

SISTEMAS DE PROCESAMIENTO DE DATOS

Unidad 1 – Estructura de una computadora

Contenido

Unidad 1. Estructura de una computadora	1
Historia de la computación	1
Primera generación de computadoras	4
Segunda generación de computadoras	5
Tercera generación de computadoras.....	5
Cuarta generación de computadoras	6
¿Qué hay luego de la cuarta generación de computadoras?	6
Componentes Fundamentales del Hardware de una Computadora	7
1. UCP (Unidad Central de Procesamiento)	7
2. Memoria principal (Memoria principal, interna o central).....	9
3. Buses	11
4. Unidades o módulos de Entrada / Salida (Canales)	11
5. Periféricos	12
Síntesis sobre el funcionamiento básico de una computadora	13
Arquitectura de computadoras.....	14
Problemática en la arquitectura de computadoras	14
Arquitectura clásica de computadoras o Von Neumann	15
Arquitectura moderna o Harvard	17
Arquitecturas RISC y CISC.....	18
ARQUITECTURA CISC.....	18
ARQUITECTURA RISC.....	19

Unidad 1. Estructura de una computadora

Desde los inicios de la informática, numerosas tecnologías fueron presentándose y con ella, nuevas arquitecturas de computadoras y sistemas informáticos en general. De todas maneras, aun hoy seguimos utilizando los principios básicos formulados hace más de 60 años. En esta unidad realizaremos un breve resumen de aquellos principios, de las arquitecturas básicas y los componentes que la conforman, para eso será necesario conocer parte de la historia que dio origen a ello.

Historia de la computación

La historia de la computación no tiene una fecha exacta de inicio. Generalmente se suele tomar como inicio la creación de algún dispositivo, una tecnología, etc. De todas maneras, en muchos casos se toma como inicio la creación del ábaco hace 4000 años. Luego del ábaco, el siguiente aporte fue hecho por Blaise Pascal en 1641, que introdujo la Pascalina. Esta era una máquina que podía sumar y restar, aunque un año más tarde, Gottfried Leibnitz realizó mejoras a la Pascalina para que esta además pueda multiplicar, dividir y hallar la raíz cuadrada de un número.

Podríamos decir que la primera vez que se pensó en un equipo con características más similares a lo que hoy vemos en las computadoras fue recién 200 años después de la Pascalina, en 1833. Charles Babbage y Lady Augusta Ada Byron comenzaron a pensar en la máquina analítica. La idea era que esta máquina analítica fuera capaz de procesar información de manera automática, sin la necesidad de que una persona intervenga. Esta máquina terminó siendo solo un modelo teórico ya que en ese momento no existía la tecnología para fabricarla y no existía financiación para el desarrollo de esas tecnologías (en esa época, la mayoría del financiamiento estaba dedicado a asuntos bélicos).



Fig. 1. Máquina tabuladora utilizada en los primeros censos de Estados Unidos.

En Estados Unidos, cada 10 años se realiza un censo entre sus habitantes. El año 1880 fue un año en que se realizó el censo, pero con la tecnología primitiva de la época el mismo duró 8 años. Dada la gran cantidad de tiempo que demandó la realización del censo, el gobierno decidió realizar una competición que premiaba a aquel que desarrollara algún método para agilizar el proceso. Allí es donde entra en escena Herman Hollerith, que desarrolló una máquina, la “máquina tabuladora”, que “automatizaba” el proceso de recuento gracias a la utilización de tarjetas perforadas para representar la información de cada persona. Hollerith, no se detuvo con la creación de aquella máquina y fundó International Business Machines, mejor conocida como

IBM y hoy en día una de las empresas más reconocidas en la industria.

Resumen sobre ANTECEDENTES HISTÓRICOS de las computadoras (“prehistoria”)

Mucho antes de la aparición de los primeros computadores electrónicos hubo calculadores electromecánicos, y mucho antes que éstos existieron dispositivos mecánicos.

La prehistoria: utilizaban los dedos de las manos y los huesos para la contabilización de objetos de usos cotidianos (animales, utensilios, personas, etc.).

3000 a.C. – Abaco: primer maquina digital, 1° elemento automático de cálculo creado por el hombre. Constituido por bolitas ensartadas en un cordón que a su vez se fija en un soporte de madera. Manejaba el sistema de unidad, decena y centena.

1614 – Aparece el Logaritmo: tabla de logaritmos. Creado por un escocés llamado Neper. Para multiplicar se suman los logaritmos de los n° que se han de multiplicar, para dividir se restan, y para calcular potencias se multiplican.

1650 – Regla de Cálculo: se utilizan escalas que se gradúan de acuerdo a los logaritmos de las cantidades que se calculan. Se realiza en pocos pasos.

1650 – Pascal: era un francés y crea la **Pascalina**, primer calculadora mecánica que utilizaba una serie de ruedas (engranajes) de 10 dientes en las que cada uno de los dientes representaba un dígito del **0 al 9**. Manejaba la adición y sustracción. No puede guardar el resultado.

1671 – Leibnitz: era un alemán y perfecciono los **principios de Pascal** para realizar la **multiplicación y división mediante la adición y sustracción repetidas**.

1800 – Jacquard: Otro francés que **creó las Tarjetas Perforadas**, usa tarjetas perforadas para manejar agujas de tejer en telares mecánicos. **Un conjunto de tarjetas constituían un programa**.

