

# **Universidad Central “Marta Abreu” de Las Villas**

**Facultad de Ingeniería Eléctrica**

**Departamento de Automática y Sistemas  
Computacionales**



## **TRABAJO DE DIPLOMA**

**Entrenador digital de lazo de control con  
consideraciones prácticas reales**

Autor: Loan Dani Rodríguez Herrera

Tutor: Dr. Roberto Luis Ballesteros Horta

Santa Clara

2016

"Año 58 de la Revolución"

*“El éxito no es definitivo, el fracaso no es fatal: es el coraje  
para continuar lo que cuenta”*

*Winston*

*Churchill*

*A mi familia y a mi novia por estar en los momentos buenos  
y malos.*

*A mi tutor Roberto Ballesteros por su paciencia y apoyo en  
todo este tiempo.*

*A mis compañeros de aula por ser motivo de inspiración y  
apoyo en el transcurso de la carrera.*

*A mis padres y hermanos que han estado presentes en toda  
mi carrera universitaria.*

## **Resumen**

En la actualidad los entrenadores de lazo de control contribuyen a la adquisición de conocimientos y habilidades en situaciones cotidianas que se imitan con la simulación. La presente investigación se realiza para el laboratorio de computación de la carrera de Ingeniería Automática con el objetivo de diseñar un entrenador digital de lazo de control, que permita la operación en tiempo real de consideraciones prácticas reales. A partir de esto se realiza el diseño del mismo en el software LabVIEW. Para la valoración de los resultados se analizan los casos de: control de velocidad de un motor de corriente directa y el control de posición de un servomotor de corriente directa. Para cada caso se simula con los valores de configuración inicial y posteriormente se introducen varias consideraciones prácticas para interpretar el efecto en la salida del sistema. Finalmente se arriba a las conclusiones y recomendaciones del trabajo demostrando resultados favorables.

## Índice

Introducción .....	1
Capítulo I: Entrenador de lazo de control con consideraciones prácticas reales	4
1.1 Tiempo real.....	4
1.1.1 Software para trabajo en tiempo real .....	4
1.2 Simulación .....	7
1.2.1 Entrenador.....	8
1.2.2 Simulador de procesos .....	8
1.3 Sistemas de control.....	10
1.3.1 Clasificación general de sistemas de control .....	11
1.3.2 Controlador PID.....	11
1.4 Consideraciones prácticas en la implementación de algoritmos de control	15
1.5 Antecedentes de entrenadores de procesos .....	16
1.6 Consideraciones finales del capítulo .....	18
Capítulo II: Diseño del entrenador digital de lazo de control .....	19
2.1 Arquitectura del entrenador .....	19
2.2 Sistema de control .....	19
2.2.1 Controlador PID.....	20
2.2.2 Filtro paso bajo.....	22
2.2.3 Convertidor digital-analógico .....	23
2.2.4 Actuador .....	24
2.2.5 Proceso .....	25
2.3 Consideraciones prácticas.....	28
2.4 Interfaz del entrenador de lazo de control.....	29
2.5 Consideraciones finales del capítulo .....	31
Capítulo III: Análisis de los resultados.....	32
3.1 Control de velocidad de un motor de corriente directa .....	32
3.1.1 Configuración para la simulación .....	32
3.1.2 Consideraciones prácticas para el control de velocidad del motor de corriente directa .....	34
3.2 Control de posición de un servomotor de corriente directa .....	40
3.2.1 Configuración para la simulación. ....	40

3.2.2 Consideraciones prácticas para el control de posición del servomotor de corriente directa .....	41
3.3 Valoración económica.....	44
3.4 Consideraciones finales del capítulo .....	44
Conclusiones .....	45
Recomendaciones.....	46
Bibliografía .....	47



## **Introducción**

La simulación se ha utilizado en el descubrimiento de soluciones exactas a problemas clásicos de la industria y la ingeniería. Durante los últimos años el avance de la informática y las tecnologías ha posibilitado el uso de la simulación prácticamente en todos los ámbitos de la ciencia y la ingeniería.

La construcción de los primeros computadores de propósito general como el ENIAC a mediados de los años 40 y el desarrollo de los primeros programas de simulación general en los años 60, sentaron las bases de la simulación en el campo del proceso de control industrial (Lander Simulation & Training Solutions, 2015).

La simulación se encuentra relacionada con el entrenamiento. En la actualidad se han creado entrenadores con el objetivo de adquirir conocimientos y habilidades en situaciones cotidianas que se imitan con la simulación. El entrenamiento de pilotos de aviación en simuladores físicos, el uso de simuladores en la industria petrolera para la optimización y reducción de costos y por último los entrenadores de sencillos sistemas de control son algunos de estos ejemplos.

Actualmente en los laboratorios de computación de la carrera de Ingeniería Automática de la Universidad Central “Marta Abreu” de Las Villas no se dispone de entrenadores de lazo de control que reproduzcan los diversos procesos que han surgido en las industrias. En las simulaciones no se tiene en cuenta el tiempo real y las consideraciones prácticas reales en estos sistemas de control. Estos elementos traen consigo que estudiantes y profesores no puedan realizar prácticas y entrenamiento en tiempo real de estos sistemas de control. El análisis y diseño de los lazos de control se realiza a partir de estrategias convencionales. De esta manera los estudiantes y profesores no se benefician de las ventajas de la simulación en tiempo real.

El problema científico que se plantea es la ausencia de entrenadores digitales de lazos de control con consideraciones prácticas reales en los laboratorios de la carrera de Ingeniería Automática. De esta manera no se adquieren conocimientos y habilidades de la vida práctica para el análisis de los sistemas de control.

La hipótesis a demostrar es la siguiente: El uso del entrenador digital de lazo de control puede contribuir a la adquisición de conocimientos de los sistemas de control. De esta manera se permite un acercamiento de los estudiantes y profesores a situaciones reales de las industrias mediante el uso de consideraciones prácticas, sin la necesidad de grandes inversiones de tiempo y recursos.

Se plantea como objetivo general de la investigación: Diseñar un entrenador digital de lazo de control, que permita la operación en tiempo real de consideraciones prácticas reales.

Para el cumplimiento del objetivo general, se proyectan los objetivos específicos siguientes:

- Analizar la bibliografía existente sobre simuladores, entrenadores en tiempo real, lazos de control y consideraciones prácticas.
- Diseñar la arquitectura del entrenador de lazo de control.
- Diseñar los diferentes módulos del entrenador de lazo de control.
- Seleccionar las consideraciones prácticas a implementar.
- Desarrollar el software con las interface de comunicación hombre-máquina.
- Simular en tiempo real un proceso (LabVIEW) con consideraciones prácticas reales.
- Desarrollar pruebas en tiempo real con el proceso simulado, con consideraciones prácticas reales.
- Valorar los resultados de la simulación.

Con el análisis de diferentes consideraciones prácticas para un lazo de control en tiempo real se contribuye a la sistematización de conocimientos sobre el control de procesos industrial. Para esto se utiliza el entrenador de lazo de control, el cual es una herramienta factible en la simulación de sistemas de control y se aportan herramientas desarrolladas para el entrenamiento de usuarios en el estudio de estos sistemas.

El informe incluye tres capítulos, además de las conclusiones, recomendaciones y referencias bibliográficas. Los temas que se abordan en cada capítulo se estructuran de la forma siguiente:

Capítulo I: En este capítulo se exponen los conceptos generales de simulador, entrenador, controlador PID, los sistemas de control y consideraciones prácticas reales para el trabajo en tiempo real. Se presenta como antecedentes algunas aplicaciones sobre el uso de entrenadores de procesos en tiempo real.

Capítulo II: En este se describe la metodología de diseño del entrenador de lazo de control y se seleccionan las consideraciones prácticas que se utilizarán en la simulación.

Capítulo III: En este capítulo se realizan las simulaciones de los procesos seleccionados y posteriormente se muestran los resultados obtenidos en las pruebas.

## **Capítulo I: Entrenador de lazo de control con consideraciones prácticas reales**

Los entrenadores de procesos son ampliamente utilizados a nivel industrial. El uso de los mismos permite la preparación o entrenamiento del personal cuando es prohibitivamente caro o simplemente muy peligroso permitirle usar equipo real a un aprendiz en circunstancias reales. Estos aportan valiosas lecciones en un ambiente virtual seguro y posibilita que se cometan errores durante el entrenamiento.

En este capítulo se exponen los conceptos generales asociados con el entrenador de lazo de control. En el mismo se presenta los conceptos de tiempo real, simulador, entrenador, controlador PID, los sistemas de control y consideraciones prácticas reales para el trabajo en tiempo real. Se presenta algunos antecedentes sobre el uso de entrenadores de procesos en tiempo real y finalmente se exponen las consideraciones finales del capítulo.

### **1.1 Tiempo real**

Según (Veiga) un sistema de tiempo real puede definirse como un sistema que debe satisfacer restricciones explícitas en el tiempo de respuesta o arriesgarse a severas consecuencias, incluida la falla. Este es un sistema que responde a un estímulo externo dentro de un tiempo especificado y su eficiencia no solo depende de la exactitud de los resultados de cómputo, sino también del momento en que los entrega. La predictibilidad es su característica principal y a diferencia de los sistemas tradicionales, que tienden a distribuir en forma equitativa los recursos disponibles entre las diferentes tareas a ejecutar, los sistemas de tiempo real deben asegurar la distribución de recursos de tal forma que se cumplan los requerimientos de tiempo.

#### **1.1.1 Software para trabajo en tiempo real**

Actualmente, existen muchas aplicaciones para el trabajo y desarrollo de sistemas de tiempo real. Estas aplicaciones presentan una serie de herramientas y bibliotecas para tiempo real desarrolladas para interactuar activamente con un entorno con dinámica conocida en relación con sus entradas, salidas y restricciones temporales, para darle un correcto

funcionamiento de acuerdo con los conceptos de predicción, estabilidad, control y alcance. Algunas de estas aplicaciones se muestran a continuación.

### **LabVIEW**

El software LabVIEW (acrónimo de Laboratory Virtual Instrumentation Engineering Workbench) es una plataforma y entorno de desarrollo creado por National Instruments en 1976 para diseñar sistemas con un lenguaje de programación visual gráfico llamado Leguaje G (Holguín Londoño, Pérez Londoño, & Orozco Gutiérrez, 2002).

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales, o VI, y su origen provenía del control de instrumentos. Este es recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad, además puede combinarse con todo tipo de software y hardware, tanto del propio fabricante como de otros. Este presenta facilidades para el manejo de interfaces de comunicaciones, capacidad de interactuar con otros lenguajes y aplicaciones, herramientas gráficas y textuales para el procesamiento digital de señales, visualización y manejo de gráficas con datos dinámicos, adquisición y tratamiento de imágenes, control de movimiento, sincronización entre dispositivos, programación de FPGA para control o validación y herramientas para el trabajo en tiempo real.

Dentro de sus principales características encontramos su facilidad de uso y rapidez de programación. Con LabVIEW pueden crearse programas de miles de VI, equivalente a millones de páginas de código texto para aplicaciones complejas, programas de automatizaciones de decenas de miles de puntos de entradas-salidas (E/S) (Hernández Cevallos & Ledesma Marcalla, 2010).

### **MATLAB**

MATLAB es un lenguaje de programación de alto nivel orientado al cálculo técnico que integra un entorno amigable para el cálculo, la visualización de resultados y la codificación de programas. Entre sus prestaciones básicas se hallan: la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos

hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las cajas de herramientas y las de Simulink con los paquetes de bloques.

El nombre MATLAB proviene de Matrix Laboratory (Laboratorio de Matrices) dado que en sus orígenes fue escrito para facilitar el desarrollo de software matricial. MATLAB ha evolucionado desde 1970 a través de la atención de las necesidades de sus principales usuarios, tanto en ámbitos académicos como empresariales (Ataurima Arellato, 2013).

Tanto en el mundo universitario como en el industrial, MATLAB se ha convertido en una herramienta básica para la resolución de complejos problemas matemáticos en diferentes áreas como la computación, el cálculo numérico, prototipaje algorítmico y teoría de control automático, y por último en estadística (Cilento, 2007)

### **LabWindows/CVI**

El software LabWindows/CVI (en inglés, CVI: acrónimo de C for Virtual Instrumentation) es un entorno de desarrollo integrado para el lenguaje C enfocado en el desarrollo de programas de prueba, medida y control gracias a la instrumentación virtual (National Instruments, 2016).

Este es un producto de National Instruments muy similar a LabVIEW, con lo cual comparte las funcionalidades y permite concebir aplicaciones de adquisición de datos, así como de las interfaces gráficas permitiendo almacenar y analizar estos datos (Jolly, 2016).

Este entorno de desarrollo integrado funciona bajo Windows y se lleva también bajo Unix. Las bibliotecas especializadas en prueba, medida y control de National Instruments son también accesibles para la programación en C++, C# o Visual Basic gracias a Measurement Studio.

### **RTLinux**

El software RTLinux es un sistema operativo de tiempo real que ejecuta Linux como un thread (hilo de ejecución) de menos prioridad que las tareas de tiempo real. Con este diseño, las tareas de tiempo real y los manejadores de interrupciones nunca se ven retrasados por operaciones que no son de tiempo real (Ripoll, 2001).

RTLinux proporciona la capacidad de ejecutar tareas de tiempos reales y manejar las interrupciones en la misma máquina que el Linux estándar. Estas tareas y los manejadores se ejecutan cuando se necesitan en detrimento de lo que se estuviera ejecutando Linux. El peor caso de tiempo es entre que se detecta la interrupción hardware y el procesador ejecuta la primera instrucción del manejador de la interrupción. Este tiempo es del orden de los 10 microsegundos en la plataforma x86.

## **1.2 Simulación**

Se podría considerar que la simulación nace en 1777 con el planteamiento del problema "La aguja de Buffon", un método matemático sencillo para ir aproximando el valor del número  $\pi$  a partir de sucesivos intentos (Lander Simulation & Training Solutions, 2015). La simulación según (Shannon, 1998) es el proceso de diseñar un modelo de un sistema real y llevar a término experiencias con él, con la finalidad de comprender el comportamiento del sistema o evaluar nuevas estrategias dentro de los límites impuestos por un cierto criterio o un conjunto de ellos para el funcionamiento del sistema.

