

Introducción a Arquitectura de las Computadoras

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

Arquitectura del Computador - LCC - FCEIA - UNR

Agosto 2022



Contenido de la presentación

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



Definición de computadora

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.



Definición de Arquitectura de Computadores

- El término **Arquitectura del Computador** se refiere a múltiples aspectos, los cuales se pueden dividir en tres grandes grupos:
 - Diseño del conjunto de instrucciones,
 - Organización de la computadora ¹,
 - Diseño del sistema.
- El conjunto de instrucciones de la arquitectura (ISA, *Instruction Set Architecture*) es el conjunto de instrucciones real disponible para programar.
- 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*).

¹Organización de la computadora, también llamada microarquitectura, es la forma en la que se implementa el conjunto de instrucciones.



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

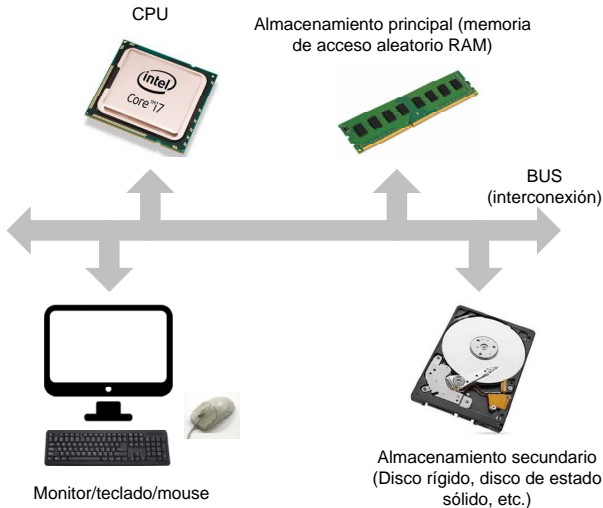


Referencia bibliográfica

Una breve historia de la computación se puede encontrar en el Apunte 0: “Introducción a Arquitectura de las Computadoras”.



Descripción básica de la arquitectura de una computadora



Descripción básica de la arquitectura de una computadora

- Los componentes básicos de una computadora incluyen una Unidad Central de Procesamiento (CPU), almacenamiento primario o memoria de acceso aleatorio (RAM, Random Access Memory), almacenamiento secundario, dispositivos de entrada/salida (pantalla, teclado, mouse, etc.) y buses.
- La unidad central de procesamiento (CPU), o simplemente el procesador, 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, *Aritmetic Logic Unit*), que es la parte del chip que realmente realiza los cálculos aritméticos y lógicos.



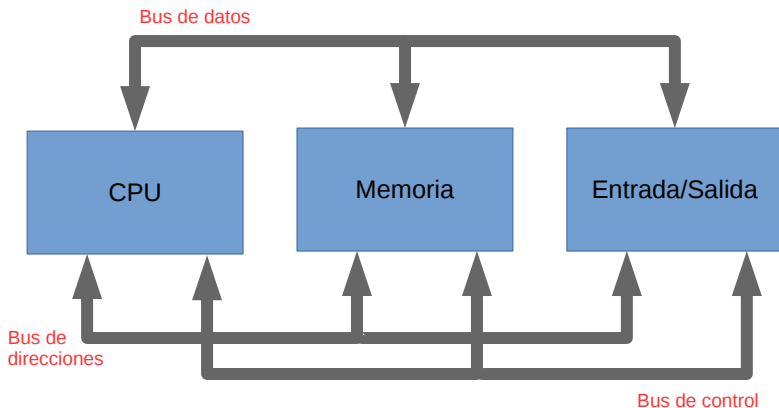
Arquitectura de von Neumann

- La arquitectura de la figura anterior se conoce típicamente como Arquitectura de von Neumann, o Arquitectura de Princeton, y fue descrita en 1945 por el matemático y físico John von Neumann.
- La arquitectura von Neumann significa básicamente que se usa el mismo almacenamiento para el programa que está siendo ejecutado y para sus datos.
- El “cuello de botella” de von Neumann se produce porque los procesadores son cada vez más rápidos pero la memoria aumentó su velocidad a un ritmo mucho menor.
- Una respuesta parcial a este problema es la creación de una jerarquía de almacenamiento.



