



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA EN CONTROL Y AUTOMATISMO

TEMA:

Diseño de un módulo de simulación para el control de posicionamiento y de velocidad de un motor DC utilizando control predictivo con supervisión a través de un sistema SCADA.

AUTOR:

DOMÍNGUEZ REYES, ALEX FRANCISCO

Trabajo de Titulación previo a la obtención del título de
INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO

TUTOR:

MENDOZA MERCHAN, EDUARDO VICENTE

Guayaquil, Ecuador

13 de septiembre del 2017



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr.
Domínguez Reyes, Alex Francisco como requerimiento para la obtención
del título de **INGENIERO ELECTRÓNICO EN CONTROL Y
AUTOMATISMO.**

TUTOR

Mendoza Merchán, Eduardo Vicente

DIRECTOR DE CARRERA

Heras Sánchez, Miguel Armando

Guayaquil, a los 13 del mes de septiembre del año 2017



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

DECLARACIÓN DE RESPONSABILIDAD

Yo, **DOMÍNGUEZ REYES, ALEX FRANCISCO**

DECLARÓ QUE:

El trabajo de titulación “Diseño de un módulo de simulación para el control de posicionamiento y de velocidad de un motor DC utilizando control predictivo, con supervisión a través de un sistema SCADA” previo a la obtención del Título de **Ingeniero Electrónico en Control y Automatismo**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 13 del mes de septiembre del año 2017

EL AUTOR

Domínguez Reyes, Alex Francisco



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

AUTORIZACIÓN

Yo, **Domínguez Reyes, Alex Francisco**

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: “Diseño de un módulo de simulación para el control de posicionamiento y de velocidad de un motor DC utilizando control predictivo, con supervisión a través de un sistema SCADA”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 13 del mes de septiembre del año 2017

EL AUTOR

Domínguez Reyes, Alex Francisco

REPORTE DE URKUND

The screenshot displays the URKUND interface. On the left, a document summary is shown: 'Documento: TT-Alex Dominguez - Final.docx (D30493088)', 'Presentado: 2017-09-11 18:52 (-05:00)', 'Presentado por: aljr94@hotmail.com', 'Recibido: eduardo.mendoza01.ucsg@analysis.orkund.com', and 'Mensaje: Trabajo de Titulación - Alex Dominguez - Corregido 2. 2% de estas 50 páginas, se componen de texto presente en 13 fuentes.' The right panel, titled 'Lista de fuentes', contains a table of sources with columns for 'Categoria' and 'Enlace/nombre de archivo'. The table lists various sources with associated similarity percentages (77%, 90%, 90%, 60%, 67%, 62%). Below the table, a toolbar includes icons for search, navigation, and actions like 'Reiniciar', 'Exportar', and 'Compartir'. A yellow warning banner at the bottom right indicates '1 Advertencias'. The main content area shows a vertical list of text extracted from the sources, including university names and technical details. At the bottom, a summary bar shows '77%' and '#2 Activo'.

Categoria	Enlace/nombre de archivo
	TESIS VALERIANO-28-Agosto.docx
	Proyecto de graduacion Diseño del proyecto ...
77%	Guayaquil, Ecuador 2014
90%	UNIVERSIDAD CATOLICA DE SANTIAGO DE GU...
90%	UNIVERSIDAD CATOLICA DE SANTIAGO DE GU...
60%	UNIVERSIDAD CATOLICA DE SANTIAGO DE GU...
67%	previo a la obtención del Título de Ingeniero ...
62%	cuyo contenido, ideas y criterios son de mi ex...
	https://documents.mx/documents/apuntes-...
	http://www.buenastareas.com/ensayos/Mag...

TUTOR

Mendoza Merchán, Eduardo Vicente

AGRADECIMIENTO

Agradezco primeramente a Dios por haberme dado la oportunidad de mi existencia, llena de sabiduría para ver cumplida la meta que me propuse al comenzar la carrera de mis estudios superiores.

Agradezco a mis Padres Alex Domínguez y Mónica Reyes que siempre estuvieron atentos y prestos a brindarme su apoyo, transmitiéndome su amor, confianza y seguridad en los momentos más difíciles que se presentan en la vida de estudiante.

Agradezco a toda mi familia que de una u otra manera me dieron su apoyo y algunos hasta consejos que estimularon mi ser.

A mis hermanas Rommyna y Dasha Domínguez por su comprensión y Apoyo.

No puedo pasar por alto, un reconocimiento especial a mi querida novia, Ninoska Méndez que siempre estuvo acompañándome durante la realización de mi trabajo, a sus padres por comprenderme y quererme mucho.

Alex Francisco

DEDICATORIA

Este trabajo de mi Tesis de Grado, la dedico a mis padres Dr. Alex Domínguez y Dra. Mónica Reyes, así como a mis queridas hermanas Rommyna Alejandra y Dasha Rommyna, que con mucha paciencia, amor y comprensión me supieron apoyar en todo momento.

Alex Francisco



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

TRIBUNAL DE SUSTENTACIÓN

f. _____

HERAS SÁNCHEZ, MIGUEL ARMANDO
DIRECTOR DE CARRERA

f. _____

PALACIOS MELÉNDEZ, EDWIN FERNANDO
COORDINADOR DE ÁREA

f. _____

VEGA URETA, NINO TELLO
OPONENTE

ÍNDICE GENERAL

Índice de figuras.....	XIII
Resumen	XVI
Introducción	1
CAPÍTULO I: ASPECTOS GENERALES.....	3
1.1.Planteamiento del Problema.....	3
1.2.Justificación	4
1.3.Delimitación del Problema	4
1.4.Objetivos.....	4
1.4.1. Objetivo General.....	5
1.4.2. Objetivo Específicos.....	5
1.5.Hipótesis	5
1.6. Tipo de Investigación y Metodología de Investigación.....	6
CAPÍTULO II: MARCO TEÓRICO	7
2.1. Estado del Arte	7
2.2. Control predictivo	14
2.2.1. Ventajas del control predictivo.....	16
2.2.2. Estrategia de control.....	17
2.2.3. Elementos básicos del control predictivo.....	19
2.2.3.1. Modelo de predicción	19
2.2.3.2. Función objetivo	25
2.2.3.3. Obtención de la ley de control	27
2.3. Plataforma de simulación.....	28
2.3.1. Descripción de la plataforma	28
2.3.2. Funcionalidades	29
2.3.3. Requisitos del sistema operativo Windows 10 para instalar Matlab	30
2.3.4. Descripción del entorno virtual	30
2.3.5. Librerías del entorno Matlab.....	32
2.3.6. Tipos de datos.....	33
2.3.7. Simulink.....	33

2.3.8. Entorno Simulink	34
2.3.9. Librerías Simulink	35
2.4. Sistemas supervisorios	36
2.4.1. Control supervisorio.....	37
2.4.2. Concepto de control	38
2.4.3. Sistemas de visualización	39
2.4.4. Sistemas SCADA	40
2.4.5. Objetivos de los sistemas SCADA.....	41
2.4.6. Componente de un sistema SCADA	42
2.4.7. Software Labview	42
2.4.7.1. Principales funcionalidades del software Labview.....	43
2.4.7.2. Interfaz de comunicación del software Labview	44
2.5. Motores DC.....	44
2.5.1. Características constructivas.....	44
2.5.2. Ventajas y desventajas de los motores de corriente continua.....	46
2.5.3. Clasificación de los motores de corriente continua	47
2.5.4. Modelado matemático de los motores DC.....	47
CAPÍTULO III: DISEÑO DE MÓDULO DE SIMULACIÓN APLICANDO CONTROL PREDICTIVO.....	51
3.1. Consideraciones iniciales del diseño del sistema.....	51
3.1.1 Descripción del módulo de simulación	51
3.1.2. Características del módulo de simulación	53
3.1.3. Funciones.....	55
3.2 Análisis de las plataformas de simulación	55
3.2.1. Análisis de la plataforma virtual Matlab	55
3.2.1.1. Script	56
3.2.2. Análisis del entorno grafico de programación Simulink.....	56
3.2.2.1. Transfer Fcn.....	57
3.2.2.2. MPC toolbox.....	57
3.2.2.3. Serial configuration.....	57
3.2.2.4. Serial receive.....	58
3.2.2.5. Scope	59

3.2.2.6. Diseño de la etapa en matlab	59
3.2.3. Análisis de la plataforma virtual labview	60
3.2.3.1. Front panel	60
3.2.3.2. Block diagram.....	61
3.2.3.3. While loop.....	62
3.2.3.4. Numeric data	62
3.2.3.5. Build array	63
3.2.3.6. Byte array to string	63
3.2.3.7. Visa serial port.....	64
3.2.3.8. Visa write	64
3.2.3.9. Visa open	64
3.2.3.10. Visa close	65
3.2.3.11. Wait until next ms multiple.....	65
3.2.4. Análisis de los puertos virtuales	68
3.3. Análisis de algoritmos	68
3.3.1 Análisis del modelo matemático de la planta.....	69
3.3.1.2. Modelo matemático para el control de velocidad de un motor DC	70
3.2.3. Análisis y diseño del módulo de simulación.	72
3.2.3.1. Matlab.....	72
3.2.3.2. Simulink.....	72
3.2.3.3. Labview	74
3.2.3.4. Virtual serial port.....	75
3.4. Pruebas de funcionamiento	75
3.4.1. Consideraciones iniciales	75
3.4.2. Pruebas de simulación para el control de posicionamiento.....	75
3.4.3. Pruebas de simulación para el control de velocidad	86
CAPÍTULO IV: DISCUSION DE RESULTADOS.....	99
4.1. Resultados del módulo de simulación.....	99
4.1.1. Resultados de la plataforma Matlab	99
4.1.2. Resultados de la plataforma Labview.....	100
4.1.3. Resultados de software virtual serial port.....	101

4.2 Discusión de resultados.....	101
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.....	103
5.1. Conclusiones	103
5.2. Recomendaciones	104
5.3. Trabajo futuro.....	104
Referencias Bibliográficas	106

ÍNDICE DE FIGURAS

CAPÍTULO 2

Figura 2.1: Estrategia de control predictivo.....	17
Figura 2.2: Estructura general del control predictivo basado en modelo.	18
Figura 2.3: Modelo de predicción.....	20
Figura 2.4: Respuesta al impulso.....	21
Figura 2.5: Respuesta al escalón.....	21
Figura 2.6: Trayectoria de referencia.	27
Figura 2.7: Ventana de inicio del entorno virtual Matlab.	31
Figura 2.8: Ventana del Workspace.....	31
Figura 2.9: Ventana command history.	32
Figura 2.10: Acceso entorno Simulink.	34
Figura 2.11: Ventana de Simulink.....	34
Figura 2.12: Acceso librería Simulink.....	35
Figura 2.13: Ventana de librería Simulink.	36
Figura 2.14: Primer PLC – Modicon 084.....	37
Figura 2.15: Representación en lazo abierto de un sistema.	38
Figura 2.16: Representación en lazo cerrado de un sistema.....	38
Figura 2.17: Cuarto de control de una antigua central nuclear.	40
Figura 2.18: Partes que conforman el estator de un motor DC.....	45
Figura 2.19: Partes que conforman el rotor de un motor DC	46
Figura 2.20: Esquema del circuito perteneciente a un motor DC.....	47

CAPÍTULO 3

Figura 3.1: Diagrama de flujo del funcionamiento del módulo de simulación.....	52
Figura 3.2: Datos de la planta en el Workspace.	56
Figura 3.3: Bloque transfer fcn de Simulink.	57
Figura 3.4: Descripción del bloque MPC tool de Simulink.	57
Figura 3.5: Bloque serial configuration.....	58
Figura 3.6: Bloque serial receive.....	59
Figura 3.7: Bloque de visualización Scope.	59
Figura 3.8: Controlador mpc del entorno gráfico Simulink.	60
Figura 3.9: Ventana front panel de Labview.	61

Figura 3.10: Ventana block diagram de Labview.	61
Figura 3.11: Ciclo while loop.....	62
Figura 3.12: Bloque numeric data.....	62
Figura 3.13: Bloque build array.....	63
Figura 3.14: Bloque byte array to string.....	63
Figura 3.15: Bloque visa serial port.....	64
Figura 3.16: Bloque visa write.....	64
Figura 3.17: Bloque visa open.....	65
Figura 3.18: Bloque visa close.....	65
Figura 3.19: Bloque wait until next ms multiple.....	66
Figura 3.20: Configuración de la interfaz en la plataforma Labview.....	67
Figura 3.21: Interfaz para el control de posicionamiento.....	67
Figura 3.22: Interfaz para el control de velocidad.....	67
Figura 3.23: Ventana del software virtual serial port driver.....	68
Figura 3.24: Diagrama de flujo del desarrollo del proyecto en Matlab.....	73
Figura 3.25: Diagrama de flujo del desarrollo del proyecto en Simulink.....	73
Figura 3.26: Diagrama de flujo del desarrollo del proyecto en Labview.....	74
Figura 3.27: Respuesta de la planta conectad en lazo abierto.....	76
Figura 3.28: Respuesta de la planta conectada en lazo cerrado.....	77
Figura 3.29: Comportamiento de la planta, simulación 1.....	78
Figura 3.30: Comportamiento de la planta, simulación 2.....	79
Figura 3.31: Comportamiento de la planta, simulación 3.....	80
Figura 3.32: Comportamiento de la planta, simulación 4.....	81
Figura 3.33: Comportamiento de la planta, simulación 5.....	82
Figura 3.34: Comportamiento de la planta, simulación 6.....	83
Figura 3.35: Comportamiento de la planta, simulación 7.....	84
Figura 3.36: Comportamiento de la planta, simulación 8.....	85
Figura 3.37: Comportamiento de la planta, caso con perturbaciones.....	86
Figura 3.38: Respuesta de la planta conectada en lazo abierto.....	87
Figura 3.39: Respuesta de la planta conectada en lazo cerrado.....	87
Figura 3.40: Comportamiento de la planta, simulación 1.....	89
Figura 3.41: Comportamiento de la planta, simulación 2.....	90
Figura 3.42: Comportamiento de la planta, simulación 3.....	91
Figura 3.43: Comportamiento de la planta, simulación 4.....	92

Figura 3.44: Comportamiento de la planta, simulación 5.	93
Figura 3.45: Comportamiento de la planta, simulación 6.	95
Figura 3.46: Comportamiento de la planta, simulación 7.	96
Figura 3.47: Comportamiento de la planta, simulación 8.	97
Figura 3.48: Comportamiento de la planta, caso con perturbaciones.	98

CAPÍTULO 4

Figura 4.1: Configuración del MPC en el entorno Matlab.	99
Figura 4.2: Interfaz de sistema para el control de posicionamiento.	100
Figura 4.3: Interfaz de sistema para el control de velocidad.	100
Figura 4.4: Pantalla principal del virtual serial port driver.	101

Resumen

El presente trabajo de titulación se fundamenta en el control predictivo basado en modelo (MPC), aplicado al control de posicionamiento y de velocidad de un motor DC, a través de un módulo de simulación. El módulo de simulación está implementado en base a las plataformas de simulación Matlab y LabVIEW, las cuales brindan las herramientas necesarias para simular el control MPC tanto para el control de posicionamiento como el control de velocidad. El módulo de simulación posee una interfaz HMI a través de un computador, implementado en la plataforma de simulación LabVIEW. El siguiente trabajo de investigación se fundamenta tanto en la metodología bibliográfica como en la metodología experimental, mediante la cual encierra los fundamentos teóricos del control predictivo basado en modelo, aplicado a la función de transferencia de una planta basada en datos reales y el funcionamiento del sistema se realiza a través de una plataforma de simulación, el cual permite visualizar y controlar el proceso a través de una interfaz HMI virtual. Concluido el módulo de simulación, se puede observar que el comportamiento de la planta sometida al control predictivo basado en modelo permite contrastar los fundamentos teóricos con la práctica de simulación, evidenciando que el control MPC presenta mejores resultados en plantas con un comportamiento lineal, y forzando la salida a estabilizarse en un Setpoint, debido a las restricciones, característica principal que diferencia al MPC de las diferentes técnicas de control. Como punto final, la implementación de técnicas de control predictivo requiere una amplia capacidad de procesamiento, puesto que se requiere calcular constantemente las variables de control en cada instante de muestreo. Como ventajas del control predictivo se evidencia una mejor eficiencia del sistema, robustez, posee un menor tiempo de respuesta, previene las variaciones del sistema provocadas por perturbaciones y ruidos entre otras. Sobre todo, en sistemas donde las técnicas de control clásicas no surgen efecto.

Palabras Clave: MATLAB; LABVIEW; MPC; SETPOINT; HMI; TIEMPO DE RESPUESTA.

INTRODUCCIÓN

La necesidad de implementar control automático en los diversos sistemas que nos rodean data del siglo III antes de cristo mediante sistemas mecánicos como el reloj de agua de Ktesibios o la regulación del nivel de aceite en una lámpara por parte de Philon, así como en el siglo I antes de cristo, el dispensador de vino de Herón, catalogado como uno de los primeros sistemas retroalimentados de la historia, estas técnicas han llevado al ser humano a desarrollarse tecnológicamente, con el fin de mejorar la calidad de vida, obtener productos de mayor calidad con eficiencia y optimizar el rendimiento de las máquinas y consumo energético.

Con la llegada de los semiconductores en 1948, los grandes bulbos de vacío fueron reemplazados por transistores permitiendo que los diseños fueran más pequeños y con menos consumo de potencia. El primer CI (circuito Integrado) se creó en 1958, constituido por seis transistores en un chip. El primer uP (microprocesador) se desarrolló en 1970 por Intel. Por lo tanto los campos de la electrónica son amplios y es una de las ramas de la ingeniería con mayor proyección ya que la tecnología avanza constantemente.

La electrónica de control es una de las aplicaciones en el campo de la ingeniería eléctrica, que dependiendo del comportamiento del sistema a controlar se puede aplicar técnicas de control clásico o técnicas de control moderna. Las técnicas de control moderno como el control predictivo basado en modelo, posee grandes ventajas al momento de controlar tanto sistemas lineales como no lineales, este tipo de controlador interviene en procesos donde las técnicas de control clásicas no se ajustan al sistema, esto se logra debido a la característica principal que posee el MPC, el cual se basa en restricciones que no permiten que la variable de salida presente cambios.

Los desarrolladores de software, has hecho posible el diseño, desarrollo y simulación de cualquier tipo de sistema en plataformas de simulación virtual, lo cual son herramientas poderosas inmersas en el campo de la ingeniería eléctrica y electrónica. Una de las plataformas de simulación más utilizada es Matlab, la cual posee un lenguaje de programación escrita y un lenguaje grafico de bloques el cual nos permite ingresar datos de la planta en base a

un modelo real, para posteriormente ver su comportamiento de forma gráfica. Otra de las plataformas de simulación es el VIS de LabVIEW, el cual nos permite crear interfaz HMI entre el usuario y el proceso, posee un lenguaje gráfico de programación, caracterizado por su facilidad de uso.

En el mundo de la industria se debe estar constantemente innovando e implementando mejoras al sistema, aplicando nuevas técnicas de control que permita obtener un sistema más eficiente. En base a esto se llegó al desarrollo del siguiente trabajo de investigación, el cual tiene como objetivo principal desarrollar un módulo de simulación para el control de posicionamiento y de velocidad de un motor DC, aplicando técnicas de control predictivo, implementado en un entorno virtual y supervisado a través de un interfaz humano máquina.

El siguiente documento describe específicamente, todo el trabajo realizado para la implementación del módulo de supervisión basado en control predictivo, aplicado al control de posicionamiento y de velocidad de un motor DC, sujeto a simulación en una plataforma virtual, dando como resultado el comportamiento de la planta sometida a un control MPC, supervisado a través de una interfaz virtual por medio de un computador.

El siguiente trabajo de investigación se espera que sea de gran ayuda, en futuras investigaciones con respecto a técnicas de control moderna específicamente MPC, puesto que en nuestro entorno este tipo de controladores es muy poco conocido e implementado. Adicionalmente se incentiva a la investigación del funcionamiento de otras técnicas de control moderna para ser aplicadas en sistemas ya sea de tipo lineales o no lineales.

Se invita a seguir con la lectura de los capítulos ya que se describe con mayor detalle el tema de investigación propuesto.

CAPÍTULO I: ASPECTOS GENERALES

En este capítulo se abordan aspectos fundamentales que intervinieron para definir el tema del trabajo de investigación planteado. Como primer punto se aborda la problemática a tratar en el ámbito de las técnicas de control moderna, específicamente en control predictivo, posteriormente se plantea la debida justificación al problema en la cual se debe abordar una solución efectiva a la problemática planteada, luego se definen los objetivos que dan forma al trabajo de investigación, consecuentemente se debe plantear la hipótesis relacionado con los aspectos de la investigación, finalmente se describe la metodología de investigación a utilizar en el trabajo de titulación, todos estos aspectos son pilares fundamentales, para dar inicio con rigor al proceso de investigación, los cuales permiten desarrollar el tema planteado en este trabajo de titulación.

1.1. Planteamiento del Problema

En la actualidad se tiene un mundo muy competitivo, donde la experiencia es un factor fundamental para el crecimiento laboral de un ingeniero electrónico, más aun en la rama del control y la automatización, ya que cada uno de los sistemas funcionan de maneras diferentes por lo tanto las técnicas de control deben ser acorde al proceso; de las teorías de control moderna, se ha escogido al control predictivo como tema principal a tratar ya que normalmente en las aulas de clases no se imparten estos fundamentos teóricos debido a razones académicas; otro aspecto fundamental es que interviene tanto en sistemas lineales como no lineales, por lo tanto se ajusta a los requerimientos que demanda la industria actual, para analizar las ventajas del control predictivo se ha optado por la implementación de un módulo de simulación, para visualizar el comportamiento del sistema, tanto para el control de posición como para el control de velocidad de un motor DC, en un entorno virtual y un sistema supervisorio SCADA.

1.2. Justificación

El control y la automatización se han vuelto fundamentales en el mundo de la industria, ya que permite monitorear y controlar diversos sistemas, optimizando el rendimiento de las máquinas y obteniendo un producto final con eficiencia. Las técnicas de control moderno, en especial el control predictivo, son fundamentos teóricos que normalmente no son impartidos en las aulas de clases, debido a diversas razones académicas. En los últimos años se están implementando estas técnicas de control por parte de la industria ya que presenta varias ventajas. Ya que se busca aplicar estas técnicas en los procesos automatizados, para mejorar la funcionalidad de los sistemas, se ha planteado el diseño de un módulo de simulación basado en un entorno virtual, donde se implementará el control predictivo, para analizar las ventajas que nos brinda esta técnica de control moderna, aplicada al control de posición y de velocidad de un motor DC, mediante una plataforma virtual y un sistema SCADA a través de un ordenador.

1.3. Delimitación del Problema

Este trabajo se centra en la aplicación del control predictivo basado en modelos en control de procesos. Para lo cual se desarrolla un módulo de simulación para el control tanto de posicionamiento como de velocidad de un motor DC, implementado en una plataforma virtual con es Matlab Simulink y a través de un ordenador, que dispone de un sistema supervisorio para mando y monitoreo del proceso como es LabVIEW.

1.4. Objetivos

Los objetivos que se muestran a continuación guiarán el presente trabajo de investigación.

1.4.1. Objetivo General

Diseño de un módulo de simulación para el control de posicionamiento y de velocidad de un motor DC implementado en una plataforma de simulación para el análisis de las técnicas de control predictivo basado en modelo.

1.4.2. Objetivo Específicos

- Revisar los aportes más significativos relacionados a la implementación de controladores utilizando control predictivo basado en modelo, por medio de una búsqueda de bibliografía actualizada.
- Diseñar un módulo de simulación que permita implementar el control de posición y de velocidad de un motor DC, a través de una plataforma virtual y supervisada con una Interface Humano máquina.
- Implementar un controlador aplicando control predictivo en un módulo de simulación para el control de posición y de velocidad de dos motores DC, que permita realizar pruebas experimentales de precisión y funcionalidad.

1.5. Hipótesis

Por medio del control predictivo se espera obtener un sistema más eficiente, para que el control de posición y de velocidad de un motor DC se efectúe con precisión, esto se realizará mediante las plataformas virtual Matlab en la cual se ingresaran los datos de la planta y se le aplicara un controlador MPC para el control del sistema, la comunicación con la interfaz LabVIEW se efectuara a través de puertos virtuales generados por el software Virtual Serial Port. Usando una herramienta de trabajo virtual, podremos visualizar el comportamiento del sistema, de forma gráfica. Mediante la interfaz LabVIEW se podrá controlar el sistema y visualizar la información con el fin de crear un sistema más eficiente.

En este trabajo de investigación se plantea que:

“Utilizando técnicas de control predictivo se podrá controlar la posición de un motor DC con un tiempo de estabilización menor en comparación con el control de velocidad de un motor DC”.

1.6. Tipo de Investigación y Metodología de Investigación

La implementación de técnicas de control moderna, especialmente el control predictivo aplicado a procesos, está siendo mayormente considerado por las industrias en la actualidad, por el motivo de que estos fundamentos teóricos no son impartidos normalmente en las aulas de clase y son temas poco tratados en los trabajos de titulación, se ha decidido efectuar un trabajo de investigación de tipo correlacional con un enfoque cuantitativo, en el cual se requiere medición y análisis de los datos numéricos obtenidos, junto con los ajustes y pruebas se obtendrá el correcto funcionamiento del trabajo efectuado.

Para poder entender los fundamentos teóricos se hicieron uso de las metodologías bibliográficas y experimentales de trabajos relacionados que se han efectuado en los últimos años.

En base a la metodología propuesta se ha planteado el siguiente procedimiento para llevar a cabo el trabajo de titulación:

1. Análisis del problema orientado al trabajo de investigación y examinar las posibles soluciones.
2. Búsqueda de información vinculada al trabajo de investigación, ya sea en libros, revistas científicas, páginas web.
3. Definir la función general del método de control predictivo y relacionarlas con la ecuación característica de los motores DC para implementar en el módulo didáctico de prácticas.
4. Determinar el sistema de control a través de un diagrama de flujo.
5. Análisis y desarrollo del sistema SCADA.
6. Pruebas experimentales de funcionamiento del módulo de simulación.
7. Análisis de los resultados obtenidos.

CAPÍTULO II: MARCO TEÓRICO

El marco teórico desarrollado para esta investigación proporciona un enfoque requerido sobre el tema de investigación, dándole una importancia a los fundamentos teóricos que permiten estructurar un modelo científico, en función a teorías enlazadas con el tema de investigación. En este capítulo se ha recopilado una base teórica fundamental relacionada con el tema de investigación que han sido desarrollados en los últimos cinco años, que permite aplicar los conceptos y técnicas de control predictivo, aplicado al control de posicionamiento y de velocidad de un motor DC, y el diseño de una interfaz gráfica para el mando del sistema a través de un ordenador.

2.1. Estado del Arte

En nuestro entorno las técnicas de control de procesos no son explotadas en su totalidad, ya que el controlador PID se ha convertido en la herramienta de regulación de variables de procesos más utilizada, tanto en sistemas dinámicos lineales y no lineales, por lo tanto este tipo de controladores no se ajusta a muchos de los requerimientos que demanda la industria actual, por tal motivo en el estado del arte se ha procurado reunir información basada en la teoría de control moderna, específicamente en control predictivo basado en modelos, para ver las ventajas y desventajas que ofrece este tipo de control implementándolo en el control de posicionamiento y de velocidad de dos motores DC; cuyos trabajos se hayan efectuado en los últimos 5 años.

Según (Verne, Gonzales, & Valla, 2014) en su trabajo titulado “Motor de Inducción Accionado por CAMC Control Predictivo”. Trata sobre una estrategia de control para un motor de inducción alimentado por un convertidor asimétrico en cascada (CAMC) basándose en la técnica de control predictivo del modelo de estados finitos, utiliza el modelo de motor dinámico y el modelo de convertidor para pre-calcular la evolución de las variables de interés, para comparar conjuntamente con sus valores de referencia mediante una función

de optimización, permitiendo determinar el mejor estado entre todas las combinaciones de conmutación. El controlador desarrollado es capaz de realizar la regulación del par y del flujo magnetizante de la máquina, y, simultáneamente regular las tensiones internas del convertidor. Se establece un valor de tensión para los capacitores flotantes que permite aumentar el número de niveles de 5 a 7 sin aumentar el número de componentes.

