

CAPITULO V

Programación del PLC

Introducción

Cuando hablamos de los lenguajes de programación nos referimos a diferentes formas en las que se puede escribir el programa del 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 o diagramas de escalera
- Esquema funcional

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

Es un lenguaje gráfico, derivado del lenguaje de relevadores, 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.

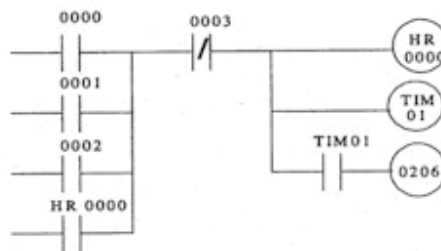
CONCEPTOS GENERALES DE LOS DIAGRAMAS DE ESCALERA

Un programa en DIAGRAMA DE ESCALERA o esquema de contactos, la constituyen una serie de ramas de contactos.

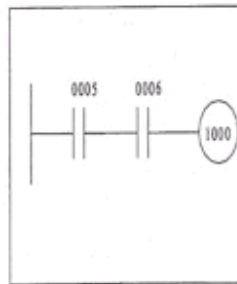
Los símbolos básicos son:



- **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.**

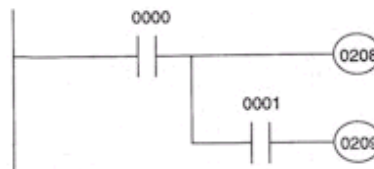


Dirección	Instrucción	Datos
0000	LD	0005
0001	AND	0006
0002	OUT	1000
0003	END	

- 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.



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 de los microprocesadores.

Ejemplo: La lista de instrucciones son las del autómata 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.

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 mas expresivos que si se utiliza para el mismo fin el lenguaje en lista de instrucciones o mnemónico.

5.1 Funciones Lógicas

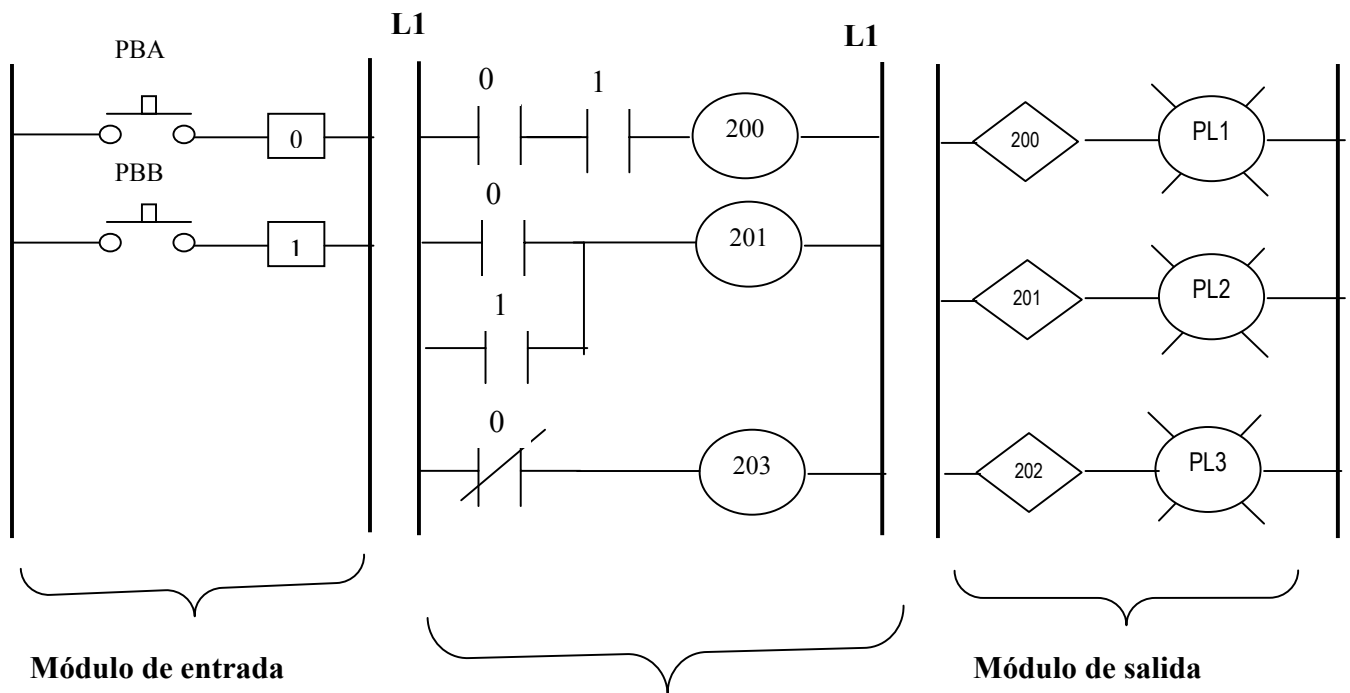
La implementación de las funciones lógicas es muy simple como ya hemos visto al final del capítulo anterior .

