

Organizacion de las Computadoras

Guia teoria



Clase 2: BCC, Ca1, Ca2, Ex2

Representación Binaria con Signo

La representación binaria con signo (BCS) es un método de codificación numérica que permite expresar tanto números positivos como negativos utilizando bits. Esta representación se basa en asignar un bit para indicar el signo y los restantes bits para la magnitud del número.

Formato de Representación

- En una representación BCS con n bits, el bit más significativo (extremo izquierdo) se utiliza para indicar el signo, mientras que los bits del 0 al $n-2$ representan la magnitud del número.

Indicación del Signo

- Un bit de signo con valor 0 indica que el número es positivo.
- Un bit de signo con valor 1 indica que el número es negativo.

Magnitud en Binario

- Los bits que representan la magnitud del número son interpretados de manera similar a la representación binaria convencional, es decir, cada bit contribuye a un valor en potencias de 2.

Rango de Valores

- El rango de valores representados en BCS es desde $-(2^{(n-1)} - 1)$ hasta $+(2^{(n-1)} - 1)$, con dos ceros positivos y negativos.

Ejemplos

- **Ejemplo 1:** En una representación de 8 bits, 00100000 representa el número +32, mientras que 10100000 representa el número -32.
- **Ejemplo 2:** Con $n = 8$ bits, el número binario 11111111 representa -127, y el número binario 01111111 representa +127.

Características de BCS

- El intervalo de valores es simétrico, es decir, incluye valores positivos y negativos.
- El primer bit (MSB) es utilizado para indicar el signo.
- Los números positivos comienzan con 0, y los negativos con 1.
- Existen dos representaciones de cero (positivo y negativo).
- En una representación BCS de n bits, pueden representarse 2^n números distintos.

Técnica de Complementos en la Representación Binaria con Signo

La técnica de complementos es una estrategia utilizada en la representación binaria con signo (BCS) para representar números positivos y negativos. Esta técnica se basa en el concepto de obtener el complemento de un número en relación a una base dada, lo que permite realizar operaciones de suma y resta de manera eficiente.

Complemento a la Base Disminuida (Ca1)

Para números en sistema binario, el complemento a 1 (Ca1) se obtiene invirtiendo todos los bits del número.

Representación en Ca1

- En la representación Ca1, los n bits representan el número, y el bit de signo se encuentra en el bit más significativo ($n-1$).
- Los números positivos se representan como en la notación binaria convencional.
- Para números negativos, se utiliza el Ca1 del valor absoluto del número deseado.

Características de Ca1

- Los números positivos comienzan con 0, y los negativos comienzan con 1.
- El rango de valores representados es simétrico, desde $-(2^{(n-1)} - 1)$ hasta $+(2^{(n-1)} - 1)$, con dos representaciones de cero.

Complemento a la Base (Ca2)

- En el sistema binario, el complemento a 2 (Ca2) de un número se obtiene invirtiendo todos los bits del complemento a 1 (Ca1) y luego sumándole 1.

Representación en Ca2

- Al igual que en Ca1, en la representación Ca2, los n bits representan el número, y el bit de signo se encuentra en el bit más significativo ($n-1$).
- Los números positivos se representan como en la notación binaria convencional.
- Para números negativos, se utiliza el Ca2 del valor absoluto del número deseado.

Características de Ca2

- Los números positivos comienzan con 0, y los negativos comienzan con 1.
- El rango de valores representados es asimétrico, desde $-(2^{(n-1)})$ hasta $+(2^{(n-1)} - 1)$, con un único cero.

Técnica del Exceso en la Representación Numérica

La técnica del exceso es un método de representación numérica en el que un número se define como la suma de sí mismo y un valor constante denominado exceso. Esta técnica es utilizada para representar valores en sistemas digitales y puede ser aplicada en diferentes bases numéricas.

Representación con Exceso

- La representación de un número A se obtiene sumándole un valor constante E , conocido como exceso:
 - Exceso E de $A = A + E$
- Para recuperar el valor original A a partir de su representación con exceso, se resta el valor del exceso:
 - $A = (\text{Exceso } E \text{ de } A) - E$
- El signo del número A resulta de una resta, lo que implica que el exceso puede ser positivo o negativo dependiendo de la convención utilizada.

Rango de Valores

- El rango de valores representables utilizando la técnica del exceso depende de la cantidad de bits utilizados y del valor del exceso.

Exceso 2^{n-1}

- Si se utiliza un exceso de $2^{(n-1)}$, donde n es la cantidad de bits, el rango de valores representables será:
 - $-2^{(n-1)} \leq x \leq 2^{(n-1)} - 1$
- Por ejemplo, si $n = 6$, el rango sería desde -32 hasta 31.

Nuevas Banderas Aritméticas en la Representación Numérica

Flag V (Overflow) - Desbordamiento

- La bandera V, también conocida como "overflow", se establece en 1 cuando ocurre una condición de desbordamiento durante una operación aritmética.
- El desbordamiento ocurre cuando el resultado de una operación no puede ser expresado dentro del rango de valores permitidos por el número de bits utilizados.
- Es comúnmente utilizado en sistemas que emplean la representación en complemento a dos (Ca2) para detectar situaciones en las que el resultado no puede ser representado adecuadamente debido a limitaciones de bits.

Flag N (Negativo) - Signo del Resultado

- La bandera N, o "negativo", refleja el bit más significativo (el bit de signo) del resultado de una operación aritmética.
- Si el resultado es negativo, la bandera N se establece en 1; si es positivo, la bandera N se pone en 0.
- Esta bandera proporciona una manera rápida de determinar el signo del resultado de una operación, lo que es especialmente útil en situaciones donde se necesitan decisiones rápidas basadas en el signo de un número.

Operaciones Aritméticas en Complemento a Dos (Ca2)

Suma en Ca2

- En la suma de dos números en Ca2, el resultado puede generar un desbordamiento si se suman dos números positivos y el resultado es negativo, o si se suman dos números negativos y el resultado es positivo.

- Para sumar dos números en $Ca2$, simplemente se suman los n bits de los números de manera directa.
- Si los números involucrados tienen signos opuestos, no puede ocurrir un desbordamiento.

Resta en $Ca2$

- En la resta de dos números en $Ca2$, se pueden seguir dos enfoques: restar directamente los n bits o transformar la resta en una suma sumando el complemento a dos del sustraendo.
- Si los números a restar tienen el mismo signo, no habrá desbordamiento.
- Sin embargo, si se resta un número positivo de un número negativo y el resultado es negativo, o si se resta un número negativo de un número positivo y el resultado es positivo, se presentará un desbordamiento en la resta.

Clase 3: Representación de números en punto flotante

Es una técnica que permite manejar tanto números enteros como números con componentes fraccionarias de manera eficiente. Esta metodología ofrece un rango amplio y versátil para representar números de diversos tamaños. A continuación, exploraremos los conceptos clave relacionados con la representación de números en punto flotante.

Números en Punto Fijo

- En la representación de números en punto fijo, todos los números comparten la misma cantidad de dígitos, y la coma fraccionaria se presume que se encuentra en una posición predeterminada.
- La principal diferencia entre la representación en papel y en una computadora es que no se almacena la coma, sino que se supone que se encuentra en un lugar específico.

Rango y Resolución

- El rango se refiere a la diferencia entre el número mayor y el menor que se puede representar.
- La resolución es la diferencia entre dos números consecutivos en la representación.

Error en Punto Fijo

- El error máximo en una representación en punto fijo puede considerarse como la mitad de la resolución, ya que corresponde a la diferencia entre dos números consecutivos.
- Este error máximo es importante para comprender la precisión de la representación en punto fijo.

Números en Punto Flotante

- Los números en punto flotante permiten representar tanto números enteros como números con componentes fraccionarias.
- En comparación con la representación en punto fijo, los números en punto flotante son más flexibles y pueden representar un rango más amplio de valores.
- Esta representación utiliza un formato dinámico para mover la coma decimal y utiliza un exponente para mantener la "pista" de la posición de la coma.

Formato de Números en Punto Flotante

- Los números en punto flotante se representan en el formato $\pm M \times B^E$, donde M es la mantisa y E es el exponente.
- La base B es implícita y no necesita almacenarse, ya que es constante para todos los números en el sistema.
- Almacenar la mantisa y el exponente requiere menos bits en comparación con almacenar el número completo en su base correspondiente.

Mantisa y Exponente en Ca2

- En el formato de punto flotante, la mantisa y el exponente desempeñan un papel fundamental.
- El rango y la resolución de la representación en punto flotante dependen de la configuración de la mantisa y el exponente.

