

CONTROLADORES

INTRODUCCION A LOS CONTROLADORES LOGICOS PROGRAMABLES

TABLA DE CONTENIDO

1- PLC.....	1
1.1 CAMPOS DE APLICACIÓN.....	1
1.2 VENTAJAS E INCONVENIENTES.....	2
1.3 FUNCIONES BÁSICAS DE UN PLC	3
2. ESTRUCTURA	4
2.1 ESTRUCTURA INTERNA	5
2.1.1 CPU	5
2.1.2 FUENTE DE ALIMENTACION.....	7
2.1.3 INTERFACES	7
2.1.4 UNIDAD DE PROGRAMACION	8
2.1.5 ENTRADAS Y SALIDAS	8
2.1.6 MEMORIA	11
2.2 ESTRUCTURA EXTERNA	14
2.2.1 Estructura compacta.....	14
2.2.2 Estructura semimodular.....	14
2.2.3 Estructura modular.....	14
3 FUNCIONAMIENTO DEL PLC.....	18
3.1 MODO DE FUNCIONAMIENTO	19
3.2 CICLO DE FUNCIONAMIENTO.....	20
3.3 TIEMPO DE EJECUCIÓN Y CONTROL EN TIEMPO REAL	22
4 LA PROGRAMACION	23
4.1 GRAFCET	23
4.2 LENGUAJES DE PROGRAMACION.....	23
4.2.1 CONTACTOS.....	24
4.2.2 MNEMOTECNICO.....	26

1- PLC

Un autómata programable industrial (API) o Programmable logic controller (PLC), es un equipo electrónico, programable en lenguaje no informático, diseñado para controlar en tiempo real y en ambiente de tipo industrial, procesos secuenciales.

Un PLC trabaja en base a la información recibida por los captadores y el programa lógico interno, actuando sobre los accionadores de la instalación.

1.1 Campos de aplicación

El PLC por sus especiales características de diseño tiene un campo de aplicación muy extenso. La constante evolución del hardware y software amplía constantemente este campo para poder satisfacer las necesidades que se detectan en el espectro de sus posibilidades reales.

Su utilización se da fundamentalmente en aquellas instalaciones en donde es necesario un proceso de maniobra, control, señalización, etc. , por tanto, su aplicación abarca desde procesos de fabricación industriales de cualquier tipo a transformaciones industriales, control de instalaciones, etc.

Sus reducidas dimensiones, la extremada facilidad de su montaje, la posibilidad de almacenar los programas para su posterior y rápida utilización, la modificación o alteración de los mismos, etc., hace que su eficacia se aprecie fundamentalmente en procesos en que se producen necesidades tales como:

- Espacio reducido
- Procesos de producción periódicamente cambiantes
- Procesos secuenciales
- Maquinaria de procesos variables
- Instalaciones de procesos complejos y amplios
- Chequeo de programación centralizada de las partes del proceso

Ejemplos de aplicaciones generales:

- Maniobra de máquinas
- Maquinaria industrial de plástico
- Máquinas transfer
- Maquinaria de embalajes
- Maniobra de instalaciones:
 - Instalación de aire acondicionado, calefacción...
 - Instalaciones de seguridad
- Señalización y control:
 - Chequeo de programas
 - Señalización del estado de procesos

1.2 Ventajas e inconvenientes

No todos los autómatas ofrecen las mismas ventajas sobre la lógica cableada, ello es debido, principalmente, a la variedad de modelos existentes en el mercado y las innovaciones técnicas que surgen constantemente. Tales consideraciones me obligan a referirme a las ventajas que proporciona un autómata de tipo medio.

Ventajas

- Menor tiempo empleado en la elaboración de proyectos debido a que:
 - No es necesario dibujar el esquema de contactos
 - No es necesario simplificar las ecuaciones lógicas, ya que, por lo general la capacidad de almacenamiento del módulo de memoria es lo suficientemente grande.
- La lista de materiales queda sensiblemente reducida, y al elaborar el presupuesto correspondiente eliminaremos parte del problema que supone el contar con diferentes proveedores, distintos plazos de entrega.

- Posibilidad de introducir modificaciones sin cambiar el cableado ni añadir aparatos.
- Mínimo espacio de ocupación.
- Menor coste de mano de obra de la instalación.
- Economía de mantenimiento. Además de aumentar la fiabilidad del sistema, al eliminar contactos móviles, los mismos autómatas pueden indicar y detectar averías.
- Posibilidad de gobernar varias máquinas con un mismo autómata.
- Menor tiempo para la puesta en funcionamiento del proceso al quedar reducido el tiempo cableado.
- Si por alguna razón la máquina queda fuera de servicio, el autómata sigue siendo útil para otra máquina o sistema de producción.

Inconvenientes

- Como inconvenientes podríamos hablar, en primer lugar, de que hace falta un programador, lo que obliga a adiestrar a uno de los técnicos en tal sentido, pero hoy en día ese inconveniente está solucionado porque las universidades ya se encargan de dicho adiestramiento.
- El coste inicial también puede ser un inconveniente.

1.3 Funciones básicas de un PLC

■ Detección:

Lectura de la señal de los captadores distribuidos por el sistema de fabricación.

■ Mando:

Elaborar y enviar las acciones al sistema mediante los accionadores y preaccionadores.

■ Diálogo hombre máquina:

Mantener un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.

