

Introducción a Arquitectura de las Computadoras

Dr. Diego Feroldi
feroldi@cifasis-conicet.gov.ar

Arquitectura del Computador 2024
LCC - FCEIA - UNR



Contenido de la presentación

- 1 Definición de computadora
- 2 Definición de Arquitectura del Computador
- 3 Descripción básica de la arquitectura de una computadora
- 4 Breve historia de la computación
- 5 Jerarquía de memoria
- 6 Representación de programas a nivel de máquina

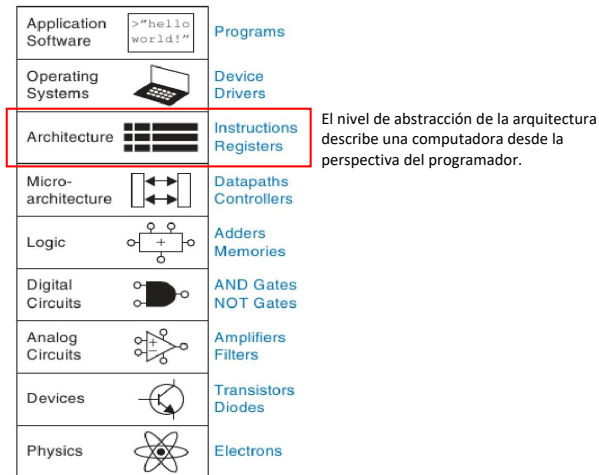
Definición de computadora

“Una computadora digital es una máquina que puede resolver problemas ejecutando las instrucciones que recibe de las personas”^a.

^aAndrew S Tanenbaum. Organización de computadoras: un enfoque estructurado. Pearson educación, 2000.

- Esta definición indica que a pesar de las enormes prestaciones de las computadoras actuales, en su nivel más elemental las computadoras resuelven operaciones muy simples.
- Es importante destacar que los circuitos integrados de una computadora solo pueden reconocer y ejecutar un conjunto muy limitado de instrucciones sencillas.

Niveles de abstracción para una computadora



Harris, David and Harris, Sarah L. Digital Design and Computer Architecture.

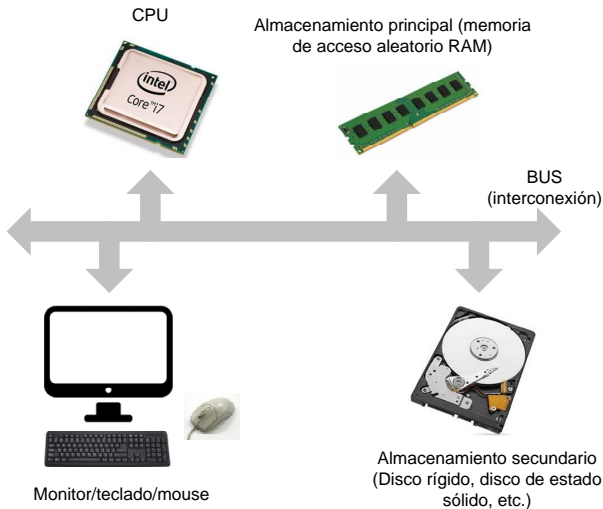
Definición de Arquitectura del Computador

Arquitectura del Computador

Es el diseño y la estructura de los componentes internos de un sistema de computación, que incluye la organización de hardware, el conjunto de instrucciones que el procesador puede ejecutar, y cómo los distintos componentes (como la CPU, la memoria y los dispositivos de entrada/salida) interactúan para realizar tareas de procesamiento de datos.

- La arquitectura es la perspectiva del programador sobre una computadora.
- En esta asignatura nos vamos a focalizar fundamentalmente en la arquitectura [x86-64](#).
- x86-64 es un diseño de CPU de conjunto de instrucciones complejas (CISC, *Complex Instruction Set Computing*).

