



Universidad de la Defensa Nacional
Facultad de Ingeniería del Ejército

Ingeniería en Informática

Técnicas Digitales II

**Modos de funcionamiento de timers/counters
en el microcontrolador 8051**

Introducción

El **timer** es un elemento de hardware completamente separado de la ejecución de instrucciones. Es un elemento adicional a la CPU que realiza su tarea sin la intervención de instrucciones. Sólo se utilizan instrucciones para configurarlo, pero luego de esto funciona completamente aparte de la CPU y sólo se consulta su estado. Esto se suele hacer mediante interrupciones, por lo que los timers tienen la capacidad de interrumpir al procesador en el caso de un evento importante.

Muchos timers también pueden funcionar como **counters**, que cuentan pulsos. Este modo, no interesa el tiempo que pasa, si no la cantidad de veces que ocurre algo, ya que cuenta eventos. En cambio, el timer cuenta pulsos que van siempre a la misma frecuencia, que representan el paso del tiempo.

En el 8051, la función de timer o counter se selecciona mediante un bit de control (C/T). Los timers/counters tienen 4 modos de operación.

Modos

Modo 1

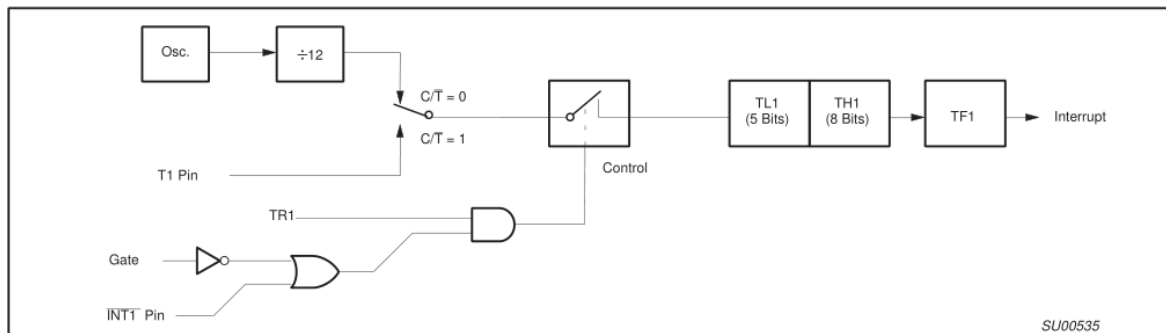
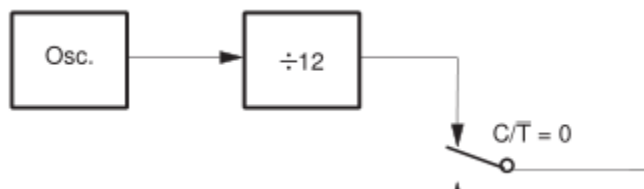
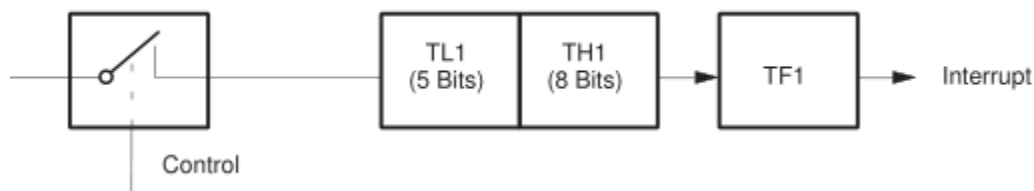


Figure 7. Timer/Counter Mode 0: 13-Bit Counter

En el caso de timer, la llave del bit de control está puesta hacia arriba, con el bit C/T=0.



En la parte superior, se observa el oscilador del microcontrolador y un divisor fijo por 12. Es decir, a la salida de esto se obtiene una frecuencia constante, la del oscilador en el cristal dividida por 12.



