

Universidad de Chile
Facultad de Ciencias Forestales
Dpto. de Tecnología de la Madera

Apunte N° 1
Automatización y Control de Sistemas

CONTROLADORES LÓGICOS PROGRAMABLES

Junio 2001
Christian Ramírez Cortés

TABLA DE CONTENIDOS

1	INTRODUCCIÓN	1
1.1	DEFINICIÓN	1
1.2	HISTORIA	1
2	TEORÍA DE OPERACIÓN	3
2.1	CONFIGURACIÓN INTERNA	3
2.2	MODO DE OPERACIÓN	4
2.3	TIEMPO DE RESPUESTA	5
2.4	EFFECTO DEL TIEMPO DE RESPUESTA	6
3	CREACIÓN DE PROGRAMAS	9
3.1	LOS RELÉS	9
3.2	REEMPLAZANDO RELÉS	9
3.3	INSTRUCCIONES BÁSICAS	11
3.4	EJEMPLO DE APLICACIÓN	13
4	SET DE INSTRUCCIONES PRINCIPALES	17
4.1	INSTRUCCIONES CERROJO	17
4.2	CONTADORES	18
4.3	TEMPORIZADORES	20
4.4	EXACTITUD DEL TIEMPO	23
4.5	CAMBIO DE REGISTROS	26
5	CONEXIÓN DEL PLC	28
5.1	ENTRADAS DE CORRIENTE CONTINUA	28
5.2	SALIDAS A RELÉS	30
5.3	SALIDAS A TRANSISTORES	32

1 INTRODUCCIÓN

1.1 Definición

Un PLC, denominado así por las siglas en inglés de Controlador Lógico Programable, es un aparato que fue inventado para remplazar los circuitos secuenciales de relés¹ utilizados en el control de máquinas. El PLC trabaja revisando sus entradas, y dependiendo del estado de éstas, manipula el estado de sus salidas, encendiéndolas o apagándolas. El usuario debe ingresar un programa, usualmente vía software, que lleva a obtener los resultados de operación deseados.

Los PLC son usados en varias aplicaciones de tareas cotidianas. Su uso involucra operaciones de maquinado, embalaje, manejo de materiales, ensamblaje automatizado y en casi todas las tareas que requieren aplicar movimientos repetitivos. Su uso produce ahorro de costos y tiempo, además de evitar a los operarios la realización de tareas alienantes o peligrosas. Casi cualquier operación que requiera controlar aparatos eléctricos necesita la aplicación de un PLC.

Como ejemplo, supongamos que se necesita encender un relé por 5 segundos y luego apagarlo. Para hacer esto bastaría un temporizador. Ahora si se desea hacer algo similar con 10 relés, sería necesario contar con 10 temporizadores. Si a esto agregamos que necesitamos monitorear en todo momento e independientemente cuantas veces se enciende cada relé, deberíamos agregar un contador para cada uno. De esta forma, mientras más grande es la complejidad del proceso se hace más evidente la necesidad de contar con un PLC.

1.2 Historia

Los PLC fueron introducidos a fines de los años 60. La razón de su aparición fue la necesidad de eliminar los complicados y costosos sistemas de control de máquinas basados en relés. Bedford Associates propuso algo llamado Controlador Modular Digital (MODICON) a la General Motors. Al mismo tiempo otras compañías propusieron esquemas basados en computadoras, uno de los cuales fue PDP-8. El MODICON 084 llegó a ser el primer PLC en producción a escala comercial.

Cuando hay cambios en los requerimientos de producción estos involucran al sistema de control. Estas modificaciones llegan a ser muy caras si los cambios requeridos son frecuentes. Debido a que los relés son aparatos mecánicos, estos tienen una vida limitada que obliga apearse a estrictos programas de mantención. El encontrar las fallas en uno de estos sistemas es una tarea bastante tediosa cuando involucra una cantidad importante de relés.

Estos nuevos controladores debían ser fáciles de programar por los ingenieros de mantención o de planta. También ser capaces de funcionar en los agresivos ambientes industriales. La forma de lograr esto fue usar técnicas de programación con las que los

¹ Un relé es básicamente un switch que conecta a algún componente eléctrico mediante la aplicación de un pequeño voltaje, denominado señal de control. La aplicación de la señal puede llevar al relé un estado encendido o apagado, según sea el tipo de relé (abierto por defecto, o cerrado por defecto).

programadores estaban familiarizados y reemplazar los relés mecánicos con elementos electrónicos de estado sólido.

A mediados de los 70 los PLC comenzaron a tener habilidades de comunicación. El primer sistema de comunicación fue el MODBUS de MODICON. Ahora los controladores se podían comunicar entre sí para coordinar el accionar de un conjunto de máquinas. También se les agregaron capacidades de transmitir y recibir voltajes variables que le permiten recibir señales análogas. Infortunadamente la carencia de estandarización en estos sistemas, unido a los continuos cambios de tecnología convirtió las comunicaciones de PLC en una pesadilla de protocolos y redes físicas, lo que originó la decadencia de su aplicación.

Durante los años 80 se apreció un intento por estandarizar las comunicaciones con el protocolo de automatización de la manufactura de la General Motors (MAP). Al mismo tiempo se tendió a la miniaturización de los equipos y la utilización de lenguajes simbólicos de programación en computadoras personales o programadoras portátiles. Hoy en día los PLC más pequeños son del tamaño de un solo relé.

En los 90 se ha visto una reducción gradual en la introducción de protocolos nuevos, y la modernización de las capas físicas de algunos de los protocolos más populares que sobrevivieron a los años 80. El último estándar ha tratado de reunir los lenguajes de los PLC bajo un estándar internacional único. Ahora se cuenta con controladores programables en función de diagramas de bloques, listas de instrucciones, lenguaje de programación C o texto estructurado, todo al mismo tiempo. También se ha visto que se están introduciendo computadoras personales para reemplazar en algunas aplicaciones específicas a los PLC. Es el caso de la General Motors, que ha llevado sus sistemas a control basado en computadoras.

2 TEORÍA DE OPERACIÓN

2.1 Configuración Interna

Un PLC consiste básicamente en una CPU (Unidad Central de Proceso), áreas de memoria, y circuitos apropiados para gestionar los datos de entrada y salida. Bajo el punto de vista de reemplazar a los antiguos circuitos de relé, el PLC se puede considerar como una caja llena de miles de relés, contadores, temporizadores y lugares para almacenamiento de datos. Esto es sólo una analogía, ya que estos elementos son emulados por el PLC y no existen realmente.

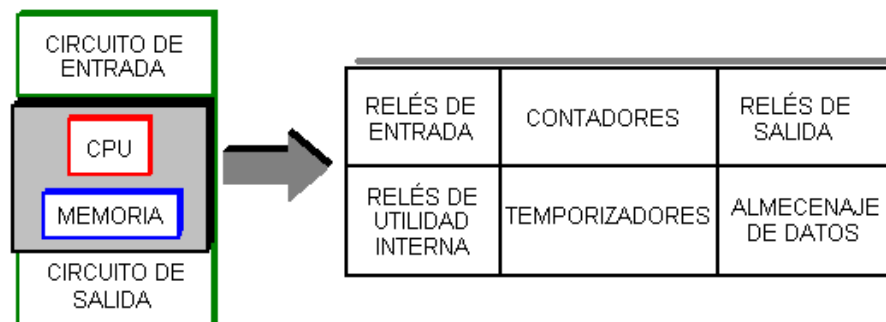


Figura 1. Esquema de los componentes principales de un PLC.

Analizando el esquema presentado podemos establecer las tareas que cada unidad realiza.

2.1.1 Relés de entrada

Están conectados al mundo exterior. Estos existen realmente y reciben señales desde sensores y switch de límites de carrera, de operación, etc. Por lo general en un PLC moderno han sido reemplazados por transistores.

2.1.2 Relés de tareas internas

Estos relés no reciben señales externas y no existen físicamente. Estos relés son emulados por la CPU y son los parte de los PLC que ha cambiado en las unidades modernas. El estado de estos relés emulados son los que realizan las lógicas de control que luego se traducen a los relés de salida. Por el hecho de ser emulados por la CPU, programando las tareas de ésta es posible establecer nuevas lógicas de control, lo que brinda flexibilidad al sistema.

2.1.3 Contadores

Estos tampoco existen físicamente. Estos son contadores simulados que son programados para contabilizar pulsos. Típicamente estos contadores cuentan incrementándose, decrementándose, o ambas situaciones a la vez. Debido a que son emulados existen limitaciones sobre su velocidad de conteo. Algunos fabricantes de PLC incluyen contadores de alta velocidad basados en un circuito anexo. En este caso se puede decir que los contadores existen físicamente.

2.1.4 Temporizadores

Este componente tampoco existe. Los hay de muchas clases e incrementos. El tipo más común es el *retardo positivo* (on delay). Otros incluyen *retardo negativo* (off delay) y del tipo de *retentivo* y *no retentivo*. Los incrementos varían de 1 milisegundo a 1 segundo.

2.1.5 Relés de salida

Estos están conectados al mundo exterior y existen físicamente. Se encargan de enviar las señales de encendido y apagado a los solenoides (bobinas), luces o actuadores. Estos pueden ser transistores, relés o triacs, dependiendo del modelo de PLC particular.

2.1.6 Almacenadores de datos

Típicamente estos elementos corresponden a registros que simplemente almacenan datos. Son utilizados temporalmente para almacenar datos para operaciones matemáticas o manipulación de datos en general. También son utilizados para almacenar datos cuando se ha cortado el suministro de energía del PLC. De esta forma, al retornar la energía, el PLC recuerda en que estado estaba antes de la falla o interrupción.

