



Trabajo Final Arquitectura de Computadoras : DMA

Por Ezequiel Valle Y Garcia

1. Introducción:	3
2. Funcionamiento e interacción con otros componentes:	3
3. Ventajas:	5
4. Modos de Operación:	5
Modo ráfaga	5
Modo robo de ciclo	5
Modo transparente	6

1. Introducción:

(Direct Memory Access o DMA). El acceso directo a memoria es una característica de las computadoras y microprocesadores modernos que permite que ciertos subsistemas de hardware dentro de la computadora puedan acceder a la memoria del sistema para la lectura y/o escritura, independientemente de la CPU. De lo contrario, la CPU tendría que copiar cada porción de datos desde el origen hacia el destino, haciendo que ésta no esté disponible para otras tareas. Los subsistemas de hardware que utilizan DMA pueden ser: controladores de disco duro, tarjetas gráficas, tarjetas de red, tarjetas de sonido y tarjetas aceleradoras. También es utilizado para la transferencia de datos dentro del chip en procesadores con múltiples núcleos. DMA es esencial en los sistemas integrados.

2. Funcionamiento e interacción con otros componentes:

El mecanismo de acceso directo a memoria DMA es bastante complejo en sus detalles, y por supuesto, el movimiento de grandes volúmenes de datos entre memoria y un dispositivo requiere cierta intervención del procesador. El movimiento se hace a ráfagas, y cada transferencia se inicia con una interrupción que obliga al procesador a suspender su tarea para permitir un nuevo intercambio.

Empezaremos señalando que el DMA permite mover datos entre un puerto y memoria, o entre memoria y un puerto, pero no entre dos puertos o entre dos posiciones de memoria. Cada transferencia DMA requiere cierta preparación previa; conocer el volumen de datos a transferir (la más simple es de 1 byte) y la dirección de inicio del bufer de memoria involucrado (del que se leerán los datos o donde se escribirán). Para esto dispone de dos registros para cada línea; el contador y el registro de direcciones. Según se refieran a operaciones de lectura (memoria dispositivo) o escritura (dispositivo memoria) reciben distintos nombres porque su significado difiere:

Escritura: Dirección de inicio ("Write starting address"). Contador ("Write starting word count")

Lectura: Dirección actual de lectura ("Read current address"). Contador ("Read remaining word count")

Después de cada transferencia (de 1 byte) el registro de direcciones del (DMAC Controlador de DMA) es incrementado en una unidad, y el contador es

disminuido en una unidad. Cuando este último llega a cero, la transferencia ha concluido, el DMAC pone en nivel alto la línea T/C ("Terminal Count") en el bus de control y procede a enviar al procesador la señal EOP ("End of Process"). A partir de este momento el controlador no puede realizar otra transferencia hasta que sea programado de nuevo por la CPU. Aunque existen varios canales, el sistema de prioridades garantiza que solo uno de ellos puede estar en funcionamiento cada vez, de forma que sus funcionamientos no pueden solaparse, y la señal EOP se refiere forzosamente al canal activo en ese momento.

El sistema DMA tiene distintas formas de operación, pero en el caso más simple, el proceso es como sigue: cuando un dispositivo, que tiene asignado el canal x DMA, solicita una transferencia, activa la línea DRQx, avisando así al controlador DMAC de la petición.

