

LABORATORIO DE PLC

FRANCISCO FERNÁNDEZ PIÑA

ELIAS PIMIENTA TATIS

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA ELÉCTRICA

ÁREA DE AUTOMATIZACIÓN

CARTAGENA, D.T Y C.

2001.

LABORATORIO DE PLC

FRANCISCO FERNÁNDEZ PIÑA

ELIAS PIMIENTA

Trabajo de Grado presentado
como requisito para optar al título de
Ingeniero Electricista.

Director
Jorge Duque
Ing. Electricista.

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERIA ELÉCTRICA

ÁREA DE AUTOMATIZACIÓN

CARTAGENA, D.T. Y C.

2001.

Nota de aceptación

Presidente del Jurado

Jurado

Jurado

Cartagena, 16 de Abril de 2001.

A Camilo Andrés, mi hijo quien me motivó con su ternura, a mis padres Eva y Pacho por el apoyo que me brindaron.

AGRADECIMIENTOS

Los autores expresan su agradecimiento a:

Jorge Duque, Ingeniero Electricista y Director del Proyecto

Ruben Morales, Ingeniero Electrónico y Programador de PLC, por su valiosa ayuda en la implementación de los programas de Control.

CONTENIDO

	pág.
INTRODUCCIÓN	!
1. LABORATORIO DE PLC	1
1.1 GENERALIDADES	1
2. DESCRIPCIÓN DEL BANCO	5
2.1 MÓDULO DE LLENADO Y VACIADO DE TANQUES	6
2.2 ARRANCADOR ESTRELLA – TRIÁNGULO	10
2.3 MÓDULO BANDA TRANSPORTADORA	13
2.4 CONTROL EN CRUCE DE VÍAS	13
3. CARACTERÍSTICAS DEL PLC SIMATIC S7 – 200 CPU 214 DE SIEMENS.	15
3.1 CARACTERÍSTICAS MECÁNICAS.	16
3.2 CARACTERÍSTICAS TÉCNICAS.	16
3.3 CARACTERÍSTICAS ADICIONALES DE LA CPU 214	17
3.3.1 Funciones de la CPU 214	17
3.3.2 Comunicaciones	18
4. INTRODUCCIÓN A LA PROGRAMACIÓN KOP Y AWL	24
4.1 OPERACIONES LÓGICAS CON CONTACTOS	26
4.1.1 Descripción	26
4.1.2 Representación	26
4.2 MANEJO DEL PROGRAMA STEP 7 MICRO/WIN	29

4.2.1 Configuración del PLC	31
4.2.2 Crear Proyectos	35
4.2.3 Compilar el Programa	39
4.2.4 Documentar el Programa	39
4.2.5 Cargar un Proyecto en la CPU	41
4.2.6 CPU en modo RUN	44
4.3 EJEMPLOS	46
4.3.1 Encendido de una Bombilla	46
4.3.2 Llenado y Vaciado de un Tanque	48
4.3.3 Manejo de Interrupciones Temporizadas	57
4.3.4 Programación de un Secuenciador	64
4.3.5 Control de Tiempo para la Iluminación de una Escalera	71
5. PRACTICAS	73
5.1 SISTEMA DE LLENADO Y VACIADO DE TANQUES	73
5.2 ARRANQUE ESTRELLA – TRIÁNGULO E INVERSIÓN DEL SENTIDO DE GIRO DE UN MOTOR TRIFÁSICO DE INDUCCIÓN	75
5.3 SISTEMA DE DOS BANDAS TRANSPORTADORAS B1 Y B2	79
5.4 SEMÁFORO PARA VEHICULOS Y PEATONES EN EL CRUCE DE DOS CALLES	80
6. CONCLUSIONES	82
BIBLIOGRAFÍA	83
ANEXOS	

LISTA DE CUADROS

	Pág.
Cuadro 1. Datos Técnicos PLC Simatic7 – 200, CPU 214	20
Cuadro 2 Cuadro comparativo CPU 214, Salidas D.C, Relé y A.C	21
Cuadro 3. Operaciones lógicas con contactos en los lenguajes KOP y AWL.	22
Cuadro 4. Ejemplo de operaciones lógicas con contactos KOP y AWL	28

LISTA DE FIGURAS

	pág.
Figura 1. Banco de pruebas PLC	5
Figura 2. Esquema del sistema de llenado y vaciado de tanques	6
Figura 3. Sistema de seguridad en los tanques	7
Figura 4. Switches de nivel	7
Figura 5. Bornera para el sistema de control de nivel	9
Figura 6. Circuito arrancador estrella – triángulo (Y – D)	10
Figura 7. Bornera de contactores	12
Figura 8. Bornera de control para arrancador (Start – Stop), líneas de alimentación, pulsadores y bombillas de indicación.	12
Figura 9. Esquema banda transportadora	13
Figura 10. Semáforo en cruce de vías	14
Figura 11. Bornera para la conexión del control en el cruce de vías	14
Figura 12. Identificación de los terminales de conexión para la CPU 214 AC/DC/Relay	22
Figura 13. Bornera de conexión del PLC ubicada en el banco de pruebas	23
Figura 14. Estructura del programa	25
Figura 15. Ventana inicial del Step7 – Micro/Win	30
Figura 16. Ventana para configurar las preferencias del PLC	31
Figura 17. Ventana que muestra la configuración del puerto y dirección de la CPU	33

Figura 18. Ventana para iniciar la programación	35
Figura 19. Ventana para cambiar al formato AWL	36
Figura 20. Ventana para la selección de contactos	37
Figura 21. Ventana para la asignación bobina de salida	38
Figura 22. Ventana para elegir END y finalizar un programa	38
Figura 23. Cuadro de diálogo del menú CPU	39
Figura 24. Ventana Asignación de símbolos	40
Figura 25. Ventana programa documentado	41
Figura 26. Ventana que nos permite colocar la CPU en STOP	42
Figura 27. Ventana cargar en CPU	43
Figura 28. Ventana Conmutar CPU en modo RUN	45
Figura 29. Ventana Programa terminado en formato AWL	45
Figura 30. Esquema de conexión al PLC del encendido de una Bombilla	47
Figura 31. Dibujo ilustrativo del proceso de llenado y vaciado de un Tanque	49
Figura 32. Esquema de conexión al PLC, del sistema de llenado y vaciado de TK1	49
Figura 33. Esquema de conexión interrupción temporizada	57
Figura 34. Esquema de conexión secuencia de pasos	64
Figura 35. Esquema de conexión control luces en escalera	71
Figura 36. Esquema eléctrico de potencia del arrancador estrella – triángulo	76
Figura 37. Esquema eléctrico de potencia del inversor del sentido de giro de un motor trifásico	78
Figura 38. Esquema de bandas transportadoras	79

Figura 39. Señalización de vías y disposición del conmutador para el control del semáforo

81

LISTA DE ANEXOS

Anexo A. Solución a los problemas planteados

Anexo B. Manual de usuario Step 7 – Micro/Win

Anexo C. Manual de referencia SIMATIC Software

GLOSARIO

AC: (Alternating Current) es la abreviatura de Corriente Alterna.

AUTÓMATA PROGRAMABLE (PLC): Un autómata programable es un controlador de lógica programable concebido inicialmente para sustituir los sistemas de control de relés fijamente cableados. Actualmente, los autómatas programables incorporan una gran variedad de funciones de control

BIT: Un bit es un dígito binario que puede tener uno de los valores siguientes: 0 ó 1 (activado/desactivado; verdadero/falso)

BOBINA: Una bobina es un símbolo del lenguaje de programación Esquema de Contactos () que representa una bobina de relé que se excita al pasar la corriente.

BYTE: Un byte se compone de ocho bit.

CABLE PC/PPI: El cable PC/PPI permite conectar el puerto RS-485 del S7-200 al puerto RS-232 de una PC estándar.

CONTACTO: Es un símbolo del lenguaje de programación Esquema de Contactos (KOP). Representa un contacto por el cual pasa corriente cuando se cierra. Los contactos pueden ser normalmente cerrados o normalmente abiertos.

CONTADOR: Un contador es un elemento que cuenta los flancos positivos en las entradas de conteo. La CPU 214 dispone de 128 contadores.

DC: DC(Direct Current) es la abreviatura de corriente continua.

ENTRADA: Una entrada es una información que lee el autómata programable y que utiliza como condición para el control o estado.

ENTRADAS Y SALIDAS DIGITALES: Las entradas y salidas digitales pueden tener uno de los estados de señal siguientes. 0 ó 1 (activado / desactivado; verdadero / falso).

LISTA DE INSTRUCCIONES (AWL): Es un lenguaje de programación textual (a diferencia del lenguaje KOP que imita un esquema de circuitos) AWL se utiliza para programar el autómata programable S7-200. Cada línea del programa AWL contiene una instrucción determinada que contiene uno o más operandos según la operación.

MARCA INTERNA: La marca interna también denominada relé de control, ofrece espacio de memoria para informaciones de estado y control temporales.

PALABRA: una palabra consta de 16 bits.

SEGMENTO: Un segmento consta de varias operaciones KOP que forman juntas una línea.

SELECTOR DE MODO: El selector de modo tiene tres posiciones y sirve para elegir el modo de operación del autómata programable.

SUBROUTINA: Una subrutina es una parte del programa que ha de ser llamada para su ejecución. En el autómata programable S7-200 pueden utilizarse subrutinas, aunque no es necesario. Las subrutinas se agregan al final del programa principal.

TEMPORIZADOR: Es un elemento que cuenta incrementos de tiempo. En el S7-200 los temporizadores tienen incrementos de 1, 10 ó 100 milisegundos. La CPU 214 ofrece un total de 128 temporizadores.

TIERRA: Se denomina tierra a la masa conductiva cuyo potencial eléctrico se puede desactivar (poner a 0) en cualquier punto.

RESUMEN

El objetivo principal del proyecto es implementar unos módulos para el desarrollo de practicas de laboratorio en el área de control. Cada módulo trabaja de manera independiente, permitiendo así la coordinación de las practicas.

El corazón de los practicas lo constituye el PLC Simatic S7 – 200 CPU 214 de SIEMENS, el cual gracias a sus características técnicas de desempeño en la gama media de PLC's, con entradas y salidas digitales, es una herramienta poderosa para la implementación de las practica en cada módulo, la flexibilidad del PLC también brinda la opción de implementar practica con sistemas que trabajen con variables análogas.

Los sistemas implementados en cada módulo poseen elementos de control como los utilizados en algunos procesos industriales, tales como los interruptores de nivel empleados en las plantas de procesamiento de bebidas, sensores inductivos empleados común mente en sistemas de empaquetamiento, contactores empleados en arranque y control de motores.

Compilando todo lo que ofrece el proyecto, la CUTB esta adquiriendo una herramienta más para el desarrollo de practicas en el laboratorio de control.

INTRODUCCIÓN

Teniendo presente que en la Corporación Universitaria Tecnológica de Bolívar se viene trabajando por un completo laboratorio en las áreas de Automatización y Control, el presente documento viene a formar parte del grupo de proyectos que se han encaminado a estas áreas.

El Laboratorio de PLC que se plantea en este proyecto nos permite implementar prácticas para el control de Nivel, Arranque de Motores de corriente alterna, sistema de bandas transportadoras, y controlar un semáforo en el cruce de dos vías, utilizando el PLC SIEMENS Simatic S7 – 200 CPU 214.

Estas prácticas están agrupadas en cuatro (4) módulos, los cuales permiten al estudiante coordinar mejor la ejecución del laboratorio. Cada módulo se implementa de manera independiente, interconectando las borneras respectivas en el banco de pruebas. El proyecto también pretende iniciar al estudiante en la programación de PLC's en los formatos Lista de Instrucciones (AWL) y Esquema de Contactos (KOP) de esta manera el estudiante adquiere una valiosa herramienta para implementar controles a escala industrial.

1. LABORATORIO DE PLC

1.1 GENERALIDADES

Los PLC's se introdujeron por primera vez en la industria en 1960 aproximadamente. La razón principal de tal hecho fué la necesidad de eliminar el gran costo que se producía al reemplazar el complejo sistema de control basado en relés y contactores. Bedford Associates propuso algo denominado Controlador Digital Modular (MODICON, MODular DIGital CONtroller) a un gran fabricante de coches. Otras compañías propusieron a la vez esquemas basados en ordenador, uno de los cuales estaba basado en el PDP-8. El controlador digital MODICON 084 resultó ser el primer PLC del mundo en ser producido comercialmente.

El problema de los relés era que cuando los requerimientos de producción cambiaban también lo hacía el sistema de control. Esto comenzó a resultar bastante caro cuando los cambios fueron frecuentes. Dado que los relés son dispositivos mecánicos y poseen una vida limitada se requería una estricta manutención planificada. Por otra parte, a veces se debían realizar conexiones entre cientos o miles de relés, lo que implicaba un enorme esfuerzo de diseño y mantenimiento.

Los "nuevos controladores" debían ser fácilmente programables por ingenieros de planta o personal de mantenimiento. El tiempo de vida debía ser largo y los cambios

en el programa tenían que realizarse de forma sencilla. Finalmente se imponía que trabajaran sin problemas en entornos industriales adversos. La solución fué el empleo de una técnica de programación familiar y reemplazar los relés mecánicos por relés de estado sólido.

A mediados de los 70 las tecnologías dominantes de los PLC eran máquinas de estado secuenciales y CPU basadas en desplazamiento de bit. Los AMD 2901 y 2903 fueron muy populares en el Modicon y PLC's A-B. Los microprocesadores convencionales cedieron la potencia necesaria para resolver de forma rápida y completa la lógica de los pequeños PLC's. Por cada modelo de microprocesador había un modelo de PLC basado en el mismo. No obstante, el 2903 fué de los más utilizados.

Las habilidades de comunicación comenzaron a aparecer en 1973 aproximadamente. El primer sistema fué el bus Modicon (Modbus). El PLC podía ahora dialogar con otros PLC's y en conjunto podían estar aislados de las máquinas que controlaban. También podían enviar y recibir señales de tensión variables, entrando en el mundo analógico. Desafortunadamente, la falta de un estándar acompañado con un continuo cambio tecnológico ha hecho que la comunicación de PLC's sea un maremagnum de sistemas físicos y protocolos incompatibles entre si. No obstante fué una gran década para los PLC's.

En los 80 se produjo un intento de estandarización de las comunicaciones con el protocolo MAP (Manufacturing Automation Protocol) de General Motor's. También fué un tiempo en el que se redujeron las dimensiones del PLC y se pasó a programar

con programación simbólica a través de ordenadores personales en vez de los clásicos terminales de programación. Hoy día el PLC más pequeño es del tamaño de un simple relé.

Los 90 han mostrado una gradual reducción en el número de nuevos protocolos, y en la modernización de las capas físicas de los protocolos más populares que sobrevivieron a los 80. El último estándar (IEC 1131-3) intenta unificar el sistema de programación de todos los PLC en un único estándar internacional. Ahora se dispone de PLC's que pueden ser programados en diagramas de bloques, lista de instrucciones, C y texto estructurado al mismo tiempo.

Los PC están comenzando a reemplazar al PLC en algunas aplicaciones, incluso la compañía que introdujo el Modicon 084 ha cambiado al control basado en PC. Por lo cual, no sería de extrañar que en un futuro no muy lejano el PLC desaparezca frente al cada vez más potente PC, debido a las posibilidades que éste último puede proporcionar.

En casa y en la oficina, el ordenador personal continua su progreso triunfal. El PC se ha establecido en un gran número de campos. Análogamente al lema "Más rápido, más alto, más lejos", los componentes hardware y software estan siendo cada vez más potentes y más rentables. Es lógico, por tanto, que la industria quiera tomar provecho de este hecho, para reducir costes y/o incrementar la productividad.

Ciertas tareas industriales están actualmente en manos de los ordenadores desde hace tiempo: desde emplear la tecnología Windows cuando se manejan pedidos y/o se ajustan parámetros de maquinaria hasta preparar o visualizar datos prácticamente de cualquier tipo.

No hay que sorprenderse entonces, que los especialistas en automatización y los usuarios estén pensando ahora en que forma se pueden transferir al PC otras tareas, para poder llegar a un mayor ahorro. Más recientemente un gran número de simuladores de PLC por software han aparecido en el mercado, que están ayudando a transferir el control de tareas al disco duro y presentan una automatización más efectiva en costes en una simple pieza de hardware (el PC).

Los controladores lógicos programables, en la mayoría de los casos, están diseñados específicamente para ser empleados en ambientes industriales exigentes y han sido continuamente desarrollados de forma que sus sistemas operativos en tiempo real representan su mayor virtud. Ellos son y seguirán siendo, no obstante, la primera elección para todo control de tareas críticas o extremas por su rendimiento y simpleza, en los que un PC podría estar simplemente "sobrecargado".

2. DESCRIPCIÓN DEL BANCO

El banco se construyó con ángulos de 1" x 1" x ¼", pintado de azul policromado, la mesa es de madera forrada con formica tipo PINO.

Los cuatro módulos que conforman el banco están distribuidos según figura 1.

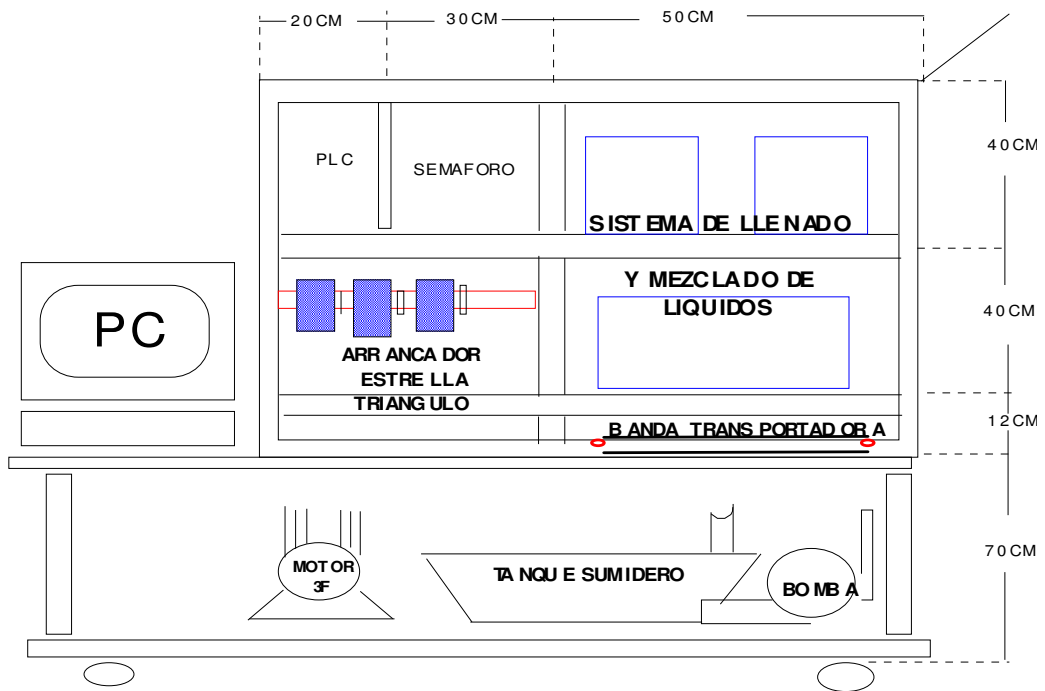


Figura 1. Banco de Pruebas PLC.

Los módulos se detallan a continuación.

2.1 MÓDULO DE LLENADO Y VACIADO DE TANQUES

Éste módulo consta de cuatro (4) recipientes con los cuales se implementará un sistema de llenado y vaciado según se muestra en la figura 2.

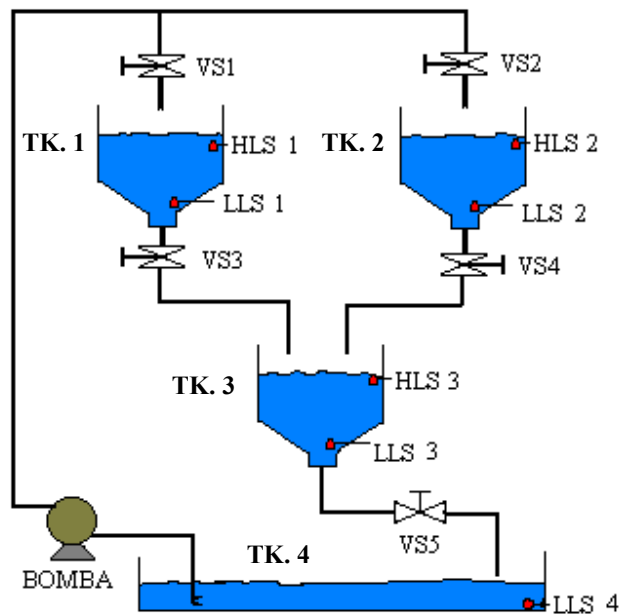


Figura 2. Esquema del sistema de llenado y vaciado de tanques

Los recipientes están hechos con láminas de acrílico de 5mm y sus esquinas se reforzaron con ángulos de aluminio. Cada tanque está provisto con switches de alto y bajo nivel. Además de un sistema de seguridad, el cual reciclará el exceso de agua si llegara a fallar uno de los switches de alto nivel. Ver Figura 3. Los switches permitirán implementar controles ON/OFF. Los Switches HLS son normalmente cerrados (NC) y los LLS son normalmente abiertos (NO). La Figura 4. muestra los modelos de Switches utilizados.

