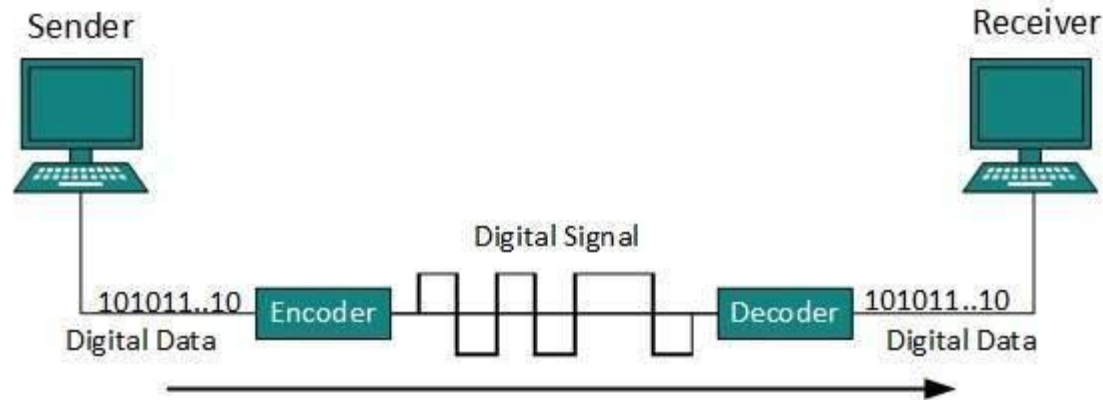




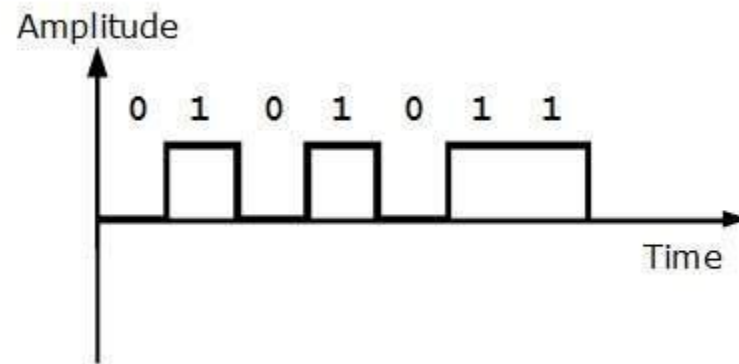
Arquitecturas de las computadoras

Intro a Transmisión digital

Codificación de línea

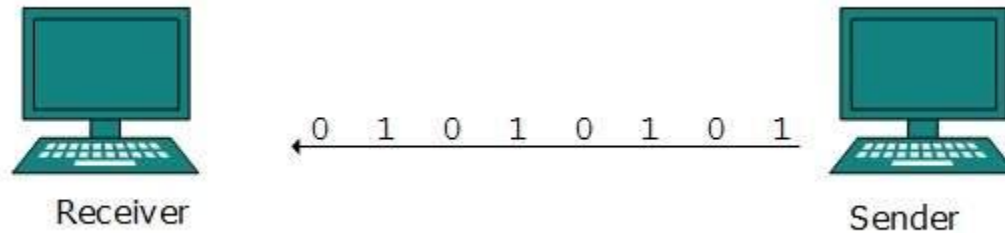


Codificación unipolar

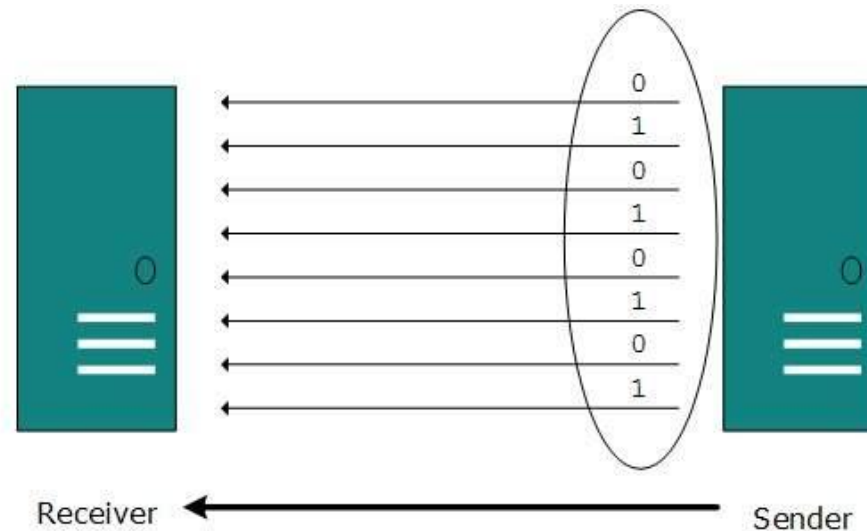


Intro a Transmisión digital

