



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA E  
INSTRUMENTACIÓN

PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO EN ELECTRÓNICA E  
INSTRUMENTACIÓN

AUTORES: VACA GUILCAMAIGUA, ANDREA ELIZABETH  
CURAY FLORES, SEGUNDO DAVID

TEMA: DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO DE  
CONTROL AVANZADO APLICADO A UN PROCESO DE  
PRESIÓN, UTILIZANDO UN CONTROLADOR DE  
AUTOMATIZACIÓN PROGRAMABLE PARA EL LABORATORIO  
DE REDES INDUSTRIALES Y CONTROL DE PROCESOS DE LA  
UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN  
LATACUNGA

DIRECTORA: ING. LLANOS, JACQUELINE.  
CODIRECTOR: ING. PILATASIG, MARCO.

LATACUNGA, ENERO 2015

UNIVERSIDAD DE LA FUERZAS ARMADAS - ESPE  
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

**CERTIFICADO**

Ing. Llanos, Jacqueline, (DIRECTORA)

Ing. Pilatasig, Marco, (CODIRECTOR)

**CERTIFICAN**

Que el trabajo titulado “*DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO DE CONTROL AVANZADO APLICADO A UNA PLANTA DE PRESIÓN, UTILIZANDO UN CONTROLADOR DE AUTOMATIZACIÓN PROGRAMABLE PARA EL LABORATORIO DE REDES INDUSTRIALES Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA*” realizado por Segundo David Curay Flores y Andrea Elizabeth Vaca Guilcamaigua, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Universidad de las Fuerzas Armadas - ESPE.

Debido a que constituye un trabajo de alto contenido de investigación y que ayuda a la formación profesional y aplicación de conocimientos, **SI** recomiendan su publicación.

El mencionado trabajo consta de dos documentos empastados y dos discos compactos el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan a Segundo David Curay Flores y Andrea Elizabeth Vaca Guilcamaigua que lo entregue al Ing. Franklin Silva, en su calidad de Director de la Carrera.

Latacunga, Enero del 2015.

---

Ing. Llanos, Jacqueline  
DIRECTORA

---

Ing. Pilatasig, Marco  
CODIRECTOR

UNIVERSIDAD DE LA FUERZAS ARMADAS - ESPE  
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

**DECLARACIÓN DE RESPONSABILIDAD**

CURAY FLORES, SEGUNDO DAVID  
VACA GUILCAMAIGUA, ANDREA ELIZABETH

DECLARAMOS QUE:

El proyecto de grado denominado “DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO DE CONTROL AVANZADO APLICADO A UNA PLANTA DE PRESIÓN, UTILIZANDO UN CONTROLADOR DE AUTOMATIZACIÓN PROGRAMABLE PARA EL LABORATORIO DE REDES INDUSTRIALES Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA”, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan el pie de las páginas correspondiente, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

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

Latacunga, Enero del 2015

---

*Curay Flores, Segundo David*  
C.C. 050336004-2

---

*Vaca Guilcamaigua, Andrea Elizabeth*  
C.C. 050292319-6

UNIVERSIDAD DE LA FUERZAS ARMADAS - ESPE  
INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

**AUTORIZACIÓN**

Nosotros, CURAY FLORES, SEGUNDO DAVID.  
VACA GUILCAMAIGUA, ANDREA ELIZABETH.

Autorizamos a la Universidad de las Fuerzas Armadas - ESPE la publicación, en la biblioteca virtual de la Institución del trabajo “DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO DE CONTROL AVANZADO APLICADO A UNA PLANTA DE PRESIÓN, UTILIZANDO UN CONTROLADOR DE AUTOMATIZACIÓN PROGRAMABLE PARA EL LABORATORIO DE REDES INDUSTRIALES Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA”, cuyo contenido, ideas y criterios son de nuestra exclusividad responsabilidad y autoría.

Latacunga. Enero del 2015

---

*Curay Flores, Segundo David*  
C.C. 050336004-2

---

*Vaca Guilcamaigua, Andrea Elizabeth*  
C.C. 050292319-6

## DEDICATORIA

Este trabajo va dedicado a Dios, quién siempre guía mis pasos y me protege para siempre seguir adelante, a mis padres “Salvador y Blanca” que son el ejemplo de esfuerzo, perseverancia e inculcaron en mí el valor de la honradez y el respeto otorgándome la mejor herencia de esta vida que es el estudio, a mis hermanos “Diego y Edgar” y a mi hermanita querida “Maribel” a los cuales quiero mucho y supieron brindarme siempre su apoyo en las buenos y malos momentos, a mi tío “Fabián” que es ejemplo de superación y tenacidad ante pruebas que le dio la vida, por esta razón este trabajo dedico a toda mi familia que supo apoyarme en todos los años de estudio hasta alcanzar mi profesión anhelada.

***David.***

Dedico este proyecto primeramente a Dios, quién me ha brindado la fortaleza para seguir adelante fortaleciendo mi corazón, iluminando mi mente y por haber puesto en mi camino a personas maravillosas que han sido mi apoyo, mi soporte y mi compañía durante toda mi vida académica, a mis padres “Carlos y Susana” que siempre han estado junto a mí, protegiéndome en todo momento, y brindándome su apoyo incondicional, recordándome lo importante que es la humildad y la dedicación para conseguir nuestros objetivos, a mis hermanos “Carlos, Adrian y Paola” quiénes me han sabido acompañar en los buenos y malos momentos, a mis abuelitos “Guillermo, Segundo y Blanca” que desde el cielo me cubren con su manto de amor y sobre todo a mi abuelita “Rosita” que sigue junto a mi dándome su amor y su apoyo incondicional como aquel primer día que empecé mi formación profesional.

***Andrea.***

## AGRADECIMIENTO

A Dios, por darme una familia maravillosa que siempre estuvo a mi lado y supieron estar en el lugar y la hora correcta cuando más los necesite, gracias por todo su esfuerzo y confianza depositada en mí. A mis amigos con quienes compartí una amistad sincera ya sea dentro y fuera de las paredes de la Universidad y quienes son el pilar fundamental de mi superación, en especial gracias Andrea compañera y amiga incondicional que desde el inicio formaste parte de mi vida y que ahora juntos finalizamos una meta prometida. A mis tutores de tesis Ing. Jaqueline Llanos e Ing. Marco Pilatasig que supieron extender su mano de confianza, apoyo y dedicación en mi formación académica y personal sobre todo gracias por su amistad.

***David Curay***

A Dios por iluminar nuestra mente, fortalecer nuestro corazón y guiarnos por el sendero del bien para culminar exitosamente este proyecto de investigación. A mi compañero y buen amigo David con quién compartí momentos de alegrías y tristezas durante nuestra formación profesional hasta culminar una meta más en nuestra vida, gracias amigo por apoyarme y por estar siempre a mi lado. A los Ingenieros Jaqueline Llanos y Marco Pilatasig quiénes nos brindaron no sólo su tiempo, su apoyo sino también su amistad.

***Andrea Vaca***

## ÍNDICE DE CONTENIDO

PORTADA.....	i
CERTIFICADO.....	ii
DECLARACIÓN DE RESPONSABILIDAD .....	iii
AUTORIZACIÓN.....	iv
DEDICATORIA .....	v
AGRADECIMIENTO .....	vi
ÍNDICE DE CONTENIDO .....	vii
ÍNDICE DE FIGURAS .....	x
ÍNDICE DE TABLAS .....	xv
RESUMEN .....	xvi
ABSTRACT .....	xvii
<b>CAPÍTULO 1.....</b>	<b>1</b>
<b>MARCO TEORICO .....</b>	<b>1</b>
1.1 Concepto de sistema .....	1
1.1.1 Modelo de un sistema.....	1
1.1.2 Tipos de modelo.....	2
1.1.3 Métodos de obtención de modelos para sistemas dinámicos. ....	4
1.2 Identificación de Sistemas .....	4
1.2.1 Proceso de Identificación.....	5
1.2.2 Consideraciones al utilizar el método de identificación de sistemas. ....	7
1.3 Identificación de sistemas utilizando MATLAB .....	13
1.3.1 System Identification Toolbox de MATLAB.....	14
1.3.2 Manejo de la herramienta de MATLAB (IDENT). ....	15
1.4 Controles avanzados .....	25
1.4.1 Tipos de controles avanzados.....	26
1.5 Control Predictivo .....	29
1.5.1 Actualidad. ....	30
1.6 Control Predictivo Basado en Modelo (MBPC o MPC) .....	30
1.6.1 Ventajas del MPC. ....	31

1.6.2 Desventajas del MPC.....	32
1.6.3 Estrategia del Control Predictivo.....	33
1.6.4 Elementos del Control Predictivo.....	34
1.7 Controladores de Automatización Programable (PAC).....	40
1.7.1 Diferencias y similitudes entre un PAC y un PLC.....	41
1.7.2 Ventajas de los PACs en la adquisición de datos.....	41
1.7.3 NI Single-Board RIO de National Instruments.....	42
1.7.4 Instrumentos basados en FPGA pueden acelerar las mediciones.....	44
1.7.5 SBRIO 9636.....	46
<b>CAPÍTULO 2.....</b>	<b>49</b>
<b>DISEÑO Y SIMULACIÓN DEL CONTROL PREDICTIVO</b>	
<b>EN LA PLANTA DE PRESIÓN.....</b>	<b>49</b>
2.1 Descripción de la planta de presión.....	49
2.2 Modelamiento de la planta de Presión.....	52
2.2.1 Obtención de datos.....	53
2.2.2 Tratamiento previo de los datos.....	65
2.2.3 Modelación de la planta de presión mediante MATLAB.....	70
2.2.4 Validación del modelo.....	72
2.3 Diseño del controlador predictivo para la planta de presión.....	76
2.3.1 CD create mpc controller.....	78
2.3.2 CD update mpc windows.....	84
2.3.3 CD implement mpc controller.....	85
2.3.4 CD set mpc controller.....	86
2.3.5 Discrete State-Space.....	88
2.3.6 Datos obtenidos de la simulación del MPC.....	89
<b>CAPÍTULO 3.....</b>	<b>98</b>
<b>IMPLEMENTACIÓN DE LOS CONTROLADORES MPC Y PID.....</b>	<b>98</b>
3.1 Implementación en hardware.....	98
3.2 Implementación en software (labview).....	99
3.3 Estructura del proyecto.....	103
3.3.1 Labview FPGA.....	105



3.3.2 Labview RT. ....	107
3.3.3 Labview PC. ....	109
3.4 Acondicionamiento de señales. ....	112
3.4.1 Entrada de la SBRIO. ....	113
3.4.2 Salida de la SBRIO. ....	114
3.5 Programación del controlador MPC. ....	116
3.5.1 Programación de la FPGA ....	116
3.5.2 Programación en la RT. ....	117
3.5.3 Programación en la PC. ....	120
3.6 Programación del controlador PID. ....	123
3.6.1 Creando el proyecto PID. ....	125
3.6.2 Programación FPGA_PID ....	125
3.6.3 Programación PC_PID. ....	126
<b>CAPÍTULO 4.....</b>	<b>128</b>
<b>ANÁLISIS DE RESULTADOS DE LA IMPLEMENTACIÓN .....</b>	<b>128</b>
4.1 Análisis del controlador MPC en la planta de presión.....	128
4.2 Análisis del controlador PID en la planta de presión.....	132
4.3 Comparación entre los controladores MPC vs. PID.....	134
<b>CAPÍTULO 5.....</b>	<b>139</b>
<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>139</b>
5.1 CONCLUSIONES .....	139
5.2 RECOMENDACIONES.....	142
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>143</b>

## ÍNDICE DE FIGURAS

Figura 1.1: Representación de un sistema.....	1
Figura 1.2: Esquema general de la Identificación de Sistemas.....	5
Figura 1.3: Proceso de Identificación.....	7
Figura 1.4: a) Entrada binaria aleatoria para sistemas lineales	
b) Entrada escalonada aleatoria para sistemas no lineales.....	9
Figura 1.5: Ventana de la Interfaz Gráfica de Usuario IDENT.....	15
Figura 1.6: Selección de importar datos.....	16
Figura 1.7: Cuadro de diálogo para importar datos.....	17
Figura 1.8: Cuadro de diálogo de importar datos ampliado.....	19
Figura 1.9: Agregar datos a la herramienta.....	19
Figura 1.10: Gráfico para las entradas y salidas vs tiempo.....	20
Figura 1.11: Eliminación de medios.....	21
Figura 1.12: Gráfica de entrada/salida originales y de eliminación de medios.....	22
Figura 1.13: Datos para la identificación del sistema.....	22
Figura 1.14: Parámetros para la selección de rango.....	23
Figura 1.15: Colocación de los datos en el cuadro de estimación y validación para el modelo.....	24
Figura 1.16: Información del conjunto de datos.....	25
Figura 1.17: Expectativas y posibilidades de las técnicas de control.....	28
Figura 1.18: Estrategia del MPC.....	33
Figura 1.19: Estructura general de los controladores predictivos.....	34
Figura 1.20: Principio básico del control predictivo basado en un modelo...	35
Figura 1.21: Diferentes partes de un FPGA.....	44
Figura 1.22: SBRIO 9636.....	46

Figura 2.1: Estación de proceso de Presión. ....	50
Figura 2.2: Conversor de voltaje a corriente. ....	51
Figura 2.3: Resistencias de 249 Ohm. ....	52
Figura 2.4: Diagrama de bloques para la adquisición de datos. ....	54
Figura 2.5: Validación de rangos del sistema. ....	56
Figura 2.6: Determinación del tiempo muerto. ....	57
Figura 2.7: Dinámica del sistema. ....	59
Figura 2.8: Comparación entre la señal original y promediada. ....	60
Figura 2.9: Experimento para calcular el $T_s$ . ....	61
Figura 2.10: Entrada escalón Binaria. ....	62
Figura 2.11: Entrada escalón multinivel. ....	62
Figura 2.12: Respuesta del sistema ante una entrada escalón. ....	64
Figura 2.13: Diagrama de bloques de adquisición de datos. ....	66
Figura 2.14: Estructura del proyecto de adquisición de datos. ....	66
Figura 2.15: a) Panel frontal, b) Diagrama de bloques del VI's de la computadora. ....	67
Figura 2.16: a) Panel frontal, b) diagrama de bloques del VI's FPGA. ....	68
Figura 2.17: Revisión de los datos almacenados. ....	69
Figura 2.18: Importación de datos. ....	70
Figura 2.19: Respuesta del sistema excitada con diferentes entradas. ....	71
Figura 2.20: Modelos matemáticos del sistema. ....	72
Figura 2.21: Validación del modelo matemático obtenido. ....	73
Figura 2.22: Matrices de la ecuación de estados. ....	76
Figura 2.23: Simulación MPC. a) Diagrama frontal b) Diagrama de bloques. ....	77
Figura 2.24: Diagrama de bloques de CD create MPC controller. ....	78

Figura 2.25: Panel frontal del ingreso de los parámetros de control. ....	80
Figura 2.26: Panel frontal del ingreso de valores a la matriz costo. ....	81
Figura 2.27: Elección del tipo de restricción. ....	82
Figura 2.28: Panel frontal del método DUAL. ....	83
Figura 2.29: Panel frontal del elemento de modelo del sistema. ....	84
Figura 2.30: Diagrama de bloque del CD Update MPC Window. ....	85
Figura 2.31: Diagrama de bloque de CD Implement MPC controller. ....	85
Figura 2.32: Diagrama de bloque del CD set MPC controller. ....	87
Figura 2.33: Opciones para actualizar del controlador MPC. ....	87
Figura 2.34: Diagrama de bloque del Discrete State-Space. ....	88
Figura 2.35: Modelación del sistema en espacio de estados. ....	89
Figura 2.36: Parámetros del control MPC. ....	90
Figura 2.37: Respuesta del MPC cuando el $H_c > H_P$ . ....	91
Figura 2.38: Respuesta del controlador, a) $H_p$ corto, b) grande. ....	92
Figura 2.39: Respuesta del controlador cuando el $H_c$ es muy grande. ....	93
Figura 2.40: a) Sin matrices costo, b) Aplicando las matrices costo Q, R. ....	94
Figura 2.41: Respuesta del controlador al aplicar la matriz costo. ....	95
Figura 2.42: Respuesta del sistema implementado el controlador MPC. ....	96
Figura 3.1: Implementación del MPC y PID por hardware. ....	98
Figura 3.2: Conexión del hardware del controlador MPC y PID. ....	99
Figura 3.3: Crear proyectos de Labview. ....	100
Figura 3.4: Configuración de los dispositivos mediante MAX. ....	101
Figura 3.5: Instalación de drivers. ....	102
Figura 3.6: Añadiendo dispositivo nuevo. ....	103
Figura 3.7: Detectando el dispositivo. ....	104
Figura 3.8: Componentes de la tarjeta. ....	104

Figura 3.9: Estructura del proyecto. ....	105
Figura 3.10: Nuevo VI para la FPGA. ....	106
Figura 3.11: Manipulación de las entradas y salidas. ....	106
Figura 3.12: Creación de un VI en Real-Time. ....	107
Figura 3.13: Creando el REAL-TIME VI. ....	108
Figura 3.14: Nuevo VI en la PC. ....	109
Figura 3.15: Creando las variables compartidas. ....	110
Figura 3.16: Propiedades de la variable compartida. ....	111
Figura 3.17: Tipos de acceso. ....	112
Figura 3.18: Acondicionamiento por hardware Input. ....	113
Figura 3.19: Acondicionamiento por software (PV). ....	114
Figura 3.20: Acondicionamiento por software Output. ....	115
Figura 3.21: Acondicionamiento por hardware (CV). ....	115
Figura 3.22: Diagrama de flujo FPGA. ....	116
Figura 3.23: Programa de la FPGA en Labview. ....	117
Figura 3.24: Diagrama de flujo RT. ....	118
Figura 3.25: Programa en la RT mediante Labview. ....	119
Figura 3.26: Diagrama de flujo PC. ....	120
Figura 3.27: Programa en la PC mediante Labview. ....	121
Figura 3.28: Diagrama principal del proyecto MPC. ....	122
Figura 3.29: Proyecto creado en Labview. ....	123
Figura 3.30: SubVI PID. ....	124
Figura 3.31: Estructura del proyecto PID. ....	125
Figura 3.32: Programa en la FPGA_PID mediante Labview. ....	126
Figura 3.33: Programa en la PC_PID mediante Labview. ....	127
Figura 4.1: Respuesta real del proceso con $H_p=4$ . ....	129

Figura 4.2: Respuesta real del proceso con $H_p=50$ . .....	130
Figura 4.3: Respuesta del proceso con $H_p=150$ . .....	130
Figura 4.4: Resultado real con $H_p= 200$ . .....	131
Figura 4.5: Resultado real del controlador PID. ....	133
Figura 4.6: Respuesta del sistema del control PID. ....	133
Figura 4.7: Respuesta de los sistemas de control MPC y PID. ....	135
Figura 4.8: Indicadores de la respuesta transitoria. a) MPC, b) PID. ....	136
Figura 4.9: Acción de control MPC y PID. ....	137

## ÍNDICE DE TABLAS

Tabla 1.1: Porcentaje de satisfacción de los controles. ....	28
Tabla 1.2: Cuadro comparativo entre PLC vs. PAC .....	42
Tabla 1.3: Especificaciones generales SBRIO-9636 .....	47
Tabla 2.1: Rangos del sistema.....	56
Tabla 2.2: Tiempos muertos del sistema .....	58
Tabla 2.3: Parámetros utilizados para la identificación del proceso.....	65
Tabla 2.4: Comparación de los modelos matemáticos .....	74
Tabla 2.5: Comparación de la respuesta transitoria obtenidos en simulación .....	97
Tabla 2.6: Porcentaje de trabajo del “CV” obtenidos en simulación.....	97
Tabla 4.1: Respuesta transitoria del MPC.....	132
Tabla 4.2: Parámetros de sintonización de los controladores.....	134
Tabla 4.3: Comparación de la respuesta transitoria del MPC y PID .....	136
Tabla 4.4: Porcentaje de trabajo realizado por el actuador.....	138

## RESUMEN

En este proyecto se ha diseñado e implementado un algoritmo de control avanzado para un proceso de presión, optando por realizar un controlador MPC (Control Predictivo Basado en Modelo) el cual es parte de las metodologías de control avanzado, para la implementación del MPC se usa un sbRIO 9636 de National Instrument el cual posee un sistema de operación en tiempo real embebido y chasis configurable que incluye un FPGA. El MPC como su nombre lo indica necesita un modelo matemático experimental del proceso ya sea continuo o discreto para realizar el control, por esta razón es necesario obtener el modelo dinámico del proceso de presión, para ello se utiliza la herramienta de “Identificación de Sistemas” que posee el software MATLAB. Se usa una función costo que está sujeto a tres restricciones de desigualdad, además de un horizonte de predicción de número de pasos que opera de modo deslizante y autónomamente. El controlador diseñado es simulado e implementado en tiempo real. Finalmente se realizará el estudio comparativo de la respuesta transitoria del control MPC con respecto a un control tradicional PID, se incluye una interfaz gráfica para la visualización y monitoreo de cada controlador implementado a dicho sistema.

**PALABRAS CLAVES:** PROCESO DE PRESIÓN / CONTROL AVANZADO / MPC / IDENTIFICACIÓN DE SISTEMAS / PAC.



## ABSTRACT

This project is designed and implemented an advanced control algorithm for a pressure process, opting for making a MPC (Model Predictive Control) driver which is part of the advanced control methodologies, to implement the MPC is used a sbRIO 9636 National Instrument which has an operating system and embedded real-time configurable chassis that includes an FPGA. The MPC as its name indicate to need an experimental mathematical model of the process that can continuous or discrete for to do control, therefore it is necessary to obtain the dynamic model of pressure process, for this is used "System Identification" tool who owns the MATLAB software. Is used one cost function that is subject to three inequality constraints, and a prediction horizon number of operating steps and independently slidably. The controller is simulated designed and implemented in real time. Finally, the comparative study of the transient response of the MPC control over a traditional PID control is performed, a graphical interface is included for viewing and monitoring of each controller implemented.

**KEYWORDS:** PRESSURE PROCESS / ADVANCED CONTROL / MPC / IDENTIFICATION SYSTEMS / PAC.

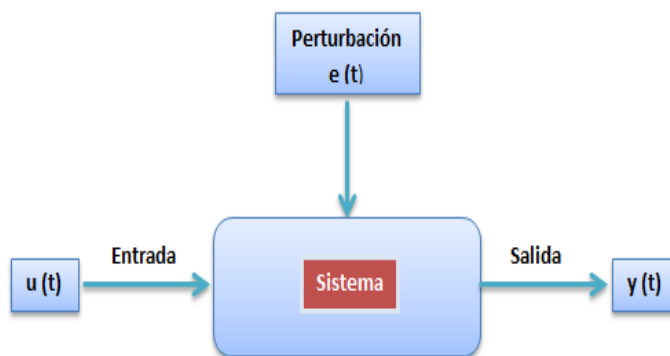
## CAPÍTULO 1

### MARCO TEORICO

#### 1.1 Concepto de sistema

Es la combinación de componentes que actúan conjuntamente y cumplen un objetivo determinado, conceptualizando profundamente se puede decir que un sistema es toda realidad en la que interactúan variables de diferentes tipos para producir señales observables, en la Figura 1.1 se muestra la representación de un sistema con sus respectivas señales.

Las señales observables que son de interés para el observador se denominan salidas del sistema, mientras que las señales que pueden ser manipuladas libremente por dicho observador son las entradas del mismo. El resto de señales que influyen en la evolución de las salidas pero no pueden ser manipuladas por el observador se denominan perturbaciones. [2]



**Figura 1.1:** Representación de un sistema.

Elaborado por: Curay, D. y Vaca, A.

##### 1.1.1 Modelo de un sistema.

Es un esquema teórico, generalmente en forma matemática, de un sistema o de una realidad compleja, es decir, es una herramienta que permite predecir el comportamiento de un sistema sin necesidad de experimentar sobre él, ya que en muchos casos la experimentación puede resultar compleja o incluso imposible de llevar a cabo. [1]

El modelo matemático equivale a una ecuación matemática o un conjunto de ellas en base a las cuales podemos conocer el comportamiento general del sistema.

Construir modelos para sistemas desconocidos es un objetivo importante de la ingeniería de control. Estos modelos necesitan simular el comportamiento real en los casos en que existe un conocimiento previo limitado de la estructura del sistema. [8]

### 1.1.2 Tipos de modelo.

Los modelos de sistemas físicos pueden ser de muy diversos tipos, las cuales se clasifican:

En función del grado de formalismo matemático que poseen, es la siguiente:

**Modelos mentales, intuitivos o verbales.** Estos modelos carecen de formalismo matemático. Para conducir un coche, por ejemplo, se requiere un modelo mental o intuitivo sobre el efecto que produce el movimiento del volante, pero no es necesario caracterizar dicho efecto mediante ecuaciones matemáticas exactas.

**Modelos no paramétricos.** Muchos sistemas quedan perfectamente caracterizados mediante un gráfico o tabla que describa sus propiedades dinámicas mediante un número no finito de parámetros.

Por ejemplo, un sistema lineal queda definido mediante su respuesta al impulso o al escalón, o bien mediante su respuesta en frecuencia.

**Modelos paramétricos o matemáticos.** Para aplicaciones más avanzadas, puede ser necesario utilizar modelos que describan las relaciones entre las variables del sistema mediante expresiones matemáticas como pueden ser ecuaciones diferenciales (para sistemas continuos) o en diferencias (para sistemas discretos). [1]

En función del tipo de sistema y de la representación matemática utilizada, los sistemas pueden clasificarse en:

**Determinísticos.** Se dice que un modelo es determinístico cuando expresa la relación entre entradas y salidas mediante una ecuación exacta.

**Estocástico.** Un modelo es estocástico si posee un cierto grado de incertidumbre. Estos últimos se definen mediante conceptos probabilísticos o estadísticos.

**Estáticos.** Un sistema es estático cuando la salida depende únicamente de la entrada en ese mismo instante (un resistor, por ejemplo, es un sistema estático). En estos sistemas existe una relación directa entre entrada y salida, independiente del tiempo.

**Dinámico.** Un sistema dinámico es aquél en el que las salidas evolucionan con el tiempo tras la aplicación de una determinada entrada (por ejemplo, una red RC). En estos últimos, para conocer el valor actual de la salida es necesario conocer el tiempo transcurrido desde la aplicación de la entrada.

**Continuos.** Los sistemas continuos trabajan con señales continuas, y se caracterizan mediante ecuaciones diferenciales.

**Discreto.** Los sistemas discretos trabajan con señales muestreadas, y quedan descritos mediante ecuaciones en diferencias.

Todo modelo matemático o paramétrico, consta de una o varias ecuaciones que relacionan las entradas y salidas (en los modelos dinámicos la variable  $t$  -tiempo juega también un papel primordial). Por tal razón que a los modelos matemáticos, se les conozca más comúnmente como modelos paramétricos, ya que pueden definirse mediante una estructura y un número finito de parámetros.[6]

### 1.1.3 Métodos de obtención de modelos para sistemas dinámicos.

Existen dos métodos principales para la obtención del modelo de un sistema dinámico.

**Modelado teórico.** Se trata de un método analítico, en el que se recurre a leyes básicas de la física para describir el comportamiento dinámico de un fenómeno o proceso.

**Identificación del sistema.** Se trata de un método experimental que permite obtener el modelo de un sistema a partir de datos reales recogidos de la planta bajo estudio.

El modelado teórico tiene un campo de aplicación restringido a procesos muy sencillos de modelar o aplicaciones en que no se requiera gran exactitud en el modelo obtenido. En muchos casos, además, la estructura del modelo obtenido a partir del conocimiento físico de la planta posee un conjunto de parámetros desconocidos y que sólo se pueden determinar experimentando sobre el sistema real. De ahí la necesidad de recurrir a los métodos de "Identificación de Sistemas".

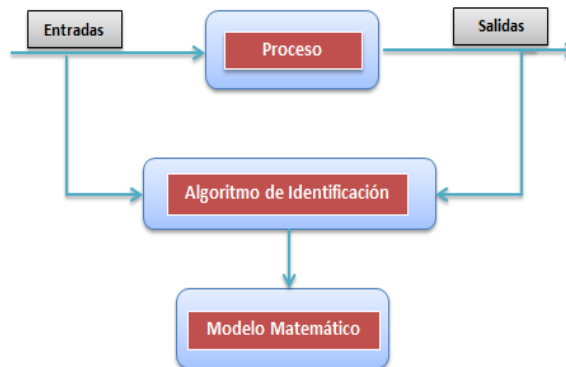
En la práctica, lo ideal es recurrir a una mezcla de ambos métodos de modelado para obtener el modelo final.

El uso de datos reales para identificar los parámetros del modelo provee a éste de una gran exactitud, pero el proceso de identificación se ve tanto más facilitado cuanto mayor sea el conocimiento sobre las leyes físicas que rigen el proceso. [2]

## 1.2 Identificación de Sistemas

La identificación de procesos consiste en determinar u obtener de forma experimental el modelo de un sistema dinámico que reproduzca con suficiente exactitud las características dinámicas del proceso a partir de las mediciones o datos de entrada y salida correspondiente de dicho sistema

objeto de estudio, en la Figura 1.2 se muestra el esquema general de la identificación.



**Figura 1.2:** Esquema general de la Identificación de Sistemas.

Elaborado por: Curay, D. y Vaca, A.

El término Identificación de sistemas fue presentado por Lofti Zadeh [9] en 1962, como:

“Identificación es la determinación, en base a la entrada y la salida, de un sistema, dentro de una clase de sistemas especificada, al cual el sistema probado es equivalente”.

Puede decirse que la identificación de sistemas quedó establecida como un campo de investigación reconocido dentro del área de control automático a mediados de los sesenta: en el tercer congreso de la IFAC (INTERNATIONAL FEDERATION OF AUTOMATIC CONTROL) en Londres, 1966 en el que fue presentado un artículo de visión general sobre identificación de sistemas (Eykhoff et al. 1966). Un año después fue organizado el primer Symposium IFAC sobre identificación de sistemas en Praga. En la actualidad es el Symposium con una serie más larga de la IFAC. [7]

### 1.2.1 Proceso de Identificación.

Para obtener el modelamiento de sistemas dinámicos, el proceso de identificación comprende los siguientes pasos:

**Obtención de datos de entrada – salida.** Se debe excitar el sistema mediante la aplicación de una señal de entrada y almacenar en un registro la evolución de sus entradas y salidas durante un intervalo de tiempo.

**Tratamiento previo de los datos registrados.** Los datos almacenados están generalmente acompañados de ruidos indeseados u otro tipo de imperfecciones que puede ser necesario corregir antes de iniciar la identificación del modelo. Se trata, por tanto, de ‘preparar’ los datos para facilitar y mejorar el proceso de identificación.

**Elección de la estructura del modelo.** Si el modelo que se desea obtener es un modelo paramétrico, el primer paso es determinar la estructura deseada para dicho modelo. Este punto se facilita en gran medida si se utiliza el toolkit de “System Identification Toolbox” de MATLAB.

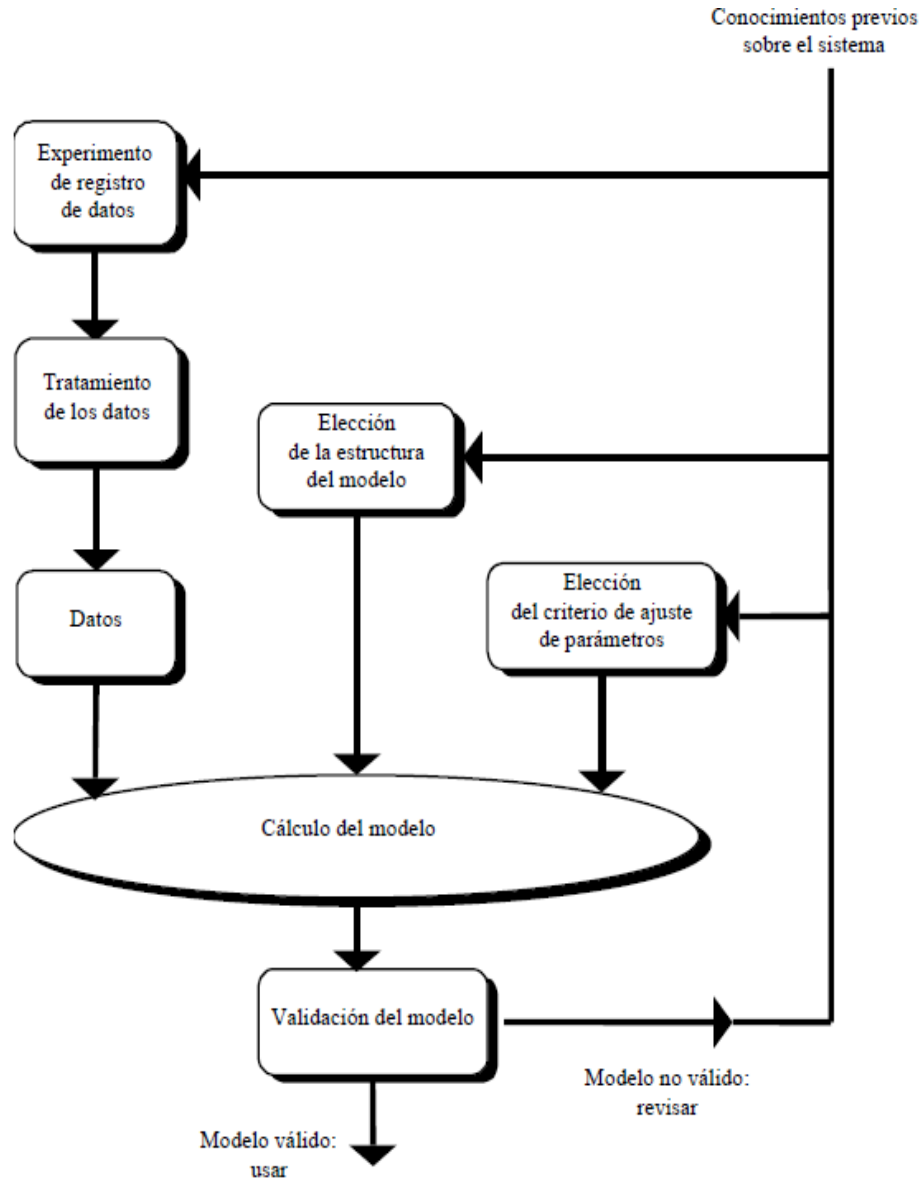
**Obtención de los parámetros del modelo.** A continuación se procede a la estimación de los parámetros de la estructura que mejor ajustan la respuesta del modelo a los datos de entrada-salida obtenidos experimentalmente.

**Validación del modelo.** El último paso consiste en determinar si el modelo obtenido satisface el grado de exactitud requerido para la aplicación en cuestión. Si se llega a la conclusión de que el modelo no es válido, se deben revisar los siguientes aspectos como posibles causas:

- ❖ El conjunto de datos de entrada-salida no proporciona suficiente información sobre la dinámica del sistema.
- ❖ La estructura escogida no es capaz de proporcionar una buena descripción del modelo.
- ❖ El criterio de ajuste de parámetros seleccionado no es el más adecuado.

Dependiendo de la causa estimada, deberá repetirse el proceso de identificación desde el punto correspondiente. Por tanto, el proceso de

identificación es un proceso interactivo, cuyos pasos pueden observarse en el organigrama de la Figura 1.3. [2]



**Figura 1.3:** Proceso de Identificación.

Fuente: [2]

### 1.2.2 Consideraciones al utilizar el método de identificación de sistemas.

Consideraciones prácticas a tener en cuenta durante el proceso de obtención del modelo mediante la utilización del método de identificación de sistemas.



### **a. Obtención de los datos.**

El primer paso dentro del proceso de identificación, es realizar algún tipo de experimento sobre el sistema bajo estudio para obtener los datos de entrada-salida, que servirán de base para la obtención del modelo final.

Para que el proceso de identificación sea satisfactorio, es necesario que los datos utilizados para tal fin, contengan información significativa sobre el sistema. Esto implica un cuidadoso diseño del experimento de adquisición de datos, debiéndose tomar una serie de decisiones respecto a las señales que deben ser medidas, el periodo de muestreo a utilizar, el tipo de entrada más adecuada, el número de datos a almacenar, etc.

#### **Elección de la señal a medir.**

La primera decisión es saber que señales se deben registrar (mediante algún tipo de sistema de adquisición y el correspondiente sistema de almacenamiento de datos), y qué señales deben ser manipuladas para excitar al sistema durante el experimento. Se debe tener en cuenta que pueden existir señales que, aunque afecten a la evolución de la salida, no pueden considerarse como entradas debido a la imposibilidad de actuar sobre ellas.

En el caso de que estas señales puedan ser medidas, pueden considerarse también como entradas al sistema (midiéndose sus valores durante el experimento). En caso contrario, deben ser consideradas como perturbaciones.

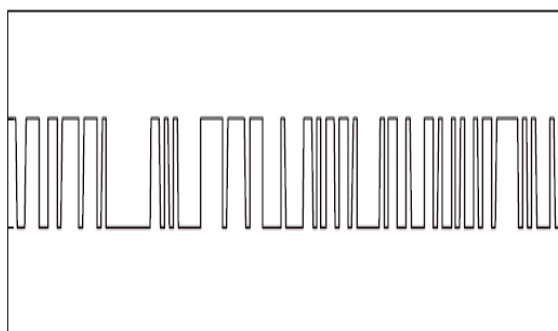
#### **Elección del tipo de entrada(s).**

La(s) entrada(s) al sistema deben ser cuidadosamente elegidas de forma que los datos recogidos proporcionen toda la información posible sobre el sistema. Por tal razón, conviene tener en cuenta los siguientes aspectos:

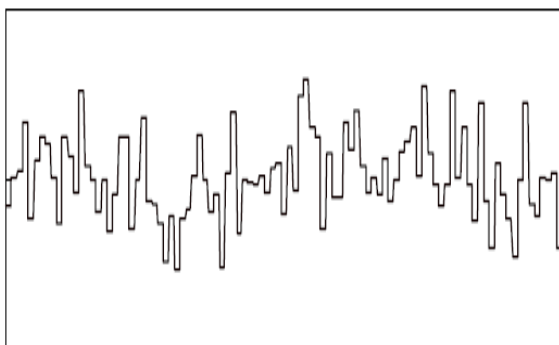
- ❖ La señal de entrada debe contener el mayor número de frecuencias posibles. Por ejemplo, una señal senoidal pura no es adecuada en un

experimento de identificación, puesto que sólo se obtendrá la respuesta del sistema para la frecuencia de dicha señal. Por el contrario, las señales escalonadas (con cambios bruscos) son muy utilizadas, puesto que contienen un espectro suficientemente amplio de frecuencias.

- ❖ Para sistemas lineales, basta con utilizar dos niveles de entrada, preferiblemente barriendo todo el rango de variación permitido. En este tipo de sistemas se suelen utilizar señales binarias de duración aleatoria (conocidas como señales binarias aleatorias o pseudoaleatorias), como la mostrada en la Figura 1.4.a). Sin embargo, para sistemas no lineales es necesario trabajar con más de dos niveles de entrada, como se muestra en la Figura 1.4.b).



a)



b)

**Figura 1.4:** a) Entrada binaria aleatoria para sistemas lineales, b) Entrada escalonada aleatoria para sistemas no lineales.

Fuente: [2], [6].

- ❖ Si se sabe que el sistema va a trabajar preferentemente en torno a un determinado punto de trabajo, es conveniente realizar el registro de datos en ese mismo entorno, este aspecto adquiere especial importancia si el sistema no es lineal.

### **Elección del periodo de muestreo.**

La elección del periodo de muestreo está directamente relacionada con las constantes de tiempo del sistema, y tiene una influencia decisiva en el experimento de identificación. Así, un periodo de muestreo muy pequeño puede llevar a la obtención de datos redundantes, que no aportan información sobre el sistema (pero sí ocupan espacio en la memoria del dispositivo de almacenamiento de datos), mientras que un periodo de muestreo demasiado grande provoca grandes dificultades a la hora de identificar la dinámica del sistema.

Una regla comúnmente usada consiste en escoger una frecuencia de muestreo alrededor de diez veces el ancho de banda del sistema. Esto corresponde aproximadamente a muestrear en torno a cinco u ocho valores del tiempo de subida de la respuesta al escalón del sistema.

### **Elección del número de muestras a tomar.**

En principio, cuanto más información se tenga sobre el sistema, más exacto será el proceso de identificación. En la práctica, el número de muestras a recoger durante el experimento de identificación viene limitado por la capacidad del dispositivo de memoria utilizado. Por tanto, es importante llegar a un buen compromiso en la elección del periodo de muestreo y el número de muestras a tomar. [2]

#### **b. Pretratamiento de los datos.**

Los datos registrados pueden tener deficiencias que implican efectos devastadores en el resto del proceso de identificación, como son las siguientes:

- ❖ Presencia de perturbaciones de alta frecuencia, por encima de las frecuencias de interés en la respuesta del sistema.
- ❖ Datos claramente erróneos, producidos por fallos en el hardware o software utilizados en el experimento de recogida de muestras.
- ❖ Desviaciones, desplazamientos o perturbaciones de baja frecuencia.

A continuación, se describe la forma de tratar cada una de estas deficiencias para conseguir datos adecuados para el proceso de identificación.

### **Eliminación de perturbaciones de alta frecuencia.**

Estas perturbaciones se producen por fuentes de ruido ajenas al sistema y pueden ser evitadas mediante una correcta elección del período de muestreo. Si, tras el experimento, se observa que el período de muestreo escogido era innecesariamente pequeño (captándose por tanto estas perturbaciones indeseadas), se puede recurrir al diezmado de los datos, para evitar repetir el experimento con un período de muestreo mayor.

### **Eliminación de datos erróneos.**

Estos datos suelen presentarse de forma aislada, pero pueden tener un efecto muy negativo en el proceso de identificación. Por tanto, es fundamental eliminarlos antes de iniciar el proceso.

Esto se realiza generalmente manualmente, eliminando dicho dato y aproximando su nuevo valor mediante interpolación. Para aplicaciones más avanzadas, existen algoritmos de detección de fallos que permiten corregir estos datos de forma casi automática.

### **c. Validación del modelo.**

En todo proceso de identificación es conveniente probar varias estructuras y diferentes órdenes dentro de cada estructura hasta dar con el modelo que mejor se ajuste a los datos obtenidos experimentalmente de la planta real.

En definitiva, se trata de determinar cuándo un determinado modelo es lo suficientemente exacto para la aplicación requerida, proceso que se conoce habitualmente como “*validación del modelo*”. En general, la mayoría de los métodos de validación tratan de determinar si la respuesta del modelo se ajusta con suficiente exactitud a los datos de entrada-salida obtenidos mediante experimentación.

A continuación se exponen algunos criterios típicos a la hora de descartar o elegir unos modelos respecto a otros.