Los pulsos llegan a la llave de control. Si la llave está cerrada, los pulsos pasan y se encuentran con dos contadores que cuentan pulsos. Estos en su interior tienen registros indicando cuántos pulsos entraron. Están uno a continuación del otro de manera que cuando el primero llegue al máximo, incrementa el siguiente, teniendo así la parte alta y baja de un número de 16 bits. Cuando desborda el segundo contador (parte alta), se setea el bit TF1 (Timer Flag) y produce la interrupción.

Si la llave del bit de control está hacia abajo, con el bit C/T=1, la entrada viene de un pin externo (T1) en vez desde el oscilador. Esto correspondería a un contador. Pasaría lo mismo que con C/T en 0, pero en este caso los pulsos no llegarán a la misma frecuencia, sino que serán en base a un evento que se desea contar.



Viendo todo esto en general, se puede notar que el funcionamiento es de la siguiente forma: si se tienen las llaves cerradas, como los pulsos vienen a la misma frecuencia, habrá un tiempo en el que se seteará TF1, y se producirá la interrupción, midiendo tiempo (timer). En cambio, si se tiene la llave de C/T abierta, los pulsos no vendrán a la misma frecuencia y se puede contar la cantidad de veces que ocurrió un evento en T1 (counter).

Modo 2

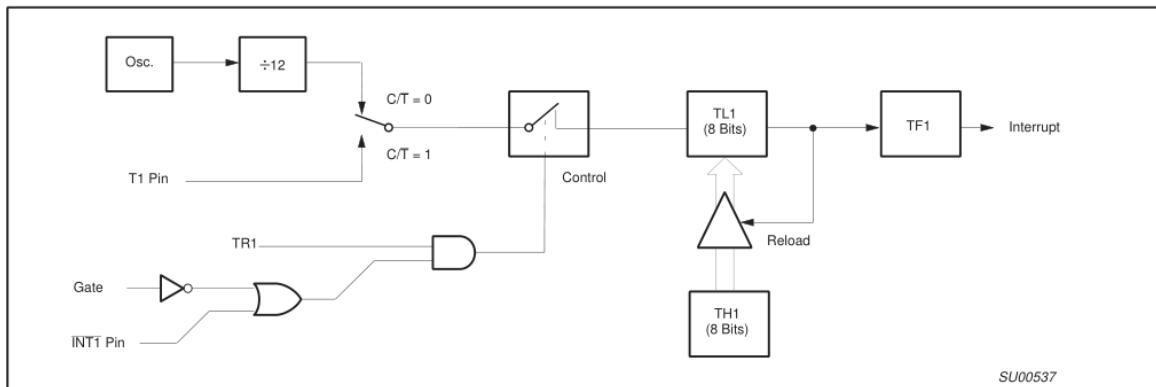
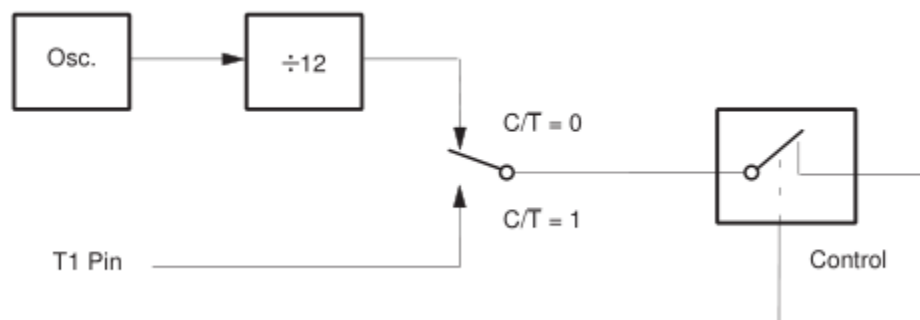
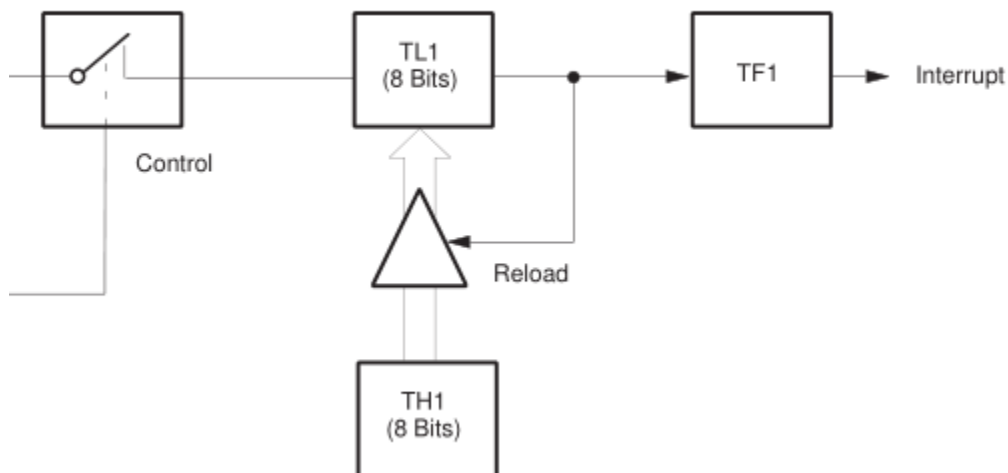


