

**UNIVERSIDAD NACIONAL DE LA PATAGONIA
SAN JUAN BOSCO**

**AUTOMATIZACIÓN INDUSTRIAL
TRABAJO FINAL**

Banco de pruebas para motor trifásico

Alumnos

**CAAMIÑA, Daniela
YAPURA, Cristian**

Docentes

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

MES AÑO



Índice

1. Introducción	4
2. Objetivo	5
3. Definiciones	6
4. Elementos	8
4.1. Motor	8
4.2. Variador de velocidad	8
4.3. Módulo Didáctico PLC M340	8
4.4. Banco de pruebas	9
4.4.1. Transductor de presión	10
4.4.2. Sensor de caudal	10
4.5. Diagrama	11
4.6. Presupuesto	11
5. Preliminares	12
5.1. Programación variador de velocidad	12
5.2. Comunicación variador de velocidad - PLC	13
5.3. Lectura de registros	14
5.3.1. Programación Unity Pro	14
5.4. Medición de caudal	17
6. Desarrollo	18
6.1. Programa final Unity PRO	18
6.1.1. Direcciones utilizadas	18
6.2. Adquisición de datos	18
6.2.1. Uso de Matlab	19
6.2.2. Estimación de la planta	19
6.2.3. PID calculado	19
6.3. SCADA	20
6.3.1. Configuración iFix o Configuracion MBE?	21
6.3.1.1. MBE	21
6.3.2. DataBase	21
6.3.2.1. Lista de direcciones utilizadas	21
6.3.2.2. Pruebas mediante ModSim	21
6.3.3. Distintas pantallas	21
6.4. Alarmas	21
6.5. iHistorian	21
6.6. Direcciones	21
7. Conclusiones	21
8. Bibliografía	22
9. Anexos	22
10. Manual SCADA	22

Índice de figuras

4.1. Motor Altium	8
4.2. Variador Altivar 312	8
4.3. Módulo Didáctico PLC M340	9
4.4. Banco de pruebas completo	9
4.5. Banco de Pruebas	10
4.6. Transductores	10
4.7. Diagrama i&pd	11
5.1. Elección de Altivar 312	12
5.2. Parámetros del variador	13
5.3. Lista de parámetros modificados	13
5.4. Diagrama comunicación PC- Variador	13
5.5. Cable de comunicación	13
5.6. Conexión fichas RJ45- DB9	14
5.7. Elección del bastidor	15
5.8. Módulos PLC	15
5.9. Dirección IP	16
5.10. Dirección módulo Ethernet	16
5.11. HMI simple	16
5.12. Programa con bloques MFB	17
5.13. Diagrama de flujo del caudalímetro	17
6.1. Configuración OPC	18
6.2. Conexión servidor OPC	18
6.3. Cliente OPC en Simulink	19
6.4. Placa con sensores	20
6.5. Pantalla SCADA	20
6.6. ModSim	21
6.7. Pantalla SCADA	22

Lista de Acrónimos

CANOpen:

HMI: *Interfaz Humano-máquina*

MBE :

Modbus: *Modicon Bus*

PDO : *Objetos de Datos de Proceso*

PLC : *Controlador Lógico Programable*

SCADA : *Supervisión, Control y Adquisición de Datos*

SDO : *Objetos de Datos de Servicio*

:

:

:

:

:

:

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 prueba 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

Un banco de pruebas de un motor cuenta con un punto de apoyo 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 los procedimientos de prueba. Un banco de pruebas puede ser un prototipo de un 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 generar un banco de pruebas que sea capaz de controlar la presión o caudal de agua a través de un sistema construido por nosotros mismos que cuente con:

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

3. Definiciones

Motor eléctrico

Los motores eléctricos son máquinas que transforman la energía eléctrica en movimiento (energía cinética). Estos aparatos se componen, básicamente, del rotor y de un estator donde tiene bobinas inductoras desfasadas entre sí 120°

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 del torque y operaciones que mejoran la eficiencia energética.

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.

no me cierra esta definicion, buscar otra

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. El HMI se utiliza para monitorear o visualizar lo ejecutado por SCADA.

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) asincrónico que se utiliza es de la marca **Altium** perteneciente a la firma **Schneider Electric**. Las especificaciones se muestran a continuación

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 Altium

4.2. Variador de velocidad

El variador de velocidad que se utilizó pertenece a la marca **Schneider Electric** (Figura 4.2) y posee las siguientes características.

Altivar 312

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



Figura 4.2: Variador Altivar 312

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 de la empresa Schneider Electric. Esto cuenta con entradas analógicas, digitales, distintos métodos de comunicación y la capacidad de agregarle otros módulos según las necesidades de los proyectos a desarrollar en un riel.

Los módulos con los que se cuenta son:

- falta esto

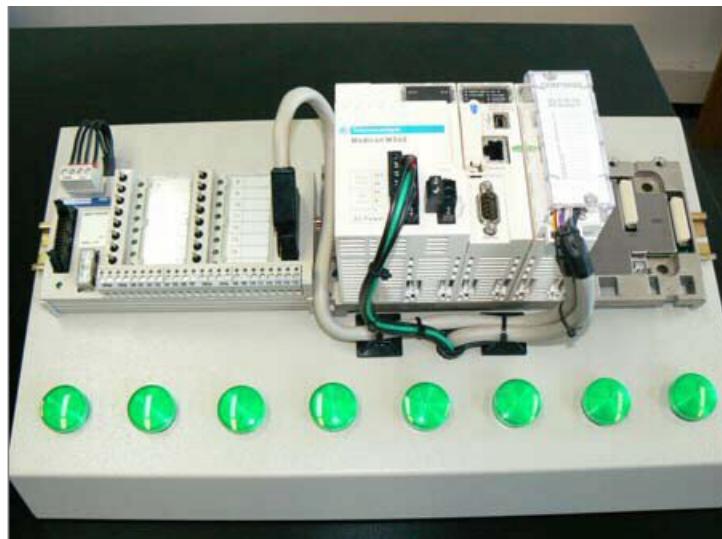


Figura 4.3: Módulo Didáctico PLC M340

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, diseñado y construido por el profesor Gerardo Arthz. A estos elementos se realizó las correspondientes conexiones, y se agregó elementos adicionales: 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.
- Circuito hidráulico, que incluye un tanque, válvulas y sensores de caudal y presión.



