Tema 10 - Arquitectura de microprocesadores actuales

Sistemas Electrónicos para Automatización
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Hipólito Guzmán Miranda

Contenido

- Revisión de arquitecturas clásicas
- Concepto de pipeline
- Memorias caché

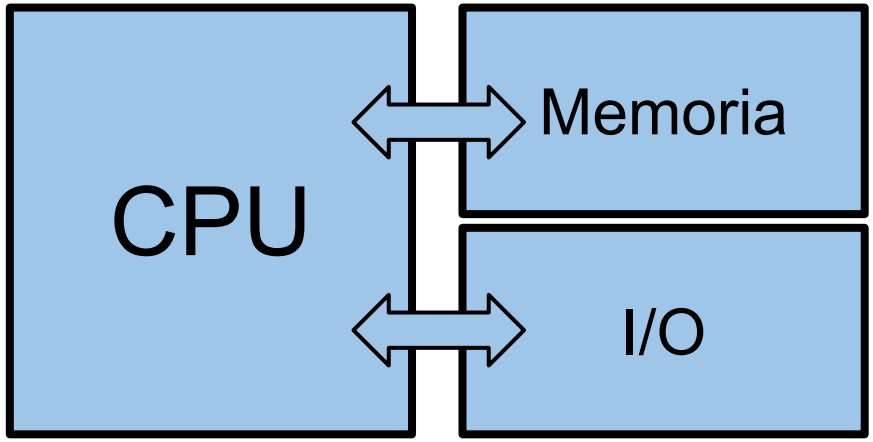
Contenido

- Revisión de arquitecturas clásicas
- Concepto de pipeline
- Memorias caché

Revisión arquitecturas clásicas

La separación entre la CPU y la memoria hace que el computador sea programable

- CPU (Unidad Central de Proceso)
- Memoria
- Perif. Entrada/Salida



Von Neumann vs Harvard

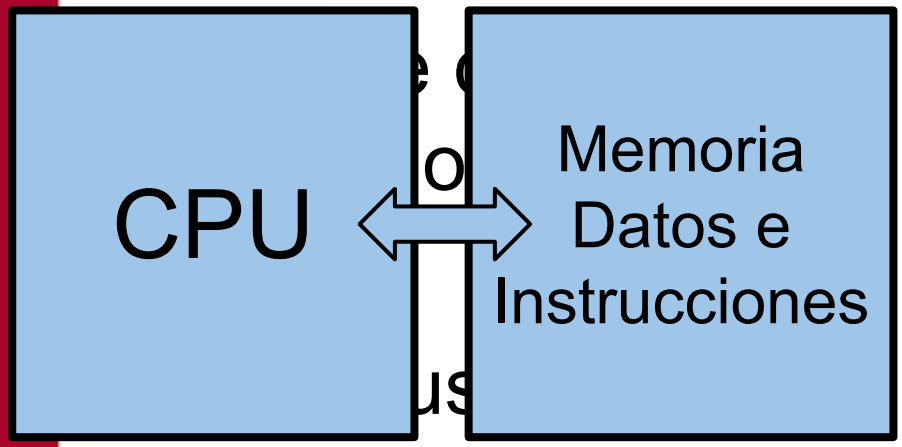
Von Neumann	Harvard
La misma memoria contiene datos e instrucciones	Memorias separadas para datos e instrucciones
Único bus datos/direcciones entre CPU y memoria	Dos buses datos/direcciones entre CPU y
Von Neumann bottleneck!	Mayor uso de recursos ($)

Revisión arquitecturas clásicas

Von Neumann vs Harvard

Von Neumann

La misma memoria

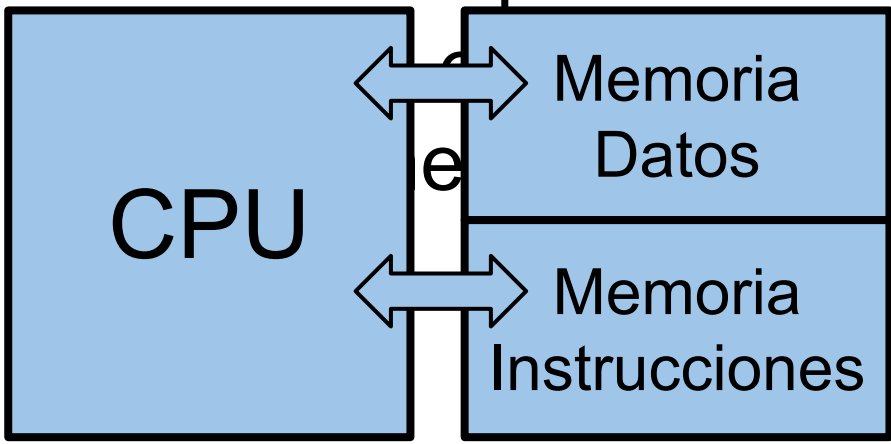


datos/direcciones
entre CPU y memoria

Von Neumann bottleneck!

