

# UNIVERSIDAD NACIONAL DE LA PATAGONIA SAN JUAN BOSCO

## AUTOMATIZACIÓN INDUSTRIAL TRABAJO FINAL

---

### Banco de pruebas para control de presión o caudal

---

*Alumnos*

**CAAMIÑA, Daniela**  
**YAPURA, Cristian**

*Docentes*

**Ing. LORENC, Marcelo**  
**Dr. PEÑA, Ramiro**

AGOSTO 2022



## Agradecimientos

*Agradecemos a los docentes de la cátedra de Automatización Industrial que nos dieron la posibilidad de realizar este proyecto y nos facilitaron los dispositivos necesarios para la implementación. A Gerardo Arthz quien diseñó y fabricó el soporte para el motor y variador de velocidad. También al equipo del taller de soldadura y tornería dependientes del departamento de Ingeniería Mecánica quienes fabricaron el soporte de la bomba y ayudaron con el acople del motor- bomba.*

*Al ser la última materia para recibirnos de Ingenieros Electrónicos, y tal como dijimos en el proyecto final de carrera, este trabajo está especialmente dedicado a nuestros padres quienes nos brindaron el apoyo y estuvieron en cada paso de nuestra carrera universitaria, también agradecemos infinitamente a nuestros hermanos, novia/o, familia y amigos quienes fueron nuestro soporte y que nos acompañaron durante todos los años de estudio. Reconocemos y agradecemos a cada persona que de alguna forma nos ayudó a transitar este camino.*

# Índice

<b>1. Introducción</b>	<b>8</b>
<b>2. Objetivo</b>	<b>9</b>
<b>3. Definiciones</b>	<b>10</b>
<b>4. Elementos</b>	<b>12</b>
4.1. Motor . . . . .	12
4.2. Variador de velocidad . . . . .	12
4.3. Módulo didáctico PLC M340 . . . . .	12
4.4. Banco de pruebas . . . . .	13
4.4.1. Transmisor de presión . . . . .	14
4.4.2. Sensor de caudal . . . . .	15
4.5. Diagrama . . . . .	15
<b>5. Preliminares</b>	<b>17</b>
5.1. Programación variador de velocidad . . . . .	17
5.2. Comunicación variador de velocidad - PLC . . . . .	17
5.3. Programación Unity Pro . . . . .	19
5.3.1. Entradas analógicas . . . . .	21
5.3.2. Entradas analógicas para termocuplas o RTD . . . . .	26
5.3.3. Medición de caudal . . . . .	27
<b>6. Desarrollo</b>	<b>28</b>
6.1. Adquisición de datos . . . . .	28
6.1.1. Uso de Matlab . . . . .	29
6.1.2. Estimación de la planta . . . . .	30
6.1.2.1. Comparación numérica- real . . . . .	32
6.1.3. Cálculo del controlador PID . . . . .	34
6.1.4. Pruebas de control . . . . .	36
6.1.4.1. Perturbación al control con PIT02 . . . . .	36
6.1.4.2. Perturbación al control de FT01 . . . . .	36
6.1.5. Valores extremos/críticos de estudio . . . . .	37
6.1.5.1. Máxima presión . . . . .	37
6.1.5.2. Mínima presión . . . . .	38
6.1.5.3. Máximo caudal . . . . .	38
6.2. SCADA . . . . .	40
6.2.1. Configuración driver Modbus . . . . .	40
6.2.1.1. Pruebas mediante ModSim . . . . .	41
6.2.2. Alarmas . . . . .	43
6.2.3. Paradas por bloqueo . . . . .	43
6.2.4. Datos históricos . . . . .	43
<b>7. Mejoras futuras</b>	<b>45</b>
<b>8. Conclusiones</b>	<b>46</b>

<b>9. Referencias y Versiones de programas utilizados</b>	<b>47</b>
<b>A. Anexo: Diagrama P &amp; ID</b>	<b>48</b>
<b>B. Anexo: Tabla de Direcciones ModBus</b>	<b>49</b>
<b>C. Anexo: Manual BANCO-SCADA</b>	<b>51</b>
C.1. Características generales . . . . .	51
C.2. Guía de uso . . . . .	51
C.2.1. Pantalla principal . . . . .	51
C.2.1.1. Diagrama del banco de pruebas . . . . .	52
C.2.1.2. Estado del variador - Motor . . . . .	52
C.2.1.3. Alarmas . . . . .	53
C.2.1.4. Modo remoto/ Modo local . . . . .	53
C.2.1.5. Fallas y ventanas de gráficos . . . . .	54
C.2.2. Pantalla de datos en tiempo real . . . . .	54
C.2.3. Pantalla de datos históricos . . . . .	54
<b>D. Anexo: Archivos extras</b>	<b>58</b>

## Índice de figuras

4.1. Motor . . . . .	12
4.2. Variador de velocidad . . . . .	12
4.3. Módulo didáctico PLC M340 . . . . .	13
4.4. Banco de pruebas completo . . . . .	14
4.5. Banco de pruebas . . . . .	14
4.6. Transmisores . . . . .	15
4.7. Diagrama p&id . . . . .	16
5.1. Elección de Altivar 312 . . . . .	17
5.4. Diagrama de comunicación PC- variador . . . . .	17
5.2. Parámetros del variador . . . . .	18
5.3. Lista de parámetros modificados . . . . .	18
5.5. Cable de comunicación . . . . .	18
5.6. Conexión fichas RJ45- DB9 . . . . .	19
5.7. Elección del bastidor . . . . .	20
5.8. Módulos PLC . . . . .	20
5.9. Dirección módulo Ethernet . . . . .	21
5.10. Configuración variador CANopen . . . . .	22
5.11. Configuración variador CANopen . . . . .	23
5.12. HMI simple . . . . .	23
5.13. Programa con bloques MFB . . . . .	24
5.14. Módulo AMI0410 . . . . .	24
5.15. Rango y escalado . . . . .	25
5.16. Bloques de escalado . . . . .	25
5.17. Accesorio TELEFAST ABE-7CPA412 . . . . .	26
5.18. Tipos de conexión . . . . .	26
5.19. Rango y escalado . . . . .	27
5.20. Bloque de escalado . . . . .	27
5.21. Diagrama de flujo del caudalímetro . . . . .	27
6.1. Mapa de memoria . . . . .	28
6.2. Configuración OFS . . . . .	29
6.3. Conexión servidor OPC . . . . .	30
6.4. Cliente OPC en Simulink . . . . .	30
6.5. Configuración de frecuencia de muestreo . . . . .	31
6.6. Parámetros de Strejc con retardo . . . . .	32
6.7. Comparación de la planta estimada y los valores obtenidos para FT01 . .	32
6.8. Comparación de la planta estimada y los valores obtenidos para PIT01 .	33
6.9. Comparación de la planta estimada y los valores obtenidos para PIT02 .	33
6.10. PID Controller . . . . .	34
6.11. Comparación PI para FT01 . . . . .	35
6.12. Comparación PI para PIT01 . . . . .	35
6.13. Comparación PI para PIT02 . . . . .	35
6.14. Diagrama p&id . . . . .	36
6.15. Perturbación al control con PIT02 . . . . .	37
6.16. Perturbación al control con FT01 . . . . .	37
6.17. Máxima presión . . . . .	38
6.18. Mínima presión . . . . .	38

6.19. Máximo caudal . . . . .	39
6.20. Pantalla SCADA . . . . .	40
6.21. Configuración MBE . . . . .	41
6.22. Database con MBE . . . . .	41
6.23. ModSim . . . . .	43
6.24. Datos históricos . . . . .	44
A.1. Diagrama p&id . . . . .	48
C.1. Pantalla principal . . . . .	51
C.2. Partes del sistema SCADA . . . . .	52
C.3. Subpantalla 1 . . . . .	52
C.4. Subpantalla 4 . . . . .	53
C.5. Modo lazo cerrado . . . . .	54
C.6. Pantalla de datos en tiempo real . . . . .	55
C.7. Datos históricos . . . . .	56
C.8. Programa para lectura de datos Históricos . . . . .	57

## **Índice de tablas**

6.1. Valores de los PID . . . . .	34
6.2. Matriz causa- efecto de las alarmas . . . . .	42
B.1. Tabla de variables y direcciones . . . . .	49
C.1. Rangos de las variables . . . . .	54

## Lista de Acrónimos

**CS** : Controlador de velocidad

**DTM** : Gestor de tipos de dispositivo

**FT** : Transmisor de caudal

**FV** : Válvula reguladora de caudal

**HMI** : Interfaz hombre-máquina

**ISA** : Sociedad de Instrumentos de América (EEUU)

**MFE** : Bloques de funciones de movimiento

**OFS** : OPC Factory Server, software de Schneider

**P & ID** : Diagrama de cañerías e instrumentación

**PIT** : Transmisor con indicador de presión

**PLC** : Controlador Lógico Programable

**RTD** : Detectores de temperatura resistivos

**SCADA** : Supervisión, Control y Adquisición de Datos

**SDO** : Objetos o mensajes de servicio

**TE** : Elemento primario de temperatura

**UNPSJB**: Universidad Nacional de la Patagonia San Juan Bosco

**VSD** : Variador de velocidad

## **1. Introducción**

Actualmente en el Laboratorio de Automatización y Control de la Universidad, se cursan distintas materias en las cuales se necesitan herramientas para realizar diversas prácticas, con el fin de afianzar los conocimientos que se adquieren a lo largo del año.

Para llevar a cabo estas actividades con varias etapas, se requiere demasiado tiempo en realizar pruebas sobre un esquema complejo, es decir con varios elementos, ya que se necesita armar un prototipo de banco de pruebas cada vez que sea necesario. Por ejemplo, realizar la conexión de un PLC, variador de frecuencia y un motor puede ser una tarea repetitiva que se busca suprimir.

## 2. Objetivo

El banco de pruebas cuenta con un soporte donde se conecta el motor y sus componentes mecánicos, ademas dentro de esta plataforma existe un sistema de medición que posee sensores, variador y PLC para poder realizar prácticas de laboratorio. Un banco de pruebas puede ser un prototipo de gran desarrollo industrial o simplemente un banco formado para realizar pruebas educativas.

El objetivo de este trabajo final para la cátedra de Automatización Industrial es construir un banco de pruebas para ser utilizado por cualquier persona dentro el laboratorio de Automatización y Control. Se espera realizar uno que sea capaz de controlar la presión o caudal de agua a través de un sistema ideado y construido por nosotros, que cuente con:

- Motor trifásico 1,5kW (*Altium - Schneider Electric*) -Proporcionado por la cátedra-
- PLC (M340 - *Schneider Electric*) -Proporcionado por la cátedra-
- Variador de velocidad (ATV312 - *Schneider Electric*) -Proporcionado por la cátedra-
- Panel de control
  - Botón de emergencia
  - Encendido/ apagado
  - Potenciómetro para variar velocidad
  - Alarmas visuales
- HMI
  - Control general del banco
  - Información en tiempo real
  - Histórico de datos
  - Alarmas

### 3. Definiciones

#### Motor eléctrico

Los motores eléctricos son máquinas que transforman la energía eléctrica en energía mecánica a través de la generación de campos magnéticos.

#### Variador de velocidad

Es utilizado para controlar la velocidad de giro de un motor. Para regular las revoluciones, se debe tener en cuenta las características del motor, ya que este tiene una curva propia de funcionamiento. Un variador es capaz de generar elementos control de aceleración, frenado, seguridad, control de torque y operaciones que mejoran la eficiencia energética del motor.

#### PLC

Es una computadora que se utiliza en la ingeniería de automatización para controlar procesos en las industrias.

#### SoMove

Software que permite configurar variadores de velocidad pertenecientes a la empresa *Schneider Electric*.

#### Unity Pro

Software común de programación, puesta a punto y explotación de los autómatas Modicon, M340, Premium, Quantum y coprocesadores Atrium de la empresa *Schneider Electric*.

#### CANopen

CANopen es un protocolo con aplicación industrial de bajo nivel para aplicaciones de automatización. Conecta dispositivos entre sí mediante mensajes entre pares. Basado en el estándar de comunicaciones físicas CAN. Se utiliza en redes de comunicación tipo esclavo, multamaestro.

#### ModBus

Modbus es un protocolo de comunicaciones utilizado para transmitir información a través de redes en serie entre dispositivos electrónicos, basado en la arquitectura maestro/esclavo o cliente/servidor, diseñado en 1979 por Modicon para su gama de PLC. Convertido en un protocolo de comunicaciones estándar en la industria. Además, esta red de comunicación industrial usa los protocolos RS232/RS485/RS422.

### HMI - SCADA

Ambas tecnologías, HMI y SCADA, son utilizadas en conjunto en la industria de la automatización. SCADA proporciona funciones de supervisión, alarmas y control, mientras que HMI proporciona las herramientas que necesita para desarrollar imágenes que los operadores pueden usar para monitorear su proceso.

### iFIX

Software desarrollado por *General Electric* donde se puede desarrollar aplicaciones sencillas típicas de HMI, o bien, aplicaciones SCADA más complejas como la gestión de elementos y distribución de alarmas.

## 4. Elementos

### 4.1. Motor

El motor (Figura 4.1) posee las siguientes especificaciones:

#### Altium Eff2

- Tipo: TE2A90SP2
- Tensión nominal: 380 V
- Corriente nominal: 3,46 A
- Frecuencia nominal: 50 Hz.
- Potencia: 1.5kW / 2 HP
- Fases: 3
- Factor de Potencia: 0.84



Figura 4.1: Motor

### 4.2. Variador de velocidad

El variador de velocidad que se utilizó (Figura 4.2) tiene las siguientes características:

#### Altivar 312

- Modelo: ATV312HU15N4
- Tensión: 380-500 V
- Frecuencia de entrada: 50-60 Hz
- Potencia: 1.5kW / 2 HP
- Fases: 3



Figura 4.2: Variador de velocidad

### 4.3. Módulo didáctico PLC M340

El Laboratorio de Control de la *UNPSJB* cuenta con un módulo didáctico que posee un PLC modelo Modicom M340. Este cuenta con entradas analógicas, digitales, distintos métodos de comunicación y la capacidad de agregarle otros módulos según las necesidades de los proyectos a desarrollar.

Los módulos con los que se cuenta son:

- P342030: procesador y módulo de comunicación.
- DDM16022: módulo de entradas y salidas digital.
- ART0414: módulo entradas analógicas preparada para sensores de temperatura.
- AMI0410: módulo entradas analógicas de tensión o corriente.

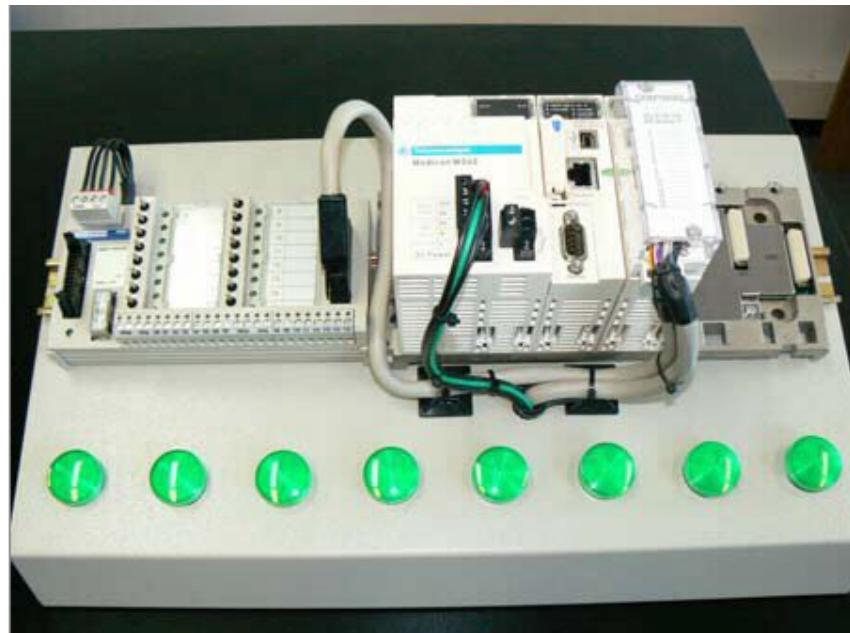


Figura 4.3: Módulo didáctico PLC M340

#### 4.4. Banco de pruebas

Como se nombró en el objetivo, se busca realizar el control de caudal o presión de un sistema hidráulico. Para esto fue necesario realizar la implementación de un banco de pruebas que cuente de tres partes (Figura 4.4).

- Soporte para el motor y variador de velocidad al que se añadió 3 señales luminosas, llave selectora de dos puntos para seleccionar el modo de comunicación, llave selectora de tres puntos (encendido y sentido del motor) y un pulsador de parada de emergencia (Figura 4.5.a).

Tanto el motor y los elementos adicionales fueron cableados (Figura 4.5.b) hacia las borneras del variador de velocidad y se tuvo en cuenta para esto las características y funciones del bornero de control proporcionado por el manual del variador de velocidad[1].

- Soporte para una bomba en desuso de características no conocidas, con su bobinado quemado. Para facilitar el acople motor-bomba se mantuvo la carcasa y eje completo de la misma.
- Circuito hidráulico cerrado, que incluye un tanque, válvulas y sensores de caudal y presión.



Figura 4.4: Banco de pruebas completo

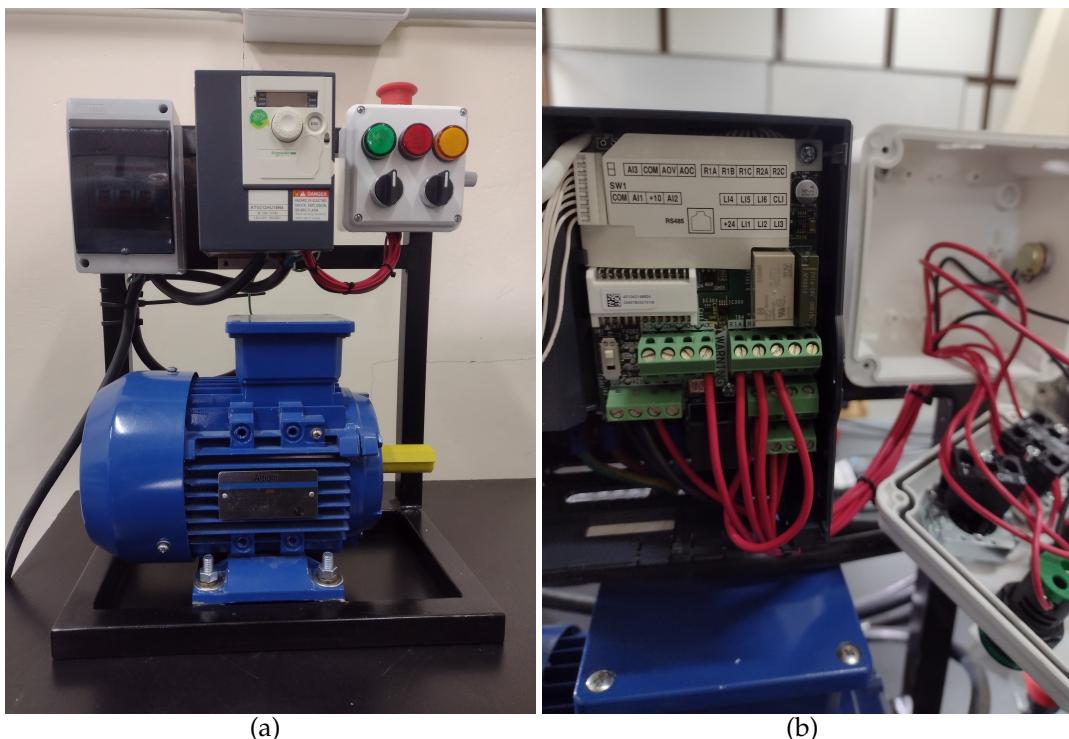


Figura 4.5: Banco de pruebas

### 4.4.1. Transmisor de presión

Para este proyecto se utilizan dos transmisores de presión de montaje en línea modelo EJA530E (Figura 4.6.a).