■ Programación:

Para introducir, elaborar y cambiar el programa de aplicación del autómeta. El dialogo de programación debe permitir modificar el programa incluso con el autómeta controlando la maquina.

Nuevas Funciones

■ Redes de comunicación:

Permiten establecer comunicación con otras partes de control. Las redes industriales permiten la comunicación y el intercambio de datos entre autómetas a tiempo real. En unos cuantos milisegundos pueden enviarse telegramas e intercambiar tablas de memoria compartida.

■ Sistemas de supervisión:

También los autómetas permiten comunicarse con ordenadores provistos de programas de supervisión industrial. Esta comunicación se realiza por una red industrial o por medio de una simple conexión por el puerto serie del ordenador.

■ Control de procesos continuos:

Además de dedicarse al control de sistemas de eventos discretos los autómetas llevan incorporadas funciones que permiten el control de procesos continuos. Disponen de módulos de entrada y salida analógicas y la posibilidad de ejecutar reguladores PID que están programados en el autómeta.

■ Entradas- Salidas distribuidas:

Los módulos de entrada salida no tienen porqué estar en el armario del autómeta. Pueden estar distribuidos por la instalación, se comunican con la unidad central del autómeta mediante un cable de red.

■ Buses de campo:

Mediante un solo cable de comunicación se pueden conectar al bus captadores y accionadores, reemplazando al cableado tradicional. El autómeta consulta cíclicamente el estado de los captadores y actualiza el estado de los accionadores.

2. ESTRUCTURA

Introducción

Este capítulo está dedicado a conocer al autómeta en su parte física o hardware, no sólo en su configuración externa, sino también y fundamentalmente en la parte interna.

El autómata esta compuesto de diferentes elementos como CPU, fuente de alimentación, memoria, E/S, etc. que están colocados de diferente forma y modo según la estructura externa del autómata.

En el apartado dedicado a la estructura externa veremos donde y como se colocan los diferentes elementos que componen el autómata programable.

En el apartado dedicado a la estructura interna veremos la función que desempeña cada uno de los diferentes elementos como CPU, E/S, fuente de alimentación, etc..

2.1 ESTRUCTURA INTERNA

Introducción

En este apartado vamos a estudiar la estructura interna de cada uno de los diferentes elementos que componen el autómata, las funciones y funcionamiento de cada una de ellas.

El autómata está constituido por diferentes elementos, pero tres son los básicos:

- CPU
- Entradas
- Salidas

Con las partes mencionadas podemos decir que tenemos un autómata pero para que sea operativo son necesarios otros elementos tales como:

- Fuente de alimentación
- Interfaces
- La unidad o consola de programación
- Los dispositivos periféricos

2.1.1 CPU

Introducción

La CPU(Central Procesing Unit) es la parte inteligente del sistema. Interpreta las instrucciones del programa de usuario y consulta el estado de las entradas. Dependiendo de dichos estados y del programa, ordena la activación de las salidas deseadas.

La CPU está constituida por los siguientes elementos:

- Procesador
- Memoria monitor del sistema
- Circuitos auxiliares

Procesador

Está constituido por el microprocesador, el reloj(generator de onda cuadrada) y algún chip auxiliar.

El micropocesador es un circuito integrado (chip), que realiza una gran cantidad de operaciones, que podemos agrupar en:

- Operaciones de tipo lógico.
- Operaciones de tipo aritmético.
- Operaciones de control de la transferencia de la información dentro del autómata.

Para que el microprocesador pueda realizar todas estas operaciones está dotado de unos circuitos internos que son los siguientes:

- *Circuitos de la unidad aritmética y lógica o ALU*: Es la parte del μp donde se realizan los cálculos y las decisiones lógicas para controlar el autómata.
- *Circuitos de la unidad de control (UC) o Decodificador de instrucciones*: Decodifica las instrucciones leídas en memoria y se generan las señales de control.
- *Acumulador*: Es la encargada de almacenar el resultado de la última operación realizada por el ALU.
- *Flags*: Flags, o indicadores de resultado, que pueden ser consultados por el programa.
- *Contador de programa*: Encargada de la lectura de las instrucciones de usuario.
- *Bus(interno)*: No son circuitos en si, sino zonas conductoras en paralelo que transmiten datos, direcciones, instrucciones y señales de control entre las diferentes partes del μp .

Memoria monitor del sistema

Es una memoria de tipo ROM, y además del sistema operativo del autómata contiene las siguientes rutinas, incluidas por el fabricante.

- Inicialización tras puesta en tensión o reset.
- Rutinas de test y de respuesta a error de funcionamiento.
- Intercambio de información con unidades exteriores.
- Lectura y escritura en las interfaces de E/S.

Funciones básicas de la CPU

En la memoria ROM del sistema, el fabricante ha grabado una serie de programas ejecutivos, software del sistema y es a estos programas a los que accederá el μp para realizar las funciones.

El software del sistema de cualquier autómata consta de una serie de funciones básicas que realiza en determinados tiempos de cada ciclo.

En general cada autómata contiene y realiza las siguientes funciones:

- Vigilar que el tiempo de ejecución del programa de usuario no exceda de un determinado tiempo máximo. A esta función se le denomina Watchdog.
- Ejecutar el programa usuario.
- Crear una imagen de las entradas, ya que el programa de usuario no debe acceder directamente a dichas entradas.
- Renovar el estado de las salidas en función de la imagen de las mismas, obtenida al final del ciclo de ejecución del programa usuario.
- Chequeo del sistema.

