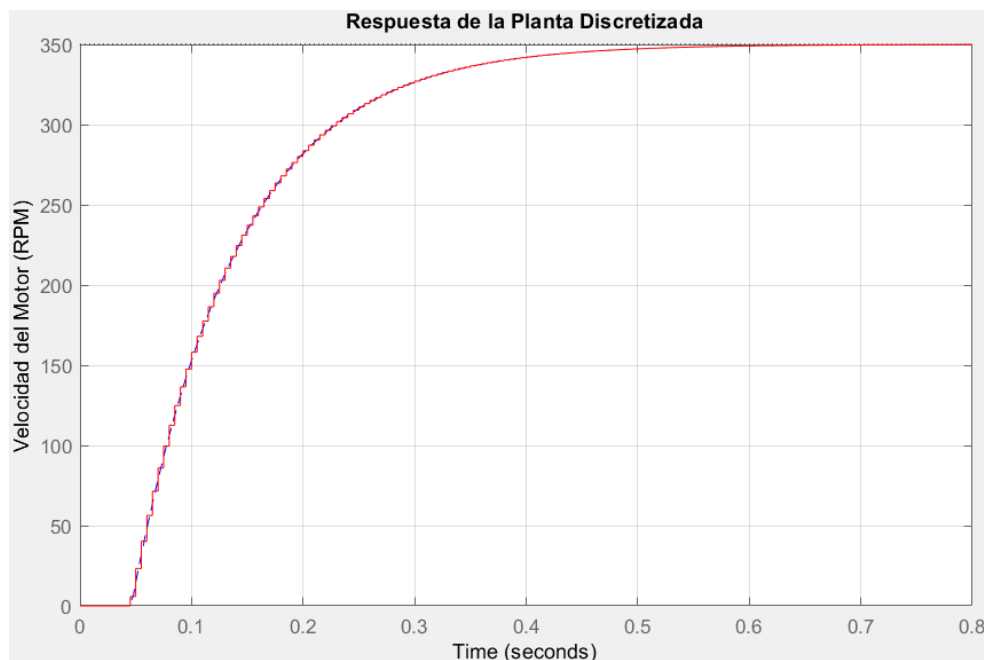
Para realizar este paso se da inicio con la consecuencia de la transformada z, partiendo de la función de transferencia hallada por el método 1 de Ziegler y Nichols la cual se puede observar en la ecuación 18, teniendo en cuenta el código que se implementó en el software de Matlab entregado en la ilustración 31.

$$Hd = z^{-9} * \frac{5.874z^2 + 11.9z + 0.3543}{z^2 - 0.9482z} \quad (\text{Ecu.21})$$

Se ejecutó el código de Matlab de la ilustración 31, para discretizar la planta obtenida por el método 1 de Ziegler y Nichols la cual entrega como respuesta la función de transferencia en el dominio de Z visible en la ecuación 21.

Ilustración 32.

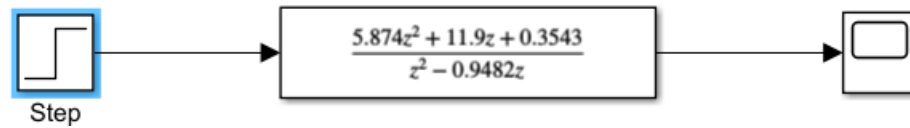
Respuesta grafica de la discretización de la planta.



En la ilustración 32 se puede observar el sistema lineal como referencia y el sistema discretizado los cuales presentan un comportamiento de un sistema de segundo orden subamortiguado.

Ilustración 33.

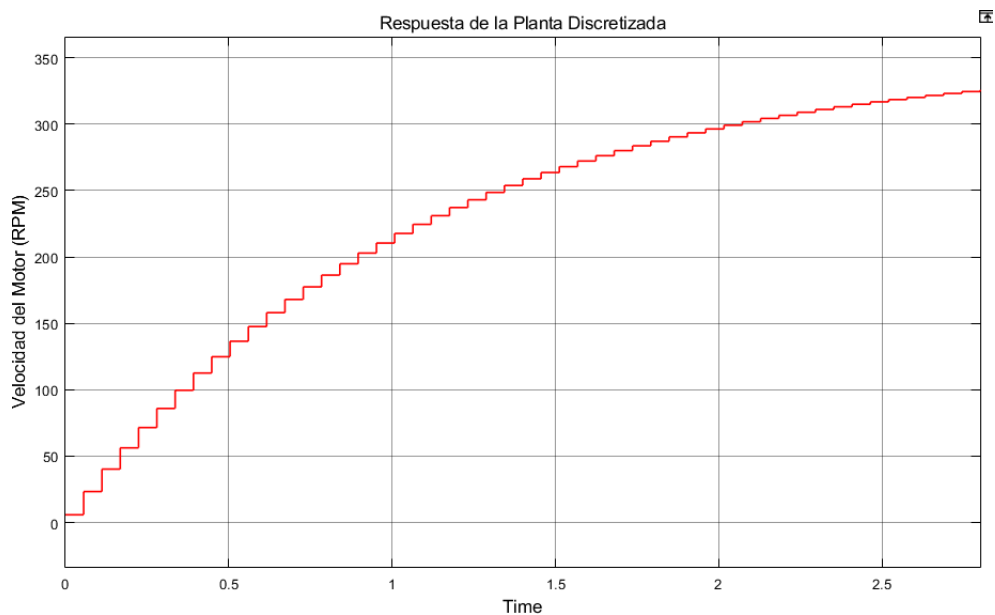
Modelo de la planta por el método 1 de Ziegler y Nichols discretizada.



Se procedió a realizar el diagrama de bloques con la respuesta entregada por el código realizado en Matlab como se puede observar en la ilustración 33.

Ilustración 34.

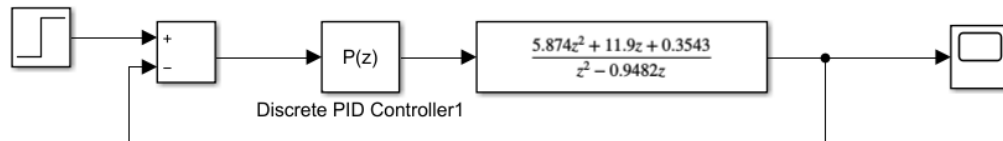
Grafica del modelo de la planta por el método 1 de Ziegler y Nichols discretizada.



En la ilustración 34 se muestra un comportamiento similar a la respuesta ya visualizada anteriormente en la ilustración 32, partiendo de ella se corrobora si viabilidad y funcionalidad de la planta discretizada.

Ilustración 35.

Modelo de la planta por el método 1 de Ziegler y Nichols discretizada Implementando el control proporcional.



Se anexo el controlador discreto al modelo de la planta para obtener las variables de un controlador proporcional como se observa en la ilustración 35.

Ilustración 36.