Las características del transmisor son:

- Precisión:  $\pm 0,055\%$
- Fiabilidad:  $\pm 0,1\%$  (estabilidad por 10 años)

- Tiempo de respuesta: 90mseg.
- Lazo de corriente de 4-20mA
- Se puede configurar en la unidad necesaria, en este caso mbar.

### 4.4.2. Sensor de caudal

Se utilizó un sensor de caudal (Figura 4.6.b) con rotor a paleta de las siguientes características:

- Rango de caudal: 2- 60 l/min
- Máxima presión de agua: 1,75MPa
- Conversión de caudal: aprox 477 pulsos/L ± 10 %



(a) Transmisor de presión      (b) Sensor de caudal

Figura 4.6: Transmisores

### 4.5. Diagrama

La figura 4.7 corresponde al diagrama P&ID realizado en base al banco de pruebas donde se observa los respectivos elementos, sus nombres y las conexiones eléctricas, hidráulicas y mecánicas. En el Anexo A se encuentra la imagen con mayor tamaño. Para la nomenclatura de los instrumentos utilizados se empleó las normas ISA-S5.1 [2].

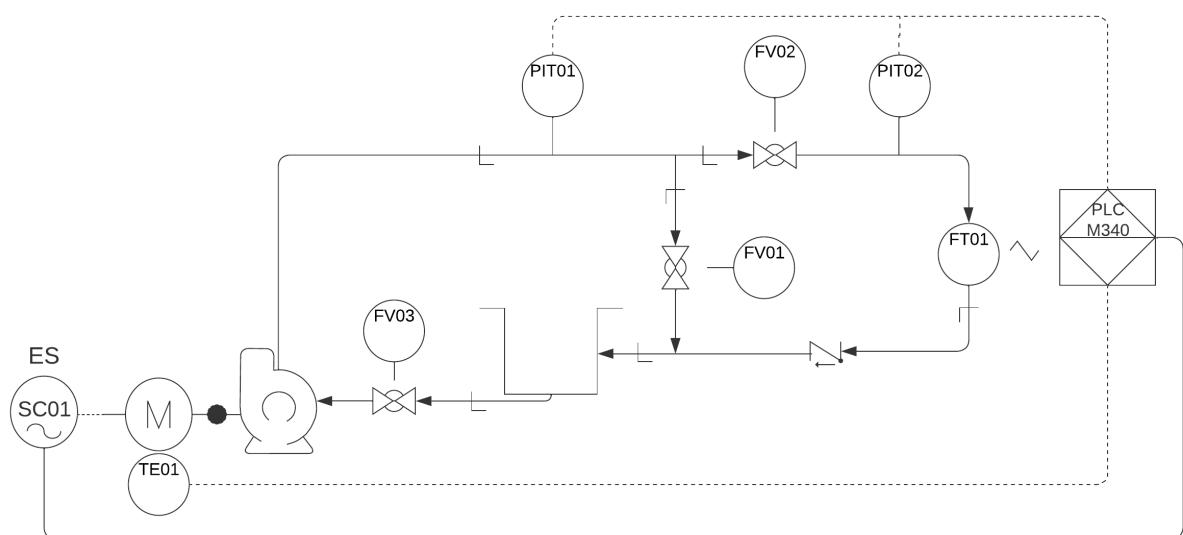


Figura 4.7: Diagrama p&id

## **5. Preliminares**

### **5.1. Programación variador de velocidad**

Para realizar la configuración del variador de velocidad con los parámetros del motor se utilizó el software *SoMove* a través del protocolo *ModBus*. Se descargó la ultima versión desde la página oficial de *Schneider Electric*<sup>1</sup> y luego, la librería DTM correspondiente al variador a utilizado<sup>2</sup>.

Una vez instalado se procedió a generar un nuevo proyecto donde se eligió las características del variador (Figura 5.1 y 5.2). El próximo paso fue realizar por medio del software la carga de los parámetros del motor (Figura 5.3), establecer el modo de funcionamiento de las entradas y configurar el protocolo de comunicación.

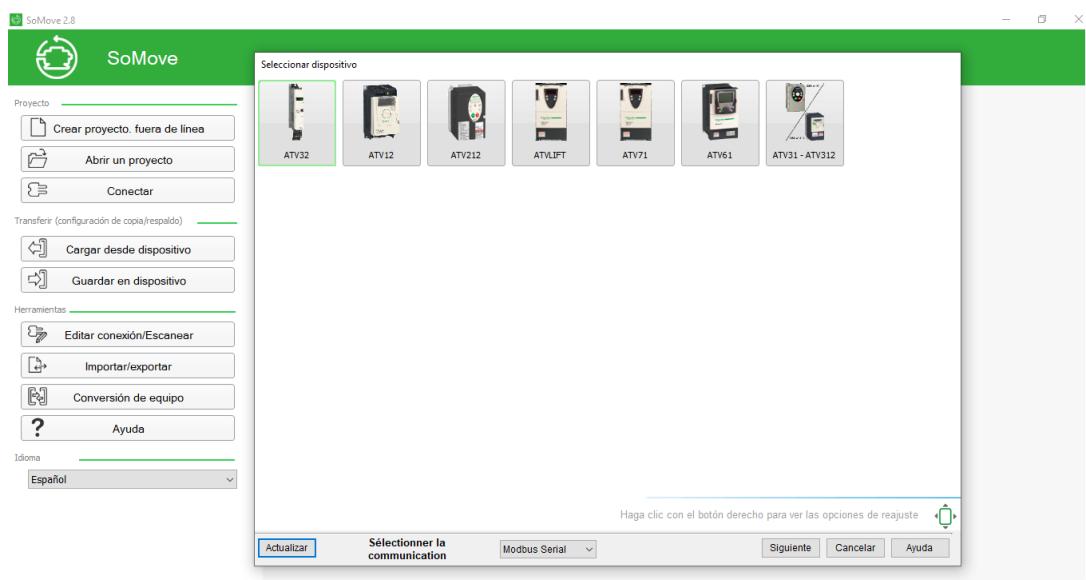


Figura 5.1: Elección de Altivar 312

Para realizar esta primera configuración se comunicó la computadora con el variador a través del protocolo *Modbus RTU* (Figura 5.4) por medio de un cable RJ45 / par trenzado y un conversor RS485 / USB (Figura 5.5).

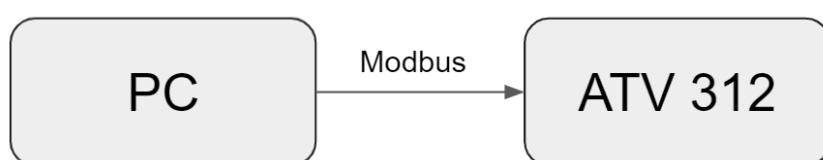


Figura 5.4: Diagrama de comunicación PC- variador

### **5.2. Comunicación variador de velocidad - PLC**

Para poder realizar la comunicación entre el variador y el PLC fue necesario contar con un cable RJ45 (variador) a DB9 (PLC) a través del protocolo CANopen (Figura 5.6),

<sup>1</sup><https://www.se.com/ar/es/product-range-presentation/2714-somove/>

<sup>2</sup>[https://www.se.com/ar/es/download/document/Altivar\\_DTM\\_Library/](https://www.se.com/ar/es/download/document/Altivar_DTM_Library/)

## Automatización Industrial - Trabajo Final

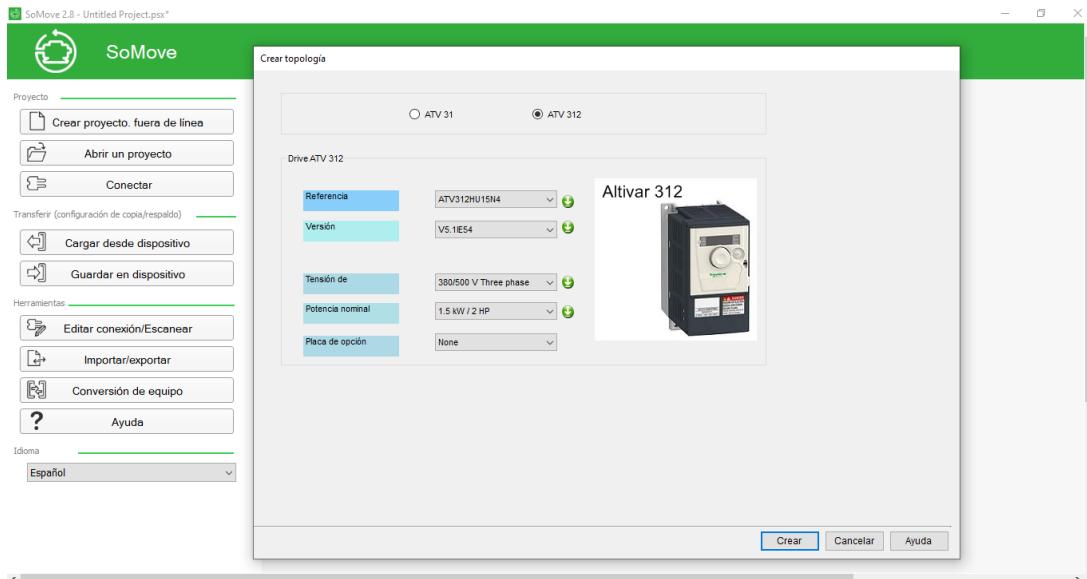


Figura 5.2: Parámetros del variador

Code	Long Label	Current Value	Default Value
UNS	Tensión nominal motor	380 V	400 V
NCR	Intensidad Nom Motor	3.4 A	3.5 A
COS	Motor 1 cos fi	0.84	0.79
ACC	Rampa aceleración (s)	10 s	3 s
DEC	Rampa deceleración (s)	10 s	3 s
NSP	Velocidad nom motor	2840 rpm	1420 rpm
BRL	Apertura de freno (Hz)	2.6 Hz	2139.8 Hz
LAC	Nivel acceso funciones	Nivel 3	Nivel 1
FR2	Canal Referencia 2	Modbus	No
RFC	Asig.conmut.ref.(1a 2)	LI4	Canal1 act.
CHCF	Config. modo control	Separados	No separad.
CCS	Comutación canal ctrl	LI4	Canal1 act.
ROT	Sent marcha autorizado	Ambos	Avance
DO	Salida Analog/logica	Fallo equipo	No
R1	Asignación del relé R1	Var.marcha	Sin fallo
FST	asignación stop rápida	LI5	No
PS2	2 velocidad preselecc	No	LI3
PS4	4 velocidad preselecc.	No	LI4
LET	Config. fallo externo	Activo a 0	Activo a 1
EPL	Gestión fallo externo	Paro rampa	Rueda libre

Figura 5.3: Lista de parámetros modificados

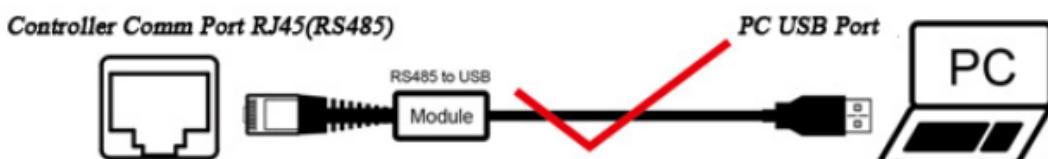


Figura 5.5: Cable de comunicación

y se colocó en los finales de línea una resistencia de  $120 \Omega$  para evitar ruidos eléctricos

y fenómenos de reflexión en la línea.

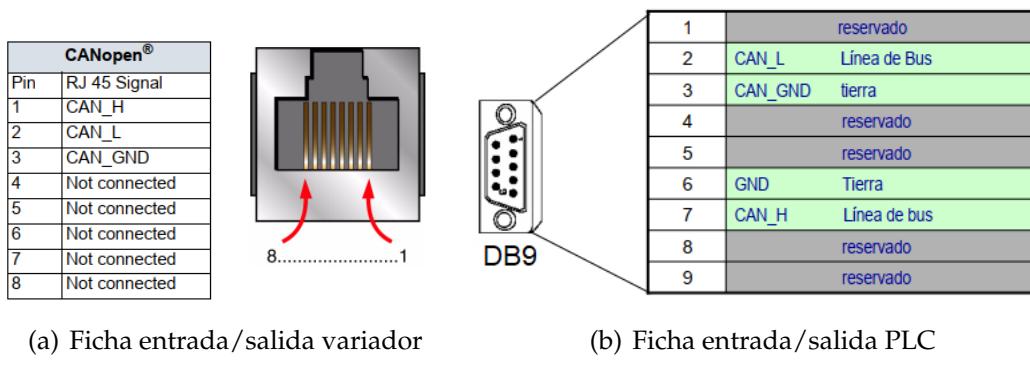


Figura 5.6: Conexión fichas RJ45- DB9

### 5.3. Programación Unity Pro

Para generar la base del proyecto de trabajo, se descargó e instaló el software *Unity Pro XL* y la librería DTM utilizada en el software *soMove* correspondiente al variador que se posee. Una vez que fue instalado se creó y configuró un nuevo proyecto a través de los siguientes pasos.

1. Se seleccionó el bastidor (Figura 5.8).
  2. En la configuración gráfica del bastidor se introdujo los módulos deseados (Figura 5.7) correspondientes al PLC didáctico del laboratorio (Sección 4.3).
  3. Se configuró el módulo Ethernet, desde el explorador de proyectos se desplegó la carpeta *Comunicación* y se creó una nueva red, Ethernet (Figura 5.9).
  4. Se añadió en el bus CANopen el variador utilizado (Figura 5.10). En nuestro caso se eligió el ATV31 para usar los bloques de funciones de control de movimiento preestablecidos por el software. Dentro de las configuraciones de este protocolo se configuró la velocidad de transmisión de este dispositivo. 5.11
  5. Se creó una nueva sección de lenguaje FDB para visualizar los parámetros básicos.
    - Los *Diagramas de Bloques de Función* consisten en un editor gráfico orientado al dibujo de bloques. Este se basa en la utilización de funciones reusables elementales y derivados.

## Automatización Industrial - Trabajo Final

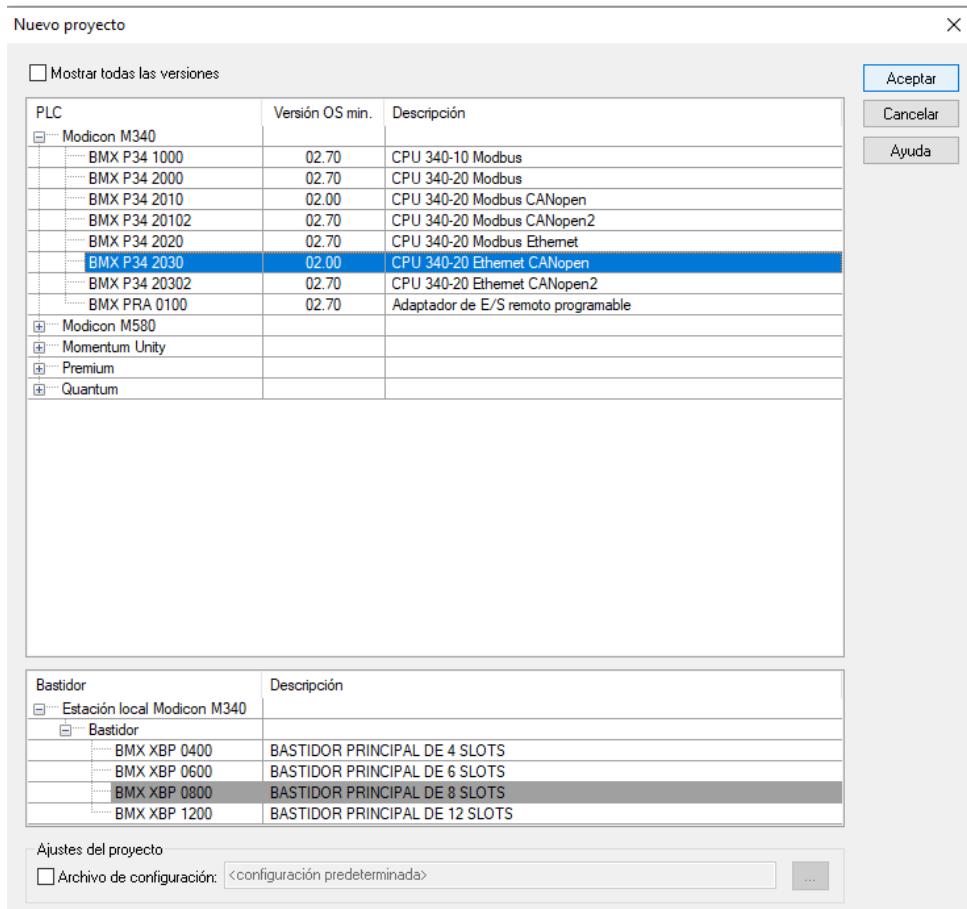


Figura 5.7: Elección del bastidor

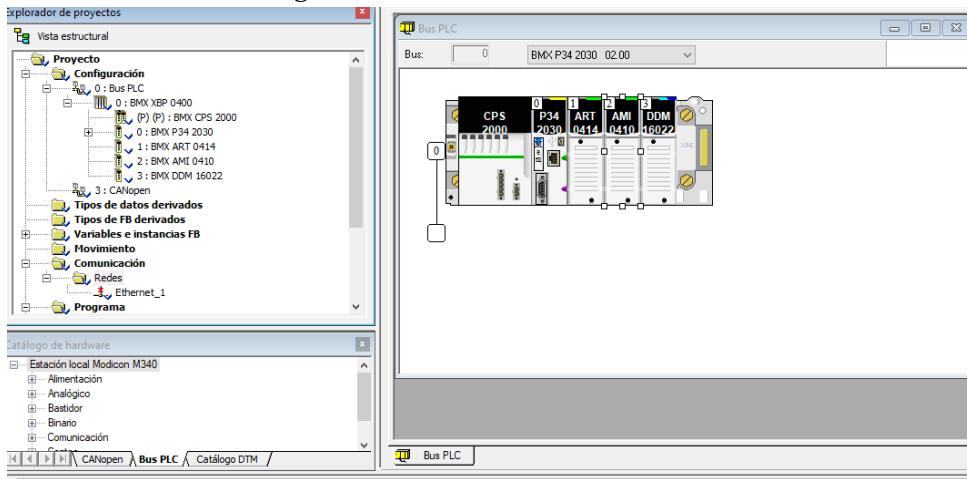


Figura 5.8: Módulos PLC

Una vez que se completó la configuración de la comunicación variador - PLC se procedió a crear un HMI simple en *iFix* (Figura 5.12), el cual fue utilizado para interactuar y observar diversos parámetros, modificar velocidades, observar señales luminosas y ver distintos valores proporcionados por el variador de velocidad.

