

Sección 1 – Parte 2ª

# ARQUITECTURA HARDWARE EQUIPOS INFORMÁTICOS Y DE TELECOMUNICACIONES ESTRUCTURA S.I.

# Estructura de sistemas informáticos

Un sistema informático como todo sistema, es el **conjunto de partes interrelacionadas, hardware, software y de recurso humano** que permite **almacenar y procesar información**.

El **hardware** incluye **computadoras o cualquier tipo de dispositivo electrónico inteligente, que consisten en procesadores, memoria, sistemas de almacenamiento externo, etc.**

**El software incluye al sistema operativo, firmware y aplicaciones.**

Por último el **soporte humano** incluye al **personal técnico** que crean y mantienen el sistema (analistas, programadores, operarios, etc.) y a los **usuarios** que lo utilizan.

wikipedia

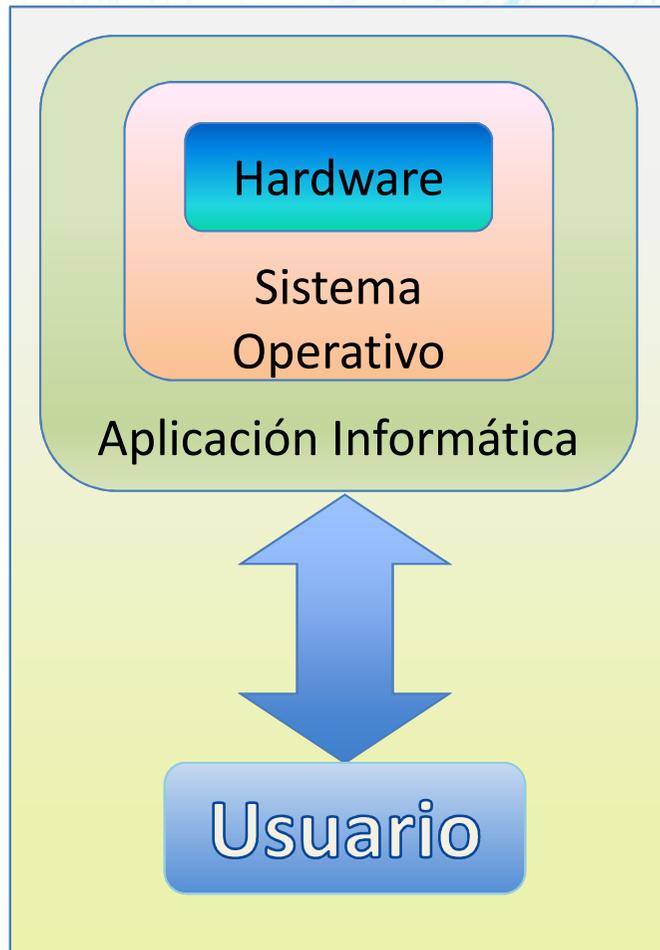
Un **sistema informático** se compone al menos, de una **unidad central de proceso** (CPU) y los **periféricos** necesarios para **'capturar'** datos y dar **'salida'** a los datos **'procesados'** por la CPU.

Los sistemas **periféricos están comunicados** con la CPU y pueden estar **integrados en el mismo equipo** o pueden ser **externos** y estar conectados al sistema, incluso **a distancia** de kilómetros.

Ejemplo: CPU/memoria/disco duro/teclado <-> impresora <-> servidor de datos remoto

**Ver más ejemplos y comentar tipos de conexión / comunicación**

# Estructura de sistemas informáticos



- **Hardware**

- ✓ Componentes internos
- ✓ Periféricos
- ✓ Componentes externos
- ✓ Etc.

- **Software**

- ✓ Sistema Operativos
- ✓ Drivers
- ✓ Software de aplicación
- ✓ Etc.

- **Personal**

- ✓ Usuario/s del sistema
- ✓ Mantenimiento
- ✓ Programadores, analistas, etc.

Documentación: Toda la documentación necesaria para la gestión y buen funcionamiento de las partes antes mencionadas.

