

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

**CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN
Y CONTROL**

**PROYECTO DE GRADO PARA LA OBTENCIÓN DEL TÍTULO EN
INGENIERÍA**

**DESARROLLO E IMPLEMENTACIÓN DE CONTROLADORES PARA
LA PLANTA DE VELOCIDAD *DC SPEED CONTROL SYSTEM*
MEDIANTE LA UTILIZACIÓN DE LA HERRAMIENTA RTW (REAL-
TIME WORKSHOP) DE MATLAB**

ROBERTO JAVIER GONZÁLEZ PUMA

CÉSAR ANDRÉS VILLACÍS FIALLOS

SANGOLQUÍ – ECUADOR

2012

Declaración de Responsabilidad

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

DECLARACIÓN DE RESPONSABILIDAD

ROBERTO JAVIER GONZÁLEZ PUMA

CÉSAR ANDRÉS VILLACÍS FIALLOS

DECLARAMOS QUE:

El proyecto de grado denominado: “Desarrollo e implementación de controladores para la planta de velocidad DC Speed Control System mediante la utilización de la herramienta RTW (Real-Time Workshop) de Matlab”, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan al pie, de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, 8 de Noviembre del 2012

Roberto Javier González Puma

César Andrés Villacís Fiallos

Autorización de publicación

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORIZACIÓN

Nosotros, Roberto Javier González Puma y César Andrés Villacís Fiallos

Autorizamos a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la institución del trabajo “Desarrollo e implementación de controladores para la planta de velocidad DC Speed Control System mediante la utilización de la herramienta RTW (Real-Time Workshop) de Matlab”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Sangolquí, 8 de Noviembre del 2012

Roberto Javier Gonzales Puma

César Andrés Villacís Fiallos

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

CERTIFICADO

Ing. Víctor Proaño
Ing. Rodolfo Gordillo

CERTIFICAN

Que el trabajo titulado “Desarrollo e implementación de controladores para la planta de velocidad DC Speed Control System mediante la utilización de la herramienta RTW (Real-Time Workshop) de Matlab”, realizado por Roberto Javier González Puma y César Andrés Villacís Fiallos, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

Debido a que se trata de un trabajo de investigación recomiendan su publicación.

El mencionado trabajo consta de un documento empastado un disco compacto el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan a Roberto Javier Gonzales Puma y César Andrés Villacís Fiallos que lo entreguen al Ingeniero Víctor Proaño, en su calidad de coordinador de la Carrera.

Ing. Víctor Proaño

Ing. Rodolfo Gordillo

CERTIFICACIÓN

Por medio de la presente certificamos que el proyecto de grado, titulado “DESARROLLO E IMPLEMENTACIÓN DE CONTROLADORES PARA LA PLANTA DE VELOCIDAD DC SPEED CONTROL SYSTEM MEDIANTE LA UTILIZACIÓN DE LA HERRAMIENTA RTW (REAL-TIME WORKSHOP) DE MATLAB”, ha sido desarrollado en su totalidad por el señor ROBERTO JAVIER GONZÁLEZ PUMA con C.I.: 171422706-1 Y CÉSAR ANDRÉS VILLACÍS FIALLOS con C.I.: 180426303-4, bajo nuestra dirección.

Atentamente,

Ing. Víctor Proaño
DIRECTOR

Ing. Rodolfo Gordillo
CODIRECTOR

RESUMEN

El presente documento describe los controladores PI, PID, Digital e Inteligentes para el control de velocidad de la unidad DC Speed Control System.

Los controladores PI, PID y PI FUZZY utilizan una técnica de optimización para encontrar sus respectivas constantes. Para los controladores Predictivo, Neuronal por modelo de referencia y Digital están basados en sus respectivas teorías de control. Adicionalmente se utilizó el método de Smith para realizar el modelo matemático de la unidad.

La implementación de cada uno de los controladores en tiempo real se lo realiza por medio de la herramienta Real Time Workshop de Matlab y la tarjeta de adquisición de datos NI PCI 6221

.

AGRADECIMIENTO

A mi madre por darme un hogar estable en el que juntos aprendimos a sobrellevar nuestras dificultades para poder salir adelante.

A mis tíos por ser mis hermanos y padres cuando más los necesitaba.

A mis abuelitos por ser las personas que con su ejemplo me han ayudado a crecer en una familia llena de amor y cariño.

A mi padre por darme su tiempo para escucharme y poder conversar como los buenos amigos que éramos.

A Dios por estar siempre conmigo y darme la fuerza, paciencia y sabiduría para poderme guiar a mi mismo en todos estos años que he vivido acá solo en Quito.

Al Ing. Víctor Proaño e Ing. Rodolfo Gordillo, por brindarnos su tiempo y tenernos paciencia para lograr la culminación de este proyecto.

Finalmente quisiera agradecer a mis amigos del colegio que juntos hemos caminado cada quien por su camino, dándonos aliento cuando uno está caído para que de alguna forma esperar este momento la culminación de nuestras respectivas carreras.

Andrés Villacís F.

AGRADECIMIENTO

Quiero iniciar expresando mi agradecimiento a Dios y a la santísima Virgen por la fuerza espiritual que me supieron brindar en el transcurso de mi vida de estudiante y de la formación de mi familia.

A mis padres quienes desde el primer momento que inicio mi vida me brindaron su apoyo, confianza y su fortaleza para poder alcanzar mis sueños.

A mi madre María Inés Puma por su apoyo incondicional, por todo su amor y por su enseñanza de siempre ser positivo y nunca rendirme ante las dificultades que la vida nos tiene en el camino.

A mi padre Juan González por ser un guía en todo momento, por su confianza y sabiduría que me ayudaron en los momentos más difíciles de mi vida, brindándome la fortaleza de su espíritu y su amor incondicional.

A mi hermano Juan Carlos con quien compartí gran parte de mi vida, por siempre impulsarme alcanzar la mejor preparación en todos los aspectos personales y profesionales.

A mi incondicional esposa Karolina López, que con su amor y su fuerza de espíritu supo soportar los momentos más difíciles en el arduo trabajo de formar el camino correcto de la familia con quien logre sobresalir y alcanzar el éxito profesional y familiar, quien me brindo la oportunidad de sentir una de las más grandes alegrías que un hombre puede recibir en su corazón y en toda su vida, mis princesas Camila y Emily, quienes son mi dulce inspiración y mi amor más grande de las cuales me siento muy orgulloso.

Javier González P.

DEDICATORIA

A mi familia por su paciencia y acogida cuando pase los momentos más difíciles de mi vida.

A Dios por darme la sabiduría para resolver los diferentes problemas que aparecían en mi vida estudiantil.

A mis amigos que con sus consejos, puntos de vista y el “ya a de ser hora de que termines”, que me motivo a continuar y culminar mi carrera.

A mi papá que donde quiera que este se que él está muy orgullo de mi y feliz porque al fin logre terminar su más grande y anhelado sueño, verme convertido en Ingeniero.

Andrés Villacís F.

DEDICATORIA

La vida es hermosa, tanto así que nos brinda felicidades y tristezas, problemas y soluciones, tuve el honor de recibir una de ellas la cual afecto y produjo un cambio en mi vida, pero logre superar hasta el obstáculo más grande que percibí en mi camino, dedico este trabajo a todas las personas que en algún momento de su vida sintieron miedo y el deseo de rendirse y dejar de lado un sueño.

Va dedicado a toda mi familia en especial a mi hijas, las cuales en algún momento de su vida podrán conocer la gran inspiración de trabajo y alegría que crearon en mi persona, para quienes espero ser un ejemplo valioso donde puedan refugiar sus inquietudes y tengan siempre en cuenta que cualquier adversidad se la puede superar con luchan, dedicación y entrega, teniendo mucho en cuenta a DIOS quien nunca las dejara solas, porque tengo la plena experiencia de decir que Dios siempre estuvo con seguridad en mi vida.

A mis buenos amigos que siempre los recuerdo por todos esos momentos de dificultades y alegrías que pasamos juntos, con los cuales soñamos tener un gran futuro en nuestro camino profesional y familiar, a quienes les deseo el mayor de los éxitos en toda su vida.

Javier González P.

PRÓLOGO

La Escuela Politécnica del Ejército posee diferentes unidades de entrenamiento como la DC SPEED CONTROL SYSTEM, utilizadas para realizar las prácticas de laboratorio de la cátedra de control de procesos, en la cual el estudiante únicamente se dedica al diseño del controlador.

El presente proyecto de grado abarca el estudio, diseño e implementación de controladores: PI, PID, digital, predictivo, difuso y neuronal para controlar la velocidad en la unidad DC SPEED CONTROL SYSTEM mediante la utilización de la herramienta RTW (Real-Time Workshop) de Matlab y la tarjeta NI PCI 6221 que permitieron ejecutar los controladores en tiempo real, además se realiza la identificación de la planta por medio del método de Smith que nos proporciona un modelo matemático de la unidad.

Este proyecto pretende brindar más información sobre el diseño e implementación de los controladores mencionados en un sistema real de velocidad.

CONTENIDO

INTRODUCCIÓN	14
1.1 Antecedentes.....	14
CAPÍTULO II	17
2.1 Características de la unidad de velocidad DC Speed Control System.....	17
2.2 Características de la tarjeta de adquisición de datos National Instrument PCI-6221	20
2.2.1 Características de las entradas y salidas analógicas de la tarjeta de adquisición de datos National Instrument PCI-6221	23
2.3 Sistema de control mediante la herramienta Real-Time Workshop de Matlab	23
CAPÍTULO III	25
MODELAMIENTO MATEMÁTICO E IDENTIFICACIÓN DE LA UNIDAD DE VELOCIDAD DC SPEED CONTROL SYSTEM	25
3.1. Modelamiento Matemático	25
3.2. Identificación del sistema.....	27
CAPÍTULO IV.....	38
APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES PI Y PID	38
4.1 Especificación del problema.....	38
4.2 Marco Teórico	38
4.3 Solución del problema.....	39
4.4 Simulación.....	44
4.5 Implementación.....	48
CAPÍTULO V	53
APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES DIGITALES	53
5.1 Controlador Digital.....	53
5.2 Introducción:.....	53
5.3 Análisis para la Ubicación de polos.....	54
5.4 Ecuación Diofantina:	60

5.5	Simulación.....	66
5.6	Implementación.....	70
CAPÍTULO VI.....		74
APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES		
INTELIGENTES		74
7.1	Controlador Predictivo.....	74
7.1.1	<i>Marco Teórico</i>	74
7.1.2	<i>Desarrollo</i>	75
7.2	Controlador PI Fuzzy.....	78
7.2.1	<i>Marco Teórico</i>	78
7.2.2	<i>Desarrollo</i>	80
7.3	Controlador Neuronal por modelo de referencia	87
7.3.1	<i>Marco Teórico</i>	87
7.3.2	<i>Desarrollo</i>	87
CAPÍTULO VII.....		92
CONCLUSIONES Y RECOMENDACIONES		92
8.1	Conclusiones.....	92
8.2	Recomendaciones.....	93
REFERENCIAS BIBLIOGRAFIA.....		94

CAPÍTULO I

INTRODUCCIÓN

1.1 Antecedentes

El diseño e implementación de controladores para procesos industriales en la actualidad es una de las especialidades relevantes en la Ingeniería de Control Automático.

Si bien las técnicas de control sobre sistemas lineales ya han sido probadas con suficiencia, en la práctica los sistemas son no lineales. Esto obliga a que se hagan ajustes en la sintonización de los controladores, para que sean útiles en una configuración experimental.

El motor DC es uno de los sistemas dinámicos más utilizados en la prueba de algoritmos de control debido a que su función de transferencia puede ser reducida a un sistema lineal de primer orden; por esto se ha convertido en el ejemplo obligado en los textos de control clásicos. Además es un dispositivo económico, tanto así que es popular en la generación de movimiento, rotacional o lineal; también en procesos más complejos, como en la robótica, la aviónica y la automatización de procesos en general, entre muchos más.

La unidad DC Speed Control System, perteneciente a los laboratorios de Control Industrial del Departamento de Eléctrica y Electrónica de la Escuela Politécnica del Ejército, es un módulo de entrenamiento con el cual, los alumnos pueden implementar diseños de controladores a través de sus entradas analógicas, permitiendo realizar el control de la unidad por diferentes medios como: tarjetas de adquisición de datos conectados hacia un computador, PLCs, FPGA y microcontroladores.

Sin embargo, el módulo de velocidad es utilizado para pruebas de laboratorio de control de procesos a través de controladores de tipo PI y PID diseñados e implementados en un PLC.

Uno de los softwares comúnmente utilizados a la hora de realizar el diseño e implementación de controladores es Matlab que con la herramienta Real Time Workshop permite modelar algoritmos generados en C como bloques de Simulink y códigos de Matlab.

Con el propósito de obtener un sistema didáctico que permita a los estudiantes conocer paso a paso el diseño e implementación de controladores para la Unidad DC Speed Control System por medio de la herramienta Real Time Workshop de Matlab se diseñó un modelo matemático que simula la unidad DC Speed Control System en un rango de funcionamiento.

Se utilizó una tarjeta de adquisición de datos DAQ PCI-6221 para la obtención de la señal de respuesta que tiene la unidad DC Speed Control System a diferentes rangos de entrada de voltaje.

Se realizó la planta que determina las constantes de tiempo, tiempo muerto y ganancia de la unidad DC Speed Control System para definir un modelo matemático para la unidad.

El Diseño e implementación se lo realizó en Simulink. Los controladores implementados para la unidad, fueron: PI, PID, Digital, PI FUZZY, Predictivo y Neuronal por modelo de referencia.

En el caso de los controladores PI y PID se resolvió un problema de optimización en Matlab, que proporcionó las constantes K_p , K_i , K_d respectivamente, que generaron la mejor respuesta del sistema ante los diferentes cambios que fue obteniendo el modelo matemático al momento de su diseño.

Finalmente, se realizó experimentos en tiempo real, con la ayuda de la herramienta Real Time Workshop de Matlab para acoplar los controladores diseñados e implementados con el modelo matemático a la unidad de velocidad DC Speed Control System, donde se observó las variables de error, control y salida del sistema controlado, producidas al manipular la entrada del sistema.

Una de las razones por las cuales se utiliza el PC es porque se espera que este desarrollo se utilice en la enseñanza del control, y el PC resulta adecuado para el almacenamiento, la manipulación y la visualización de información.

El programa sobre el cual se prueban las diferentes técnicas de control es Matlab; en particular se usa la herramienta de trabajo en tiempo real, la cual permite la conexión con Simulink; así otras herramientas de Matlab son aprovechadas de manera directa, por ejemplo, el diseño de filtros, diseño de sistemas de control SISO o de la herramienta de identificación.

CAPÍTULO II

CARACTERÍSTICAS DEL SISTEMA A IMPLEMENTAR

2.1 Características de la unidad de velocidad DC Speed Control System

El control preciso de la velocidad es un requisito en muchos procesos industriales porque aumenta la calidad de los productos y el volumen la producción, disminuyendo su precio de producción.

En la figura 2.1 se puede observar la unidad de velocidad DC Speed Control System la cual posee los siguientes módulos: interfaz de motor, medidor M2, freno electromagnético, entre otros.

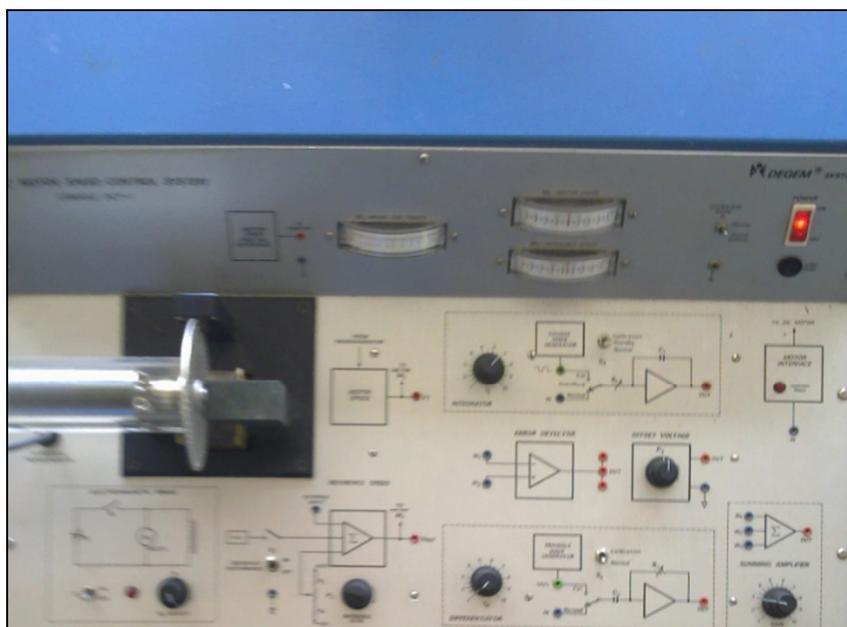


Figura 2.1 Unidad de velocidad DC SPEED CONTROL SYSTEM

La **interfaz de motor**, sirve para energizar al motor DC. Maxon cuyas características se encuentran en la figura 2.2. Para conocer los valores de velocidad que genera el motor DC se acopla un tacómetro que convierte la

energía rotacional producida por el eje del motor en energía eléctrica, la estructura del tacómetro acoplado al motor DC se encuentra en la figura 2.3.



Figura2.2: Motor DC Maxon



Figura 2.3: Tacómetro

El Medidor M2 es un galvanómetro que mide la señal de voltaje DC generada por el tacómetro, el cual permite al usuario observar las variaciones de voltaje en un rango de -4000 a 4000 RPM.

El Freno Electromagnético es un dispositivo electromagnético que actúa directamente sobre un disco de aluminio acoplado al eje del motor demostrado en la figura2.4.

Tiene como función frenar (desacelerar) la velocidad del motor eléctrico; en la unidad de velocidad DC Speed Control System el freno electromagnético puede ser activado en cualquier momento en el que se requiera frenar el motor como se

muestra en la figura 2.5, adicionalmente, posee una fuente de voltaje variable V_b con la cual podemos ajustar la fuerza del campo electromagnético.

De la misma manera, en el Medidor M2, el freno electromagnético está conectado y acoplado a un galvanómetro que permite al usuario observar la potencia del campo electromagnético.



Figura2.4: Acoplamiento disco-eje del motor.

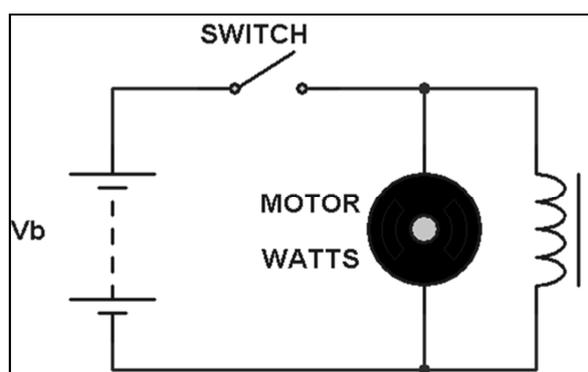


Figura2.5: Circuito del freno electromagnético

Los demás módulos, son circuitos analógicos que permiten realizar controladores de tipo P, PI, PID, mismos que no son utilizados en el presente trabajo.

2.2 Características de la tarjeta de adquisición de datos National Instrument PCI-6221

La PCI-6221 de National Instruments (figura2.6) es una tarjeta de adquisición de datos (DAQ) multifunción de bajo costo con un formato físico tipo PCI, permitiéndole funcionar en diferentes sistemas operativos como Windows, Linux, Mac, entre otros.

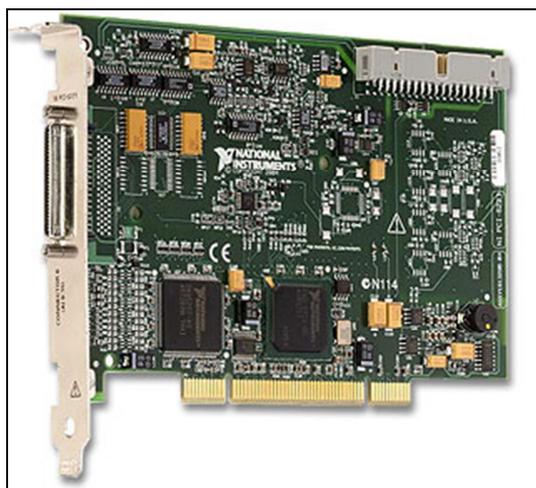


Figura2.6: Tarjeta DAQ PCI-6221

Con este tipo de tarjeta de adquisición de datos, podemos enviar desde el computador y recibir desde el exterior la señal producida por sensores y transductores; este tipo de señales tanto de entrada como de salida son del tipo digital, analógico, frecuencia.

Para conectar las entradas y salidas de la tarjeta DAQ PCI-6221 se lo realiza mediante un cable NI SHC68-68.EPM (figura2.7), el cual posee cables cruzados blindados individuales para las entradas y salidas analógicas y digitales.



Figura2.7: Cable NI SHC68-68.EPM

Como elemento final de conexión de las señales se tiene un conector de tipo SCB-68 (figura 2.8), este conector, posee 68 pines de montaje vertical ajustados por tornillos para su fácil utilización. La distribución que posee el conector de tipo SCB-68 se lo muestra en la figura 2.9.



Figura 2.8: Conector SCB-68

AI 0	68	34	AI 8
AI GND	67	33	AI 1
AI 9	66	32	AI GND
AI 2	65	31	AI 10
AI GND	64	30	AI 3
AI 11	63	29	AI GND
AI SENSE	62	28	AI 4
AI 12	61	27	AI GND
AI 5	60	26	AI 13
AI GND	59	25	AI 6
AI 14	58	24	AI GND
AI 7	57	23	AI 15
AI GND	56	22	AO 0
AO GND	55	21	AO 1
AO GND	54	20	NC
D GND	53	19	P0.4
P0.0	52	18	D GND
P0.5	51	17	P0.1
D GND	50	16	P0.6
P0.2	49	15	D GND
P0.7	48	14	+5 V
P0.3	47	13	D GND
PFI 11/P2.3	46	12	D GND
PFI 10/P2.2	45	11	PFI 0/P1.0
D GND	44	10	PFI 1/P1.1
PFI 2/P1.2	43	9	D GND
PFI 3/P1.3	42	8	+5 V
PFI 4/P1.4	41	7	D GND
PFI 13/P2.5	40	6	PFI 5/P1.5
PFI 15/P2.7	39	5	PFI 6/P1.6
PFI 7/P1.7	38	4	D GND
PFI 8/P2.0	37	3	PFI 9/P2.1
D GND	36	2	PFI 12/P2.4
D GND	35	1	PFI 14/P2.6

Figura 2.9: Distribución de pines para el conector SCB-68

Tanto la tarjeta de adquisición de datos DAQ PCI-6221, cable NI SHC68-68.EPM y conector SCB-68 cumplen con las directivas dadas por la RoHS que restringe el uso de seis materiales (plomo, mercurio, cadmio y retardantes para llama PBB y PBDE), en la fabricación de equipos eléctricos y electrónicos.

Además, a pesar de ser elementos fabricados y diseñados para realizar aplicaciones en las que se utiliza código elaborado en el software LABVIEW puede ser compatible a código generado en Matlab.

2.2.1 Características de las entradas y salidas analógicas de la tarjeta de adquisición de datos National Instrument PCI-6221

Las características que posee la tarjeta de adquisición de datos National Instrument PCI-6221 para sus entradas y salidas analógicas son factores muy importantes a la hora de trabajar, ya que de ello depende el rango de funcionamiento y precisión del mismo.

