ORGANIZACIÓN DE COMPUTADORAS

UNIDAD 1 – COMPUTADORAS DIGITALES

Conceptos Introductorios

Cuando hablamos de la **Arquitectura** de computadoras hacemos referencia a los atributos de un sistema que son visibles para un programador, o, en otras palabras, son aquellos atributos que tienen un impacto directo en la ejecución lógica de un programa.

En cambio, la **Organización** de computadoras refiere a las unidades funcionales y sus interconexiones, que dan lugar a especificaciones arquitectónicas. Entre los atributos de organización se incluyen detalles de hardware invisibles al programador tales como señales de control, interfaces entre el computador y los periféricos y la tecnología de memoria usada. Se puede decir que una arquitectura puede perdurar muchos años, pero la organización cambia con el avance de la tecnología.

¿QUE ES UNA COMPUTADORA?

Es una maquina digital porque se maneja con señales eléctricas y la información se representa por medio de dos valores (0 Y 1) y sincrónica porque todas las operaciones internas se realizan en instantes de tiempos definidos y controlados por un reloj central. Posee capacidad de cálculo numérico y lógico mediante mediante la ALU, que es la encargada de realizar las operaciones. Tiene comunicación con el mundo exterior mediante el módulo de E/S que permite conectar dispositivos periféricos. Todas estas funciones están controladas por la Unidad de Control.

ESTRUCTURA

Es el modo en el cual los componentes se relacionan entre sí. La computadora cuenta con 4 componentes principales:

* Unidad Central de Procesamiento (CPU): controla el funcionamiento de la computadora y lleva a cabo sus funciones de procesamiento de datos.
* Memoria Principal: almacena datos.
* Módulo de Entrada/Salida: transfiere datos entre la computadora y el mundo exterior.
* Sistema de Interconexión: es un mecanismo que proporciona la comunicación entre la CPU, memoria y E/S.

FUNCION

Es la operación de los componentes individuales como parte de la estructura. Las funciones básicas de una computadora son 4:

* Procesamiento de Datos: el procesador tiene que ser capaz de procesar datos. Estos datos pueden adoptar una gran variedad de formas, y el rango de los requisitos de procesado es amplio, pero, hay sólo unos pocos métodos o tipos fundamentales de procesado de datos

(La CPU efectúa operaciones aritméticas o lógicas de datos).

* Almacenamiento de Datos: es esencial que una computadora almacene datos, incluso si la computadora está procesando datos al instante los tiene que guardar temporalmente, al menos aquellos datos con los que está trabajando en un momento dado. También, la computadora lleva a cabo una función de almacenamiento de datos a largo plazo

(Datos pueden transferirse entre memoria y CPU)

* Transferencia de Datos: (Datos pueden transferirse entre CPU y entrada/salida) Cuando se reciben o se llevan datos a un dispositivo que está directamente conectado con la computadora, el proceso se conoce como E/S y el dispositivo recibe el nombre de periféricos. El proceso de transferencia de datos a largas distancias se lo conoce como comunicación de datos.
* Control: debe haber un control de las últimas tres funciones. Este control es ejercido por los entes que proporcionan a la computadora instrucciones. Una unidad de control gestión los recursos de la computadora y dirige las prestaciones de sus partes funcionales en respuesta a las instrucciones .

(Alterar la secuencia de ejecución de instrucciones)

Evolución Histórica de la Computadoras

* **Primera Generación: Tubos de vacío**
* Computadoras construidas por válvulas de vacío. Programadas manualmente por llaves (6000). Disipaban gran cantidad calor y ocupaban una superficie muy amplia. Las tareas se ejecutaban en forma secuencial y las operaciones de E/S estaban encadenadas en el tiempo.
* **ENIAC (1943-1943):** Construida en la Universidad de Pennsylvania, por Eckert and Mauchley**.** Primera computadora de propósito general. Decimal
* **EDSAC (1949):** Construida en la Universidad de Pennsylvania. Programa almacenado. Binaria.
* **IAS (1946):** creado por von Neumann, era una computadora de programa almacenado. La estructura general de esta computadora contada con: Una MEMORIA PRINCIPAL, una ALU, una UC (IR, IBR, PC, MAR) y un dispositivo de E/S.
* **UNIVAC I (1949):** Primer computadora comercial. Primera en utilizar un compilador para traducir idioma de programa en idioma de máquinas. Máquina decimal con 12 dígitos por palabra. Principal avance: sistema de cintas magnéticas que podían leerse hacia adelante y hacia atrás. procedimientos de comprobación de errores. Memoria de líneas de retardo de mercurio y tecnología a válvulas de vacío.
* **IBM:** Equipos de procesamiento con tarjetas perforadas. En **1953** se creó el 701, fue el primer computador con programas almacenados de IBM, tenía aplicaciones científicas. En **1955** se creó el 702 que tenía aplicaciones de gestión.

En **1964** sustituyen la serie 7.000 (no compatibles), eran estructuras de computadoras multiplexadas. Coste creciente. Tamaño de memoria creciente. Número creciente de puertos de E/S. Velocidad creciente.

* **DEC PDP-8 1964**: Primer minicomputador. No necesitaba una habitación con aire acondicionado. Tamaño pequeño. Contenía estructura de BUS.
* **Segunda Generación: Transitores (1947)**
* Sustituyen a los tubos vacíos. Caracterizada por el descubrimiento del transistor que permitió, al ser más pequeño, barato y de menor consumo, la creación de computadoras más accesibles en tamaño y precio.
* Aparecen los primeros lenguajes de programación de alto nivel (FORTRAN, ALGOL, COBOL). Los programas eran hechos a medida por un grupo de expertos.
* **Tercera Generación: Circuitos Integrados:**
* Comienza con la invención del Circuito Integrado de Silicio (varios componentes en un solo bloque) que posibilito la inserción de varios transistores en un solo chip.
* Integración a pequeña escala: desde **1965.** Más de 100 componentes en un chip
* Integración a media escala: desde **1971**. 100-3.000 componentes por chip
* Integración a gran escala: **1971-1977**. 3.000 - 100.000 componentes por chip
* Integración a muy gran escala: desde **1978**. 100.000 - 100 millones de componentes por chip.

Memoria Semiconductora 1970 ¿Qué generación es?

Fairchild fabrica la primera memoria con 256 bits. Tamaño de un núcleo de ferrita (1 bit de almacenamiento de núcleo magnético). Lectura no destructiva. Mucho más rápida que el núcleo. La capacidad se duplica aproximadamente cada año.

Microprocesadores: Intel ¿Qué generación es?

**1971: 4004:** Primer microprocesador de 4 bits. Todos los componentes de la CPU en un solo chip. En 1972 evoluciona al 8008 de 8 bits. Ambos diseñados para aplicaciones específicas.

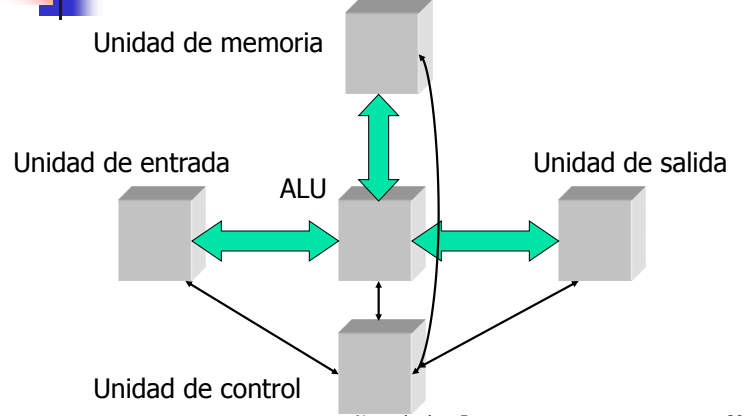
**1974: 8080:** Primer microprocesador de Intel de uso genérico.

Arquitectura Von Neumann

Esta arquitectura define que una computadora está compuesta por *cinco* elementos principales: Una **Unidad Central de Proceso** (procesa los datos), una **Memoria Principal** (donde almacena datos e instrucciones), un **Módulo de Entrada** (donde se proveen los datos e instrucciones), un **Módulo de Salida** (donde se envían los resultados)y un **Sistema de Interconexión** para estos componentes (dirige la operación - control).

La característica principal de estar arquitectura es que la Unidad Central de Proceso está conectada a una única memoria donde se guardan conjuntamente, datos e instrucciones. La memoria es accesible por posición, independientemente si se trata de datos o instrucciones, y la ejecución de las instrucciones se realiza de forma secuencial. En la arquitectura von Neumann, el tamaño de la unidad de datos o instrucciones está fijado por el ancho del bus que comunica la memoria con la CPU, por lo que, si tenemos un procesador con un bus de 8 bits, este tendrá que manejar datos e instrucciones de una o más unidades de 8 bits de longitud. Si se tiene que acceder a una instrucción de mayor tamaño, tendrá que realizar más de un acceso a memoria, lo que implica las siguientes limitaciones:

* Se requieren múltiples accesos para manejar instrucciones complejas.
* Al tener un único bus, no se puede buscar una nueva instrucción hasta que no finalice la que está en curso.
* Esto supone una limitación en la velocidad de operación del procesador.



Aspectos más importantes:

* Utilización del sistema binario: simplifica la implementación de funciones. Disminuye la probabilidad de fallos.
* Instrucciones y datos residen en memoria: Ejecución del programa en forma secuencial. Aumenta la velocidad.
* La memoria es direccionable por localidad sin importar el dato almacenado.

UNIDAD 2 – ARITMETICA DE COMPUTADORAS

Bit: Es el acrónimo de Binary Digit (digito binario) el cual utiliza dos dígitos (el 0 y el 1). Es la unidad mínima de información empleada en informática o en cualquier dispositivo digital. Podemos representar dos valores cuales quiera, basta con asignar uno de esos valores al estado de “apagado” (0) y el otro al estado de “encendido” (1).

Nibble o Cuado: Es el conjunto de 4 bits. Este puede representar una cifra en hexadecimal.

Byte: Es un conjunto de 8 bits contiguos. Un byte comprende el sub-campo direccionable más pequeño del tamaño de palabra natural de la computadora. Esto es, la unidad de datos binarios más pequeña en que la computación es significativa

Palabra: Es una cadena finita de bits que son manejados como un conjunto por la máquina. Su tamaño hace referencia al número de bits contenidos en ella. El tamaño de una palabra se refleja en muchos aspectos de la estructura y las operaciones de las computadoras. Los ordenadores modernos normalmente tienen un tamaño de palabra de 16, 32 o 64 bits.

Teorema Fundamental de la Numeración

El Teorema Fundamental de la Numeración relaciona una cantidad expresada en cualquier sistema de numeración posicional, con la misma cantidad expresada en el sistema decimal. Dice que valor decimal de una cantidad expresada en otro sistema de numeración, está dado por la fórmula:

Texto, Carta

Descripción generada automáticamente

i = posición del digito con respecto a la coma // base = sistema de numeración // n = números de dígitos de la parte entera // m = números de dígitos de la parte fraccionaria

**Sistemas de numeración posicionales**

**SISTEMA DECIMAL:** es un sistema de notación posicional formado por 10 dígitos (b=10) {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}. Por ejemplo, el número 842,9710 = (8\*102 + 2\*101 + 4\*100 + 9\*10-1 + 7\*10-2 ) = 800 + 20 + 4 + 0,9 + 0,07

**SISTEMA BINARIO:** es un sistema de notación posicional formado por 2 dígitos (b=2) {0, 1} a los que se denominan bits. El número 10112= (1\*23 + 0\*22 + 1\*21 + 1\*20 ) = 8 + 0 + 2 + 1 = 1110

**SISTEMA HEXADECIMAL:** está formado por 16 dígitos (b=16) {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F} que representan los decimales del 0 al 15. El número 1A216= (1\*162 + 10\*161 + 2\*160 ) = 256 + 160 + 2 = 41810

**SISTEMA OCTAL:** es un sistema de notación posicional formado por 8 dígitos (b=8) {0,1,2,3,4,5,6,7}. El número2758= (2\*82 + 7\*81 + 5\*80 ) = 128 + 56 + 5 = 18910

Representación de Datos

Las computadoras almacenan datos e instrucciones, los dos aspectos fundamentales de la aritmética del computador son la forma de representar los números (sistema binario) y los algoritmos utilizados para realizar operaciones aritméticas básicas (resta, suma, etc.).

Tipos de Datos

Las computadoras manejan 4 tipos básicos de datos binarios

* Números enteros sin/con signo
* Números reales con signo
* Números decimales codificados en binario (BCD)
* Caracteres

Representación de números enteros

* **Sin signo:** Si el número tiene n bits, puedo representar 2^n = números distintos

El rango va desde +0 a (2n – 1)

* **Módulo y signo BCS:** utiliza el bit más significativo (MSB) para representar el signo (0: positivo, 1: negativo) y el resto del bit para el modulo del bit.

**Rango de representación:** para 8 bits,(-(2n-1 + 1); +(2n-1 – 1)) (rango simétrico)

**Mayor numero:** +12710 = 011111112

**Menor número:** -12710 = 1111111112

**+010** = 000000002  / **-010** = 100000002  (doble representación del cero)

* **Complemento a uno (Ca1) o Complemento a la base reducida:** Utiliza el MSB para el signo. Para los números positivos, los n-1 bits restantes representan el modulo y los números negativos se obtienen invirtiendo todos los dígitos (incluido el de signo) del número positivo. Otro método: el peso que tiene el primer dígito ahora es –(2n-1 –1 ) y el resto de los dígitos con pesos positivos como siempre

11100000= -1x(27 –1) + 1x26 + 1x25= - 127 + 64 + 32 = -31. (N= basen – 1)

**Rango de representación:** para 8 bits,(-(2n-1 + 1); +(2n-1 – 1)) (rango simétrico)

**Mayor numero:** +12710 = 011111112

**Menor número:** -12710 = 100000002

**+010** = 000000002  / **-010** = 111111112  (doble representación del cero)

* **Complemento a dos (Ca2) o Complemento a la base:** Igual que C1 pero el negativo se obtiene en dos pasos: se realiza el C1 al número positivo y al resultado se le suma 1(N= basen).