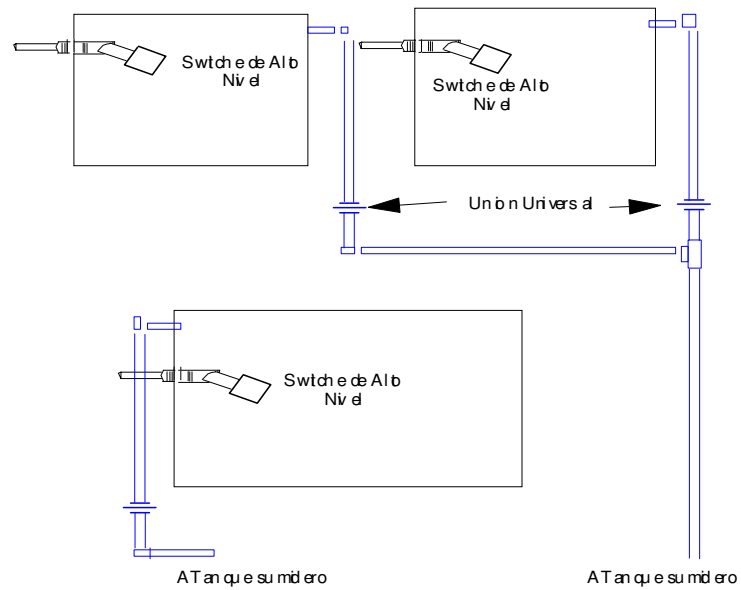


Figura 3. Sistema de Seguridad de los Tanques

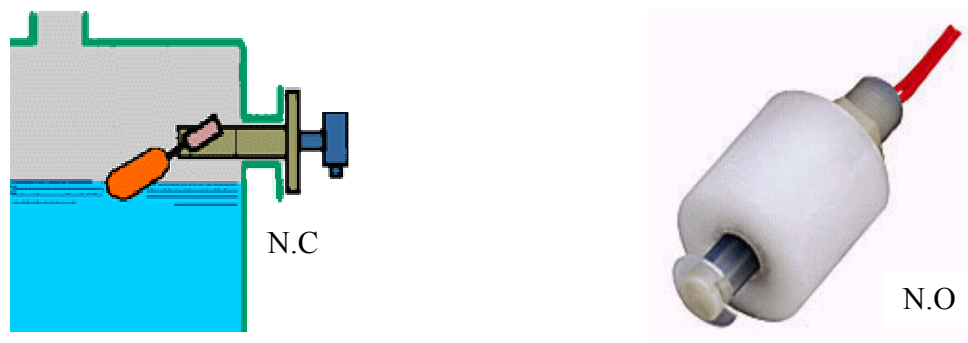


Figura 4. Switches de Nivel

Los tanques TK1 y TK2 en el nivel superior del sistema, reciben el liquido (H_2O) proveniente del tanque TK4 (Sumidero); la entrada y salida de TK1 y TK2 esta controlada por Válvulas Solenoides (ASCO RED - HAT, 120V – 60 Hz) Normalmente Cerradas (NC).

El agua almacenada en TK1 y TK2 es enviada por gravedad al TK3 cuando se abren las válvulas VS3 y VS4 que corresponden a los TK1 y TK2 respectivamente. El llenado de los tanques TK1 y TK2 se realiza por medio de una Bomba sumergible (BB1) colocada en el fondo del tanque TK4 (Sumidero). Un switch de bajo nivel actúa como protección para BB1, evitando que ésta entre en funcionamiento si TK4 se encuentra sin agua.

Las características de BB1 son las siguientes:

GEOR. MAGNETIC SEALLESS PUMPS

Model: 32102 110 VAC. 2000 GPH

La tubería utilizada es CPVC ½", las señales de los switches, válvulas y la alimentación del sistema se ubican en una bornera para su conexión al PLC. Como lo muestra la figura 5.

En la bornera se debe tener presente que el punto común es independiente para Switches de Nivel, como para Válvulas Solenoides. La BOMBA1 está formada por un Switch de bajo nivel en serie con la bomba, el cual actúa como protección para evitar que la bomba trabaje en vacío, también se tienen las salidas de 24, 12 y 5 VDC y unos bornes de reserva (RES) para cualquier conexión futura.

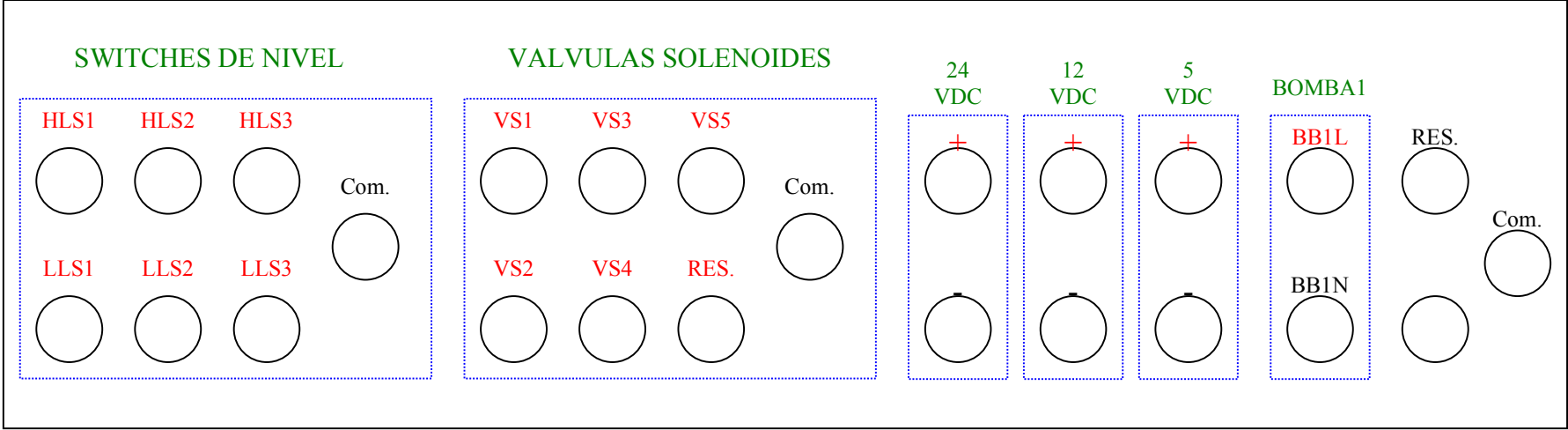


FIGURA 5. Bornera para el Sistema de Control de Nivel

2.2 ARRANQUE ESTRELLA - TRIÁNGULO

En éste módulo se implementa un Arrancador Estrella Triángulo, para poner en funcionamiento un motor trifásico de 3 HP. , también contempla la opción de invertir el sentido de giro del motor. El montaje de los contactores para manejar la parte de potencia en el sistema, se realizará sobre rieles omega, los contactores empleados serán *Telemecanique LC1 D09 10* y *LC1 D12 10 con bobinas a 110 V - 60 Hz.* Todos los puntos de conexión de los Contactores estan ubicados en la Bornera de Contactores. La Figura No. 6 muestra el esquema de potencia del arrancador Y – D.

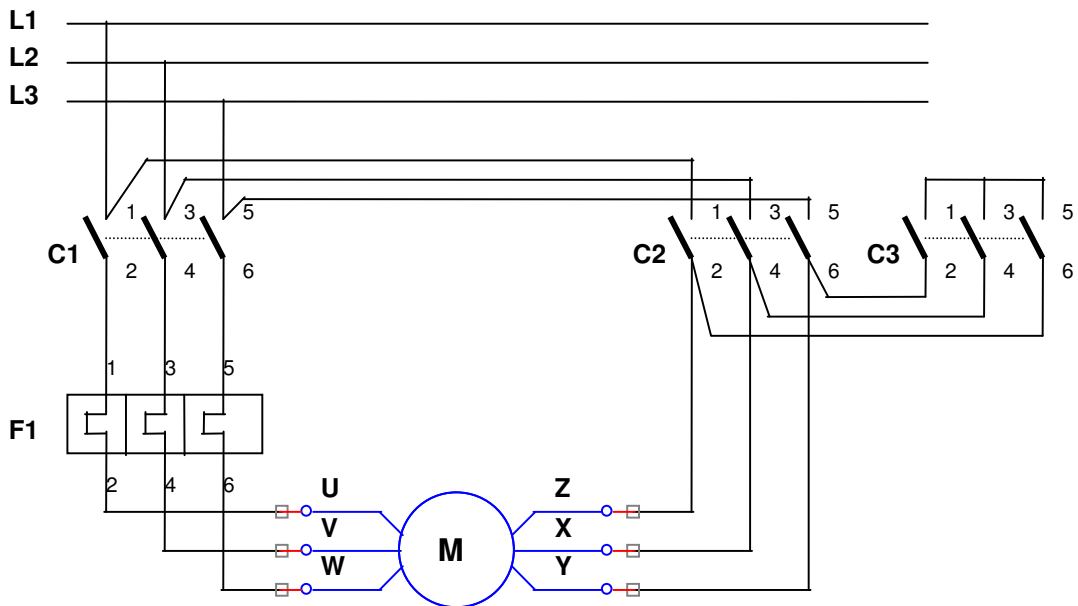


Figura 6. Circuito Arrancador Estrella – Triángulo (Y – D)

Todos los puntos de los contactores se llevan a la bornera de contactores ver figura 7. Mediante ésta bornera se implementa el arrancador estrella triángulo y el inversor de giro para motores de corriente alterna

El sistema de arranque START(S1, pulsador normalmente abierto) – STOP(S2, pulsador normalmente cerrado), las fases L1, L2, L3 y las protecciones están ubicadas en la bornera de control que muestra la figura 8. Esta bornera también está formada por un grupo de interruptores on – off, pulsadores normalmente abiertos, bombillas incandescentes 110 VAC y un grupo de bornas para conexión.

En la bornera de control el conmutador está alimentado a 24 VDC., de tal forma que entre las posiciones 1, 2, 3 y LN están presentes 24 VDC. Este conmutador es utilizado en el desarrollo del módulo control de semáforos en un cruce de vías.

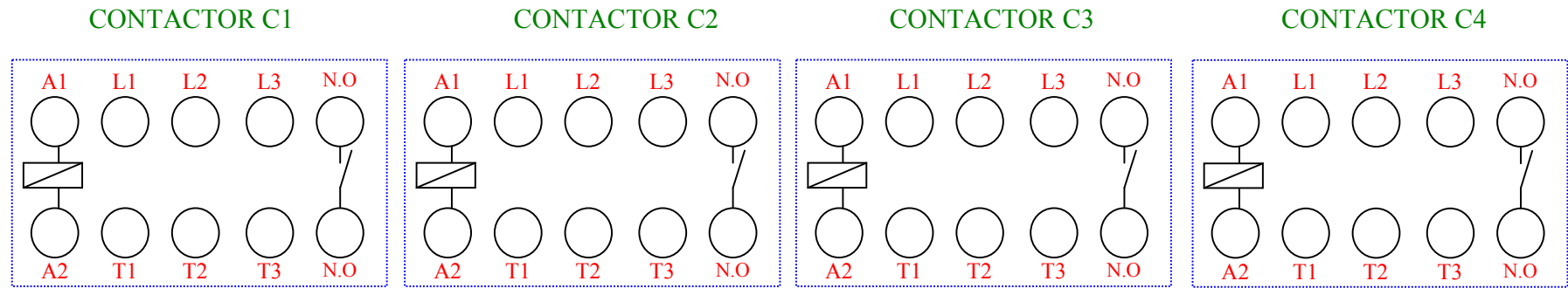


Figura 7. Bornera de Contactores

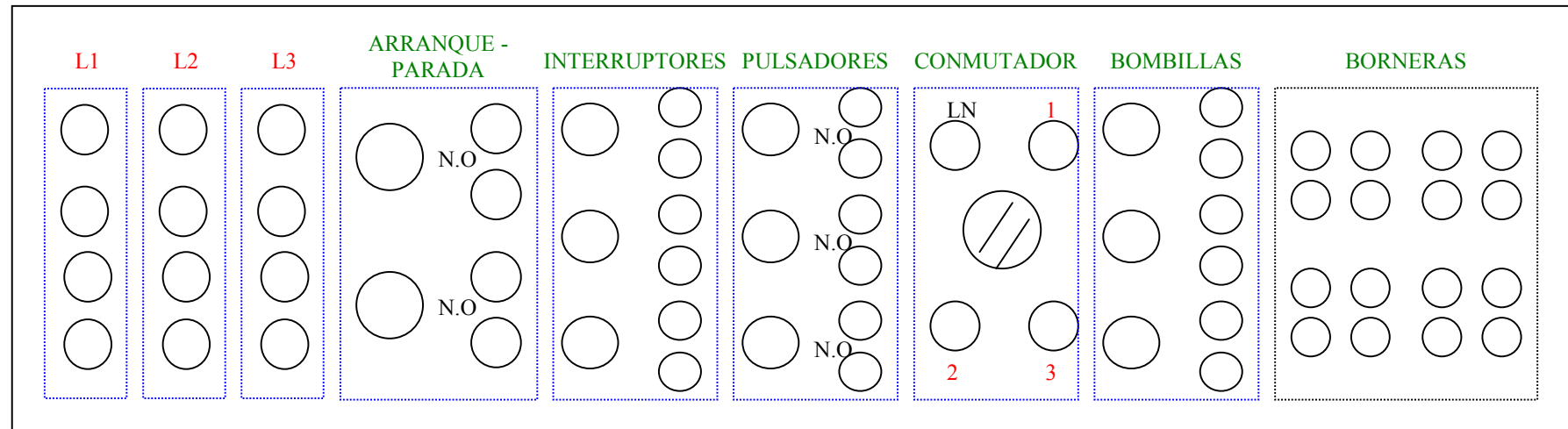


Figura 8. Bornera de Control

2.3 MÓDULO BANDA TRANSPORTADORA

Las dos bandas transportadoras están accionadas por motores de 12 VDC con reductor de velocidad, tienen una longitud de 35 y 50 cm respectivamente, en sus extremos están colocados fotodetectores, para determinar el número de objetos que pasan por la banda, cada banda transportadora es independiente la una de la otra, y sus motores pueden hacerlas mover de izquierda a derecha o de derecha a izquierda.

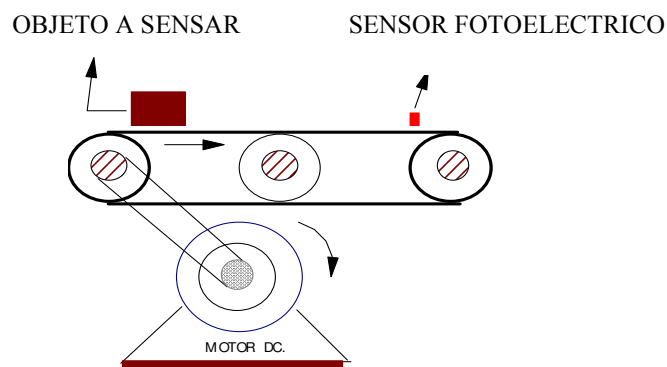


Figura 9. Esquema banda transportadora

2.4 CONTROL EN CRUCE DE VÍAS.

A través de éste modulo se implementará un semáforo que controlará el cruce de vehículos y peatones en la intersección de dos calles, teniendo presente alternativas de preferencias en calles controlando los tiempos de duración de encendido de luces en los semáforos la Figura 10. describe la calle a controlar.

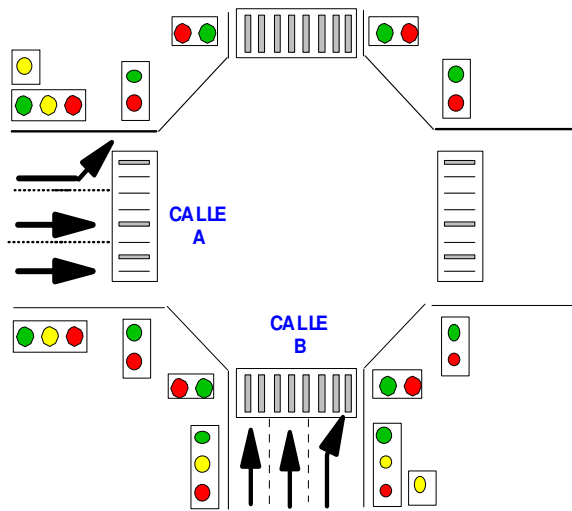


Figura 10. Semáforo en Cruce de Vías

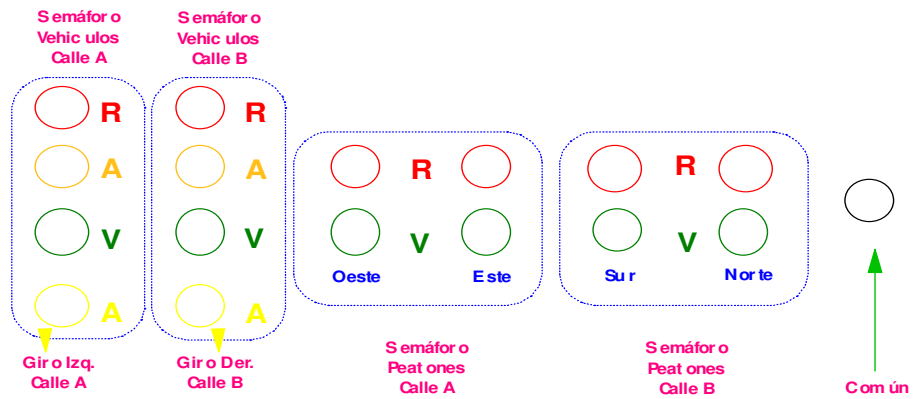


Figura 11. Bornera para la conexión del control en el cruce de vías

El módulo del Semáforo en Cruce de Vías presenta un conexionado interno, de tal manera que la bornera se muestra lista para recibir las señales de control que vienen del PLC, el punto denotado como común es la referencia de todas las conexiones de los LED que forman el semáforo.

3. CARACTERÍSTICAS DEL PLC SIMATIC S7 – 200 CPU 214 DE SIEMENS.

En el anteproyecto se planteó la utilización de un PLC de Gama Media, se tomó como base el Siemens S5 – 95U, el cual presenta características básicas para la implementación de cada módulo, pero al momento de adquirir el PLC se encontró en el mercado de automatismos un Modelo nuevo de Siemens, el Simatic S7 – 200, el cual nos brindó características mejores en lo que a PLC de gama media se refiere, y con la CPU 214 se obtuvieron excelentes resultados ya que sus salidas a relé nos permiten un mejor desarrollo de los módulos que contiene el proyecto, en cuanto al manejo de entradas y salidas, y la versatilidad al momento de interconectar varios PLC del mismo tipo.

Se consideró el modelo con salidas a relé porque nos permite implementar módulos con niveles de tensión desde 5 – 30 VDC y de 120 – 230 VAC, lo cual es una gran ventaja en el momento de incrementar y variar el número de módulos para las prácticas de laboratorio.

La CPU 214 facilita el control destinado a tareas exigentes; con un alto número de entradas / salidas, gran memoria y más funciones especiales integradas. A continuación se detallan sus características mecánicas y técnicas más importantes.

3.1 CARACTERÍSTICAS MECÁNICAS

El Simatic S7 – 200 presenta las siguientes características mecánicas:

- Caja de plástico rígida y compacta.
- Elementos de conexión y control de fácil acceso.
- Ensamblaje en estándar horizontal o vertical.
- Bloque terminal ensamblado y conectado como permanente.

3.2 CARACTERÍSTICAS TÉCNICAS

- Estándares internacionales; el Simatic S7-200 presenta los requerimientos en seguridad VDE, UL, CSA y FM. La calidad en el sistema de fabricación empleado está certificado por el ISO 9001.
- Integridad en los datos; el programa de usuario y los más importantes parámetros de configuración están almacenados en la EEPROM interna.
- Fuente de alimentación DC 24V integrada, destinada para la directa conexión de sensores y actuadores.
- Entradas / salidas digitales integradas
(CPU 214 con 14 entradas y 10 salidas).
- Tomas de interrupción.
- Contadores de alta velocidad.
- Fácil expansión.
- Simulador (opcional).

- Potenciómetros analógicos.

3.3 CARACTERÍSTICAS ADICIONALES DE LA CPU 214:

- 2 salidas de impulsos de alta frecuencia.
- Reloj en tiempo real.
- Submódulo de memoria EEPROM (opcional).
- Módulo de batería para copia de seguridad de gran longitud.

3.3.1 Funciones de la CPU 214. La CPU 214 posee una serie de funciones que se describen así:

- Rápida ejecución de instrucción;
Tiempos de ejecución de 1.2 ms o 0.8 ms
- Juego de instrucciones extendido;
Una amplia variedad de operaciones básicas como lógica binaria, asignamiento de resultado, escritura, conteo, generación de tiempos, carga, transferencia, comparación, intercambio, rotación, generación de complementos, llamadas a subrutinas, instrucciones de comunicación integradas y otras funciones útiles como modulación de duración de pulso, funciones para manejar pulsos, funciones aritméticas, aritmética en coma flotante, control en lazo cerrado PID, funciones de salto, funciones de bucle y conversiones para simplificar la programación.
- Conteo.

- Control de interrupciones:
 - Interrupciones controladas por evento.
 - Interrupciones controladas por tiempo.
 - Interrupciones por conteo.
 - Interrupciones de comunicaciones.
- Control y supervisión directa de entradas y salidas.
- Protección mediante clave de acceso:
 - Acceso total.
 - Acceso de sólo lectura y Protección completa
- Funciones de diagnóstico y depurado.
- "Forzado" de entradas y salidas en modos de diagnóstico y depurado.

3.3.2 Comunicaciones. El Simatic S7 – 200 CPU 214 presenta un sistema de comunicación muy versátil, el cual permite a través de la PPI (Interface Punto a Punto) integrada, un amplio rango de posibilidades de comunicación.

Con la interface RS 485 puede ser utilizado en dos modos:

- **Modo PPI:**

Varios PLC's Simatic S7-200 pueden ser manejados o monitorizados sobre una línea simple de dos conductores. Las conexiones punto a punto pueden ser realizadas con las siguientes unidades:

 - Dispositivos de programación PG 720/720C, PG 740 y PG 760.
 - PC's.

