

Manual de Contenido
del Participante

PLC Básico



TX-TIP-0005

ESPAÑOL

Propósito y Objetivos de este Manual

Este manual tiene como propósito entender qué es un PLC, cómo funciona y la razón que lo hace imprescindible en cualquier proceso que realice una máquina.

Los objetivos de este manual se orientan al cumplimiento de los siguientes puntos:



Entender cómo opera un PLC y para qué sirve en una industria.



Detectar fallas en el funcionamiento del controlador.



Manejar un circuito PLC.

Es importante comprender las consecuencias que el desconocimiento de los conceptos y principios explicados en este manual puede ocasionar en el ambiente, seguridad y salud ocupacional y en la calidad del producto final.

Cómo Utilizar este Manual

Este manual le muestra cuáles son las partes que componen un circuito de PLC, el método de programación y la forma en que lleva a cabo los controles necesarios.

En este manual usted va a encontrar gráficos demostrativos de un circuito de PLC, algunos de los códigos que utiliza, cómo manejar un PLC.



CAPÍTULO 1 5
Conceptos Básicos



CAPÍTULO 2 18
Controladores Lógicos Programables



CAPÍTULO 3 31
Configuración Básica del PLC



CAPÍTULO 4 42
Lógica en el PLC



CAPÍTULO 5 56
Codificación de Señales



CAPÍTULO 6 56
Programación de PLC



CAPÍTULO 7 84
Instalación, Puesta a Punto y Mantenimiento del PLC



CAPÍTULO 8 89
Interfases de Comunicación

El manual contiene pequeñas figuras que se repiten en todos los capítulos y que son una forma de organización de la información para hacer más fácil y dinámica la lectura. Estas figuras se denominan íconos.

A continuación hay una descripción de la utilización de cada icono, es decir en qué oportunidad aparecen:



GLOSARIO

Explica términos y siglas.



RECUERDE

Refuerza un concepto ya mencionado en el texto del manual.



ANEXO

Profundiza conceptos.



MANTENIMIENTO

Resalta procedimientos necesarios de mantenimiento.



PREGUNTAS

Presenta preguntas disparadoras.



ATENCIÓN

Destaca conceptos importantes.



EJEMPLO

Ilustra con situaciones reales los temas tratados.



ACTIVIDAD

Señala el comienzo de un ejercicio que le permitirá reforzar lo aprendido.



EXAMEN FINAL

Señala el comienzo de la evaluación final.



FIN DE CAPÍTULO

Señala la finalización del capítulo.



FIN DE MANUAL

Señala la finalización del manual.

Conceptos Básicos

TEMAS DEL CAPÍTULO 1

| | |
|------------------------------------|----|
| 1.1 Procesos Industriales | 6 |
| 1.2 Señales Discretas y Analógicas | 7 |
| 1.3 Automatismos | 8 |
| 1.4 Mando y Regulación | 10 |

Los procesos industriales están caracterizados por los sistemas de control que automatizan sus procesos.

Según cuales sean los medios de trabajo, habrá un criterio para la selección de los medios de mando.



1.1 Procesos Industriales

PROCESOS CONTINUOS

Estos procesos están **caracterizados por parámetros que cambian lentamente**. Una vez fijados los valores de entrada de referencia (set point) éstos podrán ser cambiados por el operador, pero los cambios no serán de gran magnitud. Los arranques y paradas son pocos y espaciados. El estado estacionario es la norma. Estos procesos **son controlados por sistemas de control analógicos**.

PROCESOS DISCRETOS

Estos procesos los encontramos en máquinas herramientas automatizadas - por ejemplo líneas de ensamble - en las cuales **se tiene una serie de acciones individuales sucesivas y, a veces, simultáneas**. Es decir, hay una cantidad de estados discretos que se suceden separados por distintos tiempos, siendo en algunos casos la separación de unos pocos milisegundos. Permanentemente se producen arranques y paradas de la línea. Estos procesos **fueron el tradicional dominio de utilización de los relés electromagnéticos y en la actualidad de los Controladores Lógicos Programables (PLC)**.

PROCESOS BATCH

Estos pueden ser **uno o más procesos continuos separados por procesos de tipo discreto, a menudo secuencial, además de la secuencia de arranque y la de parada**.

EJEMPLO

Como ejemplo se puede citar la carga de varios componentes para obtener una mezcla bajo ciertas condiciones de temperatura, presión, humedad, etc.



1.2 Señales discretas y Analógicas

¿Qué es una señal analógica?

Son aquellas señales registradas continuamente.

A diferencia de las señales discretas, las señales analógicas están presentes en diferentes niveles.

Por lo general son señales de tensión o de corriente de 0-10 V ó 0-20 mA ó 4-20 mA.

EJEMPLO

Por ejemplo, la temperatura es un caso típico de señal analógica ya que continuamente cambia en el tiempo en cantidades infinitesimales.

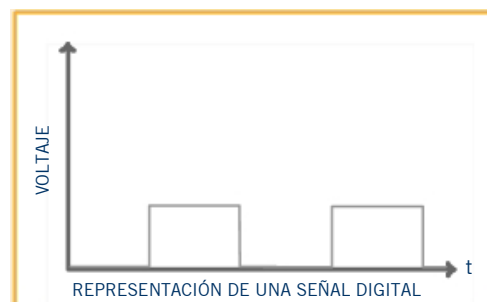
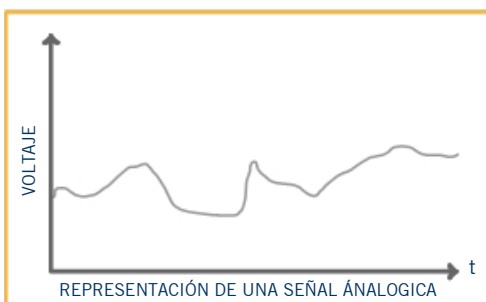
¿Qué es una señal analógica discreta?

Son aquellas señales registradas continuamente.

A diferencia de las señales discretas, las señales analógicas están presentes en diferentes niveles.

Por lo general son señales de tensión o de corriente de 0-10 V ó 0-20 mA ó 4-20 mA.

EJEMPLO



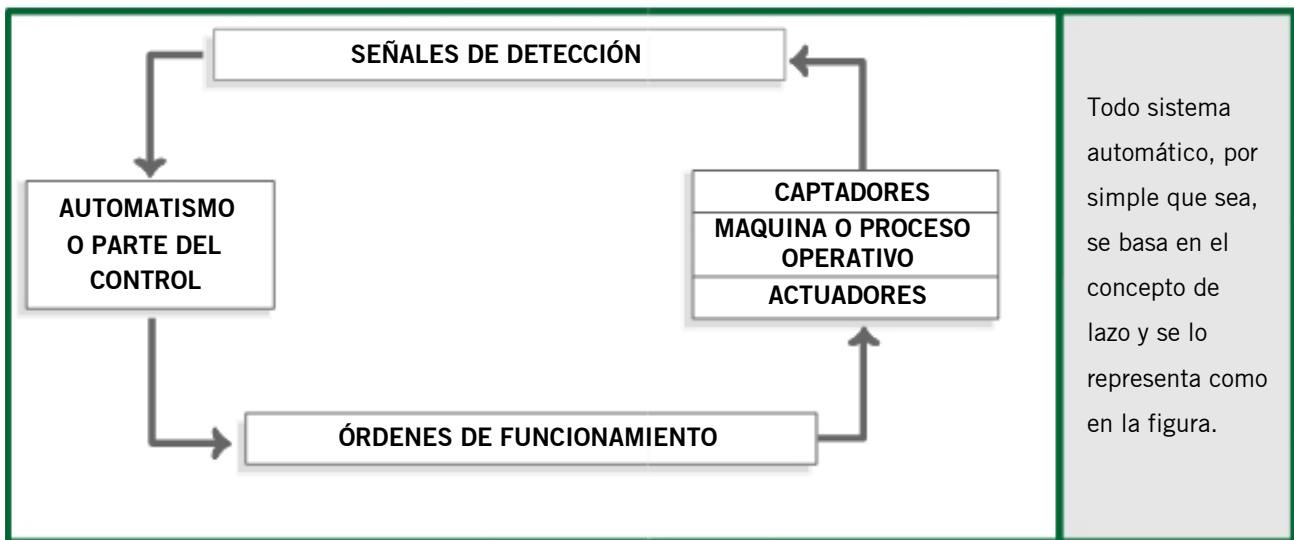
1.3 Automatismos

RECUERDE

La automatización de una máquina o proceso productivo simple tiene como consecuencia la liberación física y mental del hombre de dicha labor.



Se denomina automatismo al dispositivo físico que realiza ésta función controlando su funcionamiento.



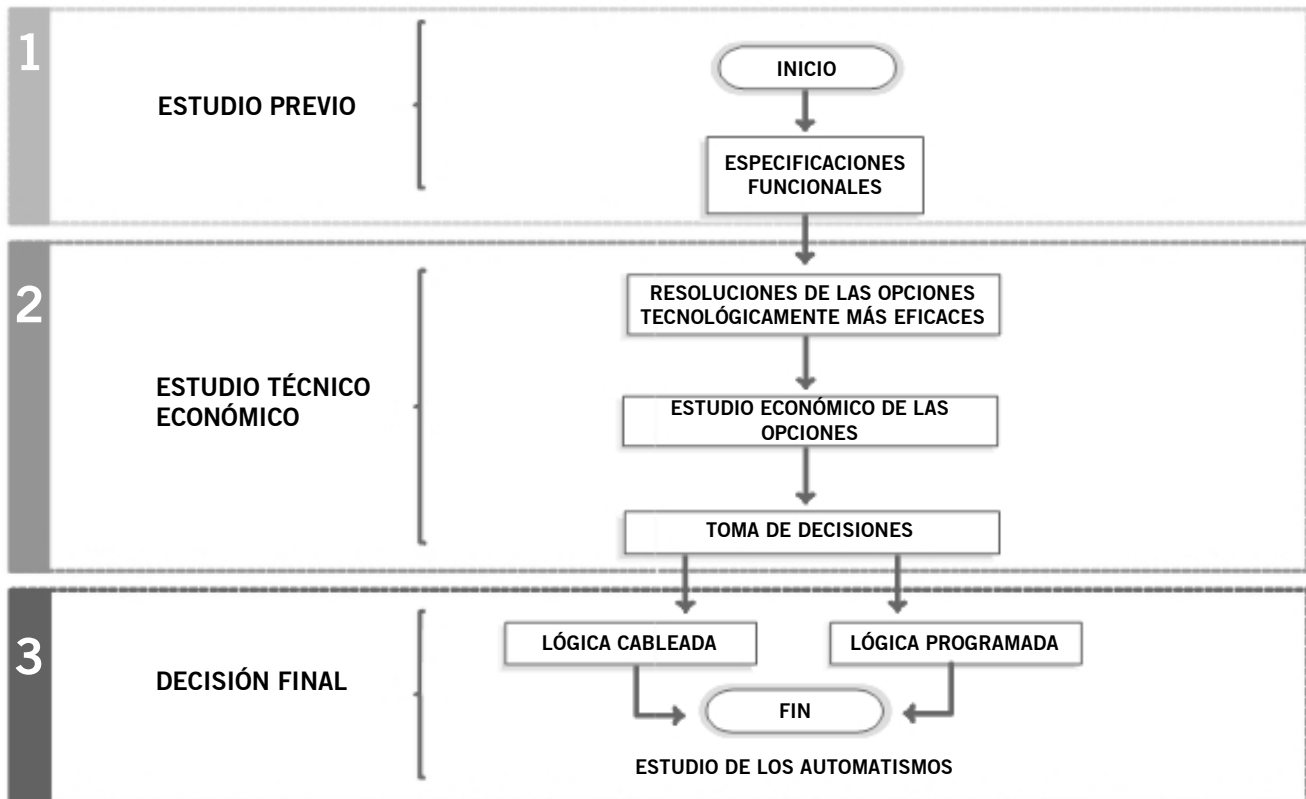
Fases de estudio en la elaboración de un automatismo

Para el desarrollo y elaboración correcta de un automatismo es necesario conocer previamente los datos siguientes:

- Las especificaciones técnicas del sistema o proceso a automatizar y su correcta interpretación.
- La parte económica asignada para no caer en el error de elaborar una buena opción desde el punto de vista técnico, pero no viable económicamente.
- Los materiales, aparatos, etc., existentes en el mercado que se van a utilizar para diseñar el automatismo.

En este apartado es importante conocer también:

- Calidad de la información técnica de los equipos.
- Disponibilidad y rapidez en cuanto a recambio y asistencia técnica.



1 ESTUDIO PREVIO

Es importante, antes de acometer cualquier estudio medianamente serio de un automatismo, conocer con el mayor detalle posible las características, el funcionamiento, las distintas funciones, etc., de la máquina o proceso a automatizar. Esto lo obtenemos de las especificaciones funcionales y es la base mínima a partir de la cual podremos iniciar el siguiente paso: estudiar cuales son los elementos más indicados para la construcción del automatismo.

2 ESTUDIO TÉCNICO ECONÓMICO

Es la parte técnica de especificaciones del automatismo: relación de materiales, aparatos, su adaptación al sistema y al entorno en el que se haya inscrito, etc. También aquí se ha de valorar la parte operativa del comportamiento del automatismo en todos sus aspectos como ser mantenimiento, fiabilidad, etc.

3 DECISIÓN FINAL

En la etapa anterior, estudio técnico-económico, se han debido estudiar las dos opciones tecnológicas posibles: lógica cableada y lógica programada.

Los parámetros más comunes que se deben valorar para una decisión correcta son los siguientes:

- Ventajas e inconvenientes que se le asignan a cada opción en relación a su fiabilidad, vida media y mantenimiento.
- Posibilidades de ampliación y de aprovechamiento de lo existente en cada caso.
- Posibilidades económicas y rentabilidad de la inversión realizada en cada opción.
- Ahorro desde el punto de vista de necesidades para su manejo y mantenimiento.

1.4 Mando y Regulación

Control por cableado y control por programa

Control por cableado

En esta técnica, mejor conocida como control convencional, **la unión física de diferentes elementos es la que determina la lógica o secuencia según la cual trabaja el control** elementos pueden ser botones pulsadores, relevadores, contadores, etc.

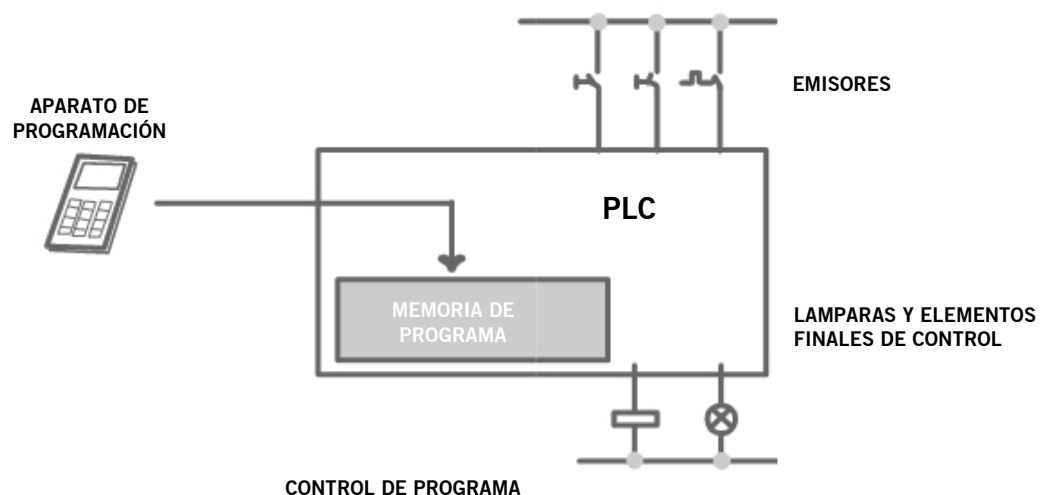
Si se quiere modificar la lógica de control en un sistema de control por cableado, se necesita hacer cambios en el cableado reorganizando los diversos elementos que participan en el circuito de control, esto es, descablear y recablear para obtener lo que se desea.

En un sistema sencillo esto puede parecer un problema sin importancia. Pero, cuando en el control están involucradas decenas o centenas de señales, hacer modificaciones al cableado resulta un problema relevante.

Control por Programa

Un problema así se resuelve mediante la técnica de control por programa donde correspondiente es independiente de la lógica o secuencia de control deseada.

En el control por programa los contactos de los emisores del proceso y los contactos de los elementos finales de control se conectan a las terminales de conexión o bornes de un dispositivo conocido como el PLC.



La lógica o secuencia según la cual trabaja el control se escribe en forma de programa en la memoria del PLC con la ayuda de un aparato de programación. Este programa se compone de una serie de instrucciones equivalentes a las conexiones serie y paralelo del control convencional.

La unidad de control del PLC lee las instrucciones almacenadas, interpreta su contenido y se encarga de su ejecución. Al hacerlo, el controlador consulta los estados de los emisores o entradas (botones pulsadores, finales de carrera, fotoceldas, etc.) y produce resultados a las salidas, tales como conexión o desconexión de bobinas, lámparas, etc.

En caso de querer hacer una variación a la secuencia de control, no se necesita modificar el cableado, sino el contenido de la memoria del controlador. La independencia del cableado con respecto a la secuencia de control es la diferencia y ventaja principal en el uso de PLC para las tareas de automatización.

Sin extendernos demasiado, es necesario tratar el importante avance de la regulación y el mando en la industria. Las designaciones concernientes a las técnicas de regulación y mando, se exponen como sigue.

MANDO

REGULACIÓN

- Dispositivo que sirve para gobernar grandes energías empleando otras menores.
- Conjunto de órganos que sirven para modificar, a menudo automáticamente, la potencia de una máquina o su funcionamiento.
- Control, sin intervención humana, de la alimentación de una máquina ó materia prima.
- Conjunto de fenómenos en un ciclo, que no pueden ser desarrollados por el hombre en forma secuencial y con seguridad.

Según la norma DIN 19226, mandar o controlar, es el fenómeno engendrado en el interior de un sistema, durante el cual uno o varios parámetros considerados de entrada, actúan sobre otros considerados de salida, según leyes propias del sistema. Este fenómeno origina una acción a través del órgano de transferencia, como tal o a través de la cadena de mando.

En la figura se representan las entradas y salidas en un bloque de un sistema de control:

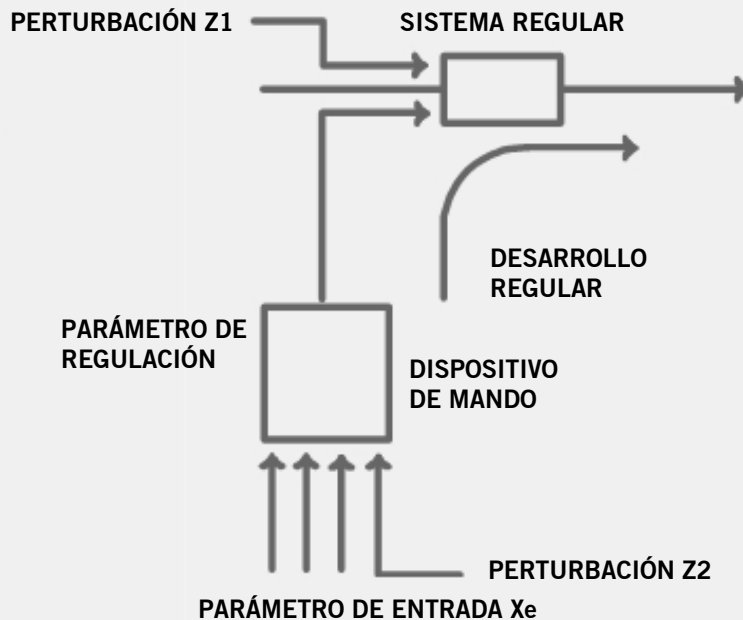


**REPRESENTACIÓN DE UN SISTEMA CON
ENTRADAS Y SALIDAS**

1. Al sistema lo simplificamos con un bloque.
2. Los parámetros de entrada (identificados con x), introducen las señales de información; son tratadas por el sistema, y luego restituidas bajo la forma de parámetros de salida.

La norma se extiende aún más sobre el término de mando.

Mando se utiliza muy a menudo, no solamente para designar el acto de controlar, sino también para denominar el dispositivo global en el interior del cual se desarrolla esta acción. Según esta definición, el conjunto representado en la figura, puede también denominarse bajo el término general de mando.

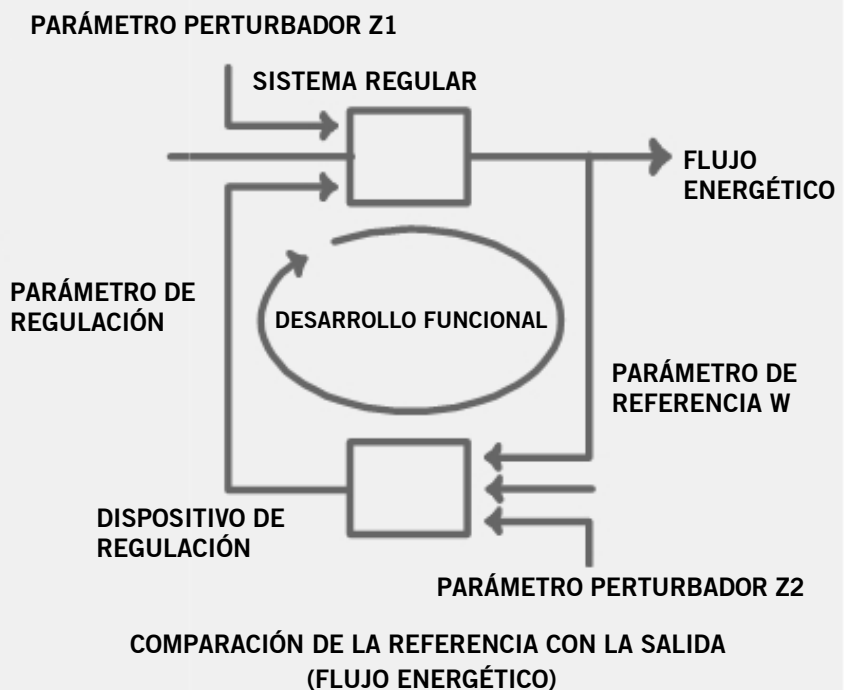


MANDO

REGULACIÓN

Es el proceso mediante el cual, el parámetro de salida se toma constantemente en consideración y se lo compara con otro valor de referencia (entrada), antes de ser adaptado en función del resultado a otro valor del parámetro de entrada. El desarrollo funcional que resulta entonces es un circuito cerrado.

La regulación tiene por finalidad adaptar el valor del parámetro a regular, a pesar de influencias parásitas o perturbaciones, al valor predeterminado como parámetro de referencia.



En el caso de la regulación, los parámetros disponibles a la salida del dispositivo, intervienen igualmente en el flujo energético, pero en este caso el **parámetro de entrada del sistema está influenciado por la comparación con el de salida.**

Formas de energía para elementos de trabajo y de mando

Para la técnica de mando, la posibilidad de poder convertir señales de una forma de energía en otra, significa que dentro de un automatismo puede operarse con diferentes tipos de energías. Existe pues la posibilidad de dimensionar un mando según criterios económicos y técnicos, que resulten óptimos.

La siguiente recopilación ha de proporcionar una idea general de los medios más corrientes de trabajo y de mando, y de sus criterios de elección. Sin embargo, no se trata de una enumeración completa de todos los factores, sino sólo de una exposición de los puntos más importantes:

Medios de trabajo

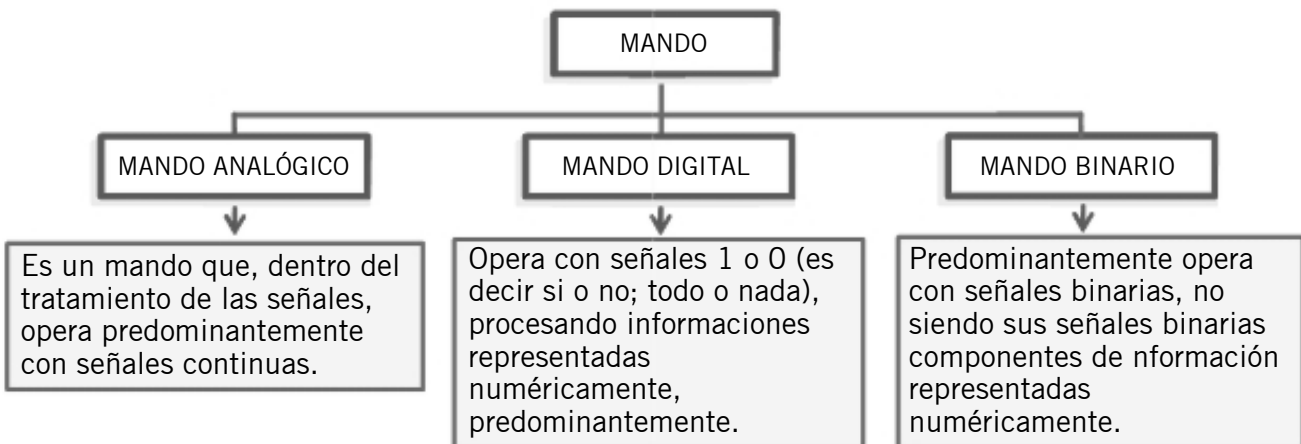
- Electricidad: Corriente eléctrica.
- Hidráulica: Líquidos.
- Neumática: Gases.

Criterios para la elección del sistema

- Fuerza.
- Recorrido.
- Tipo de movimiento (lineal, rotativo, etc.).
- Velocidad.
- Tamaño constructivo.
- Sensibilidad.
- Seguridad.
- Regulación.
- Gasto de energía.

Distinción de Mandos

- Analógico.
- Digital.
- Binario.



Nota: El mando binario transforma señales binarias de entrada, en señales binarias de salida, principalmente con órganos combinatorios, temporizadores y órganos de memoria.

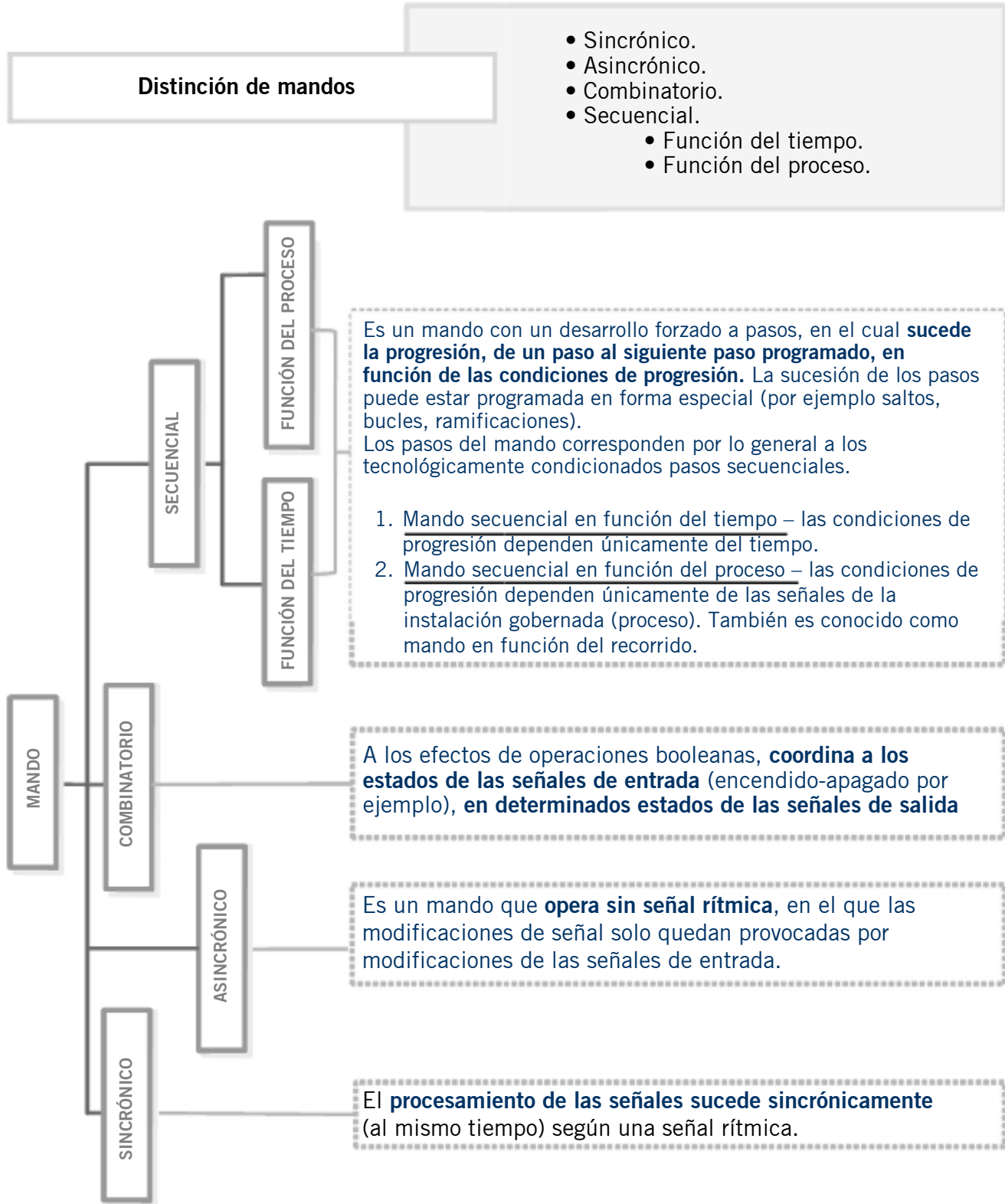
EJEMPLO

Los contadores, registros, memorias, totalizadores son las informaciones a procesar, normalmente están representadas por un código binario.



Distinción de los mandos según la forma de representación de la información

Para este caso, se trata de la forma de cómo quedan las señales de un mando, combinadas, influidas y finalmente procesadas. En la figura se presenta los tipos de mandos según la forma de representación de la información.