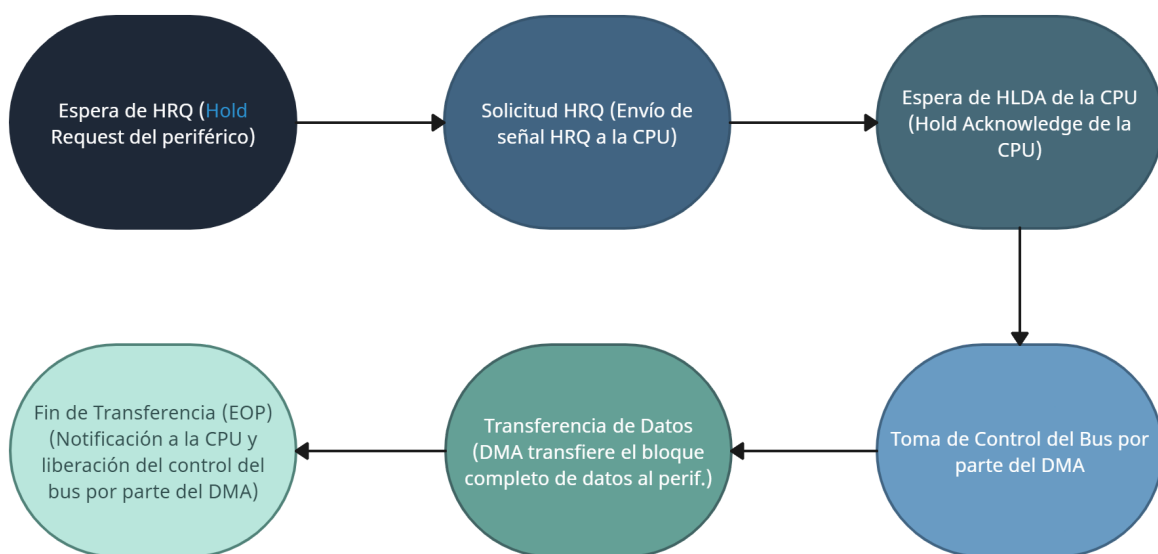
Al recibir la señal, el controlador DMAC comprueba que dicha línea está programada y activada, y que no existe ninguna petición en otra línea DRQ de prioridad más alta. A continuación solicita a la CPU que le ceda el control del bus enviándole una señal HRQ (Hold Request).

Cuando el procesador recibe la petición HRQ, termina la instrucción en curso, y una vez que está en condiciones de ceder el control del bus, deja "flotar" las señales relacionadas con su gobierno (MEMR, MEMW, IOR, IOW, Etc). Es un estado denominado tri-estado, en el que el procesador no coloca los pines de salida correspondientes en positivo ni en negativo. A continuación envía una señal HLDA (Hold Acknowledge) al DMAC por el pin correspondiente, indicándole que debe tomar el control.

A partir de este momento, el DMAC toma control del bus y monitoriza las señales que dejó flotar la CPU; activa las líneas MEMR, MEMW, IOR e IOW. Si suponemos que la transferencia DMA programada consiste en transferir 1 byte a la dirección xxxxh (de memoria), el DMAC coloca la dirección en el bus de direcciones y envía una señal al dispositivo que solicitó la transferencia activando la línea DACKx correspondiente, lo que indica al dispositivo que debe depositar el dato en el bus de datos. A continuación de la señal DACKx el controlador DMA espera un ciclo y, suponiendo que el dispositivo no haya solicitado tiempo adicional mediante la señal CHRDY (channel ready) del bus, desactiva las líneas MEMW e IOR, lo que permite que se complete la operación

("Latch"). Los datos del bus de datos son escritos en la dirección contenida en el bus de direcciones.

Suponemos que se trata de un dispositivo lento, por lo que desactiva la señal DRQx anunciando al DMAC que no lo necesita por el momento. El DMAC responde desactivando la señal DACKx que recuerda al dispositivo que no puede transmitir más, y comprueba si existen otras peticiones que atender. A continuación coloca en tri-estado las señales del bus y envía al procesador la señal EOP. El procesador contesta desactivando a su vez la señal HLDA; deja de flotar las líneas de bus y continua con el proceso que venía ejecutando.



3. Ventajas:

El uso de DMA permite que el rendimiento del sistema sea mucho más rápido, especialmente en dispositivos de alta velocidad. Ayuda a liberar la carga de trabajo de la CPU para que pueda dedicarse a otras tareas importantes en lugar de tener que dedicarse a las transferencias de datos. Además, reduce la probabilidad de errores en la transferencia de datos, lo que hace que el sistema sea más confiable.

