Von Neumann

1. **La arquitectura von Neumann explicaba una memoria única interconectada con CPU y E/S. ¿es posible otra arquitectura con memoria para programas separado de otro para datos? ¿qué cambios deberían hacerse en la CPU y las interconexiones para q la maquina funcione? Analice ventajas y desventajas.**

Si es posible de hecho, existe la arquitectura denominada hardware q utiliza esa estructura de memoria. La memoria del programa y la de datos constituyen dos espacios separados cuyo acceso puede ser mediante buses distintos, es decir, puede haber dos buses de direcciones, un bus de datos y un bus de instrucciones. En el caso q haya un único bus de direcciones, debe existir alguna señal de control q permita diferenciar (ej: señales de READ, WRITE, FETCH). Las ventajas: instrucciones y datos de distinto largo, memorias de distinto tamaño y tecnología, permite el acceso simultaneo a instrucciones y datos, por lo q disminuye el cuello de botella en el acceso a memoria de la arquitectura von Neumann. Las desventajas: complejidad del sistema, debido a la inclusión de rutas separadas para datos e instrucciones.

Descripción de cómo se arma una pc: prog, datos->mem principal construidas por-> celdas = tamaño identif por-> dirección de memoria. Instrucciones almacenadas por celdas ubicadas consecutivamente.

1. **Describir los elementos de la IAS (Von Neumann). Qué diferencias hay con las actuales?**

5 componentes principales:

-unidad de entrada: provee las instrucciones y los datos.

-unidad de memoria: donde se almacenan los datos e instrucciones a los cuales se accede por dirección (posición) dentro de la misma sin considerar el tipo.

-unidad aritmética lógica: procesa los datos (encargada de operaciones aritméticas y lógicas).

-unidad de central: dirige la operación (interpreta las instrucciones almacenadas en memoria para su ejecución).

-unidad de salida: se envían los resultados.

Diferencias con las maquinas actuales: la ALU era para trabajar con bits enteros. Hoy también tenemos la alu en punto flotante. Además tenemos registros; la memoria cache antes no existía . Hoy está dentro de la cpu; el avance notable en la velocidad de los microprocesadores; aumentaron la capacidad y el tamaño de las celdas de memoria; dejamos de tener unidad de entrada salida paralelos. Y ahora tenemos nuevas puertas de entrada como el USB. Se modernizó a nuevas técnicas “inalámbricas”.

**VN: 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 sinimportar el dato almacenado.

**Que ventajas presentan las familias de computadoras?**

Los distintos modelos de una familia eran compatibles en el sentido de q un programa hecho para un modelo, podía ser ejecutado por otro modelo, con la única diferencia del tiempo de ejecución.

Características y ventajas: conjunto de instrucciones similar o idéntico. Sistema operativo similar o idéntico. Memoria creciente. Mayor velocidad. Mayores números de puertos de e/s. Sin embargo el costo es creciente.

**Concepto de programa:** Para cada paso se necesita un nuevo conjunto de señales de control.Las instrucciones proporcionan esas señales de control**.** Aparece el nuevo concepto de programación**.** ¡No hay que cambiar el hardware!

**¿Qué es un programa?:** Es una secuencia de pasos. Se hace una operación aritmético/lógica por cada paso. Diferentes señales de control se necesitan para cada operación: la UC saca información de cada instrucción.

**Características de IAS (maq de von Neumann):** Memoria con 4096 palabras de 40 bits: Números Binarios, 2 instrucciones de 20 bits. Set de registros (almacenamiento en CPU): Registro Buffer de Memoria (MBR); Registro de Direcciones de Memoria (MAR); Registros de Instrucción y Buffer de Instrucción; Registro Contador de Programa (Program Counter); Registros Acumulador y Multiplicador/Cociente.

Diseñada en 1945, estableció la arq de las computadoras que, con algunas modificaciones, es empleada hoy en día. Constituida por: ALU: encargada de las operaciones aritméticas y lógicas; Unidad de control: Interpretar instrucciones almacenadas y ejecutarlas; Disp de E/S: para comunicarse con el interior; Mem principal: donde se almacenan los datos e instrucciones.

La ejecución de instrucciones se realiza siguiendo una secuencia, una tras otra. Se dividen en 5 grupos: trasferencias de datos, salto incondicional, salto condicional, aritméticas, modificación de direcciones.

**Arq von Neumann :** La unidad central de procesamiento(cpu) está constituida por la unidad de control (uc) y la unidad arit-log(alu). Los datos e instrucciones deben introducirse en el sist y los resultados se proporcionaran mediante componentes de e/s--> la maq genera resultados. Se necesita almacenar temporalmente datos e instrucciones en la mem principal de la computadora. Esa m. principal está constituida por celdas, todas del mismo tamaño, identificadas por una direc de memoria. Las instrucciones tienen que estar almacenadas por celdas ubicadas consecutivamente (una detrás de otra). Utiliza un sist binario que simplifica la implementación de funciones y disminuye la probabilidad de fallos. Además la ejec del programa se realiza en forma secuencial.

**Assembly:** es un lenguaje que define un mapeo directo de códigos (codops) a instrucciones comprensibles por un humano. Assembler es el programa que genera códigos que la PC entenderá.

Ciclo de Instrucción

1. **En qué momento del ciclo de instrucciones se fija la CPU si hay pedida de instrucción, Por qué? Describa los pasos q se llevan a cabo cuando se encuentra el pedido. (INTERRUPCIONES)**

Para saber si hay interrupción la CPU se fija el estado del flag de interrupción, si es 1 hay interrupción.

Se guarda el contexto (la palabra de estado del procesador). Necesita un lugar de almacenamiento rápido ya q necesita llegar al gestor de int. Lo más rápido posible. Tmb guarda la dirección de retorno. Las próximas direcciones y la información de lo ejecutado se guardan en memoria en la pila ya q esta es rápida. Primero guarda la palabra de estado y desp la dirección de retorno.

Cuando hago un CALL tmb uso el puntero de pila antes de cambiar la dirección del proc. Debo guardar la dirección de retorno para ello uso la pila. El CODOP de MOV reg, reg es distinto al de MOV reg, [reg] ya q es diferente MDD. No puedo usar MDD inmediato para mover de memoria a registro.

Las diversas formas con la q se pueden hacer diferencias tanto a los operandos fuente como los resultados, originan los MDD. Este procedimiento sirve para determinar la ubicación del operando q dependiendo del tipo de instrucción, puede estar en la propia instrucción, memoria principal o en registros internos de la CPU.

