



UNIDAD 1

Arquitectura de Computadoras



START



MENU

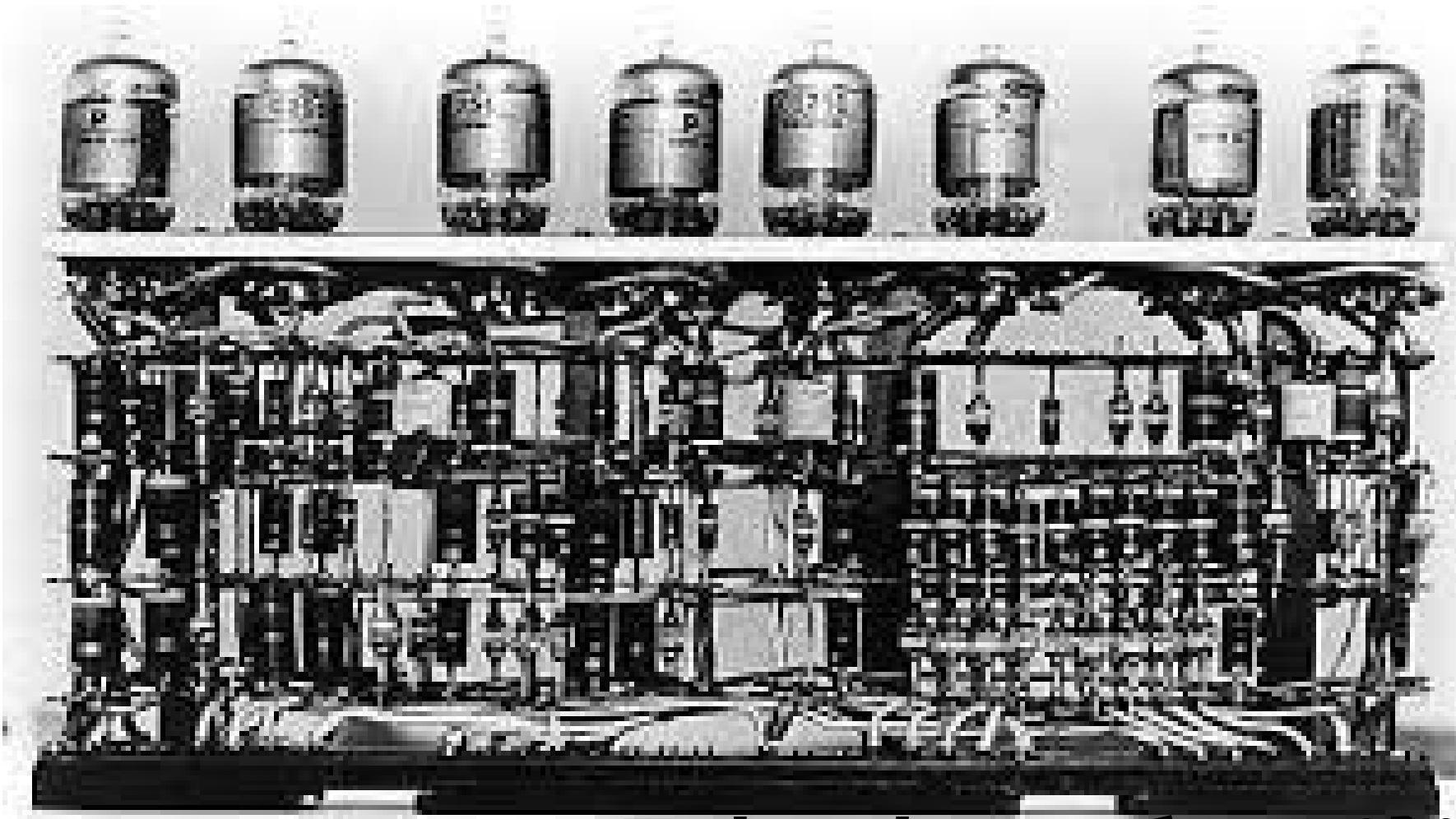
1.1

Arquitectura de Computo

- **Modelos de arquitectura
de computo**
- **Clásicos**
- **Segmentados**
- **Multiprocesamiento**

Arquitecturas Clásicas

- Estas arquitecturas se desarrollaron en las primeras computadoras electromecánicas y de tubos de vacío.
- Aún son usadas en procesadores empotrados de gama baja y son la base de la mayoría de las arquitecturas modernas.



**Tubo de vacío utilizado en
computadoras**

Arquitectura Mauchly-Eckert (Von Newman)

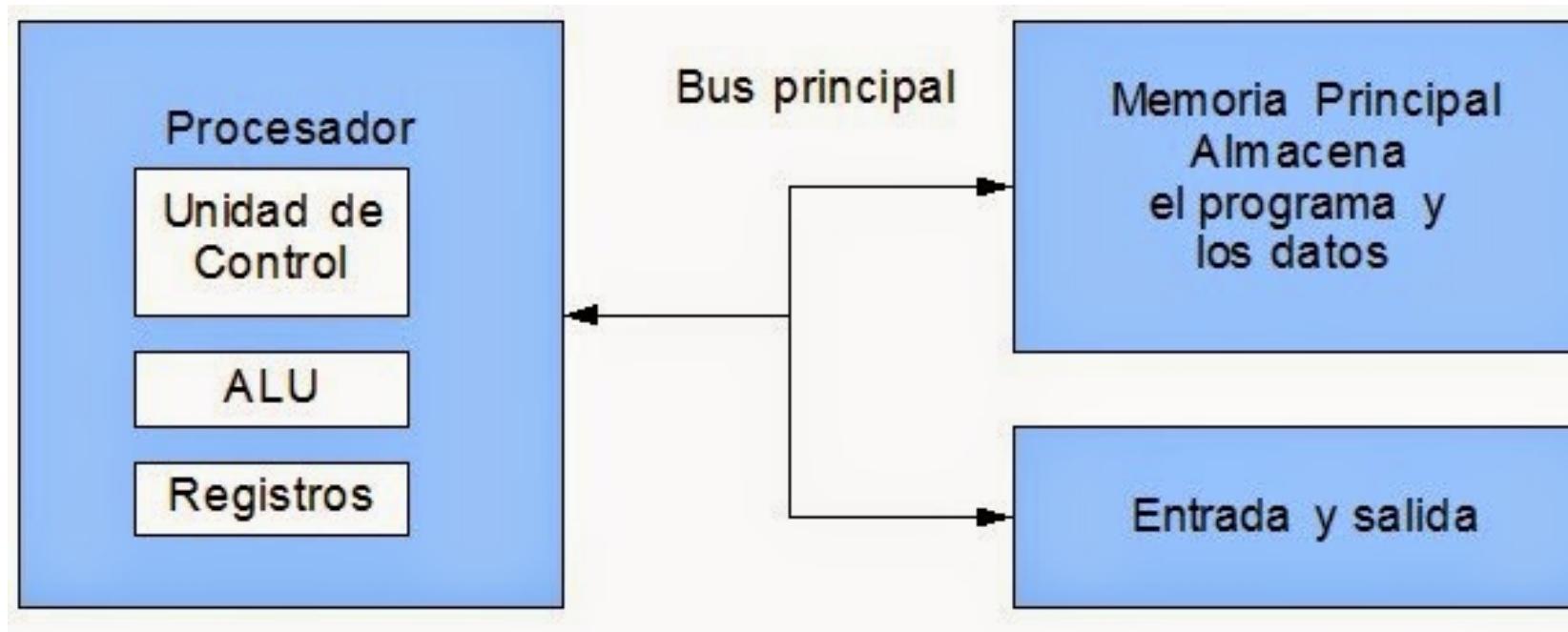
- Consiste en una unidad central de proceso que se comunica a través de un solo bus con un banco de memoria en donde se almacenan tanto los códigos de instrucción del programa, como los datos que serán procesados por este.**

Arquitectura Mauchly-Eckert (Von Newman)

Esta arquitectura es la más empleada en la actualidad ya, que es muy versátil. Ejemplo de esta versatilidad es el funcionamiento de los compiladores

Arquitectura Mauchly-Eckert (Von Newman)

Son programas que toman como entrada un archivo de texto conteniendo código fuente y generan como datos de salida, el código máquina que corresponde a dicho código fuente (Son programas que crean o modifican otros programas).



Estos datos de salida pueden ejecutarse como un programa posteriormente ya que se usa la misma memoria para datos y para el código del programa.

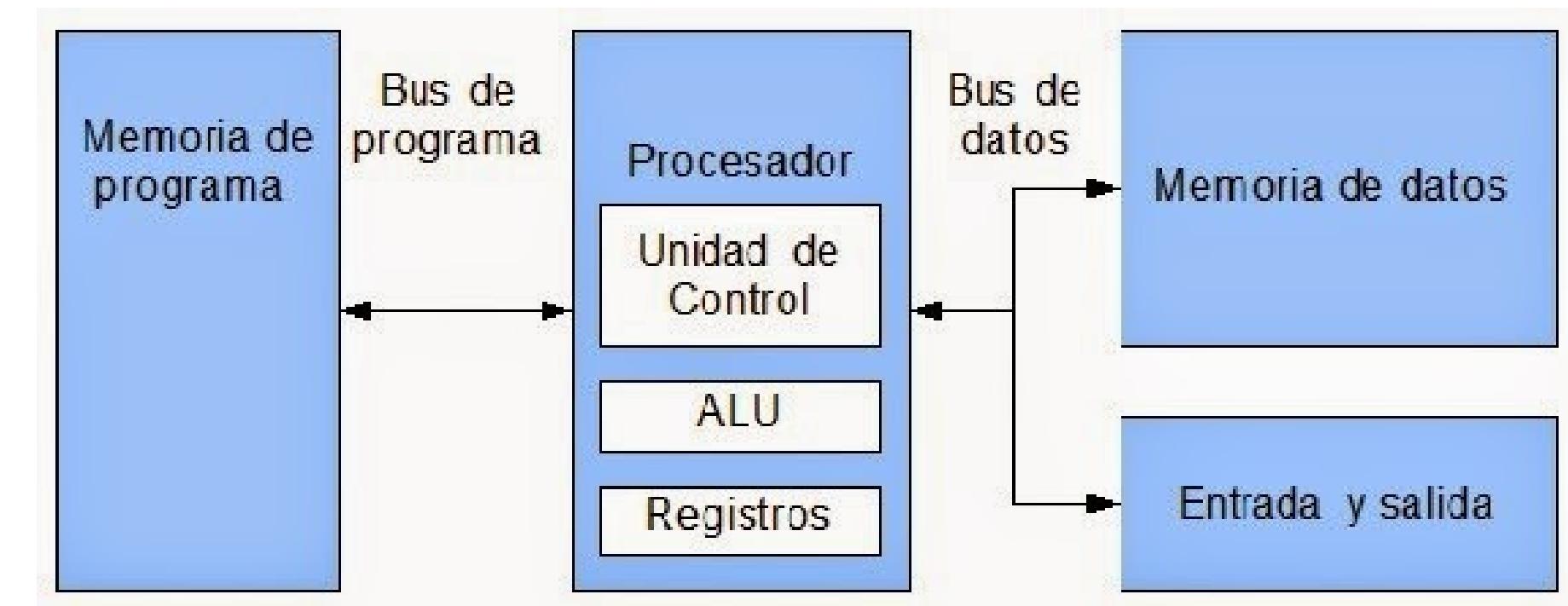
Una desventaja es que el bus de datos y direcciones único se convierte en un cuello de botella por el cual debe pasar toda la información que se lee de o se escribe a la memoria, obligando a que todos los accesos a esta sean secuenciales.

Limita el grado de paralelismo (acciones que se pueden realizar al mismo tiempo) y por lo tanto, el desempeño de la computadora.