En esta sección vamos a mostrar como se programa usando diagrama de escaleras o diagrama de contactos y listado de instrucciones las funciones lógicas básicas .

Supongamos dos variables de entrada usando los push buttons PBA y PBB conectados a las entradas 0 y 1 del PLC. Entonces para implementar con un programa las funciones AND , OR y Not tenemos que identificar las variables de entrada, los contactos internos a los que estarán relacionados y la salida afectada.

Dentro de las funciones lógicas podemos incluir los relevadores control maestro que nos permiten manejar en forma conveniente una zona del programa sin afectar las partes no involucradas dentro de el dominio.

PL1= PBA.PBB
PL2=PBA+PBB
PL3= not(PBA)



Programa en diagrama de contactos

```

LOD 0
AND 1
OUT 200
LOD 0
OR 1
OUT 201
LOD NOT 0
OUT 202
END

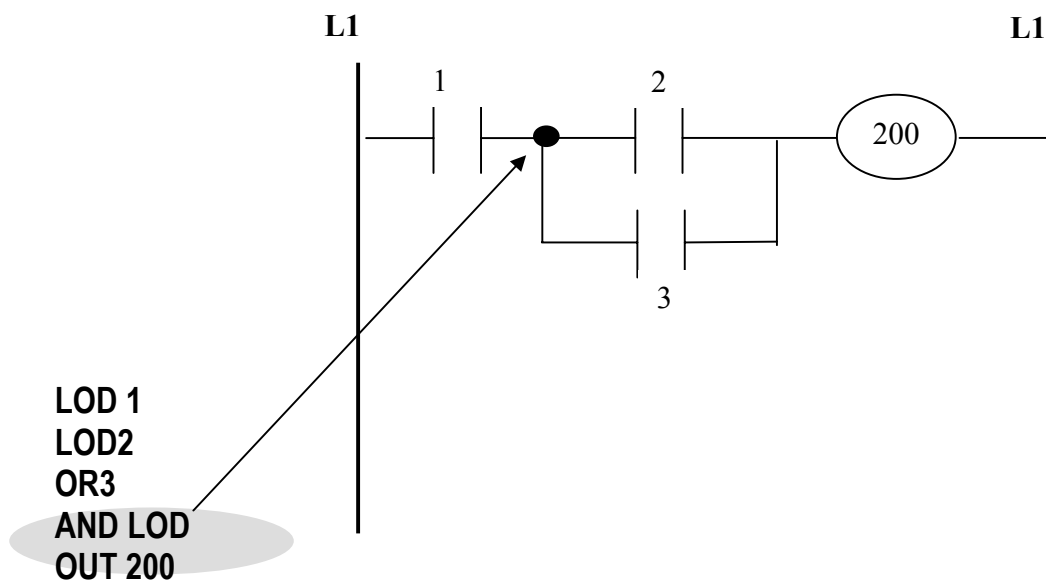
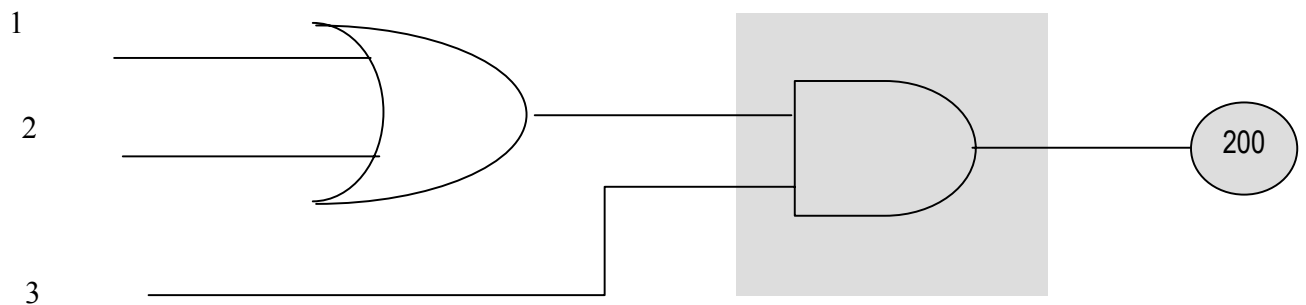
```

PROGRAMA EN DIAGRAMA DE CONTACTOS
PARA EL MICRO1 DE SQUARE D

Para la implementación de funciones lógicas compuestas por circuitos serie paralelo y paralelo seri, se tiene las funciones AND LOD y OR LOD respectivamente.
Estas funciones se programan como a continuación mostramos.

