27-3-2017

Pablo González Martín

INFORME SEMÁFORO

AUTOMATIZACIÓN INDUSTRIAL

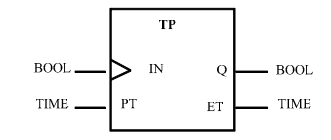
En esta primera práctica haremos una primera pequeña toma de contacto para recordar los conocimientos dados anteriormente. Para ello, estudiaremos el comportamiento simple de un problema físico sencillo en el que se emplean contadores y temporizadores; de forma que vayamos familiarizándonos con los elementos del S7-300 para las prácticas venideras.

Cuestiones previas

La norma IEC 61131‐3 define explícitamente 3 tipos de temporizadores que deben ser soportados por cualquier lenguaje de programación para PLC:

* Temporizador de pulso (TP): Se activa la salida lógica Q tras un flanco de subida de la señal de disparo. Se desactiva al transcurrir PT.
* Temporizador de retardo a la conexión (TON): Se activa Q después de transcurrir PT tras producirse un flanco de subida de la señal disparo.
* Temporizador de retardo a la desconexión (TOFF): Se desactiva Q cuando transcurre PT tras producirse un flanco de bajada de la señal de disparo.

Intuitivamente el temporizador TP sigue al disparo y cae tras el paso del tiempo. En cambio, el temporizador TON  tarda un tiempo en seguir a la señal una vez conectada (de ahí el nombre). Finalmente, el TOFF se activa tras conectarse la señal y tras su desconexión, Q tarda un tiempo en caer (véase figura 1). La señal IN es la señal de disparo, la señal PT (Preset Time) es el valor de carga, Q es la salida lógica y ET (Elapsed Time) es el tiempo transcurrido desde la puesta en marcha (con valor máximo PT).



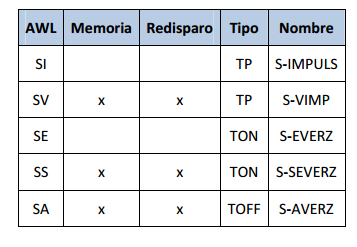
**Figura 1: Bloque de temporización descrito según la norma**

Además de estas tres tipologías bien definidas, existen dos propiedades muy importantes para caracterizar el comportamiento del temporizador: la memoria y la redisparabilidad.

Memoria: Describe el comportamiento de un temporizador activo con la llegada del primer flanco opuesto al de activación de la señal de disparo (si se dispara con el flanco de subida, la memoria hace referencia al comportamiento ante el primer flanco de bajada posterior y viceversa). Se dice que un temporizador tiene memoria cuando tras su puesta en marcha, continúa con la temporización ante la llegada del primer flanco opuesto al de disparo. Es decir, el temporizador se acuerda de lo que estaba haciendo antes del evento y continúa con ello. Si no tiene memoria, para la cuenta y se desactiva (Q=0).

Redisparabilidad: Describe el comportamiento de un temporizador activo con la llegada del primer flanco de disparo posterior al de activación. Esta propiedad sólo describe a los temporizadores con memoria ya que los que no tienen memoria ven al redisparo como un disparo convencional. Se dice que un temporizador es redisparable cuando, tras su puesta en marcha, reinicia la cuenta atrás con el siguiente disparo.

STEP7 realiza la especificación ofreciendo 5 temporizadores distintos que se describen en la tabla 1:

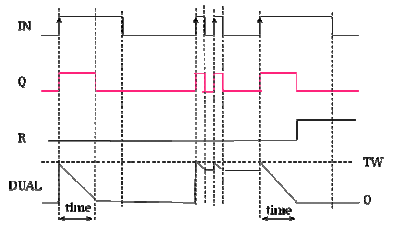


**Tabla 1: Temporizadores en Step 7**

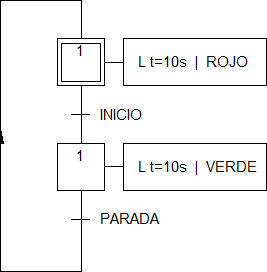
La columna de la izquierda es el nemónico del operador AWL para cada temporizador. Se emplea como operador en una instrucción del tipo: S T y tiene dos argumentos implícitos, la señal de disparo y el valor de tiempo de carga, que los obtiene de los registros de la ALU: la señal del biestable RLO y el tiempo de carga de un registro denominado Acumulador de 32 bits que se emplea para manipular tamaños de datos superiores al bit.