Arquitectura Harvard

El programa se almacena como un código numérico en la memoria, pero no en el mismo espacio de memoria ni en el mismo formato que los datos.

El hecho de tener un bus separado para el programa y otro para los datos permite que se lea el código de operación de una instrucción, al mismo tiempo se lee de la memoria de datos los operandos de la instrucción previa.



Arquitecturas Segmentadas

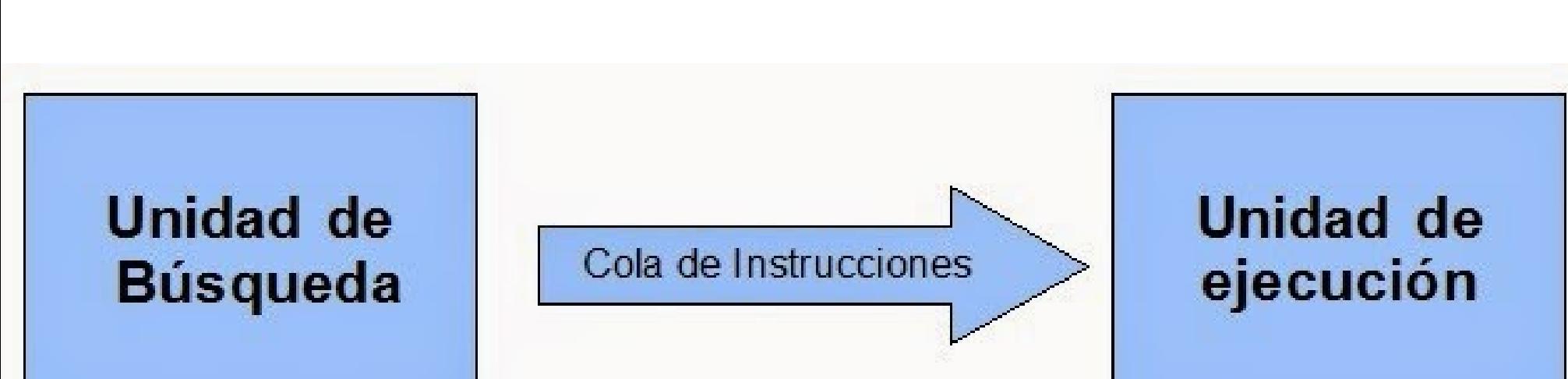
- Buscan mejorar el desempeño realizando paralelamente varias etapas del ciclo de instrucción al mismo tiempo.
- El procesador se divide en varias unidades funcionales independientes y se dividen entre ellas el procesamiento de las instrucciones.

Arquitecturas Segmentadas

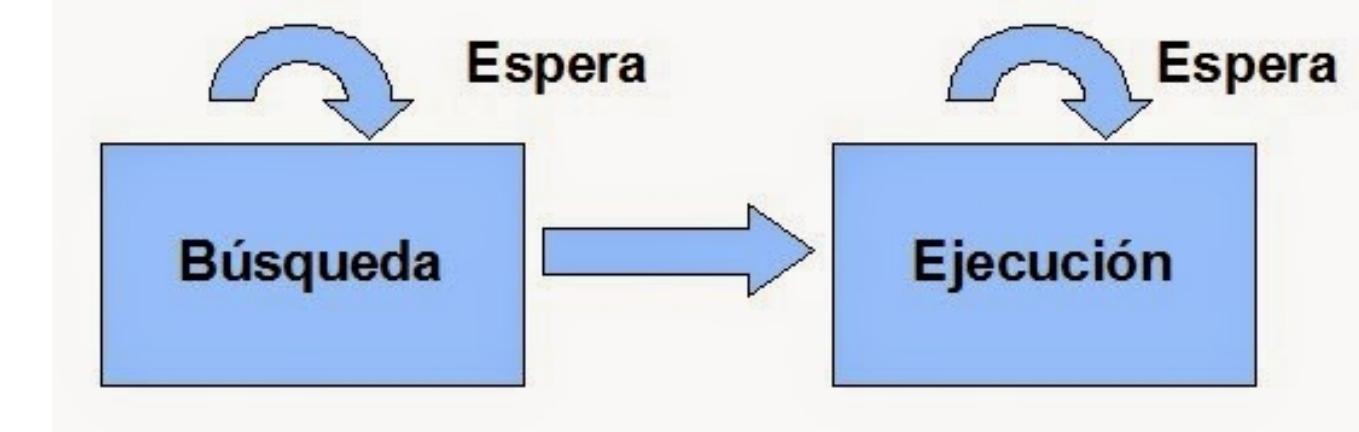
Un procesador simple tiene un ciclo de instrucción sencillo consistente solamente en una etapa de búsqueda del código de instrucción y en otra etapa de ejecución de la instrucción.

Arquitecturas Segmentadas

- Cada una de estas etapas se asigna a una unidad funcional diferente, la búsqueda a la unidad de búsqueda y la ejecución a la unidad de ejecución.
- Estas unidades se comunican por medio de una cola de instrucciones en la que la unidad de búsqueda coloca los códigos de instrucción que leyó para que la unidad de ejecución los tome de la cola y los ejecute.



La unidad de búsqueda comenzaría buscando el código de la primera instrucción en el primer ciclo de reloj. Durante el segundo ciclo de reloj, la unidad de búsqueda obtendría el código de la instrucción 2, mientras que la unidad de ejecución ejecuta la instrucción 1 y así sucesivamente.



Arquitecturas de Multi–procesamiento.

Cuando se desea incrementar el desempeño más alla de lo que permite la técnica de segmentación del cauce (límite teórico de una instrucción por ciclo de reloj), se requiere utilizar más de un procesador para la ejecución del programa de aplicación.

Arquitecturas de Multi–procesamiento.

Las CPU multiprocesamiento se clasifican en:

- SISO – (Single Instruction, Single Operand)
computadoras Monoprocesador**
- SIMO – (Single Instruction, Multiple Operand)
procesadores vectoriales, Exenciones MMX**
- MISO – (Multiple Instruction, Single Operand) No
implementado**
- MIMO – (Multiple Instruction, Multiple Operand)
sistemas SMP, Clusters, GPUs**

Arquitecturas de Multi–procesamiento.

Procesadores vectoriales: Computadoras pensadas para aplicar un mismo algoritmo numérico a una serie de datos matriciales.(Detectar el clima, reacciones químicas complejas)

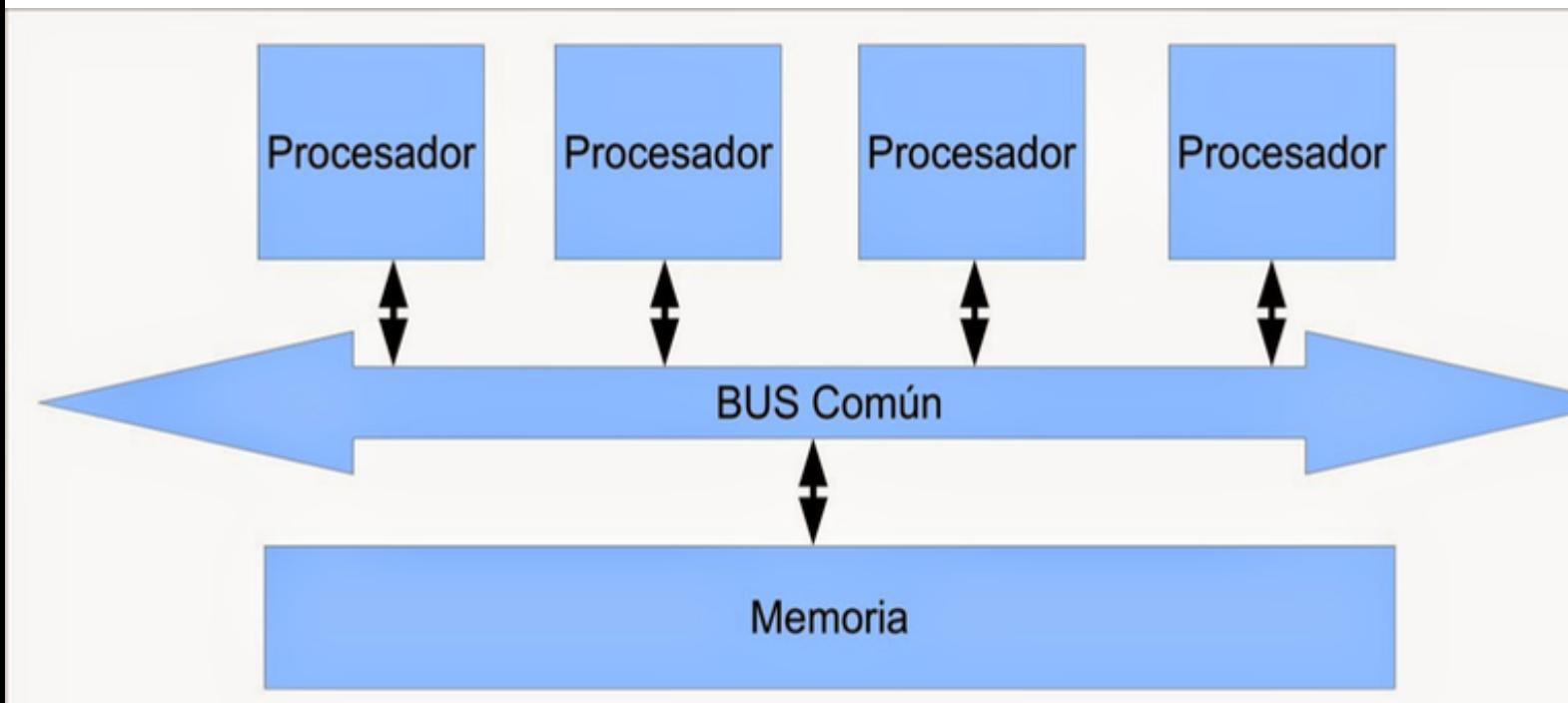
- La mayoría de los procesadores modernos incluye algunas instrucciones de tipo vectorial, tales como las extensiones al conjunto de instrucciones tales como MMX y SSE. Estas instrucciones les permiten procesar flujos multimedia más eficientemente.

Arquitecturas de Multi–procesamiento.

Los Procesadores Digitales de Señales (DSP), son procesadores especializados en el procesamiento de señales tales como audio, vídeo, radar, sonar, radio, etc.

