

**TERCER INFORME DE AUTOMATIZACIÓN DE PROCESOS:**  
**DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR CON PREDICTOR**  
**SMITH.**

LEONARDO FABIO FERNANDEZ DIAZ  
CARLOS CRUZ PÉREZ

UNIVERSIDAD DE LA SALLE  
BOGOTÁ D.C  
2020

## CONTENIDO

<b>INTRODUCCIÓN.....</b>	<b>4</b>
<b>OBJETIVOS.....</b>	<b>5</b>
<b>MARCO TEÓRICO .....</b>	<b>6</b>
<b>METODOLOGÍA.....</b>	<b>9</b>
<b>RESULTADOS Y DISCUSIÓN.....</b>	<b>20</b>
<b>CONCLUSIONES. ....</b>	<b>40</b>

## RESUMEN

El presente informe nace como una evidencia del trabajo realizado en la práctica de laboratorio consistió en el diseño e implementación de un controlador por lugar geométrico de las raíces, complementado por un predictor Smith, para hacerlo más robusto a perturbaciones y lograr mantener la estabilidad del sistema.

Se optó por el desarrollo de un controlador PI, ya que se quería lograr el seguimiento de referencias y además establecer un error estacionario cercano a cero, por lo que al ser un sistema que controla variables como flujo y presión, muchas veces no se requiere que sea tan rápido, por lo que con las componentes proporcional e integral se cumplían los requerimientos.

Primeramente, se realizó el desarrollo en Matlab, donde se llevó a cabo la identificación del modelo, la simulación del controlador con su predictor Smith en el dominio continuo, discreto y ecuación en diferencias para luego pasar a su implementación en PLC, con ayuda del entorno de desarrollo TIA Portal se implementó el controlador, encontrando unos controladores que seguían referencias y sin error estado estacionario, teniendo en cuenta mayores aspectos de la planta. Sin embargo, se logró un paso aceptable de la simulación a la vida real, dando por cumplidos los objetivos de la práctica.

*Palabras Claves:* Identificación, Diseño, Controlador, Smith, PLC.

## INTRODUCCIÓN.

El predictor de Smith es una estrategia de control, que permite complementar los controles tradicionales para hacerlo más robusto ante perturbaciones externas, se compone por una configuración distinta dentro de su diagrama de bloques, es de gran importancia en sistemas con tiempo de retardo, donde poseen un tiempo muerto donde su comportamiento no puede ser descrito por una ecuación de un sistema lineal, por lo que el predictor intenta acoplar el comportamiento del sistema en el tiempo conocido, dentro de su tiempo muerto, para así lograr una mejor caracterización de la dinámica del sistema, es por esto que trabaja tanto con el retardo del sistema como sin él, es de ahí que se resalta la importancia de las cualidades de la implementación de esta configuración de control, ya que existen en la industria muchos sistemas que poseen este tipo de retardos.

Para el módulo de procesos de la Universidad de la Salle, esto no es la excepción, ya que el rango de operación del sistema no arranca de una entrada en cero, sino que tiene límite superior en inferior, por lo que existe un lapso de tiempo donde el sistema recibe una entrada pero no genera una salida, generando así un tiempo muerto que luego para el diseño de controladores puede llegar a hacer más complejo su desarrollo, por lo que una herramienta como el predictor de Smith son de gran utilidad para este tipo de sistemas.

El PI por lugar de las raíces es diseñado teniendo en cuenta el trasciente del sistema, por lo que se hace necesario conocer su respuesta en el tiempo, para luego establecer unas condiciones de error en estado estacionario e ir acoplando la dinámica para lograr ese comportamiento.

# **OBJETIVOS.**

## **Objetivo General:**

Diseñar e implementar un controlador predictivo Smith para la unidad de entrenamiento en automatización.

## **Objetivos Específicos:**

- Identificar el sistema de flujo y presión con el que cuenta la unidad de mantenimiento.
- Diseñar y discretizar un controlador predictivo Smith para el sistema de tiempo muerto.
- Implementar por ecuaciones en diferencias el controlador diseñado en un PLC.
- Realizar pruebas de funcionamiento y correcciones sobre la unidad de entrenamiento.

# MARCO TEÓRICO

## IDENTIFICACIÓN:

La identificación de sistemas es la teoría de construir modelos matemáticos de sistemas dinámicos basándose en las entradas y salidas observadas. Sin embargo, en la industria es muy común encontrarse con sistemas de tiempo muerto, es decir, que el sensor de medida no está cercano al proceso por lo que se debe recurrir a ciertas técnicas de identificación como las propuestas por Victor M. Alfaro (2007). La más común de ellas es:

- Modelo de primer orden más tiempo muerto (POMTM):

Este modelo presenta la siguiente estructura:

$$G_{p1}(s) = \frac{k_p e^{-t_m s}}{\tau s + 1} \quad (1)$$

Donde:

$$\begin{aligned} k_p &= \frac{\Delta y}{\Delta u} \\ \tau &= 0,9102 (t_{75} - t_{25}) \\ t_m &= 1,2620 t_{25} - 0,2620 t_{75} \end{aligned} \quad (2)$$

Como se observa en la expresión (1), el modelo identificado resulta tener una ganancia  $K_p$ , un retardo expresado por el exponencial y un polo. Diseñar un controlador para esta clase de sistemas resulta ser complicado, es por ello que se crearon técnicas de sintonización y el controlador predictor Smith.

## CONTROLADOR:

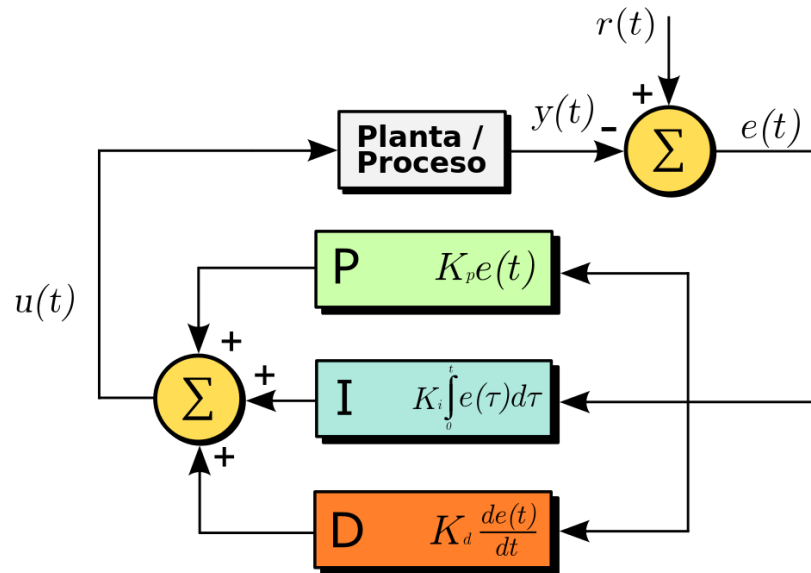
Es un programa informático que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz (posiblemente estandarizada) para utilizar el dispositivo.

- Controlador PID:

Es un mecanismo de control simultáneo por realimentación ampliamente usado en sistemas de control industrial. Este calcula la desviación o error entre un valor medido y un valor deseado. Consta de tres parámetros distintos: el proporcional, el integral, y el derivativo. El valor proporcional depende del error actual, el integral

depende de los errores pasados y el derivativo es una predicción de los errores futuros. La suma de estas tres acciones es usada para ajustar el proceso por medio de un elemento de control, como la posición de una válvula de control o la potencia suministrada a un calentador. (Norman S. Nise, 2015).

Figura 2. Estructura de un controlador PID.

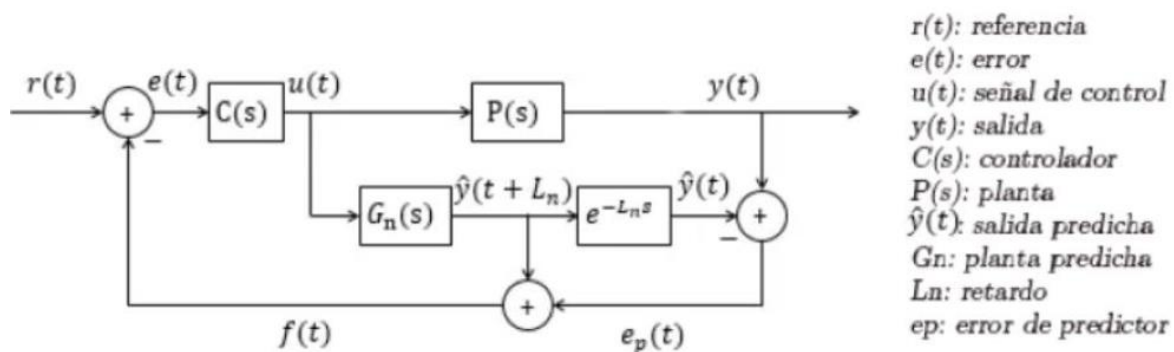


- Controlador Predictor Smith:

Uno de los principales problemas de los controladores clásicos, como es el PID, es su comportamiento frente a plantas con un retardo considerable. Este retardo se puede deber a una distancia física entre el proceso y el lugar de medición de la variable, una demora en los actuadores o cualquier otra causa.

El predictor Smith es un tipo de controlador predictivo diseñado para controlar sistemas con un retraso de tiempo de retroalimentación significativo. Su estructura se ve en la siguiente figura:

Figura 3. Estructura del predictor Smith.



## DISCRETIZACIÓN:

Es el proceso de transferir funciones continuas, modelos, variables y ecuaciones a contrapartes discretas. Este proceso generalmente se lleva a cabo como un primer paso para hacerlos adecuados para la evaluación numérica y la implementación en computadoras digitales.

## ECUACIONES EN DIFERENCIAS:

Una ecuación en diferencias es una expresión que relaciona distintas sucesiones, siendo una de ellas una sucesión desconocida. Son similares a las ecuaciones diferenciales, sustituyendo las funciones por sucesiones y para su resolución suele utilizarse el método de la transformada Z.

## CONTROLADOR LÓGICO PROGRAMABLE (PLC):

Es una computadora utilizada en la ingeniería automática o automatización industrial, para automatizar procesos electromecánicos, electroneumáticos, electrohidráulicos, tales como el control de la maquinaria de la fábrica en líneas de montaje u otros procesos de producción, así como atracciones mecánicas.

Figura 4. Aspecto de un PLC 1500 de siemens.





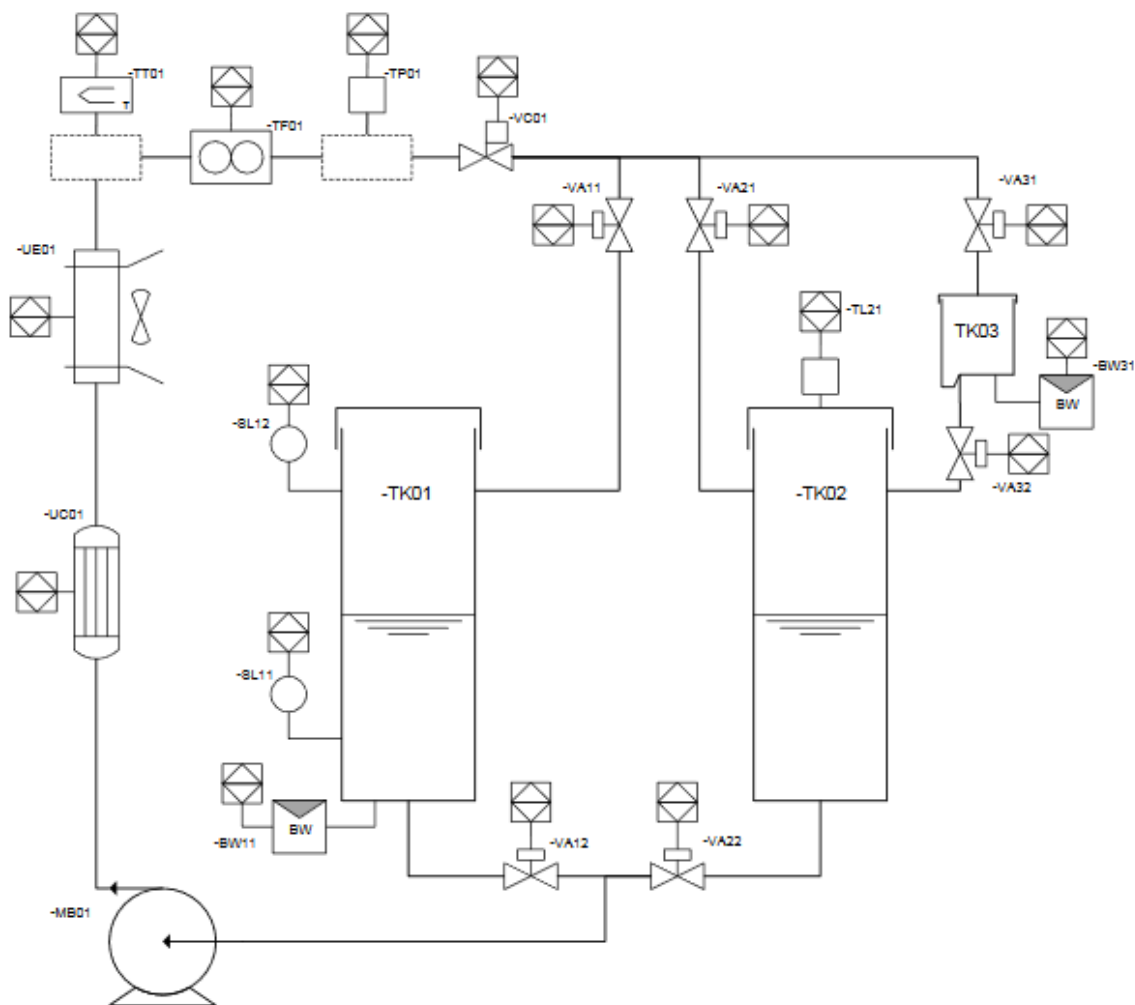
## METODOLOGÍA

Se desea implementar un controlador de Flujo y Presión en la unidad de entrenamiento en Automatización proporcionada por la Universidad de la Salle.

