

## INDICE

### UNIDAD 1

<b>Introducción a los PLCs.....</b>	<b>5</b>
Objetivos.....	5
<b>PLC. DIAGRAMA DE BLOQUES.....</b>	<b>5</b>
Diagrama de bloques de un sistema con PLC.....	6
<i>Dispositivos de entrada.....</i>	<i>7</i>
<i>Dispositivos de salida.....</i>	<i>8</i>
<b>MODULO DE ENTRADA DEL PLC. CABLEADO EXTERNO DE LOS DISPOSITIVOS DE ENTRADA.....</b>	<b>9</b>
<i>Entradas DC.....</i>	<i>10</i>
Cableado de modulos de entrada DC.....	11
<i>Ejercicios.....</i>	<i>12</i>
Sensores.....	13
<i>Sensores NPN.....</i>	<i>14</i>
<i>Sensores PNP.....</i>	<i>15</i>
<i>Cableado.....</i>	<i>16</i>
<i>Ejercicios.....</i>	<i>19</i>
<b>MODULO DE SALIDA DEL PLC. CABLEADO EXTERNO DE LOS DISPOSITIVOS DE SALIDA.....</b>	<b>20</b>
<i>Salidas a relé.....</i>	<i>20</i>
<i>Salidas a transistor.....</i>	<i>20</i>
<i>Salidas a triacs.....</i>	<i>21</i>
Cableado.....	21
<i>Ejercicios.....</i>	<i>22</i>
<i>Cableado de cargas inductivas.....</i>	<i>23</i>

<b>ORGANIZACION DE LA MEMORIA. PRINCIPIOS DE DIRECCIONAMIENTO.....</b>	<b>24</b>
Direccionamiento de puntos E/S.....	24
Organización de la memoria y direcciones.....	25
<i>Area IR</i> .....	25
<i>Area SR</i> .....	26
<i>Area TR</i> .....	26
<i>Area HR</i> .....	27
<i>Area AR</i> .....	27
<i>Area LR</i> .....	27
<i>Area TC</i> .....	28
<i>Area DM</i> .....	28
<i>Area UM</i> .....	28
Direccionamiento de E/S en los PLCs C200HX/HG/HE.....	29
<i>Clasificación de los módulos de Entrada/Salida</i> .....	29

## **UNIDAD 2**

<b>Programación.....</b>	<b>32</b>
Objetivos.....	32
<b>PROCEDIMIENTO DE PROGRAMACION.....</b>	<b>32</b>
<b>PROGRAMACION BASICA EN DIAGRAMA DE ESCALERA.....</b>	<b>35</b>
Instrucciones LOAD/AND/OUT.....	36
<i>LOAD Y LOAD NOT</i> .....	36
<i>AND Y AND NOT</i> .....	37
<i>OUTPUT Y OUTPUT NOT</i> .....	38
Instrucción END (01).....	39
Instrucción OR.....	41
<i>OR Y OR NOT</i> .....	41
<i>Combinación de instrucciones AND Y OR</i> .....	42

Instrucciones de bloque lógico AND LOAD/OR LOAD.....	45
AND LOAD.....	45
OR LOAD.....	47
ACTUALIZACION DE E/S.....	50
CICLO DEL PLC.....	51
DUPLICACION DE SALIDAS.....	52
<i>Ejercicios de programación</i> .....	54
Reles temporales (TRs).....	56
Temporizadores/Contadores.....	59
<i>Temporizador (TIM)</i> .....	59
<i>Contador (CNT)</i> .....	61
<i>Contador Reversible (CNTR (12))</i> .....	67
<i>Temporizador de Alta Velocidad (TIMH (15))</i> .....	70
Instrucción KEEP (11).....	71
Instrucción DIFFERENTIATE UP- DIFU (13).....	73
Instrucción DIFFERENTIATE DOWN- DIFD (14).....	76
Instrucción INTERLOCK e INTERLOCK CLEAR- IL (02) e ILC (03).....	77
<i>Anidamiento de INTERLOCKS</i> .....	78
Instrucción JUMP y JUMP END- JMP (04) y JME (05 ).....	78
CONSOLA DE PROGRAMACION MANUAL.....	80
<i>Descripción</i> .....	80
Manejo de la Cónsola de Programación.....	81
<i>Teclas y sus funciones</i> .....	80
Inicialización y arranque del PLC.....	83
Escritura y Lectura de un programa.....	84
Búsqueda, Inserción y Borrado de un paso de programa.....	87
Monitoreo de estado, forzado a ON y forzado a OFF de E/S.....	89
Cambio del Valor Fijado (SV) de temporizadores.....	91
Cambio del Valor Fijado (SV) de contadores.....	92

<b>PLCs CPM1A/CQM1.....</b>	<b>93</b>
Direccionamiento de E/S en los PLCs Serie CPM1A.....	93
Direccionamiento de E/S en el PLC CQM1.....	95

## UNIDAD 1

### Introducción a los PLCs

#### Objetivos

Al completar esta unidad el estudiante estará en capacidad de:

- Nombrar aplicaciones industriales más comunes de un PLC
- Describir de qué elementos básicos consiste un sistema con PLC
- Diagramar y realizar el cableado externo de dispositivos eléctricos y electrónicos a los módulos de entrada y salida del PLC
- Conocer la organización de la memoria del PLC y los principios de direccionamiento

#### PLC. DIAGRAMA DE BLOQUES

Un PLC o Controlador Lógico Programable es un equipo integrado, capaz de tomar información del mundo externo, procesarla realizando operaciones lógicas y matemáticas, y ejecutar acciones programadas como respuesta.

Su uso es extensivo en el **control** de una gran variedad de procesos industriales de distinta magnitud y su aplicación va desde la automatización de máquinas de fabricación y líneas de ensamblaje en un proceso aislado, hasta aplicaciones que requieran comunicación en red de PLCs, ordenadores y otros dispositivos de control, permitiendo una integración y manejo total de la información en planta.

El potencial de los PLCs en el mejoramiento de los procesos industriales se basa fundamentalmente en las siguientes cualidades:

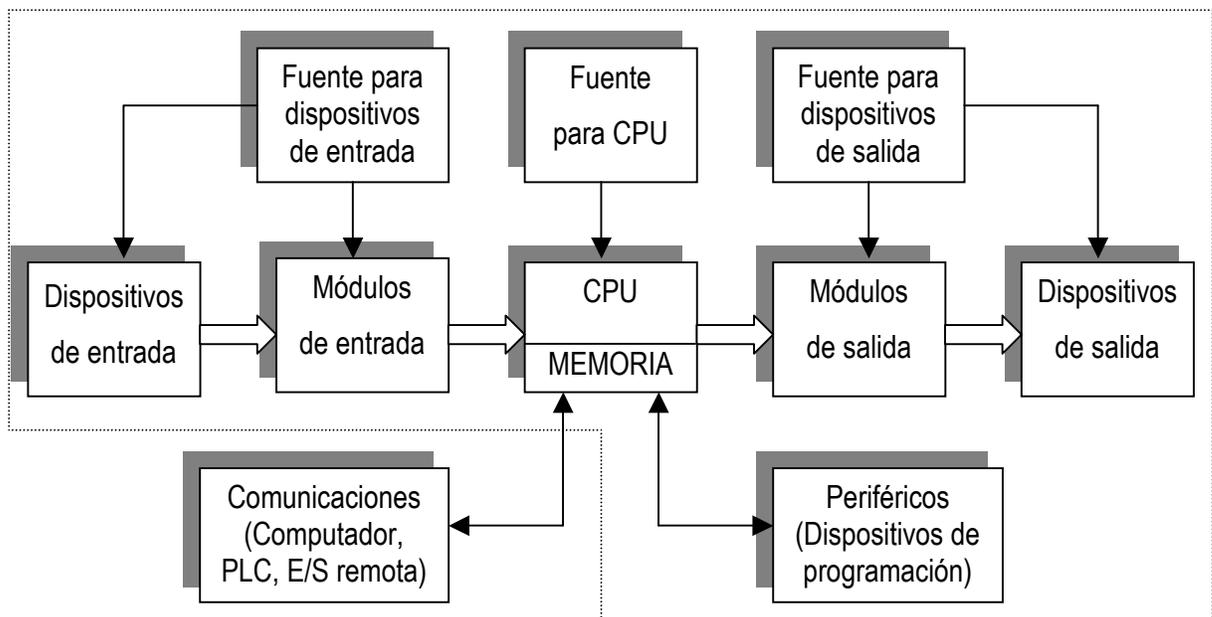
- ✓ Alta confiabilidad
- ✓ Alta integración

- ✓ Simplificación del cableado
- ✓ Mayor flexibilidad y funcionalidad en los procesos controlados
- ✓ Alta velocidad de respuesta del sistema
- ✓ Comunicación en red
- ✓ Bajo costo

## Diagrama de bloques de un sistema con PLC

Un sistema con PLC mínimo está basado en los siguientes elementos:

1. Unidad Central de Procesamiento (CPU)
2. Memoria
3. Fuente de Poder
4. Módulos de entrada
5. Módulos de salida
6. Dispositivos de entrada
7. Dispositivos de salida
8. Periféricos

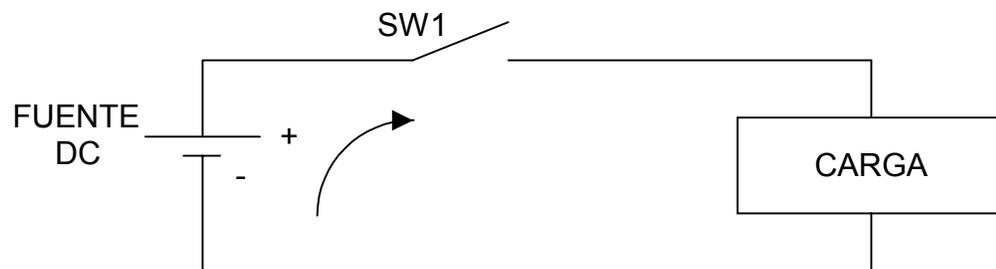


**Figura 1.1**

## Dispositivos de entrada

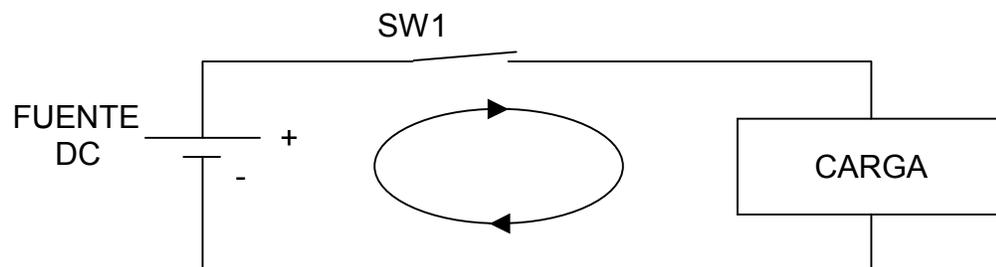
**DISCRETA:** Cuando la información que introduce es uno de dos valores posibles: ON u OFF (1 ó 0). Son dispositivos de entrada discreta los selectores, finales de carrera, pulsadores, sensores fotoeléctricos, etc.

En la figura 1.2, tan pronto como se abre el contacto la carga se apaga.



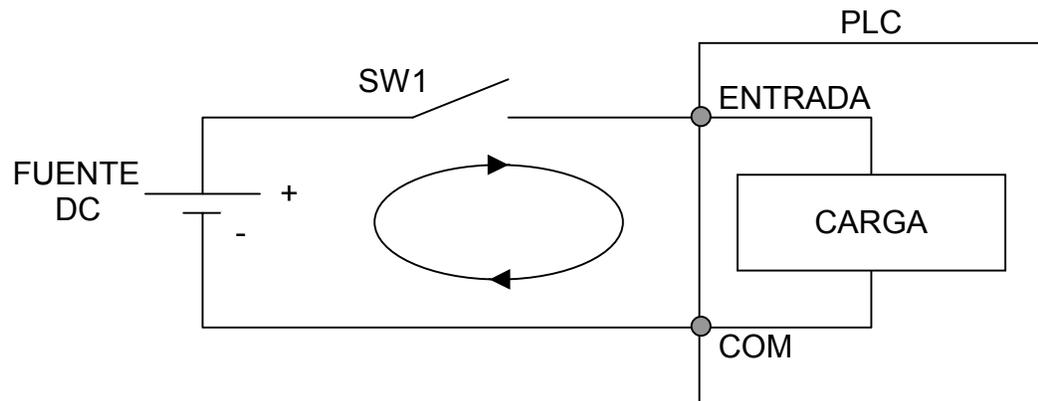
**Figura 1.2**

En la figura 1.3, cuando la llave se cierra, la corriente fluye y la carga se energiza.



**Figura 1.3**

En la figura 1.4, la carga se reemplaza por el circuito de entrada del PLC. Cuando se cierra el contacto, se activa la entrada del PLC.



**Figura 1.4**

**ANALÓGICA:** Cuando la información que introduce es un valor de un rango continuo de valores posibles: 4-20mA, 0-5Vdc, 0-10Vdc. Son dispositivos de entrada analógica las termocuplas, transmisores de presión, etc.

Nota: En este curso serán considerados sólo los dispositivos de entrada discreta.

### **Dispositivos de salida**

**DISCRETA:** Cuando admite uno de sólo dos estados posibles: ON u OFF. Son dispositivos de salida discreta los relés de control, contactores, solenoides, lámparas indicadoras.

**ANALÓGICA:** Cuando admite un valor dentro de un rango continuo de valores posibles. Son dispositivos de salida analógica los controladores de velocidad de motor, actuadores lineales, válvulas de control de flujo.

Nota: En este curso serán considerados sólo los dispositivos de salida discreta.

## MODULO DE ENTRADA DEL PLC. CABLEADO EXTERNO DE LOS DISPOSITIVOS DE ENTRADA

El módulo de entrada es una interfaz, la cual recupera información del mundo real a través de los dispositivos de entrada y la convierte en datos para la CPU. Para convertir estos datos a los niveles de voltaje requeridos por la CPU se utiliza acopladores ópticos.

En la figura 1.5 realizar el cableado de los switches y fuente a las entradas ópticas.

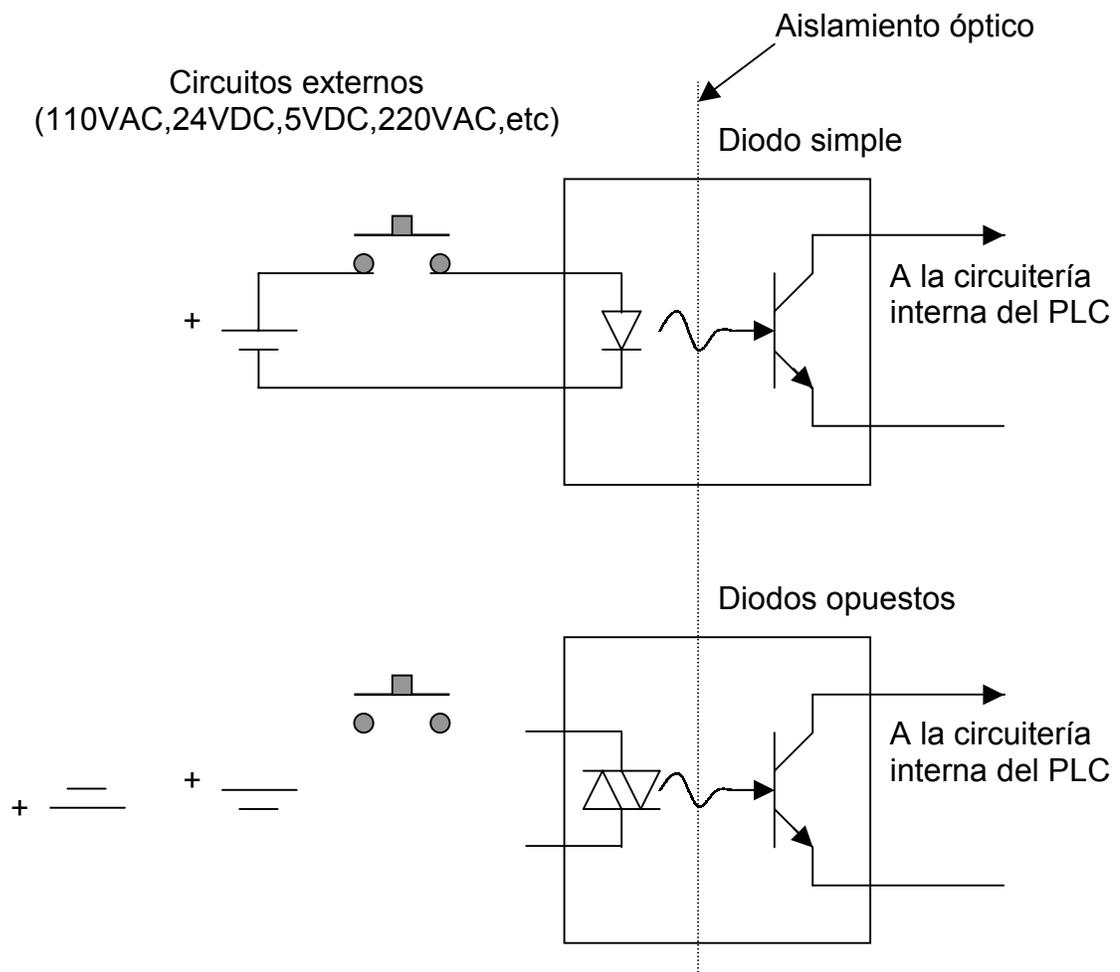


Figura 1.5

Los módulos de entrada estándar pueden ser divididos en tres tipos generales:

1. ID- Entradas DC
2. IA- Entradas AC
3. IM- Entradas AC/DC, 24V solamente

### **Entradas DC:**

Existe una amplia variedad de módulos de entrada DC disponibles según el número de puntos de entrada, requerimientos de voltaje y tipo de conexiones.