Figura 4.4: Banco de pruebas completo

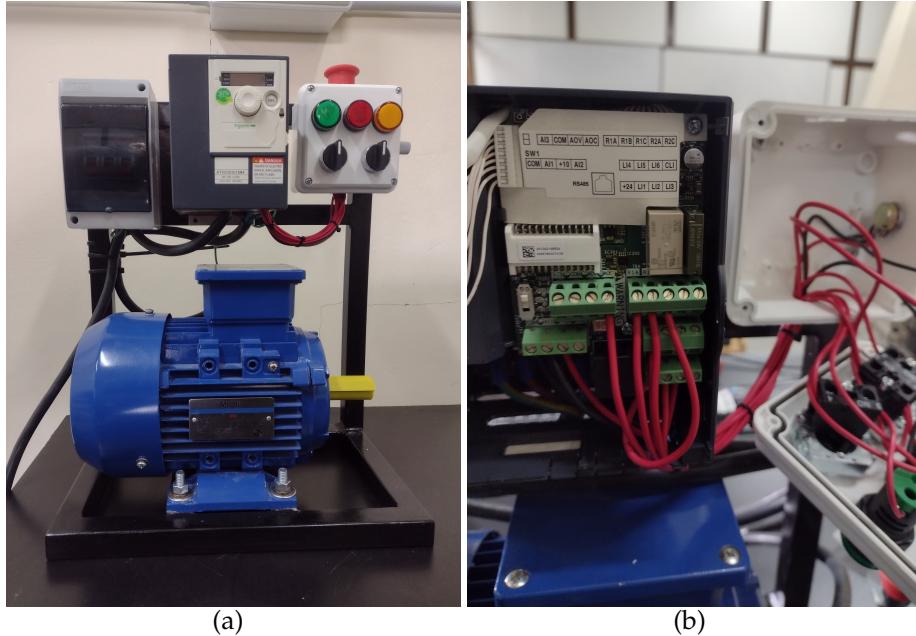


Figura 4.5: Banco de Pruebas

4.4.1. Transductor de presión

Para este proyecto se utilizan dos transductores de presión de montaje en línea modelo EJA530E de la familia DPharp de Yokogawa (Figura 4.6.a).

Las características del EJA530E son:

- Precisión: $\pm 0,055\%$ de precisión
- 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) genérico con las siguientes características:

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



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

Figura 4.6: Transductores

4.5. Diagrama

Colocar el diagrama tipo imagen pero ver si se agrega el variador y el plc
FIGURA4.7

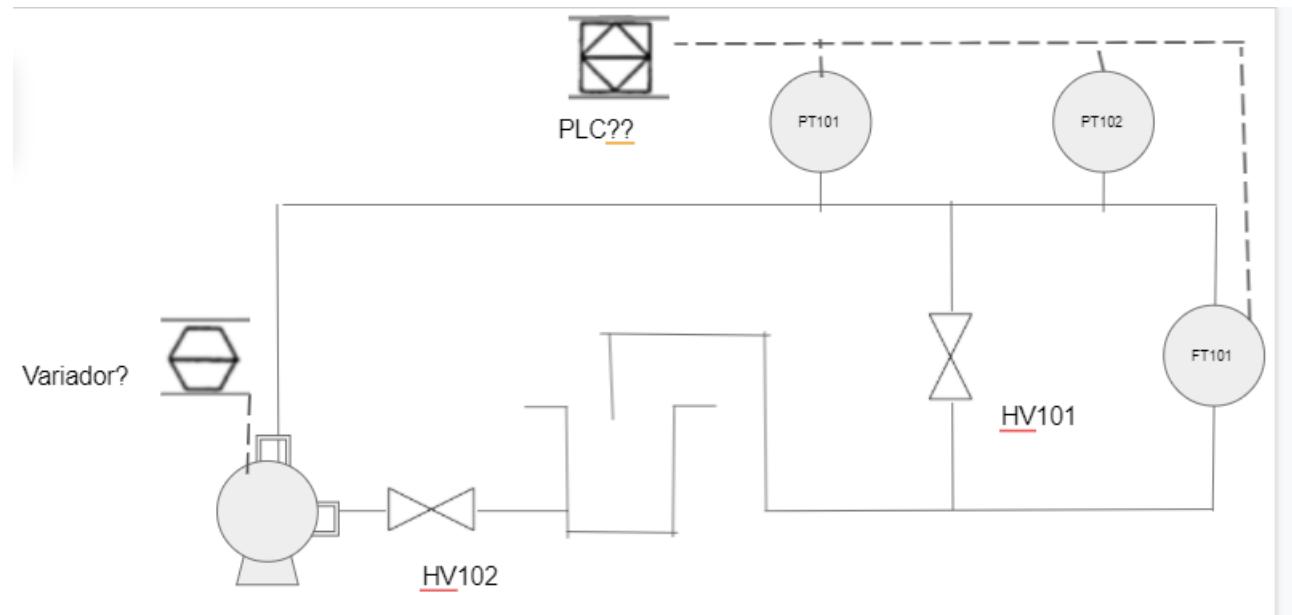


Figura 4.7: Diagrama i&pd

4.6. Presupuesto

falta lo de la bomba https://docs.google.com/spreadsheets/d/1mFoNvgJXUdL2bNnspaBJ_wS5fFA1y1c8fT09Rfod7H0/edit#gid=0

5. Preliminarios

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 Schneider¹ y luego, la librería DTM correspondiente al variador a utilizado².

Una vez realizado esto 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) y establecer el modo de funcionamiento de las entradas y el protocolo de comunicación.

