

**DESARROLLO E IMPLEMENTACIÓN DE UN MODELO EXPERIMENTAL DE CONTROL LOCAL A PARTIR DEL PLC QUE ESTÁ INSTALADO EN EL MÓDULO DE MONITOREO Y CONTROL DE INSTRUMENTACIÓN (M2CI) QUE PERMITA OPERAR Y CONTROLAR LAS PLANTAS PILOTOS MEDIANTE ACCESO REMOTO.**

**ANDERSON VILLABONA GARCIA  
DIEGO POLANCO CORONADO**

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA  
CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA  
INGENIERIA ELECTRONICA  
NEIVA  
2016**

**DESARROLLO E IMPLEMENTACIÓN DE UN MODELO EXPERIMENTAL DE CONTROL LOCAL A PARTIR DEL PLC QUE ESTÁ INSTALADO EN EL MÓDULO DE MONITOREO Y CONTROL DE INSTRUMENTACIÓN (M2CI) QUE PERMITA OPERAR Y CONTROLAR LAS PLANTAS PILOTOS MEDIANTE ACCESO REMOTO.**

**ANDERSON VILLABONA GARCIA  
DIEGO POLANCO CORONADO**

**TRABAJO DE GRADO**

**PEDRO TORRES SILVA  
Director de tesis  
Docente Asistente**

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA  
CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA  
INGENIERIA ELECTRONICA  
NEIVA  
2016**

Nota de Aceptación

---

---

---

---

Presidente del Jurado

---

Jurado

---

Jurado

Neiva, 9 Septiembre de 2016

Dedicamos este trabajo a Dios que nos permitió culminar esta nueva etapa de nuestras vidas, a nuestras familias por el apoyo incondicional durante todo el proceso de aprendizaje de nuestra carrera y durante la elaboración de este trabajo.

## **AGRADECIMIENTOS**

Al director de tesis Pedro Torres Silva, quien nos dio la oportunidad de poder poner en práctica todos los conocimientos adquiridos durante nuestro proceso académico y quien asesoro durante todo el proceso de desarrollo de la tesis.

A todos los tutores que nos asesoraron, guiaron y contribuyeron en nuestro desarrollo personal y profesional.

A compañeros que aportaron en el desarrollo de actividades colaborativas y que propiciaron el trabajo en grupo en las diferentes asignaturas.

A la UNAD por brindar el espacio virtual para el desarrollo de nuestra carrera y que permite a muchas personas la posibilidad de desarrollarse profesionalmente.

## TABLA DE CONTENIDOS

1. INTRODUCCIÓN.....	15
2. OBJETIVOS .....	16
2.1 OBJETIVO GENERAL.....	16
2.2 OBJETIVOS ESPECÍFICOS.....	16
3 PLANTEAMIENTO DEL PROBLEMA .....	17
3.1 DEFINICIÓN DEL PROBLEMA .....	17
3.2 JUSTIFICACIÓN.....	17
4. MARCO TEÓRICO .....	19
4.1 Sistemas de control .....	20
4.2 Dispositivos de electrónica industrial .....	21
4.3 Microwin .....	32
4.4 Labview .....	34
4.5 Laboratorios remotos.....	44
5. MATERIALES Y MÉTODOS.....	48
5.1 Materiales .....	48
5.2 Metodología.....	48
6. DESARROLLO DEL PROYECTO.....	50
6.1 Conexión a la CPU-224 .....	52
6.2 Monitoreo del horno desde labview.....	79
6.3 Análisis del desarrollo del proyecto.....	88
6.4 Cronograma.....	89
CONCLUSIONES .....	90
RECOMENDACIONES.....	91
BIBLIOGRAFÍA.....	92

## LISTA DE TABLAS

	Pág.
Tabla 1. Datos técnicos CP243-1	27
Tabla 2. Respetabilidad y precisión media de las CPUs S7-200 con alimentación DC y AC	29
Tabla 3. Datos técnicos EM231-4A	29
Tabla 4. Tabla de estados y mapa de karnaugh pulsador L1 verde.	85
Tabla 5. Tabla de estados y mapa de karnaugh pulsador horno.	86
Tabla 6. Tabla de estados y mapa de karnaugh pulsador cooler.	86

## LISTA DE FIGURAS

	Pág.
<i>Figura 1.</i> PLC S7-200	22
<i>Figura 2.</i> PT100	23
<i>Figura 3.</i> Sensor Inductivo PR08 1.5DN	25
<i>Figura 4.</i> Relé de Estado Solido	25
<i>Figura 5.</i> Relé Electromecánico	26
<i>Figura 6.</i> Cooler	27
<i>Figura 7.</i> Modulo Ethernet CP 243-1	28
<i>Figura 8.</i> Curva de definición de la precisión	31
<i>Figura 9.</i> Interfaz de usuario MicroWIN	33
<i>Figura 10.</i> Ejemplo de un Panel Frontal (1) Ventana de Panel Frontal   (2) Barra de Herramientas   (3) Paleta de Controles	34
Figura 11. Paleta de Controles	35
<i>Figura 12.</i> Controles e Indicadores Numéricos (1) Botones de Incremento/Reducción   (2) Control Numérico   (3) Indicador Numérico	36
Figura 13. Controles e Indicadores Booleano	37
<i>Figura 14.</i> Controles e Indicadores de cadena de caracteres	37
Figura 15. Ejemplo de un Diagrama de Bloques y Panel Frontal Correspondiente (1) Terminales de Indicador   (2) Cables   (3) Nodos   (4) Terminales de Control	38



Figura 16. Diagrama de Bloques	38
Figura 17. Panel Frontal del Área de un Triángulo	39
<i>Figura 18.</i> Diagrama de Bloques del Área de un Triángulo con Vista de Terminal de Ícono (1) Controles   (2) Indicador   (3) Constante	40
<i>Figura 19.</i> Diagrama de Bloques del Área de un Triángulo sin Vista de Terminal de Ícono	41
<i>Figura 20.</i> Paleta de Funciones	43
<i>Figura 21.</i> Esquema de conexión del S7-200 al PC utilizando un enlace PPI	50
<i>Figura 22.</i> Aspecto exterior del módulo de comunicación Ethernet CP 243-1	51
<i>Figura 23.</i> Creación de un nuevo proyecto con Step 7 Micro WIN	52
<i>Figura 24.</i> Elegir comando Asistente Ethernet	53
<i>Figura 25.</i> Primera ventana del asistente Ethernet Industrial	53
<i>Figura 26.</i> Parámetros de la red Ethernet	54
<i>Figura 27.</i> Determinación del número de enlaces	54
<i>Figura 28.</i> Configuración de enlaces	55
<i>Figura 29.</i> Configuraciones CRC e intervalo Keep Alive	55
<i>Figura 30.</i> Asignación de tabla de bytes para uso del módulo	56
<i>Figura 31.</i> Última ventana del asistente Ethernet Industrial	56
<i>Figura 32.</i> Subrutina ETHx_CTRL	57
<i>Figura 33.</i> Cargar la configuración en el CPU o PLC	58

<i>Figura 34.</i> Escoger la comunicación con el PLC	58
<i>Figura 35.</i> Escoger la opción de Ajustar Interfaz PG/PC	59
<i>Figura 36.</i> Configuración del puerto de comunicación.	59
<i>Figura 37.</i> Configuración del enlace de comunicación con la CPU	60
<i>Figura 38.</i> Carga del programa a la CPU	60
<i>Figura 39.</i> La ventana emergente de modo STOP	61
<i>Figura 40.</i> Desarrollo de programación en KOP	61
<i>Figura 41.</i> Cargar la configuración en el CPU o PLC	62
<i>Figura 42.</i> Escoger la comunicación con el PLC	62
<i>Figura 43.</i> Escoger la opción de Ajustar Interfaz PG/PC	63
<i>Figura 44.</i> Configuración del puerto de comunicación.	63
<i>Figura 45.</i> Configuración propiedades de red.	64
<i>Figura 46.</i> Configuración de la tarjeta de propiedades de red del PC.	65
<i>Figura 47.</i> Configuración propiedades de red.	65
<i>Figura 48.</i> Configuración del puerto de comunicación.	66
<i>Figura 49.</i> Configuración del enlace de comunicación con la CPU	66
<i>Figura 50.</i> Carga del programa a la CPU	67
<i>Figura 51.</i> La ventana emergente de modo STOP	67
<i>Figura 52.</i> Programa NI OPC Server	68
<i>Figura 53.</i> Asignación de identificación del canal	69

<i>Figura 54.</i> Configuración del modo de comunicación	69
<i>Figura 55.</i> Selección de la dirección IP de la tarjeta de red	70
<i>Figura 56.</i> Selección del método de optimización	70
<i>Figura 57.</i> Configuración del canal	71
<i>Figura 58.</i> Pantallazo para agregar canales	71
<i>Figura 59.</i> Asignación del nombre	72
<i>Figura 60.</i> Selección del modelo de PLC	72
<i>Figura 61.</i> Introducción de la dirección IP al canal	73
<i>Figura 62.</i> Configuración del Timing.	73
<i>Figura 63.</i> Configuración del Auto Demotion	74
<i>Figura 64.</i> Configuración los parámetros de comunicación	74
<i>Figura 65.</i> Configuración los parámetros de los puerto de comunicación	75
<i>Figura 66.</i> Configuración de las opciones de direccionamiento.	75
<i>Figura 67.</i> Configuración del dispositivo.	76
<i>Figura 68.</i> Añadir etiquetas en el NI OPC server	77
<i>Figura 69.</i> Configuración de una etiqueta	77
<i>Figura 70.</i> Confirmación de etiquetas en el NI OPC server	78
<i>Figura 71.</i> Validación del estado de conexión de las variables a PLC	79
<i>Figura 72.</i> Creación de un nuevo proyecto en labview	80
<i>Figura 73.</i> Creación de un I/O Server	80

<i>Figura 74.</i> Selección del tipo I/O Server	81
<i>Figura 75.</i> Configuración de OPC Client I/O Server	81
<i>Figura 76.</i> Creación de variables ligadas	82
<i>Figura 77.</i> Adición de las variables leídas del PLC	82
<i>Figura 78.</i> Verificación de la adición de las variables	83
<i>Figura 79.</i> Programación de VI para control del horno	83
<i>Figura 80.</i> Diagrama a bloques del diseño de VI control del horno	84

## GLOSARIO

**Bit:** El bit es la unidad mínima de información empleada en informática. Con él, podemos representar dos valores cuales quiera, basta con asignar uno de esos valores al estado de "apagado" (0), y el otro al estado de "encendido" (1).

**Byte:** El byte es la unidad de capacidad de almacenamiento estándar. Con esta unidad de medida se mide desde el almacenamiento de datos hasta la capacidad de memoria de un ordenador. Representa un carácter (un número, una letra, un espacio, o cualquier otro signo) y está constituido por 8 bits consecutivos, de modo tal que un byte equivaldría a 8 bits.

**Carga:** La transferencia de datos desde el controlador a un dispositivo de programación o almacenamiento.

**Contactador:** Relé de propósito especial que se utiliza para controlar la corriente eléctrica grande.

**E/S (entrada/salida):** Se utiliza para hablar de la cantidad de entradas y salidas que necesitan para un sistema, o el número de entradas y salidas que un determinado controlador lógico programable puede manejar.

**LAN:** Red de área local.

**PLC:** Controlador lógico programable.

**RS232:** Norma de comunicaciones seriales comunes. Esta norma especifica el propósito de cada uno de los 25 pines. No especifica conectores o cuales pines deben ser utilizados.

**RS485:** Los receptores tienen sensibilidad adicional que permite distancias más largas y más caídas de comunicación. Incluye protección extra para los circuitos del receptor.

**Salida digital:** Una salida que puede tener dos estados: encendido o apagado. También son llamadas salidas discretas.

## RESUMEN

Se realizara la implementación de la planta térmica del módulo M2CI, con el propósito de aprender a manipular la variable de temperatura con el módulo de siemens PLC S7-200.

Para implementar el proyecto primero fue necesario realizar las pruebas básicas de funcionamiento y calibración de los instrumentos del módulo. Se descargaron los programas MicroWin, Labview, OPCservers y Realtime, los cuales permiten poder realizar las configuraciones necesarias para poder controlar los procesos deseados.

Se controló a través de labview la variable de temperatura y adicionalmente poderlo controlar de forma remota por medio del PLC S7-200, lo cual permite que el estudiante interactúe con el modulo.

## 1. INTRODUCCIÓN

El presente trabajo de tesis es para recibir el grado de Ingeniería Electrónica de la Universidad Nacional Abierta y A Distancia (UNAD). Es un trabajo en donde se pondrán en práctica todos los conocimientos adquiridos durante el proceso de aprendizaje y donde se tendrá que profundizar en el estudio de control de procesos industriales y programación de PLC.

En la actualidad se hace necesario automatizar procesos industriales, que nos permita monitorear, supervisar y tomar control sobre determinadas variables, establecer límites (superior y/o inferior ) según parámetros operacionales y esto se logra con la programación del PLC, permitiendo reducir tiempo en procesos, minimiza la posibilidad de que se generen errores, lo que permitirá reducir costos.

Actualmente la Universidad Nacional Abierta y a Distancia (UNAD) tiene tres módulos de monitoreo y control de instrumentación (M2CI). El módulo M2CI consta del PLC S7-200 y una serie de dispositivos electrónicos que sirven para simular cuatro procesos industriales entre los cuales tenemos (Temperatura, nivel, posición y conteo) lo que nos permite poder automatizar procesos utilizando software común en la industria actual, permitiéndonos desarrollar destrezas, competencias y habilidades. El modulo permite realizar aplicaciones reales que se requieren a nivel industrial y poder realizar las pruebas necesarias antes de llevarlo a la práctica, permitiendo bajar los costos de implementación y de procesos de producción.

Nos enfocaremos principalmente en el diseño, programación e implementación de la planta de Temperatura mostrando el proceso mediante el cual se logran controlar las variables tanto en forma local, como también de forma remota y con una interfaz gráfica, que para nuestro proyecto se llevará a través de Labview.

## **2. OBJETIVOS**

### **2.1 OBJETIVO GENERAL**

- Diseñar e implementar un modelo experimental de control local a partir del PLC que está instalado en el Módulo de Monitoreo y Control de Instrumentación (M2CI) que permita operar y controlar la planta de temperatura mediante acceso local y remoto.

### **2.2 OBJETIVOS ESPECÍFICOS**

- Identificar la estructura y funcionamiento del módulo M2CI
- Plantear un sistema de conmutación que nos permita encender y controlar de forma local la planta de temperatura que integra el módulo M2CI.
- Diseñar los recursos necesarios para acceder y programar el PLC y la configuración cableada de la planta piloto de temperatura.
- Incorporar en el diseño el sistema lógico y la representación gráfica del funcionamiento del módulo mediante el uso de la herramienta LABVIEW.



## **3 PLANTEAMIENTO DEL PROBLEMA**

### **3.1 DEFINICIÓN DEL PROBLEMA**

La Universidad Nacional Abierta y a Distancia (UNAD) tiene tres módulos de monitoreo y control de instrumentación (M2CI), el cual cuenta con cuatro (4) plantas pilotos: Temperatura, nivel, posición y conteo; que se pueden configurar mediante cableado físico y controlar mediante la tarjeta de adquisición de datos DAQ USB 6008 (TAD de National Instruments – NI). Asimismo, se puede simular las operaciones sobre cada una de estas plantas pilotos mediante el software LabView.

El módulo M2CI, contiene además, una unidad de CONTROL LÓGICO PROGRAMABLE (PLC) del fabricante Siemens y de referencia S7-200, el cual permite mediante configuración cableada y programación vía Ethernet la realización de prácticas asociadas a los cursos de control y de software avanzado. La condición de una oferta académica mediada por la virtualidad le permite a la UNAD a garantizar la realización de prácticas de forma remota y controlada totalmente por los estudiantes. Bajo esta premisa, se han adelantado algunos trabajos de investigación y se han desarrollado hardware muy puntuales a la operación de una de las plantas pilotos instaladas en el M2CI. Pero aún queda mucho por hacer. Una de esas necesidades está la de diseñar prácticas, implementarlas y operarlas con la vinculación del PLC instalado.

Siendo un elemento vital, el PLC, en la automatización de procesos y que es una de las líneas de profundización del programa de ingeniería Electrónica, es importante incorporar a los sistemas de laboratorio, este dispositivo y su integración al manejo de una de las cuatro (4) plantas pilotos del M2CI.

### **3.2 JUSTIFICACIÓN**

Este es un proyecto viable, de actualidad y que tiene un destino concreto: facilitar los procesos de apropiación del conocimiento a los estudiantes de los programas de Ingeniería Electrónica, Telecomunicaciones, de las Tecnologías en Automatización electrónica y de menor forma en Ingeniería Industrial.

El recurso existe, pues en el UNAD se tiene tres (3) módulos para ser utilizados en las prácticas de laboratorios.

Se incorpora otros recursos que la UNAD posee como es el software Labview y las tarjetas electrónicas diseñadas en proyectos similares.

## 4. MARCO TEÓRICO

En el marco conceptual y teórico nos centraremos en dos elementos: de una parte, los laboratorios remotos; y de la otra, el componente práctico como mecanismo para la apropiación de la teoría. Y a partir de dichas teorías, se pueden implementar dispositivos que permitan la realización de aplicaciones donde el estudiante pueda verificar los conocimientos adquiridos, fortalecer el proceso de aprendizaje y profundizar en los mismos y mediante su uso en ambientes diferentes a lo que es para nosotros un laboratorio propiamente dicho, lo pueda realizar desde su propia casa, oficina u otro sitio.

Sin embargo, sus implementaciones prácticas presentan numerosas restricciones debido al costo de los equipos, limitaciones en capacidades espaciales, problemas de seguridad, tiempo y mantenimiento. El uso de la tecnología de la información ofrece nuevas oportunidades con el fin de reducir estas limitaciones. Los laboratorios remotos [1] pueden proporcionar acceso remoto a los experimentos y se puede permitir que los principiantes tengan acceso a los experimentos con menos restricciones de tiempo y lugar, proporcionar la orientación necesaria y garantizar un funcionamiento seguro y seguro tanto para el equipo y el personal a cargo.

La experimentación es una parte fundamental en los procesos formativos e investigativos en las áreas de la ingeniería. La apropiación del conocimiento se plantea en una movilidad del pensamiento entre la interpretación de las teorías y su aplicación a situaciones reales. Es por ello, que en el desarrollo de los procesos de aprendizaje en las ingenierías el componente práctico se hace indispensable. Pero este debe tener múltiples significados [2] desde el estudiante: Que sea un elemento motivador, que sea generador de nuevas técnicas tanto para el diseño de soluciones como para la construcción de modelos experimentales; que de una u otra forma favorezca las denominadas “actitudes científicas” y contribuya al fortalecimiento y comprensión de los conceptos científicos, y, por último; que la práctica a realizar sea coherente con el ejercicio profesional.

## 4.1 Sistemas de control

En los procesos de automatización es posible detener o iniciar procesos sin la necesidad de la intervención del usuario, para ello es necesario contar con información de entrada (sensores) y así poder procesar y emitir una o varias señales de salida de forma preestablecida. Existen dos clases comunes de sistemas de control, sistemas de lazo abierto y sistemas de lazo cerrado.