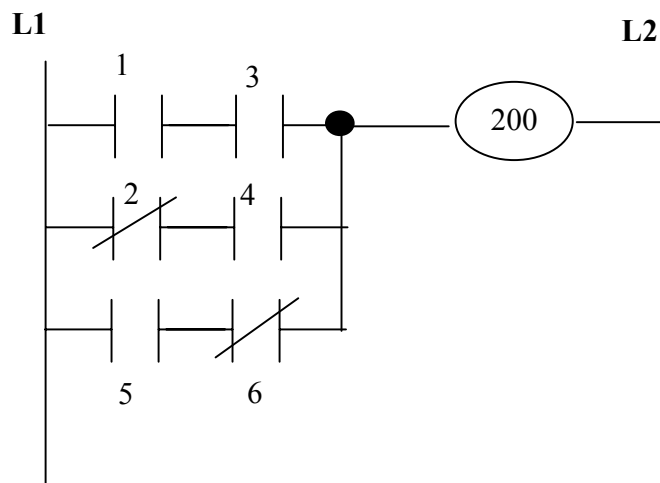
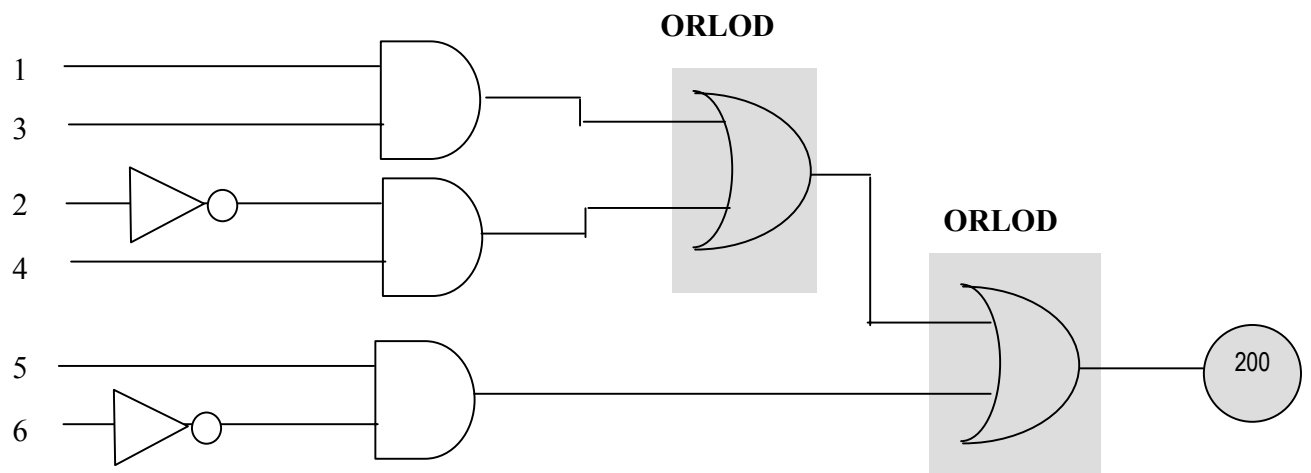
Ejemplo de la implementación de una función lógica AND LOD

Equivalente con compuertas de una función AND LOD



Ejemplo de la implementación de una función lógica OR LOD

Equivalente con compuertas de una función OR LOD



```

LOD 1
AND3
LOD NOT 2
AND 4
OR LOD
LOD 5
AND NOT 6
OR LOD
OUT 200

