

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DISEÑO, SIMULACIÓN E IMPLEMENTACIÓN DE
CONTROLADORES PARA LAS PLANTAS DIDÁCTICAS DEL
LABORATORIO DE CONTROL Y SISTEMAS**

**DISEÑO, SIMULACIÓN E IMPLEMENTACIÓN DE
CONTROLADORES PARA LA PLANTA DIDÁCTICA DEL
LABORATORIO DE CONTROL Y SISTEMAS QNET DC MOTOR
CONTROL TRAINER**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
ELECTRÓNICA Y AUTOMATIZACIÓN**

EDGAR ROLANDO CASCO MILLINGALLI

edgar.casco@epn.edu.ec

DIRECTOR: Ing. YADIRA LUCIA BRAVO NARVÁEZ MSc.

yadira.bravo@epn.edu.ec

DMQ, Octubre 2022

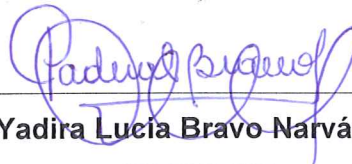
CERTIFICACIONES

Yo, Edgar Rolando Casco Millingalli declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



Edgar Rolando Casco Millingalli

Certifico que el presente trabajo de integración curricular fue desarrollado por Edgar Rolando Casco Millingalli, bajo mi supervisión.



Ing. Yadira Lucía Bravo Narváez, MSc.

DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

EDGAR ROLANDO CASCO MILLINGALLI

Ing. YADIRA LUCIA BRAVO NARVÁEZ MSc.

DEDICATORIA

El trabajo realizado está dedicado a mis padres Manuel y Dolores, que gracias a su esfuerzo y apoyo incondicional me fue posible crecer como persona y profesionalmente.

AGRADECIMIENTO

A mis padres Manuel y Dolores, que me guiaron, aconsejaron y apoyaron durante el transcurso de mi carrera universitaria inculcándome valores de humildad, honestidad y respeto.

A mis compañeros de carrera con los que pasamos gratos momentos y situaciones difíciles en la universidad, especialmente a Samantha quien más que una compañera fue una amiga y sin su apoyo incondicional no fuera sido posible la culminación de este trabajo.

A mi tutora MSc. Yadira Bravo y el MSc. Marco Herrera por su ayuda e interés mostrado durante el desarrollo del Trabajo de Integración Curricular.

Y finalmente a mis hermanos, quienes me han apoyado en mí día a día.

ÍNDICE DE CONTENIDO

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN.....	VII
ABSTRACT.....	VIII
1. INTRODUCCIÓN.....	1
1.1 OBJETIVO GENERAL	2
1.2 OBJETIVOS ESPECÍFICOS	2
1.3 ALCANCE	2
1.4 MARCO TEÓRICO	3
1.4.1 QNET DC MOTOR CONTROL TRAINER.....	3
1.4.2 COMPONENTES DE LA DC MOTOR CONTROL TRAINER.....	3
1.4.2.1 Motor DC	4
1.4.2.2 Amplificador de potencia con PWM	4
1.4.2.3 Sensor de corriente	4
1.4.2.4 Sensor de posición.	4
1.4.2.5 Sensor de velocidad.....	4
1.4.2.6 Fusible.....	5
1.4.2.7 Fuente de alimentación.....	5
1.4.3 ESQUEMA DEL SISTEMA QNET-DCMCT.....	5
1.4.4 ESPECIFICACIONES TÉCNICAS DEL SISTEMA DCMCT.....	5
1.4.5 MODELADO DEL SISTEMA QNET DCMCT.....	6
1.4.5.1 Modelado teórico.....	6
1.4.5.1.1 Ley de mallas de Kichhoff.....	7
1.4.5.1.2 Segunda Ley de Newton.....	7
1.4.5.1.3 Funciones de transferencia del motor DC.....	8
1.4.5.1.4 Representación en Espacio de Estados.....	9
1.4.5.2 Identificación de sistemas.....	10
1.4.5.2.1 Método a lazo abierto o curva de reacción.....	10
1.4.6 CONTROLADORES PID.....	12

1.4.6.1 Sintonización del controlador mediante el método de Dahlin.....	13
1.4.7 CONTROL POR REALIMENTACIÓN DE ESTADOS.....	13
1.4.8 APP DESIGNER DE MATLAB.....	14
1.4.8.1 Componentes de app designer.....	15
1.4.9 LABVIEW.....	16
1.4.9.1 Características principales.....	16
2. METODOLOGÍA.....	17
2.1 MODELO MATEMÁTICO DEL SISTEMA QNET-DCMCT.....	17
2.1.1 MODELO ANALÍTICO.....	17
2.1.2 MODELO EXPERIMENTAL O IDENTIFICACIÓN.....	18
2.2 DISEÑO CONTROLADOR PID.....	20
2.2.1 CONTROLADOR DE VELOCIDAD.....	20
2.2.2 CONTROLADOR DE POSICIÓN.....	21
2.3 DISEÑO DEL CONTROLADOR POR REALIMENTACIÓN DE ESTADO CON OBSERVADOR DE ORDEN COMPLETO.....	23
2.3.1 CONTROL DE LA VELOCIDAD.....	24
2.3.2 CONTROL DE LA POSICIÓN.....	26
2.4 DISEÑO DEL LA INTERFAZ GRÁFICA EN APP DESIGNER.....	26
2.5 DISEÑO DEL LA INTERFAZ GRÁFICA EN LABVIEW.....	28
3. RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	31
3.1 VALIDACIÓN DEL MODELO DEL SISTEMA QNET DCMCT.....	32
3.2 CONTROL DE VELOCIDAD EN APP DESIGNER.....	33
3.3 CONTROL DE POSICIÓN EN APP DESIGNER.....	35
3.4 CONTROL DE VELOCIDAD EN LabVIEW.....	37
3.5 CONTROL DE POSICIÓN EN LabVIEW.....	41
3.6 CONCLUSIONES.....	46
3.7 RECOMENDACIONES.....	47
4. REFERENCIAS BIBLIOGRÁFICAS.....	48
5. ANEXOS.....	50
ANEXO I.....	51
ANEXO II.....	59
ANEXO III.....	62

RESUMEN

En el presente trabajo se realiza el diseño, simulación e implementación de controladores PID y controladores en Espacio de Estados para la planta didáctica QNET DC Motor Control Trainer donde se controla tanto la posición y la velocidad. La adquisición de datos se la realiza mediante la tarjeta NI ELVIS II y proporciona mediciones de posición y corriente, las variables velocidad y voltaje de armadura son calculadas matemáticamente para su posterior uso en el controlador por Espacio de Estado. La prueba de funcionamiento de los controladores se realizan haciendo uso del software Matlab/Simulink con el modelo matemático del sistema y posteriormente se realizan pruebas con el sistema físico utilizando el software LabVIEW.

Los controladores diseñados fueron sometidos a cambios de referencia y pruebas ante perturbaciones no modeladas en la señal de control y salida del sistema, para observar el comportamiento que cada uno de los controladores presenta ante cada una de las pruebas mencionadas. De las pruebas realizadas se obtuvieron los índices de desempeño ISE, ISU y tiempo de establecimiento tanto para pruebas físicas y a nivel de simulación para realizar un análisis comparativo entre los controladores diseñados dependiendo de la variable a controlar.

PALABRAS CLAVE: Espacio de Estado, ISE, ISU, motor DC, posición, velocidad, perturbaciones.

ABSTRACT

In the present work, the design, simulation and implementation of PID controllers and State Space controllers for the QNET DC Motor Control Trainer didactic plant where both position and speed are controlled is carried out. The data acquisition is done through the NI ELVIS II card and provides position and current measurement, the speed and armature voltage variables are mathematically calculated for later use in the state space controller. The operation test of the controllers is carried out using the Matlab/Simulink software with the mathematical model of the system and later tests are carried out with the physical system using the LabVIEW software.

The controllers designed were subjected to changes of reference and tests against disturbances not modeled in the control signal and output of the system, to observe the behavior that each of the controllers presents before each of the mentioned tests. From the tests carried out, the performance indices ISE, ISU and establishment time were obtained for both physical tests and a simulation level to carry out a comparative analysis between the controllers designed according to the variable to be controlled.

KEY WORDS: State space, ISE, ISU, DC motor, position, velocity, disturbances.

1. INTRODUCCIÓN

En los últimos dos años debido a los efectos de la pandemia cambió la modalidad de estudios a virtual, debido a esto se tiene la necesidad de desarrollar plataformas que permitan al estudiante tener experiencias e interactuar con los sistemas de hardware QNET disponibles en el laboratorio de Sistemas de Control. Existen plataformas virtuales que trabajan con los sistemas de hardware anteriormente mencionados, de la marca QUANSER, pero estos tienen un costo elevado.

El QNET DC Motor Control Trainer es un sistema que consiste en un motor DC, un encoder, electrónica de control y una tarjeta de adquisición de datos, cada uno de los componentes que la integran cumplen funciones específicas; donde la velocidad o la posición del motor DC son las salidas del sistema, el encoder medirá la posición del motor y junto con la DAQ cerraran el lazo de control, por último, la electrónica de control enviará la señal de control.

Las salidas de velocidad o posición permiten el diseño y la implementación de varios controladores, generalmente un motor DC tiene un control PWM en su mayoría se usan controladores PID que calculan el ciclo de trabajo del PWM dado que la variable de control es el nivel de voltaje que alimenta el motor DC. El QNET DC motor control trainer al ser un sistema físico está expuesto a perturbaciones que pueden ser manuales o digitales.

El modelo matemático del QNET DC motor control trainer se puede modelar con una función de transferencia de primer orden, con la cual se realizará el diseño de los controladores para cada una de las salidas requeridas. En salida de posición el encoder se encarga de proporcionar la realimentación de la posición real del motor DC, al no tener un sensor para medir la salida de velocidad del sistema está se calcula matemáticamente usando la derivada de la posición proporcionada, adicionalmente se cuenta con un filtro de primer orden pasa bajos que disminuirá el error de cálculo y así tener una medición más exacta.

Por todo esto, se plantea realizar una interfaz gráfica tanto en la App Designer de Matlab y en LabVIEW que permitan a los estudiantes de la asignatura de Sistemas de Control diseñar, simular e implementar los controladores de la planta didáctica QNET DC Motor Control Trainer, donde la interfaz en App Designer les permita simular los controladores diseñados y las pruebas de funcionamiento con el sistema físico las realicen con la interfaz gráfica desarrollada en LabVIEW.

1.1 OBJETIVO GENERAL

Diseñar, simular e implementar dos controladores, controlador PID y controlador en Espacio de Estados para la planta didáctica QNET DC motor control trainer.

1.2 OBJETIVOS ESPECÍFICOS

1. Determinar el modelo matemático que representa a la planta estudiada,
2. Diseñar un controlador tipo PID y un controlador en Espacio de Estado.
3. Desarrollar una interfaz gráfica con las ventanas necesarias para poder seleccionar el controlador y visualizar los datos en pantalla.
4. Realizar pruebas de funcionamiento de la interfaz, los controladores diseñados y el sistema físico.

1.3 ALCANCE

Se realizará una revisión bibliográfica de modelos matemáticos para la planta didáctica QNET DC Motor Control Trainer.

Se realizará una revisión bibliográfica sobre el diseño de controladores tipo PID y control en Espacio de Estado.

Se diseñará, simulará e implementará un controlador PID y un controlador en Espacio de Estado para controlar el sistema estudiado.

Se desarrollará una interfaz gráfica en App Designer de Matlab donde el estudiante pueda identificar el modelo matemático del sistema físico, así como la respuesta y desempeño de los dos controladores diseñados. Adicionalmente se tiene que la interfaz no contara con animaciones en 2D o 3D del sistema QNET DC Motor Control Trainer.

Se realizará pruebas de funcionamiento mediante una interfaz en LabVIEW donde se comparará los controladores diseñados y el funcionamiento del sistema en la interfaz desarrollada con el sistema físico disponible en el laboratorio, realizando una cuantificación por índices de desempeño.

1.4 MARCO TEÓRICO

1.4.1 QNET DC MOTOR CONTROL TRAINER (QNET DCMCT)

QNET DCMCT es un equipo versátil utilizado para la enseñanza y demostración de los fundamentos de control de los motores DC, tales como, el control PID, control en Espacio de Estado, entre otros. Esto mediante el uso de una computadora capaz de realizar control en tiempo real y el NI ELVIS II. El DC Motor Control Trainer puede configurarse fácilmente para controlar la velocidad y la posición del motor. El sistema consiste en un motor DC con un encoder y una rueda de inercia en el eje del motor [1].

1.4.2 COMPONENTES DE LA DC MOTOR CONTROL TRAINER

Los componentes de la QNET DC Motor Control Trainer se indican en la Figura 1.1 y en la Figura 1.2 y se describen en la Tabla 1.1.

Tabla 1.1. Nomenclatura de los componentes de DCMCT [2].

ID #	Descripción	ID #	Descripción
1	Motor DC	6	Placa QNET PWM/Encoder
2	Codificador de alta resolución	7	Toma de alimentación QNET de 24 V
3	Cámara metálica del motor	8	Fusible
4	Carga de inercia	9	LEDs +B, +15 V, -15 V, +5 V
5	Conector PCI a NI ELVIS: para interconectar el módulo QNET con el DAC.		



Figura 1.1. Componentes del motor QNET DC [2].

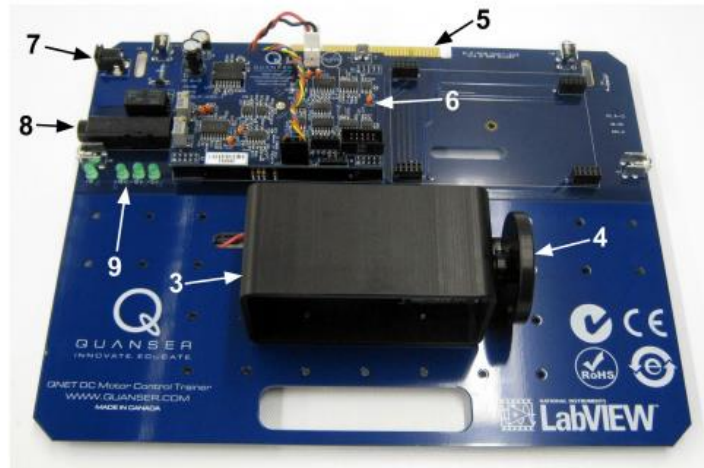


Figura 1.2. Diseño general del QNET DCMCT [2].

1.4.2.1 Motor DC

El motor DC es de 12 V conformado por 5 segmentos de conmutadores, 64 devanados por polo y un anillo de flujo. Alcanza una velocidad constante de 286 rpm/V [2].

1.4.2.2 Amplificador de potencia con PWM

Este componente es el encargado de accionar el motor DC. La entrada del amplificador es la salida del convertidor digital-analógico del canal #0 de la DAQ. La tensión máxima que puede entregar el amplificador es de 24 V, su corriente pico máxima es de 5 A y la corriente máxima continua es de 4 A. Su ganancia es de 2.3 V/V [2].

1.4.2.3 Sensor de corriente

La medición analógica de corriente se realiza mediante una resistencia de detección de corriente de 0.1 ohm en serie con la salida del amplificador PWM, internamente la señal del sensor es amplificada dando lugar a una sensibilidad de 1.0 V/A. Esta señal está disponible en el convertidor análogo-digital del canal #0 [2].

1.4.2.4 Sensor de posición.

La medición digital de posición se la realiza mediante un codificador óptico de alta resolución en cuadratura, el cual es montado directamente en la parte trasera del motor DC. Esta señal está disponible en el canal #0 de la entrada digital de la DAQ [2].

1.4.2.5 Sensor de velocidad

El sistema no cuenta con un sensor de velocidad. En su lugar, la placa cuenta con un circuito que calcula la derivada de la señal del encoder, es decir tiene un tacómetro digital. Sin embargo, para disminuir el ruido en la medición de esta señal y aumentar la robustez del sistema se utiliza un filtro pasa bajo de primer orden [2].

1.4.2.6 Fusible

El fusible del amplificador de potencia QNET es de 250 V y 3 A [2].

1.4.2.7 Fuente de alimentación

El DCMCT tiene una entrada de alimentación de 24 V y la fuente debe suministrar como máximo 3 A. El LED +B de la palca QNET se ilumina en verde cuando se está alimentando al amplificador[2].

1.4.3 ESQUEMA DEL SISTEMA QNET-DCMCT

En la Figura 1.3 se indica el esquema del sistema QNET DC Motor Control Trainer interconectado con un dispositivo de adquisición de datos (DAQ).

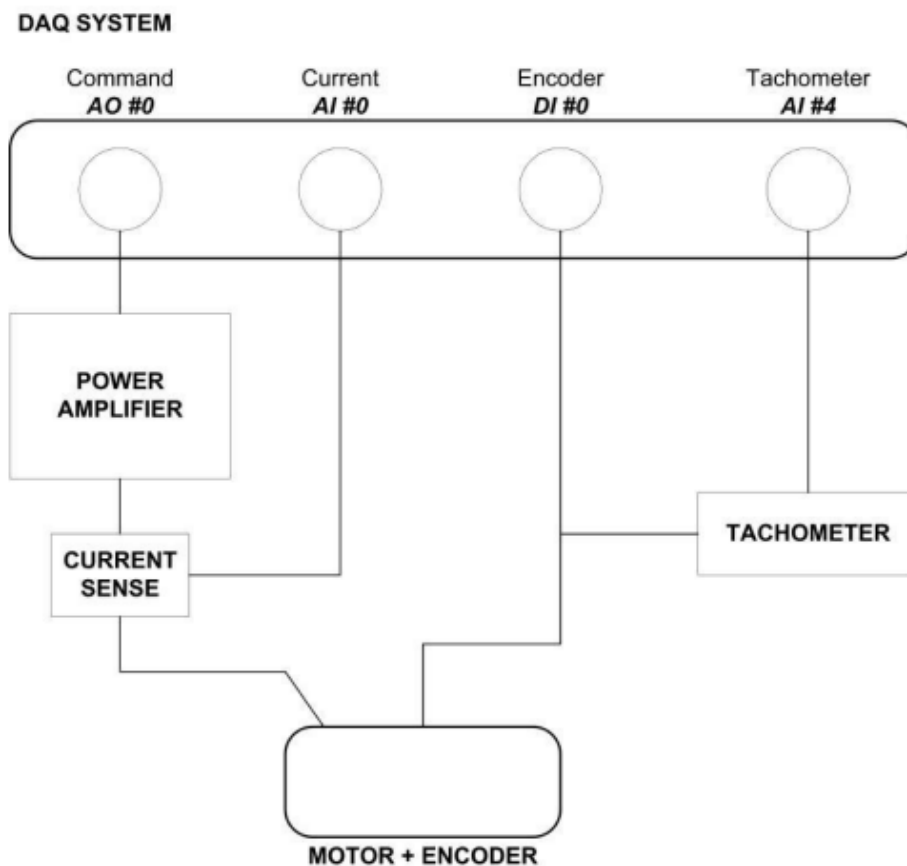


Figura 1.3. Esquema del sistema QNET-DCMCT [2].

1.4.4 ESPECIFICACIONES TÉCNICAS DEL SISTEMA DCMCT

Los parámetros para el modelo estudiado DC Motor Control Trainer fueron tomadas del Manual de Usuario y se exponen en la Tabla 1.2.

Tabla 1.2. Especificaciones del motor y el amplificador de ancho de pulso modulado [2].

Símbolo	Descripción	Valor	Unidad
Motor			
Rm	Resistencia del inducido del motor	8.7	Ohm
Kt	Constante de corriente-par del motor	0.03334	N-m
Km	Constante de contrafase del motor	0.03334	V/(rad/s)
Jm	Momento de inercia del rotor del motor	1.80x10-6	Kg-m2
Bm	Coeficiente de fricción	1x10-5	-
Lm	Inductancia	8.3x10-3	H
	Par máximo continuo	0.10	N-m
	Potencia máxima	20.0	W
	Corriente continua máxima	1.0	A
MI	Masa del disco de carga	0.033	Kg
RI	Radio de disco de carga	0.0242	M
Amplificador de ancho de pulso modulado			
Vmax	Tensión máxima de salida del amplificador PWM	24	V
	Corriente de salida máxima del amplificador PWM	5	A
	Ganancia del amplificador PWM	2.3	V/V

1.4.5 MODELADO DEL SISTEMA QNET DCMCT

El modelado de un sistema proporciona un bosquejo del sistema real con cierto grado de precisión y manteniendo el mayor grado de complejidad posible. Los métodos más utilizados se describen a continuación:

1.4.5.1 Modelado teórico

Es un método analítico, el cual utiliza las leyes básicas de la física para detallar el comportamiento dinámico de un proceso o fenómeno [3].

El sistema QNET DCMCT es descrito mediante las ecuaciones eléctricas y mecánicas de un motor DC. Según Chapman [4] “el circuito del inducido se representa por medio de una fuente de voltaje ideal E_a y un resistor R_a . Esta representación es en realidad el equivalente de Thevenin de toda la estructura del rotor, la cual incluye las bobinas, interpolos y devanados de compensación, sí que estos existen. La caída de voltaje en las escobillas se representa por medio de una pequeña batería V_{escob} en sentido opuesto a la dirección de flujo de corriente en la máquina” [4].

Existen diversas variaciones y simplificaciones del circuito equivalente del motor DC. Generalmente no considera la caída de voltaje en las escobillas o puede ser incluida en el valor de R . Además, la resistencia de las bobinas de campo se agrupa con el resistor de variable teniendo así la resistencia R_f . Con todas las consideraciones mencionadas en la Figura 1.4 se tiene el circuito equivalente de un motor DC [4].

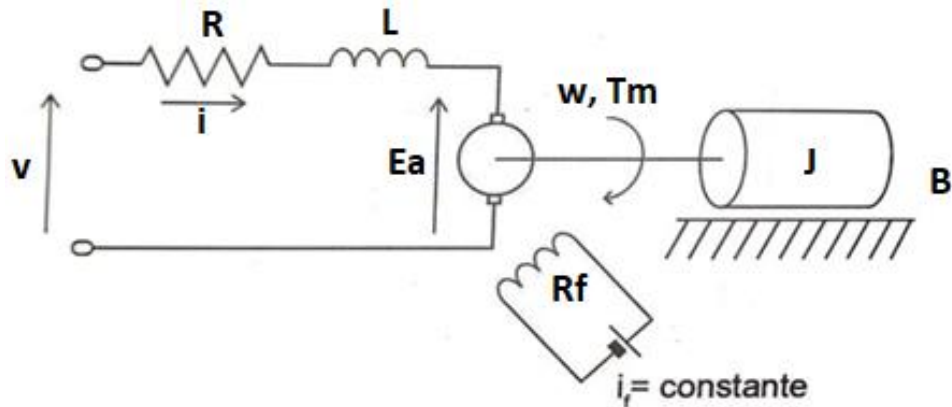


Figura 1.4. Elementos más importantes de un motor DC [5].

