

Counter – Manual de Usuario

Índice de contenido

1.Características.....	2
2.Descripción general.....	2
3.Descripción funcional.....	3
4.Estructura interna.....	3
4.1.Bloque de Control.....	3
4.2.Bloque Constante.....	3
4.3.Bloque Counter.....	3
4.4.Bloque Control de Interrupciones.....	4
5.Programación	4
5.1.Constante a decrementar.....	4
5.2.Palabra de control.....	4
5.2.1.Bit 7, habilitación de interrupciones.....	4
5.2.2.Bit 6, configuración de trigger.....	5
5.2.3.Bit 5, RESET por software.....	5
6.Descripción de pines e integración.....	5
6.1.Pines.....	5
6.2.Integración a un sistema con microprocesador T80.....	6
7.Timing.....	6
7.1.Ciclos de Lectura.....	6
7.2.Ciclos de Escritura.....	6
8.Funcionamiento durante Interrupciones.....	7
8.1.Solicitud de interrupción.....	7
8.2.Reconocimiento de interrupción.....	7

1. Características

- Counter compatible con el microprocesador T80.
- Valor a decrementar configurable y automáticamente recargable cuando la cuenta llega a cero.
- Permite la lectura de la cuenta en curso.
- Elección de flanco de subida o bajada de trigger.
- Posee RESET por software.
- Posibilidad de manejo mediante polling, interrupciones en modo 1 o interrupciones en modo 2 utilizando un controlador de interrupciones.

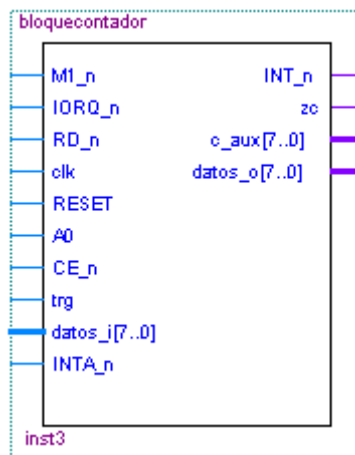


Figura 1: Bloque Counter

2. Descripción general

Este contador de eventos cuenta flancos en su entrada trg y puede ser conectado con un microprocesador T80 de modo que este puede configurarlo y utilizarlo como un periférico más.

La salida ZC da un pulso a '1' lógico cuando la cuenta expira permitiendo la conexión en cascada con otro bloque counter.

El sistema ofrece la posibilidad de configurarse para responder a flancos de subida o bajada en la entrada trg.

En caso de reconfigurar alguna característica del sistema, los cambios tendrán validez luego de expirar la cuenta en curso o, en caso de desearlo, puede realizarse un RESET por software cuando se realiza la reconfiguración y de este modo la actualización es instantánea.

El counter permite su utilización mediante polling chequeando el estado de la bandera ZC, cuando esta se encuentra en '1' lógico quiere decir que la cuenta expiró. La lectura del estado de esta bandera, borra la misma automáticamente sin necesidad de tener la constancia de hacerlo. El counter también permite su utilización mediante interrupciones, gracias a su salida Int_n la cual toma valor '0' lógico cuando la cuenta expira (y se configura para habilitar las interrupciones) para luego borrarse automáticamente cuando se activa la entrada Inta_n.

3. Descripción funcional

Una vez conectado, el counter puede configurarse para contar flancos en su entrada trg. Puede elegirse el tipo de flanco al cual responda el counter y el valor de la cuenta puede ser consultada en cualquier momento. También se tiene la posibilidad de chequear mediante polling si se cumplió una cantidad de eventos a contar o que el counter produzca una interrupción cuando esto suceda. Tanto el conteo a medir como el resto de las características mencionadas pueden modificarse en cualquier momento, teniendo validez o bien cuando la medición en curso culmine, o bien inmediatamente (dependiendo del valor elegido para el bit RESET de software en la palabra de control).

4. Estructura interna

La estructura interna del counter puede descomponerse en 4 bloques: control, constante, counter y control de interrupciones.