Rango y Resolución en Ca2

- El rango en Ca2 se determina por el valor máximo y mínimo que la mantisa y el exponente pueden representar.

- La resolución varía en función del rango y se calcula como la diferencia entre dos representaciones sucesivas.
- Es crucial tener en cuenta la resolución en diferentes puntos del rango para comprender la precisión de la representación en punto flotante.

Mantisa Fraccionaria

- En el caso de una mantisa fraccionaria, el rango y la resolución se calculan de manera similar a la representación en punto flotante en Ca_2 .
- La mantisa fraccionaria normalizada comienza con "0,1" y se utiliza para representar una amplia gama de valores fraccionarios.

Formato Final y Bit Implícito

- El formato de representación en punto flotante incluye un bit implícito en la mantisa, que siempre es "1" debido a la normalización.
- La presencia del bit implícito permite representar números con mayor precisión en un espacio limitado.

Normalización y Recta Numérica

- La normalización es un proceso clave en la representación en punto flotante, ya que garantiza que todos los números estén en un formato uniforme.
- La normalización implica desplazar la coma decimal y ajustar el exponente para mantener la precisión y el rango adecuados.
- La presencia del bit implícito influye en cómo se ubican los números en la recta numérica.

Estandar IEEE 754

- El estándar IEEE 754 define los formatos de punto flotante para la representación de números en sistemas informáticos.
- Existen formatos de simple precisión y doble precisión, con diferentes tamaños para la mantisa (23 / 52), el exponente (8 / 11) y los bits de signo (1 / 1).

Casos especiales en estandar IEEE 754

- **Cero positivo y cero negativo:**
 - Mantisa y exponente son cero ($M = 0$, $E = 0$).
- **Infinito positivo e infinito negativo:**
 - Mantisa es cero y exponente está en su valor máximo ($M = 0$, $E =$ todos los bits en 1).
- **NaN (Not a Number):**
 - Mantisa no es cero y exponente está en su valor máximo ($M \neq 0$, $E =$ todos los bits en 1).
- **Valores denormalizados o subnormales:**
 - El exponente es cero y la mantisa no es cero ($E = 0$, $M \neq 0$).
- **Valores fuera de rango:**
 - Si el exponente está fuera del rango representable para el formato utilizado.

Operaciones Aritméticas en Punto Flotante

- Realizar operaciones aritméticas en punto flotante requiere ajustar mantisas, sumar o restar exponentes, y considerar signos.
- Para sumar o restar, se deben alinear los exponentes y ajustar las mantisas antes de normalizar el resultado.
- En operaciones de multiplicación y división, se realizan pasos similares, incluyendo la manipulación de exponentes y mantisas.

Clase 4: Circuitos logicos combinacionales y secuenciales

Circuitos Lógicos Combinacionales:

- **Características:** Los circuitos lógicos combinacionales responden a los valores lógicos presentes en sus entradas en un instante determinado. La salida de estos circuitos se calcula exclusivamente a partir de los valores de las entradas en ese momento.
- **Cambios Instantáneos:** Si se modifica una entrada en un circuito combinacional, la salida se altera inmediatamente en respuesta a ese cambio.

- **Independencia del Pasado:** Los valores previos de las entradas no tienen ningún efecto en los valores actuales de las salidas.

Circuitos Lógicos Secuenciales:

- **Dependencia del Estado Interno:** A diferencia de los circuitos combinacionales, en los circuitos lógicos secuenciales, las salidas dependen tanto de las entradas como del estado interno del circuito en ese momento.
- **Estado Interno:** El estado interno de un circuito se refiere a la información o valores lógicos almacenados internamente en el circuito.
- **Capacidad de Almacenamiento:** Los circuitos secuenciales tienen la capacidad de "almacenar" valores lógicos en su estado interno, lo que permite que la salida no solo dependa de las entradas actuales, sino también de los valores almacenados en el circuito.
- **Persistencia de Valores:** Aunque las entradas cambien, los valores almacenados en el estado interno pueden persistir y afectar las salidas incluso después de que las entradas hayan cambiado.

Circuitos Secuenciales: Almacenamiento de Valores Lógicos

Almacenamiento de Valores Lógicos:

- En los circuitos secuenciales, se requiere la capacidad de almacenar valores lógicos a lo largo del tiempo. Esto es fundamental para implementar la lógica de memoria y estado interno en estos circuitos.
- **Salida como Entrada:** En los circuitos secuenciales, la salida de un componente puede servir como entrada para el mismo componente en el siguiente ciclo de operación. Esto permite que la información fluya y se almacene en el circuito.

Realimentación y Retroalimentación:

- **Salida hacia Entrada:** A diferencia de los circuitos combinacionales, donde la salida no influye en las entradas, los circuitos secuenciales permiten que la salida regrese y se utilice como entrada en ciclos posteriores.
- **Ciclos de Realimentación:** Esta realimentación crea ciclos en el circuito, lo que implica que las operaciones y estados anteriores influyan en las operaciones y estados futuros.

Ecuación Lógica y Almacenamiento:

- **Funciones Lógicas y Ecuaciones:** Los circuitos secuenciales pueden describirse mediante ecuaciones lógicas que relacionan las entradas actuales, las salidas actuales y los valores almacenados internamente.
- **Mecanismo de Almacenamiento:** Al actualizar el estado interno con base en una ecuación lógica, los circuitos secuenciales pueden almacenar información y mantenerla a lo largo del tiempo.

Importancia de la Realimentación:

- **Memoria y Estado:** La realimentación en circuitos secuenciales permite que el circuito mantenga y actualice su estado interno. Esto es crucial para crear circuitos con memoria y capacidad de almacenamiento de información.
- **Implementación de Funcionalidades:** La realimentación permite implementar funciones como registros, contadores y máquinas de estado, que son esenciales para muchas aplicaciones en electrónica y sistemas digitales.

Flip-Flop: Componentes de Memoria en Circuitos Secuenciales

Memoria y Biestable:

Los flip-flops son componentes fundamentales en circuitos secuenciales que permiten la creación de memoria de 1 bit. Estos dispositivos son conocidos como biestables debido a sus dos estados posibles de funcionamiento, que se mantienen a menos que las entradas provoquen un cambio.

Dos Estados Estables: Un flip-flop puede estar en uno de dos estados estables: encendido o apagado, 1 o 0, alto o bajo. Estos estados se mantienen hasta que se apliquen las señales apropiadas para cambiarlos.

Clasificación de Flip-Flops:

- **Tipos de Biestables:** Los flip-flops se clasifican en diferentes tipos según cómo respondan las salidas a las señales lógicas en la entrada. Algunos tipos comunes son:
 - **SR (Set-Reset):** Utiliza las señales S y R para establecer y borrar el estado.

- **J-K (Jump-Keep):** Utiliza las señales J y K, que permiten cambiar de estado y mantenerlo.
- **D (Data or Delay):** Utiliza la señal de entrada D para actualizar el estado.
- **T (Toggle):** Utiliza la señal T para alternar entre los dos estados.

Asincrónicos y Sincrónicos:

- **Cambios de Estado:** Los flip-flops pueden clasificarse según cuándo cambian sus salidas en relación con las señales de entrada.
- **Asincrónicos:** Cambian de estado inmediatamente después de que las entradas cambian.
- **Sincrónicos:** Cambian de estado en momentos específicos, determinados por una señal especial llamada reloj.

El Reloj como Señal de Tiempo:

- **Importancia del Orden:** En muchos circuitos secuenciales, el orden de los eventos es crucial para su funcionamiento.
- **Necesidad de Simultaneidad:** En algunos casos, eventos deben ocurrir simultáneamente para lograr la funcionalidad deseada.
- **Reloj:** El reloj es una señal precisa que determina cuándo se realizan eventos en el circuito.

Clase 5: Arquitectura de Von Neumann

Arquitectura de Von Neumann

Generación 1 de Máquinas de Cómputo

En la década de 1940, surgieron las primeras máquinas con características de sistemas de cómputo. Estas máquinas, conocidas como Generación 1, tenían como objetivo resolver cálculos matemáticos complejos en tiempos reducidos. Estaban dirigidas principalmente a resolver problemas matemáticos y se aplicaban en áreas científicas y militares. Utilizaban tecnologías como válvulas electrónicas y dispositivos electromecánicos.