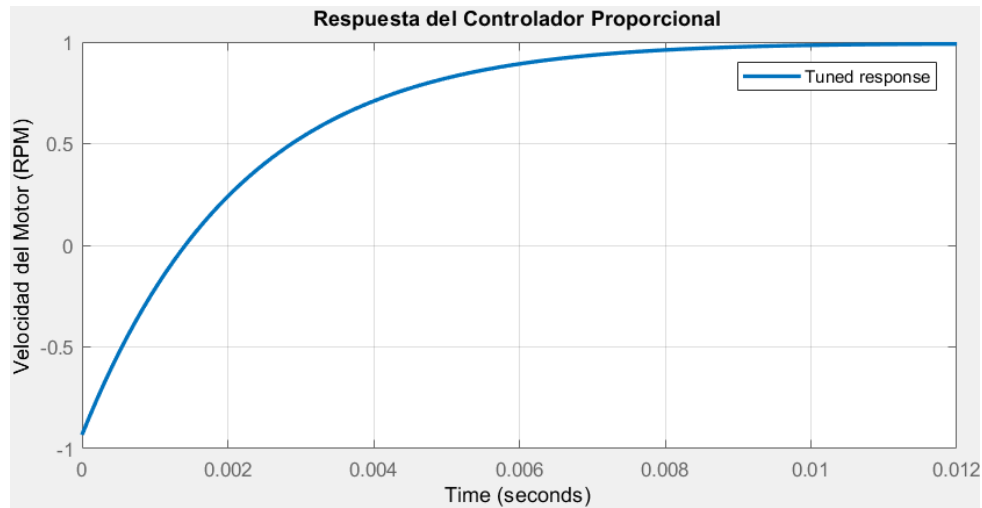
Parámetros del controlador proporcional otorgados por el PID TUNER.

Controller Parameters	
	Tuned
P	0.1657
I	n/a
D	n/a
N	n/a
Performance and Robustness	
	Tuned
Rise time	0.00371 seconds
Settling time	0.0061 seconds
Overshoot	0.832 %
Peak	0.991
Gain margin	6.33 dB @ Inf rad/s
Phase margin	60 deg @ 339 rad/s
Closed-loop stability	Stable

Se ejecuto la aplicación del PID TUNER de Matlab para hallar dichas variables, el cual entrego como resultado la ilustración 36, en la cual se observa el valor de K_p .

Ilustración 37.

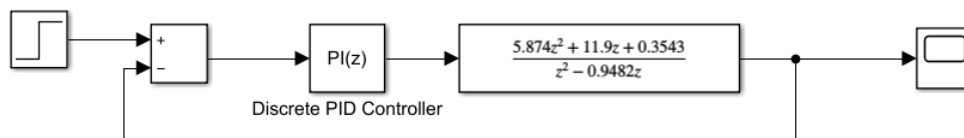
Grafica de respuesta del PID TUNER para el controlador proporcional.



La ilustración 37 da como respuesta del controlador proporcional otorgada por el PID TUNER de Matlab un sistema sub amortiguado con un tiempo pico de 0.012 sin margen de sobre paso.

Ilustración 38.

Modelo de la planta por el método 1 de Ziegler y Nichols discretizada implementando el control integral.



Se anexa el controlador discreto al modelo de la planta para obtener las variables de un controlador integral como se observa en la ilustración 38.

Ilustración 39.

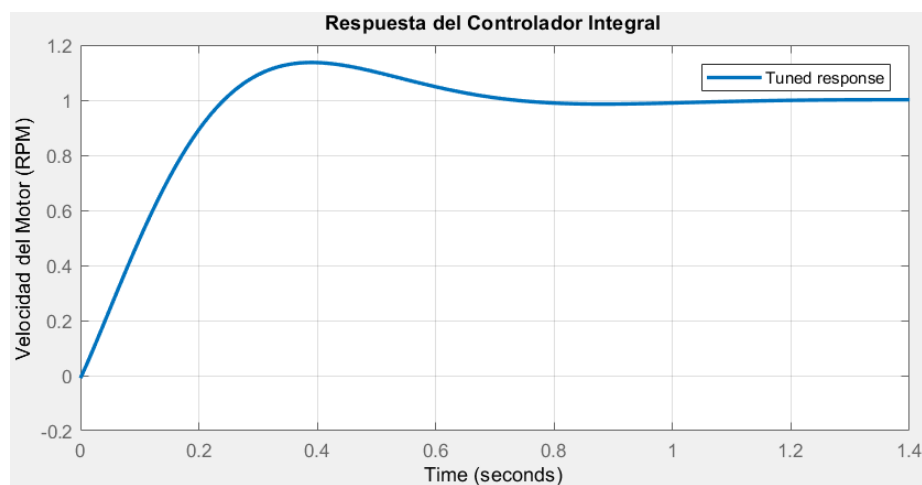
Parámetros del controlador integral otorgados por el PID TUNER.

Controller Parameters	
	Tuned
P	0.0030939
I	0.047358
D	n/a
N	n/a
Performance and Robustness	
	Tuned
Rise time	0.151 seconds
Settling time	0.554 seconds
Overshoot	13.7 %
Peak	1.14
Gain margin	40.9 dB @ Inf rad/s
Phase margin	60 deg @ 9.52 rad/s
Closed-loop stability	Stable

Se ejecuto la aplicación del PID TUNER de Matlab para hallar dichas variables, el cual entrego como resultado la ilustración 39, en la cual se observa el valor de K_p y K_i .

Ilustración 40.

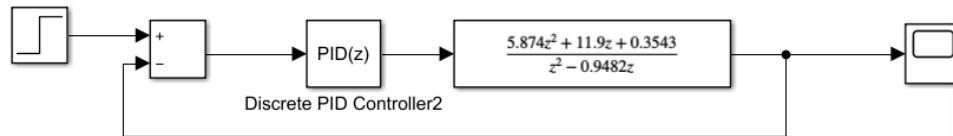
Grafica de respuesta del PID TUNER para el controlador integral.



La ilustración 40 da como respuesta del controlador integral otorgada por el PID TUNER de Matlab un sistema sub amortiguado con un tiempo pico de 0.4 y un margen de sobre paso de 0.14.

Ilustración 41.

Modelo de la planta por el método 1 de Ziegler y Nichols discretizada implementando el control integral derivativo.



Se anexo el controlador discreto al modelo de la planta para obtener las variables de un controlador integral derivativo como se observa en la ilustración 41.

Ilustración 42.

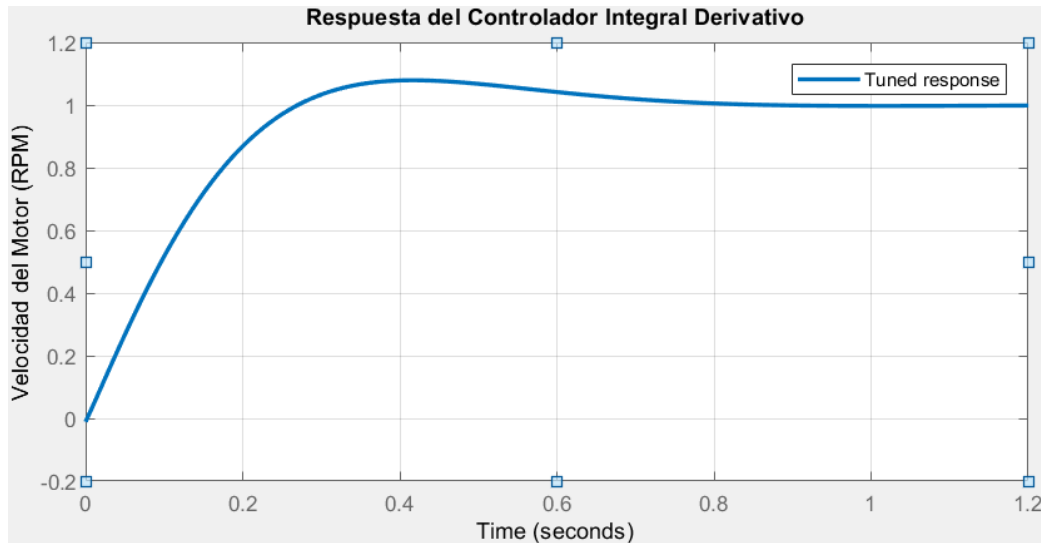
Parámetros del controlador integral derivativo otorgados por el PID TUNER.