Tipos de interrupciones:

Por programa: como resultado de una instrucción, se genera una condición.

Temporizador: generada por un temporizador interno al procesador.

E/S: generada por un controlador e/s. para indicar que no hubo problemas o para divisar un error.

Fallo de hardware: generada por un fallo como falta de potencia o un error de la memoria.

**Ciclo de instrucción básico:**

- Dos pasos: Captación. Ejecución

INICIO --> CAPTAR LA SIG INSTRUCCION --> EJECUTAR LA INSTRUCCION --> Parada -->CICLO DE CAPTACION >CICLO DE EJECUCION

**DIAGRAMA DE ESTADOS DEL CICLO DE INSTRUCCIÓN :**

1 Calculo de la dirección de instrucción. 2 Captación de instrucción. 3 Decodificación de la operación de la instrucción. 4 cálculo de la dirección de operando. 5 captación de operando. 6 operación de datos. 7 cálculo de la dirección de operando. 8 almacenamiento del operando.

1. **Q es un ciclo de instrucciones ¿?**

Es el periodo q tarda el CPU en ejecutar una instrucción del lenguaje máquina para esto, se tiene en cuenta el ciclo de búsqueda (pasos = a todas las instrucciones) y el ciclo de ejecución (pasos cambian depende la instrucción). Pasos a seguir en una instrucción:

Buscar la instrucción en la memoria principal: se vuelca el valor del pc sobre el bus de direcc.

Decodificar la instrucción: el decodif interpreta e implementa la instruccion

Ejecutar la instrucción

Almacenar el resultado: en mem principal o enviado a un disp. De salida.

Memoria

Tipos de memoria

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipos de memoria | clase | Borrado | Escritura | Volatilidad |
| RAM(de acceso aleatorio) | Mem de lect/ escr | Eléctricamente por bytes | Eléctricamente | Volátil |
| ROM(de solo lectura) | Mem de solo lectura | No es posible | Mediante mascaras | No volátil |
| PROM(rom programable) | “ | “ | Electricamente | “ |
| EPROM(PROM borrable) | Mem de sobre lectura | Luz ultravioleta, chip completo | “ | “ |
| Mem FLASH | “ | Electricamente por bloques | “ | “ |
| EEPROM(prom borrable eléctricamente) | “ | Electricamente por bytes | “ | “ |

1. **Cuáles son las características principales de la organización de memoria 2 ½ D (memoria dinámica)? Describir la memoria 2 D(memoria estática). Indicar diferencias de memoria.**

**Describe y grafique el conexionado de un subsistema de memoria 256 mega palabra de 32 bits realizada con chips de 128 Mb de memoria. Nota: el subsistema se conoce como de 1 gb.**

Organización 2 D: se puede pensar como una lista de celdas apiladas, las cuales tienen un tamaño N. Tiene una conexión q le permite conseguir los datos a ingresar (bus datos) y otra q le indica q celda tiene q escribir (decodificador) (siempre se lee o escribe una celda completa). Debería tener otra conexión con el lugar a donde tiene q guardar los datos en el caso de una lectura, pero se puede utilizar la misma conexión para leer o escribir, indicando con un flag q operación hacer. Este tipo de disposición es físicamente muy difícil de implementar con una cantidad de registros grandes, pero el acceso a las celdas es constante.

Memorias de poco tamaño, lo cual es difícil de implementar con registros grandes (tamaño entre 512kb - 1mb. Si tengo mil celdas y quiero poner 2mil, que hago? replico las celdas pero a la primeras le agrego un bit 0 a cada celda y a la segundas les pongo un 1 para indicar que son distintas.

Organización 2 ½ D: Tiene una estructura compuesta por varios chips, s cuales deben retornar el valor de un bit, el cual se utiliza para formar la palabra n bits q se quiere leer o escribir. Es por ello q es necesario proveerle la dirección q quiere acceder. La misma se divide en 2 partes (a través de decodificadores), una q indica q “reglón” se quiere leer y otra q indica en q parte de ese chip se debe buscar el bit correspondiente a una fracción del reglón a devolver. Una vez identificado cada bit de la palabra se reconstruye la misma y se la ubica a la interfaz de la salida.

La organización 2 ½ D mantiene una gran ventaja con respecto a la 2 D: puede extender su capacidad, a pesar de q obtener una palabra requiere más procesamiento. Por una cuestión de costos la 2 ½ D se construye con transistores (son más baratas q las 2 D) por lo q son más lentas, teniendo en cuenta el procesamiento q lleva decodificar una dirección.

-2 D: memorias estáticas (registros cache).

-2 ½ D: memorias dinámicas (DRAM (memoria principal)).

-3D (modelo teórico): tiene implementado toda la estructura para funcionar el 2D ½. C/U de los chips reflejara una capa de modelo 3D.

|  |  |  |
| --- | --- | --- |
| Aspecto | Memoria 2D | Memoria 2D 1/2 |
| Espacio, físico | Muy larga y estrecha, pudiendo guardar muchas palabras en pocos bits. Ocupa mucha superficie. | Tiende a ser rectangular. Ocupa poca superficie en relación a la cantidad de información. |
| Corrección de error | Difícil de usar. | Hay menos chances de error por ser distintos chips. |
| Complejidad de los decodificadores | Mucha porque son muchas palabras que manejar. | Se reduce la complejidad porque usa decodificación separada de las filas y columnas. |
| Ubicación de los bits | Todos en el mismo chip. | Los de una misma palabra están en distintos chips. |

1. **Describa las características de las maquinas q ejecutan instrucciones de 2 direcciones. Compare la cantidad de instrucciones y las de acceso a memoria.**

Las maquinas q tiene instrucciones de 2 direcciones utilizan una de las direcciones para hacer el servicio doble de uno de los operandos y del resultado. Este formato reduce el espacio necesario. La dirección de la instrucción siguiente está implícita en la pc.

Tienen los siguientes elementos la maquina q ejecuta las instrucciones de 2 direcciones: Código de operación y operando; Código de operación y 2 operandos; Código de operación y 2 operandos(el primero se utiliza para almacenar resultados); Código de operación.

Las instrucciones de una dirección son más simples y deben tener una segunda dirección implícita. Para ello se utiliza el registro acumulador, q contiene uno de los operandos y se emplea para almacenar el resultado. La dirección de la próxima instrucción tmb está implícita en la pc.

