



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE INGENIERÍA MECATRÓNICA

MANUAL DE PRÁCTICAS DE LA PLANTA A, EMPLEANDO EL
SISTEMA CONECTADO AL PLC SCHNEIDER MODICON PREMIUM
DEL LABORATORIO DE LACTI EN LA UNIVERSIDAD POLITÉCNICA
SALESIANA SEDE CUENCA

Trabajo de titulación previo a la obtención
del título de Ingeniero Mecatrónico

AUTORES: YANDRY ISMAEL MEDINA CHILIGUANO
RICARDO ANDRÉS TORRES SERRANO
TUTOR: ING. MONICA ALEXANDRA ROMERO SACOTO, MSc.
CO-TUTOR: ING. ANGEL EUGENIO CARDENAS CADME, MSc.

Cuenca – Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Yandry Ismael Medina Chiliguano con documento de identificación N° 0104772009 y Ricardo Andres Torres Serrano con documento de identificación N° 1105078404; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 29 de julio del 2022

Atentamente,



Yandry Ismael Medina Chiliguano
0104772009



Ricardo Andres Torres Serrano
1105078404

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Yandry Ismael Medina Chiliguano con documento de identificación N° 0104772009 y Ricardo Andres Torres Serrano con documento de identificación N° 1105078404, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: "Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca", el cual ha sido desarrollado para optar por el título de: Ingeniero Mecatrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 29 de julio del 2022

Atentamente,

Yandry Ismael Medina Chiliguano
0104772009

Ricardo Andres Torres Serrano
1105078404

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Monica Alexandra Romero Sacoto con documento de identificación N° 0302209911, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, realizado por Yandry Ismael Medina Chiliguano con documento de identificación N° 0104772009 y Ricardo Andres Torres Serrano con documento de identificación N° 1105078404, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 29 de julio del 2022

Atentamente,



Ing. Monica Alexandra Romero Sacoto, MSc.

0302209911

Dedicatoria

A Dios por haberme permitido crecer profesionalmente, a mi abuelita Laura, mis padres Ximena y Jose, a mis tíos Luis, Héctor y Jenny. Que este logro sirva como punto de referencia en las metas de mis hermanos Dalton y Gabriela, recordando que: “Cuando las ganas de superarse son grandes, no hay destino que aplaque.”

Yandry

Este proyecto lo dedicó a mi familia que siempre ha sido parte importante en mi vida. En especial a mi padre Fabian Torres y madre Jazmín Serrano, que me han apoyado económica y emocionalmente. Mis abuelos y hermanos que me brindaron su apoyo desde pequeño.

Ricardo

Agradecimientos

A mis padres por el esfuerzo realizado para darme una educación de calidad.

A mi abuelita Laura por haberme apoyado incondicionalmente.

A los ingenieros que estuvieron involucrados en torno a este proyecto de titulación. En especial a la Ing. Monica Romero, el Ing. Mauricio Villacis y Ing. Eugenio Cárdenas.

Finalmente quiero agradecer de todo corazón a la empresa Ecuacorriente S.A, por el apoyo brindado.

Yandry

A mi papá y mamá por todo el apoyo brindado. Aunque me tocó separarme de ellos siempre estuvieron ahí para cualquier momento difícil que ocurra.

A la Ing. Monica que aportó como tutora.

Al Ing. Mauricio Villacis por toda la ayuda y el conocimiento dado a lo largo del desarrollo de la tesis.

Ricardo

Este documento fue realizado enteramente en L^AT_EX

Índice

Certificado de responsabilidad y autoría del trabajo de titulación	I
Certificado de cesión de derechos de autor del trabajo de titulación a la Universidad Politécnica Salesiana	II
Certificado de dirección del trabajo de titulación	III
Dedicatoria	IV
Agradecimientos	V
Resumen	XIII
Abstract	XIV
1. Introducción	1
2. Problema	1
2.1. Antecedentes	1
2.2. Descripción del problema	1
2.3. Importancia y alcances	2
2.4. Delimitación	2
2.4.1. Espacial o geográfica	2
2.4.2. Temporal	2
2.4.3. Sectorial o institucional	2
2.5. Problema General	3
2.6. Problemas Específicos	3
3. Objetivos	4
3.1. Objetivo General	4
3.2. Objetivos Específicos	4
4. Hipótesis	4
4.1. Hipótesis General	4
4.2. Hipótesis Específicas	5

5. Marco Teórico	5
5.1. Laboratorio LACTI planta A	5
5.2. PLC Schneider	6
5.3. Protocolo de comunicación Modbus	7
5.3.1. Modbus RTU	7
5.3.2. Modbus TCP/IP	8
5.4. Protocolo de comunicación HART	9
5.5. Software para control y monitoreo de la planta	10
5.5.1. Unity Pro XL	10
5.5.2. Ubidots	10
5.5.3. Node RED	11
5.6. SIMATIC IOT 2040	11
5.7. Sistema de control	12
5.7.1. Control PID	13
5.8. Generación del modelo matemático del sistema	14
5.8.1. Método de Smith	15
5.9. Señal PRBS para la identificación del sistema	16
6. Marco metodológico	17
6.1. Variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A	17
6.1.1. Procesos	17
6.1.2. Subprocesos	18
6.1.3. Variables	18
6.1.4. Protocolos de comunicación	19
6.1.5. Instrumentos	20
6.2. Control y monitoreo de la Planta A	22
6.2.1. Planteamiento inicial	22
6.2.2. Red Modbus para conmutar el PLC con el concentrador Hart	23
6.2.3. Red general	25
6.2.4. Control y monitoreo de la planta por medio de la red (IOT)	25
6.3. Algoritmo de control (PID)	35
6.3.1. Identificación del sistema	36
6.3.2. Consideraciones de la señal de entrada	38
6.3.3. Tiempo de muestra	39

6.3.4.	Obtención de datos del sistema	42
6.3.5.	Identificación de la función de transferencia	44
6.3.6.	Control proporcional, integral y derivativo (PID)	46
6.3.7.	Implementación del control PID	55
6.4.	Manual de prácticas y guías de aprendizaje	58
7.	Resultados	60
7.1.	Variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A	60
7.1.1.	Variables	60
7.1.2.	Proceso	60
7.1.3.	Instrumentos	60
7.2.	Control y monitoreo de la Planta A	64
7.2.1.	Pantalla HMI	64
7.2.2.	Ubidots	66
7.3.	Algoritmo de control (PID)	69
7.4.	Manual de prácticas y las guías de aprendizaje de la planta A	73
8.	Conclusiones	73
9.	Recomendaciones	74
	Referencias	77
	ANEXOS	78

Lista de Tablas

- 1. Trama de datos de Modbus RTU 8
- 2. Trama de datos de Modbus TCP/IP 8
- 3. Constantes para el método de Smith 16
- 4. Sensores de la planta A del laboratorio de LACTI 20
- 5. Actuadores de la planta A del laboratorio de LACTI 21
- 6. Parámetros PARA (PIDFF) 57
- 7. Variables introducidas en el bloque PIDFF 58
- 8. Salidas digitales 61
- 9. Entradas digitales 62
- 10. Salidas analógicas 63
- 11. Entradas analógicas 63
- 12. Matriz de consistencia lógica. 80

Lista de Figuras

1.	Universidad Politécnica Salesiana	3
2.	LACTI	6
3.	Señal HART	9
4.	Trama de datos de HART	10
5.	Tablero Ubidots	11
6.	IOT 2040	12
7.	Sistema de control en lazo cerrado	13
8.	Control PID	14
9.	Método de la curva de Smith	15
10.	Señal PRBS	17
11.	Planta A del laboratorio de LACTI	19
12.	Red para obtener los datos del HCS	23
13.	Implementación del HMI en la planta	24
14.	Red general	25
15.	Configuración de IP del PLC Schneider	26
16.	Configuración de las variables	27
17.	Variables monitoreadas por KEPServerEX	28
18.	Librerías Node-red	28
19.	Instalar librería ubidots	29
20.	Parámetros Modbus	29
21.	Parámetros conexión Modbus	30
22.	Parámetros conexión Ubidots	31
23.	Dispositivos en Ubidots	32
24.	Crear dispositivo nuevo en Ubidots	32
25.	Nombre de variable creada	33
26.	Parámetros de bloque Ubidots In	33
27.	Conversión de variables para envío de datos en Ubidots	34
28.	Conversión de variables para recepción de datos en Ubidots	34
29.	Lazo de control	35
30.	Identificación del sistema	37
31.	Modelo ARX	37
32.	Dinámica del sistema	39
33.	Respuesta de lazo abierto (Smith)	40

34.	Respuesta de lazo abierto/Función de transferencia estimada	41
35.	Obtención de datos del sistema	42
36.	Panel de control (LabView)	43
37.	Respuesta del sistema/Señal PRBS	44
38.	Validación de la función de transferencia	45
39.	Configuración de la Toolbox Ident Matlab	46
40.	Respuesta en lazo cerrado de la función de transferencia	47
41.	Lugar de raíces de la función polinómica	49
42.	Aproximación de la función de transferencia a una función polinómica	50
43.	Lugar de raíces del sistema e inclusión del polo y cero	51
44.	Lugar de raíces de la relocalización de los polos y del cero	52
45.	Respuesta del sistema con el controlador	55
46.	Bloque PIDFF	56
47.	Implementación del bloque PIDFF	58
48.	Pantallas de inicio del HMI	64
49.	Pantalla PID en el HMI	65
50.	Pantalla para el control manual en el HMI	66
51.	Menú principal de Ubidots	67
52.	Ventana PID en Ubidots	67
53.	Ventana del control manual en Ubidots	68
54.	Ventana HCS en Ubidots	69
55.	Respuesta del sistema controlado con valor de referencia de 30 cm.	70
56.	Respuesta del sistema controlado con valor de referencia de 40 cm.	70
57.	Respuesta del sistema controlado con valor de referencia de 50 cm.	71
58.	Tiempo de asentamiento y tiempo de levantamiento del sistema.	72
59.	Respuesta del sistema controlado, frente a una perturbación.	72

Resumen

El presente trabajo aborda el manejo y monitoreo de procesos de la planta A del laboratorio LACTI de la Universidad Politécnica Salesiana, usando el PLC Schneider. La idea surge debido a que la universidad adquirió el PLC cuando se implementó la planta A, sin embargo, no se le ha dado el uso apropiado. Por tal motivo, el siguiente proyecto busca realizar guías prácticas de control y monitoreo con el fin de ampliar los conocimientos de los estudiantes de la Universidad Politécnica Salesiana, y restablecer el aporte del autómatas programable.

Para el desarrollo de este proyecto se partió del análisis de variables, procesos, protocolos e instrumentos de la planta, para aprender sobre el funcionamiento de la misma. Después se realizaron dos interfaces para monitorear las variables del proceso: uno mediante un HMI y otro por Ubidots (IOT). Para la interfaz con Ubidots se generó una red entre el PLC, módulo IOT2040 y el router con el que cuenta la planta, se usa los protocolos ethernet y Modbus TCP/IP. La pantalla HMI es utilizada como una pasarela entre el PLC y el concentrador HCS, con el fin de obtener los datos del protocolo de comunicación Hart, además, se diseñó un entorno gráfico en el software KincoDTools, programa de la propia marca, para el control y monitoreo.

Por otra parte, se desarrolló un control PID del tanque abierto, en el cual se aplicó el método ARX y mínimos cuadrados recursivos para obtener una función de primer orden que represente la dinámica del sistema; por medio de la aproximación PADE se transformó a una función de segundo orden, y por asignación de polos se obtuvieron las ganancias proporcional, integral y derivativa. Mismas que fueron implementadas en el PLC Schneider.

A partir de estos pasos se realizaron cinco guías prácticas orientadas a los estudiantes para vincular los conocimientos adquiridos en clase con un entorno laboral.

Palabras clave: LACTI, Planta A, PID, Control de nivel, Guías de prácticas.

Abstract

The present work deals with the management and monitoring of processes of plant A of the LACTI laboratory of the Salesian Polytechnic University using the Schneider PLC. The idea arises because the university acquired the PLC when plant A was implemented, however, it has not been used, limiting the processes carried out in the plant. For this reason, the following project seeks to carry out practical control and monitoring guides in order to expand the knowledge of the students of the Salesian Polytechnic University.

For the development of this project, we started from the analysis of variables, processes, protocols and instruments of the plant, to learn about its operation. Afterwards, two interfaces were made to monitor the variables of the plant: one through an HMI and the other through Ubidots. For the interface with Ubidots, a network was completed between the PLC, the IOT2040 module and the plant's router using the Ethernet and Modbus TCP/IP protocols. For the interface with the HMI, the HMI and the HART Concentrator System were connected via Modbus RTU to read the plant variables. KincoDTools software, a proprietary program, was used to program the HMI, and it was connected to the Schneider PLC via Modbus TCP/IP.

For the PID control of the open tank, the ARX method and recursive least squares are used to obtain a first order equation, through the approximation it was transformed into a second order function; and by estimating the poles, the proportional, integral and derivative gains will be acquired for the control. Same that were implemented in the Schneider PLC using the control block.

From these steps, five practical guides aimed at students were made to link the knowledge acquired in class with a work environment.

Keywords: Plant A, PID, Level control, Practice guides.

1. Introducción

La Universidad Politécnica Salesiana en el compromiso de la enseñanza y preparación de futuros profesionales, ha creado laboratorios en los cuales los estudiantes pueden involucrarse y asimilar las aplicaciones prácticas con los conceptos teóricos, tratando de asemejar un entorno de trabajo real. Es así que en el año 2010 la Universidad Politécnica Salesiana sede Cuenca conjuntamente en convenio con el Ministerio de Industrias y Producción, crea el laboratorio LACTI (Laboratorio de Control y Automatización Totalmente Integrado), con el objetivo de generar una vinculación entre el campo industrial y el sector académico (Universidad Politécnica Salesiana, 2013).

A lo largo de estos últimos 11 años se ha producido diferentes proyectos de investigación en las dos plantas que contiene el laboratorio LACTI, con la integración de carreras técnicas como: Mecánica, Mecatrónica, Electrónica y Telecomunicaciones. En el ejercicio de prácticas en las áreas de instrumentación, PLC's, control, automatización, entre otros. Sin embargo el PLC Schneider dentro de la planta A, se mantiene fuera de servicio debido a que no se tiene un uso específico; además, se desconoce como manipular el autómatas programable y lo que se mantiene conectado a sus puertos. De tal manera, el presente trabajo de titulación se enfoca en rehabilitar el uso del PLC para poder realizar guías de prácticas que ayuden a la preparación y generación de conocimientos en los estudiantes.

2. Problema

2.1. Antecedentes

Enfocados principalmente en el proceso completo que dispone el laboratorio (planta A), se evidencia que se realiza el control con los autómatas programables de las marcas comerciales Siemens y Schneider. En general, los proyectos realizados se han trabajado únicamente con el PLC Siemens; y con el tiempo se ha dejado de lado el PLC Schneider produciendo paulatinamente que no exista documentación para el uso efectivo del autómatas en la planta.

2.2. Descripción del problema

El proceso de control en la planta A del laboratorio LACTI, puede ser comandado por dos PLC's, Siemens o Schneider. En la actualidad, el manejo de la sección A se ejecuta únicamente bajo los programas desarrollados en la marca Siemens, debido a que no existe

la documentación donde se especifique el modo de uso del autómata Schneider. Siendo la utilidad del segundo PLC casi nula y al no contar con un esquema de conexión con la planta, no se puede operar, ni manipular un proceso; a pesar de que las variables que intervienen en los sistemas de control están conectadas a dicho equipo.

2.3. Importancia y alcances

El aporte del presente proyecto se focaliza principalmente en la comunidad estudiantil de la Universidad Politécnica Salesiana sede Cuenca, donde se pretende dar un uso efectivo a los equipos con los que cuenta el laboratorio LACTI. Además, se dispondrá a maestros y estudiantes la documentación necesaria para realizar diferentes propuestas de prácticas e indagar nuevas aplicaciones en la línea de proceso, en el ejercicio de vincular los conocimientos con un entorno profesional.

2.4. Delimitación

2.4.1. Espacial o geográfica

El proyecto se va a desarrollar en la Universidad Politécnica Salesiana. Ubicada en la ciudad de Cuenca, provincia del Azuay, Calle Vieja 12-30 y Elia Liut (Figura 1). Específicamente, se llevará a cabo el control de la planta A del laboratorio de LACTI en el área de mecánica.

2.4.2. Temporal

El periodo de tiempo seleccionado para la investigación y desarrollo del proyecto es de ocho meses.

2.4.3. Sectorial o institucional

La Universidad Politécnica Salesiana forma parte de un sector educativo, más aún, el proyecto apunta a un área de ciencia y tecnología para el desarrollo del conocimiento en la sección industrial.

Figura 1

Universidad Politécnica Salesiana



Nota: Se puede observar la ubicación de la Universidad Politécnica Salesiana, sede Cuenca. Adaptado de Universidad Politécnica Salesiana, Google (2022). © 2022 Google.

2.5. Problema General

- ¿Es idóneo elaborar las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium?

2.6. Problemas Específicos

- ¿Es posible identificar las variables, procesos, subprocessos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje?
- ¿Es idóneo realizar control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómatas programables Schneider; para las guías de aprendizaje y el manual de prácticas?
- ¿Se podrá diseñar e implementar en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje?

- ¿Es factible generar el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI?

3. Objetivos

3.1. Objetivo General

- Elaborar las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium.

3.2. Objetivos Específicos

- Identificar las variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje.
- Realizar control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómata programable Schneider; para las guías de aprendizaje y el manual de prácticas.
- Diseñar e implementar en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje.
- Generar el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI.

4. Hipótesis

4.1. Hipótesis General

- Se elaborará las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium.

4.2. Hipótesis Específicas

- Se identificará las variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje.
- Se realizará control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómata programable Schneider; para las guías de aprendizaje y el manual de prácticas.
- Se diseñará e implementará en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje.
- Se generará el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI.

5. Marco Teórico

5.1. Laboratorio LACTI planta A

La Universidad Politécnica Salesiana para la integración de las carreras técnicas con el campo laboral, ha implementado el laboratorio LACTI (Laboratorio de Control y Automatización Industrial) como un ambiente industrial donde los alumnos pueden simular procesos de producción. Esta sección dispone de la planta A (Véase la figura 1), que es un sistema de control de procesos donde se puede comparar las diferentes tecnologías de buses de datos industriales, analizar los niveles de automatización, integrar el diseño mecánico y principalmente controlar variables de: presión, temperatura, nivel y caudal. La planta está compuesta por válvulas proporcionales, sistema de calentamiento de agua utilizando la caldera o el calefón, intercambiador de calor, sistema de bombeo (bombas multi-etapa y un depósito), dos tanques de acero inoxidable (uno para la presión controlada y el otro para la presión atmosférica), panel de control con el acceso de cada señal digital y analógica; toda la ejecución del proceso de planta puede ser manipulado por dos autómatas programables de gama media alta de las marcas Siemens (PLC S7-300) y Schneider (PLC Modicom Premium); además cuenta con un controlador industrial JUMO y un registrador industrial de señal JUMO (Escobar y Domínguez, 2015).

Cabe mencionar que el proceso incorpora instrumentos industriales de medición de presión

basado en el principio piezo – resistivo. Instrumentos para la toma de datos de temperatura basado en principios RTD y resistivo. Equipos de medición de caudal fundados en los principios de presión diferencial, Coriolis, turbina y electromagnética. Finalmente, contiene elementos para la medición de nivel fundamentados en los principios capacitivos, ultrasónicos, radar de onda guiada y presión hidrostática. (Escobar y Domínguez, 2015).

Por otra parte, los protocolos de comunicación con los que cuenta la planta son: HART, Profibus Dp/PA, Modbus, Fieldbus Foundation y IWLAN (inalámbrica industrial). Para poder tener una mayor interacción entre el proceso y el operario, se dispone de sistemas HMI, OPC'S y SCADA (Escobar y Domínguez, 2015).

Figura 2

LACTI



Nota: Se puede visualizar una sección de la Planta A, del laboratorio de control y automatización industrial.

5.2. PLC Schneider

El autómata programable que mantiene la planta A del laboratorio de control y automatización industrial, es el PLC Schneider, su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente. Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento,

como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos por lo general son para aplicaciones específicas de seguridad, contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos (Schneider Electric, 2005).

5.3. Protocolo de comunicación Modbus

Modbus es un protocolo de comunicación del tipo solicitud – respuesta, ubicado en el séptimo nivel del modelo OSI, con una arquitectura maestro – esclavo (RTU) o cliente – servidor (TCP/IP) entre los componentes conectados en distintos tipos de redes o buses. La relación maestro – esclavo, propone trabajar en pares, donde un equipo debe de iniciar con la solicitud y el otro dar una respuesta; el dispositivo que da la iniciación (maestro) es el responsable de dar comienzo a cada interacción. Dentro de una conexión Modbus estándar se puede disponer de un maestro y hasta 247 esclavos, con una transferencia de datos de 1200 metros y una dirección única para cada dispositivo; además se mantiene dos protocolos, RTU que usa comunicación en serie RS485/RS232/RS422 y TCP/IP que utiliza comunicación Ethernet (Modbus, 2006) (National Instruments, 2019).

5.3.1. Modbus RTU

Modbus Remote Terminal Unit (RTU), tiene la característica de enviar los bytes en codificación binaria plana, sin realizar ninguna conversión. Consta de una comunicación de bus en serie, mejorando el canal de comunicación y la velocidad de transmisión. Su delimitación radica en que se debe de gestionar tiempos entre los bytes recibidos, con el fin de dar una idea de cuando empieza y termina la trama. La Tabla 1 muestra la trama de comunicación, misma que para detectar errores utiliza un código de 16 bytes Cyclical Redundancy Check (CRC). Donde el emisor calcula el código a partir de los bytes de la trama, luego compara con el cálculo de la combinación del receptor, y si el valor obtenido es similar se da paso a la validación de los datos (Candelas, 2011).

Tabla 1*Trama de datos de Modbus RTU*

Inicio	Dirección esclavo	Función	Datos	CRC	Final
Mínimo de 3.5 caracteres de silencio	1 byte	1 byte	N bytes depende de la función	2 bytes	Mínimo de 3.5 caracteres de silencio

Nota: La tabla presenta el formato de la trama del protocolo de comunicación Modbus RTU. Extraído de Comunicación con RS-485 y Modbus, Candelas (2011). p, 21. © 2011 Universidad de Alicante.

5.3.2. Modbus TCP/IP

TCP/IP es un protocolo de internet y control de transferencia de datos, provee el medio físico para la emisión de mensajes Modbus TCP/IP. El servicio TCP, asegura que los paquetes se reciban sin error o problema de información, mientras el IP mantiene la funcionalidad de asegurar que sean abordados los mensajes y colocados correctamente. Cabe mencionar, que se permite una conexión con la disponibilidad de ser reconocida, supervisada y cancelada, sin que el servidor o cliente tome una decisión específica, permitiendo al protocolo tener una gran tolerancia frente a los cambios del rendimiento de red (Bouchakri, 2021).

Tabla 2*Trama de datos de Modbus TCP/IP*

MBAP	Función	Datos
7	1	n

Nota: Se puede observar el formato de la trama para la comunicación Modbus TCP/IP. Extraído de Diseño e implementación de una aplicación de control de proceso industrial sobre pantalla HMI, Bouchakri (2021). p, 22. © 2021 Universidad Politécnica de Catalunya.

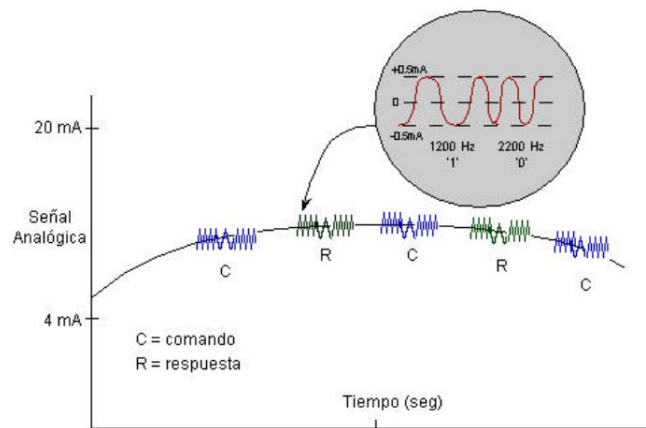
Dentro de la trama de comunicación Tabla 2, el encabezado Modbus Application Protocol (MBAP) dispone de un tamaño de 7 bytes; divididos en 2 bytes de sincronización de mensaje entre cliente y servidor, 2 bytes para referenciar el protocolo Modbus, 2 bytes para la cantidad de trama y 1 bytes para la identificación remota del esclavo. Por otra parte, la función indica el código a utilizar y tiene una longitud de 2 bytes. Finalmente, los datos proveen la información de solicitud al maestro, la longitud dependerá del tamaño del mensaje (Bouchakri, 2021).

5.4. Protocolo de comunicación HART

Protocolo de comunicación del tipo maestro - esclavo; combina información analógica y digital en un solo cableado por medio de la técnica de codificación de frecuencia (SFK), mismo que propone sobreponer comunicación digital en el lazo de corriente de 4 a 20 mA del sistema de control que conecta con el dispositivo de campo. Se considera dos frecuencias de 1200 Hz o 2200 Hz, para representar un 0 y 1 binario (Figura 3). La señal DC se sobrepone a la toma de datos a un bajo nivel; mientras que, el AC toma valores promedios de cero, produciendo que no exista ningún cambio DC en la señal de 4 a 20 mA. Es decir, el instrumento puede utilizar la señal analógica para el control y la señal digital en la adquisición de datos (Grumstrup, s.f.).

Figura 3

Señal HART.

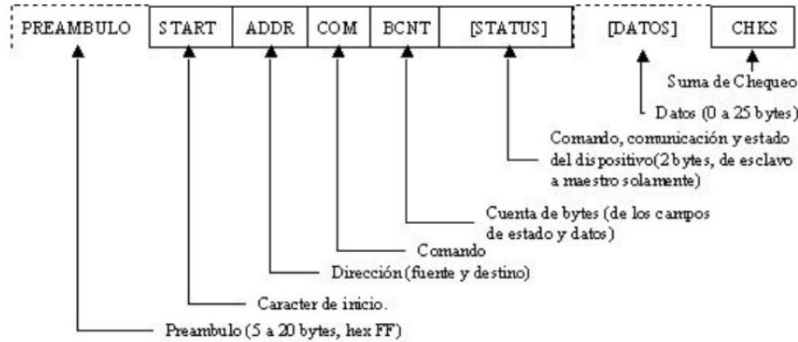


Nota: En la figura se muestra el comportamiento de la señal HART. Extraído de Protocolo de comunicación HART para los instrumentos de Flotech s.a, Febres (2001). p, 09. © 2001 Universidad Simón Bolívar.

Cabe recalcar, que para asegurar la información se cuenta con la dirección de destino y fuente para cada mensaje; la suma de verificación denominado Checksum ayudara a detectar cualquier error en la comunicación. El estado de cada instrumento de campo estará incluido en la respuesta, indicando su óptima operabilidad (Febres, 2001).

Figura 4

Trama de datos de HART



Nota: En la presente figura se indica la estructura del mensaje del protocolo Hart. Extraído de Protocolo de comunicación HART para los instrumentos de Flotech s.a, Febres (2001). p, 09. © 2001 Universidad Simón Bolívar.

5.5. Software para control y monitoreo de la planta

5.5.1. Unity Pro XL

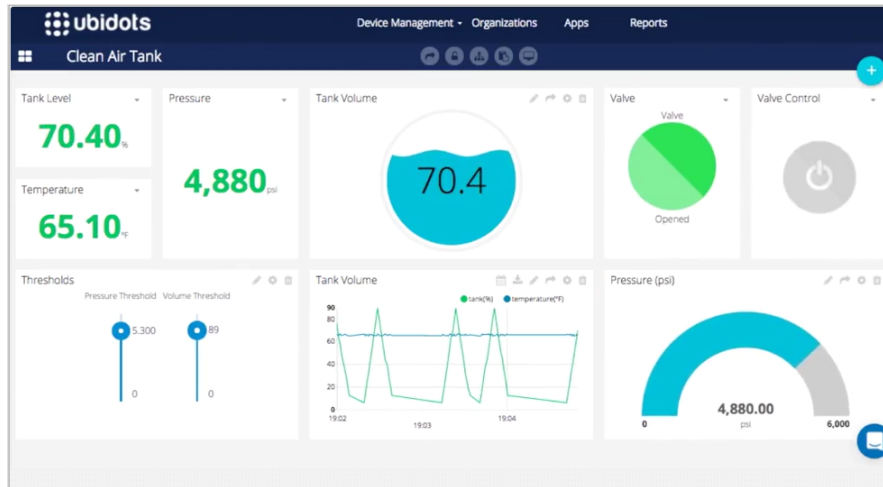
Unity Pro es un software de programación desarrollado por la marca Telemecanique, utilizado para la puesta a punto de los autómatas programables de la gama media alta de la plataforma Modicon. Cuenta con cinco lenguajes estándar de programación: lista de instrucciones (IL), diagrama de escalera (LD), tabla de funciones secuenciales (SFC), texto estructurado (ST) y diagrama de bloques de función (FBD). El software es presentado por Schneider como versátil y de fácil uso, ya que dispone de bloques de función para una óptima estandarización y elevado diseño funcional que proporcionen una mayor productividad en el proceso (Schneider Electric, 2012).

5.5.2. Ubidots

Ubidots es una plataforma de internet que permite realizar un monitoreo y control en tiempo real, dando la posibilidad de tomar decisiones oportunas para una mayor productividad del sistema. Las principales ventajas de esta herramienta, radican en que sin tener un equipo de desarrollo de software o sin escribir un código de programación se pueda ensamblar y ejecutar una aplicación del internet de las cosas (IOT) (Ubidots, 2022).

Figura 5

Tablero Ubidots



Nota: En la imagen se puede observar un ejemplo del entorno de control en la plataforma Ubidots. Extraído de Ubidots, Ubidots (2022). © 2022 Ubidots.

5.5.3. Node RED

Node RED se trata de una herramienta de programación que posibilita conectar servicios en línea, dispositivos hardware y API. Proporciona una forma de programación visual debido a su editor de flujo, facilitando la interconexión de los procesos por medio de una amplia biblioteca de nodos. Esta plataforma se ha convertido en un modelo open – source, es un mediador entre la gestión y proceso para disponer datos en tiempo real en un entorno IOT (Nodo - RED, 2022).

5.6. SIMATIC IOT 2040

El dispositivo IOT 2040 de la marca comercial Siemens (Figura 6), se define como una pasarela inteligente diseñado para la adquisición de información entre varias fuentes de datos, posteriormente su procesamiento y su envío a los correspondientes receptores. La necesidad del uso de datos combinados de máquinas y producción abre una amplia utilidad al dispositivo, ya que se dispondrá como puerta de enlace en el proceso y la nube/TI. El equipo soporta distintos protocolos de comunicación y lenguajes de alto nivel, siendo una solución rentable y de fácil uso en el propósito de controlar y monitorear variables de campo. Además, las ventajas apuntan en tener opciones de conexión flexibles, respuestas de rendimiento y deterministas, acompañado

de un grado de robustez de un funcionamiento continuo en aplicaciones industriales (Siemens, 2022).

Figura 6

IOT 2040



Nota: Se puede observar el módulo IOT 2040 instalado en la planta B del laboratorio LACTI.

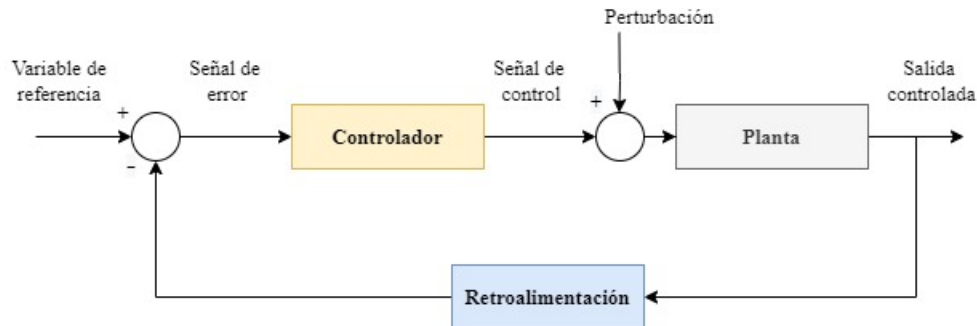
5.7. Sistema de control

Para Bouchakri (2021) un sistema de control se define como “la combinación de elementos que actúan sobre una planta o proceso, tratando de fijar o variar alguno de sus parámetros, en el transcurso del tiempo.” Es decir, el objetivo del controlador es manipular la planta en búsqueda de un valor de estabilidad con respecto a la referencia, sin que un operador tenga que intervenir en el proceso.

Un sistema de control se propone en dos escenarios, lazo cerrado y lazo abierto, siendo el primero el más eficiente y permitiendo mitigar el error del sistema; debido a que dispone de una retroalimentación para generar una señal de error entre la diferencia de la entrada y la salida, misma que será procesada por el controlador a fin de obtener una señal de control, posteriormente se aplica a la planta y se obtiene la respuesta de la variable a manipular (Figura 7).

Figura 7

Sistema de control en lazo cerrado



Nota: En la figura se muestra el diagrama de un sistema de control en lazo cerrado.

Según Alberto, Perez, y Perez (2007), un esquema general de control esta integrado por los siguientes elementos:

- **Variable de referencia:** Es la entrada del sistema, tiene la función de ser un parámetro de ajuste al cual se va a regir el lazo de control.
- **Planta:** Se define como planta a cualquier sistema físico que se pueda controlar. Es representado por una formulación matemática que mantiene toda la dinámica del sistema.
- **Controlador:** Es el encargado de compensar el valor de la salida del sistema para llevarlo al estado deseado.
- **Variable controlada:** Es la respuesta obtenida del sistema controlado.
- **Retroalimentación:** Acción de control que permite comparar la salida del sistema, con la consigna de entrada de referencia a la cual se quiere que llegue el sistema.
- **Perturbación:** Es un valor que tiende a afectar de una manera adversa la variable de salida del sistema. Su influencia implica la interacción del control.

5.7.1. Control PID

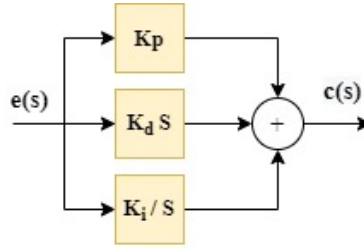
Un algoritmo PID es una herramienta que permite controlar un sistema en lazo cerrado para obtener un valor deseado en la salida. Un PID clásico es compuesto por tres bloques de ganancias: proporcional, integra y derivativo. Su principal característica está ligada a que se

puede emplear como un controlador ideal, combinando las ganancias según el sistema que se requiere controlar.

$$G_c(s) = K_P + K_d s + \frac{K_i}{s} \quad (1)$$

Figura 8

Control PID



Nota: En la figura se puede visualizar la estructura de un controlador PID.

Para (Carpio, 2013) el bloque de control PID se comporta de la siguiente manera:

- **Ganancia proporcional:** Su función se concentra en amplificar o atenuar la señal de control, tratando de compensar la señal de error, penalizando a la salida del sistema al alejarse del valor de referencia.
- **Ganancia derivativa:** Con el uso de este bloque se puede obtener la señal derivada del error, misma que permite proyectar la tendencia del sistema, para paulatinamente incorporar una acción anticipativa del controlador buscando ajustar el tiempo de respuesta y el sobrepaso máximo del proceso.
- **Ganancia integral:** Este bloque tiene el objetivo de variar la señal de control utilizando la integración como una memoria de la señal de error, da la posibilidad de minimizar o en ciertos casos eliminar en el sistema el error en estado estable, con el único defecto de producir una acción lenta.

5.8. Generación del modelo matemático del sistema

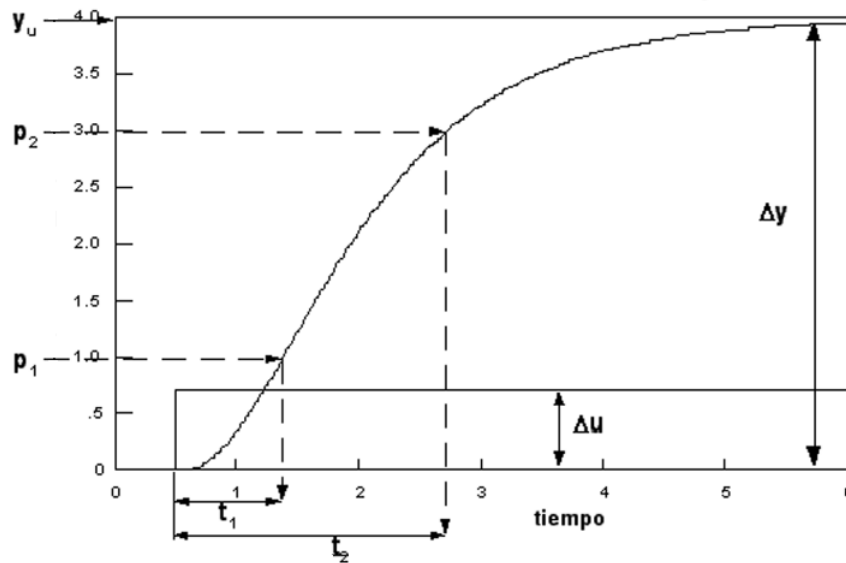
Dentro de la industria, varios de los procesos tienen dificultad para plantear una función de transferencia que represente dinámicamente su comportamiento, ya sea por la complejidad del sistema, por no contar con todas las variables o por mantener muchos parámetros. Por tal motivo, se recurre a caracterizar el sistema con métodos experimentales a partir de un estímulo tipo escalón.

5.8.1. Método de Smith

Sabiendo que un sistema de primer orden es representado por la ecuación (2). Smith propone utilizar dos puntos de la curva formada por el comportamiento del sistema, específicamente en los instantes de tiempo donde el proceso alcanza el 28.3% (t_{28}) y el 63.2% (t_{63}) del valor final.

Figura 9

Método de la curva de Smith



Nota: La curva normalizada de Smith, donde (p_1) será el 28.3% y (p_2) el 63.2%; el (Δu) pertenecerá al valor escalón que ingresa al sistema y (Δy) será el valor final de salida. Adaptado de Identificación de procesos sobreamortiguados utilizando técnicas de lazo abierto, Alfaro (2001). p, 16. © 2001 Universidad de Costa Rica.

$$G_p(s) = \frac{k e^{-t_m s}}{(\tau s + 1)} \quad (2)$$

De tal manera, los parámetros para una función de primer orden bajo retardo, se pueden obtener con:

$$k = \frac{\Delta y}{\Delta u} \quad (3)$$

$$\tau = at_1 + bt_2 \quad (4)$$

$$t_m = ct_1 + dt_2 \quad (5)$$

Los valores en las ecuaciones estarán ligados a la tabla 3.

Tabla 3

Constantes para el método de Smith

Método	% p_1 (t_1)	% p_2 (t_2)	A	B	C	D
Smith	28.3	63.2	-1.5	1.5	1.5	-0.5

Nota: La tabla muestra las constantes del método de Smith, para la identificación del modelo de primer orden más tiempo de retardo. Adaptado de Identificación de procesos sobreamortiguados utilizando técnicas de lazo abierto, Alfaro (2001). p, 16. © 2001 Universidad de Costa Rica.

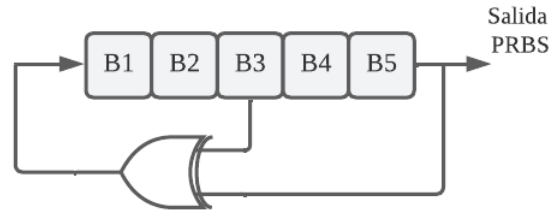
5.9. Señal PRBS para la identificación del sistema

Dentro de la identificación de un modelo dinámico del proceso, a partir de datos experimentales se debe elegir una señal de extinción con una trama rica en el espectro frecuencial, de manera que la salida refleje el comportamiento de la planta frente a ese amplio espectro. La opción ideal para la obtención de mediciones es una señal de ruido blanco, pero al tener amplitudes diversas se corre el riesgo de llegar a lesionar el sistema. Por lo cual Minchala y Valdez (2010), proponen utilizar una señal binaria pseudoaleatoria denominada Pseudo Randomic Binary Sequence (PRBS), que asemeja su contenido frecuencial en gran medida a un ruido blanco.

La definición de pseudoaleatoria se debe a que la señal PRBS dispone de una longitud definida donde los pulsos presentan una variación aparentemente aleatoria; sin embargo, en una distancia de tiempo largo los pulsos se tornaran periódicos (Carpio, 2013). Esta señal discreta de ruido blanco es generada a partir de N bit de un registro de corrimiento, y una retroalimentación por parte de una compuesta lógica XOR que acoge un dato de salida y un valor anterior para tratar de impedir repetición en la secuencia.

Figura 10

Señal PRBS



Nota: La figura muestra la idea en la que se basa la señal PRBS. Se ejemplifica una prueba con cinco bits. Adaptado de Diseño e implementación de un controlador difuso tolerante a falla, aplicado al control de nivel de líquido, Criollo (2015). p, 18. © 2015 Universidad Politécnica Salesiana.

6. Marco metodológico

6.1. Variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A

6.1.1. Procesos

El proceso en el que se fundamenta la planta A del laboratorio de LACTI, es el control de llenado de tanques, manteniendo un contenedor atmosférico y uno presurizado de 125 cm de altura, donde se puede contar con un flujo de entrada de agua por parte de tres líneas, cada una manejada por diferentes protocolos de comunicación. En la salida del líquido se mantiene la disponibilidad de trabajar con una interconexión entre los dos tanques, o con un desfogue directo hacia el depósito, mismo que servirá para recolectar y paulatinamente volver a introducir líquido al sistema.

Para el llenado de los depósitos se tiene la disponibilidad de trabajar con agua a una temperatura ambiente o caliente, que es enviada a la planta a partir de una bomba manipulada por un variador de frecuencia. El control y monitoreo del proceso se lo puede realizar por medio del PLC Schneider, Siemens o por el panel de control. Tanto los tanques, como también, cada sección de la tubería cuentan con transmisores y transductores de temperatura, nivel, presión y caudal.

6.1.2. Subprocesos

El subproceso que maneja la planta A, está en torno a la línea del protocolo de comunicación Profibus, ya que da la opción de trabajar con agua caliente en los tanques. Para la variación de la temperatura del líquido se lo realiza por medio de calefones que están subconectados a la tubería, o utilizando una caldera para generar vapor que ingrese al intercambiador de calor, mezclándose con el agua ambiente, y aumente paulatinamente la temperatura del flujo de entrada.