* Disparo de un Temporizador SI (Temporizador de disparo de pulso)

El temporizador sigue a la entrada tras ser disparado y se desconecta cuando pasa el tiempo (véase fig. 2). Al no tener memoria, se desactiva si la señal de disparo cae. La gráfica también muestra la variación de la salida DUAL correspondiente a la cuenta atrás desde que se puso en marcha.

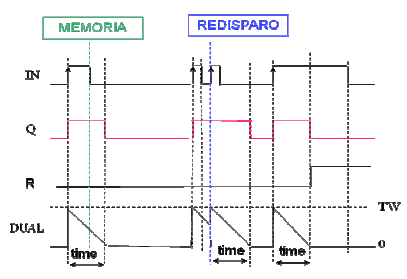


**Figura 2: Muestra el comportamiento de un temporizador disparado por SI**

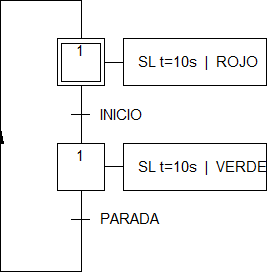


* Disparo de un Temporizador SV (Temporizador de disparo de Pulso prolongado)

Se comporta de manera similar al de pulso normal (SI) en la puesta en marcha, pero tiene memoria luego continúa funcionando ante la caída de la señal de disparo (véase fig 3). Finalmente, al ser redisparable reinicia la cuenta atrás ante un nuevo evento disparo. Se indica en la figura el evento que refleja ambas propiedades.

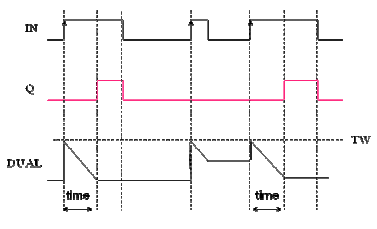


**Figura 3: Muestra el comportamiento del temporizador de impulso prolongado**

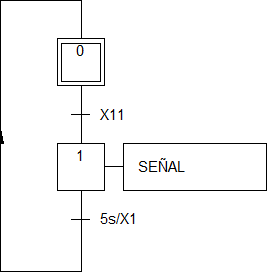


* Disparo de un Temporizador SE (Temporizador de disparo de retardo a la conexión)

Con la puesta en marcha el temporizador empieza la cuenta atrás pero no sigue a la señal de disparo hasta que termina la cuenta, (véase fig. 4). Como no tiene memoria el temporizador se desactiva con el flanco de bajada de la señal de disparo. Activada la salida lógica, el temporizador se mantiene en este estado hasta que cae la señal de activación. El funcionamiento de la entrada Reset es análogo a los casos anteriores por lo que se ha omitido.

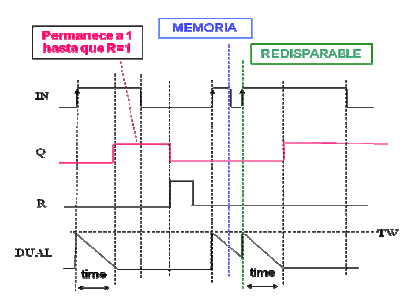


**Figura 4: Describe el comportamiento del TON sin memoria que ofrece STEP7.**

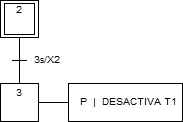


* Disparo de un Temporizador SS (Temporizador de disparo de Retardo a la desconexión)

Los cronogramas reflejan el comportamiento esperado habida cuenta de que este temporizador tiene memoria y es redisparable, (véase fig. 5). Como nota destacable hay que tener en cuenta que este temporizador, una vez activada su salida lógica, requiere un evento Reset (por nivel, como siempre) para que la salida se desactive. Estando a nivel alto Q, no atiende a eventos de disparo. Esta característica no es compartida por ninguno de los otros temporizadores, y hay que tenerla muy presente. Se resume en que, una vez necesita un reinicio explícito para volver a ser utilizado.

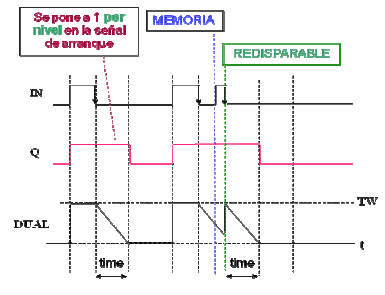


**Figura 5: Describe el comportamiento del temporizador, el cual es una versión con memoria del TON anterior.**

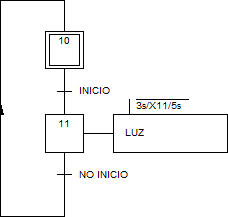


* Disparo de un Temporizador SA (Temporizador de disparo de Retardo a la desconexión)

Este es el único temporizador de tipo TOFF programable en STEP7, tiene memoria y es redisparable. Al ser de retardo a la desconexión, este temporizador activa su salida lógica con el nivel alto de la señal de disparo. Estando conectado y tras recibir un flanco de bajada por la entrada de disparo se pone en marcha y no se desconecta (Q=0) hasta que ha pasado el tiempo establecido. Al ponerse en marcha con el flanco de bajada de la señal de disparo, la memoria tiene que ver con el flanco de subida siguiente y el redisparo con un nuevo flaco de bajada.