1.4.5.1.1 Ley de mallas de Kichhoff

Liwschitz-Garik [6] indica que, “en cada malla de una red, la suma de todas las fems (fuerza contra electromotriz) aplicadas e inducidas tomadas en consideración con los signos adecuados es igual a la suma de todas las caídas de tensión resistivas”.

$$\Sigma(V + E) = \Sigma(I * R) \quad (1.1)$$

Aplicando la Ley de Kirchhoff al circuito eléctrico de la Figura 1.4 se obtiene la siguiente ecuación:

$$v(t) - E_a(t) = R * i(t) + L \frac{di(t)}{dt} \quad (1.2)$$

Donde $E_a(t)$ es la fuerza contraelectromotriz, $i(t)$ y $v(t)$ son valores momentáneos de la corriente y voltaje aplicadas respectivamente, R y L la resistencia e inductancia de la armadura respectivamente.

1.4.5.1.2 Segunda Ley de Newton

Ignacio Martín Bragado [7] establece que, “es la ley más importante en cuanto permite establecer una relación numérica entre las magnitudes fuerza y aceleración”. Se podía enunciar como “La aceleración que toma un cuerpo es proporcional a la fuerza neta externa que se le aplica”.

En movimiento rotacional se tiene un análogo para la segunda ley de Newton [4], “Un torque neto sobre un objeto que tiene un punto fijo de rotación ocasiona una aceleración angular sobre él, la cual es inversamente proporcional a cierta cantidad J ”. Las pérdidas por fricción y parte de la energía desarrollada son almacenadas como energía cinética, teniendo así la ecuación de la sección mecánica del motor DC [4].

$$T_m(t) = J * \frac{d\omega(t)}{dt} + B * \omega(t) \quad (1.3)$$

Donde $T_m(t)$ es el torque del motor DC, B es el coeficiente de fricción y J es el momento de inercia total del rotor y la carga, $\omega(t)$ es la velocidad angular y $\frac{d\omega(t)}{dt}$ es la aceleración angular del motor.

La relación proporcional entre la constante contra electromotriz K_a [V/rad s] y el voltaje inducido en la armadura y la velocidad angular del eje del motor se indica en la ecuación 1.4. [4].

$$E_a(t) = K_a * \omega(t) \quad (1.4)$$

Además, se tiene que el rotor realiza su movimiento por acción del torque electromagnético T_m generado por el campo magnético producido en el estator, el valor del mismo depende de la corriente que circula a través de la armadura, teniendo así la siguiente relación [4].

$$T_m = K_m * i(t) \quad (1.5)$$

1.4.5.1.3 Funciones de transferencia del motor DC

Para determinar la función de transferencia del sistema estudiado se utiliza la herramienta matemática de la transformada de Laplace, ya que permite realizar una representación del comportamiento estacionario y dinámico de cualquier sistema [8]. Al aplicar la transformada de Laplace en las ecuaciones 1.2, 1.3, 1.4 y 1.5 se obtiene lo siguiente:

$$V(s) = R * I(s) + L * sI(s) + E_a(s) \quad (1.6)$$

$$T_m(s) = J * s * \omega(s) + B * \omega(s) \quad (1.7)$$

$$E_a(s) = K_a * \omega(s) \quad (1.8)$$

$$T_m(s) = K_m * I(s) \quad (1.9)$$

Sustituyendo las ecuaciones 1.7, 1.8 y 1.9 en la ecuación 6 se determina la función de transferencia que relaciona la velocidad angular con la entrada de voltaje (ecuación 1.10). Para determinar la posición angular del motor DC se integra la velocidad angular (ecuación 1.11).

$$\frac{\omega(s)}{V(s)} = \frac{K_m}{LJs^2 + (RJ + LB)s + (K_aK_m + BR)} \quad (1.10)$$

$$\frac{\theta(s)}{V(s)} = \frac{K_m}{s(LJs^2 + (RJ + LB)s + (K_a K_m + BR))} \quad (1.11)$$

1.4.5.1.4 Representación en Espacio de Estados

A diferencia de la teoría clásica la representación se la realiza mediante el concepto de estado, donde se entiende como variable de estado a la característica más pequeña que define el comportamiento dinámico del sistema a controlar[8]. Partiendo de las ecuaciones 1.6 a la 1.9 se define como variables de estado a la posición angular, velocidad angular y a la corriente de armadura.

$$x_1 = \theta(t)$$

$$x_2 = \omega(t)$$

$$x_3 = i(t)$$

$$u = v(t)$$

La representación en Espacio de Estados del modelo del motor es:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{B}{J} & \frac{K_m}{J} \\ 0 & -\frac{K_a}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} u$$

$$y = [1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u$$

La variable posición para el control de velocidad resulta innecesaria, por lo que se plantea una nueva representación en Espacio de Estados:

$$x_1 = \omega(t)$$

$$x_2 = i(t)$$

$$u = v(t)$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{B}{J} & \frac{K_m}{J} \\ -\frac{K_a}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u$$

$$y = [1 \quad 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

1.4.5.2 Identificación de sistemas

Es un método práctico que usa los datos reales recogidos de la planta para obtener el modelo del sistema. A continuación, se detalla el método más utilizado.

1.4.5.2.1 Método a lazo abierto o curva de reacción

Esta técnica se realiza en lazo abierto, es decir, colocando el controlador en modo manual, es decir, se envía la señal de control directo al sistema sin ningún lazo de control. El método consiste en realizar la prueba de escalón, obteniendo una curva de reacción a la salida, la curva obtenida se puede modelar a un sistema de primer o segundo orden, dependiendo del sistema se puede determinar los parámetros de ganancia (K), constante(s) de tiempo (τ, τ_1, \dots) y tiempo muerto (t_m). Para determinar las constantes anteriormente mencionadas se sigue el siguiente procedimiento [9]:

- Abrir el lazo de control y poner en modo manual.
- Hacer un cambio escalón del 10% del valor máximo de la señal de control.
- Registrar la curva de reacción del proceso.

Generalmente para procesos donde se aplica este método se obtiene una función de transferencia de primer orden:

$$\frac{X(s)}{U(s)} = \frac{K e^{-t_m s}}{\tau s + 1} \quad (1.12)$$

O de forma general:

$$\frac{X(s)}{U(s)} = \frac{K e^{-t_m s}}{(\tau_1 s + 1)(\tau_2 s + 1) \cdots (\tau_n s + 1)} \quad (1.13)$$

Pero generalmente se usa el modelo de primer orden para plantas donde se tenga una curva de reacción sin máximo sobre impulso o respuesta inversa. En la Figura 1.5 y Figura 1.6 se indica la forma de obtener los dos puntos que servirán para realizar el cálculo de los parámetros de modelo de primer orden.

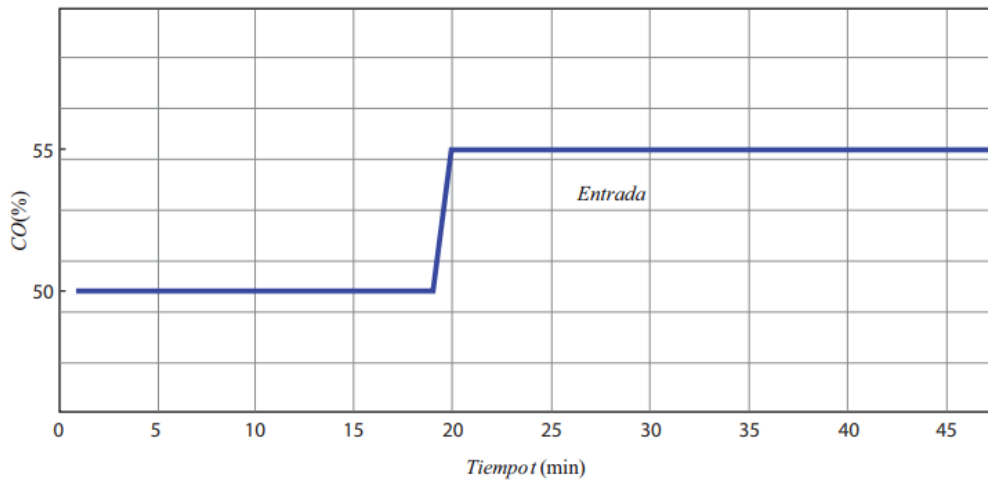


Figura 1.5. Entrada paso con un cambio del 10% en la señal de control [9].

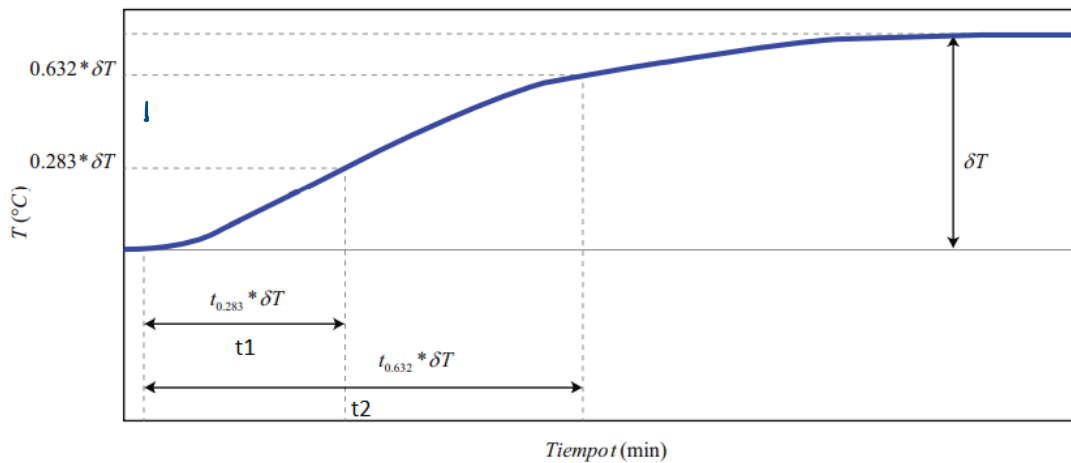


Figura 1.6. Curva de reacción del proceso usando el método de los dos puntos de Smith [9].

Uno de los métodos más utilizados para calcular las constantes de la función de transferencia es el método de dos puntos de Smith [10], el cual consiste en obtener datos de dos puntos de la curva de reacción, el primero es el tiempo que le toma al proceso alcanzar el 63.2% del valor en estado estable al cual le denominaremos t_2 , y el segundo punto es el tiempo en alcanzar el 28.3% del valor en estado estable, al cual denominaremos t_1 [10]. Las ecuaciones proporcionadas por Smith para el cálculo de la constante de tiempo (τ) y el tiempo muerto (t_m) son las que se indican a continuación:

$$\tau = at_1 + bt_2 \quad (1.14)$$

$$t_m = ct_1 + dt_1 \quad (1.15)$$

Alfaro [10] proporciona una tabla con las constantes para cada uno de los métodos de identificación de proceso sobre amortiguados, para el método de Smith las constantes a, b, c y d tiene el valor de -1.5, 1.5, 1.5 y -0.5 respectivamente.

Con los datos obtenidos tanto para los tiempos t_1 y t_2 se calcula la constante de tiempo y el tiempo muerto. Para el cálculo de la ganancia K se divide el valor en estado estable de la salida para el valor de la entrada paso, tal como se muestra en las siguientes ecuaciones:

$$K = \frac{\Delta\delta}{\Delta U} \quad (1.16)$$

$$\tau = 1.5(t_2 - t_1) \quad (1.17)$$

$$t_m = t_2 - \tau \quad (1.18)$$

1.4.6 CONTROLADORES PID

El controlador PID es un control a lazo cerrado cuyo objetivo es hacer que el error en estado estacionario sea nulo. Los controladores PID son mayormente utilizados en aplicaciones industriales abarcando casi un 95% del total de los lazos de control, siendo los controladores PI los más utilizados [11].

El principio de realimentación puede ser expresado como: “Incrementar la variable manipulada cuando la variable del proceso sea más pequeña que la referencia y disminuirla cuando esta sea más grande”. La realimentación es negativa, dado que la variable manipulada se mueve en dirección contraria a la variable del proceso. El esquema general de un control PID se muestra en la Figura 1.7 [12].

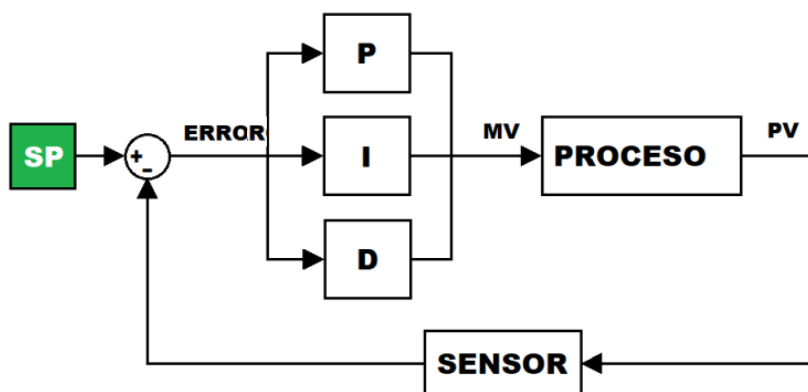


Figura 1.7. Control PID [12].

La estructura básica del controlador PID viene dada en la ecuación 1.19:

$$u(t) = k_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (1.19)$$

Donde $u(t)$ es la variable de control, $e(t)$ es el error existente entre la referencia y la salida del proceso. El modelo del sistema de primer orden permite la sintonización del controlador PID, para lo cual se aplica las ecuaciones propuestas por Dahlin para determinar los parámetros de ajuste del controlador.

1.4.6.1 Sintonización del controlador mediante el método de Dahlin

Las expresiones para realizar el cálculo de los parámetros del controlador PID propuestos por Dahlin se indican en la Tabla 1.3; si se requiere un controlador PD, PI u otra combinación únicamente se calcula el parámetro de ajuste necesario dependiendo del tipo de controlador a utilizar.

Tabla 1.3. Parámetros de sintonización de un controlador usando el método de Dahlin

Controlador	Parámetro de ajuste	Ecuación
Proporcional + Integral + Derivativo, PID	k_p	$\frac{1}{2K} \left(\frac{t_0}{\tau} \right)^{-1}$
	T_i	τ
	T_d	$\frac{t_0}{2}$

1.4.7 CONTROL POR REALIMENTACIÓN DE ESTADOS

En el control por variables de estado se supone que todas las variables del sistema son medibles y se tiene acceso total a las mismas. En la Figura 1.8 se indica el esquema de control por realimentación de estados, el sistema de control por realimentación de estados consta con un proceso dinámico, que es considerado lineal [8].

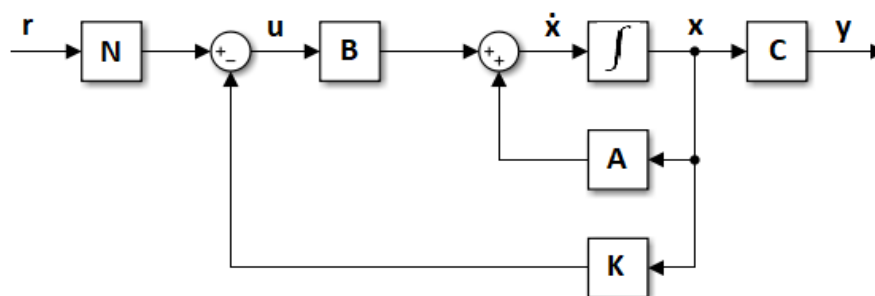


Figura 1.8. Esquema de control por realimentación de estados [8].

En el diseño de controladores en Espacio de Estado se parte de una función de transferencia a lazo cerrado con las características transitorias deseadas, con lo cual se tiene una nueva representación en Espacio de Estados.

$$\begin{aligned}\dot{x} &= Ax + Bu \\ \dot{x} &= Ax + B(-Kx + r) \\ \dot{x} &= (A - BK)x + Br \\ \frac{Y(s)}{U(s)} &= C[sI - (A - BK)]^{-1}B\end{aligned}\tag{1.20}$$

El control indicado en la Figura 1.8 funciona perfectamente para el seguimiento de referencia, pero no es óptimo para el rechazo de perturbaciones. Para resolver este problema se añade un integrador a la realimentación de estados obteniendo así el esquema indicado en la Figura 1.9.

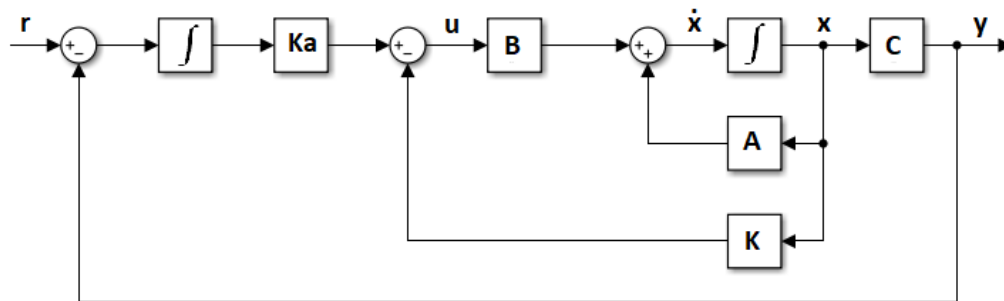


Figura 1.9. Esquema de control por realimentación de estados para el seguimiento robusto y rechazo a perturbaciones.

1.4.8 APP DESIGNER DE MATLAB

Las tareas básicas de AppDesigner son la creación de aplicaciones y la programación de la misma, donde se tiene control total de la ubicación y distribución de los componentes visuales de una interfaz gráfica de usuario. Para la creación de aplicaciones profesionales simplemente se arrastra y coloca los componentes visuales y se los coloca en el diseño, además ofrece ayuda para la alineación de los componentes y así conseguir un diseño preciso como se observa en la Figura 1.10 [13].

Este software tiene la ventaja de comprobar automáticamente los errores en la ejecución del código, mostrando los mensajes de advertencia mientras se está escribiendo el código, además de poseer una gran variedad de componentes que puede ser utilizado por el diseñador. Todos los componentes brindan al usuario del software con una amplia gama de combinaciones. Al igual que en el GUI de Matlab la mayoría de los componentes poseen las funciones llamadas Callback.

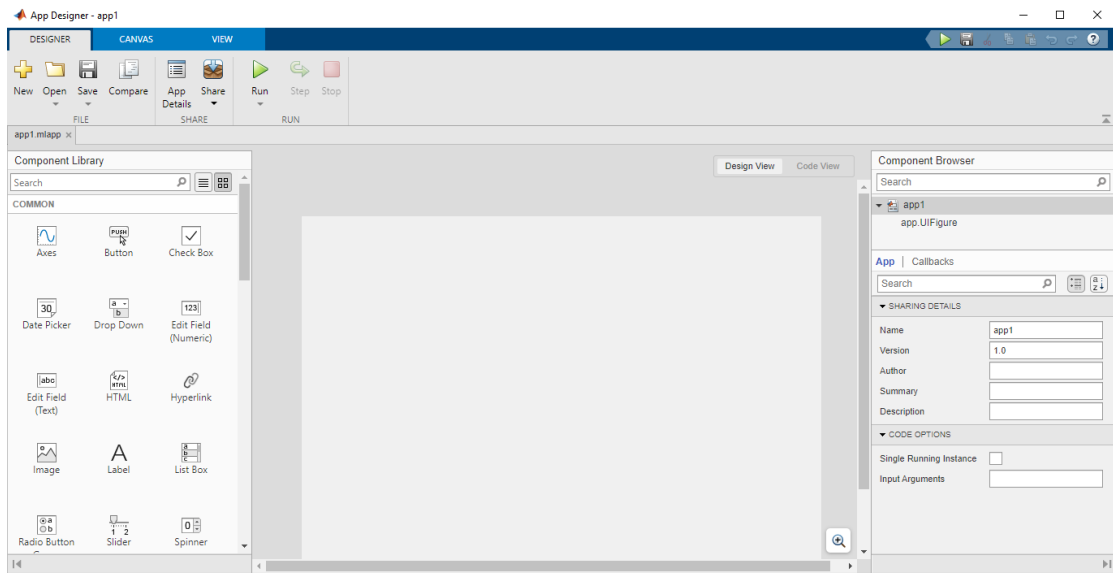


Figura 1.10. Pantalla principal de App Designer

1.4.8.1 Componentes de app designer

App Designer cuenta con una gran variedad de componentes interactivos, donde la mayoría de estos están disponibles dentro de la biblioteca de componentes y permiten al usuario una fácil creación de aplicaciones modernas únicamente arrastrando y soltando en el lienzo. Los componentes que están disponibles son: componentes comunes, ejes, contenedores, componentes de instrumentación, componentes extensibles y componentes de toolbox.

Tabla 1.4. Componentes comunes [13].

Componente	Función
Uiaxes	Crea ejes para gráficos.
Uibutton	Crea un botón pulsador.
Uicheckbox	Crea componente de casilla de verificación.
Uidropdown	Crea componente desplegable.
Uieditfield	Crea componente de campo de edición de texto numérico.
Uilabel	Crea componente de etiqueta
Uilistbox	Crea componente de cuadro de lista
Uimenu	Crea menú o elementos de menú.
Uislider	Crea componentes de control deslizante.
Uitable	Crea componente de interfaz de usuario de tabla.
Uitextarea	Crea componente de área de texto.
Uitogglebutton	Crea componente de botón de alternancia.

1.4.9 LABVIEW

LabVIEW es un entorno de programación gráfica y al igual que el AppDesigner de Matlab permite el desarrollo de aplicaciones de Ingeniería tales como: adquisición de datos, análisis de medidas y presentación de datos gracias al lenguaje de alta programación que permiten hacer diversas tareas sin el uso de otras herramientas de desarrollo complejas. La creación de interfaces gráficas de usuario resulta relativamente sencillo, un ejemplo de esta se muestra en la Figura 1.11.

1.4.9.1 Características principales

- Lenguaje de programación intuitivo.
- Librerías de alto nivel y herramientas de desarrollo.
- Gran variedad de funciones para entradas y salidas, control, presentación de datos y análisis.
- Posibilidad para crear aplicaciones sin programación.
- Depuración gráfica controlada y control del código fuente.
- Ayuda y una gran variedad de ejemplos.