Para las entradas analógicas se cuenta con 16 pines de montaje vertical ajustados por tornillos, que trabajan a una velocidad de 250 Ks/s, con una resolución de 16 bits en un rango de voltaje de -10 a 10 voltios, con una precisión de 3.1 mV y una memoria interna de 4095 muestras.

En el caso de las salidas analógicas se cuenta con 2 pines de montaje vertical ajustados por tornillos que trabajan a una velocidad de 833 Ks/s, con una resolución de 16 bits en un rango de voltaje de -10 a 10 voltios, con una precisión de 3.2 mV y una capacidad de corriente de 5mA, rangos suficientes para la implementación.

Es necesario reiterar, que el rango de voltaje que posee la tarjeta de adquisición de datos tanto para las entradas y salidas analógicas, es un factor que delimita el rango de trabajo que tuvo en el presente proyecto.

2.3 Sistema de control mediante la herramienta Real-Time Workshop de Matlab

El Real Time Workshop (RTW) y Real Time Workshop Target (RTWT) son herramientas del Matlab para la generación y desarrollo rápido de software. Estas herramientas permiten la implementación en tiempo real a partir de los bloques de Simulink .

La herramienta de Matlab REAL TIME WORKSHOP TARGET es considerada como una tarjeta de adquisición de datos en el software del Simulink por medio de bloques (figura 2.10) con diferentes características de funcionamiento: bloques de

salidas y entradas analógicas. Estas tarjetas de adquisición de datos permiten trabajar en tiempo real para comprobar modelos y diseños desarrollados.

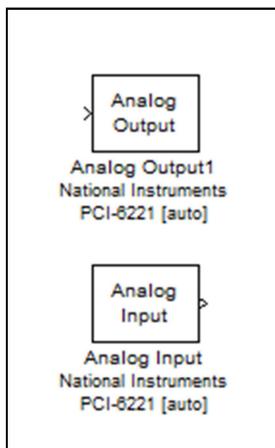


Figura2.10: Bloques para RTW

CAPÍTULO III

MODELAMIENTO MATEMÁTICO E IDENTIFICACIÓN DE LA UNIDAD DE VELOCIDAD DC SPEED CONTROL SYSTEM

3.1. Modelamiento Matemático

Para el modelamiento matemático se considera como elemento primario de control en la unidad de velocidad DC Speed Control System al motor de corriente continua. En la figura 3.1 se muestra el circuito eléctrico de la armadura y el diagrama de cuerpo libre del rotor.

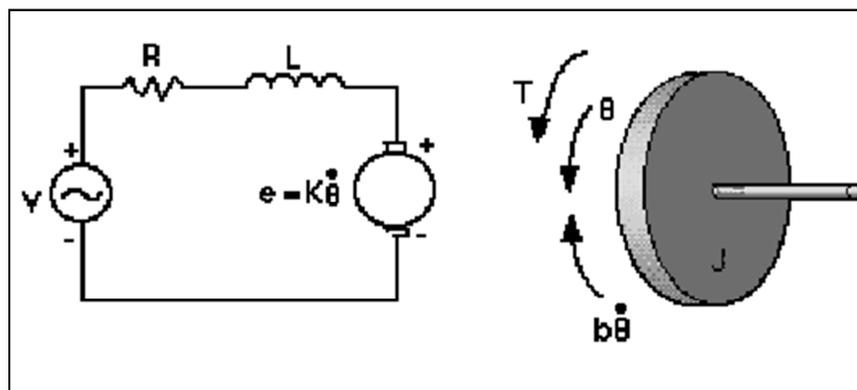


Figura 3.1: Circuito eléctrico de la armadura y diagrama del cuerpo libre del rotor.

Donde:

V = Voltaje de alimentación.

R = Resistencia eléctrica.

L = Inductancia eléctrica.

e = Fuerza contra electromotriz.

T = Torque del motor.

J = Momento de inercia del motor.

θ = Posición del eje del rotor.

b = Coeficiente de amortiguamiento del sistema mecánico.

k = Coeficiente de fuerza electromotriz.

Se establecen las correspondientes ecuaciones que rigen en el sistema:

Ecuación Eléctrica Leyes de Kirchoff:

$$V(t) - e(t) = L \frac{d.i(t)}{dt} + R. i(t) \quad \text{ec (3.1)}$$

Ecuación mecánica Leyes de Newton:

$$T(t) = J \frac{d^2\theta(t)}{dt^2} + b \frac{d\theta(t)}{dt} \quad \text{ec(3.2)}$$

El torque del motor T se relaciona directamente con la corriente de armadura i por un factor k_t , de la misma manera la fuerza contra electromotriz e se relaciona directamente con la velocidad de rotación $\frac{d\theta(t)}{dt}$ por un factor k_e , de donde se determinan las siguientes ecuaciones:

$$T(t) = k_t. i(t) \quad \text{ec(3.3)}$$

$$e(t) = k_e \frac{d\theta(t)}{dt} \quad \text{ec(3.4)}$$

Sustituyendo la ecuación 3.3 en 3.1 y la ecuación 3.2 en 3.4 obtenemos las siguientes ecuaciones:

$$V(t) - k_e \frac{d\theta(t)}{dt} = L \frac{d.i(t)}{dt} + R. i(t) \quad \text{ec(3.5)}$$

$$k_t. i(t) = J \frac{d^2\theta(t)}{dt^2} + b \frac{d\theta(t)}{dt} \quad \text{ec(3.6)}$$

Aplicando la transformada de Laplace con condiciones iniciales nulas, se obtienen las siguientes ecuaciones:

$$V(s) - k_e \cdot \theta(s) \cdot s = L \cdot I(s) \cdot s + R \cdot I(s) \quad \text{ec(3.7)}$$

$$k_t \cdot I(s) = J \cdot \theta(s) \cdot s^2 + b \cdot \theta(s) \cdot s \quad \text{ec(3.8)}$$

En el sistema internacional de medidas, la constante k_t (constante de armadura) y la constante k_e (constante del motor) son las mismas, por lo tanto se las puede sustituir por una K y además eliminando el valor de la corriente $I(s)$ se puede relacionar ambas ecuaciones con el fin de obtener una función de transferencia de la velocidad con respecto al voltaje de entrada la cual es expresada en la ecuación 3.9.

$$\frac{\theta(s)}{V(s)} = \frac{K}{J \cdot L \cdot s^2 + (b \cdot L + R \cdot J) \cdot s + b \cdot R + K^2} \quad \text{ec(3.9)}$$

3.2. Identificación del sistema

La identificación de sistemas es un proceso ampliamente utilizado en la industria, para la obtención de controladores altamente eficientes y estables. Se usa para Obtener modelos de sistemas dinámicos a partir de mediciones realizadas mediante experimentos. Existen numerosos procesos y métodos de identificación tanto paramétricos (Mecanísticos) como no paramétricos (Empíricos) que ofrecen información variada sobre el sistema en estudio.

Los métodos mecanísticos, se construye a partir de principios físicos, químicos, biológicos, etc las ecuaciones de balance de energía que permite identificar el sistema en estudio. Mientras que los métodos empíricos, no requieren de un conocimiento del sistema en estudio.

Para la identificación de un sistema se tienen que seguir los siguientes pasos: recolección de datos de entrada y salida, selección de un modelo apropiado, criterio de ajuste y validación del modelo.

La recolección de datos de entrada y salida, se realizó por medio de la tarjeta de adquisición de datos DAQ PCI-6221 utilizando la entrada analógica AI0 (borna

68), salida analógica AO0 (borna 22) y tierra GND (borna 56), dichas conexiones se las realiza como se observa en la figura 3.2.

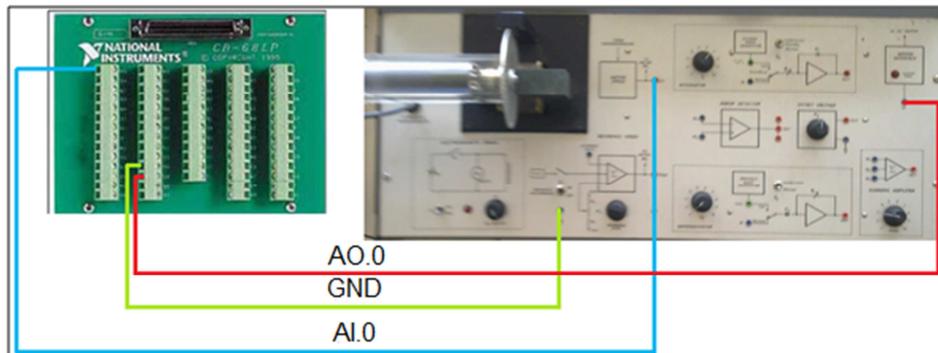


Figura 3.2: Diagrama de conexiones tarjeta-unidad de velocidad.

Después de realizar las conexiones pertinentes entre el modulo de velocidad y la tarjeta de adquisición de datos, se realiza un programa en Simulink que nos permite observar en la pantalla del computador todas las señales de interés que se van generando en el transcurso de los experimentos.

En la figura 3.3 y 3.4, se observa la configuración del tiempo de muestreo, canal de salida y rango de salida para la entrada y salida analógica de la tarjeta de adquisición de datos PCI-6221.

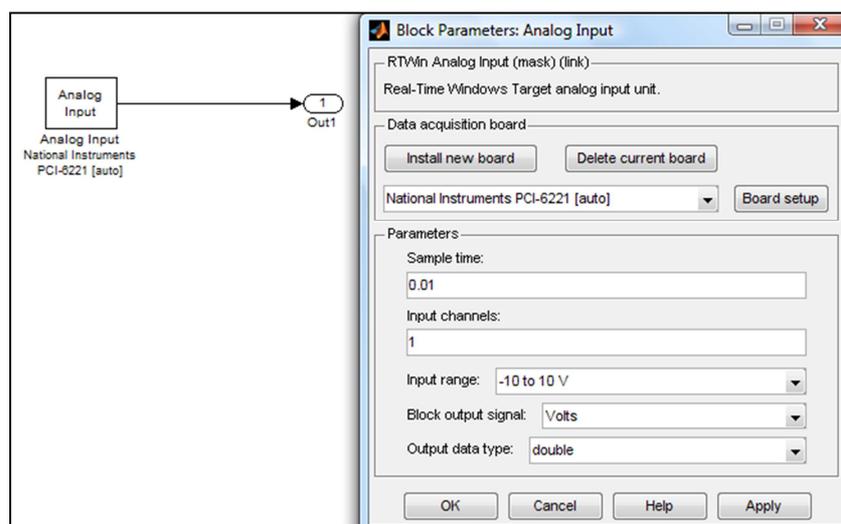


Figura 3.3: Configuración de entrada analógica

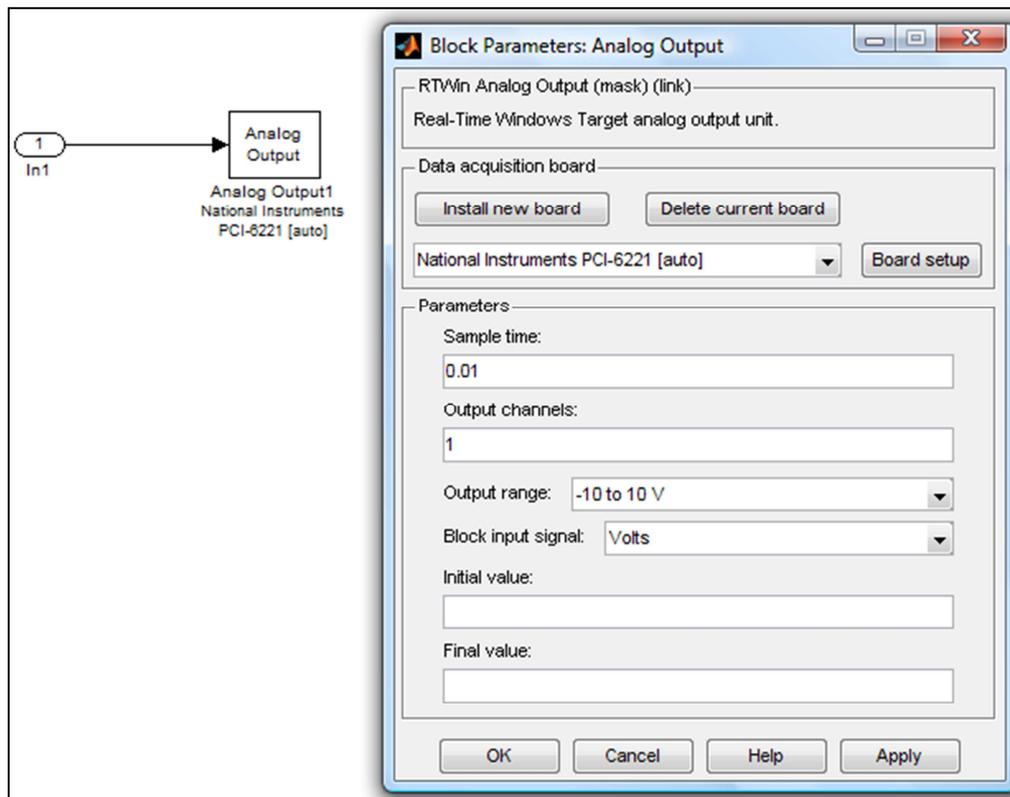


Figura 3.4: Configuración de salida analógica

Cabe recalcar que el tiempo de muestreo elegido para realizar todos los experimentos es de 10 ms y que el rango de entrada y salida de la tarjeta se encuentra entre -10 a 10 V.

Para la entrada analógica, es necesario transformar su señal (Voltaje) a una señal que represente la velocidad de salida del sensor (RPM). Esta relación es de 500 la cual es sugerida por el fabricante.

Esta relación, se lo multiplica a la salida de la entrada analógica como se observa en la figura 3.5.

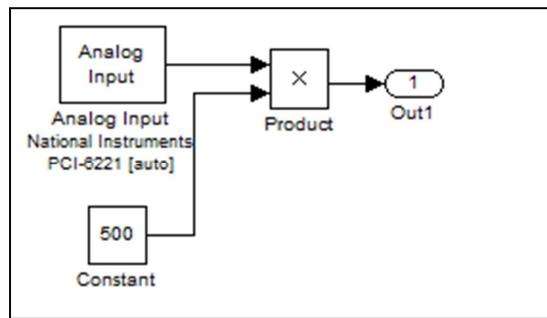


Figura3.5: Entrada analógica con factor de conversión.

Al realizar las primeras pruebas a un salto de escalón de 8V (figura 3.6) en la entrada de la interfaz de motor de la unidad de velocidad, se obtuvo una señal característica de este tipo de tacómetro.

Este tipo de señal contiene ruido como se observa en la figura3.7, por lo tanto es necesario colocar un filtro pasa bajos de primer orden con una frecuencia de corte de 2.5 rad/s para eliminar dicho ruido.

La configuración del filtro se la puede apreciar en la figura3.8. Con lo cual la salida del sistema tiene ya una respuesta como la que se puede observar en la figura 3.9.

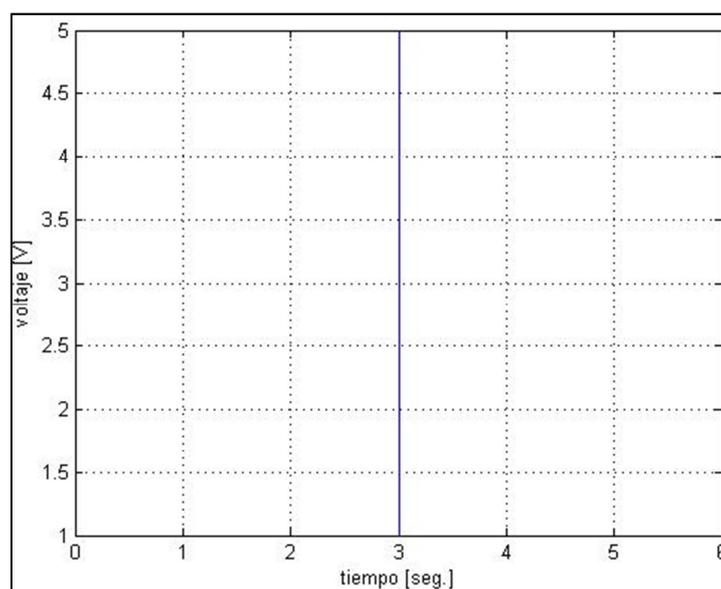


Figura3.6: Escalón de voltaje.

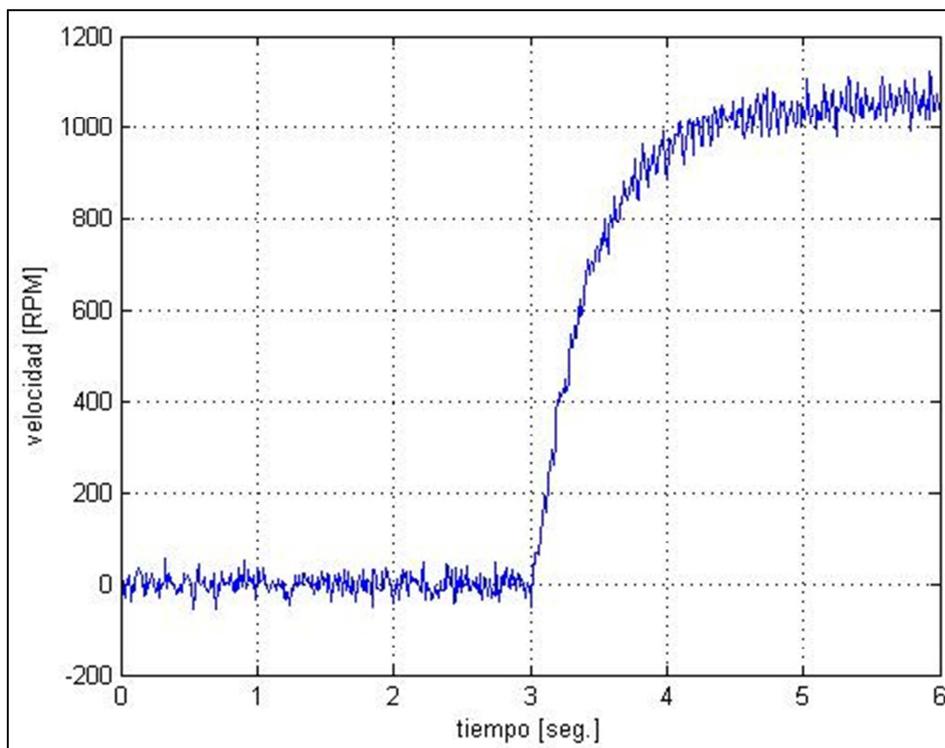


Figura3.7: Señal de la entrada analógica sin filtro.

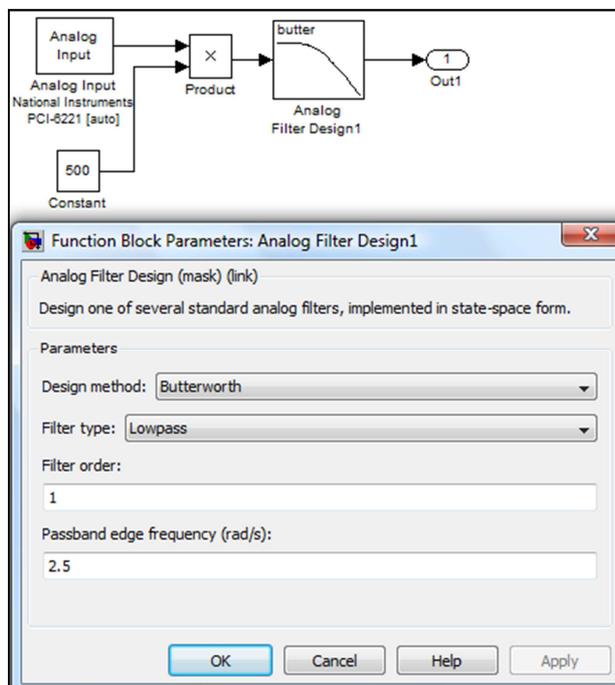


Figura3.8: Configuración del filtro pasa bajos.

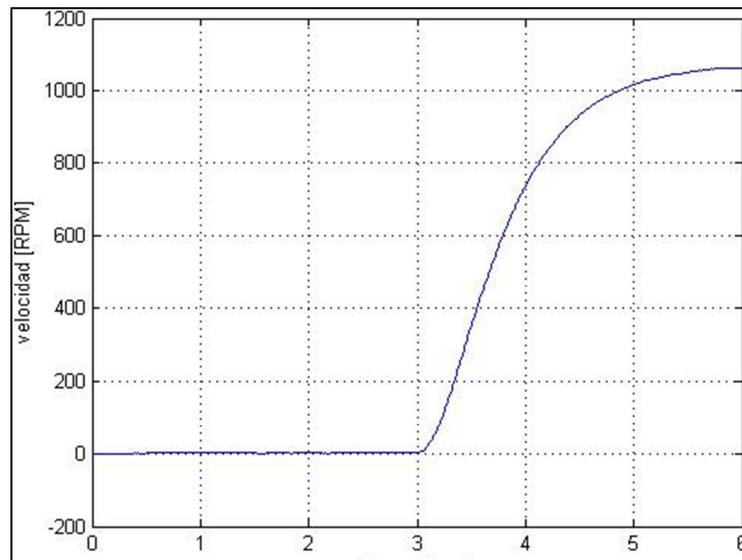


Figura3.9: Señal de la entrada analógica con el filtro.

El modelo para realizar la identificación del sistema, es una estructura del tipo caja negra (figura 3.9), en la cual se tiene una entrada (Voltaje) generando una salida (Velocidad en RPM).



Figura 3.10: Estructura tipo caja negra

Dentro de la caja negra, se tiene un modelo de primer orden con tiempo muerto del que se necesita conocer los valores de la ganancia del sistema (K), constante de tiempo (τ) y tiempo muerto (t_d). Todas estas constantes son obtenidas al observar la respuesta que tiene el sistema ante diferentes niveles de voltaje.

Una de las formas para obtener estas constantes es por medio del método de identificación de Smith. Este método estudia las respuestas temporales que tiene el sistema al 63.2% y 28.3% de su salida ante un cambio en su entrada (figura

3.10). Estas respuestas temporales determinando dos instantes de tiempo t_1 y t_2 ; con estos dos instantes y las ecuaciones 10 y 11 se obtienen las constantes τ y θ .

$$\tau = \frac{t_2 - t_1}{\ln\left(\frac{y(t_2) - y(\infty)}{y(t_1) - y(\infty)}\right)} \quad (\text{ec 10})$$

$$\theta = t_1 - \tau \ln\left(\frac{y(t_1) - y(\infty)}{\Delta y}\right) \quad (\text{ec 11})$$

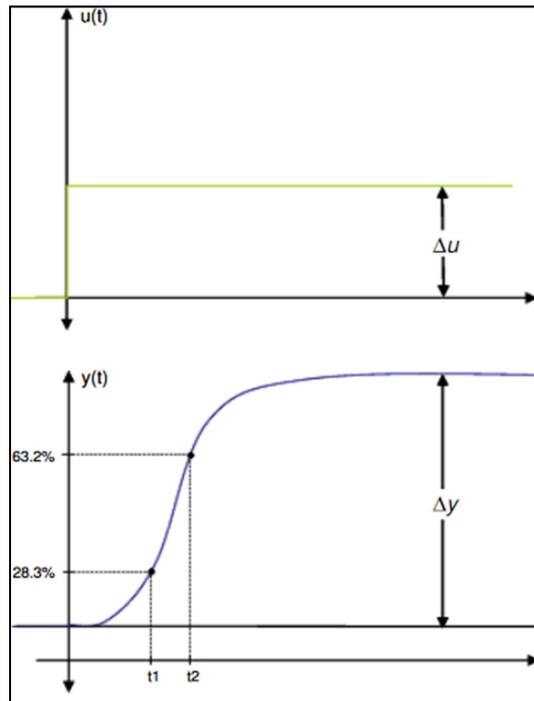


Figura 3.11: Entrada escalón y puntos de estudio tomados de la salida del sistema.

