

# Organización de la CPU

Departamento de Arquitectura<sup>1</sup>

<sup>1</sup>Instituto de Computación  
Facultad de Ingeniería  
Universidad de la República

Arquitectura de Computadoras, 2024

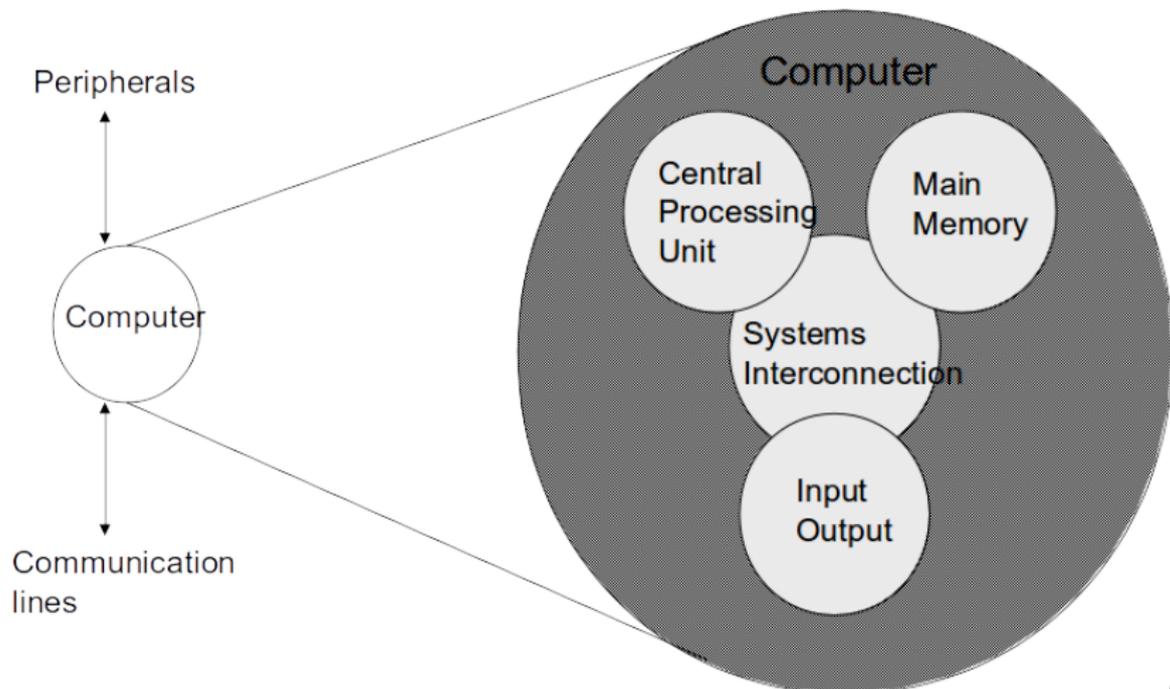
# Contextualización

- Arquitectura Von Neumann.
- Conjunto de instrucciones de la arquitectura.
- Cerrando la brecha hardware-software.

# Contenido

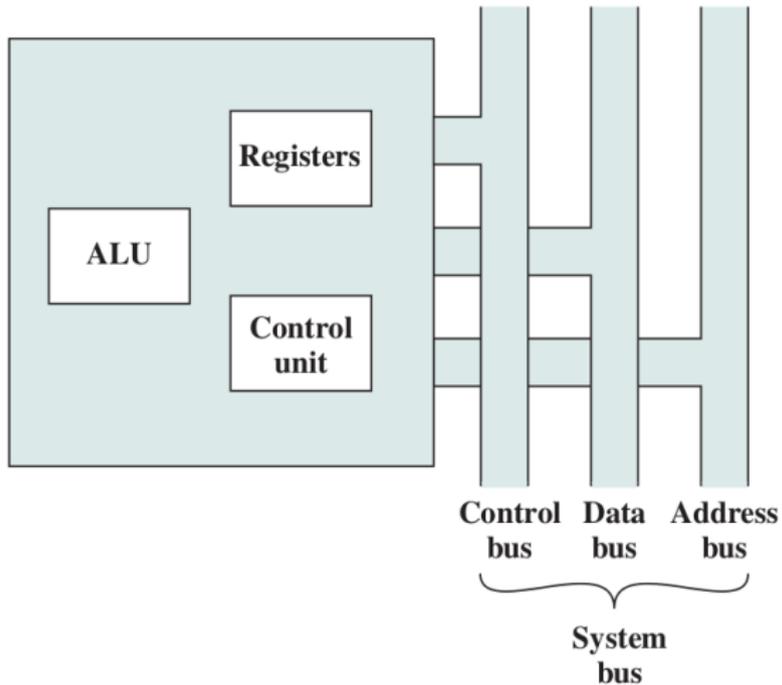
- 1 Introducción
- 2 Organización
  - Unidad aritmética y lógica
  - Registros
  - Unidad de control
- 3 Unidad de control
  - Introducción
  - Control cableado vs microprogramado
  - Ciclo de instrucción
  - Ejemplo de CU microprogramada