Las siguientes características de las entradas DC las convierte en la mejor elección para muchas aplicaciones:

- ✓ Se trabajan en niveles de voltaje y corriente relativamente seguros (5-24Vdc típicamente)
- ✓ Ofrece tiempos de respuesta más rápidos (1.5 ms típicamente)
- ✓ Los requerimientos del cableado de campo son más flexibles y menos estrictos que en sistemas de 120 Vac
- ✓ Se conectan fácilmente a dispositivos electrónicos como sensores fotoeléctricos, de proximidad y de fibra óptica
- ✓ Los módulos de entrada DC se ofrecen en altas densidades, llegando a 32 y 64 puntos en algunos estilos de PLCs

## Cableado de módulos de entrada DC

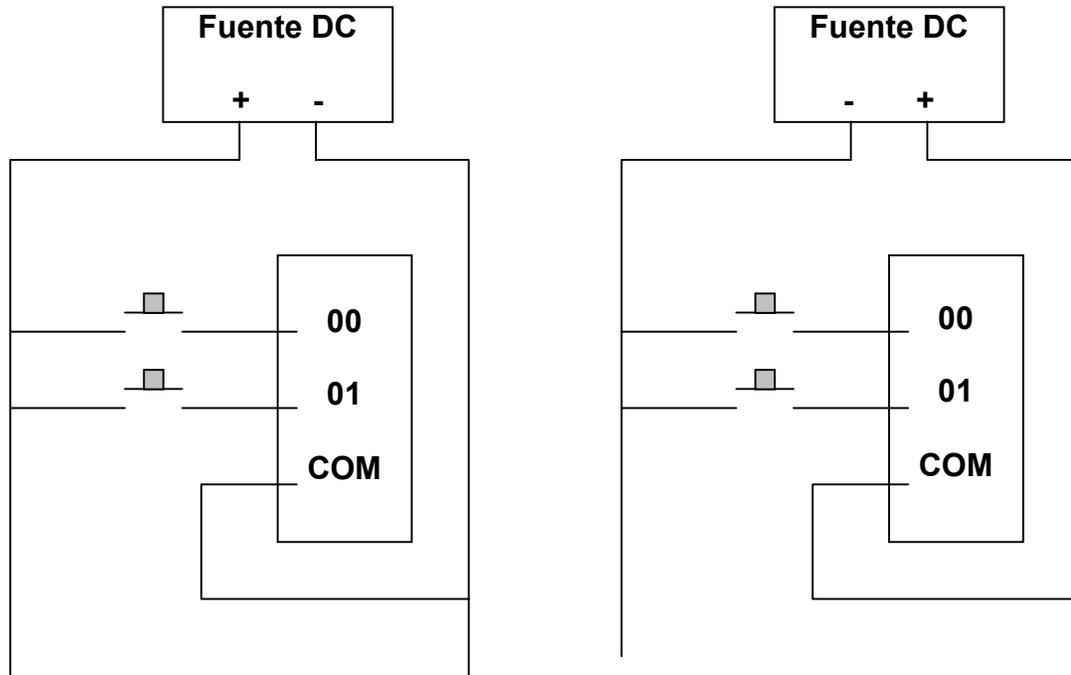


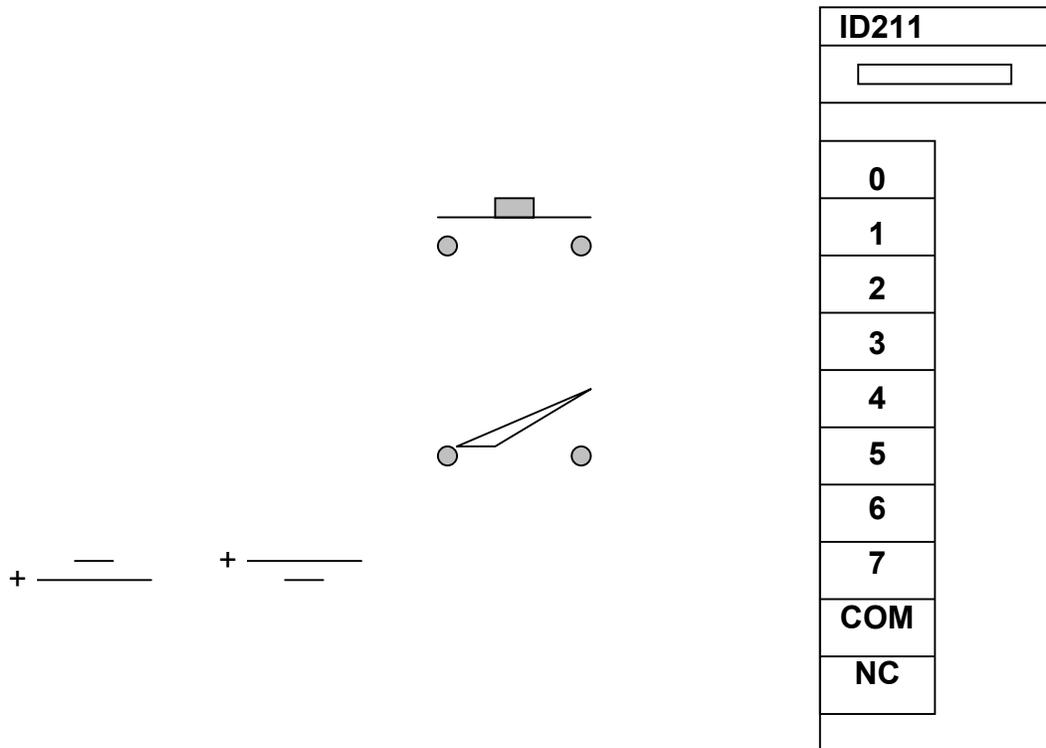
Figura 1.6

### Conceptos fundamentales:

1. La fuente de alimentación suele ser externa. Se pueden usar los 24Vdc generados por el propio PLC.
2. El terminal común (**COM**) del módulo está internamente conectado (en común) con los circuitos del PLC. Algunos módulos requieren un común (+), otros (-), y algunos aceptan (+) Y (-)
3. Los terminales del módulo que no tienen función y que no están conectados se designan con "**NC**", o sea, "NO CONNECTION"
4. Cada entrada tiene asignado un número, empezando por 0 (cero)

## Ejercicios

En la figura 1.7, realice el cableado externo del pulsador y del final de carrera, a las entradas 0 y 4 respectivamente.



**Figura 1.7**

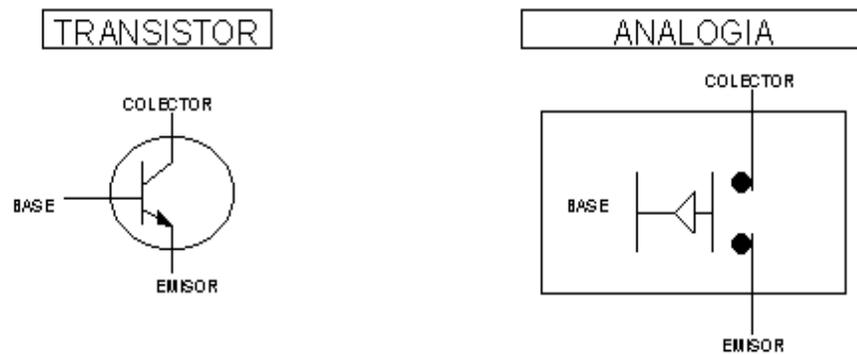
Nota: 1. Los optoacopladores tienen diodos opuestos

**2. El común puede conectarse al lado (+) o (-) de la fuente**

3. La fuente puede ser de 12 a 24 VDC

## Sensores

Los sensores pueden ser imaginados como una forma de conmutador con tres conexiones: colector, base y emisor. En la figura 1.8 se muestra la analogía eléctrica donde una fuerza aplicada a la base (acción de sensado) mueve el contacto que cierra el circuito de potencia, es decir, que hace fluir corriente entre colector y emisor.



El voltaje de Base debe ser más alto que el voltaje de emisor para que el conmutador "cierre"

**Figura 1.8**

## Sensores NPN

Un principio electrónico controla la conmutación del sensor. En la analogía de la figura 1.9 se ve que cuando el contacto se cierra, fluye corriente a través de la carga y luego a través del conmutador. Esto es un dispositivo que drena corriente.

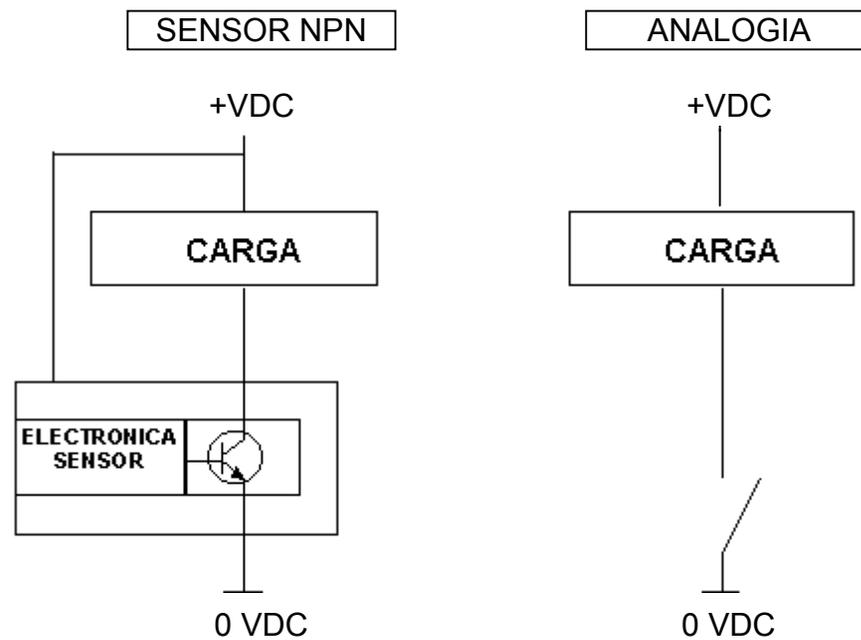


Figura 1.9

## Sensores PNP

La posición de la carga cambia con los sensores PNP. En la analogía de la figura 1.10, el sensor PNP queda arriba, es decir la corriente fluye a través del sensor y luego a través de la carga. Significa que el dispositivo es fuente de corriente.

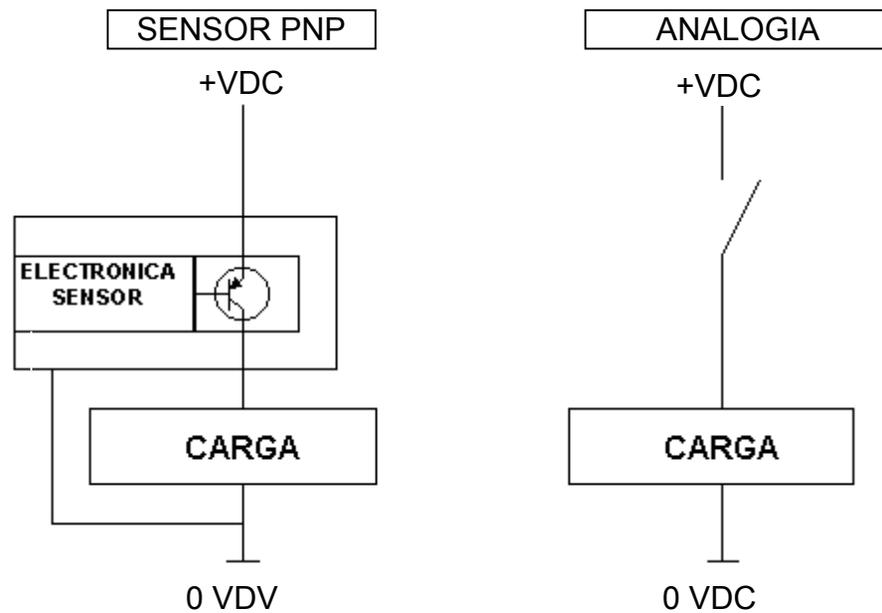


Figura 1.10

## Cableado

El cableado de sensores es ligeramente más complejo que el cableado de un límite de carrera, ya que los sensores requieren fuente de alimentación para operar la electrónica interna. Los sensores DC son de 3 hilos y se codifican por color como sigue:

<b>Código EEUU/JAPON</b>	<b>Código EUROPA</b>	
Rojo	Marrón	12-24 Vdc
Negro	Azul	0 Vdc
Blanco	Negro	Salida del sensor

Cuando se cablea, siempre hay que conectar los cables de alimentación a la fuente. La salida del sensor se conecta a la entrada del PLC, y hay una sola decisión que tomar y es cuál será el común en el circuito de entrada.

### Cableado de sensores NPN

Los sensores NPN requieren que la entrada del PLC tenga un común (+) para un cableado simplificado.

En la figura 1.11, realice el cableado del sensor NPN a la fuente y la entrada del PLC (Asistido por el Instructor).

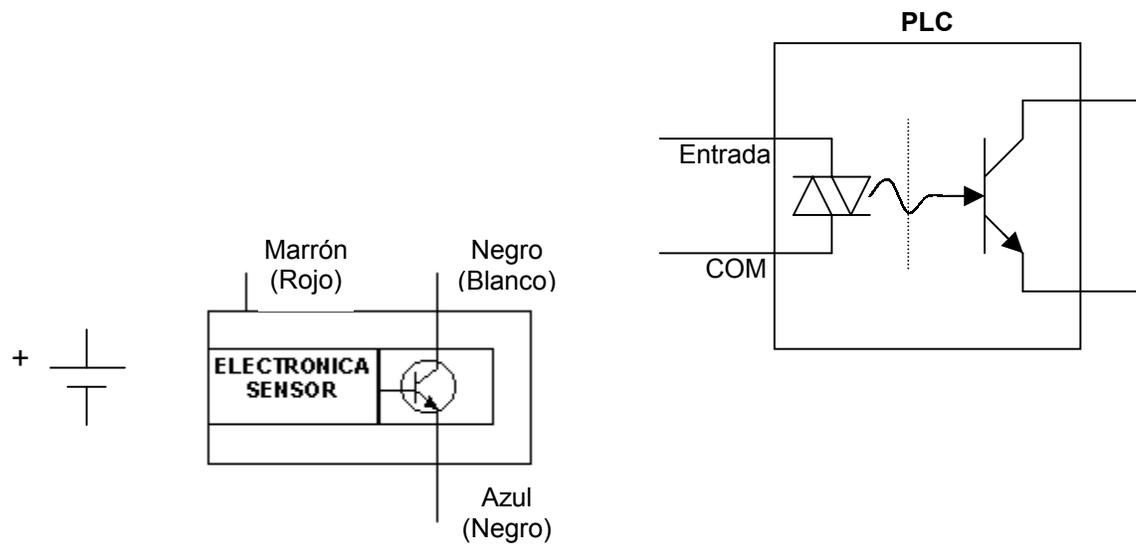


Figura 1.11

### Cableado de sensores PNP

Los sensores PNP requieren que la entrada del PLC tenga un común (-) para un cableado simplificado.

En la figura 1.12, realice el cableado del sensor PNP a la fuente y la entrada del PLC (Asistido por el Instructor).

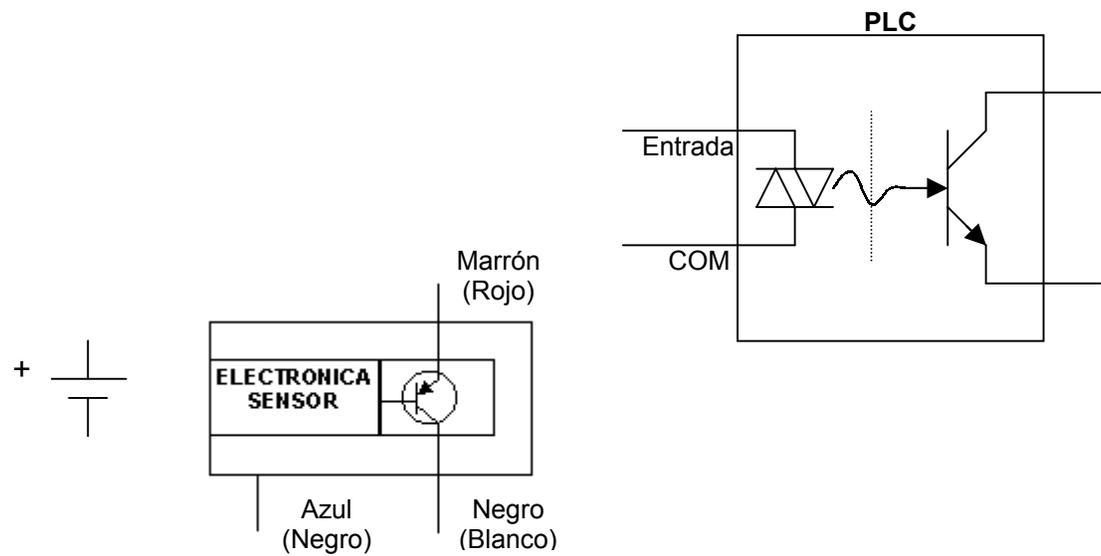


Figura 1.12

## Ejercicios

1. Realice el cableado del sensor NPN a la entrada 11.

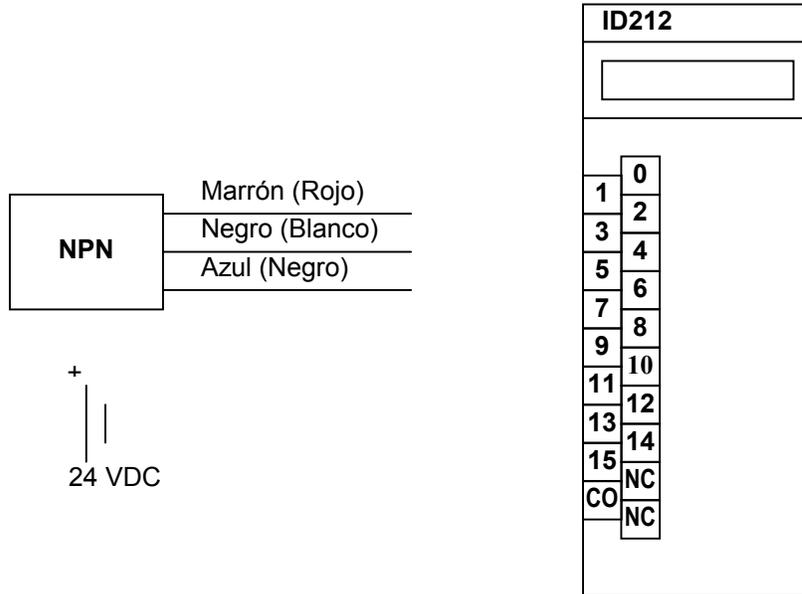


Figura 1.13

2. Realice el cableado del sensor PNP a la entrada 5.

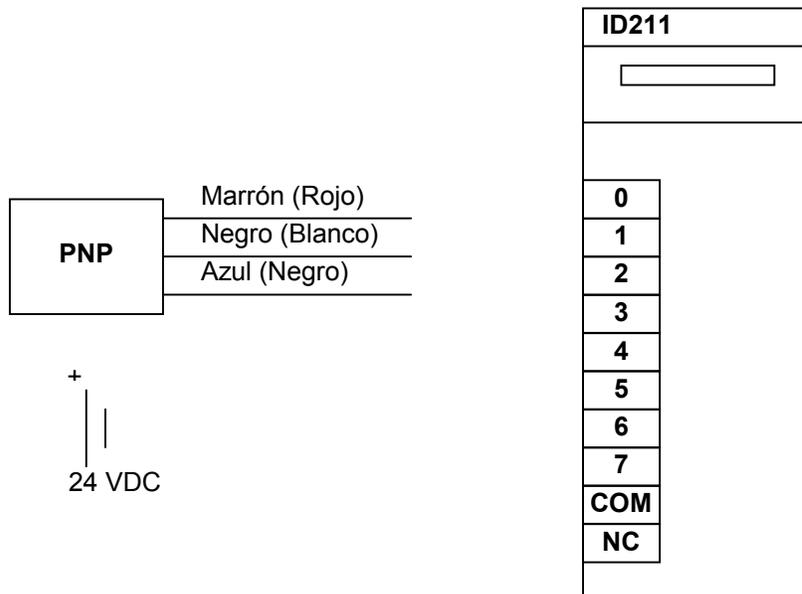


Figura 1.14

## MODULO DE SALIDA DEL PLC. CABLEADO EXTERNO DE LOS DISPOSITIVOS DE SALIDA

Una vez que el programa del usuario ha procesado todos los datos de entrada, la CPU envía datos a los dispositivos de salida a través de las **MODULOS DE SALIDA**. Los módulos de salida convierten las señales de la CPU a los niveles de voltaje que pueden ser usados por los dispositivos de salida (lámparas, solenoides, relés, etc.)