Según (Días Dorado, 2015) en su proyecto final de grado titulado “Control Predictivo Basado en Modelo Aplicado a un Motor de Inducción”. El siguiente trabajo de titulación tiene como fin relacionar las estrategias de control aplicadas a un motor de inducción, como punto principal se debe abordar la técnica de control predictivo basado en modelo, abordando los fundamentos teóricos tanto para la función de coste, como para la estabilidad. Para corroborar los resultados obtenidos se propone el diseño eléctrico de una máquina asíncrona constituida por un rotor de jaula de ardilla, adicionalmente se incorpora un inversor de dos niveles, aprovechando las ventajas de la plataforma de simulación Matlab y su entorno gráfico Simulink. Como resultado se obtuvo una mejora en la función de coste en la etapa del diseño, en cambio no se pudo conseguir estabilidad el sistema.

Según (Perez Losada, 2014) en su paper titulado “Diseño e implementación de un controlador predictivo para motores BLDC”. La potencia de los motores BLDC es una de las ventajas principales debido a sus altas velocidades y reducido tamaño, permitiendo su implementación en diferentes aplicaciones, otro punto importante de los motores BLDC es el costo del mantenimiento, el cual se caracteriza por ser reducido, como punto desfavorable se tiene la tracción eléctrica que genera en los robots móviles en comparación con los motores DC con escobilla. Adicionalmente se debe implementar un driver de motor en la etapa de potencia, seguida de la etapa de control, incrementando el costo en la adquisición de los equipos. Aprovechando las características que presentan los motores BLDC, en el siguiente trabajo de investigación, se pretende evitar la implementación de encoder aplicados a la medición de la

velocidad de giro del eje del rotor, en lo que respecta a los motores de media potencia aplicados a la robótica móvil, presenta un favorable ahorro en cuanto al costo de la adquisición de los equipos. En el siguiente proyecto se realizó el diseño y construcción de dos robots móviles BellBot y WatchBot, para ser utilizados en varias aplicaciones.

Según (Lusson Cervantes, Beltran Aguedo, & Bausa Ortiz, 2010) en su trabajo titulado “Control Predictivo Lineal en un Reactor Tanque Agitado Continuo”. El siguiente trabajo habla sobre el diseño y la simulación del Control predictivo basado en modelo aplicado a un Reactor tanque agitado continuo, en el cual la variable a controlar es el caudal de refrigerante. El diseño y simulación del controlador predictivo se realizan con la utilización de las herramientas del programa Matlab lo cual facilita la comprensión y posterior aplicación del MPC en otros casos de estudio. Los resultados de la simulación muestran la potencialidad del control predictivo en el manejo de restricciones, desde la fase de diseño del sistema de control, al lograr un desempeño apropiado de este sistema lineal. La aplicación del MPC lineal al proceso bajo estudio, a través del Matlab, permitió evaluar la capacidad de estos controladores para seguir cambios en la referencia, y rechazar disturbios no medidos y medidos. La prestación del MPC incluye una alta capacidad para tratar restricciones y dinámicas lentas, tal y como se puede apreciar en las simulaciones desarrolladas.

Según (Lopera & Mejia, 2010) en su trabajo titulado “Implementación de un Control Predictivo Basado en Modelo Aplicado a un Sistema de Control de Caudal de Agua Didáctico”. El trabajo consiste en implementar un control predictivo basado en modelo aplicado a un sistema de control de caudal de agua en tipo didáctico, utilizando equipos de control e instrumentación comercial, así como las pruebas necesarias para obtener el modelo matemático de la planta, finalmente los resultados obtenidos evidencian las posibilidades de aplicar esta técnica de control en la regulación de variables de sistemas didácticos generales. Se demostró experimentalmente que con

equipos de control e instrumentación industrial convencional es posible lograr una implementación efectiva de un CPM y obtener de una forma sistemática un mejor desempeño en la regulación de variables de proceso como lo puede ser el caudal.

Según (Flor Unda, 2013) en su trabajo final de maestría titulado “Control Predictivo Basado en Modelo del Campo de Colectores Solares Tipo ACUREX”. El siguiente trabajo de titulación se basa en el diseño de estrategias para el control del fluido de aceite aplicando técnicas de control MPC implementada en el campo de colectores solares tipo ACUREX. Mediante el modelado matemático se analiza el comportamiento dinámico del sistema, self-tuning y controlador predictivo basado en modelo sometido a simulación. Como herramienta principal se implementó la plataforma de simulación Simulink de Matlab, el cual incorpora un diseño de planta, tomando como referencia los datos previamente adquiridos de una planta real. Como punto final, se comparan los resultados del comportamiento del controlador MPC con relación a un control PI, para posteriormente ser aplicadas en futuras implementaciones. Como conclusión se obtiene que la aplicación del controlador MPC no cumple las expectativas en este caso en particular, sugiriendo próximas implementaciones de esta técnica de control moderna, en conjunto con otras posibles soluciones. No obstante los resultados obtenidos en este trabajo de investigación serán de utilidad en futuras aportaciones.

Según (González & Zambrano, 2013) en su paper titulado “Implementación de un Algoritmo de Control Predictivo en Espacio de Estados sobre una Plataforma de Simulación Desarrollada en Matlab”. En este trabajo se determina un modelo matemático y los procedimientos para realizar el control predictivo en el programa de Matlab, también se hace la muestra de los resultados de una planta SISO y una planta MIMO. Se implementan diferentes variables para cada proceso, tomando en consideración la mejor alternativa para este proceso. Con el trabajo realizado se puede comprobar la fácil

adaptación del algoritmo de caso SISO al caso multivariables, únicamente se debe prestar atención a las dimensiones de las matrices y al orden en el que se formulan las restricciones. La aplicación del algoritmo de control predictivo dejó ver que su estructura se basa en operaciones matriciales las cuales demanda un coste computacional elevado. Para la implementación del caso multivariables con restricción, el tiempo de cálculo de la acción de control alcanzó hasta los 0.4 segundos, lo cual debe ser analizado antes de decidir utilizar este controlador en procesos con dinámica rápida.

Según (Javier, 2011) en su proyecto fin de carrera titulado “Diseño y Construcción de un Conversor DC/DC Reductor Elevador con control Predictivo Controlado por una Tarjeta FPGA”. En el siguiente trabajo de investigación, para el aprendizaje de la técnica de control MPC, se implementó un software que desarrolla una interfaz gráfica. Primero se diseña un conversor DC-DC reductor elevador, en el cual se obtiene el modelo matemático de la planta cuya implementación del control predictivo se lleva a cabo tanto en estado estable como en un estado transitorio, la comunicación entre el controlador MPC y el Conversor DC-DC reductor elevador, se realiza a través de una interfaz generada por una tarjeta de control que contiene restricciones para el sistema. Los valores de sintonización del controlador MPC, se obtienen a través de pruebas de ensayo y error, ya que a manera de práctica se pueden variar las ponderaciones del controlador, las restricciones, los horizontes de control y predicción, y finalmente se observan el comportamiento en cuanto a las variaciones en la salida del conversor DC-DC reductor elevador, a través de este método se busca obtener la mejor acción de control aplicada al sistema propuesto.

Según (Alejandro, 2016) en su trabajo de tesis titulado “Control Predictivo Basado en Escenarios para Sistemas Lineales con Saltos Markovianos”. En el siguiente trabajo de investigación se encuentran dos escenarios diferentes, conocidos como escenarios no arriesgados y escenarios arriesgados. Cuando se presentan casos de cambios abruptos del modo estudiado, se permite

calcular la probabilidad de éxito que tenga una ley de control tomando en cuenta las condiciones iniciales. Así se podrán diseñar leyes de control que garanticen un menor tiempo de fallo o al menos garanticen una cota mínima de confiabilidad. Una de las desventajas es que a grandes horizontes la carga computacional aumenta, debido a las grandes explosiones de combinaciones numéricas. Por tal motivo la creación de escenarios con gran posibilidad de obtener horizontes extendidos, es fundamental en la aplicación del control MPC para sistemas con saltos Markovianos.

Según (Gonzales Lecaros, 2014) en su trabajo de tesis titulado “Desarrollo de un Sistema de Control Predictivo Multivariables de un Generador de Vapor de Tubos de Agua”. En el desarrollo del siguiente trabajo de investigación, se analizó un modelo matemático no lineal y de tipo multivariables pertenecientes a un generador de vapor, dicho modelo obtenido de datos reales de una planta, fue implementado para simular el funcionamiento de la planta, en la etapa del diseño del controlador MPC se utilizó el modelo linealizado de la planta generadora de vapor, con el propósito de simplificar los cálculos. Para las pruebas realizadas con el controlador MPC, se toma en cuenta la señal de control estimando las respectivas restricciones, adicionalmente se colocaron señales de perturbación y ruido para visualizar el comportamiento del sistema en las diferentes pruebas realizadas. Todas estas pruebas permitieron determinar la robustez y el desempeño del controlador y haciendo comparación con un controlador PID, obteniendo como resultado que el controlador predictivo basado en modelo, genera menos errores y presenta un mejor tiempo de estabilidad en comparación con el control PID. Como resultado del trabajo de investigación se genera la propuesta de que el controlador MPC multivariables es viable para la implementación en el generador de vapor en este caso específico. La eficiencia del sistema permita el ahorro de combustible, aumentar la vida útil de los actuadores y trabajar en puntos de operación que generen mayor productividad de manera segura, con respecto a técnicas de control tradicional basado en control PID.

Según (Ferramosca, Gruber, Limon, & Camacho, 2013) en su paper titulado “Control Predictivo para Seguimiento de Sistemas no Lineales. Aplicación a un Plan Piloto”. En el presente trabajo de investigación las técnicas de control predictivos se caracterizan por presentar restricciones, además poseen leyes de control aplicadas tanto para sistemas lineales como no lineales, garantizando un punto de equilibrio. Pero si se llegase a presentar un inconveniente en el punto de equilibrio, estaría afectando la estabilidad del sistema, siendo incapaz de seguir la referencia deseada. El sistema propuesta se trata del control de temperatura de un laboratorio que imita una reacción química exotérmica. A través de las restricciones térmicas generadas se garantizó la estabilidad del sistema, siguiendo un punto de equilibrio. Como conclusión se obtiene que los resultados experimentales validan los resultados teóricos previamente obtenidos, puesto que el controlador es capaz de seguir un Setpoint establecido, garantizando la estabilidad del sistema.

Según (Oliden Semino, 2016) en su trabajo de tesis titulado “Desarrollo de un Controlador Predictivo Basado en Modelo para Plataforma Industrial”. En el siguiente trabajo de titulación se plantea el uso de algoritmos simplificados, resolver el problema de programación cuadrática. Usando como herramienta de identificación un controlador lógico programable. Se ha desarrollado una estructura de programación en la que se incluye los bloques de datos, las funciones y los bloques de organización, destinados a una futura implementación del controlador. Presenta la ventaja de incluir parámetros de sintonización intuitiva, que permiten al usuario sintonizar una velocidad de respuesta a lazo cerrado adecuada. El modelo matemático basado en el disco rotatorio de secado de harina ha sido exitosamente implementado en un controlador predictivo EPSAC, permitiendo mejorar la calidad de harina en cuanto a su valor de humedad, reducir los costos de consumo de vapor y evitando reprocesamiento del producto. Se ha realizado la implementación de sistemas para las pruebas de identificación múltiple entrada y múltiple salida del módulo de cuatro tanques acoplados, lo que permitió ajustar los

parámetros del modelo de principio físico e identificar un modelo en espacio de estados.

Según (López Caisaguano & Paredes Anchatipán, 2015) en su trabajo de tesis titulado “Diseño e Implementación de un Algoritmo de Control Predictivo para una Planta de Flujo Utilizando un Control de Automatización Programable para el Laboratorio de Redes Industriales y Control de Procesos de la Universidad de las Fuerzas Armadas ESPE” En el siguiente trabajo de investigación se implementó la plataforma de simulación LabVIEW para realizar la simulación, adicionalmente se adquirió un dispositivo Ni myRIO. El modelado matemático de la planta de flujo se obtuvo empleando ecuaciones en espacio de estados, una vez obtenido el controlador del proceso, este será implementado en la planta real. La respuesta obtenida del control predictivo basado en modelo aplicado a una planta de flujo en comparación a un controlador PID es superior, puesto que posee un menor tiempo de estabilización ante los cambios efectuados en los valores de referencia, también presenta un mínimo porcentaje de sobreimpulso, además posee una señal de control más estable y con mayor eficiencia. Se llegó a la conclusión de que la respuesta simulada de una planta con datos estimados aproximados a los reales, es similar a la respuesta real del sistema.

2.2. Control Predictivo

A finales de los años 70 se dio inicio del desarrollo de las técnicas de control predictivo, producto de la cual se generaron un conjunto de algoritmos que puntualmente se basaba en un modelo dinámico característico del proceso, mediante las acciones futuras de control se puede predecir el resultado que se tendrá en la salida, establecida a través de la disminución del error predicho sometido a las restricciones de operación las cuales eran determinadas minimizando el error predicho sujeto a restricciones de operación. Como resultado de los algoritmos se obtenían funciones con propiedades heurísticas y se trataba de aprovechar el desarrollo exponencial

de la capacidad de procesamiento de los computadores digitales que existían en aquel entonces (Ramírez Pérez, 2011).

En la actualidad se precisa que la implementación del controlador MPC posee más de 20.000 aplicaciones en la industria, que a través de la respectiva implementación muchos de los procesos se podrían beneficiarse con esta técnica. A continuación se detallan tres motivos por el cual el control MPC es un éxito en la industria (Cevallos Ulloa, 2010):

- Es fundamental tener un modelo real del proceso, ya que el modelo matemático permite al controlador tomar en cuenta todas las características relevantes de la dinámica del proceso.
- Para visualizar el efecto de las perturbaciones en realimentación y pre-alimentación, se debe tomar en cuenta el comportamiento a través de un horizonte futuro del proceso, para que el controlador pueda forzar la salida hacia la referencia deseada.
- Las restricciones que forman parte del control predictivo es la característica más importante del control MPC con respecto a otros métodos. Esto evita que se altere en lo posible el resultado, puesto que se considera en el diseño del controlador, dando como resultado un control más preciso con referencia a un punto óptimo de operación.

El Control Predictivo Basado en Modelo (CPM, MBPC o MPC) Abarca un campo muy extenso de técnicas de control, el cual está conformado de diferentes características como el control estocástico, control óptimo, control multivariables, control de procesos en tiempo muerto, y control con restricciones (Bordóns Alba, 2000).

El MPC abarca métodos de control que han sido desarrollados en base a ciertos planteamientos relacionados, por lo tanto no se habla de una estrategia específica, más bien es un gran campo de desarrollo. Estos procedimientos de diseño trasladan a las líneas de controladores que toman las mismas estructuras y muestran más grados de libertad. Las representaciones de los

valores de mayor a menor medida que se muestran en los controladores predictivos son básicamente las siguientes:

- Minimizar la función objetivo para calcular las señales de control.
- En cada instante de muestreo se debe considerar la señal de control inicial, descartando el resto, para así obtener la estrategia deslizante desplazada hacia el futuro.
- Modelo matemático que permite predecir la salida del proceso en un instante de tiempo futuro.

2.2.1. Ventajas del Control Predictivo

El control predictivo es un tipo de control de naturaleza abierta en el cual se han efectuado muchos desarrollos, por lo cual es muy considerado tanto en aplicaciones industriales como en el mundo académico. Hoy en día muchas aplicaciones industriales operan bajo el control predictivo con un grado de aceptación muy alto. Las ventajas más relevantes del MPC con respecto a otros controladores son:

- Es de gran ayuda si las referencias futuras son tratadas (robótica o batch processing).
- Es una metodología completamente abierta fundamentada en algunos principios básicos que permite futuras extensiones.
- Su intervención en los procesos con una dinámica característicamente simple, hasta procesos con un nivel de complejidad mayor, como por ejemplo un sistema inestable o con retardos considerables.
- Debido a que son muy intuitivos los conceptos de MPC, y a su vez que los parámetros de sintonización son fáciles de comprender, se vuelve de interés para las personas que deseen trabajar con esta técnica de control y que no posean vasta información sobre el MPC.
- En caso de retardos se genera automáticamente una compensación.
- Cuando se presentan procesos multivariables, estos casos son abordados fácilmente.

- Mientras se efectúa la etapa de diseño, fácilmente se pueden ingresar las restricciones sistemáticamente.

2.2.2. Estrategia de Control

La metodología de todos los controladores pertenecientes a la familia del control predictivo basado en modelo se caracterizan por la siguiente estrategia (Ponce Arias, 2014):

- Aplicando el modelo del proceso en cada instante t , se obtiene el horizonte de predicción, en un determinado horizonte N se predicen las salidas futuras. Las salidas anunciadas, $y(t+k|t)$ para $k=1 \dots N$ requiere hasta el instante t todos los valores conocidos (entradas y salidas pasadas) y de las futuras señales de control $u(t+k|t)$, $k=0 \dots N-1$ que se requieren calcular una vez enviadas al sistema, tal como se muestra en la figura 2.1.

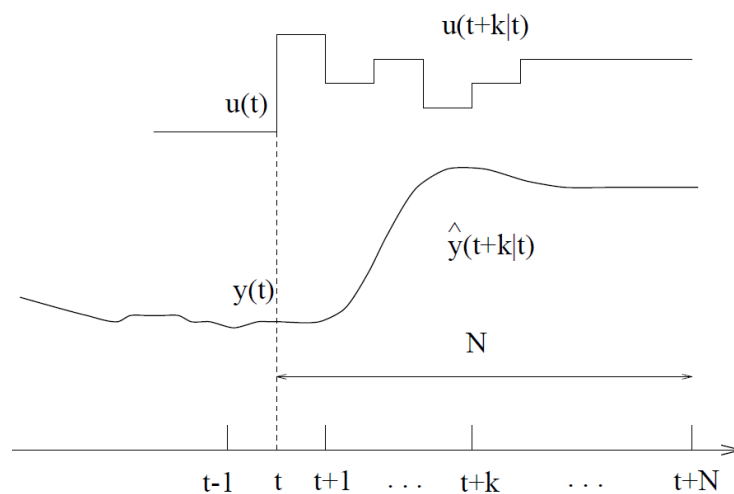


Figura 2.1: Estrategia de control predictivo.

Fuente: (Kanchwala & Bordons, 2015)

- Para calcular las futuras señales de control, se debe optimizar un criterio específico, en el cual se requiere que el proceso permanezca lo más cerca posible a la referencia $w(t+k)$ (Setpoint o aproximado). A través de los errores entre la salida y la trayectoria de referencia anunciada, este método podría tomar la forma de una función

cuadrática. Se puede conseguir una solución clara si el método es cuadrático, además el modelo debe ser lineal y las restricciones deben ser nulas, también se podría implementar un método de optimización iterativo. Con respecto a la ley de control futura se generan determinadas suposiciones.

- iii. La señal de control $u(t|t)$ es enviada al proceso y se descartan las siguientes señales de control, puesto que $y(t + 1)$ se conocerá al instante y se repetirá el proceso pero con diferentes valores, actualizando las siguientes secuencias. Se calcula por tanto $u(t + 1|t + 1)$, usando el criterio de horizonte deslizante.

Apoyado en las futuras señales de control, se realiza una demostración para determinar las salidas futuras del proceso. Basada en la función de coste, el optimizador muestra las señales calculadas (donde de muestra el error futuro de seguimiento) así como las restricciones mostradas en la figura 2.2. La capacidad del modelo establecido para tomar en cuenta la dinámica del proceso, permite predecir las futuras salidas. Por lo tanto se puede decir que el modelo a implementar en el controlador, juega un papel muy fundamental.

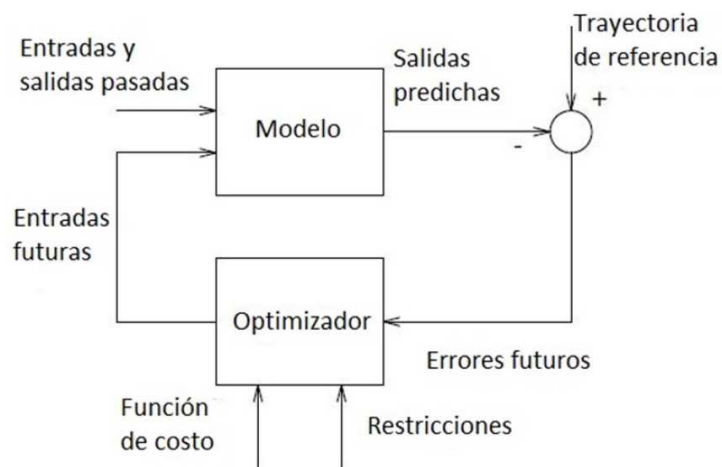


Figura 2.2: Estructura general del control predictivo basado en modelo.

Fuente: (Llanos, Pilatasig, Curay, & Vaca, 2016)

En lo que respecta a la estrategia de control predictivo, el optimizador es otro punto muy importante del modelo, ya que genera la acción de control.

Se puede lograr una función clara de las entradas y salidas pasadas además de la referencia, si la función de coste es cuadrática.

2.2.3. Elementos Básicos del Control Predictivo

Los controladores predictivos poseen elementos comunes, se pueden elegir diversas opciones para cada uno de estos elementos, dando lugar a diferentes algoritmos, los cuales se mencionan a continuación:

- Función objetivo
- Obtención de la ley de control
- Modelo de predicción

2.2.3.1. Modelo de Predicción

En el MPC, es el componente más crucial para realizar un control óptimo, el modelo matemático que describe el comportamiento de la planta a controlarse, la representación y las características del modelo pueden ser lineales, no lineales, en tiempo continuo, o en tiempo discreto, en variables de estado o en funciones de entrada y salida. El diseño del modelo debe capturar la máxima dinámica del sistema, para permitir el cálculo de las predicciones y el análisis teórico.

Este componente puede incluir los efectos de las perturbaciones no medibles y errores de modelo, así como un modelo de perturbación medibles, de tal forma que muestre de manera anticipada los acontecimientos a efectuar los correctivos necesarios (feedforward). Se aplicara el principio de la superposición, cuando el modelo de la perturbación medible y la perturbación no medible sean lineales, realizando una sumatoria de las salidas para formar una salida común. Tal como se muestra en la figura 2.3.

Modelo del Proceso: Las diferentes formas de modelar un proceso aparecen en las formulaciones del MPC siendo las más usadas:

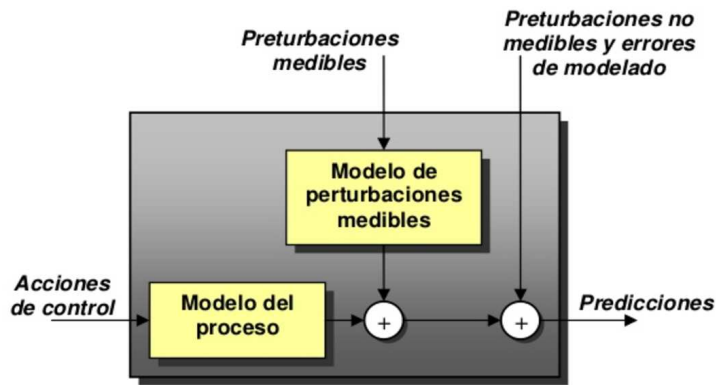


Figura 2.3: Modelo de predicción

Fuente: (González Querubín, 2011)

Modelo de Respuesta Impulsiva.- conocida también como secuencia de ponderación o modelo de convolución. La relación entrada-salida de este modelo está representada por la Ecuación 2.1.

$$y(t) = \sum_{i=1}^N h_i u(t - i) \quad \text{Ecuación 2.1}$$

Donde h_i son los valores muestreados, y se consideran N valores de muestra, de esta forma se permite representar procesos estables y sin integradores, un inconveniente es el alto valor de N cerca los 40 – 50 muestras. La predicción viene dada con la Ecuación 2.2. (Ver figura 2.4).

$$y(t) = \sum_{i=1}^N h_i u(t - i) = H(z^{-1}) u(t) \quad \text{Ecuación 2.2}$$

Donde $H(z^{-1}) = h_1 z^{-1} + h_2 z^{-2} + \dots + h_N z^{-N}$ y z^{-1} simboliza el retardo unitario. Finalmente la señal de la salida predicha en el instante t , está dada por la Ecuación 2.3.

$$\hat{y}(t + k|t) = \sum_{i=1}^N h_i u(t + k - i|t) = H(z^{-1}) u(t + k|t) \quad \text{Ecuación 2.3}$$

La especificación de la dinámica del proceso debido a su sencillez es una de las ventajas de este modelo pudiendo ser de fase no mínima o con retardos. En cuanto a desventajas de este modelo se puede concluir que no es factible

reemplazar procesos inestables y adquiere varios coeficientes, producto del muestreo propio del modelo.

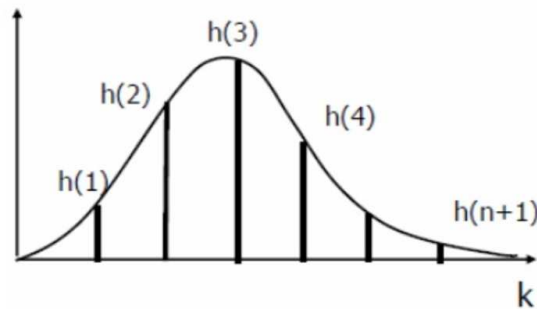


Figura 2.4: Respuesta al impulso

Fuente: (González Querubín, 2011)

Modelo de Respuesta Escalón.- con respecto a la respuesta del modelo de impulso, se puede decir que posee las mismas ventajas y desventajas, tal como se observa la respuesta en la figura 2.5 y a través de la Ecuación 2.4, se representada la salida.

$$y(t) = y_0 + \sum_{i=1}^N gi \Delta u(t - i) = y_0 + G(z^{-1})(1 - z^{-1}) u(t) \quad \text{Ecuación 2.4}$$

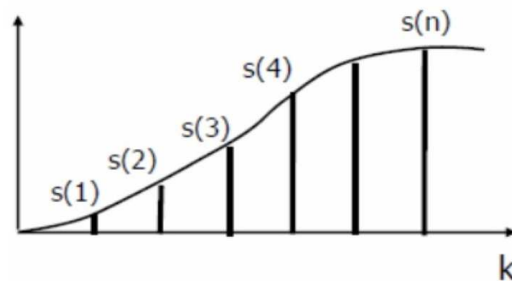


Figura 2.5: Respuesta al escalón

Fuente: (González Querubín, 2011)

Donde gi son los valores muestreados, este método presenta las mismas ventajas y desventajas que el modelo de la respuesta impulsiva, el valor de y_0 , puede tomarse como 0, por lo cual la salida predicha está definida por la Ecuación 2.5.

$$\hat{y}(t + k|t) = \sum_{i=1}^N gi \Delta u(t + k - i|t) \quad \text{Ecuación 2.5}$$

Modelo de Función de Transferencia.- para adquirir la salida mostrada en la Ecuación 2.6, es importante utilizar el modelo del proceso en tiempo discreto.

$$A(z^{-1}) y(t) = B(z^{-1}) u(t) \quad \text{Ecuación 2.6}$$

A continuación se describe el numerador B y el denominador A con sus respectivos coeficientes:

$$A(z^{-1}) = 1 + a_1z^{-1} + a_2z^{-2} + \dots + a_{na}z^{-na} \quad \text{Ecuación 2.7}$$

$$B(z^{-1}) = b_1z^{-1} + b_2z^{-2} + \dots + b_{nb}z^{-nb} \quad \text{Ecuación 2.8}$$

Para poder calcular la predicción de salida, se debe implementar la Ecuación 2.9.

$$y(t + j|t) = \frac{B(z^{-1})}{A(z^{-1})} u(t + j|t) \quad \text{Ecuación 2.9}$$

Se debe contar con un modelo específico, ya que en su interior representa exactamente las características de la planta, por lo tanto se necesita de un buen proceso en cuanto a su orden para adquirir los polinomios A y B. Es fundamental saber el orden de los polinomios puesto que presenta una ventaja para representar procesos inestables y un reducido número de parámetros.

