

TEMPORIZADORES

Hasta el momento hemos utilizados dos tipos de espacios de memoria del PLC. Las **entradas**, aquellas que reaccionan al estado de la entrada física y se prenden y apagan según los sensores o mandos que estén conectados. Y las **salidas**, aquellas que se prenden o se apagan según el resultado de la ejecución del programa y que cierran o abren una salida física para activar o desactivar los actuadores o señalizadores que estén conectados.

Aparte de estos dos, el PLC dispone de una gran variedad de tipos de espacios de memoria. Uno de ellos es el espacio de memoria de **temporizadores**. Este espacio trae asociado, como su nombre lo indica, la función de contar tiempo, muy útil en prácticamente todas las aplicaciones de programación en máquinas industriales.

Veamos primero los tipos de temporizadores más comunes que existen.

TIPOS DE TEMPORIZADORES

Vamos a destacar cuatro tipos de temporizadores (el fabricante puede disponer de una cantidad mayor de tipos de temporizadores en su PLC, para conocerlos es necesario leer el manual del equipo).

- Temporizador On delay
- Temporizador Off delay
- Temporizador Intermitente
- Temporizador Monoestable

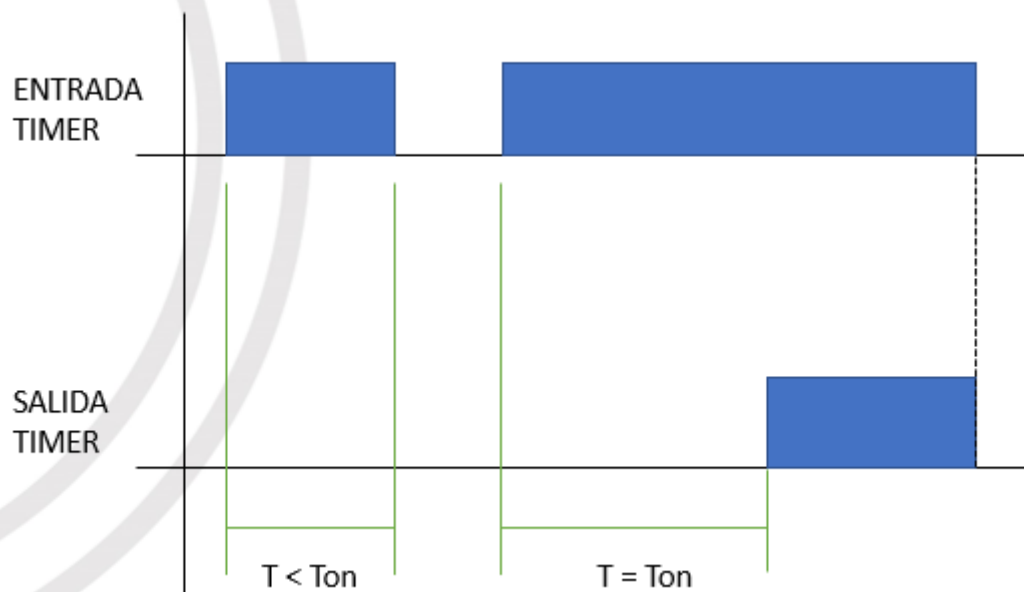
Nuestro PLC marca Izumi del laboratorio dispone de temporizadores tipo **On delay**, el cual explicaremos con mucho más detalle. Igualmente empecemos viendo cada uno de estos de forma resumida

TEMPORIZADOR ON DELAY

Descripción:

- El temporizador On delay comienza a contar tiempo en el instante en que recibe una señal de entrada.
- Mientras esta señal de entrada se mantenga encendida, el temporizador sigue contando tiempo.
- Si la señal de entrada desaparece antes de alcanzar el tiempo requerido, no sucede nada (y el tiempo contado por el temporizador se reestablece).
- En cambio, si la señal de entrada alcanza el tiempo requerido, el temporizador pasa a mandar señal de salida.
- Si luego de esto, el temporizador deja de recibir señal de entrada, se reestablece y deja de mandar señal de salida.