### **Validación en base a la aplicación del modelo.**

Puesto que en la práctica es imposible determinar si un modelo responde exactamente al comportamiento de un sistema real, suele ser suficiente comprobar que el modelo es capaz de resolver el problema para el cual ha sido hallado (simulación, predicción, diseño de un controlador, etc.). Así, por ejemplo, si el controlador que ha sido ajustado por medio del modelo da buen resultado sobre el sistema real, se puede asegurar que el modelo era ‘válido’ para esta aplicación.

### **Coherencia con el comportamiento de entrada-salida.**

Para determinar si el comportamiento de entrada-salida está suficientemente caracterizado, puede ser necesario recurrir a diferentes métodos de identificación y comparar los resultados obtenidos. Por ejemplo, comparando los diagramas de Bode de los modelos obtenidos mediante identificación paramétrica de diferentes estructuras, por el método de variables instrumentales y por análisis espectral, se puede determinar si la dinámica del sistema ha quedado suficientemente caracterizada.

### **Reducción del modelo.**

Un procedimiento para determinar si un modelo proporciona una descripción simple y apropiada de un sistema consiste en aplicarle algún método de reducción de modelos. Si una reducción en el orden del modelo

no produce alteraciones apreciables en el comportamiento de entrada-salida del mismo, entonces el modelo original era innecesariamente complejo. [6]

### **1.3 Identificación de sistemas utilizando MATLAB**

Matlab es un paquete de software orientado hacia el cálculo numérico científico e ingeniería. Integra cálculo numérico, computación de matrices y gráficas en un entorno de trabajo cómodo para el usuario. Su nombre significa Laboratorio de Matrices y fue escrito inicialmente en base a los ya existentes paquetes de cálculo matricial LINPACK y EISPACK.[13]

Posteriormente se han añadido librerías, denominadas Toolboxes, especializadas en diferentes áreas científicas. De entre ellas podemos destacar:

- ❖ Simulink Toolbox.
- ❖ Control System Toolbox.
- ❖ System Identification Toolbox.
- ❖ Robust Control Toolbox.
- ❖ Signal Processing Toolbox.
- ❖ Filter Design Toolbox.
- ❖ Symbolic Math Toolbox.

Matlab se utiliza para investigar y resolver problemas prácticos y cálculos de ingeniería. Son aplicaciones típicas el cálculo numérico, la realización de algoritmos, la resolución de problemas con formulación matricial, la estadística, la optimización, etc.

Cabe destacar la aplicación en el estudio, la simulación y diseño de los sistemas dinámicos y de control [9]. Por su particular interés para nuestra área de conocimiento, el toolbox System Identification posee las herramientas necesarias para el trabajo de modelación de sistemas dinámicos.

### 1.3.1 System Identification Toolbox de MATLAB.

El sistema de Identificación de MATLAB ofrece funciones, bloques de Simulink y una aplicación para la construcción de modelos matemáticos de sistemas dinámicos a partir de los datos de entrada-salida medidos. Permite crear y utilizar modelos de sistemas dinámicos no fáciles de modelar a partir de primeros principios o especificaciones.

Se puede utilizar el dominio del tiempo y los datos de entrada-salida de dominio de frecuencia para identificar en tiempo continuo y de tiempo discreto funciones de transferencia, función de procesos de modelos, y los modelos de espacio de estado. La caja de herramientas también proporciona algoritmos para la estimación de parámetros en línea. [13]

Como se mencionó anteriormente para comenzar con la identificación se necesita seguir pasos los cuales se encuentran resumidos a continuación:

- ❖ Recolectar datos de entrada y salida del proceso a ser identificado (planta).
- ❖ Examinar los datos. Pulirlos para eliminar ruido excesivo y seleccionar porciones útiles de los datos originales.
- ❖ Seleccionar una estructura del modelo.
- ❖ Calcular el mejor modelo en base a los datos de entrada y salida.
- ❖ Examinar las propiedades del modelo obtenido.
- ❖ Si el modelo es suficientemente bueno, termina el procedimiento. De lo contrario regresar al paso 3 y probar otro modelo.

El System Identification Toolbox de MATLAB ofrece varias funciones para realizar estos pasos además de una interfaz gráfica (GUI) para realizar la identificación de manera interactiva.

Para poder realizar el procedimiento descrito en este documento, se necesita tener instalado en MATLAB los siguientes Toolbox:

- ❖ System Identification Toolbox.

❖ Data Acquisition Toolbox.

Para saber que toolbox tiene instalados la plataforma MATLAB se ejecuta el comando “*VER*” en MATLAB. Este comando muestra una lista de todos los toolbox actualmente instalados. [13]

### 1.3.2 Manejo de la herramienta de MATLAB (IDENT).

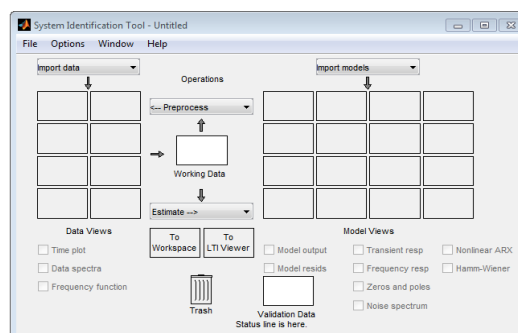
Para empezar con el manejo del toolbox de Matlab (IDENT), se elijará un ejemplo guardado en el entorno de MATLAB para la preparación de los datos y obtener la identificación del sistema, a continuación se detalla el procedimiento:

#### a. Cargar los datos en el espacio de trabajo de MATLAB.

Se debe cargar los datos en dryer2.mat escribiendo el comando “Load dryer2” en la ventana de comandos de MATLAB. Este comando carga los datos en el espacio de trabajo de MATLAB como dos vectores columna, U2 y Y2, respectivamente. El U2 es variable de los datos de entrada y Y2 es los datos de salida.

#### b. Abriendo la herramienta de identificación del sistema.

Para abrir la herramienta de identificación del sistema, escribir en la ventana de comandos de MATLAB “Ident”. Por defecto el nombre de la sección aparece en la barra de títulos en la Figura 1.5 se muestra la Interfaz Gráfica para el Usuario.



**Figura 1.5:** Ventana de la Interfaz Gráfica de Usuario IDENT.

Elaborado por: Curay, D. y Vaca, A.

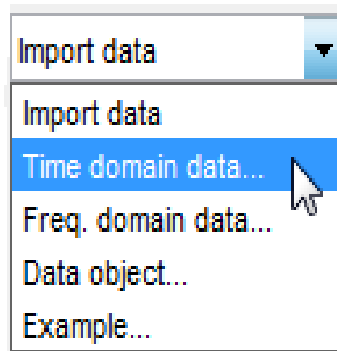


### c. Importar arreglos de datos en la herramienta de identificación de sistemas.

Se puede importar los datos single-input/single-output (SISO) de una muestra de archivo de datos dryer2.mat en la interfaz gráfica de usuario desde el espacio de trabajo de MATLAB, teniendo en cuenta que los datos de entrada/salida deben estar ya cargados en el área de trabajo de MATLAB.

Para importar conjuntos de datos en la herramienta de identificación del sistema se debe seguir los siguientes pasos:

- ❖ En la herramienta de identificación del sistema, seleccione “Import data” seguido de “Time domain Data” como se muestra en la Figura 1.6. Esta acción abre el cuadro de diálogo Importar datos.



**Figura 1.6:** Selección de importar datos.

Elaborado por: Curay, D. y Vaca, A.

- ❖ En el cuadro de diálogo Importar datos, especifique las siguientes opciones:

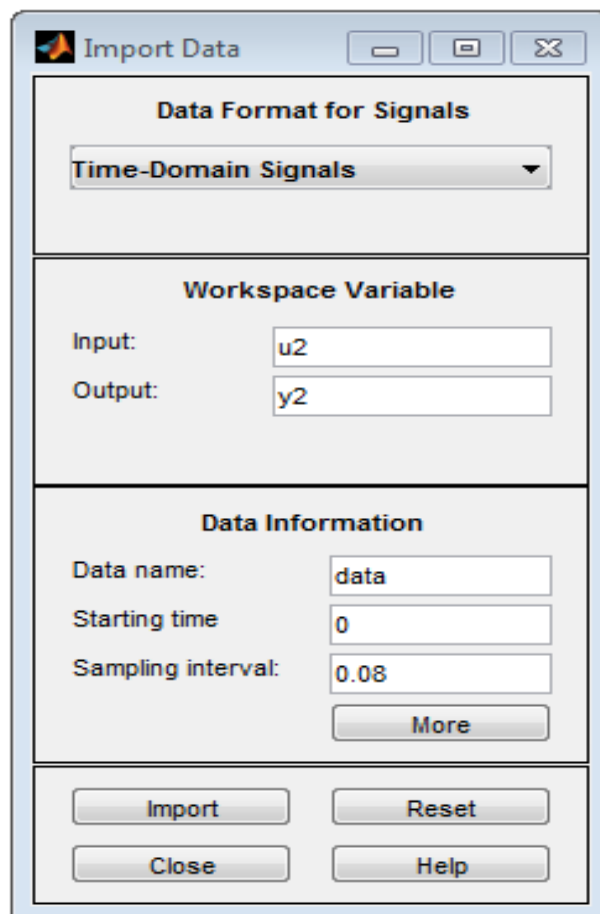
**Entrada.** Introducir el nombre de la variable de entrada en la cual se encuentra guardado los datos, en este caso será u2 (ver Figura 1.7).

**Salida.** Introducir el nombre de la variable de salida en la cual se encuentra guardado los datos, en este caso será y2 (ver Figura 1.7).

**Nombre de los datos.** Cambiar el nombre por defecto a los datos. Este nombre etiqueta los datos de la herramienta de identificación del sistema después de que se complete la operación de importación.

**Hora de inicio.** Designar el valor de partida del eje de tiempo en el gráfico.

**Intervalo de muestreo.** Es el intervalo de muestreo real en el experimento. El cuadro de diálogo Importar datos ahora se asemeja a la Figura 1.7:



**Figura 1.7:** Cuadro de diálogo para importar datos.

Fuente: [13].

- ❖ En el área Información de datos, haga clic en “More” para expandir el cuadro de diálogo y especifique las siguientes opciones:

**Propiedades de entrada:**

- ❖ **Entre muestras.** Aceptar por defecto “zoh” (mantenedor de orden cero) para indicar que la señal de entrada era por tramos y constante entre muestras durante la adquisición de datos. Esta configuración específica el comportamiento de las señales de entrada entre las muestras o cuando transforma las representaciones de modelos resultantes entre tiempo discreto y en tiempo continuo.
- ❖ **Período.** Aceptar la información predeterminado para especificar una entrada no periódica.

**Nombres de los canales:**

- ❖ Entrada.
- ❖ Salida.

Permite ingresar los nombres a los canales con su respectiva variable física ayudando a identificar los datos de la gráfica.

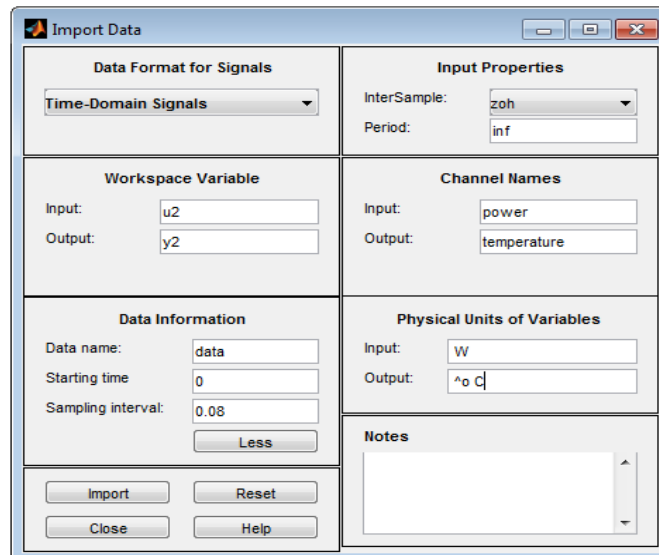
Para las señales de entrada y salida de varias variables, puede especificar los nombres de los canales de entrada y salida individuales separados por comas.

**Unidades físicas de Variables:**

- ❖ Entrada.
- ❖ Salida.

Permite ingresar las unidades de las variables físicas utilizadas para la identificación de sistema, cuando se tiene múltiples entradas y salidas, introducir una lista separada por comas de las unidades de entrada y salida correspondiente a cada canal. Como otra opción se puede introducir comentarios acerca del experimento o los datos. Por ejemplo, se puede introducir el nombre del experimento, la fecha y una descripción de las condiciones experimentales. Al estimar los modelos de estos datos, estos

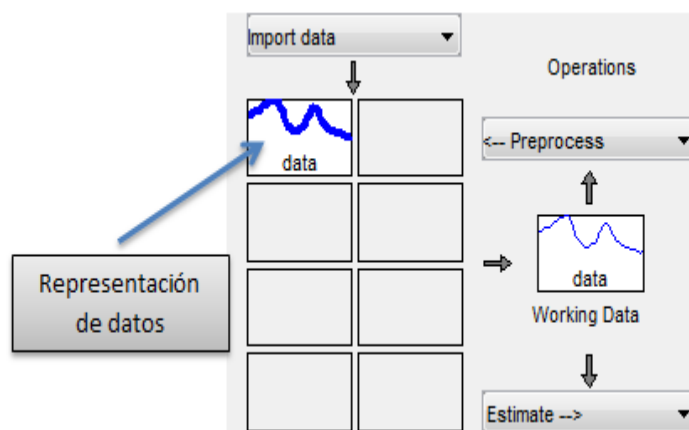
modelos heredan sus notas de datos. El cuadro de diálogo de importar datos ampliado ahora se asemeja a la Figura 1.8:



**Figura 1.8:** Cuadro de diálogo de importar datos ampliado.

Fuente: [13].

- ❖ Hacer clic en Importar para agregar los datos a la herramienta de identificación del sistema. La herramienta muestra un icono para representar los datos tal como se presenta en la Figura 1.9.



**Figura 1.9:** Agregar datos a la herramienta.

Fuente: [13.]

- ❖ Hacer clic en “Close” para cerrar el cuadro de diálogo de importar datos.

#### d. Trazado y procesamiento de datos.

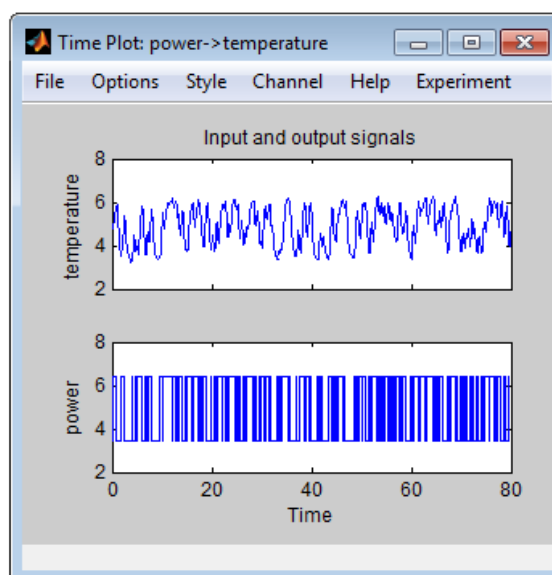
En esta parte se aprenderá a evaluar los datos y el proceso para la identificación del sistema:

- ❖ Representar gráficamente los datos.
- ❖ Retirar compensaciones a partir de los datos, restando los valores medios de la entrada y la salida.
- ❖ Dividir los datos en dos partes para utilizar el modelo de estimación de una parte y la otra parte para la validación del modelo.

La razón de restar los valores medios de cada señal se debe a que, típicamente, va a construir modelos lineales que describen las respuestas de las desviaciones a partir de un equilibrio físico. Con datos en estado estacionario, es razonable suponer que los niveles medios de las señales corresponden a tal equilibrio.

Para representar y procesar los datos se sigue los pasos descritos a continuación:

- ❖ En la herramienta de identificación del sistema, seleccione la “Time plot” para abrir el Gráfico de tiempo.



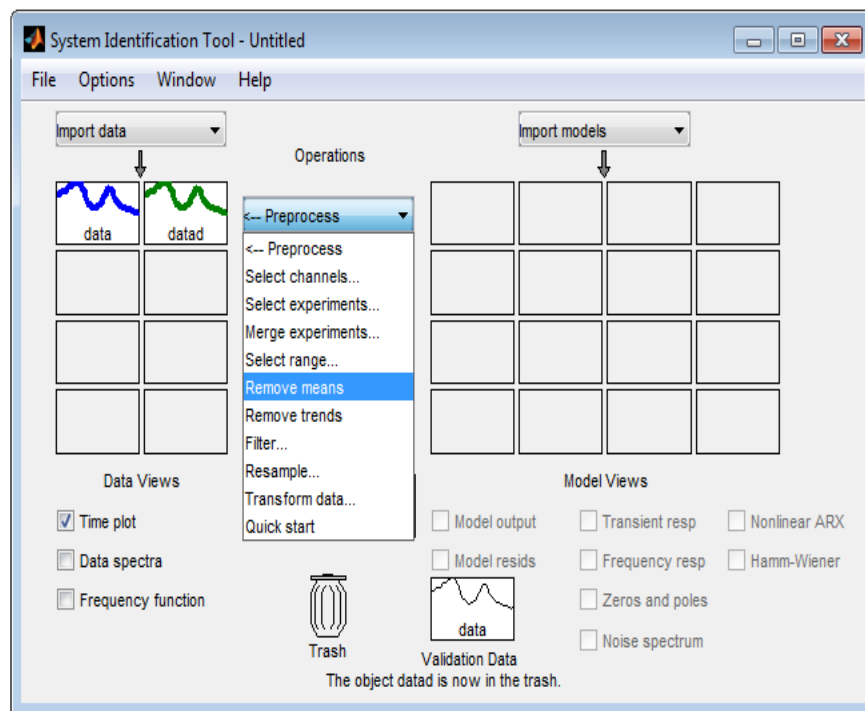
**Figura 1.10:** Gráfico para las entradas y salidas vs tiempo.

Fuente: [13].

Si la ventana se encuentra vacía, hacer clic en el icono de datos en la herramienta de identificación del sistema, en la Figura 1.10 se muestra los gráficos de entrada y salida en el tiempo.

Los ejes principales muestran los datos de salida (temperatura), y los ejes inferiores muestran los datos de entrada (alimentación). Tanto la entrada y los datos de salida tienen valores medios distintos de cero.

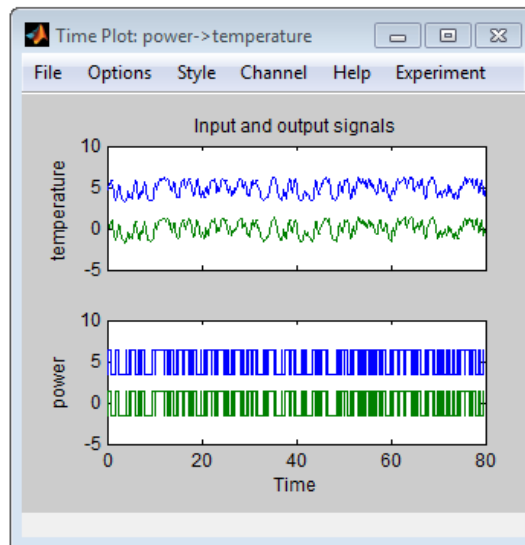
- ❖ Reste el valor de entrada a partir de los datos de entrada y el valor de la producción media de los datos de salida. En la herramienta de identificación del sistema, seleccione “Preprocess” y luego “Remove means”, como se observa en la Figura 1.11.



**Figura 1.11:** Eliminación de medios.

Fuente: [13].

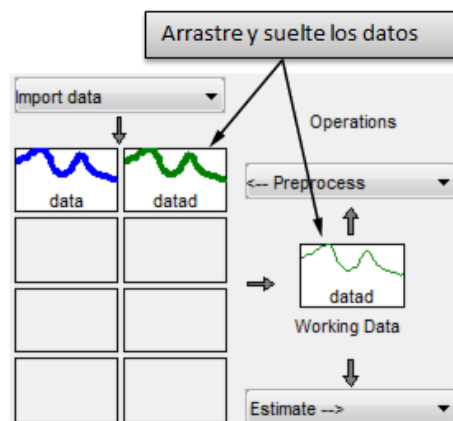
Esta acción agrega un nuevo conjunto de herramientas para la identificación del sistema con el nombre datad por defecto (el sufijo d significa Detrend) de datos y actualiza la ventana de gráfico de tiempo para que aparezca el original y los datos sin tendencia son indicados en la Figura 1.12. Los datos sin tendencia tienen un valor medio de cero.



**Figura 1.12:** Gráfica de entrada/salida originales y de eliminación de medios.

Fuente: [13].

- ❖ Especificar los datos sin tendencia a ser utilizados para la estimación de los modelos. Arrastrar el conjunto de datos al rectángulo de datos de trabajo como se indica en la Figura 1.13.

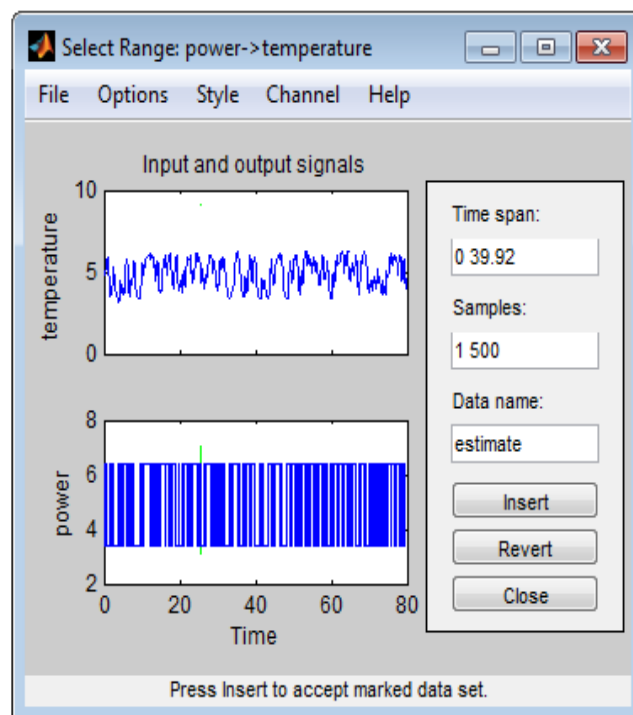


**Figura 1.13:** Datos para la identificación del sistema.

Fuente: [13.]

- ❖ Dividir los datos en dos partes y especificar la primera parte de la estimación del modelo, y la segunda parte de la validación del modelo, como se describe en los siguientes pasos.
  - ❖ Seleccionar “Preproceso” y luego “Select range” para abrir la ventana de selección de rango para la estimación.

- ❖ En la ventana Seleccionar Rango, crear un conjunto de datos que contiene las primeras 500 muestras. En el campo de las muestras, especificar 1 500. También se puede seleccionar muestras de datos con el ratón haciendo clic y arrastrando una región rectangular en la parcela. Si se selecciona las muestras en los ejes de entrada de los canales, la región correspondiente se selecciona también en los ejes de salida de los canales en la Figura 1.14 se observa la elección de rango para la estimación.
- ❖ En el campo Nombre, escribir el nombre de la estimación de los rangos para la validación.



**Figura 1.14:** Parámetros para la selección de rango.

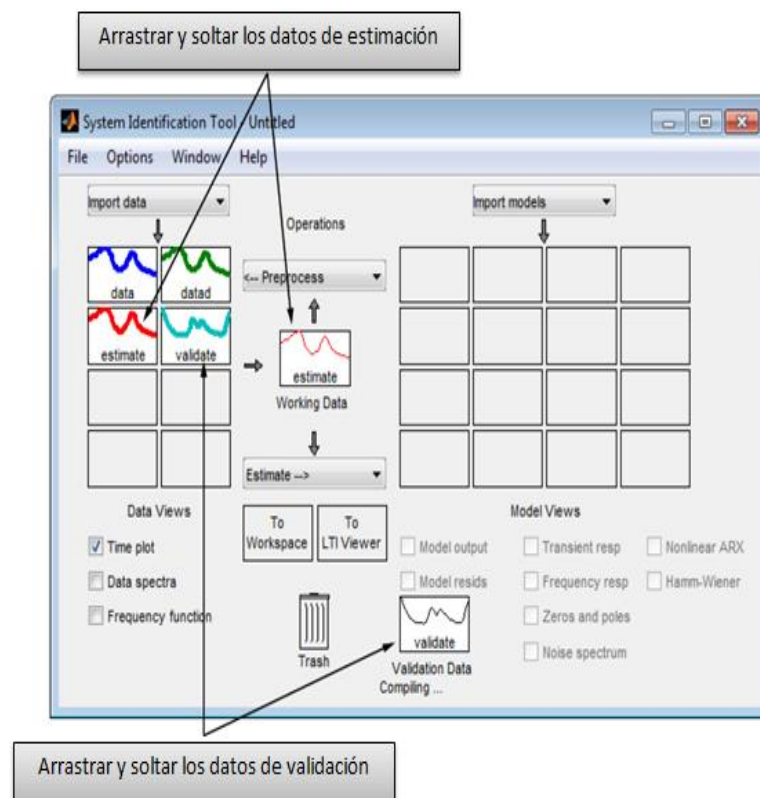
Fuente: [13].

- ❖ Hacer clic en “Insert” para agregar este nuevo conjunto de herramientas para la identificación del sistema que se utilizará para la estimación del modelo de datos.
- ❖ Repetir este proceso para crear un segundo conjunto de datos que contiene un subconjunto de los datos a utilizar para la



validación. En la ventana Seleccionar Rango, especificar los últimos 500 muestras en el campo de las muestras. Escribir el nombre de validación en el campo Nombre de Datos. Hacer clic en Insertar para agregar este nuevo conjunto de herramientas para la identificación del sistema de datos.

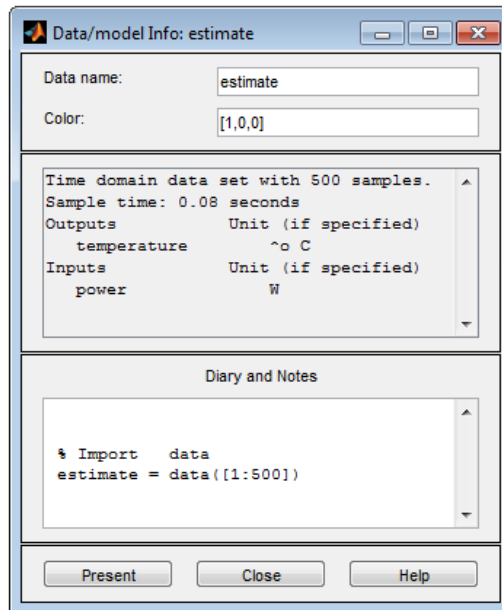
- ❖ En la herramienta de identificación del sistema, arrastrar y soltar la estimación al rectángulo de datos de trabajo, y arrastrar y soltar validate al rectángulo de validación de datos como se indica en la Figura 1.15.



**Figura 1.15:** Colocación de los datos en el cuadro de estimación y validación para el modelo.

Elaborado por: Curay, D. y Vaca, A.

- ❖ Para obtener información sobre un conjunto de datos, haga clic en su icono. Por ejemplo, haga clic en los datos de estimación establecidos para abrir el cuadro de diálogo Información de Datos / modelo, Figura 1.16.



**Figura 1.16:** Información del conjunto de datos.

Fuente: [13].

#### e. Selección de una estructura de modelo para el sistema.

De la lista desplegable “estimate”, seleccionar el modelo preferentemente con el que se va a trabajar en este proyecto y el que represente de mejor manera las características del sistema dinámico a estudio. [13]

### 1.4 Controles avanzados

Control Avanzado hace referencia a la aplicación de estrategias de control automático que trasciendan las que usualmente se aplican en control de procesos. Ejemplos de las técnicas de control más usuales de sistemas SISO son el control PID en sus diversas formas de ajuste manual o automático, control de razón, sistemas de “Adelanto-Atraso” por reasignación de polos y ceros, control digital discreto (usando PLC’s), etc., las mismas que pueden aplicarse sobre sistemas MIMO (multivariables). En general, estas estrategias pueden denominarse de “parámetros óptimos” ya que parte de una estructura algorítmica fija, se modifican sus parámetros con el objeto de lograr la respuesta del proceso que mejor se adecue a los requerimientos del mismo, conforme a un criterio de optimización fácil de entender y manejar por un operador de planta.

Esta forma de operar tiene entre sus ventajas principales, que puede ser fácilmente implantada con circuitos electrónicos analógicos y/o digitales, pudiendo construirse controladores PID en gabinetes compactos y comercializarse individualmente como tal. Igualmente la simpleza de la programación de un algoritmo PID digital ha posibilitado que la mayor parte de las empresas fabricantes de equipos del rubro de la automatización (PLC's, sistemas de adquisición de datos (DAS), o sistemas de control distribuido (DCS)) incorporen módulos de control PID dentro de sus productos.

Por lo general, este tipo de control presenta buenas condiciones de desempeño para la mayoría de los procesos industriales tradicionales, lo que prácticamente asegura su permanencia en el tiempo.

Sin embargo, existen procesos que por su grado de complejidad no pueden ser operados usando técnicas de control convencional de forma eficiente, y requieren aplicar técnicas de control avanzado para alcanzar las prestaciones deseadas. [17], [18]

#### **1.4.1 Tipos de controles avanzados.**

Existen diversas estrategias de control que responden a esta clasificación, sin embargo, la dificultad para poder ser operadas por personal con escaso nivel de formación profesional así como para generalizarlas a sistemas de diferente naturaleza, ha provocado que pocas sean las que han trascendido del ámbito académico al industrial. A continuación, se hace referencia a los controles más importantes en cuanto a la utilización y aplicación industrial.

- ❖ **Control Experto.** Este tipo de control se basa en la recopilación de conocimiento de un sistema, a partir de operadores del mismo que pueden considerarse como “expertos” en su área de conocimiento. El mejor exponente de este tipo de control, por el grado de aceptación que ha tenido a nivel industrial es el control Difuso o por Lógica Difusa (“Fuzzy Logic Control”). [17]

- ❖ **Control Óptimo:** Este control se basa en la definición de una función que por lo general incluye el error de control y la acción de control y/o sus desviaciones, con ponderaciones que permiten “pesar” en forma relativa cada una de ellas, y se establece un criterio de optimización sobre dicho funcional que se ajuste a los objetivos del control.

El mejor representante de este tipo de control, por la gran aceptación que ha tenido en el campo del control de procesos Control Predictivo por Modelo (“Model Predictive Control” o “MPC”) [17].

- ❖ **Control Robusto:** Consiste en definir una estructura de control que tenga un desempeño acorde a las especificaciones del sistema, independientemente de las perturbaciones a las que esté expuesto. A nivel académico, los mayores desarrollos en torno a este tema tienen relación con el Control Óptimo  $H_\infty$  [17].

- ❖ **Control Adaptivo, Adaptativo o Adaptable (“Adaptive Control”):** Este sistema de control es apropiado para sistemas que sean variantes en el tiempo. Aunque dentro de esta categoría podrían colocarse los “PID autoajustables”, el concepto es mucho más amplio, ya que suelen incorporar técnicas de identificación de parámetros por mínimos cuadrados u otras técnicas propias del sistema resultante [17].

- ❖ **Control Neuronal:** Las Redes Neuronales (“Neural Networks”) son estructuras matemáticas que procuran representar la información en forma similar a como se estructura en nuestro cerebro. El objetivo de usar redes neuronales es disponer de un sistema que se comporte como una “caja negra” que pueda emular el comportamiento de un sistema [17].

En la Tabla 1.1 se indica el porcentaje de satisfacción de las distintas técnicas de control.

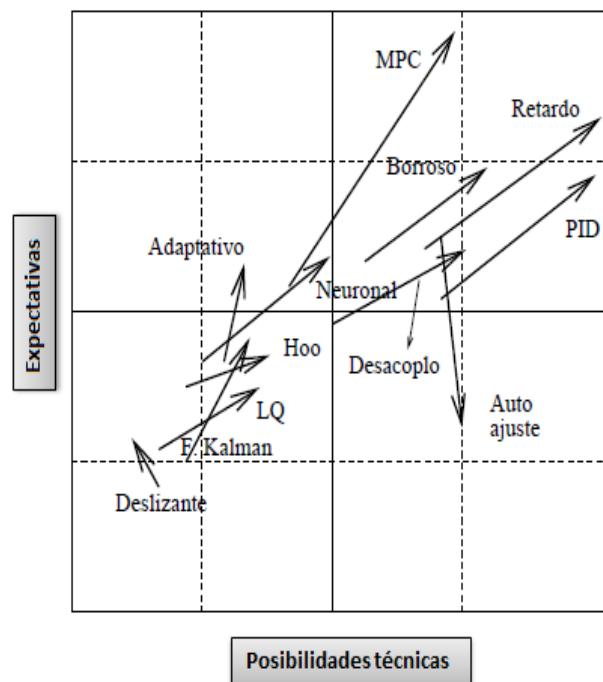
**Tabla 1.1.**

Porcentaje de satisfacción de los controles.

<b>Técnica</b>	<b>1989</b>	<b>2006</b>
Control Predictivo	76	94
PID avanzado	77	89
Compensación de retardo	72	89
Gain-scheduling	78	87
Difuso	67	83
Neuronal	-	69
Desacoplo	64	66
Filtro de Kalman	70	66
Autoajuste	60	65
Observador	67	62
Basado en reglas	43	61
Control adaptivo	50	56
$H^\infty$	-	50

Fuente: [19].

En la Figura 1.17 se indica las expectativas al utilizar distintas técnicas de control avanzado, notando el MPC tiene el mayor incremento.



**Figura 1.17:** Expectativas y posibilidades de las técnicas de control.

Fuente: [19].

Este estado actual y futuras tendencias en el campo del control de procesos industriales indican que el Control Predictivo Basado en Modelo se puede considerar una tecnología suficientemente introducida en la industria y que además sigue despertando muchas expectativas. Estos hechos, unidos a la existencia de campos abiertos tanto en investigación como en temas relacionados con la implementación justifican un estudio más detallado de esta tecnología y motiva la selección del control avanzado en el que se enfoca este proyecto.

### **1.5 Control Predictivo**

El Modelo de Control Predictivo o MPC se desarrolló en base a dos líneas básicas, por un lado primero, a finales de los años setenta surgieron diversos algoritmos que usaban explícitamente un modelo dinámico del proceso para predecir el efecto de las acciones de control futuras en la salida, las cuales eran determinadas minimizando el error predicho sujeto a restricciones de operación.

La optimización se repetía en cada instante de muestreo con información actualizada del proceso. Estas formulaciones eran de naturaleza heurística y algorítmica e intentaban aprovechar el creciente potencial de los computadores digitales por aquella época.

Rápidamente el MPC adquirió gran popularidad en las industrias de procesos químicos principalmente debido a la simplicidad del algoritmo y al uso del modelo de respuesta impulsional o en escalón, que aunque posea muchos más parámetros que las formulaciones en el espacio de estados o función de transferencia suele ser preferido por ser intuitivo y necesitar menos información para identificar. La mayoría de las aplicaciones fueron llevadas a cabo sobre sistemas multivariados.

Los algoritmos utilizados fueron principalmente el IDCOM (Identification-Command) y el DMC (Control con Matriz Dinámica, *Dynamic Matrix Control*).[1]

### 1.5.1 Actualidad.

La situación actual de aplicaciones de MPC en la industria está bien reflejada en la recopilación de Qin y Badgwell, que recoge unas 2200 aplicaciones, principalmente en el sector petroquímico (desde entonces el número de aplicaciones puede estimarse en torno a las 3000). La mayoría de las aplicaciones son en procesos multivariables, registrándose casos como un controlador con 40 entradas y 80 salidas. Sorprendentemente, MPC ha tenido menor impacto en otro tipo de industrias, aunque estudios de 1993 sugieren que unas 20.000 aplicaciones podrían beneficiarse de esta técnica. [19]

El éxito actual del MPC en la industria se debe a tres razones principales:

- ❖ La incorporación de un modelo explícito del proceso en los cálculos permite al controlador tratar con todas las características importantes de la dinámica del proceso.
- ❖ La consideración del comportamiento del proceso a lo largo de un horizonte futuro permite tener en cuenta el efecto de las perturbaciones en realimentación y *pre*-alimentación, permitiendo al controlador conducir la salida a la trayectoria de referencia deseada.
- ❖ La consideración de restricciones en la fase del diseño del controlador evita en lo posible su violación, resultando en un control más preciso en torno al punto óptimo de operación.

La inclusión de restricciones es quizás la característica que más distingue al MPC respecto a otras metodologías. [19]

### 1.6 Control Predictivo Basado en Modelo (MBPC o MPC)

El control Predictivo basado en un Modelo, es un método para resolver problemas de control en el dominio del tiempo. El Control Predictivo se basa en el empleo de modelos dinámicos que relacionan las variables manipuladas y controladas de un proceso. Estos permiten predecir su

evolución futura y calcular, mediante un criterio de optimización, las actuaciones que lleven al proceso a los objetivos deseados [20]. El criterio de optimización se denomina función de costo, y el valor futuro se predice por el modelo dinámico del sistema, el cual se denomina modelo de predicción, el intervalo de tiempo futuro que interviene en la optimización, es denominado horizonte de predicción. El comportamiento futuro del sistema va a depender de las acciones de control que se van a aplicar durante el horizonte de predicción, la aplicación directa de las acciones de control sobre el sistema determina un control en lazo abierto.

Esta realimentación se consigue utilizando la técnica de horizonte deslizante, que va a aplicar las acciones de control secuencialmente, obtenidas durante un periodo de tiempo, después se muestra otro estado del sistema, y se resuelve un nuevo problema de optimización, por lo tanto el horizonte de predicción se va deslizando continuamente en el tiempo.

Una de las características principales del MPC es la formulación abierta y flexible que permite la configuración de varios modelos de predicción, para sistemas lineales o no lineales, sistemas SISO, MISO, MIMO y es importante señalar la consideración de restricciones sobre las señales del sistema, esto genera que el control predictivo sea una de las pocas técnicas que permite controlar sistemas, asociando restricciones como parte del diseño del controlador. Para sistemas lineales, en la ausencia de restricciones, el controlador es una ley de control lineal; mientras que cuando las restricciones están presentes aparece un problema de programación cuadrática. [21]

### **1.6.1 Ventajas del MPC.**

El MPC presenta una serie de ventajas sobre otros métodos, entre las que destacan: [21]

- ❖ Resulta particularmente atractivo para personal sin un conocimiento profundo de control, puesto que los conceptos resultan muy intuitivos, a la vez que la sintonización es relativamente fácil.



- ❖ Cuando se conoce el comportamiento futuro del sistema de control, el sistema puede reaccionar antes que el cambio se haya presentado en la planta, evitando de este modo retardos en la respuesta del proceso.
- ❖ Es una estrategia de control, que maneja sistemas de varias entradas y salidas, también sistemas lineales y no lineales.
- ❖ Se toman las restricciones físicas de actuadores, las restricciones ambientales, restricciones de seguridad como parte del diseño del algoritmo de control.
- ❖ Puede ser usado para controlar una gran variedad de procesos, desde aquéllos con dinámica relativamente simple hasta otros más complejos incluyendo sistemas con grandes retardos, de fases no mínimas o inestables.
- ❖ Permite tratar con facilidad el caso multivariables.
- ❖ Resulta conceptualmente simple la extensión al tratamiento de restricciones, que pueden ser incluidas de forma sistemática durante el proceso de diseño.
- ❖ Es una estrategia flexible, puesto que su implementación se ha realizado en la industria y puede seguir avanzando en forma intuitiva, a medida que se da el sustento teórico para esta técnica de control.

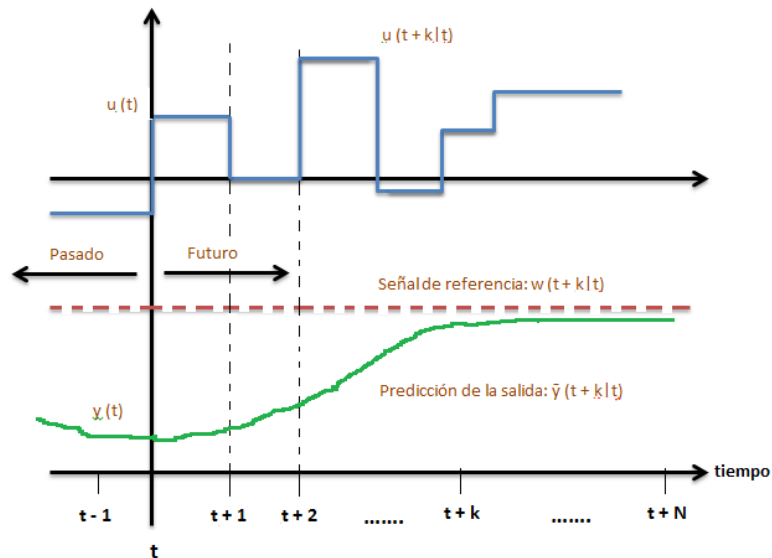
### **1.6.2 Desventajas del MPC.**

Pero, lógicamente, también presenta inconvenientes las cuales se pueden tomar en cuenta: [21]

- ❖ Carga de cálculo necesaria para la resolución de algunos algoritmos.
- ❖ Los algoritmos de control del MPC, requieren un nivel de procesamiento de datos relativamente rápido.
- ❖ Necesidad de disponer de un modelo apropiado del proceso.
- ❖ El rendimiento del MPC estará determinado por la relación cercana que exista entre la planta real y el modelo matemático.
- ❖ El proceso de optimización del MPC requiere un nivel matemático complejo, lo cual genera una carga de computación alta.

### 1.6.3 Estrategia del Control Predictivo.

La metodología de todos los controladores pertenecientes a la familia del MPC se caracteriza por la estrategia siguiente, representada en la Figura 1.18:

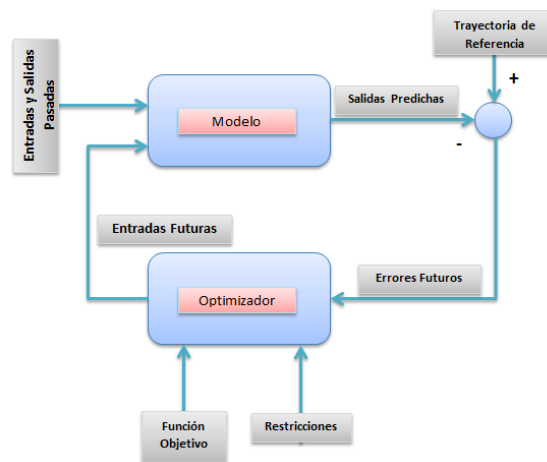


**Figura 1.18:** Estrategia del MPC.

Elaborado por: Curay, D. y Vaca, A.