# Estructura del sistema computador

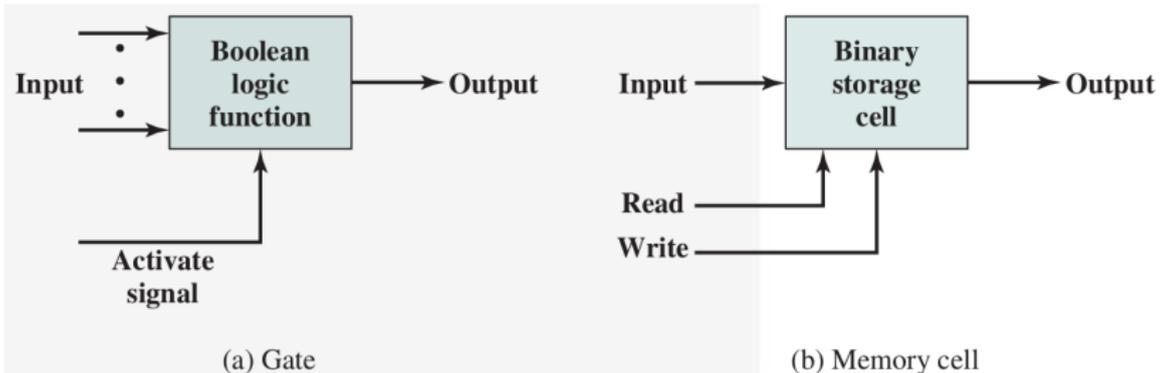




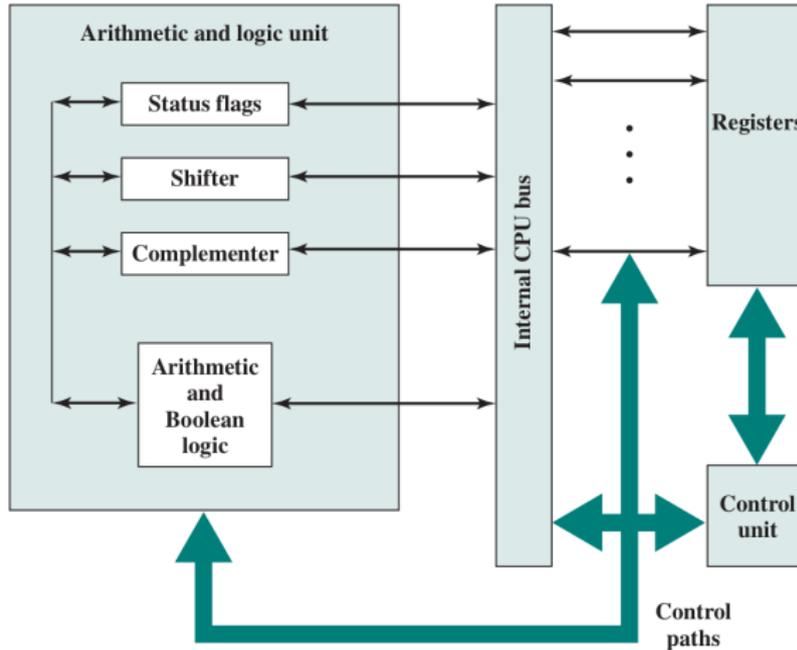
# El CPU y el bus del sistema



# Elementos fundamentales de una computadora



# Organización del CPU



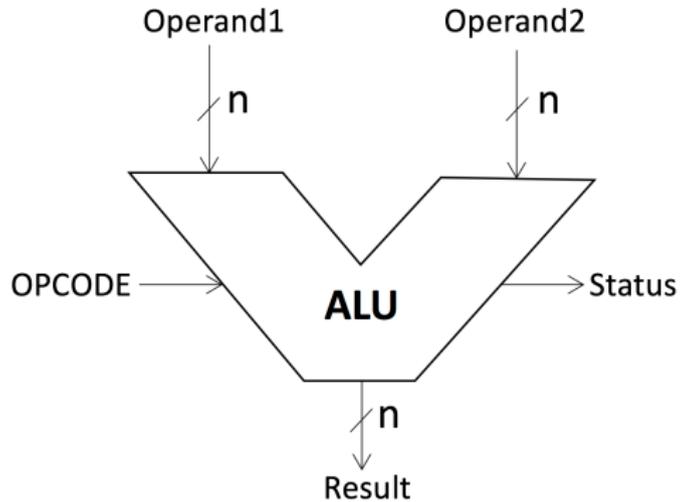
# Unidad aritmética y lógica

- ALU
- Realiza operaciones aritméticas y lógicas.
- *Circuito combinatorio*. ¿Multiplicación iterativa?

# Unidad aritmética y lógica

- Circuito lógico que implementa operaciones (aritméticas y lógicas).
- Utiliza como “representaciones internas” binario o complemento a 2 (o BCD).
- Operaciones aritméticas: suma, resta, multiplicación y división.
- Operaciones lógicas (a nivel de bit): and, or, xor, not, shift y rotate.

# Unidad aritmética y lógica



# Registros

- Memoria rápida
- Memoria ubicada físicamente dentro de la propia CPU.
- Sin penalizaciones.
- Acceso veloz a operandos.
- Algunos registros son de uso propio de la CPU y otros son accesibles por el programador.
