

**Universidad Nacional Mayor de San Marcos**  
**Facultad de Ingeniería de Sistemas e Informática**

**Escuela Profesional de Ingeniería de Software**

**Lima, Perú**



**Asignatura: Arquitectura de computadores**

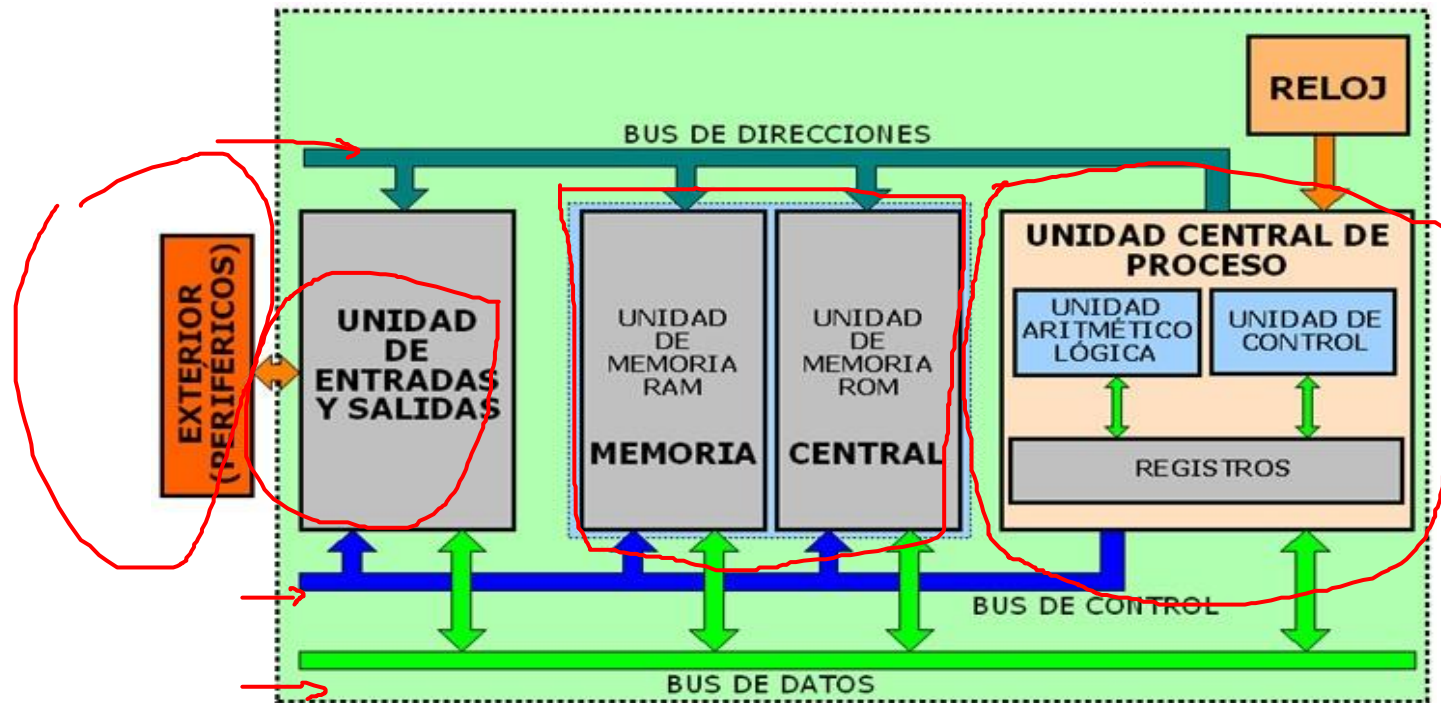
# **Controlador de Acceso Directo a Memoria - DMA**

**Dr. Igor Aguilar Alonso**

---

## **9. Controlador de Acceso Directo a Memoria**

- Los datos que ingresan tienen que atravesar una ruta por la CPU.
- La velocidad de transferencia de los periféricos E/S no es la misma que el procesador.



## Acceso directo a memoria

El acceso directo a memoria es una técnica de diseño del hardware que permite a los periféricos conectados a un sistema realizar transferencias sobre la memoria sin la intervención del procesador. De esta manera, las lentas operaciones de E/S de bloques de datos, se pueden realizar en la sombra mientras la CPU se dedica a otras tareas más útiles.

