

## Tema 2

### Introducción a la arquitectura Von Neumann

Curso 2007-2008

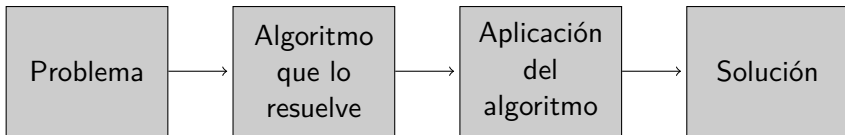
# Objetivos

- Comprender la diferencia entre máquina específica y programable.
- Conocer las necesidades de la máquina programable.
- Conocer la arquitectura ideada por Von Neumann.

# Esquema

- 1 Introducción
  - Máquinas para resolver problemas
  - Máquina genérica
  
- 2 Arquitectura Von Neumann
  - La arquitectura Von Neuman
  - El concepto de instrucción

# Planteamiento



## Ejemplo de algoritmo

### Problema

Encontrar el máximo de una serie de números.

### Algoritmo

- Inicializar  $\text{máximo} \leftarrow 0$
- Para cada número de la lista hacer:
  - Comparar número con máximo
  - Si  $(\text{número} > \text{máximo})$   $\text{máximo} \leftarrow \text{número}$

# Máquina específica

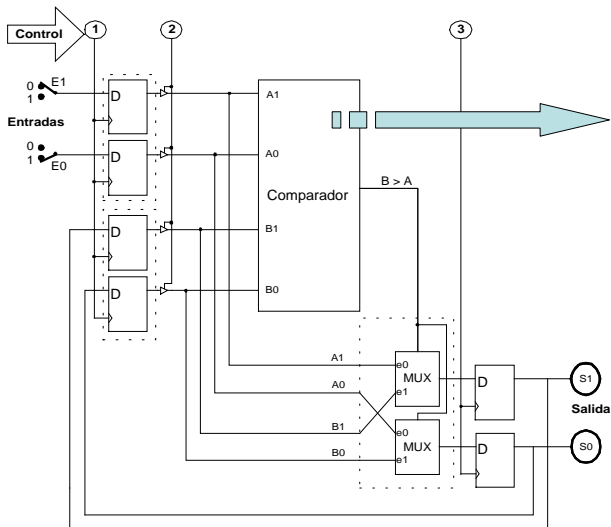


Tabla de verdad del COMPARADOR.

A1	A0	B1	B0	B > A
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

## Operación de la máquina

- ① Colocar los conmutadores de entrada en 0 ó 1 para representar el dato a introducir.
- ② Pulsar botón 1 para cargar los datos en los registros
- ③ Mantener pulsado el botón 2 mientras el comparador genera su salida.
- ④ Manteniendo pulsado el botón 2, pulsar el botón 3 para cargar el nuevo máximo en el registro de salida
- ⑤ Soltar el botón 2 y volver al paso inicial.

# Máquina genérica

## Objetivo

Construir una máquina genérica que pueda ejecutar cualquier algoritmo.

## Necesidades

- El algoritmo a ejecutar debe almacenarse de alguna forma, para poder cambiarlo por otro si se desea.
- Los datos que maneja el algoritmo también deben ser almacenados.
- Es necesario inventar una “maquinaria” que “lea” el algoritmo desde su almacén y lo “ejecute”
- Es necesario un mecanismo que permita introducir los datos (y los algoritmos) en la máquina, y que permita que los resultados sean volcados al exterior.



# Solución

## Tecnología

Electrónica digital (a día de hoy)

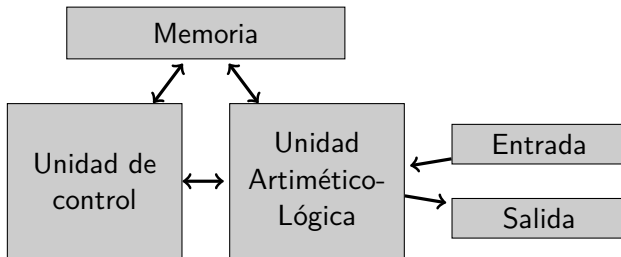
## Soluciones a las necesidades

- **Almacén (código y datos)**  
Memoria digital
- **Maquinaria de ejecución**  
Unidad de Control + Unidad Aritmético-Lógica = CPU
- **Comunicación con exterior**  
Dispositivos de Entrada/Salida (E/S) y periféricos.

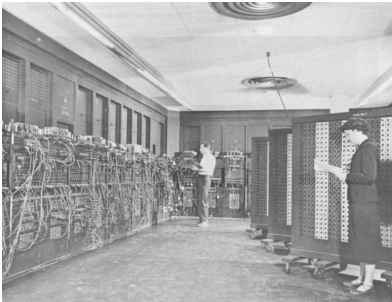
**NOTA:** Para poder ser almacenados en una memoria digital, ***datos y algoritmos*** deben codificarse como masas de bits

# Origen

- La *arquitectura Von Neumann* aparece en un artículo en 1945.
- **Idea clave**  $\Rightarrow$  *programa almacenado* en la misma memoria que los datos.
- Idea también desarrollada por Mauchly y Eckert.
- Base del primer computador programable: EDVAC.
- Aún en uso, con variaciones, en prácticamente todos los computadores.



## Máquina específica/Máquina programable



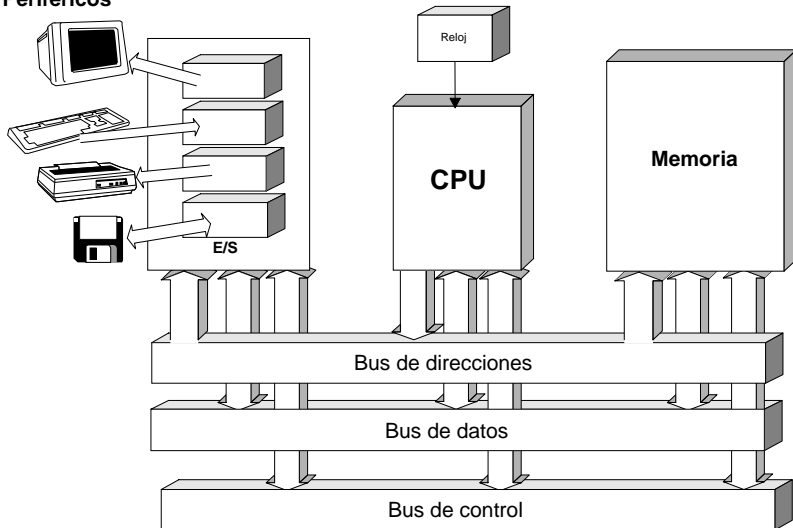
ENIAC



EDVAC

# Moderna arquitectura Von Neumann

## Periféricos



## Concepto clave

- Una instrucción es una operación elemental que la CPU “sabe” hacer por *hardware*.
- El número de instrucciones que posee una CPU es finito.
- La secuencia en que se ejecuten las instrucciones da lugar a diferentes comportamientos.

## Codificación de algoritmos

- Ya que el número de instrucciones es finito, se puede asignar un código diferente a cada una.
- Un algoritmo se convierte en una secuencia de códigos
- Por tanto se puede almacenar como una masa de bits.