Descripción básica de la arquitectura de una computadora

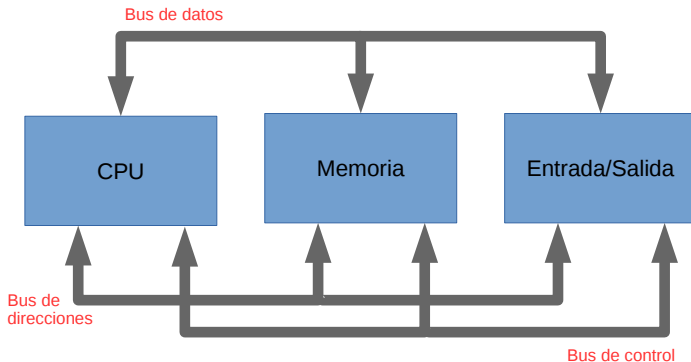


Descripción básica de la arquitectura de una computadora

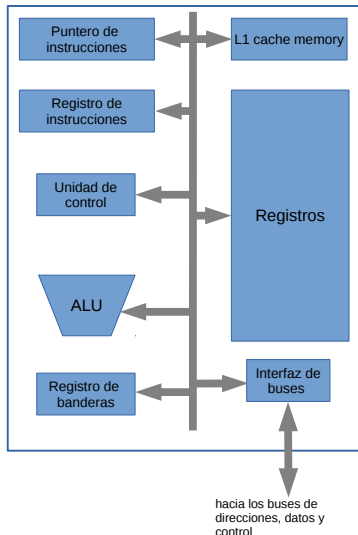
- Los componentes básicos de una computadora incluyen la Unidad Central de Procesamiento (CPU), la memoria principal o de acceso aleatorio (RAM), el almacenamiento secundario, los dispositivos de entrada y salida (pantalla, teclado, mouse, etc.), y los buses.
- La arquitectura de la figura anterior se conoce típicamente como Arquitectura de von Neumann y fue descrita en 1945 por el matemático y físico John von Neumann.
- La unidad central de procesamiento (CPU) es el “motor” que interpreta (o ejecuta) las instrucciones almacenadas en la memoria principal.
- La CPU incluye varias unidades funcionales, incluida la unidad aritmética lógica (ALU, *Arithmetic Logic Unit*), que es la parte del chip que realmente realiza los cálculos aritméticos y lógicos.

Subsistemas de una computadora

La CPU, la memoria y los periféricos de entrada/salida se comunican entre sí a través de tres buses diferentes.



Unidad Central de Procesamiento (CPU)



- Este es un esquema muy simplificado de la CPU.
- Los subsistemas de la CPU están conectados entre sí a través de buses internos.
- La CPU se conecta con el resto de los componentes de la computadoras (memoria, periféricos de entrada/salida, etc.) a través de buses de comunicación: bus de datos, bus de direcciones y bus de control.

Unidad Central de Procesamiento (CPU)

- **Interfaz de buses:** Se encarga de comunicar con el resto del sistema informático: dispositivos de memoria y de E/S.
- **Registros:** Un registro es una pequeña cantidad de almacenamiento rápido integrado en la propia CPU que se utiliza para almacenar datos temporales.
- **Memoria caché:** La mayoría de las CPU modernas incluyen una memoria caché muy rápida en el chip de la CPU.
- **Puntero de instrucción:** Es un registro que siempre contiene la dirección de la siguiente instrucción que se ejecutará.
- **Registro de instrucción:** Este registro contiene la instrucción que se está ejecutando en ese momento. Su contenido específica determina las acciones que la unidad de control debe realizar para dirigir la CPU.

Unidad Central de Procesamiento (CPU)

- **Unidad de control:** El contenido del registro de instrucciones se decodifican en la unidad de control. Genera las señales que controlan los otros subsistemas de la CPU para llevar a cabo las acciones especificadas por la instrucción.
- **Unidad aritmética lógica (ALU):** Dispositivo que realiza operaciones aritméticas y lógicas entre los operandos especificados en la instrucción. Para ello utiliza circuitos lógicos.
- **Registro de banderas:** Cada operación realizada por la ALU da como resultado varias condiciones (acarreo, desbordamiento, cero, paridad, etc.) que se guardan en este registro.

Memoria principal

- La memoria principal es un dispositivo de almacenamiento temporal que contiene tanto el programa como los datos que manipula mientras el procesador ejecuta el programa.
- Físicamente, la memoria principal consiste en una colección de chips de memoria dinámica de acceso aleatorio (DRAM, *Dynamic Random Access Memory*).
- A nivel lógico, la memoria está organizada como una matriz lineal de bytes cada uno con su propia dirección única que comienza en cero.
- El almacenamiento primario o la memoria principal también se conoce como memoria volátil dado que cuando se corta la energía, la información no se retiene y, por lo tanto, se pierde.

