

TEMA 4. FAMILIA TMS320C3x: PERIFÉRICOS INTERNOS.

1. Introducción.
 2. Entradas-salidas digitales: XF0, XF1.
 3. Temporizadores.
 - 3.1 Registros de control y configuración.
 - 3.2 Estructura interna y configuración del periférico temporizador.
 4. Puertos serie síncronos.
 - 4.1 Registros de control y configuración.
 - 4.2 Estructura interna y configuración del periférico puerto serie.
 - 4.3 Modos de operación: Diagramas de tiempo.
 5. Controlador DMA.
 - 5.1 Registros de control y configuración.
 - 5.2 Modos de operación: Sincronización de los eventos DMA.
-

1. INTRODUCCIÓN

Se estudian en este tema los distintos periféricos internos de que dispone la familia TMS320C3x de Texas Instruments. Obsérvese que, cuando se habla de periféricos internos, nos referimos a periféricos que existen integrados en la misma pastilla de silicio de la CPU. El acceso a este tipo de periféricos no precisa de los buses externos ni de los registros de control de los mismos (entiéndase generación de estados de espera, etc.). Los periféricos funcionan de forma autónoma e independiente de la CPU del DSP. Cada periférico dispone de un cierto número de registros que están ubicados en el mapa de memoria del DSP –similar al MC68HC11–. De esta forma, cuando el usuario desea configurar y utilizar alguno de estos periféricos sólo tiene que escribir o leer en los registros asociados al periférico en cuestión.

Complementos de Sistemas Electrónicos Digitales

Por otro lado, el fabricante le asigna a estos periféricos, para su funcionamiento, una serie de *pines* del circuito integrado que, como en el caso del MC68HC11, tienen una doble funcionalidad (pueden funcionar como líneas de entrada o salida de propósito general o como líneas asociadas periférico). Por ejemplo, uno de los periféricos internos es un puerto serie al que el fabricante asigna una línea, denominada DX, para la transmisión del bit. Estas líneas, inicialmente y tras un reset, funcionan como líneas de entrada de propósito general y no como líneas asociadas al periférico en cuestión. En el caso del periférico puerto serie, la línea DX, inicialmente y tras un reset, no funciona como línea de transmisión del dato sino como entrada de propósito general.

Los diferentes periféricos internos de que dispone la familia TMS320C3x son:

- ✓ Puertos de entrada y salida digitales. Dos puertos *de un bit* cada uno, XF0 y XF1.
- ✓ Periféricos *Timers* o temporizadores de 32 bits: *Timer0* y *Timer1*.
- ✓ Periféricos *Puertos Serie Síncronos*.
- ✓ Periférico controlador de eventos o *DMA*.

2. ENTRADAS-SALIDAS DIGITALES: XF0, XF1

En la Figura 15.12 (tema 15) se mostró el registro (IOF) interno que configura el modo de funcionamiento de los puertos de entrada y salida digital y en la Tabla 15-5 se describió la funcionalidad de cada uno de los bits de dicho registro.

Cada puerto digital puede configurarse, por software, para ser empleado como entrada o salida. Si se emplea el puerto como entrada, el dato digital se puede leer de los bits 3 (*XF0*) y 7 (*XF1*) del registro IOF mientras que si se emplea como salida, el dato digital de salida se debe escribir por software en los bits 2 (*XF0*) y 6 (*XF1*) del registro IOF.

3. TEMPORIZADORES

Un temporizador es un periférico que se basa en un registro contador que, siempre que se encuentre habilitado, se va incrementando con los pulsos de alguna señal (reloj del temporizador) hasta que alcanza el valor de otro registro, denominado registro periodo. Cuando el registro contador alcanza al registro periodo el periférico temporizador activa una línea de salida, para indicar fin de un proceso de cuenta, e inicializa el registro contador a cero (comienza un nuevo proceso de cuenta).

La familia TMS320C3x de Texas Instruments dispone de **dos temporizadores internos**. Cada temporizador tiene un registro contador y un registro periodo (indicativo del fin de cuenta), ambos de 32 bits. Además de los dos registros anteriores el temporizador dispone de un registro de control y configuración, que le sirve al usuario para programar el modo de funcionamiento del temporizador.

El fabricante le asigna a cada periférico temporizador un pin, denominado *TCLK*, del DSP. La señal de reloj, que emplean los temporizadores del DSP para incrementar su registro contador, es configurable por el usuario y puede ser interna (obtenida a partir del ciclo máquina del sistema, $H1$ ¹) o externa (la fuente de reloj se genera externamente al chip y se le pasa al temporizador mediante el pin *TCLK*). El pin *TCLK* del DSP tiene varias funciones alternativas (el usuario debe programar la función del pin): Puede servir como señal de reloj de entrada al periférico temporizador, como señal de salida del periférico temporizador indicando el final de un proceso de cuenta o, finalmente, se puede usar como línea de entrada o salida de propósito general como *XF0* ó *XF1*.

En la figura 4.1 se observa el esquema básico de cada periférico temporizador del DSP.

¹ Si el cristal es de $40MHz$ (el periodo de reloj es de $25ns$), el ciclo máquina $H1$ será de $50ns$.

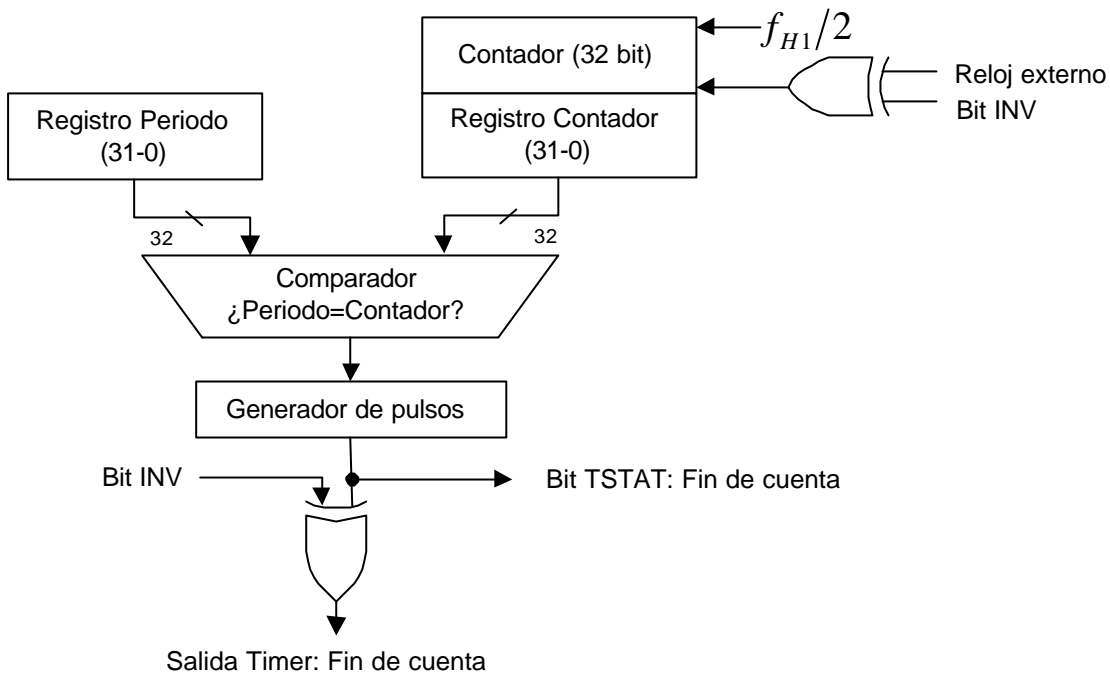


Figura 4.1. Diagrama de bloques de los periféricos temporizadores.

El temporizador siempre genera, como señal de fin de cuenta, un evento en la bandera TSAT –bit 11 del registro de control y configuración–. Se pueden configurar dos tipos de eventos o modos de generación de la señal de fin de cuenta del periférico temporizador, *modo pulso* o *modo reloj*.

- **Modo pulso.** Cada vez que el registro contador alcanza al registro periodo se genera un pulso de activación de la señal TSTAT. La señal TSTAT está normalmente a cero y cuando se igualan los registros periodo y contador se pone a uno durante un tiempo igual al ciclo máquina (H1). En la figura 4.4 se muestra la forma de onda que tendría la señal TSTAT. En el caso particular en el que el usuario programe en el registro periodo el valor cero, la salida del periférico temporizador será un valor fijo (TSTAT = 1).
- **Modo reloj.** En este caso, cada vez que los registros periodo y contador del periférico sean iguales, cambia el valor del bit TSTAT. El tiempo de activación de la señal TSTAT coincide con el valor del registro periodo que coincide, a su vez, con el tiempo de desactivación de la señal. La forma de onda de TSTAT es, en este caso, análoga a la de

Complementos de Sistemas Electrónicos Digitales

un reloj. En la figura 4.3 se muestra la forma de onda que tiene la señal TSTAT. En el caso particular de que un usuario escriba en el registro periodo el valor cero, el periférico emplea siempre la fuente de reloj interna y la frecuencia de la señal TSAT es siempre $f_{H1}/2$.

En la figura 4.6 se muestran diferentes ejemplos de generación de la señal TSTAT (la fuente de reloj del registro contador se supone interna).

Cada vez que el registro contador pasa a valer cero, el periférico temporizador realiza una petición de interrupción a la CPU. Esta petición de interrupción puede generar un evento interrupción asociado a la CPU (ruptura de la secuencia normal de ejecución del programa y ejecución de la rutina de servicio de la interrupción asociada al periférico) y puede servir para la sincronización de eventos DMA (se verá mas adelante).

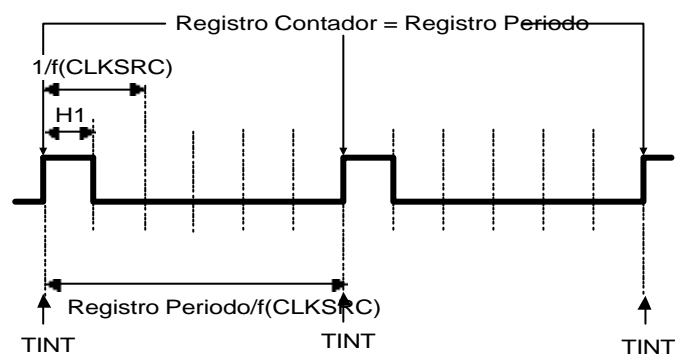


Figura 4.2. Generación del pulso de fin de cuenta en el periférico temporizador, mediante el modo pulso.

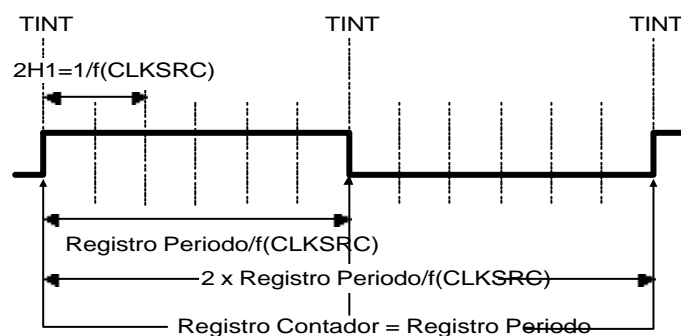


Figura 4.3. Generación del pulso de fin de cuenta en el periférico temporizador, mediante el modo reloj.

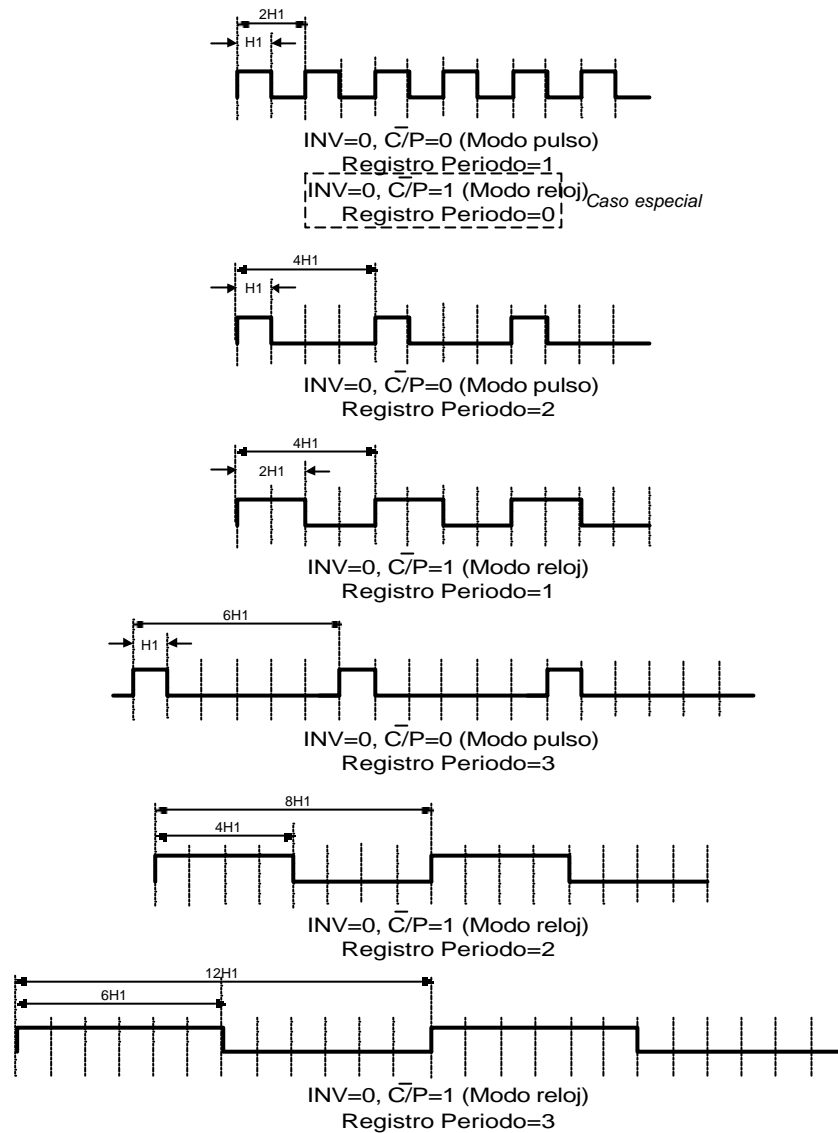


Figura 4.4. Ejemplos de generación de la señal de fin de cuenta (*TSTAT*) del periférico interno Temporizador suponiendo que el registro contador se incrementa a partir del reloj interno ($f_{H1}/2$).

3.1 Registros de control y configuración.

Cada periférico temporizador del DSP posee tres registros de 32 bits ubicados en una zona determinada del mapa de memoria del DSP, figura 4.5. Estos registros son:

- **Registro global de control y configuración del periférico.** Ubicado en la posición 808020H (temporizador 0) y en la posición 808030H (temporizador 1). Determina el modo de operación, monitoriza el estado y controla las funciones del pin de entrada/salida *-TCLK-* del periférico.