### 4.1.1 Lazo abierto

Es aquel sistema en que solo actúa el proceso sobre la señal de entrada y da como resultado una señal de salida independiente a la señal de entrada, pero basada en la primera. Esto significa que no hay retroalimentación hacia el controlador para que éste pueda ajustar la acción de control. Es decir, la señal de salida no se convierte en señal de entrada para el controlador. Se compone de la señal de entrada (Input), Elementos de control, Proceso, y Señal de salida (Output)

Ejemplo: El llenado de un tanque usando una manguera de jardín. Mientras que la llave siga abierta, el agua fluirá. La altura del agua en el tanque no puede hacer que la llave se cierre y por tanto no nos sirve para un proceso que necesite de un control de contenido o concentración.

### 4.1.2 Lazo cerrado

Son también llamados sistemas de realimentación o feedback.

El proceso de decisión depende tanto de la entrada como de la salida. Si la señal de salida captada por el sensor no es la deseada o se sale de los parámetros establecidos el sistema puede reaccionar a esta situación y reaccionar. Se compone de la señal de entrada (Input), Comparador, Elementos de control, Proceso, Señal de salida (Output) y realimentación (Sensores).

El control en lazo cerrado es imprescindible cuando se da alguna de las siguientes circunstancias:

- Cuando un proceso no es posible de regular por el hombre.
- Una producción a gran escala que exige grandes instalaciones y el hombre no es capaz de manejar.
- Vigilar un proceso es especialmente difícil en algunos casos y requiere una atención que el hombre puede perder fácilmente por cansancio o despiste, con los consiguientes riesgos que ello pueda ocasionar al trabajador y al proceso.

Sus características son:

- Ser complejos, pero amplios en cantidad de parámetros.
- La salida se compara con la entrada y le afecta para el control del sistema.
- Su propiedad de retroalimentación.
- Ser más estable a perturbaciones y variaciones internas.

#### 4.2 Dispositivos de electrónica industrial

Actualmente la electrónica está inmersa en muchos de los procesos que realizamos tanto como cotidianos como industriales para el beneficio de todos, ya sea ayudando hacer una tarea más fácil o realizar un proceso sofisticado que antes requería mucho tiempo y un enorme riesgo para la integridad de las personas que estaban involucrados, con el pasar del tiempo y la creación de nuevos avances tecnológicos en el ámbito de la electrónica, existen muchas herramientas que nos ayudan tanto medir señales físicas tales como: (Temperatura, presión, altura.. entre otras); de forma ya sea de forma de señal análoga o de manera digital.

Todo nos permite que podemos involucrar todo en un solo sistema ofimático y más fácil de poder procesar y control muchas cosas a la necesidad que tengamos ya sea de manera local o de manera remota. Gracias a que contamos con dispositivos electrónicos que nos permite hacer estas tareas de manera más fácil y eficiente. A continuación se describirán algunos de los dispositivos y software que usaremos en nuestro proyecto.

#### 4.2.1 PLC SIMATIC S7-200 – Siemens



Figura 1. PLC S7-200

#### Características del PLC S7-200 [4]

- El micro-PLC para el máximo efecto de automatización
- Montaje, programación y uso particularmente fáciles.
- De alta escala de integración, requiere poco espacio, potente.
- Aplicable tanto para los controles más simples como también para tareas complejas de automatización.
- Aplicable aislado, interconectado en red o en configuraciones descentralizadas.
- El PLC también para campos donde, por motivos económicos, no se aplicaban hasta ahora autómatas programables.
- Con destacadas prestaciones de tiempo real y potentes posibilidades de comunicación (PPI, PROFIBUS-DP, AS-Interface).

#### Características mecánicas

Carcasa robusta y compacta, elementos de conexión y mando fácilmente accesibles, protegidos por tapas frontales.

#### Características de equipamiento

#### Normas internacionales: [5]

El SIMATIC S7-200 satisface las normas VDE, UL, CSA y FM (clase I, categoría 2, grupos de clase de peligro A, B, C y D, T4A). El sistema de gestión de calidad aplicado durante la fabricación ha sido certificado conforme a ISO 9001.

## Comunicación [6]

El SIMATIC S7-200 ofrece diferentes posibilidades de comunicación:

### Puerto de comunicación:

Operable en modo de esclavo PPI, MPI o interfaz libremente programable.

### Conexión a AS-Interface:

Actuando como maestro a través del módulo de comunicaciones CP 242-2.

Conexión a PROFIBUS-DP como esclavo: vía PROFIBUS-EM 277.

### Red PPI

PPI se denomina el sistema de comunicación integrado en cada CPU S7-200. Se pueden programar, manejar y vigilar varias conexiones punto a punto a través de un cable bifilar común. Así, las CPUs pueden comunicarse simultáneamente con varias CPUs, TDs u OPs. Velocidad de transmisión de datos 9,6, 19,2 y 187,5 kbits/s.

### Es posible conectar:

Programadoras Field PG, Power PG. PCs (compatibles AT), a través de cable PC/PPI. Visualizador de textos TD 200, panel táctil TP070; Paneles de operador SIMATIC.

### SIMATIC S7-200:

A través de una misma línea de datos y un PG, PC, TD 200 u OP es posible acceder hasta máx. a 31 S7-200. Intercambio directo de datos entre micro-PLCs S7-200 (actuando como maestros) con NETR y NETW.

Comunicación en el modo PPI.

#### 4.2.2 Sensor de temperatura pt100



*Figura 2. PT100*

[7] La PT100 es un sensor de temperatura que a 0 °C tiene 100 ohms y que al aumentar la temperatura aumenta su resistencia eléctrica. El incremento de la resistencia de la PT100 no es lineal pero si creciente y característico del platino de tal forma que mediante tablas es posible encontrar la temperatura exacta a la que corresponde.

PT100 3 hilos – El modo de conexión de 3 hilos es el más común y más utilizado para procesos industriales resuelve bien el problema de error generado por los cables. El único requisito es que los tres cables tengan la misma resistencia eléctrica pues el sistema de medición se basa (casi siempre) en el “puente de Wheatstone”. Por supuesto el lector de temperatura debe ser para este tipo de conexión. La mayoría de los equipos industriales vienen preparados para conexión PT100 3 hilos.



#### 4.2.3 Sensor de proximidad inductivo pr08 1.5dn



Figura 3. Sensor Inductivo PR08 1.5DN

[8] Los sensores inductivos son una clase especial de sensores que sirve para detectar materiales ferrosos. Son de gran utilización en la industria, tanto para aplicaciones de posicionamiento como para detectar la presencia o ausencia de objetos metálicos en un determinado contexto: detección de paso, de atasco, de codificación y de conteo.

En el proyecto el sensor inductivo se utiliza para la apertura y cierre de la puerta del horno.

#### 4.2.4 Relé de estado solido



Figura 4. Relé de Estado Solido

[9] Un relé de estado sólido (SSR en inglés) es un dispositivo interruptor electrónico que conmuta el paso de la electricidad cuando una pequeña corriente es aplicada en sus terminales de control. Los SSR consisten en un sensor que responde a una entrada apropiada (señal de control), un interruptor electrónico de estado sólido que conmuta el circuito de carga, y un mecanismo de acoplamiento a partir de la señal de control que activa este interruptor sin partes mecánicas. El relé puede estar diseñado para conmutar corriente alterna o continua. Hace la misma función que el relé electromecánico, pero sin partes móviles.

#### 4.2.5 Relé electromecánico



Figura 5. Relé Electromecánico

[10] La bobina es el principal componente del relé. A su alrededor se genera un campo electromagnético cuando el relé es energizado.

Este campo electromagnético genera una fuerza capaz de mover un conjunto mecánico (armadura fija) con contactos móviles, cambiando así su estado de normalmente abierto a cerrado, o de normalmente cerrado a abierto de acuerdo con el tipo de relé

#### 4.2.6 Cooler



*Figura 6. Cooler*

Es utilizado para ayudar a controlar la temperatura interior del horno, permitiendo acelerar el proceso de reducir la temperatura.

#### 4.2.7 Modulo ethernet cp 243-1

El CP 243-1 es un procesador de comunicaciones que está previsto para utilizarlo en un autómata programable S7-200. Permite conectar un sistema S7-200 a Industrial Ethernet (IE). De este modo, con el CP 243-1 también es posible realizar comunicaciones vía Ethernet en la gama de prestaciones inferior de la serie de productos S7. Y así, usando un S7-200 y STEP 7 Micro/WIN 32 se puede, por un lado, configurar, programar y diagnosticar vía Ethernet incluso a distancia, y por otro lado, un S7-200 puede comunicarse vía Ethernet con otro autómata S7-200, con un S7-300 o con un S7-400. También es posible la comunicación con un servidor OPC.

En el sistema de comunicaciones SIMATIC NET, Industrial Ethernet es la red para el nivel de control y para el nivel de célula. Físicamente, Industrial Ethernet es una red eléctrica sobre la base de una línea coaxial apantallada, un cableado Twisted Pair o una red de fibra óptica sobre la base de un conductor de fibra óptica. Industrial Ethernet está definida por el estándar internacional IEEE 802.3. [12]



Figura 7. Modulo Ethernet CP 243-1

### Velocidad de transferencia

Tasa de transferencia

- en la interfaz 1 10 ... 100 Mbit/s

### Interfaces

Número de interfaces / según Industrial Ethernet 1

Número de conexiones eléctricas

- en la interfaz 1 / según Industrial Ethernet 1
- para alimentación 1

Tipo de conexión eléctrica

- en la interfaz 1 / según Industrial Ethernet Puerto RJ45
- para alimentación Regleta de bornes de 3 polos

### Tensión de alimentación, consumo, pérdidas

Tipo de corriente / de la tensión de alimentación DC

Tensión de alimentación / 1 / del bus de fondo 5 V

Tensión de alimentación / externa 24 V

Tensión de alimentación / externa / con DC / valor nominal 24 V

tolerancia positiva relativa / con DC / con 24 V 20 %

tolerancia negativa relativa / con DC / con 24 V 15 %

corriente consumida

- del bus de fondo / con DC / con 5 V / típico 0,06 A

- de la tensión de alimentación externa / con DC / con 24 V / típico 0,053 A

- de la tensión de alimentación externa / con DC / con 24 V / máx. 0,06 A

Pérdidas [W]	1,5 W
<b>Condiciones ambientales admisibles</b>	
Temperatura ambiente	
• con instalación vertical / durante el funcionamiento	0 ... 45 °C
• con posición de montaje vertical / durante el funcionamiento	0 ... 55 °C
• durante el almacenamiento	-40 ... +70 °C
• durante el transporte	-40 ... +70 °C
humedad relativa del aire / con 25 °C / sin condensación / durante el funcionamiento / máx.	95 %
Grado de protección IP	IP20
<b>Diseño, dimensiones y pesos</b>	
Formato de módulos	Módulo compacto S7-200 de doble anchura
Anchura	71,2 mm
Altura	80 mm
Profundidad	62 mm
Peso neto	0,15 kg
<b>Características, funciones y componentes del producto / Generalidades</b>	
Número de módulos	
• por CPU / máx.	1
• Observación	-
<b>Datos de prestaciones / Comunicación S7</b>	
Número de conexiones posibles / para comunicación S7	
• máx.	8
• en conexiones PG / máx.	1
• en conexiones PG/OP / máx.	8
<b>Datos de prestaciones / Funciones TI</b>	
Número de conexiones posibles	
• como cliente / mediante FTP / máx.	1
• como servidor / mediante HTTP / máx.	4
• como cliente de correo electrónico / máx.	1
Número de correos electrónicos con 1024 caracteres / del cliente de correo electrónico / máx.	32
Número de derechos de acceso / de la protección contra el acceso	8
Capacidad de memoria / de la memoria de usuario	
• como sistema de archivos de memoria FLASH	8 Mibyte

Número de ciclos de escritura posibles / de las celdas de la memoria Flash	100000
<b>Datos de prestaciones / Telecontrol</b>	
Protocolo / soportado	
● TCP/IP	Sí
Función del producto / Soporte de MIB	No
Protocolo / soportado	
● SNMP v1	No
Software de configuración	
● necesario	STEP 7-Micro/WIN, V4.0 SP8 o superior

Tabla 1. Datos técnicos CP243-1

#### 4.2.8 Módulo análogo em 231

El EM231 es un módulo de entradas analógicas de 12 bits rápido y económico. Dicho módulo puede convertir una entrada analógica en su correspondiente valor digital en 171  $\mu$ s para la CPU 212 y en 139  $\mu$ s para las demás CPUs S7-200. La conversión de la señal analógica se efectúa cada vez que el programa de usuario accede a la entrada analógica. Los tiempos mencionados se deben agregar al tiempo de ejecución básico de la operación utilizada para acceder a la entrada analógica.

El EM231 proporciona un valor digital no procesado (sin linealización ni filtraje) que corresponde a la tensión o a la corriente analógicas en los terminales de entrada del módulo. Puesto que se trata de un módulo rápido, la señal de entrada analógica puede cambiar rápidamente (incluyendo interferencias internas y externas). Las diferencias de un muestreo a otro, causadas por interferencias de una señal de entrada analógica que cambie constante o lentamente, se pueden reducir creando un promedio de una serie de muestreos. Cuanto mayor sea la cantidad de muestreos utilizados para calcular el promedio, tanto más lento será el tiempo de respuesta a cambios en la señal de entrada.

Los datos relativos a la repetibilidad describen las diferencias de un muestreo a otro en el caso de las señales de entrada que no cambien. Dichos datos definen el margen que contiene un 99% de todos los muestreos. La precisión media describe

el valor promedio del error (la diferencia entre el valor promedio de los muestreos individuales y el valor exacto de la señal real de la entrada analógica). La repetibilidad se describe en la curva representada en la figura 8. Dicha figura muestra el margen de repetibilidad (que contiene un 99% de los muestreos), el valor promedio de los muestreos individuales y la precisión media. En la tabla 2 se indican los datos relativos a la repetibilidad y la precisión media con respecto a los márgenes configurables.

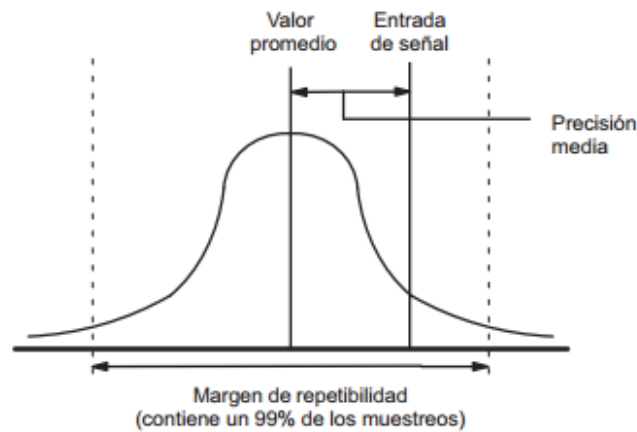


Figura 8. Curva de definición de la precisión.

Margen máximo	Repetibilidad <sup>1</sup>		Precisión media <sup>1, 2, 3, 4</sup>	
	% del margen máximo	Contajes	% del margen máximo	Contajes
<b>CPUs S7-200 con alimentación DC</b>				
0 a 5 V	± 0.075%	± 24	± 0.1%	± 32
0 a 20 mA				
0 a 10 V				
<b>CPUs S7-200 con alimentación AC</b>				
0 a 5 V	± 0.15%	± 48	± 0.1%	± 64
0 a 20 mA				
0 a 10 V				

Tabla 2. Respetabilidad y precisión media de las CPUs S7-200 con alimentación DC y AC

<b>Producto</b>	EM231, 4 salidas analógicas:
<b>No. de referencia</b>	6ES7 232-0HD22-0XA0
<b>Dimensiones (An, Al, P)</b>	71.2 mm, 80 mm, 62 mm
<b>Alimentación:</b>	+5V DC (bus posterior): 20mA +24V DC: 60mA
<b>No. de salidas:</b>	4
<b>Formato palabra datos:</b> Bipolar Unipolar:	De -32 000 a +32 000 De 0 a 32 000
<b>Resolución:</b> Bipolar Unipolar	11 bits más signo 11 bits
<b>Rangos de señales:</b>	Tensión: +/-10V Intensidad: de 0 a 20mA
<b>Precisión:</b>	+/- 0.5% (25°C), +/- 2% (55°C)

Tabla 3. Datos técnicos EM231-4A

#### 4.3 Microwin

El Software Microwin hace parte de la firma Siemens y permite programar el PLC S7-200 por medio de un entorno gráfico. Utiliza lenguaje universal para la programación de autómatas.

Con microwin se realiza la programación que se desea cargar al PLC, para que se ejecute determinada actividad, por lo general se trata de automatizar procesos industriales.



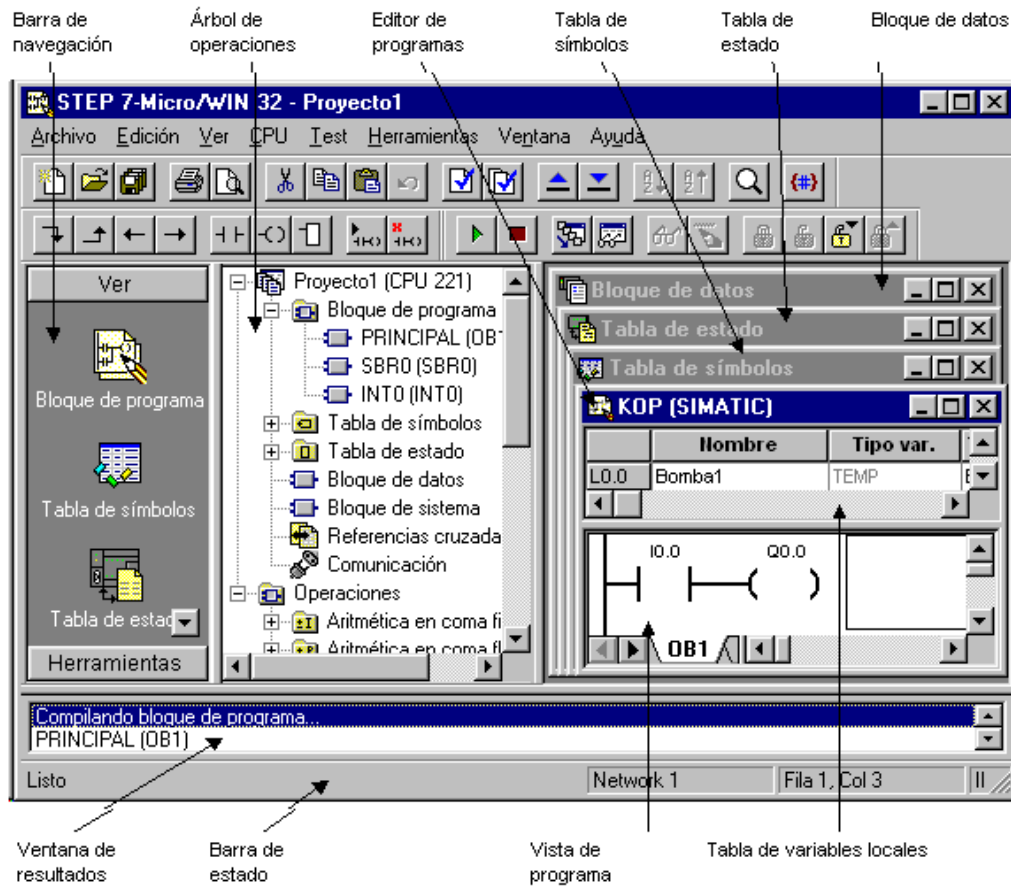


Figura 9. Interfaz de usuario MicroWIN

[11] ¿Cómo se pasa de un esquema eléctrico a un programa para PLC?

