

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE INGENIERÍA

DISEÑO Y CONSTRUCCIÓN DE UN CONTROLADOR LÓGICO PROGRAMABLE

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
EN ELECTRÓNICA Y CONTROL**

**LASCANO TOBAR CARLOS EDUARDO
VALLEJO PROAÑO FABIAN IGNACIO**

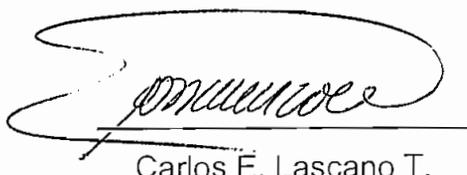
DIRECTORA: ING. ANA RODAS

Quito, Diciembre 2001

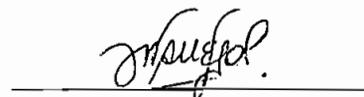
DECLARACIÓN

Nosotros, Carlos Eduardo Lascano Tobar y Fabián Ignacio Vallejo Proaño declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido en la ley de propiedad intelectual, por su Reglamento y por la normatividad institucional vigente.



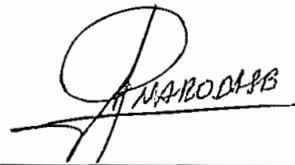
Carlos E. Lascano T.



Fabián I. Vallejo P.

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Carlos Eduardo Lascano Tobar y Fabián Ignacio Vallejo Proaño, bajo mi supervisión.

A handwritten signature in black ink, appearing to read 'ANARODAS', written over a horizontal line.

Ingeniera Ana Rodas
DIRECTORA DEL PROYECTO

AGRADECIMIENTO

Nuestros más sinceros agradecimientos a todas las personas que nos prestaron su ayuda desinteresada para la realización de este trabajo y de manera especial:

- A la Ing. Ana Rodas por su acertada dirección.
- Al Ing. Nelson Díaz quien nos encaminó en el proceso.
- A nuestros padres que gracias a su apoyo hemos culminado nuestros estudios.

INDICE

CAPÍTULO 1 PRESENTACIÓN

1.1	Antecedentes Históricos	1
1.2	Definición de un Controlador Lógico Programable	2
1.3	El PLC como elemento de automatización	2
1.4	Ventajas de la automatización con PLC	4
1.5	Desventajas de la automatización	5
1.6	Estructura Básica de un PLC	5
1.6.1	Unidad Central de Proceso CPU	6
1.6.2	Memorias	6
1.6.3	Periferia	7
1.7	Elementos de Programación de un Controlador Lógico Programable	7
1.7.1	Entradas	8
1.7.2	Salidas	8
1.7.3	Relés Internos	8
1.7.4	Relés Temporizados On Delay	9
1.7.5	Relés Temporizados Off Delay	10
1.7.6	Relés de Pulsos	10
1.7.7	Relés Contadores	11
1.8	Algunos PLC existentes en el mercado	12
1.8.1	PLC Siemens	12
1.8.2	PLC Omrom	13
1.8.3	PLC Cuttler Hammer	14
1.9	Programación de PLC	16
1.9.1	Lenguajes de Bajo Nivel	16
1.9.2	Lenguajes de Alto Nivel	18
1.10	Comunicación con otros dispositivos	19

CAPÍTULO 2 CARACTERÍSTICAS DEL CONTROLADOR LÓGICO PROGRAMABLE A DISEÑAR

2.1	Consideraciones Generales	25
2.2	Consideraciones de Hardware	25
2.2.1	Unidad Central de Proceso	26
2.2.2	Entradas Físicas	27
2.2.3	Salidas	28
2.2.4	Fuente	29
2.3	Configuración de Software	30
2.3.1	Software para el Microcontrolador	30
2.3.2	Software para Programación y Comunicación	31
2.3.2.1	Software para Programación	32
2.3.2.2	Software para Comunicación	32

CAPÍTULO 3 DISEÑO DEL HARDWARE

3.1	Diseño de la Etapa de Control	33
3.1.1	Unidad Central de Proceso	34
3.1.1.1	Elección del Microprocesador	34
3.1.1.2	Hardware para el Microcontrolador MCS 51	35
3.1.2	Almacenamiento del Programa	35
3.1.3	Almacenamiento del Programa de Usuario	36
3.1.4	Comunicación	36
3.2	Diseño de la Etapa de Entradas del PLC	37
3.2.1	Descripción del Funcionamiento	38
3.2.2	Cálculo de Resistencias de Limitación	39
3.3	Diseño de la Etapa de Salidas del PLC	41
3.3.1	Descripción del Funcionamiento	41
3.4	Diseño de Fuente	42
3.4.1	Batería de Respaldo	46

CAPÍTULO 4 DISEÑO DEL SOFTWARE

4.1	Diseño del Software para el Microcontrolador MCS 51	48
4.1.1	Descripción del Software para Simulación y Depuración PINNACLE 52	48
4.1.2	Configuración de la memorias del Microcontrolador MCS 51	49
4.1.2.1	Distribución de la memoria RAM Interna del Microcontrolador MCS 51	50
4.1.2.2	Memoria EPROM Interna del Microcontrolador MCS 8751	52
4.1.2.3	Memoria EPROM Externa del Microcontrolador MCS 8751	52
4.1.2.4	Memoria RAM Externa al Microcontrolador MCS 8751	52
4.1.3	Programa para el Microcontrolador de la Serie MCS 51	53
4.1.4	Subrutina de Administración del PLC	55
4.1.4.1	Descripción del Funcionamiento	55
4.1.5	Subrutina de Tratamiento de Datos	78
4.1.5.1	Código de envío de Datos INTEL Estándar	78
4.1.5.2	Pórtico de Comunicaciones	80
4.1.6	Software para Pruebas	85
4.2	Diseño del Software para PC	88
4.2.1	Programa en Visual Basic	89
4.2.1.1	Ventana de Diseño	89
4.2.1.2	Ventana de Información	104
4.2.1.3	Ventana de Ensamblado	106
4.2.1.4	Ventana de Introducción de Datos	107
4.2.2	Subrutinas del Programa	109
4.2.3	Procedimientos para el Diseño	121

CAPÍTULO 5 ANALISIS ECONÓMICO

5.1	Descripción de los elementos usados	126
5.1.1	Elementos utilizados en Tarjeta de Control Principal	126
5.1.2	Elementos utilizados en Etapas de Entradas	127
5.1.3	Elementos utilizados en Etapas de Salidas	127
5.1.4	Elementos utilizados en Tarjeta de Fuente	128
5.1.5	Elementos generales utilizados	128
5.2	Desarrollo de Software	129
5.3	Valor de mercado del PLC	130

CAPÍTULO 6 PRUEBAS Y RESULTADOS

6.1	Enclavamiento Eléctrico	134
6.2	Relé de Pulsos	134
6.3	Semáforo para Control Vehicular	135
6.4	Control de Nivel de un Tanque de Almacenamiento de Líquidos	138
6.5	Arranque Estrella Triángulo	140

CAPÍTULO 7 CONCLUSIONES Y RECOMENDACIONES

7.1	Conclusiones	141
7.2	Recomendaciones	143

BIBLIOGRAFÍA	144
---------------------	------------

ANEXOS

Fotografías	A.1
Diagramas Esquemáticos	A.2
Ruteados de Tarjetas	A.3

PRESENTACIÓN

Se ha observado que en la pequeña y mediana industria ecuatoriana muchos procesos se realizan sin automatización, por lo que no siempre se logra resultados satisfactorios que a la postre aumentan los costos de producción.

Muchos industriales no automatizan sus procesos debido a que los equipos electrónicos para este fin a menudo están sobredimensionados por lo que se compra mucho más de lo que se necesita.

Por otro lado, las pequeñas y medianas industrias no cuentan con personal calificado para la programación de estos equipos. Generalmente es necesario contratar personal para que realice la programación y el mantenimiento de estos equipos y los industriales no quieren mantener dependencia para realizar estos trabajos posteriormente.

Por esto se ha creído conveniente presentar a la industria ecuatoriana un equipo que sea capaz de automatizar pequeños procesos con elementos electrónicos comunes en nuestro mercado y sobre todo que cualquier persona con conocimientos básicos de sistemas de control industrial pueda programar.

Es un primer paso en procesos de automatización industrial que a la postre puede lograr resultados económicos satisfactorios y procesos de automatización más completos de fabricación ecuatoriana.

La tesis pretende introducir en el mercado un Controlador Lógico Programable con funciones básicas de control, de programación sencilla, amigable y económico, para solucionar problemas pequeños de automatización.

RESUMEN

En este proyecto de tesis se presenta el Diseño y la Construcción de un Controlador Lógico Programable, PLC.

En primera instancia se define un PLC con las funciones básicas de control que éste tiene y las características más relevantes necesarias para desarrollar pequeñas aplicaciones.

Luego se presenta el diseño del hardware de la manera más simple y completo posible, para que permita realizar ampliaciones en trabajos posteriores. Se ha diseñado un equipo modular a fin de poder darle un matiz didáctico y al mismo tiempo permitir reemplazar algunos elementos sin problema. Para este fin se ha utilizado únicamente elementos que se pueden encontrar en el mercado local.

Para programar el microcontrolador se utilizó PINNACLE 52, un programa que permite realizar aplicaciones en Assembler para microcontroladores de la serie MCS 51 y MCS 52, amigable si lo comparamos con versiones anteriores de esta serie de programas.

El desarrollo del software para PC fue implementado en Visual Basic 5.0. que es una herramienta de programación poderosa sobre todo cuando se requiere programar eventos gráficos. Este software para PC se lo ha elaborado de tal manera que pueda ser flexible y entendible para el usuario al establecer el diagrama de escalera.

Se han colocado funciones de control básicas y simples a través de interfases gráficas; de manera de simular a los elementos electromecánicos de antaño, con todos sus parámetros presentados sólo en una pantalla.

Para completar el estudio se incluye un análisis económico que servirá como referencia para que el lector tenga bases comparativas con el mercado local para la toma de decisiones.

CAPÍTULO 1.

PRESENTACIÓN

1.1 ANTECEDENTES HISTÓRICOS

La historia de los Controladores Lógicos Programables (Programmable Logic Controller) mejor conocidos como PLC no es extensa, se inicia en 1968 en la General Motors Corporation; a raíz de que se intensificó la producción en masa para la manufactura de automóviles, lo que involucra mucha maquinaria y sistemas de control. La duplicación exacta de una pieza originaba problemas puesto que los tableros de control que existían eran complejos, no exactos, con mucho cableado y difíciles de hacer mantenimiento y reparaciones. El cambio de características tomaba demasiado tiempo; se habla concretamente de los tableros con elementos electromecánicos (relés, temporizados, contadores, etc.).

En la Compañía General Motors existieron los suficientes incentivos para eliminar los complicados sistemas electromecánicos de control y encontrar un camino alternativo que los reemplace garantizando el funcionamiento y facilitando el mantenimiento; es así como el primer Controlador Lógico Programable fue introducido en la industria.

Las especificaciones de los PLCs llamaron la atención de varias empresas que manufacturaban sistemas de control; como resultado de esto aparece la primera generación de PLC que solamente tenían funciones básicas, las cuales con el pasar del tiempo se han ido incrementando hasta darle un carácter general y de uso casi universal.

Estos nuevos equipos ofrecen muchas ventajas industriales, más aún cuando son capaces de auto diagnosticarse, involucran el manejo de señales analógicas así como funciones de control tales como acciones PID y permiten tener un control centralizado por medio de una computadora personal.

1.2 DEFINICIÓN DE UN CONTROLADOR LÓGICO PROGRAMABLE

Un Controlador Lógico Programable es un dispositivo electrónico que puede ser programado para cumplir determinadas tareas de control en sistemas automáticos a nivel industrial. Básicamente es una unidad central de proceso que contiene un programa y se conecta a periféricos de entradas y salidas. Las funciones de un PLC son ejecutadas a través de los periféricos de entrada, que de acuerdo con la lógica del programa se transmitirán hacia las señales de salida, las cuales serán usadas para cumplir acciones de control programadas por el usuario.

Su nombre viene del inglés “Programmable Logic Controller” que significa “Controlador Lógico Programable” o PLC y como se ha indicado sustituyen a los antiguos sistemas de control electromecánicos, aumentan su capacidad con funciones conjuntas con el computador dándole una mayor flexibilidad y confiabilidad que la que se disponía antiguamente.

Como se puede observar de la definición, un PLC tiene una configuración similar a una computadora, la diferencia básica es la manera de comunicación con el medio externo; una computadora necesita de elementos auxiliares para que pueda controlar algo específico en el medio industrial, mientras que un PLC por su estructura es auto suficiente para realizar labores de control, además es capaz de soportar el duro trabajo industrial.

1.3 EL PLC COMO ELEMENTO DE AUTOMATIZACIÓN

Los primeros sistemas que se convirtieron en un dolor de cabeza para el personal de mantenimiento de una planta industrial fueron los tableros de mando de arranque de motores eléctricos con inversión de giro, pues para realizar un cambio de operación, se debían invertir demasiadas horas de trabajo; de ahí que las primeras actuaciones en el control fueran orientadas al reemplazo de relés,

temporizados y controles on/off auxiliares, de manera que las modificaciones se realicen económica y rápidamente.

Se transformaron en sistemas sencillos, económicos, modulares y con capacidad de ser readaptados sin necesidad de tener que cablear nuevamente, sino que estos cambios se realizaban por medio de un pequeño cambio en software.

Con lo anterior se puede decir que la automatización de un sistema básicamente está supeditada por dos aspectos: **económico y técnico**.

Técnicamente un sistema debe ser automatizado cuando:

- Se desea un sistema pequeño y sencillo de instalar.
- Que sea más versátil para realizar ampliaciones y/o cambios.
- Que preste facilidades para realizar mantenimiento.
- Que sea confiable.
- Que reduzca el número de fallas y por ende paros de producción.

Económicamente un sistema debe ser automatizado cuando:

- El número de elementos que se requieren supere el costo de un PLC (por experiencia se conoce que el número de elementos debe ser superior a cinco).
- El costo de paro de producción por efectos de algún daño en el tablero de control sea alto.

En definitiva, se hace completamente necesario pensar en la **AUTOMATIZACIÓN** cuando se desea **CONFIABILIDAD, EFICIENCIA Y ECONOMÍA** en un proceso de producción y el elemento ideal para ello es el PLC.

Los mismos puntos que indican cuando se debe automatizar dan la pauta para indicar como automatizar.

Para la automatización de un sistema se hace necesario:

- Que el sistema sea completamente conocido. Para ello es necesario que el personal encargado del cambio tenga pleno conocimiento del sistema que se desea modificar.
- Que el sistema pueda ser modificado y/o ampliado permanentemente, de así requerirlo.
- Que el estado de las salidas dependan de las entradas y un sistema de control que realice las correcciones.

En el mercado actual existen muchas marcas, modelos y tamaños de PLC los cuales están disponibles para diferentes aplicaciones; es función de las personas encargadas el encontrar la mejor opción para su aplicación.

1.4 VENTAJAS DE LA AUTOMATIZACIÓN CON PLC

Son innumerables las ventajas que ofrecen los sistemas de control automatizados, se mencionará algunas que involucran un PLC como elemento de control:

- Sistemas más confiables y eficientes.
- Menor número de fallas.
- Menor número de elementos interconectados.
- Menor cableado.
- Facilidad de mantenimiento.
- Menor espacio necesario para el montaje.
- Posibilidad del uso del PLC para distintas aplicaciones.
- Posibilidad de incremento de capacidad simplemente aumentando módulos adicionales.
- Suficientemente robusto para trabajar en ambientes industriales.
- No hay desgaste de elementos electromecánicos.

- Pueden realizar sistemas complicados de control como PID (de acuerdo al tipo de PLC).
- Diagnóstico de fallas centralizado cuando se requiere.

1.5 DESVENTAJAS DE LA AUTOMATIZACIÓN

- Requieren una buena regulación de voltaje.
- Cuando se daña en muchos casos es necesaria la reposición total del PLC.
- Para pequeñas aplicaciones es costoso.
- Es necesario personal capacitado para el mantenimiento y programación de los PLC.
- Los equipos que se han adquirido con PLC, se convierten en una caja negra la cual no puede ser analizada en caso de existir daños.

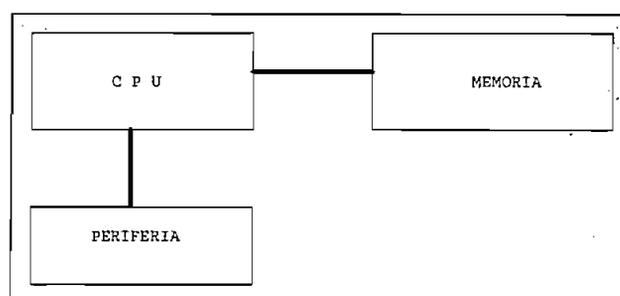
1.6 ESTRUCTURA BÁSICA DE UN PLC

En su entorno físico los PLC contiene tres componentes básicos:

- Unidad Central de Proceso CPU.
- Memoria de datos y de programa.
- Periféricos de entradas y salidas.

La **Figura 1.1** se indica la estructura básica de un PLC.

Figura 1.1 Estructura Básica de un PLC



1.6.1 UNIDAD CENTRAL DE PROCESO (CPU)

La Unidad Central de Proceso de un controlador lógico programable, es el centro físico y lógico donde se realizan todas las operaciones aritméticas y lógicas definidas por el fabricante. Así mismo, es aquí donde se almacenan las operaciones intermedias, las variables de proceso, las señales de entrada y salida, es decir aquí se organiza toda la información del PLC así como su distribución. También se encarga de establecer el interfaz de comunicación con el usuario que será el encargado de la programación de las funciones específicas de operación.

1.6.2 MEMORIAS

El PLC tiene dos tipos de memoria en su interior:

Memoria RAM

Es la memoria donde se guardan las aplicaciones que el usuario realizará a lo largo del tiempo; permite al usuario reprogramar tantas veces como sus aplicaciones lo requieran; en caso de necesitar para una única función específica esta memoria se puede reemplazar por una memoria tipo EPROM, (la mayoría de fabricantes de PLC permiten introducir estas modificaciones) ya que este tipo de memoria permite almacenar los datos por mayor tiempo, sin que estos se pierdan por un error de programación o por interrupción de energía.

Memoria ROM

Ya sea dentro del microprocesador o fuera de este, es la memoria en la cual se halla almacenado el programa de funcionamiento del PLC; aquí se encuentran todas las funciones lógicas y aplicaciones que ha proporcionado el fabricante; es aquí donde se define la estructura así como las funciones que tendrá el PLC.

1.6.3 PERIFERIA

La comunicación con los diferentes elementos de control tanto de entrada como de salida constituyen la periferia que permite la comunicación con el exterior.

Estas señales ya sean de entrada o de salida pueden ser analógicas o digitales en función a las aplicaciones que el proceso requiera o las que el equipo disponga.

La mayoría de PLC vienen con un banco básico de entradas y salidas: (generalmente digitales), adicionalmente los fabricantes proveen módulos de señales analógicas que son utilizables únicamente con pequeños cambios de hardware (seteo de interruptores).

1.7 ELEMENTOS DE PROGRAMACIÓN DE UN CONTROLADOR LÓGICO PROGRAMABLE

Son muchos los elementos que se pueden programar por el usuario en los PLC actuales, de hecho cada fabricante le ha dado su toque "personal" de manera que éstos cuenten con las mayores posibilidades de aplicaciones. Sin embargo todos parten de una base que viene a ser el control industrial de antaño. Los elementos de control básico son los siguientes:

- Entradas.
- Salidas.
- Relés Internos.
- Relés Temporizadores On Delay.
- Relés Temporizadores Off Delay.
- Relés de Pulsos.
- Relés Contadores.

1.7.1 ENTRADAS

Señales analógicas o digitales que provienen del medio externo conectadas directamente a los procesos que van a definir la lógica de funcionamiento junto con el programa disponible en el PLC.

Las señales de entrada pueden provenir de: un contacto ya sea abierto o cerrado de un relé de protección, de un sensor de nivel, de un sensor fotoeléctrico, un sensor de temperatura, un pulsante, un interruptor o cualquier elemento sensor de control en un proceso.

El PLC contiene en su interior elementos de aislamiento entre el sistema de control interno y las entradas físicas del proceso.

1.7.2 SALIDAS

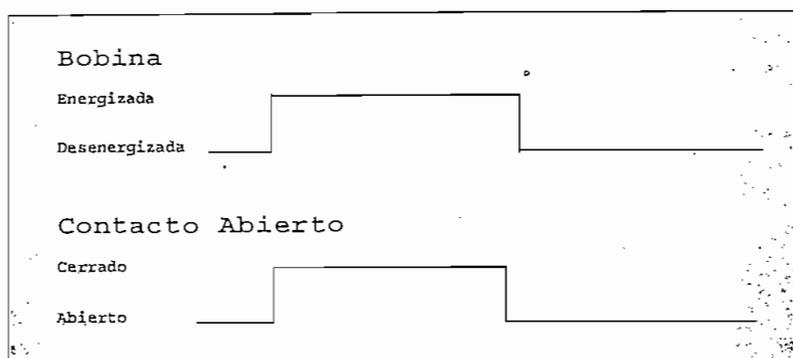
Son señales que luego de una lógica de control establecida por el programa en el PLC se enviarán al exterior para ejecutar acciones sobre elementos de mando, de la misma manera que las entradas, las salidas pueden ser analógicas o digitales.

Las señales de salida pueden controlar: un contactor, un elemento de estado sólido, una válvula, etc.

1.7.3 RELÉS INTERNOS

Son relés usados para realizar la lógica de control; al igual que los relés electromagnéticos son interruptores electrónicos, usados para activar o desactivar a un elemento de salida u otro elemento interno de un determinado proceso de control. Cambian el estado lógico de sus contactos al ser excitados eléctricamente. Los relés en sus contactos lógicos tienen sólo dos posibles estados: abiertos y cerrados.

Figura 1.2 Diagrama de Tiempos de Bobina y Contacto Abierto de Relé Interno

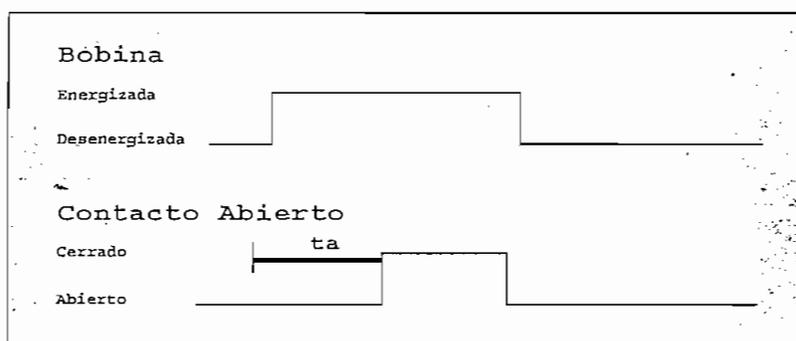


1.7.4 RELÉS TEMPORIZADORES ON DELAY

En diversas operaciones y procesos industriales se hace necesario retardar ciertas acciones de una duración definida. Los relés de tiempo On Delay emiten una señal de excitación un tiempo (programado por el usuario) después de excitado, es decir, son relés que tienen retardo a la conexión, si la desconexión se produce antes de que este haya operado, el ciclo empezará nuevamente.

La precisión de los retardos tiene una gran influencia sobre los productos obtenidos; por ejemplo en una soldadura de punto, la calidad de cada punto depende de la exacta duración del paso de corriente.

Figura 1.3 Diagrama de Tiempo de Bobina y Contacto Abierto de Temporizado On Delay

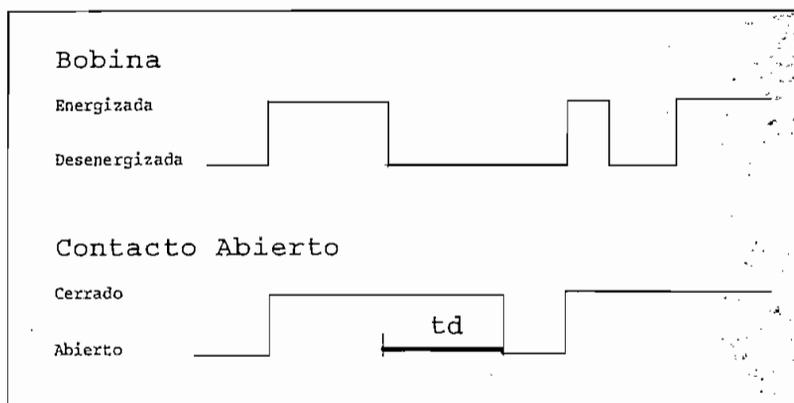


Donde: t_a representa el tiempo de retardo a la activación.

1.7.5 RELÉS TEMPORIZADORES OFF DELAY

Los relés de tiempo Off Delay emiten una señal de excitación un tiempo (programado por el usuario) después de eliminada la excitación, es decir, son relés que tienen retardo a la desconexión, si la conexión se produce antes que haya operado, el ciclo empezará nuevamente.

Figura 1.4 Diagrama de Tiempo de Bobina y Contacto Abierto de Temporizado Off Delay



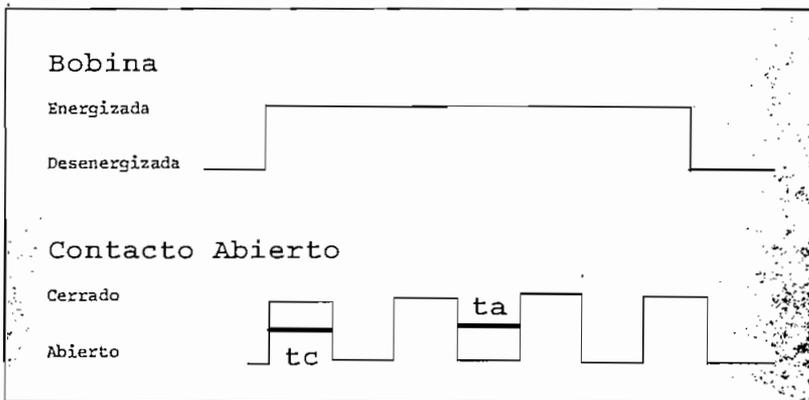
Donde: t_d representa el tiempo de retardo desde la desactivación.

Si el tiempo de excitación de la bobina es corto (menor que el tiempo de retardo programado), los contactos cambiarán su estado y comenzará el ciclo otra vez.

1.7.6 RELÉS DE PULSOS

Los relés de pulsos, una vez que sus bobinas han sido excitadas, emiten una señal intermitente en sus contactos. El tiempo de intermitencia es programado por el usuario. Generalmente los tiempos de apertura y cierre de sus contactos son los mismos, aunque se pueden programar diferentes tiempos para estos dos eventos.

Figura 1.5 Diagrama de Tiempo de Bobina y Contacto Abierto de Relé de Pulsos



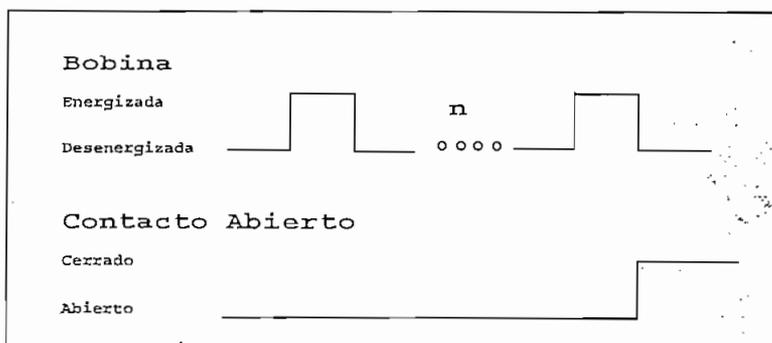
Donde: t_a representa el tiempo en estado cerrado.

t_c representa el tiempo en estado abierto.

1.7.7 RELÉS CONTADORES

Los relés contadores incrementan su cuenta cada vez que un evento excita su bobina, hasta llegar a la cuenta programada por el usuario en cuyo caso, cambia el estado de sus contactos lógicos de control, excitando así una señal de control de salida. Disponen de una señal de reset que actúa una vez que la cuenta ha finalizado, la que también puede ser activada por alguna señal externa previamente programada para empezar la cuenta nuevamente.

Figura 1.6 Diagrama de Tiempo de Bobina y Contacto Abierto de Relé Contador



Donde: n representa el número de pulsos (eventos) que se cuentan.

1.8 ALGUNOS PLC EXISTENTES EN EL MERCADO

Actualmente en el mercado casi todas las empresas que se dedican a la fabricación de equipo electrónico industrial y aún doméstico, tienen su versión de PLC; de ahí que resulte muy común encontrarse a nivel industrial con una gama muy amplia tanto en marcas como en aplicaciones de PLC, dependiendo principalmente de las preferencias de los usuarios de los equipos.

A continuación se presenta un pequeño resumen con algunos de los fabricantes más importantes de PLC existentes a nivel mundial

1.8.1 SIEMENS

Siemens lanzó al mercado la versión denominada SIMATIC, aunque en la actualidad está en vigencia la serie SIMATIC NET en cuya denominación se encuentra una gama muy amplia de modelos de acuerdo a las necesidades y las aplicaciones requeridas; se mencionan algunas de las características en la **Tabla 1.1**:

Tabla 1.1 PLCs SIEMENS: Características

Unidad Central	CPU 100	CPU 102	CPU 103
Funciones	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales
Procesador	1 estándar	1 estándar	1 estándar 1 coprocesador
Tiempo de ejecución para $1*2^{10}$ instrucciones	70 ms	7 ms	1.6 ms
Memoria de programa RAM Interna EPROM y EEPROM	$2*2^{10}$ bytes $2*2^{10}$ bytes	$4*2^{10}$ bytes $4*2^{10}$ bytes	$20*2^{10}$ bytes $20*2^{10}$ bytes
Temporizadores internos Margen de tiempo externos	16 10 ms a 9990 s con módulo externo	32 10 ms a 9990 s con módulo externo	128 10 ms a 9990 s con módulo externo
Contadores Internos Margen de Computo Externo	16 0 a 999 con módulo externo	32 0 a 999 con módulo externo	128 0 a 999 con módulo externo
Entradas Digitales Salidas Digitales	128 en total	256 en total	256 en total
Entradas Analógicas Salidas Analógicas	8 en total	16 en total	32 en total

1.8.2 OMROM

Omrom Electronics Inc. ofrece una variedad muy amplia de memoria y funciones como se indica en la **Tabla 1.2** en donde se detalla las especificaciones más importantes.

Tabla 1.2 PLCs OMROM: Características

Unidad Central	CPU C500	CPU C1000	CPU C2000H
Funciones	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales
Procesador	1 estándar	1 estándar	1 estándar 1 coprocesador
Tiempo de ejecución para instrucciones básicas	0.4 a 2.4 μ s	5 μ s	0.4 a 2.4 μ s
Memoria de programa RAM Interna EPROM y EEPROM	24 Kbytes 24 Kbytes	32 Kbytes 32 Kbytes	32 Kbytes 32 Kbytes
Temporizadores internos Margen de tiempo externos	128 0 s a 999.9 s con módulo externo	512 0 s a 999.9 s con módulo externo	512 0 s a 999.9 s con módulo externo
Contadores Internos Margen de Computo Externo	128 0 a 9999 con módulo externo	512 0 a 9999 con módulo externo	512 0 a 9999 con módulo externo
Entradas Digitales Salidas Digitales	512 en total	1024 en total	2048 en total
Entradas Analógicas Salidas Analógicas	2 2	2 2	2 2

1.8.3 CUTTLER HAMMER

Las especificaciones más importantes dadas para los PLCs Cuttler Hammer de la serie D500, que es la que mayores aplicaciones ofrece al programador, se indican en la **Tabla 1.3**.

Tabla 1.3 PLCs CUTTLER HAMMER: Características

Unidad Central	CPU 20	CPU 25 ^a	CPU 50A
Funciones	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales	Binarias, de cálculo, de cómputo, de carga, de comparación, de salto y funciones especiales
Procesador	1 estándar	1 estándar	1 estándar 1 coprocesador
Tiempo de ejecución para instrucciones 1 Kbyte	1.5 ms	1.5 ms	0.9 ms
Memoria de programa RAM Interna EPROM y EEPROM	4 Kbytes 4 Kbytes	4 Kbytes 4 Kbytes	8 Kbytes 8 Kbytes
Temporizadores internos Margen de tiempo externos	64 0 s a 999.9 s con módulo externo	64 0 s a 999.9 s con módulo externo	128 0 s a 999.9 s con módulo externo
Contadores Internos Margen de Computo Externo	64 0 a 9999 con módulo externo	64 0 a 9999 con módulo externo	128 0 a 9999 con módulo externo
Entradas Digitales Salidas Digitales	40 en total	40 en total	128 en total
Entradas Analógicas Salidas Analógicas	32 32	32 32	64 64

Además de lo indicado todos los fabricantes de PLC proveen de módulos que son compatibles con el equipo y cumplen tareas necesarias en el ambiente industrial pero se presentan como paquetes adicionales; entre éstas se mencionan:

- Acciones de control PID.
- Contadores de alta velocidad.
- Entradas y salidas analógicas.
- Controladores de temperatura tipo RTD y termocuplas.

- Programadores manuales.
- Software para control del proceso en tiempo real.

1.9 PROGRAMACIÓN DE PLC

El software usado en la programación de los controladores lógicos programables es la parte más importante de estos sistemas. Es el software lo que permite al usuario dar la lógica de funcionamiento al PLC en un formato que el usuario pueda fácilmente entender.

Hay dos tipos de lenguajes usados en la programación de PLC:

- Lenguaje de bajo nivel o básico.
- Lenguaje de alto nivel.

1.9.1 LENGUAJE DE BAJO NIVEL

Dentro de este grupo están:

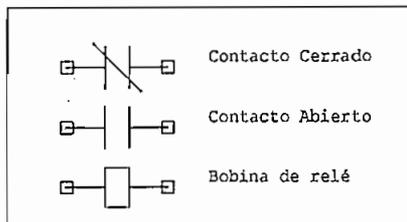
- Diagramas Ladder (Escalera).
- Lenguaje Booleano

El Diagrama Ladder (Escalera) es el lenguaje más ampliamente usado para la programación de PLC, es una programación gráfica en forma de escalera (de ahí el nombre de Ladder); tiene líneas verticales que representan las líneas de potencia, la cual fluye a través de los peldaños de los cuales se halla compuesto y respetando la secuencia a manera de una escalera.

Otra característica importante de éste tipo de programación es que los elementos pasivos se encuentran al lado izquierdo de la programación mientras que los elementos activos van al lado derecho.

Algunos de los símbolos comúnmente usados se muestran en la **Figura 1.7**.

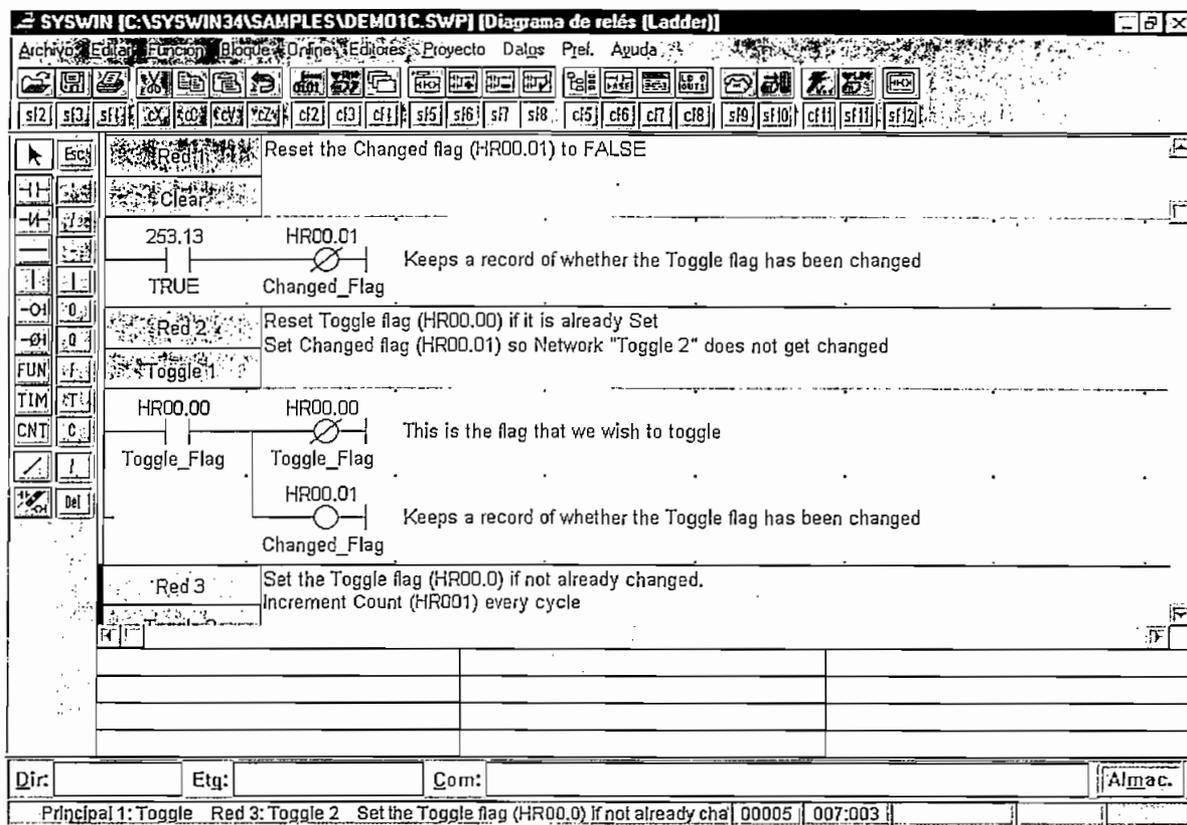
Figura 1.7 Algunos elementos usados en Programación Escalera



Cada fabricante ha puesto en el mercado su propia simbología, pero casi todos tienden a una similar a la indicada respetando normas internacionales.

Una aplicación típica desarrollada en diagrama escalera podría verse así:

Figura 1.8 Diagramación en Lenguaje Escalera



La programación en lenguaje booleano se basa en los principios del álgebra booleana, se puede programar introduciendo operadores booleanos y generalmente se ingresa como mnemónicos en la programación: AND, OR, NOT, NAND, NOR, etc., que son símbolos comúnmente usados para proveer un lenguaje de programación muy similar al utilizado en la programación de microprocesadores. Generalmente este tipo de lenguaje de programación se halla relacionado con los programadores manuales de los PLCs.

La razón de su popularidad es que es intercambiable con la programación escalera; símbolos booleanos tienen equivalencia en diagramas escalera. Pero de acuerdo con los diferentes fabricantes, este lenguaje es usado para realizar funciones más complejas que con la diagramación escalera.

1.9.2 LENGUAJES DE ALTO NIVEL

Los lenguajes de bajo nivel tienen su aplicación limitada cuando se trata de programar rutinas complejas usando un formato ya sea de diagrama escalera o booleano, en este caso se recurre a lenguajes de alto nivel, se considera dos tipos:

- Diagramas de Bloque.
- Lenguaje tipo Computadora.

Los Diagramas de Bloque son usados para formar un lenguaje de alto nivel que permita al programador implementar rutinas complejas usando un formato de diagrama escalera; los símbolos de bloques son ubicados en peldaños de manera similar a la realizada con los diagramas escalera.

Los Lenguajes Tipo Computadora son lenguajes que utilizan instrucciones dadas en inglés, usualmente muy similares a lenguajes de programación conocidos como por ejemplo el BASIC, de amplia difusión en computadoras personales, de ahí que muchos fabricantes hayan optado por estos lenguajes.

Tienen instrucciones de uso muy sencillo, y los paquetes que los soportan son muy amigables, con una amplia flexibilidad, lo que da al operador la facilidad de realizar programaciones muy complejas.

Este es el más poderoso de todos los lenguajes de programación de PLC y se hallan disponibles únicamente con las versiones grandes de los diferentes fabricantes.

1.10 COMUNICACIONES CON OTROS DISPOSITIVOS

Para comunicar entre sí PLCs y sistemas electrónicos pertenecientes a diferentes fabricantes existen varios esquemas de transmisión que son usados universalmente y que se han logrado estandarizar:

- Transmisión Paralela.
- Transmisión Serial.

La Transmisión Paralela es usada para comunicación entre el PLC y la computadora u otro dispositivo electrónico; los datos en forma de bits binarios, son almacenados en las compuertas lógicas del PLC y no son transmitidos hasta que un pulso de transmisión es recibido por estas compuertas. Hay que indicar, que el pulso de transmisión es también enviado al periférico conectado al PLC; al recibirlo las compuertas son reseteadas, lo que significa que los datos son previamente limpiados de los datos previos para preparar la posterior recepción.