- No es un elemento esencial, si no tecnológico de la arquitectura.

# Registros

## Categorías

- Propósito general (ortogonalidad).
- Datos.
- Direcciones.
- Estado y control.

# Registros

## Categorías (otra)

- Visibles
- Internos
- (Parcialmente visibles).

# Registros

## Internos

- PC (Parcialmente visible).
- IR.
- MAR y MBR.
- PSR (Parcialmente visible ).

## Banco (Set / Conjunto) de Registros

- Los registros se organizan en bancos.
- Serie de posiciones especiales de memoria.
- 8086 organiza los registros en dos bancos, uno de uso general y otro para segentos.
- Sparc los organiza en ventanas.
- Z80 usa un banco principal y otro alternativo.

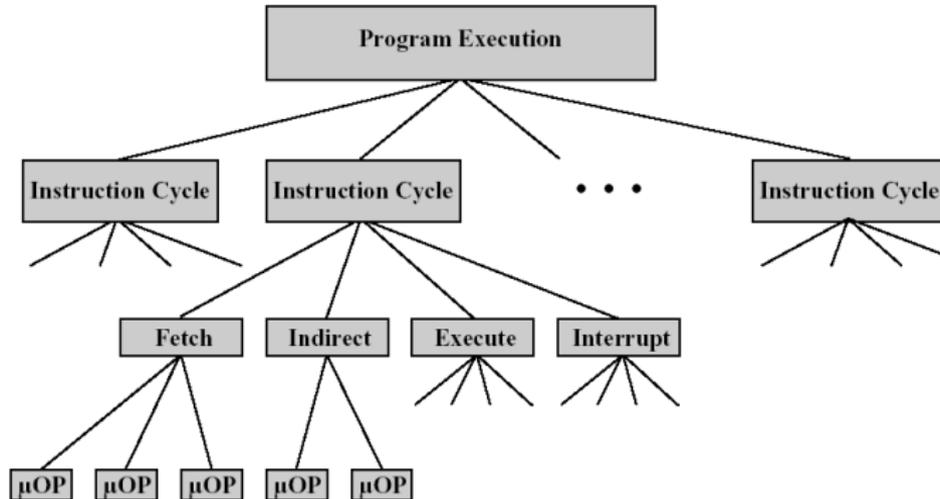
# Unidad de control

- Circuito lógico que implementa el **ciclo de instrucción**.
- *Circuito secuencial*.
- Determina:
  - Acceso a la siguiente instrucción.
  - Leer los operandos.
  - Efectuar la operación indicada.
  - Controlar la ALU.
  - Almacenar el resultado.