Los módulos de salida están disponibles en tres versiones:

- |       |                               |        |
|-------|-------------------------------|--------|
| 1. OC | Relé de contacto              | AC(DC) |
| 2. OD | Transistor                    | DC     |
| 3. OA | Triac (relé de estado sólido) | AC     |

### ***Salidas a relé***

Están disponibles en versiones de 16 y 32 puntos. En algunos PLCs el módulo está construido de tal manera que los relés sean reemplazables si alguno llegara a fallar. Las salidas a relé son la mejor opción cuando:

1. Se requieren corrientes de 2 Amps.
2. No se tolera ninguna corriente de pérdida
3. La salida cambia con baja frecuencia
4. Se deben controlar tensiones AC/DC no estándares
5. Se acepta un tiempo de respuesta de 15 ms
6. La tensión de AC tiene picos y transitorios

### ***Salidas a transistor***

Hay un espectro muy amplio de salidas a transistor, las cuales están disponibles en módulos de 16 a 64 salidas, y con capacidades de carga desde 300mA hasta 1 A. Hay disponibles salidas NPN y PNP. Las salidas a transistor son preferibles por las siguientes razones:

1. Tienen una alta velocidad de respuesta (0.2ms típico)
2. Pueden conectarse directamente a equipos electrónicos
3. Los niveles de tensión son mucho más seguros que los de AC

### **Salidas a triacs**

Los módulos de salida a triacs pueden ser la mejor opción cuando:

1. Se requieren cargas de hasta 1 Amp
2. La salida cambia muy frecuentemente

Las salidas a triacs deben ser usadas con más precaución que las salidas a relé por tres razones fundamentales:

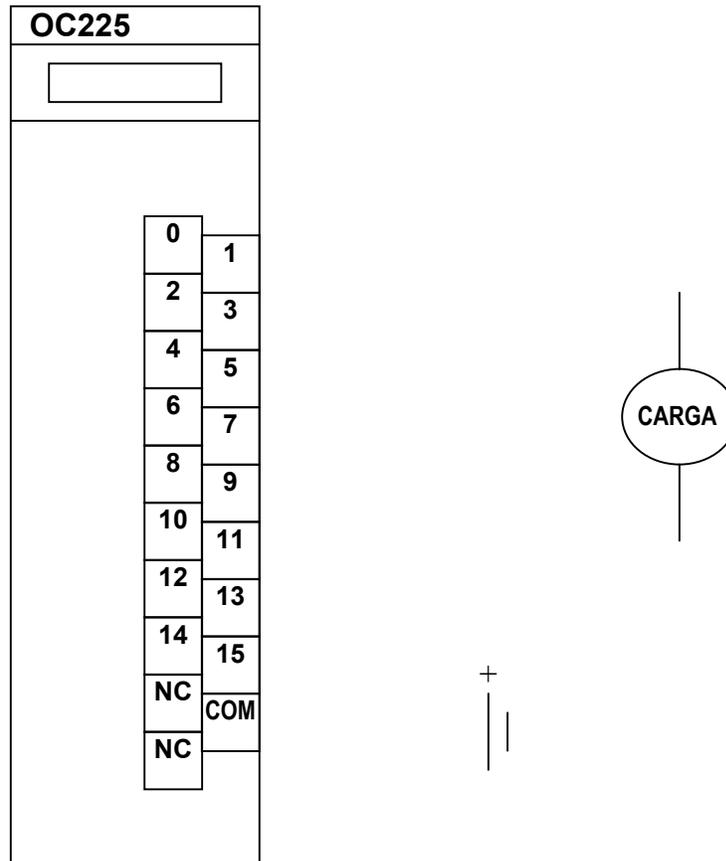
- Los triacs tienen una corriente de pérdida considerable en estado OFF, por lo cual algunos dispositivos podrían inadvertidamente energizarse
- Los picos y transitorios de la fuente podrían hacer que la salida vaya a ON indeseadamente
- Los triacs requieren una carga mínima (corriente de mantenimiento) para operar confiablemente

### **Cableado**

El entendimiento de las especificaciones de los módulos de salida es muy importante para determinar el correcto esquema de cableado para un módulo en particular. Los módulos de salida están disponibles en gran variedad con diferentes tipos de borneras, voltaje y corriente, requerimientos de densidad, etc., que es muy difícil establecer reglas generales. En este curso se tratarán las conexiones de las unidades de salida a relé.

## Ejercicios

Realice el cableado del dispositivo de carga a la salida 0 del PLC (Asistido por el Instructor).



**Figura 1.15**

- Notas:
1. Las salidas a relés pueden manejar hasta 24 VDC/250VAC
  2. Se requiere fuente de alimentación externa
  3. Chequee los requerimientos de corriente cuidadosamente.  
Por ej.: En el OC225 cada salida puede manejar hasta 2 Amp, pero el módulo puede manejar un máximo de 8 Amp
  4. Vea la guía de instalación para tomar las precauciones de cableado de entradas y salidas

## Cableado de cargas inductivas

Las cargas inductivas son aquellos dispositivos que pueden contener o actuar como bobinas. Los relés de control, solenoides, contactores, motores, etc., son cargas inductivas. Cuando la salida de estos dispositivos se apaga, el campo magnético almacenado en las bobinas se transforma en una fuerza electromotriz contraria, o voltaje, que puede llegar a 10 veces el voltaje de la fuente. Este pico de tensión puede acortar dramáticamente la vida de los relés o los transistores de salida. El uso de supresores de pico y diodos es una aconsejable manera de proteger las salidas.

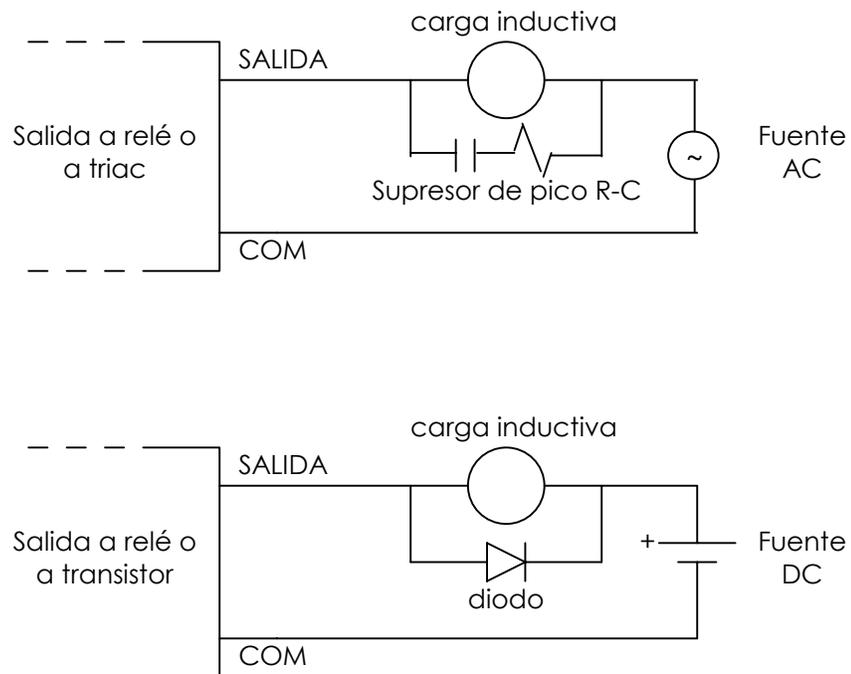


Figura 1.16

## ORGANIZACION DE LA MEMORIA. PRINCIPIOS DE DIRECCIONAMIENTO

### Direccionamiento de puntos E/S

Todos los PLCs usan un número para identificar un punto de E/S (Entrada/Salida) y es denominado dirección.

Cada dispositivo del mundo real es reconocido en el programa del PLC por su dirección.

Las direcciones referencian a dos partes: el **canal** y el **bit**. El canal es una celda o grupo de bits que pueden ser manipulados como una unidad. En los PLCs OMRON un canal es una palabra (en inglés word), es decir, un grupo de 16 bits de longitud.

La dirección de un punto E/S es un número de 5 dígitos. Los tres primeros dígitos identifican el canal, y los dos siguientes identifican el bit. Así por ejemplo un punto de entrada con la dirección 00011 está asignado al canal 000 y bit 11. Un punto de salida con la dirección 00104 está asignado al canal 001 y bit 04.

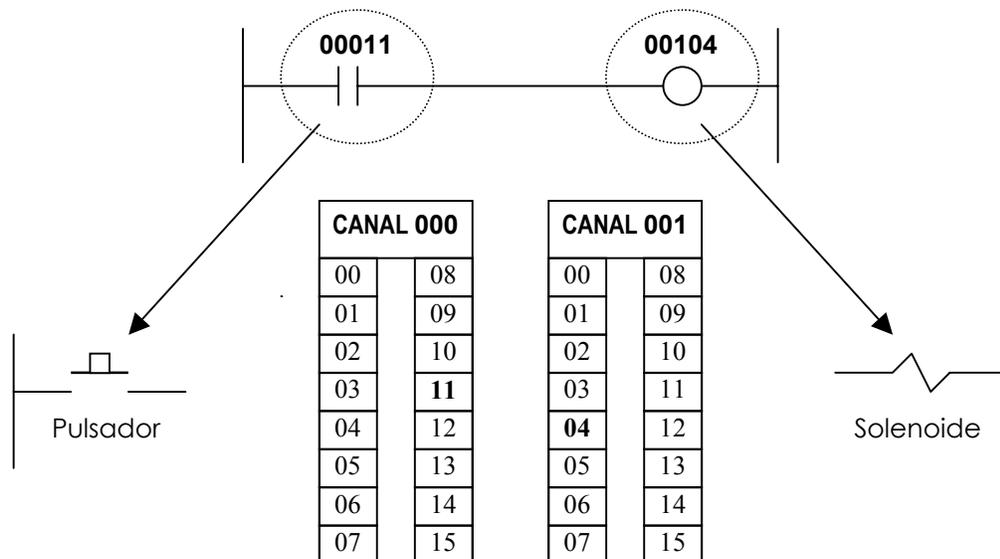


Figura1.17

## Organización de la memoria y direcciones

La CPU requiere instrucciones para procesar todos los datos que están siendo introducidos, y requiere, además, un área para almacenar estos datos. La **MEMORIA** proporciona esta área y es en ella que el programa del usuario es almacenado.

La memoria de los PLCs está organizada por áreas de funciones diferenciadas y claramente definidas. La estructura de la memoria de los PLCs es conocida como matriz de memoria, mapa de memoria o mapa E/S. La matriz se divide en secciones o áreas con el fin de que sea más manejable para el usuario. La mayoría de las áreas tienen un prefijo tal como HR, T, DM y un número de 3 o 4 dígitos para definir su localización específica en la memoria. Por ejemplo, el temporizador 0 (Timer 0) será referenciado como T000, el relé de mantenimiento 0 (Holding Relay 0) como HR0000 y la memoria de datos 0 (Data Memory 0) como DM0000.

Todos los PLCs OMRON tienen las siguientes áreas dentro de su matriz de memoria:

### **Area IR (relés internos)**

Las funciones de los bits de ésta area son:

**Area de entradas y salidas:** Son bits del área IR designados a los módulos de entradas y salida conectados al PLC. Ellos reflejan el estado ON/OFF de las señales de entrada y salida del mundo real.

**Area de trabajo:** Son bits que pueden ser usados libremente en el programa del PLC para almacenar un estado. Los bits de trabajo son reseteados (es decir, llevados a OFF) cuando la fuente de poder del PLC es apagada, o cuando la operación del PLC comienza o se detiene.

Los bits restantes del área IR tienen funciones específicas. La mayoría de éstos, pero no todos, pueden ser usados como bits de trabajo cuando la función específica no está siendo usada.

Los bits del área IR tienen direcciones con prefijo IR seguido de un número de 5 dígitos; los tres primeros dígitos indican el canal y los dos siguientes el bit, así, la dirección IR00205 referencia al bit 5 del canal 2 (direccionamiento por bit). Si la dirección se señala con sólo tres dígitos, se estará haciendo referencia a un canal completo, por ejemplo, IR300 referencia a todos los bits del canal 300 (direccionamiento por canal).

### **Area SR (relés especiales)**

Los bits de esta área funcionan esencialmente como banderas (en inglés flags) relacionadas con la operación del PLC, algunas de ellas son:

- Pulsos de reloj.
- Bandera de acarreo.
- Bandera de error.
- Bandera alarma de batería.

Esta area de memoria es direccionada en forma similar al área IR.

Nota: Las áreas de memoria IR y SR pueden prescindir de los prefijos IR y SR respectivamente al ser direccionadas.

### **Area TR (relés temporales)**

Estos bits son usados para almacenar temporalmente condiciones ON/OFF de ejecución en las ramificaciones de un diagrama en escalera complejo. Son útiles sólo cuando se requiere programar un diagrama de escalera en código mnemónico.

Todos los PLCs tienen 8 relés temporales identificados TR0 hasta TR7 (TR es el prefijo para acceder al área).

### **Area HR (relés de mantenimiento)**

El área HR es usada para almacenamiento y manipulación de datos internos. Esta área de memoria es retentiva, es decir, retiene el estado ON/OFF de los bits aún luego de que la fuente de poder del PLC se haya apagado o cuando la operación del mismo comienza o se detiene. Ella es usada en la misma forma que el área de trabajo.

El área de memoria HR puede ser accesada por canal (HR00) o por bit (HR0012).

### **Area AR (relés auxiliares)**

Similarmente al área SR, los bits del área AR funcionan principalmente como banderas relacionadas con la operación del PLC, algunas de ellas son:

- Tiempo de ciclo largo
- Contador de potencia-off
- Banderas de diagnósticos

Con pocas excepciones, el estado de los bits y words AR son refrescados cada ciclo. Esta área de memoria es retentiva.

El área AR puede ser accesada por canal (AR11) o por bit (AR1100).

### **Area LR (relés de enlace)**

El área LR es usada para almacenamiento interno de datos enlazados o compartidos entre dos o más PLCs.

Los bits LRs pueden ser usados como bits de trabajo cuando no están siendo usados para enlaces de datos.

Esta área de memoria no es retentiva, al igual que el área HR y AR puede ser direccionada por canal (LR00) o por bit (LR0010).

### **Area TC (temporizadores/contadores)**

Esta área está dedicada para uso de temporizadores (TIM), temporizadores de alta velocidad (TIMH), contadores (CNT) y contadores reversibles (CNTR). Ya que temporizadores y contadores ocupan la misma área, un contador no puede ser especificado con el mismo número que un temporizador. Por ejemplo: CNT 010 y TIM 010 no pueden ser usados a la vez.

Los registros de temporizadores/contadores son memoria de tipo retentiva. Los valores fijados SV (descrito con detalle más adelante) para temporizadores y contadores son retenidos aún si la potencia falla. El valor presente PV de los contadores es conservado en caso de que la potencia falle, no así para los temporizadores.

### **Area DM (memoria de datos)**

Es usada para manipulación y almacenamiento de datos. Esta área no puede ser accesada por bit, sino en unidades de palabra. Cierta zona del área DM es destinada a la configuración del PLC.

El área de memoria DM es retentiva.

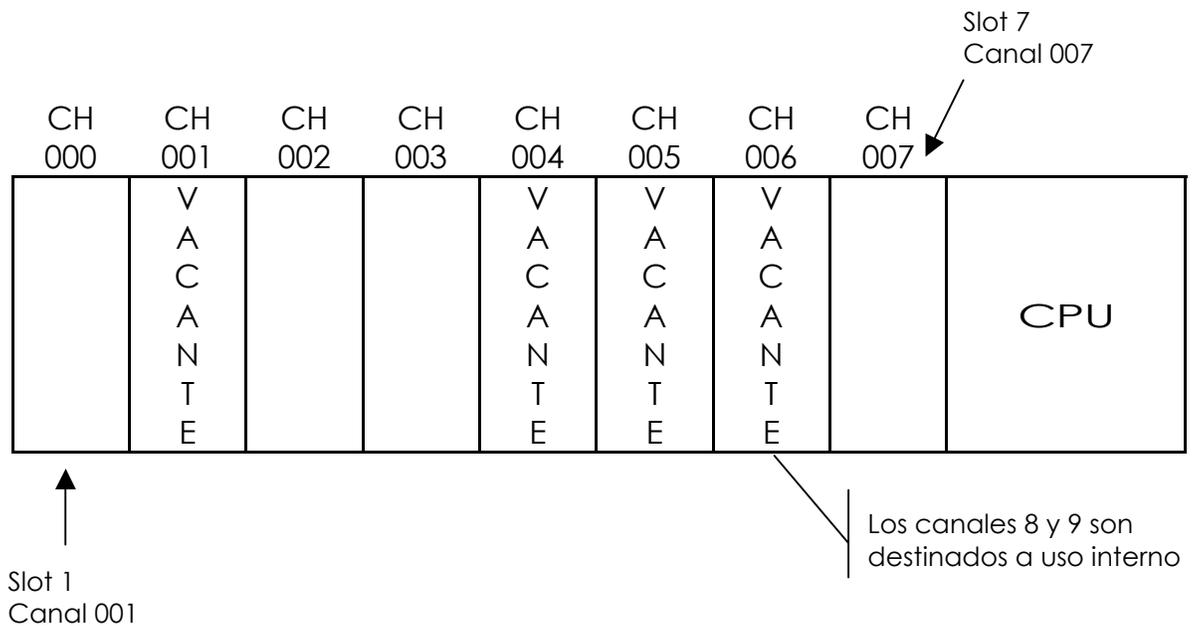
### **Area UM (memoria de programa)**

Usada para almacenar el programa del usuario. Las direcciones en la memoria de programa son ligeramente diferentes que aquellas en las otras áreas de memoria, ya que cada dirección no necesariamente contiene la misma cantidad de datos. Cada dirección mantiene una instrucción y todos los definidores y operandos requeridos por esa instrucción (descrito con detalle más adelante). Ya que algunas instrucciones no requieren operandos, mientras que otras requieren hasta tres operandos, una dirección de la memoria de programa puede ser de una a cuatro palabras de largo. Esta área de memoria es retentiva.

## Direccionamiento de los PLCs C200HX/HG/HE

Los PLCs C200HX/HG/HE pertenecen a la Serie-Alfa de Omron, cada uno de estos PLCs es un sistema de racks con 3, 5, 8 o 10 slots disponibles. Cada rack tiene asignado 10 canales, y un sistema puede soportar hasta dos racks de expansión (o tres, dependiendo del modelo de la CPU). El direccionamiento de canales se inicia desde la izquierda en el rack de la CPU y progresa hacia la derecha. El direccionamiento es FIJO, es decir, cada slot del backplane tiene un canal correspondiente (ver figura 1.18). Si se deja un slot vacante, el canal queda reservado para ese slot, y está disponible para el agregado futuro de un módulo.

Ventaja: Agregar un módulo no modifica el direccionamiento



**Figura 1.18**

## Clasificación de los Módulos de Entrada/Salida

Los módulos de E/S están divididos en tres grupos: módulos estándares, módulos de alta densidad y módulos especiales.

**Módulos estándares:** Requieren sólo un canal (word) para transferir datos. Los módulos de 8 y 16 puntos son considerados ESTANDARES.

**Dirección:** El módulo estándar utiliza el canal correspondiente al slot que está ocupando (Ver tabla).

Rack	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10
CPU	IR 000	IR 001	IR 002	IR 003	IR 004	IR 005	IR 006	IR 007	IR 008	IR 009
1ª Expa	IR 010	IR 011	IR 012	IR 013	IR 014	IR 015	IR 016	IR 017	IR 018	IR 019
2ª Expa	IR 020	IR 021	IR 022	IR 023	IR 024	IR 025	IR 026	IR 027	IR 028	IR 029
3ª Expa	IR 030	IR 031	IR 032	IR 033	IR 034	IR 035	IR 036	IR 037	IR 038	IR 039

**Módulos de alta densidad:** Diseñados para ofrecer alta capacidad de E/S. Requieren más de un canal para transferir datos. Los módulos de 32 puntos (2 canales) y 64 puntos (4 canales), son considerados DE ALTA DENSIDAD.

**Dirección:** Los módulos de alta densidad ocupan los canales entre IR 030 e IR 049 de acuerdo al número especificado para el módulo (Ver tabla).

Módulos de 32 pts.		Módulos de 64 pts.	
Número	Canales	Número	Canales
0	IR 030 a IR 031	0	IR 030 a IR 033
1	IR 032 a IR 033	1	IR 032 a IR 035
2	IR 034 a IR 035	2	IR 034 a IR 037
3	IR 036 a IR 037	3	IR 036 a IR 039
4	IR 038 a IR 039	4	IR 038 a IR 041
5	IR 040 a IR 041	5	IR 040 a IR 043
6	IR 042 a IR 043	6	IR 042 a IR 045
7	IR 044 a IR 045	7	IR 044 a IR 047
8	IR 046 a IR 047	8	IR 046 a IR 049
9	IR 048 a IR 049	9	No puede ser usado

**Módulos Especiales:** Diseñados para realizar una tarea específica. Dentro de esta clasificación se incluyen los módulos Conversores A/D, D/A, Sensores de Temperatura, Posicionadores, Contadores de Alta Velocidad, etc. Necesitan más de un canal para transferir datos, hasta un máximo de 10 canales.

**Dirección:** La tabla siguiente especifica la localidad de memoria que utiliza la unidad de acuerdo al número especificado para la misma (de 0 a 9, A a F).