- Display de texto TD 200;

Paneles de operación COROS OP3; OP5, OP15, OP25 y OP35.

- SIMATIC S7-200.

- **Modo interfaz de programación libre:**

Especificado por el usuario, pueden ser empleados protocolos de comunicación orientados a bit (ej. un protocolo ASCII o un Modo bus).

Pueden ser establecidas conexiones punto a punto con:

- Cualquier DTE que posea interfaces serie, ej. impresoras o lectores de códigos de barras.
- S7-200 micro PLC, ej. para el simple intercambio de datos entre dos CPU's.
- Las altas tasas de transferencia del PROFIBUS-DP proporcionan una total utilización de las características en tiempo real del S7-200 dentro de una estructura de distribución en planta.
- Los tiempos de respuesta en el rango de los milisegundos permite, por ejemplo:
 - Sincronización de motores.
 - Aplicaciones orientadas a interrupción.
 - Coordinación de operaciones de contaje en alta velocidad

En el Cuadro 1 se muestra los datos técnicos del PLC SIMATIC S7 – 200 CPU 214, y a través de un cuadro comparativo se muestran las características de los PLC SIMATIC S7, ver Cuadro 2.

La distribución de los bornes de conexión al PLC se describen y detallan en la Figura 12, y en La Figura 13 se muestra estos puntos de conexión en la Bornera del Banco.

Cuadro 1. Datos Técnicos PLC Simatic7 – 200, CPU 214

	CPU 214
Memoria de programa	4 KB / aprox. 2 K líneas
Memoria de datos	2,048 palabras
Módulos de memoria (opcional)	1 incluida (EEPROM);
Lenguaje de programación	STEP 7 Micro/WIN o STEP 7 Micro/DOS
Ejecución del programa	- Ciclo libre - Por interrupción - Por tiempo (5 to 255 ms)
Tiempo de ejecución para operaciones de bit	0.8 ms
Marcas	256
Contadores	128
Temporizadores	128
Funciones rápidas integradas:	4
- Entradas de interrupción	1 contador bidireccional
- Contadores	2 contadores rápidos
- Salidas de pulso	2
Interfaces	- Comunicación RS 485 - Bus Backplane
E/S integradas:	
- Entradas digitales	14
- Salidas digitales	10
- Potenciómetros analógicos	2
E/S conectables:	
- E/S digitales	Máx. 64 entradas y 64 salidas
- E/S analógicas	16 entradas y 16 salidas
Grado de protección	IP 20

Cuadro 2. Cuadro Comparativo CPU 214, Salidas D.C, Relé, y A.C.

CPU 214	DC	Relés p lectura / m lectura	AC entrada 24 V / entrada 120 V
Tensión de alimentación Potencia perdida Corriente de salida	DC 24 V 8 W 280 mA	AC 120 - 230 V 9 W 280 mA	AC 120 - 230 V 11 W 280 mA
<u>Entradas integradas</u> Tensión de entrada Aislamiento	14 DC 24 V Optoacoplador	14 DC 24 V Optoacoplador	8 AC 24 V(e. 24V) AC 120 V(e. 120V) Optoacoplador
<u>Salidas integradas</u> Tensión de carga típica Aislamiento Corriente de salida	10 (transistor) DC 24 V Optoacoplador Máx. 0.75 A	10 (relé) DC 24 V/ AC 24 a 230V Relé Máx. 2 A	10 (triac) AC 24 a 230V Optoacoplador Máx. 1.2 A
Dimensiones en mm Peso aproximado	197 x 80 x 62 390 g	197 x 80 x 62 490 g	197 x 80 x 62 490 g

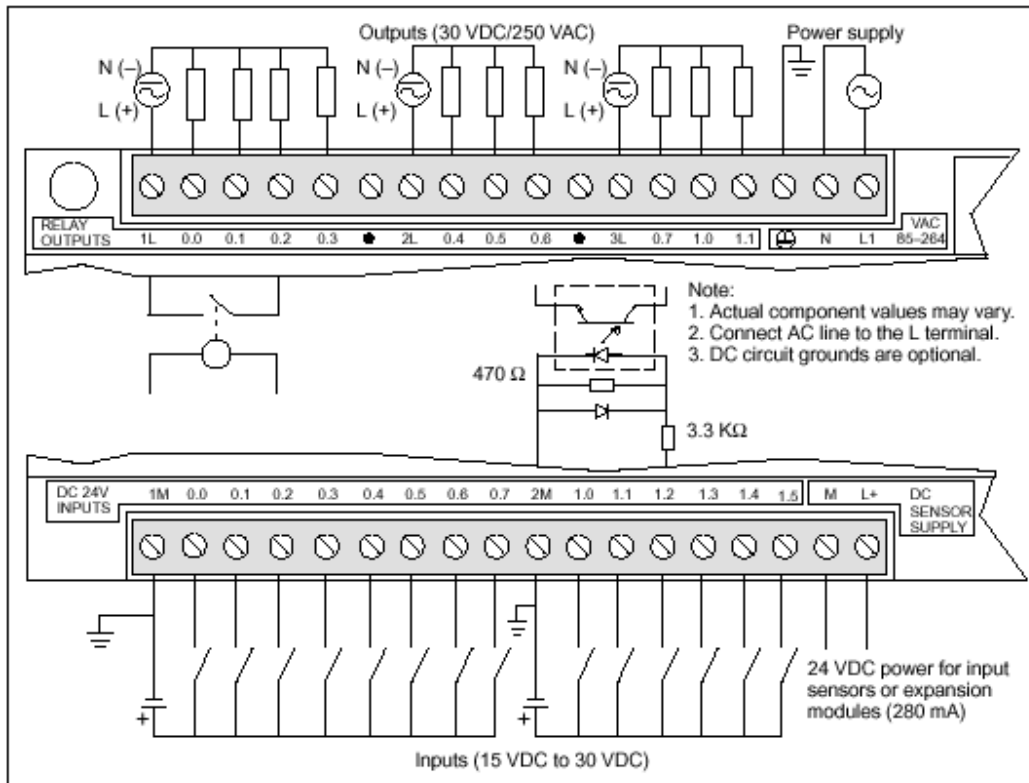
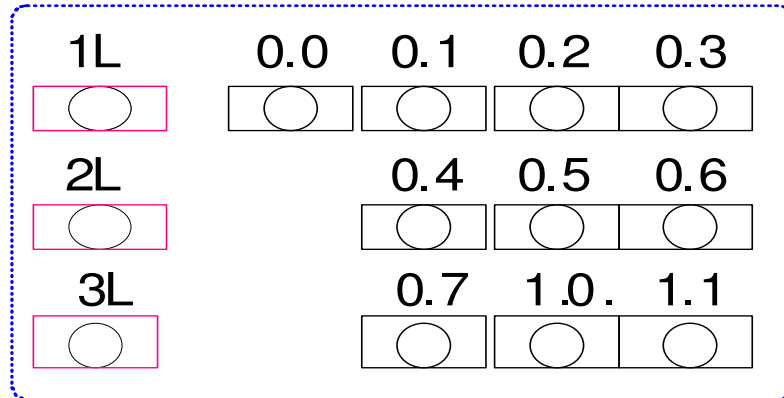
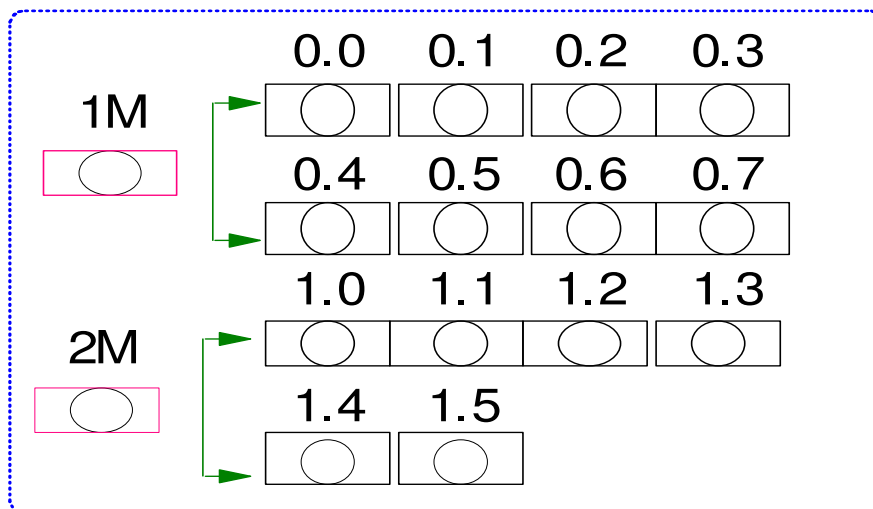


Figura 12. Identificación de los terminales de conexión para la CPU 214 AC/DC/Relay

RELAY OUTPUTS



INPUTS (15VDC to 30 VDC)



24 V DC Sensor Supply

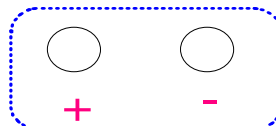


Figura 13. Bornera de conexión del PLC ubicada en el banco de pruebas.

4. INTRODUCCIÓN A LA PROGRAMACION KOP Y AWL

La versión SIMATIC CPU 214 dispone de entradas y salidas así como de una unidad de procesamiento central (CPU). El sistema internacional denota las entradas y salidas como I0.0, Q0.0 respectivamente, mientras que en el SIMATIC las entradas son E0.0 y las Salidas A0.0.

El PLC S7 – 200 puede programarse mediante los dos lenguajes siguientes:

- Lista de instrucciones (AWL), donde cada instrucción del programa comprende una operación que utiliza una abreviatura nemotécnica para representar una función del PLC.
- Esquema de contactos (KOP), que es un lenguaje gráfico cuyos símbolos se asemejan a los elementos de un esquema de circuitos.

Para crear un programa se utiliza uno de los siguientes métodos:

- STEP 7 – Micro/Dos, un paquete de programación sobre la plataforma del DOS.
- STEP 7 – Micro/Win, un paquete de programación sobre la plataforma de WINDOWS.
- El PG 702, una unidad de programación portátil o una PC.

Todos los programas S7 – 200 tienen que encajar en la estructura del programa principal, al que le siguen subrutinas y rutinas de interrupción. Ver Figura 14.

- El programa principal se termina con la instrucción MEND (Finalizar Programa)
- Las Subrutinas son una parte opcional del programa. Para poder utilizar una subrutina en los programas tiene que añadirse al final del programa principal, donde se encuentra la instrucción MEND.

- Las rutinas de interrupción también son opcionales. Para poder utilizarlas hay que agregarlas al final del programa principal, detrás de la instrucción MEND.
- Agrupando al final del programa todas las subrutinas, seguidas de todas las interrupciones, se obtiene un programa bien estructurado que resulta fácil de leer y comprender. Las interrupciones y las subrutinas pueden mezclarse a voluntad.

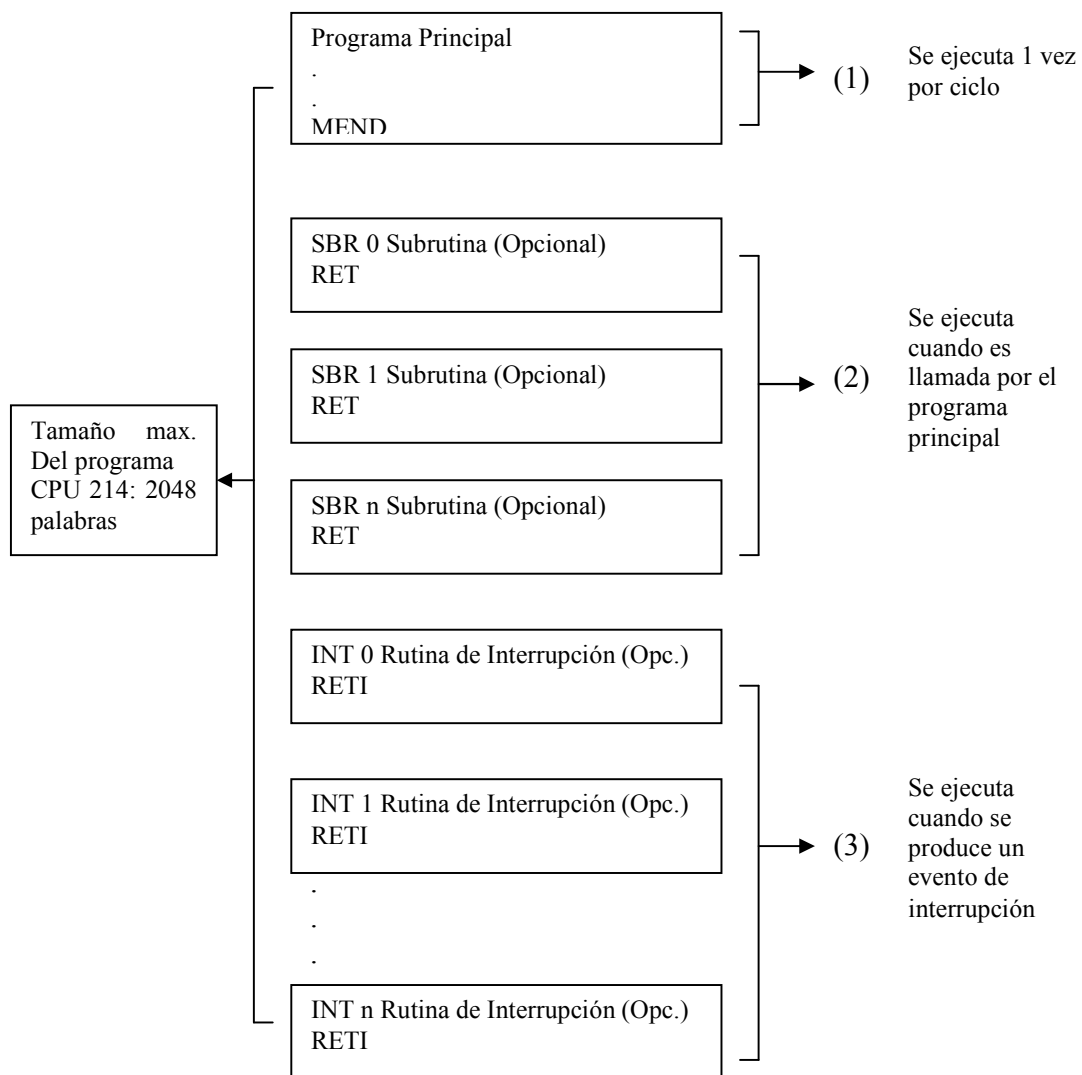


Figura 14. Estructura del Programa

4.1 OPERACIONES LÓGICAS CON CONTACTOS.

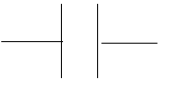
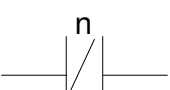
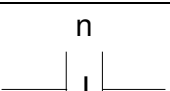
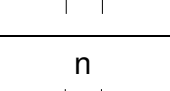
Las operaciones lógicas con contactos sirven para crear y conectar circuitos lógicos.

4.1.1. Descripción. En el lenguaje de programación KOP, los contactos pueden ser contactos normalmente abiertos y contactos normalmente cerrados. Las operaciones Cargar, Y y O con bits se usan en AWL para función de contacto normalmente abierto. Para la función de contacto normalmente cerrado se usan operaciones Cargar valor negado, Y-NO y O-NO.

En AWL y KOP se usa una I (immediate) para indicar que la operación se va a ejecutar directamente. Esta operación directa, o contacto, lee el valor direccionado en la entrada física al ejecutarse la operación o contacto. Sin embargo, la imagen de proceso no se actualiza. La operación no directa, o contacto, lee el valor direccionado en la imagen de proceso, que se actualiza por el autómata antes de cada ciclo.

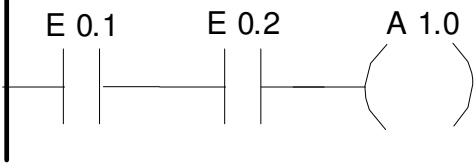
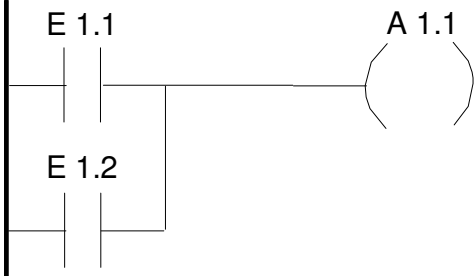
4.1.2. Representación. A continuación se representan las operaciones lógicas con contactos en los lenguajes KOP y AWL.

Cuadro 3. Operaciones Lógicas con Contactos en los Lenguajes KOP y AWL.

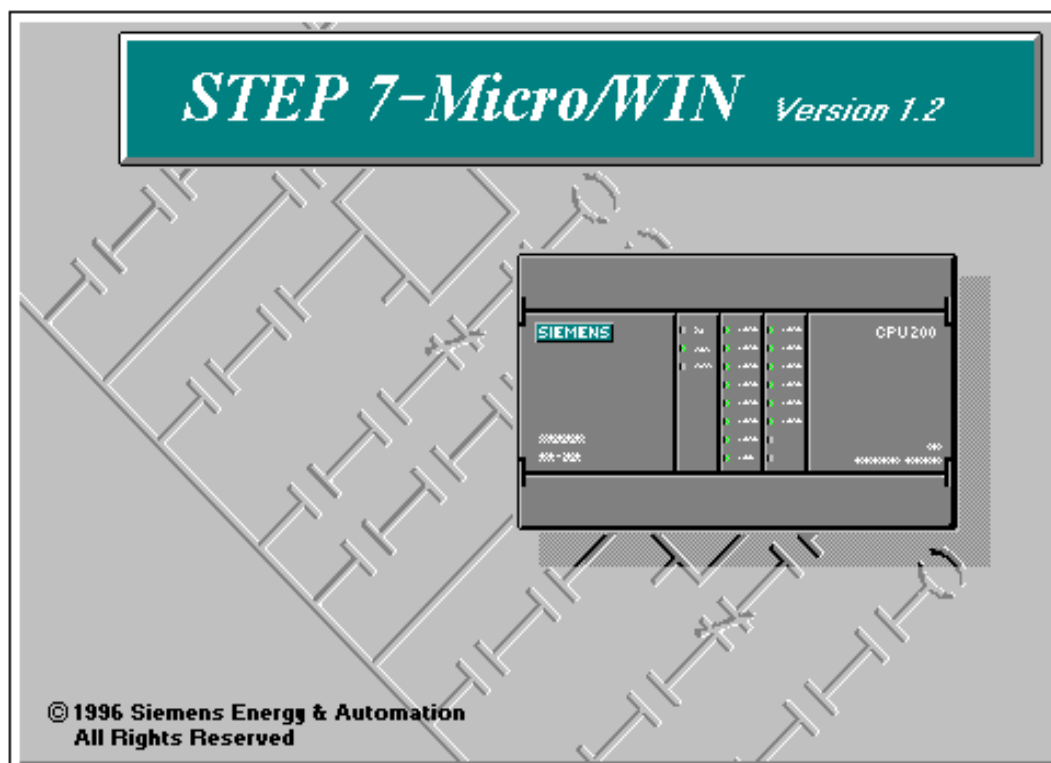
KOP	Descripción	Operandos
	El <i>Contacto Abierto</i> se cierra (se activa) si $n = 1$	$n : E, A, M, SM,$ T, Z, V (Bit)
	El <i>Contacto Cerrado</i> se Activa si $n = 0$	
	El <i>Contacto Abierto Directo</i> se Activa si $n = 1$	$n : E$ (Bit)
	El <i>Contacto Cerrado Directo</i> se Activa si $n = 0$	

AWL	Descripción	Operandos
LD n U n O n	Las operaciones Cargar (LD), Y (U) y O (O) cargan el valor n en el nivel superior de la pila y lo combinan mediante Y y O	$n : E, A, M, SM,$ T, Z, V (Bit)
LDN n UN n ON n	Las operaciones Cargar Valor Negado (LDN), Y-NO (UN) y O-NO (ON) cargan el valor n invertido en el nivel superior de la pila y lo combinan mediante Y u O	
LDI n UI n OI n	Las operaciones Cargar Directamente (LDI), Y Directa (UI), O Directa (OI) cargan el valor n negado en el nivel superior de la pila y lo combinan mediante Y u O	$n : E$ (Bit)
LDNI n UNI n ONI n	Las operaciones Cargar Valor Negado Directamente (LDNI), Y Directa (UNI), O-NO Directa (ONI) cargan el valor n negado en el nivel superior de la pila y lo combinan mediante Y u O	

Cuadro 4. Ejemplos de Operaciones Lógicas con Contactos KOP y AWL

KOP	AWL
	<pre>LD E 0.1 U E 0.2 = A 1.0</pre>
	<pre>LD E 1.1 O E 1.2 = A 1.1</pre>

4.2 MANEJO DEL PROGRAMA STEP 7 MICRO/WIN



Para ejecutar el programa hacemos clic sobre el botón Inicio; escogemos Programa, y luego STEP7 - Micro/WIN, del submenú Programas. Después de unos segundos se vera una ventana como la que se muestra en la Figura 15.