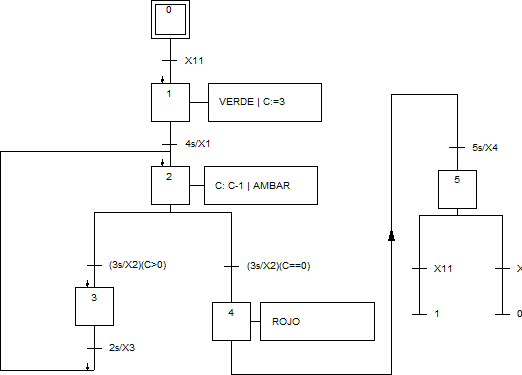


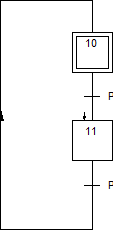
**Figura 6: Resume el comportamiento de un temporizador SA**



DESARROLLO DE LA PRÁCTICA

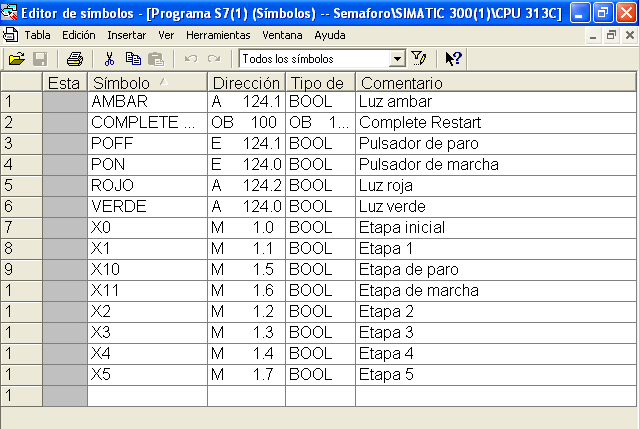
Para el desarrollo de la práctica en cuestión se nos plantea como problema la programación y asignación de responsabilidades de las luces de un semáforo para coches. Lo primero que haremos será la asignación de responsabilidades, como se trata de un caso práctico bastante sencillo lo que haremos será la asignación de responsabilidades mediante un Grafcet, (véase figura 7).





**Figura 7: Describe mediante un método gráfico (Grafcet) la asignación de responsabilidades del semáforo.**

Al igual que hemos realizado la asignación de responsabilidades mediante el método gráfico también es necesario realizar un mapeo de E/S, para poder realizar una comprensión de la simbología abstracta con la que se desarrolla el Grafcet (véase tabla 2).

