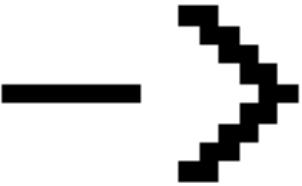


Organización del computador

Entrada / Salida

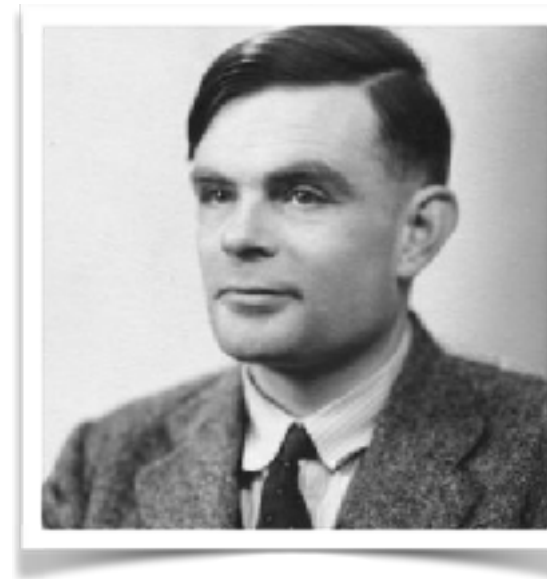
Jerarquía de máquina



Nivel 6	Usuario	Programa ejecutables
Nivel 5	Lenguaje de alto nivel	C++, Java, Python, etc.
Nivel 4	Lenguaje ensamblador	Assembly code
Nivel 3	Software del sistema	Sistema operativo, bibliotecas, etc.
Nivel 2	Lenguaje de máquina	Instruction Set Architecture (ISA)
Nivel 1	Unidad de control	Microcódigo / hardware
Nivel 0	Lógica digital	Circuitos, compuertas, memorias

- ➤ Cada nivel funciona como una máquina abstracta que oculta la capa anterior
- ➤ Cada nivel es capaz de resolver determinado tipo de problemas a partir de comprender un tipo de instrucciones específico
- ➤ La capa inferior es utilizada como servicio

Von Newman / Turing



- * Los programas y los datos se almacenan en la misma memoria sobre la que se puede leer y escribir
- * La operación de la máquina depende del estado de la memoria
- * El contenido de la memoria es accedido a partir de su posición
- * La ejecución es secuencial (a menos que se indique lo contrario)

Arquitectura de von Neumann

—> 3 componentes principales:

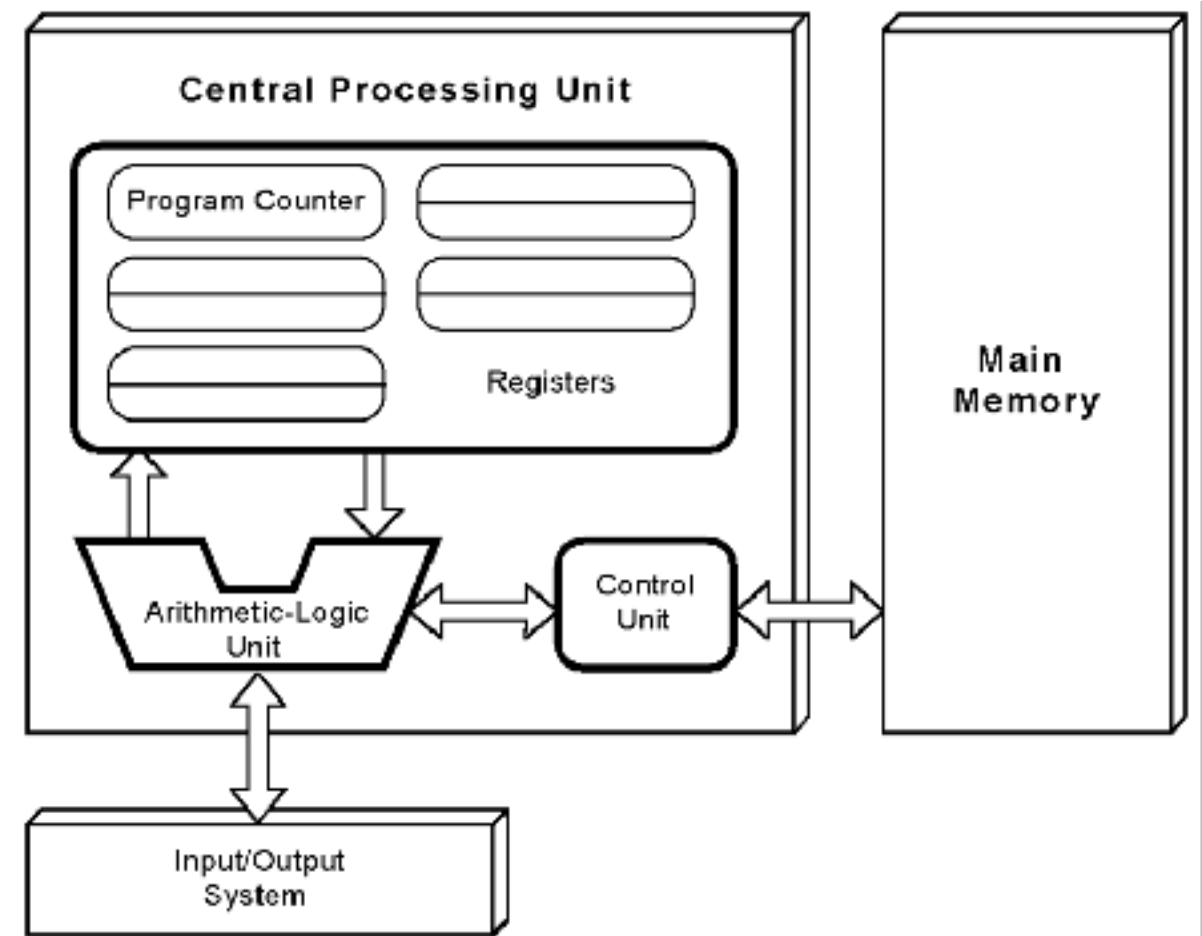
- **CPU:** Unidad de control, Unidad Aritmética lógica, Registros
- **Memoria:** Almacenamiento de programas y datos
- **Sistema** de Entrada y Salida