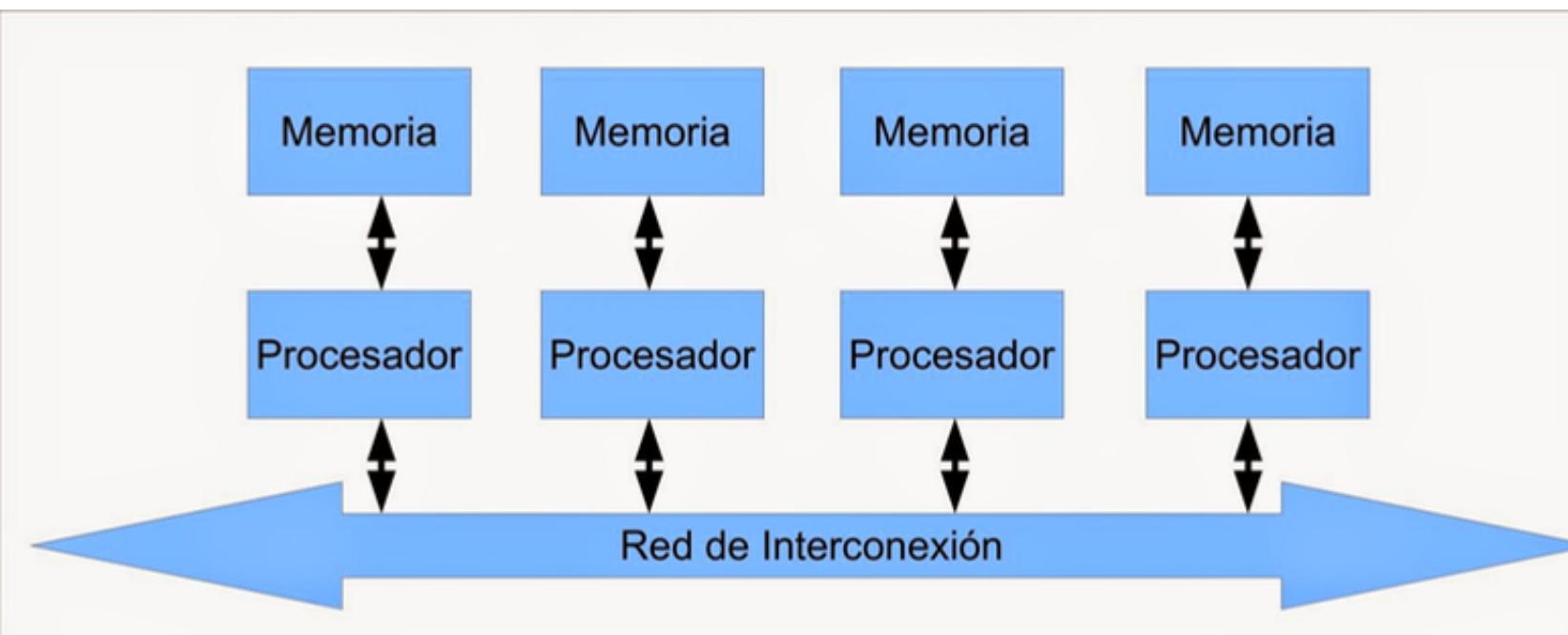
En los sistemas SMP (Simetric Multiprocesores), varios procesadores comparten la misma memoria principal y periféricos de I/O, Normalmente conectados por un bus común.

Arquitecturas de Multi–procesamiento.



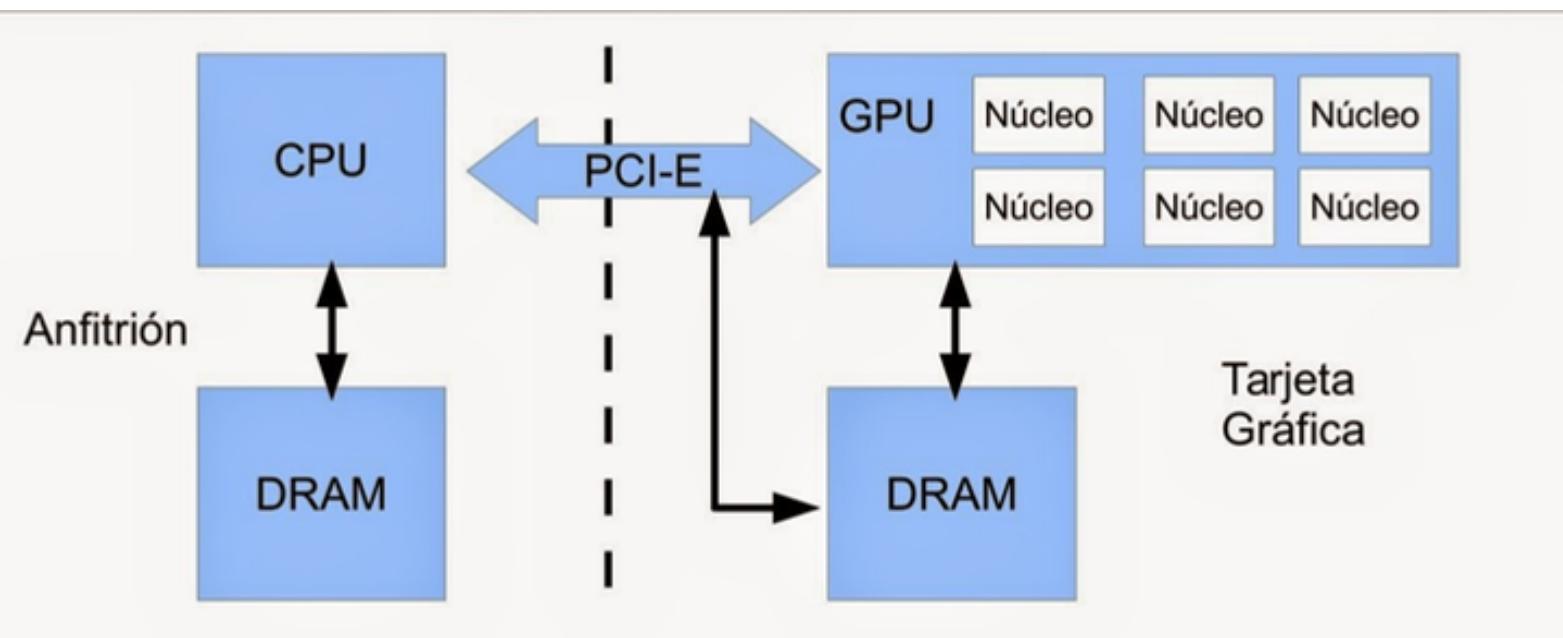
Todos tienen derechos similares en cuanto al acceso a la memoria y periféricos y ambos son administrados por el sistema operativo.

Arquitecturas de Multi–procesamiento.



Los Clústers son conjuntos de computadoras independientes conectadas en una red de área local o por un bis de interconexión y que trabajan cooperativamente para resolver un problema. Es clave en su funcionamiento contar con un sistema operativo y programas de aplicación capaces de distribuir el trabajo entre las computadoras de la red.

Arquitecturas de Multi–procesamiento.



Las unidades de procesamiento gráfico (GPU) son sistemas diseñados originalmente para el procesamiento de Gráficos, con múltiples procesadores vectoriales sencillos compartiendo la misma memoria, la cual también puede ser accedida por el CPU.



1.2.1 Unidad Central de Procesamiento

- Arquitecturas
- Tipos
- Características
- Funcionamiento
 - ALO
 - Unidad de Control
 - Registros
 - Bases

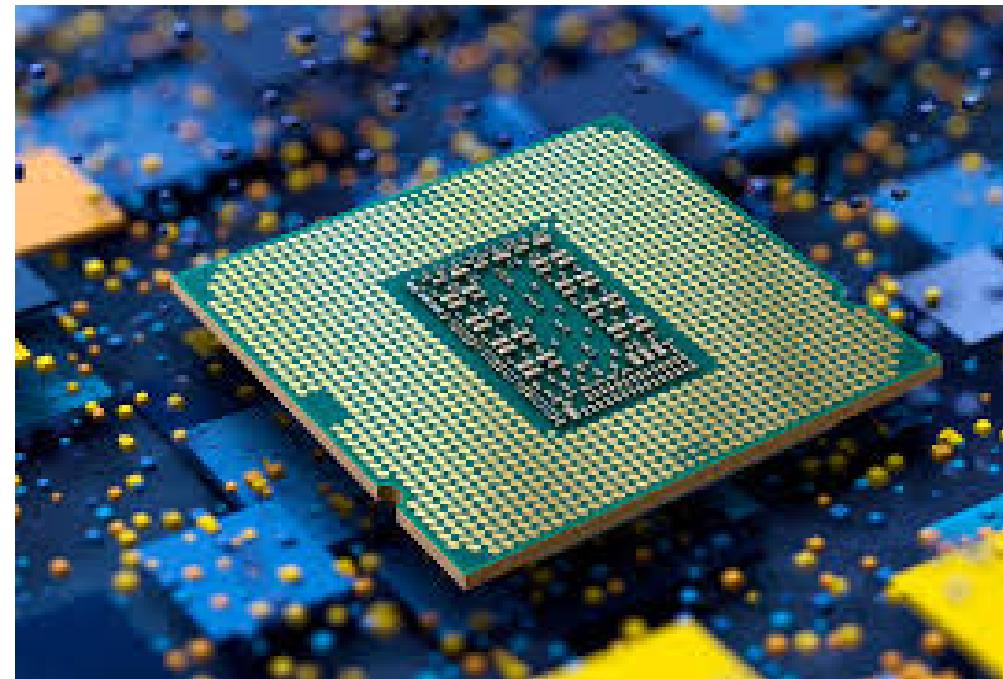
Arquitectura

Además de las arquitecturas clásicas mencionadas anteriormente, en la actualidad han aparecido arquitecturas híbridas entre la Von Neumann y la Harvard, buscando conservar la flexibilidad, pero mejorando el rendimiento. Los programas cada vez más grandes y complejos demandan mayor velocidad en el procesamiento de información como lo que implica la elección de microprocesadores más rápidos y eficientes.



Para el diseño de un microporcesador debemos de visualizar y decidir cuál será su juego de instrucciones. La decisión por dos razones, primero, el juego de instrucciones decide:

- El diseño físico del conjunto.**
- Cualquier operación que deba ejecutarse en el microporcesador deberá poder ser descrita en términos de un lenguaje de estas instrucciones.**



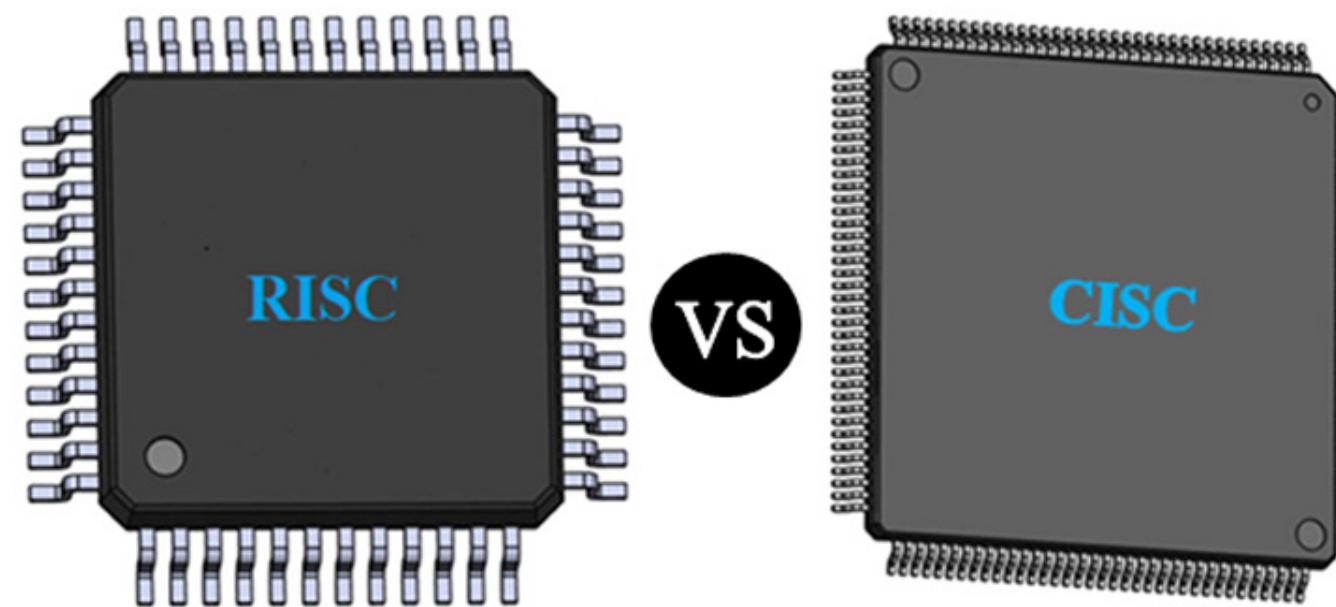
Arquitecturas CISC y RISC

CISC: Es un modelo de arquitectura en donde los microporcesadores tienen un conjunto de instrucciones que caracterizan por ser muy amplio y permitir operaciones complejas entre operandos, situadas en la memoria o en los registros internos.

