



# ARQUITECTURA DE COMPUTADORAS

---

## Unidad N° 4

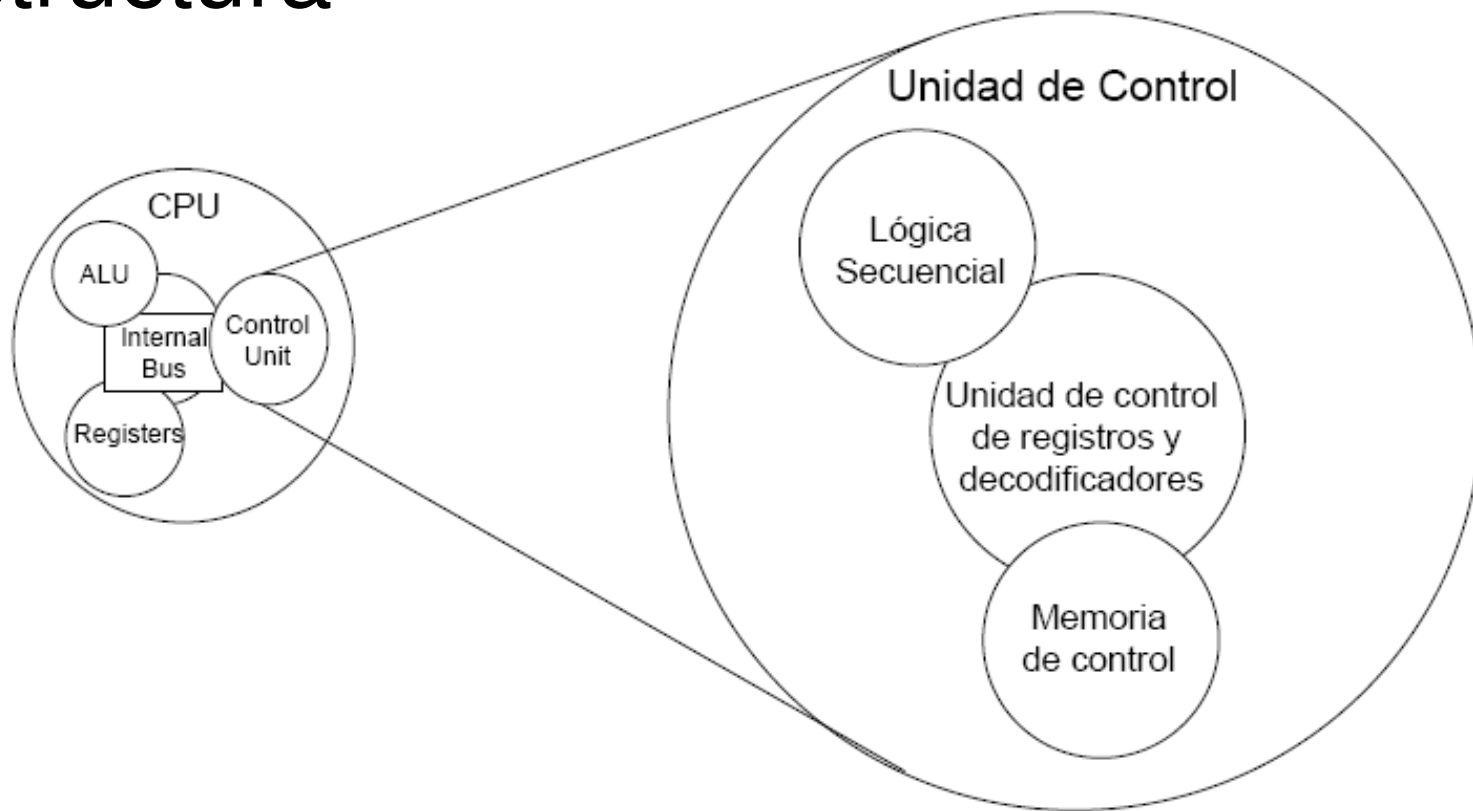
- Unidad de Control
- Buses

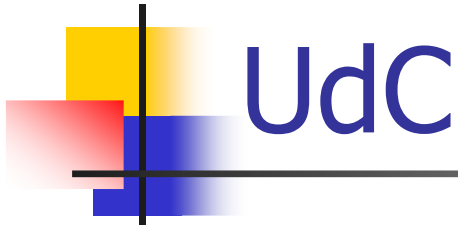
Profesor: Fabio Bruschetti

Ver 2013-01

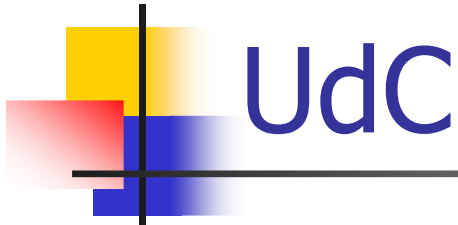
# Unidad de control (UdC o CU)

## ■ Estructura



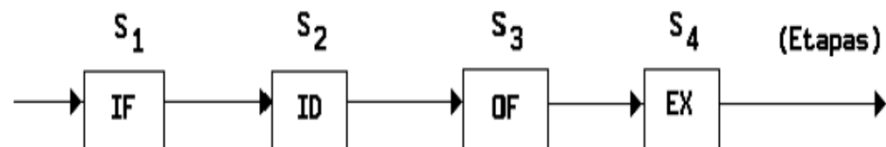


- La UDC es la esencia funcional del computador
- Será la responsable de decodificar las instrucciones, generar todas las señales eléctricas necesarias para su ejecución tanto dentro como fuera de la CPU
- En un principio, las UDC estaban internamente “cableadas”; es decir, se las construía con circuitos discretos fijos para decodificar y ejecutar instrucciones. Esto tenía como ventaja una alta velocidad de ejecución pero el inconveniente de no poder incorporar nuevas instrucciones al set original; prácticamente había que realizar un nuevo diseño.
- Actualmente no son cableadas, son microprogramadas. Estas unidades poseen una Memoria de Control donde se encuentran almacenados los Microprogramas correspondientes para cada una de las instrucciones que es capaz de ejecutar la CPU