Complementos de Sistemas Electrónicos Digitales

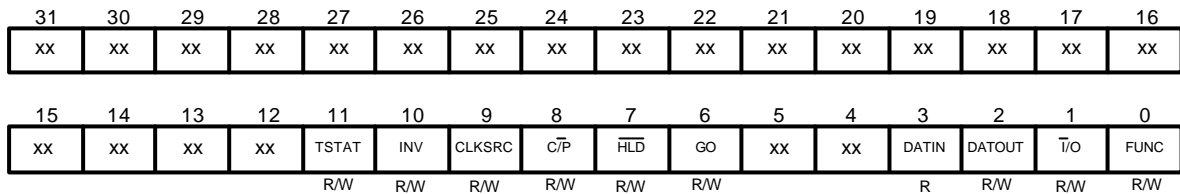
- **Registro contador.** Ubicado en la posición 808024H (temporizador 0) y en la posición 808034H (temporizador 1). Registro contador ascendente. Se incrementa, a la mitad de la frecuencia del ciclo máquina o a la frecuencia programada como reloj externo, hasta alcanzar el valor del registro periodo instante en el que se reinicializa a cero.
- **Registro periodo.** Ubicado en la posición 808028H (temporizador 0) y en la posición 808038H (temporizador 1). Determina el valor de fin de cuenta del registro contador.

En la figura 4.5 se muestra la posición en el mapa de memoria de estos registros. En la figura 4.6 se muestra el registro global de control del periférico y en la tabla 4-1 se muestran las funciones de cada bit de dicho registro.

Dirección		Descripción
<i>Timer0</i>	<i>Timer1</i>	
808020h	808030h	Registro Global de Control
808021h	808031h	Reservado
808022h	808032h	Reservado
808023h	808033h	Reservado
808024h	808034h	Registro Contador
808025h	808035h	Reservado
808026h	808036h	Reservado
808027h	808037h	Reservado
808028h	808038h	Registro Periodo
808029h	808039h	Reservado
80802Ah	80803Ah	Reservado
80802Bh	80803Bh	Reservado
80802Ch	80803Ch	Reservado
80802Dh	80803Dh	Reservado
80802Eh	80803Eh	Reservado
80802Fh	80803Fh	Reservado

Figura 4.5. Registros de los periféricos temporizadores (*Timer0* y *Timer1*) de los DSPs de la familia TMS320C3x.

Complementos de Sistemas Electrónicos Digitales



xx = Bit Reservado
R = Bit accesible en lectura
W = Bit accesible en escritura

Figura 4.6. Registro Global de configuración y control de los periféricos temporizadores internos del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0	FUNC	0	Controla el modo de funcionamiento de TCLK. Si $FUNC=0$, TCLK es una línea de entrada o salida de propósito general, independiente del periférico. Si $FUNC=1$, se emplea el pin TCLK como línea del periférico.
1	\bar{T}/O	0	Si $FUNC=0$ entonces TCLK es una línea de entrada o salida de propósito general. En ese caso, si $\bar{T}/O=0$, TCLK es una línea de entrada de propósito general mientras que si $\bar{T}/O=1$, TCLK es una línea de salida de propósito general.
2	DATOUT	0	El usuario impone, por software, el valor en el pin TCLK, cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.
3	DATIN	X	El usuario lee, por software, el valor en el pin TCLK, cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
5-4	Reservado	0-0	Se lee siempre un cero.
6	GO	0	Reinicializa el registro contador del periférico. Cuando $GO=1$ y el periférico no se encuentra en estado de paro el contador se pone a cero. El siguiente flanco de subida del reloj empleado como entrada al periférico sirve para incrementar al contador y para poner a cero el bit GO . $GO=0$ no tiene efectos sobre el estado o funcionamiento del periférico.
7	\overline{HLD}	0	Señal de mantenimiento en paro del contador. Cuando $\overline{HLD}=0$, se impide el incremento del registro contador que mantiene el valor que tenía antes de que se activase \overline{HLD} . Si el pin TCLK es impuesto por el periférico temporizador, mantiene también su valor constante a partir de la activación de la señal de paro. El divisor interno por 2 de H1 se para. Cuando $\overline{HLD}=1$, el contador sigue por donde iba. Los registros del periférico se pueden leer y escribir mientras el periférico esté en paro. La señal de RESET tiene prioridad sobre \overline{HLD} .
8	C/\bar{P}	0	Control de la forma de onda de TSTAT: modo pulso o modo reloj. $C/\bar{P}=1$ selecciona el modo reloj. $C/\bar{P}=0$ selecciona el modo pulso. figura 4.4 y figura 4.5.
9	CLKSRC	0	Especifica la fuente de reloj al temporizador. $CLKSRC=1$ indica el uso del reloj interno de frecuencia $f_{H1}/2$. $CLKSRC=0$ indica el uso del reloj externo, de frecuencia programable que siempre debe ser menor que $f_{H1}/2.6$ (el reloj externo se sincroniza y latched internamente limitándose su frecuencia máxima a valores estrictamente menores de $f_{H1}/2$).
10	INV	0	Bit de control de la puerta or-exclusiva integrada. Figura 4.1.
11	TSTAT	0	Indica el estado del periférico. Este bit activa el bit de petición de interrupción a la CPU (registro IF) asociado al periférico temporizador.
31-12	Reservado	0-0	Se lee siempre un cero.

Tabla 4-1. Funciones de los bits del registro de control y configuración de los periféricos temporizadores internos del DSP.

3.2 Estructura interna y configuración del periférico temporizador.

Los diferentes modos de operación disponibles para el temporizador interno del DSP hacen referencia a las diferentes configuraciones que se pueden definir a partir de la estructura interna del propio temporizador. Los diferentes modos de operación se definen mediante las banderas CLKSRC y FUNC del registro global de control del temporizador. Existen cuatro modos de operación, figura 4.7, asociados a cada periférico temporizador del DSP. Estos modos de operación son:

- **Modo a** CLKSRC=1 y FUNC=0. Las principales características de este modo de funcionamiento son:
 - ✓ La fuente de reloj del temporizador es interna ($f_{H1}/2$).
 - ✓ La línea *TCLK* se emplea como puerto digital de entrada o salida de propósito general y no como línea asociada al temporizador. Si en el registro global de control del periférico se programa la bandera \bar{I}/O a 0, *TCLK* funciona como entrada de propósito general –su valor lo puede leer el usuario por software accediendo en lectura al bit DATIN del mismo registro–. Si, por el contrario, $\bar{I}/O = 1$, *TCLK* actúa como salida de propósito general –el usuario impone un valor lógico en la línea *TCLK* escribiendo en la bandera DATOUT del registro de control del periférico, valor que se puede leer por software accediendo en lectura al bit DATIN–.

- **Modo b**: CLKSRC=1 y FUNC=1. Las principales características de este modo de funcionamiento son:
 - ✓ La fuente de reloj del temporizador es interna ($f_{H1}/2$).
 - ✓ *TCLK* se emplea como línea de salida asociada al periférico temporizador. El bit INV del registro global de configuración y control del temporizador se puede emplear para invertir el valor de *TCLK* respecto del valor de la bandera TSTAT.

Complementos de Sistemas Electrónicos Digitales

- ✓ A través del bit DATIN el usuario puede leer por software el valor de *TCLK*.

Complementos de Sistemas Electrónicos Digitales

- **Modo c** $CLKSRC=0$ y $FUNC=0$. Las principales características de este modo de funcionamiento son:
 - ✓ El registro contador utiliza, para incrementarse, como entrada de reloj la línea $TCLK$ del circuito integrado.
 - ✓ Si $\bar{I}/O = 0$, $TCLK$ se configura como entrada al DSP. La entrada de reloj al temporizador, $TCLK$, puede modificarse empleando INV . En todo caso el usuario puede acceder al valor de $TCLK$ leyendo la bandera $DATIN$.
 - ✓ Si $\bar{I}/O = 1$, $TCLK$ se configura como salida del DSP. El usuario genera por software, escribiendo en el bit $DATOUT$ del registro de control del periférico, el valor que aparece en la línea $TCLK$ del DSP. INV no afecta, en este caso, ni a $DATOUT$ ni a $TCLK$. Obsérvese que la entrada de reloj al registro contador es una salida del DSP que genera el usuario escribiendo en la bandera $DATOUT$.

- **Modo d**: $CLKSRC=0$ y $FUNC=1$. Este caso se puede considerar como un caso particular del caso anterior, modo c, en el que se define $TCLK$ como línea de entrada al DSP. Las principales características de este modo de funcionamiento son:
 - ✓ Como fuente de reloj de entrada al temporizador se utiliza la línea $TCLK$ del circuito integrado.
 - ✓ Si la bandera INV vale 0, un flanco de subida de $TCLK$ incrementa el registro contador. Si por el contrario vale 1, el registro contador se incrementa con el flanco de bajada de $TCLK$.
 - ✓ El valor de $TCLK$ se puede ver accediendo por software en lectura al bit $DATIN$ del registro de control del periférico.

Para poder usar cualquier periférico interno del DSP, primero es necesario configurarlo. La configuración debe realizarse, según aconseja el fabricante, manteniendo inactivo el periférico. En el caso del periférico temporizador, la configuración se debe realizar siguiendo el proceso indicado en la figura 4.8: En primer lugar se debe parar el temporizador –poner los bits GO y HLD del registro global de control del temporizador a cero–, posteriormente se realiza la configuración del periférico manteniendo los bits GO y HLD a cero –se escribe en el registro de control global del temporizador la palabra de control y configuración que se desee y en el

Complementos de Sistemas Electrónicos Digitales

registro periodo se programa el tiempo de cuenta–, finalmente se *lanza* el temporizador –se ponen las banderas GO y HLD a uno–.

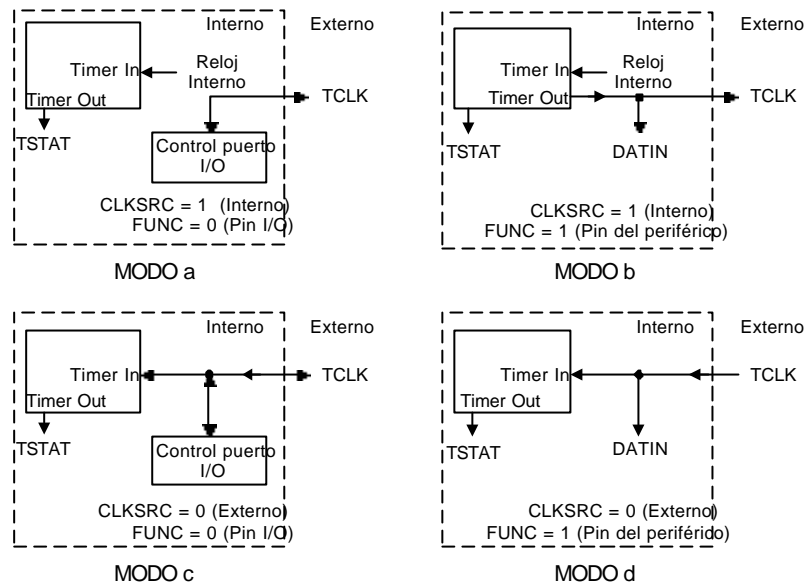


Figura 4.7. Configuraciones y modos de operación del temporizador interno del DSP.

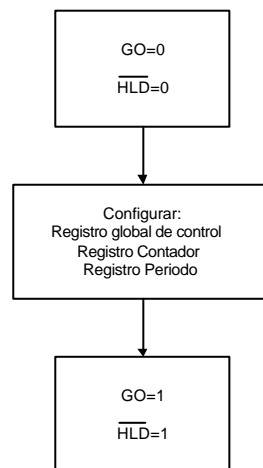


Figura 4.8. Proceso de inicialización del periférico temporizador del DSP.

4. PUERTOS SERIE SÍNCRONOS

Un puerto serie es un periférico cuya función básica es la transferencia, bit a bit, de datos digitales. Los periféricos puerto serie pueden ser de dos tipos, según el protocolo que se emplee en la transferencia de los datos, síncronos o asíncronos. La diferencia básica entre ambos protocolos estriba en la transferencia o no, por una línea diferente de la empleada en la transferencia del dato, de una señal de reloj. En el caso de los periféricos serie con protocolo síncrono el periodo de la señal de reloj sirve para indicar el periodo de validez de un bit transferido.

La familia TMS320C3x de Texas Instruments dispone de periféricos internos dedicados a la transmisión serie síncrona de datos con un ancho configurable a 8, 16, 24 ó 32 bits. Cada periférico puerto serie está compuesto de dos partes independientes entre sí, una dedicada a la recepción de datos y la otra a la transmisión. El periférico puerto serie se dice, por tanto, que es **bidireccional**: La transmisión y la recepción se pueden producir simultáneamente y son independientes entre sí.

El protocolo serie síncrono que utiliza Texas Instruments en sus DSPs maneja tres líneas para la transmisión de datos y otras tres, independientes de las anteriores, para la recepción de datos. El fabricante asigna al puerto serie, por tanto, seis líneas de entrada o salida denominadas *FSX*, *DX* y *CLKX* en la parte de transmisión y *FSR*, *DR* y *CLKR* en la parte de recepción. Como en el caso de los temporizadores estas líneas tienen dos funciones alternativas (el usuario debe configurar la funcionalidad de la línea) pudiendo servir como señales por las que se realiza la transferencia serie síncrona o como líneas de entrada o salida de propósito general (por defecto, y tras un *reset*, estas líneas se configuran como líneas de entrada al circuito integrado). La funcionalidad de estas líneas, en el caso en que sean usadas por el periférico puerto serie, es la siguiente:

- Para la transferencia de los bits se emplean las líneas *DX* (bit transmitido) y *DR* (bit recibido).

Complementos de Sistemas Electrónicos Digitales

- El reloj que se emplea en la sincronización de la transferencia de los bits circula por las líneas *CLKX* (reloj base para la transmisión de bits) y *CLKR* (reloj base para la recepción de bits).
- Finalmente, el protocolo serie síncrono de Texas Instruments incorpora una señal que indica el comienzo de la transferencia de un dato (cadena de bits). Esta señal está asociada a las líneas *FSX* (señal que indica al puerto serie el comienzo de la transmisión de un dato) y *FSR* (señal que indica al puerto serie el comienzo de la recepción de un dato).

Los procesos que se siguen en la realización de una transferencia, transmisión y recepción, serie síncrona son los siguientes:

- **Transmisión de datos:** Primero el usuario escribe por software en el registro DXR. Posteriormente, el periférico vuelca el contenido de DXR sobre XSR y, con un retraso mínimo de dos periodos del reloj base de la transmisión síncrona desde que se llenó DXR, inicia la transferencia del dato (se genera un evento FSX). Cuando se almacena DXR en XSR, el bit XRDY se pone a 1 indicando que el buffer DXR está libre para admitir otro dato. El dato se transmite bit a bit, ordenados de MSB a LSB, por la línea *DX* del DSP con un protocolo síncrono (el bit se transmite y cambia en los flancos de subida de la línea *CLKX*, la señal de reloj base del transmisor). El puerto serie se dice que tiene, por disponer de dos registros para transmitir los datos, un **doble buffer en transmisión**.
- **Recepción de datos:** Primero el periférico recibe un evento FSR que inicia la recepción del dato. El dato llega bit a bit, ordenados de MSB a LSB, por la línea *DR* del DSP (el bit se lee en los flancos de bajada de la línea *CLKR*, la señal de reloj base del receptor). Conforme los bits van llegando se van almacenando, desplazándose a izquierdas, en el registro RSR. Cuando han llegado todos los bits que conforman el ancho del dato especificado por el usuario en la configuración del periférico, el contenido del registro RSR se vuelca sobre el registro DRR y el bit RRDY se pone a 1. RRDY a 1 indica, por tanto, que el buffer DRR contiene un dato recogido por el periférico y que está listo para ser leído por software por el usuario. Si en ese momento el usuario decide no leer DRR y

Complementos de Sistemas Electrónicos Digitales

siguen llegando nuevos datos, que recoge el registro RSR, el último dato que llegó se pierde –no se vuelca el contenido de RSR en DRR si éste último registro contiene un dato no leído–. Cuando DRR es accedido en lectura RRDY se desactiva, se pone a 0, permitiendo la recogida de nuevos datos. El puerto serie se dice que tiene, por disponer de dos registros para recibir los datos, un **doble buffer en recepción**.