**Rango de representación:** para 8 bits,(-(2n-1); +(2n-1 – 1)) (rango asimétrico)

**Mayor numero:** +12710 = 011111112

**Menor número:** -12810 = 100000012  (hay un número negativo de mas)

**+010** = 000000002

* **Exceso a 2n-1:** No utiliza el MSB para representar el signo. El número a representar viene dado por su valor más el valor del exceso (en caso de 8 bits,

27 = 128). De esta manera:

+1010 = +10 + 128 = 13810 =100010102

**Rango de representación:** para 8 bits,(-2n-1 ; 2n-1 – 1) (rango asimétrico)

**Mayor numero:** +12710 = 011111112

**Menor número:** -12810 = 000000002

**010** = 0 + 128 = 100000002

**(**Si tengo un decimal y debo pasarlo a binario, le tengo que sumar el exceso. Si tengo un binario y quiero representar el numero en exceso debo restarle el exceso**)**

* **Punto Fijo:** Se considera que todos los números a representar tienen exactamente la misma cantidad de dígitos y la coma fraccionaria está siempre ubicada en el mismo lugar.

La diferencia principal entre la representación en el papel y su almacenamiento en computadora es que no se guarda coma alguna, se supone que está en un lugar determinado.

**Rango:** está dado por el número mínimo representable y el número máximo representable. El número máximo para la parte entera es b^n – 1.

**Resolución:** diferencia entre dos números consecutivos.

**Representación y Error en Punto Fijo**

Al convertir un número decimal a sistema binario tendremos 2 casos:

* Sin restricción en la cantidad de bits a usar: 3,12510 = 11,0012
* Con restricción, por ejemplo 3 bits para parte entera y 4 bits para parte fraccionaria 3,12510 = 011,00102.

**Ejemplo:**

Convertir 3,210 con distintas restricciones

* 3 bits para parte fraccionaria: 011,0012 = 3,12510 Error = 3,2 – 3,125 = 0,075
* 4 bits para parte fraccionaria: 011,00112 = 3,187510 Error = 3,2 – 3,1875 = 0,0125
* 5 bits para parte fraccionaria: 011,001112 = 3,2187510 Error = 3,2 – 3,21875 = 0,01875

El error más pequeño es 0,0125 entonces 3,1875 es la representación más cercana a 3,2 y podría utilizar sólo 4 bits para la parte fraccionaria.

* **Decimal Codificado en Binario (BCH):** Es un estándar para representar número decimales en sistema binario, en donde cada digito decimal es codificado con una secuencia de 4 bits. Con esta codificación especial de los dígitos decimales en el sistema binario, se pueden realizar operaciones aritméticas como suma, resta, multiplicación y división de números en representación decimal *sin perder en los cálculos la precisión ni tener las inexactitudes* en que normalmente se incurre con las conversiones de decimal a binario puro y de binario puro a decimal.

Los dígitos decimales se convierten uno a uno en binario. Para expresar un digito decimal se utilizará siempre 4 bits, esto es así porque es el número de bits necesario para representar el valor 9, el número más alto que se puede representar en BCD. Se asocia cada digito con si valor de binario puro. *La ventaja del código BCD frente a la representación binaria clásica es* ***que no hay límite para el tamaño de un número.***

Hay dos maneras comunes para almacenar datos en BCD:

* E/S y periféricos, los números se codifican usando un byte por dígito. Se dice que el número está **desempaquetado**.
* Desempaquetado SIN signo: por cada digito se usan 8 bits, los primeros 4 se completan con 1 y los 4 que sobran con el binario puro.

834= 11111000 11110011 11110100= F8 F3 F4.

* Desempaquetado CON signo: lo mismo que el SIN signo con diferencia que los 4 primeros bits que acompañan al último número son reemplazados por el signo (C16=1100=positivo, D16=1101=negativo).

+834= 11111000 11110011 11000100= F8 F3 C4.

* En cálculo, se reservan 4 bits por dígito. Se dice que el número está **empaquetado.**
* Empaquetado SIN signo: normal. 34= 00110100 (el numero siempre tiene que completar 1 byte)
* Empaquetado CON signo: los últimos 4 bits que acompañan al último digito son el signo. + 834= 10000011 01001100

Suma en BCD

De las 16 representaciones posibles con 4 bits, usamos 10 para los dígitos del 0 al 9, nos sobran 6 combinaciones de 4 bits. En la suma de BCD a cada resultado de los dígitos que sea mayor a 9 se le debe sumar 6 (0110)

* **Sistema Hexadecimal Codificado en Binario (BCH):** Los dígitos hexadecimales se convierten uno a uno en binario. Para expresar un digito hexadecimal se utilizará siempre 4 bits. Se asocia cada digito con si valor de binario puro

Bits de Condición (Banderas)

Son bits que el procesador establece de modo automático acorde al resultado de cada operación realizada. Sus valores permitirán tomar decisiones como: Realizar o no una transferencia de control. Determinar relaciones entre números (mayor, menor, igual).

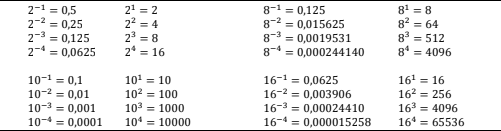
Tipos de Banderas

* **Z (cero):** toma el valor 1 si el resultado son todos bits 0,de lo contrario toma el valor 0.
* **N (negativo):** toma el valor del MSB (1 negativo y 0 positivo).
* **V (overflow):** Si sumamos dos números + y el resultado es – ó si sumamos dos – y el resultado es + hay overflow,si los números son de distintos signos nunca puede haber overflow. Overflow. Si a un No + le restamos un No – y el resultado es – ó si a un No – le restamos un + y el resultado es + hay overflow en la resta, si son del mismo signo nunca puede haber overflow.
* **C (carry):**  toma el valor 1 si existe acarreo en la resta. En operaciones sin signo, toma el valor 1 si el resultado está fuera de rango.

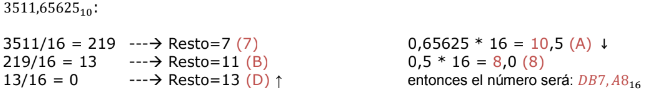
**Sistema de Numeración Posicional**

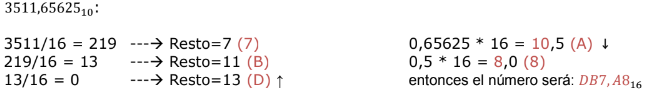
**Sistema Decimal** (b=10) // **Sistema Binario** (b=2) // **Sistema Hexadecimal** (b=16) // **Sistema Octal** (b=8).

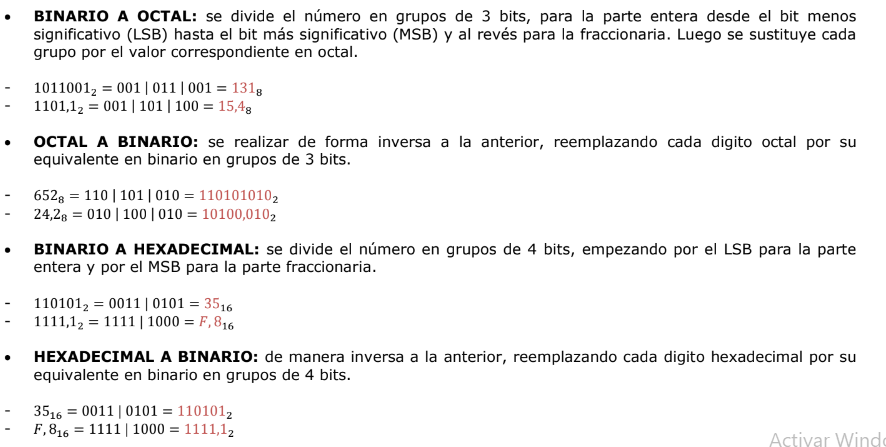
Métodos de conversión entre los sistemas numéricos



* **Decimal a Binario=** método de divisiones en la parte entera: se efectúa sobre el numero el método de visiones sucesivas con respecto a la base hasta que el cociente sea 0. La unión de todos los restos, escritos en orden inverso, forman el numero convertido y multiplicaciones en la parte fraccionaria:se obtiene multiplicando sucesivamente la parte fraccionaria por la base, hasta que sea 0.
* **Decimal a Hexadecimal=** se utiliza el mismo método anterior, pero la parte entera se divide por 16 y la parte fraccionaria se multiplica por 16.

****

* **Decimal a Octal=** método anterior pero en base 8.



Fundamentos de la Representación en Punto Flotante, Normalización y Error de la Representación

Punto Flotante surge de la necesidad de representar números reales y enteros con un rango de representación mayor que el que nos ofrece Punto Fijo y así posibilitar a la computadora el tratamiento de números muy grandes y pequeños. Un número se puede representar de la forma: **± M x B±E**. Donde M es la mantisa, B es la base y E el exponente.

Diagrama, Esquemático

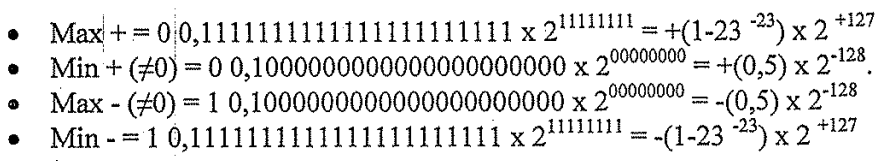
Descripción generada automáticamente

Como la base en binario siempre es 2, es implícita y no necesita almacenarse. Entonces almacenamos M y E en vez del número completo, *ahorramos bits*.

Mantisa y Exponente se almacenan en sistemas de punto fijo (BSS, BCS, Ca1, Ca2, Exceso) y de esto dependerá el rango y la resolución (el rango en punto flotante es más grande que en punto fijo).

Con el objetivo de tener un único par de valores de mantisa y exponente para un número, se introduce la **normalización,** donde todas las mantisas comienzan con 0,1 y el 1 no se almacena (para ahorrar bits), entonces tenemos mas bits para representar. El 1 que no almacenamos se llama bit implícito.

*Ejemplo:* Mantisa BCS de 23 bits, fraccionaria normalizada y exponente Exceso 8 bits entero.



**Ahora, ¿Cómo se escribe un numero en Punto Flotante normalizado?**

1. Se escribe el numero en el sistema propuesto para la mantisa.
2. Se desplaza la coma y se cambia el exponente hasta obtener la forma normalizada (Se le suma tanto como lugares se corrió la coma hacia la izq.).
3. Se convierte el exponente al sistema propuesto para él

*Ejemplo*: -13,510 en el formato del ejemplo anterior.

1. 1 1101,10000 … 00000 = 1 1101,100 … 000 x 20
2. 1 0, 1101100 … 00000 x 24 (Se corre la coma 4 lugares, se suma 4 al exponente)
3. 4 en Ca2 = 00000100; 4 en exceso = 10000100.
4. Y finalmente, lo representamos en la recta:

• Sin bit implícito: 1 10000100 11011000000000000000000

• Con bit implícito: 1 100001000 10110000000000000000000

Resolución: es la diferencia entre dos representaciones sucesivas, y varia a lo largo del rango, no es constante como en el caso del punto fijo.

Error Absoluto: es la diferencia del valor representado y el valor a representar. (Error Absoluto máximo <= Resolución/2)

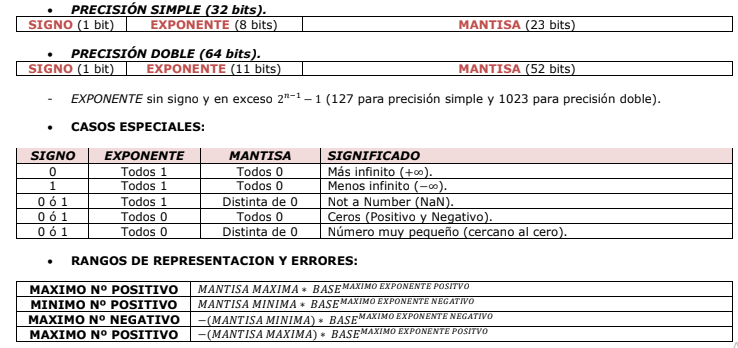
Error Relativo = Error Absoluto / Número a representar.

Estándar IEEE754

Este estándar se desarrolló para facilitar la portabilidad de los programas de un procesador a otro y para alentar el desarrollo de programas numéricos sofisticados. El estándar del IEEE define el formato para precisión simple de 32 bits y para precisión doble de 64 bits. Donde el EXPONENTE es sin signo y en exceso 2n-1 -1 y la MANTISA es fraccionaria normalizada, con la coma después del primer bit que siempre es 1.

Tabla

Descripción generada automáticamente



Texto, Word

Descripción generada automáticamente

UNIDAD 03- LOGICA DIGITAL

Circuito digital

Es en el que están presente dos valores lógicos.

Una **función lógica** es donde cualquier valor o expresión que puede evaluarse como VERDADERO o FALSO

Una **tabla de verdad** es una tabla que muestra el [valor de verdad](https://es.wikipedia.org/wiki/Valor_de_verdad) de una [proposición](https://es.wikipedia.org/wiki/Proposici%C3%B3n) compuesta, para cada combinación de verdad que se pueda asignar

Compuerta Lógica

Son dispositivos electrónicos que pueden realizar distintas funciones con estos dos valores lógicos (0 o 1). Las compuertas básicas son: AND, OR, NOT, NAND (compuerta universal), NOR (compuerta universal), XOR (si son iguales da 0, distintas da 1), XNOR.

Diagrama

Descripción generada automáticamente

|  |  |  |  |
| --- | --- | --- | --- |
| **XNOR** |  |  |  |

Diagrama

Descripción generada automáticamente

Algebra Booleana

Se creó para describir los circuitos que pueden construirse combinando compuertas, donde las variables y funciones solo pueden adoptar valores 0 o 1.

Una función booleana tiene una o más variables de entrada y produce un resultado que depende solo de los valores de dicha variables.

Puesto que una función booleana de n variables tiene 2n combinaciones de los valores de entrada, la función puede describirse totalmente como una tabla de verdad de 2n renglones, donde c/u indica un valor de la función (0 o 1) para cada combinación distinta de valores de entradas.