2.2 Modo de Operación

Un PLC trabaja *barriendo* continuamente un programa. Podemos entender estos ciclos de barrido como la ejecución consecutiva de tres pasos principales. Típicamente hay más de tres pasos, pero podemos enfocarnos en estos ya que dan una buena idea del funcionamiento.

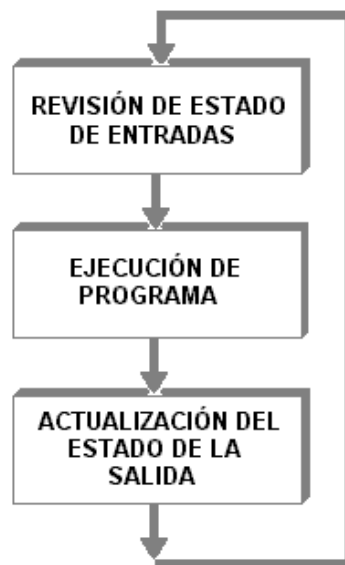


Figura 2. Pasos principales en la operación de un PLC.

2.2.1 Revisar estado de las entradas

Como primer paso el PLC revisa cada entrada para determinar si esta encendida o apagada (entrada binaria o de dos estados). Revisa las entradas desde la primera a la última, y graba estos estados en la memoria del PLC para ser utilizados en el paso siguiente.

2.2.2 Ejecución del programa

Ahora el PLC ejecuta el programa preestablecido instrucción por instrucción y en el orden en que se determinó. Como ya se ha revisado el estado de las entradas, el programa puede tomar decisiones basado en los valores que fueron guardados. Las decisiones que toma el programa, en última instancia, corresponden a los valores que van a tomar cada una de las salidas. Estos valores son almacenados en registros para ser utilizados en la etapa final.

2.2.3 Actualización del estado de las salidas

Finalmente el PLC toma los resultados almacenados después de la ejecución del programa. Los resultados se van reflejando, uno a uno, en cada una de las salidas, en el orden por defecto del equipo, o en el definido si se permite configurar el orden de actualización.

2.3 *Tiempo de Respuesta*

El tiempo de respuesta total de un PLC es lo que más se ha de considerar en la compra de un equipo. Así como nuestro cerebro, el PLC toma un cierto tiempo para reaccionar a los cambios. En muchas operaciones la velocidad no es importante, pero en otras es determinante en la respuesta adecuada de nuestro PLC.

Por ejemplo, cuando se esta realizando una tarea como puede ser conducir un vehículo en una carretera, nuestros ojos están recibiendo continuamente una imagen del entorno, esta imagen es la entrada para nuestro cerebro. Si de pronto nuestra pista es bloqueada por un vehículo que ingresa a la carretera, la imagen entregada por nuestros ojos mostrará nuestra vía bloqueada, el cerebro tomará esta imagen y al procesarla, nuestras lógicas internas indicaran que existe un peligro y que se debe accionar el freno. El cerebro determina que se debe accionar el pedal central que esta en piso y envía una señal para tal efecto. Estas señales neurológicas constituyen la salida del cerebro. Finalmente esta señal llegará a los actuadores, que en nuestro caso son los músculos, los que finalmente convierten la señal en una acción física en el mundo exterior. Sin duda que este ejemplo simplifica de manera grosera nuestro sistema nervioso, pero lo rescatable es que todo el accionar descrito toma un cierto tiempo. Veamos el caso en que el obstáculo se interpone de manera repentina. Si nuestro tiempo de proceso es muy largo, tal vez no lleguemos ni a tocar el pedal de freno, con los consiguientes problemas para nuestra integridad.

El tiempo de respuesta del PLC considera el tiempo necesario hasta la salida de la señal, dejando de lado la consideración del tiempo que toman los actuadores para realizar lo mandado por la señal. Esto es muy importante y no se debe olvidar a la hora de determinar el tiempo de respuesta requerido para nuestro PLC. Desde el punto de vista del PLC existe una división de las acciones que consumen tiempo:

2.3.1 Entrada

Esta acción necesita el tiempo que se invierte en revisar el estado de las entradas. El PLC puede realizar una acción a la vez, y el revisar una entrada constituye una, de esta forma para revisar el estado de todas las entradas se tiene una cantidad de tiempo considerable. El PLC toma el estado de la entrada y la guarda en un registro de memoria que se puede acceder desde el programa.

2.3.2 Ejecución

Es la acción que consume el tiempo que transcurre durante la ejecución de las rutinas programadas en el PLC. Aquí se procesa la información del estado de las entradas y se toma decisiones de acuerdo a la lógica programada. Depende del número de líneas de programa, lo que a su vez depende de la eficiencia de las lógicas ingresadas al equipo.

2.3.3 Salida

Durante esta acción se traspasan los estados de los registros de memoria reservados para la salida de nuestra rutina lógica a cada una de las salidas físicas de nuestro equipo. En este caso el tiempo transcurrido es mayor mientras más salidas se tenga que actualizar, que casi siempre corresponde a actualizar todas las salidas, aun las no modificadas.

De este modo el tiempo de respuesta total de nuestro PLC corresponde a la suma de los tiempos de entrada, de ejecución y salida.

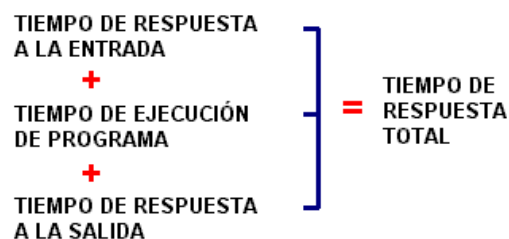


Figura 3. Tiempo de respuesta de un PLC.

2.4 Efecto del Tiempo de Respuesta

Ahora que hemos definido el tiempo de respuesta analizaremos porque es determinante en como se adecua el PLC a nuestros requerimientos. El PLC sólo puede ver el estado de las entradas durante el tiempo que transcurre en la etapa de entrada. Como es evidente, el tiempo de respuesta corresponde al tiempo invertido por el PLC en realizar un barrido. Veamos que ocurre con el PLC suponiendo que este debe sensor una sola entrada (Figura 4).

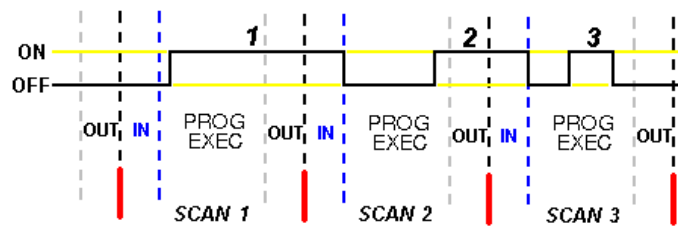


Figura 4. Tres tipos de entrada revisadas por el PLC.

En el esquema, el primer cambio de estado de la entrada no es apreciado por el PLC hasta el segundo barrido. Esto se debe a que el cambio ocurrió una vez finalizado el intervalo de tiempo durante el cual el PLC verifica el estado de las entradas. Para el segundo cambio de estado de la entrada ocurre lo mismo que en el primero. En el tercer cambio en el valor de la entrada ocurre una situación en la que el cambio no será advertido por el PLC debido a que este nunca fue advertido por un chequeo de entrada, ni en el barrido actual ni en los subsecuentes. Para evitar esta situación debemos imponer que el cambio en la entrada este disponible por lo menos durante un tiempo igual al tiempo de entrada más el tiempo que toma un barrido, como se aprecia en el esquema de la Figura 5. Si no se hace esto el PLC jamás verá el cambio de estado.



Figura 5. Menor tiempo posible para que la entrada sea revisada por el PLC.

Para realizar la corrección propuesta existen dos métodos que presentamos a continuación

2.4.1 Función de extensión de pulso

Una señal que cambia y vuelve a su estado original tan rápido origina una entrada que se denomina pulso. Ahora si este pulso se extiende en el tiempo según se mencionó antes, nos aseguramos que este cambio será registrado por nuestro PLC en el siguiente barrido.

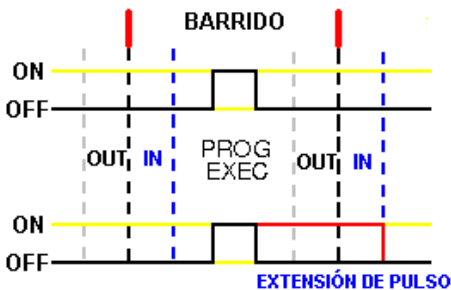


Figura 6. Esquema de la función de extensión de pulso.

2.4.2 Función de interrupción

Esta función interrumpe el proceso de barrido en el momento mismo del cambio en el valor de la entrada y llama a una rutina especial que se deberá preparar para cuando se reciba este tipo de señal. La rutina se escribe como un mini programa dentro de la rutina principal. Una vez ejecutadas las acciones programadas la ejecución del programa retorna a la rutina principal, en el estado en que quedo antes de salir a ejecutar la rutina de interrupción, continuando con el proceso de barrido normal.

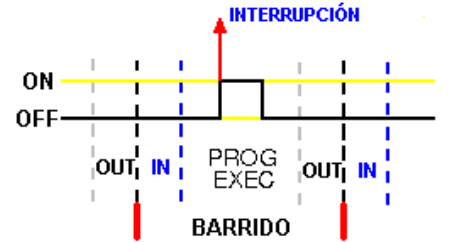


Figura 7. Esquema de la función interrupción.

Veamos ahora el mayor tiempo para que una salida se mantenga encendida. Asumiendo que cuando encendemos una salida necesitamos mover una carga conectada a la salida del PLC. El diagrama de abajo muestra el mayor retraso en el encendido de la salida una vez que la entrada a recibido señal. El peor caso considera que la entrada no es vista por el PLC hasta el segundo barrido). El tiempo de atraso máximo es de dos ciclos de barrido menos un tiempo de entrada.