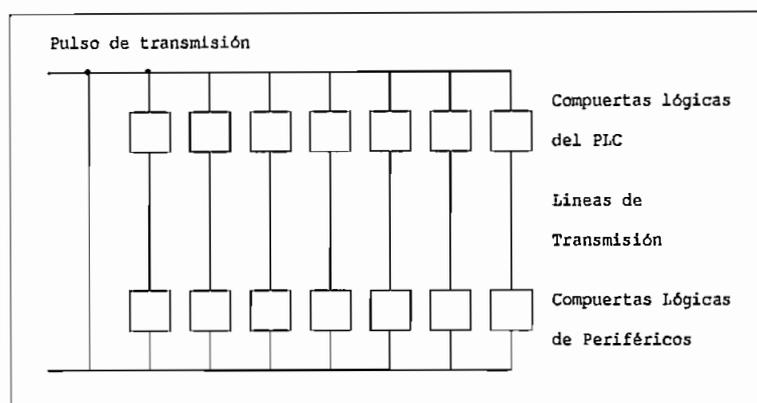
La característica más importante de esta transmisión es el número de líneas involucradas. En la transmisión paralela cada bit de cada compuerta en el PLC es transportado por líneas separadas. Una transmisión paralela implica que muchas líneas serán necesarias para las comunicaciones con sus periféricos.

La transmisión paralela presenta algunas desventajas, una de ellas es que se limita a pequeñas distancias (hasta 2 metros), esto por cuanto la transmisión es de un bit por línea; otra gran desventaja es que por ser líneas dedicadas,

generalmente éstas no son intercambiables con interfaces para diferentes periféricos.

Por otro lado su principal ventaja es la velocidad a la que se transmite la información; ya que muchos bits son enviados simultáneamente; desafortunadamente la velocidad a la que se envía la información es mayor que la que los pórnicos generalmente pueden procesar.

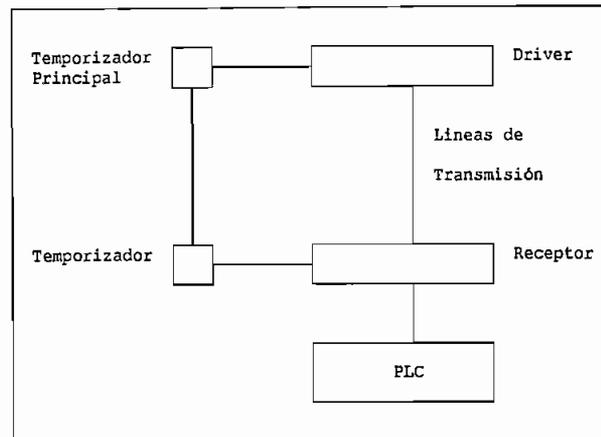
Figura 1.9 Comunicación Paralela del PLC



En la **Figura 1.9** un pulso de transmisión es usado para enviar información de las compuertas lógicas del PLC a las compuertas de los periféricos; resulta necesario el sincronizar el pulso de transmisión lo que se consigue usando un reloj en el PLC que es el que provee pulsos de transmisión a intervalos regulares; la transmisión inicia con el pulso de reloj y se ejecuta durante el intervalo de tiempo entre pulsos.

En la transmisión serial utilizada para la comunicación PC a PLC ó de PLC a otros dispositivos electrónicos los datos son enviados uno a la vez por una sola línea de comunicación.

Figura 1.10 Transmisión Serial del PLC



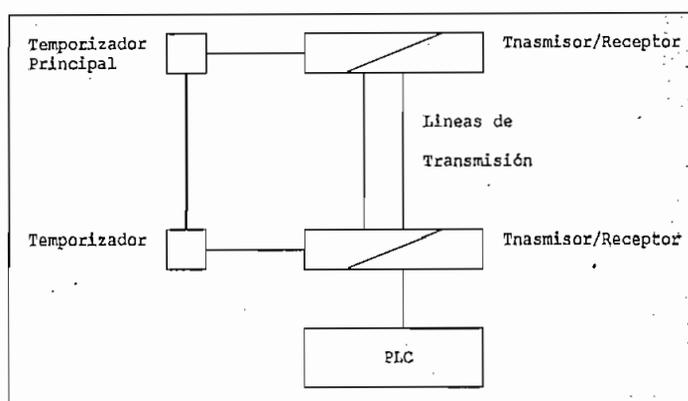
Los datos desde las compuertas lógicas del PLC son aceptados uno a la vez por los driver del dispositivo receptor. Cuando cada bit de datos es recibido por este driver, es manipulado o transmitido a través de una única línea. La transmisión serial es usada más comúnmente en sistemas de controladores lógicos programables, porque los periféricos de éstos son generalmente lentos (en algunos casos llega a ser de sólo 110 baudios) y requieren cables largos de conexión. Una distancia típica de transmisión serial puede ser de 15 metros en comparación con los 2 metros de transmisión paralela.

Dentro de la transmisión serial se puede diferenciar tres formas de realizarla:

- **Simplex o Unidireccional.-** Transmisión de datos en una sola dirección por medio de una única línea.
- **Half Duplex.-** La transmisión de los datos puede ser en las dos direcciones por medio de una sola línea, sin embargo la transmisión ocurre solo en una dirección a la vez.

- **Full Duplex.-** La transmisión de datos se realiza en las dos direcciones simultáneamente, pero necesita de dos líneas para este propósito, como se indica en la **Figura 1.11**.

Figura 1.11 Comunicación Paralela Full Duplex



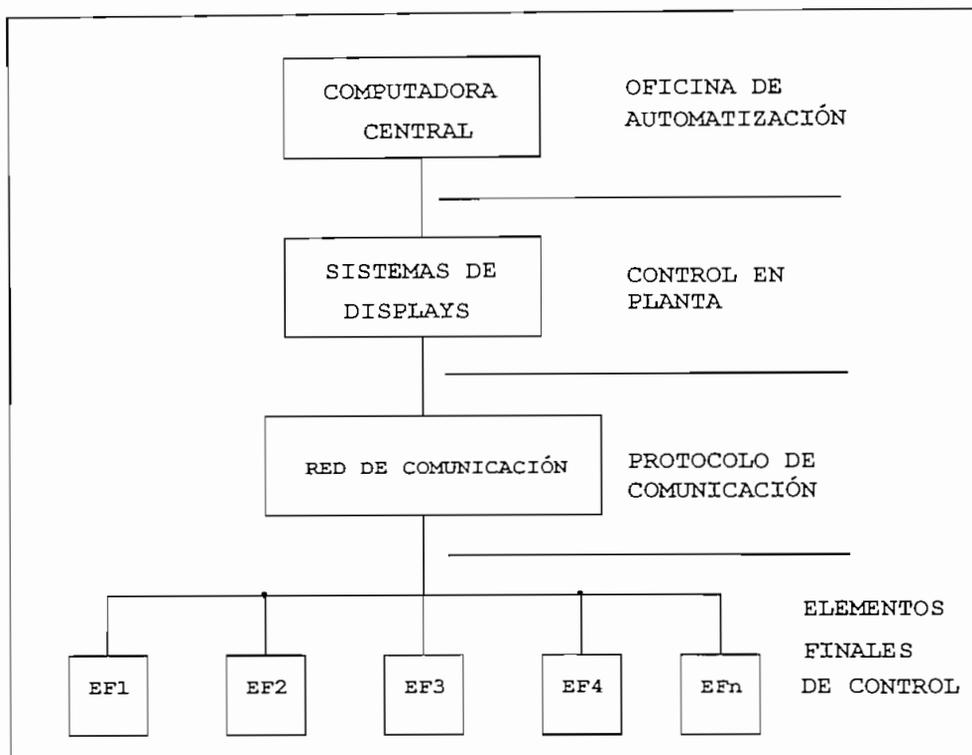
Las interfaces seriales más comúnmente utilizadas y estandarizadas para la transmisión de datos son: RS 232C, RS 449 y el IEEE 488, siendo el más popular el RS 232C.

En la actualidad se han formado redes completas de comunicación y control que se denominan sistemas SCADA y permiten enlazar PLC, elementos finales de control y una computadora que actúa como control central; estos sistemas permiten realizar el comando, control y monitoreo permanente de la información y proporcionan la posibilidad de ejecutar acciones correctivas de control en cualquiera de los elementos de la red.

Estos sistemas están diseñados para que la comunicación sea rápida y segura entre los elementos de la red.

Un sistema SCADA se lo puede apreciar en la **Figura 1.12**.

Figura 1.12 Sistemas SCADA



Para comunicación de este tipo de sistemas existen algunos formatos introducidos en el mercado, entre ellos se pueden mencionar:

- Comunicación tipo MODBUS.
- Comunicación tipo FIELDBUS.
- Comunicación tipo PROFIBUS.

La comunicación tipo MODBUS permite enlazar PLCs con elementos finales de control y una computadora central por medio de sólo dos hilos; es un sistema de comunicación de datos digitales; realiza un barrido total de todas las señales del sistema antes de ejecutar una acción de control.

La comunicación tipo MODBUS transmite datos digitales a la central de procesos para que éstos a su vez realicen acciones correctivas de control.

La comunicación tipo FIELDBUS es la evolución tecnológica para comunicaciones digitales en instrumentación y procesos de control.

Es diferente de otros protocolos de comunicación, porque está diseñado para resolver procesos de control en vez de sólo transferir datos en modo digital.

FIELDBUS es un sistema de control de dos vías, que interconecta equipos de campo como sensores, actuadores y controladores; FIELDBUS es una red local para instrumentos (LAN) usados en automatización de procesos y manufactura con capacidad de distribuir aplicaciones de control en la red.

PROFIBUS es un proceso digital del conjunto de líneas de la red, capaz de comunicar información entre un controlador maestro (PLC) y un proceso de dispositivo esclavo inteligente, también como de un PLC a otro. Profibus actualmente consiste en tres redes intercompatibles con protocolos diseñados para usarse en diferentes aplicaciones requeridas.

CAPÍTULO 2.

CARACTERÍSTICAS DEL CONTROLADOR LÓGICO PROGRAMABLE A DISEÑAR

En este capítulo se definen las características de diseño del PLC construido, no sólo en función de las especificaciones mencionadas en el capítulo anterior, sino de las necesidades básicas para el medio industrial en el país.

2.1 CONSIDERACIONES GENERALES

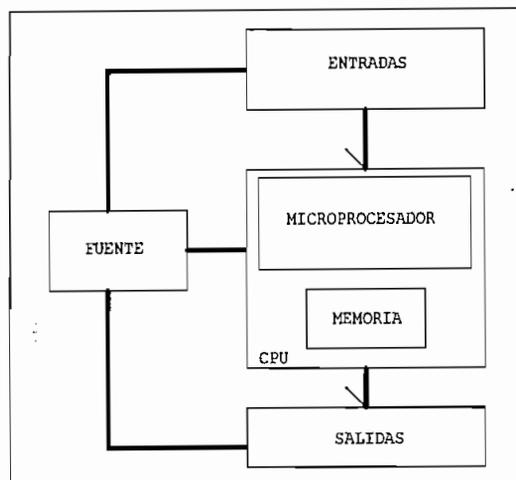
El PLC deber poseer una estructura solvente capaz de poder ser introducida en la industria, para lo cual debe cumplir con requisitos como los que se indican:

- Estructura para soportar temperaturas ambientes elevadas (30 °C).
- Facilidad para realizar conexiones con el medio externo: entradas, salidas, alimentación con la red eléctrica y comunicación con una PC.
- Ocupar el menor espacio físico posible.
- Protecciones internas contra posibles corto circuitos y / o sobrecargas.
- Facilidades para realizar el montaje donde sea requerido.
- El calor generado internamente debe ser disipado hacia el exterior.
- Visualmente agradable.
- Fácil para conectar con una computadora.
- Liviano, compacto.

2.2 CONSIDERACIONES DE HARDWARE

La configuración del PLC es la indicada en la **Figura 2.1**.

Figura 2.1 Diagrama de Bloques de un Controlador Lógico Programable



2.2.1 UNIDAD CENTRAL DE PROCESO

La Unidad Central de Proceso ó Etapa de Control Principal contiene como elementos principales el microprocesador, las memorias y circuitos adicionales y se encargará de ejecutar las siguientes funciones:

- Procesamiento de las señales de entrada.
- Envío de acciones de control modificadas a las salidas
- Almacenamiento de la información introducida por el usuario.
- Monitoreo del estado de las señales de entradas y salidas.
- Proporcionar los parámetros de control adecuados de manera que se puedan manejar todo su software sin problemas.
- Almacenar la lógica de control previamente programada así como la lógica personalizada introducida por el usuario de cada PLC.
- Retener la información por tiempo indefinido hasta que sea el operador el que introduzca una nueva secuencia de programa y cumpla otras funciones.
- La información debe ser alterable el momento en que se requiera.
- Posibilidad de ampliar la capacidad de memoria.
- Mantener una interfaz para la comunicación con una computadora y su

software; de manera que las aplicaciones se puedan realizar en un entorno fácil de entender y aplicar.

- Aislar de cualquier señal externa que pueda ocasionar daños ya que la pérdida de ésta provocaría el daño total del control; debe estar protegida contra el calor externo excesivo y permitir su disipación de calor.

La tarjeta de control que se diseñará abarcará en su interior todas las características arriba descritas, de las cuales se detallará su diseño en software y hardware en los siguientes capítulos. Se usará como elementos principales un microprocesador de la serie MCS 51; sumados a él memorias tipo RAM y ROM externas.

2.2.2 ENTRADAS FÍSICAS

En la actualidad existen un sin número de tipos de señales de entrada en los procesos de control tanto analógicas como digitales que pueden ser utilizadas para monitorear un proceso; se pueden tener: pulsadores, interruptores de fin de carrera, sensores ópticos, sensores de temperatura, señales provenientes de contactores, breakers, sensores de nivel capacitivos, sensores de proximidad, etc.

En suma, en la industria, sin importar el tipo de proceso, las señales para control pueden ser analógicas o digitales siendo las más utilizadas en procesos tipo On/Off las señales digitales.

Las entradas físicas pueden ser provenientes de elementos pasivos o activos. Los Elementos Pasivos son todos aquellos que emiten una señal sin energía (voltaje o corriente), es decir sólo emiten un estado (abierto o cerrado); por ejemplo: un contacto de un relé, una señal de un interruptor, una señal de un pulsante, de un sensor magnético, un sensor de fin de carrera, etc. En su mayoría señales que provienen de alguna acción mecánica.

Los Elementos Activos son los que emiten una señal con energía (voltaje o corriente), al ser actuadas. Como ejemplo se menciona a un sensor óptico: al interrumpir el haz de luz un transistor se satura enviando una señal de voltaje a través de éste hacia el medio externo. Este tipo de elementos también emite señales analógicas o digitales; las señales digitales vienen en dos niveles de tensión: $+V_{cc}$ y 0 V ó $-V_{cc}$ y $+V_{cc}$.

También hay que considerar que las señales de entrada se encuentran actuando con el medio externo, es por ello que están expuestas a las interferencias que se puedan presentar, tales como: picos de corrientes, variaciones de tensión, ruido, etc. que afectan el funcionamiento de cualquier equipo electrónico, de ahí que sea necesario el aislar las señales de entrada a la tarjeta de control principal.

La tarjeta de entradas físicas a diseñarse tendrá 8 entradas digitales pasivas e independientes, cada una con aislamiento hacia la tarjeta de control principal y con indicación visual del estado en que se encuentra.

2.2.3 SALIDAS

La etapa de salida es el medio que va a producir la comunicación con el proceso, sus funciones serán:

- Enviar las señales emitidas desde la tarjeta de control hacia las señales físicas de salida.
- Indicar cuando han sido energizadas.
- Manejar cargas de control (menores a 5 amperios) tanto en corriente continua como en corriente alterna.
- Retener el estado de las salidas para que la tarjeta de control pueda cumplir otras funciones.

La tarjeta de salidas será de 8 salidas digitales aisladas e independientes, capaz de manejar una carga de control tanto en corriente alterna como en corriente

continua y con indicación visual del estado en que se encuentra. Se usará relés como elemento de salida con un contacto normalmente abierto.

2.2.4 FUENTE

Es necesario contar con una fuente capaz de alimentarse desde cualquier red existente en la industria ecuatoriana, lo más común es tener tensiones de 110 Voltios y 60 Herzios, por ello que se ha tomado como voltaje de entrada a la fuente este valor.

La fuente proveerá alimentación a todas las etapas que involucrarán el PLC; concretamente a las etapas de entradas, salidas y la tarjeta de control principal (CPU).

La fuente debe estar aislada eléctricamente de manera de evitar que los campos electromagnéticos que se produzcan a su alrededor interfieran sobre el funcionamiento de las diferentes tarjetas del PLC sobretodo con la de control principal.

También debe ser capaz de disipar su calor generado, para que éste no afecte a su vez el normal funcionamiento del equipo.

Otro aspecto de relevancia es que la información introducida por el usuario debe permanecer en el equipo aún cuando se hayan producido cortes de energía, es por ello que se necesita disponer de una batería, para respaldar la información contenida en la memoria RAM, cuando algún problema en el suministro eléctrico se haya producido.

La fuente de alimentación en el lado primario será 110 V, 60 Hz, regulada para la etapa de control, capaz de soportar la carga que produzcan todos los elementos del hardware con adecuada disipación de calor y con sus respectivas protecciones. Además batería de respaldo para la memoria RAM.

2.3 CONFIGURACIÓN DE SOFTWARE

Para el diseño se debe considerar que existen dos partes complementarias y mutuamente relacionadas en software:

- Software para el microcontrolador.
- Software para la comunicación y programación.

2.3.1 SOFTWARE PARA EL MICROCONTROLADOR

En el microcontrolador se encontrarán programadas todas las funciones que el PLC realizará para poder proporcionar el control y la lógica de programación así como también la comunicación con una computadora para cambio de programas.

Se considerará para el diseño las siguientes funciones básicas que son las más usadas en la industria:

- Contactos abiertos y cerrados de relés internos.
- Contactos abiertos y cerrados de entradas y de salidas.
- Contactos abiertos y cerrados de temporizados ON DELAY y OFF DELAY.
- Contactos abiertos y cerrados de contadores.
- Bobinas de relés internos.
- Bobinas de relés de salidas.
- Bobinas de temporizados ON DELAY y OFF DELAY.
- Bobinas de contadores.

Además se contará con una interfaz de comunicación con una computadora de manera que el usuario pueda introducir el programa diseñado una rutina para hacer mantenimiento.

Con las características mencionadas el software para el microprocesador manejará las siguientes funciones:

- *8 Entradas Digitales.*
- *8 Salidas Digitales.*
- *16 Temporizados On Delay.*
- *16 Temporizados Off Delay.*
- *16 Contadores de eventos.*
- *64 Relés Internos.*
- *Contactos Abiertos para Temporizados On Delay y Off Delay, Contadores, Relés, Salidas y Entradas.*
- *Contactos Cerrados para Temporizados On Delay y Off Delay, Contadores, Relés, Salidas y Entradas.*
- *Comunicación con una computadora.*
- *Software para mantenimiento.*

2.3.2 SOFTWARE PARA PROGRAMACIÓN Y COMUNICACIÓN

Para que el PLC pueda ser fácilmente programado debe disponer de un entorno amigable, con programación sencilla y comprensible, de manera que se pueda visualizar y ubicar errores en la programación fácilmente y proveer ayudas al operador.

Con estas características, el software para la computadora está dividida en dos bloques:

- Software para programación.
- Software para comunicación.

2.3.2.1 Software para Programación

El usuario necesita crear el programa para la aplicación requerida, antes de ser enviado al controlador lógico programable. Por ello se debe diseñar una interfaz que sea de comprensión rápida y aplicación sencilla; por lo que se trabajará en entorno Windows.

Se tendrá una programación en un lenguaje universal de PLC's como es el lenguaje escalera, en donde se encuentren definidos todos los elementos que se pueden utilizar al diseñar un programa.

Es también necesario que el programa sea capaz de encontrar errores de concepto, y proporcionar ayudas cuando se lo requiera.

Se utilizará Visual Basic versión 5.0 para programar la interfaz en el que se realizarán las aplicaciones para el PLC.

2.3.2.2 Software para Comunicación

Es necesario poder ingresar al PLC las instrucciones que han sido programadas en lenguaje escalera, es por ello que el programa debe ser capaz de compilar la información para transformarlo en códigos hexadecimales para poder enviar los datos hacia el PLC. Para la comunicación se utilizará el pórtico serial RS 232C.

CAPÍTULO 3.

DISEÑO DEL HARDWARE

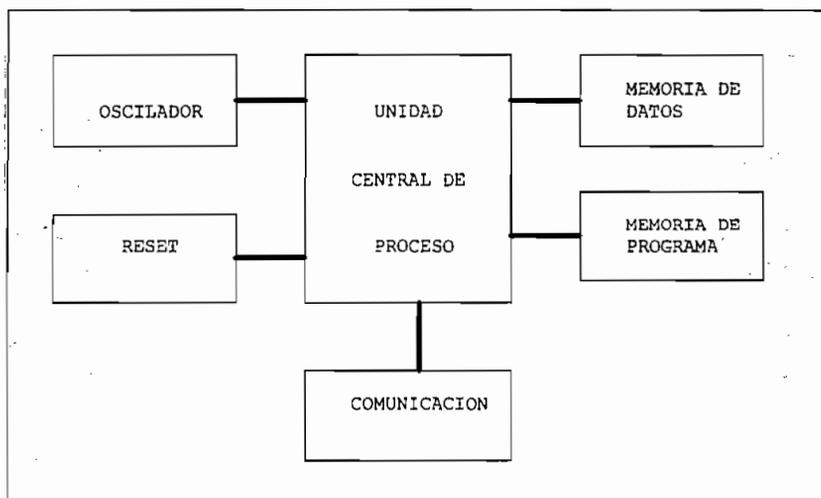
Al hardware se lo ha dividido en cuatro bloques conforme lo explicado en el capítulo anterior:

- Etapa de control.
- Etapa de entradas.
- Etapa de salidas.
- Fuente de alimentación.

3.1 DISEÑO DE LA ETAPA DE CONTROL

El diagrama de bloques de la etapa de control se indica en la **Figura 3.1**

Figura 3.1 Diagrama de Bloques de Etapa de Control



3.1.1 UNIDAD CENTRAL DE PROCESO.

3.1.1.1 Elección del Microprocesador.

Es necesario contar con un microprocesador que maneje un set de instrucciones amplio, con capacidad para realizar comunicaciones, que sea compatible con elementos externos, que necesite pocos elementos físicos adicionales para su control, entre otras características.

El microcontrolador de la serie MCS 51 es un procesador óptimo para aplicaciones de control, que cuenta con las siguientes características:

- 32 líneas de entrada / salida (bidireccional).
- 64 Kbytes de espacio de memoria externa de programa.
- Set de 111 instrucciones (64 de un sólo ciclo).
- Amplio rango de temperatura de trabajo (-40° C a 85° C).
- 128 bytes en memoria RAM interna.
- Canal Serial Full Duplex Programable.
- Procesador Booleano
- Espacio de memoria RAM direccionable bit a bit.
- 5 interrupciones con nivel de prioridad programables.
- Pórtico de comunicaciones RS232C.

El microcontrolador MCS 8751 dispone en su interior de 4 Kbytes de memoria EPROM y 128 bytes de memoria RAM interna, lo que va a permitir probar las aplicaciones en el propio microcontrolador.

Este microcontrolador cuenta además con mucho software de apoyo para emulación, compilación y simulación.

Otra característica por la que se seleccionó este microcontrolador es su similitud con el microcontrolador MCS 8X52, con las mismas características pero con el

doble de capacidad en memoria RAM y EPROM internas, lo que permitiría en el proceso cambiar de microcontrolador si es necesaria una mayor capacidad en memorias.

3.1.1.2 Hardware para el microcontrolador MCS 51

Los microcontroladores MCS 51 necesitan un circuito oscilador, el MCS 87H51 es capaz de soportar hasta 12 MHz de frecuencia de operación, sin embargo para el diseño se va a usar un cristal de 3.579545 MHz, el cual define la velocidad a la que se ejecutan las instrucciones dentro del microcontrolador y al mismo tiempo deja la ventana abierta para el reemplazo con otro microcontrolador de la serie.

Se usará el latch 74LS573 conectado a los pórtricos P0 y P2 para trabajar bidireccionalmente manejando datos y direcciones con acceso a las memorias RAM y EPROM. Un circuito de reset para reiniciar el microcontrolador que envía al puntero de datos del microcontrolador a su posición inicial, limpia las banderas y los registros del microcontrolador. Se encuentra formado por un pulsante, un diodo de control para evitar corrientes reversas y un capacitor para corrientes parásitas; el circuito para el reset al igual que el oscilador se tomó del manual del microcontrolador MCS 51 dado por el fabricante.

En el microcontrolador se usará el menor número de pines posible para dejar la posibilidad de ampliar el circuito en posteriores aplicaciones.

3.1.2 ALMACENAMIENTO DE PROGRAMA.

La unidad central de proceso necesita de un software para que pueda ejecutar acciones de control sobre el PLC, este software es necesario almacenarlo en una memoria tipo EPROM. Se usará la memoria interna del microcontrolador para almacenar el programa; sin embargo, en el hardware se ha considerado agregar memoria EPROM externa: la memoria 27256 de 8K*8 bytes, 64 Kbytes de memoria, para poder reemplazar el microcontrolador MCS 8751 por el MCS 8031 o permitir agregar más aplicaciones al programa.

3.1.3 ALMACENAMIENTO DEL PROGRAMA DE USUARIO.

La base del funcionamiento de un PLC es la facilidad que tiene el usuario para realizar aplicaciones propias, estas aplicaciones necesitan espacio de memoria que permita hacer modificaciones. El tipo de memoria que permite alterar la información cada vez que se requiera es RAM y se usará la memoria DS1230Y de 64 Kbytes aunque su capacidad es muy superior a la requerida para aplicaciones típicas de un PLC; sin embargo se la usó para permitir futuras ampliaciones del equipo.

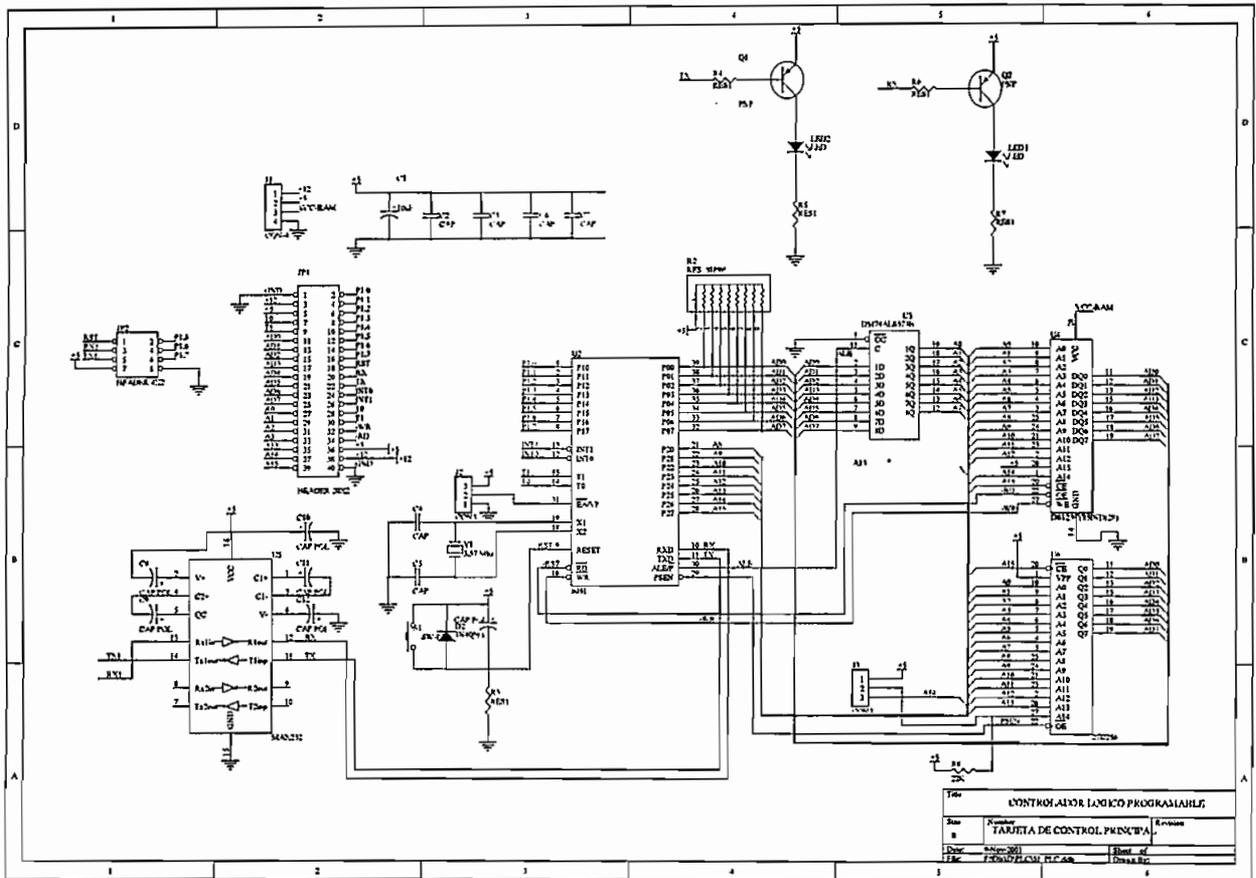
Para aplicaciones permanentes, el programa puede ser cambiado a una memoria tipo EEPROM, aunque esto va a limitar el uso del PLC a una aplicación definida, pero como ventaja mantiene almacenados los datos por un mayor tiempo.

3.1.4 COMUNICACIÓN.

En esta etapa del diseño se considera la comunicación, tanto para programación como para mantenimiento con una computadora, para ello se usa transmisión serial por el pórtilo estándar RS232C, mediante el conversor MAX232 que transforma señales TTL enviadas por el microcontrolador en señales de voltaje +/- 10V aceptadas por el pórtilo de comunicaciones y viceversa; la comunicación no necesita ser rápida ya que se usa sólo en programación, se hará a una velocidad de 9600 bps.

El diagrama de la tarjeta de control del PLC se halla detallado en la **Figura 3.2.**

Figura 3.2 Diagrama de Etapa de Control del PLC

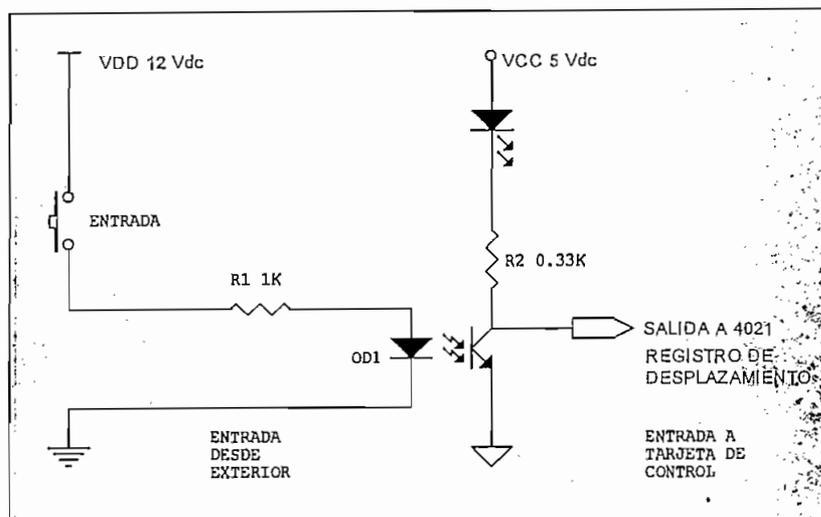


3.2 DISEÑO DE LA ETAPA DE ENTRADAS DEL PLC

Se tendrá 8 entradas pasivas aisladas e independientes, todas similares a la mostrada en la **Figura 3.3**; sin embargo con las consideraciones indicadas se deja la posibilidad para hacer modificaciones en sus especificaciones en función a las aplicaciones individuales, para ello existe una fuente interna que será la encargada de suministrar la alimentación adecuada a las 8 entradas pasivas.

Una posible modificación sería la modificación para trabajar con señales externas alternas, para ello se debe cambiar la configuración de las entradas y utilizar optoacopladores tipo optotriacs.

Figura 3.3 Etapa de Entradas a la Tarjeta de Control



3.2.1 DESCRIPCIÓN DEL FUNCIONAMIENTO

Al recibir una señal de entrada, cualquier tipo de entrada de elementos de control pasivos, circulará corriente por el diodo de la etapa de entrada del optoacoplador, la resistencia R1 será la encargada de limitar la corriente que circula por el diodo. Al circular corriente por el optodiodo, el transistor de la etapa de salida del optoacoplador se satura cambiando el estado de 0 a 1 lógico permitiendo el paso de corriente hacia el diodo led que visualmente indica que una señal ha sido activada y de la misma manera hacia la entrada del registro de desplazamiento, la corriente del led se controlará mediante R2.

Se tomará para el diseño el optoacoplador NTE 3041, que va a proporcionar el aislamiento entre las entradas y la tarjeta de control; evita que se reflejen directamente los posibles problemas que se puedan presentar a las entradas; y también dar la posibilidad de trabajar con niveles de tensión diferentes.

Las ocho entradas independientes pasarán al registro de desplazamiento NTE 4021 e ingresarán los datos al microcontrolador de manera serial. Se ha escogido

esta forma de enviar los datos básicamente porque se usa un menor número de señales en el microcontrolador aunque se aumenta el tiempo de transmisión.

Esta etapa requiere de tres señales de envío de información hacia y desde el microcontrolador:

- Señal de reloj para el desplazamiento de los datos de entrada en el registro de desplazamiento (viene del microcontrolador).
- Señal de habilitación para el paso de los datos serialmente al microcontrolador (viene del microcontrolador).
- Señal que envía los datos hacia el microcontrolador (va al microcontrolador).

Para estos datos se van a usar los pines:

- P1.0 Señal de reloj.
- P1.1 Pulso de habilitación para transmisión de datos.
- T0 Entrada de datos al microcontrolador desde pin Q8 de registro de desplazamiento NTE 4021.

3.2.2 CALCULO DE RESISTENCIAS DE LIMITACIÓN

Para el diseño se definen los siguientes parámetros:

- Nivel de tensión de entrada 12 Vdc. Para el medio externo.
- Nivel de tensión de salida 5 Vdc.
- Corriente de entrada al registro de desplazamiento 500 μ A.
- Corriente máxima por el diodo 10 mA.

El optoacoplador NTE 3041 de fácil adquisición es un optotransistor tipo NPN con un voltaje de aislamiento de 7500 Vdc y una potencia de disipación de 250 mW.

Características del optoacoplador NTE 3041:

- $I_{Fm\acute{a}x} = 60 \text{ mA}$. (Corriente máxima por el diodo)
- $I_{Cm\acute{a}x} = 100 \text{ mA}$. (Corriente máxima de colector)

Con estos datos la corriente de máxima en la etapa de entrada del optoacoplador:

$$R1_{\min} = (V_{dc} - V_d) / I_{Fm\acute{a}x}$$

$$R1_{\min} = (12V - 1.3 V) / 60 \text{ mA}$$

$$R1_{\min} = 179 \Omega$$

La resistencia de entrada será la de mayor valor normalizada, con ello estará limitada la corriente para proteger al optoacoplador.

$$R1 = 1 \text{ K}\Omega$$

$$I_{in} = 10.7 \text{ mA}$$

En la etapa de salida el calculo de la resistencia es:

$$R2_{\min} = (V_{cc} - V_{led} - V_{tsat}) / I_{Cm\acute{a}x}$$

$$R2_{\min} = (5 V - 1.3V - 0.7 V) / 10 \text{ mA}$$

$$R2_{\min} = 300 \Omega$$

El diodo led es el limitante de la corriente de salida, por ello, se escoge una resistencia normalizada similar al valor calculado.

$$R2 = 330 \Omega$$

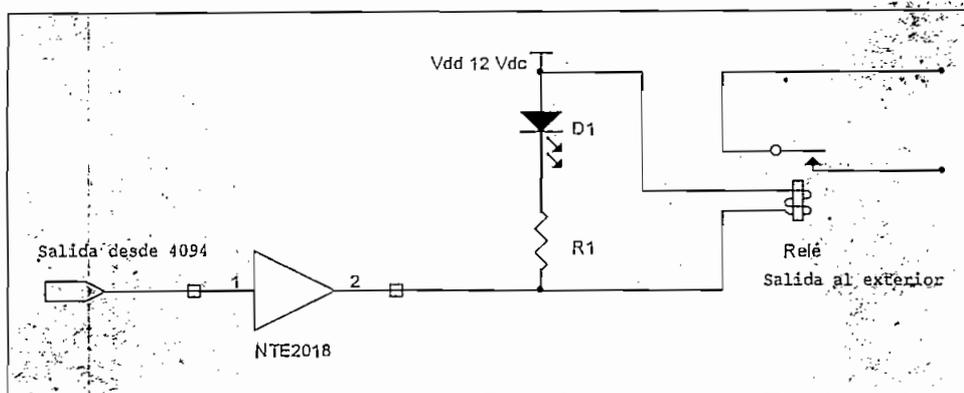
$$I_{out} = 9.0 \text{ mA}$$

3.3 DISEÑO DE LA ETAPA DE SALIDAS DEL PLC

Para el diseño de la etapa de salidas se ha considerado que deben manejar tanto señales de corriente alterna como de corriente continua; es por ello que como elementos de salida final se usan relés; con un contacto normalmente abierto.

La configuración de cada una de las salidas se pueden ver en la **Figura 3.4**.

Figura 3.4 Etapa de Salidas de la Tarjeta de Control



3.3.1 DESCRIPCIÓN DEL FUNCIONAMIENTO

Una vez que la lógica de control se ha ejecutado, la señal se envía serialmente del microcontrolador hacia un registro de desplazamiento NTE 4094, en el cual se almacenan las señales que van arribando hasta que todas hayan llegado. De ahí salen a un amplificador de corriente NTE 2018 para luego pasar a manejar la bobina de un relé.

De ser necesario se puede modificar el circuito para aplicaciones específicas de corriente continua o alterna, ya sea con transistores o triacs; así mismo se puede cambiar la configuración para usar el contacto cerrado en vez del abierto de requerirlo en una aplicación determinada.

Cada salida dispone de un led de indicación del estado en el que se encuentra, el mismo que cumple también con la función de eliminar corrientes parásitas.

Para el funcionamiento se requiere que por el led circule máximo una corriente de 10 mA, por lo cual la resistencia será de 1.2 K Ω .

Esta etapa requiere de tres señales de datos para envío de información hacia y desde el microcontrolador:

- Señal de reloj para el desplazamiento de datos de entrada en el registro de desplazamiento (viene del microcontrolador).
- Señal de habilitación para el paso de datos paralelos al microcontrolador (viene del microcontrolador).
- Señal para envío de datos desde el microcontrolador hacia el registro de desplazamiento.

Para estos datos se van a usar los pines:

- P1.0 Señal de reloj.
- P1.2 Pulso de habilitación para transmisión de datos.
- T1 Salida de datos desde el microcontrolador hacia el registro de desplazamiento.

3.4 DISEÑO DE FUENTE

Para el diseño de este PLC se ha visto la necesidad de tener internamente dos niveles de tensión:

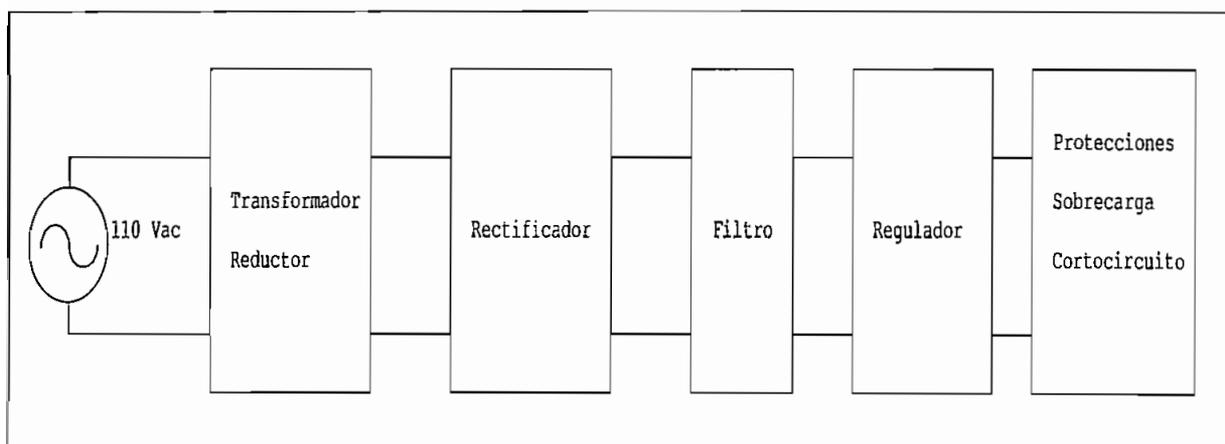
- Una tensión de 5 Voltios de corriente continua regulada para el manejo de la tarjeta de control principal que contendrá al microcontrolador y circuitos

integrados complementarios. Por exigencias del microcontrolador esta fuente debe ser regulada a $5\text{ V} \pm 5\% \text{ V}$.