```


5.2 Relevador Control Maestro

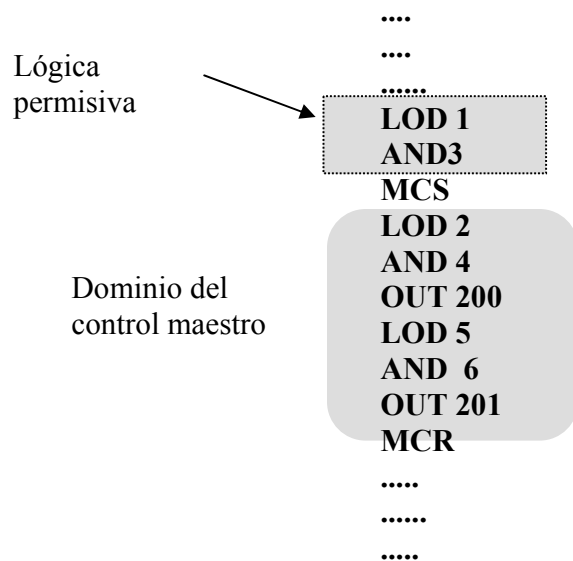
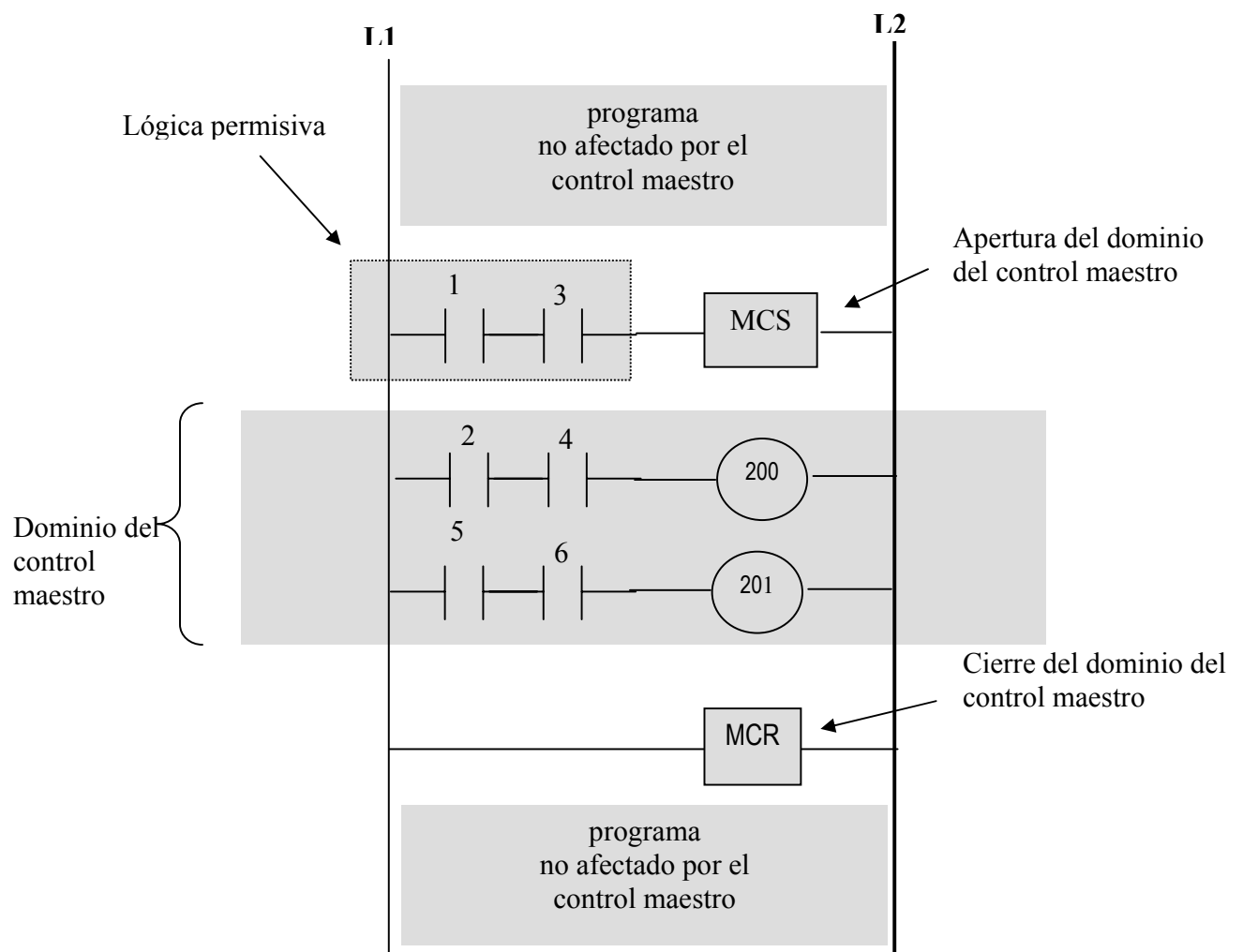
La instrucción de salida control maestro es usada para activar o desactivar la ejecución de un grupo o zona de peldaños de la escalera (véase la figura de abajo).

El dominio del control maestro, se encierra como en brackets, esto es, se enmarca la zona o grupo de los peldaños de la escalera que se desea controlar con un bracket de apertura y uno de cierre.

Siendo más explícito, el inicio del dominio o la apertura de los “brackets” se hace con un bloque de función MCS (Master Control Set), precedido de una lógica permisiva. El final del dominio del Control maestro es indicado con un bloque MCR (Master Control Reset). Todos los peldaños del diagrama de escalera que queden entre estas dos funciones , MCR y MCS, son los únicos que se ven afectados por el control maestro.

Cuando fluye “potencia” al MCS, la lógica encerrada operará normalmente. Si no hay flujo de “potencia” en MCS, todas las bobinas serán forzadas a su estado de apagadas sin importar las condiciones de la lógica interna.

La cantidad de lógica controlada por un control maestro es limitada solo por la memoria proporcionada para este fin.



5.3 Temporizadores (TIMERS)

Conceptualmente un Temporizador (TIMER) es un dispositivo electrónico utilizado para proveer señales de base de tiempo o para generar señales de acción retardada variable.