Para observar en el HMI se utilizó los MFB del software *UnityPro* (Figura 5.13), los cuales necesitan de un bloque maestro “CAN\_HANDLER” el cual permite comprobar la comunicación CANopen.

Otros de los bloques más utilizados dentro del programa fue “MC\_READPARAMETER” que se utiliza para leer, mediante mensajes SDO, una

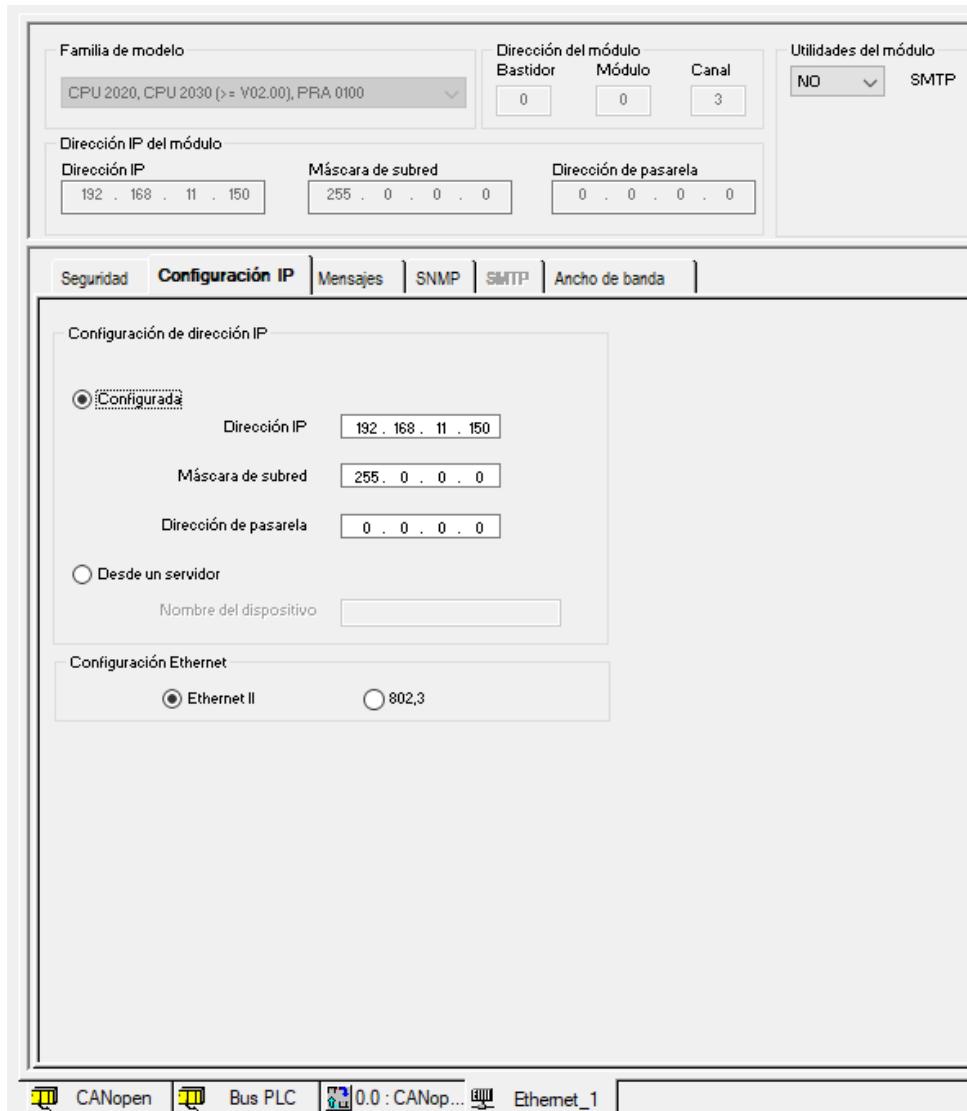


Figura 5.9: Dirección módulo Ethernet

variable del variador definida en una dirección *CANopen* dada por el fabricante [3].

### 5.3.1. Entradas analógicas

En el rack del PLC se encuentra un módulo AMI0410 el cual consiste en cuatro entradas analógicas (Tensión/Corriente) aisladas del siguiente tipo:

- Corriente +/- 20 mA
- Corriente 0 a 20 mA
- Corriente 4 a 20 mA
- Tensión +/- 10 V
- Tensión +/- 5 V
- Tensión 0 a 10 V
- Tensión 0 a 5 V
- Tensión 1 a 5 V

El módulo dispone de 20 bornes accesibles al usuario donde el diagrama de conexión tanto para entradas de tensión o de corriente es el mostrado en la figura 5.14.

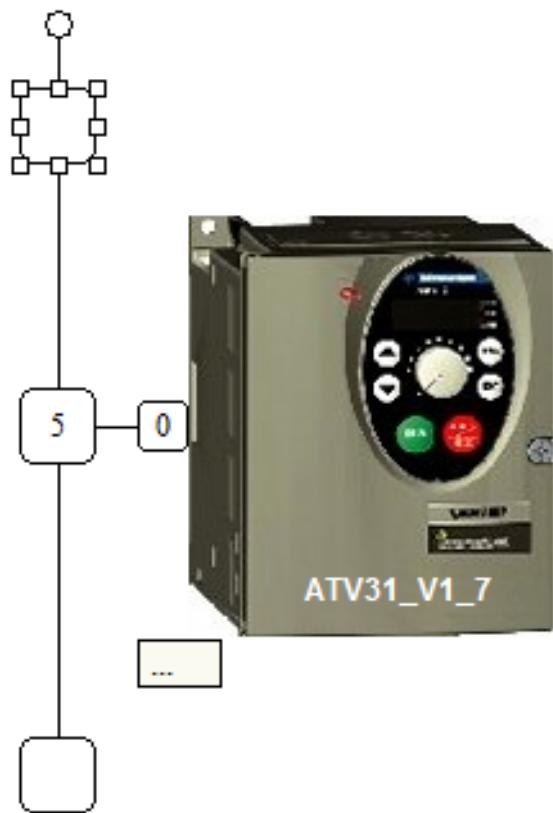


Figura 5.10: Configuración variador CANopen

Para realizar la configuración de los transmisores de presión en *Unity PRO* se elige, de las opciones nombradas anteriormente, la que se utilizó en este caso de 4 a 20 mA y el escalado predefinido de fábrica de 0 a 10000 cuentas, pero modificable por el usuario entre -32768 a 32767 ya que la resolución de las entradas analógicas es de 16 bits. También se agregó un filtro de primer orden a cada entrada por software (Figura 5.15).

Con el rango de cuentas del módulo determinado, se requirió escalar la variable para obtener el valor físico y así utilizarla para la visualización y control del banco de pruebas.

Para el escalado de la señal de entrada se realizó un nuevo bloque FB derivado, a partir elementos primarios, para evitar realizar un proceso repetitivo en el escalado de varias señales (Figura 5.16).

En la variable de entrada se coloca la señal que se desea escalar, en los límites de entrada se escribe el rango del módulo visto anteriormente, y en los límites de salida, los escalados obtenidos del instrumento que se utilizó.

Finalmente, a la salida del bloque se obtuvo el valor físico visualizado o transmitido por el instrumento.

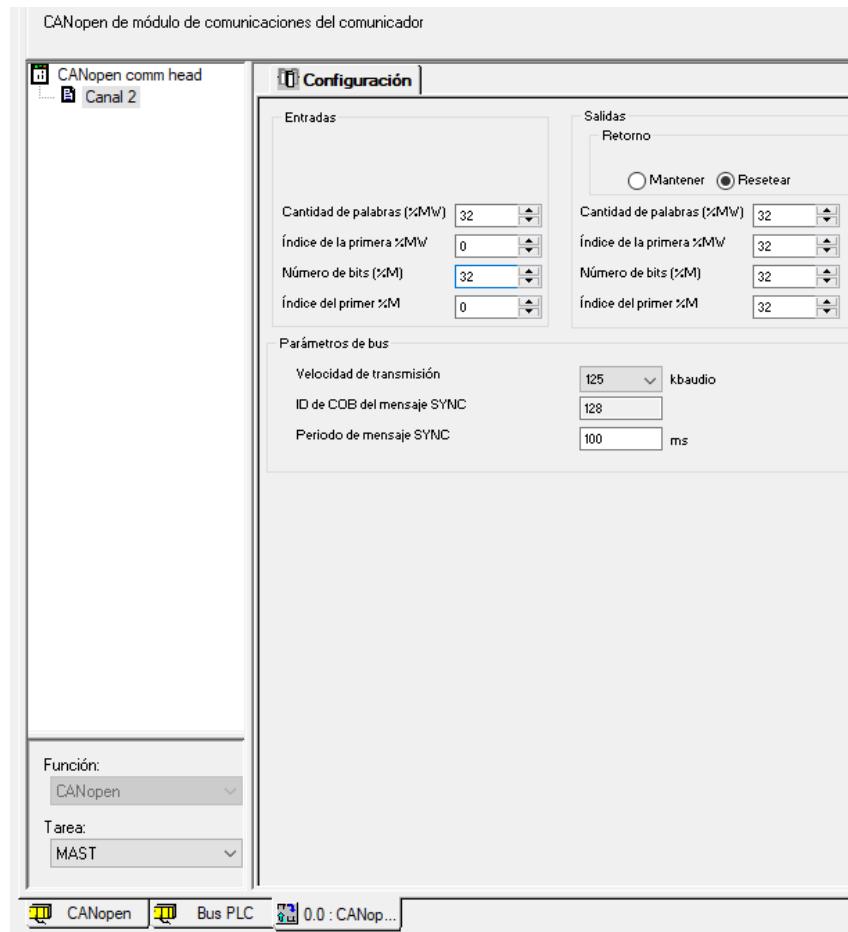


Figura 5.11: Configuración variador CANopen



Figura 5.12: HMI simple

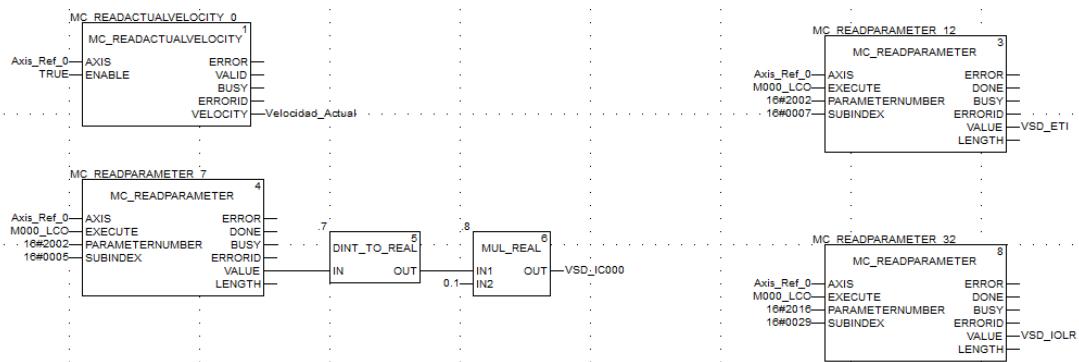


Figura 5.13: Programa con bloques MFB

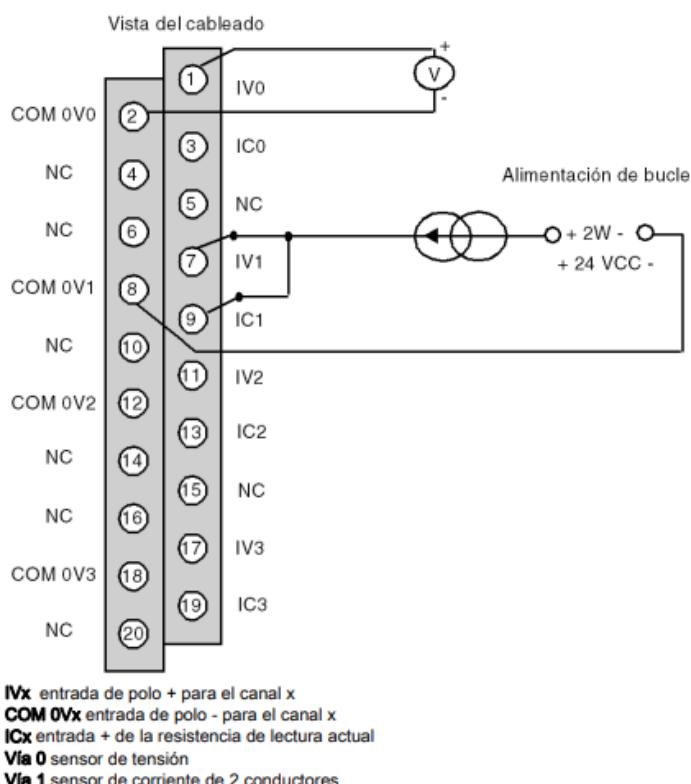
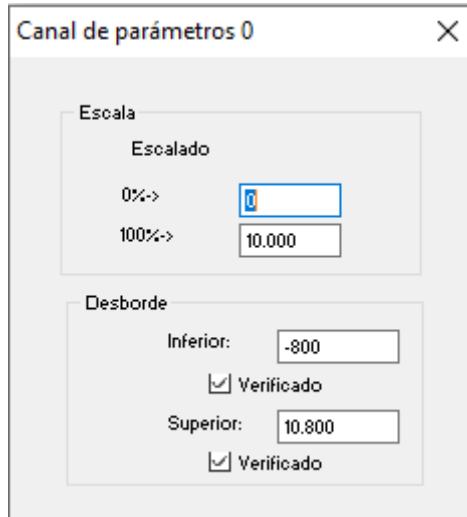


Figura 5.14: Módulo AMI0410

**Configuración**

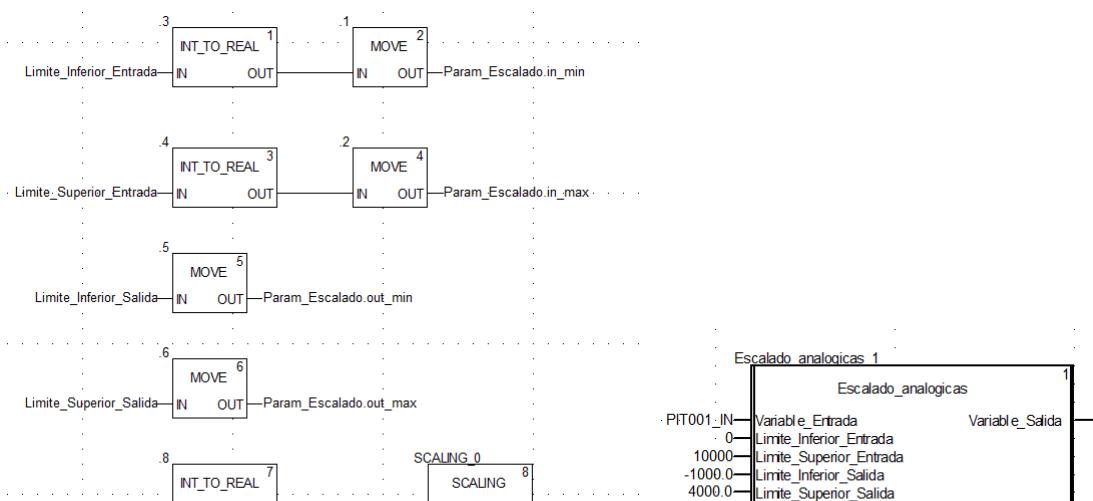
	Utilizado	Símbolo	Rango	Escala	Filtro
0	<input checked="" type="checkbox"/>	PIT001_IN	De 1 a 5 V / De 4 a 20 mA	<input checked="" type="checkbox"/> %..	5
1	<input checked="" type="checkbox"/>	PIT002_IN	De 1 a 5 V / De 4 a 20 mA	<input checked="" type="checkbox"/> %..	4
2	<input type="checkbox"/>		De 1 a 5 V / De 4 a 20 mA	<input checked="" type="checkbox"/> %..	0
3	<input type="checkbox"/>		De 1 a 5 V / De 4 a 20 mA	<input checked="" type="checkbox"/> %..	0

(a)



(b)

Figura 5.15: Rango y escalado



(a)

(b)

Figura 5.16: Bloques de escalado

### **5.3.2. Entradas analógicas para termocuplas o RTD**

Se utilizó el módulo ART0414 que consiste en cuatro entradas aisladas en las que se puede conectar sensores de temperatura del tipo termocupla y RTD con las siguientes características:

- RTD IEC Pt100/Pt1000, US/JIS Pt100/Pt1000, Cu10, Cu50, Cu100, Ni100/Ni1000 en 2, 3 o 4 conductores
- Termoelemento del tipo B, E, J, K, L, N, R, S, T, U
- Tensión  $+/- 40 \text{ mV}$  a  $1,28 \text{ V}$

Para la conexión de estos sensores se utiliza el accesorio TELEFAST ABE-7CPA412 (Figura 5.17) y según el tipo de sensor se tiene las conexiones que se muestran en la figura 5.18.