En la figura 4.9 se observa el esquema básico de cada periférico puerto serie del DSP.

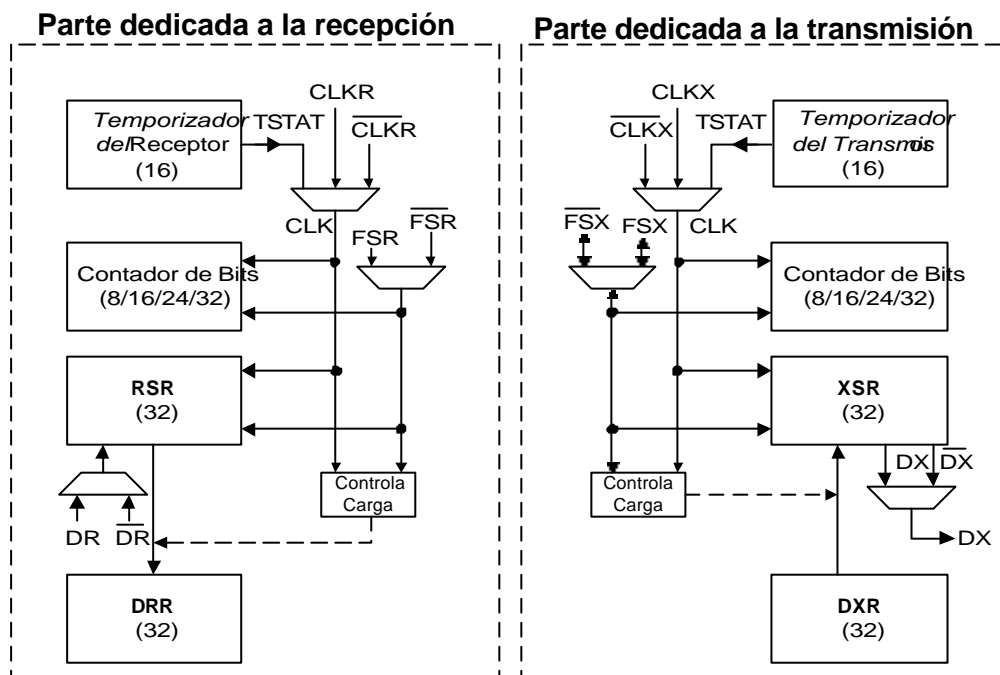


Figura 4.9. Diagrama de bloques y estructura interna del periférico puerto serie síncrono.

Cada una de las partes del periférico dispone de un temporizador de 16 bits, equivalente al visto en el apartado anterior aunque con un tamaño de los registros periodo y contador de 16 bits y con una posible fuente de reloj interna de frecuencia la mitad del ciclo máquina. La finalidad de estos temporizadores es su posible empleo en la generación del reloj que utiliza el puerto serie en la transferencia del dato (la frecuencia del reloj se determina programando el valor del registro periodo asociado al temporizador del puerto serie). Es posible definir, por tanto, como fuente de reloj para la transferencia serie síncrona la salida de fin de cuenta de los temporizadores del puerto serie aunque no es imprescindible –se puede usar un reloj externo

Complementos de Sistemas Electrónicos Digitales

para sincronizar la transferencia serie, reloj ligado a la línea *CLKX* en transmisión y a la línea *CLKR* en recepción–.

A cada parte (transmisión y recepción) del periférico puerto serie el fabricante le asigna una petición de interrupción a la CPU. Esta petición de interrupción puede generar un evento interrupción asociado a la CPU (ruptura de la secuencia normal de ejecución del programa y ejecución de la rutina de servicio de la interrupción asociada al periférico) y puede servir para la sincronización de eventos DMA (se verá mas adelante). Aunque cada parte tiene asignada una única petición de interrupción, dispone de dos fuentes o posibles causas para generar dicha petición –fuentes que se habilitan escribiendo un 1 en los bits *XTINT*, *XINT*, *RTINT* y *RINT* del registro global de control del periférico–. Las fuentes de interrupción asociadas al periférico puerto serie son:

1. Fin de cuenta del temporizador de 16 bits. El registro contador del temporizador de 16 bits alcanza al registro periodo.
2. Final del proceso de transferencia de un dato. En transmisión la petición se realiza si el registro *DXR* está vacío porque se ha volcado sobre el registro *XSR* (se pueden enviar nuevos datos) y en recepción si el registro *DRR* está lleno porque se volcó sobre él el registro *RSR*.

Como consecuencia, la interrupción asociada a la parte de transmisión del puerto serie puede generarse por quedar libre el buffer de transmisión *DXR*, al volcarse el contenido de *DXR* sobre el registro *XSR* –*XINT*=1, *XTINT*=0–, por alcanzar el registro contador del temporizador de 16 bits asociado a la parte de transmisión del periférico al registro periodo del mismo temporizador –*XINT*=0, *XTINT*=1– o por ambas causas –*XINT*=1, *XTINT*=1–. De forma análoga, la interrupción de recepción del puerto serie puede generarse por haberse llenado el buffer de recepción *DRR*, al volcarse sobre él el registro *RSR* –*RINT*=1, *RTINT*=0–, por alcanzar el registro contador del temporizador de 16 bits asociado a la parte de recepción del periférico al registro periodo del mismo temporizador –*RINT*=0, *RTINT*=1– o por ambas causas –*RINT*=1, *RTINT*=1–.

4.1 Registros de control y configuración.

Cada periférico puerto serie posee ocho registros de 32 bits, ubicados en una zona determinada del mapa de memoria del DSP, que pueden ser accedidos por software en lectura o escritura. En la figura 4.10 se muestra la posición, en el mapa de memoria, de estos registros que son:

- **Registro de control global del periférico.** Ubicado en la posición 808040H (Puerto Serie0) y en la posición 808050H (Puerto Serie1). Determina el modo de operación del periférico y monitoriza el estado del mismo. En la figura 4.11 se muestra el registro de control global del periférico y en la tabla 4-2 se muestran las funciones de cada bandera de dicho registro.
- **Registro de control de FSX, DX y CLKX.** Ubicado en la posición 808042H (Puerto Serie0) y en la posición 808052H (Puerto Serie1). Controla las funciones de las líneas de entrada/salida asociadas a la parte de transmisión del puerto serie. En la figura 4.12 se muestra el registro de control de las líneas *FSX*, *DX* y *CLKX* y en la tabla 4-3 se muestran las funciones de cada bandera o bit de dicho registro.
- **Registro de control de FSR, DR y CLKR.** Ubicado en la posición 808043H (Puerto Serie0) y en la posición 808053H (Puerto Serie1). Controla las funciones de las líneas de entrada/salida asociadas a la parte de recepción del puerto serie. En la figura 4.13 se muestra el registro de control de las líneas *FSR*, *DR* y *CLKR* y en la tabla 4-4 se muestran las funciones, de cada bandera o bit, de dicho registro.
- **Registro de control de los dos temporizadores de 16 bits.** Ubicado en la posición 808044H (Puerto Serie0) y en la posición 808054H (Puerto Serie1). Es un registro de 32 bits en el que los 12 bits menos significativos se corresponden con el registro de control del temporizador asociado a la parte de transmisión del puerto serie y los 12 bits que le siguen constituyen el registro de control del temporizador asociado a la parte de recepción del puerto serie. En la figura 4.14 se muestra el registro de control de los temporizadores del periférico y en la tabla 4-5 se muestran las funciones de cada bit de dicho registro.

Complementos de Sistemas Electrónicos Digitales

- **Registro contador de los temporizadores de 16 bits.** Ubicado en la posición 808045H (Puerto Serie0) y en la posición 808055H (Puerto Serie1). Es un registro de 32 bits constituido por los 16 bits del registro de cuenta del temporizador asociado a la parte de recepción del puerto serie (16 MSBs) y los 16 bits del registro de cuenta del temporizador asociado a la parte de transmisión del puerto serie (16 LSBs).
- **Registro periodo de los temporizadores de 16 bits.** Ubicado en la posición 808046H (Puerto Serie0) y en la posición 808056H (Puerto Serie1). Determina el valor de fin de cuenta del registro contador asociado a la recepción (16 MSBs) y a la transmisión (16 LSBs) del puerto serie.
- **Registro de datos en transmisión, DXR.** Ubicado en la posición 808048H (Puerto Serie0) y en la posición 808058H (Puerto Serie1). Registro de 32 bits. El usuario escribe en este registro los datos que desea que transmita el periférico puerto serie. El puerto serie dispone de un doble buffer en transmisión: El contenido del registro DXR se vuelca, al iniciar la transmisión del dato por la línea serie, sobre otro registro que se denomina XSR –registro de desplazamiento en transmisión–. El registro XSR es el que se encarga de generar la salida de los bits del dato por la línea *DX*.
- **Registro de datos en recepción, DRR.** Ubicado en la posición 80804CH (Puerto Serie0) y en la posición 80805CH (Puerto Serie1). Registro de 32 bits. El usuario lee en este registro los datos que ha recibido el periférico. El puerto serie dispone de un doble buffer en recepción: Dispone de un registro de desplazamiento en recepción, denominado RSR, que se encarga de capturar los bits del dato que le llega al periférico por la línea *DR*. Posteriormente, el contenido de RSR se vuelca sobre el registro DRR que es el registro desde el cual el usuario recoge, por software, el dato recibido.

Complementos de Sistemas Electrónicos Digitales

Dirección		Descripción
Puerto serie0	Puerto Serie1	
808040h	808050h	Registro Global de Control
808041h	808051h	Reservado
808042h	808052h	Registro de Control de FSX, DX, CLKX
808043h	808053h	Registro de Control de FSR, DR, CLKR
808044h	808054h	Registro de Control de los Timers (RxD, TxD)
808045h	808055h	Registro Contador de los Timers (RxD, TxD)
808046h	808056h	Registro Periodo de los Timers (RxD, TxD)
808047h	808057h	Reservado
808048h	808058h	Registro de DATOS en transmisión
808049h	808059h	Reservado
80804Ah	80805Ah	Reservado
80804Bh	80805Bh	Reservado
80804Ch	80805Ch	Registro de DATOS en recepción
80804Dh	80805Dh	Reservado
80804Eh	80805Eh	Reservado
80804Fh	80805Fh	Reservado

Figura 4.10. Registros del periférico Puerto Serie del DSP.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
xx	xx	xx	xx	RRESET	XRESET	RINT	RTINT	XINT	XTINT	RLEN		XLEN		FSRP	FSXP
				R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRP	DXP	CLKRP	CLKXP	RFSM	XFSM	RVAREN	XVAREN	RCLK SRCE	XCLK SRCE	HS	RSR FULL	XSR EMPTY	FSXOUT	XRDY	RRDY
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R

xx = Bit Reservado
R = Bit accesible en lectura
W = Bit accesible en escritura

Figura 4.11. Registro Global de configuración y control de los periféricos puertos serie internos del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0	RRDY	0	Si RRDY=1, el buffer de recepción (registro DRR) dispone de datos nuevos listos para ser leídos. El flanco de subida de esta señal, activa RINT. Luego de leer el registro DRR, RRDY se pone a 0. Existe un retraso de 3 ciclos de H1 desde que se llena el registro DRR, hasta que se activa RRDY. Si RRDY=0, el buffer de recepción (registro DRR) no dispone de datos nuevos listos para ser leídos, luego de la última lectura que se hizo.
1	XRDY	1	Si XRDY=1, el buffer de transmisión (registro DXR) ha puesto el último bit del dato que tenía en el registro de desplazamiento para la transmisión (XSR) y está preparado para recoger un nuevo dato para transmitir. Existe un retraso de 3 ciclos de H1 desde que se envía el último bit desde DXR hacia XSR, hasta que se activa XRDY. El flanco de subida de esta señal, activa XINT. Si XRDY=0, el buffer de transmisión (registro DXR) no ha puesto el último bit del dato que tenía en el registro de desplazamiento para la transmisión (XSR) y por tanto, no está preparado para recoger un nuevo dato para transmitir
2	FSXOUT	0	FSXOUT=1, configura FSX como pin de salida. FSXOUT=0, configura FSX como pin de entrada.
3	XSREMPY	0	Si XSREMPY=0, indica que el registro de desplazamiento para transmisión, XSR, está vacío. Si XSREMPY=1, XSR no está vacío.