En primer lugar gire el esquema 90° hacia la izquierda. De esta manera, la barra de fase queda normalmente a la izquierda y la barra de masa a la derecha. En el medio se encuentran los contactos de su circuito.

La parte del circuito que representa la lógica de maniobra de la máquina es sustituida por el PLC (relé de tiempo, contactores de mando, etc. así como su cableado).

Un PLC no puede sustituir a los sensores (p. ej. interruptores, selectores) por el lado de entrada ni a los actuadores (p. ej. contactores de motor, contactores-inversores, válvulas) por el lado de salida.

## 4.4 Labview

[17] Los programas de LabVIEW son llamados instrumentos virtuales o VIs ya que su apariencia y operación generalmente imitan a los instrumentos físicos, como osciloscopios y multímetros. LabVIEW contiene una extensa variedad de herramientas para adquirir, analizar, visualizar y almacenar datos, así como herramientas para ayudarle a solucionar problemas en el código que escriba. Cuando crea un nuevo VI, ve dos ventanas: la ventana del panel frontal y el diagrama de bloques.

### 4.4.1 Panel Frontal

Cuando abre un VI nuevo o existente, aparece la ventana del panel frontal del VI. La ventana del panel frontal es la interfaz de usuario para el VI. La Figura 10 muestra un ejemplo de una ventana del panel frontal.

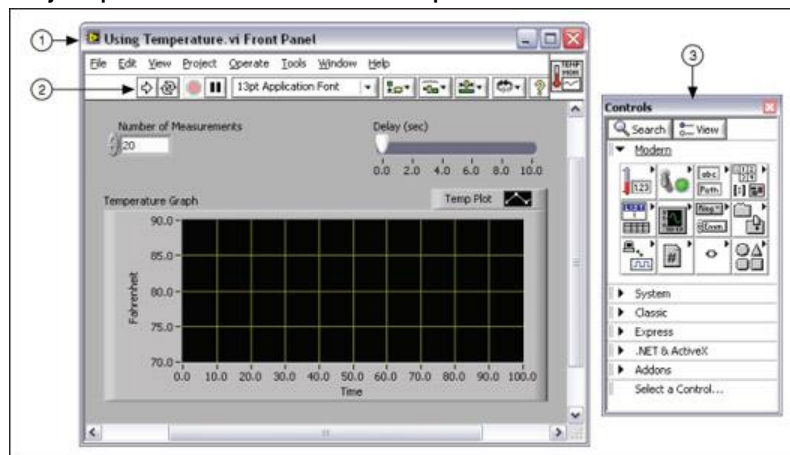


Figura 10. Ejemplo de un Panel Frontal (1) Ventana de Panel Frontal | (2) Barra de Herramientas | (3) Paleta de Controles

### 4.4.2 Paleta de Controles

La paleta de Controles contiene los controles e indicadores que utiliza para crear el panel frontal. Puede tener acceso a la paleta de Controles de la ventana del panel frontal al seleccionar View»Controls Palette o al dar clic con botón derecho en cualquier espacio en blanco en la ventana del panel frontal. La paleta de Controles está dividida en varias categorías; puede exponer algunas o todas estas

categorías para cumplir con sus necesidades. La Figura 11 muestra la paleta de Controles con todas las categorías expuestas y la categoría Moderna expandida.

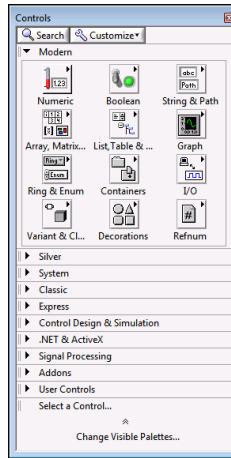


Figura 11. Paleta de Controles

Para ver o esconder categorías (subpaletas), haga clic en el botón Customize y seleccione Change Visible Palettes.

#### 4.4.3 Controles e Indicadores

Cada VI tiene un panel frontal que usted puede diseñar como una interfaz de usuario. Usted también puede usar paneles frontales como la manera de transmitir entradas y recibir salidas cuando se llama al VI desde otro diagrama de bloque. Usted crea la interfaz de usuario de un VI al colocar controles e indicadores en el panel frontal de un VI. Cuando interactúa con un panel frontal como una interfaz de usuario, puede modificar los controles para transmitir entradas y ver los resultados en los indicadores. Los controles que definen las entradas y los indicadores muestran las salidas.

Los controles generalmente son perillas, botones, perillas, deslizadores y secuencias. Simulan dispositivos de entrada de instrumentos y suministran datos al diagrama de bloques del VI. Los indicadores generalmente son gráficas, tablas, LEDs y secuencias de estado. Los indicadores simulan dispositivos de salida de instrumentos y muestran los datos que diagrama de bloques adquiere o genera. La Figura 10 tiene dos controles: Number of Measurements y Delay (sec). Tiene un indicador: una gráfica XY llamada Temperature Graph.

El usuario puede cambiar el valor de entrada para los controles Number of Measurements and Delay (sec). El usuario puede ver el valor generado por el VI en el indicador Temperature Graph. El VI genera los valores para los indicadores basados en el código creado en el diagrama de bloques.

Cada control o indicador tiene un tipo de datos asociado con él. Por ejemplo, el slide horizontal Delay (sec) es un tipo de datos numérico. Los tipos de datos utilizados con mayor frecuencia son numéricos, Booleano y cadena de caracteres. Controles e Indicadores Numéricos

El tipo de datos numérico pueden representar números de varios tipos como un entero o real. Los dos objetos numéricos comunes son el control numérico y el indicador numérico, como se muestra en la Figura 12. Los objetos como medidores y perillas también representan datos numéricos.

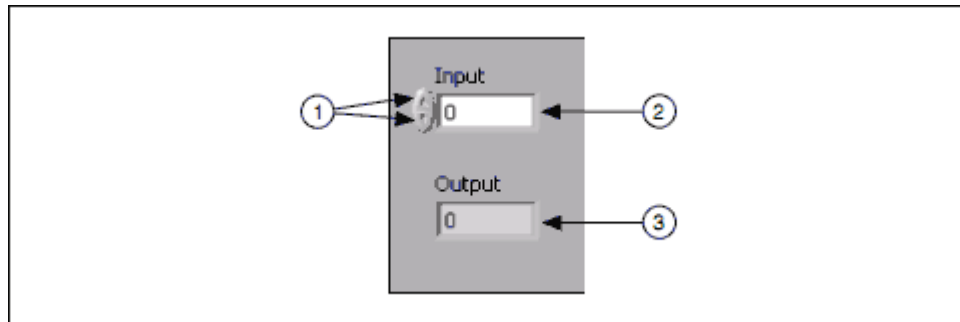


Figura 12. Controles e Indicadores Numéricos (1) Botones de Incremento/Reducción | (2) Control Numérico | (3) Indicador Numérico

Para proporcionar o cambiar valores en un control numérico, haga clic en el botón de incremento y reducción o haga doble clic en el número, introduzca un nuevo número y presione la tecla <Enter>.

#### 4.4.4 Controles e Indicadores Booleano

El tipo de datos Booleano representa datos que solamente tienen dos estados posibles, como TRUE y FALSE u ON y OFF. Use los controles e indicadores Booleano para proporcionar y visualizar valores Booleano. Los objetos Booleano simulan interruptores, botones y LEDs. El interruptor de encendido vertical y los objetos LED redondos se muestran en la Figura 13.

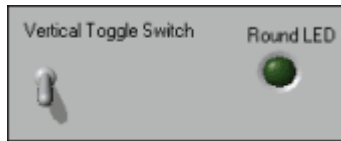


Figura 13. Controles e Indicadores Booleano

#### 4.4.5 Controles e Indicadores de Cadena de Caracteres

El tipo de datos cadena de caracteres es una secuencia de caracteres ASCII. Use controles en cadena para recibir texto desde el usuario como una contraseña o nombre de usuario. Use indicadores en cadena para mostrar texto al usuario. Los objetos en cadena más comunes son tablas y cuadros de texto, como se muestra en la Figura 14.

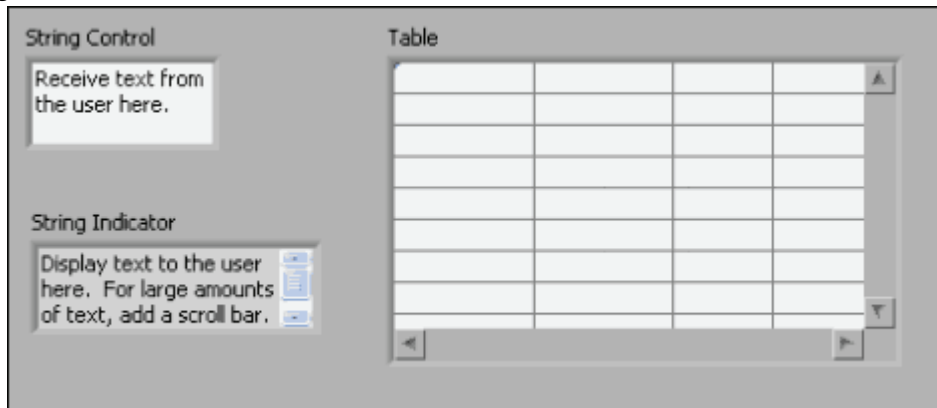


Figura 14. Controles e Indicadores de cadena de caracteres

#### 4.4.6 Diagrama de Bloques

Los objetos del diagrama de bloques incluyen terminales, subVIs, funciones, constantes, estructuras y cables, los cuales transfieren datos junto con otros objetos del diagrama de bloques.

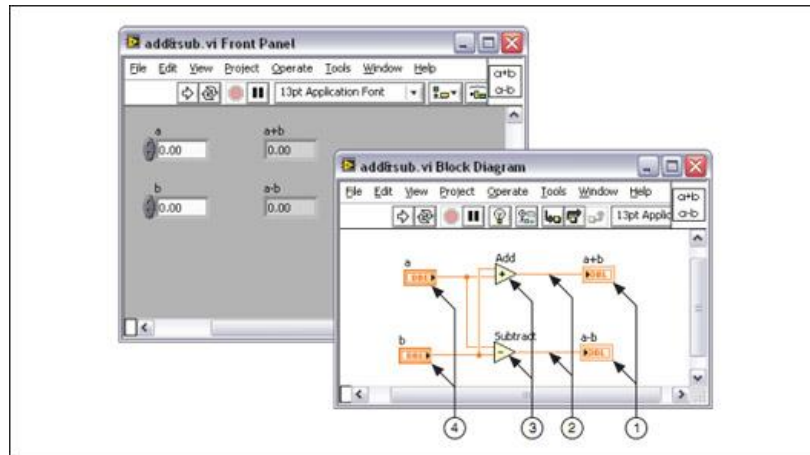


Figura 15. Ejemplo de un Diagrama de Bloques y Panel Frontal Correspondiente  
 (1) Terminales de Indicador | (2) Cables | (3) Nodos | (4) Terminales de Control

Después de que crea la ventana del panel frontal, añade código usando representaciones gráficas de funciones para controlar los objetos del panel frontal. La ventana del diagrama de bloques contiene este código de fuente gráfica.

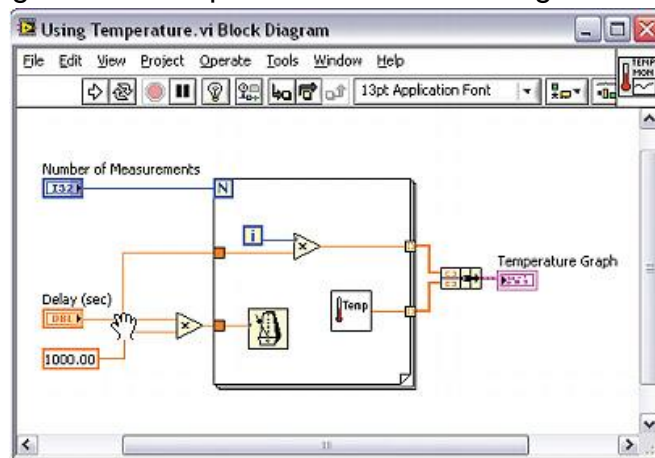


Figura 16. Diagrama de Bloques

#### 4.4.7 Terminales

Los objetos en la ventana del panel frontal aparecen como terminales en el diagrama de bloques. Las terminales son puertos de entrada y salida que intercambian información entre el panel frontal y diagrama de bloques. Son análogos a parámetros y constantes en lenguajes de programación basados en texto. Los tipos de terminales incluyen terminales de control o indicador y terminales de nodo. Las terminales de control e indicador pertenecen a los

controles e indicadores del panel frontal. Los puntos de datos que usted proporciona en los controles del panel frontal (a y b en el panel frontal anterior) pasan al diagrama de bloques a través de las terminales de control. Entonces los puntos de datos ingresan las funciones de Suma y Resta. Cuando las funciones de Suma y Resta terminan sus cálculos, producen nuevos valores de datos. Los valores de datos van a las terminales de indicador, donde actualizan los indicadores del panel frontal (a+b y a-b en el panel frontal anterior).

#### 4.4.7 Controles, Indicadores y Constantes

Los controles, indicadores y constantes se comportan como entradas y salidas del algoritmo del diagrama de bloques. Considere la implementación del algoritmo para el área de un triángulo:

$$\text{Área} = .5 * \text{Base} * \text{Altura}$$

En este algoritmo, Base y Altura son entradas y Área es una salida, como se muestra en la Figura 17.

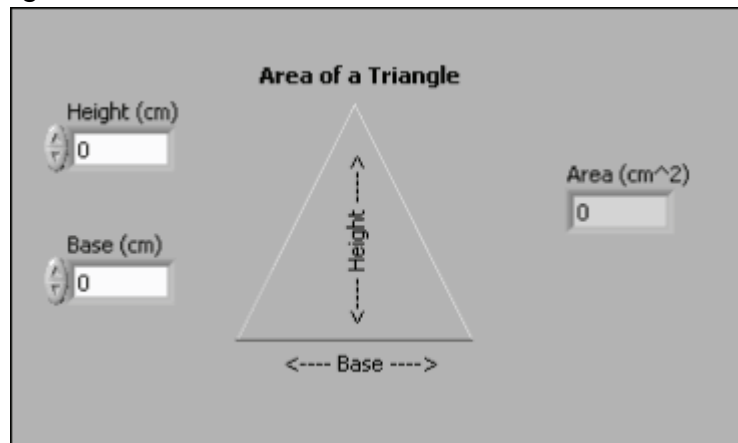


Figura 17. Panel Frontal del Área de un Triángulo

El usuario no cambiará o tendrá acceso a la constante 0.5, así que no aparecerá en el panel frontal a menos que se incluya como documentación del algoritmo. La Figura 18 muestra una posible implementación de este algoritmo en un diagrama de bloques de LabVIEW. Este diagrama de bloques tiene cuatro terminales diferentes creadas por dos controles, una constante y un indicador.

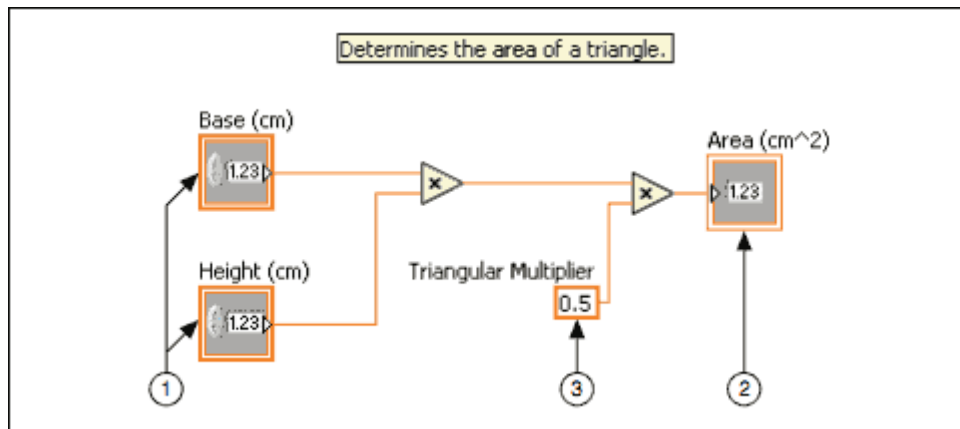


Figura 18. Diagrama de Bloques del Área de un Triángulo con Vista de Terminal de Ícono (1) Controles | (2) Indicador | (3) Constante

Note que las terminales del diagrama de bloques Base (cm) y Altura (cm) tienen una apariencia diferente de la terminal Área (cm<sup>2</sup>). Existen dos características distintivas entre un control y un indicador en el diagrama de bloques. La primera es una flecha en la terminal que indica la dirección del flujo de datos. Los controles tienen flechas que muestran los datos que salen de la terminal, mientras que el indicador tiene una flecha que muestra los datos que entran a la terminal. La segunda característica distintiva es el borde alrededor de la terminal. Los controles tienen un borde grueso y los indicadores tienen un borde delgado.

Puede ver terminales con o sin vista de íconos. La Figura 19 muestra el mismo diagrama de bloques sin usar la vista de ícono de las terminales; sin embargo existen las mismas características distintivas entre los controles y los indicadores.

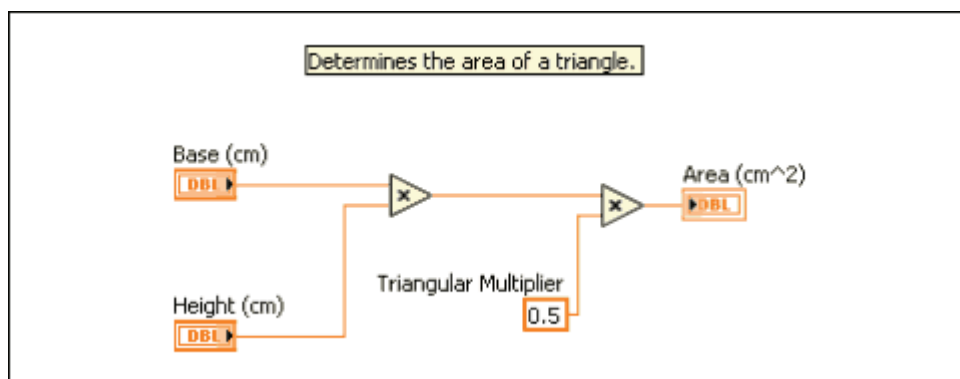


Figura 19. Diagrama de Bloques del Área de un Triángulo sin Vista de Terminal de Ícono



#### 4.4.9 Nodos de Diagrama de Bloques

Los nodos son objetos en el diagrama de bloques que tienen entradas y/o salidas y realizan operaciones cuando el VI se ejecuta. Son análogos a instrucciones, operaciones, funciones y subrutinas en lenguajes de programación basados en texto. Los nodos pueden ser funciones, subVIs, Express VIs o estructuras. Las estructuras son elementos de control de procesos, como Estructuras de Casos, Ciclos For o Ciclos While.

#### 4.4.10 Funciones

Las funciones son los elementos de operación fundamentales de LabVIEW. Las funciones de Suma y Resta en la Figura 15 son nodos de función. Las funciones no tienen ventanas del panel frontal o ventanas del diagrama de bloques pero no tienen paneles conectores. Al hacer doble clic en la función solamente selecciona la función. Una función tiene un fondo amarillo pálido en su ícono.

#### 4.4.11 SubVIs

Después de desarrollar un VI, usted puede usarlo en otro VI. Un VI llamado desde el diagrama de bloques de otro VI es llamado un subVI. Usted puede reutilizar un subVI en otros VIs. Para crear un subVI, necesita desarrollar un panel conector y crear un ícono.