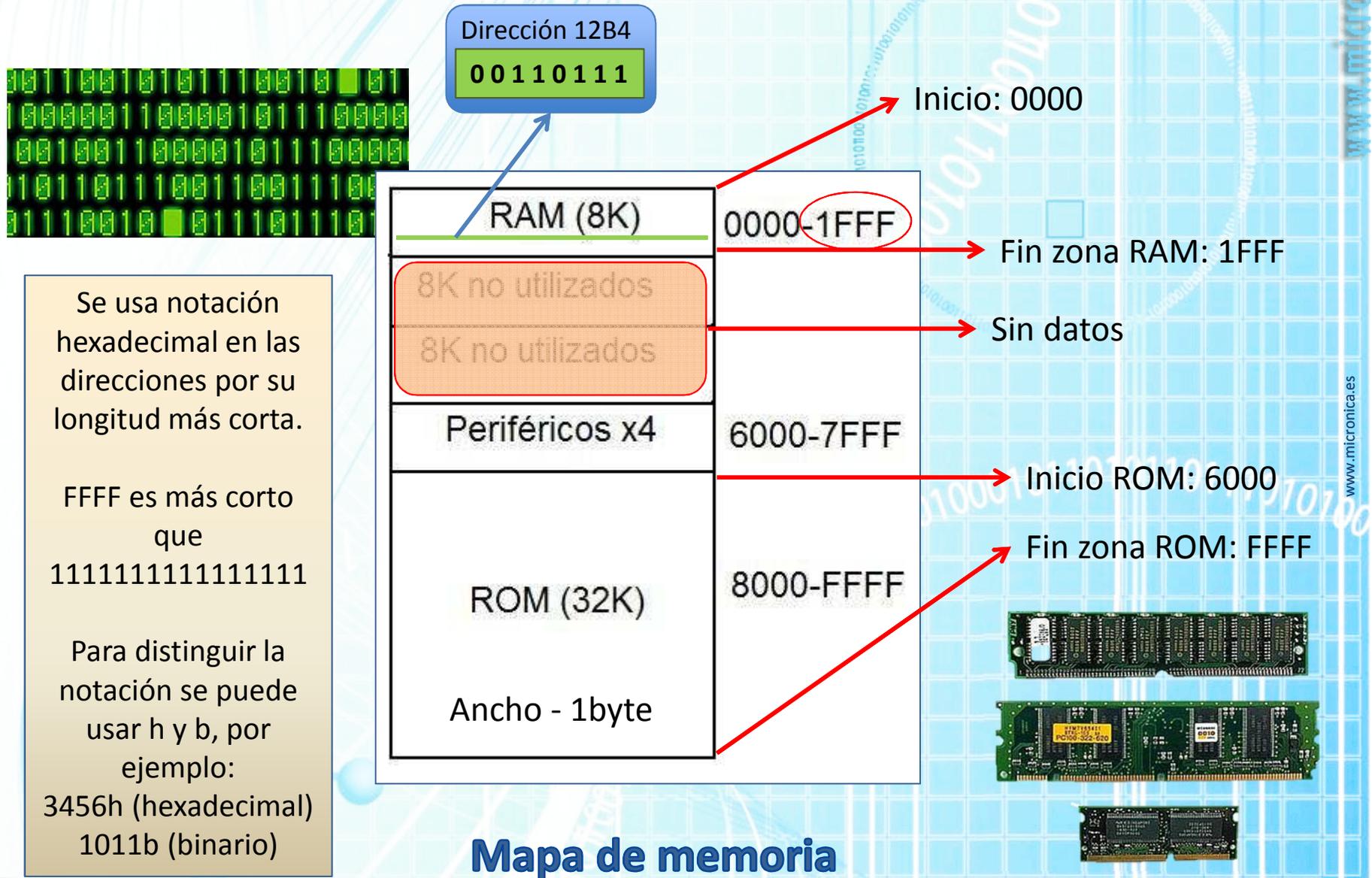
## Sistemas Informáticos

- Según arquitectura funcional (funcionamiento y servicios)
- Según arquitectura física (hardware que lo componen)