6.1.3. Variables

Las variables que se encuentran presentes dentro del laboratorio de LACTI a partir de sus procesos y subprocesos son las siguientes:

- a) **Nivel:** Es la principal variable a considerar dentro del laboratorio, ya que en torno a ella se realiza todo el proceso del control de líquido de los tanques. Con esto se mantiene una retroalimentación del tanque abierto y cerrado, sabiendo con exactitud a cuantos centímetros llega la altura del agua. Para todos los contenedores, incluyendo el depósito que se encuentra en la parte posterior de la infraestructura del edificio; se conoce si mantiene el líquido, un estado mínimo o máximo.
- b) **Caudal:** Esta constante es definida por la cantidad de fluido que pasa por la tubería en un tiempo determinado. Mediante caudalímetros presentes en las entradas de líquido de los tanques, se puede conocer el valor de la variable en cada una de las tres líneas.
- c) **Temperatura:** Por medio de esta variable se logra conocer la temperatura del líquido en los tanques; se mantienen dentro de los contenedores, un sensor de cada protocolo de comunicación para identificar la noción de calor del agua. Si bien la línea Hart por la disponibilidad de sus instrumentos puede indicar el valor de esta magnitud física, su aporte es mínimo; debido a que Profibus es la única línea que puede trabajar con agua caliente, de tal manera, solo este tramo presenta transmisores de temperatura a lo largo de su tubería.
- d) **Presión:** Mediante esta variable se puede conocer en las tres líneas, la fuerza que aplica el agua en la tubería. Además, al tener un tanque cerrado se puede trabajar con esta magnitud física.

6.1.4. Protocolos de comunicación

Dentro del proceso del llenado de tanques se cuenta con tres líneas que manejan diferentes protocolos de comunicación. En la figura 11, se puede observar una sección de la planta A del laboratorio de LACTI; donde la línea inferior pertenece a Profibus, la línea del medio a Fielbus y la primera al protocolo Hart.

Figura 11

Planta A del laboratorio de LACTI.



Nota: En la figura se observa la distribución de la planta A. Línea superior (Protocolo Hart). Segunda línea (Protocolo Fielbus). Última línea (Protocolo Profibus).

- a) **Profibus:** Es la línea más completa que dispone el laboratorio de LACTI, ya que se puede inyectar agua caliente al proceso. Si bien el PLC Schneider dispone de un módulo (TCSEGPA23F14F) para conmutarse directamente con Profibus; toda la red de este protocolo de comunicación está levantada únicamente en el automata programable Siemens, dificultando la conexión de un segundo PLC.
- b) **Fielbus:** El protocolo de comunicación de esta línea no se encuentra conectada con ninguno de los dos PLCs. De tal manera, se puede conmutar el concentrador Fielbus al puerto Ethernet del autómata Schneider; pero la restricción que presenta esta línea, es el de ser semiautomática; es decir, se tienen que abrir manualmente las válvulas para direccionar el flujo del agua a los tanques.
- c) **Hart:** Al igual que la línea de Fielbus, la red de Hart no se encuentra levantada en ningún automata programable. De tal forma, existe la posibilidad de conectar el concentrador HCS al PLC Schneider.

Dentro del proyecto surge la necesidad de seleccionar un protocolo de comunicación para poder trabajar en torno a los elementos que dispone; para ello, según los parámetros anteriores se optó por emplear la red Hart, ya que no existen restricciones ligadas a este protocolo. Además, la consigna en la que se fundamenta el trabajo de titulación es el de habilitar una segunda línea, diferente a Profibus, debido a que esta red es la única que se estaba utilizando en el laboratorio.

6.1.5. Instrumentos

En este apartado se van a exponer los diferentes equipos con los que cuenta la planta A del laboratorio LACTI, en relación con el protocolo de comunicación Hart (Tabla 4).

Tabla 4

Sensores de la planta A del laboratorio de LACTI

Sensores	Características
Deltabar M PMD55	Sensor de nivel ^a Comunicación: 4-20 mA HART Alimentación: 11-30 V DC Precisión de referencia: 0.1%
Cerabar T PMC131	Sensor de nivel ^b Comunicación: 4-20 mA Alimentación: 11-30 V DC Precisión de referencia: 0.5%
Proline Promag 10W	Sensor de caudal ^c Comunicación: 4-20 mA HART Alimentación: 11-40 V DC Precisión de referencia: 0.5%
Cerabar S PMP71	Sensor de presión ^d Comunicación: 4-20 mA HART Alimentación: 10.5-45 V DC Precisión de referencia: 0.05%
Omnigrad T TR25	Sensor de temperatura ^e Comunicación: 4-20 mA HART Alimentación: 24 V DC Rango de medición: -200 a 600°C
Liquiphant T FTL20	Sensor de nivel ^f Entrada Digital Alimentación: 10-35 V DC Longitud del sensor: 64mm

Nota: ^aEndress + Hauser (2019c), ^bEndress + Hauser (2019b), ^cEndress + Hauser (2019e), ^dEndress + Hauser (2019a), ^eEndress + Hauser (2019f), ^fEndress + Hauser (2019d).

Los actuadores se presentan en la siguiente tabla:

Tabla 5

Actuadores de la planta A del laboratorio de LACTI

Actuadores	Características
Válvula Neumática 2/2 vías	Válvula de apertura y cierre ^a Normalmente cerrada 0-25 bar
Válvula Jordan con posicionador SIPART	Posicionador electroneumático ^b Capaz de trabajar en HART, Profibus Pa y Fieldbus Foundation Tiene tres modos: Manual, Automático y Diagnostico
Danfoss BB024DS	Bobina para Solenoide ^c Alimentación: 24 V DC Potencia: 16 W
Micromaster 440	Convertidor de frecuencia para velocidad de motor trifásico Salidas analógicas ^d 6 entradas digitales 2 entradas analógicas Comunicación: HART, Profibus y Fieldbus Foundation
Válvula manual	Válvula encontrada abajo del tanque utilizada para desfogar agua en el tanque

Nota: ^aBurket (2022), ^bSiemens (2019), ^cDanfoss (2021), ^dSiemens (2001).

Se cuenta con otros equipos que son:

- a) **PLC Schneider:** De modelo P57 3634m, alimentado con 5 V DC, tiene una capacidad 8 módulos, y capaz de comunicarse por medio de Modbus RTU (mediante la tarjeta TSX SCP 114) y por Ethernet (Schneider Electric, 2020).
- b) **Modulo TSX DEY 32D2K:** Módulo de entrada discreta, que cuenta con capacidad de manejar hasta 32 entradas, con un voltaje de cada entrada de 24 V DC y corriente de 3.5 mA (Schneider Electric, 2018b).
- c) **Modulo TSX DSY 32T2K:** Módulo de salidas analógicas, que cuenta con capacidad de manejar hasta 32 salidas discretas, con un voltaje de salida discreta de 24 V DC y corriente de cada canal de 0.1 A (Schneider Electric, 2018b).
- d) **Modulo TSX AEY 1600:** Módulo de entradas analógicas, que cuenta con capacidad de

manejar hasta 16 entradas, y recibe entradas de 4-20 mA, 0-20 mA, 0-10 V, etc. Tiene una conversión analógica de 12 bits (Schneider Electric, 2018a).

- e) **Modulo TSX AsY 800:** Módulo de salidas analógicas, cuenta con la capacidad de manejar hasta 8 salidas analógicas, y envía salidas de 0-20 mA, 4-20 mA y +/-10 V. Tiene una conversión analógica de 13 bits(Schneider Electric, 2018a).
- f) **Hart Concentrator System (HCS):** Convertidor de HART a modbus RTU, puede trabajar con máximo 16 equipos y la salida ofrece un puerto RS485 para la comunicación Modbus (Moore Industries, 2017).
- g) **IOT2040:** Plataforma abierta para adquirir, procesas y transferir datos en un entorno de producción. Cuenta con dos entradas Ethernet, una entrada USB y 2 puertos COM para comunicación con RS232/485/422 desde Linux (Siemens, 2022).

6.2. Control y monitoreo de la Planta A

6.2.1. Planteamiento inicial

Dentro de la planta A, se puede manipular el proceso mediante la red de 4 a 20 mA; ya que los actuadores y el sensor de nivel (Cerabar T) del tanque abierto están conectados directamente a los puertos analógicos y digitales del PLC Schneider. Pero la demás información, de los transmisores y traductores en la línea del protocolo de comunicación Hart están llegando únicamente al concentrado HCS; algunos de esos datos pertenecen al sensor de presión diferencial para determinar la cantidad de líquido en el tanque cerrado, el caudalímetro, los traductores de temperatura y el transmisor de presión.

De tal manera, el punto a tomar en consideración en la adquisición de los datos del HCS; es que únicamente pueden ser obtenidos, al aplicar un protocolo de comunicación Modbus RTU RS 485 o RS 232. Para ello, a un inicio del proyecto de titulación se buscó conmutar el PLC al HCS, por medio de la tarjeta Modbus (Tsx scp 114) con la que cuenta el autómata programable; pero mediante pruebas en el circuito se llegó a determinar que la tarjeta era obsoleta al estar dañado el puerto de recepción de caracteres, debido a un error de conexión en los pines de la placa.

Al estar inoperable y discontinuada la tarjeta se debe buscar una segunda opción para poder enviar los datos del concentrado al PLC; se partirá de que el autómata programable cuenta con una entrada de Ethernet (Ety port), de la cual surgen dos propuestas, la primera es utilizar un conversor de Modbus RTU a Ethernet, y la otra elección es utilizar una pantalla

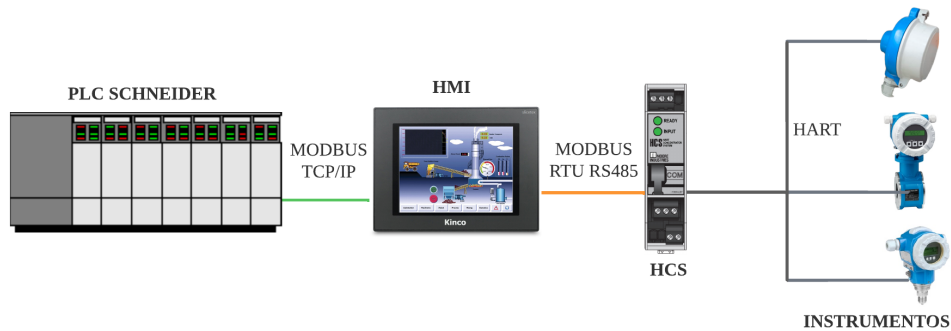
HMI como pasarela entre los dos elementos. De modo que, realizando un análisis costo - beneficio, se llegó a establecer que la opción número dos es la más viable para este proyecto.

6.2.2. Red Modbus para conmutar el PLC con el concentrador Hart

La propuesta empezará de la selección del HMI GL070E de la marca comercial Kinco, para funcionar como un gateway entre el automata programable Schneider y el concentrador. Por lo tanto, la red configurada para este proceso inicia en la emisión de datos por parte de los transmisores y transductores de la línea Hart; esta información de la planta arriba al HCS y es enviado por el protocolo Modbus RTU RS485 al HMI, dentro de la pantalla se sobrescriben las variables y son reenviadas al PLC por medio de Modbus TCP/IP (Figura 12); se obtiene de tal manera los datos de ciertos sensores que no estaban conectados directamente al controlador.

Figura 12

Red para obtener los datos del HCS



Nota: En la figura se puede visualizar la arquitectura de la red planteada para enviar los datos del concentrador HCS al PLC Schneider; configurando como una puerta de enlace a la pantalla HMI.

Cabe mencionar que antes de ejecutar el proceso anterior, se verificó que el HMI disponga de un puerto Ethernet y una entrada RS 485 para establecer la configuración de la red. Dentro de la programación de la pantalla, se utilizó el software dado por el proveedor (Kinco Dtools), en el cual se realizan los siguientes pasos:

- a) Ejecutar el software Kinco Dtools.
- b) Seleccionar dentro de las librerías el modelo de la pantalla HMI, y los protocolos de comunicación a utilizar. (HMI GL070E, Modbus RTU Extend para el HCS y Modbus

TCP Slave para el PLC).

- c) En la pantalla de la configuración de red, establecer la arquitectura a utilizar. Para esto se dispone a la pantalla HMI como maestro. Mientras que los esclavos pasaran a ser el PLC y el concentrador HCS, mismos que serán conectados por Ethernet y RS 485 respectivamente.
- d) En una opción dentro del menú del proyecto se crea una Macro, la cual servirá para sobrescribir las variables. Se introduce la dirección del esclavo perteneciente al HCS, y se establece la variable auxiliar del PLC para guardar el dato.
- e) Se crea un Frame, que es la pantalla donde se programa el entorno gráfico del HMI. En esta sección se llama a la Macro del punto anterior.
- f) Descargamos la configuración en el HMI.
- g) Se conecta el HCS a la pantalla HMI, mediante el cable RS 485,
- h) Conmutamos por medio de un cable Ethernet, al HMI con el PLC.
- i) Finalmente ponemos a prueba la red.

Un punto importante a resaltar dentro de la utilización de la pantalla Kinco, es que no solo opera como un gateway entres el HCS y el automata programable Schneider; sino también, se puede generar un entorno gráfico para controlar y monitorear la planta. Dicho esto, se diseña un soporte para montar el HMI dentro de un gabinete del laboratorio, quedando como una opción para que el usuario pueda manejar el proceso del llenado de tanques.

Figura 13

Implementación del HMI en la planta



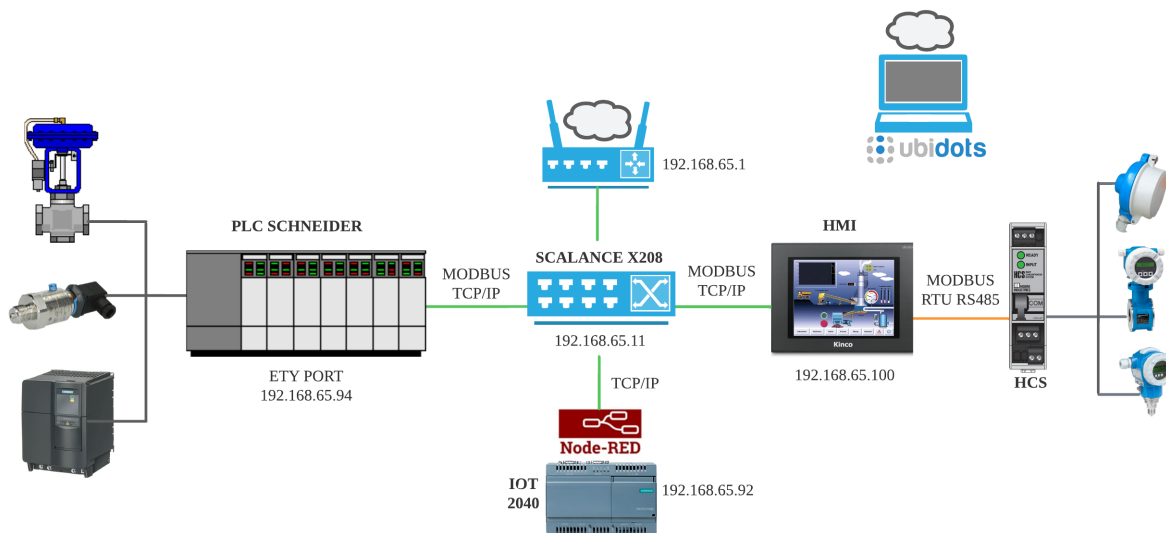
Nota: En la figura se puede observar a pantalla HMI instalada en un gabinete de la planta. Dando la disponibilidad de que el operador pueda controlar o monitorear el proceso.

6.2.3. Red general

Una vez obtenida la información de todos los sensores, y al poder controlar la planta a partir de la línea del protocolo Hart; se integra en la red general que maneja el laboratorio de LACTI, la arquitectura de red planteada en el punto anterior. Por consiguiente, se utiliza el conmutador (Scalance X208) como puerta de enlace para disponer de una mejor manera los datos de cada equipo. El siguiente diagrama expone la conexión realizada (Figura 14):

Figura 14

Red general



Nota: En la figura se puede visualizar la configuración de la red general del laboratorio de LACTI, para el proyecto.

6.2.4. Control y monitoreo de la planta por medio de la red (IOT)

Al poner en red la línea Hart conjuntamente con el PLC Schneider, se da la opción de emplear el IOT 2040 que se mantiene en la planta B, a fin de que el router administre los datos del proceso a Node-red, y paulatinamente disponerlos en la nube (Figura 14). Con la información en la web se pretende utilizar la plataforma de Ubidots como gestor, y diseñar un dashboard para que el usuario pueda interactuar con la planta.

De tal manera, se logra realizar el control y monitoreo de la planta por medio de la red, proporcionando la posibilidad de aplicar el internet de las cosas para que no solo se manipule la planta con ayuda de la pantalla HMI, sino también desde cualquier parte del mundo a través

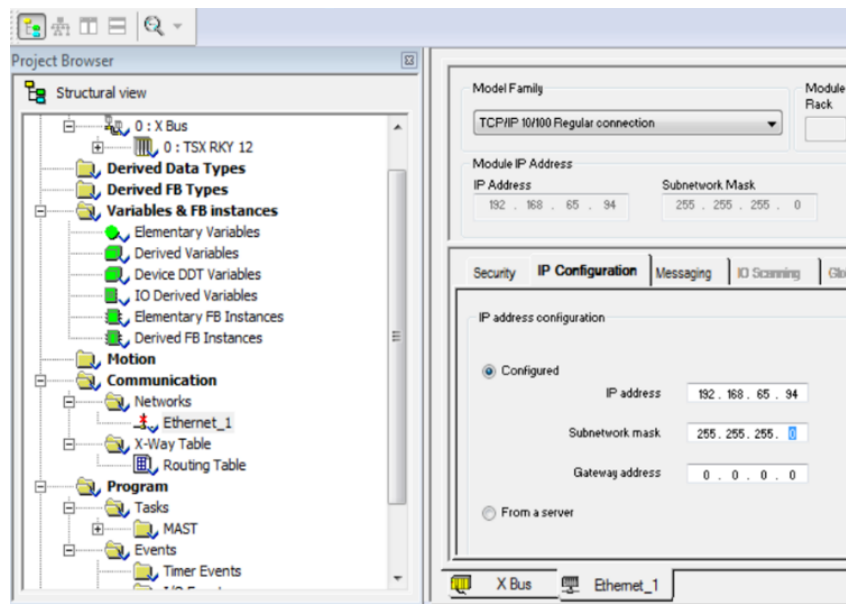
de Ubidots. El proceso realizado para la configuración del IOT se describe en los siguientes pasos.

- **IOT:**

Para el monitoreo del PLC mediante Ubidots, se conecta al conmutador Ethernet para establecer una conexión con el router mediante TCP/IP. Para esto se debe configurar el puerto de Ethernet del PLC con la IP de 192.168.65.94 utilizando el programa Unity Pro (figura 15).

Figura 15

Configuración de IP del PLC Schneider

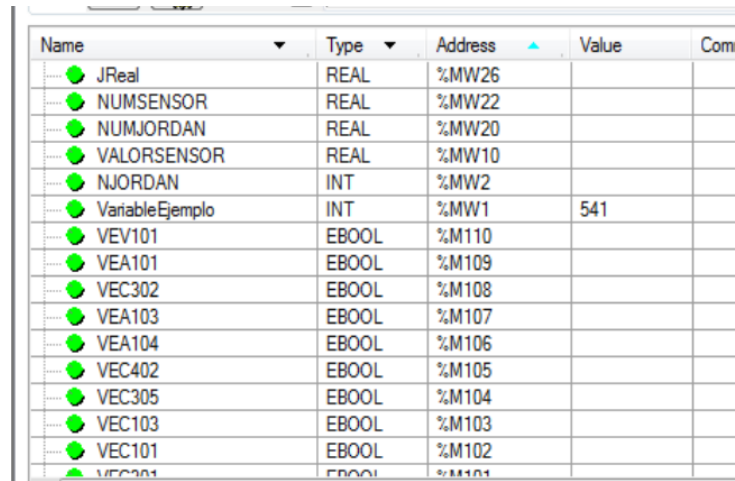


Nota: Se visualiza la configuración del puerto Ethernet en el programa Unity Pro.

Es necesario asignar una dirección a las variables dentro del Unity Pro, en este caso se utiliza la dirección %MW para variables de hasta 16 bits como números flotantes o enteros, y %M para variables booleanas (figura 16).

Figura 16

Configuración de las variables



Name	Type	Address	Value	Com
JReal	REAL	%MW26		
NUMSENSOR	REAL	%MW22		
NUMJORDAN	REAL	%MW20		
VALORSENSOR	REAL	%MW10		
NJORDAN	INT	%MW2		
VariableEjemplo	INT	%MW1	541	
VEV101	EBOOL	%M110		
VEA101	EBOOL	%M109		
VEC302	EBOOL	%M108		
VEA103	EBOOL	%M107		
VEA104	EBOOL	%M106		
VEC402	EBOOL	%M105		
VEC305	EBOOL	%M104		
VEC103	EBOOL	%M103		
VEC101	EBOOL	%M102		
VEC201	EBOOL	%M101		

Nota: En la figura se puede visualizar la configuración de las variables con cada una de sus direcciones. Software Unity Pro.

Posteriormente se realiza una comunicación mediante un OPC para comprobar las variables. Se utiliza el software KEPServerEX. Donde se crea un canal con un protocolo de comunicación Modbus TCP/IP Ethernet, y también se configura un dispositivo con la dirección IP del PLC Schneider.

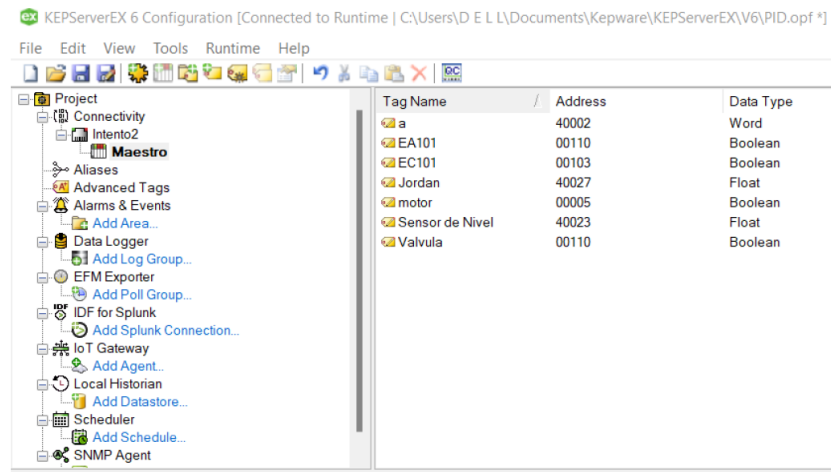
Luego se generan las variables que se desean monitorear utilizando los Function Code (FC) de Modbus. Para esto es necesario tener en consideración que dependiendo de la variable que se quiere enviar, se usa una función diferente. Por ejemplo:

- a) **FC01:** con acceso a la escritura y lectura, utilizado para variables booleanas. Su dirección se escribe con 0000x.
- b) **FC02:** con acceso solo a lectura, utilizado para variables booleanas. Su dirección se escribe con 1000x.
- c) **FC03:** con acceso a la escritura y lectura, utilizado para variables de hasta 16 bits. Su dirección se escribe con 4000x.
- d) **FC04:** con acceso solo a lectura, utilizado para variables de hasta 16 bits. Su dirección se escribe con 3000x.

Utilizando las direcciones configuradas en el Unity Pro, se monitorea las variables que se muestran en la figura 17:

Figura 17

Variables monitoreadas por KEPServerEX

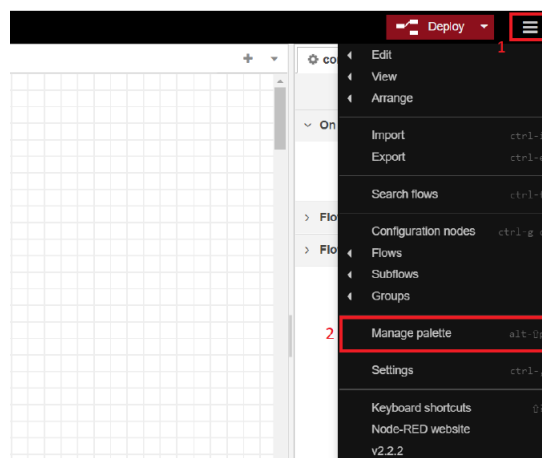


Nota: Se puede visualizar las direcciones de ciertas válvulas y el sensor de nivel.

Para realizar una conexión entre el PLC Schneider y Node-red, se utiliza el IOT2040. Para ello es necesario conectarse a la red, y colocar en el navegador la dirección 192.168.65.92:1880, que corresponde a Node-red. En la página se descarga la librería de Ubidots, para eso hay que dirigirse al manage palette (figura 18).

Figura 18

Librerías Node-red

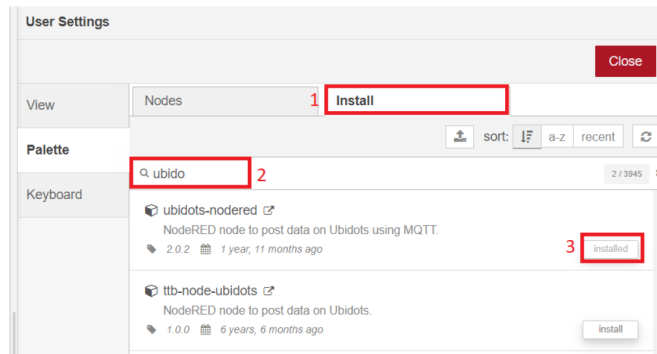


Nota: Pasos para instalar librerías en Node-red.

En la figura 17, se observa la pantalla en la cual se debe buscar la librería de ubidots, e instalar la opción Ubidots-Nodered.

Figura 19

Instalar librería Ubidots

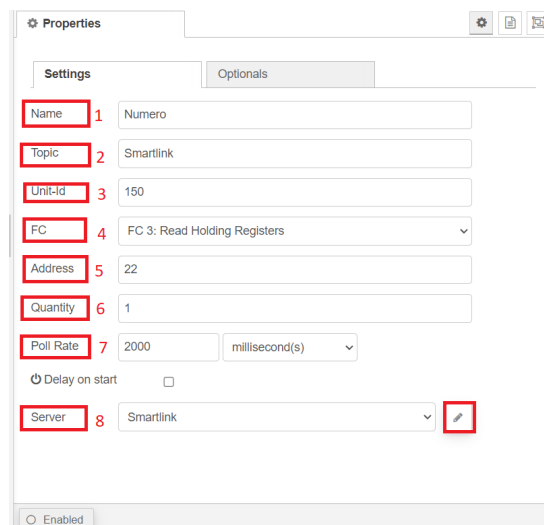


Nota: Pasos para instalar la librería de ubidots.

Para recibir los datos desde el PLC se utiliza el bloque Modbus-Read; y Modbus-Write para enviar datos al PLC. Estos deben ser configurados bajo los siguientes parámetros:

Figura 20

Parámetros Modbus



Nota: Parámetros para establecer conexión entre Node-red y el PLC Schneider.

Donde:

1. **Name:** Nombre del nodo.
2. **Topic:** Una descripción de la utilidad del nodo.
3. **Unit-id:** Número de identificación del nodo.
4. **FC:** Código de función de Modbus, donde la función 1 te permite recibir valores tipo booleano y la función 3 valores numéricos o string.
5. **Address:** Dirección del PLC de donde se obtienen los valores.
6. **Quantity:** Número de valores que va a recibir.
7. **Poll rate:** Tiempo que deberá esperar para recibir otro dato del PLC.
8. **Server:** Configuración de la comunicación Modbus. Para configurar la comunicación dirigirse a la herramienta lápiz como se muestra en la figura 20.

En la ventana de configuración de la conexión Modbus se ajustan los siguientes parámetros:

Figura 21

Parámetros conexión Modbus

The screenshot shows a configuration window for a Modbus connection. At the top, there are three buttons: 'Delete', 'Cancel', and 'Update'. Below this is a 'Properties' section with a gear icon and a refresh icon. The configuration fields are: 'Name' (1) with value 'Smartlink', 'Type' (2) with value 'TCP', 'Host' (3) with value '192.168.65.94', 'Port' (4) with value '502', and 'TCP Type' (5) with value 'DEFAULT'. Each field label and its corresponding number are highlighted with a red box.

Nota: Parámetros para establecer conexión entre Node-red y el PLC Schneider, mediante modbus.

Donde:

1. **Name:** El nombre de la conexión.
2. **Type:** Tipo de Modbus utilizado, en este caso se va a establecer conexión por TCP, otra opción sería serial.

3. **Host:** La dirección del puerto Ethernet del PLC Schneider, en este caso es 192.168.65.94.
4. **Port:** Puerto utilizado para esta conexión, el puerto normalmente usado es el 502.
5. **TCP Type:** Tipo de conexión TCP, recomendable dejar en DEFAULT.

Posteriormente, se configuran los bloques de Ubidots como se muestra en la figura 22:

Figura 22

Parámetros conexión Ubidots

The image shows a 'Properties' configuration window for a Ubidots connection. It contains the following fields and options:

- 1** Account Type: Ubidots for Education (dropdown menu)
- 2** Name: Salida (text input)
- 3** Token: BBFF-GrX5fmUJjCgwh67IMICFTaYNwkeRnn (password input)
- 4** Device Label: Var1 (text input)
- Enable secure TLS connection:

Nota: Parámetros para establecer conexión entre Node-red y Ubidots.

Donde:

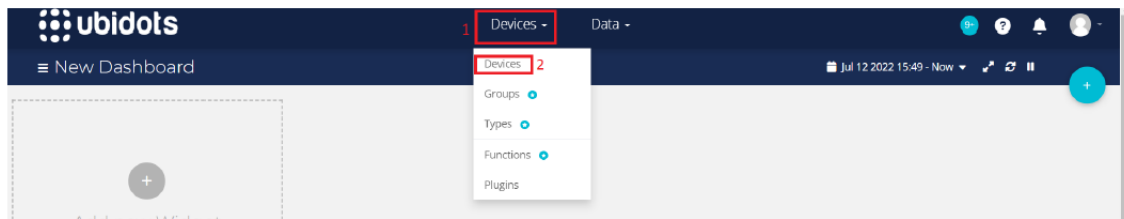
1. **Account Type:** Tipo de cuenta de Ubidots.
2. **Name:** Nombre del bloque.
3. **Token:** Código generado por ubidots para establecer conexión con node-red.
4. **Device Label:** Nombre de la variable, con este nombre se encuentra la variable en Ubidots.

En el bloque de Ubidots In los parámetros son similares, la única diferencia es que se debe colocar un nombre de dispositivo y nombre de la variable, estos deben ser creados primero en Ubidots.

Para crear la variable nos dirigimos a la página de los dispositivos en Ubidots (figura 23).

Figura 23

Dispositivos en Ubidots

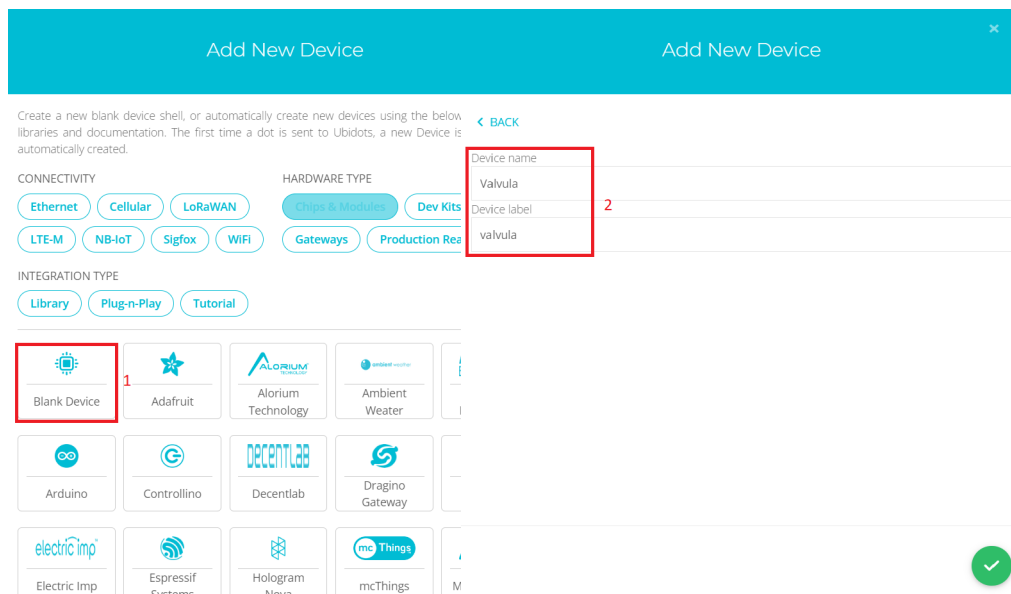


Nota: Dirección para encontrar a los dispositivos en ubidots.

En la página se crea un nuevo dispositivo y selecciona blank device, y se asigna un nombre al dispositivo, el cual debe coincidir con el que se coloca en la configuración del bloque Ubidots In (figura 24).

Figura 24

Crear dispositivo nuevo en Ubidots

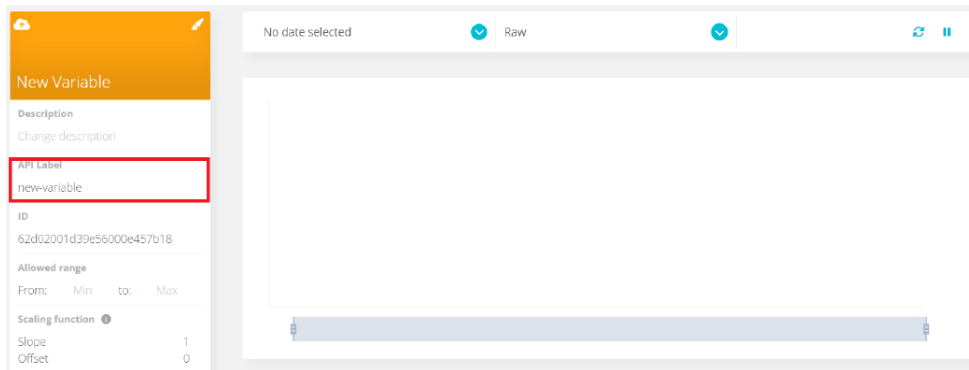


Nota: Pasos para crear dispositivos en ubidots.

Una vez generadas las variables, se observa una imagen similar a la que se presenta en la figura 25. Hay que cambiar el nombre del API Label para que coincida con el cuadro de variable label del bloque de Ubidots In.

Figura 25

Nombre de variable creada

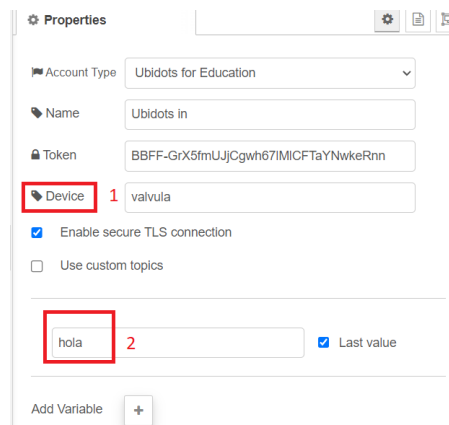


Nota: Página de la variable creada.

En Node-red y se coloca el bloque de Ubidots In, donde aparecieron los parámetros vistos en la figura 26:

Figura 26

Parámetros de bloque Ubidots In



Nota: Configuración de parámetros de bloque de Ubidots In.

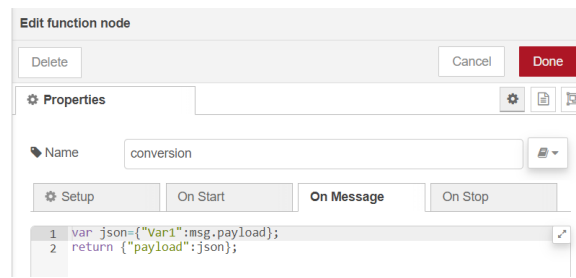
Donde:

1. **Device:** Se coloca el nombre del dispositivo creado en Ubidots.
2. **Label:** Se coloca el nombre de la variable creada en Ubidots.

Se agregan nombres a las variables obtenidas por el Modbus Read, debido a que Ubidots necesita que envíen el nombre de la variable y su valor. Para ello se emplea el bloque función y las siguientes sentencias de programación :

Figura 27

Conversión de variables para envío de datos en Ubidots.



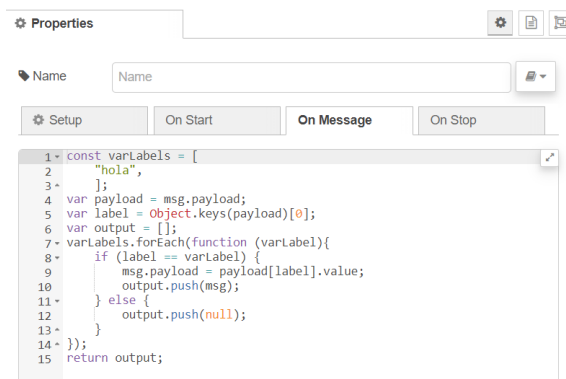
Nota: Se observa las líneas de programación utilizadas para convertir las variables a json.

En la figura 27 se observa como se crea una variable que va a almacenar lo enviado por el PLC con el nombre que se le asigna, y en la segunda línea se envía la variable almacenada.

Para el caso contrario, cuando se recibe la variable de Ubidots y enviarla al PLC, es necesario otro tipo de programación (figura 28).

Figura 28

Conversión de variables para recepción de datos en Ubidots



Nota: Líneas de programación utilizadas para extraer las variables del bloque de Ubidots y pasarlas a Modbus.

En la primera línea se crea la variable varLabels en la cual se coloca el nombre de la dirección creada en Ubidots in. En la línea 4, se guarda la variable del Ubidots. Después se

compara el nombre puesto en el bloque Ubidots in con varLabels, si son iguales se envía al bloque de Modbus Write, si son diferentes no se envía.

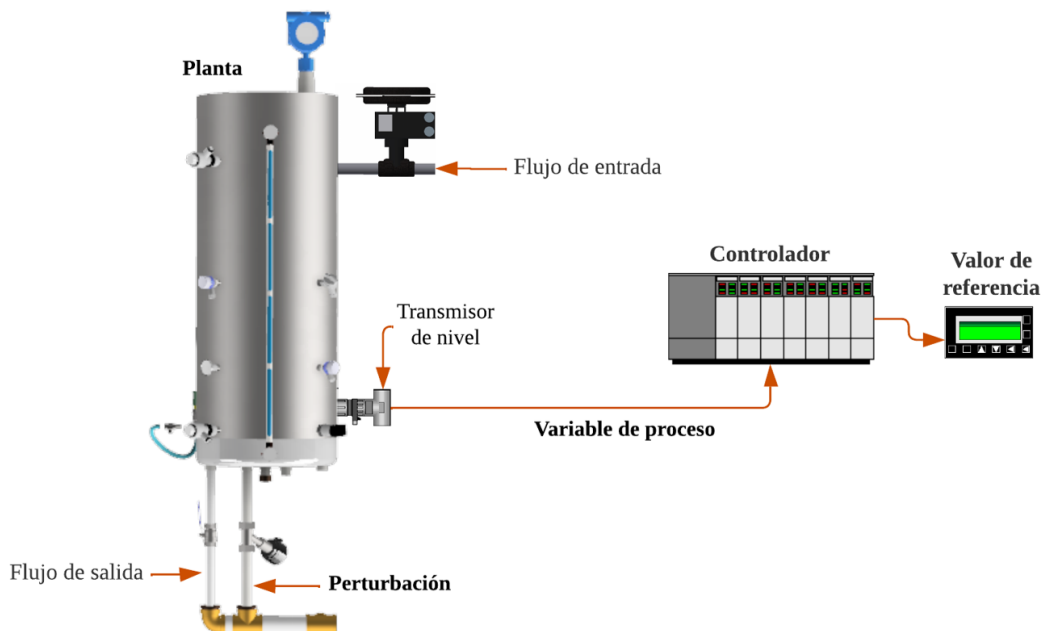
6.3. Algoritmo de control (PID)

Dentro de la propuesta del trabajo de titulación, se plantea desarrollar un algoritmo de control para el manejo de la variable de nivel en la planta A. Se opta por utilizar una arquitectura de control proporcional, integral y derivativo (PID), dando la posibilidad de hacer frente las probables fallas o errores que pueda presentar el proceso.

El lazo de control configurado está ligado a la figura 29; de tal manera, se tendrá como la planta al flujo de entrada de líquido manipulado por la válvula proporcional, el tanque y un flujo de salida constante dado por una válvula manual. Todo el proceso será controlado por el autómata programable Schneider (PLC) y se fijará un valor de referencia a seguir. Además, para medir la cantidad de agua se utilizará un transmisor de nivel; y se mantendrá una segunda línea de flujo de salida de líquido para introducir perturbaciones al sistema.

Figura 29

Lazo de control



Nota: En la figura se observa los elementos que componen al lazo de control, con el que se diseña el PID.

Definiendo el lazo de control cerrado con los respectivos elementos y con las especificaciones a cumplir (Figura 7), se tiene que:

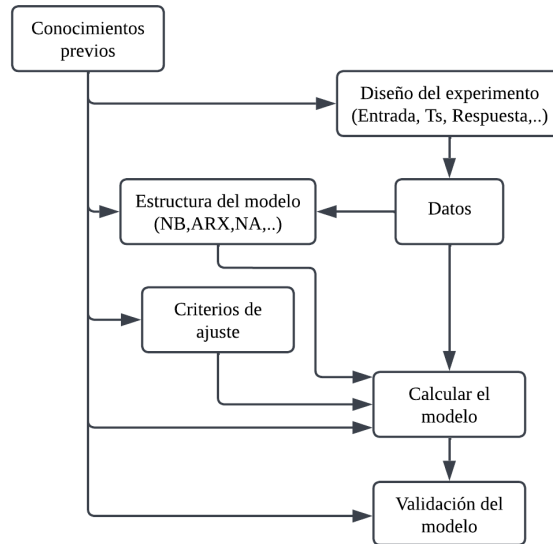
- a) El valor de referencia a cubrir será de 40 cm de nivel. La cantidad de líquido máximo que puede mantener el tanque debido al censado y a su estructura es de 100 cm.
- b) El controlador a utilizar será de la arquitectura proporcional, integral y derivativo (PID).
- c) La perturbación será introducida al activar la válvula neumática On/Off, ubicada debajo del tanque.
- d) La planta será el tanque abierto. La válvula manual que se encuentra debajo del mismo, se mantendrá con una apertura del 50% para contar con un flujo de salida constante. El flujo de entrada, será la variable a regular por la señal de actuación del controlador para tener el nivel de líquido deseado.
- e) La retroalimentación estará dada por el sensor de nivel (Cerabar T PMC131). Este transmisor se fundamenta bajo un principio de presión absoluta y relativa.

6.3.1. Identificación del sistema

La identificación del sistema se lo realizo a partir de datos experimentales; debido a que, la mejor manera de obtener un modelo ligado al proceso real es mediante valores extraídos en el campo físico. Al no disponer de una constante específica de la salida de caudal, como también, el principio de histéresis que mantiene la válvula proporcional, vuelve al sistema dificultoso para abordarlo desde un punto analítico - matemático. De tal forma, los criterios para la identificación de sistemas con un enfoque de obtención de datos experimentales estará dado por las siguientes etapas (Figura 30):

Figura 30

Identificación del sistema



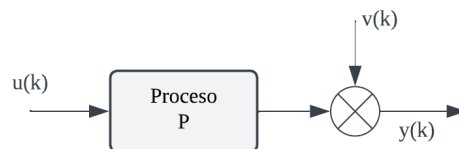
Nota: En la figura se puede visualizar las principales etapas que se debe llevar en la identificación de los sistemas.