- ❖ En cada instante  $t$  de tiempo, se predicen las salidas futuras para un horizonte de predicción de valor  $N$ , usando un modelo del proceso a controlar. Estas salidas predichas  $\hat{y}(t+k|t)$  para  $k=1, \dots, N$ , dependen de las entradas y salidas pasadas, cuyos valores son conocidos y las señales de control futuras  $u(t+k|t)$  para  $k=1, \dots, N-1$ , que son las que se calcularán y enviarán al sistema.
- ❖ Las señales de control futuras ( $u$ ) serán calculadas por medio de la función de optimización, la cual se ajustará lo más cercano posible el proceso a la referencia dada  $w(t+k|t)$ .
- ❖ La señal de control  $u(t|t)$  se envía al proceso, mientras que las demás señales de secuencia de control se eliminan. En el siguiente instante de muestreo se calcula la secuencia actualizada, esto se realiza porque ya se conoce el  $y(t+1)$  y se obtiene  $u(t+1|t+1)$ , haciendo uso del concepto de horizonte deslizante.

El control predictivo se enfoca en la optimización y el uso explícito del modelo del proceso, es porque se utilizan valores óptimos anticipados, quienes se encargan que el modelo experimental se ajuste a referencia dada, este tipo de control también se puede aplicar a un sistema multivariables, tratar con procesos no lineales y manejar eficientemente las limitaciones. [20]



**Figura 1.19:** Estructura general de los controladores predictivos.

Elaborado por: Curay, D. y Vaca, A.

#### 1.6.4 Elementos del Control Predictivo.

Todos los controladores predictivos poseen elementos comunes y para cada uno de estos elementos se pueden elegir diversas opciones, dando lugar a distintos algoritmos. Estos elementos son:

- ❖ Modelo de predicción
- ❖ Función objetivo
- ❖ Ley de control

##### a. Modelo de Predicción.

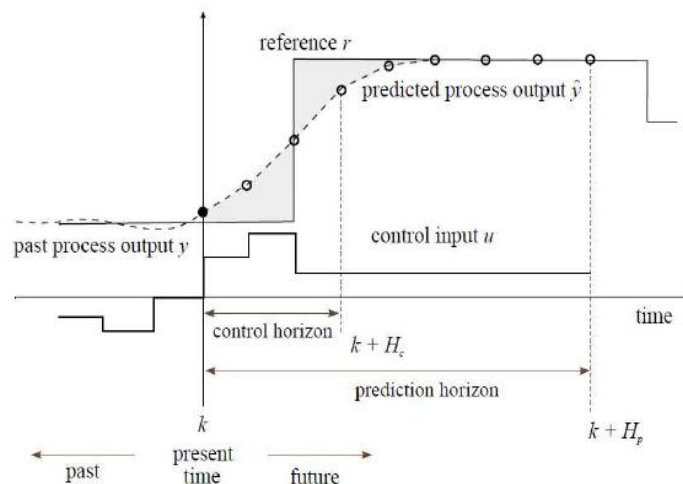
Lo más importante del controlador predictivo basado en un modelo, es que este permita obtener la dinámica del proceso y además de tener un entero conocimiento teórico del proceso. Las salidas futuras procesadas son predichas en el horizonte de predicción ( $H_p$ ) utilizando un modelo del

proceso. Los valores de las salidas predichas  $y(k+1)$  dependen del estado del proceso en tiempo actual ( $k$ ) y de las señales de entrada y salida pasadas; las señales de control  $u(k+1)$  son predichas en el horizonte control, donde  $H_c$  (*horizonte de control*)  $\leq H_p$  (*horizonte de predicción*).

La señal de control es manipulada solo en el horizonte control y se mantiene constante a través del tiempo, Figura 1.20.

**Horizonte de Predicción.-** Es el intervalo de tiempo futuro que se considera en la optimización para la señal de control.

**Horizonte de control:** Número de movimientos calculados para cada variable manipulada, donde el horizonte de control es menor al horizonte de predicción.



**Figura 1.20:** Principio básico del control predictivo basado en un modelo.

Fuente: [20].

Existen diferentes estrategias del control predictivo basado en un modelo donde se puede usar distintos modelos que representan la relación de las salidas con las entradas medibles, algunas de las cuáles serán las variables manipuladas y otras se pueden considerar como perturbaciones medibles, que pueden ser compensadas por la realimentación. Este modelo puede ser lineal, no lineal, en tiempo continuo, en tiempo discreto, puede ser expresado como función de transferencia o en el espacio de estados, puede ser un modelo multivariable o mono variable, los modelos más usados son:

- ❖ **Respuesta Impulsional.** La salida viene relacionada con la entrada por la ecuación:

$$y(t) = \sum_{i=1}^N h_i U(t-i) \quad (1.1)$$

Donde  $h_i$  de la Ecuación 1.1, son los valores muestreados obtenidos al someter al proceso a un impulso unitario de amplitud igual al periodo de muestreo. Esta suma es truncada y solo se consideran  $N$  valores (solo para procesos estables y sin integradores).

- ❖ **Respuesta ante escalón.** Para sistemas estables se tiene la respuesta truncada que será:

$$y(t) = y_o + \sum_{i=1}^N g_i \Delta u(t-i) \quad (1.2)$$

Donde  $g_i$  de la Ecuación 1.2, son los valores muestreados ante la entrada en escalón y  $\Delta u(t) = u(t) - u(t-1)$ . El " $y_o$ " de la ecuación 1.2 es valor de estado inicial que puede ser cero.

- ❖ **Función de Transferencia.** Utiliza el concepto de función de transferencia donde la salida viene dada por:

$$y(t) = \frac{B z^{-1}}{A z^{-1}} u(t) \quad (1.3)$$

Para este tipo de método se debe conocer el proceso, además se aplica en métodos inestables y se necesitan pocos parámetros.

- ❖ **Espacio de Estados.** Tiene la siguiente representación:

$$x(t) = A x(t-1) + B u(t-1) \quad (1.4)$$

$$y(t) = C x(t) + D u(t) \quad (1.5)$$

Este método tiene una ventaja, ya que se puede aplicar para un control multivariable, ya que permite analizar la estructura interna del proceso. [20]

### b. Función Objetivo.

Conocida como función de costo, es aquella que indica el criterio a optimizar. Es una función definida positiva que expresa el costo asociado a la evolución del proceso en el intervalo del horizonte predicción  $N$ . La siguiente función expresa el criterio de optimización, de la cual se obtienen valores positivos por ser una función cuadrática:

$$J(N1, N2, Hc) = \sum_{j=N1}^{N2} \delta(j) [y(t+j) - w(t+j)]^2 + \sum_{j=1}^{Hc} \gamma(j) \Delta u(t+j-1)^2 \quad (1.6)$$

Donde  $w(t+j)$  es la trayectoria de la referencia futura, la cual se conoce con anterioridad, con lo que se evitan los efectos de retardo en la respuesta del proceso;  $y(t+j)$  es la salida futura, la diferencia que existe entre la salida y la referencia es el error;  $\delta(j)$  y  $\gamma(j)$  son valores de penalización, que ponderan el comportamiento futuro.  $N1$  y  $N2$  son los horizontes mínimo y máximo de costo y  $Hc$  es el horizonte de control. Los primeros valores de la secuencia son para minimizar la diferencia de la salida y la referencia, mientras que los valores consecuentes representan un esfuerzo en el control.

Para sistemas con tiempo de retardo de “ $n$ ” muestras, solo las salidas para el tiempo  $j+n$  son consideradas en la función objetivo, porque las salidas antes de este tiempo no pueden ser influenciadas por la señal de control, debido que la señal de control es cero para los tiempos pasados. Un razonamiento similar para sistemas de fase no mínima. [20]

### c. Optimización.

Existen diversas formas de tomar decisiones óptimas: por experiencia, probando todas las opciones o analizando el problema y formulándolo como un problema matemático. En términos generales, se desea conocer que valores deberían tener estas variables para que la expresión matemática tenga el mayor valor numérico posible (máximo) o el menor valor numérico posible (minimización). A este proceso general de maximización y minimización se lo conoce como optimización.

La optimización sirve para encontrar la opción que proporciona el mejor resultado, entre varias posibles opciones.

#### ❖ **Elementos de la Optimización.**

Los problemas de optimización tienen los siguientes elementos:

- ❖ Criterio a optimizar o función objetivo
- ❖ Variables de decisión
- ❖ Restricciones sobre las variables de decisión.

**Función objetivo:** Es la expresión matemática que enuncia la medida cuantitativa del funcionamiento del sistema que se desea optimizar (maximizar o minimizar).

**Variables de decisión:** Representan las decisiones que se pueden tomar para afectar el valor de la función objetivo. Estas variables se pueden ajustar para obtener un óptimo resultado.

**Restricciones:** Constituyen el conjunto de relaciones, expresadas mediante ecuaciones e inecuaciones, que ciertas variables están obligadas a satisfacer. Existen restricciones de igualdad y desigualdad.

#### ❖ **Restricciones.**

Indican los límites dentro de los cuales debe funcionar el sistema. El desarrollo de las señales del sistema no debe exceder sus límites impuestos, ya sean por límites físicos o bien por motivos de seguridad.

Las restricciones tienen su importancia ya que se aplican dentro del proceso de control, porque se las utiliza como medida seguridad además de no incrementar gastos económicos sino reducirlos [20]. 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:

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

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

$$y_{min} \leq y \leq y_{max}$$

(1.7)

### ❖ Principio del Horizonte Deslizante.

Consiste en aplicar las actuaciones obtenidas durante un periodo de tiempo, tras el cual se muestrea el estado del sistema y se resuelve un nuevo problema de optimización. De esta manera, el horizonte de predicción se va deslizando a lo largo del tiempo.

Solo la señal de control  $u(k)$  es aplicada para el proceso. En el siguiente instante de muestreo, la salida del proceso  $y(k+1)$  es disponible. Para el siguiente instante de muestreo se repite la optimización y la predicción con la actualización de los valores. A esto se le llama el principio de horizonte en retroceso.

La acción de control  $u(k+1)$  calcula en el paso del tiempo  $k+1$  será generalmente diferente de un cálculo en el paso de tiempo  $k$ , desde cuándo se ha actualizado la información acerca del proceso es disponible. Este concepto es similar a la estrategia de control en lazo abierto. [20]

### d. Obtención de la ley de control.

Para obtener los valores de la señal de control  $u(k+1)$  es necesario minimizar la función de costo  $J$  (Ecuación 1.6), pero antes se debe obtener primero las salidas predichas  $y(k+1)$  en función de los valores actuales y pasados, tanto de la entrada como de la salida, de igual forma las señales futuras de control, mediante el uso del modelo escogido y reemplazando en la función de costo, obteniendo una expresión que al minimizar da como resultado los valores buscados [21]. Al momento de minimizar, la ley de control produce una mejora en la robustez del sistema y consiste en que



después de un intervalo donde el horizonte de control es menor al horizonte de predicción, la señal de control permanece constante.

En esta estructura de la ley de control se aplica el uso del concepto de horizonte de control, que tras un cierto intervalo no existe variación en la señal de control aplicado, lo que equivale a dar pesos infinitos a los cambios en control, los cuales ayudan a lograr el valor deseado. [20]

### **1.7 Controladores de Automatización Programable (PAC)**

Un PAC es una tecnología industrial orientada al control automatizado avanzado, al diseño de equipos para laboratorios y a la medición de magnitudes análogas.

El PAC se refiere al conjunto formado por un controlador (una CPU típicamente), módulos de entradas y salidas, y uno o múltiples buses de datos que lo interconectan todo. [23]

Este controlador combina eficientemente la fiabilidad de control de un autómatas o PLC junto a la flexibilidad de monitorización, cálculo y desempeño de un computador industrial.

Los PACs pueden utilizarse en el ámbito investigador y de laboratorios, pero es sobre todo en el industrial, para control de máquinas y procesos, donde más se utiliza.

Se puede destacar a continuación las siguientes aplicaciones: [23]

- ❖ Múltiples lazos cerrados de control independientes.
- ❖ Lazos de control robusto.
- ❖ Adquisición de datos de precisión
- ❖ Análisis matemático y memoria profunda
- ❖ Monitorización remota
- ❖ Visión artificial
- ❖ Control de movimiento y robótica
- ❖ Seguridad controlada.

### 1.7.1 Diferencias y similitudes entre un PAC y un PLC.

Los PACs y PLCs tienen varias cosas en común. Internamente, ambos incluyen una fuente de potencia, un CPU, un plano trasero o dispositivo de E/S, y módulos. Tienen registros de memoria que reflejan los canales de E/S individuales en los módulos. Sin embargo, las siguientes diferencias resultan muy significativas. En su estudio de “Generalidades de los Controladores Lógicos Programables a Nivel Mundial”, ARC identificó 5 características principales en los PAC:

- ❖ Funcionalidad de dominio múltiple, al menos dos de lógica, movimiento, control PID, y proceso en una sola plataforma
- ❖ Plataforma de desarrollo sencillo de disciplina múltiple incorporando etiquetas comunes y una base de datos sencilla para tener acceso a todos los parámetros y funciones
- ❖ Herramientas de software que permiten diseñar flujo del proceso a través de varias máquinas o unidades de proceso, junto con el IEC 61131-3, guía del usuario y administración de datos
- ❖ Arquitecturas modulares, abiertas que reflejan las aplicaciones industriales a partir de un despliegue de maquinaria en fábricas en plantas de proceso
- ❖ Uso de estándares de la industria para interfases en red, lenguajes, etc., como búsquedas TCP/IP, OPC y XML, y SQL. [23]

### 1.7.2 Ventajas de los PACs en la adquisición de datos.

Una ventaja de los PAC al compararse con los PLCs, es la habilidad para procesar y desempeñar medidas complejas.

Con esta característica, puede combinar diferentes sistemas de adquisición de datos como frecuencias, formas de onda, voltajes, corrientes, control de movimiento e incluso, adquisición de imágenes.

Esto crea un nivel sin precedentes de manipulación y estandarización en términos del tipo de señales que pueden manipularse y procesarse. Los

PACs ofrecen cientos de funciones para procesar, analizar y extraer información de estas señales. [23]

En la Tabla 1.2 se indica el cuadro comparativo entre las dos tecnologías de control.

**Tabla 1.2**

Cuadro comparativo entre PLC vs. PAC.

Características	PLC	PAC	PC Estándar
Soporta shocks eléctricos y vibración	√	√	X
Seguridad y estabilidad	√	√	X
Rango de temperatura industriales	√	√	X
Trabajo en tiempo real	√	√	X
Entradas de fuente de poder redundantes	√	√	X
Procesador de punto flotante	X	√	√
Memoria no volátil	X	√	√
Conectividad a Ethernet vía WEB	X	√	√
Capacidad de administración de recursos	X	√	√
Capacidad ilimitada de lazos de control	X	√	√

Fuente: [23].

### 1.7.3 NI Single-Board RIO de National Instruments.

La plataforma NI Single-Board RIO combina dispositivos embebidos de despliegue que tienen un procesador en tiempo real, arreglo de compuertas programable en campo (FPGA) y E/S analógica y digital en una sola tarjeta programada con el software NI LabVIEW. Los dispositivos NI Single-Board RIO están diseñados para aplicaciones OEM embebidas de control y adquisición y de alto volumen que requieren alto rendimiento y fiabilidad. El sistema base NI Single-Board RIO es un sistema embebido y reconfigurable de control y adquisición que consiste en un procesador en tiempo real,

FPGA programable por el usuario y E/S integrada y periféricos como USB, RS232, CAN y Ethernet, todo en una sola tarjeta.

NI Single-Board RIO ofrece acceso directo de hardware a los circuitos de E/S de cada módulo de E/S usando funciones elementales de E/S de NI LabVIEW FPGA. Cada módulo de E/S contiene acondicionamiento de señales integrado y conectores de terminal de tornillo, BNC o D-Sub.

LabVIEW y NI Single-Board RIO hacen a la tecnología FPGA más accesible, así se puede definir un propio circuito de control y reducir la complejidad y los costos asociados con hardware tradicional y personalizado. [25]

#### **a. Sistema de Desarrollo de LabVIEW.**

LabVIEW es un entorno de programación gráfica usado por miles de ingenieros e investigadores para desarrollar sistemas sofisticados de medida, pruebas y control usando íconos gráficos e intuitivos y cables que parecen un diagrama de flujo.

Ofrece una integración incomparable con miles de dispositivos de hardware y brinda cientos de bibliotecas integradas para análisis avanzado y visualización de datos, todo para crear instrumentación virtual.

La plataforma LabVIEW es escalable a través de múltiples objetivos y sistemas operativos, desde su introducción en 1986 se ha vuelto un líder en la industria. Con los módulos LabVIEW Real-Time y FPGA, puede desarrollar, depurar y desplegar aplicaciones al microprocesador y FPGA en el NI Single-Board RIO. [25]

#### **b. Módulo LabVIEW Real-Time.**

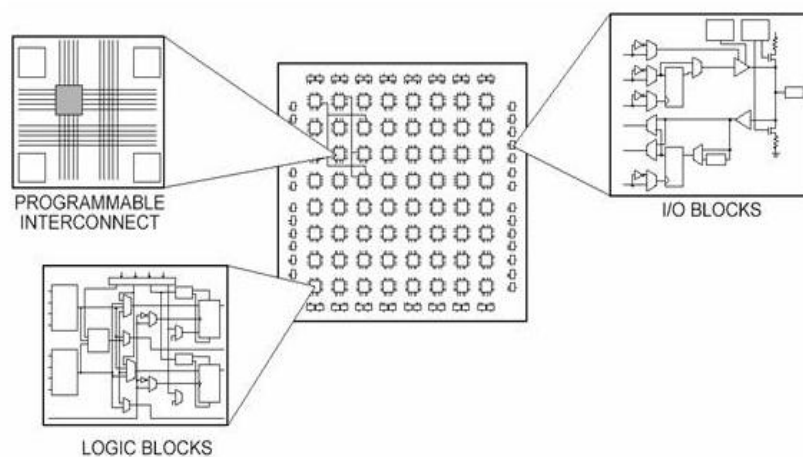
La tecnología en tiempo real ofrece rendimiento confiable y determinístico para sus aplicaciones de tiempo crítico. Utilice el Módulo LabVIEW Real-Time para desarrollar y desplegar sistemas complejos en tiempo real, de manera rápida y eficiente al microprocesador del NI Single-Board RIO.

### c. Módulo LabVIEW FPGA.

Con el Módulo LabVIEW FPGA, se puede crear hardware personalizado de medidas y control usando programación gráfica sin tener experiencia en lenguajes de descripción de hardware de bajo nivel o diseño a nivel de tarjeta.

Se puede usar este hardware personalizado para rutinas únicas de temporización y disparo, control de ultra alta velocidad, establecer interfaz con protocolos digitales, procesamiento digital de señales (DSP), RF y comunicaciones y muchas otras aplicaciones que requieren fiabilidad de hardware de alta velocidad y determinismo estrecho.

Usar el Módulo LabVIEW FPGA para desarrollar, compilar y desplegar aplicaciones al FPGA interno del NI Single-Board RIO en la Figura 1.21 se muestra las partes de un FPGA.



**Figura 1.21:** Diferentes partes de un FPGA.

Fuente: [25].

### 1.7.4 Instrumentos basados en FPGA pueden acelerar las mediciones.

En aplicaciones de pruebas automatizadas de alto rendimiento, el ancho de banda y la potencia de procesamiento del CPU pueden limitar la velocidad de prueba. Usted puede resolver estas limitaciones al integrar tecnología de arreglo de compuertas programables en campo (FPGA) muy

de cerca con hardware de sistema de E/S para tiempo real, procesamiento continuo y medición de datos a medida que son adquiridos.

La alta velocidad y rendimiento de procesamiento de los FPGAs permiten nuevas capacidades para aplicaciones de prueba de alto rendimiento. [25]

#### ❖ **Procesamiento de cada muestra.**

Los FPGAs pueden procesar las muestras de entrada más rápida que el ADC, por lo que es posible garantizar que cada muestra es vista y procesada de manera continua. En el sistema de ejemplo, el NI 5734 está adquiriendo datos siempre y transfiriéndolos de manera continua al FPGA. Al implementar la lógica de disparo de manera digital en el FPGA, se puede lograr un tiempo de reconfiguración de cero muestras, lo cual garantiza que ninguna muestra se pierda.

#### ❖ **Procesamiento en línea para reducción de datos.**

Un método efectivo para reducir el ruido de una señal periódica es promediar múltiples registros de datos. Este procesamiento es implementado en el FPGA, el cual utiliza un búfer para promediar las muestras a medida que son adquiridas. Debido a la baja latencia y alto rendimiento de este algoritmo ejecutándose en el FPGA, el resultado está listo en microsegundos.

#### ❖ **Procesamiento en paralelo.**

En el Módulo de NI LabVIEW FPGA, cada instancia de VI reentrante se sintetiza en el FPGA utilizando recursos únicos para verdadero procesamiento en paralelo. Esto le da a su sistema de prueba un rendimiento cuatro veces mayor que la adquisición y procesamiento en una sola instancia. [25]

### 1.7.5 SBRIO 9636.

El dispositivo embebido de control y adquisición NI sbRIO-9636 integra un procesador en tiempo real, un a FPGA reconfigurable por el usuario y E/S en una sola tarjeta de circuito impreso (PCB). Tiene un procesador industrial de 400 MHz, un FPGA Xilinx Spartan-6 LX45, 16 canales de una sola terminal / 8 canales diferenciales de entrada analógica de 16 bits a 200 kS/s y cuatro canales de salida analógica de 16 bits y 28 líneas de E/S digital (DIO) en la Figura 1.22 se presenta tarjeta de la SBRIO 9636 y en los Anexos B. [24]



**Figura 1.22:** SBRIO 9636.

Fuente: [24]

- ❖ Ofrece un rango de temperatura de operación ambiente y local de -40 a 85 °C junto con un rango de entrada de suministro de potencia de 9 a 30 VDC. Brinda 256 MB de DRAM para operación embebida y 512MB de memoria no volátil para almacenamiento de programas y registro de datos.

- ❖ Tiene un puerto de Ethernet integrado de 10/100 Mbits/s que usted puede usar para llevar a cabo comunicación programática en la red y Web integrada (HTTP) y servidores de archivos (FTP). También tiene puertos USB, CAN, SDHC, seriales RS232 y seriales RS485 para controlar dispositivos periféricos.

El sbRIO-9636 está diseñado para ser fácilmente embebido en aplicaciones de alto volumen que requieren flexibilidad, fiabilidad y alto rendimiento, son ideales para aplicaciones de volumen bajo y medio y rápida generación de prototipos. [24]

#### a. Resumen de las especificaciones de la SBRIO-9636

**Tabla 1.3**

Especificaciones generales SBRIO-9636.

General	
<b>Producto</b>	sbRIO-9636
<b>Familia de Productos</b>	Board-level Embedded
<b>Número de Parte</b>	000000-00
<b>Sistema Operativo/Objetivo</b>	Real-Time
<b>Soporte para LabVIEW RT</b>	Sí
<b>Formato Físico</b>	Single-Board RIO
<b>Certificaciones de Productos</b>	RoHS , UL - Product Safety
<b>Compatibilidad con RoHS</b>	Sí
<b>Conformal Coated</b>	No
Entrada Analógica	
<b>Canales</b>	16
<b>Resolución</b>	16 bits
<b>Velocidad de Muestreo</b>	200 kS/s
<b>Máx. Voltaje de Entrada Analógica</b>	10 V
<b>Rango de Voltaje Máximo</b>	-10 V - 10 V
Salida Analógica	

Continua





<b>Canales</b>	4
<b>Resolución</b>	16 bits
<b>Máx. Voltaje de Entrada Analógica</b>	10 V
<b>Rango de Voltaje Máximo</b>	-10 V - 10 V
<b>E/S Digitales</b>	
<b>Canales Bidireccionales</b>	28
<b>Niveles Lógicos</b>	3.3 V
<b>Contadores/Temporizadores</b>	
<b>Número</b>	de 0
<b>Contadores/Temporizadores</b>	
<b>FPGA Reconfigurable</b>	
<b>FPGA</b>	Spartan-6 LX45
<b>Especificaciones Físicas</b>	
<b>Longitud</b>	15.4 cm
<b>Ancho</b>	10.3 cm
<b>Temperatura de Operación</b>	-40 °C
<b>Temperatura de Operación</b>	85 °C
<b>Temperatura de Almacenamiento</b>	-40 °C
<b>Temperatura de Almacenamiento</b>	85 °C

Fuente: [24].

## **CAPÍTULO 2**

### **DISEÑO Y SIMULACIÓN DEL CONTROL PREDICTIVO EN LA PLANTA DE PRESIÓN**

#### **2.1 Descripción de la planta de presión**

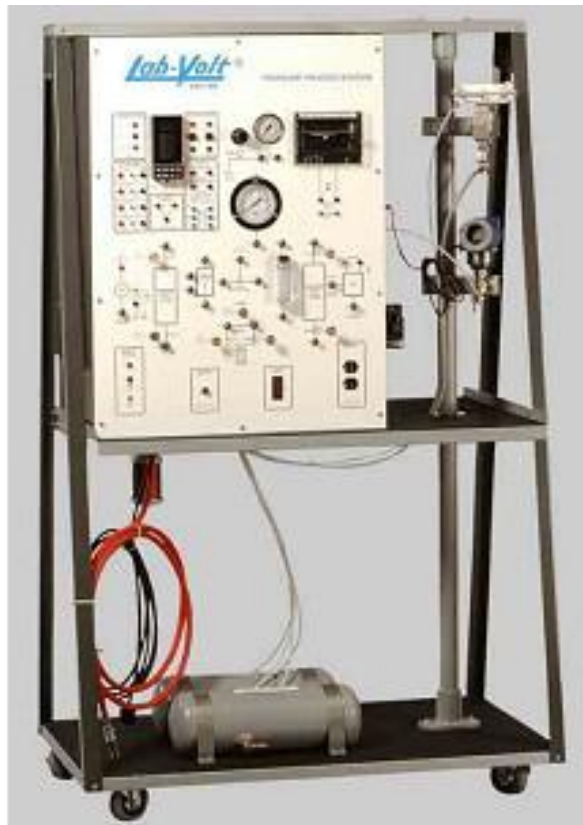
Para el diseño de un control predictivo, se debe realizar un estudio detallado al sistema modular de la planta de presión, con el objetivo de controlar un rango de presión variable (0-100 PSI) según la necesidad del operario, usuario o personal a cargo de la planta.

Este estudio permitirá conocer las unidades de trabajo tanto de las entradas como de las salidas, el rango de operación de la planta, perturbaciones, etc., con los cuales se obtendrá el modelamiento óptimo de la planta.

La estación de Proceso de Presión, Modelo 3501, consta de dos tanques de 7.5-1 (2 galones americanos), un regulador de la presión atmosférica, una válvula de descarga conectada a un silenciador, una válvula neumática para el control de flujo de aire.

Todos los dispositivos tienen sus conexiones en el panel de control principal de la estación. Cada uno de los dispositivos permite a los estudiantes conectarlos independientemente según la configuración que se desee utilizar (ver Figura 2.1).

También incluye un indicador graduado de kPa/Psi, un transmisor de presión electrónico, un conversor de corriente a presión (I/P), ver Figura 2.1.



**Figura 2.1:** Estación de proceso de Presión.

Elaborado por: Curay, D. y Vaca, A.

### **Convertor de voltaje a corriente (V/I)**

Los acondicionadores de señal de tensión de DRF-VDC y DRF-VAC admiten tensiones CC y CA respectivamente y proporcionan una salida aislada de 0 a 10 Vcc o de 4 a 20 mA. El dispositivo se muestra en Figura 2.2 y en los Anexos A se puede visualizar todas las configuraciones respectivas del convertor.

Los modelos están disponibles con tres opciones de alimentación diferentes: 24 Vcc, 120 Vca y 240 Vca.

Los modelos DRF-VDC y DRF-VAC son idóneos para aplicaciones industriales. Todos los modelos se montan en un riel DIN estándar de 35 mm y ofrecen un aislamiento galvánico entre la entrada, la salida y la alimentación.



**Figura 2.2:** Conversor de voltaje a corriente.

**Características:**

- Precisión : <0,2% escala completa
- Linealidad : <0,1% escala completa
- Deriva térmica: 150 ppm/°C típica (máx. <200 ppm/°C)
- Tiempo de respuesta (modelos de salida de señal Vdc): < 70 mseg. (90% de la señal) a 20Hz -3dB
- Tiempo de respuesta (modelos de entrada de señal Vac ) : <250 mseg. (90% de la señal).
- Impedancia de entrada: 1 MΩ para rangos de < 1 v, 10 MΩ para rangos de >1v
- Protección por encima del rango: 1.000v para rangos superiores a 100v ,500v para rangos inferiores o iguales a 100v.

La SBRIO-9636 se caracteriza por tener entradas y salidas únicamente de voltaje, esta característica es una desventaja al momento de realizar una acción directa sobre la válvula de control debido a que ésta válvula de control se maneja mediante presión y por ende la estación posee un conversor de corriente a presión (I/P) con la cual maneja el porcentaje de apertura y cierre de la válvula. Por este motivo el uso de un conversor de

voltaje a corriente (V/I) es primordial para actuar directamente sobre la válvula de control.

### **Resistencias de precisión.**

El juego de resistores de corriente SCXI es un paquete de cuatro resistores de precisión para medidas de entradas de corriente de 0-20 mA y 4-20 mA. El juego incluye cuatro resistores de alta precisión de 249 Ohm, 0.1%, 5 ppm, 0.25 W. Se las muestra en la Figura 2.3.



**Figura 2.3:** Resistencias de 249 Ohm.

Elaborado por: Curay, D. y Vaca, A.

## **2.2 Modelamiento de la planta de Presión**

El paso previo a la modelación de la planta de presión es la identificación de sistemas, la cual se lo realiza por medio de la herramienta MATLAB gracias a su toolbox denominado IDENT, que obtiene el modelo matemático de manera sencilla, representándolas en función a ecuaciones de estado debido a que este es el tipo de expresión matemática que se utiliza posteriormente para el control predictivo en el software LABVIEW.

Se utiliza las ecuaciones de estado para conocer el comportamiento del sistema (planta de presión), por la ventaja que presenta frente a otras maneras de expresión matemática. Para la identificación de este sistema se

observa la respuesta de salida que se obtiene del proceso ante unas determinadas entradas, es decir se lo realiza en lazo abierto sin retroalimentación o control. La identificación se realiza mediante el siguiente orden:

- ❖ Obtención de datos.
- ❖ Tratamiento previo de los datos.
- ❖ Elección de la estructura de modelo.
- ❖ Validación del modelo.

### **2.2.1 Obtención de datos.**

La obtención de datos para el modelado mediante la identificación de sistemas se lo realiza de manera experimental, puesto que así permite conocer las entradas y salidas del sistema que son necesarias para obtener un modelo satisfactorio del proceso bajo estudio. El MPC (Control Predictivo Basado en Modelo) tiene como principal característica el uso de un modelo matemático para realizar el control y así predecir las señales futuras, entonces es primordial que los datos obtenidos sean los adecuados y contengan la suficiente información para obtener un modelo matemático óptimo y así evitar un controlador inestable e incluso que el funcionamiento este por debajo de los controladores tradicionales.

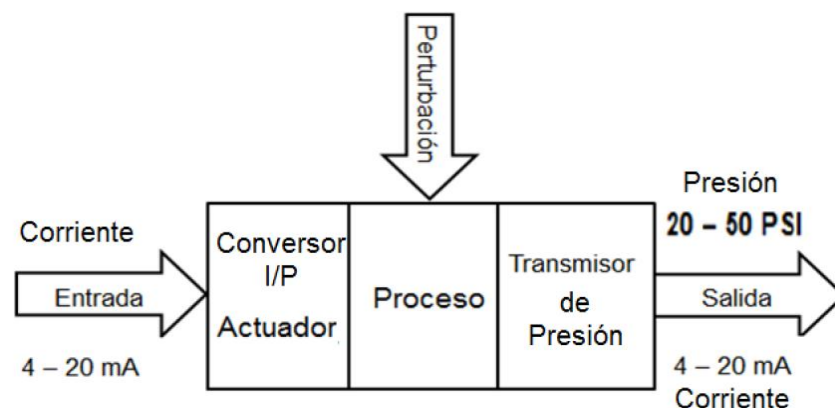
Por esta razón se debe definir las señales a medir, el periodo de muestreo a utilizar, el tipo de entrada más adecuada, el número de datos a almacenar, etc.

#### **a. Señales a medir.**

La determinación de las señales que afecta al sistema es el primer paso para la obtención de la entrada y salida, además deben ser medibles y cuantificables debido a que existen señales que afectan directamente a la salida pero no se pueden actuar sobre ellas, si estas señales se pueden medir se deben considerar como entradas caso contrario se las deben tomar como perturbaciones.

La capacidad de controlar múltiples entradas y múltiples salidas (MIMO) que posee el control predictivo hace más fácil la manipulación de  $n$  entradas que influyan en el comportamiento de la o las salidas del sistema, con el cual se puede garantizar el rendimiento óptimo de éste controlador.

En la planta de presión se toma en consideración un sistema de entrada y salida simple (SISO), debido a que se tiene una sola entrada y una sola salida, siendo la entrada el conversor de corriente a presión que respectivamente provocará el cierre y apertura de la válvula de control (la cual se denomina actuador), mientras que la salida será proporcionada por el transmisor el cual se ve afectado por el cambio en el actuador, en la Figura 2.4 se observa la distribución tanto de la entrada como de la salida del proceso.



**Figura 2.4:** Diagrama de bloques para la adquisición de datos.

Elaborado por: Curay, D. y Vaca, A.

#### **b. Definición de rangos del sistema.**

Se analiza los rangos válidos del proceso, es decir, el límite inferior y superior en las variables de entrada y salida del sistema. Los rangos vienen definidos por el hardware, software y las condiciones físicas del sistema.

#### **Hardware.**

La primera limitación se da en el hardware del proceso, debido a que la entrada y salida ya identificadas en el punto anterior, están limitadas en el

rango de 4 – 20 mA. La entrada definirá la apertura y cierre del actuador (0 – 100%), mientras tanto la salida será el rango configurado en el transmisor de presión con el que se piensa trabajar (20 – 50 PSI).

La segunda limitación está dada por el dispositivo utilizado para la adquisición de datos debido a que funciona con entrada y salida de voltaje, pero no es problema debido a que se utiliza elementos capaces de transformar corriente a voltaje. Todas las conversiones para la visualización correcta de cada variable serán implementadas a nivel de software.

### **Software.**

Se utiliza el software LabVIEW para la adquisición de datos y el software MATLAB para la identificación del sistema, por tanto, no existe ninguna limitación en cuanto a los programas informáticos a utilizar.

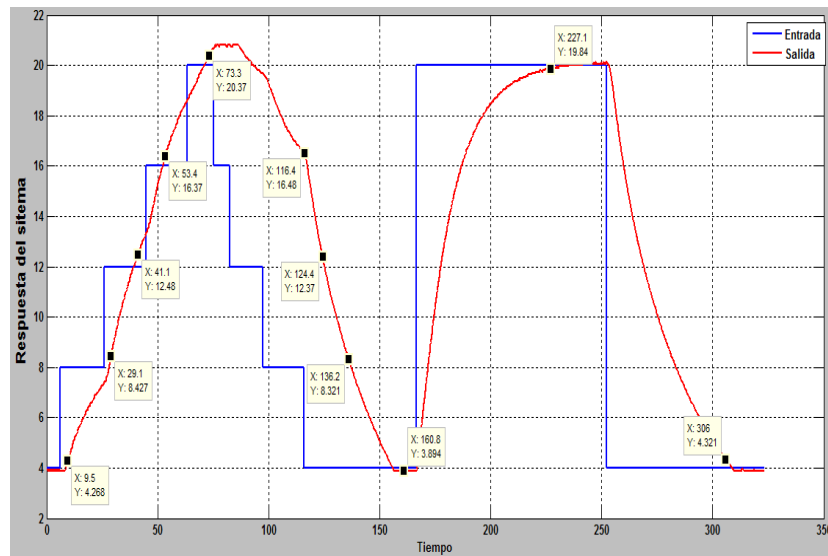
### **Rangos físicos del sistema.**

No existe ninguna restricción tanto para la entrada y salida del sistema:

- ❖ Cualquier valor del rango de entrada provoca un cambio a la salida. Pero en límite inferior y superior de la entrada ocasiona que la salida se sature, es decir, que sobrepasa el límite permitido en el transmisor provocando así que la salida no responda inmediatamente a un cambio en la entrada.  
Esta consideración es tomado en cuenta únicamente en los límites, más no en el resto del rango donde la salida (20 a 50 PSI) responde inmediatamente.
- ❖ El rango de salida no posee ninguna restricción, pues el rango de presión establecido alcanza todo el rango de corriente limitado por hardware.

En la Figura 2.5, se puede apreciar el cambio de la salida ante diferentes valores del rango de la entrada.





**Figura 2.5:** Validación de rangos del sistema.

Elaborado por: Curay, D. y Vaca, A.

Como se puede apreciar cualquier valor dentro del rango de la entrada tiene efecto en la salida. En la Tabla 2.1 se puede observar los rangos de validación del sistema con los cuales se va a trabajar.

**Tabla 2.1**

Rangos del sistema.

Rangos del sistema			
	Entrada	Salida	Salida(Presión)
<b>Límite inferior</b>	4 mA	4 Ma	20 PSI
<b>Límite</b>	20 mA	20 mA	50 PSI

Elaborado por: Curay, D. y Vaca, A.

### c. Periodo de muestreo.

La implementación del controlador se lo realiza de manera digital, es decir que el análisis, modelo matemático y adquisición de datos se efectúan de manera discreta, por tanto la determinación de un periodo de muestreo ( $T_s$ ) adecuado es de suma importancia para la identificación de sistemas del proceso de presión.

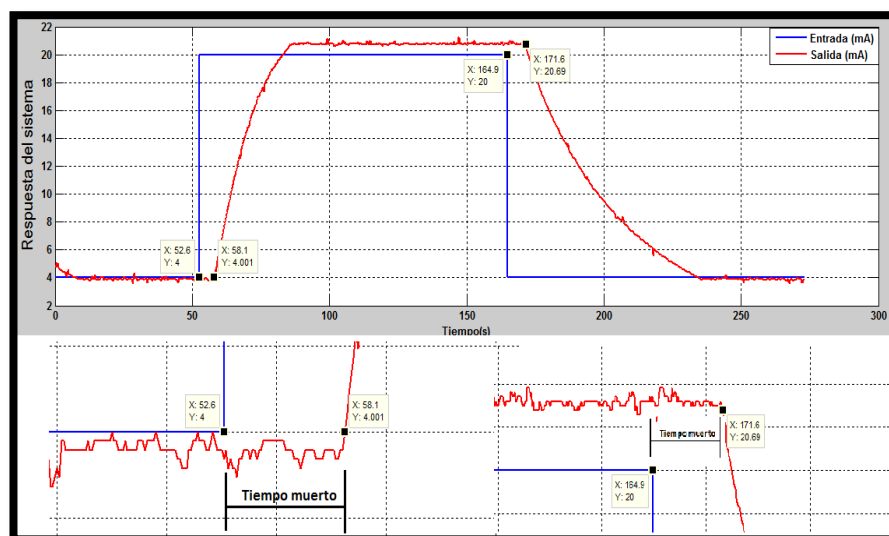
### Tiempo muerto de respuesta del sistema.

El tiempo muerto de respuesta del sistema para este experimento, es aquel tiempo que le toma reaccionar al proceso desde que hubo un cambio en la entrada.

Este tiempo es un parámetro muy importante a considerar, pues de este valor depende el resto del experimento así como el diseño del controlador mismo, ya que proporciona una limitante en cuanto a la velocidad de control, puesto que cualquier cambio en el actuador con tiempos menores al tiempo muerto no afectará a la salida y por lo tanto la acción de control será innecesaria y nada práctica.

Para determinar el tiempo muerto, se realiza un cambio instantáneo en la entrada del sistema y se toma el tiempo hasta tener alguna respuesta en la salida, como se ilustra en la Figura 2.6.

Como la mayoría de procesos no son lineales, el tiempo muerto varía en cada punto del sistema, por tanto, no será el mismo valor en un cambio ascendente que en uno descendente, y dependerá también del valor actual en el que se encuentre la variable a medir.



**Figura 2.6:** Determinación del tiempo muerto.

Elaborado por: Curay, D. y Vaca, A.

El tiempo muerto ( $t_m$ ) se obtiene de la diferencia del tiempo en el cual el sistema responde ( $t_f$ ) y el tiempo al cual se realizó el cambio ( $t_i$ ), como se muestra en la Ecuación 2.1.

$$t_m = t_f - t_i \quad (2.1)$$

El resultado del tiempo muerto se realiza del límite superior e inferior aplicando la Ecuación 2.1.

Tiempo muerto del límite inferior:

$$tm = 58.1 - 52.6 = 5.5 \text{ seg.}$$

Tiempo muerto del límite superior:

$$tm = 171.6 - 164.9 = 6.7 \text{ seg.}$$

Estos tiempos muertos son calculados cuando la salida está saturada y el proceso toma tiempo en reaccionar. Pero cuando se evita esta saturación los tiempos muertos disminuyen notablemente, tal como se muestra en la Tabla 2.2, además se toman tiempos muertos de diferentes cambios en la entrada los cuales estarán dentro del rango.

**Tabla 2.2**

Tiempos muertos del sistema.

Valor inicial (mA)	Valor final (mA)	Tiempo muerto (s)
4	20	0.5
20	4	0.5
4	12	0.2
12	20	0.1

Continua 

20	12	0.2
12	4	0.1
Máximo		0.5

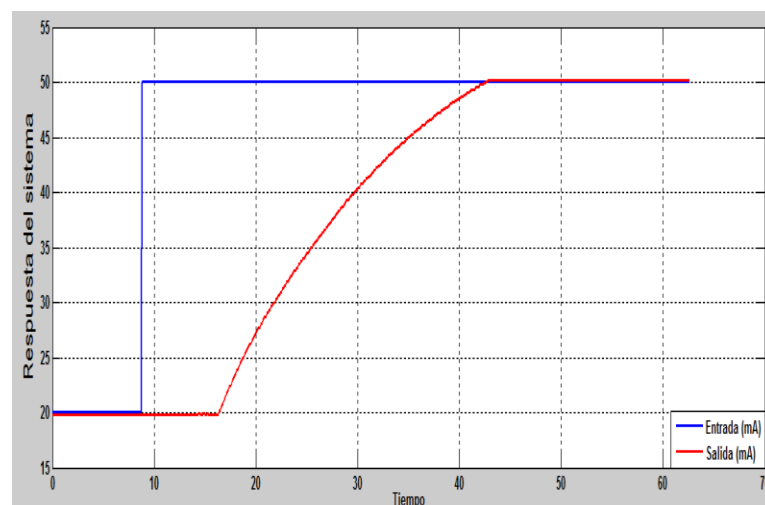
Elaborado por: Curay, D. y Vaca, A.

Como se puede observar evitando la saturación en la salida, el tiempo muerto en los límites del rango no es una restricción para el control debido a que la reacción del sistema es inmediata.

### Dinámica del sistema.

La dinámica del sistema se considera como la variación de la salida a través del tiempo, sin que tenga un cambio en la señal de entrada, es decir, en estado estable.