Modelo de Espacio de Estados.- está representado por las siguientes ecuaciones:

$$X(t) = Ax(t-1) + Bu(t-1) \quad \text{Ecuación 2.10}$$

$$Y(t) = Cx(t) \quad \text{Ecuación 2.11}$$

Como se puede observar X es la variable que simboliza el estado mientras que A, B y C conforman las matrices de estado del proceso.

En la Ecuación 2.12, se representa la ecuación de salida.

$$\hat{y}(t+k|t) = C\hat{x}(t+k|t) = C[A^k x(t) + \sum_{i=1}^N A^{i-1} Bu(t+k-i|t)]$$

Ecuación 2.12

La ventaja que presenta la Ecuación 2.12 es para sistemas multivariados y a través de ellas se puede analizar la estructura interna del proceso. Su principal desventaja es que los cálculos pueden ser complicados, e incluir un observador de estado para sistemas estocásticos.

Modelo de Perturbación: Este punto es importante ya que conforma el modelo de predicción principal, como su nombre lo indica, posibilita el modelado de perturbaciones ambientales, así como problemas de modelado, entre otras. El autorregresivo integrado de media móvil (ARIMA) es un modelo extendido, donde la diferencia entre la salida medida y la predicha se da obtiene a través de la Ecuación 2.13.

$$n(t) = \frac{C(z^{-1}) e(t)}{D(z^{-1})}$$

Ecuación 2.13

Donde $D(z^{-1})$ incluye únicamente un integrador $\Delta = 1 - z^{-1}$, y el modelo está en función de $e(t)$ que es ruido de un valor promedio equivalente a cero y normalmente el polinomio del numerador C es igual a uno, este modelo considera dos tipos de perturbaciones:

- Perturbaciones aleatorias en instantes aleatorios.
- Movimiento browniano.

Al incluir un integrador se consigue un error nulo en estado permanente, cuya mejor predicción será:

$$\hat{n}(t+k|t) = n(t)$$

Ecuación 2.14

Una característica principal del MPC es el empleo de los conceptos de respuesta libre y forzada y está definida como la suma de dos señales y una secuencia de acción:

$$u(t) = u_f(t) + u_c(t) \quad \text{Ecuación 2.15}$$

Las señales de entradas pasadas $u_f(t)$ se obtienen antes del instante t y su último valor es igual en el futuro.

$$u_f(t - j) = u(t - j) \quad \text{Para } j = 1, 2, \dots \quad \text{Ecuación 2.16}$$

$$u_f(t + j) = u(t - j) \quad \text{Para } j = 0, 1, 2, \dots \quad \text{Ecuación 2.17}$$

El valor 0 en el pasado, está representado por $u_c(t)$ y corresponde a las señales de control futura.

$$u_c(t - j) = 0 \quad \text{Para } j = 1, 2, \dots \quad \text{Ecuación 2.18}$$

$$u_c(t + j) = u(t + j) - u(t - 1) \quad \text{Para } j = 1, 2, \dots \quad \text{Ecuación 2.19}$$

De esta forma se define los siguientes modelos estocásticos dependiendo de la modelación del proceso y las perturbaciones utilizadas.

Modelo ARX

$$A(z^{-1})y(t) = B(z^{-1})u(t) + e(t) \quad \text{Ecuación 2.20}$$

Modelo ARMAX

$$A(z^{-1})y(t) = B(z^{-1})u(t) + C(z^{-1})e(t) \quad \text{Ecuación 2.21}$$

Modelo ARIX

$$A(z^{-1})y(t) = B(z^{-1})u(t) + \frac{e(t)}{\Delta} \quad \text{Ecuación 2.22}$$

Modelo ARIMAX

$$A(z^{-1})y(t) = B(z^{-1})u(t) + \frac{c(z^{-1})e(t)}{\Delta} \quad \text{Ecuación 2.23}$$

2.2.3.2. Función Objetivo

Conocida como función de costo, es aquella que indica el criterio a optimizar. Es una función positiva que expresa el costo asociado a la evolución del proceso en el intervalo del horizonte de predicción N , donde posee la forma:

$$J(N_1, N_2, N_u) = \sum_{j=1}^{N_2} \delta(j) [\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2 \quad \text{Ecuación 2.24}$$

Donde los parámetros N_1, N_2 son los horizontes mínimos y máximos de predicción respectivamente y N_u es el horizonte de control, de esta forma se marcan los límites donde la salida sigue a la referencia. Los coeficientes $\delta(j)$ y $\lambda(j)$ son secuencias que ponderan el comportamiento futuro, donde el valor de ponderación se define por:

$$\delta(j) = \alpha^{N_2-j} \quad \text{Ecuación 2.25}$$

Donde:

- $0 < \alpha < 1$ penaliza en mayor proporción los errores más alejados del instante t , y se obtiene un control suave y de menor esfuerzo.
- $\alpha > 1$ penaliza en mayor proporción los primeros errores, y la respuesta es un control más brusco.

Todos estos valores pueden ser parámetros de sintonización y ofrece una gama de control desde un control estándar hasta un control específico.

Trayectoria de Referencia: El sistema es capaz de reaccionar para evitar los efectos producidos por el retardo del proceso, esto es posible si se sabe la evolución de la referencia en el futuro, es muy utilizada en aplicaciones de robótica, como el control de servos.

Es muy importante cuando la referencia es constante ya que se podría conseguir una leve mejora, y conociendo el instante donde el valor de la respuesta cambia y adelanta la acción de control a esta circunstancia.

$w(t+k)$ es la trayectoria de referencia en la cual se basan la mayoría de los métodos y no es imprescindible que el valor real corresponda, sino el valor actual de la salida $y(t)$ y a través de un sistema de primer orden se debe conseguir un valor cercano a la referencia.

$$w(t) = y(t)$$

$$w(t+k) = \alpha w(t+k-1) + (1-\alpha) r(t+k) \quad k = 1 \dots N \quad \text{Ecuación 2.26}$$

De esta forma se puede establecer $0 < \alpha < 1$, si el valor de α se encuentra más cerca de 1, y más leve será la aproximación siendo este un valor fácil de modificarlo para que influya en la respuesta dinámica del sistema.

Mientras que si el valor de α es más pequeño, se tiene un seguimiento rápido como se puede observar $w_1 = (t+k)$ en la figura , mientras que si aumenta la trayectoria de referencia será la respuesta $w_2 = (t+k)$ y se puede obtener una respuesta más suave como se demuestra en la figura 2.6.

Restricciones: Las restricciones están inmersas en todo tipo de procesos, puesto que los actuadores presentan una determinada velocidad de respuesta, así como un campo limitado de acción, las razones por las cual se

imponen restricciones van desde la seguridad o medioambiente, hasta por la construcción de los sensores.

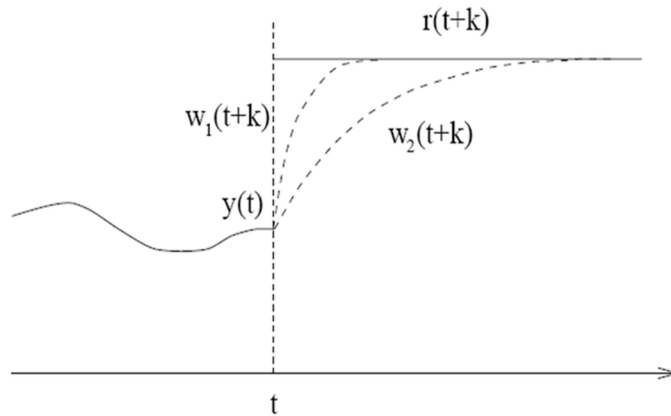


Figura 2.6: Trayectoria de referencia.

Fuente: (Moromenacho, Rosero, & Bravo, 2009)

El éxito de la industria, en la implementación del MPC radica en la consideración de las restricciones, ya que se considera los límites en la amplitud, en la velocidad de cambio de la señal de control y los límites en la salida:

$$u_{min} \leq u(t) \leq u_{max} \forall t$$

$$du_{min} \leq u(t) - u(t - 1) \leq u_{max} \forall t$$

$$y_{min} \leq y(t) \leq y_{max} \forall t$$

Ecuación 2.27

2.2.3.3. Obtención de la Ley de Control

Se debe reducir la función de coste J como se observa en la Ecuación 2.24, para obtener los valores de la señal de control, como primer punto debe obtener las salidas predichas en función de los valores actuales y pasados, tanto de la entrada como de la salida, de igual forma las señales futuras de coste, obteniendo una extensión que al minimizar da como resultado los valores buscados.

N_u es el horizonte de control, pilar en el cual se fundamenta la estructura de la ley de control, que luego de un intervalo de tiempo considerable $N_u < N_2$, donde la señal de control no presenta cambio alguno, es decir:

$$\Delta u(t + j - 1) = 0 \quad j > N_u \quad \text{Ecuación 2.28}$$

Se refiere a que en ciertos instantes, los cambios en el control equivalen a dar pesos infinitos. Para que todas las acciones sean igual a $u(t)$ se considerará N_u equivalente a 1 (Ortiz Guevara, 2011).

2.3. Plataforma de Simulación

El origen de la plataforma de simulación Matlab se remonta a 1984, donde el matemático Clever Moler crea la primera versión, la cual surge con la idea principal de emplear paquetes de subrutinas escritos en lenguaje de programación Fortran para ser implementados en los cursos de álgebra lineal y análisis numérico. Matlab como lenguaje de programación fue creado en 1970 con el objetivo de facilitar el acceso al software de matrices Linpack y Eispack sin la necesidad de hacer el uso de Fortran.

2.3.1. Descripción de la Plataforma

La plataforma de simulación Matlab es reconocida porque es una herramienta de cómputo la cual trabaja de la mano con cálculos, simulaciones y visualizaciones. Entre sus prestaciones más representativas se encuentra el análisis numérico, cálculo matricial, programación, procesamiento de señal, simulación, representación gráfica, interacción con otras plataformas, entre otras. Por lo que lo hace un programa interactivo para el usuario.

El nombre Matlab es la combinación de las iniciales de “Matrix Laboratory” que traducido al español se denomina “Laboratorio Matricial”, el cual es una herramienta muy poderosa, la cual está basada en una matriz adimensional, se puede decir que Matlab es un lenguaje de computación técnico de alto nivel. La implementación de un lenguaje de alto nivel en Matlab es la ventaja

que permite a este software, resolver los problemas efectuados de manera más rápida en comparación con los lenguajes de programación clásicos como el C++, Pascal entre otros. La plataforma Matlab nace con el fin de crear una herramienta fiable y poderosa, que a través de la gran capacidad de procesamiento de datos aprovechada de los ordenadores, permite resolver problemas de cálculos complejos.

Matlab es un lenguaje de computación técnico de alto nivel que permite manipular vectores y matrices como simples variables y posee un entorno interactivo para el desarrollo de cálculos numéricos, algoritmos, visualización de datos, análisis de datos, etc. En cuanto a las necesidades que tenga el usuario ya sea (un algoritmo, un análisis, una estadística, una gráfica, una simulación un informe, etc.), Matlab está diseñado para realizar todos estos trabajos (Andrade Moreira & Cortez Bohórquez, 2012).

2.3.2. Funcionalidades

Una de las grandes ventajas del entorno Matlab es que posee una serie de soluciones específicas llamadas ToolBoxes, las cuales son herramientas extendidas del entorno Matlab para la solución de problemas particulares como: Diseño de sistemas de Control, simulación de sistemas dinámicos, identificación de sistemas, procesamiento de señales, entre otras.

Matlab agrupa más de 35 funcionalidades en cajas de herramientas (ToolBoxes) y paquetes de bloques para (Simulink). Entre las herramientas más representativas que dispone el entorno Matlab se mencionan algunas de las funcionalidades a continuación:

- Herramientas para adquisición de datos.
- Herramientas para la exploración análisis de datos.
- Prototipaje y desarrollo de algoritmos.
- Programación y desarrollo de algoritmos.
- Modelado y simulación.
- Visualización y procesamiento de imagen.

Para un correcto manejo de la plataforma Matlab se deben tomar en cuenta tres aspectos esenciales, los cuales se describen a continuación:

- Es fundamental conocer cómo se ingresan las variables al programa y cómo pueden ser manipuladas.
- Comprender el manejo de cada una de las funciones en general.
- Reconocer de dónde adquirir la información correspondiente a cada función específica.

2.3.3. Requisitos del sistema operativo Windows 10 para instalar Matlab

Como requisitos mínimos para poder instalar y ejecutar la plataforma de Matlab 2017 sin ningún inconveniente sobre el sistema operativo Windows 2010 debemos cumplir los siguientes requisitos (MathWorks, 2017):

- Cualquier procesador Intel o AMD x86-64.
- El espacio del disco duro debe ser de 2 GB para Matlab y 4-6 GB para una instalación típica.
- Se requiere un espacio mínimo en la memoria RAM de 4 GB para ejecutar Matlab con Simulink.
- No se requiere ninguna tarjeta gráfica específica.
- Puerto USB o CD-ROM para drivers de Matlab.

2.3.4. Descripción del Entorno Virtual

Lo primero que se observa una vez que se ha cargado la plataforma de simulación Matlab es la ventana Command Windows como se observa en la figura 2.7, la cual se considera la ventana más importante del entorno Matlab la misma que permite la interacción entre el usuario y el núcleo del procesamiento, en ella se muestran los resultados correspondientes a cada acción realizada. En el margen superior izquierdo de esta ventana aparece un prompt `>>` el mismo que precisa la línea correspondiente para el ingreso de las instrucciones, y a través de la tecla Enter dicha línea se mandará a ejecutar. En general se podría decir que todos los resultados obtenidos se

podrán visualizar en el mismo Command Windows con la excepción de los comandos gráficos.

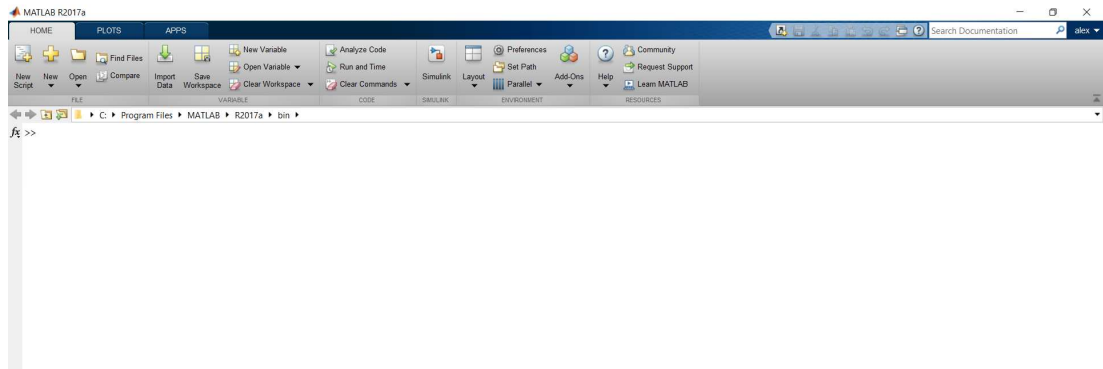


Figura 2.7: Ventana de inicio del entorno virtual Matlab.

Fuente: Autor

Para ingresar a la ventana del Workspace como se muestra en la figura 2.8, se debe dar clic en la opción “New” de la barra de tareas, y luego seleccionar “Script”. El entorno Workspace está conformado por el conjunto de funciones y variables de usuario que en un tiempo dado están definidas en la memoria del programa. Se puede acceder a la información del Workspace a través de las líneas de comando digitando “who” o “whos” en el Command windows.

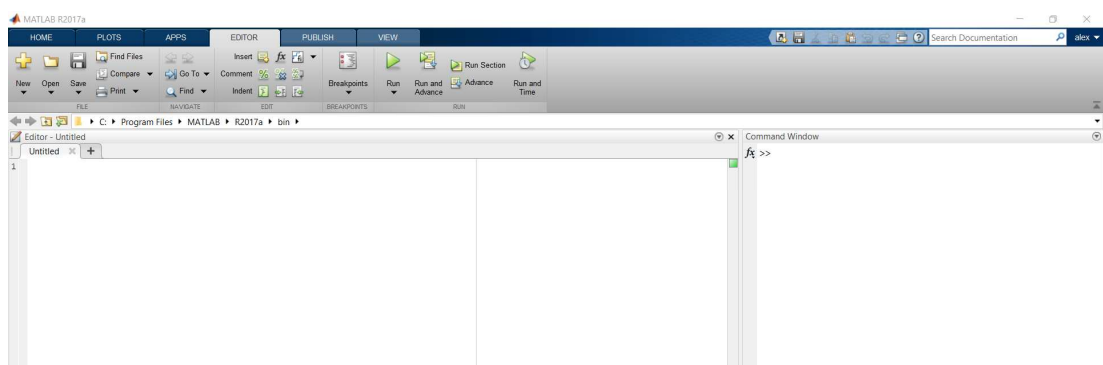


Figura 2.8: Ventana del Workspace.

Fuente: Autor

La ventana de Command History como se aprecia en la figura 2.9, se obtiene ubicando el cursor en la parte superior derecha de la pantalla, posteriormente dando clic en el botón y automáticamente se despliega una lista de funciones, luego seleccionamos la opción Command History. Esta ventana permite el acceso a las líneas de comando que se han ejecutado anteriormente en el Command windows. Los comando ejecutados con anterioridad también pueden volver a ejecutarse ya sea dando doble clic sobre la línea de comando en el Command History o a través del menú contextual dando clic derecho.

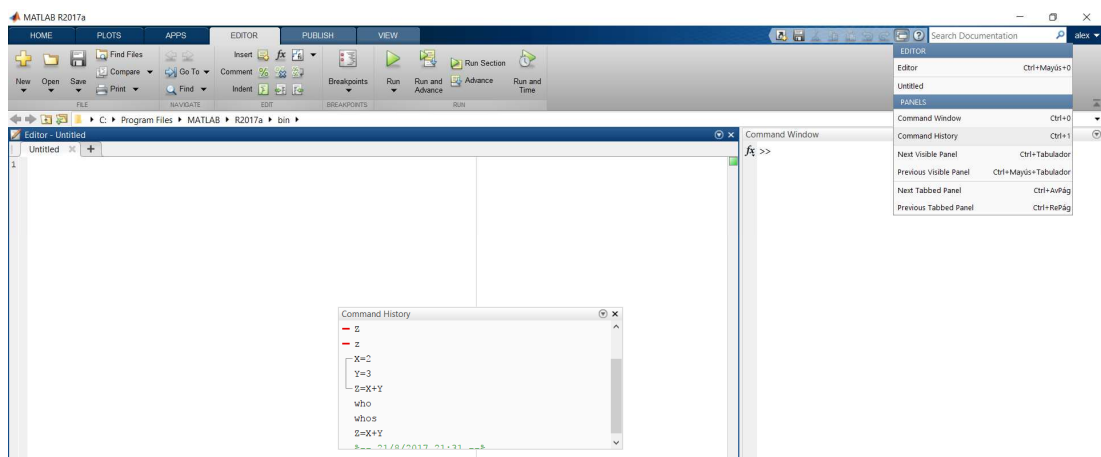


Figura 2.9: Ventana Command History.

Fuente: Autor

2.3.5. Librerías del Entorno Matlab

La plataforma virtual Matlab incorpora un vasto número de funciones, una de ellas son las funciones intrínsecas, esto quiere decir que estas funciones se encuentran incluidas en el propio código ejecutable del programa, como características principales que poseen estas funciones es que son más rápidas y eficientes. A continuación se clasifican las funciones de mayor importancia según su finalidad:

- Funciones para procesamiento de señal.
- Funciones matemáticas elementales y especializadas.
- Integración numérica.
- Resolución de ecuaciones no-lineales y optimización.
- Funciones especiales.

- Matrices elementales y manipulación de vectores.
- Funciones para la descomposición y/o factorización de matrices.
- Funciones para integración de ecuaciones diferenciales ordinarias.
- Polinomios e interpolación.
- Estadística y análisis de datos.
- Matriciales especiales.

2.3.6. Tipos de Datos

La plataforma Matlab está diseñada para trabajar correctamente con todos los tipos de datos que se describen a continuación:

- Vectores o matrices de celdas (cell arrays), es un tipo de dato con contenedores de datos indexados denominados celdas, donde cada celda puede incluir cualquier tipo de datos.
- Cadena de caracteres o String, es un conjunto de dato de tipo Char el cual finaliza con un carácter de tipo nulo y es elemental en cualquier lenguaje de programación.
- Estructuras o agrupaciones bajo un mismo nombre de datos de diferente naturaleza.
- Matrices dispersas, son matrices de grandes dimensiones y con un gran porcentaje de valores en cero.
- Hipermatrices, son matrices que manejan tres o más dimensiones.

2.3.7. Simulink

El entorno gráfico Simulink, es un programa de simulación que se caracteriza por tener una interfaz gráfica que forma parte del programa principal Matlab. Esta herramienta virtual, nos permite generar diagramas de bloques con las funciones de transferencias para simular los sistemas, también permite visualizar el comportamiento, evolución y resultados de una planta de manera gráfica, es de gran ayuda en la simulación de sistemas dinámicos.

2.3.8. Entorno Simulink

El acceso al entorno grafico Simulink es fácil e intuitivo, ya que podemos ingresar a través del Command windows, ingresando la palabra “Simulink” o también se puede ingresar haciendo clic en el icono ubicado en la barra de herramientas de Matlab como se muestra en la figura 2.10.

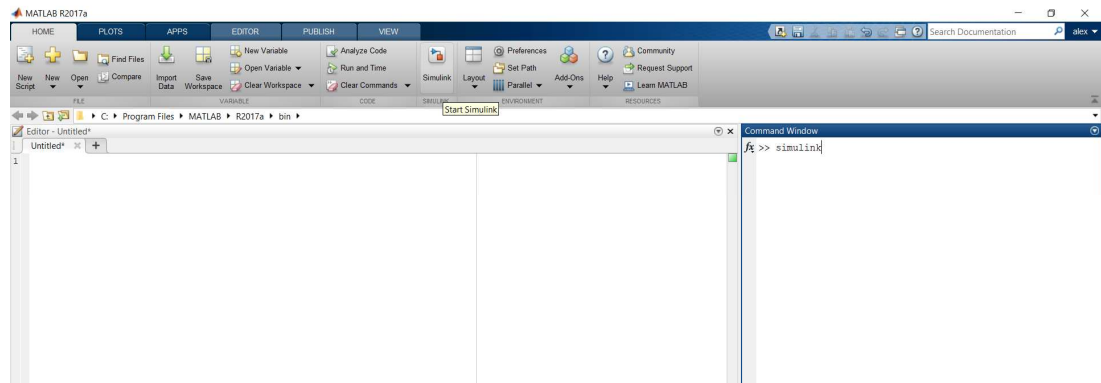


Figura 2.10: Acceso entorno Simulink.

Fuente: Autor

Cuando que se inicia el entorno grafico Simulink se observa una ventana como la que se muestra en la figura 2.11, la cual es el área de trabajo, en donde se va a efectuar la programación grafica de cualquier sistemas que se desee simular.

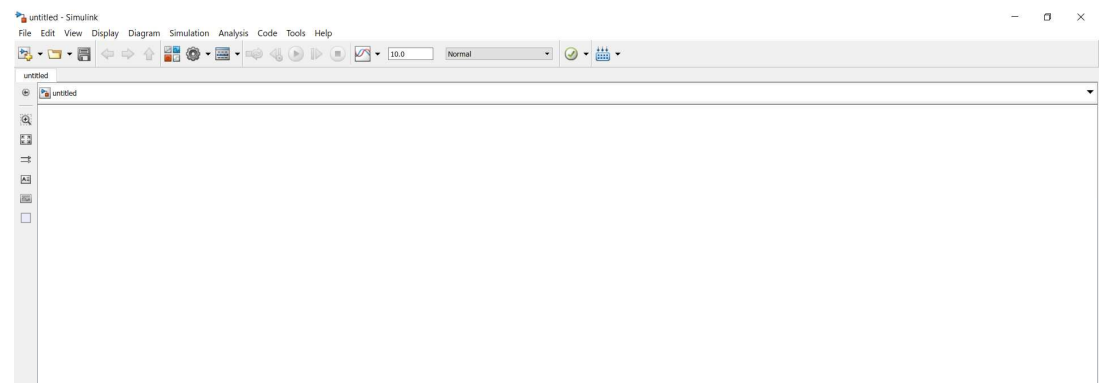


Figura 2.11: Ventana de Simulink.

Fuente: Autor

Para empezar a programar, se debe llamar a las librería del entorno de programación grafica Simulink, la cual se puede acceder haciendo clic en el

icono de la barra de herramientas de Simulink o también se puede ingresar haciendo clic en el botón “View” ubicado en la barra de herramientas de Simulink y luego seleccionar “Library Browser” como se muestra en la figura 2.12.

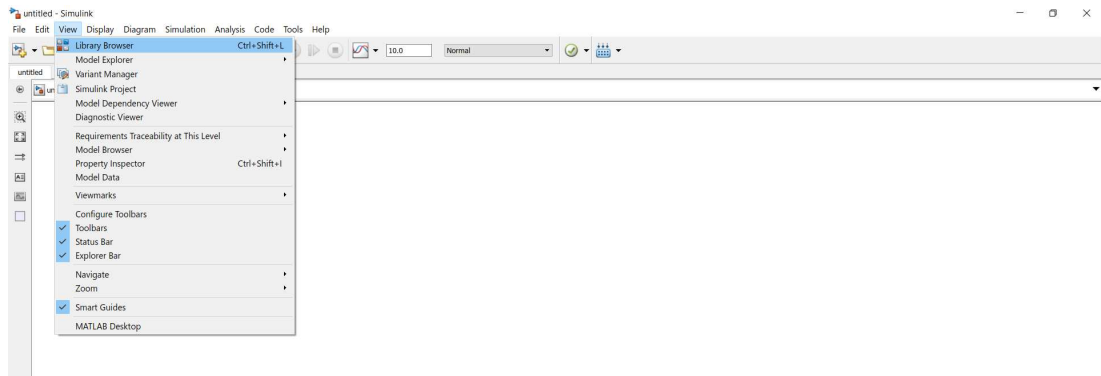


Figura 2.12: Acceso librería Simulink.

Fuente: Autor

2.3.9. Librerías Simulink

La lista de las librerías que se encuentran en el entorno gráfico de programación Simulink como se observa en la ventana al lado izquierdo de la figura 2.13, entre las más significativas se encuentran:

- Commonly used blocks.
- Continuous
- Discrete
- Math Operations
- Signal Routing
- Sinks
- Sources
- User-defined functions

Al seleccionar mediante un clic una de las opciones que nos muestra la librería de Simulink, en la pantalla ubicada a la derecha nos muestra más opciones en forma de bloques relacionadas al tema principal.

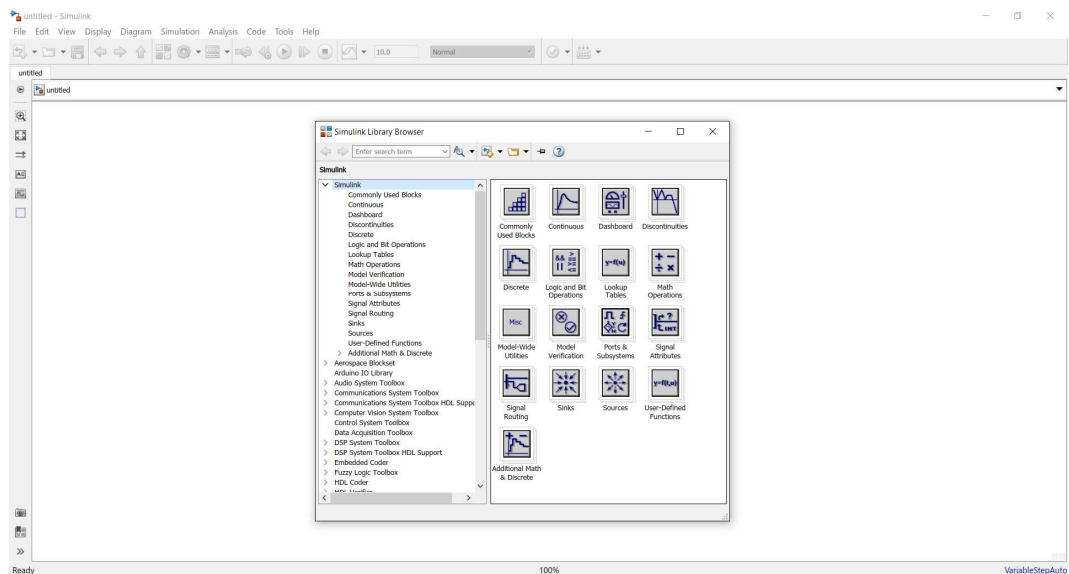


Figura 2.13: Ventana de librería Simulink.