El método empleado para hallar la función de transferencia que represente la dinámica del sistema se denomina modelo Autorregresivo con Entrada Externa (ARX), representado por una ecuación en diferencia:

$$y(k) = a_1y(k-1) + \dots + a_ny(k-na) + b_0u(k-d) + b_1u(k-1-d) + \dots + b_mu(k-nd-d) \quad (6)$$

Figura 31

Modelo ARX



Nota: En la figura se puede observar la estructura del modelo Autorregresivo con Entrada Externa (ARX). Adaptado de Diseño e implementación de un controlador difuso tolerante a falla, aplicado al control de nivel de líquido, Criollo (2015). © 2015 Universidad Politécnica Salesiana.

La gráfica (31) denota la forma simplificada del modelo ARX. Se tiene una entrada $u(k)$, un proceso (P) que representa la dinámica del sistema, un ruido aditivo $v(k)$ y una salida $y(k)$. Este algoritmo de lectura continua de datos nuevos, ajusta paulatinamente los valores de cada coeficiente. Dentro de la ecuación 6, la incógnita (a) con (n) muestras retrasadas corresponde a la salida $y(k)$, la variable (b) con (m) muestras retrasadas representa la entrada $u(k)$ y el coeficiente (d) estará vinculado al retraso del sistema.

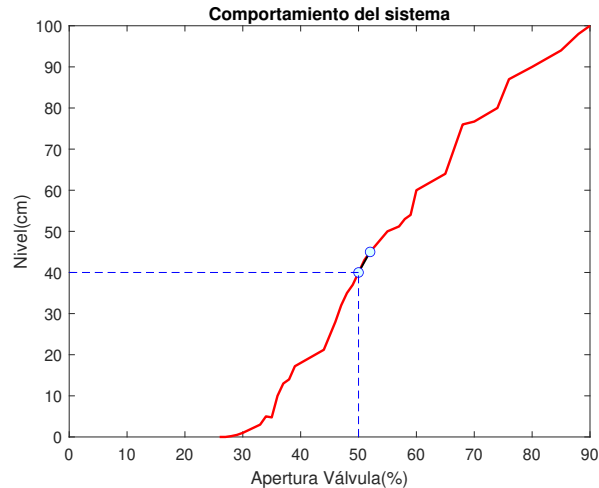
En la identificación de las incógnitas (a y b), se aplicará el método denominado mínimos cuadrados recursivos, mismo que vincula una identificación paramétrica en un modelo discreto para minimizar el error de la estimación entre lo experimental y el sistema. De tal manera, se propone ajustar una estructura ideal del modelo ARX a la lectura continua de los valores obtenidos en el funcionamiento del sistema real. Además, se debe de garantizar que los parámetros elegidos para el método se ajusten a “La suma de los cuadrados de las diferencias entre los valores de la salida observados (reales) y los estimados, multiplicada por factores que midan el grado de precisión sea un mínimo” (Carpio, 2013). Todo el proceso anteriormente nombrado es el principio en el que se basa la herramienta Toolbox Ident ; mediante una serie de interacciones puede estimar los parámetros de la función de transferencia que represente al sistema.

6.3.2. Consideraciones de la señal de entrada

Para ingresar la señal de entrada al sistema en la toma de datos experimentales, se aplica el concepto del PRBS; misma que es una señal de secuencia binaria pseudoaleatoria enviada a la planta por un periodo de tiempo. Al ser el sistema no lineal (Figura 32), el salto del escalón para el porcentaje de apertura de la válvula no puede variar significativamente debido a los datos erróneos que concibe esta acción; el criterio a tomar en estos casos, según Dasgupta y Michalewicz (1997) es el de conocer la dinámica de la planta para variar una amplitud del +/- 5% alrededor de un punto de equilibrio identificado; con ello se busca llegar a linealizar un tramo de la curva del comportamiento del sistema, para que en torno a esta sección pueda obtenerse la función de transferencia.

Figura 32

Dinámica del sistema



Nota: En la figura se puede visualizar la dinámica del sistema (Nivel del líquido), con respecto al porcentaje apertura de la válvula proporcional; manteniendo en la configuración de la planta un flujo de salida constante con la apertura de la válvula manual al 50%. Además, se presenta el tramo de linealización para identificar la función de transferencia.

De tal manera, la consigna planteada para el diseño de control de la variable de nivel apunta a llenar el 40% del tanque. Esta acción, sin un lazo de control, se puede realizar con un escalón del 50% de la apertura de la válvula proporcional; aplicando la consideración anterior del 5% , la variación del ruido blanco se centra entre el 50% y 52% (Figura 32).

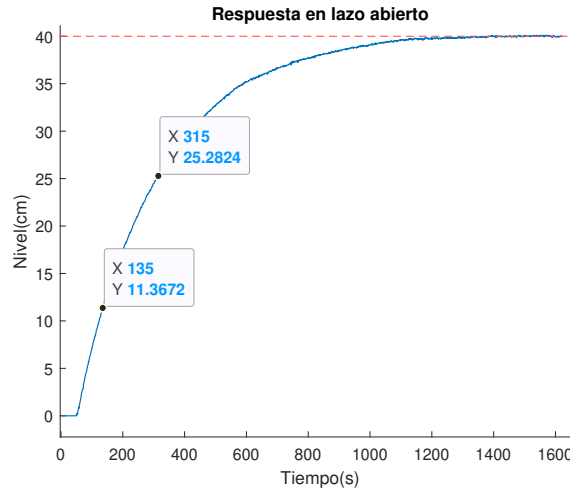
Si bien se conoce la apertura que debe de tener la válvula, se requiere definir un periodo de tiempo para su cambio. Por tal motivo, se plantea utilizar el método de Smith para identificar la función de transferencia que simule el nivel de llenado del 40% del tanque, y tomarlo como punto de referencia para un tiempo de muestra.

6.3.3. Tiempo de muestra

El sistema se aproxima a uno de primer orden, por lo cual se utiliza el método de identificación de función de transferencia Smith, considerando el tiempo al 28.3% y 63.2% de la salida del sistema (Figura 4). Mediante la adquisición de datos en la plataforma la LabView, se obtiene el comportamiento del proceso.

Figura 33

Respuesta de lazo abierto (Smith)



Nota: En la gráfica se observa la respuesta en lazo abierto del sistema para un escalón unitario de 50. Los parámetros de tiempos señalados se utilizarán para el método de identificación Smith.

Con base a la figura 33, se obtienen los datos:

$$t_{63} = 315 \text{ s}$$

$$t_{28} = 135 \text{ s}$$

A partir de la ecuación 4, se puede obtener τ :

$$\tau = -1.5(135) + 1.5(315)$$

$$\tau = 270$$

Considerando la ecuación 5, se tiene t_m :

$$t_m = 1.5(135) - 0.5(315)$$

$$t_m = 45$$

La constante k , vendrá dada por la división de la salida del sistema y el valor escalón de

entrada.

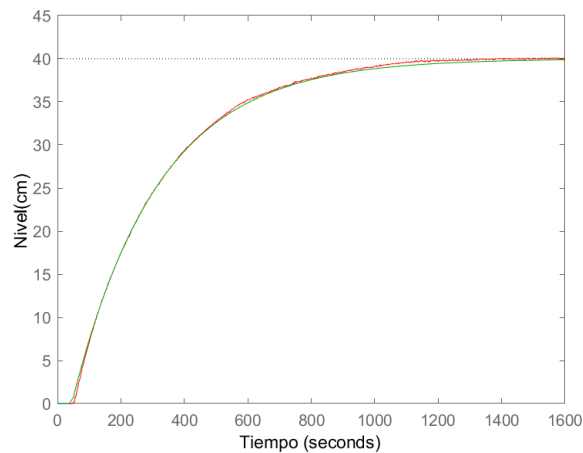
$$k = \frac{40}{50}$$
$$k = 0.8$$

De tal manera, se logra obtener la función de transferencia de primer orden en lazo abierto que representa el sistema cuando se tiene una entrada escalón de 50. Reemplazando en la ecuación 2:

$$G_{ps} = \frac{0.8 * e^{-45s}}{270s + 1} \quad (7)$$

Figura 34

Respuesta de lazo abierto/Función de transferencia estimada



Nota: En la presente gráfica se puede visualizar la respuesta en lazo abierto (rojo) y la función de transferencia estimada por el método Smith (verde). Para una entrada escalón de 50.

La gráfica muestra la aproximación y aceptabilidad del método Smith. Si bien la respuesta es la indicada, la función de transferencia solo servirá para un escalón de 50. Por tal motivo, la incidencia de la función de transferencia es únicamente para identificar el tiempo a variar la señal PRBS; mas no, para diseñar el controlador. El valor del periodo T_s , será diez veces más rápido que τ . Dado esto tenemos:

$$T_s \leq \frac{\tau}{10} \quad (8)$$

$$T_s \leq \frac{270}{10}$$

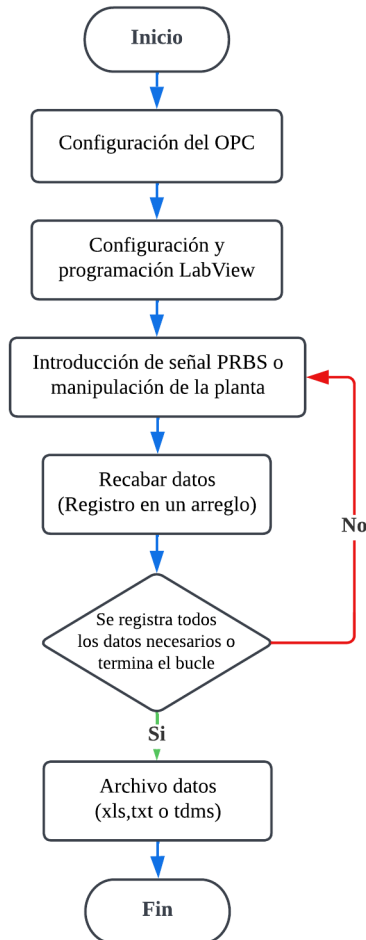
$$T_s \leq 27 \text{ s}$$

6.3.4. Obtención de datos del sistema

Para recabar datos del sistema se desarrolló la configuración y un programa en el software LabView; el cual mediante un OPC, da la posibilidad de establecer una conexión entre la planta y el computador (Figura 35). De esta forma, se puede introducir la señal PRBS al proceso, o gestionar el control y monitoreo en la obtención de valores que ayuden a comprender el comportamiento del llenado del tanque abierto.

Figura 35

Obtención de datos del sistema

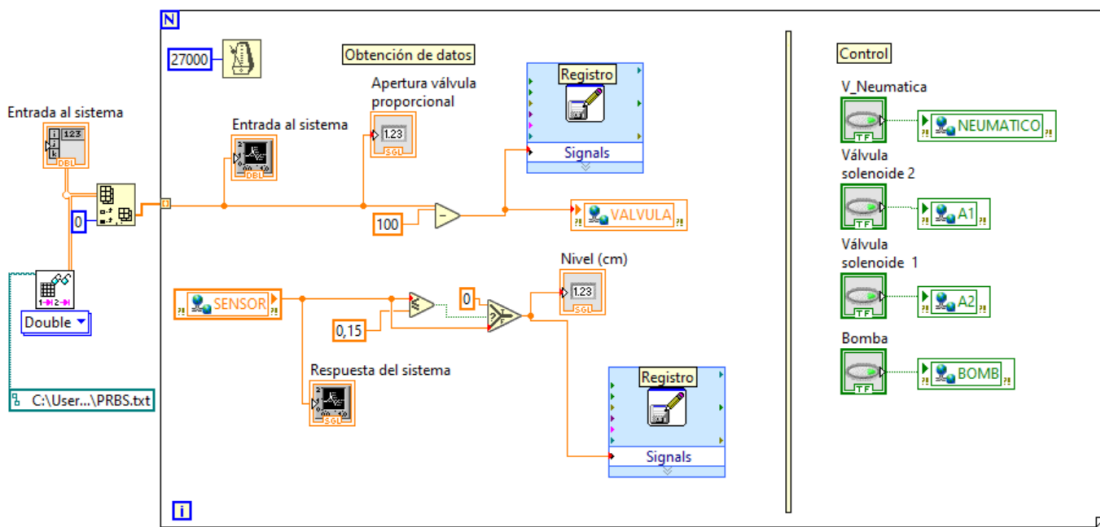
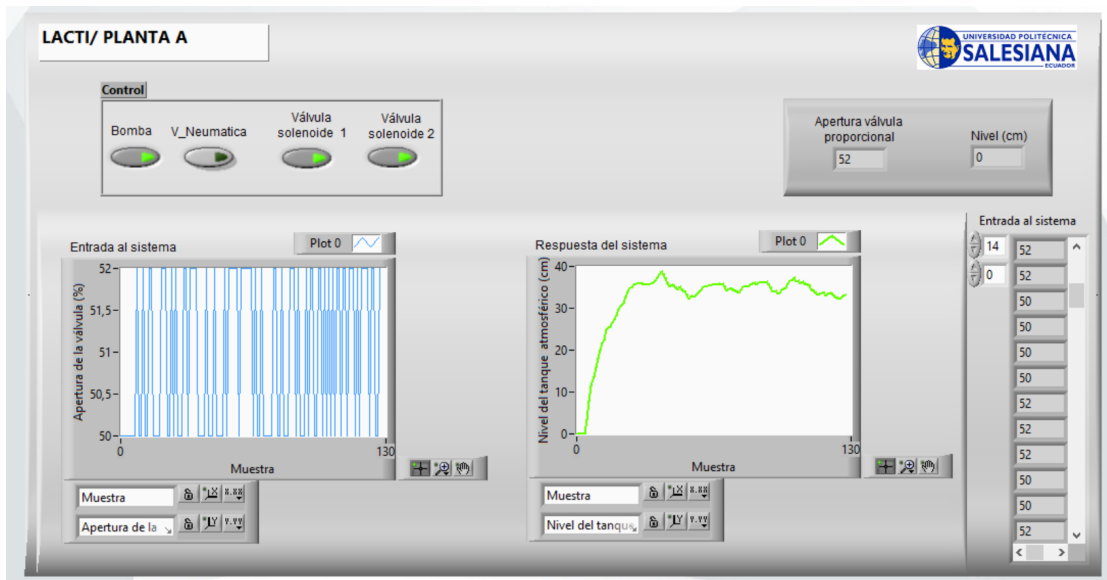


Nota: En la figura se puede observar el procedimiento que se debe de llevar para poder obtener los datos que ayuden a la identificación del sistema.

Un ejemplo del proceso anterior se puede visualizar en la figura 36, que alude al dashboard empleado en la adquisición de los parámetros de entrada y salida para la identificación de la función de transferencia.

Figura 36

Panel de control (LabView)



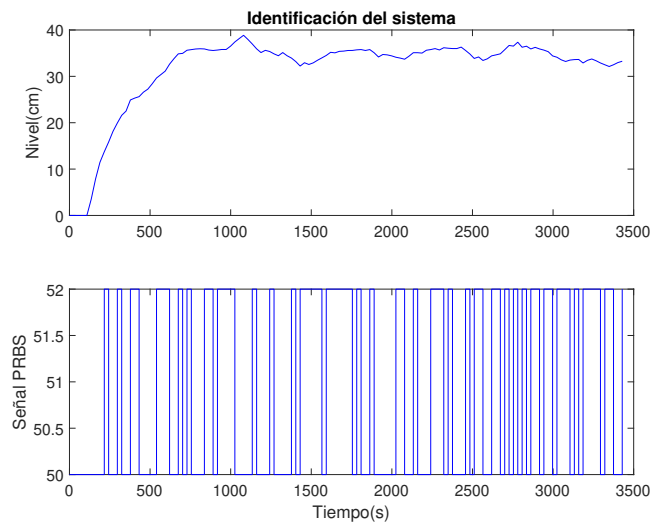
Nota: Se puede observar en la primera figura el dashboard diseñado para monitorizar el comportamiento del sistema y manipular la planta. Mientras que en la segunda figura se expone los bloques utilizados en la programación del entorno gráfico y la obtención de los registros de datos.

6.3.5. Identificación de la función de transferencia

El tiempo propuesto para cada tramo de la señal PRBS será de 27 s para conseguir 7 bits de datos; es decir, se considera 128 muestras en la identificación del sistema. Las entradas tomadas para el análisis son las propuestas por la señal pseudoaleatoria PRBS y la salida es la respuesta del nivel del tanque ante el valor de excitación (Figura 37). Además, se garantiza que el comportamiento del proceso consiga aproximarse al menos a tres veces τ , de la dinámica tomada como referencia.

Figura 37

Respuesta del sistema/Señal PRBS



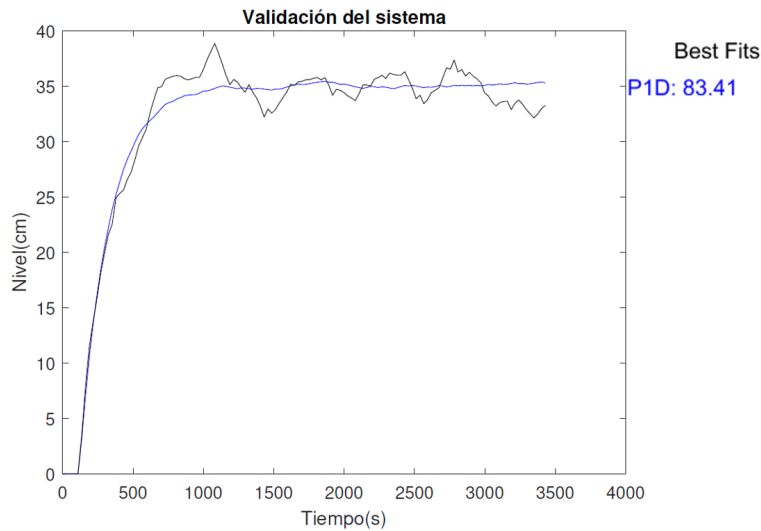
Nota: En la gráfica se puede apreciar la respuesta del sistema (Primera figura) aplicando una señal PRBS (Segunda figura), cada 27 segundos para 128 muestras.

Dentro de la identificación se considera que el sistema es de primer orden con un valor de retardo, debido a dos factores, la distancia de la tubería entre el depósito de agua y el tanque abierto; como también, la sección ondulada que se debe llenar antes de que el sensor tome un dato de nivel. De tal manera, la herramienta de Matlab Ident proporciona la siguiente función de transferencia:

$$G(s) = \frac{0.6871}{223.4s + 1} e^{-80s} \quad (9)$$

Figura 38

Validación de la función de transferencia

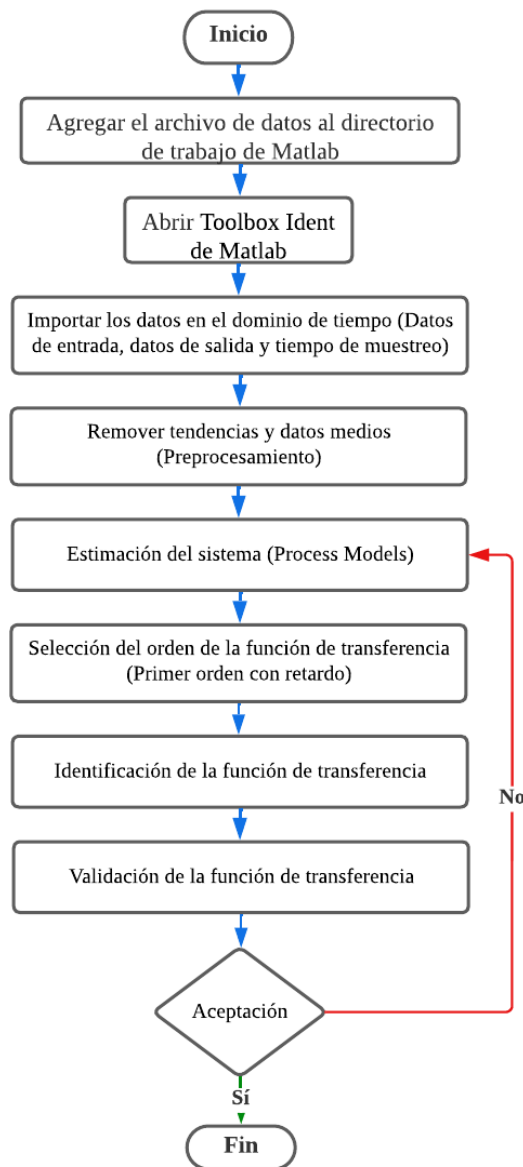


Nota: Se presenta en la gráfica la similitud que existe entre los datos experimentales y la función de transferencia.

Para dar validación a la función de transferencia obtenida se debe introducir una segunda toma de datos al gestor Ident, comparando la aproximación que se puede dar entre la ecuación y los valores experimentales. El sistema presenta un 83% de semejanza (Figura 38); sabiendo que el porcentaje de similitud debe de ser mínimo del 70%, se da por aceptado la función que representa al sistema. Los pasos que se deben de seguir para la configuración de la librería Ident de Matlab en la identificación del proceso, se lo describe en la en el siguiente diagrama (Figura 39):

Figura 39

Configuración de la Toolbox Ident Matlab



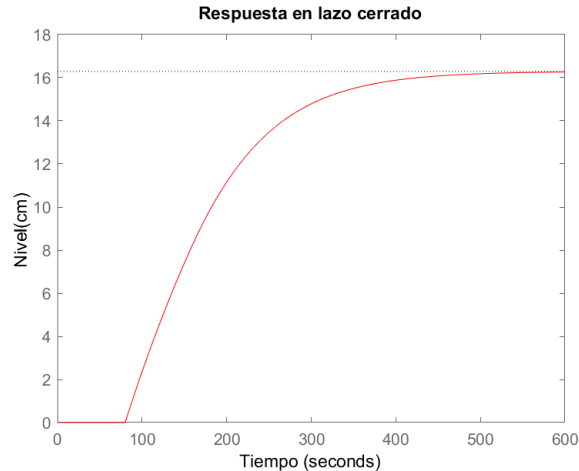
Nota: En la figura se puede observar los pasos a seguir en la configuración de la librería Ident de Matlab, para hallar la función de transferencia que represente a la dinámica del sistema.

6.3.6. Control proporcional, integral y derivativo (PID)

Para el diseño del controlador, se debe visualizar la respuesta en lazo cerrado de la función de transferencia; buscando tener una mejor apreciación acerca del comportamiento para proporcional las ganancias adecuadas a la planta.

Figura 40

Respuesta en lazo cerrado de la función de transferencia



Nota: La gráfica muestra el comportamiento del sistema en lazo cerrado, con un valor de referencia de 40 cm de nivel.

La figura 40, representa la dinámica del sistema para obtener un nivel de 40 cm. De tal manera, para la arquitectura del control PID se seleccionará únicamente la ganancia proporcional e integral. La justificación de la selección va ligada a que si se inyecta en el sistema únicamente una parte proporcional se tendrá problemas con la oscilación, por lo cual se debe de acompañar por una segunda ganancia. Se descarta utilizar la parte derivativa porque el sistema no tiene conflicto en la sección transitoria; mas bien, el proceso no sigue el valor de referencia, por lo que se debe proponer una ganancia integral, dando así que el controlador recuerde el comportamiento de la curva de tendencia del sistema e introduzca una constante para llegar al valor de nivel propuesto.

Por otra parte, un punto a tratar acerca de la función de transferencia que representa el llenado del tanque, es el retardo; el trabajar con una exponencial en la búsqueda de los valores PID para el controlador, puede generar problemas en la estabilidad y en la dinámica del sistema. Según Castaño (s.f.) aplicar el método de aproximación PADE es la mejor solución para trabajar con sistemas que mantienen un retardo; mismo que propone aproximar la función a otra racional polinómica. De esa manera, se buscar hacer que coincida una función $f(x)$ en un máximo orden posible de una función racional:

$$F(x) \approx R(x)$$

$$f(0) = R(0),$$

$$f' = R'(0),$$

$$f'' = R''(0),$$

...

$$f^{(m+n)}(0) = R^{(m+n)}(0)$$

Se busca que el coeficiente (m) de las dos funciones coincidan (n) veces cuando x tome valores de 0. De tal manera, esta restricción está compuesta de:

$$F(x) = a_0 + a_1x + a_2x^2 + .. + a_mx^m + a_{m+1}x^{m+1} + ..$$

$$R(x) = b_0 + b_1x + b_2x^2 + .. + b_mx^m + b_{m+1}x^{m+1} + ..$$

Se aproxima a una función racional, y se genera la equivalencia para un polinomio Q_m .

$$R_m s = \frac{1 + a_0 + a_1x + a_2x^2 + .. + a_mx^m}{1 + b_0 + b_1x + b_2x^2 + .. + b_mx^m}$$

$$R_m s = \frac{Q_m(-\theta s)}{Q_m(\theta s)} \quad (10)$$

$$Q_m(-\theta s) = \sum_{j=0}^m \frac{(m+j)!}{j!(m-j)!} (\theta s)^{m-j} \quad (11)$$

Dentro de nuestra función de transferencia se toma al retraso como (θ) y el orden de la aproximación como (m). Se resuelve la ecuación 11:

$$e^{-\theta s} = \theta = 80 \quad m = 1$$

$$Q_1(-\theta s) = \frac{(1+0)!}{0!(1-0)!} (-80s)^{1-0} + \frac{(1+1)!}{1!(1-1)!} (-80s)^{1-1}$$

$$Q_1(-\theta s) = -80s + 2$$

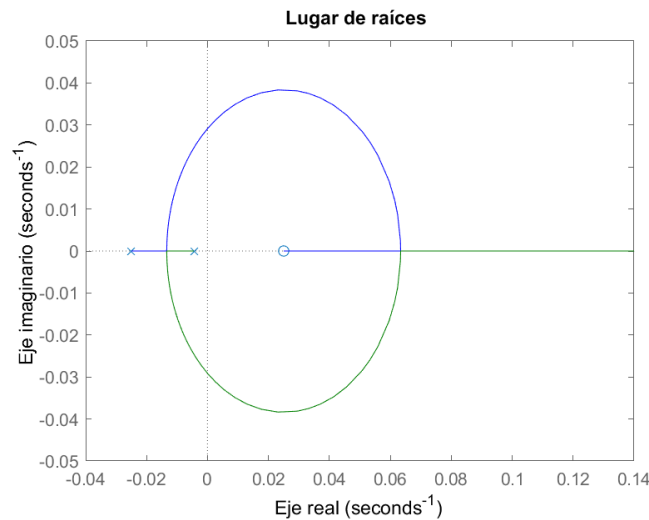
Remplazamos $Q_1(-\theta s)$ y $Q_1(\theta s)$ en la ecuación (10):

$$R_1(s) = \frac{1 - 45s}{1 + 45s} \quad (12)$$

De tal forma $R_1(s)$, reemplaza a la parte exponencial de la ecuación (9); siendo así, que se incluye un polo y un cero al lugar de raíces (Ecuación 13), para asemejar el retardo que tiene la función de transferencia. Dentro del comportamiento del sistema se presenta una oscilación negativa (Figura 42), pero se puede descartar su efecto debido a que su acción transcurre únicamente en el tiempo de retraso del comportamiento del proceso. Este pico negativo es debido a la inclusión de un cero a la parte derecha del eje imaginario y real del lugar de raíces (Figura 41).

Figura 41

Lugar de raíces de la función polinómica



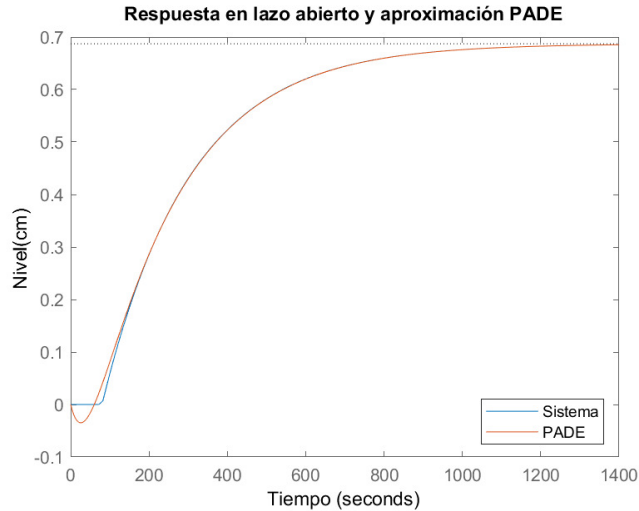
Nota: En la figura se puede visualizar la ubicación del cero y de los polos, dados por la aproximación Pade en la función de transferencia. Además, se evidencia que se encuentra un cero en la parte derecha de los ejes; mismo que no influye en la dinámica del sistema, por lo cual, la oscilación negativa presente en el tiempo de retardo se puede ignorar.

La aproximación PADE cumple con su objetivo, ya que la dinámica del sistema de la función de transferencia anterior se asemeja a una expresión polinómica (Figura 42); de tal forma, el proceso es representado por una nueva función de transferencia (Ecuación 13), en torno a la cual se trabajara para hallar las ganancias del control.

$$G_p(s) = \frac{-0.6871s + 0.01718}{223.4s^2 + 6.584s + 0.025} \quad (13)$$

Figura 42

Aproximación de la función de transferencia a una función polinómica



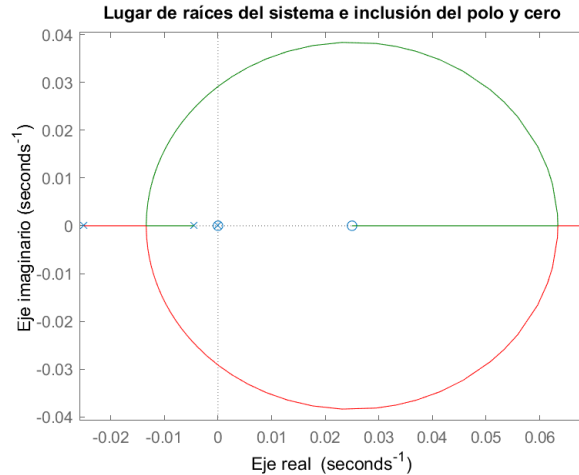
Nota: En la figura se puede observar la aproximación de la función de transferencia que representa el sistema, a una función polinómica por el método de PADE.

Una vez establecida la función para hallar las ganancias PI del controlador, se propone utilizar el método de asignación de polos, permitiendo incluir en el lugar de raíces del sistema un polo y un cero (Figura 43), a partir de la búsqueda de los valores proporcional e integral que antecede a la planta en el lazo de control cerrado.

$$G_{pc}(s) = \frac{-0.0030761s(s - 0.025)}{s(s + 0.025)(s + 0.004477)} \quad (14)$$

Figura 43

Lugar de raíces del sistema e inclusión del polo y cero



Nota: En la presente gráfica se puede visualizar el lugar de raíces de la función de transferencia, más la inclusión de un polo y un cero en el origen.

La ecuación 14 y la figura 43, muestran la inclusión de un polo y un cero en el origen del lugar de raíces; a partir de esto, se aplicará el método de asignación, para relocalizar los polos y los ceros que ayuden a concebir que el controlador PI mejore el error en estado estacionario, y en lo posible minimice el deterioro del desempeño de la parte transitoria que actualmente maneja el sistema, por lo cual se tiene las siguientes consideraciones:

Ceros

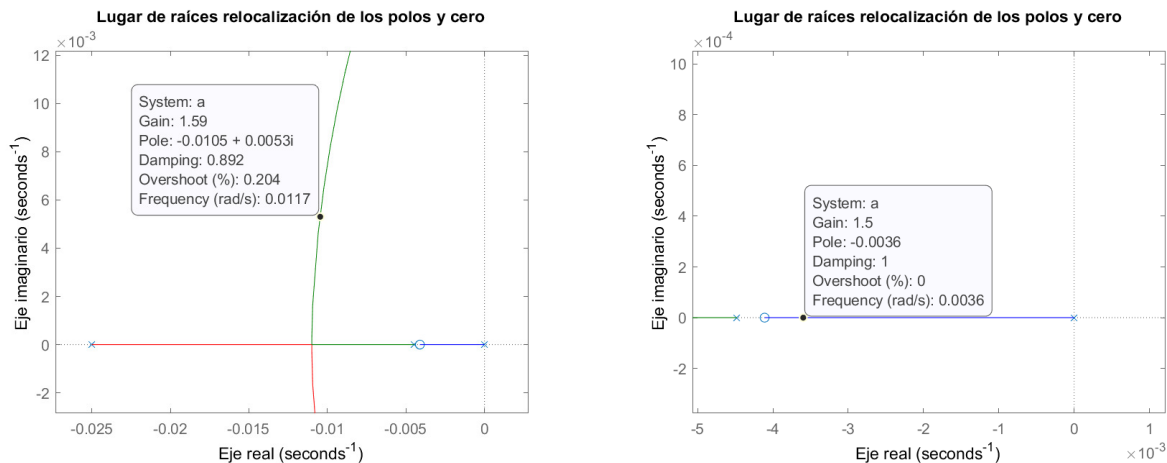
- El cero de la parte derecha del lugar de raíces no será modificado, ya que en un aspecto anterior se lo descartó en la influencia del controlador.
- Para el nuevo cero que se introduce en el lugar de raíces; será planteado lo más cercano posible al punto de origen, de tal manera se precautela la ganancia que inicialmente trae el sistema en el lazo de control abierto; es decir, se debe de proponer un valor muy pequeño con relación a la planta (Figura 44). Intuyendo la magnitud del sistema se relocaliza el cero en -0.00411 .

Polos

- Los polos que se buscan introducir en el lugar de raíces, son dos polos complejos conjugados y un polo real; esta configuración es propuesta para garantizar que el controlador mantenga una robustez, y un buen tiempo de respuesta en el sistema.
- Con respecto a la relocalización del polo real (Figura 44), se lo ubicará en la parte derecha de los polos complejos conjugados para que se convierta en el polo dominante, y se evite tener oscilaciones o cambios en la parte transitoria del sistema. Con un factor de proporcionalidad cercano a 1, estará situado en -0.0036 .
- Se debe de evitar que el proceso se vuelva lento a causa del punto anterior, por lo cual se seleccionan dos polos complejos conjugados con un overshoot del 0.2% (Figura 44); cabe destacar que la consigna del diseño del controlador proponía evitar que se tenga sobre disparos, pero más haya de producir que se altere la dinámica del proceso, este porcentaje cercano a cero ayudara a que el control sea robusto sin tener que introducirnos en una segunda relocalización en torno a la curva del diagrama de Nyquist, además, al tener el overshoot un valor pequeño no tiene efecto en la sección transitoria. De tal manera los polos se ubican en $-0.0105 \pm 0.0053i$.

Figura 44

Lugar de raíces de la relocalización de los polos y del cero



Nota: Se muestra la asignación de polos y ceros para el control PI.

Mediante el método de asignación de polos se hallará las ganancias que requiere la planta para que se relocalicen los parámetros anteriores en el lugar de raíces, para ello se parte de calcular la función equivalente del sistema en lazo cerrado con la inclusión de las constantes del PID. El siguiente proceso describe el fundamento matemático:

- a) Se plantea las fórmulas que representan el lazo cerrado del sistema (Ecuación 15) y el control PID (Ecuación 16).

$$T(s) = \frac{y(s)}{r(s)} = \frac{C(s) G_p(s)}{1 + C(s) G_p(s)} \quad (15)$$

$$C(s) = \frac{k_d s^2 + k_p s + k_i}{s} \quad (16)$$

- b) Remplazamos $T(s)$ con los parámetros necesarios, y se halla la función característica (Ecuación 19)

$$T(s) = \frac{\left(\frac{k_d s^2 + k_p s + k_i}{s}\right) \left(\frac{-0.6871s + 0.01718}{223.4s^2 + 6.584s + 0.025}\right)}{1 + \left(\frac{k_d s^2 + k_p s + k_i}{s}\right) \left(\frac{-0.6871s + 0.01718}{223.4s^2 + 6.584s + 0.025}\right)} \quad (17)$$

$$T(s) = \frac{\left(\frac{-0.6871k_d s^3 + 0.01718k_d s^2 - 0.6871k_p s^2 + 0.01718k_p s - 0.6871k_i s + 0.01718k_i}{223.4s^3 + 6.584s^2 + 0.025s}\right)}{1 + \frac{-0.6871k_d s^3 + 0.01718k_d s^2 - 0.6871k_p s^2 + 0.01718k_p s - 0.6871k_i s + 0.01718k_i}{223.4s^3 + 6.584s^2 + 0.025s}} \quad (18)$$

$$T(s) = \frac{-0.6871k_d s^3 + 0.01718k_d s^2 - 0.6871k_p s^2 + 0.01718k_p s - 0.6871k_i s + 0.01718k_i}{(-0.6871k_d + 223.4)s^3 + (-0.6871k_p + 0.01718k_d + 6.584)s^2 + (-0.6871k_i + 0.01718k_p + 0.025)s + 0.01718k_i} \quad (19)$$

Teniendo los polos y el cero a los que se quiere llegar en el lugar de raíces con el controlador, se puede proponer el polinomio característico deseado (Ecuación 20).

$$P_d(s) = 223.4s^3 + 5.489s^2 + 0.04788s + 0.0001124 \quad (20)$$

Al cerrar el lazo de control se encuentra el denominador que corresponde al polinomio característico (Ecuación 19); mismo que al ser igualado al polinomio característico deseado (Ecuación 20), se puede plantear un sistema de ecuaciones con tres incógnitas.

$$(-0.6871k_d+223.4)s^3+(-0.6871k_p+0.01718k_d+6.584)s^2+(-0.6871k_i+0.01718k_p+0.025)s+0.01718k_i = 223.4s^3 + 5.489s^2 + 0.04788s + 0.0001124 \quad (21)$$

Igualando los términos de igual potencia, y al interesarnos únicamente las ganancias proporcional e integral, se tiene las siguientes ecuaciones (Ecuación 22 y 23):

$$-0.6871k_i + 0.01718k_p + 0.025 = 0.04788 \quad (22)$$

$$0.01718k_i = 0.0001124 \quad (23)$$

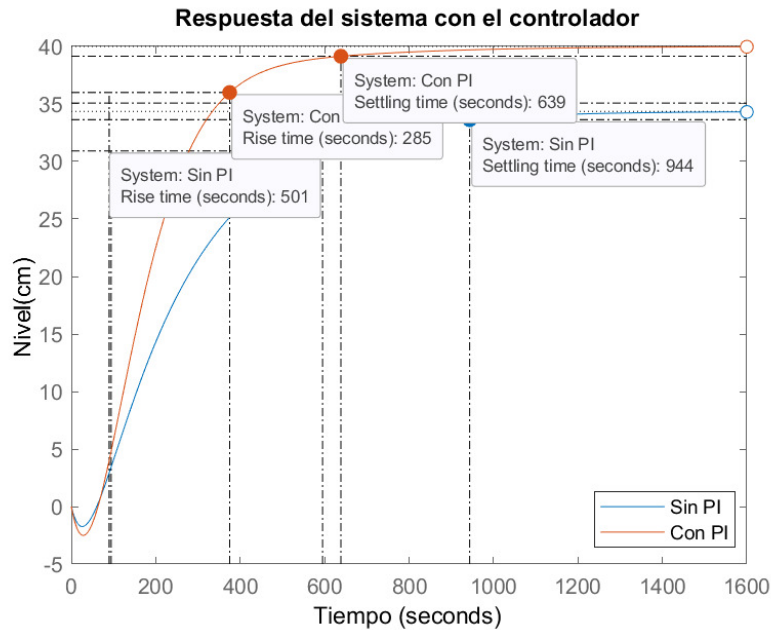
$$k_i = \frac{0.0001124}{0.01718} = 0.006546 \quad (24)$$

$$k_p = \frac{0.04788 - 0.025 + 0.6871(0.006546)}{0.01718} = 1.594 \quad (25)$$

Finalmente resolviendo las ecuaciones, se puede delimitar que la ganancia proporcional será de 1.594 y la ganancia integral de 0.006546. Introduciendo los valores hallados en el lazo de control cerrado, se obtiene una buena respuesta en la simulación del sistema; en el cual, el tiempo de levantamiento y el tiempo de asentamiento mejoran con respecto al sistema en lazo abierto; también, se consigue verificar que las constantes PI aluden a que el proceso alcance el valor de referencia de 40 cm (Figura 45).

Figura 45

Respuesta del sistema con el controlador



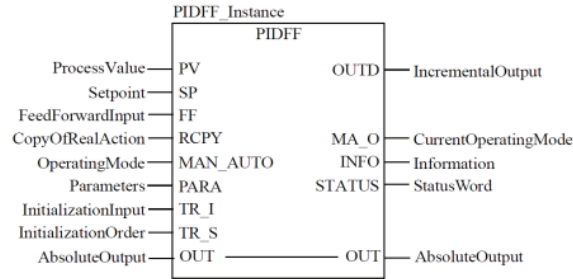
Nota: En la figura se presenta la respuesta del sistema en lazo cerrado con un control con ganancias proporcional e integral que sigue un valor de referencia de 40 cm de nivel. También, se evidencia el sistema en lazo abierto sin controlar, con una entrada de extinción del 50% de la apertura de la válvula.

6.3.7. Implementación del control PID

Para poder ingresar el algoritmo de control diseñado a la planta, lo realizamos mediante el PLC Schneider. Dentro del software que se utiliza para la configuración del autómata programable (Unity Pro XL) se mantiene varias librerías con varias funciones que facilitan la introducción de las ganancias calculadas al proceso, entre ellas, se puede encontrar el bloque PIDFF (Figura 46), mismo que se basa en una arquitectura proporcional, integral, derivativa con estructura mixta o paralela (serie/paralela). El bloque da la opción de controlar un vasto abanico de procesos, debido a sus múltiples funciones englobadas en una sola librería. Consta de 5 salidas y 8 entradas.

Figura 46

Bloque PIDFF



Nota: En la figura se presenta la representación del bloque PIDFF.

Con respecto a las entradas, disponemos de una variable (PV) que hace referencia al valor de la retroalimentación del proceso a controlar; la consigna de referencia será introducida en (SP). Además, se cuenta con una entrada de interrupción (FF), con una copia para la variable manipulada (RCPY) y con una configuración de bit para activar el modo automático o manual (MAN_AUTO). El parámetro (PARA) engloba una estructura de datos. La entrada (TR_I) según el modo será considerado o ignorado para introducir un valor inicial. Por último (TR_S) ayudará a la inicialización del bloque.

Por otra parte, en las salidas disponemos de (OUT) para la señal de actuación del control PID. (OUTD) entregará un valor incremental de la diferencia entre el ciclo nuevo y el anterior. Con (INFO) y (STATUS) se tendrá la información del bloque. Finalmente (MA_O), servirá como un indicador del modo de funcionamiento en el que se mantiene trabajando el controlador.