Controller Parameters	
	Tuned
P	0.0039063
I	0.040982
D	-1.7557e-05
N	16.3762
Performance and Robustness	
	Tuned
Rise time	0.162 seconds
Settling time	0.578 seconds
Overshoot	7.9 %
Peak	1.08
Gain margin	39.5 dB @ Inf rad/s
Phase margin	69 deg @ 9.52 rad/s
Closed-loop stability	Stable

Se ejecuto la aplicación del PID TUNER de Matlab para hallar dichas variables, el cual entrego como resultado la ilustración 42, en la cual se observa el valor de Kp, Ki y Kd

Ilustración 43.

Grafica de respuesta del PID TUNER para el controlador integral derivativo.



La ilustración 43 da como respuesta del controlador integral derivativo otorgada por el PID TUNER de Matlab un sistema sub amortiguado con un tiempo pico de 0.4 y un margen de sobre paso de 0.08.

8.3 Software Basado en Arduino para Implementar las Distintas Técnicas de Control Convencionales

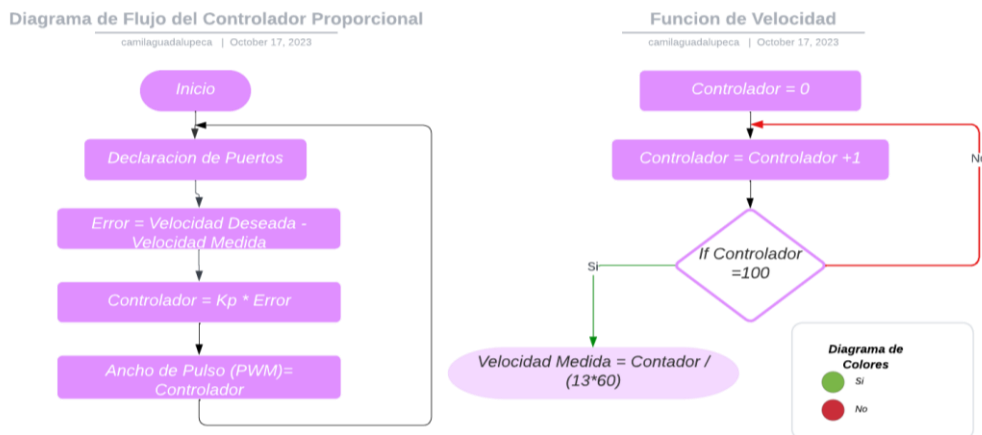
La programación de Arduino es la programación de un microcontrolador. Programar Arduino consiste en traducir a líneas de código las tareas automatizadas que queremos hacer leyendo de los sensores y en función de las condiciones del entorno programar la interacción con el mundo exterior mediante unos actuadores.

Arduino proporciona un entorno de programación sencillo y potente para programar, pero además incluye las herramientas necesarias para compilar el programa y “quemar” el programa ya compilado en la memoria flash del microcontrolador. (Crespo, 2017)

Los diagramas de flujo son útiles para escribir un programa o algoritmo y explicárselo a otros o colaborar con otros en el mismo. Puede usar un diagrama de flujo para explicar detalladamente la lógica detrás de un programa antes de empezar a codificar el proceso automatizado. (Lucid Software Inc, 2023)

Ilustración 44.

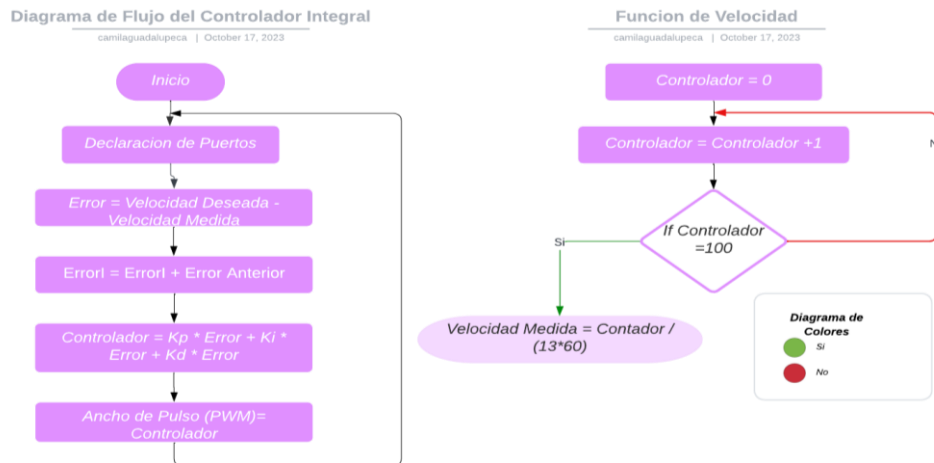
Diagrama de flujo del código implementado en Arduino para el controlador proporcional.



En la ilustración 44 se evidencia el diagrama de flujo del código implementado en el software de Arduino para la obtención de las variables del controlador proporcional y su respuesta grafica.

Ilustración 45.

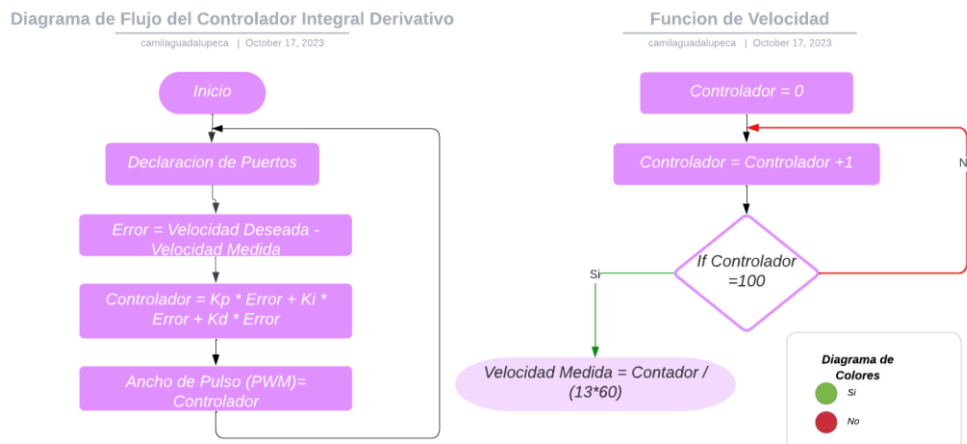
Diagrama de flujo del código implementado en Arduino para el controlador integral.



En la ilustración 45 se evidencia el diagrama de flujo del código implementado en el software de Arduino para la obtención de las variables del controlador integral y su respuesta grafica.

Ilustración 46.

Diagrama de flujo del código implementado en Arduino para el controlador integral derivativo.



En la ilustración 46 se evidencia el diagrama de flujo del código implementado en el software de Arduino para la obtención de las variables del controlador integral derivativo y su respuesta grafica.

