

Formas de Organización de Computadores

Prof.Ing. Fabián Zamora Ramírez.

CE-1102 Taller de Programación

Área de Ingeniería en Computadores

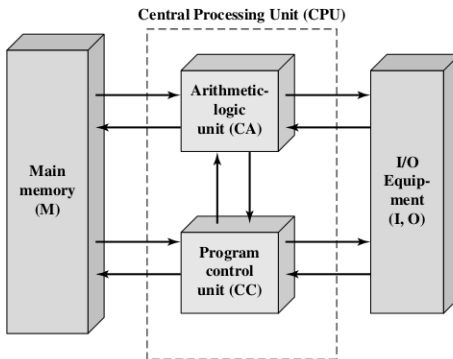
Instituto Tecnológico de Costa Rica

Agenda

- 1 Formas Básicas de Organización
 - Arquitectura Von Neumann
 - Arquitectura Harvard
 - Arquitectura Harvard Modificada
- 2 Arquitecturas CISC
- 3 Arquitecturas RISC
- 4 CISC vs RISC

Arquitectura Von Neumann

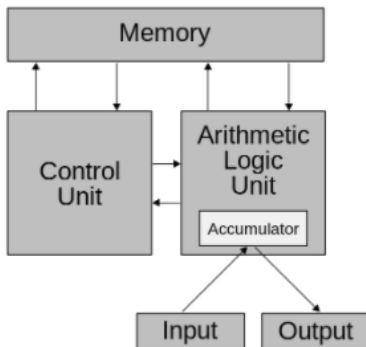
En 1946 John Von Neumann (Matemático - consultor en ENIAC) ideó un computador basado en programas almacenados (**store-program computer**) : IAS



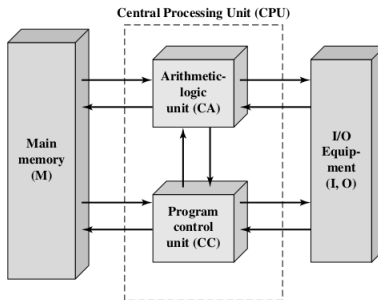
Direcciones de memoria

000	
001	
010	
011	
100	00000110
101	
110	00000100
111	

Unidad de procesamiento



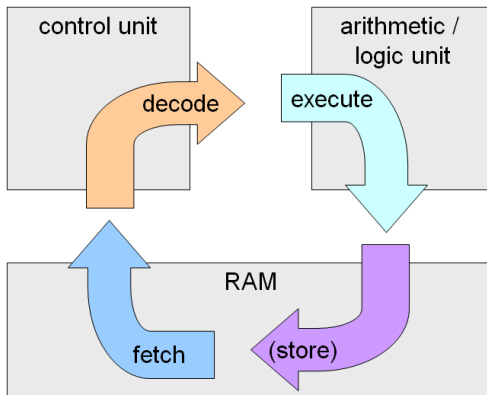
Dispositivos E/S



Características

- La información se representa por medio de direcciones
- **Memoria unificada:** una única memoria para datos y programa.
- Las instrucciones almacenadas y ejecutadas secuencialmente: Program counter debe actualizarse ($PC=PC+1$) para obtener siguiente instrucción.

Ciclo de Instrucción



Aplicación Von Nuemann

<http://vnsimulator.altervista.org/en/>

Arquitectura Harvard

La evolución en los computadores y la necesidad de rapidez y paralelismo generaron cambios en la organización de los computadores.

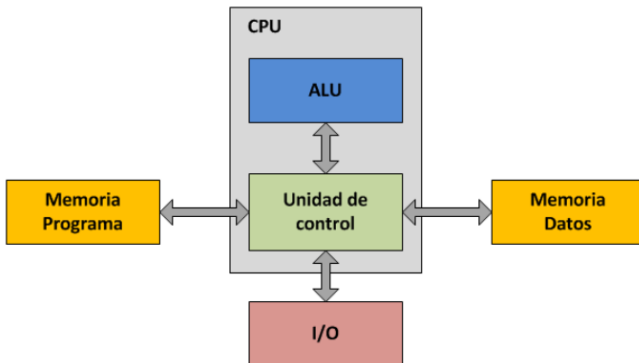
La memoria unificada de acceso secuencial representa un obstáculo (cuello de botella).

Solución: Separar la memoria - Memoria para instrucciones (I) y memoria para datos (D)

Arquitectura Harvard - Características

- Memoria de datos y memoria de programa están físicamente separadas
- Acceso a memoria de datos e instrucciones puede ser simultáneo
- Ventaja: mayor rendimiento (paralelismo)
- La mayoría DSP poseen una arquitectura tipo Harvard.

Arquitectura Harvard



Formas de Organización - Comparación

- **Arquitectura Von Neumann:** un único espacio de direccionamiento, y única ruta de acceso al CPU.
- **Arquitectura Harvard:** memoria de datos y memoria de instrucciones tienen rutas de hardware diferentes hacia el CPU, además de espacios de direccionamiento separados.