1. **Cuáles son los principios q sustentan el funcionamiento de una jerarquía de memoria? Mencione las características propias de cada nivel en términos de capacidad de almacenamiento, tiempo de acceso y tecnología de soporte.**

Las restricciones de diseño de la memoria pueden resumirse en 3 cuestiones ¿Cuánta capacidad? , ¿Q velocidad?, ¿A q precio?. Se sustenta en el principio de “localidad de referencia”.

El diseñador desearía utilizar tecnologías de memoria q proporcionen gran capacidad, pq esta es necesario y pq el costo por bit es bajo. Para ello debería utilizar memorias costosas, de capacidad relativamente baja y con tiempos de accesos reducidos. Es por ello q se usa una jerarquía de memoria (objetivo : mejorar la capacidad de almacenamiento, tiempo de acceso reducido), q combina una cantidad pequeña de memoria rápida con una cantidad grande de memoria lenta, para obtener casi la velocidad de la memoria rápida y la capacidad de la memoria grande a un precio moderado. Cuando ascendemos, también aumenta la frecuencia de accesos a ese tipo de memoria. Se denomina jdm a la forma en que se organizan los distintos tipos de memoria.

Duración: M volátiles: RAM; M no volátiles: discos, cintas; M permanentes: ROM, EPROM.

Modo de acceso: por palabra: m principal, registros; acceso por bloque: discos mag y opt, caché; registro:cintas

Tiempo de acceso (m magnéticas): t de posicionar el cabezal + t de latencia (+ t de lectura).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipos memoria | Tiempo acceso | Tamaño típico | Rápida y clara | | |
| Registros | 1ns | 1KB | ^ | | |
| M. cache | 5-20ns | 1MB | | | | |
| M. principal | 60-80ns | 1GB | | |  |
| Discos magnéticos | 10ms | 160GB | | | Lenta y barata |
| Cintas | ¿min | ilimitada |  |  |
| Discos opt cd/dvd | 100ms |  |  |  |

1. Métodos de acceso a la info almacenada en mem secundaria.

-Secuencial: el acceso debe hacerse en una secuencia lineal específica. Variable. Unidades de cintas

-Directo: los bloques ó registros individuales tienen una dirección única que se basa en la localización física. Variable. Discos magnéticos, discos opt.

-Aleatorio: el tiempo para acceder a una locación dada es independiente de lasecuencia de accesos anteriores y es constante. mem principal.

-asociativa: memorias cache, registros

1. **Q es la memoria RAM dinámica y RAM estática? ¿q diferencia notoria hay entre ellas?**

Son dos variantes de la tecnología usada en la RAM. Son memorias de acceso aleatorio. Es posible leer o escribir nuevos datos mediantes señales eléctricas. Solo se pueden utilizar como almacenamiento temporal pq son volátiles, es decir, si se corta la energía, se pierde la información.

Una RAM dinámica está hecha con celdas q almacenan los datos como cargas en condensadores y requieren refrescos periódicos para mantener memorizados los datos. En una estática los valores se almacenan utilizando configuraciones de puertas q forman biestables.

Una celda de RAM dinámica es más simple y pequeña q una de RAM estática, por lo tanto más densas y más económicas, pero requieren circuitería para el refresco.

La ventaja de las estáticas es q son, generalmente, más rápidas q las dinámicas.

1. **Qué es y para qué sirve la memoria cache?**

La cache es un tipo de memoria q se encuentra, en la jerarquía, entre los registros y la memoria principal. Es un sist especial de almacenamiento de gran velocidad. Se sustenta en el principio de localidad. Su objetivo es q la velocidad de la memoria sea lo más rápido posible. La cache contiene una copia de partes de la memoria principal. Cuando el procesador quiere leer una palabra de memoria, se “fija” si está en la cache. Si es así, se entrega al procesador. Si no, se trae un bloque de memoria principal a la cache y después la palabra es entregada al procesador. La base de todos los sistemas de cache es q las referencias a la memoria q se hacen en un periodo corto tienden a usar una fracción pequeña de la memoria total. Esto se denomina principio de localidad. La idea es q cuando se hace referencia a una palabra, ella y algunas de sus vecinas se traen de la memoria grande a la cache, para q la siguiente vez q se use, el acceso a ella sea rápido.

1. **Por q conviene más de un nivel de cache?**

Es posible tener una cache en el mismo chip del procesador, q reduce la actividad del bus externo del procesador y los tiempos de ejecución e incrementa las prestaciones globales del sistema. Cuando la instrucción o dato requeridos se encuentran en la cache on-chip se efectúan apreciablemente más rápidos q los ciclos del bus. Además, durante este periodo el bus está libre para realizar otras transferencias. Esta estructura se conoce como cache de dos niveles. Si no hay cache L2 (externa al procesador) y este hace una petición de acceso a una posición de memoria q no está en la cache L1 (interna al procesador), entonces este debe acceder a la DRAM o ROM a través del bus, obtenido bajas prestaciones. Si se utiliza una cache L2 SRAM, entonces, con frecuencia, la información q falta puede recuperarse fácilmente.

1. **Describe las características q tienen las maquinas q ejecutan instrucciones con una y dos direcciones. Suponiendo q se poseen las instrucciones adecuadas: Resuelva mediante programa la ecuación : x= (a+b).c para cada maquina.**

Las instrucciones de 2 direcciones utilizan una de las direcciones para hacer el servicio doble de uno de los operandos y del resultado. Este formato reduce el espacio necesario (48 bits de referencias). La dirección de la instrucción siguiente esta implícita en el PC. Hay que mover el Op1 a un registro temporal. Menos elección donde guardar el resultado. Add DirOp1 DirOp2

Las instrucciones de 1 dirección son mas simples y deben tener una segunda dirección implícita. Para ello se utiliza el registro acumulador, q contiene uno de los operandos y se emplea para almacenar el resultado. La dirección de la próxima instrucción también esta implícita en el PC. Instrucción más corta (24 bits de referencias) Add DirOp1

Máquina para 4 direcciones: Direcciones explícitas para operandos, resultado y próxima instrucción. Son “raras”, cada campo de dirección tiene que tener bits para “acomodar” una dirección completa. 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.