Dicha unidad cuenta con los siguientes elementos (Ver Figura 1):

- Una bomba controlada mediante un variador de frecuencia. (MB01)
- 2 tanques de almacenamiento. (TK01, TK02)
- 4 electroválvulas. (VA11, VA21, VA12, VA22)
- 1 válvula proporcional (VC01).
- Transmisor de nivel.
- Transmisor de presión (TP01)
- Transmisor de flujo (TF01).

Figura 5. Unidad de entrenamiento en Automatización.



Fuente: Universidad de La Salle, 2020.

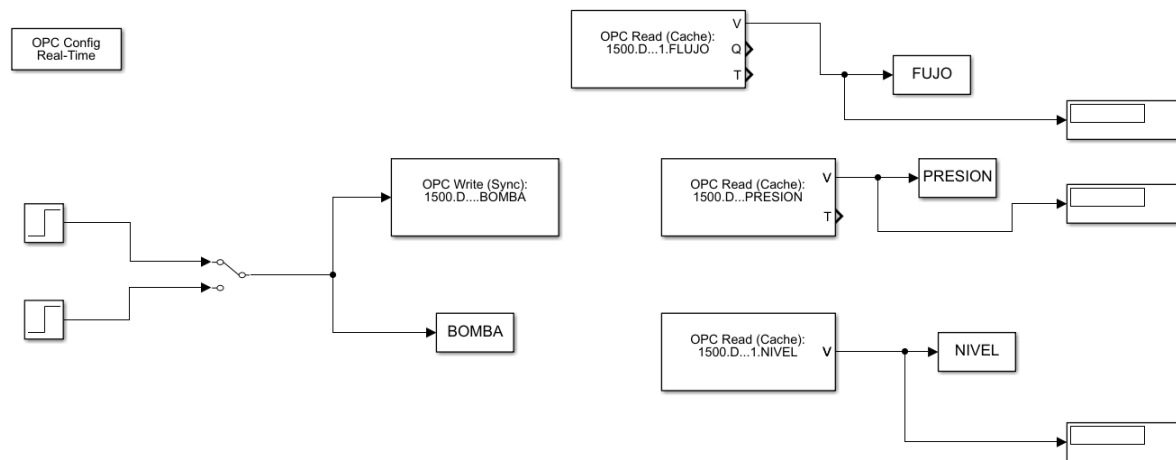
Como se describió anteriormente, la unidad de entrenamiento cuenta con un transmisor de presión y otro de flujo, los cuales miden dichas variables a medida que el fluido circula por el sistema. La apertura y cierre de las válvulas pueden modificar el comportamiento de la presión y el flujo. Sin embargo, para este caso se desea trabajar con todas las válvulas abiertas, y excitar el sistema mediante el variador que controla a la bomba.

Para lograr los objetivos propuestos, se realizaron las siguientes actividades:

### Actividad 1:

Poner en funcionamiento la unidad de entrenamiento mediante el uso del PLC 1500 de Siemens con el que cuenta, dejando todas las válvulas abiertas para que el fluido pueda circular fácilmente por el sistema. Posteriormente, realizar una conexión OPC entre TIA portal y Matlab mediante el uso de KepServer y tomar medidas de flujo y presión para el sistema.

Figura 6. Esquema en Simulink para recolectar datos de Flujo y presión de la unidad de entrenamiento.



### Actividad 2:

Realizar el respectivo tratamiento de los vectores de datos registrados mediante la conexión OPC entre Matlab y TIA (Figura 6) para el tiempo, flujo y presión. Es decir, enfocar el análisis sobre la ventana de tiempo en que se excitó el sistema hasta que se estabilizó.

Posteriormente, se deben identificar ambos sistemas y construir un modelo de primer orden más tiempo muerto [1], tanto para la presión como para el flujo.

### Actividad 3:

Diseñar un controlador predictivo Smith. Para ello se diseña un controlador PI para la planta sin considerar el tiempo muerto y luego, se construye la estructura Smith teniendo en cuenta: la planta más tiempo muerto y el controlador PI diseñado.

- Control PI para **Flujo** (Teorema del límite):

Se requiere que el sistema siga referencias con error en estado estacionario de cero. Es por ello que se recurre al diseño de un controlador PI teniendo las siguientes consideraciones en cuenta:

$$G_c = K_p + \frac{K_i}{s} = \frac{K_p * s + K_i}{s} = \frac{K_p \left( s + \frac{K_i}{K_p} \right)}{s} = \frac{K_p * (s + a)}{s} \quad (3)$$

Donde:

Kp: es la ganancia proporcional del controlador.

a: es el cero del controlador ubicado en  $-K_i/K_p$

además, que posee un polo en el origen.

- Para calcular Kp se puede utilizar el teorema del límite final de la siguiente forma:

$$K = \lim_{s \rightarrow 0} (K_p * G_p) = \lim_{s \rightarrow 0} \left( \frac{K_p * a_1}{s + b_1} \right) = \frac{K_p * a_1}{b_1}$$
$$essp = \frac{1}{1 + K} = \frac{b_1}{b_1 + a_1 * K_p}$$
$$K_p = \frac{b_1 * (1 - essp)}{essp * a_1} \quad (4)$$

Donde:

Gp es la función de transferencia del sistema.

Essp es el error de posicion deseado, en este caso cero.

K es la constante de posicion.

a1 y b1: Son constantes conocidas dadas por el modelo de identificación.

- Una vez calculada la ganancia Kp que permite seguir referencias en el controlador PI, se debe calcular la ganancia Ki. Para ello se usará la teoría del diseño por el lugar geométrico de las raíces, teniendo en cuenta que el

controlador PI ubica un polo en el origen y un cero muy cercano al origen según lo permita el sistema. Este cero viene dado por la proporción:

$$\frac{K_i}{K_p} = a$$

Es decir, que se puede calcular el valor de  $K_i$ , teniendo la constante  $K_p$  y el cero (**a**) del controlador o también, se puede calcular la ubicación del cero (**a**) teniendo las constantes del controlador PI.

La otra alternativa es utilizar el teorema del límite para velocidad, según la siguiente expresión:

$$G_{cl} = G_c * G_p = \left( K_p + \frac{K_i}{s} \right) * \left( \frac{a1}{s + b1} \right)$$

Reemplazando la función en lazo cerrado en el teorema del límite:

$$K_v = \lim_{s \rightarrow 0} s * (G_{cl}) = \lim_{s \rightarrow 0} s * \left( \frac{(K_p * s + K_i)(a1 * K_p)}{s(s + b1)} \right) = \frac{K_i * K_p * a1}{b1}$$

$$K_v = K_i * K_p * c1 \quad (\text{donde: } c1 = a1/b1)$$

Reemplazando  $K_v$  en la expresión del error para velocidad y despejando  $K_i$ :

$$essv = \frac{1}{K_v} = \frac{1}{K_i * K_p * c1}$$

$$K_i = \frac{1}{K_p * c1 * essv} \quad (5)$$

$G_{cl}$  es el sistema en lazo cerrado entre el controlador PI y la planta.

$K_p$  y  $K_i$  son las constantes del controlador.

$Essv$  es el error deseado en estado estacionario.

$C1$  es conocido ya que es la división entre  $a1$  sobre  $b1$ .

De esta forma, la expresión 4 y 5 describen la forma en que se puede calcular las constante  $K_p$  y  $K_i$  del controlador conociendo la función de transferencia de la planta y el error deseado en estado estacionario.

#### - Control PI para Presión (Asignación de polos):

La metodología anterior, se usará para diseñar el controlador PI del **Flujo**, sin embargo, para el de **Presión** se optó por diseñar un controlador PI por asignación de polos.

Para diseñar un controlador por este método, se parte de las siguientes consideraciones:

$$G_p = \frac{a1}{s + b1}$$

Donde:

Gp es la función de transferencia de la planta.

a1 es la ganancia de la planta en lazo abierto.

b1 es el polo de la planta en lazo abierto.

Se desea que la planta en lazo cerrado se comporte de acuerdo con unos criterios de diseño: Tiempo de establecimiento (Tss) y Overshoot (Os). Asignando estos criterios se puede calcular la constante de amortiguamiento y la frecuencia deseada para el sistema:

$$\xi = \frac{\ln\left(\frac{\%OS}{100}\right)^2}{\sqrt{\pi^2 + \ln\left(\frac{\%OS}{100}\right)^2}} \quad (6)$$

$$\omega_n = \frac{3}{\xi * T_{ss}} \quad (7)$$

Donde:

$\xi$  es la constante de amortiguamiento.

$\omega_n$  es la frecuencia del sistema.

%os es el porcentaje de overshoot.

Tss es el tiempo de establecimiento.

Estos criterios de diseño construyen la ecuación característica del controlador para que el sistema se comporte como se desea en lazo cerrado. Esta ecuación está dada así:

$$Gc = s^2 + (2 * \xi * \omega_n) * s + \omega_n^2 \quad (8)$$

La expresión 8, expresa la ubicación de los polos en lazo cerrado para el sistema controlado mediante una función cuadrática, de la siguiente forma:

$$ax^2 + bx + c = 0$$
$$p_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (9)$$

La ecuación característica deseada (ecuación 8) será comparada con la ecuación en lazo cerrado, es decir, controlador mas planta, de la siguiente forma:

- Planta:

$$Gp = \frac{a1}{s + b1} = \frac{A}{B}$$

- Controlador:

$$Gc = \frac{kp * Td * s^2 + Kp * s + \frac{Kp}{Ti}}{s} = \frac{d2 * s^2 + d1 * s + d0}{s} = \frac{D}{E}$$

Donde:

Td es la constante derivativa = 0

- Ecuación en lazo cerrado (H):

$$H(s) = \frac{DA}{EB + DA} = \frac{a1(d2 * s^2 + d1 * s + d0)}{s(s + b1) + a1(d2 * s^2 + d1 * s + d0)}$$

$$H(s) = \frac{a1(d2 * s^2 + d1 * s + d0)}{s^2(1 + a1 * d2) + s(b1 + a1 * d1) + a1 * d0}$$

La ecuación en lazo cerrado característica es:

$$s^2(1 + a1 * d2) + s(b1 + a1 * d1) + a1 * d0 \quad (10)$$

- Cálculo de las constantes Kp y Ti:

La expresión **10** se compara con la ecuación deseada **8** y se despejan las expresiones para cada constante del controlador (Kp, Ti, Td):

$$s^2(1 + a1 * d2) + s(b1 + a1 * d1) + a1 * d0 = s^2 + (2 * \xi * wn) * s + wn^2$$

Entonces:

$$1 = (1 + a1 * d2) = (1 + a1 * Kp * Td) \quad (11)$$

$$(2 * \xi * wn) = (b1 + a1 * d1) = (b1 + a1 * Kp) \quad (12)$$

$$wn^2 = a1 * d0 = a1 * \frac{Kp}{Ti} \quad (13)$$

Sabiendo que: Td=0, y usando las expresiones **12** y **13**, se encuentra que las expresiones para las constantes son:

$$Kp = \frac{(2 * \xi * wn) - b1}{a1}; \quad Ti = \frac{a1 * Kp}{wn^2}; \quad (14)$$

Al diseñar el controlador PI para cada sistema, se construirá el controlador predictor Smith teniendo en cuenta la planta más tiempo muerto del sistema:

$$G_{pt} = \frac{e^{(-tm)*s} a1}{s + b1}$$

**Actividad 4:** Discretizar el controlador Smith y expresarlo en ecuaciones en diferencias para su implementación en un controlador lógico programable que controle el flujo y presión dentro de la unidad de entrenamiento en automatización.

Para la discretización se debe seleccionar un tiempo de muestreo según los siguientes criterios:

$$\frac{Tr}{20} < Tm < \frac{Tr}{10} \quad (15)$$

$$\frac{Tss}{75} < Tm < \frac{Tss}{25} \quad (16)$$

Donde:

Tr es el tiempo de subida en lazo abierto.

Tss tiempo de establecimiento en lazo abierto.

Tm es el tiempo de muestro.

Se puede usar cualquiera de los dos criterios (expresión **15** o **16**) y discretizar la ecuación de la planta y el controlador usando **tustin**.

Posteriormente se expresa ambas ecuaciones discretizadas en ecuación en diferencias y se implementa en el PLC usando variables, de la siguiente forma:

#### **Implementación en PLC:**

##### **- Controlador de Flujo**

Para el controlador de flujo y presión para el módulo de procesos de la Universidad de la Salle se utilizó el entorno de programación TIA Portal de Siemens, ya que el módulo está controlado por un PLC 1500 de Siemens, por lo que, para la implementación de controladores externos, se debe crear un bloque de funciones (FB), para desarrollar ahí la ecuación en diferencias y utilizar las señales y variables globales del sistema.

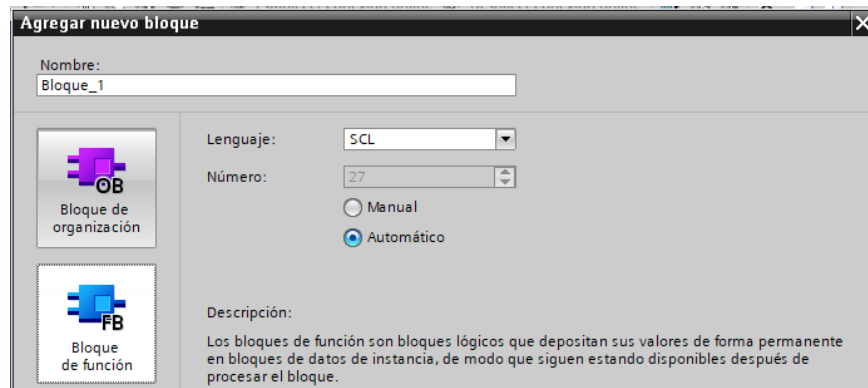


Figura 1: Creación de nuevo bloque en TIA Portal

Como lo muestra la figura 1, existen diversos bloques de programa los cuales están dados por la norma IEC-61133-3, por lo que este entorno de desarrollo está basado en la norma y listo para su ejecución, en este caso se creara un bloque de función, el cual utiliza unas entradas y salidas y el genera un procesamiento interno para la ejecución de la tarea.