Harvard

Memorias separadas



datos/direcciones
entre CPU y

Mayor uso de recursos ($)

RISC vs CISC

Reduced Instruction Set Computer

Instrucciones compactas y uniformes

Facilita el pipelining

Mayor uso de memoria

Permiten mejores optimizaciones al compilador

Complex Instruction Set Computer

Instrucciones anchas

Múltiples modos de direccionamiento

Uso de memoria más eficiente (alta densidad de código)

Posible necesidad de optimizaciones manuales (sist. empotrados)

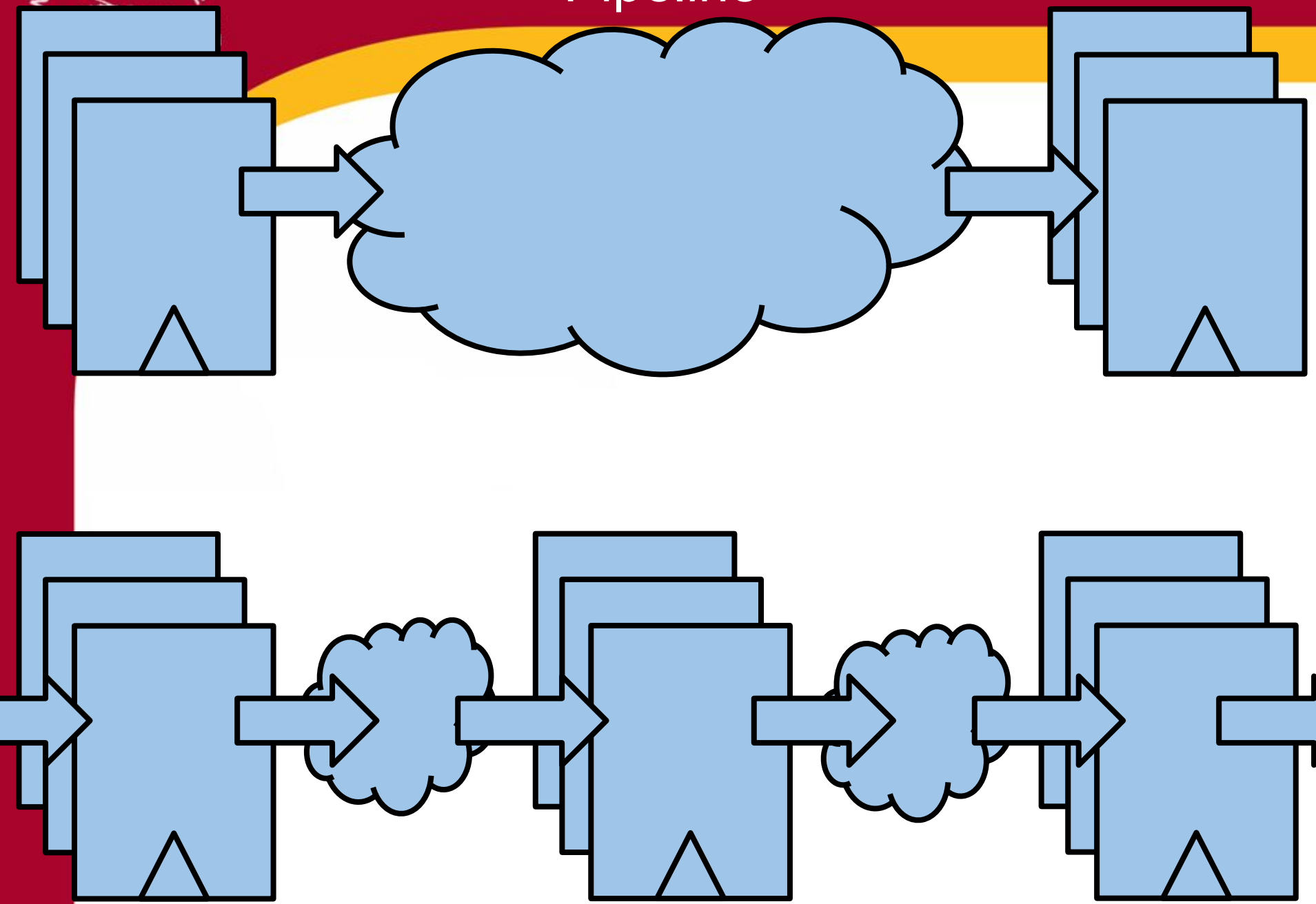
Algunos ejemplos

	Von Neumann	**Harvard**
RISC	MSP430 ARM7	MicroBlaze Leon4 ARM9
CISC	Pentium	8051 SHARC DSP DSPs en general