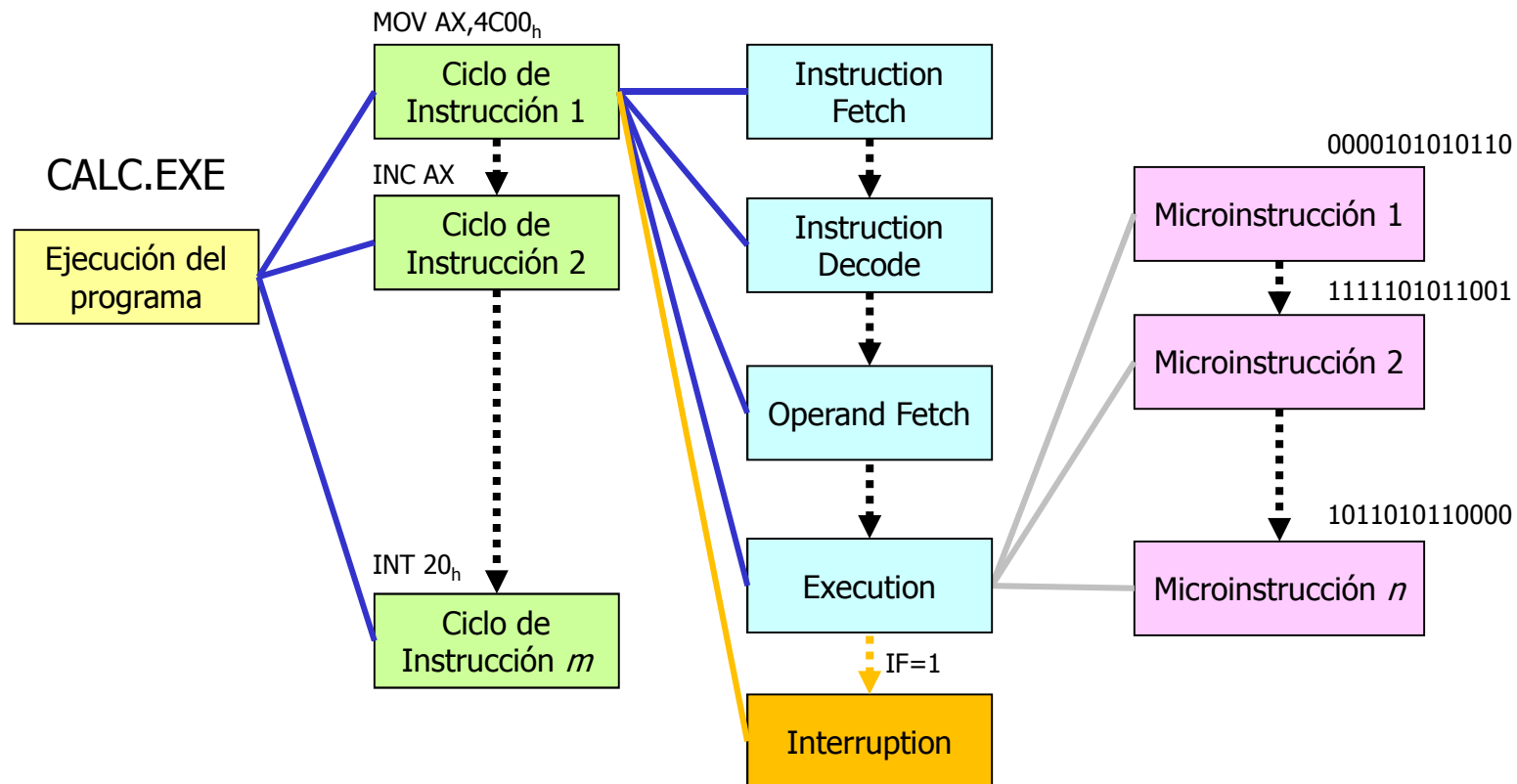
- Para cada instrucción del set, existe una secuencia de eventos necesaria para lograr su objetivo
- Cada uno de estos eventos se llama **microinstrucciones**
- El conjunto de microinstrucciones se llama **microprograma**
- Para cada instrucción, existirá un microprograma dentro de la UC, más precisamente en la Memoria de Control
- Un Microprograma es una secuencia de microinstrucciones, denominando así al conjunto de señales eléctricas necesarias para dicha ejecución
- La ejecución de dichas microinstrucciones tienen como base la señal de reloj que marca el paso de los acontecimientos

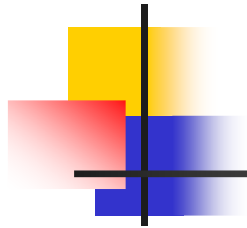
- Los “1” y “0” almacenados en la Memoria de Control se corresponderán con los valores eléctricos que tomarán las señales de control que emita la UDC
- La ventaja de las unidades de control microprogramadas es que es posible agregar nuevas instrucciones con solo almacenar los microprogramas correspondientes a la Memoria de Control
- La Memoria de Control es inaccesible al usuario y es grabada por el fabricante del Microprocesador o por un proceso especial que el fabricante pone a disposición



# UdC – Ciclo de Instrucción

- Un programa está compuesto por instrucciones
- Cada instrucción se compone de diferentes ciclos
- Cada ciclo se compone de microinstrucciones