El Modelo de von Neumann - IAS

En la década de 1940, John von Neumann desarrolló el modelo IAS en la Universidad de Princeton. Se destacó por su operación en binario en lugar de decimal. Poseía una memoria única con 4096 palabras de 40 bits. Su conjunto de instrucciones incluía 21 instrucciones de 20 bits, abarcando transferencias de datos, saltos y operaciones aritméticas. La CPU estaba compuesta por un banco de registros, incluyendo el Registro Buffer de Memoria, Registro de Direcciones de Memoria, Registros de Instrucción y Buffer de Instrucción, Registro Contador de Programa, y Registros Acumulador y Multiplicador/Cociente.

Modelo de von Neumann de 5 Subsistemas

La arquitectura de von Neumann se dividía en cinco unidades funcionales: Unidad de Control, Unidad Aritmético-Lógica, Unidad de Memoria, Unidad de Entrada de Datos y Unidad de Salida de Datos. Aunque este modelo tenía cinco unidades, se puede simplificar en tres unidades funcionales, que es la base de muchas máquinas de propósito general en la actualidad.

Modelo de von Neumann de 3 Subsistemas

El modelo de von Neumann se simplifica en tres subsistemas clave:

- **Unidad Central de Procesamiento (CPU):** Compuesta por la Unidad de Control y la Unidad Aritmético-Lógica, junto con diversos registros.
 - **Unidad de Entrada/Salida (E/S):** Responsable de introducir y extraer datos e instrucciones.
 - **Unidad de Memoria:** Almacena temporalmente datos e instrucciones, fundamental para el funcionamiento de la máquina.
-

Clase 6: Ciclo de instrucción

Función de la Computadora y Ejecución de Programas

La función principal de una computadora es ejecutar programas, los cuales están compuestos por instrucciones almacenadas en memoria. La CPU (Unidad Central de Procesamiento) procesa estas instrucciones una por una en un orden específico.

Descomposición en Etapas

El procesamiento de instrucciones se puede dividir en dos etapas clave: la búsqueda y la ejecución.

Etapas de Búsqueda

En la etapa de búsqueda, la CPU extrae una instrucción de memoria. Esto se logra utilizando el Contador de Programa (PC), un registro en la CPU que mantiene la dirección de la próxima instrucción a buscar. Después de obtener una instrucción, el PC se incrementa para apuntar a la siguiente instrucción en secuencia.

Etapas de Ejecución

Una vez que se ha obtenido la instrucción, se carga en un registro llamado Registro de Instrucción (IR). La instrucción se presenta en forma de código binario, el cual especifica las acciones que la CPU debe llevar a cabo. La CPU interpreta la instrucción y realiza las operaciones requeridas. Estas operaciones caen en cuatro categorías principales:

- **CPU - Memoria:** Se transfieren datos entre la memoria y la CPU.
- **CPU - E/S:** Se transfieren datos entre la CPU y dispositivos de entrada/salida.
- **Procesamiento de Datos:** La CPU realiza operaciones aritméticas o lógicas en los datos.
- **Control:** Se altera la secuencia de ejecución de las instrucciones.

Ciclo de Instrucción

El ciclo de instrucción consta de dos pasos principales: el ciclo de búsqueda y el ciclo de ejecución. En el ciclo de búsqueda, se obtiene una instrucción de memoria y se almacena en el Registro de Instrucción (IR). Luego, en el ciclo de ejecución, la CPU realiza las operaciones especificadas por la instrucción.

Interrupciones

El ciclo de instrucción se repite continuamente hasta que la máquina se apaga, se produce un error o una instrucción de interrupción detiene la ejecución del programa. En general, este proceso garantiza que las instrucciones se ejecuten en el orden correcto y se procesen de manera eficiente.

Ejemplo: ejecución de un fragmento de programa

Instrucciones del Programa

- Cargar en el registro D el contenido de la posición de memoria 94016.
- Sumar el contenido de la posición de memoria 94116 al registro D y guardar el resultado en D.
- Almacenar el valor del registro D en la posición de memoria 94116.

Codificación de Instrucciones

Cada instrucción está codificada en 16 bits, donde los primeros 4 bits indican la operación a realizar y los siguientes 12 bits indican una dirección de memoria.

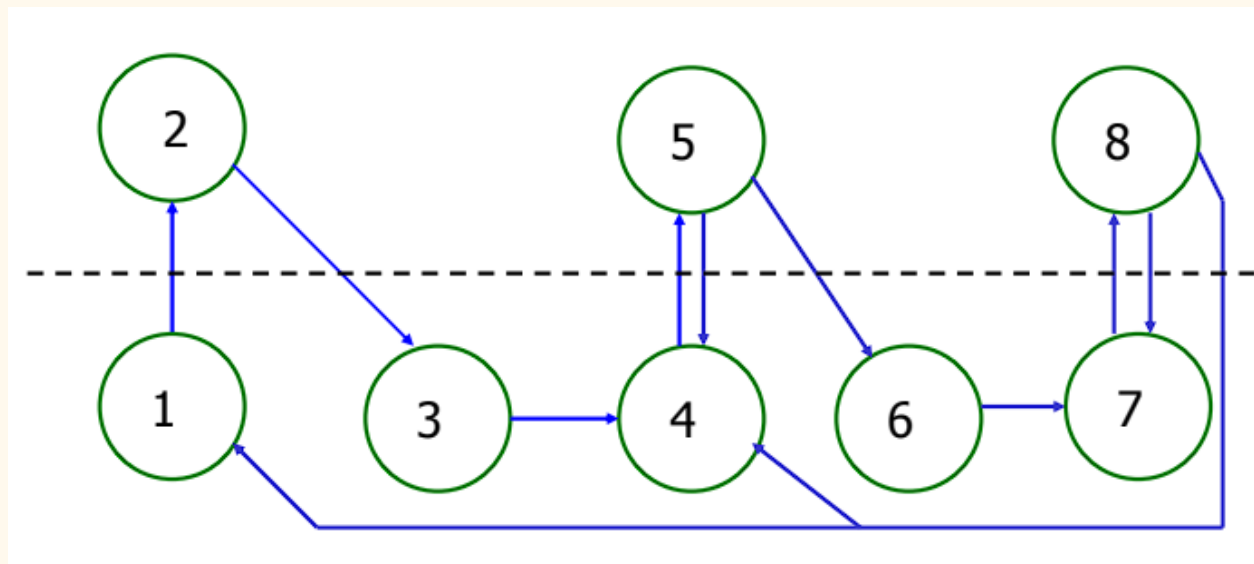
- 00012 (1): Cargar D desde memoria.
- 00102 (2): Almacenar D en memoria.
- 01012 (5): Sumar D con un dato en memoria.

Proceso Paso a Paso

- **Carga de Instrucción:** El contador de programa (PC) contiene la dirección de la primera instrucción (30016), que se carga en el registro de instrucción (IR). Los detalles técnicos como el uso de los registros MAR y MBR se simplifican por ahora.

- **Decodificación de Instrucción:** Los primeros 4 bits en IR indican que el registro D se cargará con un dato de la dirección especificada (94016).
- **Incremento de PC:** El PC se incrementa y se busca la siguiente instrucción en la dirección 30116.
- **Suma y Almacenamiento:** El valor 516 en IR indica que se debe sumar el contenido de la dirección de memoria 94116 con el contenido del registro D. El resultado se almacena en el registro D.
- **Incremento de PC:** El PC se incrementa nuevamen

Diagrama de estados



En la ejecución de instrucciones, un diagrama de estados ayuda a visualizar el proceso de procesamiento de instrucciones en una computadora. A través de una serie de estados interconectados, se representan las etapas clave que ocurren durante la ejecución de un programa.

Etapas del Diagrama de Estados

- **Cálculo de Dirección de Instrucción:** Esta etapa determina la dirección de la siguiente instrucción que se ejecutará. Es crucial para mantener el flujo del programa.
- **Búsqueda de Instrucción:** La CPU lee la instrucción desde su ubicación en memoria hacia la CPU. Esta instrucción será procesada en las etapas posteriores.

- **Decodificación de la Instrucción:** La instrucción obtenida se analiza para determinar el tipo de operación que se realizará y los operandos involucrados.
- **Cálculo de Dirección de Operando:** Si la instrucción implica acceder a un operando en memoria o en entrada/salida, se calcula la dirección correspondiente.
- **Búsqueda del Operando:** En esta etapa, se busca el operando en memoria o en los dispositivos de entrada/salida.
- **Operación sobre los Datos:** La CPU ejecuta la instrucción, realizando la operación especificada sobre los datos y operandos involucrados.
- **Cálculo de Dirección de Resultado:** Si la operación produce un resultado que debe almacenarse o utilizarse posteriormente, se calcula su dirección.
- **Almacenamiento del Resultado:** Finalmente, el resultado de la operación se almacena en la memoria o en otros registros internos de la CPU, según sea necesario.