	Nombre	Tipo de datos	Valor predet.
Input			
	ref	Real	0.0
	sensor	Real	0.0
Output			
	Motor	Real	0.0

Figura 2: Entradas y salidas del bloque de función para el controlador de flujo.

En la figura 2 se evidencia como se hace la asignación de las entradas y salidas del bloque de función, en este apartado seleccionamos el nombre de la variable para uso interno, no significa que este sea el nombre de la variable global, ya que esa asignación se realiza de manera diferente, además de selección el tipo de variable y un valor inicial que será el primer valor en el primer ciclo scan o de interrupción del bloque.

Las demás variables pueden ser llamadas como estáticas o temporales, ya que son usadas solo en el funcionamiento del bloque, por lo que no afectan variables globales de manera directa.

En la figura 1 se evidencia que el lenguaje de programación escogido es el SCL, conocido como texto estructurado, el cual consiste en una programación por código de las funciones del PLC, es ideal para la implementación de controladores y de



procesos complejos y matemáticos, por lo que nuestros bloques fueron programados en este lenguaje.

```
#e0 := (#ref - #sensor);

#u0 := #NGcz[1]*#e0+#NGcz[2]*#e1+#NGcz[3]*#e2+#NGcz[4]*#e3+#NGcz[5]*#e4+#NGcz[6];

IF #u0 > 27000.0 THEN
    #Motor := 27000.0;
ELSIF #u0 < 6500.0 THEN
    #Motor := 6500.0;
ELSE
```

Primeramente, se realizó el cálculo del error, que corresponde a la resta entre la referencia y la salida de la planta, que en este caso es la medida del sensor, luego se implementa la ecuación en diferencias del sistema, donde los coeficientes fueron declarados en un vector para su cambio rápido. Luego se implementa un saturador para la seguridad del sistema, donde se asegura que la salida del controlador no sobrepase los valores máximos y mínimos posibles por los equipos del sistema, para que así cuando se realice un cálculo que salga de estos valores, no entre en error el PLC.

```
#u5:=#u4; #u4:=#u3; #u3:=#u2; #u2:=#u1; #u1:=#u0;

#e5:=#e4; #e4:=#e3; #e3:=#e2; #e2:=#e1;
#e1 := #e0;
```

El último paso es realizar la actualización de las variables para generar ese corrimiento dentro de los valores y así poder tener valores pasados del sistema para el desarrollo del controlador, por lo que para este caso se actualizan los valores de salida y los del error.

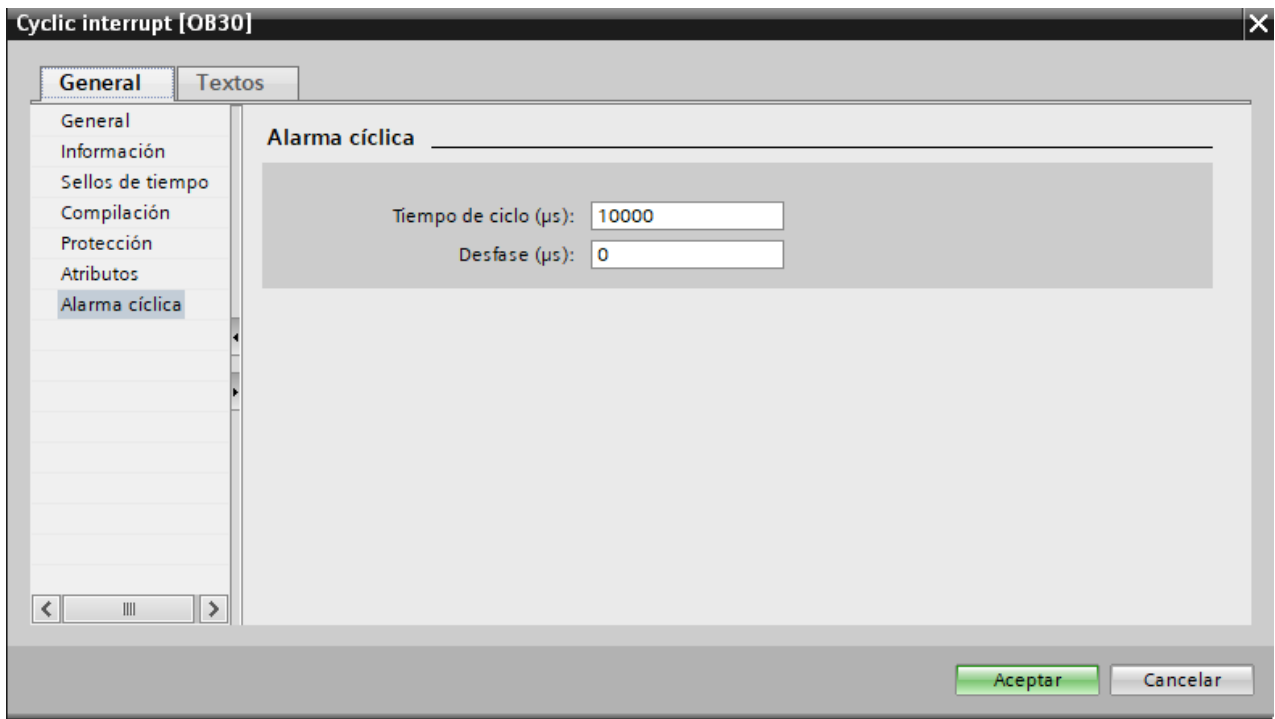


Figura 3: Configuración ciclo de interrupción.

Los bloques OB30 en Siemens corresponden a bloques de interrupción, los cuales se ejecutan y envían datos en un tiempo cíclico, en este caso se debe poner el tiempo de diseño del controlador, es decir debe ser igual al tiempo de muestreo para la discretización del controlador en el proceso de diseño, para generar una salida equivalente en una implementación real.

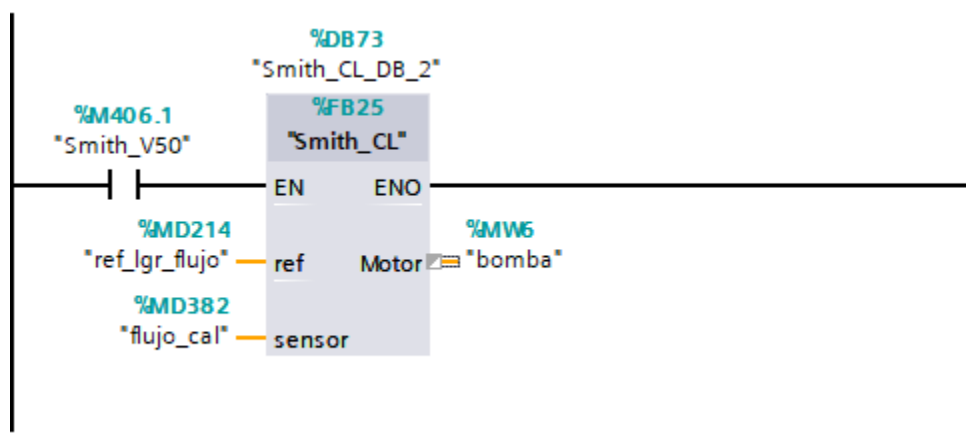


Figura 4: Llamado del bloque en el ciclo de interrupción.

Como se muestra en la figura 4, el llamado del bloque de función es mucho más gráfico, es donde se asignan las variables globales a las preestablecidas internamente del bloque, en este caso se utiliza la referencia y el sensor, para mandar la señal de control a la bomba que controla el motor y permite el flujo dentro

del módulo, además al inicio un contacto normalmente abierto, para poder activar y desactivar el controlador y así poder evidenciar su funcionamiento normal.

#### - **Controlador de Presión.**

Para la presión el proceso es análogo, solo que ahora se cambia la ecuación en diferencias y se crea en otro bloque FB, es ahí donde se crea un proceso estándar donde solo cambia los coeficientes del controlador y el orden de este, esto permite generar controladores genéricos y que pueden ser utilizados en varias plantas.

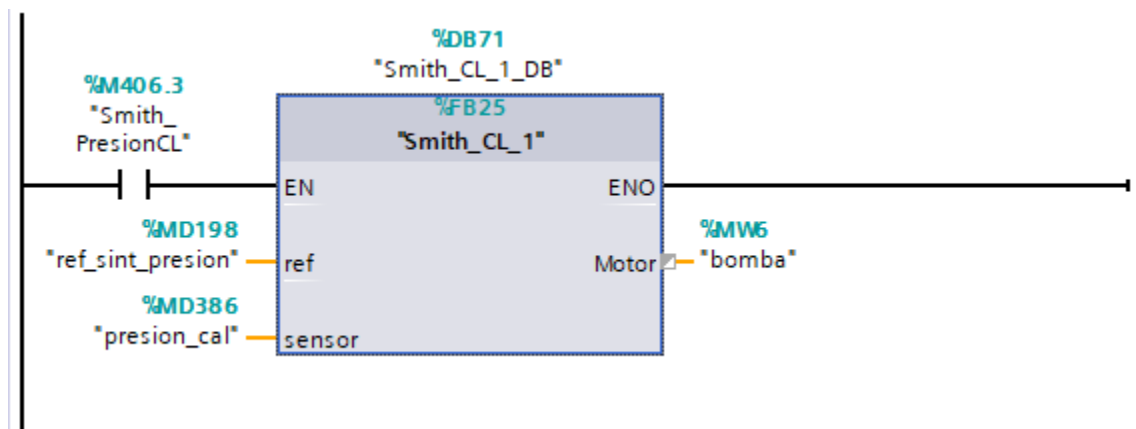


Figura 5: Llamado del bloque en el ciclo de interrupción para presión.

En la figura 5 es mostrado el llamado del bloque con el controlador de control, en este caso las variables son de presión, por lo que el sistema ya es cambiado para genera el control de esta variable, pero los demás procesos son iguales que para la implementación del controlador de flujo.

**Actividad 5:** Realizar pruebas y correcciones del controlador sobre el proceso.

## RESULTADOS Y DISCUSIÓN.

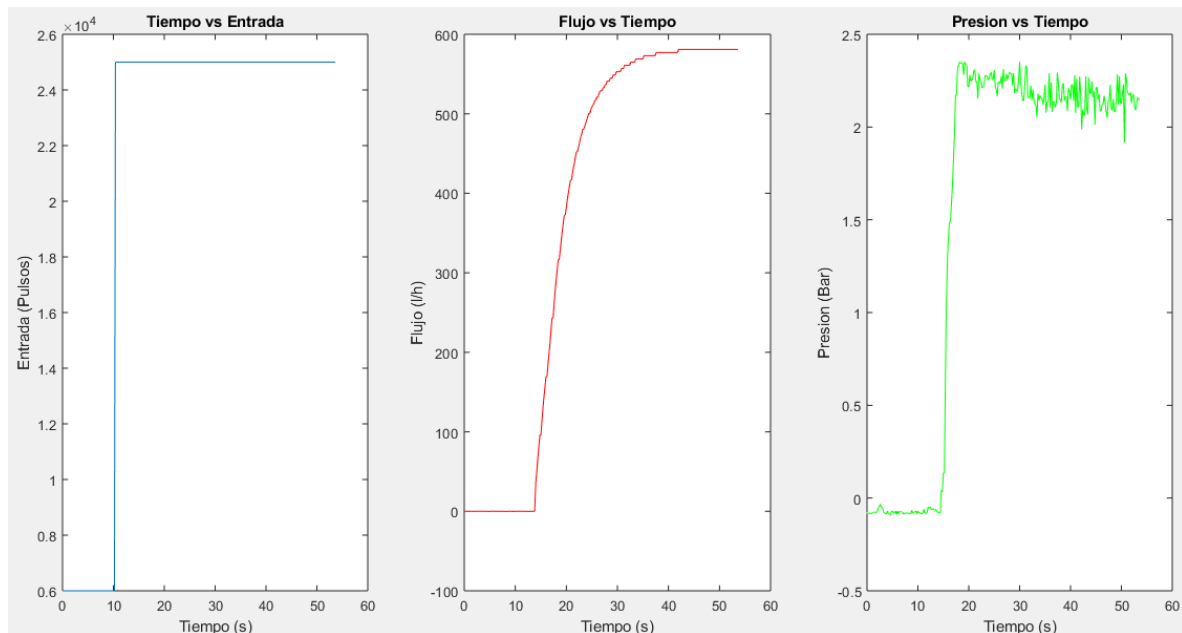
Según la metodología propuesta, los resultados obtenidos junto con su respectivo análisis son los siguientes:

**Actividad 1:** Se capturaron las medidas de tiempo, entrada (bits para el variador de frecuencia que controla a la bomba), flujo y presión mediante Matlab por OPC.

### Actividad 2:

Para el tratamiento de los datos, se enfocó el análisis en el momento en que se inyecta un step al sistema (6000 a 25000 bits) hasta que se estabilizó el sistema de flujo y presión. Además, se ajustó el vector de tiempo de acuerdo con el delta existente entre cada medida dada por Matlab. Tal y como se observa a continuación:

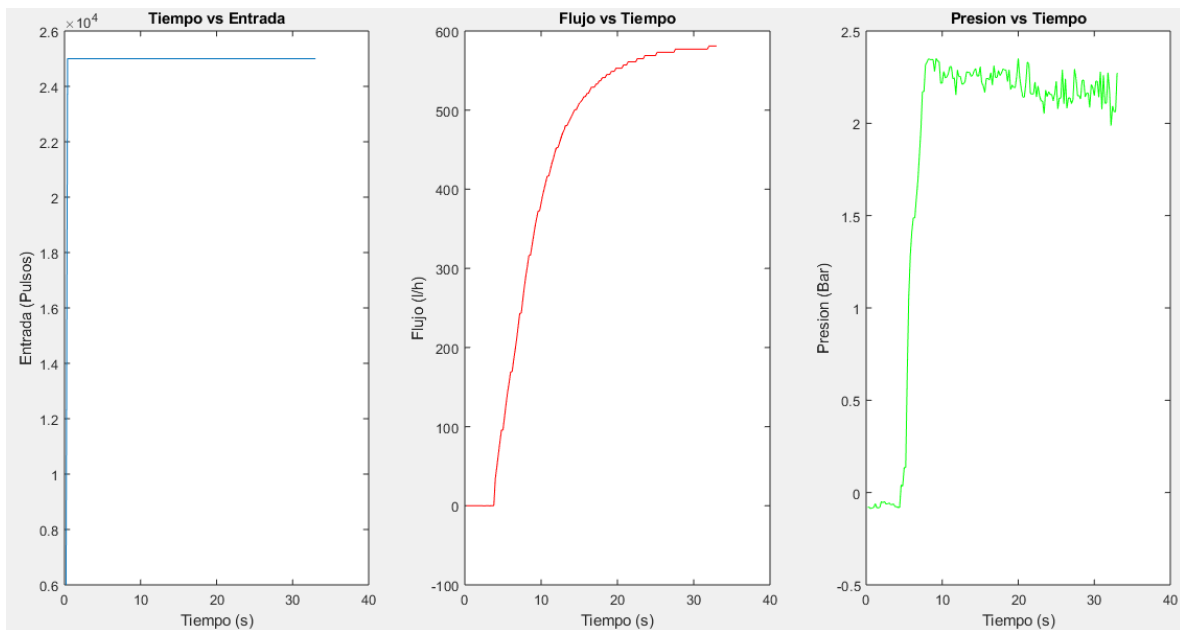
Figura 7. Datos sin tratamiento de datos.