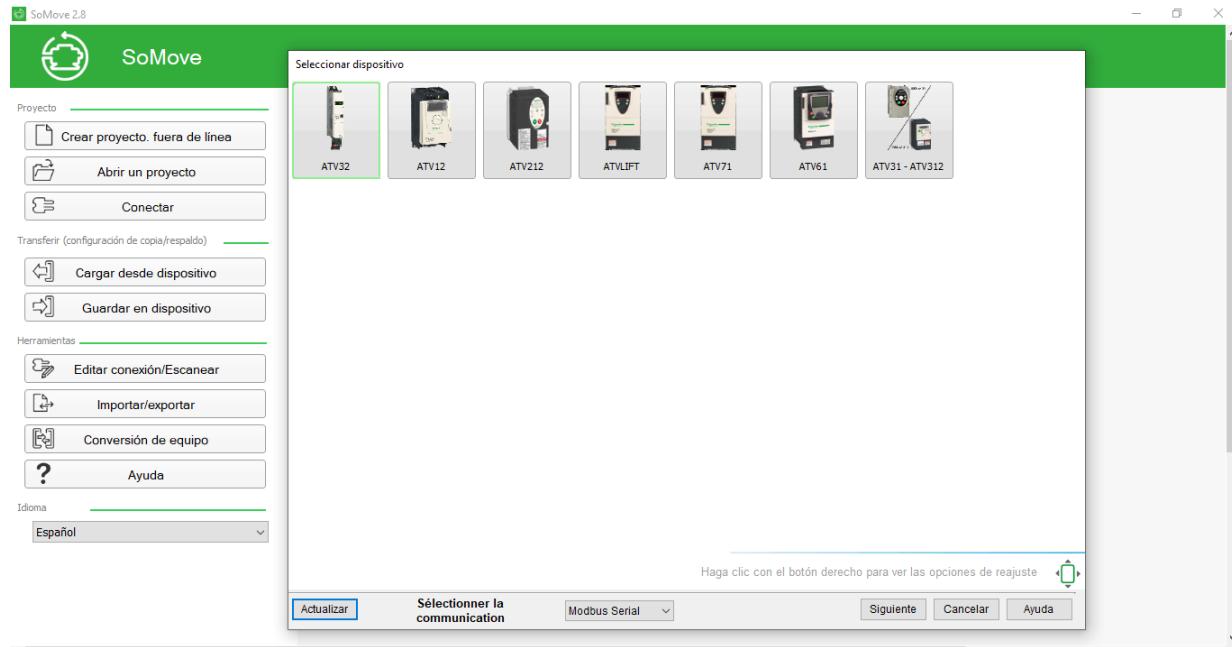


Figura 5.1: Elección de Altivar 312

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

²https://www.se.com/ar/es/download/document/Altivar_DTM_Library/

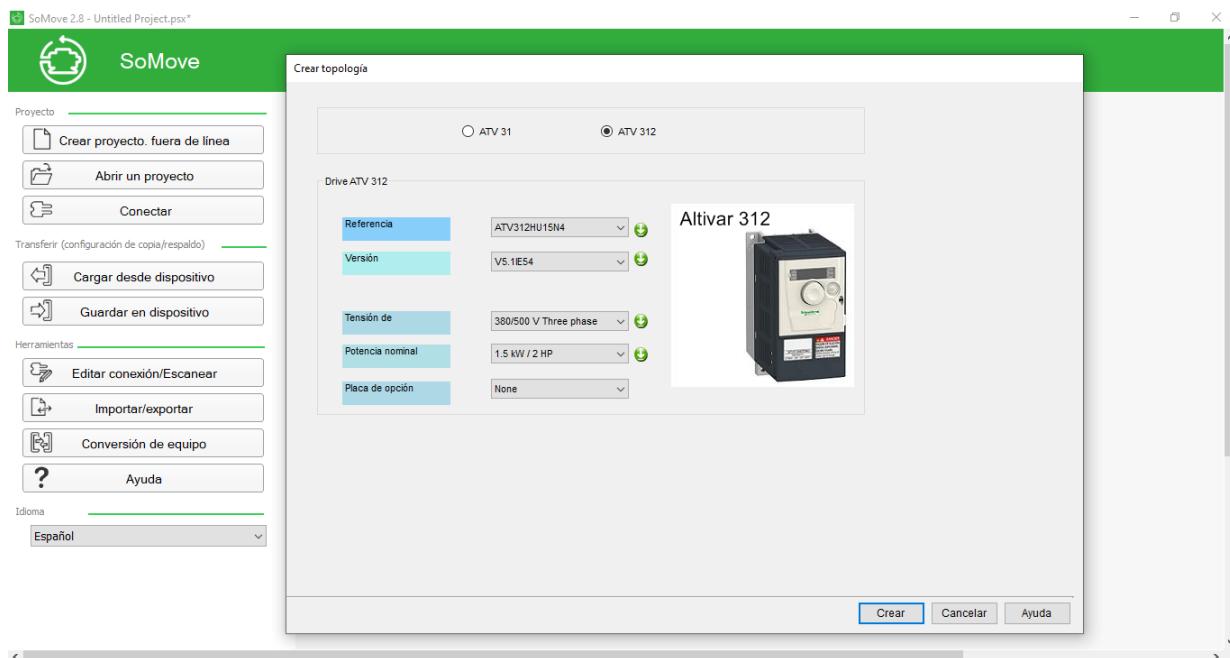


Figura 5.2: Parámetros del variador

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.commut.ref.(1a 2)	Li4	Canal1 act.
CHCF	Config. modo control	Separados	No separad.
CCS	Comunicación canal ctrl	Li4	Canal1 act.
ROT	Sent marcha autorizado	Ambos	Avance
DO	Salida Analogólogica	Fallo equipo	No
R1	Asignación del relé R1	Var.marcha	Sin fallo
FST	asignación stop rápida	Li5	No
PS2	2 velocidad preselec.	No	Li3
PS4	4 velocidad preselec.	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

Para realizar esta primera configuración se realizó la comunicación de la PC con el variador a través del protocolo **Modbus** (Figura 5.4) a través de un cable que en un extremo poseía ficha RJ45 y un conversor RS485 y en el otro ficha USB (Figura 5.5).

