

UCLV
Universidad Central
"Marta Abreu" de Las Villas



FIE
Facultad de
Ingeniería Eléctrica

TRABAJO DE DIPLOMA

Departamento de Automática y Sistemas Computacionales

Título: Software para el control automático digital del servocontrol SAD-100

Autor: Juan Alberto Martín Besada

Tutor: Dr.C. Roberto Luis Ballesteros Horta

Santa Clara, julio 2018
Copyright©UCLV

Universidad Central “Marta Abreu” de Las Villas

Facultad de Ingeniería Eléctrica

Departamento de Automática y Sistemas Computacionales



TRABAJO DE DIPLOMA

Software para el control automático digital del servocontrol SAD-100

Autor: Juan Alberto Martín Besada

Email: jmbesada@uclv.cu

Tutor: Dr.C. Roberto Luis Ballesteros Horta

Dpto. de Automática, Facultad de Ing. Eléctrica, UCLV

Email: rball@uclv.edu.cu

Santa Clara

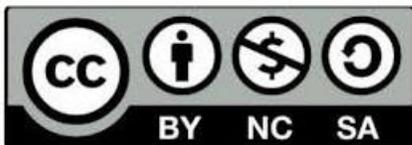
2018

"Año 60 de la Revolución"

Este documento es Propiedad Patrimonial de la Universidad Central “Marta Abreu” de Las Villas, y se encuentra depositado en los fondos de la Biblioteca Universitaria “Chiqui Gómez Lubian” subordinada a la Dirección de Información Científico Técnica de la mencionada casa de altos estudios.

Se autoriza su utilización bajo la licencia siguiente:

Atribución- No Comercial- Compartir Igual



Para cualquier información contacte con:

Dirección de Información Científico Técnica. Universidad Central “Marta Abreu” de Las Villas. Carretera a Camajuaní. Km 5½. Santa Clara. Villa Clara. Cuba. CP. 54 830

Teléfonos.: +53 01 42281503-1419

PENSAMIENTO

Si tú crees que puedes, puedes. Si tú crees que no puedes, no puedes. Tanto si piensas una cosa como la otra, estás en lo cierto.

Henry Ford

DEDICATORIA

A mis padres, Leticia y Juan Antonio, por ser mi inspiración y alentarme siempre a seguir por el difícil camino de la ciencia.

AGRADECIMIENTOS

A mis padres, por guiarme y aconsejarme desde siempre.

A mi novia Maday, por estar a mi lado en todo momento y alentarme para seguir adelante.

A mi abuela Daisy, que siempre ha estado ahí con buenos consejos.

A mi prima Odarbys, quien desde la distancia me animó en el momento más difícil, el comienzo.

A mi tutor Roberto Ballesteros, por guiarme y aconsejarme con sabiduría durante el desarrollo de este trabajo.

A Raulito y Mabel por acogerme con cariño en estos meses.

A mis amigos: Pino, José Alberto, Luis Javier, Ana Maira, Diony, Javier, Abelito, Rubén, Annalie, Yazmany y Daniel.

A todos mis profesores que me han formado en estos años.

A todos muchas gracias.

RESUMEN

El empleo de la simulación por computadora contribuye a la adquisición de conocimientos y habilidades de situaciones reales. La presente investigación se realiza para los laboratorios de la carrera de Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas con el objetivo de diseñar un software para el control automático digital del servocontrol SAD-100 que incluya consideraciones prácticas con el propósito de mejorar la formación integral de los estudiantes como ingenieros. El diseño del mismo se realiza en el software LabVIEW. Para la valoración de los resultados se efectúan pruebas simuladas y reales. En cada caso se simula inicialmente con valores ideales y posteriormente se introducen varias consideraciones prácticas para analizar su efecto en la salida del sistema. Se realiza una valoración económica y medioambiental de la aplicación de este trabajo. La utilización del software creado permite mejorar el proceso de formación de los estudiantes de la carrera Ingeniería en Automática.

TABLA DE CONTENIDOS

PENSAMIENTO	i
DEDICATORIA	ii
AGRADECIMIENTOS	iii
RESUMEN	iv
INTRODUCCIÓN	1
Organización del informe	4
CAPÍTULO 1. Control digital en procesos reales y virtuales	5
1.1 Control digital	5
1.1.1 Breve historia del control digital	5
1.1.2 Control digital directo	6
1.2 Tiempo real	8
1.2.1 Definición de un sistema de tiempo real	8
1.2.2 Software para trabajo en tiempo real	9
1.3 Simulación	12
1.3.1 Breve historia de la simulación	12
1.3.2 Simulador de procesos	13
1.4 Sistemas de control	15
1.4.1 Clasificación general de sistemas de control	15
1.4.2 Controlador PID	16
1.5 Consideraciones prácticas en la implementación de algoritmos de control en procesos simulados	20
1.6 Antecedentes de software para el control de procesos simulados	23

1.7	Consideraciones finales del capítulo.....	25
CAPÍTULO 2. Diseño del software para el control digital del SAD-100.....		26
2.1	Caracterización del SAD-100	26
2.2	Caracterización de la tarjeta de adquisición de datos NI USB-6212	32
2.3	Selección de funciones y ejemplos de LabVIEW	33
2.3.1	Ejemplos y funciones de PID.....	34
2.3.2	Funciones de adquisición de datos.....	35
2.3.3	Ejemplo seleccionado	36
2.4	Diseño	38
2.4.1	Interfaz hombre-máquina.....	38
2.4.2	Proceso.....	40
2.4.3	Regulador.....	41
2.5	Programación en LabVIEW	43
2.6	Consideraciones finales del capítulo.....	45
CAPÍTULO 3. Pruebas y resultados		46
3.1	Pruebas simuladas	46
3.1.1	Pruebas con LabVIEW	46
3.1.2	Prueba con MATLAB.....	50
3.1.3	Prueba de tiempo real	51
3.2	Pruebas reales.....	52
3.3	Guía para práctica de laboratorio	57
3.4	Valoración económica e impacto medioambiental	57
CONCLUSIONES Y RECOMENDACIONES		59
Conclusiones.....		59

Recomendaciones	59
REFERENCIAS BIBLIOGRÁFICAS	61
ANEXOS	64
Anexo I Tabla de la característica estática del servomotor	64
Anexo II Implementación de las no linealidades	66
Anexo III Lazo de control programado en LabVIEW	66
Anexo IV Pruebas simuladas	67
Anexo V Pruebas reales	71
Anexo VI Guía para práctica de laboratorio	74

INTRODUCCIÓN

En los últimos años el control digital se ha aplicado cada vez más en el manejo y supervisión de los procesos industriales. Ello se explica porque permite mayor automatización y coordinación en la operación de las plantas, que las logradas con los sistemas analógicos de control. Este proporciona múltiples ventajas ya que incrementa la productividad del trabajo humano, provee ahorros económicos por concepto de materias primas, energía, combustibles y demás insumos de una planta industrial. Por otra parte mejora el control de la calidad de la producción, el sistema de seguridad de la planta y reduce el número de accidentes (Kuo, 2000).

Los sistemas de control digital en comparación con los analógicos presentan una serie de mejoras que justifican porque en la actualidad son los más utilizados a nivel global. Cuando se desea modificar algunos de sus parámetros no es necesario realizar cambios en el cableado, basta con modificar la programación. Son menos vulnerables a los ruidos procedentes de las redes eléctricas y a los ocasionados por sucesos externos al sistema de control, como arranque y parada de motores o descargas y cortos en las líneas de potencia.

Una de las variantes del control digital es el control digital directo, en el cual el controlador del sistema lo constituye la computadora.

La simulación se ha utilizado en el descubrimiento de soluciones exactas a problemas clásicos de la industria y la ingeniería. En años recientes el avance de la informática y las tecnologías ha permitido que la simulación pueda ser empleada prácticamente en todos los campos de la ciencia y la ingeniería (Rodríguez, 2016).

La simulación tiene una estrecha relación con el entrenamiento, la misma ha permitido crear entrenadores que permiten adquirir conocimientos y habilidades en situaciones cotidianas.

El uso eficiente de la simulación tiene un alto impacto económico, pues permite estudiar y evaluar diferentes fenómenos de la ingeniería con pocos recursos (Hosseinpour and Hajihosseini, 2009). Cuando se simula es posible realizar muchas repeticiones virtuales del proceso en estudio y variar sus parámetros. Para que dicha simulación tenga éxito es necesario ajustar el modelo virtual con el real. Algunos ejemplos que evidencian lo antes mencionado son: el entrenamiento de pilotos de aviación en simuladores físicos; el uso de simuladores en la industria petrolera para la optimización y reducción de costos; en la industria química en general y los entrenadores de sencillos sistemas de control (Aguirre and Guarnizo, 2008) (Villegas, 2007).

En los laboratorios el uso de la simulación brinda la posibilidad de crear varios puestos de trabajos para estudiar un proceso sin la necesidad de tener en cada uno de ellos el equipamiento completo.

Es un hecho que en el Departamento de Automática de la Universidad Central “Marta Abreu” de Las Villas existen insuficiencias en el número de equipos que permitan a los estudiantes realizar prácticas reales durante la carrera; esto influye negativamente en el desarrollo del proceso de aprendizaje y por consiguiente en la formación integral de los alumnos. Por otra parte no es muy común tener en cuenta consideraciones prácticas cuando se diseñan los sistemas de control.

En la actualidad existen dos servomotores SAD-100, que son utilizados para impartir la asignatura Herramientas de Software para el Control, esto permite crear dos puestos de trabajo para prácticas reales, por lo que se requiere mucho tiempo para que todos los estudiantes realicen estas prácticas. Una alternativa para atenuar esta deficiencia es la realización de prácticas simuladas, para ello es necesario disponer de un software para efectuar el control automático digital del servocontrol SAD-100, los creados hasta el momento no incluyen consideraciones prácticas. Lo expuesto anteriormente motiva el desarrollo de la presente investigación.

De la problemática presentada anteriormente se plantea como problema científico: la no existencia de un software para el control automático digital del servocontrol SAD-100 que incluya consideraciones prácticas, tanto para la configuración como la operación, que se

pueda utilizar en los laboratorios de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas.

Para dar solución al problema científico se propone como objetivo general: diseñar un software para el control automático digital del servocontrol SAD-100 con consideraciones prácticas para utilizarse en los laboratorios de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas con el propósito de mejorar la formación integral de los estudiantes como ingenieros.

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

- Analizar la bibliografía sobre control digital y simuladores de control de proceso.
- Caracterizar el servocontrol SAD-100 y la tarjeta de adquisición de datos NI USB-6212.
- Diseñar el software de control digital con LabVIEW para el proceso real y virtual.
- Valorar los resultados de las pruebas realizadas.
- Proponer una guía para la realización de una actividad práctica.

Con la realización del trabajo de diploma se logra una sistematización del conocimiento sobre control digital directo con consideraciones prácticas. El software en LabVIEW para desarrollar distintas prácticas reales y simuladas por los estudiantes. Una guía que permita la realización de alguna práctica de laboratorio y sirva como base para la creación de otras.

Con la implementación del presente trabajo de diploma se darán soluciones a problemáticas modernas vinculadas con la docencia del Departamento de Automática de la Universidad Central “Marta Abreu” de Las Villas. Constituirá una herramienta tanto para alumnos como para profesores que mejorará el proceso docente educativo.

Con el software para el control automático digital del servocontrol SAD-100 se crearán varios puestos virtuales de trabajo, lo que reduce el costo por concepto de equipamiento que se requiere para la realización de prácticas reales, solo será necesario disponer de una maqueta ALECOP SAD-100.

Un análisis documental es realizado sobre la literatura existente a cerca de simuladores de procesos. Se simula el proceso con el uso de LabVIEW 2010, se realiza experimentación

práctica con la tarjeta multifuncional de adquisición de datos NI USB-6212 y LabVIEW. Distintas pruebas son efectuadas para la realización de prácticas de laboratorio.

Organización del informe

El informe está estructurado en introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas y anexos. Los temas que se abordan en cada capítulo se estructuran de la forma siguiente:

CAPÍTULO I: Se dedica al análisis de la bibliografía existente sobre control digital, consideraciones prácticas para el desarrollo de control digital directo. Se plantean los conceptos fundamentales sobre los simuladores. Se realiza un análisis sobre antecedentes de la problemática abordada. Se termina con las consideraciones finales del capítulo.

CAPÍTULO II: Se plantea una caracterización del servocontrol SAD-100. Se seleccionan las funciones y ejemplos sobre control PID que incluyen consideraciones prácticas. Un análisis de las funciones sobre PID seleccionadas es realizado, así como el diseño del sistema de control digital con LabVIEW para el proceso real y virtual. Concluye con las consideraciones finales del capítulo.

CAPÍTULO III: Se realizan diferentes pruebas tanto reales como simuladas del servocontrol SAD-100. Se valoran los resultados obtenidos. Se realizan una valoración técnico-económica y medioambiental del trabajo.

CAPÍTULO 1. Control digital en procesos reales y virtuales

En este capítulo se exponen los conceptos generales relacionados con el control digital y específicamente sobre control digital directo en los procesos industriales. En el mismo se presenta los conceptos de tiempo real, simulador, sistemas de control, controlador PID y consideraciones prácticas en la implementación de algoritmos de control para el trabajo en tiempo real. Se muestra algunos antecedentes sobre el uso de simuladores de procesos en tiempo real y se culmina con las consideraciones finales del capítulo.

1.1 Control digital

El control digital ha desempeñado un papel muy importante en el avance de la ingeniería y la ciencia. Estos controles se han vuelto una parte importante de los procesos modernos industriales de manufactura. Las aplicaciones de estos dispositivos van de un simple encendido y apagado de una lámpara hasta procesos tan complicados y veloces como el diseño de sistemas pilotos automáticos en la industria aeroespacial.

1.1.1 Breve historia del control digital

En su libro Eduard Bertran Albertí (Albertí, 2003) expone los momentos fundamentales en el desarrollo del procesamiento digital de señales y del control digital. Aquí se presenta un resumen de la historia del control digital. Antes de la aparición de las primeras computadoras ya existían sistemas que sólo trabajaban con señales digitales, normalmente binarias. La reducción en el costo de los equipos de cómputo, así como las facilidades de diseño de sistemas basados en microprocesador y de circuitos con un alto nivel de integración permitió la aparición en el mercado de productos basados en procesadores digitales, y orientados a la manipulación de señales analógicas.

Durante la década de los 40 apareció un primer interés con la utilización de sistemas muestreados en radares, pero una inexitosa experiencia en el empleo de un controlador digital a un helicóptero pospuso hasta finales de la década de los 60 la confianza en los métodos digitales de control. No obstante, en la década de los ochenta se produjeron avances en microelectrónica, lo que permitió pasar de los primeros microprocesadores de la década anterior a dispositivos más potentes, sobre todo en número de bits, capaces de calcular algoritmos complejos en tiempo real. Aunque, al principio, su ancho de banda era muy limitado, lo cual no impedía el uso de los mismos en aplicaciones en el campo del control.

El control digital de procesos y de dispositivos ha remplazado al analógico en múltiples aplicaciones. Los sistemas digitales se han mostrado más fiables, versátiles y productivos que los analógicos, desde los controladores lógicos programables (PLC), ampliamente difundidos en la solución de problemas industriales, hasta los potentes algoritmos de control robusto o adaptativo, el control digital se encuentra presente en todos los niveles de automatización. Los sistemas distribuidos de adquisición de datos y control (SCADA), poco explotados en décadas pasadas, son ahora esenciales en aplicaciones de control de procesos.

1.1.2 Control digital directo

Según (Piedrafita, 1999) el desarrollo de la computación aplicada al control de procesos industriales fue impulsado a finales de los años 50 por causa de lo complicadas que eran algunas plantas industriales de controlar como las refinerías de petróleo. Los sistemas de control disponibles estaban bastantes limitados, por lo que la calidad de la producción dependía de la experiencia y habilidades de los operadores.

El primer artículo sobre el empleo de la computadora en el control industrial fue realizado por Brown y Campbell en el año 1950 y se tituló *Instrument engineering: its growth and promise in process-control problem*. En dicho artículo se usa una computadora para controlar un sistema de lazo cerrado, se asume que los elementos de cálculo y control deben ser computadoras analógicas, pero sugieren como una posibilidad el empleo de una computadora digital.

La primera computadora dedicada al control industrial fue instalada en la refinería de Port Arthur, Texas por la compañía Texaco, que utilizó un ordenador RW-300. Esta fábrica comenzó a funcionar controlada en lazo cerrado por computadora el 15 de marzo de 1959.

Este sistema no realizaba un control digital directo (DDC), sino un control supervisor dedicado a calcular las referencias óptimas de los reguladores analógicos. El primer control digital directo fue implementado en noviembre de 1962 en la planta de amoníaco y soda de la compañía Imperial Chemical Industries en Fleetwood, Reino Unido, mediante una computadora Ferranti Argus 200. Este sistema tenía 120 lazos de control y efectuaba la medición de 256 variables.

El DDC es el control automático de un proceso mediante el uso de un dispositivo digital que puede ser una computadora o un microprocesador, en tiempo real y con programa almacenado en su memoria interna. Las ventajas básicas del DDC son: la flexibilidad al permitir cualquier configuración y variar los parámetros y algoritmos, la precisión puesto que el cálculo digital posee un mayor grado de exactitud que el analógico. Por otra parte los sistemas digitales tienen una mayor velocidad de procesamiento que los analógicos debido a que el intervalo entre las muestras, el periodo de muestreo, puede ser muy pequeño (Fadali and Visioli, 2013). En la figura 1.1 se muestra el esquema general de control digital directo.

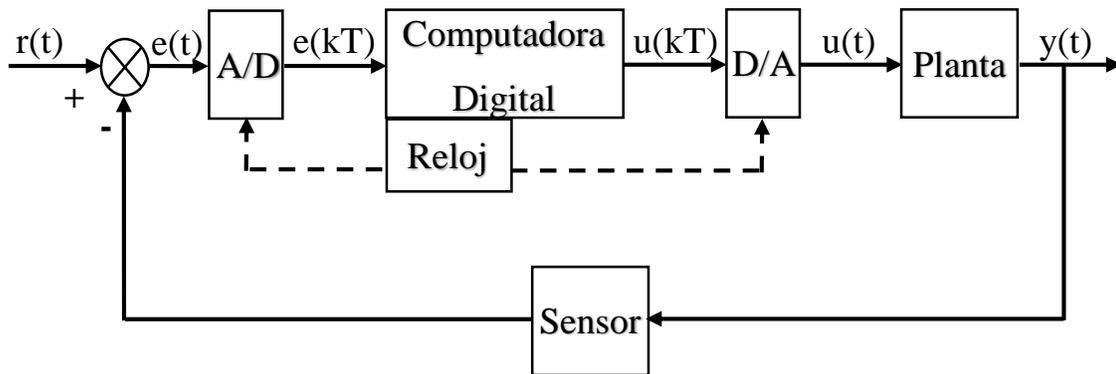


Fig. 1.1 Esquema general de control digital directo (DDC)

Otras ventajas de usar esta forma de control se reflejan en una mayor sensibilidad y una gran inmunidad a la distorsión de las señales; básicamente la sensibilidad se consigue gracias al empleo de señales de baja energía, mientras que la inmunidad a la distorsión causada por el ruido y las no linealidades, se obtiene gracias al buen acoplamiento de estas señales en los dispositivos digitales empleados.

1.2 Tiempo real

A medida que los computadores son más pequeños, rápidos, fiables y baratos, su rango de aplicaciones se amplía. Fueron construidos inicialmente para resolver ecuaciones, pero actualmente se utilizan en todos los órdenes de la vida, desde lavadoras a control de tráfico aéreo. Una de las áreas de la computación que más rápido se ha desarrollado es la que implica aplicaciones cuya función principal no es la de procesar información, pero que requiere de dicho procesamiento para realizar su función principal. Un horno de microondas es un ejemplo de este tipo de sistemas. En este caso la función principal es calentar o cocinar alimentos, sin embargo, se ejecutan distintos programas en dependencia del tipo de alimento que se quiera procesar. Este tipo de aplicaciones se conocen comúnmente como de tiempo real o embebidas. Se ha estimado que el 99 por ciento de la producción mundial de microprocesadores se utiliza en sistemas embebidos. Los mismos necesitan requisitos específicos en los lenguajes de programación que emplean, ya que presentan características diferentes a las de los sistemas de procesamiento de información tradicionales (Burns and Wellings, 2003).

1.2.1 Definición de un sistema de tiempo real

En el texto *Sistemas de Tiempo Real y Lenguajes de Programación* (Burns and Wellings, 2003) se presenta en su capítulo 1, p. 2 diferentes conceptos de un sistema de tiempo real, de los cuales en este trabajo de diploma se asume el tomado del *Oxford Dictionary of Computing* (Diccionario Oxford de computación):

“Cualquier sistema en el que el tiempo en el que se produce la salida es significativo. Esto generalmente es porque la entrada corresponde a algún movimiento en el mundo físico, y la salida está relacionada con dicho movimiento. El intervalo entre el tiempo de entrada y el de salida debe ser lo suficientemente pequeño para una temporalidad aceptable.”

¿Qué se entiende por Sistema de Tiempo Real?

Un sistema de tiempo real es un sistema informático en el que es significativo el tiempo en el que se producen sus acciones. No es suficiente que las acciones del sistema sean correctas lógicamente, sino que, además, deben producirse dentro de un intervalo de tiempo

determinado. Esto es debido a que el sistema está conectado a un proceso externo del que recibe estímulos a los que debe responder con suficiente rapidez para evitar que evolucione a un estado indeseable.

Existen sistemas que requieren una elevada exactitud en su tiempo de respuesta, una demora que sobrepase el tiempo límite especificado puede tener consecuencias fatales, por ejemplo, el sistema de navegación de un avión. Estos se nombran sistemas de tiempo real estrictos (*Hard Real Time Systems*).

Mientras que los sistemas con restricciones de tiempo en las que una respuesta tardía no produce graves daños pero si un deterioro del funcionamiento global se nombran sistemas de tiempo real no estrictos (*Soft Real Time Systems*).

1.2.2 Software para trabajo en tiempo real

Existen varios software para el trabajo y desarrollo de sistemas en tiempo real; los que presentan herramientas y bibliotecas que permiten su uso para lograr este fin. Las mismas están desarrolladas para interactuar activamente en un entorno con dinámica conocida; mediante estas se pueden relacionar las entradas y salidas de cualquier sistema, así como establecer restricciones temporales, para darle un correcto funcionamiento de acuerdo con los conceptos de predicción, estabilidad, control y alcance. Seguidamente se describen algunos de estos software.

LabVIEW

El software LabVIEW (*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 et al., 2002).

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales o VI. Su origen proviene del control de instrumentos. Este presenta varias facilidades tales como: el manejo de interfaces de comunicaciones, capacidad de interactuar con otros lenguajes y aplicaciones, herramientas gráficas y textuales para el procesado 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 (*Field*

Programmable Gate Array) para control o validación y herramientas para el trabajo en tiempo real. Por lo antes mencionado el LabVIEW propicia el desarrollo de sistemas para 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.

Algunas de sus principales características son, su facilidad de uso y rapidez de programación. Con LabVIEW pueden crearse programas de miles de VI, equivalentes a millones de páginas de código texto para aplicaciones complejas si se realizaran en otro lenguaje, programas de automatizaciones de decenas de miles de puntos de entradas-salidas (Hernández and Ledesma, 2010).

MATLAB

MATLAB es una herramienta de software matemático que brinda un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M), orientado al cálculo técnico, la visualización de resultados y la codificación de programas. Se encuentra disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux. 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, 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. Ha evolucionado desde 1970 a través de la atención de las necesidades de sus principales usuarios, tanto en ámbitos académicos como empresariales (Arellano, 2013).

MATLAB contiene una aplicación para realizar simulaciones en tiempo real, el *ToolBox Real Time Windows Target*, que ejecuta modelos hechos en Simulink. Esta herramienta posibilita crear aplicaciones de control y simulaciones en tiempo real de plantas físicas; para ello es

necesario utilizar una tarjeta de adquisición de datos DAQ que interactúe entre Simulink y el dispositivo o proceso que se quiera controlar (Luke et al., 2005).