2.1.2 FUENTE DE ALIMENTACION

La fuente de alimentación proporciona las tensiones necesarias para el funcionamiento de los distintos circuitos del sistema.

La alimentación a la CPU puede ser de continua a 24 Vcc, tensión muy frecuente en cuadros de distribución, o en alterna a 110/220 Vca. En cualquier caso es la propia CPU la que alimenta las interfaces conectadas a través del bus interno.

La alimentación a los circuitos E/S puede realizarse, según tipos, en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc.

La fuente de alimentación del autómata puede incorporar una batería tampón, que se utiliza para el mantenimiento de algunas posiciones internas y del programa usuario en memoria RAM, cuando falla la alimentación o se apaga el autómata.

2.1.3 INTERFACES

En el control de un proceso automatizado, es imprescindible un dialogo entre operador-máquina junto con una comunicación entre la máquina y el autómata, estas comunicaciones se establecerán por medio del conjunto de entradas y salidas del citado elemento.

Los autómatas son capaces de manejar tensiones y corrientes de nivel industrial, gracias a que disponen un bloque de circuitos de interfaz de E/S muy potente, que les permite conectarse directamente con los sensores y accionamientos del proceso.

De entre todos los tipos de interfaces que existen, las interfaces específicas permiten la conexión con elementos muy concretos del proceso de automatización. Se pueden distinguir entre ellas tres grupos bien diferenciados:

- Entradas / salidas especiales.
- Entradas / salidas inteligentes.
- Procesadores periféricos inteligentes.

Las interfaces especiales del primer grupo se caracterizan por no influir en las variables de estado del proceso de automatización. Únicamente se encargan de adecuar las E/S,

para que puedan ser inteligibles por la CPU, si son entradas, o para que puedan ser interpretadas correctamente por actuadores (motores, cilindros, etc.), en el caso de las salidas.

Las del segundo grupo admiten múltiples modos de configuración, por medio de unas combinaciones binarias situadas en la misma tarjeta. De esta forma se descarga de trabajo a la unidad central, con las ventajas que conlleva.

Los procesadores periféricos inteligentes, son módulos que incluyen su propio procesador, memorias y puntos auxiliares de entrada / salida. Estos procesadores contienen en origen un programa especializado en la ejecución de una tarea concreta, a la que le basta conocer los puntos de consigna y los parámetros de aplicación para ejecutar, de forma autónoma e independiente de la CPU principal, el programa de control.

2.1.4 UNIDAD DE PROGRAMACION

Es el conjunto de medios hardware y software mediante los cuales el programador introduce y depura sobre las secuencias de instrucciones (en uno u otro lenguaje) que constituyen el programa a ejecutar.

2.1.5 ENTRADAS Y SALIDAS

Introducción

La sección de entradas mediante el interfaz, adapta y codifica de forma comprensible para la CPU las señales procedentes de los dispositivos de entrada o captadores.

Hay dos tipos de entradas:

- Entradas digitales
- Entradas analógicas

La sección de salida también mediante interfaz trabaja de forma inversa a las entradas, es decir, decodifica las señales procedentes de la CPU, y las amplifica y manda con ellas los dispositivos de salida o actuadores como lámparas, relés... aquí también existen unos interfaces de adaptación a las salidas de protección de circuitos internos.

Hay dos tipos de salidas:

- Salidas digitales
- Salidas analógicas

Entradas digitales

Los módulos de entrada digitales permiten conectar al autómata captadores de tipo todo o nada como finales de carrera pulsadores...

Los módulos de entrada digitales trabajan con señales de tensión, por ejemplo cuando por una vía llegan 24 voltios se interpreta como un "1" y cuando llegan cero voltios se interpreta como un "0"

El proceso de adquisición de la señal digital consta de varias etapas.

- Protección contra sobretensiones
- Filtrado
- Puesta en forma de la onda
- Aislamiento galvánico o por optoacoplador.

Entradas analógicas

Los módulos de entrada analógicas permiten que los autómatas programables trabajen con accionadores de mando analógico y lean señales de tipo analógico como pueden ser la temperatura, la presión o el caudal.

Los módulos de entradas analógicas convierten una magnitud analógica en un número que se deposita en una variable interna del autómata. Lo que realiza es una conversión A/D, puesto que el autómata solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (número de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Los módulos de entrada analógica pueden leer tensión o intensidad.

El proceso de adquisición de la señal analógica consta de varias etapas:

- Filtrado
- Conversión A/D
- Memoria interna

Salidas digitales

Un módulo de salida digital permite al autómata programable actuar sobre los preaccionadores y accionadores que admitan órdenes de tipo todo o nada.

El valor binario de las salidas digitales se convierte en la apertura o cierre de un relé interno del autómatas en el caso de módulos de salidas a relé.

En los módulos estáticos (bornero), los elementos que conmutan son los componentes electrónico como transistores o triacs, y en los módulos electromecánicos son contactos de relés internos al módulo.

Los módulos de salidas estáticos al suministrar tensión, solo pueden actuar sobre elementos que trabajan todos a la misma tensión, en cambio los módulos de salida electromecánicos, al ser libres de tensión, pueden actuar sobre elementos que trabajen a tensiones distintas.