Add DirRes DirOp1 DirOp2 DirPróxIns

Máquina para 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. Add DirRes DirOp1 DirOp2

Dos direcciones: 1 direccion:

ADD A, B LOAD A AC <- A

MUL A, C ADD B AC + B

MOV X, C MUL C AC. C

STORE X X <- AC

1. **Qué mejoras podemos obtener en el funcionamiento de máquina que ejecuta instrucciones debido al principio de localidad de referencia?**

El principio de localidad de referencia establece q las referencias a memoria forman agrupaciones y q durante periodos cortos de tiempo el procesador trabaja con agrupaciones fijas de transferencias a memoria. Esto puede ser aprovechado formando una memoria de 2 niveles, la memoria de nivel superior (M1) es más pequeña, rápida y costosa q la del nivel inferior (M2). M1 se usa como almacenamiento temporal de bloques de alguna parte temporal de M2. Cuando se hace una referencia a memoria primero se busca en M1. Si el elemento está ahí, el acceso es rápido. Si no, se copia un bloque de M2 a M1, y el acceso se hace vía M1. Debido a la localidad, habrá un número de accesos posiciones de ese bloque, resultando un servicio rápido en sus conjuntos.

**Localidad de referencia**: principio de localidad

Nos permite predecir que instrucciones y datos se utilizan en un futuro próximo los casos mas importantes son 2:

-localidad temporal: si una posición de memoria es referenciada, es muy probable que la misma ubicación sea referenciada en un futuro cercano

-localidad espacial: si una localización de memoria es referenciada, es muy probable que las localizaciones cercanas a ella sean referenciadas.

Beneficio: el saber que yo en mi ciclo de instrucción tengo que ir primero a buscar la instrucción el saber que por anticipado voy a tener la siguiente me permitiría armar distinto tipos de memorias o utilizar distintos tipos de memorias pasamos de usar memoria lenta a registros más rápidos. El principio de localidad nos dará sustento a poder armar jerarquía de memoria.

**Ventajas y/o usos almacenamiento de RAID**

- 7 niveles propuestos (0 a 6). No es una jerarquía. Conjunto de discos físicos vistos como una sola unidad lógica por el S.O. Datos distribuidos en los discos físicos. Puede usarse capacidad redundante para almacenar información de paridad.

-0: No protege datos, separa. Discos mín. Capacidad: 4 de 40GB =160GB; 1 de 40GB y 3de 60GB = 160GB

-1:Redundancia. 2 Discos mín. Capacidad: 2 de 40GB=40GB; 1 de 40GB y 1 de 60GB= 40GB.

-2 y 3: Separa. Paridad. Capacidad: 3 de 80GB= 160GB.

-4 y 5: Separa. Paridad repartida. Capacidad: 3 de 80GB= 160GB.

-6:Separa. Paridad doble. Capacidad: 4 de 80GB = 160GB

RAID 0-1: Separa. Redundancia. 4 Discos mínimo. Capacidad: 4 de 40GB = 80GB; 1 de 40GB y 3 de 60GB = 80GB.

Beneficios: mayor integridad, mayor tolerancia a fallos, mayor rendimiento, mayor capacidad.

Formato de instrucción

1. **¿Cuáles son los elementos a tener en cuenta para el diseño del conjunto de instrucciones de un procesador? Qué son los modos de direccionamiento? Describa las variantes al denominado por desplazamiento. Qué operaciones de tipo aritmético podemos encontrar en un repertorio de instrucciones? Por qué los distintos tipos de datos q desean utilizarse en esas operaciones condicionan el hardware necesario para operar?**

(Conjunto de instrucciones(medio que tiene el programador para controlar la cpu)). Los aspectos de diseño son:

-repertorio de instrucciones: cuantas y q operaciones considerar (aritméticas, lógicas, de conversión, de entrada/salida, etc.).

-tipos de datos (**OPERANDO**): cuáles serán los datos con las q se puede ejecutar operaciones (direcciones, caracteres (codigo ascii), datos lógicos (Bits (1 o 0) ej: flags o indicadores), datos enteros, punto flotante, coma fija (entera y fraccionaria)).

-formatos de instrucciones: longitud de instrucción, numero de direcciones, tamaño de los campos. Es el esquema donde cada instrucción está representada por una secuencia de bits. La secuencia se divide en campos.

-registros: cantidad q pueden ser referenciados por instrucciones.

-direccionamiento: modos mediante los cuales puede especificarse la dirección de un operando.

Los modos de direccionamiento es la forma mediante la cual se puede referenciar la posición en memoria de algún operando o dato. Es la acción de asignar una dirección de memoria a un conjunto de datos. Son medios q facilitan la tarea de programación, permitiendo el acceso a los datos de manera natural y eficiente. Estos indican al procesador como calcular la dirección absoluta donde se encuentran los datos. En una instrucción se utilizan bits para expresar el cod de operación (codop) (se necesitan gran cant de bits).

A: contenido de un campo de dirección a la instrucción.

R: contenido de un campo de dirección en la instrucción que referencia a un registro.

EA: dirección real de la posición que contiene el operando.

(X): contenido de la posición X

Objetivos: la dirección puede que no se conozca hasta el momento de ejecutar el programa operando -> reducir el espacio ocupado en memoria por las instrucciones.

Permitir la reubicación del código. Facilitar el manejo de la estructura de datos (arreglos).

Tipos:

-Inmediato: el operando se obtiene automáticamente de la memoria al mismo tiempo que la instrucción. No requiere una referencia extra de memoria de datos (ventaja). Se utiliza para definir constantes y para inicializar variables. Desventaja: tamaño del operando limitado por el tamaño del campo de direccionamiento.

-Directo memoria o absoluto: el campo de dirección tiene la dirección efectiva del operando. Se usa para acceder a variables globales, cuya dirección se conoce en el momento de compilación. Es simple pero tiene espacio limitado de direcciones por la cantidad de bits del campo. Uso: acceder a variables globales, cuya dirección se conoce en el momento de compilación.

-Directo por registro: se especifica un registro en lugar de una posición de memoria. Usa menos bits y no requiere acceso a memoria de datos. La desventaja es que los registros no son muchos y es un recurso preciado.

-Indirecto por memoria (desuso): 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.

-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.

Las variantes del direccionamiento por desplazamiento son: 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.

-direccionamiento relativo: se referencia implícitamente el pc. La dirección de instrucción actual se suma al campo de direcciones q se trata en CA2 para esta operación.

-direccionamiento con registro- base: el registro referenciado contiene una dirección de memoria y el campo de dirección contiene un desplazamiento (en BSS) desde dicha dirección.

-direccionamiento indexado: el campo de dirección especifica una posición en memoria y el registro referenciado contiene un desplazamiento positivo desde esa dirección.