Un nodo de subVI corresponde a una llamada de subrutina en lenguajes de programación basados en texto. El nodo no es subVI, solamente una instrucción de llamada de subrutina en un programa que no es la propia subrutina. Un diagrama de bloques que contiene varios nodos de subVI idénticos llama al mismo subVI varias veces.

Los indicadores y controles de subVI reciben datos y regresan datos al diagrama de bloques del VI que hace el llamado. Cuando hace doble clic en un subVI en el diagrama de bloques, aparece la ventana del panel frontal. El panel frontal incluye controles e indicadores. El diagrama de bloques incluye cables, íconos, funciones, subVIs probables u otros objetos de LabVIEW.



Cada VI muestra un ícono en la esquina superior derecha de la ventana del panel

frontal y la ventana del diagrama de bloques. Un ejemplo del ícono predeterminado se muestra aquí. Un ícono es una representación gráfica de un VI. El ícono puede contener texto e imágenes. Si usa un VI como un subVI, el ícono identifica al subVI en el diagrama de bloques del VI. El ícono predeterminado contiene un número que indica cuántos nuevos VI abrió después de iniciar LabVIEW.



Para usar un VI como un subVI, necesita construir un panel conector, como se muestra arriba. El panel conector es un conjunto de terminales en el ícono que corresponde a los controles e indicadores de ese VI, similares a la lista de parámetros de una función llamada en lenguajes de programación basados en texto. Obtenga acceso al panel conector al dar clic con botón derecho en el ícono en la parte superior derecha de la ventana del panel frontal. Usted puede tener acceso al panel conector desde el ícono en la ventana del diagrama de bloques. Un ícono de subVI tiene un fondo blanco en su ícono.

#### 4.4.12 Express Vis

Los Express VIs son nodos que requieren cableado mínimo ya que usted los configura con ventanas de diálogo. Use Express VIs para tareas de medidas comunes. Consulte el tema de Express VIs de la Ayuda de LabVIEW para más información. Aparecen en el diagrama de bloques como nodos expandibles con iconos rodeados por un campo azul.

#### 4.4.13 Paleta de Funciones

La paleta de Funciones contiene los VIs, funciones y constantes que usted utiliza para crear el diagrama de bloques. Usted tiene acceso a la paleta de Funciones del diagrama de bloques al seleccionar View»Functions Palette. La paleta de Funciones está dividida en varias categorías; usted puede mostrar y esconder categorías para cumplir con sus necesidades. La Figura 20 muestra la paleta de Funciones con todas las categorías expuestas y la categoría de Programación expandida.

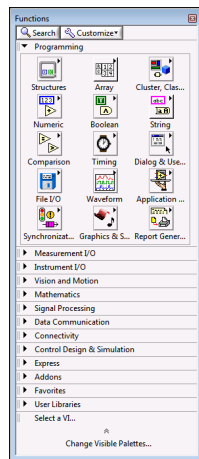


Figura 20. Paleta de Funciones

#### 4.5 Laboratorios remotos

[16] A lo largo de los años han surgido experiencias en el uso de los laboratorios virtuales en distintas áreas de las ciencias. Una de ellas está relacionada con la robótica y específicamente con la simulación de algoritmos de visión artificial y teleoperación de robots industriales. En estas prácticas se emplea un entorno de software que está disponible a través de internet (Torres, et al, 2004). La simulación de dichos ambientes se basa en software java y en VRML (Virtual Reality Modelling Language) el cual es un software estándar para representar escenarios de realidad virtual. Con la ayuda de estas herramientas, los estudiantes son capaces de practicar y obtener secuencias de ejecución de movimientos correctos en los robots de forma remota.

De igual forma, existen experiencias en la implementación de laboratorios virtuales para el estudio de dispositivos electrónicos. La visualización de los datos al estudiante se realiza mediante gráficas en Java, montadas en el equipo servidor por medio de una página web con manejo de bases de datos que permita el control de acceso y la interacción entre el estudiante usuario del laboratorio virtual y el docente encargado. La anterior página web posee el enlace a los Applets y Servlets encargados de capturar los parámetros que serán entregados a la tarjeta micro-controlada y la interfaz para la visualización de los datos retornados. (Ibarra B., et al, 2007). Lo anterior le permite al estudiante interactuar con los equipos electrónicos de los laboratorios a través de internet permitiéndole simular un laboratorio real.

Otras prácticas relacionadas con sistemas expertos son usadas como laboratorio virtual donde se implementa un agente pedagógico virtual el cual posee la experiencia de un educador experto que puede guiar, evaluar y retroalimentar al estudiante en el desarrollo del laboratorio. Para soportar dicho laboratorio se usan herramientas de realidad virtual combinadas con java. (Tzeng, 2001).

También existen experiencias en el uso de laboratorios virtuales en dispositivos móviles con las ventajas que conlleva el ejercicio en términos de portabilidad y la amplia difusión (casi todos los estudiantes tiene un teléfono móvil inteligente). Este modelo tiene restricciones tales como el poder de procesamiento del equipo móvil, el tamaño de la pantalla y la limitación en el uso de la solución, solo por mencionar algunas. El trabajo es un prototipo de laboratorio móvil de química que consta de una parte teórica y otra práctica donde los estudiantes pueden simular

ejercicios de química orgánica a través del procesamiento de datos, imágenes y sonidos (Bottentuit Junior & Clara, 2007).

### Experiencias en el uso de laboratorios virtuales en el ámbito Colombiano

A principio de Marzo de 2010 se presentó el proyecto denominado Red de laboratorios virtuales y tele-operados de Colombia. El proyecto apunta a la creación de una RED que integra los laboratorios virtuales y tele-operados de Colombia, e-Lab Colombia, mediante una plataforma web accedida a través de una red nacional de colaboración científica y académica llamada RENATA la cual se concibió como una herramienta alternativa pedagógica e investigativa, donde los estudiantes pueden desarrollar las actividades prácticas en forma remota para el fomento de las destrezas y habilidades en el manejo de materiales y equipos relacionados con las temáticas de sus campos de formación, y a la vez, permitir el desarrollo de nuevas investigaciones científicas en el país que requirieran del uso de laboratorios especializados. (Red Nacional Académica de Tecnología Avanzada RENATA, 2012).

En ese comienzo se formuló que los laboratorios virtuales, serian simulaciones multi-mediales con acceso vía RENATA, diseñados bajo el concepto de Objetos Virtuales de Aprendizaje (OVA), que permiten a los estudiantes realizar prácticas que simulen, lo más parecido posible, las condiciones que se presentan en un escenario real de práctica y permitan la manipulación remota de dispositivos más especializados ubicados en diferentes laboratorios presenciales de las instituciones de educación superior participantes, que permitirá la colaboración en tiempo real con la comunidad académica e investigativa del país.

Adicional al caso anterior, universidades como UNAD, Universidad del Valle, Universidad de Antioquia y Universidad del Cauca han incursionado en la utilización de tecnologías y redes de conocimiento similares, las cuales buscan compartir recursos, reducir costos por conceptos de adquisición, mantenimiento y reparación de equipos, compra de materiales y reactivos y solucionar las dificultades generadas por la poca disponibilidad de espacios donde desarrollar las actividades prácticas en las asignaturas de las ciencias básicas en los programas de ingenierías y las ciencias de la salud, al igual que en el equipamiento especializado para la investigación científica.

A finales de Marzo de 2008, la universidad Santo Tomás presento un proyecto para la implementación de laboratorios virtuales con énfasis en comunicaciones móviles el cual consiste en crear una plataforma basada en RENATA donde se desarrollen laboratorios virtuales implementando un módulo de comunicaciones móviles administrado por un sistema de contenidos.

Entre sus objetivos específicos estaban los siguientes:

- Implementar y desarrollar un laboratorio virtual de comunicaciones móviles sobre la una red formada por un grupo de universidades que conecta la comunidad académica y varios centros de investigación alrededor del mundo.
- Ofrecer a los alumnos de la Universidad Santo Tomás herramientas más accesibles para su aprendizaje y que vayan a la vanguardia con el avance tecnológico mundial.
- Crear vínculos con otras universidades y ampliar las oportunidades tanto para los estudiantes como para la misma Universidad Santo Tomás.
- Desarrollar e implementar aplicaciones que trabajen en red y de esta forma compartir con otras instituciones educativas los proyectos de investigación generados en la universidad.
- 

El proyecto finalizo a finales de Septiembre de 2009. (Universidad Santo Tomas, 2009) La comunidad educativa virtual del SENA ha enmarcado su objetivo en ofrecer a los ciudadanos educación virtual en los diferentes tipos de modalidades educativas, haciendo uso de las TIC para no limitar el conocimiento y aprendizaje debido a las barreras del tiempo y la ubicación geográfica, eliminando así los costos de tiempo, desplazamiento y rigidez de horarios que influyen en la calidad de la formación recibida por el estudiante. Una de las actividades desarrolladas hasta el momento ha sido la implementación de laboratorios de control remoto, donde utilizando las herramientas de formación virtual, los estudiantes pueden desarrollar prácticas e interactuar con equipos (sistemas de micro controladores, equipos de automatización, refrigeración, etc.)

Igualmente, se han hecho convenios entre algunas instituciones de educación superior de Colombia tales como la Fundación Universitaria Luis Amigó y la Institución Universitaria de Envigado. Estas Universidades formularon un proyecto para desarrollar una metodología para estructurar, diseñar, implementar y mantener laboratorios virtuales. El convenio fue firmado en abril de 2012. (Convenio específico de Colaboración y mutua ayuda entre la Institución Universitaria de Envigado y la Fundación Universitaria Luis Amigó Nro 50, 2012).

## Modelo de Laboratorios virtuales para programas 100% Virtual

Entre las características de un programa de Ingeniería Informática que se oferta bajo modalidad 100%, es que se soporta 100% en las TIC, lo cual conlleva un reto bastante exigente para efectos de implementar un esquema coherente y con calidad en el tema de laboratorios. En la figura 1 se observa la metodología propuesta para el desarrollo del proyecto.

El cambio del laboratorio presencial al laboratorio remoto representa para el estudiante un cambio de contexto en la forma en la que éste experimenta, lo que determina un cambio en la forma en la que éste construye su aprendizaje.

[15] Un laboratorio remoto es un ambiente de experimentación que permite a sus usuarios interactuar con equipos reales ubicados remotamente. En el contexto de los laboratorios remotos en Ingeniería, existe un extenso cuerpo de trabajo investigativo relacionado con las alternativas de software y hardware para su implementación de laboratorios remotos, y menos atención han recibido el aspecto instruccional (Gustavsson et al., 2009, p.263) y los resultados de aprendizaje. De hecho, atender estos aspectos no es una tarea trivial (Cooper & Ferreira, 2009, p. 347).

Los laboratorios remotos en la unidad están compuestos por: Laboratorios de Domótica y CISCO para la cadena, Planta Piloto (Bio-fermentador) para la cadena de Alimentos y la Celda Integrada de Manufactura CIM para las cadenas de Industrial y ETR.

## 5. MATERIALES Y MÉTODOS

### 5.1 Materiales

RECURSO	DESCRIPCIÓN	PRESUPUESTO
Equipo Humano	Estudiantes de la UNAD	0
Equipos y Software	Computador y modulo M2CI	0
Viajes y Salidas de Campo	NA	
Materiales y suministros	NA	
Bibliografía	NA	
<b>TOTAL</b>		

### 5.2 Metodología

Para el desarrollo del proyecto se usará el modelo experimental, donde se realizaran acciones y se observaran sus consecuencias. Se parte de conceptos relativamente limitados, pero bien definidos; para que a partir de prácticas presenciales ya desarrolladas, se puedan alcanzar los mismos resultados en prácticas remotas no asistidas.

Con el módulo M2CI, se han validado algunas prácticas presenciales haciendo uso de la unidad de PLC instalada en el módulo. Ahora, es implementar tanto un desarrollo de software como de hardware, si es el caso; para que remotamente se pueda, el estudiante, conectar y realizar la práctica propuesta.

Esto, necesariamente conlleva a crear una arquitectura de laboratorio remoto y/o seleccionar la mejor opción de las que ya han sido probadas y utilizada, ejemplo, la de la Universidad de Deusto, WebLabDeusto [3].

Para poder identificar la estructura y funcionamiento del módulo M2CI se recurrió primero al manual que dio la empresa que lo fabrico; para mirar que componentes electrónicos constituía, como estaba su cableado estructurado y que terminales se tenían que usar y como él los tenia nombrados, para no realizar alguna mala manipulación y generar una falla en los equipos y dispositivos.



Como segunda medida se realizó la consulta por internet de cada dispositivo electrónico que se usó para la práctica del control del horno recurriendo a sus datasheet.

Para el sistema de conmutación usamos como herramienta de apoyo el modulo que lo acompaña el PLC en el módulo, en este caso se realizó por medio de los equipos CP243-1 que es un módulo de Ethernet de la empresa siemens que nos Da el apoyo en parte de comunicación de red y también el EM231-4 (modulo con entradas análogas) el cual posee internamente varios convertidores análogos digitales; con todo eso podemos realizar el control de la variable temperatura.

Para poder encender y controlar los demás dispositivos que nos ayudarán al control del horno que está en el módulo M2CI. Se recurrió a las entradas y salidas digitales que poseen el PLC CP 224 y la ayuda de varios relés mecánicos que vienen en él.

Parar poder para acceder y programar el PLC y dar la configuración adecuada se recurrió al software que brinda la compañía que lo a fabricó en este caso Siemens llamado el microwin Step7 el cual es el encargado de ayudar a programar en lenguaje KOP, depurar el programa creado por nosotros y también de gestionar tanto la carga del programa la PLC y su funcionamiento físico.

Para poder realizar el cableado se recurrió a las borneras marcadas que trae el modulo y poder manejar los dispositivos que hacen funcionar y monitorear el horno.

Por medio del software Labview se incorporó una interfaz gráfica amigable para el usuario que el trae, el cual se contempló el control del horno tanto el valor de la temperatura que se desea tener funcionando como le tiempo de prendido del él.

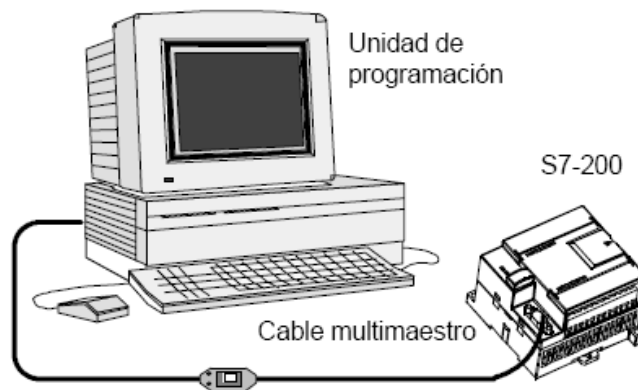
La programación tanto del control del horno como de los indicadores gráficos que se incorporaron en tanto físico como digital se hizo bajo esquema de diagrama de bloques que posee labview.

## 6. DESARROLLO DEL PROYECTO

Para poder apreciar todo el panorama de la aplicación en uso industrial se hará con una parte del módulo M2CI, con la parte del control de la temperatura del horno eléctrico que trae incluida, realizado su adquisición de datos, procesamiento de datos y control por medio de los diferentes módulos que él tiene; que este caso es el uso del PLC y los módulo de Ethernet y el análogo que va conectado a una termo culpa que el trae.

También se hará uso de los relés mecánicos para el control de horno y de su cooler, el accionamiento de los indicadores luminosos. No dejando atrás las aplicaciones del software de microwin, y los paquetes de labview, a continuación se dará un paso a paso del proceso de configuración de los dispositivos para poder llegar al control de ellos en labview.

Inicialmente la comunicación entre el S7-200 y el PC equipado con el software *Step 7 MicroWin*, debe hacerse a través de uno de los puertos USB. Se utiliza para ello un cable apropiado con un convertidor RS 485 a RS 232. El cable se denomina *cable multimaestro USB/PPI*. El protocolo de comunicación es propietario de Siemens y se llama protocolo *PPI*, que significa *Poin to Point Interface*.



La figura 21 esquemas de conexión del S7-200 al PC utilizando un enlace PPI

## Descripción del módulo de comunicación Ethernet modelo CP 243-1

La comunicación entre el centro del control de la E. de C. y los autómatas programables encargados del proceso de conexión o desconexión de un cliente se lleva a cabo mediante una red Ethernet bajo protocolo TCP/IP. Esto significa que el autómata elegido, el S7 226, debe estar provisto de una interfaz de comunicación de ese tipo. El CPU S7-226 no posee esta interfaz integrada, pero si puede añadirse un módulo de comunicación adecuado a esa función.

El módulo utilizado es el modelo CP 243-1, cuyo aspecto físico externo se ve ilustrado en la figura 22.



Figura 22. Aspecto exterior del módulo de comunicación Ethernet CP 243-1

Con el módulo CP 243-1 se puede establecer comunicación Ethernet Industrial de tres maneras distintas:

1. Conexión a un PC a través del paquete *Step 7 MicroWIN*
2. Conexión a otros autómatas de la serie S7 de Siemens
3. Conexión a aplicaciones basadas en servidores OPC

## 6.1 Conexión a la CPU-224

Para la conexión inicial con el S7-200 debe asegurarse que el PLC y el PC estén conectados como se mostró en la figura XX. Inicialmente la conexión del PC al S7-200 se hace a través del puerto Usb RS 485 del PLC, usando el protocolo PPI. Todo CPU S7 200 viene de fábrica con una dirección predeterminada, que es la número 2.

Una vez conectado físicamente el PC al CPU S7 200, dirección 2, mediante protocolo PPI, se configurará el CPU para conexión vía Ethernet Industrial asignándole una determinada dirección IP. Esta nueva configuración se cargará en el S7 226 vía puerto PPI, y de ahí en adelante ya podrá accederse al CPU igualmente vía Ethernet.

A continuación se describen en detalle los pasos necesarios para poder comunicarse con el S7 200 vía Ethernet.

### Paso 1:

Arrancar el paquete de programación *Step7-Micro/WIN* y crear un nuevo Proyecto. Ver figura 23.

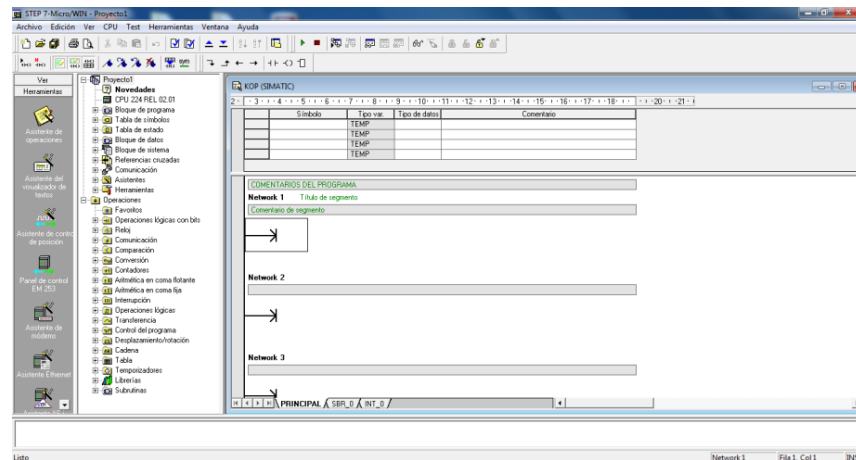


Figura 23. Creación de un nuevo proyecto con Step 7 Micro WIN

## Paso 2:

Elegir el comando *Herramientas* → *Asistente Ethernet*. Ver figura 24.