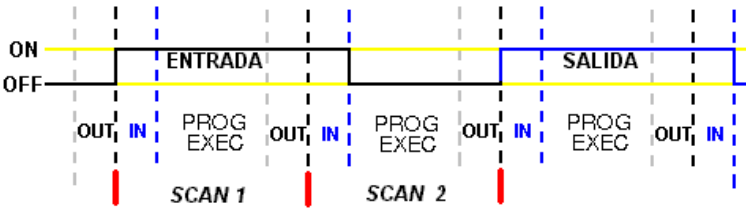


Figura 8. Esquema del tiempo de atraso máximo en el encendido de una carga como respuesta a la entrada.

3 CREACIÓN DE PROGRAMAS

3.1 Los Relés

El propósito de un PLC es reemplazar los circuitos lógicos representados por relés mecánicos, por esto es útil revisar como funcionan.

Un relé es básicamente un switch electromagnético. Aplicando un voltaje determinado a la bobina se genera un campo magnético que atrae al elemento de contacto a su posición de cerrado. Esto permite cerrar un circuito, el que tiene una potencia determinada por la capacidad de los contactores del relé, que puede ser mucho más alta que las potencias utilizadas para accionar el circuito de la bobina. Por lo general, los voltajes y corrientes utilizados para accionar el circuito de la bobina del relé se denominan señales, y permiten manejar las lógicas de control en rangos de voltajes que no involucran peligro, y son los adecuados para ser generados por circuitos electrónicos de control (ver Figura 9).

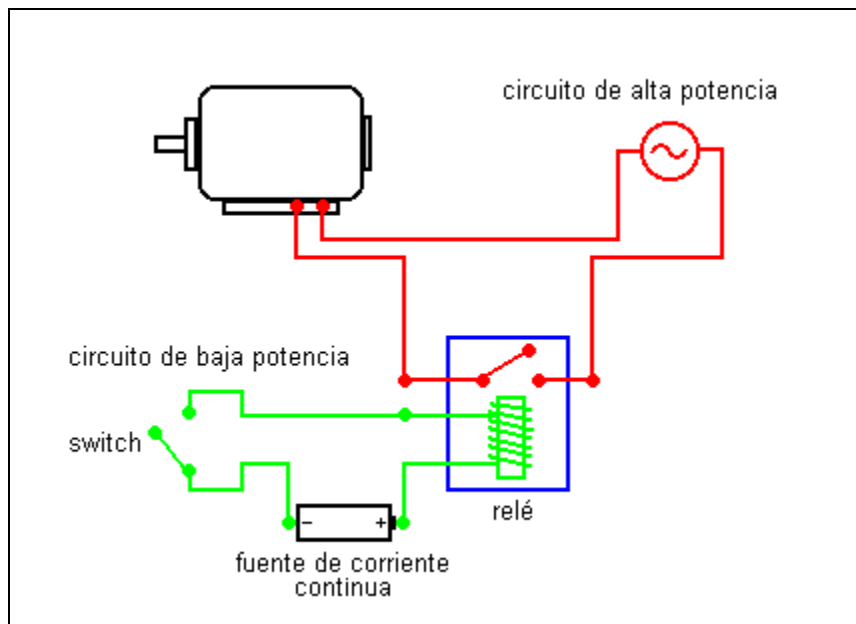


Figura 9. Esquema de aplicación básica de un relé.

En resumen, a través de un relé se pueden separar los circuitos de alta potencia, que entregan la energía casi sin restricciones, de los circuitos de baja potencia, generalmente orientados al control.

3.2 Reemplazando Relés

Ahora utilizaremos un PLC en lugar del relé de la Figura 9, lo que no es algo económicamente viable, y que sólo es útil para la explicación siguiente. Lo primero que se debe realizar es la creación de un *Diagrama de Escalera*. Esto se debe hacer debido a que el PLC no

entiende un diagrama esquemático. Este solo reconoce código. Afortunadamente la mayoría de los PLC tienen un software que es capaz de convertir el diagrama de escaleras a código. Esto nos evita la tediosa tarea de aprender el código de los PLC.

3.2.1 Primer Paso

Lo primero es trasladar todas las partes del esquema usando símbolos que sean entendibles al PLC. Un PLC no entiende términos como interruptor, relé, motor. Algo más cómodos al PLC resultan los términos entrada, salida, bobina, contacto. No hay cuidados con cual es tipo de dispositivo que es la entrada o la salida. Sólo hay que ser cuidadoso en definir cuales son las entradas y cuales las salidas.

Primero reemplazaremos la batería con un símbolo. Este símbolo es común a todos los diagramas de escalera. El símbolo se denomina barras de alambre y se visualiza como dos líneas verticales. Las líneas se ubican a ambos lados del diagrama. La barra de la izquierda se asume como el voltaje positivo (+) y la derecha como la referencia o tierra. La corriente se tomará como fluyendo de izquierda a derecha, al igual que el flujo lógico.

Ahora se define el símbolo de una entrada. En este ejemplo básico se tiene una entrada desde el mundo real (es decir, el switch), la que será reemplazada por el símbolo mostrado en la Figura 10. Este símbolo también es utilizado como contacto de un relé.

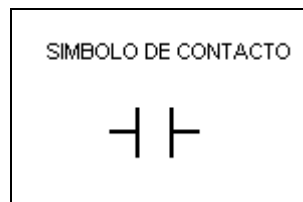


Figura 10

Ahora presentaremos el símbolo de salida. En este ejemplo se tiene una salida (el motor). Se dará el símbolo de salida al elemento al cual el motor está físicamente conectado y será como se muestra en la Figura 11. Este símbolo es usado como la bobina del relé.

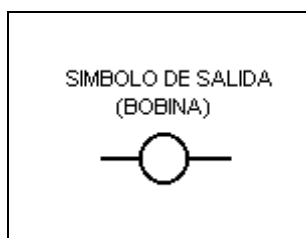


Figura 11

La fuente de corriente alterna no es parte del diagrama de escalera. El PLC sólo tiene injerencia en si sus salidas están o no conectadas y no en lo que está conectado físicamente al equipo.

3.2.2 Segundo Paso

Debemos decir al PLC donde está la localización de cada uno de los elementos del diagrama. En otras palabras debemos entregar todas las direcciones de dispositivos. Las direcciones son asignadas previamente según procedimientos establecidos por cada fabricante. Más adelante se entrará en mayor detalle, por ahora se supondrá que la entrada se encuentra en la dirección "0000" y la salida en la dirección "500".

3.2.3 El Paso Final

Ahora resta convertir el esquema en una secuencia lógica de eventos. Esto es mucho más fácil de lo que suena. El programa debe decir que se debe hacer cuando se lleve a efecto algún evento, como por ejemplo el cambio de estado de la entrada. En el ejemplo presentado hay que indicar al PLC que debe hacer cuando el operador encienda el interruptor. Obviamente queremos que el motor se encienda, pero el PLC debe recibir la instrucción para que accione el relé de salida. Sin estas instrucciones el PLC no es un dispositivo de utilidad.

Como resultado se tendrá un esquema como el de la Figura 12. Hay que notar que se han eliminado todos los relés del "mundo real".

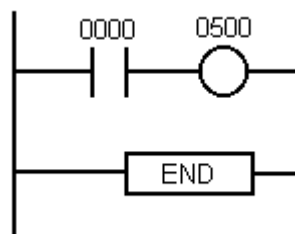


Figura 12

3.3 Instrucciones Básicas

Ahora examinaremos algunas de las instrucciones básicas en detalle para saber cual es la función de cada una de ellas.

3.3.1 Load

La instrucción Load (LD) es un contacto normalmente abierto. Esta es llamada en algunas ocasiones "examine if on" (XIO). El símbolo de la instrucción Load se muestra abajo.

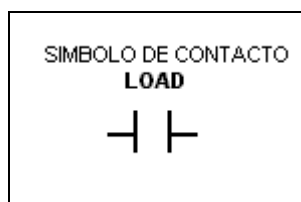


Figura 13

Este símbolo se utiliza cuando se necesita la existencia de una señal de entrada para que el símbolo se active. Cuando una entrada física se enciende podemos decir que esta instrucción es “verdadera”. Entonces se examina una entrada esperando una señal. Si la entrada está físicamente encendida el símbolo también lo estará. La condición de encendido se puede referir como estado lógico 1. Este símbolo es usado frecuentemente en entradas internas, entradas externas y salidas externas. Hay que recordar que los relés internos no existen físicamente, sino que corresponden a relés simulados por software.

3.3.2 LoadBar

La instrucción LoadBar es un contacto normalmente cerrado. También se le llama algunas veces LoadNot o “examine if closed” (XIC). El símbolo de la instrucción LoadBar se muestra abajo.

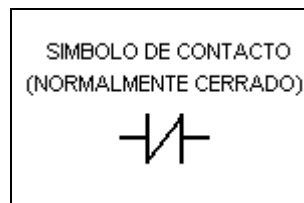


Figura 14

Este símbolo se utiliza cuando una señal de entrada no se necesita que exista una señal de entrada para que el símbolo se active. Cuando la entrada física se apaga podemos decir que esta instrucción es Verdadera. En resumen, desde el punto de vista lógico, la instrucción Load Bar corresponde a la negación lógica de la instrucción Load

Tabla 1. Tabla de verdad de la instrucción contacto.

ESTADO LOGICO	CONTACTO	CONTACTO(NC)
0	FALSO	VERDADERO
1	VERDADERO	FALSO

3.3.3 Out

La instrucción Out también se llama OutputEnergize. La instrucción Out almacena el estado de la bobina del relé al que esta conectado la salida. El símbolo de esta instrucción se muestra abajo.