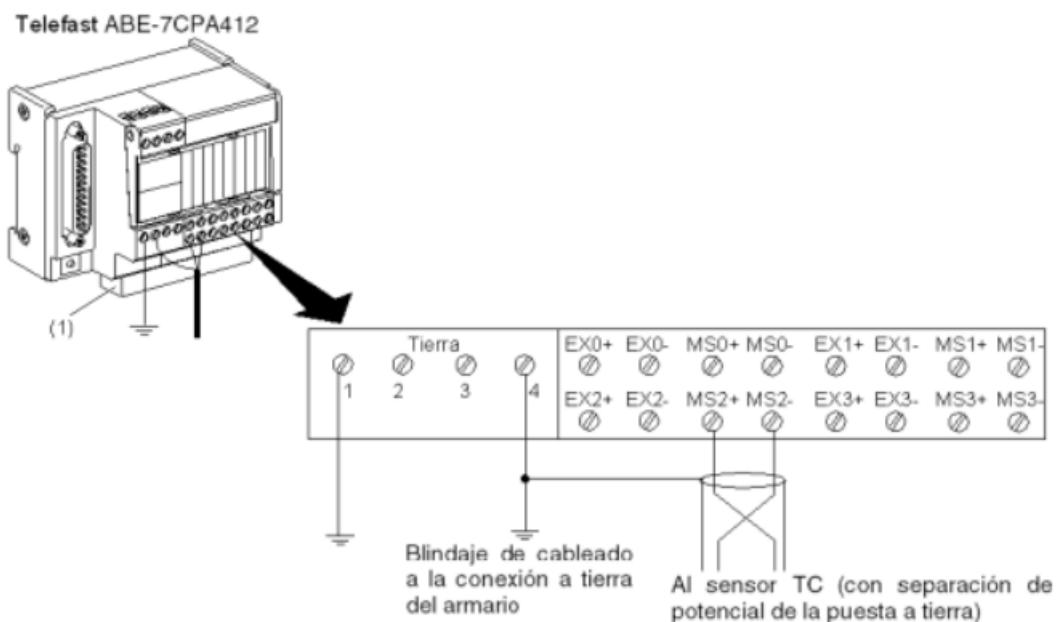
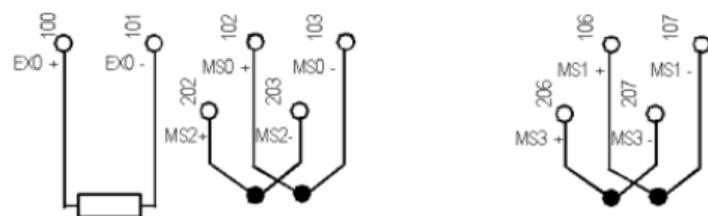


Figura 5.17: Accesorio TELEFAST ABE-7CPA412



**Leyenda:** Funcionamiento en modalidad TC con compensación de unión en frío y una sonda PT100 de 2 conductores.

Figura 5.18: Tipos de conexión

Para configurar en *Unity Pro* se seleccionó en *rango* la característica resistiva del sensor. En el caso de RTD y termocupla los valores de salida son múltiplos de 10 de la temperatura.

En la imagen 5.19 se observa dos elementos ya que se probó realizar la configuración con una termocupla y un RTD. Finalmente se eligió el RTD PT1000 ya que dispone de una superficie plana que mejora el contacto con la carcasa del motor. El valor obtenido a la salida del módulo se divide por 10 y se genera el valor de temperatura en °C con un decimal (Figura 5.19).

	Utilizado	Símbolo	Rango	Escala	Filtro
0	<input checked="" type="checkbox"/>	TT001_IN	Pt1000 IEC 751-1995, JIS C1604-1997 (2/4 conductores)	<input checked="" type="checkbox"/> 1/10 °C	<input checked="" type="checkbox"/> 0 <input checked="" type="checkbox"/>
1	<input checked="" type="checkbox"/>	TT002_IN	Termo K	<input checked="" type="checkbox"/> 1/10 °C	<input checked="" type="checkbox"/> 0 <input checked="" type="checkbox"/>
2	<input type="checkbox"/>		Termo K	<input checked="" type="checkbox"/> 1/10 °C	<input checked="" type="checkbox"/> 0 <input checked="" type="checkbox"/>
3	<input type="checkbox"/>		Termo K	<input checked="" type="checkbox"/> 1/10 °C	<input checked="" type="checkbox"/> 0 <input checked="" type="checkbox"/>

Figura 5.19: Rango y escalado

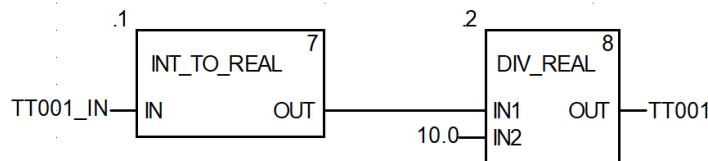


Figura 5.20: Bloque de escalado

### 5.3.3. Medición de caudal

A la salida del caudalímetro se obtiene pulsos con frecuencia proporcional al flujo. Para ser procesados por el PLC se utilizó un módulo externo al PLC por la baja tasa de refresco que posee el módulo de entradas digitales. Para esto se planteó utilizar un ESP8266 como interfaz para obtener los pulsos, colocarlos en un registro y enviarlos por un servidor Modbus TCP cada 20ms al PLC. En la sección de programación del PLC se realizaron las cuentas correspondientes para generar la conversión de pulsos a caudal (Figura 5.21). El módulo y el caudalímetro alimentó mediante una fuente externa de 3,3 V.

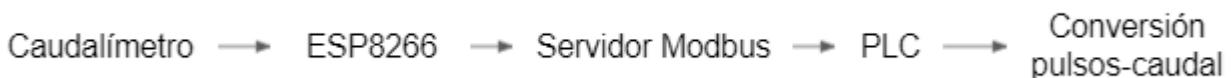


Figura 5.21: Diagrama de flujo del caudalímetro

## **6. Desarrollo**

A partir de lo programado en *Unity PRO* utilizado para obtener los primeros registros, se realizaron las modificaciones necesarias donde se eliminaron y agregaron variables de la lista de direcciones (Tabla B.1 en Anexo B). El mapa de memoria del proyecto se divide según la figura 6.1.

Valores Digitales	Device0:000003 ... Device0:000010 Device0:000011 ... Device0:000017
Valores Analógicos	Device0:400001 ... Device0:400083
Registros	Device0:400101:4 ... Device0:400101:6

Figura 6.1: Mapa de memoria

### **6.1. Adquisición de datos**

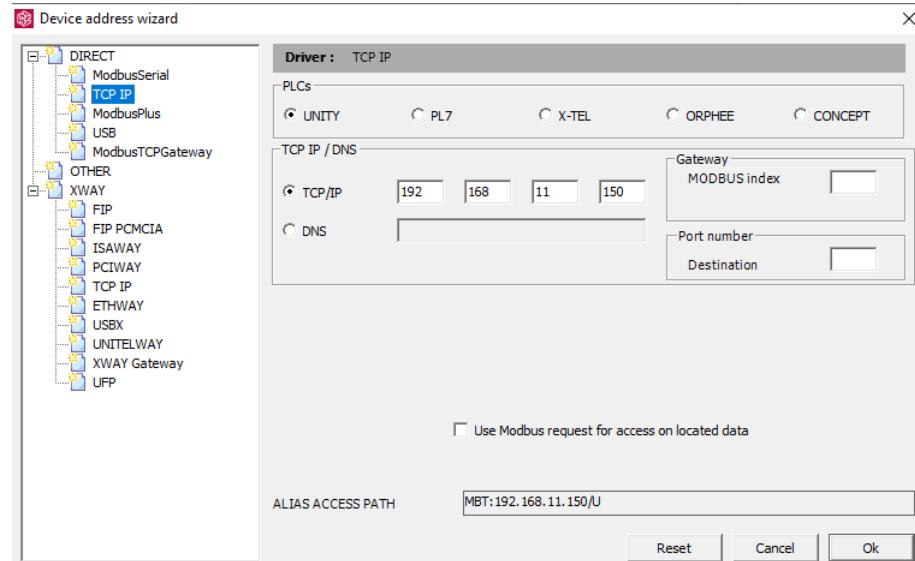
En el objetivo se propuso que el sistema sea capaz de controlar presión o caudal. Para lograr lo estipulado, como en cualquier sistema de control, es necesario conocer las plantas con las que se trabajará, dónde se distinguen en el banco de pruebas tres sistemas distintos:

- PIT01: Presión medida a la salida de la bomba.
- PIT02: Presión medida luego de la columna de derivación.
- FT01: Caudal que pasa por PIT02.

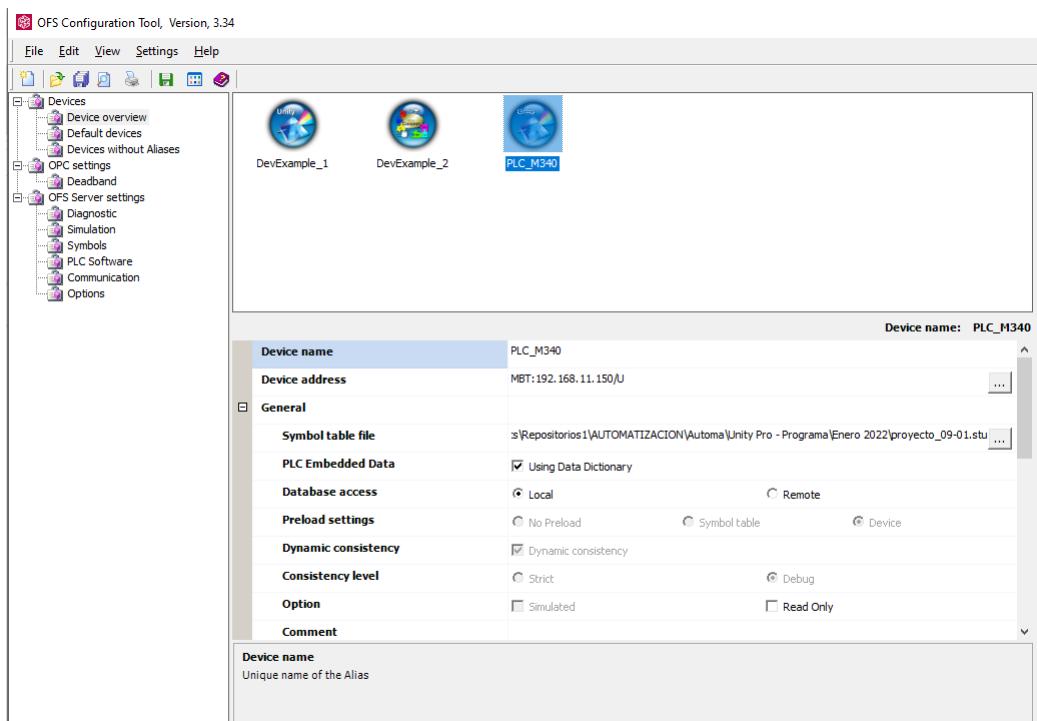
Para realizar las estimaciones de los sistemas se utilizó el protocolo OPC en conjunto con *Matlab*. Por medio de OFS, se procedió a crear y configurar un servidor(Figura 6.2).

Una vez configurado, se abre el programa *OPC Factory Server* y se da inicio al servidor (Figura 6.3.a). Para observar si la comunicación se estableció de forma correcta, se utilizó el programa *OFS Client* (Figura 6.3.b).

Una vez corroborada la comunicación con el servidor OPC, se procedió a crear un cliente OPC en *Simulink* (perteneciente a *Matlab*) para adquirir y guardar las variables necesarias.



(a)



(b)

Figura 6.2: Configuración OFS

### 6.1.1. Uso de Matlab

En el entorno *Simulink* se procedió a configurar un bloque de cliente OPC con la dirección IP donde se encuentra el servidor previamente creado. Luego, para leer las variables necesarias se creó un bloque de lectura OPC (Figura 6.4.a) y con un bloque *Scope*, se activó la opción para que se guarden los vectores de las variables a estudiar (Figura 6.4.b) para luego ser guardados en un archivo .csv .

Se configuró el bloque OPC Read (Figura 6.5) con una frecuencia de muestreo de 50 ms ya que el valor es 10 veces más rápido que el polo del sistema más rápido.

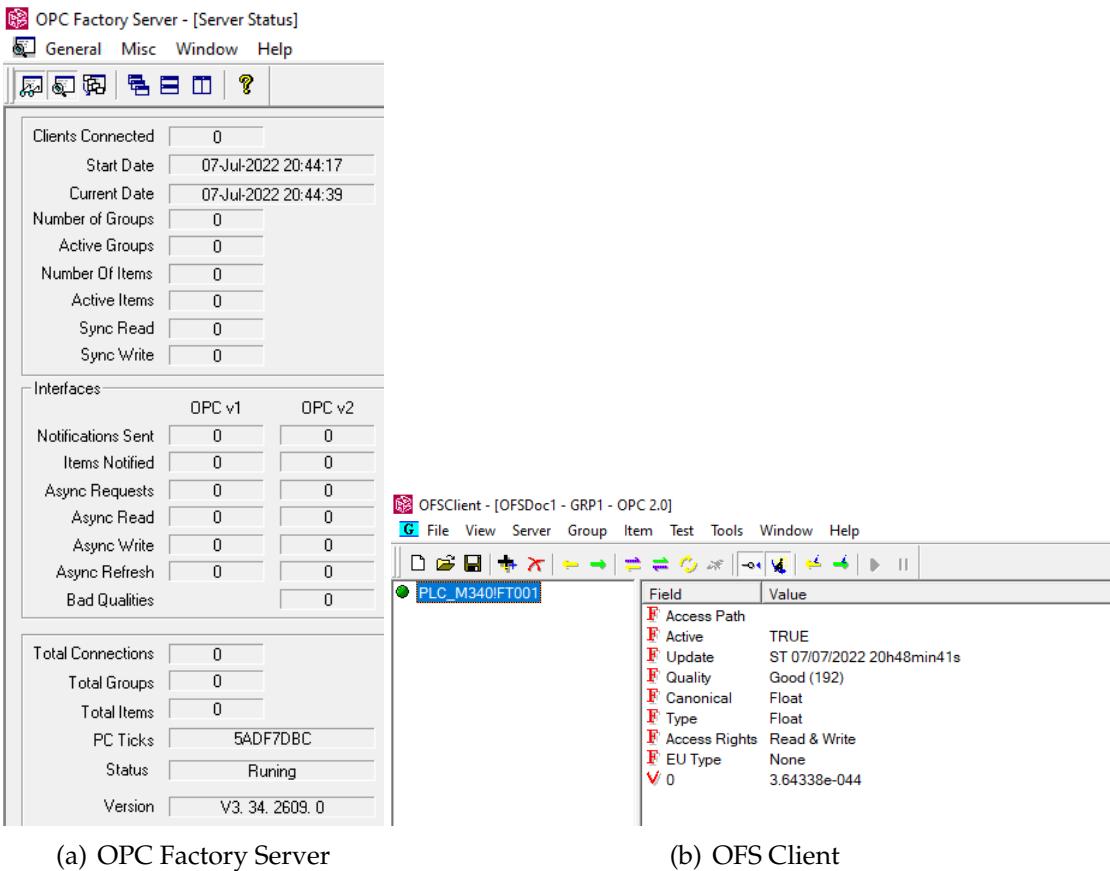


Figura 6.3: Conexión servidor OPC

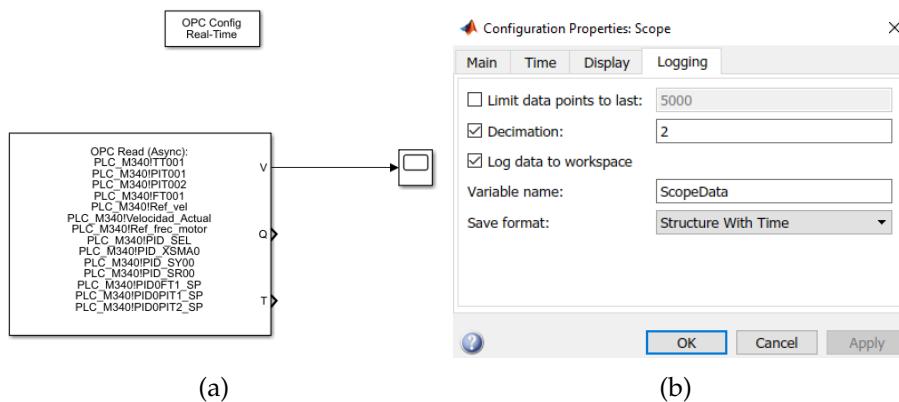


Figura 6.4: Cliente OPC en Simulink

## 6.1.2. Estimación de la planta

Para realizar la estimación de las plantas se utilizó el Método de Strejc con retardo (Figura 6.6)[4]. Este método se emplea para la identificación de sistemas de polos múltiples, mediante los parámetros  $T_u$  y  $T_a$  obtenidos sobre la respuesta del sistema. Tras obtener el valor de los parámetros, se determina la multiplicidad del polo.

La función de transferencia general para un sistema de polos múltiples es:

$$G(s) = \frac{K}{(1 + \tau \cdot s)^n} \cdot e^{-T_s} \quad (1)$$

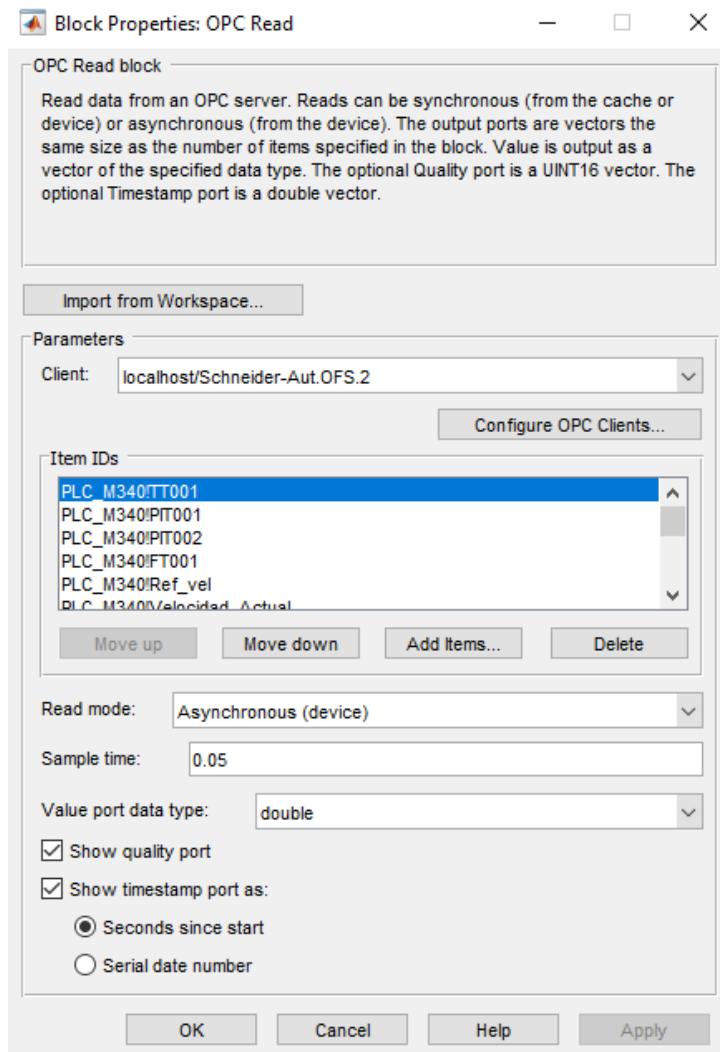


Figura 6.5: Configuración de frecuencia de muestreo

Dónde:

- K: Ganancia del sistema  $K = \frac{\Delta y}{\Delta u}$
- $\tau$ : constante de tiempo
- T= Retardo

Para obtener la función de transferencia se realizó un programa en *Matlab* que dió como resultado los siguientes sistemas:

- Planta de presión PIT01:

$$G(s) = \frac{0,135}{(1 + 1,3783 \cdot s)^3} \cdot e^{-1,2 \cdot s} \quad (2)$$

- Planta de presión PIT02:

$$G(s) = \frac{0,125}{(1 + 1,054 \cdot s)^3} \cdot e^{-s} \quad (3)$$

- Planta de caudal FT01:

$$G(s) = \frac{0,003784}{(1 + 0,7027 \cdot s)^3} \cdot e^{-s} \quad (4)$$

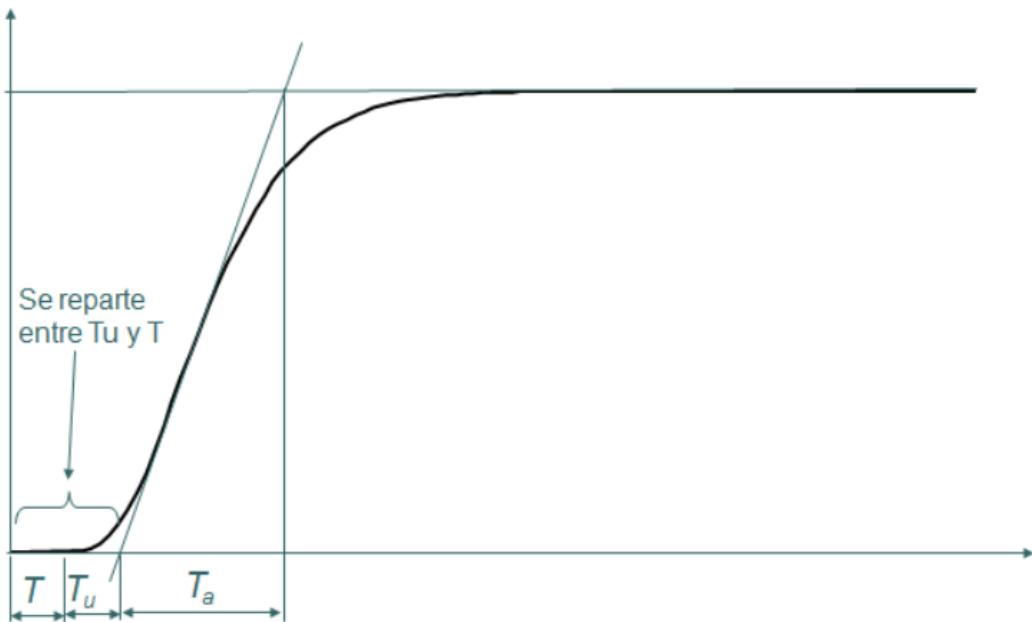


Figura 6.6: Parámetros de Strejc con retardo

#### 6.1.2.1. Comparación numérica- real

En las siguientes imágenes (Figura 6.8 ,6.9 y 6.7) se observa la gráfica de cada planta estimada comparada con los datos obtenidos en las mediciones.