Como se mencionó anteriormente, la estructura PARA posibilita ingresar al bloque los datos relacionados con el algoritmo de control calculado, por lo que es prescindible configurarlo correctamente. La tabla 6, muestra las constantes utilizadas para la implementación del controlador en el proyecto. Por otro lado, las constantes que demanda la herramienta a partir del diseño es la ganancia proporcional, el tiempo integral y derivativo, por lo cual se realiza la conversión de la ganancia integral.

$$T_i = \frac{k_p}{k_i} = \frac{1.594}{0.006546} = 243.5075 \quad (26)$$

Tabla 6*Parámetros PARA (PIDFF)*

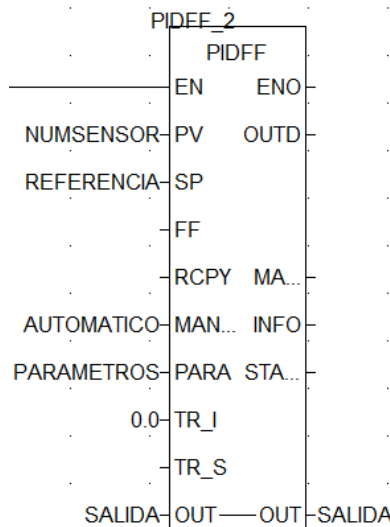
Nombre	Valor	Descripción
id	automático	Reservado para el ajuste automático.
pv_inf	0	Valor mínimo que el proceso puede tomar.
pv_sup	100	Valor máximo que el proceso puede tomar.
out_inf	0	Valor mínimo de la salida del PID.
out_sup	100	Valor máximo de la salida del PID.
rev_dir	0	Acción opuesta del controlador PID.
mix_par	0	Controlador PID con estructura mixta.
aw_type	0	Sin anti-windup.
en_rcpy	0	No se utiliza la entrada RCPY.
kp	1.594	Ganancia proporcional.
ti	T#243.5075s	Tiempo integral.
td	T#0s	Tiempo derivativo.
kd	0	Ganancia diferencial.
pv_dev	0	Contribución diferencial en relación al error.
bump	0	Modo anti-brusquedad activado.
dband	0	Sin área muerta en desviación.
gain_kp	0	Reducción de la contribución proporcional.
ovs_att	0	Modifica el peso del SP.
outbias	0	Introduce un offset a la salida del PID.
out_min	0	Límite inferior de la salida.
out_max	100	Límite superior de la salida.
outrate	0	Límite de modificación de salida por seg.
ff_inf	0	Límite inferior del rango FF.
ff_sup	0	Límite superior del rango FF.
otff_inf	0	Límite inferior del rango out_f.
otff_sup	0	Límite superior del rango out_f.

Nota: En la tabla se puede observar los valores y las constantes del registro PARA, utilizado en el bloque PIDFF. Adaptado de Control borroso industrial: Uso de la IEC 1131-7 para el control de plantas industriales, Pajarón (2012). p, 72. © 2012 Universidad de Sevilla.

Una vez establecido los parámetros principales que debe de llevar el bloque PIDFF, es implementado en el PLC con cada uno de sus datos. La figura 47, muestra la configuración; y la tabla 7, mantiene la descripción de cada una de las variables ingresadas en la función PIDFF.

Figura 47

Implementación del bloque PIDFF



Nota: En la figura se observa la implementación del bloque PIDFF, con cada una de sus variables utilizadas.

Tabla 7

Variables introducidas en el bloque PIDFF

Nombre	Descripción
EN	Se da la activación o desactivación del bloque en el proceso.
NUMSENSOR	La variable ingresa la retroalimentación al lazo cerrado. Sensor de nivel Cerabar T PMC131.
REFERENCIA	Esta variable establece el valor de referencia de 40 cm de nivel.
AUTOMATICO	Se activa con un bit ("1") el modo automático del bloque.
PARAMETROS	Se carga los parámetros de la tabla .
SALIDA	Por medio de esta variable se obtiene la señal de actuación para la válvula proporcional.

Nota: La tabla muestra las variables utilizadas en la implementación del algoritmo de control en el PLC Schneider, por medio del bloque PIDFF.

6.4. Manual de prácticas y guías de aprendizaje

El formato para los manuales y guías de prácticas de laboratorio, se fundamentó en los criterios establecidos por el consejo académico de la Universidad Politécnica Salesiana. En torno a ello se plantean los siguientes aspectos:

- a) **Título de la práctica:** Debe ser interesante para que logre atraer al estudiante, además de dar una idea sobre lo que se va a realizar.
- b) **Objetivos de la práctica:** Los objetivos se plantearán como pasos y procedimientos que se deben de realizar para llegar a concluir la práctica.
- c) **Requisitos y conocimientos previos:** Se establecen los conceptos y requerimientos que necesita tener el estudiante para poder llevar a cabo práctica de laboratorio sin ningún problema o contratiempo.
- d) **Recursos, equipo y materiales:** Se enlista los materiales y equipos que va a necesitar el estudiante para ejecutar la práctica de laboratorio.
- e) **Exposición:** Es un apartado teórico, donde se explican los conceptos básicos e importantes para que los estudiantes puedan realizar la práctica. Este apartado cuenta con bibliografía o referencias que respalden los aspectos abordados en el procedimiento del manual de laboratorio.
- f) **Descripción del desarrollo de la práctica:** Se explican los pasos a seguir en el laboratorio. Debe de estar ordenado y contener gráficos, tablas y esquemas para que sea más fácil para el estudiante el cumplimiento de la práctica.
- g) **Actividades por desarrollar:** Es un parámetro propuesto para que el estudiante pueda generar un autoaprendizaje e ir más allá de solo realizar la práctica. Creando así un nuevo en torno al proceso realizado.
- h) **Resultados:** En este apartado se debe de colocar lo obtenido en el desarrollo de los pasos planteados en los puntos anteriores; se busca con esto que es estudiante acoja una postura crítica y genere una discusión alrededor del desarrollo de la práctica.
- i) **Conclusiones:** A partir de los resultados dados, se busca que el estudiante realice un breve informe donde se expondrá si se logró o no alcanzar los objetivos, además, se debe de justificar lo obtenido en el desarrollo de la práctica con la parte teórica, tratando de autoevaluar y establecer mejor los conocimientos impartidos en clase.
- j) **Bibliografía:** Son las referencias de donde se obtuvo la teoría sobre la cual se va basa la práctica, estas deben seguir un formato, ya sea APA o IEEE.

7. Resultados

7.1. Variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A

7.1.1. Variables

La variable que se utilizó en el desarrollo del proyecto es la de nivel, porque si bien las otras variables fueron monitoreadas mediante el PLC, el nivel de los tanques es la constante requerida para el control implementado.

7.1.2. Proceso

El proceso utilizado es el llenado del tanque abierto con un control PID, y el monitoreo del tanque cerrado. Para esto se controla la bomba mediante el variador de frecuencia, se activan las válvulas solenoides para dirigir el flujo de entrada, se mantienen los sensores de nivel a fin de tener una retroalimentación y se maneja la válvula proporcional Jordan para controlar el caudal de ingreso en los tanques.

7.1.3. Instrumentos

Los instrumentos que se utilizaron para el control y monitoreo del proceso son los expuestos en la metodología. De tal forma, mediante pruebas en la planta se llegó a determinar cada una de las direcciones de los equipos que se pueden manipular a partir del PLC Schneider; en las siguientes tablas se exponen las salidas y entradas, tanto analógicas como digitales del autómatas programable.

Tabla 8*Salidas digitales*

N°	Elemento	Dirección	Comentario
1	EV1-02	%Q0.3.0	Válvula neumática (On/Off). Interconexión de los tanques.
2	EC 3-05	%Q0.3.1	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
3	EV1-01	%Q0.3.2	Válvula neumática (On/Off) del tanque atmosférico.
4	EC 1-01	%Q0.3.3	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Hart.
5	EC 1-03	%Q0.3.4	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
6	EV1-03	%Q0.3.5	Válvula neumática (On/Off) del tanque cerrado.
7	EC 3-01	%Q0.3.6	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Hart.
8	EC 4-02	%Q0.3.7	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
9	EA 1-04	%Q0.3.8	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
10	EA 1-03	%Q0.3.9	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
11	EC 3-02	%Q0.3.10	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Fieldbus.
12	EC 3-03	%Q0.3.11	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
13	EA 1-02	%Q0.3.12	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
14	EC 3-04	%Q0.3.13	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
15	EC 1-02	%Q0.3.14	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
16	EA 1-01	%Q0.3.15	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Hart.
17	EC 4-01	%Q0.3.16	Válvula solenoide (On/Off), ubicada en la línea del protocolo de comunicación Profibus.
18	Variador de Profibus	%Q0.3.17	Enciende o apaga el variador Micromaster N° 3.
19	Variador de Hart	%Q0.3.18	Enciende o apaga el variador Micromaster N° 1.
20	Variador de Fieldbus	%Q0.3.19	Enciende o apaga el variador Micromaster N° 2.
21	Motor variador de Hart	%Q0.3.20	Activa o desactiva la bomba N° 1.
22	Motor Variador de Profibus	%Q0.3.21	Activa o desactiva la bomba N° 3.
23	Motor variador de Fieldbus	%Q0.3.22	Activa o desactiva la bomba N° 2.
24	Luz indicadora verde	-	No existe acceso.
25	Luz indicadora naranja	-	No existe acceso.
26	Luz indicadora roja	-	No existe acceso.

Nota: En la tabla se puede observar las salidas digitales que mantiene el PLC Schneider de la planta A, del laboratorio de LACTI.

Tabla 9*Entradas digitales*

N°	Elemento	Dirección	Comentario
1	LS300	%I0.2.0	Interruptor de nivel en alto del depósito.
2	LS100	%I0.2.1	Interruptor de nivel en alto del tanque abierto.
3	LS106	%I0.2.2	Interruptor de nivel en bajo del tanque cerrado.
4	LS301	%I0.2.3	Interruptor de nivel en bajo del depósito.
5	LS102	%I0.2.4	Interruptor de nivel en bajo del tanque abierto.
6	LS104	%I0.2.5	Interruptor de nivel en alto del tanque cerrado.
7	HV/C2	%I0.2.6	Apertura de la válvula manual del tanque cerrado, ubicada en la parte superior.
8	HV/C5	%I0.2.12	Apertura de la válvula manual del tanque abierto, ubicada en la parte inferior.
9	HV/C4	%I0.2.8	Apertura de la válvula manual del tanque cerrado, ubicada en la parte posterior.
10	HV/C1	%I0.2.11	Apertura de la válvula manual del tanque cerrado, ubicada en la parte inferior.
11	HV/C2	%I0.2.10	Cierre de la válvula manual del tanque cerrado, ubicada en la parte superior.
12	HV/C1	%I0.2.9	Cierre de la válvula manual del tanque cerrado, ubicada en la parte inferior.
13	HV/C5	%I0.2.7	Cierre de la válvula manual del tanque abierto, ubicada en la parte inferior.
14	HV/C4	%I0.2.13	Cierre de la válvula manual del tanque cerrado, ubicada en la parte posterior.
15	Paro de emergencia	%I0.2.14	Pulsador ubicado en el panel frontal.

Nota: La tabla presenta las salidas analógicas que mantiene el PLC Schneider de la planta A, del laboratorio de LACTI.

Tabla 10*Salidas analógicas*

N°	Elemento	Dirección	Comentario
1	Valvula proporcional Jordan	%QW0.5.0	Ingreso del porcentaje de apertura de la válvula proporcional.
2	Valvula proporcional aire	%QW0.5.2	Ingreso del porcentaje de apertura de la válvula proporcional.
3	Consigna frecuencia Hart	%QW0.5.6	Ingreso del valor de la frecuencia para el Micromaster N° 1.
4	Consigna de frecuencia Fielbus	%QW0.5.7	Ingreso del valor de la frecuencia para el Micromaster N° 2.

Nota: En la tabla se puede observar las salidas digitales que mantiene el PLC Schneider de la planta A del laboratorio de LACTI.

Tabla 11*Entradas analógicas*

N°	Elemento	Dirección	Comentario
1	Sensor de presión de aire comprimido	%Iw0.4.0	Sensor ubicado en el tanque cerrado.
2	Sensor de nivel	%Iw0.4.1	Sensor Cerabar T PMC131, ubicado en la parte inferior del tanque abierto.

Nota: En la tabla se observa las entradas analógicas conmutadas al PLC Schneider de la planta A, del laboratorio de LACTI.

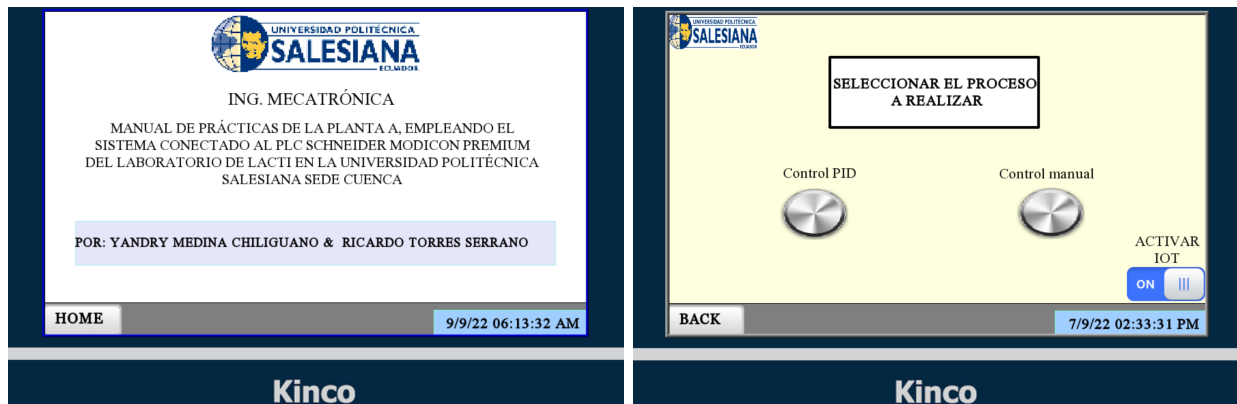
7.2. Control y monitoreo de la Planta A

7.2.1. Pantalla HMI

Como se mencionó dentro de la metodología, la pantalla Kinco más allá de ayudarnos como un gateway entre el concentrador HCS y el PLC Schneider, es utilizado para poder controlar y monitorear la planta desde el laboratorio. El diseño del entorno gráfico inicia por mostrar una pantalla principiará que mantiene información del proyecto; al introducirnos al menú de Home se muestra las opciones con la que se puede controlar el sistema. Esta subpantalla permite escoger si se desea utilizar un control PID o un manejo manual; además, por cuestiones de seguridad se establece un botón para activar y desactivar el IOT (Figura 48).

Figura 48

Pantallas de inicio del HMI

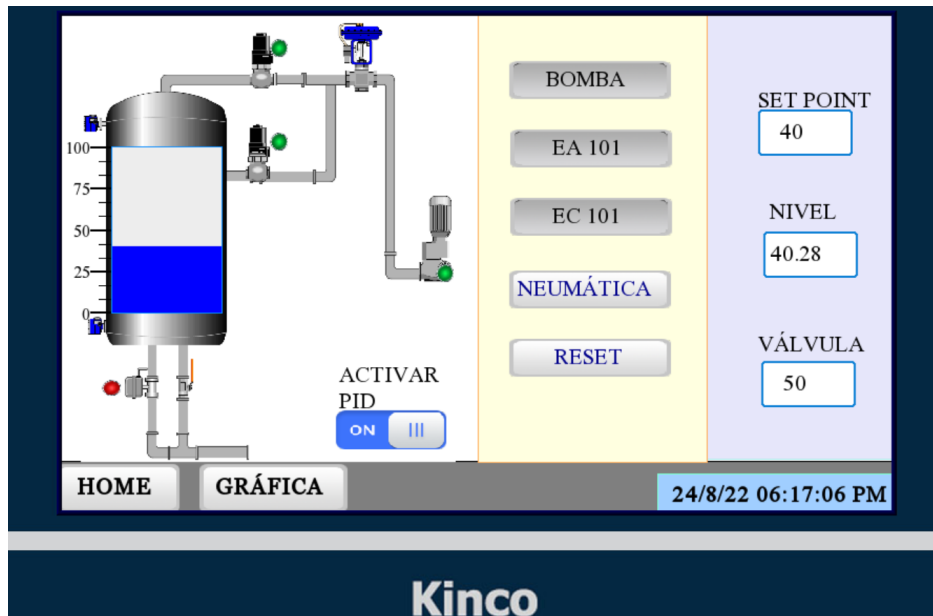


Nota: En las figuras se muestra el entorno gráfico programado para el HMI. En la primera pantalla se da una presentación del proyecto, y en la segunda pestaña se propone el menú para el control de la planta.

Dentro del control y monitoreo PID, la pantalla propone un botón de activación para seguridad del proceso (Figura 49). Se mantienen pulsadores para manipular la planta, una sección para ingresar un valor de referencia, y displays que muestran la apertura de la válvula proporcional y la el nivel del tanque. Cabe mencionar, que se puede ver la gráfica del comportamiento del sistema. El inicio del controlador PID se realiza al activar la bomba, y su reinicio al pulsar el botón de reset.

Figura 49

Pantalla PID en el HMI

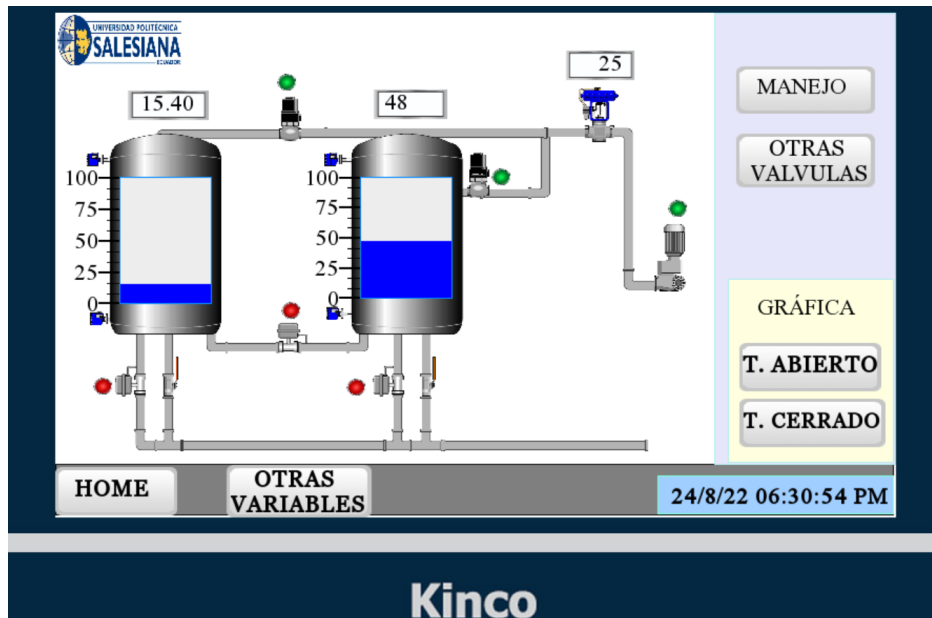


Nota: En la figura se encuentra la ventana de PID en el HMI, para el control y monitoreo del proceso.

Para el control manual de la planta se tiene dos tanques que indican el nivel de agua, y gráficas para ver el comportamiento del sistema (Figura 50). El manejo del proceso se lo realiza a partir de una pantalla emergente, donde se activan las válvulas solenoides y neumáticas, se introduce la apertura de la válvula proporcional, y se activa la bomba (Manejo). Cabe señalar, que la ventana tiene la opción de presentar las variables de presión, nivel, temperatura y caudal que se encuentran conectadas al concentrador Hart (Otras variables).

Figura 50

Pantalla para el control manual en el HMI



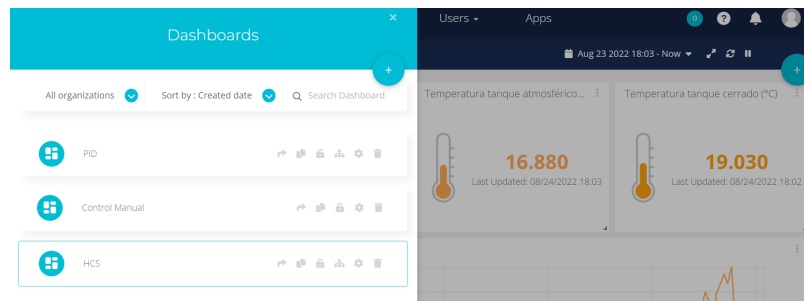
Nota: Se presenta la ventana para el control manual en el HMI implementado en la planta A, del laboratorio de LACTI.

7.2.2. Ubidots

Para el control y monitoreo de la planta aplicando IOT, el operador debe de ingresar a la página de Ubidots desde cualquier ordenador o dispositivo; colocando el usuario y contraseña ingresaremos a un menú donde se puede seleccionar el modo de trabajo, teniendo un control PID o un manejo manual. Además, dentro de esta plataforma se tiene la opción de visualizar las variables del HCS (Figura 51).

Figura 51

Menú principal de Ubidots

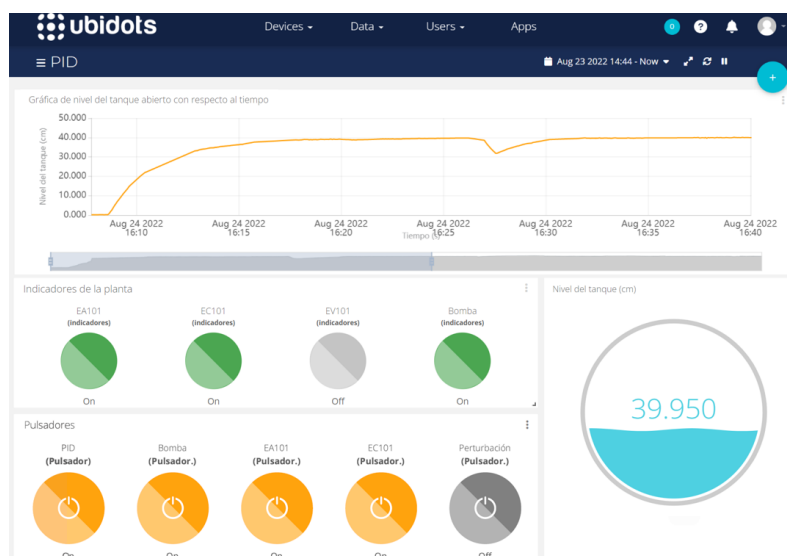


Nota: En la figura se presenta las tres modalidades de la interfaz de Ubidots para visualizar y controlar las variables del PLC Schneider.

En la ventana PID se visualiza el monitoreo del proceso de control implementado en la planta A (Figura 52); dentro de esta, se mantiene un bloque de indicadores para observar los elementos que se tienen activados en el proceso. De igual forma que el HMI, se dispone de pulsadores para manipular las válvulas y la bomba. Además, se cuenta con una gráfica en el que se expone el comportamiento dinámico del sistema, y un display que indica el nivel de agua que alcanza el tanque abierto.

Figura 52

Ventana PID en Ubidots.



Nota: En la figura se visualiza la interfaz de la ventana PID para el control y monitoreo de las variables del proceso de llenado de tanque.

En la ventana de control manual se observa el monitoreo del tanque abierto sin el PID; dando la opción de manipular todas las válvulas y la bomba de la planta A. De igual manera, existen indicadores para cada uno de los elementos del proceso. Se proporciona un display, y un gráfico para visualizar el nivel del tanque (Figura 53).

Figura 53

Ventana del control manual en Ubidots

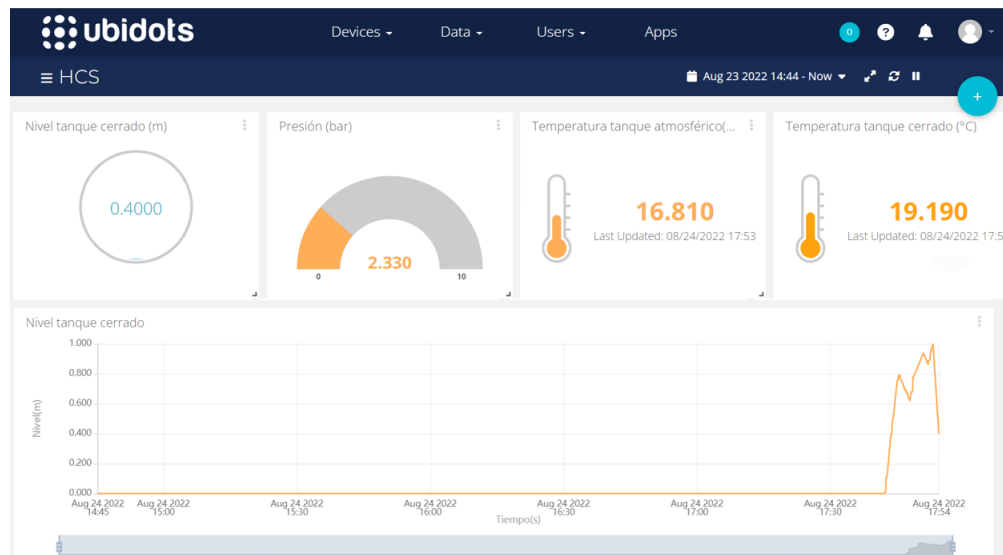


Nota: Se presenta la interfaz de la ventana manual para el monitoreo de las variables del proceso de llenado de tanque.

En la ventana HCS se observan los datos enviados del concentrador HART. De modo que, se encuentra en la dashboard el valor censado de la temperatura de cada tanque, el dato de presión de la tubería y el nivel del tanque presurizado. (Figura 54).

Figura 54

Ventana HCS en Ubidots



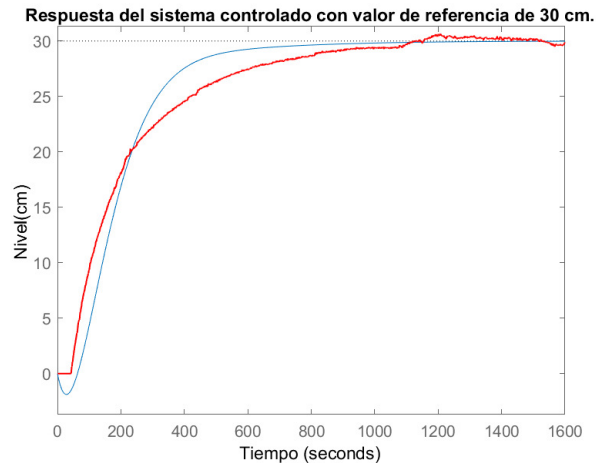
Nota: En la figura se encuentra la interfaz de la ventana HCS para el monitoreo de las variables del concentrador Hart.

7.3. Algoritmo de control (PID)

Con la constitución del controlador proporcional – integral, se ingresa en el bloque PIDFF que dispone el PLC Schneider, las ganancias obtenidas en la metodología, y se verifica la respuesta. Para validar el funcionamiento del control diseñado con respecto al sistema físico se realizan tres pruebas con distintos valores de referencia. Cabe mencionar, que se trabaja en torno al punto en que se generó la toma de muestras para hallar la planta. Si bien nos interesa obtener 40 cm en el nivel del tanque, la respuesta para 50 cm y 30 cm son bastante aceptables.

Figura 55

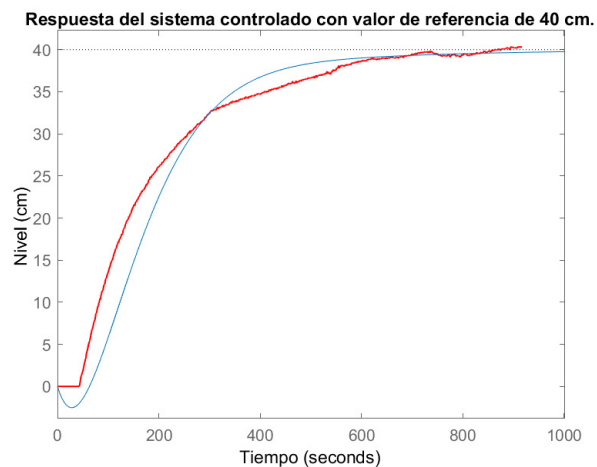
Respuesta del sistema controlado con valor de referencia de 30 cm.



Nota: En la presente figura se puede observar la respuesta del sistema físico, como también, la respuesta de la simulación implementando un controlador PI para seguir un valor de referencia de 30 cm de nivel de tanque.

Figura 56

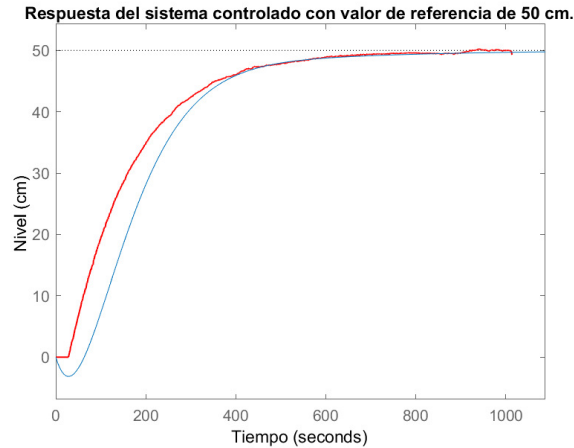
Respuesta del sistema controlado con valor de referencia de 40 cm.



Nota: Se presenta la respuesta del sistema físico, como también, la respuesta de la simulación implementando un controlador PI para seguir un valor de referencia de 40 cm de nivel de tanque.

Figura 57

Respuesta del sistema controlado con valor de referencia de 50 cm.



Nota: Se muestra la respuesta del sistema físico, como también, la respuesta de la simulación implementando un controlador PI para seguir un valor de referencia de 50 cm de nivel de tanque.

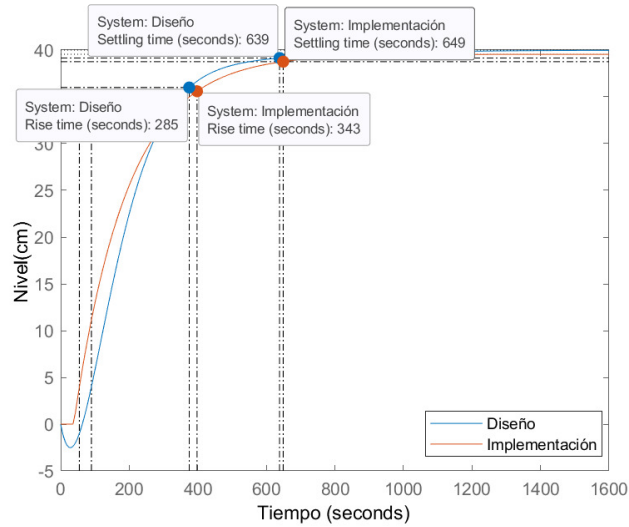
Dentro de las figuras 55, 56 y 57, se puede ver la similitud de la respuesta. Las pequeñas variaciones que existe entre la salida del sistema físico y el diseño, se debe a que la simulación del modelo de la función de transferencia es idealizada, por lo cual presenta cierta discrepancia del sistema real. Además, factores como la densidad del agua, clima, histéresis de las válvulas y purgado de la bomba, son otros de los factores que tendrán una mínima incidencia en del proceso.

Para poder abordar el análisis de tiempos se utiliza la herramienta de Matlab pidtuner para convertir la salida del sistema físico en una función de transferencia. De esa manera, para la consigna de planteamiento de 40 cm de nivel en el tanque; se presenta un sistema con un overshoot de 0, sin tiempo pico, con un tiempo de respuesta en la simulación de 285 s y en la implementación de 343 s. El tiempo de asentamiento varía 10 s entre el diseño y la parte física por lo cual el control PI es aceptable (Figura 58).

En la parte transitoria se tiene una desfase entre las dos curvas, pero no se da mayor énfasis porque el valor de retraso en la función de transferencia simulada, es genérico para el proceso. Con respecto a la sección estacionaria del sistema se obtiene un error en estado estable de ± 0.7 , esto deduce que el proceso tiende a seguir el valor referencial, haciendo que el nivel se mantenga incluso si se produce una perturbación (Figura 59).

Figura 58

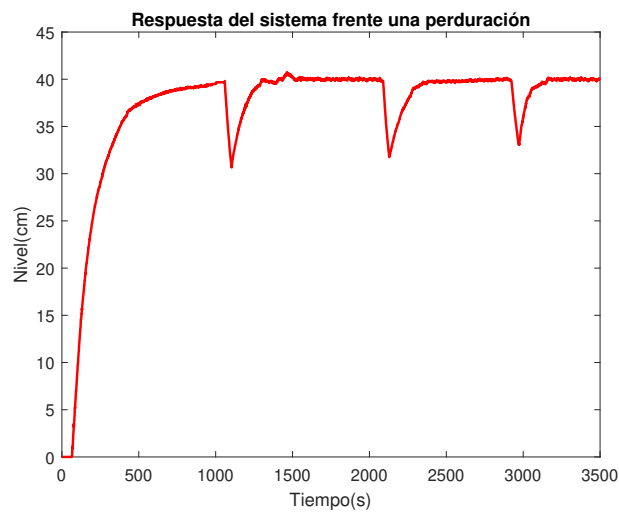
Tiempo de asentamiento y tiempo de levantamiento del sistema.



Nota: En la figura se puede apreciar el tiempo de asentamiento y el tiempo de levantamiento que mantiene el sistema, para un valor de referencia de 40 cm.

Figura 59

Respuesta del sistema controlado, frente a una perturbación.



Nota: Se puede evidenciar la respuesta que tiene el controlador, frente a perturbaciones. El valor de referencia estipulado es de 40 cm.

7.4. Manual de prácticas y las guías de aprendizaje de la planta A

Con los objetivos realizados se plantean 5 guías prácticas con los siguientes temas:

- Configuración del PLC Schneider con el uso del programa Unity Pro XL.
- Manejo de la planta mediante el PLC Schneider.
- Conexión Modbus con instrumentos de la planta A.
- Control de nivel en la planta A del laboratorio de LACTI.
- IOT mediante el PLC Schneider y el módulo IOT2040.

Las guías realizadas se encuentran en el anexo B. Para validar las prácticas se utilizaron las encuestas del anexo C, y se las aplico a docentes de asignaturas relacionadas con el proyecto de titulación. De tal forma, se obtuvo el visto bueno para que los manuales y guías de practicas de laboratorio sean aplicados en la formación profesional de los estudiantes.

8. Conclusiones

- En la elaboración de las guías prácticas se utilizó la línea Hart de la planta A, debido a que se tenía la consigna de habilitar una segunda línea diferente al protocolo de comunicación Profibus. Por otra parte, los instrumentos empleados son los que se encuentran conectados directamente con el PLC Schneider, como también los que llegan al concentrador HCS; de tal manera se puede recibir la información en torno a las variables de nivel, caudal, presión y temperatura. El proceso realizado en la planta A, se enfocó en implementar un control de la arquitectura PID en el tanque abierto, mientras que en el tanque presurizado, al haber realizado una red para obtener los datos de ciertos transmisores y transductores del sistema, se puede realizar un manejo manual de la cantidad de agua en el contenedor cerrado.
- Se realizaron 2 interfaces para controlar y monitorear la planta A, en el caso del HMI al tener una conexión por TCP/IP la respuesta es menor a 1 segundo, lo que ayuda a tener una mayor reacción en caso de algún problema. En cambio, la interfaz con Ubidots, al enviar los datos por internet tiene una respuesta de 1 a 2 segundos de retraso debido a las restricciones de la Universidad, por lo que su utilidad será enfocada para monitorear a largas distancias.

- Dentro del diseño del algoritmo de control se utilizó la arquitectura PID, donde se aplicó únicamente una ganancia proporcional e integral debido a que el problema del sistema en lazo cerrado radicaba en su estabilidad, y no en la parte transitoria. Las constantes obtenidas en el modelamiento del control se ingresaron en la planta por medio del PLC Schneider, generando un alto grado de similitud entre el modelo matemático y la implementación física del sistema; dando así que el tiempo de asentamiento entre las dos partes tengan una variación de 10 s. Para la dinámica del proceso no se presenta ningún sobrepaso, consiguiendo que la salida del sistema se comporte de manera sobre amortiguado, como el diseño propuesto para el control PI, además, el proceso sigue el valor de referencia, dando así un error en la estabilidad de 2 cm en el caso más crítico.
- La generación del manual de prácticas contribuye al uso de los equipos presentes en el laboratorio LACTI, haciendo que cada uno de los usuarios pueda realizar un proceso de aprendizaje que contribuya a su formación académica. Por otra parte, se rompe la barrera de tener un equipo sin uso, generando la oportunidad de realizar nuevos proyectos en torno al PLC Schneider, además de, utilizar las guías de prácticas como punto de partida en el estudio, análisis e implementación de procesos industriales. Los cinco formatos de prácticas fueron validados para su uso, comprobando su utilidad en asignaturas relacionadas con procesos industriales y sistemas de control totalmente automatizados.

9. Recomendaciones

- Se debe tener un conocimiento previo del funcionamiento de cada equipo antes de poder manipular la planta, lo cual permite saber como actuar ante una posible falla.
- Tomar en cuenta el principio de histéresis, en la válvula proporcional. Ya que el posicionador provoca un cambio en el caudal de entrada del tanque.
- Conocer a cabalidad como se obtiene los paquetes de caracteres de datos en los protocolos de comunicación.
- Al utilizar un módulo IOT, tener en cuenta que se disponga accesibilidad completa por parte del distribuidor del internet para gestionar los datos a la nube; además, priorizar una conexión estable para evitar fallas.

- Calibrar los equipos como el variador de frecuencia, la válvula proporcional, la bomba, los sensores; buscando evitar errores en el proceso.
- Finalmente, el presente trabajo de titulación sirve como punto de partida para futuros proyectos relacionados al PLC Schneider, algunos de los temas que se pueden abordar, es el conmutar los dos PLCs que se encuentran en la planta para que trabajen a la misma vez. Por otro lado, se puede desarrollar un lazo de control en cascada para no manipular únicamente nivel, sino también, caudal o presión. Y la última propuesta, es el de ingresar otra perturbación, ya que en el sistema se está inyectando únicamente una interrupción en el flujo de salida, y sería bastante interesante trabajar con una segunda línea para introducir líquido al control PID.

Referencias

- Alberto, M., Perez, A., y Perez, E. (2007). *Introducción a los sistemas de control y modelo matemático para sistemas lineales invariantes en el tiempo*. Disponible en: <http://dea.unsj.edu.ar/control1/apuntes/unidad1y2.pdf>: Universidad Nacional de San Juan.
- Alfaro, V. (2001). *Identificación de procesos sobreamortiguados utilizando técnicas de lazo abierto* (Vol. 11) (n.º 1 y 2). Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=5206301>: Universidad de Costa Rica.
- Bouchakri, A. (2021). *Diseño e implementación de una aplicación de control de proceso industrial sobre pantalla HMI* (Tesis de pregrado). Universidad Politécnica de Catalunya, España. Disponible en: <https://upcommons.upc.edu/>.
- Burket. (2022). Manual de instrucciones Válvula de asiento inclinado de 2/2 y 2/3 vías [Manual de software informático]. Disponible en: <https://www.burkert.com/en/Media/plm/MAN/MA/MA2100-Manual-ES-ES.pdf>.
- Candelas, F. (2011). *Comunicación con RS-485 y Modbus* (Inf. Téc. n.º 3). Universidad de Alicante.
- Carpio, M. (2013). *Evaluación de desempeño de los controladores digitales PID y predictivo tolerante a fallas, aplicados al control de nivel del líquido en un tanque* (Tesis de maestría). Universidad Politécnica Salesiana, Ecuador. Disponible en: <https://dspace.ups.edu.ec>.
- Castaño, S. (s.f.). Retardo o tiempo muerto de un sistema. Disponible en: <https://controlautomaticoeducacion.com/>.

- Criollo, E. (2015). *Diseño e implementación de un controlador difuso tolerante a falla, aplicado al control de nivel de líquido* (Tesis de maestría). Universidad Politécnica Salesiana, Ecuador. Disponible en: <https://dspace.ups.edu.ec>.
- Danfoss. (2021). Data Sheet Solenoid coil [Manual de software informático]. Disponible en: <https://assets.danfoss.com/documents/188867/AI237186440089en-000802.pdf>.
- Dasgupta, D., y Michalewicz, Z. (1997). *Evolutionary algorithms in engineering applications*. Memphis: Springer Science Business Media.
- Endress+Hauser. (2019a). Technical Information Cerabar S PMC71 [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000124/6263/000/00/Ti383pes.pdf>.
- Endress+Hauser. (2019b). Technical Information Cerabar T PMC131 [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000557/7706/000/07/TI00415PEN1915.pdf>.
- Endress+Hauser. (2019c). Technical Information Deltabar M PMD55 [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000557/8051/000/10/TI00434PEN1814.pdf>.
- Endress+Hauser. (2019d). Technical Information Liquiphant T FTL20 [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000000/1175/000/01/TI364Fen0108.pdf>.
- Endress+Hauser. (2019e). Technical Information Proline Pro-mag 10W [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000000/0584/000/03/TI093DEN1109.pdf>.
- Endress+Hauser. (2019f). Technical Information Temperature Sensor [Manual de software informático]. Disponible en: <https://portal.endress.com/wa001/dla/5000000/5655/000/00/TI270ten0304.pdf>.
- Escobar, H., y Domínguez, G. (2015). *Estudio y elaboración de un manual de prácticas de la planta de control de procesos A del Laboratorio de Control y Automatización totalmente integrado de la Universidad Politécnica Salesiana sede Cuenca* (Tesis de pregrado). Universidad Politécnica Salesiana, Ecuador. Disponible en: <https://dspace.ups.edu.ec>.
- Febres, E. (2001). *Protocolo de comunicación HART para los instrumentos de Flotech s.a* (Tesis de pregrado). Universidad Simón Bolívar, Venezuela. Disponible en: <http://www.usb.ve/>.
- Google. (2022). Universidad Politécnica Salesiana. Disponible en: <https://www.google.com.ec/maps/>.
- Grumstrup, H. (s.f.). Salidas HART. Disponible en: <http://www.texca.com/hart.htm>.
- Minchala, I., y Valdez, A. (2010). Identificación de procesos usando mínimos cuadrados por

lotes.

- Modbus. (2006). *Modbus over serial line specification and implementation guide v1.02*. Modbus-IDA.ORG.
- Moore Industries. (2017). HART-to-MODBUS RTU Converter [Manual de software informático]. Disponible en: <https://www.miinet.com/>.
- National Instruments. (2019). Introducción a Modbus. Disponible en: <https://www.ni.com>.
- Nodo - RED. (2022). Node - RED. Disponible en: <https://nodered.org/>.
- Pajarón, R. (2012). *Control borroso industrial: Uso de la IEC 1131-7 para el control de plantas industriales* (Tesis de pregrado). Universidad de Sevilla, España. Disponible en: *Universidad de Sevilla*.
- Schneider Electric. (2005). *Plataforma de Automatización Modicon Premium*. Telemecanique.
- Schneider Electric. (2012). Unity Pro XL software - safety PLC specifics, operating mode manual [Manual de software informático]. Disponible en: <https://www.se.com/>.
- Schneider Electric. (2018a). Analog Input/Output modules User Manual [Manual de software informático]. Disponible en: https://download.schneider-electric.com/files?p_enDocType=User+guidep_FileName=35010447K0100012.pdfp_DocRef=35010447K01000.
- Schneider Electric. (2018b). Discrete I/O modules User Manual [Manual de software informático]. Disponible en: https://download.schneider-electric.com/files?p_enDocType=User+guidep_FileName=35010512K0100010.pdfp_DocRef=35010512K01000.
- Schneider Electric. (2020). OS Loader User Manual [Manual de software informático]. Disponible en: https://download.schneider-electric.com/files?p_enDocType=User+guidep_FileName=35006156K0100017.pdfp_DocRef=35006156K01000.
- Siemens. (2001). Micromaster 440 Instrucciones de usol [Manual de software informático]. Disponible en: <https://docs.rs-online.com/3ed6/0900766b80f658ef.pdf>.
- Siemens. (2019). Posicionadores electroneumáticos SIPART PS2 con 4 a 20 mA/HART [Manual de software informático]. Disponible en: <https://cache.industry.siemens.com>.
- Siemens. (2022). Simatic IOT 2040 [Manual de software informático]. Disponible en: <https://mall.industry.siemens.com/mall/>.
- Ubidots. (2022). Ubidots. Disponible en: <https://ubidots.com/>.
- Universidad Politécnica Salesiana. (2013). Cuenca: Se inauguraron laboratorios de control y automatización industrial en la UPS. Disponible en: <https://www.ups.edu.ec/noticias?articleId=112450>.