En la actualidad la simulación es realizada por diferentes métodos, dentro de los más utilizados podemos encontrar la simulación por computadora. Este es un intento de modelación de situaciones de la vida real por medio de un programa informático, lo que requiere ser estudiado para ver cómo es que trabaja el sistema (Tarifa).

Gracias al avance tecnológico el uso de la simulación por computadoras es usada en el entrenamiento o preparación de las personas para la correcta manipulación de equipos o para la optimización de procesos industriales u otras actividades de la vida cotidiana. Su importancia está en que permite

cometer errores durante el entrenamiento y aporta valiosas lecciones a los usuarios en un ambiente seguro.

En áreas como la educación, la simulación por computadora cumple un rol muy importante ya que se usa para evaluar el aprendizaje, resolver problemas de habilidades y disposición de los estudiantes, y el servicio de los profesores.

### **1.2.1 Entrenador**

El entrenamiento es la acción que se refiere a la adquisición de conocimientos, habilidades, y capacidades como resultado de la enseñanza de habilidades o prácticas y conocimiento relacionado con aptitudes que encierran cierta utilidad. Este forma el centro del aprendizaje y proporciona la base de los contenidos en las personas que estén desarrollando esta actividad.

El entrenamiento de pilotos en los inicios de la aviación y en la actualidad demuestra la relación que existe entre el entrenamiento y la simulación. A falta de ordenadores y programas informáticos de simulación, los primeros pilotos se entrenaban con primitivos simuladores físicos. Actualmente, todos los pilotos están obligados a entrenarse periódicamente en sofisticados simuladores para estar preparados para resolver cualquier problema que pueda aparecer en el vuelo (Montero Avendaño, 2011).

### **1.2.2 Simulador de procesos**

Los simuladores de procesos según (Scenna, 1999) son una herramienta de software que agrupan un conjunto de funcionalidades tales como, bases de datos de propiedades físicas, propiedades termodinámicas, métodos numéricos, operaciones unitarias y modelos de cinéticas químicas integradas de forma tal que reproduzca el comportamiento de un proceso o sistema físico determinado. Estos son utilizados en la simulación de sistemas naturales, en procesos químicos y en otros procesos industriales con el objetivo de mejorar e incrementar la eficiencia de las mismas, estas permiten realizar simulaciones de diferentes procesos antes de que ocurran en realidad, las cuales producen resultados que pueden ser analizados para una futura realización de los mismos.

La simulación es una herramienta que se ha hecho indispensable para la



solución adecuada de los problemas de procesos, esta permite analizar, diseñar y optimizar procesos de interés. Los simuladores de procesos constituyen una poderosa herramienta para la formación y preparación de las personas, permiten el entrenamiento o preparación de las mismas para la correcta manipulación de equipos o para la optimización de procesos industriales en un ambiente próximo a la realidad de manera controlada y segura. El empleo de estos en el proceso de la enseñanza y aprendizaje ha llevado a un número importante de autores a reconsiderar las prácticas de laboratorio. De esta manera se aprovecha el potencial de simulaciones, visualizaciones y otros recursos didácticos digitales, para favorecer el aprendizaje y optimizar el tiempo presencial disponible en el laboratorio (Romero Ariza & Quesada Armenteros, 2014).

Dentro de los simuladores más usados encontramos hardware de simuladores, destacándose IPA Virtual, LabVolt Modul A6 y PLC Sim-S7, hardware con soluciones híbridas y emuladores de procesos con microcontroladores. Otros de estos simuladores son los software de simulación por computadora (Zurita Landaeta, 2014).

Existen diversas configuraciones para la simulación, algunas de ellas utilizan un Controlador Lógico Programable (PLC), microcontroladores, circuitos electrónicos analógicos o digitales que interactúan con programas de simulación en el control de proceso, e incluso se utilizan computadoras que sirven para simular plantas industriales comunicándose con otras computadoras que sirven de elemento de control.

Los simuladores tienen como característica en común la representación virtual de situaciones reales a través de un modelo que constituye una abstracción de la realidad; con el objetivo de comprender el comportamiento del sistema y evaluar de esta manera diferentes tipos de estrategias para su mejor operación, ya sea por medio de elementos matemáticos y tecnológicos.

Según (López & Juan) existen dos tipos de simuladores. Estos son los simuladores con modelos previamente programados y los simuladores programables. A continuación se explica cada uno de estos simuladores.

### **Simuladores con modelos previamente programados**

Los simuladores con modelos previamente programados son aquellos en donde el usuario utiliza paquetes de software de uso específico, los elaborados para una operación unitaria específica y un determinado rango de operación o de uso general que contienen en su estructura varias operaciones unitarias las cuales pueden ser interrelacionados entre sí para simular un proceso que ya incluyen la programación del modelo. Estos simuladores son muy confiables y robustos, donde el usuario no tiene porqué ser un especialista en simulación, ejemplo de esto son Aspen Hysys, Aspen Plus, Chemcad, ProModel (EcuRed, 2016).

### **Simuladores programables**

Los simuladores programables son aquellos en donde el usuario programa sus propios modelos mediante el uso paquetes de software de cálculo matemático. Estos permiten al usuario aplicar sus propios modelos e interactuar de una forma más profunda con los mismos, disponiendo de una mayor libertad para simular, aunque el usuario debe tener un mayor conocimiento de los principios científicos como de manejo de lenguajes de programación. Entre los más utilizados se destacan Lenguaje C, Visual Basic, LabVIEW (Zurita Landaeta, 2014).

## **1.3 Sistemas de control**

En teoría de control y en general en instrumentación industrial se conoce como lazo de control o "Control Loop" a un sistema que mantiene una relación prescrita entre la salida y la entrada de referencia, comparándolas y usando la diferencia como medio de control. Este presenta un conjunto de componentes, tales como el elemento sensor, transductor de señal, receptor de señal, comparador de punto de ajuste, mecanismo de control y el elemento final de control. Los componentes del lazo de control están configurados en forma de circuito de tal manera que la señal de control es transmitida al elemento final de control para ajustar el proceso a controlar a un punto de consigna dependiendo de la magnitud del estímulo generado en el proceso.

### **1.3.1 Clasificación general de sistemas de control**

Los sistemas de control de acuerdo a su comportamiento se pueden clasificar en sistemas de control en lazo cerrado y sistemas de control en lazo abierto (Ogata).

**Sistemas de control en lazo cerrado:** En un sistema de control en lazo cerrado, se alimenta al controlador la señal de error de actuación, que es la diferencia entre la señal de entrada y la señal de realimentación, a fin de reducir el error y llevar la salida del sistema a un valor conveniente. El término control en lazo cerrado siempre implica el uso de una acción de control realimentado para reducir el error del sistema.

**Sistemas de control en lazo abierto:** Los sistemas en los cuales la salida no afecta la acción de control se denominan sistemas de control en lazo abierto.

En cualquier sistema de control en lazo abierto, la salida no se compara con la entrada de referencia. A cada entrada de referencia le corresponde una condición operativa fija, la precisión del sistema depende de la calibración. Ante la presencia de perturbaciones, un sistema de control en lazo abierto no realiza la tarea deseada. En la práctica, el control en lazo abierto sólo se usa si se conoce la relación entre la entrada y la salida y si no hay perturbaciones internas ni externas. Es evidente que estos sistemas no son de control realimentado.

### **1.3.2 Controlador PID**

El controlador es un componente fundamental dentro de los lazos de control, este compara el valor efectivo de salida de una planta con el valor deseado, determina la desviación y produce una señal de control que reduce la desviación a cero o a un valor pequeño (B & Tabares G, 1997).

Dentro de los controladores más usados encontramos los controles de dos posiciones (todo-nada, si-no, on-off), controles proporcionales (P), controles integrales, controles proporcionales e integrales (PI), controles proporcionales y derivativos (PD), y los controles proporcionales, integrales y derivativos (PID) (Ogata). Estos tienen como característica en común la toma de decisiones para mantener el punto de control ante los efectos de las perturbaciones.

Por tener una exactitud mayor a los controladores proporcional, proporcional derivativo y proporcional integral, el controlador PID se utiliza en aplicaciones más cruciales tales como control de presión, flujo, fuerza, velocidad, en muchas aplicaciones química, en reguladores de velocidad de automóviles y el control de ozono residual en tanques de contacto. El algoritmo de cálculo del control PID se da en tres parámetros distintos: el proporcional, el integral, y el derivativo; este puede representarse en varias formas y se suele modificar para obtener mejores prestaciones. Una valoración de las mismas se realiza a continuación (Gil Nabojas & Díaz Cordovés, 2009):

**Forma estándar o no interactiva:** La forma estándar algunas veces es llamada “algoritmo ISA”, o “algoritmo ideal”. Las acciones proporcional, integral y derivativa son no interactivas en el dominio del tiempo. La ecuación 1.1 representa la ecuación de este controlador:

$$G(s) = K \left( 1 + \frac{1}{sT_i} + sT_d \right) \quad (1.1)$$

Dónde:

$K$ : Ganancia proporcional

$T_i$ : Tiempo integral

$T_d$ : Tiempo derivativo

**Forma serie o interactiva:** La forma serie es también llamada forma clásica. El nombre “clásico” refleja este hecho. La forma serie tiene una interpretación atractiva en el dominio de la frecuencia, ya que los ceros corresponden a los valores inversos de los tiempos derivativo e integral. Todos los ceros del controlador son reales. A continuación en la ecuación 1.2 se observa la forma de este controlador.

$$G'(s) = K' \left( 1 + \frac{1}{sT'_i} \right) (1 + sT'_d) \quad (1.2)$$

Dónde:

$K'$ : Ganancia proporcional

$T'_i$ : Tiempo integral

$T'_d$ : Tiempo derivativo

La relación de los parámetros del controlador de la forma serie respecto al controlador estándar se presenta a continuación:

$$K = K' \frac{T'_i + T'_d}{T'_i} \quad (1.3)$$

$$T_i = T'_i + T'_d \quad (1.4)$$

$$T_d = \frac{T'_i T'_d}{T'_i + T'_d} \quad (1.5)$$

**Forma paralela:** La forma paralela es la más general, debido a que se pueden obtener acciones proporcional, integral y derivativa puras con parámetros finitos. El controlador puede también tener ceros complejos, siendo, por tanto, la forma más flexible. Sin embargo, es también la forma donde los parámetros tienen poca interpretación física. A continuación en la ecuación 1.6 se observa la forma de este controlador.

$$G(s) = k + \frac{k_i}{s} + sk_d \quad (1.6)$$

Dónde:

$k$ : Ganancia proporcional

$k_i$ : Ganancia integral

$k_d$ : Ganancia derivativa

La relación de los parámetros del controlador de la forma paralela respecto al controlador estándar se presenta a continuación:

$$k = K \quad (1.7)$$

$$k_i = \frac{K}{T_i} \quad (1.8)$$

$$k_d = KT_d \quad (1.9)$$

### Modificación PI-D

Según (Ogata) se denomina reacción del punto de ajuste a la función de pulso aguda que contiene la variable manipulada  $U(s)$  cuando la entrada de referencia es una función escalón, y en un controlador PID real, en lugar del término derivativo puro  $T_d S$  se emplea:  $\frac{T_d}{1+\gamma T_d S}$

Dónde:

$\gamma$ : Valor práctico que está alrededor de 0.1

Para evitar el fenómeno de la reacción del punto de ajuste se opera la acción derivativa sólo en la trayectoria de realimentación, a fin de que la diferenciación ocurra únicamente en la señal de realimentación y no en la señal de referencia. El esquema de control ordenado de esta forma se representa en la figura 1.1, esta forma es denominada control PI-D.

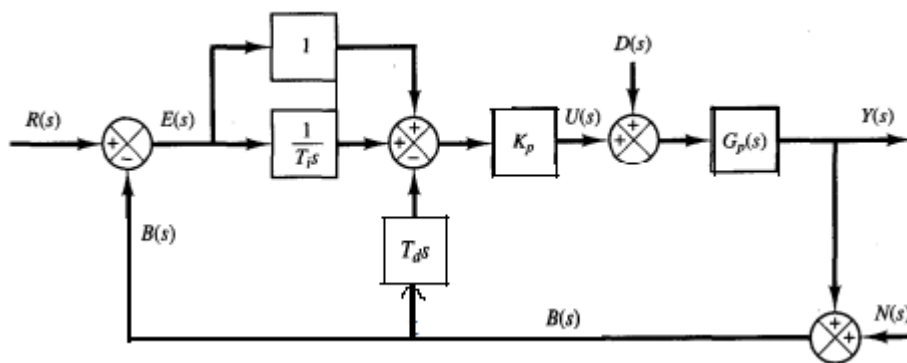


Figura 1.1: Control PI-D

Dónde:

$R(s)$ : Entrada de referencia

$U(s)$ : Señal manipulada

$D(s)$ : Perturbación

$Y(s)$ : Salida

$N(s)$ : Ruido

$B(s)$ : Señal observada

### Modificación I-PD

Si se considera el caso en el que la entrada de referencia es una función escalón. Tanto el control PID como el control PI-D implican una función escalón en la señal manipulada. Según (Ogata) en muchas ocasiones, tal cambio escalón en la señal manipulada tal vez no sea conveniente. Para solucionar esto se puede mover la acción proporcional y la acción derivativa a la trayectoria de realimentación, a fin de que estas acciones sólo afecten la señal de realimentación. El esquema de este control se muestra en la figura 1.2, este es denominado control I-PD.