Como se observa en la Figura 7, el programa en simulink toma medidas desde mucho antes que el sistema sea excitado y mucho después de que el sistema se estabilizo.

Al recortar esta informacion innecesaria, se obtiene la ventana de informacion que se desea analizar para la identificación tanto para el sistema de presión y el de flujo. En la figura 8 se observa este tratamiento, además del ajuste del vector de tiempo.

Figura 8. Datos con tratamiento de datos.



Luego de acondicionar los datos medidos, se procede a identificar el sistema mediante un modelo de primer orden más tiempo muerto [1], obtenido la siguiente expresión:

- Función de transferencia para el sistema de Flujo:

$$G_{p1F} = \frac{0.03058}{\exp(-4.09 \cdot s) * 5.348 s + 1}$$

- Función de transferencia para el sistema de Presión:

$$G_{p1p} = \frac{0.0001283}{\exp(-4.98 \cdot s) * 1.392 s + 1}$$

En ambas funciones de transferencia se observa que existe un tiempo muerto mayor a 4 segundos, debido a la distancia en que están ubicados los transmisores, además, es evidente que el transmisor de presión está más alejado del proceso por lo que tiene un tiempo muerto mayor.

Figura 9. Identificación del sistema de Flujo.

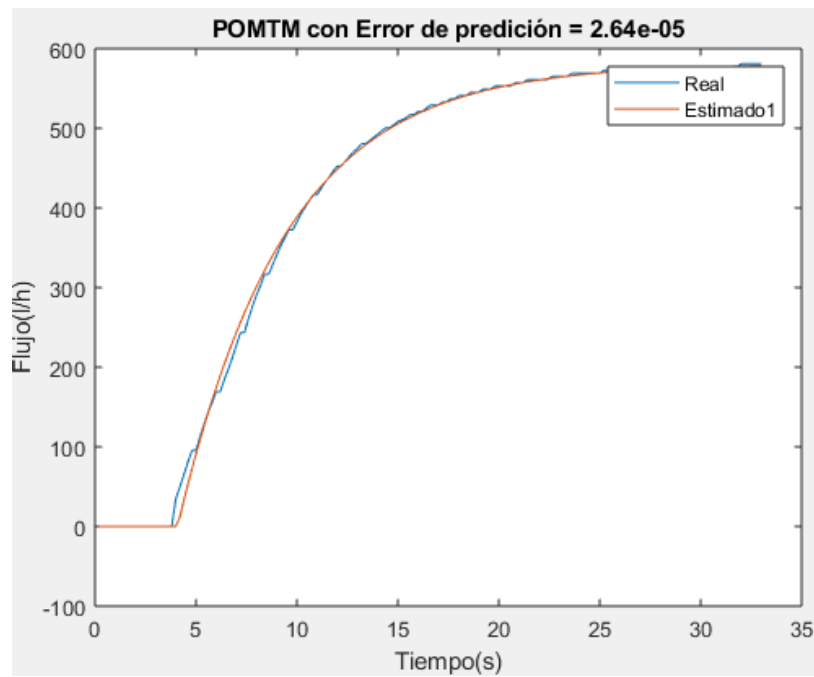


Figura 10. Características estáticas del sistema de Flujo sin control.

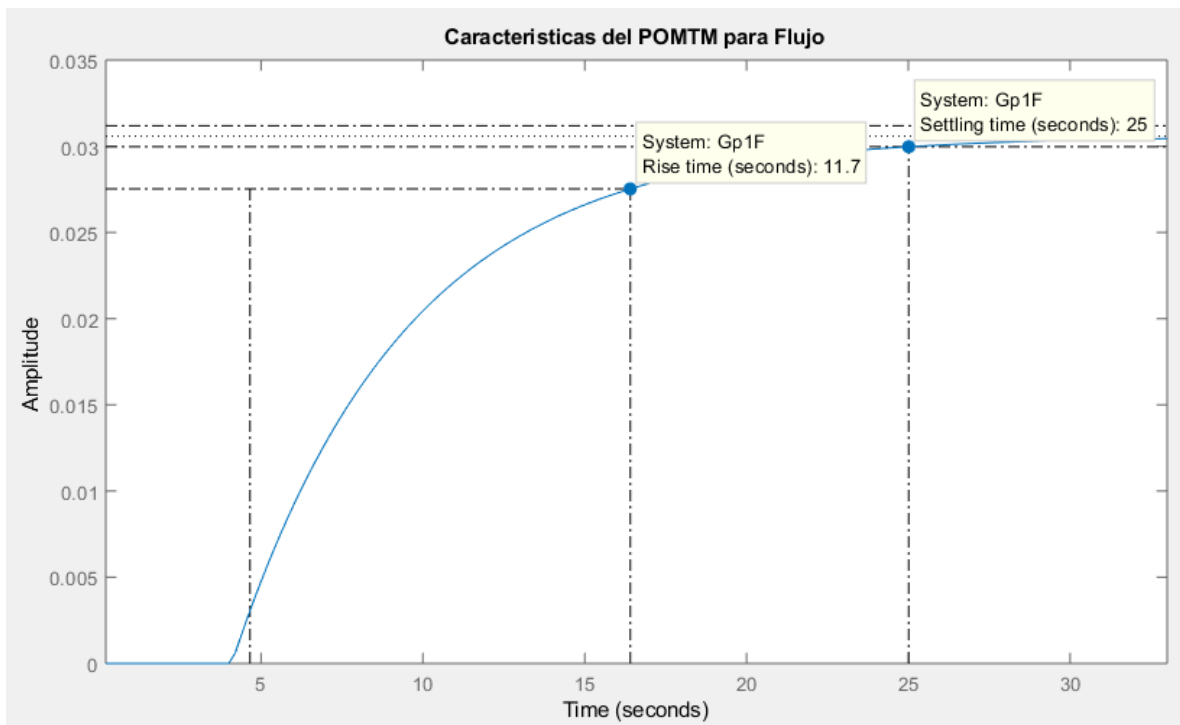


Figura 11. Identificación del sistema de presión.

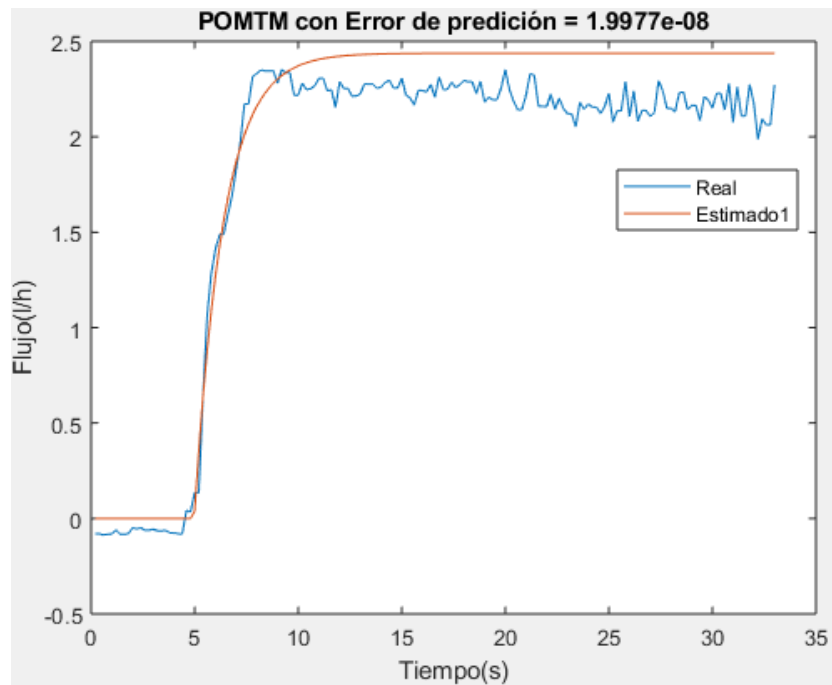
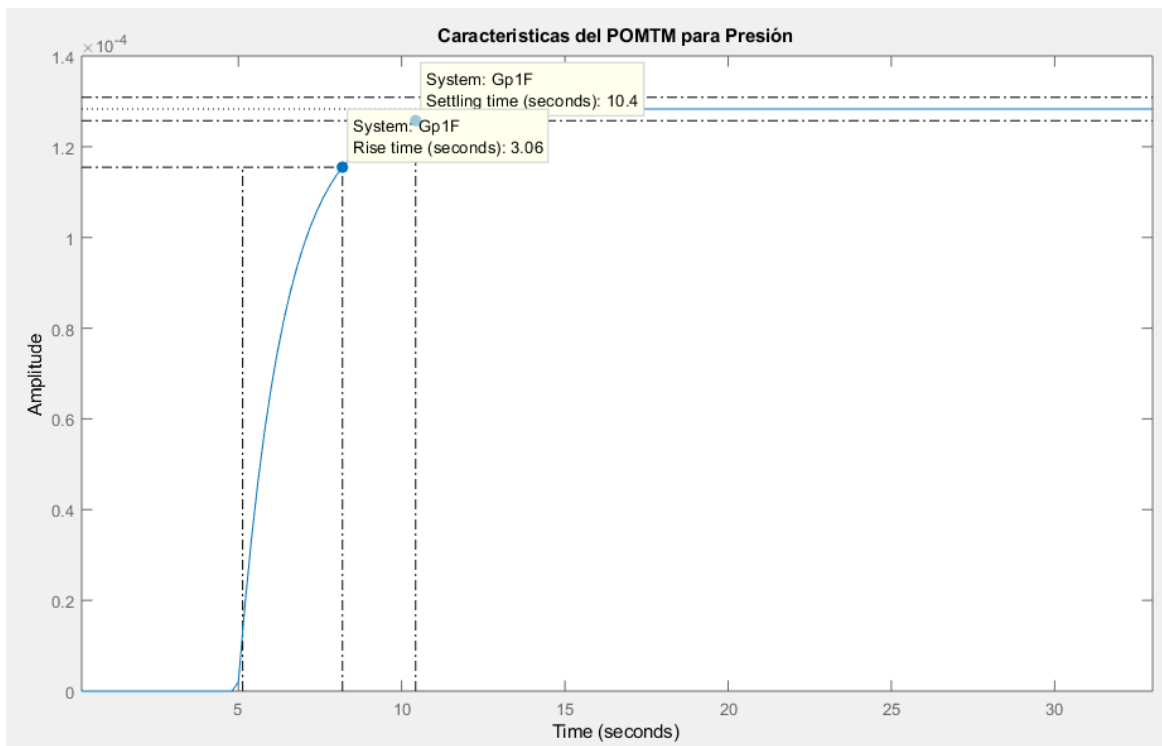


Figura 12. Características estáticas de la presión sin control.



En las figuras 9 y 11 se puede comprobar que el transmisor de presión reacciona más rápido que el de Flujo, pero su identificación posee más error de predicción.

### Actividad 3:

El diseño de los controladores PI tanto para el sistema de flujo, como para el de Presión, fue realizado sin tener en cuenta el tiempo muerto, obteniendo los siguientes resultados:

- Controlador PI para el sistema de flujo, con error en estado estacionario cero: **(Teorema del límite)**

$$PIF = \frac{621 s + 526.3}{s}$$

- Controlador PI para el sistema de presión: **(Asignación de polos)**

$$PI = \frac{1390 s + 2281}{s}$$

Se puede observar que la ganancia  $K_p$  y  $K_i$  son pequeñas para el sistema de flujo en comparación con el de presión, lo cual se debe a la rapidez con que reacciona el transmisor de presión.

Figura 13. Respuesta en lazo cerrado con Control PI para el sistema de Flujo.

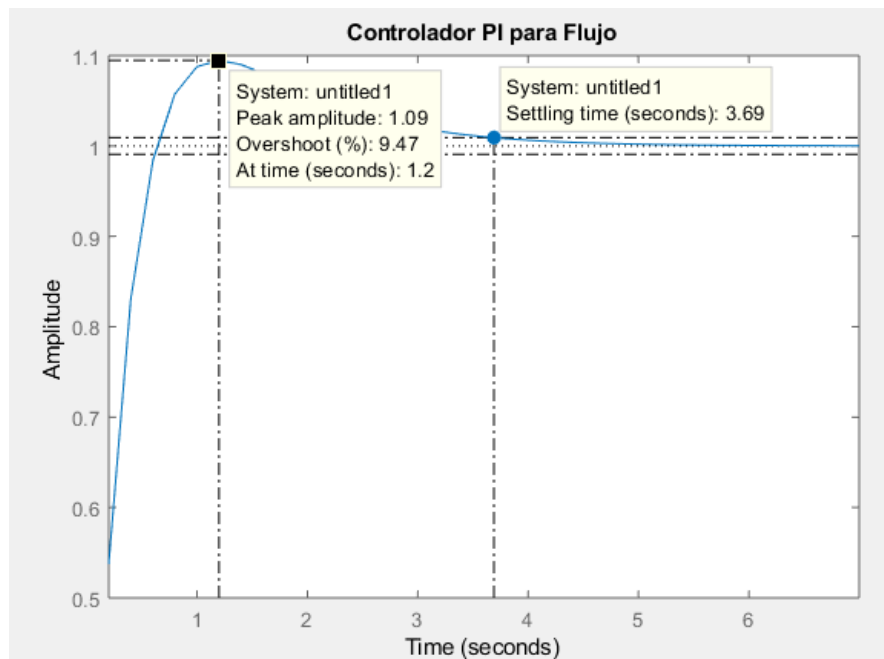




Figura 14. Respuesta en lazo cerrado para el sistema de Presión con control PI.

