

**INSTITUTO TECNOLÓGICO SUPERIOR AERONÁUTICO**

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &  
AVIÓNICA**

**“CONTROL Y MONITOREO DEL NIVEL DE TEMPERATURA DE LA  
ESTACIÓN DE NIVEL USANDO UN PLC TWIDO”**

**POR:**

**Ortiz Chicaiza Gerardo Javier**

**Trabajo de Graduación como requisito previo para la obtención del  
Título de:**

**TECNÓLOGO EN ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &  
AVIÓNICA**

**2013**

## **CERTIFICACIÓN**

Certifico que el presente Trabajo de Graduación fue realizado en su totalidad por el Sr. **ORTIZ CHICAIZA GERARDO JAVIER**, como requerimiento parcial para la obtención del título de **TECNÓLOGO EN ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN & AVIÓNICA**

---

**ING. LUCÍA GUERRERO**  
**DIRECTORA DEL PROYECTO**

Latacunga, Marzo del 2013

## **DEDICATORIA**

Mientras más grande es el sacrificio, más grande será la recompensa; es por ello que dedico este trabajo de grado a mis padres, que son un pilar muy importante en mi desarrollo personal y profesional ya que gracias a ellos pude cumplir mi meta; gracias a su apoyo y la confianza que depositaron en mi, a pesar de las alegrías y tristezas que se presentan en la vida, ellos han sabido sobresalir y brindarme la guía necesaria para resolver mis problemas.

A mis hermanos que siempre me apoyaron, confiaron en mí, siempre dándome ánimos en los momentos más difíciles por los que vivimos. Junto a ellos hemos sobrelidado las situaciones más difíciles y lo hemos logrado por haber luchado como hermanos siempre unidos, espero que eso nunca cambie entre nosotros

Y a todas las personas que de una u otra manera me ayudaron con un granito de arena muy valioso que me permitió culminar con mi tesis. Un pequeño paso en el largo camino que tengo por recorrer.

Gracias a todos de corazón.

Javier Ortiz Ch.

## **AGRADECIMIENTO**

Agradezco a Dios que me ha dado un día más de vida, por haberme regalado a los mejores padres del mundo, unos hermanos incondicionales y una familia muy unida.

Agradezco a mis padres por su apoyo moral y económico, porque cada día me ayudan a ser mejor persona dándome con sus consejos. Gracias por tenerme paciencia; porque tenerme como hijo no ha sido fácil y sé el enorme sacrificio que han hecho por ayudarme a cumplir mis metas.

A mis hermanos que siempre me brindan su apoyo y me alegran en los momentos más difíciles y tristes.

Gracias a mi familia que siempre me ayudó con una granito de arena.

A mis amigos de la carrera que gracias a ellos, mi estadía en esta institución fue muy amena y divertida.

A mis profesores, quienes me ayudaron y me apoyaron en cualquier inconveniente que se me haya presentado. Quienes conocen mis virtudes y defectos.

A todos

Gracias de corazón.

Javier Ortiz Ch.

# ÍNDICE DE CONTENIDOS

PORTADA.....	I
CERTIFICACIÓN.....	II
DEDICATORIA .....	III
AGRADECIMIENTO .....	IV

## CAPÍTULO I

1.1 ANTECEDENTES.....	1
1.2 OBJETIVOS .....	2
1.2.1 GENERAL.....	2
1.2.2 ESPECÍFICOS .....	2
1.3 JUSTIFICACIÓN E IMPORTANCIA .....	2
1.4 ALCANCE.....	3

## CAPÍTULO II

### MARCO TEÓRICO

2.1 INTRODUCCIÓN.....	4
2.2 HARDWARE TWIDO TWDLCAA24DRF .....	4
2.2.1 Descripción de los componentes del TWIDO TWDLCAA24DRF. ....	5
2.2.2 Partes Principales del PLC TWIDO TWDLCAA24DRF. ....	5
2.2.2.1 Conector de ampliación. ....	5
2.2.2.2 Terminales de potencia del sensor. ....	6
2.2.2.3 Puerto serie. ....	6
2.2.2.4 Potenciómetros analógicos.....	6
2.2.2.5 Terminales de fuente de alimentación.....	6
2.2.2.6 Terminales de entrada.....	7
2.2.2.7 Indicadores LED. ....	7
2.2.2.8 Terminales de salida.....	7
2.3 Módulo de ampliación.....	7
2.3.1 Módulos de E/S digitales .....	8
2.3.2 Módulos de E/S analógicas .....	8
2.3.2.1 Por el número de entradas/salidas .....	9
2.3.2.2 Por el tipo de señal analógica.....	9

2.4 Cable de comunicación TSX PCX1031 .....	10
2.5 Software TwidoSuite.....	10
2.5.1 Requisitos mínimos y recomendados .....	12
2.6 Edición de datos. ....	12
2.7 Ventana para la creación de símbolos.....	13
2.8 Tipos de variables básicas .....	13
2.8.1 Los Objetos bit.....	14
2.8.1.1 Tipos de objetos de bit.....	14
2.8.1.2 Bits internos.....	14
2.8.1.3 Bits extraídos de palabras. ....	15
2.8.2 Tipos de objetos de palabra .....	15
2.8.2.1 E/S analógicas.....	15
2.8.3 Memoria interna.....	16
2.9 Escribir Programa .....	16
2.9.1 Lenguaje Ladder.....	16
2.9.1 Instrucciones de Lista .....	17
2.9.2 GRAFCET (Gráfica de Control de Secuencias de Programación) .....	17
2.10 Abrir el editor de programa .....	18
2.10.1 Ventana del editor de programa .....	19
2.10.2 Navegador de datos .....	20
2.10.11 Creación de un programa en lenguaje Ladder .....	20
2.10.12 Barra de Herramientas Ladder .....	20
2.11 Transferencia del programa.....	21
2.12 Operaciones con señales analógicas .....	22
2.12.1 Potenciómetros analógicos.....	22
2.13 Módulos analógicos.....	22
2.13.1 Funcionamiento de los módulos analógicos .....	22
2.14 OPC Server. ....	23
2.15 LabVIEW. ....	23
2.16 COMUNICACIÓN INDUSTRIAL.....	24
2.17 PROTOCOLOS DE COMUNICACIÓN .....	24
2.18 MODBUS.....	24
2.18.1 Principales Características .....	25
2.18.2 Formato General de las Tramas.....	25

## CAPÍTULO III

### DESARROLLO DEL TEMA

3.1 PRELIMINARES.....	26
3.1.1 INSTALACIÓN DE EQUIPOS. ....	26
3.2 PLC Twido TWDLCAA24DFR .....	27
3.2.1 Módulo de expansión de 2 entradas y 1 salida analógica TM2AMM3HT. ....	27
3.2.2 Fuente de 24VDC. ....	28
3.3 Programación del PLC Twido. ....	29
3.3.1 INSTALACIÓN DEL SOFTWARE DE PROGRAMACIÓN TWIDO SUITE. ....	29
3.3.2 CREAR UN NUEVO PROYECTO. ....	33
3.3.3 DESARROLLO DEL PROGRAMA PRINCIPAL. ....	38
3.3.4 Bloques de operación. ....	41
3.4 DESARROLLO DEL PROGRAMA CON EL SOFTWARE OPC SERVER. ....	51
3.3.4 DESARROLLO DEL PROGRAMA EN EL SOFTWARE LABVIEW. ....	64
3.3.5 INSTRUCTIVO DE FUNCIONAMIENTO. ....	69
3.3.5.1 Voltajes de cada elemento. ....	69
3.3.5.2 Voltajes de los sensores de la estación de nivel. ....	69
3.3.5.3 Prueba de Software. ....	71
3.3.5.4 Conexión de equipos. ....	71
3.3.6 GASTOS REALIZADOS .....	85
3.3.6.1 COSTOS PRIMARIOS .....	85
3.3.6.2 COSTOS SECUNDARIOS.....	86
3.3.6.3 COSTO TOTAL.....	87

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES.....	88
4.2 RECOMENDACIONES. ....	89

## ÍNDICE DE TABLAS

Tabla 3.1 Sintaxis en Modbus.....	68
Tabla 3.2 Voltajes.....	69
Tabla 3.3 Voltajes y números de bits del sensor de nivel.....	70
Tabla 3.4 Voltajes y numero de bits del sensor de temperatura.....	70
Tabla 3.5 Costos primarios.....	86
Tabla 3.6 Costos secundarios.....	86
Tabla 3.7 Costo total.....	87



## ÍNDICE DE FIGURAS.

Figura 2.1 Hardware Twido TWDLCAA24DRF.....	4
Figura 2.2 Partes del Twido TWDLCAA24DRF.....	5
Figura 2.3 Conexiones del Twido TWDLCAA24DRF.....	7
Figura 2.4 Gama de módulos de salidas digitales.....	8
Figura 2.5 Gama de módulos de salidas analógicas.....	9
Figura 2.6 Cable TSX PCX1031.....	10
Figura 2.7 Plataforma Twido Suite.....	11
Figura 2.8 Posible cableado externo.....	12
Figura 2.9 Tabla de símbolos.....	13
Figura 2.10 Bits de entrada y salida.....	14
Figura 2.11 Bits del sistema.....	15
Figura 2.12 Bits de palabra.....	15
Figura 2.13 Entradas y Salidas Analógicas.....	15
Figura 2.14 Palabras de sistema.....	16
Figura 2.15 Instrucciones de lista.....	17
Figura 2.16 Ejemplo de un Programa GRAFCET.....	18
Figura 2.17 Editor del Programa.....	19
Figura 2.18 Barra de Herramientas Ladder.....	20
Figura 2.19 Transferencia del programa.....	21
Figura 2.20 Trama Modbus.....	25
Figura 3.1 Montaje del PLC.....	27
Fig. 3.2 Módulo de entradas y salidas analógicas.....	27
Fig. 3.3 Fuente de alimentación de 24VDC.....	28
Fig. 3.4 Circuito de la fuente de 24VDC.....	28
Figura 3.5 Ícono del archivo ejecutable.....	29
Figura 3.6 Ícono Setup.....	30
Figura 3.7 Inicio del proceso de instalación.....	30
Figura 3.8 Contrato de licencia.....	31
Figura 3.9 Nombre del organizador.....	31
Figura 3.10 Ruta de instalación del programa.....	31
Figura 3.11 Tipo de instalación.....	32
Figura 3.12 Seleccionar carpeta de programas.....	32
Figura 3.13 Opciones de apertura de TwidoSuite.....	33

Figura 3.14	Pantalla inicial de TwidoSuite. ....	33
Figura 3.15	Ventana de la pestaña de “Proyecto”. ....	34
Figura 3.16	Ícono describir. ....	35
Figura 3.17	Ventana de configuración de Hardware. ....	35
Figura 3.18	Selección de módulo de ampliación. ....	36
Figura 3.19	Configuración del módulo de ampliación. ....	36
Figura 3.20	Cuadro de programación. ....	37
Figura 3.21	Cuadro de comunicación. ....	37
Figura 3.22	Cuadro de programación. ....	38
Figura 3.23	Ingreso de variable analógica. ....	41
Figura 3.24	Operaciones aritméticas. ....	41
Figura 3.25	Operaciones aritméticas. ....	42
Figura 3.26	Operaciones aritméticas. ....	42
Figura 3.27	Variable de nivel. ....	43
Figura 3.28	Escalamiento de temperatura. ....	43
Figura 3.29	Inicio. ....	44
Figura 3.30	Encendido de la bomba. ....	45
Figura 3.31	Llenado del tanque. ....	45
Figura 3.32	Encendido de la niquelina. ....	46
Figura 3.33	Encendido de la niquelina. ....	46
Figura 3.34	Cuadro de comparación. ....	46
Figura 3.35	Mezcladora. ....	47
Figura 3.36	Tiempo de mezclado. ....	47
Figura 3.37	Temporizador. ....	48
Figura 3.38	Vaciado. ....	48
Figura 3.39	Encendido de la válvula. ....	49
Figura 3.40	Cuadro de comparación para el vaciado. ....	49
Figura 3.41	Operación cíclica del proceso. ....	49
Figura 3.42	Contador. ....	50
Figura 3. 43	Tabla de símbolos. ....	50
Figura 3.44	Menú inicio. ....	51
Figura 3.45	Crear nuevo Canal. ....	52
Figura 3.46	Identificación. ....	52
Figura 3.47	Protocolo de comunicación. ....	53

Figura 3.48 Configuración de la comunicación.....	53
Figura 3.49 Escribir Optimizaciones.....	54
Figura 3.50 Resumen de todas las características del canal generado.....	54
Figura 3.51 Nuevo dispositivo.....	55
Figura 3.52 Protocolo de comunicación.....	55
Figura 3.53 Protocolo de comunicación.....	56
Figura 3.54 Parámetros del tiempo en la comunicación.....	56
Figura 3.55 Auto-Negación.....	57
Figura 3.56 Creación de una base de datos.....	57
Figura 3.57 Configuración de los datos de acceso.....	58
Figura 3.58 Configuración de los datos a usar.....	58
Figura 3.59 Tipos de Bloques.....	59
Figura 3.60 Configuración para Importar una Variable.....	59
Figura 3.61 Ideando.....	60
Figura 3.62 Tratamiento de Errores.....	60
Figura 3.63 Resumen de la configuración del dispositivo.....	61
Figura 3.64 Propiedades de los Tags o etiquetas.....	61
Figura 3.65 Establecer comunicación.....	62
Figura 3.66 Cuadro de comunicación.....	62
Figura 3.67 Cuadro de información de la comunicación.....	63
Figura 3.68 Cuadro de inicio.....	63
Figura 3.69 Aviso de ejecución del programa.....	63
Figura 3.70 LabVIEW Front Panel.....	64
Figura 3.71 LabVIEW Block Diagram.....	65
Figura 3.72 Propiedades del control.....	66
Figura 3.73 DSTP Server.....	66
Figura 3.74 Buscar la dirección de almacenamiento.....	67
Figura 3.75 Seleccionar la variable que corresponde al control.....	67
Figura 3.76 Verificar la dirección y características.....	68
Figura 3.77 Swich ON/OFF.....	72
Figura 3.78 Entradas digitales.....	72
Figura 3.79 Terminales de alimentación.....	73
Figura 3.80 Ranura de expansión.....	73
Figura 3.81 Conexión L y N.....	74

Figura 3.82 Conexión COM 0. ....	74
Figura 3.83 Conexión Bomba. ....	75
Figura 3.84 Conexión níquelina. ....	75
Figura 3.85 Conexión Com 1. ....	76
Figura 3.86 Conexión válvula. ....	76
Figura 3.87 Módulo E/S analógicas. ....	77
Figura 3.88 Conexión señal de nivel. ....	77
Figura 3.89 Conexión señal de temperatura. ....	78
Figura 3.90 Alimentación del módulo. ....	78
Figura 3.91 Comunicación. ....	79
Figura 3.92 Interruptor ON/OFF. ....	79
Figura 3.93 Modo Programación. ....	79
Figura 3.94 Abrir un nuevo proyecto. ....	80
Figura 3.95 Bloques de programación. ....	80
Figura 3.96 Ventana de comunicación. ....	81
Figura 3.97 Ventana de ejecución. ....	81
Figura 3.98 Bloques de programación. ....	82
Figura 3.99 Desconectar. ....	82
Figura 3.100 Abrir programa. ....	83
Figura 3.101 Play. ....	83
Figura 3.102 Inicio. ....	84
Figura 3.103 Límite máximo. ....	84
Figura 3.104 Reset. ....	85

## ÍNDICE DE ANEXOS

ANEXO A Desarrollo del programa en Twido Suite.

ANEXO B Desarrollo del programa en LabVIEW.

ANEXO C Anteproyecto.

## RESUMEN.

Para el desarrollo del presente proyecto de grado se realizó la adquisición de un PLC Twido TWDLCAA24DRF, el mismo que apareció recientemente en el mercado local, fue analizado y estudiado previamente para comprender su funcionamiento, sus aplicaciones, las ventajas que posee en comparación a otros PLC's que existen en el mercado y cómo ayudará en el mejoramiento del aprendizaje de los alumnos de la carrera de Electrónica Mención Instrumentación y Aviónica. Además se realizó un ejemplo práctico donde se realiza una simulación de un proceso industrial, el cual consiste en el precalentamiento del agua y los aditivos que deben adicionarse previo al ingreso a un caldero; donde se usarán varias de las aplicaciones que posee el PLC Twido TWDLCAA24DRF.

En este proyecto se realizó la adquisición de las señales de los sensores de temperatura y de nivel en tiempo real, usando un módulo de entradas y salidas analógicas que se acopló al PLC.

Se realizó un acondicionamiento de las señales analógicas. La señal de nivel es una señal de corriente de 4-20mA, se realizó una conversión a una señal estándar de voltaje de 0-5V para su análisis. La señal de temperatura se encuentra en un rango estándar de 0-5V. En el software Twido Suite se realizó un escalamiento para poder visualizar los estados de las variables controladas en LabVIEW.

Para la comunicación entre LabVIEW y el PLC se usa el software OPC Server que es una interfaz de comunicación que permite direccionar variables.

Para el control, monitoreo y visualización de las variables adquiridas se usa el software LabVIEW que es una interfaz gráfica.

## **ABSTRACT.**

For the development of this project grade was acquiring a Twido PLC TWDLCAA24DRF, the same that appeared recently in the local market, was previously analyzed and studied to understand how it works, its applications, it has the advantages compared to other PLC 's that are on the market and how it will help in improving student learning of career Mention Electronic Instrumentation and Avionics. We also carried out a practical example where a simulation of an industrial process, which consists of water heating and additives should be added prior to entering a pot, where we will use several applications that owns the Twido PLC TWDLCAA24DRF .

This project was conducted in acquiring signals from the temperature sensors in real time level, using a module inputs and outputs coupled to the PLC.

We performed a conditioning of the analog signals. The level signal is a current signal of 4-20mA, we made a conversion to a standard signal of 0-5V voltage for analysis. The temperature signal is at a standard range of 0-5V. The Twido Suite software scaling was performed to visualize the states of the controlled variables in LabVIEW.

For communication between LabVIEW and the PLC using the OPC Server software is a communication interface that allows to address variables.

To control, monitor and display the acquired variables using the LabVIEW software is a graphical interface.

# CAPÍTULO I

## 1.1. ANTECEDENTES

La incorporación de un computador para el monitoreo de un proceso industrial ha sido una de las aplicaciones más usadas en el ámbito industrial debido a que dichos dispositivos simplemente no son usados para la visualización de la información que proporciona un autómata programable, pueden resolver algoritmos de control muy complejos, de esta manera se han convertido en un sistema que puede analizar una determinada falla y la puede resolver de acuerdo a su propio criterio.

Es por eso que se necesitan autómatas programables que se puedan acoplar a esta evolución, que sean mucho más veloces y que puedan comunicarse en el mismo idioma que usa la computadora es decir digitalmente.

El PLC Twido es un autómata programable capaz de controlar señales digitales y analógicas por lo que es un dispositivo muy versátil, además su programación es muy sencilla por lo que hace a este dispositivo muy didáctico en el ámbito educativo.

En el Instituto Tecnológico Superior Aeronáutico no se contaba con este tipo de dispositivos, además al realizar la investigación se encontró que nunca antes se ejecutó un proyecto con estas características; haciéndose indispensable la adquisición del mismo para mejorar los conocimientos de los estudiantes.



## **1.2. OBJETIVOS**

### **1.2.1. GENERAL**

- Implementar un sistema para el control y monitoreo de una estación de nivel usando el PLC Twido y de esta manera mejorar el nivel de conocimientos de los alumnos de la carrera de Electrónica Instrumentación y Aviónica del Instituto Tecnológico Superior Aeronáutico.

### **1.2.2. ESPECÍFICOS**

- ✓ Buscar información referente al PLC Twido que proporcione información técnica, bibliográfica para un correcto desempeño del trabajo de graduación.
- ✓ Desarrollar el programa que permita el control y monitoreo de la estación de nivel por medio del PLC Twido.
- ✓ Realizar las pruebas de funcionamiento para un correcto desempeño del sistema de control y monitoreo, y establecer las conclusiones y recomendaciones correspondientes.

## **1.3. JUSTIFICACIÓN E IMPORTANCIA**

El crecimiento industrial es cada vez más evidente y conlleva a la utilización de equipos que permitan realizar efectivamente el proceso de producción. Es por esto que se desarrolló la automatización de procesos industriales permitiendo realizar un proceso automático, reduciendo errores humanos y mejorando la calidad en el proceso.

El Instituto Tecnológico Superior Aeronáutico no cuenta con módulos en el laboratorio de Instrumentación Virtual que fortalezcan los conocimientos acerca de la automatización por lo que el sistema de control y monitoreo de la estación de nivel ayudará a los tecnólogos en Electrónica a desempeñarse satisfactoriamente en su ámbito laboral.

Al incorporar un PLC Twido e implementarlo en el control de un proceso que con seguridad se lo encontrará en la industria, los estudiantes podrán familiarizarse con nuevas tecnologías las mismas que están en pleno auge en el sector industrial.

#### **1.4. ALCANCE**

Este proyecto se enfoca a la implementación de un PLC Twido para el control y monitoreo de variables físicas de una estación de nivel que simula un proceso industrial estudiando la forma de programación y conexiones del mismo. De esta manera los estudiantes de la carrera de Electrónica mención Instrumentación y Aviónica desarrollarán destrezas y habilidades en nuevas tecnologías las mismas que son indispensables estudiar y aplicar en el campo profesional.

## CAPÍTULO II

### MARCO TEÓRICO

#### 2.1 INTRODUCCIÓN

Un controlador lógico programable constituye una solución eficiente para los sistemas de control y se han convertido en un procesador especializado del control industrial, siendo cada vez más importante su estudio y aplicación, es por eso que se implementaron estas prácticas para el laboratorio de Instrumentación Virtual del Instituto Tecnológico Superior Aeronáutico donde, de forma didáctica, se ayuda al estudiante al aprendizaje teórico-práctico del PLC basado en el PLC TWIDO TWDLCAA24DRF.

#### 2.2 HARDWARE TWIDO TWDLCAA24DRF<sup>1</sup>

La elección de un modelo u otro de autómatas vendrá dada por la tipología y complejidad de la aplicación que se desea automatizar. En este caso usaremos un controlador compacto.



Figura 2.1 Hardware Twido TWDLCAA24DRF  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

1.-<http://www.ebah.com.br/content/ABAAABeEkAl/manual-twido>

## 2.2.1. Descripción de los componentes del TWIDO TWDLCAA24DRF.

El controlador Twido TWDLCAA24DRF compacto está formado por los siguientes componentes:

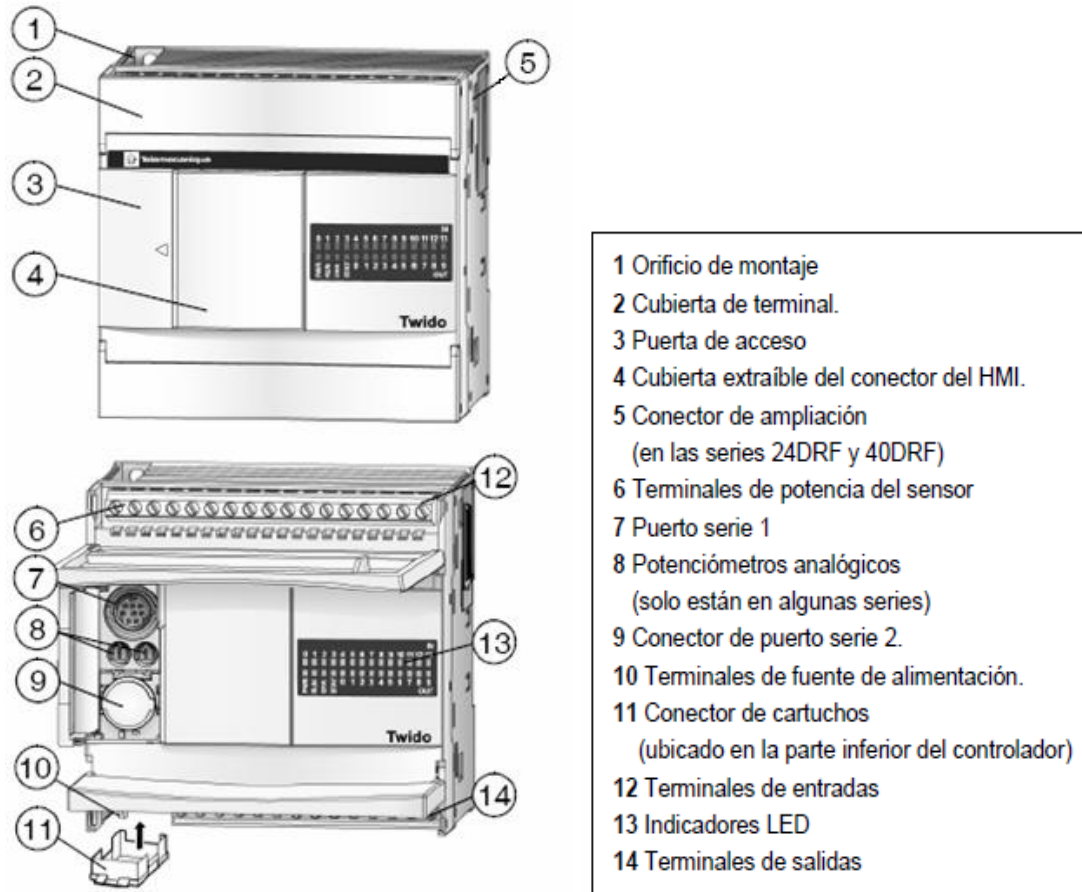


Figura 2.2 Partes del Twido TWDLCAA24DRF  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

## 2.2.2 Partes Principales del PLC TWIDO TWDLCAA24DRF.

### 2.2.2.1 Conector de ampliación.

Ranura en la cual se conecta un módulo de expansión de entradas y salidas analógicas.

### **2.2.2.2 Terminales de potencia del sensor.**

Terminales de 24V y GND mediante los cuales se pueden activar las entradas digitales del PLC.

### **2.2.2.3 Puerto serie.**

Puerto en el cual se conecta el cable de transmisión y recepción de datos y se puede programar al PLC.

### **2.2.2.4 Potenciómetros analógicos.**

Los controladores Twido cuentan con uno o dos potenciómetros analógicos. Los potenciómetros aportan unos valores numéricos, de 0 a 1.023 para el potenciómetro analógico 1 y de 0 a 511 para el potenciómetro analógico 2, correspondientes a los valores analógicos que indican estos potenciómetros, forman parte de las dos palabras de entrada siguientes:

- **%IW0.0.0** para el potenciómetro analógico 1 (a la izquierda)
- **%IW0.0.1** para el potenciómetro analógico 2 (a la derecha)

Estas palabras se pueden utilizar en operaciones aritméticas. Se pueden emplear para cualquier tipo de ajuste (preselección de un retardo o de un contador, ajuste de la frecuencia del generador de pulsos o de la duración del precalentamiento de una máquina, etc.).

### **2.2.2.5 Terminales de fuentes de alimentación**

Terminales en los cuales se conecta la fase y neutro 110VAC para alimentar al PLC.

### 2.2.2.6 Terminales de entrada

Terminales por los cuales ingresa una señal digital ya sea cero lógico (0 VDC) o uno lógico (24 VDC).

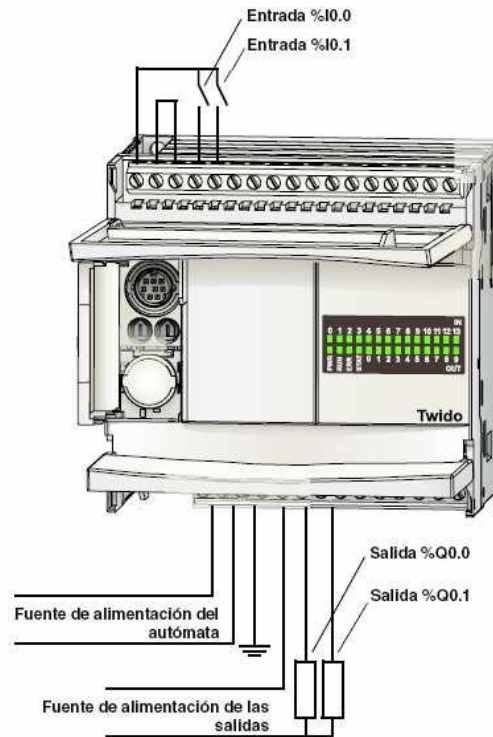


Figura 2.3 Conexiones del Twido TWDLCAA24DRF  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

### 2.2.2.7 Indicadores LED.

Son indicadores luminosos que se accionan cada vez que una salida digital se active.

### 2.2.2.8 Terminales de salida.

Este PLC funciona con relés, es decir cuando una salida se activa un contacto N/C se acciona.

## 2.3 Módulo de ampliación:

Los módulos de ampliación dan al PLC Twido la mayor adaptabilidad posible a las diferentes aplicaciones dentro de su campo de acción, pudiéndose ajustar el dispositivo

lo máximo posible a cada una de las aplicaciones concretas con el ahorro de coste que ello comporta.

### 2.3.1 Módulos de E/S digitales:

Existen una amplia gama de módulos distintos de entradas/salidas TON Twido que permiten completar las entradas/salidas integradas tanto en las bases compactas ampliables como en las bases modulares, pudiendo cada usuario adaptar el controlador a las necesidades de su aplicación, optimizando así los costes.



Figura 2.4 Gama de módulos de salidas digitales.  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

### 2.3.2 Módulos de E/S analógicas:

Los módulos de ampliación de entradas analógicas Twido permiten obtener diferentes valores analógicos presentes en las aplicaciones industriales. Los de salidas analógicas se utilizan para dirigir los preaccionadores en unidades físicas, como variadores de velocidad, válvulas y aplicaciones que requieran control de procesos.