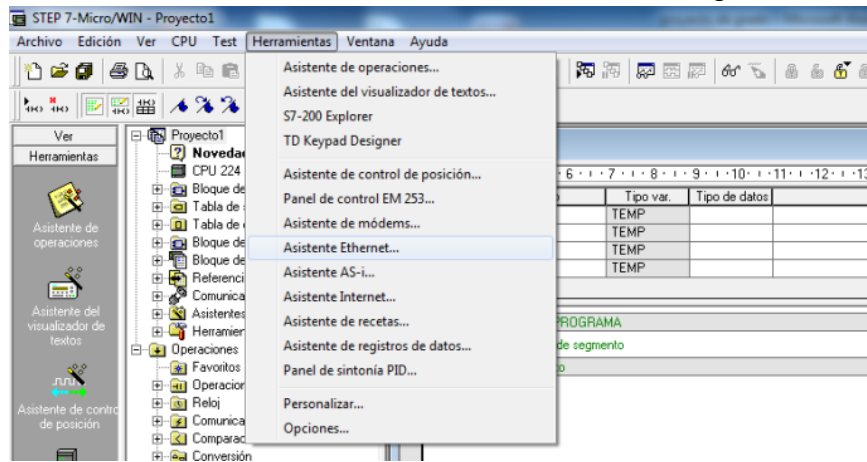


Figura 24. Elegir comando Asistente Ethernet

Este asistente ofrece la opción de configurar la comunicación Ethernet Industrial de una manera rápida y sencilla. Al elegir el comando se muestra la ventana de la figura 25. Si el módulo Ethernet está conectado y alimentado, entonces debe reconocerse automáticamente.

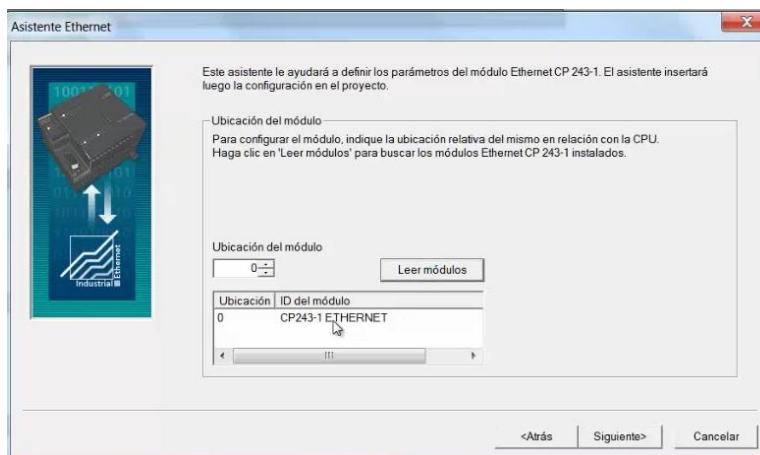


Figura 25. Primera ventana del asistente Ethernet Industrial

### Paso 3:

Se pulsa el botón *Siguiente* y en la ventana que aparece (figura 26) se definen los parámetros del módulo CP 243-1 tales como su dirección IP, la máscara de subred y la puerta de enlace (opcional). Donde se fija una dirección IP 192.168.0.2

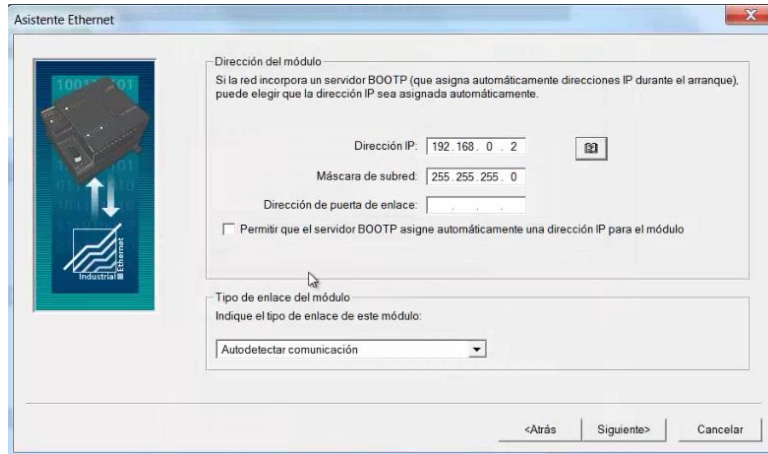


Figura 26. Parámetros de la red Ethernet

### Paso 4:

Se pulsa el botón *Siguiente* y en la ventana que aparece (figura 27) se especifica el número de enlaces que tendrá el módulo CP 243-1. Se especifica un enlace.

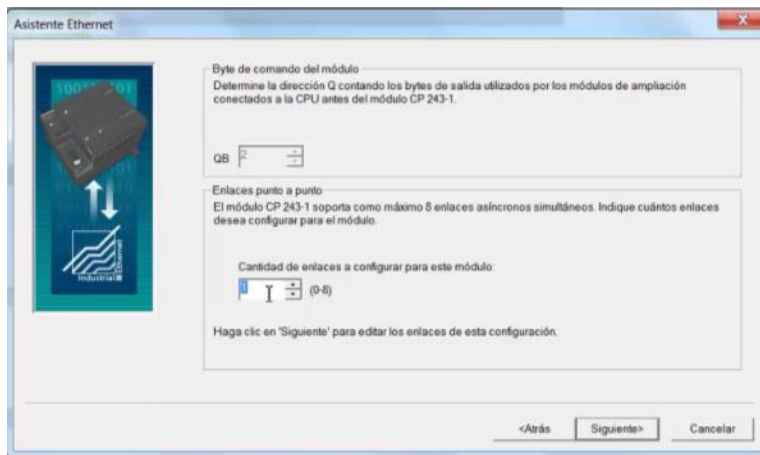


Figura 27. Determinación del número de enlaces

## Paso 5:

Se pulsa el botón *Next* y en la ventana que aparece se establecen las opciones tal cual como están indicadas en la figura 28. Observar que automáticamente se genera en el cuadro TSAP la dirección 10.00 que se utilizará luego para el servidor OPC.

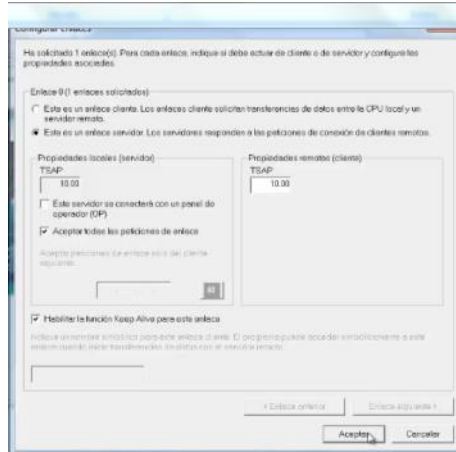


Figura 28. Configuración de enlaces

## Paso 6:

Se pulsa el botón *OK* y en la ventana que aparece (figura 29) se especifican las opciones que muestra la figura (protección CRC activada e intervalo Keep Alive en 30 s). La opción de protección CRC impide que el usuario modifique la configuración del módulo en modo de ejecución.

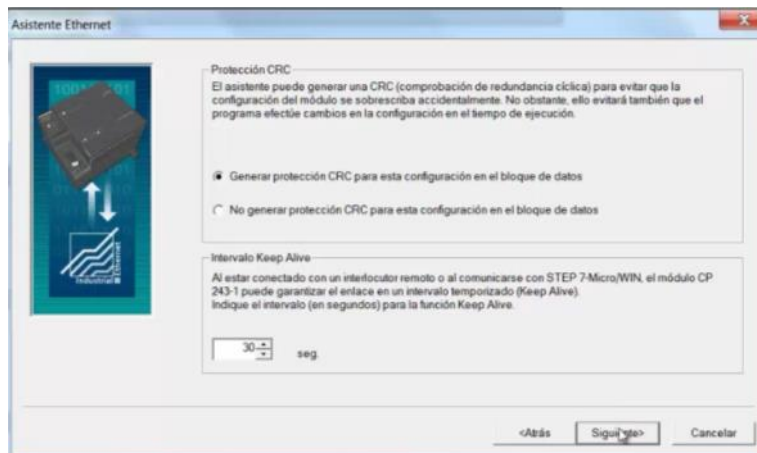


Figura 29. Configuraciones CRC e intervalo Keep Alive

## Paso 7:

Se pulsa el botón *Siguiente*. Aparece la ventana de la figura 30 donde se selecciona el principio de una tabla de bytes que hay que reservar para uso privado del módulo de comunicación. En este caso se escogió como dirección inicial de la tabla la *VB477*.

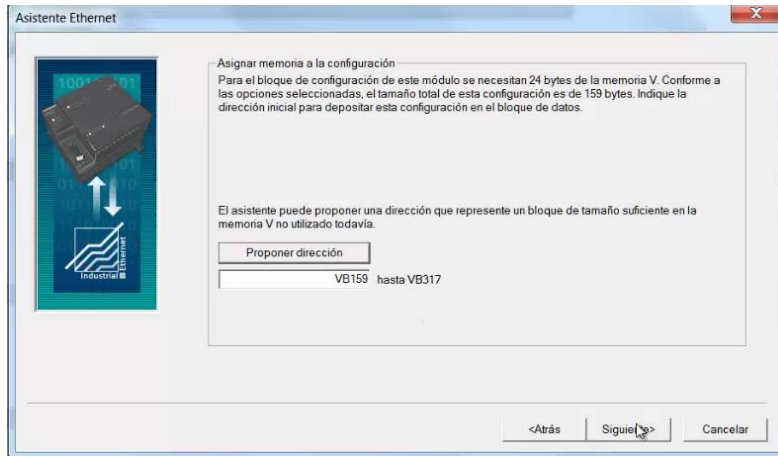


Figura 30. Asignación de tabla de bytes para uso del módulo

## Paso 8:

Se pulsa el botón *Siguiente*. Se muestra la última ventana del asistente. Además se crea automáticamente una subrutina llamada *ETHx\_CTRL* que permite chequear el estado de la conexión vía Ethernet. Ver figura 31.

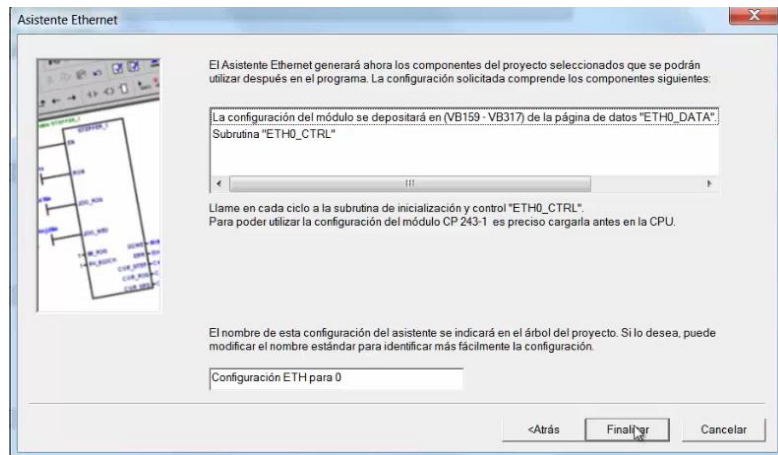


Figura 31. Última ventana del asistente Ethernet Industrial



### Paso 9:

Se crea automáticamente la rutina *ETHx\_CTRL*. Se coloca esta rutina en el programa principal para que se encargue de inicializar el módulo Ethernet y chequear cualquier error que pueda suceder, monitorizando de esa forma el estado de la conexión durante la comunicación. Observar que las salidas del bloque se asignan, respectivamente, a las variables *M2.1*, *MW20* y *MW22*. Ver figura 32.

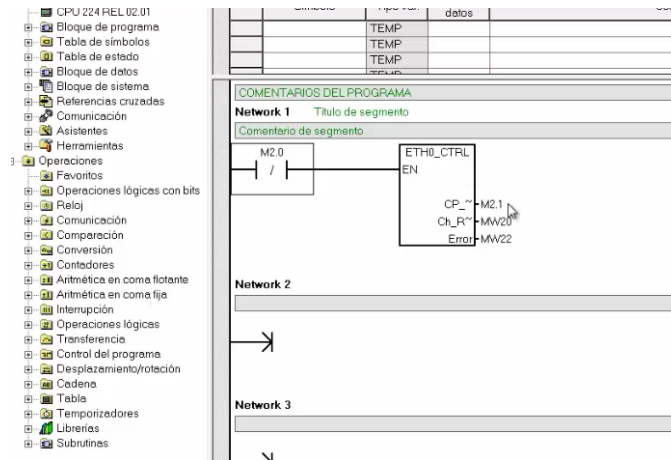


Figura 32. Subrutina *ETHx\_CTRL*

La rutina se debe llamar en cada ciclo automático con la finalidad de comprobar si existe una nueva configuración. De esta manera si ha cambiado la configuración el módulo la adoptará, activándose así la señal *CP\_Ready*. El estado del canal se verifica con los bits de la palabra asociada a *Ch\_Ready*. Cada bit corresponde a un canal. Si el bit está en 1, el canal está activado y se estableció un enlace. Si el enlace presenta algún error, se desplegará en *Error* bajo un código específico.

### Paso 10:

Una vez creada la configuración para el enlace Ethernet, se carga en el CPU. La ventana de la figura 33 muestra cómo cargarla. El CPU debe estar energizado, en modo STOP y el cable multimaestro USB/PPI debidamente conectado.

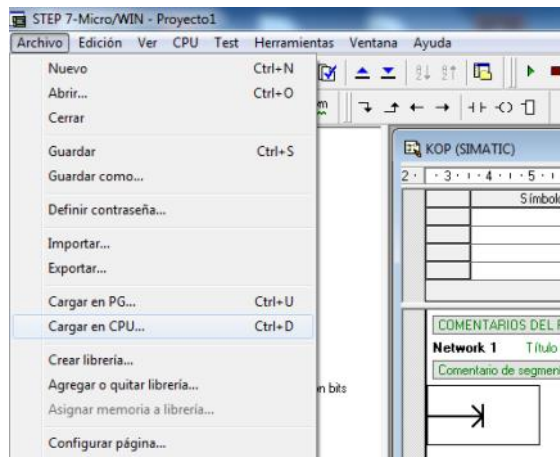


Figura 33. Cargar la configuración en el CPU o PLC

### Paso 11:

A continuación puede ahora se intenta la comunicación con el PLC. Para lograrlo se entra primero en el menú *Comunicación*, como sugiere la figura 34.



Figura 34. Escoger la comunicación con el PLC

### Paso 12:

A continuación se escoge la opción *Ajustar Interfaz PG/PC*, como sugiere la figura 35.

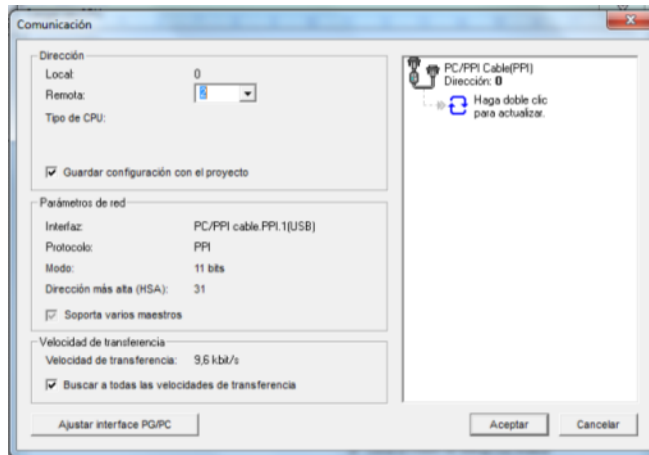


Figura 35. Escoger la opción de Ajustar Interfaz PG/PC

### Paso 13:

A continuación en la ventana que aparece se elige *PC/PPI cable* posteriormente se da aceptar, como sugiere la figura 36.

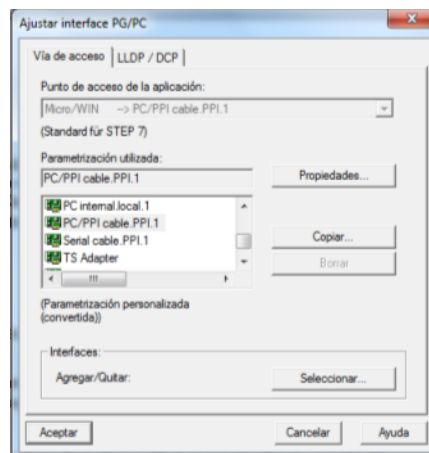


Figura 36. Configuración del puerto de comunicación.

### Paso 14:

AL aceptar la ventana de la figura 36. Ir nuevamente a la opción de *Comunicación* y en la ventana que aparece luego Hacer doble clic sobre el símbolo para actualizar (2- en la figura 37) para intentar establecer una comunicación con el

PLC. De ser exitoso el intento, debe aparecer en la misma parte derecha de la ventana el icono del CPU S7 224. (3- en la figura 37). Y le damos luego aceptar

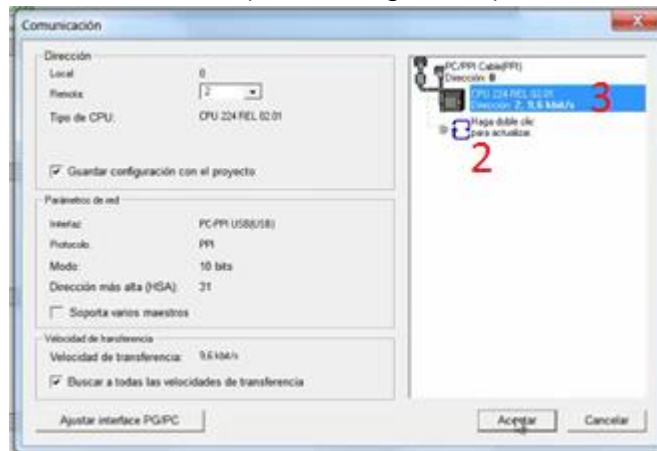


Figura 37. Configuración del enlace de comunicación con la CPU

### Paso 15:

Luego nos aparece nuevamente el cuadro de cargar CPU el cual está habilitado el botón de cargar el cual le damos clic encima del el para proceder a cargar la configuración del módulo de Ethernet, la figura 38.

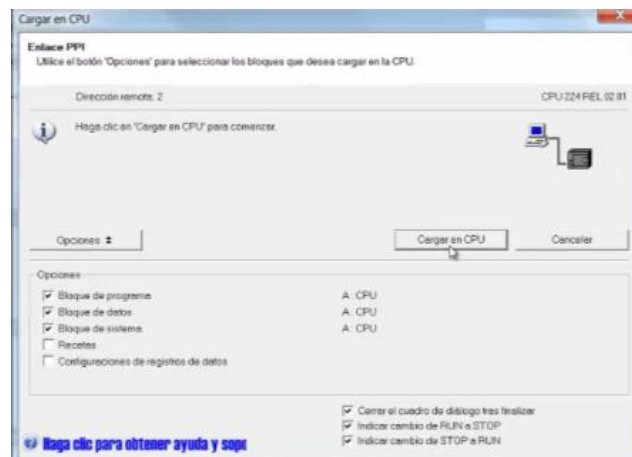


Figura 38: Carga del programa a la CPU

## Paso 16:

Luego nos aparece una ventana emergente de modo STOP donde se selecciona aceptar en la figura 39.