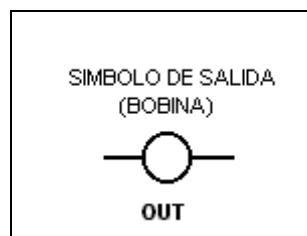


Figura 15

Cuando este símbolo está en la trayectoria de un peldaño en que todas sus instrucciones previas son verdaderas, su estado también será verdadero. Que la instrucción sea verdadera quiere decir que la salida asociada está físicamente encendida. Al igual que las instrucciones anteriores, esta se puede usar en relés internos o salidas físicas.

3.3.4 OutBar

Esta instrucción es el opuesto de la instrucción Out. La instrucción OutBar es el equivalente a un relé cuyo estado por defecto es cerrado. El símbolo de esta instrucción se muestra abajo. Hay que notar que algunas marcas de PLC no cuentan con esta instrucción.

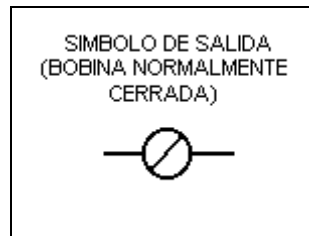


Figura 16

Cuando este símbolo se encuentra en una trayectoria de un peldaño cuyas instrucciones previas están en estado falso el valor que tomara la instrucción será verdadero.

3.4 Ejemplo de Aplicación

Ahora compararemos un diagrama de escalera muy simple con un circuito conectado por relés, para ver las diferencias.

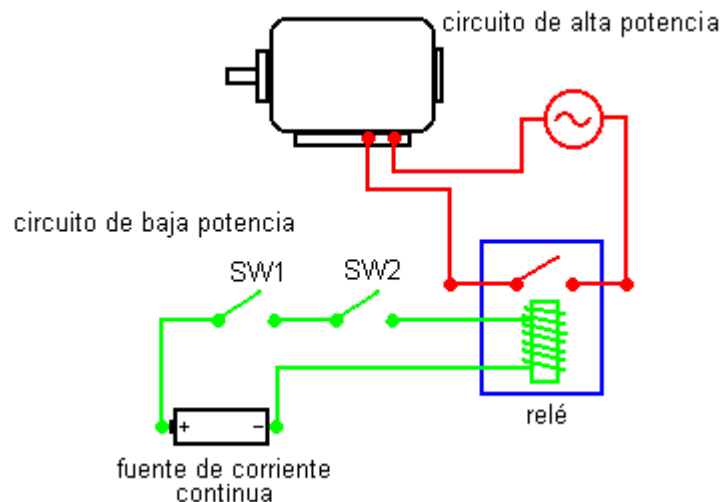


Figura 17. Esquema de un circuito real conectado por relés. Un par de relés (que serán reemplazados por el PLC) pueden encender un relé que conecta a un circuito de potencia.

En el circuito de arriba, cuando la bobina sea energizada tendremos un circuito cerrado entre los terminales positivo y negativo de la batería. Podemos simular este mismo circuito con un diagrama de escalera. Un diagrama de escalera consiste de una serie de peldaños individuales, tal como una escalera real. Cada peldaño puede contener uno o más entradas y o más salidas. La primera instrucción de un peldaño debe ser siempre una entrada y la última instrucción debe ser siempre una salida (o su equivalente).

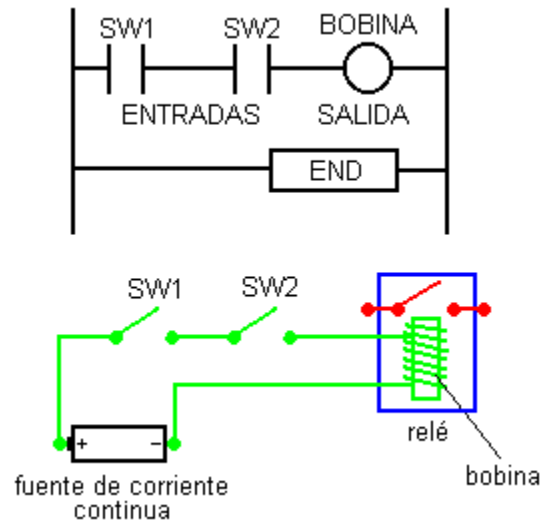


Figura 18. Relación entre el diagrama de escalera y el sistema real.

Debe destacarse que en este ejemplo un peldaño ha recreado todo el circuito externo del comienzo con un diagrama de escalera. Aquí se han usado las instrucciones Load y Out. Algunos fabricantes de PLC pueden requerir la instrucción END al final del peldaño que precede al peldaño END.

Ahora seguiremos la pista a los registros, para ver que lógica realiza el PLC.

3.4.1 Registros del ejemplo

Ahora tomaremos el ejemplo anterior y cambiaremos el interruptor 2 (SW2) por el símbolo normalmente cerrado (LoadBar). El interruptor 1 (SW1) estará físicamente apagado y el interruptor 2 estará físicamente encendido al inicio de las secuencias programadas.

El diagrama de escalera se verá como sigue:

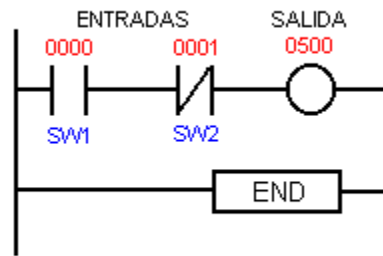


Figura 19. Diagrama de escalera en el estado del ejemplo.

Note además, que ahora hemos dado a cada símbolo (o instrucción) una dirección. Esta dirección es seteada en alguna parte de las áreas de almacenamiento de los archivos de datos del PLC, y almacena el valor del estado de la instrucción (es decir verdadero o falso). La mayoría de los PLC usan ubicaciones de memoria de 16 casillas o bit. En el ejemplo de arriba se utilizan dos ubicaciones o registros diferentes.

Tabla 2. Estados de los registros del ejemplo.

REGISTRO 00															
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
														1	0
REGISTRO 05															
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
															0

En la tablas de arriba podemos ver que en el registro 00, bit 00 (es decir en la entrada o input 0000) hay un 0 lógico y en el bit 01 (es decir en la entrada o input 0001) hay un 1 lógico. El registro 05 muestra que en bit 00 (es decir en la entrada 0500) un 0 lógico. El 0 lógico ó 1 indican que una instrucción es falsa o verdadera, respectivamente. A pesar de que la mayoría de los ítem en los registros mostrados están en blanco, cada uno de ellos contiene un 0. Estos no se muestran para concentrarnos en los bit que realmente están involucrados en nuestro ejemplo.

Tabla 3. Resumen de los estados de los símbolos.

CONDICION LOGICA DEL SIMBOLO			
BITS LOGICOS	LD	LDB	OUT
BIT 0	FALSO	VERDADERO	FALSO
BIT 1	VERDADERO	FALSO	VERDADERO

El PLC sólo acciona una salida cuando todas las condiciones del peldaño son Verdaderas.

Entonces, revisando la tabla de arriba, podemos ver en el ejemplo anterior que si SW1 tiene un 1 lógico, SW2 debe tener un 0. Entonces, y en este único caso, la bobina del relé estará en estado verdadero (es decir, activada). Si cualquiera de las instrucciones antes de la salida del peldaño (bobina) esta en una condición Falsa, la bobina del relé estará en condición Falsa, es decir, no será energizada.

Revisemos ahora la tabla de verdad del programa del ejemplo anterior, es decir, del diagrama de escalera. La tabla de verdad muestra todas las posibles combinaciones de los estados de las dos entradas.

Tabla 4. Resumen de todas las combinaciones de estado del sistema.

Entradas		Salidas	Registro de los Bits Lógicos		
SW1(LD)	SW2(LDB)	BOBINA(SALIDA)	SW1(LD)	SW2(LDB)	BOBINA(SALIDA)
FALSO	VERDADERO	FALSO	0	0	0
FALSO	FALSO	FALSO	0	1	0
VERDADERO	VERDADERO	VERDADERO	1	0	1
VERDADERO	FALSO	FALSO	1	1	0

Se debe notar de esta última tabla que los cambios de los estados de SW1 y SW2 en el tiempo influyen en la salida. Pero, la salida será verdadera sólo cuando todas las instrucciones que la preceden en el peldaño sean verdaderas.

4 SET DE INSTRUCCIONES PRINCIPALES

4.1 Instrucciones Cerrojo

Ahora revisaremos una variación de las salidas normales estudiadas anteriormente. Las bobinas de las salidas normales son una parte esencial de nuestros programas pero debemos recordar que ellas serán verdaderas solo cuando todas las instrucciones que le preceden en el peldaño son verdaderas. Si lo anterior no ocurre, el estado de la bobina será falso.

Recordando el ejemplo de la Figura 17 imagine que sucedería si no se utilizara un interruptor de tipo conectado/desconectado. Para mantener en movimiento el motor tendríamos que mantener presionado el interruptor (interruptor momentáneo). Las instrucciones cerrojo nos permiten utilizar el tipo de interruptores momentáneos para aplicaciones como las del ejemplo mencionado, de modo de programar el PLC para que encienda una salida una vez que ha sido accionado un interruptor del tipo momentáneo, y la apague cuando el botón es presionado otra vez.

La instrucción cerrojo es denominada como SET u OTL (output latch). La instrucción para retirar la función cerrojo se denomina RES, OUT (output unlatch) o RST (reset). El diagrama de abajo muestra como utilizar esta función en un programa.

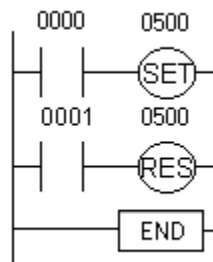


Figura 20. Diagrama de la instrucción cerrojo.