# Estructura de sistemas informáticos

- **Los sistemas informáticos no trabajan con letras, números como nosotros:**
  - ✓ Trabajan con codificación binaria (númeración en base 2, *solo con '1' y '0'*)
  - ✓ Toda la información se pasa a números compuestos por combinaciones de '1s' y '0s'
  - ✓ Textos, imágenes. Sonidos, videos: *todo se ha de convertir en '1s y 0s'*.
  - ✓ Una vez convertidos se pueden realizar operaciones de todo tipo.
- **Datos y su almacenamiento**
  - ✓ La información necesaria para procesar la información requiere ser almacenada.
  - ✓ En algunos casos solo unos instantes y en otros largo tiempo.
  - ✓ En ambos casos, *se almacenan los '1s y 0s' correspondiente a los datos.*
- **Unidades de datos**
  - ✓ **Bit:** Unidad mínima de información, solo puede contener un valor '0' o '1'
  - ✓ **Byte:** Es el agrupamiento de 8 bits, por ejemplo '10011100' es un byte.
  - ✓ Para trabajar de forma más simple, muchas veces, se pasa de binario a hexadecimal
    - 11001001 en binario, se expresa como C9 en hexadecimal.
  - ✓ También se puede expresar el tamaño de un dato por el nº de bits: 32bits
  - ✓ Para expresar la cantidad de datos, se emplean múltiplos diferentes a los habituales:
    - 1Kilobit = 1024 bits, 1 kilobyte = 1024 bytes (**1 kilo NO es 1000 en este caso**)
    - Abreviaturas: 1Kb -> Kilobit, 1KB -> KiloByte (observar diferencia).
  - ✓ 1 **Mega** = 1024 Kilo, 1 **Tera** = 1024 Megas
  - ✓ 1024 tiene su origen en  $2^{10}$ , estamos trabajando en binario (base 2).

# Estructura de sistemas informáticos



Se usa notación hexadecimal en las direcciones por su longitud más corta.

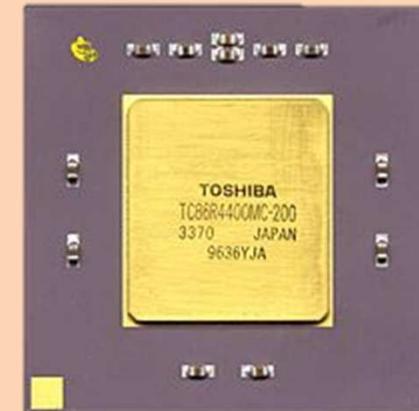
FFFF es más corto que 1111111111111111

Para distinguir la notación se puede usar h y b, por ejemplo:  
3456h (hexadecimal)  
1011b (binario)