Es un procesador especializado para transferir datos entre memoria y un periférico (y viceversa). Y además es externo al CPU y actúa simultáneamente con está, como controlador del Bus.

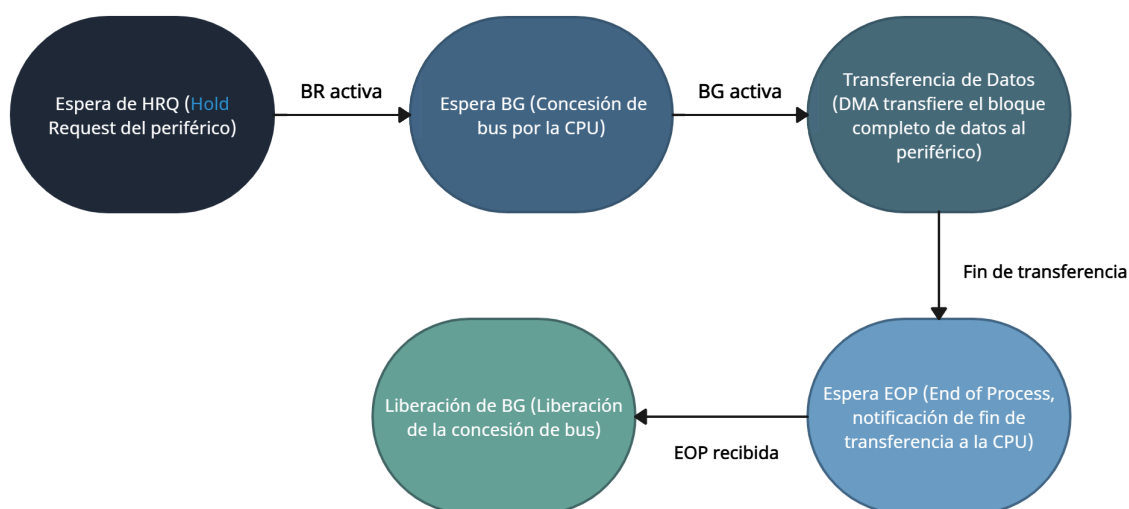
4. Modos de Operación:

Modo ráfaga

El modo de ráfaga en el DMA se refiere a la transferencia de un bloque completo de datos de manera continua. Cuando se utiliza este modo, una vez que la CPU otorga acceso al controlador DMA al bus del sistema, este último transfiere todos los bytes de datos en el bloque antes de devolver el control de los buses del sistema a la CPU. Sin embargo, durante este proceso, la CPU puede estar inactiva durante períodos de tiempo relativamente largos.