Contenido

- Revisión de arquitecturas clásicas
- Concepto de pipeline
- Memorias caché

Pipeline



Latencia y Throughput

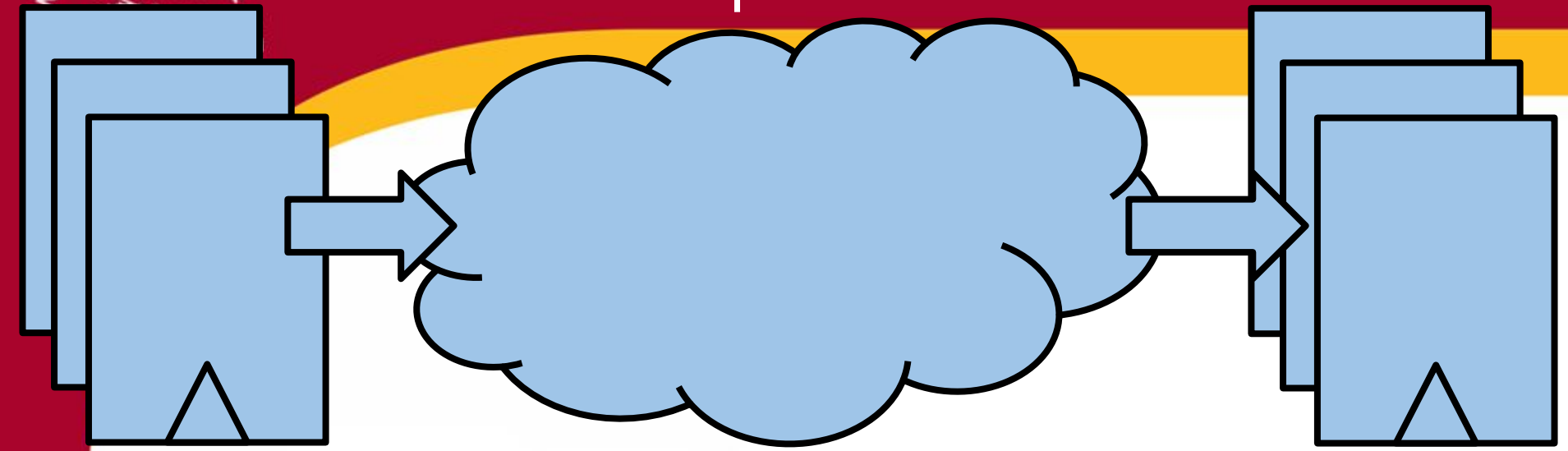
Latencia: tiempo entre que entra el dato y sale procesado

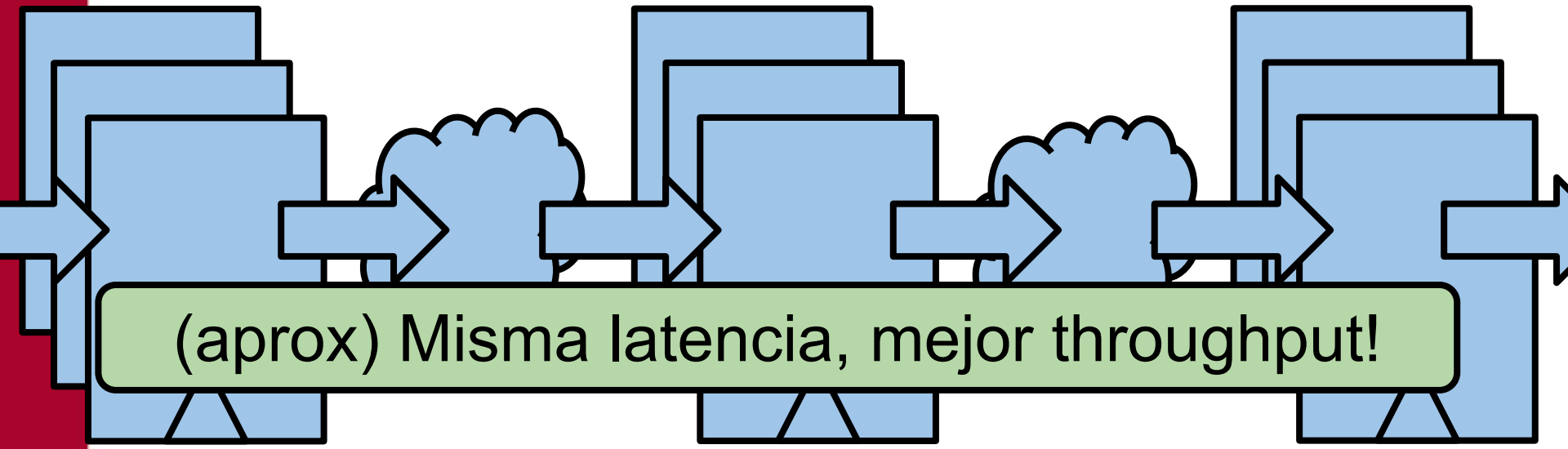
(tarda M ciclos en cruzar el pipeline completo)