1820/1835 - Babbage: Un inglés, al que apodaban “el loco Babbage”, crea la **Maquina de Diferencias (Maq. Analítica)**, **máquina diferencial para el cálculo de polinomios**; de programa exterior, que no forma parte de la máquina. El operador debía darle los datos y la operación a realizar y la máquina trabajaba automáticamente, sin requerir su intervención constante. Teóricamente funcionaba, no se concretó por la época, por la falta de tecnología.

Esta máquina poseía características de un ordenador moderno:

- Dispositivos de entrada para cargar los datos por medio de tarjetas perforadas
- Memoria para almacenar los datos introducidos, las instrucciones y los resultados de las operaciones.
- Unidad de control y unidad aritmético-lógica.
- Dispositivos de salida para enviar al exterior los resultados.

Ada Augusta Byron King (1815-1852) trabajó junto a Babbage y fue la primera persona en describir un lenguaje de programación de carácter general. Se la reconoce como la madre de la programación informática.

1847/1854 – George Boole: (Inglés) desarrolla el **Algebra Booleana** y aporta significativamente a la representación de circuitos lógicos.

1890 – Hollerith: (censista de EEUU): 1° tarjeta perforada para almacenar datos. **Máquina para censos**, utiliza tarjetas perforadas codificadas.

1907- Lee De Forest: inventa la **válvula electrónica triodo de vacío**.

1912 -1954 - Alan Turing: creó en 1936 una máquina que trabajaba con **código binario**. Por eso algunos lo consideran el **inventor de las computadoras digitales**. Influyó en la formalización de los conceptos “algoritmo” y “computación”

1939/1944 – Howard Aitken: Otro estadounidense desarrolla la **MARK I**, modifican la máquina analítica y crean el **primer computador electromecánico** (con los principios de Babbage).

1945 – Mauchly y Eckert: Dos estadounidenses desarrollan en la Universidad de Pennsylvania la **ENIAC** (*Electronic Numeric Integrator and Calculator*), **1° máquina de cálculo totalmente electrónica** (pesaba 30 toneladas).



Fig. 2. Tubo de vacío, componente que hizo funcionar las computadoras en la primera generación.

1945 – John Von Neumann: un Húngaro desarrolla sus principios para una computadora.

1947 – Primer computadora con los principios de JVN: EDVAC (*Electronic Discrete Variable Automatic Computer*, desarrollada por Mauchly, Eckert y Von Neumann).

Primera generación de computadoras

Luego de la máquina tabuladora de Hollerith, la historia de la computación se empieza a clasificar en generaciones. El inicio de cada generación se encuentra marcada por el descubrimiento y uso de un determinado componente o tecnología. **Particularmente la primera generación de computadoras se encuentra marcada por el uso de los tubos o válvulas de vacío.** Aquel componente electrónico era la unidad básica para el funcionamiento de los equipos. Los tubos de vacío se caracterizan por generar grandes cantidades de energía térmica (calor), por lo que su vida útil era demasiado corta. Esto hacía que los equipos que los utilizaban necesitaban un ambiente fresco y un mantenimiento constante para funcionar correctamente. Además de la problemática de la temperatura, los tubos de vacío eran componentes relativamente grandes, por lo que, además, ocupaban grandes volúmenes de espacio.

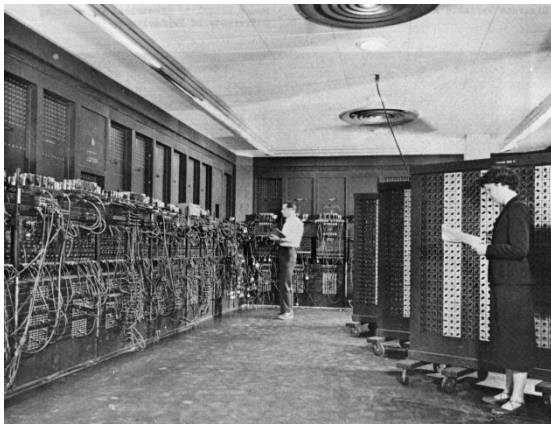


Fig. 3. Imagen tomada de la ENIAC. Nótese los cables que representaban el programa almacenado.

En 1945, el matemático **John Von Neumann** realiza el que quizás termine siendo el aporte más significativo en la historia de la computación: **la idea de la computadora que se manejaba con instrucciones almacenadas en una memoria.** Hasta el momento, los equipos recibían las instrucciones directamente de un usuario, pero no existía un medio de almacenamiento en el que se pueda guardar un proceso y la idea ni siquiera había surgido todavía. Este concepto ideado por Von Neumann llevó al desarrollo de lo que se considera como la primera computadora, la **ENIAC**. Como la mayoría de los desarrollos en computación, la iniciativa de la creación de este prototipo surge de las fuerzas armadas de Estados Unidos. La ENIAC utilizaba 18.000 válvulas de vacío, medía 30 metros de longitud por una profundidad de un metro. Funcionaba con **programas almacenados de manera cableada y utilizaba el sistema numérico decimal para representar la información.**

Luego de la creación de ENIAC surge **EDVAC**. Este equipo realiza un cambio que aún hoy en día sigue estando presente: **utiliza el sistema de numeración binario en vez del decimal**. Este cambio además permitió que **los programas se almacenen en primitivas memorias, en vez de almacenarse de forma cableada como se hacía en la ENIAC**.

A partir de los enunciados de Von Neumann, todas las computadoras fueron desarrolladas en base a esos principios.

Segunda generación de computadoras

La segunda generación de computadoras comienza en el año 1953. Esta generación está marcada por una **considerable reducción en el tamaño de las computadoras**. Esta reducción fue posible gracias a la **invención del transistor**. Este pequeño componente electrónico surge en 1948 y será el reemplazo de la válvula de vacío. Además de ser **más pequeño**, el transistor tenía un menor consumo energético, por lo que también **generaba menos calor**. Esto hacía que sea más confiable y su vida útil sea más larga. Además de todas estas ventajas que proporcionaba el transistor, se sumaba el hecho de que el **procesamiento era mucho más rápido**.