Transmisión serie



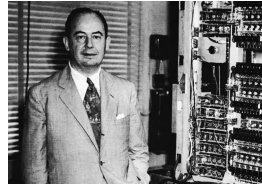
Transmisión paralela



¡Bus !

Tipos de Arquitectura

Von Neumann



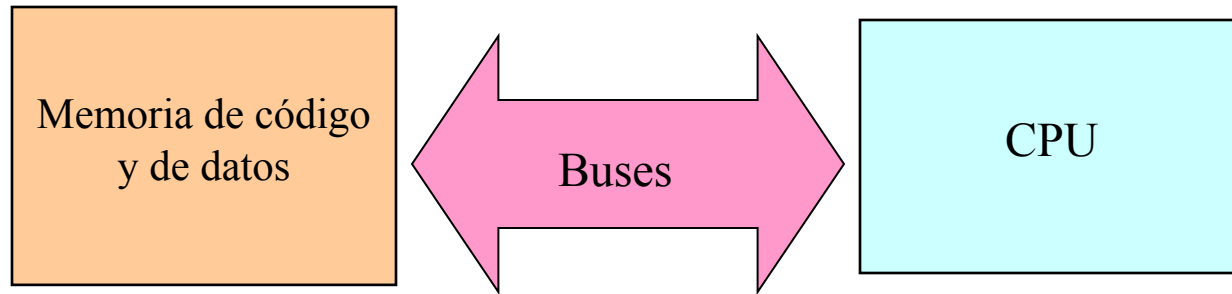
Direcciones

0000h

0001h

....

nnnn



Harvard



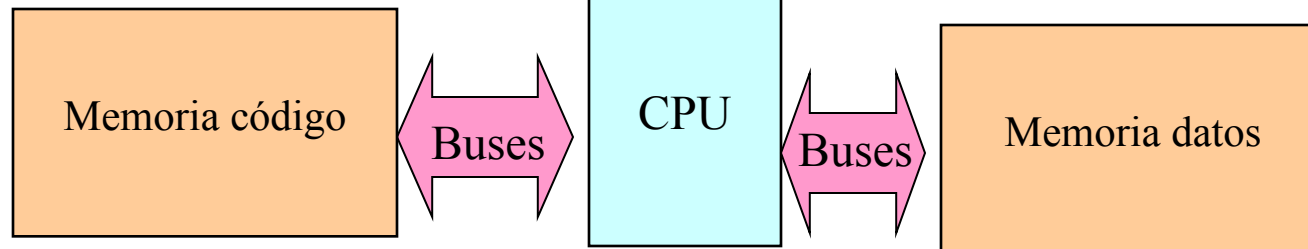
Direcciones

0000h

0001h

....