El proceso de envío de la señal digital consta de varias etapas:

- Puesta en forma
- Aislamiento
- Circuito de mando (relé interno)
- Protección electrónica
- Tratamiento cortocircuitos

Salidas analógicas

Los módulos de salida analógica permiten que el valor de una variable numérica interna del autómatas se convierta en tensión o intensidad.

Lo que realiza es una conversión D/A, puesto que el autómatas solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (numero de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Esta tensión o intensidad puede servir de referencia de mando para actuadores que admitan mando analógico como pueden ser los variadores de velocidad, las etapas de los tiristores de los hornos, reguladores de temperatura... permitiendo al autómatas realiza funciones de regulación y control de procesos continuos.

El proceso de envío de la señal analógica consta de varias etapas:

- Aislamiento galvánico
- Conversión D/A
- Circuitos de amplificación y adaptación
- Protección electrónica de la salida

Como hemos visto las señales analógicas sufren un gran proceso de adaptación tanto en los módulos de entrada como en los módulos de salida. Las funciones de conversión A/D y D/A que realiza son esenciales. Por ello los módulos de E/S analógicos se les considera módulos de E/S especiales.

2.1.6 MEMORIA

La memoria es el almacén donde el autómatas guarda todo cuanto necesita para ejecutar la tarea de control.

Datos del proceso:

- Señales de planta, entradas y salidas.
- Variables internas, de bit y de palabra.
- Datos alfanuméricos y constantes.

Datos de control:

- Instrucciones de usuario (programa)
- Configuración del autómatas (modo de funcionamiento, número de e/s conectadas, ...)

Existen varios tipos de memorias:

- RAM. Memoria de lectura y escritura.
- ROM. Memoria de solo lectura, no reprogramable.
- EPROM. Memoria de solo lectura, reprogramables con borrado por ultravioletas.
- EEPROM. Memoria de solo lectura, alterables por medios eléctricos.

La memoria RAM se utiliza principalmente como memoria interna, y únicamente como memoria de programa en el caso de que pueda asegurarse el mantenimiento de los datos con una batería exterior.

La memoria ROM se utiliza para almacenar el programa monitor del sistema como hemos visto en el apartado dedicado a la CPU.

Las memorias EPROM se utilizan para almacenar el programa de usuario, una vez que ha sido convenientemente depurada.

Las memorias EEPROM se emplean principalmente para almacenar programas, aunque en la actualidad es cada vez más frecuente el uso de combinaciones RAM + EEPROM, utilizando estas últimas como memorias de seguridad que salvan el contenido de las RAM. Una vez reanudada la alimentación, el contenido de la EEPROM se vuelca sobre la RAM. Las soluciones de este tipo están sustituyendo a las clásicas RAM + batería puesto que presentan muchos menos problemas.

Memoria interna

En un autómatas programable, la memoria interna es aquella que almacena el estado de las variables que maneja el autómatas: entradas, salidas, contadores, relés internos, señales de estado, etc. Esta memoria interna se encuentra dividida en varias áreas, cada una de ellas con un cometido y características distintas.

La clasificación de la memoria interna no se realiza atendiendo a sus características de lectura y escritura, sino por el tipo de variables que almacena y el número de bits que ocupa la variable. Así, la memoria interna del autómatas queda clasificada en las siguientes áreas.

Área de imágenes de entradas/salidas y Área interna (IR).

En esta área de memoria se encuentran:

- Los canales (registros) asociados a los terminales externos (entradas y salidas).
- Los relés (bit) internos (no correspondidos con el terminal externo), gestionados como relés de E/S.
- Los relés E/S no usados pueden usarse como IR.
- No retienen estado frente a la falta de alimentación o cambio de modo de operación.

Área especial (SR).

Son relés de señalización de funciones particulares como:

- Servicio (siempre ON, OFF)
- Diagnóstico (señalización o anomalías)
- Temporizaciones (relojes a varias frecuencias)
- Cálculo
- Comunicaciones
- Accesible en forma de bit o de canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

Área auxiliar (AR).

Contienen bits de control e información de recursos de PLC como: Puerto RS232C, puertos periféricos, cassetes de memoria...

- Se dividen en dos bloques:

Señalización: Errores de configuración, datos del sistema.

Memorización y gestión de datos

- Es un área de retención.
- Accesible en forma de bit o de canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

Área de enlace (LR).

- Se utilizan para el intercambio de datos entre dos PLC's unidos en forma PC Link(1:1).
- Dedicados al intercambio de información entre PLC's.
- Si no se utilizan como LR pueden usarse como IR.
- Accesible en forma de bit o canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

Área de retención (HR).

- Mantienen su estado ante fallos de alimentación o cambio de modo de PLC.
- Son gestionados como los IR y direccionables como bit o como canal.

Área de temporizadores y contadores (TIM/CNT).

- Es el área de memoria que simula el funcionamiento de estos dispositivos.
- Son usados por el PLC para programar retardos y contajes.

Área de datos (DM).

- Se trata de memoria de 16 bits (palabra).
- Utilizable para gestión de valores numéricos.
- Mantiene su estado ante cambios de modos de trabajo o fallo de alimentación.
- Direccionables como Canal(palabra).
- Esta área suele contener los parámetros de configuración del PLC(setup)

Las variables contenidas en la memoria interna, pueden ser consultadas y modificadas continuamente por el programa, cualquier número de veces. Esta actualización continua de los datos obliga a construir la memoria con dispositivos RAM.