Descomposición de una cadena de mando

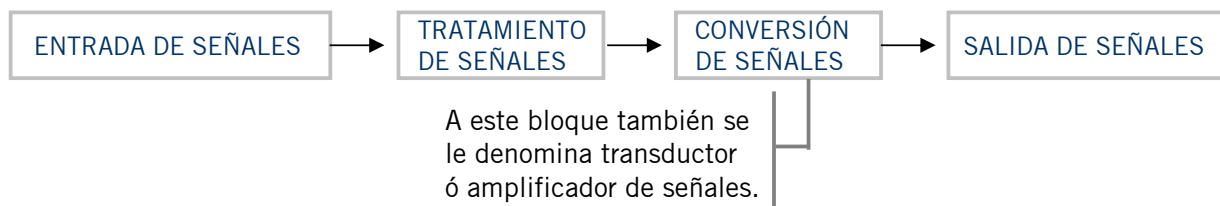
RECUERDE



Un dispositivo de mando, o bien un mando en general, está representado en muchos casos como una negra, cerrada, con entradas y salidas que representan el flujo de señales. Se trabaja así en las distintas áreas de la tecnología: hidráulica, neumática, electrónica, eléctrica

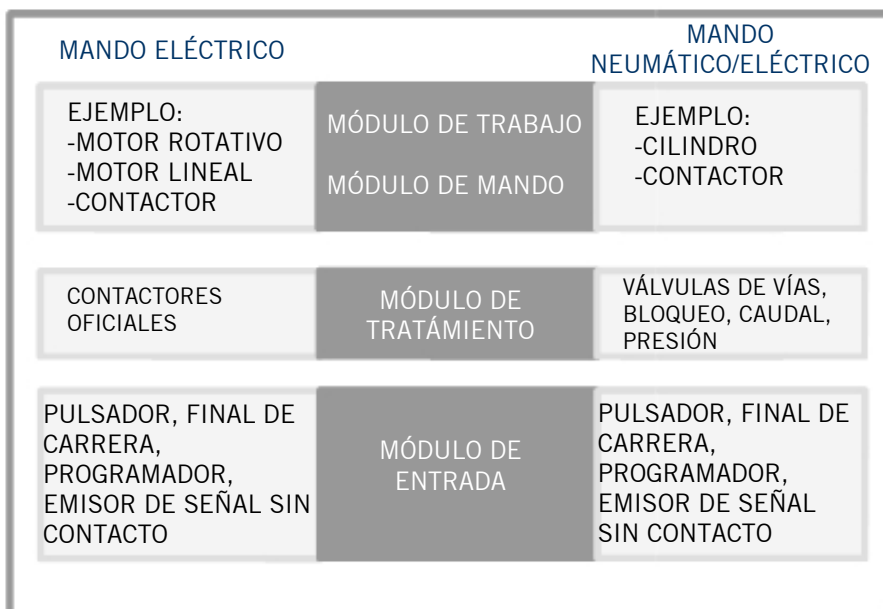


Si se combinan distintas técnicas, por ejemplo electricidad/hidráulica, o bien electrónica/hidráulica, será necesario intercalar otro bloque. En la figura se representa este bloque justo antes de la salida de señales.



La conversión de señales, sea cual sea su denominación, tiene como función convertir las señales recibidas del campo -entradas y tratamiento de señales- (de la otra técnica), para el campo de salidas de señales.

Para el diseño y proyecto de un sistema de mando, a menudo interesa únicamente una parte de la instalación total. **Existe pues, la posibilidad de extraer una parte del sistema de mando general y considerarla como sistema de mando parcial e independiente.**



En la figura se desglosan, a título de ejemplo, los elementos básicos a través de los cuales tiene que circular una señal neumática ó eléctrica, en sus sistemas de mando respectivo.

Distinción de los mandos según la forma de representación de la información

¿Cómo actúa el operador y los sensores de procesos? El procedimiento general es:

- 1) **El operador envía órdenes al sistema** de control mediante botoneras, llaves, etc. y **recibe información del mismo** por medio de luces de señalización, instrumentos indicadores, alarmas, etc.
- 2) **El proceso recibe órdenes del controlador** a través de actuadores, contactores, válvulas solenoides, actuadores de válvulas, entre otros elementos, que sirven para activar motores, válvulas, cilindros neumáticos o hidráulicos, etc.. **Además envía al controlador señales** recogidas por medio de sensores del estado de diversas variables del mismo: velocidad, corriente, presión, temperatura, proximidad, etc.
- 3) **El controlador enlaza todas estas señales y actuará en una forma predeterminada** (por ejemplo de acuerdo a la forma en que ha sido cableado si es un circuito de control con relés, contactores, etc.).

IMPORTANTE

En el caso que estamos estudiando, el controlador será un Controlador Lógico Programable o PLC, que es un dispositivo electrónico que usa memoria programable para almacenar instrucciones destinadas a implementar funciones específicas tales como lógicas secuenciales, temporizadores, contadores, funciones aritméticas, etc., que controlará máquinas y procesos. Las operaciones a realizar por el controlador están almacenadas en su memoria en forma de una sucesión de instrucciones.



ACTIVIDAD 1. Conceptos Básicos

Usted ha visto los conceptos generales y básicos para poder comprender luego cómo es un PLC.



Al finalizar la actividad grupal, complete los casilleros vacíos con el número correspondientes y escriba las palabras faltantes arriba de las líneas.

1

¿Qué entiende por automatización de una máquina? ¿Cree que es imprescindible?

2

¿Por qué cree que es importante conocer cuáles son los procesos que lleva a cabo la máquina a automatizar?

3

¿Cuál es la ventaja de realizar un control por programa?

4

Para realizar un control por programa se necesitan ciertos sensores, ¿Cuáles son algunos de los sensores que se pueden utilizar?

¡Felicitaciones!

Usted ha finalizado el capítulo 1.

A continuación se desarrollará el capítulo Controladores Lógicos Programables.



Lógicos Programables

TEMAS DEL CAPÍTULO 2

| | |
|-----------------------------------|----|
| 2.1 Principios de Funcionamiento | 19 |
| 2.2 Usos del PLC | 21 |
| 2.3 PLC y Dispositivos de Control | 23 |
| 2.4 Ventajas del PLC | 26 |

Cuando se habla de PLC, se habla de un controlador lógico programable, el cuál se encarga de ordenar las acciones de la máquina, teniendo en cuenta que las distintas variables que rigen el proceso se encuentren dentro de su rango de valores aceptables.

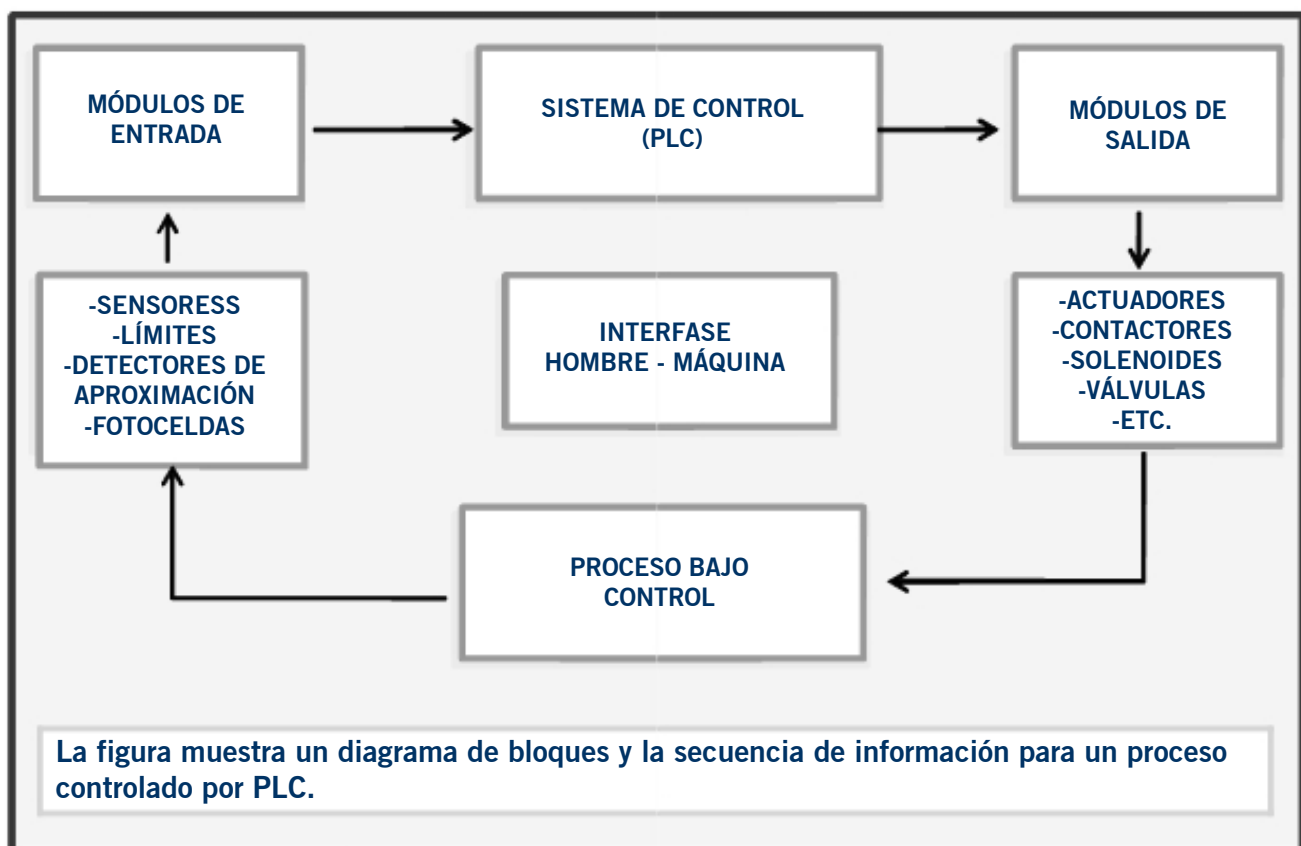


2.1 Principios de Funcionamiento

¿Qué es un PLC?

PLC (Programmable Logic Controller), significa controlador lógico programable. Su nombre mismo lo define: es un dispositivo **utilizado para controlar** y esto se realiza **en base a una lógica determinada**. Esta lógica **se define a través de un programa**.

Es esta última característica la que lo diferencia del resto de los dispositivos, que son capaces de controlar a través de la ejecución de una lógica.

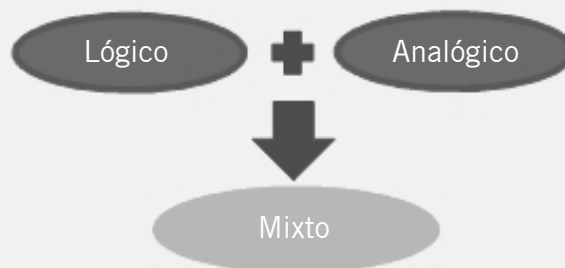


Un sistema controlado con PLC se puede dividir en ocho partes principales o componentes básicos. De acuerdo al diagrama de la figura anterior éstas son:

- **Proceso bajo control:** Proceso industrial a controlar.
- **Entrada/Sensores:** Dispositivos de retroalimentación al PLC. Sensores de temperatura, presión, proximidad, humedad, vibración, flujo, limit switches.
- **Módulos de entrada:** Interfase entre los componentes de entrada y el controlador. Su función es acondicionar la señal de entrada a niveles que el controlador pueda procesar.
- **Sistema de control o PLC:** Formado por módulos de comunicación, la unidad central de proceso (CPU), memoria y fuente de alimentación.
- **Programa de aplicación:** Es la lógica que representa el comportamiento del proceso y se almacena en la memoria del PLC.
- **Módulos de salida:** Interfase entre el controlador y los componentes de salida. Su función es acondicionar la señal de salida a los niveles para los cuales está diseñado el actuador.
- **Salida/Actuadores:** Elementos finales de control. Solenoides, motores, lámparas, válvulas proporcionales, etc.
- **Interfase hombre máquina:** Encargado de programar y/o monitorear el controlador y el proceso, por ejemplo, computadoras industriales, programadores industriales, computadoras personales, lap-top's, etc.

Este sistema de control puede ser:

- Lógico.
- Analógico.
- Mixto.



IMPORTANTE

Existen pequeños PLC's que son sólo lógicos. Debido al gran avance tecnológico y la rápida expansión en el uso de los Controladores Lógicos Programables, existen muchos fabricantes y todos estos equipos tienen características generales semejantes. Es por esto que se torna casi imposible profundizar los detalles propios de cada uno, tanto a nivel de hardware, como a nivel de software.

2.2 Uso del PLC

¿Qué puede hacer un PLC?

- ✓ Comando de secuencias a partir de la influencia de cambios técnicos, procesos y/o secuencias.
- ✓ Comando en la secuencia correcta de pasos y en el tiempo justo.
- ✓ Supervisión de sistemas a partir de la medición de parámetros.
- ✓ Entrega de avisos que permitan la corrección de sistemas, cuando se producen desviaciones.

EJEMPLO

Los PLC's pueden enlazarse, por ejemplo, a una máquina de control numérico, para manejar secuencias de orden inferior que no pertenecen al comando propio del sistema de control numérico.



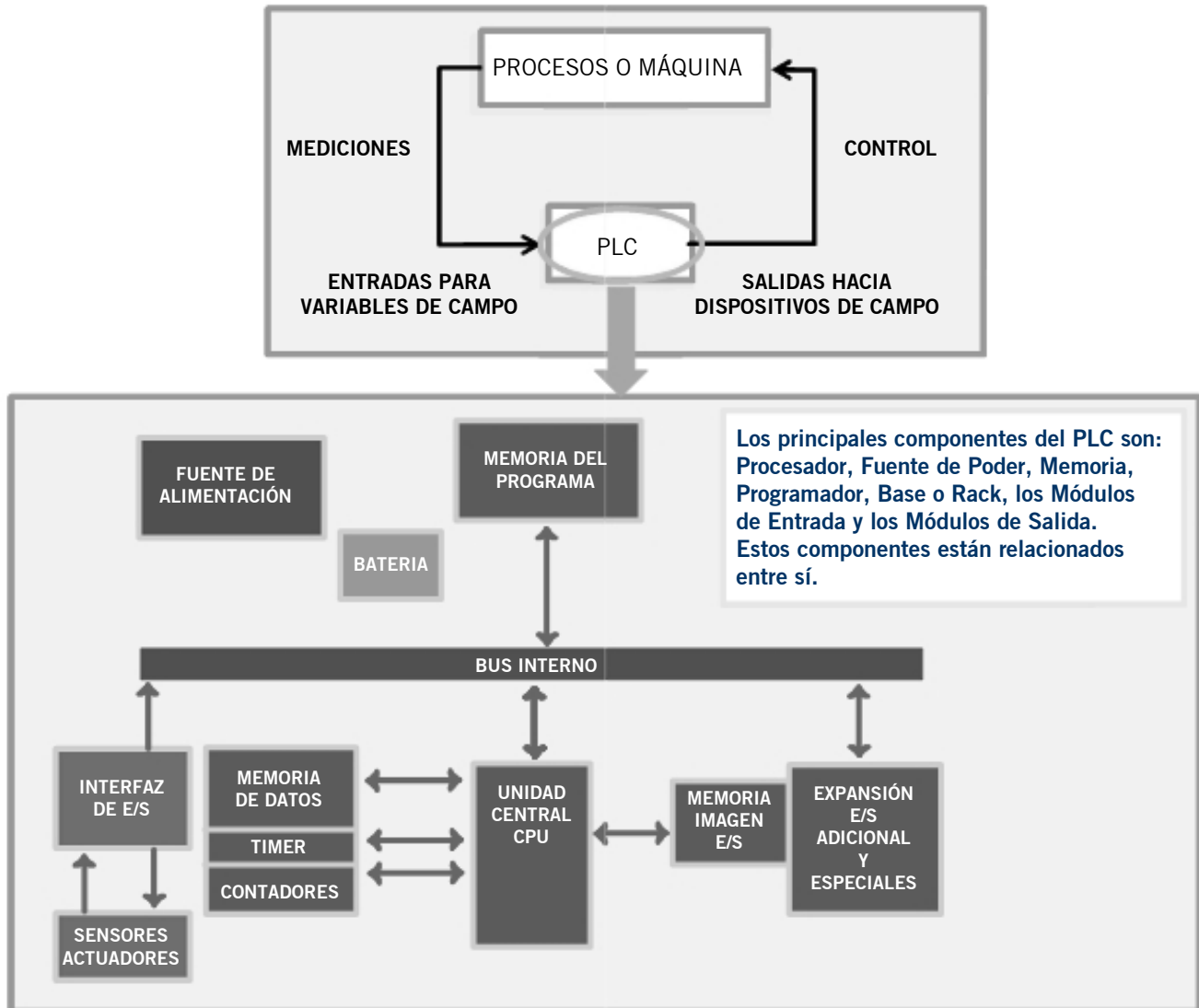
Un **Controlador Lógico** es una computadora industrial diseñada específicamente para recibir señales de púlpitos de operación y dispositivos de campo y, por medio de un programa almacenado, generar señales de salida para controlar máquinas o procesos.

Esta computadora cuenta con características especiales en cuanto en el diseño de su unidad central de procesamiento (CPU) y de sus módulos de entrada y salida (módulos I/O). **Los PLCs son utilizados para el control de procesos automáticos de producción**

EJEMPLO

Si una tarea del PLC es controlar el desarrollo secuencial de un circuito electro-hidráulico, no sólo encontramos los componentes propios de ese circuito (electro-válvulas, cilindros, etc.), sino también habrá un procesador de señales.

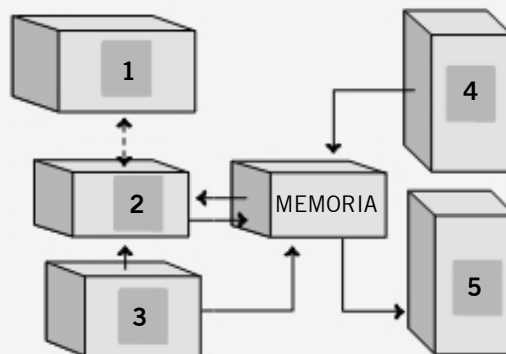
Forma de operación del PLC



ACTIVIDAD 2



Nombre cada uno de los bloques componentes de un PLC, según el esquema de la figura.



1.

2.

3.

4.

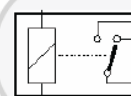
5.

2.3 PLC y Dispositivos de Control

¿Es posible reemplazar un PLC?

Si, las funciones que realizan los PLCs podrían efectuarse con otros dispositivos de control, tales como relevadores, computadoras, etc.

PLC y Relevadores



La mayoría de los procesos industriales requieren de que se cumplan ciertas condiciones de operación y seguridad para poder funcionar. Esto se da desde niveles de operación que requieran mucho apoyo por parte de un operador, hasta el nivel donde se tiene un sistema completamente automático. La automatización de procesos requiere de la ejecución de cierta lógica de permisos y de una secuencia que puede ser controlada usando relevadores y temporizadores (relevadores de tiempo), hoy se prefiere utilizar PLCs.

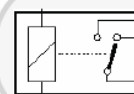
Inicialmente, los PLCs se diseñaron para sustituir relés en control secuencial. Actualmente sigue siendo su aplicación principal, pero tienen capacidad para realizar otras funciones como: control de variables (temperatura, velocidad, etc.) almacenamiento de datos y comunicación. Al utilizar relevadores para controlar la secuencia de un proceso, la lógica de operación se define mediante las conexiones entre contactos y/o bobinas (alambrado), esto hace más difícil las modificaciones o encontrar una falla (falsos contactos).

En los PLCs la lógica se define en un programa, lo que permite que se pueda modificar fácilmente. Además, por medio del sistema de diagnóstico se indican las fallas que ocurren en el sistema.

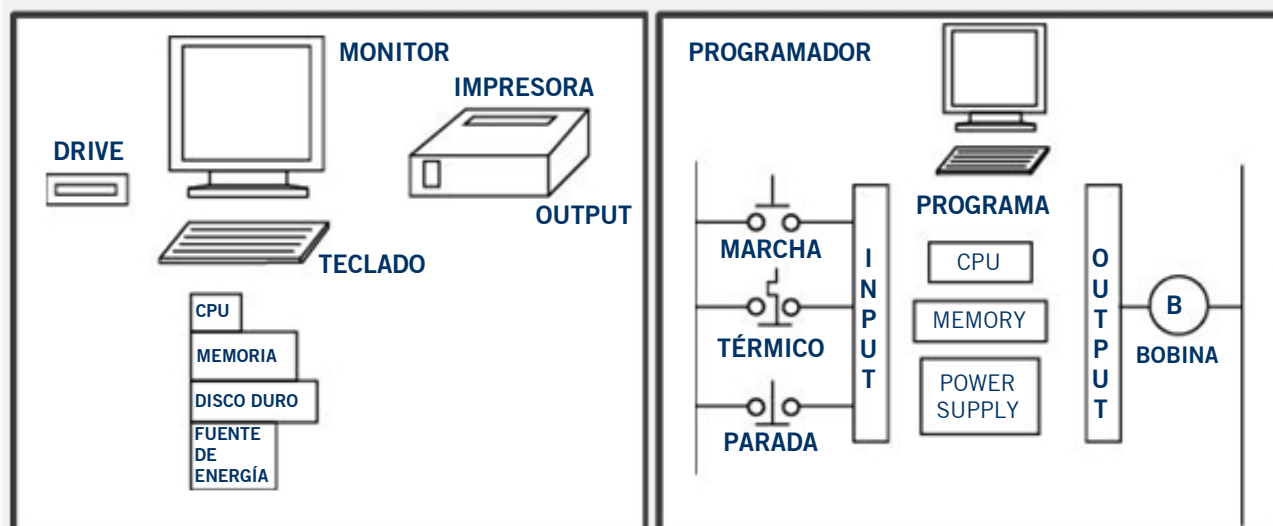
En lo que respecta a la capacidad de activar directamente a una carga, en ciertos casos es una limitación que presentan los PLCs, pues su capacidad de corriente generalmente es de 2A como máximo. Debido a esto, si se requiere activar cargas que demanden corrientes mayores es necesario conectar la salida del PLC a un relevador.



PLC y Computadoras



La arquitectura del PLC es básicamente la misma que la de la computadora, como se muestra en la figura. Sin embargo, algunas características importantes distinguen y hacen que el PLC presente en ciertos casos ventajas sobre la computadora:



El rango de temperatura de operación de los PLCs se encuentra entre los 0° C y los 55° C (32°F y 131°F), además, puede estar expuesto a una humedad relativa de hasta un 95%. Los rangos de operación los establece cada fabricante.

Los PLCs cuentan con una programación que permite fácilmente diseñar funciones lógicas, operaciones, comparaciones, etc. Así, el personal familiarizado con el diseño y la detección de fallas mediante el uso de diagramas de relevación, se adapta rápidamente a esa nueva tecnología.

Los PLCs cuentan con módulos donde se conectan los dispositivos de campo, por medio de los cuales recibe información del estado del proceso.

Cuando se dañan estos módulos, pueden ser reemplazados fácilmente, tal como se muestra en la figura. Así, los circuitos de interfases (entradas / salidas) que son módulos y con capacidad de autodiagnóstico, permiten la fácil detección de fallas y su rápida sustitución.

IMPORTANTE

A diferencia de las computadoras, el PLC está diseñado específicamente para trabajar en condiciones de ambiente industrial. Se puede usar en áreas donde exista ruido eléctrico, vibración, temperaturas altas y humedades relativas también altas, pero sin condensación.



Fácil Sustitución de Módulos

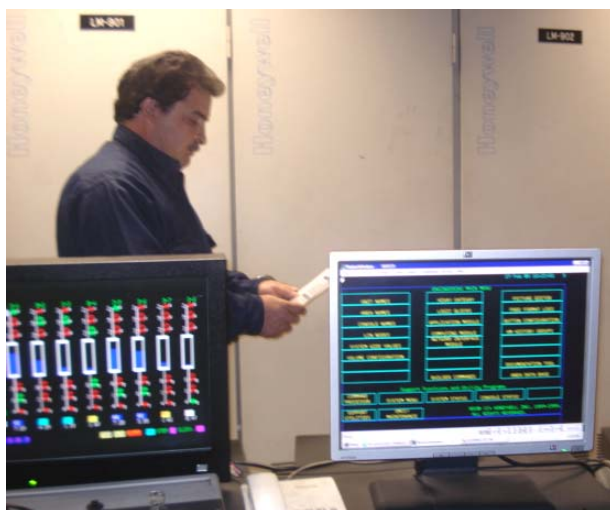
Otra diferencia entre los PLCs y las computadoras es que estas últimas están diseñadas para realizar funciones complejas de cálculo, manipulación de grandes volúmenes de información y ejecución de varias tareas o programas al mismo tiempo o en tiempo compartido.



El PLC, en cambio, tiene una limitada capacidad de cálculo y de manipulación de información, además ejecuta un programa ordenadamente en forma secuencial.

En modelos avanzados de PLCs hay instrucciones que permiten llamar a subrutinas, interrupción de tareas y saltos en el programa, entre otras instrucciones, lo cual permite mayor flexibilidad en la ejecución del programa.

Por otra parte, las computadoras se utilizan como una herramienta auxiliar del PLC por su gran capacidad, facilidad para el manejo y análisis de datos. También para programación y monitoreo, como se muestra en la figura.



ACTIVIDAD 3



Indique cuáles de las siguientes afirmaciones son falsas y cuáles verdaderas.

| | | VERDADERO | FALSO |
|---|---|--------------------------|--------------------------|
| 1 | En la Industria no es posible reemplazar un PLC ya que no existen dispositivos que puedan reemplazar su función | <input type="checkbox"/> | <input type="checkbox"/> |
| 2 | La actividad principal de un PLC es el control de proceso secuencial. | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | Los PLC tienen capacidad de corriente de hasta 2A. | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | El PLC no soporta vibraciones ni humedades relativas altas. | <input type="checkbox"/> | <input type="checkbox"/> |
| 5 | La capacidad de cálculo del PLC es limitada comparada con una PC. | <input type="checkbox"/> | <input type="checkbox"/> |

2.4 Ventajas de los PLC

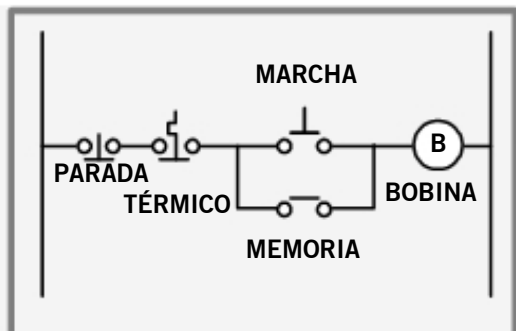
La diferencia fundamental entre un PLC y la lógica cableada, radica en el hecho de que **la lógica cableada del tablero, o los circuitos impresos de un sistema electrónico, son reemplazados en el PLC por un programa.**

Esto, además de una gran flexibilidad, confiere estas características:

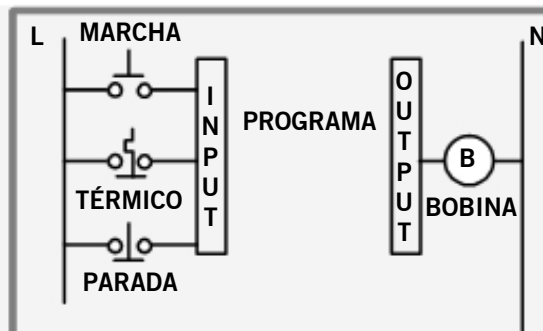
- **Menor tiempo de trabajo** en las conexiones a realizar aumenta la puesta en marcha y ajuste del sistema.
- **Facilidad de realizar cambios** durante la operación del sistema (pudiéndose cambiar la lógica completa si fuese necesario).
- **Independencia con respecto al cableado**, ya que la lógica o secuencia de control no depende de la conexión de elementos (hardware).
- **Reducción de espacio**. Los diversos elementos (hardware) que intervienen en la lógica por cableado son sustituidos por software dentro del controlador, los cuales no requieren gabinetes o tableros como el control convencional.
- **Facilidad en la prueba y puesta en marcha**. La lógica de control se prueba por secciones o en su totalidad con la ayuda del dispositivo programador y ahí mismo se hacen las modificaciones necesarias.
- **Rápida detección de fallas y averías**. Existe un software que facilita la detección de fallas, tanto del programa de control como del controlador.
- **Independencia de voltajes**. Los voltajes de operación de los emisores pueden ser distintos a los voltajes de operación de los elementos finales de control, ya que no están unidos mediante conexiones como ocurre en control convencional.

Algunas de las **ventajas de los PLCs** en comparación con sistemas automáticos y secuenciales hechos con relevadores son:

- **La arquitectura de los PLCs es modular**. Esto permite una gran flexibilidad debido a que el sistema se puede expandir o modificar simplemente agregando o cambiando módulos.
- El hecho que **la lógica o secuencias en el PLC sean programables**. Así se ajusta a nuevos requisitos de la aplicación o se puede usar en una aplicación diferente.
- **En un PLC no existe ninguna conexión física entre las entradas y las salidas**, la conexión se realiza a través del programa.

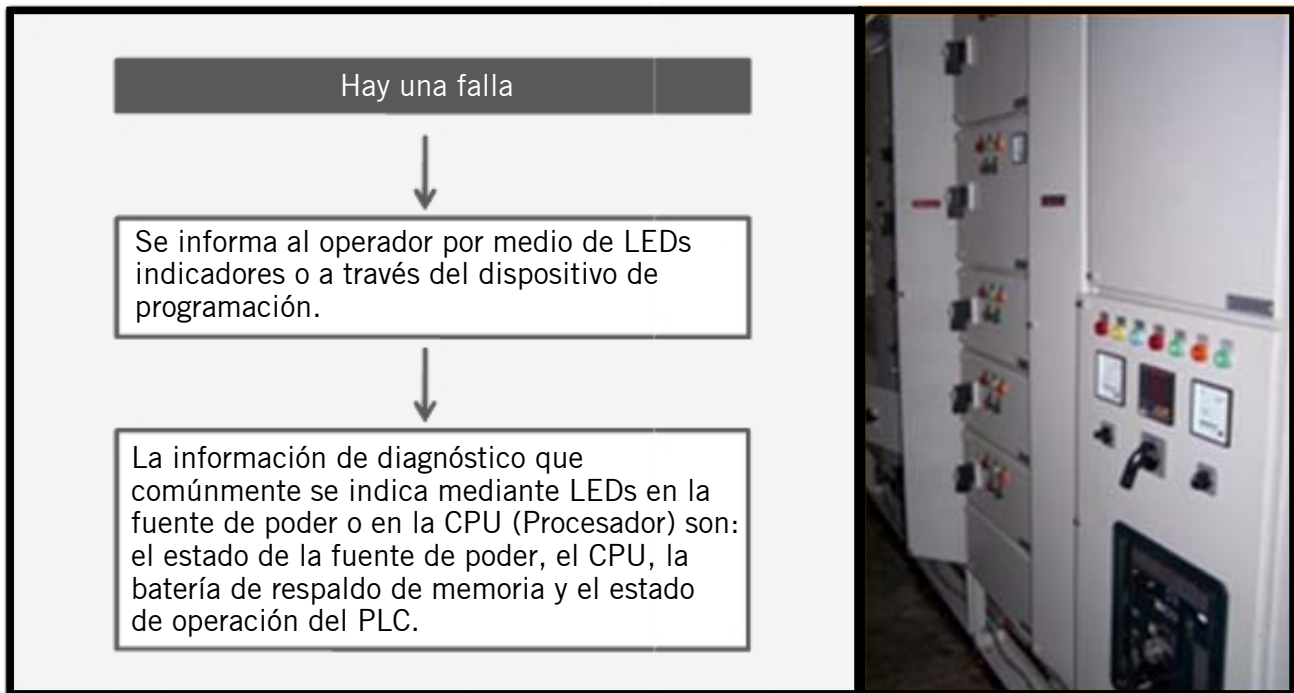


Convencional



Con PLC

Los PLCs tienen sistemas de diagnóstico muy avanzados, esto permite detectar una gran cantidad de fallas en la CPU, en los módulos de entrada/salida y, algunas veces, hasta en los circuitos de conexión a los dispositivos de campo.