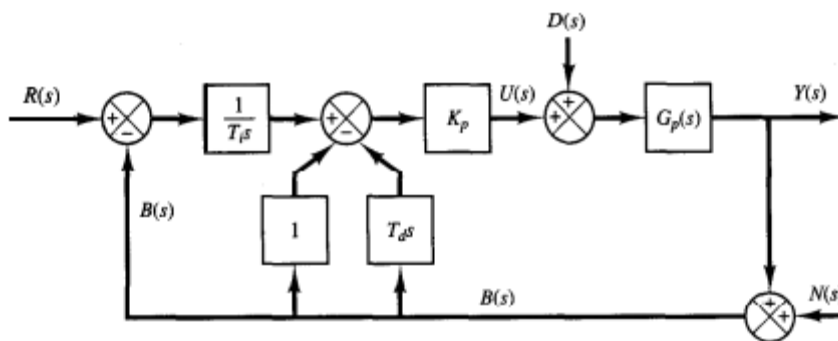


Figura 1.2: Control I-PD

### 1.4 Consideraciones prácticas en la implementación de algoritmos de control

Cuando se aplica un regulador PID a un proceso industrial existen varios puntos importantes que deben ser considerados ya que podrían proporcionar malos resultados y funcionar de manera insatisfactoria. Según (Ballesteros Horta & Hernández Maldonado, 2009) algunas de estas son:

- Transferencia suave entre los modos manual y automático: Cuando el sistema está en modo manual, el algoritmo de control produce una señal de mando que puede ser diferente de la señal de control manualmente generada. Para esto es necesario que las dos salidas coincidan al tiempo de conmutar. De esta manera se asegura que el estado

asociado con el control manual se actualice convenientemente cuando el regulador está operando en modo automático y viceversa.

- Filtrado de la señal de mando: Esta permite el filtraje de la variable de control con el objetivo de que esta no alcance los límites del actuador.
- Selección del periodo de muestreo: El periodo de muestreo es el instante de tiempo en que se realiza la simulación.
- Convertidor D/A
- Actuador: El actuador es la parte final del dispositivo de control que produce un cambio físico en el elemento final de control.
- Retardo de transporte: El efecto del retardo de transporte ocasiona retardos de tiempo en el proceso, a este fenómeno se le conoce comúnmente como tiempo muerto, (Ogata).
- Variación de la carga: Es una variación en porciento de la señal de mando.
- Efecto anti-windup: Es la acción que se realiza para contrarrestar el efecto de la variable de control cuando alcanza los límites prefijados del actuador. Básicamente existen tres esquemas para evitar o reducir al menor tiempo la saturación del término integral: métodos de integración condicional, método de seguimiento integral y el método de limitación de la entrada.
- Saturación en el mando.
- Zona muerta: Según (Ogata) la zona muerta de un componente es un rango pequeño de variaciones de entrada ante las cuales el componente es insensible.
- Unidades ingenieriles.
- Niveles de ruido: Los niveles de ruido son señales no deseadas que provocan inestabilidad en el sistema.

### **1.5 Antecedentes de entrenadores de procesos**

Se seleccionaron para el análisis tres aplicaciones:

1. Diseño de un simulador digital de procesos.



2. Diseño y montaje de un entrenador con el controlador lógico programable PLC S7-200 de SIEMENS.
3. Control de un tanque de presión mediante el módulo PID de un autómatas programable.

### **Diseño de un simulador digital de procesos**

En este trabajo el autor (Zurita Landaeta, 2014) desarrolla un simulador digital de procesos, basado en el PIC18F4550, como simulador de la planta y una interface en LabVIEW como controlador. Este simulador posibilita el entrenamiento de control discreto en sistemas de primer orden hasta cuarto orden. En el mismo los modelos son simulados en el microcontrolador, este brinda la capacidad de adaptación y expansión que permiten la creación de entornos constructivistas de aprendizaje en estas áreas del control de procesos. Las estrategias de control, parametrización y ajuste del controlador en sus opciones proporcional, PI y PID, de forma manual, son desarrolladas en LabVIEW.

### **Diseño y montaje de un entrenador con el controlador lógico programable PLC S7-200 de SIEMENS**

El entrenador que se construye en este trabajo es un módulo para configurar, programar y usar un autómatas SIEMENS. El sistema consiste en un simulador de entradas, luces pilotos; según el autor (Perez Rojas, 2008) estos dispositivos se usan para que el PLC “testee y actúe” según su programación.

El sistema de supervisión y control se realiza desde el autómatas, conjuntamente se puede realizar la monitorización en una PC. Al implementar este entrenador, el usuario tiene la posibilidad de familiarizarse de forma práctica con el software y el lenguaje de programación que se implementa para este autómatas en el área de la automatización industrial.

### **Control de un tanque de presión mediante el módulo PID de un autómatas programable**

En este proyecto el autor (Nuñez Valle, 2011) diseña un circuito neumático en el que se controla la presión de un depósito utilizando la función PID FB41

(CONT\_C) integrada en el software de programación STEP7 de SIEMENS, y se desarrolla un sistema SCADA que controla la presión del depósito neumático mediante el autómatas S7300 de Siemens.

Para esto se diseña y ensambla un circuito neumático con un acumulador neumático y los elementos necesarios para los requerimientos de control. Se utiliza una electroválvula proporcional de 5 vías y 3 posiciones que permita pasar una cierta cantidad regulable de aire hacia el acumulador, mantener el aire, o por el contrario dejar escapar, en mayor o menor medida, el aire acumulado en el depósito. Para conocer en todo momento la presión acumulada precisan de un transductor de presión-voltaje y se dispone de un manómetro para conocer de manera analógica la presión alcanzada.

Con la configuración y programación de la función PID (FB 41 CONT\_C) que lleva integrado el PLC SIMATIC S7-300, se diseña un regulador PID como elemento de control, y se crea el sistema SCADA para el control de presión del depósito neumático, de esta manera el usuario controla y supervisa la presión del acumulador neumático sin tener idea alguna de programar el autómatas SIEMENS.

### **Análisis de los casos seleccionados**

En los últimos casos se tiene como inconveniente el uso de PLC para el diseño de los simuladores. Estos son dispositivos caros que no están al alcance de muchas personas. En el último caso se utiliza un acumulador neumático y una electroválvula proporcional, por lo que también son inconvenientes a tener en cuenta. En el primer caso se utiliza un microcontrolador para la simulación. Esto sería innecesario porque este proceso puede ser simulado en un software destinado para este fin.

### **1.6 Consideraciones finales del capítulo**

El uso de los entrenadores de sistemas de control es una herramienta de aprendizaje viable para cualquier nivel de enseñanza. Mediante la simulación se producen situaciones de la vida cotidiana y ante determinado ejemplo el usuario que se entrena puede aprender valiosas lecciones y ejercitarse en un ambiente virtual seguro.

## **Capítulo II: Diseño del entrenador digital de lazo de control**

Para la implementación del entrenador de lazo de control se utiliza el software NI LabVIEW 2014. Este destaca por su facilidad de uso, rapidez de programación y el módulo PID de la función control y simulación. En este capítulo se muestra el diseño de los elementos del lazo de control mediante el uso del lenguaje gráfico de este software. Posteriormente se seleccionan las consideraciones prácticas que se implementarán en el entrenador de lazo de control y finalmente se exponen las consideraciones finales del capítulo.

### **2.1 Arquitectura del entrenador**

El entrenador de lazo de control está compuesto por dos etapas, esta estructura se representa en la figura 2.1. La primera está representada por la configuración de los parámetros de simulación. En esta etapa se seleccionan los valores de las distintas variables del proceso y las consideraciones prácticas que se utilizarán en la simulación. En la segunda y última etapa se simula el proceso seleccionado para los valores de configuración de los elementos que componen el lazo de control.



Figura 2.1: Arquitectura del entrenador de lazo de control

### **2.2 Sistema de control**

El sistema de control que se implementa en el entrenador de lazo de control es un sistema de control en lazo cerrado, el mismo es mostrado en la figura 2.2.

Los elementos que conforman el lazo son un controlador PID, un filtro paso bajo, un convertidor digital-analógico (D/A), un actuador y la planta que se desea controlar.

El lazo de control contiene un controlador PID que se alimenta de la señal de error entre la señal de entrada y la señal de realimentación. La salida de este controlador es filtrada por el filtro paso bajo antes de entrar al convertidor D/A de  $n$  bit configurable. Este convierte la señal de mando que llega al actuador de característica lineal. El actuador produce un cambio físico en el elemento final de control y la señal de salida de este actúa en la planta que se desea controlar.

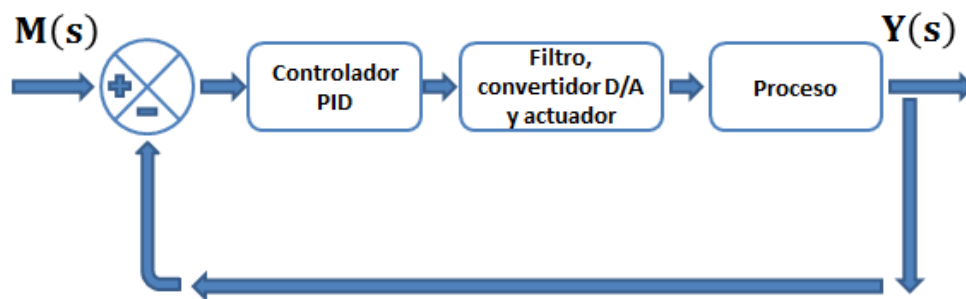


Figura 2.2: Lazo de control del entrenador

$Y(s)$ : Salida del proceso.

$M(s)$ : Entrada del proceso.

### 2.2.1 Controlador PID

El controlador PID que se selecciona para el entrenador de lazo de control es el controlador PID avanzado (PID Advanced) del software LabVIEW. Este es un VI (Instrumentos Virtuales) propio del software, el mismo posee 13 parámetros de entrada y 2 de salida, las mismas son mostradas en la figura 2.3.

Entradas:

- Alfa (alpha)
- Gamma (gamma)
- Beta (beta)
- Control manual (manual control): modo en que se desea controlar el sistema, en forma manual o forma automática

- Rango en la salida (output range): límite en que se desea tener la salida del controlador
- Setpoint: valor de referencia para la estabilidad
- Ganancias del PID (PID gains): parámetros del controlador PID
- Periodo de muestreo:  $dt$  (s)
- Linealidad (linearity)

Salidas:

- Salida del controlador (output)
- Periodo de muestreo:  $dt$  (s)

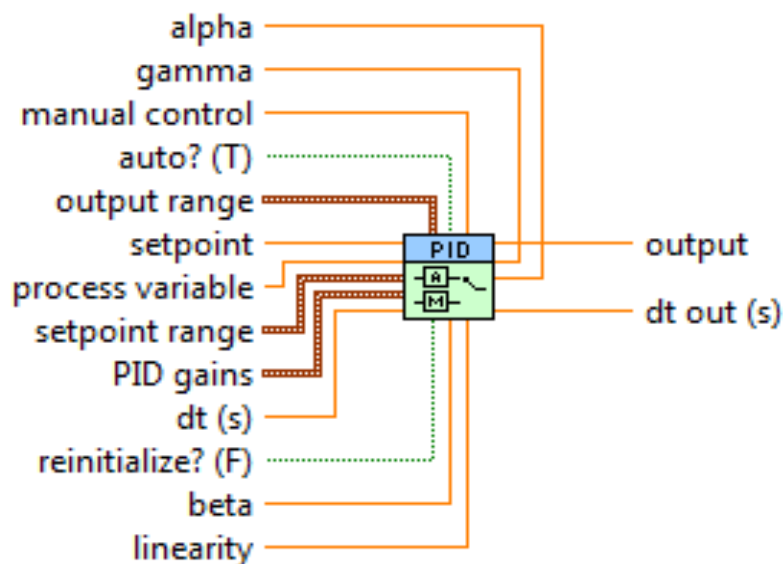


Figura 2.3: PID Advanced del software LabVIEW

En el controlador PID avanzado se destaca la transición suave entre los modos manual y automático. Cuando el sistema está en modo manual, el algoritmo de control produce una señal de control que puede ser diferente de la señal de control manualmente generada. Por esta razón es preciso que se asegure que el estado asociado con el control manual se actualice convenientemente cuando el regulador está operando en modo automático y viceversa.

En el entrenador de lazo de control el controlador PID se representa de diferentes formas. Estas son la forma estándar o no interactiva, forma serie o interactiva y la forma paralela. Para el cálculo de sus parámetros se utiliza el PID de conversión de estructuras (PID Structure Conversion) del software

LabVIEW. Este es un VI propio de software y posee siete entradas y una salida. En la figura 2.4 se muestra las entradas y salidas de este VI.

Entradas:

- Unidad proporcional (proportional unit): recibe la forma en que se desea. la unidad proporcional, en banda proporcional o ganancia.
- Estructura del PID (PID structure).
- Proporcional (proportional): valor del parámetro proporcional.
- Integral (integral): valor del parámetro integral.
- Derivativo (derivative): valor del parámetro derivativo.
- Unidad del parámetro integral (integral unit): valor en minutos o en segundos.
- Unidad del parámetro derivativo (derivative unit): valor en minutos o en segundos.

Salida:

- Ganancias del PID (PID gains): valores de los parámetros del controlador PID seleccionado.

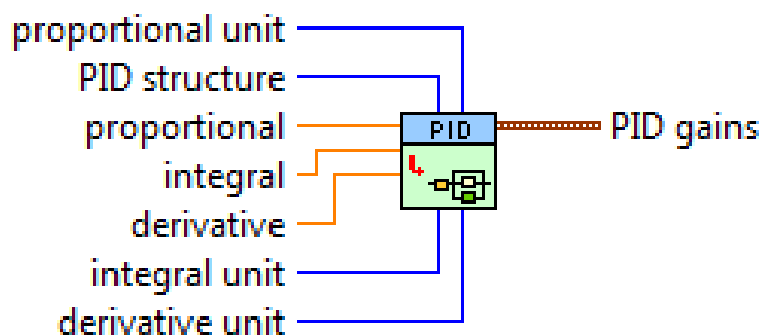


Figura 2.4: PID Structure Conversion del software LabVIEW

### 2.2.2 Filtro paso bajo

El filtro paso bajo se diseña con el objetivo de que la variable de control no alcance los límites del actuador. Para el diseño de este filtro se desarrolla mediante el lenguaje gráfico del software LabVIEW la ecuación del filtro. La ecuación del filtro se define en la ecuación 2.1 y en la figura 2.5 se muestra la misma.