Número	Canales	Restricciones
0	IR 100 a IR 109	
1	IR 110 a IR 119	
2	IR 120 a IR 129	
3	IR 130 a IR 139	
4	IR 140 a IR 149	
5	IR 150 a IR 159	
6	IR 160 a IR 169	
7	IR 170 a IR 179	
8	IR 180 a IR 189	
9	IR 190 a IR 199	
A	IR 400 a IR 409	No disponibles para los PLCs C200HE-CPU -E y C200HG/HX- CPU3 -E/4 -E
B	IR 410 a IR 419	
C	IR 420 a IR 429	
D	IR 430 a IR 439	
E	IR 440 a IR 449	
F	IR 450 a IR 459	

**Nota:** Un máximo de 16 módulos especiales pueden ser montados en un sistema.

En la tabla siguiente se indican las direcciones de las áreas de memoria disponibles por el usuario:

PLC	Area E/S	Bits de Trabaj.	Area SR	Area TR	Area HR	Area AR	Area LR	Area TC	Area DM
C200HX /HG/HE	000-029	232-235	236-299	0-7	HR00-	AR00-	LR00-	TIM/CNT 000-511	DM0000- DM6143
	300-309	310-329			HR99	AR27	LR63		
		342-349							
		460-511							

## UNIDAD 2

### Programación

#### Objetivos

Al completar esta unidad el estudiante estará en capacidad de:

- Explicar los pasos que deben realizarse en el desarrollo de un programa
- Crear un diagrama de escalera para reflejar el orden de eventos de entrada y la(s) salida(s) resultante, usando operaciones lógicas e instrucciones básicas
- Escribir el listado de mnemónicos de una representación en escalera
- Manejar la consola de programación para:
  - Introducir y verificar el programa en el PLC
  - Monitorear y cambiar el estado de E/S
  - Cambiar valores de temporizadores y contadores

#### PROCEDIMIENTO DE PROGRAMACION

El desarrollo de un programa envuelve la realización de algunos pasos básicos, ello son:

1. **DETERMINAR LOS OBJETIVOS DEL SISTEMA DE CONTROL:** El paso más importante en el desarrollo de un programa es determinar qué debe hacer el sistema de control y en qué orden las tareas deseadas deben ser cumplidas.
2. **ASIGNAR LOS DISPOSITIVOS DE ENTRADA/SALIDA:** Una vez que las tareas han sido definidas y se han determinado los dispositivos de entrada/salida necesarios para realizarlas, deben asignarse estos dispositivos a puntos de entrada/salida del PLC, esto es, asignar una dirección particular a cada entrada y salida en la forma de un número de 5 dígitos.

3. **CREAR EL DIAGRAMA DE ESCALERA:** Un diagrama de escalera es una representación gráfica de un programa de PLC. Consiste de dos líneas verticales espaciadas simbolizando los soportes de una escalera y representan cables de potencia o BUSES. Todos los circuitos conmutadores (contactos de relés) con los que se construyen CONDICIONES DE EJECUCION parten del bus izquierdo que es la línea viva de potencia. Todas las INSTRUCCIONES a ejecutar se encuentran del lado derecho que es la línea de tierra o neutro. Los peldaños de la escalera son circuitos horizontales llamados LINEAS DE INSTRUCCIÓN (en inglés rungs) que se inician desde el bus izquierdo con contactos de relés pasando a través de varios de éstos y finalizando en el bus derecho con la instrucción. El diagrama de escalera es preparado secuencialmente en el orden que la acción debe ocurrir.

### Ejemplo

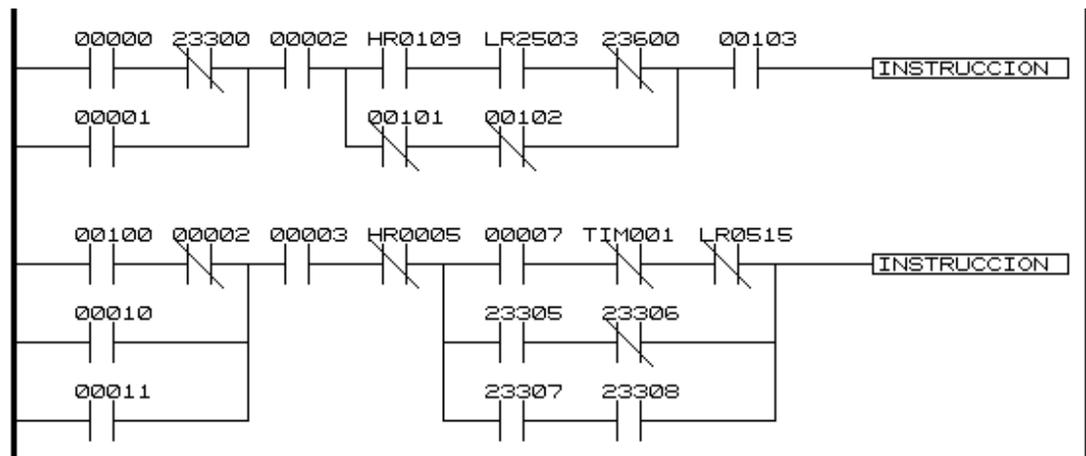


Figura 2.1

4. **INTRODUCIR EL PROGRAMA EN LA CPU:** Después que se ha creado el diagrama de escalera, el mismo debe ser convertido a un lenguaje que el PLC pueda usar. Este lenguaje consiste de **instrucciones** y

**operandos** estos últimos se refieren a direcciones y datos. La introducción del programa en la CPU se puede hacer vía Consola de Programación o vía Software gráfico (LSS/SSS, CPT o SISWIN). Cuando se introduce el programa vía consola se hace necesario convertir el diagrama de escaleras al lenguaje o **código de mnemónicos**, en los otros casos el software se encarga de la conversión, pero indistintamente de la vía, el programa es siempre almacenado en memoria en forma de mnemónicos.

5. **CHEQUEAR EL PROGRAMA:** Revisar el programa para corregir, si existen, errores de sintaxis
6. **EJECUTAR EL PROGRAMA:** Poner a prueba el programa para corregir, si existen, errores de ejecución
7. **INSTALAR EL SISTEMA:** Instalar el sistema de control completo, ejecutar el programa y realizar el ajuste fino si lo requiere.

## PROGRAMACION BASICA EN DIAGRAMA DE ESCALERA

En la figura 2.2 se muestra el circuito eléctrico emulado en un diagrama de escalera.

Nota: En los ejemplos y ejercicios siguientes se asume un PLC C200HG con entradas en el canal 000 y salidas en el canal 001. Si usted usa otro PLC debe asegurarse de direccionar en los rangos correspondientes.

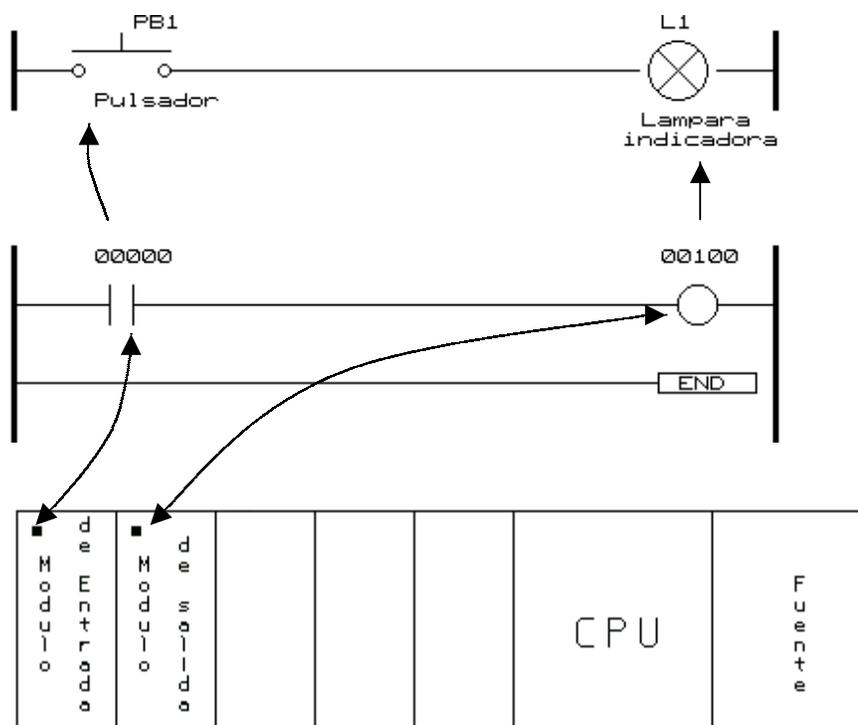


Figura 2.2

### Diagrama eléctrico

En la representación eléctrica, cuando PB1 es pulsado, la lámpara L1 enciende. Tanto tiempo como PB1 está en ON, la lámpara L1 está en ON.

### Diagrama de escalera

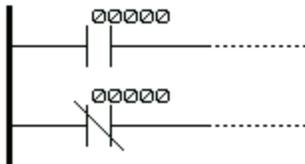
El pulsador es un dispositivo de entrada y está conectado al módulo de entrada, canal 000 bit 00. La lámpara es un dispositivo de salida y está conectado al módulo de salida, canal 001 bit 00. Las direcciones usadas en el diagrama de escalera informan al PLC qué entradas son usadas para operar cuáles salidas. Tal como el programa está escrito, sólo la entrada 00000 opera sobre la salida 00100.

## Instrucciones LOAD/AND/OUT

### LOAD y LOAD NOT:

La primera condición que inicia una línea de instrucción (rung) en un diagrama de escalera corresponde a una instrucción LOAD o LOAD NOT. Cada una de estas instrucciones requiere una línea de código mnemónico.

#### Ejemplo



#### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00000	LD	00000
00001	INSTRUCCIÓN	
00002	LD NOT	00000
00003	INSTRUCCIÓN	

En este ejemplo, para la instrucción LOAD (condición normalmente abierto) la condición de ejecución está en ON cuando IR00000 está en

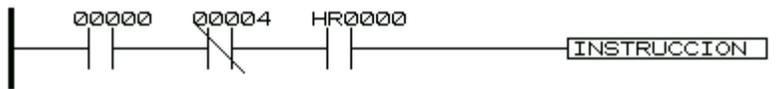
ON, y para la instrucción LOAD NOT (condición normalmente cerrado) la condición está en ON cuando IR00000 está en OFF.

### **AND y AND NOT:**

Cuando dos o más condiciones se encuentran en serie en una misma línea de instrucción, la primera corresponde a una instrucción LOAD o LOAD NOT, y el resto de las condiciones a instrucciones AND y AND NOT. El ejemplo siguiente muestra tres condiciones las cuales corresponden en orden desde la izquierda a una instrucción LOAD, una AND NOT y una AND.

Nuevamente, cada una de las tres instrucciones requiere una línea de código mnemónico.

#### **Ejemplo**



#### **Listado de mnemónicos**

PASO	INSTRUCCION	OPERANDOS
00000	LD	00000
00001	AND NOT	00004
00002	AND	HR0000
00003	INSTRUCCIÓN	

La condición de ejecución de la instrucción estará en ON sólo cuando las tres condiciones sean ON, es decir, cuando IR00000 esté ON, IR00004 esté OFF y HR0000 esté ON.

## **OUTPUT y OUTPUT NOT:**

La forma más sencilla de presentar en una salida el resultado de la combinación de condiciones de ejecución es programarla directamente con las instrucciones OUTPUT y OUTPUT NOT.

Estas instrucciones controlan el estado del bit operando designado de acuerdo al resultado de la condición de ejecución. Con la instrucción OUTPUT (se abrevia OUT), el bit se pondrá ON mientras la condición de ejecución sea ON, y se pondrá OFF mientras la condición de ejecución sea OFF. Con la instrucción OUTPUT NOT (se abrevia OUT NOT), el operando se pondrá ON mientras la condición esté en OFF y se pondrá OFF mientras la condición sea ON.

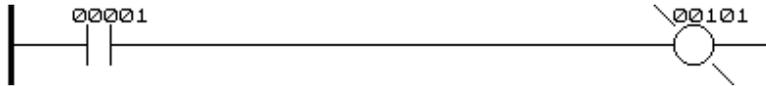
### **Ejemplo. Instrucción OUTPUT**



### **Listado de mnemónicos**

PASO	INSTRUCCION	OPERANDOS
00000	LD	00000
00001	OUT	00100

### Ejemplo. Instrucción OUTPUT NOT



#### Listado de mnemónicos

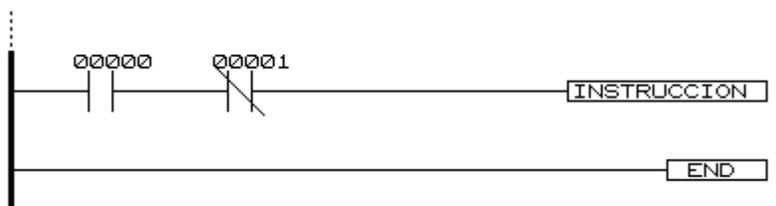
PASO	INSTRUCCION	OPERANDOS
00000	LD	00001
00001	OUT NOT	00101

### Instrucción END (01)

La última instrucción necesaria para completar un programa es la instrucción END (01). La CPU ejecuta las líneas de programa en forma cíclica, comienza por la primera instrucción y ejecuta secuencialmente las siguientes hasta encontrar la instrucción END (01), momento en que vuelve a ejecutar la primera instrucción y repite el ciclo. Las instrucciones de un programa situadas a continuación de la instrucción END (01) no se ejecutan.

Esta instrucción no necesita operando, y no debe haber ninguna condición en su línea de instrucción.

#### Ejemplo



### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00500	LD	00000
00501	AND NOT	00001
00502	INSTRUCCION	
00503	END (01)	

**Nota: El número entre paréntesis después de una instrucción (01 para la instrucción END) representa su código de función y será utilizado para insertar el programa en el PLC vía Cónsola de Programación**

### Ejercicio

Realizar el diagrama de escaleras y el listado de mnemónicos del esquema eléctrico de la figura 2.3, donde el solenoide número 1 estará ON sólo cuando el pulsador 1 esté ON y el pulsador 2 esté OFF.

Nota: Asuma las direcciones 00000 para PB1, 00001 para PB2 y 00100 para SOL1

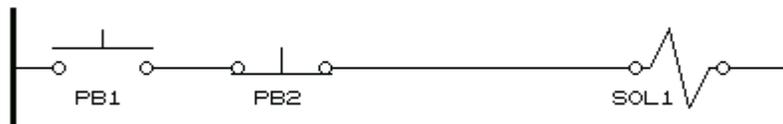


Figura 2.3

### Diagrama de escalera



### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS

### Instrucción OR

#### OR y OR NOT:

Cuando dos o más condiciones en líneas paralelas se juntan, la primera condición corresponde a una instrucción LOAD o LOAD NOT; el resto de las condiciones a instrucciones OR y OR NOT. El siguiente ejemplo, muestra tres condiciones correspondientes a las siguientes instrucciones: LOAD NOT, OR NOT, y OR. Cada una de estas instrucciones necesita una línea de código mnemónico.

### Ejemplo



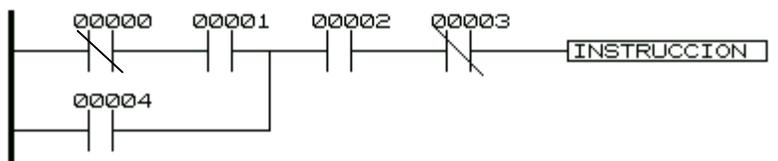
### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00000	LD NOT	00000
00001	OR NOT	00001
00002	OR	HR0000
00003	INSTRUCCION	

La condición de ejecución de la instrucción será ON cuando al menos una de las condiciones anteriores sea ON, es decir, cuando el bit IR00000 esté OFF, • el bit IR00001 esté en OFF, • el bit HR0000 esté en ON.

### Combinación de instrucciones AND y OR:

#### Ejemplo

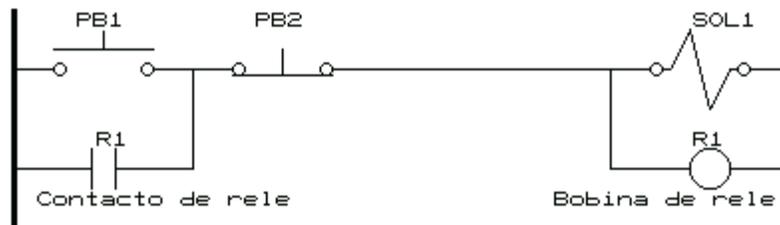


PASO	INSTRUCCIÓN	OPERANDOS
00000	LD NOT	00000
00001	AND	00001
00002	OR	00004
00003	AND	00002
00004	AND NOT	00003
00005	INSTRUCCIÓN	

En este ejemplo, se ejecuta una AND entre el estado del bit 00000 con el del bit 00001, y el resultado se hace OR con el estado del bit 00004. El resultado de esta operación es la condición de ejecución para la AND con el estado del bit 00002, que a su vez será la condición de ejecución para la AND con el estado negado del bit 00003.

### Ejercicio

Suponga ahora que en el ejercicio de la figura 2.3, se desea mantener el solenoide activado aun después que PB1 sea liberado. Electricamente, esto requeriría la introducción de un relé extra para permitir al circuito recordar que PB1 fue cerrado (ver figura 2.4).



**Figura 2.4**

Realizar el diagrama de escaleras y el listado de mnemónicos del esquema eléctrico de la figura, donde el solenoide 1 se pondrá ON cuando PB1 sea pulsado y permanecerá ON aún después que PB1 sea liberado. El solenoide 1 se pondrá OFF cuando PB2 sea pulsado.

Nota: El relé 1 puede ser sustituido por un bit de trabajo interno. Asuma las direcciones 00000 para PB1, 00001 para PB2, 00100 para SOL1 y 23300 para R1.

### **Diagrama de escalera**



### **Listado de mnemónicos**

<b>PASO</b>	<b>INSTRUCCIÓN</b>	<b>OPERANDOS</b>
00000		
00001		
00002		
00003		
00004		
00005		

## Instrucciones de bloque lógico AND LOAD/OR LOAD

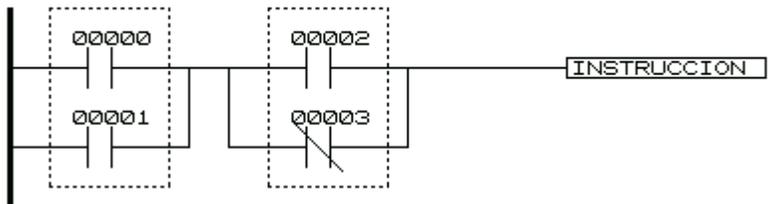
Todo grupo de condiciones unidas para crear un resultado lógico, se denomina bloque lógico. La comprensión de éstos es necesaria para programar eficazmente y es esencial cuando los programas se escriben en código mnemónico.

Las instrucciones de bloque describen la interrelación de bloques lógicos. La instrucción AND LOAD realiza la operación lógica AND de los resultados de dos bloques lógicos. La OR LOAD ejecuta la operación lógica OR de dos bloques.

### AND LOAD:

El siguiente diagrama de relés, requiere una instrucción AND LOAD.

#### Ejemplo



#### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	OR	00001
00002	LD	00002
00003	OR NOT	00003
00004	AND LOAD	
00005	INSTRUCCION	

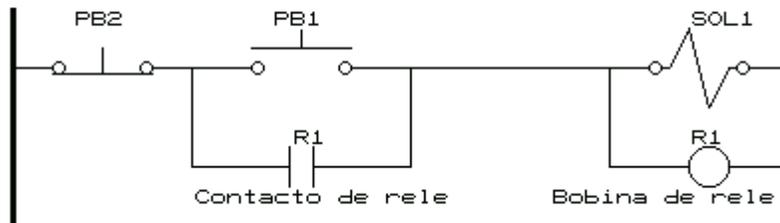
Los dos bloques lógicos son los marcados por líneas discontinuas.

Ya que los bloques están en serie se debe realizar una operación lógica AND entre ellos. La forma de hacerla es realizar las OR (NOT)s implícitas en cada uno de ellos independientemente, y luego combinar los resultados con la instrucción AND LOAD.

En este ejemplo, la condición de ejecución de la instrucción será ON cuando al menos una de las condiciones del bloque de la izquierda sea ON **y** al menos una de las condiciones del bloque de la derecha es ON.

### Ejercicio

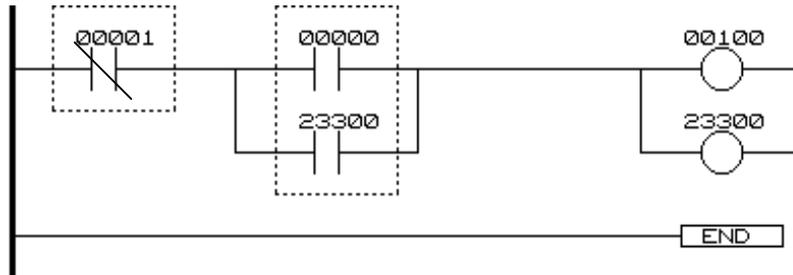
El esquema eléctrico en la siguiente figura, representa el mismo circuito de la figura 2.4, sólo que el pulsador de restauración PB2 se colocó al principio.