Figura 2.5 Gama de módulos de salidas analógicas.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

Las entradas y salidas analógicas vienen definidas en función del valor analógico, puede ser la corriente o la tensión, incluso valores de resistencia para los casos de temperatura.

Los módulos de entradas/salidas analógicas, tendrán la conexión del tipo bornero extraíble. Los módulos de entradas/salidas analógicas se elegirán teniendo en cuenta los siguientes parámetros:

### 2.3.2.1 Por el número de entradas/salidas: 2, 4 y 8

### 2.3.2.2 Por el tipo de señal analógica:

- ✓ Señal de Tensión de 0...10 V
- ✓ Señal de corriente de 0...20 mA
- ✓ Señal de corriente de 4...20 mA
- ✓ Señal de entrada termopar del tipo K, J y T.
- ✓ Señal de entrada PTC del tipo Pt 100/1000, Ni100/1000 rango de temperatura entre 50...150 °C.

Existe un tipo de módulo mixto que presentan entradas y salidas analógicas en el mismo cuerpo.

Los módulos analógicos Twido ofrecen una resolución de 10 bits, 11 bits + signo y 12 bits, con conexión mediante bornero con tornillo desenchufable.

Es necesario instalar una alimentación externa de 24 Vcc para cada módulo analógico.



Los elementos electrónicos internos y las vías de entradas/salidas de todos los módulos de entradas/salidas analógicas están aislados mediante foto acoplador.

## 2.4 Cable de comunicación TSX PCX1031.

Este cable convierte las señales comprendidas entre EIA RS232 y EIA RS485.

El cable TSX PCX1031 incorpora un conmutador rotativo de cuatro posiciones para seleccionar distintos modos de funcionamiento. El conmutador designa las cuatro posiciones como "0-3" y el ajuste apropiado de TwidoSuite para el autómata Twido es la posición 2.

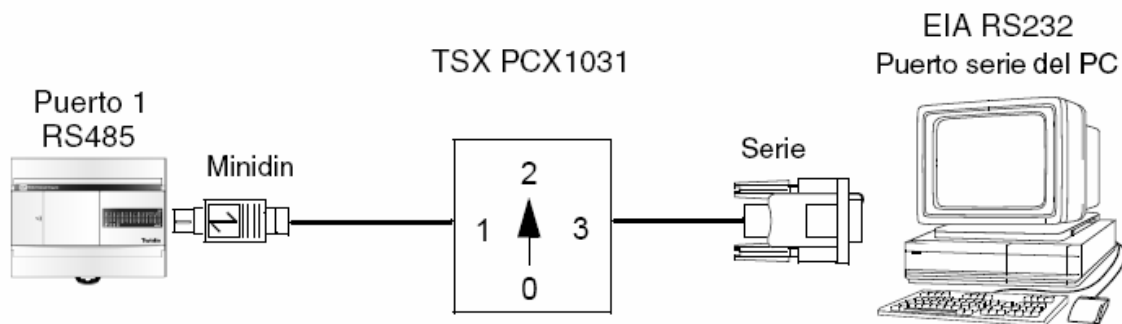


Figura 2.6 Cable TSX PCX1031.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

## 2.5 Software TwidoSuite

TwidoSuite es el primer software que está presente según el ciclo de desarrollo del proyecto. La navegación por el software es tan sencilla que se convierte en innata. TwidoSuite es un entorno de desarrollo gráfico, lleno de funciones para crear, configurar y mantener aplicaciones de automatización para los autómatas programables Twido de Telemecanique. TwidoSuite permite crear programas con distintos tipos de lenguaje, después de transferir la aplicación para que se ejecute en un autómata.



Figura 2.7 Plataforma Twido Suite.  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

TwidoSuite es un programa basado en Windows de 32 bits para un ordenador personal (PC) que se ejecuta en los sistemas operativos Microsoft Windows 2000 y XP Professional.

Las principales funciones del software TwidoSuite son:

- ✓ Interface de usuario intuitiva y orientada a proyectos.
- ✓ Diseño de software sin menús. Las tareas y funciones del paso
- ✓ Selección de un proyecto siempre se encuentra visible.
- ✓ Soporte de programación y configuración.
- ✓ Comunicación con el autómata.
- ✓ Ayuda de primera mano acerca del nivel de tareas que ofrece enlaces relevantes a la ayuda en línea.

TwidoSuite es un software fácil de usar que necesita poco o nada de aprendizaje. Este software tiene por objeto reducir de forma significativa el tiempo de desarrollo de los proyectos simplificando todas las intervenciones.

### 2.5.1 Requisitos mínimos y recomendados:

La configuración mínima necesaria para utilizar TwidoSuite es la siguiente:

- ✓ Se recomienda un equipo compatible con PC y procesador Pentium a 466 MHz o superior.
- ✓ 128 MB de RAM o más,
- ✓ 100 MB de espacio libre en el disco duro.
- ✓ Sistema operativo: Windows 2000 o Windows XP.

### 2.6 Edición de datos.

Antes de programar la aplicación, debe definir las especificaciones del cableado de las E/S de la aplicación.

Para ello se le asigna a cada una de las entradas y salidas físicas un símbolo que indique qué realiza y ayude después a la mejor comprensión del programa.

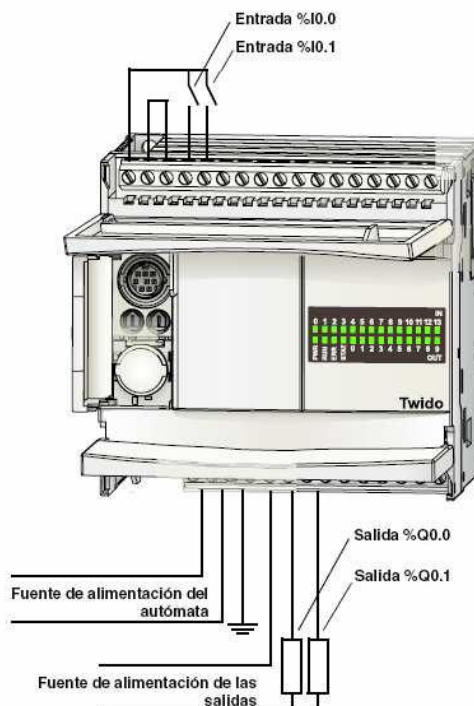


Figura 2.8 Posible cableado externo.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

## 2.7 Ventana para la creación de símbolos

En ese momento aparecerá en el área de trabajo la ventana para definir los diferentes símbolos que se utilizarán más tarde en la programación. Para crear un nuevo símbolo, se tendrá que pulsar en el icono de creación de símbolos. En ese momento aparecerá una nueva fila en el área inferior donde se rellenan los diferentes campos.

La ventana consta de cuatro columnas donde se pone el “Símbolo” (nombre descriptivo que aporte información de lo que realiza la señal), “Dirección” posición de memoria a la que está direccionada esa variable, “Comentario”.

Amplía la información descriptiva de la señal y “En uso” que indica si ese símbolo está siendo usado en el programa.

Para definir los símbolos de las E/S de la aplicación, se irá a la pestaña “Programar” de la barra de pasa de la aplicación, donde aparece directamente en la pestaña “Programa” de la barra de subpasos del programa y se tendrá que hacer clic en la opción “Definir símbolos” que es situada en la barra de tareas en la parte derecha de la ventana.



Figura 2.9 Tabla de símbolos.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

## 2.8 Tipos de variables básicas:

Una variable es una entidad de memoria de los tipos BOOL, WORD, INT, REAL...etc. Según la información que muestran se pueden definir diferentes tipos de objetos:

## 2.8.1 Los objetos bit.

Son variables binarias y como tal pueden ser consultadas por instrucciones booleanas. La información que suministran al autómata puede ser 0 ó 1 (falso ó verdadero).

### 2.8.1.1 Tipos de objetos de bit:

- **Bits de E/S:** Estos bits son las "imágenes lógicas" de los estados eléctricos de las E/S. Las entradas estarán direccionadas con la letra "%I" y las salidas "%Q".

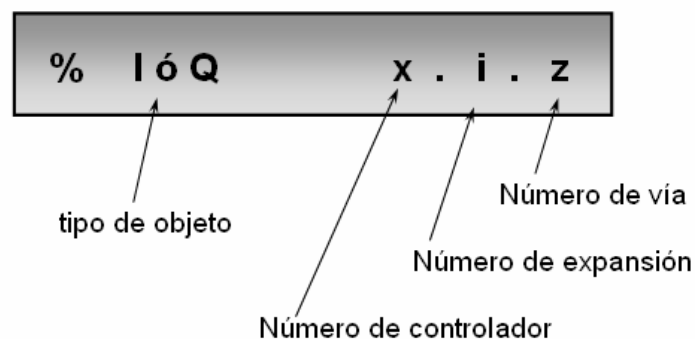


Figura 2.10 Bits de entrada y salida.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

**2.8.1.2 Bits internos.:** Los bits internos son áreas de memoria interna utilizadas para almacenar valores intermedios durante la ejecución de un programa. Los bits internos estarán diseccionados con la letra "%M".

- **Bits de sistema:** Los bits de sistema de %S0 a %S127 supervisan el funcionamiento correcto del autómata y la correcta ejecución del programa de aplicación. La letra para direccionar los bits de sistema será la "%S".

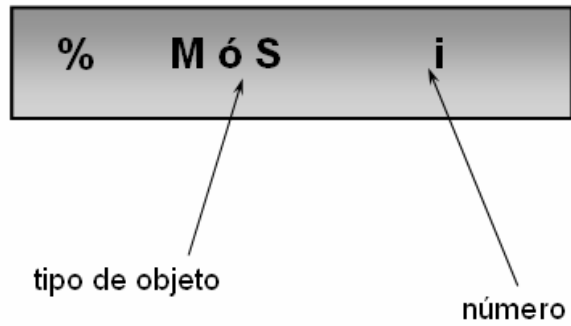


Figura 2.11 Bits del sistema.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemechanique-84695>

**2.8.1.3 Bits extraídos de palabras:** Uno de los 16 bits de algunas palabras puede extraerse como bit de operando.

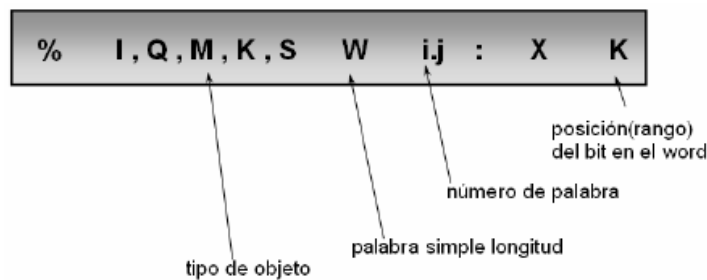


Figura 2.12 Bits de palabra.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemechanique-84695>

**2.8.2 Tipos de objetos de palabra:**

**2.8.2.1 E/S analógicas:** Es el valor eléctrico reflejado en las E/S analógicas. Las entradas estarán direccionadas con la letra "%IW" y las salidas "%QW".

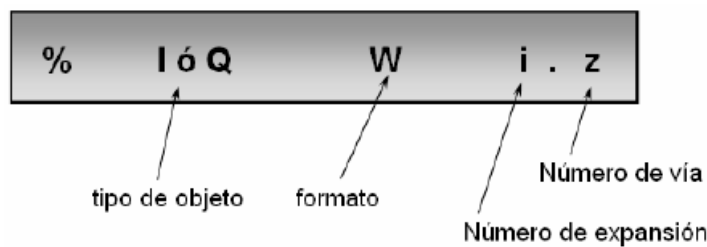


Figura 2.13 Entradas y Salidas Analógicas.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemechanique-84695>

**2.8.3 Memoria interna:** Palabras empleadas para almacenar valores durante la operación en la memoria de datos, se direcciona “%MW”.

- **Constantes:** Almacenan constantes o mensajes alfanuméricos. Su contenido solo se puede escribir o modificar mediante el Twido Suite durante la configuración. Se accede a la zona de constantes a través de “%KW”.

- **Palabras de sistema:** Palabras del sistema que ofrecen diferentes funciones, como proporcionar acceso directo a los estados internos del autómata. La letra para direccionar los bits de sistema será la “%SW”.

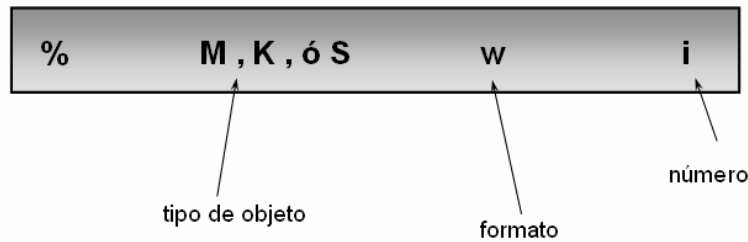


Figura 2.14 Palabras de sistema.  
Fuente: <http://octopart.com/twdlcaa10drf-telemechanique-84695>

## 2.9 Escribir Programa:

El Twido Suite proporciona instrucciones para utilizar los lenguajes de programación Ladder (diagrama de contactos) y Lista (Instrucciones de lista) y GRAFCET.

### 2.9.1 Lenguaje Ladder:

Los diagramas Ladder o de contacto son similares a los diagramas lógicos de relé que representan circuitos de control de relé. Las principales diferencias entre los dos son las siguientes funciones de la programación de Ladder que no aparecen en los diagramas de lógica de relé.

- Todas las entradas están representadas por símbolos de contactos (--| |--).
- Todas las salida están representadas por símbolos de bobinas (--( )--).
- Las operaciones numéricas están incluidas en el conjunto de instrucciones de Ladder gráficas.

### 2.9.1 Instrucciones de Lista:

Un programa escrito en lenguaje de lista está formado por una serie de instrucciones que el autómata ejecuta de forma secuencial. Cada instrucción de lista está representada por una línea de programa y tiene tres componentes: Número de línea, Código de Instrucción y Operandos.

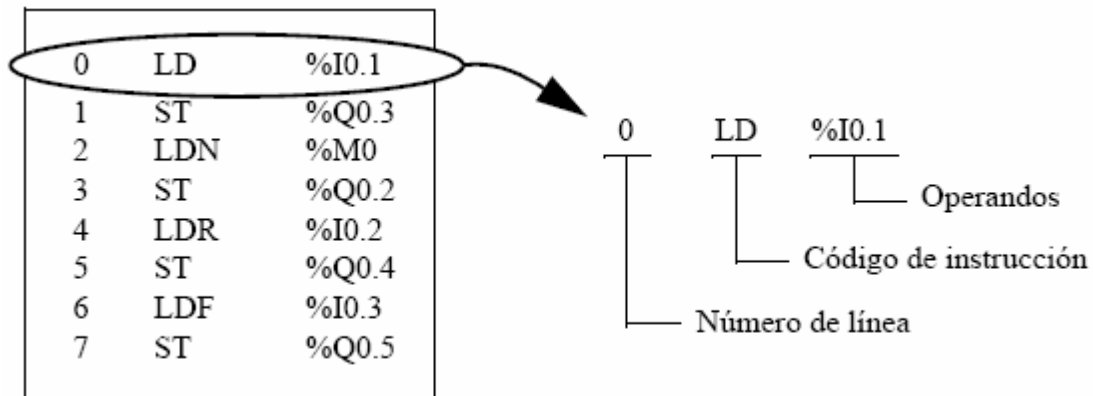


Figura 2.15 Instrucciones de lista.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

### 2.9.2 GRAFCET (Gráfica de Control de Secuencias de Programación):

Las instrucciones Grafcet de TwidoSuite ofrecen un método sencillo para traducir una secuencia de ajuste (diagrama Grafcet).



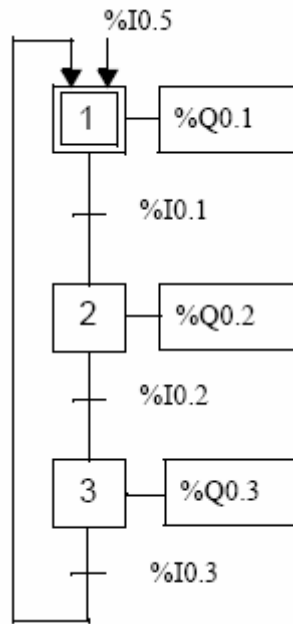


Figura 2.16 Ejemplo de un Programa GRAFCET.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

La cantidad máxima de pasos Grafcet depende del tipo de autómata Twido. La cantidad de pasos activados en cualquier momento están limitados sólo por la cantidad total de pasos.

## 2.10 Abrir el editor de programa:

El “Editor de programa” es el área de trabajo donde se crea la programación en el lenguaje que se haya escogido.

Para abrir el editor de programas se pulsará la pestaña de “Programar” en la barra de pasos de la aplicación y asegurarse que en la barra de subpasos del programa está la pestaña de “Programa”, y aparece la ventana para la edición de programa.

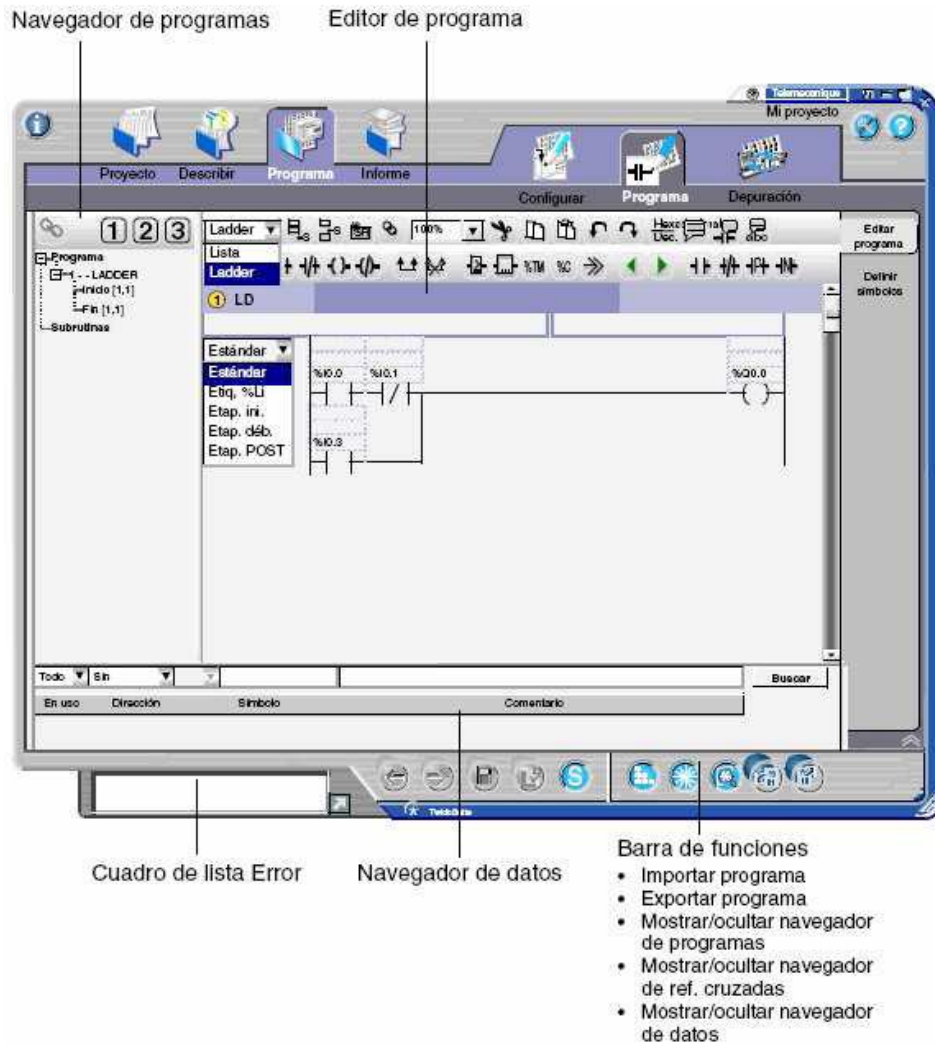


Figura 2.17 Editor del Programa.  
 Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

### 2.10.1 Ventana del editor de programa

- **Navegador de programas:** En esta área se puede observar las diferentes secciones del programa, así como las subrutinas.
- **Editor de programa:** Es el área donde se realizará la programación propiamente dicha de la aplicación.
- **Barra de funciones:** Barra donde se sitúan las funciones adicionales asociadas a la programación (como exportar o importar programas).

**2.10.2 Navegador de datos:** Esta ventana permite buscar donde están siendo usados los símbolos o direcciones, en que partes del programa.

En el editor también aparecerá una barra de instrucciones que cambiará en función del lenguaje de programación seleccionado.

### 2.10.11 Creación de un programa en lenguaje Ladder:

Existen tres barras de herramientas de programación de Ladder disponibles que permiten editar programas e introducir instrucciones de Ladder gráficamente de forma sencilla:

- ✓ Barra de herramientas del programa.
- ✓ Barra de herramientas de la paleta de Ladder.
- ✓ Paleta de Ladder ampliada.

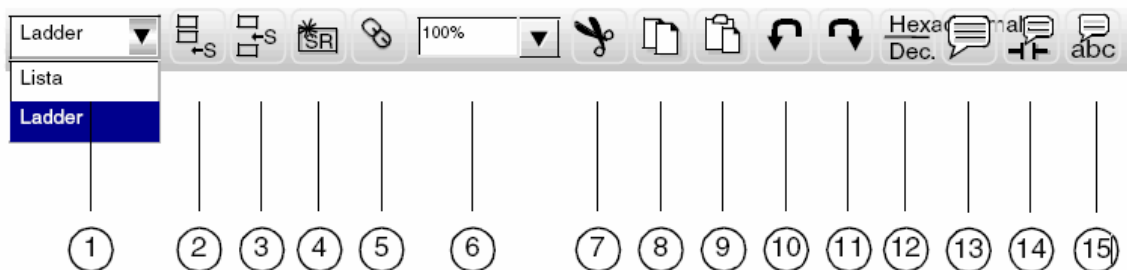


Figura 2.18 Barra de Herramientas Ladder.  
Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

### 2.10.12 Barra de Herramientas Ladder

- |                              |  |
|------------------------------|--|
| 1 Alternar modo Ladder/Lista | 8 Copiar                               |
| 2 Agregar sección            | 9 Pegar                                |
| 3 Insertar sección           | 10 Deshacer                            |
| 4 Agregar subrutina          | 11 Rehacer                             |
| 5 Cortar en dos secciones    | 12 Visualizar en hexadecimal o decimal |

6 Porcentaje del zoom

13 Mostrar/ocultar comentarios del escalón

7 Cortar

14 Mostrar/ocultar comentarios del Ladder

15 Mostrar/ocultar símbolos

## 2.11 Transferencia del programa

Para ejecutar la aplicación cuando vaya a depurarla, primero debe transferirla a la RAM del controlador.

El puerto serie EIA RS232C de su PC se puede conectar al puerto 1 del autómata con el cable TSX PCX1031. Este cable convierte las señales comprendidas entre EIA RS232 y EIA RS485.

El cable TSX PCX1031 incorpora un conmutador rotativo de cuatro posiciones para seleccionar distintos modos de funcionamiento. El conmutador designa las cuatro posiciones como "0-3" y el ajuste apropiado de TwidoSuite para el autómata Twido es la posición 2.

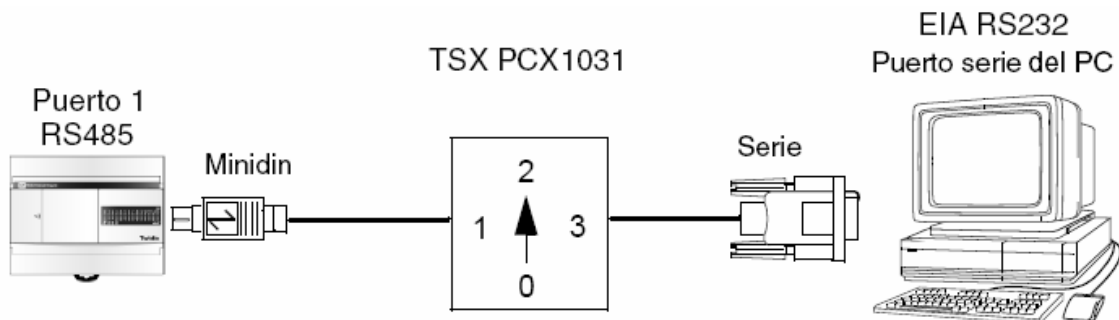


Figura 2.19 Transferencia del programa.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>

Para transferir el programa de aplicación al PLC, si aún no ha conectado el PC al controlador, seleccione una conexión en la tabla de conexiones de la tarea Programa → Depuración → Conectar y haga clic en Aceptar.

El TwidoSuite intenta establecer una conexión con el controlador y realiza comprobaciones de sincronización entre el PC y las aplicaciones del PLC.

## **2.12 Operaciones con señales analógicas:**

En este apartado se describe el modo de gestionar los potenciómetros analógicos incorporados y los módulos de ampliación analógicos.

### **2.12.1 Potenciómetros analógicos:**

Los controladores Twido cuentan con uno o dos potenciómetros analógicos. Los potenciómetros aportan valores numéricos, de 0 a 1.023 para el potenciómetro analógico 1 y de 0 a 511 para el potenciómetro analógico 2, correspondientes a los valores analógicos que indican estos potenciómetros, forman parte de las dos palabras de entrada siguientes:

- **%IW0.0.0** para el potenciómetro analógico 1 (a la izquierda)
- **%IW0.0.1** para el potenciómetro analógico 2 (a la derecha)

Estas palabras se pueden utilizar en operaciones aritméticas. Se pueden emplear para cualquier tipo de ajuste (preselección de un retardo o de un contador, ajuste de la frecuencia del generador de pulsos o de la duración del precalentamiento de una máquina, etc.).

## **2.13 Módulos analógicos:**

Además del potenciómetro integrado de 10 bits y el canal analógico de 9 bits, todos los autómatas Twido que admiten ampliaciones de E/S pueden configurar módulos de E/S analógicos y comunicarse con ellos.

### **2.13.1 Funcionamiento de los módulos analógicos:**

Las palabras de entrada y de salida (%IW y %QW) se emplean para intercambiar datos entre la aplicación de usuario y los canales analógicos. La actualización de estas palabras se realiza de manera sincronizada con la exploración del autómata en modo RUN.

## **2.14 OPC Server.**

El OPC (OLE for Process Control) es un estándar de comunicación en el campo del control y supervisión de procesos industriales, basado en una tecnología Microsoft, que ofrece un interface común para comunicación que permite que componentes software individuales interaccionen y compartan datos. La comunicación OPC se realiza a través de una arquitectura Cliente-servidor. El servidor OPC es la fuente de datos (como un dispositivo hardware a nivel de planta) y cualquier aplicación basada en OPC puede acceder a dicho servidor para leer/escribir cualquier variable que ofrezca el servidor. Es una solución abierta y flexible al clásico problema de los drivers propietarios. Prácticamente todos los mayores fabricantes de sistemas de control, instrumentación y de procesos han incluido OPC en sus productos.

## **2.15 LabVIEW.**

Constituye un revolucionario sistema de programación gráfica para aplicaciones que involucren adquisición, control, análisis y presentación de datos. Las ventajas que proporciona el empleo de LabVIEW se resumen en las siguientes:

- ✓ Se reduce el tiempo de desarrollo de las aplicaciones al menos de 4 a 10 veces, ya que es muy intuitivo y fácil de aprender.
- ✓ Dota de gran flexibilidad al sistema, permitiendo cambios y actualizaciones tanto del hardware como del software.
- ✓ Da la posibilidad a los usuarios de crear soluciones completas y complejas.
- ✓ Con un único sistema de desarrollo se integran las funciones de adquisición, análisis y presentación de datos.
- ✓ El sistema está dotado de un compilador gráfico para lograr la máxima velocidad de ejecución posible.
- ✓ Tiene la posibilidad de incorporar aplicaciones escritas en otros lenguajes.

LabVIEW es un entorno de programación destinado al desarrollo de aplicaciones, similar a los sistemas de desarrollo comerciales que utilizan el lenguaje C o BASIC. Sin embargo, LabVIEW se diferencia de dichos programas en un importante aspecto: los

citados lenguajes de programación se basan en líneas de texto para crear el código fuente del programa, mientras que LabVIEW emplea la programación gráfica o lenguaje G para crear programas basados en diagramas de bloques.

Para el empleo de LabVIEW no se requiere gran experiencia en programación, ya que se emplean íconos, términos e ideas familiares a científicos e ingenieros, y se apoya sobre símbolos gráficos en lugar de lenguaje escrito para construir las aplicaciones. Por ello resulta mucho más intuitivo que el resto de lenguajes de programación convencionales. LabVIEW posee extensas librerías de funciones y subrutinas. Además de las funciones básicas de todo lenguaje de programación, LabVIEW incluye librerías específicas para la adquisición de datos, control de instrumentación y comunicación serie; análisis presentación y guardado de datos.

## **2.16 COMUNICACIÓN INDUSTRIAL**

Son aquellas que permiten el flujo de información del controlador a los diferentes dispositivos a lo largo del proceso de producción: detectores, actuadores, sensores, etc. Los procesos a automatizar acostumbran a tener un tamaño importante y este hecho provoca que exista una gran cantidad de cables entre el autómeta los sensores y actuadores. Existen diferentes maneras de comunicar los diferentes dispositivos dependiendo de la complejidad de la red creada y el presupuesto destinada a su creación.

## **2.17 PROTOCOLOS DE COMUNICACIÓN**

### **2.18 MODBUS**

- ✓ Desarrollado por Modicon para comunicación entre PLC's.
- ✓ Debido a su simplicidad y especificación abierta, actualmente es ampliamente utilizado por diferentes fabricantes.
- ✓ Entre los dispositivos que lo utilizan podemos mencionar: PLC, HMI, RTU, Drives, sensores y actuadores remotos.
- ✓ El protocolo establece cómo los mensajes se intercambian en forma ordenada y la detección de errores.