Ecuación del filtro paso bajo del entrenador:

$$F_K = F_{K-1} + a * (C - F_{K-1}) \quad (2.1)$$

Dónde:

$F_K$ : Salida del filtro

$F_{K-1}$ : Valor anterior de la salida del filtro

$a$ : Constante del filtro,  $0 < a \leq 1$ , para  $a = 1$  no se filtra

$C$ : Señal de salida del controlador

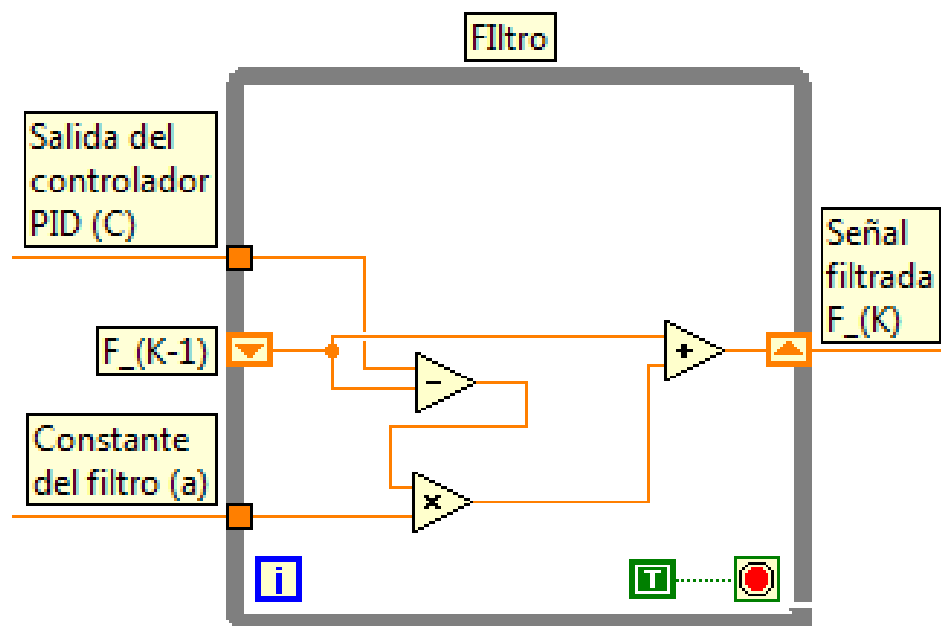


Figura 2.5: Ecuación del filtro desarrollada en LabVIEW

### 2.2.3 Convertidor digital-analógico

El convertidor D/A presenta característica lineal y la ecuación de la recta está dada por la ecuación 2.2. El desarrollo de esta ecuación en LabVIEW se muestra en la figura 2.6.

Ecuación de la recta del convertidor D/A:

$$Conv = m * F_K + n \quad (2.2)$$

Dónde:

$$m = \frac{Y_{MAX} - Y_{MIN}}{X_{MAX} - X_{MIN}}$$

$$n = Y_{MIN} - m * X_{MIN}$$

$m$ : Pendiente de la recta

$n$ : Intercepto de la recta

$F_K$ : Señal que sale del filtro

$Conv$ : Salida del convertidor D/A

$Y$ : Valor de salida del convertidor (voltaje)

$X$ : Valor de entrada del convertidor (valor digital)

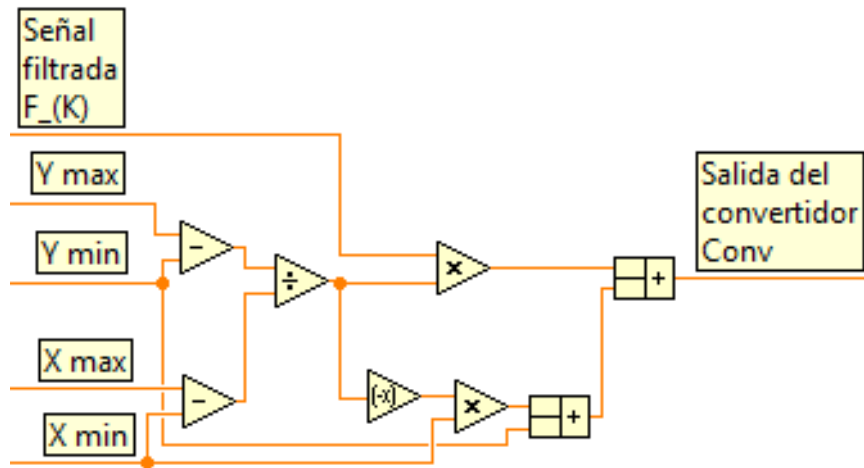


Figura 2.6: Ecuación del convertidor D/A en LabVIEW

#### 2.2.4 Actuador

El actuador del entrenador de lazo de control presenta característica lineal y la ecuación de la recta está dada por la ecuación 2.3. La representación de esta ecuación en el software LabVIEW se muestra en la figura 2.7.

Ecuación de la recta del actuador:

$$Act = m * Conv + n \quad (2.3)$$

Dónde:

$$m = \frac{Y_{MAX} - Y_{MIN}}{X_{MAX} - X_{MIN}}$$

$$n = Y_{MIN} - m * X_{MIN}$$



$m$ : Pendiente de la recta

$n$ : Intercepto de la recta

$Conv$ : Señal que sale del convertidor D/A

$Act$ : Salida del actuador

$Y$ : Valor de salida del actuador (voltaje)

$X$ : Valor de entrada del actuador (voltaje)

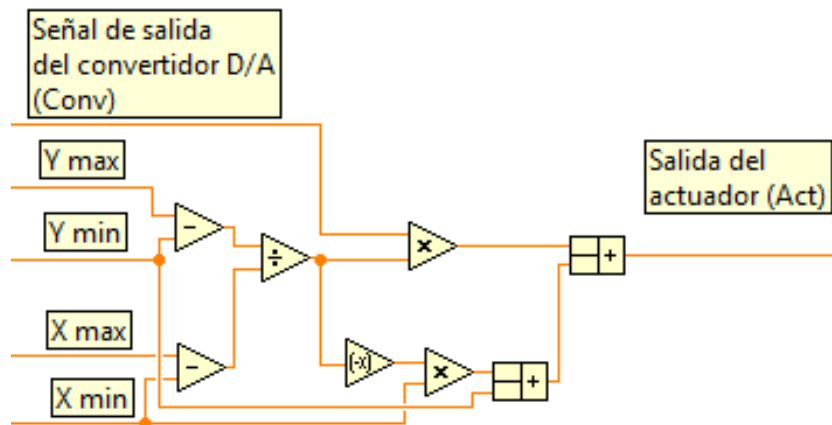


Figura 2.7: Ecuación del actuador en LabVIEW

### 2.2.5 Proceso

Para el diseño del sistema de control se establece el uso de ecuaciones en el dominio de la frecuencia para la representación de la función de transferencia del proceso. La función de transferencia en forma general se define en la ecuación 2.4. Según los valores de las constantes esta ecuación puede convertirse en un sistema de primer o segundo orden.

$$Y(s) = K * \frac{A_2 s^2 + A_1 s + A_0}{B_2 s^2 + B_1 s + B_0} * M(s) \quad (2.4)$$

Dónde:

$A_2, A_1, A_0, B_2, B_1, B_0$ : Coeficientes de la función de transferencia

$K$ : Ganancia de la función de transferencia

$Y(s)$ : Salida del proceso

$M(s)$ : Entrada del proceso

Se utiliza el método de aproximación de derivadas por diferencias finitas hacia atrás para el desarrollo de la ecuación 2.4 en el software LabVIEW. A continuación se explica el procedimiento.

De la ecuación 2.4 se multiplica y se organiza la ecuación de la forma siguiente:

$$Y(s) * (B_2S^2 + B_1S + B_0) = M(s) * K * (A_2S^2 + A_1S + A_0) \quad (2.5)$$

Desarrollamos el miembro izquierdo de la ecuación 2.5:

$$Y(s) * (B_2S^2 + B_1S + B_0) = B_2S^2Y(s) + B_1SY(s) + B_0Y(s) \quad (2.6)$$

Se desarrolla el miembro derecho de la ecuación 2.5:

$$M(s) * K * (A_2S^2 + A_1S + A_0) = KA_2S^2M(s) + KA_1SM(s) + KA_0M(s) \quad (2.7)$$

Se discretiza cada término de las ecuaciones 2.6 y 2.7, por el método de diferencia finita hacia atrás se obtiene:

$$B_0Y(s) = B_0Y_K$$

$$B_1SY(s) = B_1 \frac{dY}{dt} = B_1 \frac{Y_K - Y_{K-1}}{dt}$$

$$B_2S^2Y(s) = B_2 \left( \frac{dY}{dt} \right)^2 = B_2 \frac{Y_K - 2Y_{K-1} + Y_{K-2}}{(dt)^2}$$

$$KA_0M(s) = KA_0M_K$$

$$KA_1SM(s) = KA_1 \frac{dM}{dt} = KA_1 \frac{M_K - M_{K-1}}{dt}$$

$$KA_2S^2M(s) = KA_2 \left( \frac{dM}{dt} \right)^2 = KA_2 \frac{M_K - 2M_{K-1} + M_{K-2}}{(dt)^2}$$

Dónde:

$Y_K$ : Salida del proceso

$Y_{K-1}$ : Valor anterior de la salida del proceso

$Y_{K-2}$ : Valor anterior de la salida anterior del proceso

$M_K$ : Entrada del proceso

$M_{K-1}$ : Valor anterior de la entrada del proceso

$M_{K-2}$ : Valor anterior de la entrada anterior del proceso

$dt$ : Periodo de muestreo

Se sustituye cada parámetro discretizado en las ecuaciones 2.6 y 2.7, de esta manera se igualan el miembro derecho e izquierdo de la ecuación y se obtiene:

$$B_0 Y_K + B_1 \frac{Y_K - Y_{K-1}}{dt} + B_2 \frac{Y_K - 2Y_{K-1} + Y_{K-2}}{(dt)^2} = KA_0 M_K + KA_1 \frac{M_K - M_{K-1}}{dt} + KA_2 \frac{M_K - 2M_{K-1} + M_{K-2}}{(dt)^2}$$

Ahora se despeja la salida del proceso  $Y_K$ :

$$Y_K = \frac{X + Y_{K-1}(B_1 dt + 2B_2) - B_2 Y_{K-2}}{B_0 dt^2 + B_1 dt + B_2} \quad (2.8)$$

Dónde:

$$X = K[M_K(A_0 dt^2 + A_2 + A_1 dt) - M_{K-1}(2A_2 + A_1 dt) + A_2 M_{K-2}]$$

Una vez obtenida la ecuación 2.8 en instantes de muestreo la ecuación de salida del sistema es desarrolla en el software LabVIEW mediante la herramienta Formula Node. El desarrollo de esta ecuación en LabVIEW se representa en la figura 2.8.

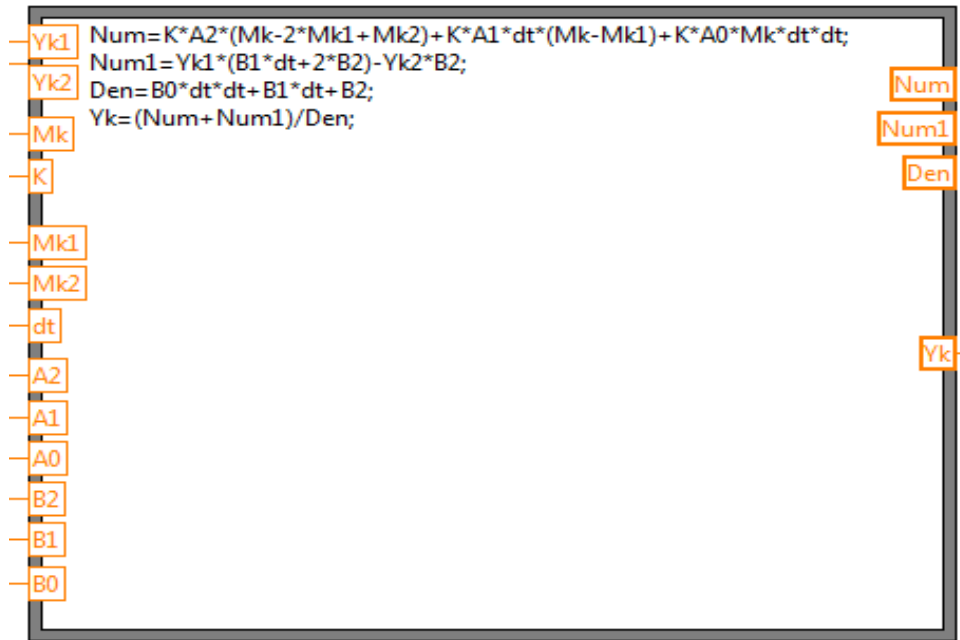


Figura 2.8: Función de transferencia en instantes de muestreo

### **2.3 Consideraciones prácticas.**

Para la implementación del entrenador de lazo de control se seleccionan algunas consideraciones prácticas. El uso de las mismas permite un acercamiento del usuario a situaciones reales de los sistemas de control industrial. El uso de las mismas durante el entrenamiento aporta valiosas enseñanzas en el estudio de estos sistemas, ya que exige del conocimiento y habilidad del usuario para enfrentarse a situaciones dadas. A continuación se presentan las consideraciones prácticas utilizadas en la simulación:

Consideraciones prácticas para el entrenador de lazo de control:

- Transferencia suave entre los modos manual y automático.
- Filtrado de la señal de mando.
- Selección del periodo de muestreo.
- Convertidor D/A y actuador.
- Retardo de transporte.
- Variación de la carga.
- Efecto anti-windup.
- Saturación en el mando.
- Zona muerta.
- Unidades ingenieriles
- Niveles de ruido: Para la creación del mismo se realiza un algoritmo de cálculo entre diferentes variables del sistema y números aleatorios entre cero y uno. Este se introduce en la entrada y en la salida del sistema, ver figura 2.9.