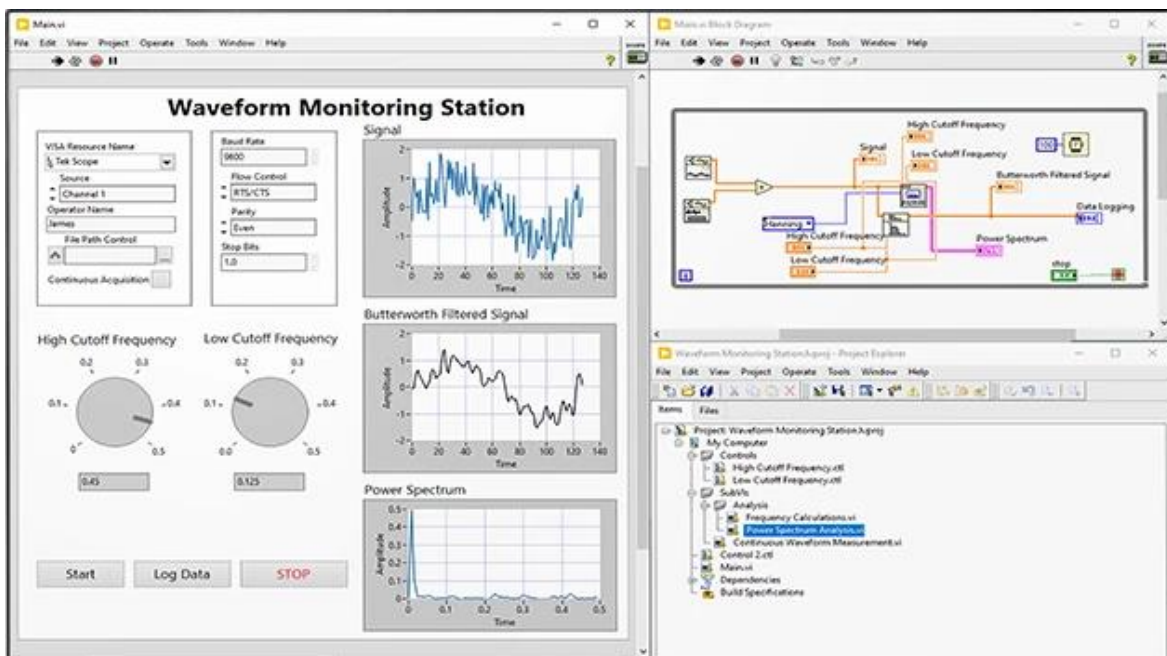


Figura 1.11. Interfaz gráfica de usuario con LabVIEW.

2. METODOLOGÍA

Realizar una descripción de lo que se realiza en este capítulo; describir las fases o pasos del proyecto para cumplir los objetivos específicos y así alcanza el objetivo general.

2.1 MODELO MATEMÁTICO DEL SISTEMA QNET-DCMCT

El QNET DCMCT es un sistema didáctico utilizado en la enseñanza de los fundamentos de control de motores DC. El modelo matemático se determina de forma analítica y práctica haciendo uso del sistema DAQ NI ELVIS II, LabVIEW y Matlab.

2.1.1 MODELO ANALÍTICO

Partiendo de las ecuaciones 1.6 a la 1.9 se obtuvieron las ecuaciones 1.10 y 1.11, las cuales relacionan la velocidad y la posición del motor respectivamente con su señal de referencia.

Haciendo uso de la Tabla 1.2 y la Tabla 2.1 se calcula la función de transferencia para ambas variables a controlar.

Tabla 2.1. Datos obtenidos de la paca el sistema QNET-DCMCT [2].

Símbolo	Descripción	Valor	Unidad
J_a	Inercia del servomotor	9.64×10^{-6}	$kg * m^2$
m	Masa de la carga	0.033	kg
r	Radio del disco de la carga	0.0242	m

El esfuerzo producido por el par mecánico es la suma de la Inercia del servomotor más la inercia de la carga.

$$J_L = \frac{1}{2}mr^2 = \frac{1}{2}(0.033)(0.0242)^2 = 9.663 \times 10^{-6} [kgm^2] \quad (2.1)$$

$$J = J_a + J_L = 9.64 \times 10^{-6} + 9.663 \times 10^{-6} = 1.93 \times 10^{-5} [kgm^2] \quad (2.2)$$

Obteniendo así las siguientes funciones de transferencia:

$$\frac{\omega(s)}{V(s)} = \frac{0.0334}{1.6 \times 10^{-7}s^2 + 1.68 \times 10^{-4}s + 1.2 \times 10^{-3}} \quad (2.3)$$

$$\frac{\theta(s)}{V(s)} = \frac{0.0334}{s(1.6 \times 10^{-7}s^2 + 1.68 \times 10^{-4}s + 1.2 \times 10^{-3})} \quad (2.4)$$

Las ecuaciones 1.10 y 1.11 pueden ser rescritas asumiendo que la inductancia de armadura (L_a) y el coeficiente de amortiguamiento (B) son muy pequeños en comparación con la resistencia de armadura, teniendo así que $L_a \approx 0$ y $B \approx 0$ por lo tanto:

$$\frac{\omega(s)}{V(s)} = \frac{K_m}{RJs + (K_a K_m)} = \frac{\frac{1}{K_a}}{\frac{RJ}{K_a K_m} s + 1} \quad (2.5)$$

$$\frac{\omega(s)}{V(s)} = \frac{29.94}{0.15s + 1} \quad (2.6)$$

$$\frac{\theta(s)}{V(s)} = \frac{29.94}{0.15s^2 + s} \quad (2.7)$$

2.1.2 MODELO EXPERIMENTAL O IDENTIFICACIÓN

La ecuación 2.3 representa de forma más exacta el sistema real a modelar, el cual mediante la curva de reacción se calcula un modelo aproximado de primer orden para posteriormente validarlo con el sistema físico haciendo uso del software LabVIEW.

En la Figura 2.1 se observa que el sistema tiene la forma de un modelo de primer orden con retardo, con lo cual mediante el método de Dahlin se calcula un modelo aproximado. Se aplica una entrada paso del 10% del valor nominal de la señal de control obteniendo así la respuesta que se indica en la Figura 2.1. El tiempo t_1 se determina cuando la salida alcanza el 28.3% del valor final de la salida y el tiempo t_2 cuando la salida alcanza el 63.2% del valor final, con estos datos y aplicando las ecuaciones 1.16 y la 1.17 se calcula los parámetros que modelan el sistema, siendo estos la constante de tiempo y la ganancia en lazo abierto [9].

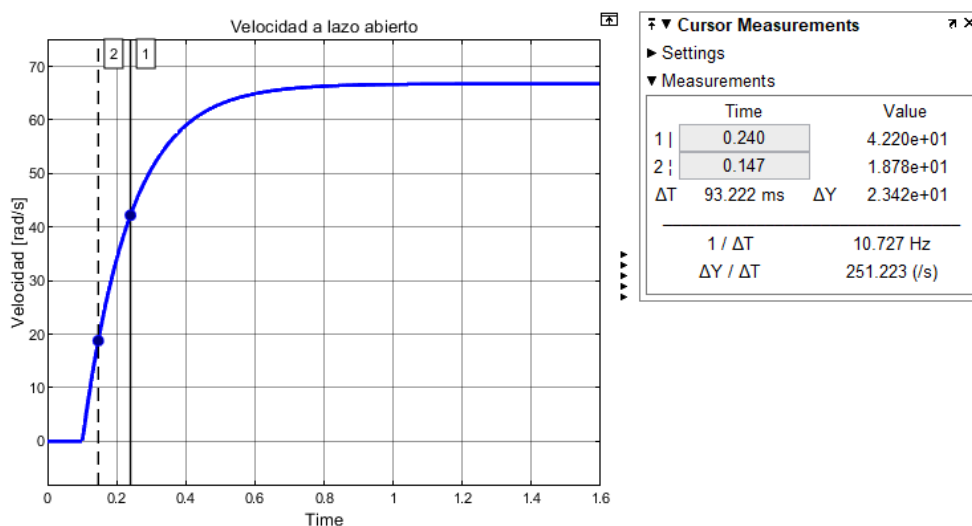


Figura 2.1. Salida a lazo abierto de velocidad ante una entrada pasa de amplitud 2.4 [V].

$$K = \frac{\Delta\delta}{\Delta U} = \frac{66.76}{2.4} = 27.82$$

$$\tau = 1.5(t_2 - t_1) = 1.5(93.22 \times 10^{-3}) = 0.1398$$

$$t_m = t_2 - \tau = 0.14 - 0.1398 = 2 \times 10^{-4}$$

$$\frac{\theta(s)}{V(s)} = \frac{27.82}{0.1398s^2 + s} e^{2 \times 10^{-4}} \quad (2.8)$$

La sintonización del controlador de velocidad mediante las ecuaciones de Dahlin dependen tanto de la ganancia, la constante de tiempo y el tiempo muerto, dado que al realizar el cálculo del tiempo muerto del sistema se obtuvo un valor muy pequeño el cual fácilmente puede ser despreciado, las fórmulas de Dahlin no sería un buen camino para sintonizar el controlador. Oscar Camacho [9] propone algunas fórmulas para el cálculo de las constantes del controlador PID dependiendo del tipo de planta a controlar, en este caso se usa las fórmulas para una planta de primer orden sin retardo.

Tabla 2.3. Ecuaciones de sintonización para diferentes modelos (con τ_c como parámetro de ajuste) [9].

Modelo	$K_c K$	τ_i	τ_d
$\frac{K}{\tau s + 1}$	$\frac{\tau}{\tau_c}$	τ	-
$\frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)}$	$\frac{\tau_1 + \tau_2}{\tau_c}$	$\tau_1 + \tau_2$	$\frac{\tau_1 \tau_2}{\tau_1 + \tau_2}$
$\frac{K}{s(\tau s + 1)}$	$\frac{2\tau_c + \tau}{\tau_c^2}$	$2\tau_c + \tau$	$\frac{2\tau_c \tau}{2\tau_c + \tau}$

El valor de la constante de tiempo en lazo cerrado (τ_c) queda a total criterio del diseñador, pero se debe tener en cuenta algunas consideraciones para un valor óptimo de τ_c :

- Un valor pequeño de τ_c implica una velocidad de respuesta rápida y un buen rechazo a perturbaciones.
- Un valor grande de τ_c implica estabilidad, robustez y pequeñas variaciones de entrada.

El parámetro de ajuste τ_c se puede considerar dentro del rango que va desde un valor mayor al tiempo muerto hasta un valor menor a la constante de tiempo en lazo abierto [9].

$$t_m < \tau_c < \tau \quad (2.9)$$

2.2 DISEÑO CONTROLADOR PID

2.2.1 CONTROLADOR DE VELOCIDAD

La velocidad del motor DC se controla mediante un controlador Proporcional-Integral. En la Figura 2.5 se indica el diagrama de bloques del sistema en lazo cerrado.

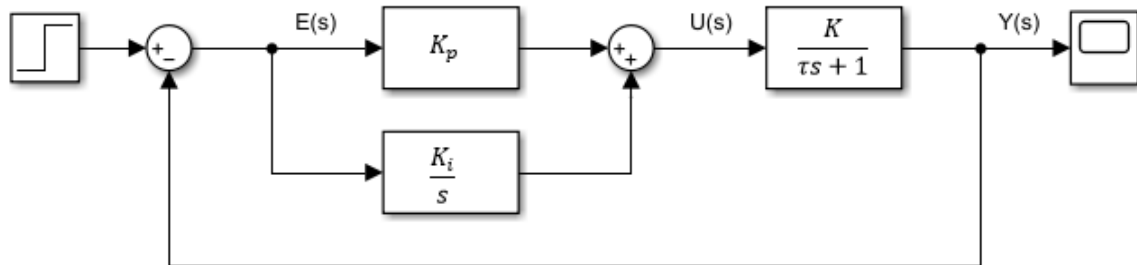


Figura. 2.5. Control en lazo cerrado para la velocidad

La función de transferencia del sistema en lazo cerrado viene dada por la siguiente expresión:

$$G_c(s) = \frac{\left(K_p + \frac{K_i}{s}\right) \left(\frac{K}{\tau s + 1}\right)}{1 + \left(K_p + \frac{K_i}{s}\right) \left(\frac{K}{\tau s + 1}\right)}$$

$$G_c(s) = \frac{\frac{K(K_p s + K_i)}{s(\tau s + 1)}}{1 + \left(\frac{K(K_p s + K_i)}{s(\tau s + 1)}\right)}$$

$$G_c(s) = \frac{K(K_p s + K_i)}{\tau s^2 + (1 + KK_p)s + KK_i}$$

$$G_c(s) = \frac{\frac{K(K_p s + K_i)}{\tau}}{s^2 + \frac{(1 + KK_p)}{\tau}s + \frac{KK_i}{\tau}} \quad (2.10)$$

El polinomio característico en lazo cerrado deseado es:

$$P(s) = s^2 + 2\varepsilon\omega_0 s + \omega_0^2 \quad (2.11)$$

Donde

ω_0 = La frecuencia de lazo cerrado no amortiguada

ε = Factor de amortiguamiento relativo

Igualando coeficientes de los denominadores de las ecuaciones 2.10 y 2.11 se tiene las siguientes expresiones para calcular las constantes del controlador.

$$K_p = \frac{2\varepsilon\omega_0\tau - 1}{K} \quad (2.12)$$

$$K_i = \frac{\omega_0^2\tau}{K} \quad (2.13)$$

Los criterios de diseño se escogen de tal manera que no afecten a la dinámica del sistema. Se requiere que el sistema en lazo cerrado tenga un tiempo de establecimiento de 0.35 segundos y un máximo sobre impulso del 2.8%, con las características transitorias deseadas se calcula el coeficiente de amortiguamiento y la frecuencia natural, siendo estas:

$$\varepsilon = 0.75$$

$$\omega_0 = 16 \left[\frac{rad}{s} \right]$$

Tabla 2.4. Constantes de sintonización del controlador de velocidad

Método	K_p	K_i
Dahlin	12.56	0.1398
Camacho	0.05	0.1398
Diseño	0.0766	1.242

2.2.2 CONTROLADOR DE POSICIÓN

La planta es de orden 1 lo que implica que no tiene error de posición y no es necesario la parte integral del controlador PID, por lo que se utiliza un controlador de tipo PD. El esquema de control utilizado para esta variable se muestra en la Figura 2.6.

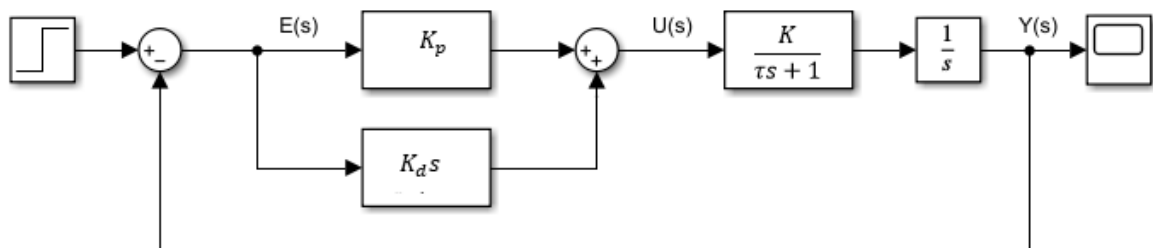


Figura 2.6. Control en lazo cerrado para la posición

La función de transferencia del sistema en lazo cerrado viene dada por la siguiente expresión:

$$G_c(s) = \frac{\frac{1}{s}(K_p + K_d s) \left(\frac{K}{\tau s + 1}\right)}{1 + \frac{1}{s}(K_p + K_d s) \left(\frac{K}{\tau s + 1}\right)}$$

$$G_c(s) = \frac{K(K_p + K_d s)}{\tau s^2 + s + K K_p + K K_d s}$$

$$G_c(s) = \frac{\frac{K(K_p + K_d s)}{\tau}}{s^2 + \frac{(1 + K K_d)}{\tau} s + \frac{K K_p}{\tau}} \quad (2.14)$$

El polinomio característico en lazo cerrado deseado es:

$$P(s) = s^2 + 2\varepsilon\omega_0 s + \omega_0^2 \quad (2.15)$$

Igualando coeficientes de los denominadores de las ecuaciones 2.14 y 2.15 se tiene las siguientes expresiones para calcular las constantes del controlador.

$$K_d = \frac{2\varepsilon\omega_0\tau - 1}{K} \quad (2.16)$$

$$K_p = \frac{\omega_0^2\tau}{K} \quad (2.17)$$

Los criterios de diseño se escogen de tal manera que no afecten a la dinámica del sistema, al igual que para la variable velocidad se escogen las características transitorias deseadas del sistema en lazo cerrado, para el caso de la variable posición se requiere un tiempo de establecimiento de 0.25 segundos y un máximo sobre impulso de 9.5%, con estas características se calcula el coeficiente de amortiguamiento y la frecuencia natural:

$$\varepsilon = 0.6$$

$$\omega_0 = 25 \left[\frac{rad}{s} \right]$$

Las expresiones para la sintonización de controladores PID mediante el método de Dahlin presentadas en la Tabla 1.3 sirven únicamente para sistemas de primer orden con retardo, para el caso de la variable posición el modelo del sistema a controlar es de segundo orden y no se puede aplicar este método para la sintonización del controlador.

Tabla 2.5. Constantes de sintonización del controlador de posición.

Método	K_p	K_d
Camacho	1.22	0.083
Diseño	3.14	0.115

2.3 DISEÑO DEL CONTROLADOR POR REALIMENTACIÓN DE ESTADO CON OBSERVADOR DE ORDEN COMPLETO

En el diseño de la realimentación del vector de estados se asume que todos los estados están disponibles para la medición, en la práctica generalmente esto no es posible, ya que solo algunos estados pueden ser medidos directamente, por ende, es necesario estimar algunos valores o bien cuantificarlos mediante un observador. El observador es un subsistema del sistema de control que realiza una estimación de estados, a partir de las mediciones de las variables de salida $y(t)$ y control $u(t)$. Existen observadores de orden reducido y de orden completo, el primero solo estima los estados que no son medibles y el segundo realiza una estimación de todos los estados. La aplicación de cada uno de los observadores depende de la complejidad del sistema y el coste que esto implica, dado que para cada estado se necesita de un sensor para su medición, por tanto, se recomienda diseñar un observador de orden completo. En la Figura 2.6 se muestra un sistema con un observador de orden completo [8].

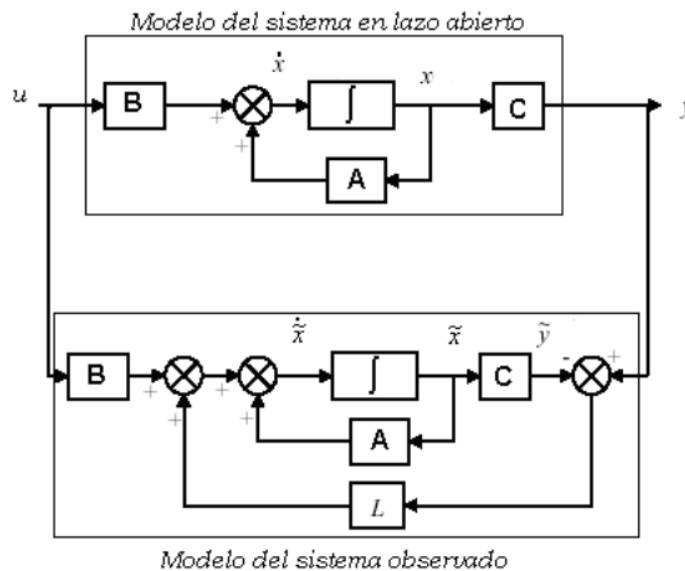


Figura 2.6. Diagrama de bloques de un sistema con observador de estados [8].

Para el diseño del observador de estados es necesario que el sistema sea completamente observable. El sistema observado está dado por:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \tilde{y}) \quad (2.18)$$

$$\tilde{y} = C\hat{x} \quad (2.19)$$

$$\dot{\hat{x}} = (A - CL)\hat{x} + Bu + Ly \quad (2.20)$$

Donde \tilde{x} denota la estimación del estado x , $\tilde{x} \rightarrow x$ cuando $t \rightarrow \infty$, el observador debe cumplir que $e \rightarrow 0$ cuando $t \rightarrow \infty$. Para el diseño del observador es necesario encontrar la matriz de ganancia L , la cual dependiendo de los valores escogidos hará que el error sea estable y sea lo suficientemente rápido para tender a un valor nulo [8].

Para encontrar la matriz L se utiliza los mismos métodos que se usan para hallar la matriz K con ciertas variaciones.

La fórmula de Ackerman es útil para calcular los valores de la matriz de ganancia del observador en Espacio de Estados cuando se tiene un sistema de una entrada.

$$L = \begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_n \end{bmatrix} \quad (2.21)$$

El polinomio característico deseado

$$p(\lambda) = \lambda^n + \beta_1 \lambda^{n-1} + \dots + \beta_n \quad (2.22)$$

Siendo la matriz de ganancia del observador

$$L = p(A)P_o^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \quad (2.23)$$

Donde P_o es la matriz de observabilidad y

$$p(A) = A^n + \beta_1 A^{n-1} + \dots + \beta_{n-1} A + \beta_n I \quad (2.24)$$

2.3.1 CONTROL DE LA VELOCIDAD

Para la salida de velocidad se tiene la siguiente representación en variables de estado:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -0.5181 & 1730.6 \\ -4.024 & -1048.2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 120.4819 \end{bmatrix} v$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

La matriz de controlabilidad del sistema es:

$$Q = \begin{bmatrix} 0 & 2.085 \times 10^5 \\ 120.4819 & -1.263 \times 10^5 \end{bmatrix}$$

Al analizar la controlabilidad del sistema se tiene que este es totalmente controlable puesto que tanto el rango de la matriz Q y su dimensión son iguales. La respuesta del sistema en lazo abierto es suave con un tiempo de establecimiento de aproximadamente 0.5 [s] y sin máximo sobreimpulso, por lo cual se implementa un controlador de tal forma que el sistema en lazo cerrado guarde las mismas características transitorias que el sistema en lazo abierto. Los polos de este sistema son $\lambda_1 = -7.2079$ y $\lambda_2 = -1041.5$. La respuesta transitoria generalmente es determinada por el polo dominante del sistema, un polo es no dominante si se tiene lo siguiente:

- Si la parte real del polo es al menos diez veces mayor que la parte real del polo más próximo al eje imaginario.
- Si hay un cero muy cercano a la ubicación del polo.

Por lo tanto la respuesta transitoria del sistema a lazo abierto depende del polo λ_1 y como se mencionó que se quiere mantener las características transitorias del sistema en lazo abierto se escoge los polos en lazo cerrado de tal forma que λ_{1c} siga siendo el polo dominante $\lambda_{1c} = -8$ y $\lambda_{2c} = -100$.

Para el diseño del observador de estados al igual que en el cálculo de la matriz de realimentación de estados se escoge una matriz L de tal forma que asigne los autovalores de la matriz A_c en lazo cerrado en:

- El semiplano izquierdo para asegurar la estabilidad del sistema.
- A la izquierda de los polos del sistema para garantizar que la dinámica del error del observador sea más rápida que la del sistema.

Con lo cual los polos del observador de estados escogidos son $\lambda_{o1} = -100$ y $\lambda_{o2} = -1000$. La rapidez del observador depende únicamente del criterio del diseñador. La matriz de ganancia de realimentación (K) y la matriz de ganancia del observador (L) se pueden obtener utilizando la fórmula de Ackerman como se indica en la ecuación 2.23 y mediante Matlab con el comando *acker*. El cálculo de la ganancia de seguimiento del controlador (N) y las matrices K y L se lo realiza mediante un archivo.m. Obteniendo así los siguientes resultados:

$$K = [-0.0298 \quad -7.8079]$$

$$L = \begin{bmatrix} 51.2891 \\ 22.3811 \end{bmatrix}$$

$$N = 0.0038$$

2.3.2 CONTROL DE LA POSICIÓN

Para la salida de posición se tiene la siguiente representación en variables de estado:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -0.5181 & 1730.6 \\ 0 & -4.024 & -1048.2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 120.4819 \end{bmatrix} v$$
$$y = [1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Los polos de este sistema son $\lambda_1 = -7.2079$, $\lambda_2 = -1041.5$ y $\lambda_3 = 0$. Con todo lo mencionado para el control de la variable velocidad se escoge los polos en lazo cerrado $\lambda_{1c} = -8$, $\lambda_2 = -100$ y $\lambda_3 = -300$.