ANEXOS

ANEXO A

Matriz de consistencia

Tabla 12

Matriz de consistencia lógica.

Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca.				
PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLES	MARCO TEORICO
<p>¿Es idóneo elaborar las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium?</p>	<p>Elaborar las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium.</p>	<p>Se elaborará las guías de aprendizaje y el manual de prácticas de la planta A del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca, empleando el sistema conectado al PLC Schneider Modicon Premium.</p>		<p>Laboratorio LACTI.</p>
ESPECÍFICOS	ESPECÍFICOS	ESPECÍFICAS		
<p>¿Es posible identificar las variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje?</p>	<p>Identificar las variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje.</p>	<p>Se identificará las variables, procesos, subprocesos, instrumentos y protocolos de comunicación que intervienen en la planta A, para la elaboración del manual de prácticas y guías de aprendizaje.</p>	<p>VI: Variables, procesos, subprocesos, instrumentos y protocolos de comunicación.</p>	<p>Variables, procesos, subprocesos, instrumentos y protocolos de comunicación.</p>
<p>¿Es idóneo realizar control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómata programable Schneider; para las guías de aprendizaje y el manual de prácticas?</p>	<p>Realizar control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómata programable Schneider; para las guías de aprendizaje y el manual de prácticas.</p>	<p>Se realizará control y monitoreo remoto de los equipos de la planta A por medio de la red, estableciendo una conexión con el autómata programable Schneider; para las guías de aprendizaje y el manual de prácticas.</p>	<p>VD: Manual de prácticas y guías de aprendizaje.</p>	<p>PLC Schneider.</p>
<p>¿Se podrá diseñar e implementar en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje?</p>	<p>Diseñar e implementar en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje.</p>	<p>Se diseñará e implementará en el PLC Schneider un algoritmo de control aplicado a la variable de nivel de la planta A, para el manual de prácticas y guías de aprendizaje.</p>		<p>Algoritmo de control (PID).</p>
<p>¿Es factible generar el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI?</p>	<p>Generar el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI.</p>	<p>Se generará el manual de prácticas y las guías de aprendizaje de la planta A, empleando el sistema conectado al PLC Schneider del laboratorio de LACTI.</p>		

ANEXO B

Guías y manuales de prácticas del laboratorio

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Configuración del PLC Schneider con el uso del programa Unity Pro XL

Nro Práctica: 1

OBJETIVOS:

1. Configurar Virtual Box para que pueda reconocer el cable TSXCUSB485, necesario para la conexión entre el Unity Pro y PLC Schneider.
2. Configurar los módulos del PLC dentro del Unity Pro XL.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

- a) Conocimientos del manejo de PLC's.
- b) Conocimientos del manejo del software Unity Pro XL.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
PLC Schneider	-	Schneider Electric	-
Virtual Box con Windows 7	1	-	-
Virtual Box Extension Pack	1	-	-
Cable TSXCUSB485	1	Schneider Electric	-
Unity Pro XL	-	Schneider Electric	-

Tabla 1. Equipos, instrumentos y software de la guía.

3. Exposición

PLC Schneider

Su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente.

Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento, como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos es para aplicaciones específicas de seguridad, contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos [1].

Unity Pro XL

Es un software que sirve para programar el PLC y los diferentes módulos comprados por el cliente. Este programa cuenta con herramientas de simulación, diferentes lenguajes de programación, librerías de bloques para diferentes funciones como conexión por Modbus [2].

Máquina Virtual

Es un software que permite simular a un ordenador dentro +de tu propio ordenador, es decir, se puede probar distintos sistemas operativos dentro de tu ordenador sin cambiar al que tienes [3].

Entre los sistemas operativos que pueden soportar la Virtual Box son: Linux, Windows, Mac Os X y Solaris [3].

4. Proceso Configuración de Virtual Box para el uso del Unity Pro XL

4.1. El programa de máquina virtual que se utilizó en la práctica fue el Oracle VM Virtualbox, y para que este reconozca a dispositivos usb se debe descargar el Virtualbox Extension Pack como se observa en la figura 1:

Note: After upgrading VirtualBox it is recommended to upgrade the guest addi

VirtualBox 6.1.34 Oracle VM VirtualBox Extension Pack

- [All supported platforms](#)

Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP, disk encryption, NV1
The Extension Pack binaries are released under the [VirtualBox Personal Use an](#)

VirtualBox 6.1.34 Software Developer Kit (SDK)

- [All platforms](#)

User Manual

Fig 1. Extensión Pack.

4.2. Dirigirse a **Herramientas y Preferencias**, y en la pestaña de Extensiones y hacer clic en agregar extensión como se observa en la figura 2:

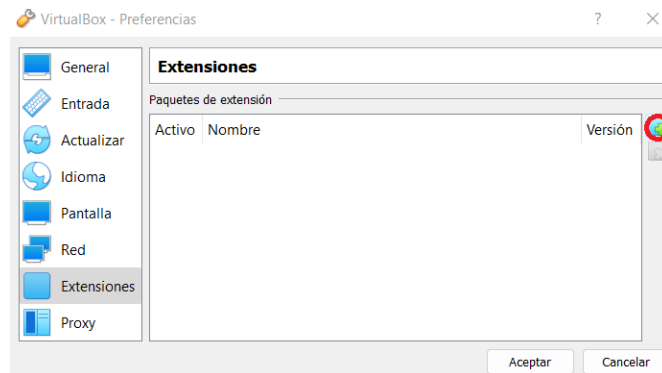


Fig 2. Pestaña de Extensiones.

4.3. Seleccionar el archivo de la Extensión Pack para instalar, al final debe aparecer como en la figura 3:

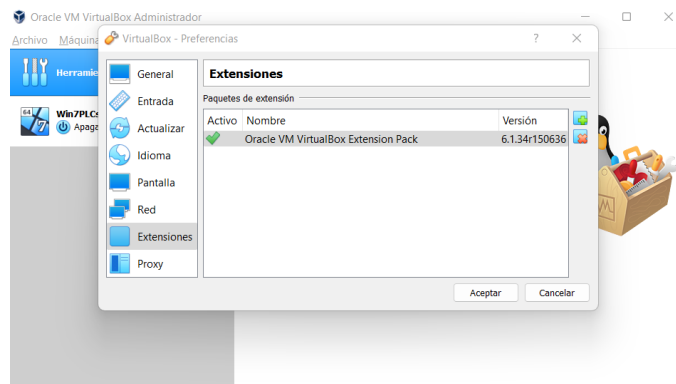


Fig 3. Extensión Pack instalado.

4.4. Descargar e instalar el driver para el cable TSXCUSB485, del siguiente link:

<https://schneider-electric.app.box.com/s/2g6x5er43rxdalczqc3uqhvt52rueku7>

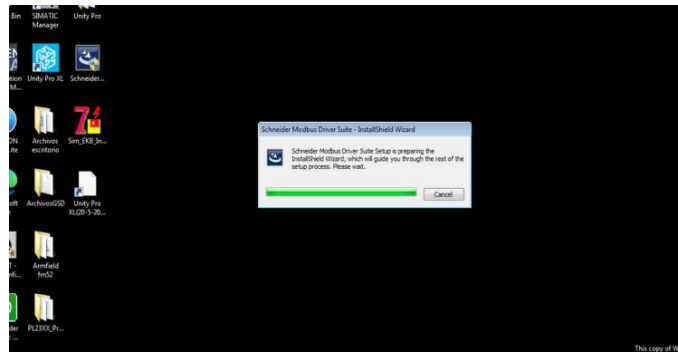


Fig 4. Instalación del Driver usb-PLC.

4.5. Abrir el Unity Pro XL y seleccionar New Project y escogemos la marca de nuestro PLC, que es TSX P57 3634M.

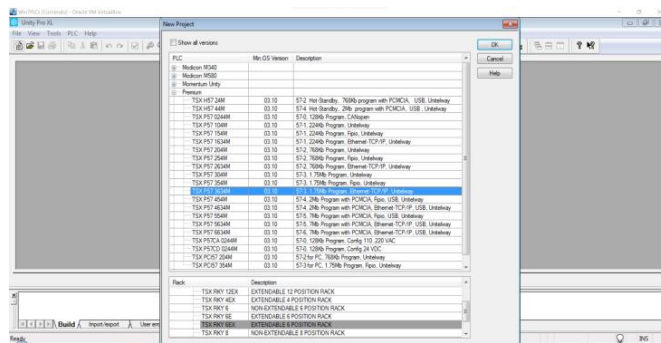


Fig 5. Ventana de elección del modelo de PLC.

4.6. Crear la conexión del Puerto de Ethernet, y para eso se dirige a Communication/Networks and New Network.

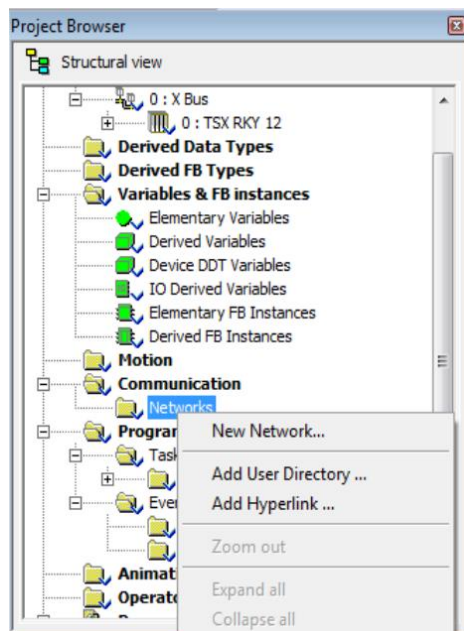


Fig 6. Crear nueva red.

4.7. Configurar de la siguiente forma:

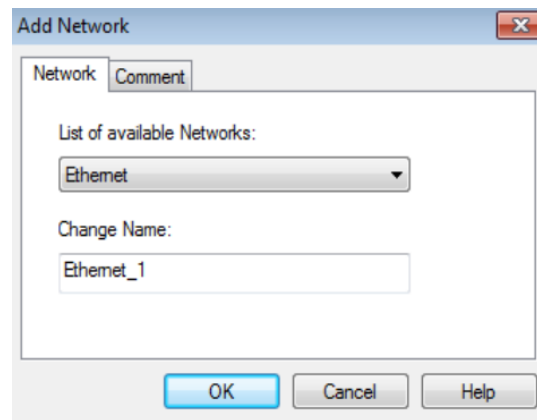


Fig 7. Configurar tipo de red 1.

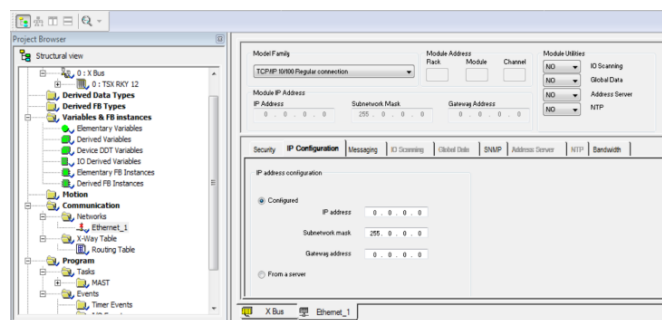


Fig 8. Configurar tipo de red 2.

4.8. Configurar la IP address como 192.168.65.94, subnetwork mask como 255.255.255.0 y hacer clic en validar.

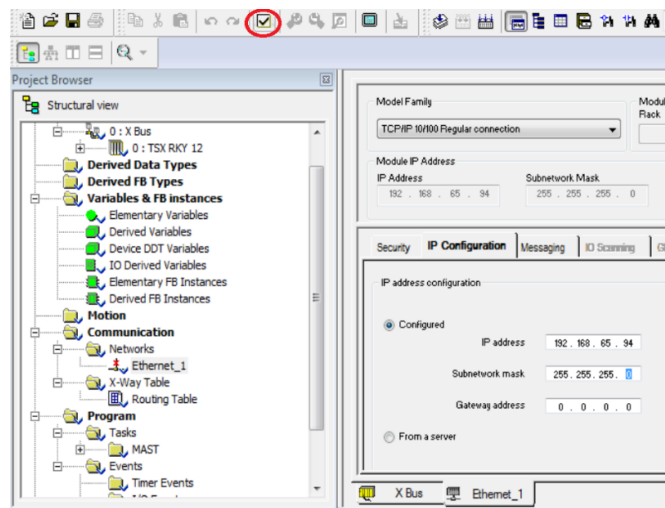


Fig 9. Validación de configuración de red.

4.9. Dirigirse a Project Browser Configuration y hacer clic en XBus, donde aparecerá la siguiente pantalla.

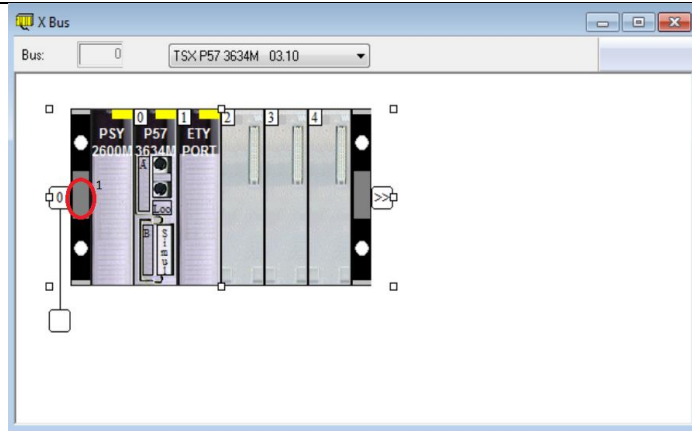


Fig 10. Bus para configurar los módulos del PLC.

4.10. Hacer click en el módulo de los racks (véase la figura 10 en el círculo rojo) y escoger **TSX RYK 12** para aumentar el número de racks del sistema. Y configurar en cada uno de los racks, los siguientes módulos del PLC, los cuales son:

- TSX DEY 32D2K
- TSX DSY 32T2K
- TSX AEY 1600
- TSX ASY 800

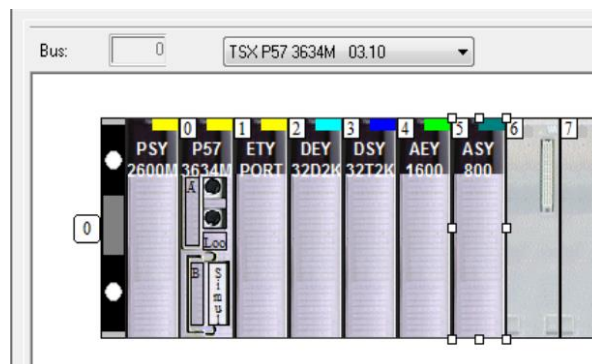


Fig 11. Módulos configurados en Unity Pro XL.

4.11. En el módulo ETY PORT, hacer click y configurar la salida de Ethernet del PLC, como se observa en la figura 12 y 13.

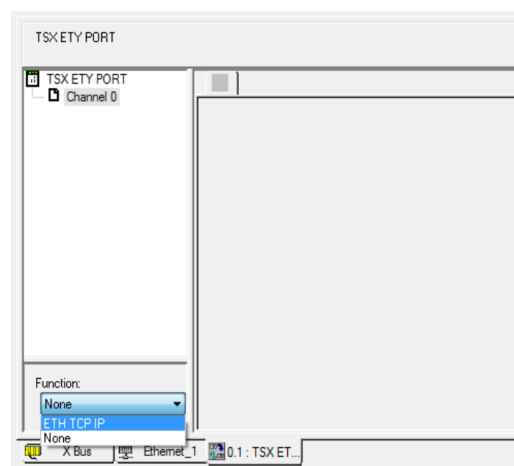


Fig 12. Configuración de puerto Ethernet del PLC Schneider.

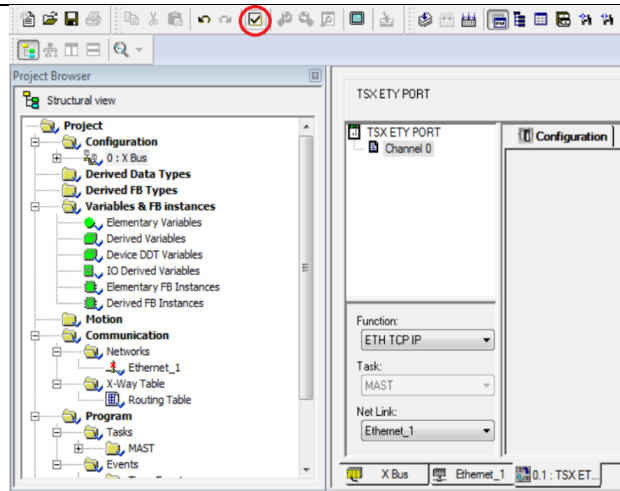


Fig 13. Validación de configuración de puerto Ethernet.

4.12. Conectar el Unity Pro con el PLC y transferir la configuración de los módulos al PLC.

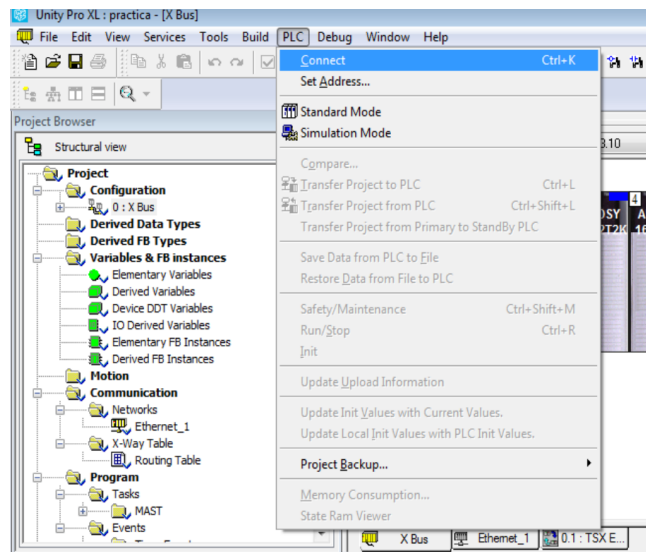


Fig 14. Conectar al PLC Schneider.

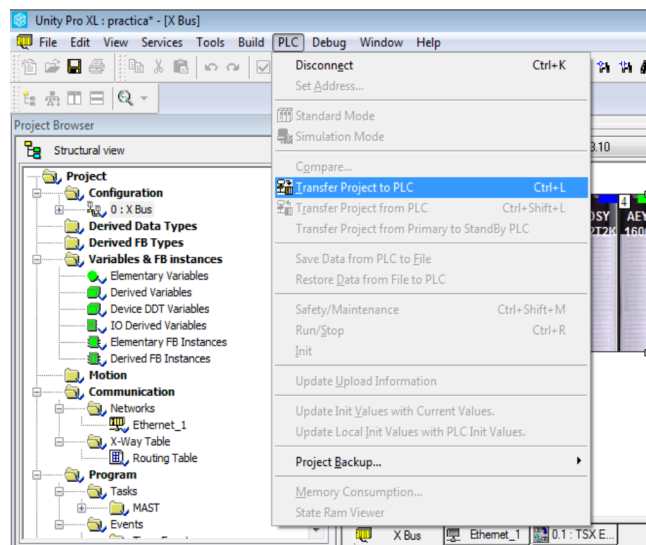


Fig 15. Transferir programación al PLC Schneider.

4.13. Verificar si existe algún error en los módulos del PLC Schneider, observe la figura 16:



Fig 16. Estado de módulos cargado el programa.

Si se requiere una mayor información y detalles de ciertos procesos, se puede revisar el proyecto de titulación adjunto. “Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca”

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Configurar la VirtualBox para reconocer los dispositivos USB.
2. Instalar el driver del cable TSXCUSB485.
3. Programar los módulos dentro del Unity Pro.
4. Conectar el Unity Pro con el PLC Schneider.

RESULTADO(S) OBTENIDO(S):

Al término de la presente práctica el estudiante debe ser capaz de configurar el PLC Schneider Modicon Premium de la planta A, del laboratorio de LACTI.

CONCLUSIONES:

Esta práctica permite que los estudiantes puedan familiarizar los conocimientos impartidos en clase, con un entorno laboral.

RECOMENDACIONES:

Verificar que los estudiantes cumplan con las normas de seguridad.
Efectuar las debidas indicaciones realizadas.

REFERENCIAS:

[1] S. Electric, Plataforma de Automatización Modicon Premium, Telemecanique, 2005.
 [2] S. Electric, Lenguajes y estructura del programa, 2009.
 [3] R. Perez, «¿Qué es una máquina virtual?» 2009. [En línea]. Disponible en: <https://grch.com.ar/docs/so/apuntes/VirtualBox.pdf>. [Último acceso: 23 Junio 2022].

Docente / Técnico Docente:

Firma:



FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Configuración del PLC Schneider con el uso del programa Unity Pro XL

Nro Práctica: 1

OBJETIVO ALCANZADOS:

1. Configurar Virtual Box para que pueda reconocer el cable TSXCUSB485, necesario para la conexión entre el Unity Pro y PLC Schneider.
2. Configurar los módulos del PLC dentro del Unity Pro XL.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Configurar la VirtualBox para reconocer los dispositivos USB.
2. Instalar el driver del cable TSXCUSB485.
3. Programar los módulos dentro del Unity Pro.
4. Conectar el Unity Pro con el PLC Schneider.

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

Nombre del estudiante: _____

Firma del estudiante: _____

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Manejo de la planta mediante el PLC Schneider

Nro Práctica: 2

OBJETIVOS

1. Uso del Unity Pro XL para el manejo de la planta A.
2. Manejo de válvula senoidal usando el Unity Pro XL.
3. Configuración de la válvula Jordán con el Unity Pro XL.
4. Recepción de datos del sensor de nivel.
5. Creación de un panel de control para la manipulación de la planta A.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

- a) Configuración del PLC Schneider.
- b) Conocimientos del manejo de software Labview.
- c) Conocimientos del manejo de software Unity Pro XL.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
PLC Schneider	-	Schneider Electric	-
Virtual Box con Windows 7	1	-	-
Virtual Box Extension Pack	1	-	-
Cable TSXCUSB485	1	Schneider Electric	-
Unity Pro XL	-	Schneider Electric	-
LabView	-	NI Software	-
NI OPC Servers	-	NI Software	-
Librería NI LabVIEW Datalogging and supervisory control module	-	NI Software	-

Tabla 1. Equipos, instrumentos y software de la guía.

3. Exposición

PLC Schneider

Su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente.

Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento, como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos es para aplicaciones específicas de seguridad, contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos [1].

Unity Pro XL

Es un software que sirve para programar el PLC y los diferentes módulos comprados por el cliente. Este programa cuenta con herramientas de simulación, diferentes lenguajes de programación, librerías de bloques para diferentes funciones como conexión por Modbus [2].

Máquina Virtual

Es un software que permite simular a un ordenador dentro +de tu propio ordenador, es decir, se puede probar distintos sistemas operativos dentro de tu ordenador sin cambiar al que tienes [3].

Entre los sistemas operativos que pueden soportar la Virtual Box son: Linux, Windows, Mac Os X y Solaris [3].

4. Manejo del programa Unity Pro XL

En primer lugar, el PLC Schneider está conectado con los protocolos 4-20 mA y HART, por lo tanto, solo los elementos con dichos protocolos podrán ser manejados desde dicho PLC, por eso, se explicará cuáles son las direcciones y como se va a programar en el Unity Pro.

4.1. Dirigirse a **Program/Task/MAST/Sections** y realizar click derecho y seleccionar **New Section**.

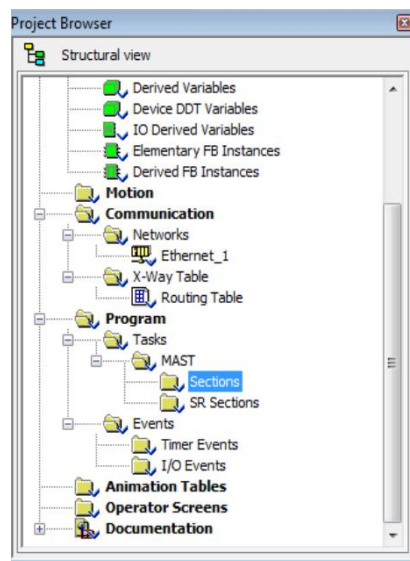


Fig 1. Crear espacio de programación.

4.2. Elegir LD (ladders) en Language y escribir cualquier nombre:

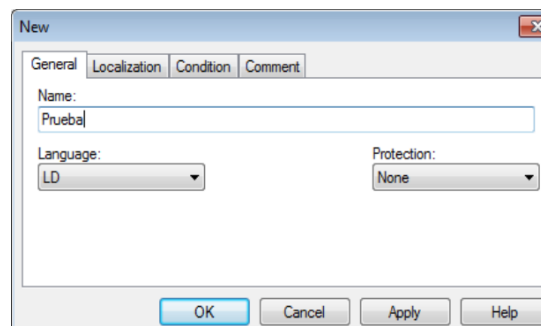


Fig 2. Configuración del tipo de lenguaje de programación.

4.3. Debe aparecer una pantalla como la vista a continuación:

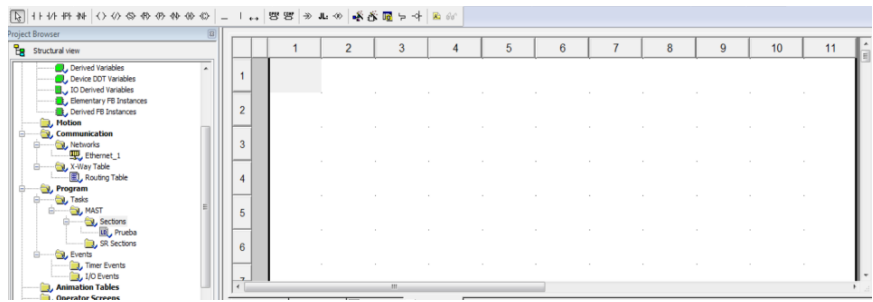


Fig 3. Pantalla para programación en ladder.

Y una lista de botones que sirve para la programación como, por ejemplo:

	Pulsantes
	Bobinas

Tabla 2. Componentes en el software Unity Pro XL.

- Encender/Apagar válvula

4.4. Arrastrar un pulsante al espacio en blanco

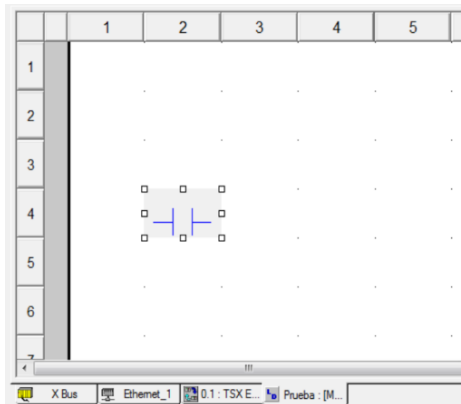


Fig 4. Pulsante.

4.5. Hacer click en el botón y crear una variable como se muestra a continuación:

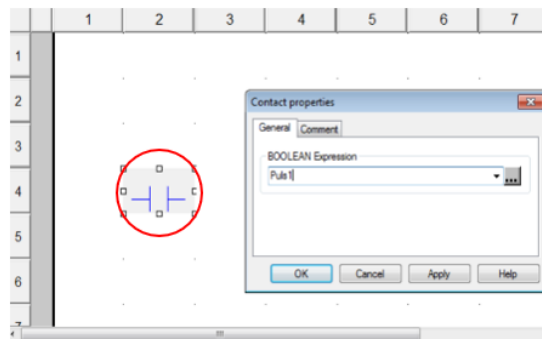


Fig 5. Crear variable booleana.

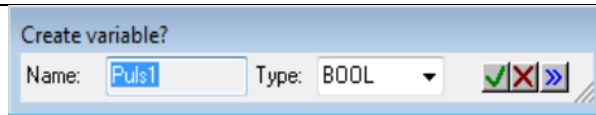


Fig 6. Nombrar variable creada.

4.6. Crear un sistema simple para prender y apagar una válvula, como se muestra a continuación:

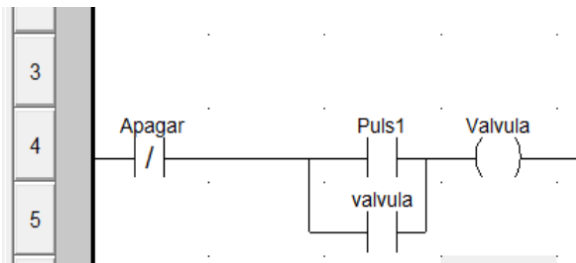


Fig 7. Circuito de encendido y apagado de una válvula.

4.7. Dirigirse a **Elementary Variables**, escribir la siguiente dirección en la variable de la válvula **%Q0.3.8** y cambiar el tipo de variable a **EBOOL** como se ve en la siguiente figura:

Name	Type	Address
Puls1	BOOL	
Apagar	BOOL	
Valvula	EBOOL	%Q0.3.8

Fig 8. Dirección y tipo de variable de la válvula.

4.8. Conectarse y transferir la programación al PLC, como se muestra a continuación:

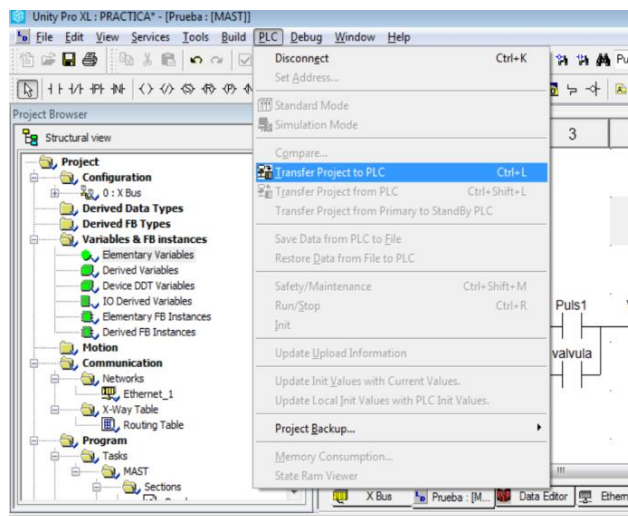


Fig 9. Transferir programación al PLC.

4.9. Activar la casilla de **PLC Run after Transfer** y hacer clic en **Transfer**:

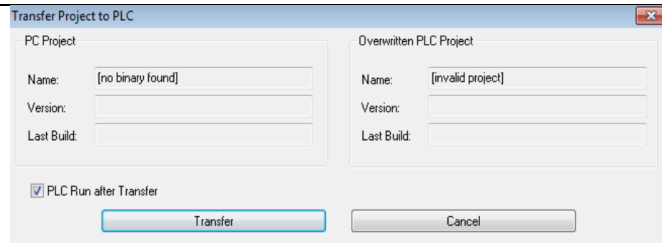


Fig 10. Transferir programación al PLC.

4.10. Crear una tabla dinámica para modificar los valores de las variables. Para eso hacer clic derecho en cualquier parte de la pantalla y clic en **Initialize Animation Table**, como se observa a continuación:

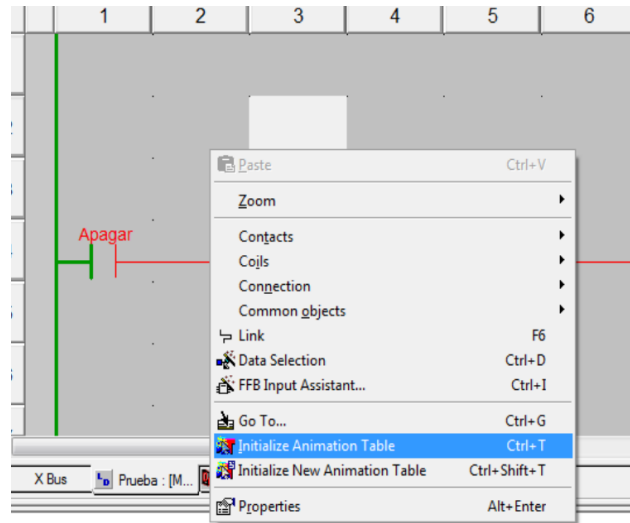


Fig 11. Crear tabla dinámica.

4.11. Debe aparecer una tabla donde se colocan los nombres de las variables que se ha creado, como se muestra a continuación:

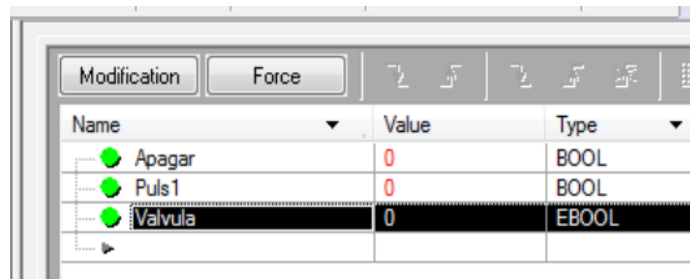


Fig 12. Variables en tabla dinámica.

4.12. Hacer clic en **Modification** y en el botón señalado:

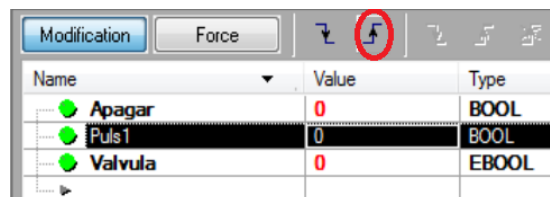


Fig 13. Activar pulsante.

4.13. Se dirige a la ventana de la programación LADDER para comprobar cómo se activa la válvula.

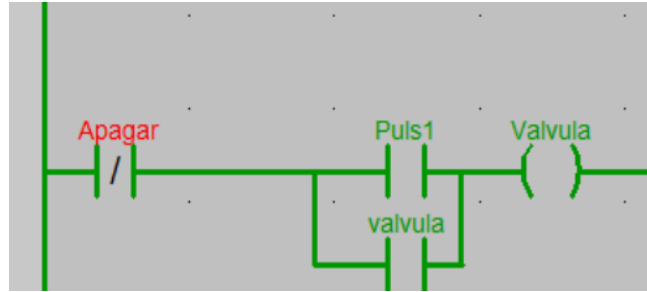


Fig 14. Funcionamiento del programa realizado.

- **Recibir datos del sensor**

4.14. Para recibir datos del sensor son necesarios tres bloques de función: **Move**, **Int_To_Real** y **Scaling**. Estos bloques se encuentran haciendo clic derecho en la ventana de programación y escogemos **FFB Input Assistant**, como se observa en la figura 15:

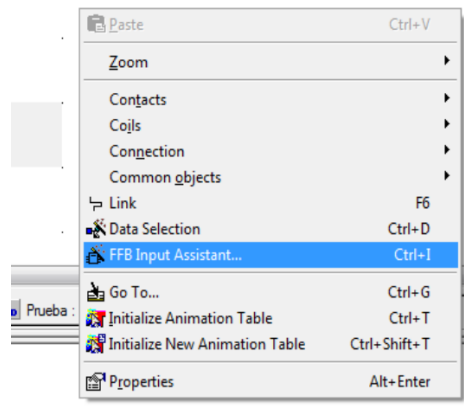


Fig 15. Pasos para crear un bloque de función.

4.15. Debe aparecer una ventana como la vista en la figura 16. Se coloca los nombres de los bloques explicados en el paso 4.14.

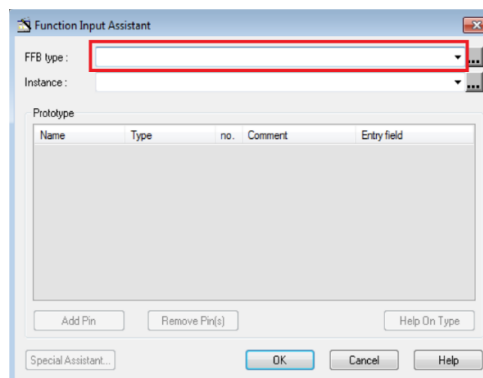


Fig 16. FFB Input Assistant.

4.16. Primero el valor que envía el sensor debe ser guardado en una variable con el bloque **Move**, después transformamos la variable de entero a real. Finalmente, se escala el número para obtener el valor; este paso se observa en la figura 17:

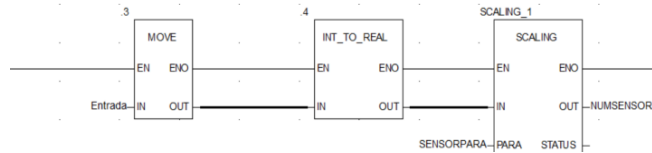


Fig 17. Programación para recibir datos del sensor.

17. Debemos crear 3 variables, una con la dirección de la entrada analógica %IW0.4.1, otra variable que es para los parámetros del bloque de Scaling y la variable donde se guarda el valor del sensor. Los parámetros del bloque Scaling son los observados en la figura 18:

	SENSOR...	Para_SCA...		
	in_min	REAL		0.0
	in_max	REAL		10000.0
	out_min	REAL		0.0
	out_m...	REAL		104.0
	clip	BOOL		

Fig 18. Parámetros del bloque Scaling.

- **Manejar el micromaster 440**

4.18. Para manejar el micromaster es necesario activar una entrada digital, para eso se crea una variable EBOOL con la dirección %Q0.3.18.

4.19. También es necesario crear una variable real para enviar la frecuencia al micromaster, para eso se utiliza la dirección %Q0.5.6. Además, es prescindible utilizar un bloque Move para enviar el dato, por lo que se realiza una programación igual a la figura 19:

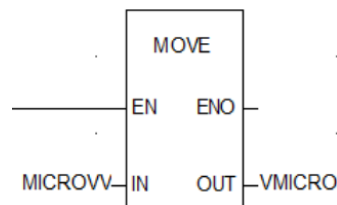


Fig 19. Programación para enviar frecuencia al Micromaster.

4.20. Para poner en funcionamiento la bomba de agua se utiliza la entrada digital con la dirección %Q0.3.20.

5. Manipulación de la planta A

Se propone realizar una interfaz Scada para poder manipular de una mejor manera y ser más interactivo con el operador. En este caso se utiliza el software LabView debido a que la universidad mantiene las licencias. Para ello se debe contar con el programa Labview actualizado, y un servidor OPC; se emplea el OPC con el que ya dispone el propio Labview (NI OPC Servers). Tomar en consideración que se cuente con la librería de control de supervisión y registro de datos (Datalogging and supervisory control module).

- **Conexión con el OPC Server**

5.1. El primer paso a llevar a cabo es el conectarse a la red del laboratorio LACTI. Luego nos dirigimos a nuestro OPC y generamos un nuevo proyecto.

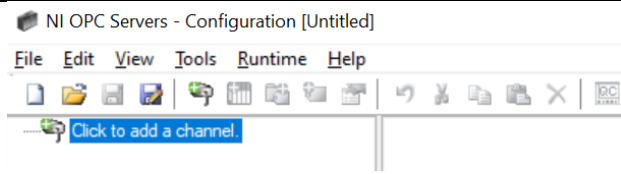


Fig 20. Añadir canal.

5.2. Añadimos un nuevo canal (figura 20); y se establece el nombre para el mismo (figura 21):

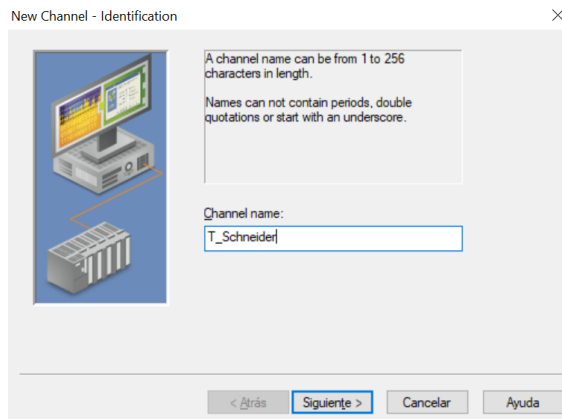


Fig 21. Establecer nombre al canal.

5.3. Damos click en siguiente, donde se debe establecer el protocolo de comunicación a utilizar para la conexión con el controlador. **Modbus TCP/IP Ethernet** (figura 22).

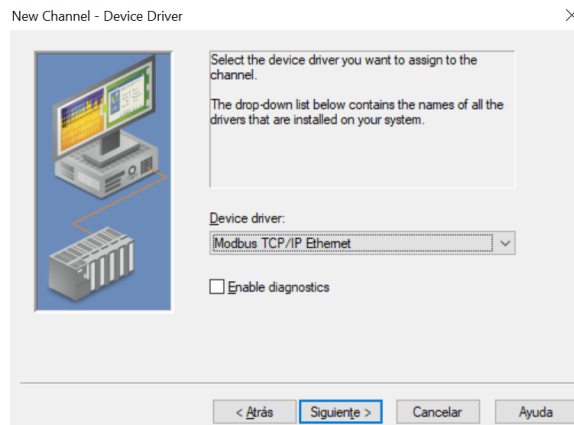


Fig 22. Selección del driver.

5.4. En la siguiente opción se selecciona el adaptador de red a utilizar (figura 23).

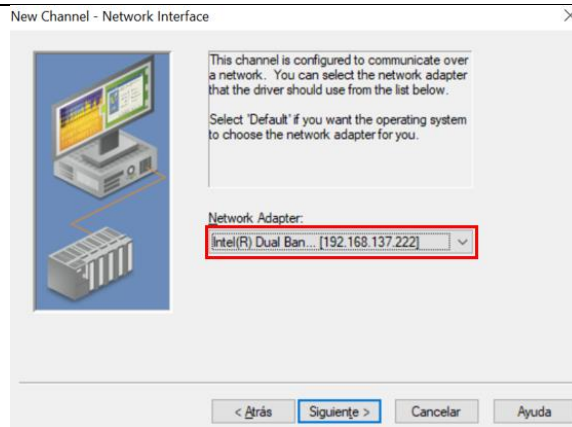


Fig 23. Selección adaptador de red.

5.5. Tomar en consideración que la dirección IP del controlador debe de estar en el mismo dominio que el del adaptador de red. Para corroborar que la selección fue la adecuada nos dirigimos a **Centro de redes y recursos compartidos**, y accedemos a **Cambiar configuración del adaptador**.