Complementos de Sistemas Electrónicos Digitales

4	RSRFULL	0	Si RSRFULL=1, problemas en recepción (OVERRUN): Tasa de recepción muy alta. En modo recepción continua, RSRFULL=1 cuando los registros DRR y RSR están llenos. En modo recepción no continua, RSRFULL=1 cuando los registros DRR y RSR están llenos y llega un nuevo FSR (comienza a recibirse un nuevo dato). Una lectura de este bit, un reset del DSP o un reset de la parte de recepción del periférico, lo ponen a 0. Si RSRFULL=0, no hay problemas en recepción (OVERRUN).
5	HS	0	HS=1 habilita el modo <i>acuse de recibo</i> . HS=0, inhabilita el modo <i>acuse de recibo</i> .
6	XCLKSRCE	0	XCLKSRCE=1 habilita el reloj interno como fuente de reloj en transmisión. XCLKSRCE=0 habilita el reloj externo como fuente de reloj en transmisión.
7	RCLKSRCE	0	RCLKSRCE=1 habilita el reloj interno como fuente de reloj en recepción. RCLKSRCE=0 habilita el reloj externo como fuente de reloj en recepción.
8	XVAREN	0	Especifica el tiempo de activación de los bits en transmisión, FIJO (XVAREN=0) o VARIABLE (XVAREN=1). Véase modos de transferencia. En modo fijo, FSX permanece activo al menos un ciclo de XCLK para volver a estar inactivo, antes de comenzar una transferencia de datos. En modo variable, FSX permanece activo cuando se están transmitiendo datos (en modo variable, si FSX es externo, los datos empiezan a transmitirse cuando FSX está activo y siguen transmitiéndose hasta completar la transferencia, independientemente del valor de FSX)
9	RVAREN	0	Especifica el tiempo de activación de los bits en recepción, FIJO (RVAREN=0) o VARIABLE (RVAREN=1). Véase modos de transferencia. En modo fijo, FSR permanece activo al menos un ciclo de RCLK para volver a estar inactivo, antes de comenzar una transferencia de datos. En modo variable, FSR permanece activo cuando se están recibiendo datos.
10	XFSM	0	Modo de transmisión síncrona de un conjunto de datos. Configura el modo de operación en transferencia de múltiples datos: XFSM=1. Modo continuo. El primer dato transmitido genera FSX, los demás, no. XFSM=0. Modo estándar. Cada dato transmitido genera FSX.
11	RFSM	0	Modo de recepción síncrona de un conjunto de datos. Configura el modo de operación en transferencia de múltiples datos: RFSM=1. Modo continuo. En la recepción del primer dato se debe generar FSR, en la de los demás, no hace falta para que sean recibidos. RFSM=0. Modo estándar. En la recepción de cada dato hay que generar FSR.
12	CLKXP	0	Indica la polaridad de CLKX. Si CLKXP=0, CLKX es activo a nivel alto. Si CLKXP=1, CLKX es activo a nivel bajo.
13	CLKRP	0	Indica la polaridad de CLKR. Si CLKRP=0, CLKR es activo a nivel alto. Si CLKRP=1, CLKR es activo a nivel bajo.
14	DXP	0	Indica la polaridad de DX. Si DXP=0, DX es activo a nivel alto. Si DXP=1, DX es activo a nivel bajo.
15	DRP	0	Indica la polaridad de DR. Si DRP=0, DR es activo a nivel alto. Si DRP=1, DR es activo a nivel bajo.
16	FSXP	0	Indica la polaridad de FSX. Si FSXP=0, FSX es activo a nivel alto. Si FSXP=1, FSX es activo a nivel bajo.
17	FSRP	0	Indica la polaridad de FSR. Si FSRP=0, FSR es activo a nivel alto. Si FSRP=1, FSR es activo a nivel bajo.
18-19	XLEN	00	Estos dos bits indican el ancho del dato a transmitir: 00 (8 bits), 01 (16 bits), 10 (24 bits) y 11 (32 bits). Todos los datos se encuentran en transmisión, justificados a derechas en el registro XSR.
20-21	RLEN	00	Estos dos bits indican el ancho del dato a recibir: 00 (8 bits), 01 (16 bits), 10 (24 bits) y 11 (32 bits). Todos los datos se encuentran en recepción, justificados a derechas en el registro RSR.

Complementos de Sistemas Electrónicos Digitales

22	XTINT	0	Habilitación de la interrupción del Temporizador asociado a la parte de transmisión del puerto serie. Si XTINT=1, se habilita la interrupción de fin de cuenta de dicho temporizador.
23	XINT	0	Habilitación de la interrupción de transmisión del puerto serie. Si XINT=1, se habilita la interrupción por dato transmitido por el puerto serie. La interrupción XINT generada por el periférico a la CPU, es la OR-lógica entre las dos posibles fuentes asociadas a la parte de transmisión del puerto serie.

Complementos de Sistemas Electrónicos Digitales

24	RTINT	0	Habilitación de la interrupción del Temporizador asociado a la parte de recepción del puerto serie. Si RTINT=1, se habilita la interrupción de fin de cuenta de dicho temporizador.
25	RINT	0	Habilitación de la interrupción de recepción del puerto serie. Si RINT=1, se habilita la interrupción por dato recibido por el puerto serie. La interrupción RINT generada por el periférico a la CPU, es la OR-lógica entre las dos posibles fuentes asociadas a la parte de recepción del puerto serie.
26	XRESET	0	Reset de la parte de transmisión del periférico. Si XRESET=0, la parte de transmisión del puerto serie se encuentra en estado de reset. Para que funcione la parte de transmisión del periférico, hay que poner XRESET=1. Poniendo XRESET=0, no se borra la configuración de la parte de transmisión del puerto serie, se sitúa el transmisor al comienzo de la transferencia de un conjunto de datos y se genera una interrupción por transmisión. La reconfiguración del modo de funcionamiento del periférico en transmisión, se debe hacer con XRESET=0 aunque es posible cambiar el valor de XFSM sin necesidad de que XRESET=0.
27	RRESET	0	Reset de la parte de recepción del periférico. Si RRESET=0, la parte de recepción del puerto serie se encuentra en estado de reset. Para que funcione la parte de recepción del periférico, hay que poner RRESET=1. Poniendo RRESET=0, no se borra la configuración de la parte de recepción del puerto serie y se sitúa el receptor al comienzo de la transferencia de un conjunto de datos. La reconfiguración del modo de funcionamiento del periférico en recepción, se debe hacer con RRESET=0 aunque es posible cambiar el valor de RFSM sin necesidad de que RRESET=0.
28-31	Reservado	0-0	Se lee siempre un cero.

Tabla 4-2. Funciones de los bits del registro de control y configuración de los periféricos temporizadores internos del DSP.

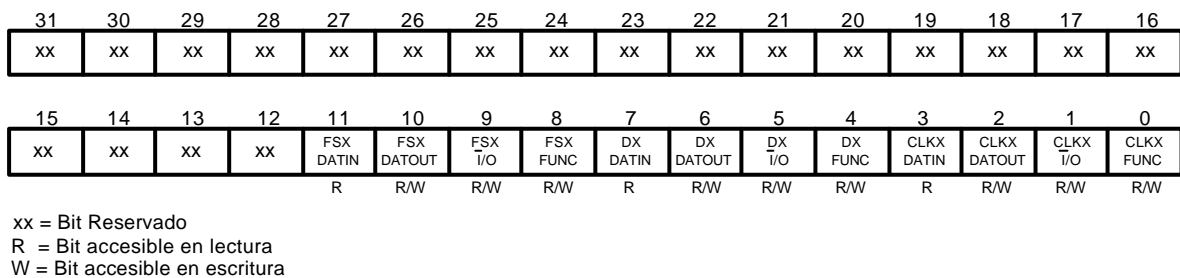


Figura 4.12. Registro de control de los pines FSX, DX y CLKX del periférico puerto serie del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0	CLKXFUNC	0	Controla el modo de funcionamiento de CLKX. Si CLKXFUNC=0, CLKX es una línea de entrada o salida de propósito general, independiente del periférico. Si CLKXFUNC=1, CLKX es una línea del periférico.
1	$CLKX\bar{I}/O$	0	Si CLKXFUNC=0, entonces CLKX es una línea de entrada o salida de propósito general. En ese caso, si CLKXI/O=0, CLKX es una línea de entrada de propósito general mientras que si CLKXI/O=1, CLKX es una línea de salida de propósito general.
2	CLKXDATOU	0	El usuario escribe por software el valor en CLKX cuando se encuentra

Complementos de Sistemas Electrónicos Digitales

	T		configurado como salida de propósito general, accediendo en escritura a este bit.
3	CLKXDATIN	x	El usuario lee por software el valor de CLKX cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
4	DXFUNC	0	Controla el modo de funcionamiento de DX. Si DXFUNC=0, DX es una línea de entrada o salida de propósito general, independiente del periférico. Si DXFUNC=1, DX es una línea del periférico.

Complementos de Sistemas Electrónicos Digitales

5	$DX\bar{I}/O$	0	Si DXFUNC=0, entonces DX es una línea de entrada o salida de propósito general. En ese caso, si DXI/O=0, DX es una línea de entrada de propósito general mientras que si DXI/O=1, DX es una línea de salida de propósito general.
6	DXDATOUT	0	El usuario escribe por software el valor en DX cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.
7	DXDATIN	x	El usuario lee por software el valor de DX cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
8	FSXFUNC	0	Controla el modo de funcionamiento de FSX. Si FSXFUNC=0, FSX es una línea de entrada o salida de propósito general, independiente del periférico. Si FSXFUNC=1, FSX es una línea del periférico.
9	$FSX\bar{I}/O$	0	Si FSXFUNC=0, entonces FSX es una línea de entrada o salida de propósito general. En ese caso, si FSXI/O=0, FSX es una línea de entrada de propósito general mientras que si FSXI/O=1, FSX es una línea de salida de propósito general.
10	FSXDATOUT	0	El usuario escribe por software el valor en FSX cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.
11	FSXDATIN	x	El usuario lee por software el valor de FSX cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
31-12	Reservado	0-0	Se lee siempre un cero.

Tabla 4-3. Funciones de los bits del registro de control de los pines FSX, DX y CLKX del puerto serie del DSP.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
xx	xx	xx	xx	FSR DATIN	FSR DATOUT	FSR I/O	FSR FUNC	DR DATIN	DR DATOUT	DR I/O	DR FUNC	CLKR DATIN	CLKR DATOUT	CLKR I/O	CLKR FUNC
				R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

xx = Bit Reservado
R = Bit accesible en lectura
W = Bit accesible en escritura

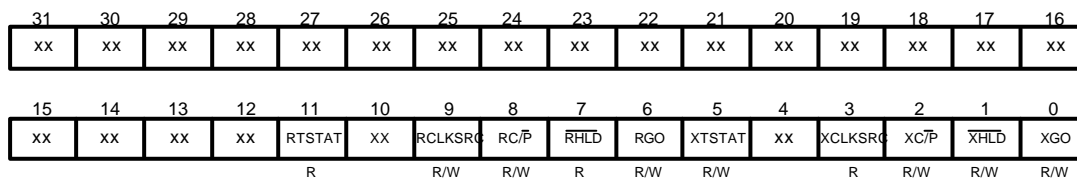
Figura 4.13. Registro de control de los pines FSR, DR y CLKR del periférico puerto serie del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0	CLKRFUNC	0	Controla el modo de funcionamiento de CLKR. Si CLKRFUNC=0, CLKR es una línea de entrada o salida de propósito general, independiente del periférico. Si CLKRFUNC=1, CLKR es una línea del periférico.
1	$CLKR\bar{I}/O$	0	Si CLKRFUNC=0, entonces CLKR es una línea de entrada o salida de propósito general. En ese caso, si CLKRI/O=0, CLKR es una línea de entrada de propósito general mientras que si CLKRI/O=1, CLKR es una línea de salida de propósito general.
2	CLKRDATOUT	0	El usuario escribe por software el valor en CLKR cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.

Complementos de Sistemas Electrónicos Digitales

3	CLKRDATIN	x	El usuario lee por software el valor de CLKR cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
4	DRFUNC	0	Controla el modo de funcionamiento de DR. Si DRFUNC=0, DR es una línea de entrada o salida de propósito general, independiente del periférico. Si DRFUNC=1, DR es una línea del periférico.
5	$DR\bar{I}/O$	0	Si DRFUNC=0, entonces DR es una línea de entrada o salida de propósito general. En ese caso, si DRI/O=0, DR es una línea de entrada de propósito general mientras que si DRI/O=1, DR es una línea de salida de propósito general.
6	DRDATOUT	0	El usuario escribe por software el valor en DR cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.
7	DRDATIN	x	El usuario lee por software el valor de DR cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
8	FSRFUNC	0	Controla el modo de funcionamiento de FSR. Si FSRFUNC=0, FSR es una línea de entrada o salida de propósito general, independiente del periférico. Si FSRFUNC=1, FSR es una línea del periférico.
9	$FSR\bar{I}/O$	0	Si FSRFUNC=0, entonces FSR es una línea de entrada o salida de propósito general. En ese caso, si FSRI/O=0, FSR es una línea de entrada de propósito general mientras que si FSRI/O=1, FSR es una línea de salida de propósito general.
10	FSRDATOUT	0	El usuario escribe por software el valor en FSR cuando se encuentra configurado como salida de propósito general, accediendo en escritura a este bit.
11	FSRDATIN	x	El usuario lee por software el valor de FSR cuando se encuentra configurado como entrada de propósito general, accediendo en lectura a este bit.
31-12	Reservado	0-0	Se lee siempre un cero.

Tabla 4-4. Funciones de los bits del registro de control de las líneas FSR, DR y CLKR del puerto serie del DSP.



xx = Bit Reservado
R = Bit accesible en lectura
W = Bit accesible en escritura

Figura 4.14. Registro de configuración y control de los temporizadores del periférico interno puerto serie del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0	XGO	0	Reinicializa los 16 LSBs del registro contador del periférico (contador del temporizador de transmisión). Cuando XGO=1 y el temporizador no se encuentra en estado de paro, el contador del temporizador de transmisión se pone a cero. El siguiente flanco de subida del reloj empleado como entrada al periférico sirve para incrementar dicho contador y para poner a cero el bit XGO. XGO=0 no tiene efectos sobre el estado o funcionamiento del periférico.
1	\overline{XHLD}	0	Señal de mantenimiento en paro del contador. Cuando XHLD=0, se impide el incremento del registro contador del temporizador de transmisión, que mantiene el valor que tenía

Complementos de Sistemas Electrónicos Digitales

			antes de que se activase XHLD. El divisor interno por 2 de $f(H1)$ se para. Cuando $XHLD=1$, el contador sigue por donde iba. Los registros del periférico se PUEDEN leer y escribir mientras el periférico esté en paro. La señal de XRESET tiene prioridad sobre XHLD.
2	XC/\bar{P}	0	Control de la forma de onda de XTSTAT: modo pulso o modo reloj. $XC/P=1$ selecciona el modo reloj, mientras que $XC/P=0$ selecciona el modo pulso. Análogo al periférico temporizador interno del DSP.
3	XCLKSRC	0	Especifica la fuente de reloj al temporizador. $XCLKSRC=1$, indica el uso del reloj interno de frecuencia $f(H1)/2$. $XCLKSRC=0$, indica el uso del reloj externo (pin CLKX) de frecuencia programable que siempre debe ser menor que $f(H1)/2.6$ (el reloj externo se sincroniza, latched, internamente limitándose la frecuencia máxima de dicho reloj a valores estrictamente menores de $f(H1)/2$).
4	Reservado	0	Se lee siempre un cero.
5	XTSTAT	0	Indica el estado del temporizador de transmisión del periférico. Este bit activa (en un flanco de subida) el correspondiente bit de petición de interrupción a la CPU.
6	RGO	0	Reinicializa los 16 MSBs del registro contador del periférico (contador del temporizador de recepción). Cuando $RGO=1$ y el temporizador no se encuentra en estado de paro, el contador del temporizador de recepción se pone a cero. El siguiente flanco de subida del reloj empleado como entrada al periférico sirve para incrementar dicho contador y para poner a cero el bit RGO. $RGO=0$ no tiene efectos sobre el estado o funcionamiento del periférico.

Complementos de Sistemas Electrónicos Digitales

7	$\overline{RHL D}$	0	Señal de mantenimiento en paro del contador. Cuando RHL D=0, se impide el incremento del registro contador del temporizador de recepción, que mantiene el valor que tenía antes de que se activase RHL D. El divisor interno por 2 de f(H1) se para. Cuando RHL D=1, el contador sigue por donde iba. Los registros del periférico se PUEDEN leer y escribir mientras el periférico esté en paro. La señal de RRESET tiene prioridad sobre RHL D.
8	RC/\overline{P}	0	Control de la forma de onda de RTSTAT: modo pulso o modo reloj. RC/P=1 selecciona el modo reloj, mientras que RC/P=0 selecciona el modo pulso. Análogo al periférico temporizador interno del DSP.
9	RCLKSRC	0	Especifica la fuente de reloj al temporizador. RCLKSRC=1, indica el uso del reloj interno de frecuencia f(H1)/2. RCLKSRC=0, indica el uso del reloj externo (pin CLKR) de frecuencia programable que siempre debe ser menor que f(H1)/2.6 (el reloj externo se sincroniza, latchea, internamente limitándose la frecuencia máxima de dicho reloj a valores estrictamente menores de f(H1)/2).
10	Reservado	0	Se lee siempre un cero.
11	RTSTAT	0	Indica el estado del temporizador de recepción del periférico. Este bit activa (en un flanco de subida) el correspondiente bit de petición de interrupción a la CPU.
31-12	Reservado	0-0	Se lee siempre un cero.