Para los polos del observador de estados se escoge $\lambda_{o1} = -100$, $\lambda_{o2} = -500$ y $\lambda_{o3} = -1200$.

El cálculo de la ganancia de seguimiento del controlador (N) y las matrices K y L se lo realiza mediante un archivo.m. Obteniendo así los siguientes resultados:

$$K = [1.1739 \quad -0.0134 \quad -7.2323]$$

$$L = \begin{bmatrix} 1.1511 \\ 0.1248 \\ -5.3179 \end{bmatrix}$$

$$N = 1.1511$$

2.4 DISEÑO DEL LA INTERFAZ GRÁFICA EN APP DESIGNER.

El diseño de una interfaz gráfica sencilla e interactiva permite al usuario un fácil entendimiento y manejo de esta, la cual contiene únicamente botones, gráficas, bloques de textos editables y sliders minimizando la utilización de componentes más complejas de la App Designer. A nivel de simulación se desarrollaron tres ventanas, siendo la primera la pantalla de inicio donde se puede escoger la variable a controlar, tanto la segunda como la tercera pantalla permite ingresar el tipo de controlador, parámetros de los controladores, características eléctricas y mecánicas de la planta, tipo de gráfica a dibujar, tipo de simulación, referencia y se indica el modelo de la planta.



Figura 2.7. Diseño de la pantalla de Inicio en AppDesigner

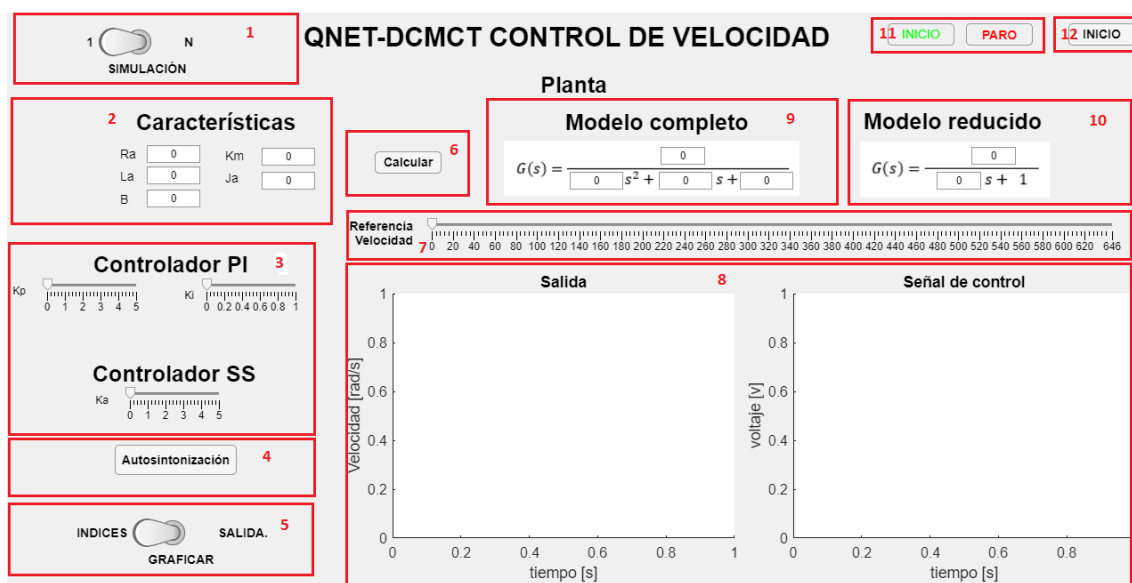


Figura 2.8. Diseño de la pantalla para el control de la velocidad en AppDesigner.

Donde

- 1: Permite escoger el tipo de simulación, siendo 1 simulación única y N simulación continua.
- 2: Ingreso de característica eléctrica y mecánicas de la planta.
- 3: Ingreso de parámetros de sintonización del controlador PI y Espacio de Estado.
- 4: Permite la auto sintonización de ambos controladores.
- 5: Permite escoger las gráficas a mostrar en 8. INDICES gráfica el ISE e ISU de ambos controladores. SALIDA gráfica la salida del sistema y la señal de control.

- 6: Calcula el modelo del sistema a partir de las características ingresadas en 2.
- 7: Permite el ingreso de la referencia del sistema.
- 8: Permite observar las gráficas escogidas en 5.
- 9: Indica el modelo completo de la planta.
- 10: Indica el modelo reducido de la planta donde se desprecia la inductancia y la viscosidad del sistema.
- 11: Botones de inicio y paro de la simulación.
- 12: Permite navegar a la pantalla de Inicio.

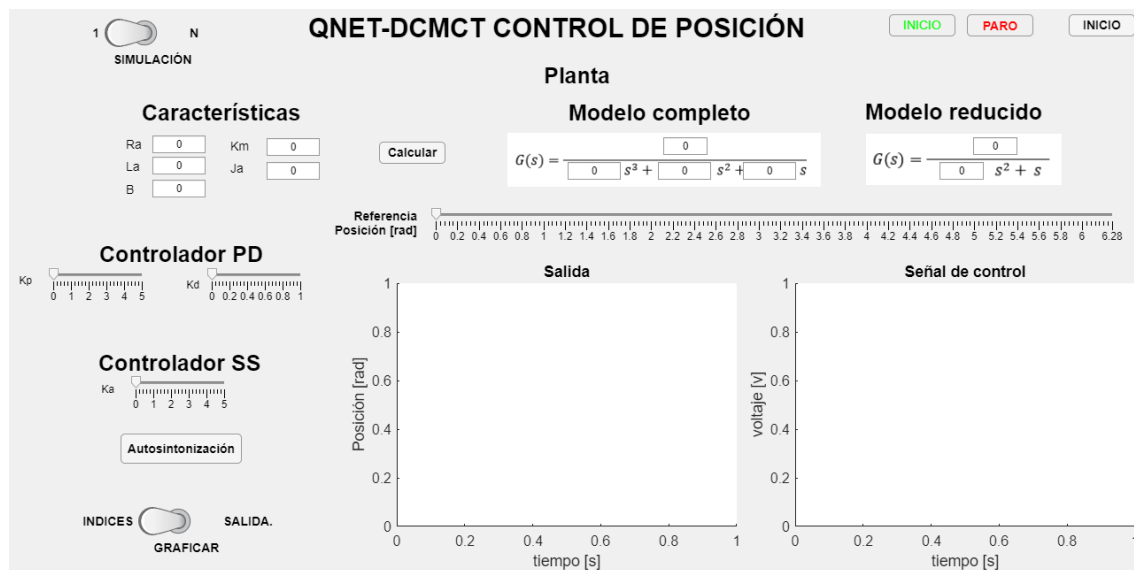


Figura 2.9. Diseño de la pantalla para el control de la posición en AppDesigner.

2.5 DISEÑO DEL LA INTERFAZ GRÁFICA EN LABVIEW

El Data Acquisition Toolbox de Matlab es compatible con los subsistemas de entradas y salidas analógicas y digitales del sistema NI ELVIS II. Este Toolbox permite acceder a los instrumentos integrados en el sistema directamente desde Matlab como son: fuente de alimentación variable, generador de funciones y multímetro digital. A pesar de que se puede adquirir datos con el sistema NI ELVIS II no se puede usar para la adquisición de datos en tiempo real, por lo cual es necesario buscar un software que permita realizar una adquisición de datos en tiempo real como es el caso del software LabVIEW.

La interfaz contiene principalmente botones, bloques de texto editables y gráficos, que permiten la configuración de las características del modelo de la planta, los parámetros del

controlador PI y Espacio de Estados para la variable velocidad y los parámetros del controlador PD y Espacio de Estado para la variable posición. Se plantea crear tres modelos en LabVIEW que cumplan con las exigencias del proyecto y permitan realizar las configuraciones necesarias para probar el correcto funcionamiento de los controladores diseñados y validar el modelo matemático obtenido en la sección 2.1.2.

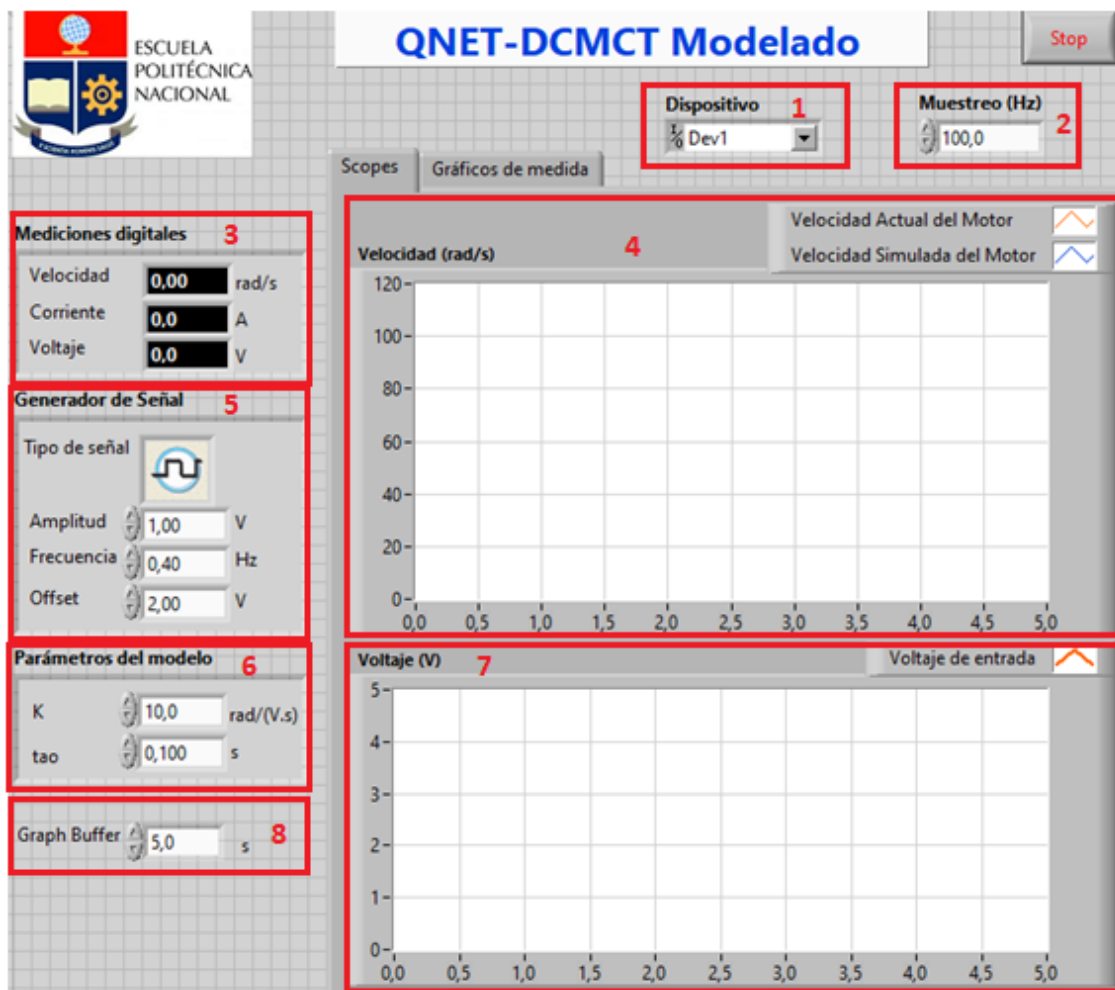


Figura 2.10. Diseño de la pantalla para modelado en LabVIEW

Donde

- 1: Configuración del dispositivo de adquisición de datos.
- 2: Ingreso de la frecuencia de muestreo [Hz].
- 3: Indica mediciones de velocidad [rad/s], corriente [A] y voltaje [V] del motor DC.
- 4: Voltaje de entrada en lazo abierto, permite escoger el tipo de señal y adicionalmente configurar la amplitud [V], frecuencia [Hz] y offset [V] de la señal escogida.
- 5: Permite observar la velocidad del motor DC [rad/s].

- 6: Configuración de los parámetros del modelo matemático del sistema físico.
- 7: Permite observar la señal configurada en 4.

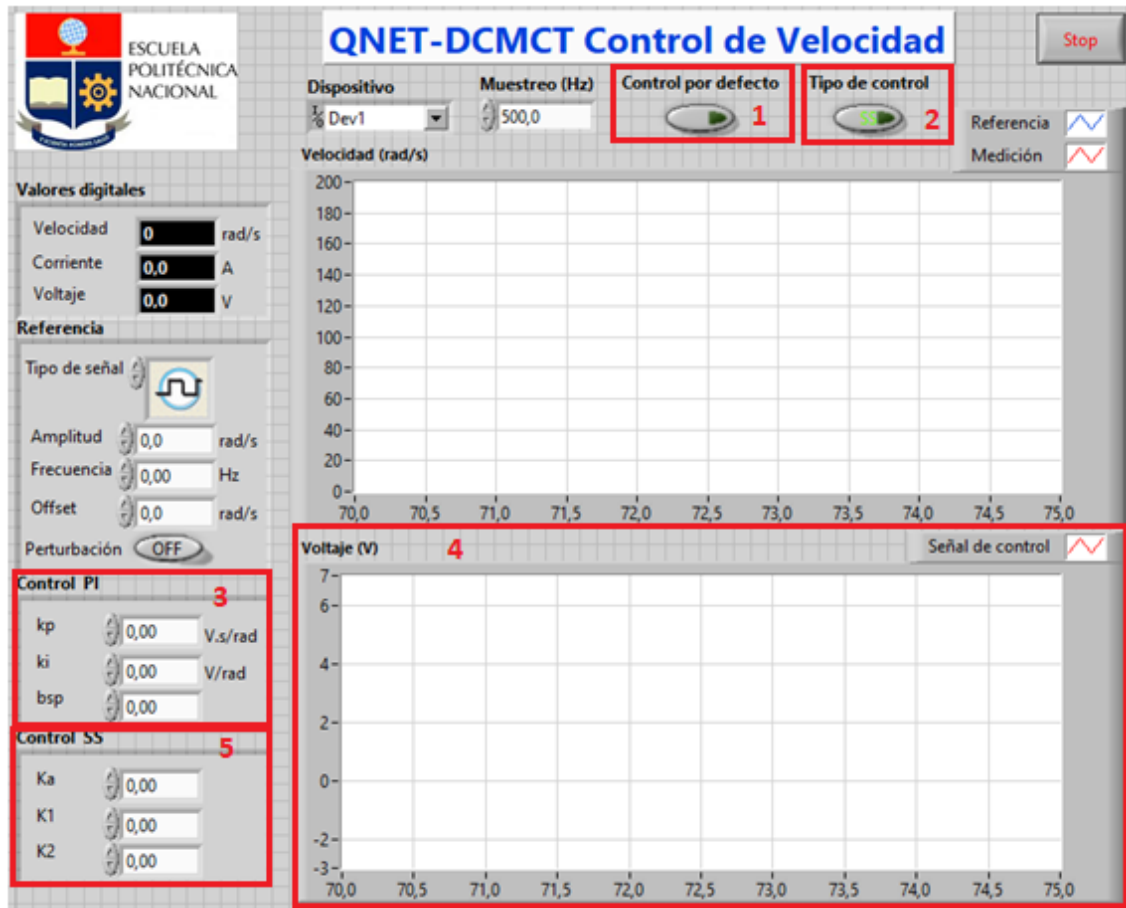


Figura 2.11. Diseño de la pantalla para el control de la velocidad en LabVIEW.

Donde

- 1: Permite cargar los parámetros del controlador PI y el controlador en Espacio de Estados calculados en la sección 2.2 y 2.3. Al estar desactivado se puede configurar dichos parámetros desde 3 y 5 dependiendo del tipo de controlador escogido.
- 2: Al estar activado el sistema tiene un control PI, en cambio al estar desactivado el control es en Espacio de Estado.
- 4: Permite observar la señal de control del motor DC.

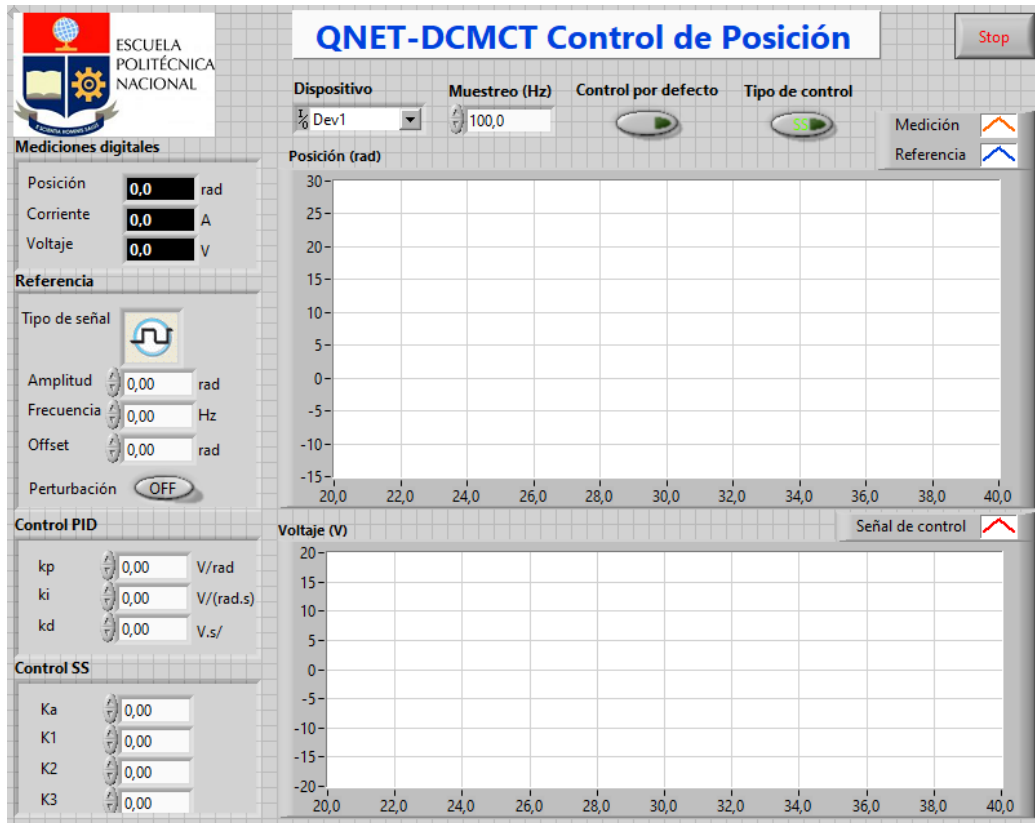


Figura 2.12. Diseño de la pantalla para el control de la posición en LabVIEW.

La Figura 2.12 presenta los mismos componentes utilizados para el diseño de la pantalla en el control de velocidad, diferenciándose en el ingreso de los parámetros del controlador PD y el controlador en Espacio de Estados.

3. RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

En este capítulo se indican los resultados obtenidos tanto a nivel de simulación como también a nivel de implementación a partir del modelo matemático del sistema y de los diferentes controladores diseñados para la velocidad y la posición del motor, con la ayuda de los softwares de simulación Simulink, AppDesigner de Matlab y LabVIEW.

Para la validación del modelo los resultados se muestran únicamente en LabVIEW y se compara el modelo de primer orden obtenido en la sección 2.1 con el sistema físico QNET DCMCT disponible en el laboratorio del Sistemas de Control Automático. En cambio, para las variables velocidad y posición los resultados se muestran primero a nivel de simulación y después a nivel de implementación de la siguiente manera:

- Valbración de desempeño ante diferentes cambios de referencia.

- Valoración de desempeño ante rechazo a perturbaciones.

Las pruebas a las que son sometidos los diferentes controladores son:

- Cambios de referencia, tanto positivos como negativos, que permitan evaluar los índices de desempeño ISE, ISU, tiempo de establecimiento (t_s) y máximo sobre impulso (M_p).
- Perturbaciones tanto en la salida como en la señal de control para evaluar el rechazo a perturbaciones de los esquemas de control propuestos.

3.1 VALIDACIÓN DEL MODELO DEL SISTEMA QNET DCMCT

El modelo matemático de la planta se determinó en el capítulo 2.1 donde se obtuvo un modelo de segundo orden, el cual mediante Simulink – Matlab se aproximó a un modelo de primer orden con retardo para la sintonización de controladores, adicionalmente se tiene un modelo reducido en el cual se desprecia el efecto de la viscosidad y la inductancia de armadura dado que estos tienen valores muy pequeños en comparación con la resistencia de armadura siendo las ecuaciones 3.1 y 3.2 respectivamente. Para validar el modelo se ocupa la pantalla indicada en la Figura 2.1 obteniendo así los resultados indicados en la Figura 3.1 y 3.2.

$$\frac{\omega(s)}{V(s)} = \frac{27.82}{0.1398s+1} \quad (3.1)$$

$$\frac{\omega(s)}{V(s)} = \frac{29.94}{0.15s+1} \quad (3.2)$$

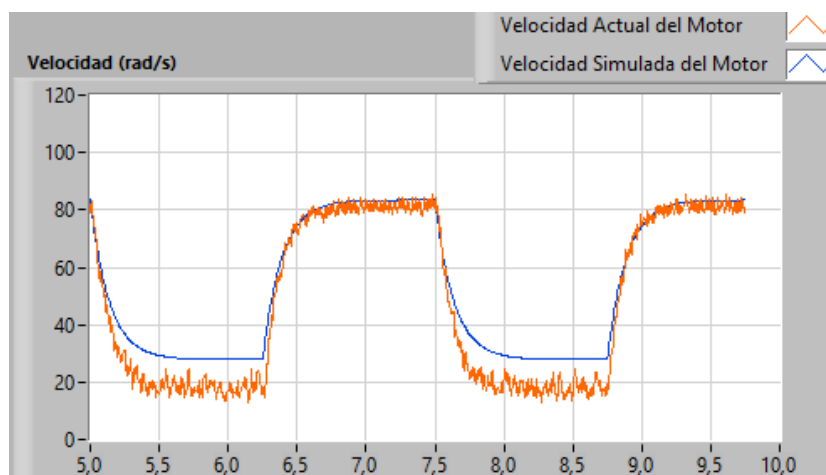


Figura 3.1. Validación del modelo aproximado de primer orden

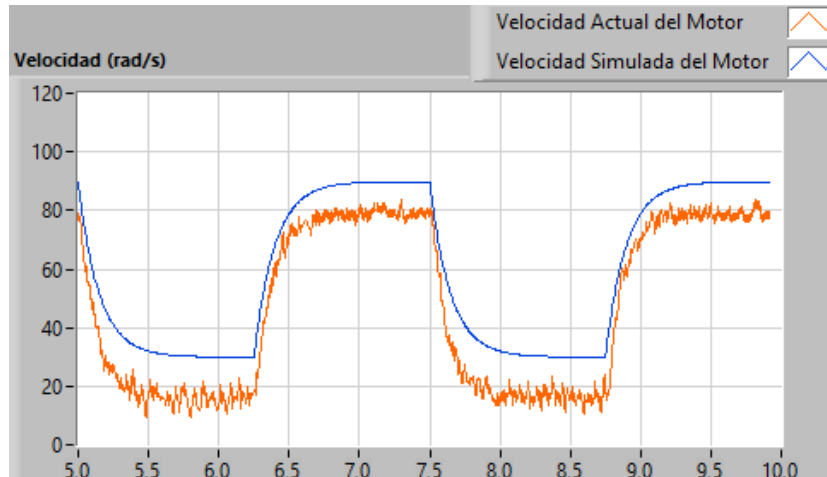


Figura 3.2. Validación del modelo de primer orden despreciando la inductancia y la viscosidad.