—> Procesamiento secuencial de instrucciones

—> Datos almacenados en sistema binario

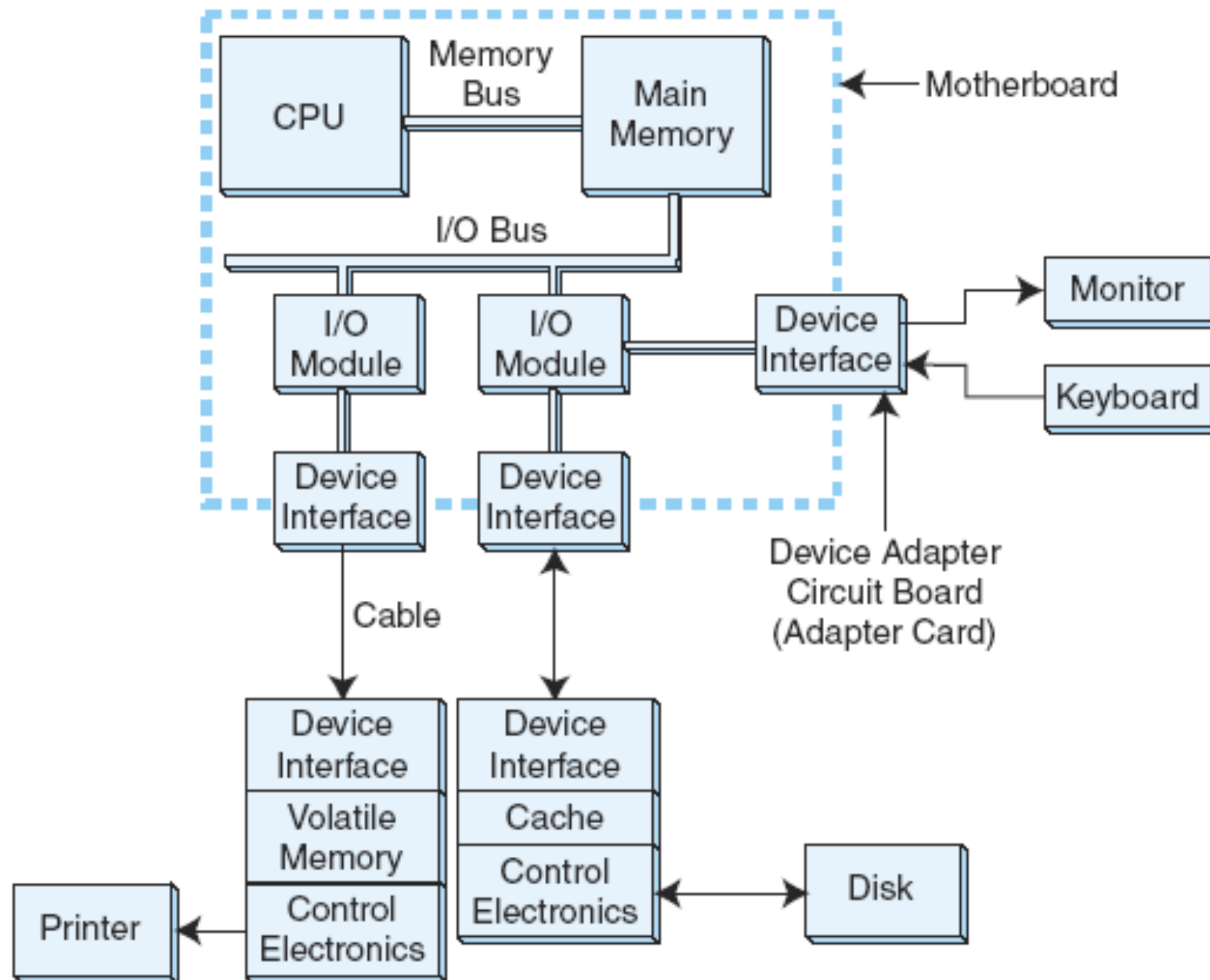
—> Sistema de interconexión de componentes:

- Conecta la Unidad de Control con la Memoria mediante un camino único
- La unicidad del camino fuerza la alternación entre ciclos de lectura / escritura y ejecución
- Esta alternación se llama cuello de botella de von Neumann (von Neumann bottleneck)^[*]



[*] El término “cuello de botella de von Neumann” fue acuñado por John Backus en su conferencia de la concesión del Premio Turing ACM de 1977.

Modelo de Entrada/Salida



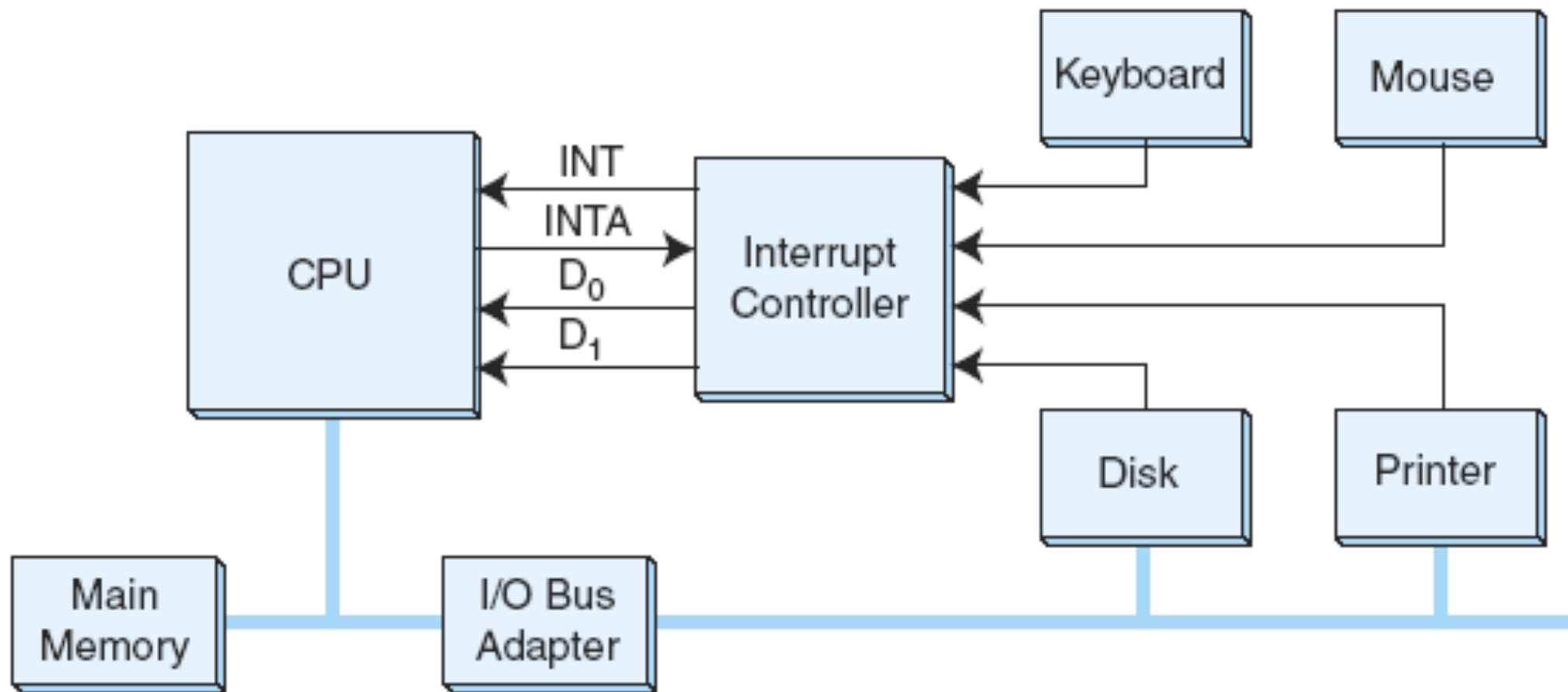
Modelo de Entrada/Salida

- ➤ **Acceso a Entrada/Salida:** refiere a cómo se envía y recibe información de los dispositivos de I/O
 - ➤ **En el espacio de memoria:** Se accede con instrucciones de lectura y/o escritura en una parte predeterminada de la memoria del sistema que es interpretada por el controlador del dispositivo
 - ➤ **En un espacio dedicado de I/O:** Se accede con instrucciones especiales de I/O normalmente **In** y **Out**