Un Temporizador o Timer digital consiste generalmente de un contador decreciente en donde cada decremento en uno del preset del contador, será realizado a una frecuencia conocida (veces por segundo) y llegar a cero se activa un relevador interno o uno de salida.

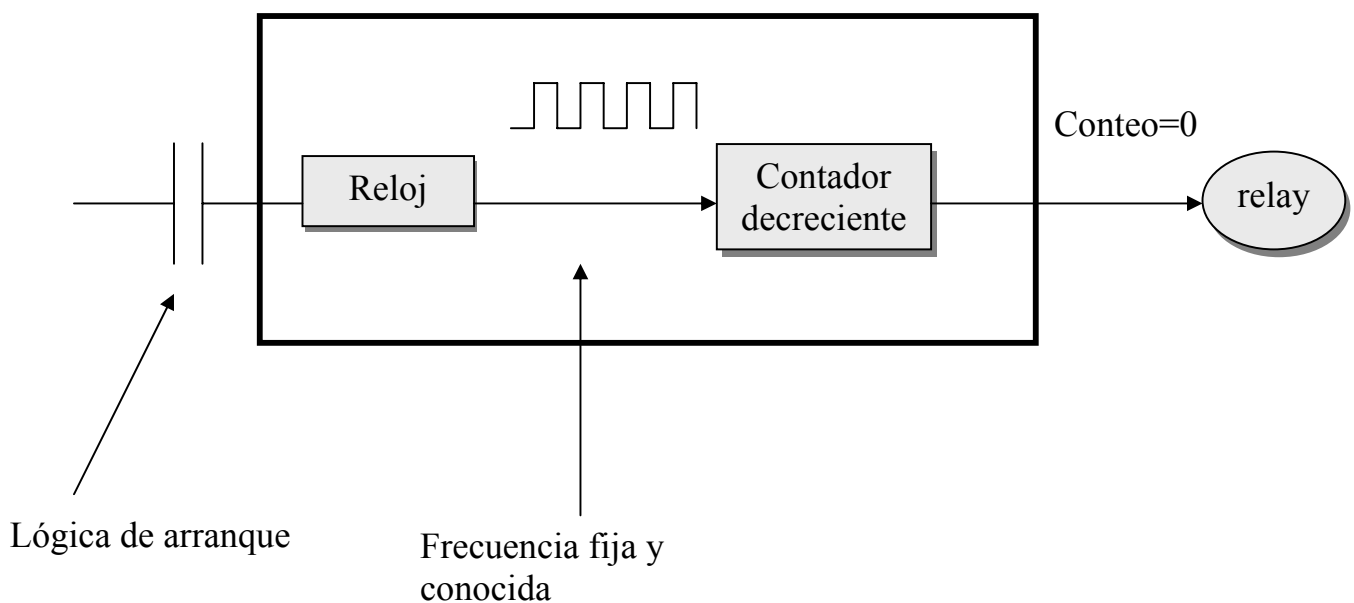
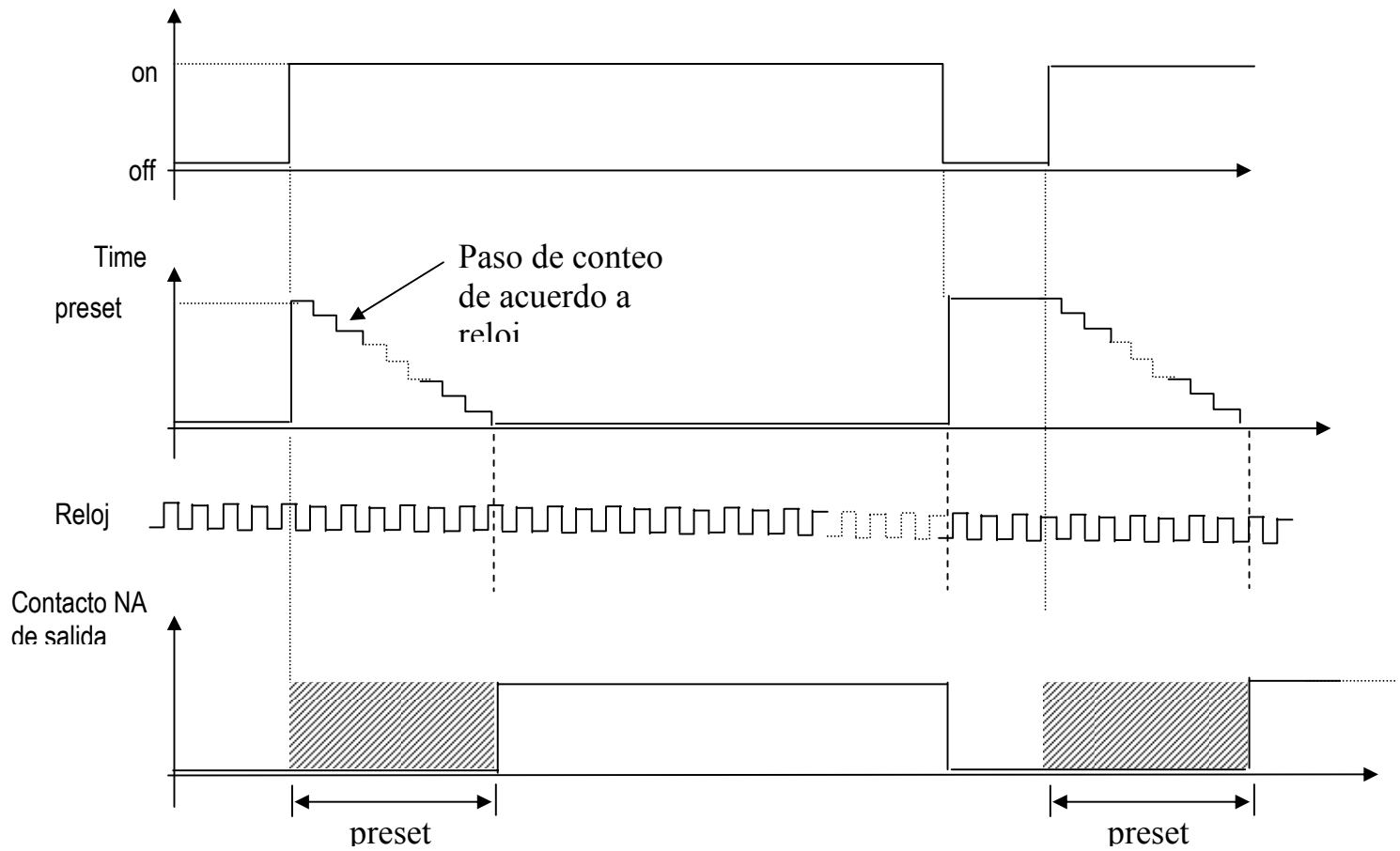
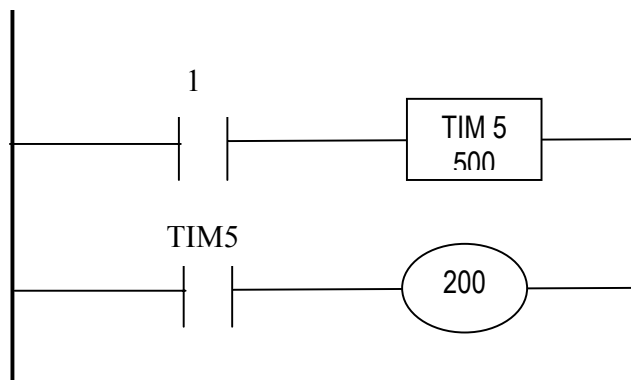