El modelo aproximado indicado en la Figura 3.2 se aleja más del modelo físico, teniendo esto en cuenta la sintonización de los controladores se las realizó mediante el modelo aproximado de primer orden determinado mediante la curva de reacción.

3.2 CONTROL DE VELOCIDAD EN APP DESIGNER

En la sección 2.2 se propuso tres parámetros diferentes para la sintonización del controlador PI de los cuales tras hacer las respectivas simulaciones se obtuvo una mejor respuesta y señal de control con los parámetros sintonizados mediante las ecuaciones de Camacho, esta salida se compara con la obtenida con el controlador en Espacio de Estado. Para el controlador en Espacio de Estado se diseñó un observador completo que proporcione los estados del sistema, la respuesta del observador se indica en la Figura 3.3.

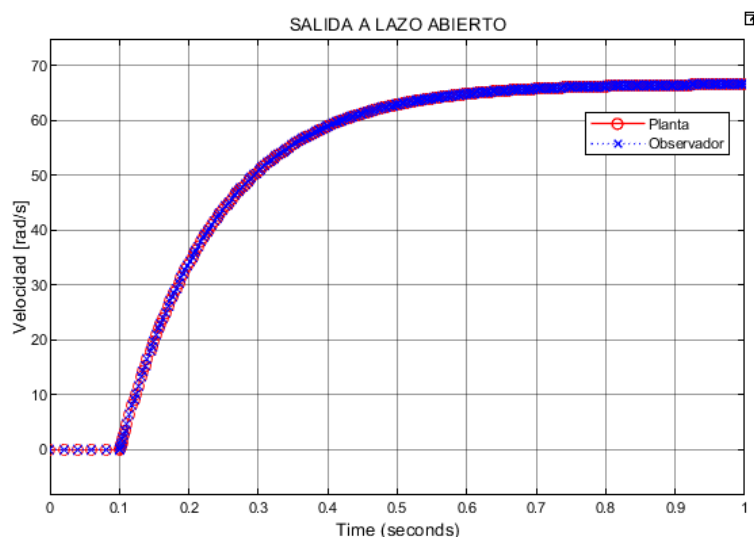


Figura 3.3. Observador de estados de orden completo

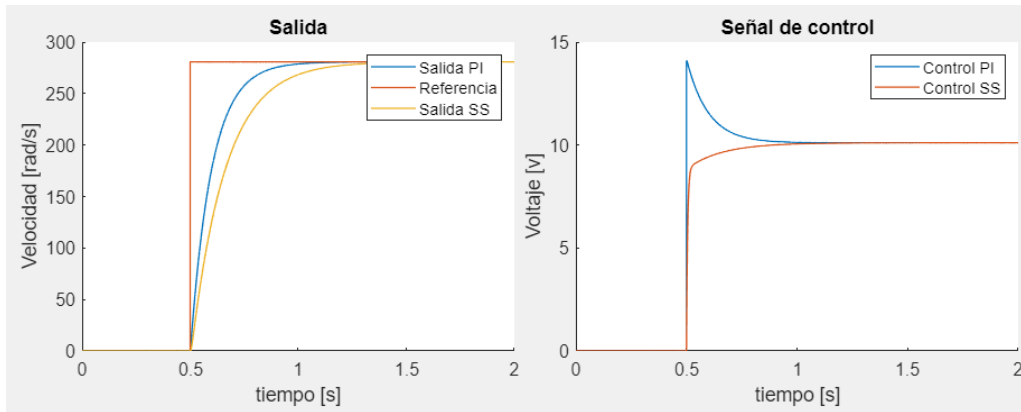


Figura 3.4. Respuesta del sistema y señal de control ante un cambio de referencia de velocidad esquema PI parámetros sintonizados mediante Camacho y el esquema de Espacio de Estado.

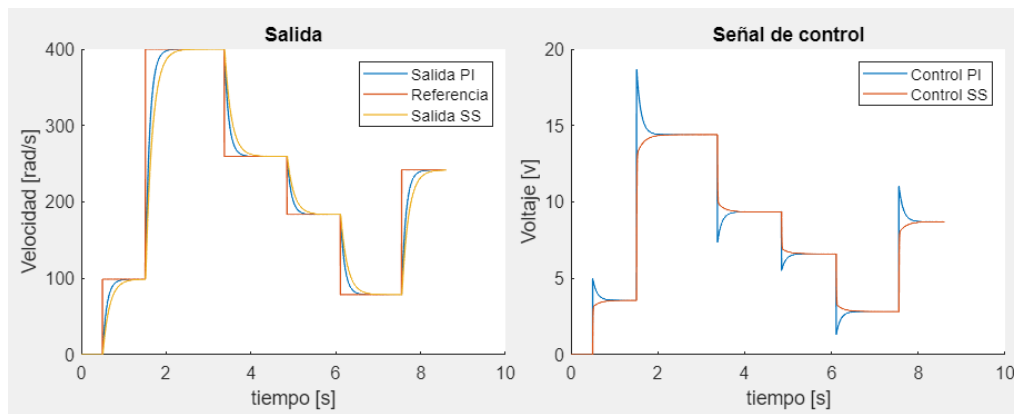


Figura 3.5. Respuesta del sistema y señal de control cambios de referencia positivos y negativos de velocidad esquema PI parámetros sintonizados mediante Camacho y el esquema de Espacio de Estado.

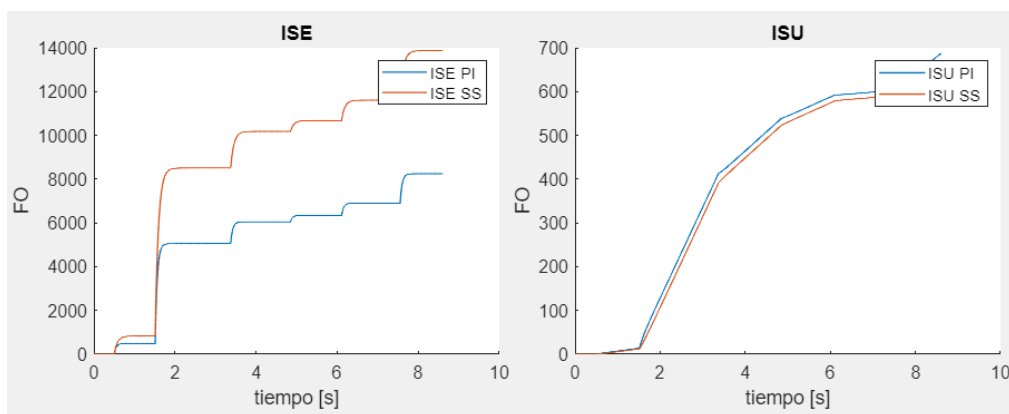


Figura 3.6. Índices de desempeño ante cambios de referencia positivos y negativos de velocidad esquema PI parámetros sintonizados mediante Camacho y el esquema de Espacio de Estado.

En la Figura 3.5 se observa que la salida con el controlador en Espacio de Estado presenta mayor tiempo de estabilización que la salida con el controlador PI pero este tiene una señal de control más brusca y un sobre impulso del 20%. De la Figura 3.6 se deduce que el mejor controlador con respecto al error en estado estable y respuesta transitoria es el controlador PI pero si tenemos en cuenta el tiempo de vida útil del actuador el controlador en Espacio de Estados tiene un menor índice ISU lo que implica un menor desgaste ante cambios de referencias.

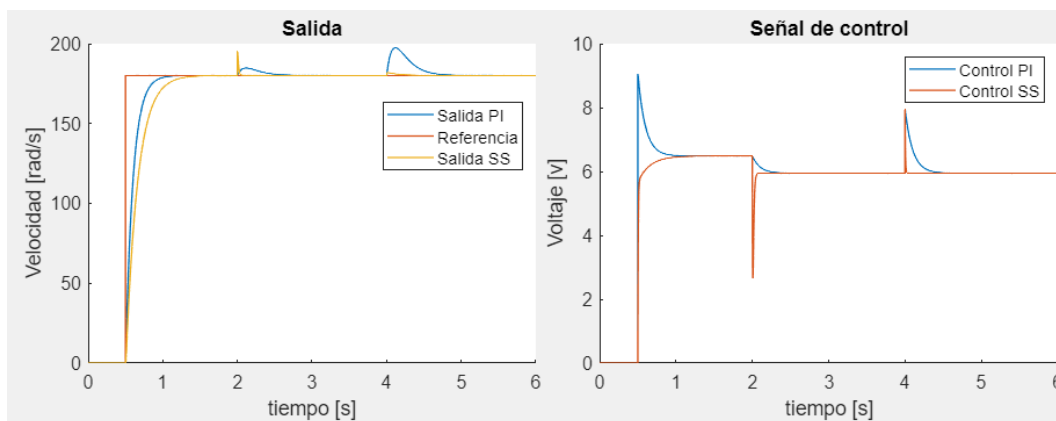


Figura 3.7. Respuesta del sistema ante el rechazo a perturbaciones

Se aplicó al sistema una perturbación de salida de 15 [rad/s] al instante $t=2$ [s] y una perturbación en la señal de control de 2 [V] al instante $t=4$ [s], siendo el controlador por Espacio de Estado el que más rápido rechazo ambas perturbaciones tal como se observa en la Figura 3.7. El controlador en Espacio de Estados presenta una señal de control suave cuando se tiene cambios de referencias pero esta presenta un pico de voltaje cuando se ingresa una perturbación ya sea en la señal de control o en la señal de salida, a diferencia del controlador PI que el comportamiento de la señal de control es lo contrario. De los resultados obtenidos se espera que el controlador utilizado sea el más rápido ante el rechazo a perturbaciones por ende el controlador por Espacio de Estado sería el que mejor rechaza las perturbaciones de salida y entrada al sistema.

3.3 CONTROL DE POSICIÓN EN APP DESIGNER

En la sección 2.3 se propuso dos valores diferentes para cada uno de los parámetros de sintonización del controlador PD obteniendo una mejor respuesta con los parámetros determinados mediante características transitorias deseadas.

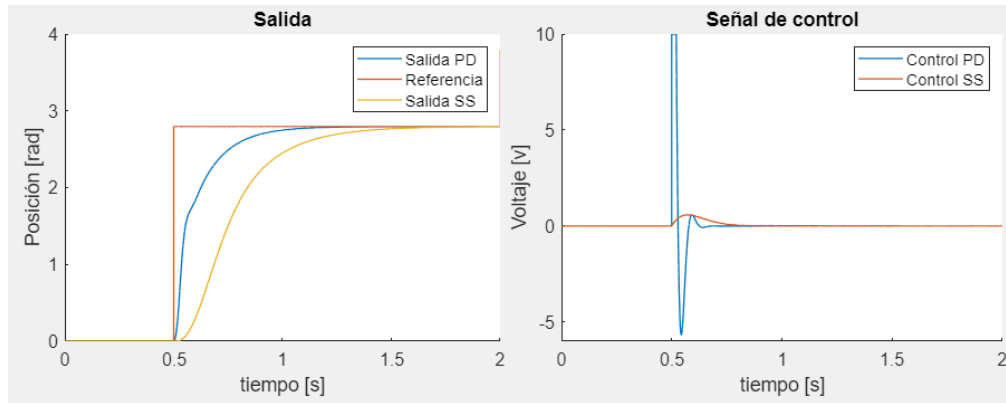


Figura 3.8. Respuesta del sistema y señal de control ante un cambio de referencia de posición esquema PD parámetros sintonizados mediante respuesta transitoria deseada y esquema de Espacio de Estado.

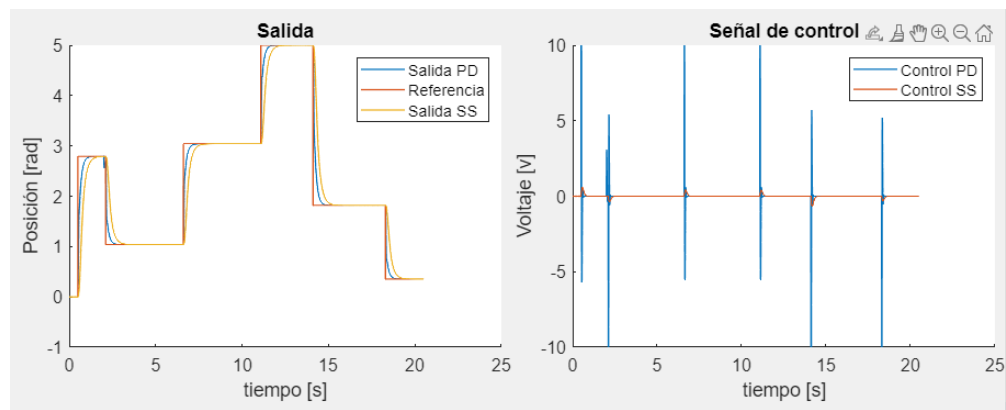


Figura 3.9. Respuesta del sistema y señal de control ante cambios de referencia positivos y negativos de posición esquema PD parámetros sintonizados mediante respuesta transitoria deseada.

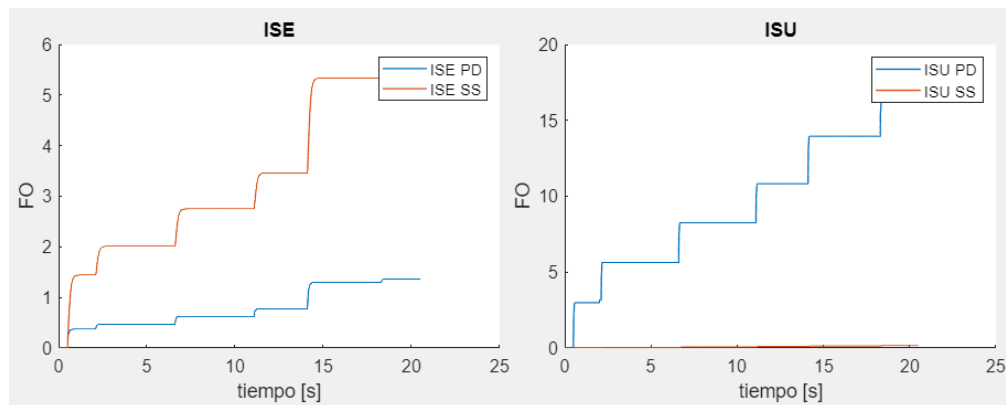


Figura 3.10. Índices de desempeño ante cambios de referencia positivos y negativos de posición esquema PD parámetros sintonizados mediante respuesta transitoria deseada y el esquema de Espacio de Estado.

Al igual que en la variable velocidad el esquema PD es el más rápido pero la señal de control presenta sobre picos muy elevados y fue necesario hacer uso de saturadores para limitar la señal de control. Dado que el tiempo de estabilización con el esquema de variables de estado es mayor se tiene un ISE más grande que en el esquema PD y un índice ISU mucho más pequeño lo que implica un mayor tiempo de vida útil del controlador.

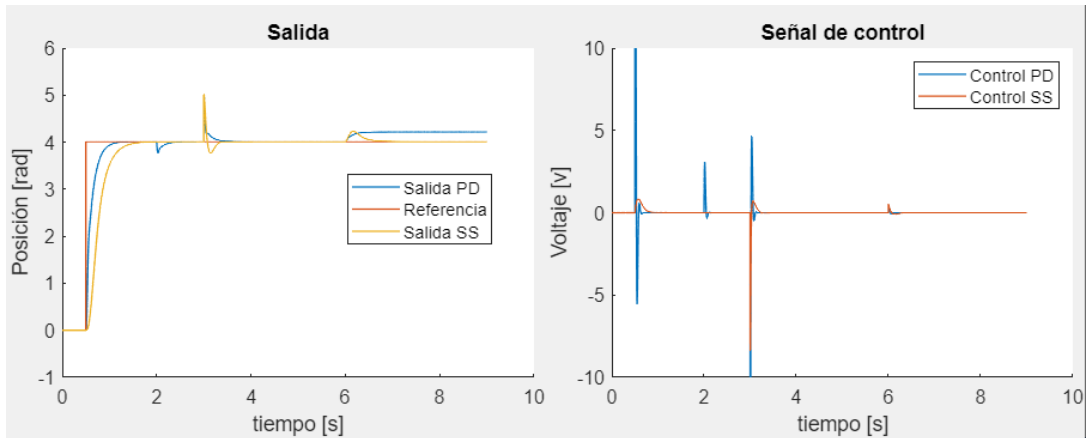


Figura 3.11. Respuesta del sistema ante el rechazo a perturbaciones

Se aplicó al sistema una perturbación de salida de 1 [rad] al instante $t=3$ [s] y una perturbación en la señal de control de 0.5 [V] al instante $t=6$ [s], el controlador PD no fue suficiente para corregir una perturbación constante en la señal de control pero si rechaza bien las perturbaciones instantáneas de salida. La señal de control para las perturbaciones ingresadas al sistema con este tipo de controlador presenta picos grandes de voltaje, en cambio el controlador por Espacio de Estados responde de forma adecuada tanto para perturbaciones de salida y perturbaciones en la señal de control. La señal de control para los dos controladores diseñados es más brusca cuando se tiene una perturbación en la salida del sistema.

3.4 CONTROL DE VELOCIDAD EN LABVIEW

Una vez probado los controladores en la interfaz de AppDesigner se procede a probar los mismos en la interfaz de LabVIEW con el sistema físico QNET-DCMCT, obteniendo así las siguientes respuestas:

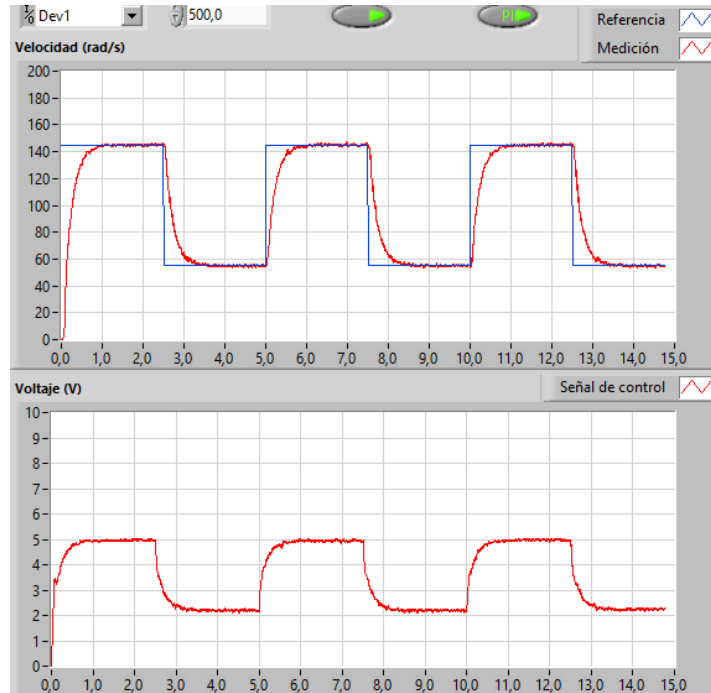


Figura 3.12. Respuesta del sistema y señal de control ante cambios de referencia de velocidad esquema PI parámetros sintonizados mediante Camacho.

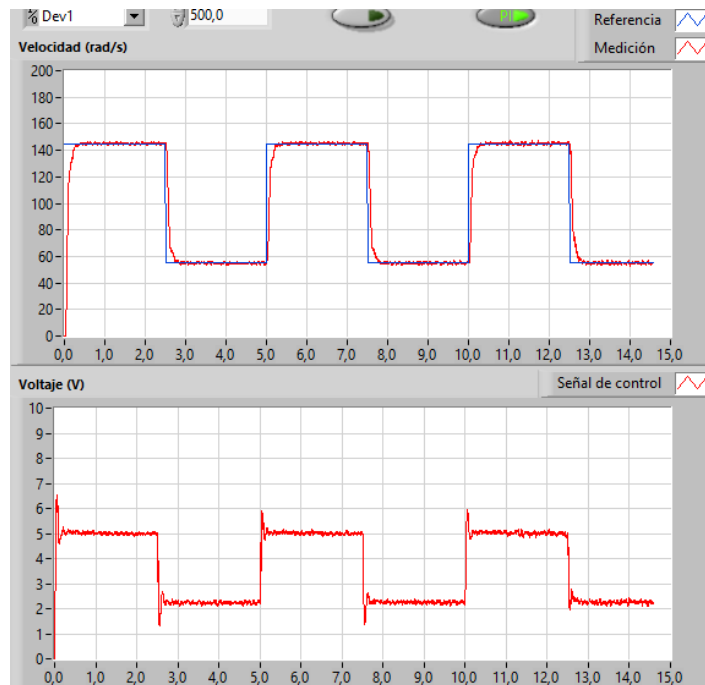


Figura 3.13. Respuesta del sistema y señal de control ante cambios de referencia de velocidad esquema PI parámetros sintonizados mediante características transitorias deseadas.

La respuesta del sistema con los parámetros sintonizados por Camacho no presenta sobre impulso y tiene un tiempo de establecimiento aproximado de 1 segundo con una señal de

control suave sin oscilaciones, en cambio la respuesta con los parámetros sintonizados mediante características transitorias deseadas igual que el caso anterior no presenta sobre impulso con un tiempo de establecimiento de casi medio segundo, pero tiene una señal de control con oscilaciones.