## 2.18.1 Principales Características

- ✓ Control de acceso al medio tipo Maestro/Esclavo.
- ✓ El protocolo especifica: formato de trama, secuencias y control de errores.
- ✓ Existen dos variantes en el formato: ASCII y RTU
- ✓ Sólo especifica la capa de enlace del modelo ISO/OSI.
- ✓ A cada esclavo se le asigna una dirección fija y única en el rango de 1 a 247.
- ✓ La dirección 0 está reservada para mensajes de difusión sin respuesta.

## 2.18.2 Formato General de las Tramas.

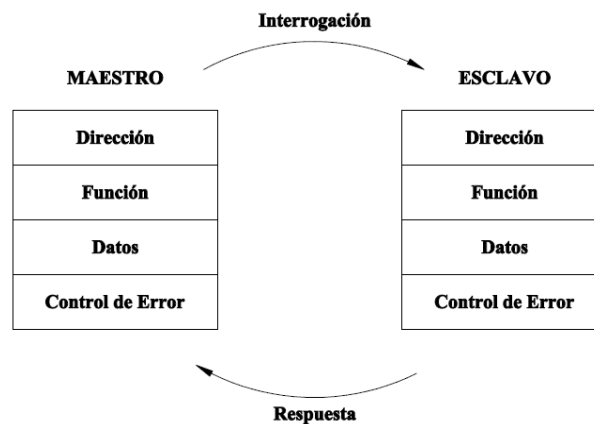


Figura 2.20 Trama Modbus.

Fuente: <http://octopart.com/twdlcaa10drf-telemecanique-84695>



## **CAPÍTULO III**

### **DESARROLLO DEL TEMA**

#### **3.1 PRELIMINARES**

En este capítulo se detalla paso a paso cómo se realizó la implementación de un PLC Twido para controlar una estación de nivel, como ejemplo práctico se realiza la simulación de un proceso industrial el mismo que de seguro se podrá encontrar en el ámbito laboral.

La variación del nivel del líquido será medida por medio del sensor de nivel que se encuentra instalado en el módulo y la variación de grados centígrados por un sensor de temperatura. Para calentar el líquido se usó una niquelina y para enfriarlo un ventilador acoplado a un radiador, además se usa una electroválvula que ayuda a abrir o cerrar el paso de líquido al tanque. Las señales adquiridas de las variables físicas ingresan a un módulo de expansión de entradas y salidas analógicas el mismo que permite ingresar estos datos al PLC.

La estación genera una señal de corriente según las variaciones del nivel, así que se realiza un acondicionamiento de la señal para transformar una señal de corriente de 4-20mA a una señal de voltaje de 0-5V.

Se usó un cable serial para la transmisión y programación del autómeta.

##### **3.1.1 INSTALACIÓN DE EQUIPOS.**

Se realizaron las conexiones del PLC, el módulo de E/S analógicas y el acondicionamiento de las señales de nivel y temperatura. Los elementos usados se detallan a continuación.



Figura 3.1: Montaje del PLC.  
Realizado por: Javier Ortiz.

### 3.2 PLC Twido TWDLCAA24DFR

Es un modulador compacto fabricado por SCHNEIDER ELECTRIC, es el controlador principal, es quien permitirá realizar diferentes acciones durante el proceso.

#### 3.2.1 Módulo de expansión de 2 entradas y 1 salida analógica TM2AMM3HT.

Este módulo permitirá ingresar las señales adquiridas al PLC, necesita de una fuente externa de alimentación de 24VDC para su funcionamiento. Se instala en el conector de ampliación del PLC.



Fig. 3.2 Módulo de entradas y salidas analógicas.  
Realizado por: Javier Ortiz

### 3.2.2 Fuente de 24VDC.

Permitirá que el módulo E/S analógicas funcione ya que proporcionará el voltaje y la corriente adecuada para alimentar al circuito.



Fig. 3.3 Fuente de alimentación de 24VDC  
Elaborado por Javier Ortiz

Su circuito interno se detalla a continuación.

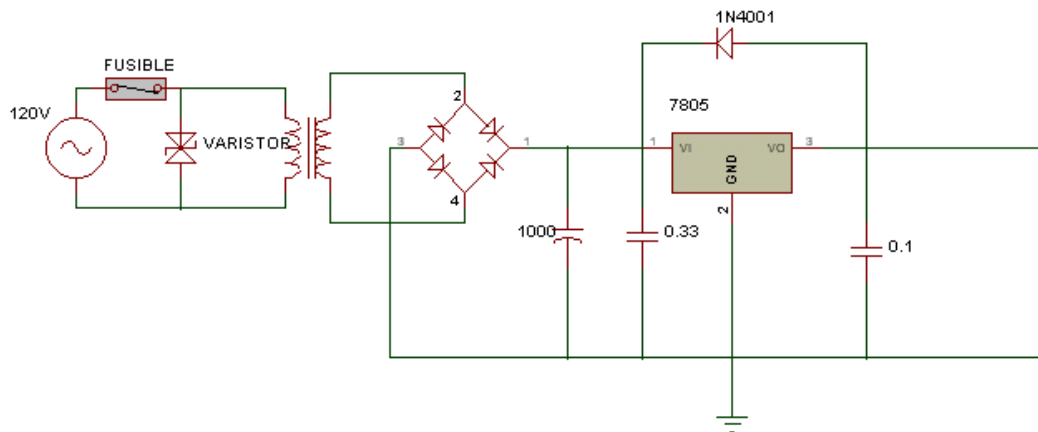


Fig. 3.4 Circuito de la fuente de 24VDC  
Fuente: [www.rac.1324.freewebspace.com](http://www.rac.1324.freewebspace.com)

### 3.3 Programación del PLC Twido.

Para poder programar el PLC, el computador debe contar con el software Twido Suite.

#### 3.3.1 INSTALACIÓN DEL SOFTWARE DE PROGRAMACIÓN TWIDOSUITE.

El TwidoSuite es un software de programación utilizado para la configuración, programación y depuración de la gama de controladores programables Twido.

El TwidoSuite es un software gratuito que se puede descargar desde la página del ISEFONLINE, a través de la siguiente dirección.

[www.isefonline.com](http://www.isefonline.com)

Para empezar la instalación, abrir el archivo ejecutable que previamente se ha descargado.



Figura 3.5 Ícono del archivo ejecutable.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

Aparecerá una ventana flotante con la información de los derechos usuario de la licencia del software TwidoSuite, pulsamos el botón “Acept.”. Una vez aceptada la licencia, se abre una ventana nueva donde se puede colocar la ruta que se quiere descomprimir el instalador. (Al no cambiar la ruta, el programa creará por defecto “C:\Burndisk”). Cuando se haya especificado la ruta, se deberá pulsar el botón “Install” para comenzar a descomprimir el archivo. Abriendo la ruta especificada donde se ha creado la carpeta de instalación y pulsando sobre el icono “Setup” mostrará la ventana inicial de instalación.



Figura 3.6 Ícono Setup.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

**NOTA:** Si se dispone de otra versión de TwidoSuite instalada en el PC, desinstalarla previamente a la instalación de la nueva versión.

Durante el proceso de instalación, se debe aceptar el contrato de licencia, colocar el nombre de la organización, especificar la ruta donde se desea instalar el programa, si se quiere colocar un ícono en el escritorio o en la barra de inicio rápido y en qué carpeta de la barra de programas se desea colocar el software.

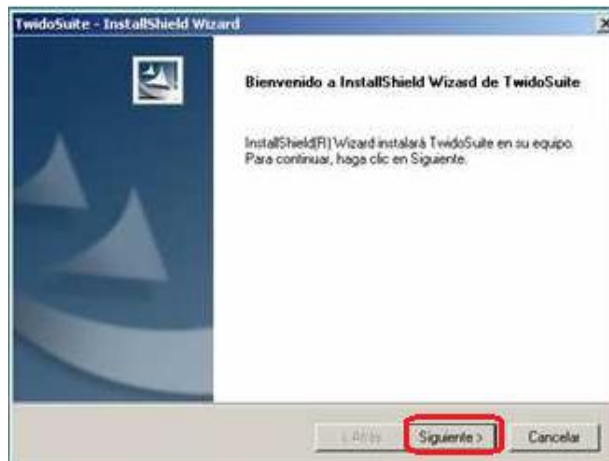


Figura 3.7 Inicio del proceso de instalación.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

Leer y aceptar la licencia provista en la ventana, para aceptar los derechos de autor.

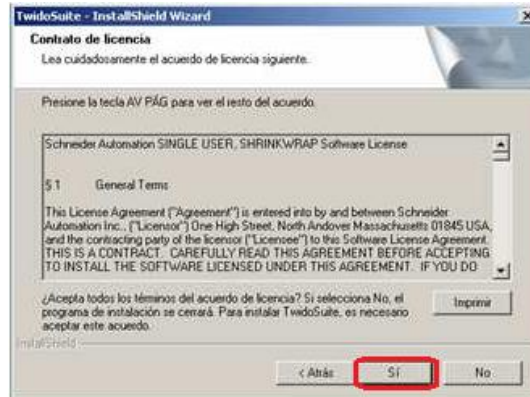


Figura 3.8 Contrato de licencia.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

Nombre del servidor en el cual se instalará el software Twido Suite.

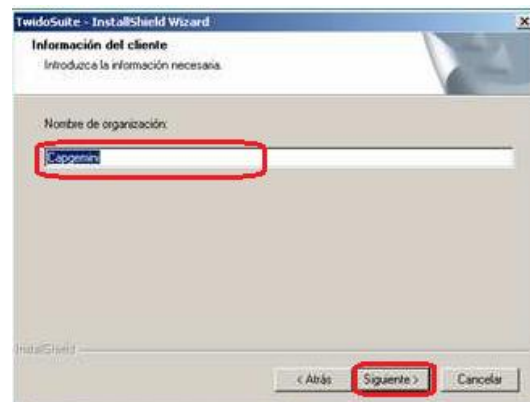


Figura 3.9 Nombre del organizador.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

Dirección en la cual se desea guardar los archivos del software Twido Suite. Se guardaran en el disco C.



Figura 3.10 Ruta de instalación del programa.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

Seleccionar si se desea que se cree un icono de acceso rápido en el escritorio y en el menú inicio.

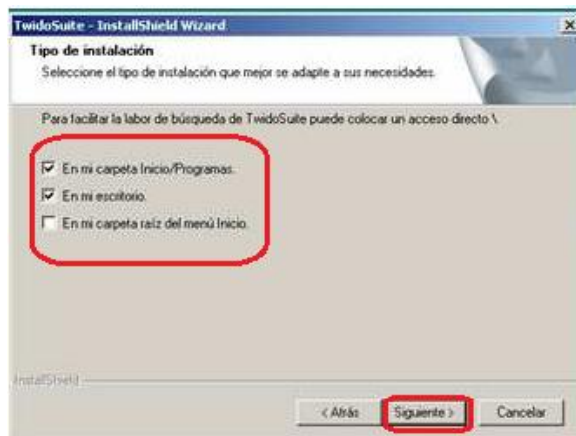


Figura 3.11 Tipo de instalación.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

De la lista de programas seleccionar el nombre Twido Suite para determinar el tipo de software que se está instalando.



Figura 3.12 Seleccionar carpeta de programas.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

### 3.3.2. CREAR UN NUEVO PROYECTO.

1. Una vez finalizada la instalación del proyecto, para abrir por primera vez el TwidoSuite debe ir al ícono que se ha generado en el escritorio (si se ha seleccionado en la instalación) o se busca dentro de la barra de programas de la PC.

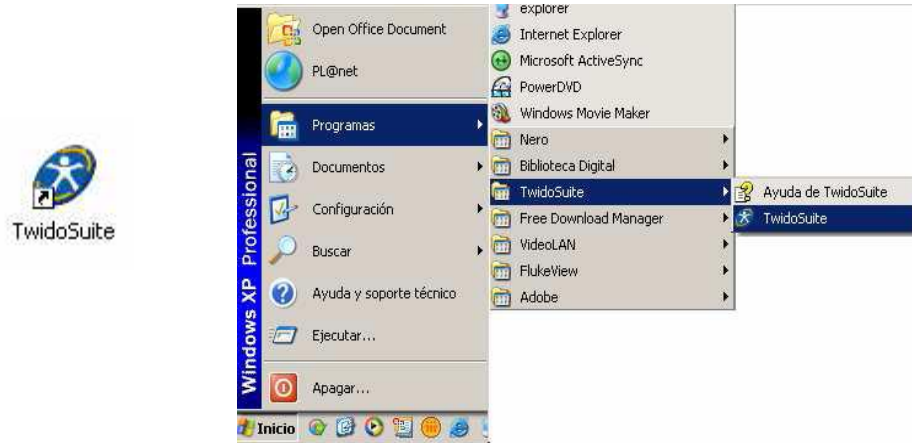


Figura 3.13 Opciones de apertura de TwidoSuite.  
Fuente: Manual Twido.

2. Una vez abierto el programa TwidoSuite se despliega la ventana principal del programa. Esta ventana permite seleccionar el idioma y la opción de trabajo, en este caso se selecciona español para idioma y modo programación en las opciones de trabajo como se muestra a continuación.

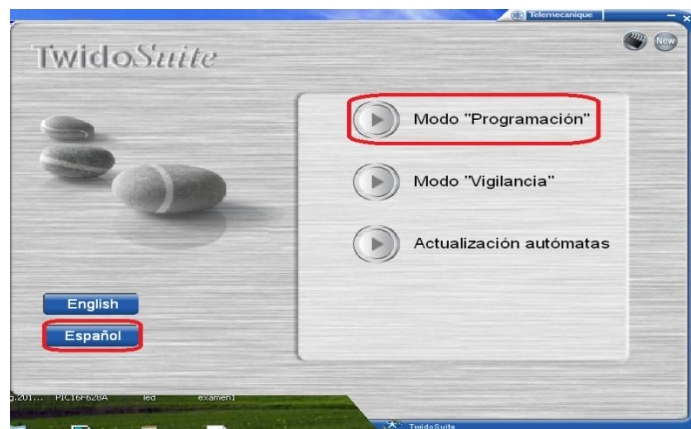


Figura 3.14 Pantalla inicial de TwidoSuite.  
Realizado por: Javier Ortiz.



3. Para crear el primer proyecto, de clic en “Modo Programación”. Al ingresar a este modo aparecerá la ventana de proyecto, donde podrá realizar las gestiones de los proyectos (Crear, abrir, guardar y cerrar un proyecto).

Pulsar en “Crear un proyecto nuevo” dentro del marco de acciones de la ventana, acto seguido rellenar los campos de información general del proyecto, como puede ser: el nombre del proyecto, la ruta donde se guardará, el autor, la compañía, etc.

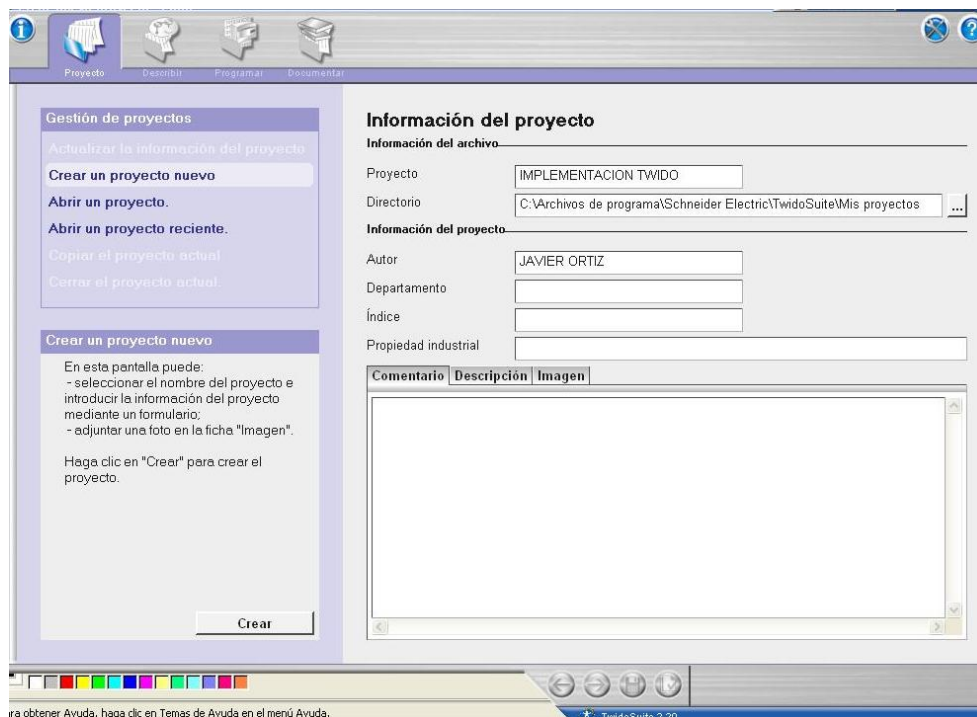


Figura 3.15 Ventana de la pestaña de “Proyecto”.

Realizado por: Javier Ortiz.

Una vez que se haya introducido la información del proyecto pulsando el botón “Crear” que aparece en la parte de abajo del marco de acciones; el nuevo proyecto se generará.

4. Para configurar el hardware Twido hacer clic sobre el ícono “Describir”.



Figura 3.16 Ícono describir.

Fuente: [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)

En el panel de catálogo de productos seleccionar bases, compactos y pulsando sobre el Controlador Compacto TWDLCAA24DFR arrastrar hacia la región del panel gráfico agregando el elemento al hardware.

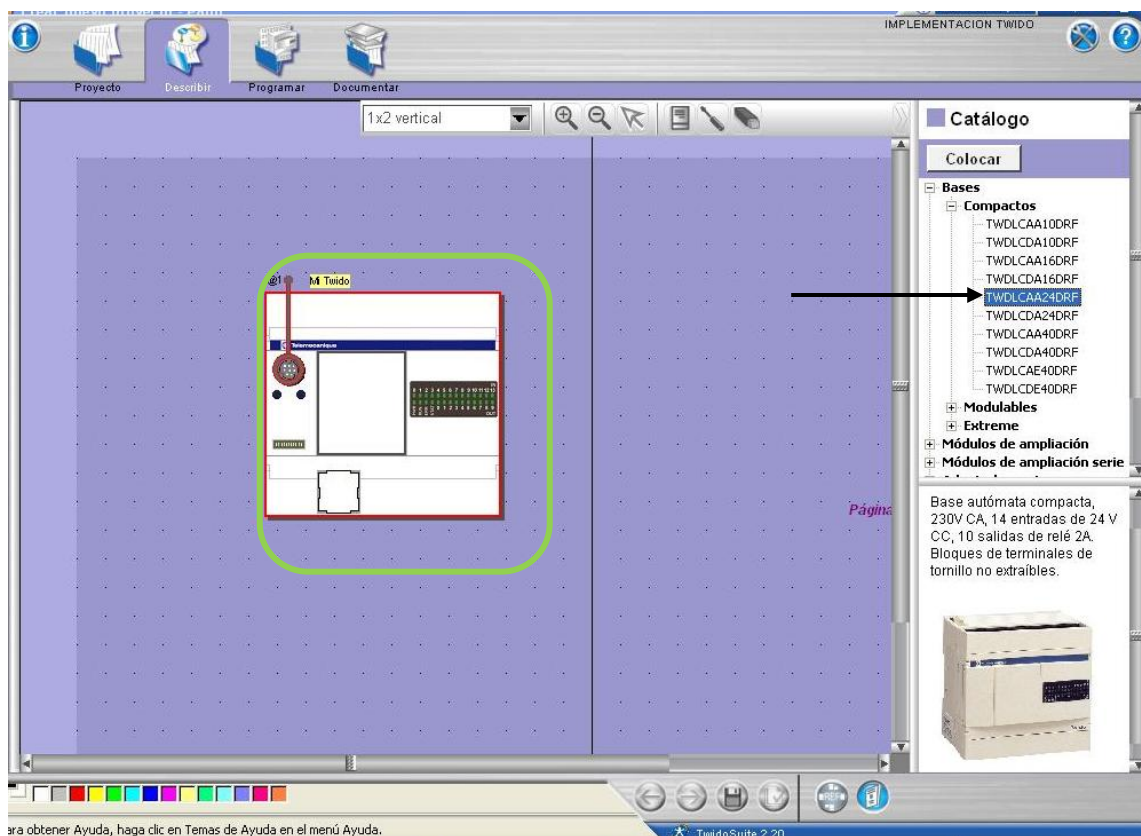


Figura 3.17 Ventana de configuración de Hardware “Describir” Selección de controlador compacto.

Realizado por: Javier Ortiz.

5. Una vez seleccionado el Controlador Compacto TWDLCAA24DFR se insertará el módulo de ampliación TM2AMM3HT con sus respectivas configuraciones como se muestra a continuación.

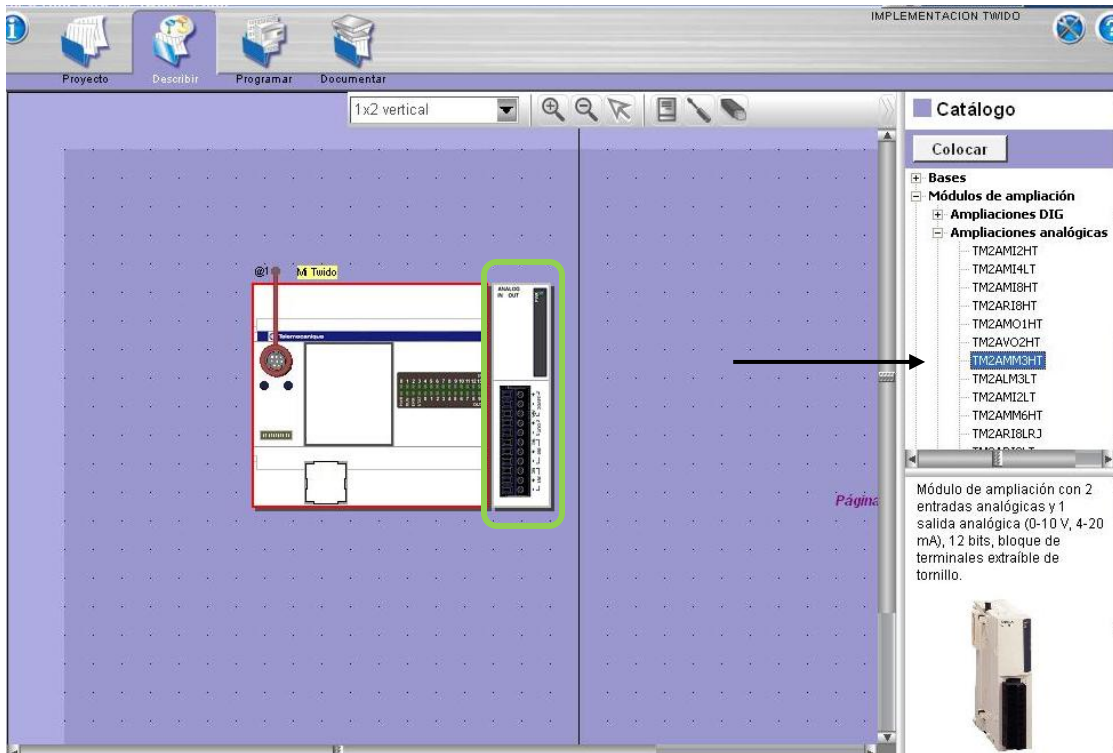


Figura 3.18 Selección de módulo de ampliación.

Realizado por: Javier Ortiz.

- En el módulo de ampliación habilitar las entradas analógicas con un rango de 0-10V.

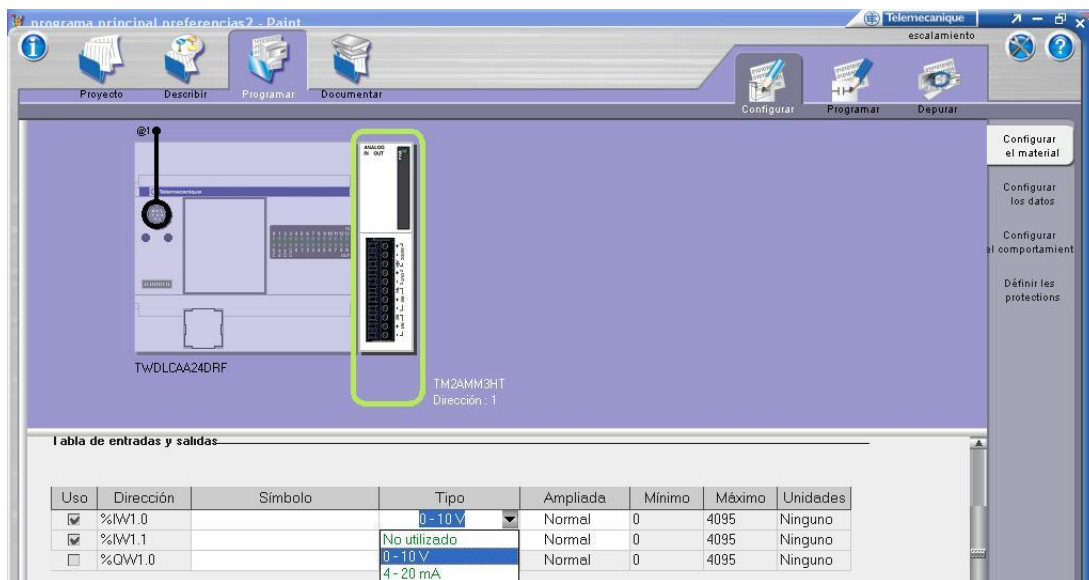


Figura 3.19 Configuración del módulo de ampliación.

Realizado por: Javier Ortiz.

7. Cuadro de programación. Aquí se realiza el algoritmo de programación y se ingresan los diferentes bloques del programa.

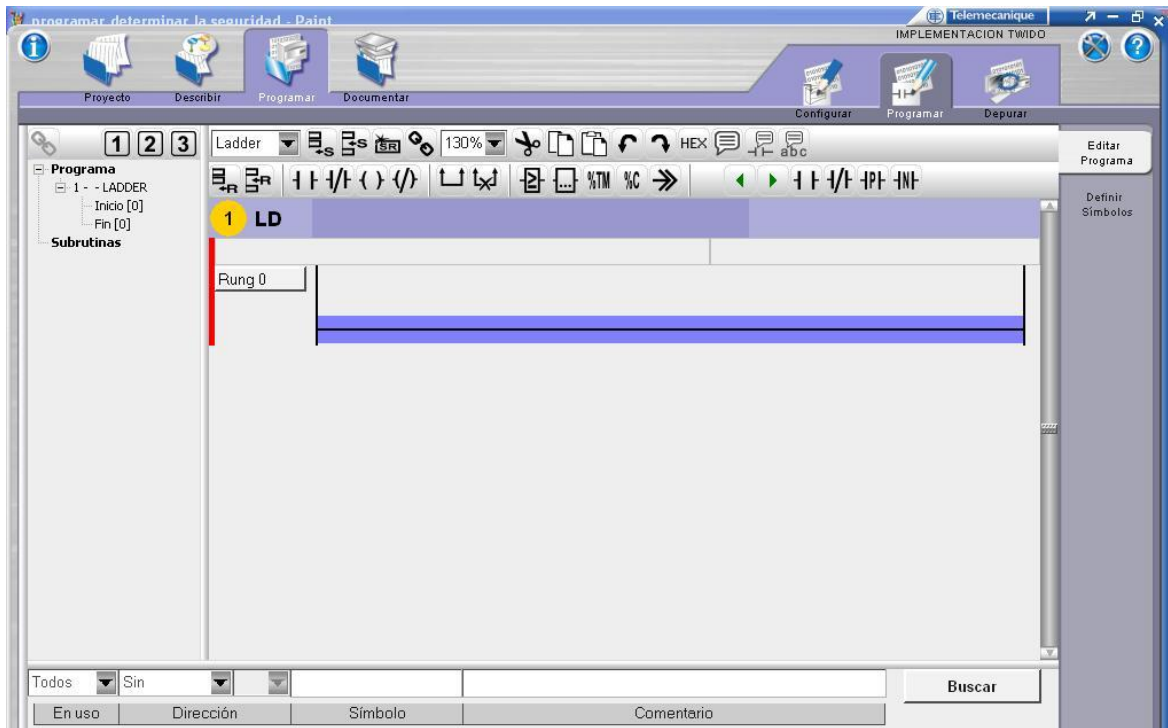


Figura 3.20 Cuadro de programación.

Realizado por: Javier Ortiz.

8. Cuadro de comunicación. Mediante este cuadro se realiza la comunicación entre la computadora y el autómatas. Aquí se configura el puerto por el cual se enviará la información y el modo de comunicación que se usará.



Figura 3.21 Cuadro de comunicación.

Realizado por: Javier Ortiz.

### 3.3.3. DESARROLLO DEL PROGRAMA PRINCIPAL.

Para realizar el programa primero hay que conocer las características del PLC TWIDO TWDLCAA24DRF como las memorias que posee, es decir las memorias tipo Word %MW que almacenan bits de hasta 16 bits y van desde %MW0 hasta %MW9999.

También se cuenta con memorias digitales como son %M, estas memorias pueden almacenar un solo bit ya sea 0 o 1 lógico y están comprendidas desde %M0 hasta %M999.

Esto será de mucha ayuda ya que se trabaja con variables muy extensas y con un gran número de bits. Las memorias digitales sirven para guardar el estado de una salida y entrada digital del PLC. Además hay que conocer las características del software Twido Suite como:

En el PLC se pueden almacenar 128 secciones, en las mismas se pueden escribir 7 líneas de programación.