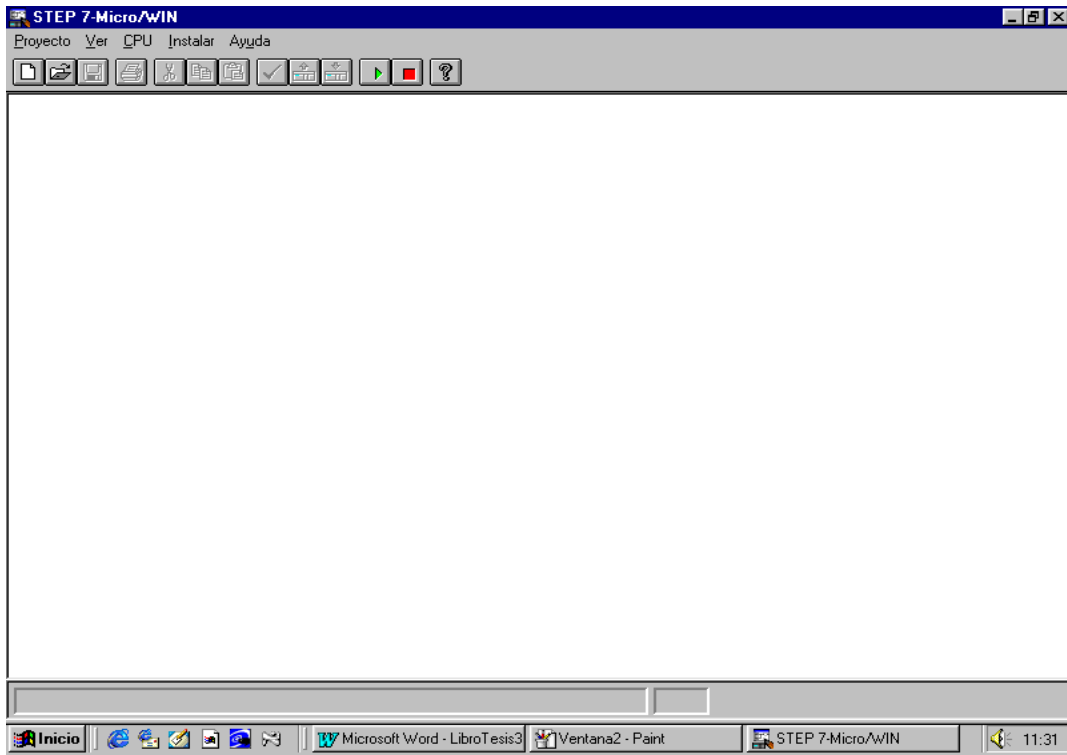


Figura 15. Ventana Inicial del Step7-Micro/Win

4.2.1 Configuración del PLC. Al momento de crear un nuevo proyecto se debe definir las opciones predeterminadas que se han de visualizar cada vez que se abra un proyecto, elija el comando Preferencias en el menú Instalar. La Figura 16. Muestra la ventana que se despliega.

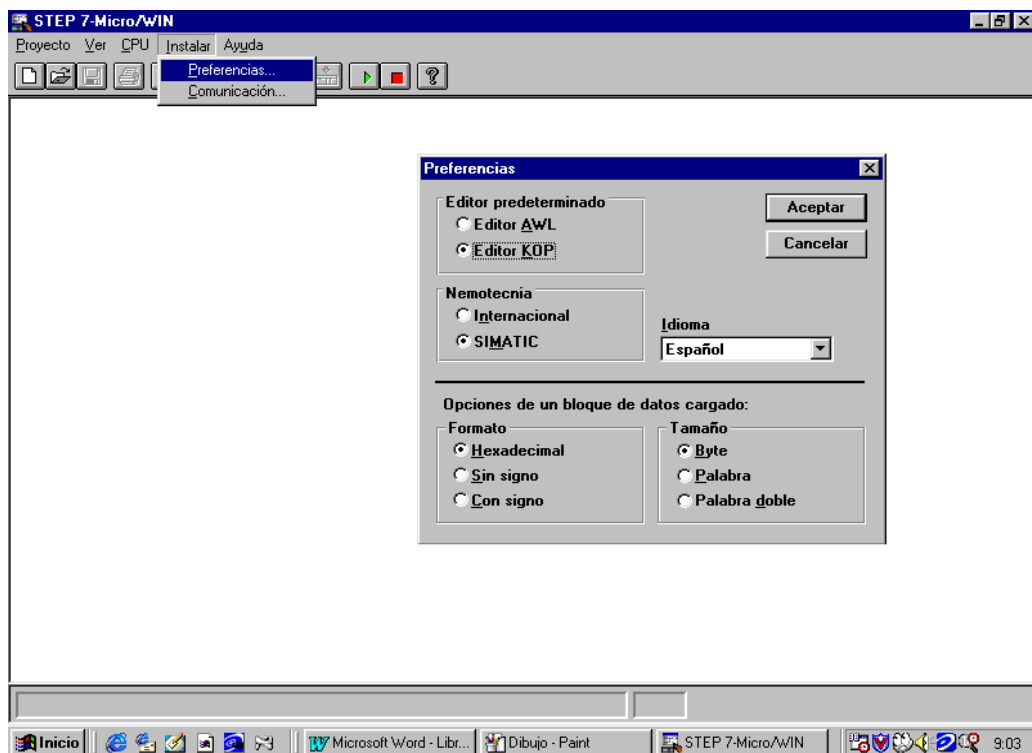


Figura 16. Ventana para configurar las preferencias del PLC

En ésta ventana podemos elegir las preferencias de programación tales como:

Editor

Elija el editor predeterminado haciendo clic en el botón de opción AWL o KOP.

Nemotecnia

Elija la nemotecnia haciendo clic en Internacional o Simatic. Si desea cambiar la nemotecnia debe salir de Micro/WIN y reiniciar el programa para poder activar la nueva nemotecnia.

Idioma

En la lista desplegable, elija el idioma con el que desea usar Micro/WIN.

Bloque de datos cargado

- Active la casilla de opción Hexadecimal, Decimal sin signo o Decimal con signo para seleccionar el formato numérico.

- Active la casilla de opción Byte, Palabra o Palabra doble para seleccionar el tamaño del formato de datos.

Los ajustes relativos al formato del bloque de datos se aplican cuando uno de dichos bloques se carga de la CPU.

También del menú Instalar eligiendo Comunicación se configura el puerto en el cual está la interfase y se direcciona la CPU. Ver la Figura 17.

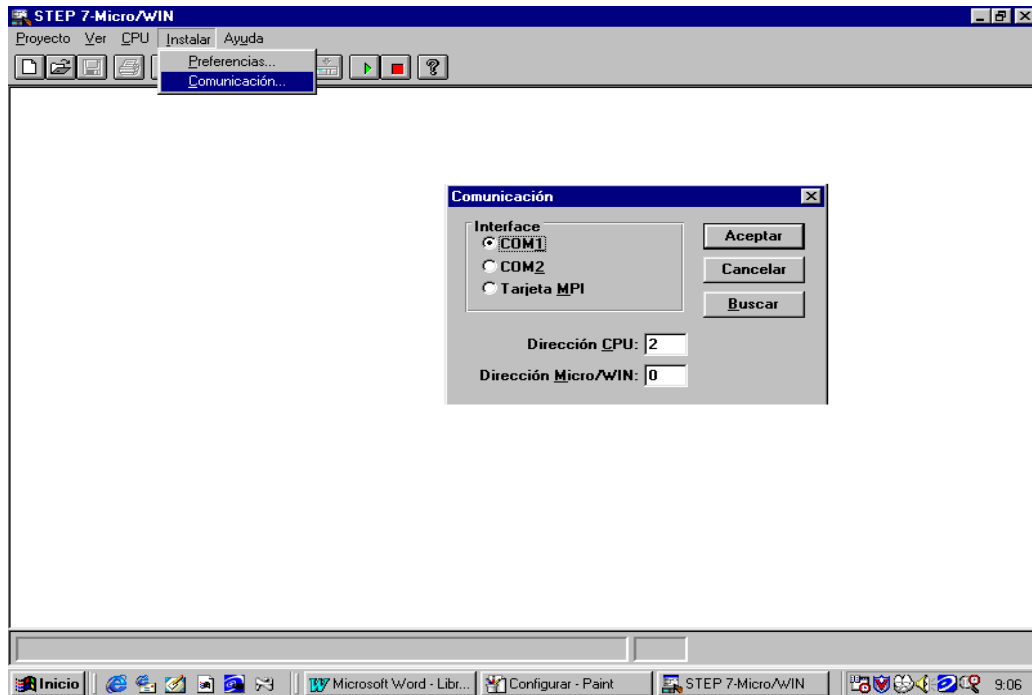


Figura 17. Ventana que muestra la configuración del puerto y dirección de la CPU.

Cuadro de diálogo "Comunicación"

Interface

Establece el enlace de comunicación con su unidad de programación. Haga clic en la casilla de verificación COM1 ó COM2 (dependiendo de la interface de enlace con la CPU) o de la tarjeta MPI.

Dirección Micro/WIN

Introduzca la dirección de red de Micro/WIN. El rango de direcciones válidas comprende entre 0 y 126. Por lo general, la dirección de Micro/WIN es 0.

Dirección CPU

Introduzca la dirección de red de la CPU que desea utilizar. El rango de direcciones válidas comprende entre 0 y 126. (Tenga en cuenta que esta dirección se debe configurar en el cuadro de diálogo "Configurar CPU").

Buscar

Haga clic en el botón Buscar para visualizar la dirección de la CPU conectada actualmente. Si hace clic en el botón Siguiente se visualizan todas las direcciones de CPU en la red.

Aceptar


Haga clic en el botón Aceptar para guardar sus cambios.


Cancelar

Haga clic en el botón Cancelar para cerrar el cuadro de diálogo sin efectuar cambios.

Una vez terminada la configuración de Preferencias y Comunicación del PLC, reiniciamos el programa para hacer efectivos los cambios.

4.2.2. Crear proyectos

1. Elija el comando Nuevo en el menú Proyecto o haga clic  en la barra de herramientas para crear los archivos.

2. Elija el comando Guardar en el menú Proyecto o haga clic  en la barra de herramientas para guardar el proyecto.

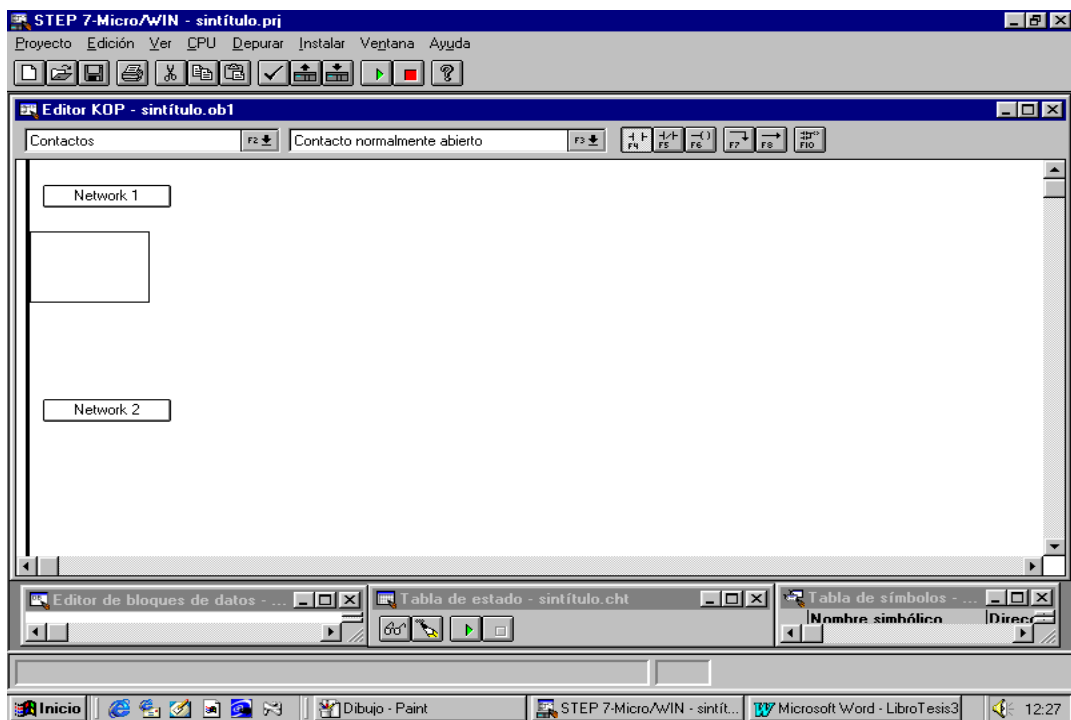


Figura. 18. Ventana para iniciar la programación

Cuando elegimos el comando Nuevo, Micro/WIN crea archivos para un bloque lógico, para un bloque de datos, para una tabla de estado, para comentarios y para la configuración. Dichos archivos conforman el nuevo proyecto.

Seguidamente como lo muestra la Figura 19, puede elegir si desea programar con un editor KOP o AWL. El editor predeterminado es el de KOP. No obstante, si desea programar en AWL, puede cambiar el ajuste utilizando el comando Preferencias en el menú Instalar. Así evita tener que seleccionar en el menú Ver el editor en cada sesión de programación.

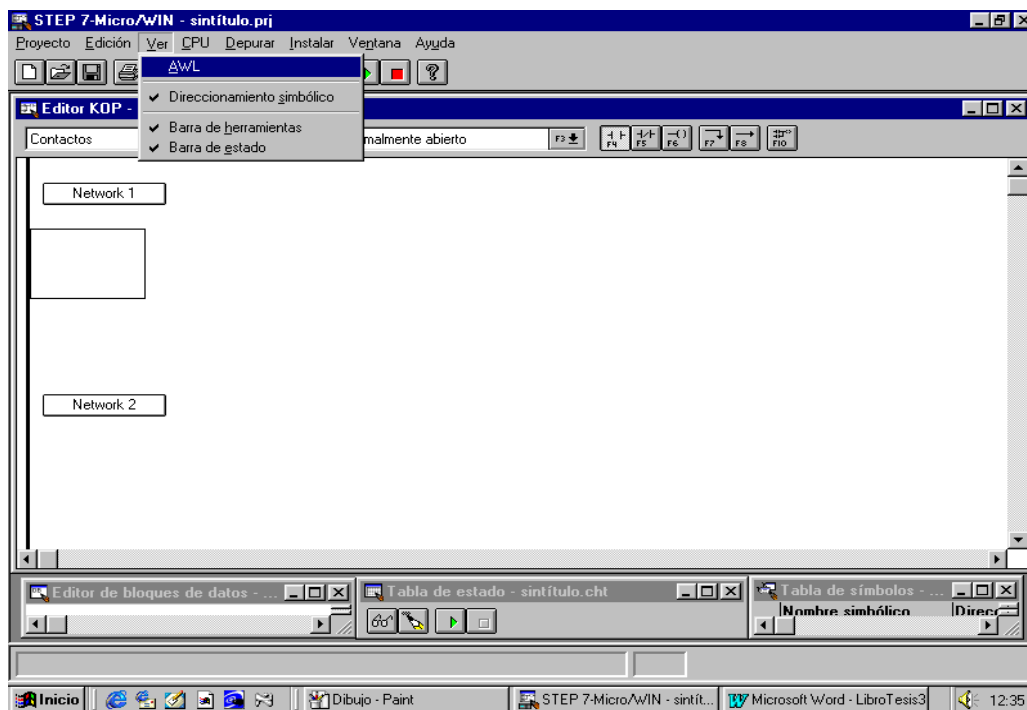


Figura. 19 Haciendo clic en el botón VER se despliega la ventana que muestra la opción AWL, al hacer clic en AWL se despliega la ventana correspondiente a este formato.

Para visualizar mejor estas opciones realizaremos un ejemplo sencillo el cual consiste en encender y apagar una bombilla. Teniendo presente los pasos descritos anteriormente, en la ventana que muestra la Figura 18 se selecciona la barra que muestra los contactos, escogemos los contactos normalmente abierto para encender y normalmente cerrado para apagar implementando de esta manera el programa que se muestra en la Figura 20.

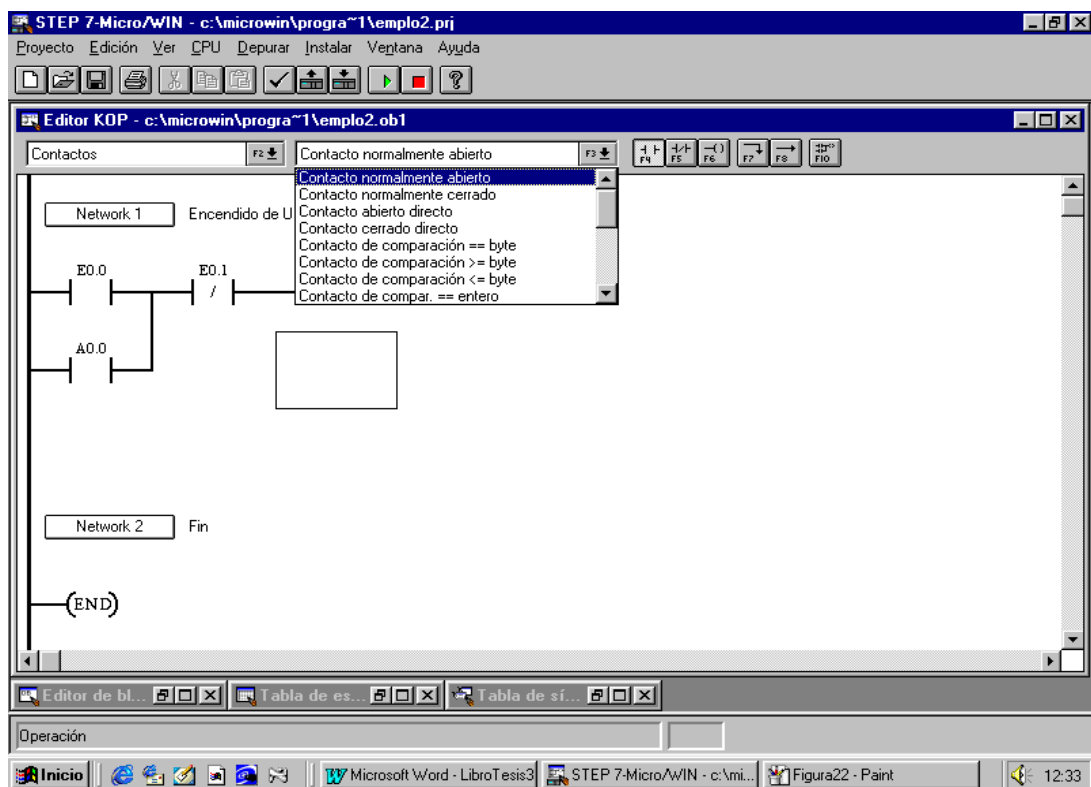


Figura .20 Ventana para la selección de Contactos

De manera similar se Asigna la Bobina de Salida y se finaliza el programa. En las Figuras 21 y Figura 22 se muestran las ventanas correspondientes a cada asignación.

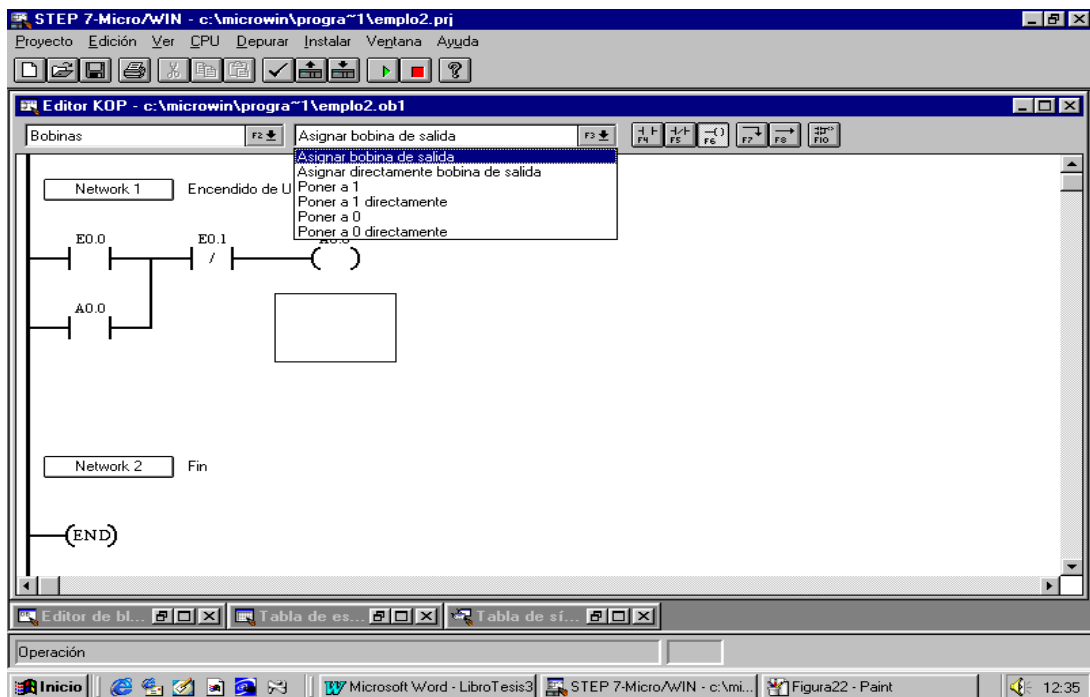


Figura 21 Ventana para la Asignación Bobina de Salida.