- La RAM se utiliza para tareas que requieren acceso rápido a datos temporales durante el funcionamiento de la computadora, pero no es adecuada para almacenamiento permanente.
- El almacenamiento secundario (por ejemplo, el disco rígido) se denomina memoria no volátil ya que la información se retiene cuando la computadora se apaga.
- Los discos duros son dispositivos de almacenamiento magnético que guardan permanentemente archivos y programas, aunque son más lentos en velocidad de acceso comparados con la RAM.
- Las unidades de estado sólido (SSD, *Solid State Drive*) utilizan memoria flash sin partes móviles, lo que les permite ofrecer un rendimiento superior en velocidad y tiempo de acceso.

- Los buses son conductores eléctricos que transportan información entre los componentes del sistema, transfiriendo datos en fragmentos de tamaño fijo llamados palabras.
- El tamaño de la palabra, que varía entre sistemas, es un parámetro fundamental.
- En la mayoría de las computadoras actuales, el tamaño de palabra es de 4 bytes (32 bits) 8 bytes (64 bits).
- Esta asignatura se enfocará principalmente en arquitectura de 64 bits.

Dispositivos de entrada/salida

- Los dispositivos de entrada/salida (I/O) conectan el sistema con el mundo externo.
- Ejemplos comunes incluyen teclado, mouse, pantalla y disco rígido, que permiten la interacción del usuario y el almacenamiento de datos.
- Cada dispositivo de E/S se conecta al bus de E/S a través de un controlador o un adaptador.
- Los controladores son chips integrados en el dispositivo o en la placa base, mientras que los adaptadores son tarjetas que se insertan en la placa base para facilitar la transferencia de datos.

Conjunto de instrucciones

- El conjunto de instrucciones (ISA, *Instruction Set Architecture*) define el conjunto de instrucciones disponibles para controlar la CPU, actuando como interfaz entre el hardware y el software.
- El ISA abarca los tipos de datos admitidos, registros, administración de memoria, características clave como la memoria virtual, y las instrucciones que puede ejecutar un microprocesador.
- Se clasifican comúnmente en CISC (*Complex Instruction Set Computer*) y RISC (*Reduced Instruction Set Computer*).
- El curso se enfoca inicialmente en la arquitectura [x86-64](#), un diseño de CPU tipo CISC. Al final del curso, se estudiará la arquitectura RISC, específicamente [ARM](#).

Breve historia de la computación

Se han construido cientos de tipos diferentes de computadoras desde sus orígenes. Aquí daremos un pantallazo muy breve sobre los orígenes de la computación marcando algunos hitos relevantes.

Para clarificar, dividimos en las siguientes categorías:

- Computadoras mecánicas
- Computadoras con válvulas de vacío
- Computadoras con transistores
- Computadoras con circuitos integrados
- Computadoras con integración a muy gran escala

- Blaise Pascal (1623-1662). Dispositivo construido en 1642 construido totalmente mecánico con engranajes. Solo podía restar y sumar.
- Goofried von Leibnitz (1646-1716). Construyó otra máquina totalmente mecánica que además podía multiplicar y dividir.
- Charles Babbage (1792-1871). Construyó una máquina diseñada para ejecutar un solo algoritmo con el objetivo de calcular tablas numéricas útiles para la navegación. Perforaba sus resultados en una placa de cobre.
- Desarrolló la máquina analítica para ejecutar distintos algoritmos, que constaba de cuatro componentes: almacén (memoria), molino (unidad de cómputo), sección de entrada (lector de tarjetas perforadas) y sección de salida (salidas perforadas e impresas). Podía realizar sumas, restas, multiplicaciones y divisiones.

- Ada Lovelace (1815-1852). Primera programadora de computadoras del mundo.
- Konrad Zuse (1910-1995). Construyó una serie de máquinas calculadoras automáticas empleando contactores electromagnéticos.
- John Atanasoff (1903-1995). Diseñó una máquina que utilizaba aritmética binaria y tenía una memoria de condensadores empleando un proceso que denominó “refrescar la memoria”. Nunca funcionó por problemas de implementación aunque el concepto era correcto.
- George Stibbitz (1904-1995). Realizó una máquina más primitiva que la de Atanasoff aunque si funcionó.