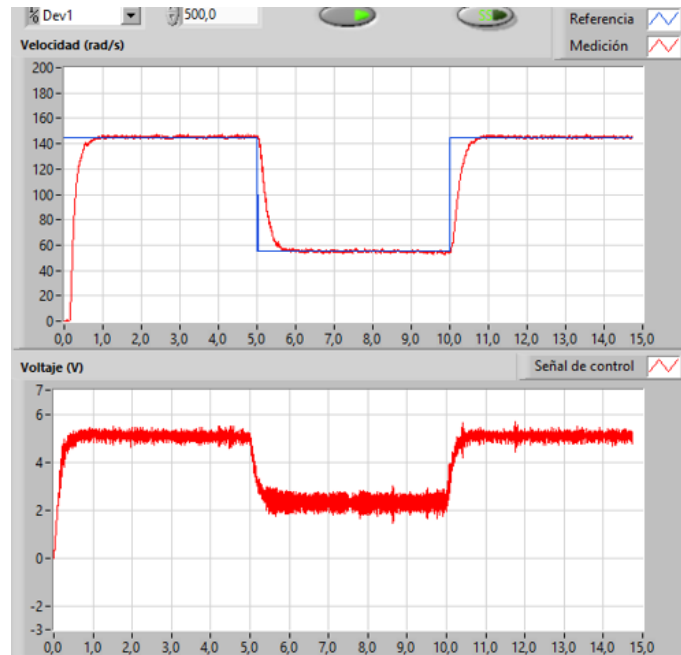


Figura 3.14. Respuesta del sistema y señal de control ante cambios de referencia de velocidad para el esquema de Espacio de Estado.

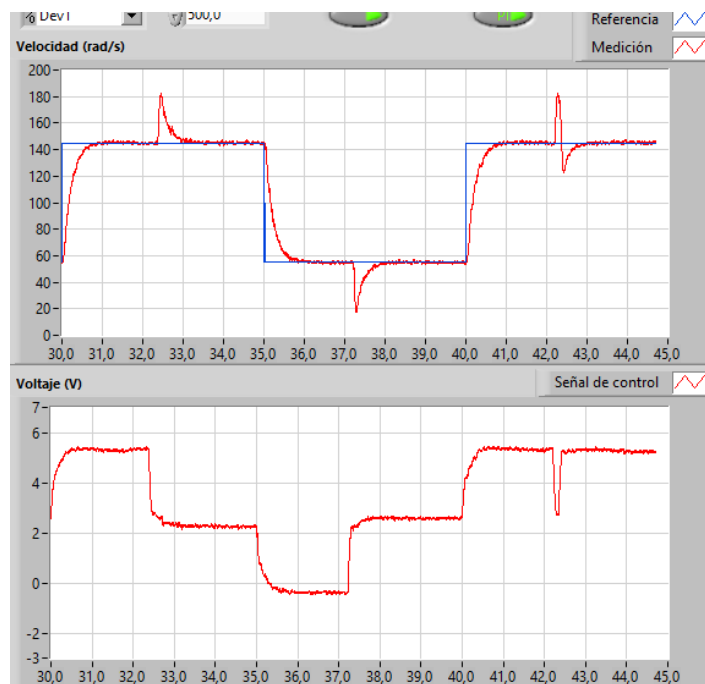


Figura 3.15. Respuesta del sistema ante el rechazo a perturbaciones con controlador PI.

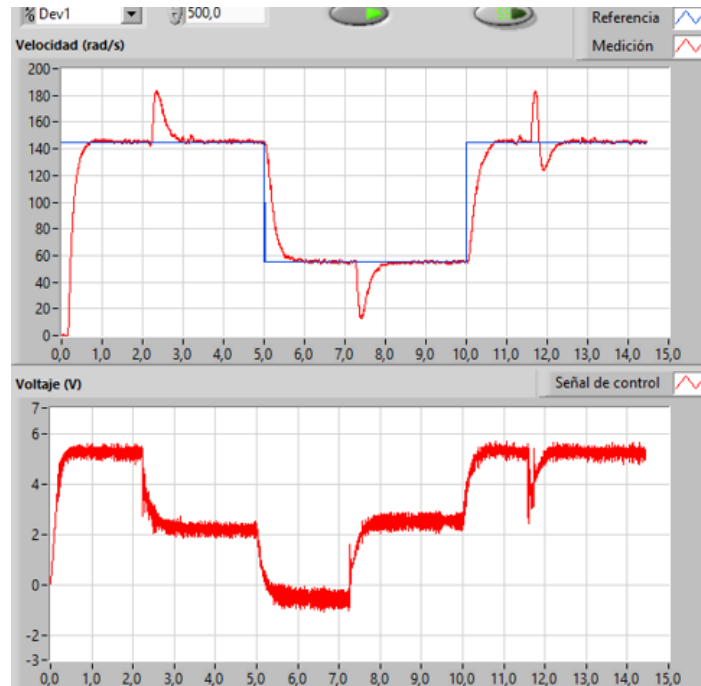


Figura 3.16. Respuesta del sistema ante el rechazo a perturbaciones con controlador en Espacio de Estado.

Se aplicó un perturbación constante de 3 [V] a la señal de control en el primer cambio de referencia la cual fue quitada en el segundo cambio, para el tercer cambio de referencia se aplicó una perturbación instantánea en la salida como si se tratara de una perturbación de carga, en los tres cambios se observa que ambos controladores responden de forma adecuada ante el rechazo a las perturbaciones tanto físicas como digitales.

Los índices de desempeño para la variable velocidad se obtuvieron para una señal de referencia de 145 [rad/s] y se muestran en la Tabla 3.4.1.

Tabla 3.4.1. Cuantificación de los índices de desempeño

Índice		Controlador	ISE	IAE	ISU
Velocidad	Simulado	PI	1064	14.6	138.6
		SS	1785	23.79	121.4
	Implementado	PI	2951.31	36.29	100.92
		SS	3773.18	41.176	100.79

El controlador PI presenta menor índice ISE e IAE tanto para simulación como implementación y se observa un menor índice ISU para el controlador en Espacio de Estados. Se puede tener un menor índice ISE e IAE si se aumenta la ganancia de la parte integral del controlador en Espacio de Estado pero se sacrifica el índice ISU.

3.5 CONTROL DE POSICIÓN EN LABVIEW

Al igual que para la variable posición se implementaron los tres controladores diseñados, el controlador PD sintonizado por las ecuaciones de Camacho, el controlador PD sintonizado mediante las características transitorias deseadas y el controlador por realimentación de estados más la parte integral.

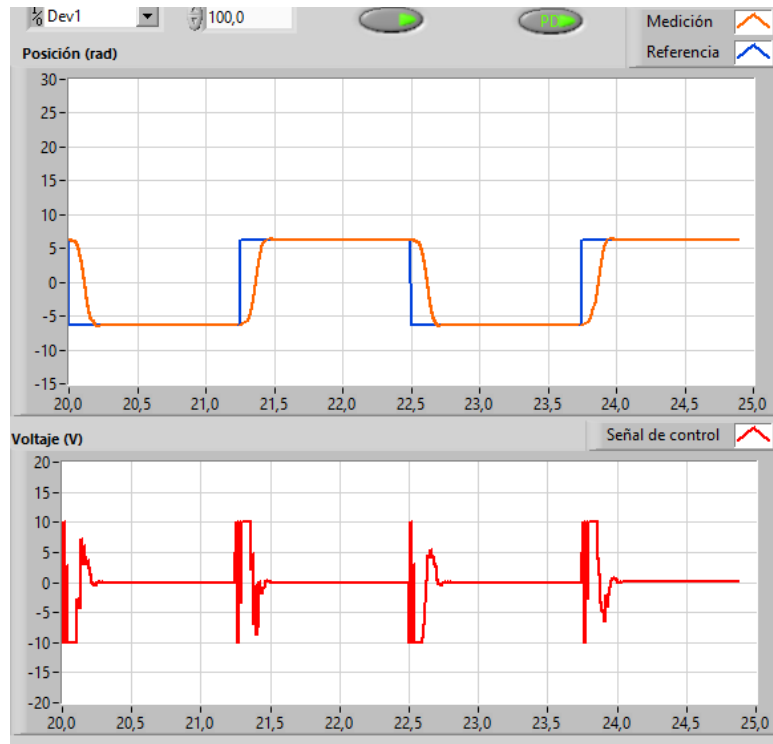


Figura 3.17. Respuesta del sistema y señal de control ante cambios de referencia de posición esquema PD parámetros sintonizados mediante Camacho.

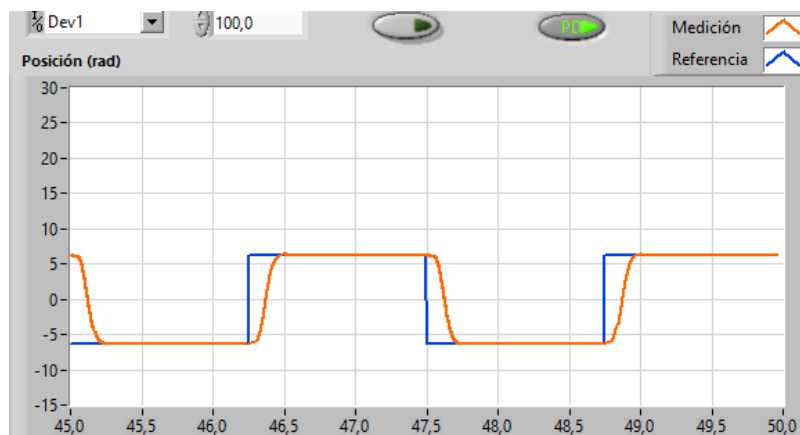


Figura 3.18. Respuesta del sistema ante cambios de referencia de posición esquema PD parámetros sintonizados mediante características transitorias deseadas.

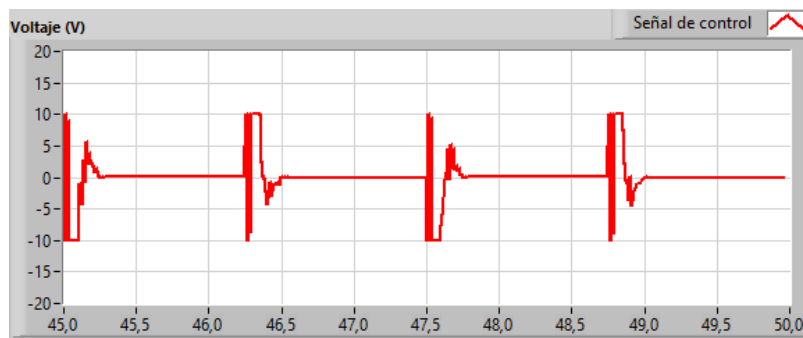


Figura 3.19. Señal de control ante cambios de referencia de posición esquema PD parámetros sintonizados mediante características transitorias deseadas.

Las gráficas indicadas en las Figuras 3.17 y 3.18 obtenidas mediante ambos métodos de sintonización proporcionan salidas suaves con un máximo sobre impulso del 2% y un tiempo de establecimiento de 0.25 [s] e igual que en la simulación la señal de control es muy brusca en este tipo de esquema de control.

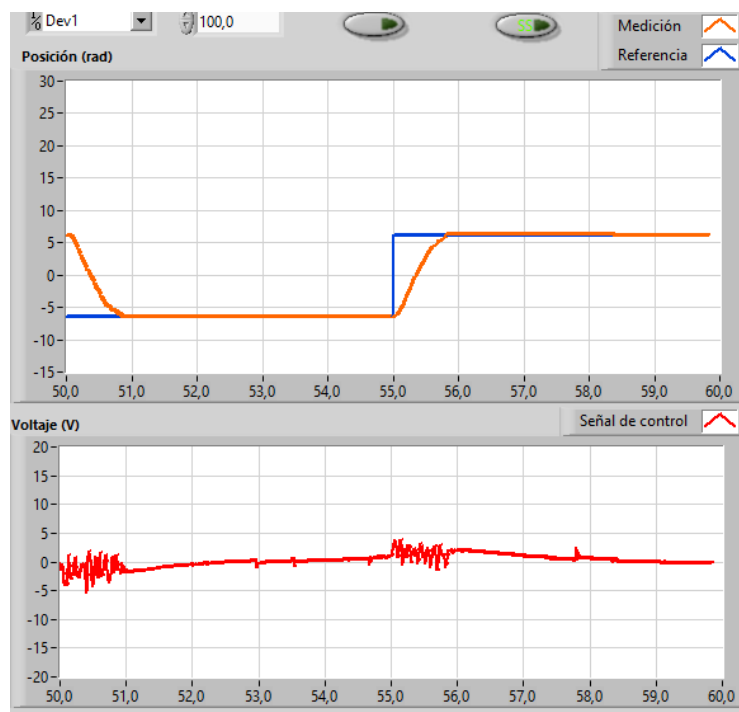


Figura 3.20. Respuesta del sistema y señal de control ante cambios de referencia de velocidad para el esquema de Espacio de Estado.

A diferencia del controlador PD la gráfica indicada en la Figura 3.20 tiene un sobre impulso del 5% y un tiempo de establecimiento de casi 1 [s] si se considera el criterio del $\pm 5\%$ del valor final y el tiempo en el cual se tiene error de posición en estado estable es a los 3.5 [s]. Se puede reducir este tiempo si se aumenta la ganancia de la parte integral del

controlador en Espacio de Estado pero se tiene un mayor sobre impulso lo que para aplicaciones reales no sería factible.

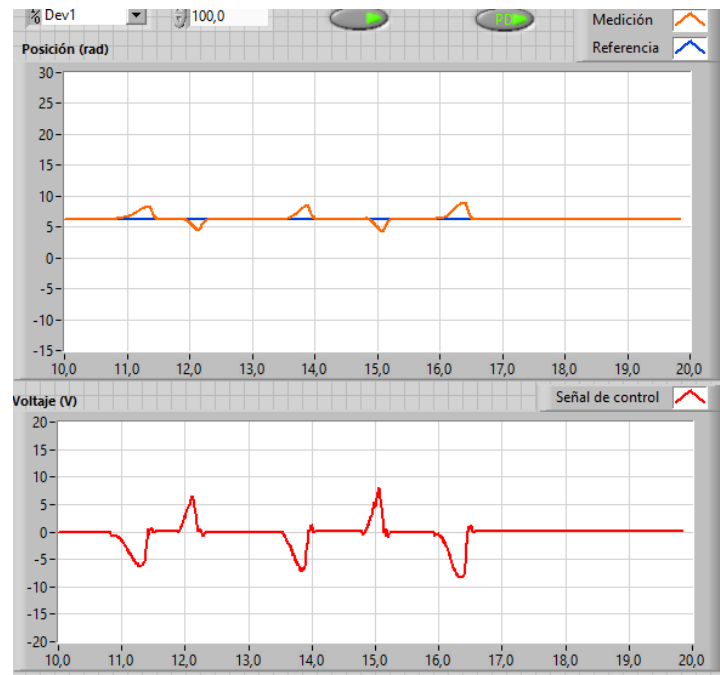


Figura 3.21. Respuesta del sistema y señal de control ante perturbaciones en la salida con el esquema PD parámetros sintonizados mediante características transitorias deseadas.

Las perturbaciones ingresadas al sistema fueron de forma manual moviendo el eje del motor en ambos sentidos y así observar la respuesta la respuesta del controlador ante el ingreso de perturbaciones, como se observa en la Figura 3.21 el esquema PD es suficiente para rechazar perturbaciones instantáneas en la salida del sistema, manteniendo la señal de control dentro de los límites permisibles.

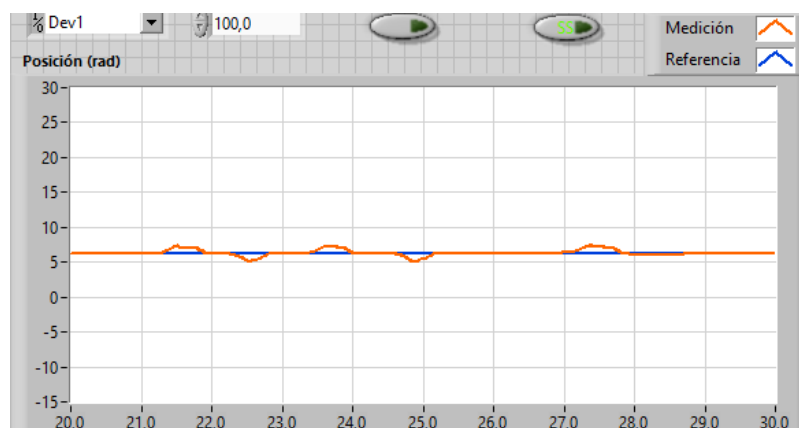


Figura 3.22. Respuesta del sistema ante perturbaciones en la salida del sistema con el controlador en Espacio de Estado.

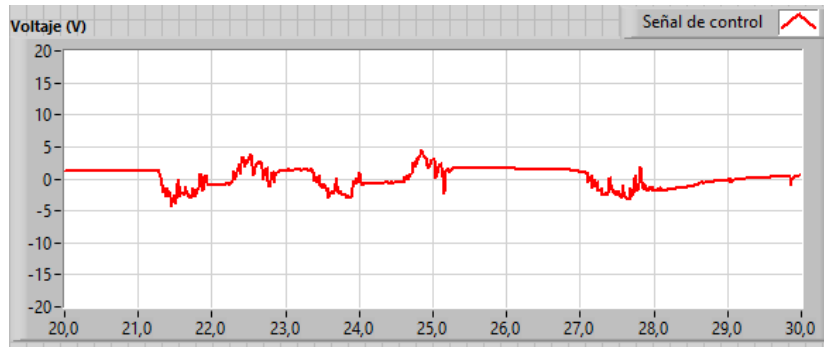


Figura 3.23. Señal de control ante perturbaciones en la salida del sistema con el controlador en Espacio de Estado.

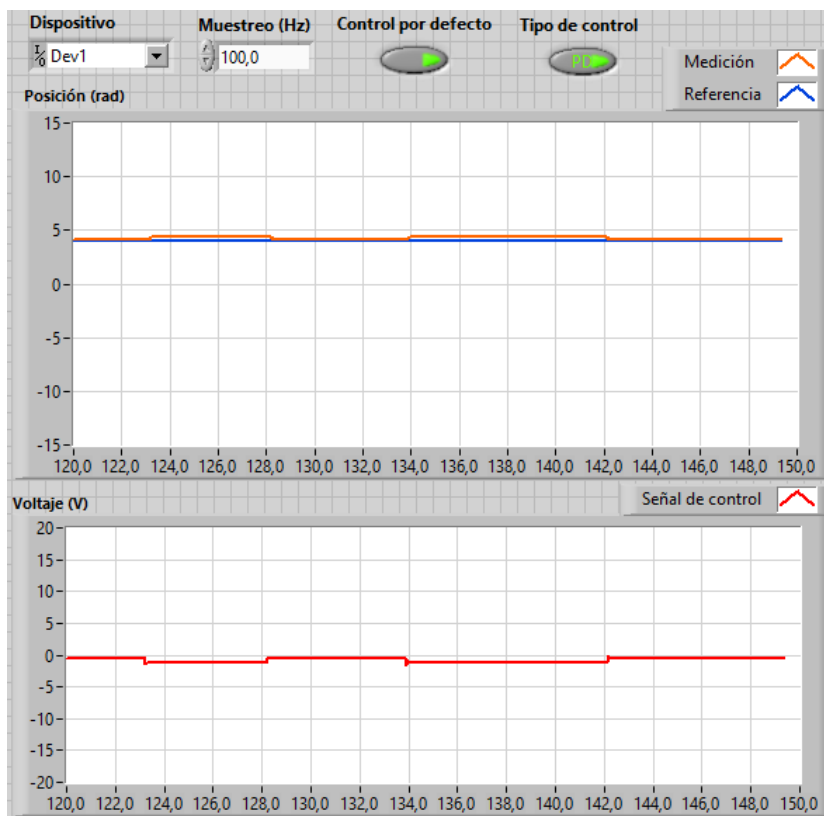


Figura 3.24. Respuesta del sistema y señal de control ante perturbaciones constantes en la señal de control con el esquema PD parámetros sintonizados mediante características transitorias deseadas.

A diferencia de la respuesta obtenida con las perturbaciones en la salida este controlador no es capaz de rechazar perturbaciones en la señal de control, dado que se añade una señal de 3 [V] a la señal de control al instante 123 [s] y se la quita a los 128 [s], adicionalmente se tiene que al momento de ingresar este tipo de perturbaciones el sistema presenta un error de posición en estado estable.



Figura 3.25. Respuesta del sistema y señal de control ante perturbaciones constantes en la señal de control con el esquema en Espacios de Estado

El controlador en Espacio de Estado responde de forma adecuada ante perturbaciones constantes en la señal de control, además se tiene que el tiempo en que se demora en rechazar la perturbación varía dependiendo si se está aumentando los 3 [V] a la señal de control o se los está quitando, es decir, responde más rápido cuando se ingresa un perturbación de entrada. Los índices de desempeño para la variable velocidad se obtuvieron para una señal de referencia de 4 [rad].

Tabla 3.5.1. Cuantificación de los índices de desempeño

Índice		Controlador	ISE	IAE	ISU
Posición	Simulado	PD	0.9164	0.4696	3.369
		SS	2.957	1.145	0.07961
	Implementado	PD	1.254	0.88	6.12
		SS	3.153	1.35	6.14

Los índices ISE e IAE para la variable posición en simulación con el controlador PD son menores a los del controlador en Espacio de Estados lo que implica que en este último se está cometiendo un mayor error pero se tiene un menor esfuerzo de control ya que el índice ISU es significativamente más pequeño a comparación con el ISU del controlador PD. En

cambio los índices ya en la implementación se mantiene que el error cometido con el controlador PD es menor por ende el ISE e IAE son menores pero a diferencia de la simulación el índice ISU para ambos controladores son similares aunque se tiene una señal de control muy diferente para cada uno de los controladores.

3.6 CONCLUSIONES

Como se explicó a lo largo de este trabajo existe una gran variedad de controladores que son diseñados para el control de las variables de un sistema físico, para la realización de este trabajo se controló tanto la velocidad y la posición de un motor DC mediante dos esquemas de control siendo la primera una variante del controlador PID y el otro un esquema basado en Espacio de Estados. Ambos esquemas de control presentaron diferencias en la respuesta del sistema ante cambios de referencia y el rechazo a perturbaciones.

- En conclusión se tiene que el modelo matemático aproximado donde no se considera la inductancia y la viscosidad presenta un error en la ganancia a lazo abierto, por lo cual se utilizó el modelo completo de segundo orden tanto para la sintonización como para la validación del modelo matemático con el sistema físico. Mediante el método de la curva de reacción se aproximó a un modelo de primer orden más retardo y en la validación del modelo se tuvo que tanto el modelo físico como el aproximado presentan la misma curva de reacción cuando se tiene un cambio positivo en la señal de control, en cambio para un cambio negativo se tiene un error en la ganancia de lazo abierto.
- Dentro del análisis de las variantes del controlador PID diseñados para el control de la posición y la velocidad se determinó que presentan un tiempo de estabilización y un índice de rendimiento ISE más bajo que el controlador por realimentación de estados, en cambio este último tiene un índice de rendimiento ISU más bajo que el controlador PI y PD a nivel de simulación, ya en la implementación se observó que los índices ISE e IAE siguen siendo menores para los controladores PID, pero el índice ISU para ambos esquemas de control eran similares a pesar de que su señal de control era diferente.
- De este modo el controlador por realimentación de estado más parte integral presenta mejor robustez que el controlador PD ante el rechazo a perturbaciones constantes en la señal de control para la variable posición. Para la variable

velocidad ambos controladores tienen un buen rechazo ante perturbaciones en la señal de control o perturbaciones en la salida del sistema.