La ganancia de la planta K se obtiene de la relación de salida/ entrada de la planta.

Por medio de un programa en Matlab que se encuentra en el Anexo1, se determina los instantes de tiempo del sistema al 63.2% y 28.3% de su salida. Con lo cual se calcula la ganancia del sistema (K), constante de tiempo (τ) y tiempo muerto (θ).

El programa en Matlab utilizado para la identificación de las constantes es implementado dentro de una función de Matlab, que recibe los datos de la entrada analógica dando como resultados las constantes K , τ y θ (figura 3.11).

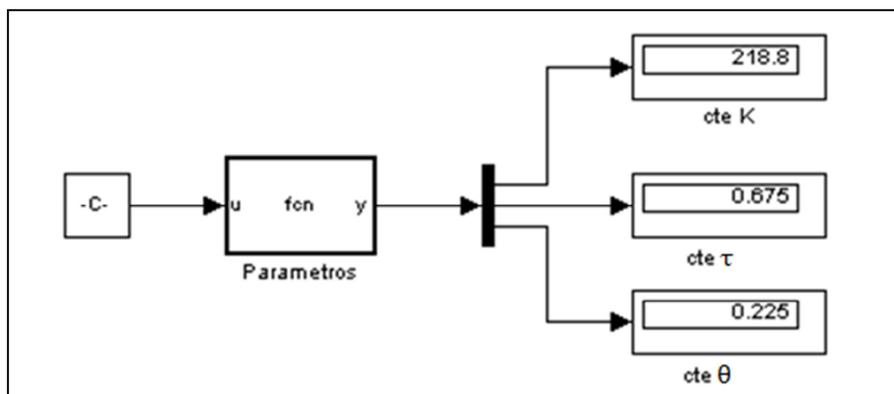


Figura 3.12: Función para encontrar los valores K, y .

Para obtener la respuesta más aproximada, se realizo experimentos a diferentes niveles de voltaje obteniendo los valores de la tabla 3.1.

Voltaje de entrada [V]	Velocidad de Salida [RPM]	Ganancia K	Tiempo muerto [s] θ	Constante de Tiempo [s]
1.4	0	0	0	0
2.5	225	131.5	0.285	0.615
3	500	166.28	0.26	0.6
4	825	206.23	0.22	0.63
5	1109.1	221.81	0.25	0.69
6	1431	238.5	0.24	0.69
7	1789.6	255.65	0.235	0.705
8	2075.8	259.5	0.26	0.675
9	2420.5	269	0.28	0.71

Tabla3.1: Valores de constantes y ganancias obtenidos a diferentes niveles de voltaje

Promediando los valores obtenidos de las constantes K, y se obtiene el sistema que se observa en la figura 3.12.

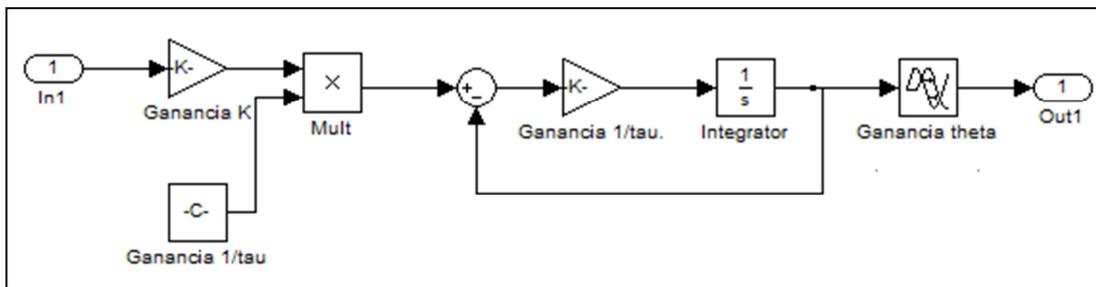


Figura 3.13: Diagrama del sistema de primer orden con tiempo muerto

Para realizar el ajuste del sistema de primer orden con tiempo muerto, se añaden los bloques de saturación y de tiempo muerto.

El bloque de saturación (figura 3.13), sirve para limitar la entrada del sistema en un rango de -9 a 9 voltios, se toma este rango por ser los valores que se usaron para encontrar las constantes K , τ y θ .

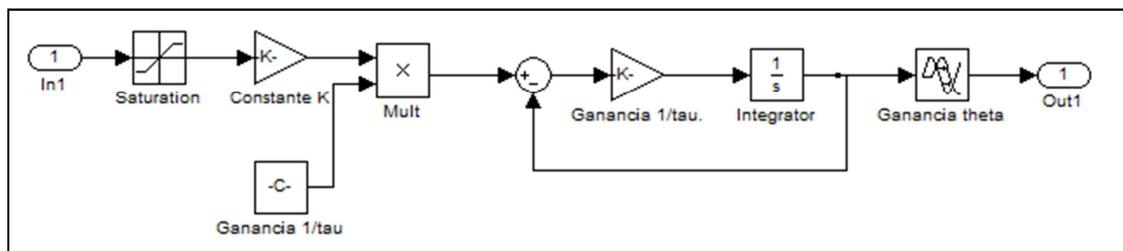


Figura 3.14: Diagrama del sistema de primer orden con tiempo muerto y bloque de saturación

Finalmente, el bloque de zona muerta (figura 3.14), envía como entrada el valor de cero cuando la entrada al sistema se encuentra en el valor de -1.4 a 1.4 voltios, este rango se lo determinó de forma experimental.

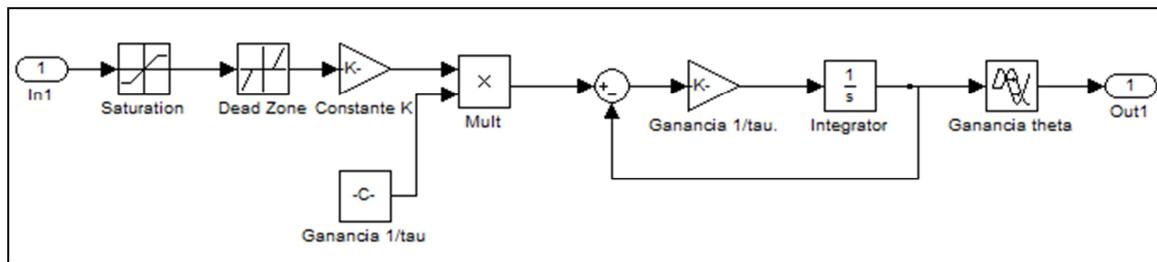


Figura 3.15: Diagrama del sistema de primer orden con tiempo muerto con bloque de saturación y tiempo muerto.

Como último paso en la identificación del sistema, se realizó la validación del modelo obtenido, sometiendo al modelo y a la planta de velocidad a una misma entrada de voltaje.

En la figura 3.15 se tiene la forma del diagrama en Simulink para realizar la validación.

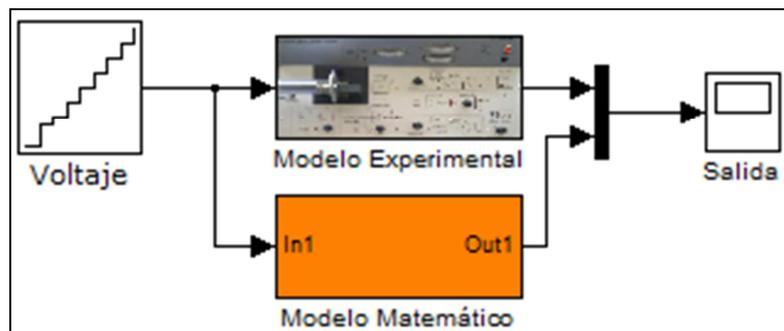


Figura 3.16: Diagrama en simulink para validar el modelo.

Al someter tanto módulo de velocidad como el modelo obtenido, se tiene la respuesta que se muestra en la figura 3.16, con lo cual queda comprobado que el modelo obtenido tiene una la similitud necesaria para realizar el diseño de los controladores.

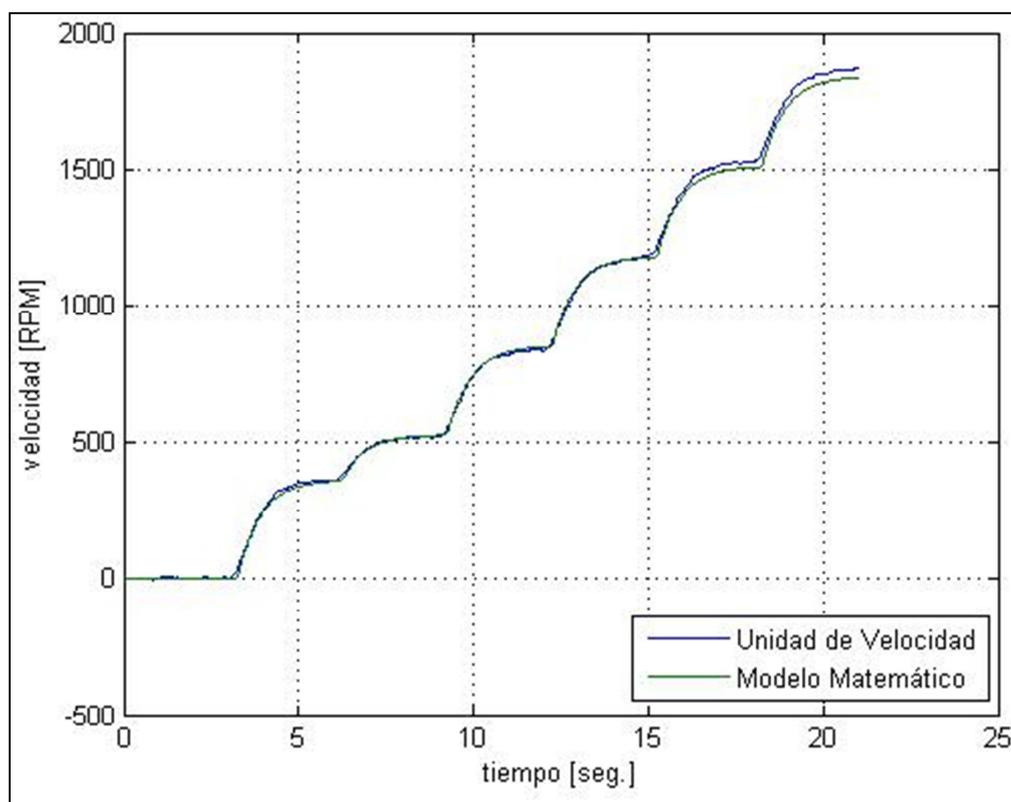


Figura 3.17: Respuesta del modelo matemático y el módulo de velocidad.

CAPÍTULO IV

APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES PI Y PID

4.1 Especificación del problema

El proceso de sintonización de controladores se puede llevar a cabo bien de manera rigurosa con modelos detallados de proceso o bien con la utilización de modelos aproximados del proceso. Existen métodos de sintonización de controladores sin modelo, entre los que destacan el modelo de sintonización en lazo cerrado de Ziegler Nichols que se puede implementar de forma totalmente experimental. Al ser un método experimental se corre el riesgo de obtener valores en las constantes K_p , K_i y K_d , que a la larga necesitan ser ajustados de forma manual para obtener la respuesta requerida.

Es por esta razón que se optó por un método de sintonización basado en la optimización de sus constantes, para lo cual es necesario tener un modelo definido del controlador PID sobre el cual se va a encontrar los valores óptimos de las constantes K_p , K_i y K_d .

4.2 Marco Teórico

El objetivo de la optimización global es encontrar la mejor solución de modelos de decisiones difíciles frente a las múltiples soluciones locales [1].

Las técnicas de optimización son empleadas para encontrar un juego de parámetros de diseño $\mathbf{x} = [x_1, x_2, \dots, x_n]$, que puede de algún modo ser definido como óptimo. En un caso simple esto podría ser la minimización o la maximización de alguna característica de sistema que es dependiente de \mathbf{x} .

Para modelos pequeños o medianos resulta adecuado el método símplex, el cual es un procedimiento iterativo que permite ir mejorando la solución a cada paso. El proceso concluye cuando no es posible seguir mejorando más dicha solución.

Matlab cuenta con la función `fminsearch`, que utilizando el método Simplex de optimización logra encontrar la solución a la función objetivo, en principio es adecuado para funciones que puedan presentar discontinuidades. La sintaxis que presenta la función `fminsearch` es la siguiente:

```
[x,fval,exitflag,output] = fminsearch(fun,x0,opciones)
```

Donde:

`fun` = función objetivo a optimizar. Puede definirse en un M-archivo.

`x0` = Semilla del algoritmo. Puede ser un escalar, un vector o una matriz.

`opciones` = opciones de optimización donde se puede graficas o ver la evolución de las variables de interés.

`x` = mínimo local de `fun`.

`fval` = devuelve el valor de la función objetivo `fun` en la solución `x`.

`exitflag` = describe la condición de salida de `fminsearch`.

`output` = Devuelve una estructura que contiene la optimización efectuada.

4.3 Solución del problema

Para encontrar las constantes del controlador PI y PID por medio de un método de optimización, se diseñó un lanzo de control (figura 4.1) en el que esta involucrado el bloque del controlador PID.

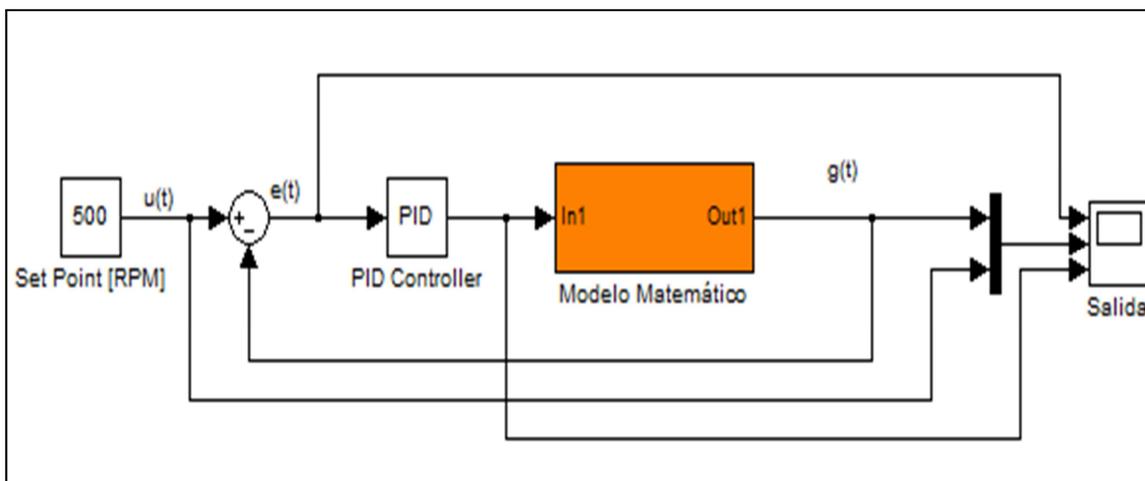


Figura 4.1: Diagrama del controlador PID

Dentro de las configuraciones para el controlador PID, se colocan las variables K_p y K_i (figura 4.2) que son los parámetros de diseño para realizar la optimización.

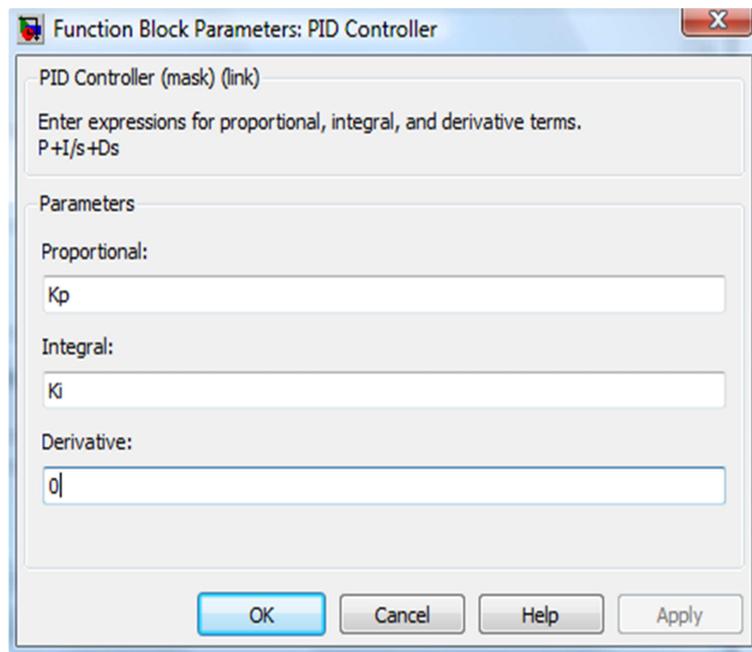


Figura 4.2: Configuración de parámetros para el bloque PID

La variable de interés para la función objetivo es el error que se produce entre el set point y la salida del sistema, por esta razón se lo envía al espacio de trabajo para ser utilizado (figura 4.3).

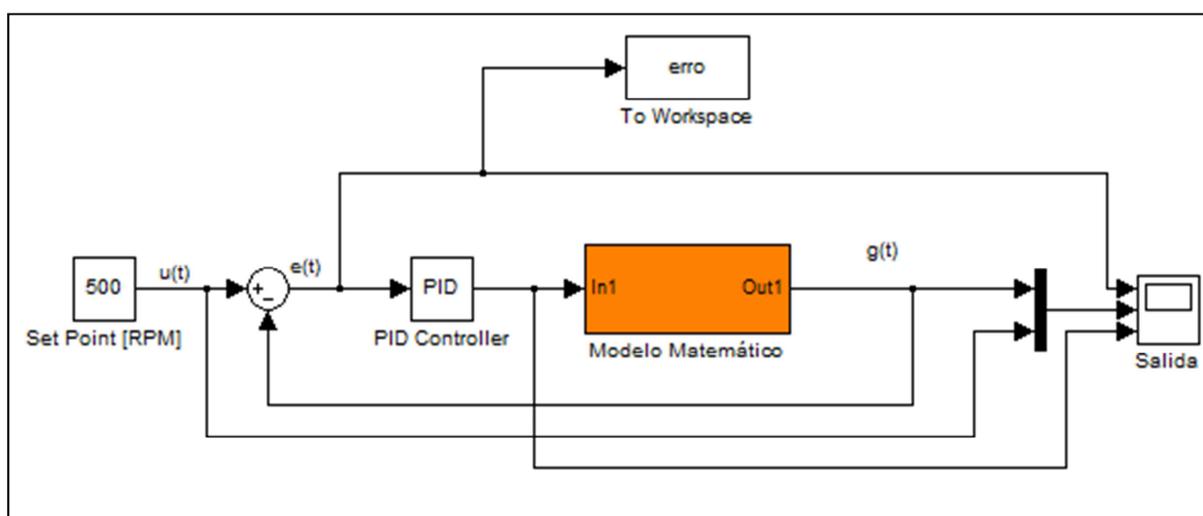


Figura 4.3: Diagrama en simulink DCMV_SIMPI donde se envía la señal de error al espacio de trabajo.

Se genera un código en Matlab ver Anexo 2 en el cual se utiliza la función `fminsearch` con su función objetivo, que contiene la variable de interés error para el cálculo de los óptimos valores de K_p y K_i .

Al ejecutar el código de Matlab se generan las variaciones que van tomando los puntos semilla como se observa en la figura 4.4., además se genera una gráfica en el cual podemos ver la evolución de la función objetivo con respecto a las iteraciones (figura4.5).

Iteration	Func-count	min f(x)	Procedure
0	1	3.11296e+006	
1	3	2.97755e+006	initial simplex
2	5	2.12592e+006	expand
3	7	1.60528e+006	expand
4	9	988738	expand
5	10	988738	reflect
6	12	988738	contract inside
7	14	988738	contract outside
8	16	924851	reflect
9	18	918853	contract inside
10	20	897948	reflect
11	22	897948	contract inside
12	24	874839	expand
13	26	874839	contract inside
14	27	874839	reflect
15	29	874839	contract inside
16	31	874839	contract inside
17	33	874839	contract inside
18	35	874829	contract inside
19	37	874533	contract inside
20	39	874459	contract inside
21	41	874459	contract outside
22	43	874449	contract inside
23	45	874444	contract inside
24	47	874442	contract inside
25	49	874439	contract inside
26	51	874439	contract outside
27	53	874439	contract inside
28	55	874439	contract outside
29	57	874439	contract inside
30	59	874439	contract inside
31	61	874439	contract outside
32	63	874439	contract inside
33	65	874439	contract inside
34	67	874439	reflect
35	69	874439	contract inside
36	71	874439	contract inside
37	73	874439	contract inside
38	74	874439	reflect
39	76	874439	contract inside
40	78	874439	contract inside
41	80	874439	contract inside
42	81	874439	reflect
43	83	874439	contract inside
44	85	874439	contract outside

Optimization terminated:
the current x satisfies the termination criteria using OPTIONS.TolX of 1.000000e-004
and F(X) satisfies the convergence criteria using OPTIONS.TolFun of 1.000000e-004

Kp=6.360091e-003
Ki=1.157097e-002

Figura 4.4: Valores óptimos para Kp y Ki

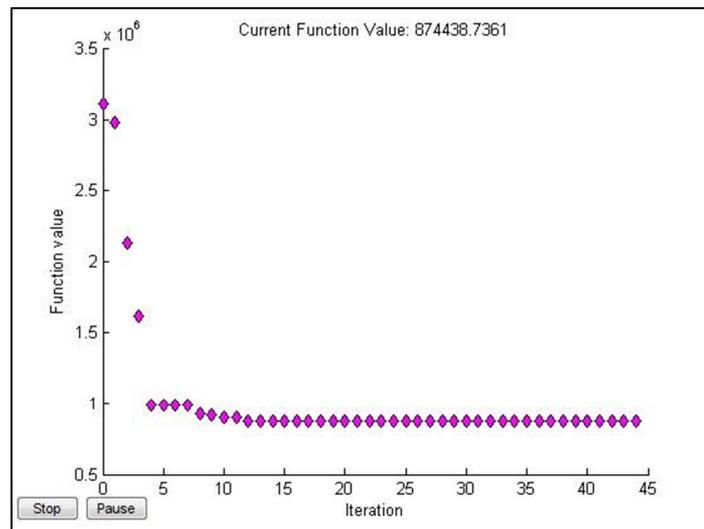


Figura 4.5: Gráfica de función objetivo vs iteraciones

Estos valores encontrados de K_p y K_i son colocados dentro del bloque del controlador PID.