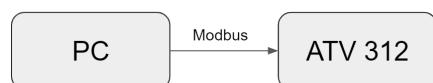


Figura 5.4: Diagrama comunicación PC- Variador

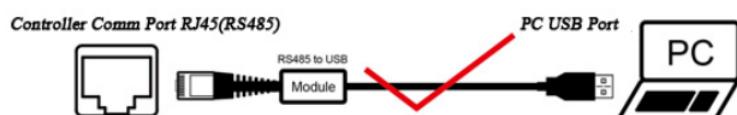


Figura 5.5: Cable de comunicación

5.2. Comunicación variador de velocidad - PLC

Para poder realizar la comunicación entre el variador y el PLC es necesario contar con un cable que realice la conexión desde la salida CANOpen a RS485, para esto

se necesitó hacer un cable con las fichas correspondientes en cada extremo según las conexiones que muestran en la Figura 5.6

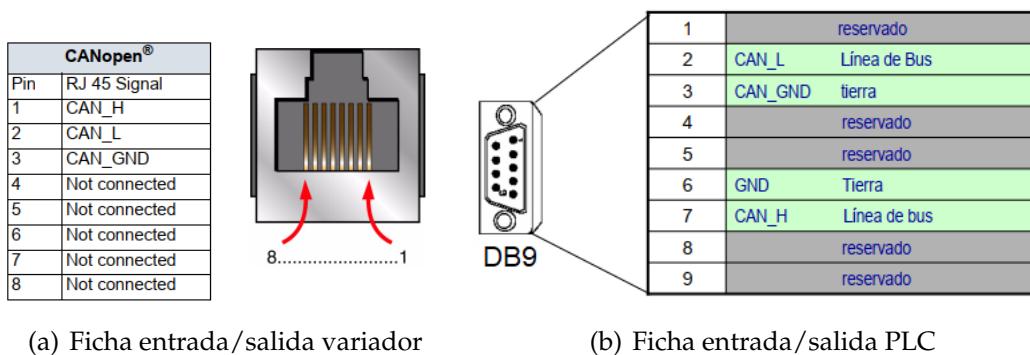


Figura 5.6: Conexión fichas RJ45- DB9

5.3. Lectura de registros

5.3.1. Programación Unity Pro

Para generar la base del proyecto para trabajar, se debe descargar desde la página oficial e instalar el software Unity Pro XL y la librería DTM utilizada en el software soMove correspondiente al variador que se posee. Una vez que esto está instalado se abre un nuevo proyecto y se configura siguiendo los siguientes pasos.

1. Se selecciona el bastidor que se posee.
2. En la configuración gráfica del bastidor (Figura 5.8) podemos introducir los módulos deseados haciendo un clic en la posición seleccionada (Figura 5.7). El laboratorio cuenta con un PLC modular didáctico de la marca **Schneider Electric** de la familia **Modicon** modelo **M340** con los módulos nombrados anteriormente (Sección 4.3).
3. Se debe configurar el módulo Ethernet desde el explorador de Proyectos desplegamos la carpeta Comunicación y se realiza clic con el botón derecho sobre Redes y luego en Nueva Red, Ethernet (Figura 5.10).
4. Se creó una nueva sección para lenguaje FDB para ver parámetros básicos.
 - Los Diagramas de Bloques de Función consisten en un Editor gráfico orientado al dibujo de bloques. El lenguaje consiste en los Bloques de Funciones reusables elementales y derivados.