- De lo anterior se deduce que el parámetro que más influye en la respuesta transitoria del sistema con el controlador por realimentación de estados es la constante de la parte integral ya que un valor grande implica un menor tiempo de establecimiento y un aumento en el máximo sobre impulso, en cambio mientras menor sea se tiene una respuesta más suave sin sobre impulso y un mayor tiempo de establecimiento.
- Se desarrolló una interfaz gráfica en AppDesigner a nivel de simulación en el cual se visualizó los resultados de la respuesta del sistema, la señal de control y los índices de desempeño, de igual forma se desarrolló una interfaz gráfica en LabVIEW para comprobar el correcto funcionamiento de los diferentes controladores al someterlos a cambios de referencia tipo onda cuadrada y a perturbaciones no modeladas.

3.7 RECOMENDACIONES

- En la implementación del controlador en Espacio de Estados tener en cuenta la forma en la que se están adquiriendo las señales que se usan para la realimentación de estados, dado que un signo puede ocasionar inconvenientes en la señal de control. Además, se tiene que tarjeta NI ELVIS II trabaja con lógica negativa por lo cual es necesario comprobar si se está cambiando el signo de la señal de control.
- En este trabajo, para la sintonización de los parámetros de los controlador PI y PD se ocupó las formulas propuestas por Oscar Camacho y Dahlin, pero a futuro se recomienda, utilizar el método de lazo cerrado o ganancia ultima (Método de Ziegler- Nichols) ya que este involucra directamente a el sistema físico y no es necesario hacer suposiciones en el parámetro de ajuste o en las características transitorias.
- En futuros trabajos, se recomienda la implementación de controladores donde no sea necesario mediciones de variables físicas para realizar el control a lazo cerrado, como puede ser un controlador SMC o un controlador con Lógica Difusa.
- Para futuras pruebas, se recomienda, modificar el eje del motor para poder acoplar carga y así cuantificar la perturbación que se está ingresando al sistema.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] O. Saleem, «ResearchGate,» Noviembre 2017. [En línea]. Available: https://www.researchgate.net/figure/QNET-DC-Motor-Control-Trainer_fig1_323135319. [Último acceso: 15 Febrero 2022].
- [2] «QNET,» 2011. [En línea]. Available: http://eelabs.faculty.unlv.edu/docs/guides/QNET_DCMCT_User_Manual.pdf. [Último acceso: 15 Febrero 2022].
- [3] M. A. Perez, 2008. [En línea]. Available: <http://dea.unsj.edu.ar/control1b/teoria/unidad1y2.pdf>. [Último acceso: 15 Febrero 2022].
- [4] S. J. Chapman, Máquinas Eléctricas, México D. F.: McGRAW-HILL INTERAMERICANA, 2012.
- [5] S. A. C. Giraldo, «Control Automatico Educacion,» [En línea]. Available: controlautomaticoeducacion.com/analisis-de-sistemas/modelo-de-motor-dc/. [Último acceso: 15 Febrero 2022].
- [6] M. I.-G. Cyde, Maquinas de corriente alterna, Mexico D. F.: Compañía Editorial Continental S. A., 2004.
- [7] I. M. Bragado, Fisica General, Madrid: Notepad Icon, 2004.
- [8] S. Dominguez, Control en el Espacio de Estado, Madrid: Pearson Educacion S. A., 2006.
- [9] O. Camacho, Control de Procesos, Quito: Escuela Politecnica Nacional, 2020.
- [10] V. M. Alfaro, «Identificacion de procesos sobreamortiguados utilizando tecnicas de lazo abierto,» *Revista de la Universidad de Costa Rica*, vol. 11, nº 1-2, p. 16, 2001.
- [11] L. A. G. Garces, 2014. [En línea]. Available: <https://repositorio.utp.edu.co/server/api/core/bitstreams/1a4fb4e4-bc53-4e89-ab3b-cd028b2e8da6/content>. [Último acceso: 15 Febrero 2022].

- [12] A. K. J. Haggund, PID Controllers: theory, design and tunig, Research Triangle Park: Instrument Society of America, 1995.
- [13] «MathWorks,» [En línea]. Available: <https://es.mathworks.com/products/matlab/app-designer.html>. [Último acceso: 15 Febrero 2022].
- [14] L. Carvajal, Metodología de la Investgación Científica. Curso general y aplicado, 28 ed., Santiago de Cali: U.S.C., 2006, p. 139.

5. ANEXOS

Para tener un completo entendimiento del trabajo realizado se tiene los siguientes anexos:

- ANEXO I. Manual de Usuario
- ANEXO II. Programa desarrollado en Matlan-Simulink.
- ANEXO III. Adquisición de datos en LabVIEW.

ANEXO I

MANUAL DE USUARIO

La tarjeta QNET DC MCT se monta fácilmente al sistema NI ELVIS. El procedimiento para la configuración se detalla para usar la QNET con el NI ELVIS II.

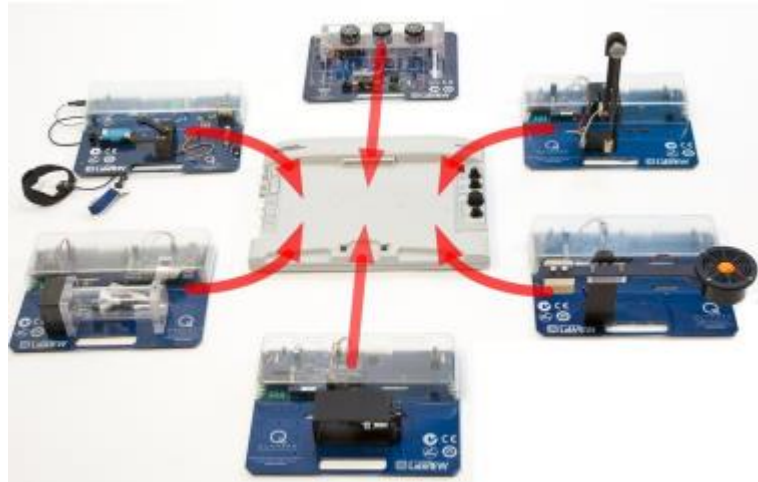


Figura I.1. Conexión de la tarjeta QNET al NI ELVIS II

En el capítulo 1 se detalló los componentes de la tarjeta QNET DC MCT y en la Figura I.2 se indican los componentes de la NI ELVIS y en la Tabla I.1 se detalla cada uno de los componentes.

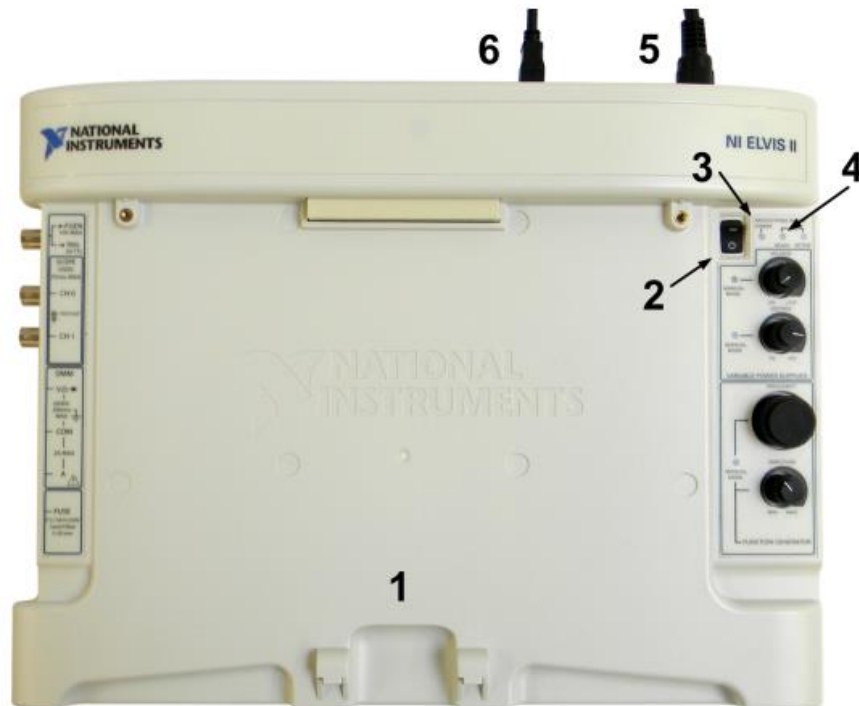


Figura I.2. Componentes de la NI ELVIS.

Tabla I.1. Descripción de los componentes de la NI ELVIS.

ID #	Descripción
1	NI ELVIS II
2	Interruptor de alimentación de la placa
3	LED de alimentación
4	LED de listo
5	Cable de alimentación para NI ELVIS II
6	Conexión USB entre PC y NI ELVIS II

Procedimiento para la configuración del NI ELVIS II

- Deslice el conector PCI del extremo del módulo QNET en el conector hembra del NI ELVIS II.
- Conecte el cable de alimentación del NI ELVIS II.
- Conecte el cable USB ELVIS II A LA PC
- Conecte la alimentación del módulo QNET.
- Encienda el NI ELVIS II con el interruptor ubicado en el panel posterior del sistema.
- Encienda el interruptor de la placa de prototipos.
- Los LEDs de encendido y listo de la unidad NI ELVIS II deben estar encendidos, como se observa en la Figura 3.
- Verifique que los LEDs +15 V, -15 V, +5 V y +B en el módulo QNET estén encendidos, tal como se indica en la Figura 4.



Figura I.3. LEDs de encendido y listo en NI ELVIS II.



Figura I.4. LEDs encendidos del módulo QNET.

Interfaz Gráfica en AppDesigner

En el capítulo 2 se explican todos los componentes de cada una de las pantallas diseñadas en AppDesigner para la simulación de los controladores del motor DC. Se tiene una pantalla de inicio donde consta el nombre del trabajo y se puede escoger entre tipo de variable a controlar, sea esta la velocidad o la posición. Las pantallas diseñadas para la posición o velocidad mantienen una estructura similar por lo cual únicamente se detallara el uso de una de las dos pantallas. Al escoger la variable velocidad aparece una nueva pantalla, tal como se indica en la Figura I.6.



Figura I.5. Pantalla de inicio de la interfaz diseñada en AppDesigner.

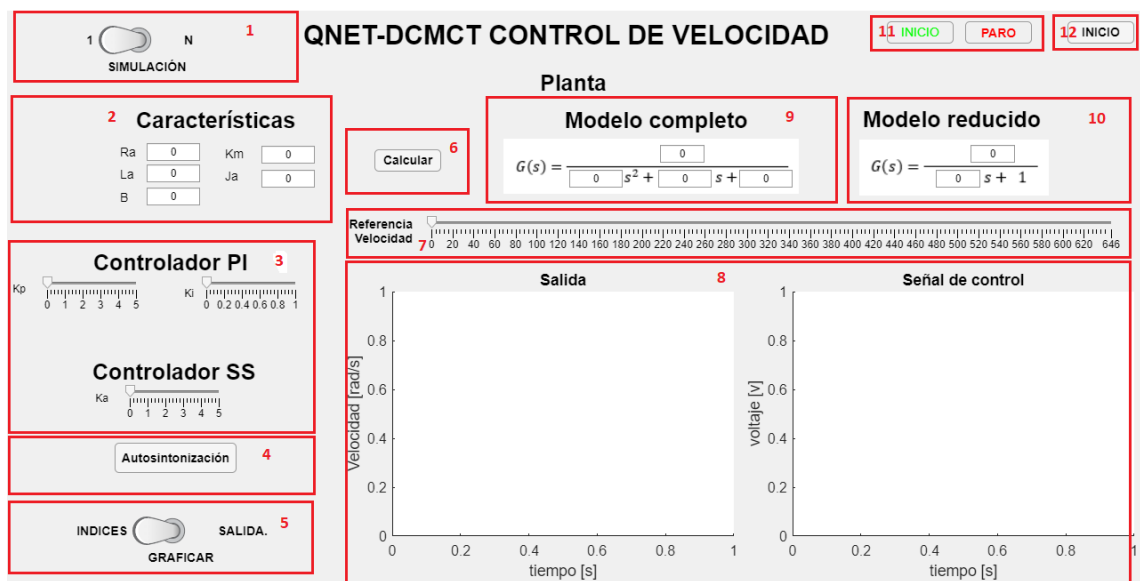


Figura I.6. Pantalla para el control de la velocidad en AppDesigner.

La pantalla esta compuestas de 12 partes que permiten realizar el control de la variable seleccionada observar graficas de salida, señal de control, índices de desempeño y cambiar los parámetros de los controladores a partir de las características eléctricas y mecánicas del motor ingresadas.

Tipo de simulación única

En este tipo de simulación únicamente se puede tener un cambio de referencia, los pasos a seguir se indican a continuación:

- Seleccionar “1” en la parte 1 de la pantalla

- Ingresar las características eléctricas y mecánicas en la parte 2 de la pantalla.
- Para observar el modelo de la planta hacer clic en el botón calcular que es la parte 6.
- Los controladores son calculadas a partir de las características de la planta y se cargan los valores pulsando el pulsador “Autosintonización” ubicado en la parte 4.
- La parte 5 permite graficar la salida y señal de control o los índices de desempeño ISE e ISU dependiendo de la posición en la que se encuentre el interruptor. Se puede seleccionar antes o después de realizar la simulación.
- La parte 7 se escoge la referencia de velocidad.
- Por ultimo en la parte 11 se tiene los botones de Inicio y Paro. Para este tipo de simulación únicamente se requiere presionar el botón de inicio.

Tipo de simulación continua

A diferencia de la simulación única se puede cambiar la referencia mientras se está simulando, para realizar este tipo de simulación se sigue los mismos pasos mencionados anteriormente a excepción del paso 1 donde se debe escoger “N” que significa simulación continua. Al escoger esta opción el tiempo de simulación en Simulink pasa a ser de 50 segundos, se puede cambiar la referencia las veces que se quiera hasta que se acabe el tiempo de simulación o a su vez se puede dar en el botón de Paro ubicado en la parte 12 de la pantalla diseñada.

En la interfaz gráfica diseñada para la posición se tiene los mismos componentes que los indicados en la Figura I.7, por tanto el manejo es la misma para ambos tipos de simulaciones. La única diferencia radica en el tipo de controlador diseñado para la variable velocidad se tenía un controlador PI y un controlador por realimentación de estados más acción integral, en cambio para la variable velocidad son un controlador PD y un controlador por realimentación de estado más acción integral.

Tanto para la variable posición y velocidad se puede variar los parámetros de los controladores PI y PD desde cada una de sus respectivas pantallas, para el caso de los parámetros del controlador en Espacio de Estados únicamente se puede variar la ganancia de la parte integral ya que esta tiene más peso en la señal de control y afecta significativamente a la señal de salida.

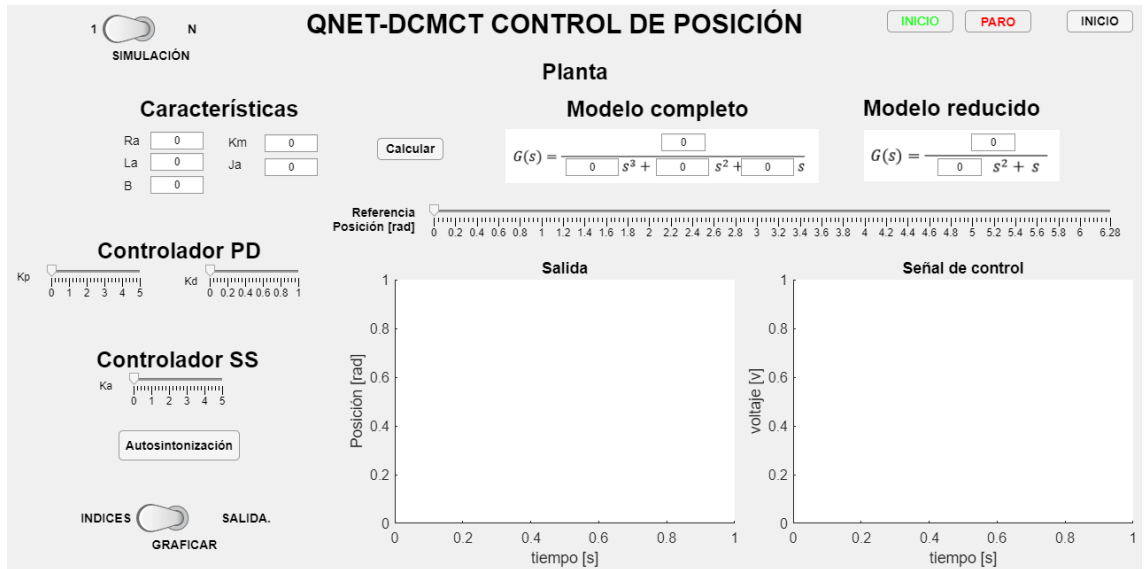


Figura I.7. Pantalla para el control de la posición en AppDesigner.

Interfaz gráfica en LabVIEW

A diferencia de la interfaz gráfica diseñada en AppDesigner esta cuenta únicamente con un tipo de simulación que es la continua. En LabVIEW se diseñó tres pantallas para la prueba de funcionamiento de los controladores para ambas variables y validación del modelo matemático.

Pantalla para validación del modelo

En la Figura I.8 se tiene la pantalla para validación del modelo matemático con el sistema físico, en el capítulo 2 se explicó cada una de las partes con las que esta cuenta. Para utilizar esta pantalla y validar el modelo se siguen los siguientes pasos:

- Conectar el sistema QNET DCMCT a la tarjeta NI ELVIS II como se indicó anteriormente.
- Escoger el nombre del dispositivo con el que la computadora reconoce a la tarjeta.
- En el generador de señal configuramos la señal de entrada.
- En los parámetros del motor se ingresa el modelo de primer orden calculado mediante la curva de reacción, dado que el modelo matemático obtenido es de segundo orden.
- Y finalmente se da al botón de inicio, se observa las gráficas de salida y de ser necesario se ajusta los parámetros de modelado.

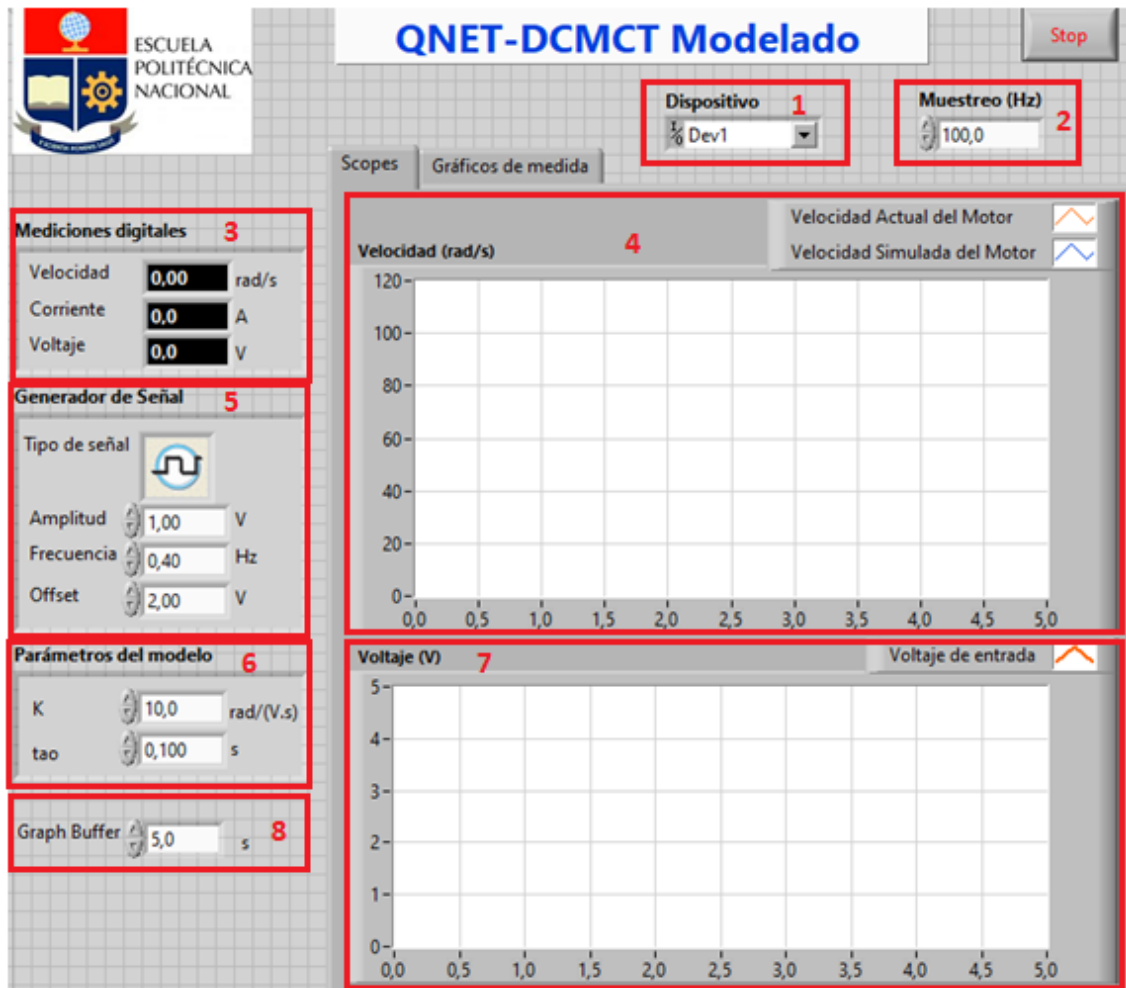


Figura I.8. Pantalla para validación del modelo matemático con el sistema físico.

Pantalla para el control de la velocidad

Los pasos para configurar la pantalla mostrada en la Figura I.9 son similares a la configuración de la pantalla indicada en la Figura I.8, una vez iniciada la simulación permite realizar modificaciones en los parámetros del controlador o el tipo de controlador a utilizar.

La interfaz permite la posibilidad de intercambiar entre el controlador PI y controlador en Espacio de Estado mediante el cambio de estado del botón “Tipo de control” ubicado en la parte 2 de la pantalla mientras se está probando el funcionamiento de uno de los controladores, adicionalmente el botón ubicado en la parte 1 de la pantalla permite al usuario ingresar manualmente los parámetros de los dos controladores y ver la respuesta del sistema en tiempo real ante los cambios realizados. Al tener activo el botón de “Control por defecto” se carga los valores calculados que mejor respuesta del sistema dan.