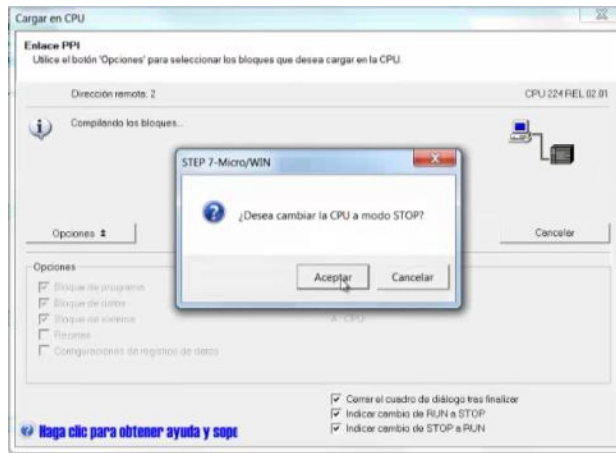


Figura 39. La ventana emergente de modo STOP

## Paso 17:

Después de realizar la configuración y carga de los protocolos para poder que se comunicara el PLC con el módulo CP 243-1 “Modulo Ethernet” se procede a abrir el programa desarrollado en lenguaje KOP dentro del entorno de microwin step7 como se ilustra en la figura 40

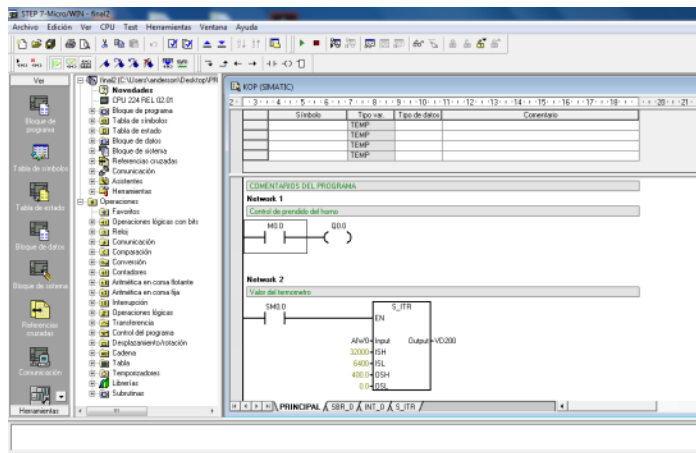


Figura 40. Desarrollo de programación en KOP

### Paso 18:

Nuevamente se carga en el CPU. La ventana de la figura 41 muestra cómo cargarla. Donde la CPU y el módulo Ethernet debe estar energizado, en modo STOP y conectado el cable de red a la tarjeta de red del PC.

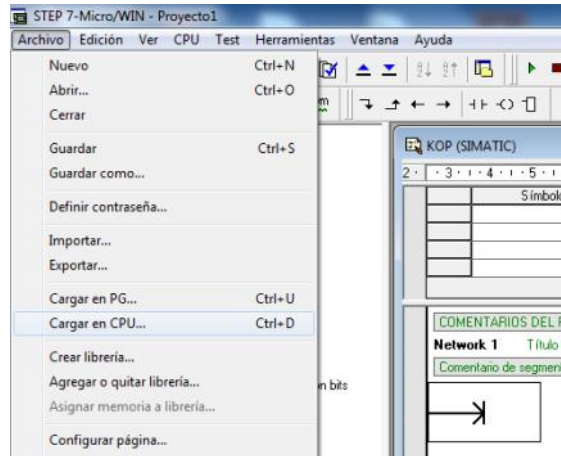


Figura 41. Cargar la configuración en el CPU o PLC

### Paso 19:

A continuación puede ahora se intenta la comunicación con el PLC. Para lograrlo se entra primero en el menú *Comunicación*, como sugiere la figura 42.

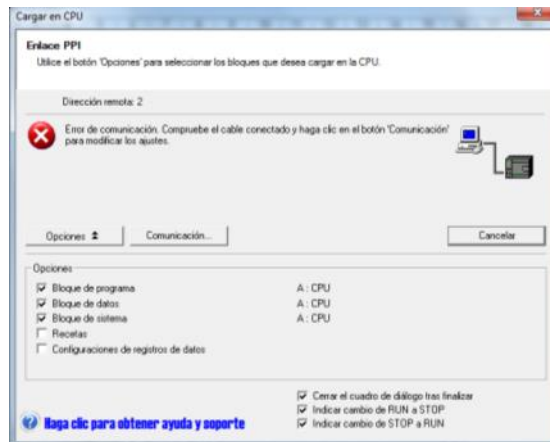


Figura 42. Escoger la comunicación con el PLC

## Paso 20:

A continuación se escoge la opción *Ajustar Interfaz PG/PC*, como sugiere la figura 43.

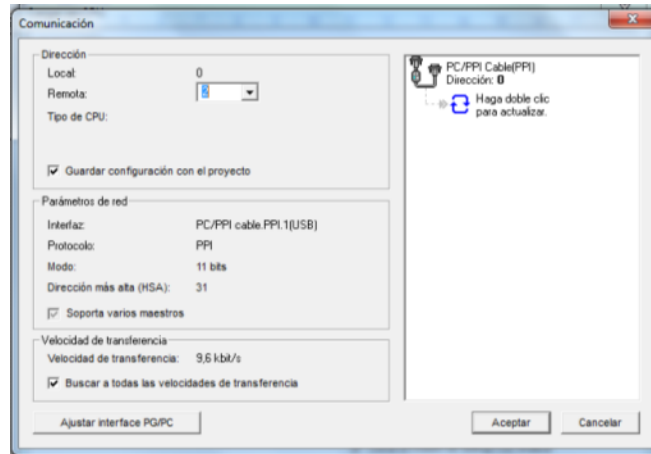


Figura 43. Escoger la opción de Ajustar Interfaz PG/PC

## Paso 21:

A continuación en la ventana que aparece se elige la opción del nombre de la tarjeta de red que termina en TCP/IP Auto, posteriormente se da propiedades, como sugiere la figura 44.

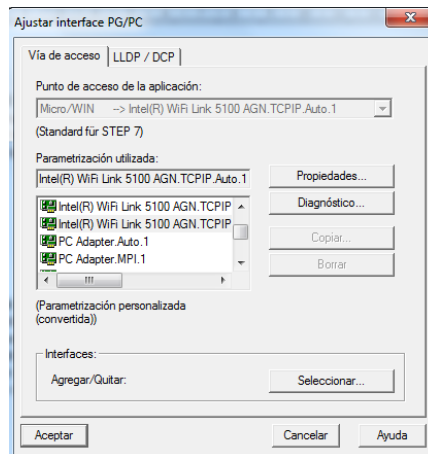


Figura 44. Configuración del puerto de comunicación.

## Paso 22:

A continuación en la ventana que aparece las propiedades de la tarjeta de red donde le damos clic en propiedades de red.. , como sugiere la figura 45.



Figura 45. Configuración propiedades de red.

## Paso 23:

AL dale clic en las propiedades de red de la figura 45. Aparece la opción de comunicación de área local le damos doble clic como ilustra en (1- en la figura 46), luego nos emergerá una ventana y le seleccionamos la opción de protocolo de internet versión 4 lo seleccionamos (2- en la figura 46) y luego le damos propiedades (3- en la figura 46), emerge nuevamente otra ventana e introducimos manualmente la dirección IP 192.168.0.20 y mascara de red de 255.255.255.0 y le damos aceptar (4- en la figura 46). Y por último seremos las ventanas anteriores.



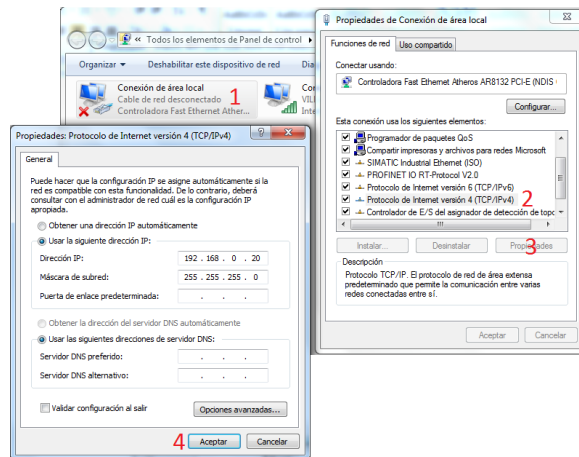


Figura 46. Configuración de la tarjeta de propiedades de red del PC.

#### Paso 24:

A continuación en la ventana que aparece las propiedades de la tarjeta de red le damos clic en OK, como sugiere la figura 47.



Figura 47. Configuración propiedades de red.

#### Paso 25:

A continuación en la ventana que aparece se elige la opción Aceptar para configuración de comunicación con el PLC, como sugiere la figura 48.

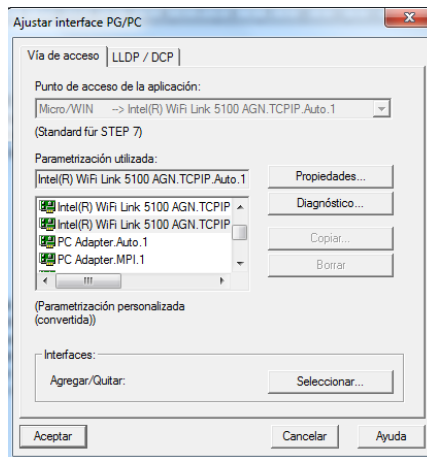


Figura 48. Configuración del puerto de comunicación.

### Paso 26:

AL aceptar la ventana de la figura 48. Ir nuevamente a la opción de *Comunicación* y en la ventana que aparece luego le damos clic donde se muestra (1- en la figura 49) e introducimos la dirección IP que establecimos en la configuración del asistente de Ethernet, luego se hace doble clic sobre el símbolo para actualizar (2- en la figura 49) para intentar establecer una comunicación con el PLC. De ser exitoso el intento, debe aparecer en la misma parte derecha de la ventana el icono del CPU S7 224. (3- en la figura 49). Y le damos luego aceptar

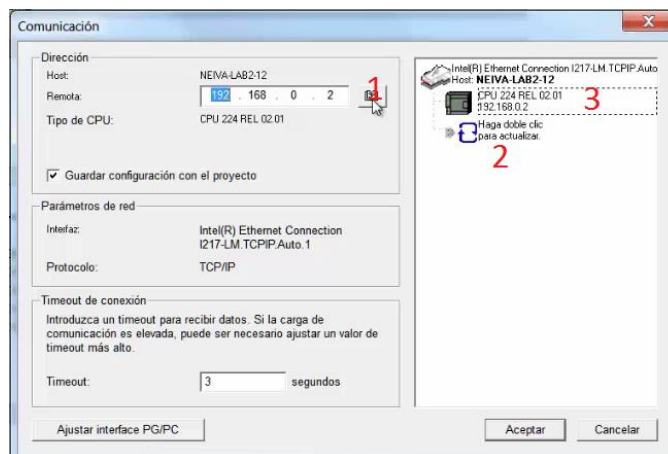


Figura 49. Configuración del enlace de comunicación con la CPU

## Paso 27:

Luego nos aparece nuevamente el cuadro de cargar CPU el cual está habilitado el botón de cargar el cual le damos clic encima del el para proceder a cargar el programa desarrollado en KOP, la figura 50.

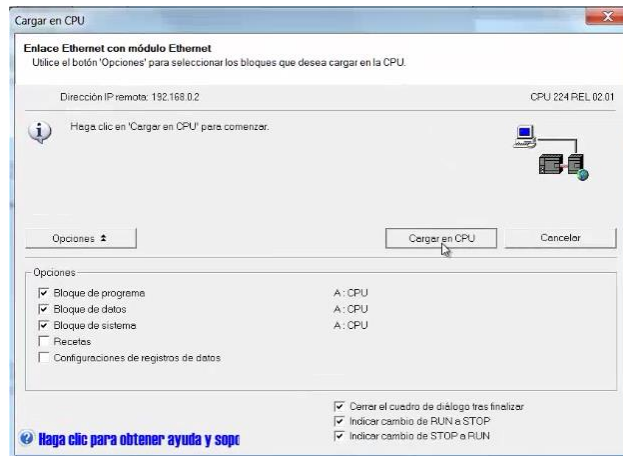


Figura 50. Carga del programa a la CPU

## Paso 28:

Luego nos aparece una ventana emergente de modo STOP donde se selecciona aceptar en la figura 51.

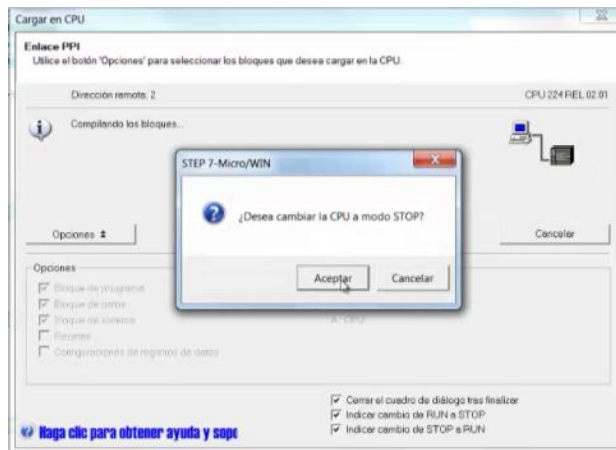


Figura 51. La ventana emergente de modo STOP

Después de realizar todo este proceso de carga de datos a la CPU por medio del módulo Ethernet CP 243-1. Procedemos a enlazar las variables por medio de un software de la empresa NI “National Instruments” llamado NI OPC server el cual nos ayuda con la gestión de la comunicación entre el PC y el PLC por medio de la tarjeta de red para que se pueda manejar de forma remota, que se explicara cómo se configura con este tipo de programa y poder garantizar una buena conexión con el PLC.

A continuación se explicara paso a paso cómo se configura por medio del uso de la tarjeta de red

### Paso 29:

Abrimos el programa NI OPC server, Abrir un nuevo proyecto usando el botón New Project ubicado en la parte superior izquierda. Aceptar, cerrar y desconectar clientes activos en caso que fuera necesario como se muestra en la figura 52.

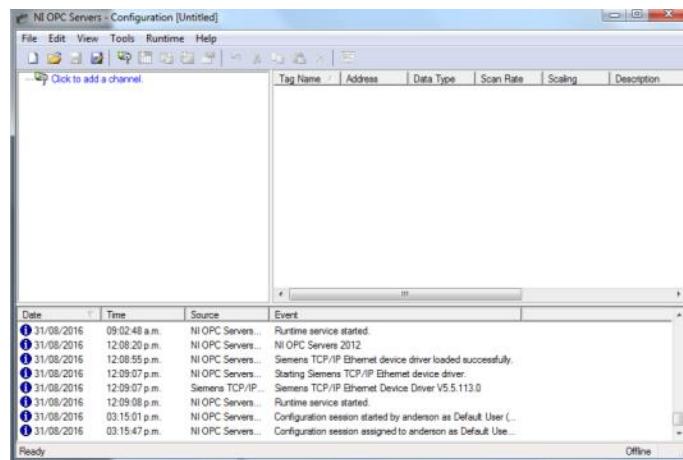


Figura 52. Programa NI OPC Server

### Paso 30:

Se Añade un canal haciendo clic en la figura del conector Click to add a channel en la parte izquierda superior de la pantalla. Como se ilustra en la figura 53 Un canal es una forma de comunicación del computador con el hardware, especificando además qué tipo de PLC se utilizará.

### Paso 31:

Después de agregar un nuevo canal nos emerge una ventana de identificación del canal en donde le damos un nombre al canal. En este caso se utilizará el nombre S7-200. Presionar el botón Siguiente. Figura 53.

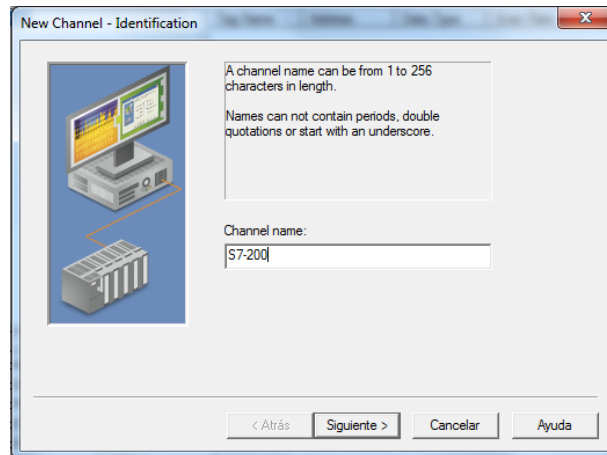


Figura 53. Asignación de identificación del canal

### Paso 32:

En el siguiente pantallazo sale una lista desplegable, selecciona Siemens TCP/IP Ethernet y se presiona el botón Siguiente, como se muestra en la figura 54

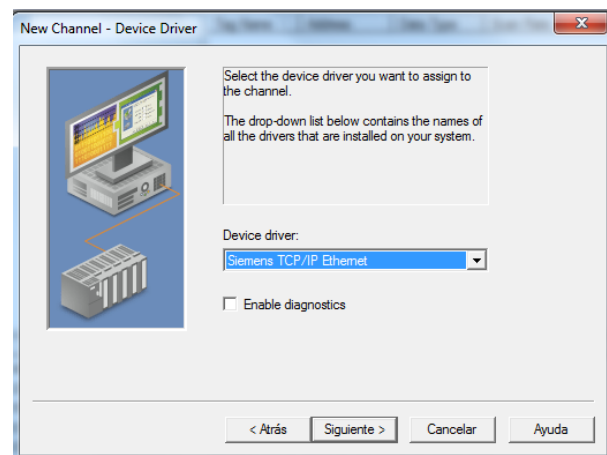


Figura 54. Configuración del modo de comunicación

### Paso 33:

Ahora se selecciona la dirección IP con la cual anteriormente se configuro la tarjeta de red del PC y se selecciona como se muestra en la figura 55 y se procede a dar clic en siguiente.

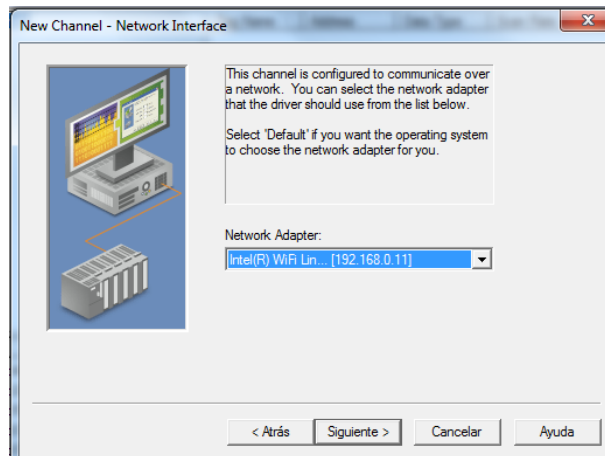


Figura 55. Selección de la dirección IP de la tarjeta de red

### Paso 34:

Ahora configuramos el método de optimización en este caso lo dejamos por defecto y le damos siguiente como esta en la figura 56.