Cabe destacar que las plantas fueron calculadas para los rangos medios que normalmente se utilizará dado que los sistemas de presión no presentan una ganancia estática constante.

Se puede observar que los sistemas de tercer orden se adaptan bien a los datos obtenidos durante las pruebas.

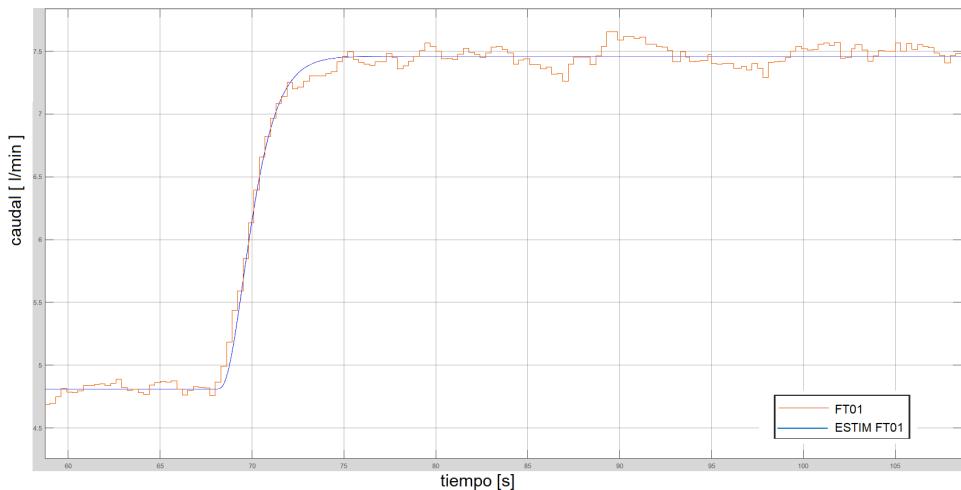


Figura 6.7: Comparación de la planta estimada y los valores obtenidos para FT01

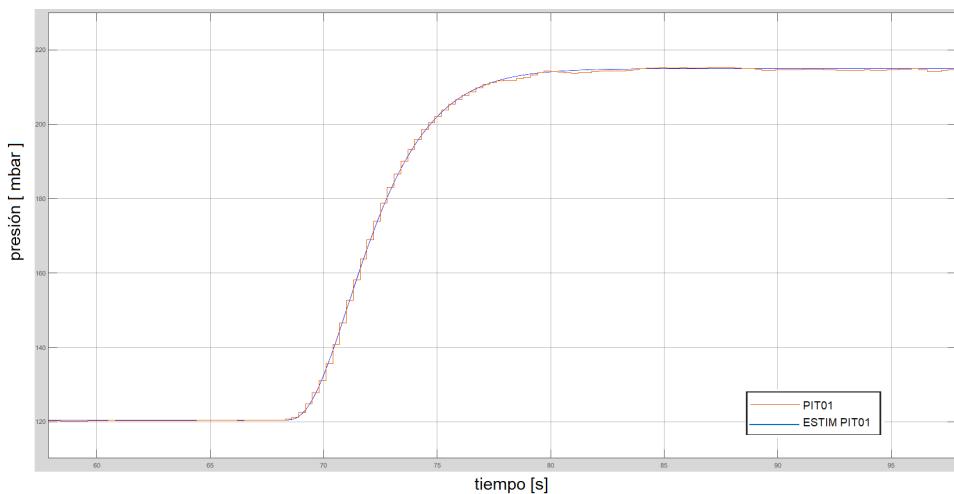


Figura 6.8: Comparación de la planta estimada y los valores obtenidos para PIT01

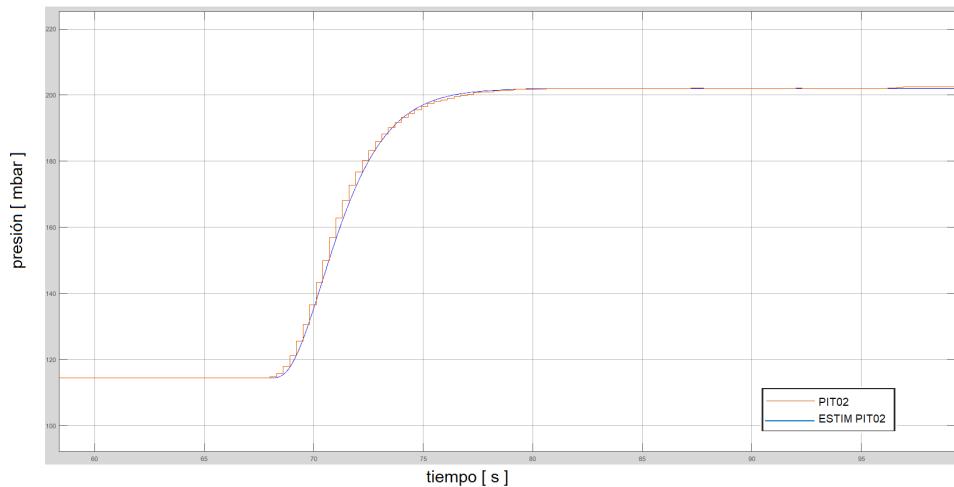


Figura 6.9: Comparación de la planta estimada y los valores obtenidos para PIT02

### **6.1.3. Cálculo del controlador PID**

El controlador PID de cada planta se calculó con *Tune PID controllers* (Figura 6.10), dónde se buscó que la respuesta al escalón en estado estacionario sea nula y además sea capaz de mitigar los cambios producidos por la ganancia variable en los sistemas de presión ocasionados por la característica de la bomba.

Se realizaron pruebas con diversos PID para observar la respuesta a cada sistema, en las figuras 6.11 , 6.12 y 6.13 se comparan dos PI para cada planta, para elegir los parámetros del controlador se buscó que las respuestas lleguen al valor de referencia de una forma suave y con el mínimo sobrepaso.

Los valores obtenidos de la aplicación perteneciente a *Matlab* se muestran en la tabla 6.1, los cuales se ingresaron en los bloques de *UnityPro* con las respectivas modificaciones numéricas según lo establecido por el software. Los valores mostrados en dicha tabla serán usados al momento de restablecer la configuración de los PI en la ventana de configuración del sistema SCADA.

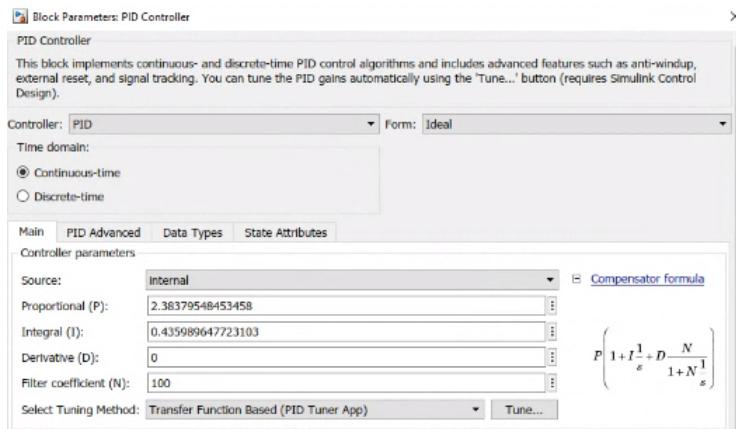


Figura 6.10: PID Controller

	PIT01	PIT02	FT01
$K_p$	3,05	5,453	104,3
$K_i$	0,396	0,348	0,719
$K_d$	0	0	0
N	1000	1000	1000

Tabla 6.1: Valores de los PID

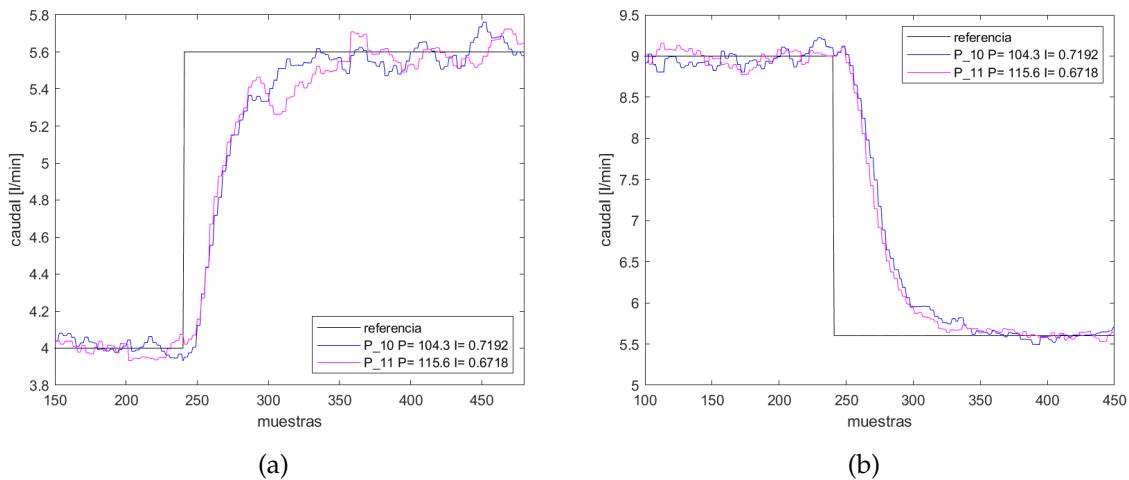


Figura 6.11: Comparación PI para FT01

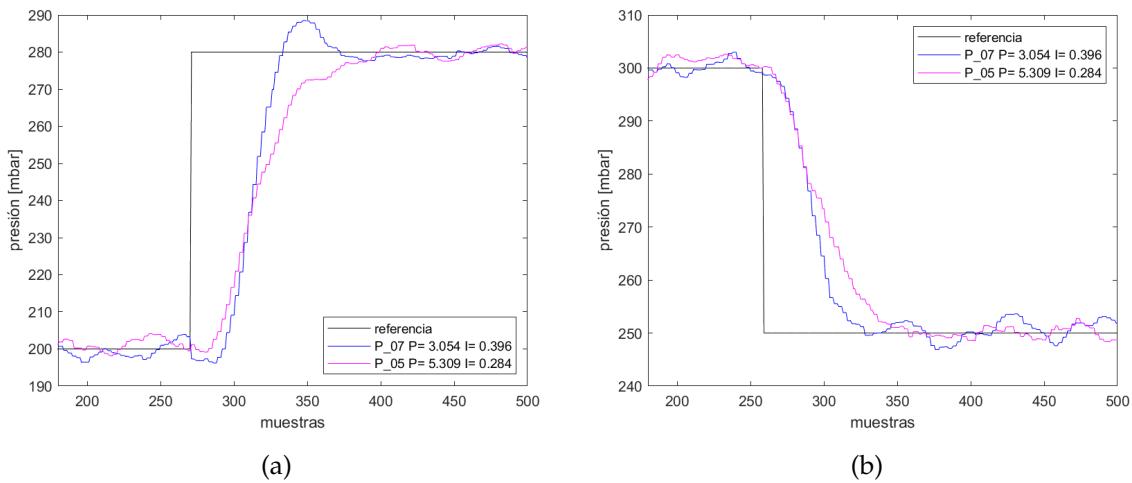


Figura 6.12: Comparación PI para PIT01

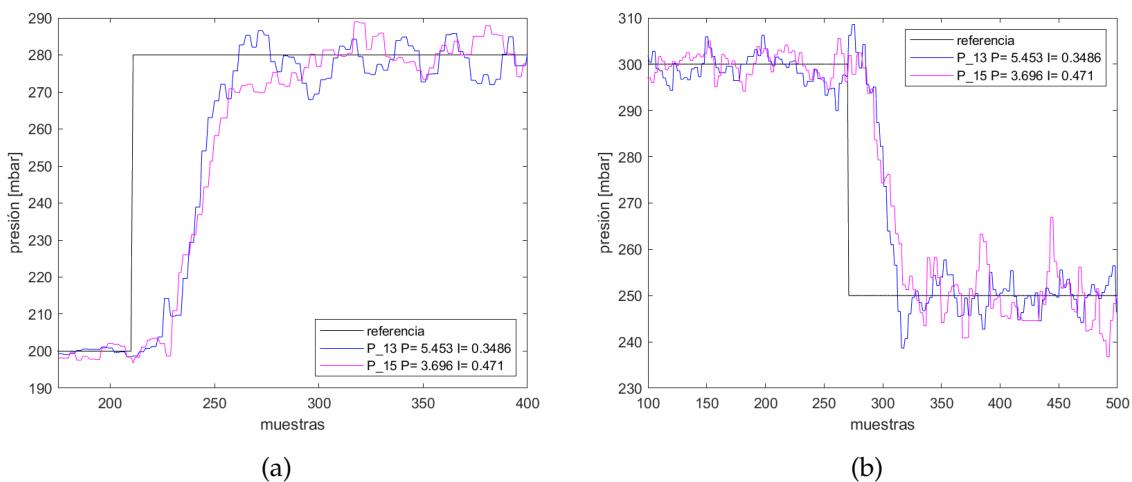


Figura 6.13: Comparación PI para PIT02

#### **6.1.4. Pruebas de control**

Para corroborar que cada sistema funcione correctamente ante perturbaciones se abrió y cerró las válvulas del banco de pruebas mostradas en el diagrama P&ID de la figura 6.14 que causan distintas alteraciones a la presión y caudal.

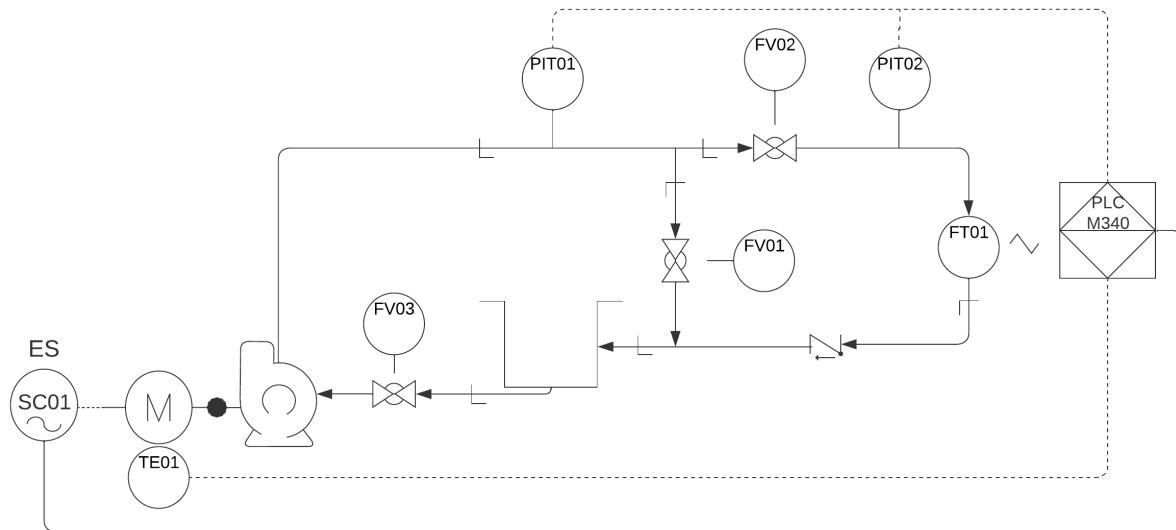


Figura 6.14: Diagrama p&id

##### **6.1.4.1. Perturbación al control con PIT02**

En la figura 6.15 se observa que el valor de referencia inicial es de 250 mbar; aproximadamente a los 15 segundos de comenzar la prueba se abrió totalmente la válvula de derivación FV01, y disminuyó la presión notablemente. Al pasar el tiempo y una vez que la presión llegó nuevamente a su punto de trabajo, se incrementó su valor a 300 mbar, se puede apreciar en el gráfico que la presión se acerca pero no llega a la referencia debido a que el motor, ya en su máxima frecuencia de trabajo, no logra elevar más la presión. Luego, a los 100 segundos de empezar la prueba, se cerró la válvula de derivación completamente y se observa como la presión se incrementó y disminuyó hasta que llegó al punto de referencia.