# Unidad de control

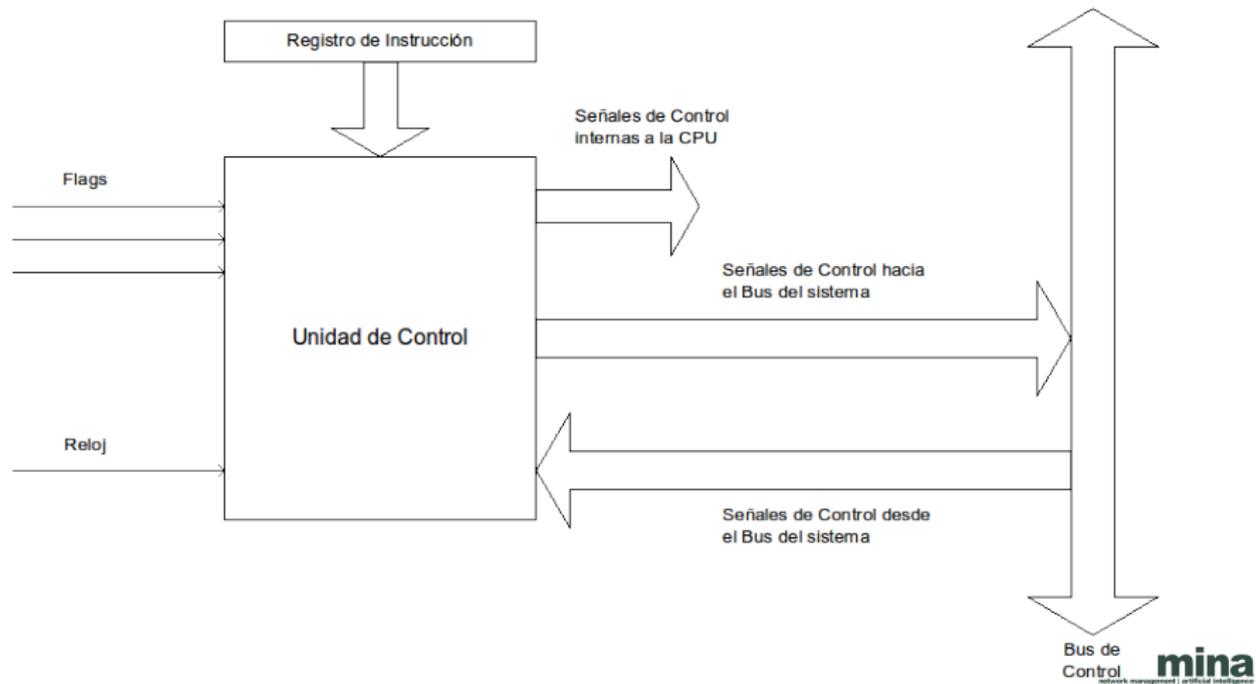
- Responsable de ejecutar el ciclo de instrucción.
- Control
  - Cableado
  - Microprogramado