- Howard Aiken (1900-1973). Construyó con relés la máquina de propósito general que Babbage no pudo construir con ruedas dentadas.
- La primera máquina de Aiken, conocida como Mark I, se completó en Harvard en 1944.
- Contaba con 72 palabras de 23 dígitos decimales cada una y un tiempo de ejecución por instrucción de 6 segundos. Las entradas y salidas se realizaban mediante cintas de papel perforadas.

Computadoras mecánicas

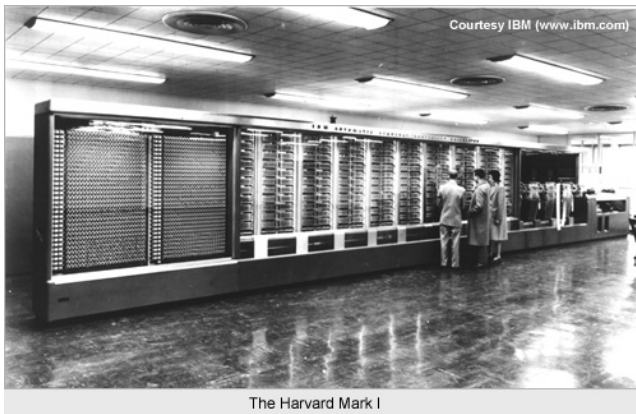


Figura: Imagen de la computadora Mark I.

Computadoras con válvulas de vacío

- Componente electrónico utilizado para:
 - Amplificar
 - Conmutar
 - Modificar señales eléctricas
- Funciona controlando el movimiento de electrones en un espacio “vacío” a muy baja presión o en presencia de gases seleccionados.



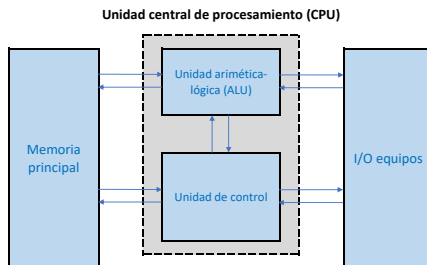
Figura: Imagen de una válvula electrónica.

Computadoras con válvulas de vacío

- Alan Turing (1912-1954). Contribuyó a la creación de COLOSSUS, la primera computadora electrónica, desarrollada por el gobierno inglés.
- John Mauchley (1907-1980). Construyó la ENIAC (*Electronic Numerical Integrator And Computer*) en 1946, la primera computadora digital, electrónica, de propósito general y programable. Tenía 18000 válvulas y 1500 relés. Pesaba 30 toneladas y consumía 140 kW. Trabajaba en sistema decimal.
- EDSAC. Máquina sucesora de ENIAC en 1949. Trabajaba en sistema binario.
- EDVAC (*Electronic Discrete Variable Automatic Computer*). Fue una de las primeras computadoras electrónicas (1951). A diferencia de la ENIAC, era binaria en lugar de decimal y fue diseñada para ser una computadora de programa almacenado. El diseño estuvo basado en el concepto de Máquina de Turing.

Computadoras con válvulas de vacío

- John von Neumann (1903-1957). En 1952, desarrolló la máquina IAS, que utilizaba el diseño ahora conocido como **máquina de Von Neumann**, el cual consta de cinco partes básicas: la memoria, la unidad aritmético-lógica, la unidad de control, y los dispositivos de entrada y salida. Este diseño sigue siendo la base de casi todas las computadoras digitales hasta el día de hoy.



- Whirlwind I (1951). Diseñada en el MIT. Tenía palabras de 16 bits y estaba diseñada para el control en tiempo real.