Tabla

Descripción generada automáticamente

**Implementación de funciones booleanas – Suma de Productos**

Para implementar una función (F) booleana se usa el “método general para determinar una formula lógica dada una tabla de verdad: el método de la suma y el producto **(SOP)**”. Con ella, dada una tabla de verdad, se continua con los siguientes dos sencillos pasos:

1. Verificar las salidas con valor 1

2. Sumar todos los términos.

Así, se obtiene F y se puede hacer el dibujo de los circuitos, usando compuertas lógicas.

Ejemplo:

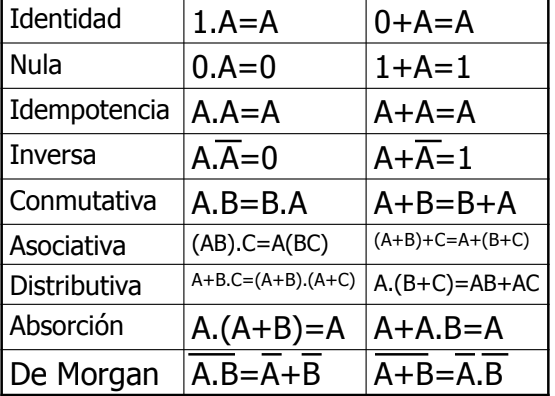
Tabla

Descripción generada automáticamente

**Implementación de funciones booleanas – Producto de Suma**

La misma definición y pasos de la Suma de Productos, pero en ves de mirar los 1 de la salida miramos los 0 de la salida para armar la función lógica.

**Algunas identidades del algebra booleana:**



**¿Que establecen las leyes de Morgan?**

* En la lógica proposicional y álgebra de Boole, las leyes de Morgan declaran las reglas de equivalencia en las que se muestran que dos proposiciones pueden ser lógicamente equivalentes.
* (A.B)° = (A°)+(B°)    ||   (A+B)°=(A°).(B°)
* comentario no importante: permite el cambio del operador de disyunción y viceversa // las proposiciones conjuntivas o disyuntivas pueden estar afirmadas o negadas)

Circuitos Combinacionales

Un circuito combinacional es un conjunto de puertas interconectadas entre sí, cuya salida en un momento dado, es función solamente de la entrada en ese instante. Consiste en n entradas y m salidas. Ejemplos son: Multiplexor, Demultiplexor, Codificador, Decodificador y Sumador. En ningún circuito combinatorio la salida trasporta información hacia la entrada (en la secuencial SI).

Un circuito Combinacional se puede definir de 3 formas:

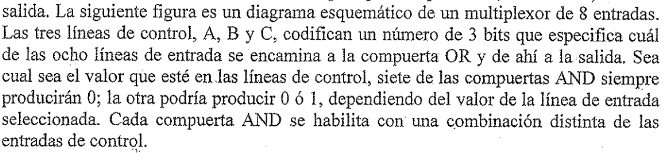
1. **Tabla de verdad:** para cada una de las 2n combinaciones posibles de las n señales de entradas, se enumera el valor binario de cada una de las m señales de salida.
2. **Símbolo Grafico:** define la organización de las interconexiones entre puertas.
3. **Ecuación Booleana**: cada función de salida se expresa como una función booleana de las señales de entrada.

**Multiplexores**

El multiplexor conecta varias entradas con una única salida. En un momento dado, se selecciona una de las entradas para que pase a la salida. Los multiplexores se usan en circuitos digitales para controlar el enrutamiento de señales y datos. También se usan como convertidor de datos paralelos en serie. Un multiplexor es un circuito con 2n entradas de datos, una salida de datos y n entradas de control que seleccionan una de las entradas de datos. La entrada de datos seleccionada se “pasa por compuertas” hacia la salida.

Diagrama, Esquemático

Descripción generada automáticamente



**Codificadores**

Un codificador es un circuito combinacional con 2n entradas y n salidas, cuya misión es presentar en la salida el código binario correspondiente a la entrada activada. Existen dos tipos fundamentales: codificadores sin prioridad y codificadores con prioridad. En el caso de los codificadores sin prioridad, puede darse el caso de salidas cuya entrada no pueda ser conocida: por ejemplo, la salida 0 podría indicar que no hay ninguna entrada activada o que se ha activado la entrada número 0. En cambio, los codificadores con prioridad, cuando existe más de una señal activa, la salida codifica la de mayor prioridad (generalmente correspondiente al valor decimal más alto). Adicionalmente, se codifican dos salidas más: una indica que ninguna entrada está activa, y la otra, que alguna entrada está activa. Esta medida permite discernir entre los supuestos de que el circuito estuviera deshabilitado por la no activación de la señal de capacitación, que el circuito no tuviera ninguna entrada activa, o que la entrada numero 0 estuviera activada.

Diagrama

Descripción generada automáticamente

**Decodificadores**

(inverso al codificador)

Un decodificador es un circuito combinacional con varias líneas de salida, con una sola de ellas seleccionada en un instante dado, dependiendo del patrón de líneas de entrada. En general, un decodificador tiene n entradas y 2n salidas. Su función es la de direccionar espacios de memoria. Un decodificador de n entradas puede direccionar 2n espacios de memoria. 19 comparadores El comparador compara dos palabras de entrada.

Diagrama, Esquemático

Descripción generada automáticamente

**Sumadores**

Para poder sumar dos números de n bits se usan los sumadores.

**Medio Sumador**

Acepta dos dígitos(o mas??) binarios en las entradas y produce dos dígitos binarios en sus salidas, un bit con la suma y un bit para el carry. El carry de salida (count) es 1 solamente cuando A Y B son 1, por lo tanto, count se expresa como el AND de las variables de entrada. La salida es 1 solo cuando las variables de entrada A y B no son iguales, por lo tanto, la suma se puede expresar como XOR de las variables de entrada.

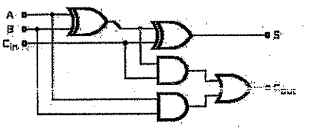
Diagrama, Tabla

Descripción generada automáticamente

Aunque un medio sumador es suficiente para sumar los bits de orden bajo de dos palabras de entrada de varios bits, no sirve para una posición de bit inferior de la palabra porque no maneja el acarreo que llega de la posición que esta a su derecha. Se necesita un sumador completo, el cual se construye de dos medios sumadores.

**Sumador Completo**

Es un circuito combinacional que forma la suma aritmética de tres bits de entrada(pueden ser mas??). Consta de tres entradas y dos salidas. Dos de las variables de entrada, que se indican por Ay B, representan los dos bits significativos que van a añadirse. La tercera entrada C, representa la cuenta que se lleva de la posición previa significativa mas baja. Son necesarias dos salidas debido a que la suma aritmética de tres dígitos binarios varia en valor de 0 a 3. Las dos salidas se denotan por el símbolo S para la suma y C para la cuenta que se lleva. La variable S da el valor del bit menos significativo de la suma.



Hay que notar que, como la salida de cada sumador depende del acarreo del sumador previo, hay un retardo que crece del bit menos significativo al mas significativo. Para sumadores grandes, este retardo puede hacerse inaceptablemente alto.

Circuitos Secuenciales

Un circuito secuencial es un conjunto de puertas lógicas interconectadas entre sí. Se usan para proporcionar memoria o información de estado. La salida actual de un circuito secuencial depende no sólo de la entrada actual, sino también de la historia pasada de las entradas. La salida actual depende de la entrada actual y del estado actual del circuito. Estos valores se almacenan, aunque las entradas no estén.

Se las puede clasificar como asincrónicas (con retardos asociados a las puertas lógicas) o sincrónicas (solamente se permiten cambios de estados en instantes marcados por el reloj). Ejemplos son: Biestables RS, JK, D, Registros y Contadores.

**Diferencias entre Circuito Secuencial y Circuito lógico**:

Circuito combinatorio es el circuito donde si cambias las entradas se produce un cambio en la salida, en ese momento, sin un reloj de por medio.

Circuito secuencial: entra en juego el tiempo, puede ser que para una entrada obtenga una salida, y dentro de un lapso se puede volver a producir la misma entrada con otra salida distinta

Biestables (Flip Flop o Latch)

Un Biestable, también llamado flip flop, es la forma más sencilla de un Circuito Secuencial, con la capacidad de permanecer en uno o dos estados posibles (0 o 1) durante un tiempo indefinido en ausencia de entrada, utilizando el principio de retroalimentación. Posee dos salidas que siempre son complementarias entre si (Q y ~Q) y pueden funcionar como una memoria de 1 bit. Dependiendo del tipo de entradas, se pueden clasificar en:

* **Sincronismo activado por nivel**: el sistema lee las entradas cuando el reloj está en estado ALTO (1 o H) o BAJO (0 o L). También llamado activación por pulso.

Diagrama

Descripción generada automáticamente

* **Sincronismo activado por flanco**: el sistema lee las entradas cuando se produce la transición de sus pulsos, puede ser: ascendente (cuando cambia de 0 a 1) o descendente (cuando cambia de 1 a 0).

Texto

Descripción generada automáticamente con confianza baja

**Tipos de Biestables:**

* Biestable RS Asincrónico: El circuito tiene dos entradas, S (set) y R (reset), y dos salidas, Q y NQ, y consiste en dos puertas NOR conectadas por realimentación. Este tipo de circuitos puede funcionar como una memoria de 1 bit, las entradas S y R sirven para escribir los valores, 1 y 0 en la memoria. El cerrojo S-R es asincrónico porque, si las entradas no cambian, tampoco cambian las salidas

Cuando **R** tiene el valor 1 pone en 0 la salida, y cuando **S** tiene el valor 1 pone en 1 la salida. Cuando las dos entradas tienen valor 0, no se producen cambios en la salida, pero cuando ambos valores valen 1, no sabe cómo actuar y la salida puede cambiar o quedarse inalterada de forma aleatorio.

Diagrama

Descripción generada automáticamente

* Biestable RS Sincrónico: además de las dos entradas R y S, posee una entrada más, denominada CLK, que consiste en un reloj. La diferencia con el S-R es que los cambios sólo ocurren en un pulso de reloj (entrada de 1 bit que va cambiando de 0 a 1 en tiempos regulares, constantemente).

Diagrama

Descripción generada automáticamente

* Biestable JK: fue diseñado para evitar el problema de indeterminación que poseen los Biestables RS cuando ambas entradas tenían el valor de 1. La entrada J tiene la función de SET y la entrada K, funciona como RESET. Cuando ambas entradas valen 0, no se producen cambios y cuando ambas entradas valen 1, se realiza la conmutación: la salida se invierte.

Diagrama, Dibujo de ingeniería

Descripción generada automáticamente

* Biestable D: posee una sola entrada D (data) y la salida Q que copia el valor de la entrada D cuando la señal del reloj se encuentra activada. Tiene la ventaja de que no hay error cuando las entradas valen 1.

Como la salida del biestable D es siempre igual al valor más reciente aplicado a la entrada, recuerda y produce la última entrada. Por lo tanto, sirve para guardar un bit. También se le llama “biestable de retardo” porque retrasa un 0 o un 1 aplicado a la entrada durante un pulso de reloj

Diagrama

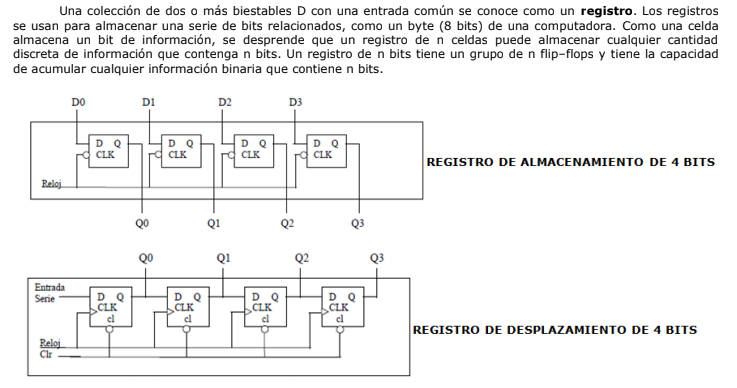
Descripción generada automáticamente

* Biestable T: se construye a partir de un J-K y siempre se niegan las salidas. Las dos entradas siempre valen 1 y lo que genera cambios en la salida es el reloj, por lo tanto, se considera un temporizador. La salida Q cambiara de 0 a 1 y de 1 a 0 en cada pulso de la entrada T.

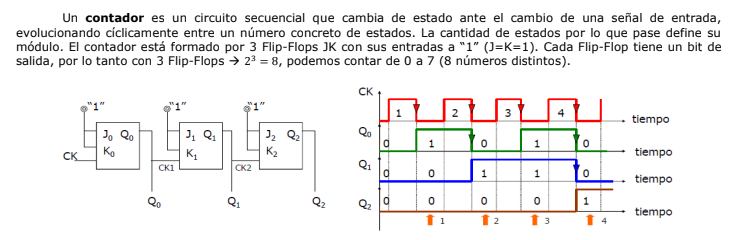
Gráfico

Descripción generada automáticamente

Registros



Contadores



Un circuito hecho con n biestables puede contar hasta 2n-1. Cuando el contador se incrementa más allá de su valor máximo, se pone a 0. Un ejemplo es el contador de programa.

*Los contadores pueden ser síncronos o asíncronos.*

Los contadores **asíncronos** son relativamente lentos, ya que la salida de un biestable produce un cambio en el estado del siguiente biestable.

Diagrama, Esquemático

Descripción generada automáticamente

En un contador **síncrono**, todos los biestables cambian de estado a la vez. Como el último tipo es mucho más rápido, es el tipo que se usa en la CPU.

Diagrama

Descripción generada automáticamente

UNIDAD 04- UNIDAD CENTRAL DE PROCECAMIENTO(CPU)

Estructura – CPU

La CPU es el centro de la computadora. Es el encargado de realizar el control y procesamiento de los datos. Sus funciones principales son:

* Ejecutar las instrucciones de los programas almacenados en la memoria del sistema.
* Controlar la transferencia de datos entre la CPU, los circuitos de memoria y de E/S.
* Responder a las peticiones de servicio procedentes de los dispositivos de E/S.