nnnn



Direcciones

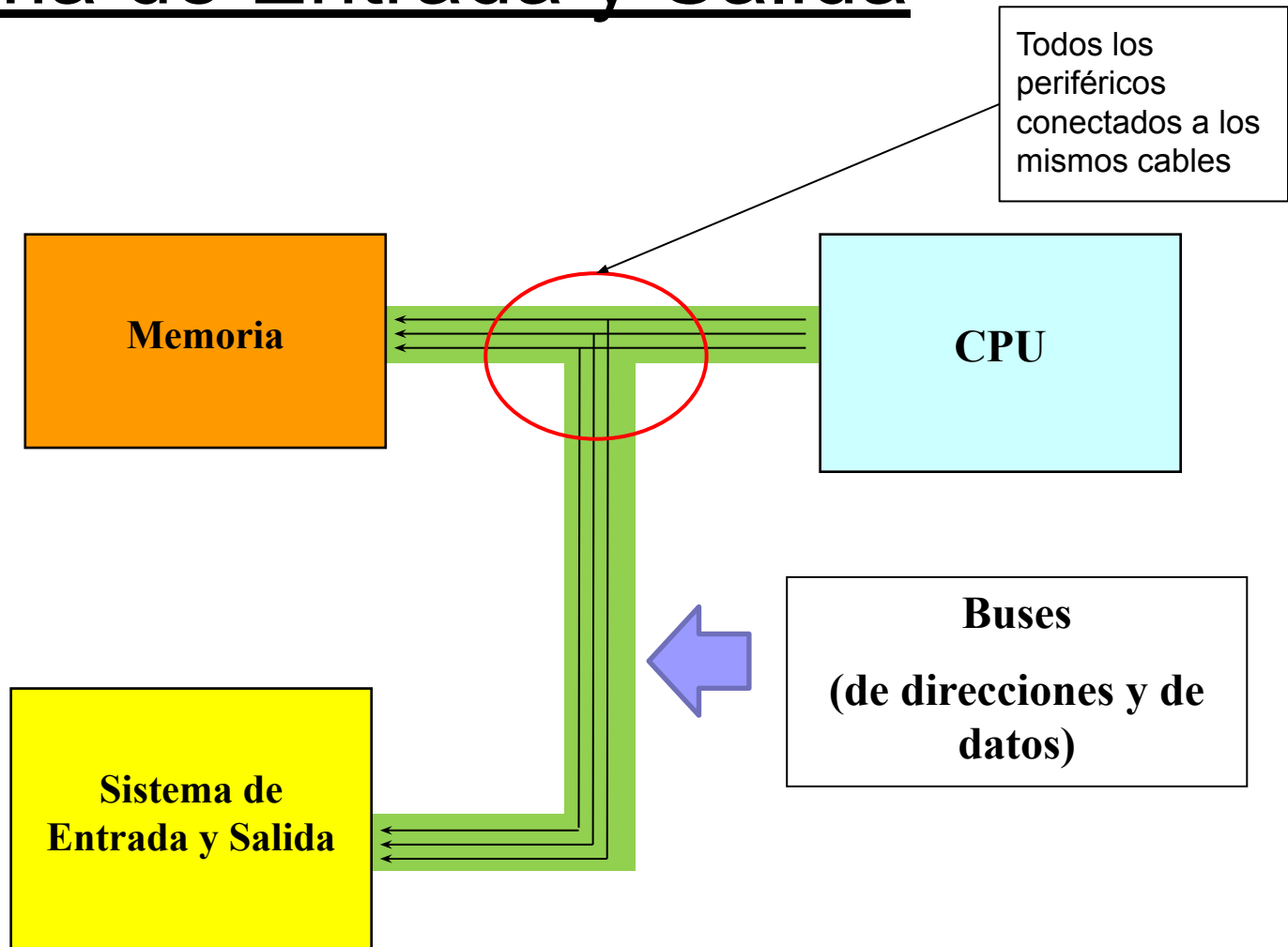