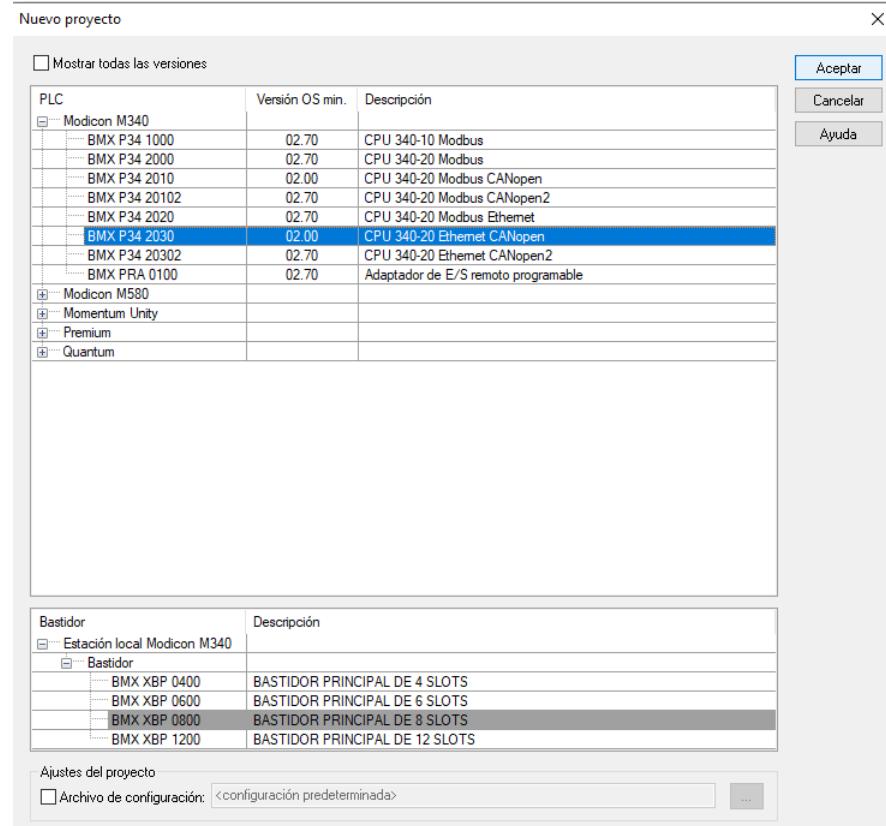


Figura 5.7: Elección del bastidor

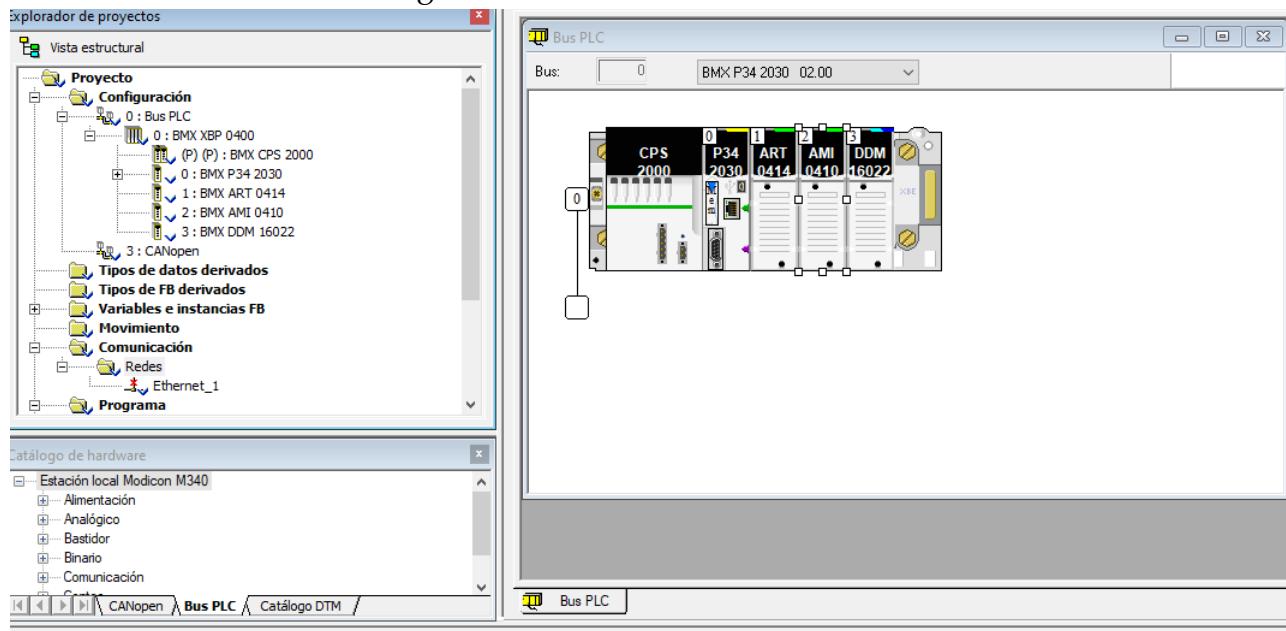


Figura 5.8: Módulos PLC

Una vez que se completó la configuración de la comunicación variador - PLC se procedió a crear una Pantalla de operador dentro del mismo programa donde por esto se es capaz de observar a través de un simple HMI parámetros del motor brindados por el variador de velocidad.

En la figura 5.11 se ve una interfaz humano máquina (HMI) simple donde se observan y se puede interactuar con los selectores, configurar velocidad, observar las señales luminosas y distintos valores de funcionamiento del motor proporcionados por el variador de velocidad.