Fuente: Autor

2.4. Sistemas Supervisorios

A inicios de los años setenta la tendencia de la automatización, era que cada fabricante se encargaba de desarrollar sus propios equipos electrónicos específicos para solventar problemas de control que se generaban. Por lo general eran elementos con una reducida capacidad de memoria, poseían entradas y salidas fijas y trabajaban con lenguajes de programación poco conocidos. Una nueva generación de autómatas surge a mediados de los años setenta, a cargo de fabricantes de equipos electrónicos como Siemens, Square-D, o Allen-Bradley, los cuales evolucionaron a los autómatas permitiendo incorporar una mayor cantidad de entradas y salidas, debido a que estos equipos estaban diseñados para tolerar las más austeras condiciones, presentaban características como: su gran tamaño, el peso y su costo de fabricación, un ejemplo de los primeros controladores representa en la figura 2.14. Con la evolución de la electrónica, los componentes electrónicos que formaban parte de los controladores, fueron reduciendo su tamaño y por consiguiente su consumo energético, lo que hizo posible disminuir el tamaño, peso y costo de los primeros controladores (Rodríguez Penin, 2012).

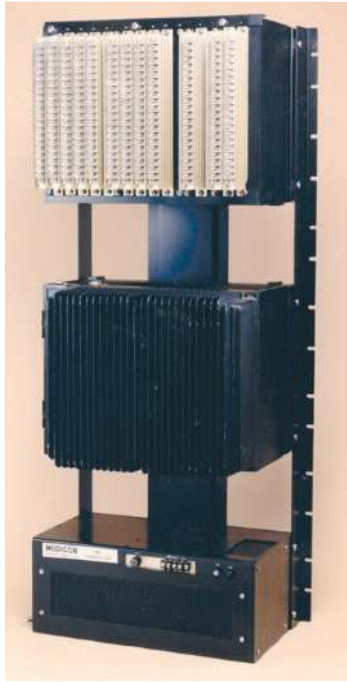


Figura 2.14: Primer PLC – Modicon 084.

Fuente: (Open automation, 2008)

2.4.1. Control Supervisorio

Es un sistema que reúne la información de los diferentes parámetros provenientes de la planta, y son procesados para que en base a un criterio se realice una acción de control. Se puede determinar como el monitoreo y control de procesos. La acción de control se puede clasificar de la siguiente manera:

- Manual: Un operador analiza y ejecuta por sí mismo la acción de control a tomar.
- Semiautomático: es la combinación de ambas acciones, en la cual ciertos procesos son automatizados completamente, mientras que otras tareas son encomendadas a un operario.
- Automático: Las acciones de control son tomadas y ejecutadas a través de un controlador.

2.4.2. Concepto de Control

Cualquier sistema controlado puede serlo de dos maneras, ya sea un sistema de regulación en lazo abierto o un sistema de regulación en lazo cerrado.

Los sistemas de regulación en lazo abierto se basan en el diseño adecuado de los parámetros y las condiciones de trabajo del elemento controlador para que este sea capaz de mantener el sistema controlado dentro de los límites deseados representado en la figura 2.15. En cambio los sistemas de regulación en lazo cerrado o realimentados, trabajan vigilando continuamente las reacciones del sistema a controlar, efectuando las acciones de corrección necesarias para mantener el control dentro de los límites deseados representado en la figura 2.16.

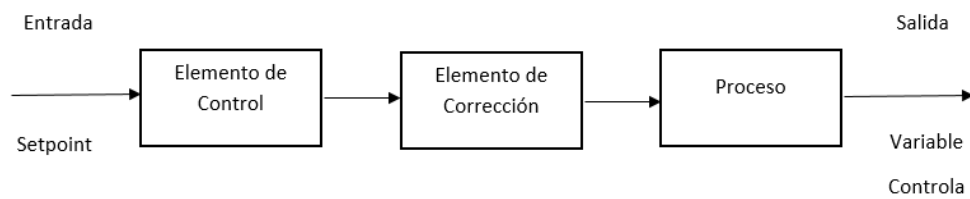


Figura 2.15: Representación en lazo abierto de un sistema.

Fuente: Autor

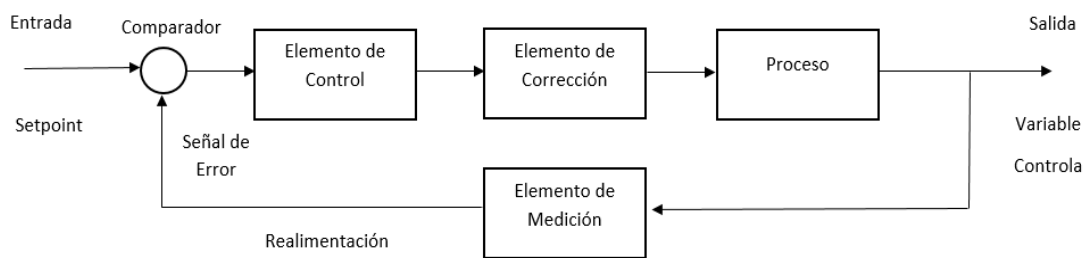


Figura 2.16: Representación en lazo cerrado de un sistema.

Fuente: Autor

En sistemas de control la información sufre tres cambios (Rodríguez Penin, 2012):

1. La información es generada por las variables a controlar del sistema, captada e interpretada a través de los sensores o transductores.
2. Posteriormente los datos son enviados a el sistema de control, donde es procesado y da como resultado una nueva información.
3. Estos datos generados por el controlador, son enviados y codificados, de tal forma que la consigna sea introducida en el sistema a través de convertidores que denominamos actuadores o elementos finales de control.

2.4.3. Sistemas de Visualización

En el control de un sistema, ya sea grande, mediano o pequeño, ha sido un factor fundamental contar con información visual de cómo se está desarrollando el sistema. A medida que los sistemas de control avanzan, su complejidad se incrementa, por lo tanto los elementos que proporcionan la información al usuario también han aumentado su grado de complejidad. Desde los grandes cuadros de control, donde numerosas luces indicaban los diversos estados de la máquina, lo cual representaba una gran cantidad de horas de trabajo y de material, para llevar la señal al panel de control como ejemplo se observa un cuarto de control en la figura 2.17. La aparición de la informática permitió realizar este tipo de control de manera más sencilla, los grandes cuadros de control empezaban a convertirse en monitores que podían mostrar la misma información y cualquier cambio en la presentación era más sencillo de realizar.

Vista la necesidad, los fabricantes aprovecharon la oportunidad y crearon paquetes de software capaces de comunicarse con los sistemas de control existentes, permitiendo así una facilidad de uso no imaginada hasta el momento (Camacho Córdova, 2015).



Figura 2.17: Cuarto de control de una antigua central nuclear.

Fuente: (Horn, 2012)

2.4.4. Sistemas SCADA

Se denomina SCADA (Supervisory Control and Data Acquisition) que en castellano significa, control con supervisión y adquisición de datos, a cualquier software que facilite la obtención de los datos de un proceso y aplicando las herramientas apropiadas de comunicación, permita el control y supervisión del sistema. Según la definición se entiende que no se refiere a un sistema de control, más bien es una plataforma que permite monitorizar, supervisar, y controlar un proceso, a través de una interfaz que comunique los niveles de control (PLC) y los de gestión. Las prestaciones de un sistema SCADA son (Rodríguez Penin, 2012):

- Monitorización: es cuando los operadores de la planta obtienen los datos en tiempo real, esto es posible gracias a la interconexión del HMI con el controlador de la planta.
- Supervisión: con esta ventaja se evita la continua supervisión humana en el campo, ya que se puede observar la evolución de las variables del proceso desde el HMI.
- Mando: permite a los operadores activar o desactivar equipos remotamente. El usuario puede modificar los parámetros del proceso según sus necesidades directamente desde el ordenador.

2.4.5. Objetivos de los Sistemas SCADA

Los sistemas SCADA se definen como una herramienta de monitorización y supervisión. Entre sus objetivos a cumplir podemos resaltar (Aguirre Zapata, 2013):

- **Gestión:** Todos los datos obtenidos del proceso pueden ser recopilados e interpretados de diferentes maneras, ya sea representándola gráficamente, estadísticamente, mediante tabulaciones, etc., permitiendo visualizar la evolución del proceso, haciendo posible que el sistema se desarrolle con el rendimiento más óptimo.
- **Accesibilidad:** Todos los elementos que forman parte de un sistema pueden ser monitoreados y controlados, permitiendo la configuración de los valores de parametrización, para mejorar el funcionamiento de cada elemento a tan solo un clic ya sea en un cuarto de control u oficina de producción.
- **Conectividad:** Los protocolos de comunicación en la actualidad, permiten la interconexión entre sistemas de diferentes proveedores, esto quiere decir que son sistemas abiertos, los cuales evitan la existencia de lagunas informativas que pueden causar fallos en el funcionamiento o en la seguridad del sistema.
- **Economía:** Se reducirán costos puesto que se evitará que operarios estén constantemente revisando lo que ocurre en el proceso, ya que es más fácil ver lo que está pasando desde la oficina. Por lo tanto ciertas revisiones se volverán innecesarias.
- **Mantenimiento:** La adquisición de datos de un proceso se puede programar de forma que nos indique cuando se aproximen las fechas de revisión o cuando una maquina tenga más fallos de lo normal.
- **Ergonomía:** Procura que la relación usuario-proceso sea lo más amigable posible.
- **Flexibilidad:** Ahorro de tiempo y recursos, puesto que cualquier modificación que se quiera realizar al proceso, no representa una modificación física.

2.4.6. Componente de un Sistema SCADA

Los componentes principales que constituyen un sistema SCADA son:

Hardware: son los elementos por los cuales se trata y se gestiona la información recibida del proceso y se clasifica en:

- RTU (Unidad remota de telemetría)
- MTU (Unidad terminal maestra)
- Instrumentación de campo.
- Red de comunicación

Software: es el programa mediante el cual se genera la interfaz HMI, para la comunicación entre los distintos niveles de la pirámide de automatización, así como la comunicación entre el nivel de campo con los niveles de supervisión, producción y gerencia, adicionalmente se restringe el acceso al sistema a personas no autorizadas, y en caso de fallos se genera una señal de alarma (Hernández Cevallos & Ledesma Marcalla, 2010).

2.4.7. Software LabVIEW

LabVIEW es un entorno de desarrollo de Vis (Instrumentos Virtuales), ya que su apariencia física y funcionamiento imitan instrumentos reales. Este software fue creado por National Instruments en 1976, y salió al mercado por primera vez en 1986. LabVIEW emplea un lenguaje G, esto quiere decir que es un lenguaje de programación gráfica. El lenguaje gráfico permite hacer diseño, control y pruebas a través de los Toolkits (Kits de herramientas o mejoras) los cuales facilitan la programación. LabVIEW a través de una ventana interactiva permite diseñar interfaz con el usuario, basándose en lenguaje de programación gráfica, LabVIEW es compatible con herramientas que comparten similitud y puede trabajar con programación de otras áreas de aplicación, como lo es la plataforma Matlab. Como ventaja podemos resaltar que LabVIEW permite una fácil integración con hardware como lo son las tarjetas de medición, adquisición y procesamiento de datos. Se puede decir

que LabVIEW es una plataforma de simulación, en el cual se puede simular el panel frontal de cualquier instrumento, basado en componentes hardware con acceso a un computador. (Adriano Macas & García Míguez, 2014)

Un VI está conformado por tres aspectos principales:

- Front panel: El panel frontal es la interfaz entre el usuario y el programa.
- Block diagram: El diagrama de bloques contiene el código gráfico que define la funcionalidad del VI.
- Icon and connector panel: El panel de iconos y conectores identifica el VI y acepta parámetros procedentes de otros VIs, un VI dentro de otro VI se denomina subVI, un subVI corresponde a una subrutina en lenguajes de programación basados en texto.

2.4.7.1. Principales Funcionalidades del Software LabVIEW

Entre las herramientas más representativas que dispone el entorno de desarrollo LabVIEW se mencionan algunas de las funcionalidades a continuación:

- Comunicación y control de instrumentos de cualquier fabricante.
- Adquisición de datos y análisis matemáticos.
- Procesamiento de señal.
- Automatización industrial y programación de PACs (Controlador de Automatización Programable).
- Diseño embebido de micros y chips.
- Diseño de controladores: simulación, Prototipaje rápido, hardware en el ciclo y validación.
- Control y supervisión de procesos.
- Robótica.
- Visión artificial y control de movimiento.
- Domótica y redes de sensores inalámbricos.

2.4.7.2. Interfaz de Comunicación del Software LabVIEW

LabVIEW es un software que posee facilidad de uso, por lo tanto posee diferentes interfaces de comunicación, las más relevantes son:

- Puerto Paralelo
- Puerto Serie
- GPIB
- VXI
- PXI
- TPC/IP
- Bluetooth
- Irda
- USB
- OPC

2.5. Motores DC

La máquina de Corriente Continua fue la primera de las maquinas rotatorias en ser desarrollada, la razón de ello fue que en un principio no se pensó que la corriente alterna tuviera las ventajas que hoy se le conocen.

La primera máquina de corriente continua fue ideada por el belga Zénobe Gramme alrededor de 1860 y empleaba un enrollado de rotor especial llamado anillo de Gramme, para lograr la conmutación o rectificación del voltaje alterno generado, posteriormente, el físico Werner von Siemens y otros, contribuyeron al desarrollo de estas máquinas realizando mejoras, hasta llegar a los motores de corriente continua que conocemos hoy en día (Flores D. , 2017).

2.5.1. Características Constructivas

Los motores de corriente continua están formados principalmente de dos partes, el estator y el rotor, más las conexiones entre ambos.

Estator: es la parte fija de la máquina. Puede estar formado por un núcleo macizo o laminado. En los polos se ubica el enrollado de excitación de la máquina, los cuales permiten la distribución uniforme del flujo magnético necesario en el entrehierro como se observa en la figura 2.18. Otra parte importante que forma parte del estator son las escobillas y los carbones, que son dispositivos que se encargan de tomar y transmitir la corriente en forma continua entre el rotor y el estator. Su contacto eléctrico con el rotor se realiza a través de las delgas del mismo. Finalmente la estructura externa soportante se denomina carcasa y se donde se ubica el sistema de lubricación y los terminales de conexión (Barria, 2012).

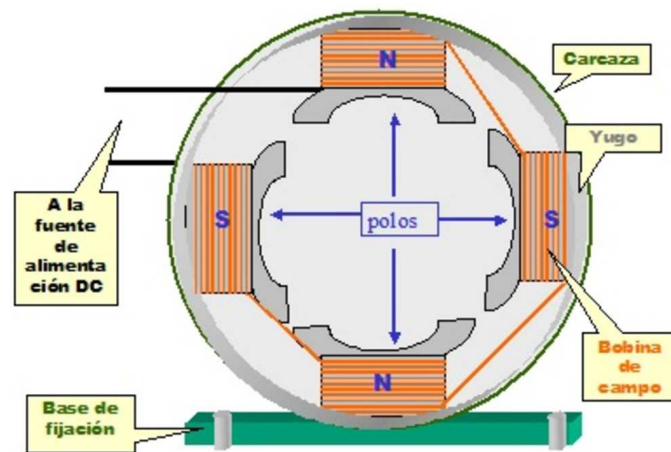


Figura 2.18: Partes que conforman el estator de un motor DC.

Fuente: (Flores L. , 2013)

Rotor: está formado por un núcleo de hierro laminado para evitar pérdidas por corrientes parásitas y de Foucault. El enrollado del rotor (armadura) está formada por bobinas que se ubican en ranuras del rotor alrededor del núcleo como se visualiza en la figura 2.19. Los terminales de las bobinas se conectan a las delgas, que en su conjunto forman el colector, donde hacen contacto rozante las escobillas o carbones, fijas al estator, permitiendo así la entrada o salida de corriente al enrollado de armadura (Barria, 2012).

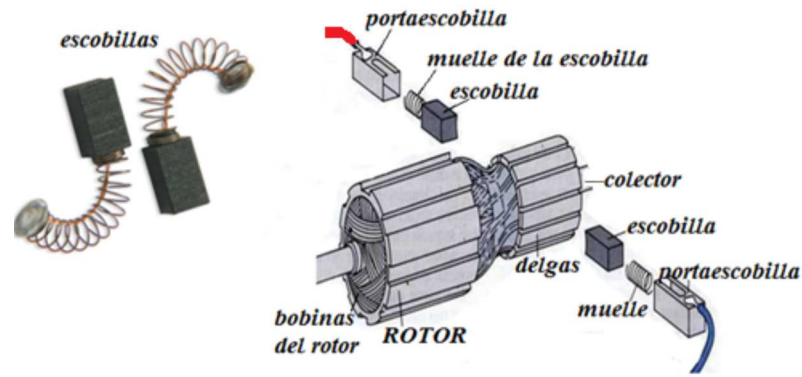


Figura 2.19: Partes que conforman el rotor de un motor DC.

Fuente: (AulaFacil, 2017)

2.5.2. Ventajas y Desventajas de los Motores de Corriente Continua

Los motores de corriente alterna (A.C.) son reemplazados por los de corriente continua (C.C.) en algunas aplicaciones donde la implementación de estos elementos es más simple y menos costosa, disminuyendo los recursos necesarios a utilizar, a continuación se describe las ventajas y desventajas de los motores CC.

Ventajas:

- Amplio rango de velocidades, ajustables de modo continuo y controlable con alta precisión.
- Características de torque-velocidad variable, constante o una combinación ideada por tramos.
- Rápida aceleración, desaceleración y cambio de sentido de giro.
- Posibilidad de frenado regenerativo (generación de energía eléctrica al frenar un motor).

Desventajas:

- Constructivamente más compleja de las máquinas de corriente alterna.
- La utilización de escobillas, colectores, etc., la hacen comparativamente menos robusta que las máquinas de corriente alterna.
- Requiere mayor mantenimiento.
- Tiene un mayor volumen y peso por Kw. (Pernía, 2011).

2.5.3. Clasificación de los Motores de Corriente Continua

- De excitación independiente
 - De excitación serie
 - El enrollado de campo serie se caracteriza por ser un enrollado de pocas vueltas con un alambre de grueso calibre, ya que debe conducir toda una fracción no despreciable de la corriente de armadura.
 - De excitación (shunt) o derivación
- El enrollado de campo paralelo (shunt) se caracteriza por ser un enrollado de muchas vueltas con un alambre de poco calibre, ya que en general soporta una corriente pequeña y se conecta a la tensión de la máquina.
- De excitación compuesta.

2.5.4. Modelado Matemático de los Motores DC

El modelo matemático lineal de un motor DC se basa en dos ecuaciones diferenciales ligadas entre sí: tanto la ecuación del modelo mecánico como la ecuación del modelo eléctrico del motor DC, como se puede apreciar en la figura 2.20. Puesto que ambos modelos se encuentran ligados entre sí, se puede generar una ecuación principal que permita describir la función de transferencia de la máquina.

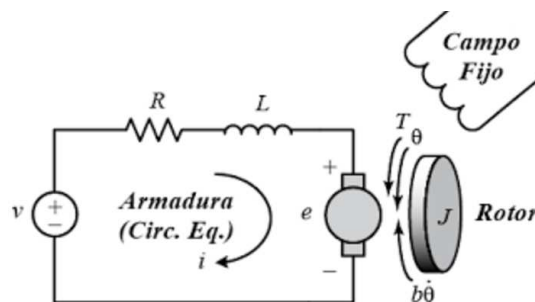


Figura 2.20: Esquema del circuito perteneciente a un motor DC.

Fuente: (Escobar, 2016)

Para conseguir el modelo eléctrico del motor DC, nos basamos en la Ley de Voltaje de Kirchhoff, donde efectuaremos el análisis del motor DC. Primero

identificaremos el voltaje aplicado al circuito de excitación como se observa en la ecuación 2.29, posteriormente se identificara el voltaje de armadura como se aprecia en la ecuación 2.30. (Meza Weber & Ramos Morales, 2015)

$$V_T = R * i(t) + L * \frac{di(t)}{dt} + E_a \quad \text{Ecuación 2.29}$$

$$E_a = K * w(t) \quad \text{Ecuación 2.30}$$

Una vez que se inicia el movimiento del rotor se genera la fuerza contra-electromotriz, ya que se produce un voltaje en el devanado de la armadura producto del campo magnético fijo del estator. En base a experimentos se puede definir la constante K, puesto que la fuerza contra-electromotriz es proporcional a la velocidad angular del eje del rotor.

Detalle:

L: Inductancia de Armadura y de Interpolo [Henrios]

R: Resistencia de la Armadura y de Interpolo [Ω]

V_T: Voltaje Aplicado al Circuito de Armadura [V]

E_a: Voltaje de Armadura [V]

i(t): Corriente de la Armadura en Funcion del Tiempo

w(t): Velocidad Angular en Funcion del Tiempo

K: Constante de la Fuerza ContraElectromotriz [$\frac{V}{rad/seg}$]

Ecuación eléctrica del motor DC conexión independiente en estado estable visualizadas en las siguientes ecuaciones.

$$V_T = R * i(t) + L * \frac{di(t)}{dt} + K * w(t) \quad \text{Ecuación 2.31}$$

$$L * \frac{di(t)}{dt} + R * i(t) = V_T - K * w(t) \quad \text{Ecuación 2.32}$$

Para obtener el modelo mecánico tomamos en cuenta la 2da ley de Newton sobre la rotación dinámica, la fuerza (Torque) como se observa en las siguientes ecuaciones.

$$T = T_{ind} - T_{roz} \quad \text{Ecuación 2.33}$$

$$J * \alpha = k * i(t) - f * w(t) \quad \text{Ecuación 2.34}$$

Detalle:

T : Torque [N.m.]

T_{ind} : Torque Inducido [N.m.]

T_{roz} : Torque de Rozamiento [N.m.]

J : Momento de Inercia del Rotor [Kg.m²]

α : Aceleracion Angular [Radianes]

F : Constante de Torque Electromagnético $\left[\frac{V}{rad/seg} \right]$

f : Coeficiente de Rozamiento Viscoso $\left[\frac{N.m.seg}{rad} \right]$

$w(t)$: Velocidad Angular en Función del Tiempo $\left[\frac{rad}{seg} \right]$

Observamos que las ecuaciones 2.32 y 2.36 están relacionadas por la función de la corriente.

$$J * \frac{dw}{dt} = (k * i(t)) - (f * w(t)) \quad \text{Ecuación 2.35}$$

$$J * \frac{dw}{dt} + f * w(t) = K * i(t) \quad \text{Ecuación 2.36}$$

Para facilitar la sustitución primero obtendremos la transformada de Laplace de ambas ecuaciones.

Transformada de Laplace de la Ecuación 2.32

$$V_T(s) = (SL + R)i(s) + Kw(s) \quad \text{Ecuación 2.37}$$

$$(SL + R)i(s) = V_T(s) - Kw(s) \quad \text{Ecuación 2.38}$$

Transformada de Laplace de la Ecuación 2.35

$$(sJ + f)w(s) = K * i(s) \quad \text{Ecuación 2.39}$$

$$i(s) = \frac{(sJ+f)w(s)}{K} \quad \text{Ecuación 2.40}$$

Por lo tanto $i(s)$ de la ecuación 2.40 es sustituida en la ecuación 2.38 para obtener la función de transferencia en lazo abierto como se demuestra en la ecuación 2.41, donde la velocidad de rotación se considera la salida y la tensión de la armadura se considera la entrada, mientras que en la ecuación 2.42 nos presenta la posición como salida y la tensión como entrada, esto se obtiene integrando la ecuación 2.41.

$$\frac{w(s)}{V_t(s)} = \frac{K}{((sJ+f)(SL+R)+K^2)} \quad \text{Ecuación 2.41}$$

$$\frac{\theta(s)}{V_t(s)} = \frac{K}{((sJ+f)(SL+R)+K^2)s} \quad \text{Ecuación 2.42}$$

CAPÍTULO III: DISEÑO DE MÓDULO DE SIMULACIÓN APLICANDO CONTROL PREDICTIVO

El módulo de simulación de prácticas está diseñado con la finalidad de analizar el comportamiento del sistema, utilizando técnicas de control predictivo vado en modelo. Esta es una técnica moderna de control, que en los últimos años está tomando fuerza en el diseño de los sistemas de control industriales, por lo que se ajusta a los requerimientos que demandan las empresas en la actualidad. Estos requerimientos, tales como la capacidad para soportar restricciones en las entradas y salidas de la planta a la hora de calcular la ley de control, obteniendo una mayor ventaja con respecto a otras estrategias de control.

3.1. Consideraciones Iniciales del Diseño del Sistema

En esta instancia se aborda el análisis y la descripción del funcionamiento del módulo de simulación de prácticas, de tal forma que sea caracterizado para la implementación del control predictivo de un motor DC.

3.1.1 Descripción del Módulo de Simulación

El módulo de simulación de un motor DC, es una herramienta de software que permite visualizar el comportamiento del sistema aplicando control predictivo tanto para el control de velocidad como para el control de posicionamiento de un motor DC. El sistema propuesto constan básicamente de: un ordenador con un procesador capaz de soportar los programas que se van a ejecutar en el desarrollo del proyecto tales como la plataforma de simulación Matlab, el entorno grafico Simulink que forma parte de Matlab, la interfaz LabVIEW para controlar y supervisar el proceso a través del PC.

El sistema propuesto trabaja siguiendo la secuencia que se explica a continuación: Una vez que el ordenado esta encendido, se debe dar inicio tanto a la plataforma de simulación Matlab como a LabVIEW, adicionalmente

se tiene que ejecutar el programa Virtual Serial Ports, luego en el programa Matlab se ingresan los datos pertenecientes a la planta, en este caso particular debemos ingresar los datos del modelo matemático del motor DC tanto para el control de velocidad así como para el control de posicionamiento, y los datos de sintonía del controlador predictivo, como punto importante se debe ingresar al programa Virtual serial ports driver el cual permite crear los puerto virtuales lo cual es fundamental al momento de comunicar ambos programas. Una vez que los datos de la planta y del controlador han sido ingresados, y todo se encuentre en orden se inicia con las pruebas de simulación como es proceder a coloca las consignas o los valores tanto para el control de posicionamiento como para el control de velocidad de los motores, esto se realiza a través de la interfaz LabVIEW, y de manera gráfica se observa el comportamiento del sistema por medio de una ventana del entorno grafico Simulink permitiendo realizar pruebas de velocidad y de giro en ambos motores, comparando y discutiendo los resultados, hasta que se concluya con las pruebas, como se observa en la figura 3.1.

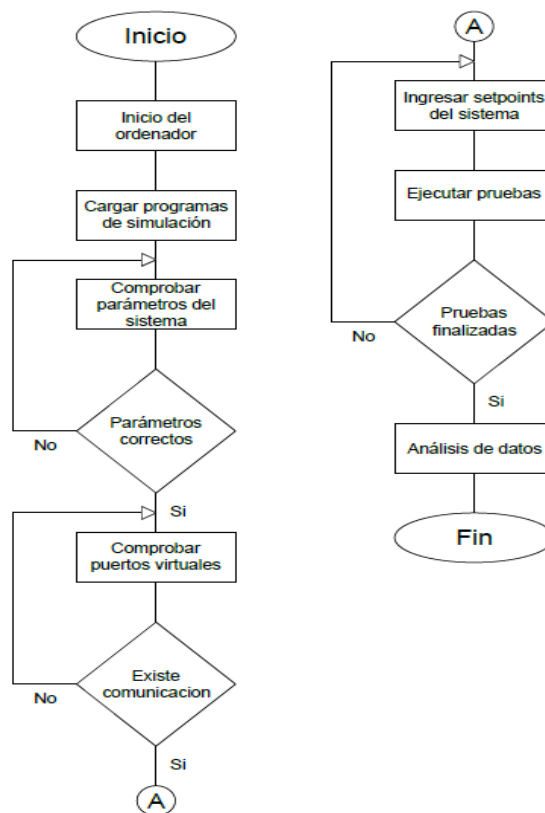


Figura 3.1: Diagrama de flujo del funcionamiento del módulo de simulación.