1. Para el desarrollo del programa se ingresó a la pestaña programa y al icono programar.

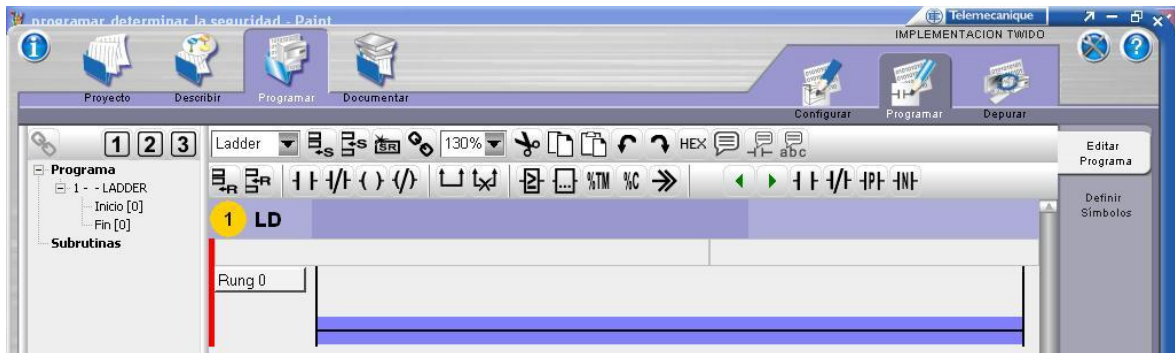


Figura 3.22 Cuadro de programación.

Realizado por: Javier Ortiz.

2. Primero se realizó el acondicionamiento de las señales de los sensores para que se encuentren en un rango de 0-10V y puedan ser procesados por el PLC, estas entradas están representadas por las direcciones %IW1.0 y %IW1.1.

Luego se realizó un escalamiento para poder visualizar los valores correspondientes a cada señal, ya que posteriormente se necesitará visualizar dichos datos.

El valor de cada señal que se adquiere, se representa con una resolución de 12 bits, o sea:

$$\begin{aligned}0 \text{ voltios} &= 0 \text{ bits o } 0 \\10 \text{ voltios} &= 12 \text{ bits o } 4095\end{aligned}$$

El rango que se usa en el tanque está determinado de acuerdo al software del PLC y corresponde a:

$$\begin{aligned}400 \text{ bits} &\text{ es a } 1 \text{ cm} \\2011 \text{ bits} &\text{ es a } 24 \text{ cm}\end{aligned}$$

Este rango se tomó de los bits que el PLC muestra en el software de programación dependiendo del nivel del tanque.

Por lo tanto la ecuación para el escalamiento queda así:

$$y = m(\# - INmin) + 1$$

Dónde:

$m$  = pendiente

$INmin$  = valor mínimo de la entrada

$\#$  = entrada analógica

$X$  = resultado equivalente en centímetros

Sustituyendo los valores indicados anteriormente:

$$\begin{aligned}m &= \frac{24-1}{2011-400} \\y &= \frac{23}{1611} (\# - 400) + 1\end{aligned}$$

El rango que se usa para la temperatura es:

0 bits es a 20°C  
2033 bits es a 70°C

Por lo tanto la ecuación para el escalamiento queda así:

$$y = m(\# - INmin) + 20$$

Dónde:

m= pendiente

INmin= valor mínimo de la entrada

# = entrada analógica

y = resultado equivalente en grados centígrados

Sustituyendo los valores especificados anteriormente tenemos:

$$m = \frac{70-20}{2033-0}$$
$$y = \frac{50}{2033} (\# - 0) + 20$$

Para la realización de la programación de la ecuación en el PLC fue necesario usar:

- a) Palabras de Memoria (%MWn): Las palabras de memoria sirven para guardar operaciones aritméticas o algún valor de cierta variable con un formato de 16 bits.
- b) Palabra de Memoria Doble (%MDn): Las palabras de memoria doble sirven para guardar operaciones aritméticas o algún valor de ciertas variables con un formato de 32 bits.

### 3.3.4 Bloques de operación.

Sirven para realizar operaciones aritméticas. Las palabras de memoria sólo pueden ser utilizadas dentro de un bloque de operación para lo cual se debe insertar una línea de conexión y seleccionarla para insertar el bloque de operación.

1. Se ingresó la variable nivel representada por la entrada analógica %IW1.0 y se almacena en una memoria interna %MW0. Como son valores con una gran cantidad de bits debido a que se incrementaran por los cálculos del escalamiento, es necesario incrementar el tamaño de la memoria usando una memoria doble Word y almacenar este valor en la dirección %MD0;=DWORD (%MW0).

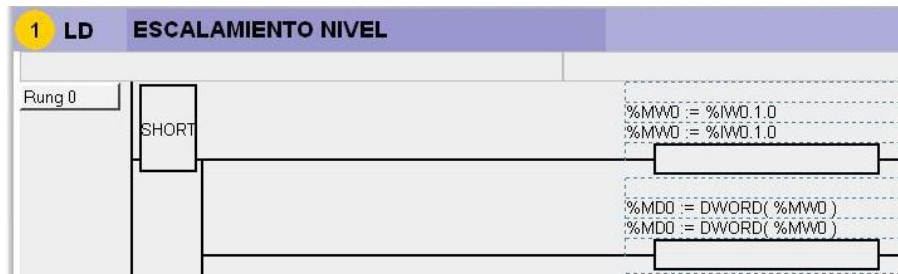


Figura 3.23 Ingreso de variable analógica.

Realizado por: Javier Ortiz.

2. En la figura 3.23 se empezó a realizar los bloques de operación. Los mismos que servirán para representar el escalamiento. A la dirección %MD0 restar una marca %MD30 y almacenar en la memoria %MD1. Cabe recalcar que la marca %MD30 será modificada desde Labview, de acuerdo a las necesidades del programador, dicha marca sirve para reducir el error residual que existe al adquirir los valores de la señal de nivel.

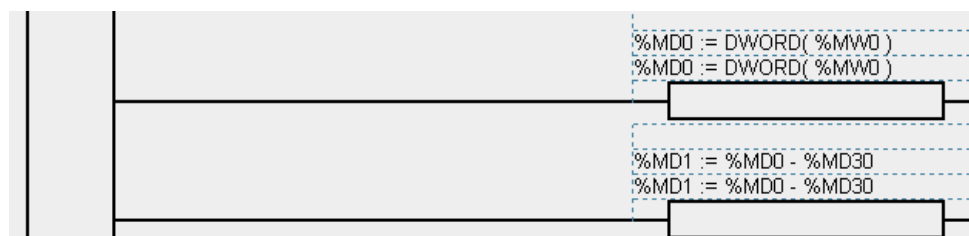


Figura 3.24 Operaciones aritméticas.

Realizado por: Javier Ortiz.



- En el siguiente bloque de operación multiplicar a la memoria  $\%MD1 * 23$  y almacenar este nuevo valor en la memoria  $\%MD2$ .

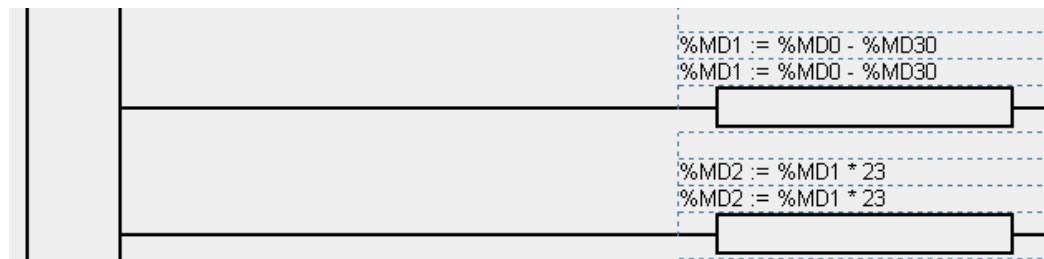


Figura 3.25 Operaciones aritméticas.

Realizado por: Javier Ortiz.

- En este bloque de operación dividir la memoria  $\%MD2 / \%MD35$  y este nuevo valor se almacena en una memoria  $\%MD3$ . Cabe recalcar que la marca  $\%MD35$  es una variable que se modificará desde LabVIEW.
- Finalmente a la memoria  $\%MD3$  sumar + 1 según la fórmula y almacenar este nuevo valor en la memoria  $\%MD4$ , siendo  $\%MD4$  el resultado y el valor de x.

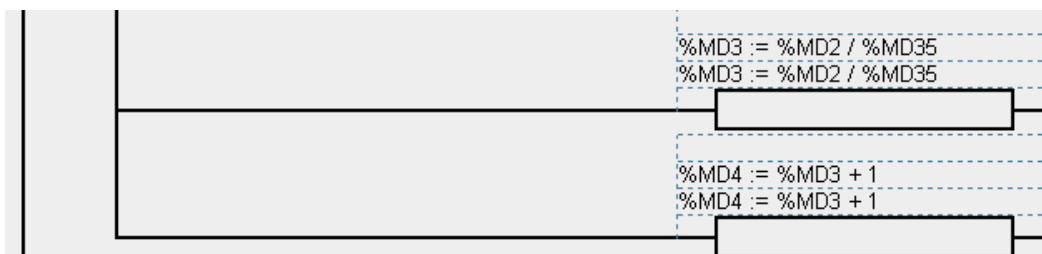


Figura 3.26 Operaciones aritméticas.

Realizado por: Javier Ortiz.

- En el bloque de operación N<sup>o</sup>2 se agrega una memoria  $\%MD60 := \%MD60$  esto se hace para poder ingresar un valor desde otra locación, es decir para direccionar un valor hacia ésta memoria y poder usarla posteriormente.



Figura 3.27 Variable de nivel.  
Realizado por: Javier Ortiz.

- Para el escalamiento de la temperatura hay que realizar el mismo procedimiento que en el caso del acondicionamiento de la señal de nivel, pero tomando en cuenta que ésta variable es mucho más lenta. Además verificar el número de bits que corresponden a la temperatura máxima y a la temperatura mínima.

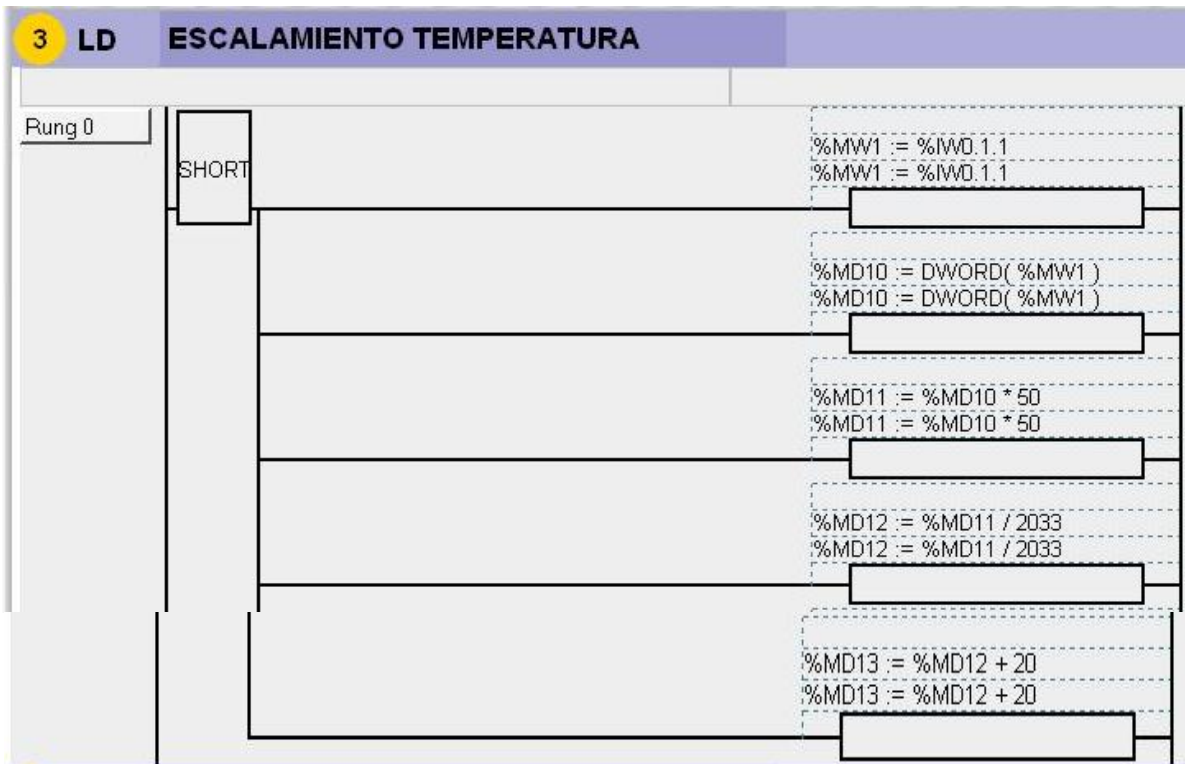


Figura 3.28 Escalamiento de temperatura.  
Realizado por: Javier Ortiz.

8. En este bloque del programa se ingresan marcas o máscaras.

En el PLC se necesita activar o desactivar entradas, pero esto solo es posible al enviar una señal de voltaje física. Para eliminar la intervención del hombre se deben usar marcas las mismas que se pueden activar mediante un comando, es decir, enviando mediante software un cero o uno para poder accionarla.

Los contactos de salida pueden ser usados solo una vez y si se los usa en otro logaritmo es posible que generen un error. Se ingresa la marca %M0 la misma que será un contacto normalmente cerrado y al ser accionado cortará la señal que llega a la marca %M21 y la desactivará y de esta manera el programa se detendrá.

La marca %M20 es un contacto normalmente abierto que al ser accionado encenderá la marca %M21 que actúa como una bobina o salida. La marca %M20 será la encargada de permitir que el programa corra con normalidad.

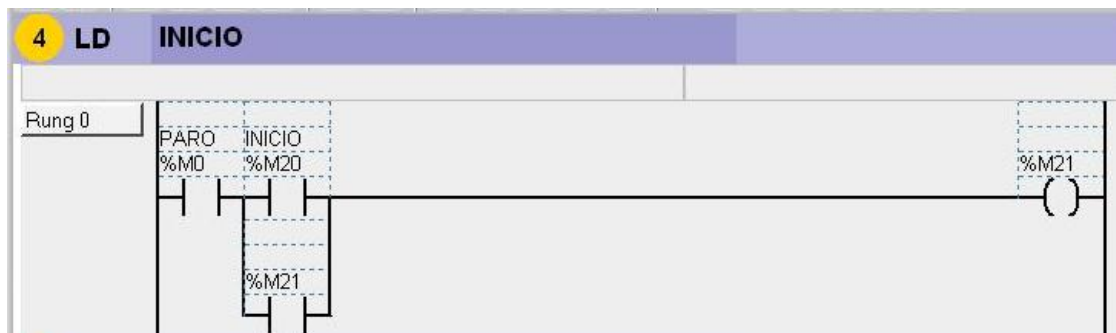


Figura 3.29 Inicio.

Realizado por: Javier Ortiz.

9. En el bloque de programa N°5 se muestra como se activa la marca %M1 cuando se cumplan todas las condiciones adjuntadas y permitirá que el contacto de la salida %Q0.0 que se encuentra en el bloque de programa N°6 se active permitiendo el llenado del tanque. Se activará una vez que el programa determine que el tanque está vacío y aún no necesita mantenimiento. También

se usa un contacto normalmente cerrado que controlará que el tanque no se llene nuevamente una vez que la niquelina se encienda.

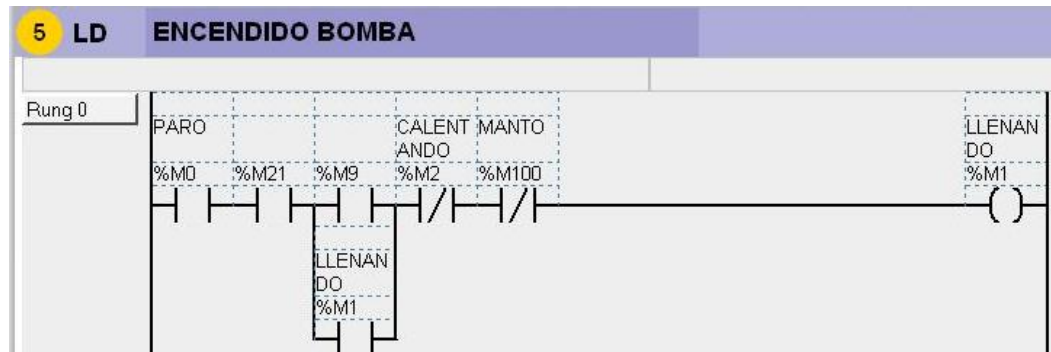


Figura 3.30 Encendido de la bomba.  
Realizado por: Javier Ortiz.

10. En este bloque de programa la marca %M1 enviará una señal a la bobina para que se active la salida física %Q0.0 la misma que encenderá la bomba para llenar el tanque hasta el nivel predeterminado y almacenado en la memoria %MD13 que se encuentra en el bloque de programación N°9.

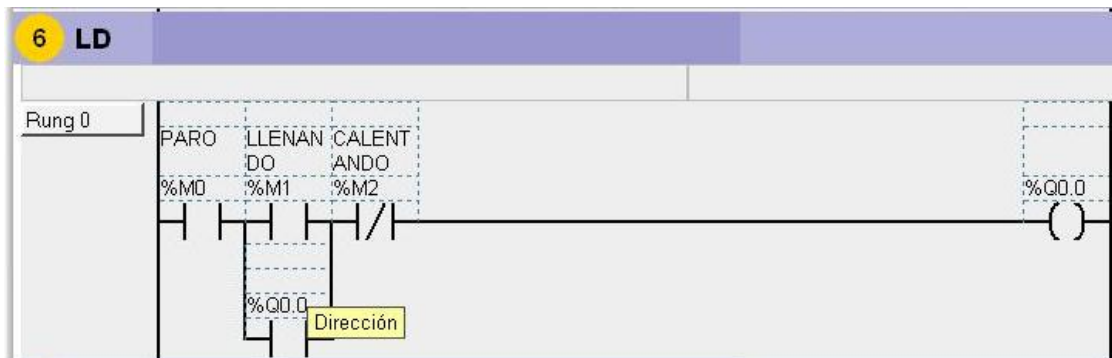


Figura 3.31 Llenado del tanque.  
Realizado por: Javier Ortiz.

11. En el siguiente bloque de programa se realiza la comparación entre la marca %MD4 y la marca %MD60. La marca %MD60 es modificada desde LabVIEW para determinar la temperatura a la cual debe empezar el calentamiento del tanque.



Figura 3.32 Encendido de la niquelina.  
Realizado por: Javier Ortiz.

12. Una vez que el nivel del líquido del tanque haya llegado al valor predeterminado se activará la bobina %Q0.2 y permitirá que la niquelina se encienda hasta que el líquido llegue a la temperatura predeterminada en la comparación realizada por software.

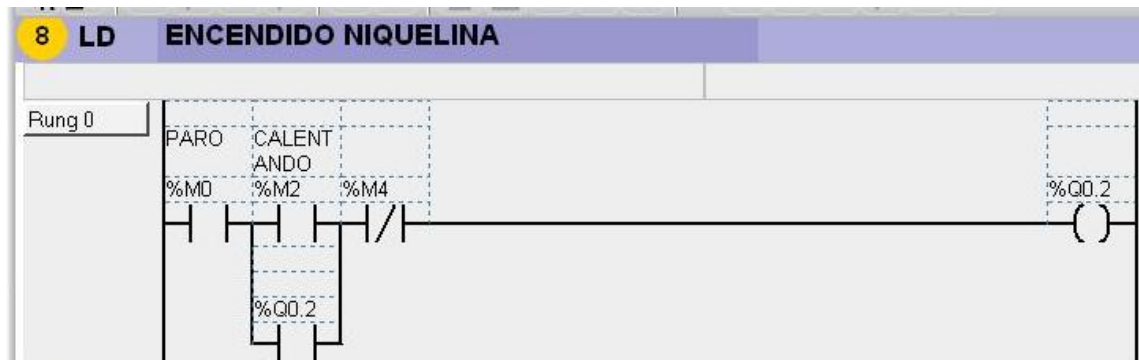


Figura 3.33 Encendido de la niquelina.  
Realizado por: Javier Ortiz.

13. En este bloque del programa se realiza una comparación para determinar el nivel máximo del líquido en el tanque. Cuando se cumpla la comparación se activará la marca %M3.



Figura 3.34 Cuadro de comparación.  
Realizado por: Javier Ortiz.

14. Al activarse la marca %M3 permite el paso de una señal hacia la marca %M4 para que se active la mezcladora.

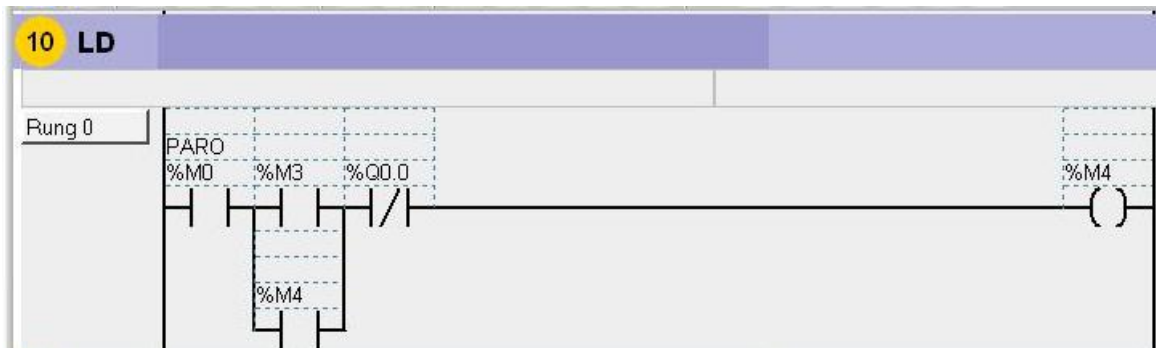


Figura 3.35 Mezcladora.  
Realizado por: Javier Ortiz.

15. Para el tiempo de mezclado es necesario una marca %M5 que se activará cuando la marca %M4 se active y se cumplan todas las condiciones adjuntas. La marca %M5 es la que activa el contador del bloque 12.

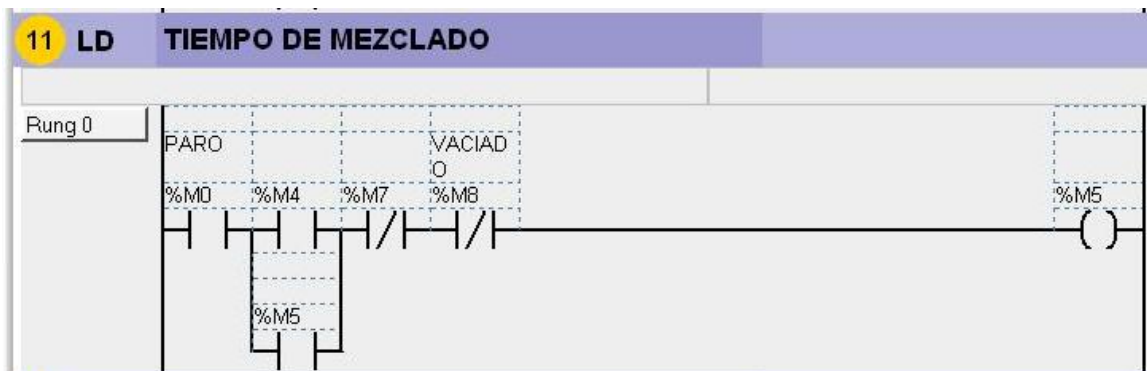


Figura 3.36 tiempo de mezclado.  
Realizado por: Javier Ortiz.

16. La marca %M5 activa al contador descendente TM0 y se enciende su salida Q activando a la marca %M6 que representa a la mezcladora del líquido, una vez que se cumpla el tiempo determinado se desactivará su salida Q y apagará a la marca %M6 apagando la mezcladora.

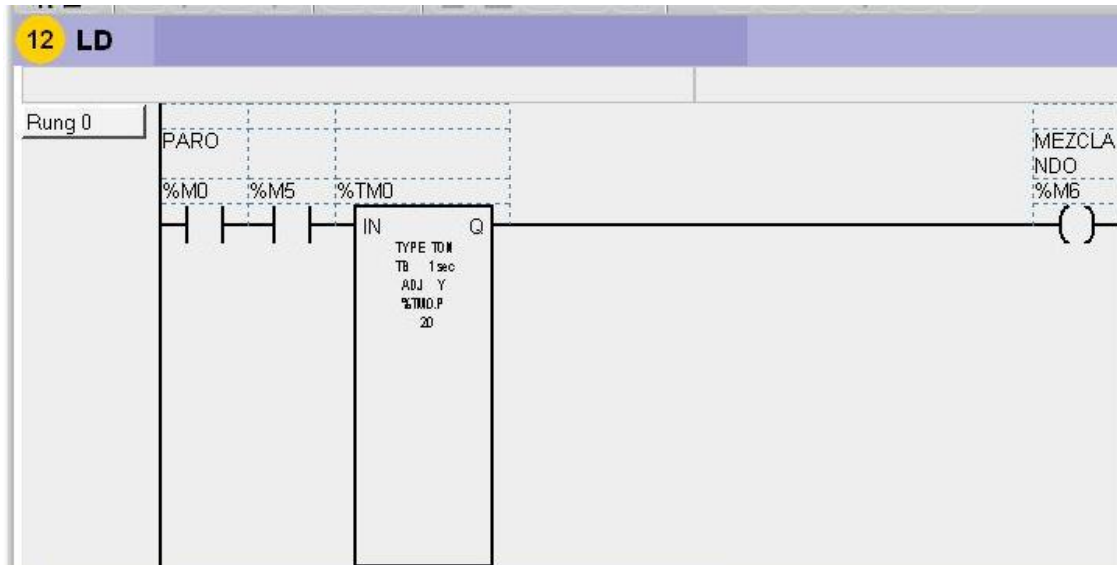


Figura 3.37 Temporizador.  
Realizado por: Javier Ortiz.

17. En este bloque de programa se activa la marca %M7 una vez que se cumplan todas las condiciones adjuntadas.

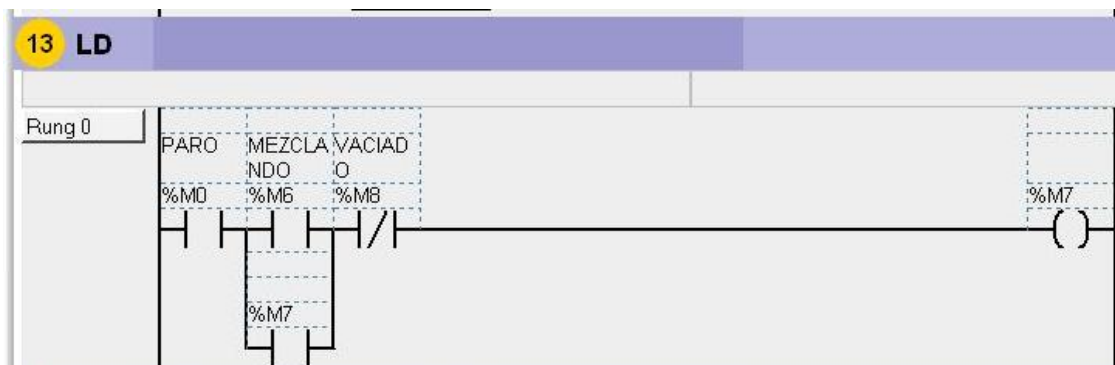


Figura 3.38 Vaciado.  
Realizado por: Javier Ortiz.

18. En este bloque de programa se activa el contacto NC %M7 y desactivará a la bobina %Q0.4 y de esta manera se activará la electroválvula. Cabe recalcar que la electroválvula se activa con GND es por eso que empieza a trabajar una vez que se desactiva la bobina %Q0.4.

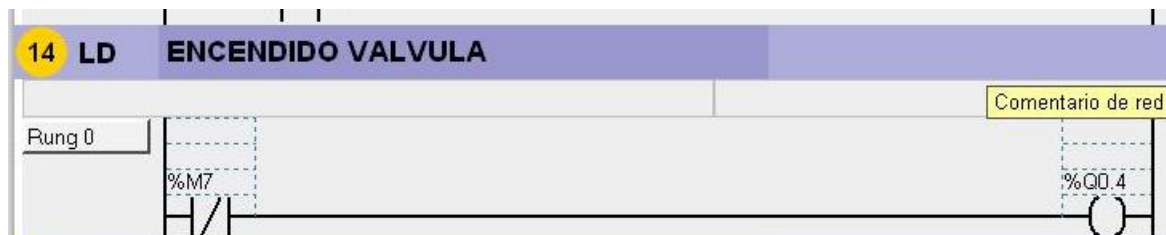


Figura 3.39 Encendido de la válvula.  
Realizado por: Javier Ortiz.

19. En el bloque de programación N°15 se realiza una comparación entre el valor almacenado en la memoria %MD4 y el valor predeterminado en la programación, en el cual una vez que se realiza 5 veces el ciclo de vaciado se activará la marca %M8 y de esta manera se cumple otra condición del bloque 16

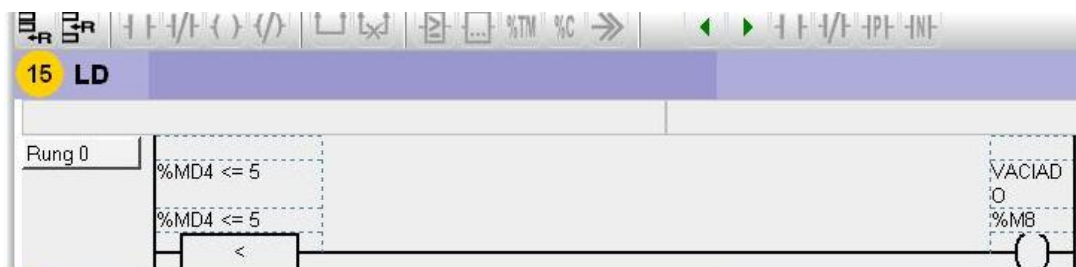


Figura 3.40 Cuadro de comparación para el vaciado.  
Realizado por: Javier Ortiz.

20. En este bloque de programación se envía un pulso cada vez que se cumplan todas las condiciones adjuntas y se activa la marca %M9. Pero la misión principal es la de hacer al sistema cíclico, es decir va a realizar el proceso de llenado, mezclado y vaciado las veces que se ha predeterminado.