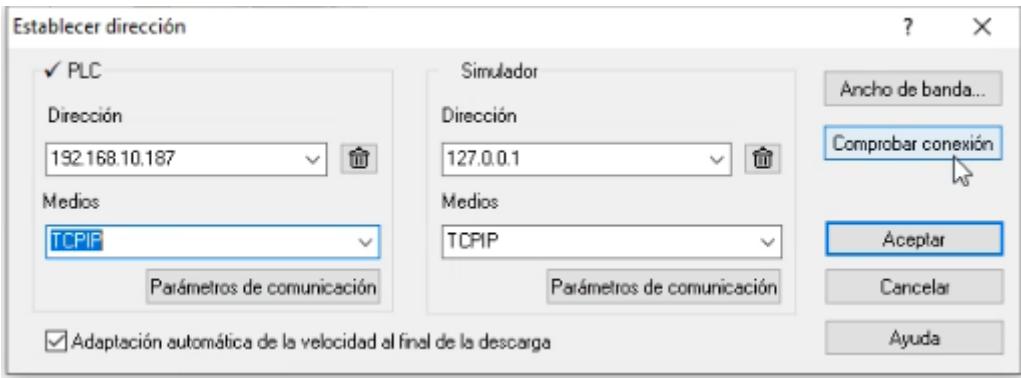


Figura 5.9: Dirección IP

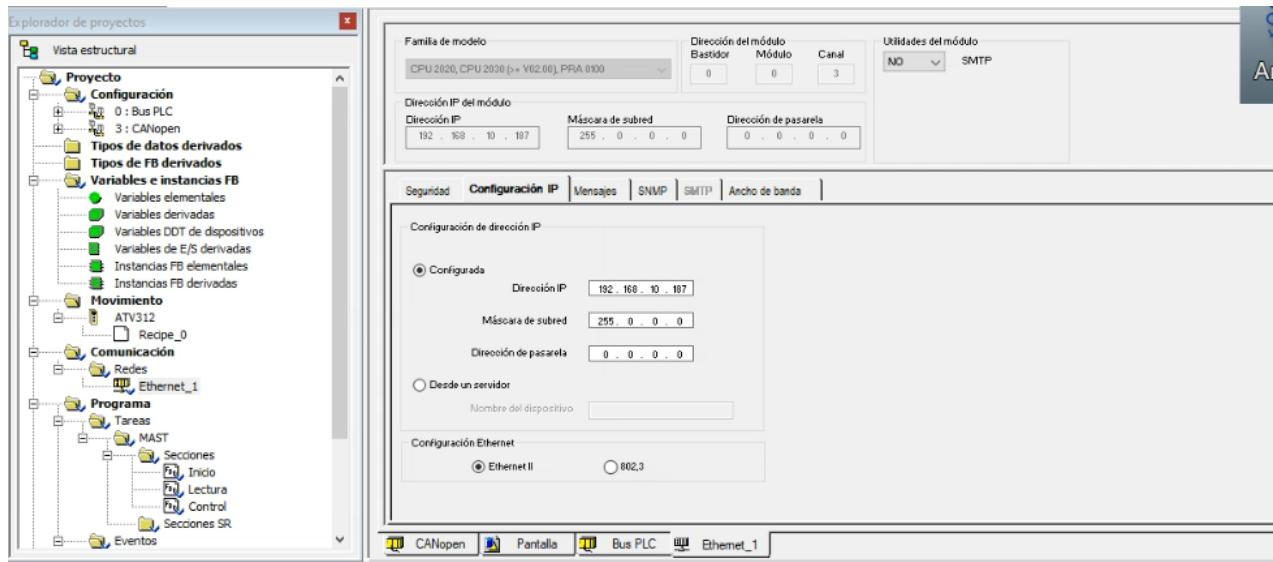


Figura 5.10: Dirección módulo Ethernet

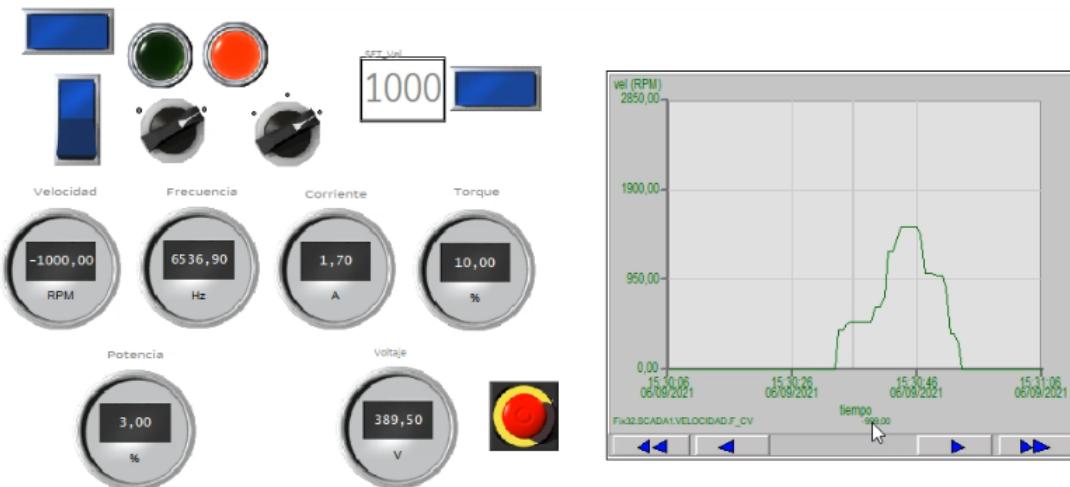


Figura 5.11: HMI simple

Para realizar este HMI se utilizó los bloques de funciones de movimiento (Motion Function Block, MFB) del software UnityPro. Estos bloques necesitan de un bloque maestro “CAN_HANDLER” el cual permite comprobar la comunicación CANopen, así como la coherencia entre las configuraciones de software y física. Otros de los bloques más utilizados dentro del programa fueron “MC_READPARAMETER” que se utiliza para leer, mediante mensajes Service Data Object (SDO), una variable del variador definida en una dirección CANOpen

dada por el mismo variador [2] (Figura 5.12).

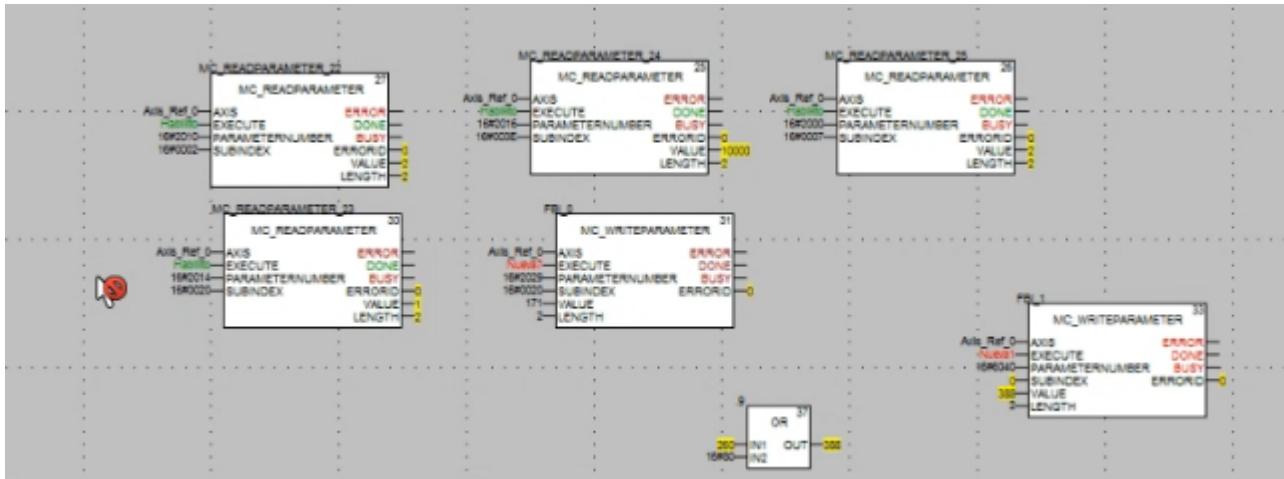


Figura 5.12: Programa con bloques MFB

5.4. Medición de caudal

Según el caudal que pase por el caudalímetro, este entregaría pulsos que debían ser leídos con un módulo externo del PLC por la baja resolución que posee. Para esto se planteó utilizar un ESP8266 capaz de obtener los pulsos, colocarlos en un registro y enviarlos por un servidor Modbus TCP/IP cada 20ms al PLC. En la sección de programación del PLC se realizó las cuentas correspondientes para realizar la conversión de pulsos a caudal (Figura 5.13). Tanto para este módulo, como el caudalímetro se usó una fuente externa de 3,3V para su alimentación.



Figura 5.13: Diagrama de flujo del caudalímetro

6. Desarrollo

6.1. Programa final Unity PRO

A partir del programa utilizado para los primeros registros, se realizó modificaciones necesarias para borrar y agregar variables que se creyeron necesarias a la hora de implementar el proyecto.

6.1.1. Direcciones utilizadas

Las direcciones utilizadas para unity + ifix ponerlo despues de SCADA?

6.2. Adquisición de datos

Para realizar la estimación de la planta del sistema se utilizó el protocolo OPC en conjunto con Matlab. Por medio de OFS (OPC Factory Server, software de Schneider), se procedió a crear y configurar un servidor con la dirección correspondiente y se seleccionó el programa realizado en UnityPro donde se encontraban las variables necesarias (Figura 6.1).