Figure 9. Timer/Counter Mode 2: 8-Bit Auto-Load



En este modo, observamos que todos los componentes anteriores a la llave de control son iguales, es decir que si se tiene la llave hacia arriba ($C/T=0$) se tendrá una entrada con frecuencia fija, midiendo tiempo (timer), y si la llave está hacia abajo ($C/T=1$) se tendrá una entrada respecto al pin T1, midiendo la cantidad de veces que sucede un evento (counter).



En este modo, el registro que incrementará en valor es TL1, mientras que TH1 se mantiene constante, guardando el valor de “reload”. TH1 es precargado con este valor de “reload”.

Una vez que TL1 llegue al máximo, se setea TF1 y se realiza la interrupción. Luego de esto, TL1 en vez de resetearse a 0 como en el modo 1, se reseteará para seguir contando con el valor de “reload” en TH1.

Entonces, por ejemplo, digamos que TH1 tiene cargado el valor FDh y TL1 está en FEh, el conteo sería de la siguiente forma:

Ciclo	TH1	TL1
1	FDh	FEh
2	FDh	FFh
3	FDh	FDh
4	FDh	FEh
5	FDh	FFh
6	FDh	FDh
7	FDh	FEh

Esto puede servir en el caso en el que se quiera empezar a contar desde un valor específico. Para poder hacer esto con el modo 1, se tendrían que escribir instrucciones de código que verifiquen si TL1 se desbordó, y si es el caso, setearle el valor correspondiente.