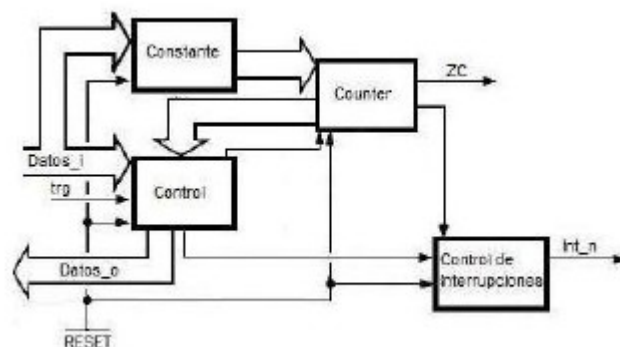


Figura 2: Estructura interna del Counter

4.1. Bloque de Control

Es el encargado de manejar los bloques Counter y Control de Interrupciones. Éste reconoce tanto los ciclos de escritura de la palabra de control (la cual es interna), de lectura de la cuenta en curso y reconocimiento de interrupción. Es responsable de setear el flag ZC, de forzar los resets por software, la elección del trigger y el reconocimiento de interrupciones. Este manejo lo logra mediante señales de control con el resto de los bloques.

4.2. Bloque Constante

Es el bloque encargado de almacenar la constante a decrementar. Cuando se realiza un ciclo de escritura en la dirección adecuada, este bloque actualiza la constante.

4.3. Bloque Counter

Es el responsable de decrementar el número a contar, el cual se recarga cada vez que la cuenta expira desde el bloque constante o cuando el bloque de control lo indica mediante señales de control entre estos.

4.4. Bloque Control de Interrupciones

Este bloque maneja la señal de salida Int_n con la cual se realiza la solicitud de interrupción al procesador. Este recibe señales de control desde el bloque Control para manejar esta salida en caso de reconocimiento de interrupción o cuando las mismas están deshabilitadas.

5. Programación

El counter permite ser configurado mediante dos palabras diferentes, una palabra de control y el valor de la constante a decrementar. La escritura de estas palabras se realiza por parte del microprocesador mediante un ciclo “out” a la dirección correspondiente. Las direcciones en el espacio de entrada/salida de cada palabra requieren la decodificación por parte del usuario mediante el manejo de las entradas A0 y CE_n. La señal CE_n determina si el counter es el que debe recibir el ciclo, la señal A0 determina si se desea escribir la palabra de control o la constante a decrementar. También puede consultarse el estado de la cuenta y el flag ZC en estas mismas direcciones mediante un ciclo de lectura.

CE_n	A0	Escritura	Lectura
0	0	Constante	Flag ZC
0	1	Control	Cuenta

Tabla 1: Decodificación de elementos

5.1. Constante a decrementar

Es el valor a decrementar por parte del counter, la misma se lee internamente cuando se reinicializa la cuenta. Esta palabra puede ser escrita en cualquier momento. (La constante a cargar corresponde a la cantidad menos uno de cuentas a realizar, la misma debe ser distinta de cero).

5.2. Palabra de control

En la encargada de determinar el modo de funcionamiento del counter, se desglosa del siguiente modo:



Figura 3: Registro de control

5.2.1. Bit 7, habilitación de interrupciones

Con este bit seteado en '1' lógico, cada vez que la cuenta expire, la señal Int_n tomará el valor '0' lógico a modo de solicitud de interrupción por parte del counter, Int_n volverá a su valor de reposo ('1' lógico) cuando se active la entrada Inta_n. En caso contrario (bit 7 en '0' lógico) la señal Int_n permanecerá siempre en '1' lógico.

5.2.2. Bit 6, configuración de trigger

Con este bit en '1' lógico, la señal de entrada trigger responderá a flancos de subida de la misma, en caso contrario responderá a flancos de bajada.

5.2.3. Bit 5, RESET por software

Con este bit seteado en '1' lógico, cada reescritura de la palabra de control forzará la reinicialización del counter, configurándose según el resto de los bits. En caso contrario (bit seteado en '0') los nuevos valores escritos en la palabra de control tendrán efecto recién en la próxima recarga del counter al llegar la cuenta a cero. La primera vez que se escribe la palabra de control la configuración tiene validez inmediatamente, independientemente del bit de RESET por software.