Throughput: cada cuántos ciclos sale un dato nuevo

(un dato cada N ciclos)

Pipeline



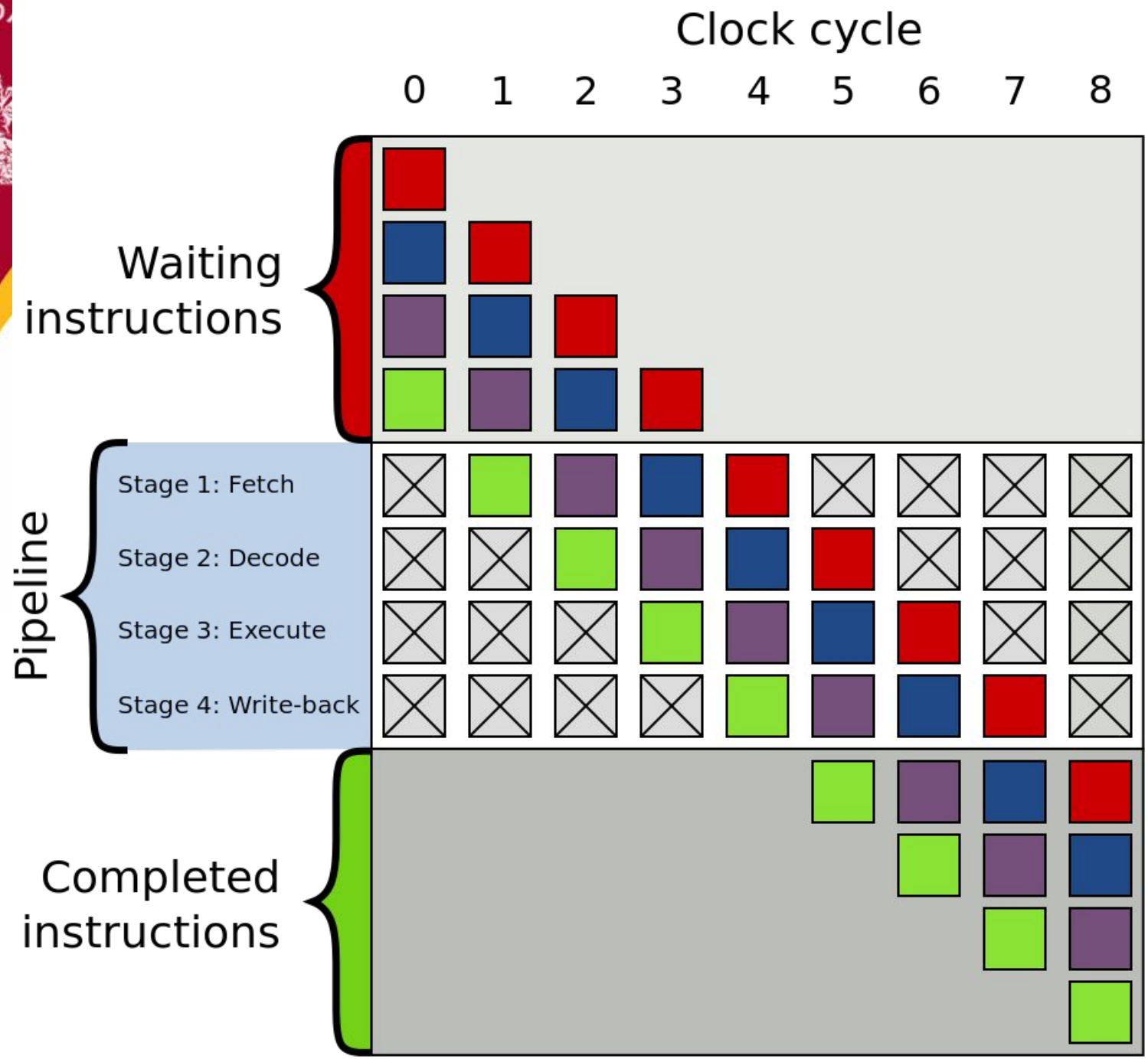


(aprox) Misma latencia, mejor throughput!