- Una alimentación de 12 Voltios de tensión continua para las etapas de entradas y salidas; ésta no necesita ser regulada, ya que va a manejar básicamente relés, elementos de aislamiento y elementos de indicación que soportan fácilmente variaciones de tensión.

La fuente en bloques se indica en la **Figura 3.5**:

Figura 3.5 Diagrama de Bloques de Fuente



Para determinar la carga que debe manejar la fuente se usa el método de suma de corrientes, agregándole un 20% de margen de seguridad conforme sugieren los fabricantes.

La fuente de potencia en 5 Voltios regulada manejará los elementos mostrados en la **Tabla 3.1**.

Tabla 3.1 Capacidad de Corriente por Elemento

Elemento Semiconductor	Capacidad de Corriente Máx. Miliamperios	Regulación de Voltaje Voltios
Microcontrolador MCS 87H51	150	+5.25 –4.75 Vdc
Memoria RAM 2128 de 4 Kbytes	100	+5.25 –4.75 Vdc
Registros de desplazamiento (2) 4021 y 4094	40	+5.25 –4.75 Vdc
Latch 74LS573	20	+5.25 –4.75 Vdc
Leds de indicación (10)	100	
Circuito de transmisión serial MAX 232	40	+5.25 –4.75 Vdc
TOTAL DE CORRIENTE	450	

Total de corriente para la fuente regulada máxima: 450 mA

Total de corriente regulada con margen de seguridad del 20%: 540 mA.

La fuente de potencia de 12 Voltios manejará los elementos mostrados en la **Tabla 3.2:**

Total de corriente para la fuente 12 voltios: 200 mA

Total de corriente de fuente no regulada necesaria: 240 mA.

Corriente de fuente máxima: 780 mA.

Tabla 3.2 Capacidad de Corriente de Entradas/Salidas

Elemento Semiconductor	Capacidad de Corriente Máx. MA	Regulación de Voltaje Voltios
Leds de indicación (8)	80	
Relés de salida (8)	80	
Optoacopladores (8)	40	
TOTAL DE CORRIENTE	200	

Transformador.- Usado para reducir el voltaje de entrada y permitir tener niveles cercanos a los requeridos para la adecuada regulación de tensión, una reducción considerable de la disipación de calor y el trabajo en condiciones más seguras con niveles pequeños de tensión.

El voltaje primario más usado es 110 Voltios 60 Herzios, aunque se pueden hacer las consideraciones adecuadas de necesitarse otro nivel de tensión o para trabajar con corriente continua.

Se tiene que la corriente total llegará a 780 mA suponiendo que se encuentran operando todos los elementos a la vez, por ello se tomará como referencia para el diseño un transformador de 110V / 12 V con 1 A.

Rectificador.- Para el microprocesador se requiere una fuente que tenga una muy buena regulación con el mínimo de rizado, es por ello que para la etapa de rectificación se usará un rectificador de onda completa tipo puente, que se lo puede fácilmente obtener en una sola pastilla integrada. Otra ventaja de usar el rectificador tipo puente es que se requiere un transformador con un solo nivel de tensión (no es necesario tap central).

Se ha tomado el rectificador ECG5332 tipo puente que tiene una capacidad de corriente de 1 A y voltajes de trabajo de hasta 110 Vac.

Filtro.- Se usará condensadores tanto a la entrada como a la salida del rectificador lo que garantiza el nivel de tensión similar a DC pura; y por supuesto reduce el rizado.

Este filtro además debe contener un condensador adicional de un décimo a un centésimo del valor del capacitor de filtro principal en paralelo, este capacitor provee un camino alternativo de fuga para las corrientes en altas frecuencias, donde por causa de la resistencia y la inductancia del filtro principal, éste no es muy efectivo. De existir aun ruido excesivo por frecuencias, un tercer capacitor de un valor de un décimo del segundo puede ser adherido de manera de eliminar el ruido existente.

Regulador.- Se usará el ECG 7805 que posee una tensión de regulación de 5 voltios con un 5% de variación y que es capaz de mantener la regulación hasta 1000 mA, con un voltaje de entrada mínimo de 8 Vdc.

El regulador requiere de un disipador de calor, se usará el recomendado por el fabricante.

Protecciones.- Contra cortocircuito y sobrecarga vienen incorporadas en el circuito integrado del regulador 7805 de voltaje. Adicionalmente se colocará un fusible a la entrada de la alimentación de corriente alterna.

3.4.1 BATERÍA DE RESPALDO

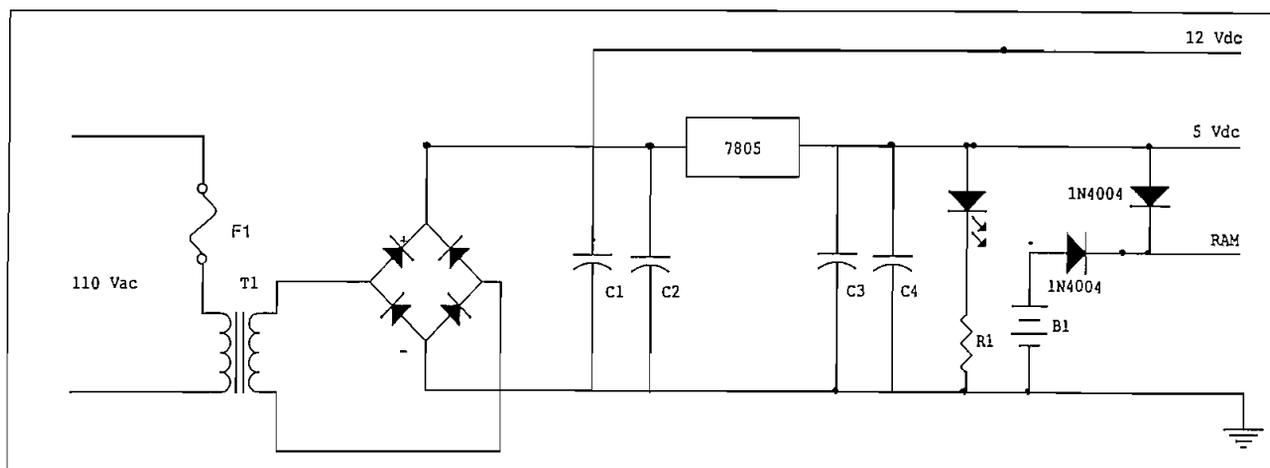
Es necesario mantener la información programada por el usuario en caso de corte de energía (la existente en memoria RAM) por ello se hace necesario colocar una batería de respaldo que mantenga la información de esta memoria (si se usa una memoria de tipo volátil); para aplicaciones en donde la información del PLC va a ser almacenada una sola vez es preferible el uso de memoria EPROM para evitar que por cualquier motivo la información se pierda.

Entre los tipos más usados de baterías de respaldo están las de Carbón y Zinc, Alcalinas, Nikel Cadmio, Litio y Carga Acida. Generalmente las más usadas son las de Carbón y Zinc, Litio y Alcalinas por su característica de ser recargables.

La batería utilizada para tener permanentemente los datos en la memoria RAM es de 3.6 V, Ni-Cd, 60 mAh, no recargable, por especificaciones del fabricante, tiene una vida útil aproximada de 2 años.

En la **Figura 3.6** se puede ver el diagrama de la fuente y en los anexos se encuentran los circuitos completos del hardware.

Figura 3.6 Fuente de Alimentación para PLC



CAPÍTULO 4.

DISEÑO DEL SOFTWARE

El PLC necesita dos programas independientes y relacionados entre sí:

- Software para el manejo del microcontrolador serie MCS 51.
- Software para la computadora.

4.1 DISEÑO DEL SOFTWARE PARA EL MICROCONTROLADOR MCS 51

El hardware se ha diseñado utilizando el microcontrolador MCS 8751, entre otras características por su bajo costo y su facilidad de encontrar en el mercado; para la realización de este programa se ha usado el editor de ensamblador denominado PINNACLE 52.

El programa implantado para este microcontrolador es compatible con la versión MCS 52; sin embargo se debe considerar la mayor capacidad de memorias RAM y EPROM que éste tiene.

4.1.1 DESCRIPCIÓN DEL SOFTWARE PARA SIMULACIÓN Y DEPURACIÓN PINNACLE 52

El PINNACLE 52 es un software que permite editar, ensamblar, compilar y simular programas para implementarlos en los microcontroladores de las series MCS 51 y MCS 52. Presenta múltiples ventajas con relación a ensambladores similares de otros fabricantes, entre las que se pueden mencionar:

- Es relativamente barato y una versión libre se puede obtener en el Internet.

- Está diseñado para Windows, por ello mantiene ventanas con las mismas características y menús que programas similares desarrollados, es mucho más amigable que otras versiones existentes para DOS.
- Permite hacer simulaciones introduciendo consideraciones de hardware como la velocidad del oscilador que utiliza el microcontrolador; lo que da la posibilidad de hacer simulaciones en tiempo real.
- Posibilidad de realizar simulaciones de comunicaciones a través del púrtico serial con otra computadora e inclusive simulaciones en la misma computadora si se usa dos programas de comunicación diferentes.
- Permite desensamblar archivos hexadecimales obtenidos de un microcontrolador de la serie y transformarlo a un archivo de mnemónicos. Sin embargo para tener disponible esta opción se hace necesario adquirir la licencia profesional del programa.

4.1.2 CONFIGURACIÓN DE LAS MEMORIAS DEL MICROCONTROLADOR MCS 51

El microcontrolador MCS 8751 tiene 4 bloques de memoria dependiendo de la configuración de hardware utilizada; en el PLC diseñado se han dejado habilitadas todas las posibilidades de memorias aunque no se usen. Se detalla a continuación la configuración utilizada:

- Memoria RAM interna (128 bytes).
- Memoria EPROM interna (4 Kbytes).
- Memoria EPROM externa (64 Kbytes).
- Memoria RAM externa (64 Kbytes).

4.1.2.1 Distribución de la Memoria RAM interna del Microcontrolador MCS 51.

El microcontrolador MCS 51 cuenta con 128 bytes de memoria RAM interna organizados de tal manera que puedan almacenar todos los elementos diseñados en el PLC.

A continuación se detallan los elementos y la memoria ocupada por cada uno de ellos.

- 8 Entradas (1 byte).
- 8 Salidas (1 byte).
- 16 Temporizados On Delay (4 bytes).
- 16 Temporizados Off Delay (6 bytes).
- 16 Contadores de Eventos (6 bytes).
- 64 Relés (8 bytes).
- Espacio para el procesamiento de la información (16 bytes).
- Espacio para el Puntero de Datos (32 bytes).
- 8 Registros que cumplen diferentes funciones de programa (8 bytes).
- Espacio reservado para el almacenamiento bit a bit (16 bytes).

En este espacio de memoria se almacena únicamente el estado de cada elemento; para su operación se usará la memoria RAM externa.

En suma se han utilizado 98 bytes de memoria RAM interna, esto permite tener disponible aún 30 bytes de memoria en reserva para ampliar funciones en el PLC.

En la **Tabla 4.1** se detalla la distribución de las localidades de memoria RAM así como las etiquetas utilizadas para su identificación.

TABLA 4.1
DISTRIBUCIÓN DE MEMORIA SFR INTERNA DE MICROPROCESADOR 8751
APLICACIÓN PLC

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
0	R0	R2	R3	R4	R5	R6	R7	DISPONIBLES									
1	FUNCIONES ESPECIALES DIRECCIONABLES Bit A Bit																
2	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP		
3	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP		
4	DISPONIBLES																
5	OPERACIONES DE ESCALERA DESDE 60H a 6FH																
6	RELES INTERNOS 60H a 6FH																
7	STCON	ON AUX	AUCON	PUAUX	SACON	SALON	STOFF	SAL	ENT								

- ENT: Almacenamiento de Localidades de Entradas desde T0
- SAL: Almacenamiento de Localidades de Salida por el pin T1
- STOFF: Almacenamiento de Localidades de Temporizados OFF DELAY
- SALON: Almacenamiento de Localidades de Temporizados ON DELAY
- SACON: Almacenamiento de Localidades de Contador de Eventos
- PUAUX: Almacenamiento de Localidades de Datos Auxiliares Para Temporizados OFF DELAY.
- AUCON: Almacenamiento de Localidades Auxiliares en Contadores.
- ON AUX: Almacenamiento de Localidades de Transferencia de Temporizados ON & OFF DELAY.
- STCON: Almacenamiento de Localidades de Contador de Eventos
- DACON: Almacenamiento de Localidades de Transferencia de Contadores a RAM.
- SP: Puntero de Datos (Stack Pointer).

4.1.2.2 Memoria EPROM Interna del Microcontrolador MCS 8751.

El microcontrolador MCS 8751 contiene internamente 4 Kbytes de memoria EPROM para almacenar el programa desarrollado; se ha definido la siguiente configuración:

- 0000H – 0030H Espacio reservado para atención de interrupciones.
- 0031H – 0EFFH Espacio para almacenar el programa de administración del microcontrolador.
- 0FF0H – 0FFFH Espacio para almacenar Tablas y Datos.

En el diseño actual se usará la memoria EPROM interna del microcontrolador, sin embargo se puede usar memoria EPROM externa y cambiar el microcontrolador por un microprocesador de la serie MCS 8031; en este caso se debe mantener la configuración de la memoria EPROM.

4.1.2.3 Memoria EPROM Externa al Microcontrolador MCS 51

En el caso de usar un microprocesador de la serie MCS 51 es necesario contar con una memoria EPROM externa donde se pueda almacenar el programa de administración del microprocesador; también es posible utilizarla cuando la capacidad de memoria supera los 4 Kbytes de memoria con que cuentan los microcontroladores MCS 8751; es posible ampliar la memoria EPROM cambiando el microcontrolador por el MCS 8752 que cuenta con 8 Kbytes de memoria EPROM interna. En el PLC desarrollado se ha dejado disponible en hardware la máxima capacidad de memoria, la selección de memoria interna o externa se hace cambiando el puente correspondiente a JP1 en la tarjeta de control principal.

4.1.2.4 Memoria RAM externa al Microcontrolador MCS 8751.

Para almacenar los datos correspondientes a las aplicaciones desarrolladas para el microcontrolador se dispone de 64 Kbytes de memoria RAM externa; sin embargo esta capacidad es muy grande para la mayoría de las aplicaciones que

se desea controlar con el PLC diseñado, por ello se ha limitado la memoria RAM externa a únicamente 1 Kbyte, distribuida de la siguiente manera:

- 0000H – 0003H Localidades para almacenar el tamaño de la memoria RAM utilizada.
- 0004H – 039FH Localidades de memoria para almacenar programas de aplicación.
- 03A0H – 03BFH Localidades de memoria para procesar temporizados ON DELAY.
- 03C0H – 03FFH Localidades de memoria para procesar temporizados OFF DELAY.
- 03F0H – 03FFH Localidades de memoria para procesar Contadores.

Con estas limitantes van a existir 924 bytes de memoria RAM efectivos para programas introducidos por el usuario, esta capacidad es suficiente para desarrollar aplicaciones considerando que cada instrucción en su mayoría tiene únicamente 2 bytes.

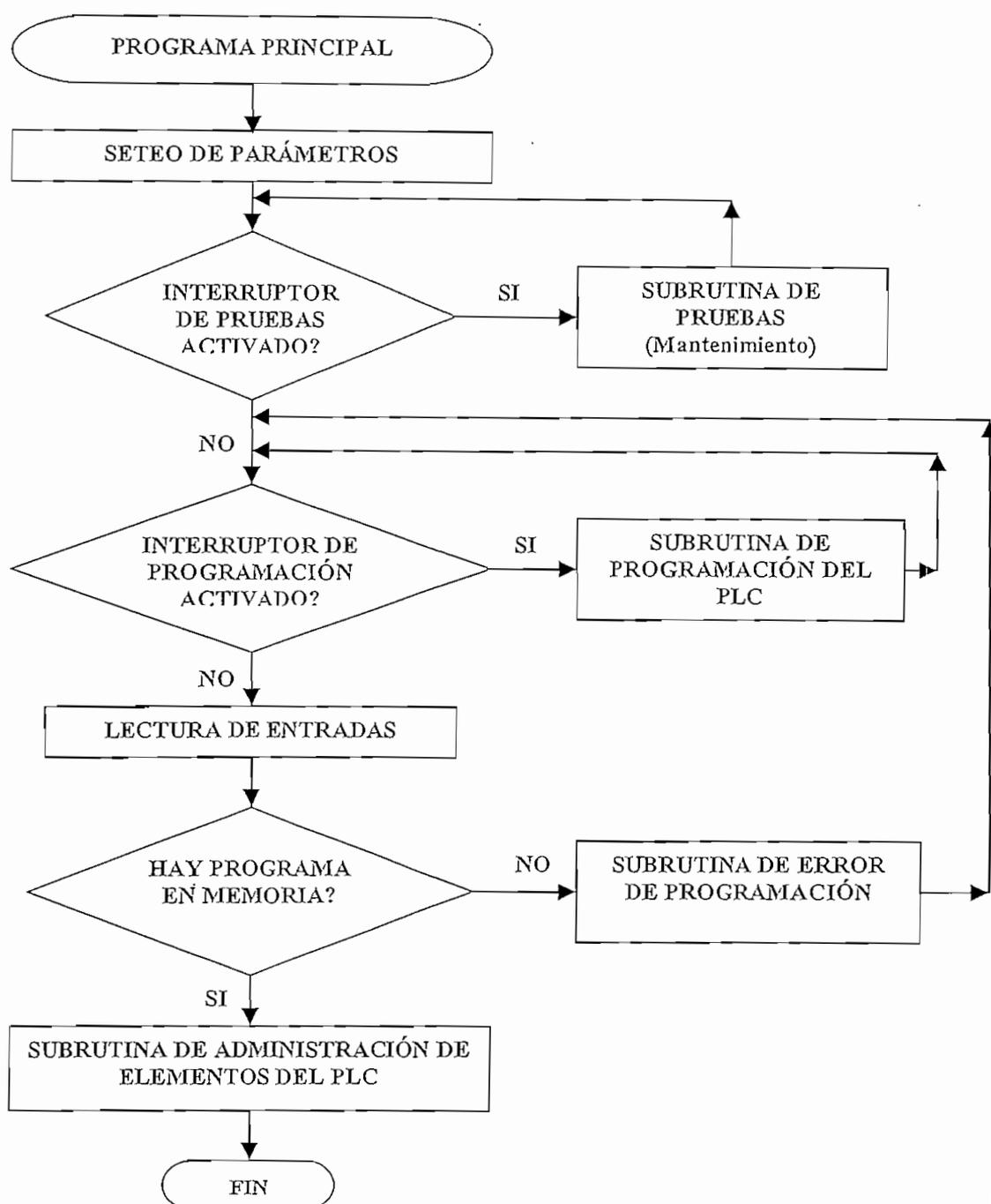
4.1.3 PROGRAMA PARA EL MICROCONTROLADOR DE LA SERIE MCS 51.

El microcontrolador de la serie MCS51 desarrollará tres tareas principales e independientes:

- Administración del PLC.
- Tratamiento de datos.
- Pruebas.

Las tres tareas principales se hallan a su vez procesadas por un programa central secuencial cuyo diagrama de flujo se indica en la **Figura 4.1**.

Figura 4.1 Diagrama de Flujo de Programa Principal



El programa principal setea los parámetros de control para el microcontrolador, chequea permanentemente el estado de los interruptores de operación y dependiendo del estado de éstos interruptores ingresa a una de las tres tareas incluidas en el programa.

Cuando no hay un programa almacenado en la memoria RAM o cuando el PLC está mal programado, se indica visualmente una señal de error en la parte frontal del PLC.

4.1.4 SUBROUTINA DE ADMINISTRACIÓN DEL PLC

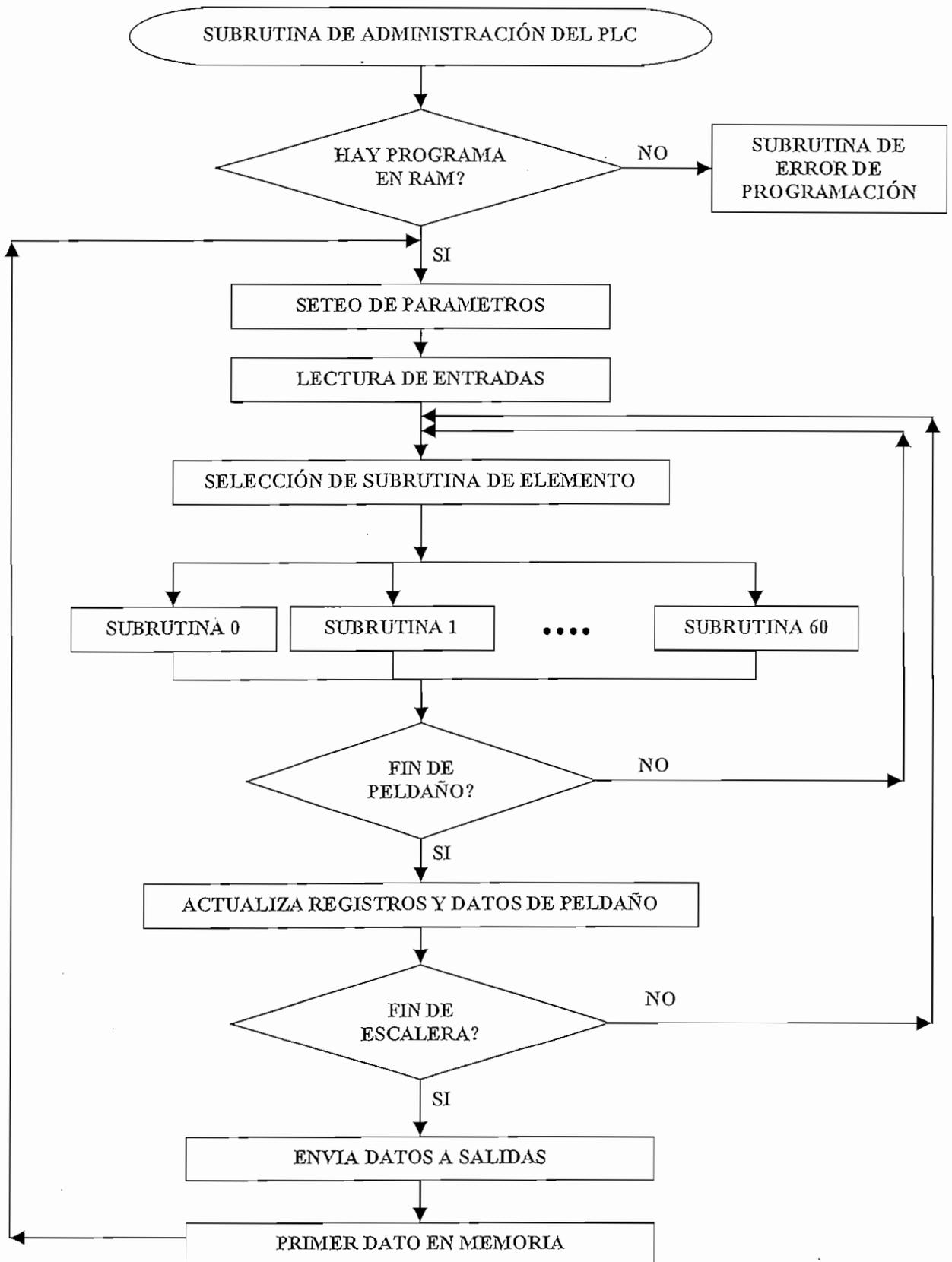
El programa de administración contiene todos los elementos en forma de subrutinas que van a ser escogidos por el programador para dar la lógica de funcionamiento del PLC.

4.1.4.1 Descripción del Funcionamiento

Este programa lee secuencialmente la memoria RAM empezando en la posición 0004H, en donde, si existe algún programa en memoria el código almacenado en esa localidad debe ser el 99H. Luego hace un barrido de las entradas, limpia los registros de funciones especiales, limpia las localidades de memoria donde se almacenan los datos de proceso, setea todos los parámetros de funcionamiento y acciona la interrupción interna T0 como contador con auto recarga para la administración de los temporizados On y Off Delay.

En la **Figura 4.2** se indica el diagrama de flujo de la subrutina de administración del Controlador Lógico Programable.

Figura 4.2 Diagrama de Flujo de Administración de Elementos.



Las subrutinas de elementos se dividen en tres grupos:

- Subrutinas de elementos pasivos.
- Subrutinas de elementos activos.
- Subrutinas de elementos auxiliares de lectura, escritura y elementos de enlace.

Los elementos pasivos programados son:

- Contactos Abiertos y Cerrados de Entradas
- Contactos Abiertos y Cerrados de Salidas
- Contactos Abiertos y Cerrados de Temporizados On Delay
- Contactos Abiertos y Cerrados de Temporizados Off Delay
- Contactos Abiertos y Cerrados de Contadores
- Contactos Abiertos y Cerrados de Relés Internos
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Entradas.
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Salidas.
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Temporizados On Delay.
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Temporizados Off Delay.
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Contadores.
- Funciones lógicas AND y OR de Contactos Abiertos y Cerrados de Relés Internos.

Los elementos activos programados:

- Bobina de Relé interno.
- Bobina de Salida.

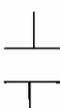
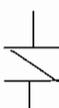
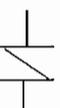
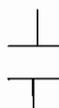
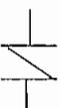
- Bobina de Temporizado On Delay.
- Bobina de Temporizado Off Delay.
- Bobina de Contador.
- Bobina de Reset de Contador.

Los elementos de escritura, lectura y enlace programados son:

- Funciones Lógicas AND y OR.
- Lectura de Entradas.
- Preparación de Nuevo Peldaño de Programa.
- Transmisión de Datos a las Salidas.

A cada subrutina de elementos pasivos, activos o auxiliares se accesa mediante un número hexadecimal que representa un código de operación. Sus códigos, así como símbolos, números de bytes y mnemónicos se resumen en la **Tabla 4.2**.

Tabla 4.2
LISTADO DE COMANDOS Y MNEMÓNICOS DEL PLC

LECTURAS DE ENTRADA DESDE PÓRTICO							
Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
Entrada Normalmente Abierta		LD	8	01H	00...07H	-	2
Entrada Normalmente Cerrada		LDNOT	8	02H	00...07H	-	2
AND de Entrada		AND	8	03H	00...07H	-	2
AND NOT de Entrada		ANDNOT	8	04H	00...07H	-	2
OR de Entrada		OR	8	05H	00...07H	-	2
OR NOT de Entrada		ORNOT	8	06H	00...07H	-	2
CONTACTOS DE SALIDA							
Contacto salida Normalmente Abierto		CSAL	8	07H	00...07H	-	2
Contacto salida Normalmente Cerrado		CNOSAL	8	08H	00...07H	-	2

Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
AND Contacto Salida		ANDCSAL	8	09H	00...07H	-	2
AND NOT Contacto Salida		ANDNOTS	8	10H	00...07H	-	2
OR Contacto Salida		ORCSAL	8	11H	00...07H	-	2
OR NOT Contacto Salida		ORNOTS	8	12H	00...07H	-	2

CONTACTOS DE RELÉS INTERNOS

Contacto Abierto de Relé Interno		RD1	64	13H	00...3FH	-	2
Contacto Cerrado de Relé Interno		RDNOT	64	14H	00...3FH	-	2
AND Relé Interno		ANDRD	64	15H	00...3FH	-	2
AND NOT Relé Interno		ARDNOT	64	16H	00...3FH	-	2
OR Relé Interno		ORRD	64	17H	00...3FH	-	2

Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
OR NOT Relé Interno		ORRDNOT	64	18H	00...3FH	-	2

ELEMENTOS AUXILIARES

Fin de peldaño		ESCAL	1	19H	-	-	1
AND lógico para elementos pasivos		ANDLD	1	20H	-	-	1
OR lógico para elementos pasivos		ORLD	1	21H	-	-	1

CONTACTOS DE TEMPORIZADO ON DELAY

Contacto Temporizado On Delay Normalmente Abierto		CTON	16	22H	00...0FH	-	2
Contacto Temporizado On Delay Normalmente Cerrado		CTONC	16	23H	00...0FH	-	2
AND Temporizado On Delay		ANDTON	16	24H	00...0FH	-	2
AND NOT Temporizado On Delay		ANDNTON	16	25H	00...0FH	-	2
OR Temporizado On Delay		ORTON	16	26H	00...0FH	-	2

Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
OR NOT Temporizado On Delay		ORNTON	16	27H	00...0FH	-	2

CONTACTOS DE TEMPORIZADO OFF DELAY

Contacto Temporizado Off Delay Normalmente Abierto		CTOFF	16	28H	00...0FH	-	2
Contacto Temporizado Off Delay Normalmente Cerrado		CTOFFC	16	29H	00...0FH	-	2
AND Temporizado Off Delay		ANDTOFF	16	30H	00...0FH	-	2
AND NOT Temporizado Off Delay		ANDNTOFF	16	31H	00...0FH	-	2
OR Temporizado Off Delay		ORTOFF	16	32H	00...0FH	-	2
OR NOT Temporizado Off Delay		ORNTOFF	16	33H	00...0FH	-	2

CONTACTOS DE CONTADOR

Contacto Normalmente Abierto de Contador		CONAB	16	36H	00...0FH	-	2
Contacto Normalmente Cerrado de Contador		CONCE	16	37H	00...0FH	-	2

Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
AND Contador		ANDAB	16	38H	00...0FH	-	2
AND NOT Contador		ANDCE	16	39H	00...0FH	-	2
OR Contador		ORCO	16	40H	00...0FH	-	2
OR NOT Contador		ORNOCO	16	41H	00...0FH	-	2

ELEMENTOS ACTIVOS

Bobina de Relé Interno		BRELE	64	42H	00...3FH	-	2
Bobina de Salida		BSALIDA	8	43H	00...07H	-	2
Bobina Temporizado On Delay		BTEMPON	16	44H	00...0FH	0000..FFFF	4
Bobina Temporizado Off Delay		BTEMOFF	16	45H	00...0FH	0000..FFFF	4
Bobina de Contador		CONTADOR	16	46H	00...0FH	0000..FFFF	4

Comando	Símbolo	Código ó Mnemónico	Número de Elementos	Mnemónico Hexadecimal	Número Hexadecimal	Tiempo o Cuenta	Número de Bytes
Reset Bobina Contador		RESETCONTADOR	16	47H	00...0FH	-	2

CÓDIGOS DE RESERVA

Disponible	-	-	-	48H	-	-	-
Disponible	-	-	-	49H	-	-	-
Disponible	-	-	-	50H	-	-	-
Disponible	-	-	-	51H	-	-	-
Disponible	-	-	-	52H	-	-	-
Disponible	-	-	-	53H	-	-	-
Disponible	-	-	-	54H	-	-	-
Disponible	-	-	-	55H	-	-	-
Disponible	-	-	-	56H	-	-	-
Disponible	-	-	-	57H	-	-	-
Disponible	-	-	-	58H	-	-	-
Disponible	-	-	-	59H	-	-	-

ELEMENTOS AUXILIARES

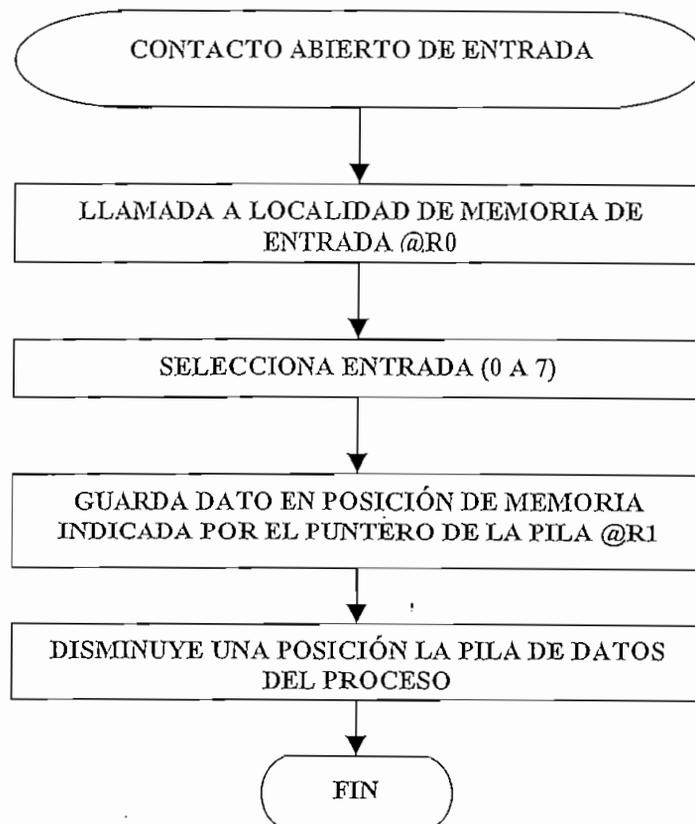
Fin de programa		TERMINO	1	60H	-	-	1
-----------------	---	---------	---	-----	---	---	---

A continuación se detallan algunos de los diagramas de flujo de elementos pasivos, activos y auxiliares utilizados; los elementos no detallados tienen un diagrama de flujo similar con diferencia únicamente en la localidad de memoria a la que accede identificada en la tabla anterior.

Contactos de Entradas.

Hay 8 entradas físicas en el PLC. La subrutina toma las 8 entradas almacenadas en un byte (7FH) de la memoria RAM interna, separa la señal de entrada seleccionada e inmediatamente el dato es almacenado en la posición de memoria dada por el registro R1 que actúa como puntero de la pila de datos de proceso (desde posición 5FH). Este es un proceso de carga de datos en la posición de operaciones, una vez que el dato es almacenado, el puntero de la pila se desplaza una posición.

Figura 4.3 Subrutina de Contacto Abierto de Entrada

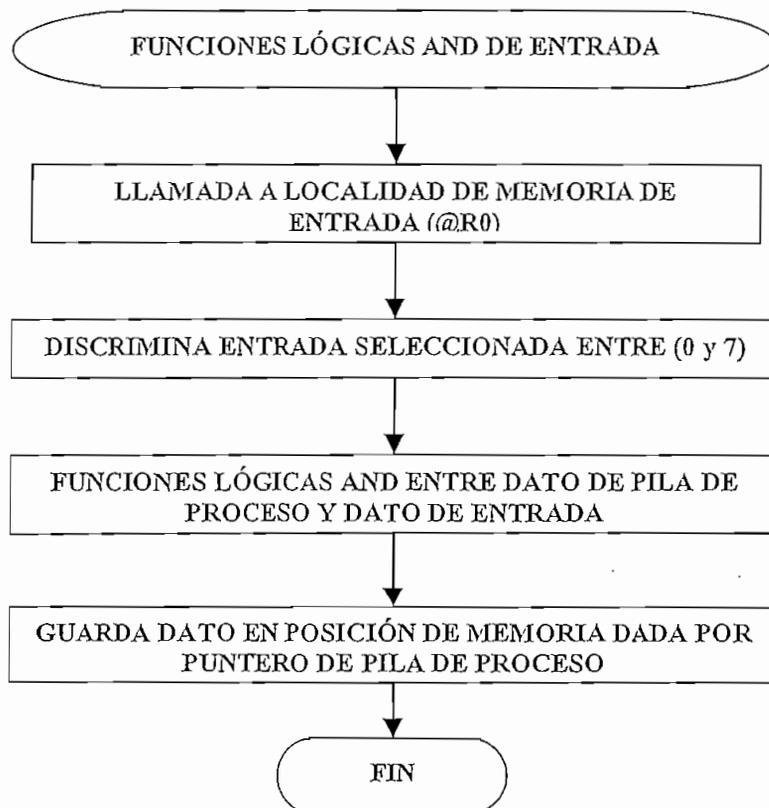


El diagrama de flujo del contacto cerrado de entrada es similar al indicado para el contacto abierto, la diferencia está en que antes de ser almacenado el dato es complementado. El proceso de carga de datos produce desplazamiento de la pila de datos; sin embargo al empezar un nuevo peldaño, el puntero de datos vuelve a la posición inicial de la pila de procesamiento de datos.

Funciones Lógicas AND de Entradas.

En las funciones lógicas AND de entradas se toma el dato almacenado en la pila de datos de proceso y se ejecuta la operación lógica AND con el dato de entrada disponible en el puntero de la pila actualmente. Una vez que la operación es realizada, no desplaza una localidad de memoria sino que se almacena en la posición de memoria a la que apunta el puntero.

Figura 4.4 Subrutina de Función Lógica AND de Entrada.

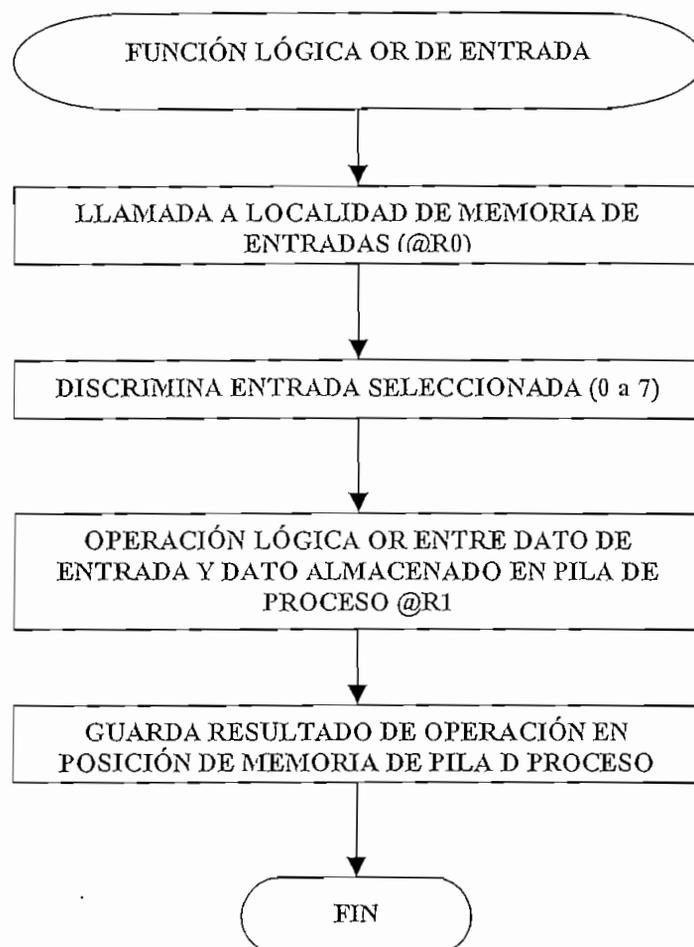


Esta función se encuentra luego de un elemento de carga en cualquier posición dentro de los peldaños, pero no al inicio.

El diagrama de flujo de la función lógica AND NEGADA es similar al mostrado para la función lógica con la diferencia que el dato antes de ser almacenado es complementado. La función lógica AND NEGADA equivale a un contacto cerrado de un elemento de entrada pasivo.

Funciones Lógicas OR de Entradas.

Figura 4.5 Subrutina de Función Lógica OR de Entrada.



Se usa la función lógica OR para operar elementos que se encuentran en paralelo. La operación lógica OR se realiza entre dos elementos, el que ingresa y el que se encuentra almacenado en la pila de proceso. Esta operación no produce desplazamiento de datos en la pila de proceso.