Modo 3

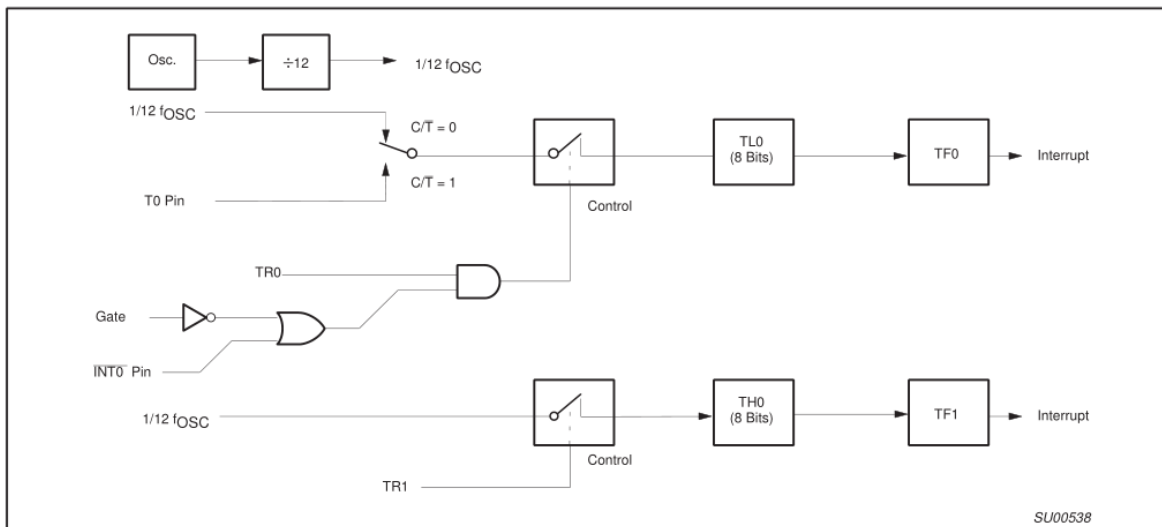
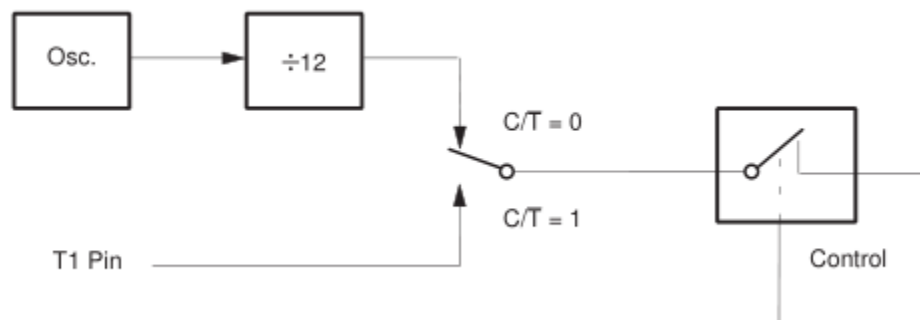
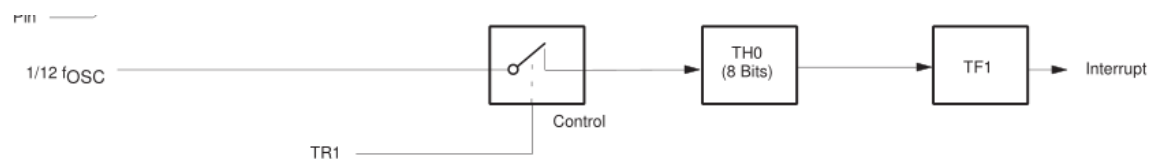


Figure 10. Timer/Counter 0 Mode 3: Two 8-Bit Counters

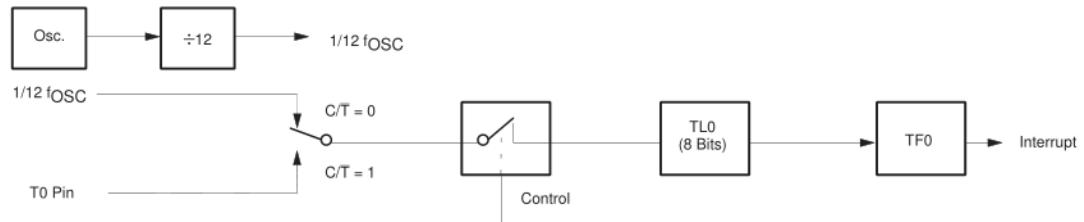


En este modo, nuevamente observamos que todos los componentes anteriores a la llave de control en la parte superior son iguales que en los anteriores, es decir que si se tiene la llave hacia arriba ($C/T=0$) se tendrá una entrada con frecuencia fija, midiendo tiempo (timer), y si la llave está hacia abajo ($C/T=1$) se tendrá una entrada respecto al pin T1, midiendo la cantidad de veces que sucede un evento (counter).



La diferencia en este caso, es que en la parte inferior se encuentra otro timer, dividiendo el original en 2. Se toman TLO y TH0 como dos counters diferentes, donde se tiene el timer superior (Timer 0) utilizando TLO y el inferior (Timer 1) utilizando TH0.

TH0 queda bloqueado en una función de timer ya que su entrada viene del oscilador con una determinada frecuencia, con TH0 contando los pulsos. Una vez que TH0 se desborda, setea el bit de TF1, contando tiempo.



El timer 0 funciona igual que en el modo 1, pero utilizando sólo los 8 bits de TL0 y seteando TF0 cuando éste desborda.

Este modo se utiliza en aplicaciones que requieran otro timer de 8 bits.