Aquí se están utilizando dos interruptores momentáneos. Uno está conectado físicamente a la entrada 0000, mientras que el otro está conectado a la entrada 0001. Cuando el operador presiona el botón 0000 la instrucción "SET 0005" pasa a ser verdadera y la salida 0500 es encendida físicamente. Incluso después de que el operador deja de presionar el botón, la salida 0500 se mantiene encendida. La única forma de apagar la salida 0500 es encender la entrada 0001. Esto origina que la instrucción "RES 0500" sea verdadera y en consecuencia la salida 0500 tome su valor original o por defecto.

Ahora cabe revisar que sucede si se presionan los dos botones al mismo tiempo. La respuesta a esta situación se encuentra recordando las secuencias de barrido con que el PLC trabaja. La escalera es revisada desde arriba hacia abajo y de izquierda a derecha, La primera cosa que el PLC revisa son sus entradas. Los registros 0000 y 0001 están en posición encendido. Lo próximo que realiza el PLC es ejecutar el programa. Partiendo desde arriba, a la izquierda, la entrada 0000 es verdadera, en consecuencia se debe encender 0500, lo que sigue es ejecutar el peldaño que viene, y debido a que la entrada 0001 estuvo encendida durante la revisión, se debe

volver a su estado por defecto a la salida 0500, es decir, se apaga. Finalmente, en la parte que el PLC actualiza las salidas, el registro 0500 se mantiene apagado.

4.2 Contadores

Un contador es un dispositivo que pretende hacer una cosa muy simple, contar. Su uso, sin embargo, puede ser algunas veces un desafío debido a que cada fabricante hace uso de ellos de una forma diferente. Se pretende que la información que se muestra sirva para ser aplicada a cualquier tipo de contador.

Existen varias clases de contadores, en particular los contadores hacia arriba (es decir, sólo cuentan 1, 2, 3...) que son llamados CTU (count up), CNT, C, o CTR. También están los contadores hacia abajo (es decir, sólo cuentan 9, 8, 7...) que son llamados CTD (count down) cuando son denominados como una instrucción separada. Finalmente está la combinación de los contadores presentados anteriormente, contadores hacia arriba y abajo (es decir, pueden realizar cosas como 1, 2, 3, 4, 3, 2, 3, 4, 5, ...). Estos se denominan típicamente UDC (up-down counters) cuando vienen como una instrucción separada.

Varios fabricantes de PLC incorporan en sus equipos uno o dos tipos de contadores, pero ellos pueden ser utilizados para realizar cualquiera de los tipos de conteo. No importa demasiado como el fabricante denomine a un contador particular, los principios de operación son similares.

Para agregar más complejidad al tema, algunos fabricantes incorporan contadores de alta velocidad, denominados comúnmente HSC (high-speed counter) o de forma similar. Típicamente un contador de alta velocidad es un componente de hardware, es decir, es un chip adicional en el PLC, no así su contraparte normal, que corresponde a instrucciones que son realizadas por el mismo PLC, es decir, son contadores manejados por software. Los contadores de alta velocidad, al no estar soportados por la CPU, tienen la ventaja de no estar relacionados con el tiempo de barrido de la CPU.

Una buena regla para recordar es la de utilizar siempre los contadores por software a menos que el tiempo que transcurre entre dos pulsos sea inferior a dos veces el tiempo que tarda en hacer un barrido la CPU, es decir, si el tiempo de barrido es de 2 milisegundos y los pulsos llegan a intervalos de 4 milisegundos o superiores, entonces utilice un contador por software. Si los pulsos llegan a intervalos de 3 milisegundos o inferiores, utilice un contador de alta velocidad.

Para utilizar un contador de alta velocidad se deben conocer 3 cosas:

1. Desde donde provienen los pulsos que se desean contar. Típicamente el pulso proviene de una de las entradas. (un sensor conectado a la entrada 0000 por ejemplo)
2. Cuantos pulsos son necesarios contar antes de reaccionar. Por ejemplo, contar 5 elementos antes de empaquetarlos en un envase.
3. Cuando y/o como se restaurará el contador para iniciar un nuevo conteo. por ejemplo, reiniciar el conteo una vez que se han contado los 5 elementos que se van envasar.

Cuando el programa está corriendo en el PLC, este es capaz de mostrarnos el valor actual o acumulado que tiene el contador.

Usualmente los contadores operan en rangos de 0 a 9999, -32768 a +32767, o de 0 a 65535. Estos números tan especiales provienen del hecho que la mayoría de los contadores de PLC son de 16 bits.

En la Figura 21 se muestra un símbolo de un contador que puede variar dependiendo del fabricante. Hay que recordar que aunque parezcan diferentes, estos se utilizan de la misma forma.

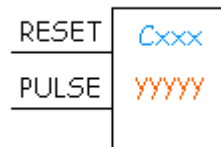


Figura 21. Símbolo de un contador.

En este contador se requiere de dos entradas. Una será la entrada de reinicio. Cuando esta entrada se encienda el valor actual o acumulado volverá a cero. La segunda entrada corresponde a la dirección del registro desde donde provienen los pulsos que serán contados.

Por ejemplo, si se quiere contar cuantas cosas pasan frente a un sensor que esta físicamente conectado a la entrada 0001, entonces pondremos un contacto normalmente abierto con la dirección 0001 frente de la línea de pulso.

Cxxx es el nombre del contador. yyyy es el número de pulsos que se requiere contar antes de realizar una acción. Cuando el contador ha llegado a su definición de yyyy pulsos, este encenderá un grupo de contactos separados que también están etiquetados por Cxxx.

Hay que recordar que el valor acumulado del contador cambiará solo después de una transición de apagado a encendido del pulso de entrada.

Aquí se tiene el símbolo de un contador en un diagrama de escalera mostrando como definir un contador (llamado contador 000) para contar 100 cosas desde la entrada 0001 antes de encender la salida 500. El sensor 0002 reinicia el contador.

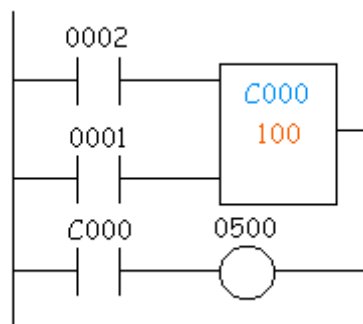


Figura 22. Diagrama de escalera utilizando un contador.

Abajo se presenta el símbolo que puede representar un contador arriba-abajo. Se ha utilizado la misma abreviación presentada en el ejemplo anterior.

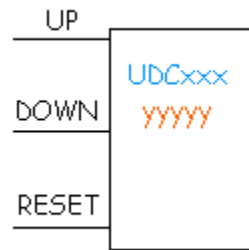


Figura 23. Símbolo para un contador UDC.

En este contador arriba-abajo es necesario asignar 3 entradas. La entrada de reinicio tiene la misma función mencionada anteriormente. Una de las entradas restantes es para realizar el conteo hacia arriba y la otra para realizar el conteo hacia abajo. En este ejemplo el contador se denominará por UDC000 y se le dará el valor de configuración 1000, es decir la intención es contar 1000 pulsos. Cuando el sensor conectado a la entrada 0001 detecte un objetivo se aumentará el valor acumulado, cuando suceda lo mismo al sensor conectado a la entrada 0003 se disminuirá el valor acumulado. Cuando el valor acumulado alcance 1000 se encenderá la salida 500. El diagrama de escalera de este ejemplo se muestra abajo.

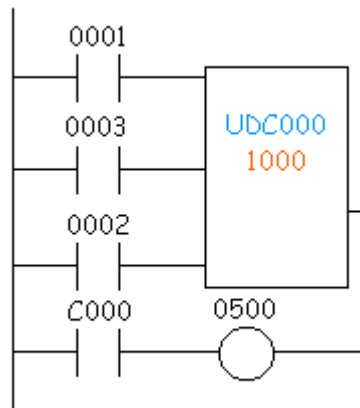


Figura 24. Diagrama de escalera utilizando un contador UDC.

Algo importante de destacar es que los contadores y temporizadores no pueden tener el mismo nombre (en la mayoría de los PLC). Esto se debe a que generalmente usan los mismos registros.

4.3 Temporizadores

En términos muy simples un temporizador es una instrucción que espera una cantidad de tiempo determinado antes de realizar una acción específica. Una gran dificultad aparece al considerar que no existe una estandarización para su implementación por parte de los fabricantes de PLC. Sin embargo, la mayoría se puede clasificar en los siguientes grupos. Como una forma de reconocerlos en los catálogos se conserva sus nombres en inglés.

4.3.1 On-Delay timer

Este tipo de temporizador simplemente retrasa el encendido de una salida. En otras palabras, después de que en un sensor se ha encendido debido a la detección de su objetivo, espera una cantidad determinada de segundos antes de activar un solenoide. Este es el más común de los temporizadores. Se denomina generalmente como TON (timer on-delay), TIM (timer) o TMR (timer).

4.3.2 Off-Delay timer

Este tipo de temporizador es el caso opuesto al anterior. Este temporizador retrasa el apagar una salida. Después de que un sensor ha detectado su objetivo este enciende un solenoide. Cuando el objetivo ya no está frente al sensor, esta instrucción demora el apagar el solenoide al que está conectado el sensor. Este temporizador se denomina como TOF (time off-delay) y es menos común que el temporizador on-delay.