6. Descripción de pines e integración

6.1. Pines

- clk (clock): reloj del sistema.
- RESET: reset hardware del sistema.
- CE_n (chip enable): activo por nivel bajo, cuando se encuentra activa reconoce los ciclos de lectura de cuenta y flag ZC, y escritura de constante y palabra de control.
- A0 (bit menos significativo del bus de direcciones): selecciona cuál es el registro interno accedido en ciclos de lectura y escritura.
- trg (trigger): entrada al contador de flancos (para el correcto funcionamiento del sistema la señal trigger debe poseer una frecuencia no mayor a la mitad de la frecuencia del reloj del sistema).
- datos_i[7..0]: bus para recibir datos a ser escritos tanto en el registro constante como en el registro de control.
- datos_o[7..0]: bus para mostrar la cuenta en curso o el flag ZC del counter.
- M1_n (machine cycle one): activo por nivel bajo, utilizado para diferenciar ciclos de escritura en I/O de ciclos de reconocimiento de interrupción del T80. Cuando el T80 realiza un ciclo de reconocimiento de interrupción las señales M1_n y IORQ_n toman el valor lógico 0. Ya que el counter no cuenta con la señal WR_n, si no fuese por M1_n, éste no podría diferenciar entre ciclos de escritura y ciclos de reconocimientos de interrupción del T80.
- IORQ_n (input/output request): activo por nivel bajo, utilizado para los ciclos de lectura y escritura (junto con CE_n, A0 y RD_n).
- RD_n (read): activo por nivel bajo, utilizado para realizar ciclos de lectura y escritura al counter (junto con IORQ_n, CE_n y A0).
- ZC (zero counter): activo por nivel alto, produce un pulso en '1' cuando la cuenta expira.
- Int_n (interrupt request): activa por nivel bajo, responsable de la solicitud de interrupción cuando se encuentra habilitada la misma.

- Inta_n (interrupt acknowledge): activa por nivel bajo, indica al counter que la interrupción fue atendida.

6.2. Integración a un sistema con microprocesador T80

El counter está diseñado para ser compatible con el microprocesador T80 por lo cual la integración a un sistema con él es simple y se describe a continuación.

- Las señales de control M1_n, IORQ_n y RD_n se conectan directamente a las señales de igual nombre del T80.
- En caso de no usar controlador de interrupciones, la señal Int_n también se conecta a la del mismo nombre en el microprocesador.
- El manejo de Inta_n depende del modo de interrupciones usado en el T80. En modo 1 puede generarse a partir de IORQ_n y M1_n o con un pulso de selección de dispositivo I/O. En modo 2 esta señal debe ser manejada por el controlador de interrupciones.
- A la entrada datos_i[7..0] se conecta la salida datos_o[7..0] del procesador T80. La salida datos_o[7..] debe hacerse llegar a la entrada datos_i[7..0] del T80 habilitándola en forma adecuada con las señales de decodificación.
- Tanto clk como RESET son las correspondientes al sistema completo.
- A0 y CE_n corresponden a la decodificación en el espacio de entrada/salida que se desee.
- No hace falta conectar el pin ZC ya que la bandera ZC puede leerse mediante polling en la dirección adecuada y en todo momento.

7. Timing

7.1. Ciclos de Lectura

En la figura 4 se muestra el ciclo de lectura del contenido de la cuenta, en la misma se observan las señales de control CE_n, RD_n e IORQ_n en nivel bajo y M1_n en nivel alto.

7.2. Ciclos de Escritura

En la figura 5 se muestra un ciclo de escritura tanto de una palabra de control como de una nueva constante, en la misma se observan las señales de control CE_n e IORQ_n en nivel bajo y RD_n junto con M1_n en nivel alto.