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, 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,” 2018).

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, 2018).

Este entorno de desarrollo integrado es compatible con Windows y con Unix. Las bibliotecas especializadas en prueba, medida y control de Nacional Instruments son también accesibles para la programación en C++, C# o Visual Basic gracias a *Measurement Studio*.

RTLinux

RTLinux es un sistema operativo de tiempo real multitarea que ejecuta Linux como un *thread* (hilo de ejecución) de menos prioridad que las tareas de tiempo real (Opdenacker et al., 2010). 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. RTLinux proporciona un entorno de ejecución bajo el *kernel* de Linux, como consecuencia de esto, las tareas de tiempo real no pueden usar los servicios de Linux (Zhou Qingguo et al., 2008) . Para disminuir este problema, el sistema de tiempo real se ha dividido en dos partes: la capa de tiempo real estricto que ejecuta encima de RTLinux, y la capa de tiempo real flexible, que ejecuta como un proceso normal de Linux.

La propuesta de dos capas es un método útil para proporcionar tiempo real estricto mientras se mantienen las características de escritorio de un sistema operativo. La separación del

kernel de tiempo real del mecanismo del *kernel* de Linux de propósito general permite optimizar ambos sistemas de forma independiente (Proctor and Shackelford, 2002).

1.3 Simulación

Gracias al avance tecnológico el uso de la simulación por computadoras brinda la posibilidad de entrenar y preparar a las personas para la correcta manipulación de equipos, para la optimización de procesos industriales o el estudio de diversos problemas de la ciencia y la ingeniería. Su importancia radica en que permite cometer errores cuando se efectúa y aporta valiosas lecciones a los usuarios en un ambiente seguro.

En el campo de la educación, la simulación por computadora desempeña un papel relevante ya que es usada para evaluar el aprendizaje, resolver problemas de habilidades y disposición de los estudiantes. Permite de forma óptima repetir tantas veces como se desee o requiera el proceso en estudio; constituye un método de enseñanza más del que disponen los profesores para transmitir conocimientos.

Según (Azarang and Garcia, 1996) la simulación es el desarrollo de un modelo lógico-matemático de un sistema, para imitar el funcionamiento de un proceso de la vida real o de un sistema a través del tiempo. Involucra la generación de una historia artificial de un sistema; la observación de esta historia mediante la manipulación experimental posibilita la deducción de las características operacionales del mismo. En la definición anterior se evidencian dos pasos básicos de una simulación: desarrollo del modelo y experimentación. El desarrollo del modelo incluye la formulación de ecuaciones lógicas representativas del sistema y la creación de un programa computacional. Después de validado el modelo del sistema, se ejecuta la segunda fase de un estudio de simulación, experimentar con el modelo para determinar cómo responde el sistema a cambios en los valores de las variables de entrada.

1.3.1 Breve historia de la simulación

En la revista (Lander Simulation & Training Solutions, 2018) aparece un artículo titulado “Historia de la simulación” donde se muestran los momentos más relevantes del surgimiento y desarrollo de la simulación. Se podría considerar que la misma nace en 1777 con el planteamiento del problema “La aguja de Buffon”, un método matemático sencillo para ir

aproximando el valor del número π a partir de sucesivos intentos. En 1812 Laplace mejoró y corrigió la solución de Buffon y desde entonces se conoce como solución Buffon-Laplace.

En la época contemporánea la construcción de las primeras computadoras de propósito general como el ENIAC y el trabajo de Stanislaw Ulam, John Von Neumann y otros científicos para solucionar problemas de difusión de neutrones en el diseño y desarrollo de la bomba de hidrógeno impulsaron el progreso de la simulación. En 1960, Keith Douglas Tocher desarrolló un programa de simulación general cuya principal tarea era la de simular el funcionamiento de una planta de producción donde las máquinas ciclaban por estados: Ocupado, Esperando, No disponible y Fallo; de manera que las simulaciones en los cambios de estado de las máquinas marcaran el estado definitivo de la producción de la planta. Este trabajo condujo a la creación del primer libro sobre simulación: *The Art of Simulation* (1963).

Entre 1960 y 1961 IBM desarrolló el Sistema de Simulación de propósito general o *General Purpose Simulation System* (GPSS) que fue diseñado para realizar simulaciones de teleprocesos, tales como: control de tráfico urbano, gestión de llamadas telefónicas y reservas de billetes de avión. Este sistema fue popularizado como el lenguaje de simulación más usado en aquel entonces, sobre todo por su facilidad de uso. El *Royal Norwegian Computing Center* inició en 1961 el desarrollo del programa SIMULA con ayuda de UNIVAC, fruto de esta investigación se creó SIMULA I, probablemente el lenguaje de programación más importante de toda la historia. La fundación del WSC (*Winter Simulation Conference*) en 1967 constituyó un hecho significativo en el vertiginoso avance de la simulación. En este lugar, desde esa fecha y hasta la actualidad, se archivan los lenguajes de simulación y aplicaciones derivadas, siendo el referente en los avances en el campo de los sistemas de simulación.

1.3.2 Simulador de procesos

El uso de los simuladores de procesos a nivel industrial y educativo se ha generalizado, pues permiten el estudio, diseño, análisis y optimización de sistemas y procesos industriales. Estos constituyen un ahorro de tiempo y recursos en el análisis y evaluación de procesos a nivel industrial, que por su complejidad serían imposibles de implementar en un laboratorio para pruebas en tiempo real.

El uso de los simuladores en las industrias se originó durante la crisis energética del petróleo en los años 70, en la que estas se vieron obligadas a optimizar sus procesos, y a reducir los costos de inversión (González, 2010). Inicialmente se desarrollaron en varias universidades de los Estados Unidos y en algunas industrias varios programas de simulación con estructuras rígidas para la introducción de las variables de entrada y con resultados de difícil comprensión para los operadores. Con el avance de las computadoras y de la informática en general, distintas empresas especializadas en desarrollo de software crearon programas para la simulación de procesos industriales con entornos más amigables, entre las que destacan Hysys, AspenPlus, CHEMCAD, ProModel, entre otros.

Los simuladores de procesos según (Scenna et al., 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.

El empleo de los simuladores 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 and Quesada, 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, 2014).

Existen varias 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. Estos tienen como objetivo comprender el comportamiento del sistema y evaluar de esta manera diferentes tipos de estrategias para su mejor operación, por medio de elementos matemáticos y tecnológicos.

1.4 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.

1.4.1 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, 2010).

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.

1.4.2 Controlador PID

El controlador es un componente imprescindible dentro de los lazos de control, este compara el valor efectivo de salida de una planta con el valor deseado o *setpoint*, determina la desviación y produce una señal de control que la reduce a cero o a un valor pequeño (Bahón and Giner, 2004).

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, 2010). Estos tienen como característica en común la toma 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, temperatura, concentración, 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 and 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, 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, 2010) 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:

γ : 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 1.2, esta forma es denominada control PI-D.

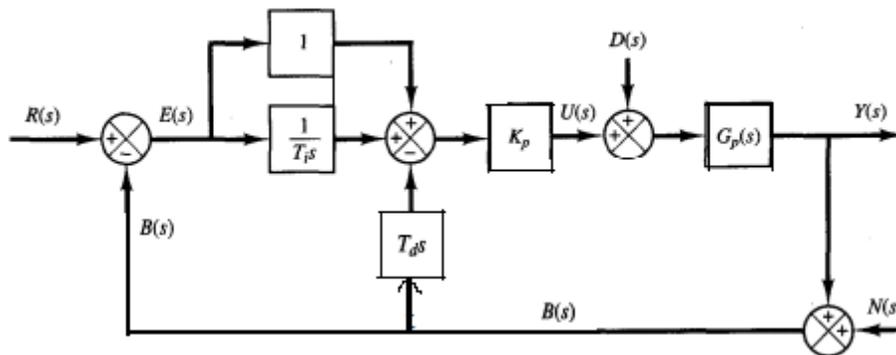


Figura 1.2 Control PI-D

Dónde:

$R(s)$: Entrada de referencia

$U(s)$: Señal manipulada

$D(s)$: Disturbio

$Y(s)$: Salida

$N(s)$: Ruido

$B(s)$: Señal medida

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, 2010) 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 1.3, este es denominado control I-PD.

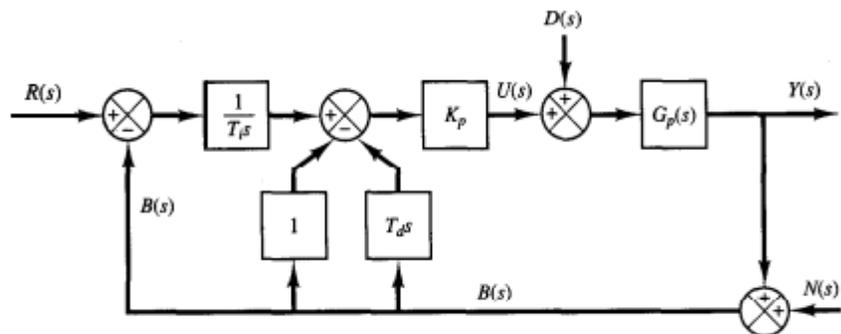


Figura 1.3 Control I-PD

1.5 Consideraciones prácticas en la implementación de algoritmos de control en procesos simulados

Cuando se implementa algún algoritmo de control en un proceso industrial existen varios aspectos importantes que deben ser considerados, en caso contrario, el sistema de control podría proporcionar malos resultados y funcionar insatisfactoriamente. De igual manera, cuando se simula un proceso estos criterios deben ser tenidos en cuenta para que dicha simulación sea lo más certera posible. Según (Ballesteros and Hernández, 2015) algunos de estos son: la transferencia suave entre los modos manual y automático, el filtrado de las señales de *setpoint* y de mando, la selección del periodo de muestreo, los convertidores A/D y D/A, el retardo de transporte, la variación de la carga, la compensación anti-windup, las saturaciones en la señal de mando, el actuador y el proceso, la zona muerta, las unidades ingenieriles y los niveles de ruido. Para realizar un mejor análisis de los mismos, han sido divididos en dos grupos, los propios del controlador PID y las consideraciones en la simulación del proceso.

Consideraciones del controlador PID

- Modos manual y automático: Todos los controladores pueden correr en dos modos: manual o automático. En el modo manual, la salida del controlador es manipulada directamente por el operador mientras que en el modo automático se ejecuta el algoritmo de control PID.
- Transferencia suave entre los modos manual y automático: Cuando ocurren cambios entre los modos manual y automático, se producen saltos indeseados. Cuando el sistema se encuentra en modo manual, el algoritmo de control produce una señal que puede ser diferente a la generada manualmente. Es necesario que las dos salidas coincidan en el instante de tiempo en el que conmutan; 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, lo que garantiza una transferencia suave (Rojas, 2007).
- Selección del periodo de muestreo: Si el periodo de muestreo es seleccionado adecuadamente el sistema se comportará similar a uno analógico. El tiempo en el que produce la lectura de la variable, su procesamiento y cálculo de la señal de control

debe ser menor que el periodo de muestreo. Uno de los criterios para su selección establece que la frecuencia de muestreo debe ser la mitad que la frecuencia de corte, esta última es llamada la frecuencia de Nyquist (Ballesteros and Hernández, 2015).

- Filtrado del *setpoint*: A la señal de entrada es necesario realizar lo que se llama procesamiento primario de la información que se realiza en los elementos de cómputo más cercanos a la variable física, de manera tal que generalmente se obtiene el valor en su unidad física. Es importante que la entrada del sistema de control no se vea afectada por algún ruido, por lo que se hace necesario, en caso de que existan, atenuarlos.
- Filtrado de la señal de mando: Posibilita el filtraje de la variable de control con el objetivo de evitar variaciones bruscas en esta señal.
- Convertidores A/D y D/A: El proceso de la conversión analógica a digital consiste en muestrear la señal analógica en intervalos regulares de tiempo. Cada muestra de voltaje analógico es convertida a un valor digital equivalente. Uno de los métodos para convertir una señal digital en una analógica es utilizar un circuito formado por un amplificador operacional y una red de resistencias en la entrada de este (Morris, 2001).
- Compensación *anti-windup*: Es un algoritmo que se implementa para compensar los efectos adversos provocados por la saturación de los actuadores o la sustitución de los controladores debido a algún cambio en las condiciones de operación. Un controlador con *windup* entrega una señal de mando, pero esta es diferente a la que recibe el proceso, debido a las limitaciones físicas presentes en los actuadores y a los cambios de puntos de operación del sistema de control (Acuña and Ríos, 2008). Existen varios métodos para evitar o reducir la saturación del término integral, entre los que se encuentran: limitación del término integral, integración condicional y seguimiento integral.
- Saturación en el mando: Es provocada por el aumento de la acción integral en los controladores PI y PID, puede ocasionar la saturación de los actuadores, lo que produce una no linealidad en el sistema de control y un funcionamiento no deseado. Está estrechamente vinculada con el efecto *windup*.

- Banda proporcional: Siempre se especifica como porcentaje, es la inversa de la ganancia del controlador, esto se cumple cuando el valor deseado y la señal de mando están expresados en la misma unidad de medida.

Consideraciones en la simulación del proceso

- Retardo de transporte: Su efecto ocasiona demoras en el proceso, a este fenómeno se le conoce comúnmente como tiempo muerto (Ogata, 2010).
- Zona muerta: Es el rango en el que las señales de entrada no provocan ningún efecto en la salida del sistema. En motores de corriente directa se relaciona con el voltaje necesario que hay que aplicar para iniciar el movimiento partiendo del reposo, debido a la fricción estática. La zona muerta causa efectos negativos en el posicionamiento y movimiento a bajas velocidades, por lo que es necesario incluir en el lazo de control un algoritmo que se anticipe a sus efectos y ayude a corregirlos (Pichardo, 2008).
- Niveles de ruido: Los niveles de ruido son señales no deseadas que contaminan las señales del sistema y pueden provocar inestabilidad en el mismo.
- Saturación del actuador: El actuador o elemento de acción final es la parte del dispositivo de control que produce un cambio físico que afecta directamente la dinámica de la planta. En todos los sistemas de control, los actuadores y las señales de salida de los controladores tienen un rango limitado de operación. Una señal de control fuerte disminuye la vida útil de los actuadores del proceso. El método más usado para evitar esta saturación es la adición de limitadores a la salida de los reguladores para disminuir este efecto nocivo (Fuente, 2010).
- Saturación del proceso: Como parte de la dinámica del proceso también pueden existir no linealidades como es el caso de saturación, esta viene dada por los límites del comportamiento de sus partes integrantes y que por lo tanto intervienen dentro del modelo limitando consecuentemente los parámetros de este último. Por ejemplo, en un motor de corriente directa el parámetro velocidad tiene un límite y por lo tanto saturación debido a su propia construcción.
- Variación de la carga: Es una variación en porcentaje de la señal de mando o de los parámetros del proceso.

1.6 Antecedentes de software para el control de procesos simulados

Durante el análisis bibliográfico realizado se han consultado artículos donde se aborda el tema del uso de los simuladores y los entrenadores de procesos. Estos artículos sirven de antecedentes a este trabajo, de los cuales se toman aspectos importantes.

Diseño de un simulador digital de procesos

En este trabajo el autor (Zurita, 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 (Pérez, 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, 2011) diseña un circuito neumático en el que se controla la presión de un depósito, para esto utiliza 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.

Comparación del Rendimiento de Controladores en el Entrenador Industrial DAC603

En este trabajo los autores (Aguirre et al., 2010) muestran el diseño e implementación de algoritmos de control en la unidad de entrenamiento de control e instrumentación industrial DAC603. El diseño de los algoritmos está basado en la teoría de control y su programación se hace en un lenguaje de alto nivel. La obtención del modelo matemático del sistema se hizo mediante identificación, con el ToolBox de MATLAB. Se estudia el desempeño de los controladores implementados mediante el uso de un índice de rendimiento.

Entrenador digital de lazo de control con consideraciones prácticas reales

Este trabajo de diploma se realizó para el laboratorio de computación de la carrera de Ingeniería Automática por el autor (Rodríguez, 2016) 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 realizó el diseño del mismo en el software LabVIEW. Para la valoración de los resultados se analizaron 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 simuló con los valores de configuración inicial y posteriormente se introdujeron algunas consideraciones prácticas para interpretar el efecto en la salida del sistema.

Análisis de los casos seleccionados

En la tesis de maestría de (Zurita, 2014) se utiliza un microcontrolador para la simulación, lo que sería innecesario porque este proceso puede ser simulado en un software destinado para este fin. En los trabajos de (Pérez, 2008) y (Nuñez, 2011) se tiene como inconveniente el uso de PLC para el diseño de los simuladores, que son dispositivos caros que no están al alcance de muchas personas. En el trabajo de (Nuñez, 2011) se utiliza un acumulador neumático y una electroválvula proporcional, que son otros inconvenientes a tener en cuenta desde el punto de vista económico. En la investigación de (Aguirre et al., 2010) existe la limitante del empleo de un lenguaje de programación de alto nivel para diseñar los algoritmos de control, esto hace que el trabajo sea más engorroso, en su lugar se puede utilizar algún software como MATLAB o LabVIEW que facilitan su desarrollo. En el trabajo de (Rodríguez, 2016) se incluyen pocas consideraciones prácticas, por lo que el software desarrollado no es tan cercano a la realidad como pudiera serlo.

1.7 Consideraciones finales del capítulo

El empleo de los simuladores de sistemas de control es una herramienta de aprendizaje viable para cualquier nivel de enseñanza. A través de la simulación y la realización de prácticas en tiempo real se puede emular situaciones que un ingeniero puede encontrar en cualquier industria. Los estudiantes pueden aprender valiosas lecciones y ejercitarse en un ambiente virtual seguro.

La implementación de las consideraciones prácticas permite que los procesos simulados se aproximen en gran medida a los reales, lo que favorece aún más el uso de la simulación por computadora para el estudio de los procesos industriales.

El software LabVIEW reúne un conjunto de características, tales como, las herramientas para el trabajo en tiempo real y la utilización de un lenguaje de programación visual gráfico que facilita su explotación por los usuarios. Por lo antes mencionado, su empleo en la simulación de procesos es una decisión totalmente acertada.

CAPÍTULO 2. Diseño del software para el control digital del SAD-100

El presente capítulo está dedicado a exponer todo lo concerniente al diseño del software para el control automático digital del servocontrol SAD-100 (SCD SAD-100); se realiza una caracterización de este sistema con el objetivo de hallar su función transferencial. Las principales características de la tarjeta de adquisición de datos NI USB-6212 son mostradas. Se efectúa una descripción de las funciones y ejemplos de LabVIEW que fueron empleados en la creación del software. La forma en la que se estructuró el mismo es explicada, y específicamente lo relacionado con el proceso y el controlador. Se muestran las partes fundamentales de la programación de dicho software.

2.1 Caracterización del SAD-100

En los laboratorios de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas existe una maqueta Alecop SAD 100 que se utiliza para impartir la asignatura Herramientas de Software para el Control. Este servosistema, que constituye la planta en estudio, es un entrenador para prácticas que permite realizar control de velocidad o posición de un motor de corriente directa (CD). Dicha maqueta tiene dos partes que la conforman según (Alecop, 1986): la primera es un eje extendido para montaje directo del motor (1), el freno magnético (2), discos de inercia o masa (3), encoder (4), potenciómetro (5) y tacodinamo (6), como se muestra en la figura 2.1. De estos componentes los utilizados para el desarrollo del presente trabajo son: el motor CD, la masa que se emplea para cambiar el momento de inercia del motor, y el tacogenerador.

El motor utiliza un amplificador que lo antecede, que permite que la señal manipulada se mantenga aproximadamente entre +100 y -100 V.

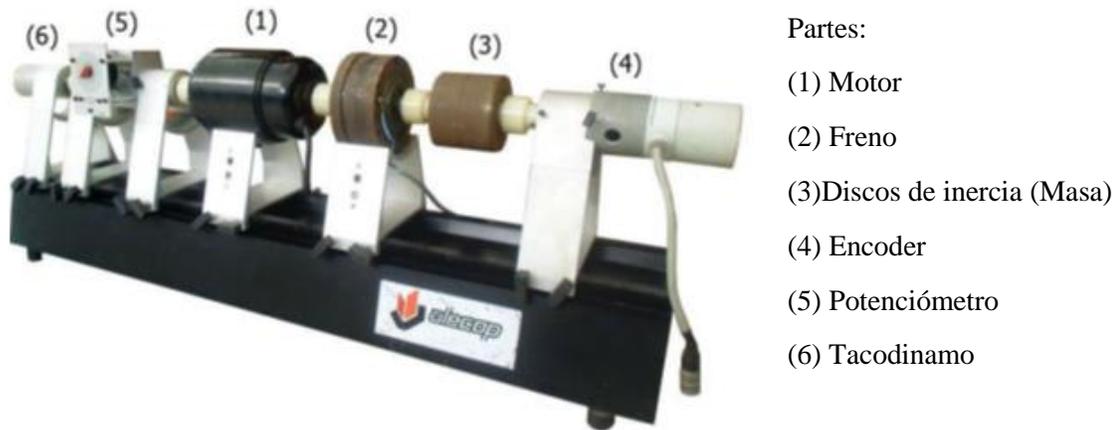


Figura. 2.1 Eje extendido de la maqueta ALECOP SAD-100

La otra parte que compone a la maqueta es el panel actuador mostrado en la figura 2.2, que posee varios módulos o bloques, entre los que destacan: la fuente de potencia (7), el acondicionamiento del sensor de velocidad (13), los acondicionamientos de los sensores de posición (9 y 10) que son el potenciómetro analógico y el encoder, los generadores del valor deseado (11), el accionamiento del freno magnético (13), el regulador PI (14). Estos módulos pueden ser conectados según sea necesario. Los empleados en este trabajo son el bloque 7, el 11 y el 13.

El bloque 7 tiene dos funciones, una es como fuente de alimentación, encargada de suministrar todos los niveles de voltaje (+15V, -15V, +5V) necesarios para la operación de la maqueta, mientras que la otra es como driver de potencia del motor. Este último es un Driver DC-DC, el cual consta de una electrónica adecuada para entregarle al motor la energía suficiente para su accionamiento. Al mismo se le suministra una referencia entre 10 V y -10 V, donde 0 V corresponde al reposo del motor, 10 V a la velocidad máxima en el sentido directo y -10 V la velocidad máxima en sentido inverso.

El bloque 11 está formado por amplificadores de instrumentación que tienen una alta impedancia de entrada, lo que permite que se conecten dispositivos como la tarjeta de adquisición de datos y no se produzcan sobrecargas que dañen a los mismos. También posee potenciómetros multivoltas para variar los valores de resistencia. Este bloque tiene como función principal generar voltajes que oscilan entre -10 y 10 V para ser utilizados como valores deseados o *setpoint* en lazos de control.