**Figura 2.5**

### Diagrama de escalera

(Diagrama de escalera manteniendo la estructura del circuito)



Realizar el listado de mnemónicos.

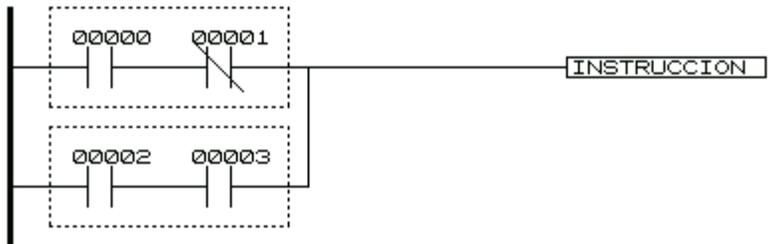
#### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000		
00001		
00002		
00003		
00004		
00005		
00006		

#### OR LOAD:

Cuando en una línea de instrucción existen bloques paralelos, la instrucción que combina los resultados de cada uno de ellos corresponde a una instrucción OR LOAD.

### Ejemplo



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	AND NOT	00001
00002	LD	00002
00003	AND	00003
00004	OR LOAD	
00005	INSTRUCCION	

La condición de ejecución de la instrucción será ON cuando el bit 00000 sea ON y el bit 00001 sea OFF • cuando el bit 00002 y el bit 00003 sean ON.

### Ejercicio

Realizar el listado de mnemónicos del siguiente diagrama de escalera.

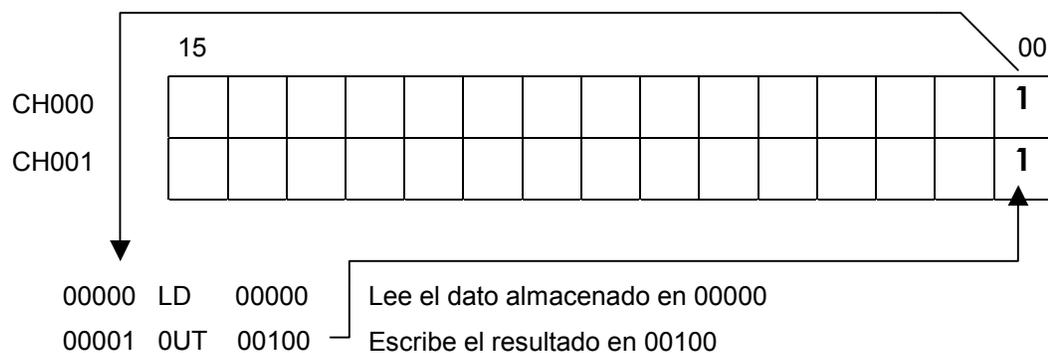


## ACTUALIZACION DE E/S

Ahora que se ha visto algunos ejemplos de programa, se tomará en consideración cómo el PLC ejecuta el programa y maneja la información de entrada y salida.

El PLC realiza tres pasos básicos:

1. Lee todas las condiciones de entrada desde los dispositivos del mundo real y las almacena en una tabla que es usada durante la ejecución del programa. Por ejemplo, LD 00000 está preguntando a la CPU acerca de la condición de la entrada 00000 almacenada en la tabla (un "1" o un "0").
2. Ejecuta el programa usando los datos almacenados en la tabla. LD, AND, OR son instrucciones de lectura de la tabla, mientras que OUT es una instrucción de escritura.



3. Después que la instrucción END es encontrada, las salidas del mundo real son actualizadas.

## CICLO DEL PLC

Cuando el PLC opera, es decir, cuando ejecuta su programa para controlar un sistema externo, una serie de operaciones son ejecutadas dentro de él. Estas operaciones internas pueden ser clasificadas en las siguientes cuatro categorías:

1. Procesamientos comunes, tales como reseteo del temporizador perro-guardián (en inglés watchdog) y otras operaciones de diagnósticos.
2. Ejecución del programa
3. Actualización de entradas y salidas
4. Servicio a dispositivos periféricos

El tiempo total requerido por el PLC para ejecutar las operaciones anteriores es llamado **Tiempo de Ciclo**.

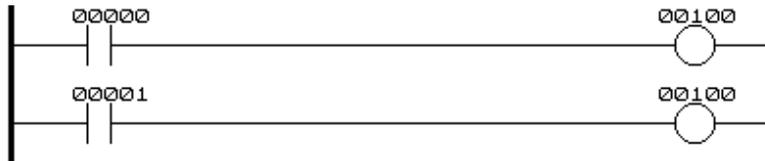
## DUPLICACION DE SALIDAS

Debido a la forma en que la CPU ejecuta el programa y actualiza las salidas, **no se recomienda duplicar salidas**.

El PLC almacena la información de entrada en una **tabla de bits de entrada** y trabaja desde la tabla durante la ejecución del programa.

Cada vez que una línea de instrucción del diagrama escalera es efectuada, el resultado se almacena en la **tabla de bits de salida**. Por esto, si una salida (bobina) es duplicada, su última ocurrencia en el programa es la única válida con respecto al mundo real. Las salidas son actualizadas después que todo el programa ha sido ejecutado.

### Ejemplo



### Tabla de Bits de E/S

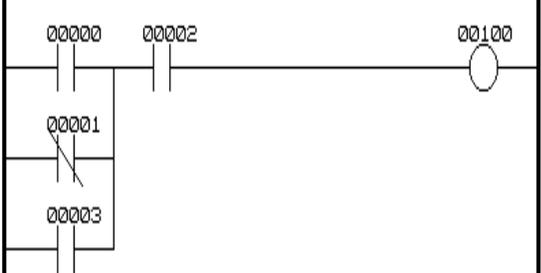
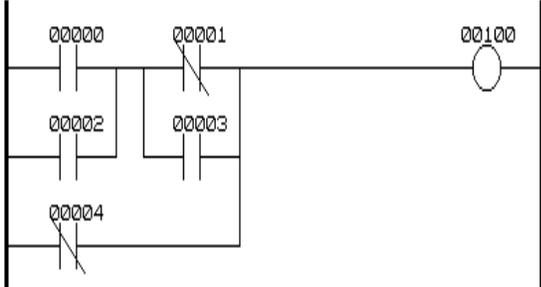
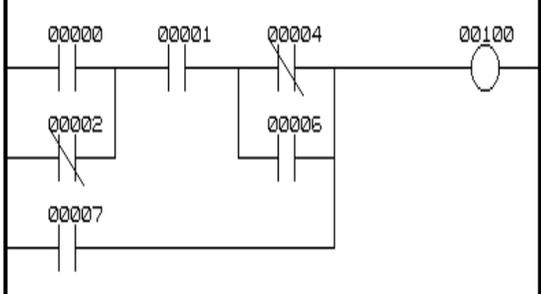
BIT CA	15	.....	00
000			
001			
.....			

Figura 2.6

En el ejemplo anterior, la salida 00100 es controlada por la entrada 00000 y luego en el programa por 00001. La tabla de bits de E/S es escrita durante la ejecución del programa, tal que, las localidades de salida que fueron escritas por última vez son las condiciones que serán enviadas a las salidas reales.

## Ejercicios de programación

Crear el listado de mnemónicos de los diagramas de escalera mostrados.

Diagrama de escalera	Listado de mnemónicos		
	Paso	Instrucc.	Operand.
1. 			
2. 			
3. 			

Crear el diagrama de escalera usando el listado de mnemónicos mostrado.

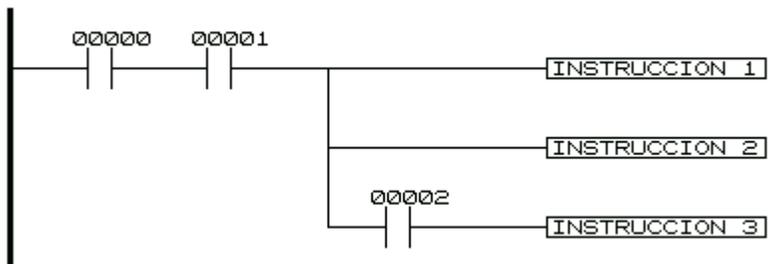
Diagrama de escalera	Listado de mnemónicos		
	Paso	Instrucc.	Operand.
	00000 00001 00002 00003 00004 00005 00006 00007	LD OR LD OR OR AND LD OUT END	00000 00001 00002 00003 00004  00100
	00000 00001 00002 00003 00004	LD NOT AND NOT OR OUT END	00000 00001 00002 00100
	00000 00001 00002 00003 00004 00005 00006 00007	LD LD AND OR AND LD OUT OUT NOT END	00000 00001 00002 00003  00100 00101
	00000 00001 00002 00003 00004 00005 00006 00007 00008 00009	LD OR LD AND LD AND OR LD AND LD OUT END	00000 00001 00002 00003 00004 00005   00100

## Relés temporales (TRs)

Cuando una línea de instrucción se divide en más de una línea, en ciertas ocasiones será necesario utilizar bits TRs para almacenar la condición de ejecución existente en el punto de ramificación.

Como en el diagrama A, si después del punto en que se divide la línea, no existe(n) más condición(es) de ejecución para las instrucciones, o existe(n) únicamente en la última instrucción; entonces la programación en mnemónicos se desarrolla tal como se ha visto hasta ahora.

### Diagrama A



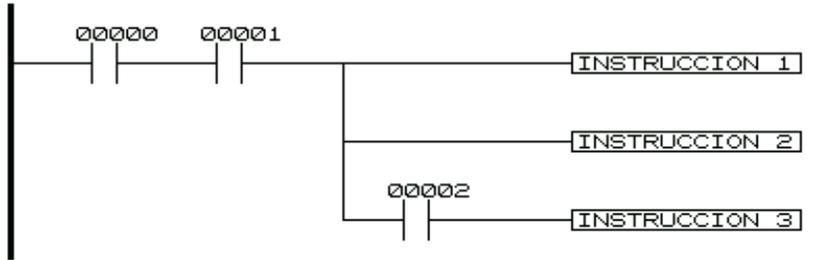
### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00000	LD	00000
00001	AND	00001
00002	INSTRUCCION 1	
00003	INSTRUCCION 2	
00004	AND	00002
00005	INSTRUCCION 3	

Como en el diagrama B, si después del punto en que se divide la línea existe(n) otra(s) condición(es) de ejecución para las instrucciones, es

necesario identificar el punto de ramificación usando un bit TR. Hay 8 relés temporales (TR0 a TR7) en el PLC C200HG. La programación en mnemónicos es como se muestra.

### Diagrama B



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	AND	00001
<b>00002</b>	<b>OUT</b>	<b>TR0</b>
00003	AND	00002
00004	INSTRUCCION 1	
<b>00005</b>	<b>LD</b>	<b>TR0</b>
00006	AND	00003
00007	INSTRUCCION 2	
<b>00008</b>	<b>LD</b>	<b>TR0</b>
00009	AND	00004
00010	INSTRUCCION 3	

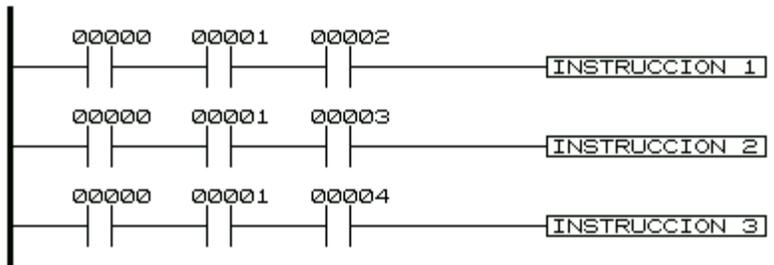
#### Instrucciones TR

**OUT TR--** Identifica el punto de ramificación

**LD TR--** Hace un llamado al punto de ramificación

El diagrama B es una versión simplificada del diagrama de escalera de la figura 2.6, cuya programación en mnemónicos no requiere el uso de bits TRs.

Obsérvese que prescindir de los bits TRs implica el uso de mayor número de contactos, dado que, es necesario repetir la condición de ejecución en cada línea de instrucción.



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	AND	00001
00002	AND	00002
00003	INSTRUCCION 1	
00004	LD	00000
00005	AND	00001
00006	AND	00003
00007	INSTRUCCION 2	
00008	LD	00000
00009	AND	00001
00010	AND	00004
00011	INSTRUCCION 3	

## Temporizadores/Contadores

Se requiere el uso de temporizadores, cuando se necesita activar una salida que dependa de un tiempo programado, por ejemplo:

- 1) Activar una sirena durante 5 segundos
- 2) Retardar 10 minutos la apertura de una válvula
- 3) Activar el mezclador de un tanque durante 1/2 hora

Se requiere el uso de contadores, cuando se necesita indicar que ha sucedido un evento un número determinado de veces, por ejemplo:

- 1) Producir una señal sonora cada 20 productos terminados
- 2) Contar el número de piezas que ha producido una línea
- 3) Contar el número de personas que han atravesado una puerta

El PLC C200HG tiene un total de 512 temporizadores/contadores.

Nota: Recuerde que, el número que identifica a un temporizador no puede ser usado por un contador, y viceversa. Esto es, TIM 010 y CNT 010 no pueden ser ambos usados

Hay tres valores asociados a un temporizador/contador:

**N:** Es el número que identifica al temporizador/contador

**SV** (Valor Fijado): Es el valor para el cual se programa el temporizador/ contador

**PV** (Valor Presente): Es el valor actual de operación del temporizador/contador

### Temporizador (TIM)

Los temporizadores pueden ser usados como **instrucciones** y como **operandos** de otras instrucciones.

### Diagrama de escalera de un temporizador como instrucción



### Listado de mnemónicos

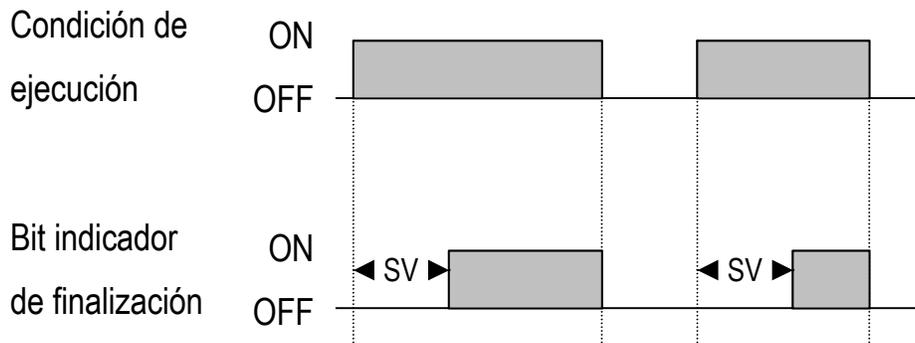
PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	TIM	N
		#SV

- El rango de N va de 000 a 511.
- El rango de SV va de 0000 a 9999 décimas de segundo.

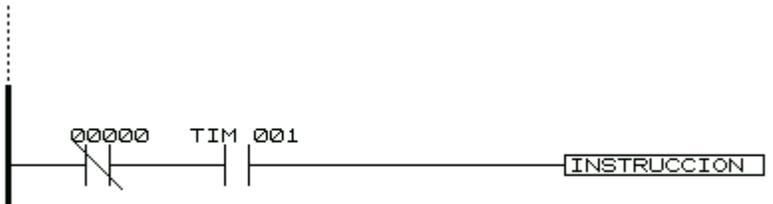
Nota: La instrucción TIM ocupa un paso de programa. El operando SV no toma un paso.

En este ejemplo, la instrucción **TIM** se activa cuando el bit 00000 se pone ON. La condición de ejecución (bit 00000) deberá permanecer ON por lo menos un tiempo igual al SV para que el temporizador pueda restablecerse.

El temporizador de la instrucción TIM es del tipo ON-delay, es decir, se pondrá a ON transcurrido un tiempo SV después que su condición de ejecución es ON. El temporizador vuelve a OFF cuando su condición de ejecución vuelve a OFF. Ver ilustración:



Cuando un temporizador es usado como operando de instrucciones LD, AND, OR, AND LD, OR LD; se referirá a su bit indicador de finalización. Ver figura:



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00015	LD NOT	00000
00016	AND	TIM 001
00017	INSTRUCCION	

### Contador (CNT)

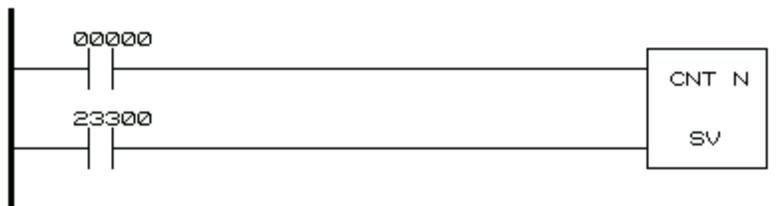
Los contadores pueden ser usados como **instrucciones** y como **operandos** de otras instrucciones.

Como instrucción, CNT tiene dos entradas de control:

**Entrada de conteo:** La condición de esta entrada produce un impulso cada vez que sucede el evento que se desea contar.

**Entrada de Reset:** La condición de esta entrada produce un impulso para restablecer el contador.

### Diagrama de escalera de un contador como instrucción



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	LD	23300
00002	CNT	N
		#SV

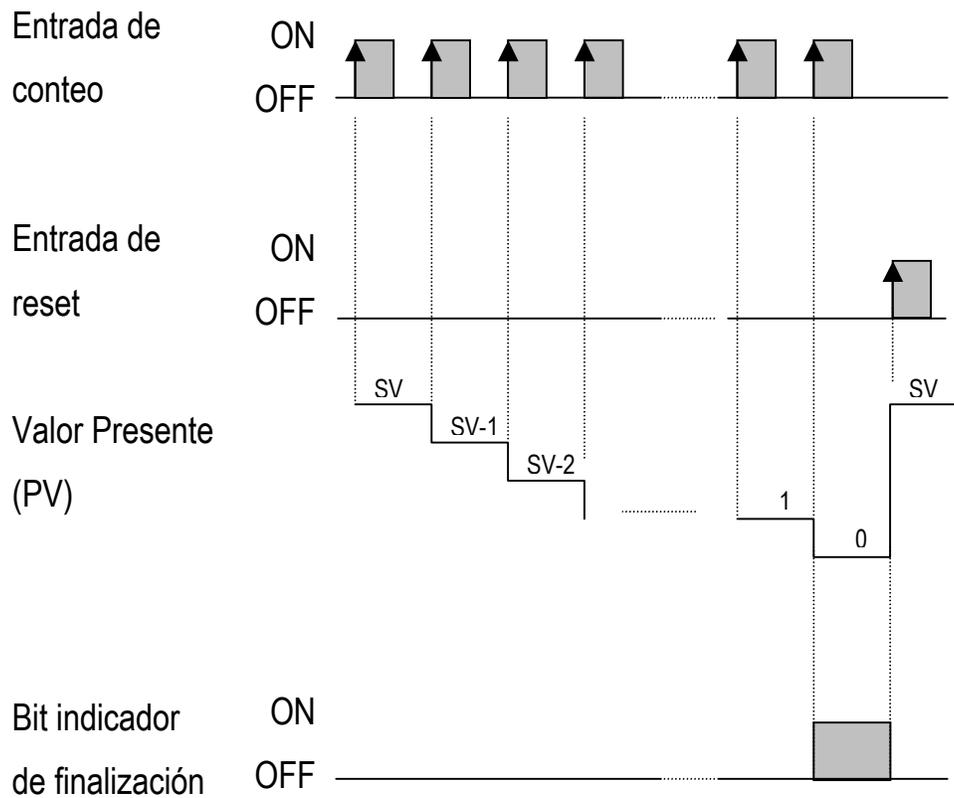
El rango de N va de 000 a 511.

El rango de SV va de 0000 a 9999.

Nota: La instrucción CNT ocupa un paso de programa. El operando SV no toma un paso.