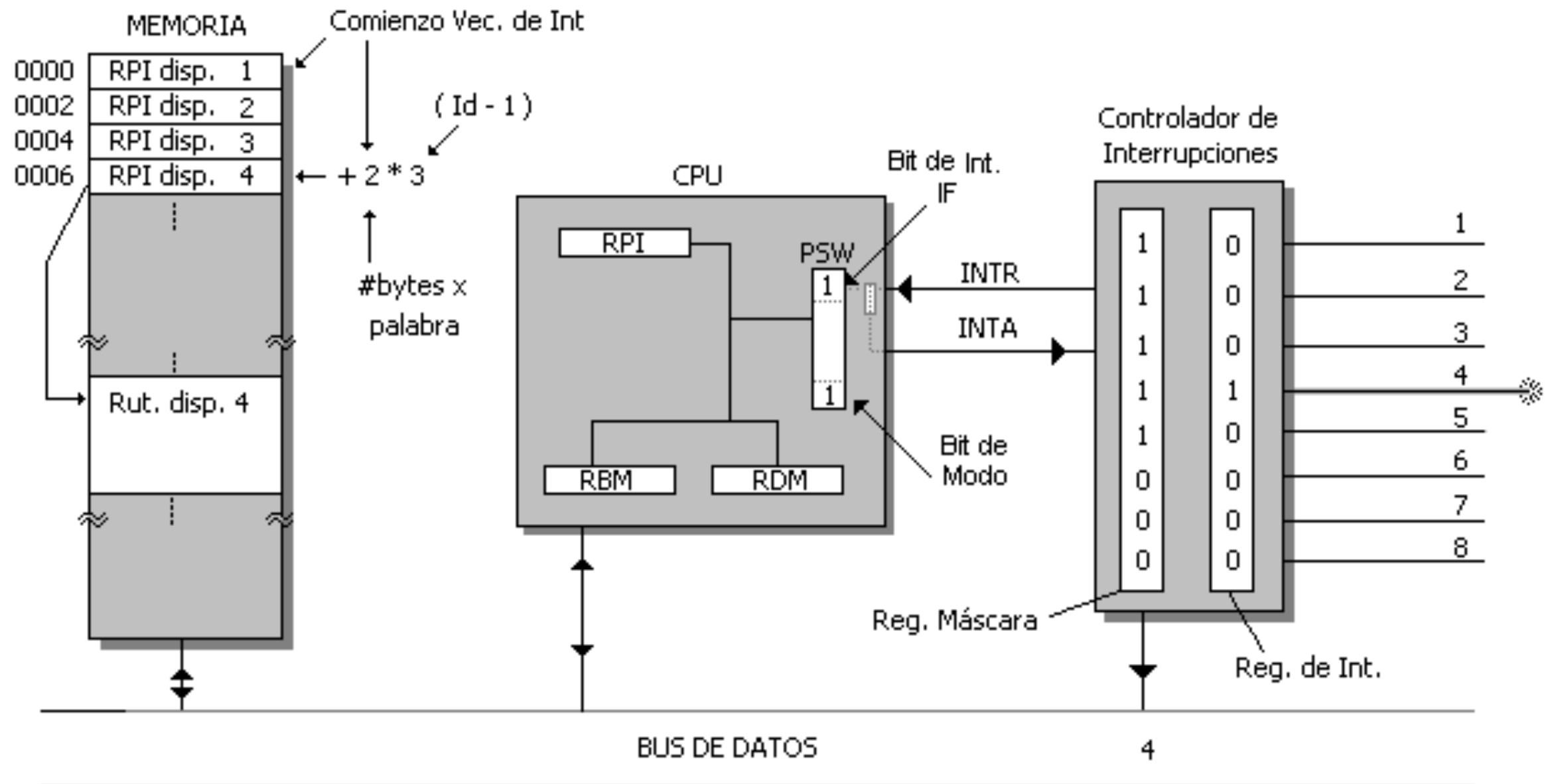
Modelo de Entrada/Salida

- ➤ **Comunicación con el dispositivo:** refiere a cómo el dispositivo se comunica con la CPU para notificarla de un evento de I/O
 - ➤ **Polling:** La CPU consulta periódicamente a los dispositivos de I/O en la eventualidad de que este desee notificara de algún evento I/O
 - ➤ **Interrupciones:** El dispositivo de I/O posee señales que permiten interrumpir a la CPU para notificarle de un evento de I/O

Entrada/Salida con Interrupciones



Interrupciones en el 8086



Interrupciones: hardware

1. El controlador del dispositivo de I/O activa la señal de interrupción a la que se encuentra conectado
2. La CPU termina de ejecutar la instrucción en curso y verifica si el flag IF está en 1 y la señal INTR está alta
3. En este caso, sube la señal INTA
4. El PIC envía por el bus de datos el identificador del dispositivo que produjo la interrupción.
5. Guarda el contexto del programa en la pila (PSW y PC)
6. Deshabilita las interrupciones globalmente usando el flag IF colocándolo en 0
7. Se pasa el procesador a modo **kernel** para que la rutina ejecute con privilegios de sistema operativo
8. Coloca $\text{Mem}[\text{Inicio_vector} + ((\text{id_disp} - 1) * 2)]$ en PC.