Para el controlador PID de la misma manera se realiza un programa en Matlab (Anexo 3) el cual nos permite encontrar las constantes K_p , K_i y K_d como se muestra en la figura 4.6 y la gráfica que muestra la evolución que va teniendo la función objetivo con respecto al número de iteraciones (figura 4.7)

Iteration	Func-count	min f(x)	Procedure
64	116	405969	contract inside
65	118	405969	contract inside
66	120	405969	contract inside
67	122	405969	contract inside
68	124	405969	contract outside
69	126	405969	contract inside
70	128	405969	contract inside
71	129	405969	reflect
72	131	405969	contract outside
73	133	405969	contract inside
74	135	405969	contract inside
75	137	405969	contract inside
76	138	405969	reflect
77	140	405969	reflect

Optimization terminated:
the current x satisfies the termination criteria using OPTIONS.To1X of 1.000000e-004
and F(X) satisfies the convergence criteria using OPTIONS.To1Fun of 1.000000e-004

$K_p=9.399357e-003$
 $K_i=1.551686e-002$ |
 $K_d=8.938740e-004$

Figura 4.6: Valores óptimos para K_p , K_i y K_d

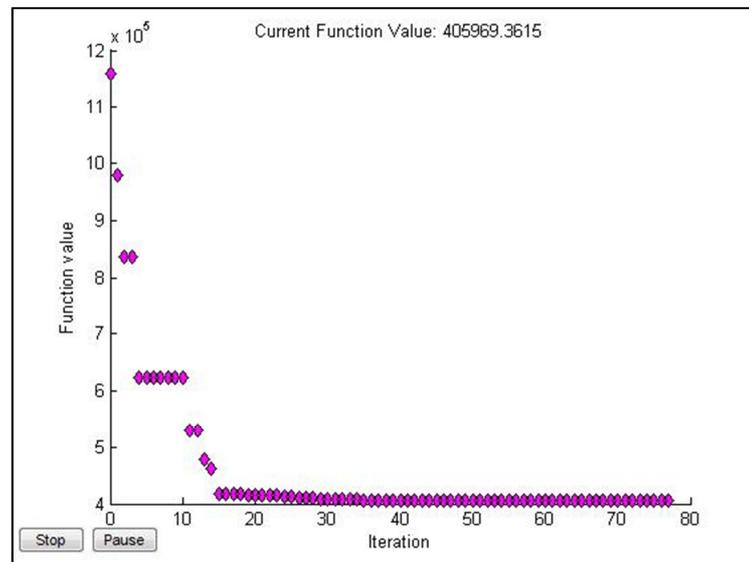


Figura 4.7: Grafica de función objetivo vs iteraciones

4.4 Simulación.

AL simular el sistema para el controlador PI, se obtiene la señal de salida del sistema (figura 4.8), señal de error (figura 4.9) y señal del controlador (figura 4.10) para una entrada de 500 rpm.

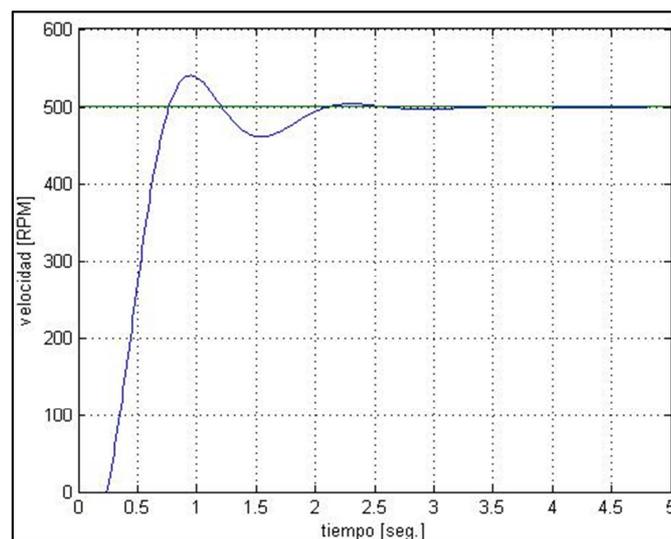


Figura 4.8: Señal de salida del sistema controlador.

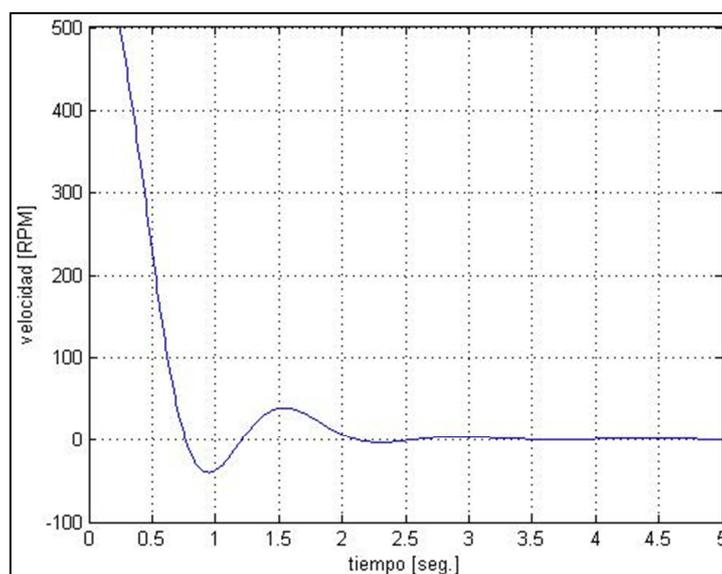


Figura 4.9: Señal de error del sistema.

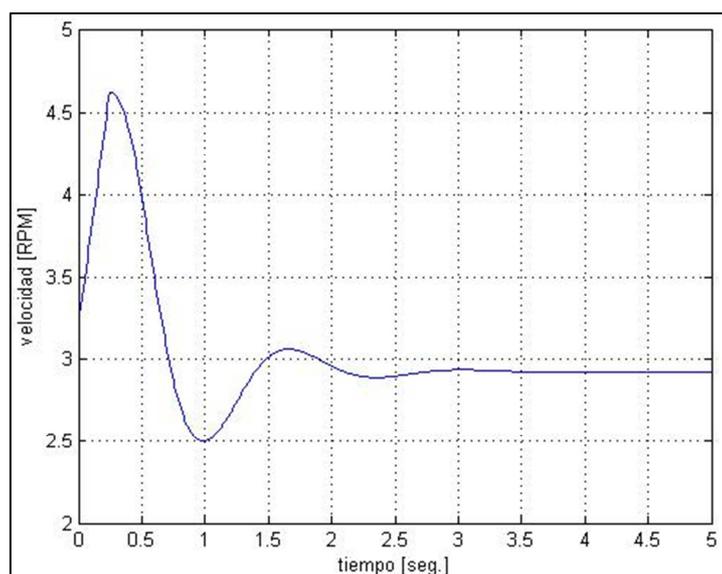


Figura 4.10: Señal de control del sistema.

En la figura 4.11 se puede observar la respuesta del sistema controlador por el controlador PI ante diferentes niveles de referencia.

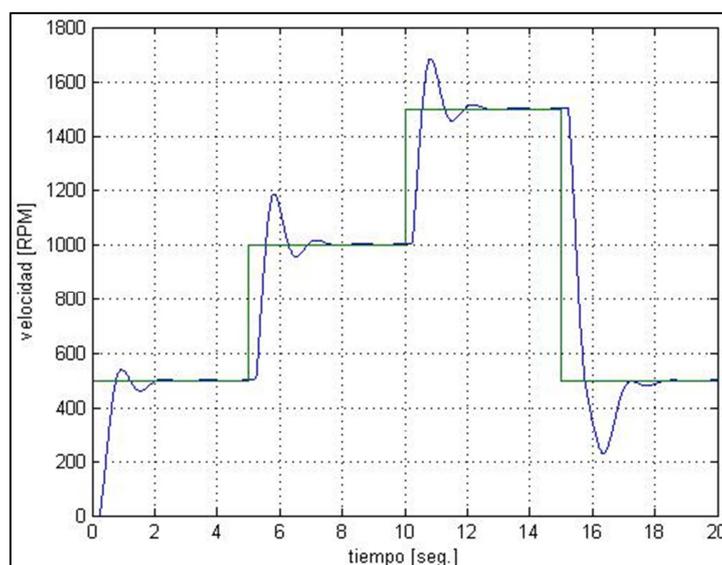


Figura 4.11: Respuesta del controlador PI ante diferentes niveles de referencia.

De la misma manera al simular el controlador PID se obtiene la señal de salida del sistema (figura 4.12), señal de error (figura 4.13) y señal del controlador (figura 4.14) para una entrada de 500 rpm.

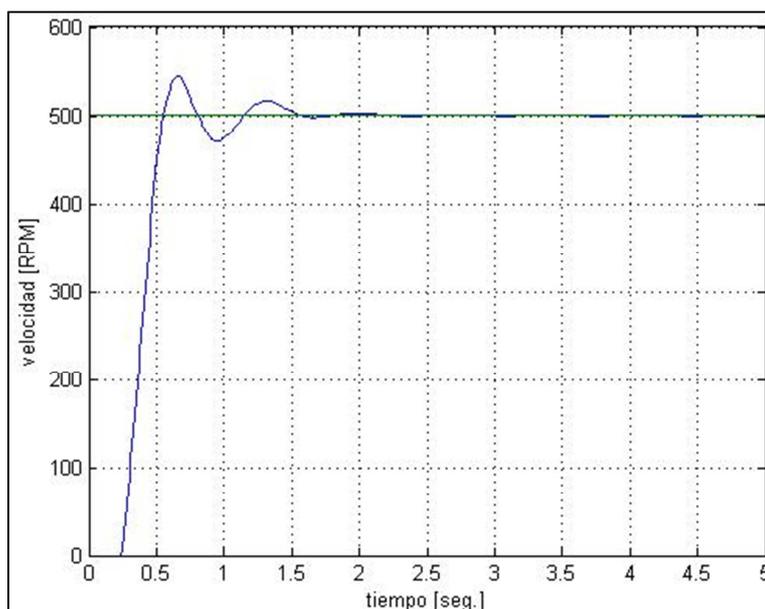


Figura 4.12: Señal de salida del sistema controlado.

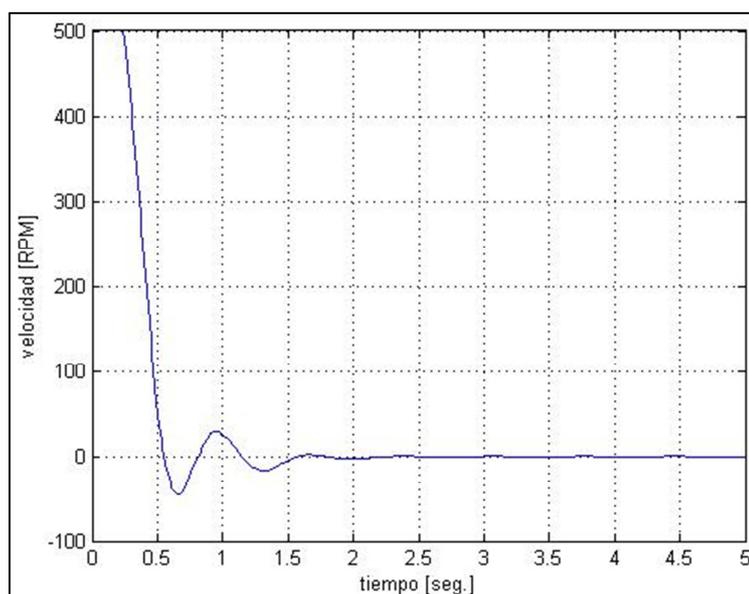


Figura 4.13: Señal de salida del error.

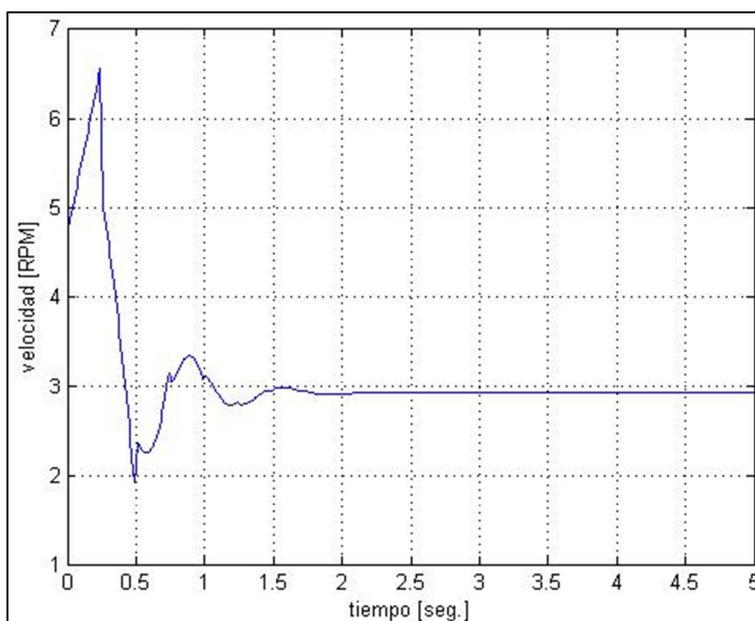


Figura 4.14: Señal de control del sistema

En la figura 4.15 se puede observar la respuesta del sistema controlador por el controlador PID ante diferentes niveles de referencia.

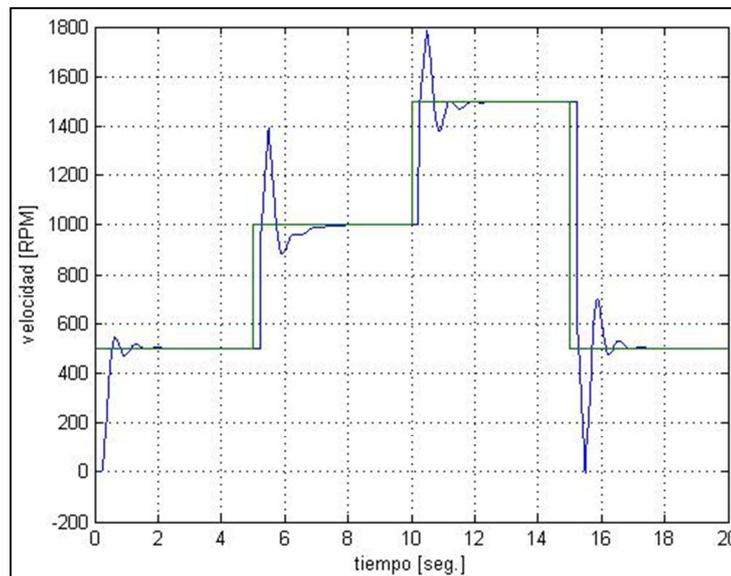


Figura 4.15: Respuesta del controlador PID ante diferentes niveles de referencia.

4.5 Implementación

Se implementa el lazo de control para el controlador PI con el modelo experimental, el cual contiene las entradas y salidas analógicas de la tarjeta de adquisición de datos respectivamente acondicionadas (figura 4.16).

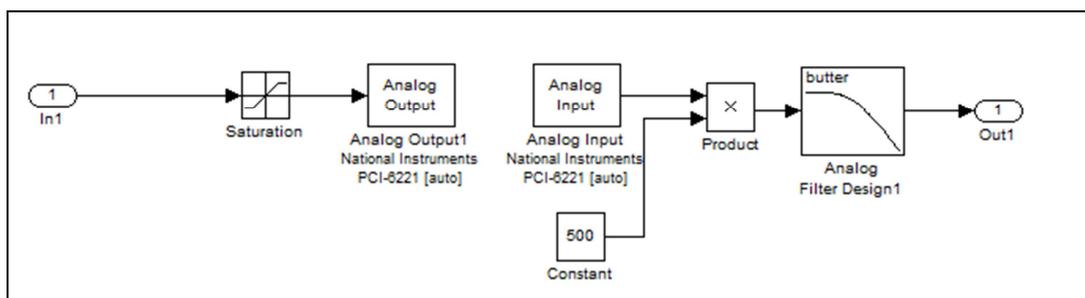


Figura 4.16: Contenido del modelo experimental

Colocamos los valores de las constantes K_p y K_i para el controlador PI en el lazo de control (figura 4.17).

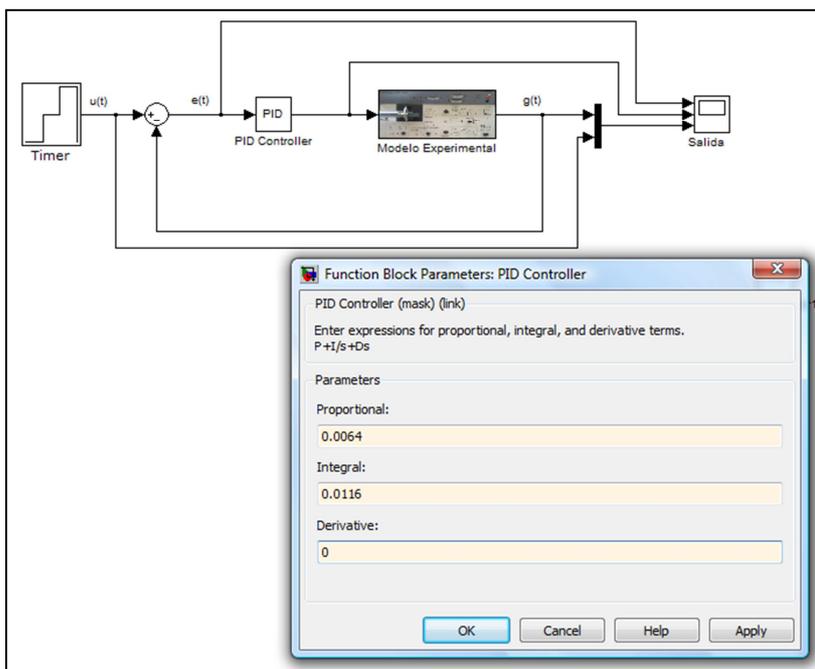


Figura 4.17: Configuración del controlador PI para el modelo experimental

Después de realizar la configuración del controlador, se obtuvieron las señales de salida del sistema controlado (figura 4.18), señal del controlador (figura 4.19) y señal de error (figura 4.20).

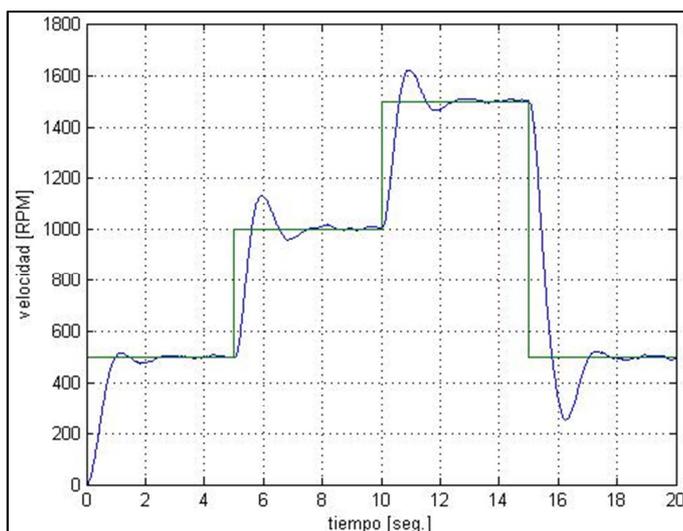


Figura 4.18: Señal de salida del sistema controlado

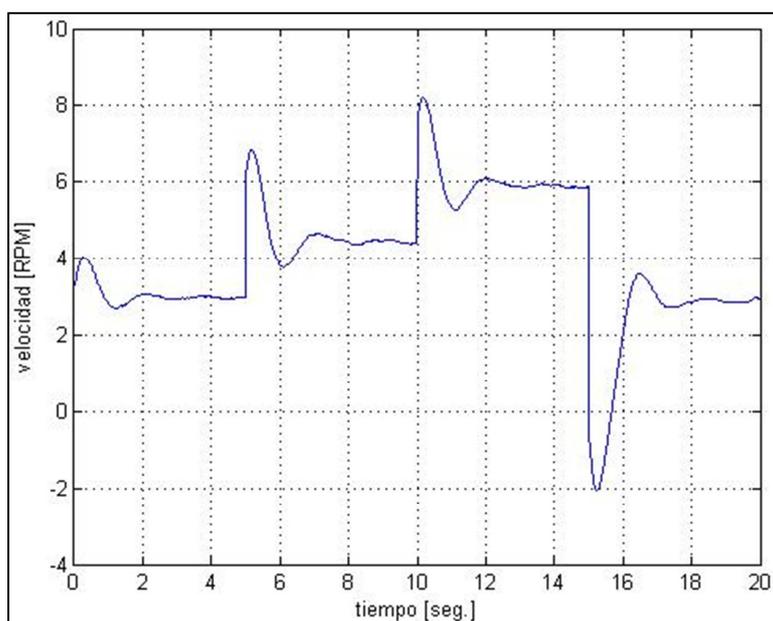


Figura 4.19: Señal de salida del controlador

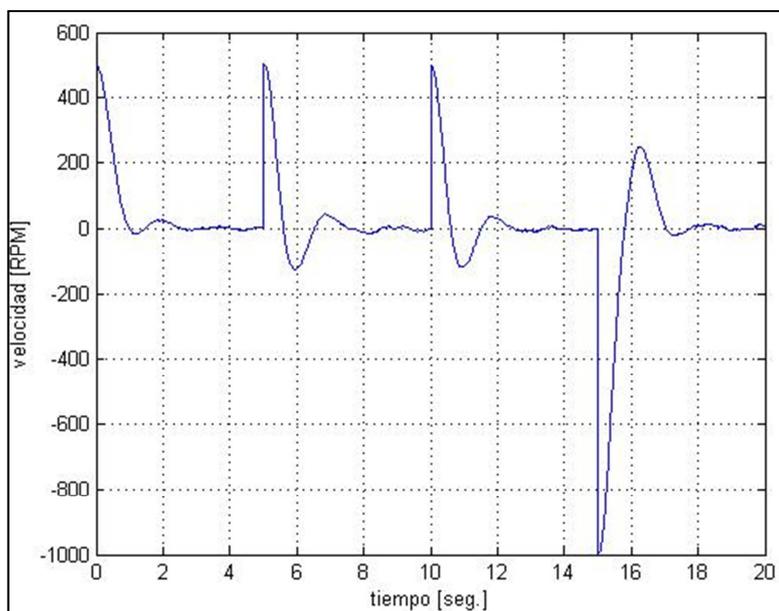


Figura 4.20: Señal del error.

Para el controlador PID, de la misma manera se colocan las constantes K_p , K_i y K_d obtenidas por la función de optimización dentro del bloque PID (figura 4.21).

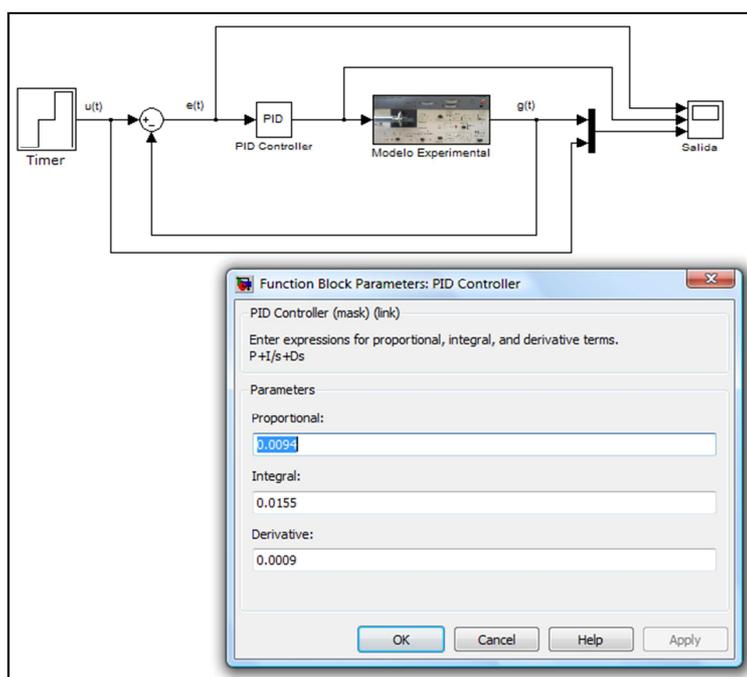


Figura 4.21: Configuración del controlador PID para el modelo experimental

Después de realizar la configuración del controlador, se obtuvieron las señales de salida del sistema controlado (figura 4.22), señal del controlador (figura 4.23) y señal de error (figura 4.24).