Organización de los Estados en el Diagrama

- Los estados ubicados en la parte superior del diagrama representan las interacciones entre la CPU y la memoria o los dispositivos de entrada/salida.
- Los estados en la parte inferior del diagrama representan las operaciones internas realizadas dentro de la CPU.

Clase 7: Formatos de instrucción y modos de direccionamiento

En la arquitectura de una computadora, las instrucciones que la CPU ejecuta están representadas por secuencias de bits. Estas secuencias se organizan en campos correspondientes a los diferentes elementos de la instrucción, y esta estructura se conoce como formato de la instrucción.

Representación de Instrucciones

- Cada instrucción en una computadora está representada por una secuencia de bits.
- El formato de la instrucción organiza esta secuencia en campos que corresponden a partes específicas de la instrucción.
- Los campos pueden contener información sobre la operación a realizar, los operandos involucrados y otros detalles.

Representación Simbólica

- Las representaciones binarias de las instrucciones son difíciles para los programadores de comprender.
- Para facilitar la programación, se utilizan representaciones simbólicas, donde los códigos de operación se reemplazan con abreviaturas llamadas mnemónicos.
- Los mnemónicos indican la operación que se llevará a cabo en la instrucción.

Tipos de Instrucciones

- En lenguajes de alto nivel, las instrucciones se expresan de manera concisa utilizando variables y operaciones.
- Las instrucciones de máquina, en contraste, involucran movimientos de datos y uso de registros.
- Una instrucción en lenguaje de alto nivel puede traducirse en varias instrucciones de máquina.

Categorías de Instrucciones de Máquina

- **Procesamiento de Datos:** Incluye operaciones aritméticas y lógicas que manipulan los datos almacenados en registros o memoria.
- **Almacenamiento de Datos:** Se refiere a instrucciones que implican transferencias de datos dentro del sistema, como entre registros y memoria.
- **Instrucciones de E/S:** Son instrucciones que facilitan la transferencia de datos entre la computadora y dispositivos externos.
- **Control:** Estas instrucciones controlan el flujo de ejecución del programa, como saltos condicionales o incondicionales.

Numero de direcciones

En base a la cantidad de campos de direcciones en las instrucciones, existen cinco tipos básicos de máquinas:

- **Máquina de 4 Direcciones:** Formato: COP DirRes DirOp1 DirOp2 DirPróxIns
- **Máquina de 3 Direcciones:** Formato: COP DirRes DirOp1 DirOp2
- **Máquina de 2 Direcciones:** Formato: COP DirOp1 DirOp2

- **Máquina de 1 Dirección:** Formato: COP DirOp/DirRes
- **Máquina de 0 Direcciones (Tipo Pila):** Formato: COP

Diseño del Conjunto de Instrucciones en una Computadora

El diseño del conjunto de instrucciones es esencial para brindar a los programadores el control necesario sobre la CPU de una computadora. Para lograr un conjunto de instrucciones coherente y funcional, se deben considerar varios aspectos clave.

Importancia del Conjunto de Instrucciones

- El conjunto de instrucciones permite a los programadores comunicarse con la CPU y controlar su funcionamiento.
- Aspectos a considerar en el diseño del conjunto de instrucciones:
 - Tipos de operaciones a realizar.
 - Tipos de datos con los que trabajar.

Aspectos del Diseño

- **Formato de Instrucciones:** Se define cómo se estructuran las instrucciones en términos de longitud en bits, número de direcciones y el tamaño de cada campo.
- **Registros:** Se determina cuántos registros pueden ser referenciados mediante instrucciones y cómo se utilizarán.
- **Direccionamiento:** Se establece la forma en que se especifica la dirección de un operando o una instrucción.

Tipos de Operaciones

Se definen diversas operaciones que el conjunto de instrucciones debe ser capaz de ejecutar:

- **Transferencia de Datos:** Movimiento de datos entre memoria y registros (load/store).
- **Operaciones Aritméticas:** Realización de operaciones de suma, resta, multiplicación, etc.
- **Operaciones Lógicas:** Realización de operaciones de lógica booleana como AND, OR, XOR, NOT.
- **Conversión:** Transformación de datos de un formato a otro.
- **E/S:** Instrucciones para entrada y salida de datos.

- **Transferencia de Control:** Instrucciones de salto o bifurcación para controlar el flujo de ejecución.
- **Control del Sistema:** Instrucciones utilizadas por el sistema operativo para el control del sistema.

Tipos de Datos

- Los tipos de datos más comunes incluyen:
 - **Direcciones:** Para acceder a la memoria y registros.
 - **Números:** Enteros, punto fijo y punto flotante.
 - **Caracteres:** Representados en formatos como ASCII o BCD.
 - **Datos Lógicos:** Valores booleanos que representan verdadero o falso.

Modos de Direccionamiento en Instrucciones

El diseño de los modos de direccionamiento en las instrucciones es un elemento esencial para lograr un uso eficiente de los recursos de una computadora. Estos modos determinan cómo se especifican y acceden los operandos durante la ejecución de las instrucciones.

Optimización de la Representación

- Las instrucciones contienen campos para el código de operación y los operandos. La reducción del tamaño de estas especificaciones es crucial para un funcionamiento eficiente.
- ¿Cómo se puede lograr esta optimización?

Métodos de Direccionamiento

Por Registro:

Los operandos se almacenan en registros.

Beneficios:

- Acceso más rápido.
- Menos bits requeridos en la instrucción.

Ejemplo: Uso de registros para variables.

Implícito:

Operandos especificados implícitamente, como en operaciones con acumuladores.

Objetivos:

- Reducción de bits en la instrucción.
- Manejo eficiente de datos, especialmente arreglos.

Tipos de Modos de Direccionamiento

Se presentan varios tipos de modos de direccionamiento, cada uno con sus propias ventajas y desventajas:

- **Inmediato:** El operando se obtiene junto con la instrucción.
- **Directo:** El campo de dirección contiene la dirección efectiva del operando.
- **Por Registro:** Similar al directo, pero usa un registro en lugar de memoria.
- **Indirecto por Memoria:** La instrucción contiene la dirección de la dirección del operando.
- **Indirecto por Registro:** Se especifica el registro que contiene la dirección.
- **Por Desplazamiento:** Combina capacidades de indirecto y directo.

Tipos comunes de Modos de Direccionamiento por desplazamiento

- **Relativo:** La dirección de la instrucción se suma al campo de dirección para obtener la dirección efectiva.
- **De Registro Base:** El registro contiene una dirección de memoria y el campo de dirección es un desplazamiento.
- **Indexado:** Se usa un registro más un desplazamiento para acceder a la memoria.

Modo de Direccionamiento del Stack

- El stack o pila es una estructura lineal de memoria con registros asociados.
 - Útil para el almacenamiento temporal y la administración de llamadas a funciones.
 - Se utiliza un registro puntero de pila para rastrear el tope del stack.
-

Clase 8: Organización de registros, instrucciones

La organización de registros en una CPU es fundamental para el funcionamiento y control de las operaciones. Se dividen en dos categorías: registros visibles al usuario y registros de control y estado.

Registros Visibles al Usuario

- Estos registros son utilizados directamente por el programador y desempeñan diversos roles en el proceso de ejecución.
- **Propósito General:** Son registros utilizados para una variedad de funciones, como almacenar operandos para operaciones específicas.
- **Datos:** Almacenan datos que son procesados y manipulados por el programa.
- **Dirección:** Utilizados para almacenar direcciones de memoria necesarias para acceder a datos.
- **Códigos de Condición:** Almacenan información sobre resultados de operaciones, como banderas de estado (carry, zero, etc.).

Funcionalidades y Uso de Registros Visibles al Usuario

- Los registros de propósito general pueden asignarse a varias funciones, permitiendo flexibilidad en el diseño de instrucciones.
- **Restricciones:** Pueden existir limitaciones en el uso de ciertos registros, como registros dedicados a operaciones específicas.
- **Direccinamiento:** Algunos registros pueden utilizarse para direccionamiento indirecto o específico.
- **Especialización vs. Propósito General:** La elección entre registros especializados y de propósito general impacta en la longitud de las instrucciones y en la flexibilidad del programador.
- **Registro de Acumulador:** Ejemplo de registro especializado que ahorra bits y limita la flexibilidad.