Interrupciones: software

1. Se guarda la máscara de interrupciones
2. Se modifica la máscara a fin de habilitar selectivamente las interrupciones que permitiremos durante la ejecución de la rutina
3. Se habilitan globalmente las interrupciones colocando el flag IF en 1
4. Se ejecuta la rutina de atención de la interrupción solicitada
5. Se deshabilitan globalmente las interrupciones colocando el flag IF en 0
6. Se reemplaza la máscara de interrupciones con la original
7. Retorna de la interrupción con una instrucción que restaura el estado del programa antes del llamado (IRET).
 1. Restaura el PC de la pila
 2. Restaura el PSW de la pila (datos de interrupción)

Programmable Interrupt Controller

Intel 8259A



8259A PROGRAMMABLE INTERRUPT CONTROLLER (8259A/8259A-2)

- 8086, 8088 Compatible
- MCS-80, MCS-85 Compatible
- Eight-Level Priority Controller
- Expandable to 64 Levels
- Programmable Interrupt Modes
- Individual Request Mask Capability
- Single +5V Supply (No Clocks)
- Available in 28-Pin DIP and 28-Lead PLCC Package
(See Packaging Spec., Order #231389)
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

Intel 8259A

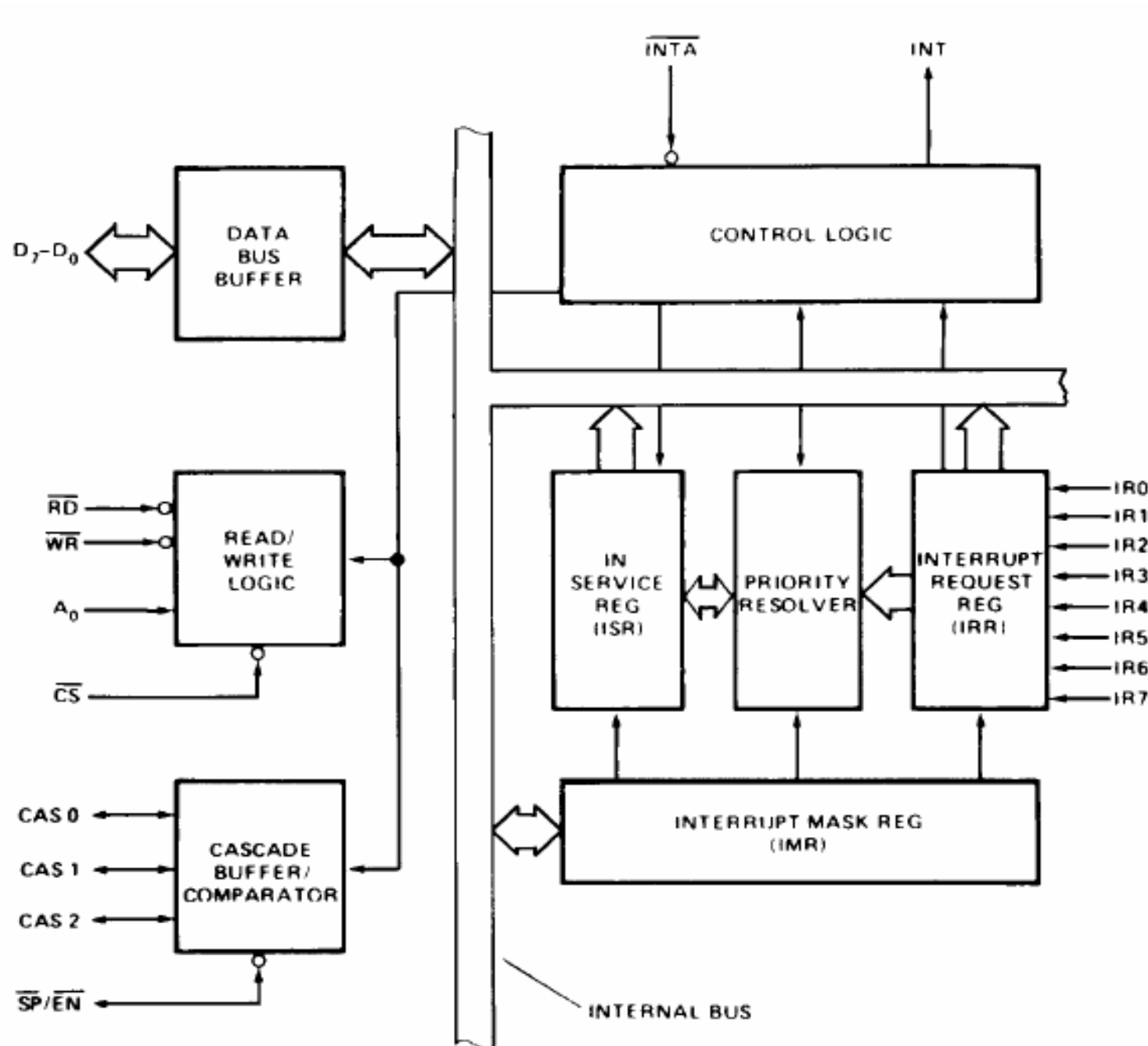


Figure 1. Block Diagram

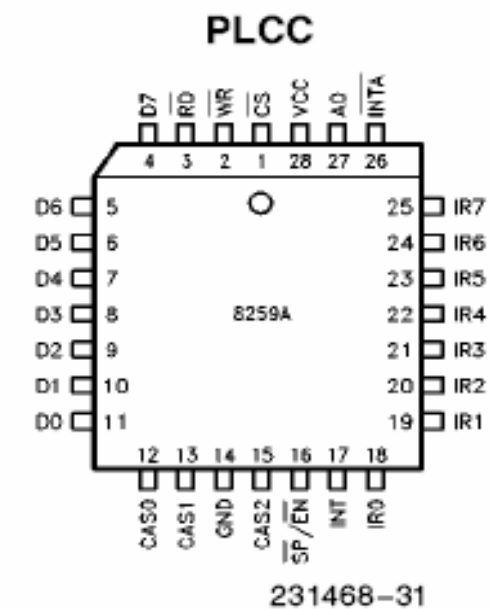
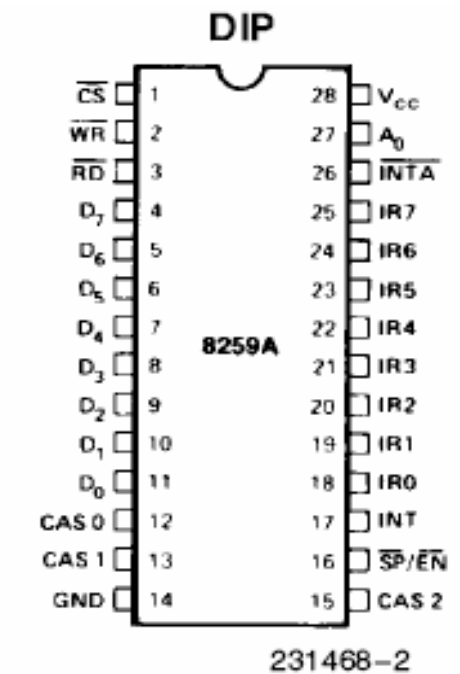
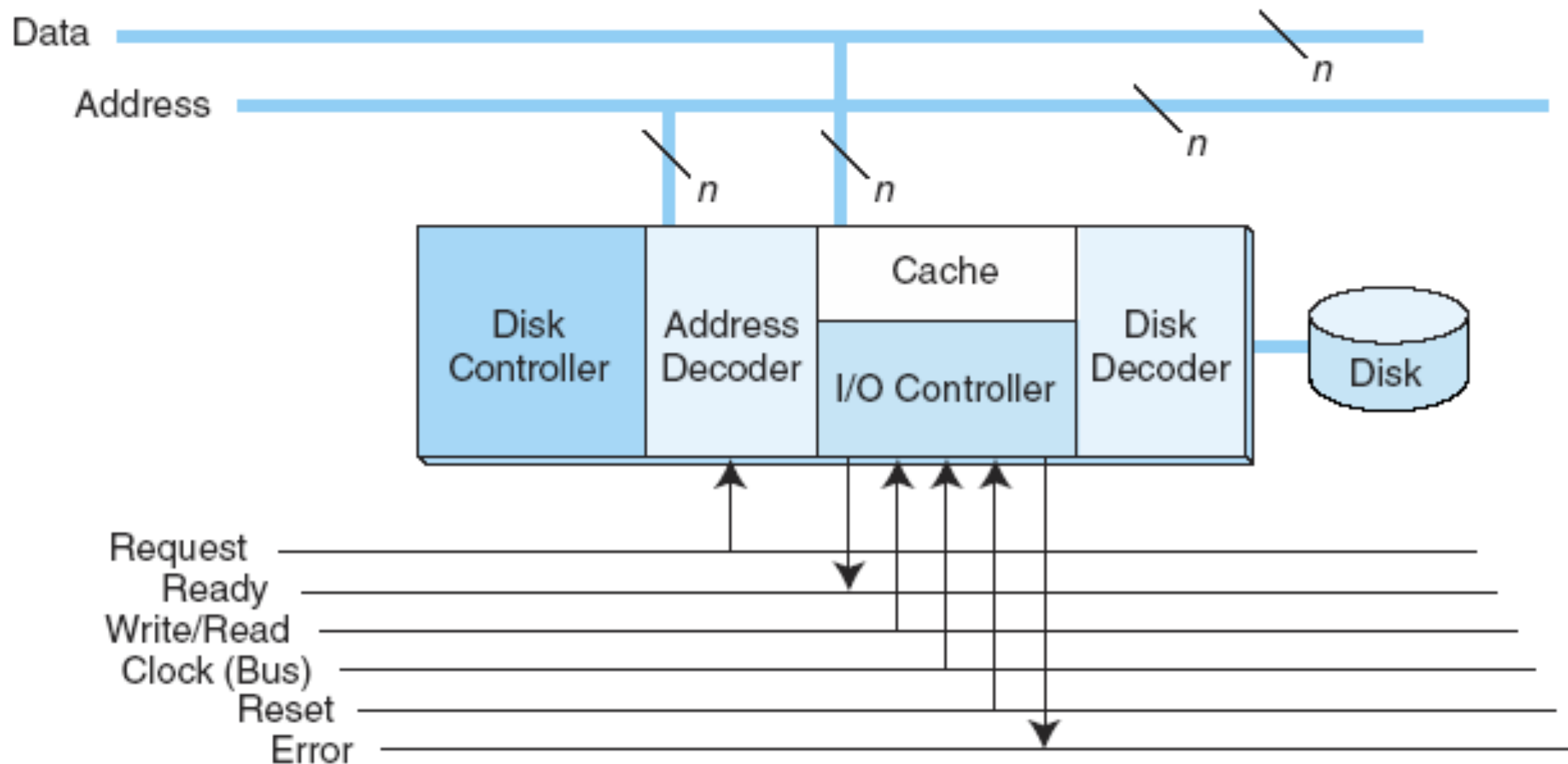


Figure 2. Pin Configurations

Programmed Input/Output (PIO)



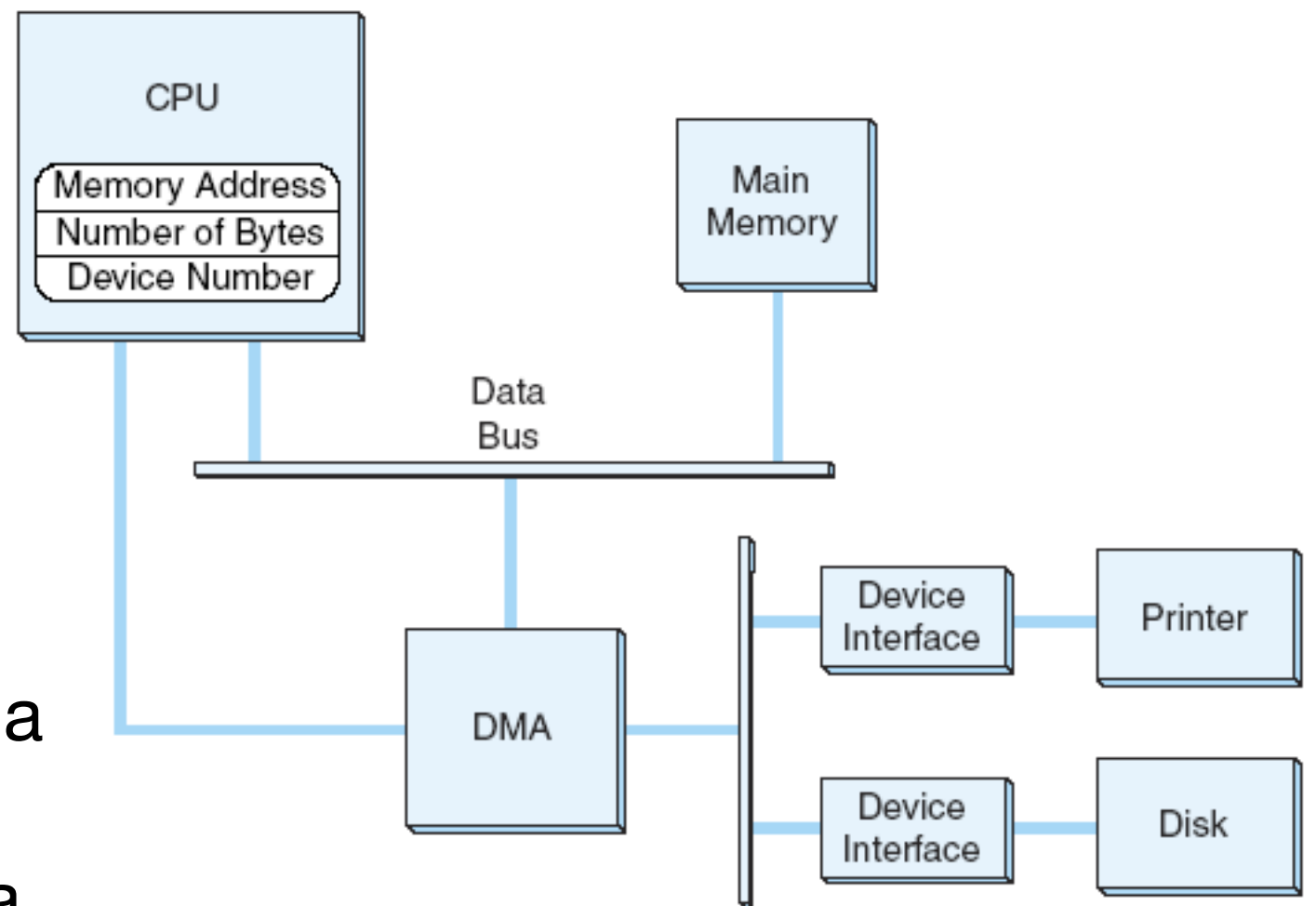
Programmed Input/Output (PIO)