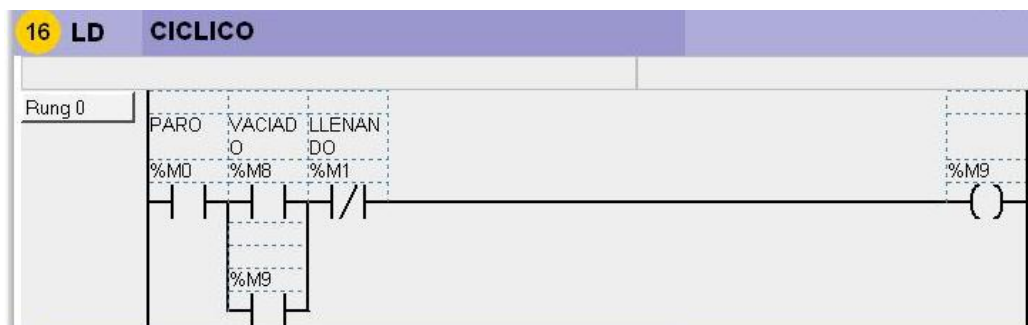


Figura 3.41 Operación cíclica del proceso.  
Realizado por: Javier Ortiz.



21. En este bloque de programa se usa un contador ascendente y en cuanto la marca %M9 genere un pulso el contador aumentará una unidad. Al haberse realizado las veces predeterminadas de llenado y vaciado del tanque, es necesario dar mantenimiento preventivo a los equipos.

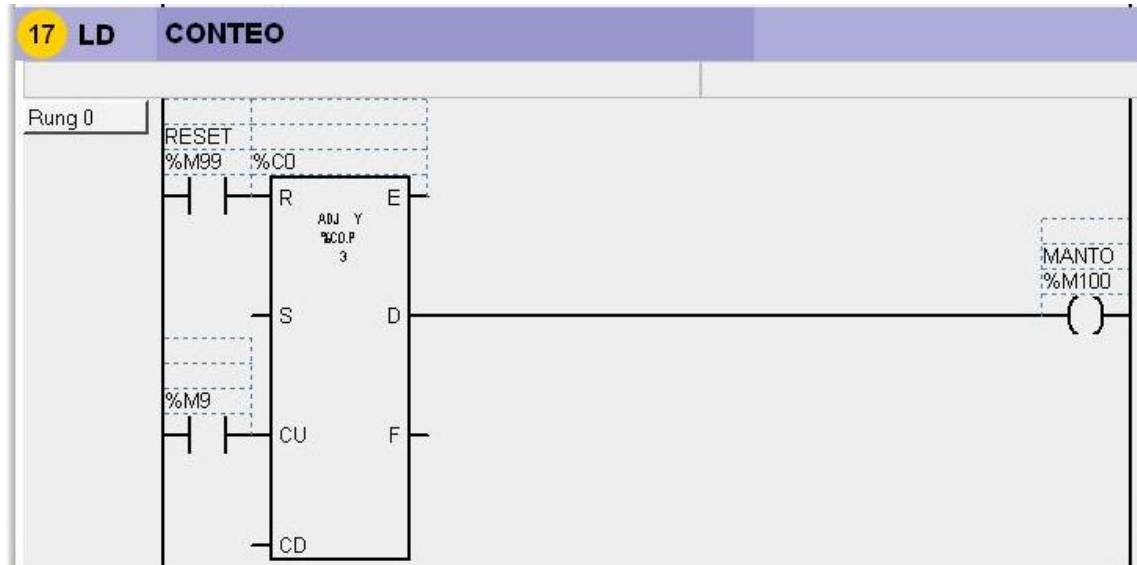


Figura 3.42 Contador.

Realizado por: Javier Ortiz.

22. Esta es la tabla de símbolos en la que constan todas las marcas y bobinas usadas y el nombre que se les dio.

En uso	Dirección	Símbolo	Comentario
<input checked="" type="checkbox"/>	%M2	CALENTANDO	
<input checked="" type="checkbox"/>	%M20	INICIO	
<input checked="" type="checkbox"/>	%M1	LLENANDO	
<input checked="" type="checkbox"/>	%M100	MANTO	
<input checked="" type="checkbox"/>	%M6	MEZCLANDO	
<input checked="" type="checkbox"/>	%M0	PARO	
<input checked="" type="checkbox"/>	%M99	RESET	
<input checked="" type="checkbox"/>	%M8	VACIADO	

Figura 3. 43 Tabla de símbolos.

Realizado por: Javier Ortiz.

### 3.4 DESARROLLO DEL PROGRAMA CON EL SOFTWARE OPC SERVER.

Este software sirve para el control del programa mediante otro software compatible.

Mediante este software es posible la visualización del desarrollo y desempeño del hardware.



Figura 3.44 Menú inicio.

Realizado por: Javier Ortiz.

1. Buscar en el menú inicio el ícono NI OPC Servers para poder abrir el software.
2. Seleccionar la pestaña file y dar clic en el ícono NEW para crear un nuevo canal por el cual se comunicará el software LabVIEW con el PLC Twido.

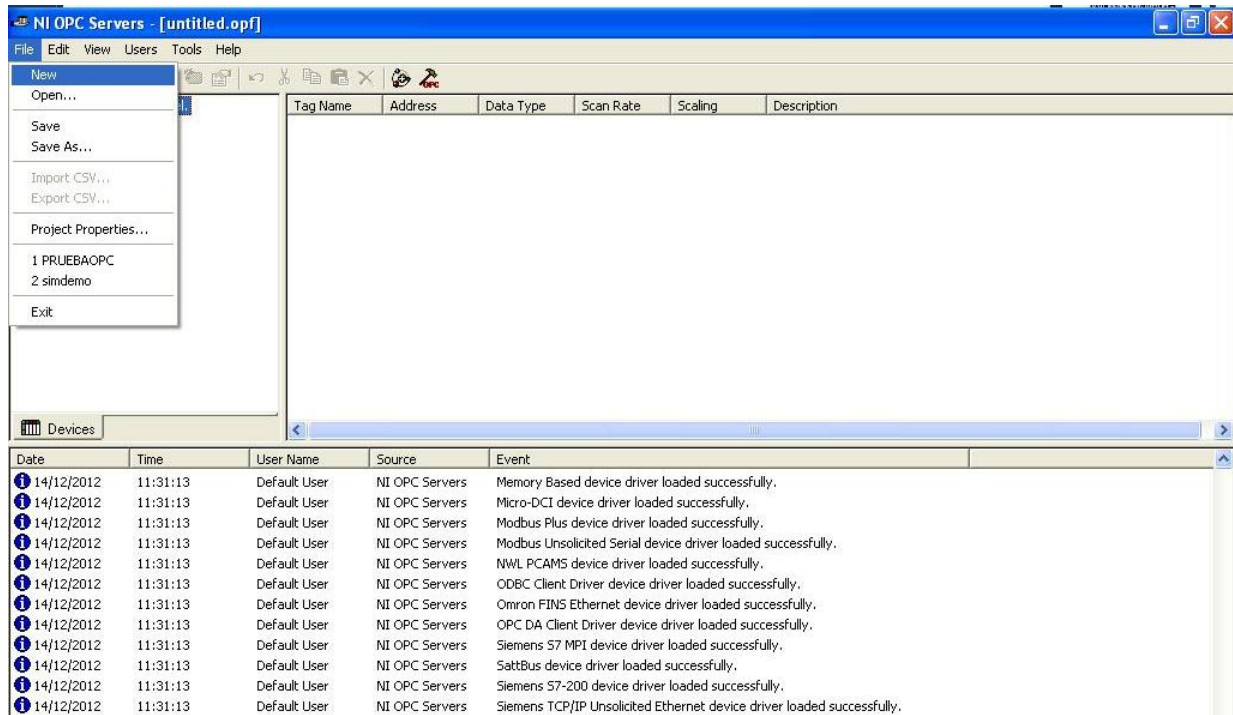


Figura 3.45 Crear nuevo Canal.

Realizado por: Javier Ortiz.

### 3. Dar un nombre al canal para poder identificarlo.

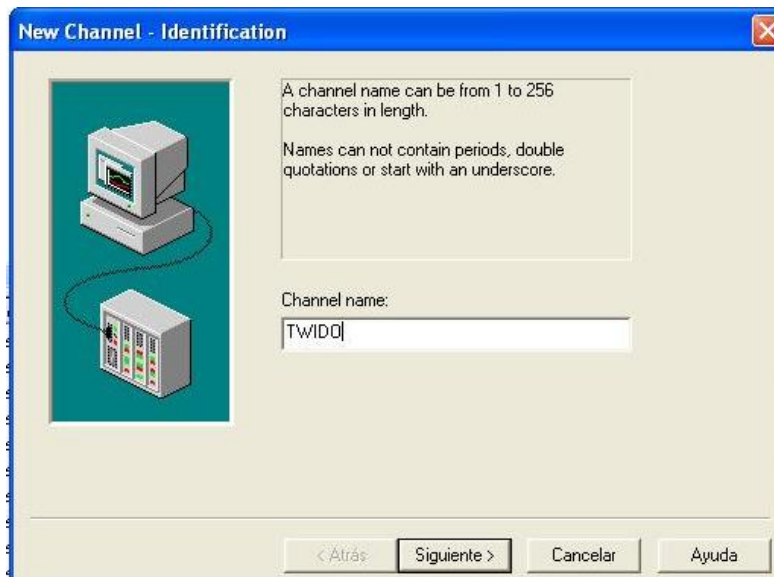


Figura 3.46 Identificación.

Realizado por: Javier Ortiz.

4. Seleccionar el protocolo de comunicación que se usará.



Figura 3.47 Protocolo de comunicación.

Realizado por: Javier Ortiz.

5. Configure la comunicación especificando el puerto por el cual se comunicará, la velocidad de transmisión en Baudios, bits de paridad y bits de parada.

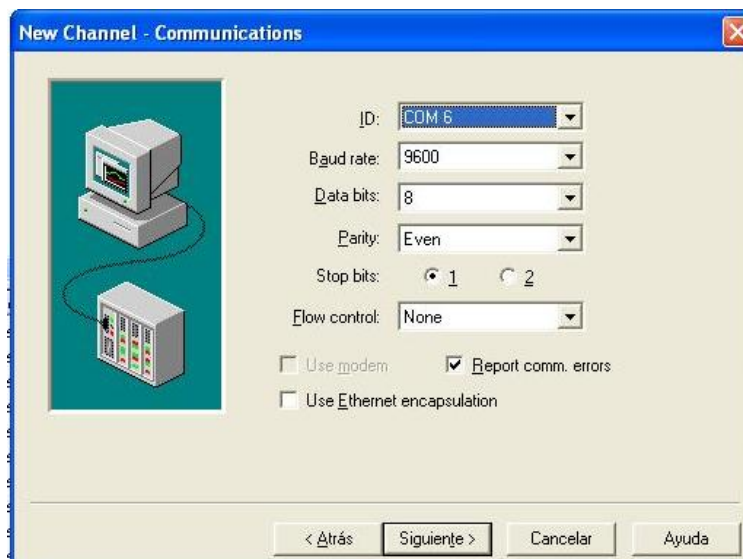


Figura 3.48 Configuración de la comunicación.

Realizado por: Javier Ortiz.

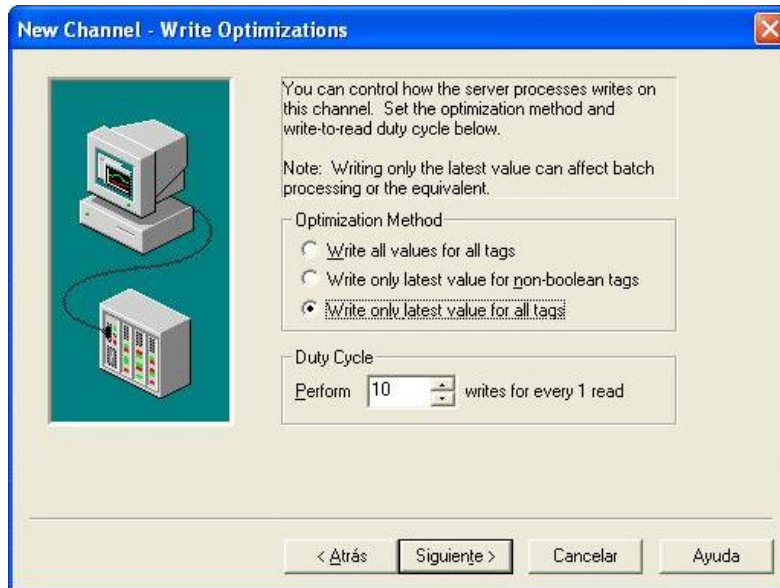


Figura 3.49 Escribir Optimizaciones.

Realizado por: Javier Ortiz.

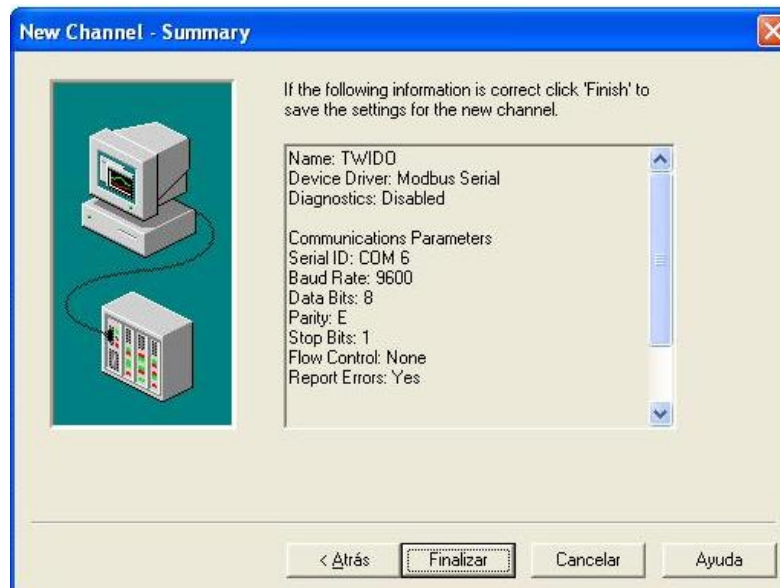


Figura 3.50 Resumen de todas las características del canal generado.

Realizado por: Javier Ortiz.

6. Identificar al nuevo dispositivo con un nombre específico.

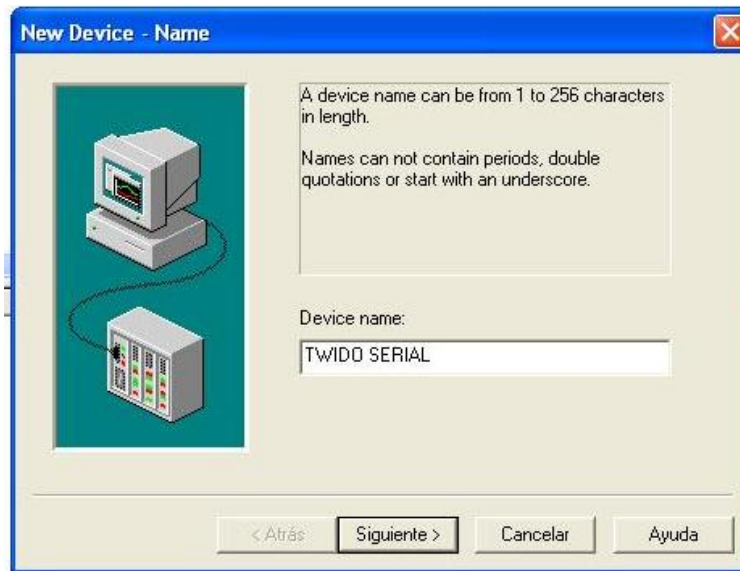


Figura 3.51 Nuevo dispositivo.  
Realizado por: Javier Ortiz.

7. Seleccionar el protocolo de comunicación que se usará.



Figura 3.52 Protocolo de comunicación.  
Realizado por: Javier Ortiz.

8. Seleccione un modelo que mejor describa al dispositivo que usted está definiendo.

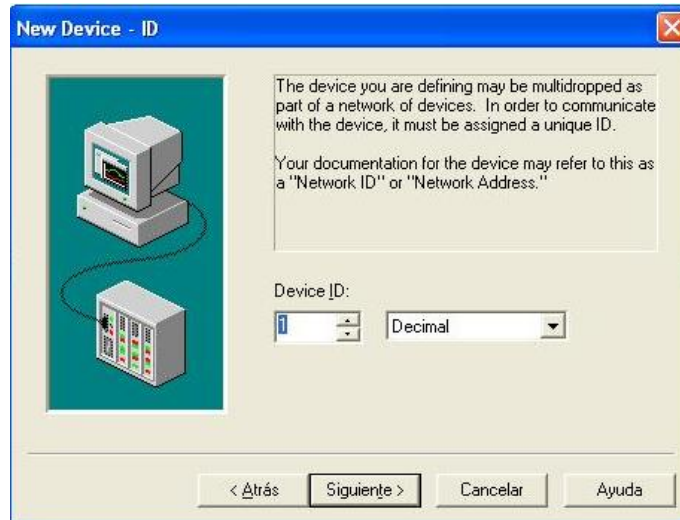


Figura 3.53 Protocolo de comunicación.

Realizado por: Javier Ortiz.

9. El dispositivo que usted está definiendo puede ser parte de una red de dispositivos. Para comunicarse con el dispositivo, se debe asignársele una única Identificación.

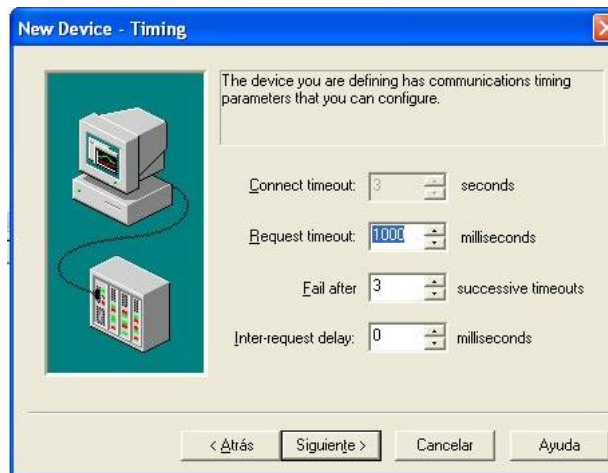


Figura 3.54 Parámetros del tiempo en la comunicación.

Realizado por: Javier Ortiz.



Figura 3.55 Auto-Negación.

Realizado por: Javier Ortiz.

10. Usted puede degradar un dispositivo para un número específico en los fracasos de comunicaciones. Durante este tiempo no se enviará ningún requerimiento de lectura al dispositivo. Al degradar un dispositivo fallado, se previene tener pendiente la comunicación con otros dispositivos en la red.



Figura 3.56 Creación de una base de datos.

Realizado por: Javier Ortiz.



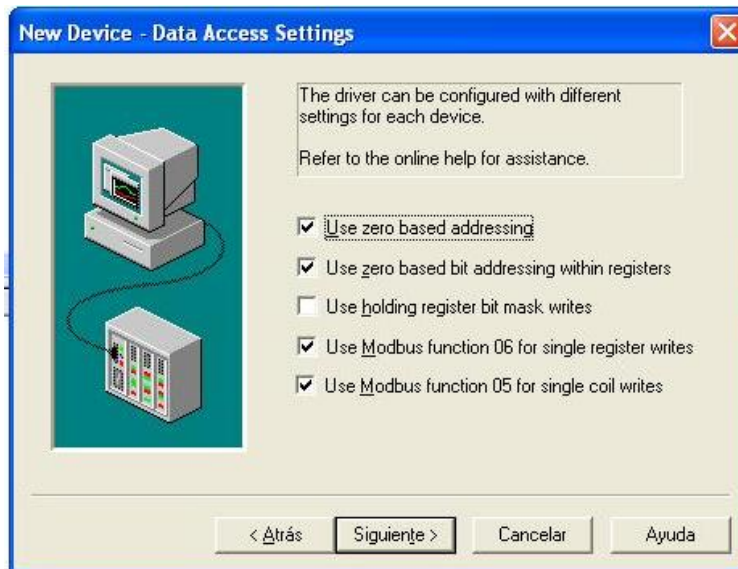


Figura 3.57 Configuración de los datos de acceso.

Realizado por: Javier Ortiz

**11.** Configurar los datos de acceso, es decir cada canal puede ser configurado para que se comunique con un sólo dispositivo a la vez.

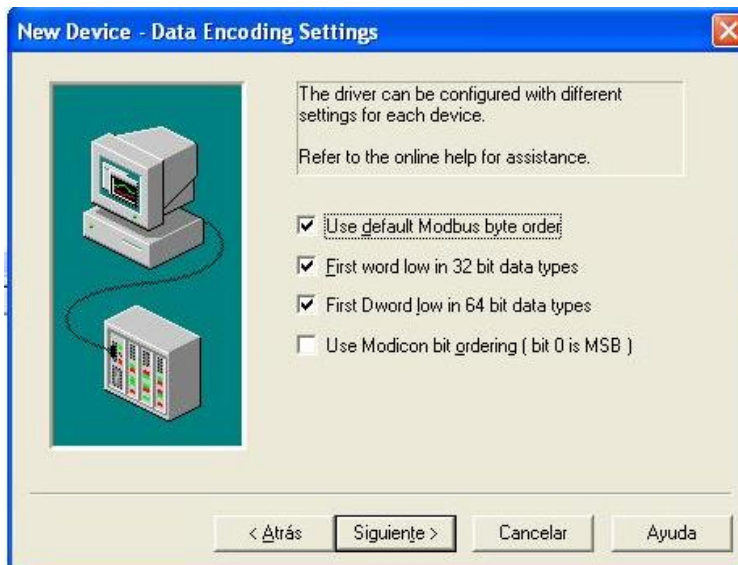


Figura 3.58 Configuración de los datos a usar.

Realizado por: Javier Ortiz.

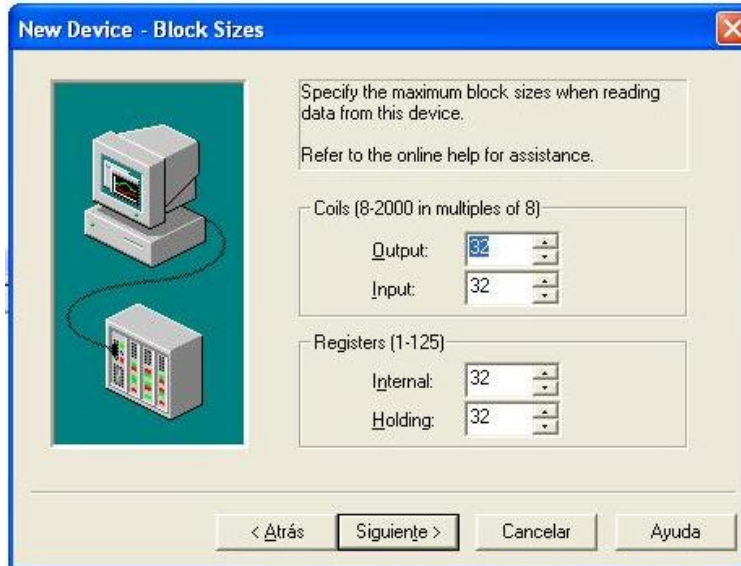


Figura 3.59 Tipos de Bloques.  
Realizado por: Javier Ortiz.

12. Especificar el número máximos de bloques originarios del programa, que puede leer.



Figura 3.60 Configuración para Importar una Variable.  
Realizado por: Javier Ortiz.

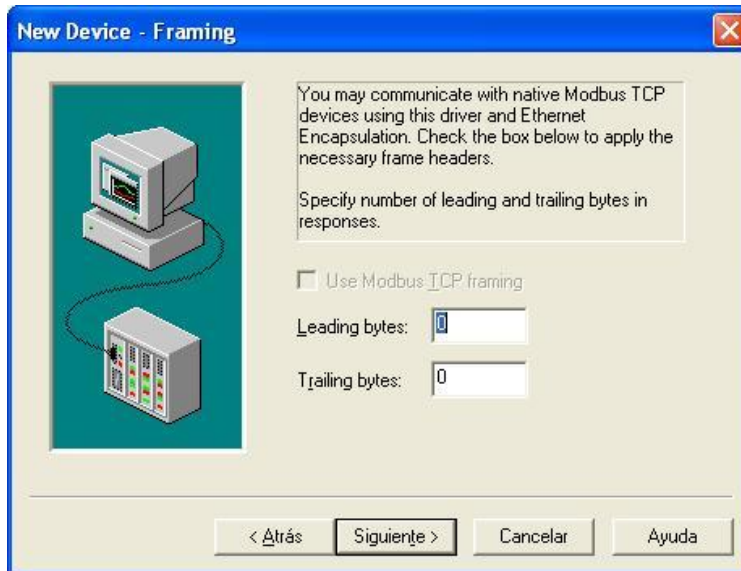


Figura 3.61 Ideando.

Realizado por: Javier Ortiz.

13. Se puede realizar la comunicación mediante el protocolo Modbus TCP usando un dispositivo Ethernet así que hay que determinar bytes que se enviarán y los que se recibirán.



Figura 3.62 Tratamiento de Errores.

Realizado por: Javier Ortiz.

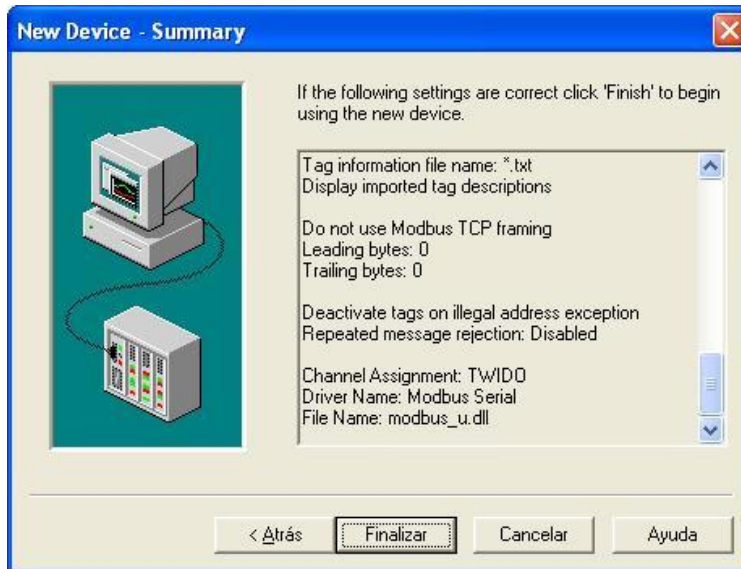


Figura 3.63 Resumen de la configuración del dispositivo.

Realizado por: Javier Ortiz.



Figura 3.64 Propiedades de los Tags o etiquetas.

Realizado por: Javier Ortiz.

14. En esta ventana se ingresan datos de un Tag, como el nombre con el que se identifica, la dirección y una descripción. El tipo de dato, por ejemplo un dato Booleano, Chart, Byte, Word, Dword, decimal, etc. Y la función de la etiqueta, es

decir si se la usará para leer, escribir o ambas. Y cada qué tiempo escaneará el programa para recibir y enviar datos.

15. Proceder a comunicarse con el PLC Twido. Dar clic en el ícono Transferencia PC a Autómata, identificando el puerto por el cual se realizará la comunicación.

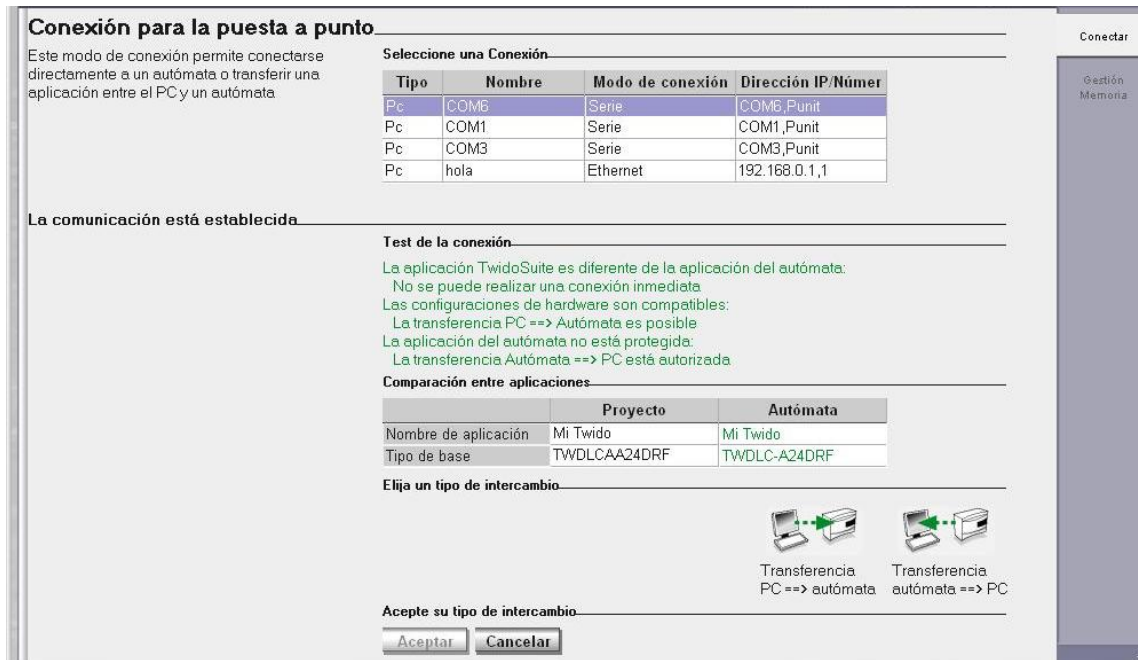


Figura 3.65 Establecer comunicación.

Realizado por: Javier Ortiz.