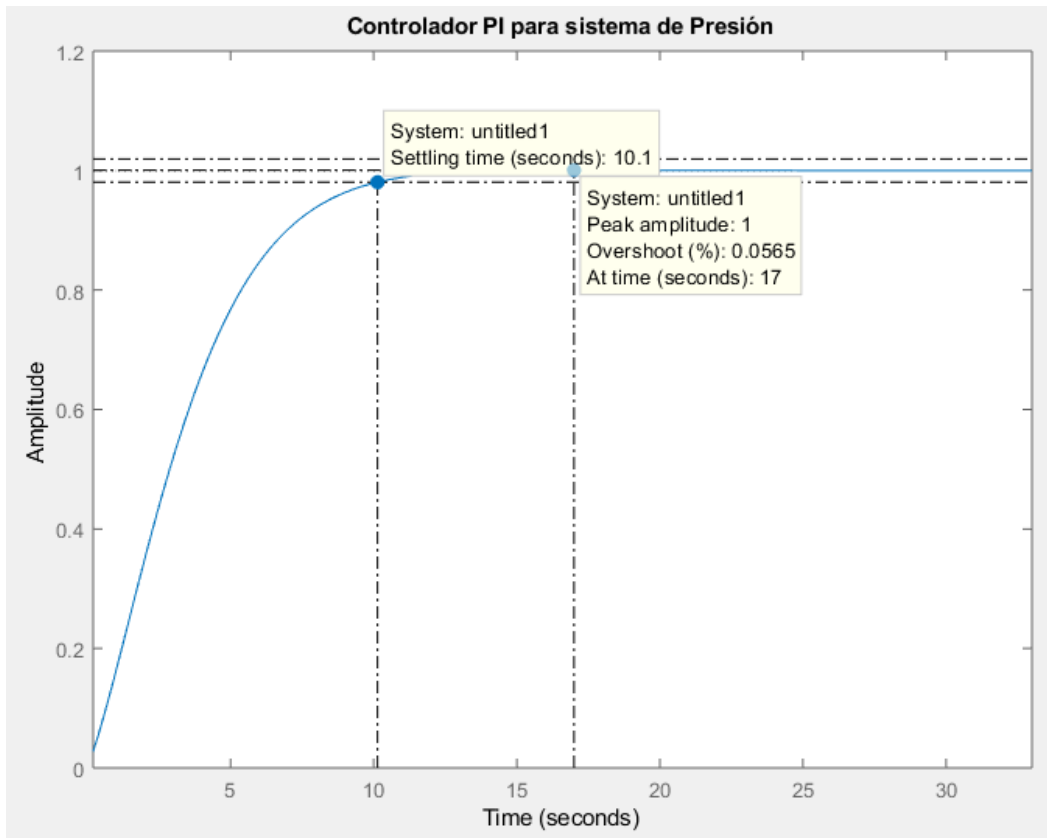


Tabla 1. Constantes de comportamiento para los controladores PI.

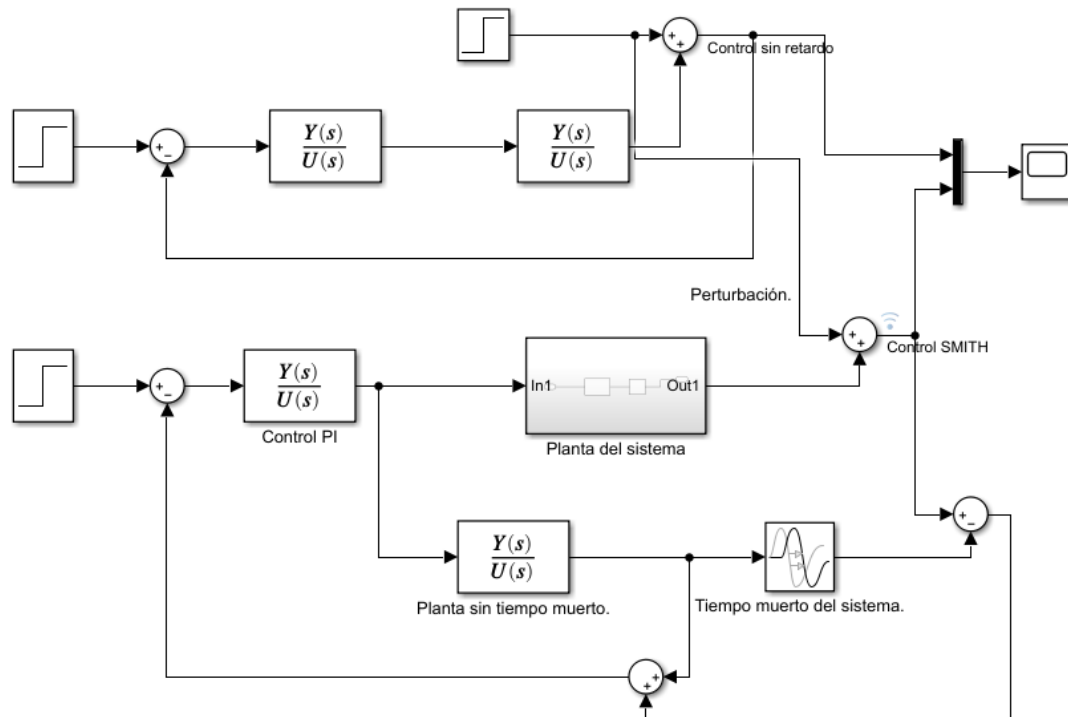
Constantes.	Control PI sistema de Flujo.	Control PI sistema de Presión.
Kp	621	1390
Ki	526.3158	2281
Tiempo de establecimiento.	3.69 segundos.	10.1 segundos.
Overshoot.	9.47%	5.65%

Del comportamiento del sistema de presión y Flujo en lazo cerrado con sus respectivos controlador PI se puede concluir, que los **Overshoots** son aceptables ya que son menores al 25%, el **tiempo de establecimiento** es rápido en ambos, en comparación con el sistema sin control.

## - CONTROL SMITH:

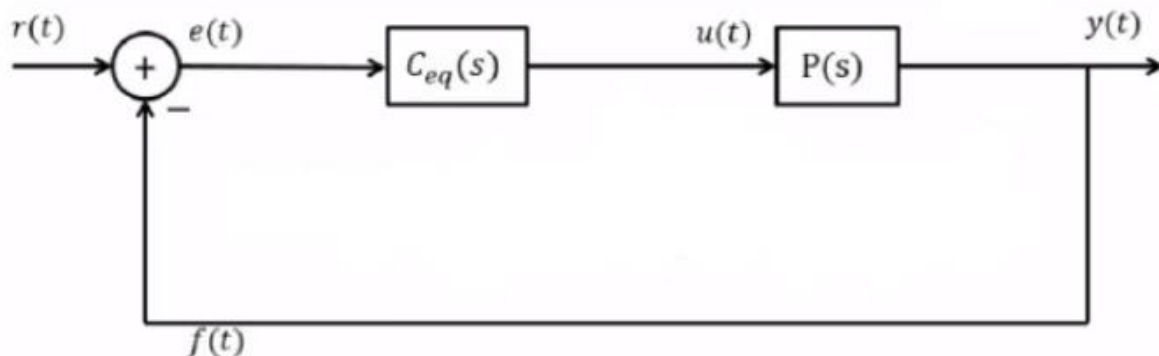
Para realizar el controlador predictivo Smith es necesario utilizar el control PI diseñado y la planta más tiempo muerto. Su estructura se puede construir en Matlab Simulink (Figura 15) para comprobar su efectividad frente a perturbaciones y el tiempo muerto para ambos sistemas de flujo y presión.

Figura 15. Esquema del controlador Smith.



Sin embargo, la estructura mostrada en Simulink resulta ser compleja a la hora de implementarla en ecuaciones de diferencias, es por ello que se debe realizar una reducción de bloques para trabajar el controlador Smith fácilmente, así:

Figura 16. Controlador Smith por reducción de bloques.



Donde:

$$C_{eq}(s) = \frac{C(s)}{1 + C(s)G_n(s) - C(s)P_n(s)}$$

$r(t)$ : referencia

$e(t)$ : error

$u(t)$ : señal de control

$y(t)$ : salida

$C(s)$ : controlador

$P(s)$ : planta

$\hat{y}(t)$ : salida predicha

$G_n$ : planta predicha

$L_n$ : retardo

$ep$ : error de predictor

Usando la reducción de bloques se puede implementar fácilmente este controlador. Los resultados de implementación fueron los siguientes:

Figura 17. Controlador Smith para el sistema de **Flujo**.

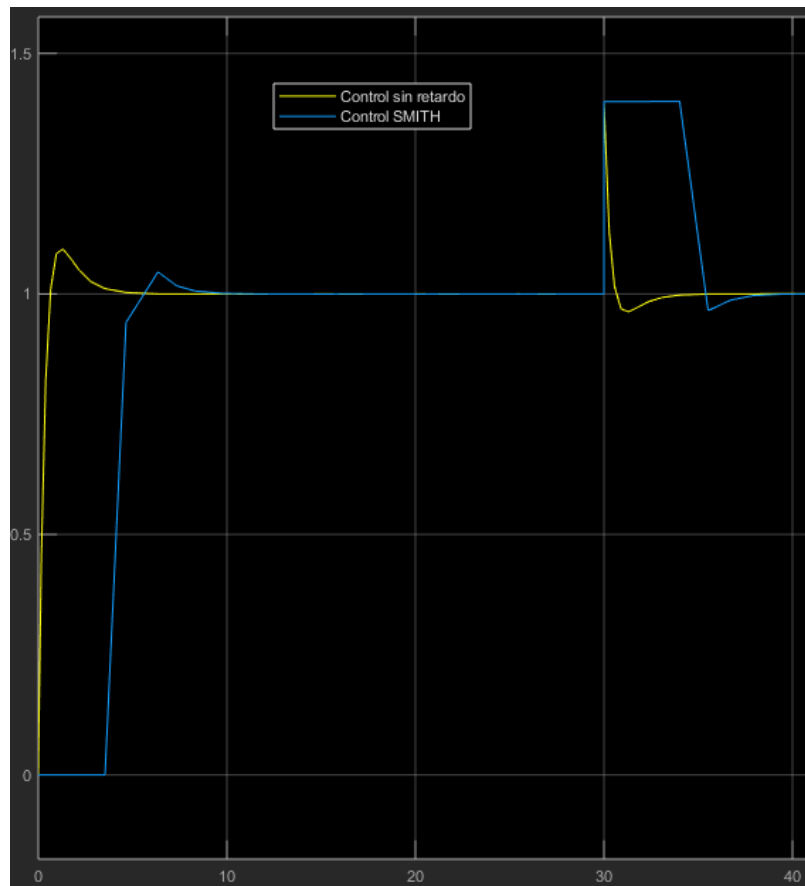
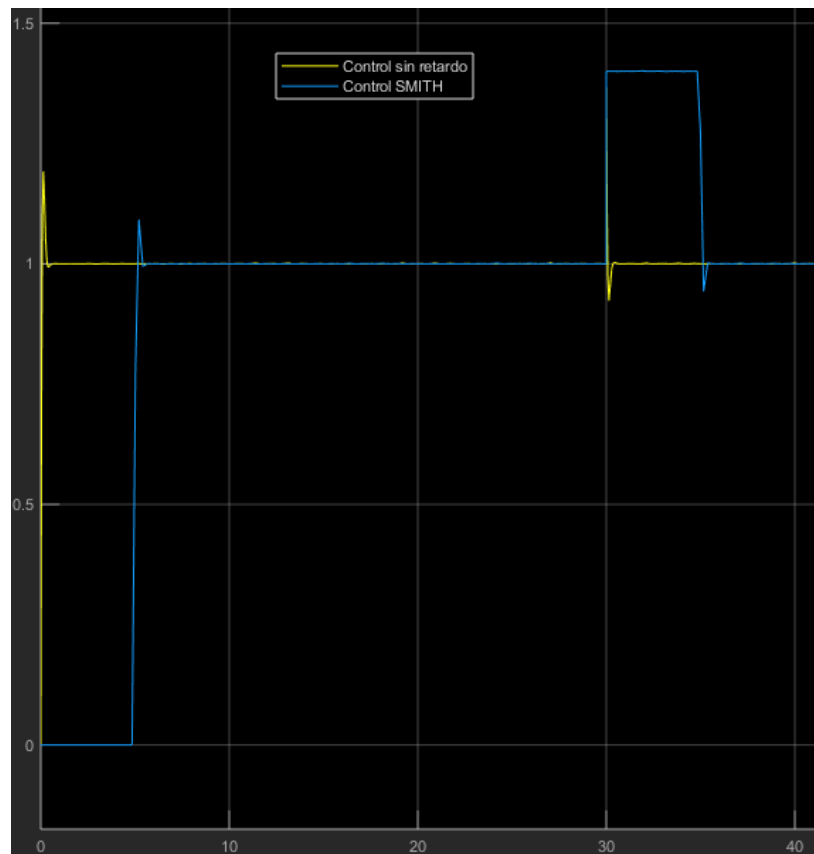


Figura 18. Controlador Smith para el sistema de **Presión**.



En las figuras 17 y 18 se puede observar el controlador Smith tanto para flujo como para presión respectivamente, demostrando su efectividad ante perturbaciones y el efecto del tiempo muerto (señal azul).