8.4 Interfaz Gráfica para la Supervisión de la Velocidad del Motor Dc

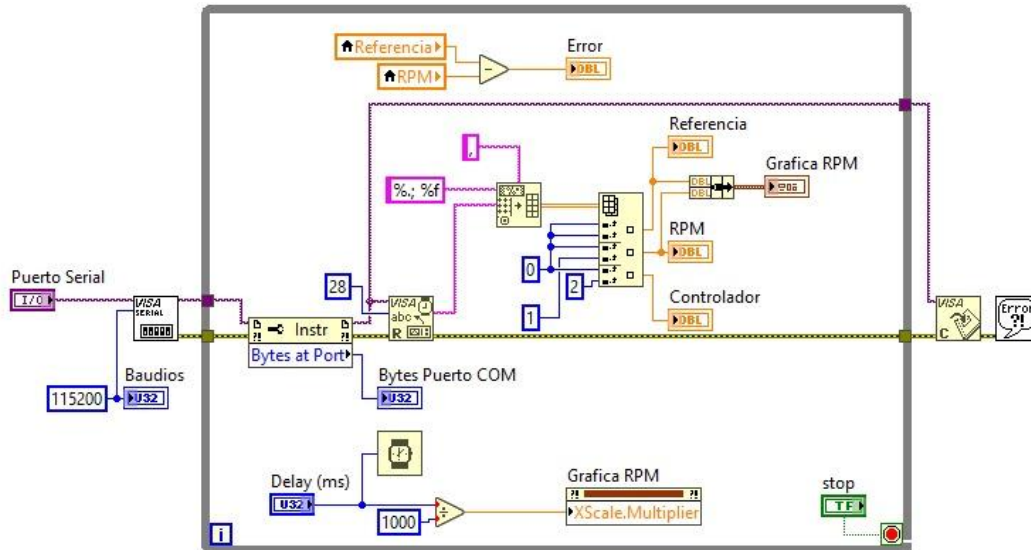
La interfaz gráfica de usuario, conocida también como GUI (del inglés *graphical user interface*), es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador (Wikipedia, 2023)

Los ingenieros suelen usar LabVIEW para construir una interfaz de usuario en minutos. LabVIEW tiene elementos visuales específicos para pruebas en una pantalla personalizada para el monitoreo y control de los sistemas de pruebas. (National Instruments Corp, 2023)

LabVIEW es un software que proporciona un potente entorno de desarrollo gráfico para el diseño de aplicaciones de Ingeniería de adquisición de datos, análisis de medidas y presentación de datos gracias a un lenguaje de programación sin la complejidad de otras herramientas de desarrollo. (Universidad de Cantabria, 2023)

Ilustración 47.

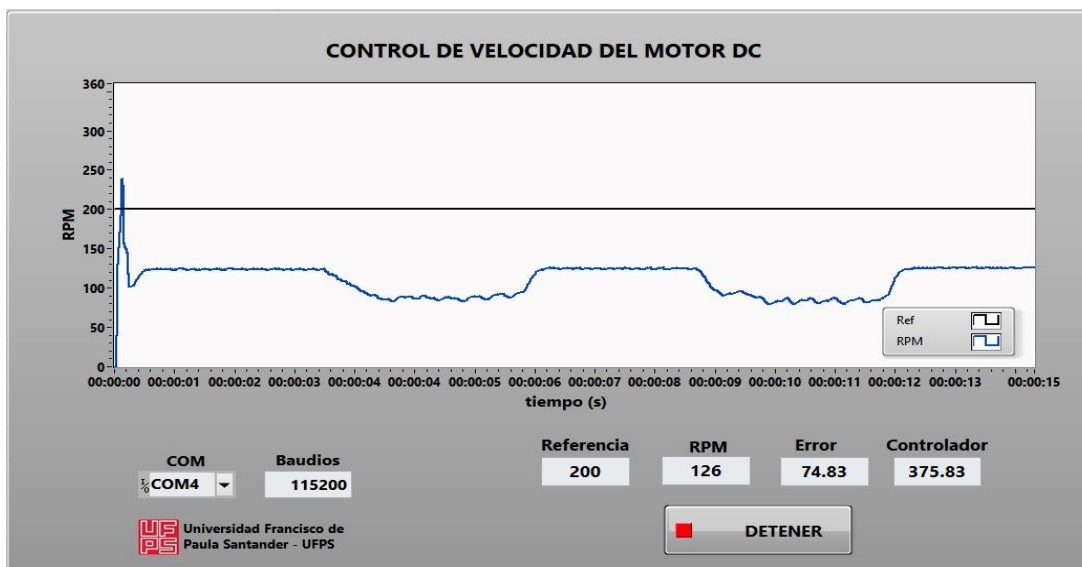
Pantalla de Programación en LabVIEW.



Se realizó en el software de LabVIEW debido a su facilidad de uso, un código de programación el cual se puede observar en la ilustración 45.

Ilustración 48.

Interfaz gráfica diseñada en LabVIEW.



Se observa la facilidad de uso e interacción de la interfaz gráfica realizada en el software de LabVIEW en la ilustración 46.

8.5 Validación de las Técnicas de Control Convencionales

Los valores de las constantes obtenidas mediante la programación realizada en el software Arduino para cada uno de los controladores se pueden observar a continuación:

Método 1 de Ziegler y Nichols:

P: $K_p=0.002*1$

PI: $K_p = 0.0024486*0.6$ y $K_i = 0.01632426*0.3$

PID: $K_p = 0.0024486*3$ $K_i = 0.01632426*2.5$ $K_d = 0.000091823*1$

Mediante el modelo matemático:

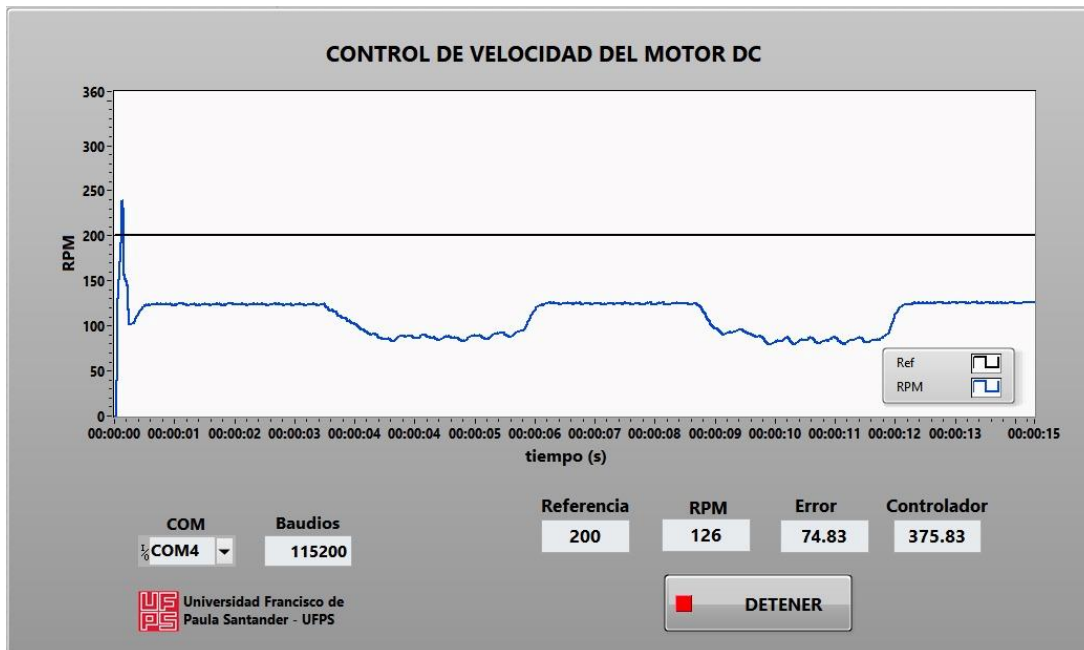
P: $K_p=0.0026*13$.

PI: $K_p = 0.00604928*0.2$. y $K_i = 0.0251980*0.05$.

PID: $K_p = 0.0329664*0.1$ $K_i = 0.20733585*0.1$ $K_d = 0.0013104144*0.1$

Ilustración 49.