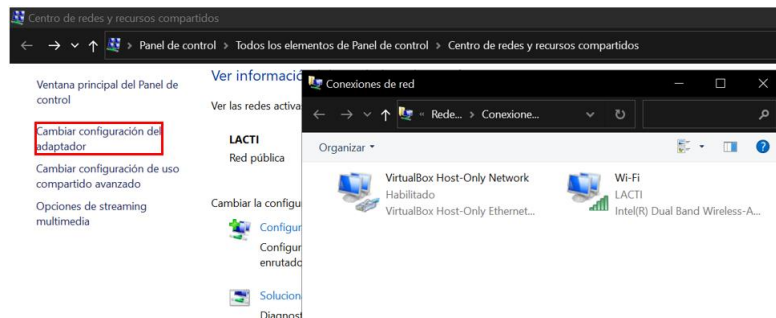


Fig 24. Verificación de adaptador de red.

5.6. Dentro de la pestaña damos doble click en la red conectada, en este caso la red del laboratorio LACTI. Nos dirigimos a detalles y confirmamos la dirección IP mostrada en la sección del adaptador de red.

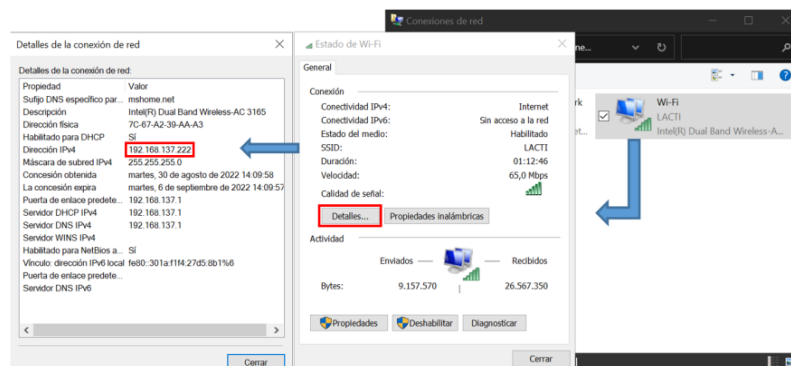


Fig 25. Verificación de adaptador de red.

5.7. Reanudando con la configuración del canal, damos click en todas las pestañas siguientes y finalizamos (figura 26).

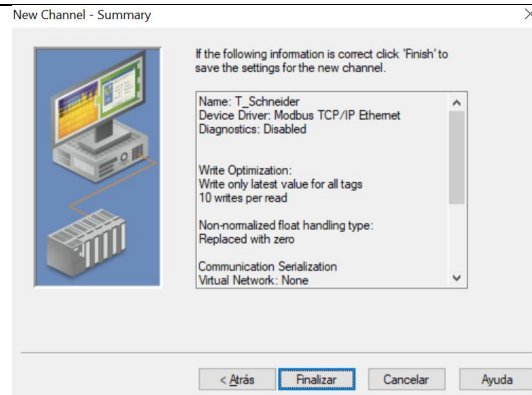


Fig 26.Final de la configuración del canal.

5.8. El siguiente paso será añadir un nuevo dispositivo. Damos click en el menú desplegable.



Fig 27.Añadir dispositivo.

5.9. Definimos el nombre del dispositivo (figura 28):

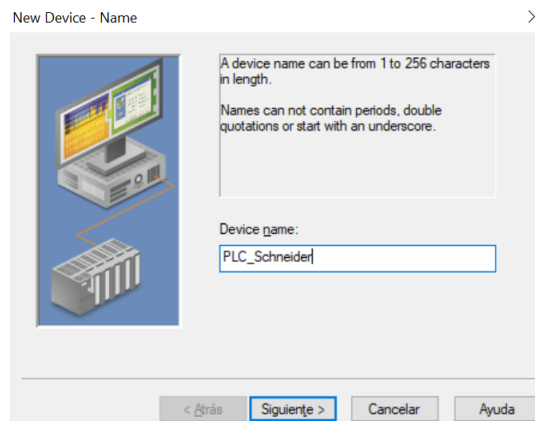


Fig 28.Nombre del dispositivo.

5.10. Pulsamos siguiente. Se presenta la pestaña para seleccionar el modelo que es Modbus.

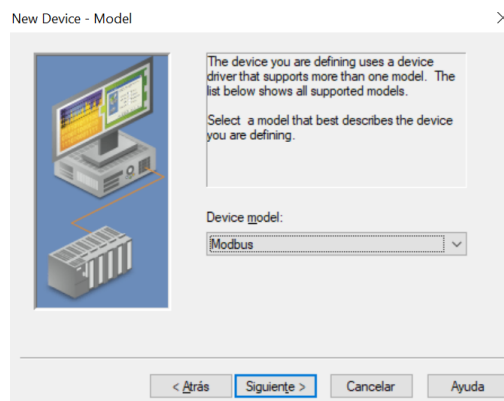


Fig 29. Selección del modelo.

5.11. Seguido se requiere ingresar la dirección IP del PLC Schneider, 192.168.65.94:

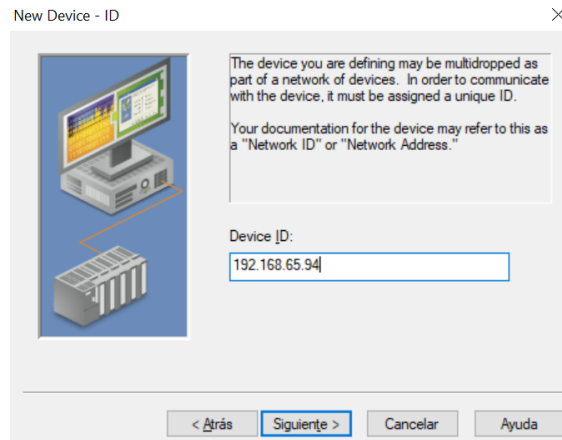


Fig 30. Dirección IP del controlador.

5.12. La configuración de las siguientes pestañas se deja por defecto y se finaliza con la inclusión del dispositivo (figura 31).

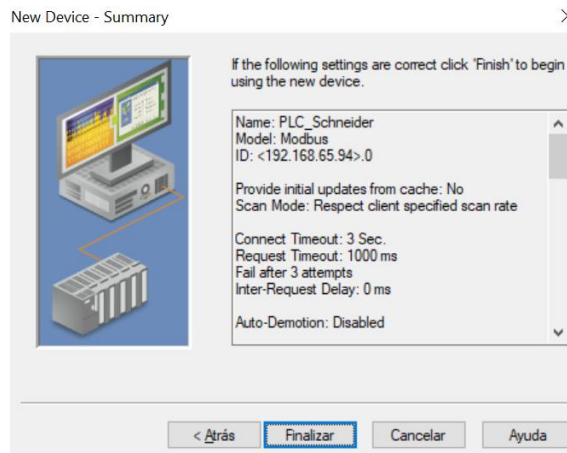


Fig 31. Final de la configuración del dispositivo.

5.13. Después de haber configurado el dispositivo y el canal, se tiene que incluir como una marca cada uno de los elementos que se quieren controlar. Para ello damos click en añadir una nueva marca:

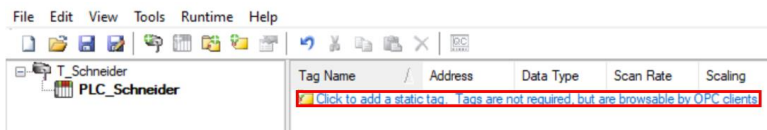


Fig 32. Añadir marcas.

5.14. En esta sección se debe ingresar el nombre de la variable como también la dirección. Se establece el tipo de dato y el acceso que tendrá. Finalmente se crea la marca. La figura 33, muestra la marca con los parámetros para el manejo de la válvula proporcional.

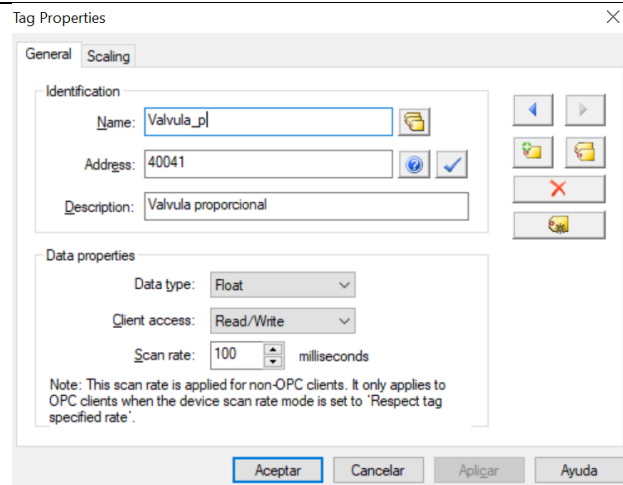


Fig 33. Marca válvula proporcional.

Cabe destacar que la dirección es precedida por 40000 debido a que es un dato Float, además, se tiene que sumar 1 a la dirección original. Es decir, si la variable auxiliar dentro de la programación mantiene la dirección %MW40, en el OPC será 41. Esta es la razón por la cual la dirección del control de la apertura y cierre de la válvula proporcional toma un valor de 40041. Por otra parte para los datos tipo Boolean, se descarta los 40000, y solo se considera el sumar 1 a la dirección original.

5.15. De tal manera, según la programación realizada en Unity pro XL, se debe de incluir las variables que se establecieron para el control y monitoreo de los elementos de la planta.

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
Bomba_Hart	00005	Boolean	100	None	Enciende y apaga la bomba
SENSOR	40023	Float	100	None	Tanque atmosférico
V_EA101	00110	Boolean	100	None	Valvula
V_Neumatica	00111	Boolean	100	None	Tanque atmosférico
V_S101	00103	Boolean	100	None	Valvula
Valvula_P	40041	Float	100	None	Valvula proporcional

Fig 34. Ejemplo de marcas.

En la figura 34, se presenta un ejemplo de los criterios anteriores. A partir de la programación del PLC Schneider se declara en el OPC las direcciones de las variables auxiliares utilizadas para el encendido y apagado de la bomba, la retroalimentación del tanque atmosférico por parte del sensor de nivel, la apertura y cierre de las válvulas solenoides que dirigen el líquido, el control de la válvula proporcional, y el manejo de la válvula neumática On/Off.

5.16. Para evidenciar el correcto funcionamiento se puede crear un cliente rápido, pulsando el botón mostrado en la figura 35:



Fig 35. Cliente rápido.

5.17. Dentro del menú desplegable ingresamos en el apartado **T_Schneider.PLC_Schneider**, y corroboramos que las marcas presentan un buen estado (Good) (figura 36).

Item ID	Data Type	Timestamp	Quality	Update Count
T_Schneider.PLC_Schneider B...	Boolean	17:23:14.971	Good	1
T_Schneider.PLC_Schneider S...	Float	17:23:15.033	Good	1
T_Schneider.PLC_Schneider.V...	Boolean	17:23:15.096	Good	1
T_Schneider.PLC_Schneider.V...	Boolean	17:23:15.096	Good	1
T_Schneider.PLC_Schneider.V...	Boolean	17:23:15.096	Good	1
T_Schneider.PLC_Schneider.V...	Float	17:23:15.159	Good	1

Fig 36. Cliente rápido, buen estado.

5.18. Se puede modificar el valor de las variables dando click derecho en la marca deseada, y seleccionando **Synchronous Write** (figura 37).

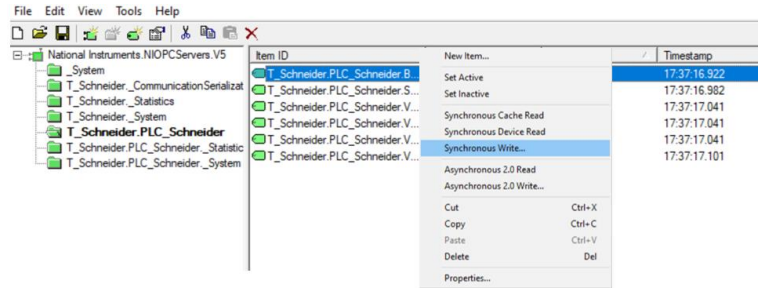


Fig 37. Modificar valor de las marcas.

- **Panel de control**

5.19. Para generar el panel de control, abrimos LabView y creamos un nuevo proyecto. Nos dirigimos a **I/O Server** para declarar el OPC.

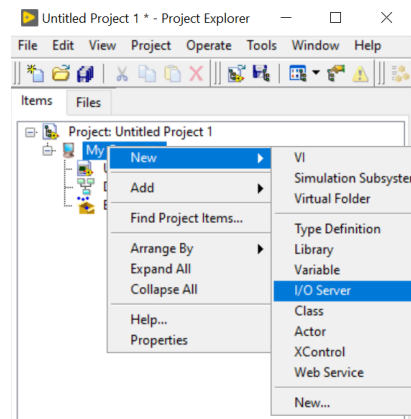


Fig 38. I/O Server.

5.20. Seleccionamos **OPC Client** y continuamos.

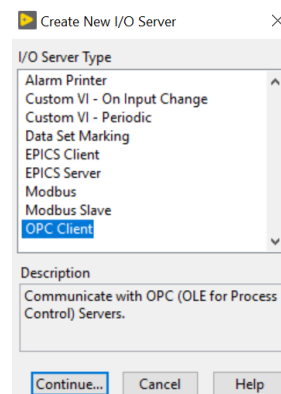


Fig 39. Selección OPC Client

5.21. En la siguiente pestaña seleccionamos el OPC utilizado, **National Instruments.NIOPCServers.V5**; seguido damos en **Ok** y finalizamos (figura 40).

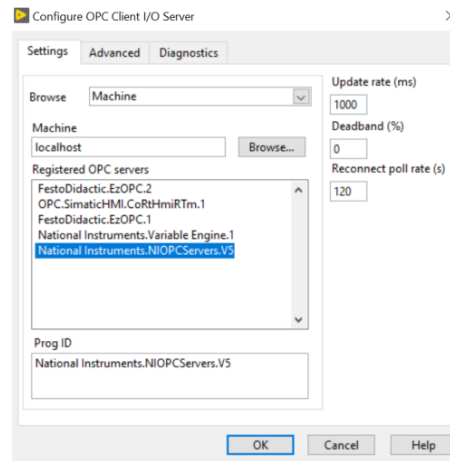


Fig 40. Configuración del cliente.

5.22. El siguiente paso es el generar variables compartidas, para ello nos dirigimos a **Variable** (figura 41).

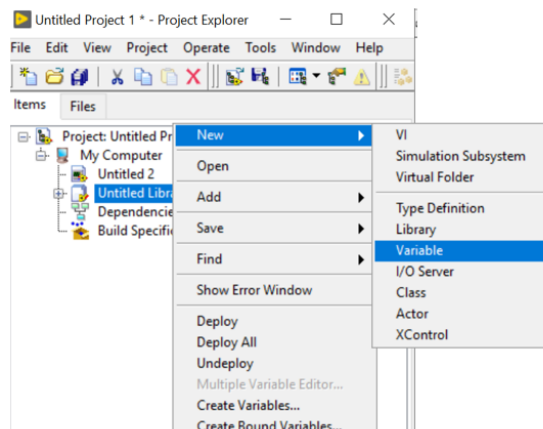


Fig 41. Selección variable.

5.23. Aparecerá una pestaña, donde se debe de activar **Enable Aliasing** y se ingresa a **Browse**.

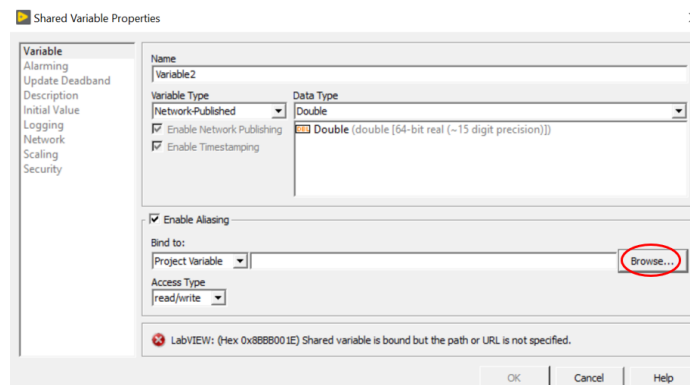


Fig 42. Búsqueda de variables.

5.24. En el menú desplegable buscamos las marcas declaradas en el OPC. Seleccionamos la variable deseada y presionamos **Ok** (figura 43).

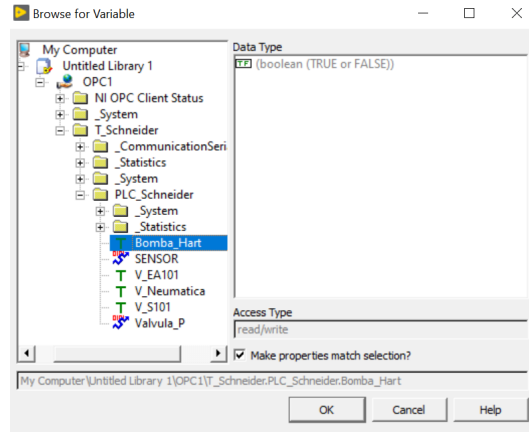


Fig 43. Selección de variables de OPC.

5.25. Se establece un nombre a la variable y se confirma la acción (figura 44).

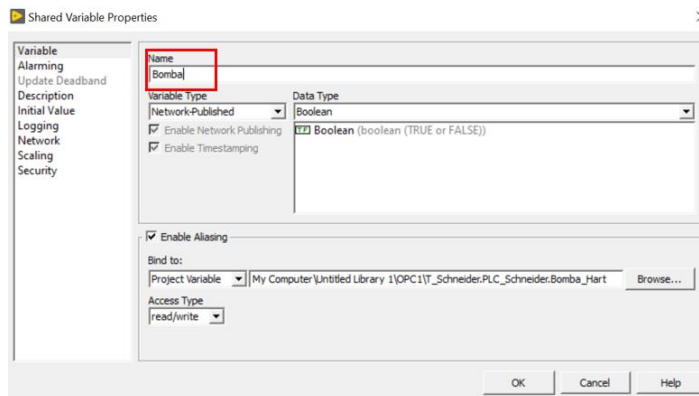


Fig 44. Nombrar variables.

5.26. De tal manera, se incluye la variable compartida al proyecto.

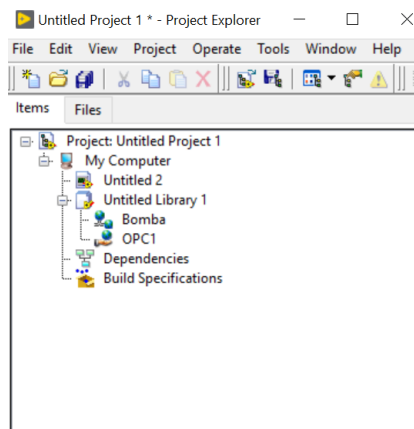


Fig 45. Variables compartidas en el proyecto.

5.27. Según se requiera se pueden añadir más variables al proyecto, y arrastrar a un VI para poder generar el panel de control.

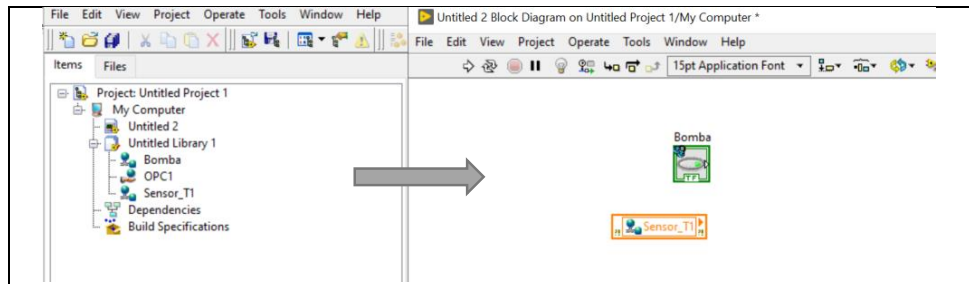


Fig 46. Variables arrastradas en el VI.

Según conceptos con los que ya se cuenta acerca del manejo de LabView, y ligado a la programación que se cargó en el PLC Schneider se genera el sistema Scada, para la interacción hombre – maquina. Se puede visualizar en la figura 47, un ejemplo del panel de control diseñado.

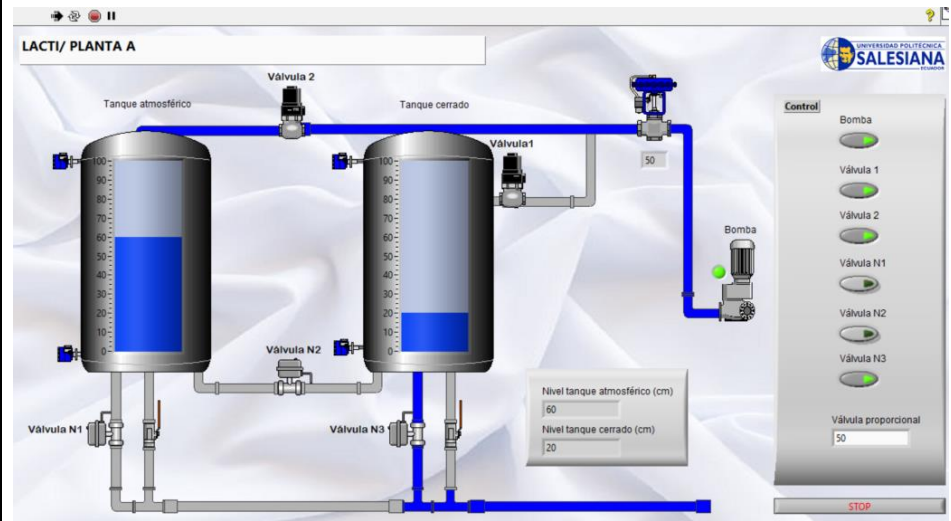


Fig 47. Panel de control

Si se requiere una mayor información y detalles de ciertos procesos, se puede revisar el proyecto de titulación adjunto. “Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca”

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Programar el software Unity Pro XL para manipular las válvulas planta A.
2. Programar el software Unity Pro XL para recibir los datos del sensor del tanque abierto de la planta A.
3. Manejar el micromaster 440 usando el PLC Schneider.
4. Generar un panel de control mediante el software LabView.
5. Registrar los datos del nivel de llenado del tanque atmosférico en un archivo Excel, TDMS o en un bloc de notas; con una apertura de la válvula y una frecuencia de la bomba constante, durante 300 segundos.
6. Medir el nivel del tanque cada 60 segundos hasta llegar a los 300 segundos. Tanto la apertura de la válvula, como también, la frecuencia la bomba es la que usted crea necesario; estos valores se mantienen para todo el proceso. Se puede ayudar de los datos registrados del punto anterior.

Tiempo (s)	Nivel del tanque (m)
60 s	
120 s	
180 s	

	240 s		
	300 s		
7. Graficar la curva dada por el punto anterior a partir del registro realizado (Archivo Excel, TDMS o bloc de notas), se debe de considerar todas las mediciones desde el segundo 1 al segundo 300.			
RESULTADO(S) OBTENIDO(S): Al término de la presente práctica el estudiante debe ser capaz de manipular y monitorear los procesos de llenado de tanques del laboratorio LACTI.			
CONCLUSIONES: Esta práctica permite que los estudiantes puedan familiarizar los conocimientos impartidos en clase, con un entorno laboral.			
RECOMENDACIONES: Verificar que los estudiantes cumplan con las normas de seguridad. Efectuar las debidas indicaciones realizadas.			
REFERENCIAS: [1] S. Electric, Plataforma de Automatización Modicon Premium, Telemecanique, 2005. [2] S. Electric, Lenguajes y estructura del programa, 2009. [3] R. Perez, «¿Qué es una máquina virtual?,» 2009. [En línea]. Disponible en: https://grch.com.ar/docs/so/apuntes/VirtualBox.pdf . [Último acceso: 23 Junio 2022].			

Docente / Técnico Docente:

Firma:



FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Manejo de la planta mediante el PLC Schneider

Nro Práctica: 2

OBJETIVO ALCANZADOS:

1. Uso del Unity Pro XL para el manejo de la planta A.
2. Manejo de válvula senoidal usando el Unity Pro XL.
3. Configuración de la válvula Jordán con el Unity Pro XL.
4. Recepción de datos del sensor de nivel.
5. Creación de un panel de control para la manipulación de la planta A.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Programar el software Unity Pro XL para manipular las válvulas planta A.
2. Programar el software Unity Pro XL para recibir los datos del sensor del tanque abierto de la planta A.
3. Manejar el micromaster 440 usando el PLC Schneider.
4. Generar un panel de control mediante el software LabView.
5. Registrar los datos del nivel de llenado del tanque atmosférico en un archivo Excel, TDMS o en un bloc de notas; con una apertura de la válvula y una frecuencia de la bomba constante, durante 300 segundos.
6. Medir el nivel del tanque cada 60 segundos hasta llegar a los 300 segundos. Tanto la apertura de la válvula, como también, la frecuencia la bomba es la que usted crea necesario; estos valores se mantienen para todo el proceso. Se puede ayudar de los datos registrados del punto anterior.

Tiempo (s)	Nivel del tanque (m)
60 s	
120 s	
180 s	
240 s	
300 s	

7. Graficar la curva dada por el punto anterior a partir del registro realizado (Archivo Excel, TDMS o bloc de notas), se debe de considerar todas las mediciones desde el segundo 1 al segundo 300.

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

Nombre del estudiante: _____

Firma del estudiante: _____

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Conexión Modbus con instrumentos de la planta A

Nro Práctica: 3

Objetivos

1. Uso del Unity Pro XL para el manejo de la planta.
2. Manejo del HMI para la comunicación Modbus entre en el Hardware Control System (HCS) y PLC Schneider.
3. Recepción de datos mediante sensores configurados por protocolo HART.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

- a) Conocimientos del manejo del PLC Schneider.
- b) Revisar y ejecutar la guía de práctica 1.
- c) Revisar y ejecutar la guía de práctica 2.
- d) Conocimiento de redes industriales.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
PLC Schneider	-	Schneider Electric	-
Virtual Box con Windows 7	1	-	-
Virtual Box Extension Pack	1	-	-
Cable TSXCUSB485	1	Schneider Electric	-
Unity Pro XL	-	Schneider Electric	-
HMI GL070E	-	Kinco	-
Concentrador Hart (HCS)	-	HCS	-

Tabla 1. Equipos, instrumentos y software de la guía.

3. Exposición

PLC Schneider

Su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente.

Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento, como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos es para aplicaciones específicas de seguridad, contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos [1].

Unity Pro XL

Es un software que sirve para programar el PLC y los diferentes módulos comprados por el cliente. Este programa cuenta con herramientas de

simulación, diferentes lenguajes de programación, librerías de bloques para diferentes funciones como conexión por Modbus [2].

Máquina Virtual

Es un software que permite simular a un ordenador dentro +de tu propio ordenador, es decir, se puede probar distintos sistemas operativos dentro de tu ordenador sin cambiar al que tienes [3].

Entre los sistemas operativos que pueden soportar la Virtual Box son: Linux, Windows, Mac Os X y Solaris [3].

Hart Concentrator System (HCS)

Es un convertidor de señal digital Hart a señal serial Modbus RS-485, permitiendo manejar válvulas y otros equipos HART mediante Modbus. Cuenta con capacidad de trabajar con 16 equipos simultáneamente [4].

4. Proceso

4.1. Es necesario programar el HMI para que reciba datos del Hart Concentrator System (HCS) y los envíe al PLC, para eso es necesario descargar el programa Kinco DTools.

4.2. Se abre el programa y se hace click en nuevo proyecto, ahí aparece para nombrar al proyecto y se escoge el modelo del HMI que se va a utilizar, en este caso el modelo es GL070E, este proceso se observa en la figura 1:

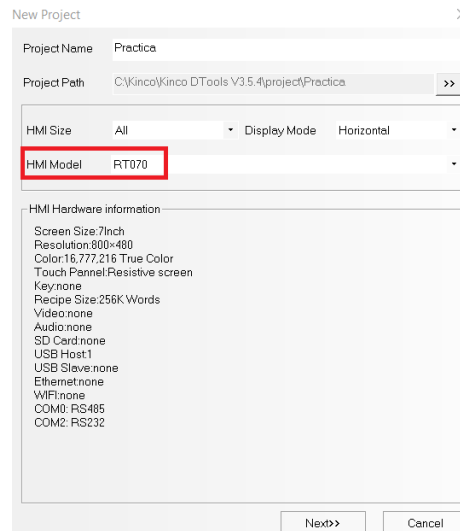


Fig 1. Crear proyecto en Kinco DTool.

4.3. Luego, aparece una pantalla para configurar el puerto Ethernet del HMI, ahí se va a colocar la siguiente dirección IP 192.168.65.100 y se hace click en Finish, como se muestra en la figura 2:

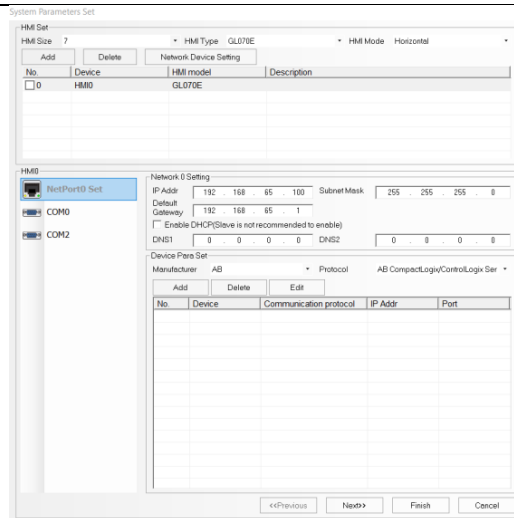


Fig 2. Configurar puerto Ethernet.

4.4. Se dirige a la parte de PLC que se encuentra en la parte izquierda baja de la pantalla, y se arrastran dos PLC, el Modbus TCP Slave y Modbus RTU Extend, este proceso se puede observar en la figura 3:

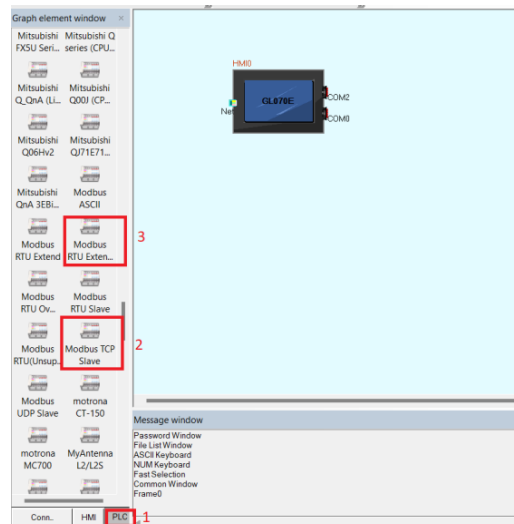


Fig 3. Bloques de Modbus RTU y TCP.

4.5. Se hace doble click en la entrada Ethernet del bloque Modbus TCP Slave, se le da click en Network Device Setting y add, como se observa en la figura 4:

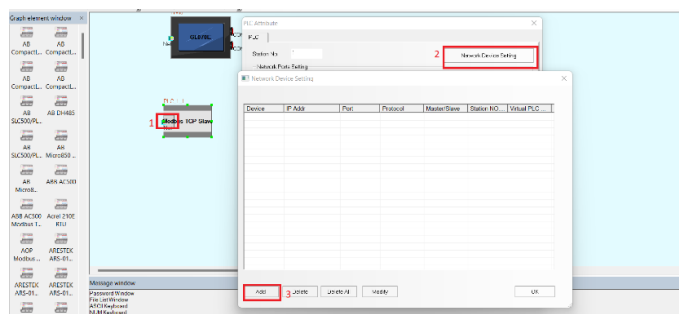


Fig 4. Configuración de la conexión entre HMI y PLC.

4.6. Se declara al HMI como maestro y PLC Schneider como esclavo, como se observa en la figura 5.

Device	IP Addr	Port	Protocol	Master/Slave	Station NO...	Virtual PLC ...
HMI0	192.168.85.100	502	Modbus TCP	M		
PLC_1_1	192.168.85.94	502	Modbus TCP...	S	1	

Fig 5. Lista de equipos conectados.

4.7. Se debe dirigir a conectar y arrastrar serial port hacia uno de los puertos seriales del HMI y se conecta con el bloque de Modbus RTU Extend, como se muestra en la figura 6:

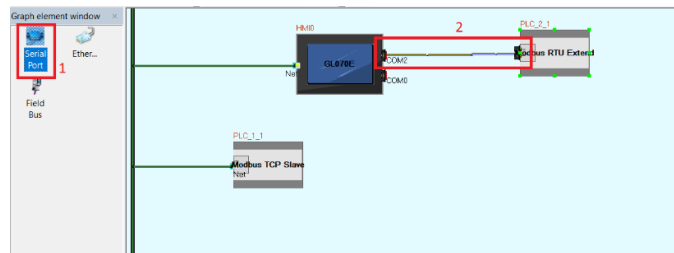


Fig 6. Conexión serial entre HMI y bloque de Modbus RTU.

4.8. Se dirige a la ventana de estructura del proyecto, y en HMI0/Frame0 se hace click para abrir una ventana donde se configura lo que se va a ver en el HMI, como se observa en la figura 7:

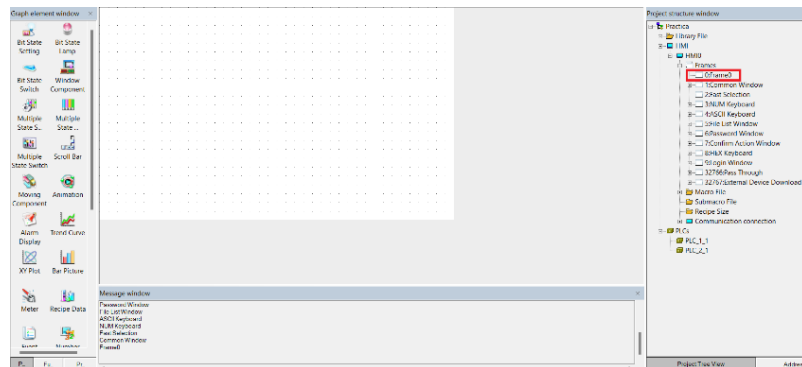


Fig 7. Ventana de HMI.

4.9. Se desplaza el bloque de Number component y se configuran los parámetros vistos en la figura 8:

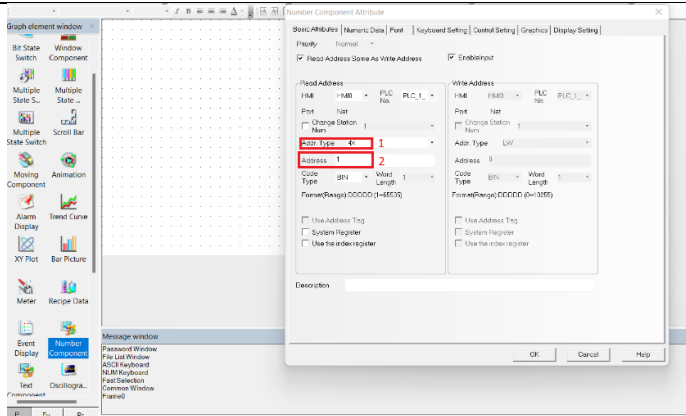


Fig 8. Parámetros de bloque de número.

- Donde:
 1. **Addr. Type:** tipo de dirección que va a enviar, como se va a utilizar Modbus para la comunicación entre el HMI y el PLC, se van a escoger los tipos 4x para el envío de números, y el 0x para envío de variables booleanas.
 2. **Address:** dirección puesta en el PLC Schneider.

4.10. Para que las variables se pasen rápidamente del HCS al PLC, es necesario que se use un macro en el HMI, solo para decir que la variable que llega del HCS, sea la misma que la enviada al PLC, para eso se dirige a HMI0/Macro File/ 0:group0 y se hace click derecho, y se selecciona add macro. Como se observa en la figura 9:

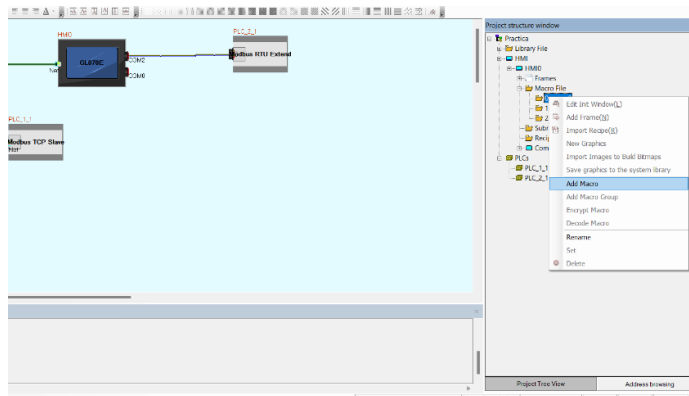


Fig 9. Crear macros.

4.11. Dentro del macro, solo se va a cambiar la línea 26 por esta programación vista en la figura 10:

```

1  #include "macrotypedef.h"
2  #include "math.h"
3
4  /*
5   Read,Write Local address function:
6   int ReadLocal( const char *type, int addr, int nRegs, void *buf, int flag );
7   int WriteLocal( const char *type, int addr, int nRegs, void *buf , int flag );
8
9   Parameter:  type   is the string of "LW","LB" etc;
10              address is the Operation address ;
11              nRegs  is the length of read or write ;
12              buf    is the buffer which store the reading or writing data;
13              flag   is 0,then codetype is BIN,is 1 then codetype is BCD;
14   return value : 1 ,Operation success
15                  0, Operation fail.
16
17   eg: read the value of local lw200 and write it to the lw202,with the codetype BIN,
18   The code is :
19
20   short buff[2] = {0};
21   ReadLocal("LW", 200, 2, (void*)buf, 0);
22   WriteLocal("LW", 202, 2, (void*)buf, 0);
23 */
24 int MacroEntry()
25 {
26   C=P;
27   T1=T;;
28 }

```

Fig 10. Línea de programación de macros.

4.12. Hay que dirigirse a los parámetros de los macros para declarar las direcciones de las variables que se colocan antes, para eso hay que ir al cuadro en blanco de parameters[macro] y hacer click derecho en add. Y se configura al parámetro como se observa en la figura 11:

Macro Code Variable

Variable	P	HMI	HMI0
Data Type	unsigned short	PLC No.	PLC_1_1
Port	Net	<input type="checkbox"/> Change Station Num	1
Address Type	4X	Address	1
Code Type	BIN	<input type="checkbox"/> AddrTag	
Format(Range):DDDDD (1-65535)			
Word Length	1	<input type="checkbox"/> Array variable	
R/W	Read	Array Length	

OK Cancel

Fig 11. Parámetros de variable de macros.

- Donde:

Variable: nombre de la variable.

Data Type: tipo de variable (float, bool), en este caso se coloca unsigned short.

Address Type: tipo de dirección, se coloca 4x.

R/W: si se quiere que la variable sea de lectura, escritura o ambos, en este caso será solo de lectura.

Después de la configuración se tiene las siguientes variables:

Parameters[macro_0.c]							
Data Type	Param name	PLC No.	PLC Add...	Address	Wor...	OptMode	Array
unsigned ...	P	1	4X	1	1	Read	No
unsigned ...	C	0	4X	302	1	Write	No
unsigned ...	T	1	4X	6	1	Read	No
unsigned ...	T1	1	4X	8	1	Write	No

Fig 12. Lista de variables de macros.

4.13. Se agrega un Timer para que se repita lo puesto en la macro, para eso se regresa a la ventana del HMI, se selecciona funtion paths, se busca el bloque

de Timer y lo arrastra a la ventana del HMI, y se cambia su tiempo de duración a 1 y se aceptan los cambios, como se muestra en la figura 13:

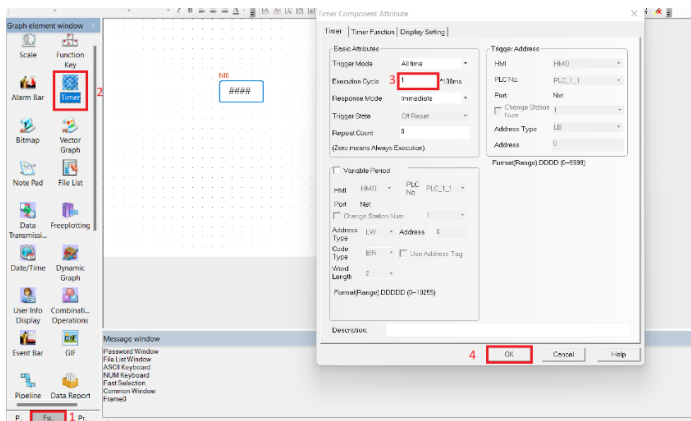


Fig 13. Parámetros de Timer.

4.14. Luego de programar el HMI, se ingresa a Unity Pro, para programar el PLC, aquí lo que hay que hacer es crear variables con la dirección %M para booleanos y %MW para números. Como se observa en la figura 14:

NUMSENSOR	REAL	%MW22	
NUMJORDAN	REAL	%MW20	
VALORSENSOR	REAL	%MW10	
NJORDAN	INT	%MW2	
VariableEjemplo	INT	%MW1	541
VEV101	EBOOL	%M110	
VEA101	EBOOL	%M109	
VEC302	EBOOL	%M108	
VEA103	EBOOL	%M107	
VEA104	EBOOL	%M106	
VEC402	EBOOL	%M105	
VEC305	EBOOL	%M104	

Fig 14. Ejemplos de variables con direcciones %M.

Si se requiere una mayor información y detalles de ciertos procesos, se puede revisar el proyecto de titulación adjunto. “Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca”

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Crear proyecto en KinkoDTools.
2. Definir las conexiones del PLC, HMI y HCS en el KinkoDTools.
3. Configurar los bloques de números en el KinkoDTools.
4. Crear macros.
5. Configurar variables en Unity Pro para recibir los datos del HMI.
6. Crear un entorno grafico para manipular la planta.

RESULTADO(S) OBTENIDO(S):

Al término de la presente práctica el estudiante debe ser capaz de monitorear por medio del PLC Schneider, las variables del concentrador HCS. Además de manipular la planta por medio de la pantalla HMI.

CONCLUSIONES:

Esta práctica permite que los estudiantes puedan familiarizar los conocimientos impartidos en clase, con un entorno laboral.

RECOMENDACIONES:

Verificar que los estudiantes cumplan con las normas de seguridad.
Efectuar las debidas indicaciones realizadas.

REFERENCIAS:

[1] S. Electric, Plataforma de Automatización Modicon Premium, Telemecanique, 2005.

[2] S. Electric, Lenguajes y estructura del programa, 2009.

[3] R. Perez, «¿Qué es una máquina virtual?,» 2009. [En línea]. Disponible en: <https://grch.com.ar/docs/so/apuntes/VirtualBox.pdf>. [Último acceso: 23 Junio 2022].

[4] M. Industries, «HCS Datasheet Moore Industries,» Mayo 2016. [En línea]. Available: https://www.miinet.com/images/pdf/datasheets/HCS_Datasheet_Moore_Industries.pdf. [Último acceso: 18 Julio 2022].

Docente / Técnico Docente:

Firma:



FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Conexión Modbus con instrumentos de la planta A

Nro Práctica: 3

OBJETIVO ALCANZADOS:

1. Uso del Unity Pro XL para el manejo de la planta.
2. Manejo del HMI para la comunicación Modbus entre en el Hardware Control System (HCS) y PLC Schneider.
3. Recepción de datos mediante sensores configurados por protocolo HART.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Crear proyecto en KinkoDTools.
2. Definir las conexiones del PLC, HMI y HCS en el KinkoDTools.
3. Configurar los bloques de números en el KinkoDTools.
4. Crear macros.
5. Configurar variables en Unity Pro para recibir los datos del HMI.
6. Crear un entorno grafico para manipular la planta.