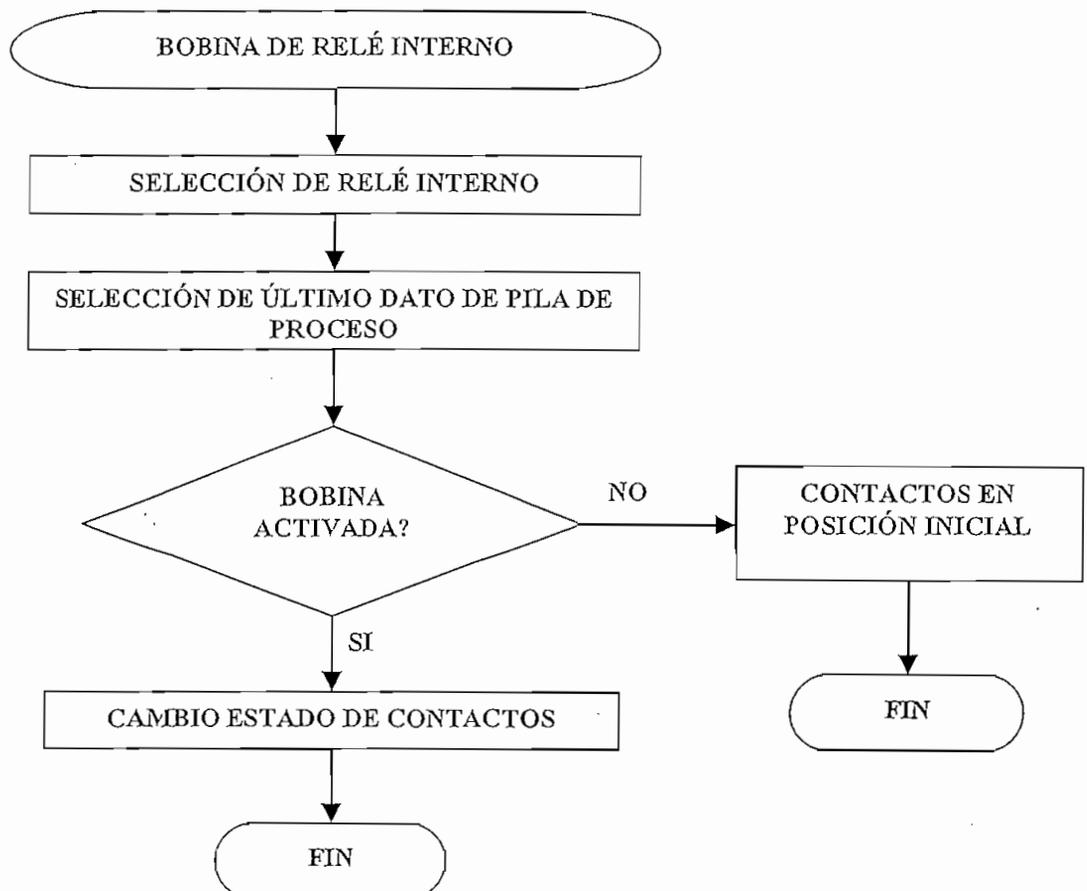
La función lógica OR NEGADA que equivale a un contacto normalmente cerrado en paralelo tiene un diagrama de flujo similar al indicado para la función lógica OR; la diferencia está en que el dato antes de ser almacenado es complementado.

Bobina de Relé Interno.

La bobina de relé interno es un elemento lógico colocado al final de un peldaño, chequea el estado del dato de la pila de proceso. Si luego de la lógica de control el elemento final de la pila es un uno lógico, activa la bobina y cambia el estado de los contactos. Se encuentran almacenadas en las posiciones (50H a 57H).

En la **Figura 4.6** se muestra el diagrama de flujo de la subrutina correspondiente a los relés internos del PLC.

Figura 4.6 Subrutina de Bobina de Relé



Bobina de Salida.

El diagrama de flujo para una bobina de salida es similar a la bobina de relé interno con la diferencia que hay únicamente 8 salidas físicas y se almacenan en una única localidad de memoria (7FH).

Bobina de Temporizado On Delay.

Como se indicó, un temporizado On Delay cambia el estado de sus contactos un tiempo después de que su bobina ha sido energizada.

Si la energía lógica se elimina antes que el tiempo de retardo haya transcurrido, el ciclo empieza nuevamente. El diagrama de flujo para el temporizado On Delay se indica en la **Figura 4.7**.

Bobina de Temporizado Off Delay

El temporizado Off Delay cambia el estado de sus contactos inmediatamente después que ha sido energizada su bobina, si la energía se reconecta antes que actúe, el ciclo empieza nuevamente. El temporizado off delay demora un tiempo de retardo programado para volver sus contactos a su posición inicial. El diagrama de flujo para el temporizado off delay se indica en la **Figura 4.8**.

Los temporizados han sido diseñados de manera que tengan un reloj interno, este reloj disminuye la cuenta con la que cada temporizado ha sido programado cada 40 milisegundos. El diagrama de flujo de la rutina de actualización del tiempo se detalla en la **Figura 4.9**.

Bobina de Contador

El contador de eventos establece la comparación de un número de pulsos que ha sido programado por el usuario con el número de pulsos producto de un evento programado, al tener el mismo valor cambia el estado de los contactos del contador, su diagrama de flujo se muestra en la **Figura 4.10**.

Figura 4.7 Diagrama de Flujo de Bobina de Temporizado On Delay

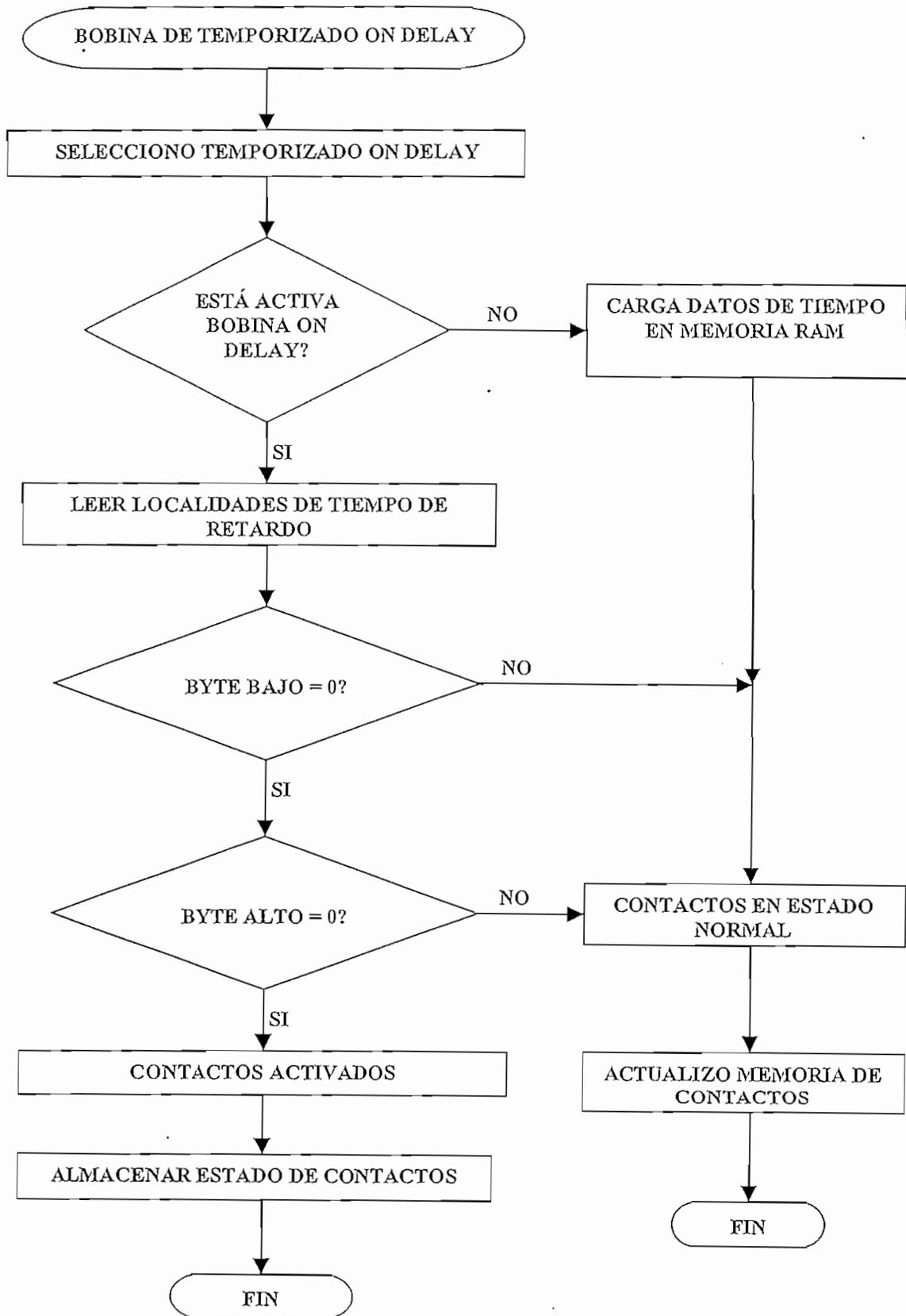


Figura 4.8 Diagrama de Flujo de Bobina de Temporizado Off Delay.

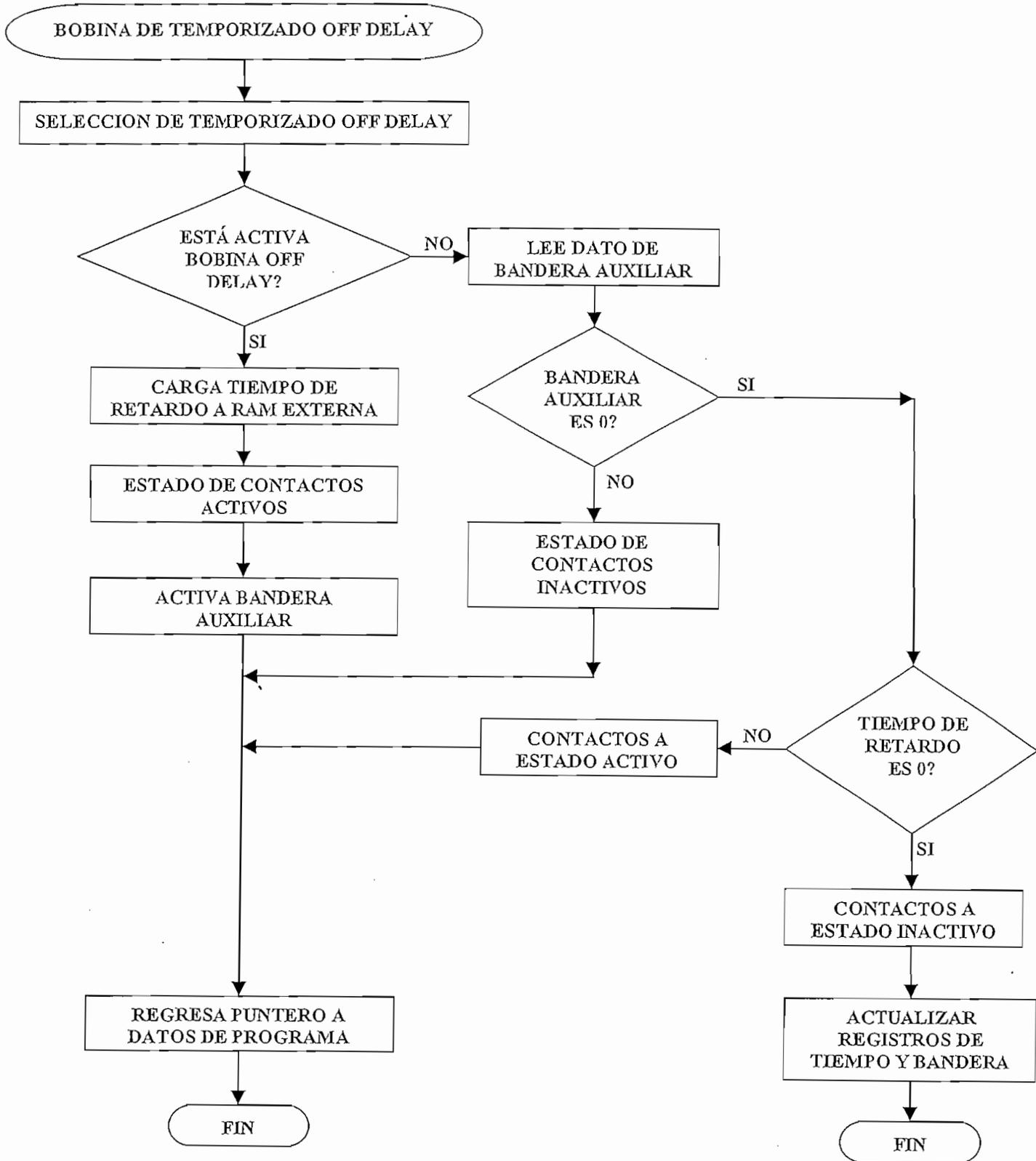


Figura 4.9 Diagrama de Flujo de Actualización de Tiempo.

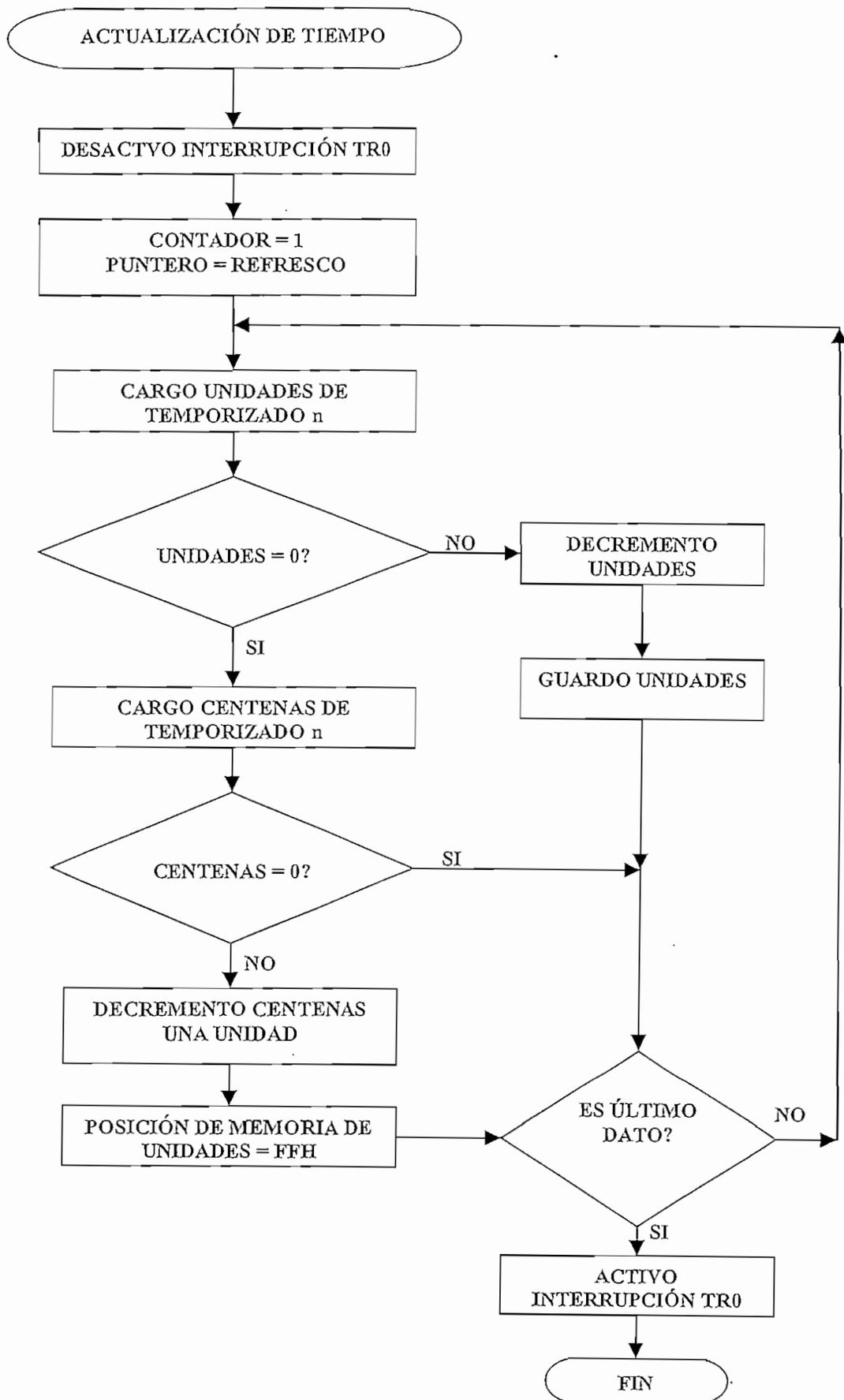
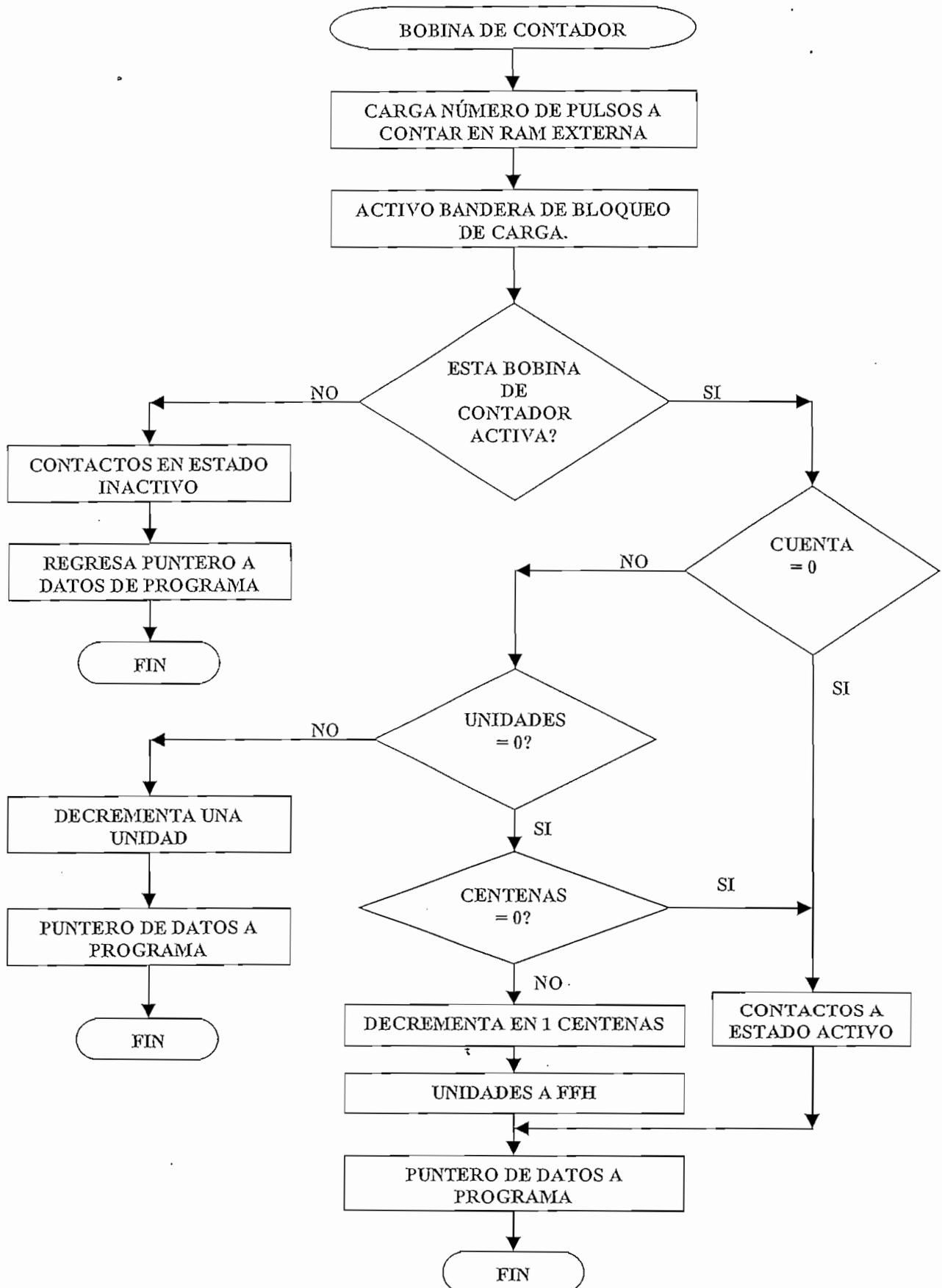


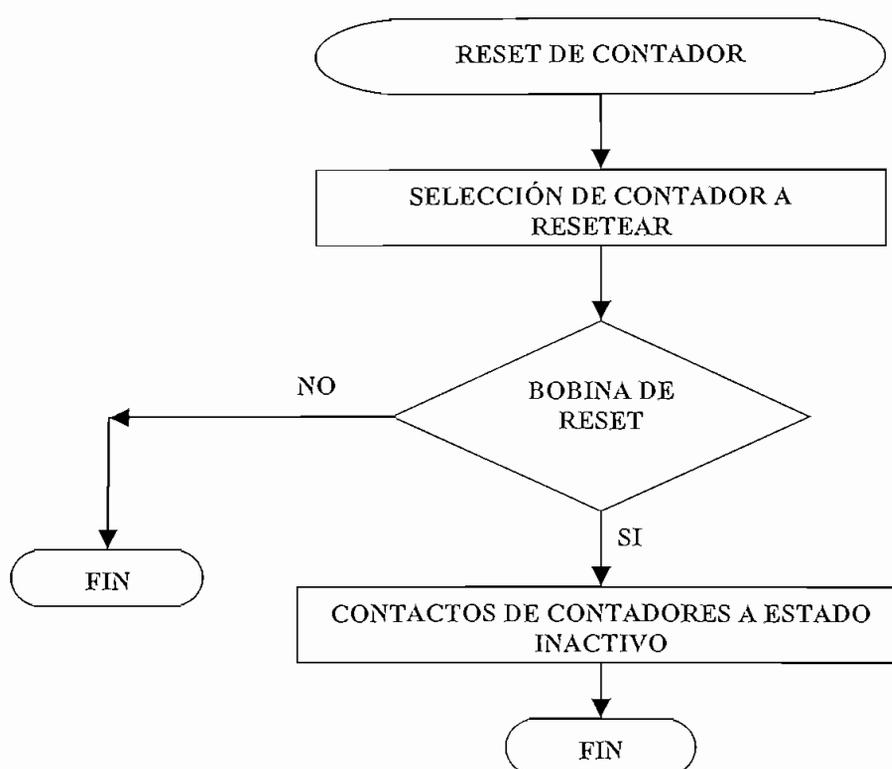
Figura 4.10 Diagrama de Flujo de Bobina Contador.



Bobina Reset de Contador

Para regresar a condiciones iniciales el contador, es necesario contar con una señal de reset que se ha simbolizado como una bobina que al ser activada regresa los contactos de un contador a su posición inicial. Su diagrama de flujo se muestra en la **Figura 4.11**.

Figura 4.11 Diagrama de Flujo de Bobina Reset de Contador.

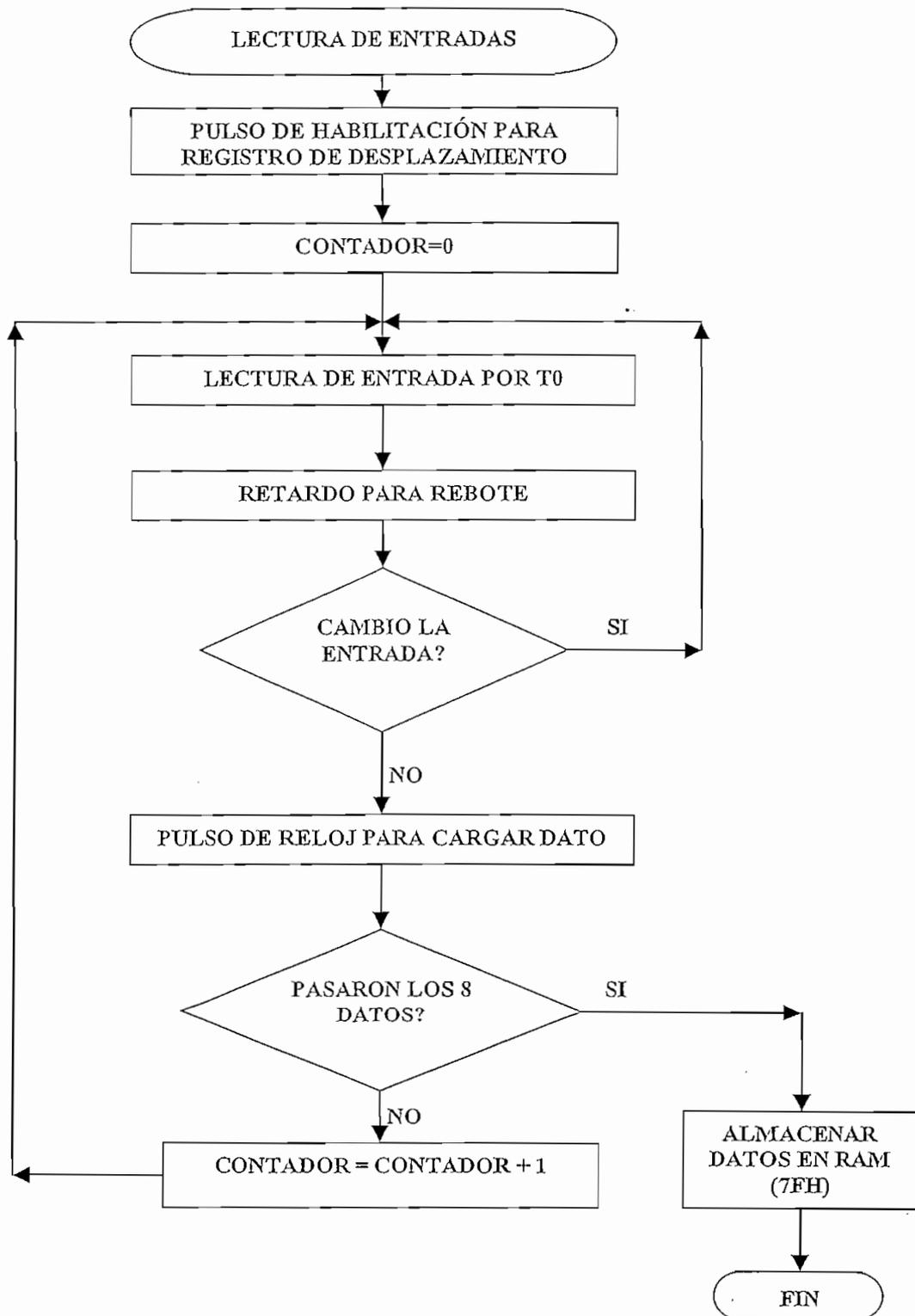


Lectura de Entradas.

Las entradas son leídas una vez en cada ejecución de programa, esto significa que el pulso de una entrada para que sea detectado por el programa debe tener un tiempo superior al tiempo de ejecución del programa. La subrutina de lectura de datos, envía un pulso de habilitación al registro de desplazamiento 4021 para tener los datos en la salida del registro de manera serial, al mismo tiempo procede a la lectura de los datos ingresados al microcontrolador por el pin T0, este dato de entrada es analizado por un lapso de tiempo para evitar leer. Una vez que todos

los datos de entrada han sido leídos, los datos de las entradas son almacenados en la posición de memoria 7FH de la memoria RAM interna del microcontrolador.

Figura 4.12 Diagrama de Flujo de Lectura de Entradas



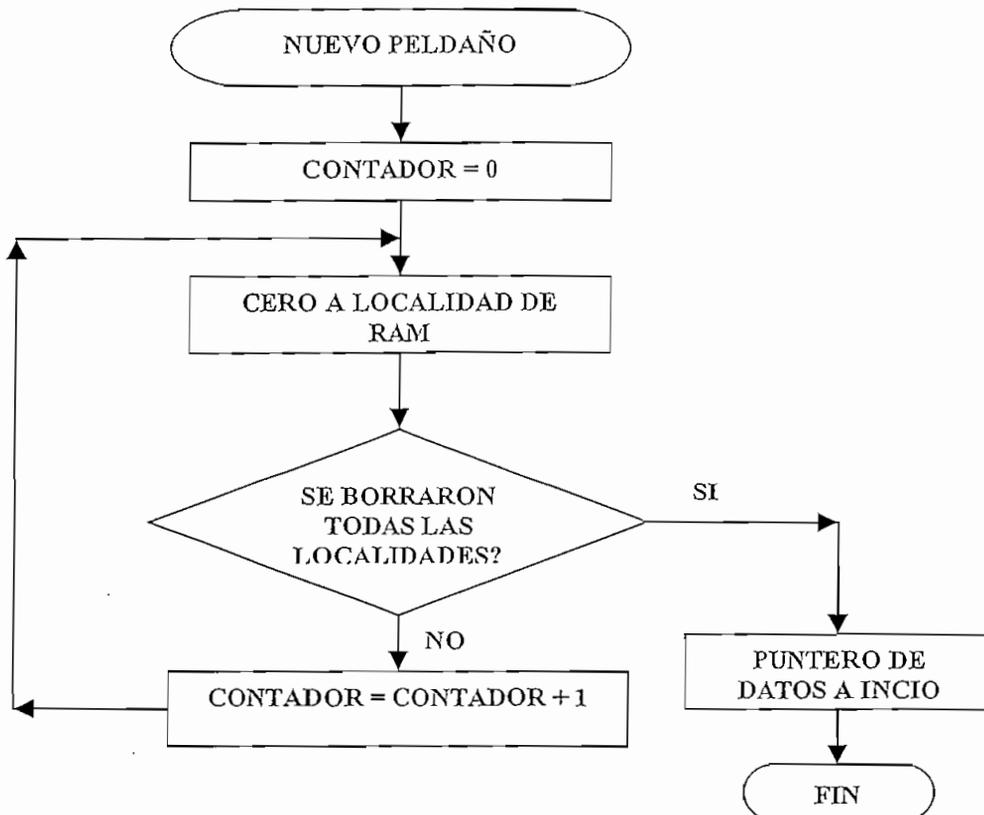
Funciones Lógicas AND y OR auxiliares.

Toman dos datos de la pila de proceso y los enlazan mediante una de las dos funciones lógicas: AND u OR; el resultado se almacena en la posición de memoria más alta de la pila de datos del proceso. Se diferencian de las funciones lógicas AND y OR de entradas cuyo diagrama de flujo fue indicado anteriormente, en que toma los dos datos de la pila de proceso mientras la subrutina de entradas opera el dato de entrada con el dato correspondiente de la pila de proceso.

Preparación de Memoria para Nuevo Peldaño.

La subrutina de preparación para un nuevo peldaño limpia las localidades de memoria correspondientes a la pila de datos de proceso. Esta subrutina se activa luego de que los datos han sido procesados en un peldaño y antes de enviar los datos a las salidas.

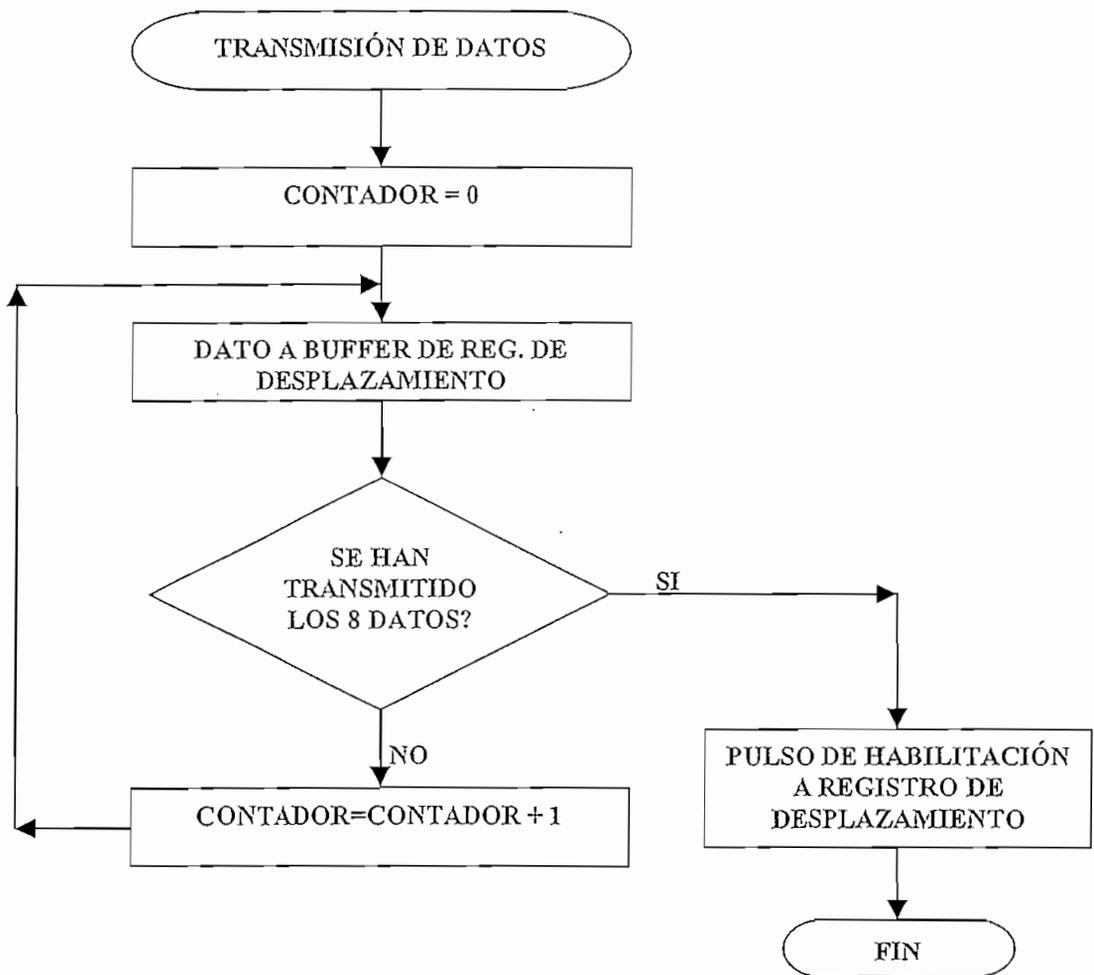
Figura 4.13 Diagrama de Flujo para Preparación de Nuevo Peldaño.



Transmisión de Datos a las Salidas

Una vez que el programa ha ejecutado todas las acciones de control, envía los datos a las salidas; para ello primero envía pulsos de reloj de habilitación al registro de desplazamiento 4094, una vez con los 8 datos en el buffer del registro de desplazamiento, el programa envía un pulso de habilitación para que los datos sean enviados a las salidas.

Figura 4.14 Diagrama de Flujo de Transmisión de Datos



4.1.5 SUBROUTINA DE TRATAMIENTO DE DATOS

La subrutina de tratamiento de datos se encargará de enviar los datos de un archivo hexadecimal generado por el programa escalera del PC al PLC. Los datos son enviados a través del puerto serial en formato 8-N-1 a 9600 bps. El baud rate y formato pueden ser alterados con pequeños cambios en la subrutina correspondiente a configuración del puerto. El programa diseñado ha sido realizado de la manera más simple posible.

Entre otras las subrutinas más importantes involucradas en el tratamiento de datos son:

- Envío de datos a la memoria RAM.
- Seteo del pòrtico de comunicaciones.
- Conversión de datos ASCII a Hexadecimal y viceversa.
- Lectura de memoria RAM.
- Chequeo de datos en memoria RAM y estado de errores.
- Retorno de información.
- Medición de la capacidad de memoria usada por el usuario.

Para el envío de datos a la memoria RAM se utilizará el código estándar de transmisión de datos diseñado por INTEL, el que se detalla a continuación:

4.1.5.1 Código de envío de Datos INTEL Estándar

La transmisión de información desde la computadora hacia cualquier micro procesador o dispositivo de memoria se lo realiza con formato INTEL estándar. La estandarización permite ampliar aplicaciones en el programa del PLC y dará facilidades de programación en trabajos posteriores.

El formato de transmisión es el siguiente:

:NN – DDDD – 00 - ##.....## - CHK

.....

:00000001FF

Donde:

- :.- Define el inicio de una línea de programa a enviar al micro controlador o micro procesador.
- NN.- Define el número de datos que se enviarán en esta secuencia.
- DDDD.- Indica la dirección inicial de envío de datos al microprocesador.
- 00.- Dos bytes 00 para espera de datos válidos y/o verificación de información.
- ##.- Los datos que van a ser enviados siempre en dos bytes.
- CHK.- Suma de todos los datos enviados, le programa chequea este valor para determinar que no existe ningún error en la transmisión; es conocido

Una vez que todos los datos han sido enviados línea por línea con este formato, se termina con una secuencia de dos puntos y 10 bytes (7 ceros, un 1 lógico y 3 números FFF hexadecimales).

:00000001FF.- Código que nos indica que la transmisión de datos han finalizado.

Al terminar el envío de datos el computador responde como dato final de recepción, con un check sum total que indica que la información ha sido enviada satisfactoriamente, luego de lo cual deja los dos puntos como indicación final para poder realizar otra operación.

4.1.5.2 Pórtico de comunicaciones.-

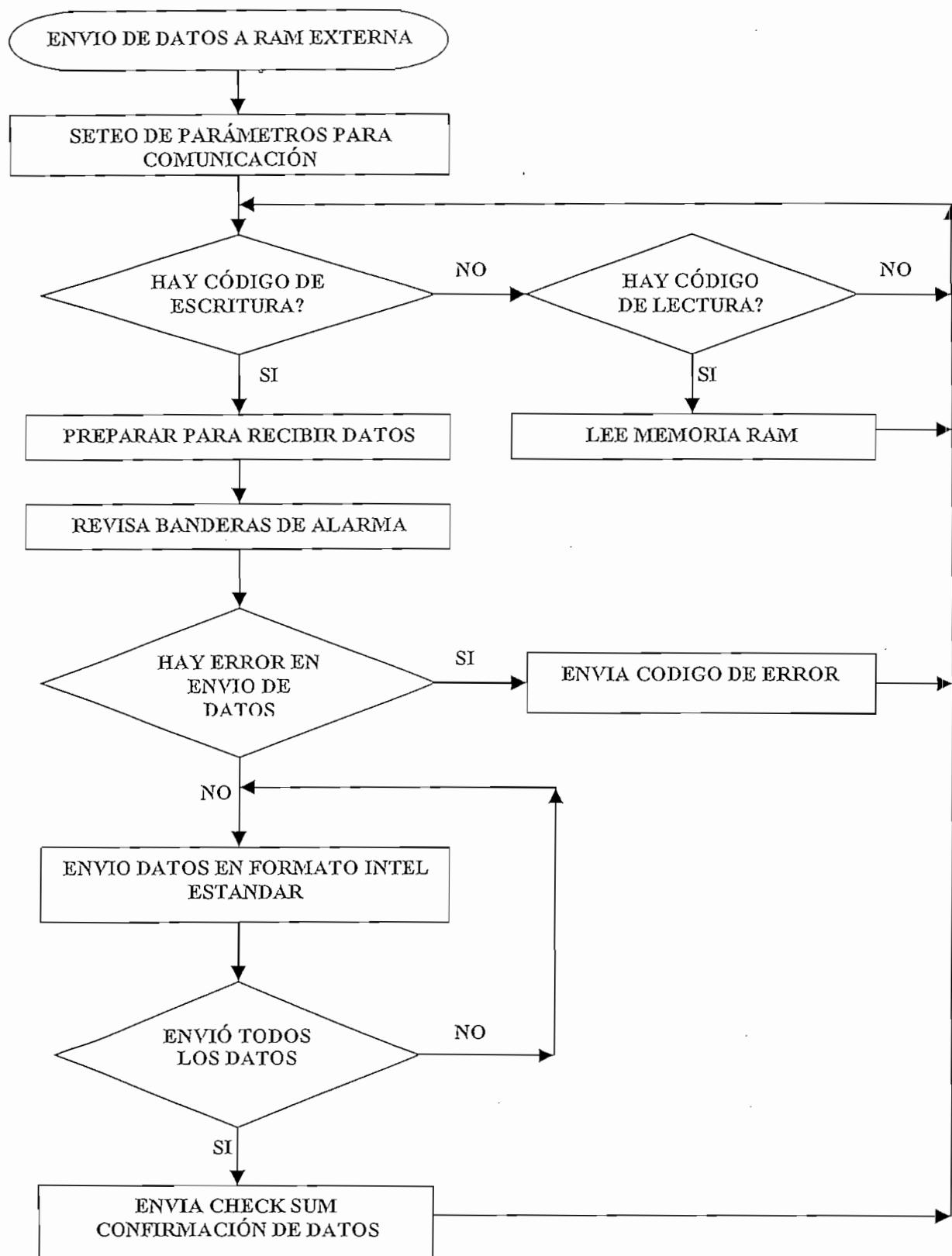
La comunicación del microcontrolador con la computadora se realizará por el pórtico serial RS232C disponible a una velocidad de transmisión de 9600 bps para tener una transmisión segura de datos.

No se ha definido otras velocidades de transmisión en función de que las comunicaciones en el presente programa son únicamente para programación de aplicaciones realizadas por el usuario; se deja abierta la posibilidad de una ampliación en un posterior trabajo, si se requiere hacer chequeo en tiempo real, es decir poder identificar la operación del PLC en el tiempo.

La comunicación sólo se necesita al programar el microcontrolador, ya que para el trabajo del programa en el PLC no es necesario que la computadora se halle en comunicación con el PLC.

Se mencionan algunas de las subrutinas que se realizaron para el programa de transmisión de datos de PC a PLC. Los diagramas de flujo para las subrutinas indicadas se detallan a continuación.