##### **6.1.4.2. Perturbación al control de FT01**

Para la prueba de control con FT01 (Figura 6.16) como variable de proceso, se fijó el valor de referencia en 3,5 L/min a los 20 segundos se empezar la prueba. Luego, se abrió la válvula FV01 completamente y se observó la disminución del caudal, la acción de control elevó las revoluciones del motor hasta llegar a su punto de trabajo. Aproximadamente a los 125 segundos se configuró el valor de referencia en 2 L/min y se cerró la válvula FV01, esto provocó que todo el caudal circule por FT01. La acción de control disminuyó las revoluciones del motor hasta llegar al valor de caudal de referencia.

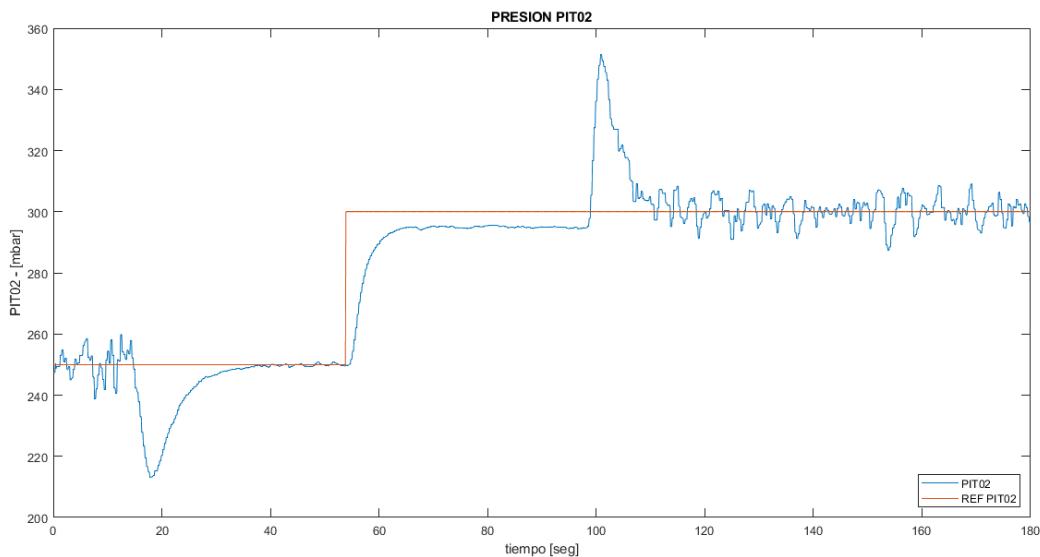


Figura 6.15: Perturbación al control con PIT02

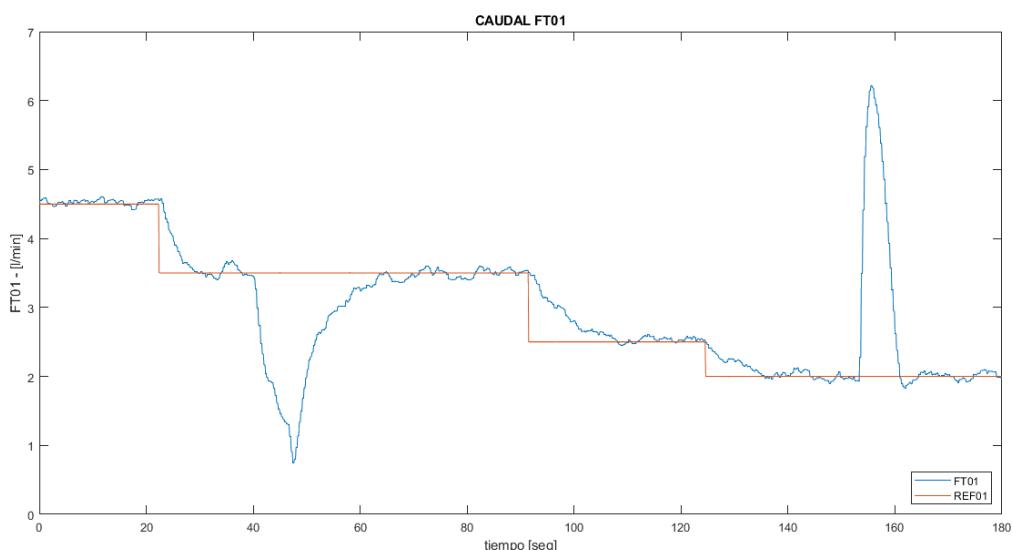


Figura 6.16: Perturbación al control con FT01

### **6.1.5. Valores extremos/críticos de estudio**

En estas sub-secciones se muestra valores máximos de presión y caudal alcanzados en el banco de pruebas y el valor mínimo de presión a velocidad baja del motor.

#### **6.1.5.1. Máxima presión**

Para esta prueba se llevó al motor a su máxima frecuencia (60Hz) de trabajo, se dejó la válvula FV03 totalmente abierta y se cerraron las válvulas FV01 y FV02 (Figura 4.7). Esto provocó que la bomba llegue a su máxima presión que se obtiene al momento en que el caudal que circula por PIT01 se hace muy próximo a cero.

**Máxima presión:** 938 mbar.

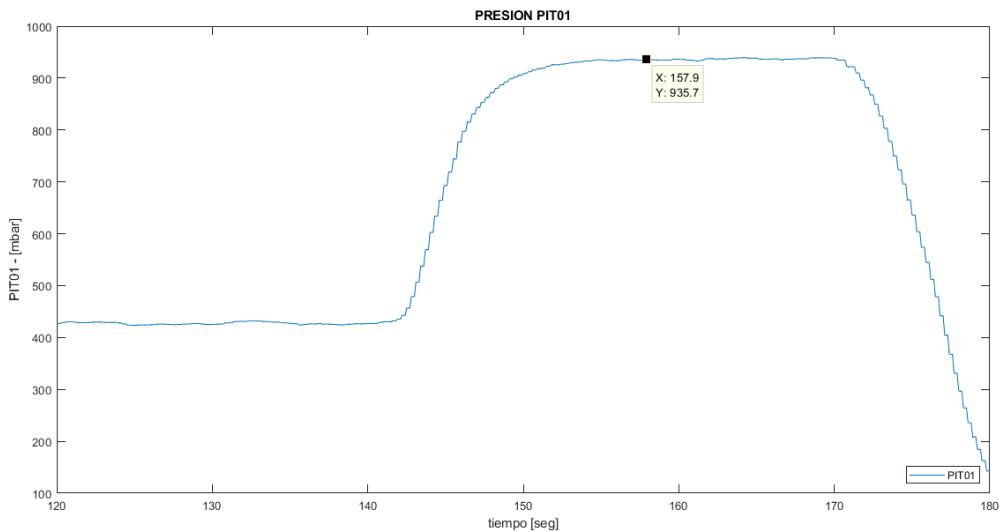


Figura 6.17: Máxima presión

### 6.1.5.2. Mínima presión

Para obtener el valor de mínima presión de trabajo se configuró el VSD a 20 Hz y se abrieron todas las válvulas (Figura 4.7), ésto generó que la bomba impulse el caudal con el menor esfuerzo.

**Mínima presión:** 65,43 mbar

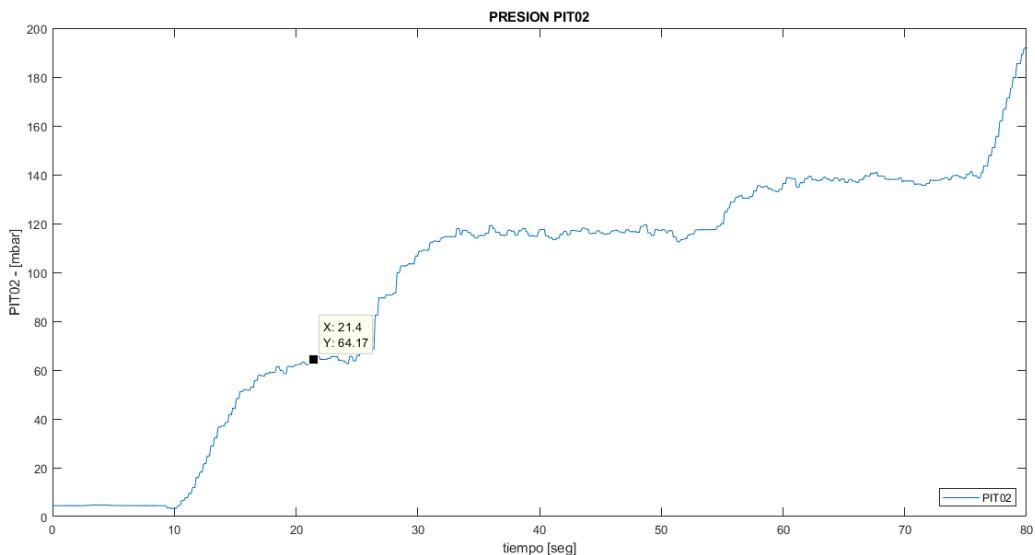


Figura 6.18: Mínima presión

### 6.1.5.3. Máximo caudal

Para obtener el valor de caudal máximo se llevó al motor a su máxima frecuencia (60 Hz), se abrió completamente la válvula FV03 y FV02, y se cerró la válvula de derivación FV01 (Figura 4.7). Esto provocó que todo el caudal impulsado por la bomba circule por

el caudalímetro FT01.

**Máximo caudal:** 11,69 l/min

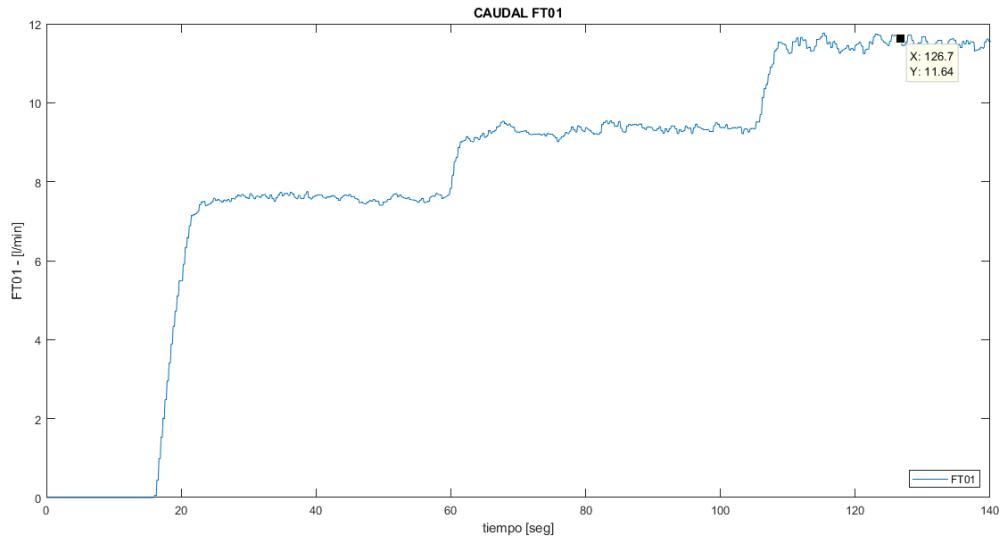


Figura 6.19: Máximo caudal

## **6.2. SCADA**

Para realizar la pantalla de supervisión, control y adquisición de datos operador se utilizó el software iFix.

El sistema SCADA creado (Figura 6.20) se dividió en las siguientes secciones:

- Esquemático del circuito hidráulico físico con las variables de presión y caudal en tiempo real.
- Valores de funcionamiento del motor obtenidos por el variador de velocidad.
- Alarma, dónde se observa de forma visual valores críticos alcanzados en el sistema.
- Indicador de modo de funcionamiento local o remoto.
- Modo de control a lazo abierto o lazo cerrado.

Para el modo de lazo cerrado se creó una ventana individual para cada sistema de presión y caudal.

- Pantalla para observar gráficos en tiempo real donde se divide según la variable a observar, con botones para abrir el control PID del sistema.
- Pantalla donde se observa datos históricos y se puede generar un archivo .txt con la información de la variable elegida en un determinado período de tiempo.

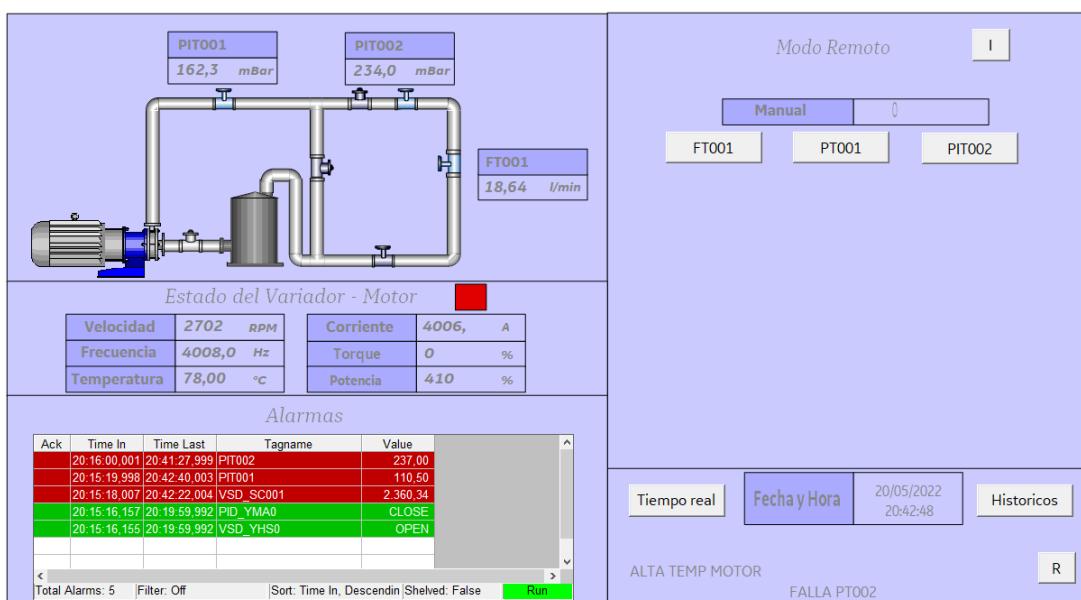


Figura 6.20: Pantalla SCADA

### **6.2.1. Configuración driver Modbus**

Para realizar la configuración de cada ícono de la pantalla SCADA con su respectiva variable, se debió crear un *MBE Driver* (Figura 6.21) dónde se estipula la dirección IP y el mapa de memoria con sus respectivas secciones que luego fueron utilizadas por el *DataBase* (Figura 6.22).

Una vez creado el *MBE Driver* se debe generar la tabla *DataBase* en donde estará el nombre, dirección IP, tipo de elemento, descripción, alarma asociada, entre otros puntos de cada elemento.

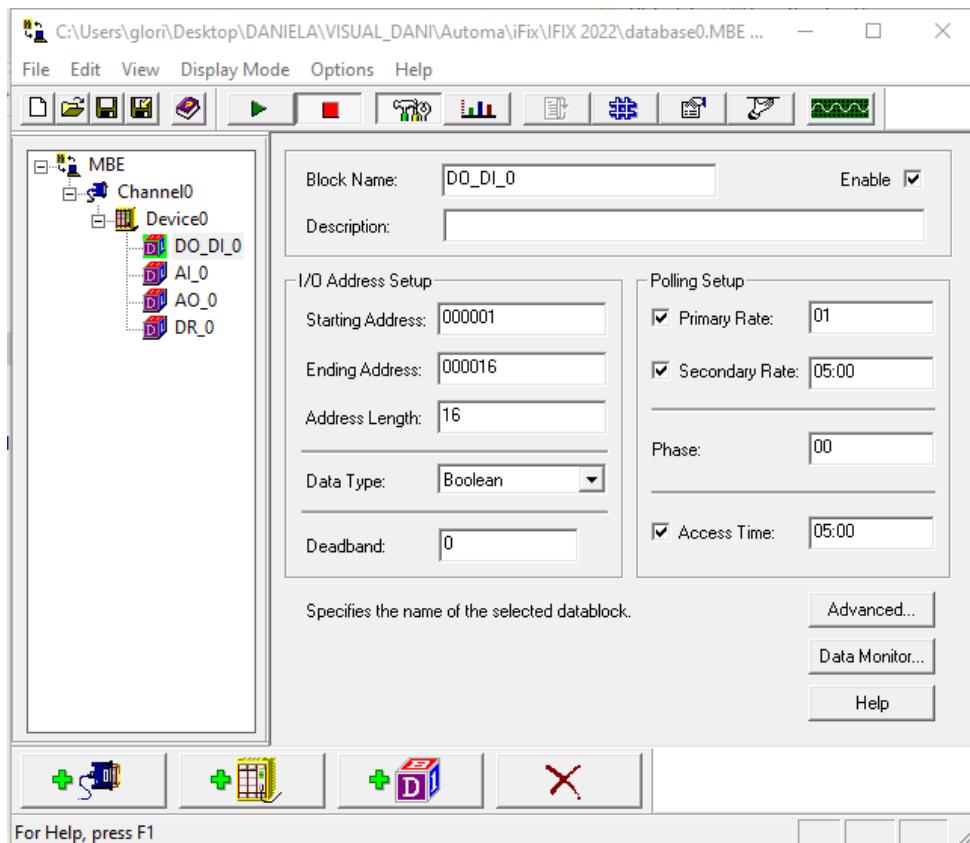


Figura 6.21: Configuración MBE

	Tag Name	Type	Description	Scan	I/O Dev	I/O Addr
1	FT001	AI	Caudal	1	MBE	Device0:400001
2	PID0FT1_KD	AI	F1 KD	1	MBE	Device0:400081
3	PID0FT1_KI	AI	F1 KI	1	MBE	Device0:400077
4	PID0FT1_KP	AI	F1 KP	1	MBE	Device0:400079
5	PID0FT1_SP	AI	Valor Ref F1	1	MBE	Device0:400061
6	PID0FT1_TD_LAG	AI	F1 TD LAG	1	MBE	Device0:400051
7	PID0PIT1_KD	AI	P1 KD	1	MBE	Device0:400067
8	PID0PIT1_KI	AI	P1 KI	1	MBE	Device0:400063
9	PID0PIT1_KP	AI	P1 KP	1	MBE	Device0:400065
10	PID0PIT1_SP	AI	Valor Ref P1	1	MBE	Device0:400057
11	PID0PIT1_TD_LAG	AI	P1 TD LAG	1	MBE	Device0:400047
12	PID0PIT2_KD	AI	P2 KD	1	MBE	Device0:400075
13	PID0PIT2_KI	AI	P2 KI	1	MBE	Device0:400071
14	PID0PIT2_KP	AI	P2 KP	1	MBE	Device0:400073
15	PID0PIT2_SP	AI	Valor Ref P2	1	MBE	Device0:400059
16	PID0PIT2_TD_LAG	AI	P2 TD LAG	1	MBE	Device0:400049
17	PID_SEL	AI	Seleccionar control P1 P2 F1	1	MBE	Device0:400083

Figura 6.22: Database con MBE

### 6.2.1.1. Pruebas mediante ModSim

Para realizar pruebas intermedias antes de unir SCADA con el programa del PLC se utilizó el software *ModSim*, dónde se generó los distintos mapas de memoria utilizados para modificar variables y observar el correcto funcionamiento de los elementos en el SCADA (Figura 6.23).