4.3.3 Retentive or Accumulating timer

Este tipo de temporizador necesita de dos entradas. Una entrada inicia la medición con el cronómetro y la otra entrada lo vuelve a su valor inicial. Los temporizadores on-off delay vistos en los párrafos anteriores deben ser reiniciados si el sensor de entrada no alcanza un estado de encendido o apagado en un ciclo completo de duración del temporizador. Este temporizador, sin embargo, retiene el tiempo actualmente transcurrido cuando el sensor se apaga a medio tiempo. Por ejemplo, si se desea conocer por cuánto tiempo un sensor ha estado encendido durante el transcurso de una hora. Si se usa uno de los temporizadores de los párrafos anteriores, se debe reiniciar el contador cuando el sensor es encendido o apagado. Este temporizador, sin embargo, nos permite mantener el tiempo total o acumulado. Este temporizador se denomina usualmente RTO (retentive timer) o TMRA (accumulating timer). Para utilizar este temporizador es necesario conocer dos cosas:

1. Que evento encenderá al temporizador: Típicamente esto se realiza con una de las entradas. (Un sensor conectado a la entrada 0000 por ejemplo).
2. Cuanto tiempo se ha de esperar antes de reaccionar. Supongamos que esperamos 5 segundos antes de encender un solenoide.

Cuando las instrucciones antes del símbolo del temporizador son verdaderas, este comenzará el conteo. Cuando el valor configurado transcurra el temporizador cerrará sus contactos. Cuando el programa está corriendo en el PLC, es habitual que el temporizador despliegue el tiempo acumulado para que podamos observarlo. Típicamente los temporizadores pueden realizar un conteo de 0 a 9999 o de 0 a 65535 unidades, donde a cada unidad se le llama tic.

El porqué de estos números se debe a que la mayoría de los PLC tienen contadores de 16 bits. Por ahora es suficiente mencionar que 0 a 9999 es un número de 16 bits del tipo binario codificado, y 0 a 65535 es un número de 16 bits de tipo binario. Cada tic del reloj es equivalente a x segundos.

Usualmente cada fabricante de PLC ofrece diferentes tics. La mayoría de los fabricantes pone a disposición incrementos de 10 a 100 ms (tics del reloj). Varios fabricantes también ofrecen incrementos de 1ms hasta 1 seg. Estos temporizadores de diferentes incrementos trabajan de la misma forma que los estudiados anteriormente, pero a veces tienen diferentes nombres para destacar la diferencias de sus incrementos. Algunos son contadores de alta velocidad (TMH), temporizadores de super alta velocidad (TMS) o temporizadores acumulativos rápidos (TMRAF).

Abajo se muestra un símbolo típico de una instrucción de un temporizador y como se utiliza. Hay que recordar que aunque parezcan diferentes, son utilizados de una forma similar. Si se sabe utilizar uno, se saben utilizar todos.

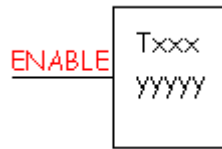


Figura 25. Símbolo de un temporizador.

Este temporizador es del tipo on-delay y se llamará Txxx. Cuando la entrada disponible es encendida y el temporizador comienza el conteo. Cuando este cuenta yyyy (un valor predefinido) veces, encenderá sus contactos, los que serán utilizados en pasos subsecuentes del programa. Recuerde que la duración de una cuenta (tic), varía de un fabricante a otro y de la base de tiempo utilizada. Por ejemplo un tic puede ser equivalente a 1 ms o a un segundo.

En la Figura 26 se presenta el símbolo de un temporizador en el contexto de un diagrama de escalera. En este diagrama se debe esperar el encendido de la entrada 0001. Cuando esto ocurre, el temporizador T000 (definido en tics de 100 ms) comienza su conteo hasta que el temporizador llegue a 10000 ms (es decir, 10 segundos) lo que se maneja definiendo el número de tic en 100. Cuando se alcanzan los 10 segundos se cierran los contactos del temporizador y el registro 500 cambia a encendido. Cuando la entrada 0001 se apaga el temporizador T000 es vuelto a cero, causando que sus contactos se abran, haciendo que la salida 500 también sea falsa.

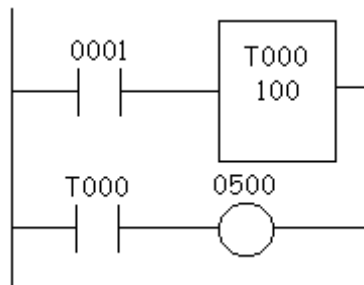


Figura 26. Temporizador en le contexto de un diagrama de escalera.

Un temporizador acumulativo luce de manera similar al siguiente esquema (Figura 26).

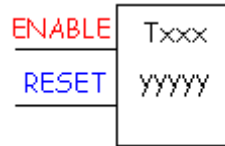


Figura 27. Temporizador acumulativo.

Este temporizador es llamado Txxx. Cuando la entrada disponible se enciende el temporizador comienza el conteo. Cuando el conteo alcanza el valor yyyy (un valor predefinido), este enciende los contactos que inician las acciones que están programadas cuando esto ocurre. Sin embargo, cuando la entrada disponible vuelve a su estado apagado antes que el valor predefinido se alcance, el valor actual será retenido. Cuando la entrada vuelva a un estado encendido, el temporizador continuará su cuenta desde donde quedó antes de entrar al estado apagado. La única forma para forzar al temporizador a su estado inicial y así comenzar un nuevo conteo, es encender la entrada para inicializarlo.

Este temporizador, en el contexto de un diagrama de escalera, luce como en la Figura 28. En este diagrama se esperaran los cambios en la entrada 0002 para encenderlo. Cuando esto ocurra (el encendido) el temporizador T000 comenzará el conteo. Este contará 100 veces (con incrementos de 10 ms), hasta completar un 1000 ms = 1s. Cuando transcurra 1 segundo, los contactos de T000 se cerraran encendiendo 500. Si la entrada 0002 se apaga, el tiempo transcurrido es retenido. Cuando la entrada 0002 vuelve a encenderse, el temporizador continuará su cuenta desde el punto donde se quedo antes de que se apagara la entrada. Cuando la entrada 0001 se enciende, el temporizador es vuelto a cero, haciendo que la salida 500 se apague.

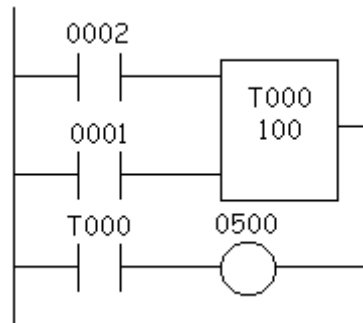


Figura 28. Temporizador acumulativo en el contexto de un diagrama de escalera.

Un hecho importante que destacar es que los contadores y los temporizadores no pueden tener el mismo nombre, esto se debe a que estos utilizan los mismos registros.

4.4 Exactitud del Tiempo

Ahora que sabemos como crear y manejar un temporizador, aprenderemos sobre las implicancias de su precisión. Cuando definimos un temporizador para unos pocos segundos, o más, no tomamos atención a las características de su precisión debido a que en estos casos es prácticamente insignificante. Sin embargo, cuando se crean temporizadores que tienen una duración del orden de milisegundos ($1 \text{ ms} = 1/1000 \text{ segundo}$) se debe tomar en cuenta la precisión.

Hay en general dos tipos de error cuando se usan temporizadores. El primero es llamado error de entrada y el otro es llamado error de salida. El error total es la suma de los errores de entrada y salida.

4.4.1 Error de entrada

Este error ocurre dependiendo de cuando el temporizador se enciende durante un ciclo de barrido. Cuando el temporizador se enciende inmediatamente después de que el PLC revisa el estado de sus entradas durante un ciclo de barrido, el error de entrada tomara su mayor valor posible, es decir, más de un ciclo de barrido completo. Esto es consecuencia de que el PLC revisa el estado de sus entradas durante la primera parte de su ciclo de barrido. Si el encendido ocurre a la siguiente etapa de revisión, en el barrido posterior, se está incurriendo en un error obvio. Aparte de lo expuesto, el ciclo de barrido incluye el tiempo utilizado en la ejecución de las instrucciones incluidas en ese barrido. Si la instrucción que maneja al temporizador es el último peldaño de la escalera, se estará incurriendo en un gran error.

4.4.2 Error de Salida

Otro error ocurre basado en condiciones similares al del error anterior, pero en el instante en el cual se deben actualizar las salidas. Esto sucede porque el temporizador alcanza el valor determinado, pero el PLC debe terminar de ejecutar la secuencia de instrucciones del ciclo de barrido antes de actualizar el estado de las salidas, según el valor que tomo el temporizador un instante antes. En el peor de los casos el error de salida puede llegar a ser del tiempo de un ciclo de barrido completo.

En la Figura 29 se muestra un diagrama que ilustra el peor caso de un error de entrada. Se puede ver aquí que el peor error podría llegar a ser de un ciclo de barrido completo más el tiempo de ejecución del programa de instrucciones. Hay que recordar que el tiempo de ejecución de un programa varia de un programa a otro, debido a que depende del tipo y cantidad de instrucciones que contenga.

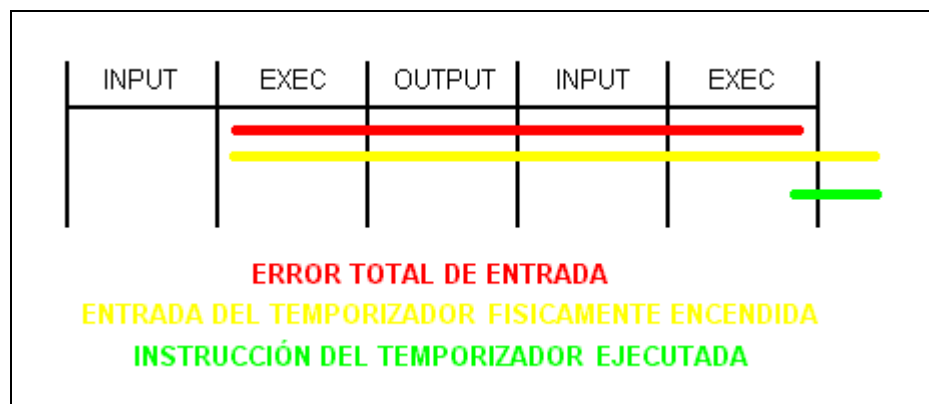


Figura 29. Peor caso de un error de entrada.