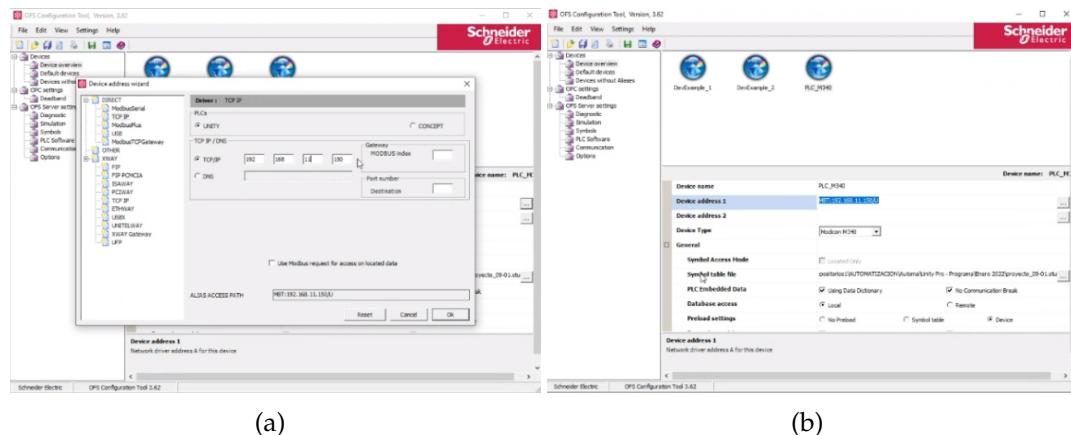


Figura 6.1: Configuración OPC

Una vez configurado el servidor se abre el programa **OPC Factory Server** dando inicio al servidor (Figura 6.2a). Para observar si la comunicación esta establecida de forma correcta, se utilizó el programa **OFS Client** dónde se debió agregar el tag correspondiente a la variable a observar (Figura 6.2b)

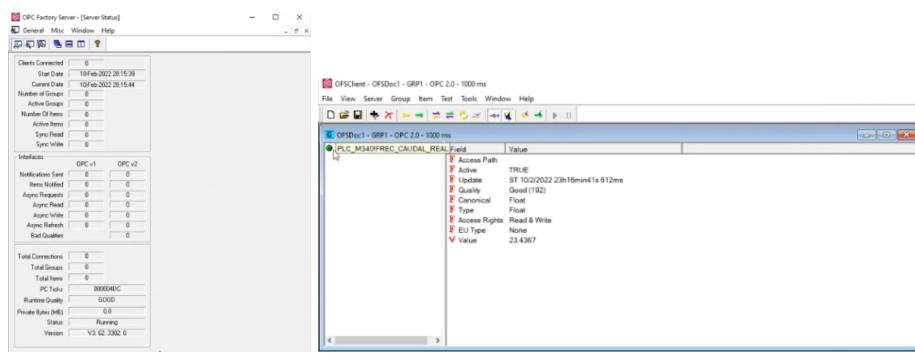


Figura 6.2: Conexión servidor OPC

	PIT001	PIT002	FT001
Ay	1.4208	13.29	4.18
Au	200	200	300
T	0.6	1.1	
\tau	1.7	2	

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.

6.2.1. 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 6.3 a) y con un bloque *Scope*, se activó la opción para que se guarden los vectores de las variables a estudiar (Figura 6.3 b).

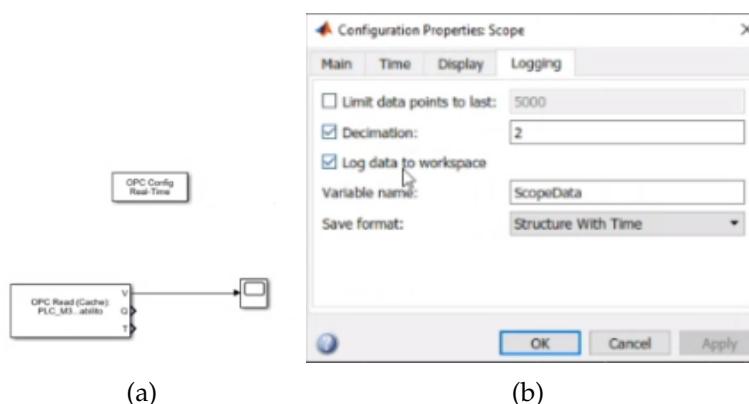


Figura 6.3: Cliente OPC en Simulink

6.2.2. Estimación de la planta

Para realizar la estimación de las plantas se utilizó el método matemático de sistema de primer orden con retardo (Figura 6.4), cuya función de transferencia es:

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

Dónde:

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

Los valores correspondientes de cada elemento para cada planta son los siguientes:

los saque de lectura_csv_vs_mod y de cvs drive

6.2.3. PID calculado

Cada PID se calculó con **Tune PID controllers**, se buscó que las respuestas sean **críticamente amortiguado**, los valores recomendados son:

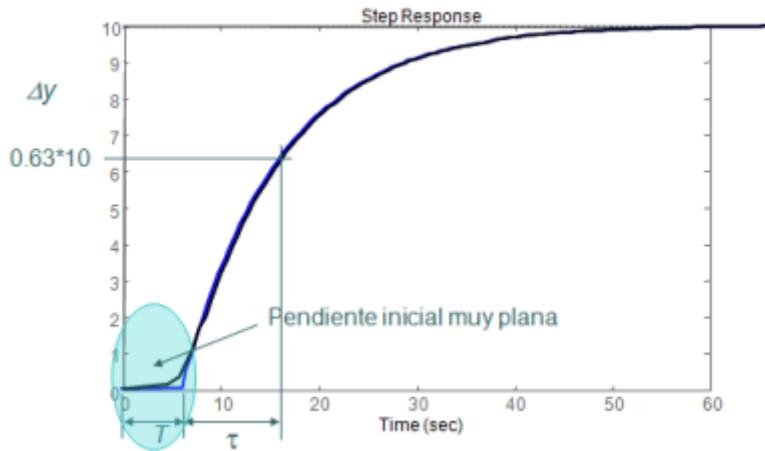


Figura 6.4: Placa con sensores

	PIT001	PIT002	FT001
Kp	???	???	???
Ki	???	???	???
Kd	???	???	???

Se puede observar que el controlador PID es del tipo proporcional integrador, se realizó de esta forma ya que la ganancia en los diferentes escalones en la planta no era constante. Al utilizar un PI genera una salida con error de estado estacionario cero.

Estuve viendo las curvas con escalones donde se ve que la ganancia estática se modifica frente a

6.3. SCADA

Para realizar la pantalla de supervisión, control y adquisición de datos operador (SCADA) se utilizó el software iFix perteneciente al grupo **General Electric**.

En la pantalla principal se creó un esquemático del circuito hidráulico físico que se posee con las respectivas variables de presión y caudal, a las cuales se sumó variables propias del motor, otra sección se destinó a las alarmas, al control de las variables y la última al Historian (Figura 6.5).