Existen inconvenientes de las E/S *programadas* y *controladas por interrupción*.

Las E/S *controladas por interrupción*, aunque son más eficientes que las E/S *programadas simples*, aún requieren la intervención activa del procesador para transferir datos entre la memoria y un módulo de E/S, y cualquier transferencia de datos debe atravesar una ruta a través del procesador.

## Acceso directo a memoria

Ambas formas de E/S adolecen de **dos inconvenientes** inherentes:

1. La *velocidad de transferencia de E/S* está limitada por la velocidad con la que el procesador puede probar y dar servicio a un dispositivo.
2. El *procesador está atado en la gestión de una transferencia de E/S*; se deben ejecutar varias instrucciones para cada transferencia de E/S.

Existe una especie de compensación entre estos dos inconvenientes. Considere la transferencia de un bloque de datos.

Usando E/S *programadas simples*, el procesador está dedicado a la tarea de E/S y puede mover datos a una velocidad bastante alta, a costa de no hacer nada más.

La interrupción de E/S libera el procesador hasta cierto punto a expensas de la velocidad de transferencia de E/S.

Sin embargo, ambos métodos tienen un impacto adverso tanto en la actividad del procesador como en la velocidad de transferencia de E/S.

Cuando se van a mover grandes volúmenes de datos, se requiere una técnica más eficiente conocida por: **acceso directo a memoria (DMA)**.

## Función de DMA

El DMA implica un *módulo adicional en el bus del sistema*. El módulo DMA (Figura siguiente) es capaz de imitar al procesador y, de hecho, de asumir el control del sistema desde el procesador.

Debe hacer esto para *transferir datos hacia y desde la memoria a través del bus del sistema*.

Para este propósito, el módulo DMA debe usar el bus solo cuando el procesador no lo necesita, o debe forzar al procesador a suspender la operación temporalmente.

La última técnica es más común y se conoce como robo de ciclo, porque el módulo DMA en efecto roba un ciclo de bus.

## Módulo de Acceso Directo a Memoria (DMA).

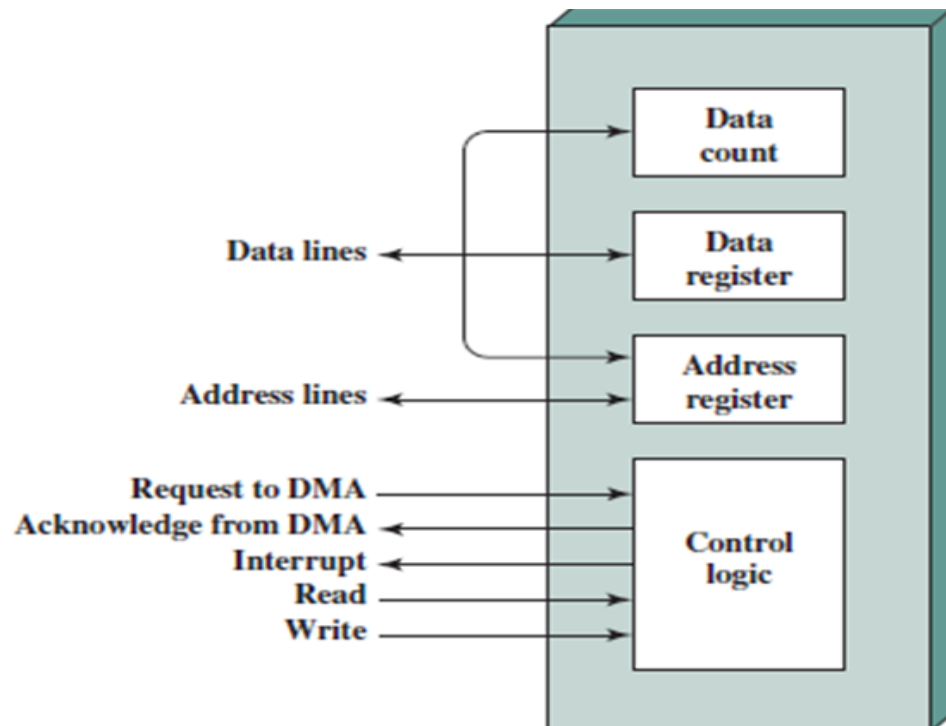


Diagrama de bloque típico de DMA



Cuando el procesador desea **leer** o **escribir** un bloque de datos, emite un comando al módulo de DMA, enviando al módulo DMA la siguiente información:

- ❑ Si se solicita una **lectura o escritura**, utilizando la **línea de control** de lectura o escritura entre el procesador y el módulo DMA.
- ❑ La **dirección del dispositivo de E/S** involucrado, comunicada en las **líneas de datos**.
- ❑ La ubicación de **inicio en la memoria para leer o escribir**, comunicada en las **líneas de datos** y almacenada por el módulo DMA en su **registro de direcciones**.
- ❑ El **número de palabras para leer o escribir**, nuevamente comunicadas a través de las **líneas de datos** y almacenadas en el **registro de conteo de datos**.