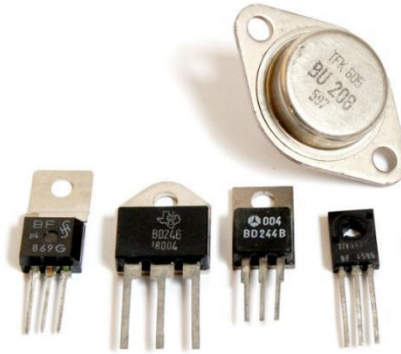


Fig. 4. Distintos tipos de transistores. Con el avance de la tecnología, los mismos se fueron haciendo cada vez más pequeños y fueron capaces de controlar distintos tipos de flujos eléctricos.

Esta generación concluye en 1962. Durante los nueve años que duró, los procesos de fabricación de los transistores fueron evolucionando, junto con la tecnología con la que se los fabricaba, haciendo que cada vez sean más pequeños, y más rápidos.

Cabe destacar también que durante la segunda generación de computadoras surge otro de los elementos fundamentales de la computación: Los **lenguajes de programación de alto nivel**¹. Este componente no supone un componente físico, más bien es parte del software de la computadora, la parte lógica de la misma. Algunos de los lenguajes de programación que surgieron en esta época son FORTRAN, COBOL, ALGOL, etc.

Tercera generación de computadoras

La tercera generación de computadoras abarca desde 1963 hasta 1971. Como se mencionó anteriormente, en el transcurso de los nueve años que abarcó la segunda generación de computadoras, la tecnología de fabricación fue haciendo que los transistores sean cada vez más pequeños. Esto puede traducirse que, cuanto más pequeños, **mayor cantidad de transistores podían**



Fig. 5. Circuito integrado NE555 utilizado ampliamente en distintos tipos de circuitos electrónicos.

¹ Cuando hablamos de lenguajes de programación, el “nivel” se refiere al nivel de abstracción del lenguaje con el hardware. Los lenguajes de programación de alto nivel son aquellos que tienen un alto nivel de abstracción con el hardware, por lo que no es necesario conocer la arquitectura del mismo para poder programar. En los lenguajes de programación de bajo nivel, es necesario conocer el hardware para poder desarrollar un programa.

ubicarse en un mismo espacio. Esto dio lugar a la invención de un nuevo componente: El circuito integrado.

El circuito integrado será la pieza fundamental de fabricación de las computadoras durante la tercera generación de computadoras. Este componente consiste de una pastilla, principalmente de silicio, que en su interior contiene una gran cantidad de componentes electrónicos (transistores, capacitores, resistencias, etc.). Esta integración de varios componentes básicos en un único componente hizo que sea posible la fabricación de computadoras mucho más pequeñas que sus antecesoras de las anteriores generaciones. Durante esta generación, y gracias al circuito integrado, surgen las primeras computadoras personales.

Desde el punto de vista del software, la tercera generación marca el nacimiento de los primeros sistemas operativos.

Cuarta generación de computadoras

A diferencia de las primeras tres generaciones de computadoras, la cuarta no está marcada por el descubrimiento o el uso de un nuevo componente electrónico. La cuarta generación viene de la mano del gran crecimiento en la densidad de componentes que se incluían en un circuito integrado. Esta densidad de componentes dentro de un circuito integrado se conoce como “escala de integración”.

Durante la cuarta generación surgen los conceptos de LSI y VLSI. LSI, por sus siglas en inglés significa “gran escala de integración” (*Large Scale Integration*) y supone una escala que va de los 500 a los 20.000 transistores en un solo circuito integrado. VLSI o “escala de integración muy grande” (*Very Large Scale Integration*) cubre aquellos circuitos integrados que tienen entre 20.000 y 1.000.000 de transistores. Estas escalas de integración dieron a luz a un componente que hoy en día es fundamental: el microprocesador.

La gran escala de integración, que redujo los tamaños de los equipos, el avance en las técnicas de fabricación, que dieron lugar a una gran baja en los precios de venta de los equipos y el nacimiento de los primeros sistemas operativos orientados al usuario, dieron lugar a las primeras computadoras personales. Entre estas computadoras personales, podemos destacar la fabricación de los primeros equipos de Apple y en cuanto a los sistemas operativos, en esta generación también se produce el nacimiento de Microsoft, y, por lo tanto, los sistemas operativos DOS primero y Windows después.

¿Qué hay luego de la cuarta generación de computadoras?

A partir de la cuarta generación, resulta complicado identificar la finalización y el inicio de nuevas generaciones. Esto sucede a raíz del vertiginoso crecimiento en las tecnologías, las técnicas de fabricación, las escalas de integración, etc. Además, por otro lado, a partir de la cuarta generación, todos los equipos fueron fabricados con base en todos aquellos principios de los que se venía hablando hace varias décadas. No surgió ninguna tecnología que cambie radicalmente la forma de construir o utilizar las computadoras. De todas maneras, existen algunos puntos que muchas veces son considerados como la finalización de la



Fig. 6. Microprocesador intel. Nótese los contactos en la parte inferior del mismo. Debajo de la placa metálica que se encuentra en la parte superior se ubican los millones de transistores que posee.

cuarta generación y el inicio de la quinta, como por ejemplo la invención de los microprocesadores multinúcleo, que permitieron un procesamiento paralelo real, el avance de las tecnologías móviles, o la realidad virtual.