MANTENIMIENTO

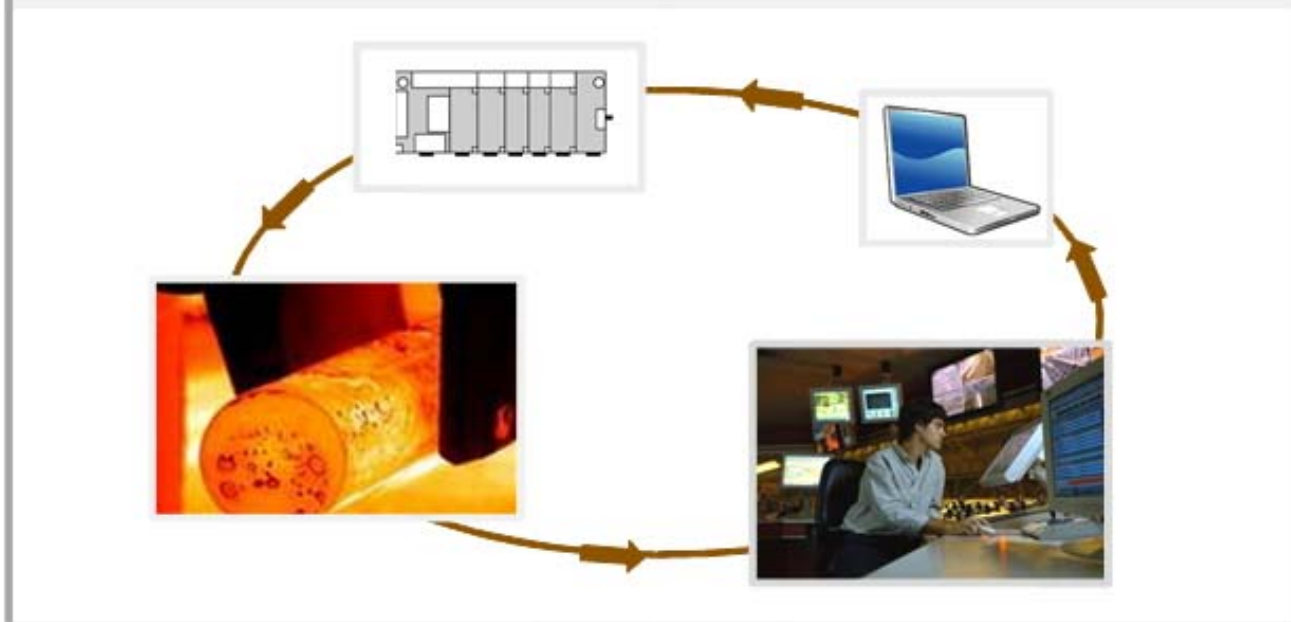


Como todos los componentes del PLC son electrónicos (estado sólido, sin movimiento ni contactos), las fallas se reducen a un mínimo y el mantenimiento en condiciones de falla se limita, generalmente, al reemplazo de módulos.

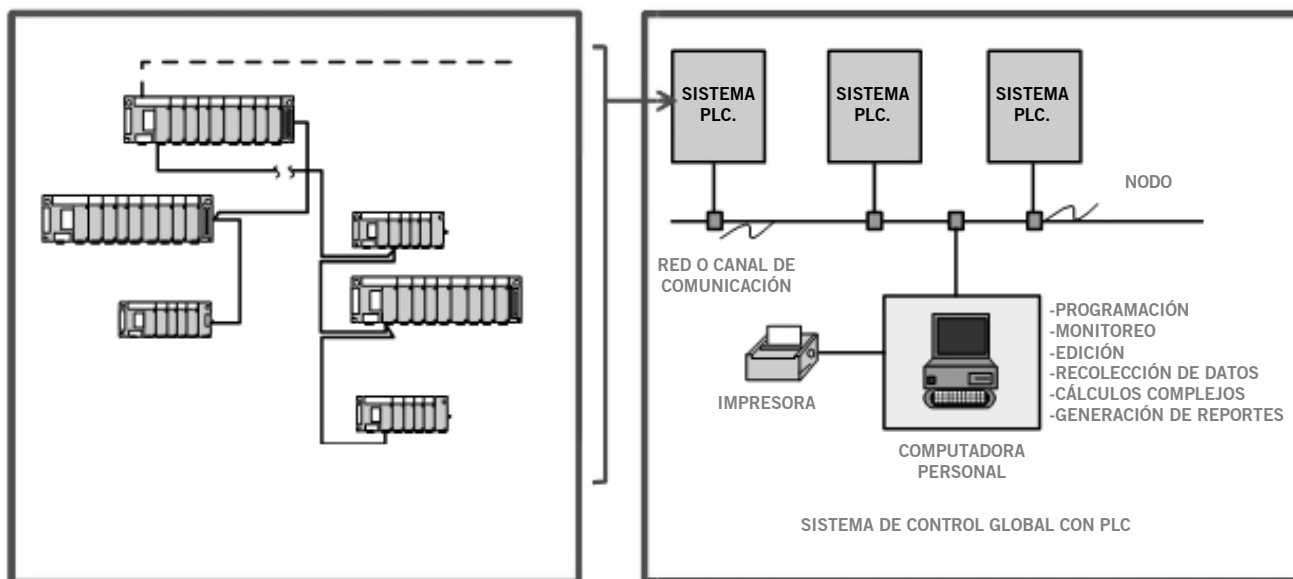
- El PLC puede realizar otras funciones, por ejemplo: operaciones aritméticas, funciones de comparación, operaciones con bits, manejo de datos y control regulatorio. También tiene **capacidad de almacenamiento de datos**.
- El PLC **se puede comunicar con otros dispositivos programables** como ser otros PLCs, computadoras, estaciones de operación y otros sistemas de control. Por ejemplo, puede recibir información de una computadora que monitorea su operación y modificar datos, tales como referencias (set points) y valores preestablecidos en retardos (timers) o contadores.

La posibilidad de comunicación entre el PLC y los módulos remotos de entrada/salida, permiten que estos últimos se puedan instalar cerca de los equipos y procesos. Las entradas/salidas remotas se conectan al CPU (Procesador) por medio de un cable coaxial o pares de cables torcidos, reduciendo considerablemente los costos de cableado.

Esquema de un sistema con PLC que cuenta con interfases o módulos de entradas / salidas tanto locales como remotas.



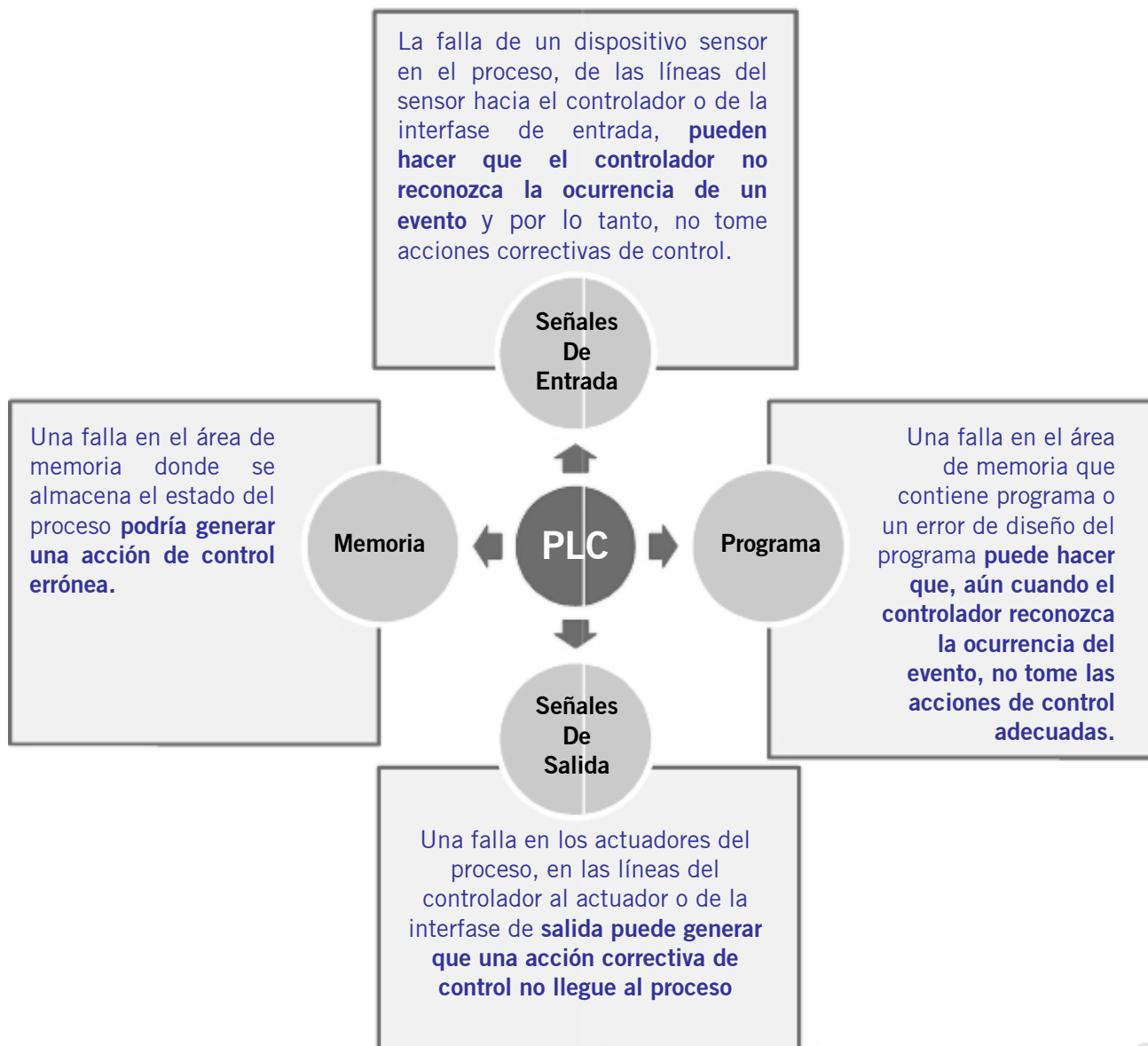
La capacidad de comunicación del PLC le permite integrarse a esquemas de comunicación y control en una red local o incluso de toda la planta (global). En la figura se muestra un sistema de control global donde forman parte varios PLCs.



En la actualidad la mayoría de los equipos de monitoreo y de control cuentan con capacidad de integrarse dentro de una red, esto permite tener acceso a información del proceso y así poder tomar acciones más rápidas y oportunas. **Los tiempos de instalación y costos de cableado se reducen significativamente.**

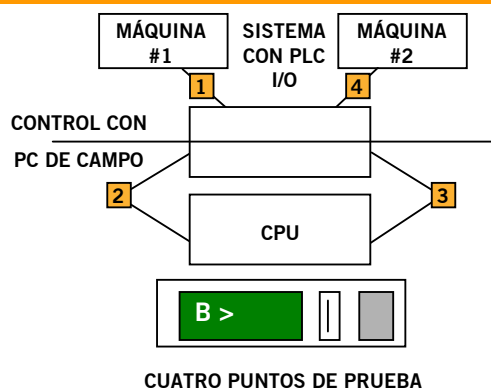
Problemas entre el PLC y el proceso

Un problema entre la interacción del PLC y el proceso se puede encontrar en **4 posibles áreas**:



IMPORTANTE

Ante la presencia de un problema en la interacción entre el PLC y el proceso, el primer paso a seguir es determinar en cuál de estas 4 áreas se encuentra el problema. En la figura se muestran los puntos de prueba, marcados con los números del 1 al 4, para determinar el área del problema.



ACTIVIDAD 4. Controladores Lógicos Programables

Usted ha visto los conceptos generales y básicos para poder comprender luego cómo es un PLC.



Al finalizar la actividad grupal, complete los casilleros vacíos con el número correspondientes y escriba las palabras faltantes arriba de las líneas.

- | | | |
|----------|---|--------------------------|
| 1 | Si tuviera que clasificar la dificultad de modificación de un circuito PLC, usted diría que es... | |
| | Fácil. | <input type="checkbox"/> |
| | Difícil. | <input type="checkbox"/> |
| | No se puede modificar. | <input type="checkbox"/> |
| <hr/> | | |
| 2 | La conexión entre entradas y salidas se realiza mediante... | |
| | Cables. | <input type="checkbox"/> |
| | Software. | <input type="checkbox"/> |
| | Microprocesadores. | <input type="checkbox"/> |
| <hr/> | | |
| 3 | Al instalar un circuito de PLC, el número de fallas se producen... | |
| | Aumenten pero es más fácil la solución. | <input type="checkbox"/> |
| | Es igual, pero es más fácil el mantenimiento. | <input type="checkbox"/> |
| | Disminuyen y es más fácil el mantenimiento. | <input type="checkbox"/> |
| <hr/> | | |
| 4 | Una falla en la señal de entrada puede ocasionar: | |
| | Que el controlador no detecte ciertas fallas en el proceso. | <input type="checkbox"/> |
| | Que la máquina no arranque. | <input type="checkbox"/> |
| | Que el PLC no funcione. | <input type="checkbox"/> |

¡Felicitaciones!

Usted ha finalizado el capítulo 2.

A continuación se desarrollará el capítulo configuración Básica del PLC.



Configuración Básica del PLC

TEMAS DEL CAPÍTULO 3

3.1 Módulos Funcionales

19

3.2 Partes de un PLC

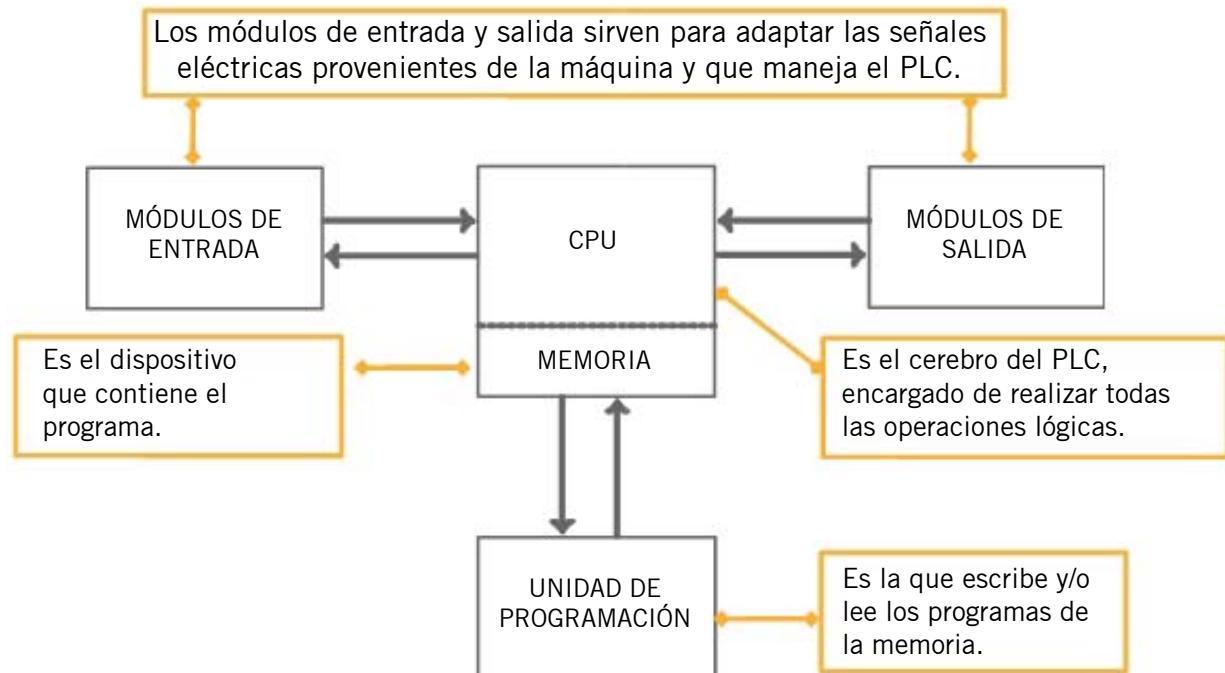
21

El conocimiento de las partes que componen un PLC permitirá una rápida detección de fallas de este mismo



3.1 Módulos Funcionales

El diagrama de la figura representa la configuración básica del Controlador Lógico Programable (PLC).



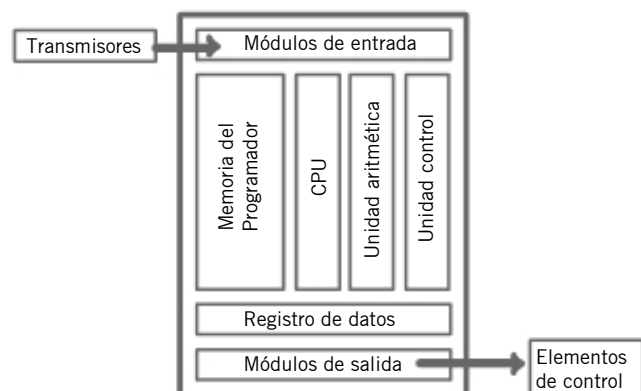
Partes de un PLC

Las instrucciones de una determinada secuencia se introducen en la memoria, mientras que los transmisores, elementos de control y consumidores se conectan directamente al PLC.

La unidad principal del PLC se compone del microprocesador (CPU), de la memoria de programa y de los módulos periféricos (de entrada y salida). **El programa de comando se escribe en la memoria con la ayuda de un módulo de programación.** Este se puede retirar del PLC una vez que se haya finalizado la programación.

El programa es leído y evaluado línea por línea por la unidad de control, en forma seriada. Este proceso seriado se realiza con mucha rapidez, de manera que apenas podría verse en la pantalla de una PC.

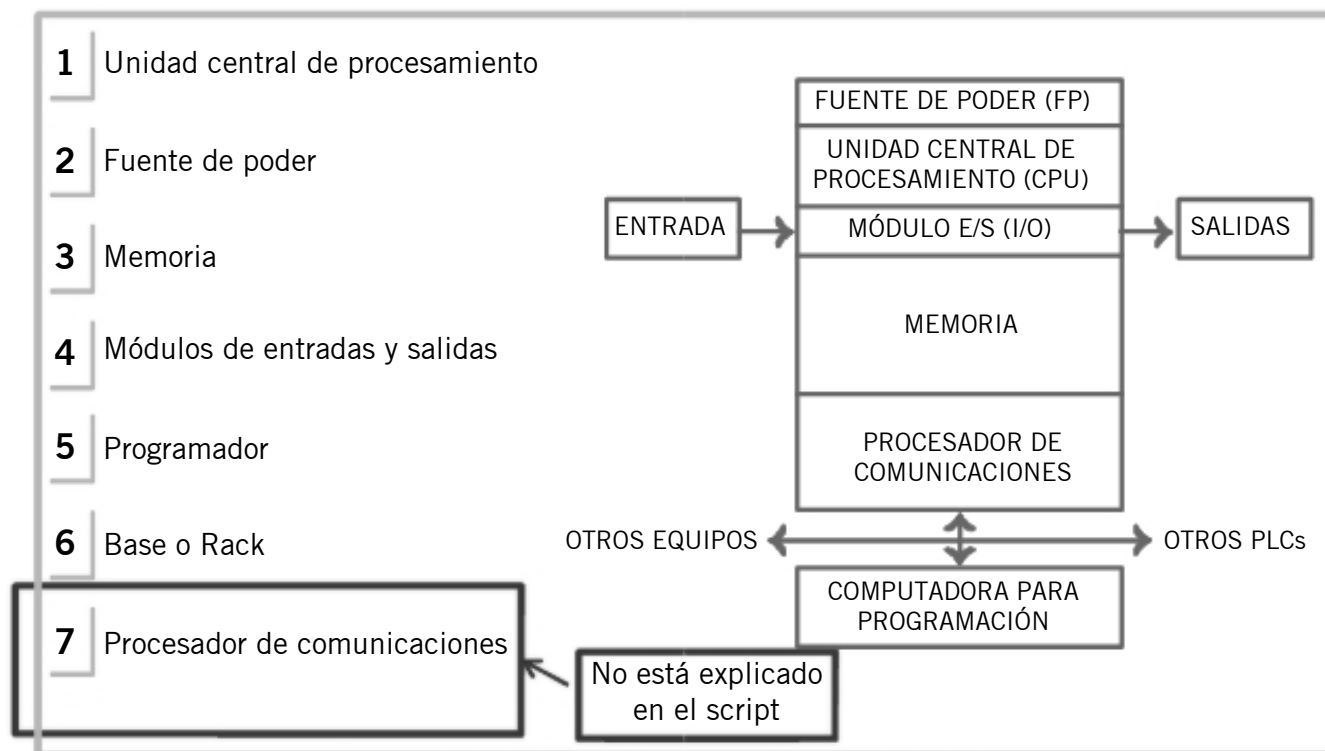
Al procesamiento de todas las instrucciones de la memoria, se le denomina ciclo. Al tiempo que tarda, se lo denomina tiempo del ciclo.



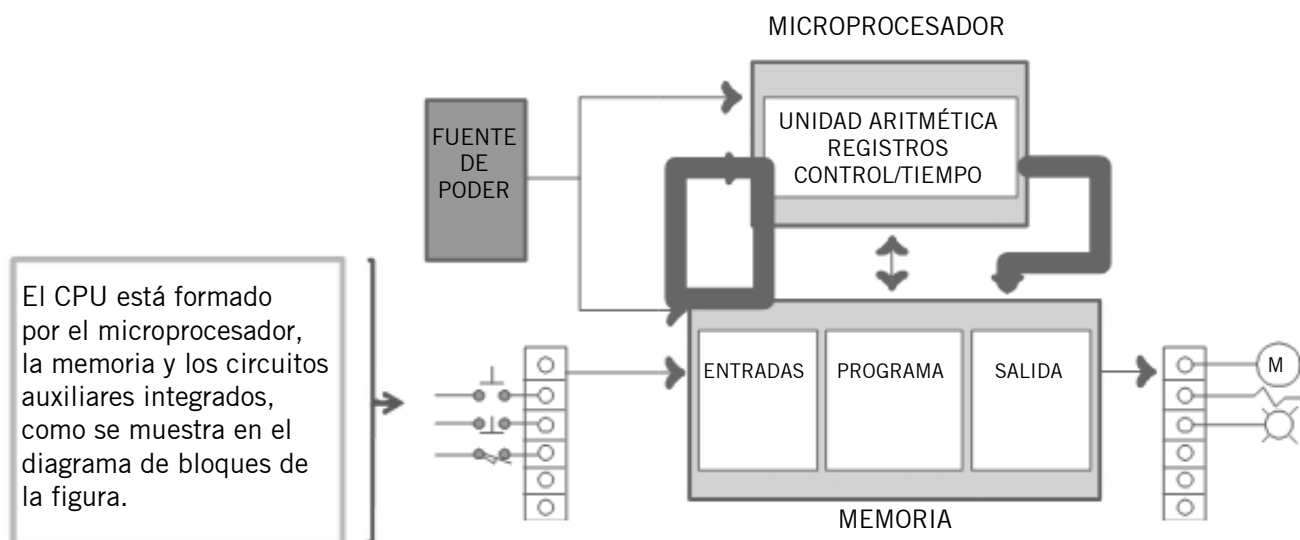
El tiempo de ciclo de un PLC, o sea su velocidad de trabajo, es aproximadamente 7 milisegundos (7 ms) para una capacidad de memoria de alrededor 1000 instrucciones.

3.2 Partes de un PLC

Existen una gran variedad de PLC, por lo tanto, sus componentes, así como sus capacidades, varían mucho, sin embargo, se puede considerar que un PLC se encuentra constituido de las siguientes

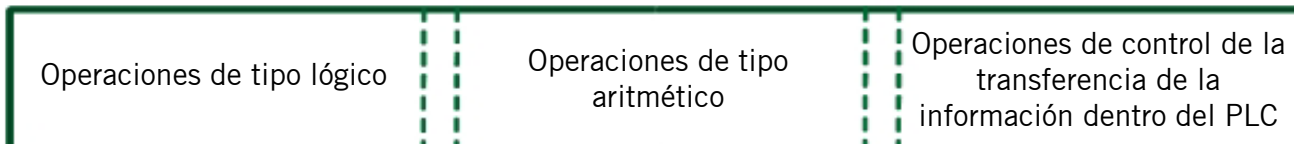


1 Unidad Central de Procesamiento



1.1 Microprocesador.

Es un circuito integrado (chip) a gran escala de integración (LSI) que realiza muchas operaciones que se pueden agrupar en:



Los circuitos internos de un microprocesador son:

- **Circuitos de la unidad aritmética y lógica (ALU):** Es la parte del microprocesador donde se realizan los cálculos y las decisiones lógicas para controlar el PLC.
- **Circuitos de la unidad de control (UC):** Organiza todas las tareas del microprocesador. Por ejemplo, cuando una instrucción del programa codificada en código binario (ceros y unos) llega al microprocesador, una pequeña memoria ROM instruye a la UC cuales es la secuencia de señales que tiene que emitir para que se ejecute la instrucción.
- **Registros:** Los registros son memorias en las que se almacenan temporalmente datos, instrucciones o direcciones mientras necesitan ser utilizados por el microprocesador. Los registros más importantes de un microprocesador son los de instrucciones, datos, direcciones, acumulador, contador de programa, de trabajo y el de bandera o de estado.
- **Buses:** No son circuitos en sí, sino zonas conductoras en paralelo que transmiten datos, direcciones, instrucciones, y señales de control entre las diferentes partes del microprocesador. Se puede hacer una diferencia entre buses internos y externos. Los primeros unen entre sí las diversas partes del microprocesador, mientras que los segundos son pistas de circuito impreso que unen chips independientes. Los buses internos y externos son continuación unos de los otros.

1.2 Funciones del CPU

En la memoria ROM del sistema, el fabricante ha grabado una serie de programas ejecutables fijos, **Firmware** o software del sistema. A estos programas accede el microprocesador para ejecutar las funciones que correspondan.

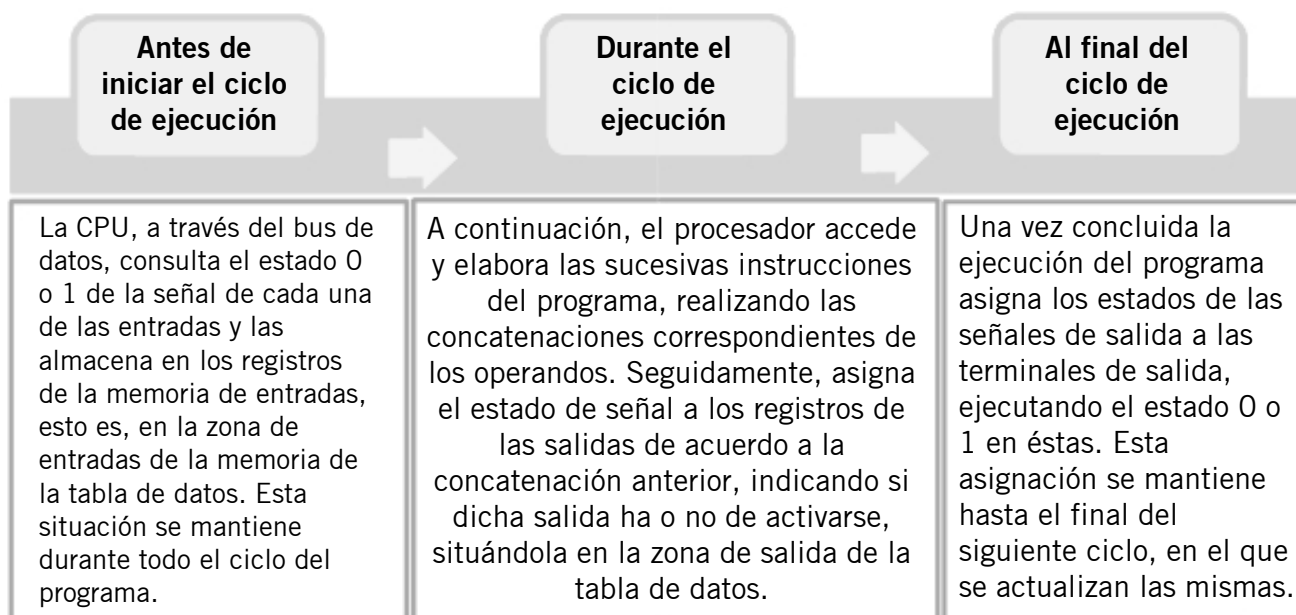
El software de sistema de cualquier PLC consta de una serie de funciones básicas que se llevan a cabo en determinados momentos de cada ciclo: en el inicio o conexión, durante el ciclo o ejecución del programa y en la desconexión.

Este software es ligeramente variable en cada PLC, pero en general contiene las siguientes funciones:

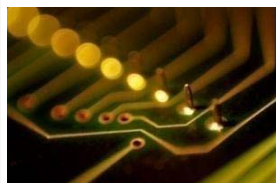
- Supervisión y control del tiempo de ciclo (watchdog), tabla de datos, alimentación, batería, etc.
- Autotest en la conexión y durante la ejecución del programa.
- Inicio del ciclo de exploración de programa y de la configuración del conjunto.
- Generación del ciclo base de tiempo.
- Comunicación con periféricos y unidad de programación.

1.3 Ciclo Básico de Trabajo del CPU

El ciclo básico de trabajo del programa en la CPU es el siguiente:



IMPORTANTE



Como ya habíamos dicho, el cerebro o la inteligencia de los controladores programables está basada en los procesadores o microprocesadores, formados por circuitos integrados con gran capacidad de cálculo y de control.

En el PLC, el procesador realiza todas las operaciones lógicas y matemáticas, manejo de datos y rutinas de diagnóstico. El procesador gobierna las actividades de todo el sistema, interpretando y ejecutando una colección de programas llamado programa ejecutable. Este programa se encuentra guardado permanentemente en la memoria de la CPU. Generalmente se lo considera como parte del mismo CPU y no puede ser accedido ni modificado por el usuario.

La CPU del PLC puede tener más de un procesador si fuera necesario aumentar la velocidad de ejecución de las operaciones.

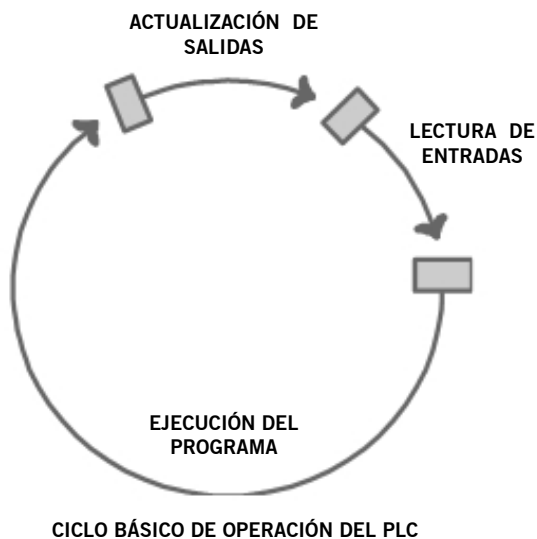
El hecho de dividir las operaciones del PLC entre dos o más procesadores se denomina **multiprocesamiento**.

La función básica del procesador es leer las entradas de todos los dispositivos de campo y ejecutar el programa de control de acuerdo a la lógica programada. En base a los resultados de introducir el estado de las entradas y ejecutar el programa, energizará o desenergizará los dispositivos de salida.

EJEMPLO

Un ejemplo de multiprocesamiento es cuando se tiene un procesador para realizar las funciones lógicas y otro para las funciones matemáticas y de manejo de datos.

Debido a la gran rapidez con que operan los PLCs, pareciera que todas estas funciones se ejecutasen al mismo tiempo. En realidad, **las instrucciones se ejecutan en secuencia** (una después de otra) y al terminar, el ciclo se repite (mientras el PLC esté en operación). Este ciclo se muestra en la figura.