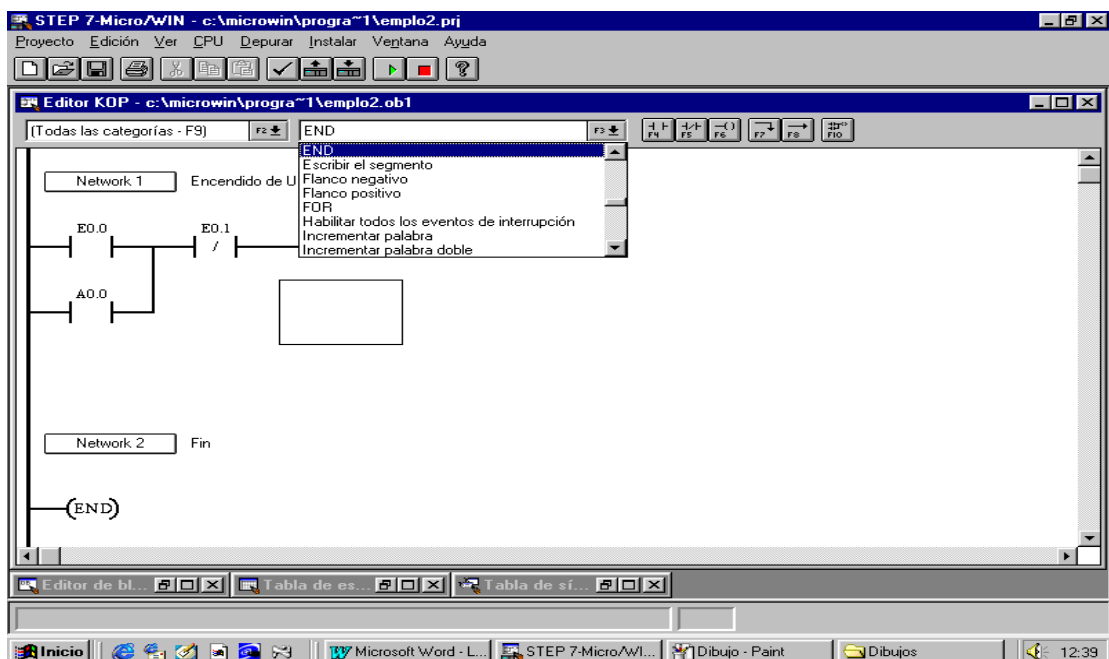


Figura. 22 Para finalizar el programa o rutina Elegimos END en Esta Ventana.

4.2.3 Compilar el Programa

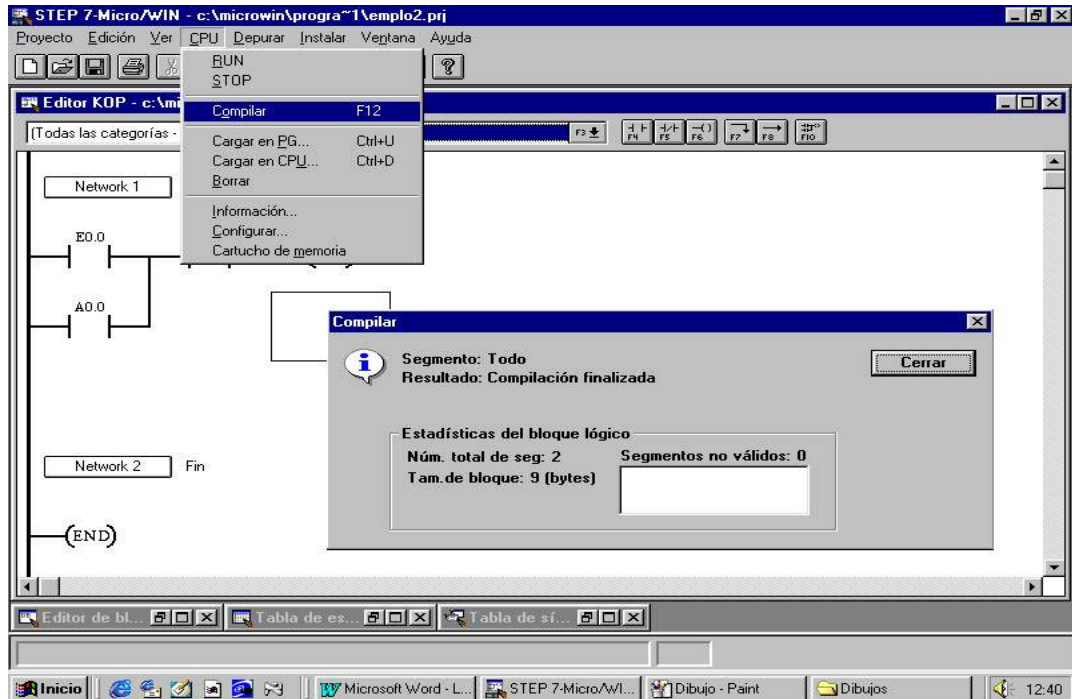


Figura 23 Cuadro de Dialogo del menú CPU,

En ésta ventana se compila el bloque lógico y el bloque de datos del programa en lenguaje máquina para que sea ejecutado por la CPU.

Para poder cargar un programa en la CPU es necesario compilarlo previamente.

4.2.4 Documentar el Programa. El programa se documenta mediante la Tabla de Símbolos, en la Figura 24 se puede apreciar la ventana y en la Figura 25 podemos ver el resultado de la asignación de Símbolos.

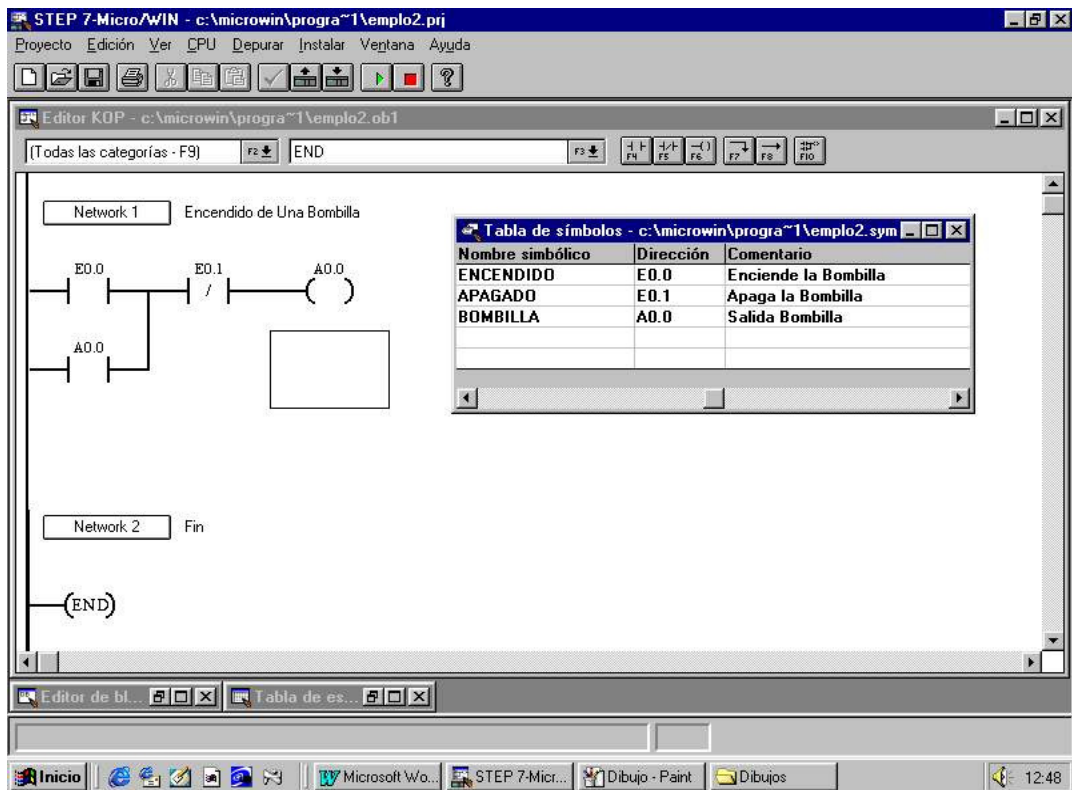


Figura 24. Haciendo clic en Tabla de Símbolos del menú Ventana se despliega un cuadro en el Cual podemos asignar nombre a cada Entrada o Salida del programa.

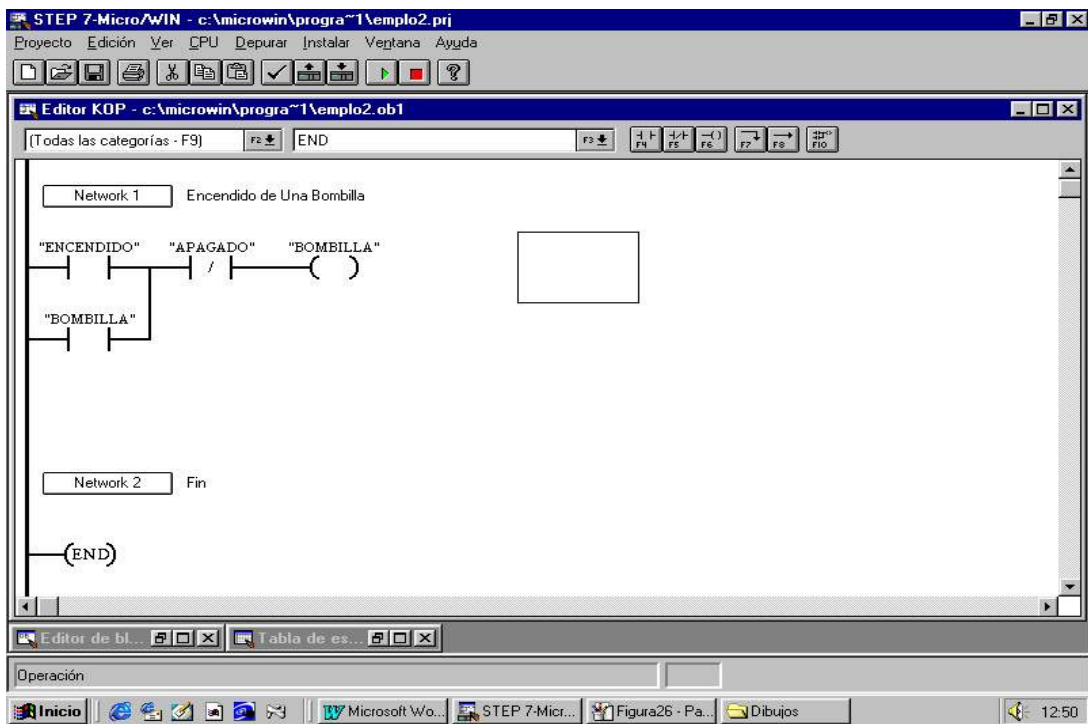



Figura 25 Una vez asignado los nombres a los símbolos el programa queda documentado y presenta una forma más fácil de interpretar.

4.2.5 Cargar un Proyecto en la CPU.

1. Conmute la CPU a modo STOP. Ver Figura 26.
2. Abra el proyecto que contiene el programa que desea cargar.
3. Elija el comando Cargar en el menú CPU o haga clic  en la barra de herramientas. Ver Figura 27.
4. Haga clic en la casilla de verificación de cada archivo que desea cargar.
5. Haga clic en el botón Aceptar para comenzar a cargar los archivos que seleccionó.

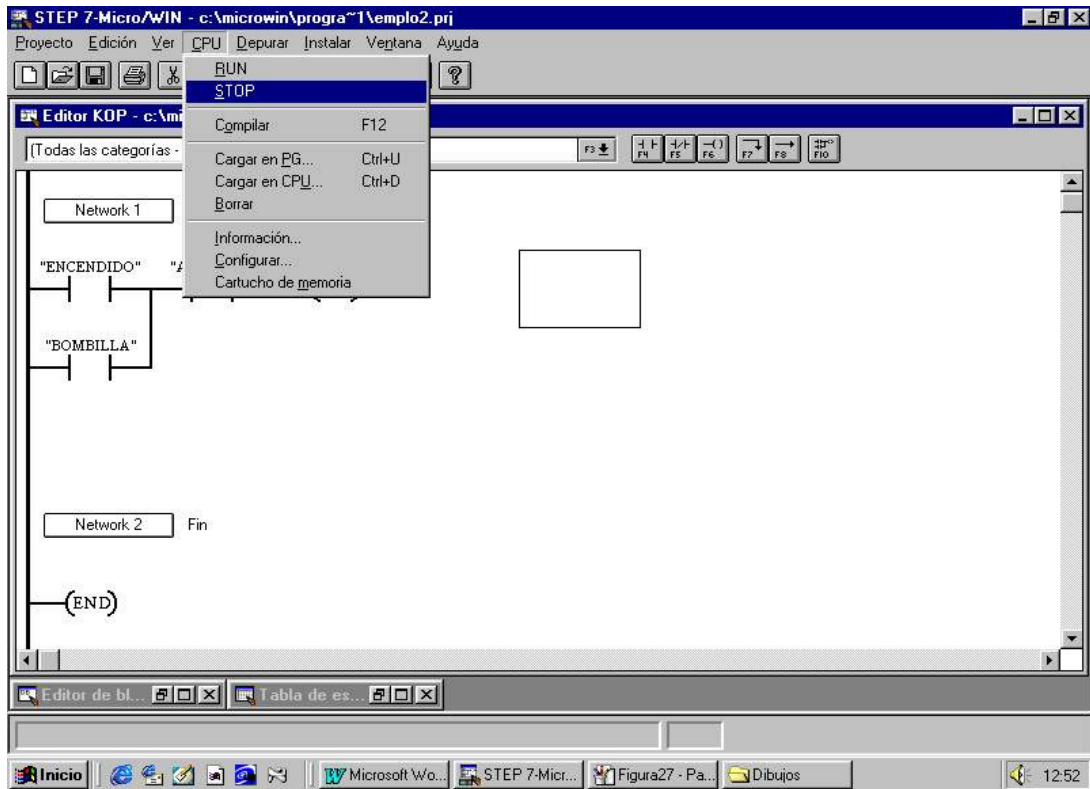


Figura 26. Ventana que nos permite colocar la CPU en STOP. También es posible colocar en STOP la CPU directamente desde el PLC.

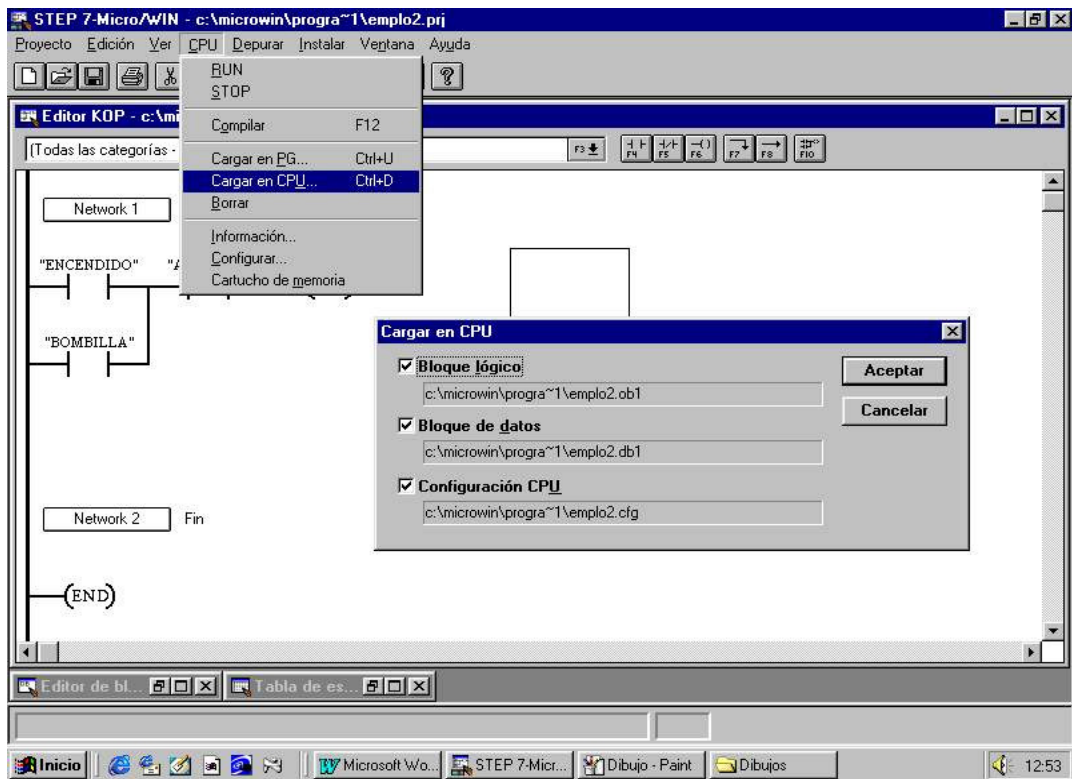


Figura 27. Ventana Cargar en CPU.

Copia en la CPU el proyecto de los archivos que se haya indicado. Los archivos que se cargan forman parte del proyecto que está abierto actualmente. Se debe tener en cuenta que el selector de modos de operación de la CPU debe estar en posición TERM y que la CPU se debe conmutar a modo STOP antes de iniciar el proceso de carga. Si el selector de la CPU se encuentra en posición TERM, es posible conmutar la CPU a modo STOP mediante Micro/WIN.

Cuadro de diálogo "Cargar en CPU"

Bloque lógico

Activar la casilla para seleccionar el bloque lógico (.ob) que desea cargar.

Bloque de datos

Activar la casilla para seleccionar el bloque de datos (.db) que desea cargar.

Configuración CPU

Activar la casilla para seleccionar la configuración de la CPU (.cfg) que desea cargar.

Aceptar

Haga clic en Aceptar para comenzar a cargar los archivos que haya seleccionado.

Cancelar

Haga clic en Cancelar para cerrar el cuadro de diálogo sin cargar nada en l

4.2.6 CPU en modo RUN. Al conmutar la CPU a modo RUN se inicia la ejecución del programa de usuario. Durante la ejecución del mismo se actualizan las entradas y salidas asociadas con el mismo. Ver Figura 28.

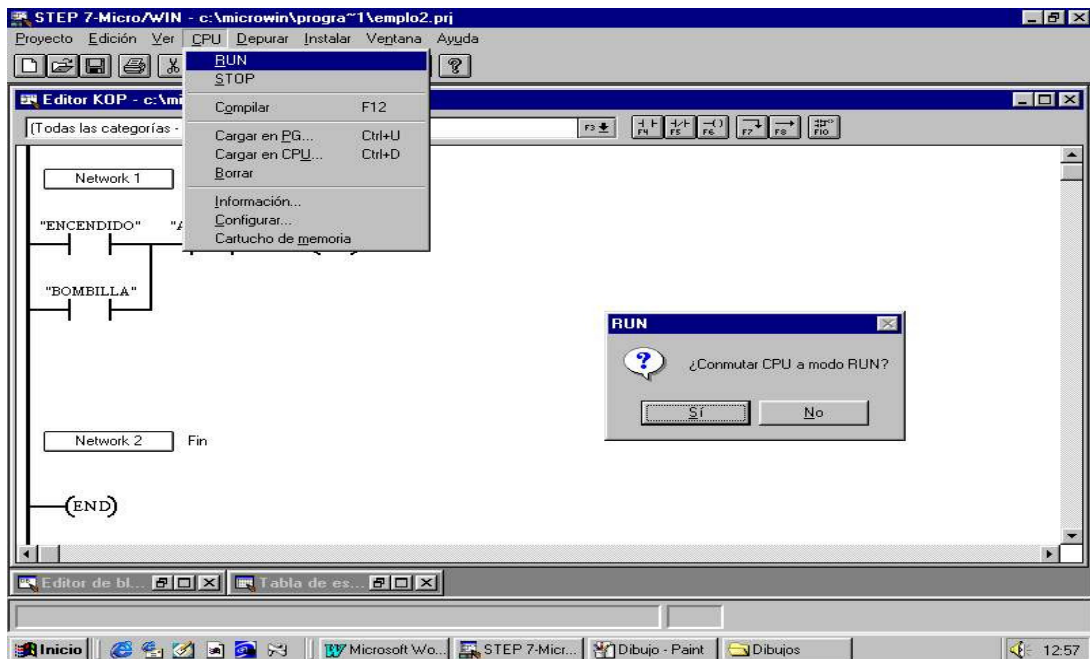


Figura No. 28 Ventana que nos muestra la conmutación de la CPU en modo RUN. Este es el paso final que nos permite ejecutar el programa.

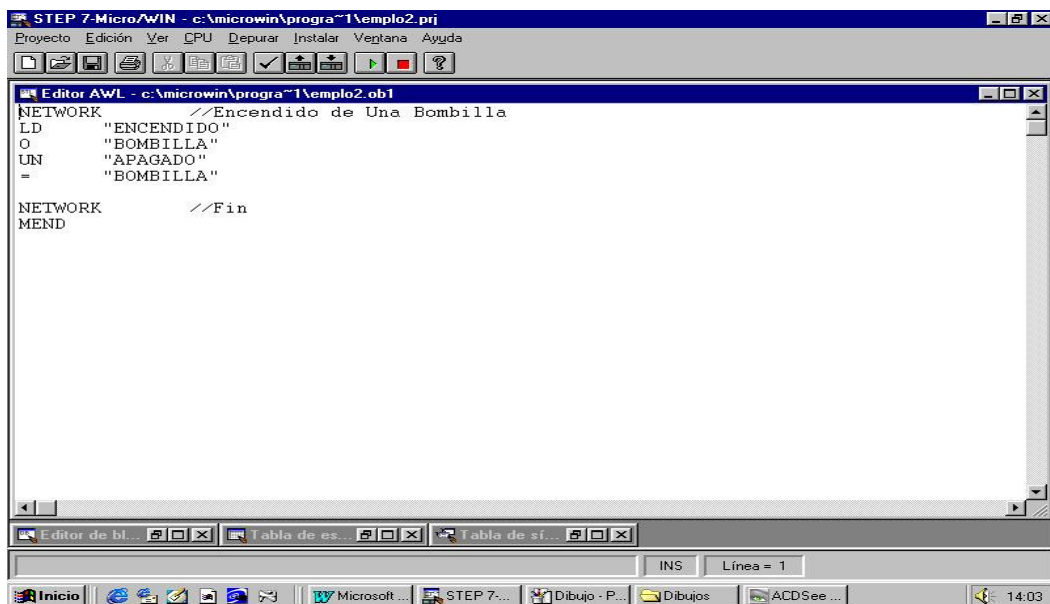


Figura. 29 Programa terminado en formato AWL

4.3 EJEMPLOS

4.3.1 Encendido de una Bombilla. Se desea encender una bombilla mediante un pulso inicial (START) y apagar mediante un Pulso (STOP).