Para que un programa pueda ser ejecutado por la CPU, debe estar guardado en un determinado lugar de la memoria del sistema y escrito en un lenguaje que la CPU pueda entender.

**Un programa** es una lista de instrucciones almacenadas en memoria que la CPU lee ordenadamente, las interpreta y posteriormente controla su ejecución una tras otra.

**La CPU se compone de varias partes:**

* La Unidad de Control (UC): Controla todas las funciones que realiza una computadora mediante los registros de control y estado. Se encarga de buscar instrucciones de la memoria principal y determinar su tipo.

*Tiene 3 funciones principales*:

1. Dirigir la secuencia de pasos de modo de la CPU lleve a cabo un ciclo completo de ejecución de una instrucción, y hacer esto con todas las instrucciones de que conste el programa.
2. Leer e Interpretar las instrucciones de los programas.
3. Controla el flujo de datos hacia y desde la CPU.

* La Unidad Aritmética Lógica (ALU): Es un circuito combinacional capaz de realizar operaciones aritméticas y lógicas con números enteros y reales. La ALU se compone básicamente de: circuito operacional, registros de entradas, un registro acumulador y un registro de estado, conjunto de registros que hacen posible la realización de cada una de las operaciones. La ALU toma datos de los registros del procesador. Estos datos son procesados y los resultados de esta operación se almacenan en los registros de salida de la ALU. Otros mecanismos mueven datos entre estos registros y la memoria.
* Registros Internos de la CPU: La CPU también consta de un conjunto de posiciones de almacenamiento, llamadas registros. Por lo general, todos los registros tienen el mismo tamaño. Pueden leerse y escribirse a alta velocidad porque están dentro de la CPU.
* Interconexión Interna de la CPU: Todas las partes de la CPU (procesador, memoria, E/S) están unidas mediante diversas líneas eléctricas. El conjunto de estas líneas se denominan **bus interno de la CPU**.

Bus :

Es un camino de comunicación que conecta los componentes principales de la computadora (procesador, memoria, e/s). Se pueden diferenciar 3 buses:

1. **Bus de datos:** transporta datos. No hay diferencia entre dato e instrucción en este nivel. El ancho es un valor determinante en las prestaciones.
2. **Bus de direcciones:** indica el origen o destino de los datos. El ancho del bus determina la máxima capacidad de memoria del sistema.
3. **Bus de control:** se utiliza para controlar el acceso y el uso de las líneas de datos y de direcciones, ya que son compartidas. Informacion de control y temporizado.

**Direcciones**

Si el bus es compartido por diferentes elementos, estos deben tener identificadores distintivos: direcciones. La dirección de memoria identifica una celda de memoria en la que almacena información.

Función de la Computadora

Ejecutar las instrucciones de los programas almacenados en la memoria del sistema. Podemos descomponer el procesamiento de instrucciones en dos etapas: Búsqueda (leer desde memoria) y Ejecución (depende de la instrucción puede aplicar varias operaciones).

La ejecución completa de cada instrucción lleva varios pasos (llamados ciclo de instrucción) y para realizarlos la CPU se vale de dos unidades: La Unidad de Control (UC) y la Unidad Aritmética - Lógica (ALU).

**En general las acciones de búsqueda y ejecución caen en 4 tipos:**

MEMORIA (datos que pueden transferirse entre memoria y CPU), E/S (datos que pueden transferirse entre CPU y entrada/salida), PROCESAMIENTO DE DATOA (CPU efectúa operaciones aritméticas o lógicas en datos), CONTROL (alterar la secuencia de ejecución de instrucciones).

Organización de los Registros Internos de la CPU

Los registros son, básicamente, elementos de la memoria de acceso rápido que se encuentran dentro del procesador. Constituyen un espacio de trabajo para el procesador y se utilizan como un espacio de almacenamiento temporal. Son imprescindibles para ejecutar las instrucciones , entre otros motivos, porque la ALU solo trabaja con los registros internos de la CPU

**Clasificación de los registros del procesador: (2 en total)**

Registros de Propósito General

Todas las CPU contienen registros internos de propósito general que pueden ser referenciados por el programador, como fuente o destino (o ambos) en una instrucción. Como si fuesen “memoria” pero mucho más rápido. Son lugares de almacenamiento temporario.

*Se clasifican de la siguiente manera:*

* **Uso general:** pueden ser asignados por el programador a diversas funciones.
* **Datos:** se usan únicamente para contener datos y no se pueden emplear en el cálculo de una dirección de operando.
* **Direcciones:** pueden ser de uso más o menos general, o pueden estar dedicados a un modo de direccionamiento particular.
* **Códigos de condición (flags):**son bits fijados por el hardware de la CPU como resultado de una operación.

Registros de Control y Estado

Son utilizados por la Unidad de Control para controlar la operación de la CPU. La mayoría no son visibles para el usuario.

*Son esenciales cuatros registros para la ejecución de una instrucción:*

* **PC (búsqueda):** Cuando un programa va a hacer ejecutado, el PC contiene la dirección de la primera instrucción. Alcanzada la primera instrucción, el PC es incrementado para apuntar a la siguiente instrucción.
* **IR (ejecución):** La instrucción captada recientemente se almacena temporalmente en el registro. La instrucción está en la forma de un código binario que especifica las acciones que tomara la CPU. La CPU interpreta cada instrucción y lleva a cabo las acciones requeridas.
* **MAR:** Registro de dirección de memoria a la que queremos acceder.
* **MBR:** Registro que contiene la palabra de datos a escribir en memoria, o la palabra leída más recientemente

Pasos del ciclo de instrucciones (SACADO DE TEORIA)

* En el primer paso, la CPU busca la instrucción en memoria. Para esto copia el valor del PC al MAR y de ahí al bus de direcciones. La Unidad de Control envía las señales necesarias para una operación de lectura. Se pueden leer uno o más bytes. A través del bus de datos al MBR y luego al IR.
* Después de buscar la instrucción, la CPU debe incrementar el PC para apuntar a “lo que sigue”. Puede ser un dato, dirección o la siguiente instrucción.
* El paso siguiente es decodificar la instrucción para saber qué operación hacer. La CPU no solo se entera de la operación, sino también donde se encuentran los datos sobre los cuales operar.
* La CPU determina si tiene que ir a buscar el operando a memoria, que ocupa una celda o más, y si existió este paso la CPU debe incrementar el PC en el valor adecuado de celdas (1, 2 o más).

Pasos del ciclo de instrucciones

1. **Calcular la dirección de la instrucción**: determina la dirección de la siguiente instrucción a ejecutar
2. **Captar la instrucción:** la CPU lee la instrucción desde su posición en memoria.
3. **Decodificar la operación indicada en la instrucción**: analiza la instrucción para determinar el tipo de operación a realizar y el/los operandos a utilizar.
4. **Cálculo de la dirección del operando:** si la instrucción implica una referencia a un operando en memoria o disponible mediante E/S, determina la dirección del operando.
5. **Captar el operando**: capta el operando desde memoria o se lee desde el dispositivo de E/S.
6. **Operación con los datos:** realiza la operación indicada en la instrucción
7. **Calculo dirección resultado**
8. **Almacenamiento del operando**: escribe el resultado en memoria o lo saca a través de un dispositivo de E/S.

Diagrama

Descripción generada automáticamente

Elementos de una instrucción de maquina

* **Código de operación:** especifica la operación a realizar. Es un código binario.
* **Referencia del operando fuente:** establece donde se encuentra el operando. La operación puede involucrar uno o más operando fuente (o de entrada).
* **Referencia del operando resultado:** establece donde almacenar el resultado.
* **Referencia de la siguiente instrucción:** le dice a la CPU donde buscar la siguiente instrucción después de la ejecución de la instrucción anterior.
* Los operandos fuente y resultado pueden estar en tres lugares: Memoria. Registro de la CPU. Dispositivo de E/S.

Representación de Instrucciones (formato de instrucción)

Dentro de la computadora cada instrucción está representada mediante una secuencia de bits. La secuencia se divide en campos en correspondencia a los elementos que la componen. Este esquema se conoce como **formato de la instrucción.** Es difícil para el programador tratar con las representaciones binarias de las instrucciones de máquina. Por lo tanto, se usa una representación simbólica. Los códigos de operación se representan por medio de abreviaturas, llamadas mnemónicos que indican la operación. Los más comunes son: ADD, SUB, MOV, AND, OR, XOR.

Características de una instrucción de maquina

El funcionamiento de la CPU está determinado por las instrucciones que ejecuta. Estas instrucciones se denominan instrucciones máquina.

**Elementos de una instrucción máquina**

* **Código de operación:** especifica la operación a realizar. La operación se indica mediante un código binario, denominado “codop”.
* **Referencia a operandos fuente:** establece donde se encuentra el operando. La operación puede involucrar uno o mas operando fuente, es decir, operandos que son de entrada para la instrucción.
* **Referencia al operando resultado:** establece donde almacenar el resultado.
* **Referencia a la siguiente instrucción:** le dice a la CPU de donde captar la siguiente instrucción tras completarse la ejecución de la instrucción actual.

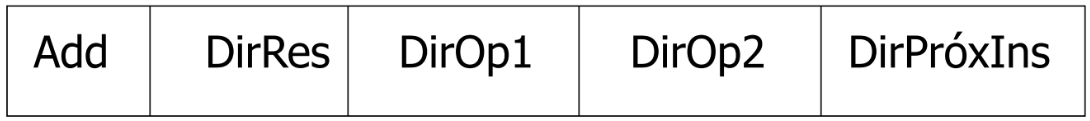
La siguiente instrucción por captar está en memoria principal o secundaria.

*Los operandos fuente y resultado pueden estar en alguna de las siguientes áreas:*

* **Memoria principal o virtual:** debe indicarse su dirección.
* **Registro de la CPU:** si es un solo registro, la referencia es implícita, si es más de uno, debe indicarse el número de registro deseado en la instrucción.
* **Dispositivo de E/S**: la instrucción debe especificar el módulo y el dispositivo para la operación.

**Numero de direcciones**

* ¿Cuántas direcciones se necesitan?
* Dos direcciones para hacer referencia a los operandos, una donde almacenar el resultado y la dirección de la próxima instrucción. Por lo tanto necesitaríamos cuatro direcciones



* Máquina de 4 direcciones: Direcciones explicitas para operandos, resultado y próxima instrucción. Son “raras”, cada campo de dirección tiene que tener bits para “acomodar” una dirección completa. Ej. si dirección = 24 bits, la instrucción tiene 96 bits de referencias.

Diagrama

Descripción generada automáticamente

* Maquina de 3 direcciones: Dirección de la próxima instrucción está almacenada en un registro de la CPU, llamado Contador de Programa PC. Referencias = 72 bits. Todavía larga.

Diagrama

Descripción generada automáticamente

* Maquina de 2 direcciones: Hay que mover el Op1 a un registro temporal. Menos elección donde guardar el resultado. Reduce el tamaño de la instrucción: 48bits de referencia (2\*24).

Diagrama

Descripción generada automáticamente

* Maquina de 1 dirección: Aparecen los registros especiales de la CPU (Acumulador). Instrucciones para cargar y descargar el acumulador. Un operando y resultado en lugar predefinido. Instrucción más corta (24bits de referencia)

Diagrama

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Tipos de Instrucciones

El repertorio de instrucciones debe ser suficientemente amplio como para expresar cualquiera de las instrucciones de alto nivel.

*Los tipos de instrucciones se clasifican de la siguiente manera:*

* **De Procesamiento de datos**: instrucciones aritméticas y lógicas. Las instrucciones aritméticas proporcionan capacidad computacional para procesar datos numéricos. Las instrucciones lógicas operan sobre los bits de una palabra, el lugar de considerarlos como números, proporcionando capacidad para el procesamiento de cualquier otro tipo de datos que el usuario deba emplear
* **De Almacenamiento de datos:** instrucciones de memoria. •
* **De transferencia de datos**: instrucciones de E/S. Las instrucciones de E/S se necesitan para transferir programas y datos a la memoria, y devolver los resultados de los cálculos al usuario.
* **De control**: instrucciones de comprobación y bifurcación. Las instrucciones de comprobación o test se emplean para comprobar el valor de una palabra de datos o el estado de un cálculo. Las de bifurcación se usan entonces para bifurcar a diferentes conjuntos de instrucciones dependiendo de la decisión tomada.

**Tamaño de las instrucciones**

* **Instrucciones de tamaño fijo**: todas las instrucciones ocuparan el mismo número de bits. Esta alternativa simplifica el diseño del procesador y la ejecución de las instrucciones puede ser más rápida.
* **Instrucciones de tamaño variable**: el tamaño de las instrucciones dependerá del número de bits necesario para cada una.

Diseño del Conjunto/Repertorio de Instrucción

El repertorio es el medio que tiene el programador para controlar la CPU, por lo tanto, deben considerarse las necesidades de éste a la hora de diseñarlos.

*Los aspectos más importantes de diseño son:*

* **Tipos de operaciones**: cuántas y qué operaciones considerar y de qué complejidad.
* **Tipos de datos:** los distintos tipos de datos con los que se efectúan las operaciones.
* **Formatos de instrucciones**: longitud de la instrucción, número de direcciones, etc.
* **Registros:** número de registros de la CPU que pueden ser referenciados y uso.
* **Direccionamiento**: el modo mediante el que puede especificarse las direcciones de un operando.

**Tipos de operaciones**

* Transferencia de datos (Mov(load/store)).
* Aritméticas (add, sub, inc, dec, mul).
* Lógicas (and, or, etc.).
* Conversión.
* Entrada-Salida (in-out).
* Control del sistema.
* Control de flujo (salto, bifurcación).

**Tipos de datos**

Las instrucciones máquina operan con datos. Las categorías más importantes de datos son:

* Direcciones: ver direccionamiento. Son enteros sin signo.
* Números: enteros o en coma fija, en coma flotante y en decimal.
* Caracteres: ASCII, EBCDIC.
* Datos lógicos: representación orientada a bits.

Direccionamiento

* Son las diferentes maneras de expresar un operando en una instrucción.
* Un modo de direccionamiento especifica la forma de calcular la dirección de memoria efectiva de un operando mediante el uso de la información contenida en registros y/o constantes, contenida dentro de una instrucción de la máquina o en otra parte (Wikipedia).