Componentes Fundamentales del Hardware de una Computadora

Hardware: es la parte física de los sistemas de computación. Consta de los medios físicos (los componentes tangibles; el equipamiento material; ejemplos: *plaquetas, circuitos, componentes electrónicos, pantallas, gabinetes, dispositivo, cables, engranajes, etc.*) que permiten llevar a cabo un proceso de datos, conforme lo ordenan las instrucciones de un cierto programa previamente memorizado en un computador.

La computadora es un conjunto de distintos componentes, cada uno llevará a cabo una tarea específica. Si bien existen muchos componentes en una computadora, algunos de ellos son fundamentales para que la misma pueda funcionar. A continuación, se mencionarán aquellos componentes imprescindibles para una computadora, a los cuales volveremos en otras unidades del apunte.

Intentando definir entonces un esquema general de la computadora, establecemos los siguientes bloques constitutivos:

1. Unidad Central de Procesamiento (UC, UAL, registros)
2. Memoria principal, interna o central
3. Unidades o módulos de Entrada / Salida
4. Periféricos (de entrada/salida)
5. Buses

1. UCP (Unidad Central de Procesamiento)

Generalmente en las computadoras actuales la **UCP** (o **CPU**, *Central Processing Unit*) está contenida dentro de un chip **MICROPROCESADOR** (circuito integrado, constituido por un conjunto de transistores encapsulados en una pastilla de silicio). Algunos nombres de microprocesadores: *Intel 4004*, *Intel 8086*, *Intel 80386*, *PowerPC 620*, *Intel Pentium*, *AMD K6-2*, *AMD Athlon*, *Intel Core Haswell*, *Intel Ivy Bridge*. Frecuentemente se lo llama simplemente “**procesador**”.

Sus funciones son:

- **Controlar y supervisar el funcionamiento del Sistema de Computación en base a un programa almacenado en la Memoria Principal.**
- Llevar a cabo la ejecución de las instrucciones que se le dan a la computadora realizando las operaciones aritméticas y lógicas que sean necesarias.
- Controlar el envío y recepción de datos desde las unidades periféricas a la memoria.

La UCP está compuesta por:

- **UC**
- **UAL**
- **Registros**

UC (Unidad de Control)

De algún modo, la UC es quien dirige el funcionamiento del equipo. Tiene a su cargo el secuenciamiento de las acciones necesarias que los diferentes circuitos deben realizar y también tiene a su cuidado el orden de ejecución de las instrucciones de un programa.

Funciones:

- **Obtener de memoria la próxima instrucción a ejecutar**
- **Analiza la instrucción**
- **Establece conexiones eléctricas**
- **Localizar en la memoria principal los datos a operar**
- **Ordenarle al circuito de la Unidad Aritmético Lógica que realice con dichos datos la operación indicada**
- **Guardar el resultado en un registro acumulador o memoria principal**

Consta de un conjunto de circuitos, integrados dentro de la CPU, los cuales cumplen funciones específicas que permiten, por ejemplo, detectar la disponibilidad o no de algún periférico o la disponibilidad de posiciones en la memoria principal.

La *unidad de control* (como dispositivo físico) junto con el *sistema operativo* (como dispositivo lógico), permiten el verdadero control y dirección de todas las acciones que se producen dentro de la computadora. Dispone de un reloj electrónico que genera impulsos a intervalos regulares, marcando lo que se denomina ciclo de máquina, el cual resulta determinante en la velocidad de procesamiento.

UAL (Unidad Aritmético Lógica)

Funciones:

- Realiza operaciones aritméticas o lógicas que le ordena la UC, auxiliadas por registros acumuladores (por ejemplo, en algunos procesadores, el registro acumulador AX) que guardan transitoriamente datos y resultados. Las instrucciones concretas a realizar la envía mediante señales la unidad de control, aunque las ejecuta de manera autónoma. Recoge los datos a operar (operandos) de los registros convenientes, proporcionando los resultados.
- No emite órdenes ni ejecuta instrucciones (en este sentido decimos que es pasiva).

Operaciones aritméticas: + ; - ; / ; *

Operaciones lógicas: AND, OR, NOT (producto (Y), suma (O) y negación lógica, respectivamente)

Registros:

Son circuitos que se encuentran en la UCP y pueden almacenar temporalmente información relacionada con la instrucción en curso de ejecución y con las próximas instrucciones a ejecutar.

2. Memoria principal (Memoria principal, interna o central)

Desde que Von Neumann planteó la idea de que las computadoras ejecutaran programas almacenados, fue necesaria una memoria. Generalmente, los programas que ejecutamos, inicialmente se encuentran almacenados en una memoria no volátil (discos rígidos, pendrives, etc.). Estas memorias no volátiles son relativamente lentas, por lo tanto, si el microprocesador trabaja con los datos directamente en éstas memorias, los tiempos de ejecución serían muy lentos. Es por eso que se agrega una memoria más, del tipo volátil (pierde la información cuando se corta la energía eléctrica), más rápida que las memorias no volátiles y directamente conectada al microprocesador. En esta memoria se almacenarán los programas que se ejecutan en el momento, quitando la necesidad de trabajar directamente con la memoria no volátil. A esta memoria no volátil se la conoce como memoria principal y comúnmente se la llama simplemente “memoria RAM”, por el tipo de tecnología con el que trabaja.