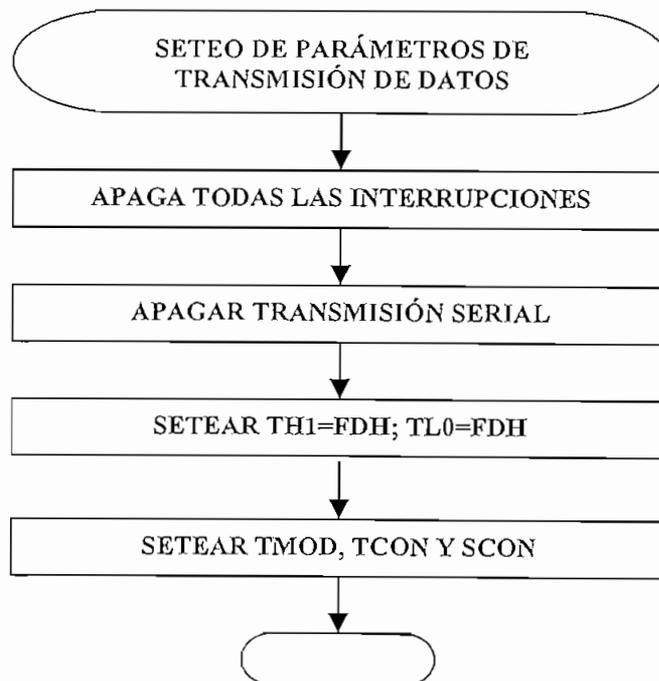
Figura 4.15 Diagrama de Flujo de Comunicación con la Memoria RAM



Seteo de Parámetros de Comunicación

El seteo de parámetros de comunicación permite enlazar la PC con el PLC a una velocidad de comunicación. Como se ha indicado sólo se ha diseñado para transmitir a 9600 bps, de ser necesario otras velocidades de transmisión se pueden cambiar los parámetros mostrados en esta subrutina. Los parámetros han sido calculados de acuerdo con las especificaciones dadas por el fabricante del microprocesador. La transmisión de datos en esta subrutina no usa la interrupción serial, se deja libre esta interrupción para ampliar las aplicaciones.

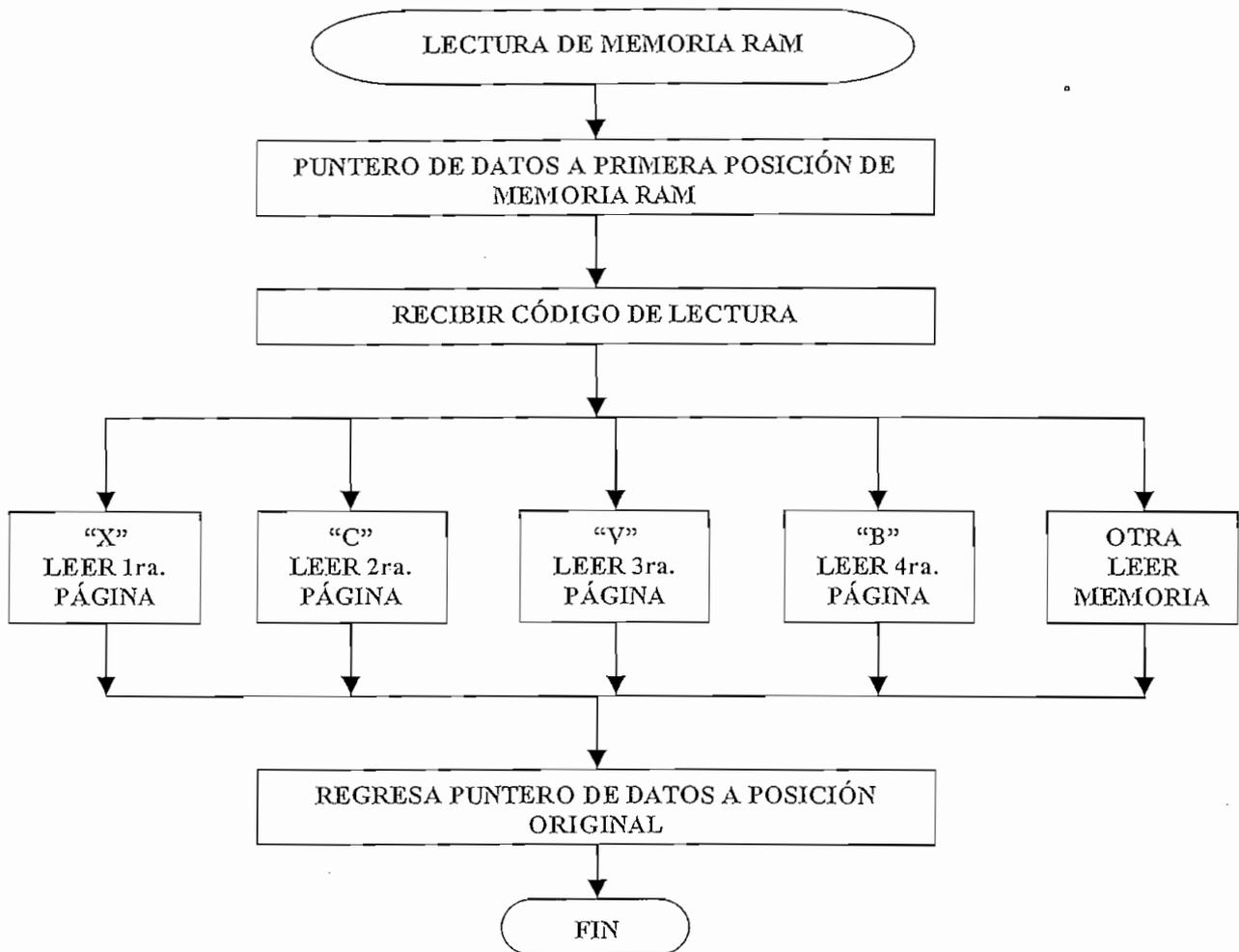
Figura 4.16 Seteo de Parámetros de Transmisión de Datos



Lectura de Memoria RAM

Permite leer los datos almacenados en la memoria RAM. Se ha dividido en 4 páginas de 256 bytes cada una de ellas a las cuales se puede acceder introduciendo el código de página respectivo. La lectura de datos compara los datos a ser enviados con los existentes en la ventana de ensamblado de la PC para verificar que el programa ensamblado es correcto.

Figura 4.17 Lectura de Memoria RAM

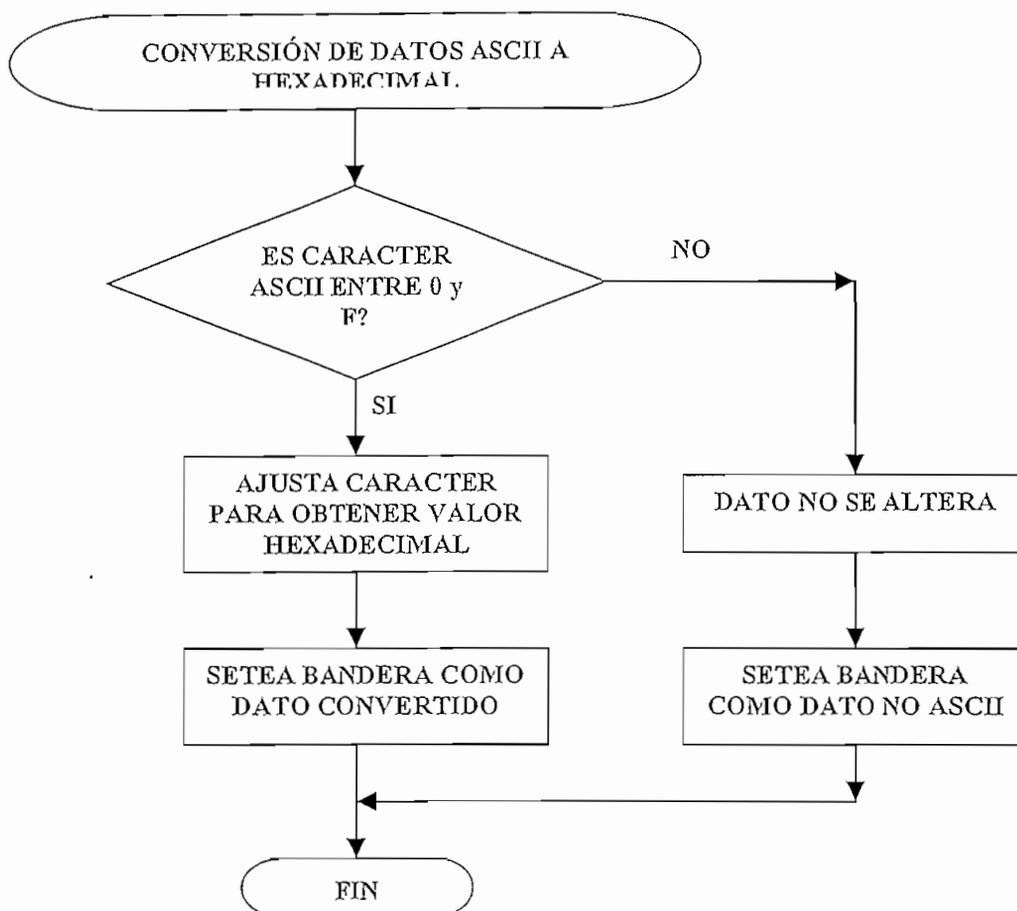


Conversión de Datos ASCII a Hexadecimal.

Esta subrutina mira si el caracter en el acumulador del microcontrolador es ASCII, para convertirlo en un caracter hexadecimal; la bandera Hexflag nos indica si el caracter fue convertido a hexadecimal. Otros caracteres diferentes de 0 a 9 y A a F son reorganizadas pero no convertidas.

Esta subrutina entra en operación cada vez que se necesita enviar datos de la PC al microcontrolador.

Figura 4.18 Conversión de Datos ASCII a Hexadecimal



Conversión de Datos Hexadecimal a ASCII.

Toma los datos almacenados en el acumulador del microcontrolador y los convierte a ASCII. Cada byte del microcontrolador contiene dos números a ser enviados al PC y sólo se convierten los números hexadecimales. Esta subrutina se usa cuando se necesita enviar los datos del PLC al PC. El diagrama de flujo es similar al mostrado para la conversión de datos de ASCII a Hexadecimal.

Chequeo de Datos en Memoria RAM y Estado de Errores.

Al final de un archivo enviado al PLC llega una etiqueta que indica que ha concluido la transmisión satisfactoriamente, sin embargo en caso de producirse algún error el programa lo detecta y envía uno de los siguientes códigos:

- 01 Caracteres no hexadecimales detectados en la línea de datos.
- 02 Mal tipo de registro encontrado.
- 04 Incorrecto check sum encontrado en una línea.
- 08 Ningún dato encontrado.
- 10 Sobreflujo por incremento de la dirección retornada a cero.
- 20 Dato de escritura RAM no verifica correctamente.

Si algún error ocurre cuando se está bajando el programa, se puede intentar nuevamente enviando un caracter de ESC. Cuando la cadena de datos enviados es correcta, el programa responde con el envío del check sum total del archivo, truncándolo siempre a 16 bits. Este valor se devuelve entre paréntesis.

Cada código de error no se halla definido como una sola subrutina, sino que actúa en el momento mismo que se encuentra con éste; por este motivo sólo se describe su función y no se detalla un diagrama de flujo.

Todas las subrutinas indicadas y las demás no detalladas se pueden analizar en el programa assembler del microcontrolador.

4.1.6 SOFTWARE PARA PRUEBAS.

Es necesario contar con algún tipo de herramienta para hacer mantenimiento del sistema, de manera que el encargado del equipo sea capaz de encontrar daños con bastante rapidez y dar una solución inmediata. Es por ello que se ha introducido un programa capaz de identificar el estado en el que se encuentran tanto las entradas como las salidas, para poder determinar si los errores son producto de la programación o del equipo en cuestión o errores de hardware; indirectamente se están probando otros elementos como el estado del microcontrolador y el pórtico de comunicaciones RS 232C tanto en transmisión como en recepción de datos.

El programa inicialmente pone a cero todas las salidas, pero para la secuencia de prueba de señales se debe tener en consideración que las salidas se hallen

físicamente desconectadas de elementos activos ya que pueden producirse accidentes al tener conectados elementos que manejen motores y/o algún elemento físico.

Se ingresa a la subrutina de pruebas haciendo un puente a 1 lógico en el pin P1.6, luego de pulsar un RESET general en el programa; se lo ha realizado de esta manera para que se eviten posibles errores de manipulación y se active por error esta subrutina.

En una primera instancia el programa chequeará independientemente cada salida de manera secuencial con aproximadamente dos segundos de diferencia entre una y otra.

Una vez que todas las salidas han sido chequeadas, envía por el pórtico de comunicaciones un mensaje escrito para indicar que éste se halla funcionando y de la misma manera informa al personal de mantenimiento que las comunicaciones trabajan.

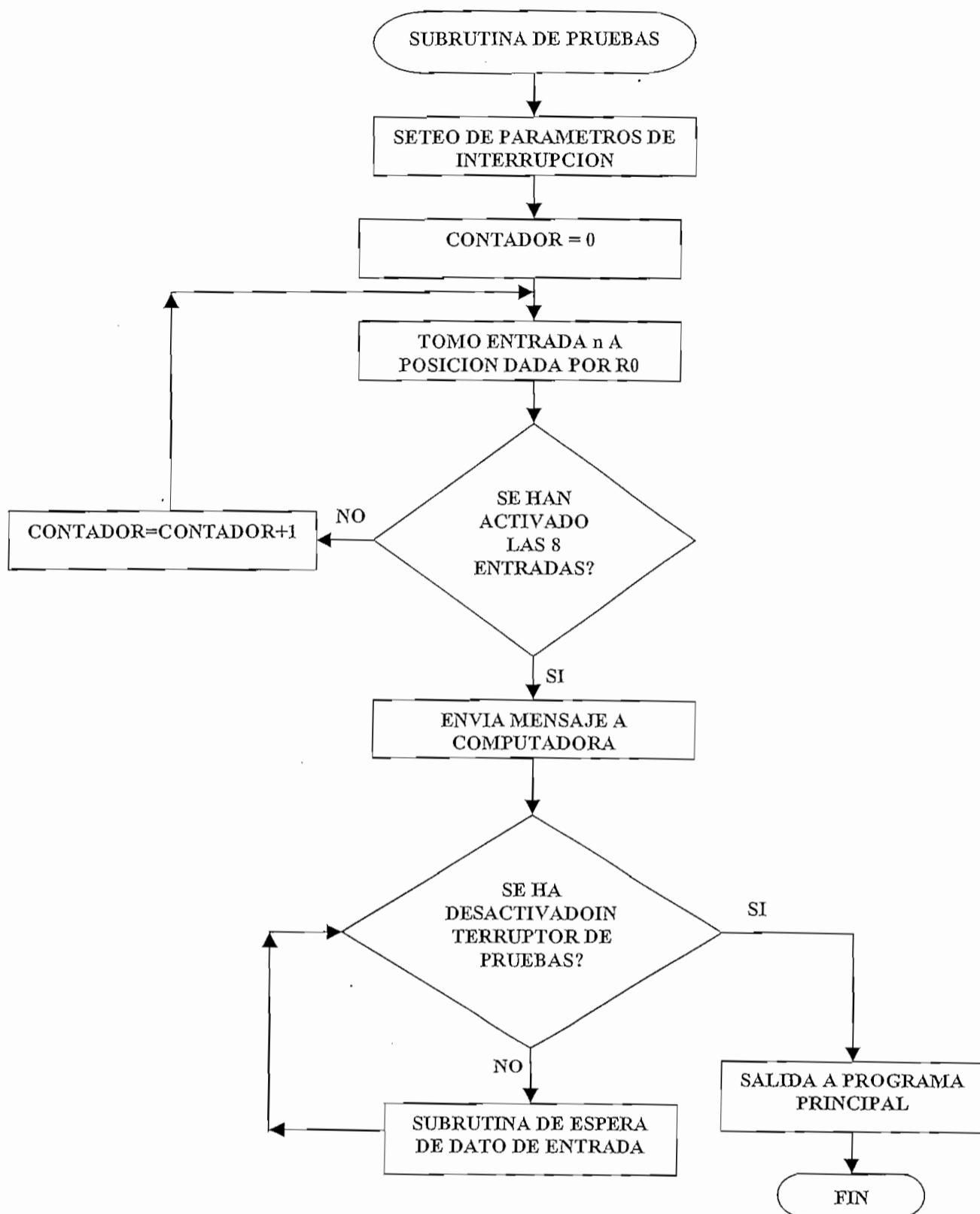
Posteriormente chequea todas las salidas simultáneamente por aproximadamente el mismo espacio de tiempo y las deja a todas desactivadas, pero listas para leer las entradas desde el programa principal.

Luego de esta secuencia se queda en espera para que el usuario active las entradas que responderán activando a su vez la salida correspondiente; se sale del estado de pruebas haciendo un puente entre el pin P1.6 y 0 lógico; luego de ejecutar un reset en el microcontrolador.

El programa de pruebas consta de los siguientes bloques:

- Seteo del pórtico de comunicaciones
- Prueba de salidas secuencial
- Prueba de entradas hechas por el usuario
- Prueba de comunicaciones

Figura 4.19 Diagrama de Flujo de Subrutina de Pruebas



Este programa utiliza las subrutinas del programa de administración de elementos del PLC.

4.2 DISEÑO DEL SOFTWARE PARA PC.

Para poder programar el PLC se ha desarrollado un software que permite al usuario diseñar el diagrama escalera que luego de ser ensamblado se envía al PLC a través del pórtico de comunicación serial.

4.2 DISEÑO DEL SOFTWARE PARA PC.

Para poder programar el PLC se ha desarrollado un software que permite al usuario diseñar el diagrama escalera que luego de ser ensamblado se envía al PLC a través del pòrtico de comunicaci3n serial.

El programa permitir3 desarrollar un diagrama de control gr3fico tipo escalera, el cual ser3 transformado en una matriz de datos conteniendo la informaci3n suficiente referente a cada elemento. Podr3 ser editado para insertar o eliminar elementos, imprimir la hoja de trabajo, transmitir datos, 3btener informaci3n del proceso de transmisi3n y estado de la memoria RAM y permitir3 ser almacenado en un archivo para su posterior uso.

Se usar3 el lenguaje de programaci3n VISUAL BASIC versi3n 5.0 que constituye una herramienta muy 3til para el desarrollo del software, precisamente porque permite establecer interfaces gr3ficas f3ciles de entender y manipular para los fines antes descritos, as3 como trabajar con la comunicaci3n serial.

Las especificaciones m3nimas para la utilizaci3n del programa son las siguientes:

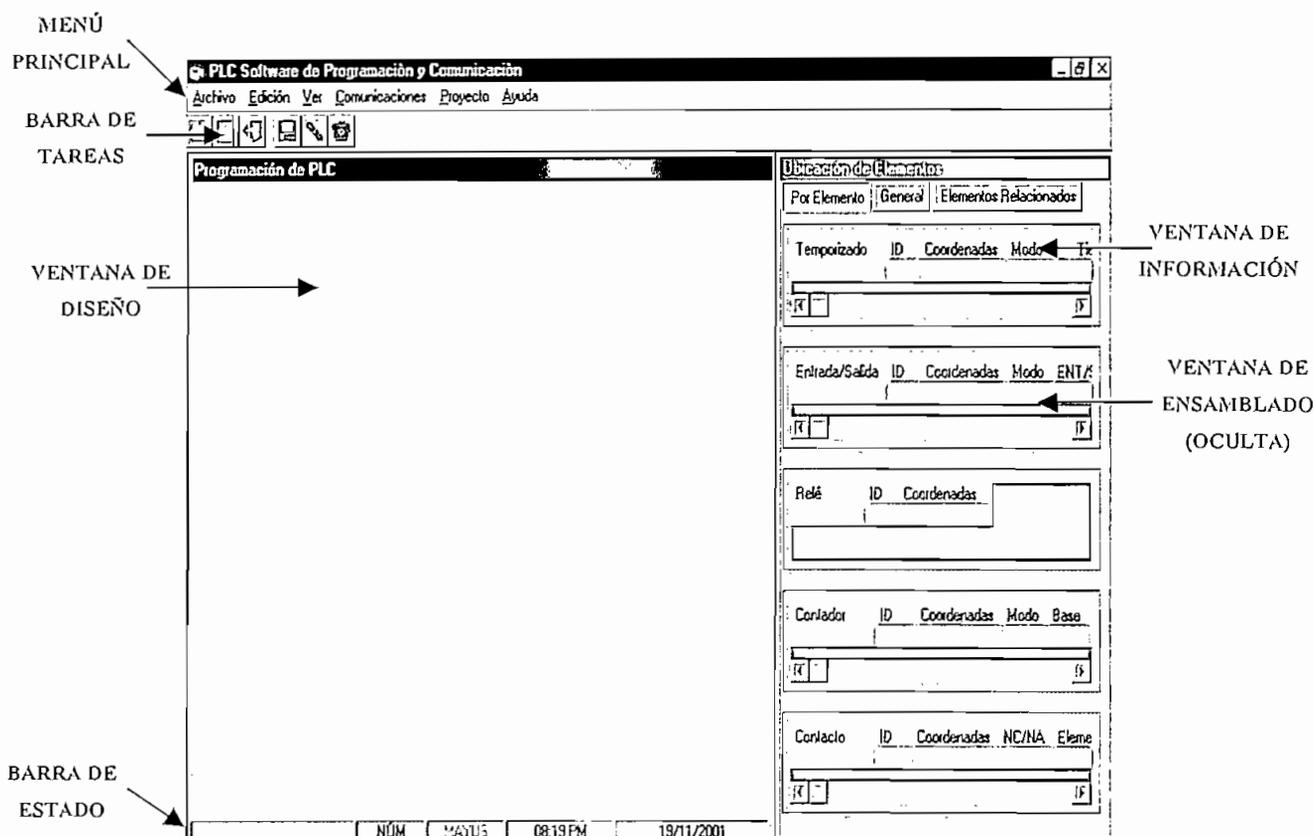
- Windows 95, Windows NT 3.5.1 o superior.
- Microprocesador 486 o superior.
- 133 MHz de velocidad de reloj o superior.
- M3nimo 10 Mbytes de espacio disponible en disco duro.
- Pantalla VGA o de mayor resoluci3n.
- 16 Mbytes de memoria RAM.
- Rat3n.
- Puerto serial de comunicaciones RS232C.
- Unidad de CD ROM o unidad de disco de 3.5 ".

4.2.1 PROGRAMA EN VISUAL BASIC

El software desarrollado tiene una pantalla principal en la que se tienen las siguientes ventanas.

- Ventana de diseño.
- Ventana de información.
- Ventana de ensamblado (oculta).

Figura 4.20 Pantalla Principal

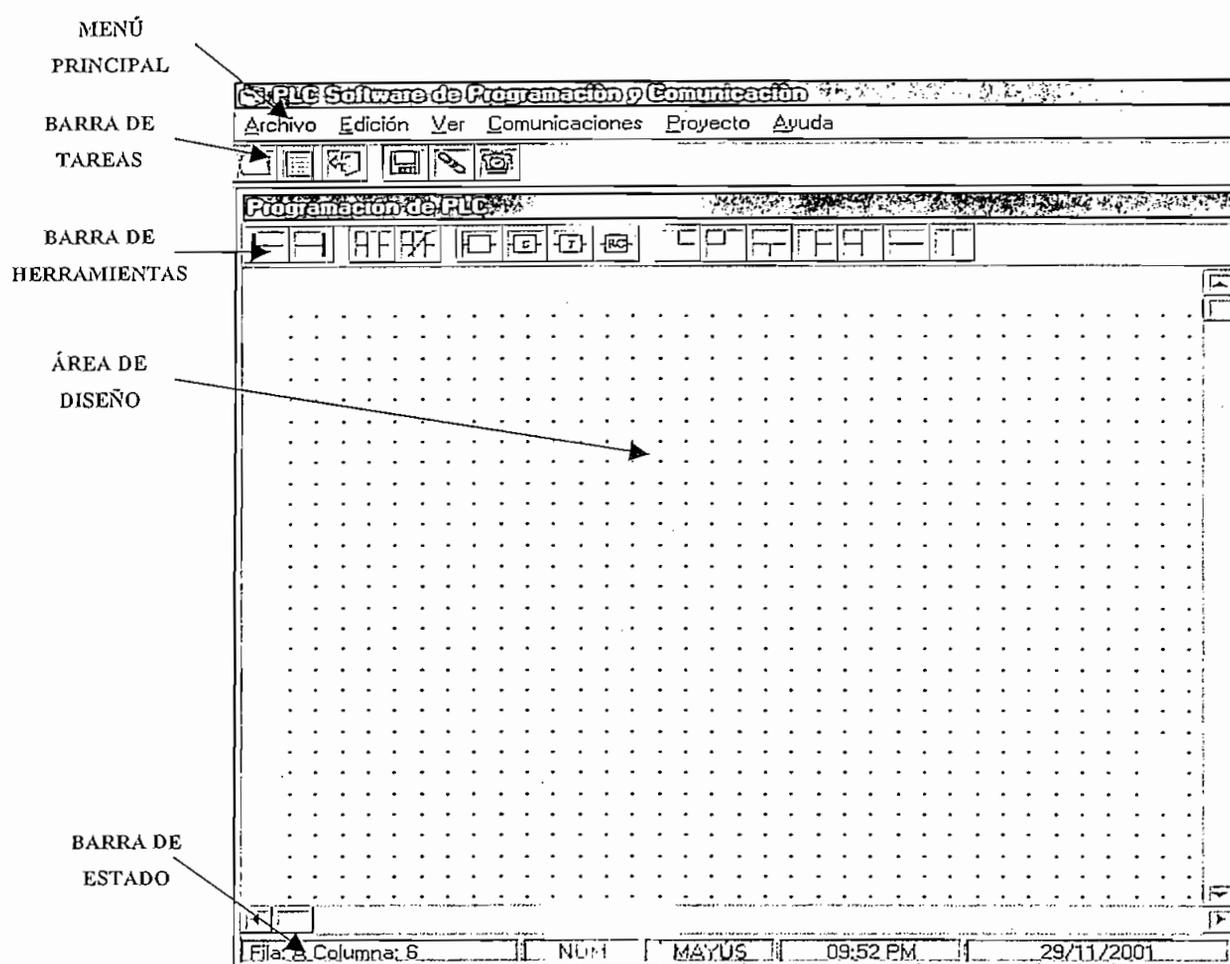


4.2.1.1 VENTANA DE DISEÑO

La finalidad de la ventana de diseño es permitir al usuario desarrollar el diagrama de escalera de forma gráfica. Consta de las siguientes partes:

- Menú Principal
- Barra de Tareas.
- Barra de Herramientas.
- Barra de Estado.
- Area de Diseño.

Figura 4.21 Ventana de Diseño



4.2.1.1.1 MENÚ PRINCIPAL

El siguiente diagrama de árbol corresponde al menú principal.

Diagrama de Menú Principal



ARCHIVO

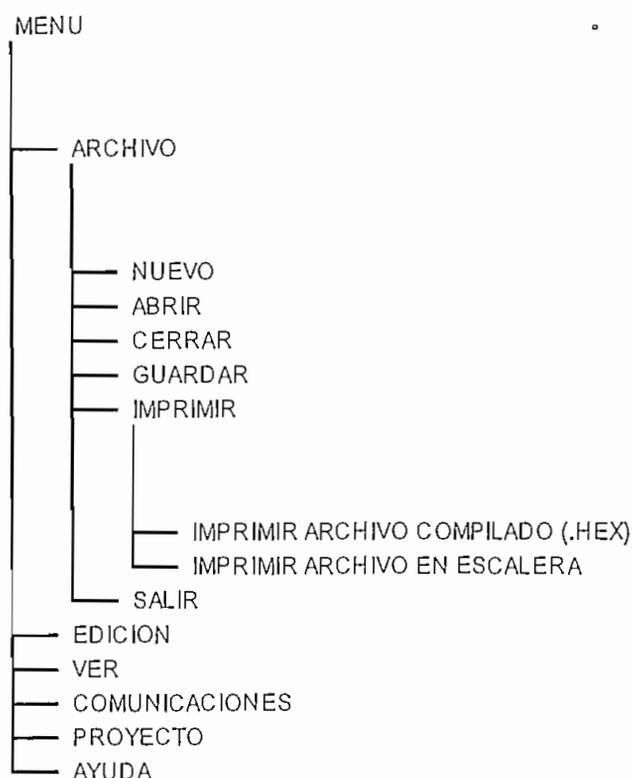
La finalidad del menú archivo es permitir al usuario manipular los archivos que se relacionan con el diagrama de escalera establecido por él, lo que implica abrir un archivo ya existente, guardar el diagrama actual desarrollado por el usuario, abrir un nuevo archivo para desarrollar un diagrama de escalera, imprimir el diagrama gráfico de escalera, así como un archivo de texto que contiene la información del ensamblado.

NUEVO: Permite realizar un nuevo diagrama de escalera para lo cual el programa borra la información que existiese en cada una de sus ventanas así como inicializa todas las variables útiles en la diagramación.

ABRIR: Cuando se requiere utilizar un diagrama de escalera previamente elaborado y archivado, el submenú Abrir, simplemente abre el archivo seleccionado a través de un cuadro de diálogo que se presentará en la ventana de diagramación. Este archivo es de texto con la extensión ".txt", el cual contiene la información en texto de la matriz de datos de un diagrama de escalera previamente guardado por medio del submenú Guardar.

CERRAR: Cierra el diagrama de escalera activo, reseteando todas las variables y base de datos así como borrando los gráficos.

Diagrama de Barra de Menús



GUARDAR: Cuando se haya diseñado el diagrama de escalera, el usuario tiene la opción de guardarlo en un archivo de texto. Para guardarlo se presenta un cuadro de diálogo en el cual el usuario podrá introducir el nombre del archivo, así como el directorio donde va a guardarse. El archivo de texto creado tiene la estructura siguiente: las dos primeras líneas de información, constituyen datos de los índices utilizados por el programa, la información restante constituye cada dato correspondiente a los elementos del diagrama de escalera.

IMPRIMIR: Permite al usuario realizar la impresión del diagrama de escalera que él ha desarrollado y que se encuentra expuesto en la ventana de diseño. Se presentan dos opciones: imprimir el gráfico e imprimir el archivo de ensamblado. El imprimir gráfico implica la impresión del diagrama de escalera así como de la información de los elementos empleados mientras que el imprimir archivo

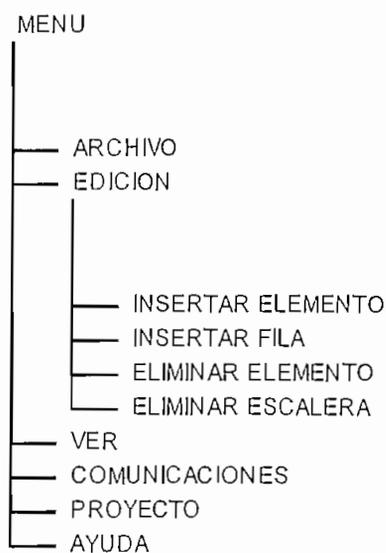
ensamblado implica la impresión del archivo con extensión ".hex" del diagrama de escalera activo, es decir el que está presente en la ventana de ensamblado.

SALIR: Permite salir del programa, cerrando todas las ventanas y archivos. Durante este proceso se pregunta al usuario mediante ventanas de diálogo si desease guardar el diseño realizado así como su respectivo archivo ensamblado.

EDICIÓN

La finalidad de este menú es permitir al usuario editar el diagrama de escalera activo insertando un nuevo elemento, una nueva fila o eliminando un elemento o eliminando peldaños.

Diagrama de Menú Edición



Para acceder a este submenú se lo puede hacer: directamente en la barra de tareas o ubicándose en el gráfico en las coordenadas donde se va a editar y al aplastar el botón derecho del ratón aparecerá este submenú activo en la posición del cursor.

INSERTAR ELEMENTO: Inserta un elemento en un peldaño y solo abre un espacio en el peldaño afectado. El usuario con esta acción requiere completar el peldaño afectado para tener éxito al ensamblar el programa.

INSERTAR FILA: Inserta una fila, desplazando todos los elementos de las subsiguientes filas hacia abajo a las coordenadas de donde se llamó a esta acción. En caso de que se haya insertado una fila dentro de algún peldaño, al igual que "Insertar elemento", el usuario requiere completar el peldaño afectado.

ELIMINAR ELEMENTO: Elimina un elemento de un peldaño. No se altera la posición de ninguno de los elementos, simplemente deja un espacio en blanco en el diagrama de escalera donde el elemento fue eliminado.

El elemento eliminado requiere ser completado por el usuario para finalizar el peldaño del diagrama.

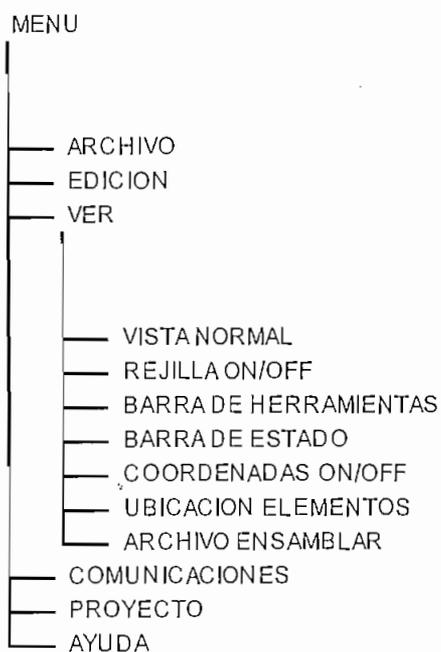
ELIMINAR PELDAÑO: Elimina un peldaño del diagrama de escalera, desplazando todos los peldaños posteriores a este hacia arriba, cerrando la brecha aparecida por esta acción.

VER

El propósito de este submenú es poder manipular algunas ayudas gráficas de la ventana de diseño tales como:

- Generación de grilla.
- Presentación de la Barra de Herramientas,
- Presentación de la Barra de Estado, etc.

Diagrama de Menú Ver



COMUNICACIONES

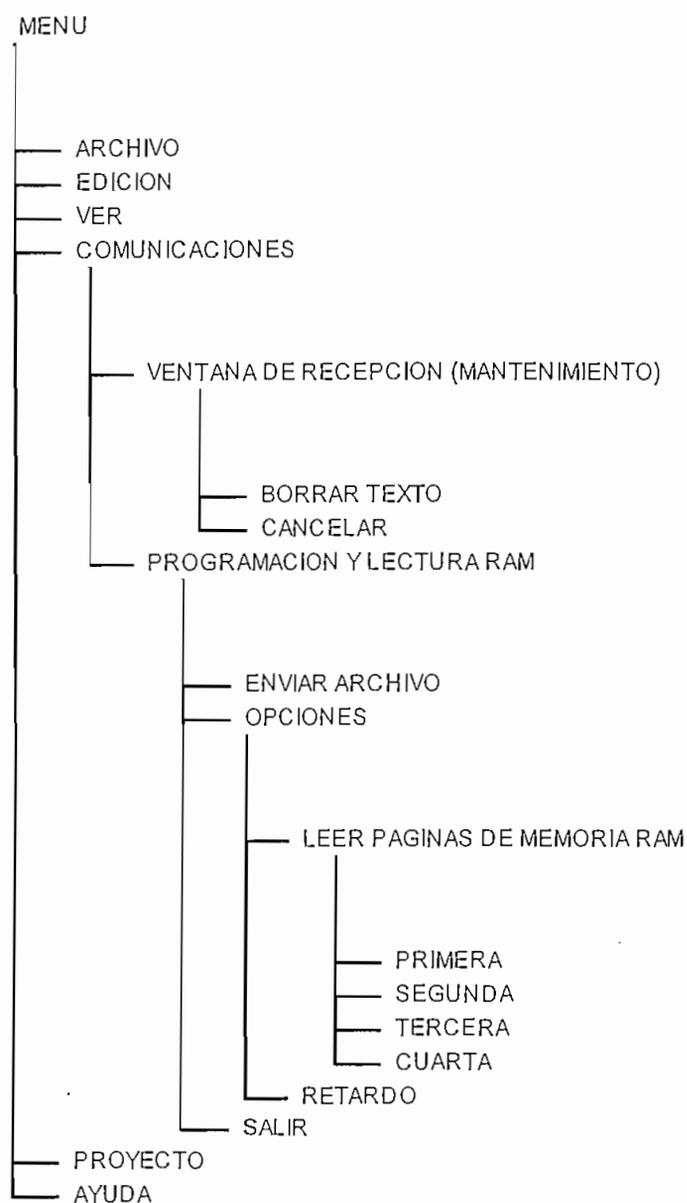
Tiene como finalidad el establecer los parámetros de comunicación para el pórtico serial. La ventana de comunicaciones permite al usuario ver el estado de la transmisión de información desde el PLC al PC o viceversa y no es editable. Generalmente se lo utiliza al encender el PLC o cuando se ha hecho un reset. En transmisión y recepción de datos ayuda a que el usuario pueda saber y enviar el archivo ensamblado y saber si fue recibido. Además se puede acceder a la memoria del PLC a través de las teclas correspondientes.

Se ha empleado el protocolo RS-232C para la comunicación serial, para lo cual es requerido usar un cable tipo null modem para interconectar al PLC con el PC. Los parámetros de comunicación utilizados son los siguientes:

Velocidad de transmisión:	9600 baudios
Paridad:	Ninguno

Número de bits: 8 bits
 Bit de parada: 1 bit

Diagrama de Tratamiento de Datos



VENTANA DE RECEPCIÓN (MANTENIMIENTO): Tiene como finalidad presentar los datos enviados desde el PLC para comprobar que existe comunicación cuando se entra en modo de servicio (mantenimiento) mediante el respectivo

interruptor. Para ingresar en esta ventana se requiere un código de acceso de dos dígitos.

PROGRAMACIÓN Y LECTURA RAM: Permite acceder a la memoria RAM del PLC. El programa envía previamente una advertencia que indica al usuario que se cerciore de que ha conectado el PC al PLC, automáticamente rastrea cual de los dos puertos seriales está activo; en caso de que no esté conectado envía otro mensaje de error de comunicaciones, saliendo de esta ventana. Establecida la comunicación se despliega una ventana para programación y lectura de la RAM, conteniendo tres botones en la parte inferior: Enviar Archivo, Opciones y Salir.

ENVIAR ARCHIVO: Permite que se envíe al PLC un archivo ensamblado que el usuario previamente lo haya abierto mediante un cuadro de diálogo.

OPCIONES: Despliega un submenú flotante en la posición del cursor con los siguientes submenús: Leer Página de Memoria RAM y Retardo.

LEER PÁGINA DE MEMORIA RAM: Permite al usuario leer la memoria RAM con el fin de poder comparar los datos que recibió el PLC luego de que un archivo ensamblado con el que se encuentra presente en la ventana de ensamblado. Se ha organizado la memoria RAM en cuatro páginas, cada una de ellas de 256 bytes de memoria.

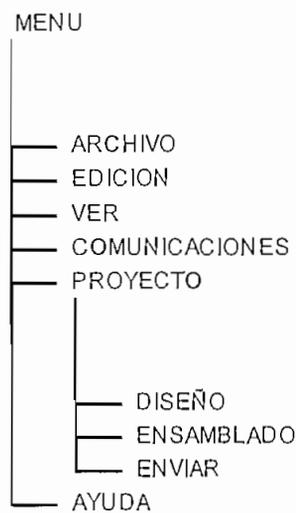
RETARDO: Su objetivo es establecer un tiempo de retardo entre dos datos consecutivos durante el envío de datos del archivo ensamblado. Se ha establecido como tiempo mínimo 50 milisegundos, puesto que se ha probado que para valores menores a este se presentan problemas en la transmisión de datos al PLC. El usuario puede determinar el valor de este tiempo de retardo mediante una ventana que se abre al ingresar a esta opción. Mientras mayor sea el tiempo de retardo mayor será el tiempo que tomará la transmisión de datos.

SALIR: Permite salir de esta ventana de comunicaciones.

PROYECTO

Constituye la parte medular del diseño del diagrama de escalera. Este menú tiene la siguiente estructura: .

Diagrama de Menú Proyecto



DISEÑO: Permite al usuario realizar su diagrama de escalera, colocando todas las variables de diseño en sus valores iniciales y borrando las ventanas de diseño anteriores para empezar con una diagramación en escalera.

Durante el diseño, mientras el usuario va ubicando los diferentes elementos del diagrama de escalera, el programa chequea si existen restricciones para ubicarlos; en caso de existirlos, el programa envía el respectivo mensaje de restricción mediante un cuadro de información. El proceso de diseño en este programa se lo explicará con mayor detalle posteriormente.

ENSAMBLAR: Permite, luego de que se ha establecido el diagrama de escalera, el ensamblado correspondiente en lenguaje hexadecimal que será enviado al PLC y cuya estructura ya ha sido explicada anteriormente.