# UdC – Ciclo de Instrucción

- Instruction Fetch (Búsqueda del código de operación “COP”)
  - Sucede al inicio de cada ciclo de instrucción
  - Consiste en leer de la MP la siguiente instrucción
- Instruction Decode
  - Se decodifica el código de operación y se determina la necesidad de operandos
- Operand Fetch
  - Lectura de uno o más operandos desde la MP
- Execution
  - La ALU realizará la operación indicada por el COP (ADD, SUB, MUL, NOT, etc...)
  - Se almacenará el resultado en el registro o posición de la MP según indique la instrucción



# UdC – Ciclo de Interrupción

---

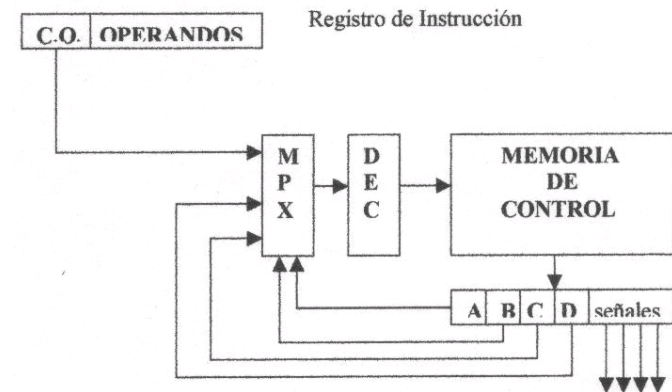
- Como las CPUs están diseñadas para ser interrumpidas durante la ejecución de sus programas (y por consiguiente en cada una de sus instrucciones), existen mecanismos para esta situación que generan una **interrupción**
- Al suceder esto, la CPU inicia un ciclo de interrupción, el que consiste en la ejecución de un programa (otro diferente al corriente) que atiende la interrupción. Este programa está compuesto también por instrucciones
- Esto ocurrirá si la bandera de habilitación de interrupciones (Interrupt flag) está en 1 (IF=1)



# UdC – Ejecución de microinstrucciones

## ■ Ejecución de una microinstrucción

- Cuando el DEC (decodificador) apunta a una posición de la Memoria de Control (MdC), indica que esta deberá ejecutarse
- “Ejecutar” esa microoperación implica que los “0” y “1” que hay en esa posición de la MdC, aparecerán como señales eléctricas que actuarán sobre los circuitos digitales de la CPU
- Estas señales activarán las conexiones y desconexiones necesarias para que, en el siguiente pulso de CLK, la información fluya de un dispositivo a otro según lo requerido



# UdC – Paralelismo

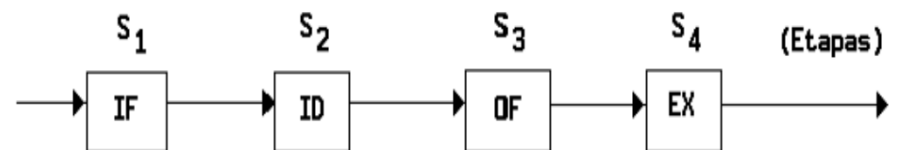
- Paralelismo Temporal y Espacial
  - Espacial: Existen varias unidades funcionales (simultaneidad)
  - Temporal: Se solapan tiempos.
- Ciclo de ejecución

- IF = Instruction Fetch

- ID = Instruction Decode

- OF = Operand Fetch

- EX = Instruction Execution



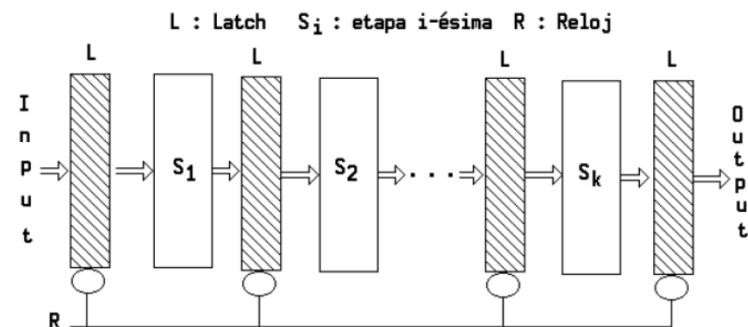
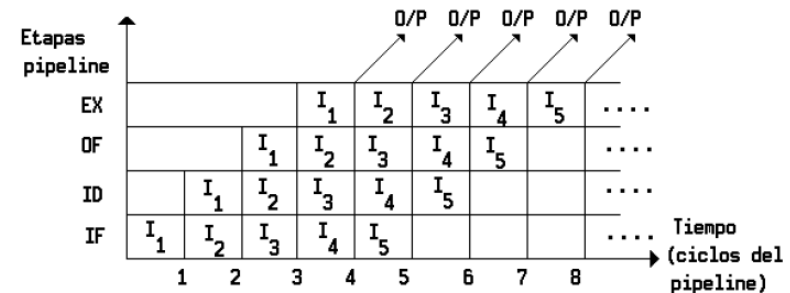
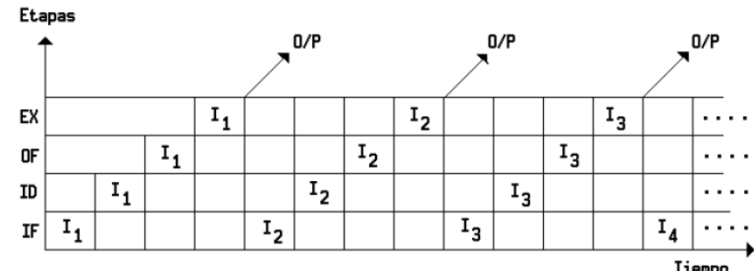
# UdC – Paralelismo

## ■ Pipeline

- En la segmentación de instrucciones, cada etapa o segmento de la cadena está especializada en una tarea específica de la “línea de ejecución” y lleva a cabo siempre la misma actividad
- Esta tecnología es propia de procesadores eficientes