Entradas I 0.0 (START) y I 0.1 (STOP)

Salidas Q 0.0 (Salida Bombilla)

Trabajaremos con el Start – Stop y una de las bombillas ubicadas en el Tablero de Control ubicado en el Banco de Pruebas.

1) Descripción del Programa

Mediante un pulsador Normalmente Abierto (N.O) asociado a la entrada I0.0 del PLC se activa la Bobina de Salida Q0.0 la cual encenderá la bombilla, luego mediante un pulsador Normalmente Cerrado (N.C) asociado a la entrada I0.1 del PLC se desenergiza la Bobina de Salida apagando así la bombilla.

2) Listado del Programa en Formato AWL

```
NETWORK //Encendido de Una Bombilla
LD      "ENCENDIDO"
O       "BOMBILLA"
UN      "APAGADO"
=       "BOMBILLA"
```

```
NETWORK //Fin
MEND
```

3) Listado del Programa en Formato KOP.

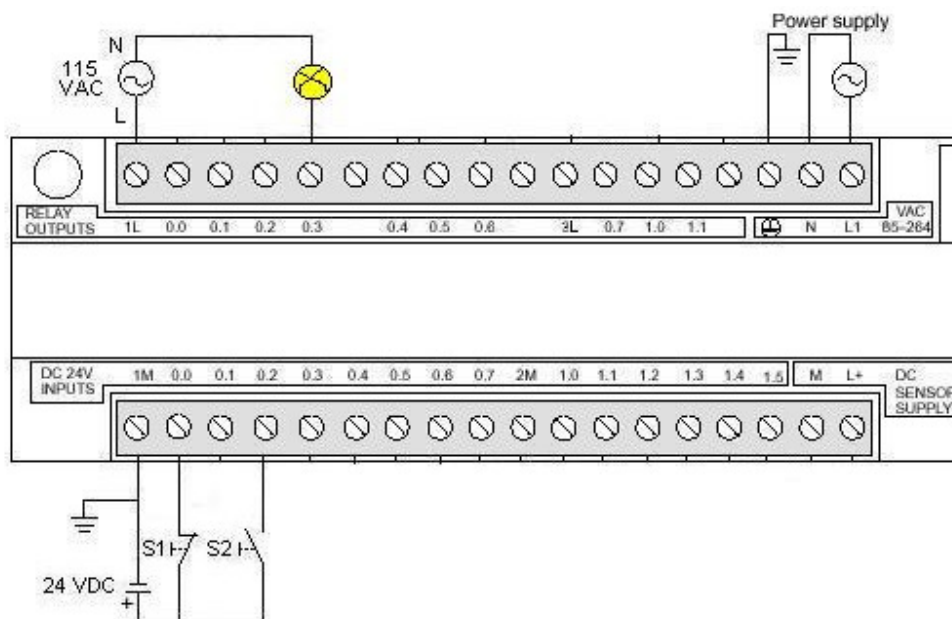
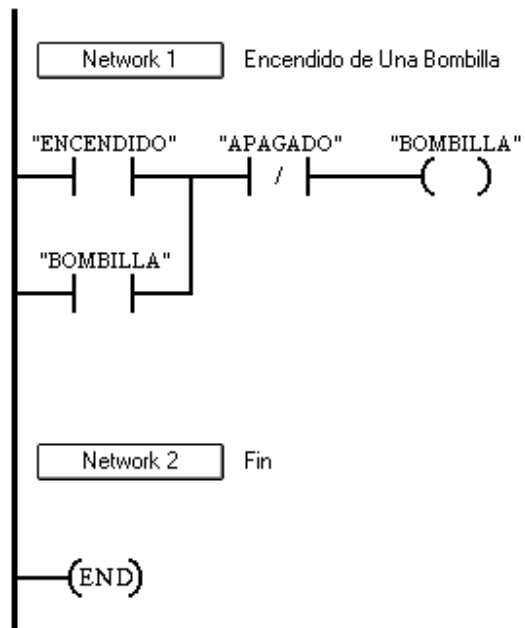


Figura 30. Esquema de conexión al PLC.

4.3.2. Llenado y Vaciado de un Tanque. Se desea controlar el motor de una bomba de agua.

El motor puede arrancarse y pararse directamente desde una estación de pulsadores (START – STOP) de marcha y paro. El motor trabaja a 12 VDC.

- Al ponerse en funcionamiento el motor, se inicia el llenado del tanque TK1 abriendo la válvula solenoide VS1.
- Al operar el interruptor de alto nivel HLS1 el motor se debe detener y VS1 se debe cerrar suspendiendo así el llenado.
- 5 segundos después se debe abrir la válvula solenoide VS3, para vaciar el tanque TK1. Terminando de ésta manera el proceso

1) Descripción del Programa

En el banco de pruebas se puede trabajar con cualquiera de los tanques TK1 o TK2, dependiendo cual elija se debe recordar que HLS1, VS1 y VS3 están asociadas a TK1, mientras que HLS2, VS2 y VS4 están asociadas al tanque TK2.

Entradas	Salidas
I0.0 Pulsador de Marcha	Q0.0 Motor de la Bomba
I0.1 Pulsador de Paro	Q0.1 Válvula Solenoide VS1
I0.2 Interrup. De Alto Nivel HLS1	Q0.2 Válvula Solenoide VS3

La Figura 31. muestra un dibujo ilustrativo del proceso.

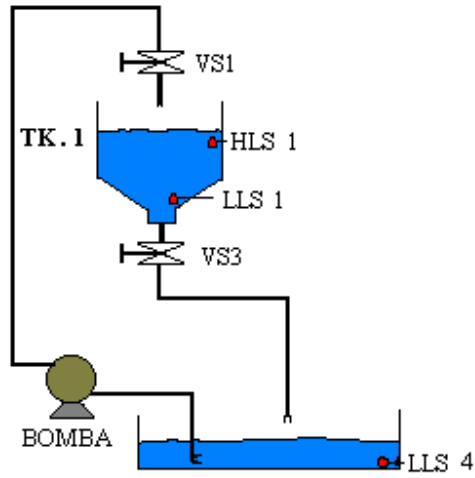


Figura 31 Dibujo ilustrativo del proceso de llenado y Vaciado de un tanque

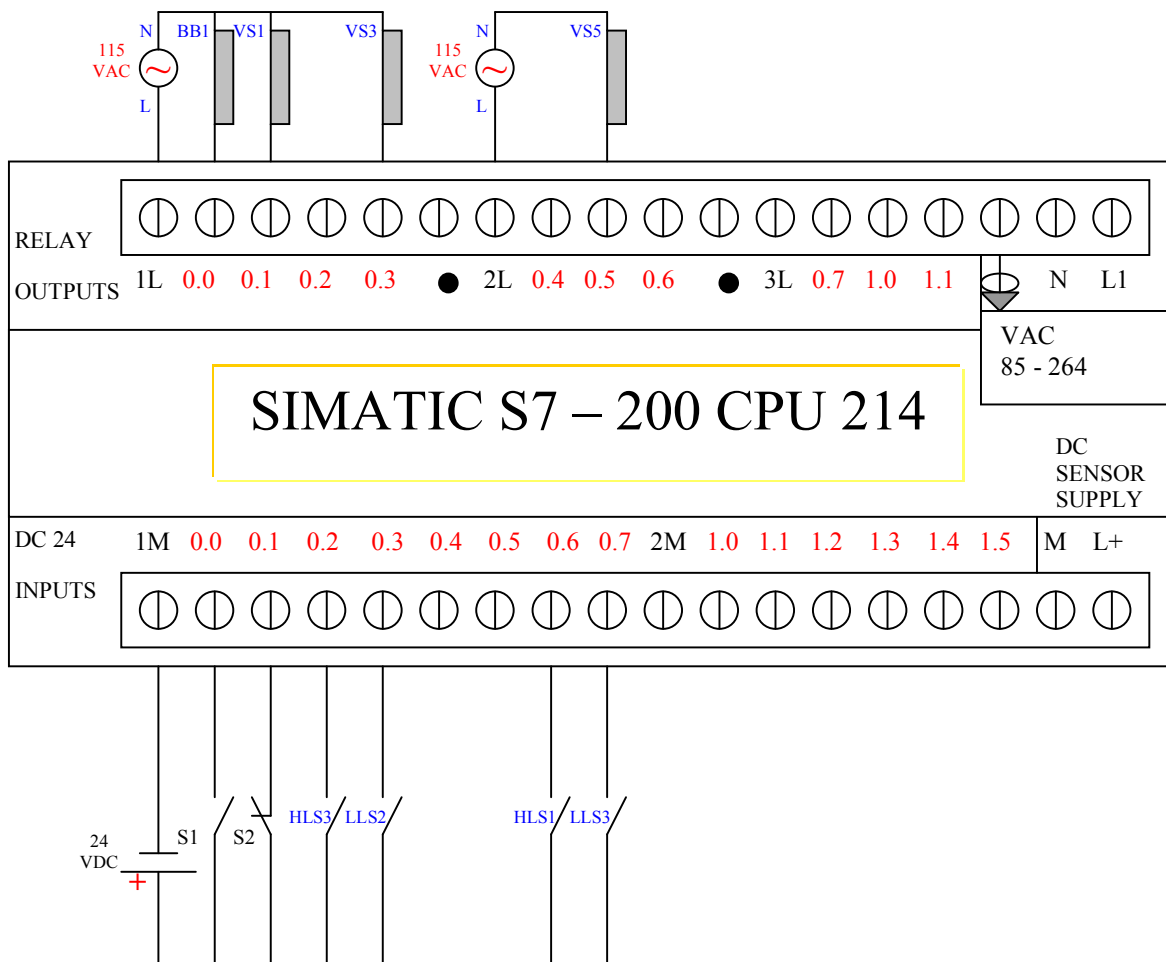


Figura 32. Esquema de conexión al PLC, del sistema Llenado y Vaciado de TK1

2) Listado del Programa en Formato AWL

```
NETWORK 1          //arranque del sistema se inicia el
proceso por medio de la marca m0.0
LD      E0.0
U      E0.6
O      M0.0
U      E0.1
=      M0.0

NETWORK 2          //apertura de valvula VS1
LD      M0.0
EU
S      A0.1, 1

NETWORK 3          //ciere de valvula VS1,cuando se da la
señal de alto nivel en el tanque se cierra la valvula
LDN     E0.6
TON     T39, +20

NETWORK 4
LD      T39
ON     M0.0
R      A0.1, 1

NETWORK 5          //encendido de la bomba;si las valvulas
están abiertas la bomba funcionara
LD      A0.1
U      M0.0
TON     T37, +30

NETWORK 6          //se espera el temporizado T37 para
activar la bomba
LD      T37
=      A0.0

NETWORK 7          //arranque temporizado para iniciar el
vaciado
LD      E0.6
ED
O      M0.1
U      M0.0
UN     T38
=      M0.1
TON     T38, +50

NETWORK 8          //arranque del vaciado tanque 1
LD      T38
ED
U      M0.0
S      A0.3, 1
```

```
NETWORK 9          //si se apaga el sistema se cierra VS3
LDN      M0.0
R        A0.3, 1

NETWORK 10         //aviso de llenado del tanque 3
LDN      E0.2
TON      T44, +30

NETWORK 11         //temporizado para el cierre de VS3
LDN      E0.4
TON      T41, +300

NETWORK 12
LD        T41
O        T44
R        A0.3, 1

NETWORK 13         //temporizador para el sensor de bajo
nivel en tanque 3
LDN      E0.7
TON      T46, +300

NETWORK 14         //vaciado tanque 3
LD        E0.7
U        M0.0
UN       A0.3
TON      T45, +50

NETWORK 15         //abrir valvula VS5
LD        T45
S        A0.5, 1

NETWORK 16
LD        T46
ON       M0.0
R        A0.5, 1

NETWORK 17
```

3) Listado del Programa en Formato KOP.

4.3.3 Manejo de Interrupciones Temporizadas. Las interrupciones temporizadas se utilizan para indicar tareas que deban ejecutarse cíclicamente. Energizando la entrada I 0.1 con un interruptor se reduce a la mitad la frecuencia del ciclo. Cerrando el interruptor de la entrada I 0.0 se restablece de nuevo la frecuencia original del ciclo.

Este ejemplo aclara las relaciones existentes entre las interrupciones temporizadas y el cambio de la base de tiempo.

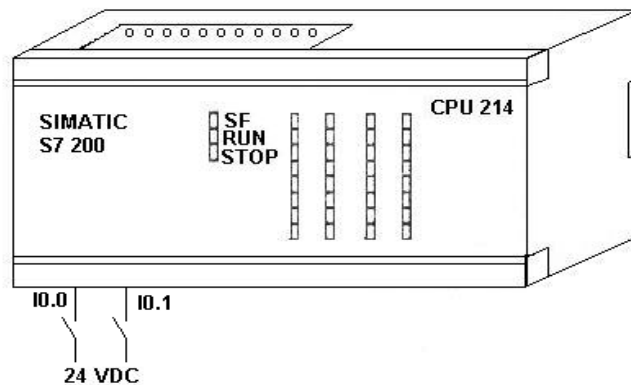


Figura 33. Esquema de conexión Interrupción Temporizada.

1) Descripción del Programa

El tiempo de ciclo de la Interrupción temporizada numero 10 (primera interrupción temporizada) se escribe en un byte de memoria especial SMB34. Y el ciclo para la interrupción temporizada número 11 (segunda interrupción temporizada) en un byte de memoria especial SMB 35.

En ambos casos, la base de tiempo es fijada en incrementos de 1 ms. El valor más pequeño permisible para la base de tiempo es de 5 ms. El valor más grande permisible para la base de tiempo es 255 ms.

El programa consiste en las siguientes rutinas:

Principal	Iniciando y especificando el tiempo
INT 0	Poniendo la salida Q 0.0
INT 1	Restableciendo la salida Q 0.0

El tamaño del programa consta de 51 palabras.

2) Listado del Programa en Formato AWL.

```
//INTERRUPCIÓN TEMPORIZADA
```

```
NETWORK //El arranque de la base de tiempo es especificado,  
//las dos Interr.de tiempo controlado son combinadas.
```

```
LD SM0.1  
MOVB 50, SMB34 //Especificando la base de tiempo para: interr.Temp 0:50ms  
MOVB 100, SMB35 //Especificando la base de tiempo para: interr.Temp 1:100ms  
ATCH 0, 10 //Combina Interr. de evento 10 con rutina de Interr. 0  
ATCH 1, 11 //Combina Interr. de evento 11 con rutina de Interr. 1  
ENI //Habilita todas las interr.
```

```
NETWORK //Cuando la entrada I0.1 tiene un flanco positivo, las bases de tiempo de las Interr.  
son duplicadas. Para llevar a cabo ésta nueva instrucción, las conexiones entre la interr. de eventos y la  
interr. de rutinas deberan ser separadas, porque los nuevos valores no serian aceptados de otra manera.  
Separando la conexion toma lugar con el comando DTCH. Despues de asignar las nuevas bases de  
tiempo las conexiones deberan ser restablecidas con un comando ATCH.
```

```
LD I0.1 //Carga la entrada I0.1  
EU //... y flanco positivo I0.1  
DTCH 10 //Separa la conexion de la interr. 0  
DTCH 11 //Separa la conexion de la interr. 1  
MOVB 100, SMB34 //Especificando la nueva base de tiempo para la interr. temporizada  
0:100ms  
MOVB 200, SMB35 //Especificando la nueva base de tiempo para la interr. temporizada  
1:200ms  
ATCH 0, 10 //Estableciendo conexion de nuevo  
ATCH 1, 11 //Estableciendo conexion de nuevo
```

```
NETWORK //La frecuencia del reloj anterior es establecida de nuevo por un flanco positivo en la  
entrada I0.0
```

```
LD I0.0 //Flanco positivo  
EU //... para la entrada I0.0  
DTCH 10 //Separa la conexion de la interr.temp 10  
DTCH 11 //Separa la conexion de la interr.temp 11  
MOVB 50, SMB34 //Inetervalo de tiempo para interr. temporizada 0:50ms  
MOVB 100, SMB35 //Inetervalo de tiempo para interr. temporizada 1:100ms  
ATCH 0, 10 //Habilitando la Interr. temp. 0  
ATCH 1, 11 //Habilitando la Interr. temp. 1
```

```
NETWORK  
MEND //Fin del programa principal
```

```

NETWORK          //La salida Q0.0 es puesta cuando la Interr. De la rutina 0 es llamada

INT  0           //Interrupcion de rutina 0

NETWORK
LD   SM0.0      //Siempre en 1
S   Q0.0, 1     //Poniendo la salida Q0.0

NETWORK
RETI           //Fin de la rutina de Interr. 0

NETWORK          //La salida Q0.0 es restablecida cuando la Interr. de la rutina 1 es llamada

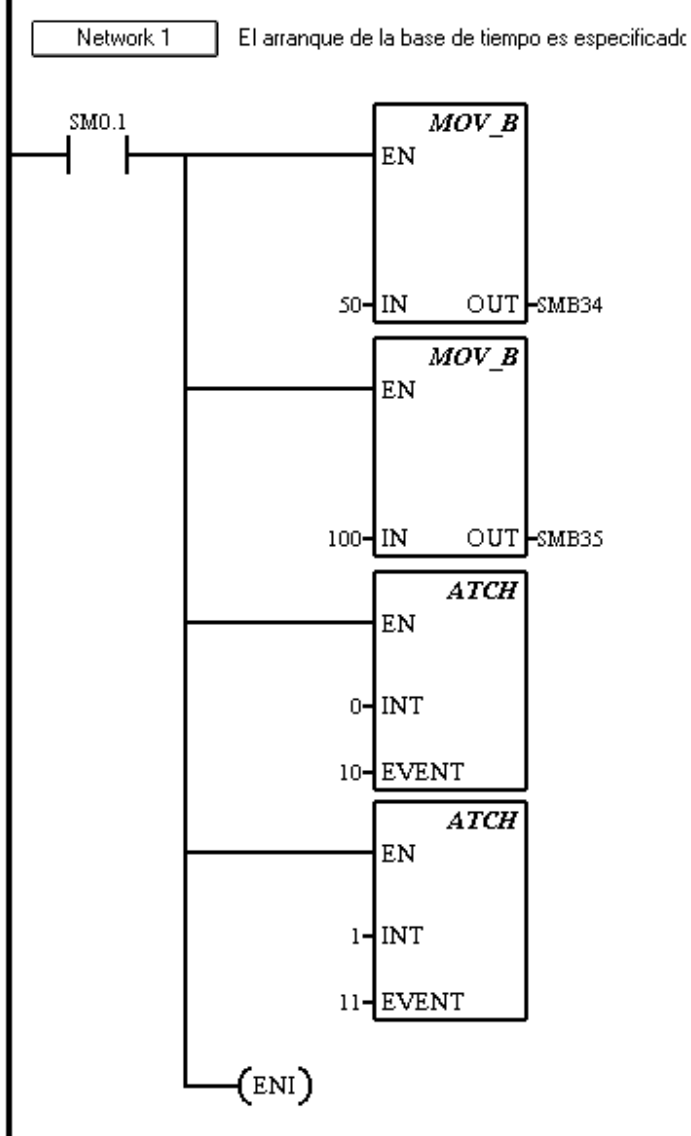
INT  1           //Interrupcion de rutina 1

NETWORK
LD   SM0.0      //Siempre en 1
R   Q0.0, 1     //Restableciendo la salida Q0.0

NETWORK
RETI           //Fin de rutina de Interr. 1

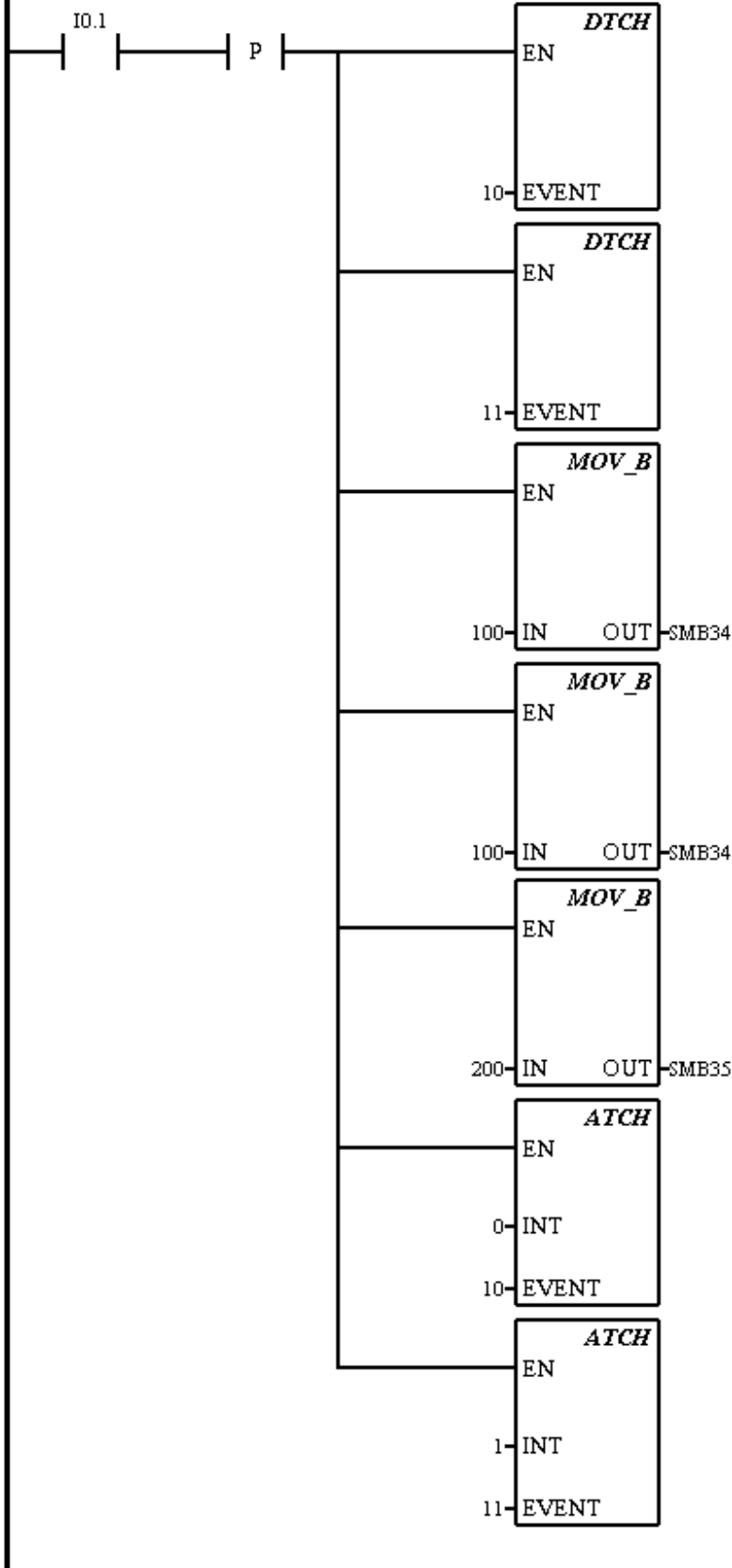
```

3) Listado del Programa en Formato KOP.