La secuencia de funcionamiento de un PLC se puede describir de la siguiente forma:

- 1 Se leen las entradas de los dispositivos de campo y con esto se actualiza una tabla de estados de las entradas (memoria).
- 2 Se ejecuta o resuelve la lógica de acuerdo al programa y en base a sus resultados se actualiza una tabla de estados de las salidas.
- 3 Se actualiza el estado de las salidas, lo cual mantiene o modifica el estado de indicadores o actuadores del proceso en campo.

1.4 Scanning del PLC

Se llama scanning al tiempo que le lleva al PLC leer y actualizar los estados de las entradas, leer la memoria del usuario, resolver una a una las instrucciones del programa, hasta guardar los valores de salida en un registro.

2 Fuente de Poder

La fuente de poder suministra el o los voltajes de polarización regulados (corriente directa) a los circuitos electrónicos que forman el procesador, la memoria y los módulos de entrada/salida del PLC.

La fuente de poder normalmente se alimenta con 120 VCA o 220 VCA y permite variaciones del suministro de hasta un 20% (varían de acuerdo a las especificaciones del PLC utilizado). Cuando el voltaje de alimentación excede ciertos límites preestablecidos, las fuentes de poder envían un comando de paro al procesador.

La fuente de poder tiene protección de límite de corriente, esto es, mantiene a su salida voltajes de corriente regulados en tanto la corriente que suministra no rebasa cierto nivel preestablecido (por ejemplo 6 A en la fuente de 5 V). Si los requerimientos de corriente son mayores, el voltaje de salida cae y protege en esta forma la máxima capacidad de potencia de la fuente. Al disminuir un cierto nivel se activa un comando de paro al procesador.

Las fuentes de poder se diseñan para que puedan proporcionar la corriente que necesita el sistema local completo. En caso de utilizar muchos módulos, se tiene la opción de instalar otra u otras fuentes de poder y de esta forma poder cumplir con los requerimientos de carga del sistema.

3 Memoria

La memoria es el **área del CPU donde se guardan todas las instrucciones (programas) y datos para la operación del PLC**. La memoria total del sistema se puede dividir en las siguientes áreas:

- **Memoria ejecutiva.** Contiene los programas que dirigen la operación del PLC y se consideran parte integral del mismo PLC.
- **Memoria de aplicación.** Es el área de la memoria donde se guardan las instrucciones (programa) y los datos que utiliza el procesador para realizar sus funciones de control.
- **Memoria del usuario.** Almacena el programa de control, registros para almacenar valores temporales de conteo, preset, tablas de calores, etc. Registros temporales internos para almacenar estados de entradas / salidas, estados de relés, conteo de relés temporales, etc.
- **Memoria que almacena el sistema operativo.** En esta memoria está el sistema operativo del programa que gobierna el funcionamiento del procesador y que no es accesible.

La memoria **almacena información en forma de dígitos binarios** (bits) de modo que cada dígito puede estar en estado de sí o no. Los bits se agrupan en palabras (words).

4 Módulos de entradas y salidas

4.1 Módulos de entrada

Son los dispositivos mediante los cuales **se hace llegar lo que está sucediendo en el proceso a la CPU**. Dependiendo de esto, la CPU resuelve la secuencia del programa y actualiza el estado de los módulos de salida.

4.2 Módulos de salida

Son el medio que permite a la CPU, en base a las entradas y estados de registros en el programa, mantener o modificar el estado de los diferentes dispositivos del proceso. Pueden ser módulos de salida, por ejemplo lámparas indicadoras, arrancadores de motores, solenoides, bombas, válvulas, etc.

Los módulos de entrada y salida se pueden catalogar en:

Módulos discretos o lógicos

Son aquellos que manipulan información digital, esto es cerrado abierto, on-off, 1 ó 0. Entre estos están, por ejemplo, señales de interruptores límite, termostatos, contactos de arrancadores o relevadores, botones, selectores, llaves límites, borneras, contactos de relés de algún panel de control. Pueden tener dos estados: abierto-cerrado ó tensión-no tensión. En la figura, se muestra un ejemplo de módulo discreto, en este caso una llave límite, la cual funciona cuando un objeto pasa sobre la palanca y provoca que se cierre un interruptor, funcionando como sensor de presencia



FIN DE CARRERA

Módulos analógicos

Son aquellos que reciben o envían señales que pueden adoptar un gran número de niveles diferentes. Entre estos están rangos de 0 a 5 V, 0 a 10 V, 0 a 15 V, 4 a 20 mA, 0 a 30 mA, 0 a 1 mA, etc. Son módulos que transforman las señales analógicas que ingresan al PLC provenientes del proceso, en dígitos binarios para su procesamiento en el PLC.

4.3 Módulos de entrada/salida especiales

Algunos PLC pueden tener, además de los módulos ya descritos de entrada/salida, otros módulos destinados a procesar ciertos tipos de entradas como trenes rápidos de pulsos o módulos inteligentes que realizan funciones de control de tipo analógico. En este caso la función de control se realiza en el mismo módulo y la señal resultante es entregada al PLC aliviando a la CPU de realizar éstas tareas con lo que el tiempo de cada ciclo será menor haciendo además más sencillo el programa del PLC.

Los módulos de E/S especiales disponibles varían según el fabricante del PLC, siendo algunos de

Tarjetas de entrada de trenes de pulsos

Los pulsos son procesados mediante contadores, comparadores, divisores, compuertas programables, etc. Se usan, por ejemplo, para procesar trenes de pulsos enviados por generadores de pulsos (encoders) utilizados para medir la velocidad de motores.

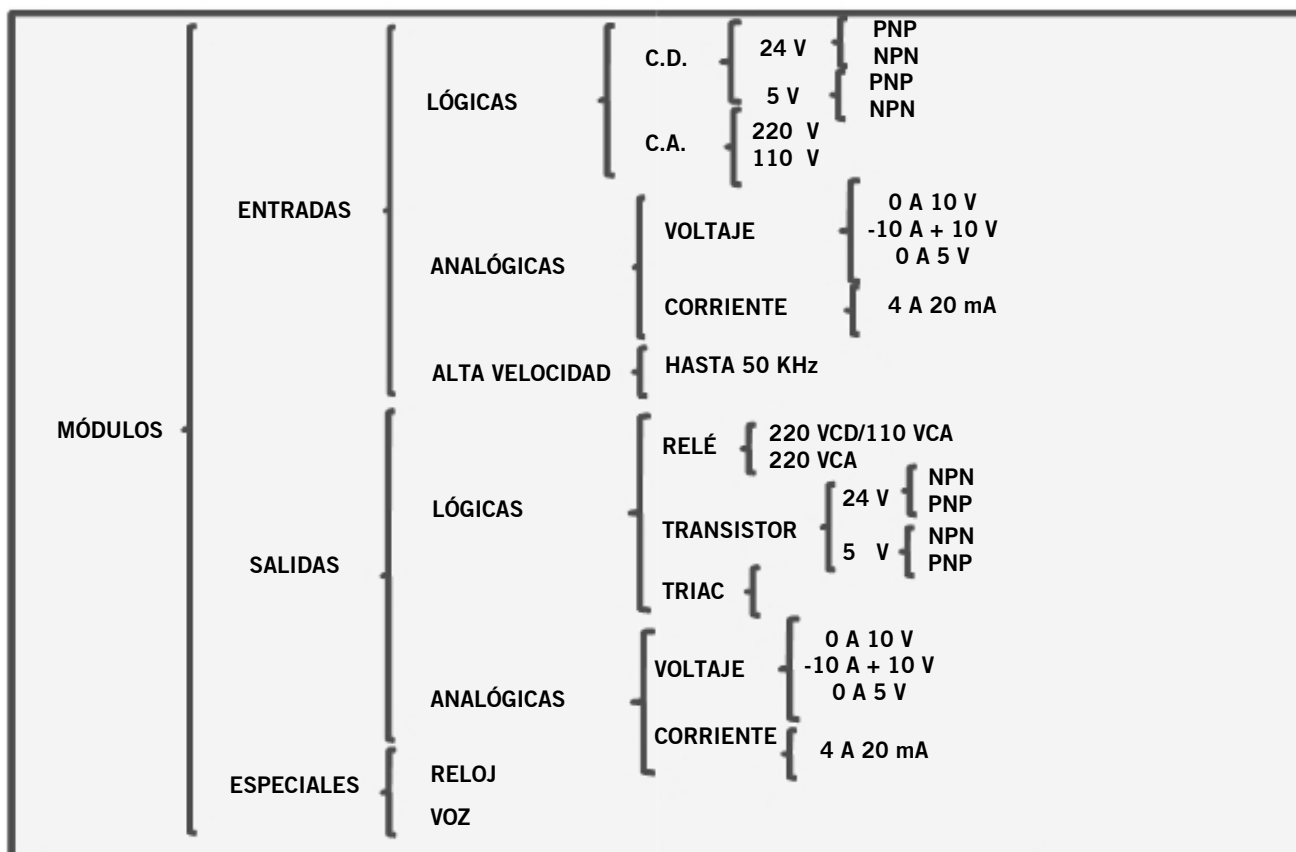
Tarjetas reguladoras de temperatura

Permiten recibir entradas estándar que provienen de termocuplas o termorresistencias y realizan funciones de control de lazo cerrado.

Controles de posición PID

Proporcional, integral y derivativo. Estos módulos son para ser utilizados en instrumentación de procesos.

Clasificación de los módulos de PLC



5 Programador

En general existen dos formas de programar o monitorear el programa de un PLC:

Programador Manual

El programador manual presenta teclas **orientadas a la programación y monitoreo de diagramas escalera**. Son de aplicación específica, esto es, sólo puede ser utilizado en el PLC para el cual fue diseñado.

La programación con el programador manual suele ser larga y tediosa, razón por la cual en la mayoría de los casos se utiliza para hacer pequeñas modificaciones al programa ó monitorear el estado de entradas, salidas o registros.



Computadora

El sistema más utilizado para programar PLCs es la computadora personal, y en especial la computadora portátil, debido a su poco peso y tamaño. En estos casos se instala un programa que permite establecer comunicación con el PLC. Este programa además presenta facilidades para la manipulación de símbolos así como para el monitoreo de variables y registros. El programa puede ser o no del mismo fabricante del PLC y generalmente es de uso restringido (requiere clave de acceso). Por supuesto, las ventajas de memoria, capacidad de procesamiento y manejo de información hacen a la computadora, por mucho, el dispositivo de programación y monitoreo de PLCs más utilizado en la actualidad.

6 Base o Rack

La base o rack es una estructura sobre la cual se instalan los módulos del PLC. En general, se considera que existen tres tipos de bases (racks): base principal o del CPU, base local o de expansión y base remota.

Base principal o del CPU

Contiene la CPU, normalmente contiene módulos de entrada/salida. En sistemas grandes de PLCs se puede tener sólo módulos de procesamiento, fuente de poder y de comunicación.

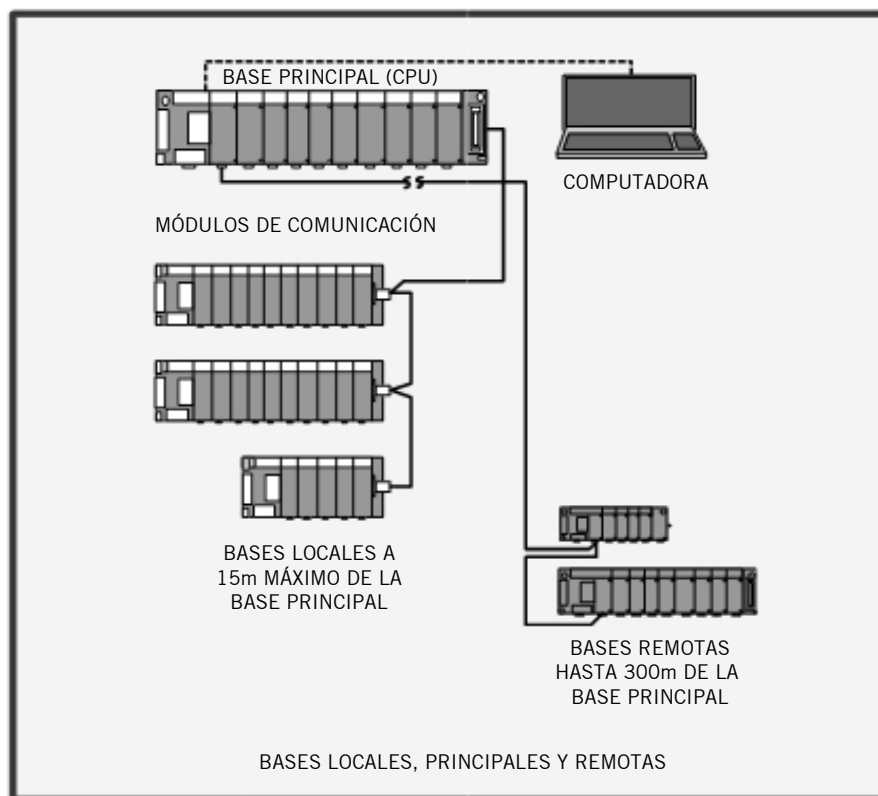
Base local o de expansión

No tienen CPU, pero generalmente cuentan con su propia fuente de poder. Se instalan a una distancia máxima de 15m de la base principal y no requieren de módulo para comunicación. Estas unidades de expansión se comunican a través de un cable que se conecta directamente entre la base principal y la base local. Cuando la base de la CPU no tiene capacidad para aceptar todos los módulos necesarios de interfase (entrada/salida) se usan las bases o racks locales o de expansión.

Base Remota

Se utilizan cuando una buena cantidad de sensores o dispositivos de campo están alejados del lugar donde se encuentra la CPU. Con esto, los módulos de entrada y salida se pueden colocar cerca de los sensores (reduciendo el problema de interferencia, ruido o distorsión en señales) y se reduce significativamente los costos de instalación y cableado. Las bases o racks remotos pueden colocarse a distancias considerables de la base principal (normalmente hasta 1000 pies ó 304.8 metros). Las bases remotas no contienen CPU, y debido a la distancia, requieren de un módulo de comunicación que permita intercambiar información de entradas/salidas con otro módulo de comunicación existente en la base principal.

En la figura se muestra un ejemplo de las racks o bases:



Tamaños de los PLCs

La clasificación de los PLCs en cuanto a su tamaño se realiza en función del número de sus entradas/salidas. Son admitidos los tres grupos siguientes:

Gama Baja

Hasta un máximo de 128 entradas/salidas. La memoria de usuario suele alcanzar un valor máximo de 4,000 instrucciones.

Gama Media

De 128 a 512 entradas/salidas. La memoria de usuario suele alcanzar un máximo de hasta 16,000 instrucciones.

Gama Alta

Más de 512 entradas/salidas. Su memoria de usuario puede superar las 100,000 instrucciones.

ACTIVIDAD 5. Partes de un PLC

Se mostraron los distintos componentes de un PLC.



Indique si las siguientes afirmaciones son verdaderas o falsas.

| | | VERDADERO | FALSO |
|---|---|--------------------------|--------------------------|
| 1 | En la memoria del PLC se realizan todas las operaciones lógicas. | <input type="checkbox"/> | <input type="checkbox"/> |
| 2 | El PLC realiza el proceso con tal rapidez que es imposible observarlo en una pantalla de computadora | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | Los buses de un microprocesador son los que se encargan de transmitir datos, instrucciones, señales, etc. entre las distintas partes del mismo. | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | No es posible aumentar la velocidad de ejecución de operaciones en la CPU del PLC. | <input type="checkbox"/> | <input type="checkbox"/> |
| 5 | La única forma de programar un PLC es utilizando una computadora personal. | <input type="checkbox"/> | <input type="checkbox"/> |
| 6 | El tamaño del PLC es independiente de la cantidad de entradas y salidas que tenga. | <input type="checkbox"/> | <input type="checkbox"/> |

¡Felicitaciones!

Usted ha finalizado el capítulo 3.

A continuación se desarrollará el capítulo configuración Lógica en el PLC



Lógica en el PLC

TEMAS DEL CAPÍTULO 4

4.1 Operaciones Lógicas Básicas

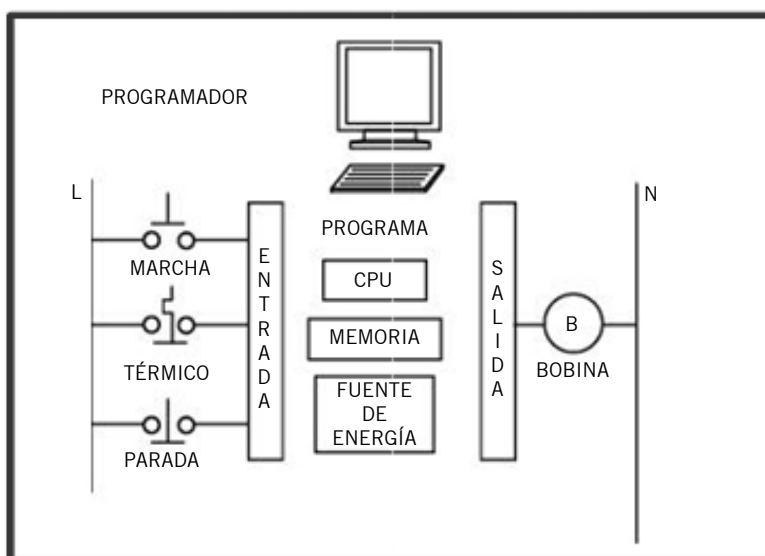
43

El PLC, como su nombre lo indica sigue una lógica en su funcionamiento. La programación está basada en operaciones lógicas y sencillas de realizar, aun por personas no expertas en lenguajes de programación.



4.1 Operaciones Lógicas Básicas

Programas en los PLCs



Programas en los PLCs

En la actualidad, el software de programación suele instalarse en el disco duro de una computadora portátil, permitiendo con esto la facilidad de su transporte y su uso en diferentes aplicaciones.

El software de programación se adquiere y se instala siguiendo el procedimiento indicado por el fabricante, que generalmente requiere de alguna llave de acceso.

Operaciones básicas

Operaciones básicas con bloques funcionales


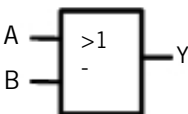
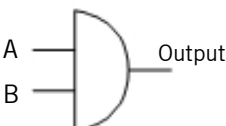
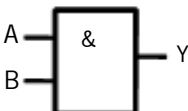
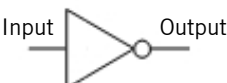
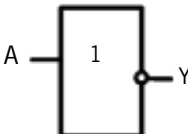
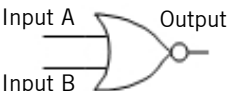
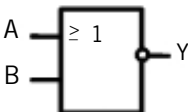
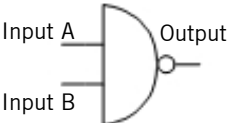
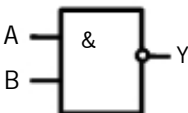

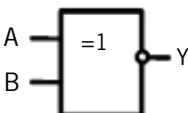
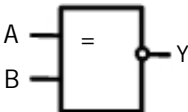
Todas las combinaciones posibles de las condiciones pueden ser representadas en una tabla, la cual se conoce como **Tabla de Verdad**.

¿Por qué debemos tomar decisiones lógicas?

Al igual que en la vida cotidiana, la operación de cualquier maquinaria depende de condiciones de operación y seguridad que permiten mantener la producción sin riesgo para el personal y para la maquinaria misma; así como asegurarse que en procesos continuos, se realicen en la secuencia que aseguren que el producto es manufacturado en tiempo y calidad.

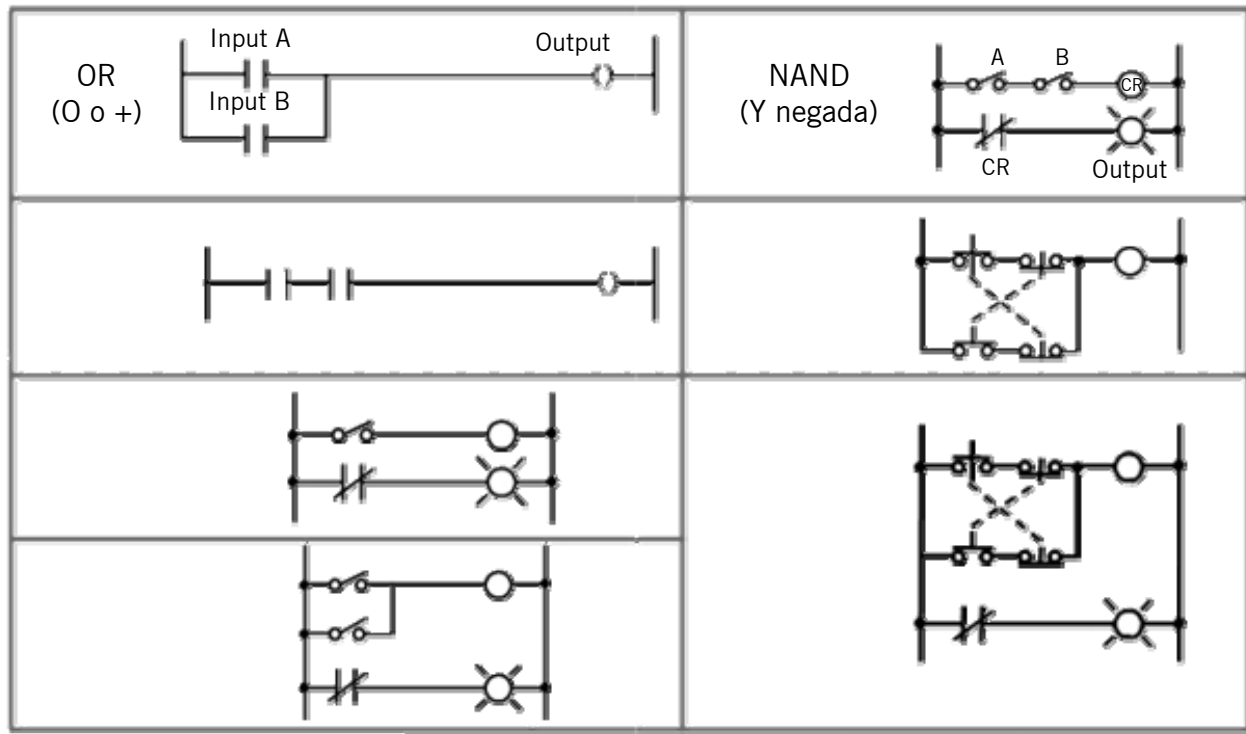
EJEMPLO

En la vida cotidiana tomamos decisiones, como las que se presentan en el caso de conducir un auto. Si el conductor ve durante su marcha un obstáculo o un semáforo en rojo, detiene el vehículo. Cada una de estas condiciones son variables a dos estados, esto es semáforo en rojo o no en rojo, obstáculo presente o no presente.

| Operación | Descripción | Ecuación | Símbolo Tradicional | Símbolo ANSI/IEEE | Tabla de verdad | | | | | | | | | | | | | | | |
|--------------------------------|--|------------------------|---|--|---|---------|---------|--------|---|---|---|---|---|---|---|---|---|---|---|---|
| OR (O o +) | Para que el resultado sea verdadero, es necesario que cualquier entrada sea verdadera. | $Y = A + B$ |  |  | <table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | A | B | Output | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| A | B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| AND (Y o x) | Para que el resultado sea verdadero, es necesario que todas las entradas sean verdaderas. | $Y = A \times B$ |  |  | <table><tr><th>Input A</th><th>Input B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | Input A | Input B | Output | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| Input A | Input B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| NOT (no) | Para que el resultado sea 1, es necesario que toda la entrada sea 0. | $Y = \bar{A}$ |  |  | <table><tr><th>Input</th><th>Output</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table> | Input | Output | 0 | 1 | 1 | 0 | | | | | | | | | |
| Input | Output | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | |
| NOR (O negada) | Para que el resultado sea verdadero, es invertido de la operación OR. | $Y = \overline{A + B}$ |  |  | <table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Output | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| A | B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| NAND (Y negada) | Para que el resultado sea verdadero, es invertido de la operación Y. | $Y = A \times B$ |  |  | <table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Output | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| A | B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| XOR (O exclusiva) | Combinación de las operaciones AND, OR y NOT: la salida será verdadera si las entradas son diferentes. | $Y = A \oplus B$ |  |  | <table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Output | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| A | B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| XOR (O exclusiva invertida) | Combinación de las operaciones AND, OR y NOT: la salida será verdadera si las entradas son diferentes. | $Y = A \oplus B$ | |  | <table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Output | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| A | B | Output | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |

EJEMPLO

A continuación se presentarán algunos ejemplos de las operaciones previamente desarrolladas.



ACTIVIDAD 6.

Una con flechas cada uno de los símbolos ANSI con la operación correspondiente.



NOT

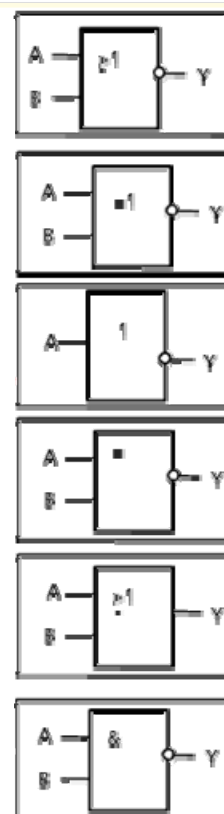
NAND

XOR

OR

NOR



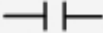

XNOR



4.2 Contactos, Cortos, Bobinas y Multivibradores

Contactos

El contacto de relé es el elemento básico de programación. Puede referirse a una bobina lógica (Ox) o a una entrada discreta (Ix). Hay 4 tipos de contactos:

| | | |
|-----------------------|---|---|
| Normalmente Abierto | Permite el paso de energía cuando referidas a él están en ON. |  |
| Normalmente Cerrado | Permite el paso de energía cuando referidas a él están en OFF |  |
| Transicional Positivo | Permite el paso de energía sólo por un scan, cuando ocurre una transición de OFF a ON. |  |
| Transicional Negativo | Permite el paso de la energía sólo por un scan, cuando ocurre una transición de ON a OFF. |  |

Cortos horizontales y verticales

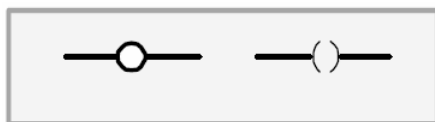
Los cortos son simplemente líneas rectas de conexión entre contactos y/o bloques de función.

| | |
|---------------------|--|
| Corto Vertical | Conecta contactos o bloques de función ubicados uno encima del otro, en una columna de la red. Un corto vertical no consume memoria del usuario. |
| Normalmente Cerrado | Se utilizan en combinación con los verticales para expandir la lógica dentro de la red sin romper el flujo de energía. |

Bobinas

Las bobinas se utilizan para activar la lógica dentro del programa del usuario y/o para controlar una salida discreta. Cuando una bobina no activa una salida se le denomina interna y, en ese caso, tiene el propósito de seguir la secuencia o lógica.

Las bobinas se pueden representar de dos maneras:



Las bobinas se encienden cuando se les aplica energía y se apagan cuando se les retira la energía.

Una bobina tiene un valor de salida discreto representado por un número de referencia Ox. Como los valores de salida son actualizados en la RAM estática por la CPU del controlador, una bobina puede usarse internamente en el programa lógico o externamente, a través del traffic cop, en un módulo de salida discreta. Existen 2 tipos de bobinas:

- Bobina normal.
- Bobina retenida.

Si una bobina retenida ha estado energizada en el momento en que se ha caído la alimentación del controlador, tomará ese mismo estado por un scan una vez que la alimentación haya sido reestablecida.

Cada red puede contener un máximo de 7 bobinas. Cada número de referencia Ox puede utilizarse como bobina sólo una vez, pero puede utilizarse para referenciar a más de un contacto.

Puentes verticales y horizontales

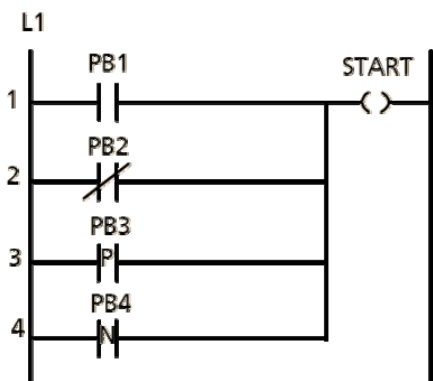
Los puentes verticales y horizontales son simplemente conexiones mediante una línea recta entre elementos lógicos y/o instrucciones en un diagrama escalera. Estos se muestran en la figura:



EJEMPLO



Uso de contactos, bobinas y puentes:



En este caso se tienen cuatro contactos (de diferente tipo cada uno). Éstos contactos están relacionados con entradas discretas. Además se tiene una bobina que representa una salida discreta.

Todos los contactos se encuentran unidos a la bobina mediante puentes verticales y/o horizontales. La bobina se energizará cuando se cumpla una de las siguientes condiciones:

1. La entrada discreta PB1 se encuentre energizada.
2. La entrada discreta PB2 se encuentre desenergizada.
3. La entrada discreta PB3 registre un cambio de desenergizada
4. La entrada discreta PB4 registre un cambio de energizada a desenergizada

Capacidad de Habilitación/Inhabilitación

A través del software del panel de programación (o una PC compatible con IBM), usted puede inhabilitar (DISABLE) una bobina lógica o una entrada discreta dentro del programa lógico.

La condición de inhabilitación provocará que el elemento de campo de la entrada no tenga control sobre la 1x (bobina discreta) que le fue asignada, y que la lógica no tenga control sobre el valor inhabilitado de salida 0x (bobina lógica).

IMPORTANTE



El interruptor Protección de Memoria debe estar en OFF antes de inhabilitar (o habilitar) una bobina o entrada discreta.

Forzado de Discretos en ON y OFF

El software de programación provee también la posibilidad de forzar ON y forzar OFF una variable discreta. Cuando una bobina o una entrada discreta han sido inhabilitadas, la manera de cambiar su estado de OFF a ON es con FORCE ON y la única forma de cambiarlo de ON a OFF es con FORCE OFF. Cuando una bobina o una entrada discreta están habilitadas (ENABLE), no pueden ser forzadas en ON/OFF.

Multivibrador

Un multivibrador es un circuito oscilador capaz de generar una onda cuadrada:

| | | |
|---------------|-----------------|--|
| Multivibrador | Monoestable | El monoestable es un circuito multivibrador que realiza una función secuencial que, al recibir una excitación exterior, cambia de estado y se mantiene en él durante un período que viene determinado por una constante de tiempo. Transcurrido dicho período de tiempo, la salida del monoestable vuelve a su estado original. |
| | Biestable | El biestable, también llamado báscula (flip-flop en inglés), es un multivibrador capaz de permanecer en un estado determinado o en el contrario durante un tiempo indefinido. Esta característica es ampliamente utilizada en electrónica digital para memorizar información. El paso de un estado a otro se realiza variando sus entradas. |
| | Astable | El astable es un multivibrador que no tiene ningún estado estable, lo que significa que posee dos estados "cuasi-estables" entre los que conmuta, permaneciendo en cada uno de ellos un tiempo determinado. La frecuencia de conmutación depende, en general, de la carga y descarga de condensadores. Entre sus múltiples aplicaciones se cuentan la generación de ondas periódicas (generador de reloj) y de trenes de impulsos. |
| | Schmitt Trigger | Un Schmitt Trigger cambia su estado de salida cuando la tensión en su entrada sobrepasa determinado nivel. La salida no vuelve a cambiar cuando baja el voltaje de la entrada, sino que el nivel de tensión que determina el cambio es otro distinto, más bajo que el primero. A este efecto se conoce como ciclo de histéresis. Ésta es la principal diferencia con un comparador normal, que es un simple amplificador operacional sin realimentación, y que su salida depende únicamente de la entrada mayor. |

ACTIVIDAD 7.