En la Figura 30 se ilustra el peor caso de un error de salida. Aquí se aprecia que el peor posible error puede ser un ciclo de barrido completo.

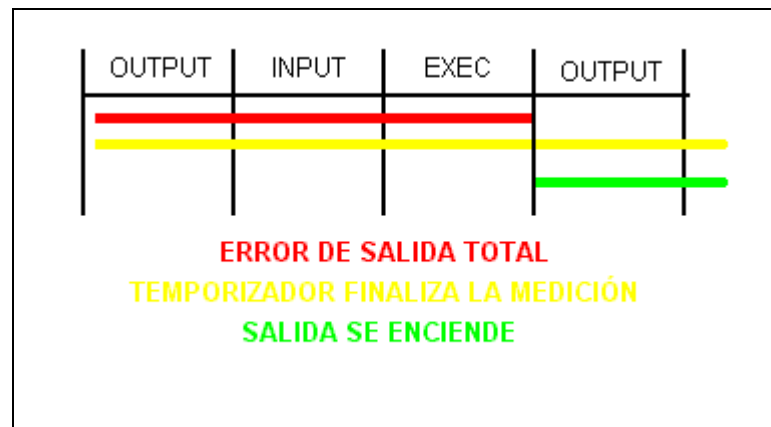


Figura 30. Peor caso de un error de salida.

Basados en lo presentado anteriormente se puede apreciar que el peor error total en la medición de tiempos puede llegar a ser igual a:

1 tiempo de barrido + 1 tiempo de ejecución de programa + 1 tiempo de barrido = 2 ciclos de barrido + 1 tiempo de ejecución de programa.

En la práctica esto significa que aunque la mayoría de los fabricantes incluyen temporizadores con incrementos de 1ms, estos no deben ser utilizados para duraciones menores a unos pocos milisegundos. Si el tiempo de barrido de su PLC es de 5ms sería recomendable no usar el temporizador con la intención de medir duraciones de menos de 15ms. Esta afirmación es válida siempre que el error que se pueda acumular es de importancia para nuestra aplicación. En la mayoría de las aplicaciones errores de este tipo son despreciables pero en algunas aplicaciones de alta velocidad, o muy precisas en el tiempo, este error puede ser muy significativo.

Hay que destacar que los errores mencionados corresponden a errores por software. También existe un error para temporizadores por hardware del tipo error de entrada y error de salida.

El error de entrada en temporizadores por hardware es causado por el tiempo que toma el PLC en encender la entrada del temporizador cuando son revisadas las entradas del PLC. Esta duración es típicamente de 10 ms. Esto se debe a que muchos PLC requieren de que la entrada deba estar físicamente encendida durante algunos barridos antes de ser actualizada como encendida (con el fin de eliminar ruidos).

El error de salida en temporizadores por hardware es causado por el tiempo que transcurre entre el instante cuando el PLC indica que una salida debe estar encendida, hasta el momento en que esta realmente lo está. Típicamente un transistor toma unos 0.5 ms en hacer esto, mientras que un relé mecánico toma cerca de 10 ms.

Si estos tiempos son considerables para la aplicación a diseñar, considere la utilización de temporizadores por hardware externos al PLC.

4.5 Cambio de Registros

En muchas aplicaciones es necesario guardar el estado de un evento que ha ocurrido con anterioridad. Como se ha visto con anterioridad esto es sencillo de realizar. Pero que se puede hacer si se debe guardar una cantidad considerable de eventos para ser utilizados en decisiones posteriores?. En este contexto se utilizan las instrucciones de cambio de registro.

Podemos utilizar registros o grupos de registros para formar una lista enlazada de bits para guardar estados previos de encendido o apagado de una entrada o salida de nuestro PLC. Cada cambio en el estado de la entrada o salida registrada originará un nuevo elemento que se enlaza a nuestra lista. Si esta situación supera el largo definido para nuestra lista, la adición de un nuevo suceso se añadirá en el primer bit o elemento, y el contenido del primer bit se cambiara al elemento siguiente de la lista , y así sucesivamente, hasta que el último elemento de nuestra lista será reemplazado por el valor del bit penúltimo, lo que origina que el valor del último elemento se pierda.

El cambio de registro recibe varios nombres. SFT (Shift), BSL (Bit Shift Left), SFR (Shift Foward Register) son los nombres más comunes utilizados por los fabricantes de PLC. Estos comandos mueven los registros hacia la izquierda, como se describe en el párrafo anterior. BSR (Bit Shift Righth) y SFRN (Shift Foward Register Not) son algunos ejemplos de comandos que realizan el cambio de registros hacia la derecha. Hay que destacar que la mayoría de los fabricantes de PLC agregan el comando para el cambio de registros hacia la izquierda, sin embargo, no todos cuentan con el comando para cambiar los registros hacia la derecha.

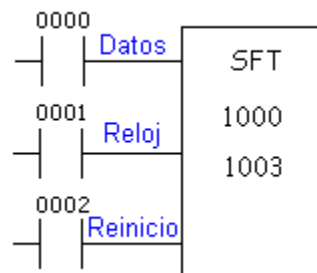


Figura 31. Símbolo del comando cambiar registro.

Un símbolo usualmente utilizado para denotar este comando se muestra en la Figura 31. Hay que notar que este comando necesita de 3 entradas y tiene necesariamente algún dato dentro del símbolo.

Las entradas utilizadas consisten en lo siguiente:

- Datos: La entrada de datos recopila los estados verdadero-falso que serán cambiados en la lista enlazada. Cuando la entrada de datos es verdadera el primer bit de la lista o registro tomará el valor 1. Este dato es entrado a la lista o registro únicamente en el borde de subida de los pulsos del reloj del PLC.
- Reloj: El reloj de entrada indica al comando cambiar registro que realice su operación sobre el registro. En el borde de subida de esta entrada, el comando cambia los datos o bits en una

posición dentro del registro e ingresa el estado de la entrada en el primer lugar del registro. En cada borde de subida de la señal del reloj el proceso se repite.

- Reinicio: Esta entrada hace exactamente lo que su nombre indica. Borra el contenido completo del registro, llevando a todos los bits del registro al estado 0.

El valor 1000 dentro del símbolo *cambiar registro* es la localización del primer bit del registro sobre el cual se quiere realizar el comando. Si pensamos en los registros como una lista de bits enlazados como una cadena, este elemento es el primer eslabón. El valor 1003 dentro del símbolo mostrado en la fig es el último bit del registro sobre el cual estamos operando. Este corresponde al último eslabón de la cadena. En este ejemplo hay bits intermedios, en consecuencia nuestro registro consta de cuatro bits (1000, 1001, 1002, 1003).

5 CONEXIÓN DEL PLC

5.1 Entradas de Corriente Continua

Ahora se revisará como funcionan los circuitos de entrada. Esto permitirá entender como se deben conectar las entradas al PLC. Es nefasto para el PLC una mala conexión, por esto se debe poner mucha atención al proceso de conexión.

Generalmente los módulos de entrada de corriente continua (CC), trabajan con voltajes de 5, 12, 24 y 48 [V]. Debe asegurarse que el dispositivo de entrada cumple con los requerimientos de su aplicación.

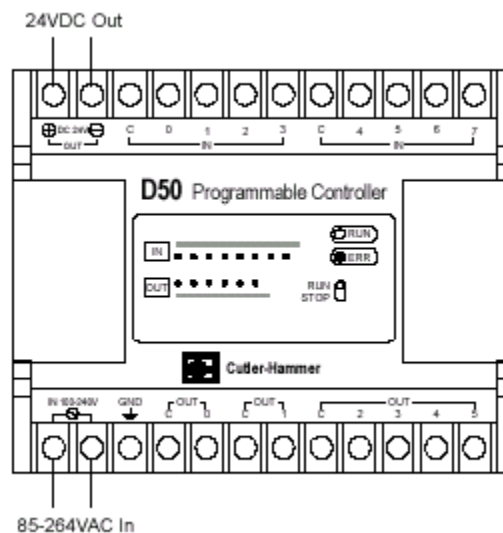


Figura 32. Vista del panel frontal del PLC Samsung SPC-10. Se aprecian las entradas (arriba) y salidas (abajo).

Los módulos de entrada de CC permiten conectar transistores del tipo PNP (fuente) o NPN (carga). Si se está utilizando un interruptor común no es necesario preocuparse de que tipo es. Sin embargo, hay que tener en cuenta que la mayoría de los PLC no permiten mezclar dispositivos NPN o PNP en un mismo módulo. Cuando se está utilizando un sensor es necesario averiguar si se debe considerar como NPN o PNP (verifique el tipo con el fabricante si es necesario).

La diferencia entre ambos tipos influye cuando la carga es conectada a tierra o a un voltaje positivo. En un sensor de tipo NPN tiene la carga conectada a tierra (Figura 33), mientras que un dispositivo PNP tiene la carga conectada a un voltaje positivo.

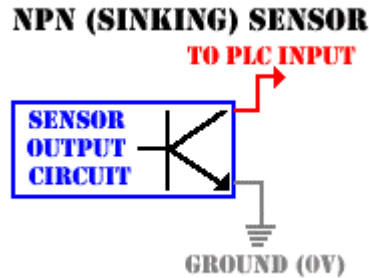


Figura 33. Esquema del circuito de un sensor NPN.