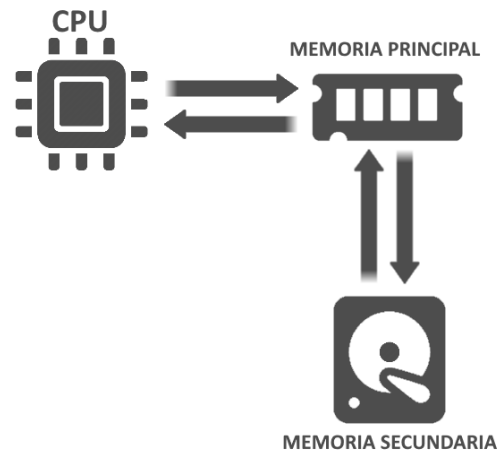


Fig. 7. Esquema de conexiones entre las memorias principal, secundaria y el microprocesador. Las flechas indican el flujo de los datos.

Funciones:

Almacena **instrucciones de programas** que próximamente serán ejecutadas en la UCP y **los datos que ellas ordenan procesar** (operar) así como resultados intermedios y finales de operaciones sobre datos recientemente llevados a cabo en la UCP. El programa que se quiera ejecutar y los datos que deban ser procesados tendrán que estar almacenados en la memoria principal durante el tiempo que sea requerido. Datos y programas comparten la memoria pero se hallan en diferentes zonas

Almacena *bits* (unos y ceros) en *celdas* independientes. Cada celda contiene 1 *byte* (8 bits) de información y se identifica mediante un número binario que constituye su *dirección* de memoria. El mismo indica su posición en ese conjunto.

Para iniciarnos en su estudio podemos decir que la Memoria Principal está constituida por:

MEMORIA RAM (*Random Access Memory*, memoria de acceso al azar: esto hace referencia a que se puede acceder rápidamente a cualquier celda de la memoria en forma directa): Decimos que es utilizada por los usuarios ya que es la parte de la memoria principal que utilizan los programas en el momento de su ejecución. Su almacenamiento, lógicamente, es temporario, o sea, que es una memoria de lectura y escritura (cualquier celda de memoria puede ser leída y escrita cuantas veces sea necesario). También se dice que es volátil, porque pierde la información almacenada cuando se corta el suministro de energía.

MEMORIA ROM (*Read Only Memory*, Memoria de Sólo Lectura: Los datos están almacenados permanentemente, el contenido de esta memoria no es modificable por los usuarios y no es volátil; no obstante también su acceso es *random*): Es utilizada para almacenar los programas provistos por el fabricante de las computadoras y que son indispensables para el arranque y su funcionamiento. Esta memoria se activa en el momento del encendido de la computadora.

3. Buses

El sistema de interconexión de las distintas partes o bloques de la computadora se realiza mediante los “buses”, que consisten en caminos formados por conjuntos de cables o líneas conductoras que permiten la comunicación eléctrica entre esas partes. O sea, los buses constituyen el mecanismo que permite el flujo de datos entre la CPU, la memoria y los módulos de entrada/salida. A través de ellos se propagan las señales eléctricas que son interpretadas como unos y ceros lógicos.

4. Unidades o módulos de Entrada / Salida (Canales)

Son dispositivos encargados de establecer enlaces de comunicación entre la CPU (Unidad Central de Proceso) y los periféricos. Cada módulo de E/S se conecta al bus del sistema o a un conmutador central y controla a uno o más periféricos. Un módulo de E/S no es solamente un conector mecánico para “enchufar” un periférico sino que contiene la lógica necesaria para permitir la comunicación entre el periférico y el bus.

Funciones de los canales:

- a) Determinar la disponibilidad de los periféricos.
- b) Controlar los formatos de E/S de los datos.
- c) Comprobar errores de lectura y/o grabación.
- d) Recuperación de errores de transmisión.

Los canales suelen especializarse según el periférico que tienen a su cargo y se clasifican por su forma de trabajar.

Las **tarjetas de interfaz** o de **expansión** cumplen la función de traducir y transmitir mediante los buses (tanto internos como externos) la información para que pueda ser interpretada tanto por los periféricos como por la C.P.U.

Estas tarjetas de expansión constan de distintos componentes que se relacionan directamente con los distintos tipos de dispositivos periféricos. Estos componentes son un **procesador** para la traducción y transmisión o recepción de los datos que serán procesados, un memoria **ROM** para almacenar el juego de instrucciones correspondientes a la tarjeta, **buses** internos para comunicar a cada uno de sus componentes y el tipo de **puerto** (*por ejemplo, serie* que transmite de a 1 bit por vez y el **paralelo** que transmite de a grupos de bits) que permita conectar mediante los buses externos al **periférico** con el cual esté relacionado. Algunas tarjetas constan además de una memoria RAM propia, como por ejemplo las tarjetas de vídeo.

5. Periféricos

Son aquellos dispositivos que se encargan de entrar datos desde el exterior hacia el computador y dar salida de resultados del computador al exterior. Cada periférico tiene su función definida y ejecuta o envía tareas a la computadora de acuerdo a la función para la que fue diseñado. Se conectan a través de los puertos característicos de las computadoras (*USB, serie, paralelo, etc.*) o mediante las tarjetas de expansión.

En un sentido amplio, los periféricos pueden clasificarse en:

- de **interacción con humanos**: permiten la comunicación con el usuario de la computadora (*monitores, impresoras, mouse, etc.*).
- de **interacción con máquinas**: permiten la comunicación con elementos del equipo (*discos, pendrive, sistema de cintas magnéticas, etc.*).

- de **comunicación**: permiten la comunicación con dispositivos remotos (*modem, placa de red, etc.*).