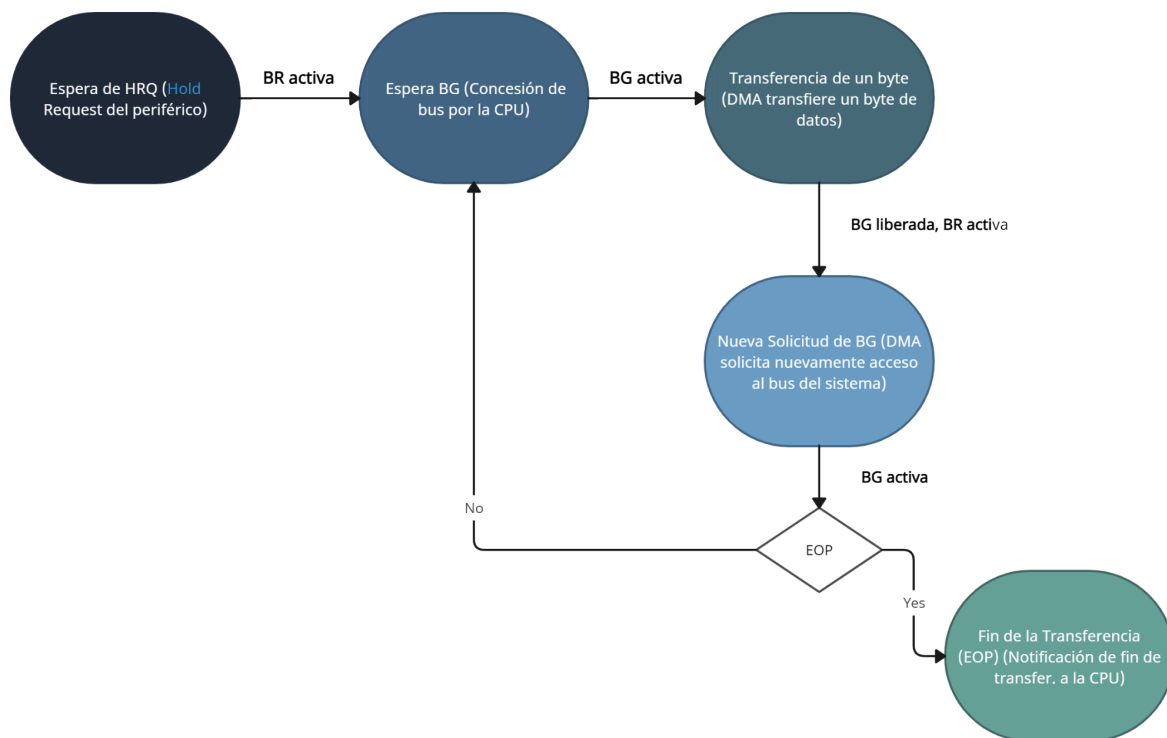
Para coordinar el acceso al bus del sistema entre la CPU y el controlador DMA en el modo de ráfaga, se utilizan las señales BR (Solicitud de bus, Bus Request) y BG (Concesión de bus, Bus Grant). BR se utiliza para solicitar el acceso al bus del sistema por parte del controlador DMA, mientras que BG se utiliza para conceder este acceso cuando el bus está disponible. Esta coordinación permite que el controlador DMA transfiera un bloque completo de datos en una secuencia contigua, aprovechando al máximo la capacidad de transferencia del bus del sistema.

El modo también se denomina "Modo de transferencia en bloque".