Sin embargo, para hacer la implementación en Matlab y posteriormente en TIA portal, se debe utilizar la planta más tiempo muerto con aproximación de **Pade**.

#### - **SISTEMA DE FLUJO:**

Se necesito aproximación de Pade=2 para realizar un buen controlador Smith, obtenido lo siguientes:

Función de transferencia de la planta con Pade=2 para **Flujo**:

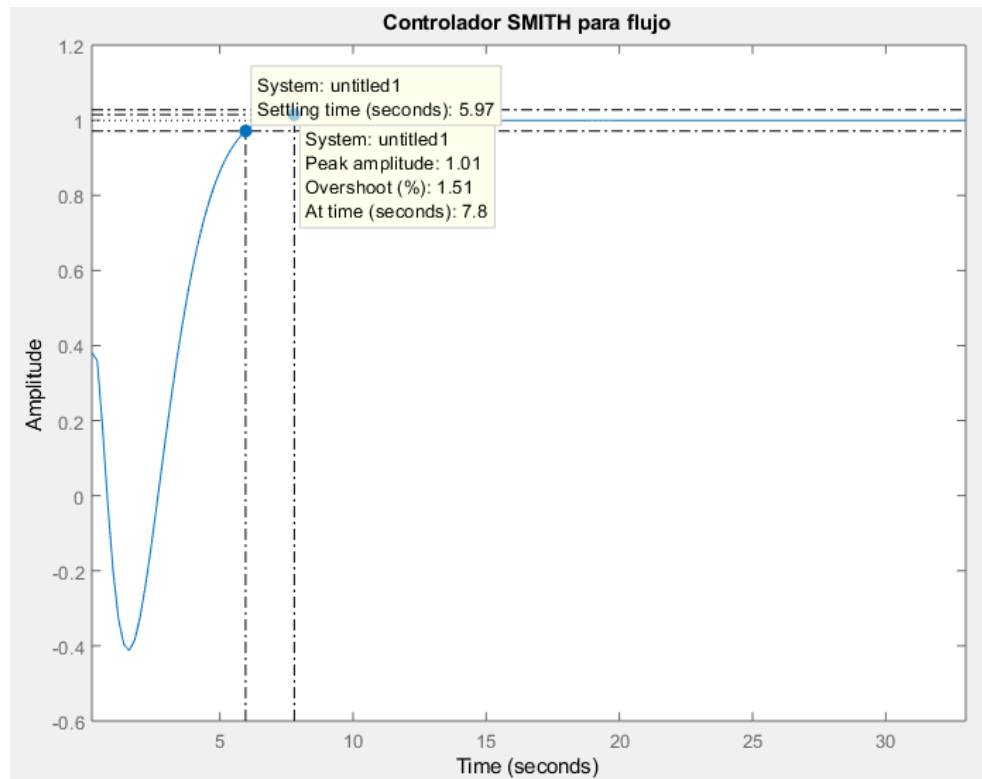
$$\text{GrF} = \frac{0.03058 s^2 - 0.04482 s + 0.0219}{5.348 s^3 + 8.837 s^2 + 5.294 s + 0.7159}$$

Controlador Smith por reducción de bloques para **Flujo**:

Ceq =

$$\frac{3321 s^5 + 8923 s^4 + 9491 s^3 + 4715 s^2 + 980.7 s + 70.43}{5.348 s^5 + 9.833 s^4 + 62.61 s^3 + 59.29 s^2 + 8.952 s + 0.0006888}$$

Figura 19. Controlador Smith para el Flujo.



#### - SISTEMA DE PRESIÓN:

Para el sistema de presión se necesitó un Pade=5 para poder realizar un buen predictor Smith, obteniendo lo siguiente:

Función de transferencia aproximada para **Presión**:

GrP =

$$\frac{-0.0001283 s^5 + 0.0007732 s^4 - 0.002174 s^3 + 0.003494 s^2 - 0.003158 s + 0.001269}{1.392 s^6 + 9.389 s^5 + 29.62 s^4 + 54.85 s^3 + 61.49 s^2 + 38.38 s + 9.887}$$

## Controlador Smith con reducción de bloques para **Presión**:

Ceq =

$$1935 \, s^8 + 1.762e04 \, s^7 + 7.425e04 \, s^6 + 1.888e05 \, s^5 + 3.139e05 \, s^4 + 3.449e05 \, s^3 + 2.404e05 \, s^2 + 9.532e04 \, s + 1.62e04$$

$$1.392 \, s^8 + 10.75 \, s^7 + 37.2 \, s^6 + 82.6 \, s^5 + 115.2 \, s^4 + 98.46 \, s^3 + 58.17 \, s^2 + 17.45 \, s - 0.0002554$$

Figura 20. Controlador Smith para presión.

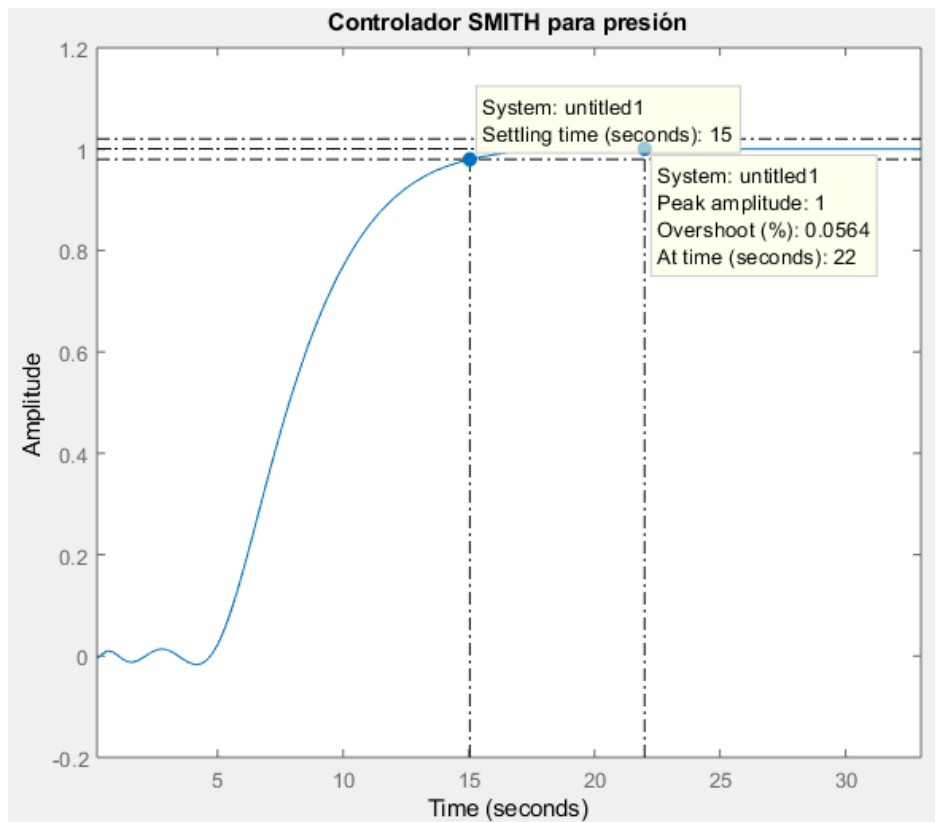


Tabla 2. Comparación entre control PI y Smith.

Constantes.	Control PI sistema de Flujo.	Control PI sistema de Presión.	Control Smith de Flujo	Control Smith de Presión
Kp	621	1390	621	1390
Ki	526.3158	2281	526.3158	2281
Tiempo de establecimiento.	3.69 segundos.	10.1 segundos.	5.95 segundos	15 segundos.
Overshoot.	9.47%	5.65%	9.47%	5.65%

Según las figuras **19** y **20** junto con la tabla 2, se puede observar que el controlador Smith aumenta el tiempo de establecimiento debido al tiempo muerto del sistema, pero las condiciones de diseño se mantienen para las constantes Kp y Ki.

#### Actividad 4:

##### Discretización:

Tabla 3. Características estáticas sin control.

Características estáticas.	Sistema de Flujo.	Sistema de presión.
Tiempo de establecimiento:	25 segundos	10.4 segundos.
Tiempo de subida:	11.7 segundos.	3.06 segundos.
Tiempo de muestreo:	0.7 segundos	0.7 segundos.

Para seleccionar el tiempo de muestreo se tomo el criterio del tiempo de subida para el de flujo y el criterio de tiempo de establecimiento para la presión.

##### **Discretización para el sistema de flujo:**

- Planta de flujo discreta:

$$G_z =$$

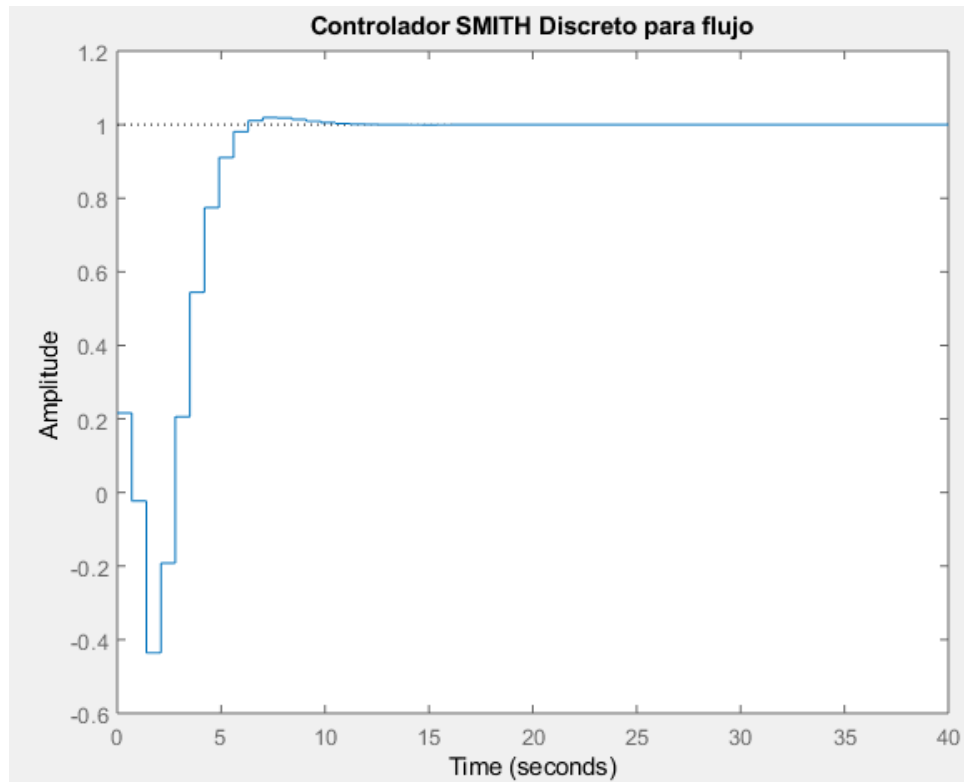
$$\frac{0.0006746 z^3 - 0.001467 z^2 - 0.0002629 z + 0.001879}{z^3 - 2.017 z^2 + 1.359 z - 0.315}$$

- Control Smith para Flujo discreto:

$$G_{cz} =$$

$$\frac{408.9 z^5 - 1405 z^4 + 1921 z^3 - 1310 z^2 + 447.3 z - 61.28}{z^5 - 2.184 z^4 + 2.101 z^3 - 1.913 z^2 + 1.37 z - 0.3746}$$

Figura 21. Controlador Smith discreto para Flujo.



### Discretización para el sistema de presión:

- Planta de presión discreta:

Gz =

$$\frac{-4.441e-07 z^6 + 3.619e-06 z^5 - 1.186e-05 z^4 + 1.829e-05 z^3 - 7.148e-06 z^2 - 1.558e-05 z + 2.578e-05}{z^6 - 2.203 z^5 + 2.287 z^4 - 1.412 z^3 + 0.5272 z^2 - 0.1115 z + 0.01031}$$

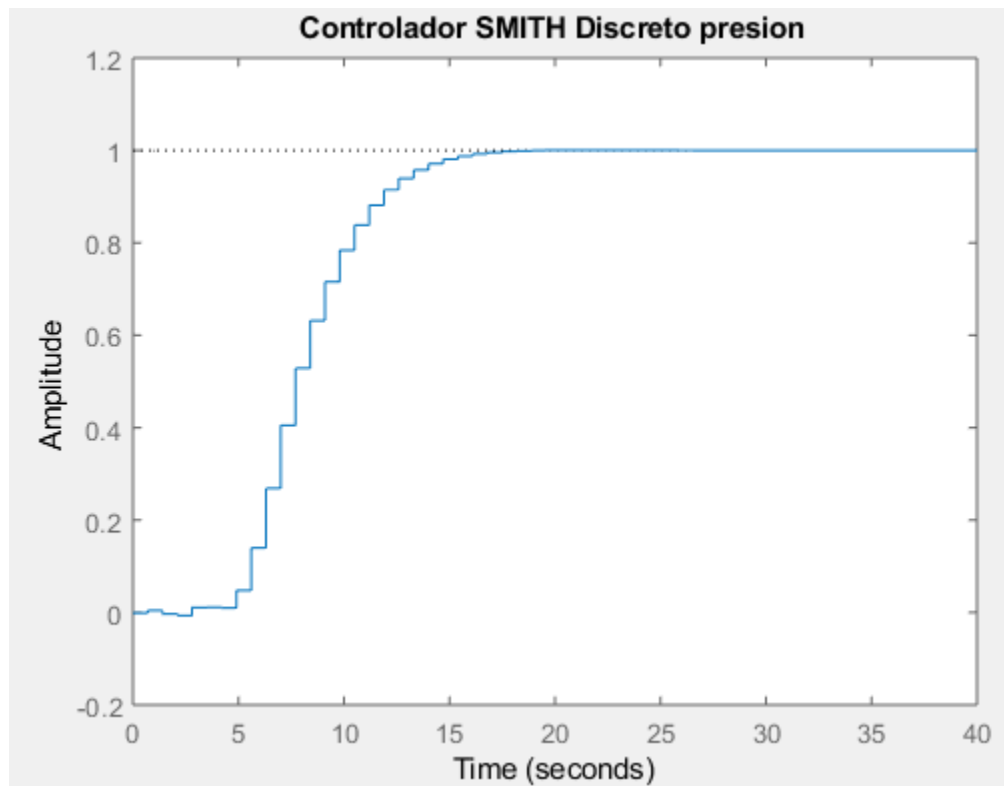
- Control Smith para presión:

Gcz =

$$\frac{2070 z^8 - 6356 z^7 + 9027 z^6 - 7770 z^5 + 4394 z^4 - 1651 z^3 + 398.2 z^2 - 55.84 z + 3.449}{z^8 - 3.681 z^7 + 6.111 z^6 - 6.053 z^5 + 3.918 z^4 - 1.653 z^3 + 0.3862 z^2 - 0.02511 z - 0.002945}$$



Figura 22. Controlador smith discreto para presión.