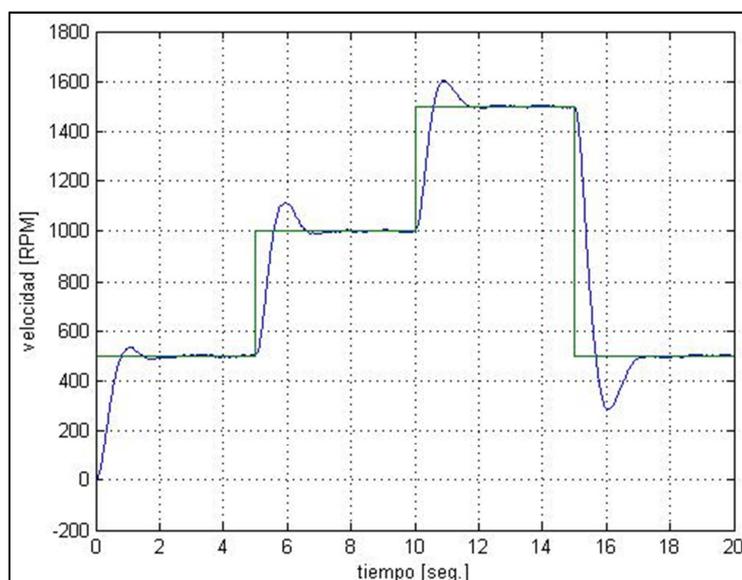


Figura 4.22: Señal de salida del sistema controlado

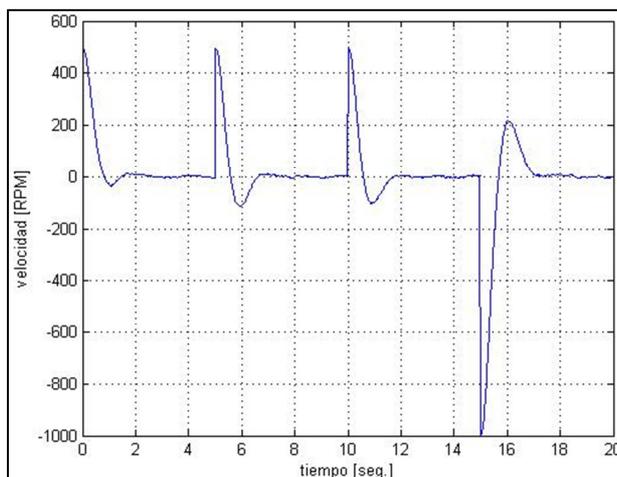


Figura 4.23: Señal de salida del controlador

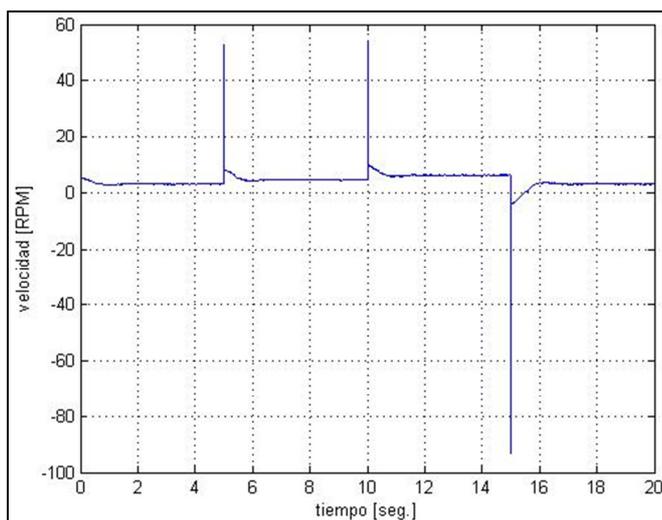


Figura 4.24: Señal del error.

CAPÍTULO V

APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES DIGITALES

5.1 Controlador Digital

Se requiere diseñar un controlador digital el cual permita realizar el control de velocidad de la Unidad Dc Speed Control System para diferentes niveles de referencia. Para el diseño realizado se baso en la teoría de ubicación de polos desde el enfoque polinomial.

La organización del capítulo se detalla a continuación: La sección 5.2 presenta la introducción y conceptos básicos. La sección 5.3 explica el análisis para ubicación de polos. La sección 5.4 presenta la ecuación Diofantina y un ejemplo simple para demostrar el enfoque de ecuaciones polinomiales.

5.2 Introducción:

Existen aspectos importantes que se requiere para el diseño del controlador digital por ejemplo. Se parte del análisis de la respuesta transitoria del sistema y de la ubicación de polos los cuales se detalla en la sección 5.3.

La característica básica de la respuesta transitoria de los sistemas de lazo cerrado, está ligada a la ubicación de polos en lazo cerrado. Por lo tanto es importante conocer como se desplaza los polos de lazo cerrado en el plano s al variar parámetros de ganancia. Los polos de lazo cerrado son las raíces de la ecuación característica.

Un enfoque diferente para el diseño de sistemas de control, se lo presenta en la sección 5.4 donde se encuentra el desarrollo de lo que es el enfoque de ecuaciones polinomiales para el diseño de sistemas de control.

Existen aspectos importantes que hacen al control digital más eficiente que el control analógico los cuales se presenta a continuación.

- Los sistemas de control digital presentan ventajas como una menor susceptibilidad al deterioro debido al transcurso del tiempo o a factores del entorno.
- Presenta componentes menos sensibles al ruido y vibraciones en las señales, pose una mejor sensibilidad frente a la variación de parámetros.
- Permite crear registros y almacenar información, presenta mayor inmunidad al ruido, facilidad en el cambio de estrategia de control, menor consumo de energía.

5.3 Análisis para la Ubicación de polos

En muchos casos las características de desempeño del sistema de control se especifican en términos de la respuesta transitoria para una entrada de escalón unitario, puesto que esta es fácil de generar y es suficientemente drástica.

La respuesta transitoria de un sistema de control puede representar con frecuencia oscilaciones amortiguados antes de alcanzar el estado estacionario donde es común identificar:

- Tiempo de subida t_r
- Sobreelongación, M_p
- Tiempo de establecimiento, t_s
- Tiempo de retardo t_d

Las especificaciones en el dominio del tiempo que se han proporcionado y que presenta en la figura 5.1, son muy importantes ya que casi todos los sistemas de control son sistemas en el dominio del tiempo, es decir deben representar respuestas de tiempo estables. Lo que significa que el sistema de control debe modificarse hasta que la respuesta transitoria sea satisfactoria.

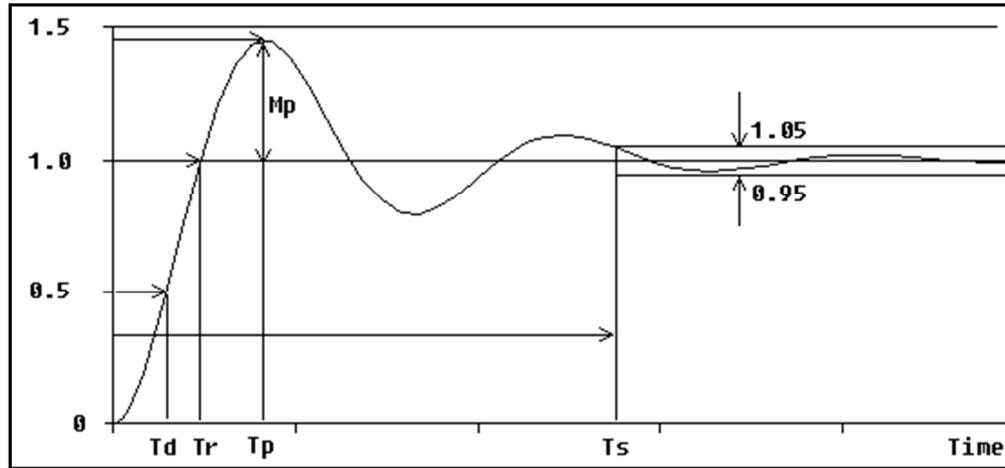


Figura 5.1 Parámetros De Respuesta Transitoria Frente A Un Escalón Unitario

A continuación se presenta el análisis de un sistema de primer y segundo orden para la ubicación de polos que se requiere conocer para el diseño.

Sistemas de primer orden

Un sistema de primer orden sin ceros puede ser descrito por la función de transferencia que se muestra en la figura 5.2. A continuación se representa la relación que existe entre la respuesta del sistema y la ubicación de los polos en el plano complejo de los sistemas de primer orden.

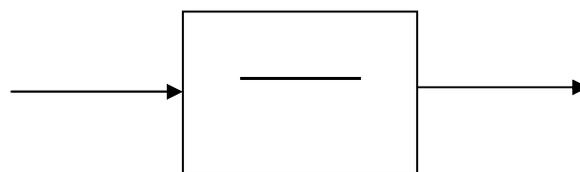


Figura 5.2 Sistema de Primer Orden

Si se aplica una entrada de escalón unitario, donde $U(s) = \frac{1}{s}$, la transformada de laplace de la respuesta al escalón es:

La respuesta al escalón se obtiene al tomar la transformada inversa de $Y(s)$

Dónde α son las ganancias del sistema y la constante de tiempo respectivamente. Presentando un polo en $-\alpha T$, representado en la figura 5.3.

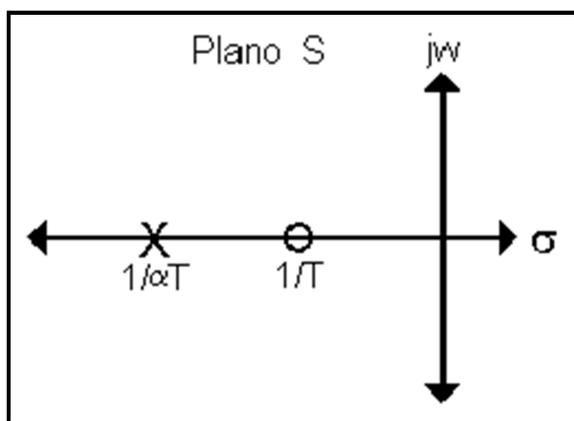


Figura 5.3. Representación del polo En el Plano (s)

La figura 5.4 representa la respuesta del sistema de la ecuación 5.1.

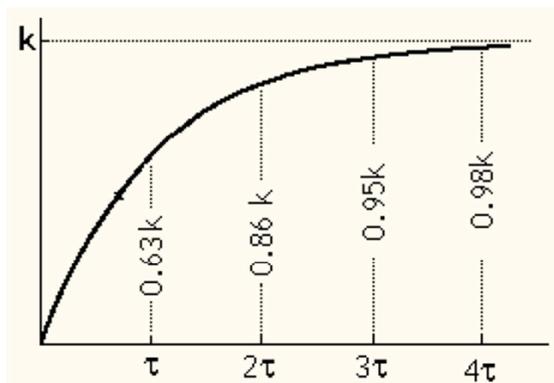


Figura 5.4. Respuesta del Sistema

Donde: τ se denomina **CONSTANTE DE TIEMPO** de la respuesta. De la ecuación 5.1, la constante de tiempo es el tiempo que toma la respuesta al escalón para alcanzar el 63% de su valor final, figura 5.4.

Como la derivada de:

$$e^{-\frac{1}{\tau}t} \text{ es } -\frac{1}{\tau} \text{ cuando } t = 0$$

τ es la rapidez de cambio inicial de la exponencial en $t = 0$

En consecuencia, la constante de tiempo se puede considerar una especificación de la respuesta transitoria para un sistema de primer orden, puesto que se relaciona con la velocidad a la que el sistema responde a una entrada escalón.

También se puede evaluar la constante de tiempo a partir de la gráfica del polo como se presenta en la figura 5.3, como el polo de la función de transferencia está en $-\frac{1}{\tau}$, podemos decir que el recíproco de la ubicación del polo es la constante de tiempo, y cuanto más alejado se encuentre el polo del eje imaginario, más rápida es la respuesta transitoria.

Sistema De Segundo Orden

El análisis de un sistema de segundo orden tiene una amplia variedad de respuestas que dependen de los parámetros:

\mathcal{E} : *factor de amortiguamiento*

ω_n : *frecuencia natural no amortiguada*

El producto de $(\mathcal{E} * \omega_n)$ se le conoce como atenuación del sistema (σ).

La forma de la ecuación general de un sistema de segundo orden es:

La función de transferencia tiene dos polos y no tiene ceros. La siguiente ecuación permite encontrar los polos del sistema.

En la siguiente figura 5.5, se presenta la ubicación de polos y sus correspondientes respuestas al escalón unitario, para el ejemplo

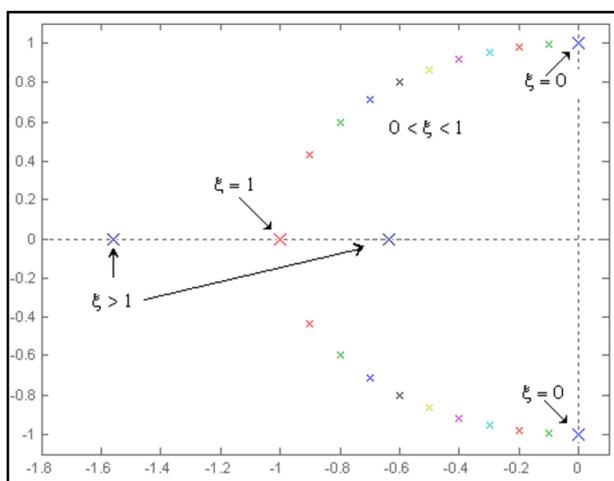


Figura 5.5: Ubicación de polos variando

El comportamiento dinámico del sistema de segundo orden puede ser descrito en términos del factor de amortiguamiento, como se observa en la figura 5.6.

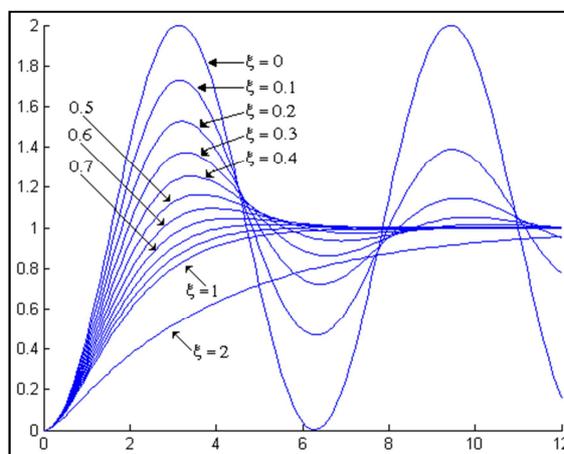


Figura 5.6: Sistema dinámico del sistema

En la siguiente tabla se especifica lo que representa el valor del factor de amortiguamiento:

Factor De Amortiguamiento	Representación
$\varepsilon = 0$	Los polos son imaginarios conjugados, Sistema denominado críticamente estable.
$0 < \varepsilon < 1$	Los polos son complejos y conjugados, Sistema denominado Sub-amortiguado.
$\varepsilon = 1$	Los polos son reales y repetidos, El sistema se denomina críticamente amortiguado.
$\varepsilon > 1$	Los polos son reales y distintos, El sistema se denomina sobreamortiguado.

Tabla 5.1 Variación de ε

Las características deseadas del comportamiento de un sistema de segundo orden, pueden especificarse en función de la respuesta transitoria ante una entrada escalón.

Tiempo de establecimiento t_s : Es el tiempo necesario para que la respuesta alcance y permanezca dentro de un porcentaje del error alrededor del valor final.

$$t_s = \frac{4}{\varepsilon\omega_n} \quad ec \ 5.3$$

Máximo sobreimpulso M_p : Es la magnitud del primer sobrepaso el cual ocurre en el tiempo pico, medido desde la señal de referencia. Si la señal de referencia no es un escalón unitario, el máximo sobreimpulso se expresa en porcentaje.

$$M_p = e^{-\frac{\varepsilon\pi}{\sqrt{1-\varepsilon^2}}} \times 100\% \quad ec \ 5.4$$

Estos parámetros se los detalla en la figura 5.1 de la sección 5.3.

5.4 Ecuación Diofantina:

Esta sección presenta una explicación al enfoque de ecuaciones polinomiales para el diseño de sistemas de control. En este enfoque se resuelve la ecuación Diofantina para determinar polinomios en (z) .

Considere el sistema definido por la función de transferencia pulso.

$$\frac{Y(z)}{U(z)} = \frac{B(z)}{A(z)} \quad ec(5.5)$$

Donde:

$$A(z) = z^n + a_1z^{n-1} + \dots + a_{n-1}z + a_n$$

$$B(z) = b_0z^n + b_1z^{n-1} + \dots + b_{n-1}z + b_n$$

Supongamos que esta función de transferencia pulso es de estado completamente controlable y completamente observable. Esto quiere decir que no existe una cancelación entre polos y ceros en la función de transferencia pulso, o $A(z)$ y $B(z)$ no tienen factores en común.

A continuación se define un polinomio estable $D(z)$ de grado $(2n - 1)$ como sigue:

$$D(z) = d_0z^{2n-1} + d_1z^{2n-2} + \dots + d_{2n-2}z + d_{2n-1}$$

Entonces existen polinomios únicos de grado $(n - 1)$, $\alpha(z)$ y $\beta(z)$ tales que

$$\alpha(z)A(z) + \beta(z)B(z) = D(z) \quad ec(5.6)$$

Donde:

$$\alpha(z) = \alpha_0z^{n-1} + \alpha_1z^{n-2} + \dots + \alpha_{n-2}z + \alpha_{n-1}$$

$$\beta(z) = \beta_0z^{n-1} + \beta_1z^{n-2} + \dots + \beta_{n-2}z + \beta_{n-1}$$

La ecuación 5.2 se denomina ecuación Diofantina, en honor a Diofanto de Alejandría. La ecuación Diofantina se puede resolver para $\alpha(z)$ y $\beta(z)$ mediante

el uso de la matriz de Sylvestre \mathbf{E} de $2n \times 2n$, la cual se define en términos de los coeficientes de los polinomios $A(z)$ y $B(z)$.

$$E = \begin{bmatrix} a_n & 0 & \dots & 0 & b_n & 0 & \dots & 0 \\ a_{n-1} & a_n & \dots & 0 & b_{n-1} & b_n & \dots & 0 \\ \vdots & a_{n-1} & \dots & 0 & \vdots & b_{n-1} & \dots & 0 \\ a_1 & \vdots & \dots & \vdots & b_1 & \vdots & \dots & \vdots \\ 1 & a_1 & \dots & a_n & b_0 & b_1 & \dots & b_n \\ 0 & 1 & \dots & a_{n-1} & 0 & b_0 & \dots & b_{n-1} \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & a_1 & 0 & 0 & \dots & b_1 \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & b_0 \end{bmatrix} \quad ec(5.7)$$

Ahora se define los vectores \mathbf{D} y \mathbf{M} tales que.

$$D = \begin{bmatrix} d_{2n-1} \\ d_{2n-2} \\ \vdots \\ d_1 \\ d_0 \end{bmatrix} \quad M = \begin{bmatrix} \alpha_{n-1} \\ \alpha_{n-2} \\ \vdots \\ \alpha_0 \\ \beta_{n-1} \\ \beta_{n-2} \\ \vdots \\ \beta_0 \end{bmatrix}$$

Entonces los coeficientes $\alpha_0, \dots, \alpha_{n-1}$ y $\beta_0, \beta_1, \dots, \beta_{n-1}$ se puede determinar a partir de:

$$\mathbf{M} = \mathbf{E}^{-1} \mathbf{D} \quad ec(5.8)$$

La ecuación 5.8 da la solución a la ecuación Diofantina, a continuación se presenta un ejemplo para su comprensión.

Ejemplo:

Considere los polinomios de $A(z)$ (De grado 2) y $B(z)$ (De grado 1) que formarn la ecuación 5.5, sin factores en común, detallada al inicio de la sección 5.4.

$$A(z) = z^2 + z + 0.5$$

$$B(z) = z + 2$$

$$D(z) = z^3$$

El problema es encontrar dos polinomios únicos $\alpha(z)$ y $\beta(z)$ tales que:

$$\alpha(z)A(z) + \beta(z)B(z) = D(z)$$

Donde:

$$\alpha(z) = \alpha_0 z + \alpha_1$$

$$\beta(z) = \beta_0 z + \beta_1$$

O bien

$$(\alpha_0 z + \alpha_1)(z^2 + z + 0.5) + (\beta_0 z + \beta_1)(z + 2) = z^3$$

La Ecuación 5.6 es una ecuación Diofantina. Para resolver esta ecuación para $\alpha(z)$ y $\beta(z)$, primero se obtiene los coeficientes de la función de transferencia $G(z)$.

$$a_1 = 1, \quad a_2 = 0.5$$

$$b_0 = 0, \quad b_1 = 1, \quad b_2 = 2$$

Y entonces la matriz de Sylvestre \mathbf{E} se escribe como:

$$E = \begin{bmatrix} 0.5 & 0 & 2 & 0 \\ 1 & 0.5 & 1 & 2 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Para obtener la inversa de la matriz \mathbf{E} se utiliza Matlab con la función $inv(E)$

Ya que $D(z) = z^3$, se tiene:

$$d_0 = 1, \quad d_1 = 0, \quad d_2 = 0, \quad d_3 = 0$$

Se forma la matriz D :

$$D = \begin{bmatrix} d_3 \\ d_2 \\ d_1 \\ d_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Al definir la matriz **M** como:

Se obtiene la solución de la ecuación Diofantina:

A partir de esto se obtiene:

Diseño del controlador:

Mediante la identificación realizada en el capítulo 3, para la unidad de velocidad, se obtuvo una función de transferencia como se muestra en la figura 5.7 representada en un bloque de Simulink.

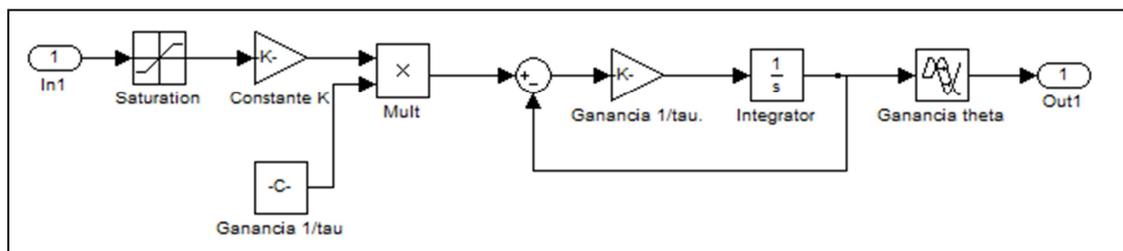


Figura 5.7. Modelo Matemático de la unidad DC SPEED CONTROL SYSTEM

De este modelo matemático utilizamos las constantes K y C para obtener una función de transferencia de primer orden, que con un tiempo de muestreo de 10 ms se obtiene la función en tiempo discreto como se muestra en la ecuación 5.9.