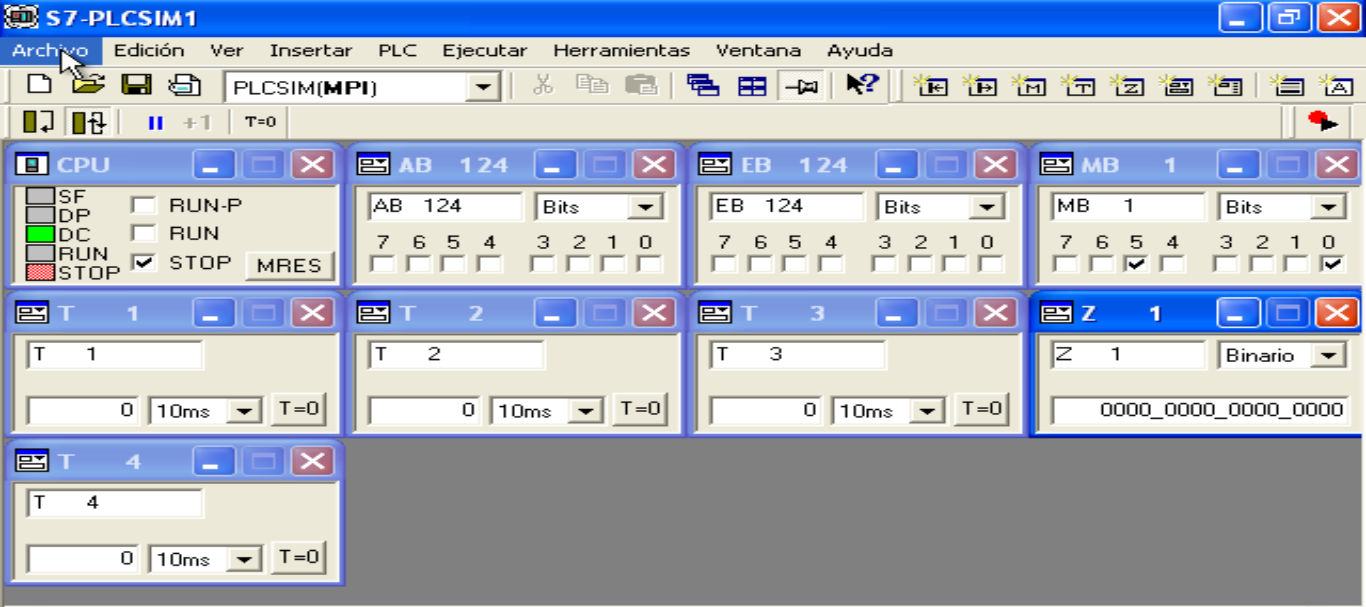


**Tabla 2: Conjunto de E/S que describen el sistema.**

Una vez llegados a este punto y con la asignación de responsabilidades realizada al igual que descrito ya el conjunto de E/S que describirán nuestro sistema, procederemos por último lugar a picar el código para que se ejecuten los actuadores, de forma que nuestro sistema físico se comporte de la forma que queremos (Anexo I). Para ello, emplearemos el entorno de programación SIMATIC Manager. En primer lugar, para más tarde no tener problemas debemos de coger las referencias del PLC para posteriormente realizar la carga del programa en el autómata programable y que este entienda nuestras líneas de código. Estas referencias vienen identificadas como el número de Bastidor, número y tipo de CPU; así como los números de referencias de la CP. Una vez que hayamos ya ajustado el hardware, procederemos a insertar en nuestro SIMATIC Manager el OB1 (programa principal) y el OB100.