Diagrama de bloques de las componentes de un timer conceptual

Lógica de arranque-



Descripción de la operación de un timer



LOD 1
TIM 5
500
LOD TIM5
OUT 200

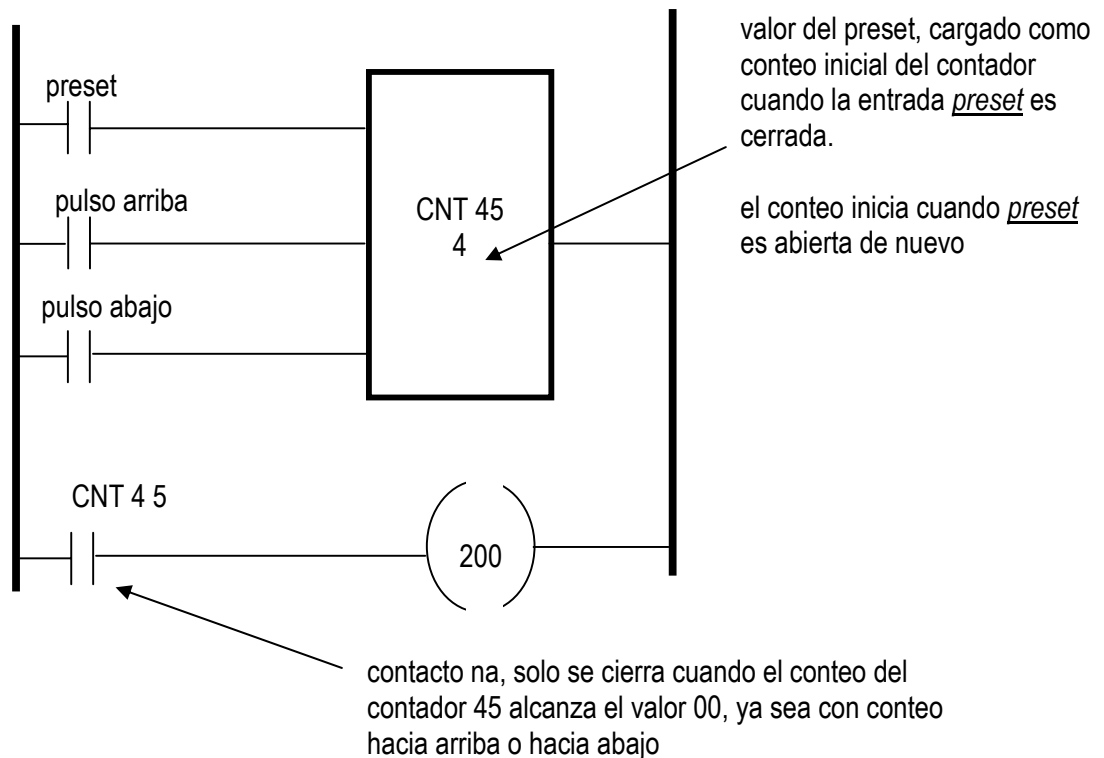
5.4 Contadores

Para la realización de tareas de conteo de eventos externos los PLC's tienen los contadores . Estos los hay de diferente tipo, al igual que los que se fabrican en circuitos TTL, los hay contadores incrementales, decrementales conteo UP DOWN , etc.

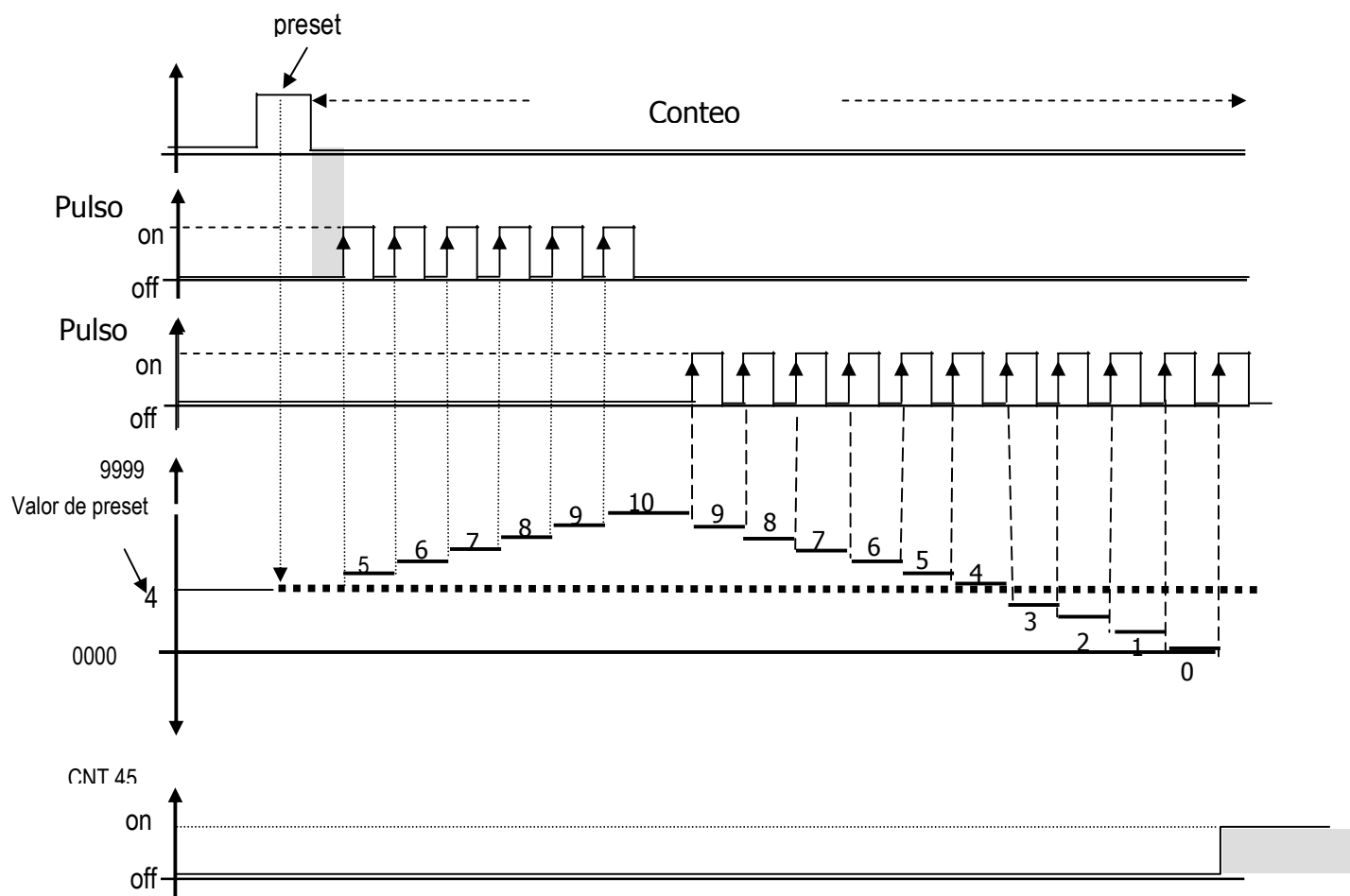
Los diferentes PLC's nos proporcionan algunos o a veces todos estos tipos de contadores.