En un sensor NPN se debe conectar una salida (del sensor) al PLC y la otra salida a la conexión a tierra de la fuente de poder. Si el sensor no es energizado por la misma fuente que el PLC, se deben conectar ambas conexiones de tierra juntas. Los sensores NPN son los más usados en Norteamérica.

En un sensor PNP se conecta una salida a un voltaje positivo y la otra salida a la entrada del PLC (ver Figura 34). Si el sensor no es energizado por la misma fuente que el PLC, se deben conectar los dos terminales positivos (del sensor y del PLC) juntos. Los sensores PNP son los más utilizados en Europa.

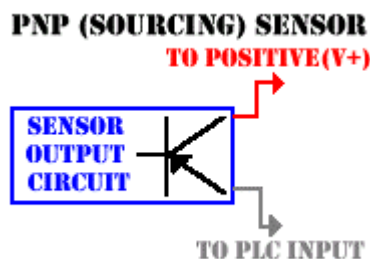


Figura 34. Esquema del circuito de un sensor PNP.

Dentro del sensor, el transistor actúa como un interruptor. El circuito interno del sensor indica al transistor de salida encenderse cuando el objetivo está presente. Entonces el transistor cierra el circuito entre las dos conexiones mostradas antes (voltaje positivo y la entrada del PLC).

Las únicas cosas accesibles al usuario del PLC son los terminales indicados como COMMON, INPUT X, INPUT X+1, etc. el terminal común debe ser conectado a al voltaje positivo o a tierra. A donde se conecte depende del tipo de sensor que se utilice de acuerdo a lo expuesto en párrafos anteriores. Cuando se usa un sensor NPN se debe conectar a voltaje positivo. Cuando se usa un sensor PNP el terminal se debe conectar a tierra. En ambos casos el otro terminal se debe conecta a la entrada del PLC (ver Figura 35).

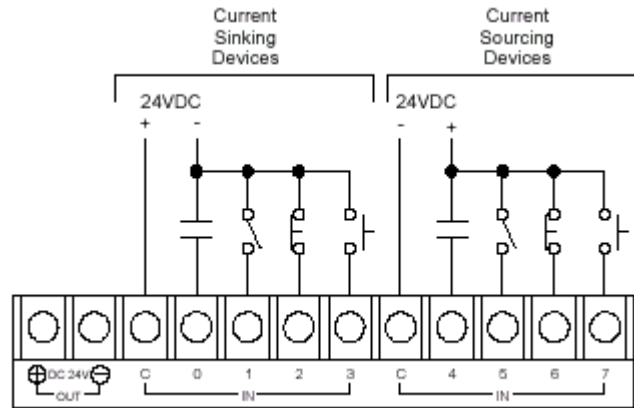


Figura 35. Esquema de las conexiones de sensores NPN y PNP en el PLC Samsung SPC-10.

Algunos fabricantes utilizan en los circuitos de entrada optoacopladores. De esta forma se aíslan los circuitos internos del PLC. Esto elimina la posibilidad de que cualquier interferencia eléctrica entre a los circuitos internos del PLC. Estos trabajan convirtiendo la señal de entrada eléctrica en luz, y luego convirtiendo la señal luminosa en un impulso eléctrico que será procesado en los circuitos internos del PLC.

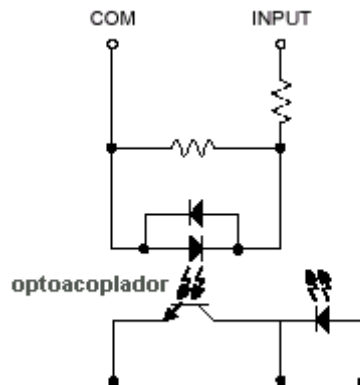


Figura 36. Esquema del circuito de sensores a través de optoacopladores.

5.2 Salidas a Relés

Uno de los tipos de salida más comunes son las salidas a relés. Las salidas a relés pueden ser ocupadas en cargas conectadas a corriente alterna o corriente continua. En estos términos, una carga será cualquier aparato que se acciona con esta salida del PLC. Se denomina carga porque impone un esfuerzo o carga a los circuitos de salida. Si no se conecta ninguna carga a la salida sería el equivalente a reemplazar una ampolla por un trozo de alambre. En este caso permitiríamos fluir una gran cantidad de corriente que de seguro dañaría la salida (no haga el experimento!). En resumen, la carga debe estar correctamente conectada (ver esquema de la Figura 37).

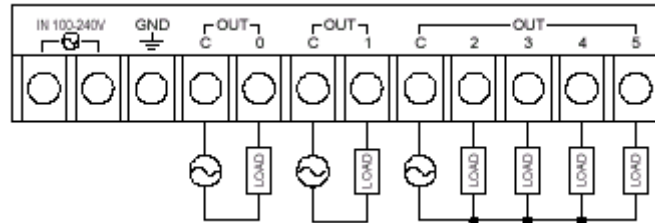


Figura 37. Conexión de las cargas a las salidas a relés del PLC Samsung SPC-10.

Algunas cargas que comúnmente se conectan a la salida del PLC son motores, solenoides, ampollitas, etc. Estas cargas son de diferentes tamaños, es decir difieren en la cantidad de corriente consumida. Entonces siempre se debe conocer la máxima corriente consumida por la carga y revisar si esta se encuentra entre los límites de las especificaciones del fabricante del PLC (Figura 38).

Specifications

Rated Voltage	110/220VAC; 30VDC	
Operating Voltage Range	85 to 132VAC	
Electrical Life	200,000 operations @ rated current	
Mechanical Life	10M operations	
Max. Load Current	Per Output	2A
	Per Common	4A
Min. Load Current	Per Output	30mA
Output Delay Time	Off → On	Less than 10ms
	On → Off	Less than 10ms
Number of Outputs	6 points	
Points per Common	2 Isolated, 1 group of 4 points/common	
Isolation	Photocoupler	

Figura 38. Especificaciones de carga admisible, duración y tiempo de respuesta de las salidas de relé del PLC Samsung SPC-10.

Algunas cargas son muy problemáticas, y corresponden a las cargas inductivas, que tienen la tendencia a devolver corriente al circuito cuando estas son conectadas, corriente estimada en unas 30 veces la corriente de consumo nominal. Esto genera picos de voltaje que pueden dañar la salida a la que esta conectada la carga. Para minimizar estos riesgos se utilizan comúnmente diodos, varistores u otros circuitos de protección.

Aunque se ha mostrado el relé conectado a fuentes de corriente alterna, las salidas de relés también pueden conectarse a salidas de corriente continua.

Los relés son dispositivos internos en el PLC y tienen una configuración similar a la mostrada en el esquema. Cuando el diagrama de escalera indica al PLC que encienda una salida, el PLC aplica internamente un voltaje a la bobina del relé. Este voltaje será el apropiado para que el contacto se cierre. Cuando el contacto se cierra, se permite el flujo de una corriente externa en

el circuito externo o controlado por el PLC. Si el diagrama indica al PLC que apague una salida, el PLC retira el voltaje a la bobina del relé que se aprecia en la Figura 39.

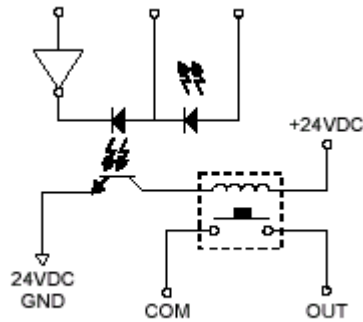


Figura 39. Esquema del circuito interno que acciona el relé de una salida del PLC Samsung SPC-10. Se muestra la aislación de la salida a través de optoacoplador y su relación con el circuito externo.

5.3 Salidas a Transistores

Otra clase de salida que se debe conocer es la salida a transistores. Es importante destacar que un transistor sólo puede encender o apagar un circuito de corriente continua.

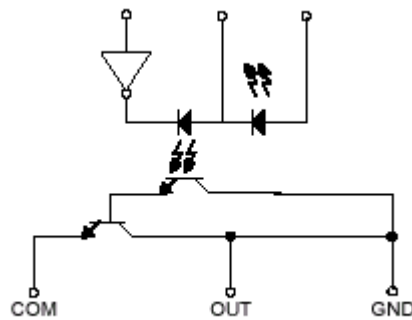


Figura 40. Esquema del circuito interno que acciona el transistor de una salida del PLC Samsung SPC-10. Se muestra la aislación de la salida a través de optoacoplador y su relación con el circuito externo.

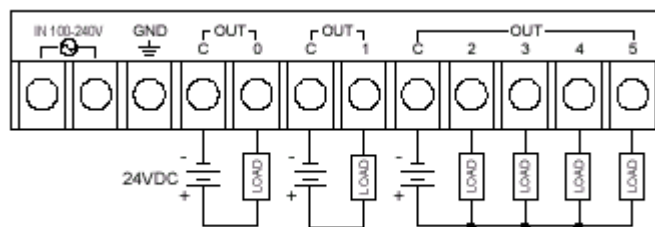


Figura 41. Conexión de las cargas a las salidas a transistor del PLC Samsung SPC-10

Specifications

Rated Voltage		24VDC
Operating Voltage Range		5 to 27VDC
Max. Load Current	Per Output	0.5A
	Per Common	4A
Min. Load Current	Per Output	10mA
Output Delay Time	Off → On	Less than 1ms
	On → Off	Less than 1ms
Number of Outputs		6 points
Points per Common		2 Isolated, 1 group of 4 points/common
Isolation		Photocoupler

Figura 42. Especificaciones de carga admisible y tiempo de respuesta de las salidas a transistor del PLC Samsung SPC-10.