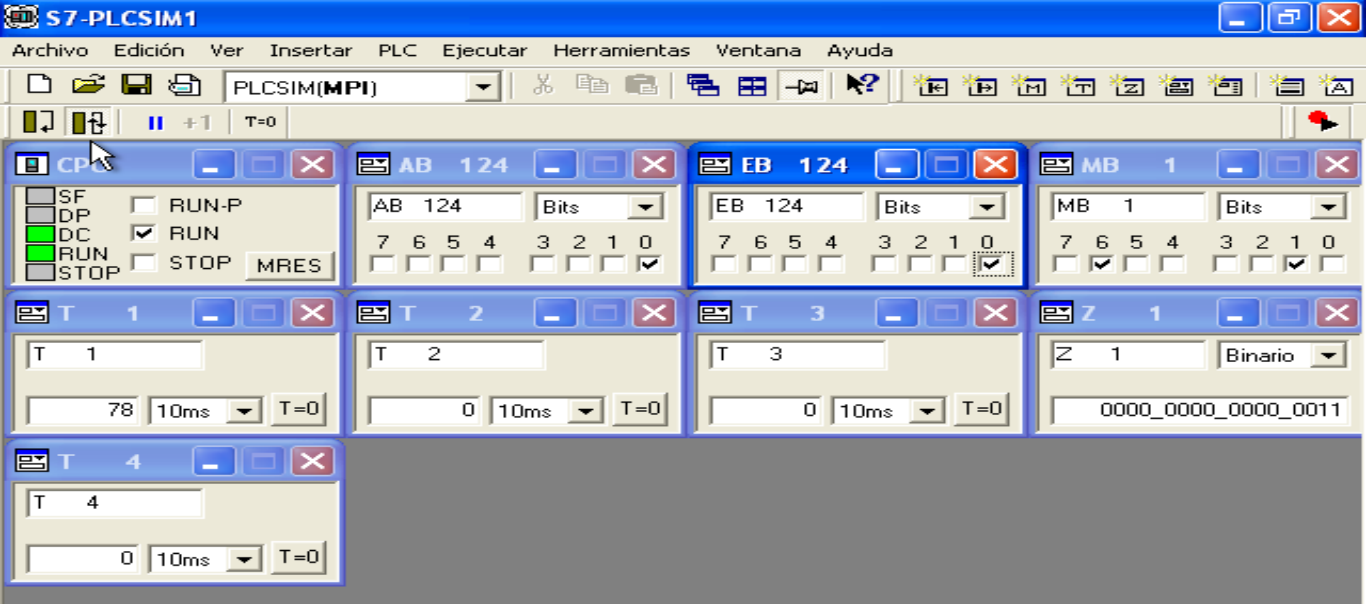
SIMULACIÓN

Con el código picado en el S7, procederemos a probar su funcionamiento en el entorno de simulación PLCSIM. Para ello, dispondremos de una interfaz gráfica tal como la que se muestra en la fig. 9; para poder hacer una lectura de sus valores tendremos que referenciarnos en la tabla 2. De esta forma, quedará reflejada la activación y desactivación de entradas y salidas así como de etapas (marcas) que hemos programado de acuerdo con nuestro Grafcet y que podemos ver si realmente se está cumpliendo lo esperado. Una vez, claro todo esto y pasamos nuestro PLC de STOP a RUN, y comenzamos. En la figura XX podemos ver la situación inicial que nos encontraríamos en él simulador, las marcas M1.0 (Etapa inicial) y M 1.6 (Etapa de parada) activas; a la espera de recibir una entrada para que comience el ciclo de actuación del programa.



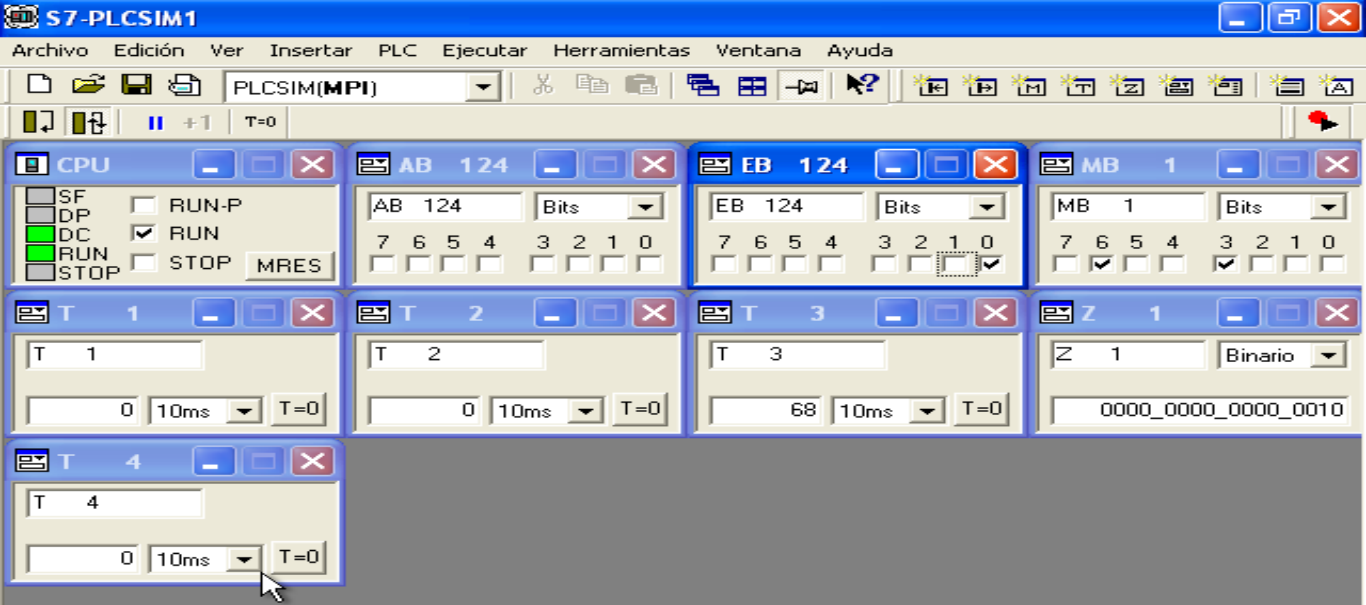
**Figura 9: Estado inicial del PLCSIM, con el interruptor de STOP activo**

Pulsaremos la entrada digital E 124.0 (Pon) para que comience el ciclo de ejecución del semáforo (véase fig. 10), en el veremos cómo inmediatamente se activa la salida A 124.0 (luz verde) y se dispara el contador T1 asociado a dicha acción. De igual manera, podemos ver cómo una vez se activa la salida A 124.0 (luz verde), se realiza la carga del contador Z1, quien nos controlará la permutación de la luz de ambar (A 124.1).