Tabla 4-5. Funciones de los bits del registro de configuración y control de los temporizadores del periférico interno puerto serie del DSP.

4.2 Estructura interna y configuración del periférico puerto serie.

El periférico puerto serie que añade el fabricante a la estructura interna del DSP es muy versátil. La estructura interna de cada parte del periférico se puede configurar de siete maneras diferentes en función de las banderas CLKXFUNC, XCLKSRCE y XCLKSRC (transmisión) y de CLKRFUNC, RCLKSRCE y RCLKSRC (recepción).

En la figura 4.15 y en la figura 4.16 se muestran las posibles estructuras internas que se pueden configurar para la parte de transmisión del periférico puerto serie síncrono. En la figura 4.15 se muestran las estructuras programables del puerto serie cuando la línea *CLKX* se usa como línea de entrada o salida de propósito general y en la figura 4.16 se muestran las estructuras programables del puerto serie cuando la línea *CLKX* se emplea como línea asociada al periférico. Las estructuras internas que se pueden configurar para la parte de recepción del periférico son análogas a las comentadas anteriormente y, por tanto, no se van a describir. Se observa que, configurando la línea de transferencia del reloj (*CLKX*) como línea de entrada o salida digital (CLKXFUNC=0) aparecen cuatro posibles estructuras internas, dependiendo de si la línea es entrada o salida ($CLKX \bar{I}/O = 0$ ó $CLKX \bar{I}/O = 1$) y de la fuente, interna o externa, de reloj empleada en la transferencia serie (XCLKSRC). Si la línea de transferencia del reloj se asocia al periférico (CLKXFUNC=1) aparecen otras tres

Complementos de Sistemas Electrónicos Digitales

posibles estructuras internas dependiendo, como en el caso anterior, de si la línea es entrada o salida y de la fuente, interna o externa, de reloj empleada en la transferencia serie.

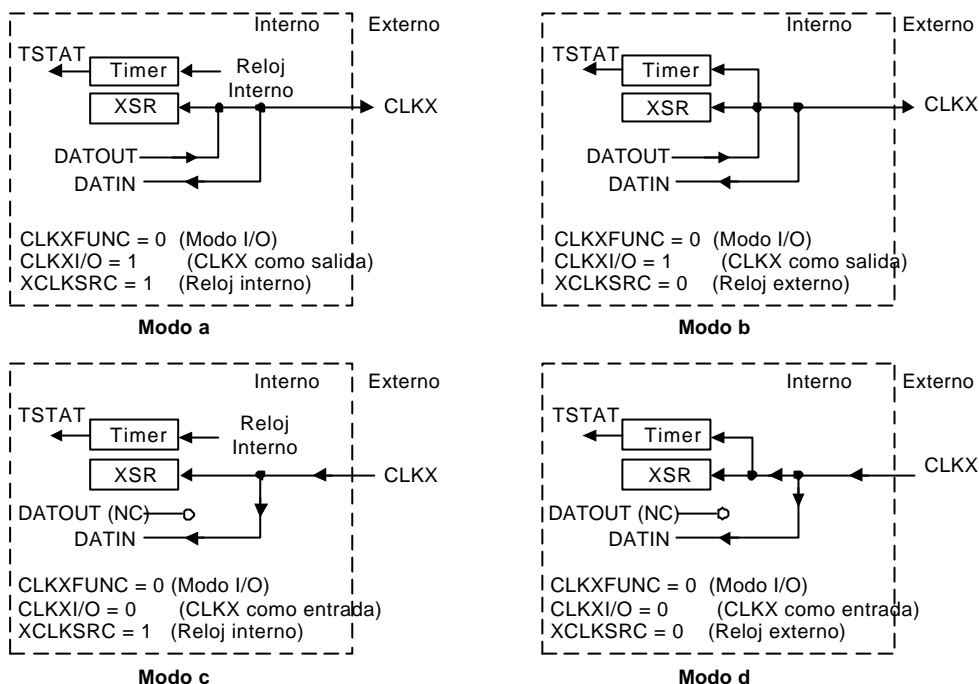


Figura 4.15. Configuración del modo de operación del puerto serie: Parte de transmisión. $CLKX$ en modo entrada o salida digital ($CLKXFUNC = 0$).

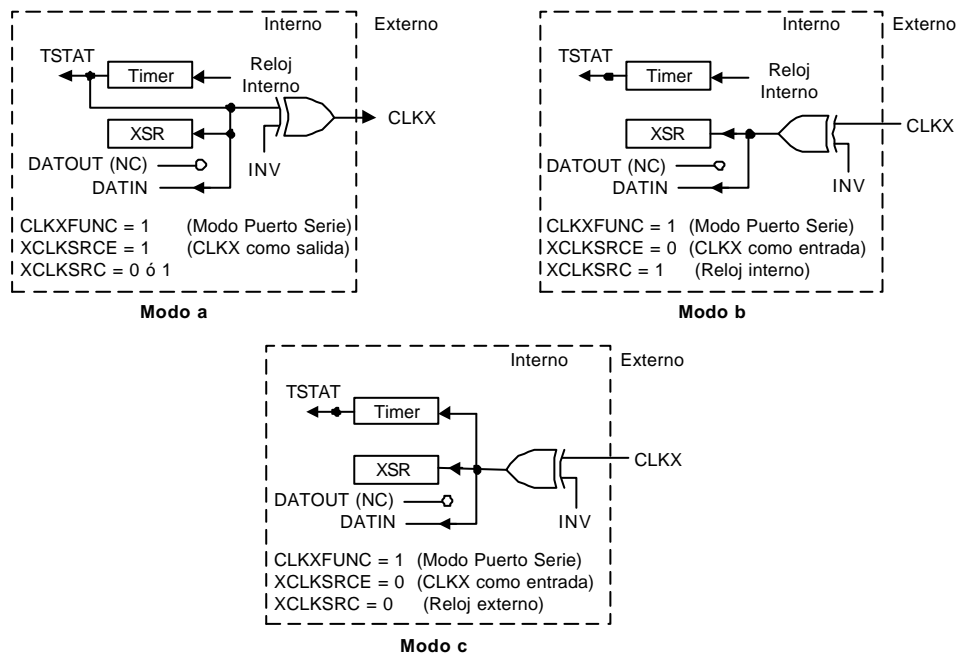


Figura 4.16. Configuración del modo de operación del puerto serie: Parte de transmisión. $CLKX$ se emplea como línea del periférico ($CLKXFUNC = 1$).

Complementos de Sistemas Electrónicos Digitales

El usuario puede, por tanto, configurar la estructura interna de cada parte del periférico puerto serie para que coincida con alguna de las siete descritas en la figura 4.15 y figura 4.16. Algunas de estas estructuras implican que el temporizador de 16 bits no se utiliza para la transferencia serie y que queda liberado para que el usuario lo emplee como otro periférico temporizador, análogo al visto en el apartado anterior, aunque sus registros internos tienen la mitad de tamaño que los periféricos temporizadores internos del DSP.

El fabricante incorpora puertos serie síncronos no sólo en los microprocesadores sino que también en muchos periféricos externos que produce. La utilidad que tiene el puerto serie interno del DSP es, por un lado, facilitar el hardware necesario para conectar periféricos externos al DSP (por ejemplo existen convertidores de analógico a digital de Texas Instruments que generan la salida digital en formato serie síncrono compatible con el de los puertos serie del DSP, esto facilita la labor del diseñador de sistemas digitales que tiene que rutar pocas líneas para conectar el periférico al DSP). Por otro lado, una de las características básicas de un DSP es su escalabilidad (son microprocesadores que deben conectarse fácilmente entre sí para trabajar en paralelo). Para cumplimentar la característica anterior, el fabricante incluye en el periférico un modo de funcionamiento especial que permite la conexión directa entre dos DSPs a través de sus puertos serie, figura 4.17. Este modo de funcionamiento es muy sencillo de configurar –basta imponer un uno en la bandera **HS** del registro global de control y configuración del periférico–. Es un modo de transferencia serie con **acuse de recibo** en el que los datos se transmiten precedidos de un bit a 1 –*leading one*– y, una vez que el transmisor ha concluido la transferencia de un dato, no procede a la transmisión de otro hasta que no reciba del receptor un bit a 0 –*acknowledge bit*–. El puerto serie del receptor genera, por tanto, el *acknowledge bit* cada vez que lee por software el registro DRR. Todo el proceso de generación de un bit de cabecera a 1 y de un bit de reconocimiento a 0 es automático (transparente al usuario). En este modo *FSX* se configura automáticamente como salida y se inhabilita el modo continuo de transferencia de datos (se verá en el apartado siguiente, donde se describe el protocolo serie síncrono y los diagramas de tiempo asociados a cada protocolo).

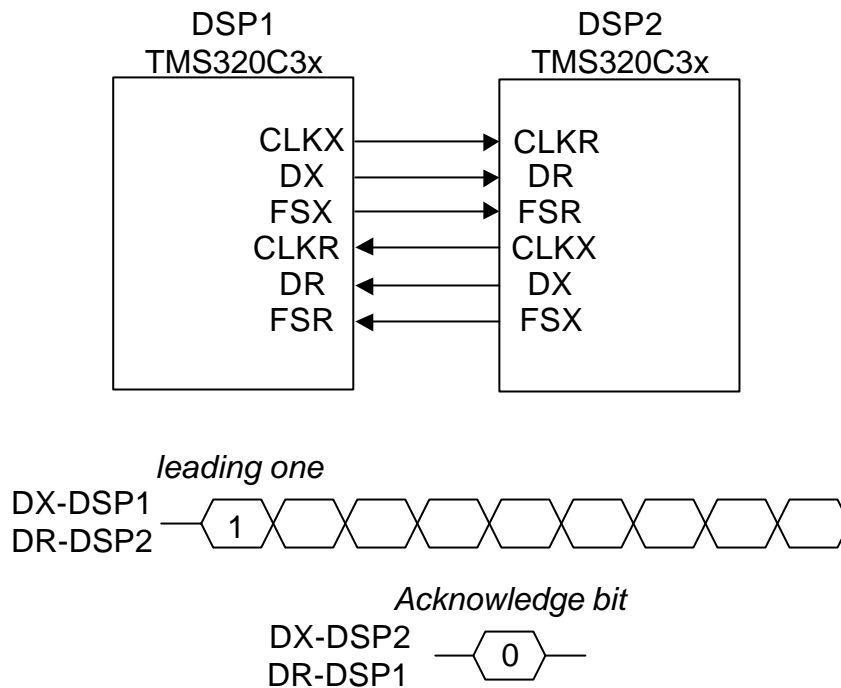


Figura 4.17. Modo Acuse de recibo: Conexionado y modo de funcionamiento.

Para poder usar el periférico puerto serie del DSP, primero es necesario configurarlo. La configuración debe realizarse, según aconseja el fabricante, manteniendo inactivo el periférico siguiendo, por tanto, el proceso indicado en la figura 4.18: En primer lugar se debe parar el periférico –poner los bits XRESET y/o RRESET (dependiendo de si se desea configurar sólo la parte de transmisión, la de recepción o ambas) del registro de control global del puerto serie a cero–, posteriormente se realiza la configuración del periférico manteniendo los bits XRESET y/o RRESET a cero –se escribe en el registro de control global del puerto serie y en los registros de control de las líneas *FSX*, *DX*, *CLKX*, *FSR*, *DR* y *CLKR* la palabra de control y configuración que se desee y, si fuese necesario, se configuran los registros ligados a los temporizadores del puerto serie (para ello habrá que escribir un cero en las banderas XGO y XHLD, si se desea utilizar el temporizador ligado a la parte de transmisión del puerto serie, ó RGO y RHL D, si se desea utilizar el temporizador ligado a la parte de recepción del puerto serie) de forma análoga a como se hace en los periféricos temporizadores–, finalmente se *lanza* la transferencia serie síncrona –se ponen las banderas XRESET y/o RRESET a uno–.

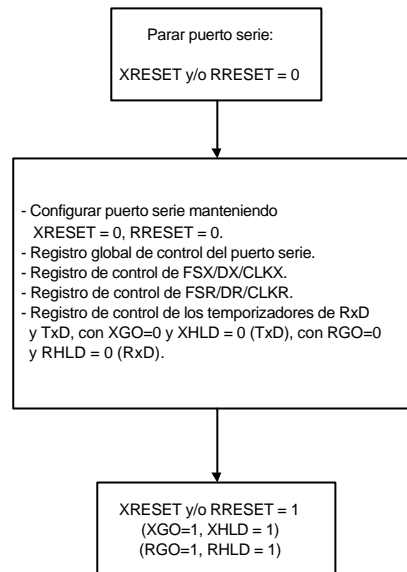


Figura 4.18. Inicialización del periférico interno puerto serie del DSP.

4.3 Modos de operación: Diagramas de tiempo.

Hasta ahora se han comentado las características básicas asociadas al periférico puerto serie interno utilizado por la familia TMS320C3x de Texas Instruments. Se ha visto que estos periféricos realizan una **transferencia bit a bit de datos**. Los bits se transfieren siempre por orden, del MSB a LSB, y como el periférico se basa en un protocolo serie síncrono, se transfieren con los flancos de una señal de reloj, *CLKX* en transmisión y *CLKR* en recepción. La transferencia de los datos comienza cuando se activan las señales *FSX* en transmisión y *FSR* en recepción. Hemos visto, además, que el ancho de los datos a transferir no es fijo sino que se puede configurar a un ancho de 8, 16, 24 ó 32 bits. Si el dato a transferir se configura con un ancho inferior a 32 bits (8, 16 ó 24) se ajusta a derechas en los registros de transferencia (el dato transferido se encuentra en la parte menos significativa de los registros *DXR*, *XSR*, *DRR* y *RSR*).

En este apartado vamos a ver los diagramas de tiempos y como evolucionan las líneas del periférico puerto serie cuando se realiza una transferencia serie síncrona. El fabricante permite que el usuario programe cuatro modos de transferencia serie, según cómo se realice la transferencia de los bits del dato y cómo se efectúa la transferencia del propio dato:

Complementos de Sistemas Electrónicos Digitales

- **Transferencia de bits.** Puede ser de dos tipos, transferencia de bits a frecuencia constante, *fixed data-rate timing operation*, o a frecuencia variable, *variable data-rate timing operation*. Prácticamente no existen diferencias entre ambos modos de operación salvo en el tiempo durante el que permanecen activos los bits del dato transferido: En el modo de transferencia a frecuencia constante todos los bits del dato permanecen activos en la línea *DX* ó *DR* el mismo tiempo (un periodo del reloj base de la transferencia serie) mientras que en el modo transferencia a frecuencia variable el primer bit transferido permanece activo menos de un periodo del reloj base de la transferencia (el resto están activos un periodo del reloj).
- **Transferencia de datos.** Puede ser, a su vez, de otros dos tipos, transferencia de datos sueltos, *modo estándar*, o transferencia continua de datos, *modo continuo*. La diferencia entre un modo de transferencia de datos y otro estriba en la necesidad (modo estándar) o no (modo continuo) de generar la señal de inicio de la transferencia serie cada vez que se desea transferir un dato. En modo continuo se genera la señal de inicio de transferencia una única vez, al comienzo de la transferencia del primer dato, a partir de ese momento se transfieren datos de forma ininterrumpida.