Registros de Control y Estado

- Estos registros son utilizados por la Unidad de Control (UC) para supervisar y gestionar el funcionamiento de la CPU.
- Controlan la operación de la CPU y su estado interno, pero no son visibles ni manipulables directamente por el programador.

La elección de cómo organizar los registros en una CPU es crucial para lograr un equilibrio entre la eficiencia de las operaciones y la flexibilidad del programador. La asignación de registros a funciones específicas puede simplificar las instrucciones y optimizar el rendimiento, pero al mismo tiempo limita la capacidad del programador para adaptarse a diferentes situaciones.

Número de Registros y Longitud de Registros

La cantidad y longitud de los registros en una CPU son aspectos cruciales que impactan en la eficiencia y funcionalidad del sistema.

Número de Registros y su Impacto

- El número de registros influye en el tamaño de las instrucciones.
- Mayor cantidad de registros requiere más bits para especificarlos en las instrucciones.
- Pocos registros pueden resultar en más referencias a memoria.
- Un número óptimo de registros se sitúa entre 8 y 32; agregar más no proporciona mejoras significativas y aumenta el tamaño de las instrucciones.

Longitud de los Registros

- **Registros de Direcciones:** Deben ser capaces de almacenar la dirección más grande posible en el sistema.
- **Registros de Datos:** Deben ser capaces de almacenar la mayoría de los tipos de datos que se manejan.
- Algunas máquinas permiten utilizar dos registros contiguos como uno solo para almacenar valores de doble longitud.

Bits de Condición (Banderas)

- Los bits de condición, también conocidos como banderas, son establecidos por la CPU como resultado de operaciones.
- Pueden ser utilizados por instrucciones de bifurcación condicional para tomar decisiones en el flujo del programa.
- Generalmente, no son alterados directamente por el programador.

Registros de Control y Estado

- Estos registros son empleados por la CPU para controlar su funcionamiento interno y son esenciales para la ejecución de instrucciones.
- **Contador de Programa (PC):** Almacena la dirección de la próxima instrucción a ejecutar.
- **Registro de Instrucción (IR):** Contiene la instrucción actual que se está ejecutando.
- **Registro de Dirección de Memoria (MAR):** Almacena la dirección de memoria para acceder a datos.
- **Registro Buffer de Memoria (MBR):** Temporalmente almacena datos entre la memoria y la CPU.

Uso de Registros de Control y Estado

- Estos registros son empleados para el movimiento de datos entre la CPU y la memoria.
- Los datos deben presentarse a la Unidad Lógico-Aritmética (ALU) para procesamiento; ésta accede al MBR y a los registros visibles al usuario.
- Además de los registros esenciales mencionados, puede haber registros temporales adicionales para intercambiar datos con el MBR y otros registros visibles.

Organización de Registros en Diferentes CPUs

Las arquitecturas de CPU varían en la forma en que organizan sus registros, lo que influye en cómo se realizan las operaciones y la programación en cada sistema.

CPU PII Intel (Principales Registros)

- **AX (Acumulador):** Es el registro principal en operaciones aritméticas.

- **BX (Puntero Base):** Se utiliza como dirección de memoria base.
- **CX (Contador):** Participa en instrucciones de ciclo, como bucles.
- **DX (Datos):** Interviene en operaciones de multiplicación y división.

Apuntadores y Registros Especiales

- **SI y DI:** Actúan como apuntadores en instrucciones que recorren arreglos o tablas.
- **BP y SP:** Son apuntadores a memoria, pero se enfocan en la zona especial de pila o stack.
- **Los registros E** son versiones de 32 bits de los registros originales.

Organización de Registros en CPU MOTOROLA 68000

- Cuenta con 8 registros de datos de 32 bits.
- Dispone de 9 registros de direcciones.
- Contiene 2 stacks: uno para el usuario y otro para el sistema operativo (S.O.).

Instrucciones y Formatos en la Arquitectura Intel

Las instrucciones en la arquitectura Intel se estructuran en formatos específicos que involucran operandos y destinos, permitiendo la realización de diversas operaciones.

Formato de Instrucción en Intel

- Las instrucciones tienen la forma "instrucción destino, fuente".
- Cada operando (destino o fuente) puede ser un mdd (modo de direccionamiento), un registro de la CPU o un dato inmediato.

Tipos de Instrucciones

- Instrucción mem, reg
- Instrucción reg, mem
- Instrucción reg, reg
- Instrucción reg, inm
- Instrucción mem, inm

Nomenclatura Destino y Fuente

- La nomenclatura de "destino" y "fuente" proviene de que en el movimiento de datos, estos fluyen de derecha (fuente) a izquierda (destino).
- En operaciones de suma, por ejemplo, el resultado se almacena en el lugar del operando izquierdo (destino).

Diseño de Formatos de Instrucción

El diseño de formatos de instrucción involucra consideraciones clave:

- ¿Instrucciones cortas o largas?
- Número de bits por segundo.
- Ancho de banda de memoria.
- Velocidad del procesador en relación con la velocidad de la memoria.
- Instrucciones más cortas hacen que el procesador parezca más rápido.

Consideraciones de Diseño Adicionales

- Suficientes bits para expresar todas las operaciones deseadas.
- Dejar bits libres para futuras expansiones y mejoras.
- Definir la cantidad de bits necesarios para representar datos adecuadamente.

El diseño de formatos de instrucción en la arquitectura Intel se basa en la eficiencia, el rendimiento y la flexibilidad para realizar una variedad de operaciones.

Clase 9: Subsistemas de memoria

La memoria es un componente esencial en una computadora que almacena datos y programas utilizados durante la ejecución. A medida que la velocidad del procesador aumenta más rápido que la velocidad de la memoria, se requiere una jerarquía de memorias para optimizar el rendimiento.

Velocidad del Procesador y Memoria

La velocidad del procesador tiende a duplicarse cada 18 meses sin un aumento significativo en el costo. En contraste, el tamaño de la memoria se cuadruplica cada 36 meses sin cambios drásticos en el precio. Además, la velocidad de la memoria aumenta alrededor de un 10% anualmente.

Brecha entre Procesador y Memoria

Con la creciente brecha entre la velocidad del procesador y la memoria, las arquitecturas de computadoras emplean diferentes tipos de memoria para mitigar este problema. Esto resulta en una jerarquía de memorias, que abarca desde registros rápidos y costosos hasta discos más lentos y económicos.

Coordinación de Tipos de Memoria

La interacción entre los distintos tipos de memoria se coordina para lograr un rendimiento similar al de una memoria única, grande y rápida. Aunque en realidad existen diferentes tipos de memoria trabajando juntos, su cooperación permite una operación eficiente.

Jerarquía de Memoria

La jerarquía de memoria organiza los diversos tipos de memoria según su velocidad y costo. En la cima de esta jerarquía se encuentran los registros ultrarrápidos, mientras que en la base están las memorias secundarias como discos magnéticos y almacenamiento "off line" como CD, DVD y cintas.

Tipos y Características de Memoria

- **Registros:** Son los más rápidos, con un tiempo de acceso en el orden de 1 nanosegundo y un tamaño típico de 1 KB.
- **Caché:** Proporcionan un tiempo de acceso de 5-20 nanosegundos y un tamaño típico de 1 MB.
- **Memoria Principal:** Ofrecen un tiempo de acceso de 60-80 nanosegundos y un tamaño típico de 1 GB.
- **Discos:** Tienen tiempos de acceso de alrededor de 10 milisegundos y tamaños típicos de hasta 160 GB.

Objetivo de la Jerarquía de Memoria

La jerarquía de memoria busca combinar diferentes tecnologías y fundamentos físicos para lograr una alta capacidad de almacenamiento y tiempos de acceso reducidos. Los diferentes tipos de memoria se distribuyen en diferentes ubicaciones dentro del sistema para optimizar su uso.

La organización de la jerarquía de memorias es fundamental para garantizar un equilibrio entre velocidad, capacidad y costo en los sistemas informáticos, permitiendo un rendimiento óptimo para una variedad de aplicaciones y necesidades.

Características de los Subsistemas de Memoria

Los subsistemas de memoria en una computadora presentan diversas características que influyen en su funcionamiento y utilidad.

Duración de la Información

- **Memorias Volátiles:** Ejemplos de estas son las memorias RAM (Random Access Memory), que retienen datos solo mientras la computadora está encendida. Son rápidas y eficientes para el acceso a datos temporales.
- **Memorias No Volátiles:** Ejemplos incluyen discos duros y unidades de cinta. Mantienen los datos incluso cuando se apaga la computadora.
- **Memorias Permanentes:** Aquí se encuentran las memorias ROM (Read-Only Memory) y EPROM (Erasable Programmable Read-Only Memory). Contienen datos permanentes, como instrucciones de arranque.