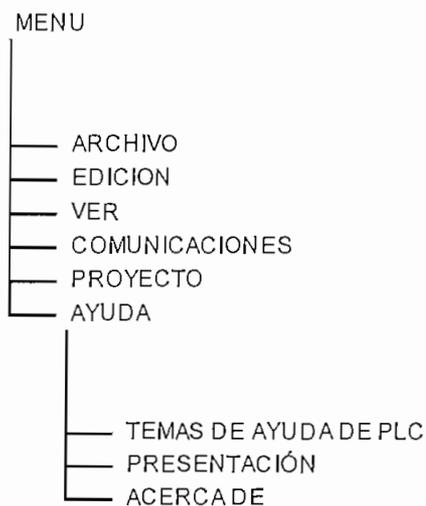
Al seleccionar este submenú el programa revisa si existen problemas durante esta acción, es decir, si existen restricciones en el diseño gráfico del diagrama de escalera; en caso de encontrarlos no se ensambla.

ENVIAR: Tiene como finalidad el envío del archivo ensamblado hacia el PLC, abriendo una nueva ventana que permite ver al usuario el estado de la transmisión del archivo ensamblado así como tener acceso a la memoria RAM del PLC, para poder chequear los datos enviados.

AYUDA

Como en todas las aplicaciones de Windows se dispone de un submenú con la finalidad de proporcionar ayudas al usuario, ya sea acerca del desarrollo del programa o de los elementos de programación por medio de explicaciones breves, sencillas y rápidas. También se presentan los datos más importantes que forman el programa de programación en Lenguaje Escalera del PLC.

Diagrama de Menú Ayuda



BARRA DE TAREAS

Contiene botones (softkeys) enlazados con los procesos más relevantes y utilizados del programa como son: Abrir archivo, Nuevo archivo, Salir, Guardar archivo, Editar, Comunicaciones. Todos forman parte del Menú Principal. Estos iconos permiten el acceso directo a la tarea solicitada.

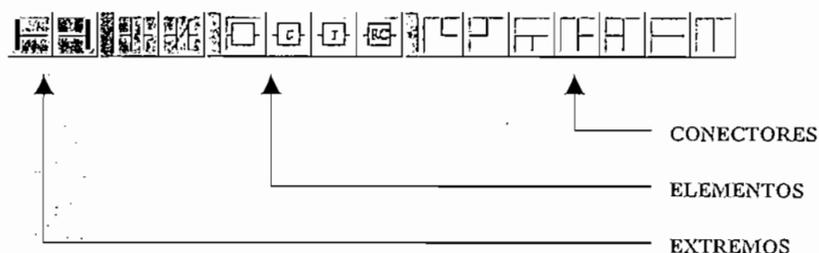
Figura 4.22 Barra de Tareas



BARRA DE HERRAMIENTAS

Contiene los botones correspondiente a los diferentes elementos que se pueden emplear para diseño. Es posible activar un botón a la vez, sin embargo éste permanecerá activo hasta que otro botón de esta barra sea elegido. Además el usuario puede saber cual herramienta está activa debido a que el cursor del ratón cambiará el icono de la respectiva herramienta.

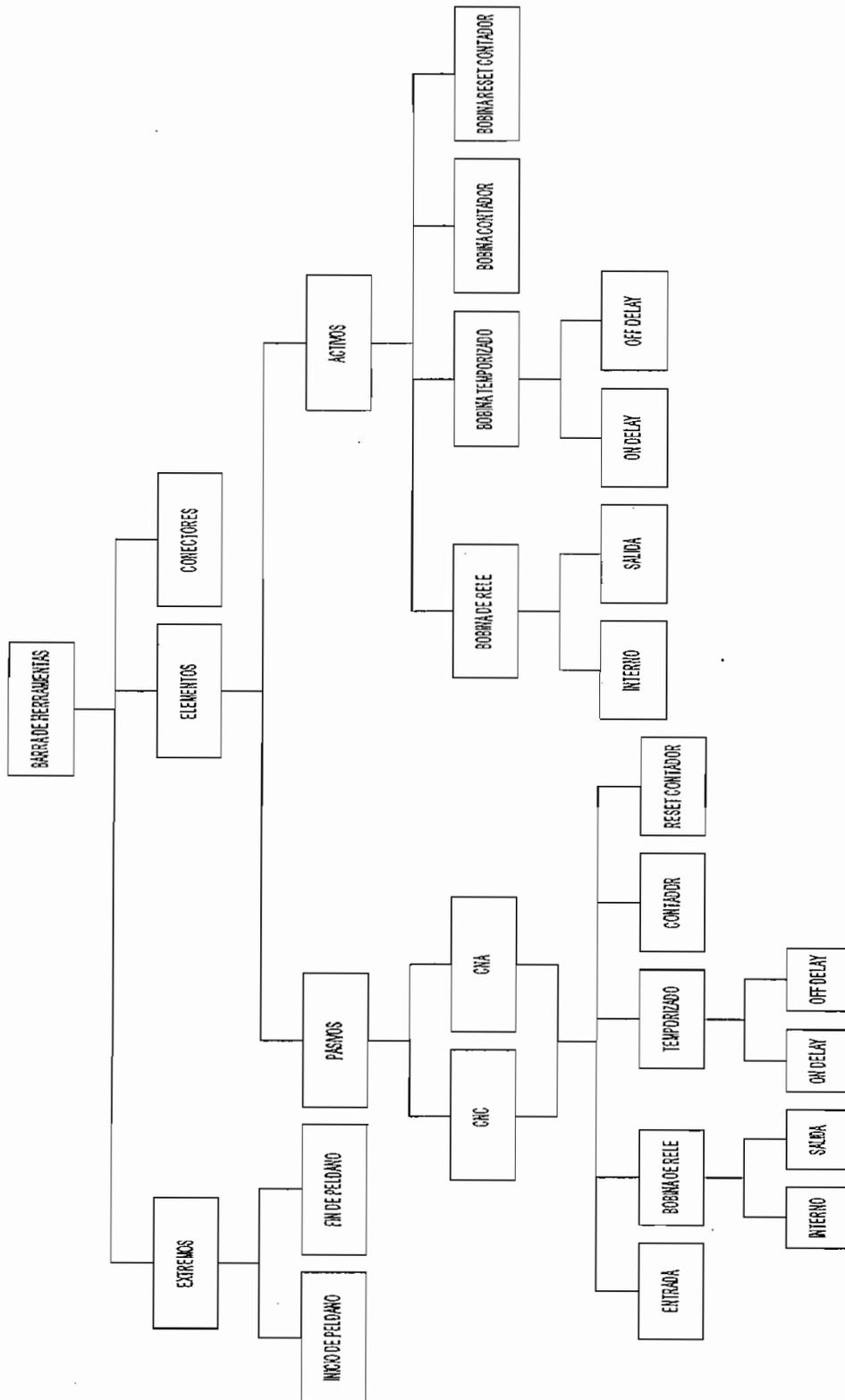
Figura 4.23 Barra de Herramientas



La barra de herramientas está dividida en tres áreas de funciones: Extremos, Conectores y Elementos (**Figura 4.23**).

Los extremos comienzan y terminan los diferentes peldaños de un diagrama escalera, los conectores son elementos de enlace que permite tener un gráfico visiblemente aplicable y los elementos sirven para formar la lógica de control.

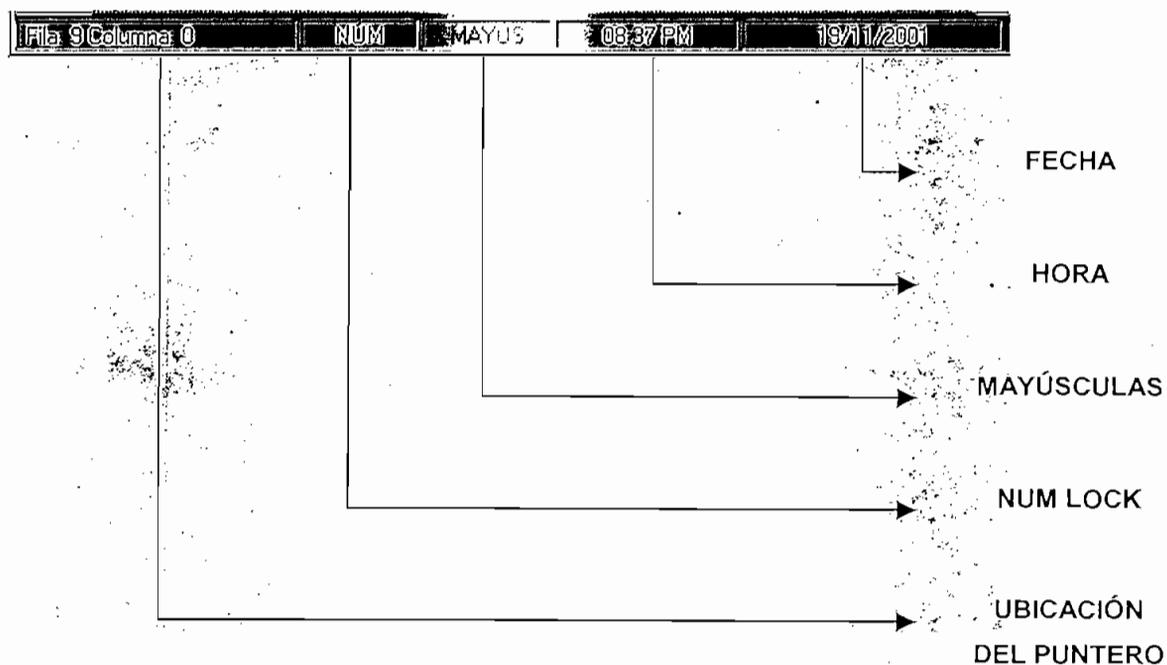
Figura 4.24 Organigrama de la Barra de Herramientas



BARRA DE ESTADO

Ubicada en la parte inferior de la ventana de diseño, presenta información general del sistema como es la hora, si está activa la tecla Caps Lock así como también Num Lock, y las coordenadas del cursor en el área de diseño presentadas en fila y columna. Esto último permitirá al usuario conocer la posición actual del cursor dentro del área de diseño.

Figura 4.25 Barra de Estado



4.2.1 ÁREA DE DISEÑO

Constituye el área sobre el cual el usuario puede desarrollar gráficamente el diagrama de escalera a enviarse al PLC. En el lado derecho e inferior se ubican los Scroll Bars vertical y horizontal respectivamente, los cuales permitirán desplazarse sobre el diagrama fuera de los límites visuales de la ventana de diseño, así también dispone como fondo una rejilla de puntos.

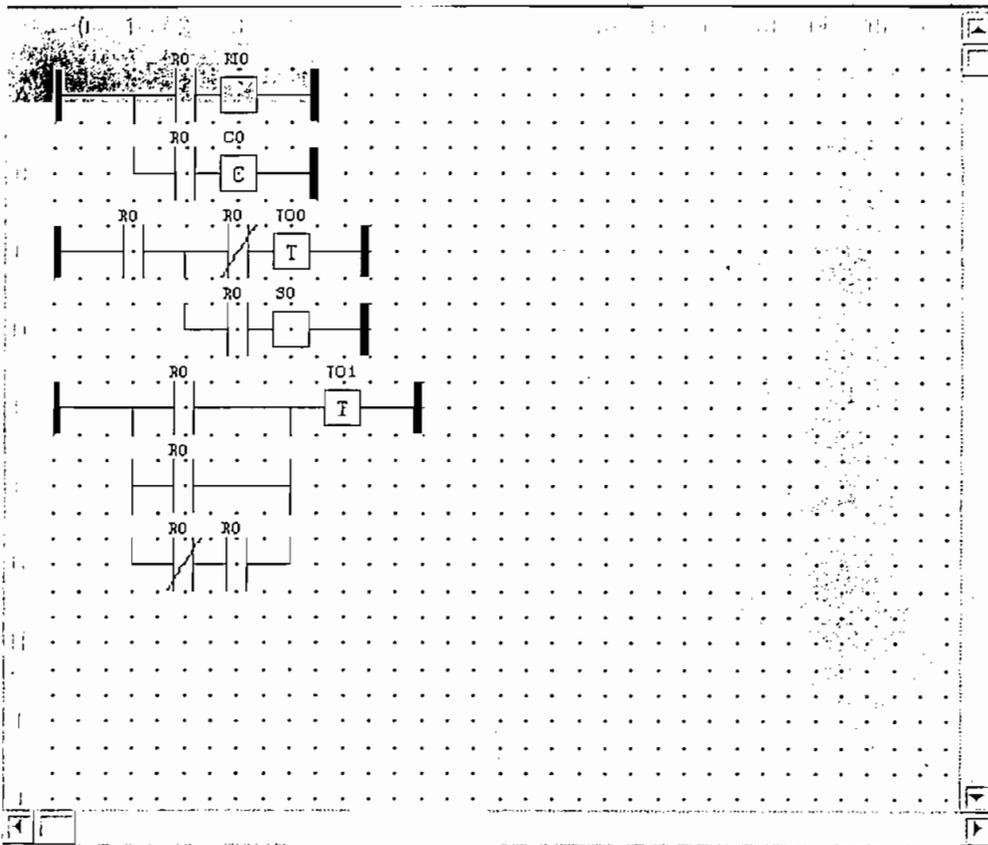
Se ha establecido 30 filas por 28 columnas, lo que implica que se tiene una matriz gráfica de 30x28, es decir es posible graficar un diagrama de escalera de 30 peldaños sin derivaciones con 28 elementos consecutivos, hasta un máximo de 840 elementos. Si bien es posible hacerlo, esto repercute en el tiempo de procesamiento gráfico del diagrama de escalera, puesto que como el programa trabaja con controles Image de Visual Basic, mientras más de dichos controles se carga en memoria más tiempo toma en presentarlos en pantalla. Esto puede evidenciarse al mover el Scroll Bar del área de diseño ya que se produce un efecto de retardo en redistribuirse los elementos del diagrama de escalera.

Cada elemento ubicado en la ventana de diseño tiene un tamaño definido 400x400 twips (unidad de medida gráfica de Windows) y sobre éstos van etiquetas que tienen un tamaño de 75x150 twips, con información referente al número de elemento asignado. La información complementaria para el caso de temporizados y contadores se los puede visualizar simplemente ubicando el cursor sobre el elemento, en cuyo caso se desplegará una etiqueta del tipo Tool Tip Text de Windows.

Es posible ubicar los elementos en esta área bajo ciertas restricciones que han sido establecidas con el fin de evitar errores durante el diseño. Una de las restricciones que tiene el programa es que es posible insertar un inicio de peldaño únicamente en la columna 0. Aun cuando se puede dibujar un cortocircuito, sin embargo durante el ensamblado el programa lo detectará enviando un mensaje de error de ensamblado.

Todas estas restricciones son soportadas con mensajes de texto que indican al usuario la causa de las mismas y en algunos casos dispone de ayudas o sugerencias que puede seguir para continuar con el diseño del diagrama de escalera.

Figura 4.26 Ejemplo de Diagrama de Escalera Diseñada en el Área.



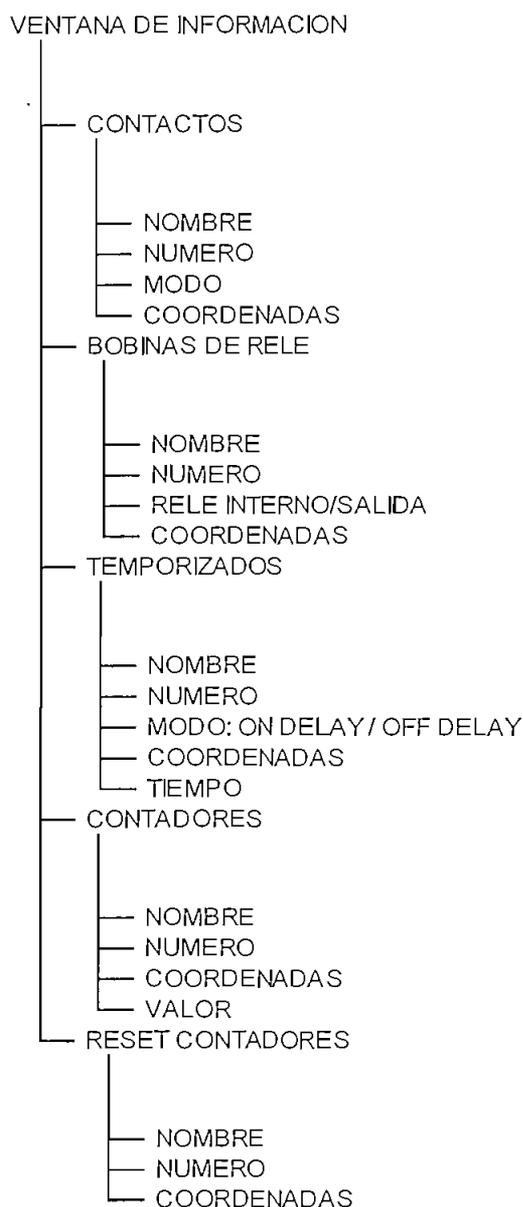
4.2.1.2 VENTANA DE INFORMACIÓN

Junto a la ventana de diseño se encuentra la ventana de información no editable, creada con el fin de dar al usuario información de los elementos constitutivos del diagrama de escalera diseñado. Esta información está distribuida en los cinco grupos de elementos activos y pasivos, los cuales son: contactos, bobinas de relé, temporizados, contadores y reset contadores.

Figura 4.27 Ventana de Información

Utilización de Elementos				
Por Elemento		General	Elementos Relacionados	
Temporizado	ID	Coordenadas	Modo	File
<input type="text"/>				
<input type="button" value="←"/> <input type="button" value="→"/>				
Entrada/Salida	ID	Coordenadas	Modo	ENT/S
<input type="text"/>				
<input type="button" value="←"/> <input type="button" value="→"/>				
Relé	ID	Coordenadas		
<input type="text"/>				
<input type="button" value="←"/> <input type="button" value="→"/>				
Contador	ID	Coordenadas	Modo	Base
<input type="text"/>				
<input type="button" value="←"/> <input type="button" value="→"/>				
Contacto	ID	Coordenadas	NC/NA	Elem
<input type="text"/>				
<input type="button" value="←"/> <input type="button" value="→"/>				

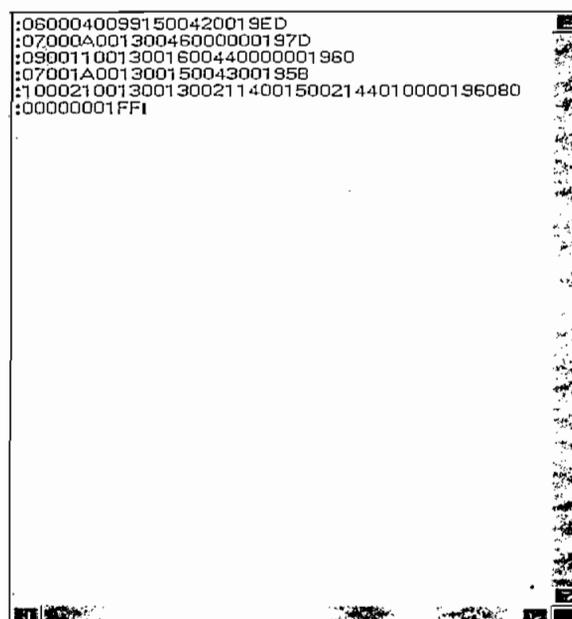
Diagrama de Ventana de Información



4.2.1.3 VENTANA DE ENSAMBLADO

Esta ventana ha sido desarrollada con el propósito de visualizar por medio de una ventana de texto, los datos del archivo ensamblado que son enviados al PLC. Esta ventana es muy útil cuando se desea comparar el contenido de las diferentes páginas de la memoria RAM con el respectivo archivo ensamblado.

Figura 4.28 Ventana de Ensamblado



4.2.1.4 VENTANAS DE INTRODUCCIÓN DE DATOS

Las ventanas de introducción de datos se presentan visibles al usuario cuando se escoge alguno de los elementos de control de la barra de herramientas. En estas ventanas es posible introducir todos los parámetros correspondientes al elemento escogido tales como: número asignado, tipo de elemento, modo de operación.

Los datos pasarán a formar parte de una matriz de datos cuando se presiona el botón Aceptar de la ventana, así también se presenta visible el elemento en el diagrama. Los datos no ingresan en la matriz de datos si el botón Cancelar es presionado.

En la **Figura 4.29** se detallan las ventanas de introducción de datos utilizadas en el programa.

Figura 4.29 Ventanas de introducción de datos

Bobinas de Relés [X]

RELÉ
Número

Tipo Relé Interno
 Relé de Salida

Aceptar Cancelar

Contactos Abiertos [X]

CONTACTO
Número

Tipo Relé
 Contador
 Temporizado
 Entrada/Salida

Aceptar Cancelar

Reset de Contadores [X]

Contador a
Resetear

Aceptar Cancelar

Bobina de Contadores [X]

CONTADOR
Número

Cuenta Base

Aceptar Cancelar

Bobinas de Temporizados [X]

TEMPORIZADOR
Número

Tiempo Retardo (mseg)

Tipo On Delay
 Off Delay

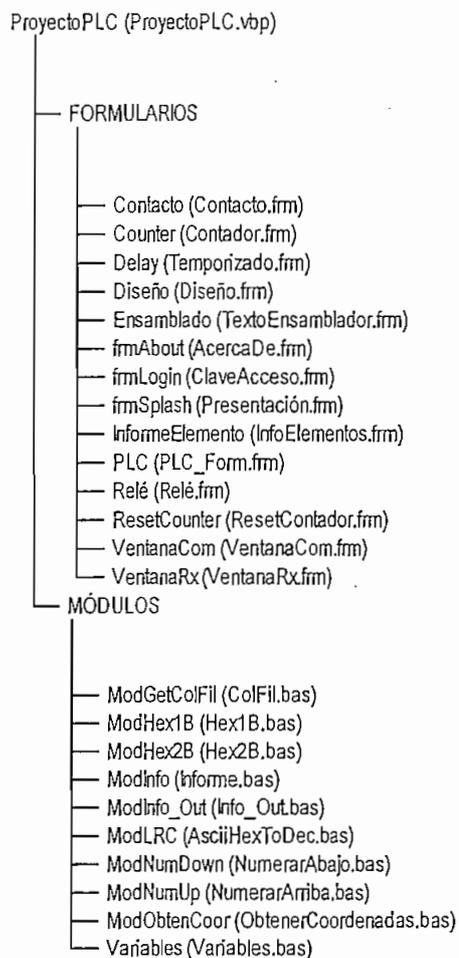
Aceptar Cancelar

4.2.2 SUBROUTINAS DEL PROGRAMA

El programa que ha sido desarrollado usando Visual Basic 5.0 contiene los siguientes formularios y módulos, de los cuales se dará una breve explicación en cuanto a su finalidad y aplicación, así como también de algunas subrutinas relacionadas con ellos. Se aclara que entre paréntesis y junto a los nombres de los formularios y módulos se declara el nombre del archivo correspondiente, que para el caso de formularios tienen la extensión “.frm”, para módulos la extensión “.bas” y para el proyecto general, cuyo nombre es “ProyectoPLC” tiene una extensión “.vbp”.

FORMULARIOS

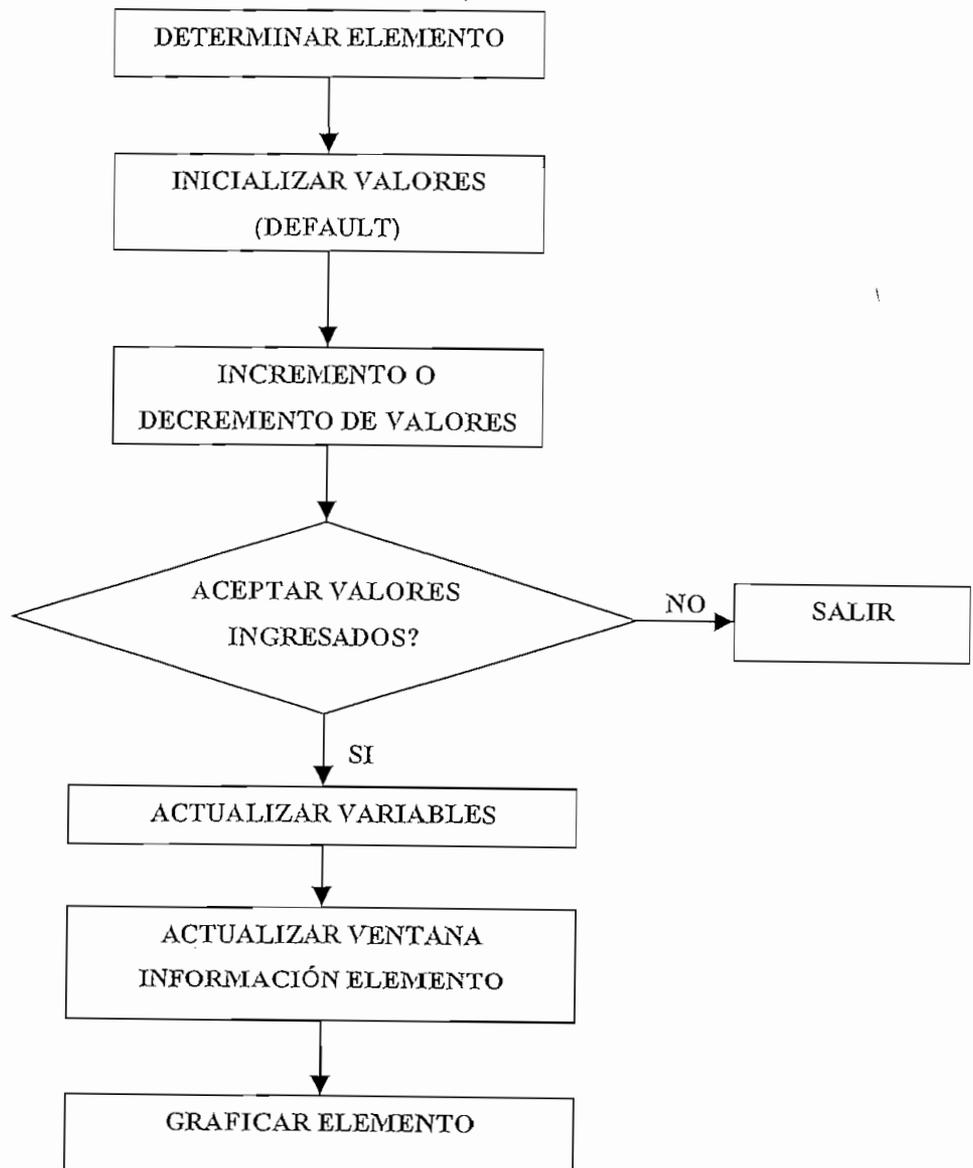
Organización del Programa en Visual Basic



CONTACTO: Este formulario presenta la ventana de introducción de datos correspondiente a los contactos de temporizados, relés, contadores, entradas o salidas.

COUNTER: Presenta la ventana de introducción de datos para contadores. La subrutina Elementos Activos está relacionada con esta ventana. (Figura 4.30).

Figura 4.30 Subrutina de Elementos Activos



DELAY: Presenta la ventana de introducción de datos de los temporizados. El dato correspondiente al tiempo de retardo está expresado en milisegundos. Al igual que el caso anterior la subrutina Elementos Activos está relacionada con esta ventana (**Figura 4.30**).

DISEÑO: Constituye el formulario principal del programa, relacionado con la ventana de diseño y su finalidad es permitir al usuario desarrollar el diagrama de escalera.

La subrutina Diseñar permite manipular los gráficos que se emplean en el diagrama de escalera que el usuario está desarrollando. Cuando el usuario ingresa mediante el menú principal *Proyecto / Diseñar* está invocando esta subrutina, siendo la primera acción el poner todos los valores de las variables en condiciones iniciales, inicializando las diferentes matrices de datos (dato y datoTemp) y activando la rejilla gráfica, barra de estado, borrando los datos de la ventana de información, entre otras acciones complementarias. Inmediatamente asigna las coordenadas actuales del cursor y los ubica junto a éste con formato (fila, columna). Este valor se actualiza cada vez que el cursor se desplaza por la ventana de diseño. Cuando se elige un elemento gráfico de la barra de herramientas y lo ubica en el área de diseño, el programa actualiza la matriz dato(col, fila) y antes de cargar el gráfico correspondiente chequea si hay restricciones con el elemento a implementarse. De no haberlos abre la respectiva ventana de introducción de datos, actualiza la matriz dato(col, fila) con dichos datos y agrega finalmente la imagen del gráfico seleccionado con su respectiva etiqueta en la parte superior de la imagen. En caso de que el usuario cancela la introducción de datos, se vacía los datos de la matriz dato(col, fila) de la posición seleccionada. (**Figura 4.31**).

El proceso de ensamblado constituye el siguiente paso a seguir durante la elaboración del diagrama de escalera y se basa en la matriz dato(col, fila) creada durante el diseño. Este es un proceso de lectura secuencial de dicha matriz, transformando sus datos en comandos que serán cargados e interpretados por el

PLC, estructurando dichos comandos en instrucciones que siguen el código de envío de datos Intel Standard.

Figura 4.31 Subrutina Diseñar

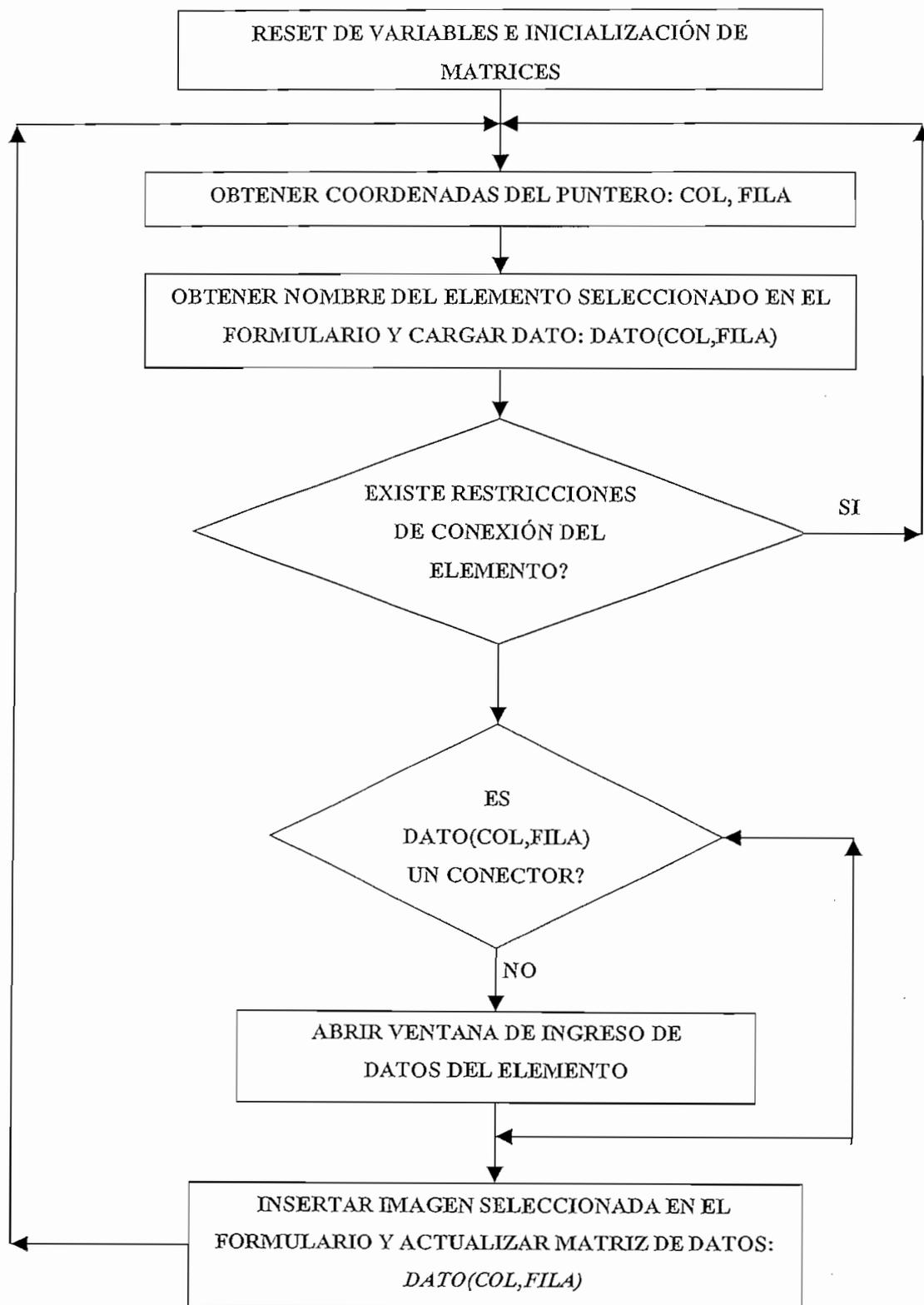
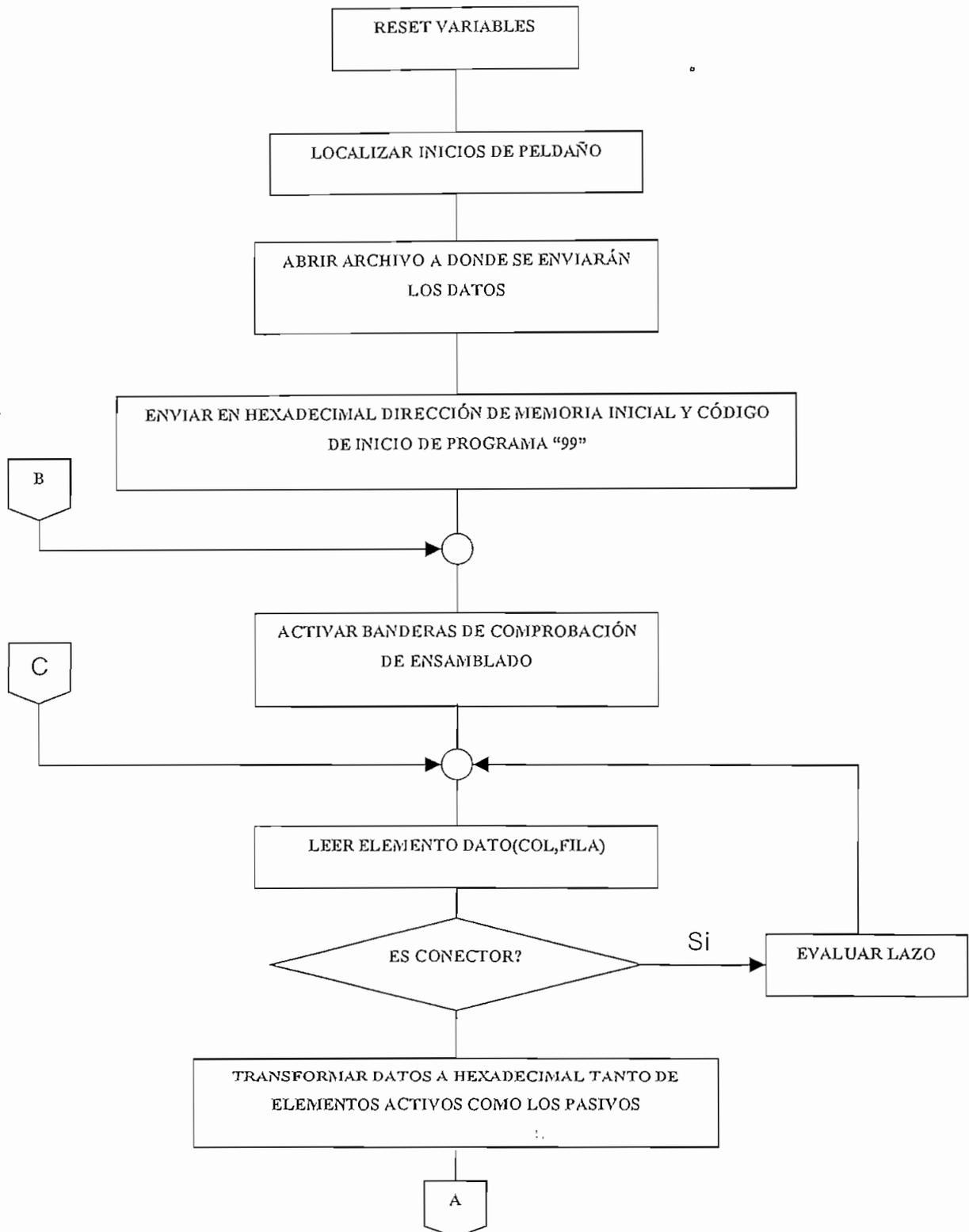
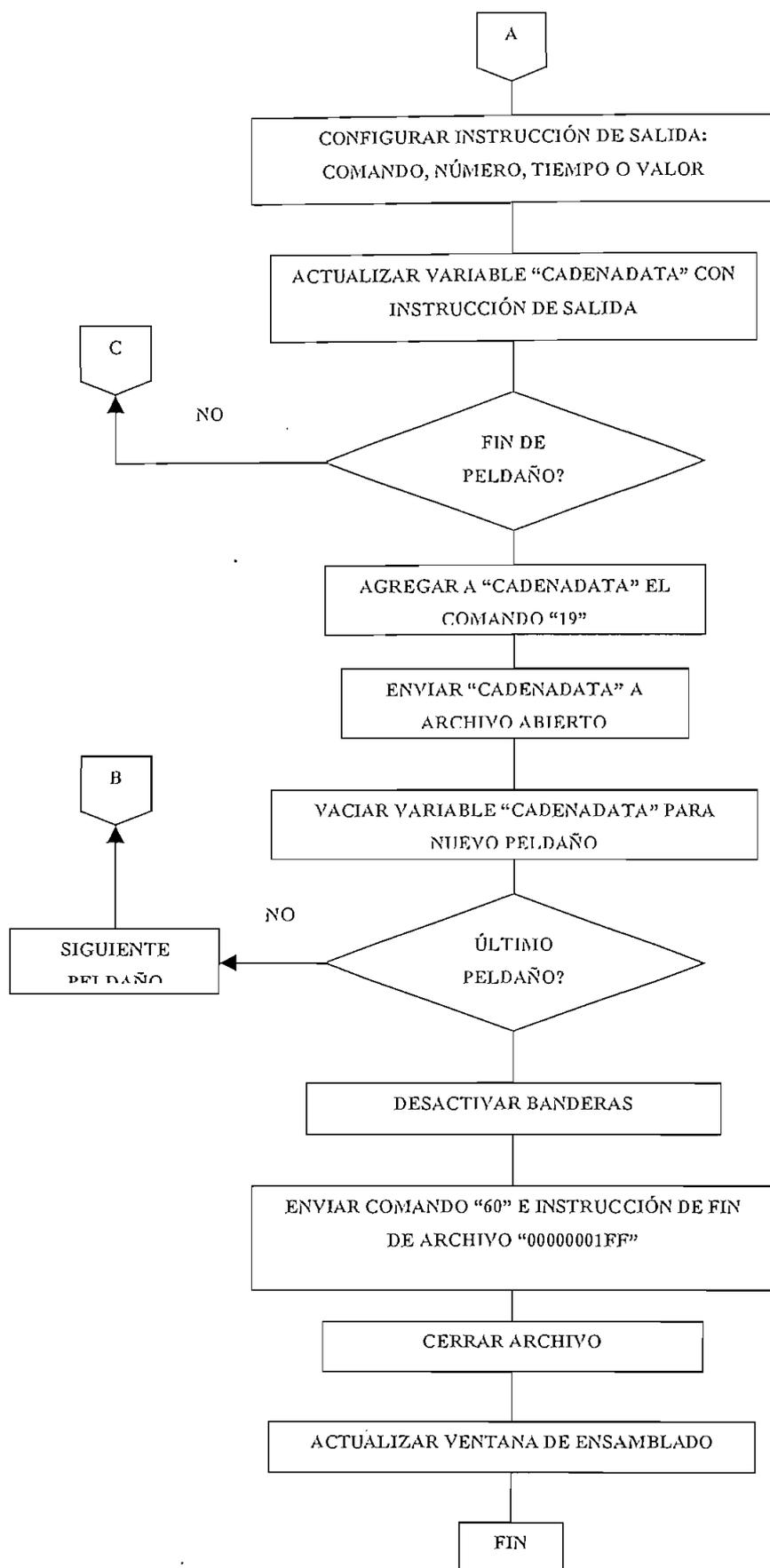


Figura 4.32 Subrutina de Ensamblado.





Para el envío del archivo ensamblado del diagrama de escalera desarrollado, la ventana de comunicaciones se activa (**Figura 4.33**).