Fuente: Autor

3.1.2. Características del Módulo de Simulación

Con este proyecto se desea analizar el comportamiento de un motor DC para el control de velocidad y de posicionamiento, a través del uso de técnicas de control predictivo basado en modelo, por esta razón se diseña un módulo de simulación el cual posee las siguientes características:

- i. El software de simulación Matlab de MathWorks permite ejecutar la programación previamente obtenida en base a los datos de la planta y el controlador predictivo basado en modelo, se escogió esta plataforma de simulación ya que posee las herramientas necesarias para simular una planta en base a su modelo matemático real, también permite implementar un controlador en este caso se trabaja un control predictivo basado en modelo en el cual se pueden establecer los valores de sintonía del controlador MPC, y también nos muestra de forma gráfica el comportamiento del sistema. Como requerimientos mínimos del ordenador para poder instalar la plataforma Matlab se debe tener un procesador Intel o AMD x86-64, el disco duro requiere un espacio de 2 GB para Matlab y 4-6 GB para una instalación típica, también se necesita un espacio mínimo en la memoria RAM de 4 GB para ejecutar Matlab con Simulink, como observación tenemos que no se requiere ninguna tarjeta gráfica específica y se debe contar con un Puerto USB o CD-ROM para los drivers de Matlab (MathWorks, 2017).
- ii. El entorno Simulink es una herramienta que forma parte de la plataforma de simulación Matlab, la misma que se caracteriza por tener un lenguaje de programación gráfico, una de las razones que permitieron elegir este entorno es que posee un ambiente de trabajo intuitivo lo cual lo hace una herramienta fácil de utilizar, otro razón es que Simulink es compatible con LabVIEW y permite realizar un sistema SCADA. Debemos tomar en cuenta al momento de instalar la plataforma de Matlab activar la función de Simulink, para contar con el entorno de desarrollo.

- iii. La plataforma LabVIEW de National Instruments es la herramienta que permite realizar la interfaz entre el usuario y el controlador predictivo, diseñado en la plataforma de simulación Matlab, motivo por el cual se decidió elegir este software es que posee un entorno de programación gráfica, lo cual hace de la plataforma un entorno fácil de programación, otra característica que motivaron elegir este software es la comunicación entre VIs de diferentes programas, lo cual permite enviar los datos desde la interfaz LabVIEW hacia el controlador diseñado en Matlab. Como requisito para instalar la plataforma LabVIEW en un sistema operativo Windows 7/8/10 (32-64 bit), un procesador mínimo Pentium 4M o equivalente (32bit) - Pentium 4G1 o equivalente (64 bit), debe poseer un espacio de disco duro de 5GB incluido los controladores predeterminados, una memoria RAM 1GB (National Instruments, 2017).

- iv. La interfaz de comunicación entre el controlador predictivo diseñado en Simulink de Matlab y la interfaz de control y supervisión diseñada en LabVIEW, está a cargo de Virtual Serial Port Driver el cual está desarrollado por Eltima Software. Virtual port es un software que permite crear puertos serie virtuales para la comunicación entre ambos programas, teniendo un comportamiento igual que un puerto serie real, emulando todas sus configuraciones. Razón por la cual se decidió escoger este programa es que posee una facilidad de uso, y cumple las expectativas en cuanto a la creación de puertos series virtuales para la comunicación entre Matlab y LabVIEW. Se puede instalar este programa en un sistema operativo windows 7/8/10 de 32 o 64 bits, requiere un espacio en el disco duro mínimo de 3.9MB (Eltima Software, 2017).

3.1.3. Funciones

El módulo de simulación de un motor DC que se va a diseñar, realizará seis funciones principales, que describen en su totalidad el trabajo que desarrollará el sistema a controlar, los cuales se detallan a continuación:

- i. Supervisión a través de la interfaz LabVIEW y la plataforma virtual Matlab, el cual nos permitirá visualizar el comportamiento de la planta.
- ii. Adquisición de Datos del sistema mediante el uso de la herramienta virtual Matlab que permite observar el comportamiento del sistema de forma gráfica.
- iii. Mando del sistema mediante el uso de la herramienta virtual LabVIEW que a través de un entorno didáctico, permite modificar las variables del sistema.
- iv. Control del sistema a través de la plataforma virtual Matlab la cual se encargará de ejecutar la programación preestablecida del control predictivo aplicado al sistema a controlar.
- v. Visualizar de forma gráfica el comportamiento del sistema en respuesta a la orden generada por medio de la interfaz, aplicando o ejecutando algoritmos de control predictivo.
- vi. Generación de mensaje advirtiendo el mal funcionamiento en el caso de ingresar un dato erróneo.

3.2 Análisis de las Plataformas de Simulación

En los siguientes capítulos se describe el análisis del diseño del módulo de simulación, dándole un enfoque desde el punto de vista del Software.

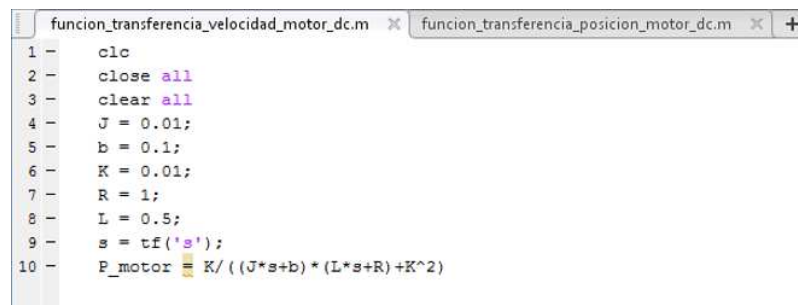
3.2.1. Análisis de la Plataforma Virtual Matlab

En este acápite se analizan las herramientas de la plataforma virtual Matlab, que van a ser implementadas para realizar el control de

posicionamiento y de velocidad de un motor DC aplicando técnicas de controlador predictivo basada en modelo.

3.2.1.1. Script

Al crear un New Script en Matlab, se genera la ventana del Workspace donde se debe ingresar las características de la función de transferencia tanto para el control de posicionamiento, como para el control de velocidad del motor DC como se observa en la figura 3.2. Una vez ingresados los datos de la planta, se procede a compilar el código, esto permite almacenar los datos de la función de transferencia de la planta como Variable.



```
funcion_transferencia_velocidad_motor_dc.m x funcion_transferencia_posicion_motor_dc.m x +
1 -   clc
2 -   close all
3 -   clear all
4 -   J = 0.01;
5 -   b = 0.1;
6 -   K = 0.01;
7 -   R = 1;
8 -   L = 0.5;
9 -   s = tf('s');
10 -  P_motor = K / ((J*s+b)*(L*s+R)+K^2)
```

Figura 3.2: Datos de la planta en el Workspace.

Fuente: Autor

3.2.2. Análisis del Entorno Grafico de Programación Simulink

Simulink es un toolbox especial de Matlab que permite simular el comportamiento dinámico de los sistemas, tanto lineales como no lineales, por tal motivo es una herramienta muy versátil a la hora de diseñar un controlador MPC, ya que nos permite ingresar los datos de la planta, además de generar un MPC tool, ingresar señales de entrada, visualizar el comportamiento de las salidas, etc., tal como se describe los diferentes bloques utilizados para el desarrollo del proyecto, en los siguientes acápite.

3.2.2.1. Transfer Fcn

El bloque Transfer Fcn se obtiene llamando esta opción desde la librería de Simulink, es un bloque que posee una entrada y una salida como se observa en la figura 3.3, sirve para representar y simular los datos que previamente han sido ingresados en el script de Matlab.

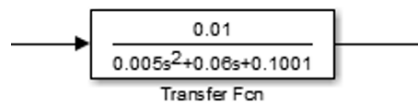


Figura 3.3: Bloque Transfer Fcn de Simulink.

Fuente: Autor

3.2.2.2. MPC Toolbox

El bloque MPC tool es una herramienta que permite diseñar y simular un controlador predictivo basado en modelo. Permite seleccionar un modelo de planta específico, además se pueden modificar las variables de sintonía del MPC tal como: las perturbaciones, el horizonte de control, restricciones y los pesos. Posee tres entradas y una salida, la entrada (mo) se refiere a las salidas medidas, la entrada (ref) significa que se ingresa la referencia o Setpoint, y la entrada (md) se refiere a las perturbaciones medidas, y la salida (mv) se refiere a las variables manipuladas como se visualiza en la figura 3.4.

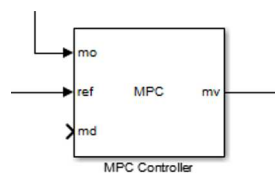


Figura 3.4: Descripción del bloque MPC tool de Simulink.

Fuente: Autor

3.2.2.3. Serial Configuration

El bloque Serial Configuration como se aprecia en la figura 3.5, permite configurar los parámetros de un puerto serie que puede ser utilizado para enviar y recibir datos. Cada bloque ya sea de envío o recepción de datos

posee un bloque de configuraciones independiente. Los parámetros en el bloque Serial Configuration que se deben ajustar son: primero seleccionar un puerto disponible de la lista, especificar la velocidad con que se transmiten los bits para la interfaz serie (Baud rate), indicar el número de bits de datos que se transmiten por medio de la interfaz serie (Data bits), elegir la paridad (Parity), indicar los bits de paridad (Stop bits), especificar el orden de los bytes (Byte order), el control de flujo (flow control) y el Timeout.

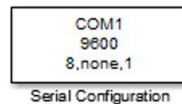


Figura 3.5: Bloque Serial Configuration.

Fuente: Autor

3.2.2.4. Serial Receive

El bloque Serial Receive como se observa en la figura 3.6., permite recibir datos binarios a través del puerto serie, este bloque abre y configura una interfaz a una dirección remota especificada utilizando el protocolo serial. La configuración y la inicialización se realizan al inicio de la ejecución del modelo. Los parámetros en el bloque Serial Receive que se deben ajustar son: primero seleccionar un puerto disponible de la lista, el encabezamiento especifica el comienzo de un nuevo bloque de datos (Header), el terminador indica el final del bloque de datos (Terminator), se debe especificar el tamaño de datos de salida (Data size), se refiere al tipo de datos de salida que se recibirá del bloque (Data type), indica el orden de bytes del dispositivo para los datos binarios (Byte order), modo de bloqueo permite bloquear la simulación mientras recibe datos (Enable blocking mode), se debe especificar la acción que debe realizar el bloque cuando los datos no están disponibles (Action when data is unavailable), en valor personalizado se especifica un valor para que el bloque emita cuando no recibe datos actuales (Custom value), especifica el tiempo de muestreo del bloque durante la simulación (Block simple time).

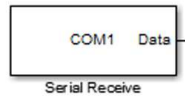


Figura 3.6: Bloque Serial Receive.

Fuente: Autor

3.2.2.5. Scope

El bloque de Scope posee una señal de entrada como se visualiza en la figura 3.7, este bloque muestra las señales de visualización generadas durante la simulación. Las señales mostradas se encuentran en el dominio del tiempo con respecto al tiempo de simulación, estas señales pueden ser de tipo continua o discreta, los tipos de datos de señal pueden ser cualquiera de los admitidos por Simulink, incluidos los reales, complejos, de punto fijo y enumerados, las dimensiones de la señal pueden ser a escala, también pueden ser unidimensional, bidimensional o multidimensional.



Figura 3.7: Bloque de visualización Scope.

Fuente: Autor

3.2.2.6. Diseño de la Etapa en Matlab

A continuación, se presenta el desarrollo que se realizó en el entorno Simulink, para efectuar el control de posicionamiento y de velocidad de un motor DC aplicando la herramienta MPCtoolbox. En el siguiente diseño se complementan los bloques de función, que fueron detallados en el capítulo anterior, los cuales conforman el diseño de la parte de control del módulo de simulación.

En la figura 3.8, se puede apreciar que el diseño integra un puerto virtual "COM 1", en el cual ingresa el Setpoint enviado desde el entorno LabVIEW, adicionalmente tenemos el bloque MPC en el cual se registran los parámetros de sintonía, también se requiere del bloque Transfer Fcn, en el cual se debe

ingresar el modelo de la planta, adicionalmente se insertó un bloque de ruidos aleatorios, el cual permite simular la planta con perturbaciones, y el comportamiento del sistema es visualizado a través de dos “Scope”, los cuales nos permiten comparar las diferentes señales del sistema, y analizar los resultados obtenidos.

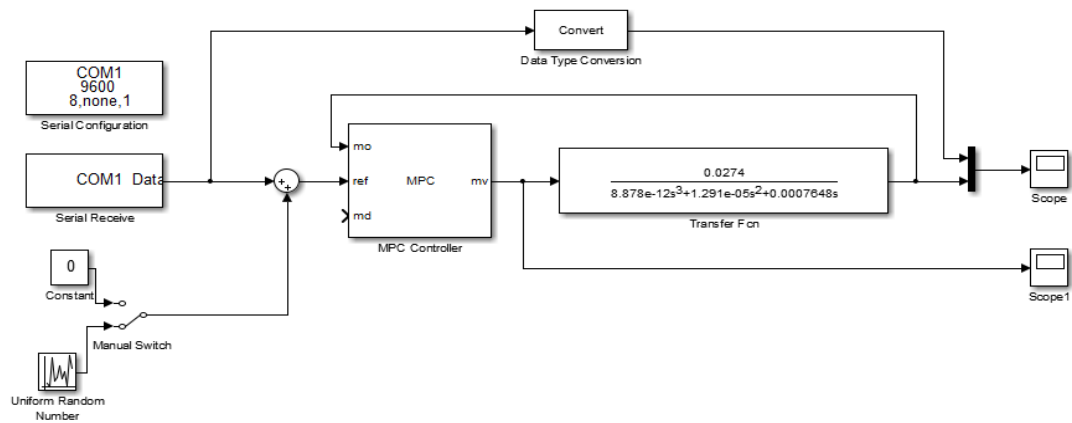


Figura 3.8: Controlador MPC del entorno gráfico Simulink.

Fuente: Autor

3.2.3. Análisis de la Plataforma Virtual LabVIEW

LabVIEW es un ambiente de desarrollo de programación que está a cargo de National Instruments. Los VIs de LabVIEW permiten diseñar instrumentos virtuales parecidos a los reales, a través de su lenguaje de programación grafica posibilitando la creación de programas en forma de diagramas de bloques. En el siguiente capítulo se detallan las herramientas utilizadas para el desarrollo del proyecto.

3.2.3.1. Front Panel

La ventana del panel frontal es la interfaz entre los VIs y el usuario. Una vez abierta la ventana del Front Panel, aparece la opción Functions, la cual nos muestra una lista de opciones como se aprecia en la figura 3.9. El panel frontal puede albergar controles e indicadores, los cuales son los terminales interactivos de las entradas y salidas. Los controles e indicadores colocados en el Front Panel, automáticamente se crean en el Block Diagram.

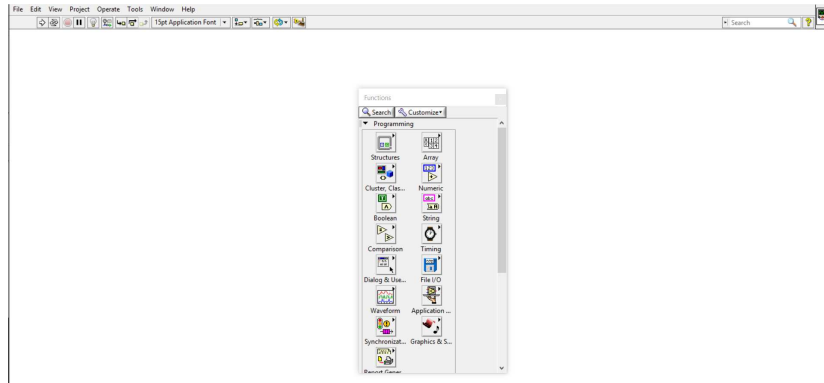


Figura 3.9: Ventana Front Panel de LabVIEW.

Fuente: Autor

3.2.3.2. Block Diagram

La ventana de diagrama de bloques es la que posee el código fuente gráfico de un programa de LabVIEW como se muestra en la figura 3.10. Los objetos del Front Panel aparecen como terminales en la ventana del Block Diagram, estos terminales reflejan los cambios realizados en los objetos del Front Panel y viceversa. Para visualizar la ventana de diagrama de bloques, se debe dar clic en la opción Window del menú y posteriormente seleccionar Show Block Diagram.

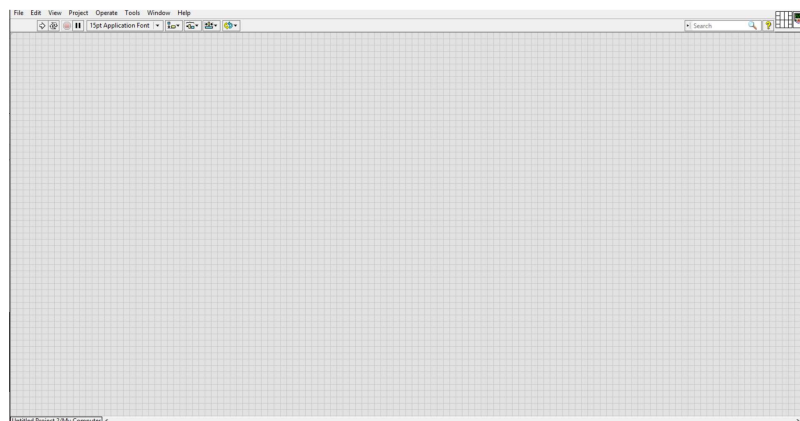


Figura 3.10: Ventana Block Diagram de LabVIEW.

Fuente: Autor

3.2.3.3. While Loop

El ciclo While se encuentra en la paleta Structures. Es similar a un ciclo Do o a un ciclo Repeat-Until en un lenguaje de programación escrito. El ciclo While ejecuta el código que contiene hasta la terminal condicional, una terminal de entrada, recibe un valor Booleano específico. La terminal de condicional puede realizar manejos básicos de error. La terminal de iteraciones es un terminal de salida que contiene el número de iteraciones terminadas. En un ciclo While las iteraciones siempre empiezan en cero.

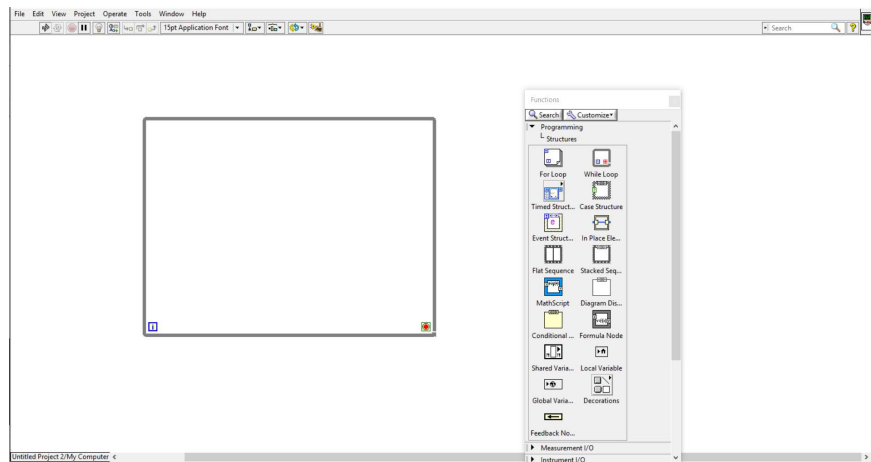


Figura 3.11: Ciclo While Loop.

Fuente: Autor

3.2.3.4. Numeric Data

El bloque Numeric, como se visualiza en la figura 3.12, se caracteriza por los tipos de dato numérico: entero con signo de palabra, Bits de almacenamiento en disco: 16, número aproximado de dígitos decimales: 4, rango aproximado: -32.768 a 32.767. La diferencia entre los tipos de datos numéricos es el número de bits que usan para almacenar datos y los valores de datos que representan.

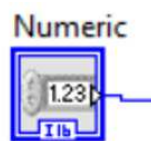


Figura 3.12: Bloque Numeric Data.

Fuente: Autor

3.2.3.5. Build Array

Cuando se elige la función Build Array en el diagrama de bloques, posee solo una entrada disponible como se observa en la figura 3.13, se pueden agregar más entradas dando clic derecho en la entrada. La función Build Array funciona de dos modos: si selecciona Concatenar Entrada, la función agrega todas las entradas en orden, formando una matriz de salida de la misma dimensión que la entrada de la matriz conectada. Si no selecciona Concatenar Entrada, la función construye una matriz de salida de una dimensión más alta que la dimensión de las entradas

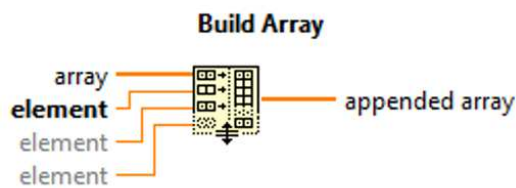


Figura 3.13: Bloque Build Array.

Fuente: (National Instruments, 2012)

3.2.3.6. Byte Array to String

El bloque de función Byte Array to String como se muestra en la figura 3.14, convierte una matriz de byte sin signo que representan caracteres ASCII en una cadena. Unsigned Byte Array es la matriz de valores ASCII que se desea convertir. String es el resultado de interpretar cada elemento de la matriz como un valor ASCII y formar una cadena fuera de los caracteres correspondientes.

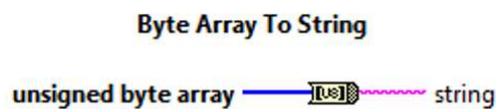


Figura 3.14: Bloque Byte Array to String.

Fuente: (National Instruments, 2012)

3.2.3.7. VISA Serial Port

El bloque de función Visa Serial Port como se observa en la figura 3.15, inicializa el puerto serie con las configuraciones especificadas. Se lo identifica por el nombre VISA Serial. Los parámetros a configurar son: enable termination char, termination char, Timeout, VISA resource name, baud rate, data bits, parity, error in, stop bits, flow control, VISA resource name out, error out.

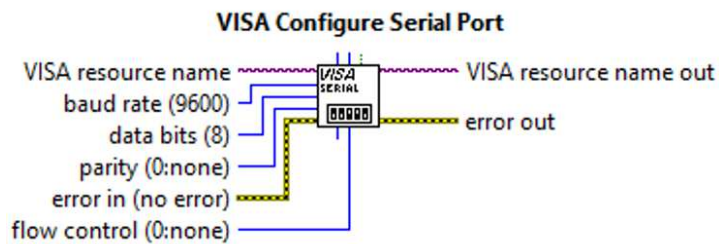


Figura 3.15: Bloque VISA Serial Port.

Fuente: (National Instruments, 2012)

3.2.3.8. VISA Write

El bloque de función VISA Write como se visualiza en la figura 3.16, escribe los datos de bufer de escritura en el dispositivo o la interfaz especificada por el nombre de recurso VISA. Los parámetros de configuración son: VISA resource name, write buffer, error in, VISA resource name out, return count, error out.

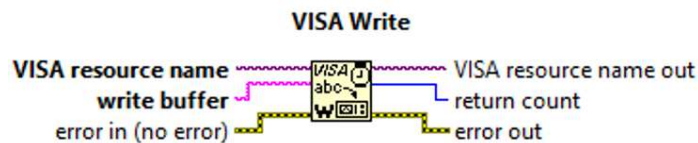


Figura 3.16: Bloque VISA Write.

Fuente: (National Instruments, 2012)

3.2.3.9. VISA Open

El bloque de función VISA Open como se muestra en la figura 3.17, abre una sesión en el dispositivo especificado por el nombre de recurso VISA y

devuelve un identificador de sesión. Los parámetros a configurar son: VISA Open Timeout, VISA resource name, duplicate sesión, Access mode, error in, VISA resource name out, error out.

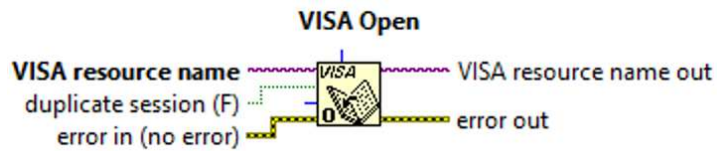


Figura 3.17: Bloque VISA Open.

Fuente: (National Instruments, 2012)

3.2.3.10. VISA Close

El bloque de función VISA Close como se aprecia en la figura 3.18, cierra una sesión de dispositivo o un objeto de evento especificado por el nombre de recurso VISA. Los parámetros de configuración son: VISA resource name, error in, error out.

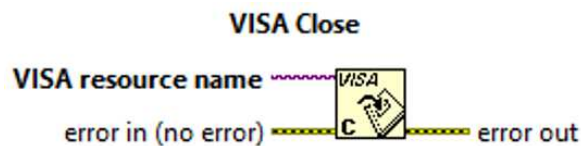


Figura 3.18: Bloque VISA Close.

Fuente: (National Instruments, 2012)

3.2.3.11. Wait Until Next ms Multiple

La función Wait Until Next ms Multiple como se observa en la figura 3.19, espera hasta que el valor del temporizador de milisegundos se convierta en un múltiplo del múltiplo de ms especificado. También puede ser útil para sincronizar actividades. Puede controlar la velocidad de ejecución en un bucle. Los parámetros a configurar son: millisecond multiple, millisecond multiple value.

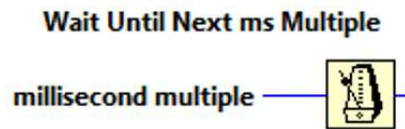


Figura 3.19: Bloque Wait Until Next ms Multiple.

Fuente: (National Instruments, 2012)

3.2.3.12. Diseño de la Etapa LabVIEW

A continuación se presenta el desarrollo que se llevó a cabo en la plataforma LabVIEW, para efectuar la interfaz entre el usuario y el sistema de control inmerso en Matlab.

En el siguiente diseño se complementan los bloques de funciones, que fueron detallados en el capítulo anterior, los cuales conforman el diseño de la etapa de visualización y mando del sistema.

En la figura 3.20, se puede apreciar que el diseño integra un puerto virtual de comunicación llamado "COM 2", por el cual son enviados los parámetros del Setpoint hacia el controlador MPC desarrollado en el entorno Matlab, adicionalmente se obtienen el bloque "VISA", en el cual se ingresan los parámetros de configuración para la comunicación serial, también se requiere del bloque "Numeric" en el cual se ingresan los valor de referencia deseados, adicionalmente se insertó los bloques de función "Build Array" y "Byte Array to String", los cuales son fundamentales para convertir el dato enviado desde LabVIEW y que Matlab lo pueda recibir e interpretar.

A través de la interfaz creada en el entorno virtual LabVIEW se modifica el Setpoint para el control de posicionamiento del eje del motor. La posición responde a un Setpoint que va desde -120 grados hasta 120 grados positivos, adicionalmente se visualiza la señal de entrada por medio de un Waveform Chart como se muestra en la figura 3.21.