**Mapa de memoria**

# Estructura de sistemas informáticos

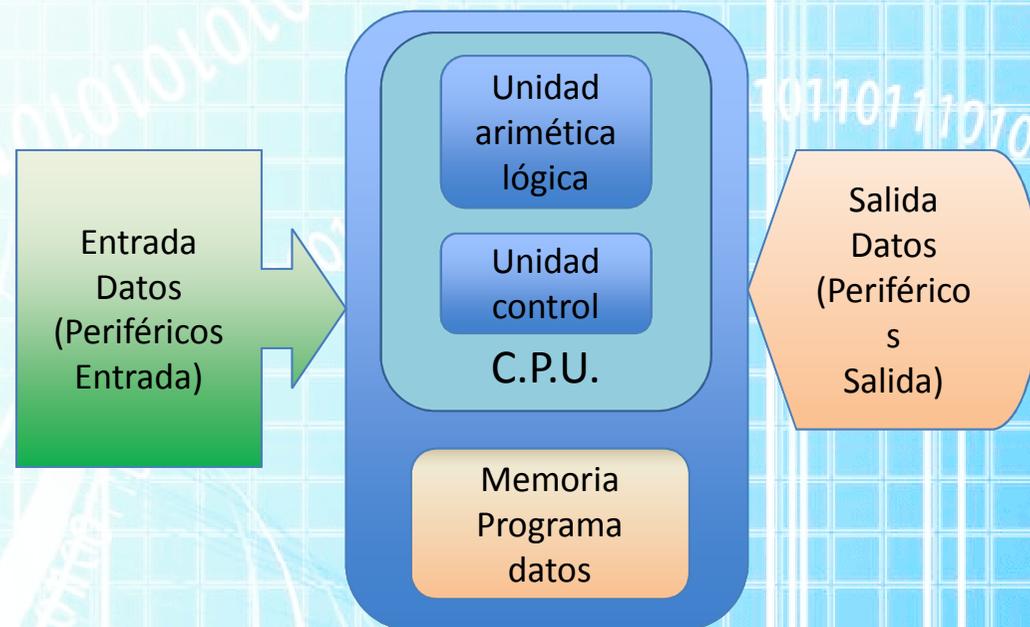
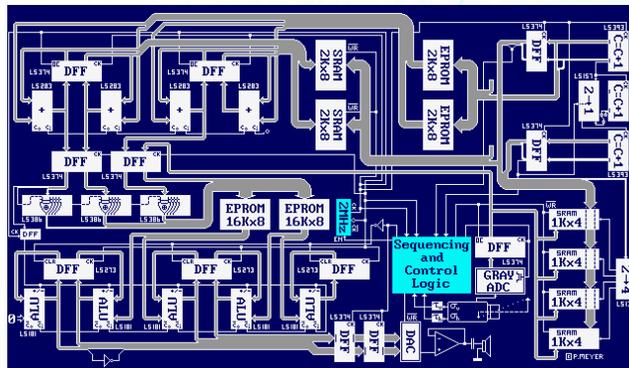
- **Las operaciones básicas de los procesadores dentro de un S. Informático son:**
  - ✓ Aritméticas básicas, sumas y restas. Muchos procesadores no multiplican ni dividen. *¿??*
  - ✓ Operaciones lógicas booleanas (álgebra de Boole) del tipo 'Y', 'O', 'No' y 'O exclusivo'.
    - *Se conocen por su traducción al inglés And, Or, Not y XOr*
  - ✓ Operaciones de salto condicional e incondicional:
    - Si se **cumple una condición** el programa se bifurca, como un cruce de caminos.
    - Si **no se cumple la condición** toma por otro camino (**decisiones**).
    - Los saltos incondicionales implican ir a un 'punto del programa' **sin condiciones**.
  - ✓ Adquisición de datos y almacenamiento en 'registros' (memoria) para procesarlos:
    - Datos de teclado
    - Posición del ratón
    - Entrada de video
    - Medida de temperaturas
    - Leer datos de un disco duro, pen drive- DVD, etc.
    - Etc.
  - ✓ Envío de datos o resultados a otros periféricos del sistema:
    - Presentar algo en pantalla.
    - Enviar petición a Internet de página Web.
    - Imprimir un documento.
    - Enviar para guardar en disco duro, pen drive, etc.
    - Producir un sonido.
  - ✓ Otras muchas específicas del **juego de instrucciones de cada procesador**.



# Arquitectura S.I.

- Definiciones clásicas:
  - Aquellas características del computador **visibles al nivel de programación en ensamblador o diseño de compiladores**
  - “La **apariencia funcional** que presenta a sus usuarios inmediatos”. (Amdahl, 1964). Describe “lo que sucede”