Este tipo de arquitectura dificulta el paralelismo entre instrucciones, por lo que, en la actualidad como la mayoría de los sistemas CISC de alto rendimiento implementar un sistema que convierte dichas instrucciones complejas en varias instrucciones simples del tipo RISC, llamadas generalmente microinstrucciones.

Para realizar una sola instrucción un chip CISC requiere de 4 a 10 siglos de reloj. Entre las ventajas de CISC destacan las siguientes:

- Reduce la dificultad de crear compiladores
- Permite reducir el costo total del sistema
- Reduce Los costos de creación de software
- Mejora la compactación de código
- Facilita la depuración de errores

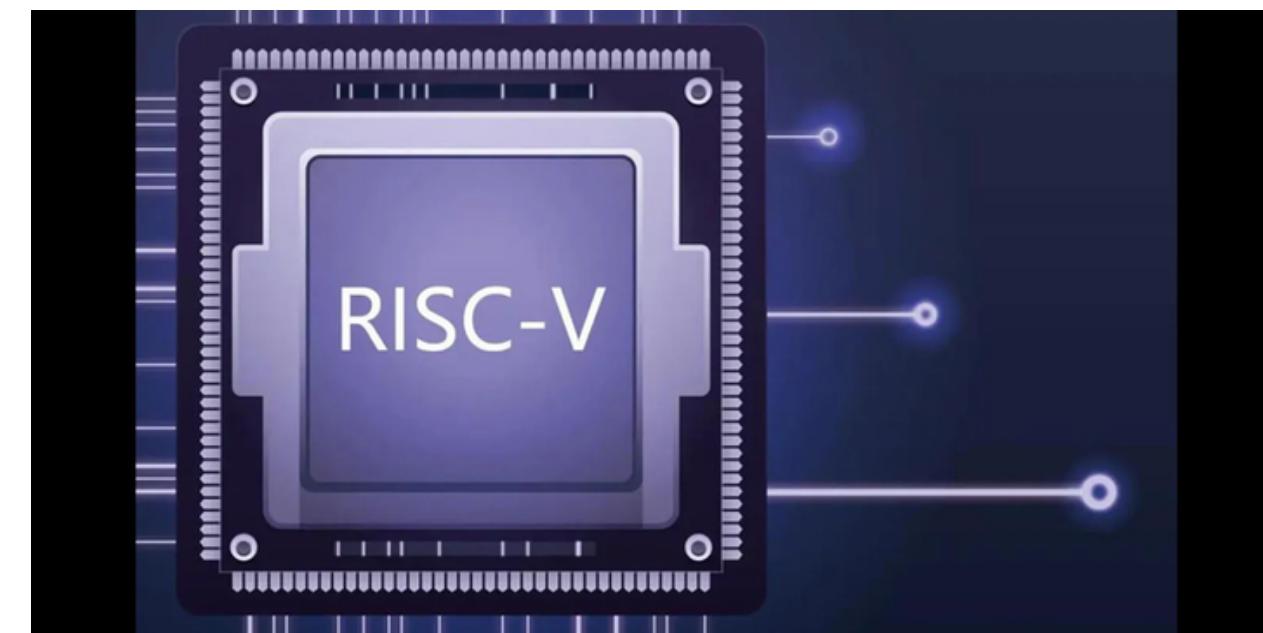


RISC: Reduced Instruction Set Computer es un tipo de un microprocesador con las siguientes características fundamentales:

- **Instrucciones de tamaño fijo y presentado en un reducido número de formatos**
- **Sólo las instrucciones de carga y almacenamiento acceden a la memoria de datos**

Los procesadores con tecnología RISC trabajan más rápido al utilizar menos ciclos de reloj para la ejecución de las instrucciones.

Además utiliza un sistema de direcciones no destructivas en RAM, significa que a diferencia de CISC, RISC conserva después de realizar sus operaciones en memoria a los dos operandos y su resultado como reduciendo la ejecución de nuevas operaciones. Y cada instrucción puede ser ejecutada en un solo ciclo del CPU.



Funcionamientos

ALU: Arithmetic Unit Logic, es un circuito digital que realiza como su nombre lo indica las operaciones aritméticas y lógicas entre los datos de un circuito; suma, resta, divide y multiplica, así como establece comparaciones lógicas a través de las condicionales lógicas "si", "no", "o".

Requiere de un mecanismo de control que le permita saber el tipo de operación a realizar.

Deberá contar con un circuito de control que le permita:

- Identificar la operación a realizar
- Administrar los recursos internos
- Generar las banderas

Unidad de control: Circuito secuencial que implementa el denominado "ciclo de instrucción", permitiendo acceder a la siguiente instrucción de un programa, leer sus operandos, efectuar la operación indicada en la ALU y guardar el resultado de la misma.

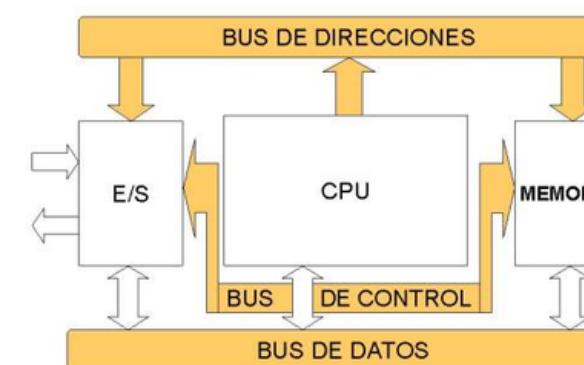
Funciones de la unidad de control:

- Decodificación de las instrucciones
- Sincronización de las tareas
- Administración de los buses internos de comunicación del microprocesador
- Indicación a la ALU sobre las operaciones a realizar

Registros: Los registros que encuentran dentro de cada procesador su función principal es almacenar los valores de cada uno de los datos, comandos, instrucciones o estados binarios que son de los que ordenan qué datos deben procesarse, así como la forma en la que se debe procesar o realizar.

En un procesador encontramos espacios con una capacidad que oscila entre 4 y 64 bits, porque cada registro debe tener un tamaño suficiente para contener una instrucción. En el caso de que un ordenador de 64 bits, cada registro debe tener un tamaño de 64 bits.

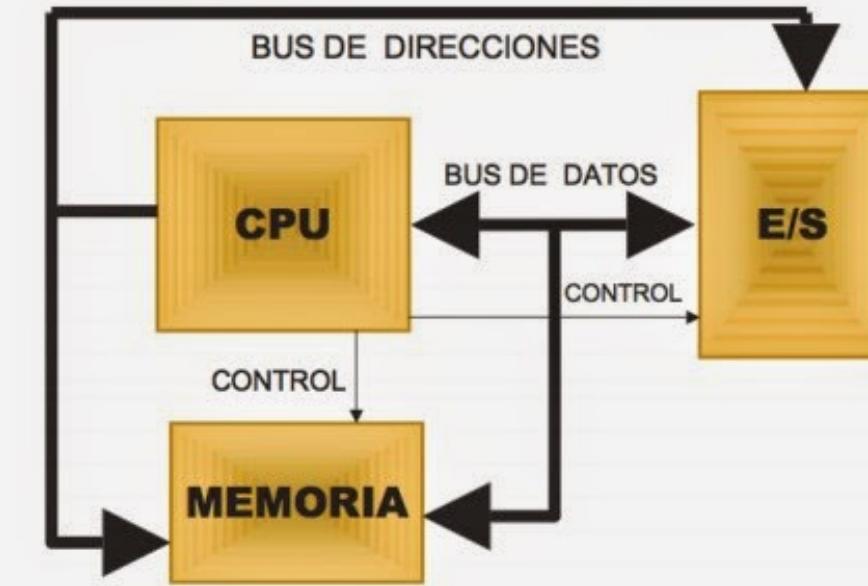
La información es recibida generalmente en código binario, procedente de las aplicaciones para, después, procesarlos de una forma determinada.



Buses: Consta de un camino que permite comunicar selectivamente un cierto número de componentes o dispositivos, de acuerdo a unas ciertas reglas o normas de conexión.

Incluye los conceptos de enlace y conmutador, ya que permite en cada momento seleccionar los dispositivos que se conectan a través suyo.

Estructura de los Buses





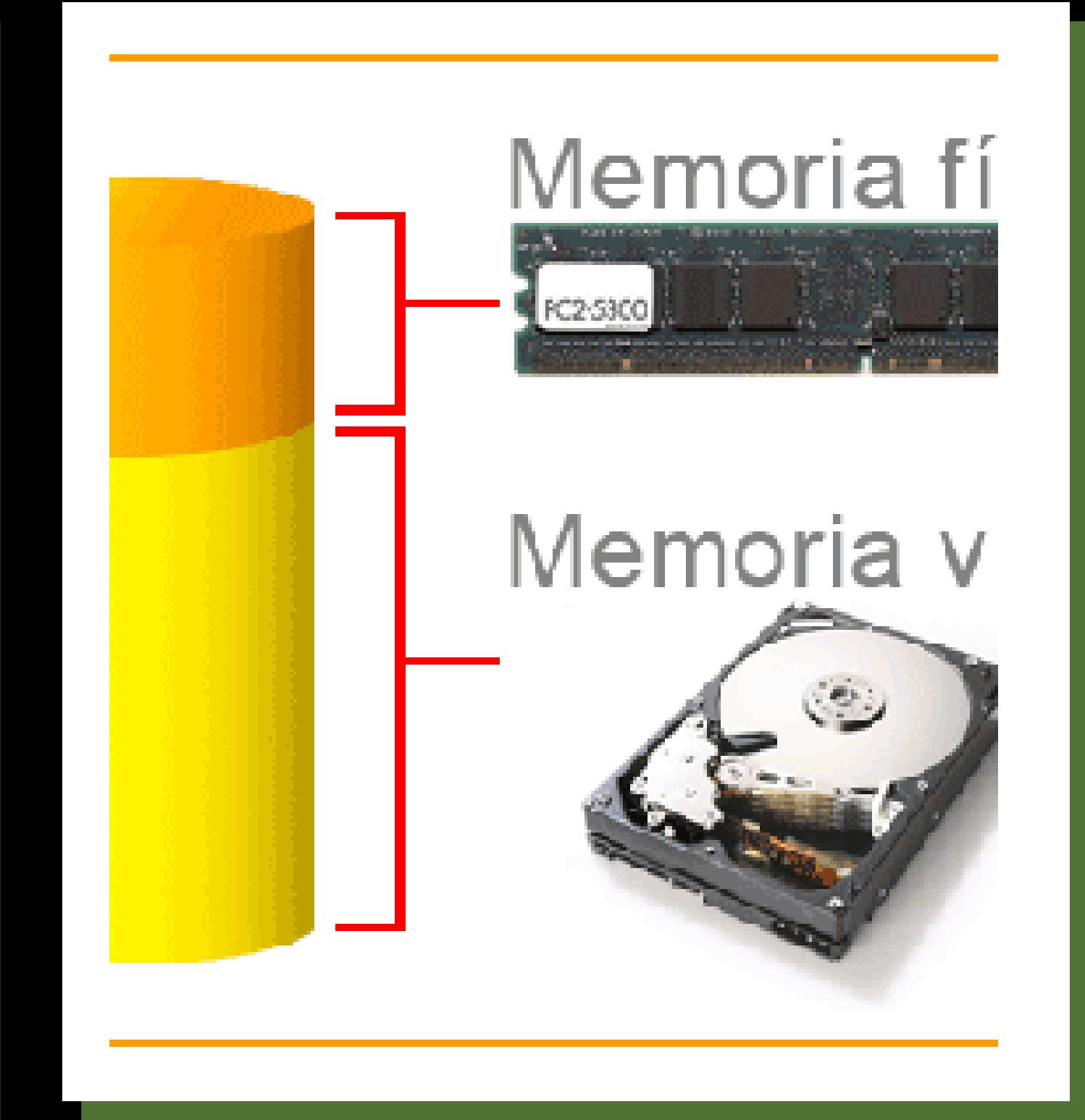
1.2.2

Memoria

- Conceptos básicos
- Memoria principal semiconductor
- Memoria Caché



Conceptos básicos de manejo de Memoria



Manejo de memoria

Memoria Física

Es la memoria real instalada en el hardware de la computadora. Incluye componentes como la memoria RAM y la memoria caché.