#### Ecuación en diferencias:

Al expresar en ecuación en diferencias el controlador Smith discreto tanto de flujo como de presión, se puede hacer la implementación en PLC. Pero antes se comprueba con Matlab su funcionamiento:

#### **Sistema de Flujo:**

El flujo se comporta de 0 l/h hasta 600 l/h con una señal de control que va desde los 6000 bits hasta los 27000 bits. La respuesta controlada se observa en la **Figura 23**, donde se observa que la señal de respuesta posee un pico negativo antes de controlar, esto se debe a la señal de control (**Figura 24**).

Según la figura 24, se observa que antes de controlar a la referencia deseada, se envía un pico en bits que puede sobrepasar el rango permitido (6k a 27k bits), por lo que se hace necesario un saturador para el controlador Smith de flujo.

Una de las razones por la cual se presenta este pico, se debe al diseño utilizado para el controlador, que para el caso del flujo fue por teorema del limite.

Figura 23. Respuesta del sistema de Flujo controlada.

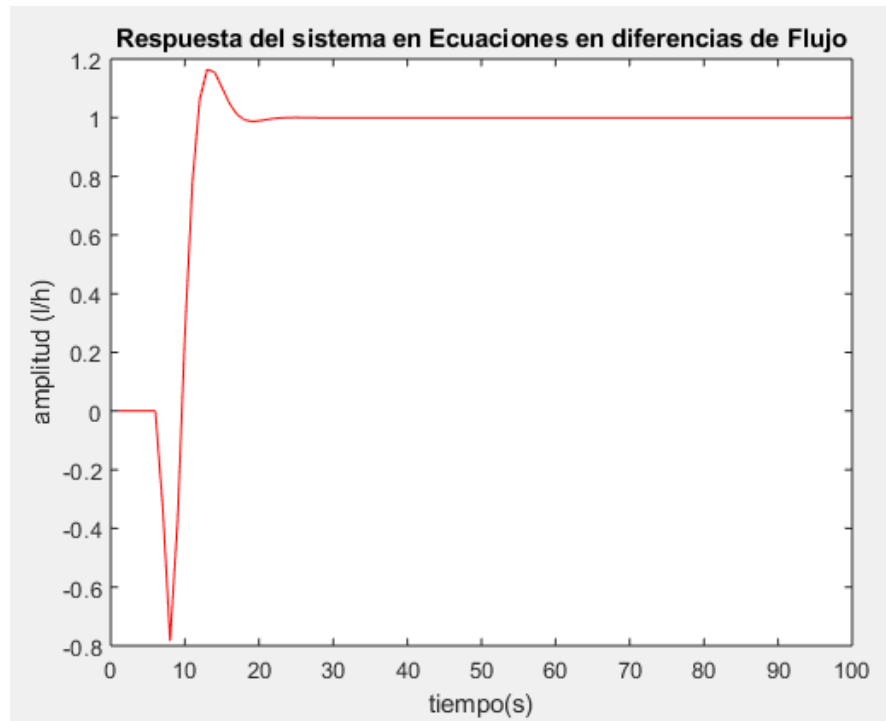
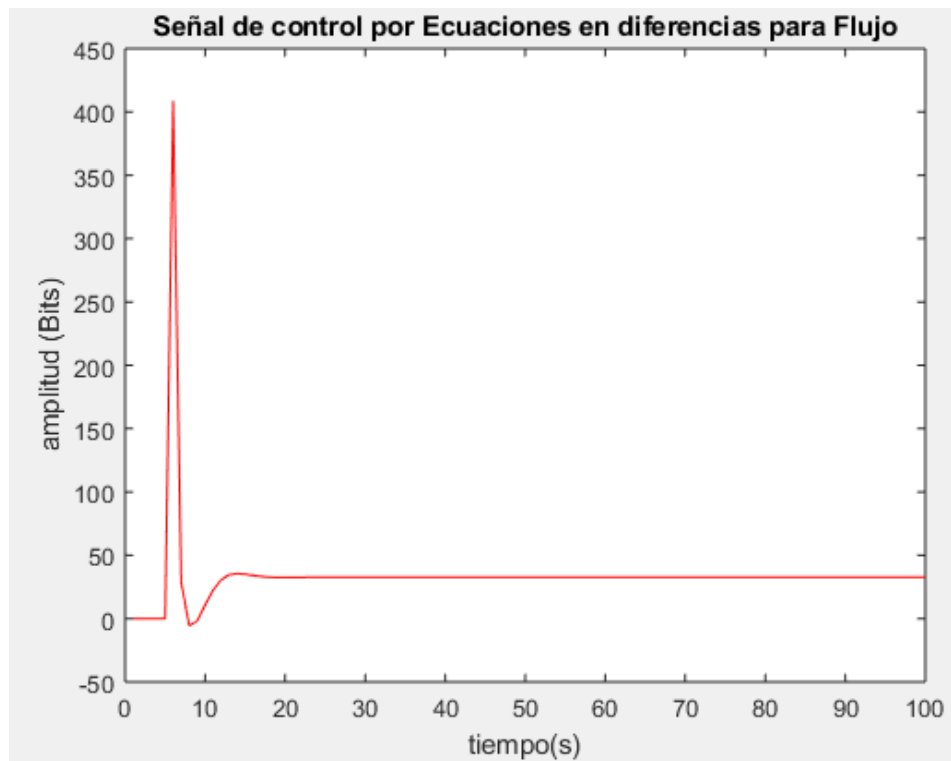


Figura 24. Señal de control para el sistema de Flujo.



### Sistema de presión:

La presión dentro de la unidad de entrenamiento en automatización se comporta entre 0 Bar y 2.5 Bar, controlando el variador de la bomba entre 6k y 27k bits respectivamente.

En la **figura 25**, se observa la respuesta de la presión con controlador Smith y se observa como el sistema se establece rápidamente sin tener valores negativos. Además la señal de control (**Figura 26**) tiene un comportamiento suave y dentro de los rangos permitidos, por lo que puede ser implementado sin saturador.

Figura 25. Respuesta para el sistema de Presión con control Smith.

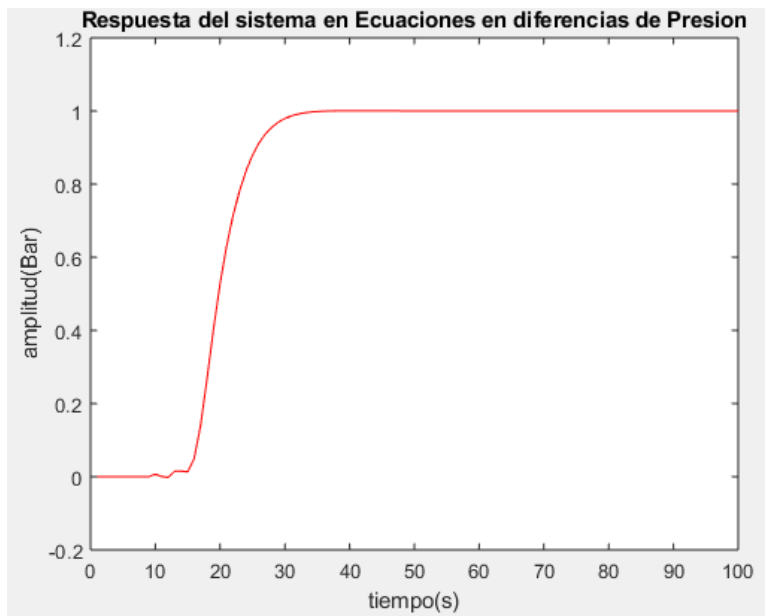
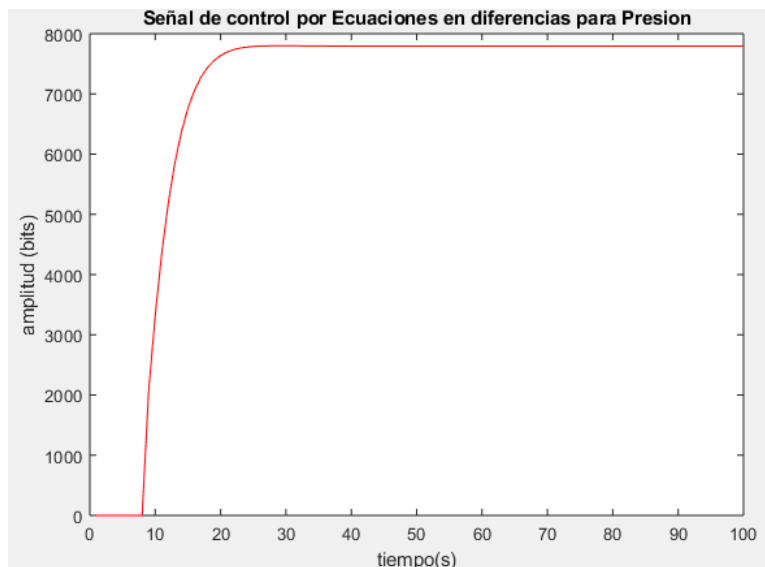


Figura 26. Señal de control para el sistema de Presión.

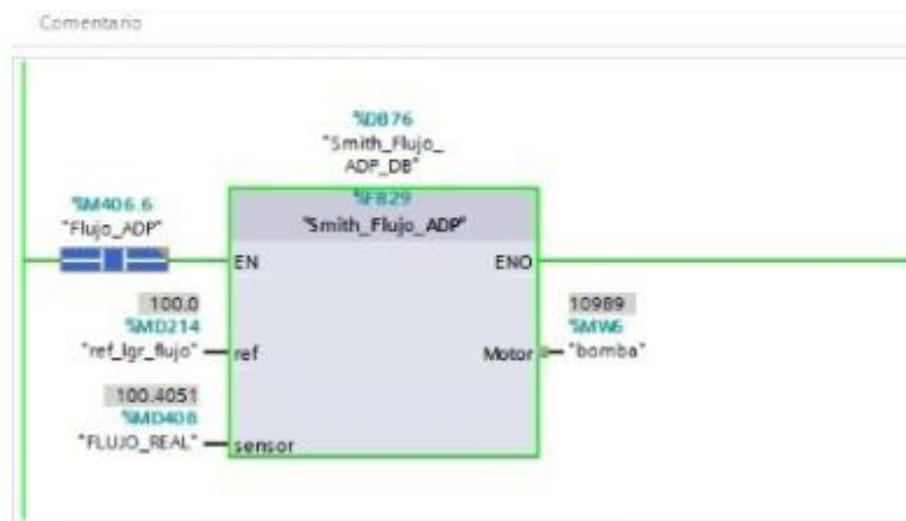


## IMPLEMENTACIÓN EN PLC:

### - Flujo:

Para el controlador de flujo se obtuvo un control que seguía referencias y además sin error en estado estacionario, sin embargo, se tuvo un pico en la señal de control en el momento de cambio de referencia, que luego lograba estabilizarse y llegar al valor establecido, a pesar de aumentar el rango de aproximación de la función de transferencia, disminuir y aumentar constantes a manera de sintonización en el diseño del controlador, pero este problema no fue solucionado.

Figura 27: Puesta en marcha controlador de flujo.



Para observar el comportamiento de forma grafica se realiza una conexión OPC entre TIA y Matlab para recolectar datos de la señal de salida y la señal de control.

En la **Figura 28**, se observa la respuesta para el sistema de flujo con control Smith y se observa que el sistema controla para cualquier referencia dentro del rango permitido (0 l/h – 620 l/h). Para la prueba se probaron dos referencias: 200 l/h y 400 l/h y el sistema logra estabilizar con error estacionario cero en **49 segundos**.

Además, se observa tanto en la respuesta del sistema como en la señal de control, que antes de estabilizar la señal, se produce un overshoot que se sale de los rangos permitidos, tal y como sucede en Matlab (Figura 24).

En la **Figura 29** se observa la señal de control enviada al sistema de flujo y se observa el efecto del saturador que evita que la señal salga de los valores permitidos. Este pico dura aproximadamente 29 segundos en los que el saturador trabaja y luego se estabiliza.

Figura 28. Respuesta del sistema de flujo en PLC.