- ➤ Esquema de acceso mediante instrucciones de I/O del procesador
- ➤ Dependiendo de la implementación se consiguen diferentes velocidades de acceso que se conocen como PIO Modes
- ➤ Utilizada en los primeros discos rígidos de tecnología IDE/ATA. Estos PIO modes fueron incluidos en las respectivas especificaciones de los estándares ATA-1 y ATA-2
- ➤ La CPU gestione la transferencia de la información

PIO Mode	Cycle Time (nanoseconds)	Maximum Transfer Rate (MB/s)	Defining Standard
Mode 0	600	3.3	ATA
Mode 1	383	5.2	ATA
Mode 2	240	8.3	ATA
Mode 3	180	11.1	ATA-2
Mode 4	120	16.7	ATA-2

Acceso directo a memoria (DMA)

- 1.El CPU configura el DMAC
- 2.El CPU otorga el control de la transferencia al DMAC y continúa con la próxima tarea
- 3.El DMAC realiza la transferencia de datos
- 4.Cuando finaliza la transferencia, el DMAC solicita interrupción al CPU
- 5.El CPU atiende la interrupción del DMAC



Programmable DMA Controller

Intel 8237A

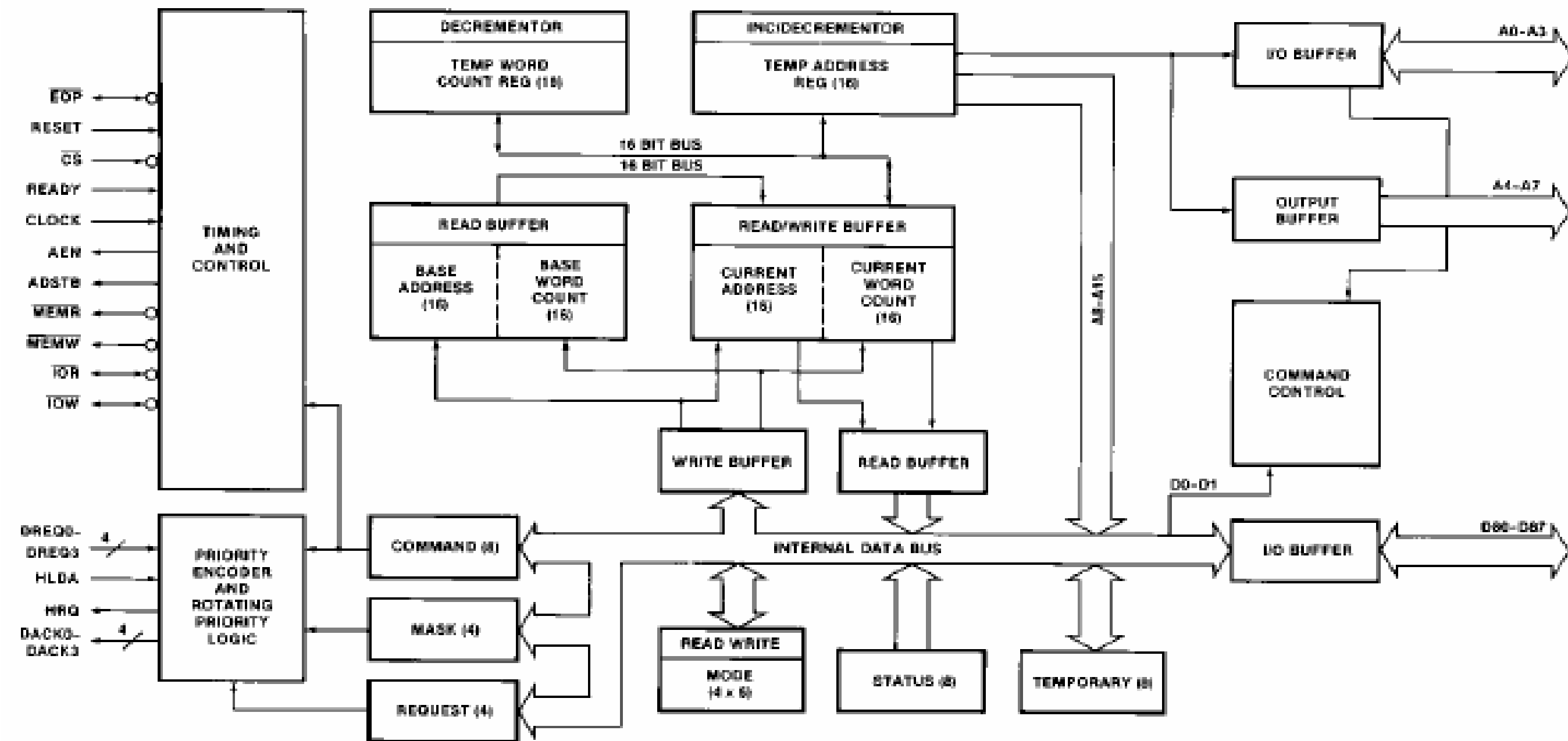


8237A HIGH PERFORMANCE PROGRAMMABLE DMA CONTROLLER (8237A-5)

- Enable/Disable Control of Individual DMA Requests
- Four Independent DMA Channels
- Independent Autoinitialization of All Channels
- Memory-to-Memory Transfers
- Memory Block Initialization
- Address Increment or Decrement
- High Performance: Transfers up to 1.6M Bytes/Second with 5 MHz 8237A-5
- Directly Expandable to Any Number of Channels
- End of Process Input for Terminating Transfers
- Software DMA Requests
- Independent Polarity Control for DREQ and DACK Signals
- Available in EXPRESS — Standard Temperature Range
- Available in 40-Lead Cerdip and Plastic Packages