La instrucción CNT se utiliza para descontar en uno (contar hacia atrás) a partir del SV, cada vez que se produce un impulso en la entrada de conteo. Un impulso significa que la condición debe pasar de OFF a ON y luego a OFF nuevamente. El contador se decrementa con el flanco

de subida del impulso de conteo. Cuando el valor presente (PV) del contador llega a cero el **bit indicador de finalización** se pone a ON y permanece en ON hasta tanto no se produzca un impulso en la entrada de reset. Un impulso en la entrada de reset restablece el valor presente (PV) del contador al valor fijado (SV) y lo prepara nuevamente para el conteo. Ver ilustración:

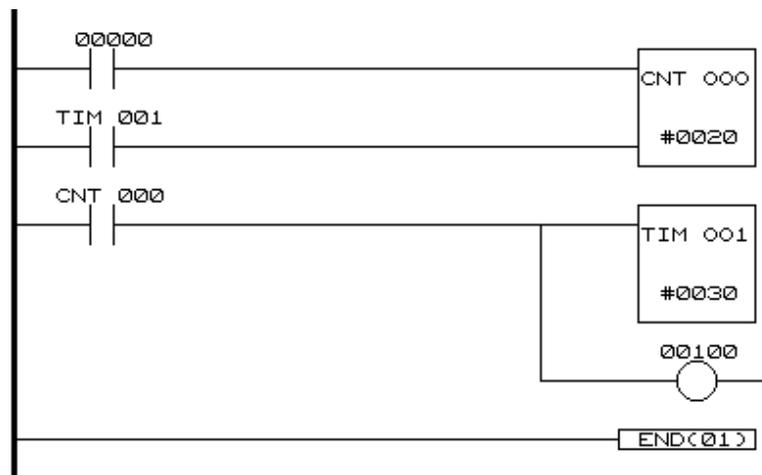


Cuando un contador se usa como operando de instrucciones LD, AND, OR, AND LD, OR LD; se estará refiriendo a su bit indicador de finalización. Ver figura:



### Ejemplo

El siguiente programa, activa la salida 00100 durante 3 segundos, cada vez que se cuentan 20 eventos en la entrada 00000.



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	LD	TIM 001
00002	CNT	000
		#0020

00003	LD	CNT 000
00004	TIM	001
		#0030
00005	OUT	00100
00006	END	

### Ejercicio

Diseñe un programa con banda transportadora para llenar cajas con una cantidad conocida de partes.

Requerimientos:

1. Número de partes/caja = 10
2. Inicio de funcionamiento MANUAL
3. Indicar potencia ON e indicar banda transportadora en funcionamiento
4. Parar la banda transportadora cuando se complete una caja e indicar caja completa
5. El operador ubica una caja vacía en posición y reinicia el funcionamiento de la banda.

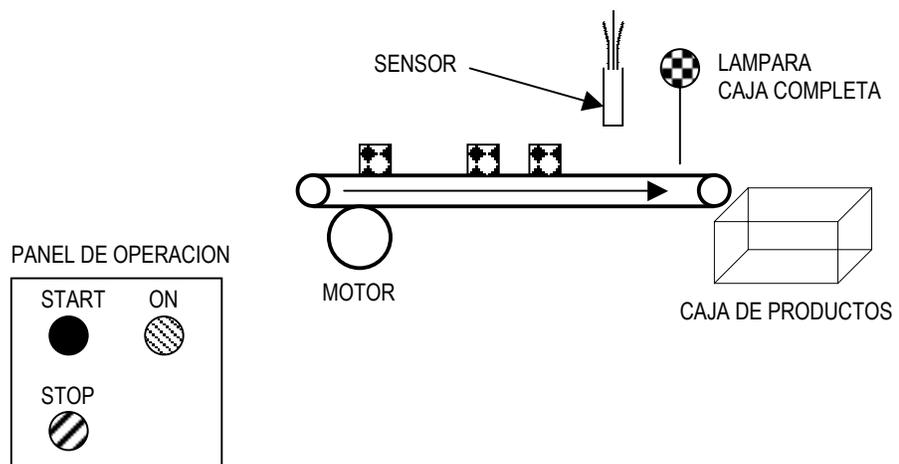
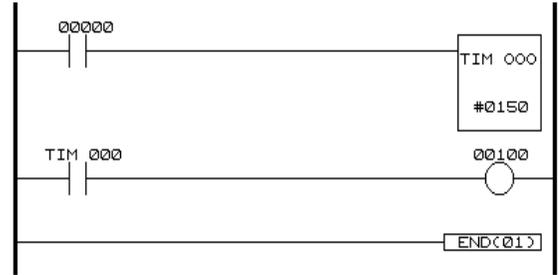
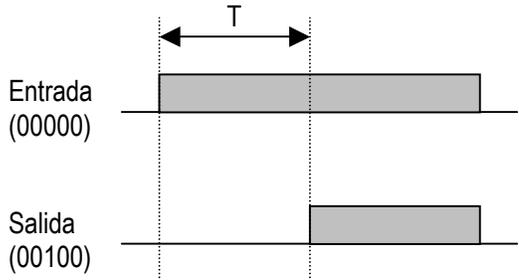


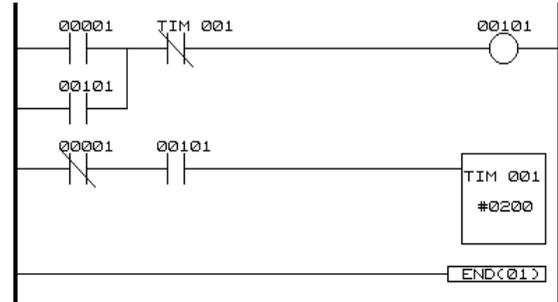
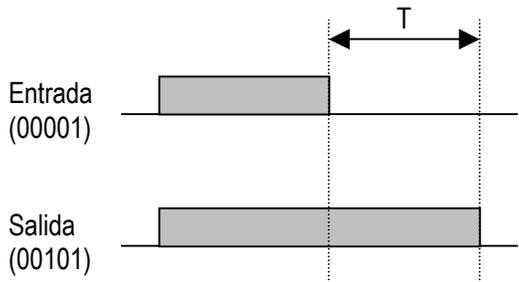
Figura 2.7

### Circuitos comunes de temporizadores

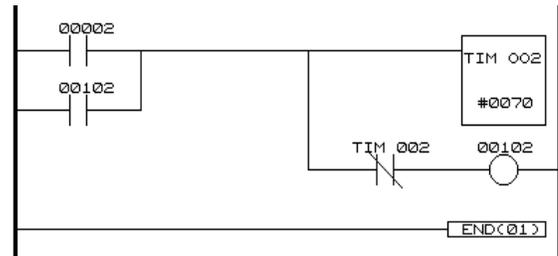
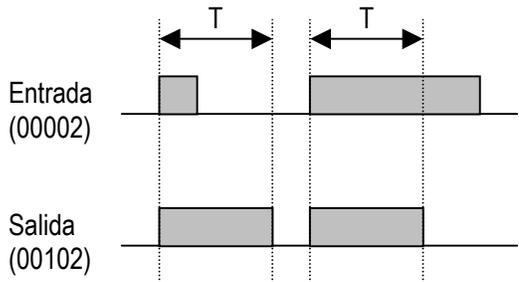
#### ON - DELAY



#### OFF- DELAY



#### ONE - SHOT



## Contador Reversible (CNTR (12))

El contador estándar es un contador descendente, se inicia con el valor fijado (SV) y cuenta hacia abajo hasta #0000.

Algunas aplicaciones necesitan un conteo hacia arriba, tal como mostrar el número de partes producidas. El valor presente (PV) del contador es el número actual de partes producidas. En estos casos puede emplearse un contador reversible.

Los contadores reversibles pueden ser usados como instrucciones y como operandos de otras instrucciones.

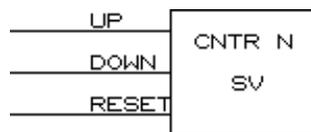
Como instrucción, CNTR tiene tres entradas de control:

**Entrada de cuenta hacia arriba (UP):** Incrementa el conteo

**Entrada de cuenta hacia abajo (DOWN):** Decrementa el conteo

**Entrada de reset (RESET):** Resetea la cuenta a #0000

### Símbolo en diagrama de escalera

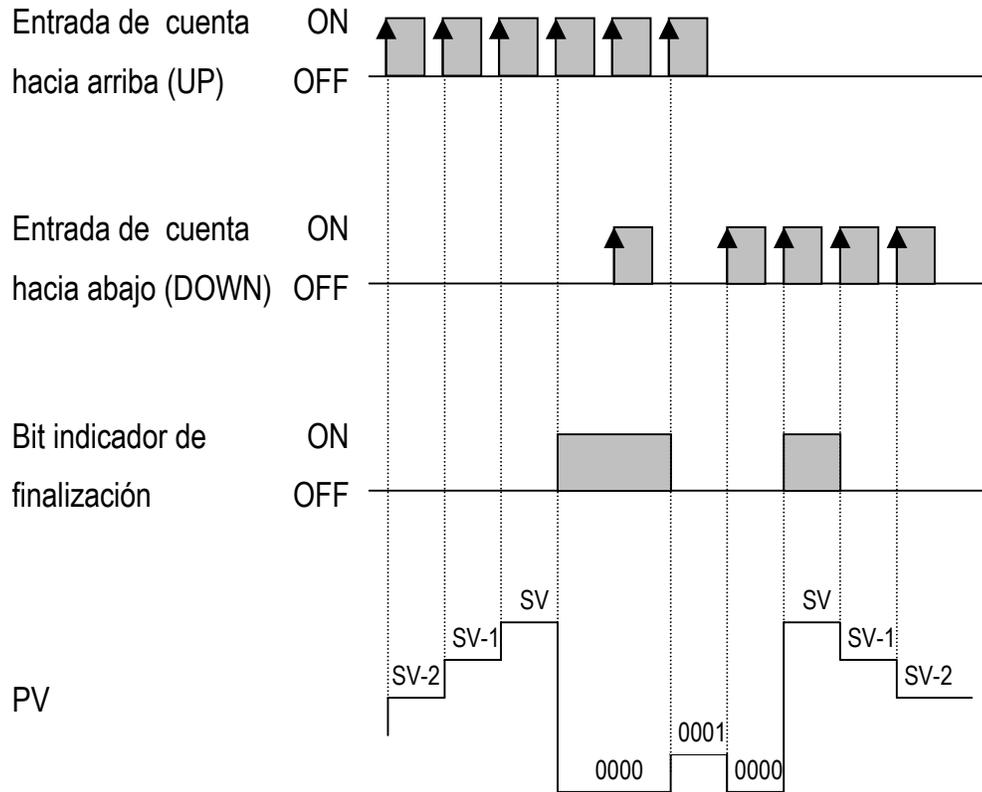


El rango de N va de 000 a 511.

El rango de SV va de 0000 a 9999.

El contador reversible opera como un contador cíclico. Por ejemplo, si se introduce un SV de #0009, contando hacia abajo se producirá una salida cuando el contador se devuelva de #0000 a #0009 (10 cuentas) y la salida permanecerá activa hasta que se produzca una nueva entrada de cuenta. Contando hacia arriba, cuando el contador

transite de #0009 a #0000 (10 cuentas) se producirá una salida. Ver ilustración:



Cuando un contador reversible es usado como operando de instrucciones LD, AND, OR, AND LD, OR LD; se referirá a su bit indicador de finalización y se identificará por su número de contador (N) precedido del prefijo CNT. El prefijo CNT se utilizará indistintamente de la instrucción contador empleada para definir el contador.

### Ejemplo

En esta aplicación, el sensor A cuenta las partes que entran a una estación particular de un proceso, y el sensor B cuenta las partes que se separan del proceso. Imagine que se trata de un proceso donde el máximo número de partes admitidas en un mismo tiempo es 25.

El contador reversible puede ser usado para registrar el número de partes sobre la banda transportadora en un mismo tiempo.

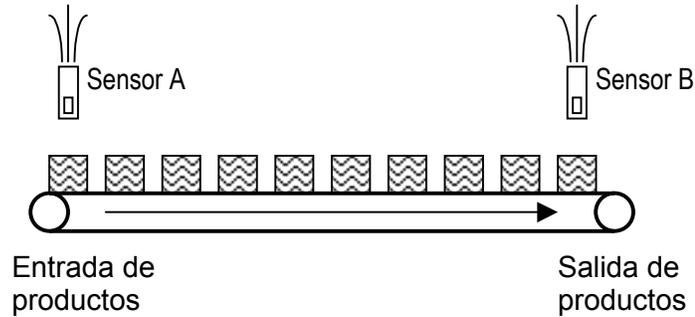


Figura 2.8

**Nº de partes sobre la banda = Valor del conteo hacia arriba menos**

**Valor del conteo hacia abajo**

Cada vez que un producto entra a la banda el contador se incrementa en 1.

Cada vez que un producto sale de la banda el contador se decrementa en 1.

Introduzca el programa siguiente y pruebe su operación:

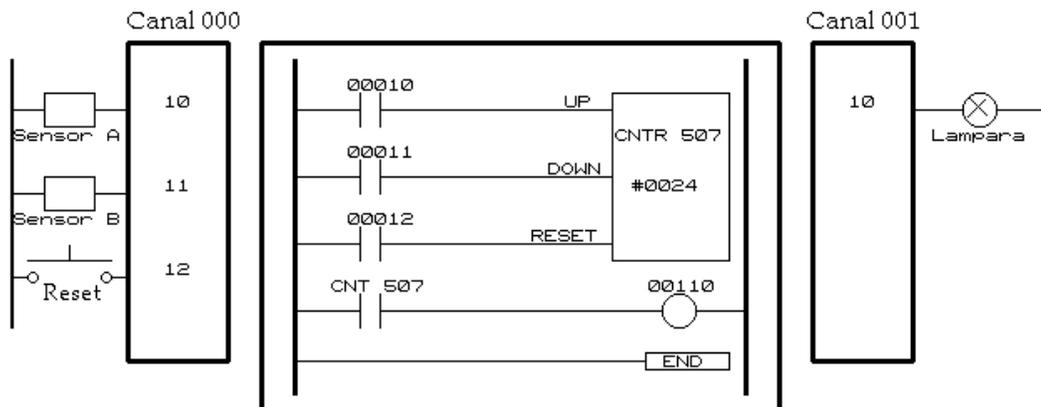


Figura 2.9

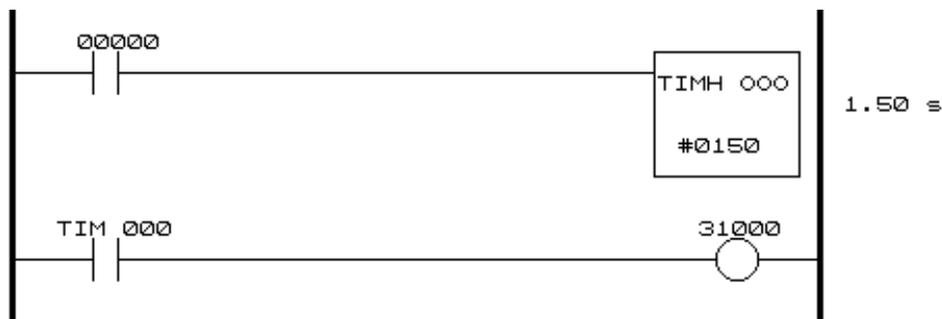
### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00000	LD	00010
00001	LD	00011
00002	LD	00012
00003	CNTR (12)	507
		#0024
00004	LD	CNT 507
00005	OUT	00110
00006	END (01)	

### Temporizador de Alta Velocidad (TIMH (15))

TIMH opera de la misma forma que TIM excepto que la unidad de medida es 0.01 segundo.

#### Ejemplo



### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDOS
00000	LD	00000

00001	TIMH (15)	000
		#0150
00002	LD	TIM 000
00003	OUT	31000

### Precauciones:

- Aunque se puede seleccionar un SV entre #0000 y #9999, con #0000 se inhibirá el temporizador, es decir, el indicador de finalización se pondrá inmediatamente a ON, y con #0001 no tendrá fiabilidad.
- No se deben utilizar los números de temporizador (N) de 016 a 511 si el tiempo de ciclo es superior a 10 ms.

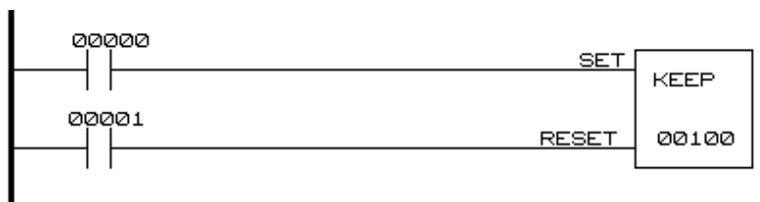
## Instrucción KEEP (11)

Keep se utiliza para simular un relé de enclavamiento. Esta instrucción tiene dos entradas de control: SET y RESET.

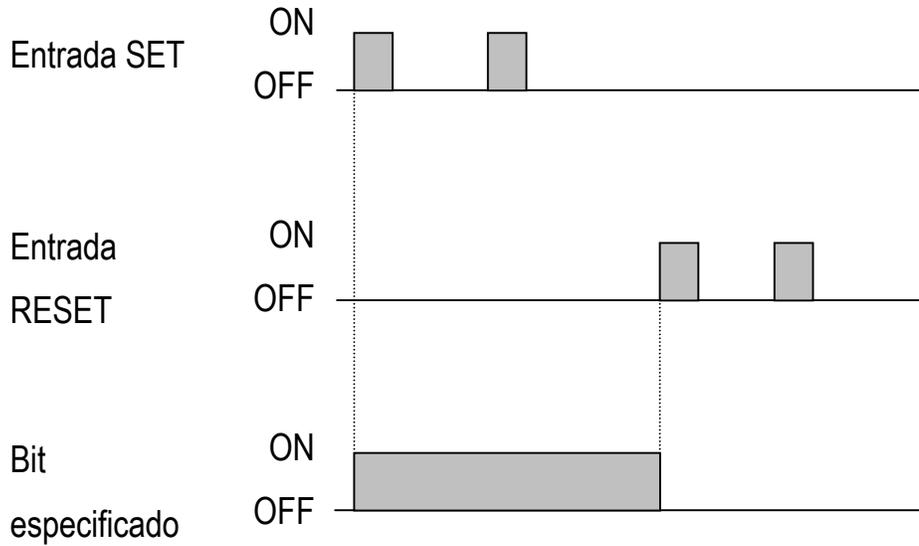
**SET:** Cuando la condición de esta entrada se pone a ON, el bit especificado en el KEEP se pone a ON.

**RESET:** Cuando la condición de esta entrada se pone a ON, el bit especificado en el KEEP se pone a OFF, independientemente del estado de la entrada SET

### Ejemplo



La figura siguiente muestra el funcionamiento de la instrucción KEEP:



### Ejemplo

En el ejemplo de la figura 2.10, la salida L1 es controlada directamente por PB1. Tanto tiempo como PB1 es cerrado, L1 ilumina. La salida L2 es controlada por PB2 y PB3. Cuando PB2 se cierra, L2 ilumina. Esta vez, cuando PB2 se abre, L2 permanece iluminada hasta que PB3 sea cerrado causando que L2 se apague.

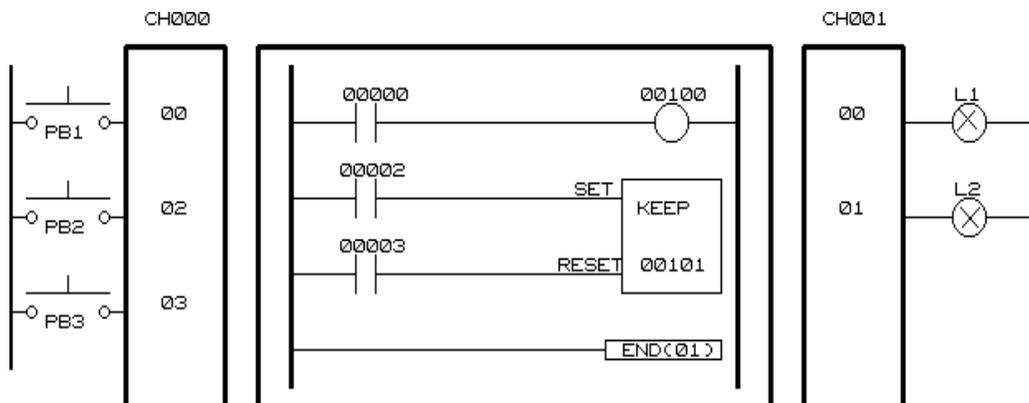


Figura 2.10

### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00000	LD	00000
00001	OUT	00100
00002	LD	00002
00003	LD	00003
00004	KEEP (11)	00101
00005	END (01)	

### Instrucción DIFFERENTIATE UP - DIFU (13)

Cuando la condición de ejecución de la instrucción DIFU presenta un flanco de subida, el bit especificado por ésta se coloca en ON durante un ciclo de programa (llamado también un Scan).