Computadoras con válvulas de vacío

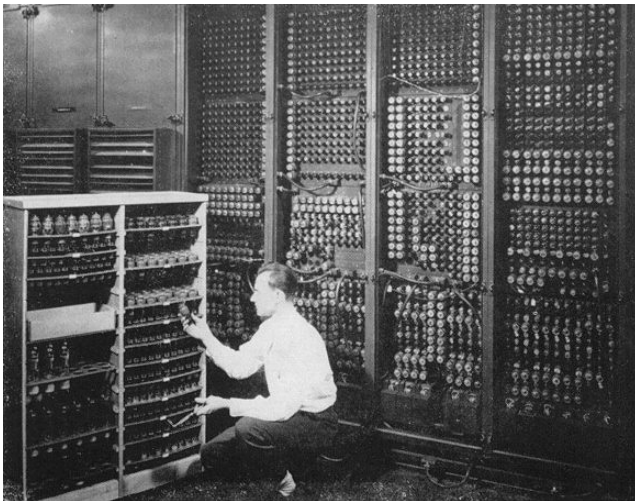
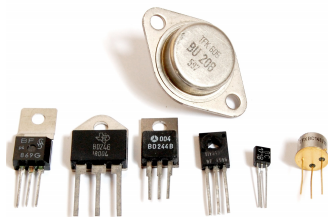


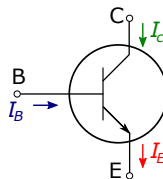
Figura: Imagen de la computadora ENIAC.

Computadoras con transistores

- En 1948, John Bardeen, Walter Brattain y William Schochley inventan el transistor trabajando en los Laboratorios Bell.
- El transistor es un dispositivo electrónico semiconductor utilizado para entregar una señal de salida en respuesta a una señal de entrada.
- Los transistores pueden ser utilizados como interruptores, al igual que las válvulas, pero con mucho menor tamaño, costo y disipación de calor.



(a) Algunos tipos de transistores.



(b) Diagrama de transistor tipo NPN.

Computadoras con transistores

- TX-0 (1956). Primera computadora transistorizada, desarrollada en el Lincoln Laboratory del MIT.
- PDP-1 (1960). Primera microcomputadora. 4K de palabras de 18 bits y tiempo de ciclo de 5 μs .
- 1401 (1961). Desarrollada por IBM y orientada a la contabilidad comercial.
- 7094 (1962). Desarrollada por IBM y orientada a la computación científica.
- 6600 (1964). Desarrollada por CDC. Primera supercomputadora científica.

En 1958, Robert Noyce inventó el circuito integrado de silicio, lo cual hizo posible colocar docenas de transistores en un solo chip.

- System/360 (1964). Desarrollada por IBM como familia de productos tanto para computación científica como comercial. Una importante innovación fue la multiprogramación.
- PDP-8 (1965). Primera minicomputadora con mercado masivo (50000 unidades vendidas).
- PDP-11 (1970). Dominó el mercado de las minicomputadoras en los años setenta.

Computadoras con circuitos integrados

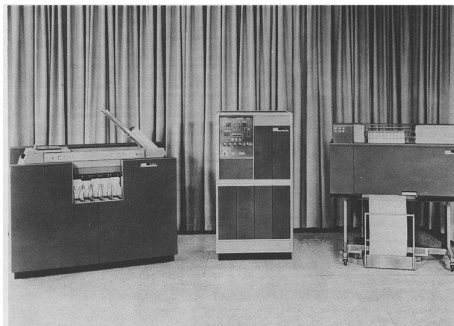


Figura: Imagen de la computadora IBM 1401.

Computadoras con integración a muy gran escala

- La integración a escala muy grande o VLSI (sigla en inglés de very-large-scale integration) es el proceso de crear un circuito integrado compuesto por cientos de miles de transistores en un único chip.
- VLSI comenzó a usarse en los años 70, como parte de las tecnologías de semiconductores y comunicación que se estaban desarrollando.
- Intel 8080 (1974). Primera computadora de propósito general de 8 bits en un chip. Trabajaba a 2 MHz y se le considera el primer diseño de microprocesador verdaderamente usable.



Figura: Microprocesador Intel 8080.

Computadoras con integración a muy gran escala

- Apple II (1977). Primera serie de microcomputadores de producción masiva hecha por la empresa Apple Computer. Arquitectura de 8 bits.
- 8086 (1978). Uno de los primeros chips individuales. Microprocesador de 16 bits. 29 K transistores. El 8088, una variante del 8086 con un bus externo de 8 bits fue utilizado en computadora personal original de IBM.



Figura: Computadora Apple II.