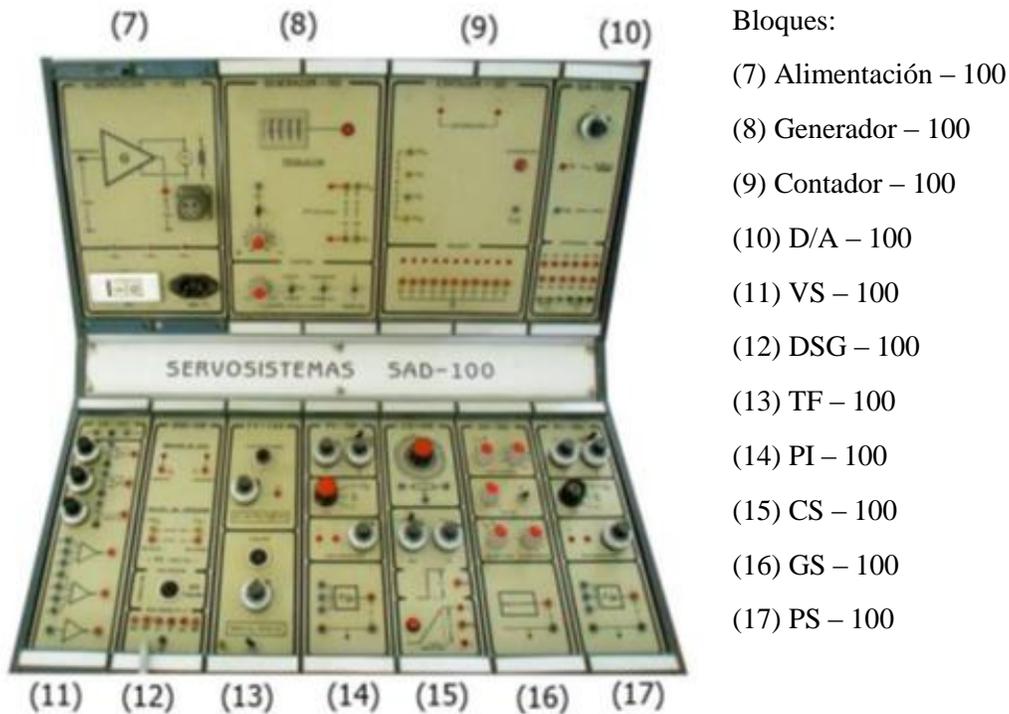


Figura. 2.2 Panel actuador de la maqueta ALECOP SAD-100

El bloque 13 se subdivide en dos bloques:

- *Accionamiento del freno*: le suministra al freno magnético una diferencia de potencial, la cual puede ser variada mediante el potenciómetro en la parte inferior de este bloque, produciendo un par de frenado proporcional a este voltaje.
- *Acondicionamiento del Tacodinamo*: consiste en una etapa de amplificación que brinda un primer acondicionamiento del Tacodinamo. Mediante el potenciómetro en la parte superior del bloque se puede variar la ganancia del Tacodinamo consiguiéndose niveles absolutos de hasta 10 V a máxima velocidad de 2 000 revoluciones por minuto (rpm).

Función transferencial del conjunto actuador-motor

La función de transferencia del sistema se obtuvo mediante identificación experimental a través de la realización de un conjunto de pruebas a la maqueta, en las que se obtuvieron los datos necesarios para hallar esta función, como su característica estática. La primera de estas pruebas fue concebida con la ayuda de un multímetro, con el objetivo de ajustar el potenciómetro multivoltas de la parte superior del módulo 13; para ello se montó el circuito

de la figura 2.3 que corresponde a un lazo abierto de control y se excitó al actuador con un voltaje de 6.98 V, después se varió el potenciómetro hasta que, para un valor de $K_p = 7.14$ vueltas, el tacogenerador entregó 6.98 V. La ecuación 2.1 describe a este último.

$$V_T = 0.7K_p Vel \quad (2.1)$$

Dónde:

V_T : Voltaje entregado por el Tacodinamo en mV

K_p : Constante del potenciómetro

Vel: Velocidad del motor en rpm

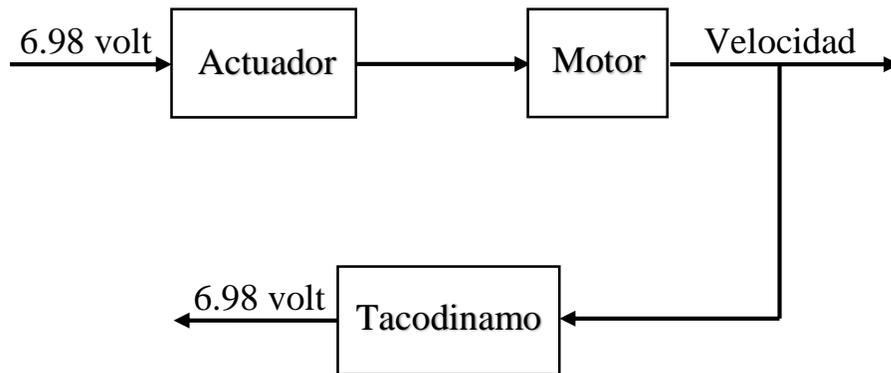


Figura. 2.3 Lazo abierto de control que ilustra la forma en la que fueron realizadas las pruebas al servosistema ALECOP SAD-100

Para encontrar la característica estática del sistema actuador-motor se realizó una segunda prueba, en la cual se empleó un multímetro para medir los voltajes de alimentación y los entregados por el Tacodinamo, posteriormente se obtuvieron los valores de velocidad correspondientes a cada voltaje suministrado en la alimentación, mediante la ecuación 2.2, que se logra a partir de un despeje de la ecuación 2.1. Con las mediciones realizadas se obtuvo la tabla de la característica estática del motor (ver anexo I), mediante esta y el uso del software MATLAB fue posible confeccionar la gráfica de la figura 2.4, que muestra la característica entrada contra salida del sistema, en la cual se puede observar claramente la zona muerta y la no linealidad que este posee. La zona muerta se encuentra aproximadamente entre +1.7 y

-1.7 V, mientras que la zona lineal está entre 250 y 2000 rpm en un sentido de giro y entre -2000 y -250 en sentido inverso.

$$Vel = \frac{V_T * 1000}{4.998} \quad (2.2)$$

La ganancia estática (K), que es relación entre la entrada y la salida en régimen permanente del conjunto actuador-motor, es decir, la velocidad en rpm que el mismo alcanza en lazo abierto en función del voltaje que se le suministra al actuador, se determinó a través de la relación $\Delta\text{Salida}/\Delta\text{Entrada}$. Para ello se tomaron los valores de la característica estática, pertenecientes a su zona lineal, de velocidad del motor de 803.1 y 1698.9 rpm y sus correspondientes voltajes de alimentación, 4 y 7.49 V respectivamente. Se obtuvo un valor de $K = 257 \text{ rpm/V}$.

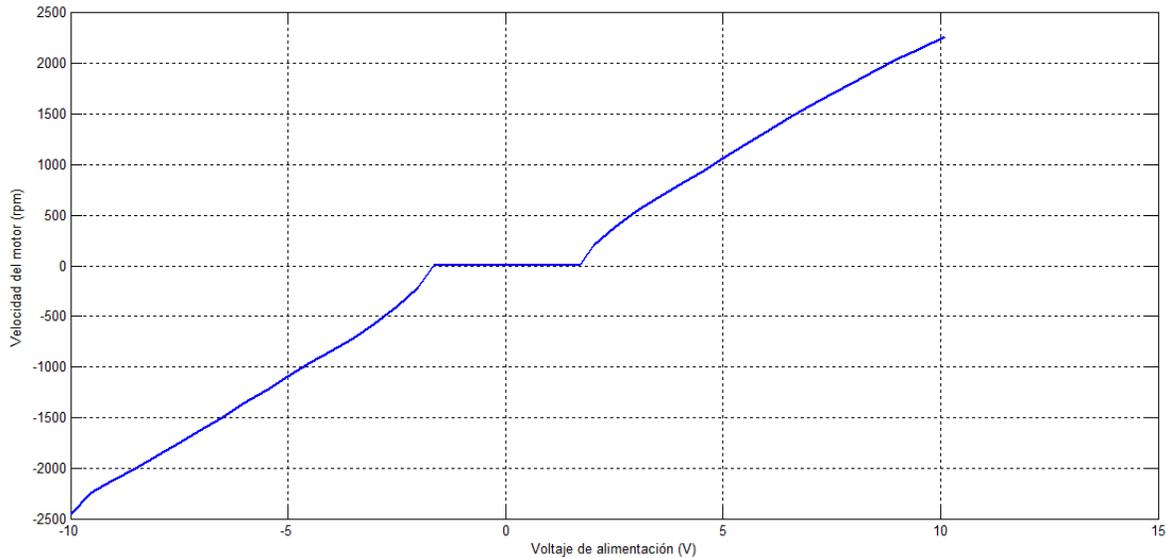


Figura. 2.4 Característica estática del conjunto actuador-motor

Con el fin de lograr un análisis más completo del servosistema SAD-100 se obtuvo la característica dinámica del mismo, mediante el software creado en la asignatura Herramientas de Software para el Control en el curso escolar 2017/2018; con el que se realizó una identificación del sistema. Las señales provenientes del tacogenerador son utilizadas para la medición de la velocidad. En la figura 2.5 se ilustra la conexión de la maqueta ALECOP con la tarjeta de adquisición de datos y con el resto de los componentes:

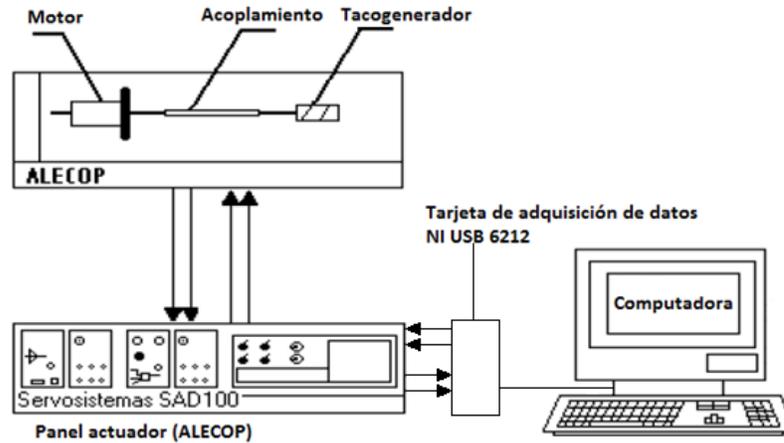


Figura. 2.5 Acoplamiento del Sistema ALECOP SAD-100

La identificación se efectuó para obtener la constante de tiempo del sistema y para corroborar lo realizado previamente para hallar el valor de K , pero en este caso la ganancia resultante fue de 212, valor que se aleja del anterior, no obstante se escogió puesto que dicha identificación ofrece resultados más fiables que las mediciones con el multímetro, ya que el tacogenerador introdujo mucho ruido, lo que provocó inexactitud en estas mediciones.

Para encontrar la característica dinámica, mostrada en la figura 2.6, que resultó ser un modelo de primer orden sin retardo, se ejecutó el programa de identificación en 5 ocasiones para lograr un mejor resultado. Mediante este programa se le dio un paso de 3 a 7 V a la entrada del servomotor, con una frecuencia de muestreo de 1000 Hz, fueron tomadas 1500 muestras por canal. Los valores de K y τ fueron promediados; además se hizo un ajuste de curva para eliminar las componentes del ruido en la medición. La constante de tiempo del sistema en estudio obtenida fue $\tau = 0.215$ seg.

En el manual de (Alecop, 1986) se especifica que el sistema actuador-motor con masa se comporta como un modelo de primer orden, descrito por la ecuación 2.3:

$$\frac{Vel(s)}{Va(s)} = \frac{K}{\tau s + 1} \quad (2.3)$$

Dónde:

Vel : Velocidad del motor en rpm

Va : Voltaje aplicado al actuador

K : Ganancia del sistema actuador-motor

τ : Constante de tiempo del sistema actuador-motor

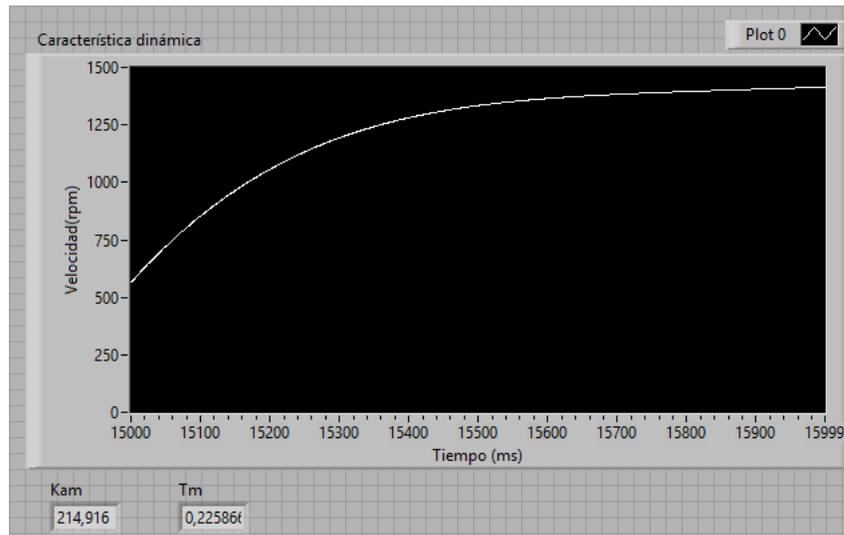


Figura. 2.6 Característica dinámica del conjunto actuador-motor

Una vez realizadas las sustituciones de K y τ se tiene que la función transferencial del sistema es la que se observa en la ecuación 2.4:

$$\frac{Vel(s)}{Va(s)} = \frac{212}{0.215s + 1} \quad (2.4)$$

2.2 Caracterización de la tarjeta de adquisición de datos NI USB-6212

La tarjeta multifuncional de adquisición de datos NI USB-6212, que se muestra en la figura 2.7, es un dispositivo que ofrece entradas y salidas analógicas y digitales, así como dos contadores de 32 bit. Posee un amplificador integrado designado para períodos de muestreo rápidos a altas velocidades de procesamiento. También presenta tecnología de transmisión de señal que le ofrece transmisión de datos bidireccional de alta velocidad de acceso directo a la memoria (DMA) a través del bus serie universal (USB). El dispositivo es ideal para aplicaciones de prueba, control y diseño, incluido el registro de datos portátil y monitorización de campo. El USB 6212 cuenta con un encapsulado mecánico liviano y está alimentado por un bus para un fácil transporte. Esta tarjeta posee un controlador NI-DAQmx que facilita y simplifica la configuración y las mediciones.

Dentro de sus características principales se encuentran (National Instruments, 2017):

- 16 entradas analógicas simples y 8 diferenciales.

- Tasa de muestreo máxima de 400 kS/s.
- Rangos de entradas analógicas de ± 0.2 V, ± 1 V, ± 5 V, ± 10 V.
- Razón de rechazo al voltaje en modo común (CMRR) con corriente directa y frecuencia de 60 Hz de 100 dB.
- Impedancia de entrada de 10 G Ω en paralelo con 100 pF.
- Corriente de entrada de ± 100 pA.
- Dos canales de salida analógicos.
- Tasa máxima de actualización de 250 kS/s por canal.
- Rango de salida analógica de ± 10 V.
- Impedancia de salida de 0.2 Ω .
- Rango de temperatura de operación de 0 °C a 45 °C.



Figura. 2.7 Tarjeta de adquisición de datos NI USB-6212

2.3 Selección de funciones y ejemplos de LabVIEW

El software LabVIEW posee una gran variedad de funciones que permiten desarrollar los instrumentos virtuales, a partir de una correcta selección de estas. Los desarrolladores han dotado a este software de numerosos ejemplos que sirven de base y facilitan crear una nueva aplicación; generalmente esto se logra modificando alguno de dichos ejemplos. Para la realización del presente trabajo se partió de un ejemplo que emplea un controlador PID, también fue necesario el uso de funciones para el trabajo con la tarjeta de adquisición de

datos con el fin de lograr la comunicación entre el VI y el servosistema. Seguidamente se detallan estos aspectos.

2.3.1 Ejemplos y funciones de PID

Para seleccionar un ejemplo que incluya un controlador PID basta con acceder a la ayuda desde el panel frontal o el diagrama de bloques y elegir la opción *Find Examples*. Dentro de la pestaña *Search* se escribe PID y se pulsa la tecla *enter*. Entonces aparece el listado de los propios de LabVIEW. En la figura 2.8 se muestra un fragmento de los mismos, donde aparece *Manual-Automatic Control with Engineering Units*, que es el ejemplo tomado como base para el desarrollo de la aplicación. Al mismo se le hicieron varias modificaciones con el objetivo de adaptarlo a las características específicas del servosistema SAD-100, más adelante en este capítulo estas transformaciones son abordadas.

General Auto PID Simulator.VI	
General PID Simulator with Alarm.vi	
General PID Simulator.vi	
General PID with Fault Protection.vi	
Manual-Automatic Control with Engineering Units.vi	
Manual-Automatic Control with Saturation.vi	
Manual-Automatic Control.vi	
Multi Channel PID.vi	
PID with Noise Plant.VI	
Setpoint Profile Simulator.vi	
Setpoint Profile with Gain Scheduling on Rising and Falling.vi	

Figura. 2.8 Fragmento de los ejemplos de PID del software LabVIEW, donde se resalta el seleccionado en este trabajo

Si se desea buscar una función de PID el camino a seguir es dar clic derecho en cualquier lugar del diagrama de bloques de LabVIEW y elegir la opción *Control Design & Simulation*, entonces aparecerá la paleta de funciones PID como se ilustra en la figura 2.9. Los primeros tres VIs son diferentes versiones de PID que pueden ser utilizados indistintamente dependiendo de las necesidades de la aplicación, los demás VIs se pueden combinar con alguno de estos VIs de PID para aumentar su funcionalidad.

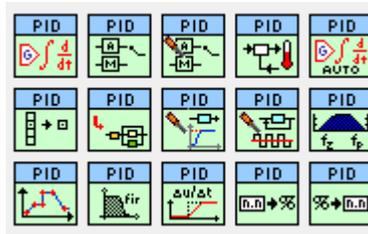


Figura. 2.9 Funciones PID propias del software LabVIEW

2.3.2 Funciones de adquisición de datos

El software LabVIEW posee un conjunto de funciones, mostradas en la figura 2.10, que permiten el uso de una tarjeta de adquisición de datos (DAQ) a través del driver NI-DAQmx, lo que ayuda al usuario de LabVIEW a desarrollar aplicaciones de instrumentación, adquisición y control. De estas funciones se seleccionaron tres de entrada para recibir los datos provenientes del Tacodinamo, las mismas son: *Create Channel*, *Read* y *Clear*, y tres de salida para enviar los datos al actuador, *Create Channel*, *Write* y *Clear*, estas funciones seleccionadas se observan en la figura 2.11. Posteriormente se realiza una descripción de cada una de dichas funciones.

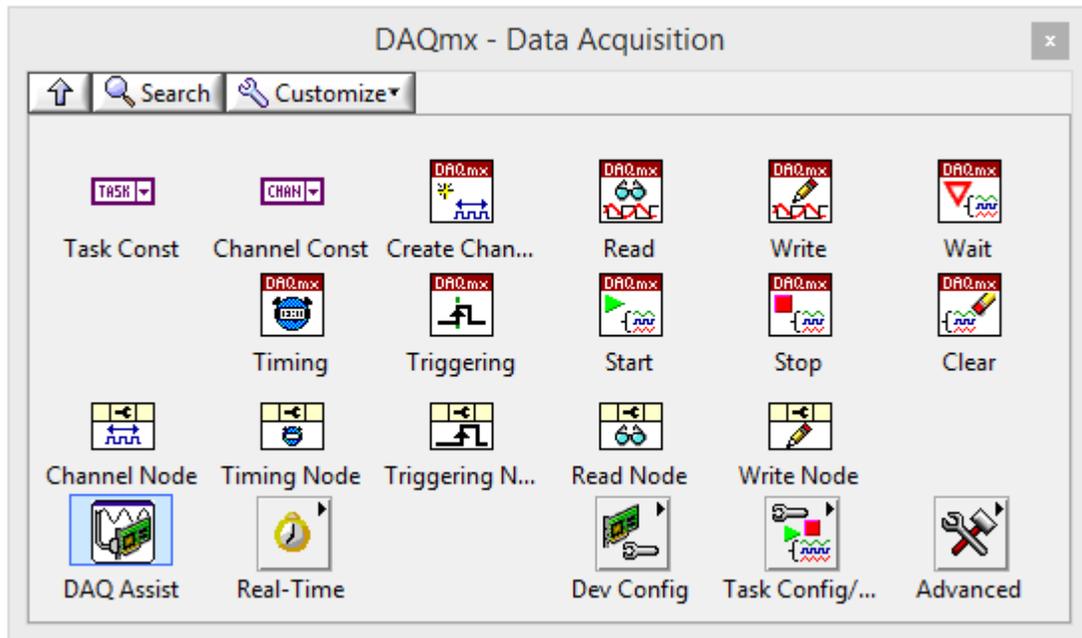


Figura. 2.10 Funciones DAQ del software LabVIEW

La función *Create Channel* crea un canal virtual o un grupo de canales virtuales y los agrega a una lista donde se almacenan estos canales (Holguín et al., 2002). Los casos de este VI

polimórfico corresponden a un tipo de canal de entrada/salida (E/S), como una entrada analógica, salida digital, o salida de contador; la medición o la generación de alguna variable, como la medición de temperatura, generación de voltaje o incluso contadores; y en algunos casos el sensor a usar. Las propiedades de canal que posee el driver DAQmx incluyen opciones de configuración adicionales.

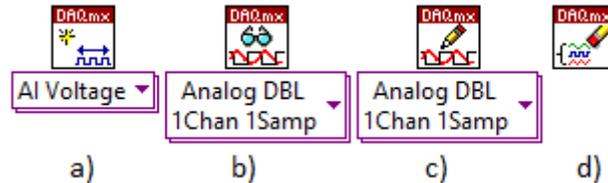


Figura. 2.11 Funciones DAQ seleccionadas: a) *Create Channel*, b) *Read*, c) *Write*, d) *Clear*

La función *Read* lee muestras procedentes de la lista o de los canales virtuales que se especifique. Las instancias de este VI polimórfico especifican en que formato se devuelven las muestras, puede leer una sola muestra o varias a la vez y puede leer de uno o múltiples canales. El driver DAQmx también tiene opciones adicionales para la configuración de las operaciones de lectura. La función *Write* escribe las muestras en la lista de canales o en los canales virtuales que se especifiquen. Las instancias de este VI polimórfico contienen el formato de las muestras que se desean escribir, además puede escribir una o múltiples muestras en uno o varios canales; al igual que las funciones anteriores el controlador DAQmx incluye opciones extras para la configuración de las operaciones de escritura. La función *Clear* borra la lista de canales, pero antes aborta dicha lista si es necesario y libera cualquier recurso que se haya reservado en ella. Después que se haya limpiado no se puede volver a utilizar, a no ser que se cree la lista nuevamente.

2.3.3 Ejemplo seleccionado

Como se planteó anteriormente el ejemplo *Manual-Automatic Control with Engineering Units* es el seleccionado, ya que demuestra como simular un lazo de control simple con el modelo general de una planta con unidades reales de ingeniería, también admite el trabajo con banda proporcional, que es el método más empleado por los operadores en las industrias. Utiliza el PID Advanced VI para cambiar entre el control manual y el automático, otra de las características que se quiere implementar en el software para el control del servosistema SAD-100.

En el panel frontal de dicho ejemplo, mostrado en la figura 2.12, aparece un indicador numérico de la variable del proceso por sus siglas en inglés (PV), un control de texto para seleccionar el tipo de unidad de entrada con la que se desee trabajar (unidades de ingeniería o banda proporcional), un control numérico para elegir el valor deseado de la variable del proceso; lo que también se logra a través de una barra vertical deslizante con una escala regulable, un control numérico y una barra horizontal deslizante que en modo manual permiten seleccionar el valor de salida del controlador y en modo automático se comportan como un indicador de este valor, un control de texto para seleccionar la estructura de PID que se quiera (académico, paralelo o serie), tres controles numéricos para elegir la ganancia proporcional y los tiempos integral y derivativo, un control de texto para escoger el tipo de unidad del Controlador PID (unidades de ingeniería o banda proporcional), un botón para cambiar entre los modos manual y automático.