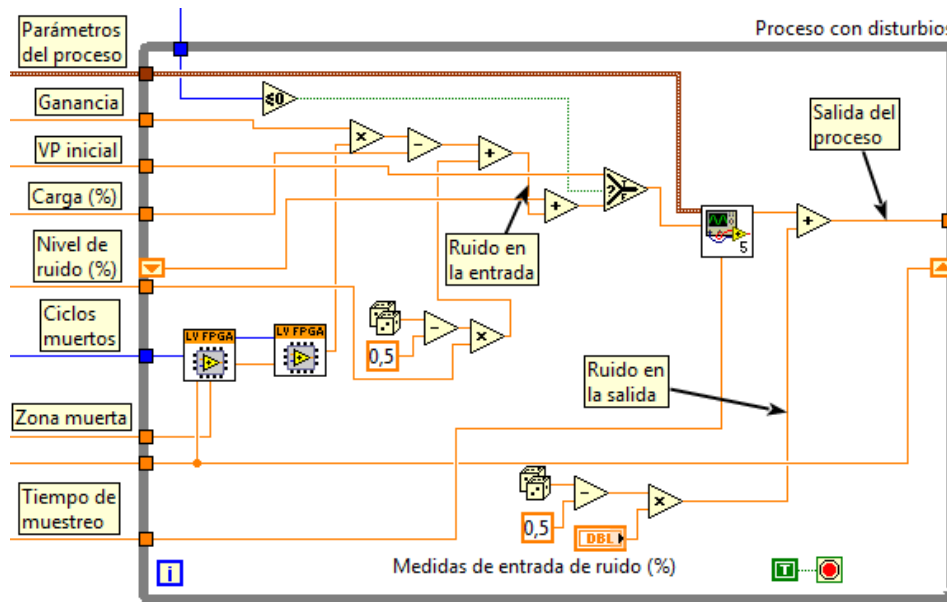


Figura 2.9: Diseño del nivel de ruido en LabVIEW

## 2.4 Interfaz del entrenador de lazo de control

El diseño de la interfaz del entrenador de lazo de control se realiza en un *Tab Control* del software LabVIEW. Este consta de dos pestañas y en cada una de ellas se realiza la configuración de los parámetros de simulación y la simulación del proceso seleccionado respectivamente.

En la figura 2.10 se muestra la etapa de configuración del entrenador de lazo de control. En la parte izquierda de la imagen se muestran todos los parámetros de simulación. En la parte superior se muestran los parámetros del proceso que se desea simular y algunas consideraciones prácticas. En el centro se muestra la estructura de la ecuación general del proceso, la forma del PID seleccionado y los parámetros de este último. En la parte inferior se muestra los datos de configuración del convertidor D/A, el actuador y el filtro paso bajo, además del periodo de muestreo se observa el botón *Stop* que termina con la simulación. En la parte derecha se encuentra un cuadro de diálogo donde se expone una descripción sobre el entrenador de lazo de control con consideraciones prácticas reales.

Configuración

Simulación

Parámetros del proceso

Ganancia K	1,00	A2	0,000
Ciclos muertos	0,00	A1	0,000
Carga %	0,00	A0	238,800
Zona muerta	0,0	B2	0,000
Nivel de ruido %	0,00	B1	0,250
VP inicial	0,00	B0	1,000

Planta

$$Y(s) = k * \frac{a_2 S^2 + a_1 S + a_0}{b_2 S^2 + b_1 S + b_0} * M(s)$$

Estructura PID

Estándar o no interactivo

Proporcional

Integral

Derivativo

2,388

0,001

0,1

$$K_c \left( 1 + \frac{1}{T_i s} + T_d s \right)$$

PID Estándar

Convertidor D/A

Entrada	Salida
-2,048k Min entrada	-10 Min voltaje
2,048k Max salida	10 Max voltaje

Actuador

Entrada	Salida
-10 Min voltaje	-10 Min voltaje
10 Max voltaje	10 Max voltaje

Periodo de muestreo dt (s)

10m

Constante del filtro (a)

1

Stop

Descripción del entrenador de lazo de control

El entrenador de lazo de control está compuesto por dos etapas. La primera está representada por la configuración de los parámetros de simulación. En esta etapa se seleccionan los valores de las distintas variables del proceso y las consideraciones prácticas que se utilizarán en la simulación. En la segunda y última etapa se simula el proceso seleccionado para los valores de configuración de los elementos que componen el lazo de control.

El sistema de control que se implementa en el entrenador de lazo de control es un sistema de control en lazo cerrado. Los elementos que conforman el lazo son un controlador PID, un filtro paso bajo, un convertidor digital-analógico (D/A), un actuador y la planta que se desea controlar.

El lazo de control contiene un controlador PID que se alimenta de la señal de error entre la señal de entrada y la señal de realimentación. La salida de este controlador es filtrada por el filtro paso bajo antes de entrar al convertidor D/A de n bit configurable. Este convierte la señal de mando que llega al actuador de característica lineal. El actuador produce un cambio físico en el elemento final de control y la señal de salida de este actúa en la planta que se desea controlar.

Figura 2.10: Configuración de los parámetros de simulación

En la figura 2.11 se muestra la etapa de simulación del entrenador. En la derecha de la imagen se distinguen tres gráficas, la más grande se corresponde con la respuesta del sistema ante diferentes instantes de tiempo y las más pequeñas a la salida del convertidor D/A y a la señal de error entre la referencia y la señal realimentada. A la izquierda de estas se muestran la señal del setpoint, la variable o salida del proceso y la señal de salida del controlador. En la parte inferior se muestra el interruptor que conmuta entre los modos manual y automático, y viceversa; además se observa el botón Stop que termina con la simulación.

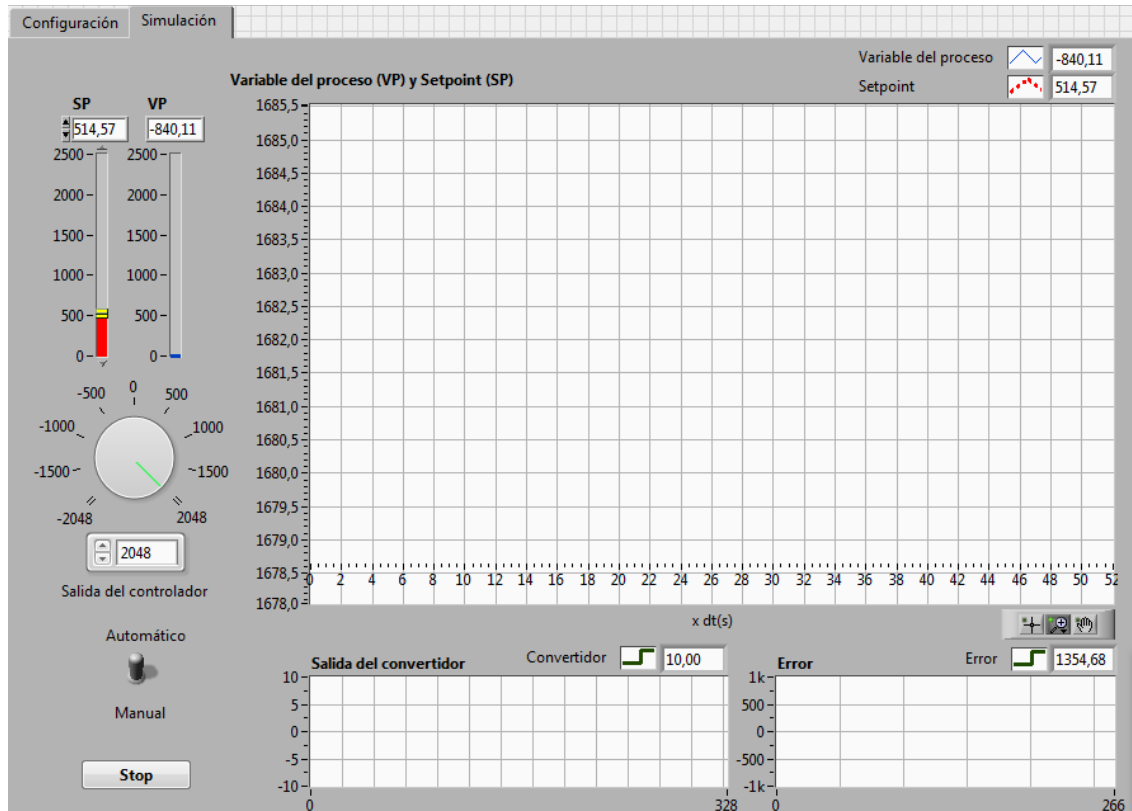


Figura 2.11: Etapa de simulación en el entrenador de lazo de control

## 2.5 Consideraciones finales del capítulo

El software LabVIEW posee características que posibilitan y facilitan el desarrollo de nuevas aplicaciones para el entrenamiento de los estudiantes en el estudio de los sistemas de control. Su facilidad de uso y rapidez de programación permite el desarrollo de innumerables situaciones que se producen en las industrias. El uso de este lenguaje gráfico fundamenta la capacidad de implementación de los entrenadores de lazo de control. Este contribuye a la adquisición de habilidades y conocimientos sin la necesidad del empleo de grandes recursos para la representación de estos sistemas.

## Capítulo III: Análisis de los resultados

En este capítulo se muestra mediante varios ejemplos, los efectos producidos por diferentes consideraciones prácticas sobre dos procesos que se simulan. Los casos seleccionados son el control de velocidad de un motor de corriente directa y el control de posición de un servomotor de corriente directa. Cada caso se simula con los valores de configuración inicial que se seleccionan y posteriormente se introducen varias de estas consideraciones prácticas. Se efectúa una valoración de los efectos provocados y finalmente se exponen las consideraciones finales del capítulo.

### 3.1 Control de velocidad de un motor de corriente directa

El motor de corriente directa (CD) seleccionado proviene de la marca FRACMO. Este es controlado por armadura y utiliza un amplificador con ganancia igual a 25. El amplificador posibilita que la señal de mando necesaria no sea excesivamente grande y el mismo antecede al motor. El modelo del sistema según (Fernández, 2008) está dado por el amplificador y el motor conjuntamente, y su ecuación es la siguiente:

$$\frac{Vel(s)}{Va(s)} = \frac{K}{\tau s + 1} \quad (3.1)$$

Dónde:

$Vel(s)$ : Velocidad del motor

$Va(s)$ : Voltaje de armadura aplicado al motor

$K = 25 \text{ rad} * s^{-1} * V^{-1}$ : Ganancia del sistema

$\tau = 0.25 \text{ s}$ : Constante de tiempo del sistema

#### 3.1.1 Configuración para la simulación

Para que el usuario esté familiarizado con la práctica se trabaja con valores rpm (revolución por minuto). De esta manera la ganancia del sistema sería aproximadamente de  $238.8 \text{ rpm} * V^{-1}$  y la ecuación del sistema quedaría de la siguiente forma:

$$\frac{Vel(s)}{Va(s)} = \frac{238.8}{0.25s + 1} \quad (3.2)$$



En el entrenador se selecciona un convertidor digital-analógico (D/A) bipolar complemento de 2, de 12 *bits* con salida de -10 a 10 V (voltaje). Este convierte la señal de salida del controlador antes de entrar al elemento de acción final. Debido a que el modelo del sistema ya incluye el actuador, en el entrenador de lazo de control se selecciona el mismo con ganancia unitaria. A continuación se muestra la configuración del convertidor D/A y el actuador seleccionado para la simulación.

Convertidor digital – analógico:

- Límite de entrada: Está dado por la siguiente ecuación 3.3.

$$-2^{n-1} a 2^{n-1} \quad (3.3)$$

Dónde:

$n = 12$ : Cantidad de bits

- Límite de salida: -10 a 10 V

Actuador:

- Límite de entrada: -10 a 10 V
- Límite de salida: -10 a 10 V

El periodo de muestreo seleccionado en el entrenador es de 10 ms (milisegundos). En un sistema no oscilatorio este debe ser de 10 a 20 veces inferior al tiempo de subida y en uno oscilatorio de 10 a 20 veces inferior del periodo de oscilación. (Ballesteros Horta & Hernández Maldonado, 2009)

Para la simulación se selecciona el modo automático y una constante de filtro igual a 1, de esta manera no se filtra la señal de salida del controlador. La velocidad máxima del motor sería de 2388 rpm y en sentido contrario -2388 rpm, en esta simulación se desea que la velocidad del motor *Vel* alcance 1500 rpm. Para el control de velocidad del motor se selecciona un PID estándar o no interactivo y de diferentes pruebas realizadas se seleccionan las siguientes constantes:

Controlador PID estándar o no interactivo:

- Ganancia:  $K_c=0.01$

- Tiempo integral:  $T_i=0.001$  s
- Tiempo derivativo:  $T_d=1$  s

La respuesta del sistema ante estos valores de configuración se muestra en la figura 3.1, donde la señal en rojo indica el setpoint y la señal en azul corresponde a la velocidad alcanzada por el motor en diferentes instantes de tiempo hasta que alcanza el valor final o estado estacionario. La respuesta transitoria del sistema presenta una oscilación amortiguada con un máximo sobreimpulso ( $M_p$ ) de aproximadamente 10.4 %. El tiempo de crecimiento ( $t_r$ ) del sistema es de 160 ms y el tiempo de establecimiento ( $t_s$ ) es de aproximadamente 600 ms.