En la Figura 2.7 se observa que la dinámica del sistema a la salida es normal en el tiempo y no se tiene variaciones de la misma. Además como no se obtiene limitaciones con el tiempo muerto se puede determinar que la respuesta del sistema va a la par con el cambio de la entrada.



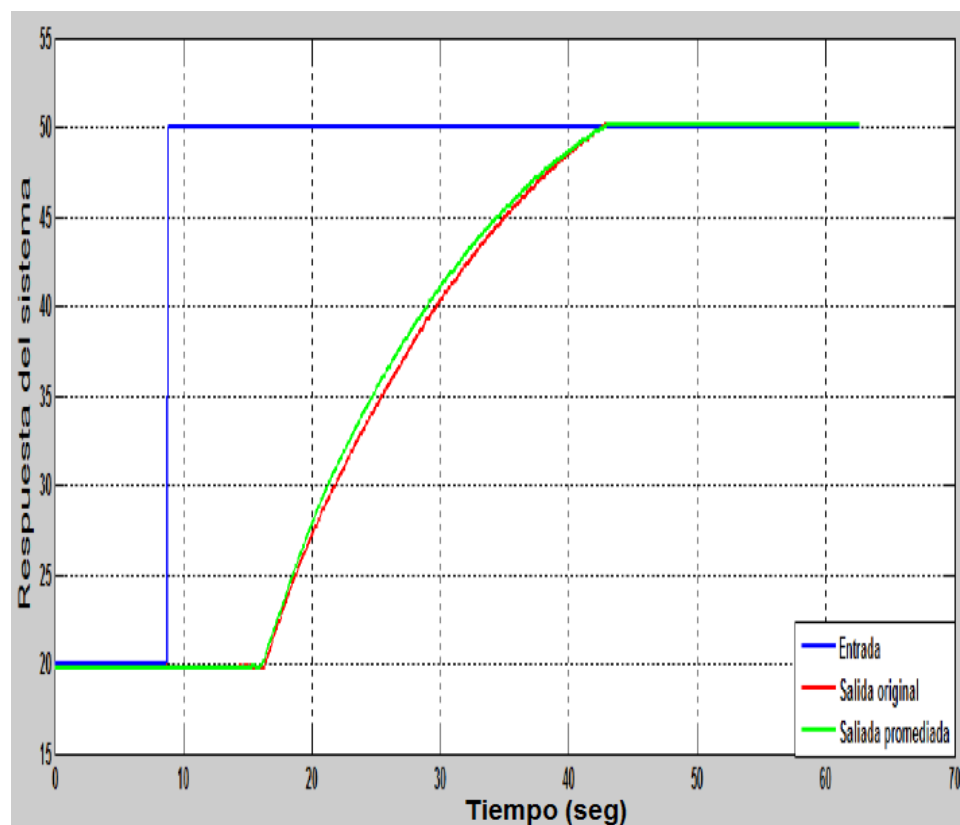
**Figura 2.7:** Dinámica del sistema.

Elaborado por: Curay, D. y Vaca, A.

Además se puede observar que no existe ruido incorporado en la señal por lo tanto no es necesario hacer un filtro. Entonces se puede considerar que el  $T_s$  va a ser igual a 0.1 seg debido a que el sistema responde de forma inmediata a este tiempo, posterior a eso se realiza un promedio de cinco valores y así se evita redundar datos innecesarios. Como se puede observar en la Figura 2.8, al realizar un promedio de datos la respuesta del sistema mejora.

Entonces el  $T_s$  será aproximadamente:

$$T_s = 0.1 \text{ seg} * 5 \text{ muestras} = 0.5 \text{ seg por muestra.} \quad (2.2)$$



**Figura 2.8:** Comparación entre la señal original y promediada.

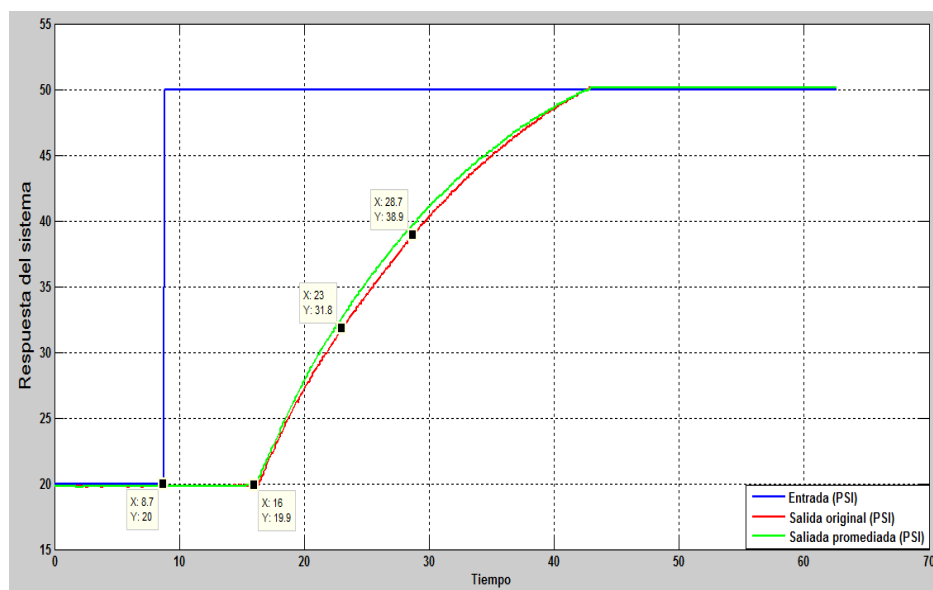
Elaborado por: Curay, D. y Vaca, A.

Otro criterio para escoger el tiempo de muestreo partirá de la respuesta al escalón, se puede observar en la Figura 2.9 la determinación de los parámetros.

Como regla general se acepta que  $T_s$  (tiempo de muestreo) debe ser 10 veces menor que el tiempo de subida ( $t_c$ ) en la salida del sistema continuo ante un escalón en lazo abierto. Este tiempo de subida se puede calcular de la siguiente manera:

$$t_c = 2 t_{63\%} - t_{39\%} - t_m \quad (2.3)$$

$$T_s = \frac{t_c}{10} \quad (2.4)$$



**Figura 2.9:** Experimento para calcular el  $T_s$ .

Elaborado por: Curay, D. y Vaca, A.

Aplicando las Ecuaciones 2.3 y 2.4 se obtiene que el periodo de muestreo ( $T_s$ ) es:

$$t_c = 2 * ( 28.7 - 23 ) - 16 - 8.7 = 11.4 \text{ seg} - 7.3 \text{ seg} = 4.1 \text{ seg.}$$

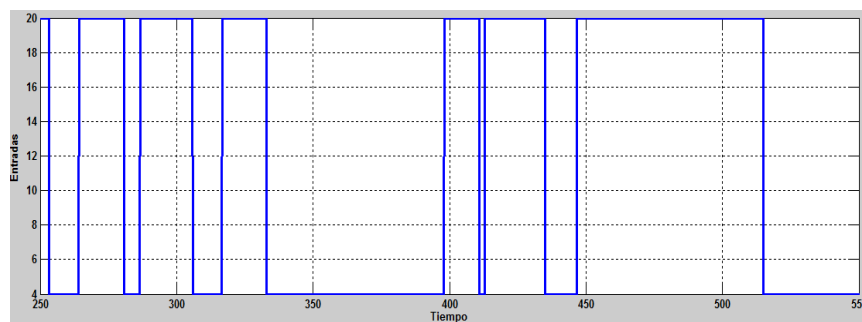
$$T_s = \frac{4.1 \text{ seg}}{10} = 0.41 \text{ seg}$$

Para la obtención de los datos se realiza una comparación entre los valores de tiempos de muestreo obtenidos, dando como resultado que el primer tiempo encontrado ( $T_s = 0,1 \text{ seg}$ ) permite obtener un mejor muestreo

de los datos ingresados ya que con este tiempo promediado con 5 muestras no se reduda en datos.

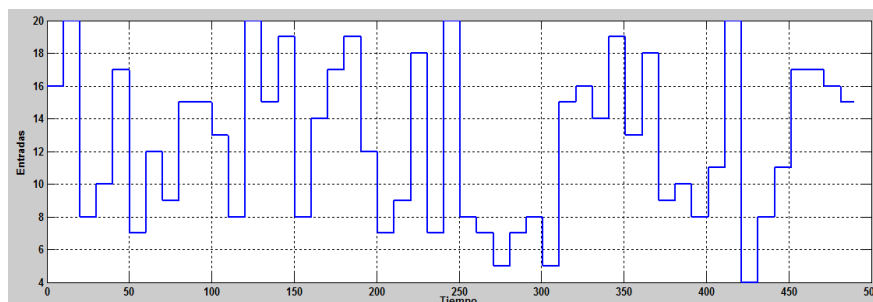
#### d. Tipo de entrada.

El tipo de entrada utilizada para excitar el sistema es el de tipo escalón (con cambios instantáneos), puesto que matemáticamente este tipo de señal contiene un espectro suficientemente amplio de frecuencias y que permite obtener a la salida la mayor cantidad de información para describirlo completamente, obteniendo así, un modelo de la planta de proceso similar al real. Existen dos tipos de señales de entrada escalón con las cuales se trabaja:



**Figura 2.10:** Entrada escalón Binaria.

Elaborado por: Curay, D. y Vaca, A.



**Figura 2.11:** Entrada escalón multinivel.

Elaborado por: Curay, D. y Vaca, A.

- ❖ La primera es una escalón binaria mostrada en la Figura 2.10, la cual trabaja con dos niveles de entradas, típicamente es el valor máximo y mínimo del rango de la señal de entrada (el cambio será de 4 a 20 mA

instantáneamente), este tipo de escalón es utilizada particularmente para identificar procesos de dinámica lineal.

- ❖ La segunda es una escalón multinivel mostrada en la Figura 2.11, este tipo de entrada permitirá realizar distintos cambios de nivel siempre y cuando esté dentro del rango mínimo y máximo del sistema, además es utilizado para sistemas no lineales.

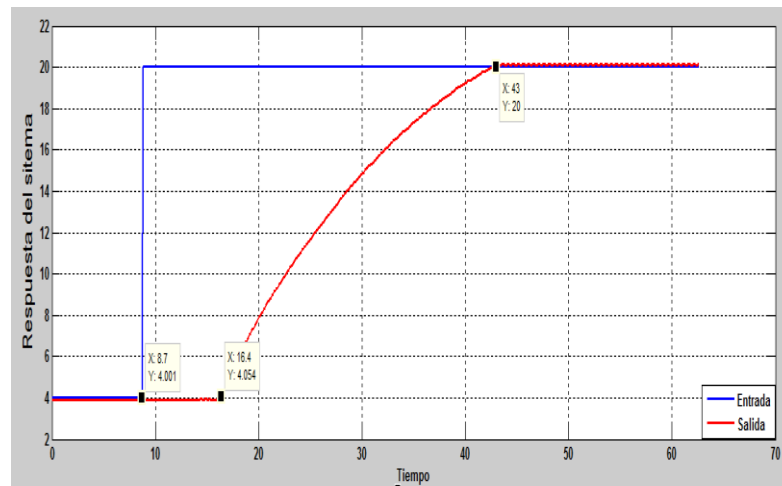
Como conocimiento previo la planta de presión utilizada tiende a ser no lineal, pero se realiza la adquisición de datos con los dos tipos de señales antes mencionadas y se opta por aquella que represente mejor al sistema dinámico.

El tiempo de duración en que permanecerá constante la entrada escalón antes del siguiente cambio es determinado por la dinámica propia del sistema tomando en cuenta que los cambios son aleatorios, calcular el tiempo de cambio es primordial debido a que un tiempo muy amplio provoca que el sistema se estabilice y dificulte estimar la respuesta estocástica del proceso, en cambio un tiempo corto no permite que el sistema reaccione adecuadamente ante una entrada específica. El tiempo mínimo del pulso para este sistema, será dado por el tiempo continuo que se calcula en la Ecuación 2.3, dando como resultado un tiempo de cambio de 4 seg.

Mientras que el tiempo amplio de pulso de cambio de las entradas aleatorias será el tiempo en que tarda el sistema en estabilizarse, este tiempo se determina mediante el ingreso de una entrada escalón desde el valor mínimo al máximo (4mA - 20 mA) y restado por el tiempo muerto dando como resultado un tiempo de cambio de 18 segundos, la respuesta del sistema ante la entrada escalón y el tiempo de cambio se puede observar en la Figura 2.12.

Para observar la estabilización del sistema la entrada y salida debe estar en la misma unidad, es decir en unidades de presión o corriente. En este caso se opta por trabajar en unidades de corriente.





**Figura 2.12:** Respuesta del sistema ante una entrada escalón.

Elaborado por: Curay, D. y Vaca, A.

- ❖ El tiempo de cambio mínimo para cada pulso es: 4 seg.
- ❖ El tiempo de cambio máximo para cada pulso es: 18 seg.

Mediante el experimento se elige el que permita obtener la mejor información para la modelación del sistema. Este apartado se realiza en la elección de la estructura de modelamiento.

#### e. Número de muestras.

La cantidad de muestras que se deben tomar durante el experimento vienen limitada por la capacidad del dispositivo de memoria utilizado, además elegir adecuadamente el número de muestras evita que la información sea repetida y que se requiera mayor procesamiento, entre otras cosas.

Se realiza cien cambios en la entrada de tipo escalón binario o multinivel y el tiempo de duración de cada pulso es aleatorio de 4 segundos que es el tiempo en que mejor reacciona la salida del sistema.

Estos cambios que multiplicados por la frecuencia de muestreo y el tiempo mínimo de cambio dan como resultado el número de muestras a tomar:

$$\#muestras = t_{cmin.} * 100 * \frac{1}{T_s} \quad (2.5)$$

$$\#muestras = 4 \text{ seg} * 100 * \frac{1}{0.1 \text{ seg}} \cong 4000 \text{ muestras}$$

### 2.2.2 Tratamiento previo de los datos.

Antes de realizar el modelamiento se debe tratar los datos registrados para eliminar datos erróneos, producidos por fallos en el hardware o software y así conseguir datos adecuados para el proceso de identificación.

#### a. Adquisición de datos.

En la adquisición de datos es importante tener los parámetros ya definidos, en la Tabla 2.3 se observa los valores.

**Tabla 2.3**

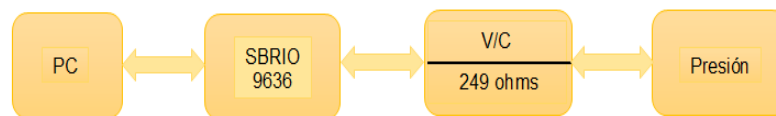
Parámetros utilizados para la identificación del proceso.

Rangos				
Entrada		Salida		
V. Inferior	V. Superior		V. Inferior	V. Superior
4 mA	20 mA	Corriente	4 mA	20 mA
		Salida acondicionada		
		Presión	20 PSI	50 PSI
Tipo de Entrada				
Binaria		Multinivel		
Bajo	Alto	La entrada tomara diferentes valores que se encuentren dentro del rango de 4 a 20 mA		
4 mA	20 mA			
–	–			
Tiempo de cambio aleatorio en la Entrada		Tiempo de muestreo (Ts)		
4 seg		0.1 seg		

Elaborado por: Curay, D. y Vaca, A.

El siguiente paso es diseñar el sistema de adquisición de datos, mediante hardware y software donde se utilizan los productos de National Instruments.

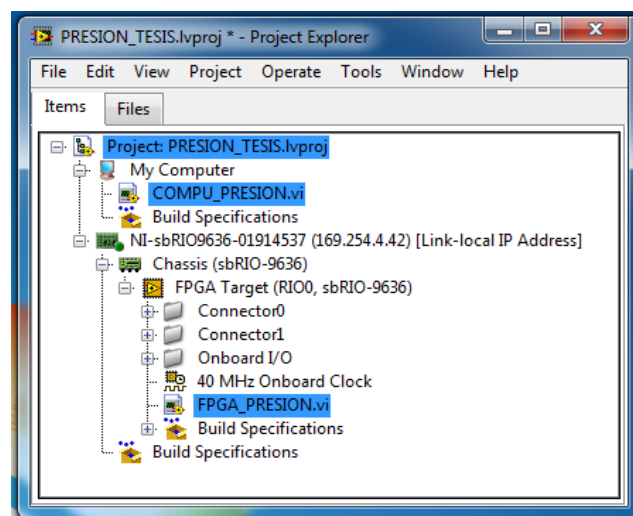
**Hardware:** Se utiliza un computador, el dispositivo SBRIO-9636, convertidor de voltaje-corriente y un juego de resistencias de precisión de 249 ohms, el convertidor y la resistencia de precisión son fundamentales debido a que la planta de presión trabaja con corriente y el SBRIO únicamente funciona con voltaje entonces estos elementos ayudan a convertir la corriente en voltaje y viceversa, en la siguiente Figura 2.13 se observa el diagrama de bloques para la adquisición de datos mediante hardware.



**Figura 2.13:** Diagrama de bloques de adquisición de datos.

Elaborado por: Curay, D. y Vaca, A.

**Software:** Se utiliza el programa Labview, para crear un proyecto como se observa en la Figura 2.14, éste permite la conexión entre la SBRIO y la computadora, además permite almacenar los datos de entrada/salida que se utilizan en la modelación del sistema.

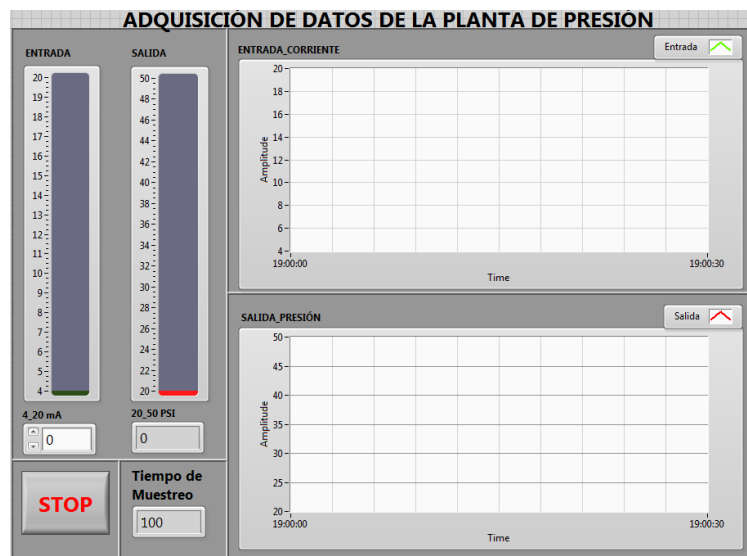


**Figura 2.14:** Estructura del proyecto de adquisición de datos.

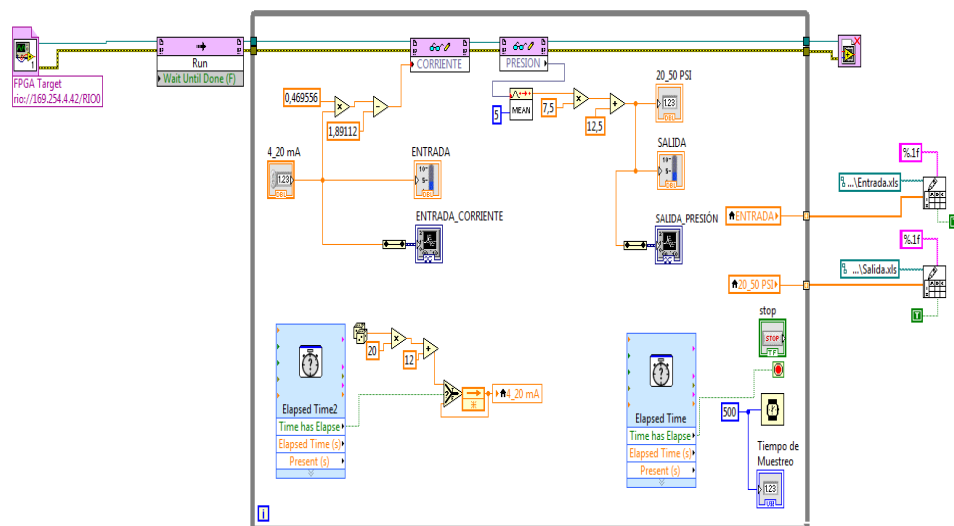
Elaborado por: Curay, D. y Vaca, A.

La comunicación entre la computadora y SBRIO es mediante Ethernet. La estructura del proyecto consta de dos VI's:

- ❖ **COMPU\_PRESION.vi:** es utilizado para guardar y almacenar los datos en el computador. En la Figura 2.15 se puede observar el panel frontal y el diagrama de bloques.



(a)

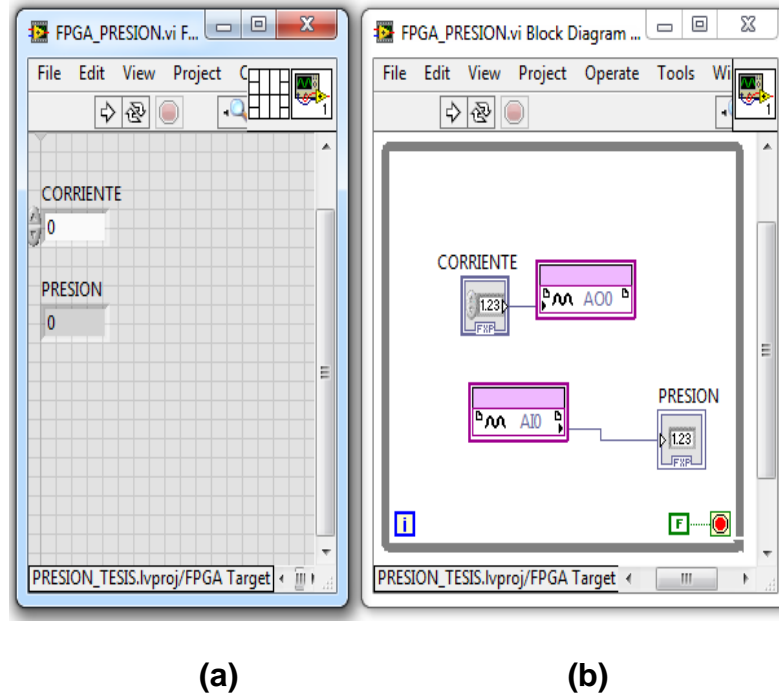


(b)

**Figura 2.15:** a) Panel frontal, b) Diagrama de bloques del VI's de la computadora.

Elaborado por: Curay, D. y Vaca, A.

- ❖ **FPGA\_PRESION.vi:** es utilizado para el manejo de las entradas y salidas de la SBRIO, es decir para la toma de datos. En la Figura 2.16 se observa el panel frontal y el diagrama de bloques del VI's FPGA.



**Figura 2.16:** a) Panel frontal, b) diagrama de bloques del VI's FPGA.

Elaborado por: Curay, D. y Vaca, A.

Una vez creado los VI's se procede a realizar el experimento de la toma de datos, como se observa en la Figura 2.15 los datos son almacenados en la computadora con sus respectivos nombres entrada/salida, recalcando que la entrada está en el rango de 4 a 20 mA, mientras que la salida está representada en unidades de presión de 20 a 50 PSI, se toman diferentes muestras para observar el comportamiento del proceso así como la respectiva validación del modelo matemático.

#### **b. Datos erróneos.**

Después de haber realizado el experimento se observa los datos erróneos, los cuales son causados por hardware y/o software del sistema de adquisición de datos y que deben ser eliminados ya que son considerados datos basura y no aportan información sobre el proceso.

Para la eliminación eficiente de estos datos erróneos, el usuario lo realiza por software, en la Figura 2.17 se observa la eliminación de éstos datos.

Entrada	Salida		
20	31,4	→	Residuos o basura en la memoria.
20	12,5	}	Datos erróneos debido a Hardware.
20	12,5		
20	12,5		
20	12,5		
20	12,5		
20	12,5		
20	12,5		
20	12,5		
20	12,5		
20	13,9		
20	15,4		
20	16,8		
20	18,3		
20	20	}	Datos válidos para la identificación del sistema.
20	20		
20	20,1		
20	20,3		
20	20,4		
20	20,6		
20	20,6		
20	20,8		
20	20,8		
20	21		
20	21,1		

**Figura 2.17:** Revisión de los datos almacenados.

Elaborado por: Curay, D. y Vaca, A.

Los datos basura deben ser eliminados y no deben ser tomados en cuenta al momento de la modelación del proceso, puesto que dichos valores nunca van a volver a aparecer en el transcurso de la toma de datos, como se puede visualizar en la Figura 2.17 donde son considerados datos no válidos porque los valores de la salida no están dentro del rango considerado.

Estos datos aparecen debido a que la estación de proceso estuvo fuera de funcionamiento y el programa ya empezó a guardar los datos.

El primer dato generalmente suele ser erróneo porque se almacena el valor antes de medirlo, siendo residuos de un experimento anterior o basura en la memoria, por lo que siempre se descarta la primera muestra de cada experimento. Una vez corregido estos errores se deben tomar n número de

muestras o las que se consideren necesarias para poder representar de manera óptima la dinámica de la planta de presión, el sistema debe ser sometido a diferentes cambios en su entrada y toda esta información debe ser guardada para posteriormente ser utilizada en la modelación y validación del proceso bajo estudio mediante el software MATLAB.

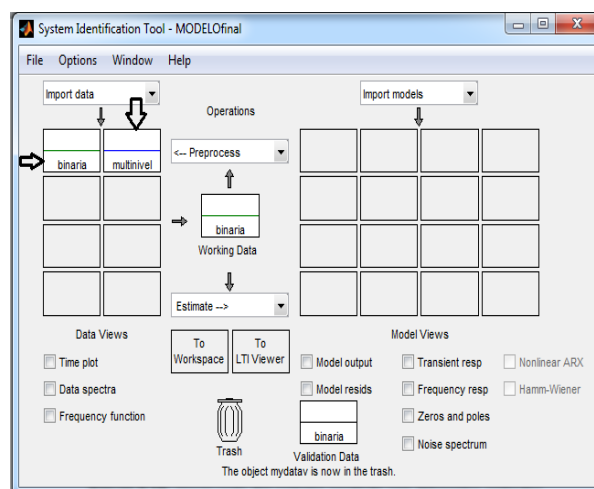
### 2.2.3 Modelación de la planta de presión mediante MATLAB.

En este apartado se utilizan los datos almacenados previamente con el fin de eliminar los inconvenientes o datos erróneos que puedan perjudicar el modelamiento del proceso. Para la modelación se utilizan las dos entradas (binaria y multinivel) y se opta por la que mejor represente la dinámica del sistema.

En el capítulo 1 se detallaron los pasos para la utilización de la herramienta IDENT por lo que no es necesario abordarlo nuevamente.

A continuación se describen los pasos a seguir para el modelamiento, con los datos tratados anteriormente:

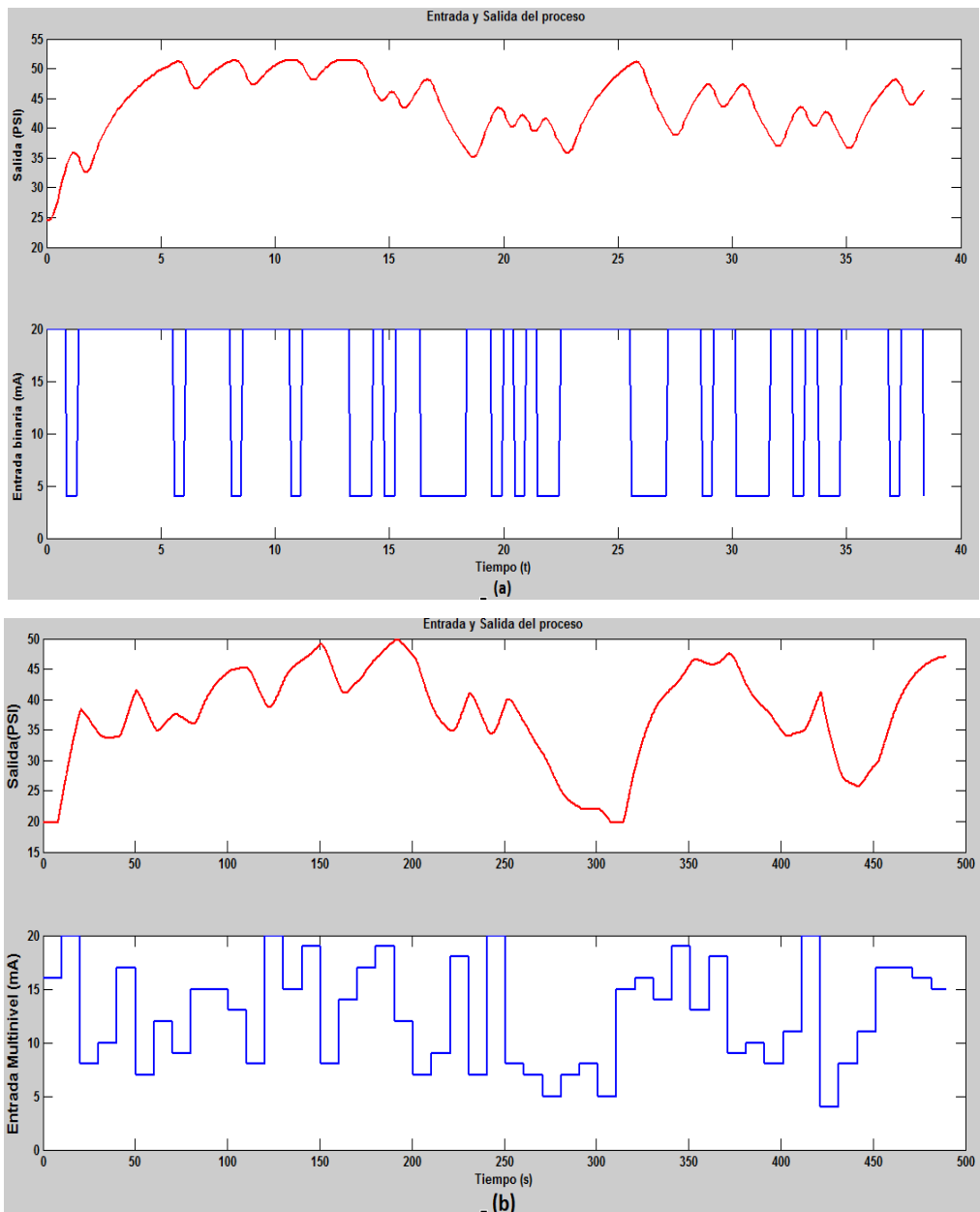
- ❖ Cargar los datos almacenados en la herramienta IDENT, esto se puede ver en la Figura 2.18.



**Figura 2.18:** Importación de datos.

Elaborado por: Curay, D. y Vaca, A.

- ❖ Se observa la dinámica del proceso en la salida cuando fue excitada con la entrada escalonada en binaria (a) y multinivel (b), mediante la opción Time plot la cual permite graficar la entrada vs. la salida, ver la Figura 2.19.



**Figura 2.19:** Respuesta del sistema excitada con diferentes entradas.

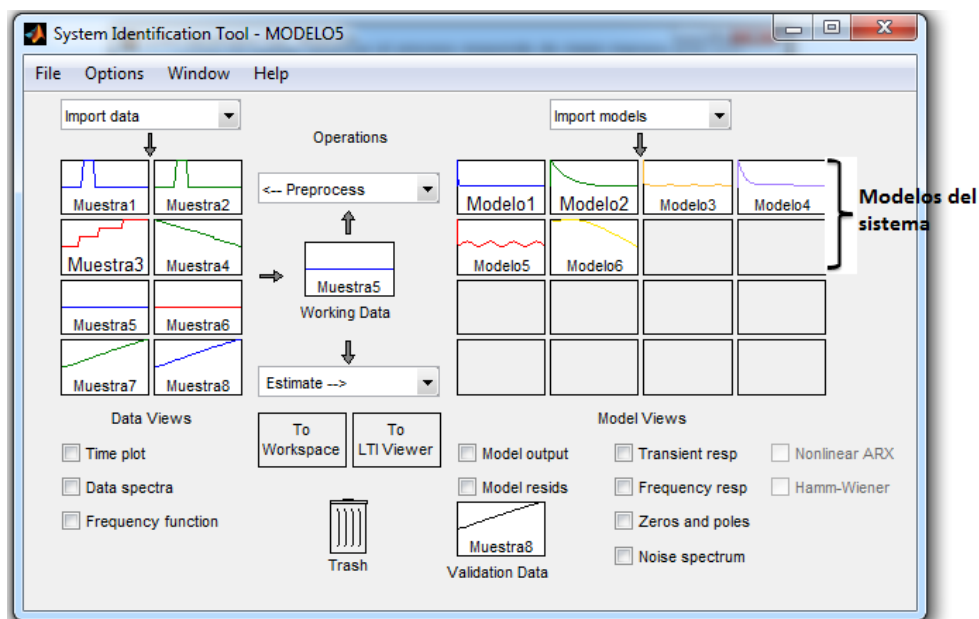
Elaborado por: Curay, D. y Vaca, A.

Como se puede observar el proceso responde de mejor manera ante una entrada escalón multinivel cubriendo todo el rango de 20 a 50 PSI en la



salida, por el contrario la entrada escalón binaria no es suficiente para que el proceso responda de manera óptima ante los cambios realizados en la entrada.

- ❖ Definido el tipo de entrada para excitar el proceso se empieza a cargar los datos para obtener diferentes modelos matemáticos la cual representan la dinámica del sistema, en este caso se representan en “Ecuaciones de Estado ” debido a que el MPC de Labview trabaja con esta forma de expresión.
- ❖ Los modelos matemáticos son obtenidos de diferentes muestras al excitar al sistema con diferentes cambios de entrada. En la Figura 2.20 se puede apreciar 6 modelos matemáticos obtenidos al aplicar la herramienta IDENT.



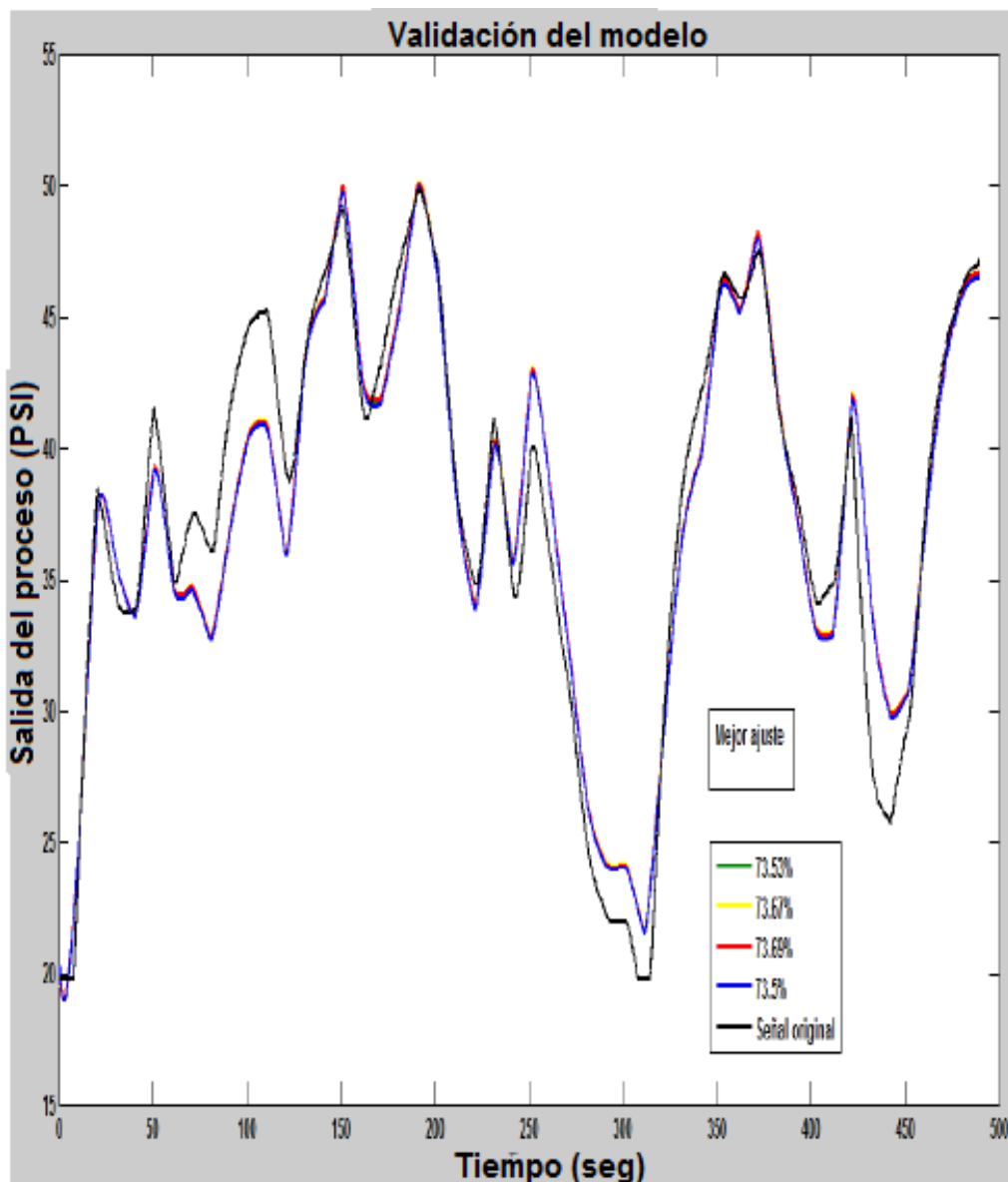
**Figura 2.20:** Modelos matemáticos del sistema.

Elaborado por: Curay, D. y Vaca, A.

#### 2.2.4 Validación del modelo.

La validación del modelo se realiza con la misma herramienta IDENT, se coloca las muestras tomadas en el cuadro de “Validation Data” para luego observar mediante la gráfica el porcentaje de similitud entre la señal original

y la proporcionada por el modelo matemático, se efectúa la validación para cada muestra y se verifica cual es el modelo matemático que tiene mejor respuesta o es similar a la señal original. En la Figura 2.21 se observa la validación del modelo matemático.



**Figura 2.21:** Validación del modelo matemático obtenido.

Elaborado por: Curay, D. y Vaca, A.

Se realiza la validación para cada muestra tomada y se elige el modelo que tenga la mejor respuesta en porcentaje de similitud con la señal original, en la Tabla 2.4 se muestra la comparación de cada modelo matemático.

**Tabla 2.4**

Comparación de los modelos matemáticos.

Datos	Modelo Matemático	Porcentaje de ajuste (%)
Muestra 1	Modelo1	66.76
	Modelo 2	66.51
	Modelo 3	37.29
	Modelo 4	73.80
	Modelo 5	74.02
	Modelo 6	74.5
Muestra 2	Modelo1	72.50
	Modelo 2	72.00
	Modelo 3	73.50
	Modelo 4	73.53
	Modelo 5	73.67
	Modelo 6	73.69
Muestra 3	Modelo1	33.09
	Modelo 2	51.34
	Modelo 3	36.37
	Modelo 4	51.48
	Modelo 5	51.58
	Modelo 6	72.10
Muestra 4	Modelo1	69.61
	Modelo 2	33.29
	Modelo 3	70.00
	Modelo 4	74.62
	Modelo 5	79.19
	Modelo 6	77.80
Muestra 5	Modelo1	52.55
	Modelo 2	75.61
	Modelo 3	79.00
	Modelo 4	78.06
	Modelo 5	75.82
	Modelo 6	83.54
Muestra 6	Modelo1	31.99
	Modelo 2	41.17
	Modelo 3	51.60
	Modelo 4	56.03
	Modelo 5	55.25
	Modelo 6	71.43
Muestra 7	Modelo1	43.27
	Modelo 2	59.84
	Modelo 3	43.02
	Modelo 4	73.76
	Modelo 5	42.47
	Modelo 6	80.11

Continua



Muestra 8	Modelo1	57.43
	Modelo 2	56.07
	Modelo 3	71.43
	Modelo 4	66.58
	Modelo 5	64.57
	Modelo 6	67.31

Elaborado por: Curay, D. y Vaca, A.

El modelo 6 se usa para la implementación del control predictivo puesto que el porcentaje de similitud con respecto a la de la señal original es alta y responde adecuadamente la dinámica del proceso de presión, cabe indicar que el modelo es discreto ya que la aplicación del MPC del módulo CONTROL DESIGN AND SIMULATION de Labview requiere una planta en modelo discreto, a continuación se detalla la presentación del modelo matemático del proceso entregado por MATLAB:

#### **Representación en espacio de estados:**

$$x(t + Ts) = A x(t) + B u(t) + K e(t)$$

$$y t = Cx t + Du t + e(t) \quad (2.6)$$

Dónde:

$x$  . → *vectorde estados.*

$y$  . → *vector de salida.*

$u$  . → *vector de entradas o control.*

$A$  . → *Matriz de estados.*

$B$  . → *Matriz de entrada.*

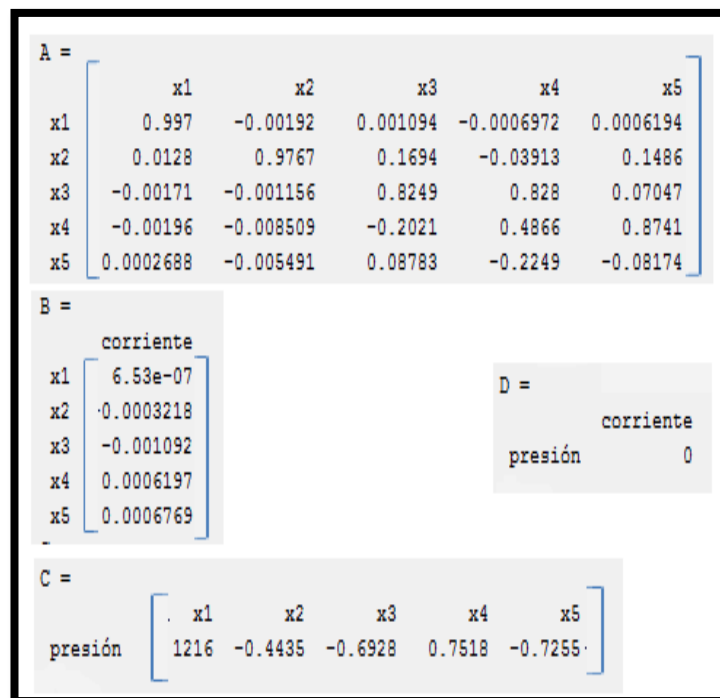
$C$  . → *Matriz de salida.*

$D$  . → *Matriz de transmisión directa.*

### Tiempo de muestreo:

$T_s = 0.5$  seg.

Las matrices A, B, C, D son aquellas que se introducen en el módulo del MPC que utiliza labview y son proporcionados por la herramienta “ident” de Matlab, ver Figura 2.22.