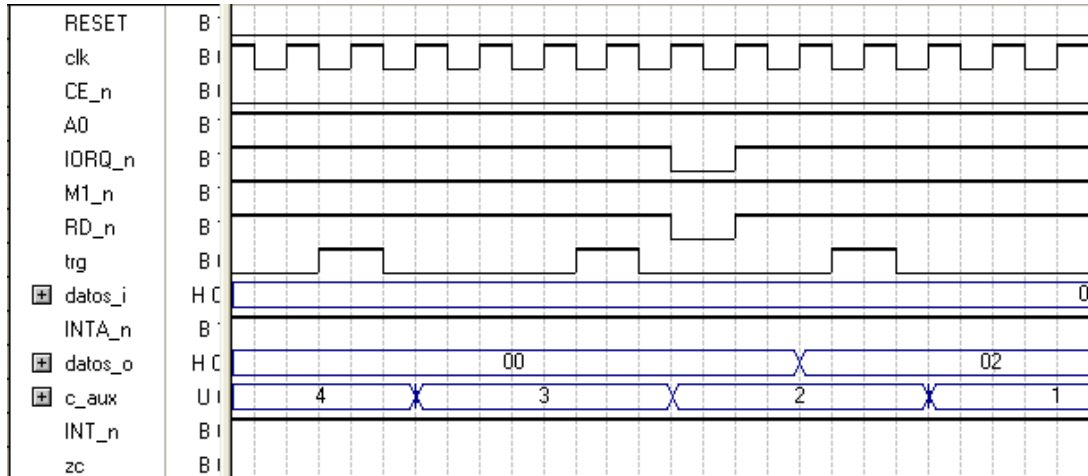


Figura 4: Ciclo de lectura del estado de la cuenta, la solicitud se realiza con IORQ_n, RD_n en nivel bajo y M1_n en nivel alto. En el flanco siguiente se detectan estos valores de las señales y se escribe en datos_o el valor de la cuenta en curso.

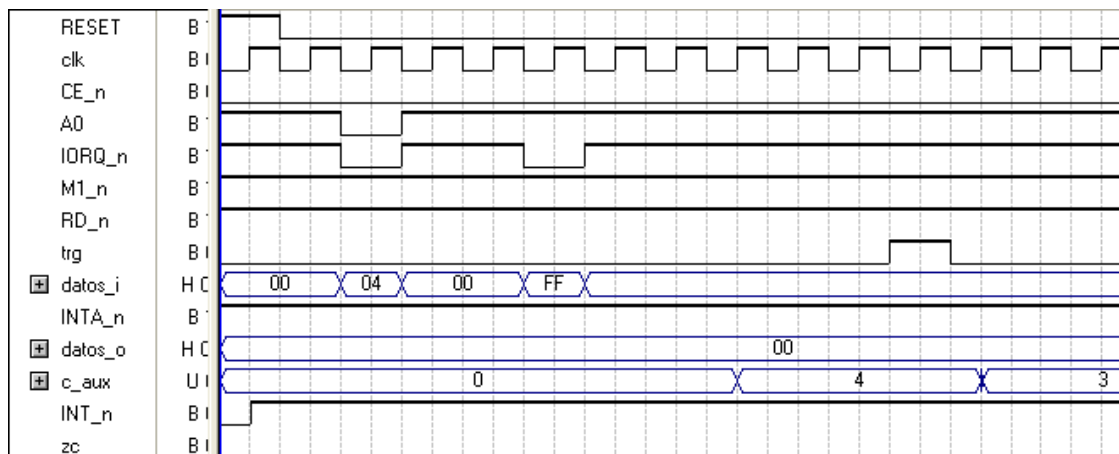


Figura 5: Ciclo de escritura de la palabra de control, en la misma la señal IORQ_n se encuentra en nivel bajo mientras que las señales RD_n y M1_n se encuentran en nivel alto. En el flanco de subida del reloj donde ocurre esta combinación de señales se escribe la palabra.

8. Funcionamiento durante Interrupciones

8.1. Solicitud de interrupción

En la figura 6 se muestra una solicitud de interrupción del counter cuando la cuenta expira, la señal Int_n toma el valor '0' lógico después de un flanco de reloj.

8.2. Reconocimiento de interrupción

En la figura 7 se muestra un ciclo de reconocimiento de la interrupción del counter mediante la señal *Inta_n* en nivel bajo, luego de esto la señal *Int_n* toma el valor '1' lógico después del flanco de reloj. Notar que en este caso *Inta_n* se activa en un ciclo de reconocimiento de interrupción del T80 por lo que *IORQ_n* y *M1_n* también se encuentran en nivel bajo,