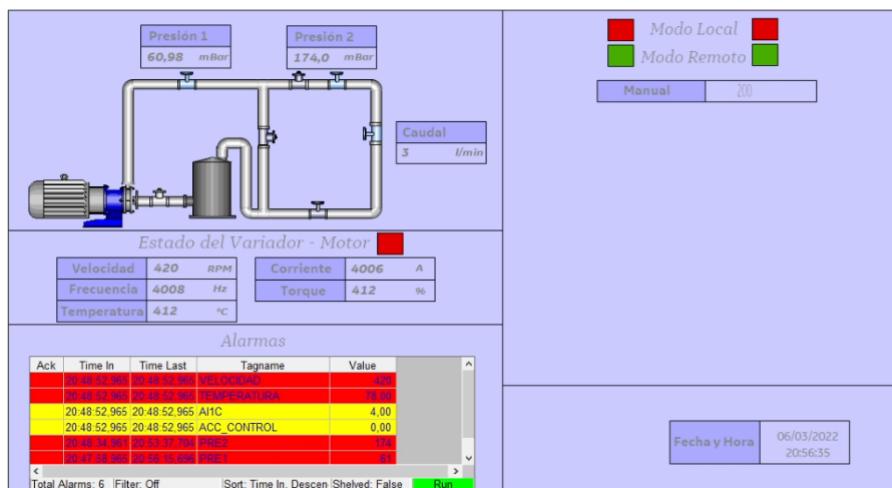


Figura 6.5: Pantalla SCADA

6.3.1. Configuración iFix o Configuracion MBE?

Para realizar la configuración de cada ícono con su respectiva variable se debió crear un MBE dónde se estipula la dirección y los mapas de memorias que luego serán utilizadas por el DataBase.

6.3.1.1. MBE

6.3.2. DataBase

6.3.2.1. Lista de direcciones utilizadas

Poner lista que esta en drive

6.3.2.2. Pruebas mediante ModSim

Para realizar pruebas previas a la implementación final se utilizó el programa ModSim, donde se generó los distintos mapas de memoria utilizados y allí se podían modificar variables para observarlas en SCADA.

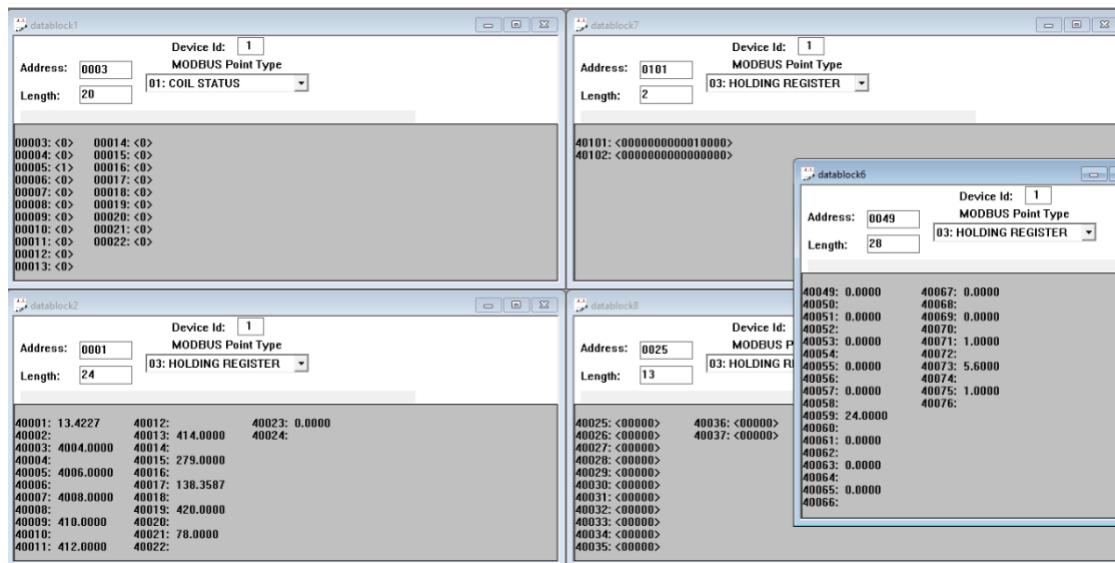


Figura 6.6: ModSim

6.3.3. Distintas pantallas

6.4. Alarmas

Dentro de la pantalla principal es posible observar el alarmero. Estas alarmas están compuestas por las variables de la siguiente **tabla** con sus respectivos valores y prioridades.

Poner lista que esta en drive

6.5. iHistorian

6.6. Direcciones

Poner acá las de ifix y unity? como la utilizada en drive?

7. Conclusiones

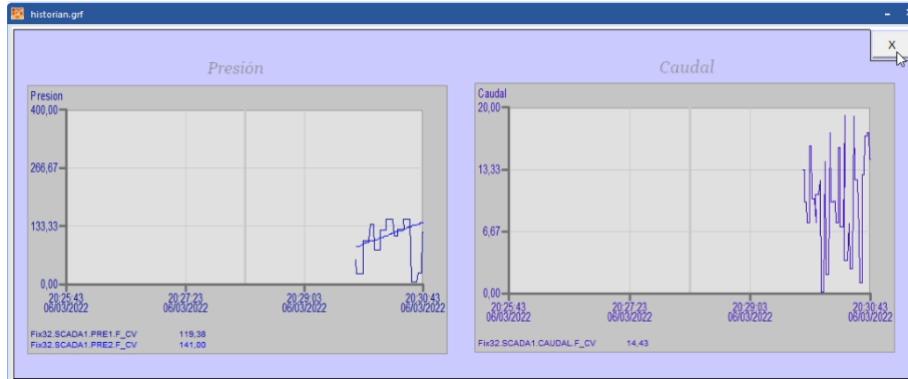


Figura 6.7: Pantalla SCADA

8. Bibliografia

Referencias

- [1] Schneider Electric. "Manual de instalación". En: *Altivar312- Variadores de velocidad para motores asincronos* (2013).
- [2] Schneider Electric. "Communication variables manual". En: *Altivar312- Variadores de velocidad para motores asincronos* (2009).

9. Anexos

10. Manual SCADA

10.1. Secciones

10.1.1. Diagrama

10.1.2. Estado del variador - motor

10.1.3. Modo local - Modo remoto

10.1.3.1. Control de presión o caudal

10.1.4. Alarmas

10.1.5. iHistorian