## ■ Tipos

- Aritméticos
- de Instrucción
- de Procesador
- Uni/Multifuncionales
- Estáticos/Dinámicos
- Escalares/Vectoriales





# UdC – Problemas en los Pipelines

- Problemas de datos → Forwarding
  - Un dato que es modificado, es leído inmediatamente luego. Como la modificación puede no haber finalizado, la lectura puede ser errónea
    - $i1.R2 = R1 + R3$
    - $i2.R4 = R2 + R3$
- Problemas de control → Predictores de saltos
  - Ocurren cuando una instrucción de salto deberá ejecutarse pero no se puede saber si la condición de salto será satisfactoria con lo cual la próxima instrucción a ejecutar no es la que físicamente está a continuación del salto
    - $i1.A = B - C$
    - $i2.JMP\ A = 0\ to\ i.n$
    - $i3.D = A \times B$



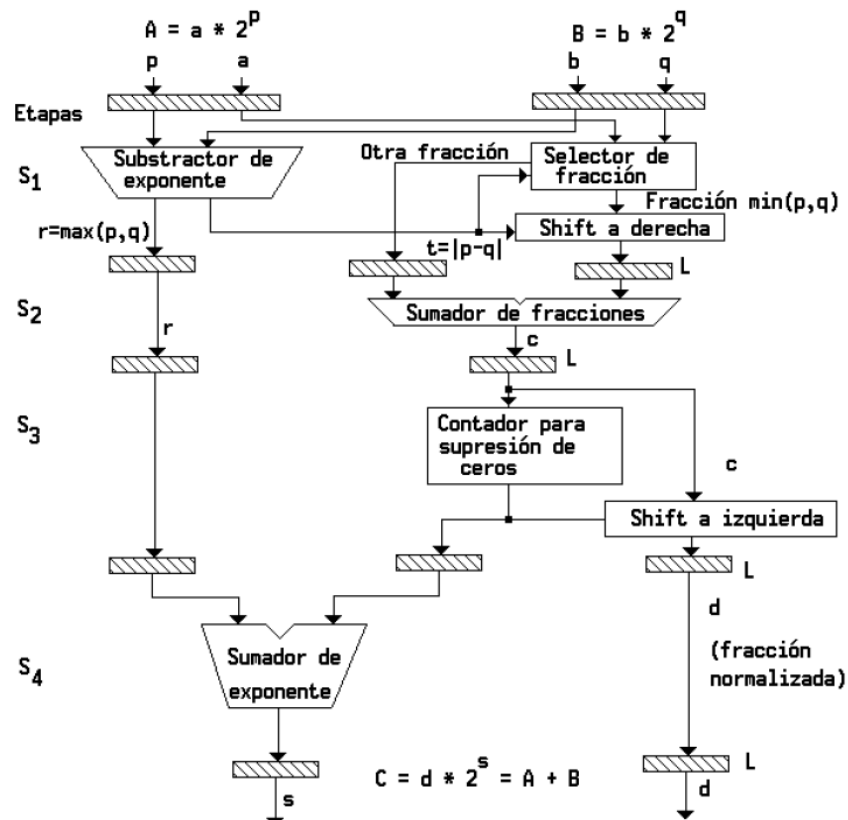
# UdC – Problemas en los Pipelines

---

- Problemas de estructura → Más unidades estructurales
  - Ocurren cuando dos instrucciones requieren una misma unidad estructural (por ejemplo la ALU)
    - i1.JMP A = 0 → Requiere la ALU para comparar
    - i2.CMP A,B → Requiere la ALU para comparar

# UdC – Ejemplos de Paralelismo

- Pipeline Sumador de Punto Flotante en 4 etapas

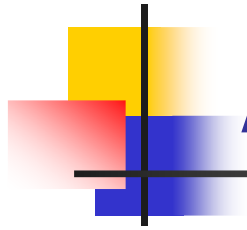


Un sumador pipeline de punto flotante de cuatro etapas de procesamiento.



# Arquitecturas RISC y CISC

- RISC (**Reduced Instruction Set Computing**)
  - Conjunto de instrucciones simples que hacen menos cosas en poco tiempo
  - Arquitectura LOAD-STORE
  - Alpha, ARC, MIPS, PA-RISC, PIC, Power Architecture (que incluye el PowerPC), SuperH, SPARC
  - Basados en el modelo de Harvard
  - Intel Core 2 y AMD K6 tienen unidades de ejecución internas del tipo RISC
- CISC (**Complex Instruction Set Computing**)
  - Cada instrucción puede requerir ejecutar una o varias instrucciones de bajo nivel (load, calculate y store en una misma instrucción)
  - IBM System/360, PDP-11, VAX, 68000, y familia x86
  - Otorga muy buena productividad en Assembler, Fortran, Algol



# Arquitecturas RISC y CISC

Principales características	CISC	RISC
Cantidad de instrucciones en lenguaje máquina	Muchas	Pocas
Cantidad de modos de direccionamiento	Muchos	Pocos
Cantidad de formatos de instrucción	Varios	Unico
Cantidad de ciclos de reloj necesarios para ejecutar cada instrucción	Muchas, más de uno	Uno
Instrucciones para acceder a la memoria	Muchas	2, Load y Store
Registros de propósito específico	Si	No