Subsistemas de una computadora

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

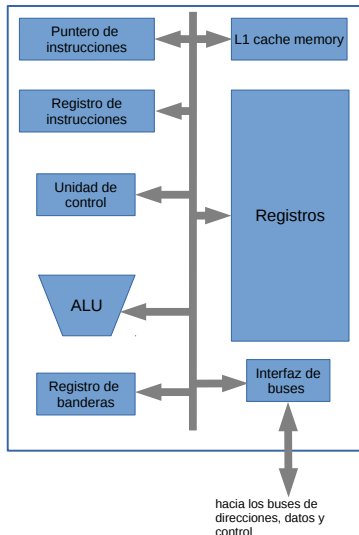


Memoria principal

- 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.
- El almacenamiento secundario (por ejemplo, el disco rígido) se denomina memoria no volátil ya que la información se retiene cuando la computadora se apaga.



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.
- A su vez, la CPU se conecta con el resto de los componentes de la computadora (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 un grupo de bits que está destinado a utilizarse como variable en un programa. Los compiladores y ensambladores tienen nombres para cada registro.
- **Memoria caché L1:** La mayoría de las CPU modernas incluyen una memoria caché muy rápida en el chip de la CPU.
- **Puntero de instrucción:** Este es un registro de 64 bits que siempre contiene la dirección de la siguiente instrucción que se ejecutará.
- **Registro de instrucción:** Este registro contiene la instrucción que se está ejecutando actualmente. Su patrón de bits determina lo que la unidad de control debe hacer para comandar la CPU.

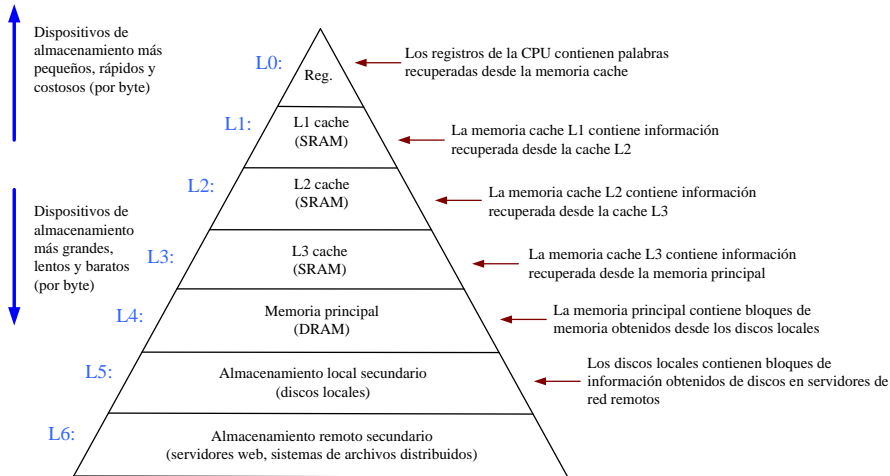


Unidad Central de Procesamiento (CPU)

- **Unidad de control:** Los bits 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.



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.



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 siguiente programa en lenguaje C que imprime por pantalla ¡Hola Mundo!

```
#include<stdio.h>
int main(){
    printf("¡Hola Mundo!\n");
    return 0;
}
```

es el siguiente (de manera simplificada):

0x401122 <+0>:	48 c7 c7 30 40 40 00	movq \$0x404030, %rdi
0x401129 <+7>:	48 c7 c0 00 00 00 00	movq \$0x0, %rax
0x401130 <+14>:	e8 fb fe ff ff	callq 0x401030 <printf@plt>
0x401135 <+19>:	c3	retq



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 de la máquina. Por ejemplo, el código escrito para un procesador x86-64 no se ejecutará en un procesador diferente, por ejemplo en un procesador RISC (popular en tabletas y teléfonos inteligentes).
- 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 HLL^a.
- **Conocimiento.** El conocimiento del lenguaje ensamblador ayuda a escribir mejores programas, incluso cuando se use un HLL.

^aHLL, programas de alto nivel por su sigla en inglés *high level languages*.