Modo de Acceso

- **Acceso por Palabra:** Caracteriza a la memoria principal, donde los datos se acceden y almacenan en unidades individuales de palabras.
- **Acceso por Bloque:** Es típico de los discos y la memoria caché, donde los datos se manejan en bloques completos en lugar de palabras individuales.

Velocidad de Acceso y Ciclo

- **Memorias Semiconductoras:** Son memorias basadas en semiconductores, como las RAM. El Tiempo de Acceso es el tiempo que transcurre desde que se inicia la operación de lectura o escritura hasta que se obtiene o almacena el dato. El Tiempo de Ciclo es el tiempo mínimo entre dos operaciones sucesivas. Siempre se cumple que tiempo de ciclo > tiempo de acceso.
- **Memorias Magnéticas:** Implica memorias magnéticas, como los discos duros. El Tiempo de Acceso se compone del tiempo de posicionamiento del cabezal y el tiempo de latencia (además del tiempo de lectura). También se mide la Velocidad de Transferencia en bytes por segundo.

Métodos de Acceso

- **Acceso Aleatorio:** Característico de la memoria principal, el tiempo de acceso a una ubicación no depende de accesos anteriores y es constante.
- **Acceso Secuencial:** Se aplica a unidades de cinta, donde el acceso debe ser en secuencia lineal específica. La velocidad puede variar.
- **Acceso Directo:** Aplicable a discos magnéticos, cada bloque o registro tiene una dirección única basada en su localización física.
- **Acceso Asociativo:** Se refiere a la memoria caché, donde los datos se acceden basados en la coincidencia con una ubicación previamente almacenada.

Estas características influyen en el rendimiento, la flexibilidad y la eficiencia de los diferentes tipos de memoria utilizados en sistemas informáticos. Cada tipo de memoria se adapta a necesidades específicas y contribuye a la jerarquía de memoria que optimiza el rendimiento general del sistema.

Memoria de Acceso Aleatorio (RAM)

La memoria de acceso aleatorio, también conocida como RAM (Random Access Memory), juega un papel fundamental en el funcionamiento de los sistemas informáticos, presentando distintas características y tipos.

Características de la RAM

- **Acceso Aleatorio:** La característica principal de la RAM es su capacidad de acceder a cualquier celda de memoria en el mismo tiempo, sin importar su ubicación en la estructura de memoria. Esto permite una búsqueda eficiente y rápida de datos.
- **Tipos de RAM:**
 - **SRAM (Static RAM):** Se basa en flip-flops y es una memoria estática. Es rápida y no requiere ser refrescada constantemente. Se utiliza en cachés y otros lugares donde la velocidad es crucial.
 - **DRAM (Dynamic RAM):** Emplea transistores y es una memoria dinámica. Almacena más información que la SRAM en la misma área, pero requiere refrescar los datos periódicamente debido a la pérdida gradual de carga en los capacitores. Es común en la memoria principal.
- **Tecnología DRAM:** Las DRAM almacenan información en capacitores. Aunque los capacitores son más pequeños que los flip-flops de la SRAM, la necesidad de refrescar la carga en ellos puede afectar su rendimiento.
- **Aplicación en ROM:** También se aplica el acceso aleatorio a las ROM (Read-Only Memory), permitiendo acceder a los datos almacenados en ellas de manera no secuencial.

Diferencias entre SRAM y DRAM

- **Capacidad y Tamaño:** Las DRAM pueden almacenar más datos que las SRAM en la misma área debido a la pequeña escala de los capacitores en comparación con los flip-flops.
- **Refresco:** Las DRAM requieren refrescarse periódicamente debido a la pérdida gradual de carga en los capacitores. En contraste, las SRAM no requieren refresco y son más rápidas.

Usos de SRAM y DRAM

- **DRAM en Memoria Principal:** La DRAM se utiliza como memoria principal en una computadora, proporcionando la capacidad necesaria para almacenar y acceder a programas y datos en uso.

- **SRAM en Caché:** La SRAM se emplea en la memoria caché, un almacenamiento de alta velocidad que actúa como intermediario entre la CPU y la memoria principal, mejorando el rendimiento al almacenar datos temporalmente.

Organización de Memoria de Semiconductores

La organización de la memoria de semiconductores, como las memorias RAM, se basa en celdas de memoria y presenta características fundamentales que permiten su funcionamiento eficiente.

Características Compartidas por las Celdas de Memoria

- **Dos Estados Estables:** Las celdas de memoria de semiconductores pueden representar los estados de "uno" (1) y "cero" (0), lo que permite almacenar información binaria.
- **Operaciones de Escritura y Lectura:** Las celdas de memoria pueden ser escritas al menos una vez para almacenar datos y leídas para conocer el estado actual.
- **Terminales Funcionales:** En general, cada celda de memoria de semiconductores tiene tres terminales funcionales:
 - **Selección:** Selecciona una celda específica de memoria.
 - **Control:** Indica si la operación es de lectura o escritura.
 - **Escritura/Lectura de Datos:** Lleva a cabo la lectura o escritura de los datos en la celda.

Organización de la Memoria y el Chip

- **Memoria de 1 Bit:** Se puede construir una memoria de 1 bit utilizando flip-flops. Los registros simples de n bits se construyen a partir de flip-flops.
- **Memorias Más Grandes:** Para construir memorias de mayor capacidad, se necesita una organización que permita direccionar palabras individuales.
- **Arreglo de Chips:** Cada chip contiene un arreglo de celdas de memoria. Se han empleado dos enfoques organizacionales: 2D y 2½D.

Organización 2D

- El arreglo se organiza en 2W palabras de B bits cada una. Cada línea horizontal se conecta a cada posición de memoria, seleccionando un renglón.
- Las líneas verticales conectan cada bit a la salida.

- Un decodificador en el chip tiene salidas para las entradas del bus de direcciones.

Organización 2½D

- El arreglo es cuadrado y funciona de manera similar al enfoque 2D.
- Los bits de una misma palabra están dispersos en distintos chips.
- La dirección se divide en selección de renglón y selección de columna, utilizando dos decodificadores.

Comparación entre 2D y 2½D

- En 2D, todos los bits están en el mismo chip, lo que puede dificultar el uso eficaz de circuitos correctores de error.
- En 2½D, los bits de una misma palabra se encuentran en diferentes chips, reduciendo la probabilidad de error.
- La decodificación separada de filas y columnas en 2½D reduce la complejidad de los decodificadores.

Problemas y Soluciones en Subsistemas de Memoria

Los subsistemas de memoria a menudo enfrentan desafíos en cuanto a la organización y tecnología utilizada. Estos problemas tienen soluciones que buscan mejorar la eficiencia y el rendimiento.

Problema 1: Cobertura Fragmentada

- Cada módulo de memoria abarca una parte del espacio de direccionamiento necesario, pero solo cubre una fracción de la palabra.
- **Solución:** Para abordar esto, se puede cubrir un rango de direcciones con módulos de memoria en serie. Cada módulo ocupa direcciones diferentes.

Problema 2: Capacidad Insuficiente de los Módulos

- Aunque la longitud de la palabra es la adecuada, los módulos de memoria no tienen la capacidad deseada.
- **Solución:** Una solución es combinar múltiples módulos para cubrir un rango de direcciones y lograr la capacidad deseada.

Nuevas Tecnologías en RAM

- **DRAM Básica y Enhanced DRAM:** Aunque la DRAM básica se ha mantenido similar, se han desarrollado variantes mejoradas como la Enhanced DRAM que incorpora una pequeña SRAM para almacenar la última línea leída, actuando como una caché.
 - **Cache DRAM:** Esta variante contiene una SRAM más grande que se puede utilizar como caché o como buffer serial.
 - **Synchronous DRAM (SDRAM):** Presente en DIMMs actuales, la SDRAM se sincroniza con un reloj externo. Al enviar una dirección a la RAM, esta encuentra los datos y la CPU espera a que se completen. El modo Burst permite a la SDRAM trabajar en bloques y mover datos en función del reloj del sistema.
 - **Beneficios de SDRAM:** La CPU puede realizar otras tareas mientras espera a que se completen las operaciones de acceso a memoria. La SDRAM permite un flujo más eficiente de datos, mejorando el rendimiento general del sistema.
-

Clase 10: Memoria caché y memoria externa

La memoria caché desempeña un papel crucial en mejorar el rendimiento de las computadoras al enfrentar la disparidad de velocidad entre las CPU y las memorias principales.