TAG INSTRUMENTO	SERVICIO	UNIDADES	RANGO	ALARMAS			ENCLAVAMIENTO	EFFECTO	PROPOSITO DE ALARMA	CONSECUENCIA DE LA NO ACCION
				MIN	MAX	DELAY				
TE001	Temperatura	°C		50					Informar alta temperatura del motor	
TE001	Temperatura	°C		70			70	P	Informar muy alta temperatura del motor	Daño al bobinado
PTI01	Presión	mbar	-1000	4000	700		700	P	Informar alta presión en cañería	Daño a bomba
PTI01	Presión	mbar						-1 P	Informar desconexión PTI01	
PTI02	Presión	mbar	-1000	4000				-1 P	Informar desconexión PTI02	
FI01	Caudal	l/min	0	60		0.5	30s	0.5 P	Informar bajo flujo	Daño a bomba
VSD_SC001	Velocidad	rpm	0	3600	200		200	P	Informar baja velocidad	Daño a motor

Tabla 6.2: Matriz causa- efecto de las alarmas

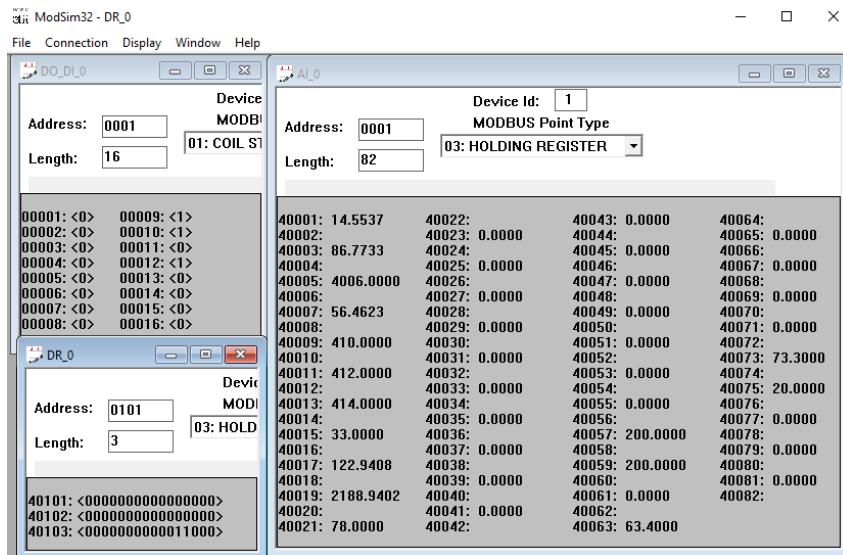


Figura 6.23: ModSim

## 6.2.2. Alarmas

Dentro de la pantalla principal es posible observar el alarmero. En la Tabla 6.2 se observa la matriz causa efecto de cada alarma, donde *P* significa paro de motor. Por ejemplo, si la temperatura del motor es 50 °C, la alarma se activará informando alta temperatura, y si la temperatura llega a 70 °C, el PLC hará que el motor se detenga para que no se produzca un daño al bobinado.

## 6.2.3. Paradas por bloqueo

El banco de pruebas está preparado para detener el motor si ocurriese alguna anomalía y mostrar la falla en la pantalla SCADA en la parte inferior derecha. Se programó para que reconozca los siguientes errores en el sistema:

- *Alta presión*
- *Bajo caudal*
- *Sensor de presión desconectado*
- *Temperatura de motor alta*
- *VSD sin comunicación*

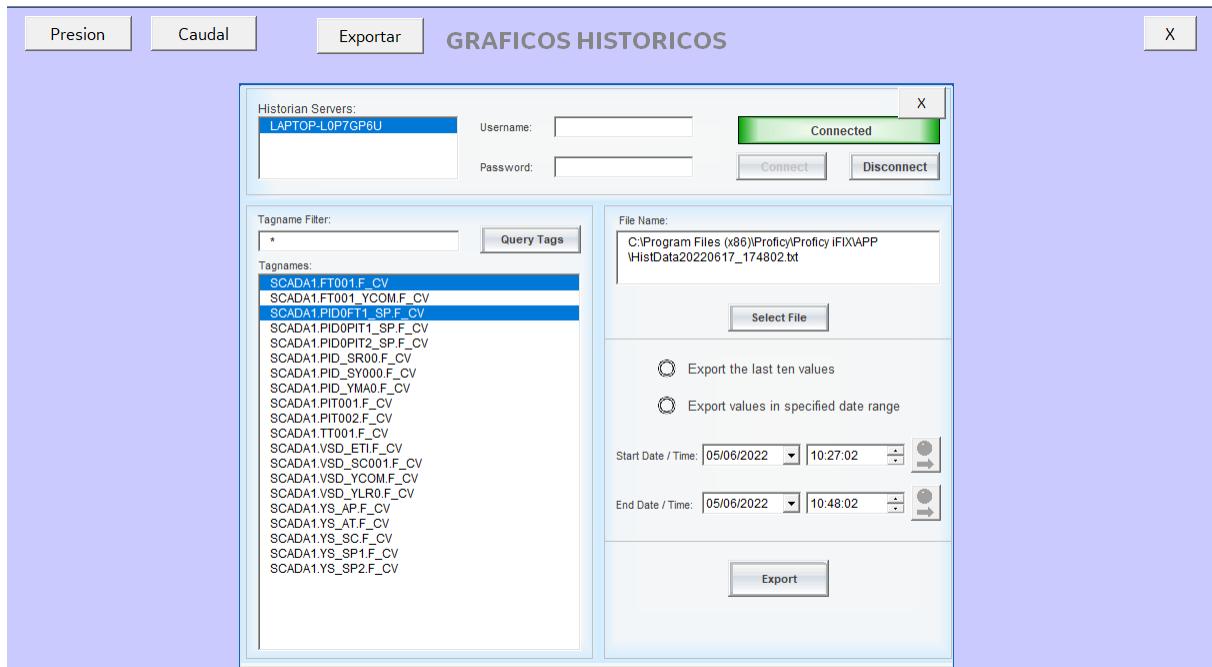
## 6.2.4. Datos históricos

Para la interpretación de datos históricos se puede realizar de dos formas, una es crear un archivo .txt y la otra es de forma gráfica.

Se generó una pantalla SCADA prediseñada de iFix dónde al conectar con el servidor Historian, busca los datos históricos de la variable elegida y guarda un archivo .txt con el horario y valores. Se puede elegir más de una variable, pero se debe tener en cuenta que no se generan columnas nuevas si no que generará en el mismo archivos más filas (Figura 6.24.a).

Otra de las opciones es observar datos históricos en un gráfico de tiempo, estas variables están preestablecidas y son las presiones y el caudal con sus respectivos valores de referencias (Figura 6.24.b).

## Automatización Industrial - Trabajo Final



(a) Pantalla para guardar datos históricos



(b) Datos históricos de forma gráfica

Figura 6.24: Datos históricos

## **7. Mejoras futuras**

En un futuro se espera que alumnos de la carrera realicen mejoras en el banco de pruebas, por ejemplo:

- Mejorar la distancia entre los sensores y los accesorios del sistemas, como válvulas o codos para que el fluido no se torne turbulento.
- Implementar sistemas sonoros o visuales de las alarmas.
- Generar una página web para observar los datos en tiempo real y/o manejar de forma remota.
- Realizar otro anexo no hidráulico para colocar al motor- variador y generar un nuevo banco de pruebas.
- Generar nuevas formas de perturbación a los sistemas.
- Implementar un sistema para controlar presión o caudal por medio de válvulas proporcionales.
- Realizar perturbaciones controladas y repetibles con válvulas proporcionales.
- Reemplazar bomba por una en mejor estado.
- Realizar pruebas de caudal y presión a mayor frecuencia.

## 8. Conclusiones

Se concluye que el banco de pruebas construido es una herramienta útil para alumnos de las carreras de ingeniería que sigan ramas orientadas al control automatizado, ya que se tiene la posibilidad de iniciarse en el uso y configuración de variadores de velocidad, programación de PLC, uso de protocolos de comunicación industriales como Modbus para sistema SCADA y CANopen para VSD- PLC.

Al decidir el objetivo, se necesitó utilizar como carga del motor una bomba periférica en desuso que se acopló mecánicamente, a raíz de esto fue necesario diseñar y construir un circuito hidráulico para generar de forma manual perturbaciones en el sistema y estudiar distintas respuestas. También con el fin de controlar caudal se implementó un prototipo capaz de medir los pulsos obtenidos de un caudalímetro a paleta y transmitirlo al PLC por Modbus TCP.

Se logró obtener el control esperado sobre tres variables distintas mediante la variación de la frecuencia del motor, dónde esta es la única acción de control en el banco de pruebas. Para realizar esto se modeló matemáticamente cada sistema y se diseñó los tres controladores PI que se implementaron.

Se generó un sistema SCADA dónde se puede observar diversas variables en tiempo real y realizar estudios de ellas a través de datos históricos. Además, en la pantalla, se puede visualizar distintos tipos de anomalías causadas por el variador y los instrumentos utilizados en el proyecto, para facilitar la detección de errores y poder solucionarlos adecuadamente.

Finalmente, la realización del proyecto tuvo un gran aporte para consolidar los conocimientos obtenidos durante la cursada de la materia *Automatización Industrial* y también para el crecimiento personal y profesional.

## 9. Referencias y Versiones de programas utilizados

### Referencias

- [1] Schneider Electric. "Manual de instalación". En: *Altivar312- Variadores de velocidad para motores asíncronos* (2013).
- [2] The Instrumentation Systems y Automation Society. "Instrumentation Symbols and Identification". En: *ISA-5.1* (1992).
- [3] Schneider Electric. "Communication variables manual". En: *Altivar312- Variadores de velocidad para motores asíncronos* (2009).
- [4] Jorge Pomares y Ángel Martínez Bueno. "Sistemas de Control Automático. Identificación experimental de sistemas". En: *Sistemas de Control Automático* (2011).

### Versiones de programas utilizados

- **Unity Pro XL** (V11.0) Schneider Electric Industries SAS (2015)
- **SoMove** (V2.8.4.0) Schneider Electric Industries SAS (2020)  
*Altivar DTM Library ATV31/312* (V2.0.2.0)
- **iFix Workspace** (V6.0) General Electric Company (2018)
- **Matlab** (V2018a) MathWorks (2017)  
*Simulink* (V2018a) MathWorks (2017)
- **Proficy Historian** (V4.5) General Electric Company (2011)
- **OPC Factory Server** (V3.6) Schneider Electric Industries SAS (2015)

## A. Anexo: Diagrama P & ID

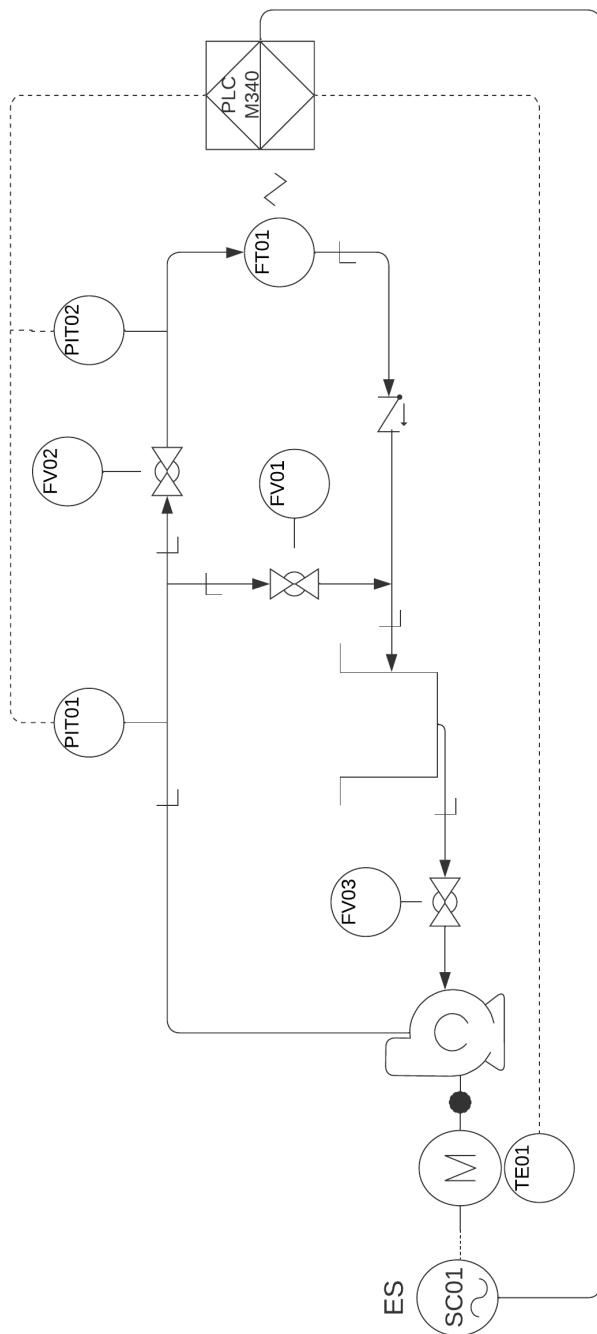


Figura A.1: Diagrama p&id

## B. Anexo: Tabla de Direcciones ModBus

Tabla B.1: Tabla de variables y direcciones

<b>Tipo de variable</b>	<b>TAG</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Dirección</b>
DO	PID_XSMA0	Habilitar control automatico manual	Boolean	Device0:000003
DO	VSD_XSM0	Marcha desde el hdmi	Boolean	Device0:000004
DO	VSD_XST0	Parada desde SCADA	Boolean	Device0:000005
DO	PLC_XSDES	Desenclave de bomba	Boolean	Device0:000006
DO	PID_XSA0	PID modo automatico	Boolean	Device0:000008
DO	PID_XSM0	PID modo manual	Boolean	Device0:000009
DO	VSD_XSFO	Restablecer fallas VSD	Boolean	Device0:000010
DI	VSD_YHS0	Parada de emergencia fisico	Boolean	Device0:000011
DI	VSD_YLR0	Modo fisico o scada	Boolean	Device0:000012
DI	PID_YMA0	Estado manual automatico	Boolean	Device0:000013
DI	PID0PIT1_XRST	Restablecer valores PID	Boolean	Device0:000014
DI	PID0PIT2_XRST	Restablecer valores PID	Boolean	Device0:000015
DI	PID0FT1_XRST	Restablecer valores PID	Boolean	Device0:000016
DI	VSD_YERR	Muestra error de VSD	Boolean	Device0:000017
AI	FT01	Caudal	Float	Device0:400001
AI	VSD_AI1C	Valor del pote	Float	Device0:400003
AI	VSD_IC000	Corriente	Float	Device0:400005
AI	VSD_SC000	Frecuencia	Float	Device0:400007
AI	VSD_JC000	Potencia	UInt	Device0:400009
AI	VSD_WC000	Torque	UInt	Device0:400011
AI	VSD_EC000	Voltaje	Float	Device0:400013
AI	PIT02	Presion2	Float	Device0:400015
AI	PIT01	Presion1	Float	Device0:400017
AI	VSD_SC001	Velocidad	UInt	Device0:400019
AI	TE001	Temperatura	Float	Device0:400021
AI	PID_SY000	Accion de control	Float	Device0:400023
AI	VSD_R_ERR	Errores, tabla de errores	UInt	Device0:400025

Sigue en la página siguiente

**Tabla B.1 – continuación**

<b>Tipo de variable</b>	<b>TAG</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Dirección</b>
AI	PID0PIT1_TD_LAG	P1 TD LAG	Float	Device0:400047
AI	PID0PIT2_TD_LAG	P2 TD LAG	Float	Device0:400049
AI	PID0FT1_TD_LAG	F1 TD LAG	Float	Device0:400051
AI	PID_SR00	Valor manual, control desac	Float	Device0:400053
AI	PID0PIT1_SP	Valor Ref P1	Float	Device0:400057
AI	PID0PIT2_SP	Valor Ref P2	Float	Device0:400059
AI	PID0FT1_SP	Valor Ref F1	Float	Device0:400061
AI	PID0PIT1_KI	P1 KI	Float	Device0:400063
AI	PID0PIT1_KP	P1 KP	Float	Device0:400065
AI	PID0PIT1_KD	P1 KD	Float	Device0:400067
AI	PID0PIT2_KI	P2 KI	Float	Device0:400071
AI	PID0PIT2_KP	P2 KP	Float	Device0:400073
AI	PID0PIT2_KD	P2 KD	Float	Device0:400075
AI	PID0FT1_KI	F1 KI	Float	Device0:400077
AI	PID0FT1_KP	F1 KP	Float	Device0:400079
AI	PID0FT1_KD	F1 KD	Float	Device0:400081
AI	PID_SEL	Seleccionar control P1 P2 F1	UInt	Device0:400083
DR	VSD_ETI	Motor encendido o apagado	Boolean	Device0:400101:4
DR	YS_AP	Alta presion	Boolean	Device0:400103:0
DR	YS_SC	Sin Caudal	Boolean	Device0:400103:1
DR	YS_SP1	Sin Sensor P1	Boolean	Device0:400103:2
DR	YS_SP2	Sin Sensor P2	Boolean	Device0:400103:3
DR	YS_AT	Alta Temp	Boolean	Device0:400103:4
DR	FT01_YCOM	Sin Caudal	Boolean	Device0:400103:5
DR	VSD_YCOM	Sin comunicación con variador	Boolean	Device0:400103:6

## C. Anexo: Manual BANCO-SCADA

### C.1. Características generales

- Para sistemas operativos Windows 7 y Windows 10.
- Se puede utilizar el banco de pruebas en modo local o remoto.
- Diseñado para controlar tres variables distintas, dos de presión o caudal.
- Fácil de transportar.
- Posibilidad de generar perturbaciones en el sistema para observar distintas respuestas.
- Posibilidad de guardar valores históricos y ver datos en tiempo real.

### C.2. Guía de uso

1. Conectar el banco de pruebas con los respectivos sensores y transmisores en el módulo correspondiente del PLC.
2. Conectar cable Ethernet entre PLC y router o PC.
3. Alimentar el sistema del variador con tensión trifásica y la fuente del PLC con tensión monofásica.
4. Verificar que los dispositivos se encuentren en la misma red.
5. Abrir iFix, la base de datos y el *driver MBE* dónde se procederá a conectar el sistema SCADA.
6. Comprobar comunicación al ejecutar la pantalla SCADA.

#### C.2.1. Pantalla principal

En la figura C.1 se observa la pantalla principal del SCADA y en la figura C.2 se divide en varias secciones.