Otra clasificación más habitual tiene en cuenta las siguientes 3 clases:

- **Entrada**: proporcionan datos y señales de control a la unidad central de procesamiento. O sea, esta clase de periféricos permiten la entrada desde el exterior hacia la memoria principal, la cual es el destino de estos datos e instrucciones. Por ejemplo: *teclado, mouse, webcam, escáner, micrófono, etc.*
- **Salida**: permiten la salida desde la memoria principal hacia el exterior (la memoria principal es el origen de resultados y deben salir al exterior). Por ejemplo: *monitores, impresoras, plotters, parlantes, etc.*
- **E / S**: aquellos que posibilitan el flujo de instrucciones y datos en ambos sentidos (hacia el interior y hacia el exterior según se requiera). Por ejemplo: *modem, unidades de disco rígido, de CD, de DVD, tarjetas de memoria, pendrive, disquetera, impresora multifunción, placa de sonido, etc.*

Paralelamente, se pueden considerar como una clase en particular todos los **periféricos o dispositivos de almacenamiento (memorias externas)**: *unidades de disco rígido, de CD, de DVD, tarjetas de memoria, pendrive, disquetera, etc.*

Síntesis sobre el funcionamiento básico de una computadora

- Datos e instrucciones del programa que los procesará deben llegar a Memoria Principal desde los periféricos. Cada instrucción está codificada mediante una combinación de 0s y 1s (*código*)
- La Unidad de Control localiza en Memoria Principal la instrucción que debe ser ejecutada para que su código llegue a la Unidad Central de Procesos. Una vez allí, la Unidad de Control interpretará (*decodificará*) la instrucción, o sea, determinará qué es lo que ese código ordena realizar. Según lo que especifique el código, se localizarán los datos con los que operará la Unidad Aritmético Lógica y la operación que ésta debe realizar. También se determinará donde guardar el resultado y dónde localizar la próxima instrucción en Memoria Principal.
- La Unidad de Control ordena operaciones de lectura y escritura sobre Memoria Principal o registros de la Unidad Central de Procesos, así como la operación a realizar por la Unidad Aritmético Lógica. A la Unidad Aritmético Lógica van los datos a operar y desde ella se generan

resultados que van a Memoria Principal para luego encaminarse hacia el exterior a través de un periférico.

Arquitectura de computadoras

Actualmente el uso de computadoras está muy extendido en nuestras actividades cotidianas, nos proporcionan mayor rapidez en nuestras tareas, así como la posibilidad de manejar gran cantidad de datos. Así podemos percibir como de un tiempo a esta parte las computadoras ganan terreno en las actividades humanas, su capacidad de cálculo, de procesar datos, generar información e incluso simular procesos las convierten en herramientas indispensables únicamente limitadas por su propia capacidad.

Debemos tener en cuenta que el uso de las computadoras está muy extendido en actividades que requieren el manejo de gran cantidad de datos a una gran velocidad como, por ejemplo, en diagnósticos médicos, investigaciones nucleares y de energía, inteligencia artificial, etc. Es en estos casos, que demandan mayor capacidad y rapidez, cuando se sobrepasan las características de las computadoras actuales. La solución pasa entonces por la construcción de mejores computadoras que logren avances substanciales en su rendimiento.

Para lograr un aumento en el rendimiento se necesita mejorar la arquitectura de las computadoras y desarrollar nuevas técnicas de procesamiento. El concepto de "arquitectura" se refiere a la integración de su estructura física con su estructura lógica. Se utiliza el término "arquitectura" para enfatizar la síntesis de elementos de ingeniería y ciencias exactas con elementos estéticos y de funcionalidad práctica, de la misma manera en que un arquitecto combinará las técnicas y conocimientos de la Ingeniería con la apreciación artística e integración de su obra con su entorno.

Esta "arquitectura de computadoras" abarca las siguientes fases:

- Definición de las necesidades que pretende cubrir la computadora.
- Planificación general de la computadora.
- Diseño del computador y sus componentes.
- Análisis del sistema obtenido.
- Especificación del sistema: características del sistema, de sus componentes y de las instrucciones ejecutables.

Problemática en la arquitectura de computadoras

Al principio, las dos causas principales de la problemática en la arquitectura de computadoras fueron:

- La independencia entre el hardware y el software, y la falta de definición de las funciones de cada uno de ellos. Hasta hace pocos años, los arquitectos de computadoras procedían del campo de la Ingeniería Electrónica, y potenciaban el equipo físico aplicando los constantes avances de la Microelectrónica, sin tener en consideración las prestaciones del sistema lógico.
- El seguimiento a ultranza de la arquitectura de Von Neumann, no diseñada para soportar los nuevos sistemas operativos, lenguajes y aplicaciones.

Hoy en día, estos problemas fueron superados de alguna manera, para dar lugar a otros factores:

- **El factor tiempo.** En el diseño del equipo físico, el tiempo es un factor de gran relevancia. Se cita como ejemplo que cada tres años por el mismo precio y calidad obtenemos el doble del número de transistores contenidos en un chip, frecuencia de trabajo y densidad de discos magnéticos y la cuatruplicidad de la densidad de la memoria RAM. La miniaturización es la constante en la evolución de los dispositivos electrónicos. Este fenómeno de desarrollo acelerado conduce a la ley de Moore (1964), que asegura que el número de transistores que incorpora un circuito integrado se multiplica por cuatro cada aproximadamente año y medio. Se piensa que esta ley, vigente durante 40 años, todavía tiene por lo menos otros 10 o 20 años de vida. Los últimos chips fabricados tienen más de 1.000 millones de transistores en poco más de un centímetro cuadrado.