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

Nombre del estudiante: _____

Firma del estudiante: _____

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Control de nivel en la planta A del laboratorio de LACTI

Nro Práctica: 4

OBJETIVOS

1. Diseñar un control proporcional, integral y derivativo (PID) en la planta A del laboratorio de LACTI.
2. Implementar el controlador en el PLC Schneider Modicon Premium.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

- a) Conocimientos del manejo del PLC Schneider.
- b) Conocimientos del manejo de software Labview.
- c) Conocimiento del manejo del software Matlab.
- d) Conocimiento de ingeniería de control.
- e) Revisar y ejecutar la guía de práctica 1.
- f) Revisar y ejecutar la guía de práctica 2.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
PLC Schneider	-	Schneider Electric	-
Virtual Box con Windows 7	1	-	-
Virtual Box Extension Pack	1	-	-
Cable TSXCUSB485	1	Schneider Electric	-
Unity Pro XL	-	Schneider Electric	-
LabView	-	NI Software	-
NI OPC Servers	-	NI Software	-
Librería NI LabVIEW Datalogging and supervisory control module	-	NI Software	-
Matlab	-	Matlab	-
Ident Toolbox	-	Matlab	-

Tabla 1. Equipos, instrumentos y software de la guía.

3. Exposición

PLC Schneider

Su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente.

Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento, como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos es para aplicaciones específicas de seguridad,

contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos [1].

Unity Pro XL

Es un software que sirve para programar el PLC y los diferentes módulos comprados por el cliente. Este programa cuenta con herramientas de simulación, diferentes lenguajes de programación, librerías de bloques para diferentes funciones como conexión por Modbus [2].

Máquina Virtual

Es un software que permite simular a un ordenador dentro +de tu propio ordenador, es decir, se puede probar distintos sistemas operativos dentro de tu ordenador sin cambiar al que tienes [3].

Entre los sistemas operativos que pueden soportar la Virtual Box son: Linux, Windows, Mac Os X y Solaris [3].

4. Planteamiento

Para el algoritmo de control se partirá del tanque abierto; mismo que será manipulado por la línea del protocolo de comunicación HART. La figura 1, muestra la configuración a cumplir dentro del lazo de control; de tal manera, se tendrá como la planta al flujo de entrada de líquido manipulado por la válvula proporcional, el tanque y un flujo de salida constante dado por una válvula manual. Todo el proceso será controlado por el autómatas programable Schneider (PLC) y se fijará un valor de referencia a seguir. Además, para medir la cantidad de agua se utilizará un transmisor de nivel; y se mantendrá una segunda línea de flujo de salida de líquido para introducir perturbaciones al sistema.

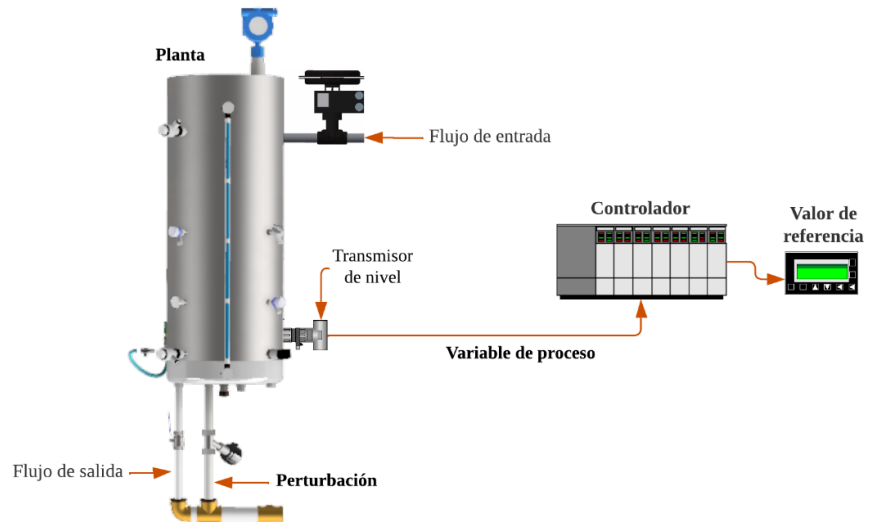


Fig 1. Elementos del lazo de control.

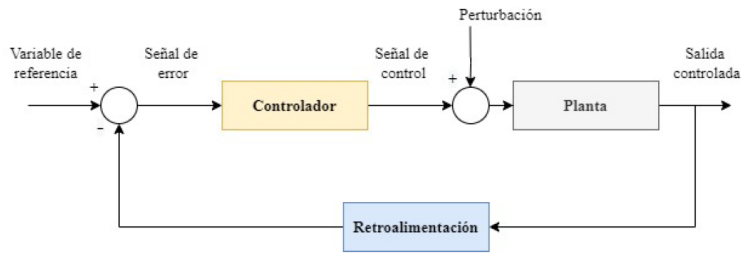


Fig 2. Lazo de control cerrado.

De tal manera definiendo el lazo de control cerrado con los respectivos elementos y con las especificaciones a cumplir guiados en la figura 2, se tiene que:

- El valor de referencia a cubrir será de 40 cm de nivel.
- El controlador a utilizar será de la arquitectura proporcional, integral y derivativo (PID).
- La perturbación será introducida al activar la válvula neumática On/Off, ubicada debajo del tanque.
- La planta será el tanque abierto. La válvula manual que se encuentra debajo del mismo, se mantendrá con una apertura del 50 % para contar con un flujo de salida constante. El flujo de entrada, será la variable a regular por el controlador para tener el nivel de líquido deseado.
- La retroalimentación estará dada por el sensor de nivel (Cerabar T PMC131). Este transmisor se fundamenta bajo un principio de presión absoluta y relativa.

5. Proceso

El punto de partida será obtener la función de transferencia que represente la dinámica del sistema. Se sabe que se puede realizar por dos diferentes métodos, experimental o un planteamiento analítico – matemático. En el caso de la práctica se realizará mediante datos experimentales, ya que se quiere dar una opción para cuando el sistema presente dificultades en el modelamiento matemático o no se disponga de ciertos los datos, por ejemplo, no se tiene el valor real del caudal de salida en la planta. Además, hay tomar en consideración que la mejor manera de obtener un modelo ligado al proceso es mediante valores obtenidos en el campo físico.

- **Parámetros para hallar la función de transferencia**

El método empleado para hallar la función de transferencia que represente la dinámica del sistema se denomina modelo Autorregresivo con Entrada Externa (ARX), misma que demanda en su ecuación en diferencia contar con una señal PBRs para los valores de entrada al sistema, y también disponer de los datos que representan la respuesta del sistema. Se reemplaza en la ecuación, y se tendrá como resultado una serie de incógnitas con respecto a muestras retrasadas; las cuales serán resueltas con el método denominado Mínimos Cuadrados Recursivos. Todo el proceso anteriormente nombrado es el principio en el que se basa la herramienta a utilizar que es la Toolbox Ident del software Matlab.

De tal forma se inicia con el planteamiento de la señal PRBS. Mediante pruebas realizadas se obtiene que para llegar al valor de 40 cm de nivel se debe de introducir en la apertura de la válvula un escalón unitario de 50 %. En la figura 3, se muestra la respuesta del sistema a la entrada escalón.

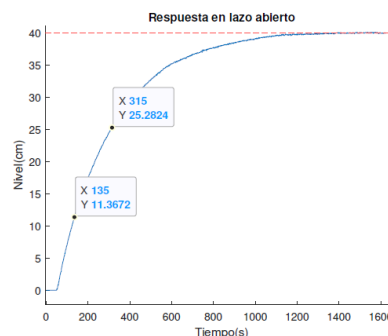


Fig 3. Respuesta en lazo abierto del sistema.

Una consideración a tomar con respecto a la planta, es que el sistema es no lineal; produciendo que el salto del escalón del ruido blanco no puede variar significativamente, debido a los datos erróneos que puede concebir esta acción. Para ello lo que proponen los criterios de control, es el de conocer la dinámica de la planta para variar una amplitud de +/- 5% alrededor de un punto de equilibrio identificado. Aplicando la consideración anterior, la variación de la señal PRBS se centra entre el 50% y 52% de la apertura de la válvula proporcional.

Conociendo los rangos a tener en la entrada del sistema se plantea obtener 7 bits de muestra, es decir 128 datos. Con ayuda del software Matlab se generan los valores pseudoaleatorios a introducir en la planta, en la figura 4 se puede evidenciar el código, donde se debe introducir la cantidad de datos y el rango.

```
a=idinput(128,'prbs',[50 52])
```

Fig 4. Comando para generar la señal PRBS.

De igual manera, se tiene que tomar en consideración cada cuanto se va aplicar el cambio del escalón de la señal PRBS. Sabiendo que la respuesta en lazo abierto se puede aproximar a un sistema de primer orden se utiliza el método de identificación de función de transferencia Smith.

Una función de primer orden vendrá dada por la siguiente ecuación:

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

Para poder hallar las incógnitas de la función, Smith considerando el tiempo al 28.3% y 63.2% de la salida del sistema (nivel). Con ello se remplaza en las ecuaciones preestablecidas por el método y se encuentra los valores de cada variable. De tal manera, se tiene en la figura 3:

$$t_{63} = 315 \text{ s}$$

$$t_{28} = 135 \text{ s}$$

Para estimar un tiempo de muestra se utiliza únicamente el valor de τ , debido a que se debe garantizar que la señal pseudoaleatoria PRBS produzca que la

salida del sistema represente al menos 3 veces τ . Con ayuda de la ecuación y de constantes del método tenemos:

$$\tau = at_{28} + bt_{63} \quad (2)$$

$$\tau = -1.5(135) + 1.5(315)$$

$$\tau = 270$$

Según criterios de control 3τ se asegura haciendo 10 o 20 veces más rápido, se tiene que:

$$T_s < \frac{\tau}{10}$$

$$T_s < \frac{270}{10}$$

$$T_s < 27 \text{ s}$$

El tiempo de muestra estará ligado a 27 s. De esa forma la entrada tomada para el análisis es la señal PRBS, y la salida es la respuesta del nivel del tanque ante el valor de excitación (figura 5).

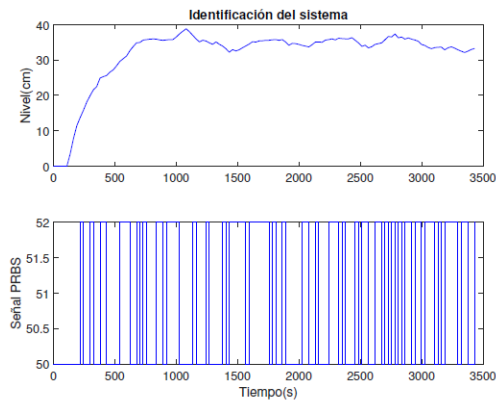


Fig 5. Entrada y respuesta del sistema aplicando una señal PRBS, cada 27 segundos para 128 muestras.

- **Identificación de la función de transferencia**

4.1. Se inicia el proceso abriendo Ident Toolbox en Matlab, escribiendo **ident** en el **Command Window** se tiene la siguiente pestaña:

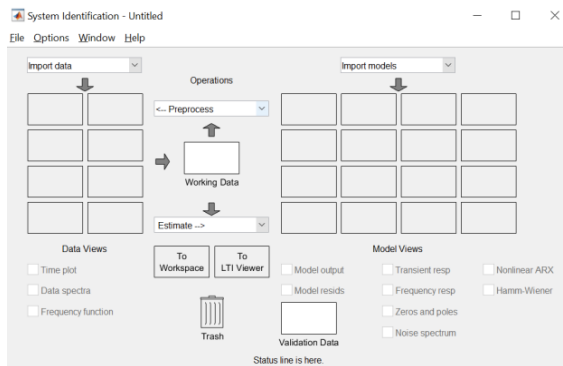


Fig 6. Ident Toolbox.

4.2. En el menú desplegable de **Import data**, seleccionamos **Time domain data**:

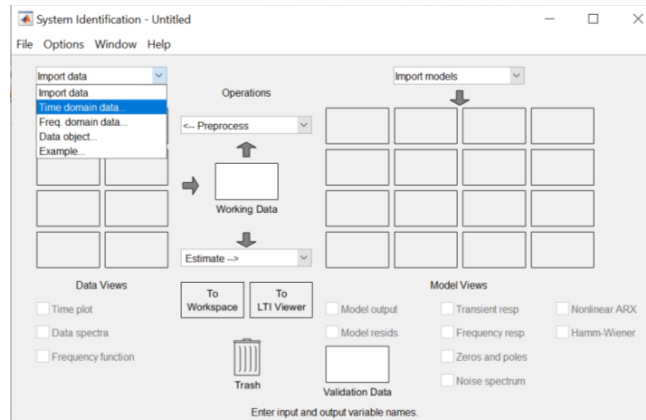


Fig 7. Importar datos.

4.3. En la siguiente pestaña introducimos los datos de la entrada (señal PRBS) y la salida del sistema. Establecemos el tiempo de inicio, y cada cuanto se ejecutó el muestreo de los datos.

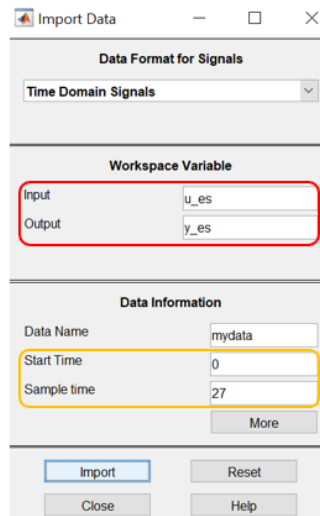


Fig 8. Establecer los parámetros para importar los datos.

4.4. Presionamos importar y tendremos los datos cargados. En el menú desplegable **Preprocess**, removemos las tendencias y los datos medios (figura 9).

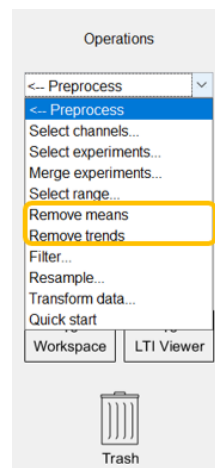


Fig 9. Remover las tendencias y datos medios.

4.5. De tal manera se mantendrá la siguiente pestaña (figura 10).

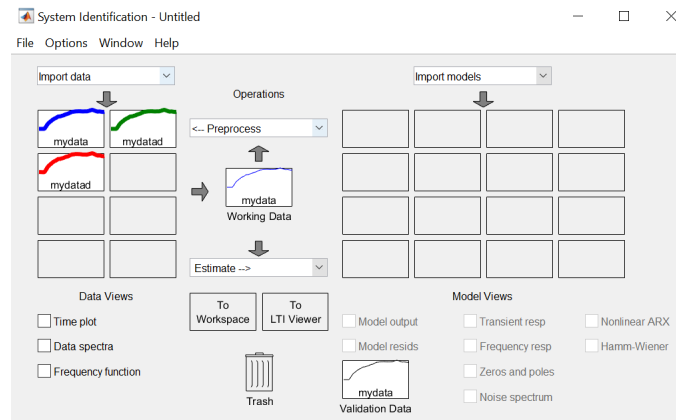


Fig 10. Pestaña del sistema de identificación.

4.6. Nos dirigimos al menú desplegable **Estimate**. La estimación de la función de transferencia se puede realizar mediante la opción **Transfer Function Models** estableciendo los polos y ceros; o **Process Models** el cual se utilizará, ya que conocemos que el sistema se puede aproximar a uno de primer orden.

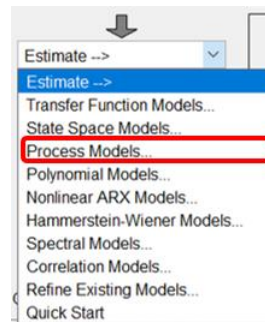


Fig 11. Process Models

4.7. Se debe de activar en la pestaña la opción de **Delay**, debido a que se tiene un retardo en el sistema, este retraso se produce por la distancia de la tubería entre el depósito de agua y el tanque abierto, además, la ondulación que existe antes de que el sensor empiece a tomar los datos es otro factor de la demora. Dando click en **Estimate**, obtenemos las constantes de la función de transferencia (figura 12).

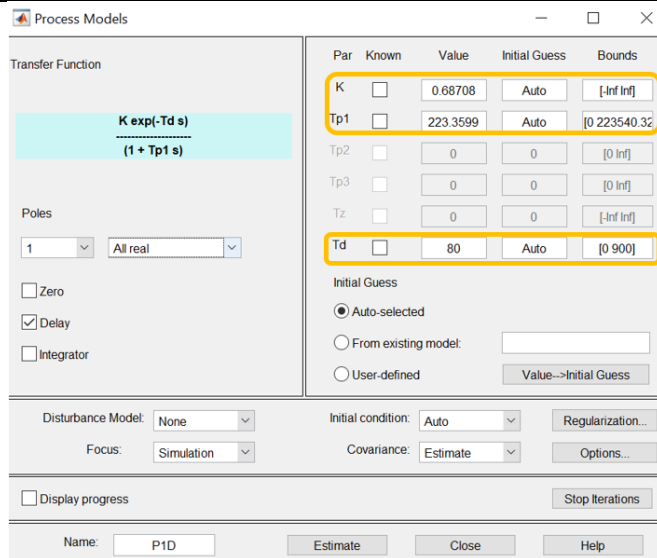


Fig 12. Constantes de la función de transferencia.

4.8. El siguiente paso será validar la función de transferencia. Nos dirigimos a **Validation Data** en la pestaña principal.

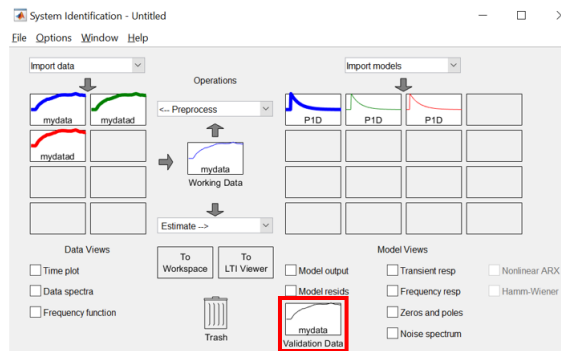


Fig 13. Validación de la función de transferencia.

4.9. Al ejecutar la validación de la función de transferencia se presenta un 83% de semejanza con el sistema (figura 14); sabiendo que el porcentaje de similitud debe de ser mínimo del 70%, se da la aceptación de la función.

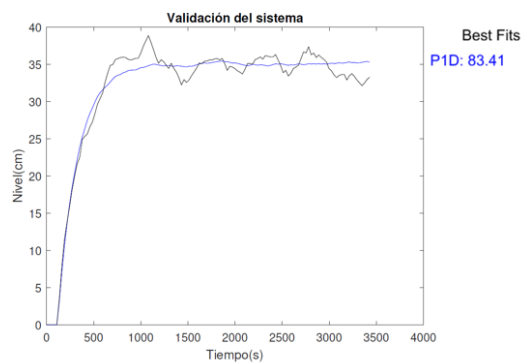


Fig 14. Validación de la función de transferencia.

De tal manera la función de transferencia que representa la dinámica del sistema será:

$$G_p(s) = \frac{0.6871}{223.4s + 1} e^{-80s} \quad (3)$$

- **Control PID**

En la arquitectura del control PID se seleccionará únicamente la ganancia proporcional e integral. La justificación de la selección va ligada a que si se inyecta en el sistema únicamente una parte proporcional se tendrá problemas con la oscilación, por lo cual se debe de acompañar por una segunda ganancia. Se descarta utilizar la parte derivativa porque el sistema no tiene problemas en la sección transitoria; más bien el proceso no sigue el valor de referencia, por lo que se debe proponer una ganancia integral, dando así que el controlador recuerde el comportamiento de la curva de tendencia del sistema e introduzca una constante para llegar al valor de nivel propuesto. Además, un punto a tratar acerca de la función de transferencia que representa el llenado del tanque, es el retardo; el trabajar con un exponencial en la búsqueda de los valores PID para el control, provocara problemas en la estabilidad y en la dinámica del sistema.

La mejor solución para trabajar con sistemas que mantienen un retardo es aplicar el método de aproximación PADE, mismo que propone aproximar la función a otra racional polinómica. De esa manera se obtiene:

$$G_{sp}(s) = \frac{-0.6871s + 0.01718}{223.4s^2 + 6.584s + 0.025} \quad (4)$$

Mediante el método de asignación de polos aplicado a la ecuación 4, se halla las ganancia proporcional e integral.

$$k_p = 1.594$$

$$k_i = 0.006546$$

Finalmente, introduciendo las ganancias en el lazo de control cerrado se tiene que el tiempo de respuesta y el tiempo de asentamiento mejoran con respecto al sistema en lazo abierto, con un valor de excitación de 50. La figura 15 perteneciente a una simulación, permite observar que la constante proporcional e integral provocan llevar al sistema al valor de referencia de 40 cm.

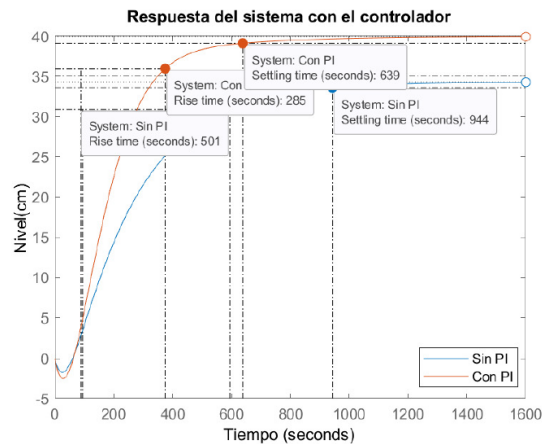


Fig 15. Respuesta del sistema con el controlador y sin el controlador.

- **Implementación del controlador.**

Con la constitución del controlador proporcional – integral, se ingresa en el bloque PID.FF que dispone el PLC Schneider las ganancias obtenidas y se verifica la respuesta. El bloque demandará la constante proporcional y el tiempo integral, por lo cual se realizará la conversión teniendo que:

$$k_p = 1.594$$

$$t_i = 243.5075$$

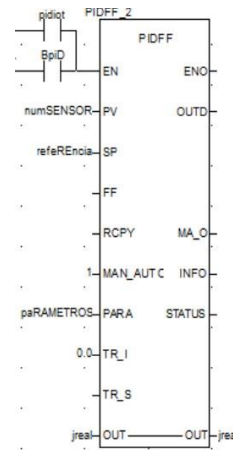


Fig 16. Bloque PID.FF

Se ingresan todos los parámetros que se requieran en la herramienta PID.FF (figura 16), y de obtienen los siguientes resultados:

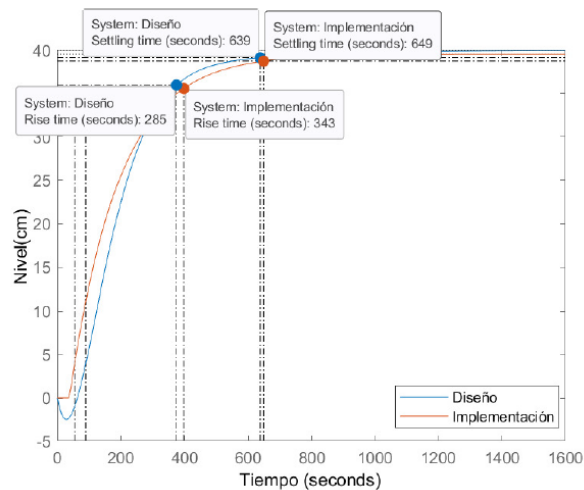
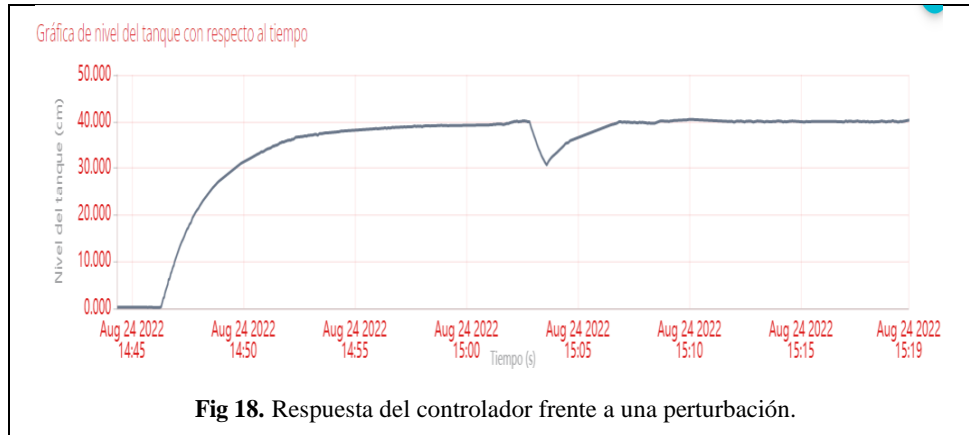


Fig 17. Resultado de la implementación del PID en la planta.



Si se requiere una mayor información y detalles de ciertos procesos, se puede revisar el proyecto de titulación adjunto. “Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca”

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Establecer una apertura para la válvula manual. (El valor puede ser el que usted desee).
2. Encender la bomba con una apertura de la válvula proporcional definida. (El valor puede ser el que usted desee).
3. Graficar el comportamiento del sistema con un flujo de entrada y salida constante.
4. Centrarse en un valor de nivel para seleccionar los límites de la señal PRBS.
5. Establecer el tiempo de muestreo.
6. Obtener el comportamiento del sistema. A partir de la señal PRBS y del tiempo de muestra.
7. Obtener la función de transferencia.
8. Diseñar un control PID con el método que usted desee, se recomienda utilizar Ziegler Nichols, pidtool, Sisotool, método gráfico, o relocalización de polos y ceros.
9. Implementar en la planta el control PID.

RESULTADO(S) OBTENIDO(S):

Al término de la presente práctica el estudiante debe ser capaz de generar un control PID para la variable de nivel de tanque, del laboratorio LACTI.

CONCLUSIONES:

Esta práctica permite que los estudiantes puedan familiarizar los conocimientos impartidos en clase, con un entorno laboral.

RECOMENDACIONES:

Verificar que los estudiantes cumplan con las normas de seguridad.
Efectuar las debidas indicaciones realizadas.

REFERENCIAS:

- [1] S. Electric, Plataforma de Automatización Modicon Premium, Telemecanique, 2005.
- [2] S. Electric, Lenguajes y estructura del programa, 2009.
- [3] R. Perez, «¿Qué es una máquina virtual?,» 2009. [En línea]. Disponible en: <https://grch.com.ar/docs/so/apuntes/VirtualBox.pdf>. [Último acceso: 23 Junio 2022].

Docente / Técnico Docente:

Firma:

Anexo - Apertura de la válvula manual





FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: Control de nivel en la planta A del laboratorio de LACTI

Nro Práctica: 4

OBJETIVOS ALCANZADOS:

1. Diseñar un control proporcional, integral y derivativo (PID) en la planta A del laboratorio de LACTI.
2. Implementar el controlador en el PLC Schneider Modicon Premium.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Establecer una apertura para la válvula manual. (El valor puede ser el que usted desee).
2. Encender la bomba con una apertura de la válvula proporcional definida. (El valor puede ser el que usted desee).
3. Graficar el comportamiento del sistema con un flujo de entrada y salida constante.
4. Centrarse en un valor de nivel para seleccionar los límites de la señal PRBS.
5. Establecer el tiempo de muestreo.
6. Obtener el comportamiento del sistema. A partir de la señal PRBS y del tiempo de muestra.
7. Obtener la función de transferencia.
8. Diseñar un control PID con el método que usted desee, se recomienda utilizar Ziegler Nichols, pidtool, Sisotool, método gráfico, o relocalización de polos y ceros.
9. Implementar en la planta el control PID.

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

Nombre del estudiante: _____

Firma del estudiante: _____

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: IOT mediante el PLC Schneider y el módulo IOT2040

Nro Práctica: 5

OBJETIVOS:

1. Uso del Unity Pro XL para el manejo de la planta.
2. Uso del módulo IOT2040 para la recepción de datos en la nube.
3. Manejar node-red para establecer una conexión entre el PLC Schneider y Ubidots.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

- a) Conocimientos del manejo de PLC's.
- b) Conocimientos del manejo del software Unity Pro XL.
- c) Revisar y ejecutar la guía de práctica 1.
- d) Revisar y ejecutar la guía de práctica 2.
- e) Conocimiento de redes industriales.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
PLC Schneider	-	Schneider Electric	-
Virtual Box con Windows 7	1	-	-
Virtual Box Extension Pack	1	-	-
Cable TSXCUSB485	1	Schneider Electric	-
Unity Pro XL	-	Schneider Electric	-
IOT2040	-	Siemens	-

Tabla 1. Equipos, instrumentos y software de la guía.

3. Exposición

PLC Schneider

Su modelo es perteneciente a la familia Modicon Premium, utilizado en la industria manufacturera por su capacidad de memoria extensible y su velocidad de ejecución creciente.

Dispone hasta 7 Mb de memoria programada, además de un CPU de alto rendimiento con 37 ns por instrucción. Tiene extensión de racks y una serie de posibilidades para buses de campo, buses de maquina y buses tanto de accionamiento, como para captación; mismos que sirven para una alta oferta de comunicación como: Profibus DP, Interbus, Modbus Plus y As Interface. La demanda de sus módulos es para aplicaciones específicas de seguridad, contadores, control de posición, movimiento, peso, procesamiento de reflejos y almacenamiento de datos [1].

Unity Pro XL

Es un software que sirve para programar el PLC y los diferentes módulos comprados por el cliente. Este programa cuenta con herramientas de

simulación, diferentes lenguajes de programación, librerías de bloques para diferentes funciones como conexión por Modbus [2].

Máquina Virtual

Es un software que permite simular a un ordenador dentro +de tu propio ordenador, es decir, se puede probar distintos sistemas operativos dentro de tu ordenador sin cambiar al que tienes [3].

Entre los sistemas operativos que pueden soportar la Virtual Box son: Linux, Windows, Mac Os X y Solaris [3].

4. Proceso para acceder al módulo IOT2040

4.1. Hay que dirigirse a cualquier navegador y escribir la IP <http://192.168.65.92:1880/>, con esto se va a poder programar por node-red, obteniendo una ventana como la que se muestra a continuación:

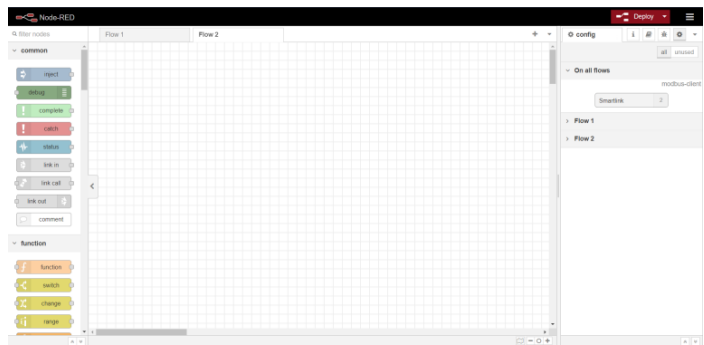


Fig 1. Ventana de Node-red.

4.2. Ya ingresado a node-red es necesario descargar las librerías de modbus y ubidots que son importantes para la conexión del PLC y de la página web, por eso, primero hay que dirigirse al **Manege palette**, esto se lo realiza de la siguiente manera:

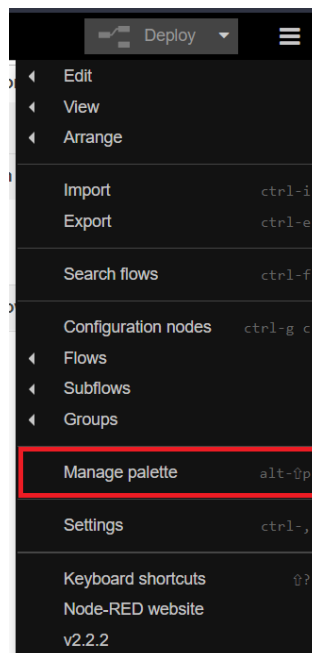


Fig 2. Manage palette.

4.3. De ahí aparece una ventana, y hay que dirigirse a **install** y se busca e instala la librería **node-red-contrib-modbus**, después, se repite el proceso con **ubidots** descargando la librería **ubidots-nodered**, como se muestra a continuación:

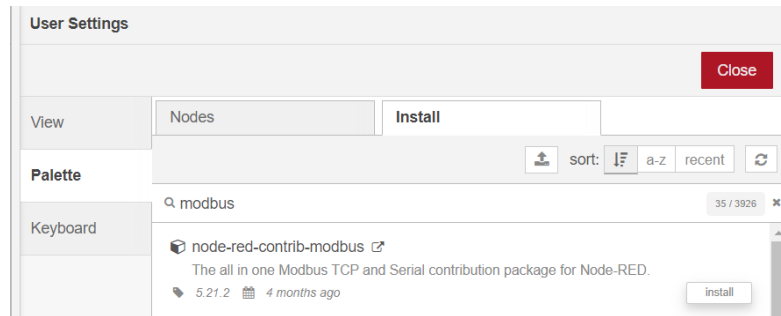


Fig 3. Instalación de librería de Ubidots.

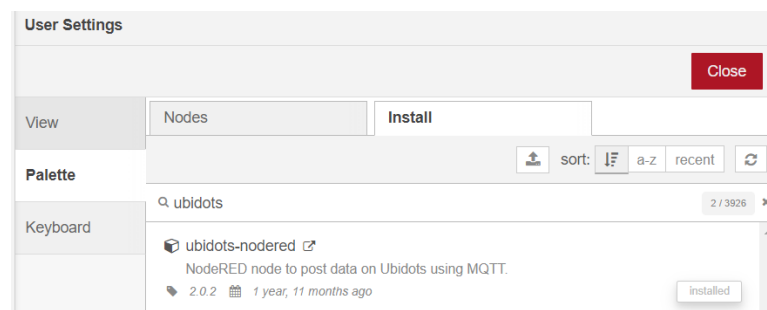


Fig 4. Ubidots instalado.

- **Recibir datos del PLC**

4.4. Después, en la lista de nodos que se encuentra en la parte izquierda de la pantalla se buscan los nodos de **Modbus-Read**, **function** y **ubidots_out** y se los arrastran a ventana de flujo, para enviar datos desde el PLC a la página web, para que queden de la siguiente forma:

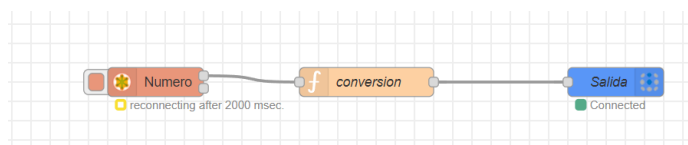


Fig 5. Esquema de conexión entre Modbus y Ubidots.

4.5. Se configuran los parámetros de modbus, y para eso se hace doble click en el nodo de **Modbus-Read**, y será configurado con los siguientes parámetros:

Fig 6. Parámetros de configuración de Modbus.

Donde:

Name: nombre del nodo.

Topic: una descripción de la utilidad del nodo.

Unit-id: número de identificación del nodo.

FC: código de función de Modbus, donde la función 1 te permite recibir valores tipo booleano y la función 3 permite recibir valores numéricos.

Address: dirección del PLC de donde se obtienen los valores.

Quantity: número de valores que va a recibir.

Poll rate: tiempo que deberá esperar para recibir otro dato del PLC.

Server: configuración de la comunicación Modbus.

La configuración del server es la siguiente:

Fig 7. Parámetros de conexión.

Donde lo importante es colocar bien la dirección IP del PLC Schneider. Y con esto, ya se configuró el nodo de Modbus.

4.6. Se configura el nodo de Ubidots, para eso se ingresa a la cuenta de Ubidots y se copia el token de la cuenta, para eso nos hay que dirigirse a **API Credentials** como se muestra a continuación:

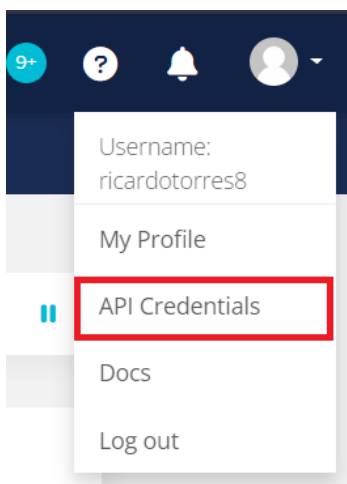


Fig 8. Api Credentials.

4.7. Y se hace click en copiar, como se observa en la siguiente figura:

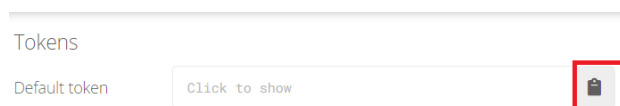


Fig 9. Token creada por Ubidots.

4.8. Se dirige al node-red y se hace doble clic en el nodo de ubidots, y aparecerá la siguiente ventana:

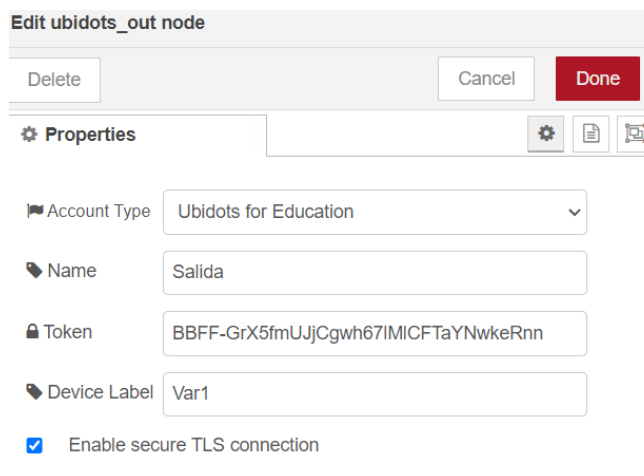


Fig 10. Parametros de configuración del nodo de Ubidots.

Donde tenemos los siguientes parámetros:

Account Type: tipo de cuenta de ubidots, preferible usar Ubidots for Education porque este tipo de cuenta es gratis.

Name: Nombre del nodo.

Token: se pega el token de nuestra cuenta de ubidots.

Device Label: nombre de la variable que aparecerá en nuestra página web.

4.9. Para acabar con node-red, es necesario agregar un nombre a cada variable obtenida, y para eso se utilizará la siguiente función:

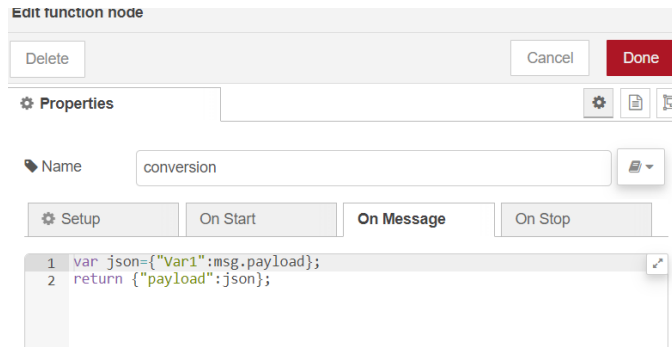


Fig 11. Conversión de msg.payload a json.

4.10. Luego hay que dirigirse otra vez a la cuenta de ubidots, y para comprobar que los datos se están pasando del node-red a Ubidots se dirigen a **Devices** y ahí debe aparecer la variable que se está recibiendo, como se observa en la figura 12:

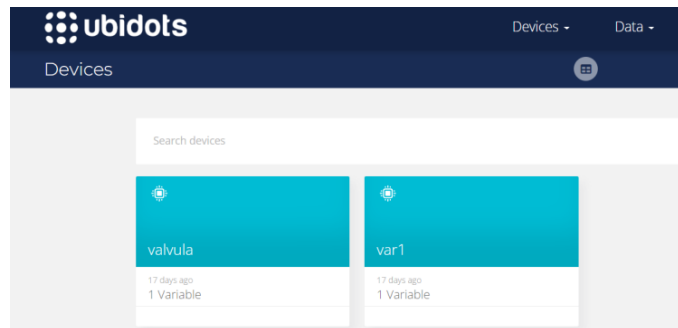


Fig 12. Variables almacenadas en Ubidots.

4.11. Debe dirigirse a **Data** y **Dashboard** para crear el indicador de los datos, y se hace clic en el icono de + que se encuentra a la derecha de la página, y se abre una ventana parecida a la siguiente figura:

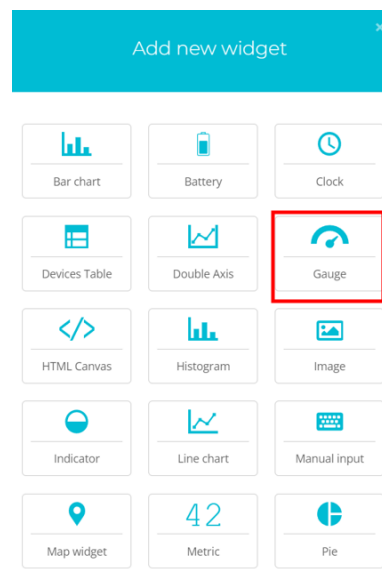


Fig 13. Selección de herramienta para el dashboard.

4.12. Y se selecciona en este caso **Gauge** para indicar le número enviado. Después, aparecerá otra ventana y se hace click en **Add Variables**, como se observa en la siguiente figura:

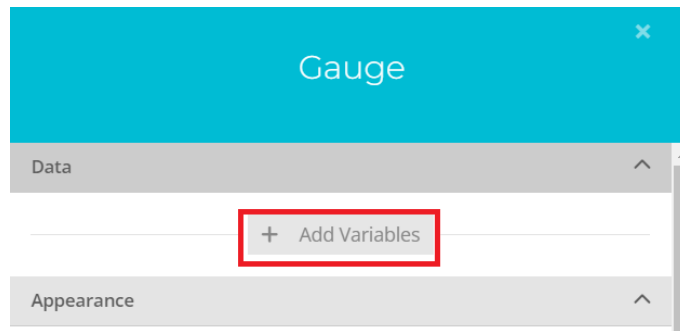


Fig 14. Agregar la variable de la herramienta.

4.13. Se selecciona la variable que se recibe del node-red, como se muestra a continuación

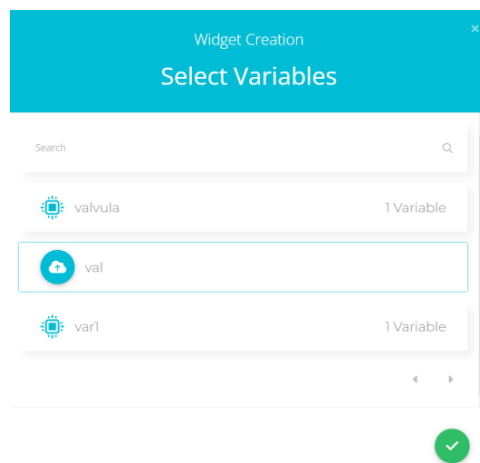


Fig 15. Selección de la variable.