Veamos esta misma descripción en un diagrama lógico:



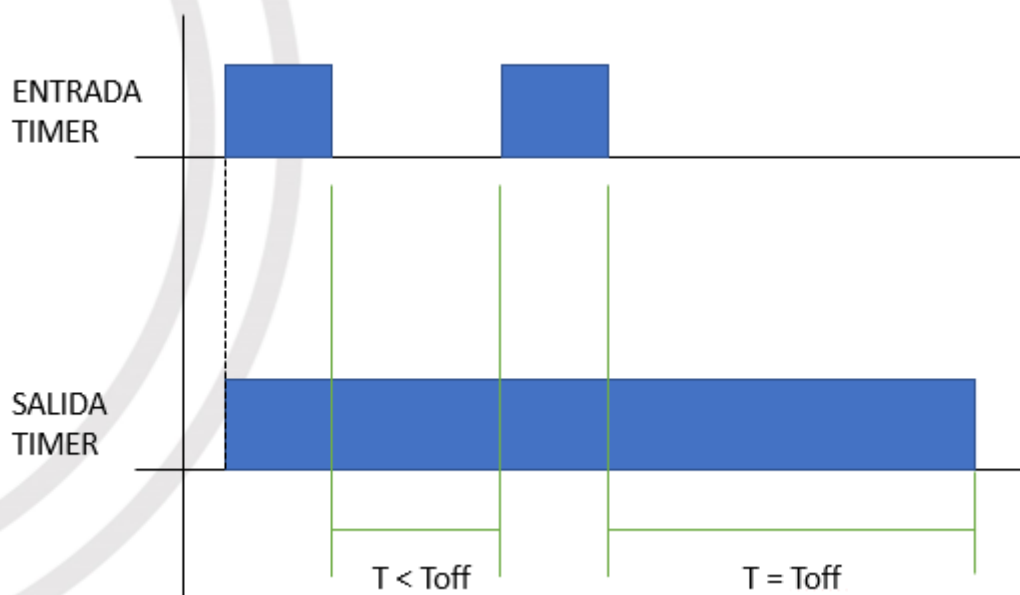
Hemos usado Ton como el valor del temporizador a alcanzar.

TEMPORIZADOR OFF DELAY

Descripción:

- En cuanto el temporizador Off delay recibe señal de entrada, manda señal de salida.
- Mientras esta señal de entrada se mantenga encendida, el temporizador se mantiene mandando señal de salida sin contar tiempo.
- Si la señal de entrada desaparece, el temporizador sigue mandando señal de salida y empieza a contar el tiempo.
- Si la señal de entrada vuelve a aparecer antes de terminar la cuenta del tiempo, el temporizador sigue mandando señal de salida y el tiempo contado se reestablece.
- Si por el contrario, el temporizador alcanza el tiempo a contar, deja de mandar señal de salida y se reestablece.

Veamos esta misma descripción en un diagrama lógico:



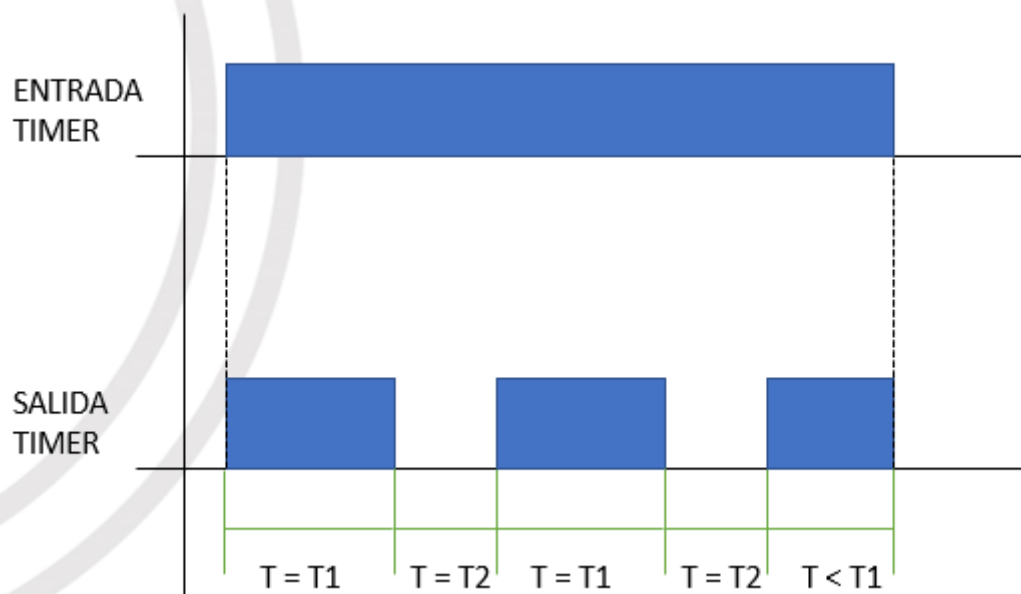
Hemos usado T_{off} como el valor del temporizador a alcanzar.

TEMPORIZADOR INTERMITENTE

Descripción:

- En cuanto el temporizador Intermitente recibe señal de entrada, manda señal de salida.
- Mientras esta señal de entrada se mantenga encendida, el temporizador comienza a contar un primer tiempo (T1).
- Cuando este tiempo T1 alcanza el valor establecido, el temporizador deja de mandar señal de salida y comienza a contar un segundo tiempo (T2).
- Cuando este tiempo T2 alcanza el valor establecido, el temporizador vuelve a mandar señal, los dos temporizadores se reestablecen y nuevamente cuenta el tiempo T1.
- Este ciclo de T1 y T2 se repite hasta que desaparece la señal de entrada, con lo que el temporizador reestablece sus tiempos y deja de mandar señal de salida.

Veamos esta misma descripción en un diagrama lógico:



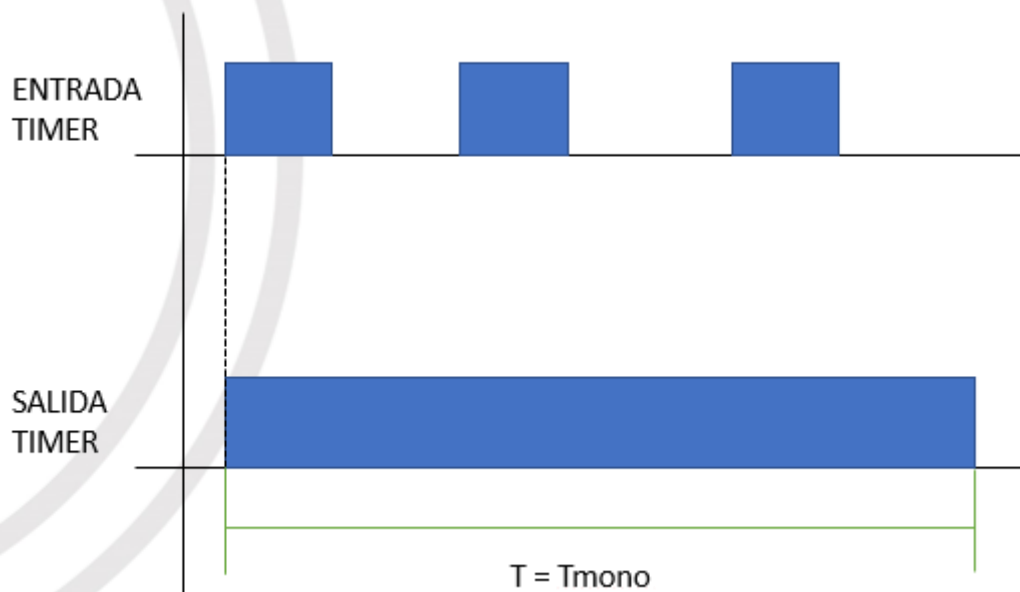
Hemos usado T1 y T2 como los valores del temporizador a alcanzar.