Desafío de Velocidad en Memoria

- Históricamente, las CPU han sido más rápidas que las memorias, y la brecha de velocidad se ha ampliado debido a la creciente capacidad de los chips.
- Los diseñadores de CPU han enfocado en el aumento de velocidad, mientras que los de memoria se han centrado en ampliar la capacidad.

Retraso en el Acceso a Memoria

- Tras emitir una solicitud de lectura a la memoria, la CPU experimenta múltiples ciclos de reloj antes de recibir los datos solicitados.
- Durante las etapas de instrucción, la CPU accede a la memoria para buscar instrucciones y operandos, lo que limita la velocidad de ejecución por el ciclo de memoria.

Necesidad de Memoria Caché

- La solución no es tecnológica, sino económica. Las memorias rápidas podrían construirse, pero deben estar en el chip de la CPU para lograr la máxima velocidad.
- Acceder a la memoria a través del bus del sistema es más lento.

Principios de Localidad

- **Principio de Localidad Espacial:** Al acceder a una palabra de memoria, es probable que el próximo acceso sea en su vecindad.
- **Principio de Localidad Temporal:** Si una posición de memoria se accede, es probable que se acceda nuevamente en un corto período.

Localidad Espacial y Temporal

- La localidad espacial se basa en la ejecución secuencial del código y la agrupación de variables relacionadas, como matrices o pilas.
- La localidad temporal se refleja en la formación de bucles, subrutinas y pilas.

Función de la Memoria Caché

- La memoria caché combina una pequeña cantidad de memoria rápida con una cantidad mayor de memoria más lenta para lograr velocidades de acceso casi rápidas.
- Al acceder a una palabra, la caché también trae palabras vecinas de la memoria principal, lo que permite un acceso más rápido en futuras operaciones.

Memoria Caché: Aciertos y Fallos en la Mejora del Rendimiento

En la implementación de la memoria caché, es fundamental entender los conceptos de aciertos y fallos, así como la razón detrás de la presencia de múltiples niveles de caché. Estos aspectos son esenciales para comprender cómo la caché mejora el rendimiento general de los sistemas informáticos.

Aciertos y Fallos en la Caché

- La efectividad de la caché se mide en términos de la frecuencia de aciertos, que se refiere al número de veces que la caché contiene los datos requeridos por el procesador.

- Un acierto de caché se produce cuando los datos que la CPU necesita están almacenados en la caché, permitiendo un acceso rápido a los mismos.
- Por otro lado, un fallo de caché ocurre cuando los datos buscados no se encuentran en la caché y deben obtenerse de la memoria principal, lo que implica un acceso más lento.

Importancia de los Niveles de Caché

- La memoria caché suele tener múltiples niveles, como **L1** y **L2**, para mejorar aún más el rendimiento.
- La distribución de porcentajes entre aciertos y fallos puede ser 90% y 10%, respectivamente.
- La estrategia es concentrarse en mejorar el 90% de aciertos mediante técnicas eficientes, ya que el pequeño porcentaje restante de fallos es menos crítico.

La memoria caché es crucial para reducir la brecha de velocidad entre la CPU y la memoria principal.

Memoria Externa

La memoria externa, en particular los discos magnéticos y ópticos, juega un papel crucial en el almacenamiento de datos y la transferencia de información en sistemas informáticos.

Tipos de Memoria Externa

- Los discos magnéticos, como los discos duros, almacenan datos en platos cubiertos con material magnético, utilizando principios de magnetización y lectura/escritura.
- Los discos ópticos, como los CD-ROM, DVD y Blu-ray, utilizan tecnologías ópticas para almacenar datos en superficies reflectantes.

Principios Físicos y Mecánica de Lectura/Escritura

- Pequeñas áreas del disco son magnetizadas en diferentes direcciones por un transductor.
- Durante la lectura/escritura, la cabeza transductora (bobina) permanece estacionaria mientras el plato gira.
- En los discos ópticos, la lectura se basa en la detección de cambios en la reflectividad de la superficie.

Organización de los Datos en Discos

- Los datos en los discos se organizan en anillos concéntricos llamados pistas o tracks.
- Cada pista se divide en sectores, que contienen información como encabezados, datos y códigos de error.
- La organización en cilindros, que agrupa pistas en platos diferentes, reduce el movimiento de cabezas y mejora la velocidad de respuesta.

Velocidad de Giro y Tiempos de Acceso

- Los discos giran a velocidad angular constante (CAV), lo que afecta la velocidad de transferencia y el tiempo de acceso.
- El tiempo de seek implica mover la cabeza a la pista correcta, el tiempo de latencia es el tiempo de espera por un sector específico y el tiempo de acceso es la suma de ambos.
- La capacidad del disco se calcula considerando zonas y sectores.

Formato y Capacidad del Disco

- El formato define la cantidad, tamaño y función de los campos en cada pista.
- El hardware establece el tamaño de sector fijo, mientras que el software lo determina en sistemas operativos.
- **Calculo de capacidad:** $\text{Capacidad} = (\text{bytes/sector}) * (\text{sectores/pista}) * (\text{pistas/superficie}) * \text{\#superficies}$

La memoria externa, en forma de discos magnéticos y ópticos, desempeña un papel vital en el almacenamiento y la transferencia de datos en sistemas informáticos.

Clase 11: Almacenamiento optico, monitores e impresoras

RAID (Redundant Array of Independent/Inexpensive Disks) es una tecnología que agrupa varios discos físicos en una sola unidad lógica, proporcionando mayor rendimiento, capacidad y, en algunos casos, redundancia. Existen varios niveles de RAID, cada uno con sus propias características y aplicaciones.

RAID 0: Rendimiento sin Redundancia

- No proporciona redundancia (copia de datos en caso de fallo).
- Distribuye los datos en los discos para aumentar la velocidad de lectura/escritura.
- Requiere al menos 2 discos.
- Capacidad total es la suma de las capacidades de los discos.

RAID 1: Redundancia Espejo

- Ofrece redundancia.
- Los datos se duplican en dos discos para mayor seguridad.
- Requiere al menos 2 discos.
- Capacidad total es la del disco más pequeño.

RAID 3: Paridad para Protección

- Divide los datos en discos y utiliza un disco adicional para almacenar información de paridad.
- Ofrece tolerancia a fallos de un disco.
- Requiere al menos 3 discos.
- Capacidad total es la suma de las capacidades de los discos menos un disco (paridad).(ultimo disco).

RAID 5: Paridad Distribuida

- Distribuye los datos y la información de paridad en los discos.
- Ofrece tolerancia a fallos de un disco.
- Requiere al menos 3 discos.
- Capacidad total es la suma de las capacidades de los discos menos un disco (paridad).

RAID 6: Doble Paridad

- Similar a RAID 5, pero con dos discos de paridad para mayor tolerancia a fallos.
- Ofrece mayor protección ante fallos de dos discos.
- Requiere al menos 4 discos.
- Capacidad total es la suma de las capacidades de los discos menos dos discos (paridad).

RAID 0-1: Rendimiento y Redundancia Combinados

- Combina RAID 0 y RAID 1.
- Ofrece rendimiento y redundancia.
- Requiere al menos 4 discos.
- Capacidad total es la mitad de la suma de las capacidades de los discos.

La tecnología RAID permite adaptarse a diferentes necesidades de rendimiento y seguridad en el almacenamiento de datos. Los distintos niveles ofrecen opciones desde mayor velocidad hasta alta redundancia, lo que permite a los usuarios elegir la configuración más adecuada para sus requerimientos específicos.

CD-ROM y Almacenamiento Óptico

El CD-ROM (Compact Disc-Read Only Memory) es un medio de almacenamiento óptico basado en el CD de audio. Está compuesto por una capa de policarbonato revestida con una capa altamente reflectiva, generalmente de aluminio. Los datos se almacenan en forma de pequeñas depresiones llamadas "pits", que son leídas mediante un láser.

Velocidades en CD-ROM

- Velocidad lineal constante: 1,2 m/seg
- Una pista en espiral de aproximadamente 5,6 km de largo.
- Velocidad angular variable: 200 a 530 rpm.
- Velocidad de reproducción: 75 sectores por segundo.
- La velocidad se expresa en múltiplos, como 24X.

Capacidad de un CD-ROM

- Capacidad aproximada de 650 MB.
- Cálculo: $2 \times 75 \times 60 \times 74 \text{ min} = 666,000 \text{ KB} \approx 650 \text{ MB}$.