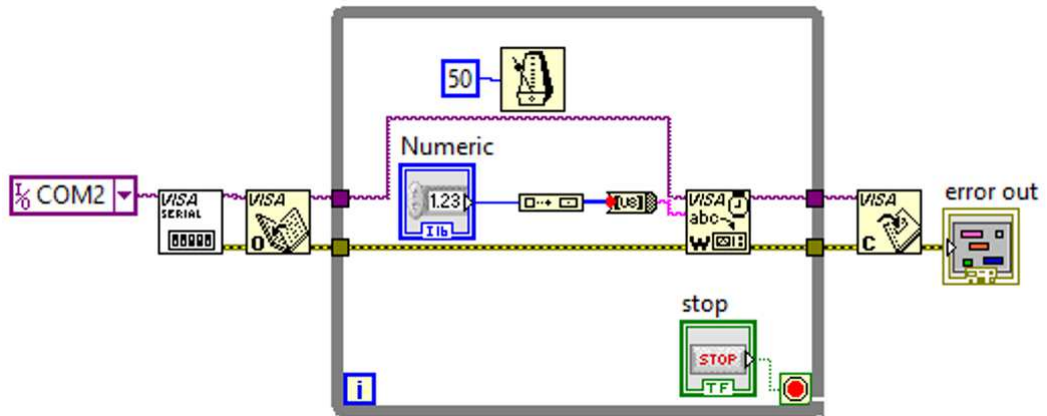


Figura 3.20: Configuración de la interfaz en la plataforma LabVIEW.

Fuente: Autor

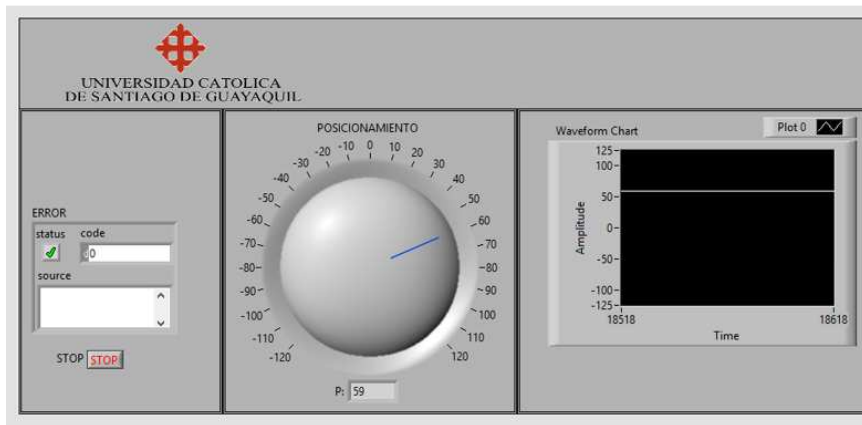


Figura 3.21: Interfaz para el control de posicionamiento.

Fuente: Autor

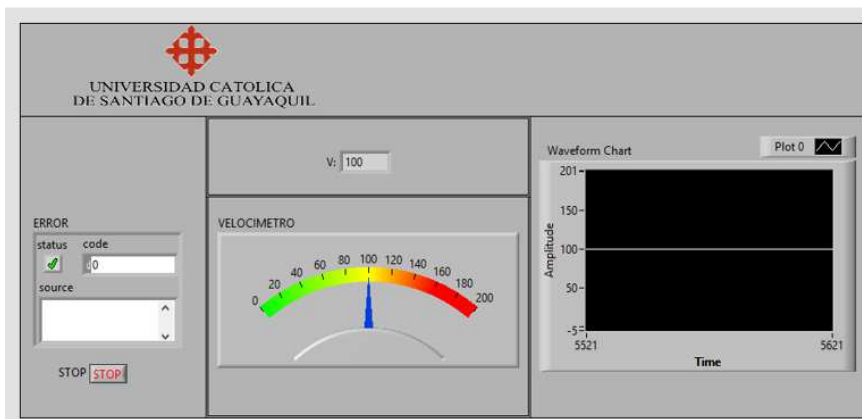


Figura 3.22: Interfaz para el control de velocidad.

Fuente: Autor

Para el control de velocidad del eje del rotor de un motor DC se realiza a través de la interfaz creada como se observa en la figura 3.22. La velocidad del motor responde a un Setpoint que va desde 0 hasta 200RPM, adicionalmente se visualiza la señal de entrada al sistema por medio de un Waveform Chart.

3.2.4. Análisis de los Puertos Virtuales

La interfaz virtual está a cargo del programa Virtual Serial Port Driver, permite la creación de un par de puertos virtuales, en este caso particular es fundamental para realizar la comunicación entre el programa Matlab y el programa LabVIEW.

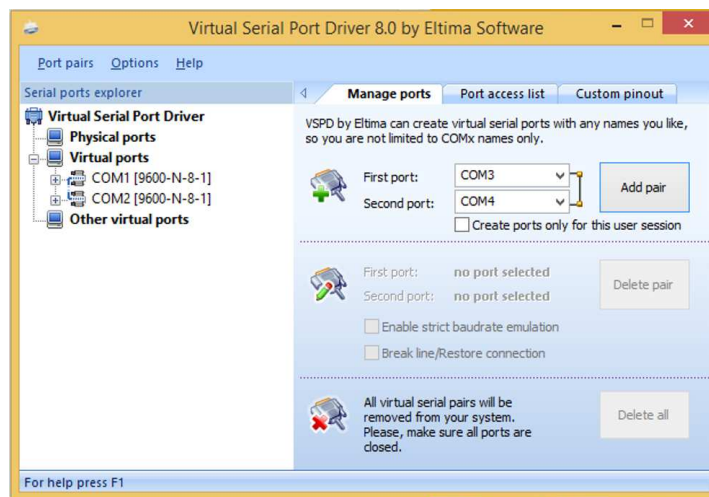


Figura 3. 23: Ventana del Software Virtual Serial Port Driver.

Fuente: Autor

3.3. Análisis de Algoritmos

El modelo matemático de un sistema a controlar debe ser idéntico al real, puesto que son la base fundamental, para poder manipular o controlar dichos sistemas con una mayor precisión, forzando a que el sistema se comporte de manera específica, disminuyendo los errores.

3.3.1 Análisis del Modelo Matemático de la Planta

El módulo de simulación requiere del modelo matemático tanto para el control de posicionamiento como para el control de velocidad de un motor DC, y con estas funciones de transferencias, posteriormente ser aplicado en la programación grafica del simulador de control predictivo basado en modelos de la herramienta Simulink.

Primero debemos considerar al motor DC como un sistema de tipo SISO (Single Input, Single Output), donde la entrada del sistema es el voltaje de alimentación del motor, y a través de la transformación de energía eléctrica en energía mecánica, la salida del sistema será la velocidad angular del motor o el posicionamiento en grados del motor, dependiendo de la variable que se desee controlar en el simulador.

3.3.1.1. Modelo Matemático para el Control de Posicionamiento del Eje de un Motor DC

El modelo matemático lineal de un motor DC se basa en dos ecuaciones diferenciales enlazadas entre sí: la primera ecuación es del modelo mecánico del motor DC, mientras que la otra ecuación es el modelo eléctrico del motor DC. debido a que ambos modelos se encuentran relacionados, se puede generar una ecuación principal que nos permita describir la función de transferencia de la máquina en la cual relacione la salida del sistema en este caso tenemos el posicionamiento del eje θ (*theta*) del motor, y la entrada del sistema el voltaje de alimentación aplicado a la armadura del motor.

Detalles:

$V_T(s)$: Voltaje Aplicado al Circuito de Armadura [V]

$\theta(s)$: Posicionamiento del Eje del Motor [rad]

K : Constante de la Fuerza ContraElectromotriz $\left[\frac{V}{\text{rad}/\text{seg}} \right]$

R_a : Resistencia de la Armadura y de Interpolo [Ω]

f : Coeficiente de Rozamiento Viscoso [N.m.seg]

J : Momento de Inercia del Rotor [$Kg.m^2$]

L : Inductancia de Armadura y de Interpolo [Henrios]

Luego de relacionar las ecuaciones del modelo eléctrico del motor y las ecuaciones del modelo mecánico del mismo, obtenemos una ecuación general para el control de posicionamiento del motor en base a una tención de entrada aplicada a la armadura del motor, como se demuestra en la ecuación 3.1.

$$\frac{\theta(s)}{V_t(s)} = \frac{K}{((SJ+f)(SL+Ra)+K^2)s} \left[\frac{rad}{V} \right] \quad \text{Ecuación 3.1}$$

En este caso particular para el control de posicionamiento del eje de un motor DC, los valores de los parámetros que posee el motor DC, fueron obtenidos en base a pruebas experimentales y se obtuvo como resultado los siguientes parámetros:

$$J = 3.2284^{-6} [Kg.m^2]$$

$$f = 3.5077^{-6} [N.m.seg]$$

$$K = 0.0274 \left[\frac{V}{rad/seg} \right]$$

$$Ra = 4 [\Omega]$$

$$L = 2.75^{-6} [Henrios]$$

Luego de reemplazar los valores experimentales del motor DC, y se obtuvo una función de transferencia con las siguientes características:

$$\frac{\theta}{V} = \frac{0.027}{8.878^{-12}S^3 + 1.291^{-5}S^2 + 0.0007648 S} \quad \text{Ecuación 3.2}$$

3.3.1.2. Modelo Matemático para el Control de Velocidad de un Motor DC

El modelo matemático para el control de velocidad describir la función de transferencia de la máquina en la cual relacione la salida del sistema en

este caso tenemos la velocidad angular ω (*omega*) del motor, y la entrada del sistema el voltaje de alimentación aplicado a la armadura del motor.

Detalles:

$V_T(s)$: Voltaje Aplicado al Circuito de Armadura [V]

$\omega(s)$: Velocidad Angular del Motor $\left[\frac{\text{rad}}{\text{seg}}\right]$

K : Constante de la Fuerza ContraElectromotriz $\left[\frac{\text{V}}{\text{rad}/\text{seg}}\right]$

R_a : Resistencia de la Armadura y de Interpolo [Ω]

f : Coeficiente de Rozamiento Viscoso [N.m.seg]

J : Momento de Inercia del Rotor [Kg.m²]

L : Inductancia de Armadura y de Interpolo [Henrios]

Luego de relacionar las ecuaciones del modelo eléctrico del motor y las ecuaciones del modelo mecánico del mismo, obtenemos una ecuación general para el control de posicionamiento del motor, por lo tanto se debe derivar la ecuación 3.1 del inciso anterior, para obtener la ecuación de la velocidad del motor DC, como se demuestra en la ecuación 3.3.

$$\frac{\omega(s)}{V_t(s)} = \frac{K}{((SJ+f)(SL+R)+K^2)} \left[\frac{\text{rad}/\text{sec}}{\text{V}}\right] \quad \text{Ecuación 3.3}$$

En este caso particular para el control de velocidad de un motor DC, los valores de los parámetros que posee el motor DC, fueron obtenidos en base a pruebas experimentales y se obtuvo como resultado los siguientes parámetros:

$$J = 0.01 \text{ [Kg.m}^2\text{]}$$

$$f = 0.1 \text{ [N.m.seg]}$$

$$K = 0.01 \left[\frac{\text{V}}{\text{rad}/\text{seg}}\right]$$

$$R_a = 1 \text{ } [\Omega]$$

$$L = 0.5 \text{ [Henrios]}$$

Luego de reemplazar los valores experimentales del motor DC, y se obtuvo una función de transferencia con las siguientes características:

$$\frac{\omega}{V} = \frac{0.01}{0.005S^2 + 0.06S + 0.1001} \quad \text{Ecuación 3.4}$$

3.2.3. Análisis y Diseño del Módulo de Simulación.

En este punto se describe el diseño del módulo de simulación, a través de las plataformas virtuales que permiten simular una planta en base a un modelo real, aplicando técnicas de control predictivo basado en modelo y supervisado a través de una interfaz HMI.

3.2.3.1. Matlab

El sistema propuesto fue diseñado siguiendo la secuencia que se explica a continuación: Una vez que se halla ingresado a la plataforma de simulación Matlab, se debe crear un nuevo Script, a continuación en la ventana del Workspace se ingresa la función de transferencia correspondiente a la planta, posteriormente se compila el programa para ver si existe algún error, adicionalmente la función de transferencia de la planta se guarda como Variable, tal como se describe en la figura 3.24.

3.2.3.2. Simulink

Después que se tiene el modelo de la planta, se procede a llamar al "mpc" desde el Command Windows, posteriormente se importa el modelo de la planta que previamente fue almacenado como Variable, a continuación se ajustan los parámetros de sintonía del MPC tales como: tiempo de muestreo, horizonte de predicción, horizonte de control, restricciones de entrada, restricciones de salida, entre otros, una vez listos los parámetros se procede a exportar el controlador MPC del modelo de la planta hacia el bloque MPC de Simulink, luego integramos los demás elementos que conforman el sistema

como son: el bloque Transfer Fcn, el bloque Serial Configuration, el bloque Serial Receive, el Scope y un Display para visualizar la constante recibida. Luego se ingresa un Setpoint y se procede a compilar el programa para verificar la respuesta del controlador MPC, tal como se describe en la figura 3.25.

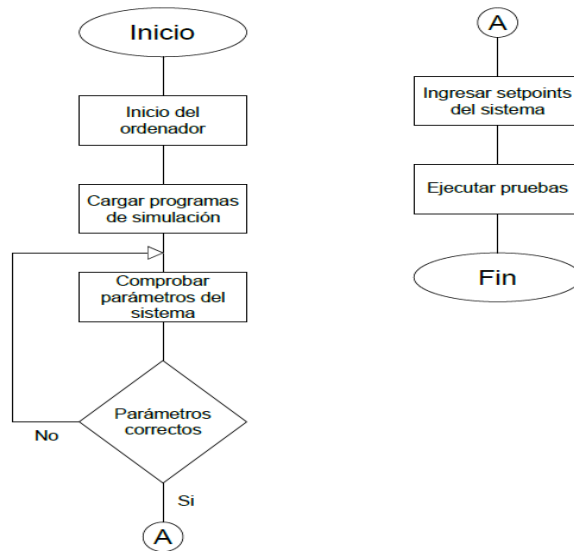


Figura 3.24: Diagrama de flujo del Desarrollo del Proyecto en Matlab.

Fuente: Autor

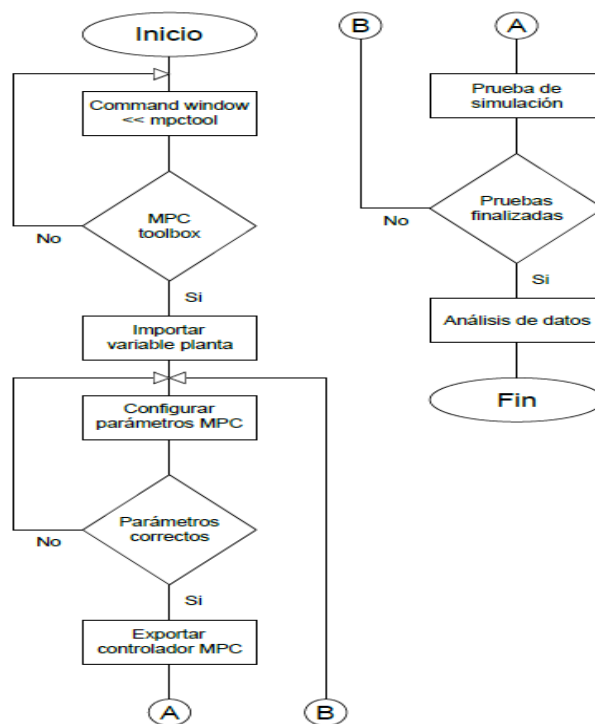


Figura 3.25: Diagrama de flujo del Desarrollo del Proyecto en Simulink.

Fuente: Autor

3.2.3.3. LabVIEW

Posterior al diseño del controlador, se da inicio al diseño de la interfaz de visualización y control a través de la plataforma LabVIEW, primero se crea un While Loop en el Block Diagram, luego se crean el Numeric Data en el cual se coloca el Setpoint del sistema, posteriormente se crea el Build Array el cual forma una matriz de salida de la misma dimensión que la entrada de la matriz conectada, después se crea un bloque de funciones Byte Array to String convierte una matriz de byte sin signo que representan caracteres ASCII en una cadena estos bloques se crean ya que LabVIEW envía datos enteros, pero Matlab los recibe como un arreglo numérico, para la comunicación entre el VI de LabVIEW y Matlab se debe crear el puerto VISA, el cual deben ser configurado previamente a través del bloque de funciones VISA Configure Serial Port, además se deben configurar el bloque de funciones VISA Write, y los bloques de funciones Visa Open y Visa Close, finalmente se procede a compilar el programa, tal como se describe en la figura 3.26.

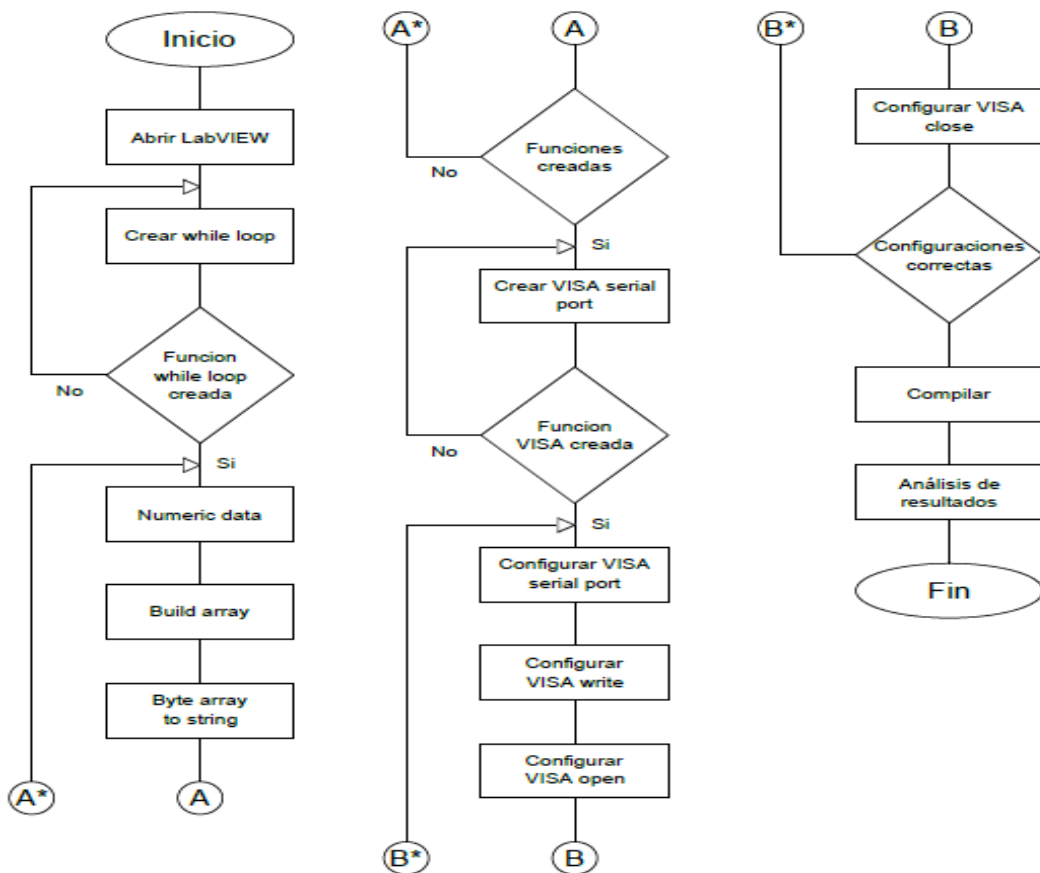


Figura 3.26: Diagrama de flujo del Desarrollo del Proyecto en LabVIEW.

Fuente: Autor

3.2.3.4. Virtual Serial Port

Para realizar la comunicación, seleccionamos el Software Virtual Serial Port Driver, posteriormente se selecciona la opción Manage Ports, a continuación se debe elegir que nombre del COM que va a representar el primer puerto y así mismo se debe elegir para el segundo puerto, se debe tomar en cuenta que los dos puertos no tengan el mismo nombre ya que esto no permitiría la comunicación. Finalmente damos clic en Add Pair para crear el par de puertos virtuales. Nota: una vez creados los puertos no hay que volver a repetir el proceso. Se debe tomar en cuenta los nombres seleccionados en la configuración de los Virtual Ports, para asignarle un port diferente tanto para la plataforma Matlab como para LabVIEW.

3.4. Pruebas de Funcionamiento

En el siguiente acápite se muestran las pruebas de funcionamiento tanto para el control de posicionamiento como para el control de velocidad de un motor DC, sometido a simulación sin controlador en lazo abierto y en lazo cerrado, y aplicando un controlador MPC.

3.4.1. Consideraciones Iniciales

Las pruebas de simulación que se van a ejecutar se dividen en dos secciones: la primera prueba de simulación es de la planta sin controlador, y la segunda prueba de simulación es de la planta con controlador. Estas prácticas se deben repetir tanto para el modelo matemático del control de posicionamiento del motor DC, como para el modelo matemático del control de velocidad del motor DC.

3.4.2. Pruebas de Simulación para el Control de posicionamiento

En el siguiente acápite se detalla la simulación del sistema diseñado, implementando del modelo matemático para el control de posicionamiento del eje del rotor de un motor DC.

A través de la plataforma de simulación Matlab primero se realizó una simulación de la planta en lazo abierto, ingresando la función de transferencia del modelo matemático previamente obtenido en la ecuación 3.23, la cual fue sometida a una señal de entrada paso con una amplitud de 12V. Como resultado obtuvimos una señal con una amplitud y un tiempo de estabilización bastante grande, tal como se observa en la figura 3.27.

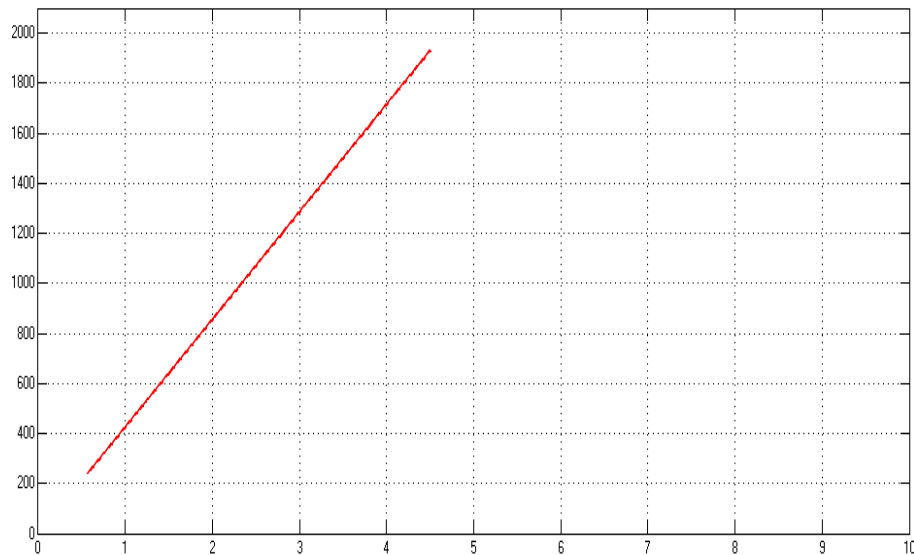


Figura 3.27: Respuesta de la planta conectada en Lazo Abierto.

Fuente: Autor

Luego se realizó la simulación de la planta en lazo cerrado con el modelo matemático para el control de posicionamiento, y fue sometida a una señal de entrada paso con una amplitud de 12V. Como resultado obtuvimos una señal con una amplitud de 12 la cual obedece al Setpoint y presenta un tiempo de estabilización de aproximadamente 1 seg, tal como se observa en la figura 3.28.

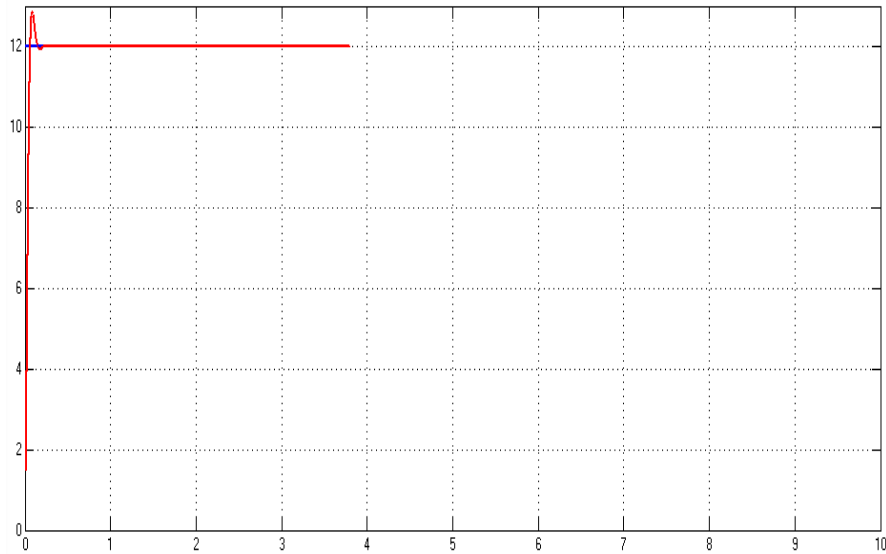


Figura 3.28: Respuesta de la planta conectada en Lazo Cerrado.

Fuente: Autor

A continuación se realizan varias pruebas de simulación para el análisis del comportamiento de la planta sometida a diversos parámetros de sintonización del controlador MPC.

Prueba de simulación #1:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 1

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No

En la figura 3.29, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 1 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 100, 42, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido.

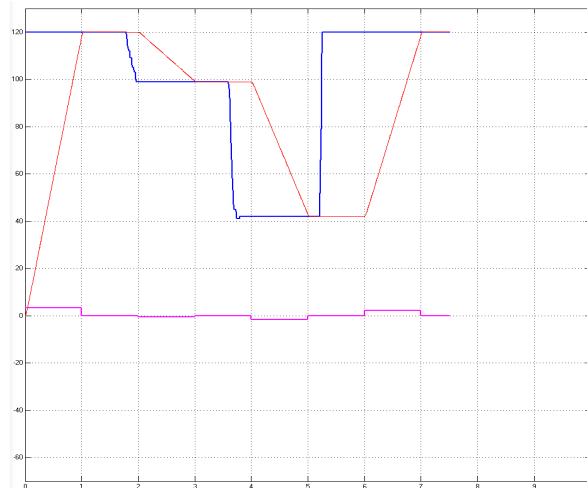


Figura 3.29: Comportamiento de la Planta, Simulación 1.

Fuente: Autor

Prueba de simulación #2:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 1

Duración del Intervalo de Control: 1

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No

En la figura 3.30, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con

la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 2 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 90, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido. Adicionalmente se presenta un pequeño overshoot en la gráfica de la respuesta del sistema.

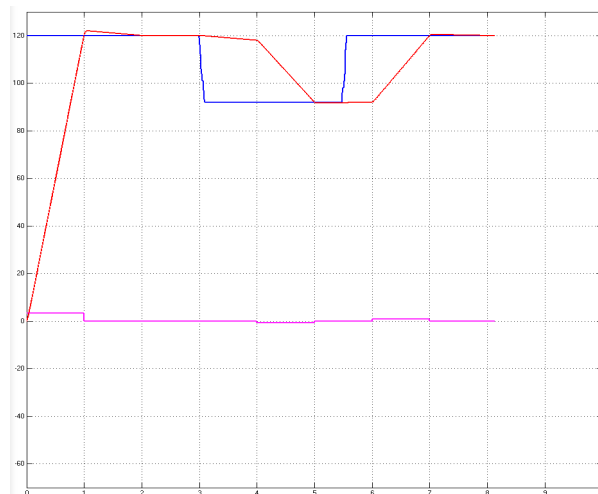


Figura 3.30: Comportamiento de la Planta, Simulación 2.

Fuente: Autor

Prueba de simulación #3:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.7

Duración del Intervalo de Control: 1

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No

En la figura 3.31, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 1.5 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 90, 55, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido. Adicionalmente se presenta un pequeño overshoot en la gráfica de la respuesta del sistema.

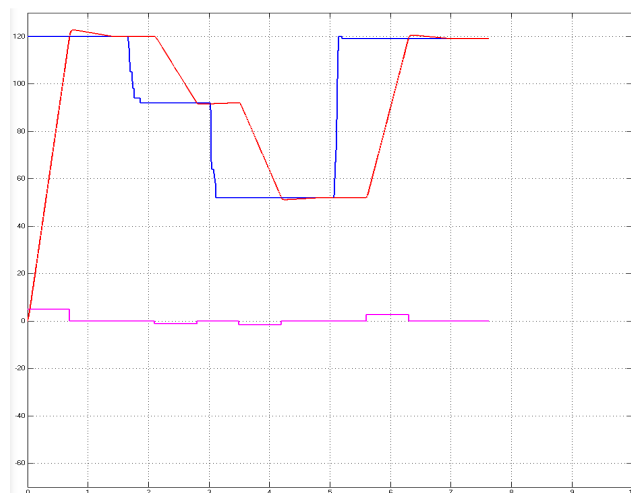


Figura 3. 31: Comportamiento de la Planta, Simulación 3.