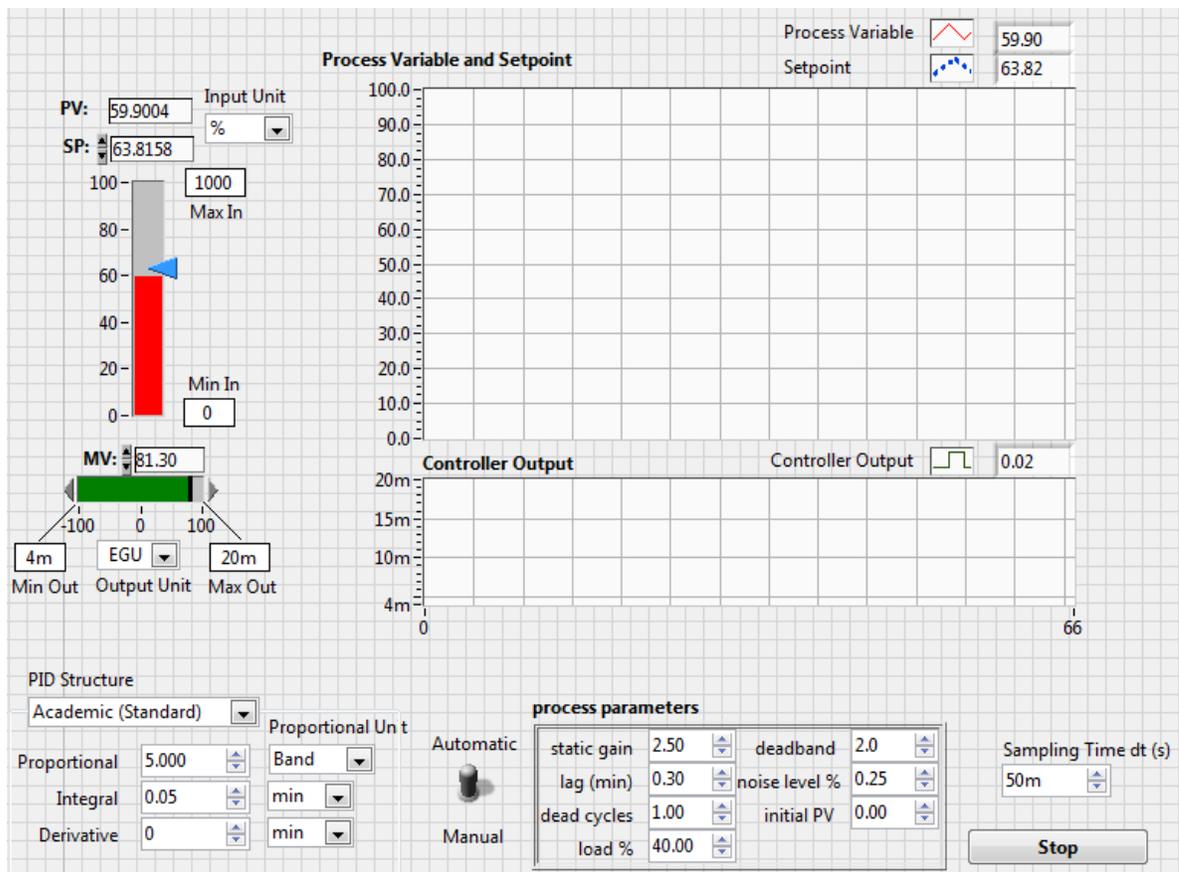


Figura. 2.12 Panel frontal del ejemplo de LabVIEW Manual-Automatic Control with Engineering Units

Además de los elementos descritos previamente en el panel frontal se encuentran un conjunto de controles numéricos que permiten variar los parámetros del proceso (ganancia, atraso, retardo de transporte, variación de la carga, zona muerta y valor inicial de la variable del proceso). Se muestran: un control numérico para la selección del período de muestreo y un botón para detener la simulación. Dicho panel se completa con los gráficos del comportamiento temporal de la variable del proceso y valor deseado y el que muestra la salida del controlador.

2.4 Diseño

El software para el control automático digital del servosistema SAD-100 ha sido diseñado con un enfoque dirigido a recrear el ambiente de trabajo que tienen los operadores en las industrias; por este motivo la interfaz de usuario se ha dividido en dos ventanas: una destinada a la configuración del controlador PID, los parámetros del proceso y la tarjeta de adquisición de datos, y la otra dedicada al control del servomotor.

La ventana de control es la que se orienta a los operadores, que son personas con conocimientos básicos de la teoría de control, acostumbradas al empleo de la banda proporcional para la supervisión de los procesos y ajenas a la interpretación de las unidades ingenieriles. Posteriormente se realiza una descripción detallada de cada una de dichas ventanas.

2.4.1 Interfaz hombre-máquina

La ventana de configuración, mostrada en la figura 2.13, posee en la parte superior, un *push button* que posibilita la selección de una de las dos formas en las que puede correr el software (real o simulado), un control numérico para elegir el período de muestreo y un bloque con tres controles numéricos que sirven para elegir las constantes de los filtros que se colocaron en el lazo de control de velocidad.

Esta ventana también posee un bloque encomendado al ajuste del regulador PID, este bloque fue tomado del ejemplo seleccionado de LabVIEW descrito previamente, al que se le agregó un *cluster* para limitar el mando. Otro de los bloques es el del proceso, que contiene sus parámetros y dos indicadores numéricos de las velocidades máxima y mínima respectivamente, que limitan la variable de salida, por lo que previenen la saturación de la

planta. El último bloque es el de la configuración de los parámetros de entrada y salida de la tarjeta de adquisición de datos empleada.

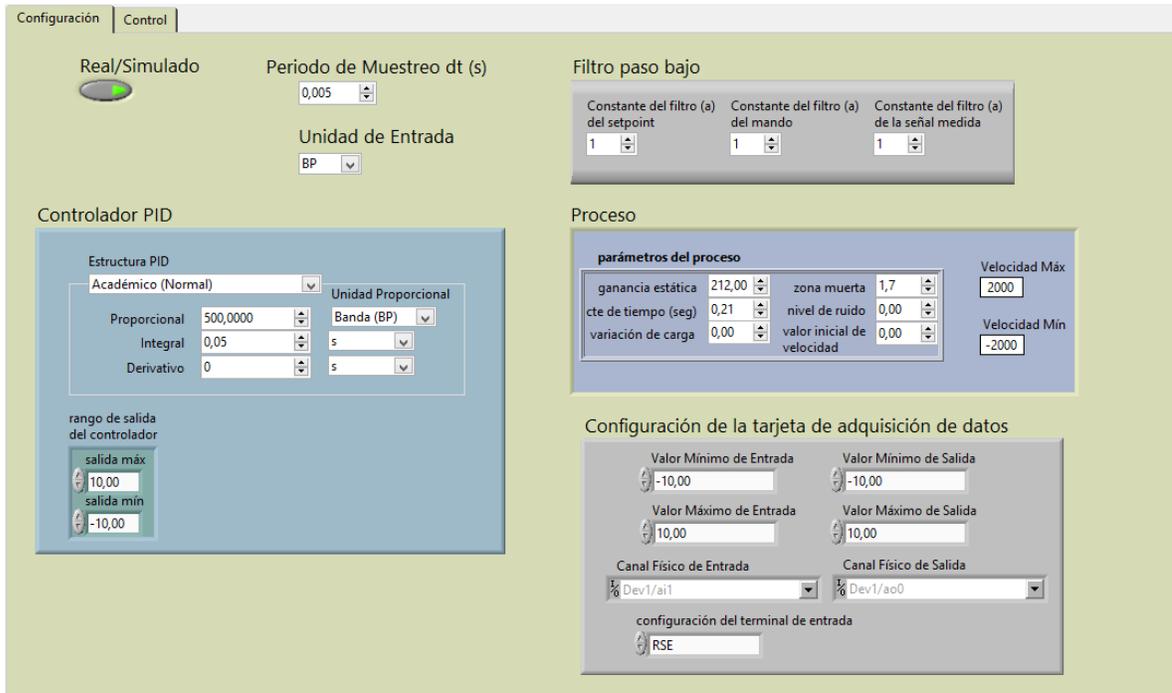


Figura. 2.13 Ventana de configuración del software SCD SAD-100

La ventana de control, mostrada en la figura 2.14, posee un indicador numérico que muestra la velocidad del motor CD en rpm, un control numérico con el valor deseado de velocidad, una barra vertical deslizable con la se puede cambiar el valor deseado, una barra horizontal deslizable que en modo automático funciona como indicador del mando y en manual posibilita realizar cambios en esta variable, un botón para detener la ejecución del software, ya sea real o simulado. Esta ventana incluye un botón para alternar entre los modos manual y automático según convenga, una gráfica que permite analizar conjuntamente el comportamiento de la velocidad del motor y el *setpoint* a medida que se muestrea la salida del sistema. Otro elemento de la ventana de control es una gráfica que muestra la señal de salida del regulador contra la cantidad de muestras tomadas por la tarjeta de adquisición de datos.

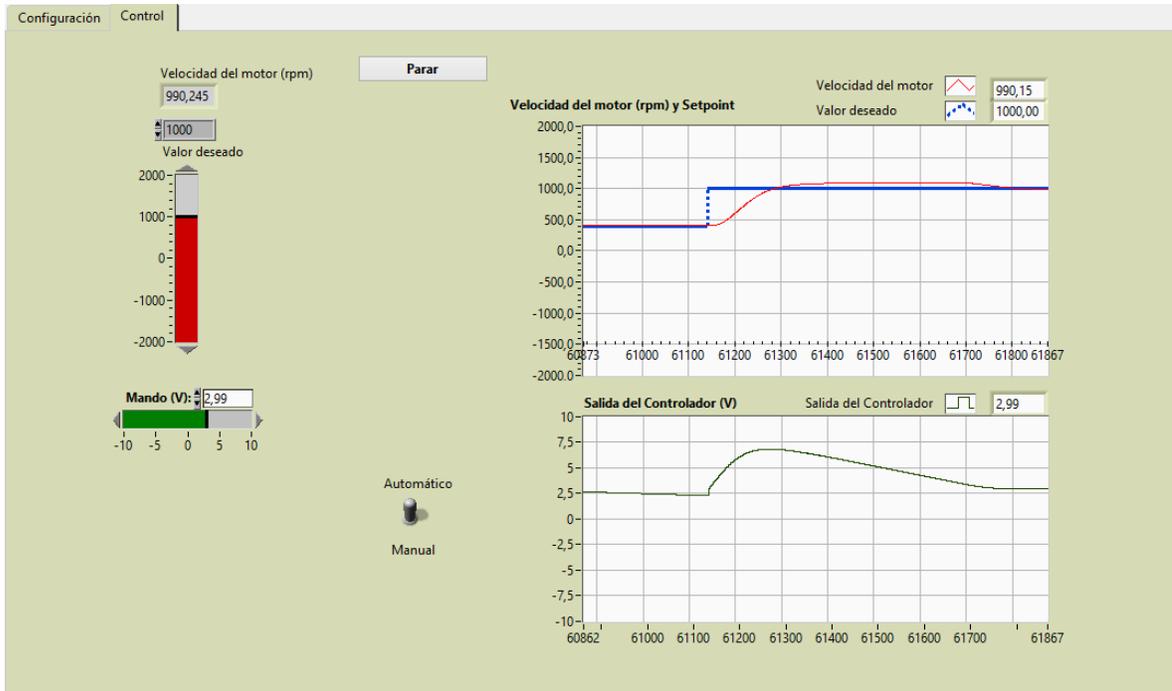


Figura. 2.14 Ventana de control del software SCD SAD-100

2.4.2 Proceso

Para realizar la simulación del proceso se tomó el instrumento virtual Plant Simulator in Engineering Units.vi perteneciente al ejemplo *Manual-Automatic Control with Engineering Units de LabVIEW* y se le hicieron varias modificaciones para adaptarlo a los requerimientos específicos del servomotor. Fue renombrado de la forma siguiente, Simulación de la planta con no linealidades (NL).vi, como se observa en la figura 2.15; donde se aprecia que se le introducen las consideraciones prácticas o parámetros del proceso tenidos en cuenta, estos son: zona muerta, saturación de la salida, nivel de ruido en la medición, selección del período de muestreo y variación de la carga. El anexo II muestra la manera en la que fueron implementados.



Figura. 2.15 Panel frontal de Simulación de la planta con NL.vi

Para representar el modelo de primer orden que caracteriza a la planta se empleó el VI Atraso adelante 10 de la figura 2.16. Para que el modelo resultante fuera el deseado se hizo cero el tiempo de adelanto, como se puede apreciar:

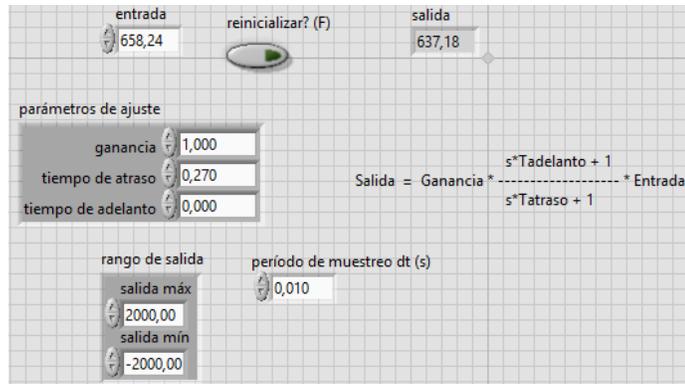


Figura. 2.16 Panel frontal de Atraso adelante 10.vi

2.4.3 Regulador

El controlador utilizado para el desarrollo del software para el control digital del servocontrol SAD-100 fue un PID, para lo cual se empleó el PID.vi, propio del LabVIEW; este instrumento virtual posee algunas consideraciones prácticas como son: la selección del período de muestreo y la compensación del efecto *windup*. Para ampliar sus prestaciones se le implementaron una serie de consideraciones adicionales, las mismas fueron: el cambio entre los modos manual y automático, el uso de la banda proporcional o las unidades de ingeniería como unidad de entrada, la saturación del mando, el filtrado del *setpoint* y de la señal de mando. EL VI modificado se muestra en la figura 2.17:

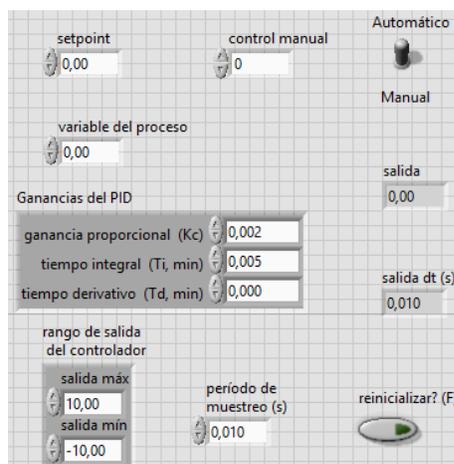


Figura. 2.17 Panel frontal de PID.vi

Este instrumento virtual posee 8 parámetros de entrada y 2 de salida, mostrados en la figura 2.18, donde se aprecia la opción de control manual y el cambio de manual a automático, entre otros:

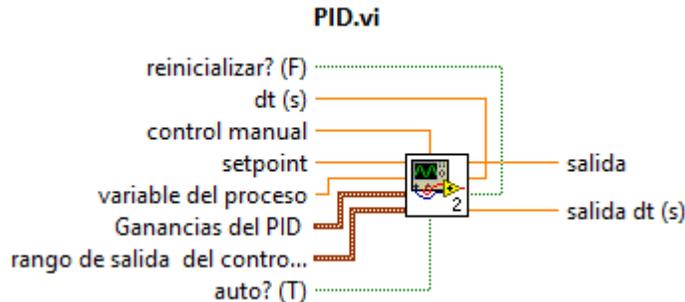


Figura. 2.18 PID.vi modificado

Para implementar el uso de la banda proporcional es necesario que la entrada y la salida del regulador tengan la misma unidad de medida, entonces se halla como el inverso de la ganancia proporcional (K_c) del controlador. Para lograrlo se convirtió el valor deseado de velocidad en rpm a voltaje. Es preciso señalar que en el ejemplo de LabVIEW *Manual-Automatic Control with Engineering Units* la banda proporcional se calcula como el inverso de la ganancia, por lo que fue necesario implementar la relación que se observa en la figura 2.19:

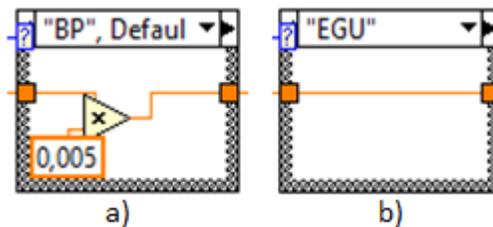


Figura. 2.19 Forma de conversión de la unidad de entrada: a) banda proporcional, b) unidades de ingeniería (EGU)

Para elegir el tipo de estructura de PID que se desee utilizar, ya sea el académico, el serie o el paralelo, se implementó el *PID Structure Conversion.vi* del software LabVIEW, el mismo fue renombrado como se muestra en la figura 2.20:

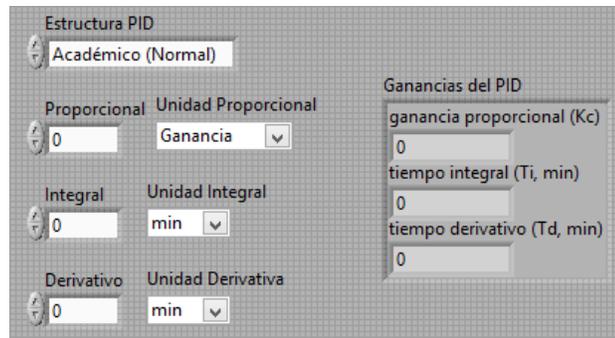


Figura. 2.20 Panel frontal de Conversor de parámetros PID.vi

Este VI posee 7 parámetros de entrada y 1 de salida, que es a su vez uno de los parámetros de entrada de PID.vi, como ilustra la figura 2.21:

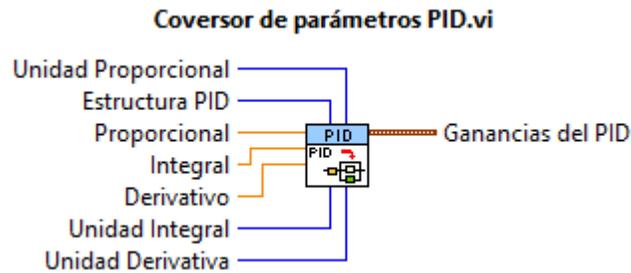


Figura. 2.21 Conversor de parámetros PID.vi

2.5 Programación en LabVIEW

El lazo de control de la velocidad del servomotor de la maqueta ALECOP SAD-100 (ver anexo III) fue programado en su totalidad dentro de un ciclo *while*, que se ejecuta cada período de muestreo e infinitamente, a menos que se oprima el botón de parar. La planta se puede incluir en este lazo de control de dos maneras, real o simulado, como se ilustra en la figura 2.22; para esto se escogió una estructura del tipo *case*.

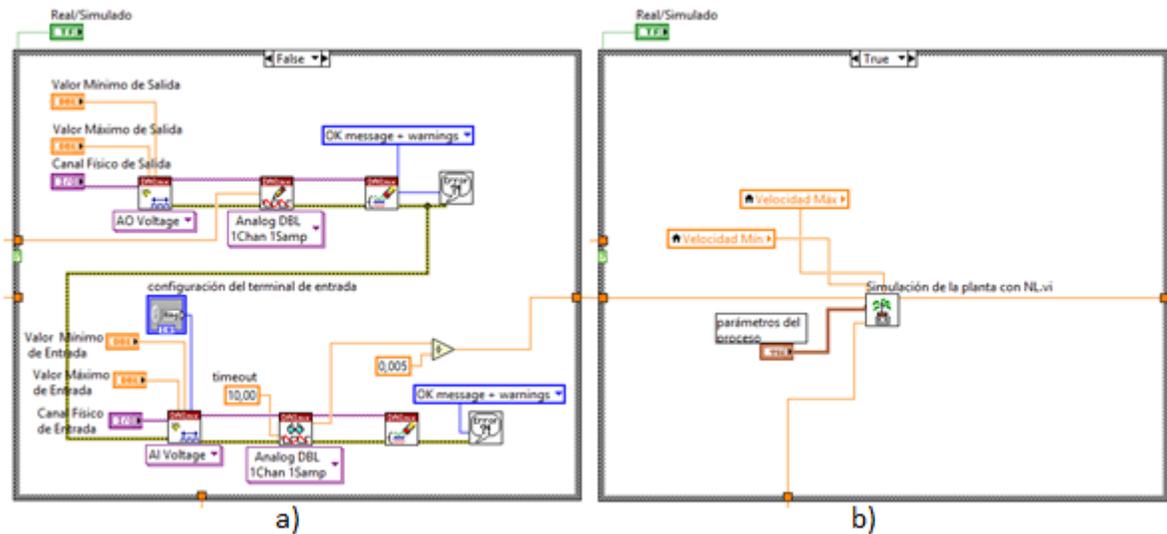


Figura. 2.22 Variantes de la planta: a) real y b) simulado

Filtro paso bajo de primer orden

En el software desarrollado se implementaron filtros exponenciales paso bajo de primer orden, puesto que el filtraje de las señales es de gran importancia para eliminar los ruidos siempre presentes. Los mismos fueron colocados en el valor deseado, en la salida del controlador y en la señal medida. Este filtro está definido por la ecuación en diferencia finita 2.5, en la figura 2.23 se muestra la forma en la que fue programada en LabVIEW.

$$F_K = F_{K-1} + a * (E - F_{K-1}) \quad (2.5)$$

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

E : Señal de entrada del filtro

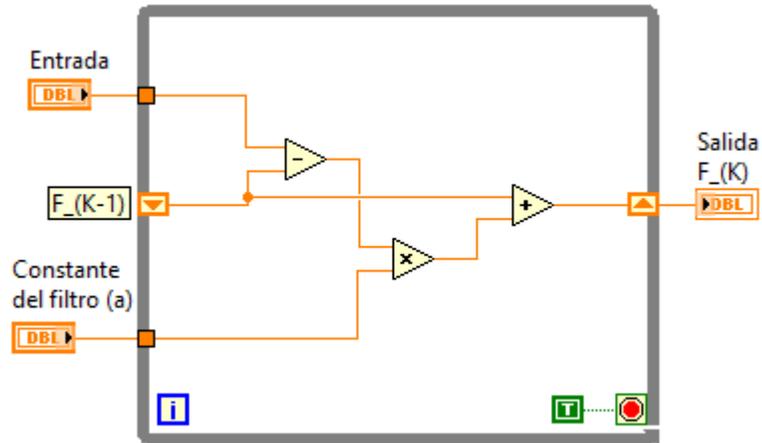


Figura. 2.23 Ecuación del filtro paso bajo implementada en LabVIEW

2.6 Consideraciones finales del capítulo

La obtención de la función transferencial del servomotor posibilita el análisis y el control automático del sistema en estudio. Con el software creado se puede controlar dicho motor digitalmente y de manera directa; la simulación de la planta permite crear varios puestos de trabajo para que los estudiantes de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas ejerciten sus conocimientos. Además pueden cometer errores sin que se afecte el servosistema, una de las ventajas que trae consigo el uso de la simulación.

CAPÍTULO 3. Pruebas y resultados

En el presente capítulo se describen un grupo de pruebas simuladas y reales que se le hicieron al software SCD SAD-100. Los resultados de cada una de estas pruebas son analizados. Se exponen los elementos básicos que posee una guía para una práctica de laboratorio que puede ser desarrollada en algún grupo de estudiantes de la carrera, fundamentalmente los que cursen el tercer año. Se realiza una valoración del impacto económico y medioambiental que provoca la implementación de este trabajo de diploma.

3.1 Pruebas simuladas

Con el propósito de comprobar el correcto funcionamiento del software SCD SAD-100 el mismo fue sometido a un conjunto de pruebas simuladas, en las que fueron variados de manera paulatina los parámetros del proceso, por lo que consecuentemente se realizaron algunos ajustes del regulador PID utilizado. La mayoría de estas pruebas fueron hechas en el software LabVIEW, pero una de ellas se realizó con el software MATLAB, las mismas arrojaron resultados satisfactorios.

3.1.1 Pruebas con LabVIEW

Se realizaron varias pruebas simuladas en tiempo real con el LabVIEW, de las cuales son descritas en el presente epígrafe las más sobresalientes. En cada caso el ajuste del regulador fue realizado por el criterio de prueba y error. Es preciso destacar que no se utilizó la parte derivativa del controlador PID, debido a que esta amplifica los niveles de ruido que posee el sistema de control para un motor CD si no se implementa alguna modificación. Una variación de la carga fue introducida en el sistema; los niveles de ruido se simularon para un valor aleatorio de 50 rpm, puesto que no se disponía de un medidor externo de velocidad.