Marque la opción correcta.



- Un contacto normalmente abierto permitirá que la bobina se energice cuando:

Él esté energizado. ☐

Él esté desenergizado. ☐
- Un transicional negativo energizará la bobina cuando:

Pasé de estar energizado a desenergizado. ☐

Páse de estar desenergizado a energizado. ☐
- Una bobina que soluciona la secuencia o lógica es:

Externa. ☐

Interna. ☐
- Una entrada discreta o una bobina lógica

Puede ser habilitada o inhabilitada. ☐

No puede ser habilitada o inhabilitada. ☐
- Los multivibradores generan ondas

Sinusoidales ☐

Cuadradas ☐

4.3 Sistemas Numéricos

El Sistema Binario

Es un sistema de numeración en el que los números se representan utilizando solamente las cifras cero y uno (0 y 1).

Los ordenadores trabajan internamente con dos niveles de voltaje, por lo que su sistema de numeración natural es el sistema binario (**encendido 1, apagado 0**).

EJEMPLO

Número binario: 101011

| | | | | |
|---|---|----|---|-------------------|
| 1 | x | 1 | = | 1 |
| 1 | x | 2 | = | 2 |
| 0 | x | 4 | = | 0 |
| 1 | x | 8 | = | 8 |
| 0 | x | 16 | = | 0 |
| 1 | x | 32 | = | 32 |
| | | | | <hr/> |
| | | | | 43 _{Dec} |

El Sistema Hexadecimal

A veces abreviado como **hex**, es el sistema de numeración posicional de base 16, que emplea por tanto 16 símbolos.

0 - 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - A - B - C - D - E - F
10 11 12 13 14 15

| | | | |
|--------|--------|--------|-----------|
| 256 | 16 | 1 | 0.0625 |
| 16^2 | 16^1 | 16^0 | 16^{-1} |

EJEMPLO

$$\begin{array}{rcll} \text{A3 (Hex)} & & & \\ 3 \times 1 & = & 3 & \\ \text{A(10)} \times 16 & = & 160 & \\ & & & \hline & & & 163 \text{ (Dec)} \end{array}$$

Binary-Coded-Decimal

Es un sistema numérico usado en sistemas computacionales y electrónicos **para codificar números enteros positivos y facilitar las operaciones aritméticas**. Es un código pesado debido a que representa los dígitos con un orden específico.

Estructura del dígito en B.C.D. (Peso)

| | | | |
|-------|-------|-------|-------|
| 8 | 4 | 2 | 1 |
| 2^3 | 2^2 | 2^1 | 2^0 |

EJEMPLO

1001 1000 0101 0011_{BCD} = 9853₁₀
0010 0101 0000 0011_{BCD} = 2503₁₀
0101 0111_{BCD} = 57₁₀

Código Gray

Es un caso particular de sistema binario. Consiste en una ordenación de $2n$ números binarios de tal forma que cada número sólo tenga un dígito binario distinto a su predecesor.

Para pasar un número binario al código binario Gray, hay una regla fácil de implementar en un lenguaje de programación:

- Un número en binario siempre empieza en 1. Los ceros a la izquierda no cuentan, tampoco en Gray.

Ej.: 1000011110000 en binario se escribe
1xxxxXXXXxxx

- Ahora nos fijamos en el segundo dígito. Si es igual al dígito anterior se pone un 0 (no cambia); si es diferente, como en este caso, pues el dígito anterior era un 1 y el que observamos un 0, se pondrá un 1 (cambia).

Ej.: El número del ejemplo anterior será:
11xxxXXXXxxx

- En los casos sucesivos se repite el paso anterior, observando en el número binario 'natural' el dígito anterior al que se evalúa.

Ej.: El número del ejemplo anterior, pasado a código Gray será: 1100010001000

Esta técnica de codificación se originó cuando los circuitos lógicos digitales se realizaban con válvulas de vacío y dispositivos electromecánicos. Los contadores necesitaban potencias muy elevadas a la entrada y generaban picos de ruido cuando varios bits cambiaban simultáneamente. El uso de código Gray garantizó que en cualquier transición variará tan sólo un bit.

EJEMPLO

| | | |
|------------------------|--------|-------------------|
| Número binario: 101011 | 1 x 1 | =1 |
| | 1 x 2 | =2 |
| | 0 x 4 | =0 |
| | 1 x 8 | =8 |
| | 0 x 16 | =0 |
| | 1 x 32 | =32 |
| | | <hr/> |
| | | 43 _{Dec} |

Otra técnica sencilla para pasar de binario a Gray sin usar un lenguaje de programación es:

1. Aplicar un XOR (disyunción exclusiva, también se puede sumar cada bit individualmente descartando cualquier acarreo) del número a sí mismo pero con un acarreo a la derecha
2. Eliminar el bit del extremo derecho.
3. Los acarreo tienen que descartarse .

EJEMPLO

Pasar diez (1010) en código binario a Gray será 1111:

| |
|-------|
| 1010 |
| 101 |
| <hr/> |
| 1111 |

Otros ejemplos

| | |
|--------|--------------|
| 111000 | 110101010001 |
| 11100 | 11010101000 |
| <hr/> | <hr/> |
| 100100 | 10111111001 |

Gray a Binario

Hacer el cambio contrario es simplemente invertir lo que hace el anterior, de forma que si se encuentra un cero (siempre que no sea al principio) se debe poner la cifra anterior; en cambio si se coloca un 1 es porque la cifra ha cambiado así que si había un 0 ahora se colocará un 1 y viceversa

RECUERDE

1. El primer dígito del código Gray será el mismo que el del binario
2. Si el segundo dígito del código Gray es "0", el segundo dígito binario es igual al primer dígito binario, si este dígito es "1" el segundo dígito binario es el inverso del primer dígito binario.
3. Si el tercer dígito del código Gray es "0", el tercer dígito binario es igual al segundo dígito binario, si este dígito es "1", el tercer dígito binario es el inverso del segundo dígito binario.... y así hasta terminar.



Código ASCII

El código **ASCII** (acrónimo inglés de American Standard Code for Information Interchange -Código Estadounidense Estándar para el Intercambio de Información), pronunciado generalmente [áski], es un código de caracteres basado en el alfabeto latino tal como se usa en inglés moderno y en otras lenguas occidentales.

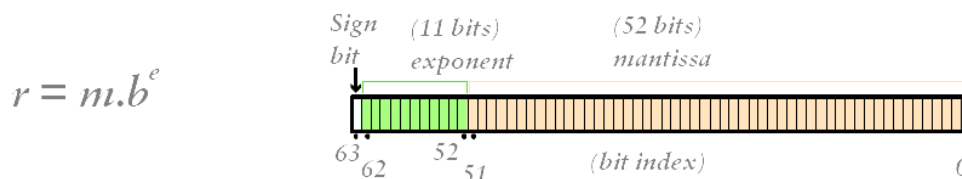
Fue creado en 1963 por el Comité Estadounidense de Estándares (ASA, conocido desde 1969 como el Instituto Estadounidense de Estándares Nacionales, o ANSI) como una refundición o evolución de los conjuntos de códigos utilizados entonces en telegrafía. Más tarde, en 1967, se incluyeron las minúsculas, y se redefinieron algunos códigos de control para formar el código conocido como **US-ASCII**.

Coma flotante o punto flotante

Es un método de representación de números reales que se puede adaptar al orden de magnitud del valor a representar, usualmente trasladando la coma decimal -mediante un exponente- hacia la posición de la primera cifra significativa del valor.

De esta forma, con un número dado de dígitos representativos se obtiene mayor precisión que con la coma fija, debido a que el valor de estos dígitos es siempre significativo sea el que sea el orden de magnitud del número a representar. Debido a esta adaptación, permite representar un rango mucho mayor de números (determinado por los valores límite que puede tomar el exponente).

Una representación en coma flotante se compone de tres números (campos) que siguen el siguiente patrón:



r: valor real del número a representar.

m: mantisa o significando, dígitos significativos del número. El tamaño máximo de este campo, usualmente fijo y limitado, **determina la precisión de la representación**. Este campo está usualmente normalizado, es decir, su parte entera sólo consta de un dígito (que será la primera cifra significativa del número a representar).

b: base del sistema de representación (10 en sistema decimal, 8 en sistema octal, 2 en sistema binario, etc.).

e: exponente, orden de magnitud del significando. El mínimo y máximo valor posible del exponente determinan el rango de valores representables. Cabe añadir que **cuando e vale cero el valor real coincide con el significando**.

4.4 Registros

Pueden definirse como **posiciones de memoria en las que se guardan valores numéricos**.

Pueden ser

Registros de entrada (Fórmula 30XXX)

- En ellos se archivan valores de entrada analógicas (previo paso por el módulo D/A) y también cantidad de pulsos que corresponden a entradas discretas de alta frecuencia que no pueden ser retenidas por el PLC y deben pasar previamente por un módulo HSC.
- Estos registros no pueden ser alterados desde el PLC a través del programador P190.
- Su rango es desde 0 (cero) a 9999.

Registros de retención (Fórmula 4XXXX)

- En ellos se archivan valores numéricos necesarios para la programación (por ejemplo, el resultado de una suma o cualquier otra operación).
- Estos registros pueden ser alterados vía P-190. Su valor queda retenido aún en caso de falta de energía.
- Los registros de retención NO pierden su valor en caso de falta de energía.
- No existe límite para la cantidad de veces que estos registros pueden ser usados en el programa. Tienen 16 bits. El bit de mayor valor es el #1.



Registros de retención (Fórmula 4XXXX)

- Constituyen un tipo especial de registro de retención.
- Su contenido puede usarse como entrada a módulo conversores D/A que permiten obtener señales analógicas de salida del PLC (4 a 20 mA, etc.)
- Todos los registros pueden ser codificados en decimal, binario o hexadecimal.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|---|---|--------------------|------------------|---------------|-------|------------|--------------|-----------|---|---|---|---|--------------|-----------|---|---|---|------------|---------------|------------|---|---|---|------------------|-----------|-------|---|---|---|------------------|------|
| <div>TON</div> <div>TIMER ON DELAY</div> <div><div>TimerT4:1</div><div>Base de tiempo1.0</div><div>Preestablecido15</div><div>Acumulado0</div></div> | <div>RETARDO AL ENERGIZAR TON</div> <div>Bits de estado</div> <div>EN- Habilitar</div> <div>TT- Timer corriendo</div> <div>DN- Dine (Hecho)</div> | <div>Si las condiciones de entrada siguen verdaderas, el timer T4:1 comienza a hacer incrementos en intervalos de 1 segundo. Cuando el valor acumulado es mayor que o igual al valor preestablecido (15), el timer se detiene y pone a UNO el bit done del timer.</div> <table><tr><td>Condición del rung</td><td>EN 15</td><td>TT 13</td><td>DN 12</td><td>Valor Acum</td><td>Edo. Del TON</td></tr><tr><td>Falsa</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Restablecido</td></tr><tr><td>Verdadera</td><td>1</td><td>1</td><td>0</td><td>Incremento</td><td>Corriendo</td></tr><tr><td>Verdadera.</td><td>1</td><td>0</td><td>1</td><td>>=preestablecido</td><td>Done</td></tr></table> | Condición del rung | EN 15 | TT 13 | DN 12 | Valor Acum | Edo. Del TON | Falsa | 0 | 0 | 0 | 0 | Restablecido | Verdadera | 1 | 1 | 0 | Incremento | Corriendo | Verdadera. | 1 | 0 | 1 | >=preestablecido | Done | | | | | | |
| Condición del rung | EN 15 | TT 13 | DN 12 | Valor Acum | Edo. Del TON | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 0 | 0 | 0 | 0 | Restablecido | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Verdadera | 1 | 1 | 0 | Incremento | Corriendo | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Verdadera. | 1 | 0 | 1 | >=preestablecido | Done | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <div>TOF</div> <div>TIMER OFF DELAY</div> <div><div>TimerT4:1</div><div>Base de tiempo.01</div><div>Preestablecido180</div><div>Acumulado0</div></div> | <div>RETARDO AL DESENERGIZAR TON</div> <div>Bits de estado</div> <div>EN- Habilitar</div> <div>TT- Timer corriendo</div> <div>DN- Dine (Hecho)</div> <div>Timer Retentivo Encendido RTO</div> | <div>Si las condiciones de entrada son falsas, el timer T4:1 comienza a hacer incrementos en intervalos de 10mS mientras el rung permanezca falso. Cuando el valor acumulado es mayor que o igual al valor preestablecido (180), el timer se detiene y pone a UNO el bit done del timer.</div> <table><tr><td>Condición del rung</td><td>EN 15</td><td>TT 14</td><td>DN 13</td><td>Valor Acum</td><td>Edo. Del TOF</td></tr><tr><td>Verdadera</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Restablecido</td></tr><tr><td>Falsa</td><td>0</td><td>1</td><td>0</td><td>Incremento</td><td>Corriendo</td></tr><tr><td>Falsa</td><td>0</td><td>0</td><td>0</td><td>>=preestablecido</td><td>Done</td></tr></table> | Condición del rung | EN 15 | TT 14 | DN 13 | Valor Acum | Edo. Del TOF | Verdadera | 1 | 0 | 1 | 0 | Restablecido | Falsa | 0 | 1 | 0 | Incremento | Corriendo | Falsa | 0 | 0 | 0 | >=preestablecido | Done | | | | | | |
| Condición del rung | EN 15 | TT 14 | DN 13 | Valor Acum | Edo. Del TOF | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Verdadera | 1 | 0 | 1 | 0 | Restablecido | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 0 | 1 | 0 | Incremento | Corriendo | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 0 | 0 | 0 | >=preestablecido | Done | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <div>RTO</div> <div>RETENTIVE</div> <div>TIMER ON</div> <div><div>TimerT4:10</div><div>Base de tiempo1.0</div><div>Preestablecido10</div><div>Acumulado0</div></div> | <div>Bits de estado</div> <div>EN- Habilitar</div> <div>TT- Timer corriendo</div> <div>DN- Dine (Hecho)</div> | <div>Si las condiciones de entrada siguen verdaderas, el timer T4:10 comienza a hacer incrementos en intervalos de 1 segundo mientras el rung permanezca verdadero. Cuando el valor acumulado es mayor que o igual al valor preestablecido (10), el timer se detiene y pone a UNO el bit done del timer.</div> <table><tr><td>Condición del rung</td><td>EN 15</td><td>TT 14</td><td>DN 13</td><td>Valor Acum</td><td>Edo. Del TOF</td></tr><tr><td>Falsa</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Restablecido</td></tr><tr><td>Verdadera</td><td>1</td><td>1</td><td>0</td><td>Incremento</td><td>Deshabilitado</td></tr><tr><td>Falsa</td><td>0</td><td>0</td><td>0</td><td>Se mantiene</td><td>Corriendo</td></tr><tr><td>Falsa</td><td>1</td><td>0</td><td>1</td><td>>=preestablecido</td><td>Done</td></tr></table> | Condición del rung | EN 15 | TT 14 | DN 13 | Valor Acum | Edo. Del TOF | Falsa | 0 | 0 | 0 | 0 | Restablecido | Verdadera | 1 | 1 | 0 | Incremento | Deshabilitado | Falsa | 0 | 0 | 0 | Se mantiene | Corriendo | Falsa | 1 | 0 | 1 | >=preestablecido | Done |
| Condición del rung | EN 15 | TT 14 | DN 13 | Valor Acum | Edo. Del TOF | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 0 | 0 | 0 | 0 | Restablecido | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Verdadera | 1 | 1 | 0 | Incremento | Deshabilitado | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 0 | 0 | 0 | Se mantiene | Corriendo | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Falsa | 1 | 0 | 1 | >=preestablecido | Done | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <div>T4:1 Reset del timer --(RES)--RES.</div> | | <div>Si las condiciones de entrada siguen verdaderas, el timer T4:10 es reseteado. Esta instrucción reseta timerss, contadores, así como bloques de control. Necesaria para poner a cero el valor RTO acumulada</div> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

ACTIVIDAD 8. Lógica en el PLC

Se mostraron los distintos pasos lógicos en la programación de un PLC.



Por favor, responda las siguientes preguntas.

1 ¿Cuál de los siguientes números no están expresados en código binario

- ☐ 1
- ☐ 10010
- ☐ 10012

2 ¿Cuál de los siguientes números está expresado en código hexadecimal?

- ☐ 1.2
- ☐ 011
- ☐ 10^{-16}
- ☐ 6780

3 ¿Cuáles de los siguientes códigos no permite representar números decimales?

- ☐ Cama flotante/punto flotante
- ☐ Gray
- ☐ Binario

¡Felicitaciones!

Usted ha finalizado el capítulo 4.

A continuación se desarrollará el capítulo Codificación de Señales.

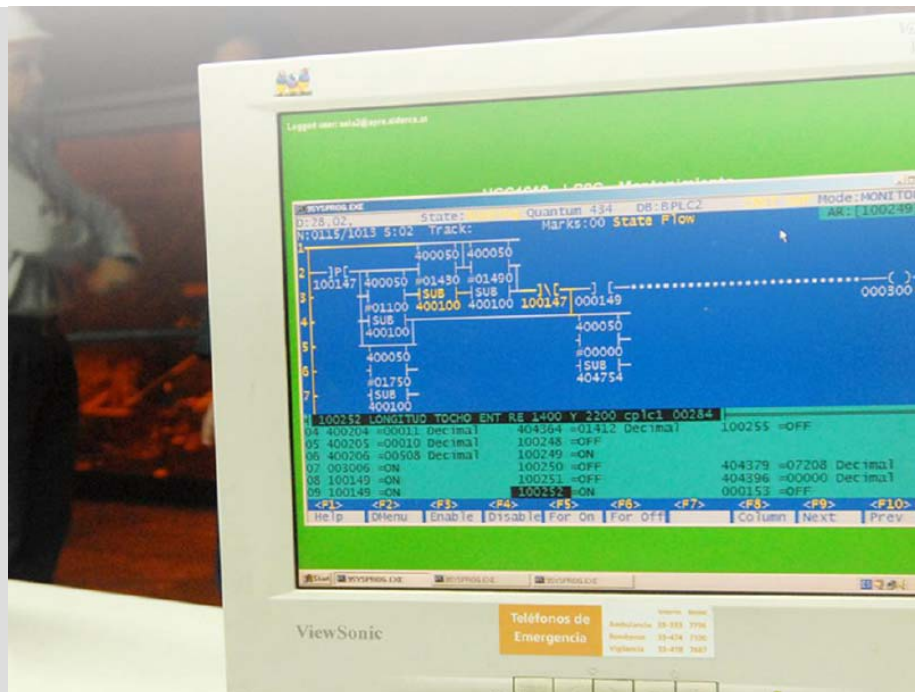


Codificación de señales

TEMAS DEL CAPÍTULO 5

| | |
|--|----|
| 5.1 Parámetros a Especificar | 57 |
| 5.2 Mantenimiento del PLC | 59 |
| 5.3 Control Centralizado y Distribuido | 62 |
| 5.4 Identificación y Codificación de Señales | 63 |

Los parámetros a verificar por el PLC y las distintas señales que recibe o entrega deben ser especificados de forma especial para este programa.



5.1 Parámetros a especificar

Configuración

El primer paso para la configuración es establecer el tipo de CPU a utilizar. Esto permite al *software* de programación asegurar que las opciones disponibles sean compatibles con el tipo de CPU utilizado.

Una vez definido el tipo de CPU o procesador del PLC, generalmente hay que seguir un proceso que incluye:

Configuración de los parámetros de comunicación

Se especifica:

- Protocolo de comunicación, esto tiene relación con el formato de los datos, su estructura.
- El número del puerto a través del cual se establecerá la comunicación.
- La velocidad de comunicación (Baud ó bits / seg).
- La paridad a utilizar (par ó impar).
- Número de bits de paro (Stop).
- La dirección del PLC (ADDRESS).

Configuración de la memoria

- Memoria para la lógica del diagrama escalera.
- Memoria para datos del usuario, tales como variables y constantes.
- Memoria del sistema, esto es, para las operaciones del sistema.

Configuración de las entradas y salidas

- Establecer la cantidad de bases o racks.
- Para cada base o rack establecer la cantidad de slots.
- Para cada slot establecer la cantidad y tipo de entradas o salidas.

En esta forma el CPU o procesador del PLC sabe dónde localizar cada señal de entrada y salida, en que rack, que slot y la naturaleza de la información

Ya instalado el software de programación en la computadora y la aplicación (que incluye la configuración), existen dos formas de trabajar: sin conectar y conectado al PLC. A esto se lo conoce normalmente como operación **online y offline**.

Modo de operación online y offline

Fundamentalmente existen dos modos de trabajar con el software de programación:

| online | offline |
|--|--|
| Cuando se trabaja en modo Online hay una comunicación entre el procesador (CPU) del PLC y la computadora. Para esto se requiere conectar un cable entre el PLC y la computadora. Los comandos pueden ser introducidos directamente en la memoria del PLC, así nadie puede acceder a información que presente en los módulos de entrada/salida. | Cuando se trabaja en modo Offline significa que no existe una comunicación entre el procesador (CPU) del PLC y la computadora. En este modo se diseña sobre la memoria de la misma computadora. Los resultados después pueden ser introducidos al PLC trabajando Online. |

Dependiendo de la forma en que se esté trabajando, ciertas opciones de los menús estarán disponible sólo en el modo Offline u Online.

5.2 Mantenimiento del PLC

Existen una gran cantidad de opciones en los diferentes **software** de programación y todos tienen una **serie de comandos que permiten realizar las tareas de mantenimiento del PLC**:

Lectura y transferencia de un programa

Una de las operaciones comunes es la **lectura del programa guardado en la memoria del PLC**. Para realizar esta tarea **es necesario estar Online**, es decir, que exista comunicación entre el PLC y la computadora.

Otro caso es la **lectura de un programa que se encuentra en la memoria de la computadora**. En este caso, se **opera en Offline** y se guarda en un disco. Para utilizarlo es necesario transferirlo a la memoria del PLC. Recuerde que para hacerlo, es necesario estar en operación Online, es decir, con comunicación entre el PLC y la computadora.

Comparación de programas

Una de las ventajas de los sistemas hechos con PLC es su facilidad para poder hacer cambios. Por ejemplo, una forma nueva y más eficiente de resolver una secuencia puede ser puesta a prueba durante una semana, ó si se presenta una situación de falla, se puede en forma provisional modificar el diagrama escalera mientras se consigue alguna refacción.

Una práctica común para evitar esto es comparar el programa que está ejecutándose en el PLC con la versión original. Así, una instrucción Compare (comparar) permitirá detectar diferencias y así asegurar que el programa en el PLC esté operando en forma adecuada. Compare generalmente genera una lista de las diferencias entre dos programas.

Creación de un respaldo

Frecuentemente en los PLC **hay una batería de respaldo que mantiene el programa y continua energizando la memoria cuando se suspende el suministro eléctrico**.

¿Qué hacer cuando el módulo del procesador (CPU) del PLC se daña?, ¿Qué hacer si la batería se agota o daña y se presenta una interrupción del suministro eléctrico?

La respuesta para ambos casos es volver a cargar en la memoria del PLC, el programa que contiene el diagrama escalera. Para hacerlo se requiere haber hecho con anterioridad un respaldo.

Un punto importante aquí es **asegurar tener siempre un respaldo actualizado**,. que incluya las últimas modificaciones que se le hayan hecho al diagrama escalera.

Para hacer un respaldo se requiere estar en operación Online, y utilizar algún comando como Backup (respaldar) o en ciertos casos un Guardar Como (Save As) en el disco duro de la computadora. También se recomienda, sobre todo en el caso de tener varios PLC, mantener un respaldo en la red.


Monitoreo

Otra herramienta importante en el software de programación del PLC es la que permite monitorear el estado de entradas o salidas discretas o analógicas, así como el contenido de registros.

Por supuesto que el monitoreo solo se puede realizar Online, y con el PLC corriendo el programa (run).

En algunos software es posible realizar el monitoreo **a través de una pantalla**. En esta pantalla se pueden especificar las direcciones de entradas, salidas o lugares de memoria que se desean monitorear. En otros es posible realizarlo **directamente en la pantalla de edición de la lógica o diagrama escalera**. En este caso, el color de cada componente (contactos, bobinas, etc.) cambia según su estado (on u off).

EJEMPLO




Un ejemplo de monitoreo es determinar el buen funcionamiento de un dispositivo de campo, por ejemplo un interruptor de límite que este fallando. Cuando se tiene un interruptor conectado a un módulo de entradas discretas del PLC, se puede revisar el estado de esa entrada y observar su comportamiento mediante el monitoreo.

Búsqueda de un elemento

Cuando no se sabe la ubicación de cierto elemento o instrucción que forma parte de un diagrama escalera, se puede encontrar mediante el uso de una búsqueda automática.

Esta sirve para determinar la ubicación de un determinado dispositivo, así como para conocer sus características. La búsqueda, que puede ser activada mediante un comando Search o Find, permite encontrar rápidamente los diferentes puntos donde se encuentra una instrucción o elemento.

PREGUNTAS



Un ejemplo de monitoreo es determinar el buen funcionamiento de un dispositivo de campo, por ejemplo un interruptor de límite que este fallando. Cuando se tiene un interruptor conectado a un módulo de entradas discretas del PLC, se puede revisar el estado de esa entrada y observar su comportamiento mediante el monitoreo.

Buscar una instrucción o elemento en un programa o diagrama escalera grande puede ser una tarea larga y no muy confiable. El software de programación de PLC proporciona un comando para realizar esta tarea.

Forzado

Otra herramienta común en el software de programación es la de forzado (force). **Mediante el comando Force se puede hacer que una entrada o una salida se active o desactive independientemente del estado de la entrada o de la salida ó del resultado de ejecutar la lógica del diagrama escalera.**

Es decir, se puede hacer que una bobina, contacto, o registro del diagrama escalera cambien de estado (ON / OFF).

Forzar a activar (Force On) activa una entrada ó activa una salida y Forzar a desactivar (Force Off) realiza la función opuesta.

IMPORTANTE

Antes de forzar una entrada, una salida ó algún registro es necesario asegurarse de que es absolutamente seguro realizar esto.

Ya una vez resuelto el problema, se puede quitar el estado forzado regresando a su estado normal de operación: las entradas dependen de los elementos, sensores, botones, interruptores, etc. del proceso, en tanto que el estado de las salidas dependen del resultado de la ejecución del programa o diagrama escalera.

Agregar, quitar y editar un elemento

El propósito fundamental de la edición de diagramas escalera es añadir, quitar o modificar los elementos de la lógica. El modo de edición suele requerir cambiar el estado del PLC del modo de ejecución (run) al de programación (program).

Para editar un elemento **generalmente se utiliza la instrucción editar (Edit), modificar (Modify) o reemplazar (Replace).** Ya en el escalón del diagrama escalera que se requiere modificar, se ejecuta la instrucción editar, con lo que se define el nuevo tipo de elemento y/o su número de referencia

Descripción y Comentarios

Una buena costumbre en el diseño de la lógica de secuencia o diagrama escalera es agregar descripciones y comentarios a los distintos elementos, esto **permitirá recordar o entender más fácilmente la tarea de secuencia a realizar en cada sección de la lógica.** La introducción de descripción y comentarios **puede ser hecha tanto en operación Online como en Offline**

Generalmente los datos introducidos en las descripciones y comentarios **no se guardan en el PLC, sino en la computadora que tiene el programa de respaldo** importante debido a que, aún cuando cualquier computadora con el software de programación pueda conectarse al PLC, sólo la que tenga los archivos de comentarios y descripciones permitirá verlos.

5.3 Control centralizado y Distribuido

La mayoría de los PLC **pueden ser interconectados con otros PLC**. Pudiendo comunicarse entre sí para intercambiar información acerca del proceso que están controlando

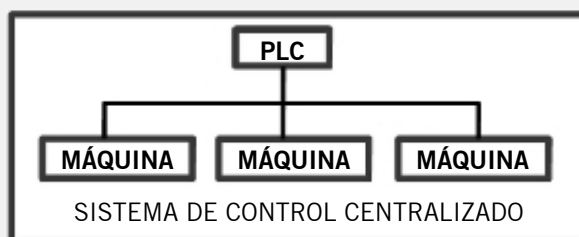
Cuando es necesario controlar varias máquinas usando PLC, se puede tener dos tipos de configuración:

- Control centralizado.
- Control distribuido.

Control Centralizado

Cuando **varias máquinas o procesos son controlados por un único PLC**. Se puede hacer uso de módulos remotos de E/S para ahorrar en conductores y canalizaciones para estos. El control centralizado se usa cuando la complejidad de los procesos es tal que la descentralización se vuelve muy compleja. La **principal desventaja consiste en que si el PLC falta se para todo el proceso**.

En procesos críticos se recurre a sistemas redundantes, es decir, a tener un PLC de reserva listo para entrar en funcionamiento cuando falla el que está funcionando.

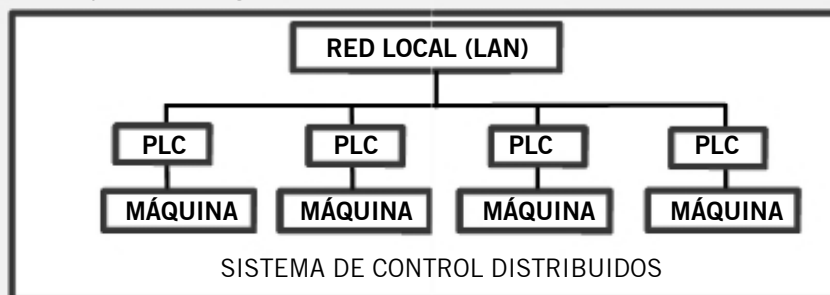


Control Distribuido

Se implementan mediante dos o más PLC, donde cada uno controla un área del proceso y están coordinados de tal manera que funcionan como un conjunto. Los PLC estarán **interconectados entre sí para formar una red local** (LAN o Local Área Network), intercambiando continuamente información entre ellos para tener la coordinación necesaria.

Si uno de los PLC falla los demás pueden seguir funcionando normalmente. Sólo es necesario redundancia en los PLC considerados críticos.

Normalmente los PLC provenientes de diferentes fabricantes no pueden comunicarse entre sí
En la figura se expone un diagrama de conexión de una red local (LAN).



5.4 Identificación y Codificación de señales

Identificación de señales

Para identificar claramente las señales que se manejan en el PLC, es necesario darles un nombre o designación. La designación de las señales puede hacerse considerando una única señal (1 bit) o un grupo de ellas (byte o palabra).

Las señales de entrada y salida llegan y salen físicamente de las terminales o bornes de conexión de las tarjetas de entrada y salida. Para efectos de identificación, todas las señales que manejan las tarjetas se agrupan en conjuntos de 8, esto es, por *bytes* y se les asigna un número *byte* (0,...,n). Cada byte contiene a su vez 8 elementos (una señal por cada uno de sus 8 bits).