Para aclarar los diferentes protocolos de comunicación serie síncrona, que se pueden establecer con el periférico serie del DSP, vamos a ver diferentes figuras en las que se muestran los diagramas de tiempo que se pueden generar asociados a la evolución de las líneas del periférico puerto serie.

En la figura 4.19, figura 4.20 y figura 4.21 se muestran los diferentes diagramas de tiempo que pueden programarse en una transferencia serie síncrona, en **modo de transferencia de bits a frecuencia constante**. Obsérvese que todos los bits permanecen activos el mismo tiempo (un ciclo del reloj de transmisión o recepción) y que el ciclo de reloj, de transmisión o recepción, previo al inicio real de la transferencia de cada dato (modo estándar) o del primer dato (modo continuo) se genera la señal de inicio de transferencia: *FSX* ó *FSR*.

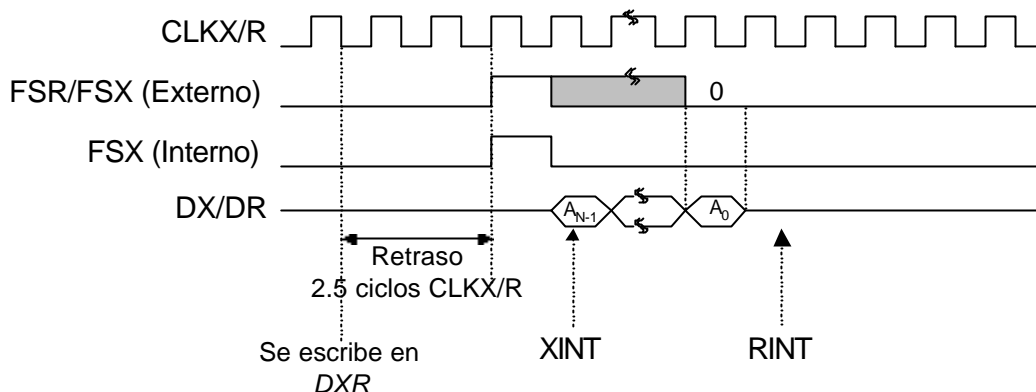


Figura 4.19. Diagrama de tiempo asociado a una transferencia serie síncrona en **modo estándar** (un único dato transferido) y **transferencia de bits a frecuencia constante** (*fixed data-rate timing operation*). Bits (R)XFSM=0 y (R)XVAREN=0.

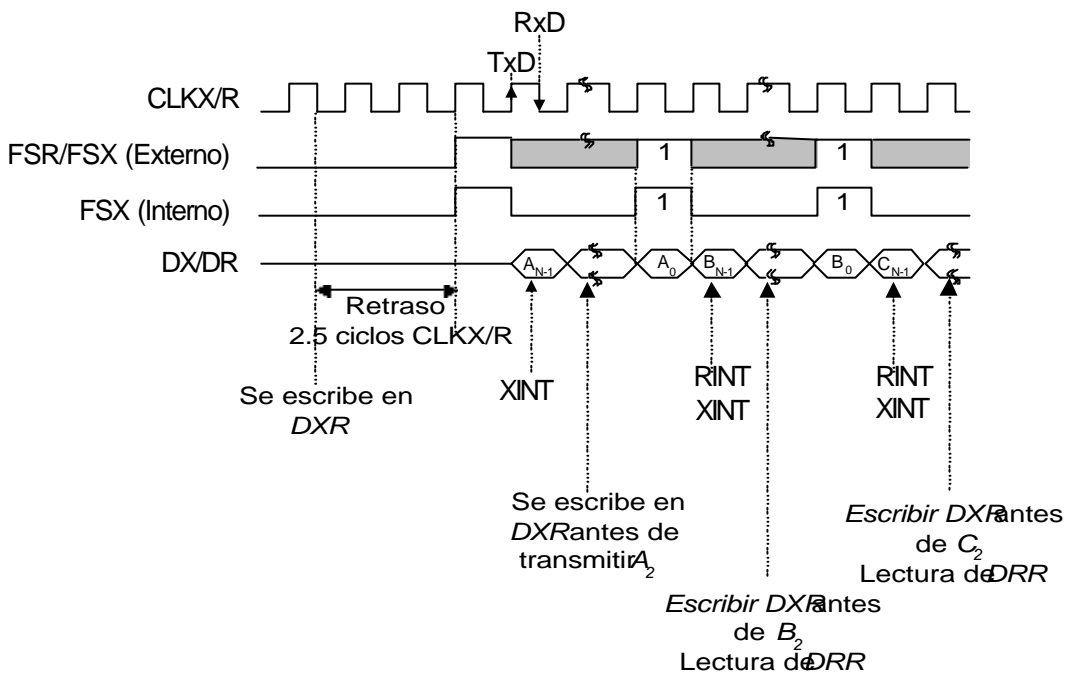


Figura 4.20. Diagrama de tiempo asociado a una transferencia serie síncrona en **modo estándar** (varios datos transferidos) y **transferencia de bits a frecuencia constante** (*fixed data-rate timing operation*). Bits (R)XFSM=0 y (R)XVAREN=0.

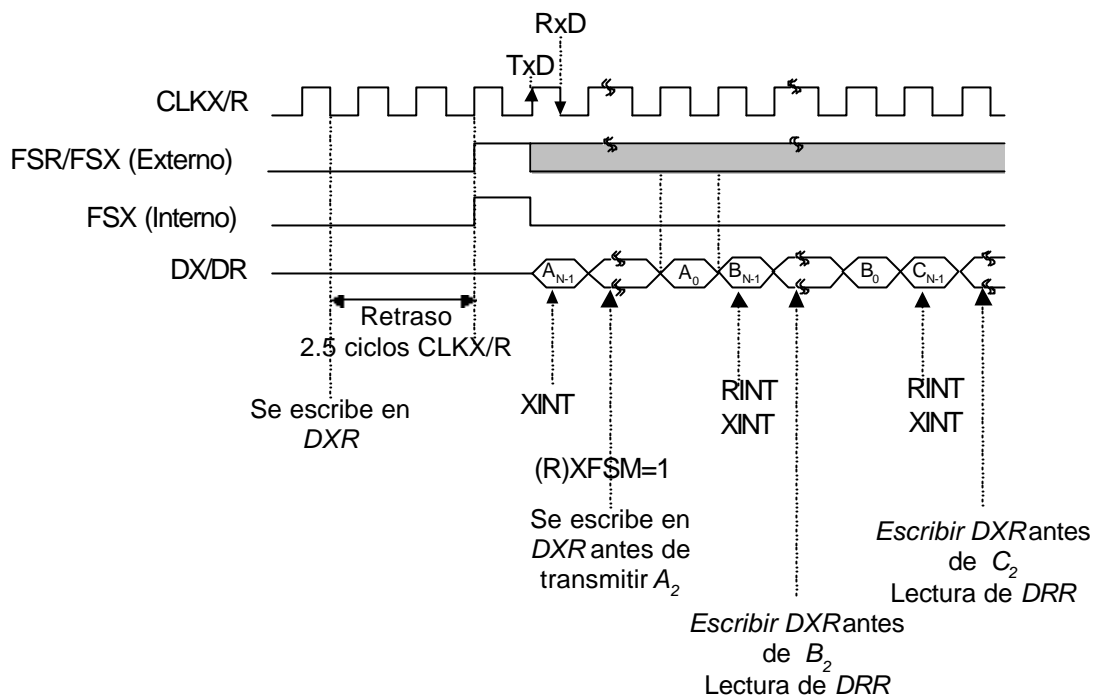


Figura 4.21. Diagrama de tiempo asociado a una transferencia serie síncrona **en modo continuo** (generación automática de la señal de sincronismo) **y transferencia de bits a frecuencia constante** (*fixed data-rate timing operation*). Bits (R)XFSM=1 y (R)XVAREN=0.

En la figura 4.22 se muestra el diagrama de tiempo asociado a una transferencia serie síncrona de datos en la que la frecuencia de transferencia de cada bit es constante. Inicialmente la transferencia se realiza en modo continuo, bit (R)XFSM a 1, para pasar luego a realizarse en modo estándar, bit (R)XFSM a 0. Se observa que, para garantizar una transferencia de bits a frecuencia constante en modo continuo es necesario escribir, antes de transferir el bit A_2 de la primera palabra que se desee transferir en modo continuo, un uno lógico en los bits XFSM (para transmisión) ó RFSM (para recepción). La conmutación de modo continuo a estándar no es inmediata. En la figura 4.20 se muestra que, desde que se escribe un cero en los bits (R)XFSM hasta que cambia de modo de transferencia síncrona (de modo continuo a modo estándar) aparece un pequeño retraso: El dato C ($C_{N-1} \dots C_0$) se transfiere en modo continuo a pesar de que XFSM=0.

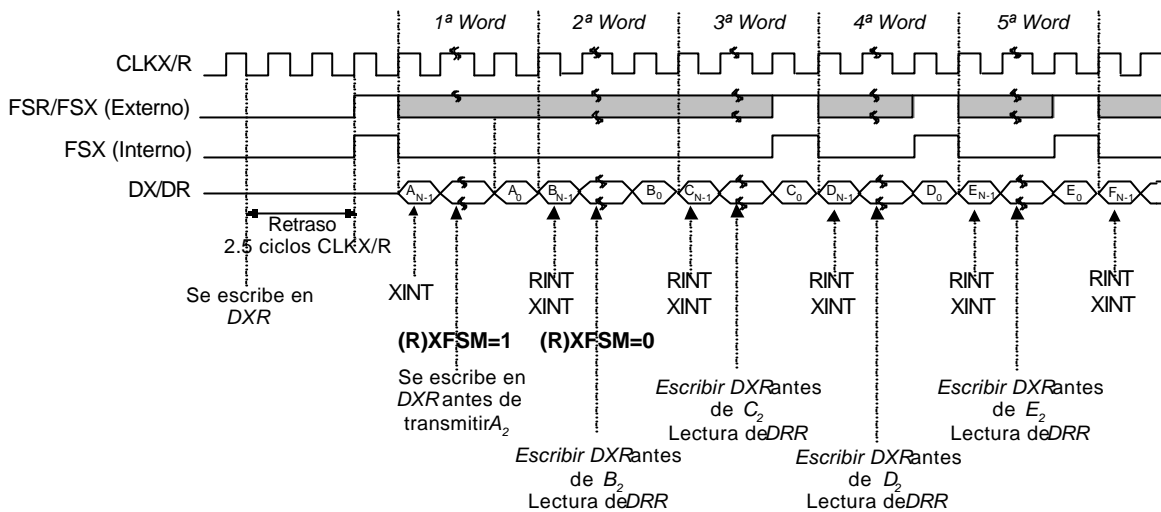


Figura 4.22. Configuración y cambio del modo de transferencia del puerto serie: Inicialmente transferencia en modo continuo (R)XFSM=1, posteriormente en modo estándar (R)XFSM=0.

En la figura 4.23, figura 4.24 y figura 4.25 se muestran los diferentes diagramas de tiempo que pueden generarse en una transferencia serie síncrona en la que el **modo de transferencia de bits es a frecuencia variable**. Obsérvese que no todos los bits permanecen activos el mismo tiempo: Todos permanecen activos un ciclo del reloj de transmisión o recepción excepto el primer bit del primer dato que permanece activo menos de un ciclo de reloj. Además, y a diferencia que en el modo de transferencia de bits a frecuencia constante, la señal de inicio de la transferencia (*FSX* ó *FSR*) se genera en el ciclo de reloj base de la transferencia que coincide con el inicio de la transferencia del dato.

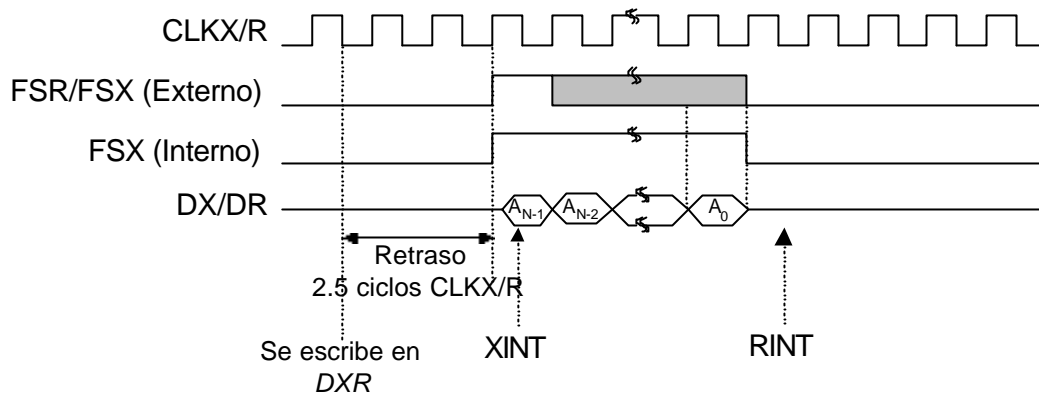


Figura 4.23. Diagrama de tiempo asociado a una transferencia serie síncrona **en modo estándar** (un único dato transferido) y **transferencia de bits a frecuencia variable** (*variable data-rate timing operation*). Bits (R)XFSM=0 y (R)XVAREN=1.

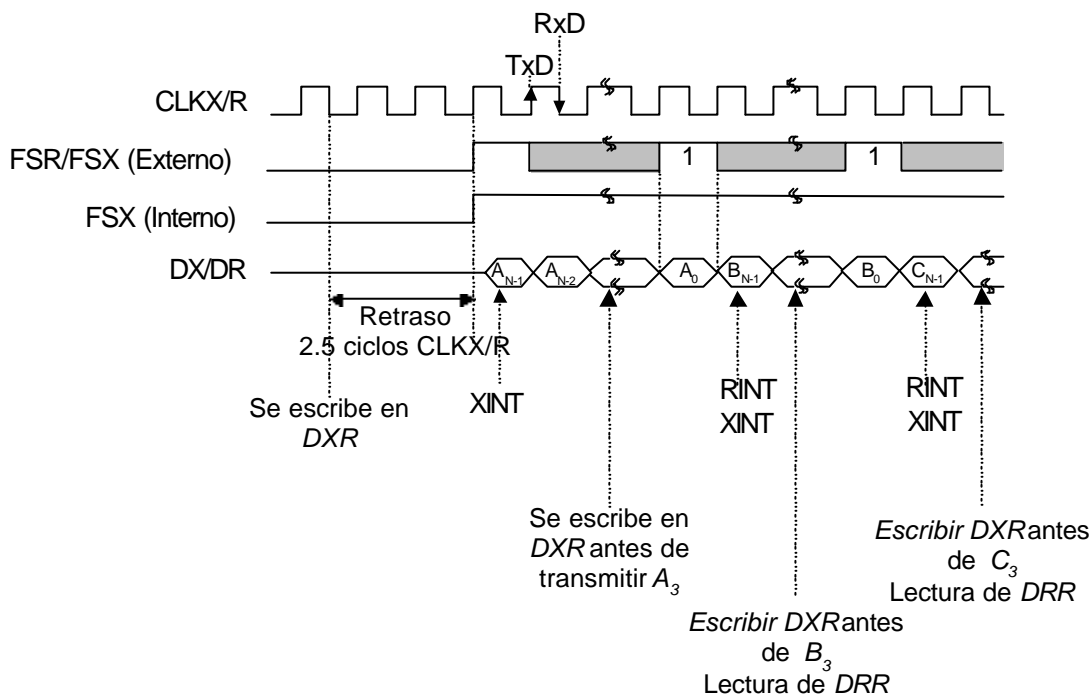


Figura 4.24. Diagrama de tiempo asociado a una transferencia serie síncrona **en modo estándar** (varios datos transferidos) y **transferencia de bits a frecuencia variable** (*variable data-rate timing operation*). Bits (R)XFSM=0 y (R)XVAREN=1.