Por otra parte el abrir o guardar el archivo gráfico implica la creación de un archivo de texto, el cual contiene la información necesaria del diagrama de escalera diseñado. Para explicar la estructura de este tipo de archivo se presenta el siguiente ejemplo:

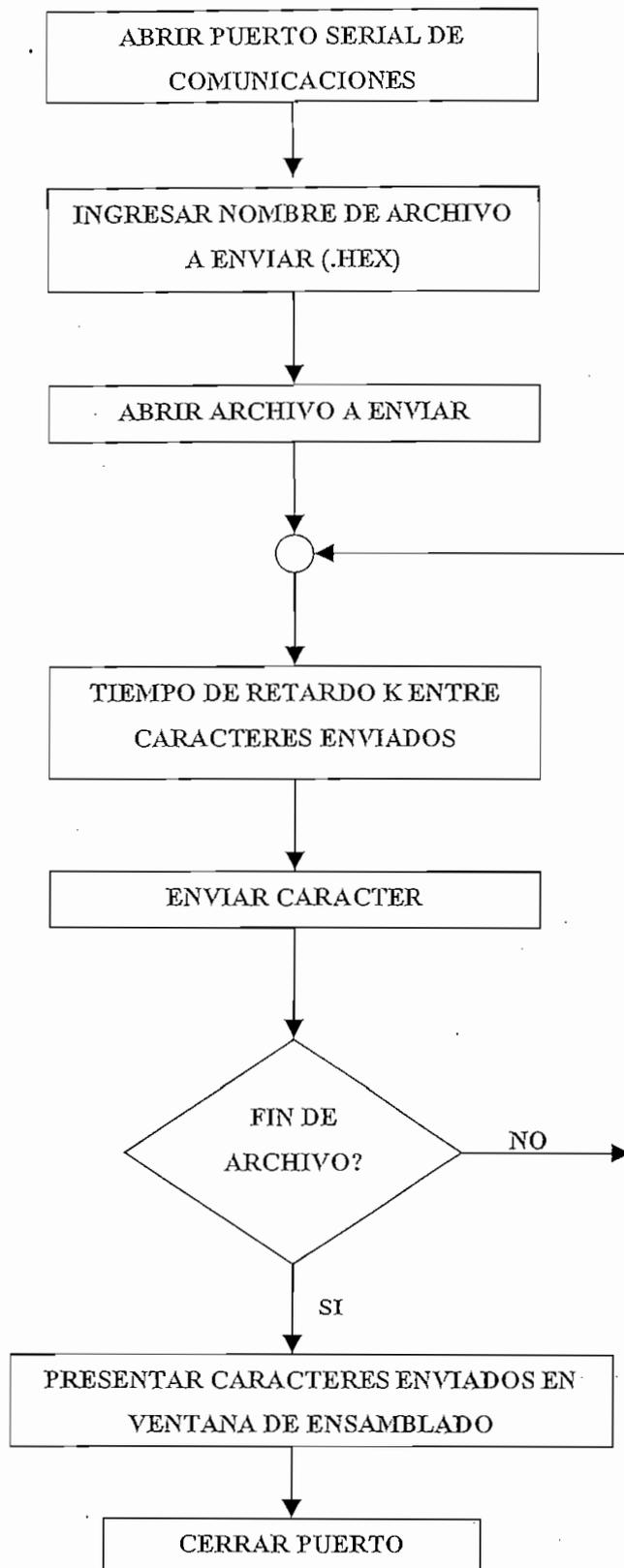
```

3,3,87,4
115
",",",", "inicio", 0, "", "", 1, "", 0, 0
"03", "E0", "cno", 0, "", "", 3, "01", 0, 1
"43", "S1", "bobina", 1, "", "", 6, "", 0, 2
",",",", "final", 0, "", "", 2, "", 0, 3
",",",", "inicio", 0, "", , 1, "", 1, 0
"15", "R0", "cno", 0, "", "", 3, "13", 1, 1
"16", "R0", "cnc", 0, "", "", 4, "14", 1, 2
"43", "S0", "bobina", 0, "", "", 6, "", 1, 3
",",",", "final", 0, "", , 2, "", 1, 4
",",",", "inicio", 0, "", , 1, "", 2, 0
"16", "R0", "cnc", 0, "", "", 4, "14", 2, 1
"47", "RC0", "reset", 0, "", "", 24, "", 2, 2
",",",", "final", 0, "", , 2, "", 2, 3

```

La primera línea contiene información de coordenadas de la columna, fila del elemento más extremo del último peldaño, índice del elemento más extremo y máximo peldaño creado. La siguiente línea da información del máximo índice en el máximo peldaño. Las líneas subsiguientes presentan el comando hexadecimal, etiqueta, nombre del elemento, número de referencia del elemento, bandera de presencia, tiempo o valor a contar (en caso de temporizados o contadores), índice de la imagen a la que corresponde, comando complementario, posición fila, posición columna.

Figura 4.33 Subrutina Enviar Archivo Ensamblado a PLC.



ENSAMBLADO: Muestra la ventana de ensamblado, la cual permanece oculta durante el proceso de diseño y se activa cuando se está ensamblando el diagrama de escalera, ocultando en este caso la ventana de información. Útil para cuando se ha cargado un archivo ensamblado en el PLC y se requiera comprobar que la memoria RAM contenga los datos correctos, pues si el usuario está familiarizado con el código de envío de datos Intel Standard, fácilmente puede chequear dichos datos en el PLC.

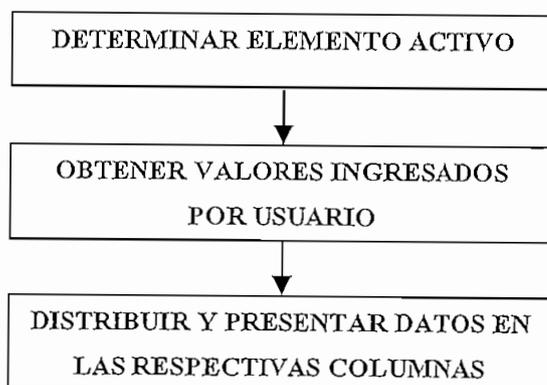
frmABOUT: Ventana que contiene información general sobre el programa. Se la activa mediante el menú *Ayuda*.

frmLOGIN: Ventana que presenta la opción de ingresar una clave de acceso para poder activar la ventana de recepción (mantenimiento).

frmSPLASH: Ventana de presentación que se activa inmediatamente al ingresar al programa, así como cuando se sale del mismo.

INFORME ELEMENTO: Constituye la ventana de información, dividida en cinco secciones correspondientes a cada elemento como son temporizados, entradas/salidas, bobinas de relé, contadores y contactos. También tiene la opción de presentar en una sola lista todos los elementos que fueron empleados en el diseño.

Figura 4.34 Subrutina Ventana Información Elementos.



PLC: Es el formulario principal o formulario con interfaz de múltiples documentos (MDI) en el depósito. La interfaz de documentos múltiples (MDI) permite crear una aplicación que mantenga varios formularios dentro de un único formulario contenedor.

Una aplicación MDI permite que el usuario vea varios documentos al mismo tiempo y cada documento se presenta en su propia ventana. Los documentos o *ventanas secundarias* están contenidos en una *ventana primaria*, que proporciona un espacio de trabajo para todas las ventanas secundarias de la aplicación.

RELÉ: Presenta la ventana de introducción de datos de bobinas de relé internas o de salida. La subrutina Elementos Activos está relacionada con esta ventana. (Figura 4.29).

RESETCOUNTER: Constituye la ventana de introducción de datos de reset de contadores. La subrutina Elementos Activos está relacionada con esta ventana. (Figura 4.29).

VENTANACOM: Relacionada con la ventana de comunicaciones. Es posible establecer la comunicación serial con el PLC mediante la subrutina Abrir Puerto (Figura 4.35), la cual busca entre dos puertos de comunicación serial COM1 y COM2, para determinar el que está conectado al PLC, mediante el envío de caracteres <CR>. Primero prueba con el puerto COM1 y si existe comunicación por este puerto chequea el buffer de ingreso serial si tiene algún dato, lo que indica que el PLC detectó caracteres en su puerto serial y respondió a dicha acción. Inmediatamente el programa indica al usuario que se ha establecido la respectiva comunicación, caso contrario intenta realizarlo por el puerto COM2 bajo el mismo procedimiento descrito. Si resultara que ninguno de los puertos está conectado al PLC el programa envía un mensaje que indica que no existe comunicación con el PLC.

Ya establecida la comunicación y luego de determinado el nombre del archivo ensamblado a enviarse, el programa procede a abrirlo y enviar los datos al PLC en intervalos de tiempo definidos por el usuario, siendo el valor mínimo de 50 milisegundos, valor que, por experimentación se lo ha determinado, puesto que para valores menores a este valor, la comunicación entre PLC y PC presenta problemas de envío debido a que la transmisión serial establecida no tiene ningún control de flujo de datos. Cada caracter enviado está en código ASCII y es transformado a hexadecimal por las subrutinas propias del PLC que también son presentadas en la ventana de información.

Otra de las opciones de esta ventana es la posibilidad de leer la memoria RAM del PLC mediante cuatro páginas de 256 bytes cada una a las que el usuario puede abrirlas indiferentemente.

VENTANARX: Similar a la anterior con la diferencia que solamente es empleado para comprobar la recepción de datos desde el PLC.

MÓDULOS

MODGETCOLFIL: Su finalidad es obtener los valores de columna y fila de los elementos ubicados en el área de diseño.

MODHEX1B: Toma un dato y lo transforma en código hexadecimal de un byte.

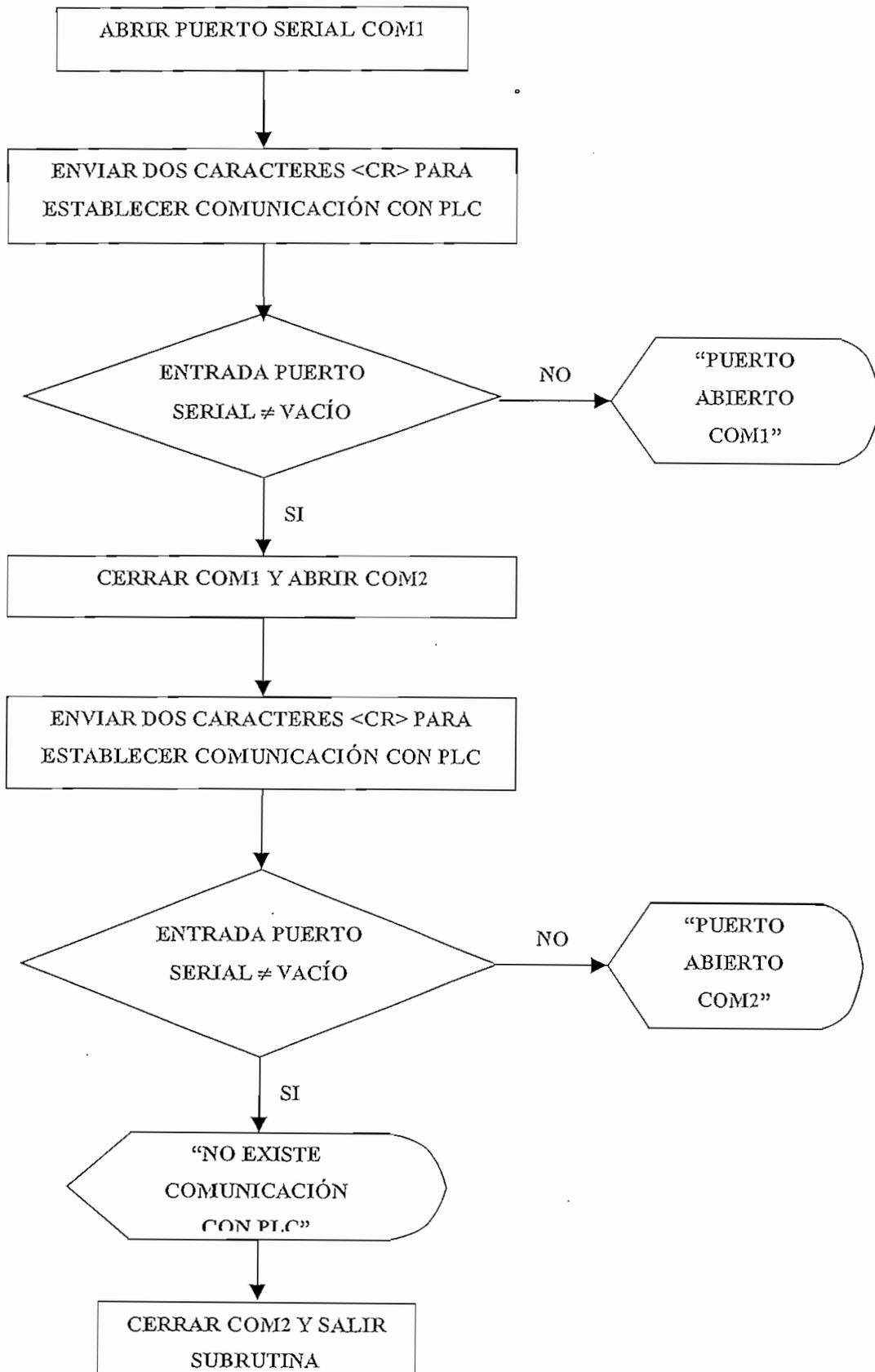
MODHEX2B: Toma un dato y lo transforma en código hexadecimal de dos bytes.

MODINFO: Prepara los datos para ser almacenados en la matriz dato(col, fila).

MODINFO_OUT: Ordena los datos para ser presentados en la ventana de información.

MODLRC: Módulo que transforma datos de código hexadecimal en decimal.

Figura 4.35 Subrutina Abrir Puerto



NUMDOWN: Módulo empleado por las ventanas de introducción de datos, que decrementa del valor numérico introducido cada vez que se pulsa el botón flecha abajo.

NUMUP: Módulo empleado por las ventanas de introducción de datos, que incrementa del valor numérico introducido cada vez que se pulsa el botón flecha arriba.

OBTENCOOR: Módulo para la obtención de coordenadas actuales del cursor.

4.2.3 PROCEDIMIENTOS PARA EL DISEÑO

Como se ha explicado anteriormente el propósito principal de este software es permitir la elaboración de un diagrama de escalera de una manera gráfica sin la necesidad de escribir ninguna clase de instrucciones, constituyéndose así en una herramienta útil para programar al PLC y asimismo el usuario puede visualizar el diagrama de escalera que está desarrollando.

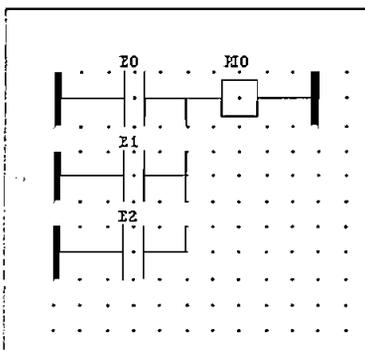
En base a estos antecedentes es importante indicar el proceso que el usuario debe seguir para conseguir programar el PLC, aun cuando para mayores detalles del manejo de este software se presenta en *El Manual del Programador*.

Inmediatamente después que el usuario ingresa al programa mediante el ícono PLC dentro del menú Inicio de la barra de tareas de Windows, se presentará en la pantalla una ventana inicial con la información del nombre del programa, sus autores, número de versión y otros detalles de carácter informativo. Esta ventana permanecerá activa durante un par de segundos aproximadamente ocultándose luego y permitiendo que las ventanas de diseño, de información de datos ingresados y de ensamblado con sus respectivas barras de herramientas aparezcan en pantalla. Cabe aclarar que el usuario no podrá realizar todavía ningún diseño sino hasta cuando abra uno existente o active el proceso de diseño.

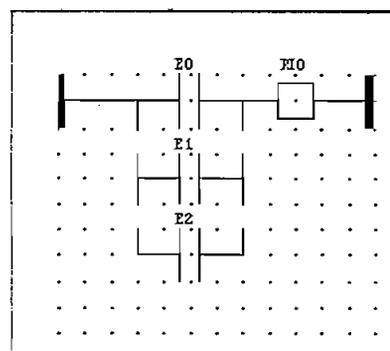
Si el usuario abre un archivo existente entonces se presentará en la ventana de diseño el diagrama de escalera correspondiente a dicho archivo, con toda la información que forman parte de aquel diagrama; de esta manera el usuario puede seguir realizando aditamentos o modificaciones sin la necesidad de activar el proceso de diseño, pues el programa automáticamente ha ingresado en este último modo. Aun más, le es posible guardar los cambios hechos ya sea con el mismo nombre de archivo o con uno diferente. En cambio si el usuario ha decidido trabajar con un nuevo diagrama de escalera, sin que esté presente en pantalla un diagrama previamente abierto o activo, debe acceder al proceso de diseño por medio del menú Proyecto/Diseño de la barra de tareas. Al hacerlo aparece inmediatamente en la ventana de diseño la rejilla, dando al usuario la indicación que el programa está listo para proceder con la elaboración del diagrama de escalera.

Lo que siempre debe hacer el usuario para poder dibujar un nuevo peldaño es ubicar en la columna 0 (cero) sin importar la fila, un extremo INICIO DE PELDAÑO: así el programa podrá interpretar que se va a realizar un nuevo peldaño del diagrama y envía un mensaje para que se confirme si constituye el inicio de un nuevo peldaño. El programa tiene como restricción usar múltiples inicios para un mismo peldaño en caso de utilizar dos o más entradas; en cuyo caso, se sugiere usar la configuración presentada en la siguiente figura:

Figura 4.36 Restricciones de Programación



Configuración NO soportada
por el programa



Configuración equivalente que soporta
el programa (sugerida).

El problema que se presenta al usar la configuración no soportada tiene que ver con el ensamblado ya que el programa está elaborado a detectar cada INICIO DE PELDAÑO y corresponderlo con un peldaño. Si se ensambla con esta configuración el programa no lo hará correctamente enviando un mensaje de error en la ventana de ensamblado.

No es posible ubicar este extremo INICIO DE PELDAÑO en otra columna, pues se presentará el respectivo mensaje de error. Es indispensable también indicar que después de haber ingresado todos los elementos que forman parte del peldaño, se lo termine graficando con un extremo FIN DE PELDAÑO, el cual debe ubicarse exactamente después de cualquier elemento activo, pues el programa puede reconocer que no es posible tener peldaños con solo elementos pasivos, enviando el respectivo mensaje de error. Si por alguna razón el usuario trata de poner cualquier tipo de elemento o conectores después de haber ubicado un elemento activo, se presentará un mensaje de error hasta cuando cierre el peldaño con un FIN DE PELDAÑO o elimine el elemento activo y proceda agregando otros elementos pasivos o conectores.

Cuando se requiera de un extremo, elemento o conector, el usuario debe usar el ratón y movilizar el cursor hasta la barra de herramientas sobre el botón del gráfico que vaya a utilizar, hacer un clic con el botón izquierdo del ratón y desplazarse a la posición donde se ubicará dicho gráfico. Inmediatamente que se haya escogido el gráfico e ingrese a la ventana de diseño, el cursor cambia de puntero de flecha, comúnmente usado en Windows, al puntero de flecha más el ícono del gráfico escogido rotado unos 90°. Se lo ha establecido de esta forma para que se diferencie de los otros gráficos del diagrama

Adjunto a este cursor se encuentra las coordenadas actuales del mismo con el formato (FILA, COLUMNA); así el usuario puede saber la posición actual del cursor, pues puede darse el caso que aparentemente se encuentre en la posición correcta, sin embargo al momento de aplastar el botón izquierdo del ratón, el gráfico se ubique en otra fila o columna aledaña. Esto ocurre únicamente cuando se ubica al cursor en los bordes de las filas o columnas y debido a que la ventana

de diseño no presenta una rejilla de líneas sino de puntos, entonces no se puede percibir los bordes mencionados.

Cada vez que se escoja un elemento, se abrirá la respectiva ventana de introducción de datos, permitiendo que el usuario pueda definir dicho elemento dándole un número de referencia, tipo de elemento, modo de operación, valor de tiempo (para el caso de temporizado) o valor a contar (para el caso de contadores). Estos datos introducidos pasan a formar parte de la matriz de datos: *dato(col, fila)*, matriz muy útil durante todo el proceso de diseño y ensamblado. Además la ventana de información se actualiza cada vez que se grafica un elemento o conector.

En cualquier momento del proceso de diseño, el usuario puede editar su diagrama insertando sea un elemento o un peldaño o eliminar un elemento o un peldaño, dependiendo del fin que busque, asimismo se verá editada la matriz *dato(col, fila)* con todos sus componentes.

El usuario debe tomar muy en cuenta no confundir la aplicación de bobinas de relés internos con los de salida, pues a pesar de usar el mismo botón en la barra de herramientas y sobretodo de poseer el mismo ícono, se diferencian por la etiqueta asignada a cada uno de ellos. La misma advertencia se hace para el caso de temporizados On Delay y Off Delay.

Los scroll bars que forman parte de la ventana de diseño tienen como finalidad permitir desplazarse más allá del área visible de diseño, de esta forma se puede cubrir toda la matriz gráfica establecida.

El programa también realiza ciertas comprobaciones en cuanto a la ubicación de elementos, en especial de conectores, ya que revisa la ubicación de los mismos con respecto a sus aledaños, aplicando de ser necesario, restricciones que a la postre afectarían al ensamblado del diagrama. Se puede graficar cortocircuitos pero durante el ensamblado se los detectará enviando el respectivo mensaje de error, para lo cual el usuario debe realizar las correcciones del caso.

Terminado de diseñar el diagrama de escalera, se procederá al ensamblado, accediendo por medio del menú *Proyecto/Ensamblar*. Inicialmente la subrutina de ensamblado comprueba la existencia de peldaños completos, es decir que a cada INICIO DE PELDAÑO le debe corresponder por lo menos un FIN DE PELDAÑO. Si no se ha completado un peldaño con un FIN DE PELDAÑO, el programa no ensamblará el diagrama y más bien enviará un mensaje de error "Escalera Incompleta". Aquí el usuario debe nuevamente revisar su circuito, en especial si ha dejado peldaños abiertos.

Queda bajo el buen criterio de diseño del usuario el elaborar un diagrama de escalera consistente y funcional, pues el programa puede ensamblar diagramas correctamente graficados pero circuitalmente no funcionales.

Durante el ensamblado, el usuario abrirá un archivo para ensamblado dándole un nombre y automáticamente el programa asignará una extensión (*.HEX) y empezará a ensamblar el circuito. Finalizado este proceso, la ventana de ensamblado presentará el archivo hexadecimal que ha sido ensamblado, tal como se puede ver en la **Figura 4.28** referida anteriormente.

CAPÍTULO 5.

ANÁLISIS ECONÓMICO

Se presenta un estudio comparativo del costo del diseño y construcción de este PLC con existentes en el mercado de similares características y funciones.

El objetivo es conocer cuan costoso es construir un PLC que cumpla con los requerimientos básicos necesarios para la industria ecuatoriana y que además contenga elementos de fácil adquisición en el mercado local.

5.1 DESCRIPCIÓN DE LOS ELEMENTOS USADOS:

Ha sido necesario la construcción de cuatro tarjetas para ensamblar al PLC:

- Tarjeta de Control Principal.
- Tarjeta de Entradas / Salidas.
- Tarjeta de Visualización de Entradas, Salidas y Periféricos.
- Tarjeta de Fuente.

5.1.1 ELEMENTOS UTILIZADOS EN TARJETA DE CONTROL PRINCIPAL.

- 1 Microcontrolador MCS 8751 u otro de la serie MCS - 51.
- 1 Latch 74LS573.
- 1 MAX 232 para comunicaciones
- 1 Cristal de 3.45 MHz.
- 1 Memoria EPROM de 64 Kbytes.
- 1 Memoria RAM de 64 Kbytes.
- 8 Resistencias de 10 K Ω .
- 6 Capacitores de 10 μ F.
- 2 Transistores 2N2222A.
- 3 Diodos led.

- 2 Resistencias de 330Ω .
- 1 Pulsante con contacto normalmente abierto.
- 2 Zócalos de 28 pines.
- 1 Zócalo de 20 pines.
- 1 Zócalo de 40 pines.
- 1 Conector DB9.
- 1 Conector 2*10 pines.
- 1 Tarjeta impresa

5.1.2 ELEMENTOS UTILIZADOS EN ETAPAS DE ENTRADAS

- 1 Registro de desplazamiento paralelo serial NTE 4021B.
- 8 Optoacopladores ECG 3041.
- 8 Diodos led.
- 8 Resistencias $1\text{ K}\Omega$ $\frac{1}{4}\text{ W}$.
- 8 Resistencias $330\ \Omega$ $\frac{1}{4}\text{ W}$.
- 8 Zócalos de 6 pines.
- 1 Zócalo de 16 pines.
- 1 Conector 2*5 pines.
- 1 Tarjeta impresa.

5.1.3 ELEMENTOS UTILIZADOS EN ETAPA DE SALIDAS

- 1 Amplificador de corriente NTE 2018.
- 1 Registro de desplazamiento NTE 4094.
- 8 Diodos led rojos.
- 8 Resistencias 10 K $\frac{1}{4}\text{ W}$.
- 8 Relés con un contacto normalmente abierto.
- 8 Zócalos para relé.
- 1 Tarjeta impresa.

5.1.4 ELEMENTOS UTILIZADOS EN TARJETA DE FUENTE

Los elementos utilizados en esta etapa son los siguientes:

- 1 Transformador 110Vac / 12Vac 1000mA.
- 1 Puente rectificador en onda completa ECG5332 ó 4 diodos.
- 1 Regulador ECG 7805.
- 4 Capacitores 1000 μ F, 100 μ , 1 μ , 0.1
- 1 Fusible de vidrio de 1 A.
- 1 Base fusible de tarjeta.
- 1 Batería Ni-Cd.
- 1 Tarjeta impresa

5.1.5 ELEMENTOS GENERALES UTILIZADOS

- Conectores.
- Cable Plano.
- Caja de protección.
- Placa frontal de presentación.
- Elementos varios.

Los costos de todos los elementos utilizados en la construcción de este PLC se detallan en la **Tabla 5.1**.

Tabla 5.1 Lista de Precios de Elementos para PLC

Elemento	Cant.	V. Unit.	V. Total	Descripción
Construcción de tarjetas	4	30.00	120.00	Circuitos Impresos
Microcontrolador 8751	1	22.00	22.00	CPU
Memoria ds1230Y RAM	1	10.00	10.00	CPU
Memoria 27256 EEPROM	1	14.00	14.00	CPU
Latch 74HC573	1	6.00	6.00	CPU
MAX 232	1	9.00	9.00	Transmisión Serial CPU
4021 Registro desplazamiento	1	4.00	4.00	Desplazamiento de Entradas
4094 Registro desplazamiento	1	4.00	4.00	Desplazamiento de Salida
4018 Amplificador corriente	1	2.00	2.00	Amplificador para relés
Optoacopladores 3041	8	0.90	7.20	Aislar entradas al CPU
Cristal	1	3.00	3.00	Oscilador para micro.
Relés de control	8	4.50	36.00	Elementos finales de control
Transformador 12 V/1 ^a	1	4.00	4.00	Transformador reductor
7805 Regulador 5 V.	1	1.50	1.50	Regulador de Voltaje
Resistencias varias	40	0.10	4.00	
Leds	24	0.20	4.80	Visualización en tarjetas
Disipador de calor	1	1.50	1.50	Fuente
Fusible 1A.	1	0.30	0.30	Protección de Alimentación
Base fusible	1	0.50	0.50	Fuente
Batería para RAM	1	3.50	3.50	Batería de respaldo para RAM
Rectificador Onda Completa	1	1.50	1.50	
Diodos de polarización	2	0.50	1.00	
Condensadores Varios	15	0.40	6.00	
Zócalos 40 pines	1	1.00	1.00	
Zócalos 28 pines	2	1.00	2.00	
Zócalos 20 pines	1	1.00	1.00	
Zócalos 16 pines	10	0.60	6.00	
Zócalos 18 pines	2	0.60	1.20	
Zócalos 6 pines	8	0.60	4.80	
Conectores	4	2.00	8.00	
Elementos varios	1	10.00	10.00	
Placa Frontal	1	15.00	15.00	
Caja de protección	1	20.00	20.00	
TOTAL			334.80	

5.2 DESARROLLO DE SOFTWARE

El desarrollo del software para el PLC consta de tres partes:

- Para el microprocesador.- Se usó como herramienta de diseño el software denominado PINNACLE 52.

- Para la computadora.- Como herramienta de diseño se ha usado el Lenguaje de programación Visual Basic versión 5.0.
- Para el ruteado de tarjetas impresas.- Se usó como herramienta de Diseño el programa Designer Explorer.

Los costos incurridos en las diferentes etapas del diseño del software se indican en la **Tabla 5.2**.

Tabla 5.2 Costos de Desarrollo de Software

Software Desarrollado	H. H.	H. E.	Costo/HH	Costo/HE	Costo Total
Para microprocesador	200	200	8.00	0.50	1700.00
Para computadora	300	300	8.00	0.50	2550.00
Para tarjetas impresas	10	10	8.00	0.50	85.00
TOTAL					4335.00

Donde:

H.H. Horas / Hombre

H.E. Horas / Equipo

Una vez desarrollado un PLC, para los siguientes equipos que se desee construir, no se requiere realizar gastos en desarrollo de software, cuando no se requiere modificaciones.

5.3 VALOR DE MERCADO DEL PLC

Se analizará el valor de mercado al que se puede comercializar el equipo en el mercado nacional considerando una utilidad del 20% y un 15% para gastos administrativos en los que se hallan incluidos los gastos que se requieren para publicidad. Se analiza el valor de mercado en consideración con el número de equipos vendidos, se determina el número óptimo de equipos a vender y el precio de cada uno.

El valor de mercado en función a la cantidad de unidades vendidas se analiza en la **Tabla 5.3**.

Tabla 5.3 Valor en Mercado de PLC.

No. Equipos	Construcción	Software	Utilidad	G. Admin.	VMU
1	334.8	4335	20%	15%	6304.23
2	334.8	4335	20%	15%	3378.11
3	334.8	4335	20%	15%	2402.73
4	334.8	4335	20%	15%	1915.04
5	334.8	4335	20%	15%	1622.43
6	334.8	4335	20%	15%	1427.36
7	334.8	4335	20%	15%	1288.02
8	334.8	4335	20%	15%	1183.51
9	334.8	4335	20%	15%	1102.23
10	334.8	4335	20%	15%	1037.21
20	334.8	4335	20%	15%	744.59
30	334.8	4335	20%	15%	647.06
40	334.8	4335	20%	15%	598.29
50	334.8	4335	20%	15%	569.03
60	334.8	4335	20%	15%	549.52
70	334.8	4335	20%	15%	535.58
100	334.8	4335	20%	15%	510.50
140	334.8	4335	20%	15%	493.78
150	334.8	4335	20%	15%	491.00
160	334.8	4335	20%	15%	488.56
200	334.8	4335	20%	15%	481.24
300	334.8	4335	20%	15%	471.49
400	334.8	4335	20%	15%	466.61
500	334.8	4335	20%	15%	463.68
600	334.8	4335	20%	15%	461.73
700	334.8	4335	20%	15%	460.34
800	334.8	4335	20%	15%	459.30
900	334.8	4335	20%	15%	458.48
1000	334.8	4335	20%	15%	457.83

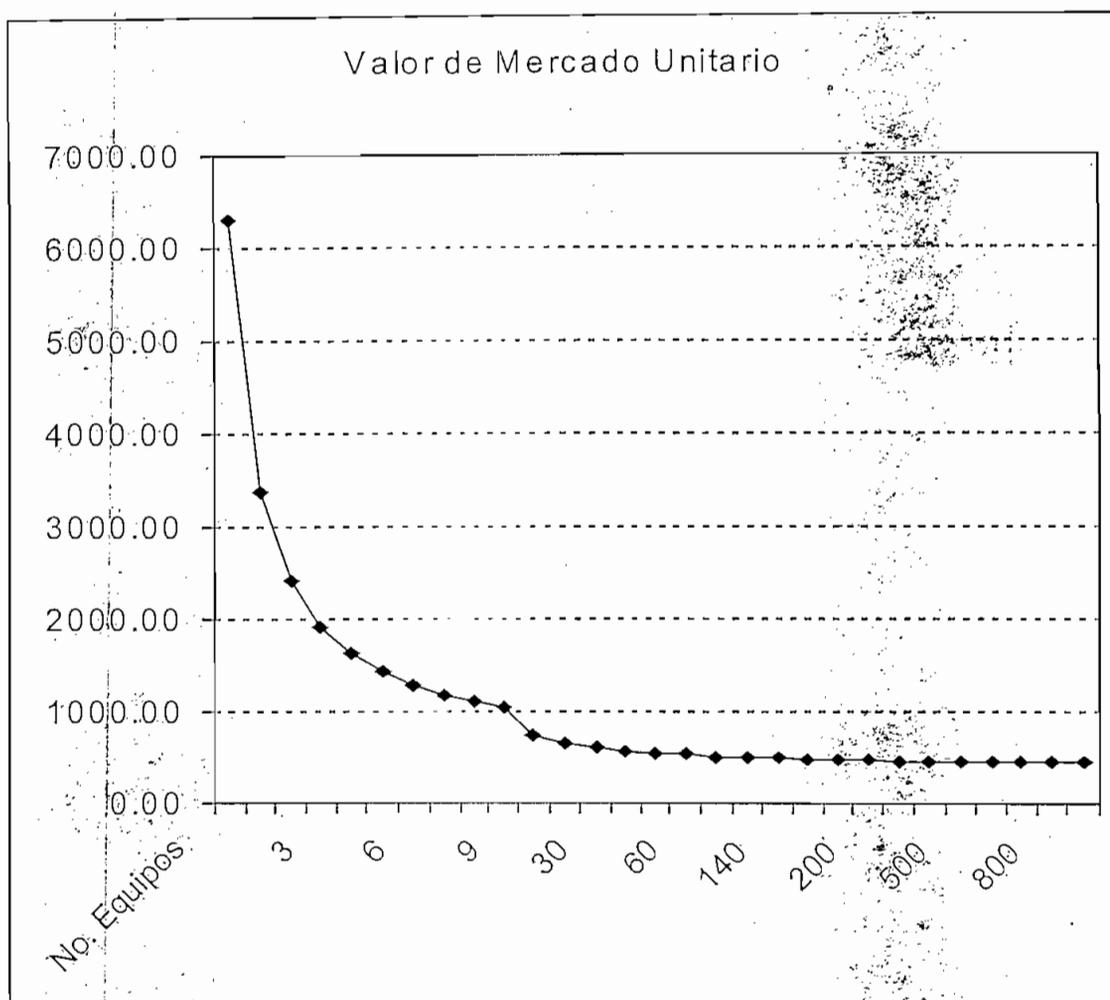
Donde:

VMU.- Valor de Mercado Unitario

G. Admin.- Gastos Administrativos

En el **Figura 5.1** se aprecia fácilmente la variación del valor de mercado unitario en función del número de unidades vendidas y se ha determinado como un precio de mercado ideal: **500 dólares** por lo que el número de unidades a vender debe ser mínimo de 100 unidades.

Figura 5.1 Valor de Mercado Unitario vs. No. Unidades Vendidas



En la **Tabla 5.4** se indica el costo de mercado de algunas marcas de PLC existentes en el mercado local con características similares al PLC diseñado.

Los fabricantes proveen elementos auxiliares como:

- Programadores manuales.
- Software para programación.
- Software para monitoreo y control.
- Programadores de tiempo.

Tabla 5.4 Valores de Mercado de Algunos PLC.

Marca	I/O	Elementos Auxiliares	VMU	Características
Cuttler Hammer	20IO	Conversor RS232	750	Software Incluido
Siemens	16IO	Software	1000	-
Omron	16IO	Programador	600	No incluye Software
Allen Bradley	14IO	Conversor RS232	480	-
L.G.	8I6O	Conversor RS232	310	Software Incluido

Comparando con la marca L.G. el PLC diseñado tiene un costo muy superior, sin embargo al observar sus especificaciones técnicas se puede apreciar que este equipo tiene únicamente dos temporizados, no tiene contadores de eventos, sus salidas de relés manejan cargas de hasta 1 amperio, tiene un número menor de salidas y el software incluido no tiene garantía por parte de la empresa distribuidora por lo que el precio superior se justifica.

Con las demás marcas se aprecia un valor de mercado similar al establecido para el PLC diseñado, en este caso el análisis se orienta a la complejidad de programación que otras marcas presentan, lo que requiere un estudio previo del software y su funcionamiento, también poseen funciones adicionales que para la aplicación que se orienta no son necesarias por lo que se está pagando más por funciones que no se van a utilizar.

Es difícil estimar el nivel de industrialización que las empresas ecuatorianas poseen, pero se conoce que para pequeñas y medianas industrias no se ha logrado un nivel de industrialización por lo que este trabajo va orientado a la automatización de estas empresas que no disponen un gran capital y requieren mantenimiento local de sus máquinas y equipos.

CAPÍTULO 6.

PRUEBAS Y RESULTADOS

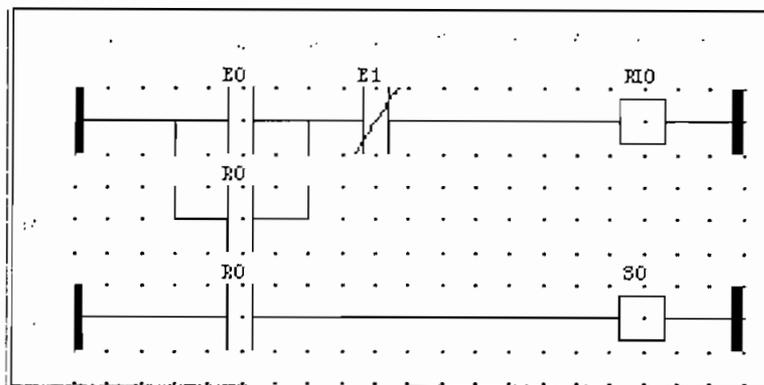
Se describe unas pocas aplicaciones y ejemplos de entre muchos que es posible realizar en el PLC; el límite para cualquier aplicación que se pudieran presentar en la industria está dada por el proceso a automatizar.

Uno de los circuitos más usados en control industrial es el denominado enclavamiento eléctrico.

6.1 ENCLAVAMIENTO ELÉCTRICO

El diagrama para esta aplicación se presenta en la **Figura 6.1**. Al dar un pulso se cierra una señal de salida y con otro pulso se desactiva la señal de salida.

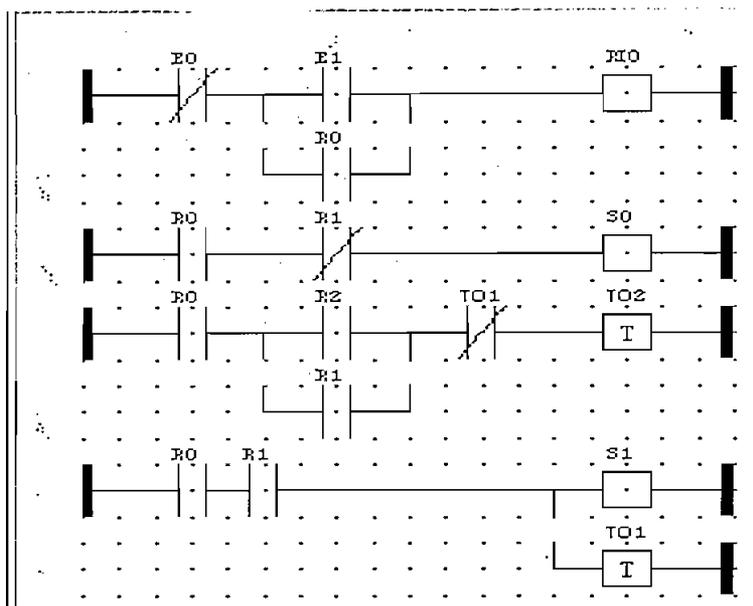
Figura 6.1 Enclavamiento Eléctrico.



6.2 RELÉ DE PULSOS

Se puede necesitar un relé de pulsos, el cual no se halla programado en el PLC; se lo puede construir usando dos temporizados acoplados como se detalla en la **Figura 6.2**. El tiempo del pulso en apertura y cierre puede ser programado independientemente para apertura y cierre.

Figura 6.2 Relé de Pulsos.



Este PLC tiene los mismos limitantes en cuanto a tiempo de conmutación presentados por los PLC comerciales, por ejemplo se menciona que en enclavamiento entre dos elementos internos dentro del PLC ocasionará que ninguna de las señales actúe, básicamente porque la conmutación en los dos casos es casi al mismo tiempo, el mando alternado no se puede realizar de la manera clásica de control industrial.

Se presentan los diagramas para 4 diferentes aplicaciones:

- Semáforo para control de tráfico vehicular.
- Control de nivel en un tanque de almacenamiento de líquidos.
- Arranque Estrella Triángulo.
- Arranque con control de tiempo para 8 motores de alta potencia.