Al aplicar un escalón a la función $G(s)$ se determina los siguientes parámetros de respuesta transitoria presentados en la sección 2.3.

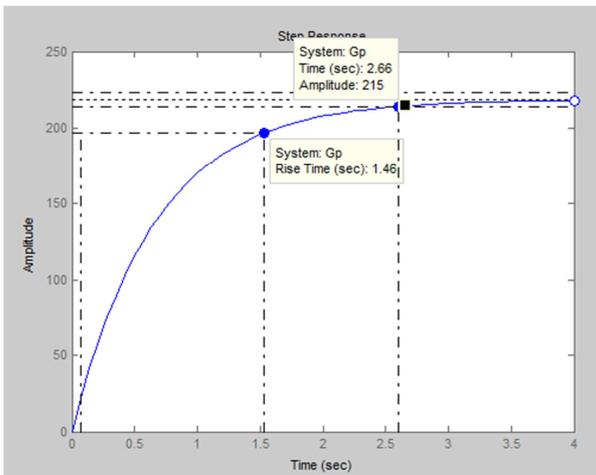


Figura 5.8 respuesta de $G(s)$ frente a un escalón

Las especificaciones para el diseño son:

Desarrollo

=====

=====

=====

Considere el diagrama de bloque de la figura 5.9, la función de transferencia realimentada — funciona como un regulador. Se determina mediante el enfoque de ecuaciones polinomiales.

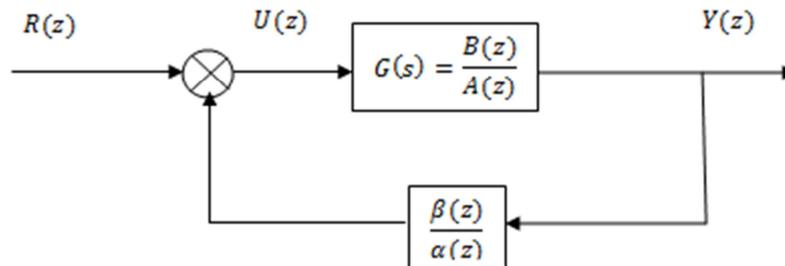


Figura 5.9 Diagrama de Bloques del Sistema

A continuación se determina la función de transferencia pulso en lazo cerrado para el sistema el cual está dado por.

Donde:

Para

Observe que $D(z)$ es un polinomio de grado $(2n-1)$, ya que

Entonces:

Se tiene que:

$$a_1 = -0.98, \quad a_2 = 1$$

$$b_0 = 0, \quad b_1 = 3.267$$

Se forma la matriz **E**

$$E = \begin{bmatrix} 0 & 1 \\ 0.3.267 & -0.985 \end{bmatrix} \text{ Matriz de parámetros de la función de transferencia}$$

$$D = \begin{bmatrix} d_2 \\ d_1 + 0.9857 \end{bmatrix} \text{ Matriz de valores de de polos y parámetros de la función de transferencia}$$

$$M = \begin{bmatrix} \beta_1 \\ \alpha_1 \end{bmatrix} \text{ Matriz de parámetros del controlador}$$

Resolviendo:

$$M = E^{-1} D$$

$$M = (\text{inv}(E)) * D$$

Para realizar la solución del sistema de ecuaciones, se realizó un programa en Matlab (Anexo 4), que permite cambiar la ubicación de los polos.

Para mejorar la respuesta del controlador ya que tiene un error en estado estacionario se colocó un bloque integrador discreto sintonizado su constante K_i en un valor de 0.35.

5.5 Simulación

Se coloca el bloque integrador y el controlador discreto como se muestra en la figura 5.10.

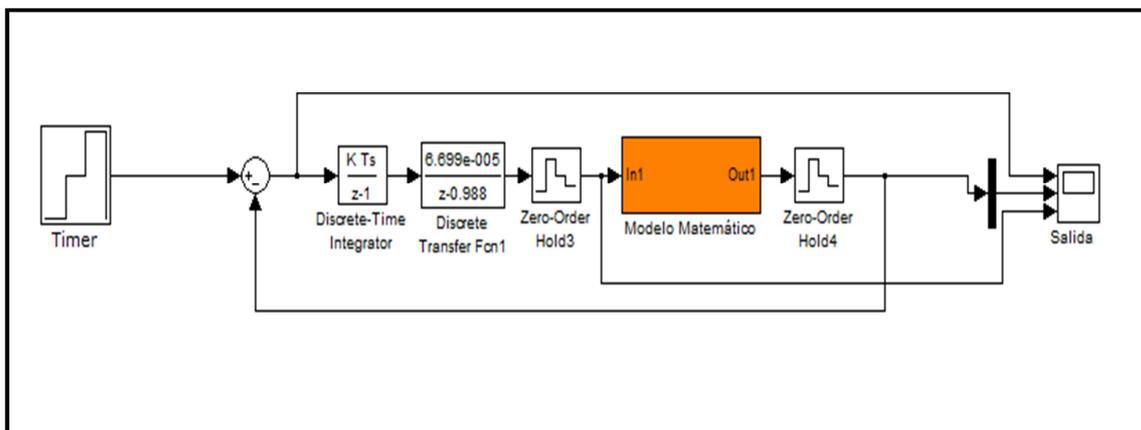


Figura 5.10: Controlador digital para el modelo matemático

Al realizar la simulación del sistema controlado con un período de muestro de 10 ms, se obtuvieron las señales de salida del sistema controlada (figura 5.11), señal del controlador (figura 5.12) y señal de error (figura 5.13) para una señal de referencia de 1500 rpm.

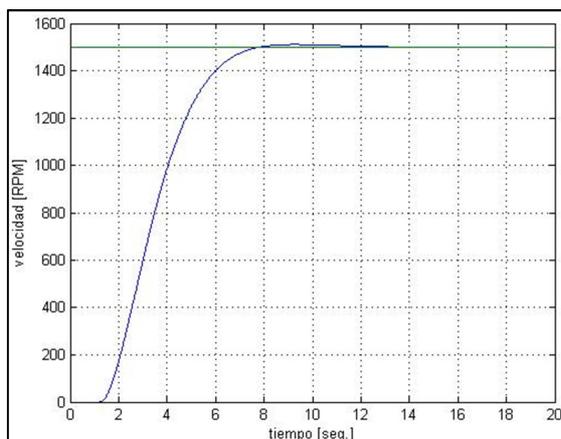


Figura 5.11: Señal de salida del sistema controlado con periodo de muestro de 10 ms

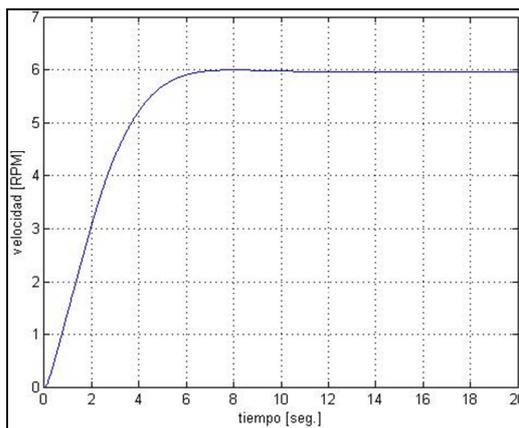


Figura 5.12: Señal de salida del controlador con periodo de muestro de 10 ms.

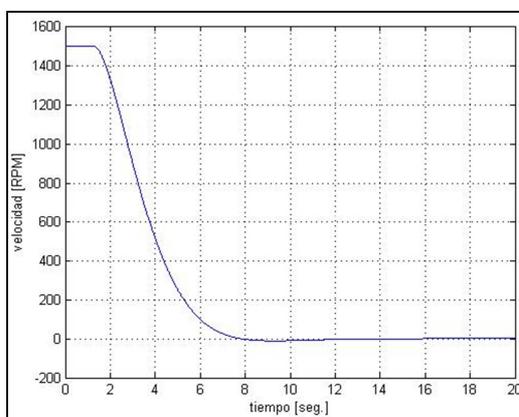


Figura 5.13: Señal del error con periodo de muestro de 10 ms.

En la figura 5.14 se puede observar la respuesta del sistema controlador por el controlador digital ante diferentes niveles de referencia un periodo de muestro de 10 ms.

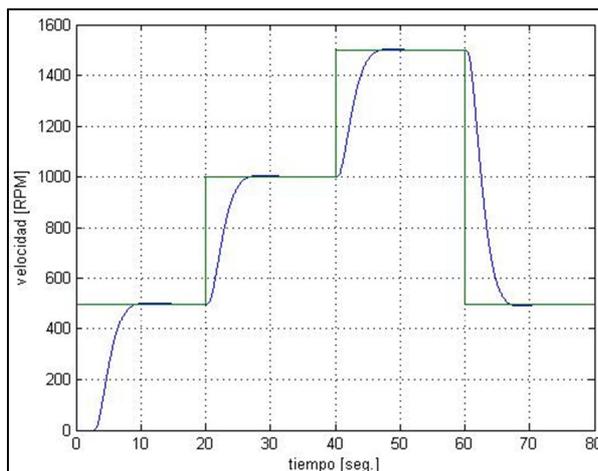


Figura 5.14: Señal de salida del sistema controlado ante diferentes señales de referencia y a un periodo de muestro de 10 ms.

Para conocer el efecto que produce el período de muestro, se realiza la simulación a 2 ms (figura 5.15), 50 ms (figura 5.16) y 100 ms (figura 5.17).

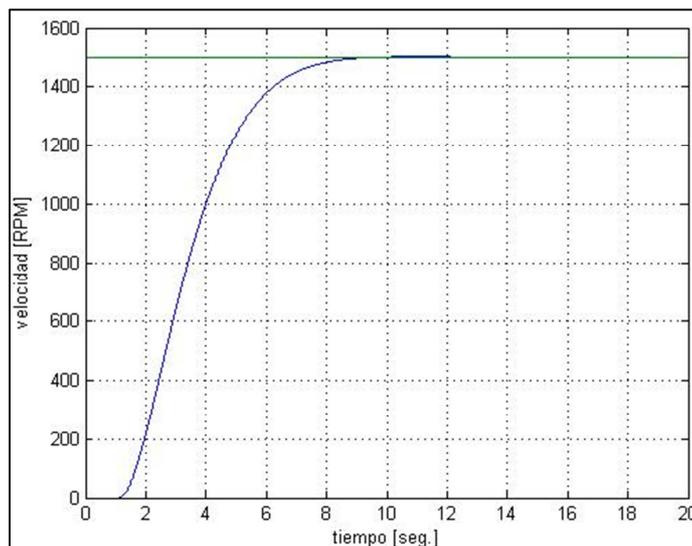


Figura 5.15: Salida del sistema con periodo de muestro de 2 ms.

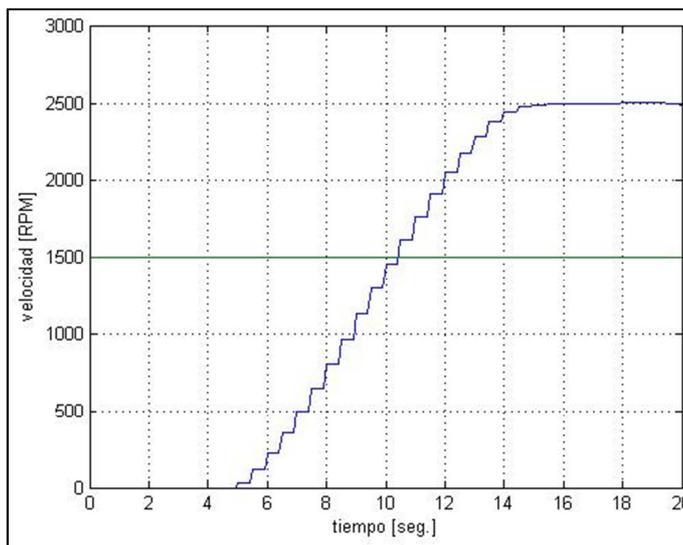


Figura 5.16: Salida del sistema con periodo de muestro de 50 ms.

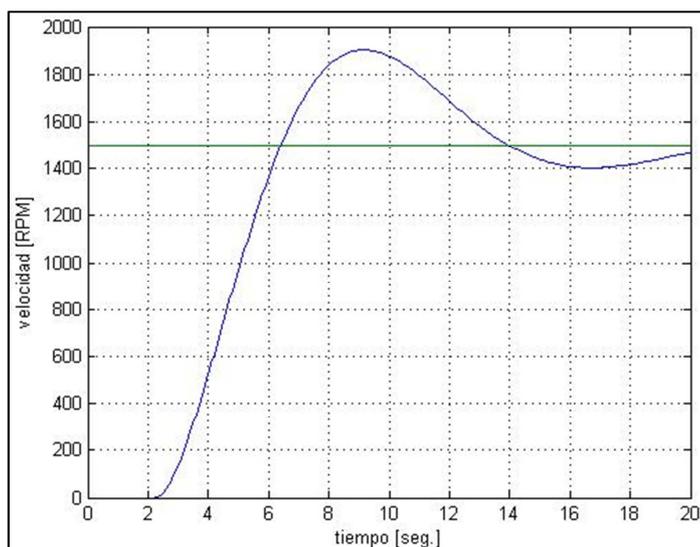


Figura 5.17: Salida del sistema con un periodo de muestro de 100 ms.

5.6 Implementación

Se implementa el lazo de control para el controlador digital con el modelo experimental, el cual contiene las entradas y salidas analógicas de la tarjeta de adquisición de datos respectivamente acondicionadas (figura 5.18).

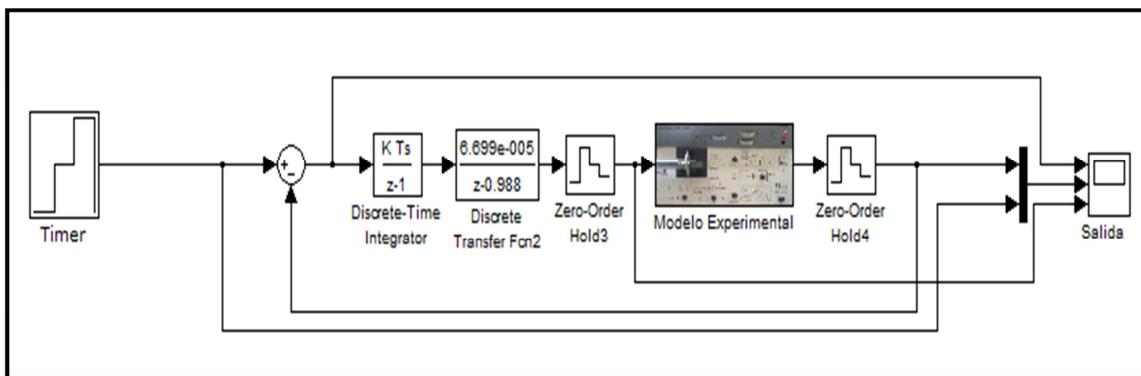


Figura 5.18: Lazo de control con controlador digital

Al someter el lazo de control a diferentes niveles de entrada con un periodo de muestro de 10 ms, se obtuvieron las señales de salida del sistema controlada (figura 5.19), señal del controlador (figura 5.20) y señal de error (figura 5.21).

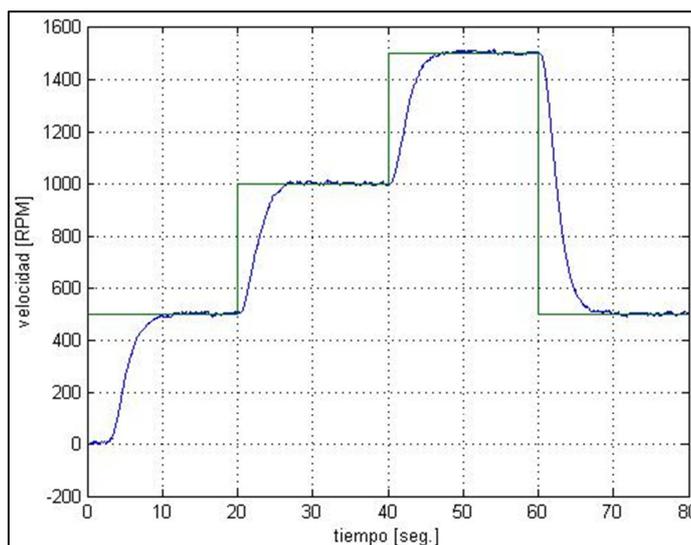


Figura 5.19: Señal de salida del sistema controlado

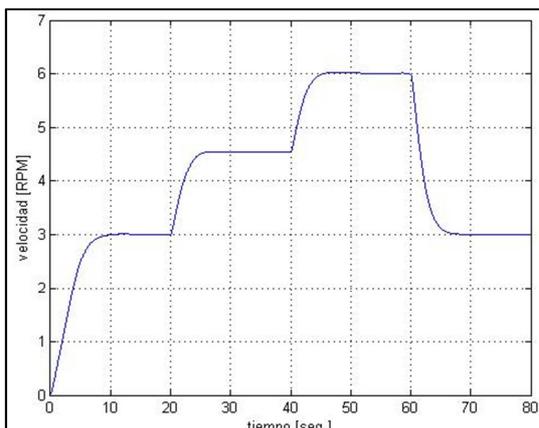


Figura 5.20: Señal de salida del controlador

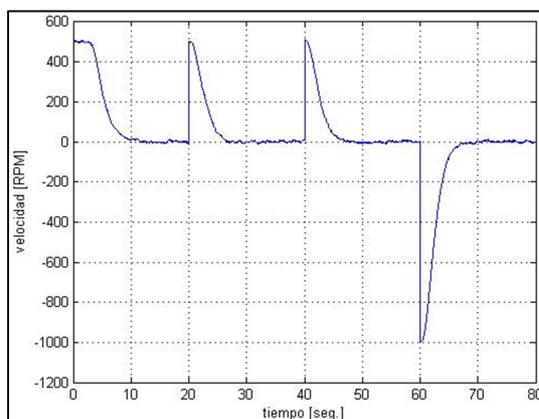


Figura 5.21: Señal del error.

De la misma manera que en la simulación, se realiza la prueba del controlador para los período de 2ms (figura 5.22), 50 ms (figura 5.23) y 100 ms (figura 5.24).

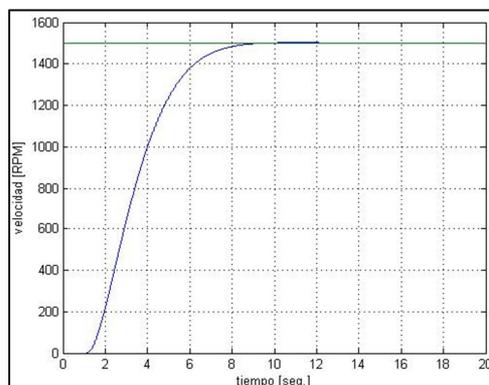


Figura 5.22: Salida del sistema con periodo de muestro de 2 ms.

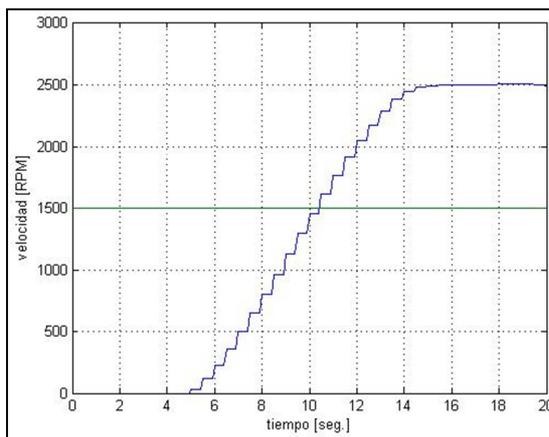


Figura 5.23: Salida del sistema con periodo de muestro de 50 ms.

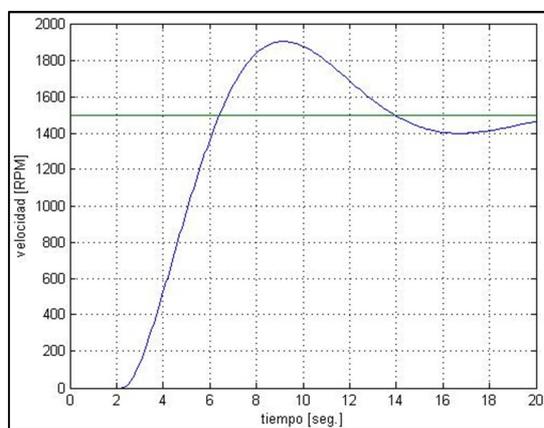


Figura 5.24: Salida del sistema con un periodo de muestro de 100 ms.

CAPÍTULO VI

APLICACIÓN DE LA HERRAMIENTA RTW PARA LOS CONTROLADORES INTELIGENTES

7.1 Controlador Predictivo

7.1.1 Marco Teórico

El control predictivo tiene como objetivo resolver de forma efectiva, problemas de control y automatización de procesos industriales que se caractericen por presentar un comportamiento dinámico complicado, multivariable, y/o inestable. La estrategia de control en que se basa este tipo de control, utiliza el modelo matemático del proceso a controlar para predecir el comportamiento futuro de dicho sistema, y en base a este comportamiento futuro puede predecir la señal de control futura.

El Control Predictivo Basado en Modelo (CPBM), conocido también como Model Based Predictive Control (MBPC) o simplemente Model Predictive Control (MPC). Esta estrategia también se conoce como control por horizonte deslizante, por ser ésta la forma en la que se aplican las señales de actuación. [5]

El control predictivo basado en modelo se puede definir como una estrategia de control que se basa en la utilización de forma explícita de un modelo matemático interno del proceso a controlar (modelo de predicción), el cual se utiliza para predecir la evolución de las variables a controlar a lo largo de un horizonte temporal de predicción especificado por el operador, de este modo se puede calcular las variables manipuladas futuras (señal de control futura) para lograr que en dicho horizonte, las variables controladas converjan en sus respectivos valores de referencia.

En Matlab existe la herramienta MPCTool que permite realizar diferentes tipos de controladores predictivos que pueden ser adaptados a la necesidad del usuario.

7.1.2 Desarrollo

Para el diseño del controlador predictivo se utiliza la herramienta mpctool de Matlab, en la cual es necesario utilizar una función de transferencia lineal del sistema.

La función de transferencia que se utiliza es la siguiente:

$$H(s) = \frac{4004}{s^2 + 10.51s + 15.02}$$

Conocida ya la función de transferencia del sistema, se establecen los valores de funcionamiento de la planta y los parámetros de diseño para el control predictivo, estos parámetros y valores están especificados en la tabla 6.1.

Parámetro	Detalle	Valor
Modelo y Horizonte	Intervalo de Control	0.1
	Horizonte de Predicción	10
	Horizonte de Control	4
Restricciones	Entrada : Voltaje	Min: 3 Max: 6.5
	Salida: Velocidad	Min: 500 Max: 1500
Pesos	Overall	0.6
	Varianza del peso	0.1
	Velocidad	1

Tabla 6.1: Parámetros de diseño para el controlador predictivo.

Al conocer ya los parámetros del controlador predictivo, se genera una estructura de datos la cual llamaremos CPREDICTIVO.

Para la simulación, se implementa un lazo de control con el bloque MPC controller, desde el cual se coloca la estructura de datos CPREDICTIVO (figura 6.1).