1. Proceso ideal y controlador proporcional

La primera prueba consistió en considerar que los parámetros del servomotor se comportan idealmente y ajustar un controlador proporcional para verificar si con esta única acción de control es posible obtener una buena respuesta transitoria y si el sistema respondía con bajo error en estado estable. Para un primer ajuste de $K_c = 0.0025$ el error fue muy elevado, al aumentar este valor a 0.005 el error en estado estable disminuyó como era de esperar, lo que se puede observar en la figura 3.1:

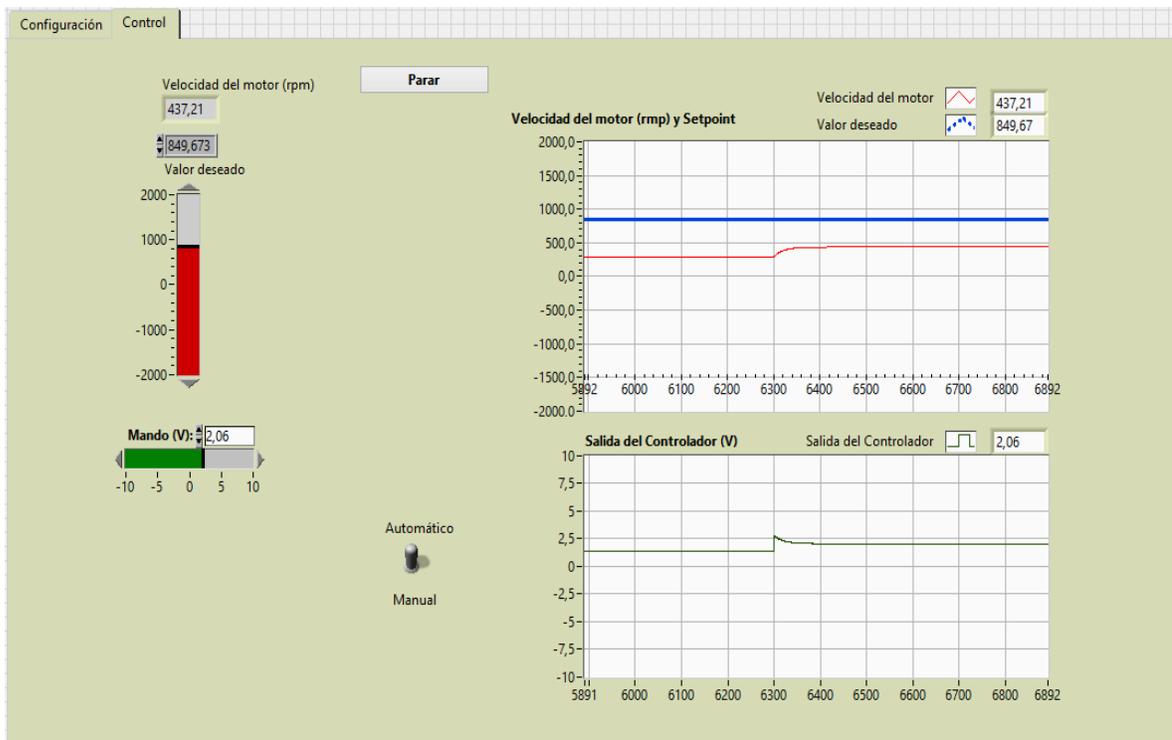


Figura. 3.1 Prueba de control proporcional sin considerar no linealidades

2. Proceso ideal con controlador PI

Como la respuesta del sistema no fue la deseada, al persistir el error, se efectuó una segunda prueba en la que se incluyó la acción integral, para este ajuste del controlador PI con $K_c = 0.0025$ y un tiempo integral (T_i) de 0.02 segundos el sistema presentó una respuesta subamortiguada y poco oscilatoria, como se muestra en la figura 3.2:

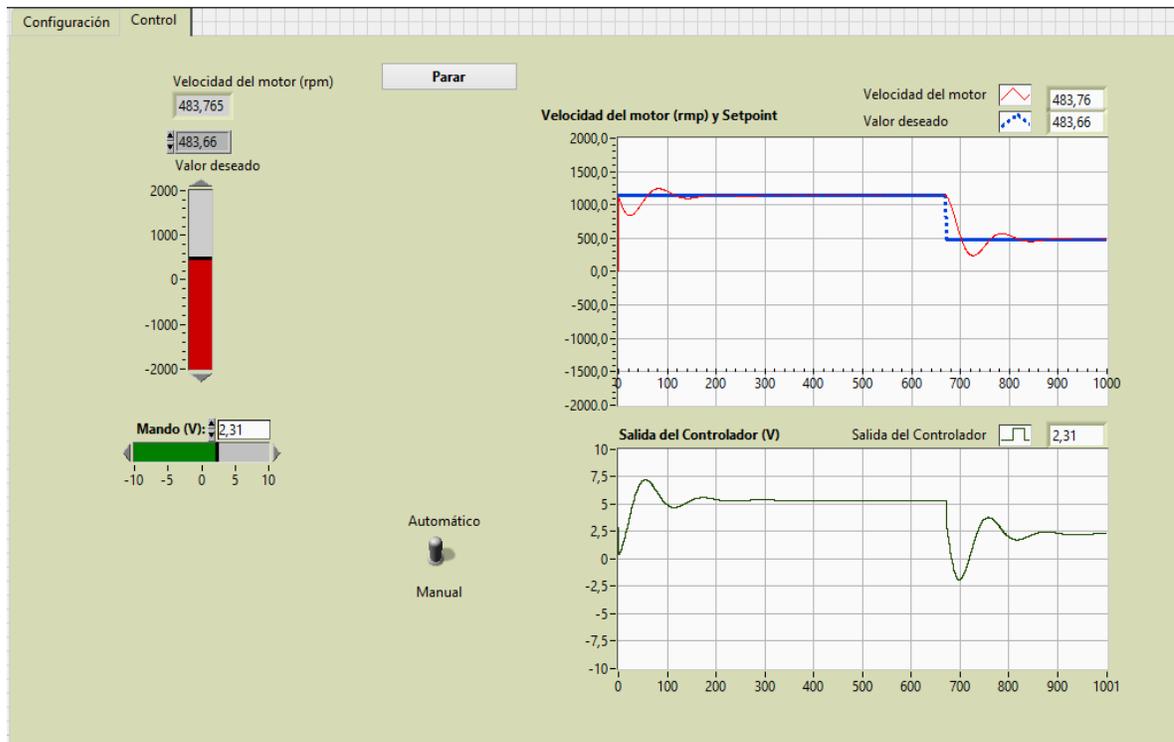


Figura. 3.2 Prueba con un controlador PI sin considerar las no linealidades.

3. Uso de banda proporcional en el regulador

Esta prueba fue realizada para comprobar el empleo de la banda proporcional (ver anexo IV.A), se utilizó el equivalente del ajuste del regulador de la prueba anterior, que es una banda proporcional de 400 %; la planta se simuló de igual manera sin considerar ninguna alinealidad. Ambos resultados fueron muy similares.

4. Proceso con zona muerta

Con el fin de analizar los efectos negativos que provoca la existencia de zona muerta en el sistema de control de lazo cerrado se realizó la presente prueba (ver anexo IV.B), al incluir esta alinealidad el sistema comenzó a oscilar sostenidamente. Por este motivo fue necesario disminuir la ganancia proporcional a 0.0005 para lograr la estabilidad (ver anexo IV.C).

5. Proceso con disturbio

Esta prueba fue concebida para comprobar que el sistema puede compensar el efecto indeseado de un disturbio, en este caso una variación en la carga de 500 rpm. El controlador PI con $K_c = 0.0025$ y $T_i = 0.02$ logró contrarrestar la entrada perturbadora (ver anexo IV.D); se puede observar como en el instante en el que entra el disturbio la velocidad del motor disminuye, e inmediatamente la salida del regulador aumenta para compensar esta variación.

6. Ruido en la medición

Esta prueba se hizo para simular las señales ruidosas que se originan en el Tacodinamo (ver anexo IV.E), las cuales fueron atenuadas a través del filtro que se colocó en la señal medida (ver anexo IV.F), para esto se le dio un valor de 0.7 a la constante del filtro (a), debido a que, si toma valores pequeños aunque el filtraje es mayor provoca un aumento en el orden del sistema.

7. Sistema con varias consideraciones prácticas

La séptima prueba se realizó tomando en cuenta todas las consideraciones prácticas, con el mismo ajuste para el controlador PI, así como los mismos valores de variación de carga y nivel de ruido del tacogenerador. El resultado de esta simulación se muestra en la figura 3.3, el sistema comenzó a oscilar sostenidamente, lo que representa el límite de la inestabilidad. Por lo antes mencionado se decidió disminuir la ganancia proporcional hasta 0.001, con este nuevo ajuste el sistema logró estabilizarse (ver anexo IV.G), aunque el tiempo de asentamiento fue relativamente elevado (5 segundos) en comparación con las pruebas anteriores, como era de esperar.

8. Cambio de manual a automático

Esta prueba simulada se hizo para comprobar el correcto funcionamiento del cambio entre los modos manual y automático que se implementó en el PID.vi. Como resultado se puede apreciar las respuestas en el cambio de manual a automático (ver anexo IV.H).

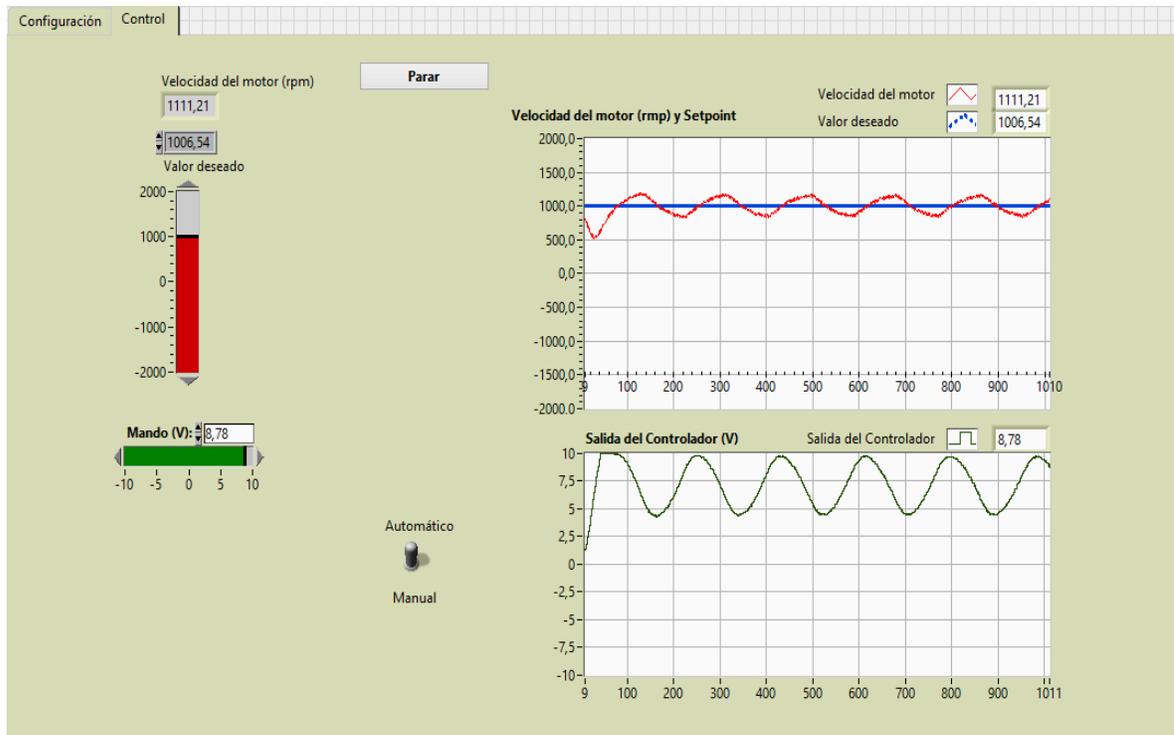


Figura. 3.3 Prueba con un controlador PI, se consideraron varias no linealidades

3.1.2 Prueba con MATLAB

La prueba con el software MATLAB fue concebida para comparar su resultado con el de una de las simulaciones hechas con LabVIEW, concretamente la que se realizó considerando ideales los parámetros de la planta y con un ajuste del regulador de $K_c = 0.0025$ y $T_i = 0.02$. Como se aprecia en la figura 3.4 la respuesta del sistema en lazo cerrado a una entrada paso con la misma amplitud, de 1150 a 483 rpm, fue muy similar a la que simuló el software creado en LabVIEW.

De igual manera la gráfica de la salida del controlador, mostrada en la figura 3.5, tuvo una gran semejanza con la obtenida con el software SCD SAD-100.

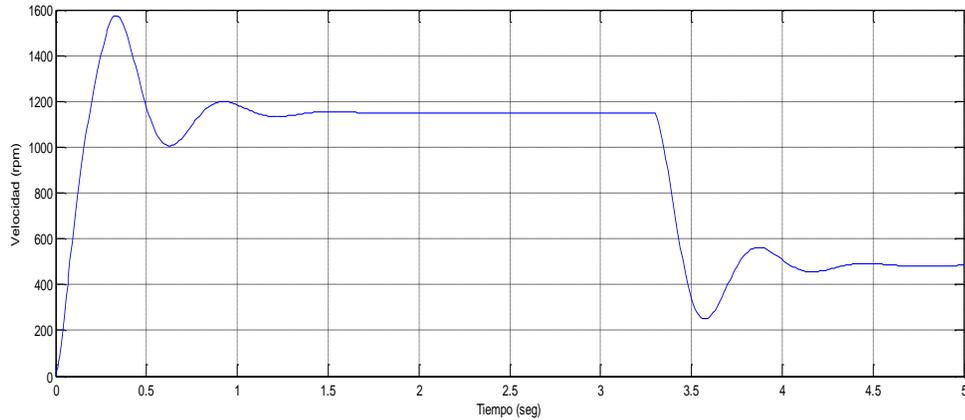


Figura. 3.4 Respuesta del sistema en lazo cerrado a entrada escalón

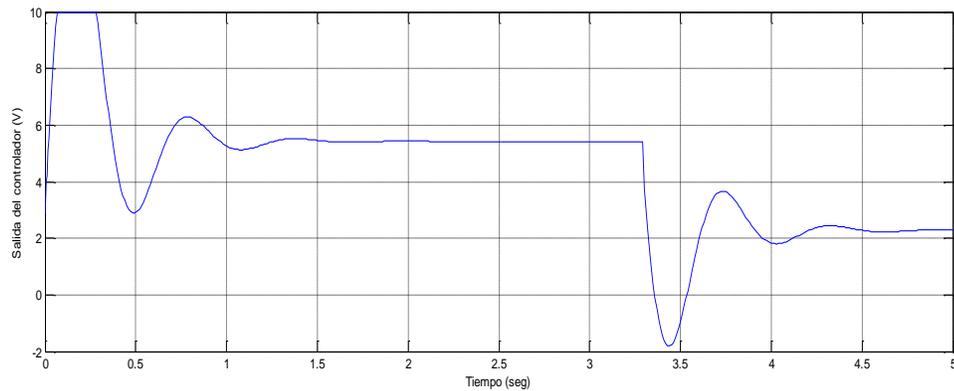


Figura. 3.5 Salida del regulador PI

3.1.3 Prueba de tiempo real

Para demostrar que el software desarrollado en LabVIEW se ejecuta en tiempo real, el mismo fue sometido a una prueba descrita seguidamente. El lazo de control programado en LabVIEW fue encerrado dentro de una estructura de secuencia acumulada (*Stacked Sequence Structure*), con el propósito de captar la cantidad de ciclos transcurridos durante una corrida del software creado en el presente trabajo. Se colocaron dos contadores, mostrados en la figura 3.6, uno para contar los milisegundos transcurridos hasta el instante en que se ejecuta el software, y el otro que devuelve el valor de tiempo en milisegundos acontecido hasta que se detiene dicha ejecución. La diferencia entre ambos representa el tiempo total de corrida del software.

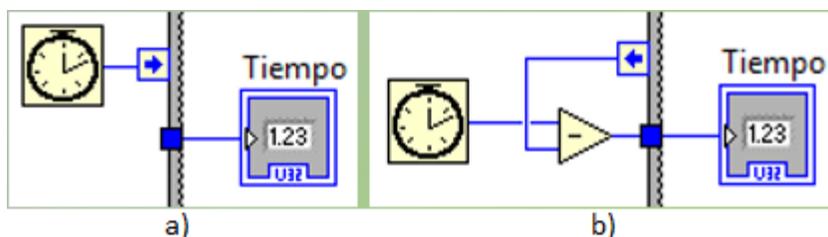


Figura. 3.6 Forma de obtención del tiempo de ejecución del software: a) contador de inicio, b) contador de fin

Al dividir el tiempo total entre la cantidad de ciclos se obtuvo un período de muestreo de 5.0004 ms, valor que es igual al fijado para la realización de esta prueba. Por lo expuesto anteriormente, se puede afirmar que las simulaciones que se realicen con el software desarrollado en LabVIEW serán en tiempo real.

3.2 Pruebas reales

En todas estas pruebas se puede apreciar la existencia de ruidos en el sistema de control, originados en el tacogenerador, además la influencia de la zona muerta propia del servomotor es evidente. Para adquirir las muestras el software se ejecutó cada 5 milisegundos (ms), el criterio utilizado para seleccionar este período de muestreo fue el de 10 a 20 veces menor que el período de oscilación de la señal de salida del sistema de control en lazo cerrado. Se realizaron un total de nueve pruebas descritas a continuación.

1. Regulador proporcional

La primera fue efectuada bajo las mismas condiciones que tuvo su correspondiente de las pruebas simuladas, es decir, un control proporcional con un ajuste inicial de $K_c = 0.0025$ y uno final de $K_c = 0.005$. El resultado, mostrado en la figura 3.7, fue prácticamente el mismo que el de la prueba simulada, se puede apreciar como al aumentar la ganancia el error en estado estable disminuye.

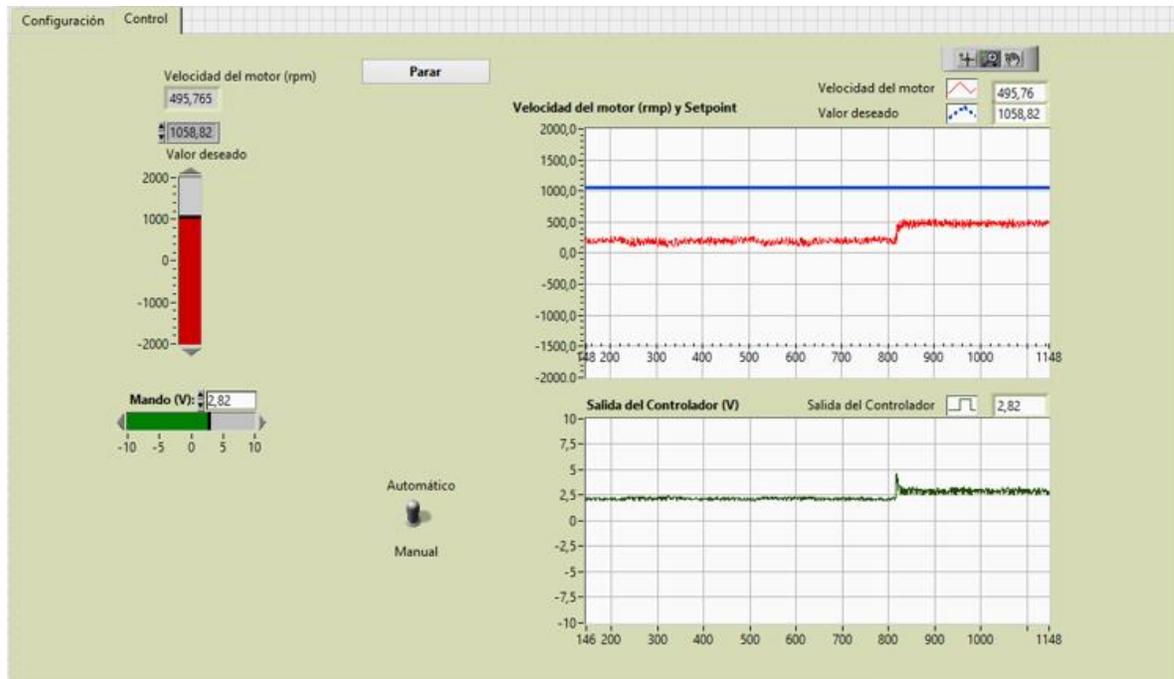


Figura. 3.7 Prueba real de control proporcional

2. Regulador PI

Esta prueba también constituye una evidencia de la buena exactitud de las simulaciones que se pueden realizar a través del software SCD SAD-100, pues se hizo con el mismo ajuste del regulador de la segunda prueba simulada, $K_c = 0.0025$ y $T_i = 0.02$. La respuesta transitoria y la gráfica de la señal de mando que se obtuvieron fueron muy parecidas a sus homólogas simuladas, como ilustra la figura 3.8.

3. Uso de banda proporcional en el regulador

La tercera prueba real fue para comprobar el uso de la banda proporcional como unidad de entrada y en el controlador (ver anexo V.A), se utilizó el equivalente del ajuste del regulador de la prueba anterior, que es una banda proporcional de 400 %. En la respuesta transitoria del sistema se aprecia el efecto de la zona muerta que posee el servomotor, que provoca que el tiempo de subida sea mayor que el de la respuesta transitoria simulada, no obstante ambos resultados fueron similares.

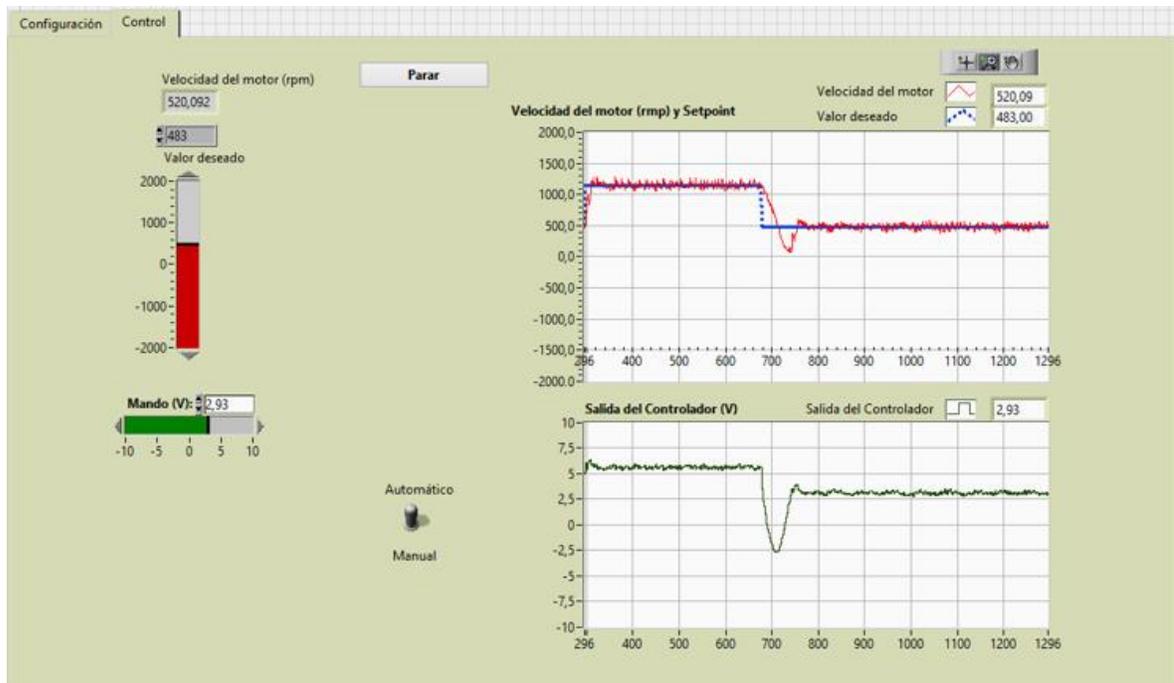


Figura. 3.8 Prueba real con un controlador PI

4. Proceso con disturbio

Esta prueba se realizó con un nuevo ajuste del regulador PI con $K_c = 0.002$ y $T_i = 0.1$, para chequear si realmente el sistema en lazo cerrado puede contrarrestar los efectos de una perturbación. Para ello se le provocó al sistema una variación de la carga al aplicar una fricción en la masa. Los resultados fueron los esperados, ya que, cuando la velocidad del motor CD sufrió una disminución, la señal de salida de voltaje del controlador se elevó hasta que el motor retomó su valor de velocidad. Esta descripción se observa claramente en la figura 3.9.