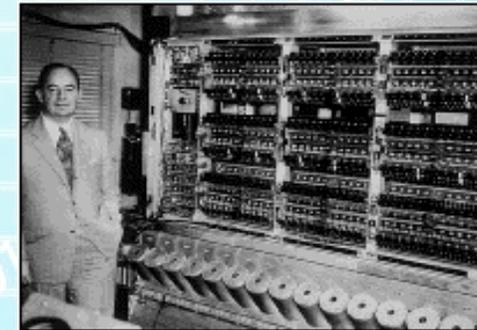
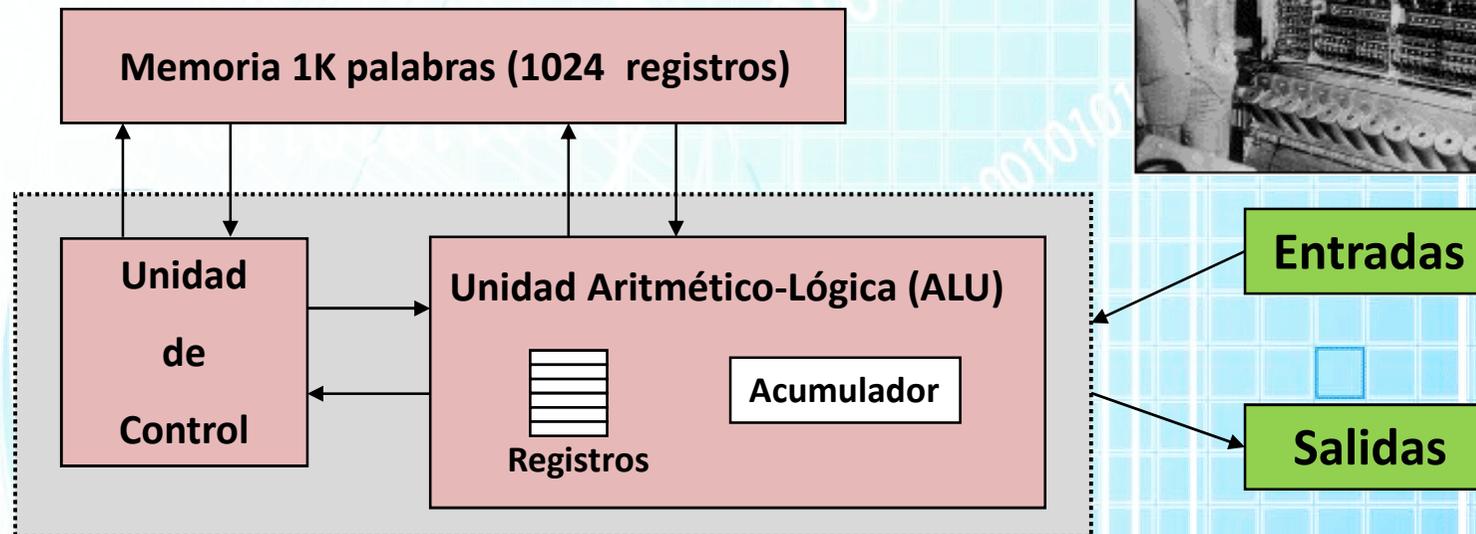
¿A qué se refieren estas definiciones?



# Arquitectura S.I.

## Arquitectura Von Neumann (John Von Neumann)

- Implantada en 1945 → se sigue con modificaciones.
- ❖ Memoria unidimensional, vector de celdas del mismo tamaño y de direcciones secuencial.
- ❖ Una misma memoria para instrucciones y datos (nº de celdas o posiciones limitado).
- ❖ Sin distinción explícita entre celdas de instrucciones y celdas datos.
- ❖ Sin especificación explícita de tipos de datos (programa o datos).
- ❖ Las instrucciones se ejecutan secuencialmente. Se requieren instrucciones de salto para romper el flujo de control.
- ❖ Camino de datos (ALU) que transforma los datos
- ❖ Registros para cálculos intermedios asociados con la ALU
- ❖ Unidad Control con solo un registro asociado, el PC (Program Counter).



# Arquitectura S.I.

## Otras arquitecturas: Harvard

### Arquitectura von Neumann

- La memoria de programa y la memoria de datos se hallan sobre un único espacio de memoria.
- Al espacio único de memoria se accede mediante un único bus de direcciones y un único bus de datos/instrucciones.
- Es la arquitectura típica de los grandes microcomputadores (ordenadores personales).
- También se encuentra en algunos microcontroladores (por ejemplo, NEC).