## Controlador de DMA Intel 8237A

- El controlador de DMA Intel 8237A se conecta a la familia de procesadores 80 <sup>2</sup> <sub>3</sub> <sub>4</sub> 86 y a la memoria DRAM para proporcionar una capacidad DMA.
- El controlador de DMA 8237A contiene **cuatro canales de DMA** que se pueden programar de forma independiente, y cualquiera de los canales puede estar activo en cualquier momento para que realicen sus tareas.
- Estos canales están numerados 0, 1, 2 y 3.

### Circuito integrado Intel 8237

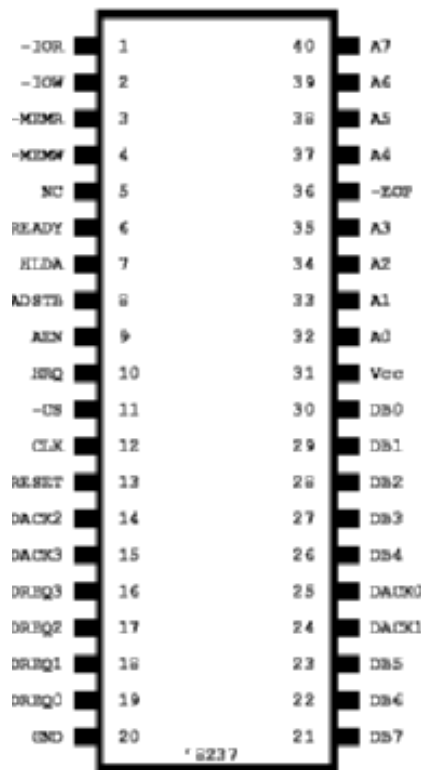


Circuito Integrado Intel 8237 en un encapsulado del tipo dual in line package

Tipo	Activo, circuito integrado
Principio de funcionamiento	Controlador de DMA
Invención	Intel

- El 8237A tiene un **conjunto de cinco registros de control / comando** para programar y controlar la operación de DMA en uno de sus canales, los cuales son:
  - *Comando.*
  - *Estado.*
  - *Modo.*
  - *Mascara individual.*
  - *Mascara completa.*
- El 8237A tiene **ocho registros de datos**: un registro de dirección de memoria y un registro de conteo para cada canal. El procesador establece estos registros para indicar la ubicación del tamaño de la memoria principal que se verá afectada por las transferencias.

# DESCRIPCIÓN DEL INTEGRADO 8237.



**CLK:** Señal de reloj básica.

**-CS:** Línea de habilitación del chip.

**RESET:** Esta señal provoca la limpieza de los registros de comando, estado, solicitud y los temporales; borra el banderín last/first y el contador de registro de modo; el registro de máscara se asigna para ignorar las solicitudes. El 8237 queda en Ciclo Inactivo.

**READY:** Señal que puede ser empleada para extender los pulsos de lectura y escritura en memoria del 8237 para trabajar con memorias lentas.

**HLDA:** Hold Acknowledge, línea por la que la CPU indica que ha liberado los buses.

**DREQ0..3:** DMA Request; son 4 líneas asíncronas de petición de DMA. En el modo de prioridad fija, DREQ0 tiene la máxima y DREQ3 la mínima. Los periféricos solicitan el servicio de DMA en estas líneas y esperan a bajarlas hasta el correspondiente DACK. La polaridad de DREQ es programable. Las líneas no usadas deben ser enmascaradas.

**DB0..DB7:** BUS de datos bidireccional y triestado. Durante los ciclos de DMA, los 8 bits más significativos de la dirección son colocados en el bus de datos con objeto de ser almacenados en un latch exterior controlado por ADSTB. En las operaciones memoria-memoria, el bus de datos recibe y envía los bytes a transferir.