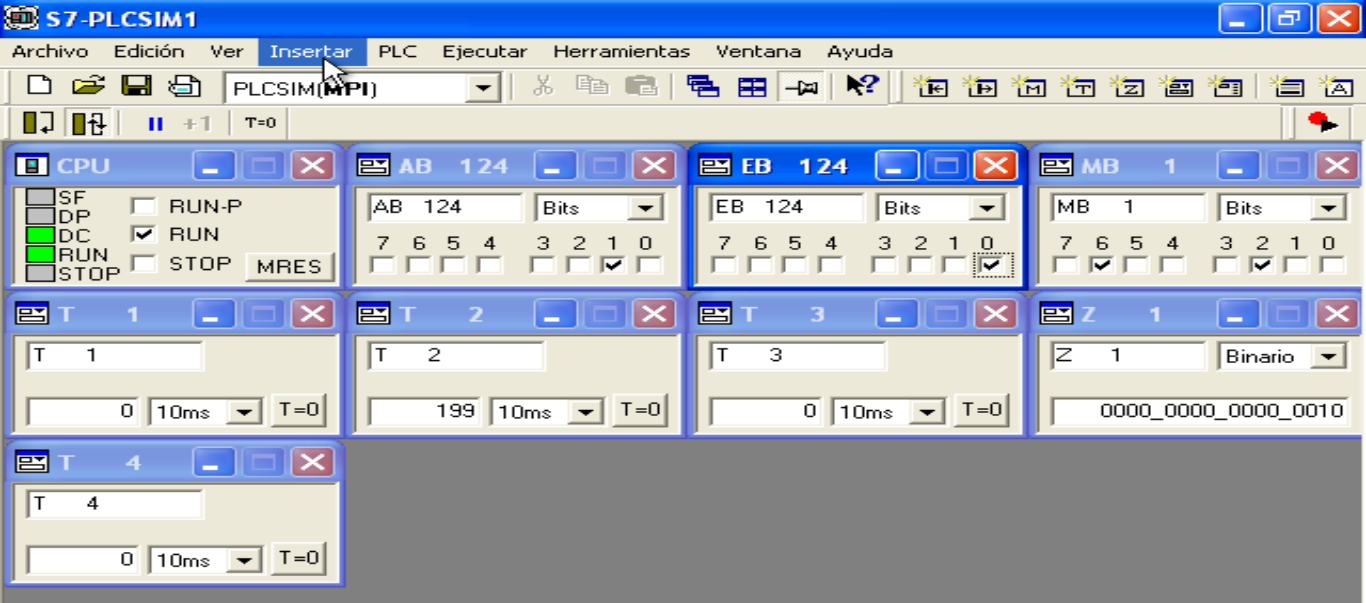


**Figura 10: Captura en la que se ve el tiempo que esta A 124.0 (verde) activa, así como las entradas y marcas que hacen que esta acción suceda.**

El siguiente paso que dará nuestro pequeño programa, es la conmutación del ámbar (véase fig 11 y 12), para ello hemos programado un Contador, que como bien hemos dicho antes se carga con la activación de la salida A 124.0 (luz verde) con un valor nominal de 3; y que decrementa en uno su valor por cada flanco de subida que se da lugar en A 124.1 (luz ámbar). De igual manera, una vez que ha terminado el temporizador T1, este resetea la etapa en la que se encuentra que en este caso se trata de la etapa 1 (correspondiente con la marca M1.1) y setea la etapa siguiente para que pueda tener lugar lo explicado anteriormente (se setea la marcar M1.2).