RECUERDE



Cualquier señal en las tarjetas queda definida mediante el número del grupo al que pertenece (número byte) y el número de elementos en el grupo (número bit). A ésta información se le conoce como **dirección de la señal**.

Codificación de señales

La acción de los relés puede ser codificada en el lenguaje de un PLC.

- 1 Direcciones:** son localizaciones de memoria en las cuales se pueden registrar instrucciones y datos.
- 2 Instrucciones:** sirven para decirle al PLC que hacer sobre la base de los datos que siguen a cada instrucción
- 3 Datos:** su codificación se compone de una cierta cantidad de cifras, y a partir de su subdivisión, será posible reconocer:
 - Las señales de entrada y salida.
 - Los relés auxiliares de las áreas de memoria.
 - Los temporizadores de los contadores.

En vista que cada PLC está caracterizado por un número máximo de entradas y de salidas (las entradas en número mayor que las salidas), y que existe la posibilidad de aumentar su número añadiendo *racks* de expansión, **la codificación (o bien el código de identificación), variará según sea el tipo de PLC.**

ACTIVIDAD 9.

Responda las siguientes preguntas.

1

Indique 5 actividades habituales relacionadas con el mantenimiento del PLC. Seleccione aquellas que Ud. considere más críticas.

1. _____
2. _____
3. _____
4. _____

2

Por favor complete el siguiente cuadro en base a lo que conoce de

| | Control centralizado | | Control distribuido |
|----------------------------|----------------------|--|---------------------|
| Cantidad de PLCs | | | |
| Que sucede si falla un PLC | | | |
| Redundancia | | | |

¡Felicitaciones!

Usted ha finalizado el capítulo 5.

A continuación se desarrollará el capítulo Programación del PLC.



Programación del PLC

TEMAS DEL CAPÍTULO 6

| | |
|--|----|
| 6.1 Lenguajes de Programación | 66 |
| 6.2 Designación de Entradas y Salidas | |
| 6.3 Caso de Estudio: Control de los movimientos de subida y bajada de un ascensor | |

El aumento de la complejidad en la programación de los autómatas programables requiere más que nunca de la estandarización de la misma.

En este capítulo se desarrollarán los lenguajes que han sido definidos.



6.1 Lenguajes de programación

Bajo la dirección del IEC el estándar IEC 1131-3 (IEC 65) ha sido definido para la programación del PLC. Alcanzó el estado de Estándar Internacional en Agosto de 1992. Los lenguajes gráficos y textuales definidos en el estándar son la base para la programación de PLCs.

Con la idea de hacer el estándar adecuado para un gran abanico de aplicaciones, cinco lenguajes han sido definidos en total:

- 1 Gráfico secuencial de funciones (grafcet)
- 2 Lista de instrucciones (LDI)
- 3 Texto estructurado
- 4 Diagrama de flujo
- 5 Diagrama de contactos (Ladder)

1 Gráfico secuencial de funciones (grafcet)

El gráfico secuencial de funciones (STL, SFC o Grafcet) es un lenguaje gráfico que **proporciona una representación en forma de diagrama de las secuencias del programa**. Soporta selecciones alternativas de secuencia y secuencias paralelas. Los elementos básicos son pasos y transiciones. Los pasos consisten en partes de programa que son inhibidas hasta que una condición especificada por las transiciones es conocida. Como consecuencia de que las aplicaciones industriales funcionan en forma de pasos, el SFC es la forma lógica de especificar y programar al más alto nivel un PLC.

2 Lista de instrucciones (LDI)

La lista de instrucciones (IL) es un lenguaje de bajo nivel, similar al lenguaje ensamblador. Con IL sólo una operación es permitida por línea (ej. Almacenar -store- cargar un valor en un registro). Este lenguaje es adecuado para pequeñas aplicaciones y para optimizar partes de una aplicación.

3 Texto estructurado

El texto estructurado (structured text o ST) es un **lenguaje de alto nivel, estructurado por bloques, que posee una sintaxis parecida al PASCAL**.

El ST puede ser empleado para realizar rápidamente sentencias complejas que manejen variables con un amplio rango tipos de datos, incluyendo valores analógicos y digitales. También especifica tipos de datos para el manejo de horas, fechas y temporizaciones, algo importante en procesos industriales. El lenguaje posee soporte para bucles repetitivos como: REPEAT UNTIL, ejecuciones condicionales empleando sentencias IF-THEN-ELSE-FOR-NEXT y funciones como SQRT (raíz cuadrada) y SIN (seno).

4 Diagrama de flujo

El diagrama de funciones (function block diagram o FBD) es un lenguaje gráfico que permite programar elementos que aparecen como bloques para ser cableados entre sí de forma análoga al esquema de un circuito.

FBD es adecuado para aplicaciones que involucren el flujo de información o datos entre componentes de control.

5 Diagrama de contactos

El diagrama de contactos (ladder diagram LD) es un lenguaje que utiliza un conjunto estandarizado de símbolos de programación. En el estándar IEC los símbolos han sido racionalizados (se ha reducido su número).

Organización de tareas

El estándar también define una nueva arquitectura para la organización e interacción de tareas con PLCs. **Una tarea controla la ejecución de un programa ejecutándolo periódicamente o en respuesta a un evento específico.** Para optimizar los recursos del controlador, una aplicación puede ser fragmentada en un número de pequeños programas concretos.

Cada programa tiene el control de una tarea que se ejecuta a la velocidad que requiera la E/S asociada.

Bloques de Funciones

Los bloques de funciones (FBs) son bloques estándar que ejecutan algoritmos como reguladores PID.

El estándar IEC asegura que los FBs son definidos empleando una metodología estándar.

Hay controles empleando parámetros externos, mientras que los algoritmos internos permanecen ocultos empleando Programación Orientada a Objetos.

ACTIVIDAD 10.

Marque la opción correcta.



¿Por qué se definieron cinco lenguajes?

Para tener un estándar para distintas aplicaciones.

☐
☐

Para utilizar el que mejor comprenda el programador.

¿De qué depende la velocidad de ejecución de tareas?

De los requerimientos de las entradas/salidas.

☐
☐

De la eficiencia en la programación.

6.2 Designación de Entradas y Salidas

Entradas y Salidas

EJEMPLO



Las siguientes son designaciones usuales de entradas y salidas:

X: para entrada Mitsubishi

I: para entrada Allen Bradley o Klockner

Y: para salida Mitsubishi

Q: para salida Klockner

O: para salida Allen Bradley

La línea Siemens identifica a las entradas como E E32.0, E32.1, etc. y las salidas como A siendo posible A 32.0, A 32.1, etc.

- Salvo excepciones y ampliaciones, los autómatas presentan 8 entradas normales (X, I, E) de un 1bit. Algunos con entradas especiales de 1bit, tienen la peculiaridad de funcionar como entradas digitales o como entrada de alarma y entrada rápida. La denominación depende de la marca del PLC, por lo tanto, se debe verificar en el manual del fabricante.
- Hay 6 salidas, típicamente empleando las letras (Y, Q, O, A), de 1bit cada una.

Marcas de memoria

También son denominadas como variables de memoria. Son de propósito general, es decir, podemos emplearlas en lo que deseemos. Se distinguen dos tipos de marcas de memoria:

- **Remanentes:** Estas marcas permanecerán en memoria aunque apaguemos el autómata. Hay diferentes denominaciones de memoria, según el fabricante, por lo tanto se debe verificar en el respectivo manual. En algunos casos son configurables por el usuario.
- **No Remanentes:** Estas marcas de memoria se borrarán en cuanto apaguemos el autómata.

Usualmente a las memorias se las designa con la letra M. Se organizan en forma básica como bit M0, M1, ... , M15 etc. en Melsec, Siemens y Klockner Moeller. Otra designación es B, es decir, B0, B1, ... en Allen Bradley y Telemecanique, etc.

Registros y Acumuladores

Todas las operaciones que hagamos con las entradas y las salidas se deben efectuar en algún sitio. Para ello, se definen:

- **Registro de estado:** Su tamaño es de 1 bit. Aquí es donde efectuaremos las instrucciones combinacionales, la carga de entradas y la asignación de salidas a nivel de bit.
- **Acumuladores (D, R, AKKU1 y AKKU2):** Sus tamaños son de 16 y 32 bits cada uno. Cada vez que carguemos un dato en los acumuladores se seguirá la siguiente secuencia:

Contenido de D2 → se pierde el contenido
Contenido de D1 → D2
DATO → D1

A su vez, cuando realicemos una operación entre registros (como suma o resta) el resultado se introducirá en el D1, perdiéndose el valor antes allí contenido.

Temporizadores y Contadores

Varían en función de marcas y modelos, pero los más usados suelen incorporar **32**

temporizadores: T0, ... , T31 **32 contadores:** C0, ... , C31 para Melsec y Allen Bradley. En Siemens se designan como T y Z.

Existen contadores que no se borran al desconectar el autómata (son remanentes), dichos contadores también deben verificarse en los respectivos manuales. Para consultar el estado de cada uno de ellos podremos usarlos como si fueran entradas (mediante operaciones combinacionales) o introduciendo su valor en los registros.

Constantes

A la hora de cargar datos en acumuladores, temporizadores, registros, etc. hay varias posibilidades en la forma de introducir el dato:

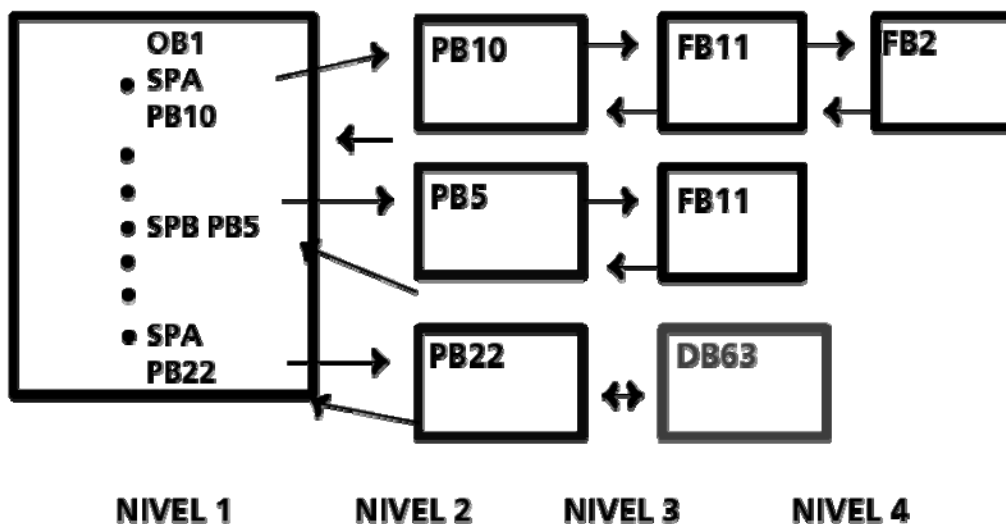
- KB: 8 bits (0 a 255 en decimal)
- KW 16 bit
- Como número decimal
- Como número hexadecimal

La modalidad de trabajar con constantes también depende del fabricante. Existen diversas prácticas para ello. En general, todos los casos admiten trabajar con valores de 8 bit (0--255), 16 bit (0-64535) y 32 bit.

Estructura del programa

Vamos a tener dos opciones para escribir el programa:

- **Lineal:** Se emplea un único módulo de programa (principal). Este módulo se procesa cíclicamente, es decir, tras la última instrucción se volverá a ejecutar la primera. Si la tarea a controlar es simple, ésta es la mejor forma.
- **Estructurada:** Para el caso de tareas complejas es más conveniente dividir el programa en módulos (sub programas). De esta forma logramos un programa más claro, con la posibilidad de poder llamar a un módulo desde distintas partes del programa (lo que evita repetir el código).



En la programación estructurada se comienza y termina en el módulo principal (llamados OB1 en Siemens) desde el cual saltaremos y retornaremos a los módulos que nos interesen. Se podrá saltar desde un módulo a otro (anidado), siempre que no superemos determinada cantidad de niveles de salto que permita como máximo un autómata.

Otras limitaciones son:

- El salto de un módulo a otro debe ser siempre hacia adelante (ej. se podrá saltar de PB1 a PB2 Siemens, pero no a la inversa).
- No se pueden dar dos saltos a un mismo módulo desde el módulo actual (ej. no se podrá saltar dos veces a PB3 desde PB2, pero sí se puede saltar a PB3 desde distintos módulos).
- Tanto en la programación lineal como en la estructurada, los módulos terminan con la instrucción BE en Siemens o con END.
- La memoria de un autómata generalmente está limitada a 2K bytes como mínimo. Cada instrucción ocupa generalmente 2 bytes, por los que se dispone 1000 líneas de programa aproximadamente.

Tipos de módulos



Generalmente existen cuatro tipos de módulos en cualquier autómatas programable Siemens:

- **Módulos Fuente de Poder:** este módulo incluye fuente de alimentación para todo el programador y sus módulos.
- **Módulos de Programa (CPU):** son los que incluyen el programa de usuario dividido, normalmente, según aspectos funcionales o tecnológicos.
- **Módulos de entradas:** son módulos de características especiales según tipo de entrada digital, analógica, lector de código barras, etc.
- **Módulos de salidas:** al igual que el anterior tienen características especiales según tipo de salida.

Las características de estos módulos son diferentes por su función en disposición de ubicación física y de hardware propio.

La línea Siemens denomina como módulos:

1 Módulos de organización (OB): son los que gestionan el programa de usuario. Numerados OB1, OB3, OB13 Y OB22.

OB1: Es el módulo del programa principal;

OB3: Es el que contiene el programa controlado por alarma;

OB13: Es el módulo para programas controlados por tiempo;

OB22: Es empleado por el sistema operativo.

2 Módulos de programa (PB): son los que incluyen el programa de usuario dividido, normalmente según aspectos funcionales o tecnológicos PB0 ... PB63.

3 Módulos funcionales (FB): son módulos de programa especiales. Aquí se introducen las partes de programa que aparecen con frecuencia o poseen gran complejidad. Poseen un juego de instrucciones ampliado. FB0 ... FB63.

- Módulos de datos (DB):** en ellos se almacenan datos para la ejecución del programa, como valores reales, textos, etc. Adoptan los valores: DB0 ... DB63. Los módulos DB1 y DB2 se emplean para definir las condiciones internas del autómata, por lo que no están disponibles.
- 256 palabras de datos: para emplear un módulo de datos es necesario activarlo previamente (como se verá más adelante).

La mayor ventaja que aporta el trabajo con módulos, es la facilidad para variar el proceso que controlan, ya que para ello basta cambiar el programa en el autómata en la mayoría de los casos. Otra ventaja es que el autómata también nos permite saber el estado del proceso, incluyendo la adquisición de datos para un posterior estudio.

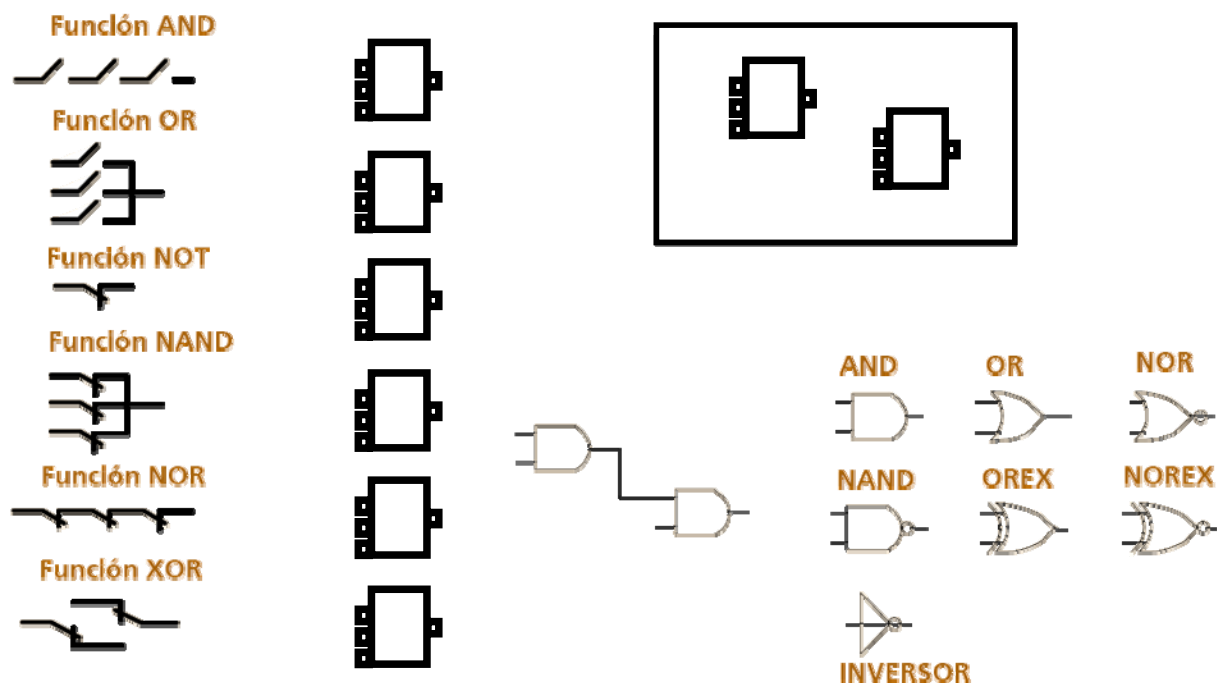
Funciones Generales

Las operaciones combinacionales más comunes se realizan con los bloques de funciones básicas, conexión serie, paralelo, negación, etc. Todas las funciones AND, OR, XOR, NAND Y NOR tienen tres entradas y una salida. Si deseamos realizar operaciones con más de tres entradas, se conectan varios bloques en cascada:

EJEMPLO














Ejemplo según normas DIN Y NEMA

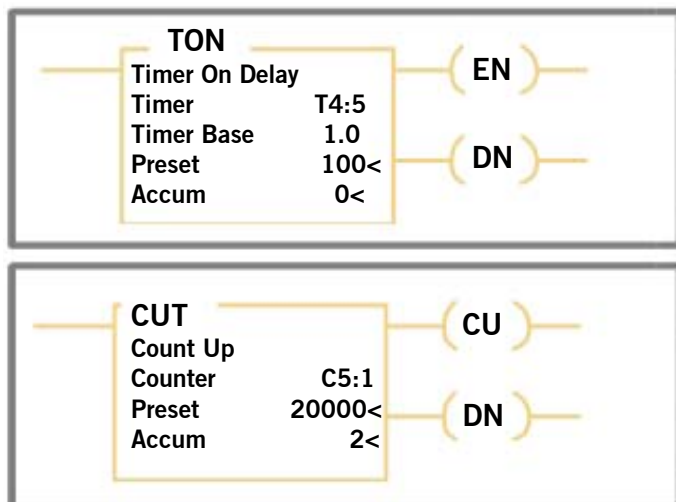


La función inversora NOT, tiene una entrada y una salida. La función OR exclusiva (XOR) posee dos entradas y una salida.

Funciones Especiales Din Siemens

| | | | |
|------------------------|---|---|---|
| TRA R A |  | Temporizador con retardo a la conexión con memoria. | Temporizador con retardo a la conexión. Activa la salida Q una vez que ha transcurrido el tiempo programado. |
| R CNT DIR PAR |  | Contador progresivo/regresivo | Temporizador con retardo a la desconexión. Desactiva la salida una vez transcurrido el tiempo programado. El temporizador se pone en marcha en flanco descendente. |
| R EN RAL PAR |  01 | Contador de horas de servicio. | Relé de impulsos. Tienen el mismo funcionamiento que un telerruptor. La salida cambia de estado, de 0 a 1, cada vez que cambia la señal en la entrada Trg. |
| TRA T |  01 | Relé de supresión. | Reloj. Permite controlar los instantes de activación y desactivación de la salida en un día de la semana y a una hora determinada. |
| FRE PAR |  01 | Conmutador de valor de umbral para frecuencia. | Relé de automantenimiento Función biestable R-S. Permite realizar la función paro-marcha típica de los automatismos a contactores. La situación no permitida R=1 S=1 se soluciona dando preferencia a R. |
| TRA T |  | Temporizador con retardo a la conexión. | Generador de pulsos. Genera pulsos de reloj a intervalos iguales. Funcionamiento similar a un intermitente. |
| TRA R T |  | Temporizador con retardo a la desconexión. | Temporizador De funcionamiento similar al temporizador a la conexión, pero con la característica que no es necesario mantener la señal en Trg. |
| TRA R (PAR) |  | Relé de Impulsos (Telerruptor) | Contador progresivo/regresivo. Permite contar y descontar los pulsos aplicados a su entrada CNT. |
| NO1 NO2 NO3 |  | Reloj Horario | Contador de horas de servicio. Permite medir el tiempo que está activada la entrada En. Esta función solamente se puede utilizar como bloque inicial. |
| S R (PAR) |  | Relé de automantenimiento. Biestable R-S. | Relé de supresión Activa la salida hasta que haya transcurrido el tiempo de T. Si éste no ha terminado y Trg se pone a 0 la salida también lo hace. Esta función solamente se puede utilizar como bloque inicial. |
| EN T |  | Generador de pulsos de reloj. | Conmutador de valor de umbral para frecuencias. Permite contar los impulsos aplicados a su entrada y dependiendo de éstos conmutar la salida. En el Logo! L con letras 24v, la entrada I12 está preparada para procesos de cálculos rápidos: función solamente se puede como bloque inicial. |

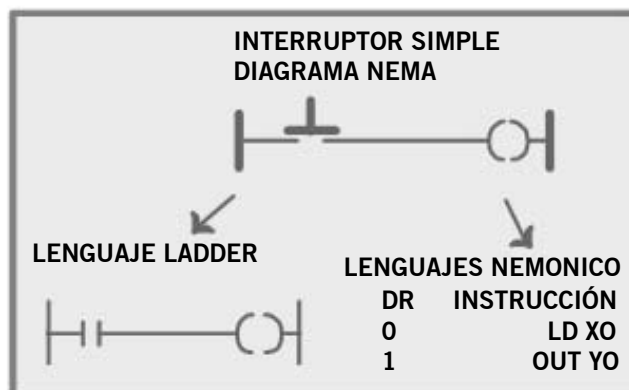
Temporizador con retardo a la conexión Allen Bradley



Activa la salida DN (Done) una vez que ha transcurrido el tiempo programado de 100 seg.

Se trata de un contador ascendente. Activa la salida DN una vez que se alcanza la cuenta 20000.

Los lenguajes de programación básicos más comunes son el Ladder y el Nemónico. En estos lenguajes, las instrucciones son equivalentes a los símbolos para contactos usados en los relés (lógica cableada) para el primero y similar a las definiciones del álgebra de Boole (lógica digital). En un diagrama, se hace a representación típica de una línea, que implemente una función de control, para una salida en lenguajes Ladder y Nemónico como se parecía en la figura.

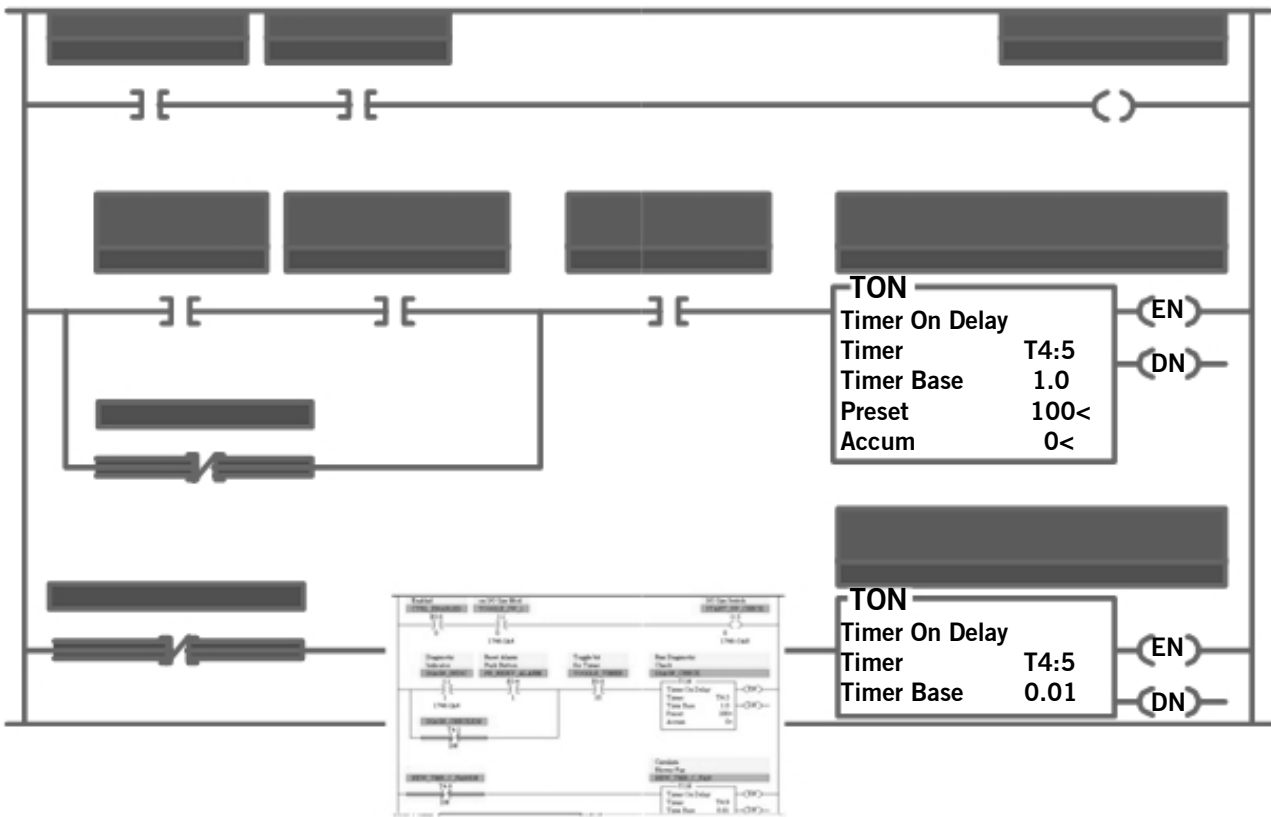


Los símbolos representados son muy parecidos al NEMA eléctrico, aún más son tan simples que emplean el símbolo de auxiliar abierto o cerrado para todo lo que se designe como "entrada" y para todo lo que es "salida" se emplea un círculo incompleto como se indica en la figura anterior.

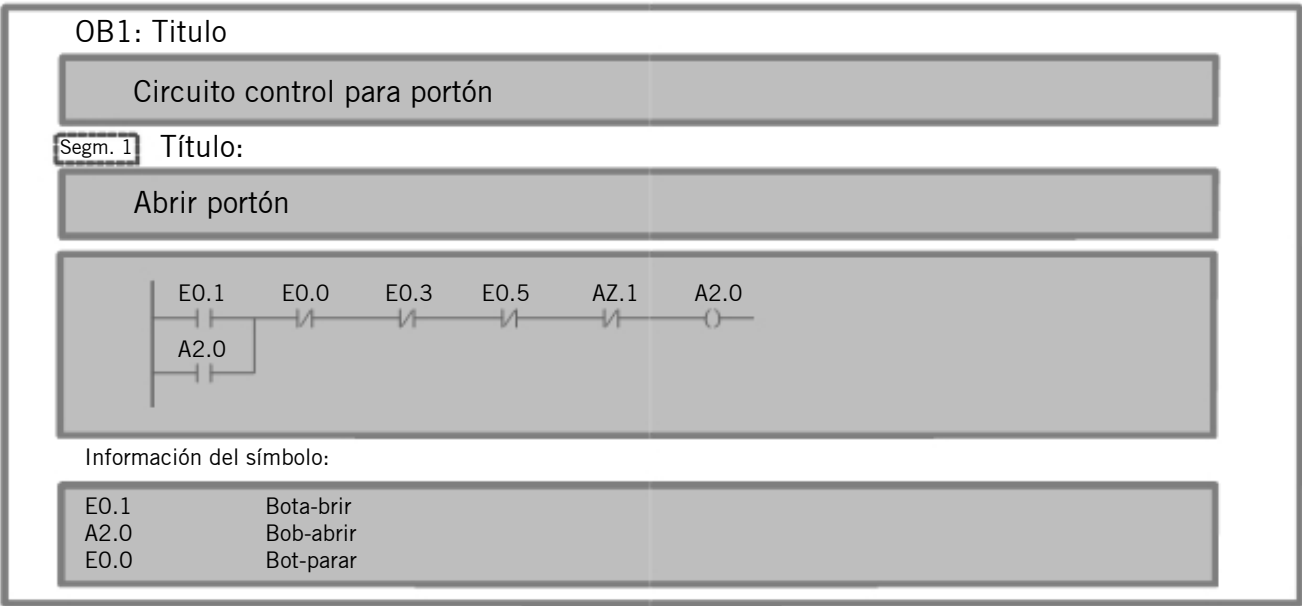
A pesar de la semejanza, hay diferencias:

1. Las salidas pueden ser del tipo interno y externo. Es decir, salida del tipo real (contactor, válvulas, etc.) o del tipo virtual (sólo en memoria).
2. El programa examina los contactos N.A. y los N.C. buscando encontrar un 1 o un 0 lógicos.
3. La salida en una línea de control tradicional se activa si cualquier paso o camino tiene todos sus contactos cerrados.
4. La salida de un control programado se activa si cumple la lógica booleana de la ecuación que ella representa.
5. Una salida interna se usa únicamente dentro del programa pues no tiene una existencia real.

Programa Ladder Allen Bradley



Programa Ladder Siemens



Lenguaje Nemónico o Lista de Instrucciones Siemens

| | | | | | |
|------------------------------|---|-----|--------------|---|--|
| OB1: Título: | | | | | |
| Circuito control para portón | | | | | |
| Segm. 1: Título: | | | | | |
| Abrir portón | | | | | |
| V(| | | | | |
| O | E | 0.1 | "bot-abrir" | — | |
| O | A | 2.0 | "bob-abrir" | — | |
|) | | | | | |
| UN | E | 0.0 | "bot-parar" | — | |
| UN | E | 0.3 | "lim-abrir" | — | |
| UN | E | 0.5 | "termico" | — | |
| UN | A | 2.1 | "bob-cerrar" | — | |
| = | A | 2.0 | "bob-abrir" | — | |
| — | | | | | |

Lenguaje de Programación Grafcet

Es el llamado Gráfico de Orden Etapa Transición. Ha sido especialmente diseñado para resolver problemas de automatismo secuenciales. Las acciones están asociadas a las etapas y las condiciones a cumplir en las transiciones.