Rutas separadas hacia el CPU permiten que instrucciones y datos sean accedidos al mismo tiempo, mejorando el throughput.

Complex Instruction Set Computer - CISC

Set de instrucciones Complejo:

- Enfoque inicial de arquitectura
- El ISA contiene gran variedad de instrucciones: instrucciones poderosas y complicadas
- Facilidad de programación
- El compilador no realiza traducciones complejas: instrucciones son similares a lenguajes de alto nivel

Ejemplo clásico: Arquitectura Intel 80x86

Complex Instruction Set Computer

Características típicas del set:

- Múltiples modos de direccionamiento (formas de acceder a datos en memoria)
- Tamaño y formato de instrucciones variable
- Tiempo de duración de instrucciones variable
- Bajo número de registros de propósito general. En 80x86: AX,BX,CX,DX
- Las instrucciones son capaces de ejecutar tareas complejas
- Decodificación de instrucciones es complicado: Hardware es complicado

Ventajas y desventajas

Ventajas

- Facilidad de programación: Tareas complejas se realizan en menos tiempo
- Múltiples modos de direccionamiento simplifican las tareas
- Tamaño de código pequeño

Desventajas

- Instrucciones de tamaño variable: diferente tiempo de búsqueda y ejecución hacen muy complicado tener sistemas determinísticos. Hardware es complicado (área, \$).
- Muchas de las instrucciones especializadas no son utilizadas con frecuencia: El 98% de las instrucciones en un programa típico corresponden al 20 % de las instrucciones del set.

Reduced Instruction Set Computers - RISC

Set de Instrucciones reducido:

- Enfoque moderno (con aplicación moderna): DSPs, CPUs para sistemas embebidos
- El set está compuesto por pocas instrucciones con funcionalidad simple.
- La dificultad está en el programador (bajo nivel) o el compilador.

Ejemplos: MIPS, ARM

Reduced Instruction Set Computers

Características típicas del set

- Pocos método de direccionamiento (1-4)
- Las instrucciones tiene un tamaño fijo.
- El tiempo de ejecución de cada instrucción es el mismo.
- Alto número de registros de propósito general (16,+32)
- Decodificación de instrucciones simple.
- Funcionalidades complejas suelen ser implementadas por Hardware.

Ventajas y desventajas

Ventajas

- Instrucciones de tiempo y tamaño fijo: simplifica hardware y brinda determinismo
- Mejor aprovechamiento del Hardware
- Permite Pipeline

Desventajas

- Tamaño de código mayor
- Carga pesada para el software (programador de bajo nivel o compilador)

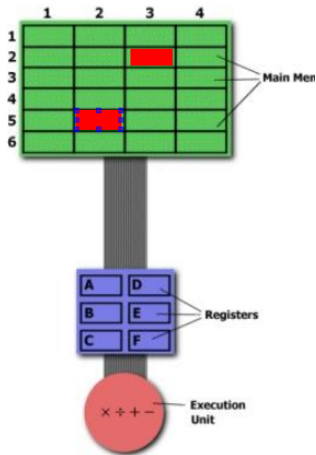
CISC vs RISC

	Complex Instruction Set (CISC) Computer			Reduced Instruction Set (RISC) Computer	
Characteristic	IBM 370/168	VAX 11/780	Intel 80486	SPARC	MIPS R4000
Year developed	1973	1978	1989	1987	1991
Number of instructions	208	303	235	69	94
Instruction size (bytes)	2–6	2–57	1–11	4	4
Addressing modes	4	22	11	1	1
Number of general-purpose registers	16	16	8	40–520	32
Control memory size (Kbits)	420	480	246	—	—
Cache size (KBytes)	64	64	8	32	128

CISC vs RISC

CISC vs RISC: Ejemplo

Se desea multiplicar 2 números en las direcciones 2:3 y 5:2 y almacenar el resultado en la dirección 2:3



CISC vs RISC: Ejemplo

Enfoque CISC:

- Tener la menor cantidad de líneas posibles de ensamblador
- Operar directamente desde memoria

MUL 2:3 , 5:2

Enfoque RISC:

- Ejecutar instrucciones simples
- Una instrucción por ciclo

LOAD A, 2:3
LOAD B, 5:2
PROD A,B
STORE 2:3, A

Referencias



Yale N. Patt & Sanjay J. Patel (2005)

Introduction to computing systems [Cap 4]



Andrew S. Tanenbaum (2000)

Organización de computadoras - Un enfoque estructurado. [Cap 2]



Jeferson González G. (2017)

Material de clase: Arquitectura de computadores I.

Sitios web recomendados

- Arquitectura Harvard
https://en.wikipedia.org/wiki/Harvard_architecture
- Arquitectura Harvard Modificada
https://en.wikipedia.org/wiki/Modified_Harvard_architecture