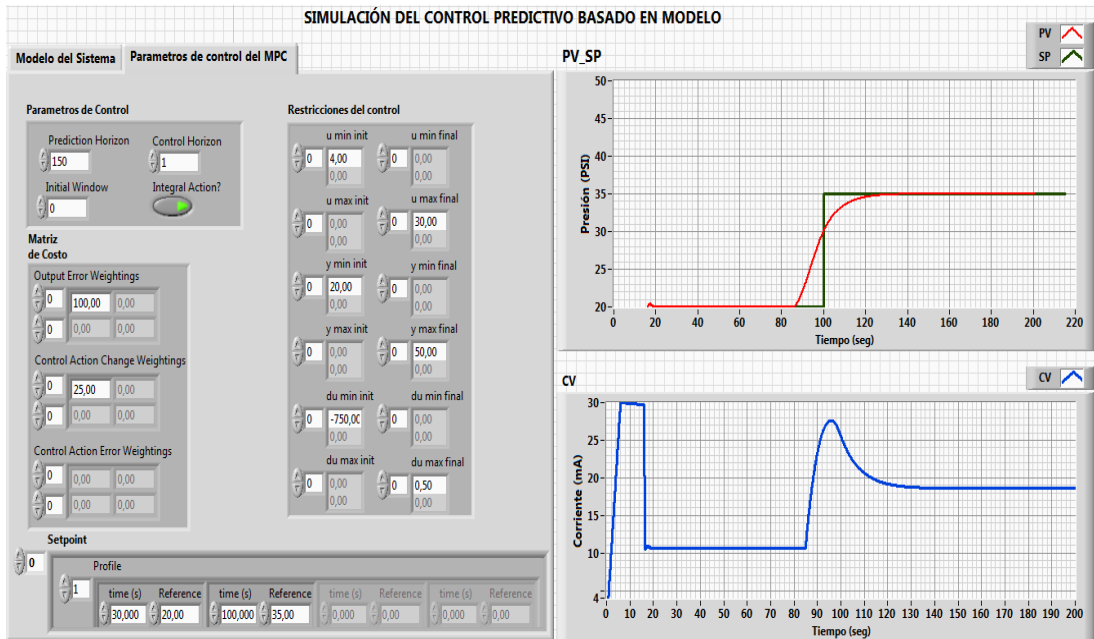
**Figura 2.22:** Matrices de la ecuación de estados.

Elaborado por: Curay, D. y Vaca, A.

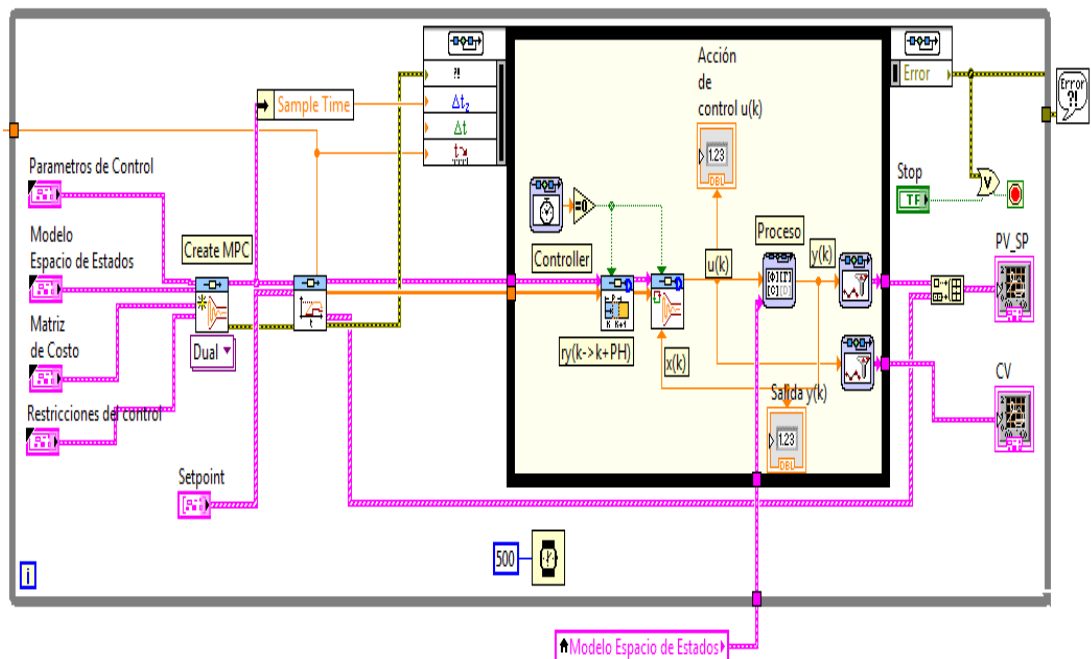
### 2.3 Diseño del controlador predictivo para la planta de presión.

La simulación del MPC se efectúa en el software Labview el cual presenta un toolkit denominada CONTROL DESIGN AND SIMULATION que posee un módulo llamado “control predictive” y facilita la implementación de dicho control, entonces se desarrolla un programa con interfaz gráfica para la comprensión y aprendizaje del control predictivo basado en modelo.

En la Figura 2.23 se presenta el panel frontal y el diagrama de bloques con sus respectivos sub VI's a utilizar para la simulación del MPC.



(a)



(b)

Figura 2.23: Simulación MPC. a) Diagrama frontal, b) Diagrama de bloques.

Elaborado por: Curay, D. y Vaca, A

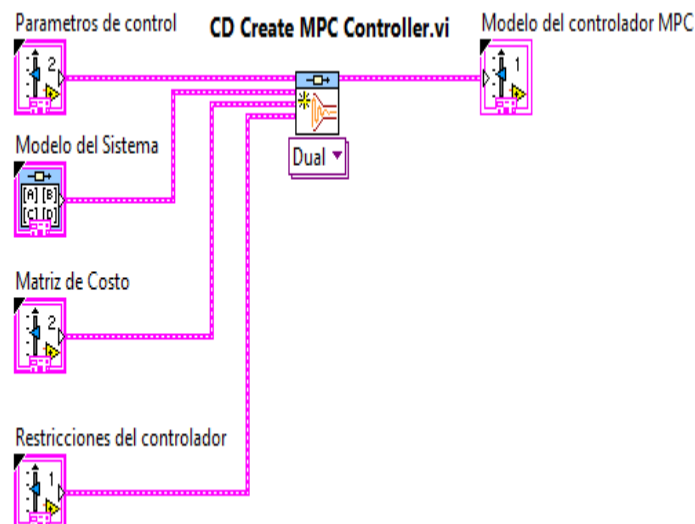
El módulo control predictivo se compone de varios sub VI's principales de los cuales se deben conocer la función que cumple cada uno de ellos, a continuación se detalla lo sub VI's:

- ❖ CD CREATE MPC CONTROLLER.
- ❖ CD UPDATE MPC WINDOW.
- ❖ CD IMPLEMENT MPC CONTROLLER.
- ❖ DISCRETE STATE-SPACE.
- ❖ CD SET MPC CONTROLLER.

### 2.3.1 CD create mpc controller

Permite crear el perfil del controlador MPC fundamentándose en el modelo del sistema bajo estudio en espacio de estados.

Anteriormente se dio a conocer las partes fundamentales del MPC que constan del modelo de predicción, función objetivo y la ley de control, cada una de estas partes juegan un papel primordial en el controlador, el subVI's "cd create mpc controller" además de crear el perfil del MPC permite manejar las configuraciones de dichas partes del controlador de manera fácil y sencilla, en la Figura 2.24 se indica los elementos principales del subVI's.



**Figura 2.24:** Diagrama de bloques de CD create MPC controller.

Elaborado por: Curay, D. y Vaca, A

El uso de estos elementos vienen definidos por la función objetivo (Ecuación 2.7) con la cual trabaja el control predictivo de labview a continuación se detalla la ecuación de la función objetivo.

$$\begin{aligned}
J(k) = & \sum_{i=N_w}^{N_p} \bar{y}(k+i|k) - r(k+i|k)^T \cdot \mathbf{Q} \cdot \bar{y}(k+i|k) - r(k+i|k) \\
& + \sum_{i=0}^{N_c-1} \Delta u^T(k+i|k) \cdot \mathbf{R} \cdot \Delta u(k+i|k) \\
& + \sum_{i=N_w}^{N_p} u(k+i|k) - s(k+i|k)^T \cdot \mathbf{N} \cdot [u(k+i|k) - s(k+i|k)]
\end{aligned}
\tag{2.7}$$

Dónde:

- ❖  $N_p$  → Número de muestras del horizonte de predicción.
- ❖  $N_w$  → Inicio del horizonte de predicción.
- ❖  $N_c$  → Número de muestras del horizonte de control.
- ❖  $\mathbf{Q}$  → Matriz de costo del error de salida.
- ❖  $\mathbf{R}$  → Matriz de costo de la tasa de cambio en la acción de control.
- ❖  $\mathbf{N}$  → Matriz de costo de la acción de control.
- ❖  $\bar{y}(k+i|k)$  → Salida del proceso predicha al tiempo  $k+i$ .
- ❖  $r(k+i|k)$  → Salida del proceso al tiempo  $k+i$ .
- ❖  $\Delta u(k+i|k)$  → Tasa de cambio de la acción de control predicha.
- ❖  $u(k+i|k)$  → Acción de control óptima predicha al tiempo  $k+i$ .
- ❖  $s(k+i|k)$  → Señal de referencia de entrada al tiempo  $k+i$ .

La función objetivo proporciona información relevante al momento de sintonizar el control predictivo.

#### a. Parámetros de control.

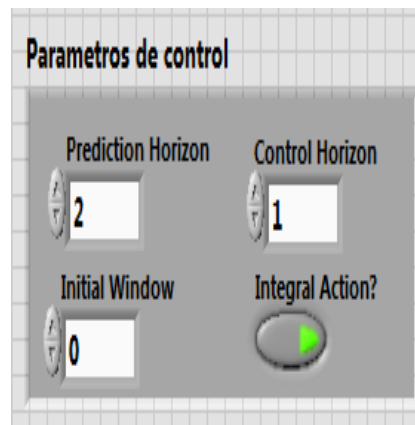
Este bloque contiene la siguiente información:

- ❖ Horizonte de predicción ( $N_p$ ): indica el número de muestras futuras en el cual el controlador MPC predice la salida de la planta.



- ❖ Horizonte de control ( $N_c$ ): indica el número de muestras dentro del  $H_p$ , este parámetro afecta directamente en el cálculo de la acción de control.
- ❖ Ventana inicial: define el rango inicial del controlador, se utiliza para procesos de control que muestran retrasos en sus modelos.
- ❖ Acción integral: Es utilizada para mejorar la diferencia existente entre la planta real y el modelo de la planta.

En la Figura 2.25 se muestra la parte frontal para el ingreso de los valores en este elemento.



**Figura 2.25:** Panel frontal del ingreso de los parámetros de control.

Elaborado por: Curay, D. y Vaca, A

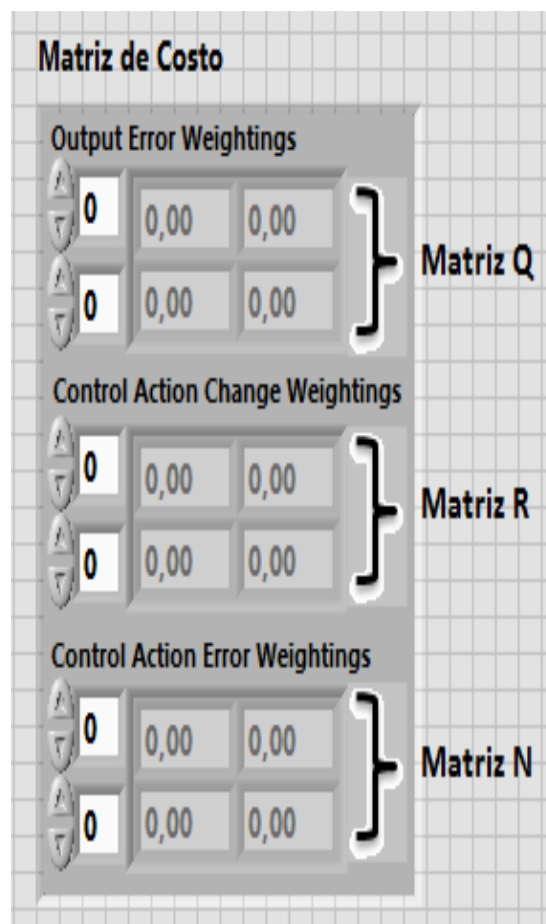
#### b. Matriz de costo.

Este elemento permite manejar el peso de las matrices costo  $Q$ ,  $R$ ,  $N$  que son parte de la función objetivo en la Ecuación 2.7:

- ❖ Matriz  $Q$ : Permite ingresar los valores del peso de ponderación de los errores en cada salida del sistema. La dimensión de la matriz es igual al número de salidas del proceso y los valores deben ser positivos.
- ❖ Matriz  $R$ : Permite ingresar los valores del peso de ponderación de la tasa de cambio en la acción de control. La dimensión de la matriz es igual al número de entradas del proceso y los valores deben ser positivos.

- ❖ **Matriz N:** Permite ingresar los valores del peso de ponderación de la acción de control. La dimensión de la matriz es igual al número de entradas del proceso, se recomienda usar esta matriz cuando el número de entradas del proceso es mayor que el número de sus salidas es decir en proceso multivariables.

En la Figura 2.26 se muestra el panel frontal para el ingreso de los valores en dicho elemento.

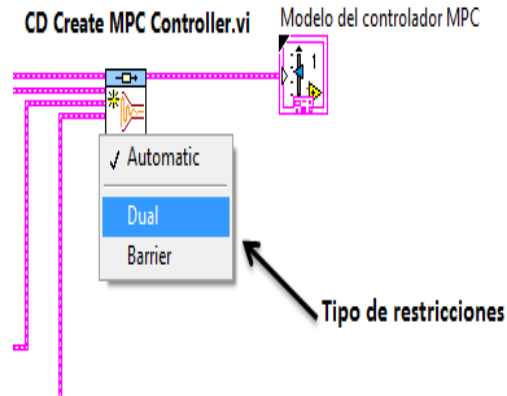


**Figura 2.26:** Panel frontal del ingreso de valores a la matriz costo.

Elaborado por: Curay, D. y Vaca, A

### c. Restricciones del controlador.

El subVI's "cd create MPC controller", permite elegir dos método de restricción (DUAL y BARRERA) para optimizar la función objetivo, ver Figura 2.27.



**Figura 2.27:** Elección del tipo de restricción.

Elaborado por: Curay, D. y Vaca, A

Como se conoce los pesos de ponderación de las matrices de costo permiten ajustar las prioridades de la salida, acción de control y la tasa de cambio en la acción de control de los procesos, sin embargo estos parámetros pueden exceder el rango de control del proceso por tal razón es importante usar un método de restricción para que estas medidas nunca excedan los rangos permitidos, la función principal de estos métodos de restricción es ajustar u otorgar al controlador los valores límites de trabajo.

Como se observa en la Figura 2.27 existen dos métodos de restricción, la restricción “dual” evita que el controlador sobrepase los límites del proceso, si ha sobrepasado lo que hace es tomar las acciones correctivas ignorando el resto de parámetros hasta que el proceso ingrese nuevamente al rango de trabajo válido. En cambio el método “barrera” utiliza penalidades y tolerancias y tiene la función que mientras se está acercando a los límites de trabajo disminuye la acción de control en forma proporcional evitando infringir los límites de trabajo, para este caso se trabaja con el método DUAL debido a su facilidad de manejo.

En la Figura 2.28 se puede observar el panel frontal de este elemento, donde la salida del proceso está representada con la letra (**y**), la acción de control o la entrada al proceso está representada con la letra (**u**) y finalmente la tasa de cambio de la acción de control está representada con la letra (**du**).

Restricciones del controlador			
$\frac{A}{T}$	u min init	$\frac{A}{T}$	u min final
0	0,00	0	0,00
$\frac{A}{T}$	u max init	$\frac{A}{T}$	u max final
0	0,00	0	0,00
$\frac{A}{T}$	y min init	$\frac{A}{T}$	y min final
0	0,00	0	0,00
$\frac{A}{T}$	y max init	$\frac{A}{T}$	y max final
0	0,00	0	0,00
$\frac{A}{T}$	du min init	$\frac{A}{T}$	du min fina
0	0,00	0	0,00
$\frac{A}{T}$	du max init	$\frac{A}{T}$	du max fina
0	0,00	0	0,00

Acción de control  
Entrada al proceso

Salida del proceso

Tasa de cambio de la Acción de control

**Figura 2.28:** Panel frontal del método DUAL.

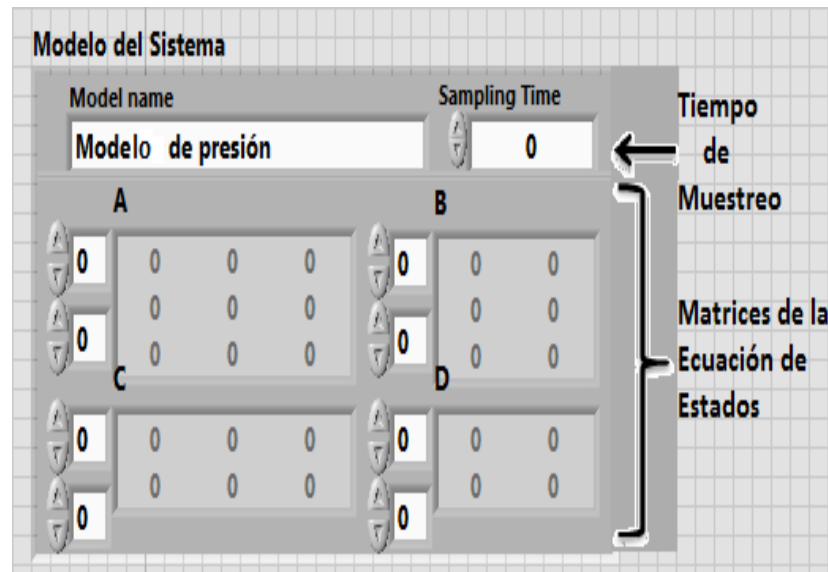
Elaborado por: Curay, D. y Vaca, A

La restricción de acción de control (**u**) mínima y máxima toma los valores de 4 a 20 mA con los que se excita el sistema. La restricción de tasa de cambio de acción de control (**du**) mínima y máxima se determina de acuerdo al controlador en cuanto la velocidad es decir se queremos un control rápido o lento esto depende de la necesidad del proceso.

La restricción de salida del sistema (**y**) mínima y máxima toma los valores de 20 a 50 PSI otorgador por el transmisor de presión, tanto la acción de control y la salida son los valores del modelo previamente hallado en Matlab. De acuerdo a las restricciones puestas el controlador no superará estos límites entonces siempre está dentro de los rangos útiles de trabajo del sistema.

#### **d. Modelo del sistema.**

Permite ingresar el modelo del proceso que va a ser controlado por el MPC en espacio de estados. Cabe mencionar que esta es una parte fundamental del control predictivo debido a que el modelo permitirá predecir las salidas futuras del controlador, es decir que el rendimiento del controlador dependerá de la precisión del modelo. En la Figura 2.22 se detalla la estructura del modelo de presión.



**Figura 2.29:** Panel frontal del elemento de modelo del sistema.

Elaborado por: Curay, D. y Vaca, A

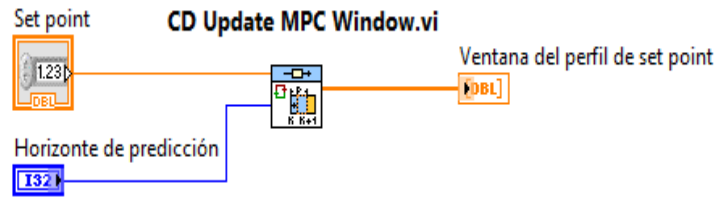
En la Figura 2.29 se aprecia la parte frontal de este elemento el cual permite ingresar los parámetros de la ecuación del modelamiento del proceso de presión.

El modelo del sistema debe ser discreto por ende es necesario insertar el parámetro del tiempo de muestreo. Finalmente al dotar de todos los parámetros necesarios a la salida del subVI's "Cd create MPC controller" se obtiene el modelo del controlador MPC.

### 2.3.2 CD update mpc windows.

Permite ingresar los valores de referencia al controlador MPC, además calcula la porción apropiada de la señal de referencia desde el instante  $k$  al instante  $k + \text{horizonte de predicción}$ , es decir proporciona la ventana de setpoint adecuada para todo el rango de predicción. La salida entregada por este subVI's es conectada a la ventana de referencia de salida del subVI's "CD implement mpc controller".

En la Figura 2.30 se observa los elementos del subVI's "CD update mpc window".

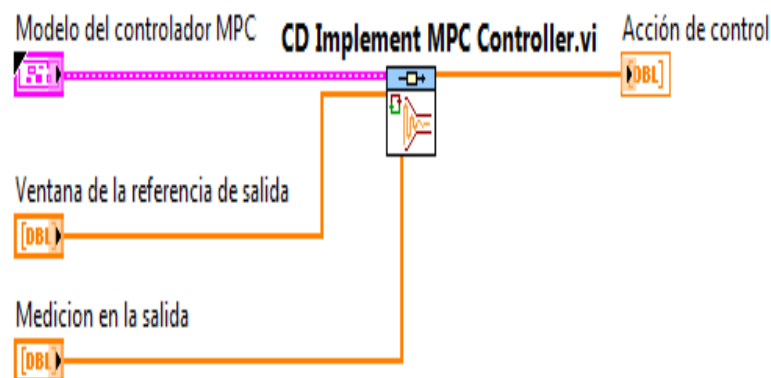


**Figura 2.30:** Diagrama de bloque del CD Update MPC Window.

Elaborado por: Curay, D. y Vaca, A

### 2.3.3 CD implement mpc controller.

Permite implementar la señal de control optimizada o llamada también ley de control la cual se obtiene al ver minimizado la función objetivo que lo hace internamente, pero antes se debe proporcionar a este subVI's las salidas predichas en función de los valores actuales y pasados, es decir los valores de referencia y la entrada realimentada del proceso, en la Figura 2.31 se indican los elementos principales del subVI's para implementar el controlador.



**Figura 2.31:** Diagrama de bloque de CD Implement MPC controller.

Elaborado por: Curay, D. y Vaca, A

Como se observa en la figura anterior el subVI's necesita a su entrada el modelo del controlador MPC, la referencia y la medición en la salida, estos elementos permiten obtener la acción de control o entrada al proceso, dicha

acción de control es aquella que tiene la función de controlar el actuador del proceso de presión para el respectivo cierre y apertura de la válvula.

**a. Modelo del controlador MPC.**

Este elemento es proporcionado por el “CD create mpc controller”, en la Figura 2.23 se puede observar la conexión respectiva de este subVI’s a la entrada del subVI’s “CD implement mpc controller”.

**b. Ventana de la referencia de salida.**

Este elemento permite ingresar la ventana de setpoint con respecto al horizonte de predicción. Este valor es proporcionado por subVI’s “CD update mpc windows”.

**c. Medición en la salida.**

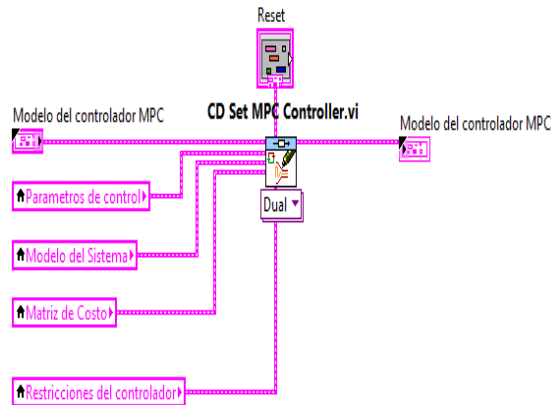
Este elemento permite ingresar el valor de salida del proceso de presión, es decir, es la realimentación del proceso.

**2.3.4 CD set mpc controller.**

Permite cambiar los parámetros del controlador MPC, es decir actualiza en tiempo real todos los elementos del perfil MPC creado por el subVI’s “CD create mpc controller”, esta opción es de mucha utilidad debido a que en la ejecución del programa el usuario opta por cambiar el modelo de predicción, las restricciones del proceso, las ponderaciones de la matriz costo o en sí cambiar el horizonte de predicción.

Para realizar estos cambios se ve forzado a detener el control, en cambio al ser simulación no se tiene ningún problema como el que se presenta cuando es un sistema real, ya que este último resulta perjudicial para la empresa.

Entonces la función del “CD set mpc controller” es permitir sintonizar el controlador MPC en tiempo real sin tener la necesidad de detener el control.

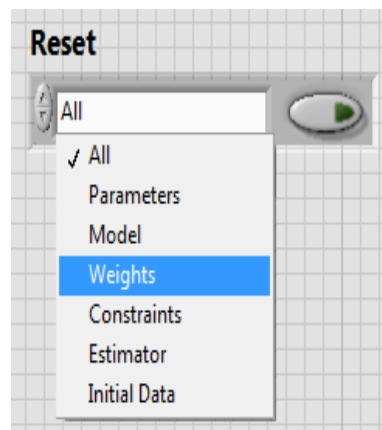


**Figura 2.32:** Diagrama de bloque del CD set MPC controller.

Elaborado por: Curay, D. y Vaca, A

Como se puede observar en la Figura 2.32, el subVI's posee las mismas características que el subVI's "CD create mpc controller", esto quiere decir que al presionar el botón RESET el "CD create mpc controller", crea un nuevo perfil de MPC con los nuevos valores proporcionados por el subVI's "CD set mpc controller".

El botón RESET permite elegir el parámetro que se quiere actualizar, en la Figura 2.33 se puede observar las opciones.



**Figura 2.33:** Opciones para actualizar del controlador MPC.

Elaborado por: Curay, D. y Vaca, A

Existen diferentes opciones cada una de ellas permite realizar la siguiente acción:



- ❖ Parámetros (Parameters): actualiza el horizonte de predicción, horizonte de control, los pesos de ponderación de la matriz costo y las restricciones del método Dual.
- ❖ Modelo (Model): actualiza el modelo del sistema que se utiliza para la predicción. Es de importancia que el nuevo modelo ingresado posea el mismo número de entradas como de salidas del anterior modelo.
- ❖ Pesos de ponderación: actualiza los pesos de ponderación de la matriz costo Q, R, N del controlador MPC.
- ❖ Restricciones (Constraints): actualiza las restricciones de la optimización del controlador MPC.
- ❖ Datos iniciales (Initial Data): actualiza las condiciones iniciales en que se encuentra el proceso al momento de ejecutar el controlador.
- ❖ Todo (All): Actualiza todos los parámetros del controlador predictivo.

### 2.3.5 Discrete State-Space

Permite simular el comportamiento de la planta real, para este caso representa la dinámica de la planta de presión. La función de este subVI's es implementar un modelo discreto del sistema en forma de espacio de estado. En la Figura 2.34 se indica los elementos del subVI's.

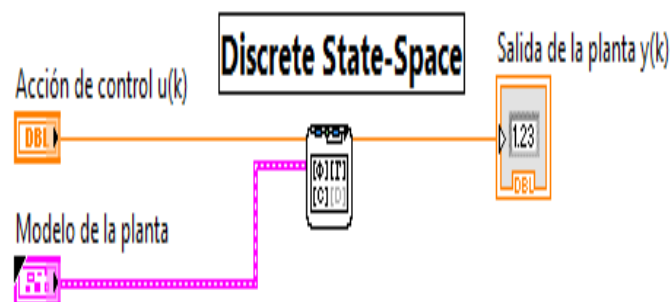


Figura 2.34: Diagrama de bloque del Discrete State-Space.

Elaborado por: Curay, D. y Vaca, A

Como se muestra en la figura anterior el “Discrete State-Space” al ingresar los datos de la acción de control (entrada del proceso) y el modelo

de la planta, entrega el comportamiento de la salida del sistema ante los cambios que se presente en su entrada.

### 2.3.6 Datos obtenidos de la simulación del MPC

Mediante la interfaz creada en la Figura 2.23 para la simulación del MPC, se puede comenzar a sintonizar los parámetros que permitan optimizar la función objetivo y así obtener un control óptimo.

La simulación permite obtener las siguientes ventajas:

- ❖ Permite ganar experiencia en el manejo del controlador MPC.
- ❖ Ayuda a conocer cómo reacciona el sistema al introducir nuevos parámetros en el control MPC.
- ❖ Permite observar el peso que tiene cada uno de los parámetros y como afectan estos en el funcionamiento óptimo del control MPC.
- ❖ Permite elegir los valores apropiados con el que va a trabajar el controlador MPC, para posteriormente ser implementado en forma real.

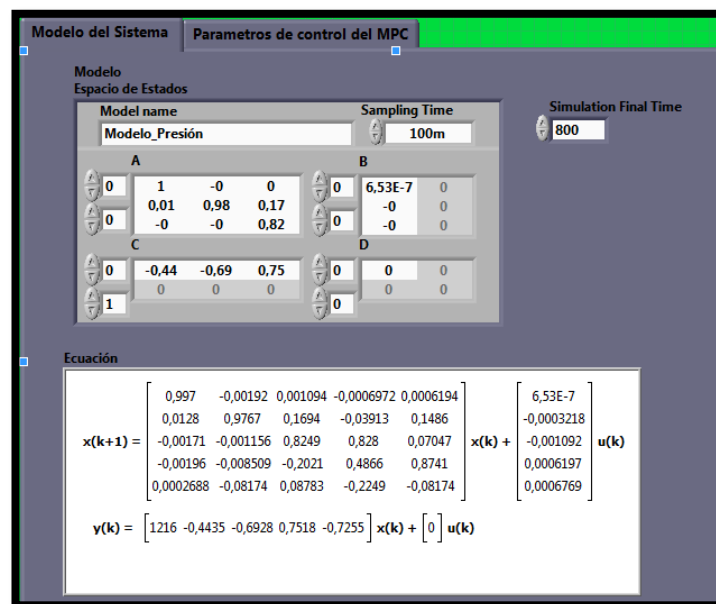


Figura 2.35: Modelación del sistema en espacio de estados.

Elaborado por: Curay, D. y Vaca, A

Entonces el primer paso para implementar el controlador MPC en simulación es ingresar la matriz en espacio de estados al controlador MPC, dicha matriz es la representación del modelamiento en tiempo discreto del sistema bajo estudio, ver Figura 2.35. Las matrices ingresadas representan el modelo entregado por MATLAB, dicho modelo presenta un alto porcentaje de similitud con la dinámica real del proceso de presión por esta razón se escoge el modelo 6, en la figura antes vista se puede observar la matrices ingresadas del modelo así como un indicador gráfico del mismo modelo.

En la siguiente pantalla (Figura 2.36), se encuentran todos los parámetros que se modifican para realizar el control Predictivo Basado en Modelo, mediante la manipulación de cada uno de estos valores se puede observar gráficamente la respuesta del modelo al aplicar el control MPC.

The screenshot shows the 'Parametros de control del MPC' interface. It is divided into two main sections: 'Parametros de Control' and 'Restricciones del control'. Below these are 'Matriz de Costo' and 'Setpoint'.

**Parametros de Control:**

- Prediction Horizon: 150
- Control Horizon: 1
- Initial Window: 0
- Integral Action?:

**Matriz de Costo:**

- Output Error Weightings:
 

0	100,00	0,00
0	0,00	0,00
- Control Action Change Weightings:
 

0	25,00	0,00
0	0,00	0,00
- Control Action Error Weightings:
 

0	0,00	0,00
0	0,00	0,00

**Restricciones del control:**

- u min init: 4,00; u min final: 0,00
- u max init: 0,00; u max final: 20,00
- y min init: 20,00; y min final: 0,00
- y max init: 0,00; y max final: 50,00
- du min init: -0,05; du min final: 0,00
- du max init: 0,00; du max final: 0,05

**Setpoint:**

0

Profile		Profile		Profile		Profile	
time (s)	Reference	time (s)	Reference	time (s)	Reference	time (s)	Reference
0,000	20,00	200,000	35,00	0,000	0,00	0,000	0,00

**Figura 2.36:** Parámetros del control MPC.

Elaborado por: Curay, D. y Vaca, A.

La simulación permite modificar todos los parámetros para comprobar el comportamiento de la planta y ver el efecto que tienen éstos en la operación óptima del control. Entonces este experimento genera la siguiente información:

- ❖ El horizonte de predicción ( $H_p$ ) debe ser mayor que el horizonte de control ( $H_c$ ). Si el  $H_c$  está fuera del rango del  $H_p$ , el controlador no responde con ninguna acción de control debido a que la predicción finalizó antes de enviar una respuesta de control. En la Figura 2.37 se puede observar que el controlador MPC no generó ninguna acción de control al proceso.



**Figura 2.37:** Respuesta del MPC cuando el  $H_c > H_p$ .

Elaborado por: Curay, D. y Vaca, A.

- ❖ Si el horizonte de predicción es corto se reduce el tiempo en la que el controlador MPC puede predecir la señal de salida del proceso, incluso provocaría que el controlador MPC opere como un control tradicional PID, ver Figura 2.38 (a).

- ❖ En cambio un horizonte de predicción grande o adecuada incrementará la capacidad de predecir al controlador MPC, ver Figura 2.38 (b).

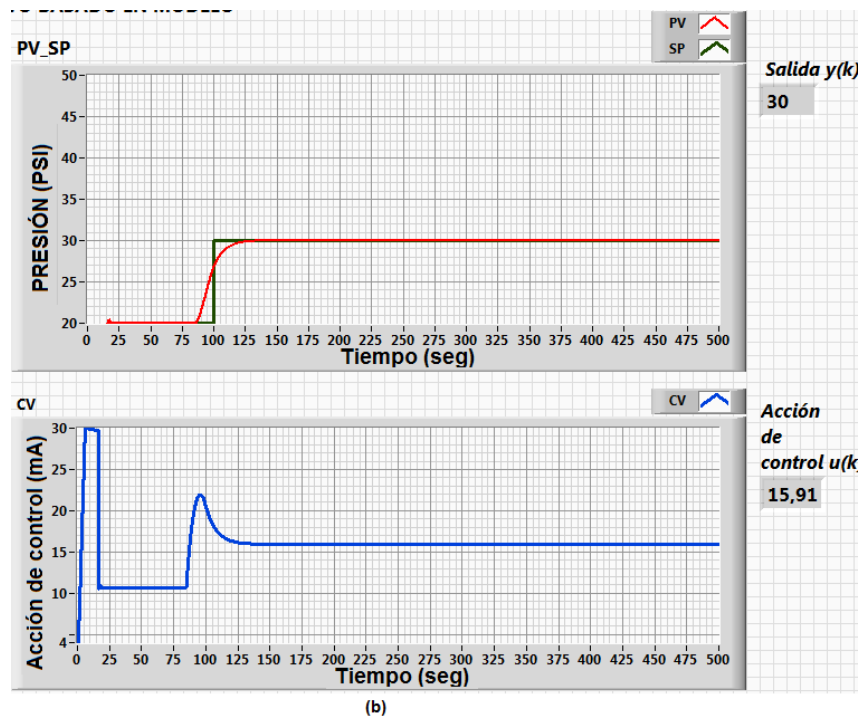
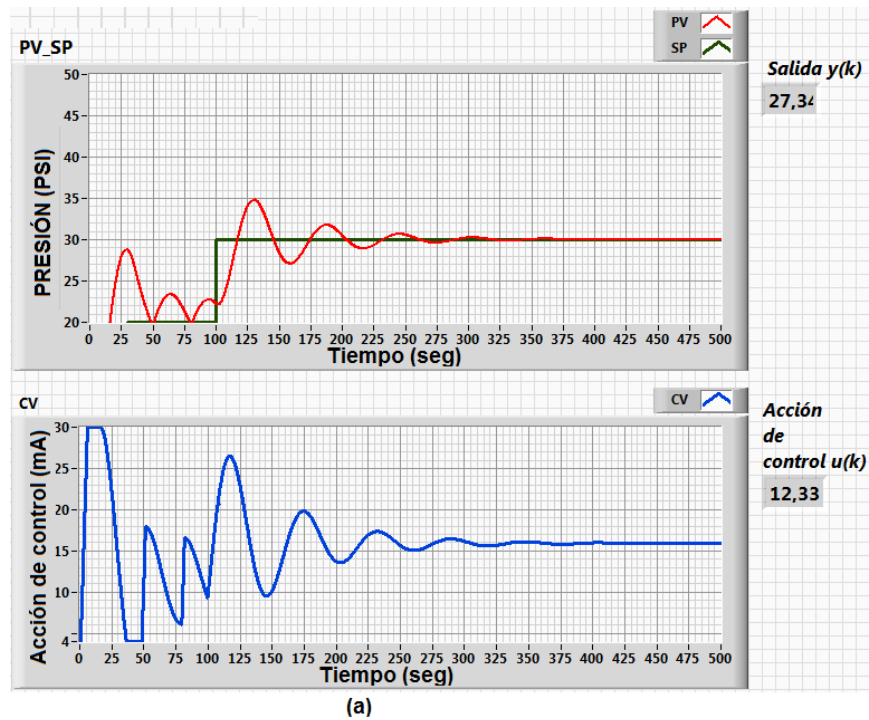
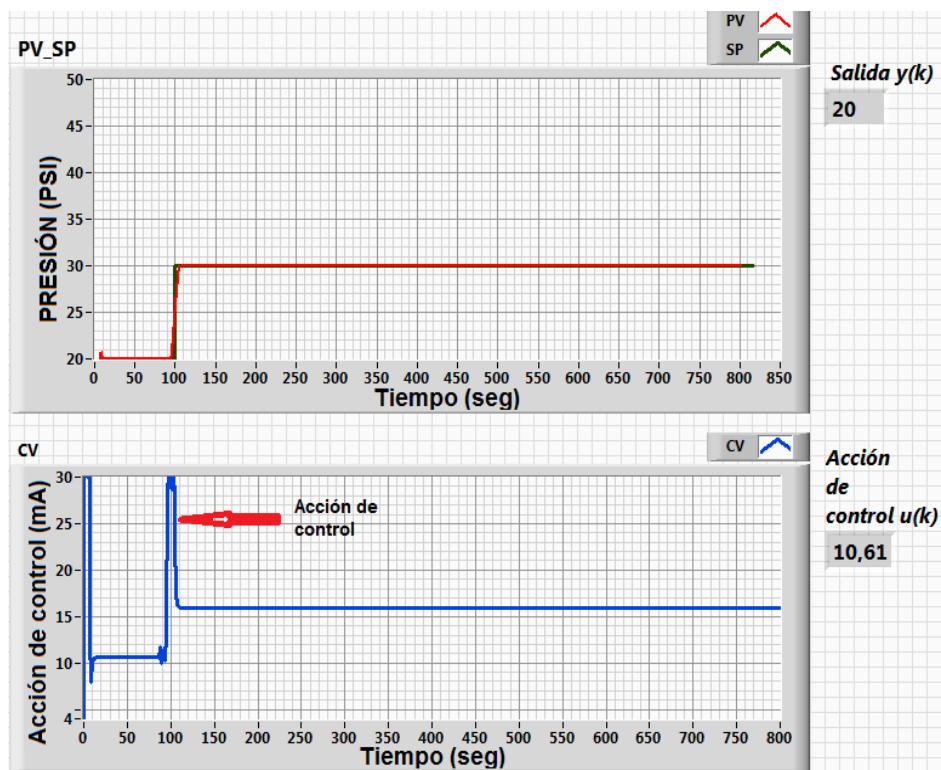


Figura 2.38: Respuesta del controlador, a) Hp corto, b) grande.

Elaborado por: Curay, D. y Vaca, A.

Se debe tener en cuenta que el controlador MPC realiza cálculos para la optimización de la acción de control empleando la función objetivo entonces si el horizonte de predicción es demasiado grande decremente el rendimiento del MPC por el aumento de cálculos adicionales que realiza el algoritmo de control.

- ❖ Si el horizonte de control es grande produce cambios agresivos en la acción de control, ver Figura 2.39. Estos cambios agresivos pueden provocar el deterioro de la válvula de control entonces es necesario usar valores pequeños en el horizonte de control.



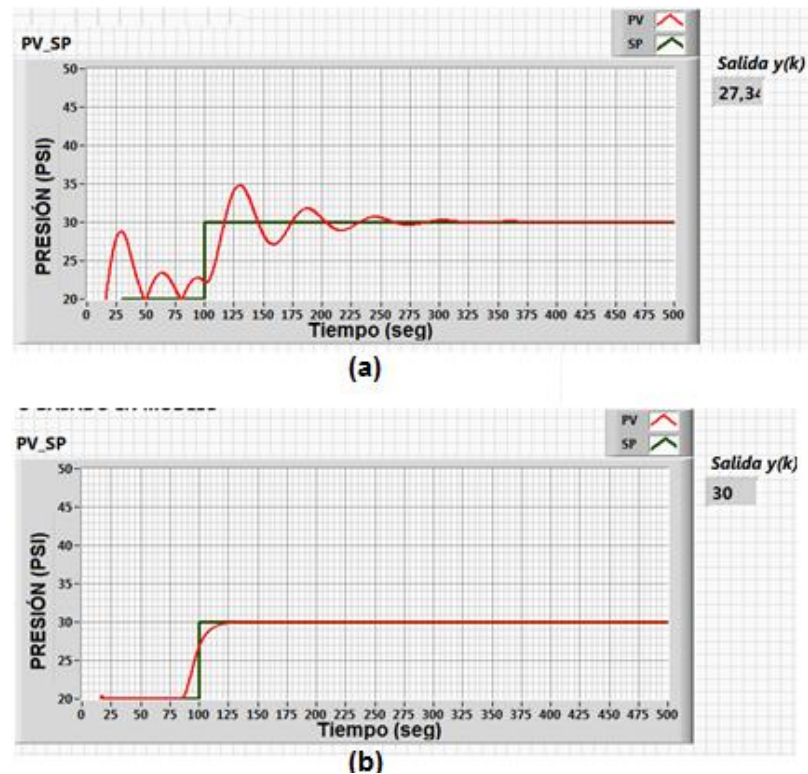
**Figura 2.39:** Respuesta del controlador cuando el  $H_c$  es muy grande.

Elaborado por: Curay, D. y Vaca, A.

- ❖ Las matrices de costo  $Q$ ,  $R$ ,  $N$  permiten mejorar el controlador MPC cuando la elección de un adecuado valor de horizonte de predicción y horizonte de control resultan insuficientes para poder controlar el proceso. Como las matrices de costo son parte de la ecuación de la función objetivo, estas afectan directamente al cálculo del algoritmo

de optimización, es decir, los valores de estas matrices mejoran o en si perjudican el rendimiento del controlador MPC, ver la respuesta del controlador en la Figura 2.40 (a) sin aplicar las matrices costo y la Figura 2.40 (b) aplicadas las matrices costo.

- ❖ El valor de la matriz Q representa el error de la salida, es decir que esta matriz determina la velocidad de reacción del controlador ante el error actual del proceso sin importar el tiempo.
- ❖ El valor de la matriz R representa la tasa de cambio en la acción de control, es decir que esta matriz determina el movimiento del actuador (provocado por la acción de control) a una velocidad o rapidez proporcional a la desviación respecto al punto de consigna (variable deseada).