Fuente: Autor

Prueba de simulación #4:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.4

Duración del Intervalo de Control: 1

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.32, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.8 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 95, 50, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido. Adicionalmente se presenta un pequeño overshoot en la gráfica de la respuesta del sistema.

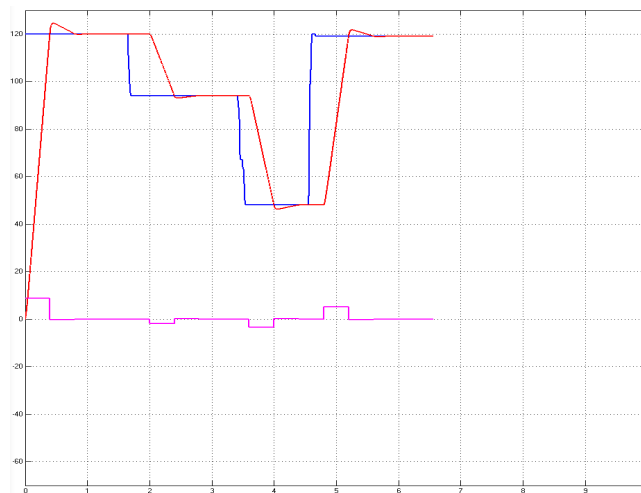


Figura 3.32: Comportamiento de la Planta, Simulación 4.

Fuente: Autor

Prueba de simulación #5:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.1, Duración del Intervalo de Control: 1, Horizonte de Predicción: 10, Lazo Cerrado: Si, Restricciones: No.

En la figura 3.33, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.4 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 90, 50, 65, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido. Adicionalmente se presenta un pequeño overshoot en la gráfica de la respuesta del sistema.

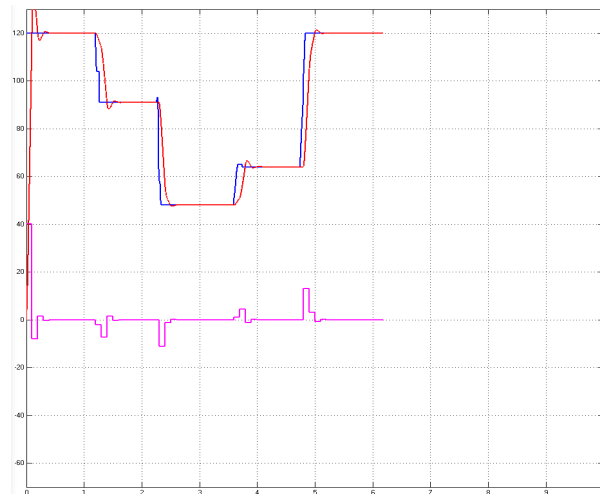


Figura 3.33: Comportamiento de la Planta, Simulación 5.

Fuente: Autor

Prueba de simulación #6:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.04

Duración del Intervalo de Control: 1

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.34, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.4 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 65, 35, 90, 120, 70), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido. Adicionalmente se presenta un pequeño overshoot en la gráfica de la respuesta del sistema.

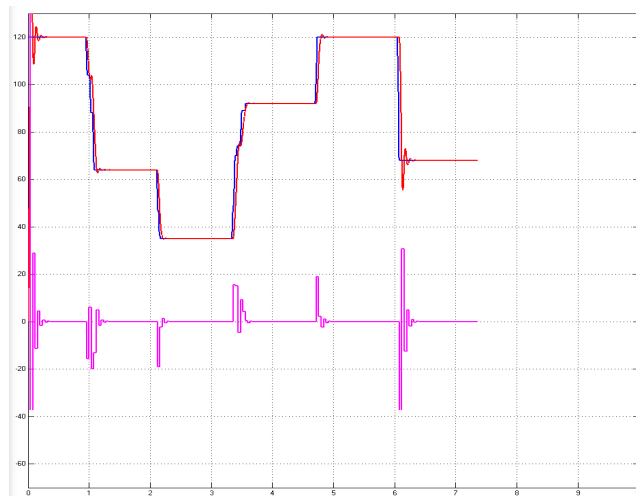


Figura 3.34: Comportamiento de la Planta, Simulación 6.

Fuente: Autor

Prueba de simulación #7:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.04

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.35, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.4 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 110, 70, 120, 60, 120, 50, 90, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido.

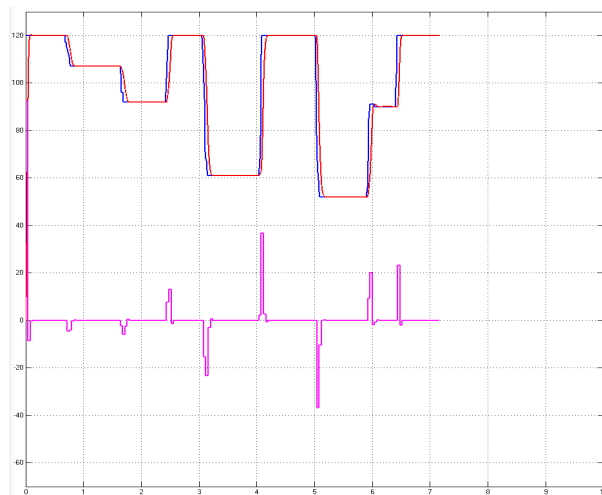


Figura 3.35: Comportamiento de la Planta, Simulación 7.

Fuente: Autor

Prueba de simulación #8:

Para el control de posicionamiento del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.02

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.36, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.4 seg, obedeciendo a los diferentes setpoints establecidos, como se observa en la gráfica de color azul (120, 70, 120, 32, 65, 120, 5, 29, 120), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido.

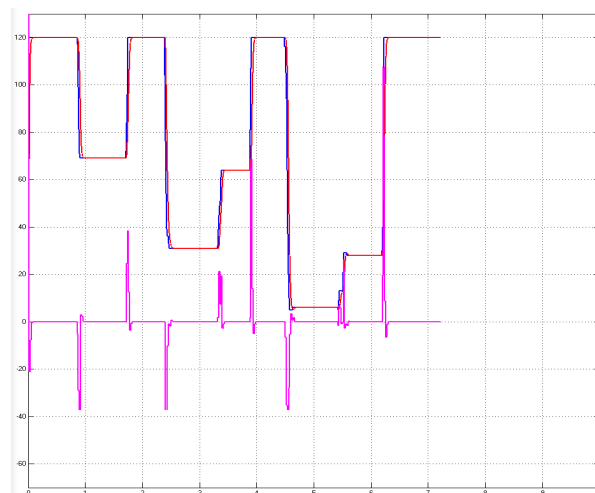


Figura 3.36: Comportamiento de la Planta, Simulación 8.

Fuente: Autor

Prueba de simulación con perturbaciones

En la figura 3.37, se observan dos ventanas, la superior en la cual se visualiza la gráfica de color rojo, representando la respuesta del sistema a los Setpoint ingresados, y en la gráfica inferior se aprecia la salida del controlador MPC, el cual está forzando, a que el sistema responda a valores establecidos.

En esta prueba la planta fue sometida a una perturbación de entrada, la cual fue compensada por el controlador MPC, para que el sistema no presente mayor variación y obedezca al Setpoint establecido.

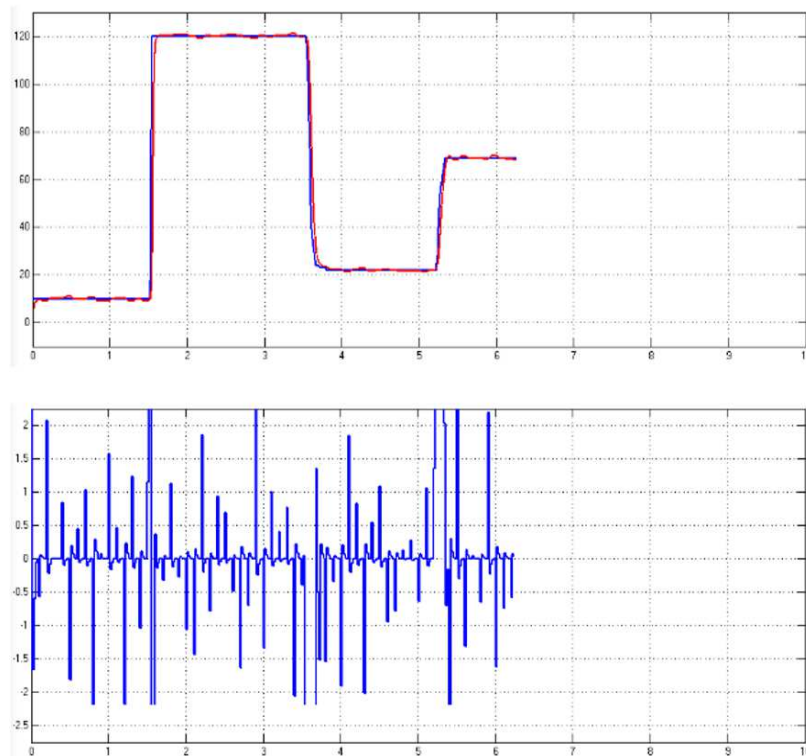


Figura 3.37: Comportamiento de la Planta, Caso con Perturbaciones.

Fuente: Autor

3.4.3. Pruebas de Simulación para el Control de Velocidad

En el siguiente bloque se detalla la simulación del modelo matemático para el control de velocidad del eje de un motor DC. A través de la plataforma de simulación Matlab primero se realizó una simulación de la planta en lazo abierto, ingresando la función de transferencia del modelo matemático previamente obtenido en la ecuación 3.4, la cual fue sometida a una señal de

entrada paso con una amplitud de 12V. Como resultado obtuvimos una señal con una amplitud de 1V la cual no obedece al Setpoint y posee un tiempo de estabilización de 2.5 seg, tal como se observa en la figura 3.38.

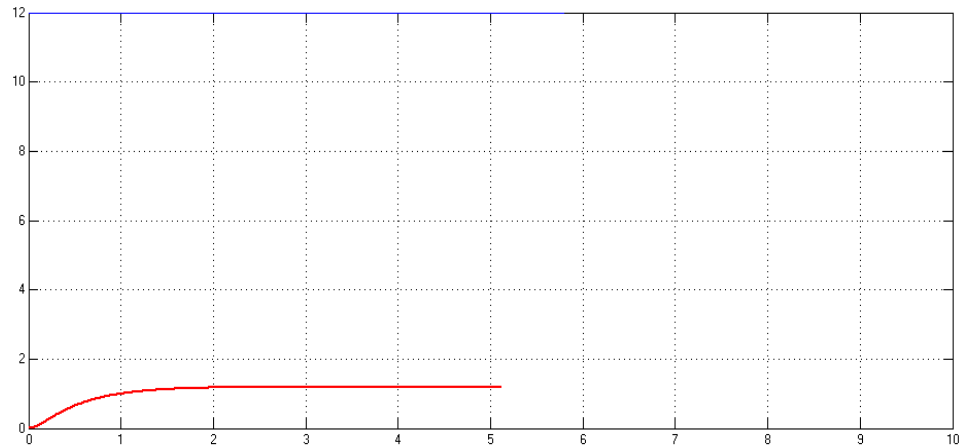


Figura 3.38: Respuesta de la planta conectada en Lazo Abierto.

Fuente: Autor

Luego se realizó la simulación de la planta en lazo cerrado con el modelo matemático para el control de velocidad, y fue sometida a una señal de entrada paso con una amplitud de 12V. Como resultado obtuvimos una señal con una amplitud de 1V la cual no obedece al Setpoint posee un tiempo de estabilización de 2 seg, tal como se observa en la figura 3.39.

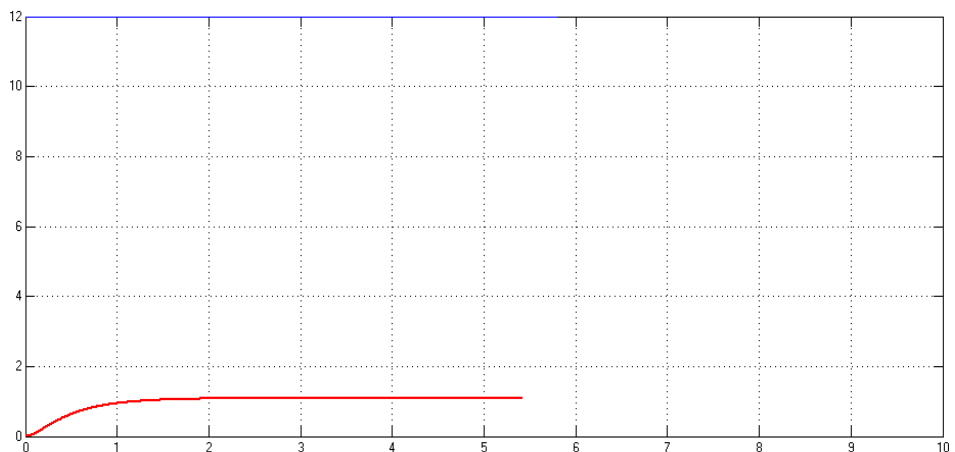


Figura 3.39: Respuesta de la planta conectada en lazo cerrado.

Fuente: Autor

A continuación se realizan varias pruebas de simulación para el análisis del comportamiento de la planta sometida a diversos parámetros de sintonización del controlador MPC.

Prueba de simulación #1:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 1

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.40, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 4.5 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

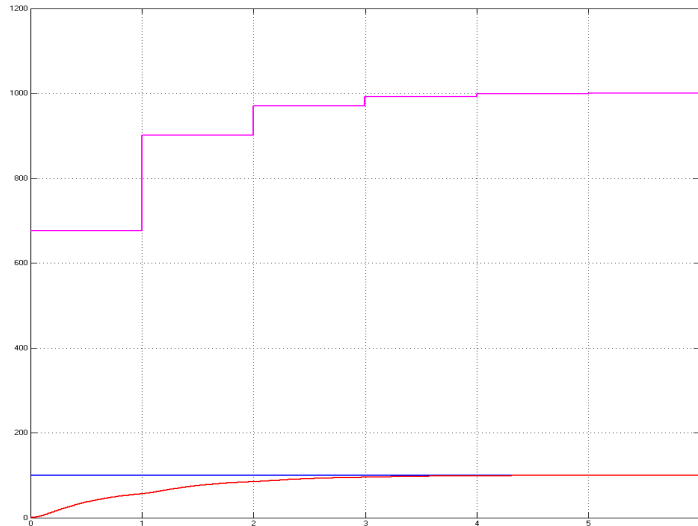


Figura 3.40: Comportamiento de la Planta, Simulación 1.

Fuente: Autor

Prueba de simulación #2:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.5

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.41, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 2.25 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el

controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

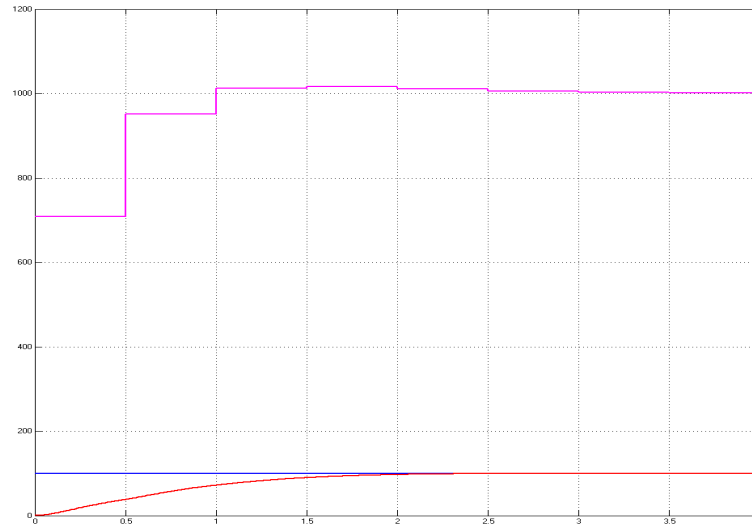


Figura 3.41: Comportamiento de la Planta, Simulación 2.

Fuente: Autor

Prueba de simulación #3:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.07

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.42, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.9 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

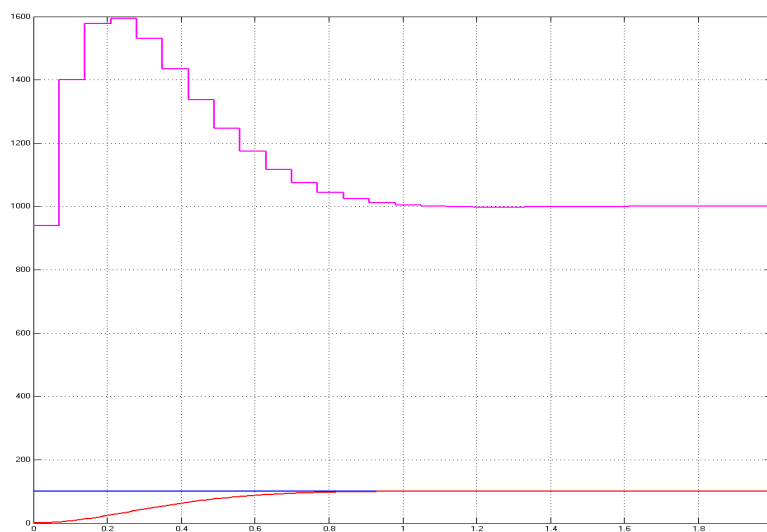


Figura 3.42: Comportamiento de la Planta, Simulación 3.

Fuente: Autor

Prueba de simulación #4:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.04

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.43, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.8 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

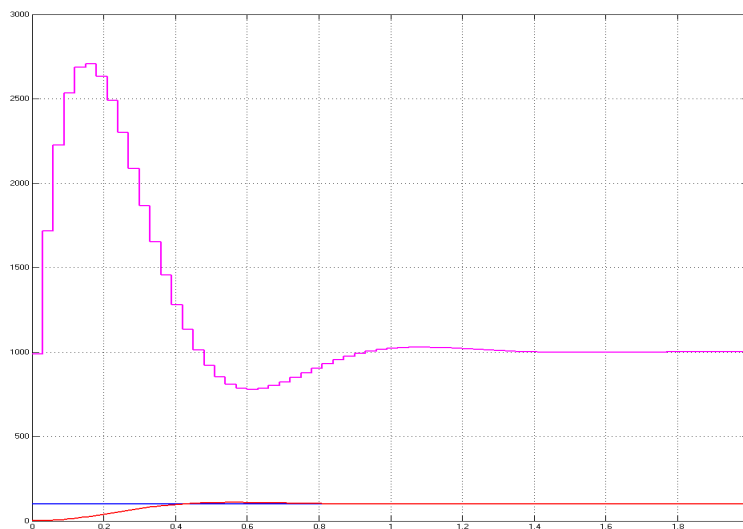


Figura 3.43: Comportamiento de la Planta, Simulación 4.

Fuente: Autor

Prueba de simulación #5:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.01

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.44, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 5 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul 100RPM, y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema. Adicionalmente se presenta oscilaciones en la gráfica de la respuesta del sistema.

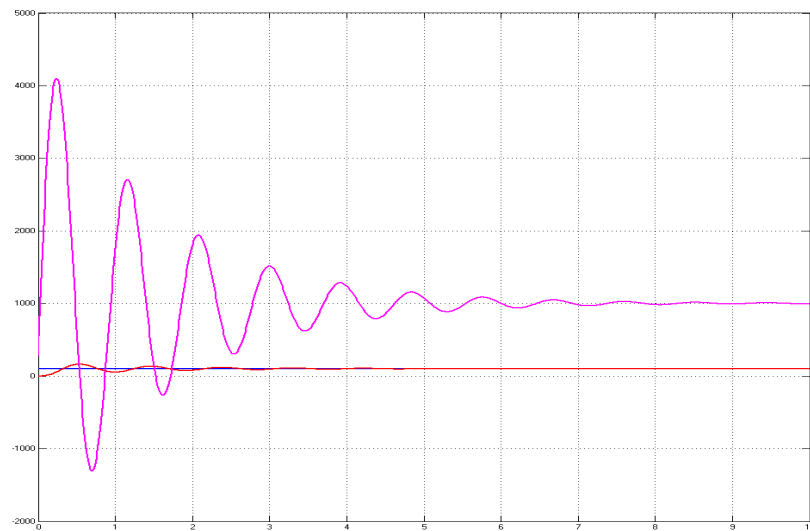


Figura 3.44: Comportamiento de la Planta, Simulación 5.

Fuente: Autor

Prueba de simulación #6:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.07

Duración del Intervalo de Control: 1

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.45, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 6.5 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

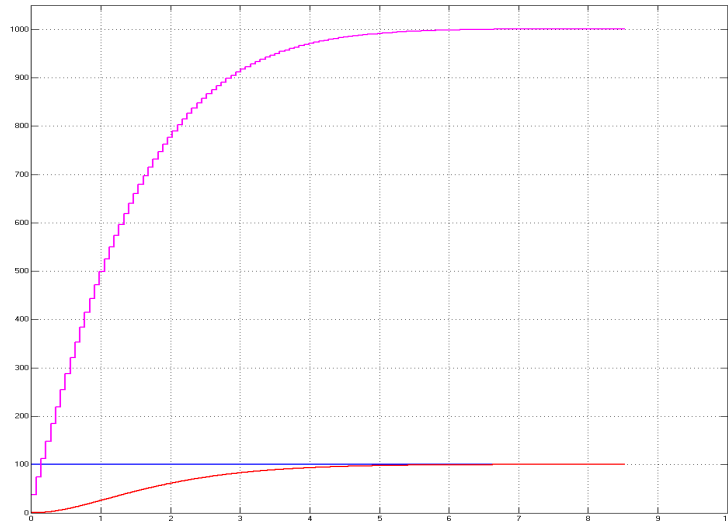


Figura 3.45: Comportamiento de la Planta, Simulación 6.

Fuente: Autor

Prueba de simulación #7:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.07

Duración del Intervalo de Control: 10

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.46, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 0.8 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el

controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

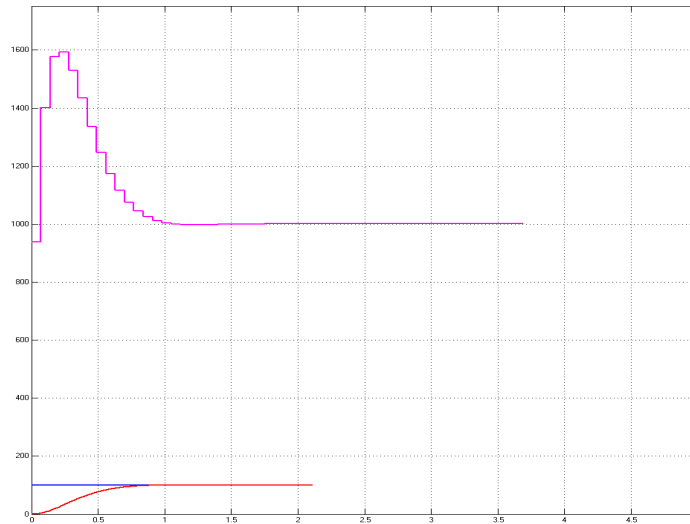


Figura 3.46: Comportamiento de la Planta, Simulación 7.

Fuente: Autor

Prueba de simulación #8:

Para el control de velocidad del eje del motor los parámetros de sintonización aplicados en el controlador MPC, se detallan a continuación:

Intervalo de Control: 0.07

Duración del Intervalo de Control: 20

Horizonte de Predicción: 10

Lazo Cerrado: Si

Restricciones: No.

En la figura 3.47, se observa el Setpoint representado con la gráfica de color azul, la señal de salida de la planta aplicando control MPC se representa con la gráfica de color rojo, y la señal de la salida del bloque MPC está representado con la gráfica de color magenta.

La respuesta de la planta a los parámetros de sintonía del controlador MPC, se puede apreciar a través del comportamiento de la gráfica de color rojo, la cual posee un tiempo de estabilización aproximadamente de 1.5 seg, obedeciendo a un Setpoint establecido, como se observa en la gráfica de color azul (100), y en la gráfica de color magenta se puede apreciar que el controlador MPC está forzando la salida para que la planta llegue hasta el Setpoint establecido, adicionalmente se puede acotar que los valores de salida del controlador MPC son mucho mayores a los valores de la salida del sistema.

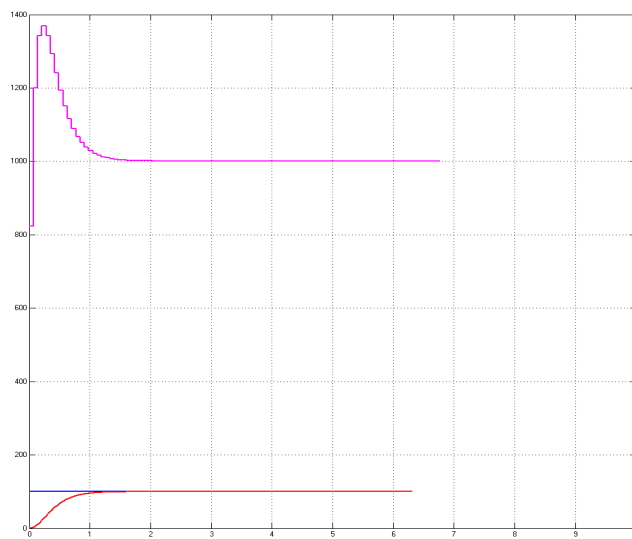


Figura 3.47: Comportamiento de la Planta, Simulación 8.

Fuente: Autor

Prueba de simulación con perturbaciones

En la figura 3.48, se observan dos ventanas, la superior en la cual se visualiza la gráfica de color rojo, representando la respuesta del sistema a los Setpoint ingresados, representados por la gráfica de color azul, y en la gráfica inferior de color magenta, se aprecia la salida del controlador MPC, el cual está forzando, a que el sistema responda a valores establecidos.

En esta prueba la planta fue sometida a una perturbación de entrada, la cual fue compensada por el controlador MPC, para que el sistema no presente mayor variación y obedezca al Setpoint establecido.

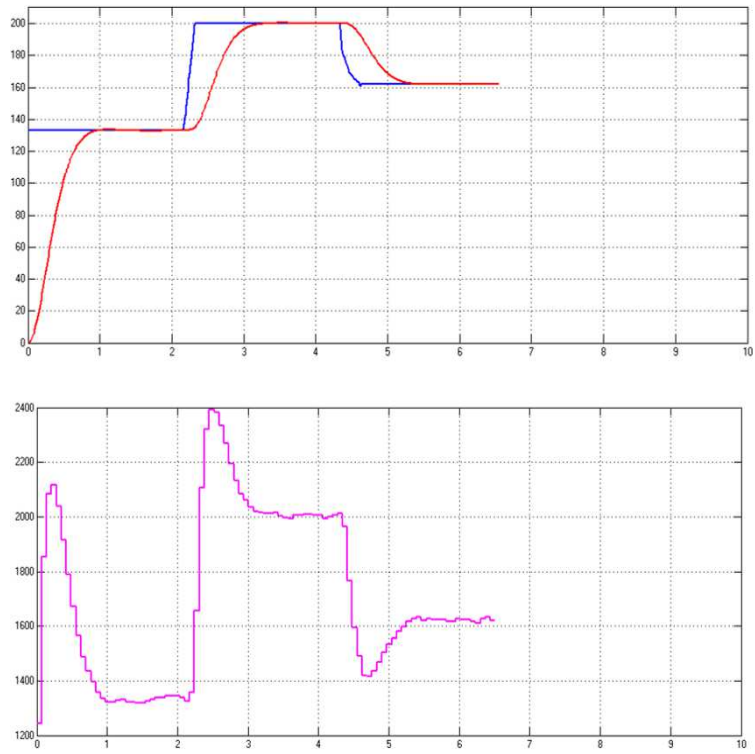


Figura 3.48: Comportamiento de la Planta, Caso con Perturbaciones.