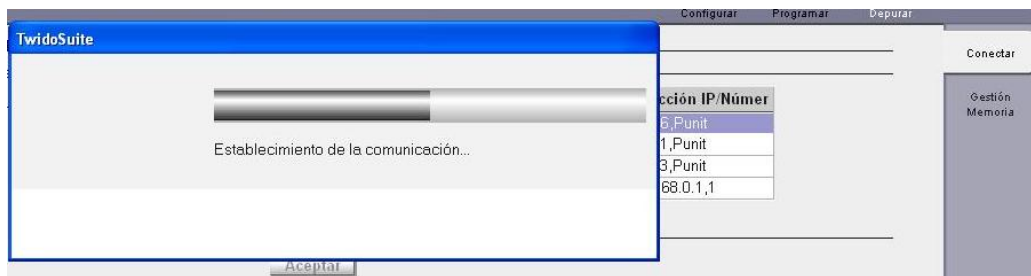


Figura 3.66 Cuadro de comunicación.

Realizado por: Javier Ortiz.

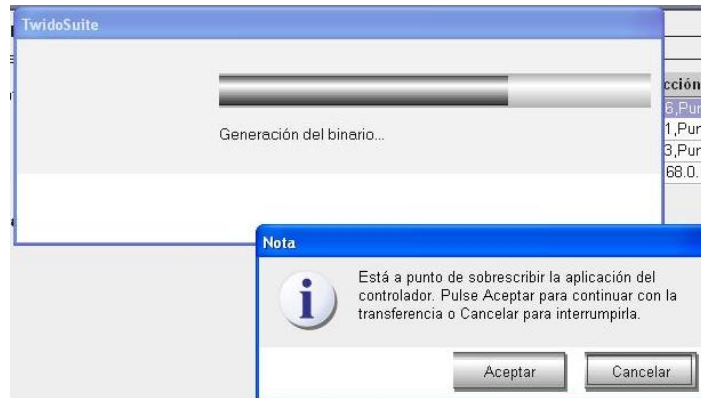


Figura 3.67 Cuadro de información de la comunicación.

Realizado por: Javier Ortiz.



Figura 3.68 Cuadro de inicio.

Realizado por: Javier Ortiz.

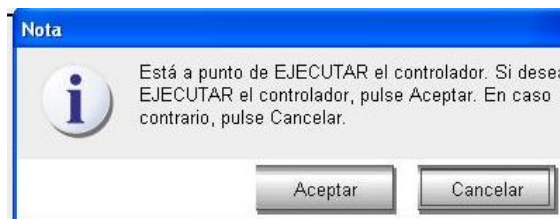


Figura 3.69 Aviso de ejecución del programa.

Realizado por: Javier Ortiz.

### 3.3.4 DESARROLLO DEL PROGRAMA EN EL SOFTWARE LABVIEW.

Continuando con el desarrollo del proyecto. Existe la necesidad de visualizar los parámetros en los que se encuentra operando el sistema.

Es por eso que se incorpora un software que permita la visualización gráfica del proceso para poder optimizar el control y monitoreo del sistema.

A continuación se explicará cómo se realiza la programación en LabVIEW.

Primero se adjuntan todos los elementos que se usarán para el control y monitoreo del proceso como botones, pulsadores, un termómetro, un tanque, controladores e indicadores.

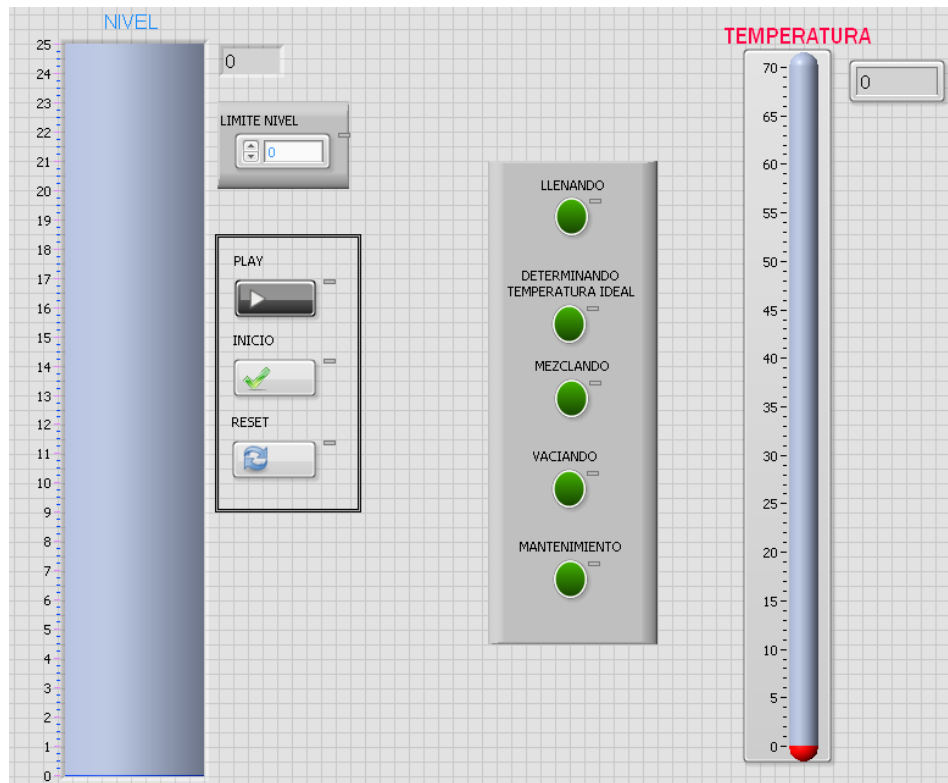


Figura 3.70 LabVIEW Front Panel.

Realizado por: Javier Ortiz.

En el diagrama de bloques se integran los íconos que representan a los elementos adjuntados en el panel de frontal adicionalmente se relacionan de acuerdo a las necesidades.

También se adjuntan los procesos aritméticos que sirven para un escalamiento interno y corrección de errores.

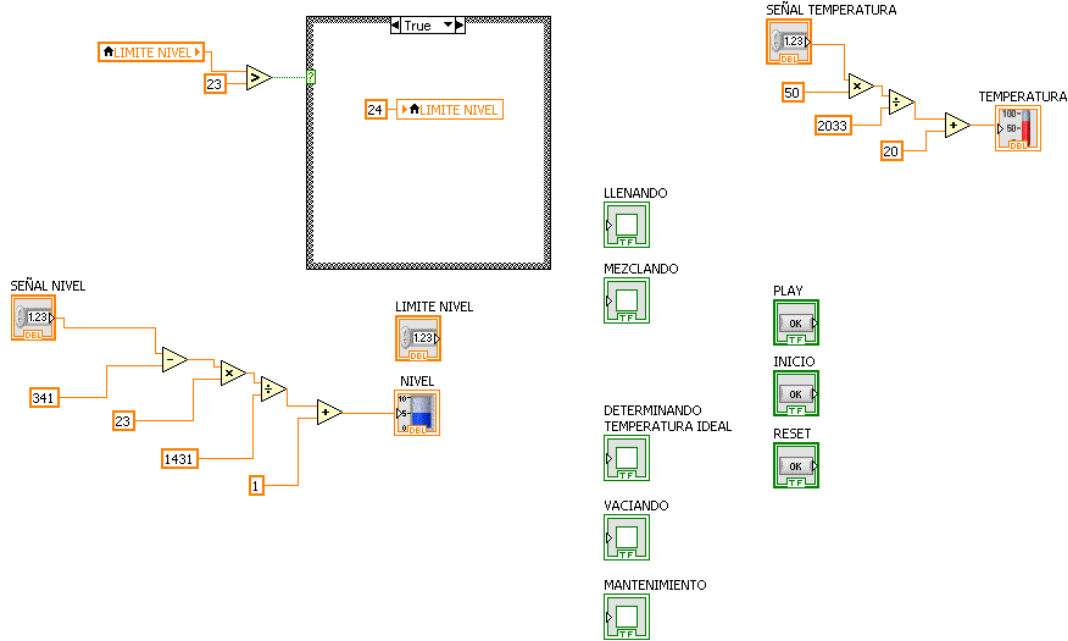


Figura 3.71 LabVIEW Block Diagram.

Realizado por: Javier Ortiz.

Para poder controlar el PLC mediante LabVIEW es necesario relacionar a los controles de LabVIEW con las variables de Twido Suite mediante el software OPC Server.

A continuación se detalla el procedimiento.

Primero se da clic derecho sobre el controlador que está en el Panel Frontal. Y seleccionar propiedades.



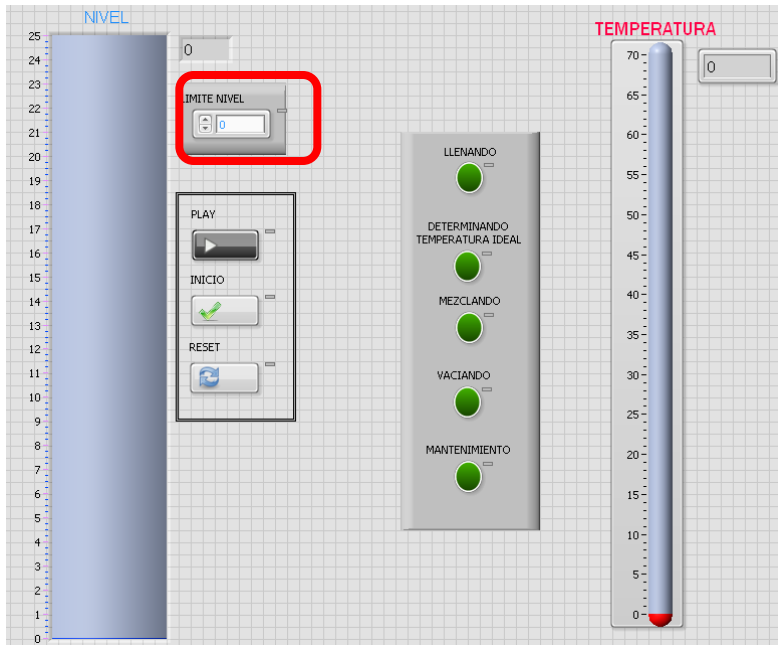


Figura 3.72 Propiedades del control.

Realizado por: Javier Ortiz.

Seleccionar la pestaña Data Binding, desplegar el menú Browse y seleccionar el ícono DSTP Server.

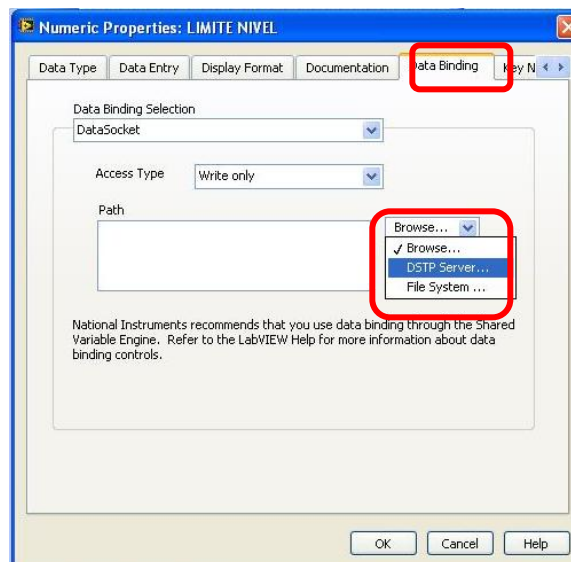


Figura 3.73 DSTP Server.

Realizado por: Javier Ortiz.

Desplegar la opción National Instruments.NIOPCServer y seleccionar la carpeta que tenga el nombre del dispositivo creado anteriormente.

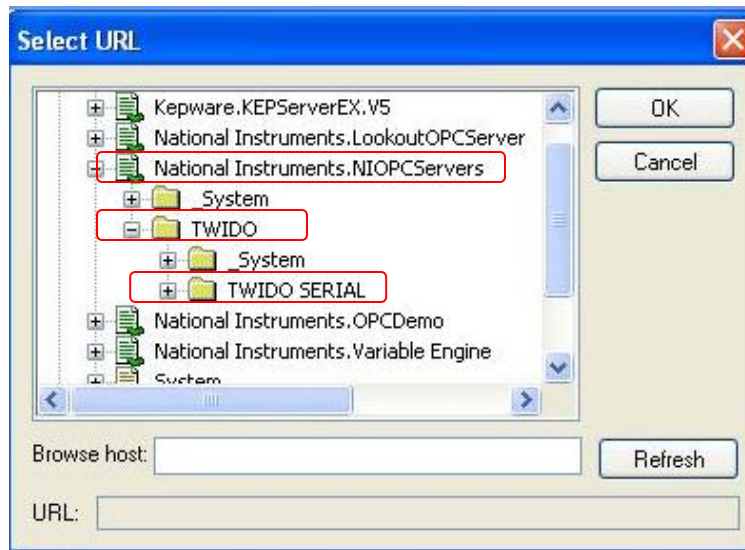


Figura 3.74 Buscar la dirección de almacenamiento.

Realizado por: Javier Ortiz.

Seleccionar el nombre del Tag correspondiente al control numérico seleccionado en LabVIEW.



Figura 3.75 Seleccionar la variable que corresponde al control.

Realizado por: Javier Ortiz.

Verificar que los datos seleccionados sean correctos. Seleccionar si es de lectura, sólo escritura o lectura/escritura

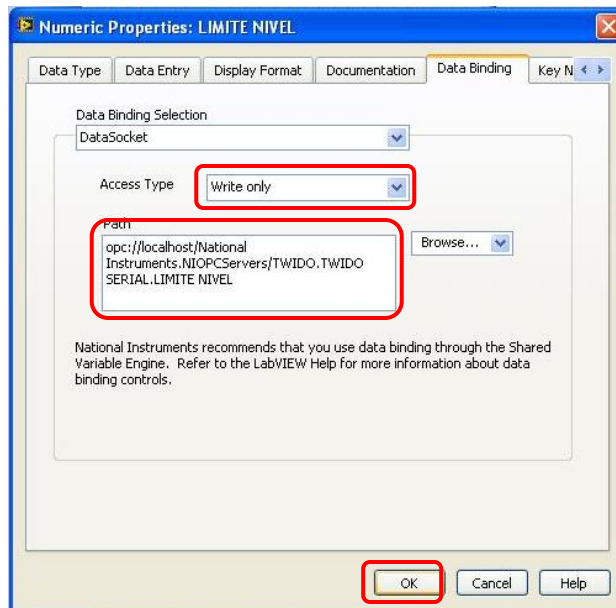


Figura 3.76 Verificar la dirección y características.

Realizado por: Javier Ortiz.

El programa OPC Server permite crear variables o Tags cuya función principal es cargar las direcciones del PLC o enviar comandos hacia el PLC, para ello se necesita escribir las direcciones del PLC para cada Tag, es primordial mencionar que la sintaxis de direcciones para Modbus es diferente a la que usa el PLC:

Para la elaboración de los Tags hay que tener en cuenta:

**Tabla 3.1 Sintaxis en Modbus.**

DESCRIPCIÓN	PLC	MODBUS
Palabra de memoria	%MWn	40000 (n+1)
Bits de memoria	%Mn	00000 (n+1)
Entradas digitales	%In	10000 (n+1)

Elaborado por: Javier Ortiz

### 3.3.5 INSTRUCTIVO DE FUNCIONAMIENTO.

Para el correcto desempeño del sistema HMI es necesario que antes de implementar el circuito para el control y monitoreo de la estación de nivel en el Laboratorio de Instrumentación Virtual lea este instructivo para que tome en cuenta su funcionamiento y las conexiones de los equipos usados

#### 3.3.5.1 Voltajes de cada elemento.

De acuerdo a las especificaciones del fabricante cada elemento necesita una determinada cantidad de energía eléctrica para su correcto funcionamiento, los rangos de voltaje están explicados en la tabla 3.2

Tabla 3.2 Voltajes.

ELEMENTO	VOLTAJE INDICADO POR EL FABRICANTE
PLC	V. min=110V V. max=125V
MÓDULO DE E/S ANALÓGICAS	V. min=23V V. max=25V
ESTACION DE NIVEL	V. min=220V V. max=240V

Elaborado por: Javier Ortiz

#### 3.3.5.2 Voltajes de los sensores de la estación de nivel.

El sensor de nivel genera una señal de corriente de 4-20mA así que es necesario transformar esa señal de corriente a una señal de voltaje conectando una carga de 250Ω.

El sensor de temperatura genera un voltaje de 0V a 5V.

A cada centímetro cúbico que aumente el nivel del líquido le corresponde un voltaje determinado y una cantidad proporcional de bits que se visualiza en el software Twido Suite.

Y a cada grado centígrado que aumente la temperatura del líquido le corresponde una cantidad de voltaje y una cantidad proporcional de bits que se visualiza en el software Twido Suite.

Estos parámetros se detallan a continuación. Cabe recalcar que los parámetros pueden variar debido a posibles descalibraciones que sufre el sensor de nivel.

**Tabla 3.3 Voltajes y números de bits del sensor de nivel.**

<b>CENTÍMETROS</b>	<b>VOLTAJE DE SALIDA DEL SENSOR</b>	<b>CANTIDAD DE BITS</b>
1	1.07V	351
5	1.81V	694
10	2.77V	1053
15	3,47V	1397
20	4,27V	1637
24	4.87V	1784

Elaborado por: Javier Ortiz

**Tabla 3.4 Voltajes y numero de bits del sensor de temperatura.**

<b>TEMPERATURA</b>	<b>VOLTAJE DE SALIDA DEL SENSOR</b>	<b>BITS</b>
20	0V	0
30	1V	407
40	2V	811
50	3V	1227
60	4V	1723
70	5V	2037

Elaborado por: Javier Ortiz

### **3.3.5.3 Prueba de Software.**

Para realizar las pruebas es necesario contar con todos los programas necesarios y correctamente instalados en la PC, si no se cuentan con dichos programas proceder a instalarlos; los programas que se usan son:

- ✓ Twido Suite.
- ✓ LabVIEW.
- ✓ OPC Server.

Para saber si existe una buena adquisición de datos y un correcto control hay que poner RUN al PLC y verificar si en el software Twido Suite se representan algunas cantidades de acuerdo a las variaciones de las variables físicas. En el caso de no representarse ningún valor hay que desconectar al PLC y volver a verificar los voltajes en las salidas de los sensores, probar continuidad de los cables, o en el software comprobar si las memorias en las que se almacenan los datos son las indicadas.

### **3.3.5.4 Conexión de equipos.**

Para hacer de todos los elementos un sólo conjunto que funcione armoniosamente se debe realizar correctamente las conexiones, a continuación se detalla paso a paso el procedimiento a seguir. Como ejemplo práctico se realiza la simulación de un proceso industrial, el cual consiste en el proceso previo al que debe someterse el agua antes de ingresar a un caldero para ser convertida en vapor.

El proceso consiste en que una cierta cantidad de agua debe estar a una temperatura determinada; se debe adicionar químicos que limpiarán al caldero por dentro, estos aditivos deben ser disueltos en el agua es por eso que es necesario mezclar todos los componentes.

1. Energizar la estación de nivel permitiendo que se accione el sensor de nivel, el sensor de temperatura, la bomba y la electroválvula. Mover el selector power a la posición ON.

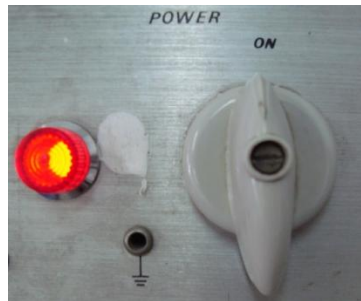


Figura 3.77 Switch ON/OFF.  
Realizado por: Javier Ortiz.

2. Verificar que los voltajes de salida de los sensores se encuentren dentro del rango especificado en las tablas 3.3 y 3.4. Encender la bomba y llenar el tanque hasta  $1\text{cm}^3$ , medir con el voltímetro el voltaje en la bornera de conexiones transductor de nivel y comparar con el valor dado en la tabla 3.3; llenar el tanque hasta los  $5\text{cm}^3$ , medir con el voltímetro el voltaje en la bornera de conexiones transductor de nivel y comparar con el valor dado en la tabla 3.3; y así para todos los niveles de líquido.
3. Para verificar los datos de la temperatura, encender la niquelina y esperar que el líquido aumente su temperatura hasta los 20 grados y medir el voltaje en la bornera de la estación de nivel, en el terminal señal analógica de temperatura y GND. Incrementar la temperatura hasta los 30 grados y medir el voltaje en la bornera de la estación de nivel, en el terminal “señal analógica de temperatura” y GND.
4. Identificar dónde se encuentran ubicadas las entradas y salidas digitales del PLC.



Figura 3.78 Entradas digitales.  
Realizado por: Javier Ortiz.

5. Identificar los terminales de alimentación (L y N).



Figura 3.79 terminales de alimentación.  
Realizado por: Javier Ortiz.

6. Descubrir la ranura en la cual se montará el módulo de E/S analógicas.

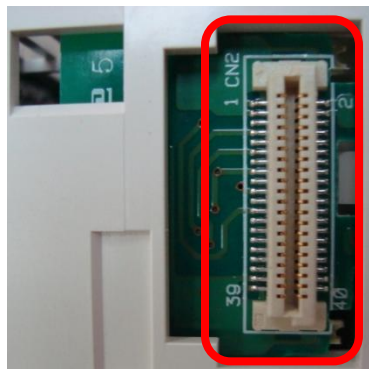


Figura 3.80 Ranura de expansión.  
Realizado por: Javier Ortiz.

7. Apagar la estación de nivel para continuar con las conexiones respectivas, poner el selector en la posición OFF.
8. Identificar la línea y neutro de la alimentación de 120V para proceder a conectar a los terminales L y N del PLC.





Figura 3.81 Conexión L y N.  
Realizado por: Javier Ortiz.

9. Conectar el Terminal COM 0 a 5Vcc en la bornera de la estación de nivel para que al activarse las salidas 0, 1, 2 y 3 alimenten con 5Vcc al elemento determinado.

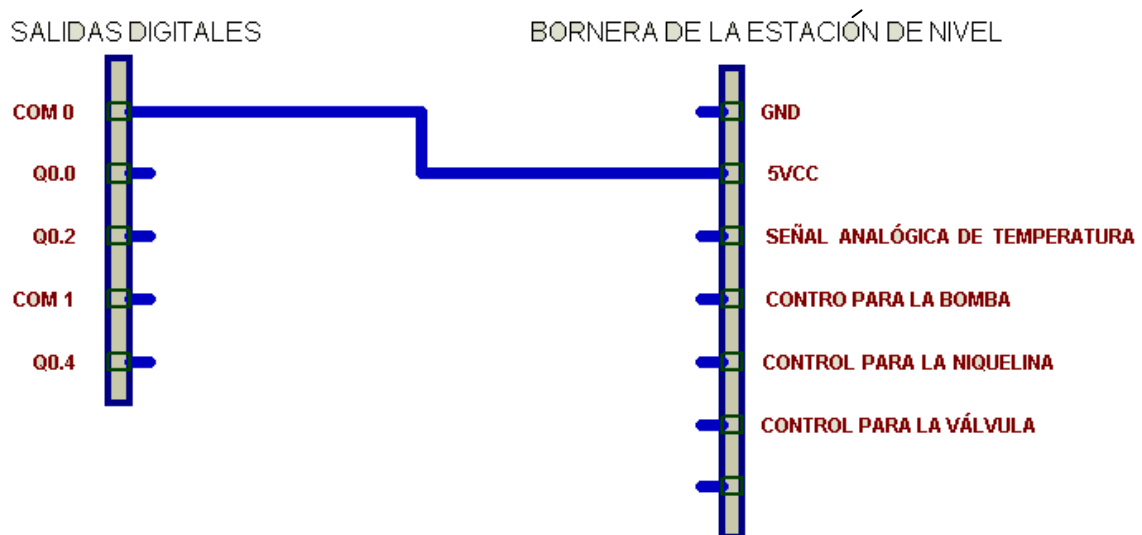


Figura 3.82 Conexión COM 0.  
Realizado por: Javier Ortiz.

10. Conectar en la salida Q0.0 el terminal de alimentación de la bomba de la bornera de la estación de nivel.

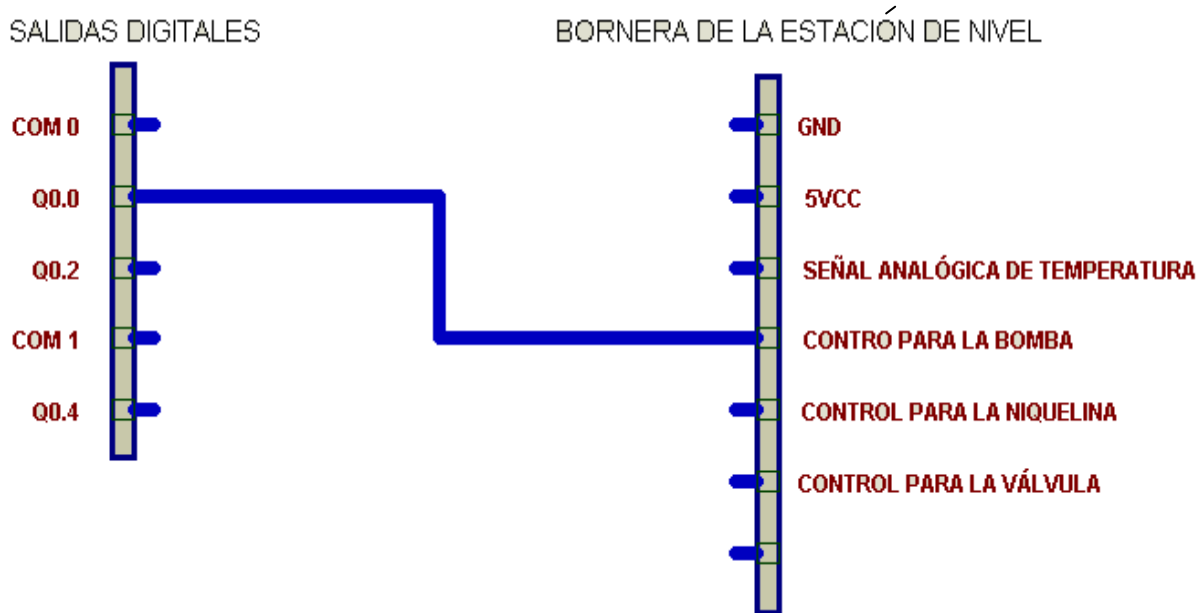


Figura 3.83 Conexión Bomba.  
Realizado por: Javier Ortiz.

11. En la salida Q0.2 conectar el terminal de la niquelina de la bornera de la estación de nivel.

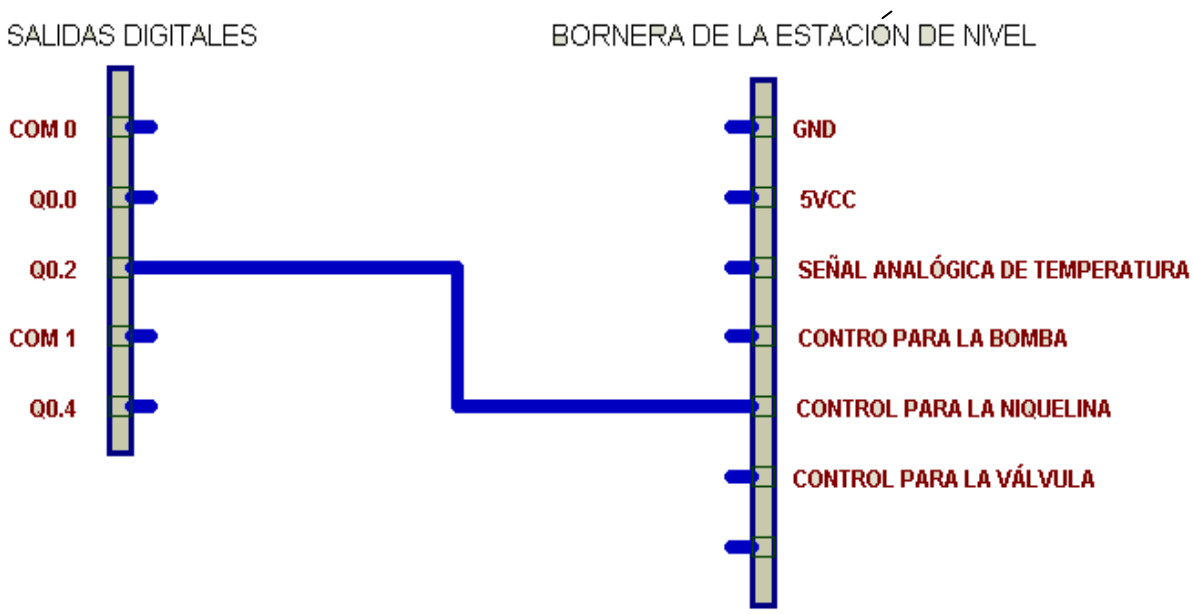


Figura 3.84 Conexión niquelina.  
Realizado por: Javier Ortiz.

12. Conectar el terminal COM 1 a GND de la bornera de la estación de nivel para que al activarse las salidas 4, 5, 6 y 7 proporcionen un nivel de voltaje bajo o tierra.