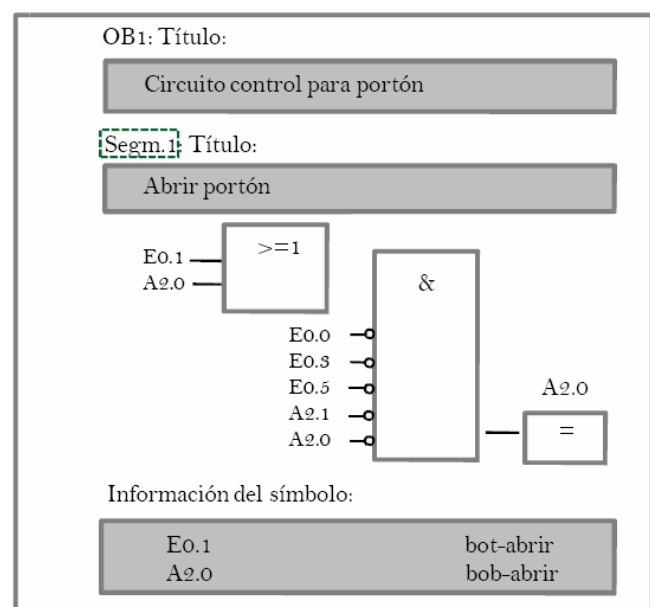
Muchos de los autómatas que existen en el mercado permiten la programación en GRAFCET, tanto en modo gráfico o como por lista de instrucciones.

IMPORTANTE

Este lenguaje resulta enormemente sencillo de interpretar por operarios sin conocimientos de automatismos eléctricos.

Plano de Funciones: FBD

El plano de funciones lógicas resulta especialmente cómodo de utilizar, a técnicos habituados a trabajar con circuitos de puertas lógicas, ya que la simbología usada en ambos es equivalente.

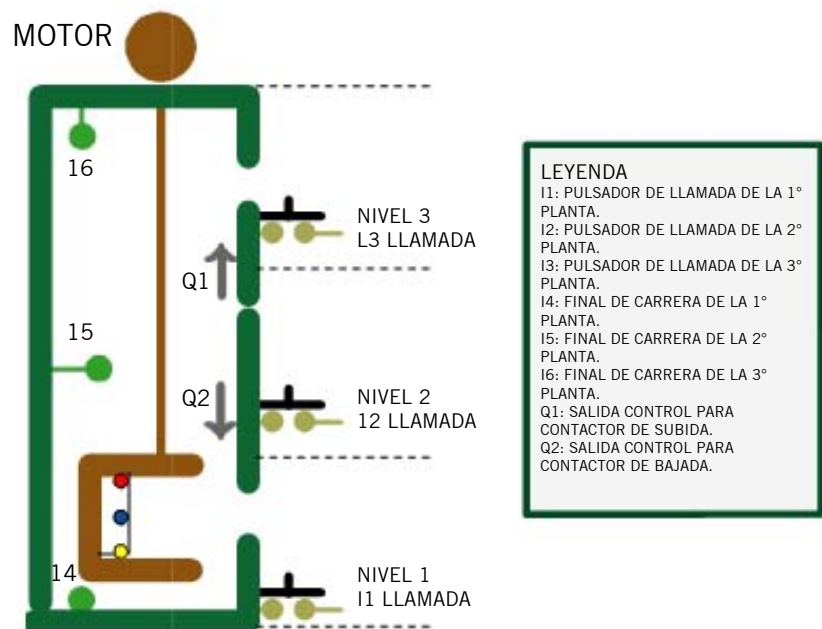


6.3 Caso de Estudio: Control de los movimientos de subida y bajada de un ascensor

Descripción

Cada planta tiene un pulsador de llamada que cuando es accionado, la cabina se posiciona en dicha planta.

Los pulsadores del interior de la cabina, son los mismos que los que se encuentran en el exterior, por lo tanto no necesitan programación, ya que se conectarán en paralelo de forma cableada.¹



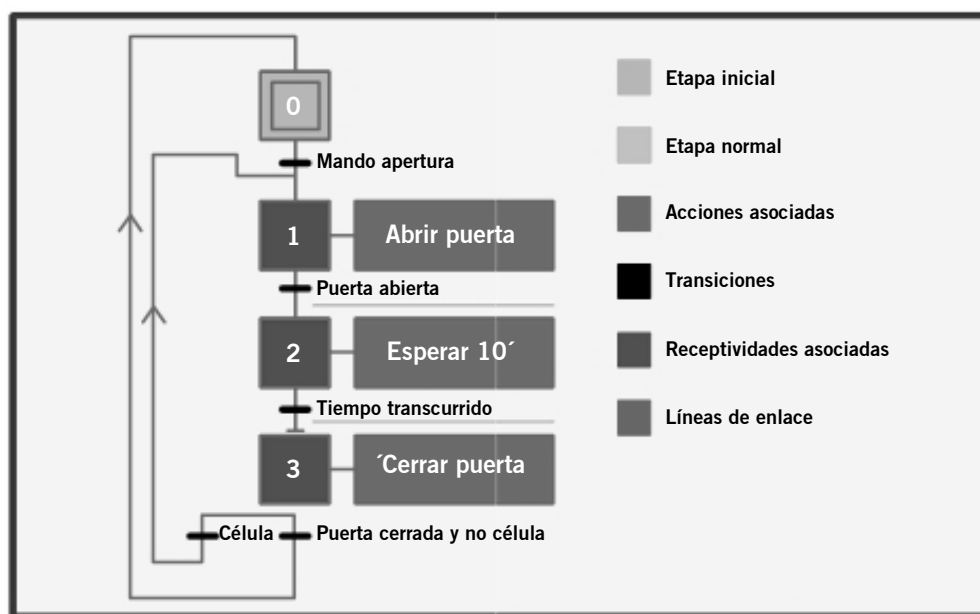
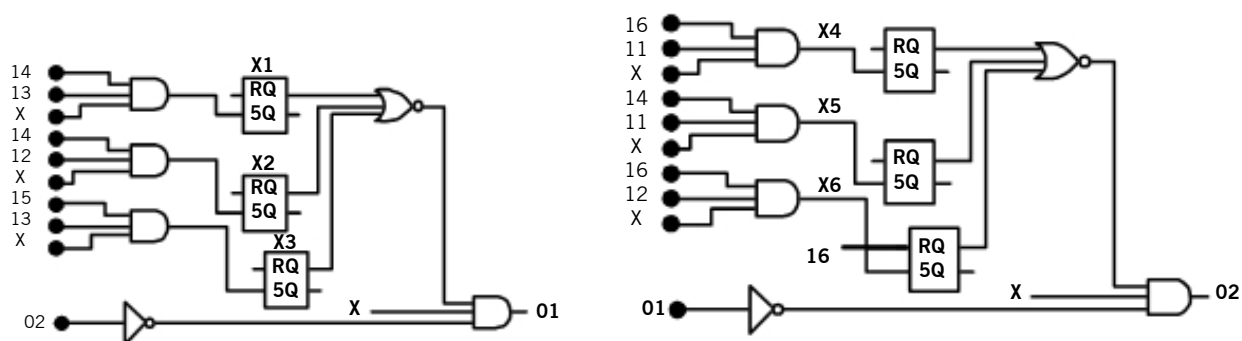
Movimientos

Cada uno de los movimientos está controlado por un biestable. En la entrada Set se establece la condición de funcionamiento y en el Reset la parada.

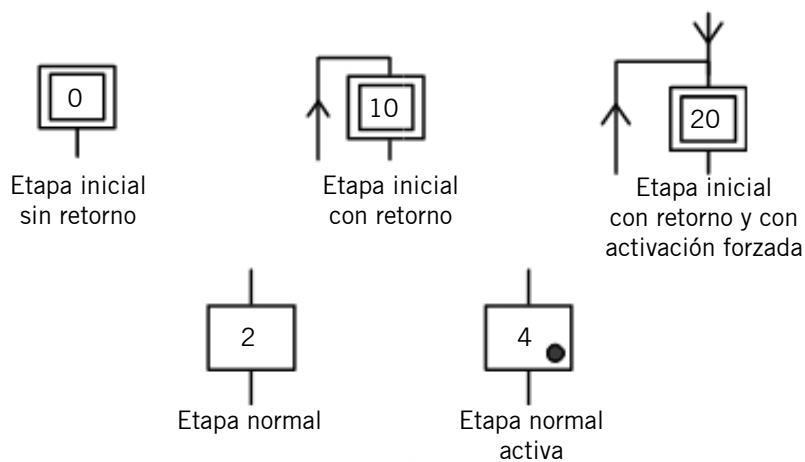
Para que el ascensor suba desde la primera planta a la tercera, movimiento X1, será necesario que el final de la carrera I4 esté accionado y se pulse I3 ($S = I4 * I3$). Cuando la cabina llega arriba, el pulsador I6 es accionado deteniendo el movimiento.

Todos los movimientos de subida (X1, X2 y X3) activarán Q1 y todos los movimientos de bajada (X3, X4 y X5) activarán Q2.

En las ecuaciones de las salidas, se realizará el producto negado de la variable de salida contraria, para evitar cortocircuitos en el circuito de fuerza. Hay que tener en cuenta, que se accionará un motor trifásico a 220v o bien a 380v



Etapas iniciales

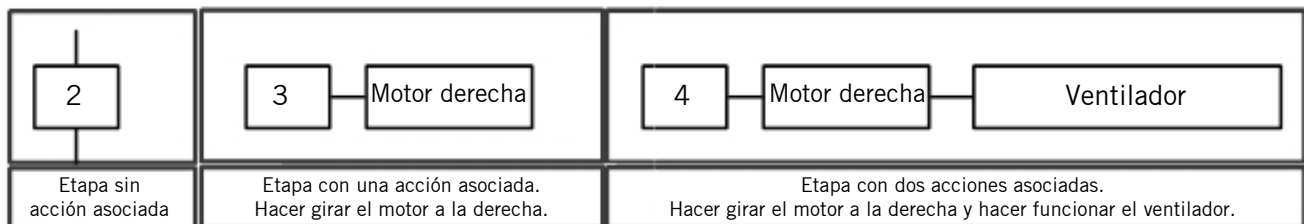


- Una etapa inicial se representa con un doble cuadrado.
- Las etapas iniciales de un sistema de activan al iniciar el GRAFCET.
- Una vez que se han iniciado, las etapas iniciales tienen el mismo tratamiento que las otras etapas.
- Un sistema debe tener como mínimo una etapa inicial.

Etapas Normales

- 1 Las etapas normales representan los estados estables del sistema.
Las etapas del GRAFCET se representan mediante un cuadrado numerado.
- 2 Las etapas deben estar numeradas; aunque no necesariamente de forma correlativa.
No puede haber dos etapas con el mismo número.
- 3 Las etapas pueden estar activas o inactivas. Al representar el estado del GRAFCET en uno momento dado, se puede indicar que una etapa está activa, con un punto de color (etapa 4).
- 4 En las etapas, puede o no haber acciones asociadas.

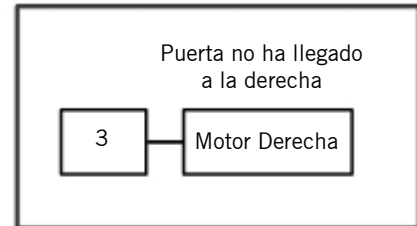
Etapas Asociadas



- 1 Una etapa sin ninguna acción asociada (etapa 2) puede servir para detener una acción mono estable que se realizaba en la etapa anterior, o como etapa de espera.
- 2 Una acción asociada (etapa 3) nos indica que al estar activa la etapa, el motor girará a la derecha.
- 3 En una etapa, puede haber múltiples acciones asociadas (etapa 4). Al estar la etapa 4 activa, el motor girará a la derecha, y al mismo tiempo el ventilador estará funcionando.
- 4 Si en un sistema en un momento concreto sólo hay una etapa activa, entonces, sólo estarán funcionando los elementos activados por las acciones asociadas a esa etapa (a no ser que en otra etapa se haya activado de forma biestable otra acción).

Acciones Asociadas

La acción a realizar en una o más de las acciones asociada a una etapa, puede estar condicionada a una función booleana adicional. En este caso, el motor girará a la derecha mientras esté activa la etapa 3 y además la puerta no haya llegado ya a la derecha. En el rectángulo donde se representa la acción asociada, hay una entrada para las condiciones.



La **norma IEC-848** propone las representaciones siguientes para las acciones asociadas condicionadas:

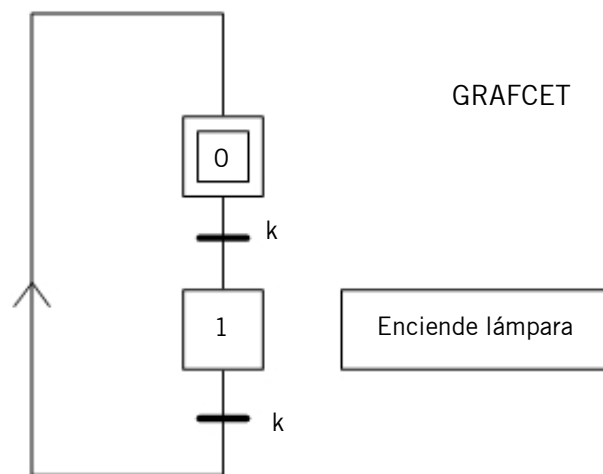
| | | | |
|----------|-------------------------------------|--|--|
| C | Acción condicionada | | <p>Supongamos un sistema en que tenemos un cuadro electrónico, para la regulación de unas máquinas. Si está activa la etapa de espera 2, y el termostato, el ventilador se pondrá en marcha. Esta condición, la podemos representar dentro del recuadro de la acción, o bien fuera.</p> |
| D | Acción retardada | | <p>El motor A se pondrá en marcha 5 segundos después de activar la etapa 10; si la transición r se activa antes de ese tiempo el motor no llegará a ponerse en marcha.</p> |
| L | Acción limitada en el tiempo | | <p>La bomba se pondrá en funcionamiento 10" después de haberse activado la etapa 11, pasado este tiempo aún que no se active la transición s, la bomba dejará de funcionar.</p> |
| P | Acción de impulso | | <p>Al activarse la etapa 12, se activará la electro válvula K con un impulso de señal..</p> |
| S | Acción memorizada | | <p>Cuando se active la etapa 13, el motor A se pondrá en marcha de forma biestable (set), y al salir de la etapa, continuará funcionando hasta que se haga un reset a la acción.</p> <p>Al activarse la etapa 14, el motor A se pondrá en marcha de forma biestable (set), y al salir de la etapa continuará funcionando hasta que se haga un reset a la acción.</p> |

Transiciones

Las transiciones representan las condiciones que el sistema debe superar para poder pasar de una etapa a la siguiente.

Al pasar una transición, el sistema deja de estar en una etapa e inmediatamente va a la siguiente.

Validar la transición implica un cambio en las etapas activas del GRAFCET.



- Las transiciones se representan con un pequeño segmento horizontal que corta la línea de enlace entre dos etapas.
- Son etapas de entrada a una transición, todas las etapas que conducen a una transición.
- Son etapas de salida a una transición, las etapas que salen de una transición.

Receptividades asociadas a las transiciones

Las condiciones que se deben superar para poder pasar una transición, reciben el nombre de receptividades.

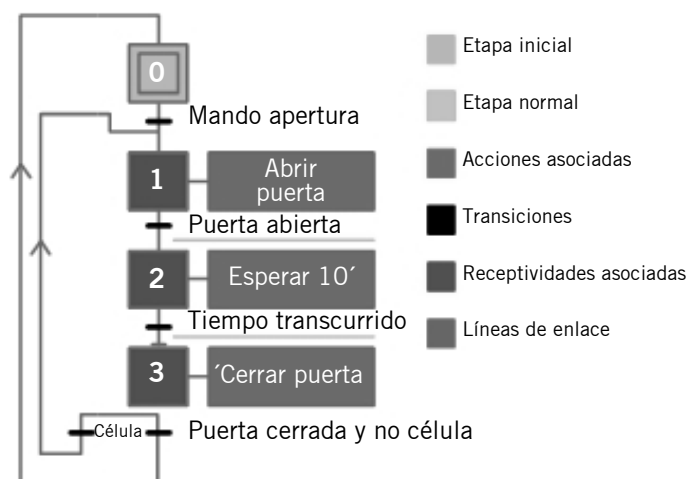
En una transición podemos tener:

- Una condición simple $[P_m]$
- Una función booleana $[(P_m + P_k) * \overline{P_p}]$
- La señal de un temporizador o contador $[T03]$. En este caso es habitual que el temporizador se haya activado a contar de la acción asociada de la etapa de entrada.

La activación de otra etapa del GRAFCET $[X12]$, donde X nos indica que la receptividad está condicionada al hecho que la etapa (en este caso la 12) esté activa.

Líneas de Enlace

Las líneas de enlace son líneas verticales u horizontales, que unen con una dirección significativa (a no ser que se indique lo contrario de arriba abajo), las distintas etapas con las transiciones, y las transiciones con las etapas.



ACTIVIDAD 10. Programación del PLC

Se dieron a conocer características principales de los códigos de programación de un PLC.



Por favor, indique cuáles de las siguientes afirmaciones son verdaderas y cuáles son falsas.

| | | VERDADERO | FALSO |
|---|--|--------------------------|--------------------------|
| 1 | La denominación de las entradas del PLC son universales, lo que no generará confusiones. | <input type="checkbox"/> | <input type="checkbox"/> |
| 2 | En los registros y acumuladores se realizan las operaciones intermedias entre entradas y salidas. | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | Cuando la tarea a controlar es simple se recomienda utilizar una estructura lineal del programa. | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | No existe una equivalencia entre los lenguajes más convencionales de programación y las instrucciones de los contactos de los relés. | <input type="checkbox"/> | <input type="checkbox"/> |
| 5 | Los movimientos se controlan con un biestable y las parada de estos se realiza mediante un Reset. | <input type="checkbox"/> | <input type="checkbox"/> |
| 6 | El PLC no admite etapas en las cuales no se realice ningún tipo de acción asociada con el control del proceso. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7 | El sistema seguirá funcionando a pesar que una condición de transición no se cumpla, avisando este inconveniente al usuario. | <input type="checkbox"/> | <input type="checkbox"/> |

¡Felicitaciones!

Usted ha finalizado el capítulo 6.
A continuación se desarrollará el capítulo
Instalación, Puesta a Punto y mantenimiento del PLC



Instalación, Puesta a Punto y Mantenimiento del PLC

TEMAS DEL CAPÍTULO 6

| | |
|----------------------------------|----|
| 7.1 Instalación y Puesta a Punto | 85 |
| 7.2 Mantenimiento | 88 |

Es necesario conocer los modos de instalación, la disposición de los distintos componentes del PLC y el mantenimiento necesario para el buen funcionamiento del sistema.



7.1 Instalación y puesta a punto

Condiciones ambientales del entorno

Normalmente y salvo indicación expresa del fabricante, el entorno donde se sitúe el PLC habrá de reunir las condiciones físicas siguientes:

- Ausencia de vibraciones, golpes, etc.
- No exposición directa a los rayos solares o focos caloríficos intensos, así como a temperaturas que sobrepasen los 50-60 °C (122-140 °F).
- No elegir lugares donde la temperatura descienda en algún momento por debajo de 5 °C (41 °F) o donde los bruscos cambios puedan dar origen a condensaciones.
- Tampoco es posible situarlos en ambientes en donde la humedad relativa se encuentre aproximadamente por debajo del 2.0% o por encima del 90%.
- Ausencia de polvos y ambientes salinos.
- Ausencia de gases corrosivos.
- Por seguridad es necesario un ambiente exento de gases.



Distribución a componentes

Es norma que el PLC se sitúe en un gabinete metálico. Antes de elegir el mismo se ha de conocer si éste necesita ventilador incorporado para forzar la ventilación del aire, en caso que la temperatura ambiente supere la especificada por el fabricante.

En cuanto a su distribución, se tendrán en cuenta las siguientes consideraciones:

Elementos disipadores de calor

Se situarán en la parte superior del gabinete, principalmente el PLC y las fuentes de alimentación.

Elementos electromecánicos

Son generadores de campos magnéticos debido a sus bobinas, por lo que es recomendable alejarlos lo más posible de la CPU y las E/S. De igual modo los transformadores deben estar lo más alejados posibles de cualquier parte del PLC.

Para poder realizar posteriormente un buen cableado, se agruparán separadamente los módulos de entrada de los de salida; las E/S digitales de las analógicas y en el resto de los elementos, los de C.D de los de C.A.

Algunos fabricantes indican que su PLC puede situarse en distintas posiciones, pero en general, éste se sitúa verticalmente sobre carril DIN o placa perforada.

Condiciones ambientales del entorno

Para un correcto cableado hay que tener en cuenta unas reglas mínimas, entre las que se encuentran:

- Separar los cables que conducen C.D. de los de C.A. para evitar interferencias.
- Separar los cables de las entradas de las de salidas.
- Si es posible, separar los conductores de las E/S analógicas de las digitales.
- Los cables de potencia que alimentan los contactores, fuentes de alimentación, etc., deben ir por canaleta distinta de los cables de E/S.

Los cables de alimentación y los de E/S se conducirán por tubo o canaleta, siendo recomendable entre ambos grupos de cables haya una **distancia mínima de 30 cm, si van paralelos** que esto no sea posible, se situarán placas metálicas conectadas a tierra que separen los distintos tipos de cables dentro de la canaleta.

Alimentación

La alimentación del PLC es otro factor importante a tener en cuenta. Hay cuatro condiciones necesarias:

- Un voltaje estable de valor indicado por el fabricante y exenta en lo posible de picos provocados por otros aparatos de instalación.
- Protección contra sobrecargas y cortocircuitos por medio de interruptores fusibles, etc. así como contra derivaciones a tierra por medio de interruptores diferenciales.
- Una tierra del valor adecuado y debidamente señalizada mediante conductor amarillo/verde.
- Un circuito de mando que nos permita conectar y desconectar en el momento preciso el circuito o parte del mismo.

Puesta a punto y en servicio

Se entiende por puesta a punto la supervisión total del sistema y la realización de todas aquellas tareas que sean necesarias para dejarlo en condiciones perfectas de poder iniciar su funcionamiento.

Esta supervisión es conveniente dividirla en dos partes:

Sin voltaje

Verificación de las partes físicas, tiene por objeto comprobar entre otros:

- La conexión correcta de todos los componentes del sistema, incluidas la alimentación, de acuerdo con los esquemas correspondientes.
- La sujeción firme de todos sus cables a sus regletas, a la CPU, E/S, fuente de alimentación, etc.
- La identificación exacta de cables mediante identificadores con letras y/o números.
- La correcta conexión del cable amarillo-verde de tierra también ha de ser comprobada.

Con voltaje

Verificación del sistema automático, se realiza de la siguiente forma:

- Con el PLC en modo STOP, alimentar el sistema pero no las cargas.
- Comprobar la no indicación de error de los leds correspondientes a la CPU.
- Comprobar el correcto funcionamiento del circuito de mando de marcha en las entradas y salidas como en la marcha y paro general.
- Con el PLC en modo RUN, verificar que las salidas responden de acuerdo al programa al actuar manualmente sobre las entradas. Esto es posible visualizarlo bien mediante los leds indicativos de salida activada o por medio de la unidad de programación.

ACTIVIDAD 11.



Indique cuáles de las siguientes afirmaciones son verdaderas y cuáles falsas.

No es necesario especificar la resistencia del PLC a vibraciones, golpes, gases corrosivos y otras características del ambiente que podrían dañarlo.

☐
☐

El PLC y las fuentes de alimentación deben colocarse en la parte superior del gabinete por ser disipadores de calor.

☐
☐

Los módulos de entrada y salida se colocan juntos para poder un buen cableado.

☐
☐

Todos los cables de potencia pueden ir por la misma canaleta sin importar cual es el destino de la alimentación en el circuito.

☐
☐

Se necesita de un circuito de mando que permita conectar y desconectar el circuito o un aparte de éste cuando sea necesario.

☐
☐

La puesta a punto implica todas Las tareas necesarias para que el circuito esté en perfectas condiciones para comenzar a funcionar.

☐
☐

7.2 Mantenimiento

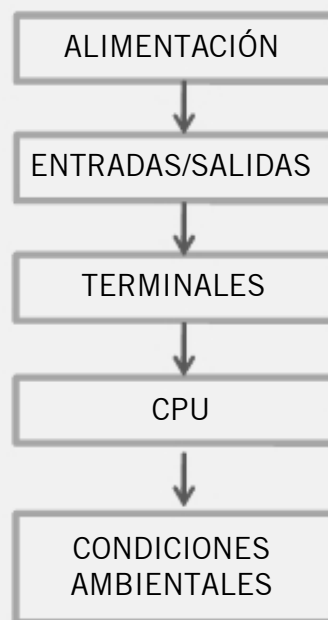
Como cualquier otra máquina, **el PLC necesita de un mantenimiento preventivo o inspección periódica**; ésta inspección ha de tener una periodicidad tanto más corta cuanto más complejo sea el sistema, y puede variar desde semanalmente hasta anualmente.

Mantenimiento correctivo

La detección de averías imputables al PLC se determina generalmente por los procedimientos que el fabricante ha desarrollado e incluido en el mismo, y son:

- Por la lista de mensajes de error correspondientes a los leds indicadores que se encuentran en el frente de la CPU.
- Por las indicaciones que aparecen en el display de la consola de programación.

En general el organigrama lógico que se debe seguir, para la detección y reparación de una avería sería el de la figura:



¡Felicitaciones!

Usted ha finalizado el capítulo 7.
A continuación se desarrollará el capítulo
Interfases de Comunicación

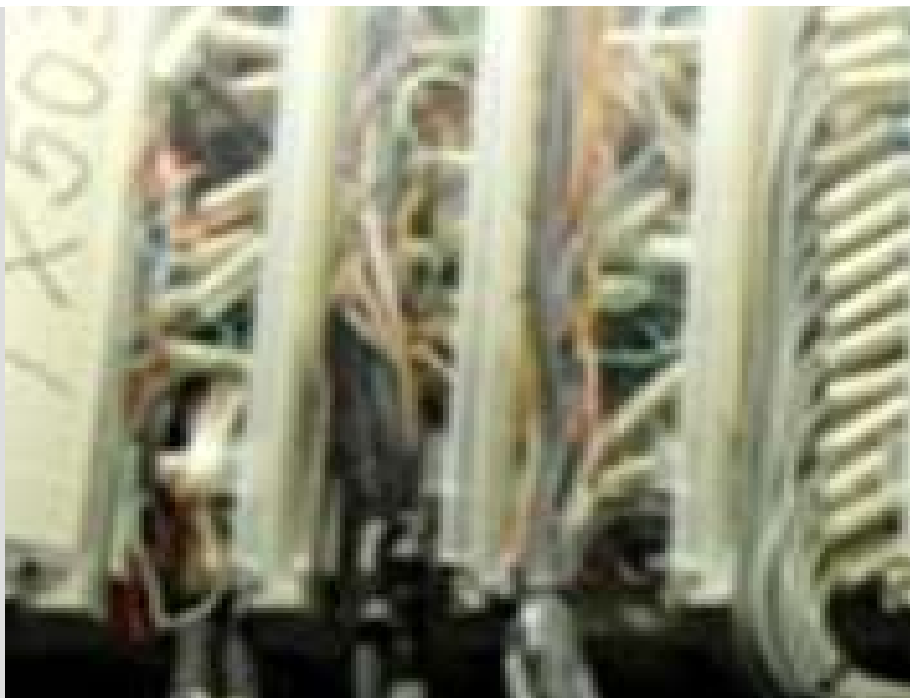


Interfases de comunicación

TEMAS DEL CAPÍTULO 6

| | |
|--------------------------------|----|
| 8.1 Estándares de comunicación | 90 |
| 8.2 Comunicación serie | 91 |
| 8.3 Medios de comunicación | 98 |

Las interfases de comunicación están diseñadas para permitir que el PLC y otros dispositivos inteligentes se comuniquen y transfieran datos en una red de área local (LAN).



8.1 Estándares de Comunicación

Normalmente, los PLCs que pueden comunicarse entre sí en una red están restringidos a productos diseñados por el fabricante de la red. Los demás dispositivos se pueden conectar dependiendo de la interfase de la red.

En general, cuando un procesador u otro dispositivo envía un mensaje, su interfase de red retransmite el mensaje a determinada cantidad de bits por segundo (baud rate). La interfase receptora acepta la transmisión y la envía al dispositivo deseado.

El protocolo para el enlace de comunicación varía de acuerdo a la red, pero todos los dispositivos en la red deben utilizar el mismo protocolo para poder comunicarse.

Un módulo de interfase de red permite conectar un gran número de controladores y además de otros dispositivos. Para ello, debe usar un cable coaxial.



Estándares de comunicación

La IEEE (Institute of Electrical and Electronic Engineers– Instituto de Ingenieros en Eléctrica y Electrónica) y la EIA (Electronic Industries Association – Asociación de Industrias Electrónicas) han definido estándares de comunicación.

Algunos ejemplos de estándares de comunicación son el IEEE— 488, el EIA RS-232C y el RS-422. Otros estándares se refieren a métodos de interfase que han tenido aceptación pero que no tienen una definición oficial.



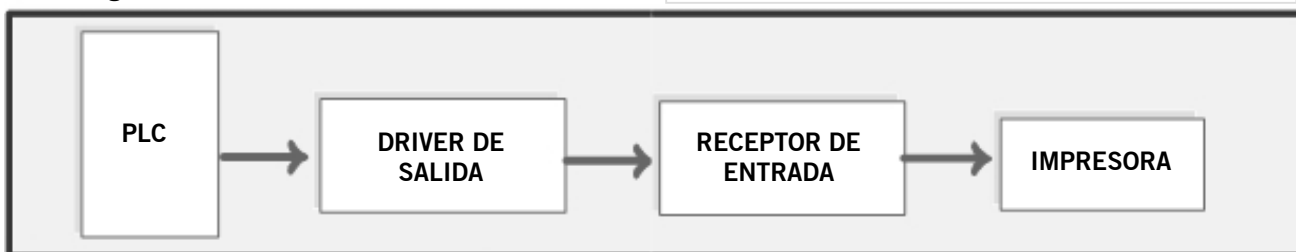
8.2 Comunicación en Serie

Los estándares más comunes para comunicación serie son el RS-232C y el RS-422. Los enlaces de comunicación con equipos periféricos pueden ser unidireccionales y bidireccionales.