Administrador de memoria

Es el componente del sistema operativo que se encarga de gestionar el uso de la memoria. Se asegura de que los programas reciban la cantidad adecuada de memoria y asigna y libera espacio cuando sea necesario.

Memoria Virtual

Es una extensión de la memoria RAM que utiliza espacio en el disco duro para simular una memoria adicional. Permite que las aplicaciones utilicen más memoria de la que está físicamente presente.

Asignación de Memoria

Es el proceso de reservar una porción de la memoria para que un programa pueda utilizarla. Puede realizarse durante el inicio del programa (asignación estática) o durante la ejecución (asignación dinámica).



Manejo de memoria

Fragmentación

Puede haber fragmentación en la memoria cuando se desperdicia espacio debido a la asignación y liberación repetida de bloques de memoria. Hay dos tipos: fragmentación interna (ocurre en asignación estática) y fragmentación externa (ocurre en asignación dinámica).

Paging y Segmentacion

Son técnicas utilizadas para administrar la memoria en sistemas operativos. Paging divide la memoria en páginas del mismo tamaño, mientras que la segmentación divide la memoria en segmentos de diferentes tamaños.

Gestión de Memoria

Involucra llevar un registro de qué partes de la memoria están en uso y cuáles están libres. Esto incluye la asignación, liberación y reubicación de bloques de memoria.

Swap

Es el proceso de mover partes de la memoria de la RAM a la memoria virtual en el disco duro cuando la RAM está llena. Ayuda a liberar espacio en la RAM para nuevos datos.



Manejo de memoria

Thrashing

Ocurre cuando un sistema está ocupado en gran medida en intercambiar datos entre la memoria principal y la virtual debido a la falta de suficiente memoria física.

Protección de Memoria

Es el conjunto de mecanismos que evitan que un proceso acceda a la memoria de otro proceso sin permiso.

Políticas de Asignación

Son reglas que el sistema operativo sigue al asignar memoria a los procesos. Ejemplos incluyen el uso de asignación contigua, paginación o segmentación.

Overflow y Underflow:

Overflow ocurre cuando un valor excede el rango de representación en una variable, mientras que underflow ocurre cuando un valor es menor que el rango de representación.





Memoria principal semiconductora





Memoria Principal Semiconductora

La memoria principal semiconductora, también conocida como memoria principal de estado sólido, es una categoría de memoria de computadora que utiliza dispositivos semiconductores para almacenar datos de forma electrónica. A diferencia de las memorias magnéticas tradicionales, como los discos duros, que utilizan medios magnéticos para almacenar datos, la memoria principal semiconductora utiliza circuitos y transistores para retener y acceder a la información de manera mucho más rápida.



Tipos

1. Memoria RAM (Random Access Memory): Es la memoria volátil que se utiliza para almacenar datos y programas en uso activo. La RAM permite a la CPU acceder rápidamente a los datos necesarios para la ejecución de programas.
2. Memoria Cache: La memoria caché es una forma de memoria principal semiconductora utilizada para almacenar datos que la CPU necesita acceder con frecuencia. La memoria caché es más rápida que la memoria RAM convencional y ayuda a mejorar el rendimiento del sistema.



Tipos

- Memoria VRAM (Video RAM): Es una memoria especializada utilizada en tarjetas gráficas para almacenar los datos de imagen y video que se muestran en la pantalla. La VRAM permite un acceso rápido a los datos gráficos, lo que es esencial para la reproducción fluida de gráficos y videos.
- Memoria de Estado Sólido (SSD): Aunque no es típicamente considerada "principal" en el mismo sentido que la RAM, los SSD son dispositivos de almacenamiento basados en memoria flash (una forma de memoria semiconductor). Los SSD son utilizados como unidades de almacenamiento más rápidas y confiables en comparación con los discos duros convencionales.

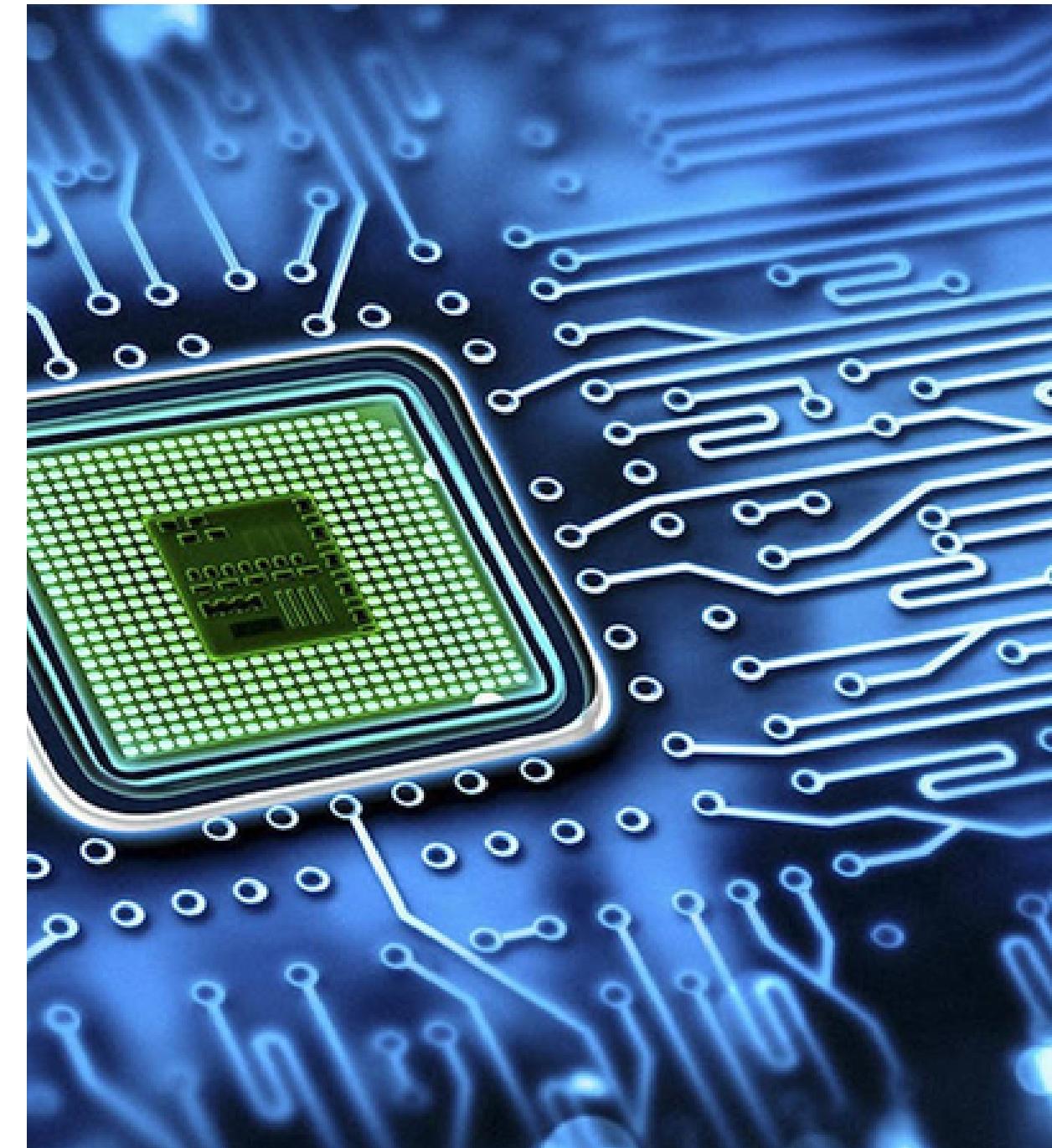


Ventajas

La memoria principal semiconductor tiene ventajas en términos de velocidad, consumo de energía y durabilidad en comparación con las tecnologías de memoria más tradicionales. Sin embargo, también es más costosa de producir en comparación con las soluciones magnéticas, lo que a menudo resulta en tamaños de almacenamiento más pequeños en comparación con los discos duros.



Memoria Caché





Memoria Caché

La memoria caché, comúnmente conocida como "caché", es una forma de memoria de alta velocidad que se encuentra entre la memoria RAM (principal) y la unidad central de procesamiento (CPU) en un sistema informático. Su objetivo principal es mejorar el rendimiento al almacenar temporalmente datos y programas que la CPU utiliza con frecuencia. La caché actúa como un búfer rápido que permite a la CPU acceder rápidamente a los datos que necesita, reduciendo los tiempos de espera y mejorando la eficiencia en la ejecución de instrucciones.



Memoria Caché

La caché opera utilizando el principio de localidad, que sugiere que un programa tiende a acceder a un conjunto limitado de datos y programas en un período corto. Cuando la CPU necesita datos, primero verifica si están en la caché. Si es así (lo que se llama "acceso a caché"), la CPU puede acceder a los datos mucho más rápido que si tuviera que acceder a la memoria principal. Si los datos no están en caché (lo que se llama "fallo de caché"), se busca en la memoria principal y se carga en la caché para futuros accesos.



Jerarquías

Existen diferentes niveles de caché en un sistema, que están organizados en jerarquías:

1. Caché L1: También conocida como caché de primer nivel, es la caché más cercana a la CPU y suele dividirse en cachés separadas para instrucciones y datos. Tiene la menor capacidad, pero la latencia más baja. Cada núcleo de la CPU suele tener su propia caché L1.
2. Caché L2: La caché de segundo nivel está ubicada en una capa intermedia entre la caché L1 y la memoria principal. Puede ser compartida entre varios núcleos de la CPU y generalmente tiene más capacidad que la caché L1, aunque con una latencia un poco mayor.



Jerarquías

- Caché L3: La caché de tercer nivel es la más grande de las cachés de CPU, y también puede ser compartida por múltiples núcleos. Su tamaño suele ser mayor para permitir un almacenamiento más amplio de datos.
- Caché de Almacenamiento: Algunos sistemas modernos también incorporan una caché de almacenamiento en las unidades de almacenamiento (discos SSD) para acelerar la lectura y escritura de datos.

En resumen, la memoria caché es una tecnología esencial para mejorar el rendimiento de la CPU y reducir la latencia en el acceso a datos frecuentes, lo que resulta en una mejor experiencia de uso y un mayor rendimiento de los sistemas informáticos.



1.2.3

dispositivos

entrada

salida

- 1.2.3.1 Modulos E/S
- 1.2.3.2 E/S programada
- 1.2.3.3 E/S mediante interrupciones
- 1.2.3.4 Acceso directo a la memoria
- 1.2.3.5 Canales y procesadores E/S



Dispositivos de Entrada/Salida (E/S):

- ¿Qué son? Los dispositivos de entrada/salida son componentes que permiten la comunicación entre una computadora y el mundo exterior. Estos dispositivos se utilizan para ingresar datos a la computadora (entrada) o enviar datos desde la computadora (salida).



Entrada/Salida (E/S)

Entrada (Input): Los dispositivos de entrada son aquellos que permiten a la computadora recibir datos o información desde el mundo exterior y llevarlos al sistema para su procesamiento. Estos dispositivos "introducen" datos en la computadora. Por ejemplo, cuando escribes en un teclado o haces clic con un mouse.

Salida (Output): Los dispositivos de salida son aquellos que permiten a la computadora enviar datos o información desde el sistema al mundo exterior, para que los usuarios puedan percibir o utilizar la información generada por la computadora. Estos dispositivos "sacan" datos del sistema. Por ejemplo, cuando la computadora muestra una imagen en la pantalla del monitor o imprime un documento en una impresora, está enviando datos hacia fuera para que los usuarios los vean o utilicen.