Como vimos, en una instrucción se utilizan bits para expresar el **código de operación**: nos dice **qué** hacer. También se necesitan una “gran” cantidad de bits para especificar de dónde provienen los datos**. (sacado de teoría)**

**¿Cómo podemos reducir el tamaño de estas especificaciones?**

**Hay 2 métodos generales:**

* Si un operando va a usarse varias veces puede colocarse en un registro. Usar registro para una variable tiene 2 ventajas: el acceso es más rápido // se necesitan menos bits.

Ej. si hay 32 reg. se necesitan 5 bits para especificar c/u de ellos (menos bits que las dir. de mem.).

* Especificar uno ó más operandos en forma implícita.

Ej: reg2 = reg2 + fuente1 ; el acumulador.

**Los modos de direccionamiento tienen como objetivo:**

* Disminuir la cantidad de bits en la instrucción.
* La dirección puede que no se conozcan hasta el momento de ejecutar el programa.
* Manejo más eficiente de datos.

*Los modos de direccionamiento son:*

* **INMEDIATO:** El operando se obtiene automáticamente de la memoria al mismo tiempo que la instrucción. Ventaja: No requiere una referencia extra a memoria de datos. Se utiliza para definir constantes y para inicializar variables. Desventaja: tamaño del operando limitado por el tamaño del campo de direccionamiento.



* **DIRECTO**: El campo de dirección tiene la dirección efectiva del operando. Es simple, pero tiene un espacio limitado de direcciones por cantidad de bits del campo. Uso: acceder a variables globales, cuya dirección se conoce en el momento de compilación.

Forma

Descripción generada automáticamente con confianza media

* **POR REGISTRO**: Conceptualmente igual al directo, pero se especifica un registro en lugar de una posición de memoria. Ventaja: la referencia a registro usa menos bits que la especificación de la dirección y no requiere acceso a memoria de datos. Desventaja: los registros no son muchos y es un recurso preciado.

Diagrama

Descripción generada automáticamente

* **INDIRECTO POR MEMORIA**: En la instrucción está la dirección de la dirección del operando. Trata de solucionar el problema del directo. Así, con una dirección de menos bits en la instrucción, se apunta a una dirección de más bits. Ventaja: espacio de direccionamiento mayor. Desventaja: múltiples accesos a memoria.

Diagrama

Descripción generada automáticamente

* **INDIRECTO POR REGISTRO:** En la instrucción se especifica el registro que tiene almacenada la dirección. Ventaja: menos bits para especificar el registro que la posición de memoria. Espacio de direccionamiento grande, accede una vez menos a memoria que el indirecto. La dirección así usada se llama apuntador.

Diagrama

Descripción generada automáticamente

* **POR DESPLAZAMIENTO**: Combina capacidades de indirecto y directo. Requiere que la instrucción tenga dos campos de dirección. Estos dos campos se suman para producir la dirección efectiva.

Los más comunes:

* **Relativo:** El registro referenciado de manera implícita es el contador de programa PC. La dirección de la instrucción actual se suma al campo de dirección para producir la dirección efectiva. El campo de dirección se trata como un número en Ca2. Este tipo de desplazamiento aprovecha el concepto de localidad.
* **De registro base**: El registro referenciado contiene una dirección de memoria y el campo de dirección tiene un desplazamiento desde dicha dirección. . La referencia a registro puede ser implícita o explicita. Este direccionamiento también aprovecha la localidad de las referencias a memoria.
* **Indexado:** Se direcciona la memoria con un registro más un desplazamiento. Es “igual” al anterior, pero se intercambian los papeles del registro y del desplazamiento. La Indexación proporciona un mecanismo eficiente para realizar operaciones iterativas. Se utiliza un registro llamado índice algunas máquinas incrementan o decrementan este registro como parte de la instrucción (Auto indexación)

Diagrama

Descripción generada automáticamente

* **POR STACK**: El stack ó pila es un arreglo lineal de localidades de memoria. Es una lista ó cola donde el último en entrar es el primero en salir. Es una zona de memoria reservada. Asociado con la pila o stack hay un registro apuntador (o registro puntero de pila), cuyo valor es la dirección tope de pila o stack.

**Lenguaje de Ensamble**

Un lenguaje ensamblador puro es una representación simbólica de un programa en lenguaje de maquina subyacente, en el que cada enunciado produce exactamente una instrucción de máquina, es decir, existe una relación uno a uno entre las instrucciones de máquina y los enunciados del programa en leguaje assembly.

Ventajas del Leguaje ensamblador

* Puede producir código mucho más pequeño y rápido que el se puede producir en un lenguaje de alto nivel.
* Buen desempeño.
* Acceso total a la maquina/hardware.
* Útil para generar los programas de computadores portátiles de escasos recursos, como las tarjetas inteligentes, procesadores incorporados a aparatos domésticos y asistentes digitales portátiles inalámbricos.

Interrupciones

Las interrupciones proporcionan una forma de mejorar la eficiencia del procesador. Existen diferentes clases de interrupciones:

* **Programa**: generadas por alguna condición que se produce como resultado de la ejecución de una instrucción, como la división por cero.
* **Temporización:** generadas por un temporizador interno al procesador. Permite al sistema operativo realizar ciertas funciones de manera regular.
* **E/S:** generadas por un controlador de E/S para indicar la finalización sin problemas de una operación o para avisar de ciertas condiciones de error.
* **Fallo de hardware**: generadas por un fallo tal como un error de paridad en la memoria.

Las interrupciones y el ciclo de instrucción

Para permitir el uso de interrupciones, se añade un ciclo de interrupción al ciclo de instrucción. En el ciclo de interrupción, el procesador comprueba si ha generado una interrupción, indicada por la presencia de una señal de interrupción. Si no hay señales pendientes, el procesador continúa con el ciclo de captación y accede a la siguiente instrucción del programa en curso. Si hay una interrupción pendiente, el procesador hace lo siguiente:

1. **Suspende la ejecución** del programa en curso y **guarda su contexto.**

2. **Carga el contador de programa** con la dirección de **comienzo de una rutina de gestión de interrupción.**

Estructuras de Interconexión

Son el conjunto de líneas que conectan los módulos de un computador. Los tipos de intercambio son:

* **Memoria:** un módulo de memoria está constituido por N palabras de la misma longitud. A cada palabra se le asigna una única dirección numérica (de 0 a N-1). Una palabra de datos puede leerse o escribirse en la memoria. La posición de memoria para la operación se especifica mediante una dirección.
* **Módulo de E/S:** la E/S es funcionalmente similar a la memoria. Hay dos tipos de operaciones: leer y escribir. Existen líneas de datos externas para la entrada y la salida de datos por un dispositivo externo. Por último, un módulo de E/S puede enviar señales de interrupción al procesador.
* **Procesador:** lee instrucciones y datos, escribe datos una vez que los ha procesado, y usa ciertas señales para controlar el funcionamiento del sistema.

**La estructura de interconexión debe dar cobertura a los siguientes tipos de transferencia:**

1. **Memoria a Procesador:** el procesador lee una instrucción desde memoria.
2. **Procesador a Memoria**: el procesador escribe un dato en la memoria.
3. **E/S a Procesador:** el procesador envía datos al dispositivo de E/S.
4. **Memoria a E/S y viceversa**: el módulo de E/S intercambia datos en memoria, utilizando DMA.

UNIDAD 5 – MEMORIA

**Características de los sistemas de memoria**

**Las características más importantes de los sistemas de memoria son:**

**Ubicación**

Puede ser interna o externa. La memoria interna suele identificarse como la memoria principal. Sin embargo, el procesador necesita su propia memoria local en forma de registros y la CU del procesador también usa su propia memoria interna, La memoria externa consta de dispositivos periféricos de almacenamiento, tales como discos y cintas.

**Capacidad**

Para memorias internas se expresa en bytes o palabras (8, 16 ó 32 bits). Para memorias externas se expresa en bytes.

**Unidad de transferencia**

Es el número de bits que se leen o escriben en memoria a la vez. Para memorias internas es igual al número de líneas de E/S de datos del módulo de memoria (una palabra o una unidad direccionable, generalmente). Para memorias externas, los datos se transfieren en bloques.

**Método de acceso**

* **Acceso secuencial:** Cintas. El acceso debe hacerse en una secuencia lineal especifica.
* **Acceso directo:** Disco. Se accede a un bloque y luego se hace un acceso secuencial hasta llegar a la posición de la memoria deseada.
* **Acceso aleatorio**: Memoria principal y algunas caché
* **Asociativa**: Memorias caché. Tipo de memoria de acceso aleatorio donde el acceso se hace basándose Enel contenido y no la dirección.

**Prestaciones**

Se utilizan tres parámetros de medida de prestaciones:

 **Tiempo de acceso**: para memorias de acceso aleatorio es el tiempo que tarda en realizarse una operación de lectura o escritura. Para otras memorias, es el tiempo de acceso en que se tarda en situar el mecanismo de lectura/escritura en la posición deseada.

 **Tiempo de ciclo de memoria:** se aplica a las memorias de acceso aleatorio y consiste en el tiempo de acceso y algún tiempo más que se requiere, antes. de que pueda iniciarse un segundo acceso a memoria.

 **Velocidad de transferencia**: es la velocidad a la que pueden transferirse los datos a, o desde, una unidad de memoria.

**Dispositivos físicos**

 Las más comunes son las memorias semiconductoras, las memorias de superficie magnética, utilizadas para discos y cintas, y las memorias ópticas y magneto-ópticas.

**Características físicas**

Volátiles o no volátiles.

**Organización**

Es la disposición o estructura física en bits para formar palabras.

### **Jerarquía de memoria**

Sería deseable poder utilizar sólo la memoria más rápida, pero al ser la más costosa, se llega a un compromiso entre el tiempo de acceso y coste, empleando más cantidad de memoria más lenta; La estrategia a seguir consiste en organizar los datos y los programas en memoria de manera que las palabras de memoria necesarias estén normalmente en la memoria más rápida. 

En todo el espectro de posibles tecnologías se cumplen las siguientes relaciones:

1. A menor tiempo de acceso, mayor coste por bit.
2. A mayor capacidad, menor coste por bits. 
3. A mayor capacidad, mayor tiempo de acceso.

Diagrama

Descripción generada automáticamente

**Cuando se desciende en la jerarquía ocurre:**

1. **Disminuye el coste por bit**
2. **Aumenta la capacidad**
3. **Aumenta el tiempo de acceso**
4. **Disminuye la frecuencia de accesos a la memoria por parte del procesador.**

La clave del éxito de esta organización está en el último ítem: la disminución de la frecuencia de acceso.

La base para la validez de la condición 4 es el principio conocido como localidad de las referencias

Se denomina **cercanía de referencias** al agrupamiento de las lecturas de memoria por medio de la CPU. Las mismas, ya sean para instrucciones o para leer datos, se mantienen por lo general dentro de grupos de direcciones relativamente cercanas entre sí. Esto se da porque los programas normalmente cuentan con un cierto número de bucles y subrutinas iterativas. Una vez dentro de una de estas estructuras, se producirán referencias repetidas a un pequeño conjunto de instrucciones. Las agrupaciones de uso con el tiempo son variables, pero considerando un período corto de tiempo se mantienen

**Cercanía de referencias** también implica que para que un programa ejecute no son necesarias todas sus páginas cargadas en la memoria principal. Si en un período corto de tiempo el procesador referencia direcciones cercanas de memoria, un proceso puede ejecutar varias instrucciones con sólo algunas partes de su código en memoria principal. Esto permite tener más procesos en memoria listos para correr, como así también evita tener que cargar y descargar sus páginas en el momento de un intercambio, perdiendo ciclos del procesador.

### **Localidad de referencias**

La localidad de las referencias, también conocida como el **principio de localidad**, es un fenómeno según el cual, basándonos en el pasado reciente de un programa podemos predecir con una precisión razonable qué instrucciones y datos utilizará en un futuro próximo.

Los casos más importantes de localidad son la **localidad espacial, la localidad secuencial y la localidad temporal.**

#### **Localidad Temporal:** Si en un momento una posición de memoria particular es referenciada, entonces es muy probable que la misma ubicación vuelva a ser referenciada en un futuro cercano. Existe proximidad temporal entre las referencias adyacentes a la misma posición de memoria. En este caso es. común almacenar una copia de los datos referenciados en caché para lograr un acceso más rápido a ellos.

1. **Localidad Espacial****:** Si una localización de memoria es referenciada en un momento concreto, es probable que las localizaciones cercanas a ella sean también referenciadas pronto. Existe localidad espacial entre las posiciones de memoria que son referenciadas en momentos cercanos. En este caso es común estimar las posiciones cercanas para que estas tengan un acceso más rápido. (acceso a una palabra en memoria)
2. **Localidad Secuencial:** Las direcciones de memoria que se están utilizando suelen ser contiguas. Esto ocurre porque las instrucciones se ejecutan secuencialmente. Para obtener beneficios de la gran frecuencia con la que ocurren casos de localidad espacial o temporal, muchos sistemas de memoria utilizan una jerarquía de niveles de memoria.

### **Memoria principal semiconductora**

#### Tipos de memorias semiconductoras de **acceso aleatorio**

**RAM**

Es la más común. Una característica distintiva es que es posible, tanto leer datos, como escribir rápidamente nuevos datos en ella. A su vez, es volátil. Las tecnologías de RAM se dividen en:

* *Dinámica(¿DRAM?):* está hecha con celdas que almacenan los datos como cargas en condensadores. Ya que los condensadores tienen una tendencia natural a descargarse, las RAM dinámicas requieren refrescos periódicos para mantener memorizados los datos. Usada como memoria principal.
* *Estática(¿SCRAM?):* los valores binarios se almacenan utilizando configuraciones de puertas que forman biestables. Una RAM estática retendrá. sus datos en tanto se mantengan alimentada. Usada como memoria cache.

Las RAM estáticas son más rápidas y menos densas que las dinámicas.

**ROM:** contiene un patrón permanente de datos que no puede alterarse. Sus aplicaciones son: la microprogramación, subrutinas de biblioteca para funciones de uso frecuente, programas del sistema y tablas de funciones.