5. Filtrado de la señal medida

Para constatar como las señales de salida del sistema y del controlador se hacen menos ruidosas cuando se filtra la señal medida fue efectuada esta prueba. La constante de filtraje tomó un valor de $a = 0.7$, el mismo que en la prueba simulada, por la razón explicada anteriormente en esa prueba. En el anexo V.B se puede observar como la respuesta del sistema ante entrada escalón es ligeramente subamortiguada y se estabiliza rápidamente, con un tiempo de asentamiento de 0.28 segundos.

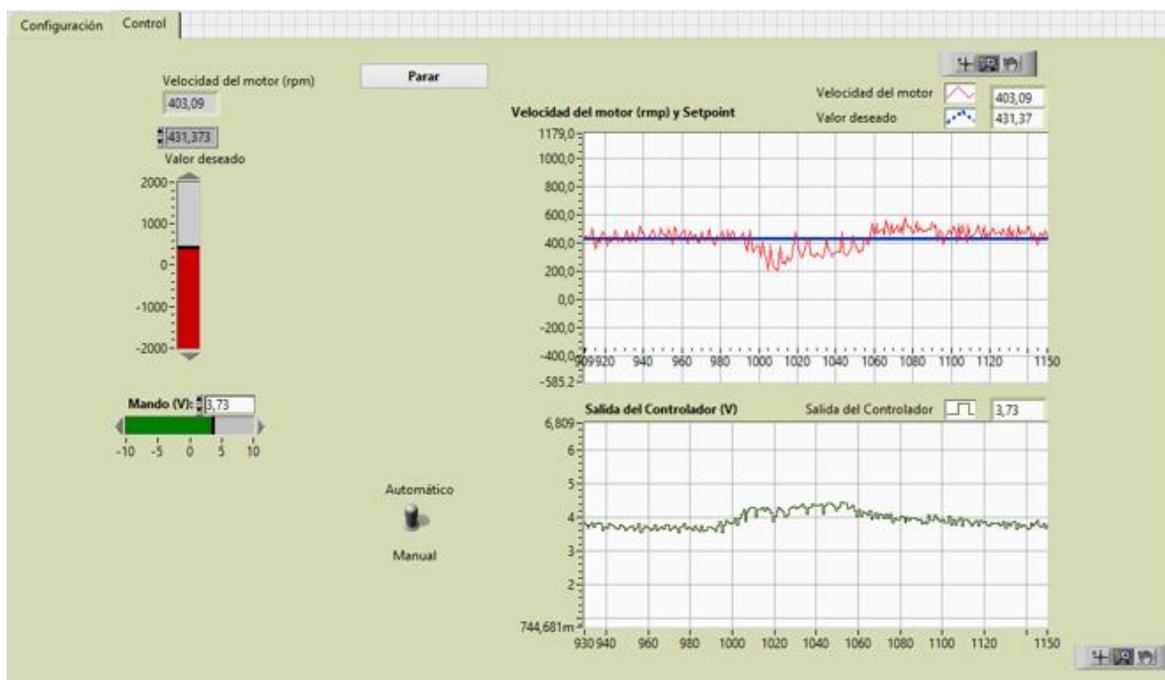


Figura. 3.9 Prueba real de variación de carga

6. Tiempo integral

La sexta prueba (ver anexo V.C) se realizó para analizar los efectos que produce el tiempo integral sobre el sistema en estudio. En las pruebas anteriores se había empleado un $T_i = 0.02$, al aumentarlo hasta un valor de 0.2 el sistema pasó de ser ligeramente subamortiguado a sobreamortiguado, el tiempo de establecimiento aumentó, lo que hizo más lenta la respuesta.

7. Tiempo derivativo

Los resultados de esta prueba real (ver anexo V.D) demuestran que el tiempo derivativo (T_d) no se debe utilizar en el ajuste de un regulador PID si la planta es un motor. Cuando T_d tomó un valor elevado de 0.1 segundos los niveles de ruido de la velocidad del motor y del voltaje de salida del controlador aumentaron significativamente, al disminuir T_d hasta un valor de 0.01 segundos dichos ruidos se atenuaron.

8. Período de muestreo

La octava prueba fue realizada con el objetivo de analizar las implicaciones que conlleva la selección del período de muestreo sobre el sistema de control. Con el valor escogido de 5 ms tanto la señal de salida de velocidad como el mando presentan pocas oscilaciones; si dicho período de muestreo aumenta gradualmente (ver anexos V.E y V.F) las oscilaciones también se incrementan como consecuencia. Para el sistema tratado en este trabajo de diploma, si se muestrea cada 60 ms, la salida del mismo oscila sostenidamente (ver anexo V.G).

9. Cambio del modo manual al automático

Esta prueba fue concebida con el propósito de comprobar el funcionamiento correcto del software cuando se realiza el cambio de manual a automático. Para efectuarla se pasó al modo manual y se ajustó la señal de mando hasta que la velocidad del servomotor alcanzó un valor cercano al deseado, seguidamente se realizó el cambio al modo automático, la salida del sistema alcanzó el *setpoint* rápidamente y el mando tomó el mismo valor que tenía antes de la transición, como se observa en la figura 3.10. Por lo antes mencionado se puede decir que esta prueba arrojó un resultado satisfactorio.

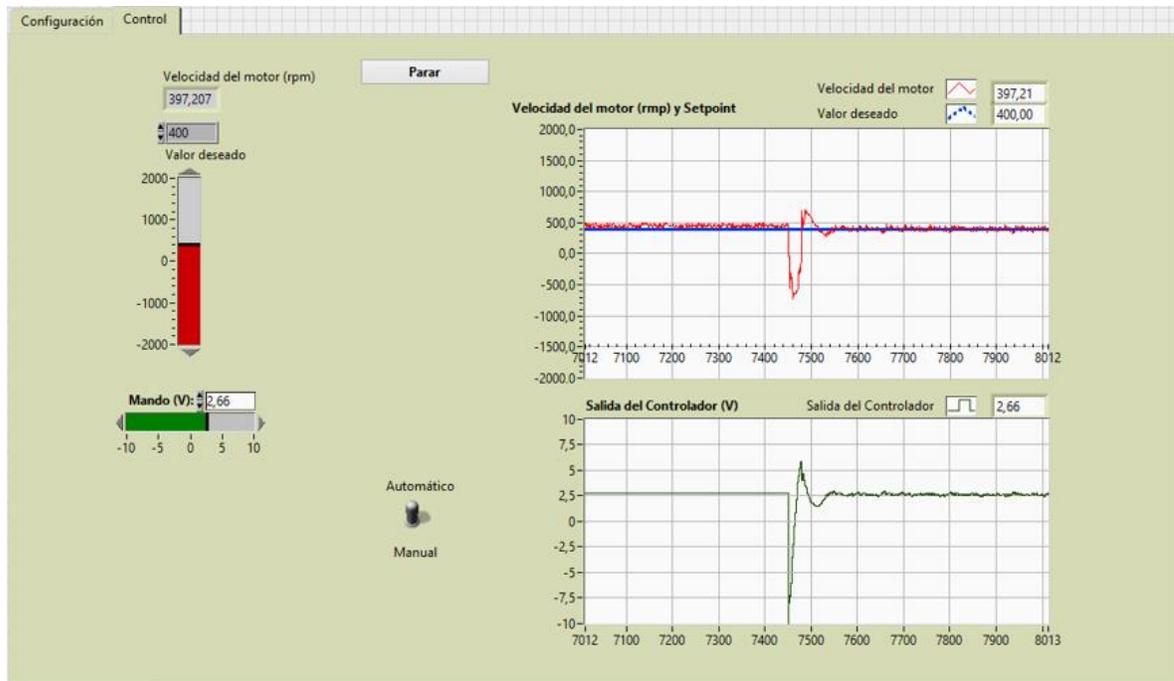


Figura. 3.10 Cambio del modo manual al automático

3.3 Guía para práctica de laboratorio

Con el software SCD SAD-100 se pueden realizar múltiples prácticas de laboratorio, en este trabajo de diploma se presenta una guía para una actividad práctica titulada “Respuesta transitoria de un sistema de primer orden” (ver anexo VI), lo que constituye un ejemplo de la utilidad e importancia que posee dicho software.

La práctica de laboratorio está concebida para que los estudiantes que cursen el tercer año de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas y hayan tenido un primer acercamiento a los fundamentos básicos de la teoría del control moderna puedan consolidar los conocimientos adquiridos en la asignatura Ingeniería de Control I. Como el software se ejecuta en tiempo real los estudiantes tendrán la posibilidad de familiarizarse con este concepto.

En la práctica de laboratorio los alumnos deberán ajustar un controlador proporcional y uno proporcional e integral por el método de prueba y error, es decir, variar la ganancia y el tiempo integral hasta que el sistema tenga un buen comportamiento transitorio. Para lograr esto deberán analizar los tiempos de asentamiento, de subida, de retardo y pico; además del máximo sobreimpulso y el error en estado estable.

3.4 Valoración económica e impacto medioambiental

Un primer aspecto a tener en cuenta en la valoración económica de esta investigación es que en el mercado existen diversos fabricantes y proveedores de sistemas de control didácticos como son: Quanser, Alecop, Feedback, Humusoft, entre otros. Con los equipos desarrollados por ellos es posible la realización de numerosos experimentos, pero los precios de estas maquetas están entre los 4000 y 10000 € aproximadamente (en dependencia del fabricante), lo que constituye un costo demasiado alto en un solo medio de enseñanza.

Por otra parte en el presente trabajo se utilizan elementos ya existentes: el servo mecanismo Alecop SAD-100 y la tarjeta de adquisición de datos NI USB-6212, por lo que el costo de estos componentes es despreciable. Otro elemento a considerar es la utilización del software propietario LabVIEW, cuya licencia es alta, pero en nuestro país es usado sin autorización

de los dueños en virtud del bloqueo económico a que está sometido, por consiguiente el costo del mismo no será considerado.

Por consiguiente el mayor costo para el uso del trabajo realizado recae en las computadoras que se empleen para realizar las simulaciones en las clases, debe señalarse que una computadora con las prestaciones necesarias para poder correr el simulador le cuesta al país aproximadamente 450 CUC; pero las mismas son usadas en numerosas actividades dentro de la docencia universitaria.

De lo anterior se evidencia que la realización e introducción de los resultados alcanzados en la investigación desarrollada son factibles económicamente y representan un ahorro significativo comparado con lo que hubiera que gastar para adquirir varios puestos de trabajo con maquetas producidas por fabricantes especializados en el desarrollo de sistemas de control didácticos.

Desde el punto de vista del impacto medioambiental es importante señalar que la maqueta ALECOP SAD-100 necesita un voltaje de alimentación de 220 V y 10 A, lo que se traduce en un consumo de energía eléctrica de 2.2 kWh. Cuba necesita quemar un promedio de 277 g de combustible fósil para generar 1 kWh; se estima que por cada kilowatt-hora generado en una de nuestras centrales termoeléctricas, se liberan a la atmósfera 715 g de gases de efecto invernadero (Arrastía, n.d.). Por lo tanto es evidente que con las simulaciones que se pueden realizar con el software SCD SAD-100 se ahorraría una cantidad considerable de energía eléctrica y por consiguiente disminuiría la contaminación del medio ambiente.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Con la realización del trabajo se arriban a las siguientes conclusiones:

- 1 Tras el análisis bibliográfico efectuado sobre el control digital y los simuladores de control de procesos se puede afirmar que son materias muy importantes y útiles dentro del campo de la automática.
- 2 Mediante la caracterización del servocontrol SAD-100 y la obtención de su función transferencial es posible realizar un buen ajuste del modelo virtual de la planta con el real.
- 3 La utilización del software SCD SAD-100, con posibilidad de ejecución simulado o real, permite mejorar el proceso de formación de los estudiantes de la carrera Ingeniería en Automática de la Universidad Central “Marta Abreu” de Las Villas.
- 4 Las simulaciones realizadas con el software desarrollado son altamente fiables, pues se aproximan en gran medida a la realidad.
- 5 La guía para una práctica de laboratorio que se propone evidencia la utilidad docente que tiene el software SCD SAD-100 y sirve de base para la confección de otras guías.

Recomendaciones

- 1 Implementar el software SCD SAD-100 en las asignaturas Ingeniería de Control I y Herramientas de Software para el Control.
- 2 Ampliar las prestaciones del software SCD SAD-100 agregándole otras consideraciones prácticas.

- 3 A partir de la práctica de laboratorio creada proponer otras guías para realizar actividades prácticas.

REFERENCIAS BIBLIOGRÁFICAS

- Acuña, W., Ríos, A., 2008. Aplicando tolerancia a fallas para la implementación de compensación de saturación en accionadores. RIAI Vol. 5, 21–28.
- Aguirre, I., Prieto, F., Ríos, M., 2010. Comparación del Rendimiento de Controladores en el Entrenador Industrial DAC603. Universidad de los Andes, Mérida, p. 8.
- Aguirre, L.A., Guarnizo, J., 2008. Diseño detallado de un simulador de vuelo dinámico (Trabajo de Diploma). Universidad de San Buenaventura, Bogotá.
- Albertí, E.B., 2003. Señales y sistemas de tiempo discreto. Edicions UPC, Barcelona.
- Alecop, 1986. Manual Servosistemas SAD-100. (Manual del fabricante).
- Arellano, M.A., 2013. MATLAB y Simulink para ingeniería, 1ra ed. Perú.
- Arrastía, M.A., n.d. Diez preguntas sobre el ahorro de energía eléctrica [WWW Document]. URL <http://www.cubasolar.cu/biblioteca/energia/Energia33/HTML/articulo03.htm> (accessed 6.12.18).
- Azarang, M., Garcia, E., 1996. Simulacion y analisis de modelos estocasticos. Mc. Graw Hill. México, Mexico.
- Bahón, C.A., Giner, C.R., 2004. Tecnología de sistemas de control. Univ. Politèc. de Catalunya.
- Ballesteros, R.L., Hernández, A.F., 2015. Control por computadora. Editorial Feijóo, Santa Clara.
- Burns, A., Wellings, A., 2003. Sistemas de Tiempo Real y Lenguajes de Programación, 3ra ed. Addison-Wesley.
- Cilento, A., 2007. Sistemas de control de motores en tiempo real mediante MATLAB. (Memoria Técnica). Universitat Rovira I Virgili.
- Fadali, M.S., Visioli, A., 2013. Digital control engineering: analysis and design, Second edition. ed. Academic Press, Elsevier, Amsterdam.
- Fuente, J. de la, 2010. “Sintonía de reguladores PID.” Facultad de Ciencias, Universidad de Valladolid, Valladolid.
- Gil, J.J., Díaz-Cordovés, A.R., 2009. Fundamentos de control automático de sistemas continuos y muestreados. J.J. Gil, San Sebastián.

- González, R., 2010. Apuntes de simulacion y optimizacion. Centro de Investigación y Estudios de Postgrado.
- Hernández, M.I., Ledesma, 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. (Trabajo de Diploma). Escuela Superior Politécnica de Chimborazo, Riobamba.
- Historia de la simulación | Lander Simulation & Training Solutions [WWW Document], 2018. URL <http://www.landersimulation.com/formacion-con-simulacion/el-mundo-en-movimiento/historia-de-la-simulacion/> (accessed 3.27.18).
- Holguín, G.A., Pérez, S.M., Orozco, A.A., 2002. Curso básico LabVIEW 6i. Universidad Tecnológica de Pereira, Pereira.
- Hosseinpour, F., Hajihosseini, H., 2009. Importance of simulation in manufacturing. *World Acad. Sci. Eng. Technol.* 51, 292–295.
- Jolly, A., 2018. LabWindows CVI que es? - Ingeniero especialista en bancos de pruebas y medición, experto en LabVIEW, TestStand, LabWindows/CVI [WWW Document]. URL <http://www.ajolly.com.mx/es/desarrollo-consulta-sistema-supervisor-control-prueba-medida/17-LabWindows-CVI.html> (accessed 3.29.18).
- Kuo, B.C., 2000. Sistemas de control digital. Cecsca.
- Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., Balan, G., 2005. Mason: A multiagent simulation environment. *Simulation* 81, 517–527.
- Morris, A.S., 2001. Measurement and Instrumentation Principles.pdf. Butterworth-Heinemann, Gran Bretaña.
- National Instruments, 2017. SPECIFICATIONS USB-6212 [WWW Document]. URL <http://www.ni.com/pdf/manuals/375196d.pdf> (accessed 4.11.18).
- National Instruments [WWW Document], 2018. URL <http://www.ni.com/lwcv/whatis/esa/> (accessed 3.29.18).
- Nuñez, A., 2011. Control de un tanque de presión mediante el módulo PID de un autómatas programable (Trabajo de Diploma). Universidad Carlos III de Madrid, Leganés.
- Ogata, K., 2010. INGENIERIA DE CONTROL MODERNA. PRENTICE HALL, Madrid.
- Opdenacker, M., Petazzoni, T., Chantepredrix, G., 2010. Realtime in embedded Linux systems.
- Pérez, E.A., 2008. Diseño y montaje de un entrenador con el controlador lógico programable PLC S7-200 de Siemens (Trabajo de Diploma). Universidad Pontificia Bolivariana, Bucaramanga.
- Pichardo, J., 2008. Compensación de zona muerta y variación de carga en el control de velocidad de motores de corriente directa (Tesis de Maestría). Instituto Politécnico Nacional, Querétaro.
- Piedrafita, R., 1999. Evolución histórica de la ingeniería de control.

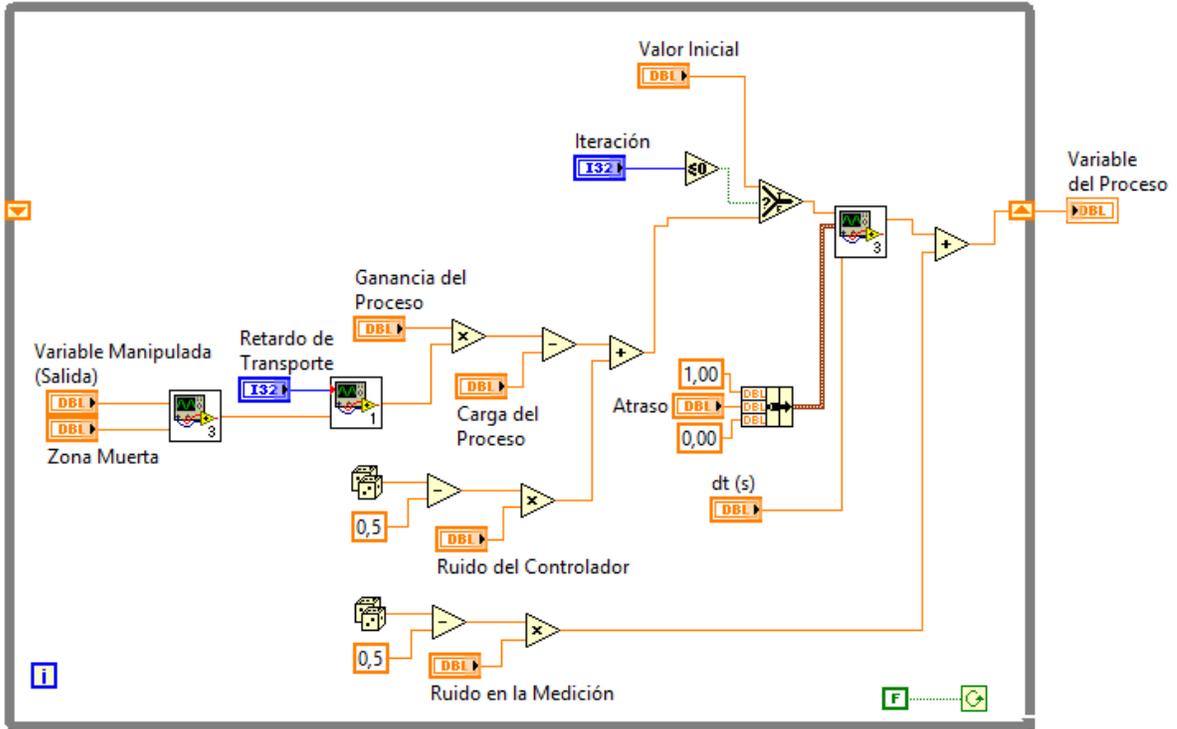
- Proctor, F.M., Shackelford, W.P., 2002. Embedded Real-Time Linux for Cable Robot Control. ASME, pp. 851–856. <https://doi.org/10.1115/DETC2002/CIE-34506>
- Rodríguez, L.D., 2016. Entrenador digital de lazo de control con consideraciones prácticas reales (Trabajo de Diploma). Universidad Central “Marta Abreu” de Las Villas, Santa Clara.
- Rojas, L.F.R., 2007. Controladores PID comerciales (Trabajo de Diploma). Universidad de Costa Rica, Ciudad Universitaria Rodrigo Facio.
- Romero, M., Quesada, A., 2014. Nuevas tecnologías y aprendizaje significativo de las ciencias. Enseñ. Las Cienc. 32. <https://doi.org/10.5565/rev/ensciencias.433>
- Scenna, J.N., Aguirre, A.P., Benz, J.S., Chiotti, J.O., Espinosa, J.H., Ferrero, B.M., Montagna, M.J., Mussati, C.M., Pérez, A.G., Rodríguez, J., Salomone, H.E., Santa Cruz, S.M.A., Tarifa, E.E., Vega, J., 1999. Modelado, simulación y optimización de procesos químicos. Dr. Nicolás José Scenna.
- Villegas, L.E., 2007. Simulación en Hysys de los procesos productivos de ENAP Magallanes (Trabajo de Diploma). Universidad de Magallanes, Punta Arenas.
- Zhou Qingguo, Zhang Wei, Huang Xingwen, McGuire, N., 2008. A case study of nanokernel in education. IEEE, pp. 915–919. <https://doi.org/10.1109/ITME.2008.4744000>
- Zurita, L.M., 2014. Diseño de un simulador digital de procesos. Instituto Superior Politécnico José Antonio Echeverría, Cumaná.