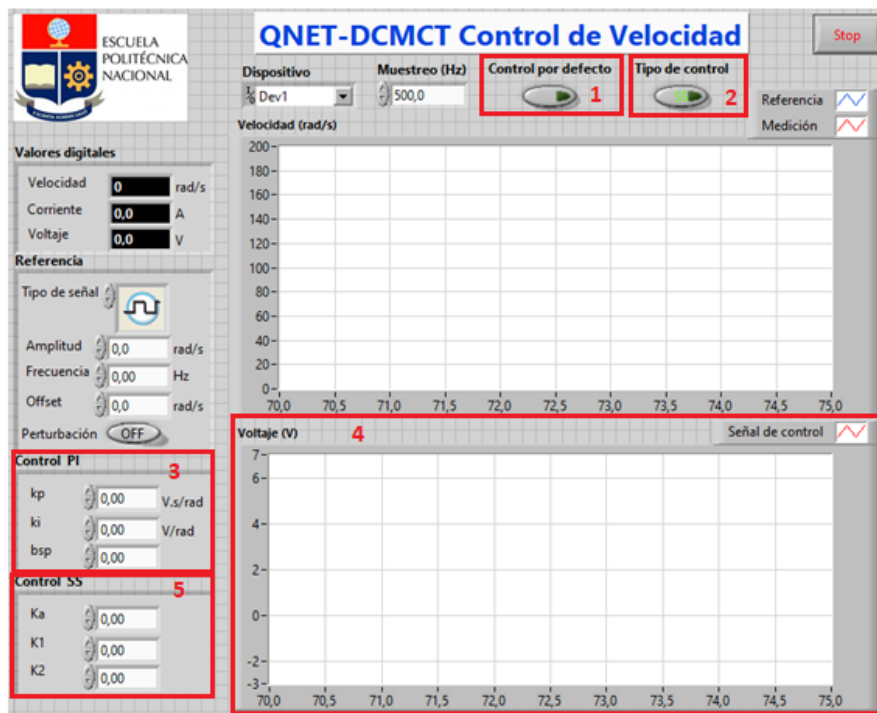


Figura I.9. Pantalla para el control de la velocidad en LabVIEW.

Pantalla para el control de la posición

La configuración de la pantalla indicada en la Figura I.10 es la misma descrita anteriormente la única diferencia radica en los parámetros que se pueden configurar para cada uno de los controladores diseñados.

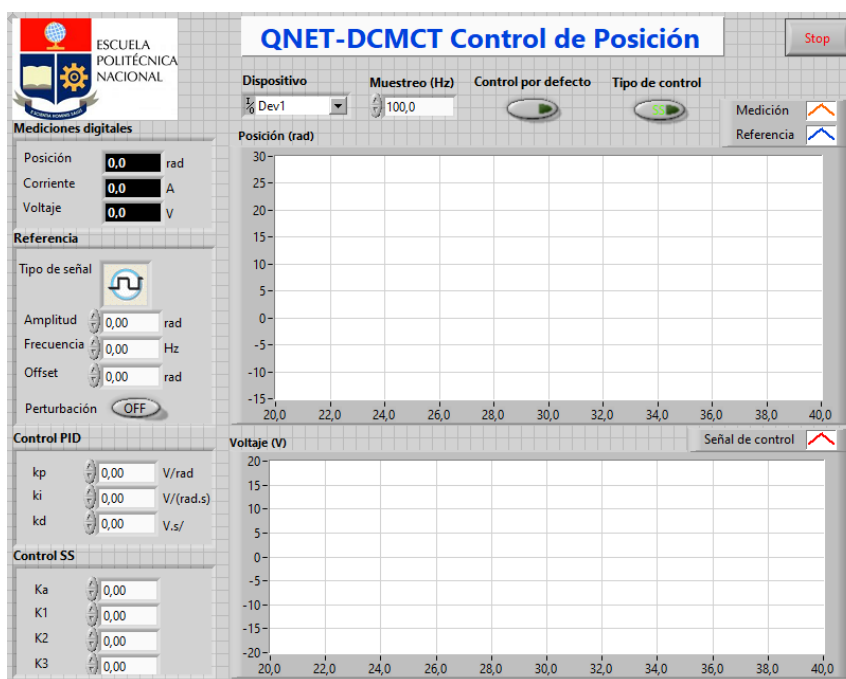


Figura I.10. Pantalla para el control de la posición en LabVIEW.

ANEXO II

En este apartado se indica los esquemas de control implementados en Matlab – Simulink que se utilizaron para realizar la simulación y prueba de funcionamiento de los controladores diseñados.

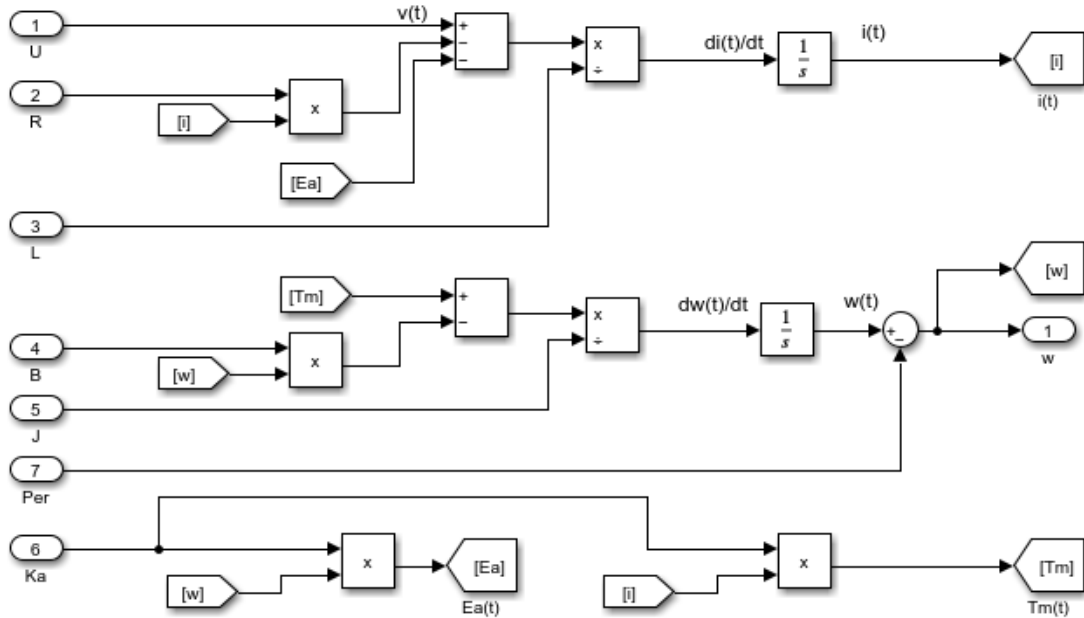


Figura II.1. Implementación de la planta en Matlab – Simulink.

El modelo matemático implementado permite realizar cambios en las características eléctricas y mecánicas del motor DC desde la interfaz gráfica desarrollada en AppDesigner.

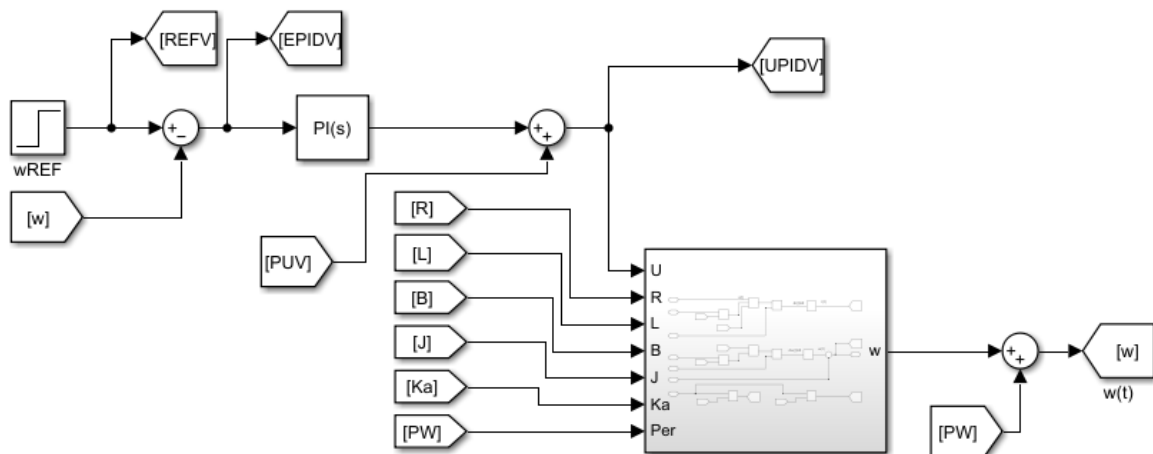


Figura II.2. Esquema de control PI para el control de la velocidad.

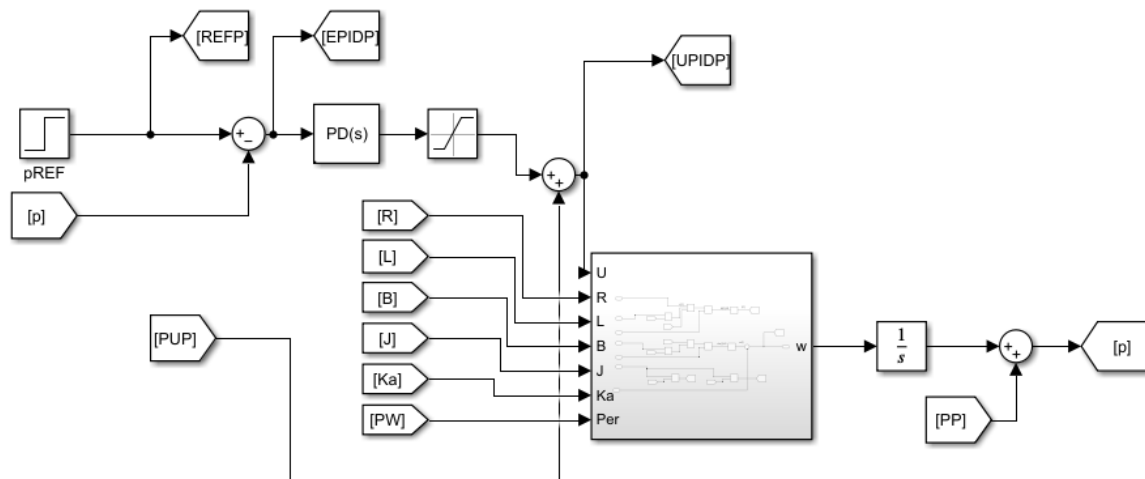


Figura II.3. Esquema de control PD para el control de la posición.

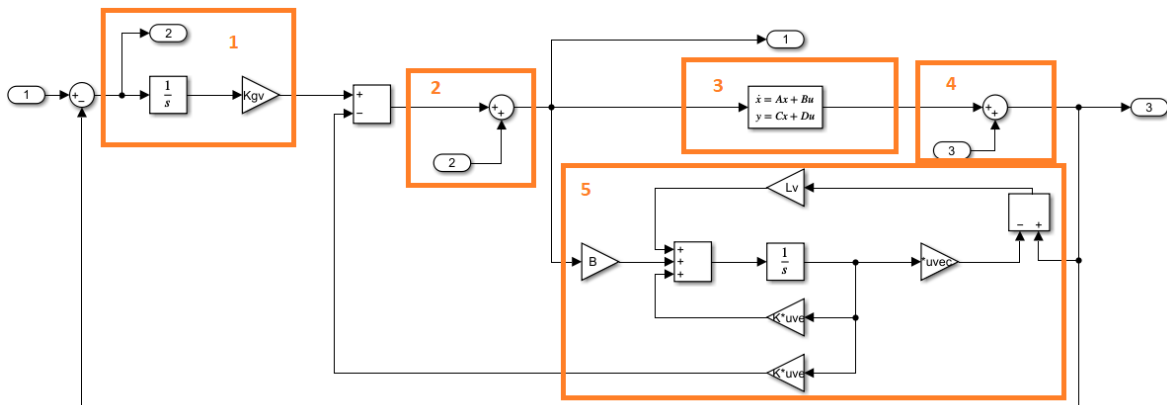


Figura II.3. Esquema de control con realimentación de estados más acción integral para el control de la velocidad

Donde

- 1: Acción integral
- 2: Ingreso de perturbación en la señal de control.
- 3: Representación en Espacio de Estados de la planta,
- 4: Ingreso de perturbación en la salida del sistema.
- 5: Observador de estados de orden completo.

El esquema de control para la variable posición únicamente varía en la forma de representación en Espacio de Estados ya que se tiene tres variables de estados.

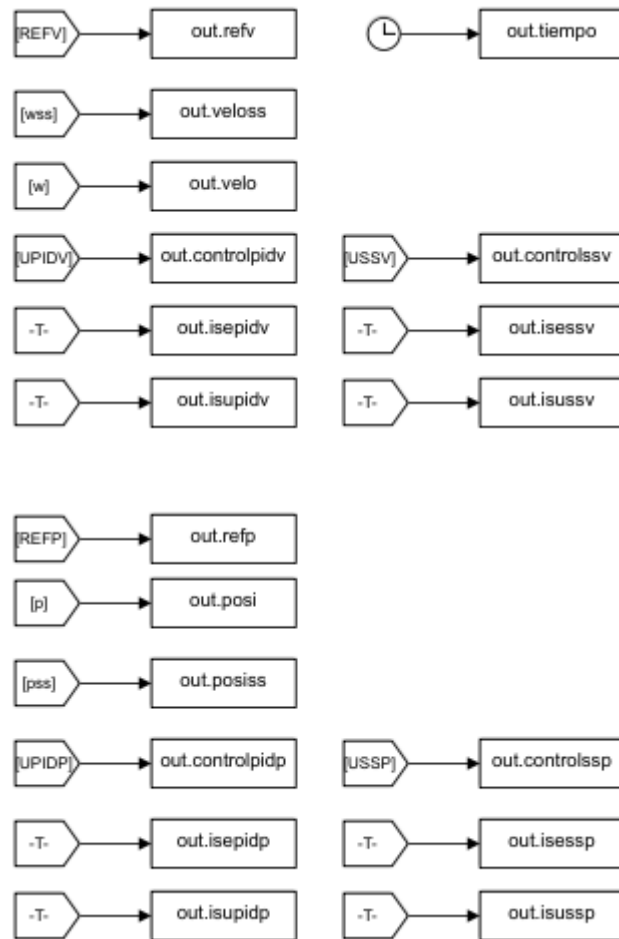


Figura II.4. Variables enviadas al Workspace para graficar en la interfaz grafica de AppDesigner.

ANEXO III

En este apartado se detalla el proceso de configuración de cada uno de los canales de la NI ELVIS II ocupados para realizar el presente trabajo.

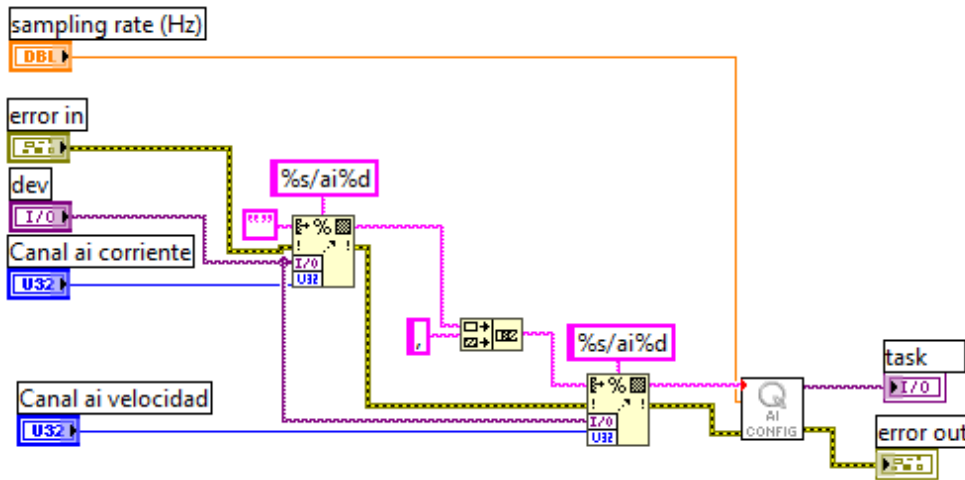


Figura III.1. Configuración de los canales de entrada de corriente y velocidad.

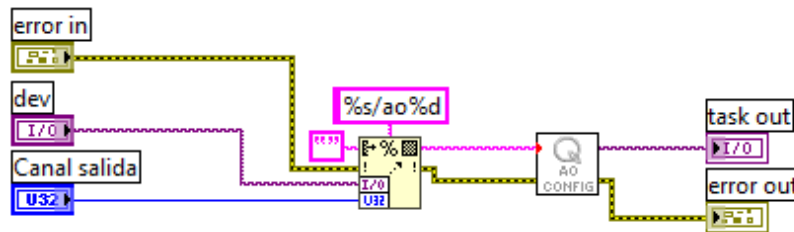


Figura III.2. Configuración del canal de salida del voltaje de alimentación.

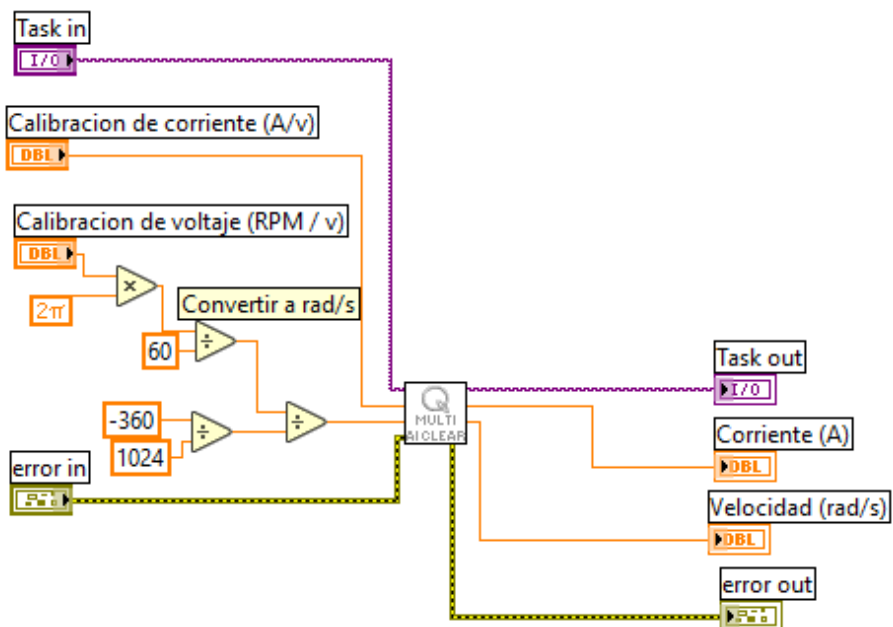


Figura III.3. Acondicionamiento de los datos leídos desde la NI ELVIS II.

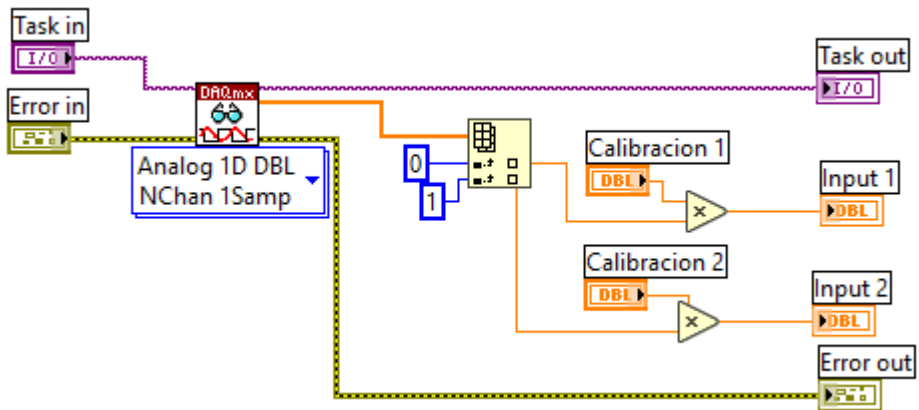


Figura III.4. Lectura de los datos obtenidos con la NI ELVIS II, corriente y velocidad.

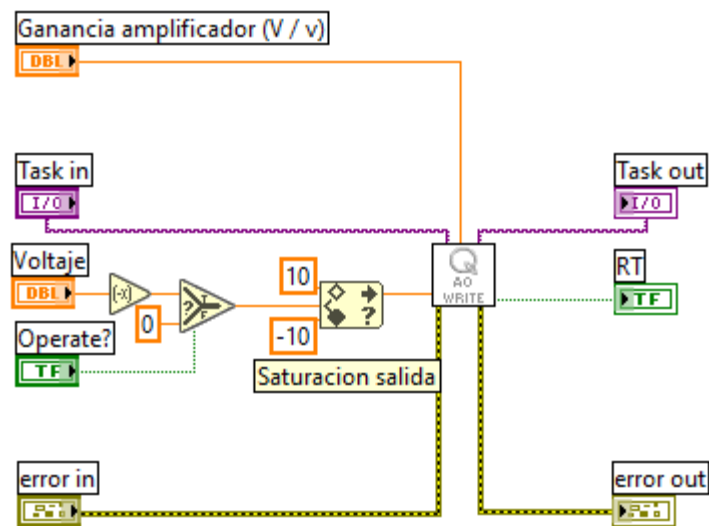


Figura III.5. Saturación de la señal de control y cambio de signo.

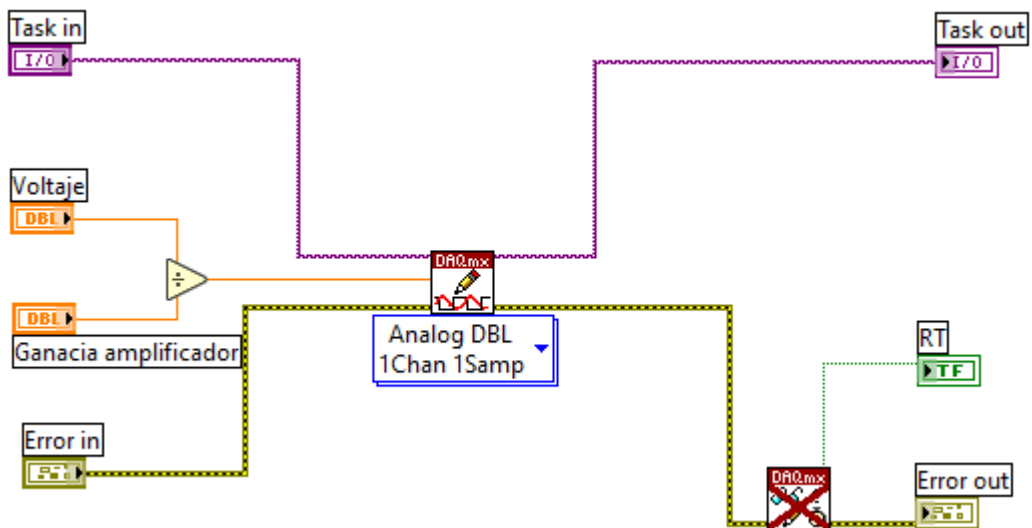


Figura III.6. Escritura del voltaje de alimentación del motor DC.

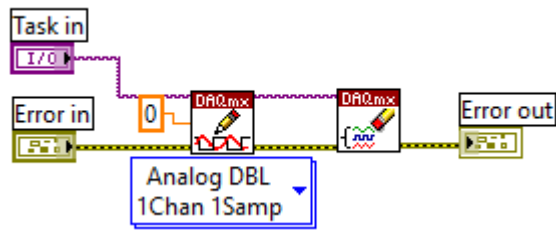


Figura III.7. Escritura de voltaje nulo cuando se detiene la simulación.