**PROM:** son no volátiles y pueden grabarse una sola vez. Proporcionan flexibilidad y comodidad. 

**Memorias de sobre-todo-lectura**: hay tres formas:

* **EPROM:** pueden modificarse múltiples veces y retienen su contenido indefinidamente. Una EPROM es más costosa que una PROM, pero se puede actualizar muchas veces.
* **EEPROM:** se puede escribir en cualquier momento sin borrar su contenido anterior. Son más costosas que las EPROM y menos densas (menos bits por chip).
* **Memoria flash:** puede borrarse entera en unos cuantos segundos. Además, es posible borrar sólo bloques concretos de memoria, aunque no a nivel de byte.

**Organización de la memoria**

Elemento básico es la celda de memoria, que presenta dos estados estables (que representa el 0 y 1), pueden ser escritas para fijar su contenido, pueden ser leídas para detectar su estado, posee tres terminales para transportar señales eléctricas:

1. **de selección:** selecciona la celda para la operación de escritura/lectura.
2. **de control:** indica el tipo de operación.
3. **de lectura/escritura**: Para la escritura, el tercer terminal proporciona la **señal** que fija el estado de la celda a 1 o a 0. En una lectura, el tercer terminal se usa como **salida** del estado de la celda.

**Direccionamiento 2D (Memoria Estática)**

Se puede pensar en el direccionamiento 2D como en una lista de celdas apiladas, las cuales tienen un tamaño N (donde N puede ser el tamaño de palabra). Un tipo de memoria que maneje este tipo de direccionamiento debe tener una conexión con alguna interfaz que le permita conseguir los datos a ingresar, y otra que le indique a que celda tiene que escribir.

También debe tener una conexión con el lugar a donde tiene que guardar los datos, si es que se realiza una lectura, aunque no siempre es necesario, porque se puede pensar en utilizar la misma conexión para escribir o leer, indicándole a la memoria con un flag qué operación se debe hacer. Esto le agrega complejidad al programador para manejar esta memoria, pero. en contrapartida se gana un diseño un poco más simple. Hay que recalcar que cuando se accede, ya sea leyendo o escribiendo, se accede a una celda completa.

La problemática que tiene este tipo de disposición de memoria es que es físicamente muy difícil de implementar con una cantidad de registros grandes, generalmente estas memorias oscilan en tamaño entre los 512Kb-1Mb. Esto sedebe a que, si pensamos en este tipo de organización como una lista de celdas, esta estructura es muy frágil (pensemos, por ejemplo, si queremos implementar una memoria de 15Mb con palabras de 32bits con celdas que almacenan palabras completas, entonces, tendríamos 3.932.160 celdas de memorias acomodadas una tras de otra; con lo que esta disposición requeriría mucho espacio a Io largo del soporte de la memoria). Hay que mencionar que, si bien puede haber muchas celdas, el tiempo de acceso a ellas con esta disposición es casi constante.

Generalmente, este tipo de direccionamiento de memoria se aplica en las Memorias Estáticas (Registros de CPU, o Caché), más que nada porque se implementa con Flip-Flops que, a diferencia de los transistores, no necesitan refresco y son más rápidos a la hora de retribuir la información que almacenan.

**Direccionamiento 2D1/2 (Memoria Dinámica)**

Dadas las dificultades a las que nos enfrentamos en el modo de direccionamiento 2D, y teniendo en cuenta el coste (económico, y técnico) que tiene su implementación, se ha optado por otro tipo de modo de direccionamiento que satisfagan requerimientos diferentes a los previamente satisfechos.

De esta manera, surge el modo de direccionamiento 2D1/2, el cual se organiza de la siguiente manera:

Una memoria que utilice este tipo de direccionamiento tendría una estructura **compuesta por varios chips**, las cuales tendrían la responsabilidad de retomar el valor de un bit, el cual se utilizaría para formar el valor de la palabra de N bits que se quiera leer, o escribir en la memoria. Para ello, sería necesario proveerle al chip, la posición de ella que quiero acceder. En consecuencia, **cuando a la memoria se le pide acceder a una dirección, la misma se divide en dos partes** (a través de 2 decodificadores), uno que le indica a la memoria, qué "renglón" se quiere leer (con la cual se ubica a todas los chips necesarios para la reconstrucción de ese "renglón") y el otro que indica en qué parte de ese chip se debe buscar el bit correspondiente a una fracción del "renglón" a devolver. Una vez identificado cada bit de la palabra, se reconstruye la misma, y se la ubica en la interfaz de salida de la estructura,

**El direccionamiento 2D1/2 mantiene una gran ventaja con respecto al direccionamiento 2D**, y esta es que, a pesar de que obtener una palabra de esta memoria requiere más procesamiento, puede extender su capacidad inmensamente con respecto al direccionamiento 2D. De hecho, las memorias RAM que utilizamos en nuestras computadoras tienen capacidades que alcanzan los Gb (más allá del hecho de que cada una de estas plaquetas está compuesta de varios chips, esto habla del tamaño de la estructura que se puede armar con este tipo de direccionamiento).

Ahora bien, a consecuencia de este gran incremento en la capacidad de almacenamiento de una memoria con este modo de direccionamiento, y debido al hecho de que, por una cuestión de costes, éstas se construyen con transistores, estas memorias, si bien son mucho más baratas que las 2D, también son considerablemente más lentas. Sobre todo, teniendo en cuenta el procesamiento extra que conlleva decodificar una dirección de memoria en particular.

Generalmente, este tipo de direccionamiento de memoria se aplica en las Memorias Dinámicas (DRAM), que se implementan con transistores, los cuales necesitan refrescos periódicos de la información que almacenan.

**Direccionamiento 3D (Modelo Teórico) – No esta en teoría, pero preguntar igual si lo tienen en cuenta**

Este es un modelo teórico, debido a que es, en realidad, un modelo que tiene implementado toda la estructura para hacer funcionar el modelo 2D1/2. Esto es planteado de forma de que, cada uno de los chips de las que se obtiene un bit, reflejaría una capa del modelo de direccionamiento 3D. Entonces obtendríamos el contenido de una dirección, si obtuviésemos el contenido de cada capa en la misma coordenada X e Y, siendo Z, el plano que representa a todas las capas en la misma coordenadas.

***Memoria caché***

El objetivo de la memoria cache es lograr que la velocidad de la memoria sea lo más rápida posible, consiguiendo al mismo tiempo un tamaño grande al precio de memorias semiconductoras menos costosas.

Hay una memoria principal relativamente grande y más lenta, junto con una memoria cache más pequeña y rápida. La cache contiene una copia de partes de la memoria principal. Cuando el procesador intenta leer una palabra de memoria, se hace una comprobación para determinar si la palabra está en la caché. Si es así, se entrega dicha palabra al procesador. Si no, hay un bloque de memoria principal, consistente en cierto número de palabras, se transfiere a la Cache y, después, la palabra es entregada al procesador. Debido al fenómeno de localidad de las referencias, cuando un bloque de datos es captado por la cache para satisfacer una referencia a memoria simple, es probable que se hagan referencias futuras a otras palabras del mismo bloque.

#### **Memoria externa**

Es la memoria secundaria, es el conjunto de dispositivos y soportes de almacenamiento de datos que conforman el subsistema de [memoria](https://es.wikipedia.org/wiki/Memoria_(inform%C3%A1tica)) de la [computadora](https://es.wikipedia.org/wiki/Computadora), junto con la [memoria principal](https://es.wikipedia.org/wiki/Memoria_principal): discos magnéticos, discos óptimos, cintas magnéticas.

¿Dentro de los discos óptimos esta LA CINTA MAGNETICA? **preguntar**

**Discos magnéticos**

Un disco es un plato circular construido con metal o plástico, cubierto por un material magnetizable. Los datos se graban en él y después se recuperan del disco a través de una **bobina, llamada cabeza.** Durante una operación de lectura o escritura, la cabeza permanece quieta mientras el plato rota bajo ella.

El mecanismo de escritura se basa en el campo magnético producido por el flujo eléctrico, que atraviesa la bobina. Se envían pulsos a la cabeza, y se graban patrones magnéticos en la superficie bajo ella, con patrones diferentes para corrientes positivas y negativas. El mecanismo de lectura se basa en la corriente eléctrica que atraviesa la bobina, producida por un campo magnético que se mueve respecto a la bobina. Cuando la superficie del disco pasa bajo la cabeza, se genera una corriente de la misma polaridad que la que produjo la grabación magnética.

Organización y formato de los datos

Los datos se organizan en un conjunto de anillos concéntricos, llamados **pistas(divididas en sectores)**. Cada pista es del mismo ancho que la cabeza. Las pistas están separadas por bandas vacías, lo que previene errores debidos a deslineamientos de la cabeza o simplemente interferencias del campo magnético.

Los datos se transfieren hacia y desde el disco en **bloques(más de un sector).** El bloque es menor que la capacidad de una pista. De acuerdo con esto, los datos se almacenan en regiones del tamaño de un bloque, denominadas sectores.

Características físicas

Las principales características son:

 **Desplazamiento de cabezas:** las cabezas pueden ser:

* Fijas: hay una cabeza de lectura/escritura por pista. Todas las cabezas se montan en un brazo rígido que se extiende a través de todas las pistas.
* Móvil: hay sólo una cabeza de lectura/escritura que se monta en un brazo. Como la cabeza debe poder posicionarse encima de cualquier pista, el brazo debe extenderse o retraerse para este propósito. 

 **Transportabilidad del disco**: se divide en dos categorías:

* No extraíble: está permanentemente montado en la unidad de disco.
* Extraíble: puede ser quitado y sustituido por otro disco.

 **Superficies:** puede ser de una sola superficie o doble. 

 **Platos:** puede tener un plato único o varios platos apilados verticalmente. En conjunto, se llama paquete de disco y disponen de varios brazos.

**Mecanismo de la cabeza**: se clasifica en tres tipos: Separación fija: la cabeza se posiciona a una distancia fija sobre el plato, dejando entre ambos una capa de aire. Contacto: la cabeza efectúa un contacto físico con el medio durante la operación de lectura o escritura. Se usa con los disquetes. Separación aerodinámica (discos Winchester): las cabezas están montadas en unidades herméticamente cerradas, casi libres de contaminación. Estos discos operan más cerca de la superficie, por tanto, permiten una densidad de datos mayor. La cabeza está en el contorno de una hoja de metal aerodinámica, que reposa suavemente sobre la superficie del plato cuando el disco no se mueve. La presión de aire generada por el giro del disco es suficiente para hacer subir la hoja encima de la superficie.



##### Parámetros para medir las prestaciones de un disco

Cuando la unidad de disco está funcionando, el disco está rotando a una velocidad constante. Para leer o escribir, la cabeza debe posicionarse en la pista deseada y al principio del sector deseado de la pista. La selección de la pista implica un movimiento de la cabeza. El tiempo que tarda la cabeza en posicionarse en la pista se conoce como tiempo de búsqueda (tiempo inicial de comienzo y tiempo necesario para atravesar las pistas que tienen que cruzarse una vez que el brazo de acceso esté a la velocidad adecuada). El tiempo que tarda el sector en alcanzar la cabeza se llama retardo rotacional, o latencia rotacional. La suma del tiempo de búsqueda, si lo hay, y el retardo rotacional se denomina tiempo de acceso, o tiempo que se tarda en llegar a la posición de lectura o escritura. Una vez posicionada la cabeza, se lleva a cabo la operación de lectura o escritura, desplazándose el sector bajo la cabeza; esta operación conlleva un tiempo de transferencia de datos, que depende de la velocidad de rotación del disco.

Tiempo medio de acceso: Tiempo medio que tarda la aguja en situarse en la pista y el sector deseado; es la suma del Tiempo medio de búsqueda (situarse en la pista), Tiempo de lectura/escritura y la Latencia media (situarse en el sector).

**Discos Óptimos: CD y CD-ROM**

Comparten una tecnología similar. Ambos discos se forman a partir de una resina, como un policarbonato, y se cubre con aluminio. La información grabada digitalmente se graba como una serie de hoyos microscópicos en el aluminio con un láser de alta densidad para crear el disco patrón. Este patrón se usa para hacer copias. La superficie con los hoyos de las copias se protege del polvo con una capa de laca transparente.

La información se recupera con un láser de baja potencia situado en un lector. El láser pasa a través de la capa protectora, mientras un motor hace girar el disco. La intensidad de la luz cambia si se encuentra un hoyo. Un fotosensor detecta este cambio, que es convertido en una señal digital.

El disco se divide en una serie de sectores y en una serie de pistas concéntricas. La información se empaqueta con densidad uniforme a lo largo del disco en segmentos del mismo tamaño y se explora a la misma velocidad, rotando el disco a una velocidad variable. El láser, por tanto, lee los hoyos a una velocidad lineal constante.

Comparado con los discos magnéticos tradicionales, tiene **tres ventajas principales: **

1. La capacidad de almacenamiento es mucho mayor.
2. Se puede replicar en grandes cantidades de forma barata.
3. Es extraíble, permitiendo usar el disco como memoria de archivo. 

**Las desventajas son**:

1. Es de sólo lectura
2. El tiempo de acceso es mayor que el de una unidad de disco magnético, tanto como medio segundo.

**Disco Optimo: DVD**

Sustituye a las cintas VHS de vídeo analógicas con su gran capacidad de almacenamiento, y sustituye al CD-ROM en los PC y servidores: el DVD lleva al video a la edad digital. Proporciona películas con una calidad de imagen impresionante; se pueden acceder aleatoriamente como en los CDs. Se puede grabar un gran volumen de datos (7 veces más que un CD-ROM). Claves que lo diferencia de los CD-ROM:

* Un DVD estándar almacena 4,7 GB por capa; los de doble capa y una cara almacenan 8,5 GB.
* Utiliza un formato de compresión (MPEG) par imágenes de pantalla completa de alta calidad.
* Un DVD de una capa puede almacenar una película de dos horas y media.

**Discos Magnéticos Ópticos (MO) preguntar**

Es un CD de una-escritura-varias-lecturas, para las aplicaciones en las que sólo se necesitan unas pocas copias de un conjunto de datos. Para proporcionar un acceso más rápido, el MO usa velocidad angular constante\*, sacrificando parte de su capacidad.