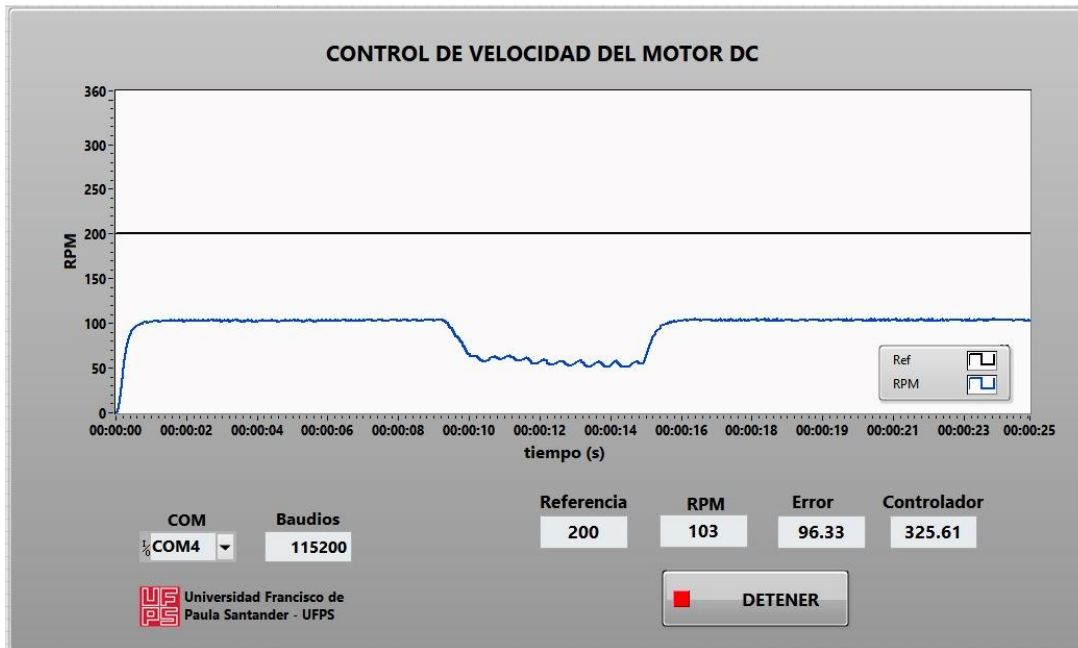
Respuesta del software de Arduino respecto al controlador proporcional por el método 1 de Ziegler y Nichols.



La ilustración 47 entrega como respuesta un sistema que contiene un error del 74.83%, esto quiere decir que dicho sistema nunca llegara al valor de referencia otorgado por los elaboradores.

Ilustración 50.

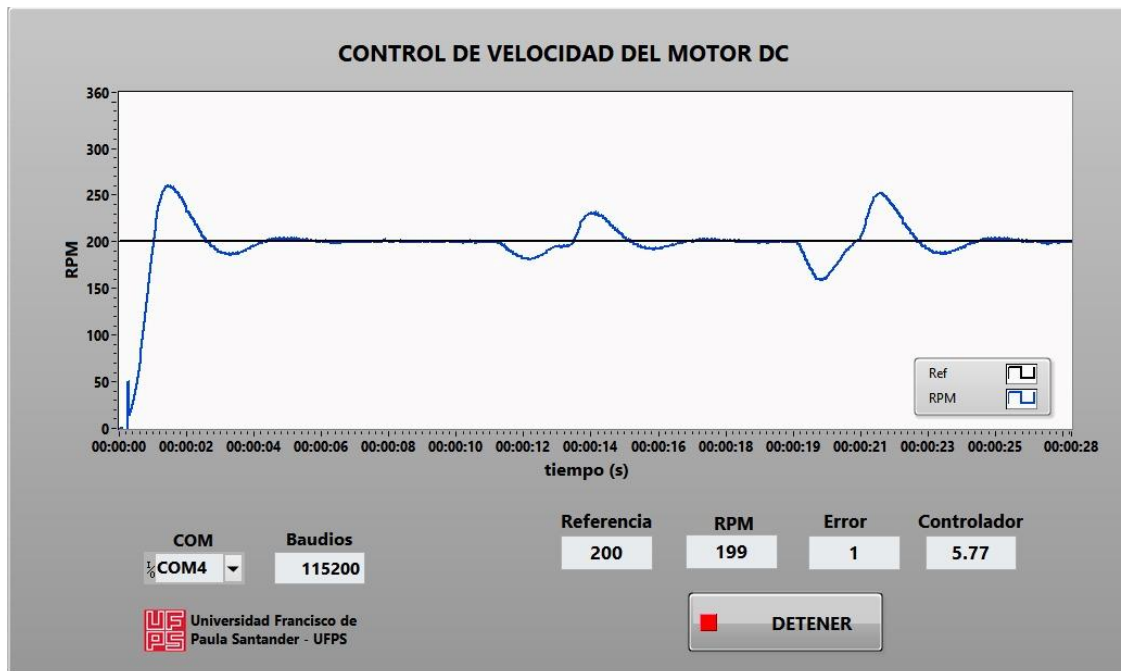
Respuesta del software de Arduino respecto al controlador proporcional por medio del modelo matemático.



La ilustración 48 entrega como respuesta un sistema que contiene un error del 96.33%, esto quiere decir que dicho sistema nunca llegara al valor de referencia otorgado por los elaboradores.

Ilustración 51.

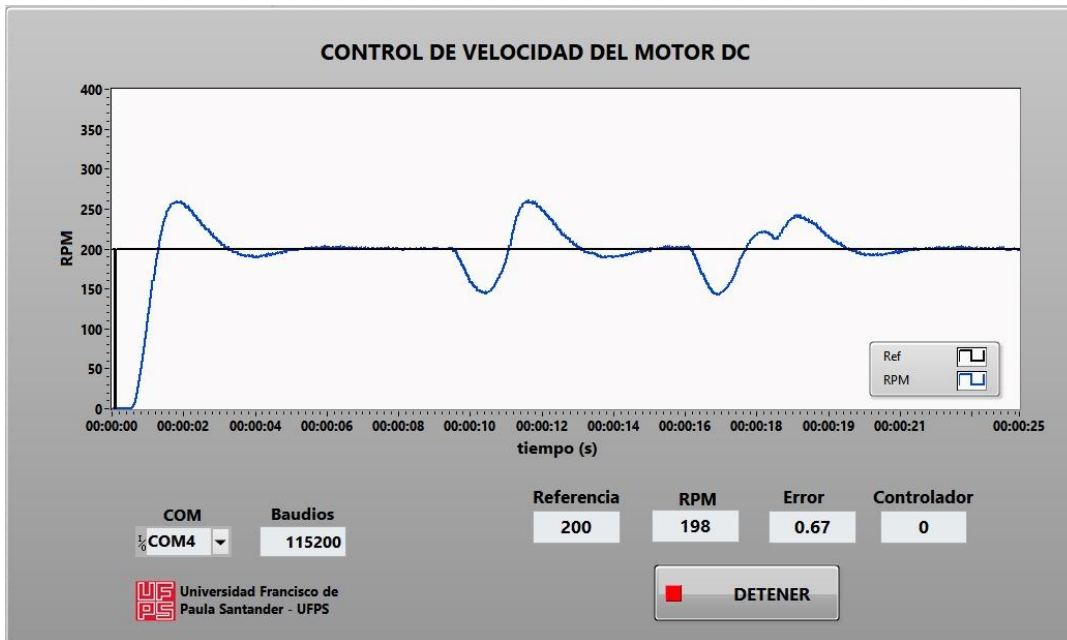
Respuesta del software de Arduino respecto a controlador integral por el método 1 de Ziegler y Nichols.