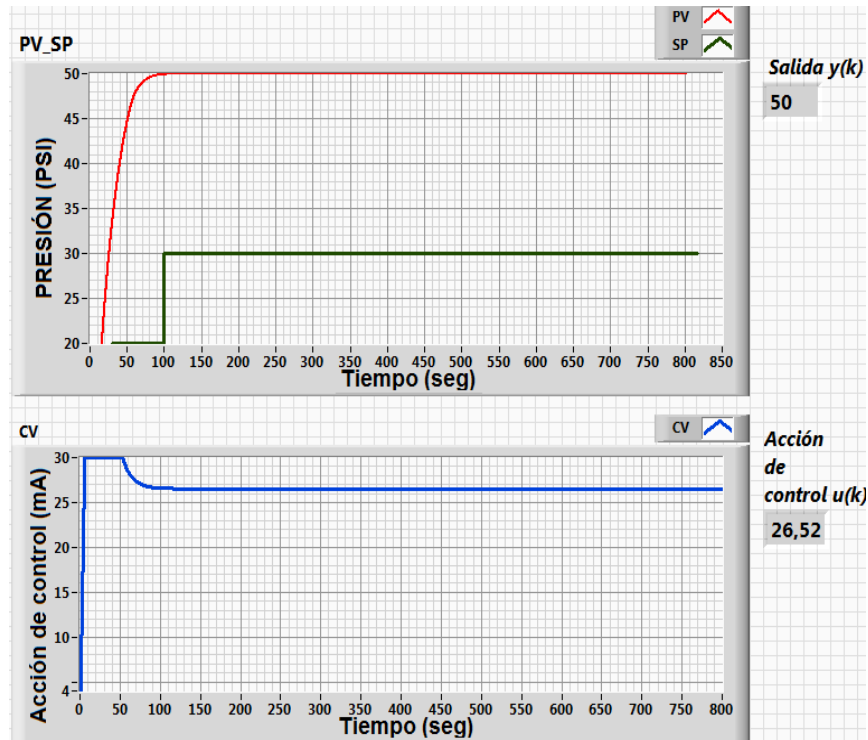


**Figura 2.40:** a) Sin matrices costo, b) Aplicando las matrices costo Q, R.

Elaborado por: Curay, D. y Vaca, A.

- ❖ La matriz N se utiliza cuando el número de entradas es mayor que el número de salidas del proceso, por esta razón no se usa este

parámetro, debido a que perjudica el funcionamiento del controlador (ver Figura 2.41).



**Figura 2.41:** Respuesta del controlador al aplicar la matriz costo.

Elaborado por: Curay, D. y Vaca, A.

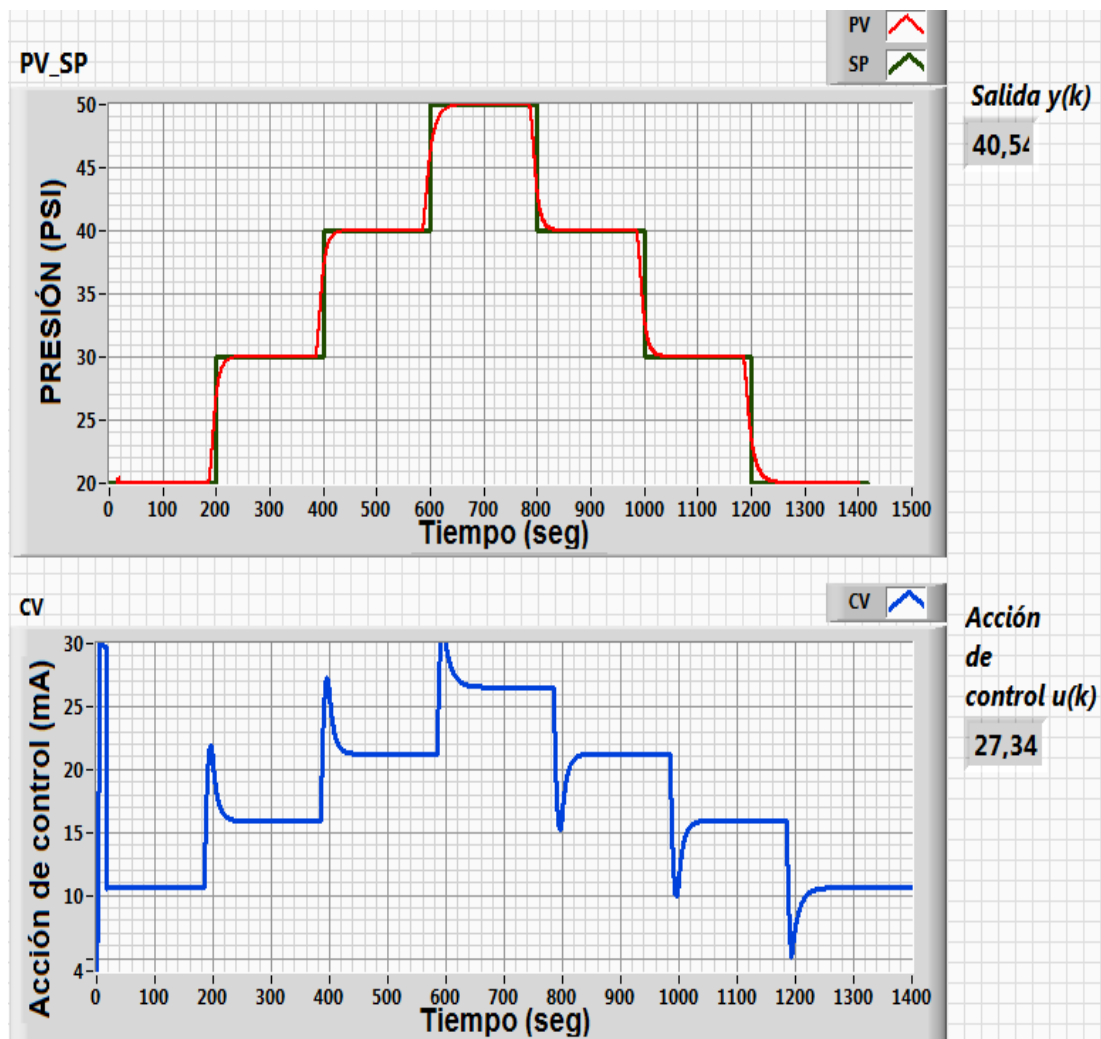
Como se puede observar no existe control al aplicar el parámetro de la matriz N.

- ❖ Las restricciones del control de la Figura 2.36, corresponden a los valores reales con los que trabaja el proceso.
- ❖ La restricción de la acción de control ( $u$ ) es el valor de la entrada de 4 a 20 mA con la cual se va a controlar el porcentaje de apertura y cierre del actuador.
- ❖ La restricción de salida ( $y$ ) corresponde al rango de trabajo del transmisor que es de 20 a 50 PSI.
- ❖ La restricción de la tasa de cambio de la acción de control permite controlar la velocidad del controlador de forma independiente ya sea de subida o de bajada.



Como se puede observar las restricciones son los mismos valores con que se trabaja para realizar la modelación de la planta de presión.

En la siguiente pantalla (Figura 2.42) se puede observar el comportamiento del controlador MPC ante diferentes cambios de setpoint (punto de consigna).



**Figura 2.42:** Respuesta del sistema implementado el controlador MPC.

Elaborado por: Curay, D. y Vaca, A.

Cabe señalar que los datos colocados en esta simulación son una aproximación a la implementación real. En el anexo C se puede observar las gráficas de la comparación dinámica entre los controladores MPC y PID implementados en simulación a diferentes puntos de referencia.

En la **Tabla 2.5** se puede observar la respuesta transitoria de los controladores MPC y PID implementados en simulación.

**Tabla 2.5.**

Comparación de la respuesta transitoria obtenidos en simulación.

Referencia "SP" (PSI)	MPC		PID	
	Sobreimpulso (%)	Tiempo de establecimiento(seg)	Sobreimpulso (%)	Tiempo de establecimiento (seg)
20-30	0	42	0	46
30-40	0	35	0	36
40-50	0	40	0	41
20-50	0	40	0	44
50-35	0	35	0	42
35-20	0	35	0	40
<b>Promedio</b>	0	37.8	0	41.5

Elaborado por: Curay, D. y Vaca, A.

En la tabla 2.6 se puede observar el resultado obtenido en la simulación del porcentaje de trabajo que realizará el actuador mediante la acción de control proporcionada para que la variable del proceso llegue al valor de set-point deseado.

**Tabla 2.6.**

Porcentaje de trabajo del "CV" obtenidos en simulación.

Referencia "SP" (PSI)	MPC	PID
	Acción de control (CV) %	Acción de control (CV) %
20-30	55.7	87.5
30-40	76.9	82.6
40-50	92	92.3
20-50	92	92.3
50-35	65.3	76.9
35-20	63.4	63.4
<b>Promedio</b>	58	82.5

Elaborado por: Curay, D. y Vaca, A.

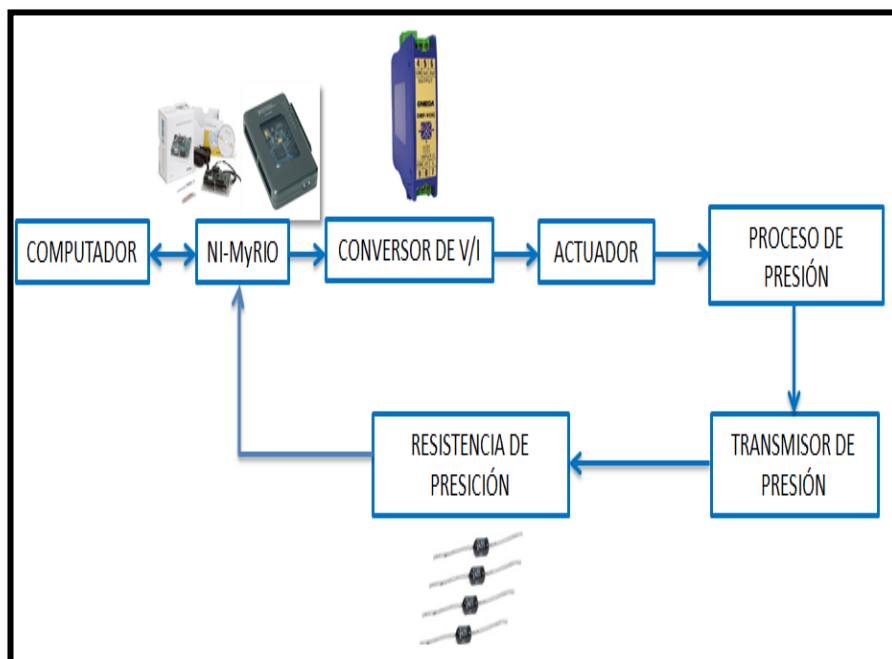
## CAPÍTULO 3

### IMPLEMENTACIÓN DE LOS CONTROLADORES MPC Y PID

En el presente capítulo se describe la implementación en hardware y software del controlador MPC y PID en tiempo real que opera de forma autónoma, además el acondicionamiento de las señales de entrada/salida a utilizar y como se acopla al sistema embebido NI Single Board RIO (NI SBRIO 9636) para adquirir y procesar las señales.

#### 3.1 Implementación en hardware

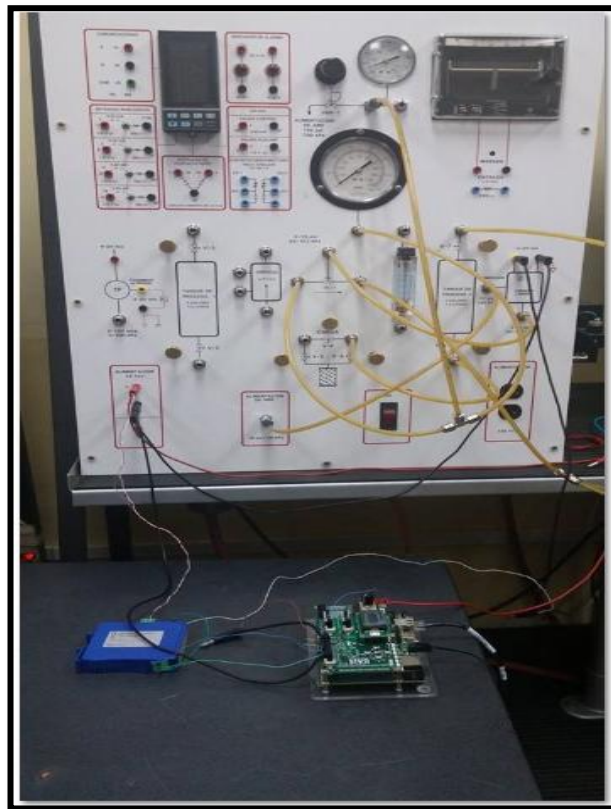
Para la implementación del controlador MPC y PID partimos del hardware, el cual es descrito por el diagrama de bloques (ver Figura 3.1), consta de un computador utilizado para el monitoreo mediante un HMI, el dispositivo (sbRIO, myRIO) donde se implementa el controlador MPC y PID, como los proceso industriales trabajan con corriente es necesario utilizar un convertor de V/I, debido al dispositivo (sbRIO, myRIO) porque únicamente tiene entradas y salidas de voltaje por ello se utiliza el convertor OMEGA.



**Figura 3.1:** Implementación del MPC y PID por hardware.

Elaborado por: Curay, D. y Vaca, A.

La señal generada del conversor Omega es enviada al actuador el cual limita la presión suministrada al proceso, seguido de esto el transmisor genera la corriente respectiva de acuerdo a la presión medida por este elemento, por ello, es necesario utilizar la resistencia de precisión para la conversión de corriente a voltaje (C/V) el cual es entregado nuevamente al dispositivo (sbRIO, myRIO) que lo utilizará para realizar el control respectivo, en la Figura 3.2 se puede observar la implementación de los controladores en hardware.



**Figura 3.2:** Conexión del hardware del controlador MPC y PID.

Elaborado por: Curay, D. y Vaca, A.

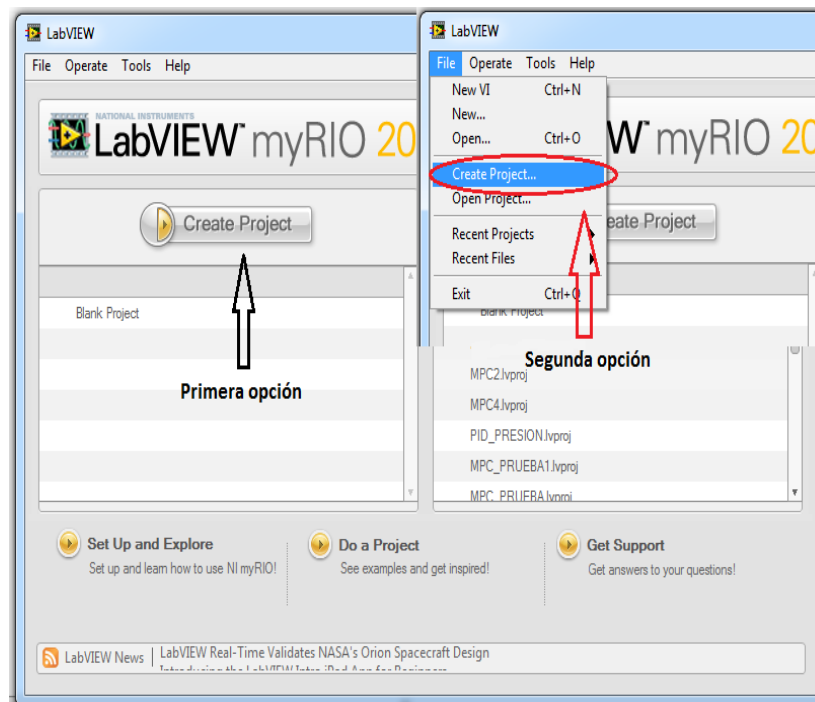
### 3.2 Implementación en software (labview)

Labview permite desarrollar proyectos, siendo esta característica muy importante a la hora de crear programas grandes, ya que permite ordenar y agrupar los VI's, administrar hardware como entradas y salidas, administrar los VI's para diferentes tareas, administrar grandes aplicaciones de Labview

entre otras actividades. Además, al crear un “*Project en Labview*” se tiene la opción de descargar el archivo en otros dispositivos en los cuales pueda ejecutarse el VI, estos dispositivos o máquinas son conocidas como tarjetas (targets).

National Instruments ofrece algunos dispositivos en los cuales se pueden descargar el VI como por ejemplo los SBRIO, MyRIO que son utilizados a nivel académico y a nivel industrial se encuentra el dispositivo NI CompactRIO con arquitectura embebida abierta los cuales funcionan de forma independiente de la PC y en tiempo real, es decir que la PC quedaría como un sistema únicamente de supervisión y en algunos casos hasta para envió de datos o parámetros utilizados en control.

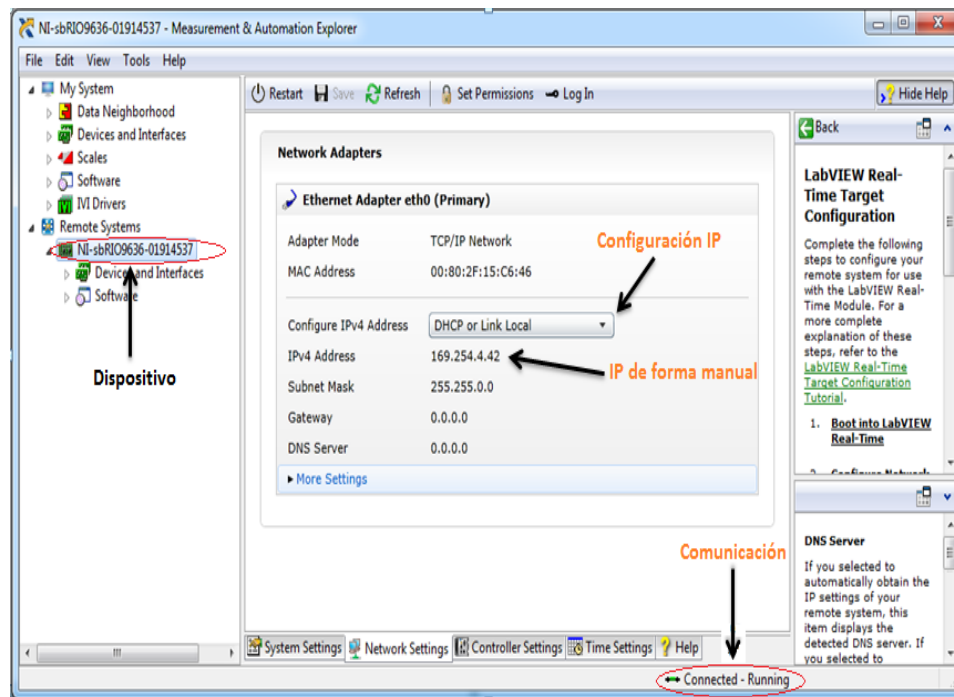
Para abrir la ventana de labview “*Proyect*” existen dos posibilidades, la primera: cuando se inicia Labview al seleccionar “*Create Project*”, la segunda opción es en la misma ventana de inicio al seleccionar “*File >> New Project*”, ver Figura 3.3.



**Figura 3.3:** Crear proyectos de Labview.

Elaborado por: Curay, D. y Vaca, A.

Antes de crear cualquier proyecto en labview, se debe ejecutar el programa MAX (NI Measurement & Automation Explorer) el cual contiene los drivers para los dispositivos (SBRIO, MyRIO, CompactRIO), ya que sin los drivers no funciona la tarjeta.



**Figura 3.4:** Configuración de los dispositivos mediante MAX.

Elaborado por: Curay, D. y Vaca, A.

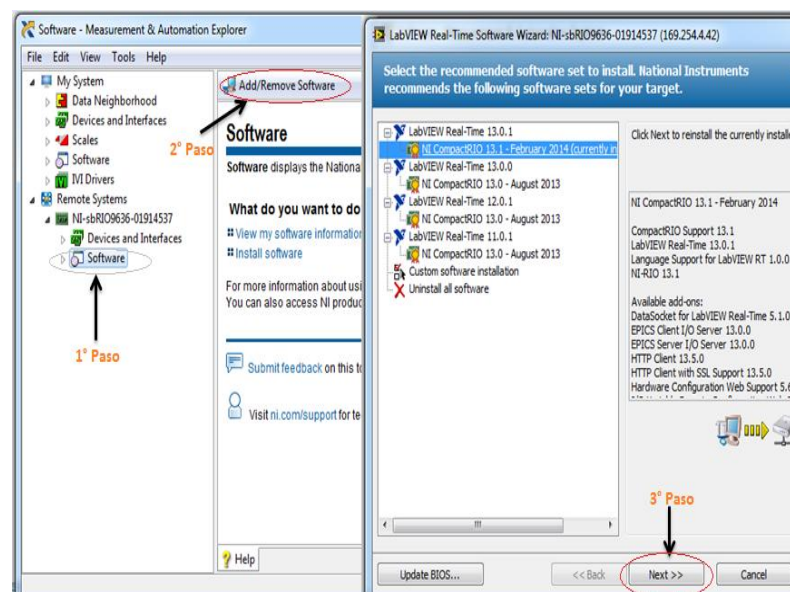
MAX busca los dispositivos que estén conectados a la computadora si los encuentra los muestra como dispositivos remotos debido a que utiliza el protocolo Ethernet para la comunicación, en la Figura 3.4 se observa las configuraciones de las tarjetas de NI con el programa MAX.

En la figura anterior se puede observar que el MAX encuentra el tipo de módulo que está conectado, en este caso es "NI-SBRIO", además permite asignar una IP o revisa si tiene alguna asignada y para la configuración en tiempo real existen dos opciones para conectar el módulo con la PC, la primera es conectar el dispositivo en una red LAN (usando DHCP) y otra directamente usando un cable cruzado para estos casos el dispositivo y la PC necesitan una dirección IP.

La configuración de la IP puede ser automática (DHCP) o estática (static) y para la operación remota ambos necesitan estar en la misma subred.

Si el dispositivo no se comunica con la red creada, sea una LAN (Red de Área Local) o directo con otra PC, se puede usar el switch IP RESET el cual se encuentra en la tarjeta del dispositivo y tiene como función restaurar de forma manual el dispositivo a opciones de fábrica, otra opción es deshabilitar el firewall de la computadora. Por último se puede verificar que existe comunicación del dispositivo con la PC, observando en la parte inferior derecha de la Figura 3.4 el indicador verde *“Connected - Running”*, el cual establece la existencia de comunicación.

Finalizado la comunicación se procede a instalar los driver al dispositivo, siendo los más importantes Labview Real-Time y Labview FPGA para controlar el dispositivo además el usuario tiene la opción de elegir todos los drivers necesarios dependiendo de la aplicación que le vaya a dar. Se debe tener en cuenta que la memoria del dispositivo es limitada. En la siguiente pantalla (Figura 3.5) se visualiza el procedimiento para descargar los drivers al dispositivo.



**Figura 3.5:** Instalación de drivers.

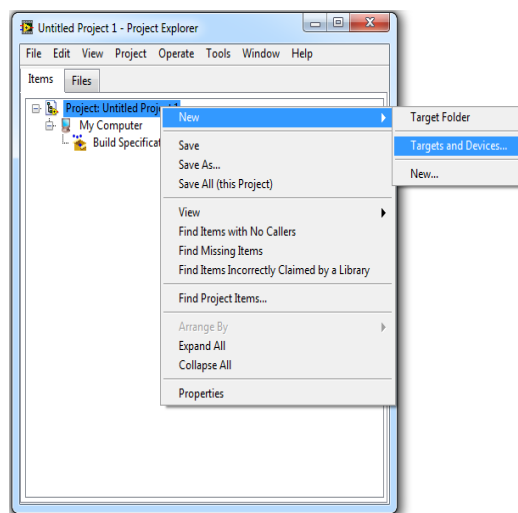
Elaborado por: Curay, D. y Vaca, A.

- ❖ El primer paso es colocarse en la opción “*Software*”, el cual muestra los componentes instalados en la tarjeta de Labview Real-Time target.
- ❖ Segundo paso es dar click en la opción “*Add/Remove software*”, que permite instalar los driver al destino remoto que se encuentre conectado.
- ❖ Finalmente se debe seguir las indicaciones de las pantallas hasta finalizar con la descarga correcta de los drivers en el dispositivo.

Para la implementación de los controladores es necesario que la tarjeta posea los drivers de toolbox “Control Desing & Simulation” el cual permitirá utilizar los componentes del MPC y PID.

### 3.3 Estructura del proyecto

Una vez realizado la comunicación del dispositivo se empieza con la etapa de adquisición e implementación con la tarjeta sbRIO, la cual se inicia creando el proyecto.



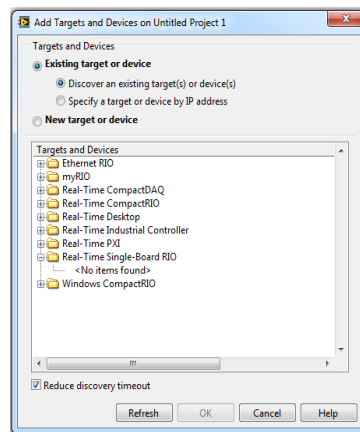
**Figura 3.6:** Añadiendo dispositivo nuevo.

Elaborado por: Curay, D. y Vaca, A.

Labview ofrece diferentes opciones para empezar a trabajar, pero es recomendable elegir un proyecto en blanco el cual se adaptará a todos los cambios que este proyecto exige.



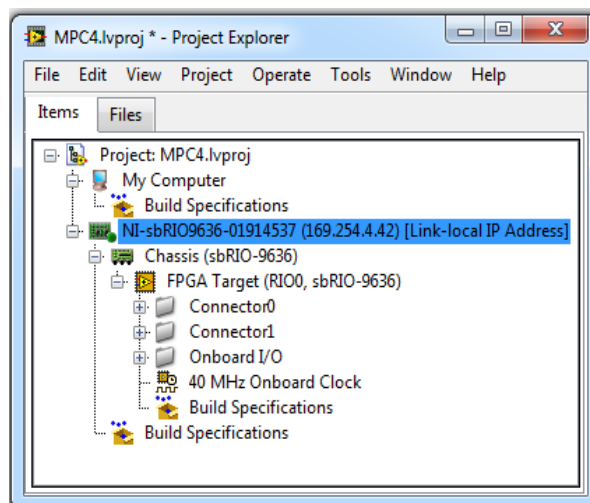
- ❖ Se añade la tarjeta de trabajo en la cual se implementará los controladores. En la Figura 3.6 se indica los pasos.
- ❖ En la pantalla (Figura 3.7) que es la ventana “*Add Targets and Devices*”, se presenta en una lista de módulos para trabajar, en esta opción se elige Real-Time sbRIO, posteriormente el programa empieza a detectar el equipo y accesorios adicionales.



**Figura 3.7:** Detectando el dispositivo.

Elaborado por: Curay, D. y Vaca, A.

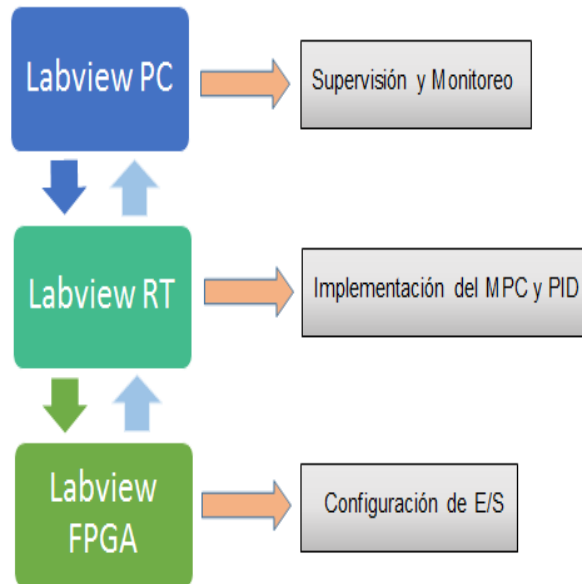
Al realizar este procedimiento se observa (Figura 3.8) los componentes del dispositivo, Chasis, FPGA y los accesorios que vienen ligados a la tarjeta.



**Figura 3.8:** Componentes de la tarjeta.

Elaborado por: Curay, D. y Vaca, A.

Agregado el dispositivo se comienza a estructurar el proyecto el cual consta de tres partes principales:



**Figura 3.9:** Estructura del proyecto.

Elaborado por: Curay, D. y Vaca, A.

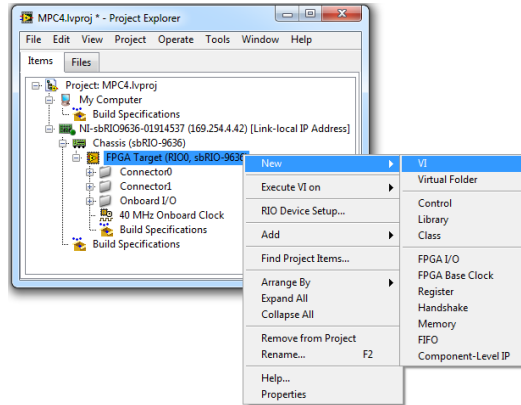
Para programar el FPGA y el controlador RT se necesita que la sbRIO esté conectada a una computadora que tenga instalado Labview, el modulo Real-Time, el módulo FPGA y los drivers para poder comunicar la tarjeta con el PC.

### 3.3.1 Labview FPGA.

Primero se trabaja con el FPGA puesto que las entradas, salidas ya sean analógicas o digitales están conectadas a él, entonces se debe realizar algún tipo de procesamiento para posteriormente enviar los datos adquiridos al microprocesador RT o viceversa, es decir enviar algún dato al exterior.

Como conocimiento previo se conoce que la programación común de un FPGA es por medio de VHDL siendo esta característica un inconveniente importante a la hora de utilizar la FPGA, pero labview tiene la ventaja de permitir al usuario programar de manera gráfica sin la necesidad de conocer

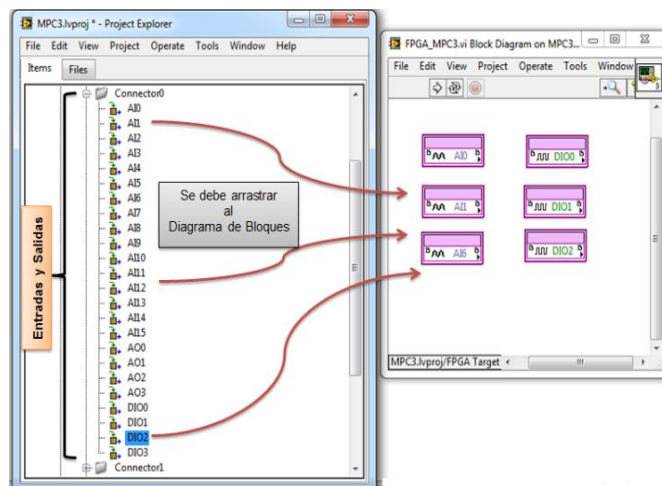
VHDL, además ahorra muchas líneas y páginas de código secuencial, también ahorra tiempo en la depuración y compilación del código.



**Figura 3.10:** Nuevo VI para la FPGA.

Elaborado por: Curay, D. y Vaca, A.

Para empezar a programar la FPGA se debe crear un nuevo VI (ver Figura 3.10), en donde se programa las configuraciones de las entradas y salidas que el dispositivo dispone. La FPGA dispone de varios conectores que están enlazados a ella, cada conector posee el nombre de cada E/S en la cual se puede acceder desplegando la pestaña de cada conector, para la manipulación de la E/S que se desee utilizar se debe arrastrar y soltar desde el Project Explorer hacia el diagrama de bloques, como se observa en la Figura 3.11.



**Figura 3.11:** Manipulación de las entradas y salidas.

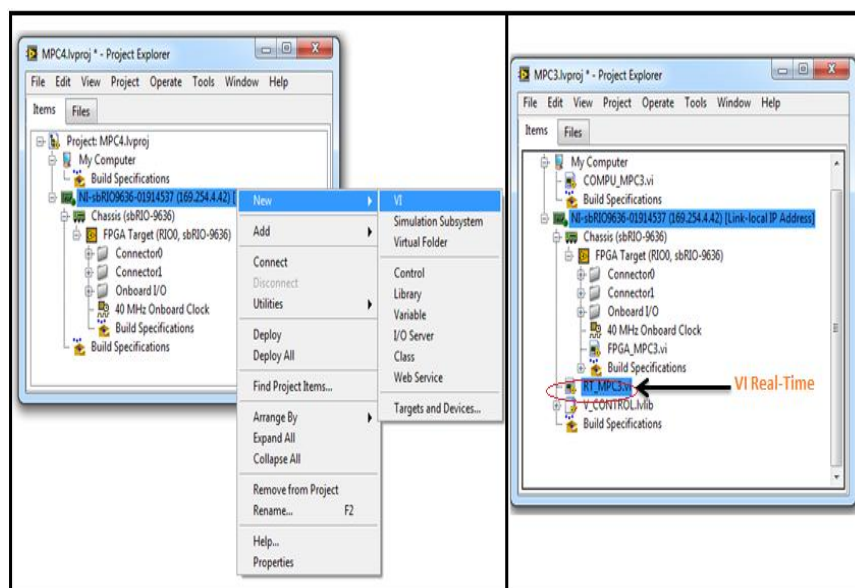
Elaborado por: Curay, D. y Vaca, A.

Finalmente para poder descargar el sistema de control FPGA, es necesario compilar el VI, al compilar el archivo, Labview FPGA determina el área utilizada en el FPGA, según el número de slices, flip-flops, lookup tables (LUTs), multiplicadores y bloques de memoria RAM utilizados en el diseño.

- ❖ Primero se guarda el proyecto FPGA creado en una carpeta nueva, luego se da click en RUN, el programa enviará al proyecto a compilar de manera automática, dependiendo del programa el tiempo necesario para la compilación tardará varios minutos.
- ❖ La compilación permite transformar la programación grafica realizada en el panel frontal de Labview en lenguaje VHDL que es reconocida por la tarjeta.

### 3.3.2 Labview RT.

Para que la sbRIO funcione es primordial que el controlador en tiempo real este programado, debido a que está conectado directamente con el FPGA y recibe la información de ella para realizar algún tipo de procesamiento o control de datos.



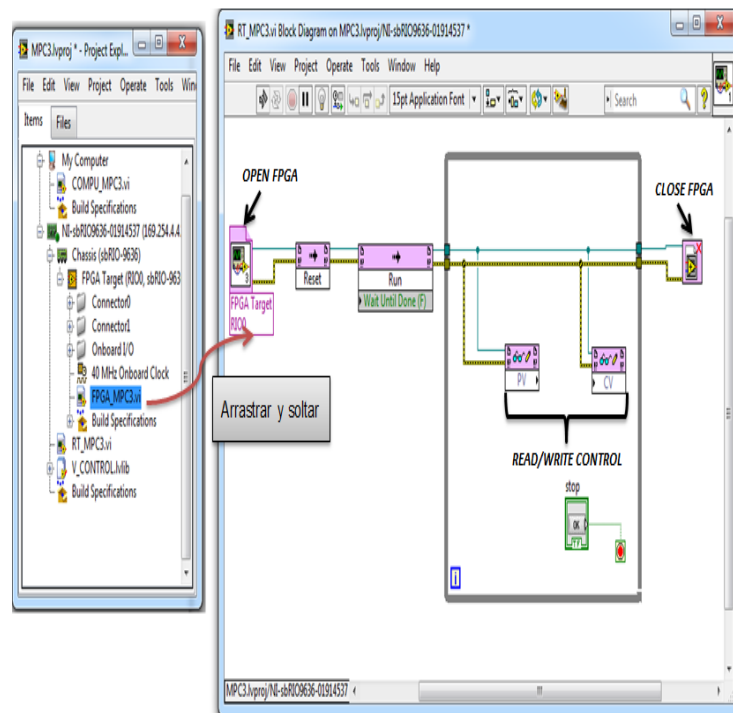
**Figura 3.12:** Creación de un VI en Real-Time.

Elaborado por: Curay, D. y Vaca, A.

Sobre el diagrama de bloques del nuevo VI se enlaza la FPGA con el chasis de la sbRIO para realizar el controlador en “*Real-Time*”, mediante la paleta de funciones de “FPGA-Interface”, esta paleta contiene los siguientes subVI’s:

- ❖ Open FPGA VI Reference: permite abrir o hacer referencia al Labview FPGA-VI situado en el mismo proyecto. Al cargar el VI de la FPGA permite utilizar las entradas que se configura en la FPGA.
- ❖ Close FPGA VI Reference: Cierra la FPGA del proyecto.
- ❖ FPGA Interface – Read/Write Control: hace referencia a los terminales de control del FPGA credo, esta función puede ser de lectura o escritura de pendiendo de la programación.

Para utilizar las características de la FPGA en la RT se debe arrastrar el VI creado del proyecto FPGA y soltarlo en el subVI “Open FPGA VI reference”, en la Figura 3.13 se puede observar la comunicación entre la RT y la FPGA.



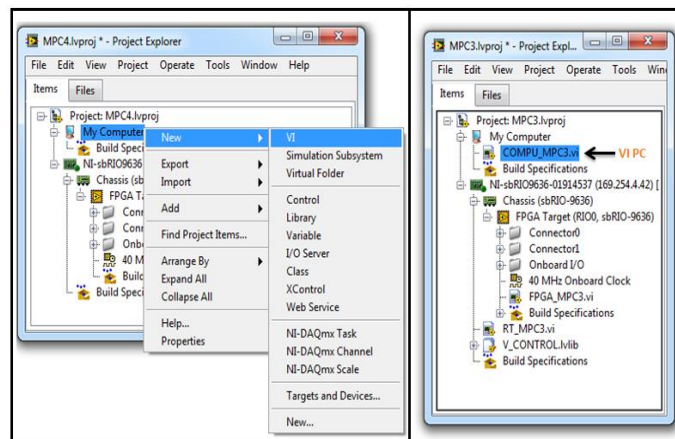
**Figura 3.13:** Creando el REAL-TIME VI.

Elaborado por: Curay, D. y Vaca, A.

### 3.3.3 Labview PC.

Los dispositivos creados por National Instrument permiten realizar proyectos de controladores, procesamiento entre otras aplicaciones, estos equipos trabajan de forma independiente, es decir se programa cualquier actividad en los módulos y estos los realizaran pero existen aplicaciones en las cuales es necesario comunicar datos entre computadores u otros dispositivos para visualizar el comportamiento de variables del sistema o simplemente actualizar parámetros de algún tipo de controlador.

Estos dispositivos no permiten visualizar el comportamiento del proceso o algún tipo de alarma que se pueda activar, entonces es recomendable utilizar un VI creado en la PC que tendrá la función de monitoreo y/o supervisión. Para crear el VI PC se debe realizar el siguiente procedimiento mostrado en la Figura 3.14. El programa de este VI dependerá de las exigencias del usuario, y de los parámetros que se desea monitorear.



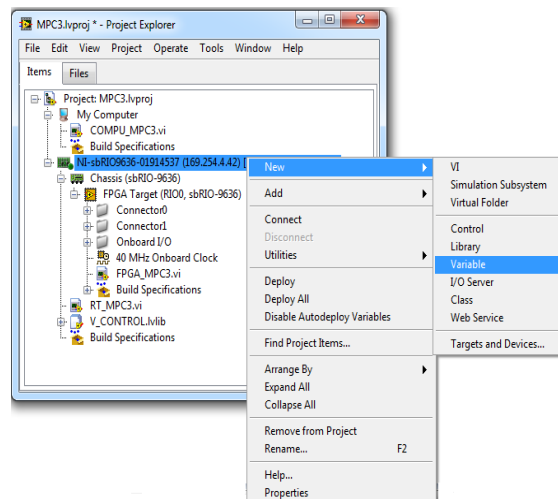
**Figura 3.14:** Nuevo VI en la PC.

Elaborado por: Curay, D. y Vaca, A.

#### a. Variables compartidas (Shared Variables)

Anteriormente se pudo observar que la comunicación entre la RT y la FPGA es de forma directa, es decir que permite enviar datos entre estas dos partes del proyecto sin ningún problema, por otra parte la comunicación entre la PC y la tarjeta RT no permite enviar datos de forma directa sin

importar que estos VI sean del mismo proyecto, entonces para resolver este problema Labview ofrece una solución con el uso de las variables compartidas o llamadas también “Shared variables”.



**Figura 3.15:** Creando las variables compartidas.

Elaborado por: Curay, D. y Vaca, A.

La función de estas variables es compartir información entre VI's de un ordenador, entre ordenadores conectados en red o entre programas ejecutándose en hardware diferente (PC a RT-FPGA), en otras palabras con las “Shared variables”, los VI's en diferentes maquinas o redes pueden leerse desde o escribirse hacia las variables.

Las variables compartidas pueden crearse en la PC (My Computer) o la RT (de la tarjeta), siguiendo el proceso mostrado en la Figura 3.15 donde posteriormente se presentara el cuadro de dialogo (ver Figura 3.16) para la configuración de las variables. El cuadro de dialogo presentan diferentes opciones para elegir a continuación se detalla:

### Variable

- ❖ Tipo de dato: Permite elegir entre un gran número de tipos de datos estándar para una nueva variable compartida.
- ❖ Tipo de variable: Permite elegir el modo de acceso al programa, entre las principales se tiene: “Network-Published”, que permite compartir la

variable a la red y pueda ser accesible desde una computadora o un tarjeta remota, “Single Process” permite que la variable sea accesible únicamente dentro de la misma PC o máquina en la cual se está programando, es decir que transfiere datos entre dos ubicaciones diferentes en el mismo VI que no pueden ser conectados por medio de cables.

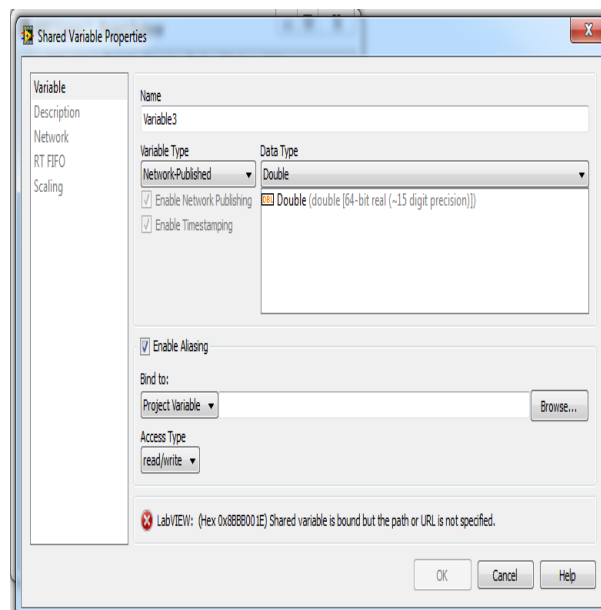
- ❖ Nombre (Name): Permite otorgar una identificación a las variables.
- ❖ Tipo de acceso (Access Type): Permite elegir si la variable es de entrada/salida o ambas.

## Network

- ❖ Use buffering: Permite activar el almacenamiento en búfer, el cual especifica el tamaño de la memoria intermedia en unidades del tipo de datos que se esté utilizando.