Programmable DMA Controller

Intel 8237A

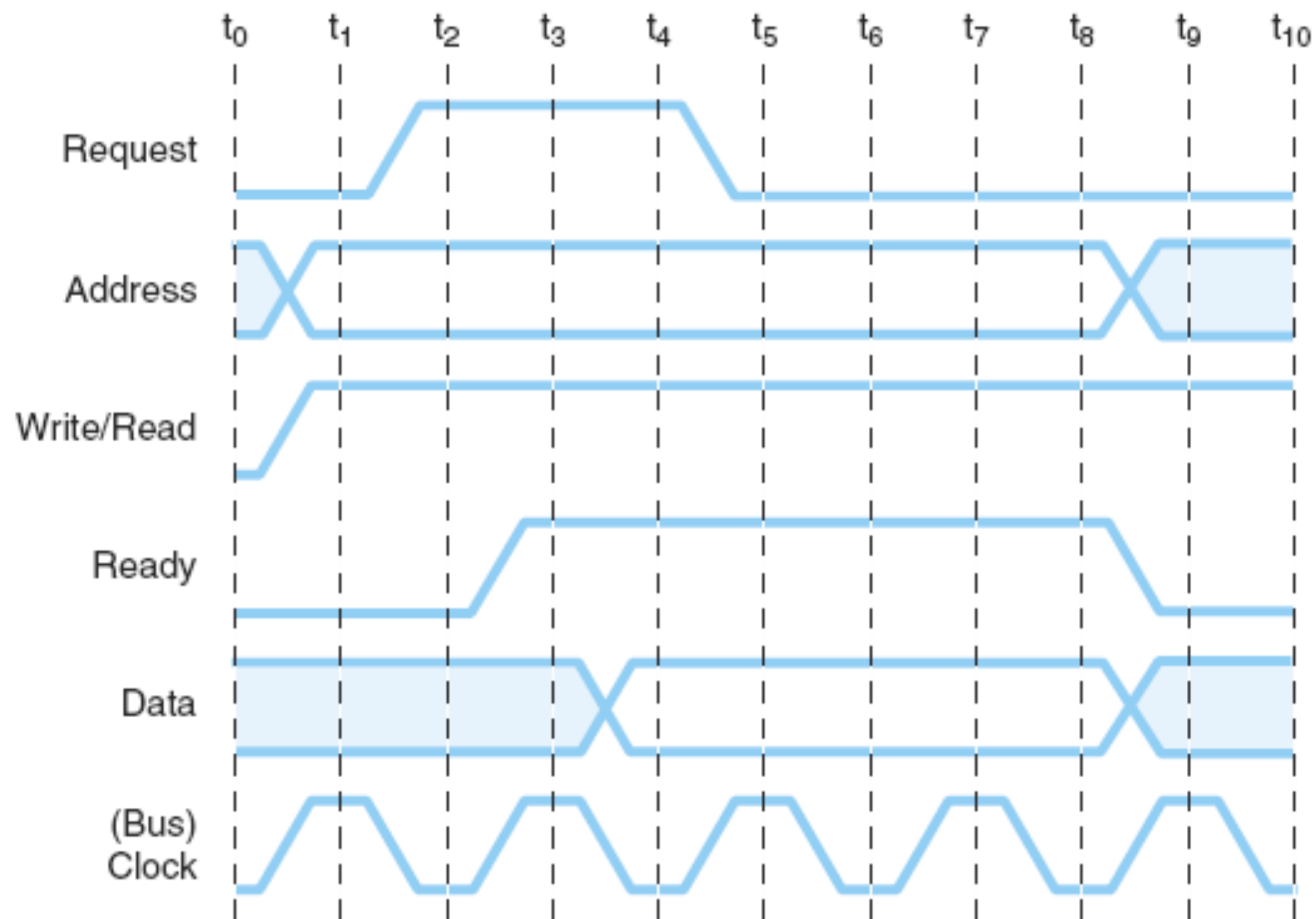


231466-1

Figure 1. Block Diagram

Programmable DMA Controller

Intel 8237A



DMA: limitaciones

- ➤ El DMA visto, introdujo mejoras en la performance de una PC con un disco rígido de 10 Mbytes
- ➤ En las máquinas modernas con transferencias de grandes volúmenes de información, el controlador de DMA 8237, se transforma en un cuello de botella
- ➤ Los chipsets que acompañan a los procesadores Core 2 Duo siguen incluyendo una lógica equivalente a este subsistema de DMA

Ultra DMA (bus mastering)

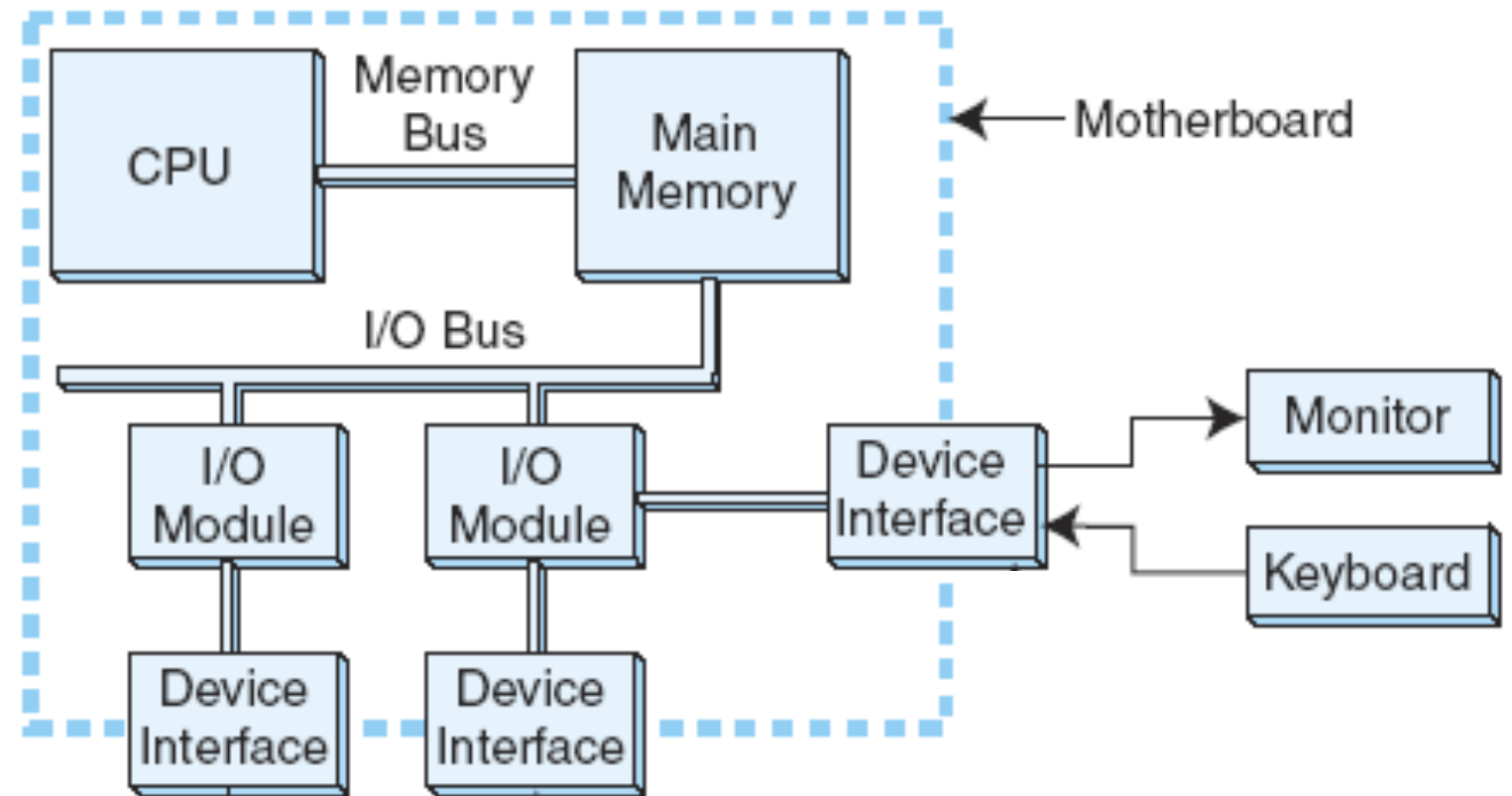
- ➤ **Ultra DMA** resuelve el problema de velocidad de DMA a partir de utilizar el bus PCI, de mayor velocidad y *double transition clocking*, se transmite tanto en el *railing edge* como en el *falling edge*
- ➤ Se encuentra clasificada por modos como PIO según la velocidad de transferencia que posibilita