En la ilustración 49 se puede observar un sistema sub amortiguado, el cual cuenta con un tiempo pico de 0.02 segundos, un margen de sobrepaso de 52 Rpm, un error de 1 Rpm y un tiempo de establecimiento de 0.07 segundos luego de presentar perturbaciones.

Ilustración 52.

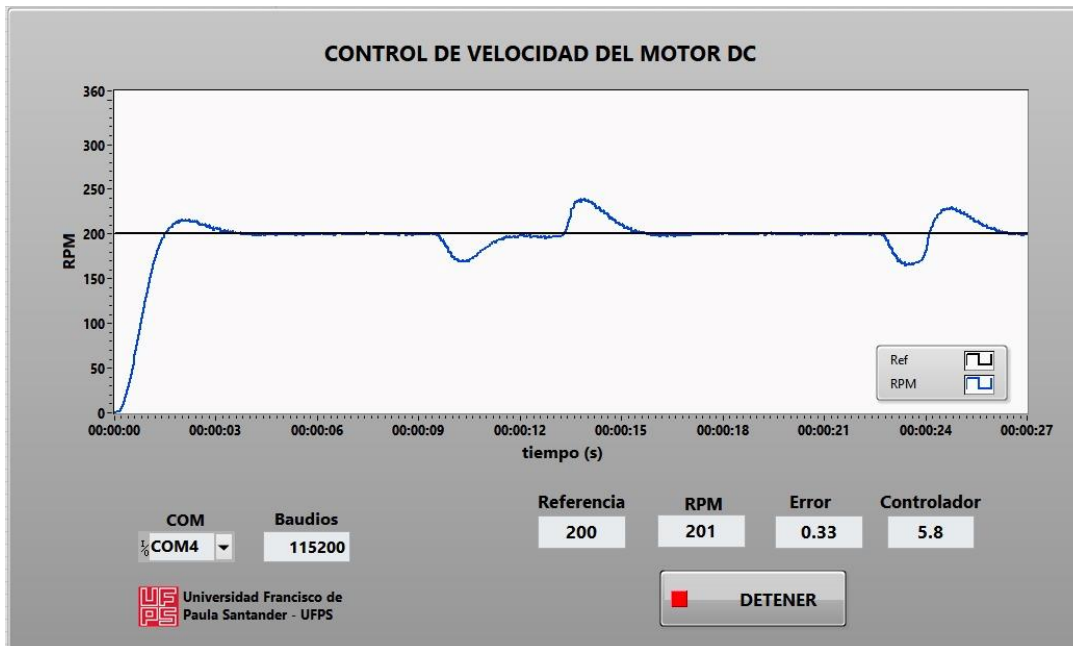
Respuesta del software de Arduino respecto al controlador integral por medio del modelo matemático.



En la ilustración 50 se puede observar un sistema sub amortiguado, el cual cuenta con un tiempo pico de 0.02 segundos, un margen de sobrepaso de 51 Rpm, un error de 0.67 Rpm y un tiempo de establecimiento de 0.055 segundos luego de presentar perturbaciones.

Ilustración 53.

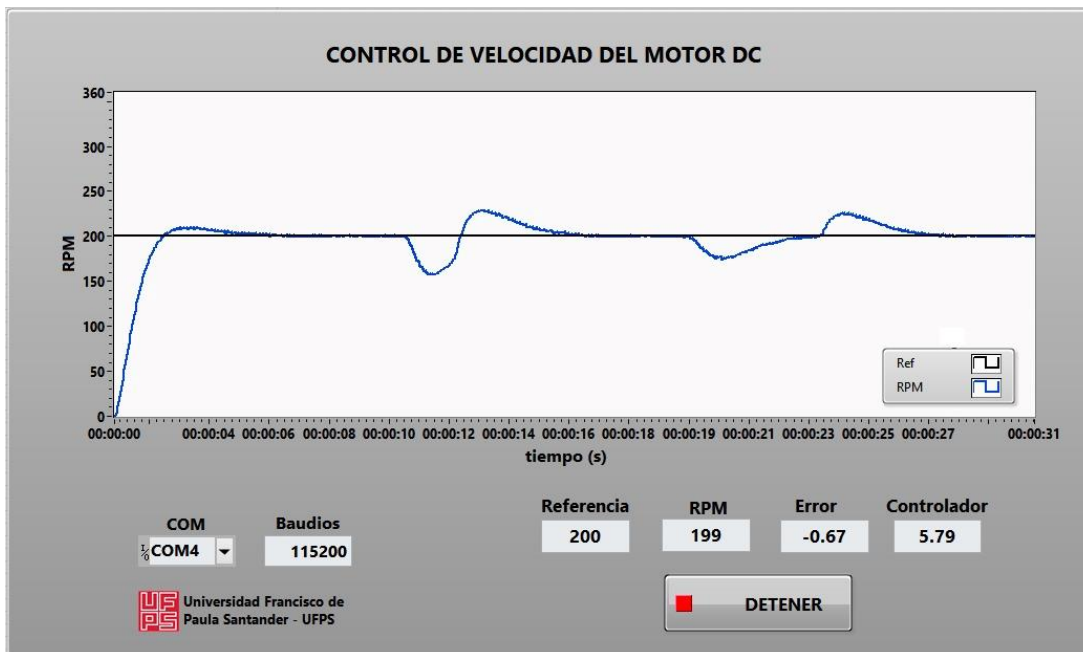
Respuesta del software de Arduino respecto al controlador integral derivativo por el método 1 de Ziegler y Nichols.



En la ilustración 51 se puede observar un sistema sub amortiguado, el cual cuenta con un tiempo pico de 0.02 segundos, un margen de sobrepaso de 1 Rpm, un error de 0.33 Rpm y un tiempo de establecimiento de 0.065 segundos luego de presentar perturbaciones.

Ilustración 54.

Respuesta del software Arduino respecto al controlador integral derivativo por medio del modelo matemático



En la ilustración 52 se puede observar un sistema sub amortiguado, el cual cuenta con un tiempo pico de 0.03 segundos, un margen de sobrepaso de 1 Rpm, un error de -0.67 Rpm y un tiempo de establecimiento de 0.0675 segundos luego de presentar perturbaciones.

9. Conclusiones

Se verificó que la precisión en la respuesta de un sistemas de control y su identificación son herramientas adecuadas para obtener el modelo lineal del motor; depende de la precisión con la que se obtenga la parametrización de dicho sistema se obtendrán las constantes que se utilizan en los modelos del motor; debido a esto es que se basa el funcionamiento real y particular de la máquina, y evita el cálculo de las constantes que no son necesarias al modelar mediante las ecuaciones diferenciales.

El manejo de softwares es muy habitual en el campo de la ingeniería de control, debido a estos se facilitan las experimentaciones y simulaciones las cuales deben ejecutarse de una manera continua hasta hallar el resultado que se asemeje a las condiciones iniciales suministradas por el operador. Matlab es una de las plataformas que más sobresale sobre los otros tipos de software teniendo como una ventaja la facilidad de interacción con su plataforma y ofrece diferentes herramientas dependiendo de las necesidades presentadas.

se comprobó la importancia de representar un sistema mediante su función de transferencia, debido a que a partir de la función se observa el comportamiento del mismo, y la conexión entre la salida y la entrada, partiendo de allí se realizaron los cálculos necesarios para hallar las constantes de los controladores convencionales (P, PI y PID) para proporcionar el control adecuado.

Los diferentes artículos utilizados muestran diferentes técnicas de control (P, PI, PID) principalmente para la velocidad de un motor de corriente continua. La técnica de control PID demostró una gran relevancia debido a varias ventajas presentadas, como se observa en la precisión de los tiempos de respuesta.