Aquí mostramos algunos de los usados por el MICRO1 de SquareD.

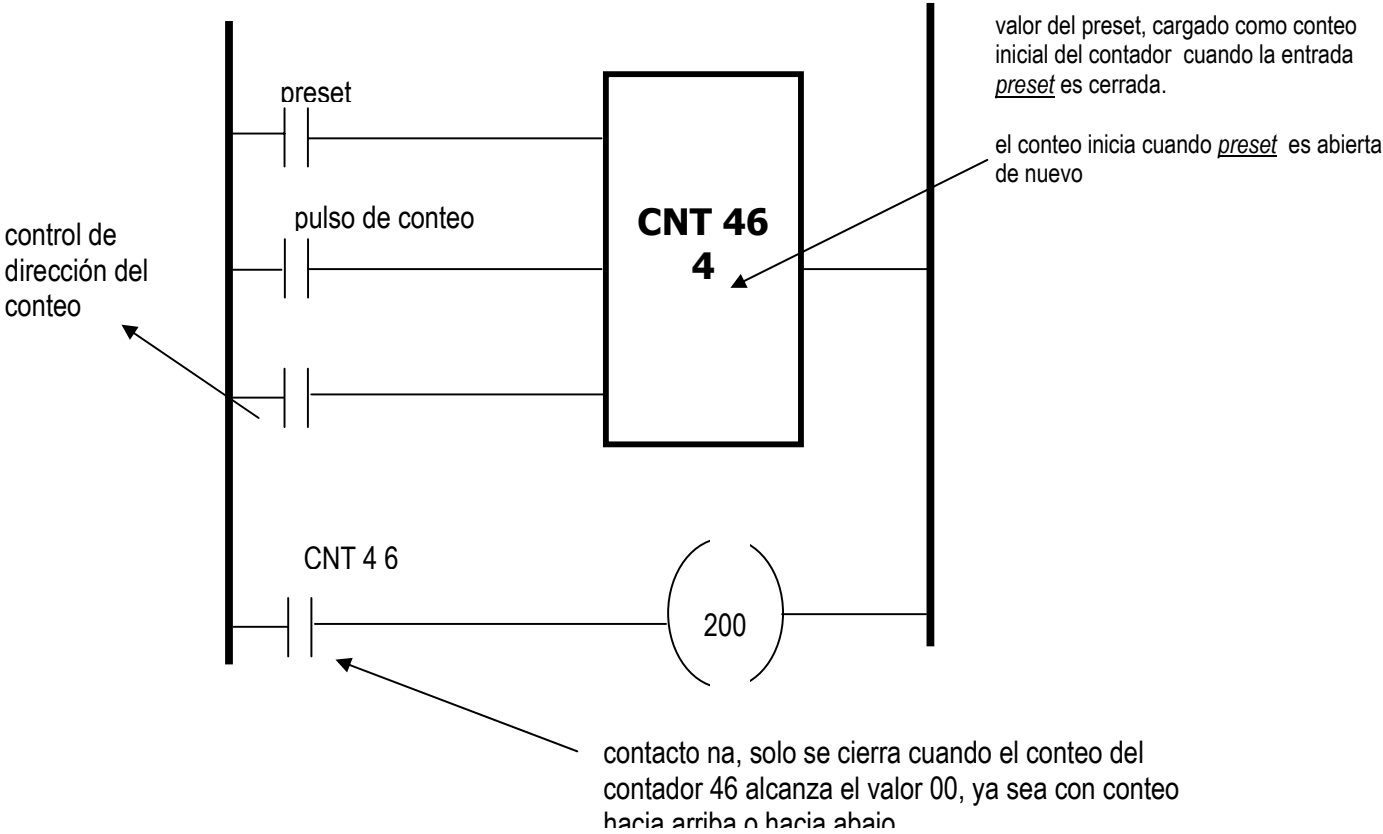
Contador reversible (Up-Down) Contador 45



Contador reversible (Up-Down) Contador 45



Contador reversible (Up-Down)
Contador 46



Contador reversible (Up-Down)
con control de dirección
Contador 46