Técnica típica para fabricar el disco: usar un láser de alta potencia para producir una serie de ampollas en el disco; así un láser de baja potencia puede producir calor suficiente para reventar las ampollas pregrabadas. Este disco óptico es atractivo para almacenar archivos de documentos y ficheros; proporciona una grabación permanente de grandes cantidades de datos del usuario.

\* Velocidad angular constante: incrementa el espacio lineal entre bits de información grabados en los segmentos más externos del disco; explorando la información a la misma velocidad, girando el disco a una velocidad fija.

**Cinta Magnética**

Los sistemas de cinta usan la misma técnica de lectura y grabación que los discos. El medio es una cinta de plástico flexible, cubierta por un óxido magnético, análoga a una cinta de grabación doméstica. Se estructura en un pequeño número de pistas paralelas. Las primeras usaban 9 pistas (almacén de datos de un byte, con un bit de paridad); las nuevas usan 18 o 36 pistas (una palabra o doble palabra digital); los datos se leen y escriben en bloques contiguos (registros físicos) que están separados por bandas vacías (bandas inter-registros).

Es un dispositivo de acceso secuencial (tiene que leer los registros uno a uno para llegar al requerido; tiene que rebobinar un cierto trecho y empezar a leer si el registro deseado está en alguna posición anterior) que está en movimiento sólo durante las operaciones de lectura/escritura. Los extremos contienen unas marcas metálicas pegadas denominadas BOT y EOT para la detección automática del inicio y fin de la cinta, respectivamente. La capacidad depende de su longitud, densidad de grabación, longitud de bloque y formato de grabación. Las cintas magnéticas fueron el primer tipo de memorias secundarias.

Tipos de unidades de cinta magnética:

* **Cintas tradicionales de columnas de vacío:** las columnas de vacío tienen como objetivo mantener constante la tensión de la cinta bajo la estación de lectura/grabación.
* **Cintas tradicionales de brazos tensores:** Son más sencillas ya que no necesitan columnas de vacío, pero con ellas obtiene menor velocidad.
* **Unidades de casette de audio:** se utilizan en microcomputadoras domésticas y pequeños sistemas informáticos; el movimiento de la cinta se realiza con motores que actúan directamente sobre los carretes, no siendo tan rápidos ni precisos como las dos anteriores. - **Unidades de casette digitales:** disponen de cabestrantes para control de la velocidad de lectura/grabación.

#### **RAID**

Son esquemas estandarizados para el diseño de base de datos para discos múltiples. El esquema RAID consta de siete niveles independientes; desde cero hasta seis. Estos niveles no implican una relación jerárquica, sino que designan métodos diferentes que poseen tres características comunes:

1. RAID es un conjunto de unidades físicas de disco vistas por el sistema operativo como una única unidad lógica. 
2. Los datos se distribuyen a través de las unidades físicas del conjunto de unidades.
3. La capacidad de los discos redundantes se usa para almacenar información de paridad que garantice la recuperación de los datos en caso de fallo de disco.

Los detalles de las características segunda y tercera cambian según los distintos niveles RAID. RAID 0 no soporta la tercera característica.

La estrategia RAID reemplaza una unidad de disco de gran capacidad por unidades múltiples de menor capacidad, y distribuye los datos de forma que se puedan habilitar accesos simultáneos a los datos de varias unidades, mejorando, por tanto, las prestaciones de E/S, y permitiendo más fácilmente aumentos en la capacidad.

La única contribución de la propuesta RAID es, efectivamente, hacer hincapié en la necesidad de redundancia. El uso de varios dispositivos, además de permitir que varias cabezas y actuadores operen simultáneamente, consiguiendo mayores velocidades de E/S y de transferencia, incrementa la probabilidad de fallo. Para compensar esta disminución de seguridad, RAID utiliza la información de paridad almacenada, que permite la recuperación de datos perdidos debido a un fallo de disco.

**Nivel O de RAID:**

No es un verdadero miembro de los RAID porque no incluye redundancia para mejorar las prestaciones,Para el RAID 0, los datos del usuario y del sistema están distribuidos a lo largo de todos los discos del conjunto. Esto tiene una ventaja frente al uso de un único y gran disco: si hay pendientes dos peticiones diferentes de E/S para dos bloques de datos diferentes, entonces es muy probable que los bloques pedidos estén en diferentes discos. Entonces, las peticiones se pueden emitir en paralelo, reduciendo el tiempo de cola de E/S.

En RAID 0 los datos están organizados en forma de tiras de datos a través de los discos disponibles. Todos los datos del usuario y del sistema se ven como almacenados en un disco lógico. El disco se divide en tiras, que pueden ser bloques, sectores o alguna otra unidad. Un conjunto de tiras consecutivas se llama "franja".

**Nivel 1 de RAID**

RAID 1 se diferencia de los niveles 2 a 5 en cómo se consigue la redundancia. En este nivel, la redundancia se logra duplicando todos los datos. Como en RAID 0, se hace una distribución de datos, pero en este caso, cada franja lógica se proyecta en dos discos físicos separados, de forma que cada disco del conjunto tiene un disco espejo que contiene los mismos datos.

La organización RAID 1 tiene tres ventajas:

1. Una petición de lectura puede ser servida por cualquiera de los discos que contienen los datos pedidos, lo que hace que disminuya el tiempo de búsqueda
2. Una petición de escritura requiere que las dos tiras se actualicen y esto puede hacerse en paralelo. Entonces, el resultado de la escritura viene determinada por la menos rápida de las dos escrituras.
3. La .recuperación tras un fallo es sencilla

La principal desventaja es el coste. Requiere el doble de espacio del disco lógico que puede soportar.

Las prestaciones de RAID 1 son próximas al doble de las de RAID 0, en cuanto a transferencias de datos y peticiones de E/S.

**Nivel 2 de RAID**

Los niveles 2 y 3 de RAID usan una técnica de acceso paralelo. En un conjunto de acceso paralelo, todos los discos miembros participan en la ejecución de cada petición de E/S.

Como en los otros esquemas de RAID, se usa la descomposición de datos en tiras. En el caso de RAID 2 y 3, las tiras son muy pequeñas; a menudo, tan pequeñas como un único byte o palabra. Con RAID 2, el código de corrección de errores se calcula a partir de los bits de cada disco, y los bits del código se almacenan en las correspondientes posiciones de bit en varios discos de paridad. Normalmente se usa el código de Hamming, que permite corregir errores en un bit y detectar errores en dos bits.

Aunque RAID 2 requiere menos discos que el nivel 1, es bastante caro. El número de discos redundantes es proporcional al logaritmo del número de discos de datos. En una sola lectura se accede a todos los discos simultáneamente. El controlador del conjunto proporciona los datos perdidos y el código de corrección asociado. Si hay un error en un solo bit, el controlador lo corrige instantáneamente. En una escritura sencilla, la operación de escritura debe acceder a todos los discos de datos y de paridad.

RAID 2 debería ser, solamente una elección efectiva en un entorno en el que haya muchos errores de disco. Si hay una alta seguridad en los discos individuales y en las unidades de disco, RAID 2 es excesivo y no se implementa.

**Nivel 3 de RAID**

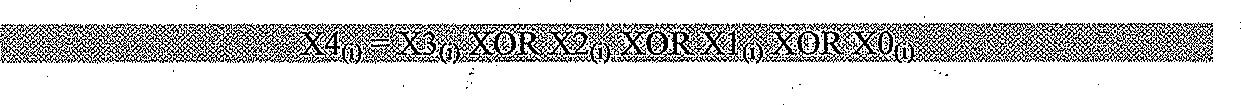
Se organiza de manera similar a RAID 2, la diferencia es que requiere sólo un disco redundante. RAID 3 usa un acceso paralelo, con datos distribuidos en pequeñas tiras. En vez de un código de corrección de errores, se calcula un bit de paridad para el conjunto de bits individuales están en la misma posición en todos los discos de datos.

RAID 3 puede conseguir velocidades: de transferencia, de datos muy altas y las mejoras de prestaciones en cuanto a E/S es notable. Por otra parte, sólo se puede ejecutar a la vez una petición de E/S; por lo tanto, el rendimiento sufre.

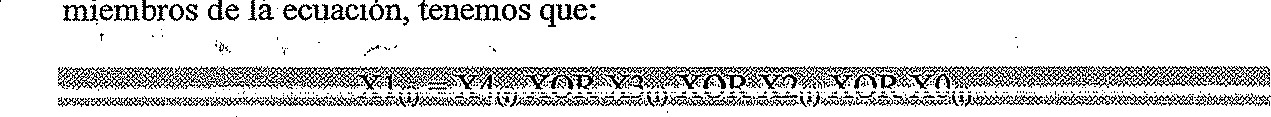
**Redundancia**

En el caso de un fallo en una unidad, se accede a la unidad de paridad y se reconstruyen los datos desde el resto de los dispositivos. Una vez que se sustituye la unidad que ha fallado, los datos que faltan se restauran en la nueva unidad y se reanuda la operación. 

La reconstrucción de los datos es bastante sencilla. Consideremos un conjunto de cinco discos, de los que de XO a X3 contienen datos, y X4 es el disco de paridad. La paridad para el i-ésimo bit se calcula de la siguiente forma:



Supongamos que la unidad XI ha fallado. Si sumamos X4(i) XOR XI (i) a ambos



Por lo tanto, se puede regenerar el contenido de cualquier tira de datos en XI a partir del contenido de las correspondientes tiras del resto de los discos del conjunto. Este principio es válido los niveles 3 a 6 de RAID.

En el caso de que un disco falle, todos los datos estarán todavía disponibles en lo que se denomina modo reducido. En este modo, para lecturas, los datos que faltan se recuperan "al vuelo" con la operación XOR. Cuando se escriben los datos en un conjunto RAID 3 reducido, se debe mantener la consistencia de paridad para regeneraciones posteriores.

Volviendo al funcionamiento global, se requiere que el disco que ha fallado se reemplace y se regenere todo su contenido en el nuevo disco.

**Nivel 4 de RAID**

Los niveles 4 y 5 de RAID usan una técnica de acceso independiente. En un conjunto de acceso independiente, cada disco opera independientemente, de forma que peticiones de E/S separadas se atienden en paralelo. Debido a esto, son más adecuados los conjuntos de acceso independiente para aplicaciones que requieren velocidades de petición de E/S altas, y son menos adecuados para aplicaciones que requieren velocidades altas de transferencia de datos.

Como en otros esquemas de RAID, se usan tiras de datos. El RAID 4 y 5 son relativamente grandes. Con RAID 4 se calcula una tira de paridad, bit a bit, a partir de las correspondientes tiras de cada disco de datos, y los bits de paridad se almacenan en la correspondiente tira del disco de paridad.

RAID 4 lleva consigo una penalización en la escritura cuando se realiza una petición de escritura de E/S pequeña. Cada vez que se realiza una escritura; el software de gestión del conjunto debe actualizar, no sólo los datos del usuario, sino también los bits de paridad correspondientes.

Para calcular la nueva paridad, el software de gestión del conjunto debe leer la antigua tira del usuario y la antigua tira de paridad. Entonces, se pueden actualizar las dos con nuevos datos y calcular la nueva paridad.

Cada operación de escritura implica al disco de paridad que se convertirá en un cuello de botella.

##### **Nivel 5 de RAID**

RAID 5 está organizado de manera similar al 4. La diferencia es que distribuye las tiras de paridad a lo largo de todos los discos. Una distribución típica es un esquema cíclico. Para un conjunto de n discos, la tira de paridad está en diferentes discos para las primeras n tiras, y este patrón se repite.

La distribución de las tiras de paridad a lo largo de todas las unidades evita el potencial cuello de botella de E/S encontrado en RAID 4.

##### **Nivel 6 de RAID**

En este esquema se hacen dos cálculos de paridad distintos, que se almacenan en bloques separados en distintos discos. Por tanto, un conjunto RAID 6 cuyos datos requieran N discos, consta de N+2 discos.

La ventaja de RAID 6 es que proporciona una disponibilidad de los datos extremadamente alta, porque además de la paridad calculada como RAID 4 y 5, tiene un algoritmo de comprobación independiente que hace posible la regeneración de los datos, incluso si dos de los discos fallan. Por otra parte, RAID 6 incurre en una penalización de escritura, ya que cada escritura afecta a dos bloques de paridad. 

UNIDAD 6 – PERIFERICOS

## **Periféricos**

Los periféricos son dispositivos que se conectan al computador mediante los módulos de E/S y que sirven para almacenar informacion o para llevar a cabo un tipo determinado de comunicación con el exterior (humanos u otras máquinas).

## **Comunicación hombre-maquina**

Permite la comunicación con el usuario de la computadora. Esta comunicación se realiza a través de periféricos (ej. un monitor)

**Terminales**

Las terminales de las computadoras constan de dos partes: un teclado y un monitor.

**Teclado**

Cuando utilizamos un teclado, al pulsar una tecla se cierra un conmutador que hay en el interior del teclado, esto hace que unos circuitos codificadores generen el código de E/S correspondiente al carácter seleccionado, apareciendo éste en la pantalla si no es un carácter de control. Los teclados contienen los siguientes tipos de teclas:

* **Teclado principal:** Contiene los caracteres alfabéticos, numéricos y especiales.
* **Teclas de desplazamiento del cursor:** Permiten desplazar el cursor a izquierda, derecha, arriba y abajo.
* **Teclado numérico:** Es habitual en los teclados de computadora que las teclas correspondientes a los caracteres numéricos (cifras decimales), signos de operaciones básicas y punto decimal estén repetidas para facilitar al usuario la introducción de datos numéricos.
* **Teclas de funciones:** Son teclas cuyas funciones son definibles por el usuario o están.
* **Teclas de funciones locales:** Controlan funciones propias del terminal, como impresión del contenido de imagen cuando el computador está conectada a una impresora.