Computadoras con integración a muy gran escala

- 8087 (1980) Intel introdujo el coprocesador 8087 de punto flotante (45 K transistores) para operar junto a un procesador 8086 o 8088 ejecutando la instrucciones de punto flotante. El 8087 estableció el modelo de punto flotante para la línea x86, a menudo denominado “x87”.
- IBM Personal Computer (1981). Se convirtió en la computadora más vendida de la historia. Venía equipada con el sistema operativo MS-DOS provisto por la compañía Microsoft Corporation:
- 80286 (1982). Agregó más modos de direccionamiento de memoria. Formó la base de la computadora personal IBM PC-AT, la plataforma original para MS Windows. 134 K transistores
- RISC-I (1982). Desarrollada dentro del proyecto RISC en la Universidad de Berkeley bajo la dirección de David A. Patterson.

Computadoras con integración a muy gran escala

- R2000 (1985). Primer diseño MIPS por John L. Hennessy, el cual mejoró drásticamente el rendimiento mediante el uso de la segmentación.
- i386 (1985). Expandió la arquitectura a 32 bits. 275 K transistores.
- i486 (1989). mejoró el desempeño de la línea x86 e integró la unidad de punto flotante en el chip pero no tuvo cambios significativos en el conjunto de instrucciones. 1.2 M transistores.
- Pentium (1993). Mejoró el desempeño pero solo agregó extensiones menores al conjunto de instrucciones. 3.1 M transistores.
- PentiumPro (1995). Introdujo un diseño de procesador radicalmente nuevo, conocido internamente como la microarquitectura P6. Se agregó una clase de instrucciones de movimiento condicional al conjunto de instrucciones. 5.5 M transistores.
- Pentium II (1997). Continuación of microarquitectura P6. 7 M transistores.

Computadoras con integración a muy gran escala

- Pentium III (1999). Introdujo el SSE, una clase of instrucciones para manipular vectores de datos enteros o de punto flotante. 8.2 M transistores pero versiones posteriores llegaron a 24 M transistores.
- Pentium 4 (2000). Extendió SSE a SSE2, agregando nuevos tipos de datos (incluyendo punto flotante de doble precisión), junto con muchas más instrucciones para estos tipos de datos. 42 M transistores.
- Pentium 4E (2004). Arquitectura x86-64. 125 M transistores.

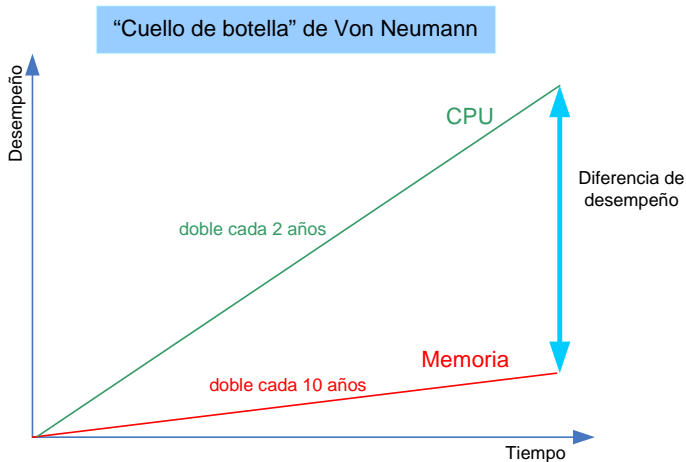
Computadoras con integración a muy gran escala

- Intel Core 2 (2006). Primer microprocesador Intel multinúcleo.
- Intel Core i7 (2008). Incorpora tanto *hyperthreading* como *multi-core*.