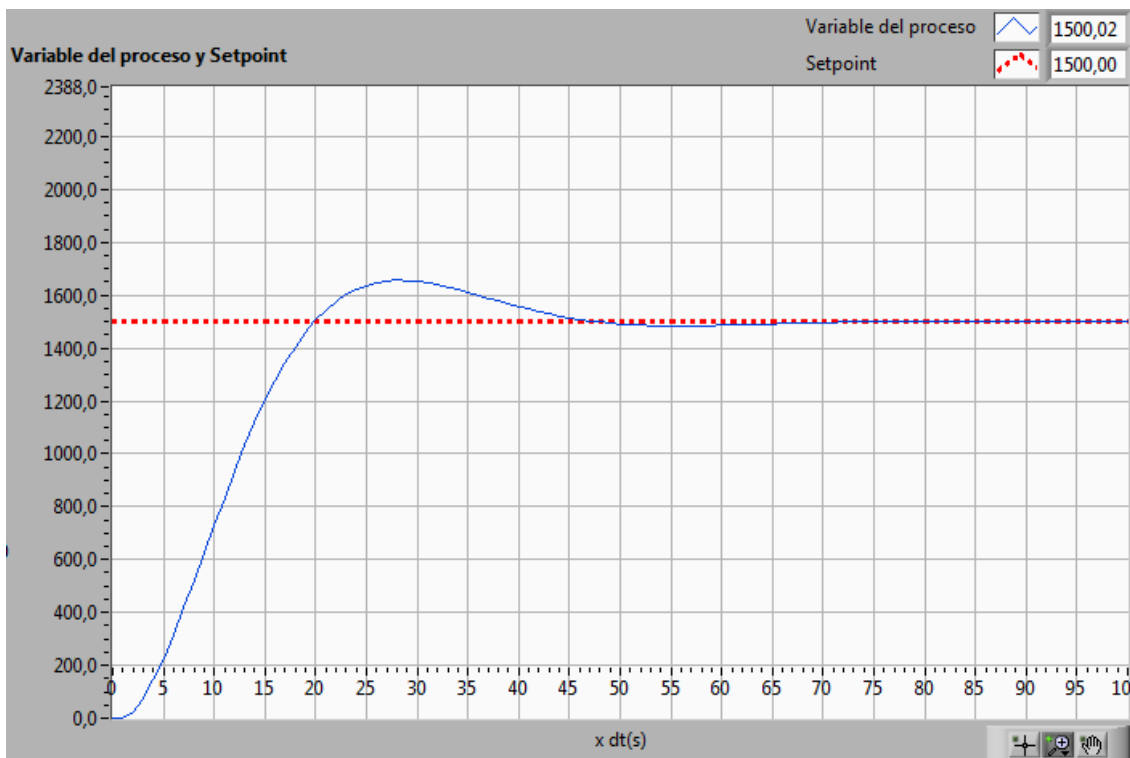


Figura 3.1: Velocidad del motor de corriente directa

### 3.1.2 Consideraciones prácticas para el control de velocidad del motor de corriente directa

El uso de las consideraciones prácticas durante la simulación produce un acercamiento a situaciones reales del control de procesos industrial. Estas producen cambios en la simulación y provocan variaciones en el comportamiento del sistema. De esta manera el usuario enfrenta mediante sus

conocimientos y habilidades las situaciones producidas en un ambiente virtual seguro.

Para este ejemplo se presenta en dos casos diferentes los efectos producidos por diferentes consideraciones prácticas, estas son: variación de la carga y variación del periodo de muestreo. Para esto se utilizan los mismos valores de configuración inicial y se simula el sistema para cada consideración práctica.

### Caso I: Variación de la carga (%)

En la figura 3.2 se observa el efecto producido por la variación en porcentaje de la señal de mando o variación de la carga en la salida. El valor seleccionado es de un 3 % y se mantiene el mismo valor de 1500 *rpm* como entrada de referencia.

La señal de salida del sistema está representada por el color azul y se observa que esta presenta más oscilaciones antes de alcanzar el valor final. El tiempo de crecimiento se mantiene igual que al principio, con un valor de 160 ms, y su máximo sobreimpulso aumenta a 25 %. El sistema presenta un tiempo de establecimiento ligeramente mayor que el de la simulación anterior, y su valor ahora es de 800 ms.

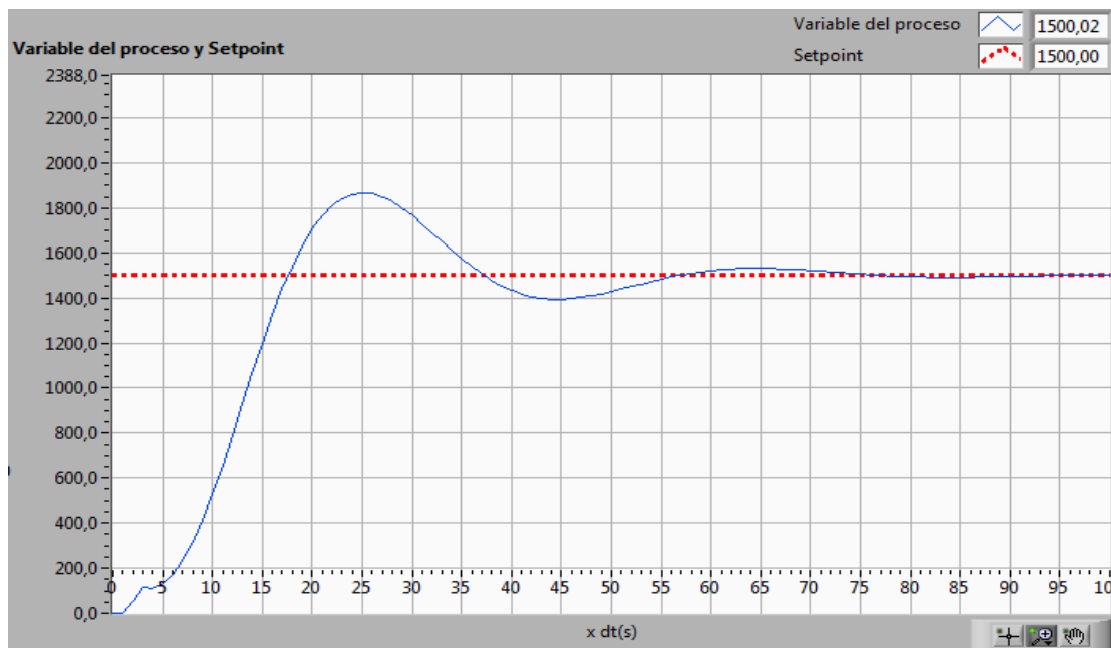


Figura 3.2: Efecto en la salida por variación en la carga

Si se disminuye la ganancia proporcional del controlador PID a 0.001 el efecto de esta consideración práctica disminuye. La salida del sistema se presenta en la figura 3.3 y en ella se observa como disminuye la oscilación en la salida del sistema. El tiempo de crecimiento y el tiempo de establecimiento aumentan respectivamente a 600 ms y 2200 ms, y en el caso del máximo sobreimpulso este disminuye a 12.6 %.

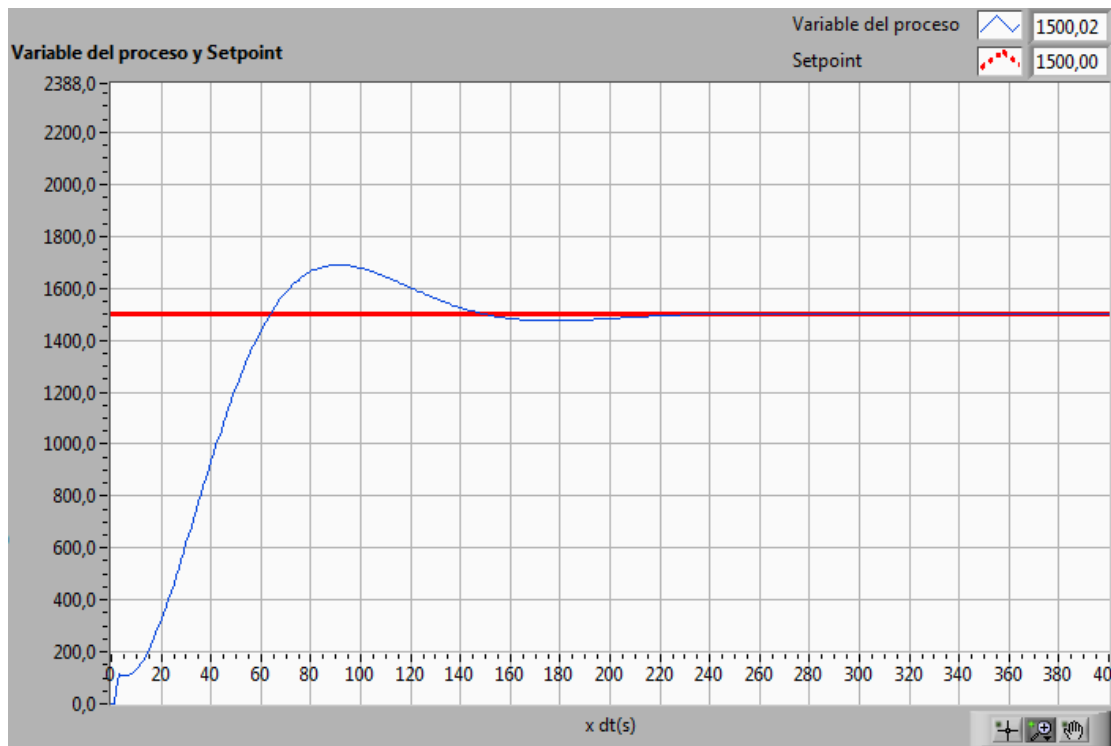


Figura 3.3: Efecto en la salida por variación en la carga

### Caso II: Variación del período de muestreo

Para este caso se seleccionan diferentes instantes de muestreo y se muestra la salida del sistema para cada uno de estos valores. En el primer caso se selecciona un instante de muestreo de 20 ms y la respuesta del sistema se muestra en la figura 3.4. Este presenta un comportamiento parecido al que se simula inicialmente con instante de muestreo igual a 10 ms. Como característica principal se observa que el tiempo de crecimiento se mantiene aproximadamente igual que el caso anterior. En el caso del máximo sobreimpulso y el tiempo de establecimiento aumentan respectivamente a 12.2 % y a 620 ms.

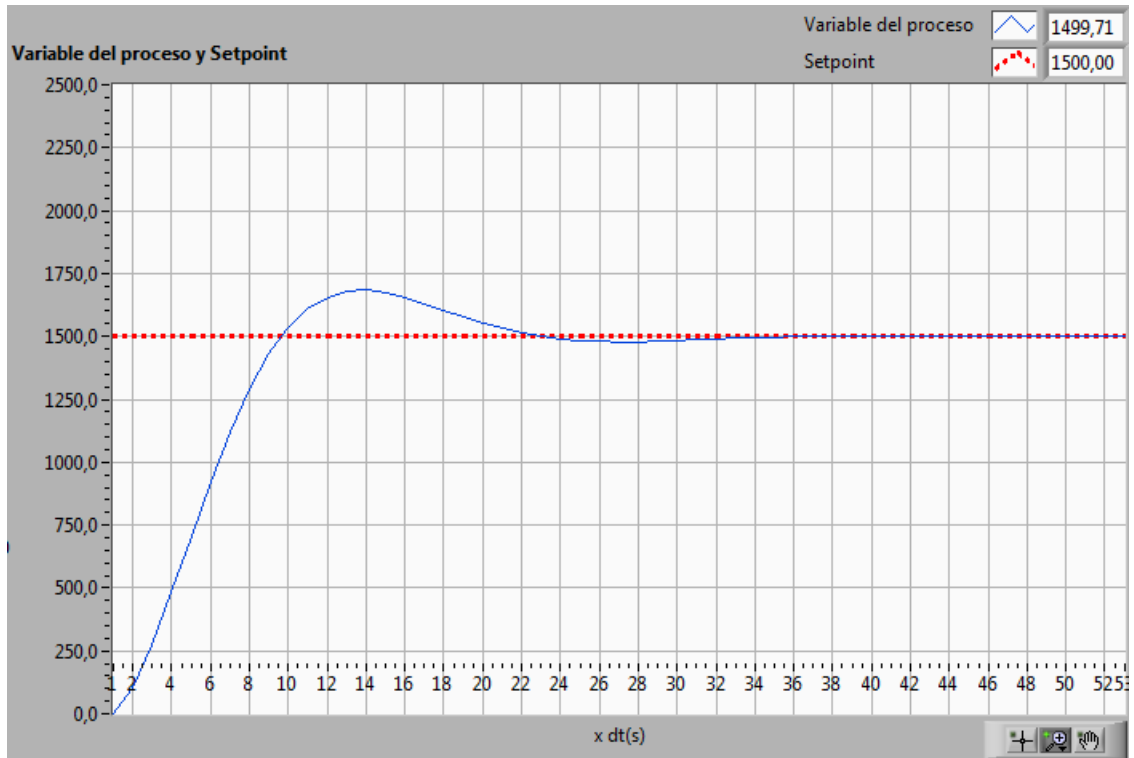


Figura 3.4: Salida del sistema para tiempo de muestreo de 20 ms

En el segundo caso se selecciona un tiempo de muestreo de 50 ms. En la figura 3.5 puede observarse la respuesta del sistema ante la nueva variación del tiempo de muestreo. Como característica fundamental tenemos una ligera oscilación en forma de pulsos debido al aumento del tiempo de muestreo. En ella se destaca el aumento del  $M_p$  y el  $t_s$  respecto a los casos en que el tiempo de muestreo era de 10 y 20 ms, aunque el  $t_r$  disminuye en un 10 %. A continuación se muestra los valores de  $M_p$ ,  $t_r$  y  $t_s$  para tiempo de muestreo de 50 ms:

Para tiempo de muestreo igual a 50 ms:

- $M_p = 28.4\%$
- $t_r = 150 \text{ ms}$
- $t_s = 700 \text{ ms}$

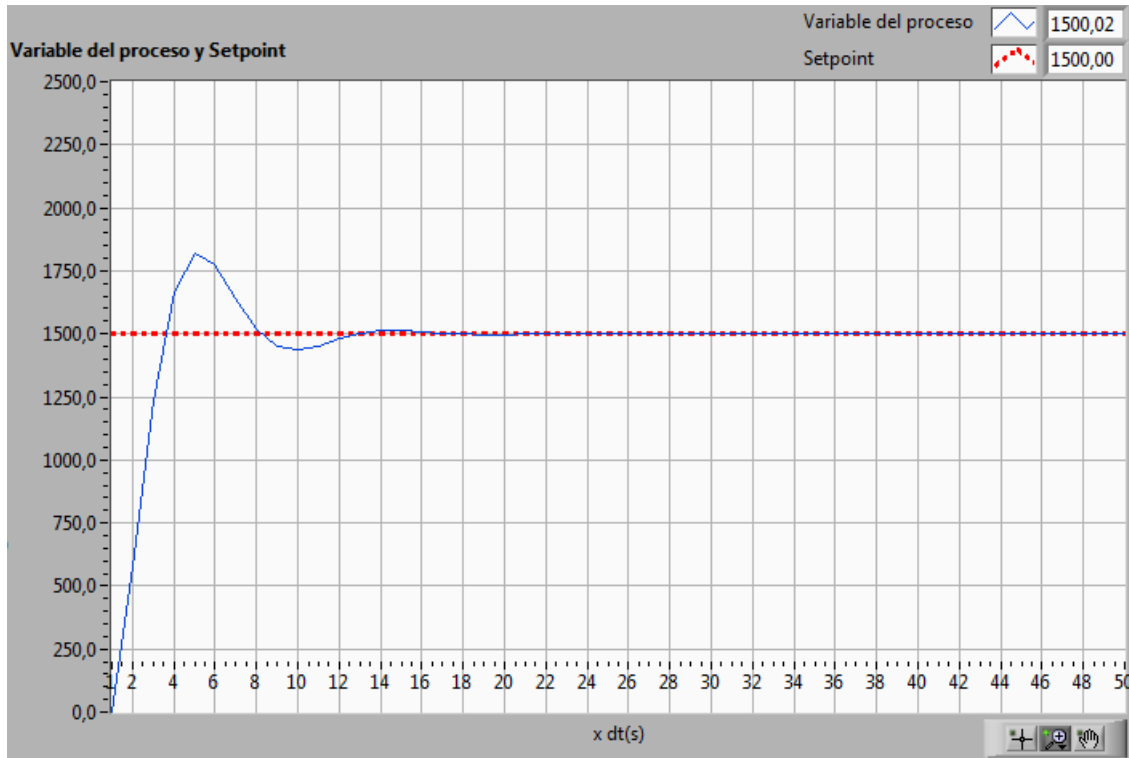


Figura 3.5: Salida del sistema para tiempo de muestreo de 50 ms

Como último caso se selecciona un tiempo de muestreo de 100 ms. En la figura 3.6 se observa el efecto producido en la salida del sistema ante la variación de este tiempo. En ella se observa como la salida del sistema se vuelve más oscilatoria y los valores de  $M_p$ ,  $t_r$  y  $t_s$  aumentan respectivamente a 53.8 %, 180 ms, 1400 ms respecto a los casos anteriores.