Los resultados de las pruebas experimentales y de simulación demuestran que el controlador proporcional propuesto no es viable debido a sus altos niveles de error, los cuales no le permiten llegar a valor referencia, partiendo de allí se observó el comportamiento de los controladores integrales y derivativos, siendo estos capaces de mantener la velocidad en el valor deseado con una precisión menor a 1%, un tiempo de establecimiento menor a 0.1 s y un sobrepaso menor a 11%.

10. Recomendaciones

Se sugiere alimentar el motor DC con una fuente externa del circuito eléctrico para minimizar los riesgos y evitar perturbaciones que afecten directamente al sistema de control o al microcontrolador a usar.

Se recomienda que la implementación del sistema de control sea mediante elementos completamente digitales, debido a que los analógicos pueden presentar mayor exposición a perturbaciones físicas y se implementaría un mayor número de elementos; Teniendo en cuenta lo anterior con los sistemas digitales se puede obtener una mayor precisión con un menor número de elementos empleados.

El uso de un motor de pequeña capacidad no presenta inconvenientes en cuestión a los picos de voltaje y corriente, en un trabajo futuro sería aconsejable la realización de un estudio similar al presentado, pero para motores de mayor potencia.

11. Referencias

- Barco, R. R. (Agosto de 2023). *Asociación Argentina de Matemáticas* . Obtenido de Discretización de Señales: <http://lcr.uns.edu.ar/fvc/NotasDeAplicacion/FVC-RodrigoBarco.pdf>
- Bautista Contreras, J., & Guevara Lidueñez, J. (2021). *Módulo didáctico para entrenamiento del control de velocidad para motores eléctricos*. Cucuta: Universidad Francisco de Paula Santander.
- Bolivar, H. (2023). *Ventajas y Desventajas Del Uso de Estrategias de Control Tipo P*. Obtenido de SCRIBD: <https://es.scribd.com/document/497178599/Ventajas-y-desventajas-del-uso-de-estrategias-de-control-tipo-P#>
- Ceron Morales , I., Vázquez Espinoza, J. O., & Aquino Díaz, E. (Febrero de 2017). *Modelo matematico del motor de corriente directa*. Obtenido de Researchgate: https://www.researchgate.net/publication/313215329_Modelo_matematico_del_motor_d_e_corriente_directa
- Chapman, S. J., & Roza Castillo, E. (2000). *Máquinas Eléctricas*. McGraw-Hill Education.
- CLR. (2023). *Motores de corriente continua y alterna: estudio y selección*. Obtenido de Compañía Levantada de Reductores: <https://clr.es/blog/es/motores-corriente-continua-alterna-seleccion/>
- coggle. (s.f.). *Acciones Basicas de Control*. Obtenido de Coggle: https://coggle.it/diagram/WAv23Cg_f-96JBrs/t/acciones-b%C3%A1sicas-de-control
- Contaval. (12 de Mayo de 2016). *Control Proporcional Integral Derivativo (PID)*. Obtenido de CONTAVAL: <https://www.contaval.es/pid-control-proporcional-integral-derivativo/>

Crespo, E. (23 de enero de 2017). *Aprendiendo Arduino*. Obtenido de Programación Arduino:

<https://aprendiendoarduino.wordpress.com/2017/01/23/programacion-arduino-5/>

Diaz, M. T. (2023). *Fisicotrónica*. Obtenido de Discretización. El salto cualitativo :

<http://fisicotronica.com/discretizacion-el-salto->

[cualitativo/#:~:text=Discretizaci%C3%B3n%20de%20tiempos%20para%20optimizar,po,sibles%20errores%20del%20espectro%20contin%C3%BAo.&text=Esto%20%C3%BAltimo%20tambi%C3%A9n%20es%20aplicable,paso%20de%20anal%C3%B3](http://fisicotronica.com/discretizacion-el-salto-cualitativo/#:~:text=Discretizaci%C3%B3n%20de%20tiempos%20para%20optimizar,po,sibles%20errores%20del%20espectro%20contin%C3%BAo.&text=Esto%20%C3%BAltimo%20tambi%C3%A9n%20es%20aplicable,paso%20de%20anal%C3%B3)

Encoder Products Company. (s.f.). *Imagen*. Obtenido de Encoder Products Company:

https://www.encoder.com/hs-fs/hubfs/articles/what-is-an-encoder/encoder-exploded-rendering-spanish_1080x608.jpg?width=1119&name=encoder-exploded-rendering-spanish_1080x608.jpg

Giraldo, S. A. (2019). *Control Automático Educación* . Obtenido de Todo sobre Ziegler y

Nichols - Sintonía de Control PID: <https://controlautomaticoeducacion.com/control-realimentado/ziegler-nichols-sintonia-de-control-pid/>

Giraldo, S. A. (2023). *Acción de Control Derivativo*. Obtenido de Control Automático

Educación: <https://controlautomaticoeducacion.com/control-realimentado/accion-de-control-derivativo-control-pid/>

Giraldo, S. A. (2023). *Acción de Control Integral* . Obtenido de Control Automático Educación :

<https://controlautomaticoeducacion.com/control-realimentado/accion-de-control-integral-control-pid/>

Giraldo, S. A. (2023). *Control PID- Acción de control proporcional*. Obtenido de Control

Automático Educación: <https://controlautomaticoeducacion.com/control-realimentado/control-pid-accion-proporcional/>

Lucid Software Inc. (2023). *Lucidchart* . Obtenido de Que es un diagrama de flujo:

<https://www.lucidchart.com/pages/es/que-es-un-diagrama-de-flujo>

MathWorks. (2023). *MATWORKS*. Obtenido de PID Tuner:

<https://la.mathworks.com/help/control/ref/pidtuner-app.html>

Mazzone, V. (Marzo de 2002). *Controladores PID*. Obtenido de PID:

<https://www.eng.newcastle.edu.au/~jhb519/teaching/caut1/Apuntes/PID.pdf>

MCI Electronics . (s.f.). *Arduino Mega 2560 R3*. Obtenido de MCI Electronics:

<https://mcielectronics.cl/shop/product/arduino-mega-2560-r3-arduino-10231/#:~:text=Arduino%20Mega%20es%20una%20tarjeta,implementa%20el%20lenguaje%20Processing%2FWiring>

MecatrónicaLATAM. (23 de Abril de 2021). *Motor de corriente continua (cc) o directa (DC)*.

Obtenido de MecatrónicaLATAM:

<https://www.mecatronicalatam.com/es/tutoriales/motor/motores-electricos/motor-de-corriente-continua/>

Modelado de un motor CC. (s.f.). Obtenido de Universidad de Oviedo:

<http://isa.uniovi.es/~idiaz/ADSTel/Practicas/ModeladoMotorCC.html>

Monte, J. C. (2023). *Normas para motores electricos GT*. Obtenido de ACADEMIA:

https://www.academia.edu/13253938/Normas_para_motores_el%C3%A9ctricos_gt

National Instruments Corp. (2023). *National Instruments*. Obtenido de Monitoree e Interactue con

su Prueba: <https://www.ni.com/es/shop/labview/monitor-interact-test.html>

pighixx. (13 de Marzo de 2013). *The Definitive Ardiono Mega Pinout Diagram*. Obtenido de

Arduino: https://cdn.shopify.com/s/files/1/0131/0792/0996/products/Arduino-MEGA-pinout_2048x2048.jpg?v=1584069019

R, J. L. (2021). *Como funciona un Encoder*. Obtenido de Como Funciona: <https://como-funciona.co/un-encoder/>