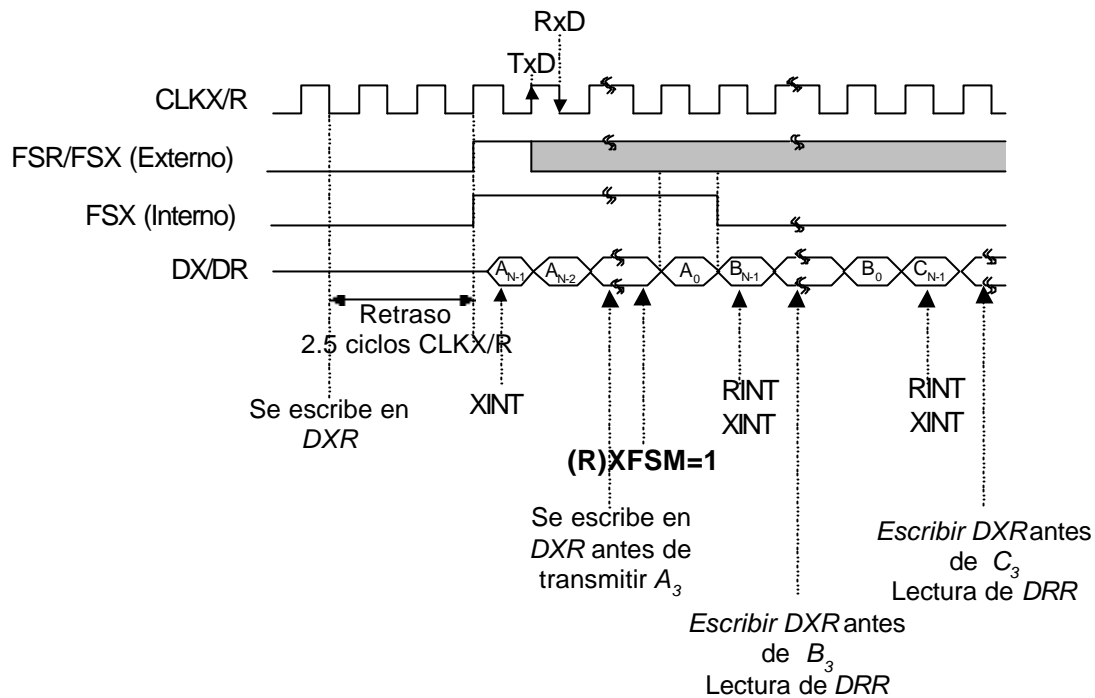


Figura 4.25. Diagrama de tiempo asociado a una transferencia serie síncrona en **modo continuo** (generación automática de la señal de sincronismo) y **transferencia de bits a frecuencia variable** (*variable data-rate timing operation*). Bits (R)XFSM=1 y (R)XVAREN=1.

5. CONTROLADOR DMA

Un periférico DMA (*Direct Memory Access*) es un periférico capaz de gestionar, sin coste alguno en tiempo de CPU para el microprocesador, transferencias de bloques de datos desde una posición en el mapa de memoria del sistema microprocesador (*origen*) hacia otra (*destino*).

Una transferencia DMA hace referencia al trasvase de un grupo de datos en el mapa de memoria del sistema digital. El número de datos que se van a trasvasar es uno de los parámetros característicos de la transferencia DMA. El periférico DMA dispone de un contador de datos transferidos que se inicializa al número de datos que se desean trasvasar. Cada vez que el periférico transfiere un dato en el mapa de memoria del sistema, disminuye en

Complementos de Sistemas Electrónicos Digitales

una unidad dicho contador. Se dice que la transferencia DMA ha concluido cuando el registro contador de datos transferidos alcanza el valor cero.

Cada dato trasvasado en el mapa de memoria implica dos operaciones, una lectura del dato en la dirección origen (*Read-DMA*) y una escritura del mismo en la dirección destino (*Write-DMA*). Al final de cada proceso de tipo *Read-DMA* (*Write-DMA*) la dirección origen (destino) puede, normalmente, incrementarse o disminuirse en una unidad o permanecer constante.

Hasta ahora hemos analizado los sistemas microprocesadores como sistemas digitales en los que había un único *maestro* (el microprocesador) y multitud de *esclavos* (periféricos de almacenamiento de programas y datos, de conversión analógico-digital y digital-analógico, etc.). El *maestro* era el que gestionaba la transferencia de información por los buses del sistema: Activaba en modo lectura o escritura al periférico al que quería acceder, mediante las líneas asociadas a los buses de control y dirección, y completaba el acceso, siguiendo un estricto diagrama de tiempo suministrado por el fabricante, leyendo (o escribiendo) información del (en el) bus de datos del sistema.

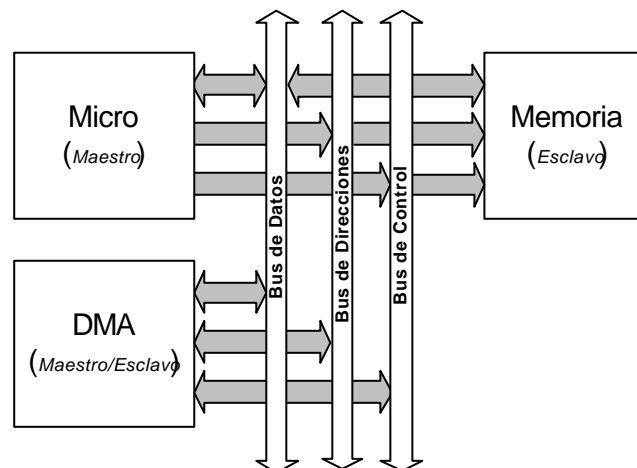


Figura 4.26. Estructura *maestro-esclavo* en un sistema microprocesador.

El periférico DMA es un dispositivo que puede funcionar, en el sistema digital en el que se encuentre inmerso, como *maestro* o *esclavo*. El microprocesador puede acceder a él para configurar su modo de funcionamiento, como haría con cualquier otro periférico,

Complementos de Sistemas Electrónicos Digitales

comportándose el periférico DMA, en ese momento, como *esclavo* en el sistema. La característica fundamental que tiene este periférico es que, una vez configurado y habilitado, su funcionamiento normal consiste en realizar transferencias de información a través de los buses del sistema, sin que dicha transferencia suponga coste de CPU para el micro. Para poder realizar esa tarea, el periférico DMA necesita poder gobernar los buses y funcionar como *maestro* en el sistema digital. Obviamente, para no provocar conflictos en los accesos a los buses, sólo puede haber un *maestro* en el sistema.

Para evitar que aparezcan dos maestros que gobiernen las líneas de los buses del sistema se añade al bus de control de muchos microprocesadores un par de líneas, una de entrada (normalmente HLD) y otra de salida (normalmente HLDA), que implementan un protocolo de acuse de recibo. El protocolo que se establece implica que, por defecto, el *maestro* en el sistema es el microprocesador. Si algún periférico desea tomar el control de las líneas asociadas a los buses de datos, dirección y control, primero debe solicitarlo al *maestro* (esta petición se realiza activando la línea HLD de entrada al microprocesador). El microprocesador analiza internamente la petición y, si el usuario que ha programado el micro lo permite, atiende la petición (deja en triestado las líneas de los buses de dirección, control y datos). Una vez que el micro ha dejado de gobernar los buses del sistema le indica, activando la línea HLDA, al periférico que quería ser *maestro* (normalmente un periférico de tipo DMA) que puede gobernar los buses.

A partir de ese momento el que controla, como *maestro*, las líneas de los buses del sistema digital es el periférico DMA. El periférico realiza, entonces, una transferencia de información (programada previamente por el microprocesador) en el mapa de memoria del sistema digital. La transferencia de información es un proceso repetitivo que consiste en un acceso en lectura a una posición de memoria (el periférico DMA impone, en las líneas asociadas al bus de direcciones del sistema, el valor de la dirección de la posición a la que desea acceder, activa la señal de lectura asociada al bus de control y copia el valor que aparece en el bus de datos en un registro interno del periférico) y un acceso posterior en escritura a otra zona de memoria (el periférico DMA impone, en las líneas asociadas al bus de direcciones del sistema, el valor de la dirección de la posición a la que accede, activa la señal de escritura asociada al bus de

Complementos de Sistemas Electrónicos Digitales

control y copia el valor que aparece en el registro interno del periférico en el bus de datos). La dirección del bloque de memoria que se desea copiar, el tamaño de dicho bloque (que indica cuantas veces consecutivas accede en lectura y en escritura la DMA como maestro del sistema digital) y la dirección donde se desea copiar el bloque de memoria son datos que programó el microprocesador cuando accedió, para configurarlo, al periférico DMA. Una vez que el periférico DMA es *maestro*, el microprocesador no puede efectuar accesos al exterior por los buses del sistema hasta que la DMA no pase a ser *esclavo*, cosa que ocurre cuando se concluye el trasvase programado.

La utilidad que ofrece un periférico de este tipo es descargar al microprocesador de la realización de operaciones de trasvase de datos. Un periférico de este tipo tiene interés sólo si el usuario tiene previsto desarrollar aplicaciones en las que se realicen gran cantidad de trasvases de datos en el mapa de memoria del sistema digital.

El **DSP** dispone de un **periférico DMA interno** diseñado para la realización de transferencias de datos desde una posición origen hacia otra posición destino diferente, sin que ello suponga coste alguno de procesado para la CPU (de forma transparente a la CPU del sistema). Las transferencias las realiza el periférico empleando los buses internos de datos y direcciones dedicados a la DMA y se pueden realizar desde y hacia cualquier posición en el mapa de memoria del DSP.

El periférico DMA interno al DSP tiene, además, asignada una fuente de petición de interrupción a la CPU. Esta fuente se corresponde con el evento final de transferencia DMA (el registro contador de la transferencia DMA alcanza el valor cero).

5.1 Registros de control y configuración.

El periférico DMA dispone de cuatro registros, de 32 bits cada uno, dedicados al control y configuración del mismo y ubicados en una determinada zona del mapa de memoria del DSP. Estos registros pueden ser accedidos en lectura o escritura por el usuario. En la figura 4.27 se muestra la posición en el mapa de memoria de estos registros, que son:

Complementos de Sistemas Electrónicos Digitales

- **Registro global de configuración y control del periférico.** Ubicado en la posición 808000H. Determina el modo de operación del periférico. Indica, además, el estado en que se encuentra la transferencia DMA. En la figura 4.28 se muestra este registro y en la tabla 4-6, tabla 4-7, tabla 4-8, tabla 4-9 y tabla 4-10 se indican las funciones de cada bandera del registro.
- **Registro de la dirección fuente en la transferencia DMA.** Ubicado en la posición 808004H. Indica la dirección en la que se encuentra el dato que se va a leer en la transferencia DMA. De los 32 bits del registro sólo tienen interés los 24 LSB (el mapa de memoria del DSP tiene 24 líneas de dirección).
- **Registro de la dirección destino en la transferencia DMA.** Ubicado en la posición 808006H. Indica la dirección en la que se desea almacenar el dato leído por el periférico DMA. De los 32 bits del registro sólo tienen interés los 24 LSB.
- **Registro contador de transferencias DMA.** Ubicado en la posición 808008H. Indica el número de transferencias que debe de realizar el periférico (número de datos que se deben reubicar en el mapa de memoria del DSP).

Complementos de Sistemas Electrónicos Digitales

Dirección	Descripción
808000h	Registro Global de Control
808001h	Reservado
808002h	Reservado
808003h	Reservado
808004h	Dirección del registro fuente DMA
808005h	Reservado
808006h	Dirección del registro destino DMA
808007h	Reservado
808008h	Contador de transferencias DMA
808009h	Reservado
80800Ah	Reservado
80800Bh	Reservado
80800Ch	Reservado
80800Dh	Reservado
80800Eh	Reservado
80800Fh	Reservado

Figura 4.27. Registros del periférico DMA interno del DSP.

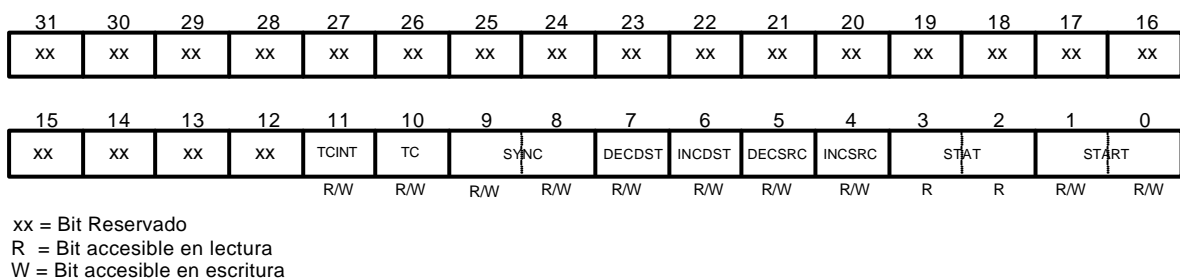


Figura 4.28. Registro Global de configuración y control del periférico DMA interno del DSP.

Bit	Nombre	Valor de Reset	Función que realiza
0-1	START	0-0	Controlan el estado en el que la DMA arranca y para. Ver tabla 4-7.
2-3	STAT	0-0	Indican el estado de la DMA. Cambian cada ciclo máquina. Ver tabla 4-8.
4	INCSRC	0	Si INCSRC=1, la posición del registro fuente para la transferencia DMA se incrementa después de cada lectura DMA.
5	DECSRC	0	Si DECSRC=1, la posición del registro fuente para la transferencia DMA se decrementa después de cada lectura DMA.
6	INCDST	0	Si INCDST=1, la posición del registro destino para la transferencia DMA se incrementa después de cada lectura DMA.
7	DECDST	0	Si DECDST=1, la posición del registro destino para la transferencia DMA se decrementa después de cada lectura DMA.
8-9	SYNC	0-0	Indican la forma de lanzar los eventos DMA (lecturas DMA y/o escrituras DMA). Ver tabla 4-9 y figura 4.29.
10	TC	0	Afecta a la operación del contador de transferencias DMA. Si TC=0, las transferencias DMA no concluyen cuando el contador de transferencias alcance el valor 0. Si TC=1, las transferencias DMA concluyen al alcanzar el contador de dichas transferencias el valor cero.
11	TCINT	0	Si TCINT=1, cuando el contador de transferencias DMA alcance el valor 0, se activa la petición de interrupción DMA. Si TCINT=0, no se activa la petición de interrupción DMA al pasar a valer 0 el contador de transferencias

Complementos de Sistemas Electrónicos Digitales

			DMA.
31-12	Reservado	0-0	Se lee siempre un cero.