ANEXOS**Anexo I Tabla de la característica estática del servomotor**

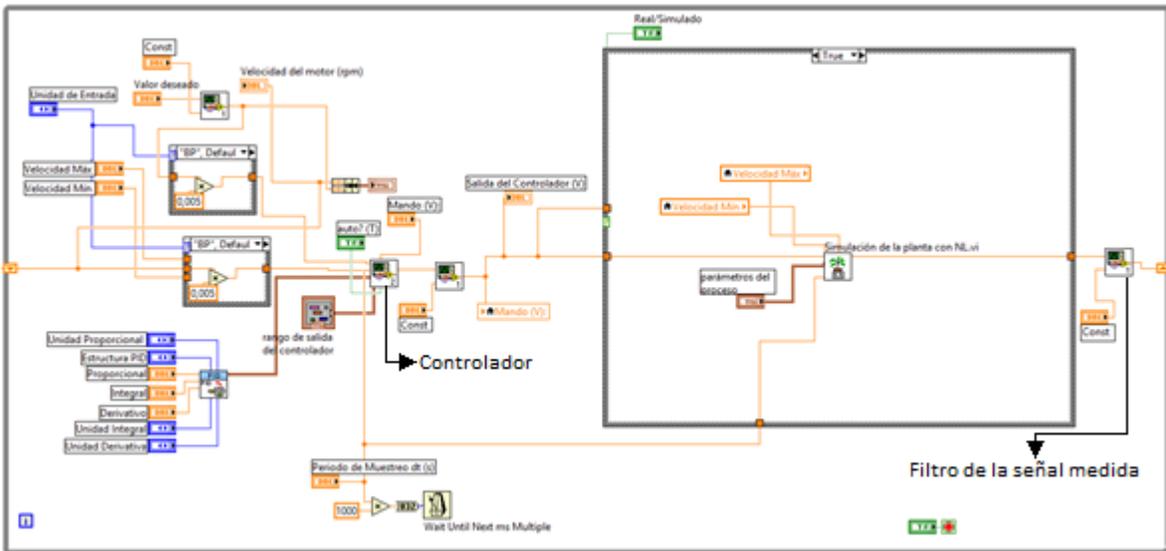
Voltaje de alimentación (V)	Voltaje del tacogenerador (V)	Velocidad del servomotor (rpm)
-10	-12.254	-2451.8
-9.53	-11.192	-2239.3
-9.02	-10.565	-2113.8
-8.51	-9.986	-1998
-7.97	-9.35	-1870.7
-7.5	-8.771	-1754.9
-7.01	-8.153	-1631.3
-6.51	-7.487	-1498
-5.99	-6.773	-1355.1
-5.50	-6.146	-1229.7
-5.00	-5.48	-1096.4
-4.50	-4.805	-961.4
-4.00	-4.188	-837.9
-3.501	-3.581	-716.5
-3.001	-2.872	-574.6
-2.501	-2.017	-403.6

-2.009	-1.07	-214.1
-1.65	0	0
+1.707	0	0
+2.027	+1.018	+203.7
+2.505	+1.89	+378.2
+3.008	+2.687	+537.6
+3.501	+3.392	+678.7
+4.00	+4.014	+803.1
+4.51	+4.631	+926.6
+4.99	+5.316	+1063.6
+5.51	+6.011	+1202.7
+5.99	+6.638	+1328.1
+6.51	+7.304	+1461.4
+7.01	+7.912	+1583
+7.49	+8.491	+1698.9
+8.00	+9.089	+1818.5
+8.50	+9.629	+1926.6
+9.00	+10.208	+2042.4
+9.49	+10.71	+2142.9
+10.1	+11.328	+2266.5