**-IOR:** I/O Read. Línea bidireccional de 3 estados. En el ciclo inactivo es una entrada empleada por la CPU para leer los registros de control; en el ciclo activo actúa como línea de salida para que el 8237 controle la lectura de datos de los periféricos.

**-IOW:** I/O Write. Línea bidireccional de 3 estados. En el ciclo inactivo es una entrada empleada por la CPU para escribir los registros del 8237; en el ciclo activo actúa como línea de salida para que el 8237 controle la escritura de datos en los periféricos.

**-EOP:** End Of Process. Línea bidireccional que informa de la finalización del servicio DMA. El 8237 permite que un ente exterior fuerce el final de un servicio bajando esta línea. El propio 8237 genera un pulso en ella cuando se alcanza un TC (Terminal Count, fin de cuenta) en algún canal, salvo en el modo memoria-memoria del canal 0 (en ese caso, la señal se produce al alcanzarse el TC del canal 1). Esta patilla está conectada en el interior del chip a un transistor en colector abierto, por lo que requiere una resistencia externa. Cuando llega una señal -EOP, el 8237 finaliza el servicio aunque en el modo de autoinicialización los registros base volverán a ser escritos en los registros en curso del canal implicado. El canal resulta enmascarado salvo en el caso del modo de autoinicialización. A0..A3: Líneas bidireccionales triestado de direcciones. En el ciclo inactivo son entradas empleadas para direccionar los registros internos a leer o escribir. En el ciclo activo, son salidas y proveen los 4 bits menos significativos de la dirección.

**A4..A7:** Líneas triestado de salida de direcciones. Proveen los 4 bits altos de la dirección durante el ciclo activo.

**HRQ:** Hold Request. Línea de salida para solicitar los buses a la CPU, en el caso en que haya que realizar una transferencia. En los sistemas en que el 8237 controla totalmente el bus, esta patilla puede ir directamente conectada a HLDA.

**DACK0..3:** DMA Acknowledge. Avisa a los periféricos de que ha sido atendida su petición. El nivel de operación de esta línea es programable. RESET las baja.

**AEN:** Address Enable. Habilita el latch de 8 bits que guarda la parte alta de la dirección. Sirve también para inhibir el acceso al bus por parte de otras fuentes.

**ADSTB:** Address Strobe. Línea que controla el almacenamiento de la parte alta de la dirección, cuando está en el bus de datos, en el latch externo.

**-MEMR:** Memory Read. Salida triestado empleada para acceder a la memoria durante la lectura o las transferencias memoria-memoria.

**-MEMW:** Memory Write. Salida triestado empleada para acceder a la memoria durante la escritura o las transferencias memoria-memoria.

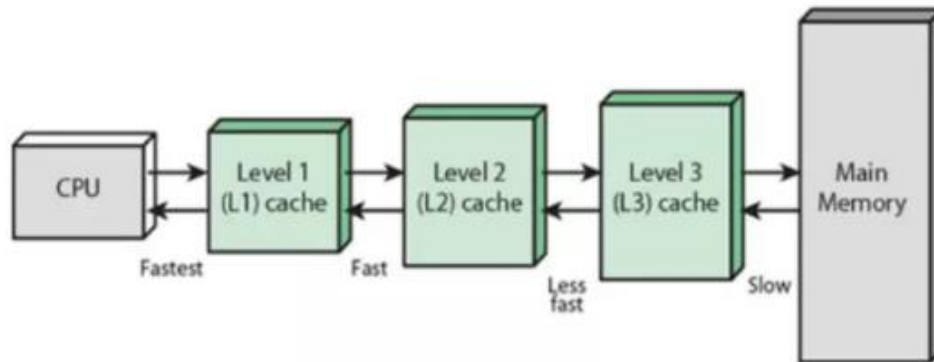
## Acceso directo a memoria caché

- El DMA es un medio eficaz *para mejorar el rendimiento de E/S con dispositivos periféricos y el tráfico de E/S de red.*
- Sin embargo, debido a los dramáticos aumentos en las velocidades de datos para E/S de red, DMA no puede escalar para satisfacer la mayor demanda.
- Esta demanda proviene principalmente del despliegue generalizado de **conmutadores Ethernet de 10 Gbps y 100 Gbps** para manejar grandes cantidades de transferencia de datos hacia y desde servidores de bases de datos y otros sistemas de alto rendimiento [STAL14a].