TEMPORIZADOR MONOESTABLE

Descripción:

- En cuanto el temporizador Monoestable recibe señal de entrada, manda señal de salida y empieza a contar el tiempo.
- Si la señal de entrada desaparece, el temporizador sigue mandando señal de salida y sigue contando el tiempo.
- Si la señal de entrada vuelve a aparecer, el temporizador no se altera, continua mandando señal de salida y contando el tiempo.
- Cuando el temporizador alcanza el tiempo a contar, deja de mandar señal de salida y se reestablece.

Veamos esta misma descripción en un diagrama lógico:

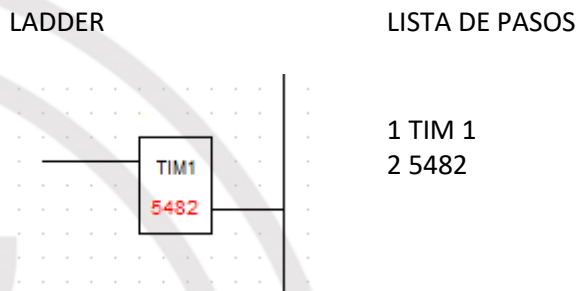


Hemos usado T_{mono} como el valor del temporizador a alcanzar.

CARACTERÍSTICAS DE LOS TEMPORIZADORES

Para todos los tipos de temporizadores las siguientes características son válidas. Vamos a centrarnos en destacar las características refiriéndonos al temporizador On delay.

- Bloque de programación



El bloque de programación se ubica **al final** de una línea de programación, de manera similar a como lo hace la instrucción OUT. El bloque de programación del temporizador posee la siguiente información:

TIM 1: Define que la función a programar es un temporizador y que el espacio de memoria de temporizador a utilizar es el 1 (el manual explica cuántos temporizadores tiene el PLC).

Debajo de TIM 1 se escribe el número a alcanzar en la cuenta del temporizador. Dicho número se lo conoce como **valor de seteo** y expresa cuántas unidades debe contar el temporizador para activarse. En este caso serían 5482 unidades. La pregunta que sigue es cómo sabemos si estas unidades son, por ejemplo, segundos. Esto lo sabemos viendo en el manual del PLC el valor de **base de tiempo** que posee el temporizador elegido (en este caso el número 1). La base de tiempo es la mínima unidad de cuenta y define en qué unidad cuenta el temporizador. Ejemplos de base de tiempo pueden ser

- 1 segundo
- 1 minuto
- 1 décima de segundo
- 1 centésima de segundo

Para el ejemplo, una base de tiempo de 1 segundo transforma el valor en justamente 5482 segundos. Una base de tiempo de 1 minuto produce una cuenta de 5482 minutos. Por otra parte una base de tiempo de décima de segundo (0,1 segundos) hace que el tiempo sea de 548,2 segundos.

El PLC Izumi de nuestro laboratorio posee temporizadores de base de tiempo de 0,1 segundos.

La fórmula para calcular esto es: $\text{Tiempo} = \text{Valor de seteo} \times \text{Base de tiempo}$.

Sin embargo, normalmente conocemos el tiempo que queremos alcanzar (parte de las características de la máquina que vamos a automatizar) y sabemos la base de tiempo por elegir el número de temporizador, por lo que la fórmula la podemos reescribir como

Valor de seteo = Tiempo / Base de tiempo

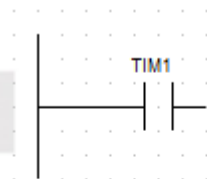
Con el tiempo que queremos contar y la base de tiempo del temporizador elegido, hacemos la división y obtenemos el valor de seteo que debemos introducir en el PLC.