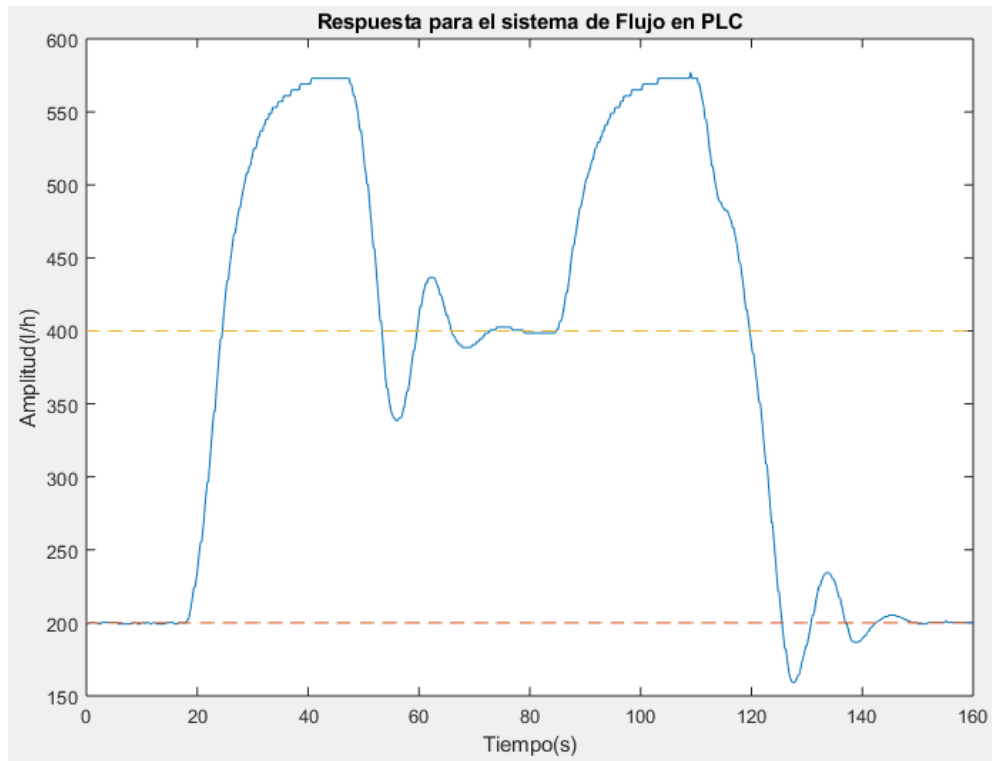
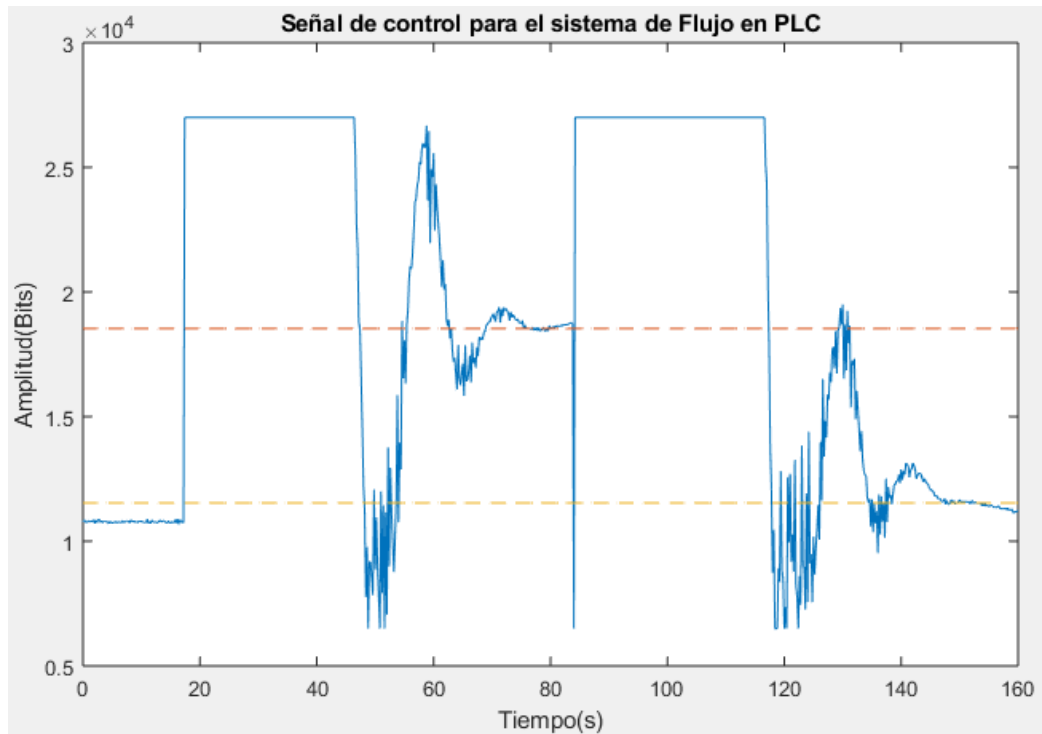


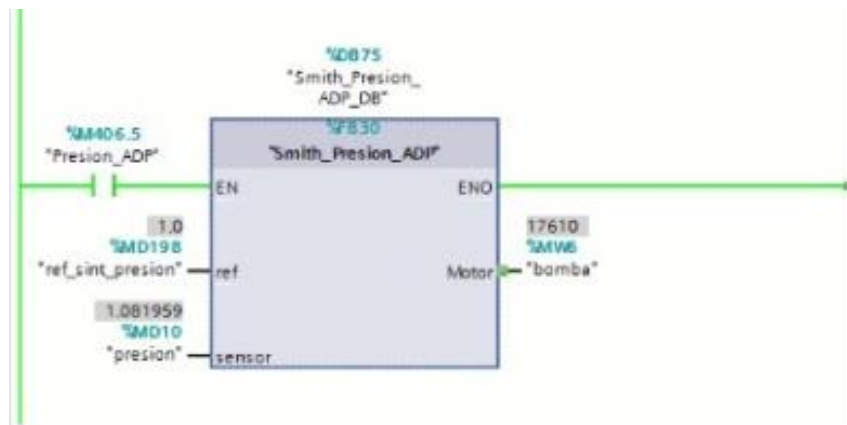
Figura 29. Señal de control para el sistema de flujo en PLC.



## Presión:

Para la presión la puesta en marcha fue mucho más adecuada al diseño del controlador, el cual fue por asignación de polos, este sistema logro controlar de manera óptima, incluso con variaciones pronunciadas tanto de subida como bajada, las oscilaciones que presenta el sistema se deben al funcionamiento del sensor, que toma lecturas ligeramente erróneas en algunos instantes de tiempo, cabe resaltar que el controlador de presión se implementó sin saturador, por lo que en ningún momento toma valores excesivamente grandes que afecten la dinámica del sistema o afecten el variador, por lo que se puede afirmar que el controlador de presión cumplió sus objetivos de manera acertada.

Figura 30. Puesta en marcha controlador de presión.



Para comprobar el funcionamiento del controlador Smith para la presión de forma gráfica, se estableció una conexión OPC entre TIA y Matlab, donde se recolectaron datos de presión y control ante diferentes referencias.

En la **Figura 31**, se observa el funcionamiento del controlador ante diferentes referencias de presión, logrando un tiempo de establecimiento de **3 segundos**. Esta rapidez se debe a la naturaleza misma del sistema y a diferencia del controlador de flujo no presenta un pico de control indeseado.

En la **Figura 32**, se observa la señal de control para la presión y se observa que no es necesario un saturador ya que los valores de control están dentro del rango permitido (6k y 27k bits).

Figura 31. Respuesta para el sistema de presión en PLC.

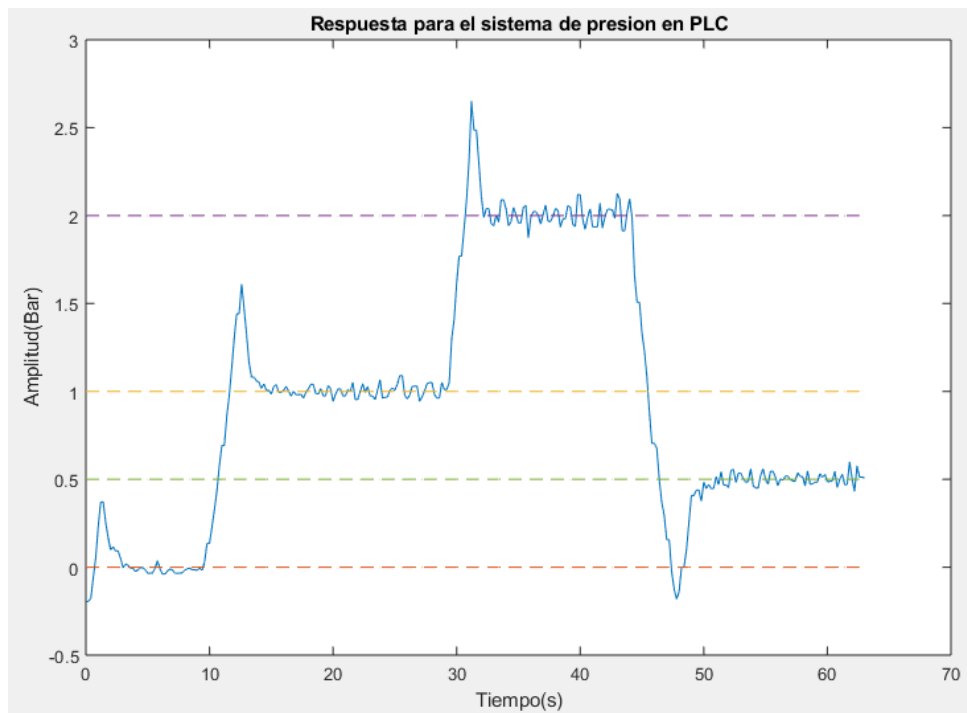
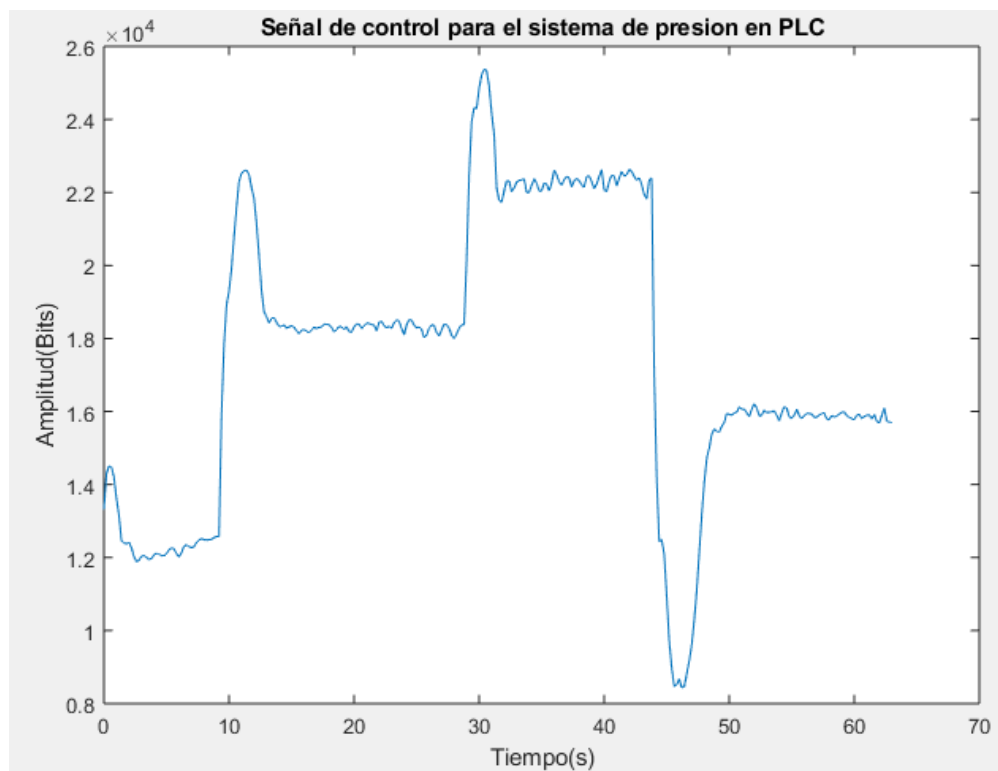


Figura 32. Señal de control para el sistema de presión en PLC.



## CONCLUSIONES.

- La distancia existente entre el proceso, los sensores y/o actuadores, produce el tiempo muerto, lo que complica el diseño de controladores eficientes. Es por ello que existen alternativas como el predictor Smith.
- Cuando se desea que el sistema solo siga referencias y tenga error en estado estacionario de cero, se pueden utilizar los teoremas del valor limite y parte de la teoría del lugar geométrico de las raíces (LGR). Sin embargo, cuando se desean valores de overshoot y tiempo de establecimiento estrictos se puede usar la asignación de polos o la condición de ángulos del LGR.
- Durante la implementación se observó que el diseño por el teorema del valor limite no permite controlar el overshoot fácilmente debido a las constantes  $K_p$  y  $K_i$  que genera. Es por ello que la señal de control para el sistema de Flujo tiene un pico que obliga implementar un saturador y así evitar una señal fuera de los rangos del variador.
- En cambio, el controlador PI para la presión, fue diseñado por asignación de polos por lo que la señal de control no presenta un overshoot fuera del rango, ósea que puede ser implementado sin saturador. Es decir, que ofrece una mejor respuesta en comparación con el control PI de flujo (teorema del límite).
- El tiempo de establecimiento de la presión es mucho más rápida que la de flujo debido a la naturaleza del sistema, donde se observa que la presión es mas sensible ante las variaciones.
- A la hora de discretizar se realiza una aproximación del modelo estimado por lo que la representación en ecuaciones de diferencias resulta ser una aproximación con errores que deben ser tenidos en cuenta sobre procesos críticos.

La implementación del PLC se realizó de manera correcta, por lo que se puede afirmar que la ecuación en diferencias corresponde a una buena representación de la realidad, es por esto que es posible leer las variables reales del sistema y realizar su manipulación interna dentro del bloque de programa, además de que permite evidenciar que la implementación de este controlador no afecta otros procesos en paralelo que se ejecutan para el funcionamiento de la planta, es necesario tener consideraciones al momento de la implementación, tales como tiempos de ejecución, manejo de ciclo scan y demás que son necesarios para generar ese salto



satisfactorio de la simulación a la planta real, por lo que observando los resultados de manera global, se obtuvieron unos controladores que seguían referencias sin error en estado estacionario y para el de presión, se logró un control no dependiente del saturador y que además era muy rápido también debido a la variable a controlar, por lo que se puede determinar que la practica fue realizada de manera óptima, como punto a mejorar para futuras implementaciones, la creación de un filtro para disminuir esas oscilaciones intrínsecas del sensor de presión y observar otras estrategias de control para evitar ese pico inicial en el controlador de flujo.

## **BIBLIOGRAFÍA**

- [1] Víctor M. Alfaro (2007). *Método de identificación de modelos de orden reducido de tres puntos 123c*. Universidad de Costa Riza, 7 páginas.
- [2] K. Warwick and D. Rees, Industrial Digital Control Systems, IET, 1988.
- [3] Control por discretización de sistemas en tiempo continuo. Facultad de ingeniería: Universidad de la Salle.