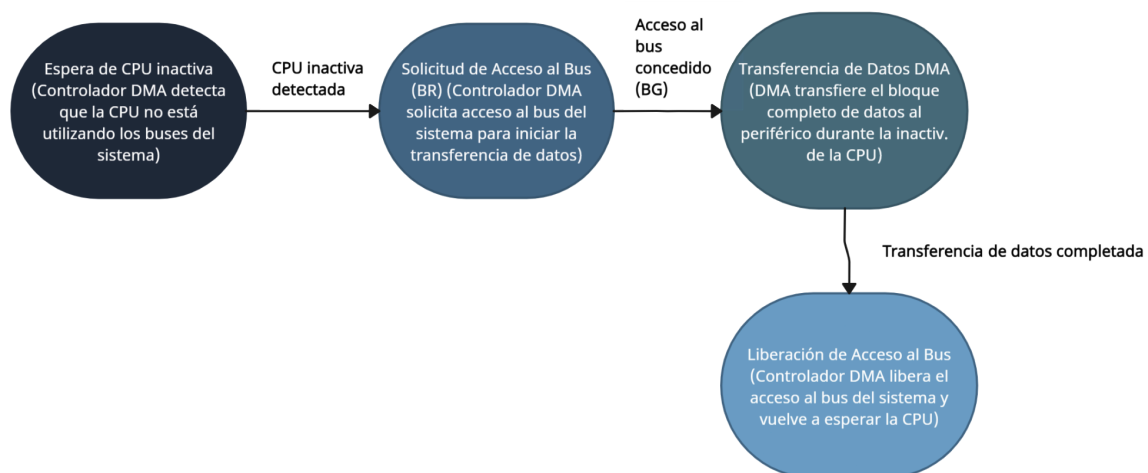
Modo robo de ciclo

El *modo de robo de ciclos* se utiliza en sistemas en los que la CPU no debe desactivarse durante el tiempo necesario para los modos de transferencia en ráfaga. En el modo de robo de ciclo, el controlador DMA obtiene acceso al bus del sistema de la misma manera que en el modo de ráfaga, usando señales BR (*Solicitud de bus*) y BG (*Concesión de bus*), que son las dos señales que controlan la interfaz entre la CPU y el controlador DMA. Sin embargo, en el modo de robo de ciclos, después de una unidad (p. ej., un byte) de transferencia de datos, el control del bus del sistema se anula a la CPU a través de BG. Luego, se vuelve a solicitar continuamente a través de BR, transfiriendo una unidad (por ejemplo, un byte) de datos por solicitud, hasta que se haya transferido todo el bloque de datos. Al obtener y liberar continuamente el control del bus del sistema, el controlador DMA esencialmente intercala transferencias de instrucciones y datos. La CPU procesa una instrucción, luego el controlador DMA transfiere un valor de datos, y así sucesivamente. Los datos no se transfieren tan rápido, pero la CPU no permanece inactiva durante tanto tiempo como en el modo de ráfaga. El modo de robo de ciclos es útil para los controladores que monitorean datos en tiempo real.



Modo transparente

El modo transparente requiere más tiempo para transferir un bloque de datos, pero también es el modo más eficiente en términos de rendimiento general del sistema. En modo transparente, el controlador DMA transfiere datos solo cuando la CPU está realizando operaciones que no utilizan los buses del sistema. La principal ventaja del modo transparente es que la CPU nunca deja de ejecutar sus programas y la transferencia DMA es gratuita en términos de tiempo, mientras que la desventaja es que el hardware necesita determinar cuándo la CPU no está utilizando los buses del sistema, lo que puede ser complejo.. Esto también se denomina "Modo de transferencia de datos DMA oculto".



5. Requisitos del Hardware

- **Controlador DMA:** El componente central es un controlador DMA dedicado que supervisa y gestiona las transferencias de datos entre los dispositivos periféricos y la memoria principal. Este controlador suele estar integrado en el chipset de la placa base o puede ser un chip independiente.
- **Bus de Datos y Direcciones:** Se necesita un bus de datos y un bus de direcciones que conecten los dispositivos periféricos, la memoria principal y el controlador DMA. Estos buses permiten la transferencia de datos y la dirección de las operaciones de lectura y escritura.

- Interrupciones: Se requiere un mecanismo de interrupciones para permitir que el controlador DMA notifique a la CPU cuando se completan las transferencias de datos o cuando se producen errores durante el proceso de DMA.
- Puertos de E/S (Entrada/Salida): Los dispositivos periféricos deben estar conectados al sistema a través de puertos de entrada/salida (E/S) que sean compatibles con DMA. Estos puertos permiten que los dispositivos periféricos envíen o reciban datos directamente a través del controlador DMA.
- Memoria Buffer: Se necesita una región de memoria dedicada, conocida como "buffer de DMA", para almacenar temporalmente los datos que se transfieren entre los dispositivos periféricos y la memoria principal. Este buffer puede ser parte de la memoria principal del sistema o puede ser una memoria dedicada para DMA.
- Soporte de DMA en Dispositivos Periféricos: Los dispositivos periféricos deben estar diseñados para admitir DMA. Esto implica que los dispositivos periféricos tengan circuitos y controladores integrados que puedan interactuar con el controlador DMA del sistema.
- Soporte del Chipset y BIOS: El chipset de la placa base y el firmware del BIOS deben ser compatibles con DMA y proporcionar las interfaces y configuraciones necesarias para configurar y controlar las operaciones de DMA.
- Sistema de Temporización: Se requiere un sistema de temporización preciso para coordinar las transferencias de datos entre los dispositivos periféricos y la memoria principal. Esto garantiza que las transferencias se realicen de manera sincronizada y eficiente.