# Ejecución de un programa



# Unidad de control

## Interfaz de la CU



# Unidad de control

## Control microprogramado

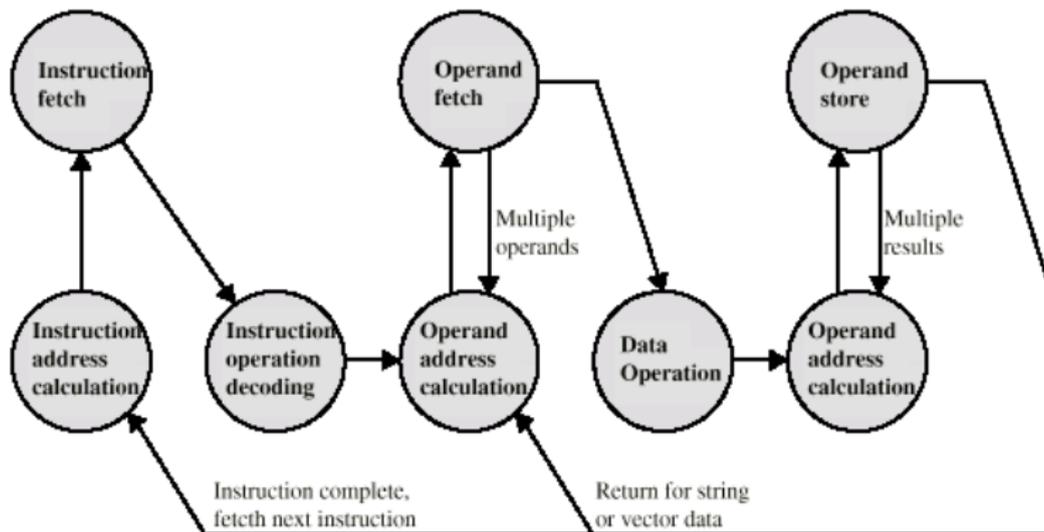
- Solución en software del control de la máquina.
- Un control microprogramado se puede ver como un “computador” interno que implementa el ciclo de instrucción de la CPU.
- Para cada “macroinstrucción” existe una secuencia específica de microinstrucciones que la implementa.
- El código (firmware) elemental necesario para implementar el ciclo de instrucción de la CPU se almacena en memoria de microprograma de tipo ROM.

# Unidad de control

## Control cableado

- Se implementa como un circuito secuencial.
- Para cada “macroinstrucción” existe una secuencia específica de microoperaciones que la implementa.
- Los sistemas cableados son rápidos pero pueden ser inflexibles.
  - Las funciones de control son fijas; si se necesita agregar instrucciones extra, el circuito debe rediseñarse
  - Los procesadores RISC usan control cableado para aumentar la velocidad
- Normalmente se usa una combinación de control cableado para instrucciones sencillas y microcódigo para instrucciones complejas y poco frecuentes

## Ciclo de instrucción



# Ciclo de instrucción

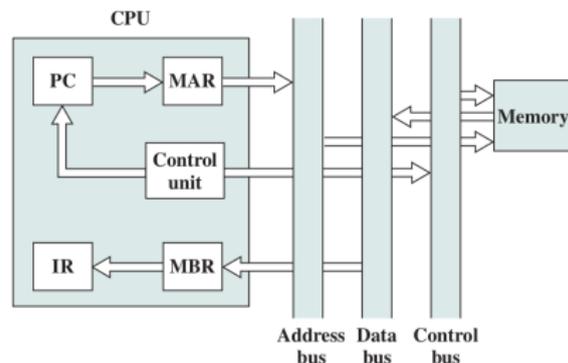
## Etapas

- Fetch.
- Decode.
- Read (memoria).
- Execute.
- Write.

# Ciclo de instrucción

## Etapa de fetch

- Microoperaciones:
  - 1:  $MAR \leftarrow PC$
  - 2:  $MBR \leftarrow memory$   
 $PC \leftarrow PC + 1$
  - 3:  $IR \leftarrow MBR$
- o también
  - 1:  $MAR \leftarrow PC$
  - 2:  $MBR \leftarrow memory$
  - 3:  $PC \leftarrow PC + 1$   
 $IR \leftarrow MBR$



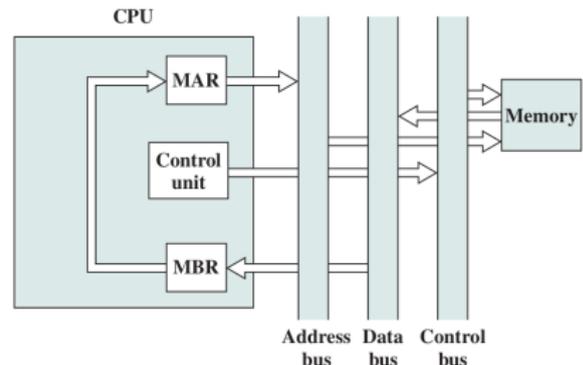
MBR = Memory buffer register  
MAR = Memory address register  
IR = Instruction register  
PC = Program counter

# Ciclo de instrucción

## Etapa indirecta

- Microoperaciones:

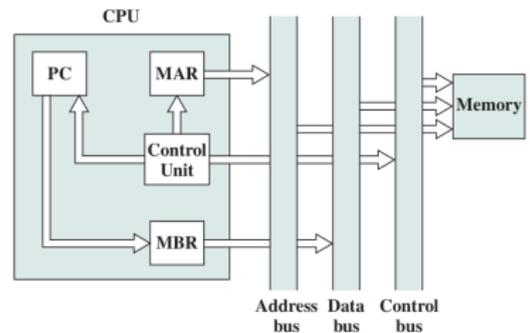
- 1:  $MAR \leftarrow IR(dirección)$
- 2:  $MBR \leftarrow memory$
- 3:  $MAR \leftarrow MBR$
- 4:  $MBR \leftarrow memory$



# Ciclo de instrucción

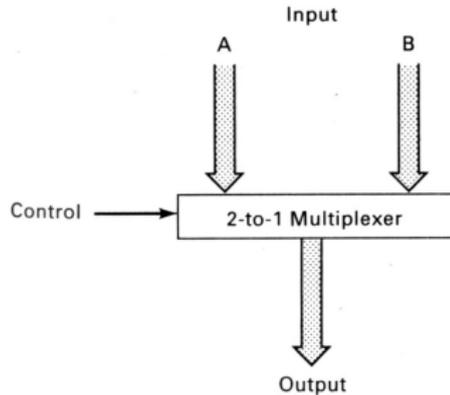
## Etapa interrupción

- Microoperaciones:
  - 1:  $MBR \leftarrow PC$
  - 2:  $MAR \leftarrow PC\_backup\_addr$   
 $PC \leftarrow handler\_addr$

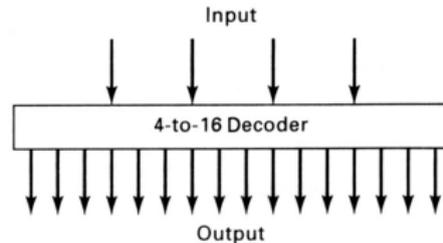


# Arquitectura microprogramada MIC 1

## Multiplexor y decodificador



(a)

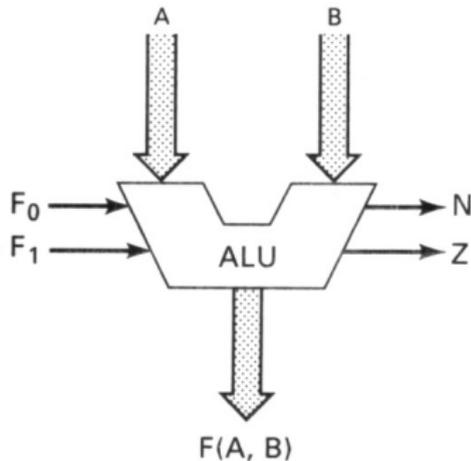


(b)

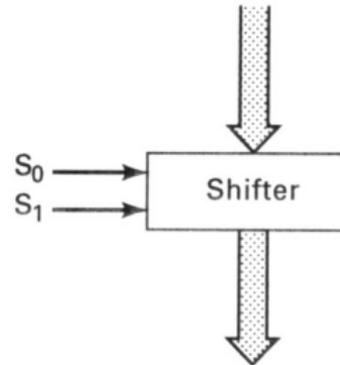
(a) A 2-to-1 multiplexer. (b) A 4-to-16 decoder.

# Arquitectura microprogramada MIC 1

## ALU y shifter



(a)

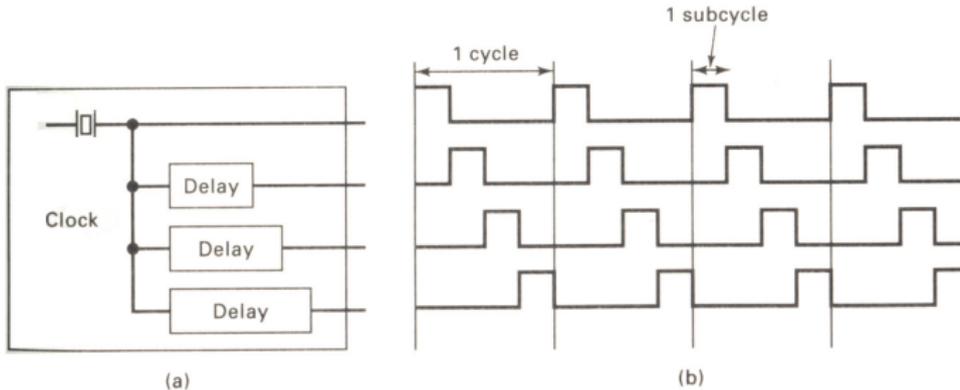


(b)

(a) An ALU. (b) A shifter.