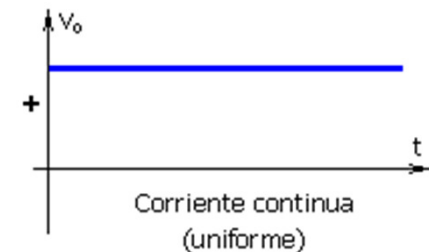


# Buses – Señales

## ■ Señales

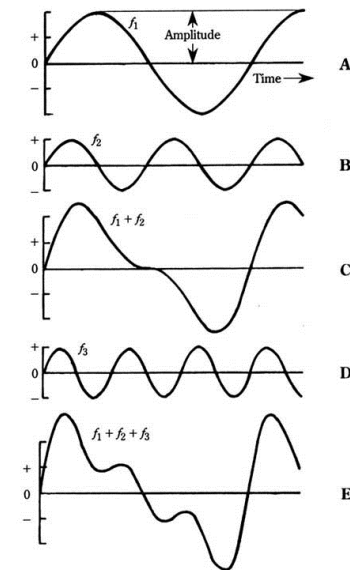
### ■ Continuas

- El valor de la señal no cambia con el tiempo, es constante



### ■ Alternadas o alternas periódicas

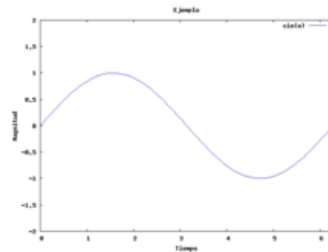
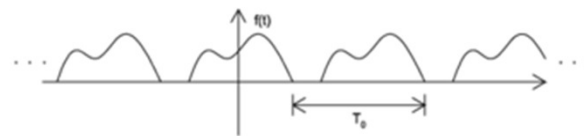
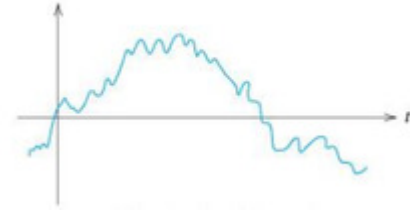
- El valor de la señal depende del tiempo
- Frecuencia = cantidad de ciclos por unidad de tiempo (seg). Se mide en Hertz (Hz) y:
  - 1 Hz = 1 ciclo / seg.
  - Período = 1 / Frecuencia
  - $V_p$  = Valor pico
  - $V_m$  = Valor medio



# Buses – Señales

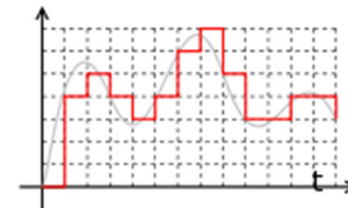
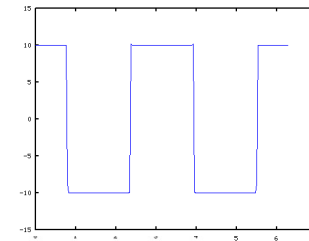
## ■ Analógicas

- Representables por una función matemática continua
- Ondas sonoras, luz



## ■ Digitales

- Pueden tomar un conjunto de valores discreto únicamente
- Código morse, interruptor





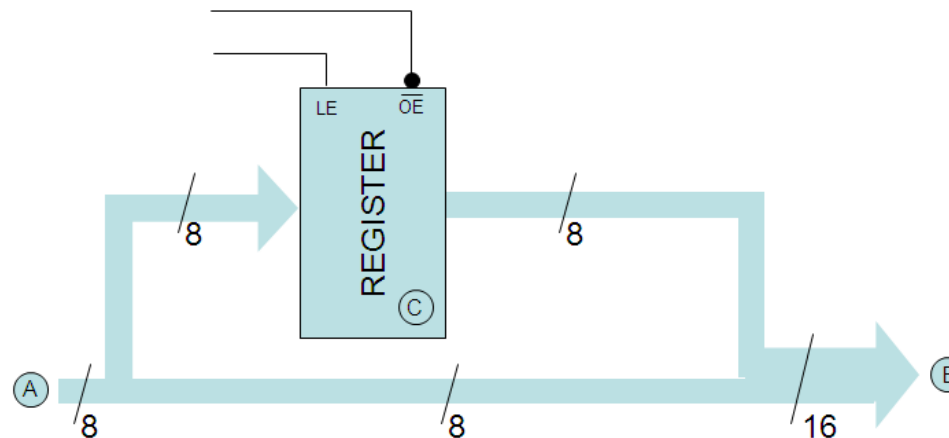
# Buses

---

- Dentro de un computador, la CPU, Memoria y E/S se comunican entre sí mediante varias vías de conexión
- Al grupo de vías que realiza esta conexión se lo denomina Sistema de Interconexión o Buses
- Los buses también pueden interconectar computadores entre sí (USB)
- Interconexión de Bus
  - Conecta dos o más dispositivos.
  - Transmite información en modo compartido
  - Todos los dispositivos reciben la misma información del bus
  - Un dispositivo a la vez puede transmitir información con éxito
  - Está compuesto por múltiples líneas de comunicación
  - Está regido por normas o reglas de conexión

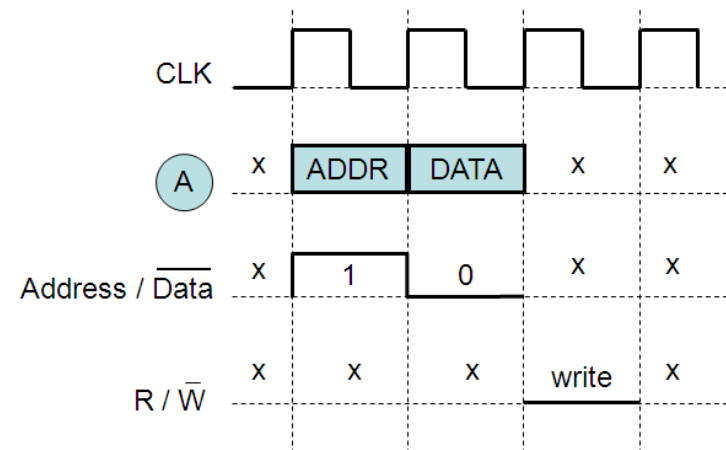
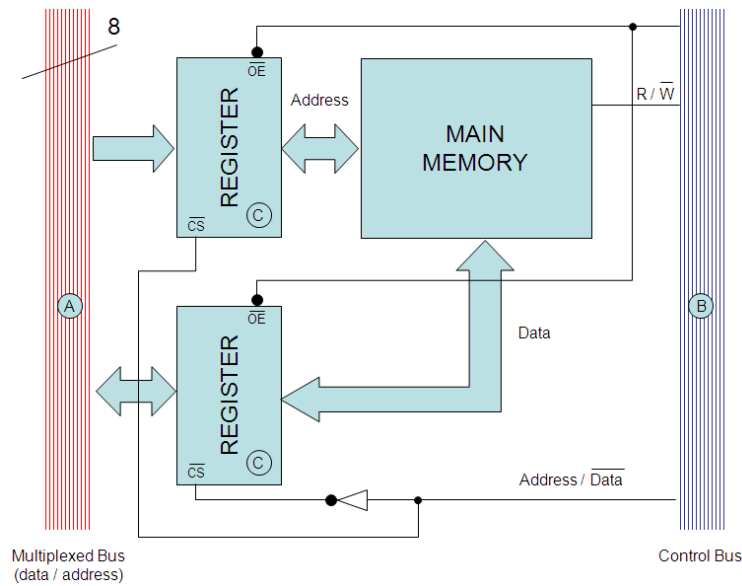
# Buses – Propiedades