Sabogal Linares, J. A., & Vargas Ortega, G. A. (2013). *Control automático de velocidad para un motor DC mediante procesamiento de imagenes*. Obtenido de Repositorio Universidad Pontificia Bolivariana:

https://repository.upb.edu.co/bitstream/handle/20.500.11912/5900/digital_25129.pdf?sequence=1&isAllowed=y

Sanchez, A. B. (Agosto de 2012). *Control de Procesos Industriales II*. Obtenido de slideshare:

<https://es.slideshare.net/BlogsalDescubierto/cpi2-clase-4-parte-1-acciones-de-control>

Teoría de Control. (2020). Obtenido de Controlador PID:

<http://www3.fi.mdp.edu.ar/control4c7/APUNTES/Clase%207%20-%20PID.pdf>

Torres Niño , D. F., & Rivas Rengifo, H. (1996). *Aplicación de la norma IEC 34-1 en la sección, recepción, ensayo y mantenimiento de maquinas electricas rotatorias*. Obtenido de Corporacion Universitaria Autonoma del Occidente :

<https://red.uao.edu.co/bitstream/handle/10614/2640/T0000916.pdf?sequence=1&isAllowed=y>

Universidad de Cantabria. (2023). *Servicio de Informatica* . Obtenido de LabVIEW:

<https://sdei.unican.es/Paginas/servicios/software/Labview.aspx#:~:text=Descripci%C3%B3n,de%20otras%20herramientas%20de%20desarrollo.>

Universidad de Cantabria. (2023). *Sistema de Informatica* . Obtenido de LabVIEW :

<https://sdei.unican.es/Paginas/servicios/software/Labview.aspx#:~:text=Descripci%C3%B3n,de%20otras%20herramientas%20de%20desarrollo.>

Villajulca, J. C. (11 de Marzo de 2010). *Métodos de control de velocidad en motores AC*.

Obtenido de Instrumentacionycontrol.net: <https://instrumentacionycontrol.net/metodos-de-control-de-velocidad-en-motores-ac/>

Villajulca, J. C. (28 de Agosto de 2018). *Control derivativo (Rate): el lado prudente del controlador*. Obtenido de Instrumentacionycontrol.net:

<https://instrumentacionycontrol.net/control-derivativo-rate-el-lado-prudente-del-controlador/>

Villajulca, J. C. (11 de Febrero de 2019). *Introducción a la sintonización PRACTICA de controladores: conociendo a los procesos PRIMERO*. Obtenido de

Instrumentacionycontrol.net: <https://instrumentacionycontrol.net/introduccion-a-la-sintonizacion-de-controladores-conociendo-a-los-procesos-primero/>

Villajulca, J. C. (02 de Junio de 2022). *El control proporcional: Definiciones prácticas y*

precisas. Obtenido de Instrumentacionycontrol.net: <https://instrumentacionycontrol.net/el-control-proporcional-definiciones-practicas-y-precisas/>

Virtualexpo Group. (2023). *Qué motor eléctrico elegir*. Obtenido de DIRECT INDUSTRY :

<https://guide.directindustry.com/es/que-motor-electrico-elegir/#2>

Wikipedia. (02 de septiembre de 2023). *Wikipedia a enciclopedia libre*. Obtenido de Interfaz

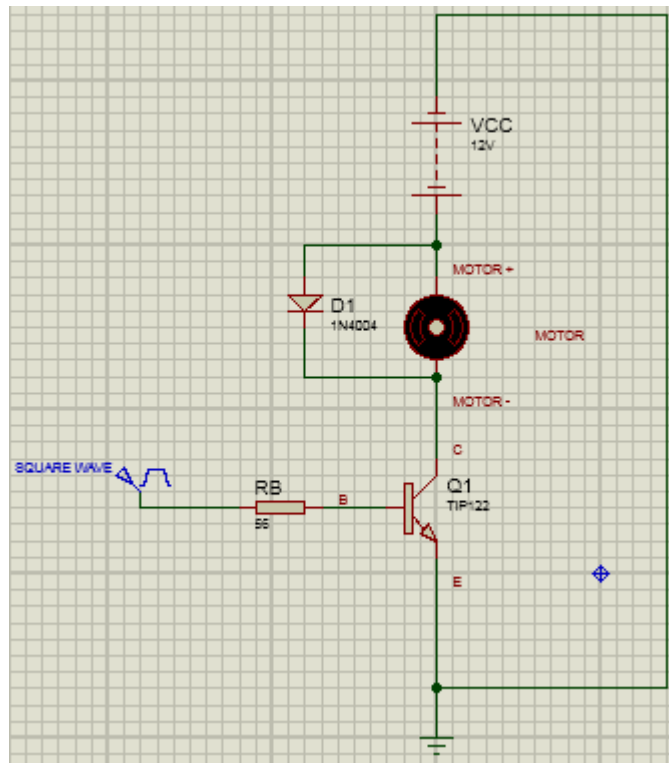
grafica de usuario.: https://es.wikipedia.org/wiki/Interfaz_gr%C3%A1fica_de_usuario

Anexos

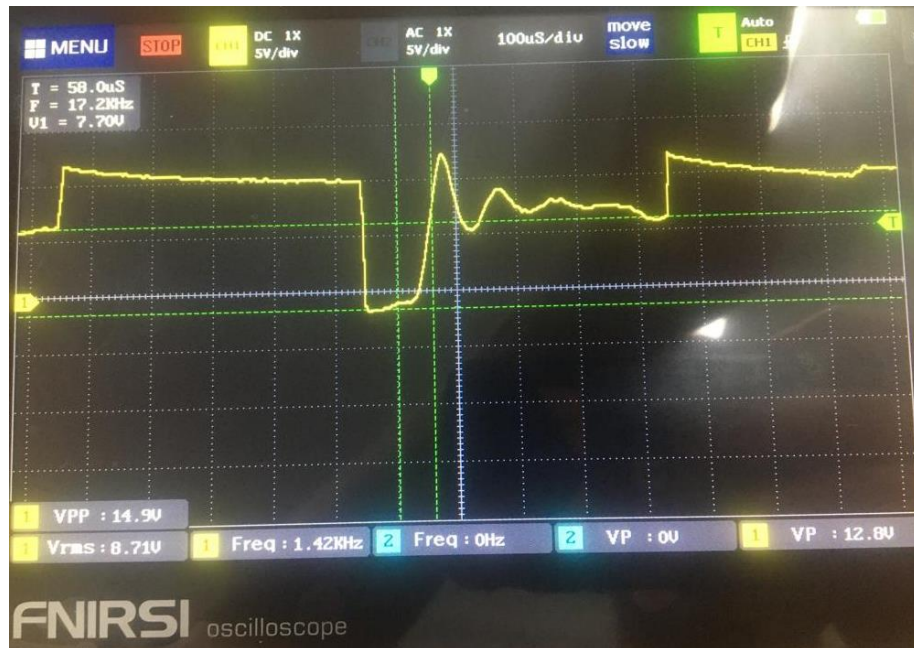
Anexo A. Diagrama de conexiones implementado para la obtención de los parámetros requeridos para determinar la constante contra electromotriz (K_a).



Anexo B. Circuito implementado para hallar la variable T_m del motor Dc.



Anexo C. Respuesta obtenida de la señal PWM de 9.99v mediante el circuito implementado en el anexo B.



Anexo D. Respuesta obtenida de la señal PWM de 5,19 V mediante el circuito implementado en el anexo B.



Anexo E. *Diagrama de conexiones para obtención y determinación del coeficiente de fricción (B)*



Anexo F. *Diagrama de conexiones Implementado para la obtención de los parámetros requeridos para determinar el torque de fricción*