-pila o stack: se automodif para apilar o desapilar un dato del programa. El stack o pila es un arreglo lineal de localidades de memoria. Es una lista o cola donde el último en entrar es el primero en salir. Es una zona de mem reservada. Asociado con la pila o stack hay in reg apuntador cuyo valor es la direc tope de pila o stack.

**ELEMENTOS DE UNA INSTRUCCION:**

1 Codigo de operacion (cod op): especifica la operación a realizar. Es un cod binario.

2Referencia a operandos fuentes: establece donde se encuentra el operando. ( la operación puede involucrar mas de un operando fuente (o de entrada).

3Referencia al operando resultado: establece donde almacenar el resultado.

4Referencias a la siguiente instrucción: le dice a la cpu donde buscar la sig instrucción.

2y3: pueden estar en 3 lugares: mem, reg de la cpu o disp. e/s.

**TIPOS DE INSTRUCCIONES:**

--Procesamiento de datos: operaciones:

Aritméticas: add, sub, mul, div (utiliza la memoria y la ALU). números enteros sin/con signo. Números en punto flotante. Pueden incluirse operaciones: Inc o dec (en 1 el operando); Neg: cambia el signo del operando (Ca2) (necesita la unidad de control además de la ALU); Abs: toma el valor absoluto del operando; shift left/right: desplaza bits a izq/der un lugar.

Lógicas: Operaciones que manipulan bits individualmente. Operaciones booleanas: AND, OR, XOR, NOT. Otras operaciones: Rotate left/right: rota las posiciones de los bits a izq/der.

--Almacenamiento de datos: transf dentro del sist. MOV, PUSH, POP, PUSHF, POPF

-Instrucciones de memoria

--E/S: Transferencia de datos: -Debe especificarse: -ubicacion del operando fuente -ubicacion del operando destino -tamaño de los datos a ser transferidos -modos de direccionamiento. Operaciones:

-Diferentes movimientos -> diferentes instrucciones. -reg,reg ; reg,mem ; o mem,reg.

- O una instruccion y diferentes direcciones -MOV destino, fuente ; copia fuente a destino

-Instrucciones de E/S: -Pocas instrucciones pero de acciones especificas IN o OUT

-Se pueden realizar utilizando inistrucciones de movimiento de datos MOVE

-Se pueden realizar a traves de un controlador aparte: DMA (Direct Memory Access)

-- Transferencia de Control: Instrucciones de testeo y flujo del programa: Operaciones:

Control de flujo (SALTO) (en estas instrucciones necesito el Contador de Programa y la ALU). Modifican el valor contenido en el registro PC:

-Salto Incondicional: -JMP equis ; saltar a la posicion 'equis'. Se puede usar el directo de memoria, o directo de registro o indirecto por registro.

-Salto Condicional: JZ equis ; saltar a la posicion equis, si bandera Z = 1 (mdd propio, la direccion queda almacenada ahi). Los saltos condicionales usan indirecto por desplazamiento: basado(ESTA PENSADO PARA DAR VELOCIDAD DE EJECUCION); indexado(el registro se modifica el valor cada vez que se usa) - para recorrer vectores; relativo al PC: indirecto via registro contador de programa (SALTOS CONDICIONALES).

-Salto con retorno o llamada a subrutina: -CALL subrut ; saltar a la posicion 'subrut' (automaticamente cuando se llama guarda la direccion de retorno). Para retornar al programa que llamó, se debe utilizar la instruccion RET como ultima del cuerpo de subrutina

--De control: control de ejecución del programa. NOP HLT

--Manejo de interrupciones: op para el control de interrupciones. INT, IRET, CLI,STI

1. **Describa las instrucciones de tipo transferencia de control q conozca y mencione los usos típicos q haría de ellas. Q modos de direccionamientos pueden ser usados en ellas?**

-salto incondicional: -JMP: es un salto q se ejecuta siempre. Se escribe JMP SALTO donde SALTO la etiqueta q marca la línea a la q quiero saltar. La instrucción para la maquina tiene un formato: Codop + dirección donde dirección es el lugar al q tiene q saltar. El direccionamiento es directo.

-salto condicional:-JC (c= condición): el salto se lleva a cabo solo si la condición es verdadera . las condicione s coinciden con los flags como carry , overflow, zero y sus negaciones. Se escribe JC SALTO (igual q en el JMP) pero la instrucción para la maquina es codop + desplazamiento donde desplazamiento es un numero en CA2 q indica el valor q se suma al PC para llegar a la posición deseada. El direccionamiento se llama relativo al contador del programa. –JZ.

-salto con retorno:-CALL : llama a una subrutina . Cuando se ejecuta el valor actual del PC se guarda en la pila y el SP (stack pointer) pasan apuntal al donde se guardo. Luego se carga en el PC la dirección del comienzo de la subrutina para q se ejecuten las instrucciones de la subrutina hasta q llegue a RET q devuelve al PC lo q apunta el SP. Se escribe CALL SUBRUT, donde SUBRUT es el nombre de la subrutina q queremos invocar.

Para la maquina el formato es codop + dirección, donde la dirección es el comienzo de la subrutina. El direccionamiento es directo.

Teorema fundamental de la numeración

1. **Qué define el teorema fundamental de la numeración?**

Relaciona una cantidad expresada en cualquier sistema de enumeración posicional con la misma cantidad expresada en el sistema decimal. El valor total del numero será la suma de cada digito multiplicado por la potencia de la base correspondiente a la posición q ocupe el número.

-i: posición respecto a la coma

-d: número de dígitos a la derecha de la coma

-n: número de dígitos a la izquierda de la coma

digito: cada uno de los q componen el número

-base: base de sistema de enumeración

La representación generalizada de un número en base b es la siguiente:

Elementos auxiliares

Lapiz óptico: Permite marcar un punto de la pantalla de TRC. La señal determina la posición del lápiz en la pantalla.

1. **Qué es un modem? Para qué sirve? Por qué se utiliza? Cuáles son los parámetros de un modem?**

Es un dispositivo interno o externo utilizado para la comunicación entre computadoras a través de líneas telefónicas. El módem convierte las señales digitales en analógicas (modularización), y las envía por línea de teléfono a las que deben estar conectadas el emisor y el receptor. Cuando la señal llega a su destino, otro modem se encarga de reconstruir la señal digital primitiva (demodularización), de cuyo proceso se encarga la computadora receptora. A demás un modem está programado para ser tolerantes a errores de transmisión.