Acceso al CD-ROM

- Proceso dificultoso.
- Movimiento de la cabeza lectora a la posición requerida.
- Establecimiento de la velocidad correcta.

- Lectura de la identificación (dirección) del sector.
- Ajuste a la posición requerida.

Pros y Contras del CD-ROM

Pros:

- Gran capacidad de almacenamiento.
- Fácil de producir en masa.
- Removible.
- Robusto.

Contras:

- Costoso en pequeñas cantidades.
- Velocidad de acceso lenta.
- Solo lectura.

Otros Ópticos

- CD-Recordable (CD-R): Permite grabar datos una vez.
- CD-RW: Permite escribir y borrar datos múltiples veces.
- Ambos formatos son compatibles y su costo ha disminuido con el tiempo.

Los medios de almacenamiento óptico como el CD-ROM han sido ampliamente utilizados para la distribución y almacenamiento de datos, pero su acceso y velocidad limitados han llevado al desarrollo de soluciones más avanzadas en la era digital.

DVD y Almacenamiento Óptico

DVD (Digital Versatile Disk)

- Significado: Inicialmente "Digital Video Disk", pero también "Digital Versatile Disk".
- Utilizado para películas y almacenamiento en computadoras.
- Mayor capacidad y versatilidad en comparación con CD.
- Puede contener películas, datos, programas y más.

Tecnología DVD

- Capacidad: Mayor capacidad en comparación con CD.
- Multi-capa: Puede tener múltiples capas de almacenamiento.
- Compresión MPEG: Permite almacenar películas y otros contenidos.
- Estandarización: DVD es un estándar establecido para almacenamiento óptico.

Comparación CD vs. DVD

- Los DVD tienen mayor capacidad de almacenamiento que los CD.
- Los DVD pueden contener tanto datos como películas, mientras que los CD están más limitados en este aspecto.

Diferencia entre Blu-ray y DVD

- **Blu-ray:** Mayor capacidad que el DVD, lo que permite almacenar videos de alta definición y contenido de mayor calidad.
- **DVD:** Capacidad limitada en comparación con Blu-ray, adecuado para contenido de definición estándar.

Cinta Magnética

- Utilizada para almacenamiento de datos en serie.
- Acceso lento en comparación con medios ópticos.
- Muy económica y adecuada para copias de seguridad y archivo a largo plazo.

Los DVD han sido una evolución significativa en el almacenamiento óptico, ofreciendo mayor capacidad y versatilidad en comparación con los CD. Además, la tecnología Blu-ray ha llevado el almacenamiento de alta definición a otro nivel. Por otro lado, la cinta magnética sigue siendo una opción económica para el almacenamiento a largo plazo y las copias de seguridad.

MODEM (Modulador, Demodulador)

Concepto del MODEM

Modulador y Demodulador.

- Convierte señales binarias (0 y 1) en tonos de audio para transmitir datos a través del sistema telefónico.
- El sistema telefónico tiene un rango de respuesta de frecuencia entre 50 y 3500 Hz.

Tasa de Bits por Segundo (bps) y Tasa Baud (Baud Rate)

- Tasa de Bits por Segundo (bps): Indica la cantidad de bits enviados por segundo.
- Tasa Baud (Baud Rate): Indica el número de cambios de señal por segundo.
- La tasa máxima de Baud para el sistema telefónico es 2400.

Envío de Varios Bits por Baud

- Es posible enviar varios bits por Baud utilizando diferentes frecuencias para representar señales.
- Ejemplo: Enviar cuatro señales diferentes 2400 veces por segundo, lo que permite enviar dos bits por Baud.
- La tasa de bits por segundo se calcula como la tasa de Baud multiplicada por el logaritmo base 2 del número de señales posibles.

"Smart" Modems

- También llamados "Hayes compatible".
- La computadora controla varios aspectos del modem, como el discado, la tasa de bits, programas de contestador, rellamada automática, etc.
- Pueden realizar compresión de datos para mejorar la eficiencia de la transmisión.
- Los modems "smart" generalmente tienen una tasa de Baud máxima de 2400 y una tasa de bits máxima de 57600 bps (56K).

Los MODEM desempeñaron un papel crucial en el pasado para transmitir datos a través de líneas telefónicas. Los avances en tecnología han llevado a la aparición de conexiones más rápidas, como DSL y fibra óptica, que han superado en gran medida las limitaciones de velocidad de los modems tradicionales.

Dispositivos de Entrada de Datos

Teclado y Mouse

- Tasas de entrada lentas.
- Teclado: alrededor de 10 caracteres de 8 bits por segundo.
- Mouse: más rápido, con 1 cambio en los bits de posición X e Y por milisegundo.
- Click de mouse: un bit por 1/10 de segundo.
- El diseño busca reducir partes móviles en los dispositivos de entrada manual.

Dispositivos de Salida de Datos

Monitores de Video

- Pueden ser alfanuméricos o de gráficos.
- Imagen trazada en pantalla línea por línea (raster).
- Puntos en pantalla (píxeles) se marcan con un haz de electrones.
- El haz se desvía horizontal y verticalmente para formar la imagen.
- Mostrado a una velocidad de 50/60 cuadros completos por segundo.
- Resolución Vertical: número de líneas (aproximadamente 500).
- Resolución Horizontal: puntos por línea (aproximadamente 700).
- Puntos por segundo aproximado: $60 \times 500 \times 700 \approx 21$ millones de puntos/s.

Tipos de Video: Terminal y Mapeado en Memoria

- **Terminal:** Monitor de video, memoria de visualización y teclado integrados.
- **Mapeado en memoria:** Monitor de video con memoria de visualización mapeada en la memoria.
- Terminales suelen ser orientados a caracteres con conexión de ancho de banda pequeño (serie).
- Mapeado en memoria permite mostrar imágenes y movimiento con cambios rápidos gracias a una conexión al bus de memoria (ancho de banda grande).

Memoria de Visualización

- **Monitores alfanuméricos:** Almacenan códigos de caracteres en memoria, convertidos en píxeles mediante ROM de caracteres. Varios píxeles se generan por carácter en líneas sucesivas.
- **Monitores gráficos (bitmap):** Cada píxel es representado por bits en memoria. Los monitores en blanco y negro pueden usar un bit por píxel, mientras que en escala de grises o color se requieren varios bits por píxel.

Los dispositivos de entrada y salida son esenciales para la interacción con las computadoras. Los teclados y ratones permiten la entrada de datos, mientras que los monitores de video muestran la información de manera visual.

ROM de caracteres

- Los bits de una línea se leen en serie.
- Acceso repetido a la misma posición horizontal y a posiciones verticales sucesivas.

Controlador de Video (Alfanumérico)

- Contadores cuentan:
 - Los 7 puntos en un carácter.
 - Los 80 caracteres a lo ancho de la pantalla.
 - Las 9 líneas en un carácter.
 - Las 64 filas de caracteres de arriba hacia abajo.

Controlador de Video (True Color)

- La memoria debe almacenar 24 bits por píxel para una resolución de 256 niveles.
- A una velocidad de 20 millones de puntos por segundo, el ancho de banda de memoria es considerable.
- Se requiere espacio para la RAM de video.

Los controladores de video son componentes esenciales para generar la salida visual en monitores. Los controladores alfanuméricos y de color verdadero presentan desafíos técnicos relacionados con el almacenamiento y el ancho de banda de memoria.

Impresoras de Impacto

- Carácter formado mediante:
 - Margarita.
 - Cinta.
- Utilizan matriz de puntos para armar caracteres.
- Punzones manejados por solenoides golpean una cinta entintada para marcar el papel.
- Baja resolución debido a la cantidad limitada de punzones.

Impresión con Matriz de Puntos

- Imprime una columna a la vez.
- Puede utilizar una ROM de caracteres para generar la salida.
- La ROM se lee en paralelo por columna, a diferencia de la lectura en serie por fila en el video alfanumérico.

Impresora Láser

- Imprime páginas completas.
- Resolución de 300 a 1200 puntos por pulgada (dpi).

Impresora Ink-Jet

- Transductor ultrasónico lanza pequeños chorros de tinta a los puntos correctos en el papel.
- Disponible en blanco/negro y color.
- Ofrece un equilibrio entre precio y resolución.

Tecnologías Ink Jet

- Burbuja Térmica
- Piezoeléctrica

Las impresoras han evolucionado con tecnologías de impacto y no impacto, como las de matriz de puntos, láser e ink-jet. Cada tecnología tiene sus ventajas y desventajas en términos de velocidad, resolución y costo.