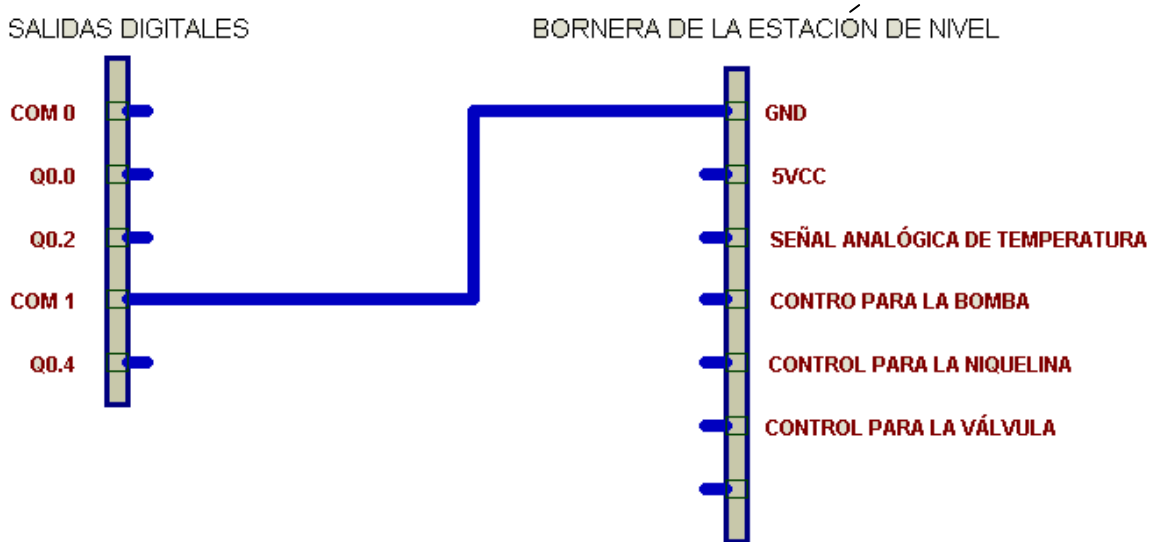


Figura 3.85 Conexión Com 1.  
Realizado por: Javier Ortiz.

13. Conectar a la salida Q0.4 el terminal de la válvula de la bornera de la estación de nivel. Cabe recalcar que la válvula se activa con GND.

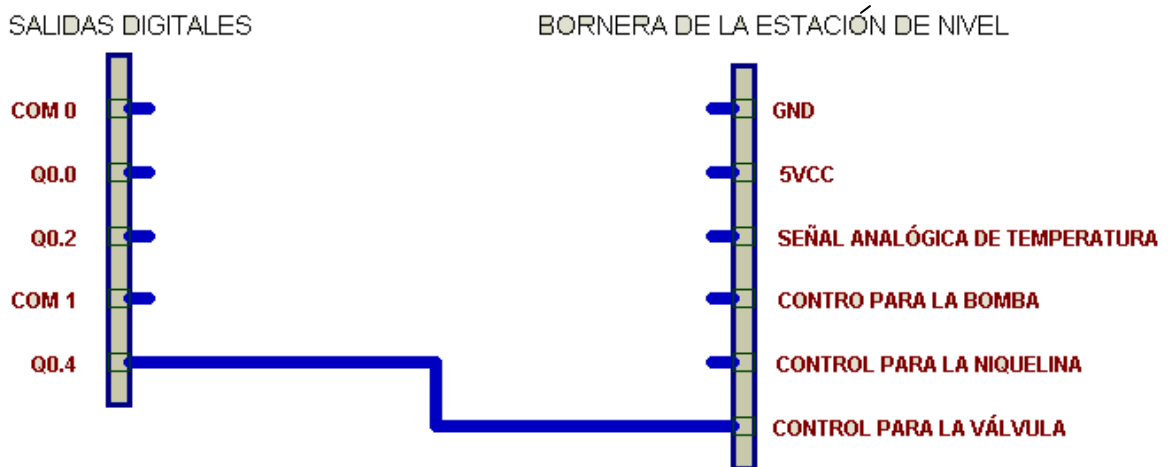


Figura 3.86 Conexión válvula.  
Realizado por: Javier Ortiz.

14. Conectar el módulo de E/S analógicas en su ranura correspondiente.



Figura 3.87 Módulo E/S analógicas.

Realizado por: Javier Ortiz.

15. Conectar la señal del sensor de nivel a través del acondicionador de señal al terminal  $\pm$ IN0.

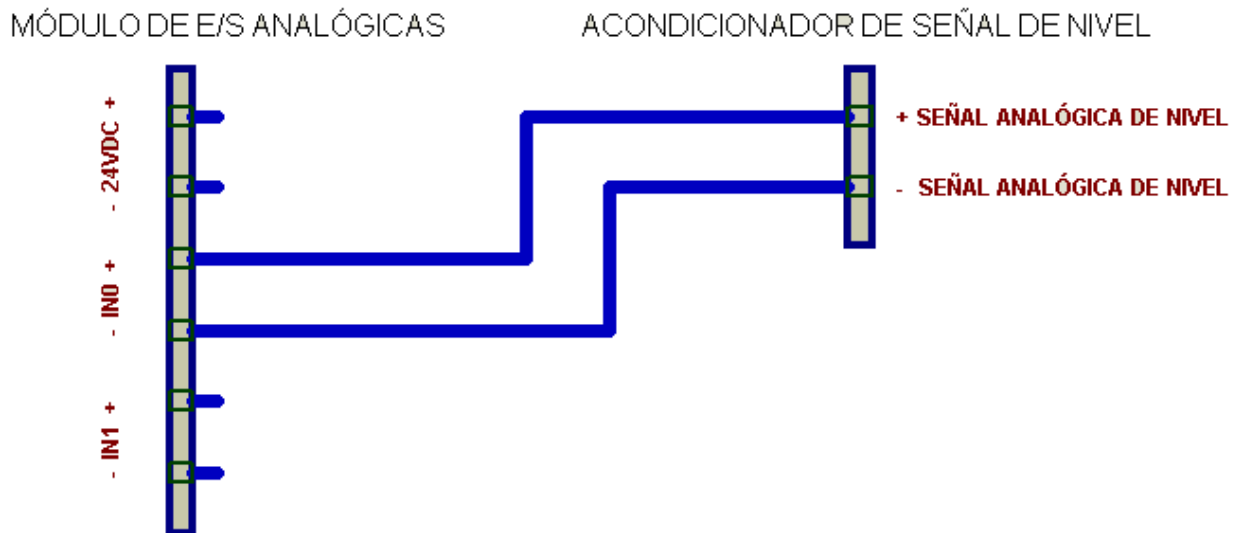


Figura 3.88 Conexión señal de nivel.

Realizado por: Javier Ortiz.

16. La señal de temperatura de la bornera de la estación de nivel se conecta al terminal  $\pm$ IN1.

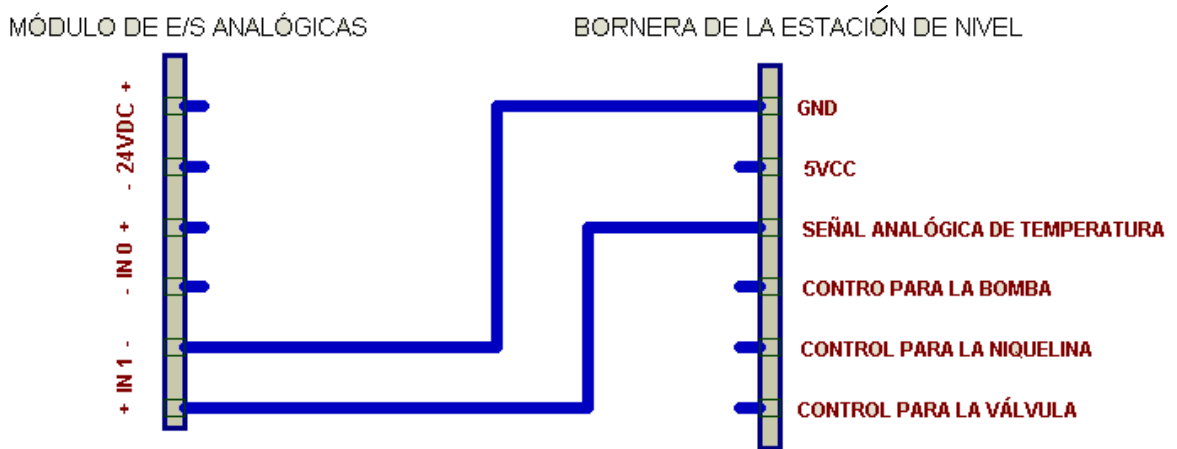


Figura 3.89 Conexión señal de temperatura.

Realizado por: Javier Ortiz.

17. Conectar la fuente de alimentación de 24Vcc a los terminales de alimentación del módulo.

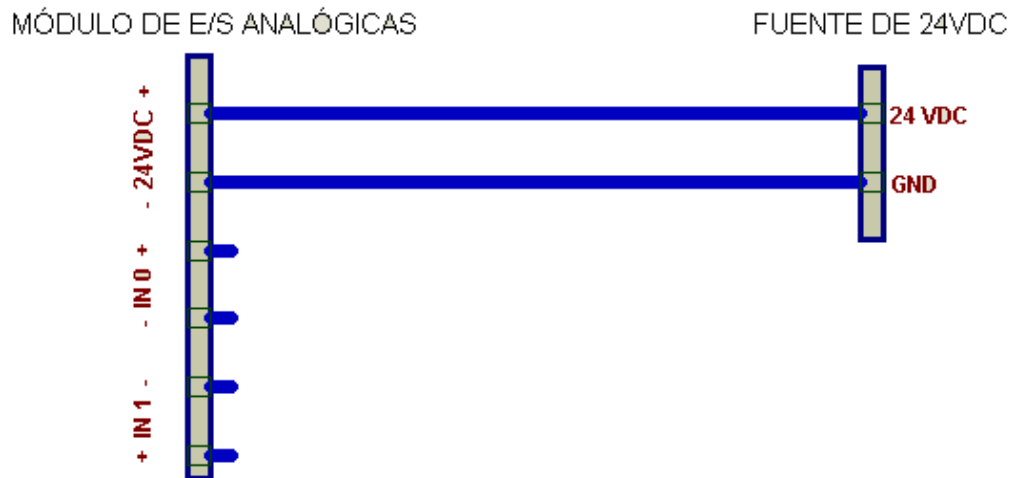


Figura 3.90 Alimentación del módulo.

Realizado por: Javier Ortiz.

18. Conectar el PLC con el computador usando el cable de transmisión para programar y monitorear al autómeta.

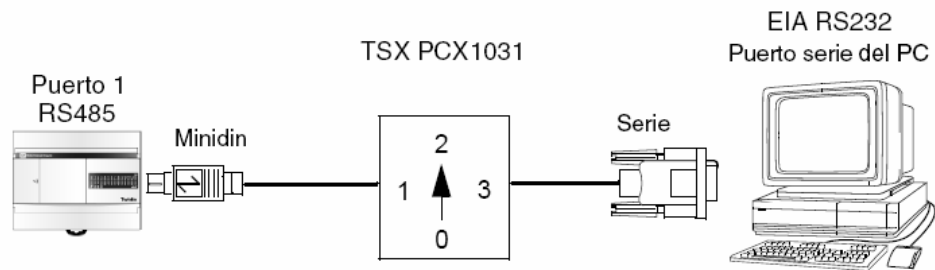


Figura 3.91 Comunicación.  
Realizado por: Javier Ortiz.

19. Una vez realizadas las conexiones correctamente, proceder a energizar todos los elementos.



Figura 3.92 Interruptor ON/OFF.  
Realizado por: Javier Ortiz.

20. Abrir el software Twido Suite y dar clic en el icono programación para ingresar al programa.

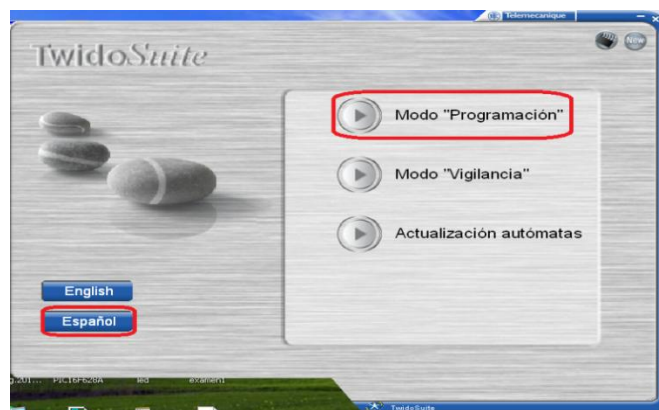


Figura 3.93 Modo Programación.  
Realizado por: Javier Ortiz.

21. Cargar el programa en Twido Suite (ver anexo A) y grabar en el PLC. Dar clic en abrir un nuevo proyecto.

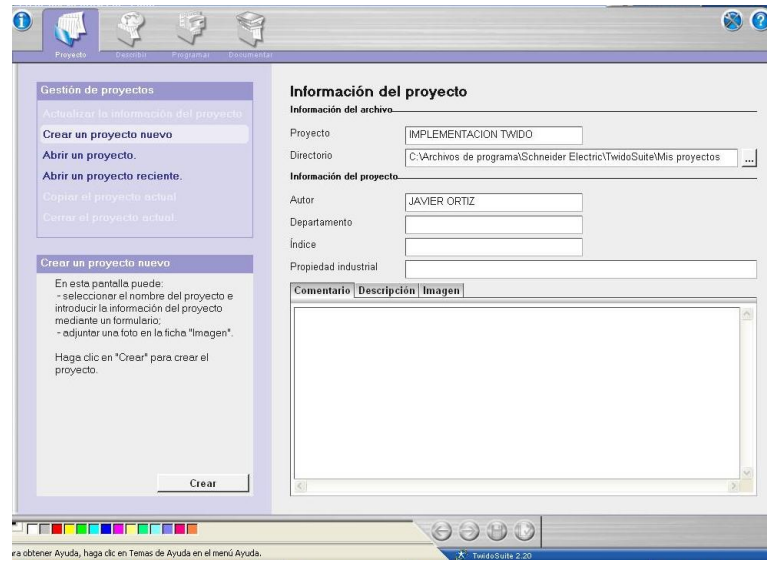


Figura 3.94 Abrir un nuevo proyecto.

Realizado por: Javier Ortiz.

22. Seleccione los íconos programar para que se despliegue la ventana donde se ingresarán los bloques del programa.

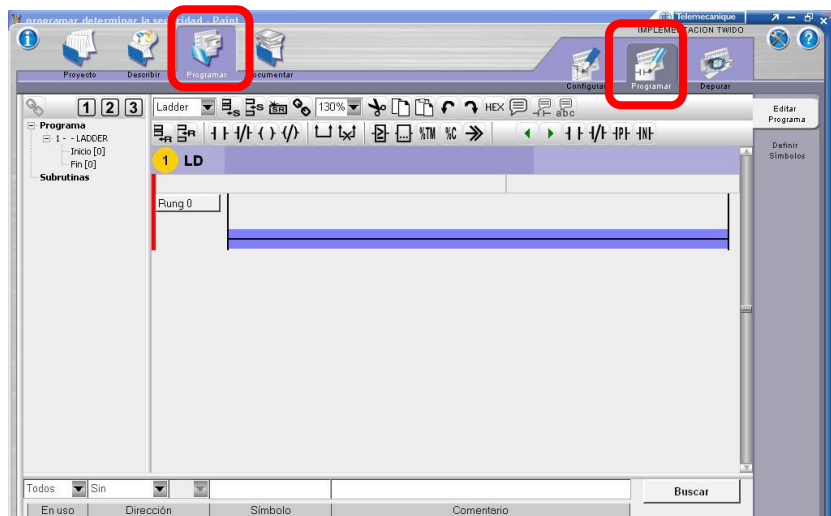


Figura 3.95 Bloques de programación.

Realizado por: Javier Ortiz.

23. Conectarse al PLC mediante el software Twido Suite para verificar la adquisición de los datos de los sensores de nivel o temperatura y verificar que se encuentran dentro de los rangos correspondientes. Seleccione el ícono depurar en la pestaña conectar y dar clic en aceptar.



Figura 3.96 Ventana de comunicación.

Realizado por: Javier Ortiz.

24. Poner en RUN al PLC haciendo clic en play y aceptar, para poder verificar los datos adquiridos y en caso de haber alguna descalibración en la señal de los sensores, realizar acciones de corrección para eliminar este problema.



Figura 3.97 Ventana de ejecución.

Realizado por: Javier Ortiz.

25. Una vez que se analizaron los datos de entrada, proceder a verificar la lógica de programación, que todos los valores se almacenen en las memorias correspondientes y que las marcas se activen con normalidad.





Figura 3.98 Bloques de programación.

Realizado por: Javier Ortiz.

26. A continuación hacer clic en el icono desconectar en el software Twido Suite para cortar la comunicación con el PLC y comunicarse usando el software LabVIEW.

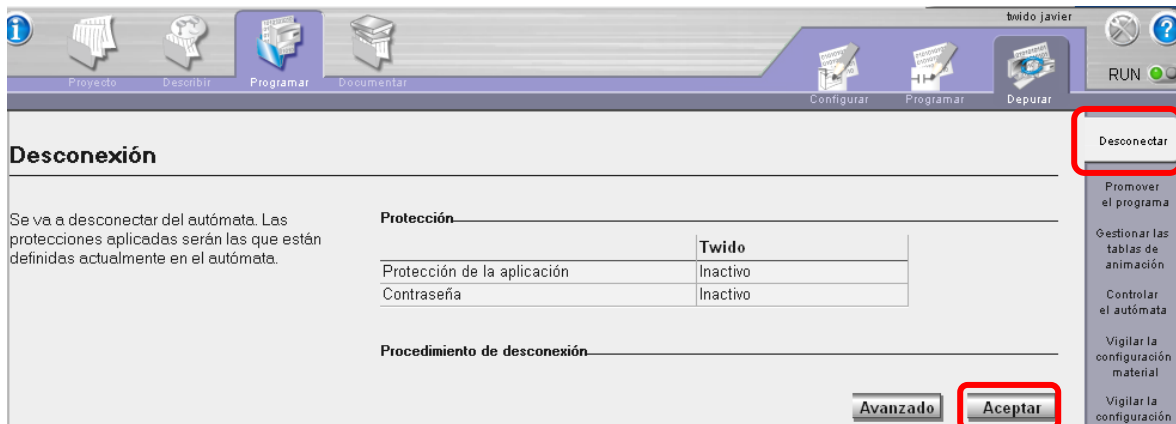


Figura 3.99 Desconectar.

Realizado por: Javier Ortiz.

27. Para el control, monitoreo y visualización de la estación de nivel referirse al programa en el software LabVIEW (mirar anexo B).

28. Ejecutar el programa en LabVIEW y verificar que los datos adquiridos por el PLC se visualicen correctamente en los indicadores numéricos. Seleccionar el nombre del programa.



Figura 3.100 Abrir programa.  
Realizado por: Javier Ortiz.

29. Dar clic en el pulsador play para dar paso a que se ejecute el programa, este pulsador también puede ser usado como un pulsador de paro de emergencia.

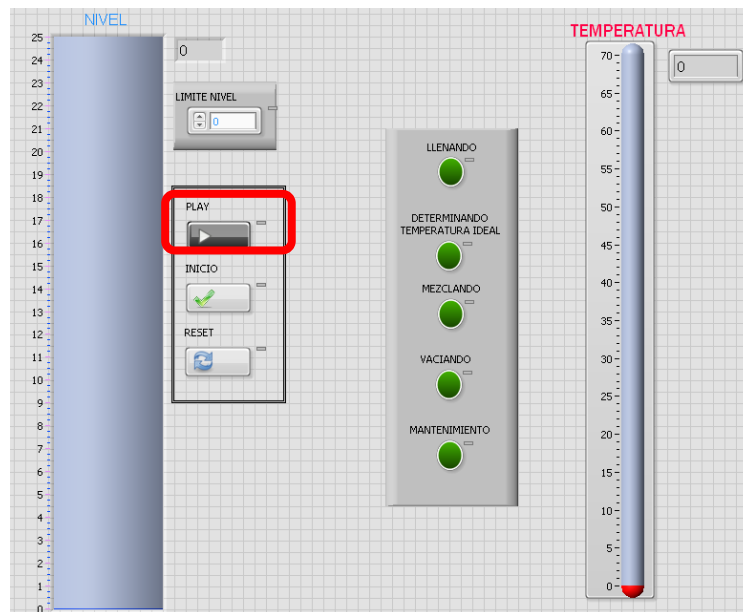


Figura 3.101 Play.  
Realizado por: Javier Ortiz.

30. Dar clic en el pulsador INICIO para que el programa empiece a trabajar.

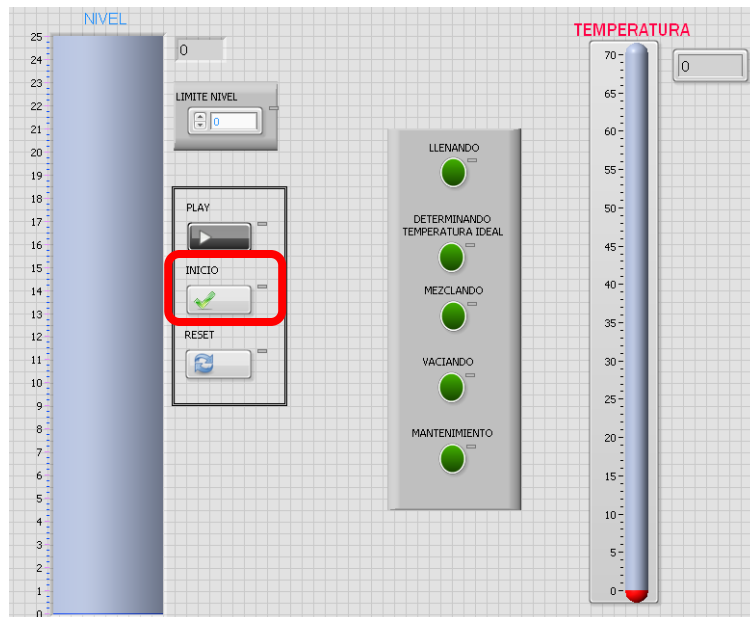


Figura 3.102 Inicio.

Realizado por: Javier Ortiz.

31. Especifique el límite máximo de líquido que desea que ingrese al tanque. El nivel máximo de líquido que puede contener el tanque es de 24 litros.

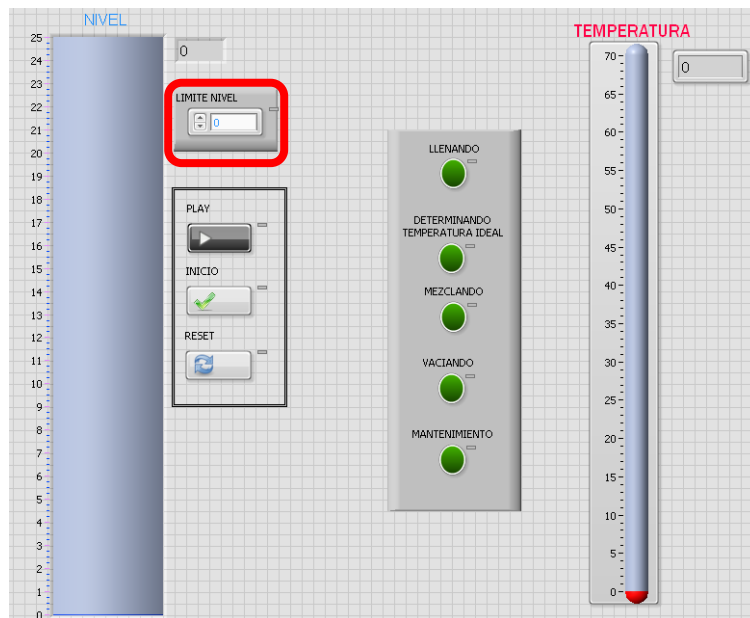


Figura 3.103 Límite máximo.

Realizado por: Javier Ortiz.

32. Una vez que se realice todos los ciclos de trabajo, la máquina necesitará de mantenimiento preventivo.
33. Cuando se haya dado el mantenimiento preventivo se debe resetear el programa para que empiece a trabajar nuevamente.

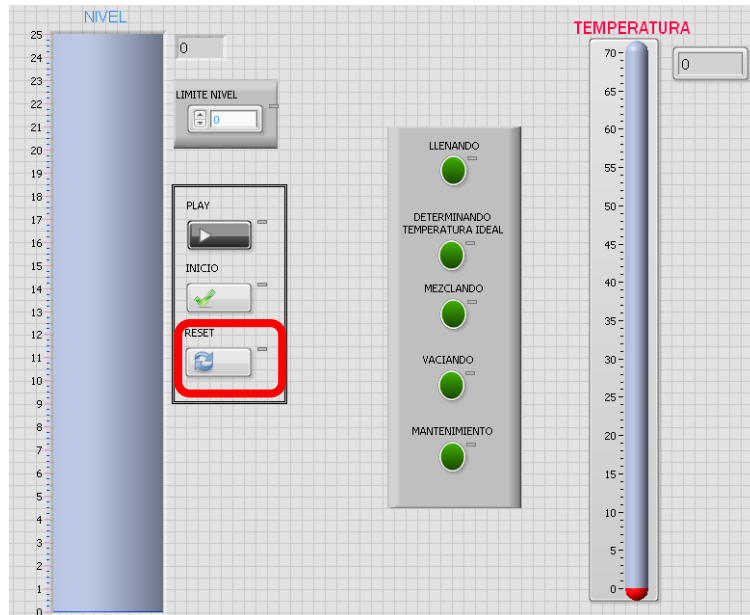


Figura 3.104 Reset.

Realizado por: Javier Ortiz.

34. De darse el caso que el sistema se detenga por haberse suscitado una emergencia, es necesario pulsar el botón reset para que los datos de los controles vuelvan a su estado inicial “cero” y de esta manera el sistema entienda que vuelve a trabajar desde el inicio.

### 3.3.6. GASTOS REALIZADOS

Para la implementación de este proyecto se determinaron los siguientes rubros:

#### 3.3.6.1. COSTOS PRIMARIOS

A continuación se detallan todos los dispositivos electrónicos y materiales usados para la realización del proyecto, y se los sintetiza en la Tabla 3.5.

**Tabla 3.5 Costos Primarios**

<b>ELEMENTOS</b>	<b>CANTIDAD</b>	<b>COSTO/UNIDAD</b>	<b>COSTO TOTAL</b>
PLC Twido TWDLCAA24DRF	1	\$603.75	\$603.75
Módulo de E/S analógicas TM2AMM3HT	1	\$260	\$260
Cable de programación TSX PCX 1031	1	\$195.69	\$195.69
Fuente externa de 24Vcc	1	\$40	\$40
		<b>TOTAL</b>	<b>\$1099.44</b>

Fuente: Investigación de campo  
Elaborado por: Javier Ortiz

### 3.3.6.2 COSTOS SECUNDARIOS

En la Tabla 3.6 se encuentran los gastos secundarios que están relacionados indirectamente con la realización del proyecto.

**Tabla 3.6 Costos Secundarios**

<b>DESCRIPCION</b>	<b>CANTIDAD</b>	<b>COSTO/UNIDAD</b>	<b>COSTO TOTAL</b>
Derechos de asesor	1	\$120	\$120
Impresiones	500	\$0.05	\$25
Internet	50	\$0.90	\$45
Tinta para impresora	2	\$10	\$20
Materiales de papelería		Varios	\$50
		<b>TOTAL</b>	<b>\$260</b>

Fuente: Investigación de campo  
Elaborado por: Javier Ortiz

### 3.3.6.3 COSTO TOTAL

El costo total se representa en la Tabla 3.7 que es la unión de los costos primario y secundario como se muestra a continuación.

**Tabla 3.7** Costo Total

Costo Primario	\$1099.44
Costo Secundario	\$260
Total	\$1359.44

Fuente: Investigación de campo  
Elaborado por: Javier Ortiz

# CAPÍTULO IV

## CONCLUSIONES Y RECOMENDACIONES

### 4.1 CONCLUSIONES

- ✓ Se desarrolló el programa para el control y monitoreo de una estación de nivel usando el PLC Twido en el laboratorio de Instrumentación Virtual del Instituto Tecnológico Superior Aeronáutico.
- ✓ Se determinó que hay que usar memorias del tipo Word y DWord en el software Twido Suite debido a que los datos adquiridos de la estación de nivel se representan en el programa con una gran cantidad de bits.
- ✓ Las marcas son muy usadas debido a que de esta manera se elimina el uso de bobinas o salidas físicas optimizando el programa.
- ✓ En el programa Twido Suite se usan direcciones %MDn, en las cuales se almacenarán los datos enviados desde LabVIEW.
- ✓ Para direccionar un dato en escrito en un control numérico hacia una marca %MDn en Twido Suite; es necesario usar el software OPC SERVER.
- ✓ Al realizar las pruebas de funcionamiento se determinó que existe una variación entre los datos que se visualiza en LabVIEW y los datos reales, para corregir esto se realizó un proceso aritmético el cual consiste en la diferencia entre el valor máximo y el valor mínimo; a este nuevo valor se denomina delta y se debe sumar en la fórmula del escalamiento y eliminar en algo el error residual.
- ✓ Una HMI (interfaz hombre-máquina) es el lugar donde interactúan humanos con computadores, y por lo tanto hay que tener en cuenta el entorno físico, es decir con lo que el hombre interactuará.

## 4.2 RECOMENDACIONES.

- ✓ Tener en cuenta la cantidad de bits que corresponden a los voltajes de salida de los sensores para determinar el tamaño de las memorias en las que se almacenarán los datos.
- ✓ Determinar los valores correspondientes a cada nivel de líquido y a cada grado de temperatura para realizar un correcto escalamiento y evitar errores residuales.
- ✓ Verificar las conexiones antes de alimentar a los equipos con voltaje.
- ✓ Usar una fuente de 24V para que el módulo de expansión funcione correctamente.
- ✓ Mientras el sistema no cumpla todos los ciclos de trabajo no se pueden cambiar los parámetros porque el sistema colapsa y es necesario resetearlo para empezar de nuevo.
- ✓ Este control usa una comunicación serial, es decir se envía y recibe bit por bit; es por esto que no pueden trabajar de una forma multidireccional. Cada programa envía un bit a la vez y si funcionan los dos programas al mismo tiempo existirán conflictos en la comunicación. Es por eso que debe trabajar un programa a la vez.