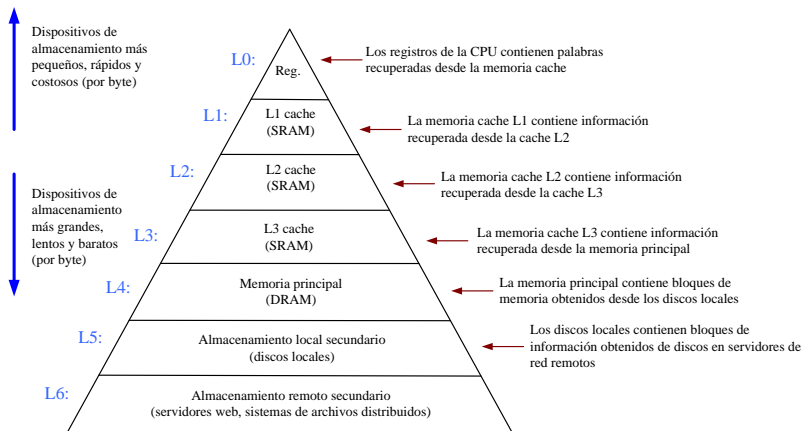


Figura: Intel Core i7

Jerarquía de memoria



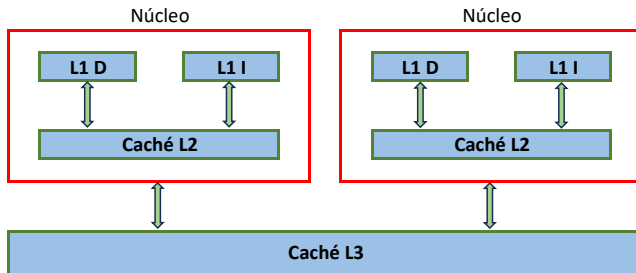
Jerarquía de memoria



Jerarquía de memoria

- Los dispositivos de almacenamiento en cada sistema informático están organizados como una jerarquía de memoria similar a la figura anterior.
- A medida que avanzamos desde la parte superior de la jerarquía hacia la parte inferior, los dispositivos se vuelven más lentos, más grandes y menos costosos por byte.
- Los registros ocupan el nivel superior en la jerarquía, que se conoce como nivel 0 o L0.
- La memoria principal ocupa el nivel 4.
- Se incluyen dispositivos de almacenamiento más pequeños y más rápidos que la memoria principal llamados memorias de caché que sirven como áreas de almacenamiento temporales.

Memoria Caché



- La memoria caché es un pequeño subconjunto del almacenamiento primario que se encuentra integrado en el chip de la CPU.
- Está diseñada para combinar la velocidad de los registros de la CPU con el tamaño de la memoria principal, equilibrando costo y rendimiento.
- Tiene su propia jerarquía organizada en tres niveles.

Representación de programas a nivel de máquina

- Las computadoras ejecutan código de máquina, secuencias de bytes que codifican las operaciones de bajo nivel.
- Un compilador genera código de máquina a través de una serie de etapas, basadas en las reglas del lenguaje de programación, el conjunto de instrucciones de la máquina de destino y las convenciones seguidas por el sistema operativo.
- Es interesante analizar el código de máquina y sobre todo su representación legible por humanos como es el código Assembler.

Representación de programas a nivel de máquina

Por ejemplo, el código equivalente del simple programa en lenguaje C que imprime por pantalla la cadena de caracteres ¡Hola Mundo! es el siguiente:

```
.data
msg: .asciz "!Hola mundo!\n"

.text
.global main
main:
0x0000000000401126 <+0>:      55                                push    %rbp
0x0000000000401127 <+1>:      48 c7 c7 30 40 40 00      movq    $msg, %rdi
0x000000000040112e <+8>:      48 c7 c0 00 00 00 00      movq    $0x0,%rax
0x0000000000401135 <+15>:     e8 f6 fe ff ff           call    printf
0x000000000040113a <+20>:     31 c0                    xorl    %eax, %eax
0x000000000040113c <+22>:     5d                        pop     %rbp
0x000000000040113d <+23>:     c3                        ret
```

Representación de programas a nivel de máquina

- En la parte central vemos el código de máquina.
- En la parte derecha vemos el equivalente en lenguaje Assembler.
- Es evidente que esta representación es mucho más legible y comprensible para un humano que el código de máquina puro.
- El lenguaje ensamblador es específico para cada tipo de máquina; por lo tanto, el código escrito para un procesador x86-64 no funcionará en otros procesadores, como los basados en la arquitectura RISC.
- El lenguaje ensamblador es un lenguaje de “bajo nivel” y proporciona la interfaz de instrucción básica para el procesador de la computadora.
- Para un programador, el lenguaje ensamblador es lo más cercano al procesador.

¿Por qué deberíamos aprender Assembler?

El lenguaje ensamblador tiene varios beneficios:

- **Velocidad.** Los programas en lenguaje ensamblador son generalmente los programas más rápidos.
- **Espacio.** Los programas en lenguaje ensamblador suelen ser los más pequeños.
- **Capacidad.** Se pueden hacer cosas en lenguaje ensamblador que son difíciles o imposibles en lenguajes de alto nivel.
- **Conocimiento.** El conocimiento del lenguaje ensamblador ayuda a escribir mejores programas.