- **Ciclo:** Llamaremos ciclo de bus a la operación básica del bus en que se realiza una transferencia elemental completa de información entre dos dispositivos conectados al bus
  - **Bus de ciclo completo:** Si la transferencia de información se hace sin interrupciones
  - **Bus de ciclo partido:** el tiempo del bus se divide en pequeños períodos (*time slots*) en donde se manda parte de los datos en cada uno de ellos



# Buses – Propiedades

- **Paralelismo:** Generalmente se transmiten todos los datos o (bits) al mismo tiempo. Si el bus es menos ancho que la palabra a transmitir, se usarán buses multiplexados o de ciclo partido, es decir, se transfiere la palabra en partes





# Buses – Propiedades

---

- **Ancho:** Número de líneas (bits) que se utilizan para transmitir una palabra
- **Banda base:** La información se envía si necesidad de una señal portadora
- **Sincronización:** Si hay una única señal de reloj (CLK) que regula la transmisión de información, diremos que el bus es sincrónico. Cuando cada dispositivo regula la sincronización según sus necesidades, será un bus asincrónico.
- **Velocidad:** se referirá a la velocidad de transmisión del medio físico empleado para realizar las conexiones entre los dispositivos (cobre, plata, etc.)



# Buses – Propiedades

---

- **Capacidad de conexión:** Dado por el máximo número de dispositivos que pueden conectarse al bus
- **Aplicación:** se los diferenciará según su uso, como ser el bus de un canal DMA, SATA, AGP, etc.
- **Soporte:** materiales físicos utilizados para soportar y realizar las conexiones físicas: circuito impreso, cables, aislamiento, etc.
- **Función:** Se los clasifica en tres grupos: Control (Control), Direcciones (Address) y Datos (Data)



# Buses – Niveles de especificación

---

- **Mecánico:** cantidad de hilos, conectores, etc.
- **Eléctrico:** circuito equivalente, niveles de tensiones, etc.
- **Lógico:** define la equivalencia lógica de los valores eléctricos
- **Temporización básica:** cronograma temporal de un ciclo completo del bus





# Buses de un computador

---

- ***Bus de Datos (Data bus)***: Contiene las líneas necesarias para mover los datos entre los diferentes módulos del sistema. Posee un ancho de 8, 16, 32, 64 bit y aún mayores (longitud de palabra)
- ***Bus de Direcciones (Address bus)***: La información que contiene designa el origen y el destino de los datos presentes en el bus de datos. Su anchura determina la máxima capacidad de direccionamiento, es decir, si su ancho es  $n$  su capacidad de direccionamiento será  $2^n$ .



# Buses de un computador

- ***Bus de Control (Control bus)***: Utilizado para coordinar y controlar el acceso a los módulos y la transferencia de información entre ellos. Provee la temporización y el control de validez de las operaciones que se realizan
- Las señales más comunes son:
  - $R/\overline{W}$  (read / write) = lectura / escritura en un dispositivo
  - $IO/\overline{M}$  (input-output / memory) = acceso a E/S o Memoria
  - BR (bus request) = solicitud del bus para su uso
  - BRA (bus request acknowledge) = aceptación del pedido del bus
  - INT (interrupt) = solicitud de interrupción
  - INTA (interrupt acknowledge) = aceptación de interrupción
  - CLK (clock) = reloj del sistema
  - INIT (initialize) = reiniciar todos los módulos del sistema



# Buses – Arquitecturas

---

- Algunas arquitecturas de bus conocidas:
  - RS-232 (Recommended Standard 232) – Electronic Industries Association – 1969 – Comunicación serie entre el computador y dispositivos externos, modems, impresoras, ratones
  - ISA (Industry Standard Architecture) – IBM – 1981 – PC
  - PATA (Parallel Advanced Technology Attachment) – 1986 – Western Digital – Discos rígidos
  - MCA (Micro Channel Architecture) – IBM – 1987 – PS/2, RS600, AS/400, Sysytem/370
  - EISA (Extended Industry Standard Architecture) – “Gang of Nine” – 1988 – PC compatibles, AlphaServer, HP-9000D
  - ESDI (Enhanced Small Disk Interface) – Maxtor Corporation – 1989 – Discos rígidos
  - USB (Universal Serial Bus) - Compaq, DEC, IBM, Intel, Microsoft, NEC y Nortel – 1990 – Comunicación y alimentación de dispositivos electrónicos conectados a un computador