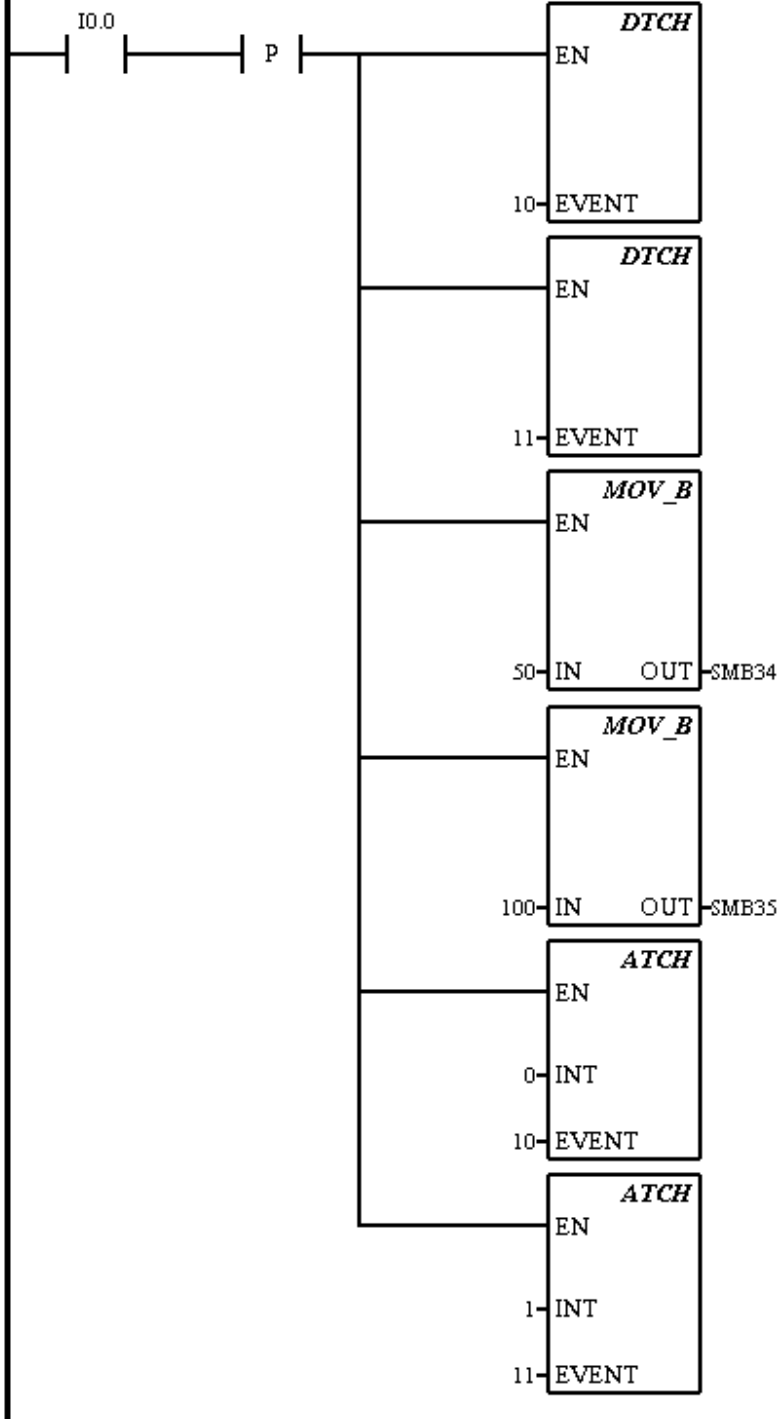
Network 2

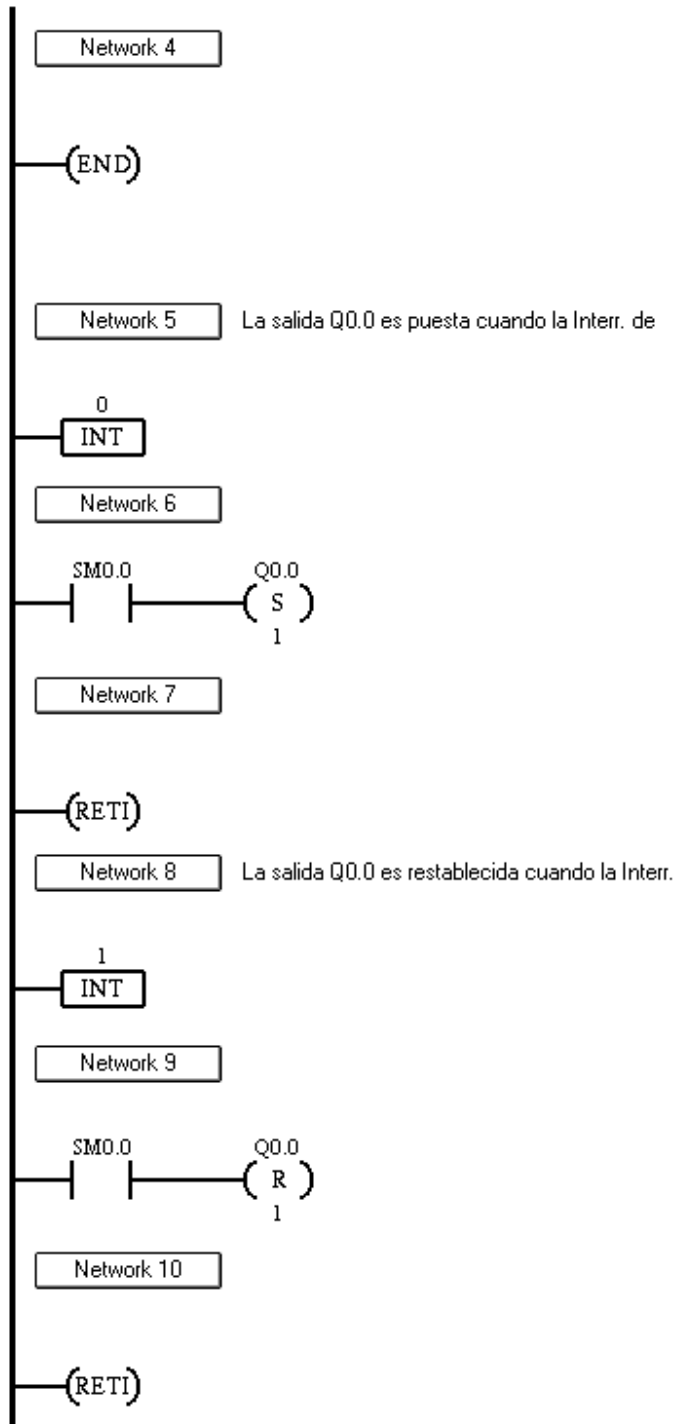
Cuando la entrada I0.1 tiene un flanco positivo,



Network 3

La frecuencia del reloj anterior es establecida de





4.3.4 Programación de un Secuenciador. Este programa realiza un ejemplo de una secuencia de pasos, por la cual cada paso es seguido por ciertas acciones. Los pasos suceden uno después del otro, por lo cual un paso es llevado fuera cuando todas las condiciones estipuladas para él han sido reunidas. La Figura 34. muestra el esquema de conexión al PLC. En las siguientes aplicaciones:

Precondiciones		Activación de Salidas
1 ^{er} . Paso	I 0.1 esta activada	Q 0.2 – Q 0.3
2 ^o . Paso	Corre Tiempo de 5 Seg. (Tempo. T37)	Q 0.1 – Q 0.4
3 ^{er} . Paso	Corre Tiempo de 5 Seg. (Tempo. T38)	Q 0.0 – Q 0.5
4 ^o . Paso	I 0.2 esta activada	Q 0.1 – Q 0.3 – Q 0.5
5 ^o . Paso	Corre Tiempo de 5 Seg. (Tempo. T 39) y I. 0.3 esta activada	Q 0.3
Restablecer secuencia de paso (I 0.0 esta activada)		Ninguna

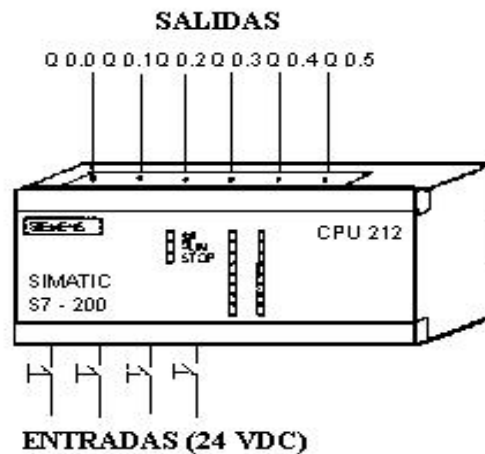


Figura 34. Esquema de conexión secuencia de pasos

1) Descripción del Programa.

El secuenciador consiste de cinco pasos los cuales se cumplen sucesivamente. Un paso consiste esencialmente en ajustar y restablecer ciertas salidas. Para que un paso se realice, desde luego tiene que reunir los prerequisites, tales como operar un interruptor, o el vencimiento del tiempo de espera. Es posible restablecer la secuencia de pasos en algún tiempo por medio de la operación del interruptor conectado en la entrada I 0.0.

2) Listado del Programa en Formato AWL.

```
NETWORK //1er. PASO
LD      I0.1           //Arrancando condicion
AN      I0.0           //y no restablecer
AN      M0.0           //y ningun paso esta ya activado
AN      M0.1
AN      M0.2
AN      M0.3
AN      M0.4
S       M0.0, 1       //Poner 1er. paso

NETWORK
LD      M0.0
S       Q0.2, 2       //Activar salidas
TON     T37, +50      //Arrancar intervalo de tiempo para el 2o
paso

NETWORK
LD      T37           //Despues de que el primer intervalo de
tiempo ha terminado
A       M0.0           //y tambien el primer paso
R       M0.0, 1       //Restablecer el 1er Paso
S       M0.1, 1       //y poner 2o Paso

NETWORK //2o. Paso
LD      M0.1
S       Q0.1, 1       //Activar Salidas
S       Q0.4, 1
```

```

R      Q0.2, 2      //Restablecer Salidas
TON    T38, +50    //Arranca cronometraje para el intervalo de
tiempo 3er Paso

NETWORK //3er. Paso
LD     T38          //Despues de que el 2o intervalo de tiempo ha
vencido
A      M0.1         //y despues de que el segundo paso fue
activado
R      M0.1, 1      //Restablecer 2o. Paso
S      M0.2, 1      //y activar 3er Paso

NETWORK
LD     M0.2         //3er Paso Activado
S      Q0.0, 1      //Activar salidas
S      Q0.5, 1
R      Q0.1, 1      //Restablecer salidas
R      Q0.4, 1

NETWORK //4o. Paso
LD     I0.2         //Despues de activar el prerequisite para el
4o paso
A      M0.2         //y despues del 3er Paso
R      M0.2, 1      //Restablecer el 3er. Paso
S      M0.3, 1      //y esta 4o PAso

NETWORK
LD     M0.3         //4o Paso
S      Q0.1, 1      //Activar salidas
S      Q0.3, 1
R      Q0.0, 1      //Restablecer salidas
TON    T39, +50    //Arranca cronometraje para el intervalo de
tiempo del 5o Paso

NETWORK //5o Paso
LD     I0.3         //Despues de que los prerequisites para el 5o
Paso se cumple
A      T39
A      M0.3         //y despues del 4o Paso
R      M0.3, 1      //Restablecer 4o Paso
S      M0.4, 1      //y poner el 5o Paso

NETWORK
LD     M0.4         //5o Paso
R      Q0.1, 1      //Restablecer salidas
R      Q0.5, 1

//Restablecer la secuencia de paso

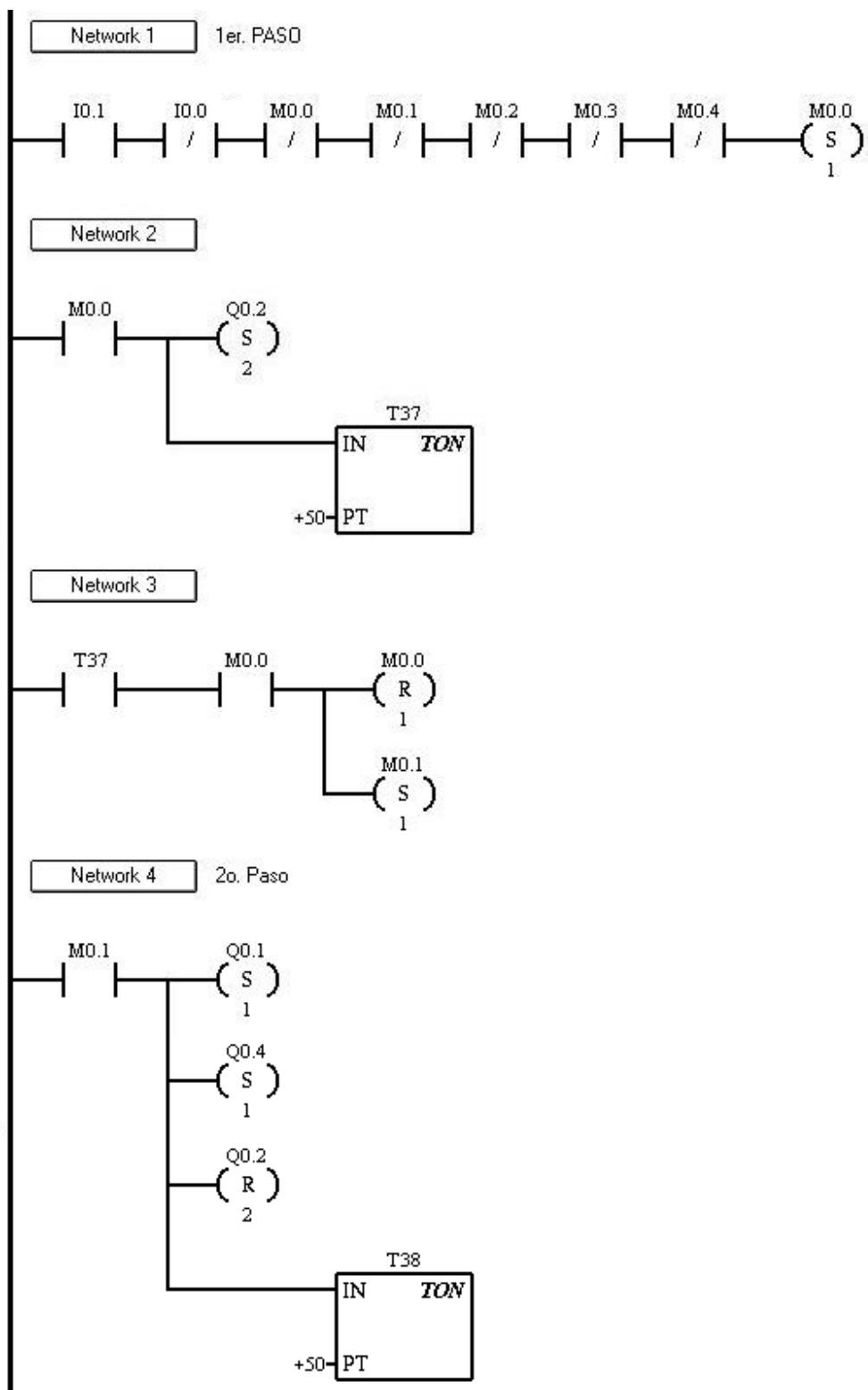
NETWORK
LD     I0.0         //Restablecer, después de que el interruptor
de la I0.0 ha sido activado

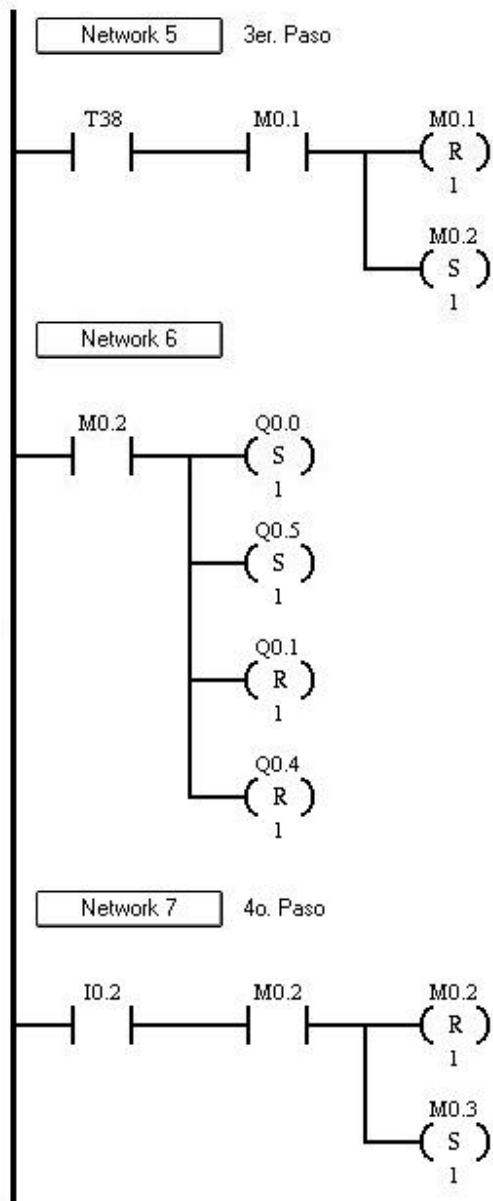
```

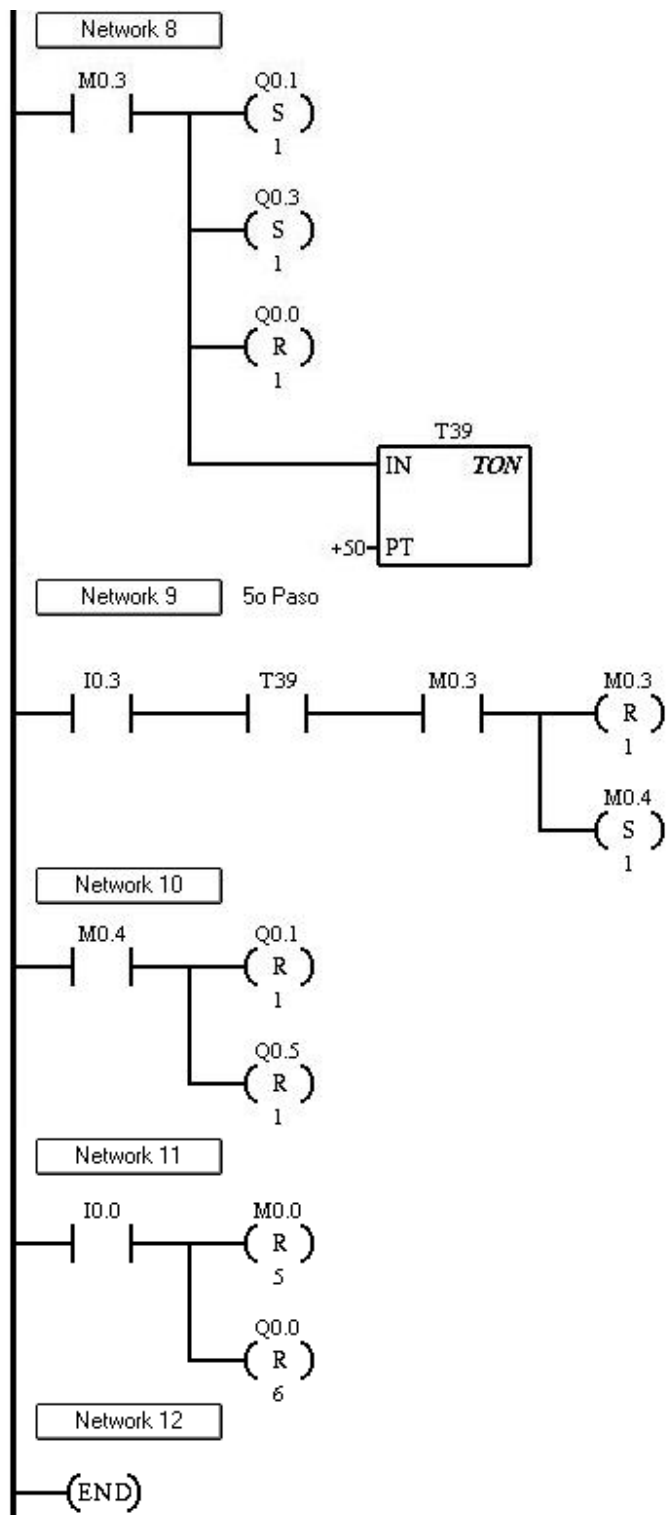
```
R      M0.0, 5      //Restablecer todas las banderas de los 5
pasos
R      Q0.0, 6      //Restablecer todas las Salidas

NETWORK
MEND      //FIN
```

3) Listado del Programa en Formato KOP







4.3.5 Control de Tiempo para la Iluminación de una Escalera. Este programa ejemplo sirve para encender la luz o luces de una escalera. Los botones pulsadores de encendido se encuentran en los diferentes pisos y todos son conectados en la entrada I0.0 del control. Después que un botón de encendido ha sido activado, el interruptor encenderá la luz por 30 segundos (Q0.0). sí durante éste tiempo otro botón de encendido es activado de nuevo, el intervalo de tiempo es reiniciado para empezar; esto asegura que la luz no durará más de 30 segundos después de que un botón fue operado. La Figura 35, muestra como se implementa este ejemplo en el PLC.