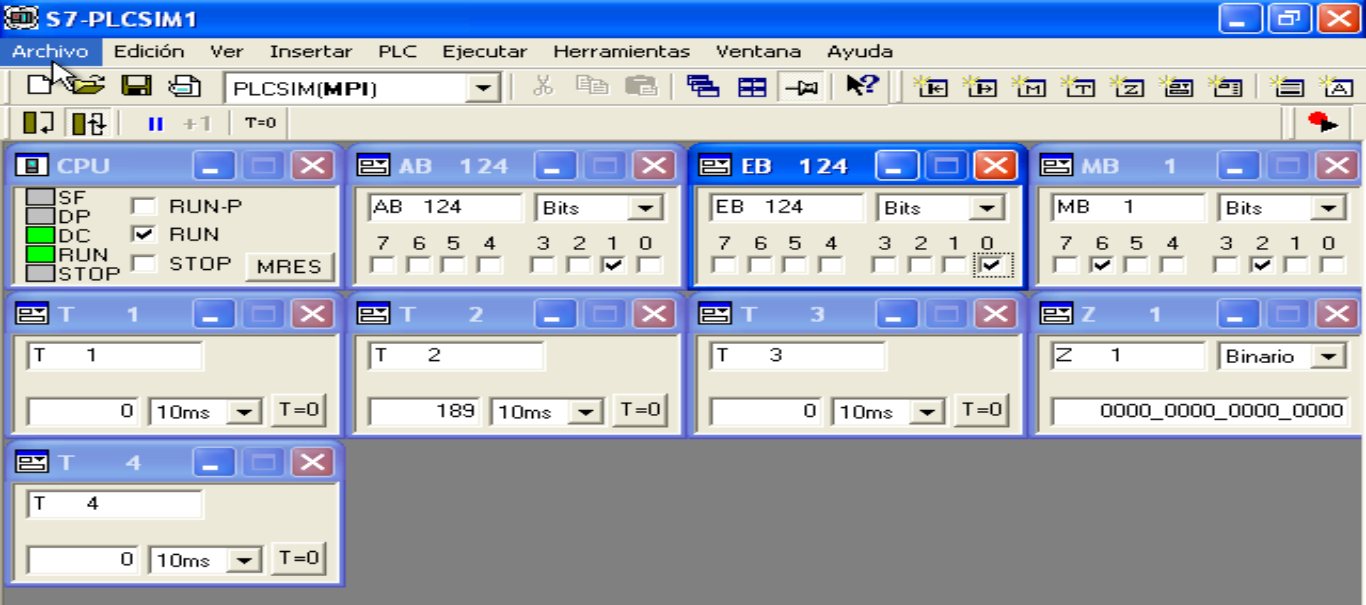


**Figura 11: Podemos la conmutación de la luz ámbar, temporizada por el temporizador T3.**

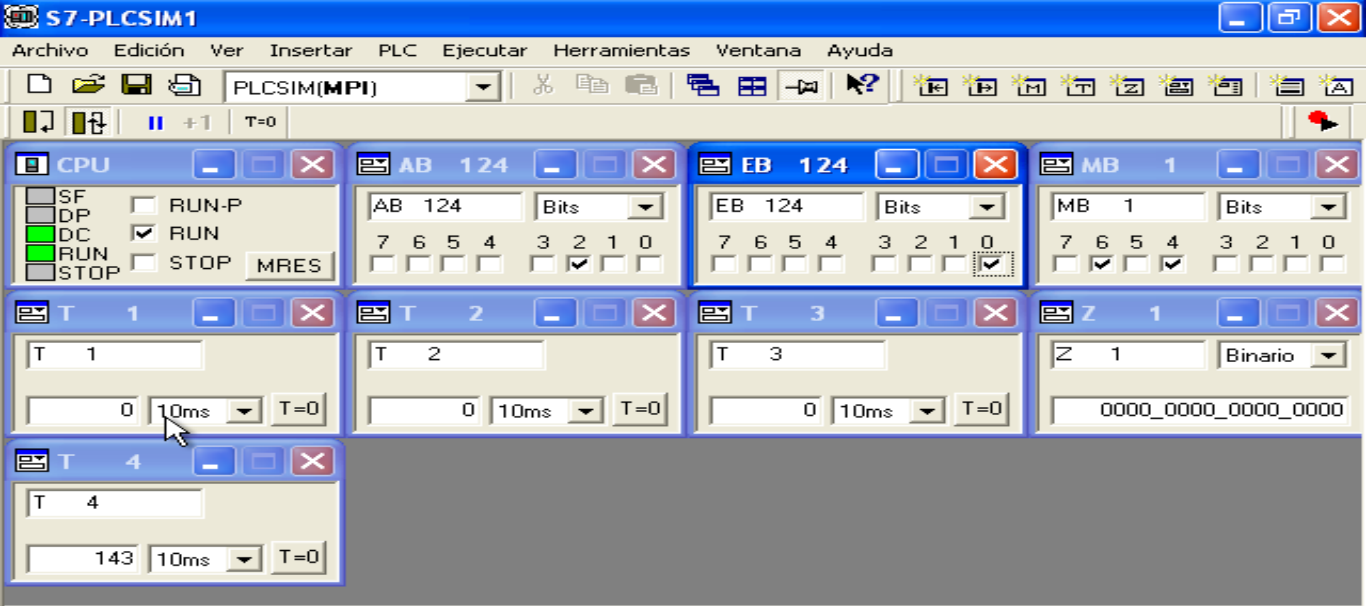


**Figura 12: Podemos observar cómo ha disminuido el valor en el contador en esta imagen debido a los flancos de subida de A 124.1 (luz ámbar).**

La conmutación seguirá sucediendo y la cuenta en función de esta seguirá disminuyendo hasta el momento que el contador pase a valer 0. Momento en el que tras agotarse el tiempo del temporizador T2, se saltará a la etapa 4 (marca M1.4), en la cual se activara la salida A 124.2 (rojo), (véase fig 13 y 14).