La selección del periodo de muestreo es una consideración práctica importante. Este contribuye en la estabilidad del sistema y ante diferentes instantes de muestreo los sistemas se comportan de manera diferente (Gil Naboja & Díaz Cordovés, 2009). Para un mismo instante de muestro de 100 ms y setpoint igual a 1588 rpm el sistema presenta más oscilaciones antes de estabilizarse e incluso supera la velocidad máxima del motor. Esto es sumamente importante porque en la vida real se hubiese presentado algún accidente y el motor hubiera sufrido daños, esto es debido a la mala selección del tiempo de muestreo. En la figura 3.7 se observa la salida del sistema para este caso.

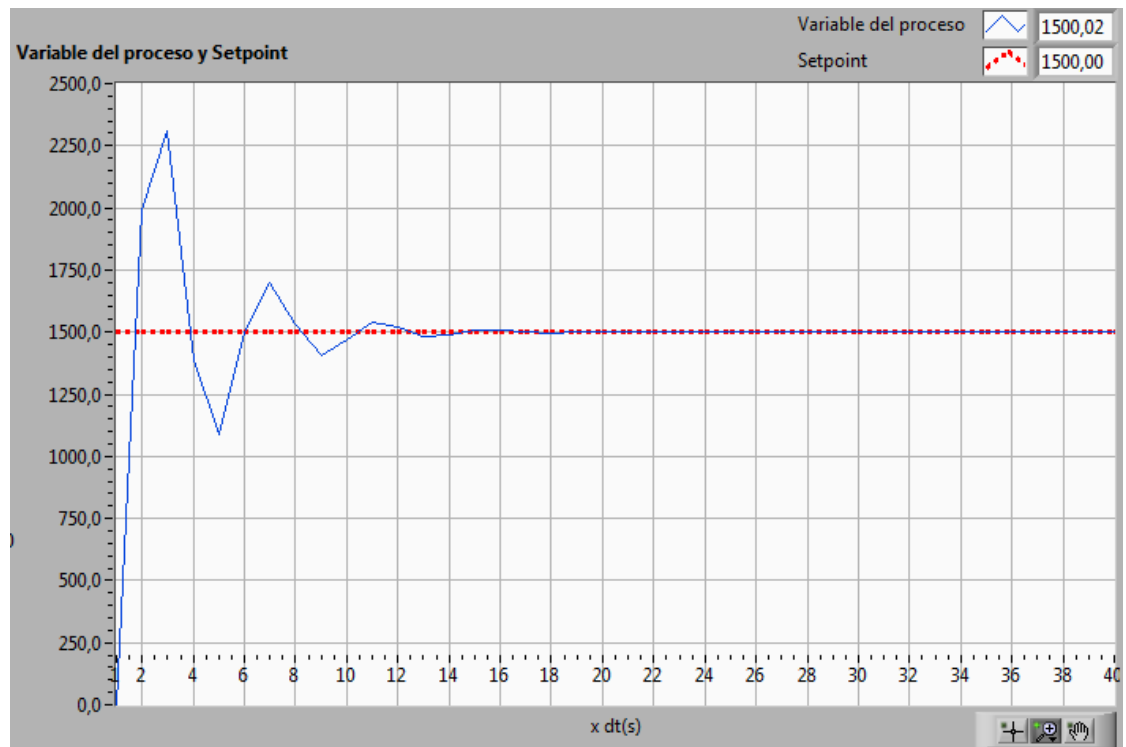


Figura 3.6: Salida del sistema para tiempo de muestreo de 100 ms

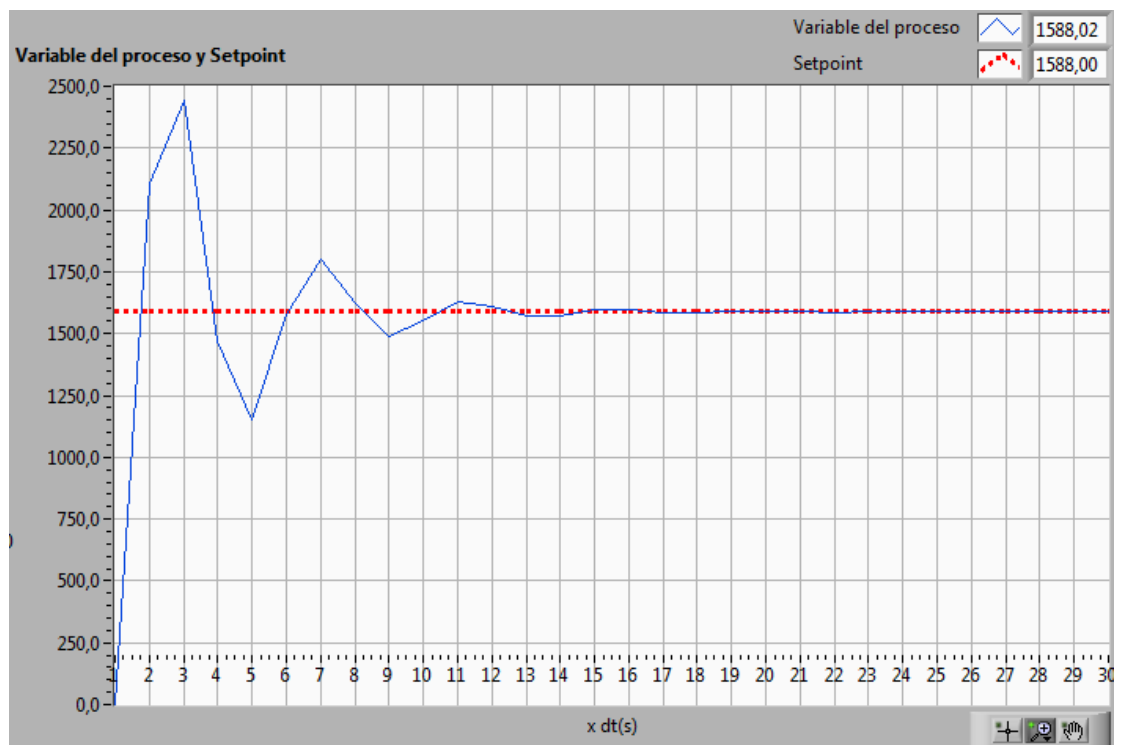


Figura 3.7: Salida del sistema para tiempo de muestreo de 100 ms y setpoint igual a 1588 rpm

### 3.2 Control de posición de un servomotor de corriente directa

Para el análisis de este ejemplo se selecciona un servomecanismo. En este se controla la posición de la carga mecánica de acuerdo con la posición angular del brazo del cursor del potenciómetro de entrada. El sistema presenta un amplificador que amplifica la señal de error que aparece en los terminales del potenciómetro antes de entrar al circuito de armadura del motor de corriente directa. El modelo del sistema según (Ogata) está dado por la ecuación 3.4 y la misma incluye el amplificador y al motor conjuntamente.

$$\frac{C(s)}{R(s)} = \frac{K}{Js^2 + Fs} \quad (3.4)$$

Dónde:

$C(s)$ : Desplazamiento angular del eje de salida

$R(s)$ : Posición angular del cursor del potenciómetro de entrada

$K$ : Ganancia del sistema

$J$ : Momento de inercia referido al eje de salida

$F$ : Coeficiente de fricción viscosa referido al eje de salida

#### 3.2.1 Configuración para la simulación.

Para la simulación de este ejemplo se toma en cuenta los mismos valores del amplificador y el motor utilizados en la simulación anterior. Se mantiene la ganancia del amplificador igual a  $238.8 \text{ rpm} * V^{-1}$  y la constante de tiempo del sistema igual a 0.25 segundos. De esta manera el modelo del sistema estaría representado de la siguiente forma:

$$\frac{C(s)}{R(s)} = \frac{238.8}{s(0.25s + 1)} \quad (3.5)$$

En la simulación se mantienen los mismos valores para el convertidor digital – analógico, el actuador, el filtro paso bajo y el tiempo de muestreo. En este caso se desea que el desplazamiento angular del eje de salida sea de  $180^\circ$ . Para el control de posición de la carga mecánica se selecciona un PID estándar o no interactivo y de diferentes pruebas realizadas se seleccionan las siguientes constantes:

Controlador PID estándar o no interactivo:



- Ganancia:  $K_c=0.009$
- Tiempo integral:  $T_i=0.0125$  s
- Tiempo derivativo:  $T_d=5$  s

La respuesta transitoria para este sistema se muestra en la figura 3.10. En ella se destaca una oscilacion amortiguada con un máximo sobreimpulso de 5.28 %. El tiempo de crecimiento para el sistema es de 1200 ms y tiempo de establecimiento de aproximadamente de 3500 ms.

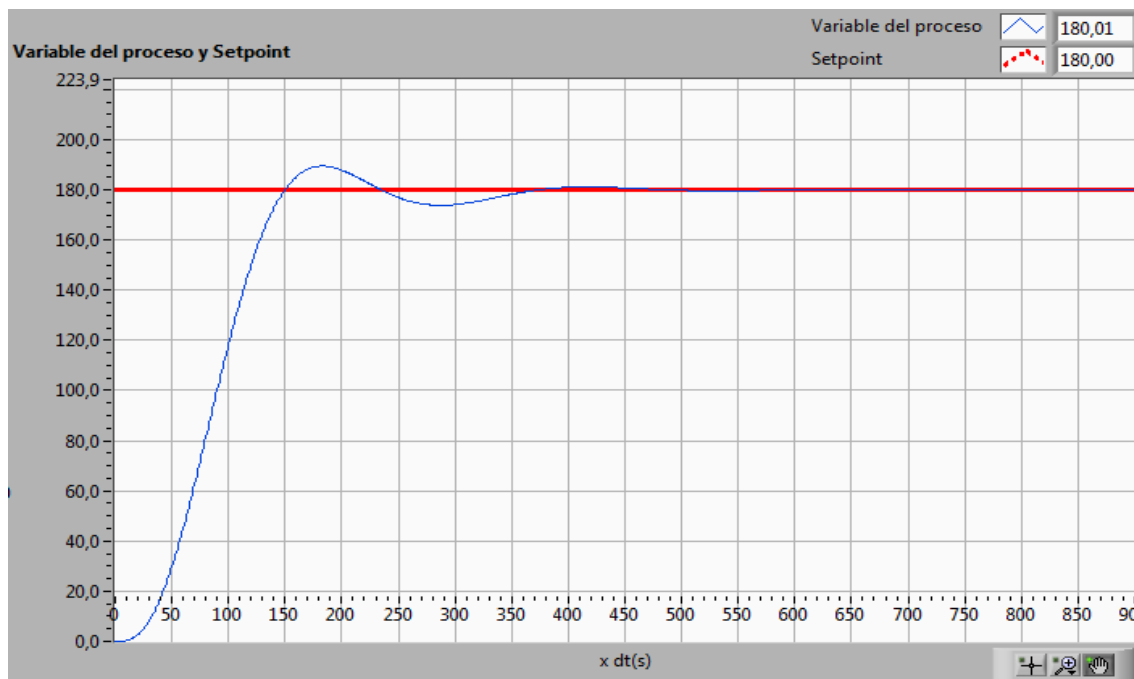


Figura 3.8: Desplazamiento angular de la carga mecánica

### 3.2.2 Consideraciones prácticas para el control de posición del servomotor de corriente directa

En este caso se presenta en dos casos diferentes los efectos producidos por diferentes consideraciones prácticas, estas son: variación del nivel ruido y filtraje en la señal de mando. Para esto se utilizan los mismos valores de configuración inicial y se simula el sistema para cada consideración práctica.

#### Caso I: Variación del nivel de ruido (%)

El nivel de ruido seleccionado es de 0.95% y la respuesta transitoria del sistema ante el mismo es mostrada en la figura 3.9. En esta se observa como la salida del sistema es afectada por el nivel de ruido, donde la misma

mantiene sus valores alrededor del punto estacionario pero nunca alcanza a estabilizarse. En la práctica los sistemas de control se ven afectados por señales externas o ruidos generados por elementos del mismo por lo que de una forma u otra inciden en la estabilidad del sistema. Para que este sistema alcance la estabilidad nuevamente se tendría que ajustar el controlador o seleccionar un nuevo control.