Memoria de programa

La memoria de programa, normalmente externa y enchufable a la CPU mediante casete de memoria, almacena el programa escrito por el usuario para su aplicación.

Cada instrucción del usuario ocupa un paso o dirección del programa.

Las memorias de programa o memorias de usuario son siempre de tipo permanente RAM + batería o EPROM/EEPROM . Por lo general la mayoría de los

fabricantes de autómatas ofrecen la posibilidad de utilizar memorias RAM con batería para la fase de desarrollo y depuración de los programas, y de pasar estos a memorias no volátiles EPROM o EEPROM una vez finalizada esta fase.

La ejecución del programa en el módulo es siempre prioritaria, de forma que si se da tensión al autómata con un módulo conectado, la CPU ejecuta su programa y no el contenido en memoria RAM interna.

2.2 ESTRUCTURA EXTERNA

El término estructura externa o configuración externa de un autómata programable industrial se refiere al aspecto físico exterior del mismo, bloques o elementos en que está dividido.

Actualmente son tres las estructuras más significativas que existen en el mercado:

- Estructura compacta.
- Estructura semimodular. (Estructura Americana)
- Estructura modular. (Estructura Europea)

2.2.1 Estructura compacta

Este tipo de autómatas se distingue por presentar en un solo bloque todos sus elementos, esto es, fuente de alimentación, CPU, memorias, entradas/salidas, etc..

Son los autómatas de gama baja o nanoautómatas los que suelen tener una estructura compacta. Su potencia de proceso suele ser muy limitada dedicándose a controlar máquinas muy pequeñas o cuadros de mando.

2.2.2 Estructura semimodular

Se caracteriza por separar las E/S del resto del autómata, de tal forma que en un bloque compacto están reunidas las CPU, memoria de usuario o de programa y fuente de alimentación y separadamente las unidades de E/S .

Son los autómatas de gama media los que suelen tener una estructura semimodular (Americana).

2.2.3 Estructura modular

Su característica principal es la de que existe un módulo para cada uno de los diferentes elementos que componen el autómata como puede ser una fuente de alimentación, CPU, E/S, etc. La sujeción de los mismos se hace por carril DIN, placa perforada o sobre

RACK, en donde va alojado el BUS externo de unión de los distintos módulos que lo componen.

Son los autómatas de gama alta los que suelen tener una estructura modular, que permiten una gran flexibilidad en su constitución.

Introducción

Los autómatas programables son máquinas secuenciales que ejecutan correlativamente las instrucciones indicadas en el programa de usuario almacenado en su memoria, generando unas ordenes o señales de mando a partir de las señales de entrada leídas de la planta (aplicación): al detectarse cambios en las señales, el autómata reacciona según el programa hasta obtener las ordenes de salida necesarias. Esta secuencia se ejecuta continuamente para conseguir el control actualizado del proceso.

La secuencia básica de operación del autómata se puede dividir en tres fases principales:

- Lectura de señales desde la interfaz de entradas.
- Procesado del programa para obtención de las señales de control.
- Escritura de señales en la interfaz de salidas.

A fin de optimizar el tiempo, la lectura y escritura de las señales se realiza a la vez para todas las entradas y salidas; Entonces, las entradas leídas de los módulos de entrada se guardan en una memoria temporal (Imagen entradas). A esta acude la CPU en la ejecución del programa, y según se va obteniendo las salidas, se guardan en otra memoria temporal (imagen de salida). Una vez ejecutado el programa completo, estas imágenes de salida se transfieren todas a la vez al módulo de salida.

El autómata realiza también otra serie de acciones que se van repitiendo periódicamente, definiendo un ciclo de operación.

Modo de funcionamiento

El autómata de OMRON CQM1H puede trabajar de tres formas

diferentes:

Program

El PLC está en reposo y puede recibir o enviar el programa a un periférico.

Monitor

El PLC ejecuta el programa que tiene en memoria.

Run

El PLC ejecuta el programa que tiene en memoria permitiendo el cambio de valores en los registros del mismo.

Ciclo de funcionamiento

El funcionamiento del autómatas es, salvo el proceso inicial que sigue a un Reset, de tipo secuencial y cíclico, es decir, las operaciones tienen lugar una tras otra, y se van repitiendo continuamente mientras el autómatas esté bajo tensión.

La figura 2.2.1 muestra esquemáticamente la secuencia de operaciones que ejecuta el autómatas, siendo las operaciones del *ciclo de operación* las que se repiten indefinidamente.

El ciclo de funcionamiento se divide en dos partes como se puede observar en el esquema de diagrama de la figura 2.2.1 llamados *Proceso Inicial* y *Ciclo de Operación*.

Proceso inicial

Como se muestra en la figura, antes de entrar en el ciclo de operación el autómatas realiza una serie de acciones comunes, que tratan fundamentalmente de inicializar los estados del mismo y chequear el hardware. Estas rutinas de chequeo, incluidas en el programa monitor ROM, comprueban:

- El bus de conexiones de las unidades de E/S.
- El nivel de la batería, si esta existe.
- La conexión de las memorias internas del sistema.
- El módulo de memoria exterior conectado, si existe.

Si se encontrara algún error en el chequeo, se activaría el LED de error y quedaría registrado el código del error.