### Arquitectura Harvard

- La memoria de programa y la memoria de datos constituyen dos espacios de memoria separados.
- El acceso a cada espacio de memoria puede llegar a ser mediante buses distintos, es decir, puede haber dos buses de direcciones, un bus de datos y un bus de instrucciones).
- Lo más habitual es que exista un único bus de direcciones, en cuyo caso debe existir alguna señal de control que permita diferenciar a que espacio de memoria se hace referencia (por ejemplo, señales de READ, WRITE y FETCH).
- Es la arquitectura propia de un buen número de microcontroladores (por ejemplo, PIC).

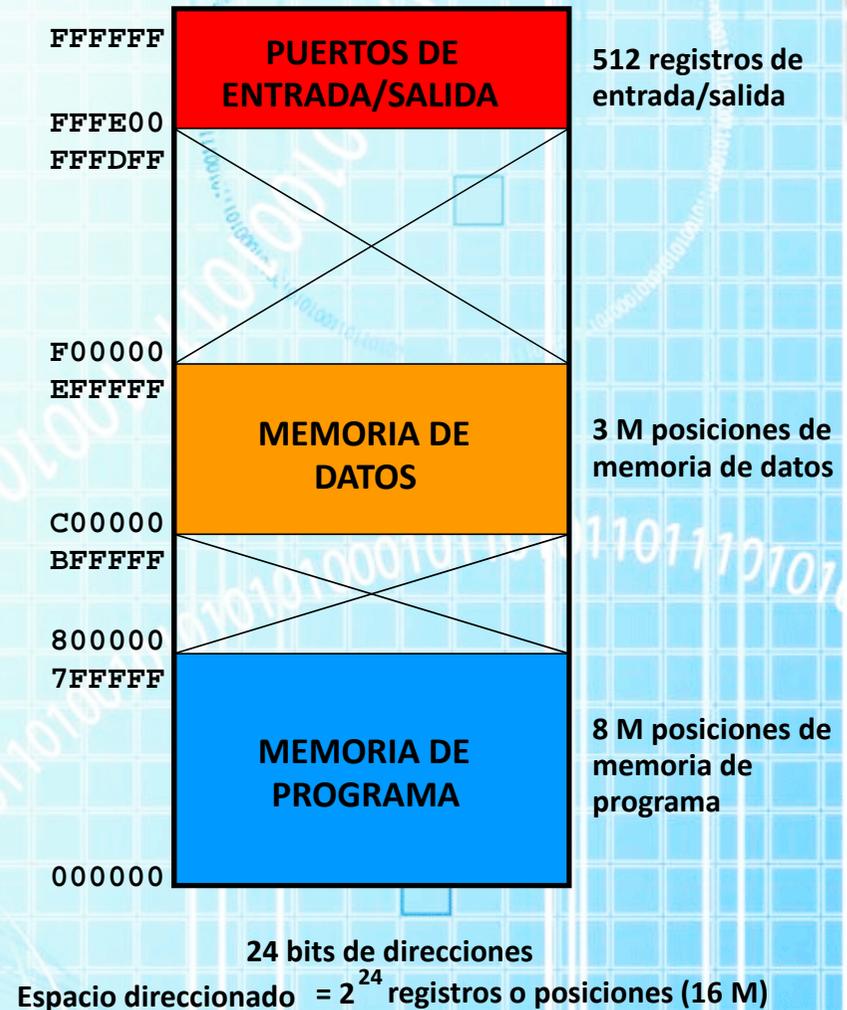


# Arquitectura S.I.

## Von Neumann básica

### Un único espacio de direccionamiento

- La *memoria de programa*, la *memoria de datos* y los puertos de *entrada/salida*, comparten un único espacio de direccionamiento.
- Basta con señales de control de READ y WRITE.
- El espacio de direccionamiento se reparte de acuerdo a las necesidades de la aplicación, no siendo necesario llenar todo el espacio de direccionamiento.
- Típico de los microprocesadores de alta gama de Motorola (68000 y posteriores).