## RT FIFO

- ❖ Enable RT FIFO: Permite que la variable compartida trabaje en tiempo real.

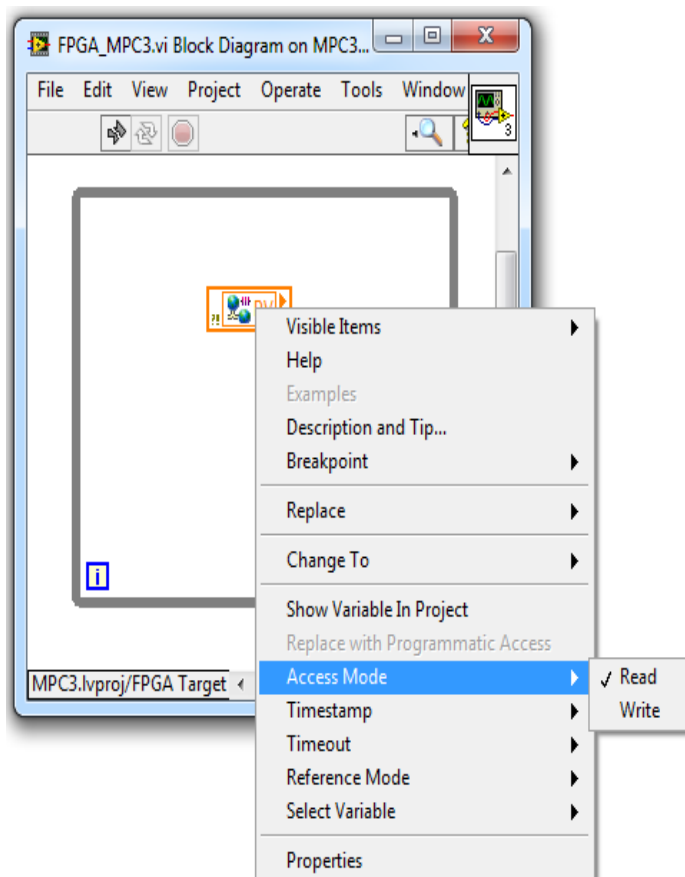


**Figura 3.16:** Propiedades de la variable compartida.

Elaborado por: Curay, D. y Vaca, A.



Para el uso de las variables compartidas creadas simplemente se debe arrastrar y soltar en la ventana de diagrama de bloques correspondiente.



**Figura 3.17:** Tipos de acceso.

Elaborado por: Curay, D. y Vaca, A.

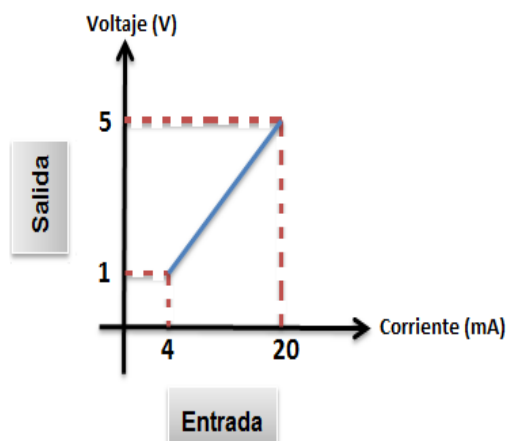
Para configurar la “Shared variable” en modo lectura o escritura se debe hacer click derecho en la variable y posteriormente presionar “Access Mode”, como se indica en la Figura 3.17.

### 3.4 Acondicionamiento de señales.

El sistema de adquisición empleando la tarjeta NI-Single Board RIO, debe pasar por una etapa de acondicionamiento antes que las señales sean enviadas o recibidas por la tarjeta. Como característica principal la tarjeta posee entradas y salidas analógicas de -10 a 10 voltios con una resolución de 16 bits.

### 3.4.1 Entrada de la SBRIO.

De la tarjeta se utiliza una entrada la cual permite obtener la señal de realimentación para los controladores MPC y PID en otras palabras será “PV” del proyecto, entonces la señal adquirida es por medio del transmisor que trabaja con corriente de salida y representa de forma proporcional a la presión marcada por el transmisor. La corriente proporcionada por el transmisor es estándar, es decir de 4 – 20 mA, esto facilita la conversión de corriente a voltaje con solo utilizar una resistencia de 250 ohms, obteniendo como resultado un voltaje de 1 a 5 voltios en forma proporcional este acondicionamiento es realizado mediante hardware (ver Figura 3.18).



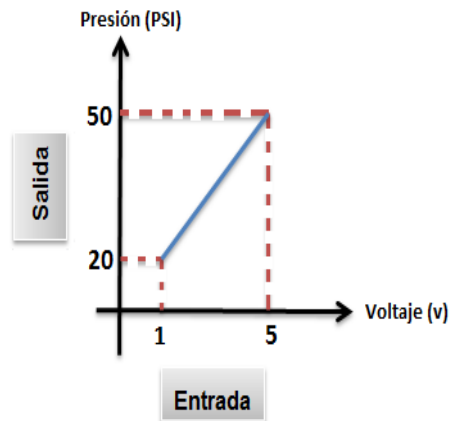
**Figura 3.18:** Acondicionamiento por hardware Input.

Elaborado por: Curay, D. y Vaca, A.

Este resultado se obtiene al aplicar la ley de Ohm (Ecuación (3.1)), así como se muestra en la ecuación siguiente:

$$V = I * R \quad (3.1)$$

Entonces la corriente que pase por la resistencia suministrará un voltaje proporcional dentro del rango de 1 a 5 voltios respectivamente. Posteriormente el valor que adquiere el controlador es el PV el cual debe ser en unidades de presión (PSI) para realizar un acondicionamiento en software para convertir el voltaje a presión (Figura 3.19).



**Figura 3.19:** Acondicionamiento por software (PV).

Elaborado por: Curay, D. y Vaca, A.

Para realizar esta conversión se utiliza la ecuación de la recta que viene representada de la siguiente manera:

$$y = m \cdot x + b \quad (3.2)$$

$$m = \frac{y_1 - y_0}{x_1 - x_0} \quad (3.3)$$

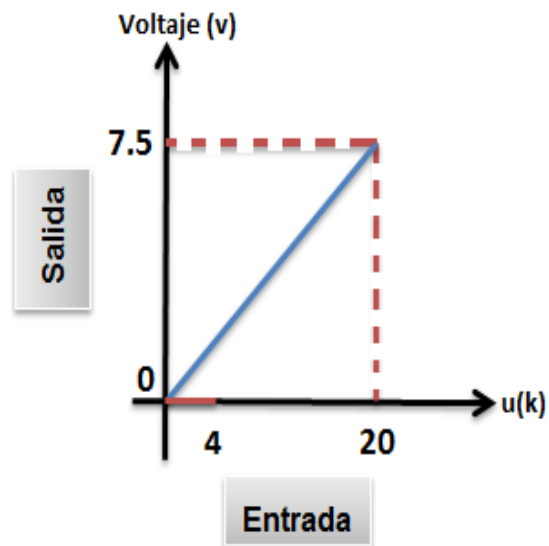
En el diagrama de bloques se realiza el escalamiento a través de multiplicaciones, sumas y restas necesarias para obtener el valor deseado para realimentar el controlador.

### 3.4.2 Salida de la SBRIO.

De la tarjeta se utiliza una salida la cual permite enviar la acción de control proporcionada por los controladores MPC y PID en otras palabras es el "CV" del proceso, este dato permitirá realizar el control de cierre y apertura del actuador, cuyo valor está en el rango de 4 a 20 que representara la corriente.

Pero como la tarjeta no posee salidas de corriente se utiliza un conversor de voltaje/corriente, donde al proporcionar a su entrada un voltaje de 0 - 7.5 voltios se obtiene a la salida una corriente de 4 – 20 mA. Entonces se debe realizar dos acondicionamientos.

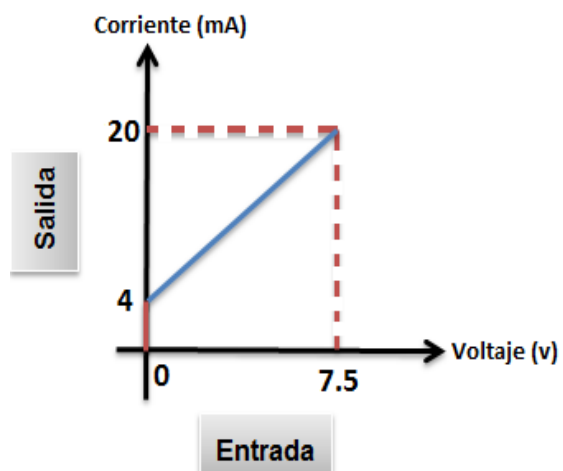
El primer acondicionamiento es realizado por software para convertir la acción de control ( $u(k)$ ) al rango de voltaje que necesita el conversor y seguido se aplica la Ecuación 3.2 (ver Figura 3.20).



**Figura 3.20:** Acondicionamiento por software Output.

Elaborado por: Curay, D. y Vaca, A.

El segundo acondicionamiento es realizado por hardware con la ayuda del conversor de V/C mencionado en el capítulo II, en la Figura 3.21 se observa los valores obtenidos.



**Figura 3.21:** Acondicionamiento por hardware (CV).

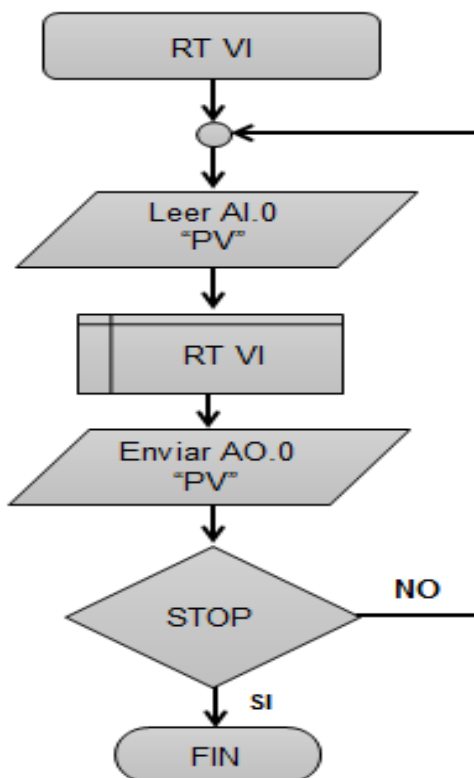
Elaborado por: Curay, D. y Vaca, A.

### 3.5 Programación del controlador MPC.

Definido los parámetros, se procede a implementar el controlador en tiempo real, el cual permitirá realizar el control del proceso mientras ella este en operación, es decir permitirá modificar los parámetros de control del MPC los cuales serán necesarios para una buena sintonización y así llevar el proceso al punto de referencia deseado.

#### 3.5.1 Programación de la FPGA

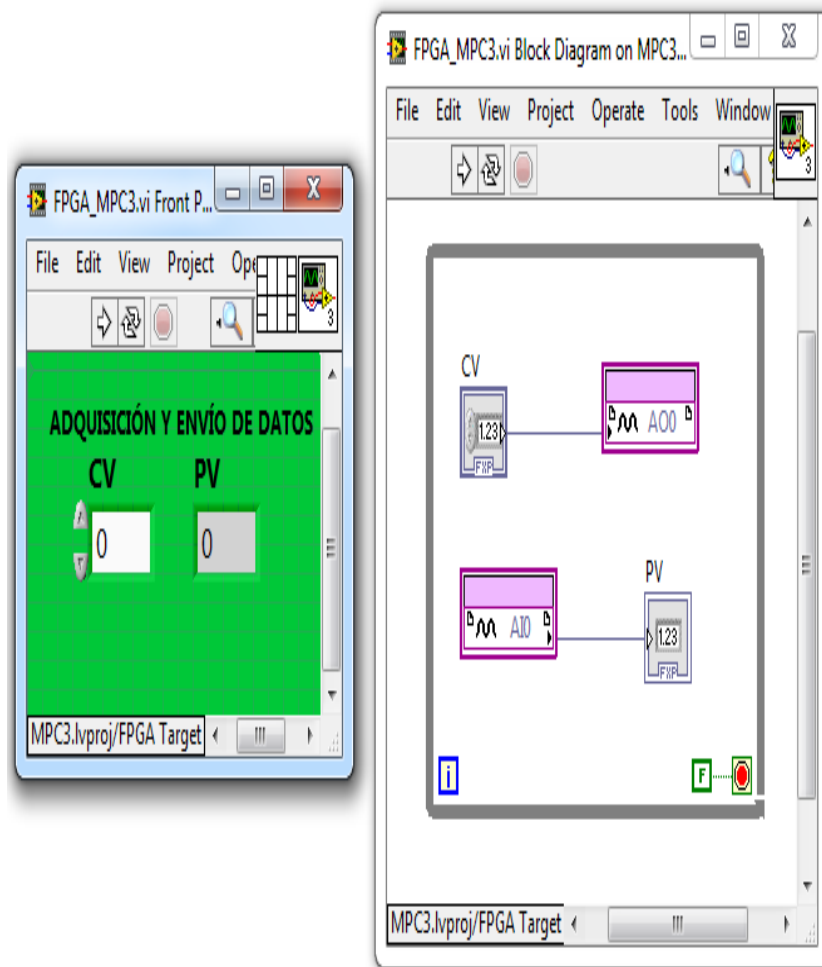
En la Figura 3.22 indica el proceso que cumple la FPGA mediante el diagrama de flujo, el cual tiene la función de enviar y recibir la señal del voltaje desde el proceso a la tarjeta sbRIO y viceversa.



**Figura 3.22:** Diagrama de flujo FPGA.

Elaborado por: Curay, D. y Vaca, A.

En la Figura 3.23, se indica la programación en Labview tanto del panel frontal como en el diagrama de bloques.



**Figura 3.23:** Programa de la FPGA en Labview.

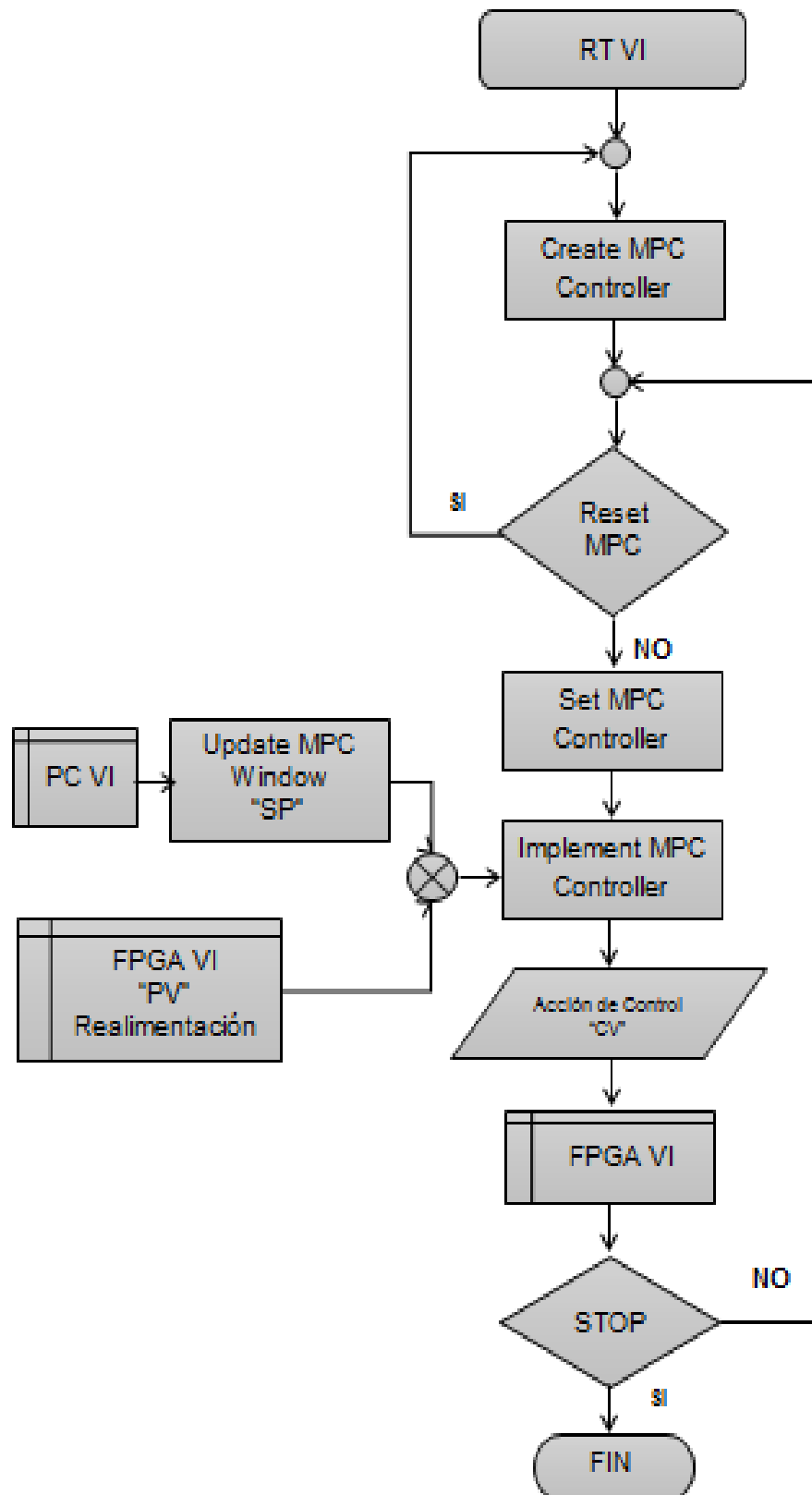
Elaborado por: Curay, D. y Vaca, A.

### 3.5.2 Programación en la RT.

En la Figura 3.24 está el diagrama de flujo RT VI e indica la secuencia de los subVI's los cuales forman parte del controlador MPC en Labview y cuya función fue descrita en el CAPITULO II en la sección de simulación del controlador.

Además de implementar el control se puede realizar aquí los acondicionamientos por software de los datos adquiridos por la FPGA.

En la Figura 3.25, se indica la programación en Labview tanto del panel frontal como en el diagrama de bloques realizado en REAL-TIME.



**Figura 3.24:** Diagrama de flujo RT.

Elaborado por: Curay, D. y Vaca, A.

## RT-- IMPLEMENTACIÓN DEL CONTRALADOR MPC

### State-Space Model

Model name:

Sampling Time:

A				B		
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

C			D		
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

### Equation

### Ptrs. Control

Prediction Horizon:

Control Horizon:

Initial Window:

Integral Action?

Reset:

STOP

### MPC Constraints (Dual)

<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0"/>	<input type="text" value="0,00"/>

### MPC Cost Weights

Output Error Weightings:

<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>

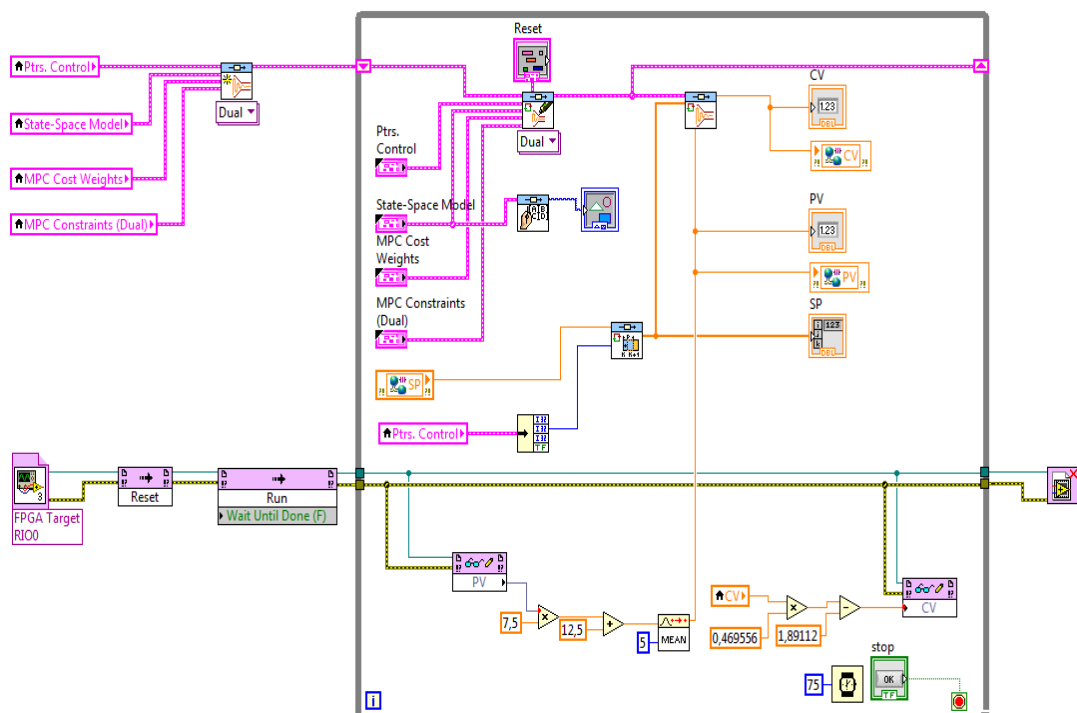
Control Action Change Weightings:

<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>

Control Action Error Weightings:

<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>
<input type="text" value="0"/>	<input type="text" value="0,00"/>	<input type="text" value="0,00"/>

SP:  CV:  PV:



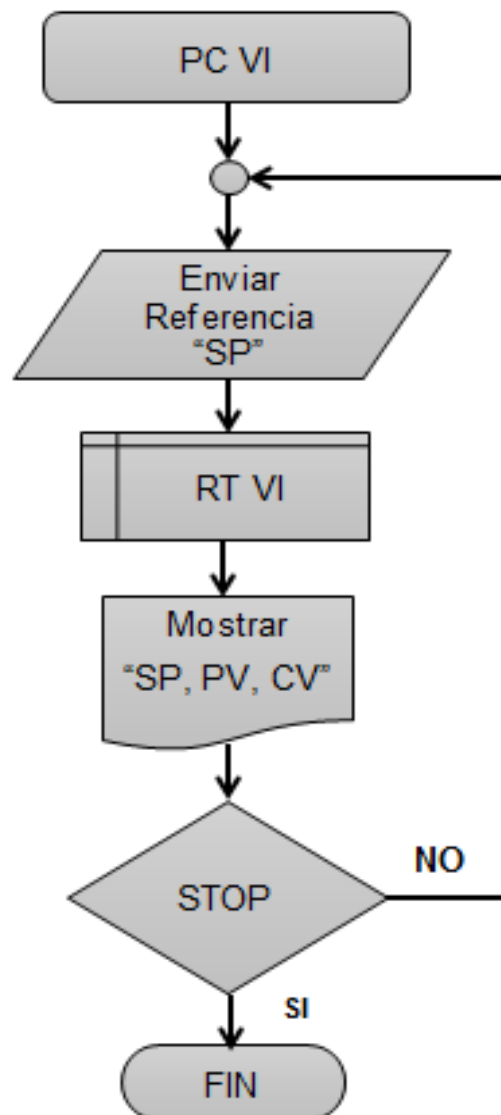
**Figura 3.25:** Programa en la RT mediante Labview.

Elaborado por: Curay, D. y Vaca, A.



### 3.5.3 Programación en la PC.

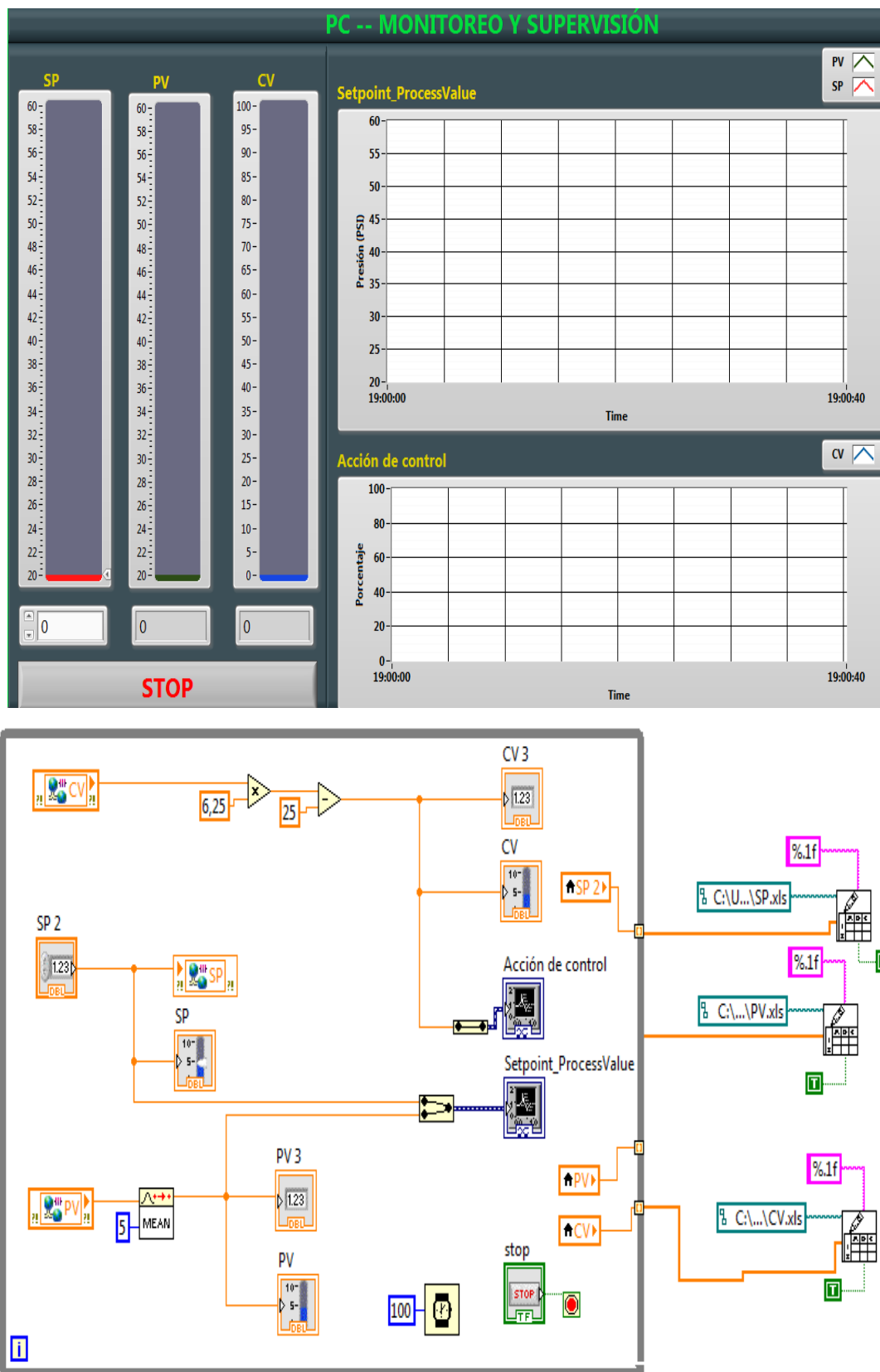
En la pantalla siguiente (Figura 3.26) se observa el diagrama de flujo de PC VI el cual permite visualizar todas las variables provenientes del proceso. Las cuales son enviadas desde la RT por medio de las variables compartidas (Shared variables).



**Figura 3.26:** Diagrama de flujo PC.

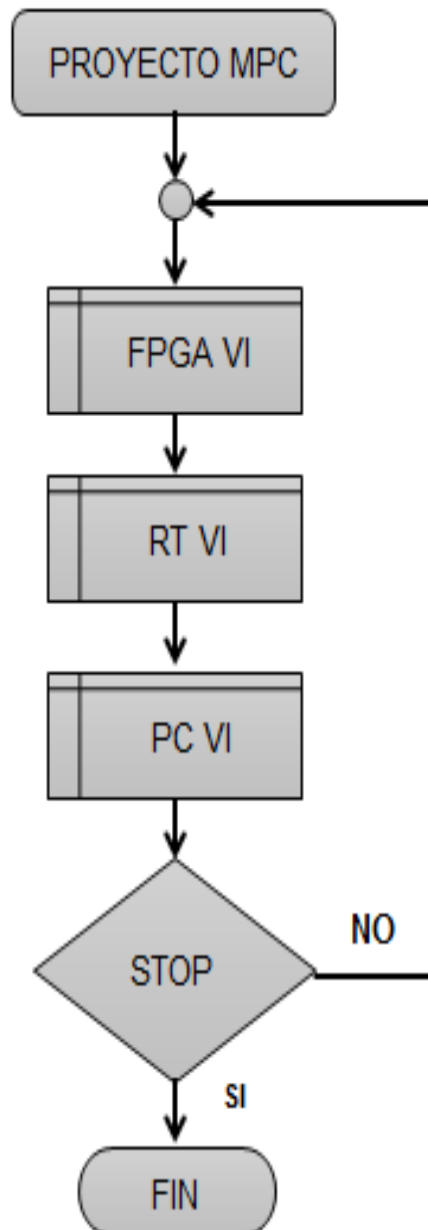
Elaborado por: Curay, D. y Vaca, A.

En la Figura 3.27, se indica la programación en Labview tanto del panel frontal como en el diagrama de bloques realizado en la PC.



**Figura 3.27:** Programa en la PC mediante Labview.  
Elaborado por: Curay, D. y Vaca, A.

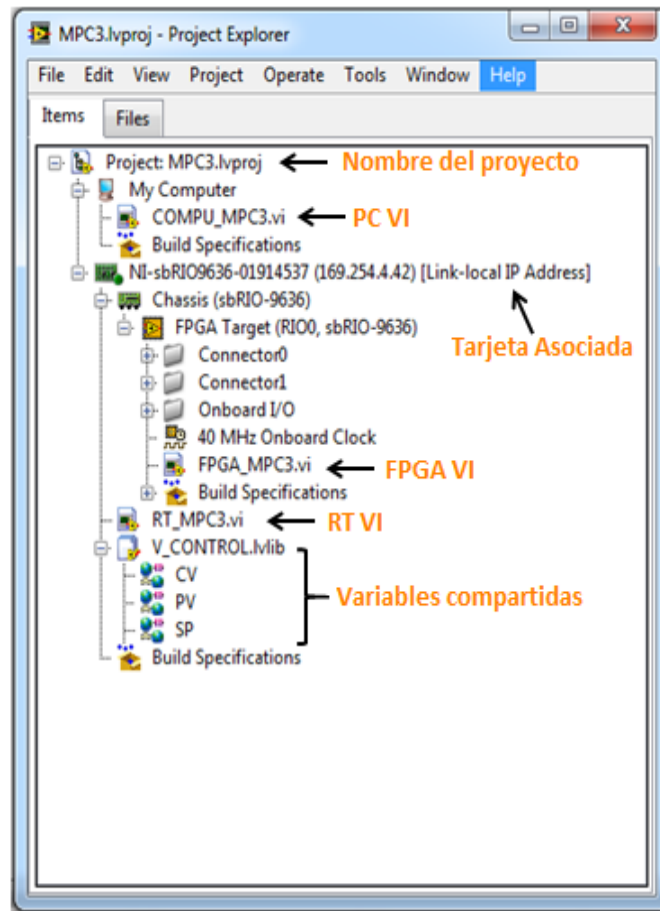
Entonces el programa principal llamado proyecto controlador MPC está dividido en tres etapas: adquisición, control, monitoreo y por último la terminación del proyecto (ver Figura 3.28).



**Figura 3.28:** Diagrama principal del proyecto MPC.

Elaborado por: Curay, D. y Vaca, A.

En la Figura 3.29, se indica el proyecto principal creado en Labview el cual contiene todos los VI del proceso.



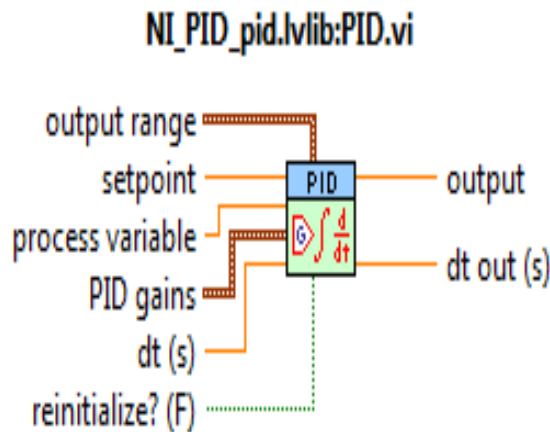
**Figura 3.29:** Proyecto creado en Labview.

Elaborado por: Curay, D. y Vaca, A.

### 3.6 Programación del controlador PID.

El PID es el más utilizado a nivel industrial y la implementación de este tipo de controlador es muy sencilla gracias al toolkit que posee Labview, donde este tipo de control es de carácter analógico, requiriendo para su funcionamiento la entrada de referencia (SP), la realimentación que se denomina variable del proceso (PV) y su salida será la acción de control (CV) que vendrá condicionada por los parámetros de sus acciones proporcional, derivativa e integral.

En labview podemos encontrarlo en el “Control Desing & Simulation” en el cual se elige el icono PID normal, en la Figura 3.30 se detalla las componentes del subVI PID.



**Figura 3.30:** SubVI PID.

Elaborado por: Curay, D. y Vaca, A.

Donde el icono del PID de Labview proporciona como estándar las siguientes características:

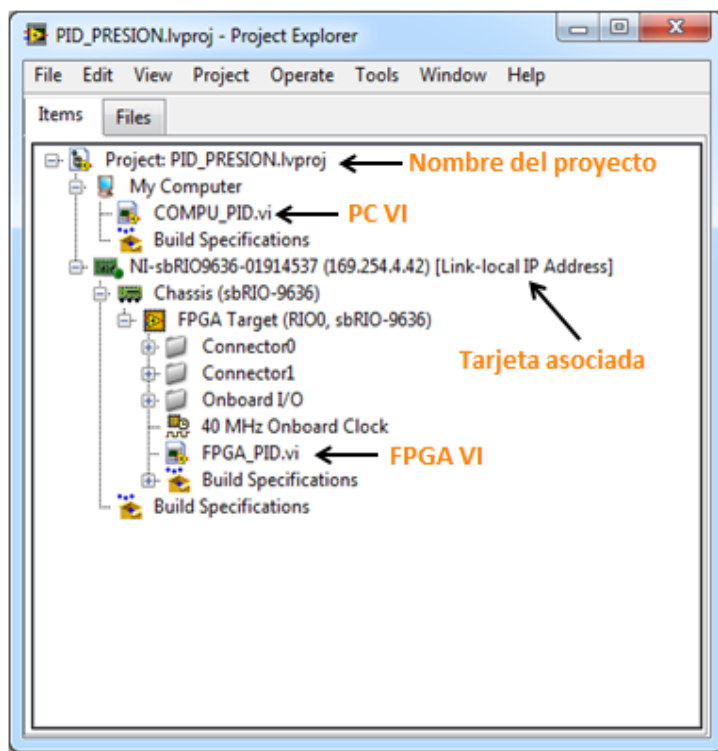
- ❖ Setpoint: Es la referencia del PID
- ❖ Process variable: Es el valor de la variable de proceso o de retroalimentación.
- ❖ PID gains (PID parámetros): Es el conjunto de las ganancias de las tres acciones: Proporcional, Integral, Derivativa.
  - Kc: Ganancia proporcional.
  - Ti: Constante de tiempo integral. Expresado en minutos. Valor 0 inhibe la acción
  - Td: Constante de tiempo derivativa. Expresado en minutos. Valor 0 desactiva la acción.
- ❖ Output rango (Rango de salida). Permite ingresar el valor mínimo y máximo de la salida del controlador. Valor mínimo por defecto -100 y valor máximo por defecto 100.
- ❖ Output (Salida): Salida del algoritmo de control, es decir proporciona la acción de control (CV).

Los puntos descritos anteriormente permiten implementar el controlador PID, cabe recalcar que los acondicionamiento de las variables son los mismo

del controlador MPC tanto en la salida y entrada de la tarjeta sbRIO, como los valores “PV” y “CV”.

### 3.6.1 Creando el proyecto PID.

De igual manera como el controlador MPC para la programación del controlador PID es similar y se empieza creando el proyecto el cual contiene dos VI en una la PC y otra en la FPGA. En la Figura 3.31 se observa la estructura del controlador PID.

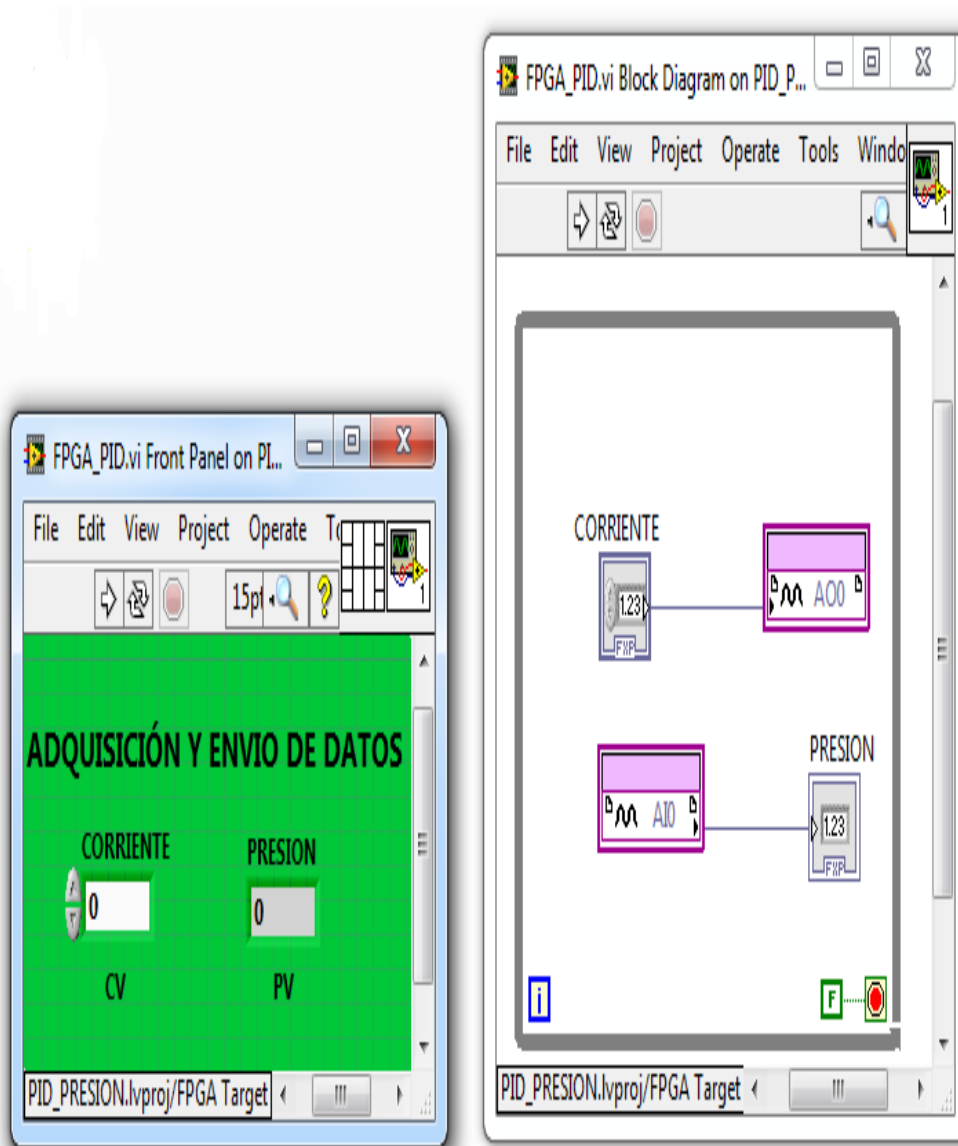


**Figura 3.31:** Estructura del proyecto PID.

Elaborado por: Curay, D. y Vaca, A.

### 3.6.2 Programación FPGA\_PID

Tiene la función de enviar y recibir la señal del voltaje desde el proceso a la tarjeta sbRIO y viceversa, estas señales se encuentran acondicionadas de acuerdo al requerimiento de cada equipo. En la Figura 3.32, se indica la programación en Labview tanto del panel frontal como en el diagrama de bloques.



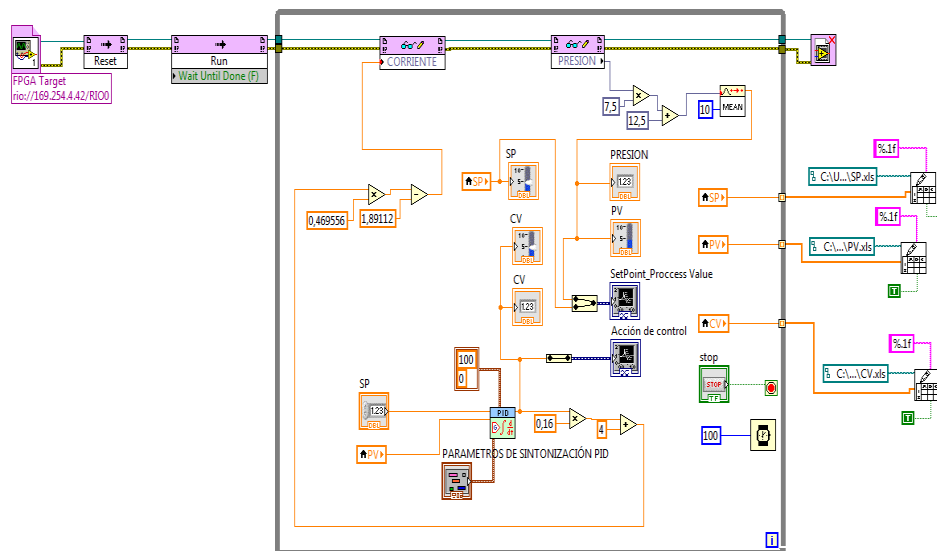
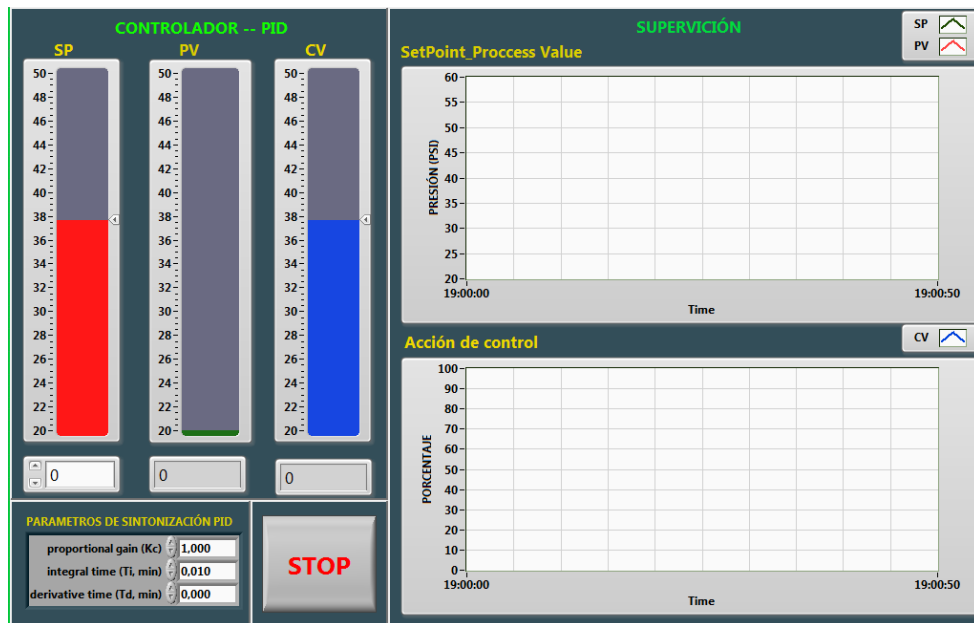
**Figura 3.32:** Programa en la FPGA\_PID mediante Labview.