- Una fuente secundaria pero cada vez más importante de tráfico proviene de Wi-Fi en el rango de gigabits.
- Los dispositivos Wi-Fi de red que manejan 3.2 Gbps y 6.76 Gbps están cada vez más disponibles y generan demanda en los sistemas empresariales [STAL14b].
- La memoria caché que está más cerca de la memoria principal, denominada memoria caché de último nivel. En algunos sistemas, este será un caché L2, en otros un caché L3.

## ADM caché en microprocesadores actuales

- A un nivel básico, la memoria caché es muy rápida y contiene un pequeño conjunto de instrucciones que el equipo usa con frecuencia para realizar sus tareas cotidianas.



- Los primeros procesadores en los que se implementaron este tipo de memoria no funcionaban así.
- Los SoC (System-on-Chip) más actuales tienen varios niveles de caché, diferenciados en L1, L2, L3 e incluso L4 en algunos casos, (la L viene de «level» o nivel, por lo tanto, hablamos de caché de nivel 1, nivel 2, etc.).

## Caché L1

- Es el nivel más básico, la más cercana al procesador y la más rápida. +
- Es la que menos capacidad tiene, por ejemplo un procesador de dos núcleos como el Pentium G4560 tiene 64 KB en total (32 KB por núcleo), aunque los modelos más potentes (serie Xeon y Epyc) pueden alcanzar los 3 MB sin problema.

## Caché L2

- Es un nivel intermedio que presenta un buen equilibrio entre capacidad, cercanía y velocidad.
- Siguiendo el ejemplo anterior el Pentium G4560 tiene 512 KB (256 KB por núcleo) de caché L2, pero un Threadripper 2990WX llega a los 16 MB en total (512 KB por núcleo)..



## Caché L3

- Se posiciona en un nivel inferior a la anterior tanto en cercanía como en velocidad, pero tiene una capacidad mucho mayor.
- El Penitum G4560 tiene 3 MB de caché L3 compartida en sus dos núcleos, mientras que el Threadripper 2990WX viene con un total de 64 MB (16 MB compartidos por cada bloque de ocho núcleos).

## Caché L4

- Tipo de memoria caché poco habitual, se utiliza normalmente como apoyo para mejorar el rendimiento de GPUs integradas.
- Por ejemplo, el Core i5 5775C venía con 6 MB de caché L3 y 128 MB de eDRAM como caché L4, que se utilizaba como buffer para la gráfica Intel Iris Pro 6200 que integraba. Mejoraba el ancho de banda y se reducía el impacto de tener que recurrir a la RAM como memoria gráfica.

## Ventajas del acceso directo a memoria caché

- **Minimiza la latencia** al permitir transferencias directas de datos entre las capas de caché, evitando el acceso a la memoria principal.
- **Reduce el tiempo de espera** al evitar el retraso asociado con el acceso a la memoria principal.
- Permite compartir información entre núcleos de procesador rápidamente sin acceder constantemente a la memoria principal.
- Es especialmente beneficioso en sistemas multiprocesador o multinúcleo, donde varios núcleos comparten la jerarquía de caché.
- Contribuye a evitar cuellos de botella de memoria al agilizar las transferencias de datos entre las capas de caché.

**¿Preguntas ...?**

# Gracias por su atención

[iaquilara@unmsm.edu.pe](mailto:iaquilara@unmsm.edu.pe)

[Igor Aguilar Alonso](#)

## Trabajo grupal

Redacte un informe de laboratorio y explíquelo en clase.

Durante la sesión de clase en forma grupal leer e investigue información adicional de los apartados del Capítulo 7, que se indican a continuación:

1. Acceso directo a memoria.
2. Acceso directo a la memoria caché.
3. Como funciona el acceso directo a la memoria en un i7 - i8
4. Como funciona el acceso directo a la memoria en un i9 – superior.
5. Como funciona el acceso directo a la memoria cache de las ultimas generaciones de microprocesadores.

## Trabajo grupal

Redacte un informe de laboratorio y explíquelo en clase.

Durante la sesión de clase en forma grupal leer e investigue información adicional de los apartados del Capítulo 7, que se indican a continuación:

1. Acceso directo a memoria.
2. Acceso directo a la memoria caché.
3. Como funciona el acceso directo a la memoria en un i7 – i9