# Arquitectura microprogramada MIC 1

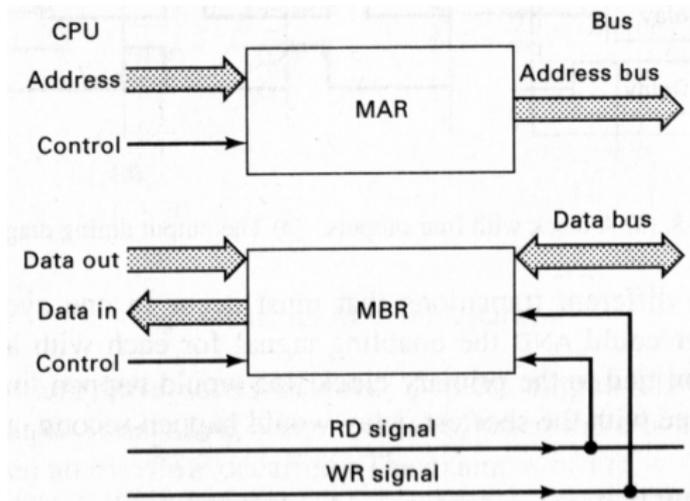
## Reloj y retardos



(a) A clock with four outputs. (b) The output timing diagram.

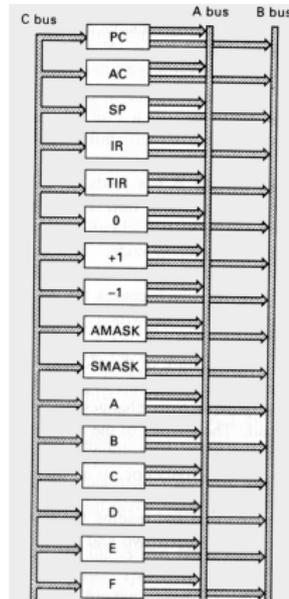
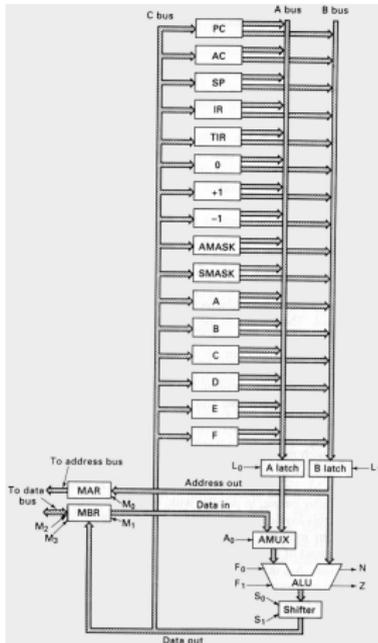
# Arquitectura microprogramada MIC 1