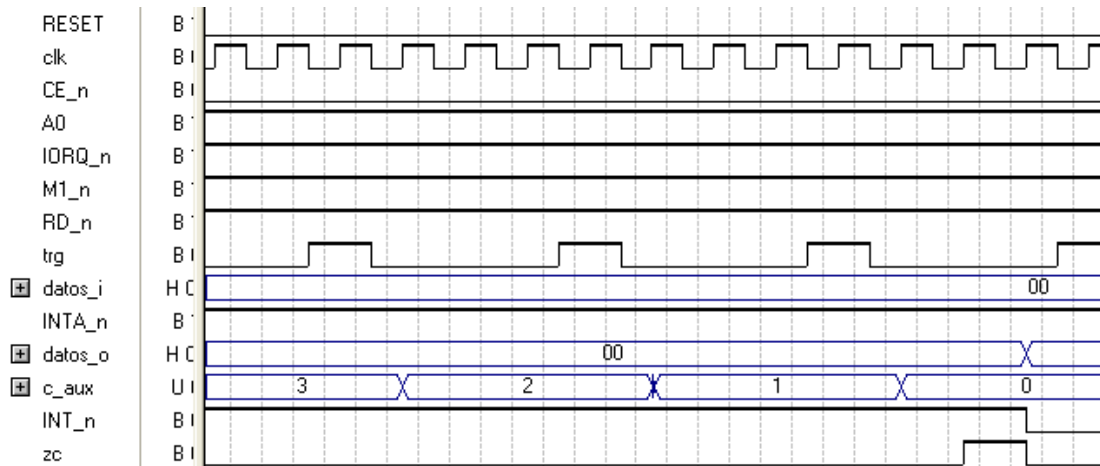


Figura 6: Solicitud de interrupción por parte del counter: cuando se detecta un flanco en la entrada *trg*, al siguiente flanco de subida del reloj del sistema la cuenta es decrementada. En cada flanco de reloj se chequea si la cuenta ha llegado a cero. Una vez que esto ocurre, la señal *Int_n* toma el valor lógico '0' si las interrupciones están habilitadas (se observa también un pulso de un ciclo de reloj de la señal *ZC* en el valor '1' lógico).

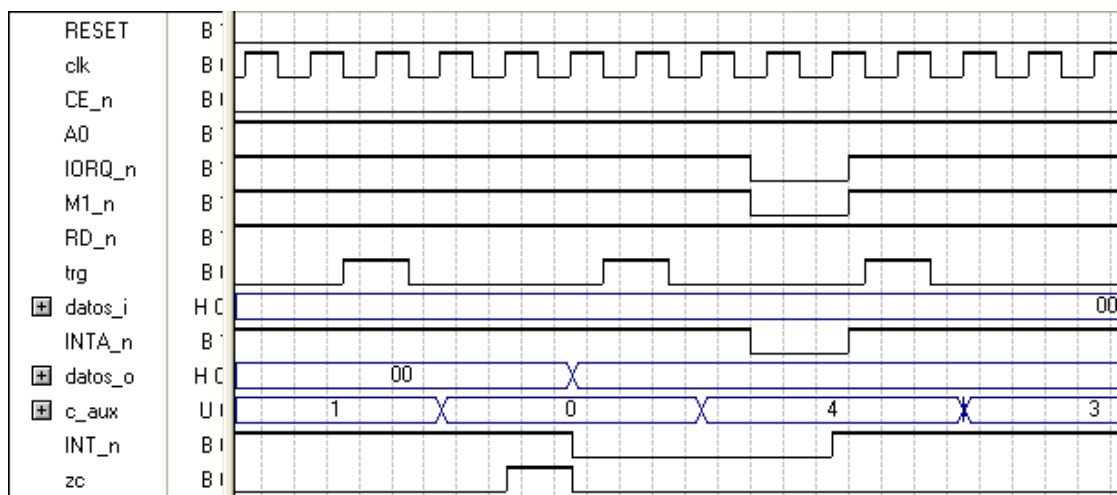


Figura 7: Reconocimiento de interrupción por parte del counter: la señal *Inta_n* toma el valor lógico 0,. La detección se realiza en el flanco de subida del reloj del sistema y cuando ocurre la señal *Int_n* vuelve a su valor de reposo. (Se observa también que la cuenta sigue decrementándose con flancos de *trg*).

Revisiones

Fecha	Versión	Comentarios
11/07/10	0	Santiago Martínez, Federico Bliman y Matías Bakalián Proyecto de curso Diseño Lógico 2010
20/05/11	1	Andrés Touya Cambio de CS por A0
28/05/12	2	Andrés Touya Se agrega Inta_n como señal de entrada