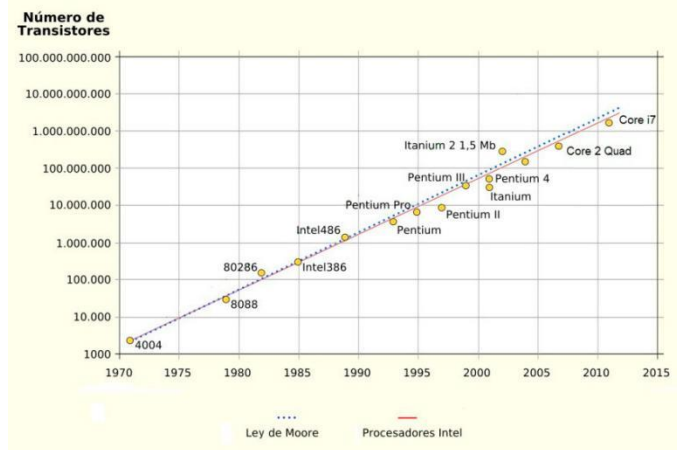


Fig. 8. Gráfico en donde se ve la recta que representa la ley de Moore y la recta que muestra el desarrollo de los microprocesadores desde los años '70 hasta el 2015

- **El alcance de límites difícilmente superables de la tecnología hardware.** La búsqueda de mayor rendimiento, se basará en nuevas arquitecturas que exploten en mayor grado las posibilidades del hardware. Un ejemplo es la paralelización del procesamiento.

Arquitectura clásica de computadoras o Von Neumann

En 1947 el matemático John Von Neumann, uno de los mejores de la historia, ideó una solución para evitar tener que "recablear" la ENIAC. Esto consistía en introducir las operaciones por medio de tarjetas perforadas, por lo que un programa previamente "almacenado" en las tarjetas, se podía mantener latente en memoria para su uso. Este modelo permite que las instrucciones se encuentren residentes en una memoria listas para ser leídas y ejecutadas.

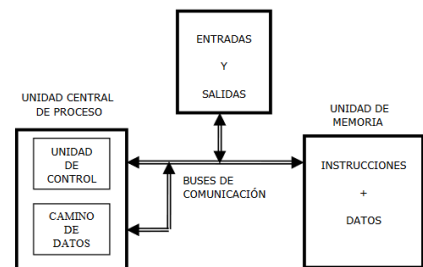


Fig. 9. Esquema representativo de la arquitectura clásica de Von Neumann.

La arquitectura de Von Neumann consta entonces de una **CPU (Unidad Central de proceso)** conectada a una **única memoria** donde se guardan conjuntamente *instrucciones* (programas) y *datos* (con los cuales operan estos programas). A dicha memoria se accede a través de un **sistema de buses único** (*control, direcciones y datos*). Además posee un **módulo de entradas y salidas** para permitir la comunicación de la máquina con los **periféricos** externos que maneja el usuario.

- Los datos y programas se almacenan en una misma memoria de lectura-escritura
- Los contenidos de esta memoria se direccionan indicando su posición sin importar su tipo
- Ejecución en secuencia (salvo que se indique lo contrario)
- Representación Binaria
- Un sistema de interconexión conecta la memoria y unidad de control

Problemas o debilidades de la arquitectura Von Neumann:

- **La limitación de la velocidad de operación a causa del bus único para datos e instrucciones:** se fuerza la alternación entre ciclos de lectura y ejecución: un bus único de instrucciones y datos impide que se puedan superponer ambos procesos, es decir, el bus no deja acceder simultáneamente a unas y otros, lo cual impide superponer ambos tiempos de acceso. Esto se suele llamar efecto cuello de botella del modelo Von Neumann. Dicho de otra manera, el tener un único bus hace que el microprocesador sea más lento en su respuesta, ya que no puede buscar en memoria una nueva instrucción mientras no finalicen las transferencias de datos de la instrucción anterior.
- **La longitud de las instrucciones está limitada por la longitud de los datos** (también por compartir el bus), entonces, para buscar instrucciones complejas, el microprocesador suele tener que realizar varios accesos a memoria. El tamaño de la unidad de datos o instrucciones está fijado por el ancho del bus que comunica la memoria con la CPU. Esto quiere decir que, por ejemplo, un microprocesador de 8 bits con un bus de 8 bits, cuando tenga que manejar datos e instrucciones de una o más unidades de 8 bits (1 byte) de longitud, tendrá que realizar más de un acceso a la memoria.

La arquitectura Von Neumann realiza o emula los siguientes pasos secuencialmente:

- 1) Obtiene la siguiente instrucción desde la memoria en la dirección indicada por el contador de programa y la guarda en el registro de instrucción.
- 2) Aumenta el contador de programa en la longitud de la instrucción para apuntar a la siguiente.
- 3) Decodifica la instrucción mediante la unidad de control. Ésta se encarga de coordinar el resto de componentes del ordenador para realizar una función determinada.
- 4) Se ejecuta la instrucción. Ésta puede cambiar el valor del contador del programa, permitiendo así operaciones repetitivas.
- 5) Regresa al paso N° 1.

Arquitectura moderna o Harvard

Esta arquitectura está diseñada para atacar las debilidades de la arquitectura Von Neumann y consiste en construir un procesador que está unido a **dos tipos de memoria diferentes**, una de datos y otra de instrucciones, por medio de **dos buses independientes**.

La existencia de **ambos buses y de ambas memorias totalmente independientes** permite que la CPU pueda acceder de forma independiente y simultánea a datos y a instrucciones. De esta manera se hace posible que la longitud de los datos y de las instrucciones puede ser distinta, lo cual optimiza la performance en general.