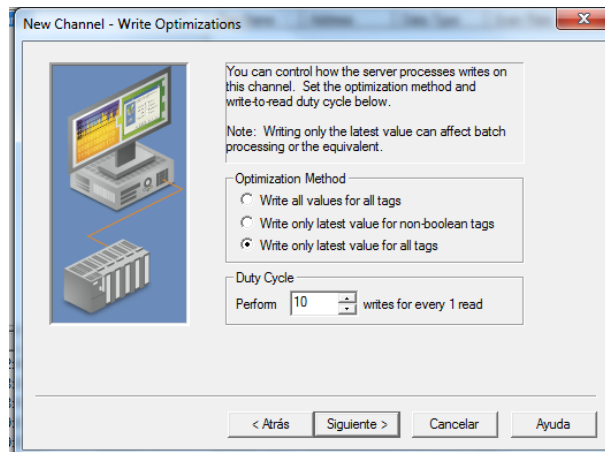


Figura 56. Selección del método de optimización

### Paso 35:

Luego Aparece la pantalla el resume la configuración del canal. Y le damos finalizar, figura 57

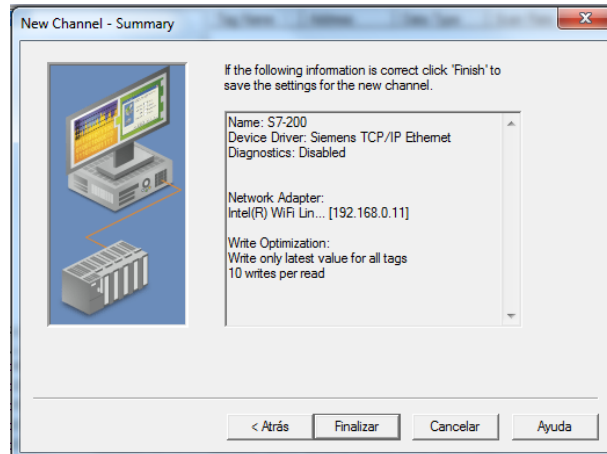


Figura 57. Configuración del canal

### Paso 36:

Creado el canal de comunicaciones se pueden conectar varios equipos, a continuación es necesario agregar un dispositivo. Presionar Click to add a device en la parte izquierda superior de la pantalla, figura 58.

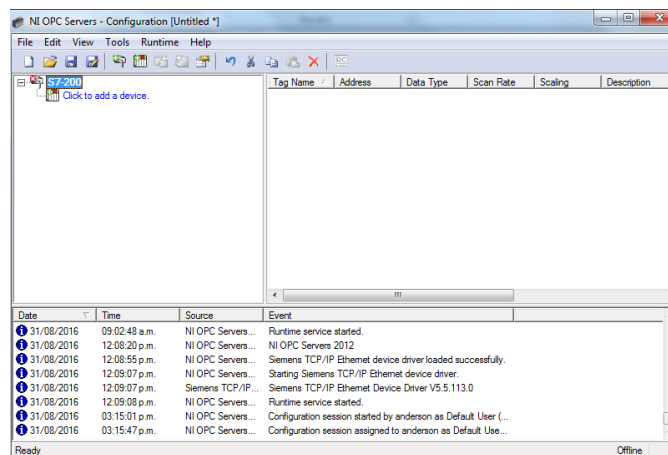


Figura 58. Pantallazo para agregar canales

### Paso 37:

LE asignamos un nombre al dispositivo. En este caso se ha nombrado CP224. Presionar el botón Siguiente. Como se muestra en la figura 59.

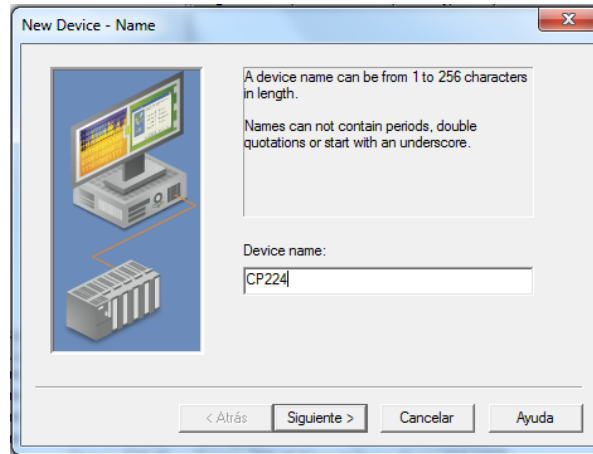


Figura 59. Asignación del nombre

### Paso 38:

Ahora se Selecciona el modelo del PLC nuestro caso es S7-200. Y le damos siguiente, Como se muestra en la figura 60

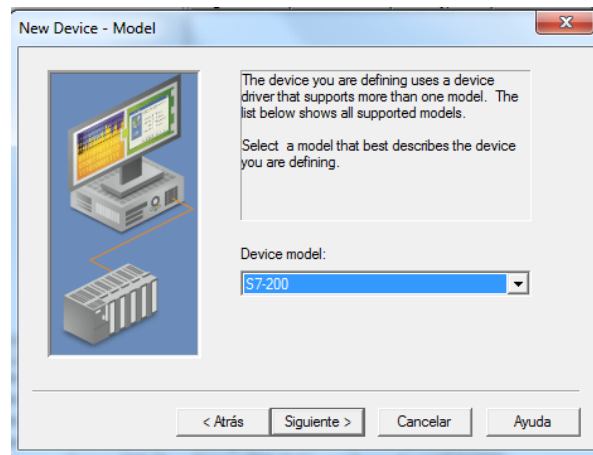


Figura 60. Selección del modelo de PLC



### Paso 39:

Ahora introducimos la dirección IP con la cual configuramos el módulo de Ethernet CP 343-1 en este caso 192.168.0.2 y le damos siguiente, como se muestra en la figura 61.

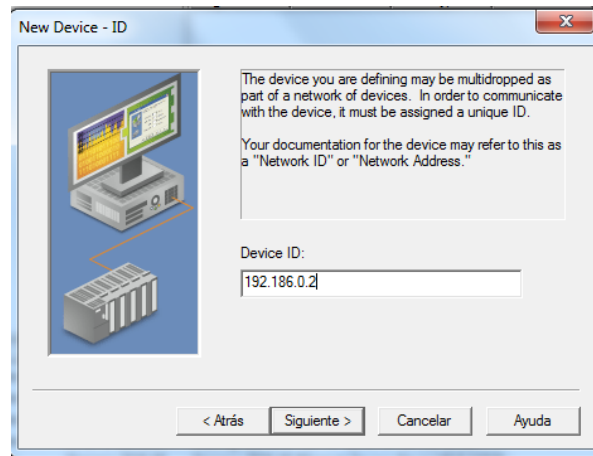


Figura 61. Introducción de la dirección IP al canal

### Paso 40:

Se Configura los parámetros de tiempo de comunicación. Request timeout es el tiempo que espera el driver sin comunicación antes de emitir una falla. Se deja los valores predeterminados. Presionar el botón Siguiente. Como se muestra la figura 62.

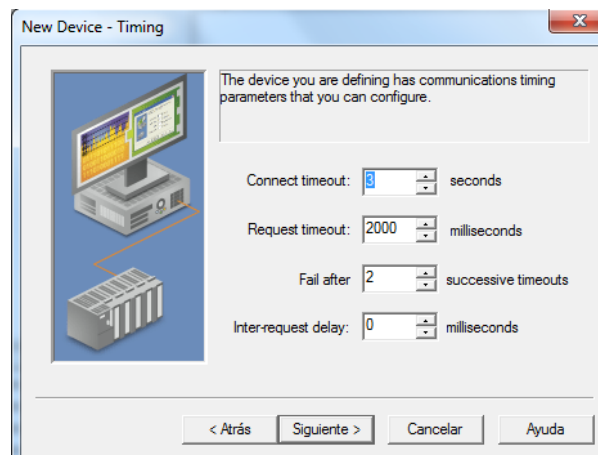


Figura 62. Configuración del Timing.

### Paso 41:

Se puede activar la opción Auto Demotion para que el driver pueda intentar reconectar el dispositivo en caso de una pérdida de comunicación, lo dejamos por defecto como esta y Presionamos el botón Siguiente. Como se muestra la figura 63.

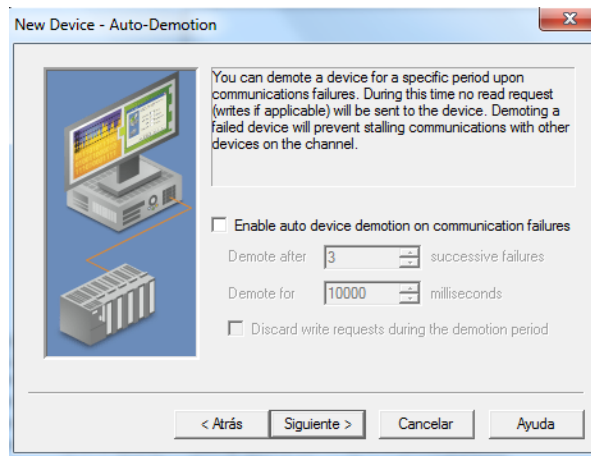


Figura 63. Configuración del Auto Demotion

### Paso 42:

Se configura los parámetros de comunicación en este caso se deja los que están por defecto y se le da siguiente, como muestra en la figura 64.

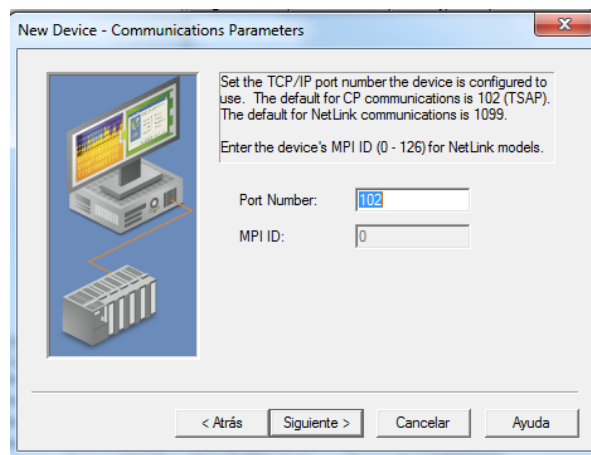


Figura 64. Configuración los parámetros de comunicación

### Paso 43:

Se configura los parámetros de los puerto de comunicación en este caso se deja los que están por defecto y se le da siguiente, como muestra en la figura 65.

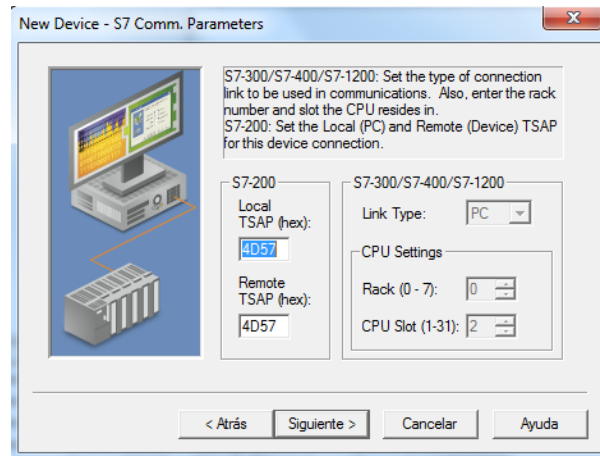


Figura 65. Configuración los parámetros de los puerto de comunicación

### Paso 44:

Se configura las opciones de direccionamiento en este caso se deja los que están por defecto y se le da siguiente, como muestra en la figura 66.

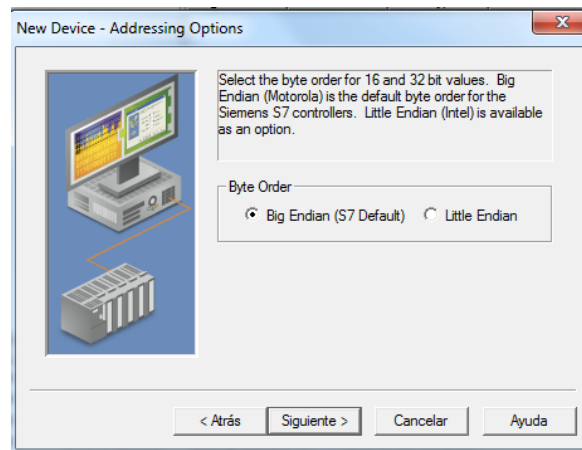


Figura 66. Configuración de las opciones de direccionamiento.

### Paso 45:

Aparece un pantallazo Summary que es el resumen de la configuración del dispositivo. Presionar el botón Finalizar para terminar la configuración. Como se muestra en la figura 67.

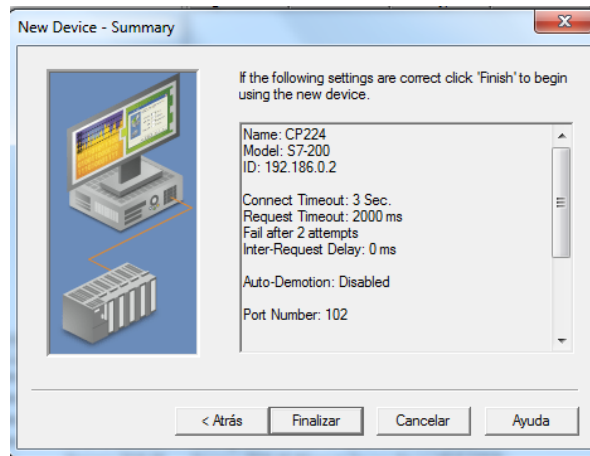


Figura 67. Configuración del dispositivo.

Ya está configurada la comunicación del PLC Siemens S7-200 con el computador mediante OPC; es decir que desde un cliente OPC se podría monitorear las entradas, salidas, y parámetros del sistema del PLC. Sin embargo es conveniente agregar una etiqueta estática (Static Tag) tanto para probar la comunicación como para que sea accedida posteriormente desde el cliente OPC.

### Paso 46:

Ahora se Añade una etiqueta estática haciendo clic en el enlace Click to add a static tag. Como esta en la figura 68.

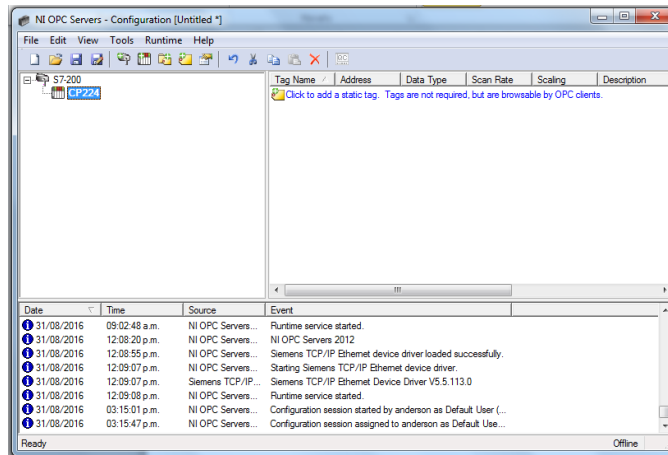


Figura 68. Añadir etiquetas en el NI OPC server

#### Paso 47:

Se Configura las propiedades de la etiqueta como se muestra a en figura 69. Notar que el tipo de dato debe ser Boolean, no Byte (que es el tipo de dato predeterminado). En este caso, cuando se especifica una dirección (por ejemplo Q0.1) en realidad apunta a una dirección de un puerto mayor (en este caso Q1.0). Se desconoce la causa y cómo direccionar al puerto 0. Sin embargo, más adelante cuando se utilice LabVIEW como cliente OPC el direccionamiento se hará correctamente. Presionar Aceptar una vez configurada la etiqueta.

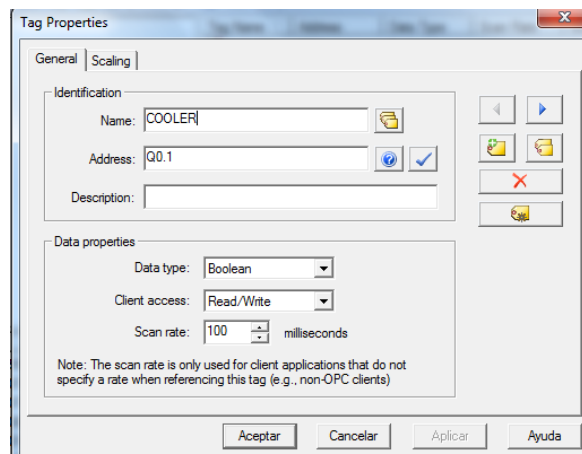


Figura 69. Configuración de una etiqueta

### Paso 48:

Ahora se mira que las etiquetas están añadidas como se muestra en la siguiente figura 70.

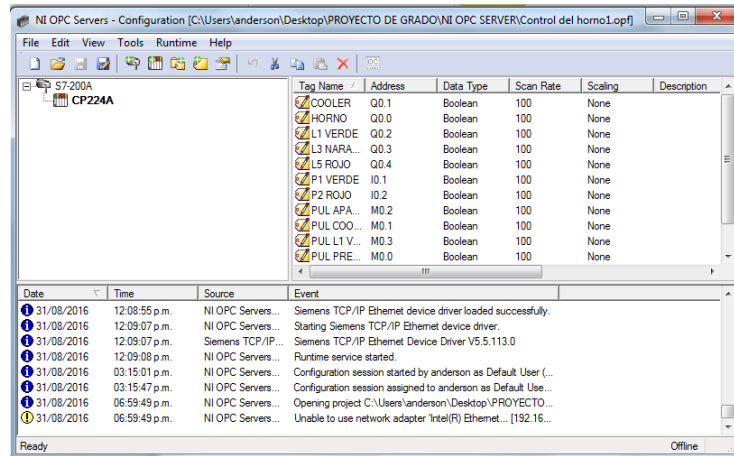



Figura 70. Confirmación de etiquetas en el NI OPC server

### Paso 49:

Para confirmar que el PLC se esté comunicando con el servidor OPC correctamente se puede lanzar el cliente OPC rápido. Presionar el botón Quick OPC Client . En el árbol de la parte superior izquierda de la pantalla seleccionar el canal y el dispositivo que se ha configurado; en este caso, S2-700,CP224. En la parte superior derecha de la pantalla aparecerá la variable con su valor y parámetros de configuración debe salir la palabra Good como se muestra en la figura 71

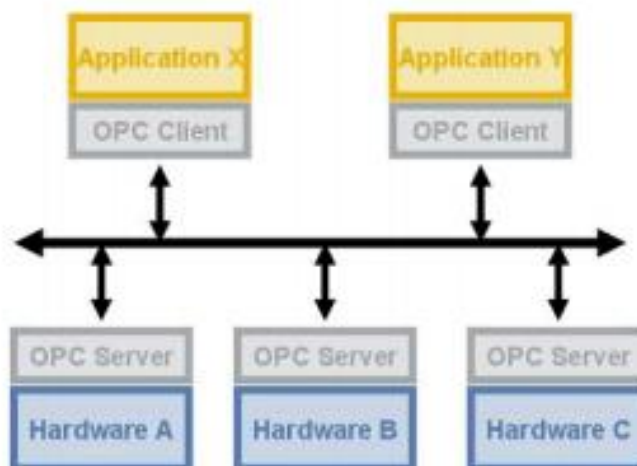
Item ID	Data Type	Value	Timestamp	Quality	LjF
S7-200A.CP224A.COOLER	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.HORNO	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.L1.VERDE	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.L3.NARA	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.L5.RDJO	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.P1.VERDE	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.P2.RDJO	Boolean	1	08.36.32.227	Good	2
S7-200A.CP224A.PUL.APA	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.PUL.COD	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.PUL.L1.V	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.PUL.PRE	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.SENSOR	Boolean	0	08.36.32.227	Good	2
S7-200A.CP224A.VALOR.T	Boolean	29.7813	08.36.33.633	Good	4

Figura 71. Validación del estado de conexión de las variables a PLC

La configuración del servidor OPC ha sido terminada. Se puede cerrar el servidor.