Fuente: Autor

CAPÍTULO IV: DISCUSION DE RESULTADOS

El desarrollo del módulo de simulación fue expuesta en el capítulo anterior, describiendo todos los aspectos que forman parte del desarrollo del proyecto. En el presente capítulo se expondrá los resultados obtenidos del módulo de simulación y se realizara un análisis del mismo.

4.1. Resultados del Módulo de Simulación

Como primer punto se realizó el diseño del módulo de simulación, la plataforma de simulación Matlab, la plataforma de interfaz LabVIEW, el programa para crear puertos de comunicación virtual, lo que llevara hacer el análisis de cada uno de los aspectos, afirmando si se logró el objetivo inicial del proyecto.

4.1.1. Resultados de la Plataforma Matlab

En cuanto a la plataforma de simulación Matlab en el proyecto se puede establecer que es una herramienta que permite simular el comportamiento de una planta específica como se visualiza en la figura 4.1, a través del modelo matemático de la misma, cumpliendo con las expectativas de visualizar de manera gráfica el comportamiento del sistema.

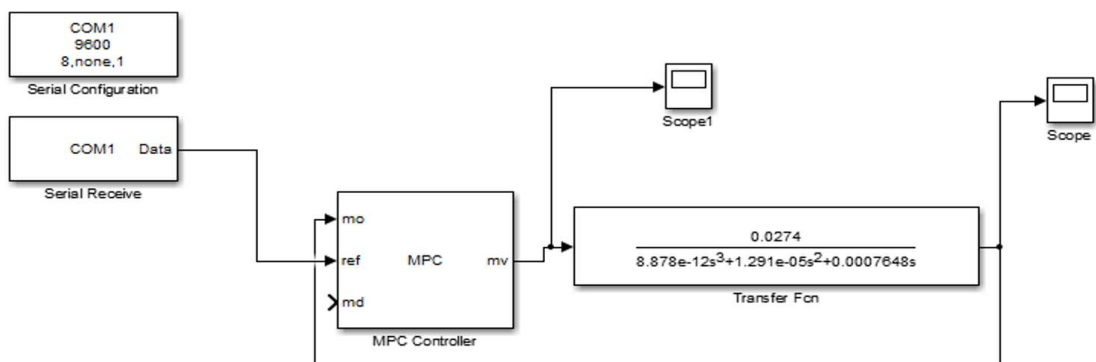


Figura 4.1: Configuración del MPC en el Entorno Matlab.

Fuente: Autor

4.1.2. Resultados de la Plataforma LabVIEW

En lo que respecta a la plataforma de simulación LabVIEW, el uso dado en el proyecto fue de gran ayuda, puesto que permitió realizar la interfaz de visualización y de control entre el sistema y el usuario a través de un computador como se observa en la figura 4.2 y 4.3, mostrando el comportamiento del sistema a través de graficas en función del tiempo. Posee la característica de mostrar un error si se desea ejecutar una prueba y la comunicación de los puertos virtuales no ha sido realizada previamente, esto es una característica importante puesto que nos permite saber que error puede estar pasando en el sistema.

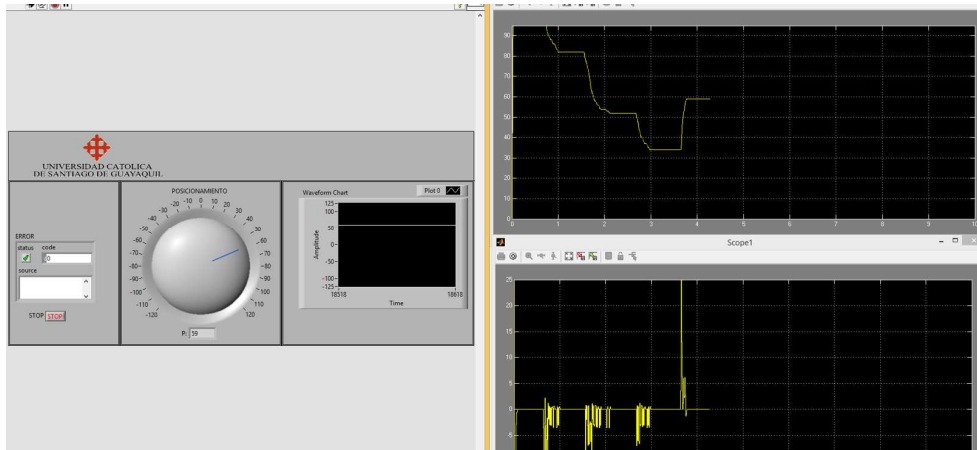


Figura 4.2: Interfaz de Sistema para el Control de Posicionamiento.

Fuente: Autor

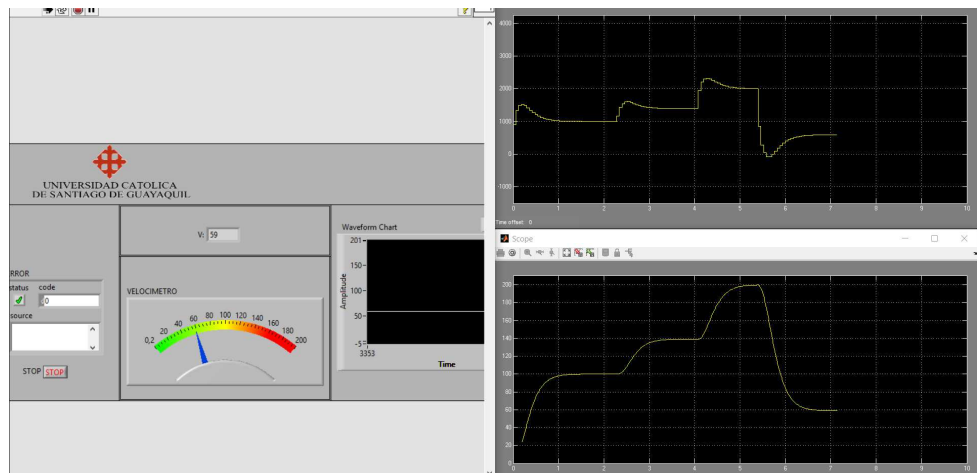


Figura 4.3: Interfaz de Sistema para el Control de Velocidad.

Fuente: Autor

4.1.3. Resultados de Software Virtual Serial Port

A través de este Software, como se representa en la figura 4.4, nos permite crear puertos virtuales, los cuales nos facilitó la comunicación entre las plataformas Matlab y LabVIEW. Esta herramienta es fácil e interactiva de utilizar, además presenta como característica que la creación de los puertos virtuales se realiza una sola vez y la configuración queda almacenada hasta que se desee modificar los puertos para otros fines.

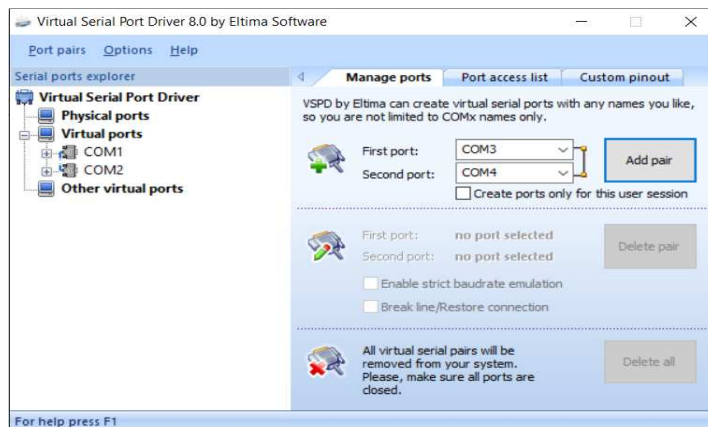


Figura 4.4: Pantalla Principal del Virtual Serial Port Driver.

Fuente: Autor

4.2 Discusión de Resultados

Este proyecto fue muy satisfactorio, puesto que cumplió con las expectativas, planteadas en forma de objetivos al inicio de este trabajo de investigación:

- Se cumplió con el objetivo planteado respecto a la visualización del comportamiento del sistema de una forma gráfica, tanto para una planta sin controlador, como para una planta aplicando un control MPC.
- Se realizó el control del sistema desde una interfaz HMI, en la cual se puede variar constantemente el Setpoint del sistema, para ver la respuesta que presenta.
- Se efectuó la configuración del controlador MPC a través de una plataforma de simulación, la cual nos permitió ingresar los parámetros de sintonía del controlador MPC.

- Se logró ingresar los datos pertenecientes tanto a una planta con las características del control de velocidad como con las características para el control de posicionamiento de un motor DC, basado en datos Reales.
- Se generó la comunicación entre ambas plataformas de simulación, a través del Software Virtual Serial Port Driver, el cual permitió crear puertos de comunicación virtuales para el envío y recepción de datos.

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

En este punto se analiza el resultado del trabajo de investigación, recopilando toda la información obtenida durante el proceso de desarrollo del proyecto, analizando los pros y contras para dictaminar una conclusión y una recomendación final del trabajo de investigación.

5.1. Conclusiones

Concluido el trabajo de investigación realizado, se tomó en cuenta toda la información obtenida durante el proceso de desarrollo del proyecto, y se llegó a las siguientes conclusiones:

- Por medio de plataformas virtuales se puede simular el comportamiento de un sistema en partículas, de tal forma que se pueda visualizar el comportamiento de la planta con y sin controlador.
- Los modelos matemáticos deben ser muy parecidos al real, para que el controlador tome en cuenta todas las características que encierra el modelo de la planta, y así obtener resultados con mayor precisión.
- En las técnicas de control predictivo, las restricciones, son una de las características que diferencian esta técnica de control de otras.
- Las técnicas de control predictivo pueden ser implementadas tanto en sistemas lineales, como no lineales.
- Las técnicas de control predictivo pueden ser implementadas tanto en sistemas SISO como en sistemas MIMO.
- La implementación de técnicas de control predictivo, demanda una capacidad de procesamiento de datos muy alta ya que constantemente está calculando en cada periodo de muestreo.

5.2. Recomendaciones

Durante el desarrollo del trabajo de investigación se analizó el diseño, funcionamiento, modelado matemático y simulación del proyecto realizado, que se describe a continuación:

- Realizar el modelado matemático en base a características reales de la planta, para obtener un modelo más real.
- Se debe tomar en cuenta la capacidad del procesador, espacio en el disco duro, espacio en la memoria RAM del ordenador, ya que estas plataformas de simulación conllevan una gran capacidad de almacenamiento y de procesamiento, evitando tener inconvenientes al momento de la simulación.
- Al momento de realizar la instalación de la plataforma de simulación Matlab, se debe tomar en cuenta los paquetes de instalación de la plataforma Simulink.
- Antes de generar la simulación, se recomienda revisar los puertos virtuales de comunicación entre ambos programas, caso contrario no se podrá efectuar la simulación.
- Antes de realizar la simulación, se debe revisar que todos los parámetros hayan sido registrados.

5.3. Trabajo Futuro

Una vez efectuadas las pruebas de funcionamiento del módulo de simulación para el control de posicionamiento y de velocidad de un motor DC, se llegó a la conclusión de que el módulo de simulación cumple con todos los objetivos planteados al inicio del trabajo de investigación, considerando de que en futuras investigaciones, se pueda ampliar más el tema, llevando el control predictivo al desarrollo en pruebas físicas de algún tipo de planta. Desarrollando pruebas de simulación mediante la comunicación entre dos ordenadores, simulando al proceso y a la interfaz HMI.

Este proyecto se enfoca en técnicas de control moderna basada en control predictivo, por lo tanto se puede extender un poco más el tema de técnicas de control moderno, aplicando otro tipo de controlador y generar nuevas investigación y aplicación.

REFERENCIAS BIBLIOGRÁFICAS

- Adriano Macas, K., & García Míguez, A. (2014). *Escuela Superior Politécnica de Chimborazo*. Obtenido de DSpace ESPOCH:
<http://dspace.esPOCH.edu.ec/bitstream/123456789/3376/1/25T00233.pdf>
- Aguirre Zapata, D. (2013). *Universidad de Piura*. Obtenido de Repositorio Institucional PIRHUA: <https://pirhua.udep.edu.pe/handle/11042/1739>
- Alejandro, H. M. (Junio de 2016). *RiuNet Repositorio Institucional UPV Universidad Politecnica de Valencia*. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/68512/Hern%C3%A1ndez%20-%20CONTROL%20PREDICTIVO%20BASADO%20EN%20ESCENARIOS%20PARA%20SISTEMAS%20LINEALES%20CON%20SALTOS%20MARKOVIANOS.pdf?sequence=1>
- Andrade Moreira, M., & Cortez Bohórquez, A. (2012). *Universidad Católica de Santiago de Guayaquil*. Obtenido de Repositorio Digital - UCSG: <http://repositorio.ucsg.edu.ec/bitstream/3317/230/3/T-UCSG-PRE-TEC-ITEL-2.pdf>
- AulaFacil. (2017). *AulaFacil*. Obtenido de Curso de Electromagnetismo: <http://www.aulafacil.com/cursos/l30072/bachillerato/fisica/electromagnetismo/partes-esenciales-de-un-motor-y-sus-nombres>
- Barria, F. (2012). *U-Cursos*. Obtenido de Generación de Energía Eléctrica con Fuentes Renvables: https://www.u-cursos.cl/ingenieria/2012/1/EL6000/1/material_docente/
- Bordóns Alba, C. (2000). *Universidad de Sevilla*. Obtenido de <https://control-ps2316-sept2009.wikispaces.com/file/view/CONTROL+PREDICTIVO.pdf>
- Camacho Córdova, G. (2015). *SCRIBD*. Obtenido de <https://es.scribd.com/document/265236634/Sistema-Scada1>
- Cevallos Ulloa, H. (2010). *Escuela Superior Politecnica del Litoral*. Obtenido de Repositorio - ESPOL: <https://www.dspace.espol.edu.ec/bitstream/123456789/13519/7/D-42648.pdf>
- Crespo, E. (2017). *Aprendiendo Arduino*. Obtenido de <https://aprendiendoarduino.wordpress.com/tag/avr/>
- Días Dorado, A. (2015). *Depósito de Investigación Universidad de Sevilla*. Obtenido de <https://idus.us.es/xmlui/handle/11441/38569>
- Dimension Engineering. (2012). *Dimension Engineering*. Obtenido de <https://www.dimensionengineering.com/datasheets/Sabertooth2x12.pdf>
- Eltima Software. (2017). *Eltima Virtual Serial Port*. Obtenido de <https://www.eltima.com/es/products/vspdxp/>
- Escobar, R. (2016). *Blogger*. Obtenido de Modelo Lineal de un Motor DC: <http://stg-pepper.blogspot.com/2016/04/modelo-simple-de-un-motor-dc.html>
- Ferramosca, A., Gruber, J., Limon, D., & Camacho, E. (2013). *ScienceDirect*. Obtenido de <http://www.sciencedirect.com/science/article/pii/S1697791212000908>
- Flor Unda, O. (12 de Septiembre de 2013). *Repositorio SENESCYT*. Obtenido de <http://repositorio.educacionsuperior.gob.ec/bitstream/28000/1486/1/T-SENESCYT-00620.pdf>
- Flores, D. (2017). *Electricalia*. Obtenido de Aparatos Eléctricos: <https://sites.google.com/site/electricalia/0202perfil>
- Flores, L. (2013). *Blogger*. Obtenido de Máquinas CD: <http://maquinascdlauraf.blogspot.com/2013/05/maquinas-de-cd.html>
- Gallego Len, A. J. (2014). *Depósito de Investigación Universidad de Sevilla*. Obtenido de

- https://idus.us.es/xmlui/bitstream/handle/11441/52410/D_T.PROV58.pdf?sequence=1&isAllowed=y
- Garrido Arévalo, B., Yarmuch Muñoz, J., & Ogden Araya, T. (2011). *U-Cursos*. Obtenido de Plataforma de Apoyo a la Docencia Presencial: https://www.u-cursos.cl/usuario/834c0e46b93fd72fd8408c492af56f8d/mi_blog/r/Infor_Micro_controlador.pdf
- Gómez Figueroa, S., Medina Jiménez, J., Muñoz Sánchez, D., Núñez Villaluz, P., & Pérez González, J. (2010). *Repositorio digital de la Facultad de Ingeniería*. Obtenido de UNAM: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/1246/1/Tesis.pdf>
- Gonzales Lecaros, S. N. (2014). *Repositorio Digital de Tesis PUCP*. Obtenido de <http://tesis.pucp.edu.pe/repositorio/handle/123456789/5355>
- Gonzales, G., & Bosso, J. (2012). *Universidad Nacional de San Luis*. Obtenido de Facultade de Ingeniería y Cicencias Agropecuarias: <http://www.fica.unsl.edu.ar/~lca/files/Control-de-Velocidad-y-Posicion-a-Lazo-Cerrado-de-un-Motor-de-CC-con-dsPIC30F4011.pdf>
- González Querubín, E. (2011). *SlideShare*. Obtenido de Universidad Politecnica de Valencia: <https://es.slideshare.net/edalgonzalez/algoritmos-de-control-predictivo-multivariable-para-procesos-con-dinmica-rpida-aplcacin-al-control-de-un-sistema-de-motores-acoplados>
- González, A. I., & Zambrano, J. C. (Junio de 2013). *Ingenius Revista de Ciencia y Tecnología*. Obtenido de <http://revistas.ups.edu.ec/index.php/ingenius/article/view/360>
- Hernández Cevallos, M., & Ledesma Marcalla, D. (2010). *Escuela Superior Politécnica de Chimborazo*. Obtenido de Repositorio ESPOCH: <http://dspace.esPOCH.edu.ec/bitstream/123456789/1137/1/25T00140.pdf>
- Horn, L. (2012). *GISMODO*. Obtenido de https://www.google.com.ec/search?q=origen+del+criterio+de+control&tbm=sch&tbs=rimg:Cd23YcBQelUCljgTuUQ74thrD3eVuPej4-MX3gWG4SF3PjynHUbvUIFQ3sfKc4Hf7RMQUATyy9_1f8S1UJm1-ZhLEACoSCRO5RDvi2GsPEenRN_1hcl-4hKhIjd5W496Pj4xcRgBaPct3v3iQqEgneBYbhIXc-PBGpQI1w4XI38
- IEEE - UTP. (2013). *IEEE*. Obtenido de IEEE Universidad Tecnológica de Panamá: <http://sites.ieee.org/sb-utp/tutorials/ti-launchpad/>
- Javier, O. G. (Octubre de 2011). *Universidad Politecnica Nacional*. Obtenido de <http://bibdigital.epn.edu.ec/handle/15000/4272>
- Kanchwala, H., & Bordons, C. (2015). *ResearchGate*. Obtenido de Publicaciones: https://www.researchgate.net/figure/282268916_fig3_Figure-3-Model-Predictive-Control-strategy-and-block-diagram-representation
- Llanos, J., Pilatasig, M., Curay, D., & Vaca, A. (2016). *ResearchGate*. Obtenido de Diseño e Implementacion de un Modelo de Control Predictivo: https://www.researchgate.net/publication/311611421_Design_and_implementation_of_a_model_predictive_control_for_a_pressure_control_plant
- Lopera, E., & Mejia, D. (Octubre de 2010). *Portal de Revistas Academicas del ITM*. Obtenido de https://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ad=rja&uact=8&ved=0ahUKEwjt1rbEx6rUAhXLQCYKHTrBCAsQFgggMAA&url=http%3A%2F%2Fitmojs.itm.edu.co%2Findex.php%2Ftecnologicas%2Farticle%2Fdownload%2F195%2F196&usq=AFQjCNHZGri2PRf1JZ4FUGkX_B1bd
- López Caisaguano, V. G., & Paredes Anchatipán, A. D. (Febrero de 2015). *Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE*. Obtenido de <http://repositorio.espe.edu.ec/handle/21000/9325>

- Lusson Cervantes, A., Beltran Aguedo, R., & Bausa Ortiz, I. (2010). *Sistema de Informacion Cientifica Redalyc*. Obtenido de <http://www.redalyc.org/articulo.oa?id=181317867007>
- MathWorks. (2017). *MathWorks Support*. Obtenido de <https://es.mathworks.com/support/sysreq.html>
- Medina Cervantes, J. (2012). *Universidad Veracruzana*. Obtenido de <https://www.uv.mx/personal/jemedina/files/2010/07/Manual-del-Taller-Introduccion-al-Microcontrolador-PIC18F4550.pdf>
- Meza Weber, F., & Ramos Morales, P. (2015). *Universidad Politécnica Salesiana*. Obtenido de Repositorio Digital UPS: <http://dspace.ups.edu.ec/bitstream/123456789/10257/1/UPS-GT001352.pdf>
- Moromenacho, E., Rosero, C., & Bravo, Y. (2009). *ResearchGate*. Obtenido de Control Predictivo: https://www.researchgate.net/publication/43188842_Control_predictivo_en_tanques_acoplados
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361J-01/lvinstio/visa_configure_serial_port/
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361H-01/lvinstio/visa_write/
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361J-01/lvinstio/visa_open/
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361J-01/lvinstio/visa_close/
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361H-01/glang/wait_till_next_ms_multiple/
- National Instruments. (2017). *NATIONAL INSTRUMENTS*. Obtenido de <http://forums.ni.com/t5/Certified-LabVIEW-Architects/ct-p/7003>
- National Instruments. (2017). *NI Development Systems*. Obtenido de <http://www.ni.com/white-paper/53740/en/>
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361J-01/glang/build_array/
- National Instruments. (2012). *NI Support*. Obtenido de http://zone.ni.com/reference/en-XX/help/371361J-01/glang/byte_array_to_string/
- Oliden Semino, J. C. (Mayo de 2016). *Repositorio del CONCYTEC*. Obtenido de <http://repositorio.concytec.gob.pe/handle/CONCYTEC/225>
- Open automation. (2008). *Open automation*. Obtenido de <http://www.openautomation.de/detailseite/40-jahre-sps.html>
- Ortiz Guevara, M. (2011). *Escuela Politécnica Nacional*. Obtenido de Repositorio Digital: <http://bibdigital.epn.edu.ec/handle/15000/4272>
- Perez Losada, D. (Septiembre de 2014). *Universidad Politecnica de Valencia*. Obtenido de http://www.ja2014.upv.es/wp-content/uploads/papers/paper_25.pdf
- Pernía, M. (2011). *ResearchGate*. Obtenido de Marino Pernía: https://www.researchgate.net/profile/Marino_Pernia/publication/235752021_Conceptos_Basicos_de_Maquinas_de_corriente_continua/links/0912f5131e8e23bfa1000000/Conceptos-Basicos-de-Maquinas-de-corriente-continua.pdf
- Pololu. (2017). *Pololu*. Obtenido de <https://www.pololu.com/product/2824>
- Ponce Arias, C. (2014). *Centro de Computo*. Obtenido de CEC - Universidad de Chile: <http://www.cec.uchile.cl/~dsaez/wp-content/uploads/2017/03/Tesis-CPonce.pdf>

- Ramírez Pérez, K. (2011). *Universidad Nacional de Colombia*. Obtenido de Bdigital Repositorio Institucional:
http://www.bdigital.unal.edu.co/5436/1/1017123749.2011_1.pdf
- Rivadeneira, D. (2012). *Blogger*. Obtenido de <http://dianis-eliz.blogspot.com/2012/09/arquitectura-de-von-neumann-y-harvard.html>
- Robot Gear. (2016). *Robot Gear*. Obtenido de <https://www.robotgear.com.au/Product.aspx/Details/4550-100-1-Metal-Gearmotor-37Dx73L-mm-with-64-CPR-Encoder>
- Rodríguez Penin, A. (2012). *Sistemas SCADA 3ra edición*. Mexico DF: Marcombo, S.A.
- Salveti, D. (2012). *Blogger*. Obtenido de Electrónica Industrial:
<http://eet602ei.blogspot.com/2012/05/sistemas-de-control-lazo-abiertocerrado.html>
- Sánchez, S. (2013). *Microcontroladores*. Obtenido de Los microcontroladores de hoy en día: <https://microcontroladoresesv.wordpress.com/los-microcontroladores-de-hoy-en-dia/>
- Sánchez, S. (2013). *Microcontroladores*. Obtenido de Introducción y Arquitectura de microcontroladores:
<https://microcontroladoresesv.wordpress.com/arquitectura-de-los-microcontroladores/>
- Texas Instruments. (2016). *Texas Instruments*. Obtenido de Microcontroladores:
<http://www.ti.com/tool/msp-exp430g2#0>
- Tosini, M., Marone, J., & Goñi, E. (2012). *Facultad de Ciencias Exactas Universidad Nacional del Centro de la Provincia de Buenos Aires*. Obtenido de Investigación en Computación Aplicada:
<http://www.exa.unicen.edu.ar/catedras/tmicrocon/>
- Ugarte, G. A. (2012). *Blogger*. Obtenido de <http://arquiuct.blogspot.com/2012/12/diferencia-entre-un-micro-procesador-y.html>
- Verne, S., Gonzales, S., & Valla, M. I. (7 de Agosto de 2014). *IEEE*. Obtenido de <http://ieeexplore.ieee.org/abstract/document/6872900/>



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

DECLARACIÓN Y AUTORIZACIÓN

Yo, **Domínguez Reyes, Alex Francisco** con C.C: # 0931069504 autor del Trabajo de Titulación: **Diseño de un Módulo de Simulación para el Control de Posicionamiento y de Velocidad de un Motor DC Utilizando Control Predictivo, con Supervisión a través de un Sistema SCADA** previo a la obtención del título de **INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 13 de Septiembre del 2017

f. _____

Nombre: Domínguez Reyes, Alex Francisco

C.C: 0931069504

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA		
FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN		
TÍTULO Y SUBTÍTULO:	Diseño de un módulo de simulación para el control de posicionamiento y de velocidad de un motor dc utilizando control predictivo, con supervisión a través de un sistema SCADA.	
AUTOR(ES)	Alex Francisco Domínguez Reyes	
REVISOR(ES)/TUTOR(ES)	M. Sc. Eduardo Vicente Mendoza Merchán	
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil	
FACULTAD:	Facultad de Educación Técnica para el Desarrollo	
CARRERA:	Ingeniería Electrónica en Control y Automatismo	
TITULO OBTENIDO:	Ingeniero Electrónico en Control y Automatismo	
FECHA DE PUBLICACIÓN:	13 de septiembre del 2017	No. DE PÁGINAS: 125
ÁREAS TEMÁTICAS:	SIMULACIÓN, AUTOMATIZACIÓN	
PALABRAS CLAVES/ KEYWORDS:	MPC; SETPOINT; MATLAB; LABVIEW; TIEMPO DE RESPUESTA; HMI.	
RESUMEN/ABSTRACT (150-250 palabras):		
<p>El presente trabajo de titulación se fundamenta en el control predictivo basado en modelo, aplicado al control de posicionamiento y de velocidad de un motor DC, a través de un módulo de simulación. El módulo de simulación esta implementado en base a las plataformas de simulación Matlab y LabVIEW, las cuales brindan las herramientas necesarias para simular el control MPC tanto para el control de posicionamiento como el control de velocidad. El módulo de simulación posee una interfaz HMI a través de un computador, implementado en la plataforma de simulación LabVIEW. El siguiente trabajo de investigación se fundamenta tanto en la metodología bibliográfica como en la metodología experimental, mediante las cual encierra los fundamentos teóricos del control predictivo basado en modelo, aplicado a la función de transferencia de una planta basada en datos reales y el funcionamiento del sistema mediante una plataforma de simulación, visualizando y controlado el proceso a través de una interfaz HMI virtual.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO
CONTACTO CON AUTOR/ES:	Teléfono: +593-992610089	E-mail: al.jr94@hotmail.com
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Mendoza Merchán Eduardo Vicente	
	Teléfono: +593-985086815	
	E-mail: eduardo.mendoza@cu.ucsg.edu.ec	
SECCIÓN PARA USO DE BIBLIOTECA		
Nº. DE REGISTRO (en base a datos):		
Nº. DE CLASIFICACIÓN:		
DIRECCIÓN URL (tesis en la web):		