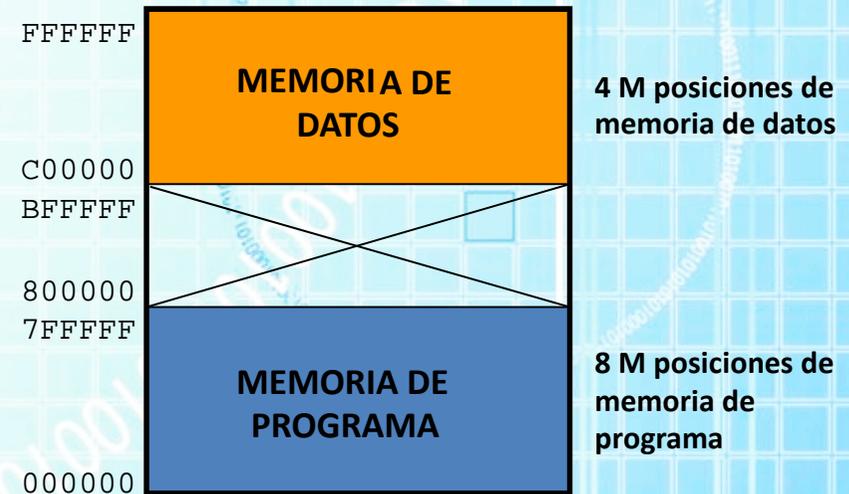


# Arquitectura S.I.

## Von Neumann (variante)

### Dos espacios de direccionamiento

- Uno para la *memoria de programa* más la *memoria de datos* y otro para los *puertos de entrada/salida*.
- Si se comparte el bus de direcciones, es necesario *añadir una señal de control* que permita distinguir a que espacio se hace referencia (por ejemplo, IO/M).
- Los dos espacios de direccionamiento se reparten de acuerdo a las necesidades de la aplicación, no siendo necesario llenarlos del todo.
- Típico de los microprocesadores de alta gama de Intel (8086 y posteriores).



24 bits de direcciones para memoria  
Espacio direccionado =  $2^{24}$  posiciones (16 M)



12 bits de direcciones para entrada/salida  
Espacio direccionado =  $2^{12}$  registros (4 K)

# Arquitectura S.I.

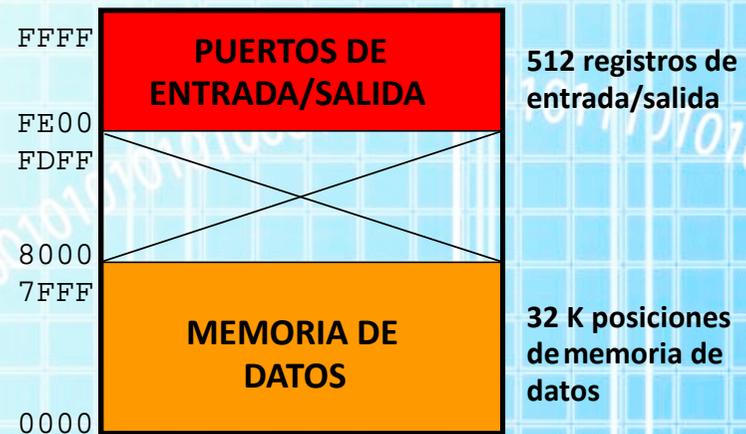
## Harvard

### Dos o tres espacios de direccionamiento

- Pueden existir hasta tres espacios de direccionamiento, uno para la *memoria de programa*, otro para la *memoria de datos* y el tercero para los puertos de *entrada/salida*.
- Es frecuente que la memoria de datos y los puertos de entrada/salida compartan un mismo espacio de direccionamiento.
- Si se comparte el bus de direcciones, es necesario añadir señales de control que permitan distinguir a que espacio se hace referencia.
- Los espacios de direccionamiento se reparten de acuerdo a las necesidades de la aplicación, no siendo necesario llenarlos del todo.
- Típico de muchos microcontroladores (por ejemplo, 8051 de Intel, derivados y PIC).