**Monitores de CRT**

Un monitor es una caja que contiene un tubo de rayos catódicos (CRT) y sus fuentes de potencia. El CRT contiene un cañón que puede disparar un haz de electrones contra una pantalla fosforescente cerca del frente del tubo (los monitores a color tienen 3 cañones: rojo, verde y azul). Una imagen de pantalla completa normalmente se redibuja entre 30 y 60 veces cada segundo. Para producir un patrón de puntos en la pantalla, hay una rejilla dentro del CRT que, cuando se aplica en ella un voltaje positivo el haz choca la pantalla y brilla y, cuando el voltaje es negativo la pantalla no brilla. Así, el voltaje aplicado a la rejilla hace que el patrón de bits aparezca en pantalla. Este mecanismo permite convertir una señal eléctrica binaria en una imagen formada por puntos brillantes y oscuros.

**Monitor LCD (Pantalla de cristal líquido)**

una pantalla LCD consiste en dos placas de vidrio paralelas entre las que hay un volumen sellado que contiene cristal líquido. Cada placa tiene conectados electrodos transparentes que crean campos eléctricos en el cristal. Una luz ilumina la pantalla desde atrás. Diferentes partes de la pantalla reciben diferentes voltajes, y con esto se controla la imagen que se exhibe. En los LCD color se utilizan filtros ópticos para separar la luz blanca en componentes rojo, verde y azul.

Clasificación de pantallas:

* **Según la capacidad o no de mostrar colores se clasifican en:**
* **Monitor monocromo**: Los colores usuales en un monitor monocromático son el blanco y negro. (1 bit)
* **Monitor color**: El color de cada punto se obtiene con mezcla de los colores rojo, verde y azul. (24bits)
* **Según su capacidad de representación se pueden clasificar en:**
* **Pantallas de caracteres**: Sólo admiten caracteres.
* **Pantallas gráficas**: Permiten trazados de líneas y curvas continuas.

Principales parámetros que caracterizan a una pantalla:

* Tamaño: Se describen en función del tamaño de la diagonal principal, y se da en pulgadas.
* - Número de celdas o caracteres: lo usual es una representación de 24 filas \* 80 col. de caracteres.
* Resolución: número de puntos de imagen en pantalla. Este número no depende del tamaño de la pantalla.

Terminales de mapa de caracteres

Se usan comúnmente tres tipos de terminales: de mapa de caracteres, de mapa de bit RS-232C. Todas ellas pueden usar cualquier tipo de teclado, pero difieren en la forma en que se comunica con ellas y en el manejo de las salidas.

En una computadora personal hay dos formas de organizar las salidas que se envían a la pantalla: un mapa de caracteres y un mapa de bits. En el mapa de caracteres, en la tarjeta de comunicación en serie hay una porción de memoria, llamada memoria de video, además de algunos circuitos para acceder al bus y generar señales de video.

Para exhibir caracteres, la CPU los copia en la memoria de video en bytes alternados. Cada carácter tiene asociado un byte de atributo que describe la forma en que debe exhibirse el carácter.

La tarea de la tarjeta de video consiste en traer una y otra vez caracteres de la RAM de video y generar la señal necesaria que se alimenta al monitor.

Terminales de mapa de bits

Usan pixeles que representan un bit de información. La RAM de video se ve como un gran arreglo de bits. El software puede crear ahí cualquier patrón que desee, y se exhibirá instantáneamente.

Las terminales de mapa de bits suelen usarse para presentar imágenes que contienen varias ventanas. Usar mapa de bits tiene dos desventajas: la primera, requieren una cantidad considerable de RAM de video. La segunda desventaja es su desempeño, ya que es muy costoso desplazar la imagen en la pantalla.

Terminales RS-232-C

Se crearon con el fin de que cualquier terminal pueda usarse con cualquier computadora.

Tienen un conector estandarizado de 25 terminales. Para comunicarse, la computadora y la terminal contienen un chip llamado transmisor-receptor universal asincrónico (UART), además de lógica para acceder al bus. Las entradas del teclado de la Terminal se someten a una convención paralelo- serie en la Terminal y el UART las reensambla en la computadora.

**Mouse**

Es una pequeña caja de plástico que descansa sobre la mesa junto al teclado. Cuando el ratón se mueve sobre la mesa, también se mueve un puntero en la pantalla, que apunta a los elementos. El ratón tiene uno, dos o tres botones en la parte superior, que permiten seleccionar opciones de menú.

Existen tres tipos de ratones: mecánicos, ópticos y optomecánicos.

**Mecánicos:** tenían dos ruedas de caucho que sobresalían por debajo y tenían sus ejes perpendiculares entre sí. Cuando el ratón se movía paralelo a su eje principal, una rueda giraba. Cuando se movía perpendicularmente, giraba la otra rueda. Cada rueda impulsaba un resistor variable. Si se medían los cambios en la resistencia, era posible ver qué tanto había girado cada rueda y así calcular qué tanto se había movido el ratón en cada dirección. Este tipo se reemplazó por uno con una esfera que sale ligeramente de su base.

**Ópticos:** no tiene rueda ni esfera, tiene un LED y un fotodetector en la base. El ratón se usa sobre una base que contiene una cuadrícula de líneas. A medida que el ratón se mueve sobre ella, el fotodetector percibe el paso de las líneas por los cambios en la cantidad de luz que se refleja. Unos circuitos internos cuentan el número de líneas que se cruzan en cada dirección.

**Optomecánicos:** tiene una esfera que hace girar dos ejes perpendiculares entre sí. Los ejes se conectan a codificadores provistos de ranuras a través de las cuales puede pasar la luz. A medida que el ratón se mueve, los ejes giran y pulsos de luz inciden sobre los detectores cada vez que una ranura pasa entre un LED y su detector. El número de pulsos detectados es proporcional a la cantidad de movimiento.

**Tabla digitalizadora**

Está formada por un dispositivo apuntador movible más un panel sensible de hasta 1m2 de tamaño. La resolución puede llegar hasta milésimas de milímetro. El dispositivo apuntador suele ser una mirilla o lápiz electrónico similar al lápiz óptico que tiene 4 botones que sirven, entre otras funciones, para recoger las coordenadas de la posición actual.

Para determinar la posición del puntero se pueden diferenciar 3 métodos

* **Electromagnéticos**
* **Por ultrasonidos**
* **Por presión**

## **Comunicación máquina-mundo físico**

Permite la comunicación con dispositivos remotos. Para realizar esta conexión, se emplean una serie de periféricos que se denominan de forma genérica periféricos de control. Existen 4 tipos de periféricos de control: Cadena de lectura analógica, Cadena de lectura digital, Cadena de acción analógica, Cadena de acción digital.

La conexión con el módulo de E/S se realiza a través de señales de control, estado y datos. Los datos se intercambian en forma de un conjunto de bits que son enviados/recibidos a/desde el módulo de E/S. Las señales de control determinan la función que debe realizar el dispositivo. Las de estado indican el estado del chip.

**Puertos normalizados**

Permiten la conexión desde el exterior hacia la placa base. Funcionan con las características estándar y su fabricante tiene que conocer las normas estándar para que no surjan problemas de conexión con los periféricos. Las conexiones o puertos son:

* **Puertos Serie (COM):** Transmiten la información BIT a BIT. No son adecuados para transferir grandes cantidades de información. En ellos se conectan el modem, cuando es externo.
* **Puertos Paralelos (LPT):** Permiten una mayor velocidad en la transmisión de datos. La información se transmite en bytes o múltiplos. Es la conexión típica de impresoras o escáneres.
* **Puerto USB (Universal Serial Bus):** En ellos se conectan, en general, dispositivos que necesitan una alta velocidad de transferencia de datos como, por ejemplo, un escáner o una cámara digital, aunque también existen impresoras con este tipo de conexión.

**Impresoras**

Son periféricos que escriben la información de salida sobre papel. Su comportamiento inicialmente era muy similar al de las máquinas de escribir, pero hoy día son mucho más sofisticadas, pareciéndose algunas en su funcionamiento a máquinas fotocopiadoras conectadas en línea con la computadora.

Clasificación y tipos de impresoras:

**A) Calidad de impresión:** Tiene en cuenta la calidad de presentación y de contraste de los caracteres impresos.

* **Impresoras normales:** Como las impresoras de línea, de rueda y térmicas.
* **Impresoras de semicalidad:** Como algunas impresoras matriciales.
* **Impresoras de calidad:** Como las impresoras margarita e impresoras láser.

**B) Sistema de impresión:** Según la forma en que realizan la impresión.

* **Por impacto de martillos:** El fundamento de las impresoras por impacto es similar al de las máquinas de escribir. Las impresoras de impacto son muy ruidosas y han sido las más utilizadas.
* **Sin impacto:** forman los caracteres sin necesidad de golpes mecánicos y utilizan otros principios físicos para transferir las imágenes al papel.

**C) Forma de imprimir los caracteres:**

* **Impresoras de caracteres:** Realizan la impresión por medio de un cabezal que va escribiendo la línea carácter a carácter.
* **Impresoras de líneas:** En estas impresoras se imprimen simultáneamente todos o varios de los caracteres correspondientes a una línea de impresión.
* **Impresoras de páginas:** Aquí se incluyen un grupo de impresoras que actúan de forma muy similar a las máquinas fotocopiadoras.

Tipos de impresoras: esto si esta en teoría

* **Impresora de Matriz**: El tipo de impresora más económico es la impresora de matriz, en la que una cabeza de impresión que contiene entre 7 y 24 agujas activables electromagnéticamente se mueve a lo largo de cada línea de impresión. Las impresoras de matriz son económicas y muy confiables, pero son lentas, ruidosas y malas para imprimir gráficos. Se usan en tickets, o para imprimir en formatos grandes, entre otros.
* **Impresora de Inyección de Tinta:** La cabeza de impresión móvil, que lleva un cartucho de tinta, se mueve horizontalmente a lo ancho del papel mientras rocía tinta con sus boquillas. Dentro de cada boquilla, una pequeña gota de tinta se calienta eléctricamente más allá de su punto de ebullición, hasta que hace explosión, sale y choca con el papel. Luego la boquilla se enfría y el vacío que se produce succiona otra gota de tinta. Son económicas, silenciosas y con buena calidad, pero también son lentas, sus cartuchos son caros y producen impresiones saturadas de tinta.
* **Impresora Láser:** La impresora láser combina una alta calidad, excelente flexibilidad, buena velocidad y costo moderado en un mismo periférico. Permite usar todos los colores y puede imprimir páginas completas de texto o grafico a gran velocidad. Su desventaja consiste en que son costosas.
* **Impresoras térmicas:** Son similares a las impresoras de agujas. Se utiliza un papel especial termosensible que se ennegrece al aplicar calor. El calor se transfiere desde el cabezal por una matriz de pequeñas resistencias en las que al pasar una corriente eléctrica por ellas se calientan, formándose los puntos en papel.

Parámetros que caracterizan a una impresora:

**A) Velocidad de escritura:** normalmente se expresa en las siguientes unidades:

* Impresoras de caracteres: Caracteres por segundo (cps).
* Impresoras de líneas: Líneas por minuto (lpm).
* Impresoras de páginas: Páginas por minuto (ppm).
  1. **Caracteres por línea:** es el número máximo de caracteres que se pueden escribir en una línea.
  2. **Ancho del papel:** se suele expresar en pulgadas.
  3. **Densidad de líneas:** se expresa normalmente en líneas por pulgada e indica el espaciado entre líneas.
  4. **Color:** es la posibilidad de imprimir en colores o no.
  5. **Resolución:** la resolución se suele expresar como número de puntos por unidad de superficie.

**Scanner**

Es un dispositivo que recuerda a una fotocopiadora y que se emplea para introducir imágenes en un computador. Las imágenes que se desee capturar deben estar correctamente iluminadas para evitar brillo y tonos no deseados. Son dispositivos de entrada de datos de propósito especial que se emplean conjuntamente con paquetes software para gráficos y pantallas de alta resolución. La mayor parte de los scanner capturan imágenes en color generando una determinada cantidad de bits por cada punto. La cantidad de espacio de almacenamiento que se necesita para una imagen depende de las dimensiones máximas de la imagen y de la resolución de captura del equipo; la resolución se describe en ‘cantidad de puntos por pulgada’ (‘dot per inch’ en inglés o con el acrónimo ‘dpi’) en el sentido horizontal (eje x) y en el vertical (eje y).

## **Comunicación máquina-máquina**

Permite la comunicación con elementos de equipo (discos magnéticos, cintas, etc.). La comunicación es digital. Si los computadores están cerca se conectan directamente en Banda Base, mediante un bus paralelo que transmite bytes o palabras. Si existe una distancia de más de 100 mts. se utilizan Módems.

**Modem**

Es un dispositivo que permite conectar dos computadores remotos utilizando la línea telefónica de forma que puedan intercambiar información entre sí. La información que maneja el computador es digital, sin embargo, por las limitaciones físicas de las líneas de transmisión, no es posible enviar información digital a través de un circuito telefónico, solo pueden transmitirse señales analógicas. Entonces, para poder utilizar las líneas telefónicas (y en general cualquier línea de transmisión) para el envío de información entre computadoras digitales, es necesario un proceso de transformación de la información. Durante este proceso la información se adecua para ser transportada por el canal de comunicación. Este proceso se conoce como modulación-demodulación.

Un **Módem** es un dispositivo que posee conversores A/D y D/A especialmente adecuados para conectar líneas telefónicas al computador. De este modo las señales provenientes de una línea telefónica (por ejemplo, por una llamada) son interpretadas y “atendidas” por el módem, permitiendo que otra computadora trasmita información directamente a la nuestra. Recíprocamente, nuestra voz, una imagen o datos guardados en nuestra computadora pueden ser manejados por el módem para ponerlos (previa llamada) sobre la línea telefónica que del otro lado tendrá una computadora receptora (o eventualmente un ser humano que descuelga un teléfono).

* Baudios: los módems envían datos como una serie de tonos a través de la línea telefónica. Los tonos se "encienden" (ON) o apagan (OFF) para indicar un 1 o un 0 digital. El baudio es el número de veces que esos tonos se ponen a ON o a OFF. Los módems modernos pueden enviar 4 o más bits por baudio.
* Bits por segundo (BPS): es el número efectivo de bits/seg que se transmiten en una línea por segundo. Un módem de 600 baudios puede transmitir a 1200, 2400 o, incluso a 9600 BPS.

¿Qué parámetros tiene un modem? Pregunta de un final