1. **Describa las característica de diferentes técnicas de impresión q puede encontrar como periférico de una computadora. Ventajas y desventajas de usar uno sobre otros**

Su objetivo es producir textos. Sin embargo algunos tipos tiene prestaciones gráficas . Según sus características se pueden calificar por:

La forma de imprimir:

-De caracteres: imprime carácter por carácter de manera unidireccional o bidireccionalmente, por lo q son muy lentas.

-De línea: imprime en una línea de caracteres simultáneamente, por lo q son muy rápidas.

-De página: imprime toda una página en una sola pasada, aunque internamente imprime línea por línea.

Mecanismo de impresión:

-De impacto: son aquellas q imprimen mediante impacto del cabezal sobre la hoja. Matriz de punto: es un tipo de impresora muy lenta, q utiliza un conjunto de agujas q golpean una cinta entintada sobre el papel , imprimiendo punto a punto; Margarita , bola , cilindro , etc; Cinta; Tambor.

-Térmicas: son aquellas q imprimen mediante la transferencia de calor al cabezal, a través de una matriz de pequeñas resistencia, a las q al pasar corriente eléctrica por ellas, se calientan formando los puntos. Estas impresoras pueden ser de carácter (las líneas se imprimen con un cabezal móvil) o de línea (contiene tantas cabezas como caracteres a imprimir por línea, son mas rapidos).

-Chorro a tinta: son aquellas q imprimen mediante la carga de gotas de tinta , por medio de electricidad estatica. El carácter se forma con la tinta q cae al papel según la carga estatica q posea la misma. Cuando finaliza, las gotas q sobran se desvían hacia un camino de retorno al cartucho. Son modelos bidireccionales y muy utilizados por su velocidad y gran calidad de impresión.

-Laser: son aquellas q imprimen mediante la radiación de un laser sobre una superficie con propiedades electroestáticas, desde un tambor q tiene la imagen impregnada en toner. Poseen un sistema similar a las fotocopiadoras, con una elevada velocidad y calidad de impresión.

-Ink-jet: pocos fabricantes. Se emite un chorro de gotas q en su recorrido es desviado por electrodos. El carácter se forma con la tinta q incide en el papel. Son bidireccionales.

1. **Describa los componentes q definen el tiempo de acceso de un disco magnético. Como podría calcular el tiempo de acceso promedio? Q elementos sugiere para disminuir el tiempo de acceso promedio?**

Son sistemas de almacenamiento. Está constituido por una superficie metálica o plástica recubierta por una capa de sustancia magnética. El plato puede ser de plástico flexible o rígido.

La lectura/escritura se hace mediante una cabeza. Esta suele ser de tipo cerámico aunque antes eran metálicas. Esta insertada en un extremo de un brazo mecánico. El brazo sitúa rápidamente la cabeza encima de la pista correspondiente y espera q el sector se posicione.

Tiempo de seek (búsqueda) : mover la pista correcta

Tiemo de latencia: esperar q el sector “pase” por debajo de la cabeza

Tiempo de acceso: t.seek + t.latencia

Tiempo total: tiempo de acceso + tiempo de transferencia de datos

Disminuir tiempo: aumentando el número de bits cm de pista (densidad lineal) también aumentando la velocidad de giro

1. **Describe las diferencias entre una computadora teclado y monitor como periféricos y una q posee un equipo denominado “terminal” como periférico. Puede considerar el tipo de comunicación entre CPU y el periférico, la cantidad y ubicación de la memoria de video , cantidad y tipo de puertas de entrada y salida, etc.**

-periféricos específicos:

Terminales: permiten al usuario interactuar directamente con el computador. Se componen de 2 elementos, teclado y monitor ( a veces otros auxiliares )

-teclado: se utiliza para introducir ordenes e información al computador. Cuando una tecla es pulsada se realiza la conversión de su posición, en una matriz de contactos, a un código alfanumérico. Seguidamente se envía el código al computador .

-monitor: Es un dispositivo de salida q mediante una interfaz, muestra los resultados del procesamiento de una computadora. El más definido es el TRC (tubo) q puede ser monocromático o de color. Existen tmb de cristal, líquido o plasma. su construcción comprende de un tubo de vacío con un alargamiento en su vértice. En la parte más estrecha q produce un haz de electrones. En el cuello lleva dos bobinas q permiten modificar la trayectoria hacia la base del tubo. Esta base se ilumina al recibir el impacto del haz. Las letras o figuras se producen modelando la intensidad de haz

Modo entrelazado:

Un monitor asi es cuando los rayos de electrones no se dibujan en forma lineal, sino entrelazada y cuando llega al final de la pantalla, regresa arriba para llenar las líneas sin refrescar.

Un monitor entrelazado con refresco de 100HZ refresca una línea 50 veces por seg.

Un monitor no entrelazado dibuja todas las líneas en cada pasado, resultando una imagen mas nítida.

Velocidad de refresco: es medida en Hertz y representa el nº de cuadreos mostrados en pantalla por segundo. Si son pocos el ojo lo nota. La velocidad aceptada es de 70 hz para arriba.

Clasificación de monitores:

Por colores que muestran: Monocromáticos (información en blanco y negro, en ambar o en verde); a color (información con las combinaciones del rojo, azul y verde).

Por como representan la inf: De caracteres (poseen 2 memorias para representar únicamente caracteres: RAM para guardar la ind de cada carácter con su atributo ROM para guardar los patrones de cada carácter almacenado); graficas (aquellos que utilizan un trazado de líneas para mostrar la información. Poseen una ROM donde se almacena la intensidad, el color y otros atributos para cada pixel representado).

Tamaño de pantalla: en pulgadas.

Resolución de pantalla:

-CGA= 16 colores

-VGA= 16 colores (640x 480 de resolución)

256 colores (320x 200 de resolución)

-SVGA= 16,7 millones de colores

**Perifericos**

Son las unidades o disp. A través de los cuales el computador se comunica con el mudo exterior. También a los sistemas que almacenan o archivan la información (funcionando como mem. Auxiliar).

La memoria auxiliar trata de suplir las diferencias de la mem principal (su baja capacidad por ej).

Los periféricos están constituidos por unidad (es) de entrada, unidad (es) de salida, unidad (es) de memoria auxiliar.

Los disp. De E/S transforman la información externa en señales codificadas, permitiendo su transmicion, detección, interpretación, procesamiento y almacenamiento de forma automática.

Los disp. De E transforman la inf externa según alguno de los códigos de E/S.

En un disp. De S se efectúa el proceso inverso.

Soporte de información: se entiende aquellos medios físicos sobre los que va la información.