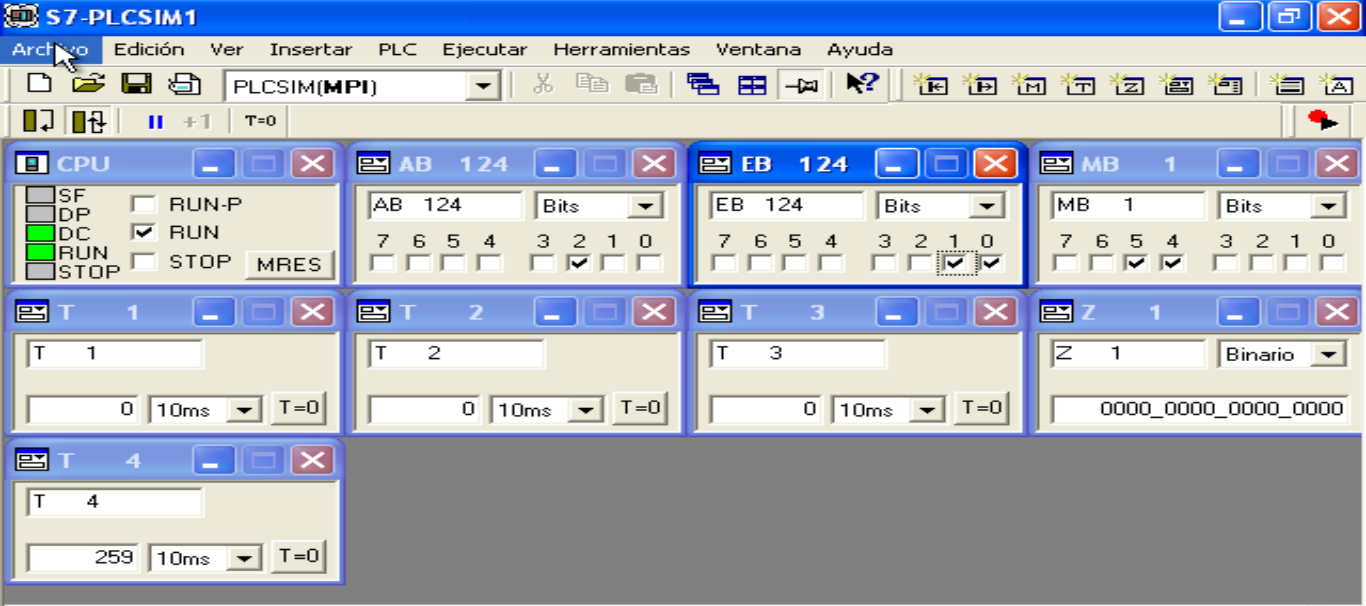


**Figura 13: Podemos observar como el contador llega a 0, y se está esperando a que el temporizador T2 termine, para saltar de la etapa 2 (valor alto de ámbar) a la etapa 4 (luz rojo).**



**Fig 14: Se puede observar cómo se ha pasado a la etapa 4, se ha activado la salida A 124.2 (luz roja) y se está efectuando el tiempo que esta dura.**

Una vez llegados a este punto del programa en la que ya hemos terminado el ciclo (termina el temporizador T4), pasaremos por una etapa fugaz, por una etapa de selección, en ella si esta activa la etapa de marcha (marca M1.6) se volverá a reiniciar el ciclo a partir de la luz verde, volviendo a realizar la carga en el contador y todo lo que con ello conlleva (véase fig. 14). Por consiguiente, si hemos seleccionado la entrada E 124.1 el proceso se detendrá al final, y es por ello que quedará seleccionado en este caso antes de que termine el temporizador T4 la marca M1.5 ( marca de paro), (véase fig. 14).



**Fig. 15: Una vez terminado el ciclo se detendrá el semáforo.**

CONCLUSIÓN

Una vez hemos concluido con éxito la simulación del mismo, procederemos a realizar la prueba del programa en un PLC real marca SIEMENS S7-S300. Para poder hacer la carga de este en el PLC es necesario cerrar previamente el simulador, y realizar la carga en el autómata programable. El funcionamiento ante un programa de una envergadura tan sencilla es completamente idéntico, por lo que el desarrollo de la práctica puede darse por exitoso.