## 6.2 Monitoreo del horno desde labview

Una vez que se ha configurado correctamente el servidor OPC se puede conectar cualquier cliente OPC para monitorear los PLC. Donde se utilizará LabVIEW como cliente OPC.



## Paso 1:

Primero abrimos el software Labview en donde abrimos un nuevo proyecto, como se muestra en la figura 72.

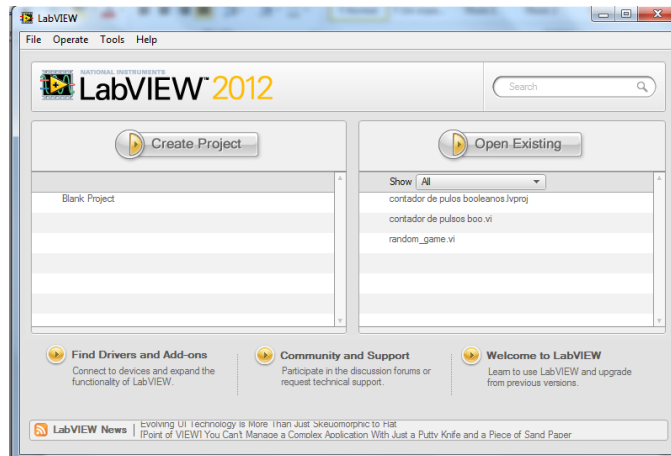


Figura 72. Creación de un nuevo proyecto en labview

## Paso 2:

Ahora nos ubicamos en My Computer y le damos clic derecho para que nos salga un menú y seleccionamos New y luego buscamos I/O Server y le damos clic, como se muestra en la figura 73

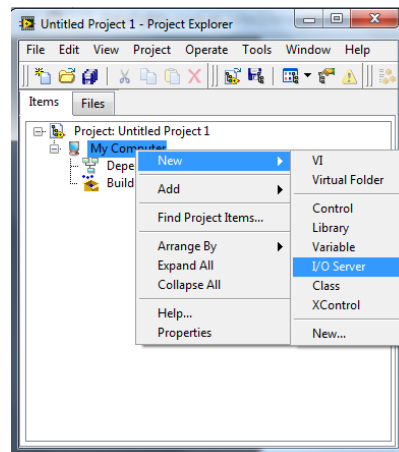


Figura 73. Creación de un I/O Server



### Paso 3:

Se selecciona entre los tipos de I/O Server OPC Client presionamos continue..., como se ilustra en la figura 74.

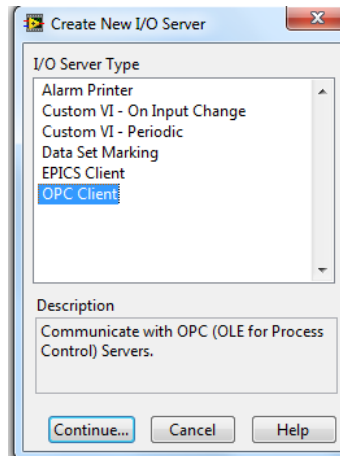


Figura 74. Selección del tipo I/O Server

### Paso 4:

La ventana de configuración de OPC Client, se selecciona *National Instruments.NIOPCServersV5*. luego en Update Rate (ms) es la tasa a la que el servidor se comunicará con el PLC. Configurar a 100 ms en lugar de los 1000 ms predefinidos debido a que el programa del PLC corre relativamente rápido. Presionar el botón OK como se ilustra en ala figura 75.

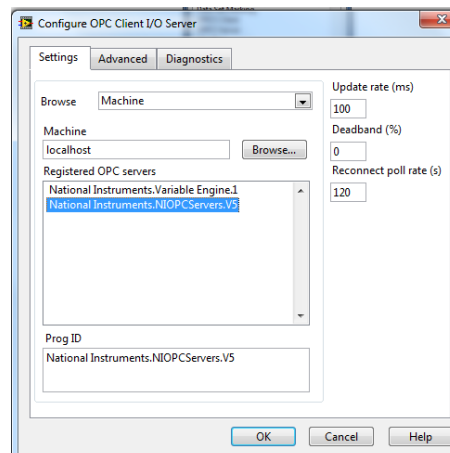


Figura 75. Configuración de OPC Client I/O Server

### Paso 5:

Ahora se despliega el menú dentro del archivo Untitled Library 1 y buscamos el OPC1 y le damos clic derecho y buscamos create bound Variables.. y lo seleccionamos como se muestra en la figura 76.

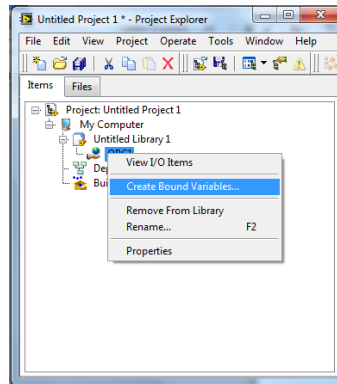


Figura 76. Creación de variables ligadas

### Paso 6:

En la ventana de create bound Variables buscamos las variables en browse Source, se selecciona las que necesitamos y le damos clic en ADD por ultimo presionamos OK, como se ilustra en la figura 77.

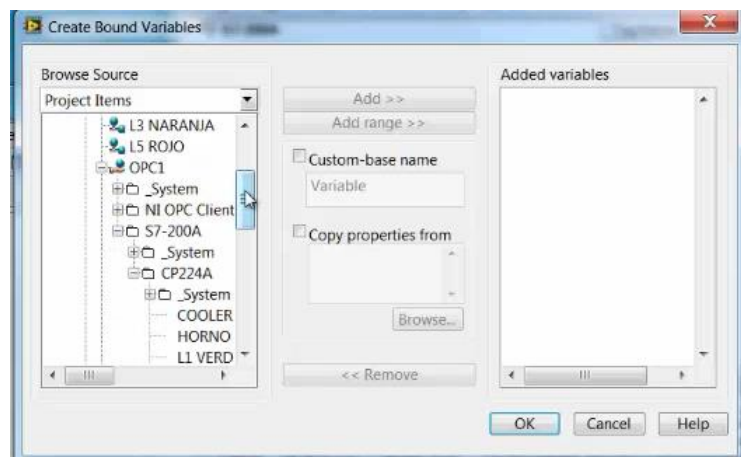


Figura 77. Adición de las variables leídas del PLC

## Paso 7:

Luego en la ventana de Project Explorer nos aparecen las variables que adicionamos anteriormente como se muestra en la figura 78

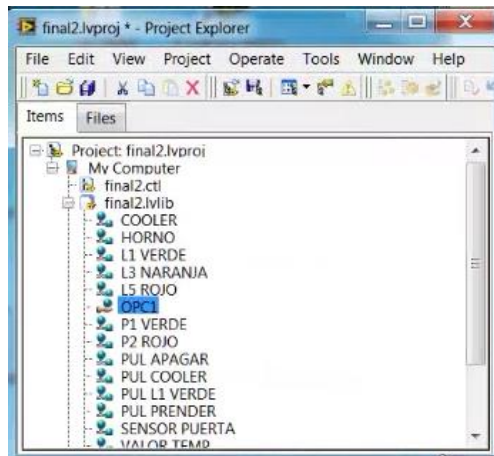


Figura 78. Verificación de la adición de las variables

Después de realizar la adicción de las variables que serán leídas o escritas dependiendo de la necesidad que tenga a controlar, en nuestro caso se hizo el diseño del control por medio de labview del horno que está en el módulo M2CI el cual se manipulo para que se programa su temperatura y la mantuviera durante el lapso de tiempo programado, bajo la creación de un VI dentro de la plataforma del software, como se ilustra en la figura 79.



Figura 79. Programación de VI para control del horno

Ahora se abordara el diseño del diagrama a bloques que controla el VI que fue creación de nosotros y se explicara cómo fue concebido este diagrama, se mostrarla en la figura 80.

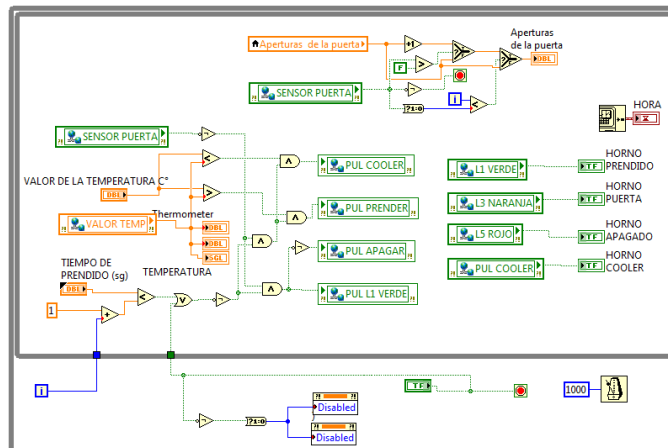


Figura 80. Diagrama a bloques del diseño de VI control del horno

Para hacer control de apagado y encendido del horno se crearon variables globales en las cuales se incluyeron el prendido del horno, prendido del cooler, sensor de la puerta, indicadores de luz verde rojo y naranja y el más importante el valor de la termocupla “temperatura”.

Para el mantener el control de la temperatura programada por el usuario dentro del horno, se realizó por medio de comparadores de mayor que el encargado de mirar si la temperatura es menor que la programada por le usuario mantenga encendió el horno cuando sea mayor que la programada apague el horno, pero tenemos otra condición menor que encargada del prendido del cooler cuando esta menor el valor de temperatura tenga apagado el cooler pero que cuando es mayor lo prenda el cooler; con el objetivo de mantener el valor de temperatura programada por le usuario. Para poder ver este proceso se incorporó al diseño del VI tres indicadores el primero es un indicador grafico de llenado de barra de color rojo, el segundo un indicador numérico y por ultimo un indicador gráfico, esto se muestra en la figura 79.

Por medio de la lógica digital se realizó los accionamientos de los diferentes variables globales ya que la respuesta y su escritura, se hace de forma booleana que es estado Falso o verdadero se construyeron tablas de estados y con su respectiva salida para construir [18] mapa de karnaugh respectivo. Siendo el

tiempo el dominador del proceso y la segunda variable más importante en este proceso.

Para construcción de estos mapas se hicieron con las variables globales que son Las condiciones de la temperatura descrita anteriormente, el sensor de la puerta, y el tiempo programado para activar o desactivar diferentes globales que son pulsador de cooler, pulsador del horno, pulsador de apagar y pulsador de L1 verde siendo estos dos últimos los indicadores luminosos que están el modulo. Quedando así como se muestra en la siguiente tabla 4.

A=Puerta del horno  
 B=Prender horno  
 C= Tiempo programado  
 F(ABC)= pulsador L1 verde

A	B	C	F(ABC)
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

		AB			
		00	01	11	10
C	0	1	1	0	0
	1	0	0	0	0

$F(ABC) = \bar{A} \bar{C}$

Tabla 4. Tabla de estados y mapa de karnaugh pulsador L1 verde.

Donde el pulsador de apagar que corresponde al indicar luminoso rojo quedo gobernado por la salida del pulsador L1 verde pero su salida se le introdujo una compuerta NOT para invertir su estado, así prenda el indicador rojo cuando el indicador verde este en estado 0. Dando a entender que el horno está apagado.

A Continuación con el del control del prendido del horno como se ilustra en la siguiente tabla 5

A=Puerta del horno

B=Tiempo programado  
 C= Comparador de Temperatura mayor que  
 F(ABC)= Prender horno

A	B	C	F(ABC)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

		AB			
		00	01	11	10
C	0	0	0	0	0
	1	1	0	0	0

$F(ABC) = \bar{A} \bar{B} C$

Tabla 5. Tabla de estados y mapa de karnaugh pulsador horno.

Ahora seguimos con el del control del prendido del cooler como se ilustra en la siguiente tabla 6

A=Puerta del horno  
 B=Tiempo programado  
 C= Comparador de Temperatura menor que  
 F(ABC)= Prender Cooler

A	B	C	F(ABC)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

		AB			
		00	01	11	10
C	0	0	0	0	0
	1	1	0	0	0

$F(ABC) = \bar{A} \bar{B} C$

Tabla 6. Tabla de estados y mapa de karnaugh pulsador cooler.

Para los indicadores luminosos se prendieran y se apagaran esto se tomó de las variables de control que generamos anteriormente relacionándolas con los colores el rojo para pagado, el verde para para encendió, el naranja para la puerta y el azul para el cooler. Si se visualiza el encendido en el VI de estos bombillos esto se ve reflejado en el módulo M2CI en accionamiento de ellos.

Se incorporó un contador de eventos para la apertura de la puerta durante el proceso, el cual se desarrolló por medio de condiciones lógicas de verdadero y falso, sumador y una variable local para llevar el conteo como se muestra en la parte superior derecha de la figura 80.

### 6.3 Análisis del desarrollo del proyecto

Para que el modulo funcione todo y correctamente lo primero que se hizo fue realizar el cableado adecuado de los dispositivos a trabajar y se les aplico sus respectivos voltaje y corrientes de alimentación, en lo que se recurrió a al manual del fabricante del módulo y también al datasheet de cada dispositivo electrónico que se usó en la práctica, ya teniendo eso se procedió a la instalación de los software microwin step 7 con el que programa el PLC que se posee.

Al ver que la versión del software del microwin step 7 y el modelo del plc se decidió instalar una máquina virtual que nos proporciona una plataforma de Windows XP, se realizó la correcta instalación del microwin, el cual funciono perfectamente pero surgió el problema de la conexión entre el PC y PLC por conflicto por poseer una tarjeta de red virtual y no poder realizar tener una sincronización con la tarjeta física. Para poder tener una correcta comunicación entre el software y la tarjeta de red fue necesario realizar la instalación del software del microwin step 7 a la plataforma de Windows 7 donde el fabricante por medio de su página de soporte da todas las herramientas necesarias para realizar este tipo de operación. Ya se pudo realizar el enlace entre el PC y el PLC por medio del uso de la tarjeta de red física que posee el computador.

Existe un cable que es el indicado o adecuado para poder programar el PLC que es un convertidor USB a RS485 el cual solo se pudo utilizar por puerto USB y no como un puerto COM lo que no deja poder utilizar para recoger las variables del PLC desde OPC server o un software remoto, para esto se construyó un cable es cual es un convertidor USB a RS485 pero con la peculiaridad que crea un puerto virtual COM y esto nos permite ya configurarlo que pueda recoger variables en software para controles remotos en nuestro caso NI OPC server .

Para poder conectar todo los dispositivos al PLC y sus módulos adiciones se realizó una serie de conexiones y que se pudieran conectar sin generar ningún tipo de corto o de fallo en el pero a su veces cumpliera con su función un ejemplo la conexión de la termocupla y el módulo EM231 a uno de sus entradas análogas, que fuese capaz de recoger el dato y que correspondiera con lo captado, dentro los parámetros que hay que se requiera y fuera de acorde con la realidad, aparte de esto se realizó una serie de configuraciones tanto el hardware y el software de los dispositivos con la ayuda de los datasheet de cada uno de los componentes involucrados en proceso de control del horno.





## CONCLUSIONES

- Se pudo Identificar la estructura y funcionamiento del módulo M2CI con la mayoría de sus componentes y todo más con los que le estaban involucrados la parte del control del horno.
- Se Planteó un sistema de conmutación por medio del módulo de Ethernet que nos permito recoger la variables de forma remota y por medio del dadas salidas y entradas digitales del PLC poder encender y controlar la planta de temperatura que integra el módulo M2CI.
- Diseño los programas o rutinas necesarias por medio de los software que gobiernan la programación y operación del PLC y demás accesorios, y su respectiva configuración física para que todo quedara funcionando e integrando el control de la planta piloto de temperatura.
- Gracias al software de Labview se Incorporó en el diseño gráfico que es amigable con el usuario y fácil de entender del funcionamiento del módulo y su programación logica se realizó con los diagramas de bloques que trae el.
- Los datasheet de cada dispositivo electrónico nos ayudó para poder saber que pines se conectaban, de qué forma y que voltajes son los permitidos para su correcto funcionamiento.

## RECOMENDACIONES

- Se recomienda para una mejor comprensión y entendimiento de los dispositivos que están involucrados en el módulo M2CI cuente con una mejor documentación de cada uno de los ellos y más aterrizadas para que el estudiante pueda contar con las cosas herramientas adecuadas.
- Se recomienda que para el correcto funcionamiento y una mejor manipulación del módulo cuente con personas que tengan un mayor conocimiento en los temas tanto de software como del hardware que están involucrado en torno al él.
- Se recomienda para sacar el máximo rendimiento del laboratorio remoto industrial cuente con conexiones no de cables si no digitales para tratar de evitar un error de conexión entre los dispositivos.
- Se recomienda para lograr el acceso remoto a este módulo y funcionamiento cuente con su etapa de comunicación de acceso a internet por medio de la instalación de un servidor.

## BIBLIOGRAFÍA

- [1] I. Santana, M. Ferre, E. Izaguirre, R. Aracil, and L. Hernández (2013) Remote Laboratories for Education and Research Purposes in Automatic Control Systems, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, VOL. 9, NO. 1, FEBRUARY 2013
- [2] HODSON, D. (1994) Hacia un enfoque más crítico del trabajo de laboratorio. ENSEÑANZA DE LAS CIENCIAS, 1994, 12.(3), 299-313
- [3] <http://weblab.deusto.es/website/>
- [4] <http://es.rs-online.com/images/F4886814-01.jpg>
- [5] <http://www.distribuidor-oficial-siemens-productos-electricos.control-technics.com.ar/productoselectricos/es/Notas%20y%20articulos/Automatizacion/1677/SIMATIC%20S7-200.htm>
- [6] <http://electricidadserrano.com.ar/marcas/siemens/plc-simatic-s7-200-siemens/>
- [7] <http://srcsl.com/que-es-un-sensor-pt100/>
- [8] [https://es.wikipedia.org/wiki/Sensor\\_inductivo#Conceptos\\_te.C3.B3ricos](https://es.wikipedia.org/wiki/Sensor_inductivo#Conceptos_te.C3.B3ricos)
- [9] [https://es.wikipedia.org/wiki/Rel%27\\_de\\_estado\\_s%27lido](https://es.wikipedia.org/wiki/Rel%27_de_estado_s%27lido)
- [10] <http://www.findernet.com/en/node/47493>
- [11] <https://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/Documents/S7200-1HORA.PDF>
- [12] <https://support.industry.siemens.com/cs/pd/368630?ptdi=td&pnid=15336&lc=es-WW>
- [13] <https://support.industry.siemens.com/cs/document/26340488/nuevos-m%C3%B3dulos-anal%C3%B3gicos-em231-y-em232-para-el-simatic-s7-200?dti=0&lc=es-WW>

[14] <http://repositorio.espe.edu.ec/bitstream/21000/5236/1/T-ESPEL-0914.pdf>

[15] <https://estudios.unad.edu.co/componente-practico-ebti/escenarios-remotos-ecbti>

[16] Londoño Salazar, Jesús Enrique, Alvarez Córdoba, Modelo para la implementación de laboratorios en programas bajo modalidad virtual – caso aplicado a la Ingeniería Informática.

[17] <http://www.ni.com/getting-started/labview-basics/esa/environment>

[18] [http://www.ee.calpoly.edu/media/uploads/resources/KarnaughExplorer\\_1.html](http://www.ee.calpoly.edu/media/uploads/resources/KarnaughExplorer_1.html)