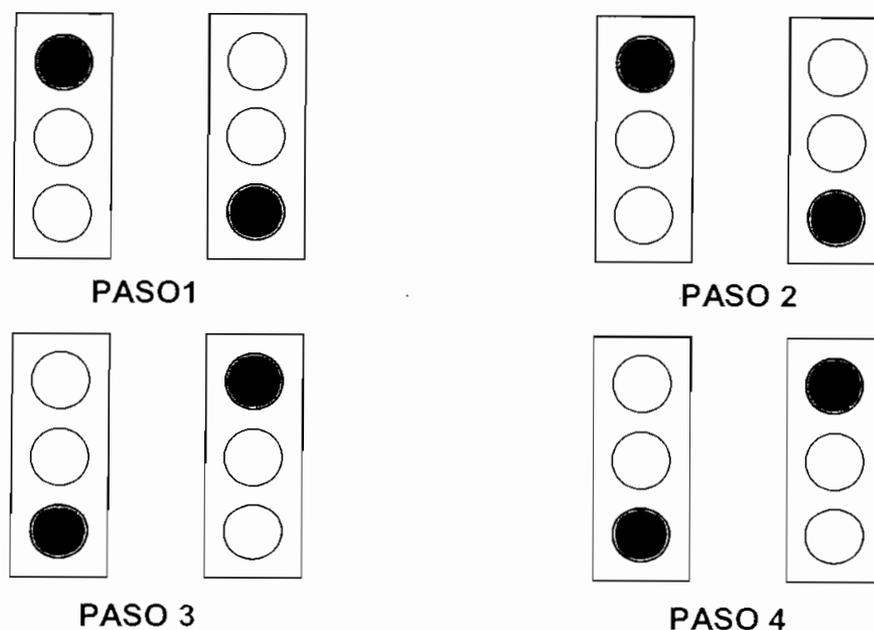
6.3 SEMÁFORO PARA CONTROL VEHÍCULAR.

El semáforo mostrado tendrá las siguientes características:

- Se controlará el encendido y apagado mediante un interruptor On/Off.
- Empieza con luz encendida en verde en un lado de la vía y por otro lado luz encendida en rojo; al transcurrir un tiempo programado (T_0), se enciende junto a la luz verde, la indicación preventiva amarilla, después de un tiempo programado (T_1), se produce la transición a luz roja en el sentido verde y viceversa; al mismo tiempo se apaga la luz preventiva amarilla. En estas circunstancias, transcurre un tiempo (T_2) y se enciende la luz amarilla junta a con la luz verde; transcurre un tiempo (T_3) y vuelve a posición inicial, repitiéndose el ciclo indefinidamente hasta accionar el interruptor a la posición de apagado.

La secuencia de operación del semáforo se la puede ver gráficamente representada en 4 pasos como se indica en la **Figura 6.3**:

Figura 6.3 Secuencia de Funcionamiento de un Semáforo

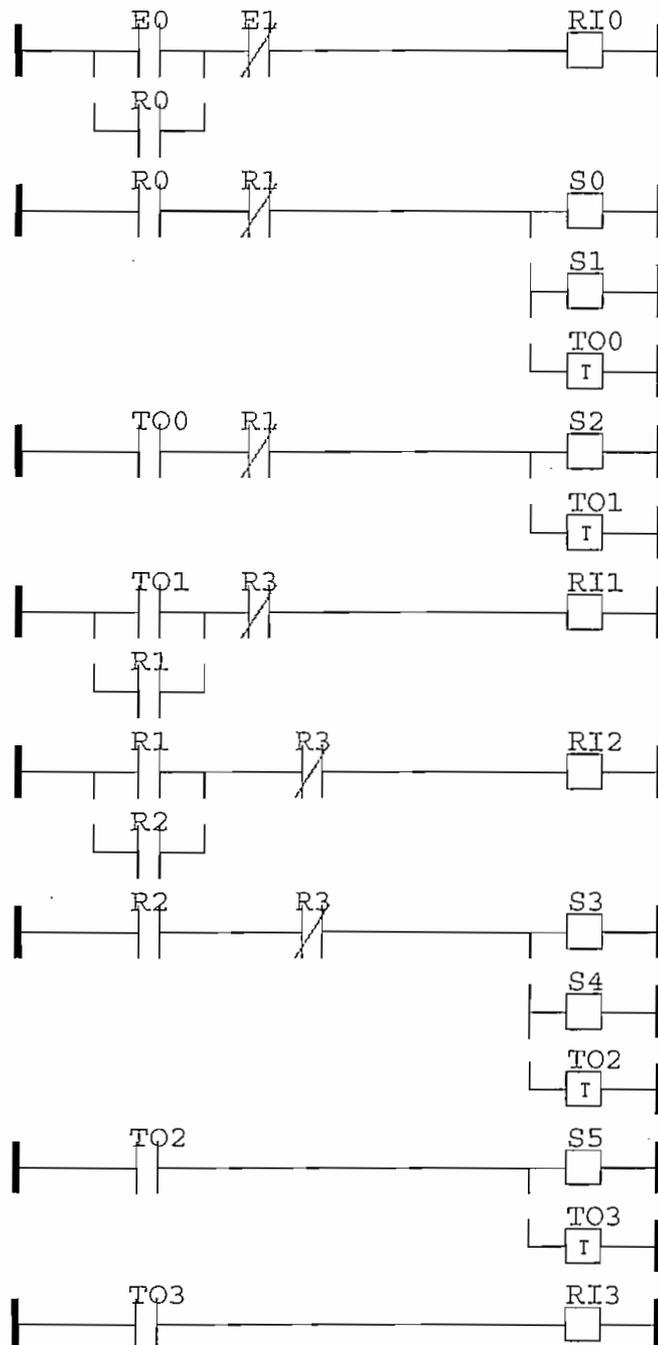


Una variación que se puede presentar no realizada en el diseño, es colocar sensores de paso del número de vehículos que transitan; después que cierto

número de vehículos ha circulado, hacer la transición al estado contrario, esta modificación se la puede realizar en ambientes de alto tráfico.

El diagrama para el circuito mencionado se indica en la **Figura 6.4:**

Figura 6.4 Semáforo para Control Vehicular



6.4 CONTROL DE NIVEL EN UN TANQUE DE ALMACENAMIENTO DE LÍQUIDOS.

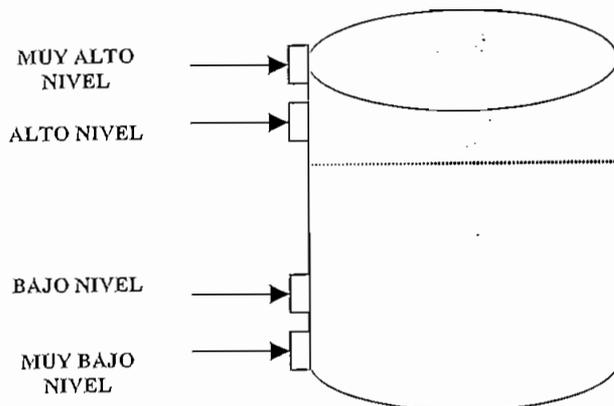
En el tanque de almacenamiento de líquidos indicado, se tienen ubicados 4 sensores, alto nivel, muy alto nivel, bajo nivel, muy bajo nivel, adicionalmente se dispone de dos bombas para el llenado del líquido.

Las consideraciones para el diseño indicado son las que a continuación se detallan:

- El sistema puede trabajar en manual o automático; en manual mediante pulsantes de marcha paro se acciona las bombas de llenado de líquido independientemente.
- En automático, la bomba # 1 arranca una vez que el nivel haya bajado del mínimo; si el nivel baja a muy bajo nivel, la bomba # 2 entra a trabajar simultáneamente.
- Las bombas #1 y # 2 se apaga con el sensor de alto nivel
- El sensor de muy alto nivel apaga las 2 bombas y bloquea el sistema hasta que el operador reinicie el control.
- Las bombas conmutan de uno a dos después de 100 operaciones.
- El PLC enviará una señal de error en caso de sobrecarga de alguna de las bombas.

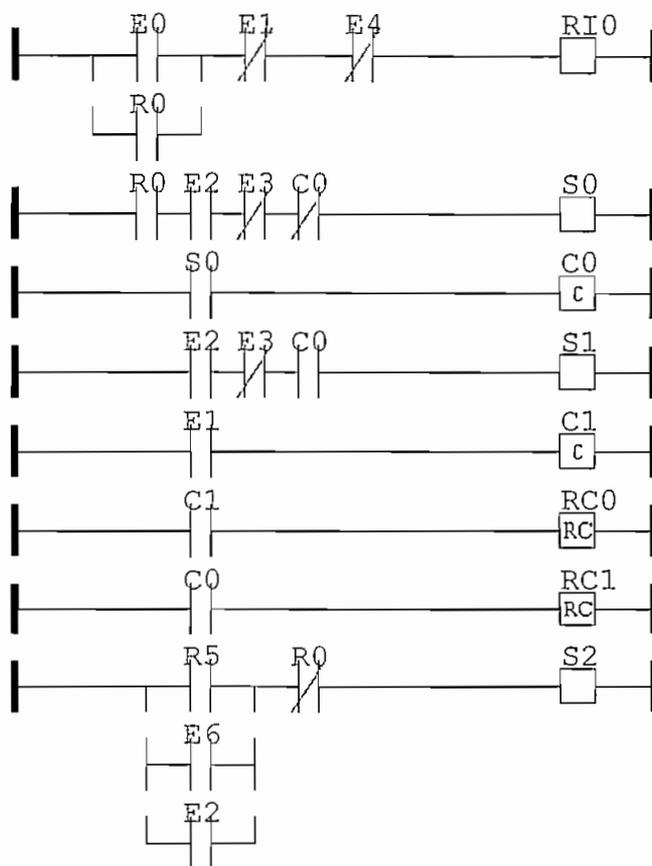
El tanque con sus elementos se indica en la **Figura 6.5**.

Figura 6.5 Elementos de Control de Tanque de Líquidos



El diagrama de control realizado se indica en la **Figura 6.6**.

Figura 6.6 Control de Nivel de un Tanque de Líquidos.

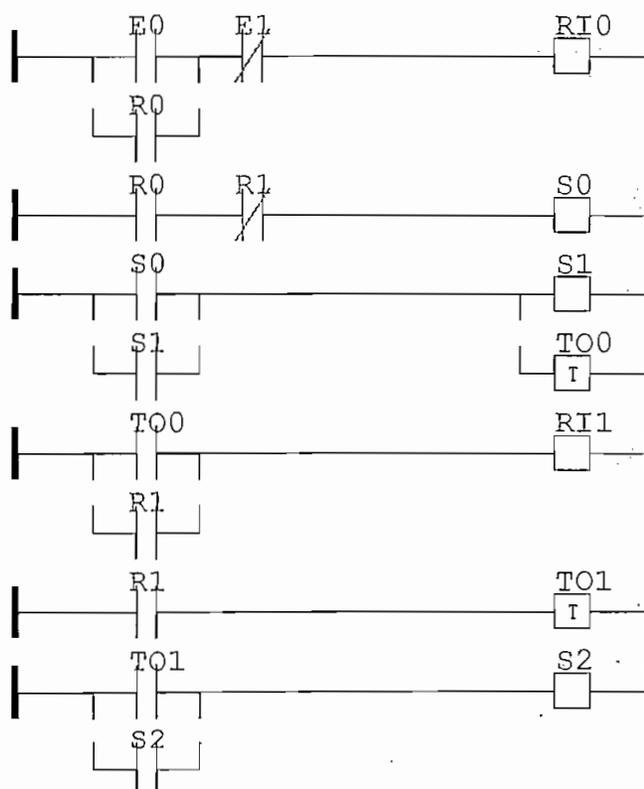


6.5 ARRANQUE ESTRELLA TRIANGULO

Con un pulsante de arranque (Entrada 0) se conecta la Salida 0 que activa al contactor estrella del motor; al mismo tiempo la salida 1 acciona el contactor de red, la Salida 1 se realimenta con un contacto propio. El tiempo de conmutación se regula mediante un temporizado On Delay conectado al mismo tiempo que el contactor de red

El diagrama de control se indica en la **Figura 6.7**.

Figura 6.7 Arranque Estrella Triángulo.



CAPÍTULO 7.

CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIONES

- El PLC diseñado presenta una alternativa más para la industria ecuatoriana y cumple con los requerimientos que ella plantea. Su operación confiable permitirá la automatización de pequeños procesos industriales y a partir de su utilización y manejo la satisfacción del cliente de manera de lograr su difusión en toda la industria.
- El programa es muy amigable y fácil de aplicar, contiene inclusive pequeñas ayudas para identificar errores de programación.
- Las herramientas de programación utilizadas fueron las adecuadas; PINNACLE 52 presenta ventanas de programación en Assembler que permitieron trabajar en un ambiente más amigable que los programadores tradicionales. Asimismo Visual Basic versión 5.0 permitió cumplir con las expectativas planteadas y desarrollar la mayoría de funciones gráficas y de comunicación; es un lenguaje muy amigable.
- Para realizar aplicaciones es necesario que el operador tenga conocimientos básicos de control automático y realizar las aplicaciones sin conectar elementos de salida para no provocar accidentes por una mal programación del equipo.
- Con el análisis económico se determinó que el PLC diseñado tiene un costo comparativo con el mercado local con algunas ventajas adicionales; como son:
 - La posibilidad de realizar modificaciones en las funciones programadas.

- Se deja abierta la ventana tecnológica para realizar ampliaciones en las funciones del PLC como:
 - Realizar monitoreo del proceso en tiempo real.
 - Hacer modificaciones con el equipo en operación.
 - Ampliar la capacidad de entradas y salidas
 - Cambiar el tipo de salida disponible
 - Agregar un módulo de señales analógicas al control.
 - Agregar más elementos de control.

- El programa es muy amigable y fácil de aplicar, contiene inclusive pequeñas ayudas para identificar errores de programación.

- El programa puede ser usado parcialmente en otras aplicaciones como la transmisión de datos para programar memorias con las debidas correcciones.

- Para realizar aplicaciones es necesario que el operador tenga conocimientos básicos de control automático y realizar las aplicaciones sin conectar elementos de salida para no provocar accidentes por una mal programación del equipo.

- El software de la computadora es simple, amigable y contiene todos los elementos necesarios para la programación del PLC. Presenta adicionalmente una ventana que permite comprobar el funcionamiento y comunicación.

- Las herramientas de programación utilizadas fueron las adecuadas; PINNACLE 52 presenta ventanas de programación en Assembler que permitieron trabajar en un ambiente más amigable que los programadores tradicionales. Asimismo Visual Basic versión 5.0 permitió cumplir con las expectativas planteadas y desarrollar la mayoría de funciones gráficas y de comunicación; es un lenguaje muy amigable.

7.2 RECOMENDACIONES

- Para bajar el costo del PLC se recomienda el cambio del microcontrolador MCS 8751 por un microprocesador MCS 8031 que son compatible pin a pin y además por la configuración de hardware propuesta trabaja sin ninguna modificación adicional. Sería conveniente además que se realizaran futuras investigaciones utilizando como herramientas los nombrados PICs muy populares en los actuales momentos aunque no muy comerciales.
- Es importante continuar con el trabajo aquí empezado, se plantea como inquietud el desarrollo de una segunda parte de este trabajo, una de las alternativas propuestas es el monitoreo en tiempo real de las señales de los diferentes elementos de control, el proceso requiere de un manejo adecuado de los registros de almacenamiento de señales de elementos de control tanto activos como pasivos que se detallan en la presente tesis, así como un manejo amplio del lenguaje de programación Visual Basic 5.0.
- Si se va a realizar aplicaciones mayores se recomienda ampliar la capacidad de memoria RAM disponible para el usuario, a fin de tener mayor cobertura de datos a manejar.
- Al desarrollar aplicaciones con elementos electrónicos de control sensibles a mal manejo se recomienda tomar las debidas precauciones sobretodo en la fuente de alimentación que se utiliza.
- Se recomienda investigar ampliamente el mercado local para conocer los elementos sobre todo de montaje final disponibles, se experimentó un serio inconveniente al tratar de repetir el mismo proceso dos veces por carencia de materiales; también se comprobó la amplia variedad de precios sobre los mismos elementos que los diferentes distribuidores mantienen.

BIBLIOGRAFÍA

- **ANGULO USATEGUI JOSE** "Microprocesadores: Fundamentos, Diseño y Aplicaciones en la Industria y en los Microprocesadores"
Editorial PARANINFO, 4ta edición, 1985, España
- **ANGULO USATEGUI JOSE** "Microprocesadores: Curso sobre Aplicaciones en Sistemas Industriales"
Editorial PARANINFO, 4ta edición, 1985, España
- **BOYLESTAD ROBERT** "Electrónica: Teoría de Circuitos"
Editorial PRENTICE / HALL, 2da. Edición, 1983, Colombia
- **ECG** "Semiconductors I, II Y III"
Philips, 6ta. Edición, 1986
- **ECG** "Digital Integrated Circuits Data Manual"
Philips, 1981, 5ta edición, USA
- **HALVORSON** "Aprenda Visual Basic 5 Ya"
McGraw Hill, 1997, España
- **INTEL** "Microprocesador MCS 8X51"
Intel, 1981, USA
- **MICROSOFT PRESS** "Visual Basic 5 Lenguaje Reference"
Microsoft, 1997, USA
- **OMRON** "Operation Manual CPU01 Model C200H"
Omron, 1992, Japón
- **OMRON** "INSTALLATION GUIDE: SYSMAC Programmable Controller C200H" Omron, 1992, Japón
- **PHILIPS** "Data Handbook : Integrated Circuits"
1988 Noruega
- **PHILIPS** "ECG Semiconductors Technical Manual"
1985 1era edición Canadá
- **PHILIPS** "Components and Materials"
1981 Noruega

- **SIEMENS** "Comunicación Industrial"
1998 Alemania
- **SMAR** "The Fieldbus Book"
2001 USA
- **WILLIAMS** "Manual de Circuitos Integrados"
TOMOS I, II, III, IV
- www.lawebdelrpogramador.com
- www.planetsourcecode.com
- www.portalvb.com
- www.systronix.com

ANEXOS

ANEXO A.1

FOTOGRAFÍAS

FOTO 1: VISTA DEL SUPERIOR DEL PLC

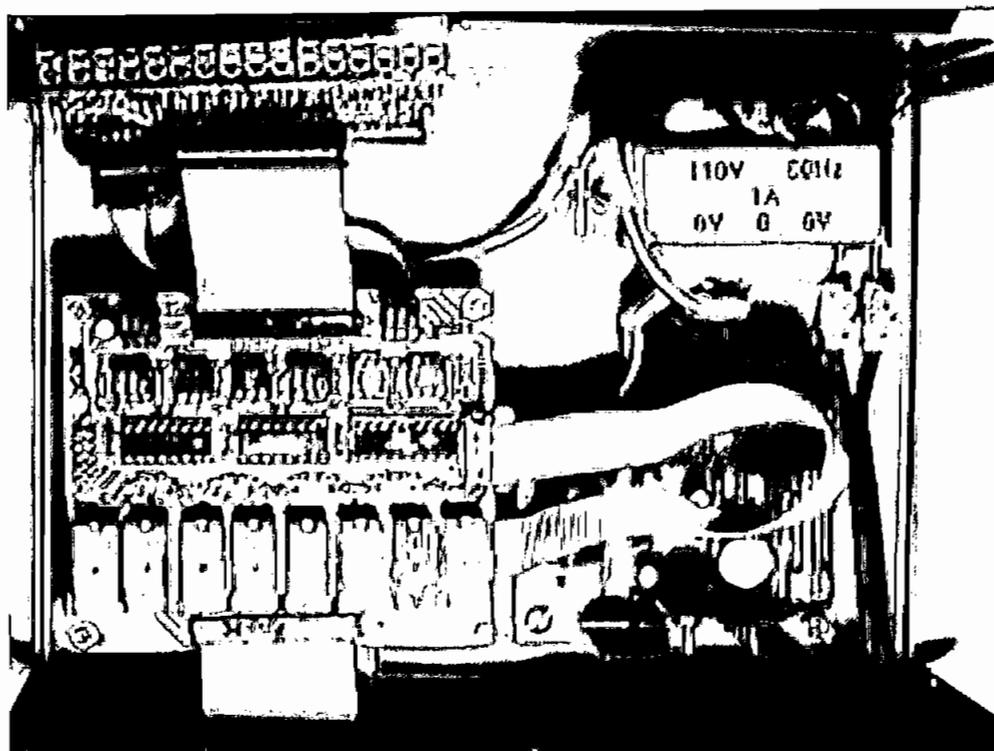


FOTO 2: TARJETA DE CONTROL PRINCIPAL

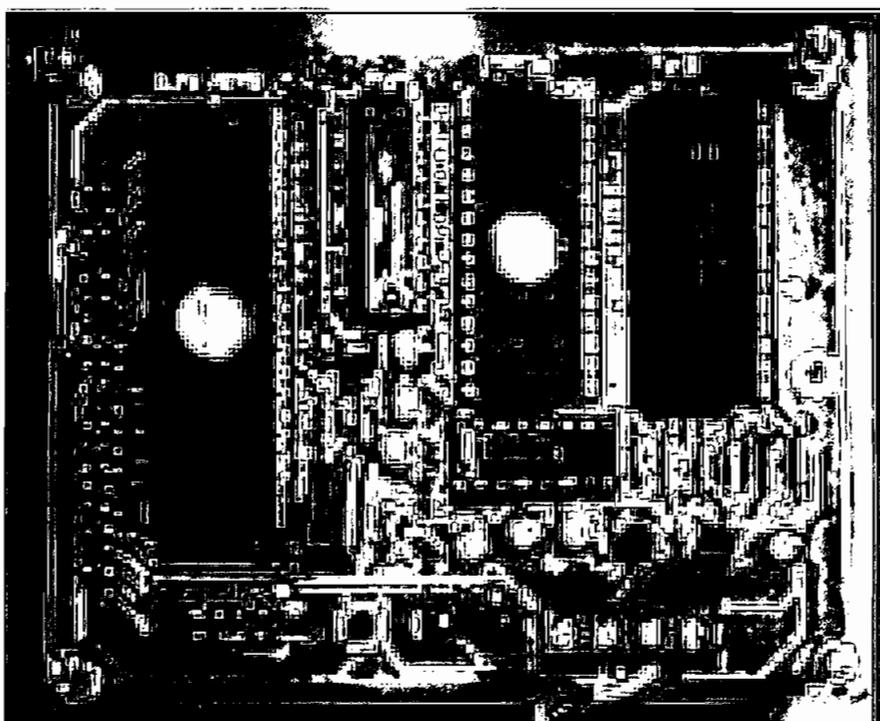


FOTO 3: TARJETA DE ENTRADAS-SALIDAS

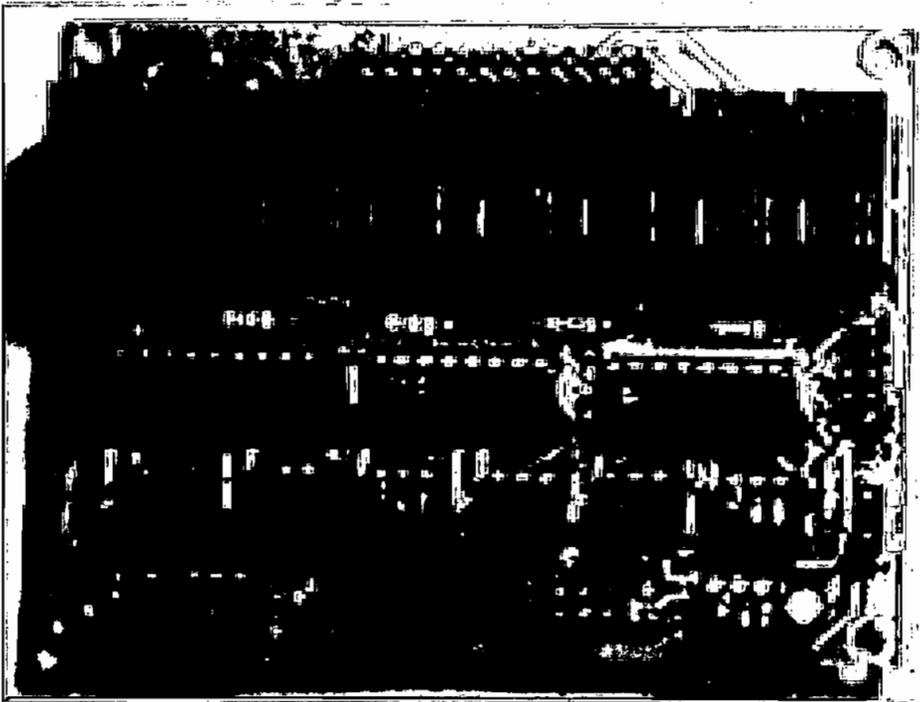
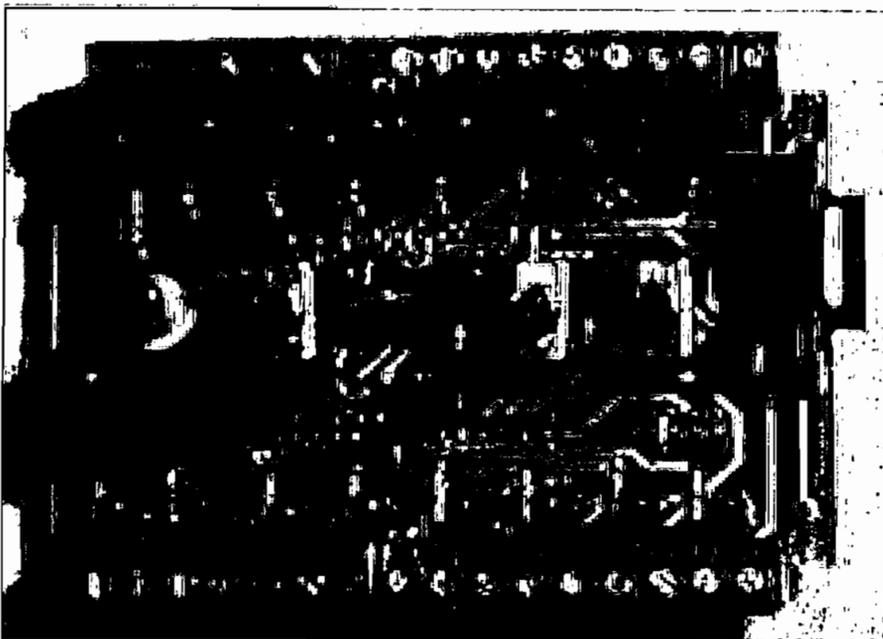


FOTO 4: TARJETA DE ELEMENTOS DE VISUALIZACIÓN



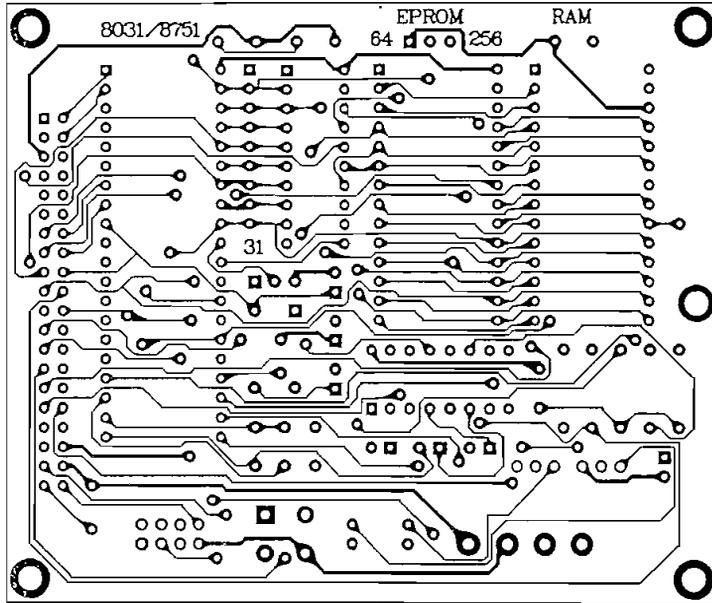
ANEXO A.2

DIAGRAMAS ESQUEMÁTICOS

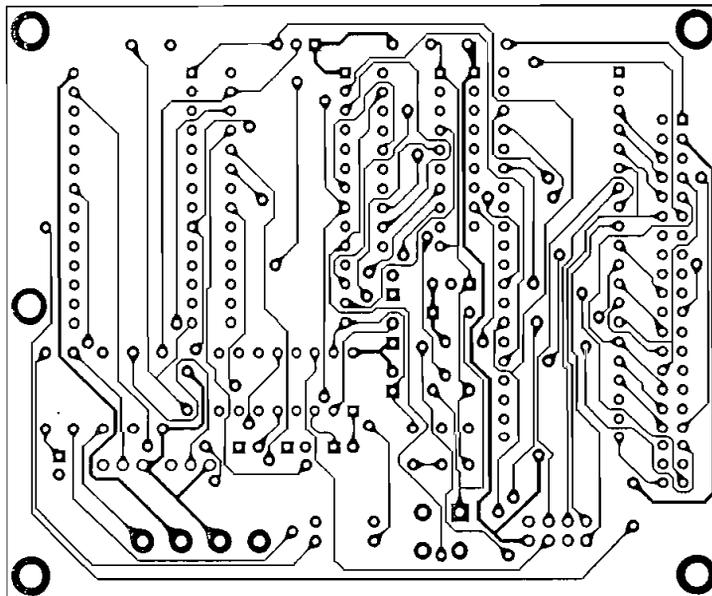
ANEXO A.3

RUTEADOS DE TARJETAS

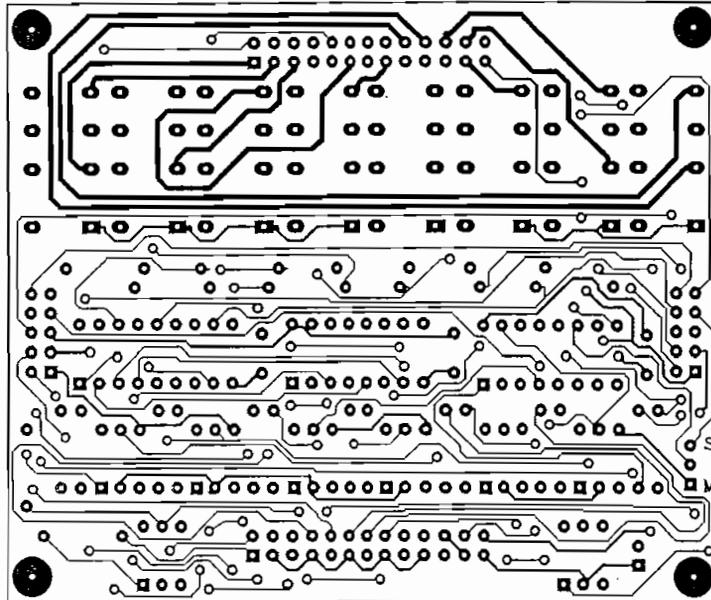
TARJETA DE CONTROL PRINCIPAL
Lado de Elementos



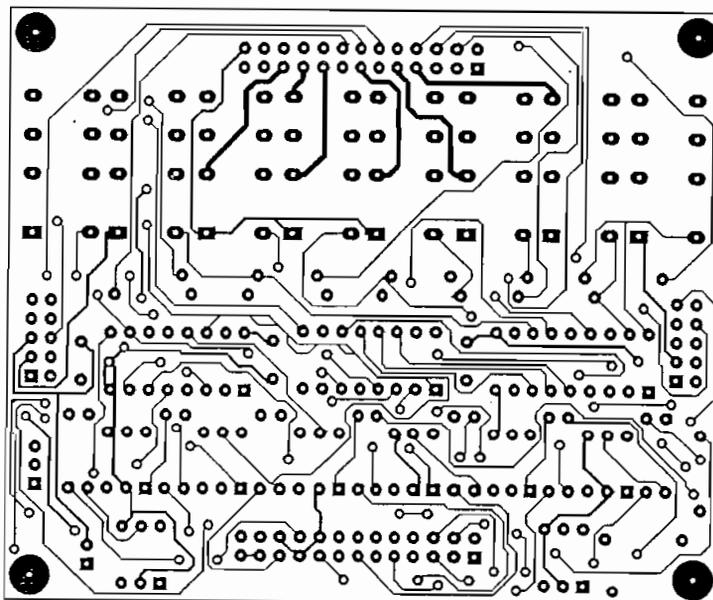
TARJETA DE CONTROL PRINCIPAL
Lado de Soldadura



TARJETA DE ENTRADAS SALIDAS
Lado de Elementos

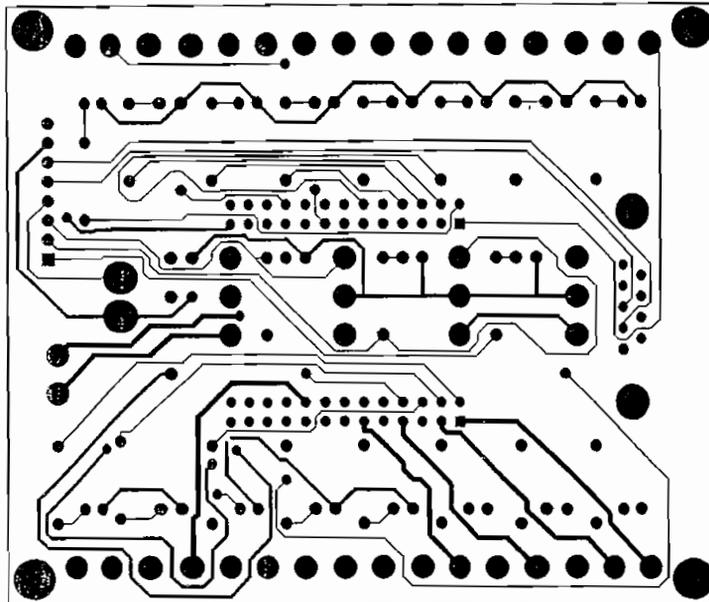


TARJETA DE ENTRADAS SALIDAS
Lado de Soldadura



TARJETA DE ELEMENTOS DE INDICACIÓN Lado de Elementos

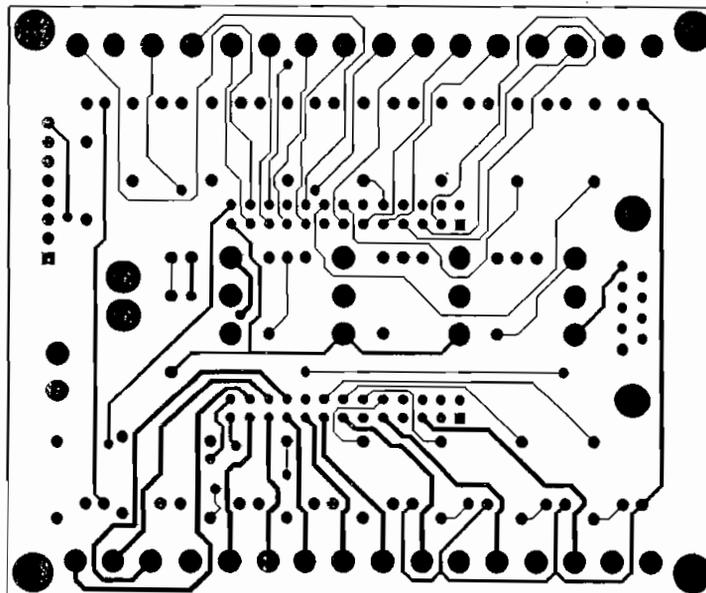
entradas



salidas

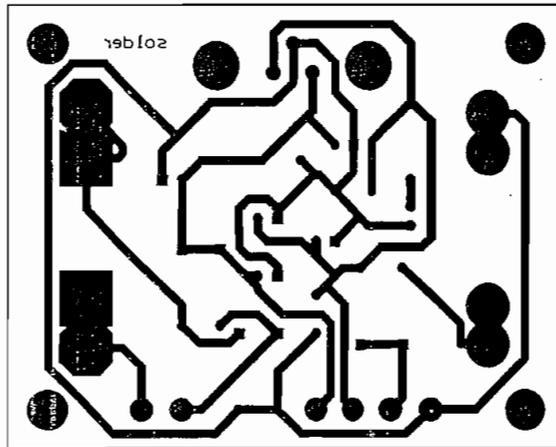
TARJETA DE ELEMENTOS DE INDICACIÓN Lado de Soldadura

entradas

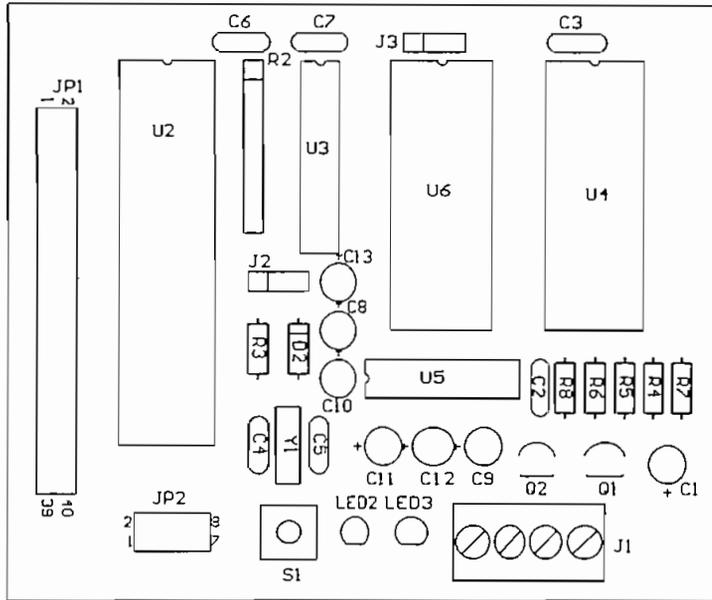


salidas

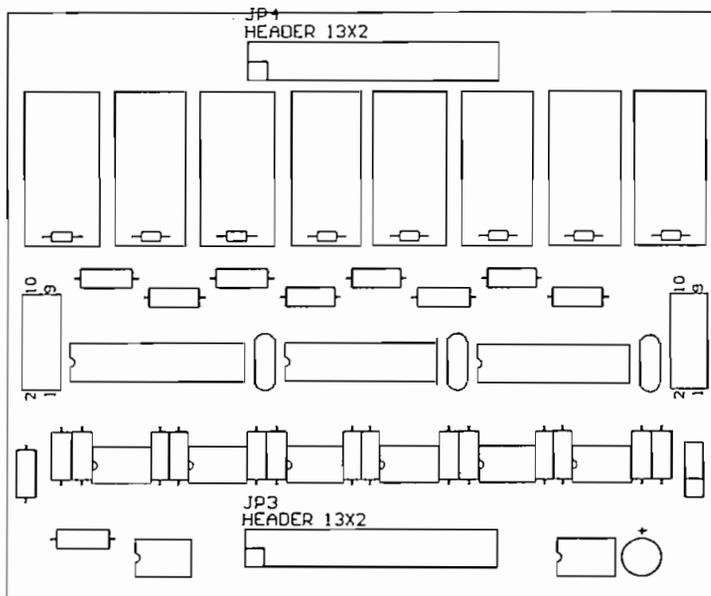
TARJETA DE FUENTE
Lado de Soldadura



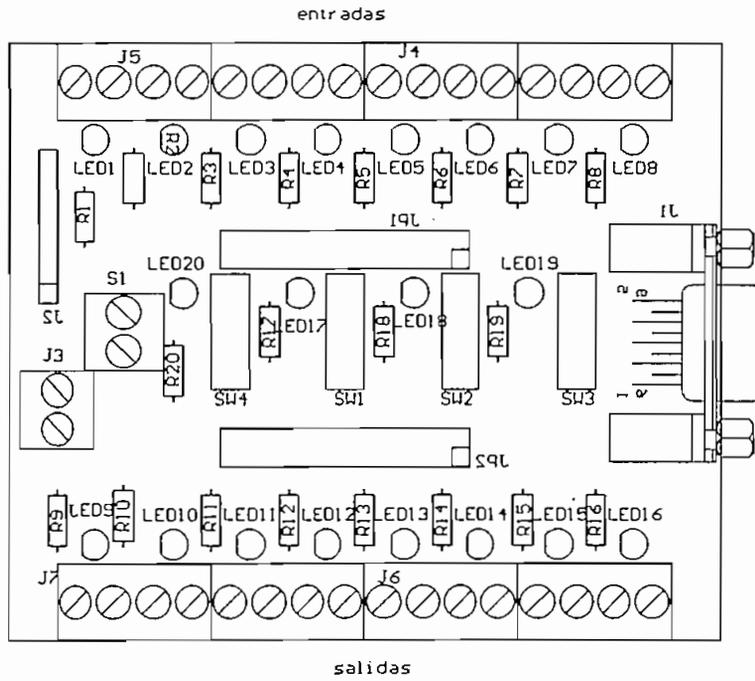
TARJETA DE CONTROL PRINCIPAL Ubicación de Elementos



TARJETA DE ENTRADAS SALIDAS Ubicación de Elements



TARJETA DE ELEMENTOS DE INDICACIÓN Ubicación de Elementos



TARJETA DE FUENTE Ubicación de Elementos