16 bits de direcciones para memoria de programa  
Espacio direccionado =  $2^{16}$  posiciones (64 K)



16 bits de direcciones para memoria de datos  
más entrada/salida  
Espacio direccionado =  $2^{16}$  registros (64 K)

# Arquitectura S.I. : Buses

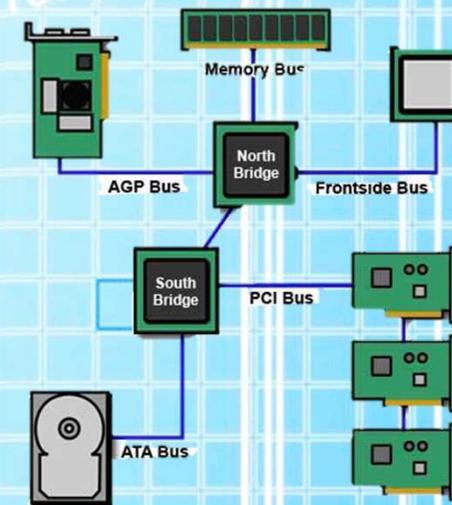
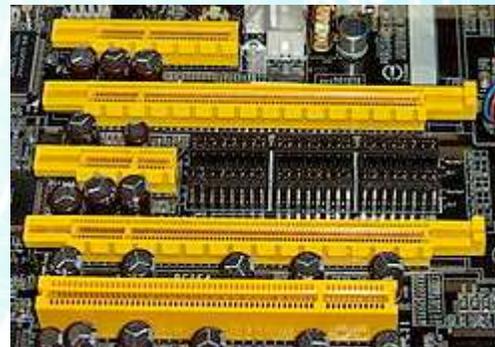
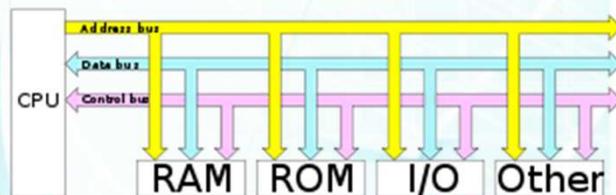
## ¿Qué es un BUS?

Se denomina **bus**, en informática, al conjunto de **conexiones físicas** (cables, placa de circuito impreso, etc.) que pueden **compartirse con múltiples componentes de hardware** y que sirven **para que se comuniquen entre sí**.

El propósito de los buses es reducir el número de rutas necesarias para la comunicación entre los distintos componentes, al realizar las comunicaciones a través de una sola vía de datos. Ésta es la razón por la que, a veces, se utiliza la metáfora "**autopista de datos**".

<http://es.kioskea.net/contents/364-que-es-un-bus-informatico>

- En general los buses son varias conexiones en **paralelo** formando un **grupo de 'n' líneas o bits**:
  - ✓ Buses de datos, direcciones, control, etc.
  - ✓ Bus ATA de conexión de disco duro.
  - ✓ Front Side Bus (FSB de los PCs).
  - ✓ Etc
- Cada vez se usan más buses '**serie**', caracterizados porque solo hay **una línea de conexión (un solo bit)**:
  - ✓ Conexiones USB.
  - ✓ Discos duros SATA.
  - ✓ 'Lanes' PCI-Express.
  - ✓ Etc.



# Arquitectura S.I. : Buses

La arquitectura de un microcomputador basado en un microprocesador más dispositivos de memoria y de entrada/salida se basa normalmente en tres buses (conjuntos de conexiones eléctricas entre zonas):

- **Bus de direcciones**

- Común para todos los espacios de direccionamiento.
- Dimensión establecida por el espacio de direccionamiento mayor.
- Generado por el microprocesador.

- **Bus de datos (datos/instrucciones)**

- Común para todos los espacios de direccionamiento.
- Dimensión establecida generalmente por la unidad de proceso.
- Bidireccional.

- **Bus de control**

- Incluye las señales de control generadas por el microprocesador para acceder a los espacios de direccionamiento (READ, WRITE, FETCH, IO/M, DMA\_ACK, ...).
- Puede incluir señales destinadas al microprocesador (READY, DMA\_REQ, ...).