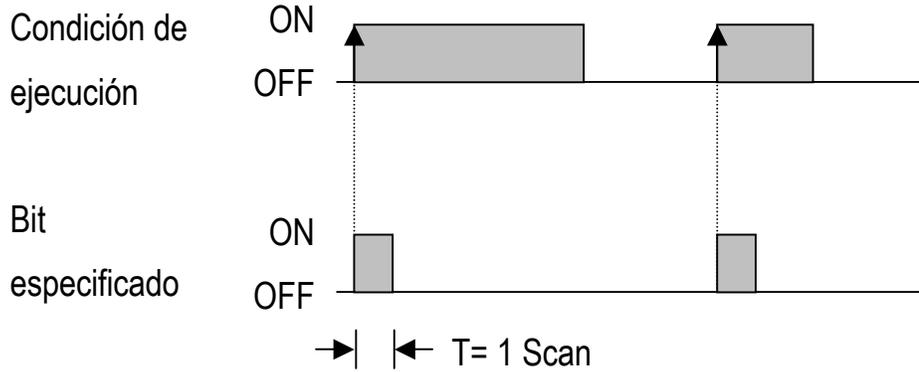
#### Ejemplo



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00020	LD	00001
00021	DIFU	23300

El siguiente diagrama muestra el funcionamiento de la instrucción DIFU:



Cuando la entrada 00001 es cerrada (pasa de OFF a ON), el bit de trabajo 23300 irá a ON durante un ciclo de programa solamente. Esta instrucción no es usada para turnar ON dispositivos de salida, ya que un ciclo de programa tiene una duración típica de 20 a 50 milisegundos. Esto no sería suficiente para activar ninguna salida real.

### Ejercicio

Un sensor detecta productos sobre una banda transportadora principal, la cual envía los productos alternadamente hacia otras dos bandas transportadoras (banda A y banda B). El primer producto detectado por el sensor va hacia la banda A; el próximo hacia la banda B; el tercero hacia la banda A, etc.

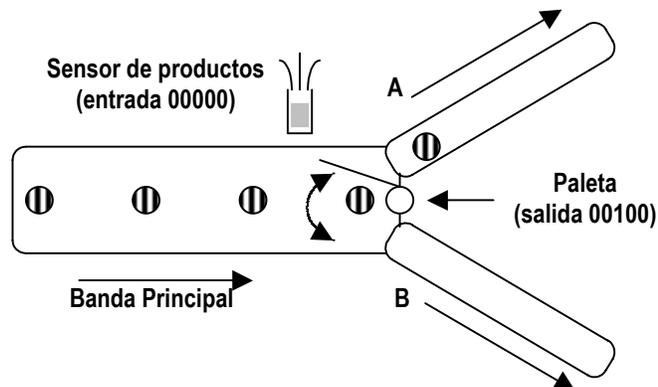


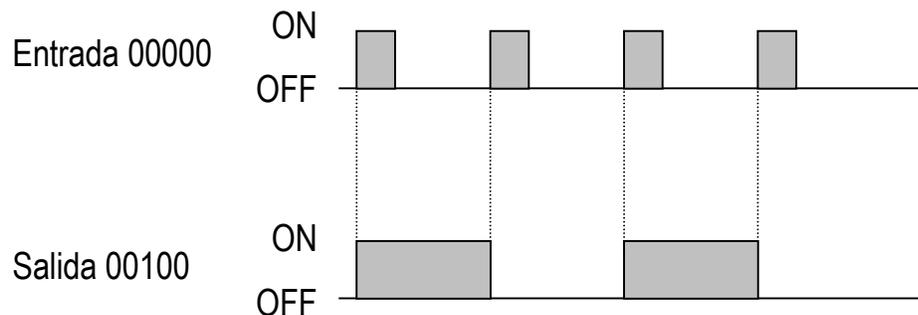
Figura 2.11

### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS

Diseñe el programa para este sistema (asistido por el instructor).

**Sugerencias:** Un pulso en la entrada 00000 causará el enclavamiento a ON de la salida 00100. Utilice la instrucción KEEP para el enclavamiento de la salida 00100. Si ocurre un nuevo pulso en la entrada 00000 **y** la salida 00100 está en ON, ésta debe volver a OFF. Ver diagrama:



## Instrucción DIFFERENTIATE DOWN - DIFD (14)

Funcionalmente, la instrucción DIFD es la misma que DIFU excepto que es activada con el flanco de bajada de su condición de ejecución, colocando en ON el bit especificado durante un ciclo de programa.

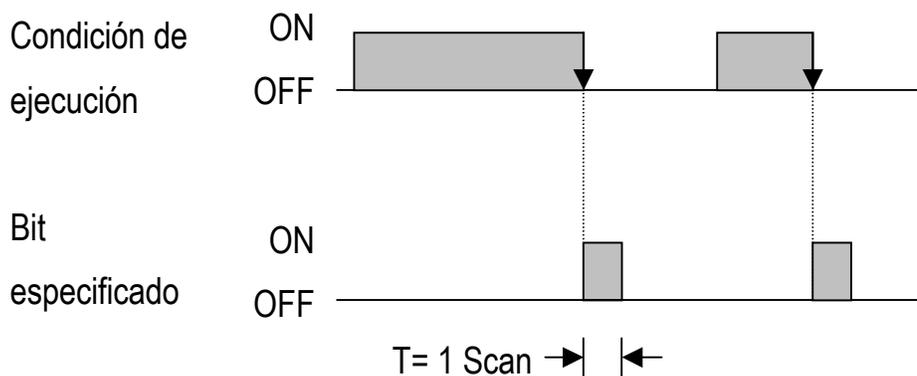
### Ejemplo



### Listado de mnemónicos

PASO	INSTRUCCIÓN	OPERANDOS
00010	LD	00001
00011	DIFD	23301

El siguiente diagrama muestra el funcionamiento de la instrucción DIFD:



Cuando la entrada 00001 pase de ON a OFF, el bit especificado 23301 irá a ON durante un ciclo de programa.

## Instrucción INTERLOCK e INTERLOCK CLEAR- IL (02) e ILC (03)

IL e ILC se usan en pareja. Cuando la condición de ejecución para IL es ON, el programa se ejecuta como está escrito, con una condición de ejecución ON utilizada para iniciar cada línea de instrucción desde el punto en que se encuentra IL hasta el siguiente ILC (ver figura 2.12). Si la condición de ejecución para IL es OFF, la sección de programa enclavada entre IL e ILC se tratará como se indica en la tabla siguiente:

Instrucción	Tratamiento
<b>OUT y OUT NOT</b>	Bit designado puesto a OFF
<b>TIM y TIMH(15)</b>	Reset
<b>CNT Y CNTR(12)</b>	Se mantiene el PV
<b>KEEP(11)</b>	Se mantiene el estado del bit
<b>DIFU(13) y DIFD(14)</b>	No ejecutada
<b>Resto de instrucciones</b>	No se ejecutan las instrucciones y todos los bits y canales de IR, AR, LR, HR y SR escritos como operandos en las instrucciones se ponen a OFF

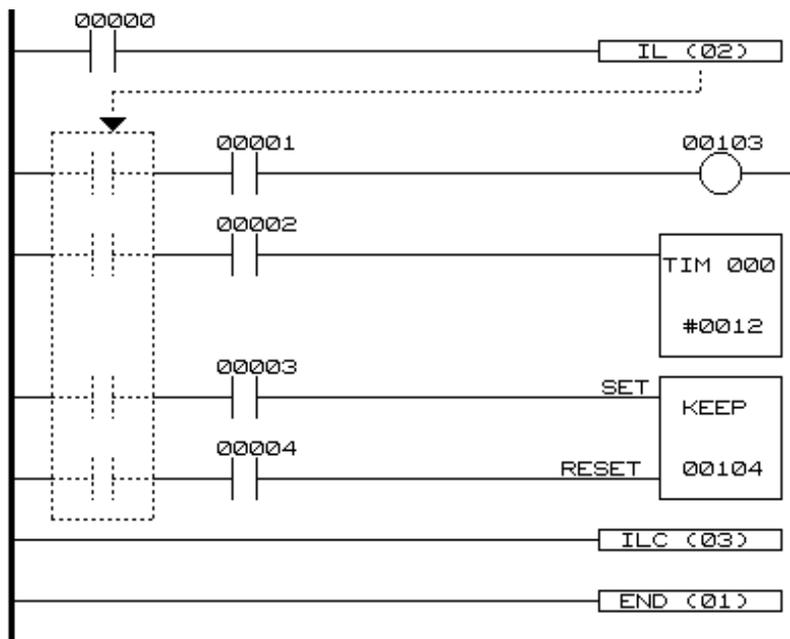


Figura 2.12

Nota: En el ejemplo de la figura 2.12, las líneas punteadas muestran el efecto de usar la función INTERLOCK. Todas las líneas de instrucción entre IL e ILC pueden considerarse que tienen un contacto imaginario controlado por la entrada 00000.

### **Anidamiento de INTERLOCKS**

IL e ILC no han de usarse exactamente por parejas. IL se puede utilizar varias veces en una columna, con cada IL creando una sección de programa enclavada hasta la siguiente ILC.

**Precauciones:** - Siguiendo a una o más IL debe seguir una ILC.  
- Aunque con una ILC se pueden utilizar tantas IL como sean necesarias, la instrucción ILC no se puede utilizar consecutivamente sin al menos una IL entre ella y cualquier ILC anterior.

### **Instrucción JUMP y JUMP END- JMP (04) y JME (05)**

JUMP y JUMP END son instrucciones que operan en forma similar a INTERLOCK e INTERLOCK CLEAR.

JUMP se utiliza siempre junto con JUMP END para crear saltos, es decir para saltar de un punto a otro en el diagrama de escalera. JMP define el punto desde el que se salta; JME define el punto destino del salto.

Los saltos están numerados desde 00 a 99.

### Símbolo en diagrama de escalera

————— JMP(04) N

**N:** Número del salto

————— JME(05) N

Cuando la condición de ejecución para JMP es ON, no se ejecuta ningún salto. Cuando la condición de ejecución para JMP es OFF, se realiza un salto a la JME con el mismo número de salto, no ejecutándose las instrucciones entre ambas JMP y JME, y se ejecuta la instrucción que sigue a JME.

#### NOTAS:

1. El número de salto 00 se puede usar tantas veces como se desee, no así los saltos numerados de 01 a 99 los cuales se utilizan en pareja y pueden ser usados una sola vez en el programa.
2. Se puede realizar anidamientos utilizando saltos con número 00, es decir, es posible utilizar la instrucción JMP (04) 00 consecutivamente y concluir todas ellas en la misma JME (05) 00.
3. Los temporizadores en secciones enclavadas no se resetearán cuando la condición de ejecución para JMP sea OFF, pero si se utiliza el número de salto 00 el temporizador parará su operación. Si se utilizan números de salto de 01 a 99, los temporizadores seguirán su operación.
4. Dado que todas las instrucciones entre JUMP (04) y JME (05) se saltan, los números de salto de 01 a 99 se pueden utilizar para **reducir el tiempo de ciclo**. Para el salto con número 00, la CPU realiza una búsqueda de la siguiente JME con número 00, por lo tanto provoca un tiempo de ciclo más largo (cuando la condición es OFF) que para los otros saltos.

## CONSOLA DE PROGRAMACION MANUAL

Los PLCs OMRON tienen 3 modos de operación: PROGRAM, MONITOR y RUN.

La Cónsola de Programación le permite seleccionar el modo de operación y efectuar las operaciones permisibles en él.

### **Descripción**

- **Modo PROGRAM**

En modo PROGRAM la CPU está parada. Use este modo para:

- Introducir el programa
- Verificar el programa
- Insertar una instrucción en el programa
- Borrar una instrucción del programa
- Borrar la memoria
- Registrar la tabla de E/S

- **Modo MONITOR**

En modo MONITOR la CPU está corriendo. Use este modo para:

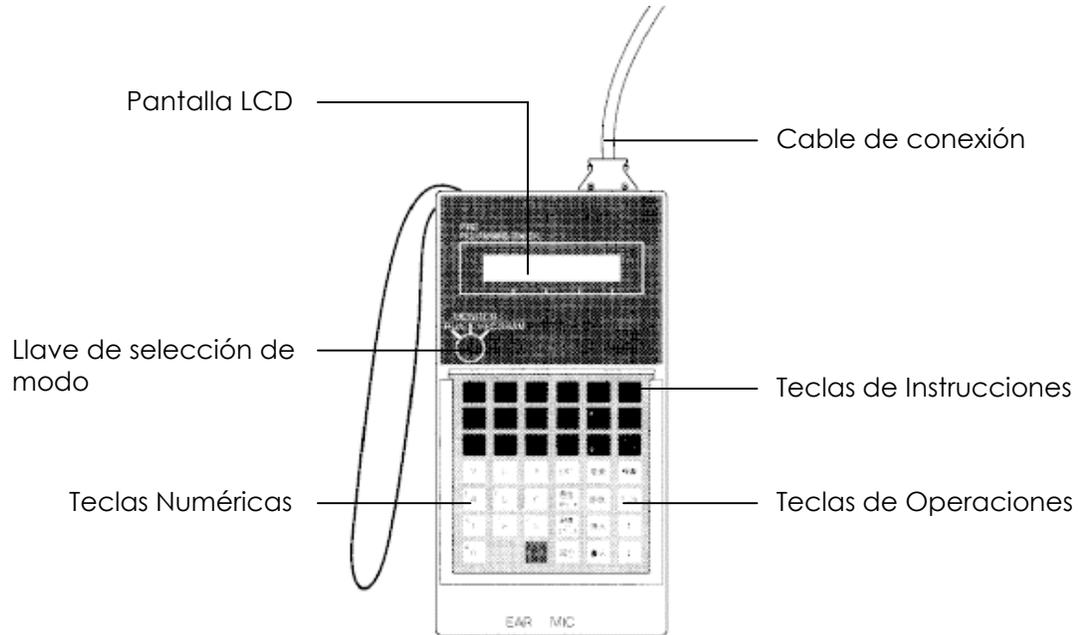
- Cambiar valores de temporizadores/contadores
- Monitorear el estado de E/S
- Monitorear canales, temporizadores y contadores
- Forzar contactos a ON/OFF
- Cambiar datos en áreas DMs, ARs, TCs, SRs e IRs
- **NOTA: Pueden escribirse datos en el PLC por HOST LINK**

- **Modo RUN**

En modo RUN la CPU está corriendo. Use este modo para monitorear el estado de E/S, canales, temporizadores, contadores, etc. No está permitido cambiar valores en modo RUN.

## Manejo de la Cónsola de Programación

### Cónsola de Programación Modelo C200H-PRO27



## Teclas y sus funciones

### • Teclas de Instrucciones

<b>FUN</b>	Se usa con las teclas numéricas para generar instrucciones	<b>TIM</b>	Genera la instrucción TIMER, o una condición de TIMER
<b>AND</b>	Genera la instrucción AND, o conexión normalmente abierta en serie	<b>TR</b>	Usada con la tecla OUT almacena una condición en un bit TR.. Con la tecla LD hace un llamado a la condición almacenada
<b>LD</b>	Genera la instrucción LOAD, o condición conectada a la barra de bus izquierdo	<b>DM</b>	Especifica una Memoria de Datos
<b>SFT</b>	Genera la instrucción SHIFT REGISTER	<b>LR</b>	Especifica un Relé de Enlace
<b>OR</b>	Genera la instrucción OR, o condición normalmente abierta en paralelo	<b>CH</b> *	Usada con la tecla SHIFT especifica un canal. Usada sola especifica direccionamiento indirecto de una Memoria de Datos
<b>OUT</b>	Genera la instrucción OUT, o bobina conectada a la barra de bus derecho	<b>SHIFT</b>	Se usa para especificar el carácter superior sobre las teclas
<b>NOT</b>	Se usa con las teclas LD, AND u OR para generar una condición normalmente cerrada	<b>HR</b>	Especifica un Relé de Mantenimiento
<b>CNT</b>	Genera la instrucción COUNTER, o una condición de contador.	<b>CONT</b> #	Especifica un bit cuando se usa con la tecla SHIFT, y especifica una constante cuando se usa sola

- **Teclas de Operaciones**

	Habilita el monitoreo de 3 canales
	Carga datos desde un cassette cuando se usa con la tecla SHIFT. Forza una condición a ON cuando se usa sola
	Graba datos en un cassette cuando se usa con la tecla SHIFT. Forza una condición a OFF cuando se usa sola
	Verifica el contenido de la tabla de E/S
	Se usa en modo MONITOR para cambiar el valor presente (PV) o valor fijado (SV)
	Se usa para buscar una condición o una instrucción en un programa
	Monitorea una instrucción, número de bit o número de canal
	Muestra la línea de programa precedente

	Muestra la próxima línea de programa
	Borra una línea
	Inserta una línea
	Introduce una línea o datos
	Inicializa la Cónsola de Programación

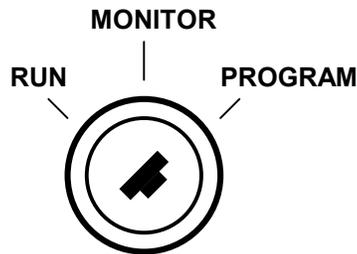
- **Teclas Numéricas**

	Introduce un dígito. Para introducir de la A a la F, presione de 0 a 5 mientras presiona la tecla SHIFT
	

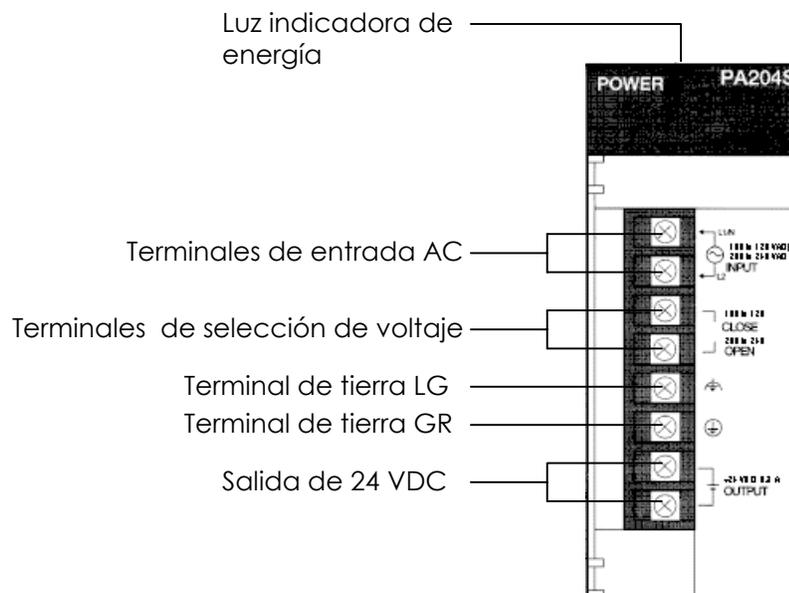
## Inicialización y arranque del PLC

### 1. Antes de conectar la fuente de alimentación

Coloque el selector de modo de la Cónsola de Programación en la posición PROGRAM.



### 2. Alimente con potencia AC



**Precaución:** Cuando la fuente de voltaje AC sea de 100 a 120 VAC, coloque en corto los terminales de selección de voltaje con la barra de metal. Cuando la fuente de voltaje AC sea de 200 a 240 VAC, elimine el corto en los terminales de selección de voltaje.

### 3. Introduzca la clave

CLR



MONTR



CLR

Primer mensaje mostrado después de alimentar

PASSWORD!

Indica el modo seleccionado con el selector de modo

PROGRAM BZ

Al obtener este mensaje se dice que se ha inicializado la Cónsola de Programación

00000

CLR

Presione la tecla CLR tantas veces como sea necesario para inicializar la cónsola, siempre que, el toque de la tecla (CLR) esté disponible.

Para hacer que el PLC y la cónsola estén listos para la programación, Ud. debe introducir la clave y aclarar (borrar) el área de memoria de programa.