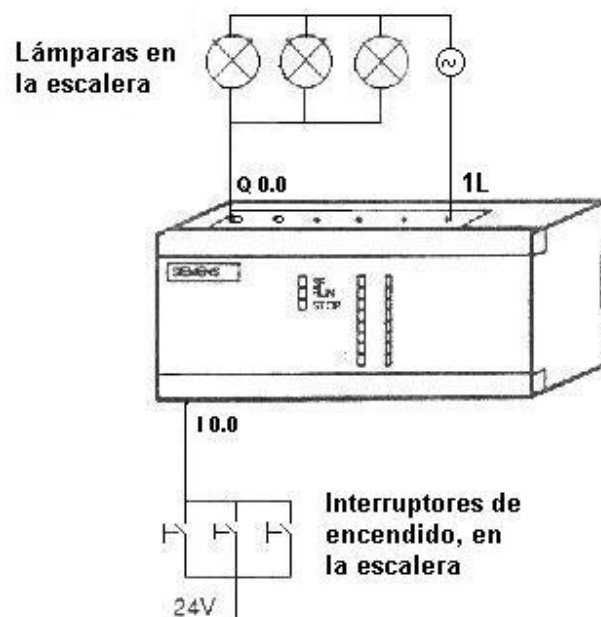


Figura 35. Esquema de conexión control luces en escalera.

1) Descripción del Programa

Si la señal de encendido fue aplicada en la entrada I0.0 (I0.0 = 1 Lógico), el bit del temporizador T37 es restablecido, así que el temporizador T37 empieza otra vez desde el arranque para cronometrar el tiempo. A la vez que la salida Q0.0 es activada. Después que han transcurrido 30 segundos de tiempo, el temporizador establecerá el bit del temporizador T37. Esto apagará la luz, (desactivará la salida Q0.0) de nuevo.

2) Listado del Programa en Formato AWL.

```
NETWORK //CRONOMETRAJE PARA ILUMINACION DE UNA ESCALERA

LD      I0.0           //Cuando un interruptor es operado
R       T37, 1        //Rearrancar el intervalo de tiempo
S       Q0.0, 1       //y encender la luz o luces

NETWORK

LD      I0.0           //Cuando el interruptor es operado
R       T37, 1        //Rearrancar el intervalo de tiempo
S       Q0.0, 1       //y encender luz o luces

NETWORK

LD      SM0.0         //Siempre en bit
TON     T37, +300     //Dejar cronome. del temp. a 30Seg.

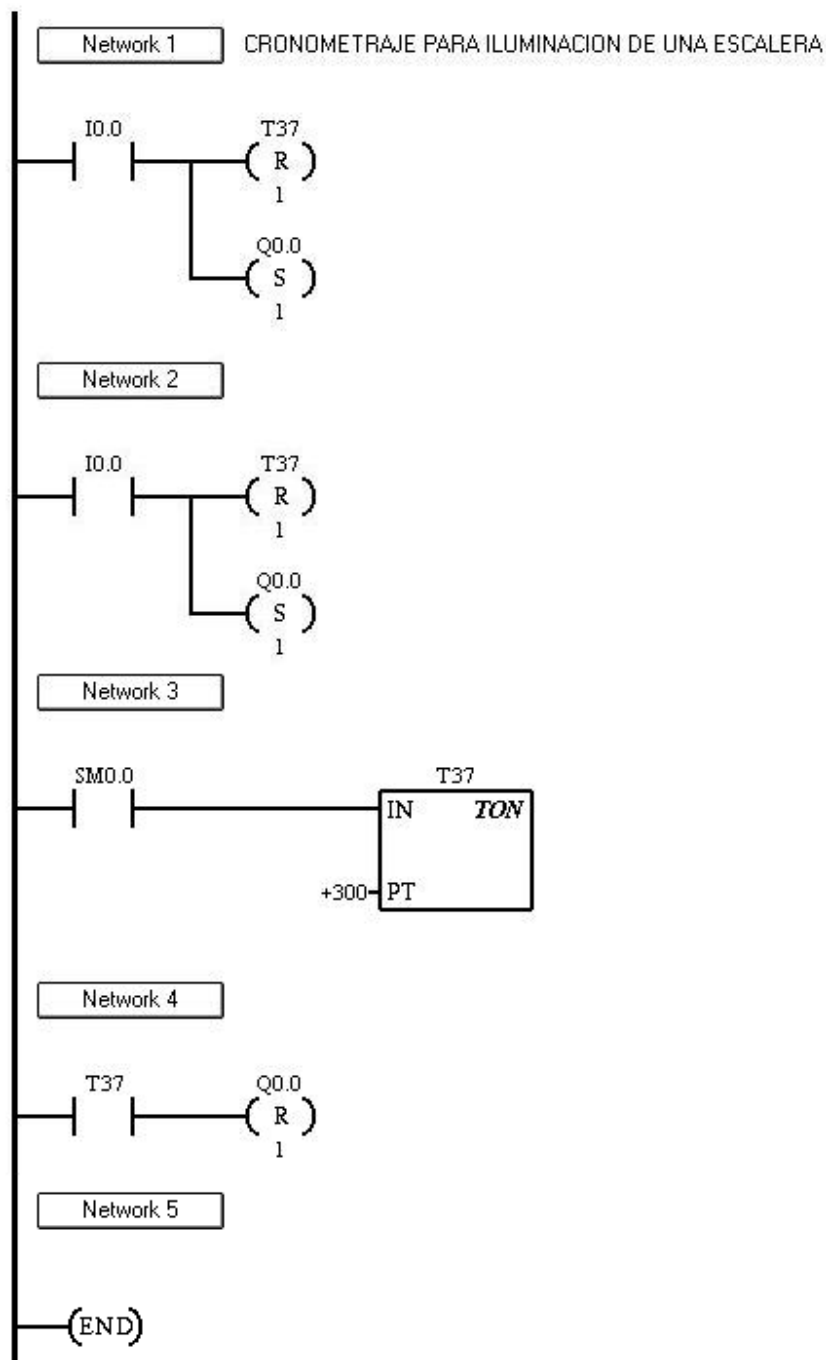
NETWORK

LD      T37           //Despues que ha terminado el tiempo,
R       Q0.0, 1       //apagar la luz

NETWORK

MEND                                //Fin
```

3) Listado del Programa en Formato KOP.



5. PRACTICAS

5.1 SISTEMA DE LLENADO Y VACIADO DE TANQUES

Implementar un programa que permita accionar la bomba (Se debe tener presente que la bomba no arrancará si el tanque sumidero TK4 esta sin agua, debido al Switch de nivel que esta en serie con ésta) y las electroválvulas VS1 y VS2 para llenar los tanques Tk1 y Tk2. En cada tanque hay switches de alto y bajo nivel, HLS y LLS respectivamente. Los de alto nivel son normalmente cerrado y los de bajo nivel normalmente abiertos.

El agua contenida en TK1 Y TK2 es enviada por gravedad a TK3 al abrir las electroválvulas VS3 y VS4, por ultimo se abrirá VS5 para Culminar el ciclo vertiendo el liquido en el tanque Tk4. Se deben tener en cuenta los switches de nivel para conformar un sistema de control TODO/NADA, y la protección a la bomba para que esta no arranque estando vacío el tanque Sumidero. Tk4.

El sistema se Arranca y detiene directamente por el START – STOP ubicado en la bornera de control.

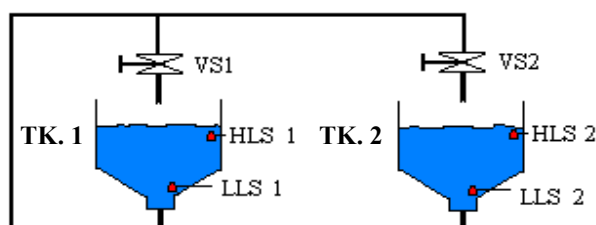


Figura No. 2. Esquema del sistema de llenado y vaciado de tanques

Se pide:

- a) Programa en Formato KOP o AWL
- b) Esquema de Conexión al PLC

5.2 ARRANQUE ESTRELLA – TRIANGULO E INVERSIÓN DEL SENTIDO DE GIRO DE UN MOTOR TRIFASICO DE INDUCCIÓN.

1) ARRANCADOR ESTRELLA – TRIANGULO CON CONFIRMACION DE FALLA.

Diseñar un programa para el arranque estrella – triángulo para un motor trifásico de inducción, El motor debe arrancar después que el pulsador de encendido es activado, con esto el motor debe arrancar en su primera etapa de estrella.

El motor debe ser conmutado en triángulo después de transcurridos unos 5 segundos. Si los pulsadores de apagado, o el cortacircuitos del motor son activados, el motor se debe detener. Si se activan los pulsadores Arranque – Parada al mismo tiempo, el motor no debe trabajar.

Si hay alguna falla en la etapa de la estrella, por ejemplo, el arrancador reconocerá o confirmará ésta falla. Después de 5 segundos del tiempo de demora, el PLC no debe hacer la conmutación para la etapa de triángulo evitando así un posible daño.

La posible falla se mostrara a través de una luz (bombilla.).

F1: Fusible Principal

F2: Relé Térmico

K1: Contactor de Línea

K2: Contactor Estrella

K3: Contactor Triángulo

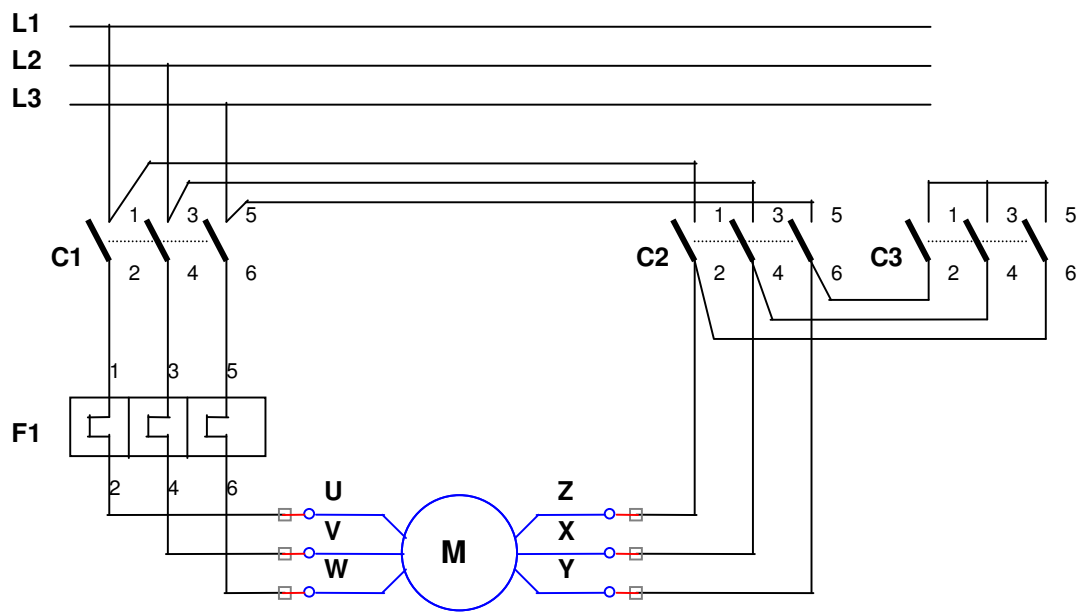


Figura 36. Esquema Eléctrico de Potencia del Arrancador Estrella – Triángulo.

Se pide:

- Programa en Formato KOP o AWL
- Esquema de Conexión al PLC

2) INVERSOR DE SENTIDO DE GIRO PARA UN MOTOR TRIFÁSICO DE INDUCCIÓN.

Diseñar un programa para el control de un motor trifásico de inducción con dos posibilidades en el sentido de giro. Un pulsador arrancará el motor y éste girará en sentido Horario, mientras que otro pulsador hará que gire en sentido antihorario. Los prerequisites son que el pulsador de protección de entrada y el de parada no estén activados (Abiertos). La conmutación de los pulsadores para hacer el cambio de rotación no se podrá hacer sino hasta después de que el pulsador de parada es presionado y un tiempo de 5 segundos ha transcurrido.

De esta manera el motor se puede detener y puede arrancar en sentido opuesto, si es necesario. Si ambos pulsadores son presionados al mismo tiempo, el motor se debe detener y no arrancar.

F1: Fusible Principal

F2: Relé Térmico

K1: Contactador Derecha

K2: Contactador Izquierda

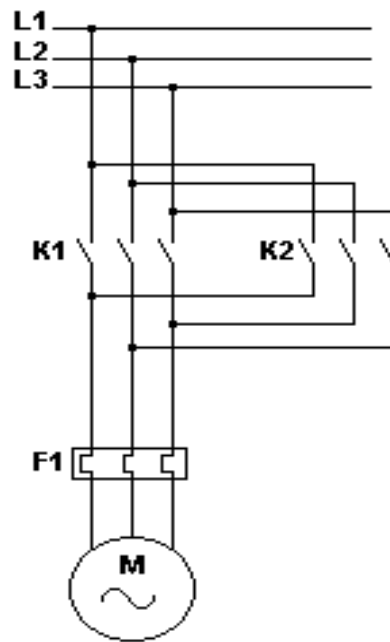


Figura 37. Esquema Eléctrico de Potencia de Inversión del Sentido de Giro de un motor trifásico.

Se pide:

- a) Programa en Formato KOP o AWL
- b) Esquema de Conexión al PLC

5.3 SISTEMA DE DOS BANDAS TRANSPORTADORAS BANDA_A Y BANDA_B.

Deben cumplir con el siguiente ciclo de trabajo:

Al activar el pulsador START comenzará a funcionar la Banda A, que transporta piezas sobre ella hasta el comienzo de la resbaladera. Al llegar a este punto, las piezas caen por gravedad por dicha resbaladera y al pasar por el sensor luminoso f1 lo activan, produciendo la parada de la Banda A, 5 segundos después y la puesta en marcha de la Banda B. Las piezas caídas en la Banda B se desplazan por ella hasta llegar al final de esta, donde esta ubicado otro censor luminoso f2. Al pasar las piezas por el sensor f2 se para la Banda B y se termina el ciclo de trabajo.

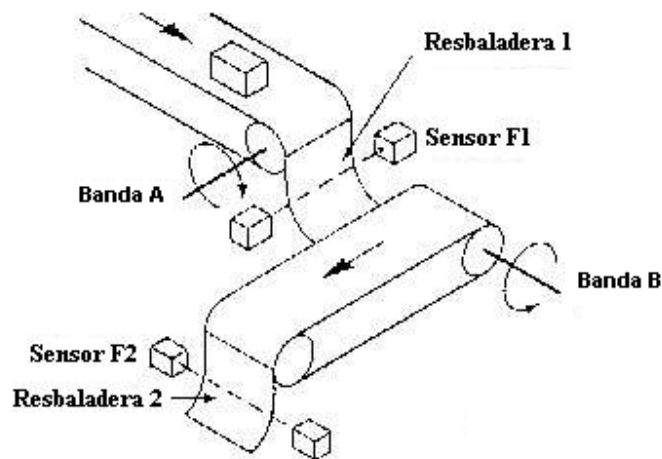


Figura. 38. Esquema Bandas Transportadoras

Se pide:

- Programa en Formato KOP o AWL
- Esquema de Conexión al PLC

5.4 SEMÁFORO PARA VEHICULOS Y PEATONES EN EL CRUCE DE DOS CALLES

Diseñar un programa para el control de semáforos para vehículos y peatones en el cruce de dos calles.

La puesta en funcionamiento se realiza por un conmutador de cuatro posiciones cuyo diagrama se indica más abajo. Ver la Figura 39.

Posición 0: Conexión de todos los semáforos ámbar en intermitente de forma permanente

Posición 1: Conexión de todos los semáforos ámbar en intermitente durante 5 segundos y seguidamente conexión normal con preferencia (Mayor tiempo en verde) para la calle A.

Posición 2: Conexión de todos los semáforos ámbar en intermitente durante 5 segundos y seguidamente conexión normal sin preferencia

Posición 3: Conexión de todos los semáforos ámbar en intermitente durante 5 segundos y seguidamente conexión normal con preferencia (Mayor tiempo en verde) para la calle B.

ENTRADA

FUNCIÓN

1	Contacto del Conmutador Preferencia Calle A
2	Contacto del Conmutador Sin Preferencia
3	Contacto del Conmutador Preferencia Calle B

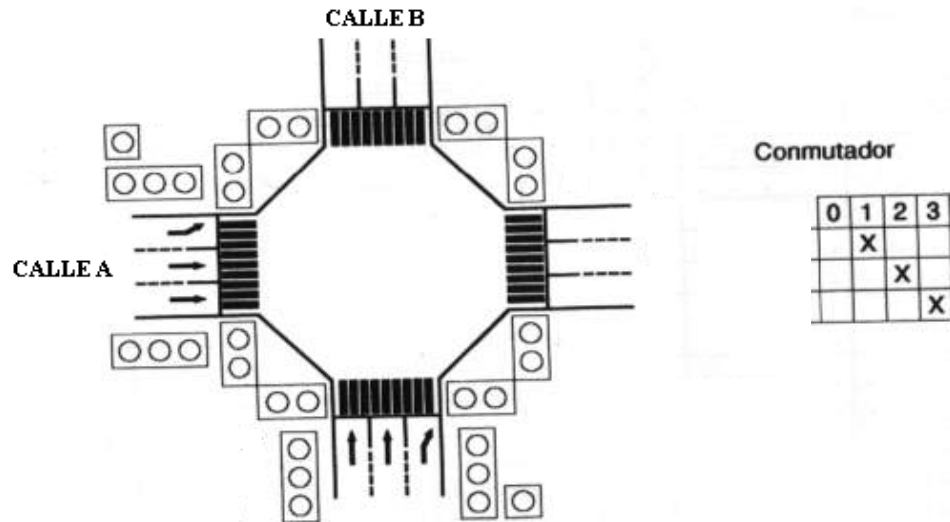


Figura 39. Señalización de las vías y disposición del conmutador para el control de semáforo.

SALIDA	FUNCIÓN
1	Contactador Lámparas: Semáforo Vehículos Rojo Calle B / Verde Calle A Semáforo peatones Verde Calle B / Rojo Calle A
2	Contactador Lámparas: Semáforo vehículo Amarillo Calle B
3	Contactador Lámparas: Semáforo vehículos Verde Calle B / Rojo Calle A Semáforo peatones Rojo Calle B / Verde Calle A
4	Contactador Lámparas: Semáforo Giro vehículos amarillo de Calle B a Calle A
5	Contactador Lámparas: Semáforo vehículo amarillo Calle A
6	Contactador Lámparas: Semáforo giro vehículos amarillo Calle A a Calle B.

Se pide:

- Programa en Formato KOP o AWL
- Esquema de Conexión al PLC

CONCLUSIONES

Al terminar con éxito las pruebas en los cuatro (4) módulos se cumple con el objetivo general del proyecto. Por ello podemos concluir que enhorabuena las facultades de Ingeniería Eléctrica y Electrónica adquieren una herramienta más para el proceso de aprendizaje en las áreas de Automatización y Control.

El Laboratorio de PLC garantiza la realización de prácticas a escala industrial, gracias a las características de desempeño del PLC Simatic S7 – 200 de SIEMENS, el cual además de trabajar con los módulos de este proyecto, se puede utilizar en el desarrollo de pruebas y practicas con los otros sistemas de control existentes en el laboratorio de Automatización y Control de la CUTB.

Al trabajar con el PLC SIMATIC7 – 200 CPU 214 de SIEMENS, el estudiante puede diseñar controles sin necesidad de utilizar interfaces de potencia adicionales, debido a que el Simatic S7-200 CPU214 posee salidas a Relé, es decir podemos ejercer control directo sobre equipos o elementos que se alimente en un rango de voltaje comprendido entre 5 – 30 VDC y de 120 – 230 VAC.

Mediante el desarrollo de las practicas el estudiante adquiere la habilidad para programar PLC's, habilidad que en un futuro le permitirá implementar programas de control a escala industrial.

BIBLIOGRAFÍA.

- CATÁLOGO TELEMECANIQUE. Octubre de 1993. Grupo Schneider
- CATALOGO GRINGER (www.grainger.com)
- INSTITUTO COLOMBIANO DE NORMAS TÉCNICAS Y CERTIFICACIÓN. Tesis y otros trabajos de grado. Santa fe de Bogotá: ICONTEC. 1996. NTC 1486
-
- LLADONOSA, Vicent. Programación de Autómatas Industriales Omron. 1ª Edición 1995. Barcelona España. Editorial Marcombo.
-
- MANUAL SIEMENS S5 - 95U
-
- MANUAL SIEMENS SIMATIC S7 – 200 (www.siemens.com)
- PORRAS C, Alejandro. Autómatas Programables. Fundamento, manejo, instalación y practica: McGraw - Hill / Interamericana de España S., 1990.
- SIMON, André. Autómatas Programables. Programación, automatismos y lógica programada. Paraninfo S.A. 1998.