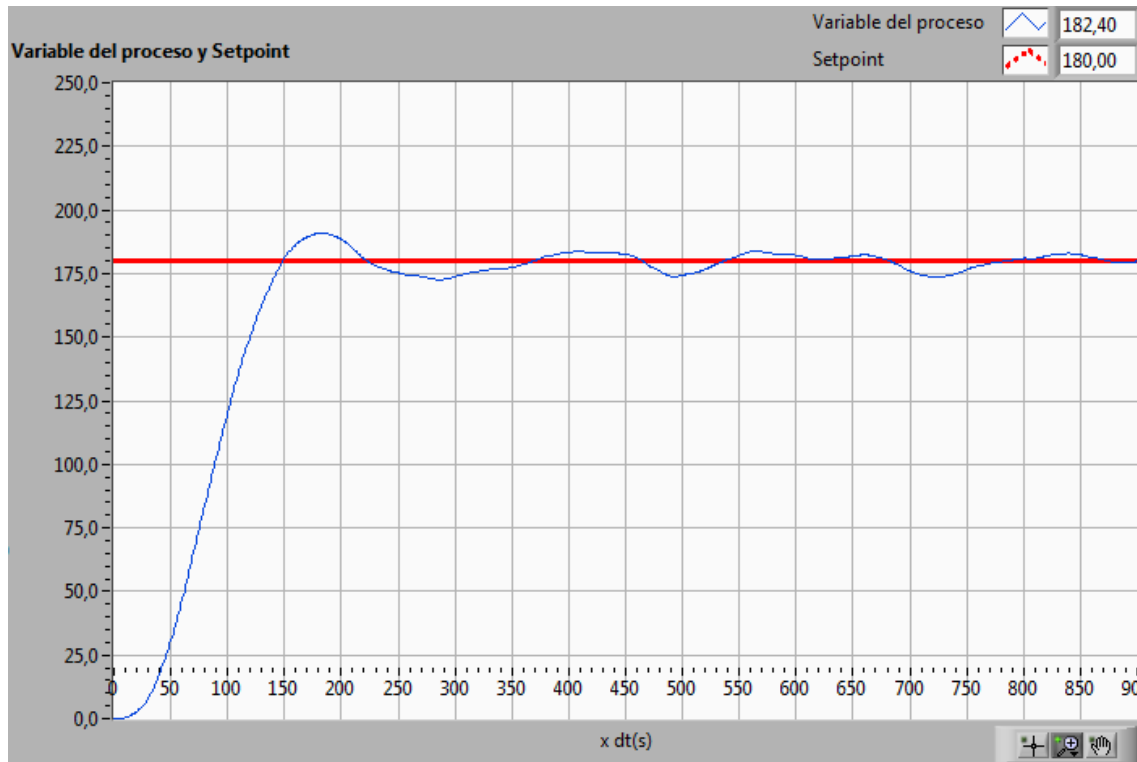


Figura 3.9: Efecto en la salida por la variación del nivel de ruido

### Caso II: Filtraje en la señal de mando

En la práctica real puede ocurrir que como consecuencia de un cambio en la referencia o por la aparición de una perturbación en el proceso, la variable de control alcance los límites del actuador. Esto es sumamente peligroso en aquellos sistemas con espacio de trabajo limitado. En aplicaciones de robótica si existe este fenómeno, el robot sobrepasaría la referencia sin que disminuya su velocidad y llegaría a los topes del mecanismo provocando eventualmente su rotura (Gil Naboja & Díaz Cordovés, 2009).

En este caso para que se produzca este efecto se mantiene el PID estándar o no interactivo y de diferentes pruebas realizadas se seleccionan las siguientes constantes:

Controlador PID estándar o no interactivo:

- Ganancia:  $K_c=2$
- Tiempo integral:  $T_i=0.001$  s
- Tiempo derivativo:  $T_d=0.1$  s

El efecto de saturación se muestra en la figura 3.10. La variable de control alcanza los límites del actuador cuando esta entra en el convertidor D/A y la salida de este produce oscilaciones desde su valor máximo hasta el valor mínimo. Esta es amplificada por el actuador y la salida oscila entre el valor máximo y mínimo del mismo.

El filtraje de la señal de mando es una consideración práctica importante ya que contribuye a que la salida del controlador nunca alcance los límites del actuador. Para este caso se selecciona la constante de filtro  $\alpha$  igual a 0.01. De esta manera el filtro paso bajo del entrenador filtra la señal de salida del controlador y disminuye estas oscilaciones, en la figura 3.11 se muestra como este efecto desaparece.

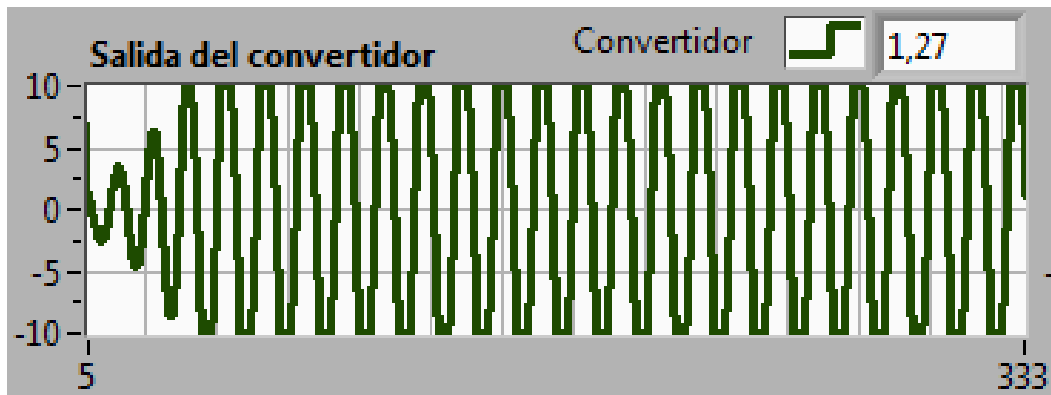


Figura 3.10: Salida del convertidor D/A, con  $\alpha = 1$

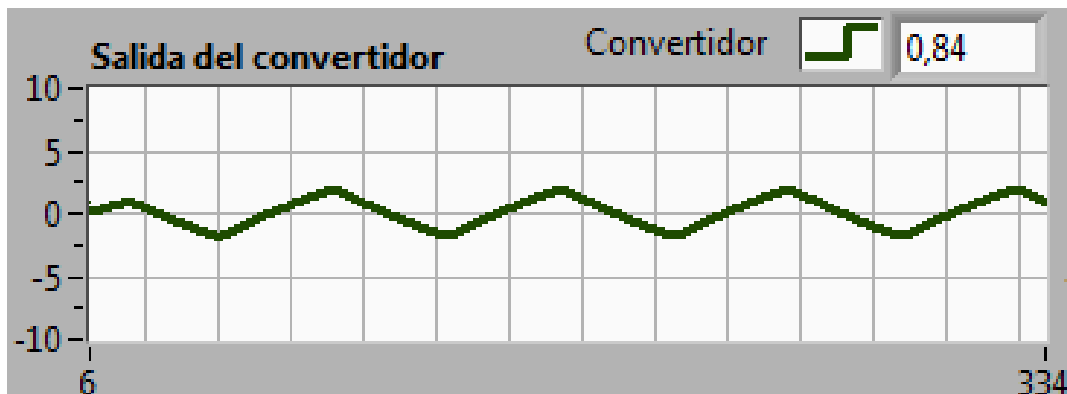


Figura 3.11: Salida del convertidor D/A, con  $\alpha = 0.01$

### **3.3 Valoración económica**

La viabilidad y fiabilidad económica de esta investigación está vinculada a las ventajas desde el punto de vista económico, que posee la utilización del software LabVIEW en la simulación de sistemas de control. El software LabVIEW por su capacidad de facilidad de uso y rapidez de programación resulta sumamente eficiente, ya que no requiere de grandes inversiones de tiempo y recursos. Esto implicaría un gasto mínimo de energía eléctrica, producto al tiempo que se emplea para la simulación en una computadora.

El hecho de dotar al software LabVIEW en una aplicación de simulación de procesos aumenta la posibilidad de su utilización. En casos en que sea necesario implementar un entrenador digital de sistemas de control no se rechazará el LabVIEW como herramienta de simulación. El diseño propuesto complementa entonces el uso de este excelente software que por su eficacia y bajo costo es cada día más utilizado en los niveles educativos.

### **3.4 Consideraciones finales del capítulo**

La introducción de consideraciones prácticas en la simulación constituye un elemento importante. El uso de las mismas durante el entrenamiento permite que el usuario se familiarice con situaciones reales de los sistemas de control industrial. De esta manera el usuario aplica sus conocimientos y habilidades para enfrentarse a las situaciones dadas. El análisis de sistemas de control mediante este entrenador de lazo cerrado con consideraciones prácticas reales fundamenta la importancia del uso de sencillos entrenadores digitales como herramienta de estudio en los sistemas de control.

## **Conclusiones**

Con la culminación de la investigación y del análisis de los resultados alcanzados se arriban a las siguientes conclusiones:

1. El análisis bibliográfico determinó que los entrenadores digitales de sistemas de control son ampliamente utilizados en el mundo. La bibliografía consultada nos permite afirmar que la utilización de los mismos, en el análisis de estos sistemas es cada vez más frecuente dado por la factibilidad de dicha herramienta en numerosos procesos.
2. La presente investigación permitió constatar la facilidad de uso y rapidez de programación del software LabVIEW para el diseño de nuevas aplicaciones, lo que hace del mismo un software con diversas posibilidades de uso.
3. El estudio realizado resaltó la importancia de las consideraciones prácticas reales en los procesos en cuestión para el análisis de los sistemas de control.
4. La aplicación del diseño desarrollado proporcionó buenos resultados tanto en el estado estable como en el estado transitorio de los sistemas a controlar.

## **Recomendaciones**

Terminado el trabajo se realizan las siguientes recomendaciones:

1. Utilizar el diseño propuesto en el análisis de sistemas de control que así lo requieran.
2. Continuar el estudio de análisis de los sistemas de control en los entrenadores digitales.

## Bibliografía

1. Ataurima Arellato, M. (2013). MATLAB & Simulink para Ingeniería Nivel I. Perú.
2. B, D. G., & Tabares G, I. (1997). *Teoría de control*.
3. Ballesteros Horta, R. L., & Hernández Maldonado, A. F. (2009). Control por computadora. Santa Clara, Villa Clara, Cuba.
4. Cilento, A. (2007). *Sistemas de control de motores en tiempo real mediante MATLAB*.
5. EcuRed. (2016). *EcuRed*. Recuperado el 2 de Febrero de 2016, de [http://www.ecured.cu/Sistema\\_SCADA](http://www.ecured.cu/Sistema_SCADA)
6. Fernández, R. (2008). Control de posición de un servomotor de corriente continua. Laboratorio de control.
7. Gil Nabojas, J. J., & Díaz Cordovés, Á. R. (2009). *Fundamentos de control automático de sistemas continuos y muestreados*. San Sebastián, España.
8. Hernández Cevallos, M. I., & Ledesma Marcalla, D. A. (2010). *Desarrollo de un sistema SCADA para la medición de voltajes con sistemas embebidos para el laboratorio de mecatrónica de la facultad de mecánica*. Chimborazo.
9. Holguín Londoño, G. A., Pérez Londoño, S. M., & Orozco Gutiérrez, Á. Á. (2002). Curso básico LabVIEW 6i. Pereira.
10. Jolly, A. (2016). *Ingeniero independiente especialista en Sistemas Industriales y Científicos, Banco de Pruebas y Mediciones, Supervisión, Control, LabVIEW, TestStand, LabWindows/CVI*. Recuperado el 4 de Abril de 2016, de <http://www.ajolly.com.mx/es/desarrollo-consulta-sistema-supervisor-control-prueba-medida/17-LabWindows-CVI.html>
11. Lander Simulation & Training Solutions. (2015). *Historia de la simulación*. Recuperado el 12 de Enero de 2016, de

- <http://www.landersistimulation.com/formacion-con-simulacion/el-mundo-en-movimiento/historia-de-la-simulacion/>
12. López, T., & J. I. (s.f.). Conceptos básicos de simulación de procesos. Valencia, Valencia, España.
  13. Montero Avendaño, A. K. (2011). *Simulación de procesos empresariales*. Recuperado el 12 de Enero de 2016, de <https://simulaciondeprocesosempresariales9.wordpress.com/category/a-historia-de-la-simulacion/>
  14. National Instruments. (2016). *National Instruments*. Recuperado el 30 de Marzo de 2016, de <http://www.ni.com/lwcv/whatis/esa/>
  15. Nuñez Valle, Á. (2011). *Control de un tanque de presión mediante el módulo PID de un autómatas programable*. Madrid.
  16. Ogata, K. (s.f.). *Ingeniería de control moderna* (Tercera ed.). Minnesota, Estados Unidos.
  17. Perez Rojas, E. A. (2008). *Diseño y montaje de un entrenador con el Controlador Lógico Programable PLC S7-200 de SIEMENS*. Bucaramanga.
  18. Ripoll, I. (2001). *Introducción a RTLinux*. Recuperado el 2016 de Marzo de 2016, de <http://www.gii.upv.es/rtportal/tutorial/01-intro/tr1.html>
  19. Romero Ariza, M., & Quesada Armenteros, A. (2014). Nuevas tecnologías y aprendizaje significativo de las ciencias. *Enseñanza de las ciencias*(32), 101-115.
  20. Scenna, N. J. (1999). *Modelado, simulación y optimización de procesos químicos*. Santa Fe.
  21. Shannon, R. E. (1998). *Introduction to the art and science of simulation*. Texas.
  22. Tarifa, E. E. (s.f.). *Teoría de modelos de simulación. Introducción a la simulación*. Jujuy.



23. Veiga, A. L. (s.f.). *Sistemas de tiempo real*. Recuperado el 9 de Febrero de 2016, de <http://www.electro.fisica.unlp.edu.ar/temas/p7/RTS-1.html>
24. Zurita Landaeta, L. M. (2014). *Simulador digital de procesos*. Cumaná.