En cuanto a las instrucciones de la lista de pasos, para la programación en lista de pasos se requieren dos líneas. La primera TIM 1 llama a la función del temporizador y la segunda S482 escribe el valor de seteo de ese temporizador (la segunda línea requiere solamente escribir el número).

- Contacto del temporizador

LADDER

LISTA DE PASOS



1 LOD T1



1 LOD NOT T1

El bloque de temporizador se relaciona con sus contactos al igual que como las entradas o las salidas del PLC. Cuando el bloque del temporizador se activa (en el caso del On delay, cuando se cumple el tiempo) los contactos reaccionan cambiando de estado. El contacto que se ve arriba se prende cuando el temporizador cumple la cuenta. El contacto de abajo, al estar programado de manera inversa, está prendido mientras el temporizador no cumpla la cuenta y al terminar de contar, se apaga.

- Alcance

El valor de **alcance** se refiere al máximo valor que puede alcanzar el PLC cuando se programa el máximo valor de seteo. Para el PLC Izumi del laboratorio, el máximo valor de seteo es de 9999. Por ende un temporizador de base de tiempo de 1 segundo alcanza 9999 segundos (unas 2 horas y 45 minutos aproximadamente), mientras que un temporizador de base de tiempo de 0,1 segundos alcanza 999,9 segundos (unos 16 minutos y 40 segundos aproximadamente).

Si queremos contar más allá del alcance de un temporizador es una buena práctica hacer contar primero el temporizador de mayor base de tiempo, el cual tiene más alcance, y cuando culmina,

activar el siguiente temporizador de menor base de tiempo para poder regular con mayor exactitud cuánto tiempo debe contar el temporizador.

REPRESENTACIÓN DEL VALOR NUMÉRICO DEL TEMPORIZADOR

En el apunte donde explicamos cómo se representan los números decimales en binario hemos comentado cómo hace el PLC para almacenar valores números en representación de bits de 0 y 1. Esto mismo se aplica al guardar los valores del temporizador.

En el ejemplo que vimos hasta ahora, el valor de seteo a programar es de 5482. Dicho valor de cuatro dígitos se almacena en cuatro grupos de cuatro bits cada uno de la siguiente manera

5 = 0101
4 = 0100
8 = 1000
2 = 0010

Lo que provoca que el valor de seteo se escriba en el PLC como 0101010010000010

Por otra parte, el temporizador tiene que contar el tiempo. En el PLC Izumi del laboratorio, el temporizador toma el valor de seteo y le **descuenta 1 por cada unidad de base de tiempo** (en el caso de ese PLC, la base de tiempo es de 0,1 segundos). Por ende, cada décima de segundo le baja su valor en 1.

Si hacemos contar nuestro temporizador, pasamos de 5482 a 5481

5 = 0101
4 = 0100
8 = 1000
1 = 0001

Por lo que el valor queda 0101010010000001. Si seguimos

5 = 0101
4 = 0100
8 = 1000
0 = 0000

Por lo que el valor queda 0101010010000000. Si seguimos

5 = 0101
4 = 0100
7 = 0111
9 = 1001

Por lo que el valor queda 0101010001111001.

Así seguiríamos hasta alcanzar el 0, que es donde todos los bits quedan en 0 (0000000000000000). Es en ese momento en el que el PLC sabe que el temporizador terminó de contar y se activa.

Algo que tenemos que tener en cuenta es que programar un temporizador le demanda al procesador mucha más tarea que simplemente una entrada o una salida. En una entrada o una salida el PLC solo debe verificar el estado de un bit (prendido o apagado). En un temporizador, debe verificar el estado de 16 bits (los 4 grupos de 4 que representan el valor de seteo), por lo que el trabajo del procesador se multiplica 16 veces cuando se le pide hacer contar un temporizador. Los procesadores suelen ser suficientemente rápidos como para que no notemos esta diferencia. Pero no podemos pasar por alto el hecho de que el procesador puede demorar un poco más en responder por el hecho de estar ocupado haciendo trabajar al temporizador.