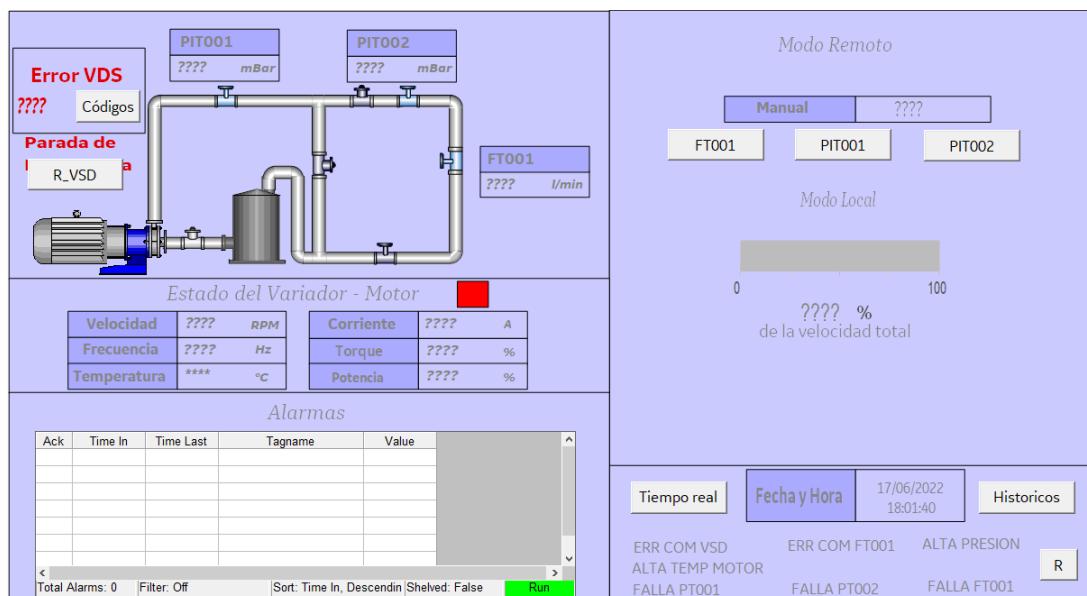


Figura C.1: Pantalla principal

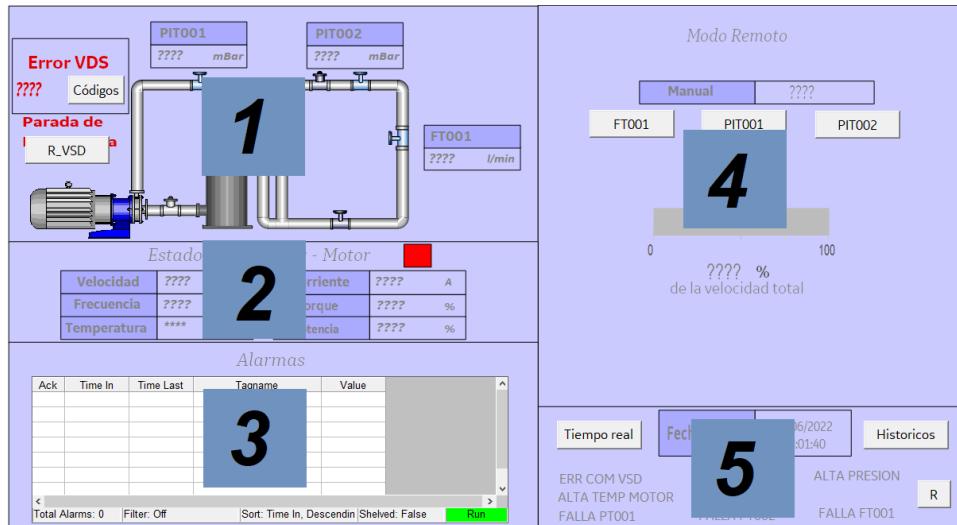


Figura C.2: Partes del sistema SCADA

### C.2.1.1. Diagrama del banco de pruebas

Se observan los valores de presión y caudal en un diagrama similar al banco de pruebas físico.

En caso de ocurrir un error sobre el VSD, se observará el código del lado izquierdo (A), a su vez al presionar sobre el botón *Códigos* se mostrará la tabla de errores. Si se desea reconocer el error se debe presionar el botón B (Figura C.3).

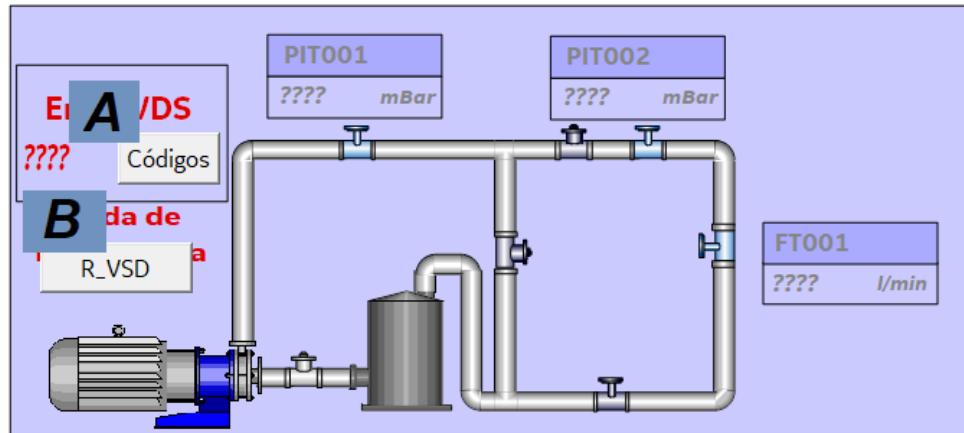


Figura C.3: Subpantalla 1

También en la parte izquierda de la pantalla, se mostrará la indicación de parada de emergencia física.

### C.2.1.2. Estado del variador - Motor

En esta sección se ve una señal piloto de color rojo, si es que el motor se encuentra en funcionamiento, lo que indica señal de cuidado o alerta; mientras que en color verde si el motor está apagado. Se observan variables de velocidad y frecuencia del motor, temperatura obtenida por el RTD y corriente, torque y potencia consumida por el motor.

### C.2.1.3. Alarmas

Sección destinada al alarmero dónde muestra el período de tiempo que ocurrió el hecho, el nombre de la variable y el valor alcanzado.

### C.2.1.4. Modo remoto/ Modo local

- Modo Local

Si la llave selectora del banco de pruebas se encuentra en modo local, se observará el cartel que indica *Modo Local* (Figura C.4.a).

En modo manual, permite comandar el motor con la llave selectora y modificar la velocidad con el potenciómetro ubicado en la parte derecha de la caja. En la subpantalla 1 (Figura C.3) se podrá ver las variables del sistema y abajo de la leyenda *Modo Local* el porcentaje de la velocidad total a la que está girando el motor. En la parte superior de la caja se encuentra un pulsador para activar la parada de emergencia del sistema que puede utilizarse en cualquier modo (Modo remoto / local).

- Modo Remoto

Si la llave selectora se encuentra en modo remoto, el sistema está preparado para recibir órdenes desde SCADA. El motor puede encenderse al presionar la tecla A con una velocidad en RPM preestablecida previamente con un valor entero entre 0 y 3600 dentro de la casilla *Manual* (B) (Figura C.4.b).

El banco solo puede realizar el control de una variable a la vez independientes entre si, debido a esto, para controlar el lazo requerido (PIT01, PIT02, FT01) se selecciona el botón deseado (Figura C.5 - B). Se abrirá una pantalla con los parámetros actuales del PID, los cuales se pueden modificar para realizar ensayos con diferentes respuestas (se debe usar la coma como separador decimal). Con el “botón C”, se podrá restablecer los parámetros del PID cuyos valores fueron fijados al momento de realizar el proyecto.

Al colocar el selector en automático (D), se cierra el lazo del sistema y puede establecerse la presión o caudal deseado en “Ref” (E). Los rangos serán establecidos según la tabla C.1.

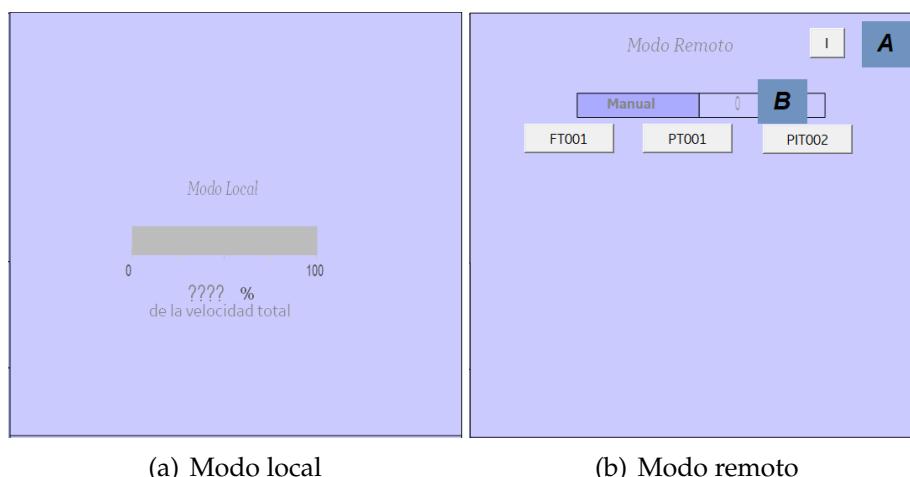


Figura C.4: Subpantalla 4

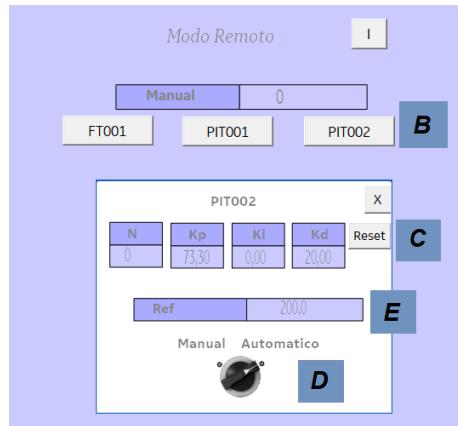


Figura C.5: Modo lazo cerrado

<b>Variable</b>	<b>Rango</b>	<b>Unidades</b>
<b>PIT001</b>	Presión	60 - 930
<b>PIT002</b>	Presión	60 - 700
<b>FT001</b>	Caudal	1/min

Tabla C.1: Rangos de las variables

#### C.2.1.5. Fallas y ventanas de gráficos

En la pantalla 5 se observan los carteles de eventuales fallas que puedan ocurrir en el sistema, y se pueden reanudar con el botón *R*.

Otra de las cosas que tiene esta subpantalla es la fecha, hora y dos botones dónde al presionarlos se abren pantallas para observar gráficamente datos en tiempo real o de forma histórica (Sección C.2.2 y C.2.3).

#### C.2.2. Pantalla de datos en tiempo real

En la figura C.6 se puede observar la pantalla con valores instantáneos (la imagen no corresponde a valores reales tomados durante las pruebas).

#### C.2.3. Pantalla de datos históricos

Para la interpretación de datos históricos se puede realizar de dos formas, una es crear un archivo .txt y la otra es de forma gráfica.

Para observar los datos históricos en el gráfico de tiempo, se debe presionar alguna opción de los botones superiores izquierdos, además se selecciona el período de tiempo que se desea visualizar (Figura C.7.b).

Para generar el archivo .txt con los valores históricos se presiona el botón *Exportar*, se conecta con el servidor Historian y se selecciona las variables de interés con el período que se quiera guardar. Finalmente se le asigna un nombre al archivo y se presiona *Export* (Figura C.7.a). Estos archivos luego pueden ser procesados con un script de Matlab generado para tal fin (C.8).

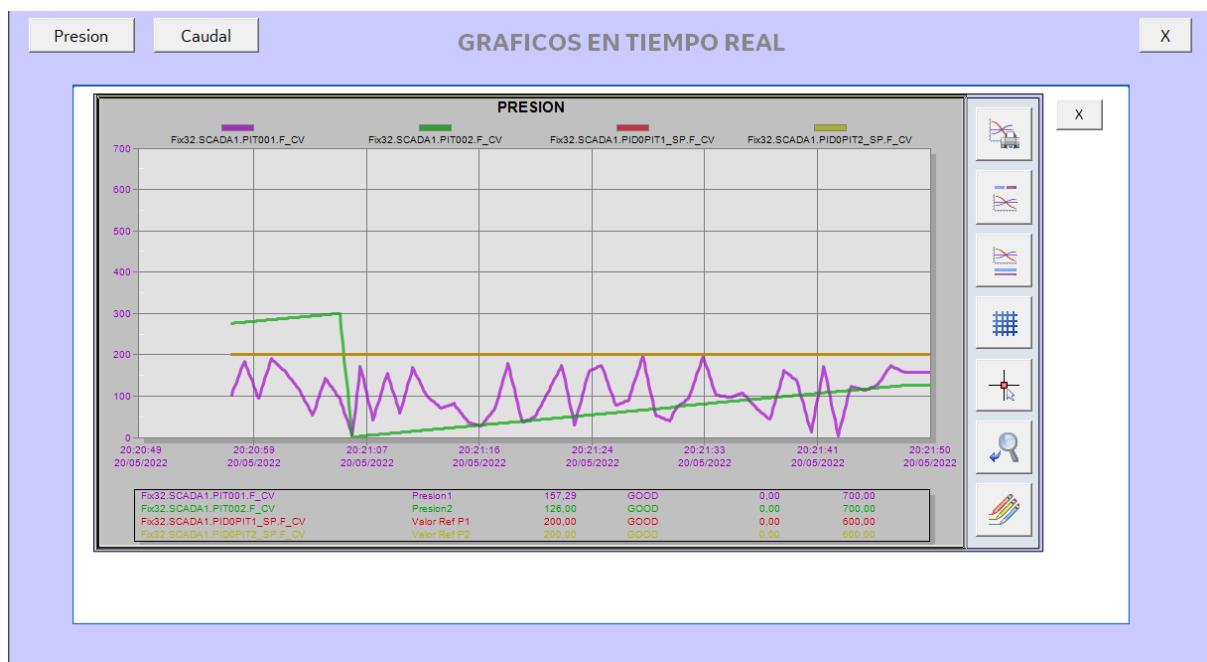
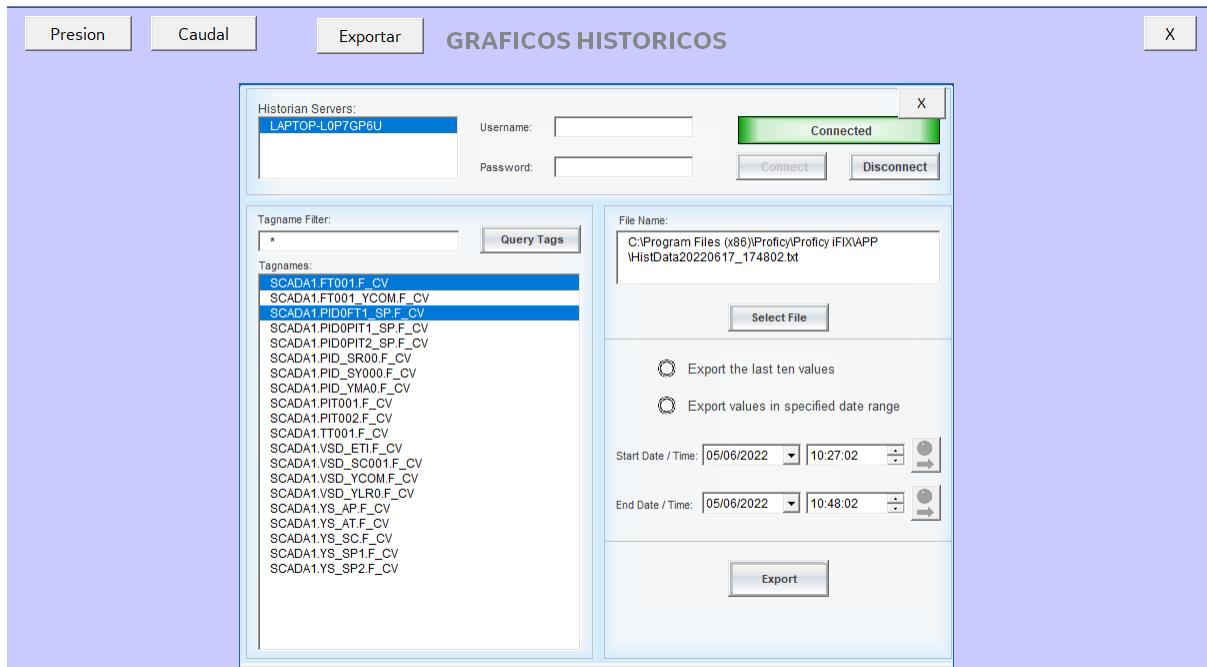


Figura C.6: Pantalla de datos en tiempo real

## Automatización Industrial - Trabajo Final



(a) Pantalla para guardar datos históricos



(b) Datos históricos de forma gráfica

Figura C.7: Datos históricos

```
Que variable desea visualizar? - Coloque vector entre corchetes. Ej: [1,2]
1= SCADA1.PID0FT1_SP.F_CV
2= SCADA1.PID_SR00.F_CV
3= SCADA1.PIT001.F_CV
4= SCADA1.FT001.F_CV
5= SCADA1.FT001_YCOM.F_CV
6= SCADA1.PID0FT1_SP.F_CV
7= SCADA1.PID_SY000.F_CV
8= SCADA1.PIT001.F_CV
9= SCADA1.PIT002.F_CV

Usted elige : [1,2,3,4,5,6,7,8,9]
fx >>
```

Figura C.8: Programa para lectura de datos Históricos

## D. Anexo: Archivos extras

Dentro del repositorio de Github [PONER LINK](#) se encuentra la carpeta *ArchivosMatlab* con ejemplos y scripts de Matlab nombrados a continuación:

- *Cargo\_Historico.m*: Se utiliza para interpretar y graficar datos históricos obtenidos del archivo que se guarda por medio de SCADA.
- *distintos\_PID.m*: compara distintos escalones y PID de las variables con ejemplos ya guardados.
- *Graficos.m*: Script preparado para graficar las variables PIT01, PIT02 y FT01 vs. tiempo.
- *metodo\_string.m*: utilizado para realizar la estimación de la planta según método String.
- *Levanto\_datos\_CSV\_V3.m*: Con este archivo se guardan datos a través de un cliente OPC.
- *Lectura\_Datos\_OPc\_V3.slx*: Archivo simulink utilizado en conjunto con *Levanto\_datos\_CSV\_V3.m*
- *Ejemplos/Referencias.png*: Datos de variable, PID y escalones de los ejemplos encontrados en la carpeta.
- Carpeta *EjHist*: Archivos de ejemplo para utilizar con *Cargo\_Historico.m*.
- Carpeta *Carpeta01*: Carpeta dónde se guardarán los archivos generados con *Levanto\_datos\_CSV\_V3.m*

En este mismo repositorio se encuentra este informe, la presentación dada el día de la exposición y los archivos de los software *UnityPro* y *iFix* utilizados para este trabajo.