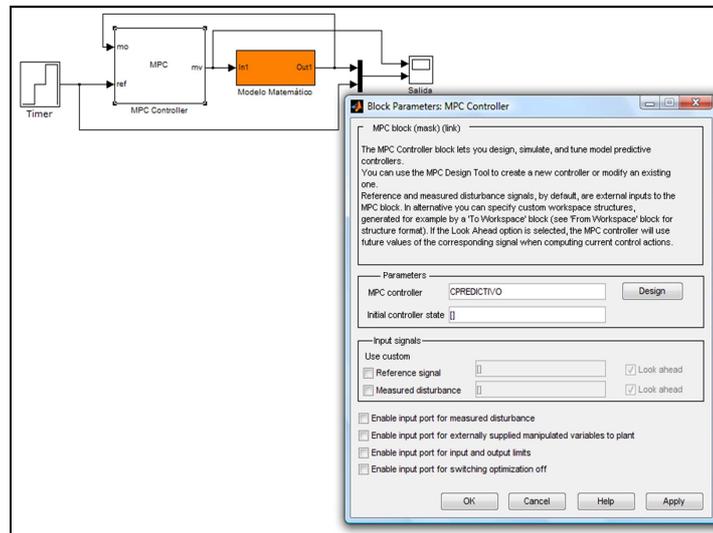


Figura 6.1: Lazo de control con controlador predictivo.

Al simular el sistema, se obtienen las señales de salida del sistema controlado (figura 6.2) y la señal del controlador (figura 6.3).

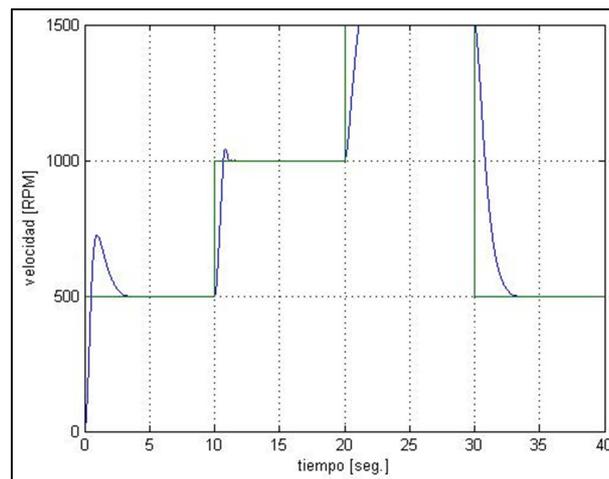


Figura 6.2: Señal de salida del sistema controlado simulado.

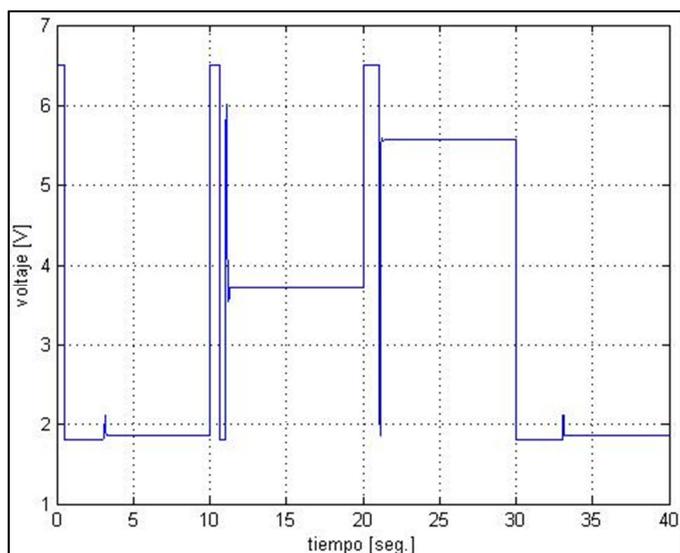


Figura 6.3: Señal de control del sistema simulado.

Para realizar el experimento con el módulo de velocidad, se enlaza el controlador predictivo al modelo experimental, del cual se obtienen las señales del sistema controlado (figura 6.4) y la señal de control (figura 6.5).

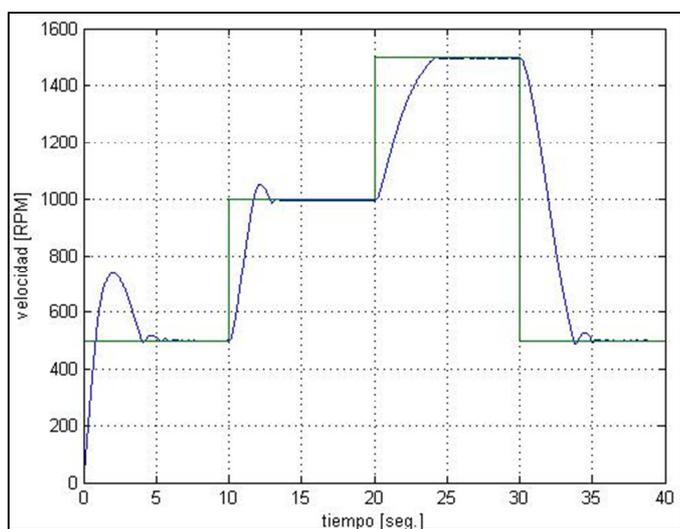


Figura 6.4: Señal de salida del sistema controlado real.

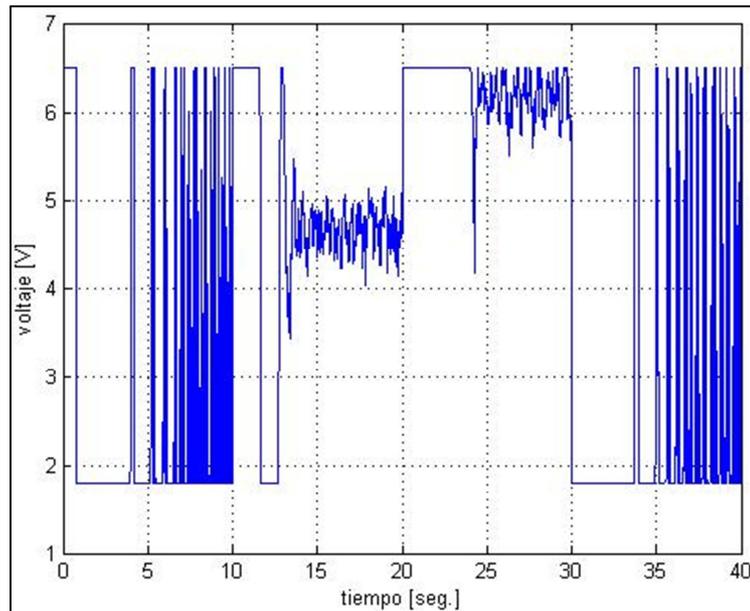


Figura 6.5: Señal de control del sistema real.

7.2 Controlador PI Fuzzy

7.2.1 Marco Teórico

La lógica fuzzy se presenta como una técnica de la Inteligencia Artificial cuyo objetivo es expresar el conocimiento común, el cual es de tipo lingüístico cualitativo, en un lenguaje matemático regido por la teoría de conjuntos difusos y funciones de pertenencia asociados a estos [6].

La lógica fuzzy está formada por conjuntos borrosos. Un conjunto borroso es un conjunto en donde la pertenencia de sus elementos se define mediante una función denominada función de pertenencia (figura 6.6).

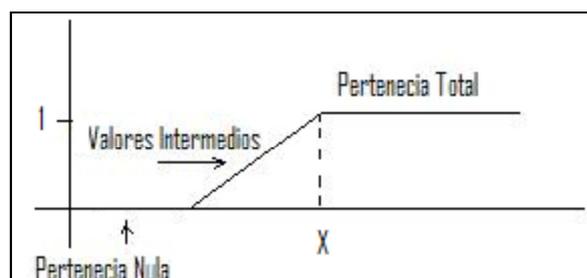


Figura 6.6: Conjunto difuso

Esa función de pertenencia marcará si un valor pertenece totalmente a un conjunto, no pertenece o bien pertenece más o menos a ese conjunto (figura 6.6). Esos valores intermedios son los que marcan las diferencias respecto a los conjuntos clásicos en los que las funciones de pertenencia devuelven los valores uno o cero, dependiendo de si el valor pertenece o no al conjunto.

La función de pertenencia puede ser representada con formas trapezoidal o triangular. Y también puede representarse con formas más teóricas como de campana, sinusoidal o irregulares.

El control difuso está compuesto por la fuzzificación, Motor de Inferencia y la defuzzificación.

La fuzzyficación transforma una medida del entorno a una representación en forma de función de posibilidad como se muestra en la figura 6.7.

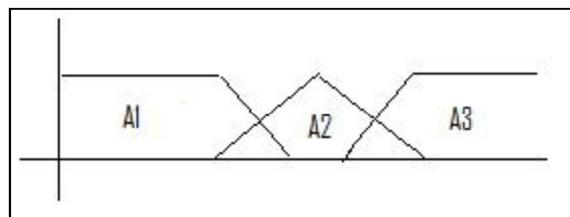


Figura 6.7: Fuzzyficación

El motor de inferencia está formado por un conjunto de reglas lingüísticas de control que tienen la forma:

1. X es A1 & Y es B1 -> Z es C1
2. X es A2 & Y es B2-> Z es C2

Donde X e Y serían unas variables de entrada, A1, A2 conjuntos borrosos de X, B1, B2 conjuntos borrosos de Y. Z sería una variable de salida, y C1, C2 conjuntos borrosos de Z.

El motor de inferencia si tuviera como entrada un hecho A' y otro B' daría como resultado C'.

La defuzzyficación es el proceso inverso a la fuzzyficación. Transforma el conjunto fuzzy final (figura 6.8) en un valor numérico, el método de centro de gravedad es el más conocido.

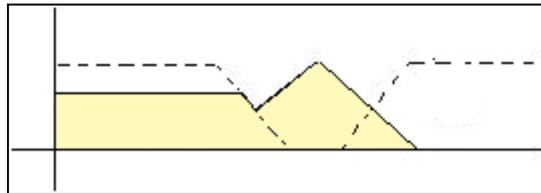


Figura 6.8: Defuzzyficacion

Para realizar todos los pasos para generar un controlador fuzzy Matlab cuenta con el toolbox Fuzzy.

Conviene señalar que la aplicación de la lógica difusa en el control de procesos puede combinarse con otras técnicas de control convencionales, formando estructuras de control muy diversas como el controlador de tipo PI Fuzzy.

7.2.2 Desarrollo

Para el diseño del controlador PI Fuzzy, se deben establecer las siguientes variables o conjuntos difusos: error, salida de voltaje e integral del error.

Para el error se establece un conjunto que va desde -1500 a 1500 RPM.

Para la integral del error se establece un conjunto que va desde los -10.000 a 10.000 RPM.

Finalmente, la salida de voltaje se establece en un rango de -7 a 7 voltios.

En el caso del error (figura 6.9), los conjuntos de pertenencia son designados como: error negativo (EN), error en cero (EZ) y error positivo (EP).

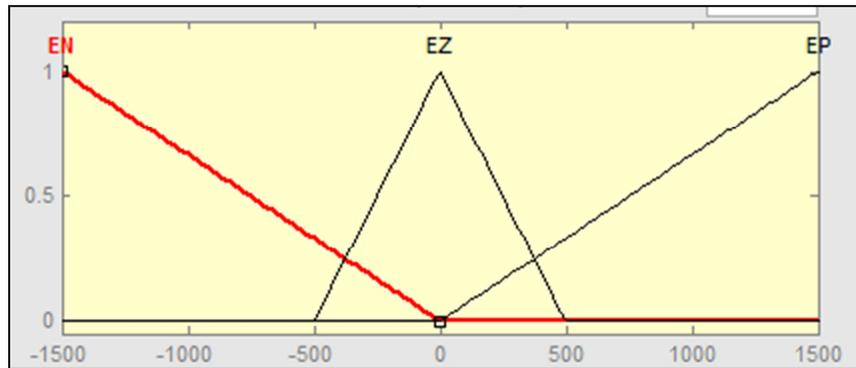


Figura 6.9: Conjuntos de pertenencia para el error.

Para la integral del error (figura 6.10), los conjuntos de pertenencia son designados como: integral del error negativo (EN), integral de error en cero (EZ) e integral del error positivo (EP).

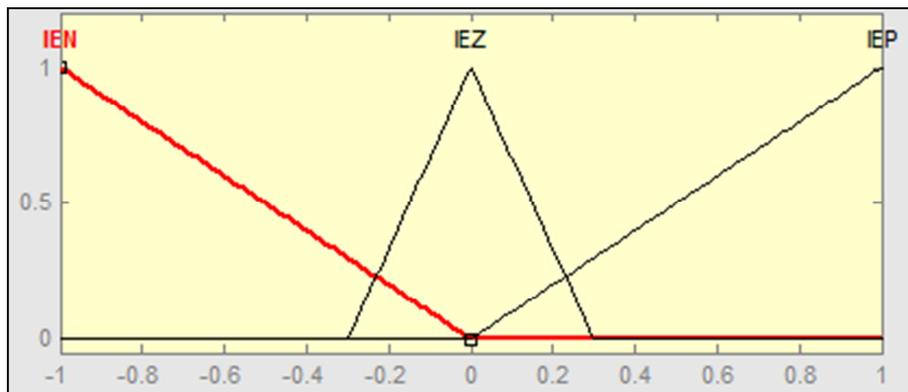


Figura 6.10: Conjuntos de pertenencia para la integral del error.

En el caso de la salida de voltaje, se establecen 5 conjuntos de pertenencia (figura 6.11), los cuales son designados de la siguiente manera: voltaje grande negativo (VGN), voltaje pequeño negativo (VPN), voltaje en cero (VZ), voltaje pequeño positivo (VPP) y voltaje grande positivo (VGP).

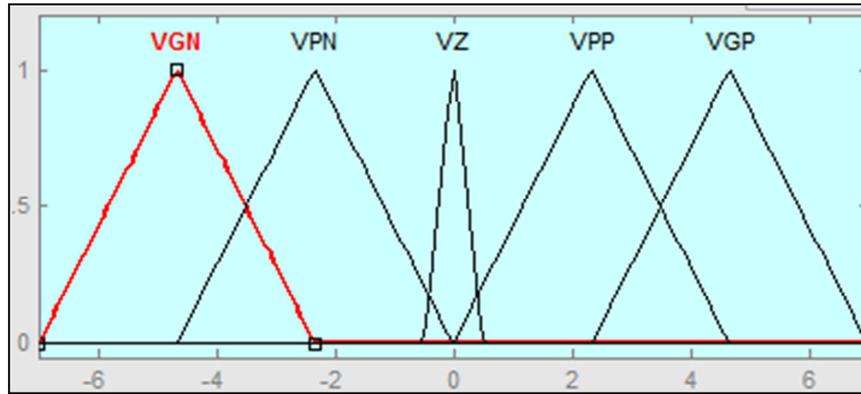


Figura 6.11: Conjuntos de pertenencia para la salida de voltaje.

Luego de establecer los conjuntos de pertenencia para cada una de las variables, se realiza las combinaciones necesarias para formar la base de reglas. La base de reglas que se utiliza está basada en función de la máquina de inferencias de Mandami.

En la tabla 6.2, podemos observar las diferentes combinaciones de las variables de error e integral de error con su respectivo efecto que es la variable salida de voltaje.

	Reglas
1	si(error es negativo y cambio de error es negativo)entonces salida grande negativo
2	si(error es negativo y cambio de error es cero)entonces salida pequeño negativo
3	si(error es negativo y cambio de error es positivo)entonces salida cero
4	si(error es cero y cambio de error es negativo)entonces salida pequeño negativo
5	si(error es cero y cambio de error es cero)entonces salida cero
6	si(error es cero y cambio de error es positivo)entonces salida pequeño positivo
7	si(error es positivo y cambio de error es negativo)entonces salida cero
8	si(error es positivo y cambio de error es cero)entonces salida pequeño positivo
9	si(error es positivo y cambio de error es positivo)entonces salida grande positivo

Tabla 6.2: Base de reglas para el controlador PI Fuzzy

Como éste controlador posee constantes que necesitan ser sintonizadas para ayudar a realizar el control, se realiza un programa en Matlab (Anexo 5), el cual resuelve un problema de optimización para encontrar las constantes K_p y k_i de controlador PI Fuzzy.

Para realizar la simulación del controlador PI Fuzzy, se enlaza los elementos que nos permitan realizar dicho controlador (figura 6.12). Del cual al realizar la simulación se obtienen la señal de la salida del sistema controlado (figura 6.13), señal del controlador (figura 6.14) y la señal de erro (figura 6.15).

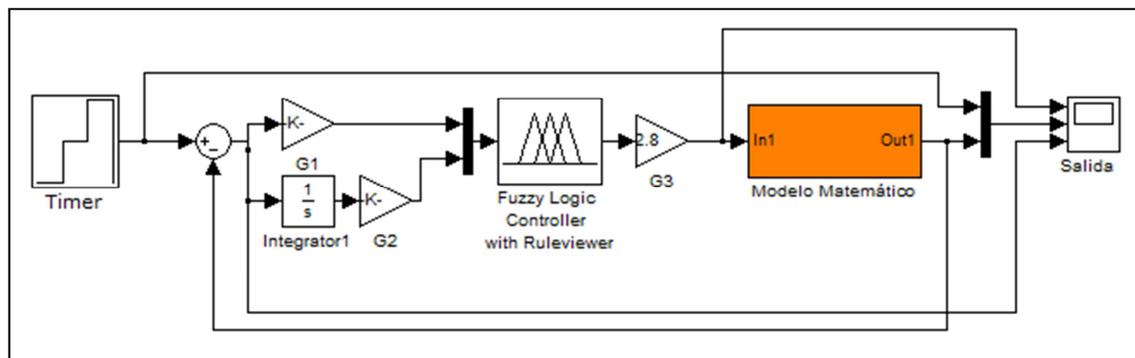


Figura 6.12: Simulación del controlador PI Fuzzy.

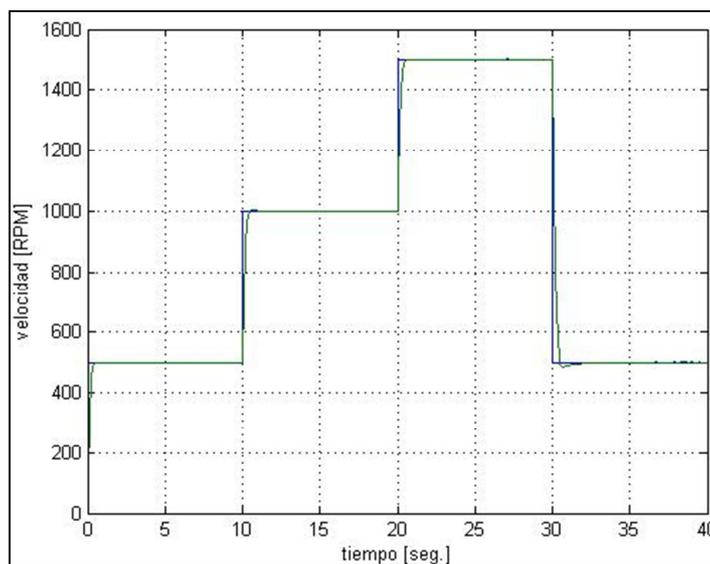


Figura 6.13: Señal de salida controlada del sistema simulado.

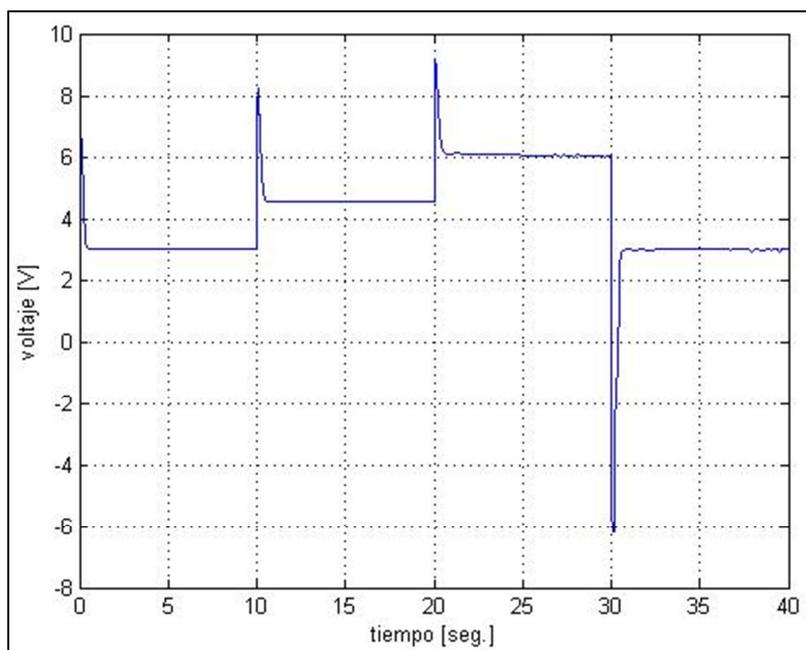


Figura 6.14: Señal de salida de control del sistema simulado.

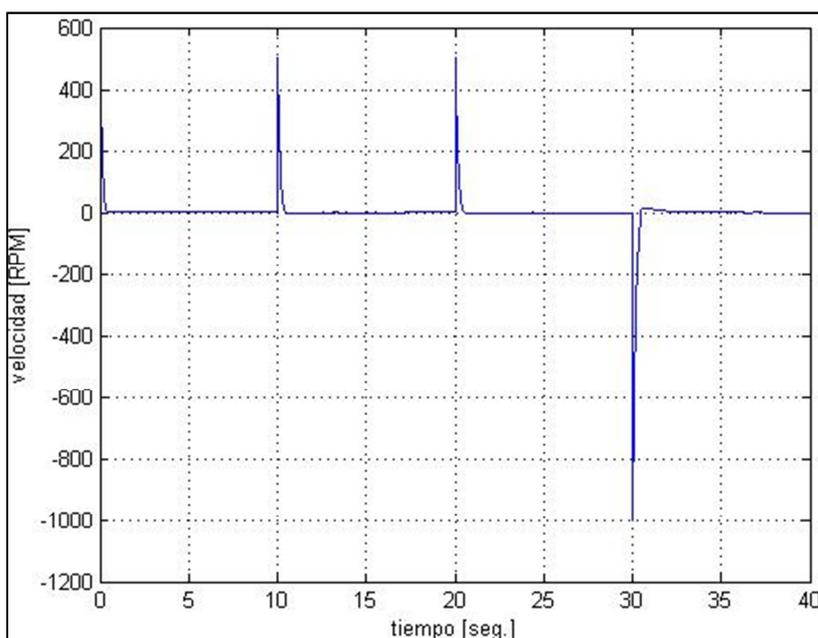


Figura 6.15: Señal de salida de error del sistema simulado.

Finalmente, colocamos el controlador PI Fuzzy con el modelo experimental (figura 6.16), del cual se obtienen las señales de salida del sistema controlado (figura 6.17), señal de error (figura 6.18) y señal de control (figura 6.19).