Comprobadas las conexiones, se inicializan las variables internas:

- Se ponen a OFF las posiciones de memoria interna (excepto las mantenidas o protegidas contra pérdidas de tensión)
- Se borran todas las posiciones de memoria imagen E/S.
- Se borran todos los contadores y temporizadores (excepto los mantenidos o protegidos contra pérdidas de tensión).

Transcurrido el *Proceso Inicial* y si no han aparecido errores el autómata entra en el *Ciclo de Operación*.

Ciclo de operación

Este ciclo puede considerarse dividido en tres bloques tal y como se puede observar en la figura 2.2.1, dichos bloques son:

- Proceso Común
- Ejecución del programa
- Servicio a periféricos

Proceso común:

En este primer bloque se realizan los chequeos cíclicos de conexiones y de memoria de programa, protegiendo el sistema contra:

- Errores de hardware (conexiones E/S, ausencia de memoria de programa, etc).
- Errores de sintaxis (programa imposible de ejecutar).

El chequeo cíclico de conexiones comprueba los siguientes puntos:

- Niveles de tensión de alimentación.
- Estado de la batería si existe.
- Buses de conexión con las interfaces.

El chequeo de la memoria de programa comprueba la integridad de la misma y los posibles errores de sintaxis y gramática:

- Mantenimiento de los datos, comprobados en el "checksum".
- Existencia de la instrucción END de fin de programa.
- Estructura de saltos y anidamiento de bloque correctas.
- Códigos de instrucciones correctas.

Ejecución del programa:

En este segundo bloque se consultan los estados de las entradas y de las salidas y se elaboran las ordenes de mando o de salida a partir de ellos.

El tiempo de ejecución de este bloque de operaciones es la suma del:

- Tiempo de acceso a interfaces de E/S.
- Tiempo de escrutación de programa.

Y a su vez esto depende, respectivamente de:

- Numero y ubicación de las interfaces de E/S.
- Longitud del programa y tipo de CPU que lo procesa.

Servicio a periféricos:

Este tercer y último bloque es únicamente atendido si hay pendiente algún intercambio con el exterior. En caso de haberlo, la CPU le dedica un tiempo limitado, de 1 a 2ms, en atender el intercambio de datos. Si este tiempo no fuera suficiente, el servicio queda interrumpido hasta el siguiente ciclo.

Tiempo de ejecución y control en tiempo real

El tiempo total que el autómata emplea para realizar un ciclo de operación se llama tiempo de ejecución de *ciclo de operación* o más sencillamente tiempo de ciclo "Scan time".

Dicho tiempo depende de:

- El número de E/S involucradas.
- La longitud del programa usuario.
- El número y tipo de periféricos conectados al autómata.

Los tiempos totales de ciclos son entonces la suma de tiempos empleados en realizar las distintas operaciones del ciclo como se puede ver en la figura 2.2.2:

- Autodiagnóstico (Proceso común)
- Actualización de E/S (Ejecución del programa)
- Ejecución de programa.(Ejecución del programa)
- Servicio a periféricos.(Servicio a periféricos)

3 FUNCIONAMIENTO DEL PLC

Introducción

Los autómatas programables son máquinas secuenciales que ejecutan correlativamente las instrucciones indicadas en el programa de usuario almacenado en su memoria, generando unas ordenes o señales de mando a partir de las señales de entrada leídas de la planta (aplicación): al detectarse cambios en las señales, el autómata reacciona según el programa hasta obtener las ordenes de salida necesarias. Esta secuencia se ejecuta continuamente para conseguir el control actualizado del proceso.

La secuencia básica de operación del autómata se puede dividir en tres fases principales:

- Lectura de señales desde la interfaz de entradas.
- Procesado del programa para obtención de las señales de control.
- Escritura de señales en la interfaz de salidas.

A fin de optimizar el tiempo, la lectura y escritura de las señales se realiza a la vez para todas las entradas y salidas; Entonces, las entradas leídas de los módulos de entrada se guardan en una memoria temporal (Imagen entradas). A esta acude la CPU en la ejecución del programa, y según se va obteniendo las salidas, se guardan en otra memoria temporal (imagen de salida). Una vez ejecutado el programa completo, estas imágenes de salida se transfieren todas a la vez al módulo de salida.

El autómata realiza también otra serie de acciones que se van repitiendo periódicamente, definiendo un ciclo de operación.

3.1 Modo de funcionamiento

El autómata de OMRON CQM1H puede trabajar de tres formas

diferentes:

Program

El PLC está en reposo y puede recibir o enviar el programa a un periférico.

Monitor

El PLC ejecuta el programa que tiene en memoria.

Run

El PLC ejecuta el programa que tiene en memoria permitiendo el cambio de valores en los registros del mismo.

3.2 Ciclo de funcionamiento

El funcionamiento del autómata es, salvo el proceso inicial que sigue a un Reset, de tipo secuencial y cíclico, es decir, las operaciones tienen lugar una tras otra, y se van repitiendo continuamente mientras el autómata esté bajo tensión.

La figura 2.2.1 muestra esquemáticamente la secuencia de operaciones que ejecuta el autómata, siendo las operaciones del *ciclo de operación* las que se repiten indefinidamente.

El ciclo de funcionamiento se divide en dos partes como se puede observar en el esquema de diagrama de la figura 2.2.1 llamados *Proceso Inicial* y *Ciclo de Operación*.