## GLOSARIO DE TÉRMINOS.

**ASCII:** Acrónimo inglés de American Standard Code for Information Interchange o Código Estándar Estadounidense para el Intercambio de Información. El código ASCII utiliza 7 bits para representar los caracteres, aunque inicialmente empleaba un bit adicional (bit de paridad) que se usaba para detectar errores en la transmisión. A menudo se llama incorrectamente ASCII a otros códigos de caracteres de 8 bits, como el estándar ISO-8859-1 que es una extensión que utiliza 8 bits para proporcionar caracteres adicionales usados en idiomas distintos al inglés, como el español.

**BIT:** Bit Es el acrónimo Binary digit (dígito binario). Un bit es un dígito del sistema de numeración binario. Mientras que en el sistema de numeración decimal se usan diez dígitos, en el binario se usan sólo dos dígitos, el 0 y el 1. Un bit o dígito binario puede representar uno de esos dos valores, 0 ó 1.

**BYTE:** Es una unidad de información utilizada como un múltiplo del bit. Equivale a 8 bits.

**Código binario:** Es el sistema de representación de textos, o procesadores de instrucciones de computadora utilizando el sistema binario.

**GRAF CET:** Es un grafo o diagrama funcional normalizado, que permite hacer un modelo del proceso a automatizar, contemplando entradas, acciones a realizar, y los procesos intermedios que provocan estas acciones. Inicialmente fue propuesto para documentar la etapa secuencial de los sistemas de control de procesos a eventos discretos. No fue concebido como un lenguaje de programación de autómatas, sino un tipo de Grafo para elaborar el modelo pensando en la ejecución directa del automatismo o programa de autómatas.

**LENGUAJE LADDER:** también denominado lenguaje de contactos o en escalera, es un lenguaje de programación gráfico muy popular dentro de los autómatas programables debido a que está basado en los esquemas eléctricos de control clásicos. De este modo, con los conocimientos que todo técnico eléctrico posee, es muy fácil adaptarse a la programación en este tipo de lenguaje.

**Magnitud:** Es una propiedad o cualidad medible de un sistema físico, es decir, a la que se le pueden asignar distintos valores como resultado de una medición.

**Módulo de E/S de aplicación:** Módulos que permiten la entrada y salida de señales a un autómata. No todos los modelos del autómata permiten la aplicación.

**PLC:** Un controlador lógico programable, más conocido por sus siglas en inglés PLC (Programmable Logic Controller), es una computadora utilizada en la ingeniería automática o automatización industrial, para automatizar procesos electromecánicos, tales como el control de la maquinaria de la fábrica en líneas de montaje o atracciones mecánicas.

**Potenciómetro analógico:** Tensión aplicada que puede ejecutarse y convertirse en un valor binario para ser usado por la aplicación.

**Sensor:** Es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc.

**Twido Suite:** Twido Suite es el primer software que está organizado según el ciclo de desarrollo del proyecto. La navegación por el software es tan sencilla que se convierte en una herramienta para la enseñanza.

**Twido:** Los controladores programables Twido, han sido optimizados para las instalaciones sencillas y las máquinas pequeñas

**Variable:** Entidad del tipo BOOL, WORD, DWORD, etc., cuyos contenidos se pueden modificar desde el programa durante su ejecución.

## BIBLIOGRAFÍA

- ✓ [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)
- ✓ <http://www.ebah.com.br/content/ABAAABeEkAl/manual-twido>
- ✓ <http://static.schneiderelectric.us/docs/Machine%20Control/Automation%20Components%20Links/Twido/DIA3ED2041102EN.pdf>
- ✓ <http://www.slideshare.net/MagdaDiaz/twido-basico1>
- ✓ <http://es.wikipedia.org/wiki/LabVIEW>
- ✓ [http://www.gte.us.es/ASIGN/IE\\_4T/Tutorial%20de%20Labview.pdf](http://www.gte.us.es/ASIGN/IE_4T/Tutorial%20de%20Labview.pdf)
- ✓ <http://es.wikipedia.org/wiki/OPC>
- ✓ <http://www.matrikonopc.es/opc-servidor/index.aspx>
- ✓ [http://www05.abb.com/global/scot/scot209.nsf/veritydisplay/8589f307e0be08b2c125723300392640/\\$file/2cdc125023m0201.pdf](http://www05.abb.com/global/scot/scot209.nsf/veritydisplay/8589f307e0be08b2c125723300392640/$file/2cdc125023m0201.pdf)
- ✓ [http://www.adtecnologias.com/Catalogos/Manual\\_Twido.pdf](http://www.adtecnologias.com/Catalogos/Manual_Twido.pdf)
- ✓ <http://es.wikipedia.org/wiki/Sensor>
- ✓ [http://es.wikipedia.org/wiki/Magnitud\\_f%C3%ADsica](http://es.wikipedia.org/wiki/Magnitud_f%C3%ADsica)

**ANEXOS**

# ANEXO A

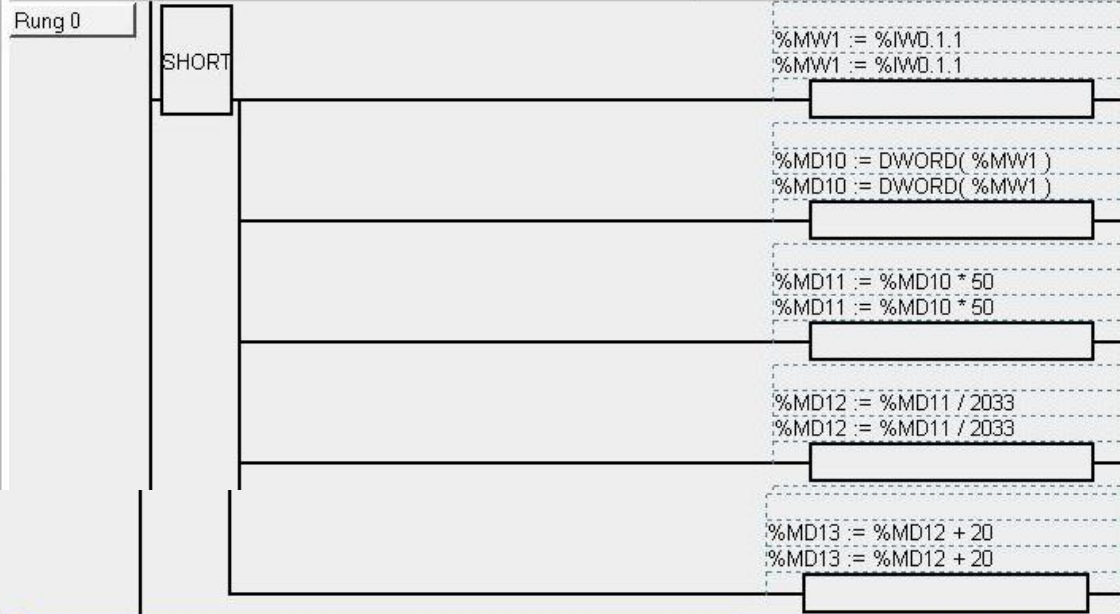
## ESCALAMIENTO NIVEL

SHORT	<pre>%MWD := %IWD.1.0 %MWD := %IWD.1.0</pre>
	<pre>%MDO := DWORD( %MWD ) %MDO := DWORD( %MWD )</pre>
	<pre>%MDO := DWORD( %MWD ) %MDO := DWORD( %MWD )</pre>
	<pre>%MD1 := %MDO - %MD30 %MDO := %MDO - %MD30</pre>
	<pre>%MD1 := %MDO - %MD30 %MDO := %MDO - %MD30</pre>
	<pre>%MD2 := %MD1 * 23 %MDO := %MD1 * 23</pre>
	<pre>%MD3 := %MD2 / %MD35 %MDO := %MD2 / %MD35</pre>
	<pre>%MD4 := %MD3 + 1 %MDO := %MD3 + 1</pre>

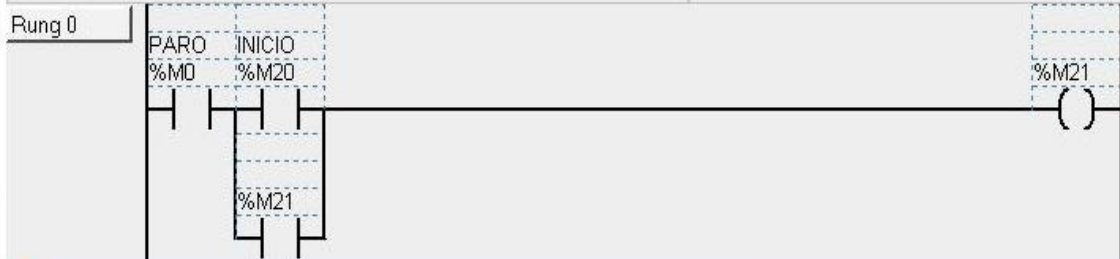
## 2 LD VARIABLE NIVEL

Rung 0	SHORT	<pre>%MD60 := %MD60 %MDO := %MD60</pre>
--------	-------	---

### 3 LD ESCALAMIENTO TEMPERATURA



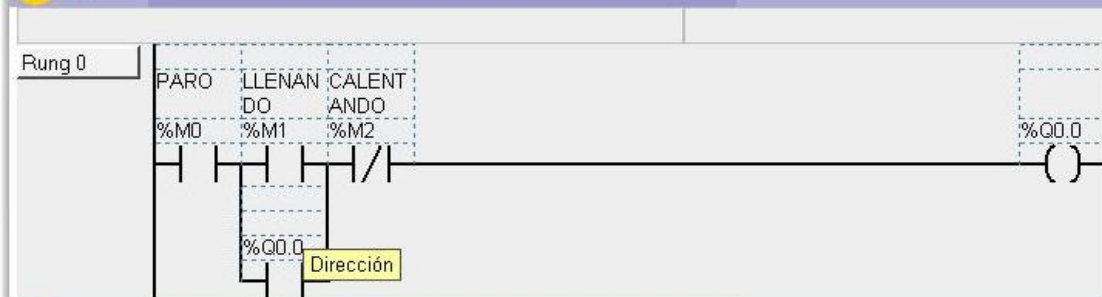
### 4 LD INICIO



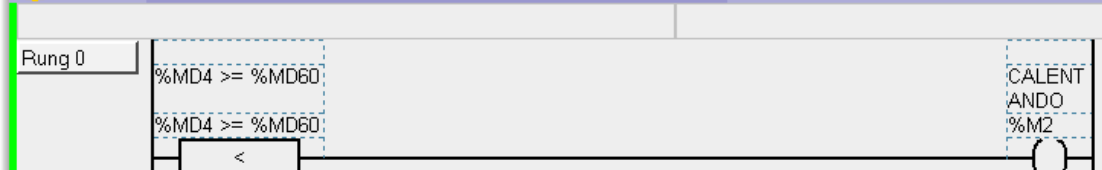
### 5 LD ENCENDIDO BOMBA



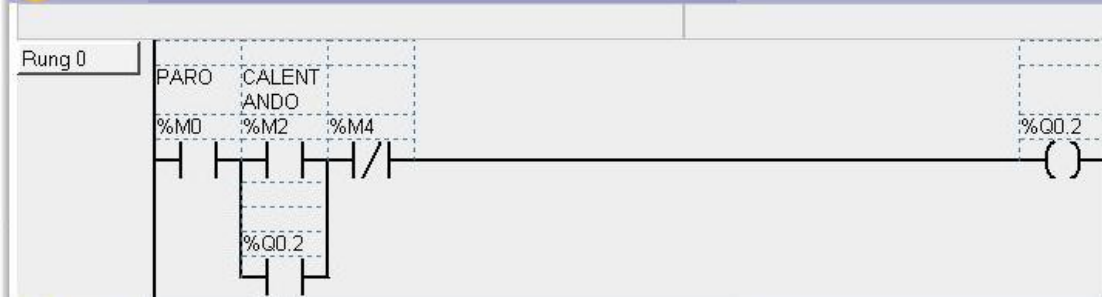
**6 LD**



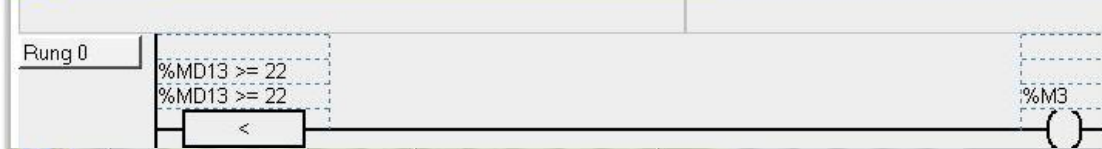
**7 LD**



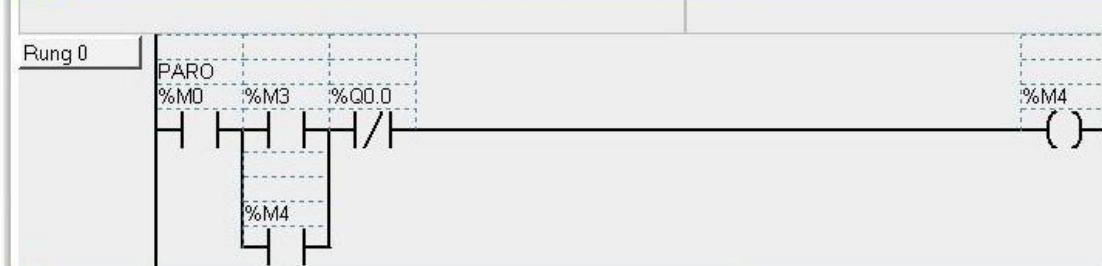
**8 LD ENCENDIDO NIQUELINA**



**9 LD**

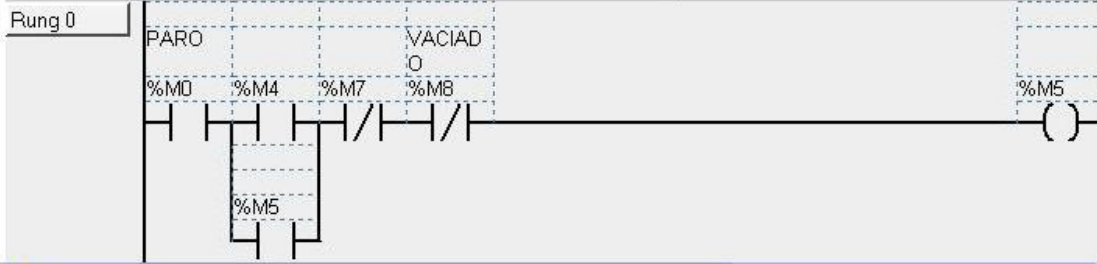


**10 LD**

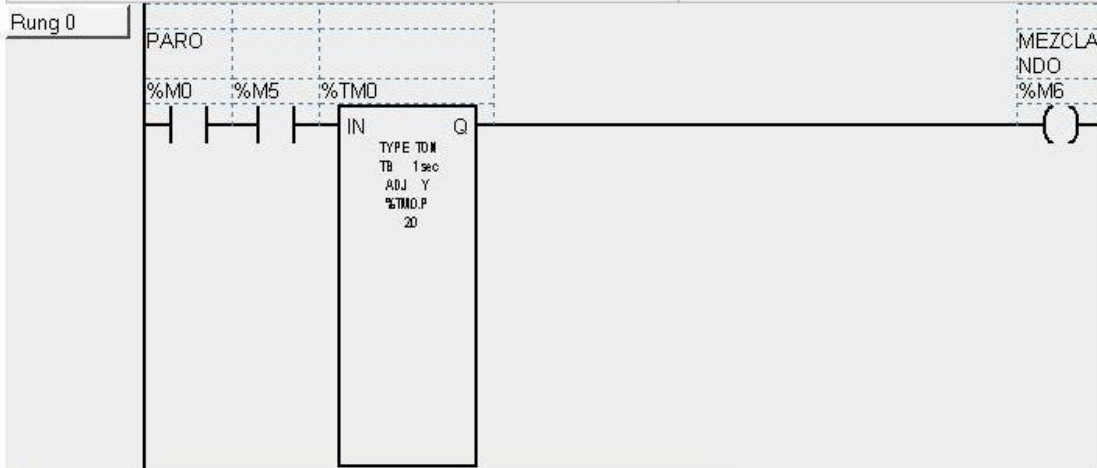




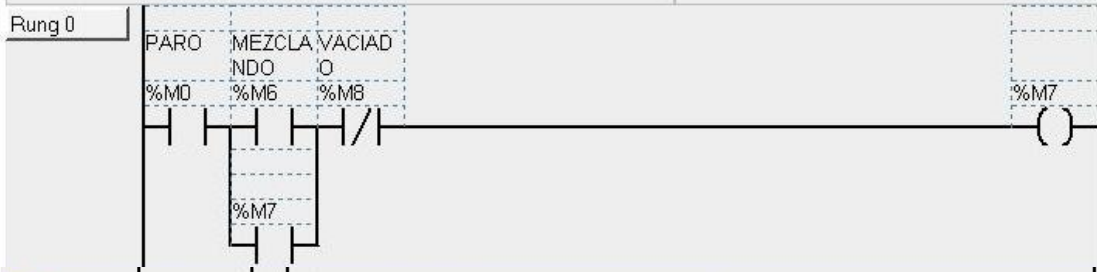
**11 LD TIEMPO DE MEZCLADO**



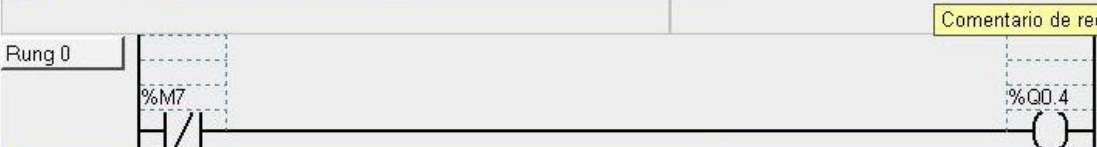
**12 LD**

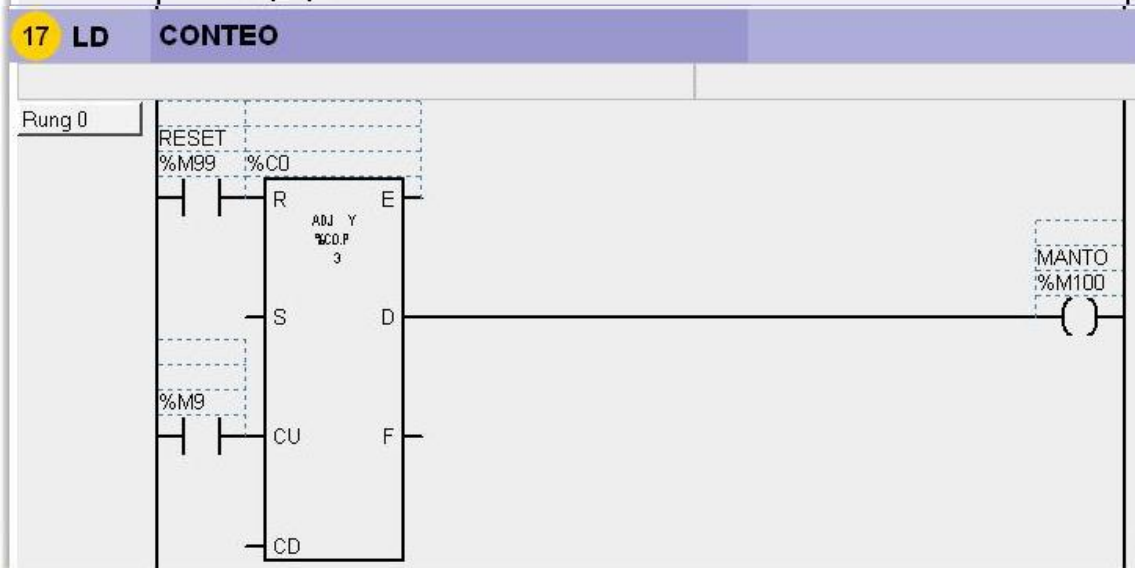
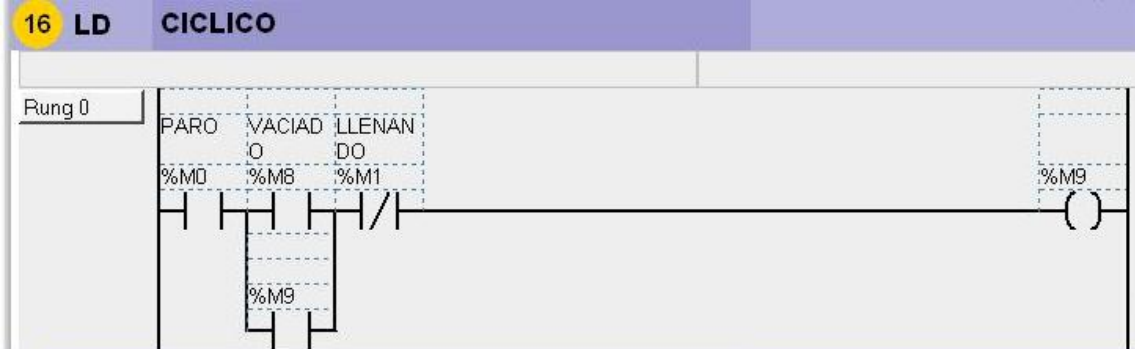
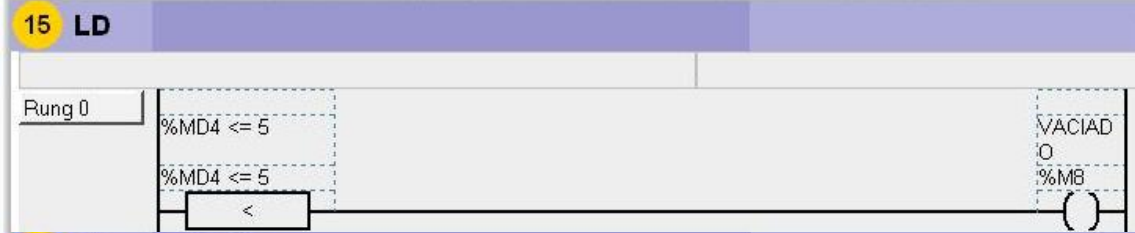


**13 LD**

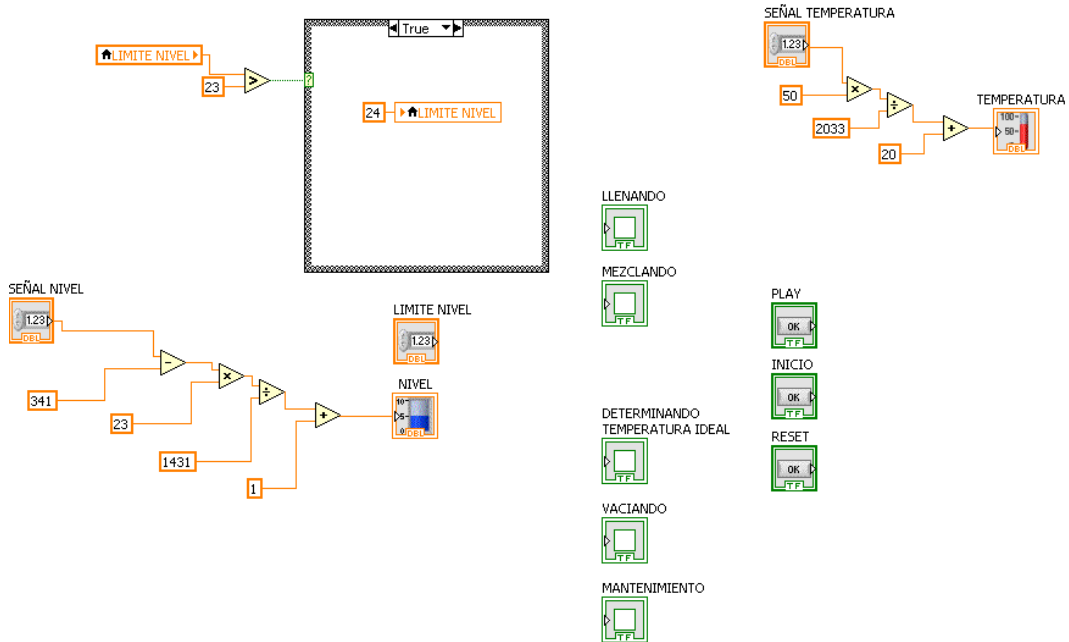
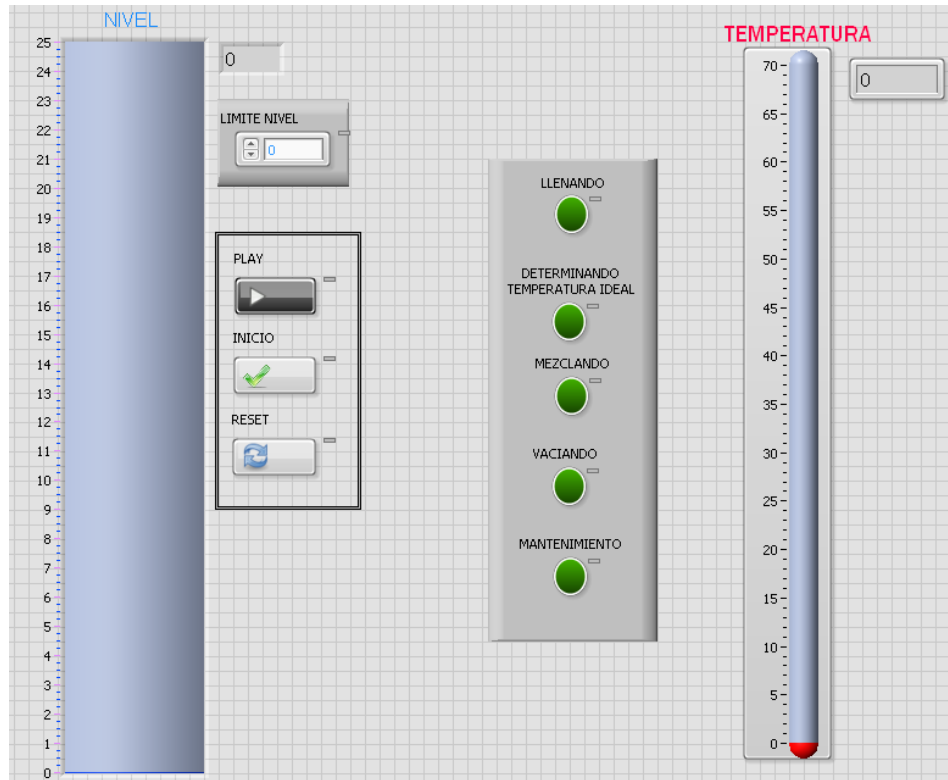


**14 LD ENCENDIDO VALVULA**





# ANEXO B



## HOJA DE VIDA



### DATOS PERSONALES

NOMBRES: ORTIZ CHICAIZA GERARDO JAVIER  
NACIONALIDAD: ECUATORIANA  
FECHA DE NACIMIENTO: 07 DE OCTUBRE DE 1991  
CÉDULA DE CIUDADANÍA: 050358614-1  
TELÉFONO: (032) 719922/ 0998308994  
CORREO ELECTRÓNICO: geraldj-j@hotmail.com  
DIRECCIÓN: LASSO

### ESTUDIOS REALIZADOS:

#### Primaria:

- ESCUELA SIMÓN BOLÍVAR

#### Secundaria:

- UEE FAE N°5

#### Superior:

- INSTITUTO TECNOLÓGICO SUPERIOR AERONÁUTICO - Egresado

### TÍTULOS OBTENIDOS:

- BACHILLER EN "ELECTRÓNICA"
- SUFICIENCIA EN INGLÉS

### EXPERIENCIA PROFESIONAL O PRÁCTICAS PREPROFESIONALES:

- Base Aérea N°23 Manta.
- DIAF-CEMA

## CURSOS REALIZADOS:

- SUFICIENCIA EN EL IDIOMA INGLÉS

## EXPERIENCIA LABORAL

- Base Aérea N°23 Manta.
- DIAF-CEMA.
- Slumberger del Ecuador

## ACEPTACIÓN DEL USUARIO

Latacunga, 27 de Marzo del 2013

Yo, ING PABLO PILATÁSIG en calidad de encargado del Laboratorio de Instrumentación Virtual del Instituto Tecnológico Superior Aeronáutico, me permito informar lo siguiente:

El proyecto de graduación elaborado por la Sr. ORTIZ CHICAIZA GERERDO JAVIER, con el tema: "CONTROL Y MONITOREO DEL NIVEL DE TEMPERATURA DE LA ESTACIÓN DE NIVEL USANDO UN PLC TWIDO", ha sido efectuado de forma satisfactoria en las dependencias de mi cargo y que la misma cuenta con todas las garantías de funcionamiento, por lo cual extiendo este aval que respalda el trabajo realizado por el mencionado estudiante.

Por tanto me hago cargo de todas las instalaciones realizadas por el Señor estudiante.

Atentamente

---

ING. PABLO PILATÁSIG

ENCARGADO DEL LABORATORIO DE INSTRUMENTACIÓN VIRTUAL

HOJA DE LEGALIZACIÓN DE FIRMAS

DEL CONTENIDO DE LA PRESENTE INVESTIGACIÓN SE RESPONSABILIZA EL  
AUTOR

---

Ortiz Chicaiza Gerardo Javier

DIRECTOR DE LA CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &  
AVIÓNICA

---

Ing. Pablo Pilatásig



Latacunga, 26 de Marzo del 2013

## CESIÓN DE DERECHOS DE PROPIEDAD INTELECTUAL

Yo, Ortiz Chicaiza Gerardo Javier, Egresado de la carrera de Electrónica Mención Instrumentación & Aviónica, en el año 2013 con Cédula de Ciudadanía N° 050358614-1, autor del Trabajo de Graduación Control y Monitoreo de una Estación de Nivel usando el PLC Twido, cedo mis derechos de propiedad intelectual a favor del Instituto Tecnológico Superior Aeronáutico.

Para constancia firmo la presente cesión de propiedad intelectual.

---

Ortiz Chicaiza Gerardo Javier  
CI. 050358614-1

Latacunga, 26 de Marzo del 2013