### 4. Aclare el área de memoria de programa

Operación para borrar instrucciones o datos almacenados en la memoria de programa:

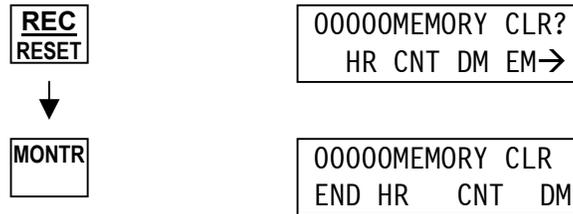
CLR



PLAY  
SET



00000



**Nota:** Si Ud. desea exceptuar las áreas HR, CNT o DM del borrado de memoria, presione la tecla HR, CNT O DM respectivamente antes de presionar la tecla MONTR en la operación de borrado que se muestra arriba.

Si Ud. desea exceptuar una parte de un programa del borrado de memoria, teclee el número del paso siguiente a esa parte (usando las teclas numéricas) antes de presionar la tecla MONTR en la operación de borrado.

## Escritura y Lectura de un programa

Programa a escribir:

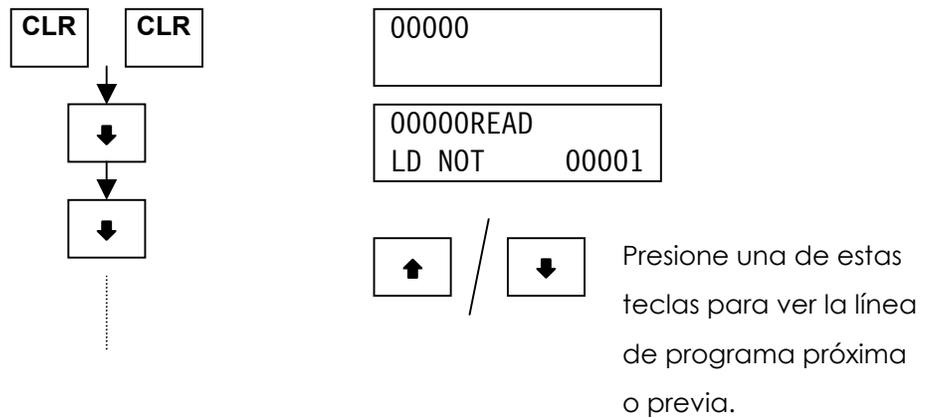
### Listado de mnemónicos (u Hoja de código)

PASO	INSTRUCCION	OPERANDO
00000	LD NOT	00001
00001	OR	00100
00002	AND	00000
00003	OUT	00100
00004	LD	00100
00005	AND	00001
00006	OR	00300
00007	AND NOT	00103
00008	OUT	00300
00009	LD NOT	00003
00010	OR	00102
00011	AND	00002
00012	OUT	00102
00013	END	





- Después de escribir el programa, verifíquelo con la hoja de código



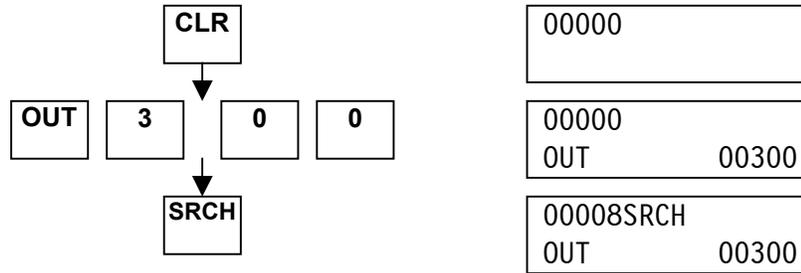
## Búsqueda, Inserción y Borrado de un paso de programa

### Listado de mnemónicos

PASO	INSTRUCCION	OPERANDO	<u>Adición al programa original</u>	
00000	LD NOT	00001	LD NOT	00002
00001	OR	00100	OR	00104
00002	AND	00000	AND	00003
00003	OUT	00100	OUT	00104
00004	LD	00100	LD	00104
00005	AND	00001	AND	00002
00006	OR	00300	OUT	00105
00007	AND NOT	00103		
00008	OUT	00300		
00009	LD NOT	00003		
00010	OR	00102		

00011 AND 00002  
 00012 OUT 00102  
 00013 END

## 1. Búsqueda de una instrucción

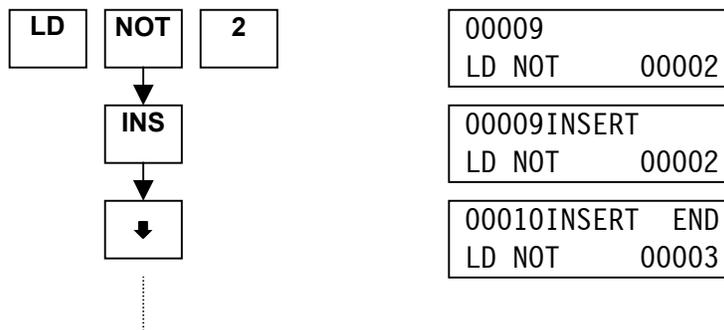


Al pulsar SRCH en la operación de búsqueda se mostrará el número del paso de programa de la primera ocurrencia de la instrucción. Pulse nuevamente la tecla SRCH para encontrar la próxima ocurrencia de la instrucción. La búsqueda finaliza al encontrar la instrucción END.

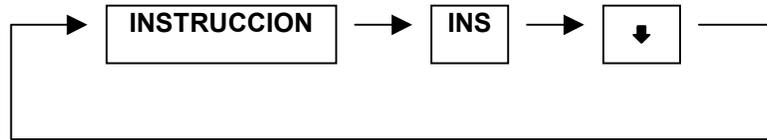
## 2. Inserción de una instrucción

Coloque el selector de modo en la posición PROGRAM, busque la instrucción del lugar de inserción (paso N° 00009).

Inserte el programa:



El procedimiento se repite para insertar las instrucciones siguientes:



### 3. Borrado de una instrucción

Ubíquese en la instrucción que desea borrar, presione la tecla DEL y luego la tecla FLECHA-ARRIBA.

## Monitoreo de estado, forzado a ON y forzado a OFF de E/S

### 1. Monitoreo de múltiples puntos de E/S

Coloque el selector de modo en la posición MONITOR y presione la tecla CLR.

Ejemplo: Monitorear el estado (ON/OFF) del punto 00001



**Nota:** Se pueden monitorear hasta 3 puntos de E/S a la vez

Ejemplo: Monitorear el estado de los puntos 00002 y 00101

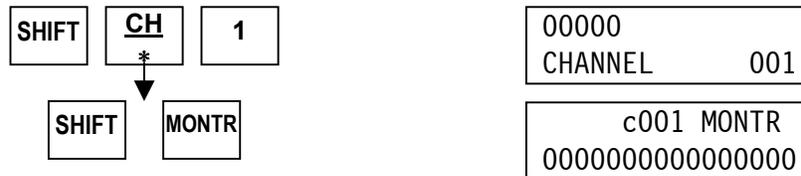


**Ejercicio:** Intercambie a ON y OFF la posición de los conmutadores 01 y 02 montados en el módulo de entrada canal 0 y observe la pantalla.

## 2. Monitoreo de canales

Coloque el selector de modo en la posición MONITOR y presione la tecla CLR.

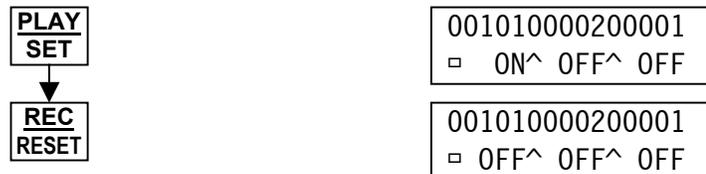
Ejemplo: Monitorear el canal 1.



Presione la tecla FLECHA-ABAJO para monitorear el canal siguiente.

## 3. Forzado de un punto a ON u OFF desde la Cónsola de Programación

El efecto es sobre el punto monitoreado más a la izquierda



El punto está en ON mientras se mantenga presionada la tecla PLAY/SET. Análogo para OFF.

### Forzado permanente a ON y forzado permanente a OFF





La tecla NOT cancela las funciones SET y RESET permanente.

## Cambio del Valor Fijado (SV) de temporizadores

El siguiente procedimiento explica cómo cambiar el valor SV de un temporizador.

Esta operación puede ser realizada en los modos PROGRAM y MONITOR. Como ejemplo, introduzca y verifique el siguiente programa:

### Programa de un temporizador ON-DELAY

```

00000 LD      00000
00001 TIM    000
           #0100
00002 LD      TIM 000
00003 OUT    00100
00004 END (01)
    
```

1. Aclare la pantalla presionando: CLR, CLR.
2. Busque el temporizador presionando: TIM, 000, SRCH. Esto le conduce al paso 00001 donde TIM 000 está localizado.
3. Presione la tecla FLECHA-ABAJO. Esto le conduce al área del valor fijado: #0100.
4. Presione la tecla CHG para cambiar el valor.
5. Introduzca el nuevo valor: 0200, WRITE.
6. Para llamar al nuevo valor fijado presione: CLR, CLR, TIM, 000, MONTR.

## **Cambio del Valor Fijado (SV) de contadores**

El procedimiento para cambiar el valor fijado de un contador es exactamente el mismo que para un temporizador. Simplemente sustituya el número de contador apropiado donde se indica un temporizador.

## PLCs CPM1A/CQM1

### Direccionamiento de E/S en los PLCs Serie CPM1A

Las unidades CPM1A se caracterizan por ser una construcción de una sola pieza que incluye CPU, fuente de poder y terminales de E/S. Las opciones de densidad son 10, 20, 30 o 40 puntos de E/S. Hasta tres módulos de expansión pueden utilizarse con CPUs de 30 o 40 puntos de E/S para un máximo de 100 puntos.

La tabla siguiente muestra la dirección de memoria utilizada por los terminales de E/S en la CPU de un CPM1A y en unidades de expansión.

No. de Terminales de E/S en la unidad de CPU		10	20	30	40
Terminales en la unidad de CPU	Entradas	6 puntos: 00000 a 00005	12 puntos: 00000 a 00011	18 puntos: 00000 a 00011 00100 a 00105	24 puntos: 00000 a 00011 00100 a 00111
	Salidas	4 puntos: 01000 a 01003	8 puntos: 01000 a 01007	12 puntos: 01000 a 01007 01100 a 01103	16 puntos: 01000 a 01007 01100 a 01107
Terminales en la unidad de expansión	Entradas	---	---	12 puntos: 00200 a 00211	12 puntos: 00200 a 00211
	Salidas	---	---	8 puntos: 01200 a 01207	8 puntos: 01200 a 01207
	Entradas	---	---	12 puntos: 00300 a 00311	12 puntos: 00300 a 00311
	Salidas	---	---	8 puntos: 01300 a 01307	8 puntos: 01300 a 01307
	Entradas	---	---	12 puntos: 00400 a 00411	12 puntos: 00400 a 00411
	Salidas	---	---	8 puntos: 01400 a 01407	8 puntos: 01400 a 01407

La tabla siguiente muestra las áreas de memoria disponible por el usuario:

PLC	Area E/S	Bits de Trabaj.	Area SR	Area TR	Area HR	Area AR	Area LR	Area TC	Area DM Lect/Escrit
CPM1A	---	200-231	232-255	TR0-TR7	HR00- HR19	AR00- AR15	LR00- LR15	TIM/CNT 000-127	DM0000- DM0999  DM1022- DM1023

## Direccionamiento de E/S en el PLC CQM1

EL CQM1 es un sistema modular sin Rack de tipo conectar-y-bloquear. Cada módulo (fuente de poder, CPU o módulo de E/S) es conectado a los lados para formar un PLC simple el cual es montado normalmente en un riel simétrico.

El direccionamiento se inicia en la izquierda con 16 puntos de entrada que trae incluidos la CPU.

CPU	Area	Canales	Restricción
CPU11/21	Entradas	CH000 - 007	Un máximo de 8 canales (128 bits) pueden ser usados para bits de E/S.
	Salidas	CH100 - 107	
CPU4	Entradas	CH000 - 015	Un máximo de 16 canales (256 bits) pueden ser usados para bits de E/S.
	Salidas	CH100 - 115	

Los canales son asignados de izquierda a derecha. Cada vez que un módulo es adicionado, el próximo canal de entrada o salida disponible es asignado.

### Clasificación de los Módulos de Entrada/Salida

Los módulos se clasifican en dos grupos: estándares y dedicados.

**Estándares:** Son todas las unidades de E/S discreta de 8, 16 y 32 puntos.

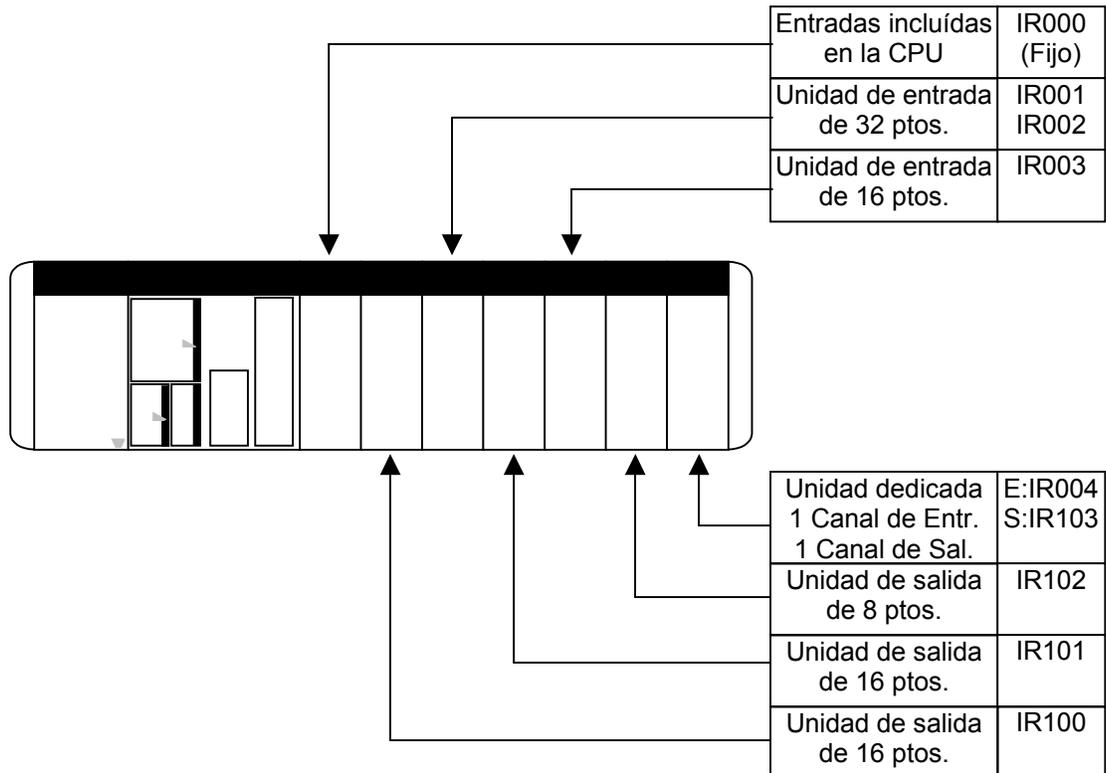
**Dedicados:** Son todas las unidades especiales disponibles para el sistema:

- Módulo I/O Link
- Módulo Interface B7A
- Módulo A/D
- Módulo D/A
- Módulo de Control de Temperatura
- Módulo Sensor
- Esclavo Device-Net

### Ejemplo de direccionamiento en el PLC CQM1

El direccionamiento de entradas se inicia en el canal 000.

El direccionamiento de salidas se inicia en el canal 100.



La tabla siguiente muestra las áreas de memoria disponibles por el usuario:

PLC	Area E/S	Bits de Trabaj.	Area SR	Area TR	Area HR	Area AR	Area LR	Area TC	Area DM Lect/Escrit
CQM1	---	016-195 116-195 216-219 224-229	244-245	TR0-TR7	HR00- HR99	AR00- AR27	LR00- LR63	TIM/CNT 000-511	DM0000- DM1023 *DM1024- DM6143

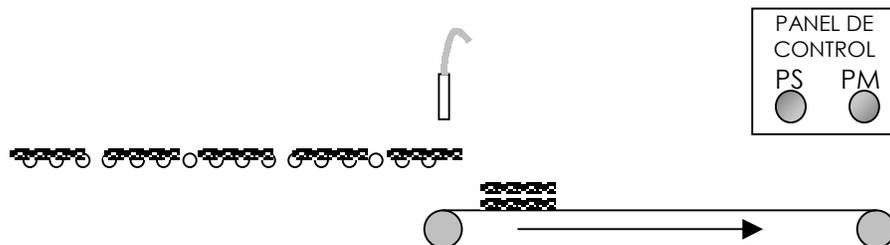
\*CPU 4

## EJERCICIOS DE APLICACION

### Ejercicio No 1

Realizar un programa de PLC que realice lo siguiente:

- Al pulsar el pulsador de marcha arranca el motor #1 que mueve un tren de rodillos de transporte.
  - Sobre los rodillos van piezas que serán contadas y al llegar a 6 unidades el motor #1 debe parar 6 segundos para dar oportunidad a una banda transportadora de llevarse el conjunto de piezas, después de lo cual arrancará nuevamente. Al mismo tiempo el motor que mueve la banda transportadora (motor #2) arrancará y durará encendido 5 segundos, después de lo cual parará de nuevo.
  - El sistema podrá ser detenido usando un pulsador de parada.
- Ver figura.



#### Elenco de Entradas y Salidas

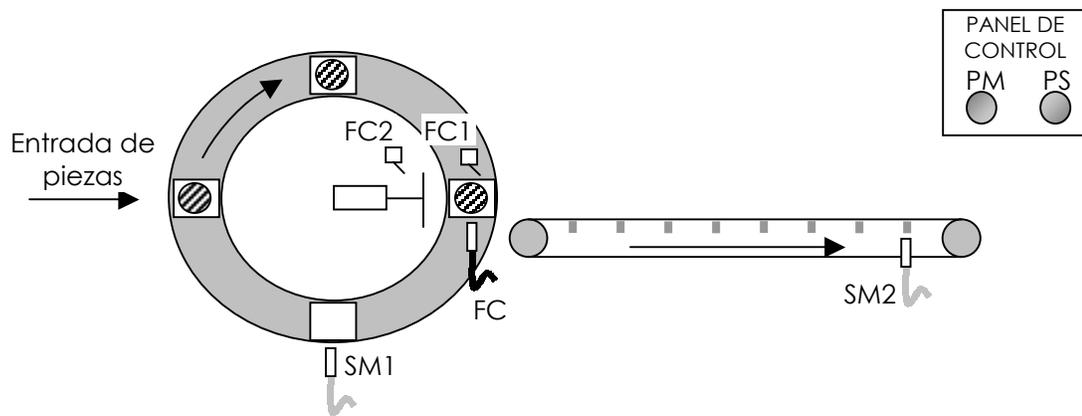
Entradas	Nombre	Descripción
00000	PM	Pulsador de Marcha
00001	PS	Pulsador de Stop
00002	FC	Fotocelda de Conteo

Salidas	Nombre	Descripción
00100	M1	Motor de rodillos
00101	M2	Motor de Banda Transportadora

## Ejercicio No 2

Un carrusel está girando y se detiene cada vez que un sensor magnético de proximidad (SM1) detecta un tornillo que está puesto sobre cada uno de los cubículos y una fotocelda (FC) indica que una pieza está en posición. Al detenerse, un cilindro neumático (SCN) es accionado para expulsar la pieza del cubículo. Al llegar el cilindro adelante, un final de carrera (FC1) da la señal para accionar el motor que mueve la banda transportadora y da la señal de retorno del cilindro. La banda avanza hasta la siguiente posición indicada por un sensor magnético de proximidad (SM2). El Carrusel reinicia el movimiento cuando un final de carrera (FC2) indica que el cilindro neumático retornó a su posición de espera. Ver figura xx

Realice el programa que controla el proceso descrito.



### Elenco de Entradas y Salidas

Entradas	Nombre	Descripción
00000	PM	Pulsador de Marcha
00001	PS	Pulsador de Stop
00002	FC1	Final de Carrera cilindro adelante
00003	FC2	Final de Carrera cilindro atrás
00004	SM1	Sensor Magnético de Prox. del Cubículo
00005	SM2	Sensor Magnético de Prox. Posicionamiento de la Banda

---

<b>Salidas</b>	<b>Nombre</b>	<b>Descripción</b>
00100	MC	Motor del Carrusel
00101	MB	Motor de Banda Transportadora
00102	SCN	Solenoide del Cilindro Neumático