4.14. Finalmente se crea el indicador de la variable enviada.

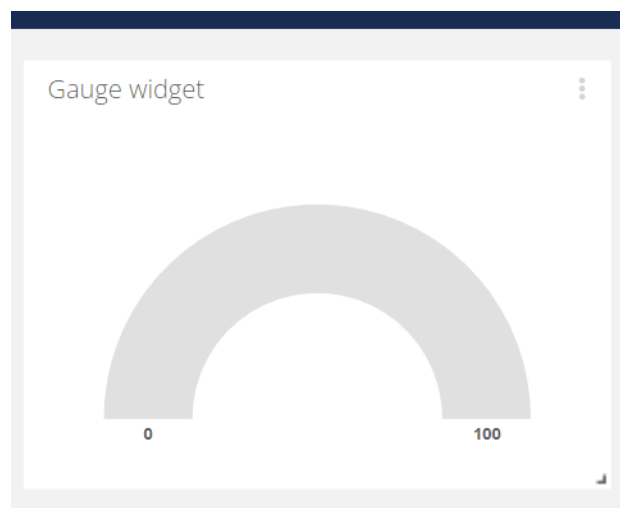


Fig 16. Indicador.

- **Enviar datos de Ubidots al PLC**

4.15. Hay que crear primero las variables que se van a enviar del Ubidots, en este ejemplo se usan variables booleanas para encender válvulas. Las variables utilizadas en este ejemplo son las siguientes:

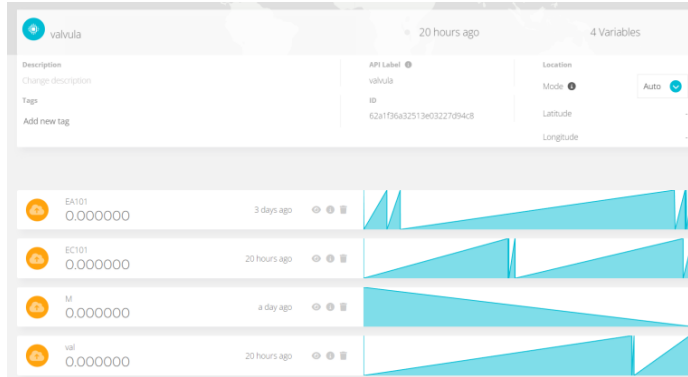


Fig 17. Lista de variables.

4.16. Para crear las variables hay que dirigirse en Ubidots/ Device/ Device, y crear la variable siguiendo los pasos de la figura 18:

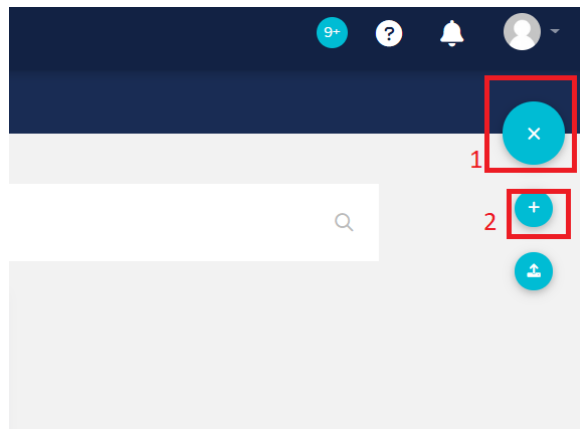


Fig 18. Pasos para crear dispositivo en Ubidots.

4.17. Existen varios tipos de dispositivos que pueden ser creados, pero se debe escoger dispositivo en blanco y se nombra al dispositivo a gusta del usuario. Como se observa en la figura 19:

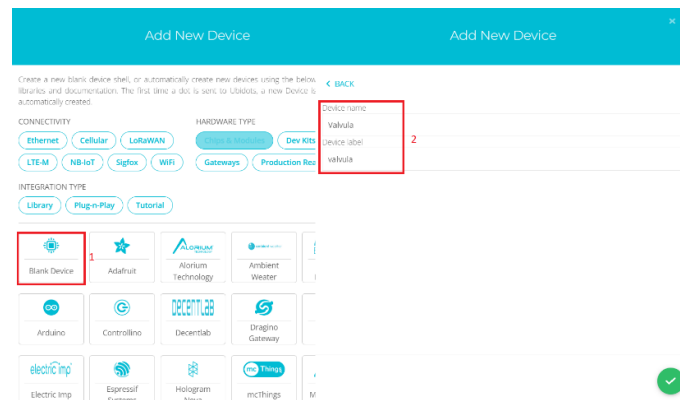


Fig 19. Crear dispositivo en Ubidots.

4.18. Aparece un dispositivo en blanco y hay que crear las variables que van a enviar al PLC, para eso hay que hacer click en el dispositivo creado y dirigirse a Add Variables y escoger Raw y se nombra la variable, como se observa en la figura 20:

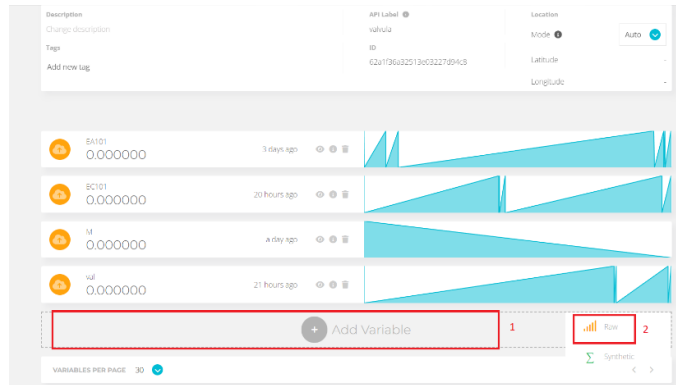


Fig 20. Crear variable en dispositivo creado anteriormente.

4.19. En node-red hay que ingresar los bloques de ubidots_in, function y Modbus_write. Para formar un lazo como la siguiente figura:

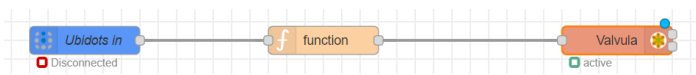


Fig 21. Lazo para comunicación de Ubidots y el PLC Schneider.

4.20. Y se configura el bloque de ubidots_in con los siguientes parámetros:

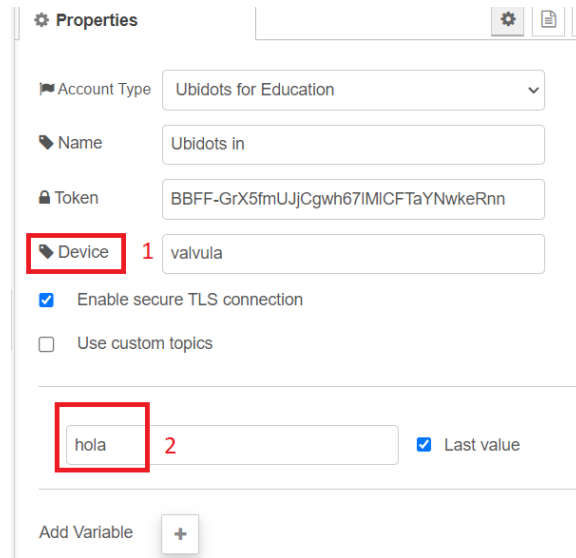


Fig 22. Parámetros de bloque Ubidots_in.

Donde:

1. Device: Nombre del dispositivo creado en ubidots.
2. Name label: nombre de la variable.

4.21. Hay que recibir y guardar las variables que llegan del Ubidots, ya que estas vienen con el nombre de la variable y el valor.

Por lo que se utiliza el siguiente código para recibir dichas variables:

Fig 23. Programación para obtener datos de Ubidots_in.

4.22. Y finalmente se configura el bloque de Modbus_write, este bloque tiene los mismos parámetros que el Modbus_read.

Si se requiere una mayor información y detalles de ciertos procesos, se puede revisar el proyecto de titulación adjunto. “Manual de prácticas de la planta A, empleando el sistema conectado al PLC Schneider Modicon Premium del laboratorio de LACTI en la Universidad Politécnica Salesiana sede Cuenca”

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Conectarse a Node-red mediante la ip del IOT2000.
2. Descargar las librerías de Modbus y Ubidots.
3. Configurar los nodos de modbus y Ubidots para establecer la conexión.
4. Realizar la conversión de datos de json a msg.payload.
5. Configurar Ubidots para recibir las variables.
6. Manejar el micromaster desde Ubidots usando el IOT2000.

RESULTADO(S) OBTENIDO(S):

Al término de la presente práctica el estudiante debe ser capaz de configurar el modula IOT, para manejar la planta desde la red.

CONCLUSIONES:

Esta práctica permite que los estudiantes puedan familiarizar los conocimientos impartidos en clase, con un entorno laboral.

RECOMENDACIONES:

Verificar que los estudiantes cumplan con las normas de seguridad.
Efectuar las debidas indicaciones realizadas.

REFERENCIAS:

- [1] S. Electric, Plataforma de Automatización Modicon Premium, Telemecanique, 2005.
- [2] S. Electric, Lenguajes y estructura del programa, 2009.
- [3] R. Perez, «¿Qué es una máquina virtual?,» 2009. [En línea]. Disponible en: <https://grch.com.ar/docs/so/apuntes/VirtualBox.pdf>. [Último acceso: 23 Junio 2022].

Docente / Técnico Docente:

Firma:



FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES

CARRERA: Ingeniería Mecatrónica

ASIGNATURA:

TÍTULO PRÁCTICA: IOT mediante el PLC Schneider y el módulo IOT2040

Nro Práctica: 5

OBJETIVO ALCANZADOS:

1. Uso del Unity Pro XL para el manejo de la planta.
2. Uso del módulo IOT2040 para la recepción de datos en la nube.
3. Manejar node-red para establecer una conexión entre el PLC Schneider y Ubidots.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Conectarse a Node-red mediante la ip del IOT2000.
2. Descargar las librerías de Modbus y Ubidots.
3. Configurar los nodos de modbus y Ubidots para establecer la conexión.
4. Realizar la conversión de datos de json a msg.payload.
5. Configurar Ubidots para recibir las variables.
6. Manejar el micromaster desde Ubidots usando el IOT2000.

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

Nombre del estudiante: _____

Firma del estudiante: _____

ANEXO C

Encuestas para la validación de prácticas



MANUAL DE PRÁCTICAS DE LA PLANTA A, EMPLEANDO EL SISTEMA CONECTADO AL PLC SCHNEIDER MODICON PREMIUM DEL LABORATORIO DE LACTI EN LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE CUENCA

Encuesta de validación del manual de prácticas

Validador:

Angel P. Hernandez P.

De su puntuación, según considere la viabilidad de cada práctica. Teniendo dentro del rango como:

1. Totalmente de acuerdo.
2. De acuerdo.
3. Ni de acuerdo ni en desacuerdo.
4. En desacuerdo.
5. Totalmente desacuerdo.

Con respecto al formato de la práctica

Nº	Criterio	Puntuación
1	La redacción, ortografía, tipo y tamaño de fuente permite una comprensión de las prácticas.	2
2	Las tablas e imágenes presentadas son las adecuadas para el desarrollo de las prácticas.	1
3	La estructura de las guías es la apropiada, y abarca todo el proceso a realizar en las prácticas.	1
4	El formato permite cumplir los objetivos de cada una de las prácticas.	2

Con respecto al procedimiento

Nº	Criterio	Puntuación
1	Se presenta una secuencia lógica, clara y comprensible.	1
2	Se aporta la información necesaria para el desarrollo de las prácticas.	1
3	Los procedimientos a realizar están acordes a los conocimientos impartidos en clase.	1
4	Los equipos, materiales y el software de programación están disponibles dentro del laboratorio.	1
5	El procedimiento permite cumplir los objetivos de cada una de las prácticas.	1

Otras consideraciones

Nº	Criterio	Puntuación
1	Las actividades planteadas sirven para un consenso de conocimientos, que promueven una acción de aprendizaje (Analizar, observar, deducir e investigar).	1
2	Las recomendaciones ayudan a la ejecución de cada práctica.	1
3	Es necesario disponer de manuales de prácticas que sirvan como punto de partida para la formación de futuros profesionales.	1

Observaciones

Angel P. Hernandez P.

Firma

Fecha: 2022/06/07



MANUAL DE PRÁCTICAS DE LA PLANTA A, EMPLEANDO EL SISTEMA CONECTADO AL PLC SCHNEIDER MODICON PREMIUM DEL LABORATORIO DE LACTI EN LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE CUENCA

Encuesta de validación del manual de prácticas

Validador:

Monica Romero

De su puntuación, según considere la viabilidad de cada práctica. Teniendo dentro del rango como:

1. Totalmente de acuerdo.
2. De acuerdo.
3. Ni de acuerdo ni en desacuerdo.
4. En desacuerdo.
5. Totalmente desacuerdo.

Con respecto al formato de la práctica

Nº	Criterio	Puntuación
1	La redacción, ortografía, tipo y tamaño de fuente permite una comprensión de las prácticas.	1
2	Las tablas e imágenes presentadas son las adecuadas para el desarrollo de las prácticas.	1
3	La estructura de las guías es la apropiada, y abarca todo el proceso a realizar en las prácticas.	1
4	El formato permite cumplir los objetivos de cada una de las prácticas.	1

Con respecto al procedimiento

Nº	Criterio	Puntuación
1	Se presenta una secuencia lógica, clara y comprensible.	1
2	Se aporta la información necesaria para el desarrollo de las prácticas.	1
3	Los procedimientos a realizar están acordes a los conocimientos impartidos en clase.	1
4	Los equipos, materiales y el software de programación están disponibles dentro del laboratorio.	1
5	El procedimiento permite cumplir los objetivos de cada una de las prácticas.	1

Otras consideraciones

Nº	Criterio	Puntuación
1	Las actividades planteadas sirven para un consenso de conocimientos, que promueven una acción de aprendizaje (Analizar, observar, deducir e investigar).	1
2	Las recomendaciones ayudan a la ejecución de cada práctica.	1
3	Es necesario disponer de manuales de prácticas que sirvan como punto de partida para la formación de futuros profesionales.	1

Observaciones

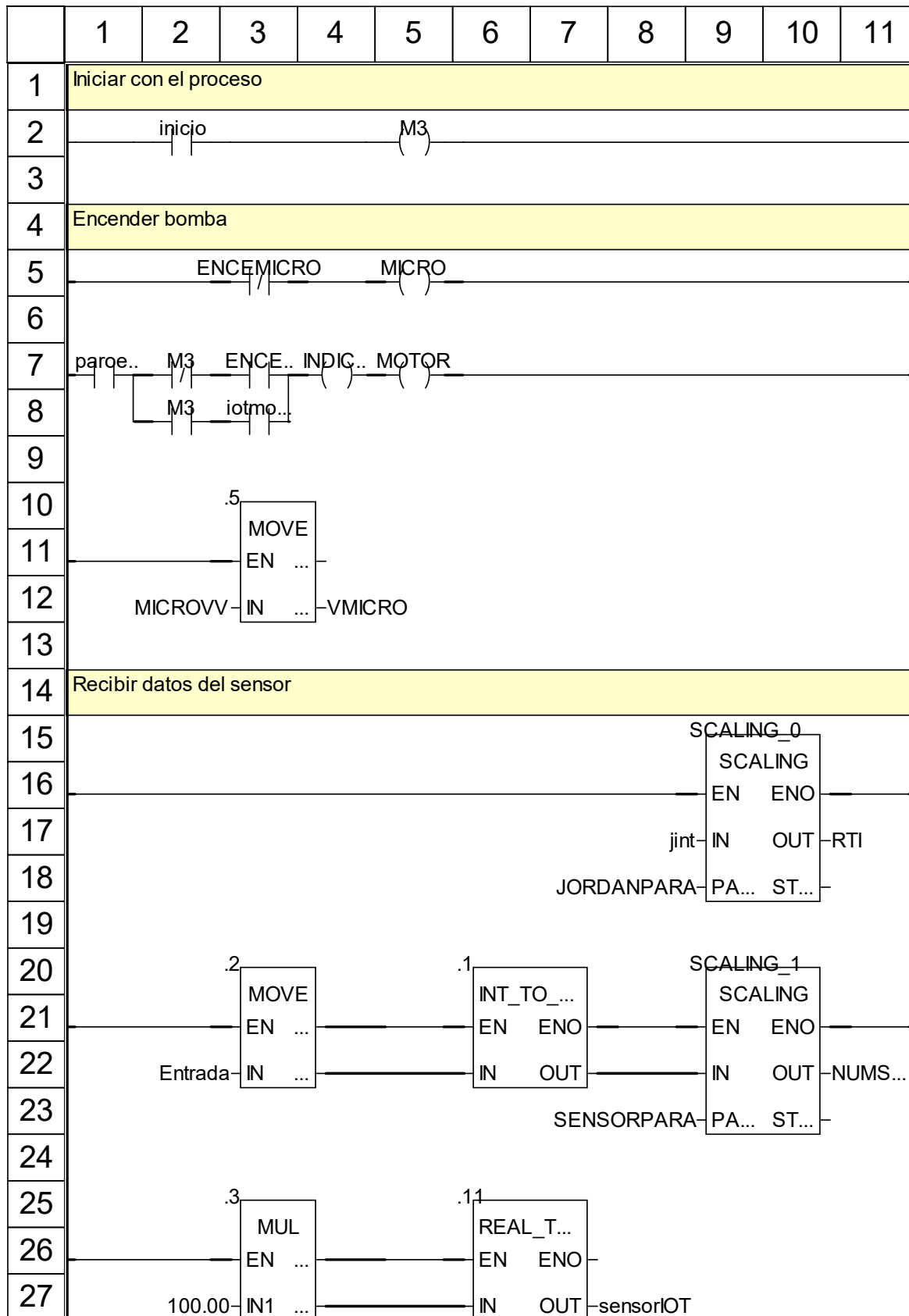
.....
Firma

Fecha: *29-07-2022*

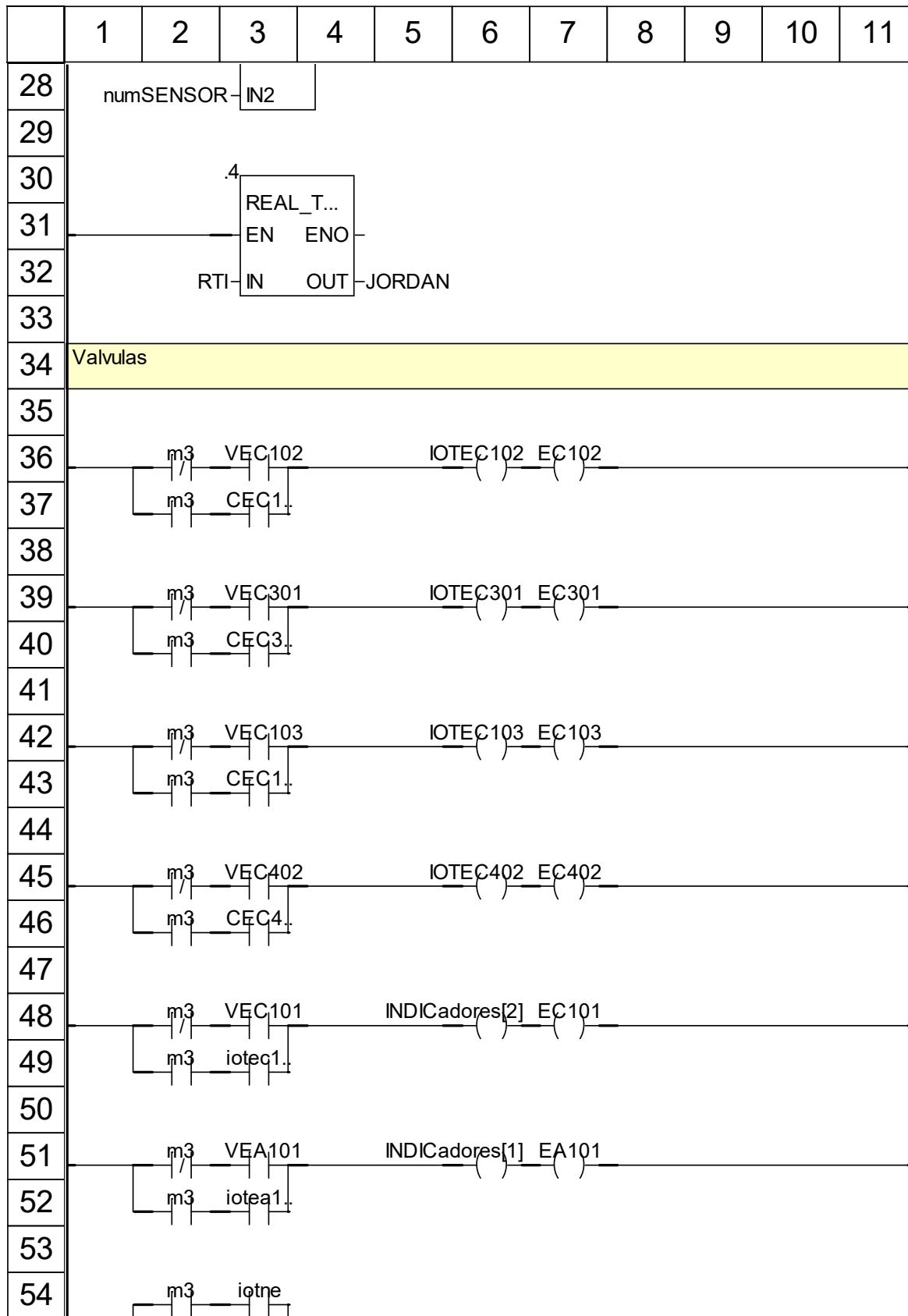
ANEXO D

Programa realizado en el software Unity Pro XL y tabla de variables

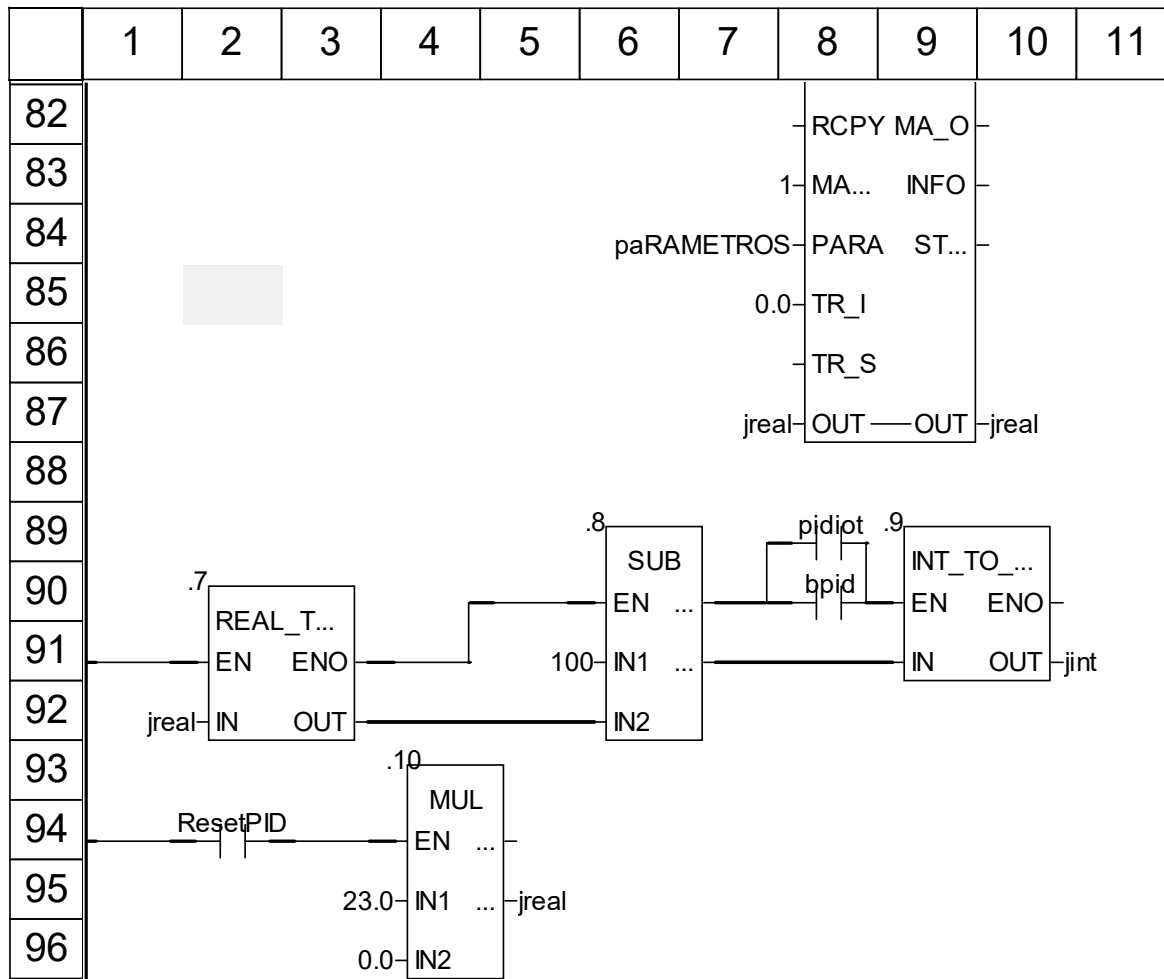
Programa



Programa



Programa



Variables

Nombre	Tipo	Dirección	Valor	Comentario	Marcas de	Derechos	Intercambio	Tarea	Retener	
BPID	EBOOL	%M5			Ninguno	NO	1	MAST	1	
CEA103	EBOOL	%M56			Ninguno	NO	1	MAST	1	
CEA104	EBOOL	%M55			Ninguno	NO	1	MAST	1	
CEC102	EBOOL	%M50			Ninguno	NO	1	MAST	1	
CEC103	EBOOL	%M52			Ninguno	NO	1	MAST	1	
CEC301	EBOOL	%M51			Ninguno	NO	1	MAST	1	
CEC302	EBOOL	%M57			Ninguno	NO	1	MAST	1	
CEC305	EBOOL	%M54			Ninguno	NO	1	MAST	1	
CEC402	EBOOL	%M53			Ninguno	NO	1	MAST	1	
CEV102	EBOOL	%M58			Ninguno	NO	1	MAST	1	
CEV103	EBOOL	%M59			Ninguno	NO	1	MAST	1	
EA101	EBOOL	%Q0.3.15			Ninguno	NO	1	MAST	1	
EA102	EBOOL	%Q0.3.12			Ninguno	NO	1	MAST	1	
EA103	EBOOL	%Q0.3.9			Ninguno	NO	1	MAST	1	
EA104	EBOOL	%Q0.3.8			Ninguno	NO	1	MAST	1	
EC101	EBOOL	%Q0.3.3			Ninguno	NO	1	MAST	1	
EC102	EBOOL	%Q0.3.14			Ninguno	NO	1	MAST	1	
EC103	EBOOL	%Q0.3.4			Ninguno	NO	1	MAST	1	
EC301	EBOOL	%Q0.3.6			Ninguno	NO	1	MAST	1	
EC302	EBOOL	%Q0.3.10			Ninguno	NO	1	MAST	1	
EC303	EBOOL	%Q0.3.11			Ninguno	NO	1	MAST	1	
EC304	EBOOL	%Q0.3.13			Ninguno	NO	1	MAST	1	
EC305	EBOOL	%Q0.3.1			Ninguno	NO	1	MAST	1	
EC401	EBOOL	%Q0.3.16			Ninguno	NO	1	MAST	1	
EC402	EBOOL	%Q0.3.7			Ninguno	NO	1	MAST	1	
ENCENM ICRO	EBOOL	%M3			Ninguno	NO	1	MAST	1	
ENCENM OTOR	EBOOL	%M4			Ninguno	NO	1	MAST	1	
Entrada	INT	%IW0.4.1				NO	1	MAST	1	
error	REAL					NO	1	MAST	1	
EV101	EBOOL	%Q0.3.2			Ninguno	NO	1	MAST	1	
EV102	EBOOL	%Q0.3.0			Ninguno	NO	1	MAST	1	
EV103	EBOOL	%Q0.3.5			Ninguno	NO	1	MAST	1	
HCS1	INT	%MW48				NO	1	MAST	1	
HCS2	INT	%MW52				NO	1	MAST	1	
HCS3	INT	%MW56				NO	1	MAST	1	
HCS4	INT	%MW60				NO	1	MAST	1	
HMI	BOOL			ACTIVA R HMI	Ninguno	NO	1	MAST	1	
Indicadores	ARRAY[1..4] OF EBOOL	%M111				NO	1	MAST	1	
Indicadores[1]	EBOOL	%M111				NO	0		0	

Variables

Nombre	Tipo	Dirección	Valor	Comentario	Marcas de	Derechos	Intercamb	Tarea	Retener	
Indicadores[2]	EBOOL	%M112				NO	0		0	
Indicadores[3]	EBOOL	%M113				NO	0		0	
Indicadores[4]	EBOOL	%M114				NO	0		0	
Inicio	EBOOL	%M1			Ninguno	NO	1	MAST	1	
IOT	BOOL			ACTIVAR IOT	Ninguno	NO	1	MAST	1	
IOTEAI01	EBOOL	%M151			Ninguno	NO	1	MAST	1	
IOTEAI03	EBOOL	%M160			Ninguno	NO	1	MAST	1	
IOTEAI04	EBOOL	%M159			Ninguno	NO	1	MAST	1	
IOTEC101	EBOOL	%M150			Ninguno	NO	1	MAST	1	
IOTEC102	EBOOL	%M154			Ninguno	NO	1	MAST	1	
IOTEC103	EBOOL	%M156			Ninguno	NO	1	MAST	1	
IOTEC301	EBOOL	%M155			Ninguno	NO	1	MAST	1	
IOTEC302	EBOOL	%M161			Ninguno	NO	1	MAST	1	
IOTEC305	EBOOL	%M158			Ninguno	NO	1	MAST	1	
IOTEC402	EBOOL	%M157			Ninguno	NO	1	MAST	1	
IOTEV102	EBOOL	%M162			Ninguno	NO	1	MAST	1	
IOTEV103	EBOOL	%M163			Ninguno	NO	1	MAST	1	
IOTMOTOR	EBOOL	%M152			Ninguno	NO	1	MAST	1	
IOTNE	EBOOL	%M153			Ninguno	NO	1	MAST	1	
J1	REAL					NO	1	MAST	1	
jint	REAL	%MW40				NO	1	MAST	1	
JORDAN	INT	%QW0.50				NO	1	MAST	1	
JORDANPARA	Para SC ALING					NO	1	MAST	1	
in min	REAL		0.0			NO	0		0	
in max	REAL		100.0			NO	0		0	
out min	REAL		0.01			NO	0		0	
out max	REAL		10000.0			NO	0		0	
clip	BOOL					NO	0		0	
JReal	REAL	%MW26				NO	1	MAST	1	
M1	BOOL				Ninguno	NO	1	MAST	1	
M2	BOOL				Ninguno	NO	1	MAST	1	
M3	BOOL				Ninguno	NO	1	MAST	1	
MANAUTO	BOOL				Ninguno	NO	1	MAST	1	

Variables

Nombre	Tipo	Dirección	Valor	Comentario	Marcas de	Derechos	Intercamb	Tarea	Retener	
MICRO	EBOOL	%Q0.3.18		MICRO MASTER	Ninguno	NO	1	MAST	1	
MICROVV	INT		2408			NO	1	MAST	1	
MOTOR	EBOOL	%Q0.3.20			Ninguno	NO	1	MAST	1	
NJORDAN	INT	%MW2				NO	1	MAST	1	
NUMJORDAN	REAL	%MW20				NO	1	MAST	1	
NUMSENSOR	REAL	%MW22				NO	1	MAST	1	
PARAMETROS	Para PIDFF					NO	1	MAST	1	
id	UINT					NO	0		0	
pv inf	REAL		0.0			NO	0		0	
pv sup	REAL		100.0			NO	0		0	
out inf	REAL		0.0			NO	0		0	
out sup	REAL		100.0			NO	0		0	
rev dir	BOOL		0			NO	0		0	
mix par	BOOL					NO	0		0	
aw type	BOOL					NO	0		0	
en rcpy	BOOL					NO	0		0	
kp	REAL		1.594			NO	0		0	
ti	TIME		T#243.5075s			NO	0		0	
td	TIME		T#0s			NO	0		0	
kd	REAL		0.0			NO	0		0	
pv dev	BOOL					NO	0		0	
bump	BOOL					NO	0		0	
dband	REAL					NO	0		0	
gain kp	REAL					NO	0		0	
ovs att	REAL					NO	0		0	
outbias	REAL					NO	0		0	
out min	REAL		0.0			NO	0		0	
out max	REAL		100.0			NO	0		0	
outrate	REAL					NO	0		0	
ff inf	REAL					NO	0		0	
ff sup	REAL					NO	0		0	
otff inf	REAL					NO	0		0	
otff sup	REAL					NO	0		0	
Paro	EBOOL	%M2			Ninguno	NO	1	MAST	1	
ParoEmer	EBOOL	%I0.2.14			Ninguno	NO	1	MAST	1	
PIDIOT	EBOOL	%M60			Ninguno	NO	1	MAST	1	
Referencia	REAL	%MW64	40.0			NO	1	MAST	1	
ResetPID	EBOOL	%M180			Ninguno	NO	1	MAST	1	
Revisar	BOOL				Ninguno	NO	1	MAST	1	
RTI	REAL					NO	1	MAST	1	
s	INT					NO	1	MAST	1	
Salida	REAL					NO	1	MAST	1	

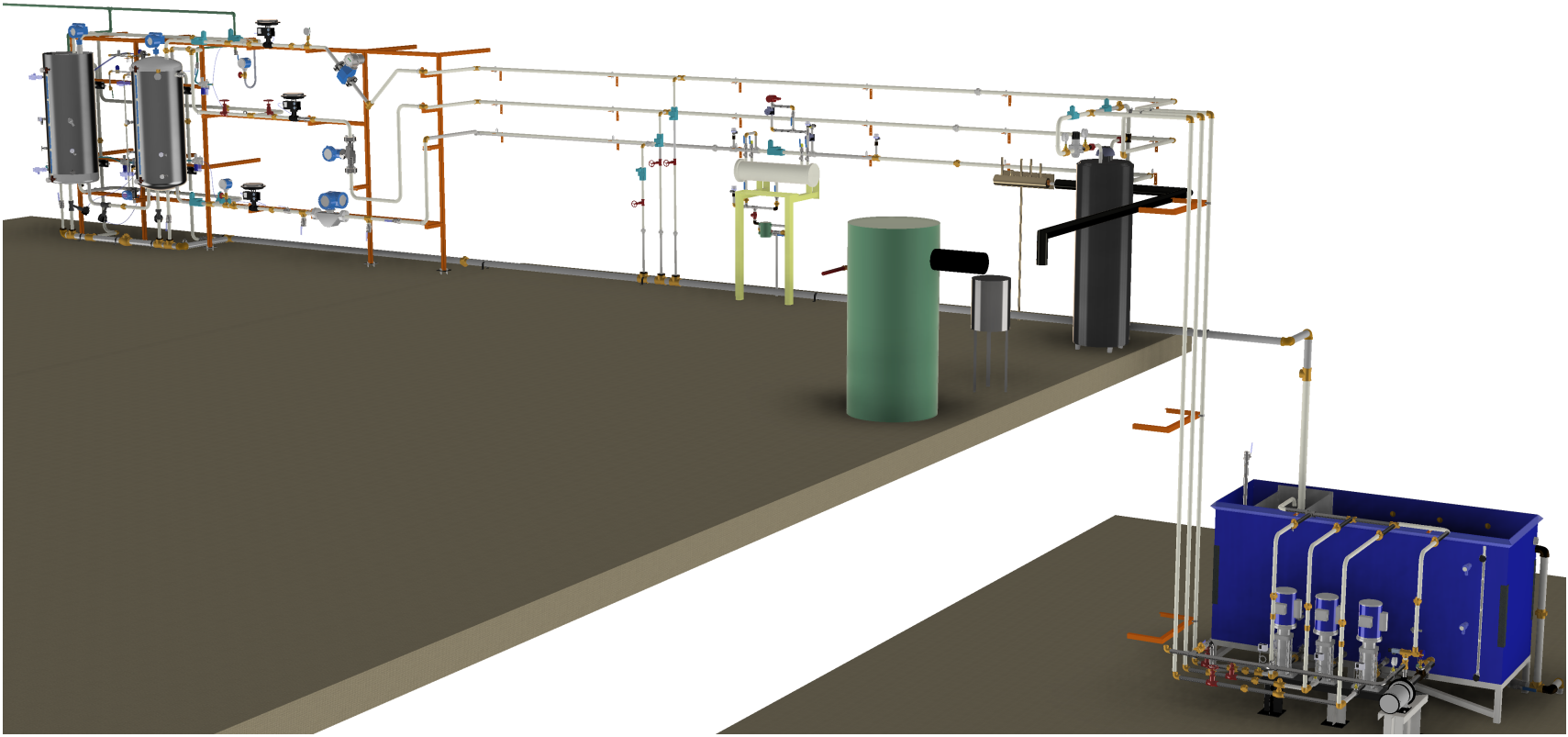
Variables

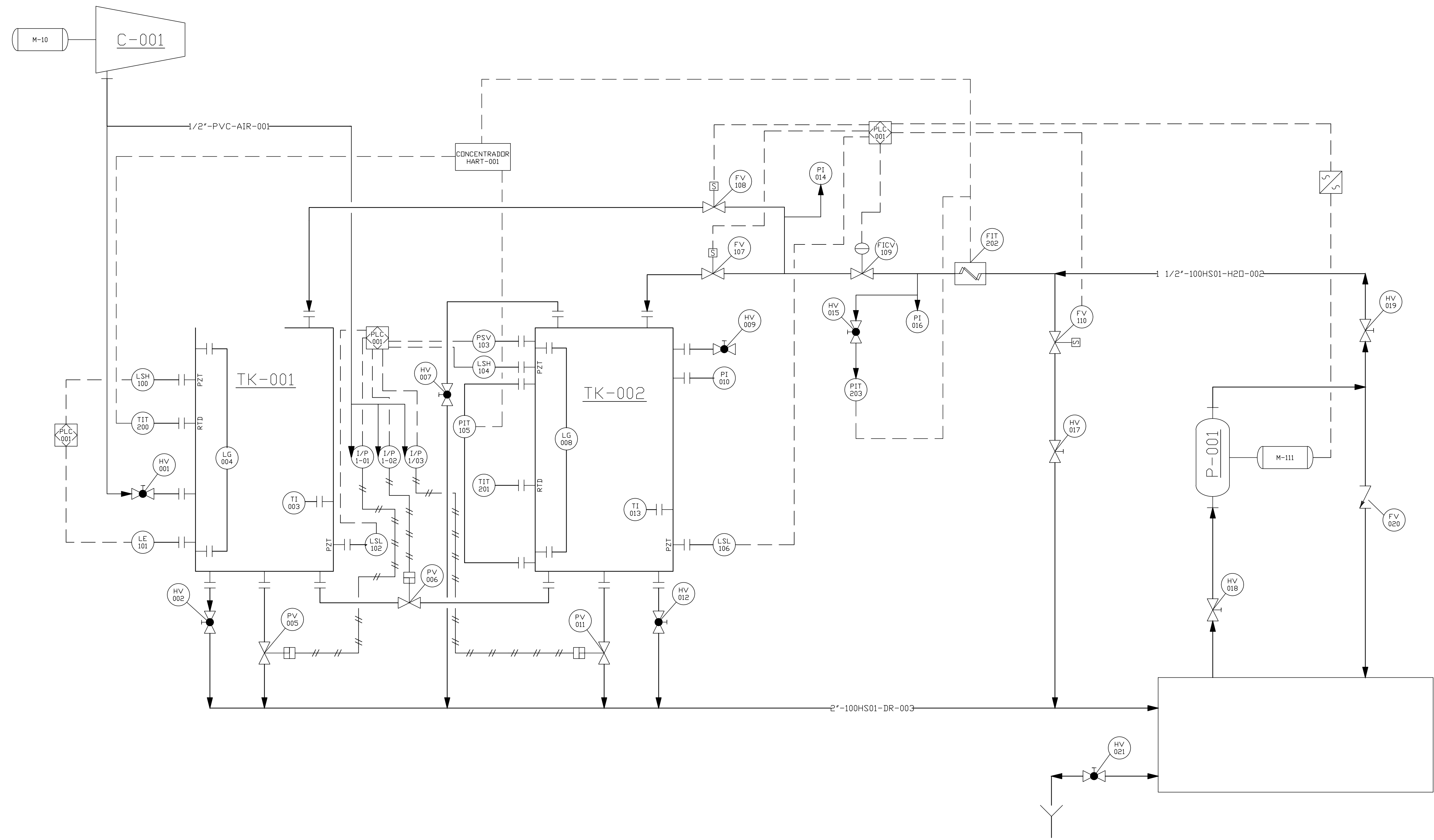
Nombre	Tipo	Dirección	Valor	Comentario	Marcas de	Derechos	Intercamb	Tarea	Retener	
SensorIOT	INT	%MW44				NO	1	MAST	1	
SENSORNIVEL	INT					NO	1	MAST	1	
SENSORPARA	Para_SC ALING					NO	1	MAST	1	
in_min	REAL		0.0			NO	0		0	
in_max	REAL		10000.0			NO	0		0	
out_min	REAL		0.0			NO	0		0	
out_max	REAL		104.0			NO	0		0	
clip	BOOL					NO	0		0	
VALOR SENSOR	REAL	%MW10				NO	1	MAST	1	
Variable Ejemplo	INT	%MW1	541			NO	1	MAST	1	
VEA101	EBOOL	%M109			Ninguno	NO	1	MAST	1	
VEA103	EBOOL	%M107			Ninguno	NO	1	MAST	1	
VEA104	EBOOL	%M106			Ninguno	NO	1	MAST	1	
VEC101	EBOOL	%M102			Ninguno	NO	1	MAST	1	
VEC102	EBOOL	%M100			Ninguno	NO	1	MAST	1	
VEC103	EBOOL	%M103			Ninguno	NO	1	MAST	1	
VEC301	EBOOL	%M101			Ninguno	NO	1	MAST	1	
VEC302	EBOOL	%M108			Ninguno	NO	1	MAST	1	
VEC305	EBOOL	%M104			Ninguno	NO	1	MAST	1	
VEC402	EBOOL	%M105			Ninguno	NO	1	MAST	1	
VEV101	EBOOL	%M110			Ninguno	NO	1	MAST	1	
VEV102	EBOOL	%M115			Ninguno	NO	1	MAST	1	
VEV103	EBOOL	%M116			Ninguno	NO	1	MAST	1	
VMICRO	INT	%QW0.5 .6				NO	1	MAST	1	

ANEXO E

Planta A del laboratorio de LACTI

P&ID de la línea utilizada en el proyecto de titulación (Protocolo de comunicación Hart)





Dibujado	Nombre	Fecha	Ingeniería Mecatrónica	UNIVERSIDAD POLITÉCNICA SALESIANA
Comprobado	RICARDO TORRES	06/04/2022	SEDE MATRIZ CUENCA	
Norma ISA 5.1	DISEÑO DE PLANTA			Lámina N° 01