Ejemplos

Dispositivos de Entrada:

- Teclado: Un dispositivo de entrada estándar que permite al usuario ingresar datos y comandos mediante la pulsación de teclas.
- Mouse: Permite al usuario controlar el movimiento del cursor en la pantalla y seleccionar elementos mediante clics y movimientos.

Dispositivos de salida:

- Altavoces: Emiten sonido y audio para que el usuario pueda escuchar música, efectos de sonido y diálogos de video.
- Impresora: Produce copias impresas de documentos, imágenes o gráficos en papel u otros medios.

input output





1.2.3.1. Módulos E/S:

Los módulos E/S son componentes esenciales en la arquitectura de una computadora que facilitan la comunicación entre la Unidad Central de Procesamiento (CPU) y los dispositivos de entrada/salida (E/S). Su función principal es gestionar la transferencia de datos entre la CPU y los dispositivos externos, garantizando que la comunicación sea eficiente y confiable.



Ejemplos de Módulos E/S:

1. Controladores de Disco Duro: Los controladores de disco duro son módulos E/S diseñados específicamente para gestionar la lectura y escritura de datos en unidades de disco duro. Controlan cómo se leen y escriben los datos en el disco y proporcionan una interfaz para que la CPU interactúe con el disco.
2. Tarjeta de Red: En una tarjeta de red, el módulo E/S maneja la comunicación entre la computadora y una red, ya sea mediante conexiones por cable (Ethernet) o inalámbricas (Wi-Fi). El controlador de la tarjeta de red permite que la CPU envíe y reciba datos a través de la red.



Ejemplos de Módulos E/S:

1. Controlador de Pantalla Gráfica (GPU): En el caso de las tarjetas gráficas, los módulos E/S gestionan la representación de gráficos en el monitor. Controlan cómo se muestran imágenes, videos y animaciones en la pantalla y permiten a la CPU interactuar con la GPU.
2. Controlador USB: Los controladores USB son módulos E/S que permiten a la CPU comunicarse con dispositivos USB, como teclados, ratones, unidades flash y otros dispositivos periféricos. Controlan la transferencia de datos y la administración de energía de los dispositivos conectados.



Características de los Módulos E/S:

1. Interfaz de Comunicación: Cada módulo E/S tiene una interfaz específica que define cómo la CPU y el dispositivo de E/S se comunican entre sí. Esta interfaz puede ser física (puertos, cables, conectores) o lógica (protocolos de comunicación).
2. Control y Temporización: Los módulos E/S controlan el flujo de datos, asegurando que se transmitan y reciban de manera sincronizada y en el momento adecuado. Esto implica el uso de temporizadores y señales de control.
3. Gestión de Errores: Los módulos E/S suelen incluir mecanismos para detectar y corregir errores de transmisión de datos. Esto es crucial para garantizar la integridad de los datos transmitidos.



E/S programada

La E/S (Entrada/Salida) programada se refiere a un enfoque en la informática donde el programa controla directamente las operaciones de entrada y salida de datos hacia y desde los dispositivos periféricos. Esto significa que el programa es responsable de leer y escribir datos en los dispositivos de manera manual en lugar de dejar que el sistema operativo gestione automáticamente estas operaciones.



E/S programada

1. Entrada/Salida Programada (Programmed I/O): En este enfoque, el programa envía comandos directamente a los dispositivos periféricos (como discos duros, impresoras o teclados) para realizar operaciones de lectura o escritura de datos. El programa espera hasta que la operación se complete antes de continuar su ejecución.
2. Control Total: La E/S programada proporciona al programa un mayor control sobre los dispositivos periféricos, lo que puede ser útil en situaciones donde se requiere un control preciso o personalizado de la comunicación con estos dispositivos.



Cuando se usa la E/S programada

1. Configuración Inicial del Hardware: Durante la fase de inicio de un sistema, se pueden usar operaciones de E/S programadas para configurar dispositivos y periféricos antes de que el sistema operativo o el software de aplicación tomen el control completo.
2. Aplicaciones de Baja Demanda de Rendimiento: En aplicaciones que no requieren una alta velocidad de E/S y donde la CPU puede permitirse esperar, como sistemas de control industrial de baja velocidad, la E/S programada puede ser suficiente.



1.2.3.3. E/S Mediante Interrupciones:

La E/S mediante interrupciones es un método de gestión de dispositivos de entrada/salida en una computadora que permite que los dispositivos de E/S interrumpan la CPU cuando están listos para transmitir datos o cuando se produce una condición especial. En lugar de que la CPU espere activamente y controle directamente cada operación de E/S, los dispositivos pueden solicitar la atención de la CPU mediante una interrupción, lo que libera a la CPU para realizar otras tareas mientras se maneja la operación de E/S en segundo plano.



Cómo Funciona la E/S Mediante Interrupciones:

- Solicitud de Interrupción: Cuando un dispositivo de E/S está listo para transmitir datos o necesita atención, genera una solicitud de interrupción al sistema.
- Manejo de la Interrupción: Cuando se genera una solicitud de interrupción, la CPU suspende temporalmente la ejecución del programa actual y comienza a manejar la interrupción.
- Identificación y Priorización: El sistema operativo identifica la fuente de la interrupción (es decir, qué dispositivo o evento la generó) y determina la acción apropiada.
- Atención de la E/S: La CPU atiende la operación de E/S asociada con la interrupción.
- Retorno a la Tarea Original: Una vez que se completa el manejo de la interrupción y la operación de E/S, la CPU regresa a la tarea original que estaba ejecutando antes de que se generara la interrupción.

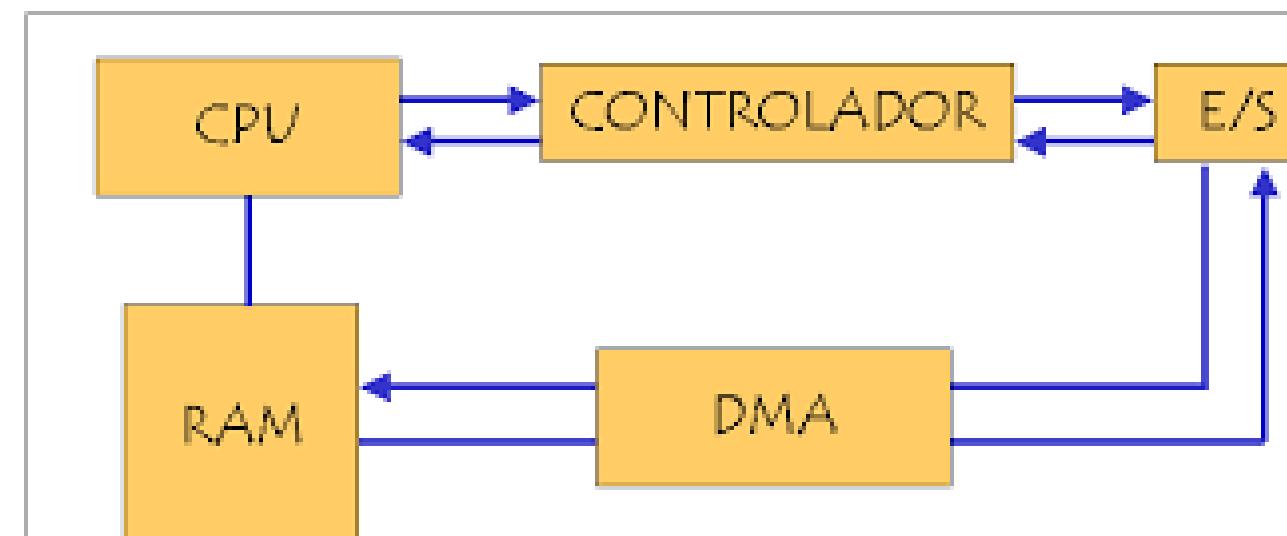


Cuándo se Utiliza la E/S Mediante Interrupciones:

- **Sistemas Multitarea:** En sistemas donde varias tareas se ejecutan simultáneamente, la E/S mediante interrupciones permite que la CPU atienda las operaciones de E/S de manera eficiente sin bloquear otras tareas.
- **Operaciones Concurrentes:** Cuando es necesario manejar múltiples operaciones de E/S concurrentes, como transferencias de datos de alta velocidad o comunicación en red, las interrupciones ayudan a coordinar y administrar estas operaciones de manera efectiva.
- **Respuesta Rápida:** En situaciones en las que se necesita una respuesta rápida a eventos externos, como una pulsación de tecla o una señal de red, las interrupciones permiten una respuesta inmediata sin demoras.

1.2.3.4. Acceso Directo a la Memoria:

El Acceso Directo a la Memoria (DMA) es una técnica utilizada en arquitectura de computadoras que permite a dispositivos de entrada/salida (E/S) transferir datos directamente entre ellos y la memoria principal de la computadora sin la intervención constante de la Unidad Central de Procesamiento (CPU). Esta técnica es especialmente útil para acelerar las operaciones de E/S y liberar a la CPU para realizar otras tareas.





Cómo Funciona el Acceso Directo a la Memoria (DMA):

- Controlador DMA: La computadora tiene un controlador DMA dedicado que supervisa y coordina las operaciones de DMA.
- Configuración de Parámetros: Antes de iniciar una transferencia de datos mediante DMA, el controlador DMA debe configurar ciertos parámetros, como la dirección de inicio y final de la transferencia de datos.
- Solicitud de Acceso: Un dispositivo de E/S que desea realizar una transferencia de datos mediante DMA envía una solicitud al controlador DMA. La solicitud puede incluir detalles como la cantidad de datos que se deben transferir y las direcciones de memoria involucradas.
- Aprobación y Configuración: El controlador DMA revisa la solicitud y, si es válida, aprueba la operación.



Cómo Funciona el Acceso Directo a la Memoria (DMA):

- Transferencia de Datos: Una vez configurado, el controlador DMA inicia la transferencia de datos directamente entre el dispositivo de E/S y la memoria principal. La CPU no está directamente involucrada en esta transferencia, lo que libera a la CPU para realizar otras tareas.
- Interrupción o Señal de Finalización: Una vez que se completa la transferencia de datos, el dispositivo de E/S notifica al controlador DMA, que a su vez puede generar una interrupción o una señal de finalización para que el sistema sepa que la operación de DMA ha concluido.



Ventajas del DMA:

- Mayor Eficiencia: Permite transferencias de datos más rápidas y eficientes al evitar la sobrecarga de la CPU, que de otro modo tendría que estar involucrada en cada operación de E/S.
- Libera la CPU: La CPU se libera para realizar tareas más importantes y no se ve obstaculizada por operaciones de E/S intensivas.
- Acelera las Operaciones de E/S: Es especialmente útil para dispositivos de alta velocidad, como unidades de disco duro y tarjetas gráficas, donde la velocidad de transferencia es crítica.



1.2.3.5. Canales y Procesadores E/S:

-

Los canales y procesadores de E/S son componentes fundamentales en la arquitectura de computadoras que se utilizan para gestionar múltiples dispositivos de entrada/salida (E/S) de manera eficiente y coordinada. Estos componentes permiten que la computadora maneje un gran número de operaciones de E/S de manera concurrente sin sobrecargar la Unidad Central de Procesamiento (CPU).

Los canales y procesadores de E/S son unidades de hardware especializadas diseñadas para facilitar y acelerar la comunicación entre la CPU y múltiples dispositivos de E/S. Tienen la capacidad de ejecutar operaciones de E/S de manera independiente de la CPU principal y pueden administrar varias operaciones de E/S al mismo tiempo.