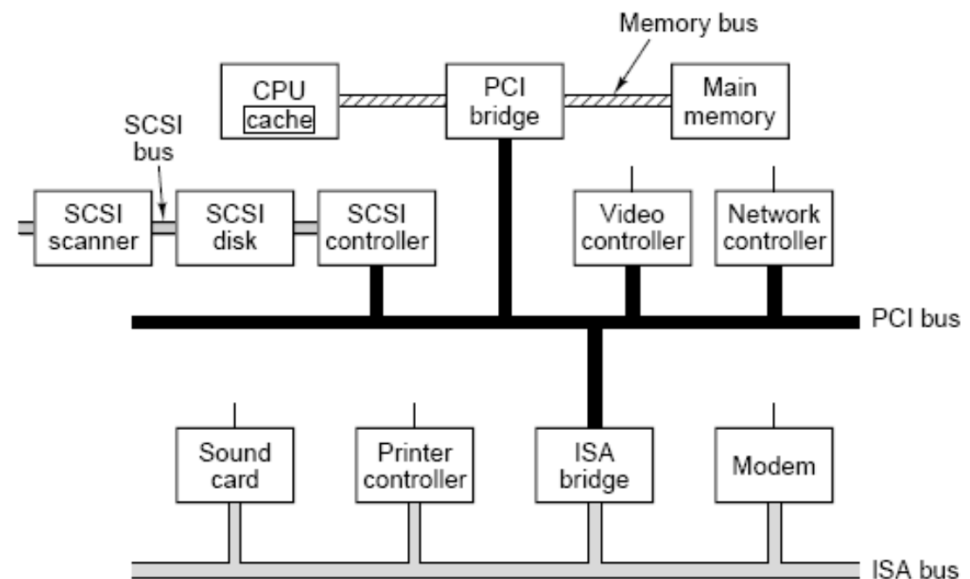
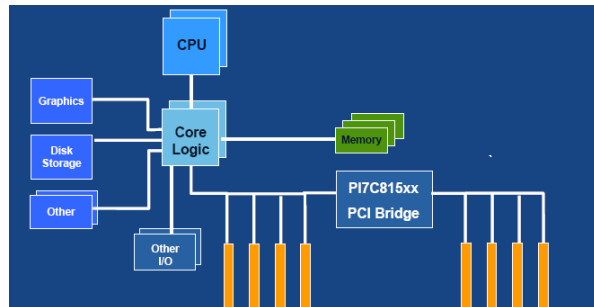
# Buses – Arquitecturas

---

- Algunas arquitecturas de bus conocidas:
  - SCSI (Small Computer System Interface) – Adaptec – 1990 – Conexión con periféricos, CDs, discos
  - VLB (Vesa Local Bus) - Video Electronics Standards Association – 1992 – I/O y DMA
  - PCI (Peripheral Component Interconnect) – Intel – 1993 – Conectar dispositivos de hardware a la computadora
  - FireWire (IEEE 1394 interface) – Apple – 1995 – Transferencia de datos en tiempo real, audio, video, aeronavegación
  - SBus (SPARC bus) – Sun Microsystems – 1997 – Interconexión CPU con periféricos
  - AGP (Accelerated Graphics Port) – Intel – 1997 – Conectar placas de vídeo a la motherboard
  - SATA (Serial Advanced Technology Attachment) – The Serial ATA International Organization – 2003 – Dispositivos de almacenamiento masivo
  - PCI-Express (Peripheral Component Interconnect Express) – 2004 – Intel, Dell, IBM y HP – Conectar periféricos montados en la motherboard

# Buses

## ■ Buses en el subsistema de Entrada-Salida



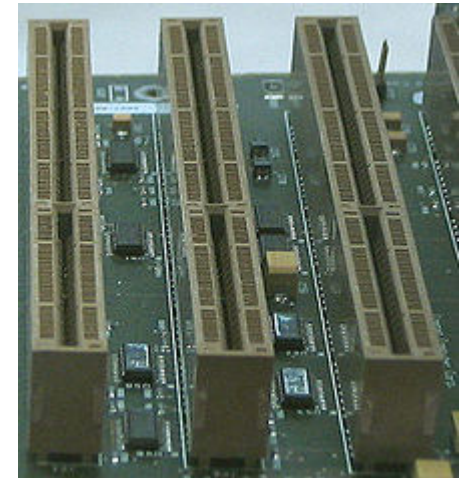
# Bus ISA (Industry Standard Architecture)

- Arquitectura de bus creada por IBM en 1981 para el procesador IBM AT (8086-8088)
- 20 bits de direccionamiento demultiplexados
- Líneas de +5V y  $\pm 12V$  para alimentación de placas
- 4 canales de DMA (Acceso Directo a Memoria (RAM, Diskette, Disco Rígido y otras). Cada canal puede ser utilizado simultáneamente por 4 dispositivos de 8 bits
- Soportaba hasta 6 dispositivos de 8 bits
- Operaba a 4, 6 u 8 MHz y en clones hasta 22 Mhz
- Fue la base del posterior bus ATA
- Actualmente se utiliza en equipos industriales
- Otra derivación del bus es la interfaz PCMCIA (Personal Computer Memory Card International Association )



# Bus EISA (Extended ISA)

- Arquitectura de bus creada por el grupo formado por AST, Compaq, Epson, Hewlett-Packard, NEC Corporation, Olivetti, Tandy, Wyse y Zenith Data Systems IBM en 1988 para PC compatibles
- Se usó hasta mediado de 1990
- Es de 32 bits con la CPU y Memoria y soporta más de un procesador conectado
- Soporta tener otros buses ISA como esclavos
- Era compatible con placas ISA
- Transmisión sincrónica de alta velocidad (33 Mb/seg)