Tabla 4-6. Funciones de los bits del registro de control y configuración del periférico DMA interno del DSP.

START	Función que realiza
0-0	Valor de Reset. Los ciclos de lectura o escritura DMA que aún no hayan terminado, se completan. La DMA no realiza más transferencias y se resetea, de manera que cuando se lance un nuevo evento DMA, comienza una nueva transferencia DMA.
0-1	Si se ha efectuado o ha comenzado un evento de lectura o escritura DMA cuando aparece este valor, se completa dicho evento antes de pararse la DMA. Si no ha comenzado ningún evento, la DMA se vuelve inactiva.
1-0	Si ha comenzado una transferencia DMA, se completa (incluyendo los eventos de lectura y escritura DMA) antes de pararse. Si no ha comenzado ninguna transferencia, la DMA se vuelve inactiva.
1-1	La DMA arranca desde el reset o desde el estado previo en que estaba.

Tabla 4-7. Inicio de eventos DMA. Funciones de los bits **START** del registro de control y configuración del periférico DMA interno del DSP.

STAT	Función que realiza
0-0	Valor de Reset. La DMA está parada al final de una transferencia DMA (entre un evento de escritura y otro de lectura).
0-1	La DMA está parada en la mitad de una transferencia DMA (entre un evento de lectura y otro de escritura).
1-0	Reservado.
1-1	La DMA está ocupada, está realizando un evento de lectura, uno de escritura o está esperando la llegada de una petición de interrupción para lanzar el(los) evento(s) programado(s).

Tabla 4-8. Estado del periférico DMA. Funciones de los bits **STAT** del registro de control y configuración del periférico DMA interno del DSP.

SYNC	Función que realiza
0-0	SIN SINCRONIZACIÓN. Valor de Reset. La sincronización de los eventos DMA con la petición de interrupciones, está inhabilitada.
0-1	SINCRONIZACIÓN DE LA FUENTE. La sincronización de los eventos de lectura de la DMA se realiza con la petición de interrupciones. La DMA genera un evento de lectura (completando una transferencia completa DMA) cuando se produzca la petición de una interrupción habilitada para la DMA.
1-0	SINCRONIZACIÓN DEL DESTINO. La sincronización de los eventos de escritura de la DMA se realiza con la petición de interrupciones. La DMA genera un evento de escritura (completando una transferencia completa DMA e iniciando otra) cuando se produzca la petición de una interrupción habilitada para la DMA.
1-1	SINCRONIZACIÓN DE LA FUENTE Y DEL DESTINO. La sincronización de

Complementos de Sistemas Electrónicos Digitales

	los eventos de lectura y escritura de la DMA se realiza con la petición de interrupciones. La DMA genera un evento de lectura o escritura (según corresponda) cuando se produzca la petición de una interrupción habilitada para la DMA. La transferencia completa DMA está garantizada cuando se produzcan dos peticiones de interrupciones habilitadas para la DMA.
--	---

Tabla 4-9. Modo de funcionamiento del periférico DMA: Sincronización de los eventos. Funciones de los bits **SYNC** del registro de control y configuración del periférico DMA interno del DSP.

TCINT-TC	Función que realiza
0-0	La CPU no es notificada cuando se completa una transferencia DMA (el registro contador de la transferencia pasa a valer cero). Se continúan realizando transferencias DMA cuando pasa a valer cero el registro contador de transferencias DMA.
0-1	La CPU no es notificada cuando se completa una transferencia DMA (el registro contador de la transferencia pasa a valer cero). Se dejan de realizar transferencias DMA cuando pasa a valer cero el registro contador de transferencias DMA.
1-0	La CPU es notificada, vía activación de la bandera de petición de interrupción asociada al periférico DMA, cuando se completa una transferencia DMA (el registro contador de la transferencia pasa a valer cero). Se continúan realizando transferencias DMA cuando pasa a valer cero el registro contador de transferencias DMA.
1-1	La CPU es notificada, vía activación de la bandera de petición de interrupción asociada al periférico DMA, cuando se completa una transferencia DMA (el registro contador de la transferencia pasa a valer cero). Se dejan de realizar transferencias DMA cuando pasa a valer cero el registro contador de transferencias DMA.

Tabla 4-10. Petición de interrupciones por parte del periférico DMA.

5.2 Modos de operación: Sincronización de los eventos DMA.

Como ya se ha comentado previamente, la única función que tiene un periférico DMA es el trasvase de datos desde una posición inicial (origen) hacia una posición final (destino), proceso conocido como transferencia DMA que consta de dos operaciones o eventos, una lectura de la dirección fuente (*Read-DMA*) y una escritura en la dirección destino (*Write-DMA*). Las transferencias DMA que un usuario puede programar en el DSP se pueden parar al final de cualquier evento DMA, bien de lectura o de escritura (bits **START** en tabla 4-6 y tabla 4-7).

El usuario puede programar cuatro formas de realización de las transferencias DMA, según la sincronización impuesta entre la fuente y el destino (bits 8 y 9, **SYNC**, del registro global de

Complementos de Sistemas Electrónicos Digitales

configuración del periférico). Los cuatro tipos de transferencia DMA se muestran en la figura 4.29.

Se programa la forma en que se realizan las transferencias de tipo DMA, estableciendo una especie de *breakpoint* (parada controlada de la transferencia DMA) al final de uno, los dos o ninguno de los eventos de que se compone la transferencia DMA (recuérdese que la transferencia se compone del evento lectura del dato en la posición de origen y el de escritura del mismo en la posición de destino), tabla 4-9. Las peticiones de las interrupciones habilitadas para generar eventos DMA son las que se encargan de relanzar, cuando dichas peticiones aparecen, las transferencias DMA (en esto consiste la generación de eventos DMA asociados a las interrupciones, tema 15):

- **SYNC = '00'**. Sin sincronización entre la fuente y el destino. Las interrupciones habilitadas para generar eventos DMA son ignoradas, no afectan a los procesos de transferencia que se realizan cuando no haya conflictos en la estructura *pipeline*. No existe ningún *breakpoint*.
- **SYNC = '01'**. Sincronización de la fuente. La DMA no realiza transferencias mientras no llegue alguna interrupción que tenga habilitada la generación de eventos de tipo DMA. Cuando llega una petición de interrupción, que tenga habilitada la generación de eventos DMA, se inicia la transferencia DMA realizándose primero un evento de lectura (*Read-DMA*) y a continuación se lleva a cabo un evento de escritura (*Write-DMA*). Existe, por tanto, un único *breakpoint* situado al comienzo del evento *Read-DMA*.
- **SYNC = '10'**. Sincronización del destino. Como en el caso anterior, la DMA no realiza transferencias mientras no llegue alguna interrupción que tenga habilitada la generación de eventos de tipo DMA. Cuando llega una petición de interrupción, que tenga habilitada la generación de eventos DMA, se inicia la transferencia DMA realizándose primero un evento de escritura (*Write-DMA*) y a continuación se lleva a cabo un evento de lectura (*Read-DMA*). Existe, por tanto, un único *breakpoint* situado al comienzo del evento *Write-DMA*.

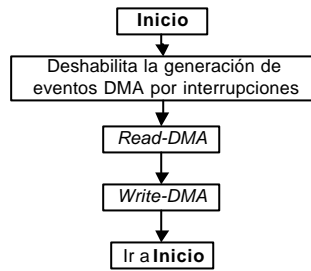
Complementos de Sistemas Electrónicos Digitales

- **SYNC = '11'**. Sincronización de la fuente y el destino. Antes de cada evento de lectura o escritura DMA hay que esperar a que ocurra una interrupción que tenga habilitada la generación de eventos de tipo DMA. Existen, por tanto, dos *breakpoints* situados al comienzo de los eventos *Read-DMA* y *Write-DMA*.

En la figura 4.30 se muestran los diagramas de tiempos simplificados asociados a las transferencias de tipo DMA. Obsérvese que las gráficas se encuentran ordenadas según la zona de memoria que hace las veces de origen y destino de la transferencia (memoria interna, memoria externa asociada al bus principal y memoria externa asociada al bus de expansión). Nótese, además, que no existe diferencia entre los accesos a través del bus de expansión con MSTRB ó IOSTRB al contrario de lo que veíamos cuando analizábamos los accesos a memoria externa del DSP. Se puede observar que:

- ✓ Los procesos de lectura y escritura de la DMA efectuados sobre memoria interna tardan un ciclo máquina.
- ✓ Los procesos de lectura y escritura de la DMA efectuados sobre memoria externa tardan más de dos ciclos máquina. Primero se efectúa el proceso de recogida del dato. Posteriormente dicho dato es almacenado en un registro de la DMA (ciclo de registro interno) y, finalmente, se realiza el trasvase del dato a la dirección de destino. *Cw* y *Cr* representan los estados de espera programados en los accesos, a través del bus externo seleccionado (principal ó de expansión). Cuando el usuario haya lanzado transferencias DMA por un determinado bus del sistema es conveniente que no modifique el registro de control de dicho bus (no altere la forma de generación ni el número de estados de espera programados al comienzo del proceso DMA) sin parar previamente la generación de los eventos DMA. Se evitan, de esta forma, accesos externos erróneos con cero estados de espera.

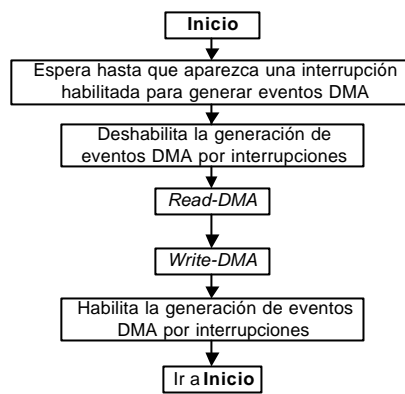
Complementos de Sistemas Electrónicos Digitales



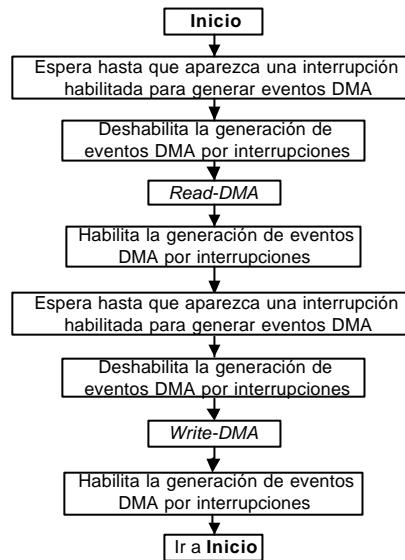
Sin sincronización
SYNC = '00'



Sincronización del destino
SYNC = '10'



Sincronización de la fuente
SYNC = '01'



Sincronización de la fuente y del destino
SYNC = '11'

Figura 4.29. Sincronización de las transferencias DMA con la petición de las interrupciones habilitadas para generar eventos DMA.

Complementos de Sistemas Electrónicos Digitales

Ciclos de H1 (ciclos máquina)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Tiempo de transferencia (ciclos H1)
Fuente: Memoria Interna Destina: Memoria Interna	R		R		R												$(1+1) T$
Fuente: Memoria Externa, $\overline{\text{STRB}}$ Destina: Memoria Interna	R	R	R	I		R	R	R	I		R	R	R	I			$(2+Cr+1) T$
Fuente: Memoria Externa, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$ Destina: Memoria Interna	R	R	R	I		R	R	R	I		R	R	R	I			$(2+Cr+1) T$

Ciclos de H1 (ciclos máquina)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	16	Tiempo de transferencia (ciclos H1)
Fuente: Memoria Interna Destina: Memoria Externa, $\overline{\text{STRB}}$	R		R				R											$1+(2+Cw) T$
Fuente: Memoria Externa, $\overline{\text{STRB}}$ Destina: Memoria Externa, $\overline{\text{STRB}}$	R	R	R	I					R	R	R	I						$(2+Cr+2+Cw) T$
Fuente: Memoria Externa, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$ Destina: Memoria Externa, $\overline{\text{STRB}}$	R	R	R	I		R	R	R	I									$(2+Cr+2+Cw)+(2+Cw+\max(0, Cr-Cw+1))(T-1)$

Ciclos de H1 (ciclos máquina)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	16	Tiempo de transferencia (ciclos H1)
Fuente: Memoria Interna Destina: Memoria Externa, $\overline{\text{STRB}}$	R		R				R											$1+(2+Cw) T$
Fuente: Memoria Externa, $\overline{\text{STRB}}$ Destina: Memoria Externa, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$	R	R	R	I		R	R	R	I									$(2+Cr+2+Cw)+(2+Cw+\max(0, Cr-Cw+1))(T-1)$
Fuente: Memoria Externa, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$ Destina: Memoria Externa, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$	R	R	R	I					R	R	R	I						$(2+Cr+2+Cw) T$

T = Número de transferencias
 Cr = Estados de espera en *Read-DMA*
 Cw = Estados de espera en *Write-DMA*
 R = *Read-DMA*
 W = *Write-DMA*
 I = Ciclo de registro interno

Figura 4.30. Transferencias DMA: Tiempo de ejecución de los accesos en lectura (*Read-DMA*) y escritura (*Write-DMA*), para diferentes fuente y destino.

Mientras no se programe, la DMA permanece inactiva (después de un *reset* el periférico DMA para las transferencias en curso y no realiza trasvase de datos). Para poder usar el periférico DMA del DSP primero es necesario configurarlo. La configuración debe realizarse, según aconseja el fabricante, manteniendo inactivo el periférico siguiendo, por tanto, el proceso indicado en la figura 4.31: En primer lugar se debe parar el periférico –poner los dos bits *START* del registro global de control de la DMA a cero–, posteriormente se realiza la

Complementos de Sistemas Electrónicos Digitales

configuración del periférico manteniendo los bits START a cero –se escribe en el registro de control global de la DMA, se programa la dirección de origen y destino así como el número de datos a transferir–, y finalmente se habilita el periférico –se ponen las banderas START a un valor distinto de 00–.

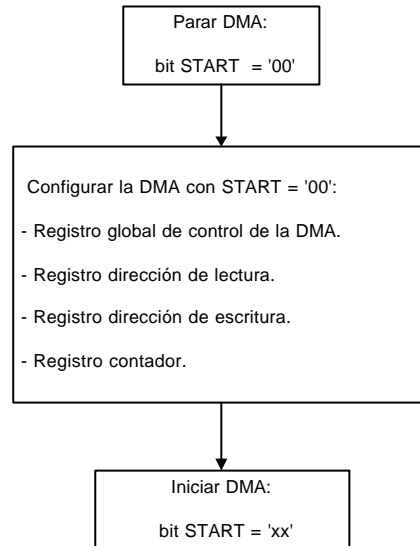


Figura 4.31. Inicialización y reconfiguración del periférico interno DMA.