Ejemplos de canales y procesadores E/S

Ejemplos de Canales de E/S:

- Canal de Control de Disco (DCC - Disk Control Channel): En sistemas de almacenamiento de datos, el DCC es un canal de E/S especializado que administra las operaciones de lectura y escritura en unidades de disco duro. Permite un acceso eficiente y rápido a los datos almacenados en discos.
- Canal de Red (Network Channel): En sistemas de comunicación en red, un canal de red se encarga de gestionar la transferencia de datos entre la CPU y las tarjetas de red. Puede manejar múltiples conexiones de red de manera concurrente.



Ejemplos de canales y procesadores E/S

Ejemplos de Procesadores de E/S:

- Procesador de Red (Network Processor): En sistemas de comunicación en red, un procesador de red es una unidad especializada que se encarga de tareas como la administración de paquetes, la detección de errores y el enrutamiento de datos. Mejora el rendimiento de las comunicaciones en red.
- Procesador de Gráficos (Graphics Processor): En tarjetas gráficas, el procesador de gráficos, o GPU, es un tipo de procesador de E/S especializado en renderizar gráficos y acelerar operaciones gráficas, como juegos y aplicaciones de diseño.



Ventajas de los Canales y Procesadores de E/S:

- Mayor Eficiencia: Permiten que la CPU principal se libere de la carga de gestionar operaciones de E/S individuales, lo que mejora la eficiencia general del sistema.
- Gestión Concurrente: Pueden manejar múltiples operaciones de E/S simultáneamente, lo que es esencial en sistemas de alto rendimiento con numerosos dispositivos de E/S.
- Optimización de Recursos: Ayudan a optimizar el uso de recursos de E/S al coordinar de manera efectiva las transferencias de datos.
- Mejor Rendimiento: Contribuyen significativamente al rendimiento general de la computadora al acelerar las operaciones de E/S y permitir una mayor concurrencia.



1.2.4 Buses

- 1.2.4.1 Buses Local
- 1.2.4.2 Buses de datos
- 1.2.4.3 Buses de Direcciones
- 1.2.4.4 Buses de Control
- 1.2.4.5 Normalizacion de Buses



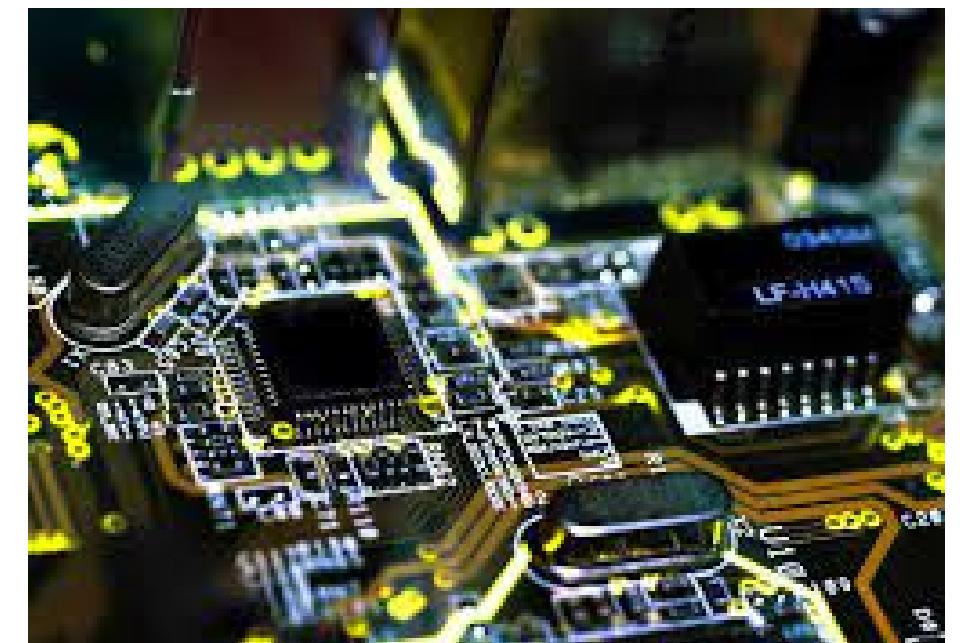
Bus Local



Bus Local

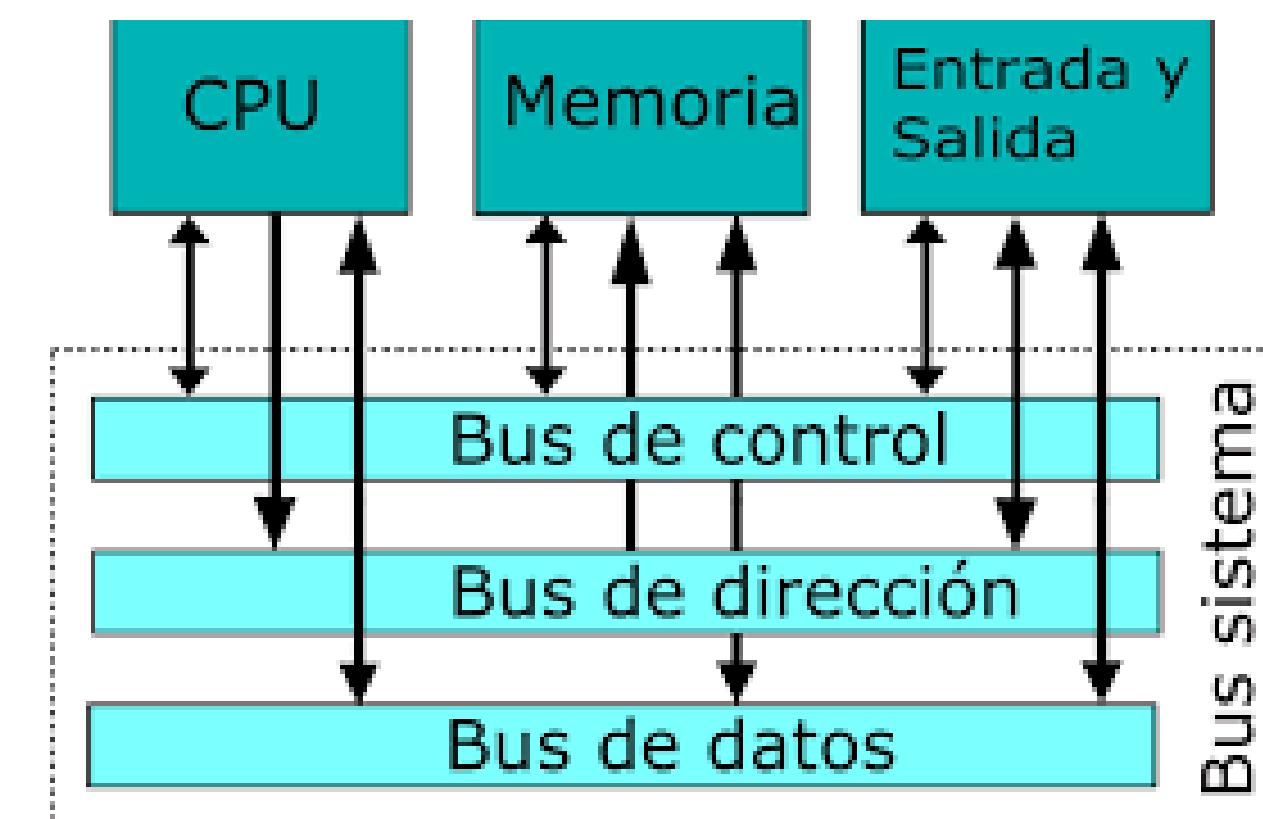
Un bus local es un bus informático que se conecta directamente, o casi directamente, desde la unidad central de procesamiento (CPU) a una o más ranuras en el bus de expansión.

La importancia de la conexión directa a la CPU es evitar el cuello de botella creado por el bus de expansión, proporcionando así un rendimiento rápido. Hay varios buses locales integrados en varios tipos de computadoras para aumentar la velocidad de transferencia de datos



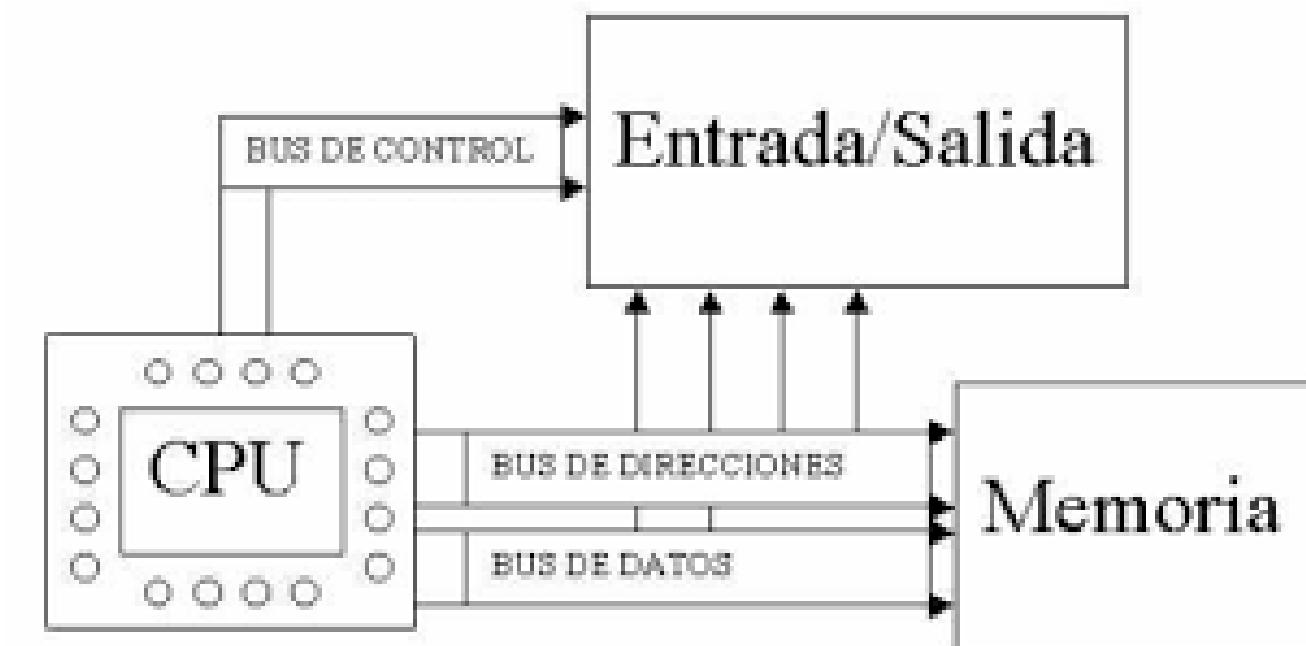
Bus de Direcciones

El bus de direcciones es el canal del micróprocesador, que es independiente del bus de datos y que es donde se establece la dirección de memoria del dato que se está transmitiendo. Este bus representa el conjunto de líneas eléctricas que se necesitan para establecer una dirección.



Bus de Control

es un canal de comunicación que se utiliza para conectar varios componentes de un sistema. Se trata de un medio por el cual los dispositivos intercambian información y se coordinan entre sí para realizar tareas específicas. El bus de control permite a los dispositivos comunicarse, compartir información y recibir instrucciones.





Funciones del bus de control

Las funciones del bus de control son importantes para el correcto funcionamiento de cualquier sistema informático. Estas funciones permiten que los dispositivos se comuniquen entre sí y que los usuarios accedan a los recursos necesarios para realizar sus tareas. El bus de control es una parte esencial del sistema, ya que proporciona servicios básicos como la dirección de memoria, el control de interrupciones y la transferencia de datos.

Funciones

- Dirección de memoria.
- Control de interrupciones
- Transferencia de datos
- Autoconfiguración.



Normalización de Buses

Antes cada fabricante definía sus buses lo cual dificultaba mucho la comunicación entre distintos componentes. Para facilitar la interacción entre componentes de distintos fabricantes los buses se han <<normalizado>>. Siguen un estándar acordado previamente.