Ultra DMA Mode	Cycle Time (nanoseconds)	Maximum Transfer Rate (MB/s)	Defining Standard
Mode 0	240	16.7	ATA/ATAPI-4
Mode 1	160	25.0	ATA/ATAPI-4
Mode 2	120	33.3	ATA/ATAPI-4
Mode 3	90	44.4	ATA/ATAPI-5
Mode 4	60	66.7	ATA/ATAPI-5
Mode 5 	40	100.0	ATA/ATAPI-6

Controlador de Teclado AT

Intel 8042

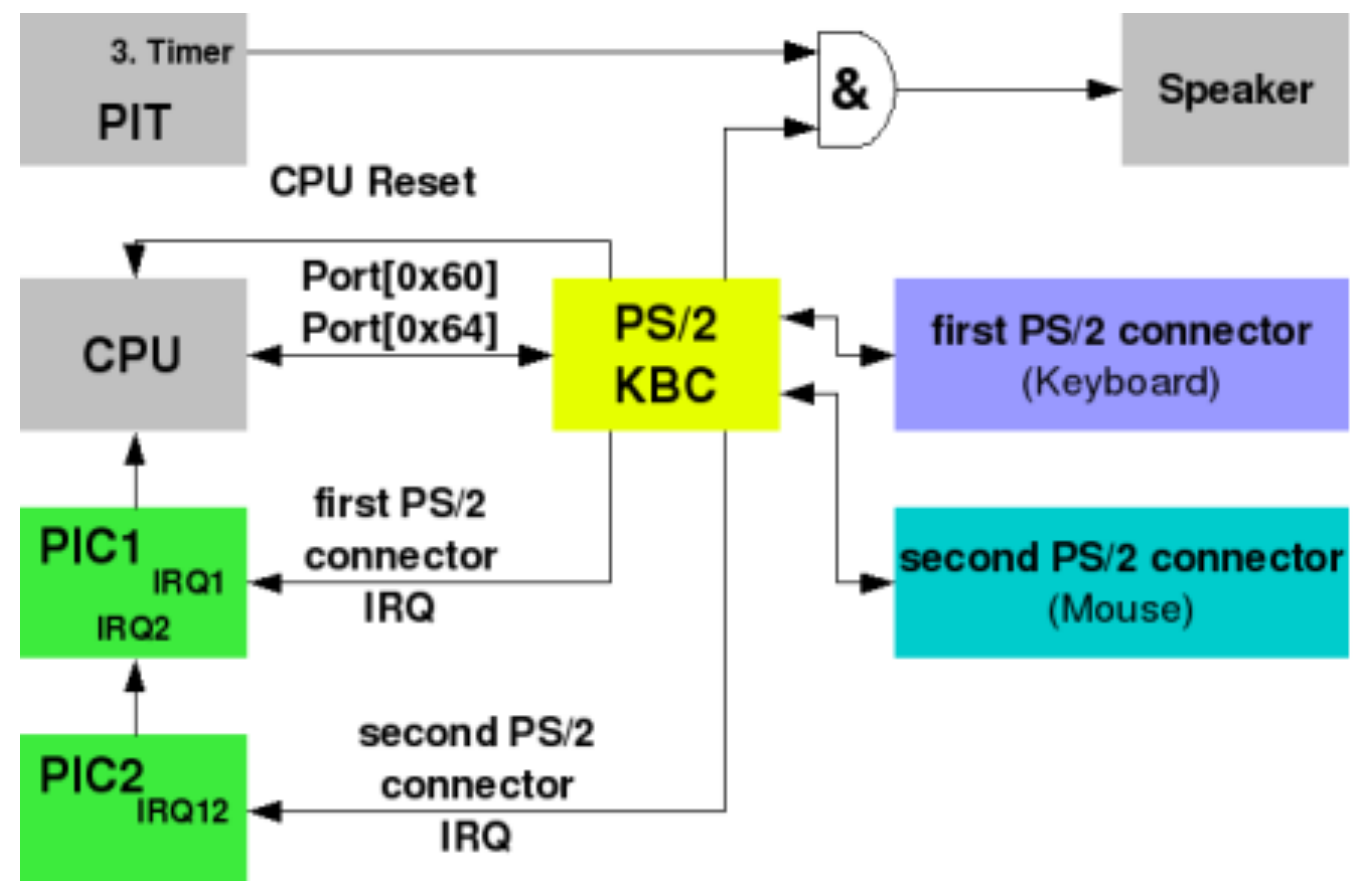
- ➤ El controlador de teclado que acompañó los procesadores Intel modernos (posterior al 8086 que usaba el Intel 8048) es el Intel 8042
- ➤ Se encuentra conectado a la señal IRQ1 del PIC8259 master
- ➤ Se accede en un espacio dedicado de I/O en las direcciones 0x60 (datos) y 0x64 (status y comandos)
- ➤ El controlador tiene tanto la capacidad de resetear el procesador como de controlar la línea A20, representación como periférico de una línea de direcciones (la número 20) para garantizar compatibilidad con el modo de direccionamiento del 8086 que solo podía direcciones 1 Mb



Controlador de Teclado AT

Intel 8042

- ➤ Si ocurre la interrupción IRQ1, al realizar un IN sobre el puerto 0x60 se obtiene un byte proveniente del teclado o dispositivo primario en el bus de datos
- ➤ Si ocurre la interrupción IRQ12, al realizar un IN sobre el puerto 0x60 se obtiene un byte proveniente del mouse o dispositivo secundario en el bus de datos
- ➤ Si se escribe con un OUT en el puerto 0x64 un comando que requiere respuesta de alguno de los dispositivos, dependiendo del firmware, puede ocurrir IRQ1, IRQ12 o ninguna cuando se pone la respuesta en en buffer. En los tres casos no se puede saber si dicha respuesta provino de los dispositivos o del controlador.



Controlador de Teclado AT

Intel 8042

smTMsc

STANDARD
MICROSYSTEMS
CORPORATION

KBD42W11

Keyboard Controller

FEATURES

- Supports IBM PC and Compatible System Designs
- Runs Much Faster Than Traditional Keyboard Controllers
- Host interface Compatible with Traditional Keyboard Controller
- 6 MHz – 12 MHz Operating Frequency
- Communicates with Keyboard Directly
- High-reliability CMOS Technology
- 40 Pin DIP and 44 Pin PLCC Package

Controlador de Teclado AT

Intel 8042