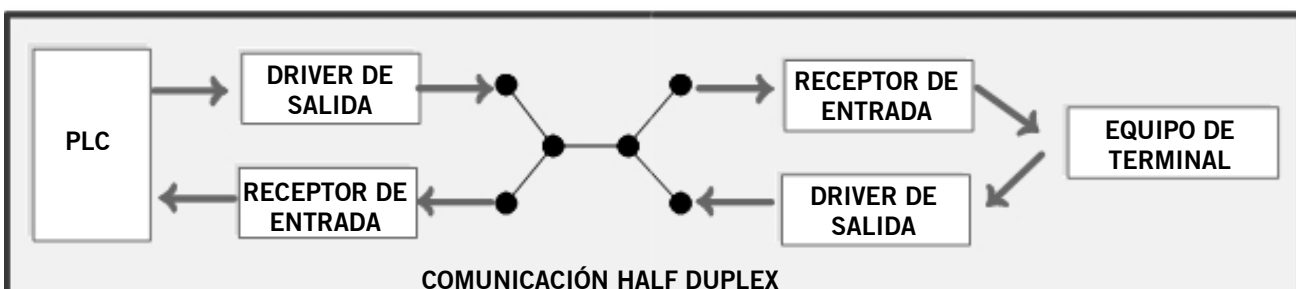
Si el equipo periférico es estrictamente una entrada o una salida, **sólo es necesario enviar datos en una sola dirección, es decir, unidireccional**. En este caso sólo se requiere una línea para completar el enlace como se muestra en la figura:

IMPORTANTE

La comunicación del PLC se hace en serie, con una rapidez de 110 a 19200 bits/seg, con paridad o sin paridad y usando diferentes estándares de interfases de comunicación. Esta comunicación utiliza pares de cable torcidos y es la más común para el equipo periférico como impresoras, terminales y módems

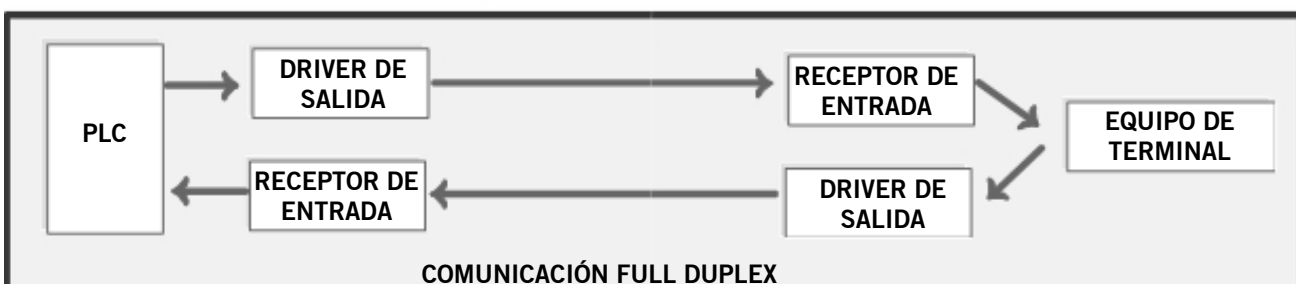


Los dispositivos que sirven como entradas y salidas requieren de un enlace bidireccional. Existen dos formas de lograr la comunicación en ambas direcciones:



Half Duplex

Una línea de comunicación se comparte. Los datos se pueden enviar en ambas direcciones, pero sólo en una dirección por vez.



Full Duplex

Cuando se requiere comunicación bidireccional simultánea se utilizan dos líneas de comunicación o full duplex. Una línea se asigna permanentemente como salida y la otra como entrada.

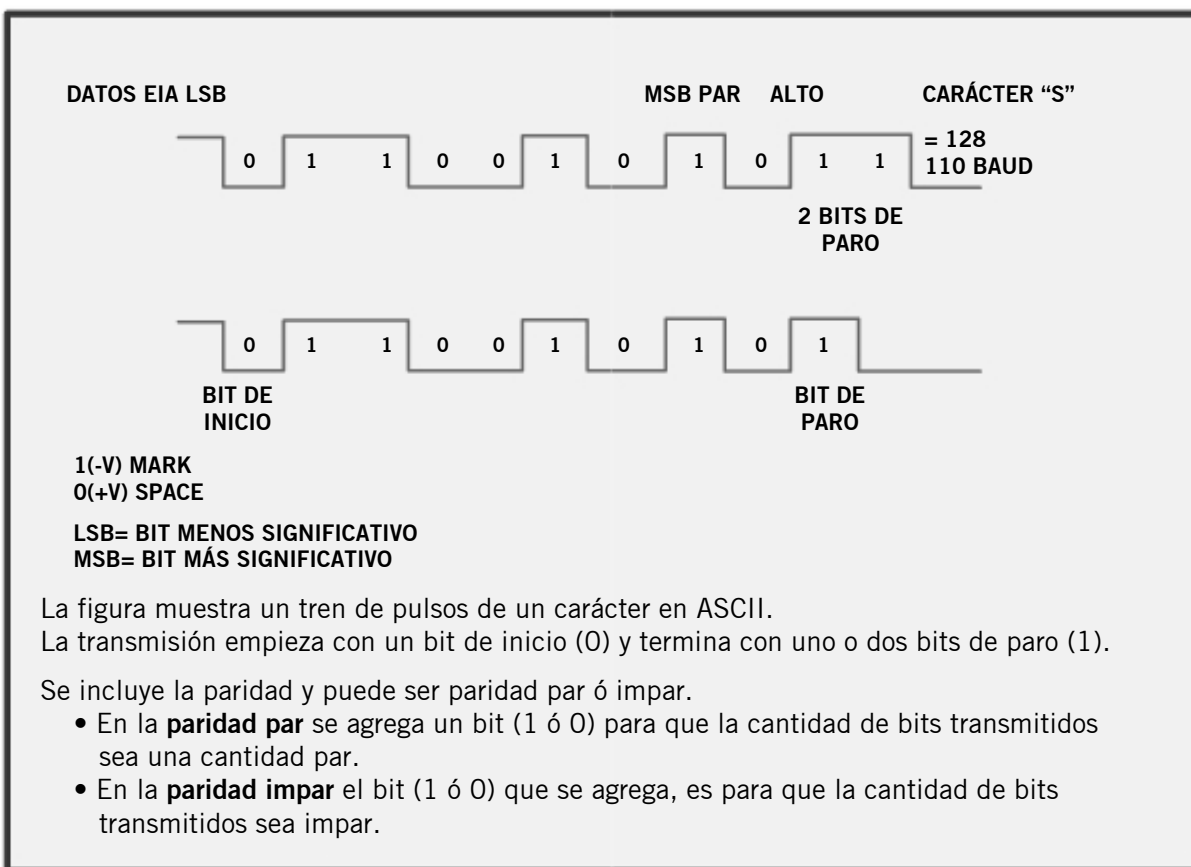
Estándar EIA RS-232C

El EIA RS-232C es un estándar que **define la interfase entre el equipo de datos y de comunicación**, empleando intercambio de datos binarios en serie. El estándar define las características mecánicas y eléctricas de la interfase.

La interfase RS-232C completa consiste de 25 líneas de datos. Aún cuando varias de estas líneas son especializadas y pocas se dejan indefinidas, la mayoría de los periféricos requieren de 3 a 5 líneas para una operación apropiada.

Algunas de las **especificaciones eléctricas** del estándar RS - 232C son:

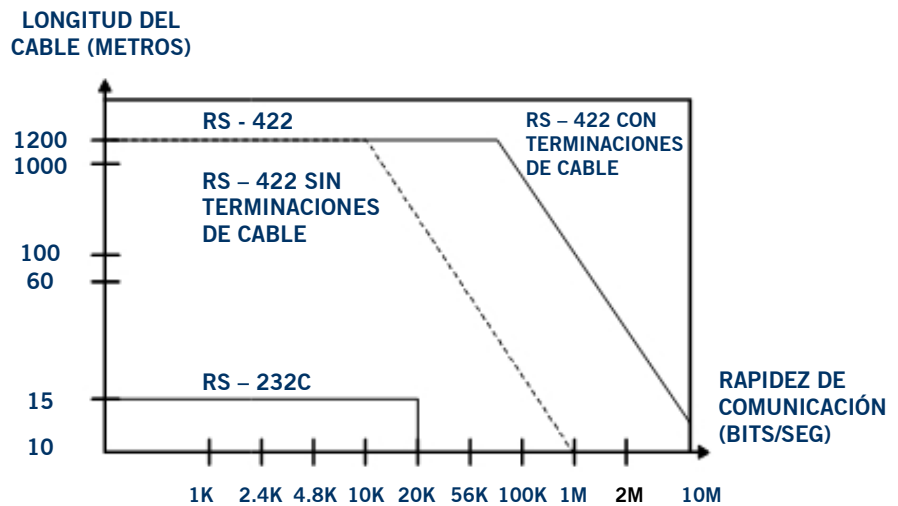
- Los voltajes de la señal en la interfase son mínimo de +5V y máximo de +15V para un 0 lógico, y un máximo de -5V y mínimo de -15V para un 1 lógico.
- La distancia máxima recomendada es de 15 m.
- Los voltajes menores a -3V se llaman *mark* o condición de señal, los voltajes sobre +3V se llaman *space* o condición de no señal. Las señales entre -3V y +3V no están definidas.



Estándar EIA RS-422

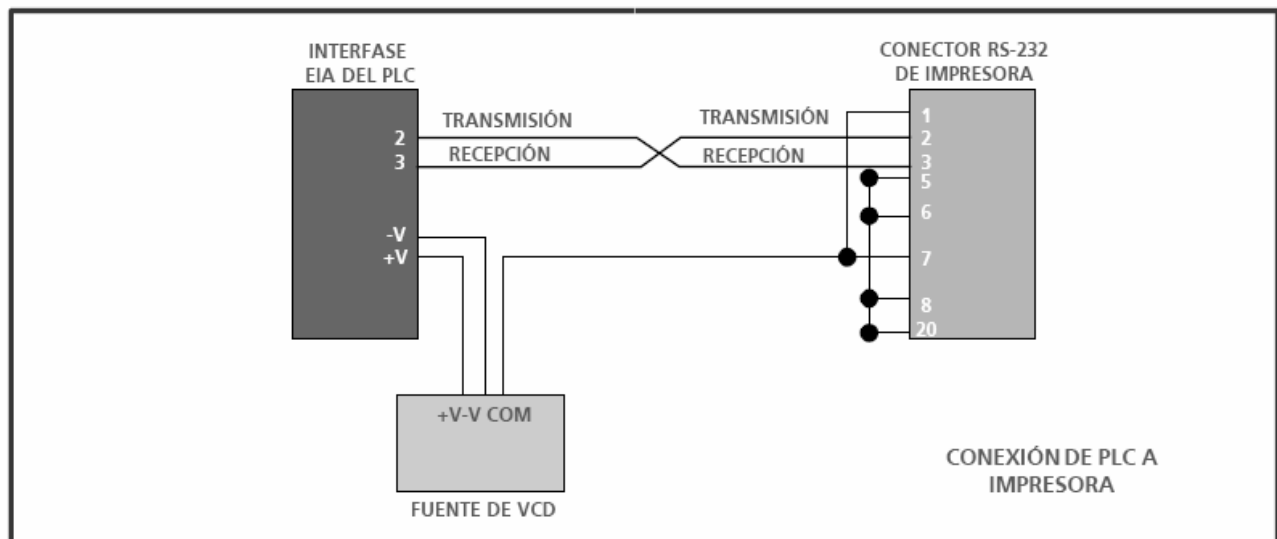
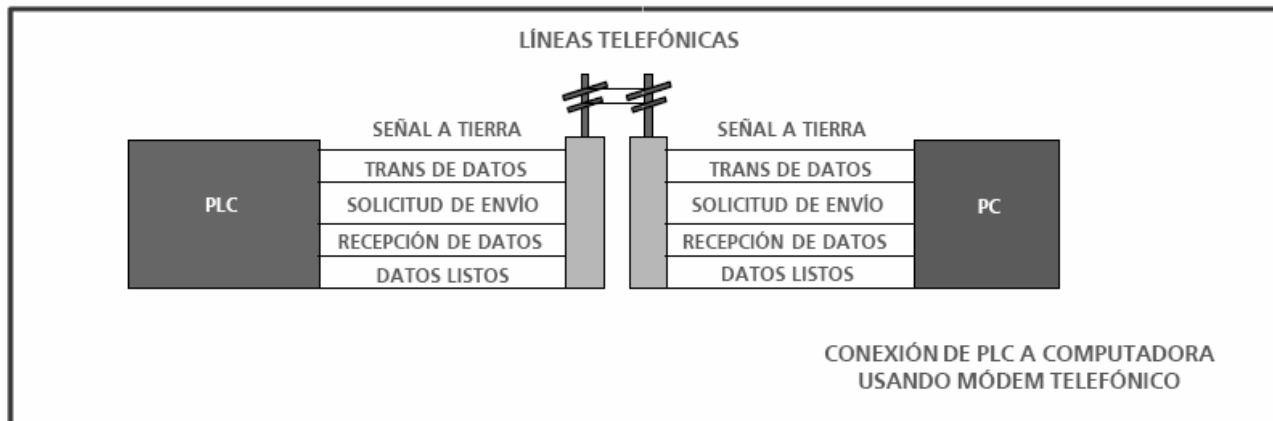
El estándar RS-422 **se diseñó para evitar las limitaciones del RS-232C**, como la rapidez máxima de transmisión de 20 Kbauds y la distancia máxima de 15m.

Con la interfase RS-422 se puede trabajar hasta 10 Mbauds y a distancias de hasta 1200m, aunque ambas condiciones no se pueden lograr al mismo tiempo, como se puede ver en la figura.



Conexión entre el PLC y la Computadora

En las figuras se muestra la conexión del PLC con diferentes dispositivos.



Topologías

Una red de área local o LAN (Local Area Network), es una red de distancia media y alta velocidad. La distancia máxima entre dos nodos de la red es de una milla, soporta al menos 100 estaciones y la rapidez de transmisión es entre 56 Kbauds y 10 Mbauds.

Una red industrial debe ser capaz de soportar control en tiempo real, tener alta integridad de datos (detección de error), alta inmunidad al ruido, confiabilidad en ambiente hostil y aplicable a grandes instalaciones.

Las redes de área local permiten que gran cantidad de datos se puedan intercambiar entre PLC's y otros dispositivos de una manera eficiente a través de un enlace de comunicación dedicado.

Las aplicaciones más comunes de redes en los PLCs son la **adquisición de datos y el control distribuido**.

Si se desea tener **adquisición de gran cantidad de datos** y procesarlos en un PLC se complica el programa de control, se usa mucha memoria y el tiempo de ciclo se incrementa. Las desventajas anteriores se pueden eliminar si se transfieren los datos, a través de una red a una computadora que los almacene y procese.

En las aplicaciones de **control distribuido**, las funciones de control no se concentran en un PLC sino que se distribuyen entre varios, eliminando la desventaja de depender de un solo PLC y mejorando la confiabilidad y el rendimiento del sistema.

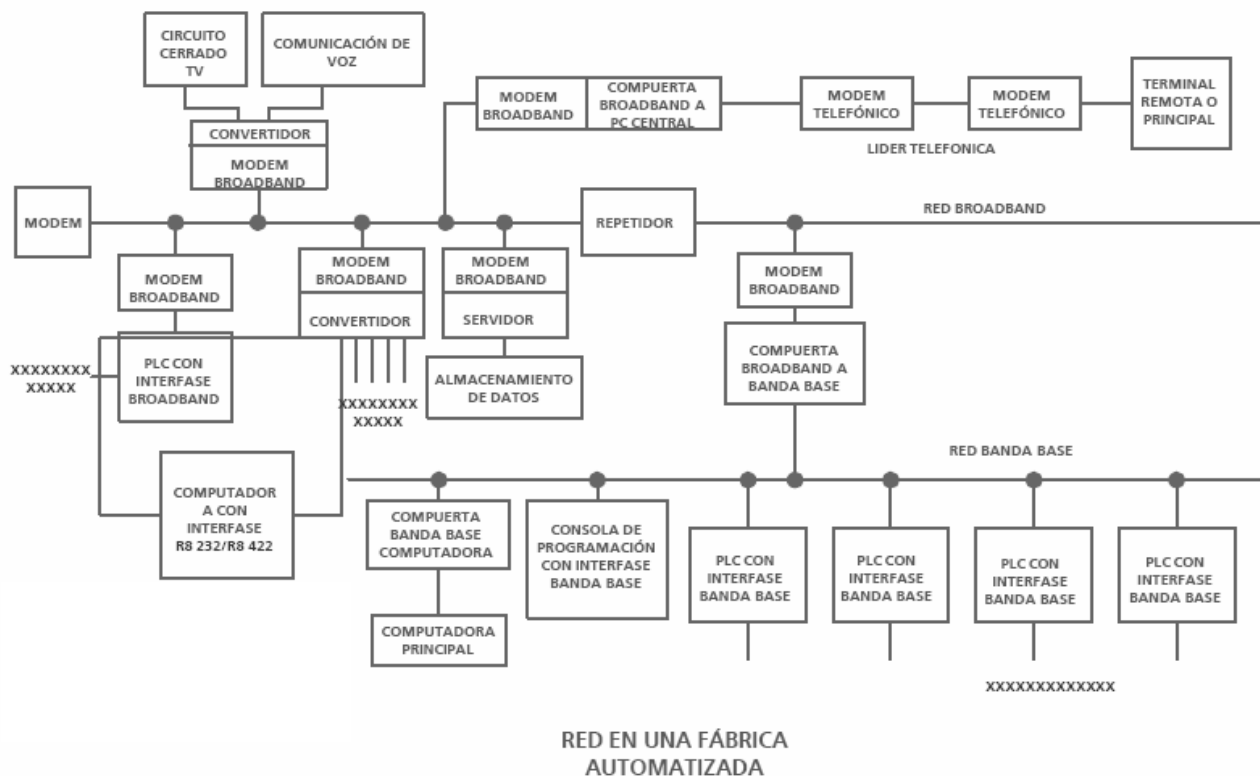
Para lograr el procesamiento de información en forma distribuida debe existir comunicación entre PLCs, transferir información de PLCs a computadoras, leer/escribir valores de entrada/salida de cualquier PLC, monitorear el estado del PLC y controlar su operación.

La topología de la red de área local define la geometría de la red o la forma en que se conectan a la red los nodos individuales.

Factores a los que afecta la topología

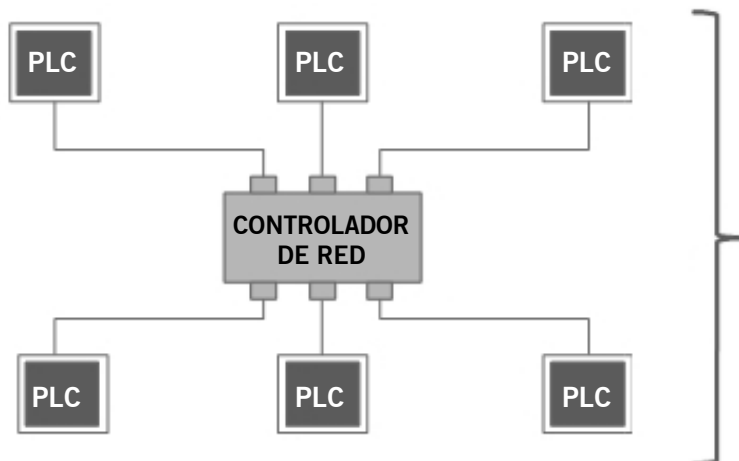
- Rendimiento;
- Costos de Implementación;
- Confiabilidad.

Las topologías básicas usadas en la actualidad son la estrella, la de bus común y la de anillo. Sin embargo, redes muy grandes como la de la figura, pueden consistir de una cantidad de topologías interconectadas.



• Topología Estrella

Las primeras redes con PLCs consistieron en una computadora central con múltiples puertos de comunicación, cada uno de los cuales se conectaba al puerto de programación de los PLC.



Cuando varios dispositivos se conectan directamente con el controlador de red, como se muestra en la figura, se establece una topología de red conocida como “estrella”.

El controlador de red puede ser una computadora, un PLC o algún otro dispositivo.

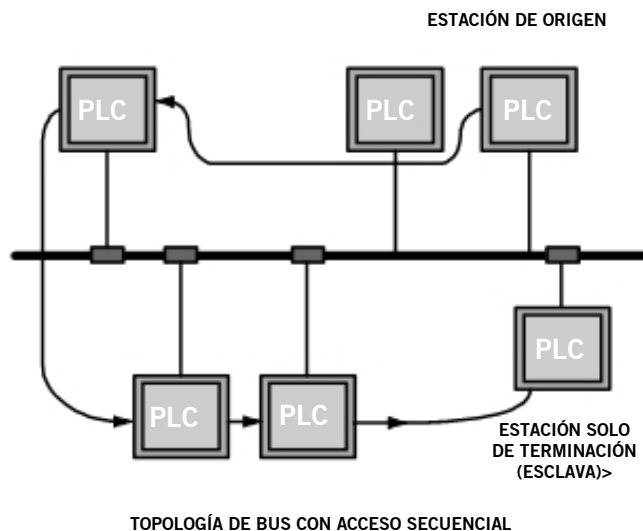
Ventajas:

- Se puede implementar con un protocolo simple de punto a punto.
- Cada nodo transmite cuando sea necesario.

Desventajas:

- Son los costos de alambrado para redes grandes.
- Los mensajes entre dos nodos tienen que pasar por el nodo central.
- Si falla el nodo central deja de funcionar toda la red.

• Topología De Bus Común



La topología de bus común se caracteriza porque tiene una línea principal a la que se conectan los nodos individuales.

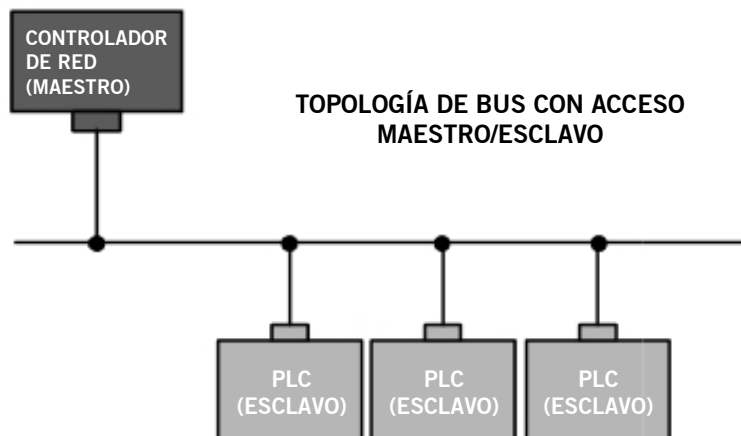
Cuando se tiene un bus común, la comunicación entre dos nodos se establece sin pasar por el controlador de red. Sin embargo, se presenta el problema de qué nodo es el que debe transmitir en un momento determinado.

Para solucionar el problema se han desarrollado métodos de acceso como el **acceso secuencial por token (token passing bus)** y el **acceso maestro/ esclavo**. En el primero, sólo una estación tiene derecho a transmitir temporalmente. Si no tiene nada que transmitir o termina su transmisión, pasa el derecho a transmitir al siguiente nodo.

Normalmente se usa un **cable coaxial** como el medio de comunicación.

Principal desventaja:

Se tiene que compartir el mismo bus o canal de transmisión de datos para darle servicio a todos los nodos, lo cual pudiera generar un aumento en el tiempo de respuesta de los dispositivos.



La topología de bus común es muy aplicable a control distribuido, ya que cada estación tiene la capacidad de control independiente y puede intercambiar información en cualquier instante.

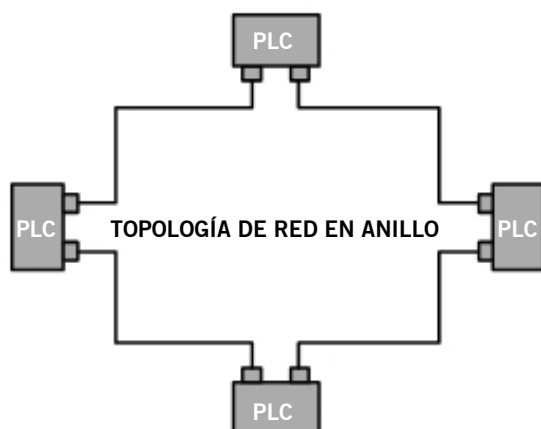
Otra implementación de la topología de bus consiste de un controlador de red (controlador maestro) y varios controladores esclavos como se muestra en la figura.

Cuando es necesario, el controlador maestro envía datos a los controladores esclavos y cuando requiere información de alguno de ellos la solicita y espera por su respuesta.

IMPORTANTE

En la configuración maestro/esclavo no existe comunicación a menos que sea iniciada por el controlador maestro.

- **Topología De Anillo**



La topología de anillo no es muy utilizada en el ambiente industrial porque **la falla de cualquier nodo hace que falle toda la red**, a menos que el nodo con falla sea "puenteado".

En la figura se muestra una red con topología de anillo

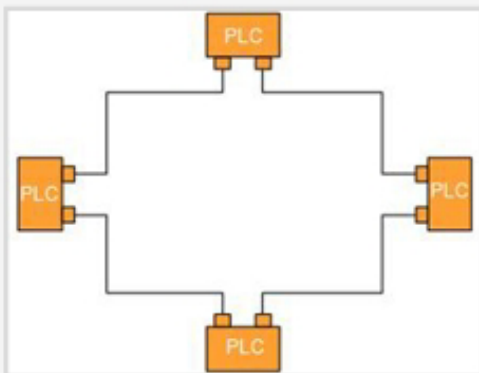
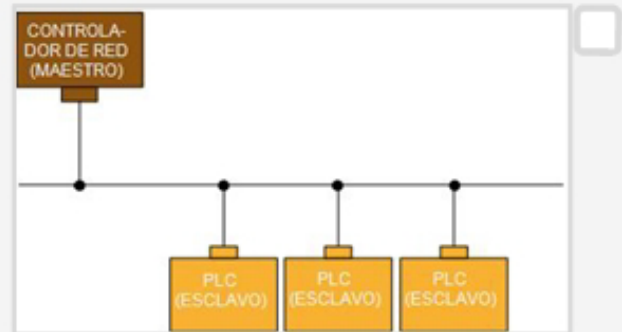
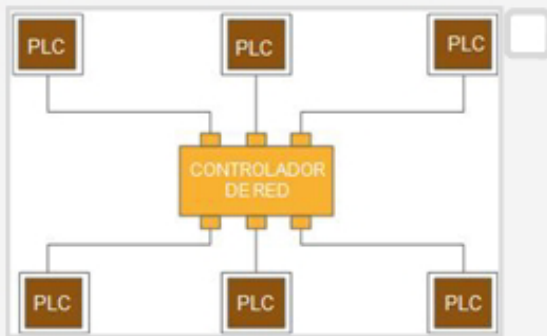
ACTIVIDAD. Interfases de Comunicación

Se dieron a conocer características principales de los códigos de programación de un PLC.



Por favor resuelva las siguientes actividades acerca de topologías de PLCs.

Usted debe diseñar una topología de PLCs. La red por la cual se transmitirán los datos tiene un ancho de banda limitado y se comparte con otras aplicaciones. El controlador de red está en un sitio remoto. Qué topología le parece conveniente en estas condiciones?



Una aplicación industrial con PLCs, que maneja procesos críticos, usa la tipología de la figura. Cree usted que es adecuada?

8.3 Medios de Comunicación

Los medios de comunicación que se usan comúnmente en las redes de PLC's incluyen cables de par torcido, cables coaxiales y fibra óptica.

El funcionamiento de la red (por ejemplo, la velocidad y la distancia) se ve afectado por el tipo de medio usado y la cantidad de nodos instalados.

PAR DE CABLES TORCIDOS

- El par torcido se ha utilizado extensivamente en la **industria en comunicación de punto a punto**, con distancias de hasta 1.2 Km y rapidez de transmisión de hasta 250 Kbaud.
- El par torcido es relativamente barato y tiene buena inmunidad al ruido, que se puede incrementar si se usa blindaje. Su funcionalidad disminuye rápidamente si se agregan nodos al bus de par torcido.

CABLE COAXIAL PARA BANDA BASE (BASEBAND)

- Las limitaciones principales del par torcido se deben a su falta de uniformidad. La impedancia característica en el cable varía haciendo difícil reducir las reflexiones debido a que no se tiene un valor adecuado de resistencia de terminación.
- El cable coaxial es muy uniforme y se elimina la reflexión. El factor que lo limita es la pérdida capacitiva y resistiva.
- Los cables coaxiales en banda base se usan en redes de área local con velocidades de hasta 2 Mbaud y distancia de hasta 5.48 Km. Normalmente es de 3/8" de diámetro.

CABLE COAXIAL DE BANDA ANCHA

- El cable coaxial de banda ancha tiene de 1/2" a 1" de diámetro y ha sido usado por muchos años para llevar señales de TV.
- La velocidad de transmisión en un canal es típicamente de 1, 5, 10 Mbaud aunque puede soportar hasta 150 Mbaud y es capaz de cubrir hasta 30 millas con repetidores bidireccionales. Puede soportar miles de nodos.

FIBRA ÓPTICA

La fibra óptica tiene ventajas impresionantes: es totalmente inmune a la interferencia electromagnética, su diámetro es muy pequeño y es muy ligera. La rapidez de transmisión es muy elevada y se puede transmitir a grandes distancias. Una de sus desventajas es el costo relativamente alto de la fibra y de los dispositivos acopladores.

Especificaciones de la Red

Existen algunas especificaciones importantes que deben tomarse en cuenta para determinar si una red puede soportar determinada aplicación. A continuación se comentarán brevemente cada una de ellas.

| | |
|--|---|
| Cantidad Máxima de Dispositivos | La cantidad máxima de dispositivos determina cuántos nodos son necesarios. Generalmente se considera cierta cantidad de nodos para expansiones futuras, por ejemplo 10%. Se debe considerar qué tipo de dispositivos se conectarán a cada nodo; los dispositivos pueden ser computadoras, terminales inteligentes y PLCs, entre otros. |
| Longitud Máxima | La longitud máxima de la red normalmente se especifica en dos partes, la longitud máxima del cable principal y la longitud máxima del cable de derivación (drop). La longitud del cable de derivación está usualmente en el rango de 10 a 30m pero se recomienda mantenerla tan corta como sea posible. |
| Tiempo de Respuesta | El tiempo de respuesta se considera como el tiempo transcurrido entre la transición de entrada a un nodo y la transición de salida del otro nodo. El tiempo de respuesta es la suma del tiempo necesario para detectar la transición de entrada (transmitir la información al nodo de salida) y operar la salida. |
| Throughput | Este valor usualmente representa la cantidad de puntos de entrada/salida que pueden ser actualizados por segundo a través de la red. Sólo proporciona una idea del tiempo de acceso y la rapidez de comunicación ya que estos valores cambian con el tiempo de procesamiento de cada nodo. El máximo throughput de un nodo en una red, es sinónimo de su capacidad. |
| Dispositivos Soportados | Se debe considerar no solamente si un determinado dispositivo puede ser soportado por la red, sino también, qué se requiere para conectar el dispositivo y el software de soporte necesario. Las redes soportan al menos una marca de PLCs. Los PLCs se conectan a la red a través de módulos de comunicación que se instalan directamente en la base del PLC. Los dispositivos de programación pueden o no conectarse directamente a la red. En caso de que no se conecten directamente, la programación debe hacerse a través del puerto de programación de cada PLC. Si la unidad de programación se conecta directamente a la red se puede programar cualquier PLC conectado a la red, así como también, tener cierta capacidad de monitoreo y control. Las terminales inteligentes son pequeñas computadoras con sistema operativo y memoria para almacenamiento de datos. Se conectan a la red de la misma forma que las computadoras principales. Si se considera utilizar una de estas terminales es necesario analizar detenidamente los requerimientos del software para determinar si el sistema operativo de la terminal cumple con dichos requerimientos. |

ACTIVIDAD 13. Interfases de Comunicación

Luego de conocer la forma de comunicación de un PLC realice la siguiente actividad.



¿Qué tipo de cable es el adecuado para cada aplicación?
Una con flechas cada tipo de cable con su correspondiente aplicación.

PAR CABLE TORCIDO



TRANSMISIÓN DE SEÑALES DE TV

COAXIAL PARA BANDA
BASETRANSMISIÓN DE ALTAS A GRANDES
DISTANCIAS

COAXIAL BANDA ANCHA



CABLEADO DE REDES LOCALES (LAN)

FIBRA ÓPTICA



COMUNICACIONES PUNTO A PUNTO

En este punto finaliza la explicación sobre interfases de Comunicación.

¡Felicitaciones! Ha finalizado el curso PLC Básico.