Estándar:

Protocolos de transmisión de datos.

QR Velocidades y temporización de las transferencias.

QR Anchuras de los sub-buses.

Sistema físico de conexión.



Especificaciones de los buses normalizados

Nivel mecánico:

deben definirse aspectos tales como el tipo de soporte, el número de hilos del bus, el tipo de conector, etc.

Nivel eléctrico:

En este nivel, debe quedar definida la forma en la que los distintos dispositivos deben conectarse eléctricamente.

Nivel lógico:

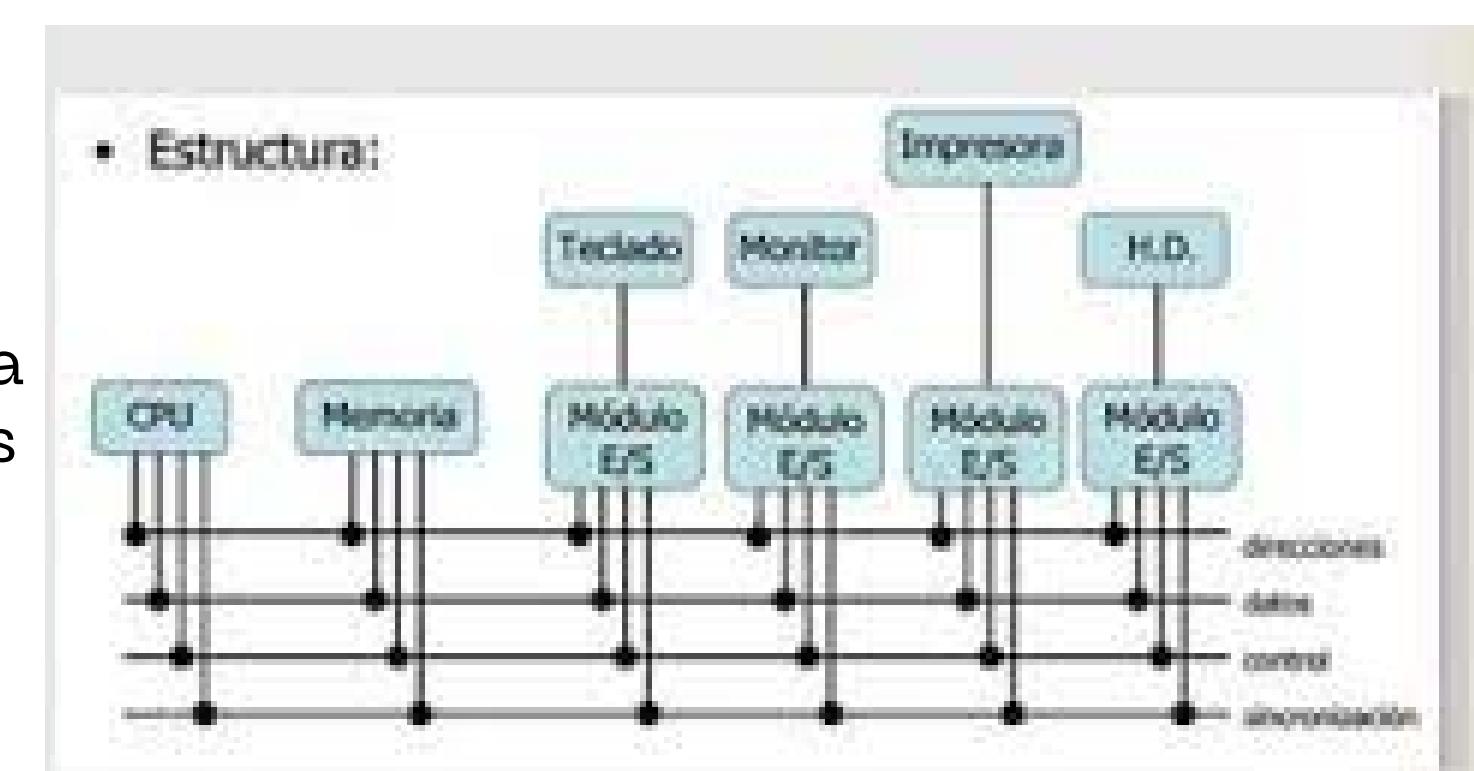
Establece las equivalencias entre los valores eléctricos de las señales y sus valores lógicos.

Especificaciones de los buses normalizados

QR Nivel de temporización básica: En este nivel se establecen los cronogramas para la realización de la operación más elemental del bus, esto es, de un ciclo.

QR Nivel de transferencia elemental: En este nivel se establece el procedimiento empleado para realizar una transferencia de un dato por el bus. Establece todas las condiciones necesarias para transferir un dato.

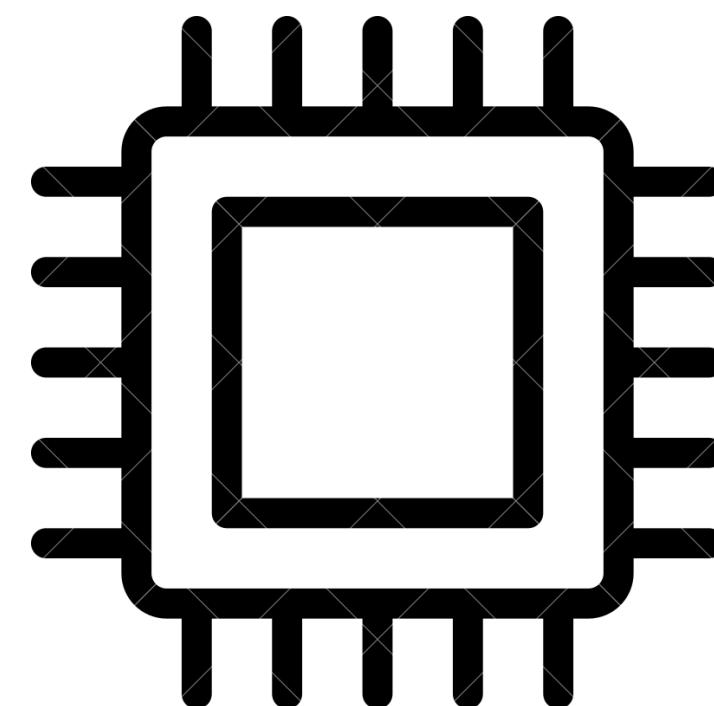
QR Nivel de transferencia de bloque:
Define el protocolo de comunicación empleado para realizar esta transferencia de bloque.





Bus de Datos

El bus es un sistema digital que transfiere datos entre los componentes de un ordenador o entre ordenadores. Está formado por cables o pistas en un circuito impreso, dispositivos como resistencias y condensadores además de circuitos integrados.





Tipo de Buses

- SERIE y PARALELO: los primeros transmiten bit a bit y los segundos varios bits a la vez.
- MULTIPLEXADOS y NO MULTIPLEXADOS o DEDICADOS: los multiplexados realizan diferentes funciones en función de las necesidades del momento.
- CENTRALIZADOS y DISTRIBUIDOS: necesidad de determinar qué elemento transmite y cuál recibe. Generalmente existe administración centralizada por la CPU o procesador.
- SÍNCRONOS y ASÍNCRONOS : cómo ocurren los diferentes eventos implicados en la transmisión de información. Utilización de una señal de reloj o unas líneas de protocolo .



1.2.5

Direcccionamiento

- 1.2.5.1 Modo Real
- 1.2.5.2 Modo protegido
- 1.2.5.3 Modo virtual



Modo Real

Este es un modo de operación específico en las CPUs x86 que se utiliza en el proceso de inicio de la computadora y la ejecución de sistemas operativos antiguos, como MS-DOS. En este modo, la CPU tiene acceso directo a la memoria y opera en un entorno de 16 bits.

La arquitectura 286 introdujo el modo protegido, permitiendo, entre otras cosas, la protección de la memoria a nivel de hardware. Sin embargo, usar estas nuevas características requirió instrucciones de software adicionales no necesarias previamente.





Modo Real

Los sistemas operativos DOS (MS-DOS, DR-DOS, etc.) trabajan en modo real. Las primeras versiones de Microsoft Windows, que eran esencialmente un shell de interface gráfica de usuario corriendo sobre el DOS, no eran realmente un sistema operativo por sí mismas, corrían en modo real, hasta Windows 3.0, que podía correr tanto en modo real como en modo protegido.





Modo Protegido

El modo protegido es un modo operacional de los CPUs compatibles x86 de la serie 80286 y posteriores. El modo protegido tiene un número de nuevas características diseñadas para mejorar las multitareas y la estabilidad del sistema, como protección de memoria, y soporte de hardware para memoria virtual así como de commutación de tareas.

El modo protegido también tiene soporte de hardware para interrumpir un programa en ejecución y cambiar el contexto de ejecución a otro, permitiendo pre-emptive multitasking.





Modo Protegido

La mayoría de los sistemas operativos x86 modernos corren en modo protegido, incluyendo Linux, FreeBSD, OpenBSD, NetBSD, y Microsoft Windows 3.0 y posteriores. (Windows 3.0 también corría en el modo real para la compatibilidad con las aplicaciones de Windows 2.x).





Modo Virtual

En el microprocesador 80386 y posteriores, el modo 8086 virtual, también llamado modo real virtual o VM86, permite la ejecución de aplicaciones de modo real que violan las reglas bajo control de un sistema operativo de modo protegido.

Era usado para ejecutar programas DOS en Microsoft Windows/386, Windows 3.x, Windows 95, Windows 98, Windows Me, y OS/2 2.x y más adelante, a través de las máquinas DOS virtuales, también en SCO UNIX a través de Merge, y en Linux por medio de dosemu.





1.2.6

Temporización

- 1.2.6.1 Reloj del sistema
- 1.2.6.2 Reset del sistema
- 1.2.6.3 Estados de espera



Reloj del sistema

El reloj de una computadora se utiliza para dos funciones principales:

1. Para sincronizar las diversas operaciones que realizan los diferentes subcomponentes del sistema informático.
2. Para saber la hora. El reloj físicamente es un circuito integrado que emite una cantidad de pulsos por segundo, de manera constante. Al número de pulsos que emite el reloj cada segundo se llama Frecuencia del Reloj.



Reloj del sistema

El reloj marca la velocidad de proceso de la computadora generando una señal periódica que es utilizada por todos los componentes del sistema para sincronizar y coordinar las actividades operativas, evitando el que un componente maneje unos datos incorrectamente o que la velocidad de transmisión de datos entre dos componentes sea distinta.

Cuanto mayor sea la frecuencia del reloj, mayor será la velocidad de proceso de la computadora y podrá realizar mayor cantidad de instrucciones elementales en un segundo.



Reset del sistema

Se conoce como reset a la puesta en condiciones iniciales de un sistema. Este puede ser mecánico, electrónico o de otro tipo. Normalmente se realiza al conectar el mismo, aunque, habitualmente, existe un mecanismo, normalmente un pulsador, que sirve para realizar la puesta en condiciones iniciales manualmente.

En el ámbito de códigos binarios, trata de poner a 0, así como set, poner a 1. resetearlo (reset).



Estados de espera

Un estado de espera es una situación en la que el procesador de la computadora experimenta un retraso, principalmente cuando accede a la memoria externa o a un dispositivo que responde lentamente.

Por lo tanto, los estados de espera se consideran un desperdicio en el rendimiento del procesador. Sin embargo, los diseños modernos intentan eliminar o minimizar los estados de espera.



Estados de espera

Cuando el procesador requiere acceso a la memoria principal, comienza colocando la dirección de la información solicitada en el bus de direcciones. Después de esto, el procesador necesita esperar la respuesta, que puede tardar varios ciclos más tarde.

Cada uno de estos ciclos se gasta en un estado de espera.