Anexo II Implementación de las no linealidades



Anexo III Lazo de control programado en LabVIEW



Anexo IV Pruebas simuladas

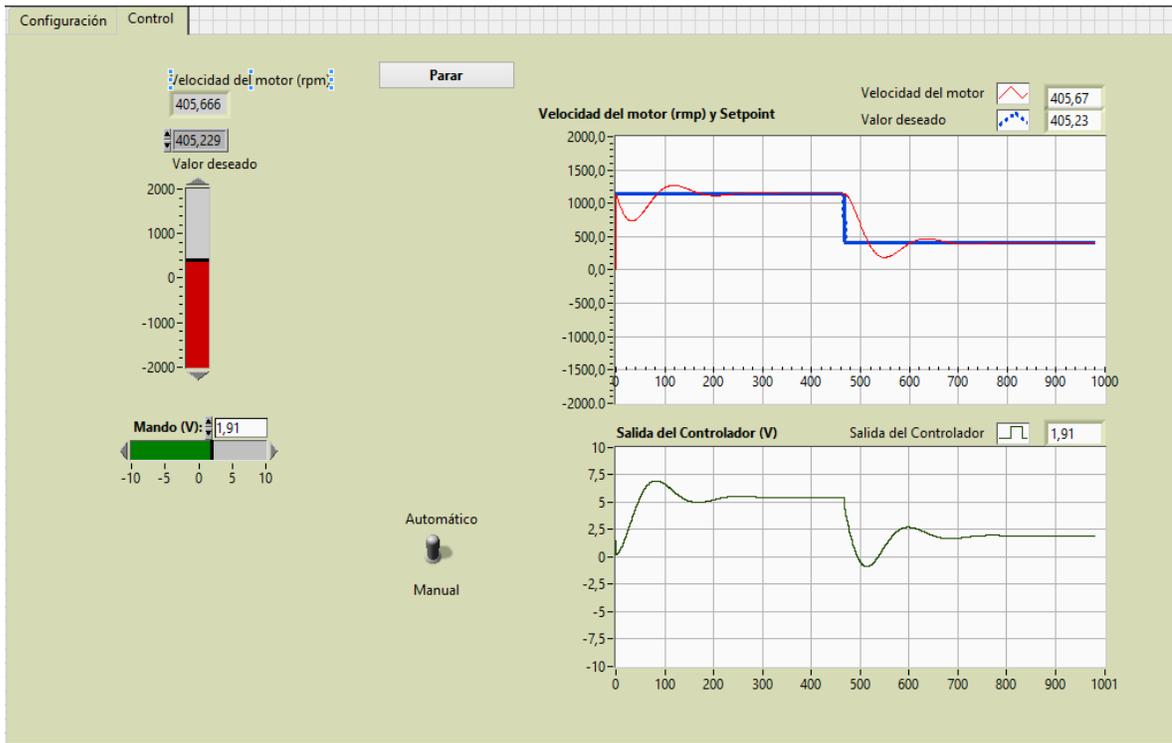


Fig. IV.A Prueba con parámetros ideales y uso de banda proporcional

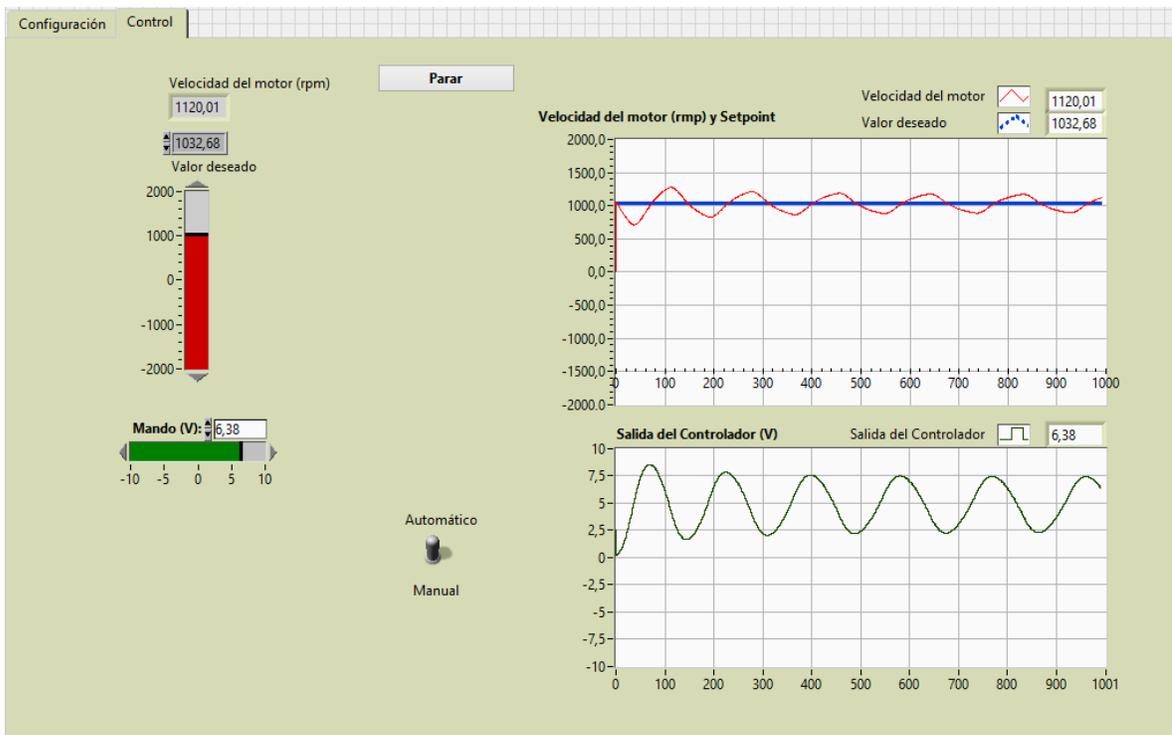


Fig. IV.B Prueba con zona muerta

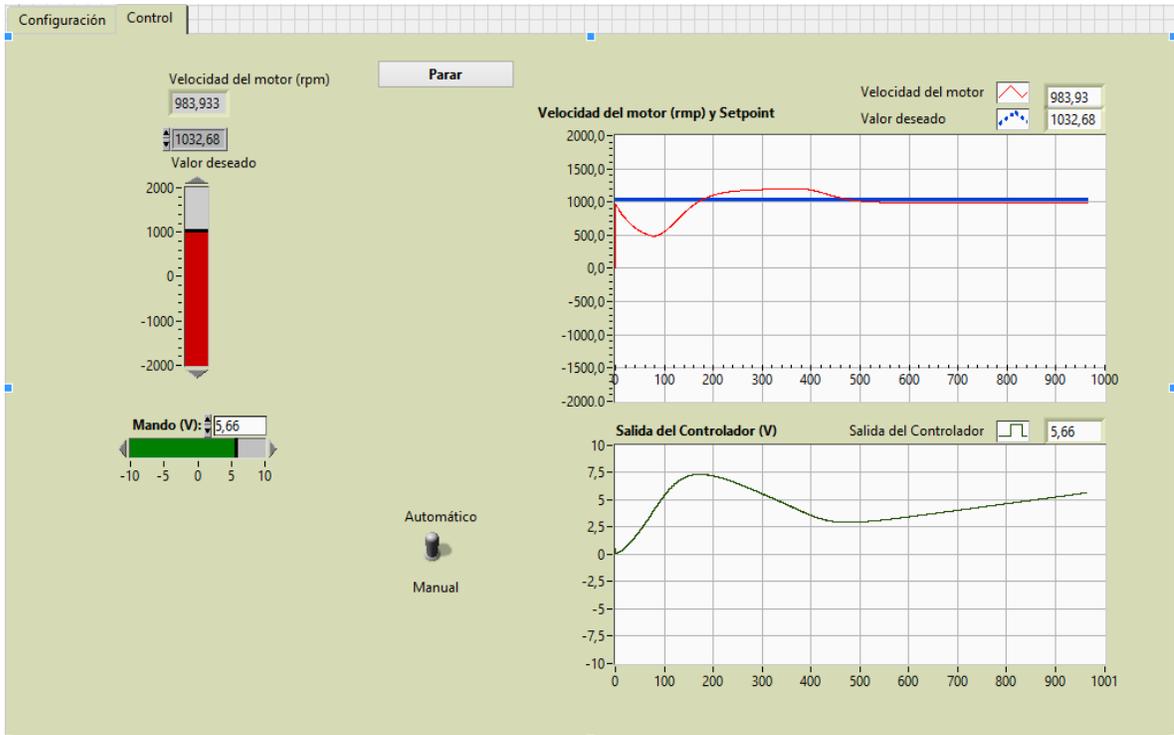


Fig. IV.C Prueba con zona muerta, nuevo ajuste del controlador

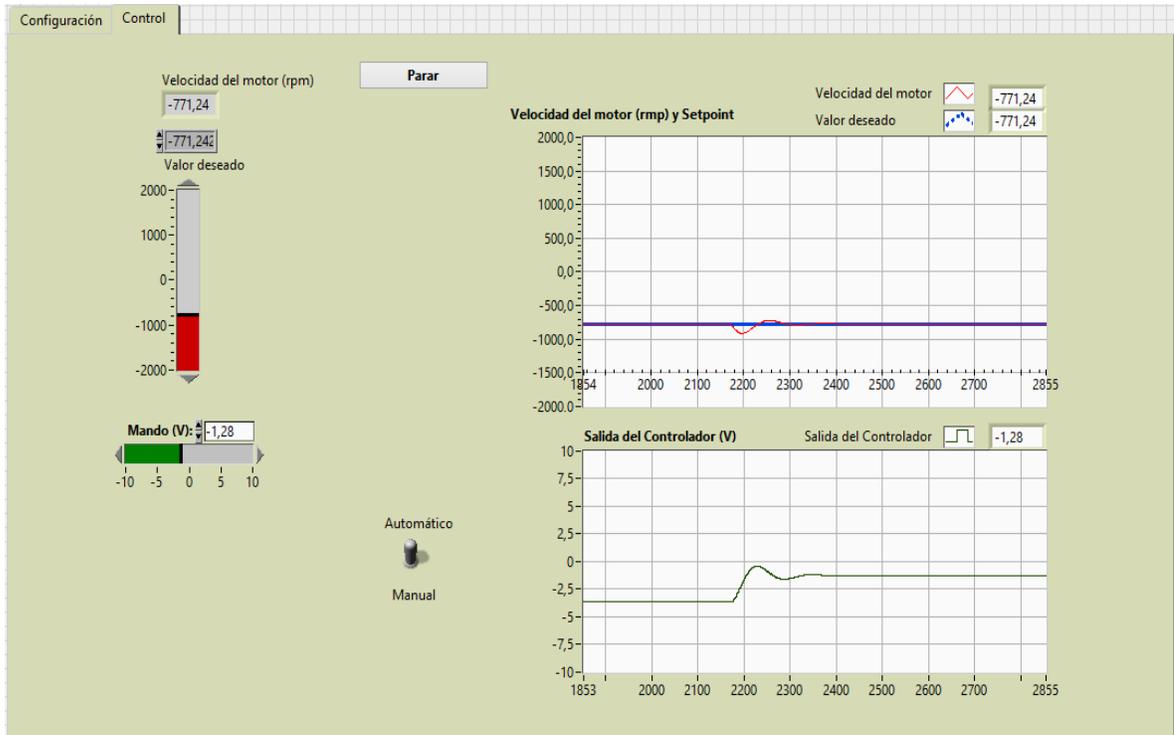


Fig. IV.D Prueba de proceso con disturbio

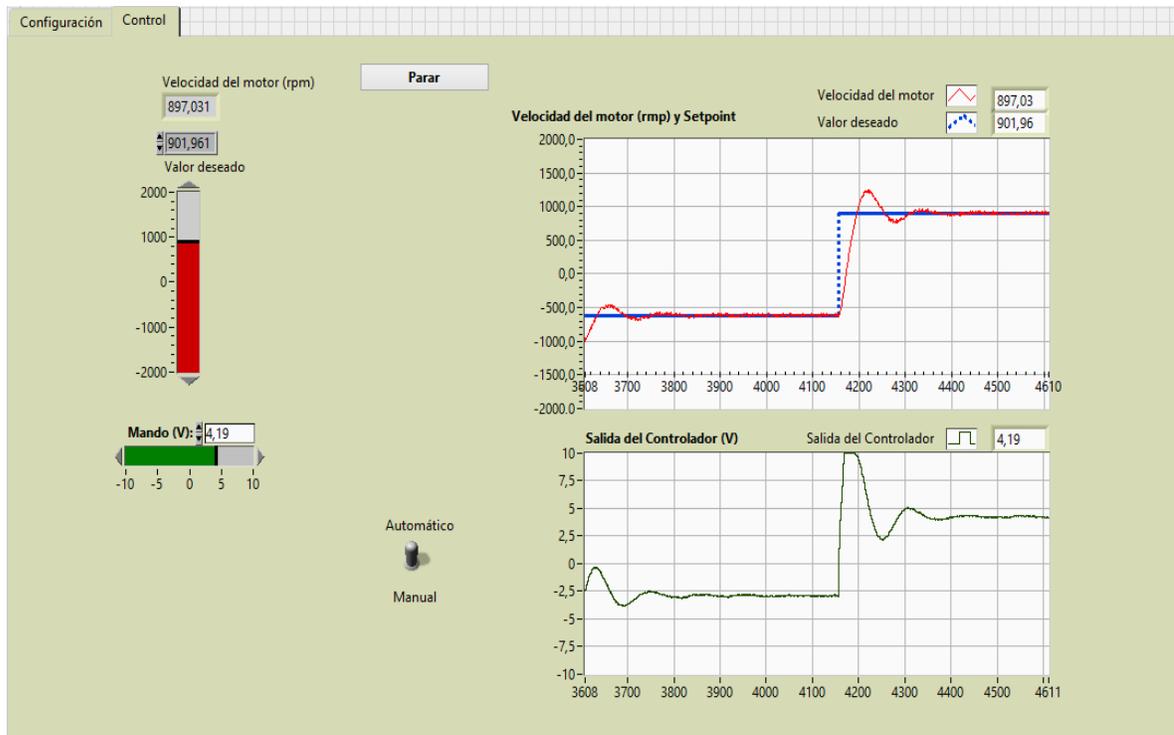


Fig. IV.E Prueba con ruido en la medición

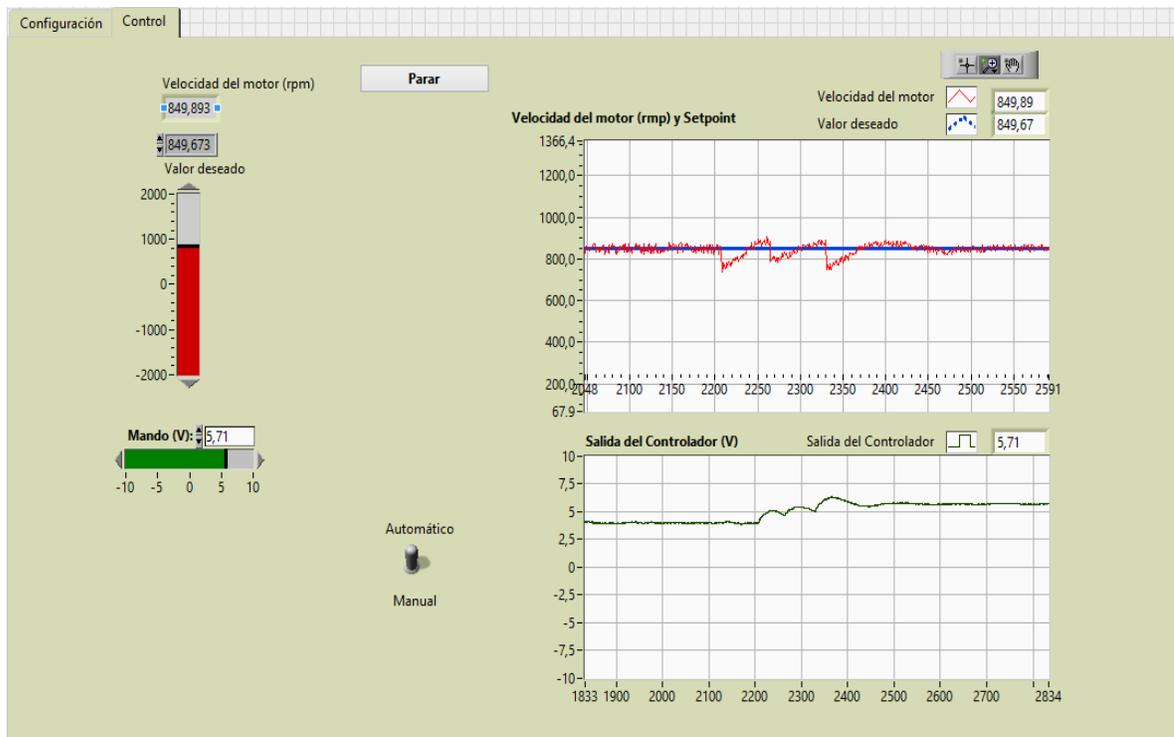


Fig. IV.F Prueba con ruido filtrado en la medición

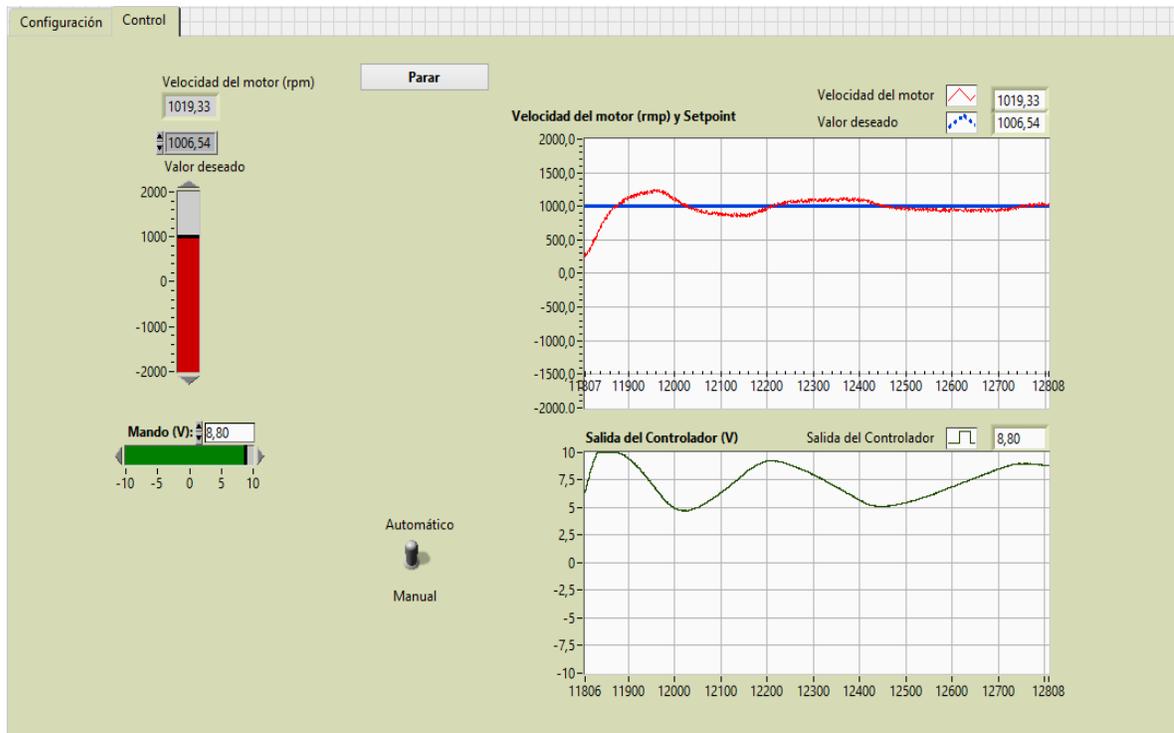


Fig. IV.G Prueba del sistema con varias consideraciones prácticas



Fig. IV.H Prueba del cambio de manual a automático

Anexo V Pruebas reales

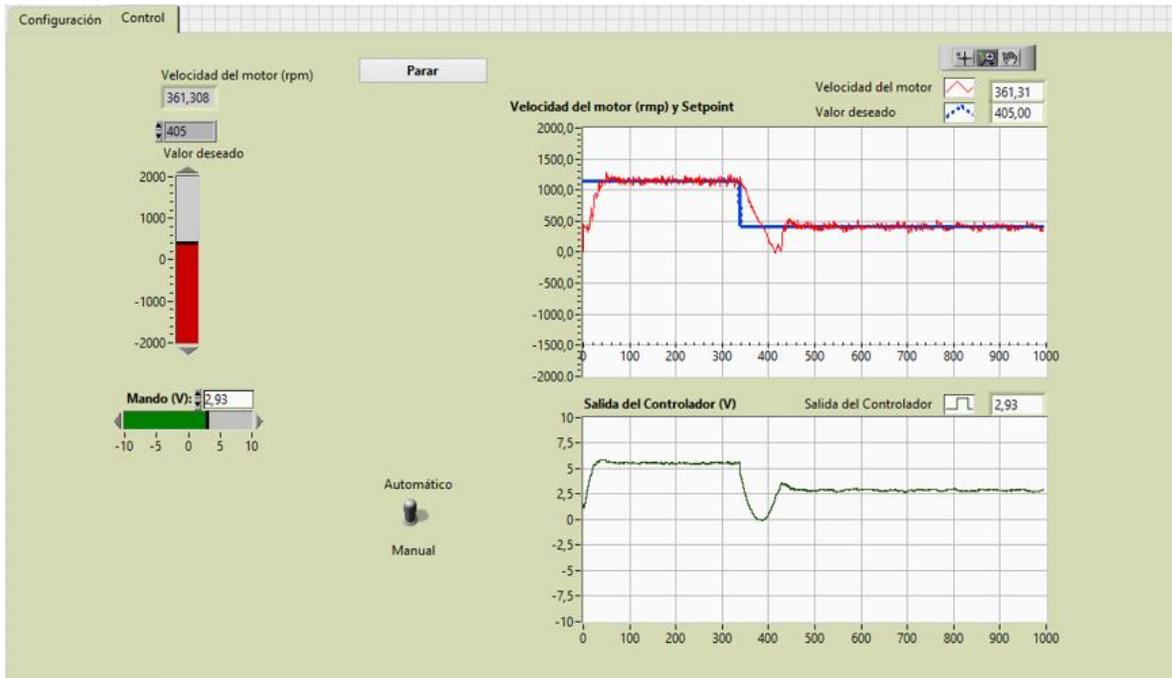


Fig. V.A Prueba del uso de banda proporcional en el regulador

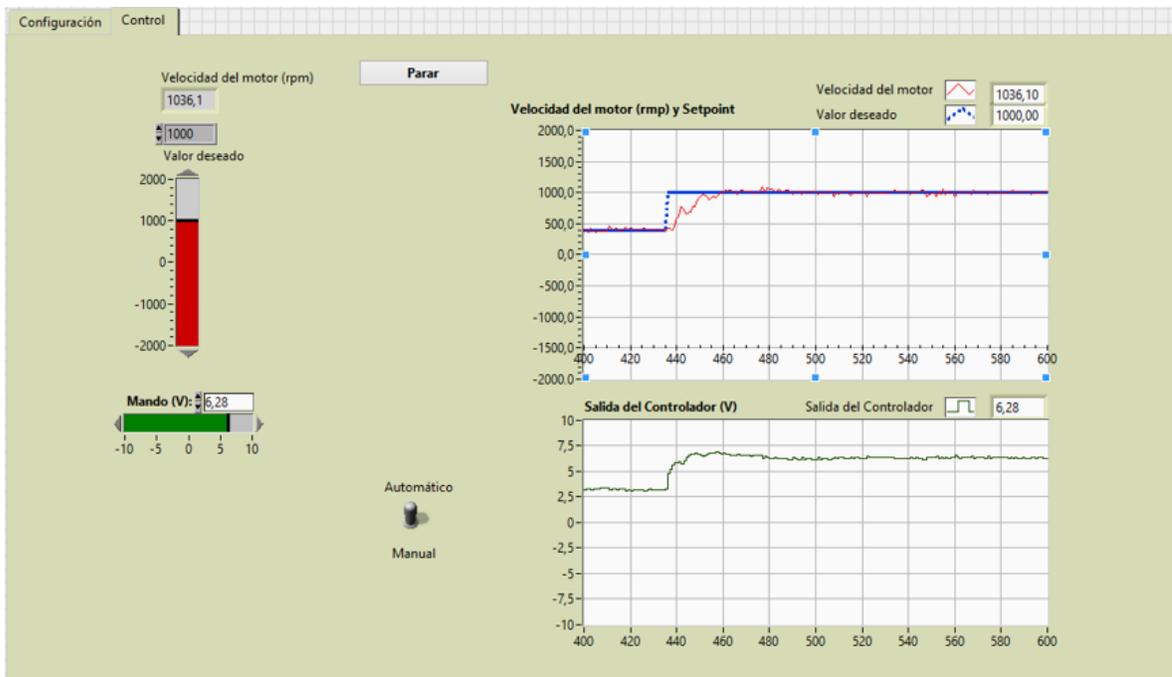


Fig. V.B Filtrado de la señal medida

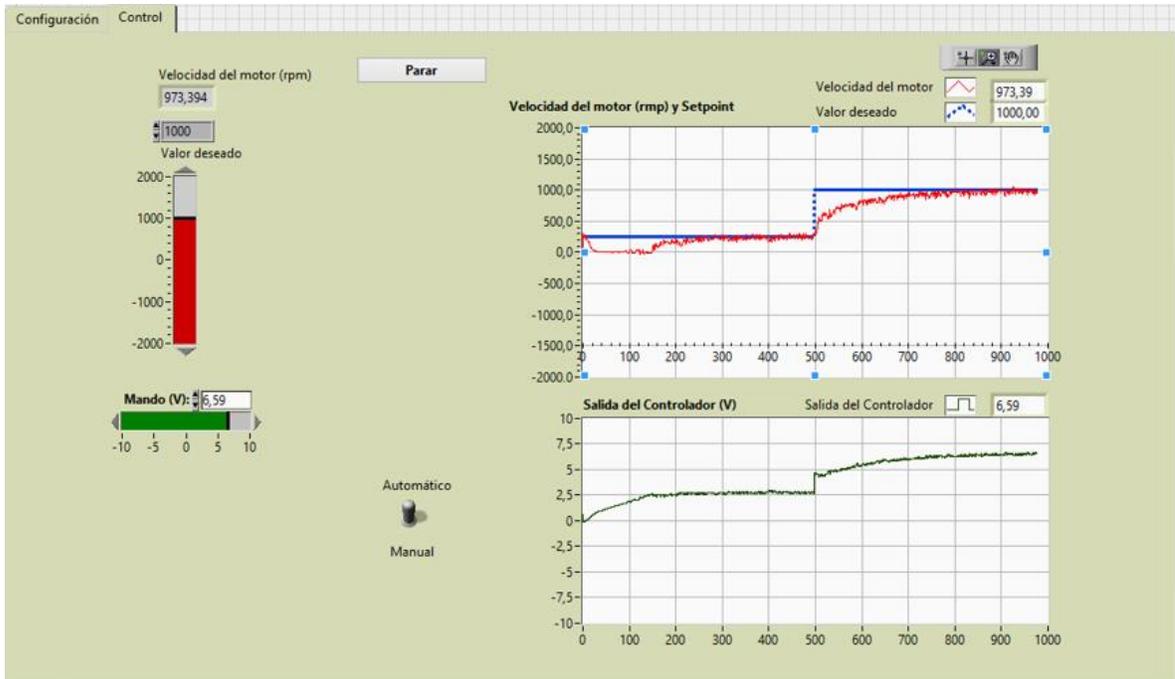


Fig. V.C Prueba de tiempo integral

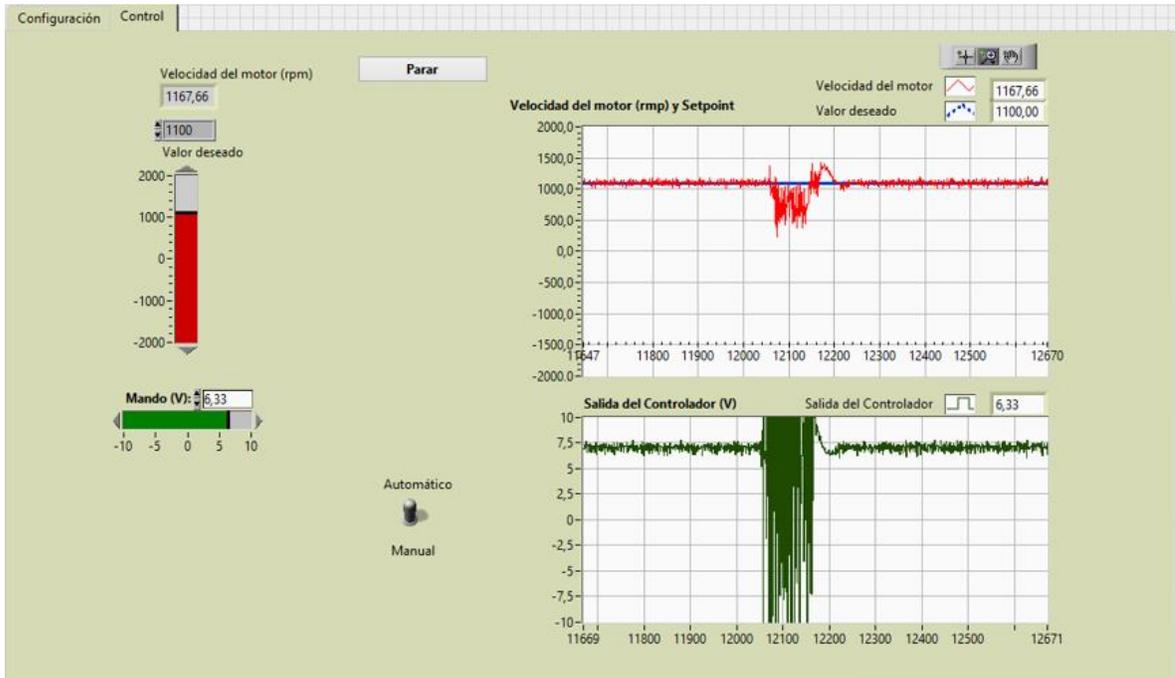


Fig. V.D Prueba de tiempo derivativo

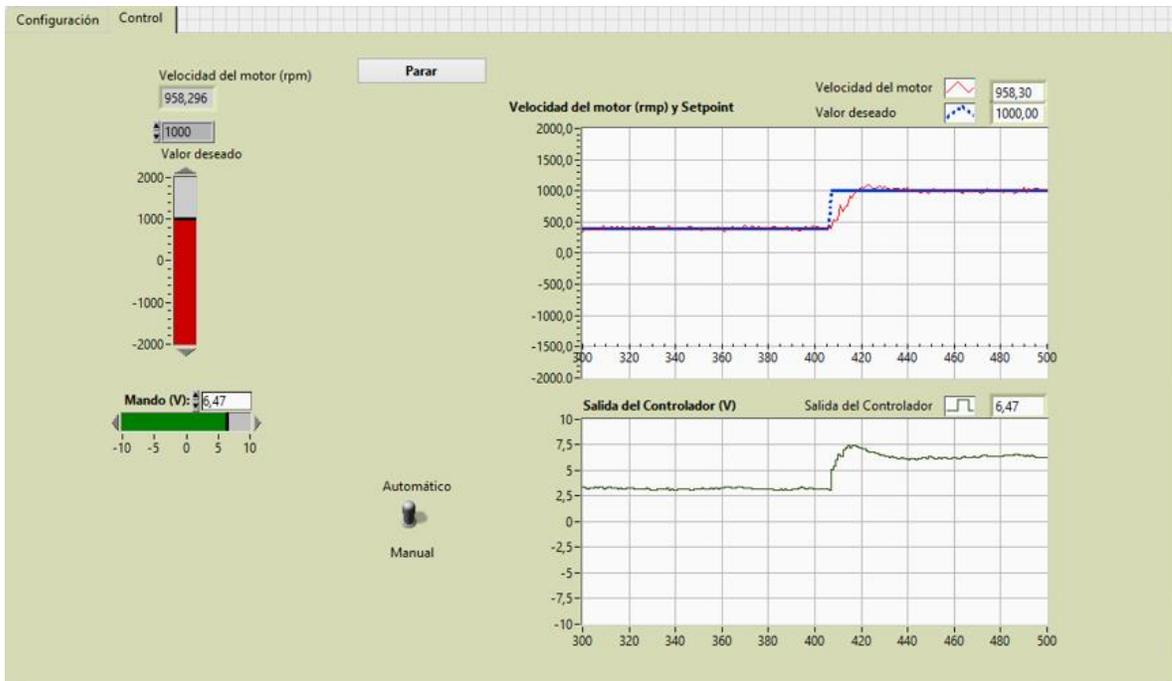


Fig. V.E Prueba del período de muestreo (10 ms)

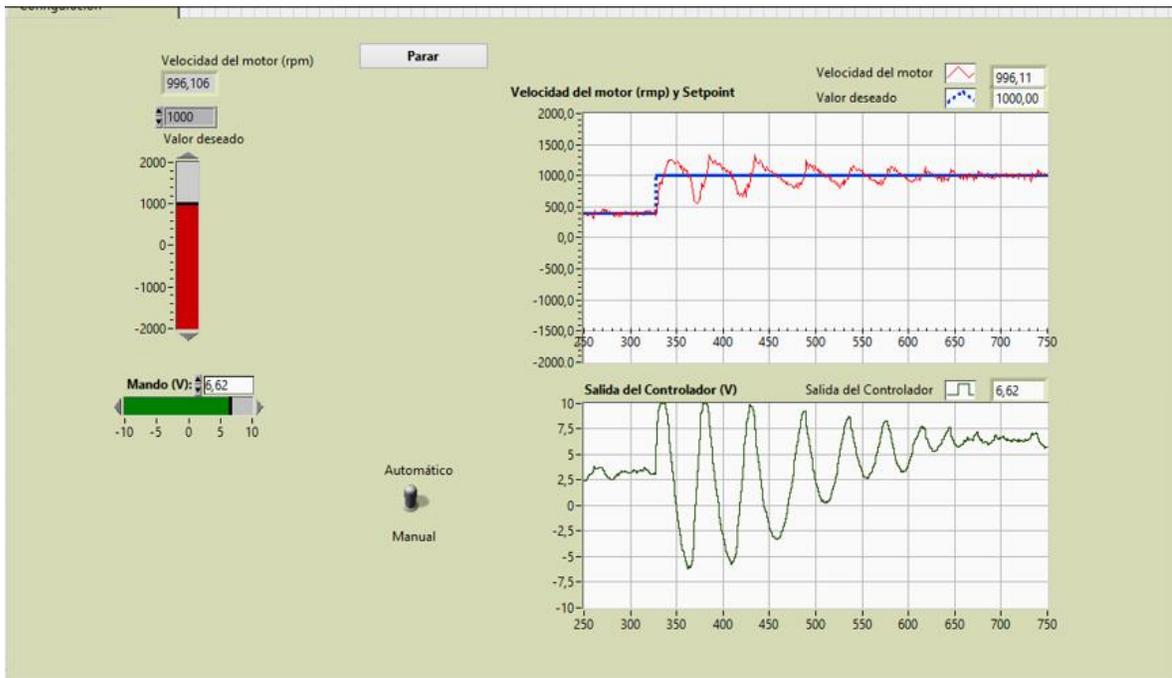


Fig. V.F Prueba del período de muestreo (30 ms)

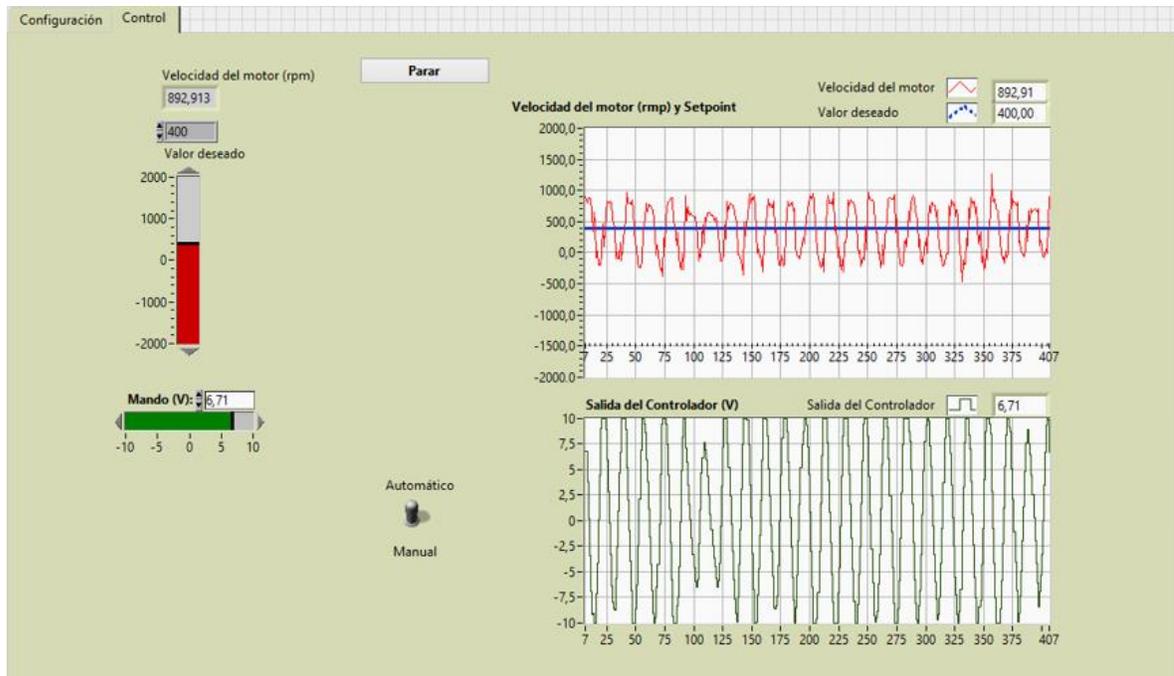


Fig. V.G Prueba del período de muestreo (60 ms)

Anexo VI Guía para práctica de laboratorio

Práctica de laboratorio

Título: Respuesta transitoria de un sistema de primer orden

Equipos y componentes: Para la realización de la práctica de laboratorio será necesario disponer de la maqueta ALECOPI SAD-100, la tarjeta de adquisición de datos NI USB- 6212, una computadora con los software LabVIEW 2010 o una versión superior y el SCD SAD-100 creado por Juan Alberto Martín Besada en su trabajo de diploma titulado “Software para el control automático digital del servocontrol SAD-100”.

Tiempo: 2 horas lectivas

Objetivos:

- Diseñar un controlador que posibilite que el sistema actuador-motor en lazo cerrado posea una buena respuesta transitoria. El modelo del mismo será suministrado por el profesor.
- Analizar la respuesta transitoria ante entrada escalón del sistema actuador-motor en lazo cerrado de manera simulada y real para un control de velocidad.

- Comparar los resultados obtenidos en las pruebas simuladas con los reales.

Introducción

- Marco teórico

El software SCD SAD-100 posee un enfoque dirigido a recrear el ambiente de trabajo que tienen los operadores en las industrias; por este motivo la interfaz de usuario se ha dividido en dos ventanas: una destinada a la configuración del controlador PID, los parámetros del proceso y la tarjeta de adquisición de datos, y la otra dedicada al control del servomotor.

La ventana de configuración, mostrada en la figura VI.A, posee en la parte superior, un “push button” que posibilita la selección de una de las dos formas en las que puede correr el software (real o simulado), un control numérico para elegir el período de muestreo y un bloque con tres controles numéricos que sirven para elegir las constantes de los filtros que se colocaron en el lazo de control de velocidad.

The screenshot displays the configuration interface for the SCD SAD-100 software, divided into several functional blocks:

- Real/Simulado:** A toggle button with a green indicator, currently set to 'Real'.
- Período de Muestreo dt (s):** A numeric input field set to 0,005.
- Unidad de Entrada:** A dropdown menu set to 'EGU'.
- Filtro paso bajo:** Three numeric input fields for filter constants, all set to 1.
- Controlador PID:**
 - Estructura PID:** A dropdown menu set to 'Académico (Normal)'.
 - Unidad Proporcional:** A dropdown menu set to 'Ganancia (Kc)'.
 - Proporcional:** Numeric input set to 0,0025.
 - Integral:** Numeric input set to 0,02.
 - Derivativo:** Numeric input set to 0.
 - rango de salida del controlador:** Two numeric input fields for 'salida máx' (10,00) and 'salida mín' (-10,00).
- Proceso:**
 - parámetros del proceso:** A table of process parameters:

Parameter	Value	Parameter	Value
ganancia estática	212,00	zona muerta	0,0
cte de tiempo (seg)	0,21	nivel de ruido	0,00
variación de carga	0,00	valor inicial de velocidad	0,00
 - Velocidad Máx:** Numeric input set to 2000.
 - Velocidad Min:** Numeric input set to -2000.
- Configuración de la tarjeta de adquisición de datos:**
 - Valor Mínimo de Entrada:** Numeric input set to -10,00.
 - Valor Mínimo de Salida:** Numeric input set to -10,00.
 - Valor Máximo de Entrada:** Numeric input set to 10,00.
 - Valor Máximo de Salida:** Numeric input set to 10,00.
 - Canal Físico de Entrada:** Dropdown menu set to 'Dev1/ai1'.
 - Canal Físico de Salida:** Dropdown menu set to 'Dev1/ao0'.
 - configuración del terminal de entrada:** Dropdown menu set to 'RSE'.

Fig. VI.A Ventana de configuración del software SCD SAD-100

Esta ventana también posee un bloque encomendado al ajuste del regulador PID, junto a una estructura para limitar el mando y así evitar la saturación del actuador. Otro de los bloques

es el del proceso, que contiene sus parámetros y dos indicadores numéricos de las velocidades máxima y mínima respectivamente, que limitan la variable de salida, por lo que previenen la saturación de la planta. El último bloque es el de la configuración de los parámetros de entrada y salida de la tarjeta de adquisición de datos empleada.

La ventana de control, mostrada en la figura VI.B, posee un indicador numérico que muestra la velocidad del motor CD en rpm, un control numérico con el valor deseado de velocidad, una barra vertical deslizante con la se puede cambiar el valor deseado, una barra horizontal deslizante que en modo automático funciona como indicador del mando y en manual posibilita realizar cambios en esta variable, un botón para detener la ejecución del software, ya sea real o simulado.

Esta ventana incluye un botón para alternar entre los modos manual y automático según convenga, una gráfica que permite analizar conjuntamente el comportamiento de la velocidad del motor y el *setpoint* a medida que se muestrea la salida del sistema. Otro elemento de la ventana de control es una gráfica que muestra la señal de salida del regulador contra la cantidad de muestras tomadas por la tarjeta de adquisición de datos.



Fig. VI.B Ventana de control del software SCD SAD-100

- Trabajo previo

Estudiar las conferencias impartidas por el profesor de Ingeniería de Control I que aborden el contenido tratado en esta práctica de laboratorio. Pueden consultar la literatura existente, como el libro Ingeniería de Control Moderna del autor Katsuhiko Ogata.

Parte práctica

La actividad se dividirá en dos momentos. En el primero los estudiantes realizarán pruebas simuladas, en un segundo momento se efectuarán las mismas pruebas de manera real con la guía del profesor, quien deberá montar y explicar el circuito para la adquisición de los datos.

Parte simulada:

1) Pruebas con el software SCD SAD-100

- Abra el LabVIEW y cargue el software SCD SAD-100.
- Compruebe que el botón Real/Simulado esté de color verde, para que el software se ejecute de forma simulada.
- Ajuste un controlador proporcional para diferentes valores de ganancia, ejecute y observe la respuesta transitoria del sistema, analice los aspectos siguientes en cada caso y determine:
 - El tipo de respuesta del sistema según el amortiguamiento.
 - El tiempo de asentamiento (t_s), no debe ser muy elevado.
 - Los tiempos de subida (t_r), el de retardo (t_d) y el pico (t_p).
 - El error en estado estable (E_{ss}).

Observación: vigile que la señal de mando no se sature.

- Ajuste un controlador PI para diferentes valores de ganancia y tiempo integral (T_i), ejecute y analice la respuesta transitoria del sistema, analice los aspectos anteriores y los siguientes:
 - El máximo sobreimpulso (M_p) debe estar entre el 5 y el 25%.
 - Compruebe que el error en estado estable desaparece con la acción integral.
 - Compruebe como esta acción acelera la velocidad de respuesta del sistema.

Parte real:

1) Pruebas con el software SCD SAD-100

- Abra el LabVIEW y cargue el software SCD SAD-100.
- Presione el botón Real/Simulado para que el software se ejecute de forma real.
- Realice las mismas pruebas simuladas y compare ambos resultados.

Informe

Cada estudiante deberá entregar un informe en formato digital en el que realizará:

1. Una valoración de cada una de las pruebas efectuadas en el laboratorio, tanto de las simuladas como las reales.
2. Una comparación de los resultados de las pruebas simuladas con los resultados de las reales.
3. Completará las dos tablas mostradas a continuación. El informe contendrá un total de cuatro tablas, dos que contengan los resultados de las pruebas simuladas y dos con los resultados de las pruebas reales.

Tabla VI.1 Resultados de la prueba con el regulador P

Ganancia Proporcional	Tipo de respuesta	t_s (seg)	t_r (seg)	t_d (seg)	t_p (seg)	Ess (rpm)

Tabla VI.2 Resultados de la prueba con el regulador PI

Ganancia Proporcional	Ti (seg)	Tipo de respuesta	t_s (seg)	t_r (seg)	t_d (seg)	t_p (seg)	Ess (rpm)