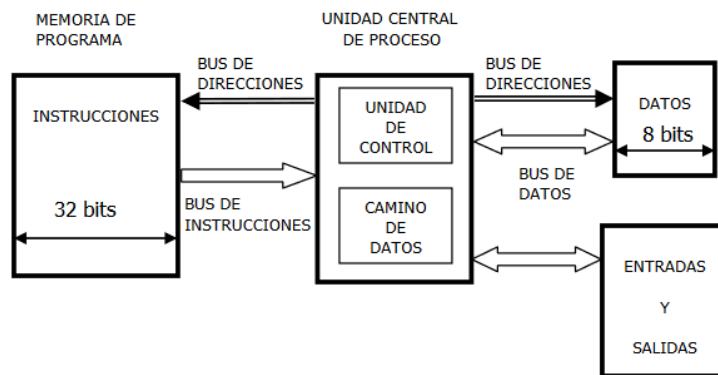


Fig. 10. Esquema representativo de la arquitectura Harvard.

Esta arquitectura es la utilizada entre otros dispositivos, en los *Microcontroladores PIC* (*Peripheral Interface Controller*).

Ventajas de la arquitectura Harvard:

- **El tamaño de las instrucciones no está relacionado con el de los datos**, por lo que puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.
- **La superposición del acceso a datos e instrucciones permite lograr mayor velocidad en las operaciones.**

Una forma de potenciar el aislamiento entre las instrucciones y los datos es la incorporación de memorias caché, que como sucede en los modelos Pentium, una se encarga de guardar los datos que va a precisar la CPU y otra las instrucciones

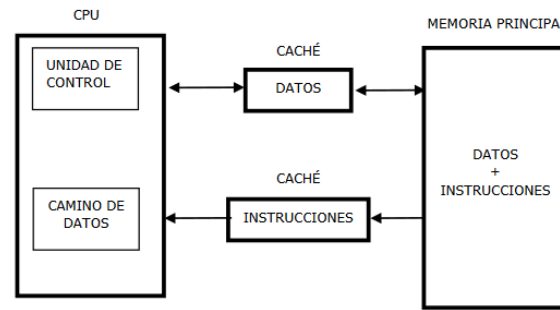


Fig. 11. Esquema representativo de la arquitectura de microprocesadores utilizadas por Intel en los modelos Pentium.

Arquitecturas RISC y CISC

Al encarar el diseño un microprocesador, una de las cuestiones principales es decidir cuál será su juego de instrucciones, por dos razones principales: primero, porque el juego de instrucciones define el diseño físico del conjunto y segundo, porque cualquier operación que deba ejecutarse en el microprocesador deberá poder ser descrita en términos de un lenguaje de estas instrucciones.

En base a esta problemática se consideran dos tipos de diseño de procesadores: **CISC** y **RISC**.

ARQUITECTURA CISC

CISC (*Complex Instruction Set Computer*, Computador con Conjunto de Instrucciones Complejas): usada en muchas de las computadoras personales, por ejemplo, *Motorola 68000*, *Zilog Z80* y toda la familia *Intel x86*, *AMD x8664*. Los microprocesadores CISC tienen un **conjunto de instrucciones** que se caracteriza por ser **muy amplio y permitir operaciones complejas** entre operandos situados en la memoria o en los registros internos, en contraposición a la arquitectura RISC.

Estas operaciones complejas exigen mucho tiempo de CPU para ser ejecutadas.

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

Para realizar una sola instrucción un chip CISC requiere de cuatro a diez ciclos de reloj.

Entre las ventajas de CISC destacan las siguientes:

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

ARQUITECTURA RISC

RISC: (*Reduced Instruction Set Computer*, Computador con Conjunto de Instrucciones Reducidas) es un tipo de diseño de CPU generalmente utilizado en microprocesadores o microcontroladores con las siguientes características fundamentales:

1. **Instrucciones de tamaño fijo y presentadas en un reducido número de formatos.**
2. Sólo las instrucciones de carga y almacenamiento acceden a la memoria de datos.

Además estos procesadores suelen disponer de muchos registros de propósito general.

El objetivo de diseñar máquinas con esta arquitectura es posibilitar la **segmentación** (para que una instrucción empiece a ejecutarse antes de que hayan terminado las anteriores y que haya entonces varias instrucciones procesándose simultáneamente) y el **paralelismo en la ejecución** de instrucciones y reducir los accesos a memoria. Las máquinas RISC protagonizan la tendencia actual de construcción de microprocesadores. *PowerPC*, *DEC Alpha*, *MIPS*, *ARM*, *SPARC* son ejemplos de algunos de ellos.

Para un procesador RISC, el set de instrucciones y el bus de memoria de programa pueden diseñarse de tal manera que todas las instrucciones tengan una sola posición de memoria de programa de longitud.

Entre las ventajas de RISC tenemos las siguientes:

- Cada instrucción puede ser ejecutada en un solo ciclo del CPU
- La CPU trabaja más rápido al utilizar menos ciclos de reloj para ejecutar instrucciones.
- Utiliza un sistema de direcciones no destructivas en RAM. Eso significa que a diferencia de CISC, RISC conserva después de realizar sus operaciones en memoria los dos operandos y su resultado, reduciendo la ejecución de nuevas operaciones.

Aclaración: El tipo de procesador más comúnmente utilizado en equipos de escritorio, el x86, está basado en CISC en lugar de RISC, aunque las versiones más nuevas traducen instrucciones basadas en CISC x86 a instrucciones más simples basadas en RISC para uso interno antes de su ejecución.