Maquina auxiliar: no esta conectada físicamente al computador, sirven para preparar o ayudar en la confeccion o utilización de información que se da a, o produce el computador.

Conexión de periféricos al computador

-Las unidades funcionales se comunican por hilos llamados canales o buses de comunicación.

-El bus del sistema recorre toda la placa base y los periféricos se interconectan (directo o por interfaces).

-Las interfaces son para adaptar las características de los perif a las del bus del sistema.

-Controlan el funcionamiento de los perif y la actividad de E/S.

-3 objetivos: conversión de datos: adaptar la representación de datos del bus del sist a los datos del perif; sincronización: reegula el trafico de inf para que no se den problemas o perdidas; selección de dispositivos: identifican la dirección del periférico que debe intervenir en trafico de datos.

-Ventajas: independizar a los fabricantes de periféricos y placas controladoras, establecer normas para la comunicación entre periféricos y procesadores.

-Estas placas de interfaz muchas veces tienen su propio procesador, memoria y driver que les permite manejar una variedad de periféricos similares.

-Regulan las operaciones de E/S con 3 tecnicas:

E/S programable: los datos se intercambian entre cpu y modulo de E/S. la CPU ejecuta el programa , envía una orden al modulo de E/S y debe esperar que concluya. La desventaja es el desaprovechamiento del tiempo de la CPU.

E/S mediante interrupciones: la CPU proporciona la orden y continua trabajando con otras instrucciones hasta que es interrumpida por el modulo de E/S. La CPU sigue siendo un puente.

DMA: No se utiliza la CPU. La mem y el módulo de E/S intercambian los datos a través del DMA (imita a la CPU). El DMA utiliza el bus del sist solo cuando la CPU no lo utiliza (robo de ciclo). Cuando la CPU desea leer o escribir datos en un dispositivo de E/S envía una orden al DMA con: si se solicita lectura o escritura; la dirección del isp.. de E/S; la posición inicial; el nº de palabras. Asi la CPU solo interviene al principio y al final de la transformación.

Tipos de interfaces:

-Paralelas: aquellas donde existen varias líneas que conectan al modulo de E/S con el perif y transfieren varios bits simultáneamente a través del bus de datos. Se utilizan para dispositivos de alta velocidad como discos rigidos.

-Seriales: aquellas donde solo existe una línea para transferir los datos. Transfieren los bits 1 a 1. Se utilizan para impresoras y teclados.

Caracteristicas generales:

Un perif suele estar formado por 2 partes: mecánica (motores, electro imanes, etc); electrónica (circuitos de la interfaz).

Cuando un periférico actua sin intervención del computador central se dice que trabaja offline.

Fiabilidad: probabilidad de que se produzca un error en la E/S.

Duracion: Permanencia sin alteración de los datos.

Densidad: cantidad de datos contenidos (bits o caracteres).

Reutilización: permitir (o no) guardar nueva información sobre datos obsoletos.

Tipo de acceso: secuencial o directo.

Tiempo de acceso: tiempo que necesita un dispositivo para leer/ grabar un dato.

Velocidad de transferencia: cantidad de información que el dispositivo es capaz de leer/ grabar por unidad de tiempo.

Transportabilidad: si puede trasladarse de una unidad periférica a otra.

Ergonomía: cuando su diseño físico externo se adapta al usuario (al hombre le resulta comodo usarlo).

Clasificacion: Por tipo de periférico: unidades de entrada, de salida, de almacenamiento masivo y de E/S mixtas. Por distancia: Locales: cerca del computador; Remotos: lejos, la conexión hay que realizarla a través de líneas especiales.

**CD-ROM**

 Basado en CD para audio. Compuesto por policarbonato revestido con capa altamente reflectiva, usualmente aluminio. La lectura es por láser reflejado. El acceso es dificultoso, hay que mover cabeza lectora a una posición cercana, establecer la velocidad correcta. Leer la identificación (dirección). Ajustar a la posición requerida. 650mb capacidad. Las ventajas son: fácil para producción en masa, Removible, Robusto. Las desventajas son: Lento, Solo lectura.

-Audio: Velocidad lineal constante: 1,2 m/seg. Velocidad angular variable: 200 a 530 rpm. Velocidad de reproducción: 75 sectores/1 segundo

-CD-Recordable: WORM, Compatible -CD-RW: Borrable, Compatible, Costo en disminución.

**DVD**

Digital Video Disk: dispositivo para films: Sólo películas

Digital Versatile Disk: dispositivo para computadoras: Puede leer disco de computadora y disco de video

Multi-capa: Simple lado, simple capa (4,7GB). Simple lado, doble capa (8,5GB). Doble lado, doble capa (17GB).

Capacidad muy alta

Toda una película: compresión MPEG

**Cinta Magnética**

Acceso en Serie. Lento. Muy económica. Backup y archivo.

Biestables/ circuito lógico

**Multiplexores:** conecta varias entradas con una única salida. En un momento, se selecciona una entrada para que pase a una salida. Se usan en circuitos digitales para controlar el flujo de señales y datos. Ej: carga del contador del programa. Como el PC tiene varios bits, se utilizan varios multiolexores, n x bit. El valor a cargar en el PC se puede venir de 1 o varias fuentes:

Contador binario: si el PC se va a incrementar para la sig instrucción.

Reg de instrucción: se acaba de ejecutar una instrucción de salto usando direccionamiento directo.

Salida de la ALU: si la instrucción de salto especifica la dirección usando modo de desplazamiento.

**El nivel de logica digital :**

- un circuito digital es en el que están presentes dos valores lógicos

- compuertas lógicas son dispositivos electrónicos que pueden realizar distintas funciones lógicas con los valores lógicos en su entrada generando un valor lógicos de salida

- las computers basicas son: AND, OR, NOT, NAND, NOR Y XOR

F = A.B' + A'.B = XOR

1. **Qué es un j-k? describa las características del funcionamiento con tabla del comportamiento y gráfica del circuito lógico.**

Es un circuito secuencial, implica q su salida depende de la entrada actual y del estado actual del circuito. Proporciona memoria o información de estado. Tiene 2 entradas y todas las combinaciones de los valores de entrada son válidos.

Usos: contadores.

1. **Qué es un biestable? Y para q se puede utilizar? Esquematice un sumador de 4 bits**

Es un circuito secuencial q está en 1 de 2 estados posibles, en ausencia de entrada, recordando el ultimo estado. Puede funcionar como una memoria de un bit y tiene 2 salidas complementarias Q y ~Q. Se puede utilizar para registros y contadores.