Técnica muy utilizada en microprocesadores

Si para cada instrucción tenemos que hacer:

- Fetch
- Decode
- Execute
- Write-back

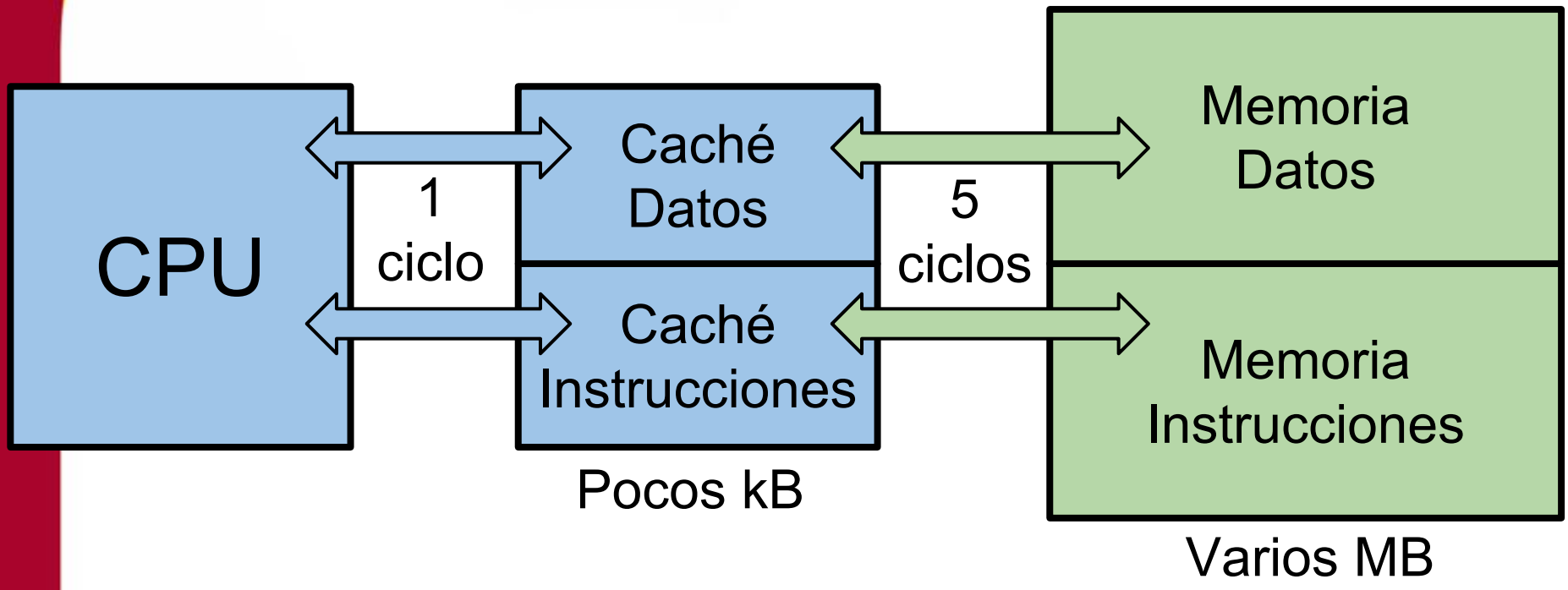

Clock cycle
0
1
2
3
4
5
6
7
8
Waiting instructions
Pipeline
Stage 1: Fetch
Stage 2: Decode
Stage 3: Execute
Stage 4: Write-back
Completed instructions

Contenido

- Revisión de arquitecturas clásicas
- Concepto de pipeline
- Memorias caché

Memorias caché

Memoria auxiliar de acceso más rápido



Funcionamiento

La caché tiene menos capacidad que la memoria externa, pero es más rápida.

La caché guarda ‘paginas’ de memoria completas

Es ‘transparente’ para el procesador: el micro habla con la caché y la caché lee la memoria externa si necesita datos

Funcionamiento

Cache hit: la dirección (address) solicitada por el microprocesador está en alguna de las páginas almacenadas en la caché (más rápido).

Cache miss: la dirección solicitada no está almacenada en la caché. La caché debe realizar el acceso externo (más tiempo).