## Bus PCI (Peripheral Component Interconnect)

---

- Arquitectura de bus creada por Intel en 1990 para familia de procesadores x86
- Espacios de direccionamiento
  - 32 bits para Memoria Principal (4 GB) – opcional 64 bits
  - 64 bits para Periféricos
  - 256 bytes para configuración de 32 dispositivos x 1 byte = 5 bits para id + 3 bits para funciones
- PCI-X
  - Extiende a 4096 bytes para configuración x 16 bits



# Bus PCI



## ■ Especificaciones

- Reloj de 33,33 MHz con transferencias sincrónicas
- Tasa de transferencia máxima de 133 MB por segundo en el bus de 32 bits ( $33,33 \text{ MHz} \times 32 \text{ bits} \div 8 \text{ bits/byte} = 133 \text{ MB/s}$ )
- Cache de datos con posibilidad de transmisión en modo de ráfagas
- Tasa de transferencia máxima de 266 MB/s en el bus de 64 bits.
- 3,3 V o 5 V, dependiendo del dispositivo

# Bus PCI – Registro

- *Device ID y Vendor ID*: identifican el fabricante
- *Subsystem ID y Subsystem Vendor ID*: identifican el dispositivo
- *Status*: características soportadas y errores ocurridos
- *Command*: conjunto de bits que activan las prestaciones
- *Cache Line Size*: sincronización con caché

31	16 15			0
Device ID		Vendor ID		00h
Status		Command		04h
Class Code			Revision ID	08h
BIST	Header Type	Lat. Timer	Cache Line S.	0Ch
Base Address Registers				10h
				14h
				18h
				1Ch
				20h
				24h
Cardbus CIS Pointer				28h
Subsystem ID		Subsystem Vendor ID		2Ch
Expansion ROM Base Address				30h
Reserved			Cap. Pointer	34h
Reserved				38h
Max Lat.	Min Gnt.	Interrupt Pin	Interrupt Line	3Ch



# Bus PCI

---

- Operación

- La BIOS o el S.O. direcciona cada uno de los dispositivos y los enumera de #0 en adelante a través de la controladora PCI
- Inicialmente, los dispositivos están inactivos. Al detectarse se configuran y se les asigna los BARs (Base Address Registers) que mantiene mientras está conectado
- Permite la función de Plug-and-play
- Permite conectar y gestionar un bus PCI secundario

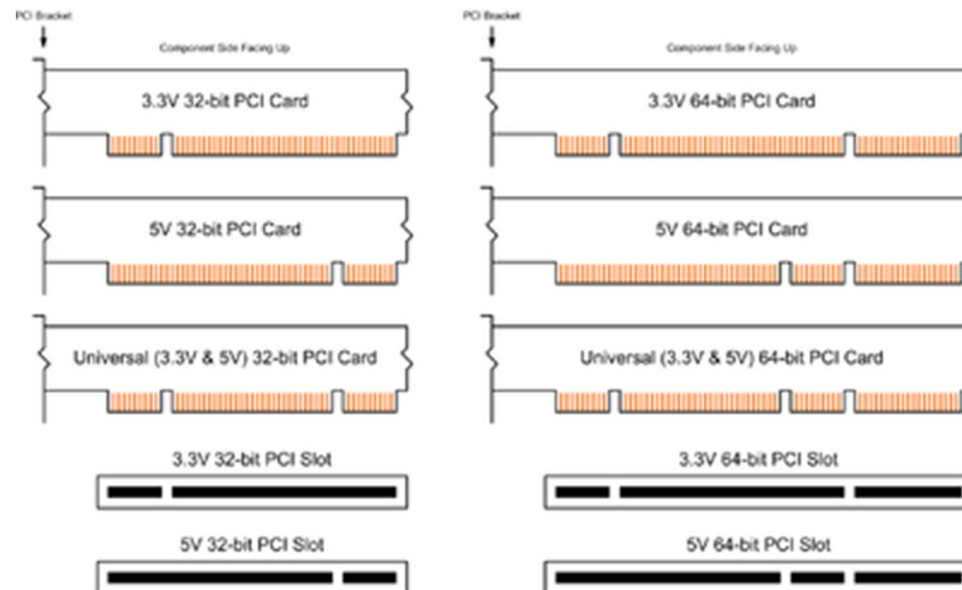
- Software de acceso

- A través de dos registros:
  - CONFIG\_ADDRESS: id del dispositivo y comando a ejecutar
  - PCI CONFIG\_DATA: datos a escribir o leer

# Bus PCI

## ■ Interrupciones

- Comparte 4 líneas de interrupciones autoconfigurables
- Versiones modernas implementan interrupciones a través de mensajes de señalización





# Bus USB (Universal Serial Bus)

- Arquitectura creada por siete empresas IBM, Intel, Northern Telecom, Compaq, Microsoft, DEC y NEC en 1996
- Permite conectar todo tipo de dispositivos electrónicos incluyendo cámaras de fotos digitales y scanners
- No hace falta detener el computador para conectar o desconectar un dispositivo (plug-and-play)
- Dispositivos no usados pasan a estado de bajo consumo
- Trafico full-duplex
- Tipos
  - 1.0: Baja velocidad. 1,5 Mbps
  - 1.1: Velocidad completa. 12 Mbps
  - 2.0: Alta velocidad. 488 Mbps
  - 3.0: Super Alta velocidad. 4,8 Gbps

# Bus USB

- Permite el funcionamiento simultáneo de 127 dispositivos
- También administra la energía a los dispositivos conectados
- El cable de interfaz puede llegar a 5 m. máximo
- Variados tipos de conectores
- Se pueden conectar:
  - Adaptadores de memorias, Cámaras de fotos, Cámaras de video, Teléfonos móviles, Disqueteras externas, Discos duros externos, Grabadoras de DVD externas, Impresoras, Ratones USB, Multifunciones, Teclados USB, MP3, MP4, Pendrives, PDA, Pedales, Sintonizadoras de TV, Volantes, Joysticks, Webcams, Tarjetas de sonido, Mini altavoces