Proceso inicial

Como se muestra en la figura, antes de entrar en el ciclo de operación el autómata realiza una serie de acciones comunes, que tratan fundamentalmente de inicializar los estados del mismo y chequear el hardware. Estas rutinas de chequeo, incluidas en el programa monitor ROM, comprueban:

- El bus de conexiones de las unidades de E/S.
- El nivel de la batería, si esta existe.
- La conexión de las memorias internas del sistema.
- El módulo de memoria exterior conectado, si existe.

Si se encontrara algún error en el chequeo, se activaría el LED de error y quedaría registrado el código del error.

Comprobadas las conexiones, se inicializan las variables internas:

- Se ponen a OFF las posiciones de memoria interna (excepto las mantenidas o protegidas contra pérdidas de tensión)
- Se borran todas las posiciones de memoria imagen E/S.
- Se borran todos los contadores y temporizadores (excepto los mantenidos o protegidos contra pérdidas de tensión).

Transcurrido el *Proceso Inicial* y si no han aparecido errores el autómata entra en el *Ciclo de Operación*.

Ciclo de operación

Este ciclo puede considerarse dividido en tres bloques tal y como se puede observar en la figura 2.2.1, dichos bloques son:

- Proceso Común
- Ejecución del programa
- Servicio a periféricos

Proceso común:

En este primer bloque se realizan los chequeos cíclicos de conexiones y de memoria de programa, protegiendo el sistema contra:

- Errores de hardware (conexiones E/S, ausencia de memoria de programa, etc).
- Errores de sintaxis (programa imposible de ejecutar).

El chequeo cíclico de conexiones comprueba los siguientes puntos:

- Niveles de tensión de alimentación.
- Estado de la batería si existe.
- Buses de conexión con las interfaces.

El chequeo de la memoria de programa comprueba la integridad de la misma y los posibles errores de sintaxis y gramática:

- Mantenimiento de los datos, comprobados en el "checksum".
- Existencia de la instrucción END de fin de programa.
- Estructura de saltos y anidamiento de bloque correctas.
- Códigos de instrucciones correctas.

Ejecución del programa:

En este segundo bloque se consultan los estados de las entradas y de las salidas y se elaboran las ordenes de mando o de salida a partir de ellos.

El tiempo de ejecución de este bloque de operaciones es la suma del:

- Tiempo de acceso a interfaces de E/S.
- Tiempo de escrutación de programa.

Y a su vez esto depende, respectivamente de:

- Numero y ubicación de las interfaces de E/S.
- Longitud del programa y tipo de CPU que lo procesa.

Servicio a periféricos:

Este tercer y último bloque es únicamente atendido si hay pendiente algún intercambio con el exterior. En caso de haberlo, la CPU le dedica un tiempo limitado, de 1 a 2ms, en atender el intercambio de datos. Si este tiempo no fuera suficiente, el servicio queda interrumpido hasta el siguiente ciclo.

3.3 Tiempo de ejecución y control en tiempo real

El tiempo total que el autómata emplea para realizar un ciclo de operación se llama tiempo de ejecución de *ciclo de operación* o más sencillamente tiempo de ciclo "Scan time".

Dicho tiempo depende de:

- El número de E/S involucradas.
- La longitud del programa usuario.
- El número y tipo de periféricos conectados al autómata.

Los tiempos totales de ciclos son entonces la suma de tiempos empleados en realizar las distintas operaciones del ciclo.

- Autodiagnóstico (Proceso común)
- Actualización de E/S (Ejecución del programa)
- Ejecución de programa.(Ejecución del programa)
- Servicio a periféricos.(Servicio a periféricos)

(Los tiempos de ejecución de instrucciones se miden en unidades de microsegundos, resultando un tiempo de escrutación del programa variable en función del número e instrucciones contenidas. Precisamente el tiempo de escrutación es uno de los parámetros que caracterizan a un autómata expresado normalmente en milisegundos por cada mil instrucciones ms/k).

$$\text{Tiempo total SCAN} = T1 + T2 + T3 + T4$$

4 LA PROGRAMACION

El sistema de programación permite, mediante las instrucciones del autómeta, confeccionar el programa de usuario. Posteriormente el programa realizado, se trasfiere a la memoria de programa de usuario.

Una memoria típica permite almacenar como mínimo hasta mil instrucciones con datos de bit, y es del tipo lectura/escritura, permitiendo la modificación del programa cuantas veces sea necesario.

Tiene una batería tampón para mantener el programa si falla la tensión de alimentación.

La programación del autómeta consiste en el establecimiento de una sucesión ordenada de instrucciones, escritas en un lenguaje de programación concreto.

Estas instrucciones están disponibles en el sistema de programación y resuelven el control de un proceso determinado.

4.1 GRAFCET

El GRAFCET surge en Francia a mediados de los años 70, debido a la colaboración de algunos fabricantes de autómetas, como Telemecanique y Aper con dos organismos oficiales, AFCET (Asociación francesa para la cibernética, economía y técnica) y ADEPA (Agencia nacional para el desarrollo de la producción automatizada). Homologado en Francia, Alemania, y posteriormente por la comisión Electrónica Internacional (IEC 848, año 1988) .

Actualmente es una herramienta imprescindible cuando se trata de automatizar procesos secuenciales de cierta complejidad con autómetas programables.