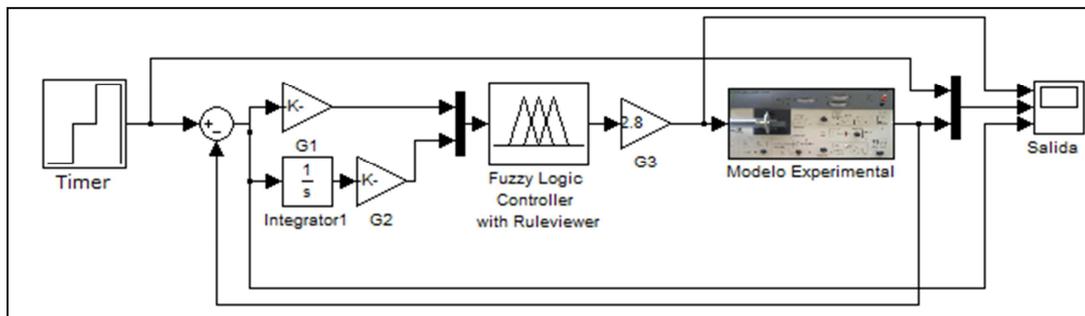


Figura 6.16: Controlador PI Fuzzy con modelo experimental.

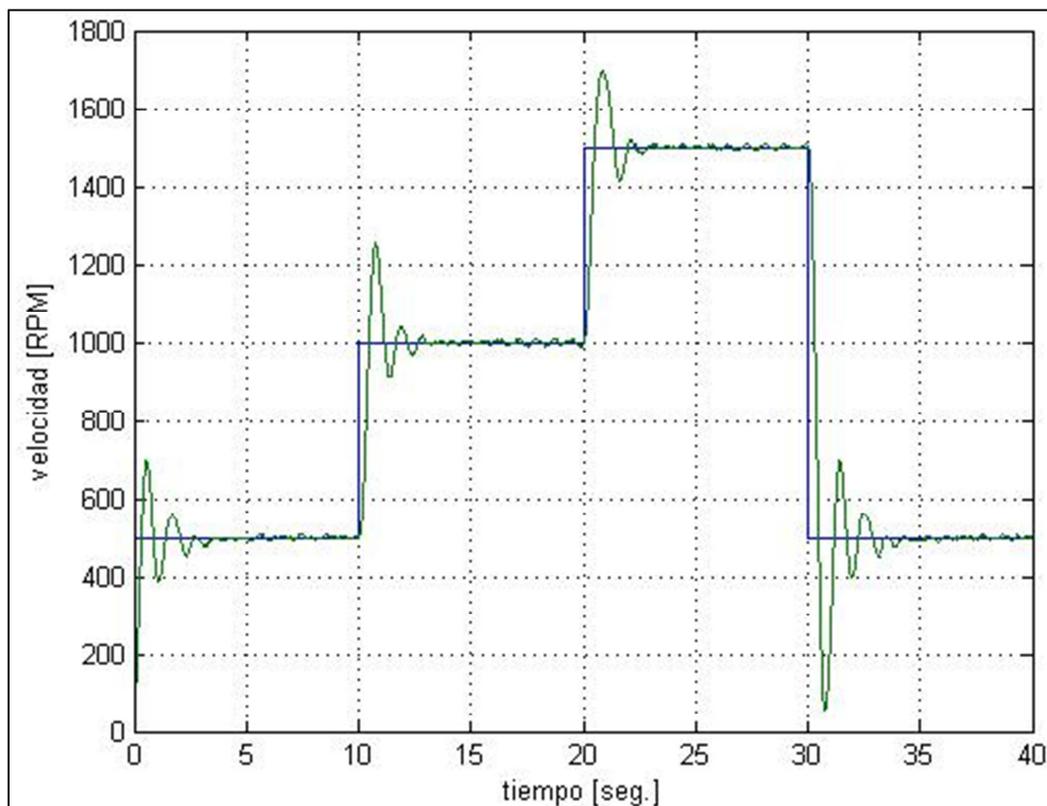


Figura 6.17: Señal de salida del sistema controlado

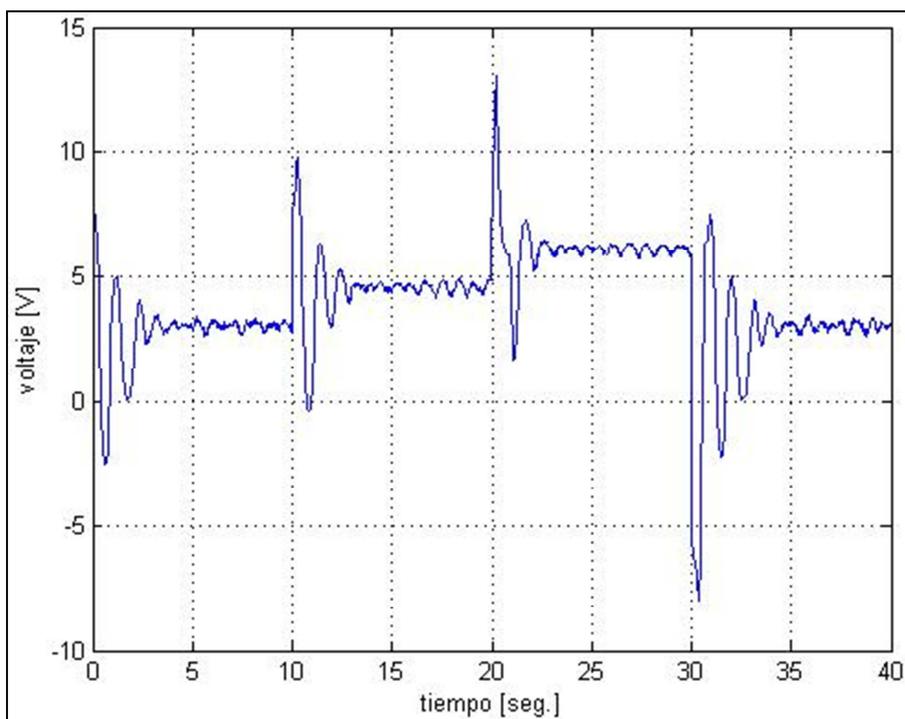


Figura 6.18: Señal de salida del controlador.

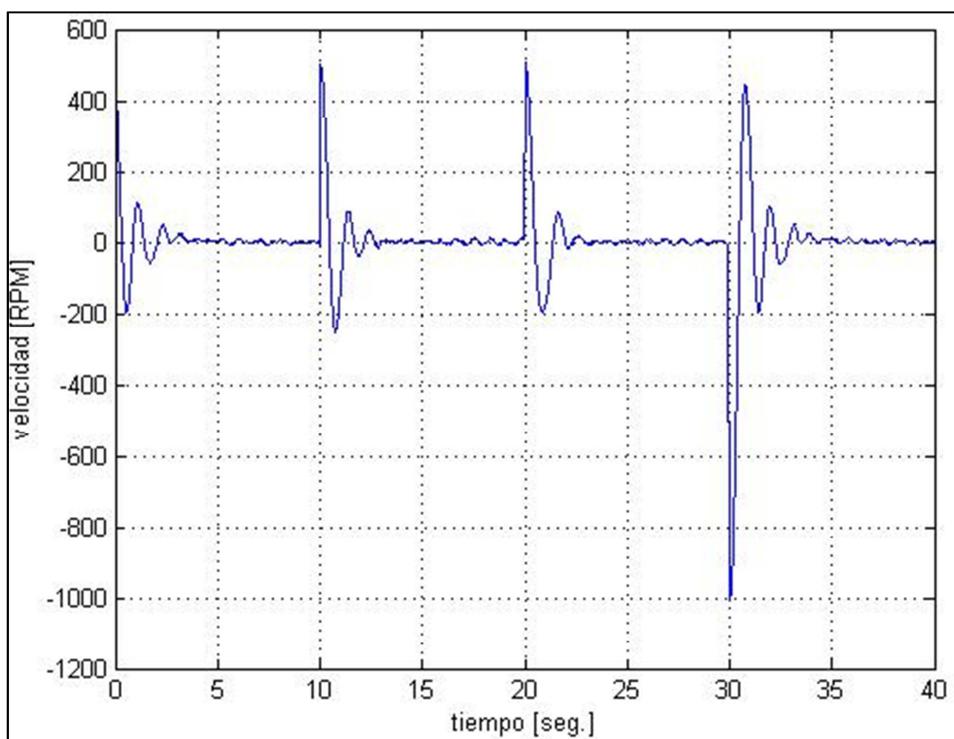


Figura 6.19: Señal de salida de error.

7.3 Controlador Neuronal por modelo de referencia

7.3.1 Marco Teórico

El control de sistemas por modelo de referencia consiste en diseñar un sistema que modifique el comportamiento natural de la planta con el objetivo de aproximar la respuesta que tiene un modelo de referencia establecido.

En este esquema de control el diseñador tiene el suficiente conocimiento del sistema como para definir el comportamiento deseado por medio de un modelo.

Uno de los primeros trabajos de la aplicación de las redes neuronales para el control por modelo de referencia de sistemas puede consultarse en (K. S. Narendra, K. Parthasarathy, 1990) [7].

7.3.2 Desarrollo

Para realizar el controlador neuronal, se identifica la unidad de velocidad por medio de una red neuronal, usando un conjunto de valores que realizan la identificación de la misma, se define una variable denominada estado la cual está comprendida entre 240 a 1573 rpm y una variable denominada entrada la cual va de 2 a 7 voltios.

```

entrada=2:7;
estado=240:100:1573;
Pm=combvec(estado,entrada);
timestep=5;
Tm=[];
length(Pm)

```

Los patrones de entrenamiento están definidos en la matriz Pm mientras que la matriz de objetivos está definida por la matriz Tm que representan los valores de entrada y salida de la red neuronal identificada.

```

for i=1:length(Pm)
    Vin=Pm(2,i);
    Vel_in=Pm(1,i);
    sim('\Velocidad_Modelo',[0 timestep]);
    dVel=Vel_out-Vel_in
    Tm=[Tm [dVel]];

```

end

Los valores de Tm son obtenidos al simular las diferentes combinaciones que tiene Pm,

Para la determinación de la red neuronal de control se lo realiza por medio de un modelo de primer orden del tipo:

$$G(s) = \frac{1}{1 + \tau s}$$

Que imita al modelo de la planta de velocidad, después de varias pruebas se encontró de que valor de τ es de 1.4 seg.

Para la red neuronal de control se evalúa la respuesta de la planta de primer orden, con respecto a los valores obtenidos de Pc, para posteriormente encontrar los valores de Tc.

```

entrada=2:7;
estado=240:100:1573;
% Matriz Pc y Tc
Pc=combvec(estado,entrada);%combinación de los vectores de estados
iniciales
%matriz de estados iniciales "Pc"
timestep = 10;
Q = length(Pc);
Tc = []; %matriz de objetivos "Tc".
for i=1:Q
    Vin = Pc(2,i);
    Vel_in = Pc(1,i);
    %Modelo para el aprendizaje
    sim('Velocidad_Modelo_Lineal',[0 timestep]);
    dVel = Vel_out-Vel_in;
    Tc=[Tc [dVel]];
end

```

Al conocer ya los valores de Pc y Tc, se crea una red neuronal con 8 neuronas en la capa oculta las cuales son de tipo tansig y purelin

```

S1 = 8;
cnet = newff(minmax(Pc),[S1 1],{'tansig' 'purelin'});

```

La red total de control viene a ser dada por la red neuronal identificada mnet y la red de control cnet, distribuidas de forma que dos capas sean usadas para la identificación mnet y dos capas para el control cnet (figura 6.20).

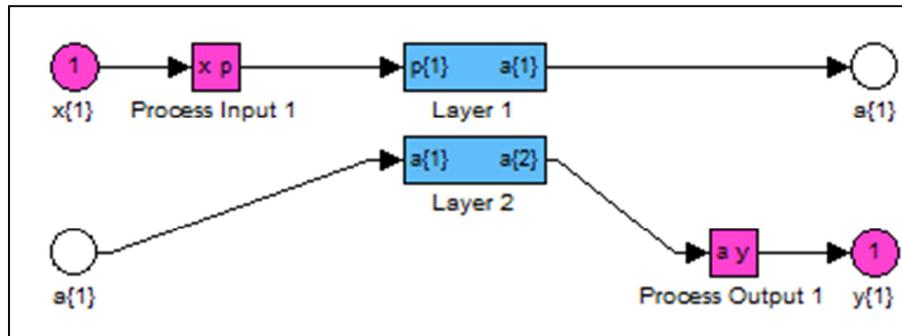


Figura6.20: Asignación de la red neuronal total de control.

Finalmente se hace la asignación y aprendizaje de los pesos y bias de las entradas de la red neuronal de control cnet a red total de control tnet

```
cnet.IW{1,1}(:,1) = tnet.IW{1,1};
cnet.IW{1,1}(:,2) = tnet.IW{1,2};
cnet.b{1} = tnet.b{1};
cnet.b{2} = tnet.b{2};
cnet.LW{2,1} = tnet.LW{2,1};
```

Para la simulación del controlador neuronal tnet se realiza el lazo de control con el modelo matemático (figura6.21), del cual se obtiene la señal de salida del sistema controlado(figura 6.22).

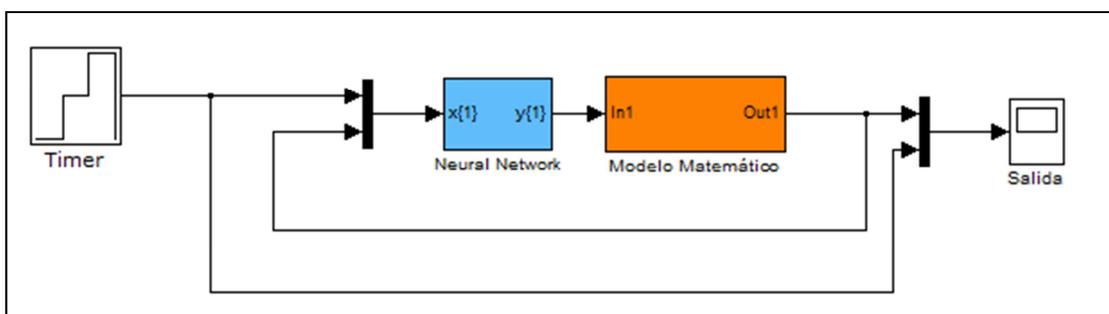


Figura 6.21: Simulación del controlador neuronal.

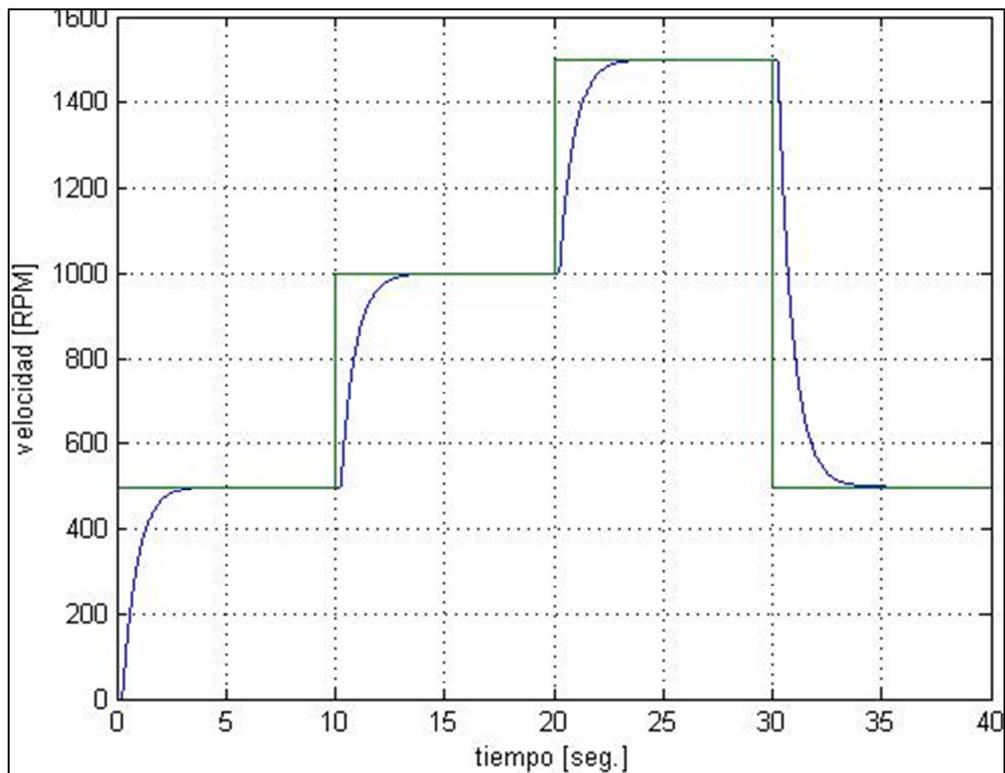


Figura 6.22: Señal de salida del sistema controlado simulado.

Finalmente, colocamos el controlador neuronal con el modelo experimental (figura 6.23), del cual se obtienen las señales de salida del sistema controlado (figura 6.24).

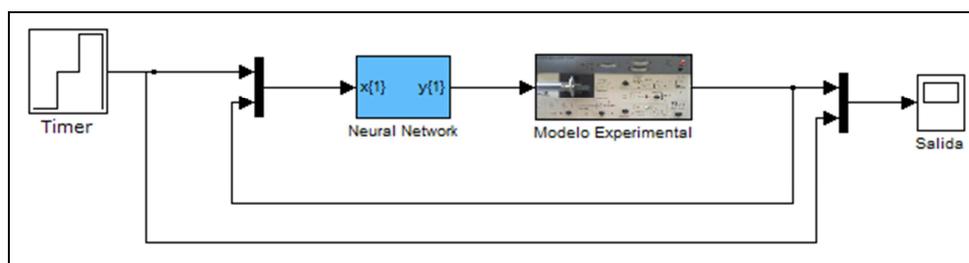


Figura 6.23: Controlador neuronal con modelo experimental

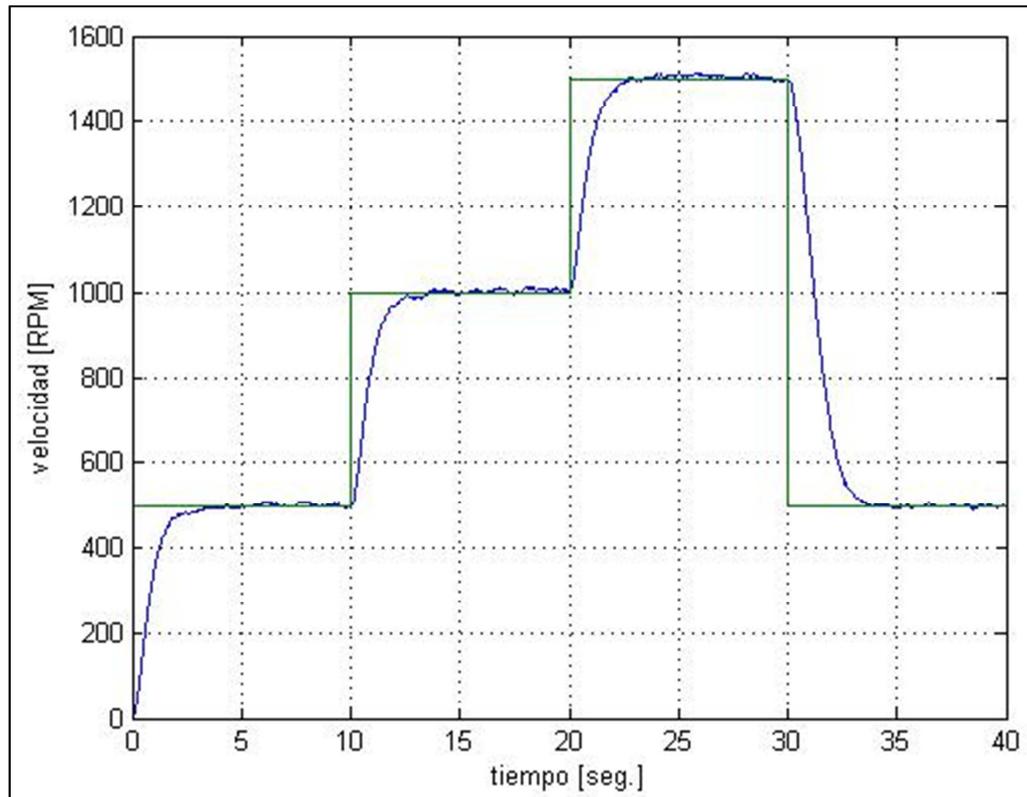


Figura 6.24: Señal de salida del sistema controlado.

CAPÍTULO VII

CONCLUSIONES Y RECOMENDACIONES

8.1 Conclusiones

- El entrenamiento continuo en diferentes técnicas de control es una de las tareas que como ingenieros debemos tener siempre en mente, ya que la tecnología avanza a pasos agigantados y cada vez es más sencilla su diseño e implementación.
- La unidad de velocidad DC Speed Control System es una herramienta que permitió realizar la implementación de las diferentes técnica de control generando respuestas aceptables.
- El uso de una técnica de identificación de Smith, ha generado una función de transferencia con una respuesta acorde a las necesidades del proyecto.
- Los algoritmos de optimización como el fminsearch, son técnicas poco usadas al momento de realizar un controlador, no obstante al implementarlo en los controladores PI, PID y PIFuzzy han generando respuestas muy aceptables.
- Es muy importante realizar un adecuado análisis para la ubicación de los polos ya que de estos dependerá la generación de nuestro controlador digital.
- Para la resolución del polinomio característico en lazo cerrado es muy importante que en la función de transferencia $\frac{B(z)}{A(z)}$ dada por el modelo, los polinomios $B(z)$ y $A(z)$ no tengan factores en común y que el grado del polinomio $A(z)$ sea mayor al grado del polinomio $B(z)$.

- El control predictivo, neuronal y PIfuzzy son técnicas de control inteligente que gracias a las herramientas de Matlab no fue necesario realizar un análisis matemático detallado de cada una de las técnicas.
- Para el controlador predictivo y neuronal por modelo de referencia, fue necesario utilizar un modelo matemático que describa la dinámica del modelo real.
- Para el controlador por redes neuronales, se observó que se debe tener un modelo matemático que genere una dinámica aproximada a la que genera la unidad de velocidad.
- Al final del proyecto se desarrolló una interfaz gráfica que permite fácilmente realizar todos los pasos requeridos para la implementación de controladores en la unidad DC Speed Control System.

8.2 Recomendaciones

- Se recomienda dar un mantenimiento preventivo y correctivo a todos los elementos del Laboratorio, ya que por ser equipos antiguos tienden a generar fallas.
- Se debe utilizar el presente proyecto en una versión de Matlab 2010 b, ya que si se la utiliza con una versión menor los programas no van a poder ejecutarse de manera correcta.
- Se recomienda utilizar los controladores únicamente en un rango de 1.4 a 9 voltios, de esta manera los controladores realizarán sus acciones de control de forma correcta.
- Se recomienda utilizar los controladores predictivos para procesos con respuesta lenta, ya que este tipo de controlador necesita de un tiempo de cálculo para obtener su señal de control.

REFERENCIAS BIBLIOGRAFIA

- [1] Hossein Arsham. "*Modelos Deterministas: Optimización Lineal*", 1994
- [2] Optimization Toolbox For Use with MATLAB®. "*Optimization Toolbox User's Guide*". Version 3.
- [3] http://www.ramonmedina.name/files/todec_g.pdf.
- [4] Enrique Castillo, Antonio J. Conejo, Pablo Pedregal, Ricardo García y Natalia Alguacil "*Formulación y Resolución de Modelos de Programación Matemática en Ingeniería y Ciencia*".20 de febrero de 2002.
- [5] <http://www.revistaciencias.com/publicaciones/EEVFFkEEEupegfSOxr.php>
- [6] http://eprints.ucm.es/9915/1/Controlador_Fuzzy_de_un_Cuadrotor.pdf
- [7] <http://www.scielo.org.mx/pdf/cys/v6n4/v6n4a5.pdf>