## Registros y memoria



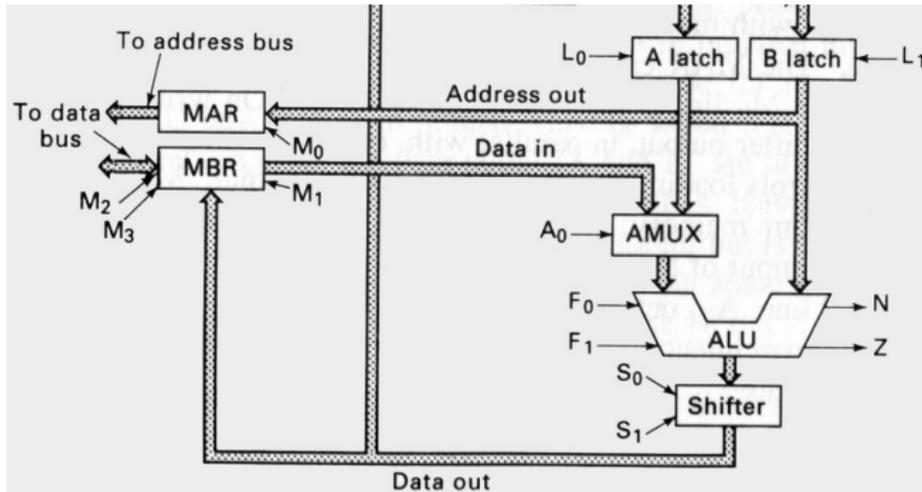
# Arquitectura microprogramada MIC 1

## Banco de registro y ruta de los datos



# Arquitectura microprogramada MIC 1

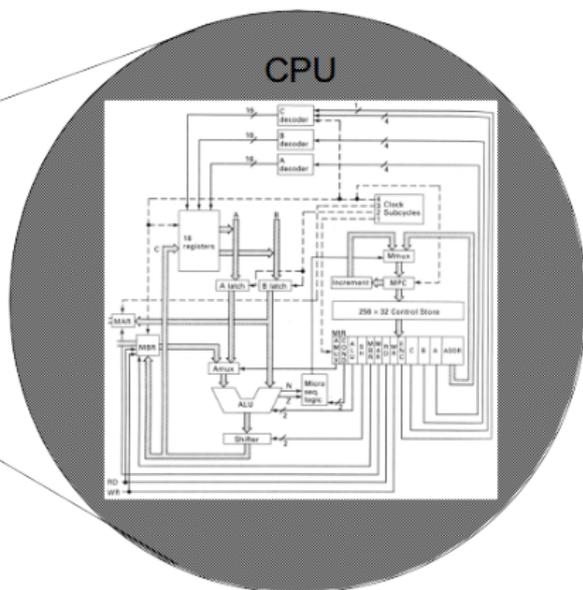
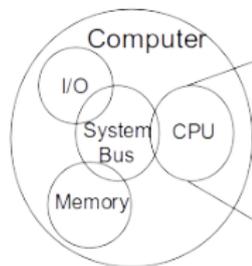
## Ruta de los datos





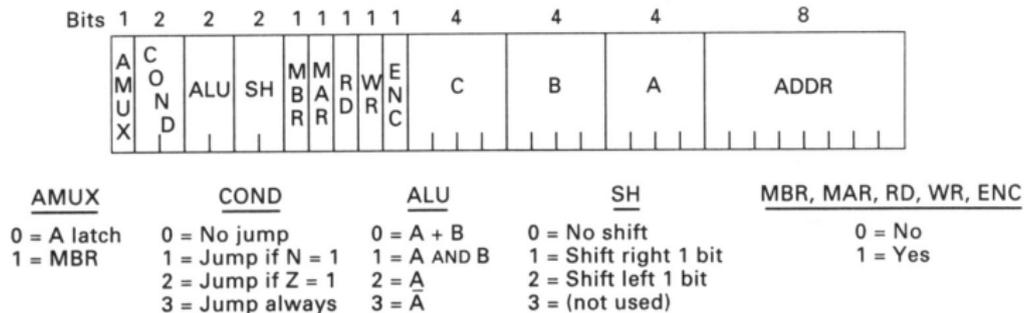
# Arquitectura microprogramada MIC 1

## Recapitulando



# Arquitectura microprogramada MIC 1

## Microinstrucción



# Microinstrucción

- Horizontal.
- Vertical.

# Arquitectura microprogramada MIC 1

## Microcódigo

- - Jump to Indirect or Execute	Fetch cycle routine
- - Jump to Execute	Indirect Cycle routine
- - Jump to Fetch	Interrupt cycle routine
Jump to Op code routine	Execute cycle begin
- - Jump to Fetch or Interrupt	AND routine
- - Jump to Fetch or Interrupt	ADD routine

# Lecturas Recomendadas



Departamento de arquitectura.

Organización del CPU.

*Notas de teórico. 2011.*



William Stallings.

Computer Organization and Architecture: Designing for Performance.

*Capítulos 14. 2010.*



OpenFing.

Video - Teórico [Ciclo de instrucción / Organización de la CPU].

*Unidad 11. 2024.*

# Preguntas