Elaborado por: Curay, D. y Vaca, A.

### 3.6.3 Programación PC\_PID

En este VI se implementa el controlador PID con la ayuda del subVI de Labview, además se realiza los acondicionamientos por software de los datos adquiridos por la FPGA para el correcto funcionamiento del controlador.

En la Figura 3.33, se indica la programación en Labview tanto del panel frontal como en el diagrama de bloques del controlador en tiempo real.



**Figura 3.33:** Programa en la PC\_PID mediante Labview.

Elaborado por: Curay, D. y Vaca, A.

Cada sistema de control implementado debe garantizar en primer lugar la estabilidad del comportamiento en lazo cerrado para asegurar la estabilidad sin tener en cuenta la dinámica interna del sistema. Dependiendo de las características de cada controlador se realizara el respectivo análisis donde se observa el comportamiento de cada uno. En el Anexo B se puede observar la conexión de los dispositivos para realizar el control MPC en la planta de presión.



## CAPÍTULO 4

### ANÁLISIS DE RESULTADOS DE LA IMPLEMENTACIÓN

En este capítulo se detalla los resultados obtenidos de la implementación de los controladores (MPC y PID) en tiempo real, es decir cuando está en operación el proceso de presión.

Además se realiza la respectiva sintonización de los controladores, mediante experimentos pequeños de pruebas de ensayo y error para comprobar cuáles son los mejores valores para los distintos parámetros, de tal forma que cuando se realice el respectivo control al proceso de presión se apliquen valores fijos.

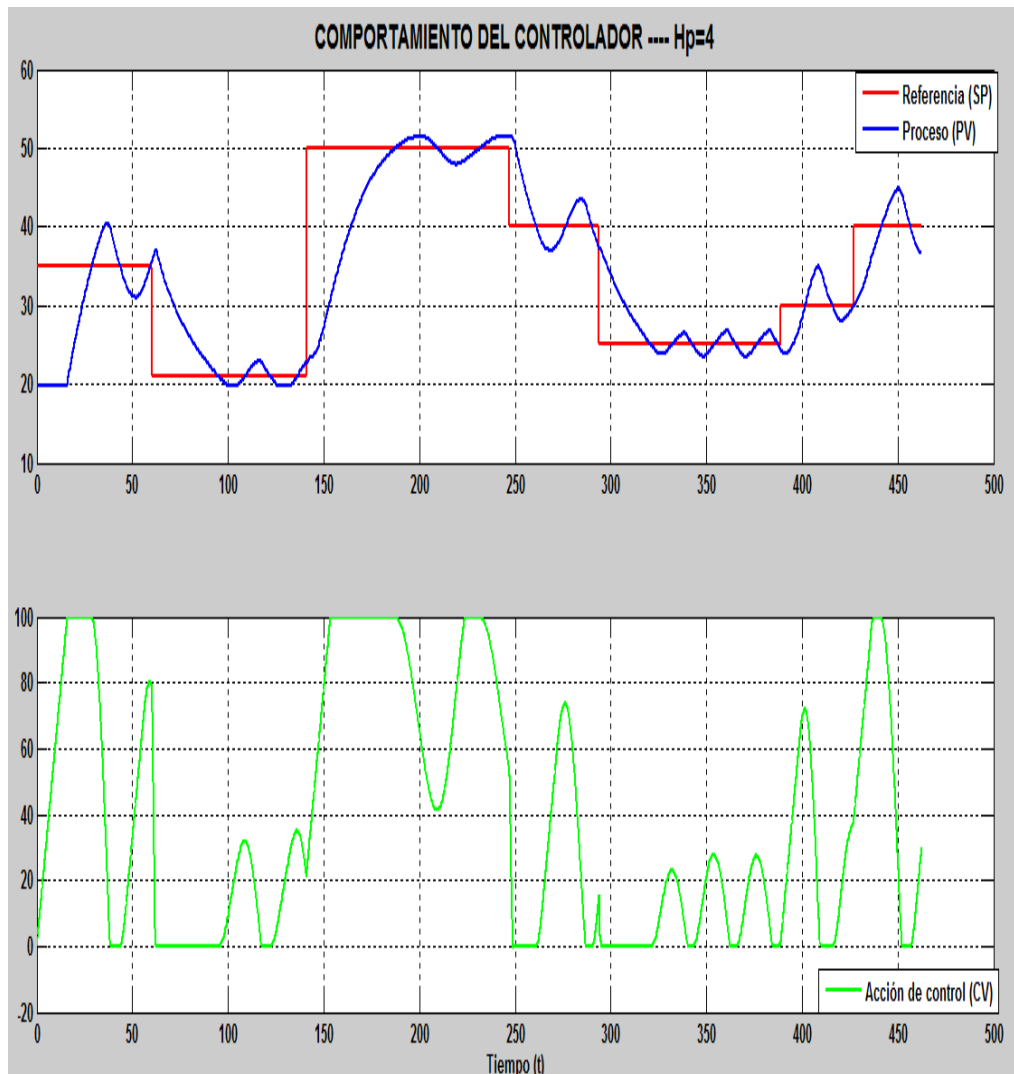
Al finalizar el capítulo se realiza la respectiva comparación entre los controladores con el propósito de observar cuál de ellos es el más eficiente en su respuesta.

#### 4.1 Análisis del controlador MPC en la planta de presión

Se empieza con la sintonización del MPC, donde se implementa un controlador predictivo cuyo horizonte de predicción variara de acuerdo al requerimiento del controlador para obtener una sintonización optima y el resto de los parámetros será igual a los de la simulación.

Los resultados reales obtenidos se ilustraran a continuación, en la cual se han registrado la referencia "SP" (trazos de color rojo), presión controlada del proceso "PV" (trazos color azul) y la señal de acción de control "CV" (trazos color verde) aplicado al actuador del proceso.

En la Figura 4.1, se implementa un controlador predictivo con un  $H_p=4$ ,  $H_c=1$  y una referencia de diferentes valores.

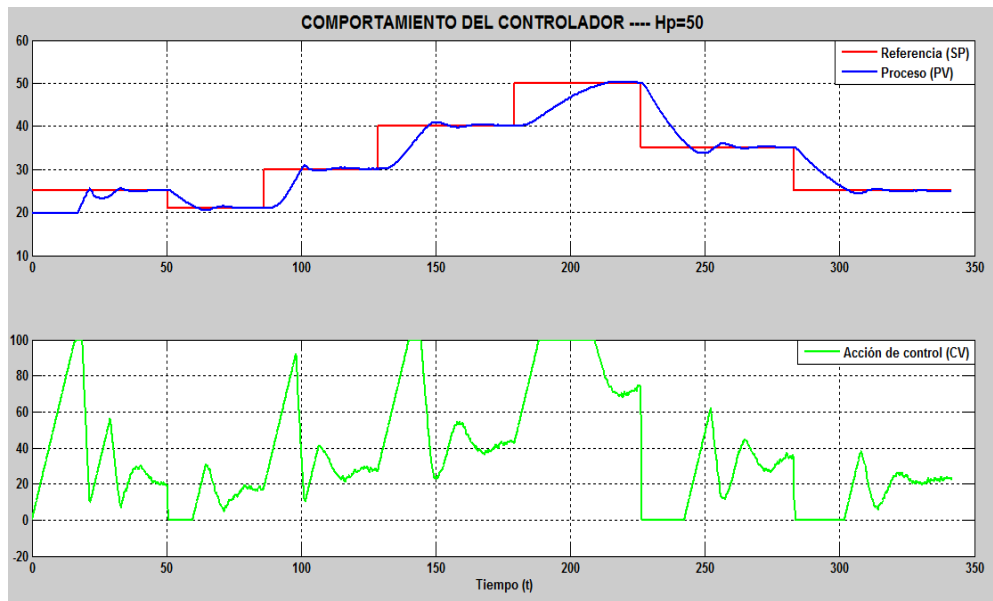


**Figura 4.1:** Respuesta real del proceso con  $H_p=4$ .

Elaborado por: Curay, D. y Vaca, A.

Como se puede observar en la Figura 4.1, la respuesta del proceso presenta una señal oscilatoria con respecto a la referencia y el intento del controlador para estabilizar el proceso es inútil debido a que provoca un control parecido al on - off y cual es perjudicial para el actuador.

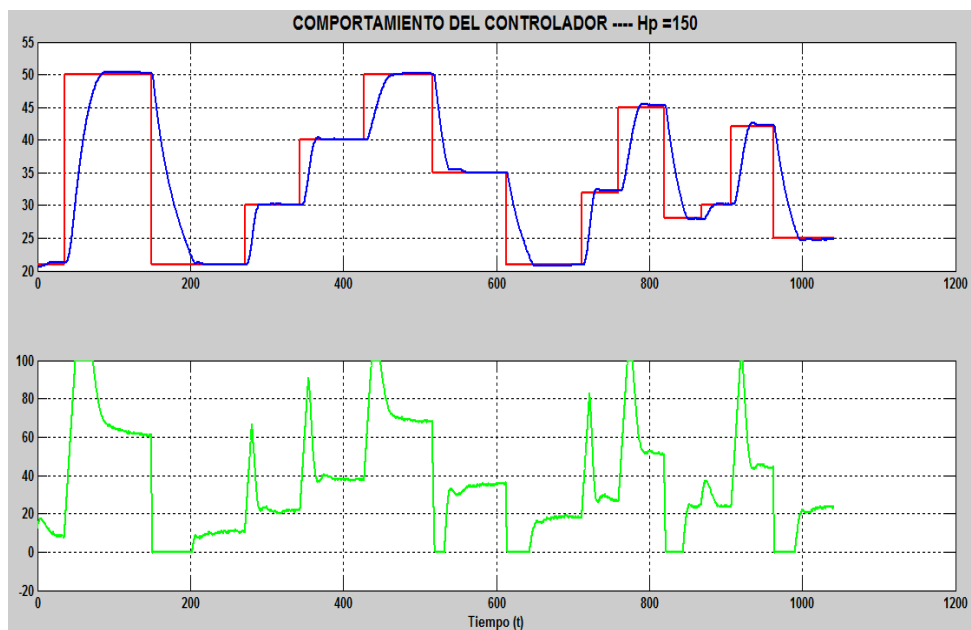
A medida que se aumenta el horizonte de predicción el controlador presenta una mejora considerable en la sintonización, debido a que aumenta la capacidad de predicción, en la Figura 4.2 se puede observar como disminuye la oscilación de la señal y el controlador ajusta el modelo experimental a la referencia.



**Figura 4.2:** Respuesta real del proceso con  $H_p=50$ .

Elaborado por: Curay, D. y Vaca, A.

En la Figura 4.3 se puede observar el comportamiento de la señal de acción de control la cual se anticipa a los cambios de la señal de referencia, debido a que el controlador proporciona el valor necesario para que el control alcance la estabilidad evitando los sobreimpulsos.

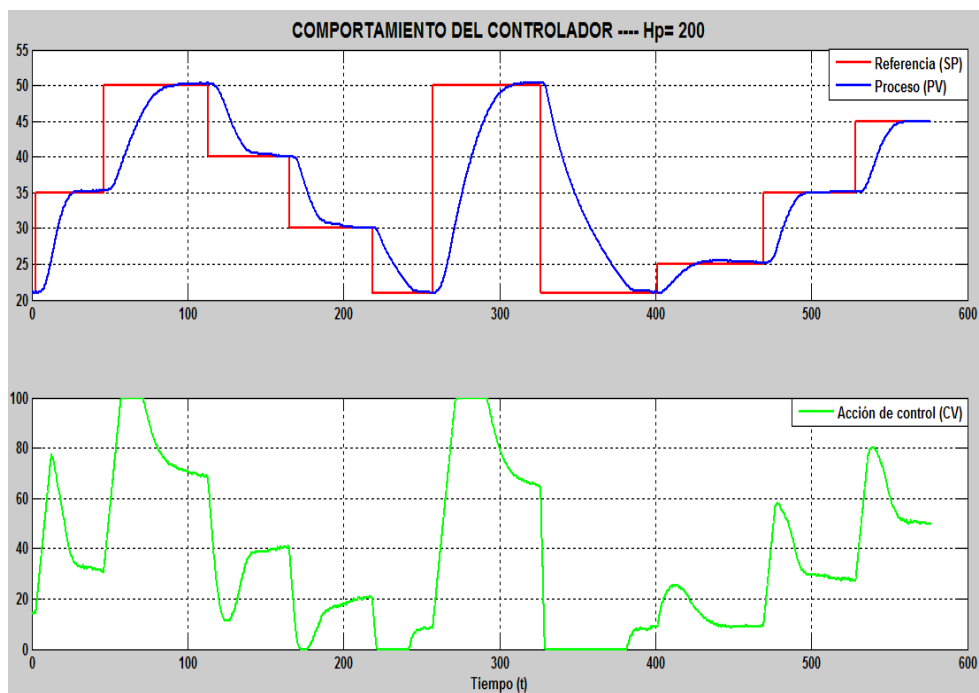


**Figura 4.3:** Respuesta del proceso con  $H_p=150$ .

Elaborado por: Curay, D. y Vaca, A.

Al aumentar el horizonte de predicción se obtiene un control óptimo y deseado sobre el proceso, pero la acción de control como se puede observar tiene caídas bruscas las cuales se pueden evitar con la elección correcta de valores de las matrices de ponderación (Q, R) y trabajando conjuntamente con la restricción de tasa de cambio de la acción de control ( $\Delta u$  o  $du$ ), se obtiene que la aproximación al punto de ajuste sea más suave, esto se debe a la característica del MPC quien permite controlar de manera individual las pendientes de subida y bajada de “CV”.

En la Figura 4.4 se puede observar un controlador MPC con  $H_p=200$ ,  $H_c=1$ ,  $Q=100$ ,  $R=25$  y restricción  $du_{\min}=-750$  y  $du_{\max}=0.10$ . Estos parámetros permitieron obtener un controlador MPC óptimo debido a que el control del proceso presión es el deseado.



**Figura 4.4:** Resultado real con  $H_p=200$ .

Elaborado por: Curay, D. y Vaca, A.

A continuación se observa la respuesta transitoria del controlador para un  $SP=30$  PSI y con diferentes horizontes de predicción proporcionados ver la Tabla 4.1.

**Tabla 4.1**

Respuesta transitoria del MPC.

Indicadores	Horizonte de Control						
	4	20	50	100	150	200	250
Tiempo de retardo (seg)	15.8	12.4	9.9	10.1	10.1	12.3	Procesamiento es muy grande.
Tiempo de subida (seg)	35.2	11.3	8.1	8.4	9.6	13.4	
Sobreimpulso (%)	1.56	2.13	2.66	2.33	0.66	0.2	
Tiempo de establecimiento (seg)	Inestable	Inestable	36.1	33.5	22.9	25.3	

Elaborado por: Curay, D. y Vaca, A.

Como se puede apreciar a medida que se aumenta el horizonte de predicción mejora notablemente el control, pero un aumento grande en este parámetro presenta problemas de procesamiento, debido a que el cálculo de optimización de la señal de control es más complejo y se necesita de un buen procesador para realizar los cálculos. Por esta razón el límite del horizonte de predicción es 200 y permite obtener un mejor control del sistema.

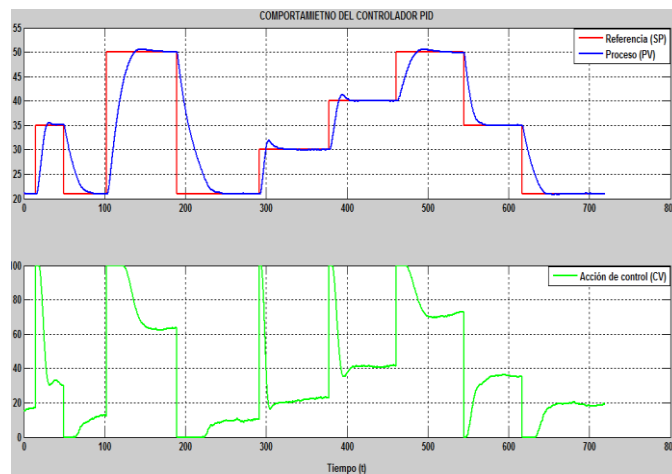
#### 4.2 Análisis del controlador PID en la planta de presión

La sintonización del PID en tiempo real para el proceso de presión es el primer paso para el análisis, el ajuste es realizado de manera empírica para encontrar los parámetros ( $K_c$ ,  $T_i$ ,  $T_d$ ) adecuados y permitirá obtener una buena aproximación a la solución del ajuste o sintonía del controlador.

Para ello se proporciona una ganancia  $K_c=10.5$ , con el objetivo de obtener un oscilación constante en la salida, es decir, que el PV (variable controlada) se mantenga cerca de la referencia sin superar el 30% de sobreoscilación por seguridad. Pero un controlador PID con ganancia proporcional presenta un error permanente el cual es solucionado al incluir la acción integral  $T_i= 0.14$  con el propósito de disminuir y eliminar el error en estado estacionario.

En este caso y luego de un análisis de las curvas de respuesta y su tiempo de estabilización se obtuvo que los dos parámetros proporcionados son suficientes para obtener el propósito deseado (mantener estable la variable del proceso en el punto de referencia designado).

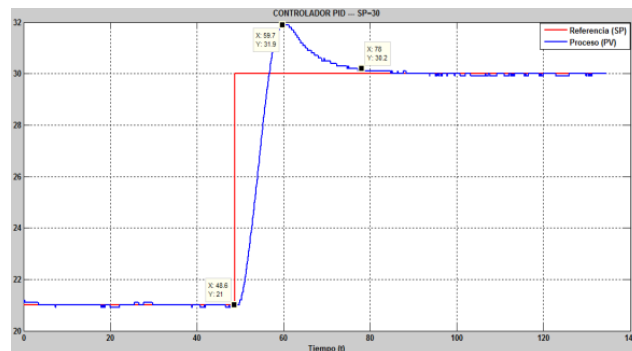
Para demostrar que los parámetros del control PID con los adecuados para el proceso, se realizó diferentes cambios de valores en el setpoint con el fin de demostrar que el control es estable, en la Figura 4.5 se indica el comportamiento del controlador PID.



**Figura 4.5:** Resultado real del controlador PID.

Elaborado por: Curay, D. y Vaca, A.

El controlador PID permite estabilizar el proceso a diferentes puntos de referencia pero la acción de control presenta un cambio brusco para llegar al "SP".



**Figura 4.6:** Respuesta del sistema del control PID.

Elaborado por: Curay, D. y Vaca, A.

En la Figura 4.6 se muestra la respuesta del sistema para un SP=30 PSI, un Sobreimpulso de 6.33% y un tiempo de establecimiento de 29.4 seg.

### 4.3 Comparación entre los controladores MPC vs. PID

En este apartado se presenta el análisis comparativo del funcionamiento de la planta de presión a la cual se ha puesto en marcha usando un controlador Predictivo Basado en Modelo (MPC) y un controlador Proporcional Integral Derivativo (PID).

El análisis es realizando con los datos almacenados que se tomaron durante la operación del proceso en tiempo real, donde la variable controlada por el MPC (trazos azul), la señal de control (trazos amarillos), por el controlador PID (trazos verdes), la señal de control (trazos celeste) y la referencia (trazos rojo), los parámetros de control MPC y PID son los siguientes: Ver Tabla 4.2.

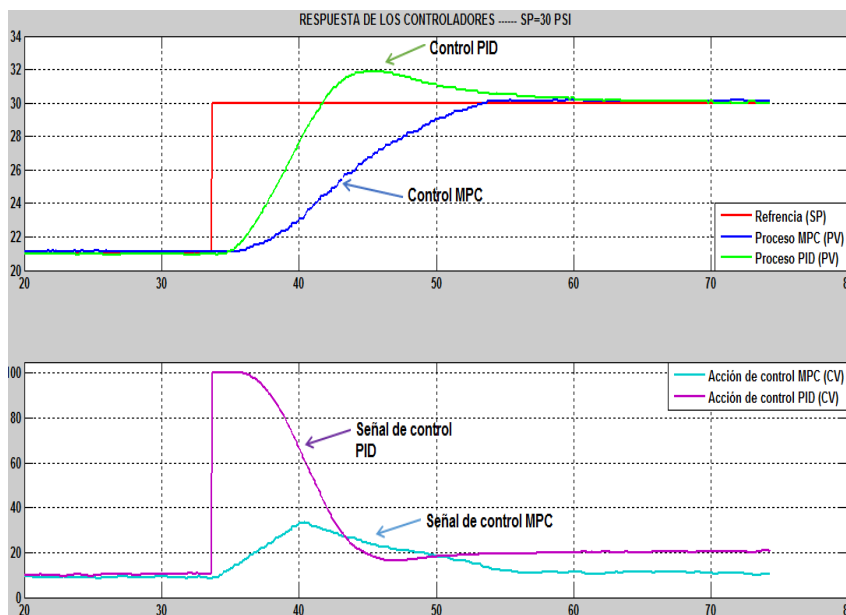
**Tabla 4.2**

Parámetros de sintonización de los controladores.

MPC		
<b>Parámetros de control</b>	Horizonte de predicción (Hp)	200
	Horizonte de control (Hc)	1
<b>Matrices de ponderación</b>	Q	100
	R	25
PID		
<b>Ganancias</b>	Kc	10.50
	Ti	0.14
	Td	0.0000

Elaborado por: Curay, D. y Vaca, A.

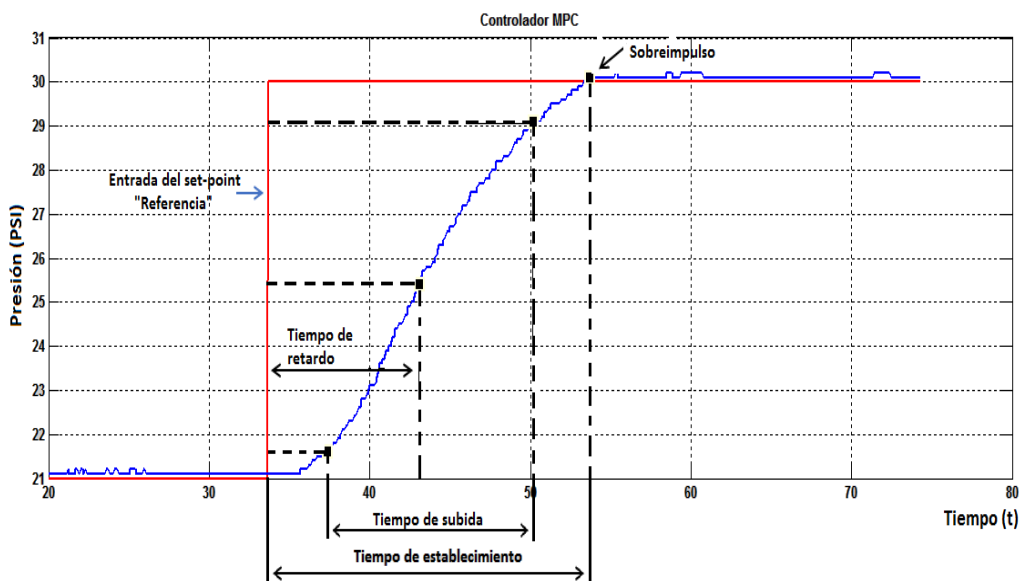
En la Figura 4.7 se indica la respuesta transitoria de los controladores a una referencia igual a 30 PSI.



**Figura 4.7:** Respuesta de los sistemas de control MPC y PID.

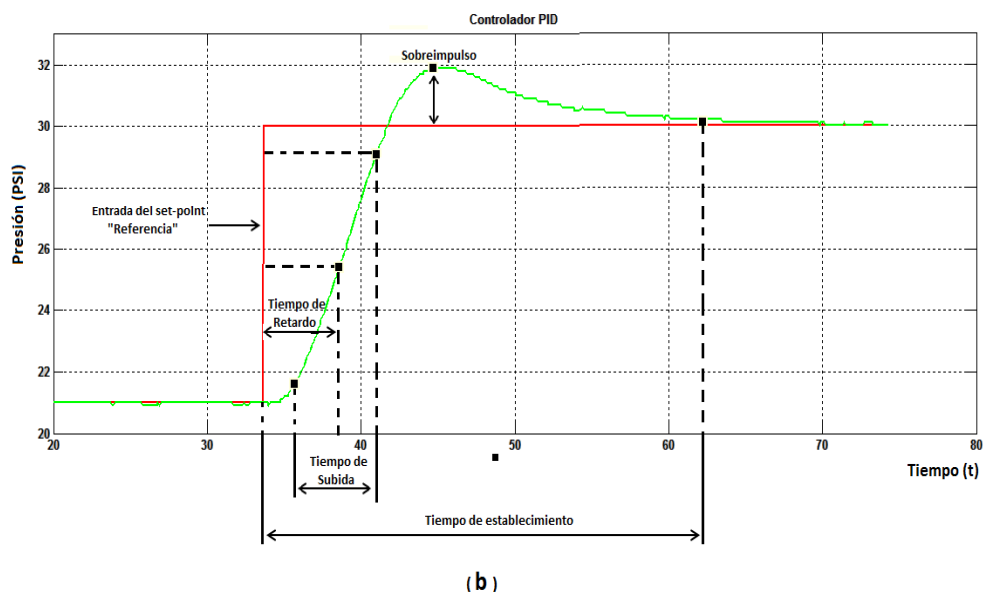
Elaborado por: Curay, D. y Vaca, A.

En la Figura 4.8 se observa los indicadores de la respuesta transitoria tomados en cuenta para facilitar la comparación de los controladores MPC (Figura 4.8 a) y PID (Figura 4.9 b).



(a)





**Figura 4.8:** Indicadores de la respuesta transitoria. a) MPC, b) PID.

Elaborado por: Curay, D. y Vaca, A.

**Tabla 4.3.**

Comparación de la respuesta transitoria del MPC y PID.

		TIPO DE CONTROL											
		PID	MPC	PID	MPC	PID	MPC	PID	MPC	PID	MPC	PID	MPC
Referencia "SP"	20_30	30_40		40_50		20_50		50_35		35_20			
	INDICADORES												
Tiempo de retardo (seg)	5	9,5	7	11,6	11,1	13,6	11,8	18,9	6,5	9,3	11,1	12,4	
Tiempo de subida (seg)	5,3	13	7,7	12,3	16,3	14,7	22,3	29,9	13	10,5	16,5	16,7	
Sobreimpulso (%)	6,33	0,33	3,00	0	1,2	0	1,2	0	0	0	0	0	
Tiempo de establecimiento (seg)	28,6	20,1	28,2	22,3	50,4	29,6	61,5	50,4	28,4	37,1	30,8	29,6	

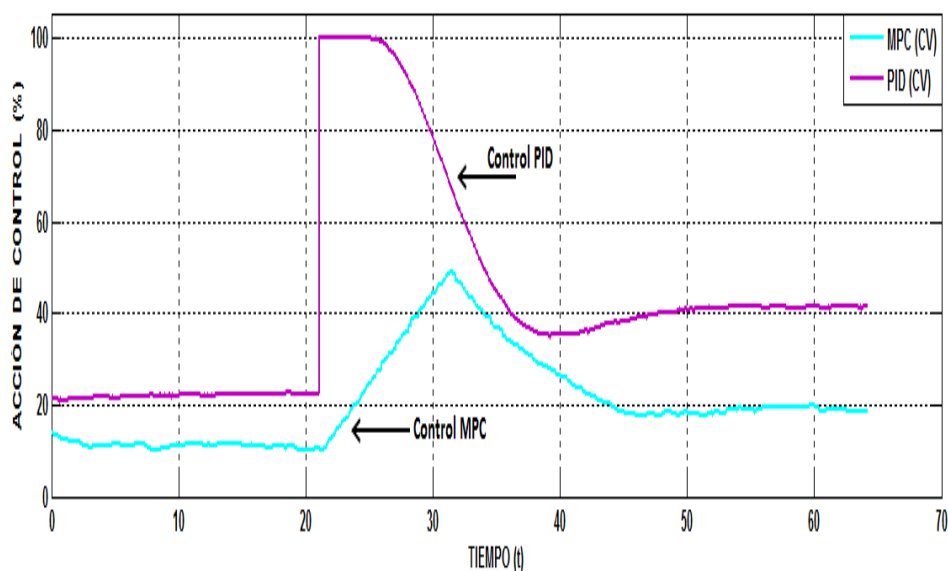
Elaborado por: Curay, D. y Vaca, A.

En la tabla 4.3 se indica los valores de la respuesta transitoria de ambos controladores sometidos a diferentes puntos de consigna con el objetivo de

visualizar cuál de ellos responde de manera eficiente al respectivo cambio, además los gráficos se puede observar en el Anexo D.

De esta tabla se puede evidenciar un mejor comportamiento y superioridad dinámica del controlador MPC sobre el controlador PID en términos del tiempo de establecimiento y el Sobreimpulso.

La rapidez de variación de la señal de control ( $du(t)$ ) como se puede observar en la Figura 4.9, indica la agresividad en la respuesta de los controladores, donde el PID es más agresivo con respecto al MPC en función del acercamiento a la señal de referencia.



**Figura 4.9:** Acción de control MPC y PID.

Elaborado por: Curay, D. y Vaca, A.

En la **Tabla 4.4** se indica la señal de control “CV” promedio aplicada al actuador es de 22.54% para el controlador PID y 13.01% para el controlador MPC, representando para el actuador un ahorro de trabajo del 9.53% en todo el tiempo de la operación realizada, con este ahorro de trabajo en el actuador se puede decir que a futuro se extenderá la vida útil de la válvula de control el cual es el objetivo del criterio de optimización en empleo de la función objetivo planteado por el controlador MPC de Labview.

**Tabla 4.4.**

Porcentaje de trabajo realizado por el actuador.

	<b>MPC</b>	<b>PID</b>
<b>Referencia "SP" (PSI)</b>	<b>Acción de control (CV) %</b>	<b>Acción de control (CV) %</b>
<b>20-30</b>	13	22.5
<b>30-40</b>	20.5	43.2
<b>40-50</b>	35.3	71
<b>20-50</b>	67.3	71
<b>50-35</b>	22.2	40.4
<b>35-20</b>	8.5	18.6
<b>Promedio</b>	27.8	44.45

Elaborado por: Curay, D. y Vaca, A.

## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1 CONCLUSIONES

Las conclusiones obtenidas son:

- ❖ El MPC requiere de un gran procesamiento de datos por el cálculo de optimización de recursos, por ello se utiliza nuevas tecnologías de controladores industriales de alto desempeño (PACs) que han incorporado dentro de su arquitectura los dispositivos FPGA lo que facilitan el diseño e implementación de controladores MPC.
- ❖ El sbRIO se programó con la herramienta de desarrollo Labview el cual interactúa directamente con el dispositivo y se encarga de la configuración de la FPGA, permitiendo implementar diferentes lazos de control como es el MPC, PID entre otros, además permite la adquisición de datos y escalar diferentes tipos de variables.
- ❖ Para la implementación del controlador predictivo se requiere del modelo matemático del proceso, el cual debe estar representada en ecuación de espacio de estados.
- ❖ El modelo del proceso debe contener la mayor información posible sobre la dinámica del sistema porque de esto depende el desempeño óptimo y precisión del controlador predictivo.
- ❖ El método utilizado para la modelación es la “Identificación de sistemas” proporcionado por el Software MATLAB que permite crear modelos dinámicos en base a los datos obtenidos de su entrada-salida.
- ❖ El modelo del sistema debe ser validado y sometido a diferentes pruebas experimentales con el fin de asegurar que los resultados obtenidos se aproxime a la realidad.
- ❖ El método de sintonización utilizado para los dos controladores, fue el método empírico o tanteo ya que este presenta mejores resultados en

el control, siendo el controlador MPC previamente realizado en simulación.

- ❖ El MPC implementado permite manejar restricciones reales del sistema los cuales son utilizados para el cálculo de la ley de control y así optimizar sus recursos, que no ocurre con otros controladores.
- ❖ Como todo controlador avanzando el MPC tiene la función de optimizar los recursos, en este caso permite optimizar el rendimiento del actuador el cual se ve reflejado en el porcentaje de acción de control utilizado para abrir y cerrar la válvula de control, permitiendo así extender la vida útil del actuador.
- ❖ Se puede apreciar, de acuerdo a los resultados obtenidos, que el control MPC tiene mejor comportamiento dinámico, ya que presenta un tiempo establecimiento y sobreimpulso menor que el control PID debido a que la acción de control se anticipa a una variación del setpoint y toma la acción respectiva antes de cualquier variación
- ❖ El control MPC en el proceso de presión, mantiene un mayor tiempo dentro de la ventana de operación con lo cual se tiene mayor eficiencia y eficacia en el control.
- ❖ En el análisis del control PID la acción de control es más fuerte provocando que el rendimiento del actuador llegue al 100% de trabajo en cualquier cambio producido a su entrada dando como resultado la disminución de la vida útil del equipo.
- ❖ En el análisis del control MPC la acción de control es suave, donde únicamente proporciona lo necesario al actuador para que el proceso llegue al valor deseado.
- ❖ Frente a perturbaciones ambos controladores responde de manera eficiente a la introducción de este tipo de señal.
- ❖ Los horizontes del controlador MPC no deben ser muy grandes debido a que este perjudica la evolución del control, y además provoca que el controlador tenga que realizar un procesamiento de datos alto y esto afecta directamente el rendimiento de la tarjeta SBRIO.

- ❖ Antes de poner en ejecución al controlador, las restricciones mínimas del MPC deben estar en valores iniciales igual a cero, con esto se evita la confusión del controlador al creer que las restricciones se infringieron.
- ❖ El horizonte de predicción debe ser pequeño al instante en que el controlador entra en funcionamiento, y luego debe ser reemplazado con el valor ideal para el control.
- ❖ El dispositivo sbRIO utilizados funcionan de forma independiente de la PC y en tiempo real es necesario crear un sistema de monitoreo para poder visualizar el comportamiento de las variables del proceso y del control.

## 5.2 RECOMENDACIONES

- ❖ Para empezar a utilizar el sbRIO es necesario ejecutar el programa MAX (NI Measurement & Automación Explorer) el cual contiene los drivers para el dispositivo, ya que sin los drivers no funciona la tarjeta.
- ❖ Se debe establecer la comunicación entre la PC y la tarjeta sbRIO por lo que es necesario ingresar una IP la cual permitirá tener una comunicación Ethernet.
- ❖ Si el dispositivo no se conecta o no establece comunicación con la PC es necesario deshabilitar el firewall de la computadora.
- ❖ Únicamente se debe instalar los drivers necesarios para la aplicación propuesta con el objetivo de evitar que el espacio de memoria del sbRIO se sature.
- ❖ Se recomienda realizar las conexiones adecuadas para adquirir y enviar las señales eléctricas respectivas al proceso y viceversa.
- ❖ Es necesario que la modelación en porcentaje de similitud sea lo más cercano a la realidad debido a que de esto depende el óptimo funcionamiento del controlador MPC.
- ❖ Se recomienda activar el parámetro de acción integral del MPC debido a que dicho parámetro permite mejorar la diferencia existente entre la planta real y el modelo de la planta.
- ❖ Debido a que la implementación de los controladores avanzados son escasos en la Universidad, se recomienda la investigación de este tipo de controlador (MPC) y entre otros para poder implementarlos en las plantas de procesos del Laboratorio de Redes Industriales y Control de Procesos con lo cual se logra conocimientos necesarios para el campo industrial.

## REFERENCIAS BIBLIOGRÁFICAS

### BIBLIOGRAFÍA

- [10] R. Ferreiro García, "El control predictivo con FIELDBUS", Tecnología de Control de Procesos con FOUNDATION Fieldbus, Ed. RA-MA Editorial: Madrid, 2007, pp. 191-230.
- [11] J. Dorsey, "Control robusto", *Sistemas de control continuos y discretos*, Ed. McGraw-hill: México, 2005, pp. 327-354.
- [12] C. Pérez López, "Control predictivo y robusto", *Matlab y su aplicaciones en la Ciencia y la Ingeniería*, I. Capella, Ed. PEARSON EDUCACIÓN, S.A: España, 2007, pp. 547-555.



## LINKGRAFÍA

- [1] F. E. Verdezoto Nuñez, “Modelo de control Predictivo Multivariable en ambiente de TAV para mejora u optimizar el sistema de bombeo en la estación #1 Lago Agrio,” Dpto. Electrónica y Automatización Control Industrial, Escuela Politécnica del Ejército, [en línea], Quito 2007. Disponibles en:  
<http://repositorio.espe.edu.ec/xmlui/handle/21000/672>
- [2] M. E. López Guillén, “Identificación De Sistemas, Aplicación al Modelado de un Motor de Continua,” Dpto. Electrónica, [en línea], 2007. Disponible en:  
<http://www.depeca.uah.es/depeca/repositorio/asignaturas/32328/Tema6.pdf>.
- [3] C. Kunusch, “Identificación De Sistemas Dinámicos,” Dpto. Electrotecnia, Universidad De La Plata, [en línea], España 2003. Disponibles en: <http://www.ing.unlp.edu.ar/cys/pdf/identificacion.pdf>.
- [4] A. Martínez Bueno, “Identificación Experimental De Sistemas,” Universidad De Alicante, [en línea], 2011. Disponible en:  
<http://rua.ua.es/dspace/bitstream/10045/18965/1/Identificacion%20experimental%20de%20sistemas.pdf>.
- [5] M. A. Carpio Alemán, “Evaluación del Desempeño de los Controladores Digitales PID y Predictivo tolerante a Fallas, Aplicadas al Control de Nivel de Líquido de un Tanque,” Universidad Politécnica Salesiana, [en línea], Cuenca 2013. Disponible en:  
<http://dspace.ups.edu.ec/bitstream/123456789/5280/6/UPS-CT002761.pdf>
- [6] A. R. Parra Alvarado, “Implementación de un Control Adaptivo de Ganancia Programada, para efectuar el Control de la Frecuencia de

salida de un Generador, ante variaciones en la Referencia,” Dpto. Electricidad y Computación, Escuela Superior Politécnica del Litoral, [en línea], Guayaquil 2011. Disponible en:

<http://www.dspace.espol.edu.ec/bitstream/123456789/19479/1/1.-%20Caratula.pdf>

- [7] S. Garrido, “Identificación, Estimación y Control de Sistemas NO-Lineales mediante RGO,” Dpto. Ingeniería de Sistemas y Automática, Universidad CARLOS III DE MADRID, [en línea], Leganés 1999. Disponible en:  
<http://roboticslab.uc3m.es/publications/SGarridoPhD.pdf>
- [8] E. Franco Mejía, “Grupo de Investigación en Control Industrial,” [en línea]. Disponible en:  
<http://www.univalle.edu.co/~automatica/Cursos/MoldelaPreg/Material/Identificacion.ppt>.
- [9] J. M. Gonzales de Duranda, “Introducción a MATLAB,” Dpto. Ingeniería de Sistemas y Automática, [en línea], 2004. Disponible en:  
<http://www.vc.ehu.es/depsi/jg/imatlab.pdf>
- [13] System identification Toolbox, “Create linear and nonlinear dynamic system models from measured input-output data,” [en línea]. Disponible en:  
<http://www.mathworks.com/products/datasheets/pdf/system-identification-toolbox.pdf>.
- [14] Jim Mclellan, “Using Matlab and the System identification Toolbox to Estimate series Models,” [en línea], Febrero 2004. Disponible en:  
[http://chemeng.queensu.ca/courses/CHEE436/matlab/documents/using\\_Matlab.pdf](http://chemeng.queensu.ca/courses/CHEE436/matlab/documents/using_Matlab.pdf).

- [15] Lennart Ljung, "System identification Toolbox for use with MATLAB" [en línea], 1997. Disponible en:  
<http://site.ntvc.edu.cn/jx/jpkc/zkyl/syix/pdf/12.pdf>.
- [16] Lennart Ljung, "System Identification Toolbox, User's Guide," [en línea], 2014. Disponible en:  
[http://www.mathworks.com/help/pdf\\_doc/ident/ident.pdf](http://www.mathworks.com/help/pdf_doc/ident/ident.pdf).
- [17] M. Fernández, "El ABC de la Automatización de Control Avanzado," [en línea]. Disponible en:  
[http://www.aie.cl/files/file/comites/ca/abc/Control\\_Avanzado.pdf](http://www.aie.cl/files/file/comites/ca/abc/Control_Avanzado.pdf).
- [18] J. W. Mark y T. T. Ming, "Advanced Process Control," Dpto. Chemical and Process Engineering, University of Newcastle, 1994.
- [19] C. Bordóns Alba, "Control Predictivo: Metodología, Tecnología y nuevas Perspectivas," Dpto. Sistemas y Automática, Universidad de Sevilla, [en línea], 2000. Disponible en: <http://control-ps2316-sept2009.wikispaces.com/file/view/CONTROL+PREDICTIVO.pdf>
- [20] S. P. Ati Villamarin, "Análisis, Diseño e Implementación de Controladores predictivos," Tesis Ingeniería Electrónica, Dpto. Eléctrica y Electrónica, Escuela Politécnica del Ejército, [en línea] Sangolqui-Ecuador 2011. Disponible en:  
<http://repositorio.espe.edu.ec/bitstream/21000/4766/5/T-ESPE-032867-.pdf>.
- [21] M. J. Ortiz Guevara, "Diseño y Construcción de un convertor DC/DC Reductor Elevador con control Predictivo controlado por una tarjeta FPGA," Tesis Ingeniería Electrónica y Control, Dpto. Eléctrica y Electrónica, Escuela Politécnica Nacional, [en línea], Quito 2011. Disponible en:  
<http://bibdigital.epn.edu.ec/bitstream/15000/4272/1/CD-3907.pdf>.

- [22] National Instrument Corporate Headquarters, “LabVIEW™ Control Design User Manual,” [en línea], June 2009. Disponible en: <http://www.ni.com/pdf/manuals/371057g.pdf>.
- [23] Sistemas Industriales BECKHOFF, “Los PAC’s”, [en línea]. Medellín-Colombia: Logic Electronic’s, 2004. Disponible en: <http://www.logicelectronic.com/BECKHOFF/Que%20es%20un%20PAC.htm>.
- [24] National Instruments, “NI Single-Board RIO 9636”, [en línea]. Disponible en: <http://sine.ni.com/nips/cds/view/p/lang/es/nid/210421>
- [25] National Instruments, “FPGA”, [en línea]. Disponible en: <http://www.ni.com/white-paper/6983/es/>