-Biestable S-R : el circuito consta de 2 entradas S (set) y R (reset) y dos salidas Q y Q~ , y consiste en dos puertas NOR conectadas en retroalimentación. Supongamos que S y R valen cero, y que Q es cero. Las entradas a la puerta NOR inferior son Q=0 y S=0 , entonces Q~=1 entonces Q=0 . por lo tanto, el circuito permanece internamente consistente y estable mientras R=S=0. Entonces podemos tomar a Q como el “valor” del bit. Las entradas S y R sirven para escribir los valores 1 y 0. El S-R presenta error cuando la entrada S=R=1 .estas entradas no esta permitidas, ya q produciría una salida inconsistente: Q= Q~=0

-Biestable S-R sincronico: es similiar al biestable S-R con la diferencia de q las entradas S y R se aplican a las entradas de las puertas solo durante el pulso del reloj. Se añaden 2 puertas NAND. El clk determina cuando se producen eventos.

-Biestable D : el biestable D permite solucionar el problema de los biestables S-R usando un inversor, y garantizando q las entradas q no son del reloj de las puertas sean opuestas. El biestable D se denomina a veces como el biestable de datos , pq almacena un bit de datos. La salida del biestable D es siempre igual al valor mas reciente aplicado a la entrada. Por lo tanto, recuerda y produce la ultima entrada. Tmb se llama biestable de retardo, pq retrasa un 0 o un 1 aplicado a la entrada durante un pulso de reloj.

-Biestable J-K : en el biestable J-K todos los valores de entrada son validos. La entrada J sola realiza la función de puesta a 1, causando q la salida sea 1; la entrada K sola, realiza la función de puesta a 0, provocando q la salida sea 0. Cuando J y K son 1, la función realizada se denomina función de conmutación: la salida se invierte. Si Q vale 1 y se aplica a 1 a J y K entonces Q se hace 0.

-Biestable T: Dispositivo de almacenamiento temporal de 2 estados (alto y bajo). Cambia de estado cada vez que la entrada de sincronismo o de reloj se dispara mientras la entrada T está a nivel alto. Si la entrada T está a nivel bajo, el biestable retiene el nivel previo. Puede obtenerse al unir las entradas de control de un [biestable JK](https://es.wikipedia.org/wiki/Biestable" \l "Biestable_JK" \o "Biestable), unión que se corresponde a la entrada T.

1. **Cuáles son las diferencias entre un flip flop tipo S-R y uno j-k? describa con q elementos se puede construir un registro capaz de almacenar 4 bits utilizando un flip-flop D . Grafique el circuitos de 4 bits q describe.**

El flip flop S-R es un circuito secuencial. Tiene 2 entrada S (set) R (reset) y 2 salidas Q y Q (negado). Consiste en 2 puertas NOR conectadas en retroalimentación. Lo q hace al flip flop J-K es evolucionar mejor a los problemas del S-R, q son la existencia de un estado prohibido y el otro es q cuando el estado de salida previo se mantiene no conocemos la salida esperada. Ambos se resuelven añadiendo una nueva retroalimentación (J-K) además se suele añadir una señal de reloj.

1. **Describa las diferencias entre un circuito combinacional y uno secuencial. Demuestre con ejemplos, por qué un flip flop S-R no debe recibir valor de entrada para R=S=1.?**

Un circuito combinacional se implementa intercambiando puertas, mientras q uno secuencial se implementa con circuitos combinacionales. Los secuenciales necesitan reloj y los combicionales no. La salida en un combinacional es única y es función solo de los valores de entrada en ese instante, mientras q la salida en un secuencial son 2 (complementarias) y dependen no solo de los valores de entrada sino también del estado actual del circuito; tienen la característica de “almacenar” valores lógicos internamente. Respecto del instante en que pueden

cambiar dichas salidas, pueden ser: Asincrónicos (cuando en la entrada se establece una combinación, las salidas cambiarán) y Sincrónicos (la presencia de una entrada especial, determina “cuando” cambian las salidas acorde a las entradas).

1. **Qué es una puerta lógica? Describa el método para la obtención del circuito combinatorio de una tabla de verdad con cuatro variables de entrada q produce una salida “1” solo cuando una de las 4 entradas está en 1.**

Una puerta lógica es un circuito electrónico digital q produce como señal de salida una operación booleana sencilla. son el bloque fundamental de todos los circuitos lógicos digitales. Una puerta puede tener varias entradas y una salida. Las puertas son AND, OR, NOT, NAND Y NOR. Cuando los valores de entrada cambian, la señal de salida correcta aparece casi instantáneamente. La forma de representar una puerta son mediante símbolo (convenio grafico conocido universalmente), tabla de verdad (enumera el valor de una operación para cada combinación posible) o función lógica algebraica (es la ecuacion).

F=A.B.C.D+A.B.C.D+A.B.C.D+A.B.C.D

1. **Qué características determinan q un circuito lógico combinacional? Describa un circuito lógico correspondiente a un sumador completo, realice un gráfico correspondiente. Describa el método para implementar el circuito de la función booleana f de 3 variables (a, b, c) cuya tabla de verdad posee valor 1 cuando solo 2 de sus variables son 1.**

Las características q definen q un circuitos sea combinacional es q la salida es función solo depende de las entradas en un instante dado. Los valores pasados de entrada no influyen en la salida. Carecen de memoria y retroalimentación. Si cambia la entrada cambia la salida. Puede definirse de 3 formas: tabla de verdad (para cada una de las 2n (n: entrada) se enumera el valor binario de c/u de las señales de salida), ecuación booleana (cada señal de salida se representa como una función booleana) y símbolo gráfico (describe la org de las instrucciones entre puertas). En la suma de enteros de 1 bit tenemos 2 resultados: la suma y el acarreo. Un circuito q realiza esta suma se lo reconoce como “medio sumador”. Sin embargo, este no sirve para realizar la suma de bits internos a una palabra ya q no maneja el acarreo de la posición a su derecha. Para ello, se utiliza un “sumador completo”, q se construye con 2 medios sumadores. En un sumador completo, las 2 entradas se conectan a una compuerta XOR y a una compuerta AND. La salida de la compuerta XOR se conecta a otra compuerta XOR, junto con el acarreo de entrada y a otra compuerta AND, tmb con el acarreo de entrada. La salida de la segunda compuerta XOR es el resultado de la suma. Las 2 salidas de ambas compuertas AND, se conectan a una compuerta OR, cuya salida será el acarreo de salida.