El GRAFCET es un diagrama funcional que describe la evolución del proceso que se quiere automatizar tal y como se muestra en la figura. Está definido por unos elementos gráficos y unas reglas de evolución que reflejan la dinámica del comportamiento del sistema.

Todo automatismo secuencial o concurrente se puede estructurar en una serie de etapas que representan estados o subestados del sistema en los cuales se realiza una o más acciones, así como transiciones, que son las condiciones que deben darse para pasar de una etapa a otra.

4.2 LENGUAJES DE PROGRAMACION

Cuando hablamos de los lenguajes de programación nos referimos a diferentes formas de poder escribir el programa usuario.

Los software actuales nos permiten traducir el programa usuario de un lenguaje a otro, pudiendo así escribir el programa en el lenguaje que más nos conviene.

Existen varios tipos de lenguaje de programación:

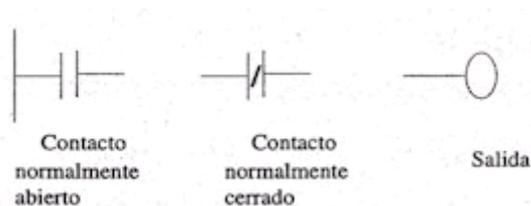
- Mnemónico o Lista de instrucciones.
- Esquema de contactos
- Esquema funcional

No obstante, los lenguajes de programación más empleados en la actualidad son, el mnemónico y el esquema de contactos.

4.2.1 CONTACTOS

Es un lenguaje gráfico, derivado del lenguaje de relés, que mediante símbolos representa contactos, solenoides... . Su principal ventaja es que los símbolos básicos están normalizados según normas NEMA y son empleados por todos los fabricantes.

Los símbolos básicos son:



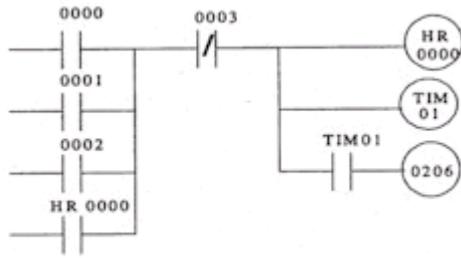
Los elementos básicos que configuran la función se representan entre dos líneas verticales que simbolizan las líneas de alimentación.

Para las líneas de función más complejas como temporizadores, registros de desplazamiento, etc, se emplea el formato de bloques. Estos no están formalizados, aunque guardan una gran similitud entre sí para distintos fabricantes y resultan mucho más expresivos que si se utiliza para el mismo fin el lenguaje en lista de instrucciones o mnemónico.

Conceptos generales

Un programa en esquema de contactos, la constituyen una serie de ramas de contactos.

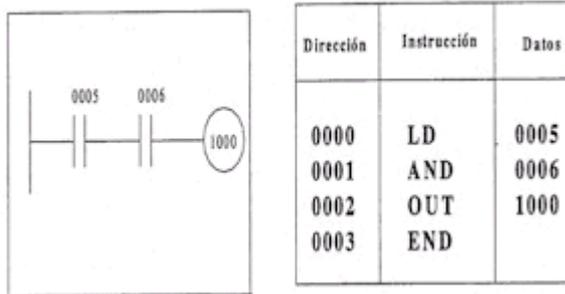
Una rama esta compuesta de una serie de contactos, conectados en serie o en paralelo que dan origen a una salida que bien puede ser una bobina o una función especial.



El flujo de la señal va de izquierda a derecha y de arriba abajo.

A una rama de circuito en esquema de contactos le corresponde una secuencia de instrucciones en forma mnemónica.

Todas las ramas de circuito se inician con una instrucción LOAD.



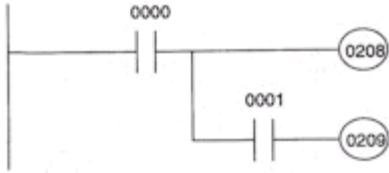
Una bobina no puede venir conectada directamente de la barra de inicio. En tal caso es necesario interponer un contacto siempre cerrado.



A la derecha de una bobina no es posible programar ningún contacto.

El número de contactos posibles en serie o en paralelo es prácticamente ilimitado.

Es posible colocar en paralelo 2 o más bobinas.



4.2.2 MNEMOTECNICO

Un lenguaje en mnemónico o lista de instrucciones consiste en un conjunto de códigos simbólicos, cada uno de los cuales corresponde a una instrucción.

Cada fabricante utiliza sus propios códigos, y una nomenclatura distinta para nombrar las variables del sistema.

El lenguaje en mnemónico es similar al lenguaje ensamblador del micro.

Ejemplo: La [lista de instrucciones](#) utilizada en este proyecto son las del autómeta CQM1H de OMRON.

Dirección	Instrucción	Parámetro
0000	LD	H0501

Instrucción: Especifica la operación a realizar.

Parámetro: Son los datos asociados a la operación (instrucción). Los parámetros son en general de formato TIPO y VALOR.

Dirección: Indica la posición de la instrucción en la memoria de programa usuario.

Las funciones de control vienen representadas con expresiones abreviadas.

La fase de programación es mas rápida que en el lenguaje de esquemas de contactos.

INFORMACION TOMADA DE:

<http://www.sc.ehu.es/sbweb/webcentro/automatica/WebCQM1H/PAGINA%20PRINCIPAL/PLC/plc.htm>