0000h

0001h

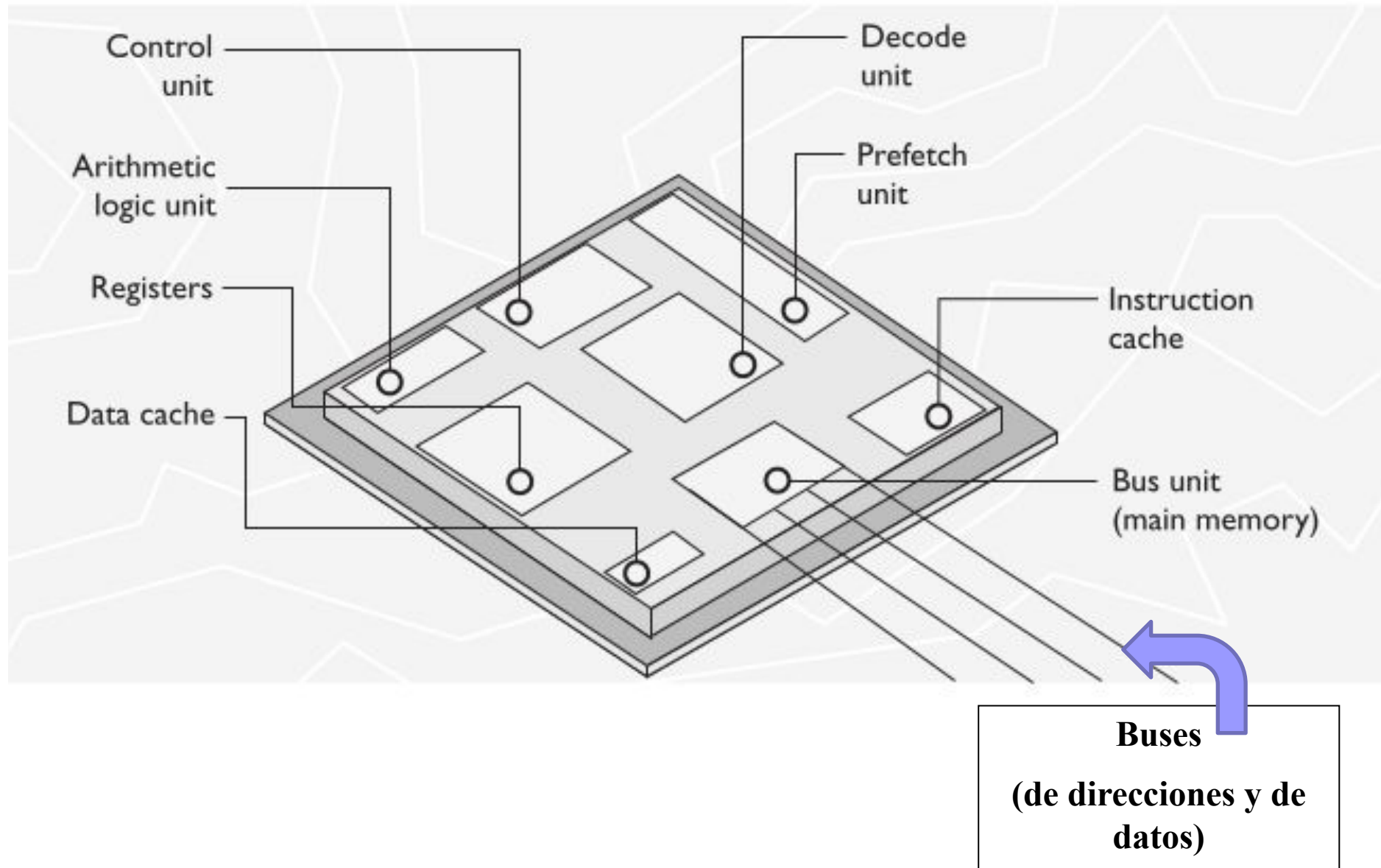
....

nnnn

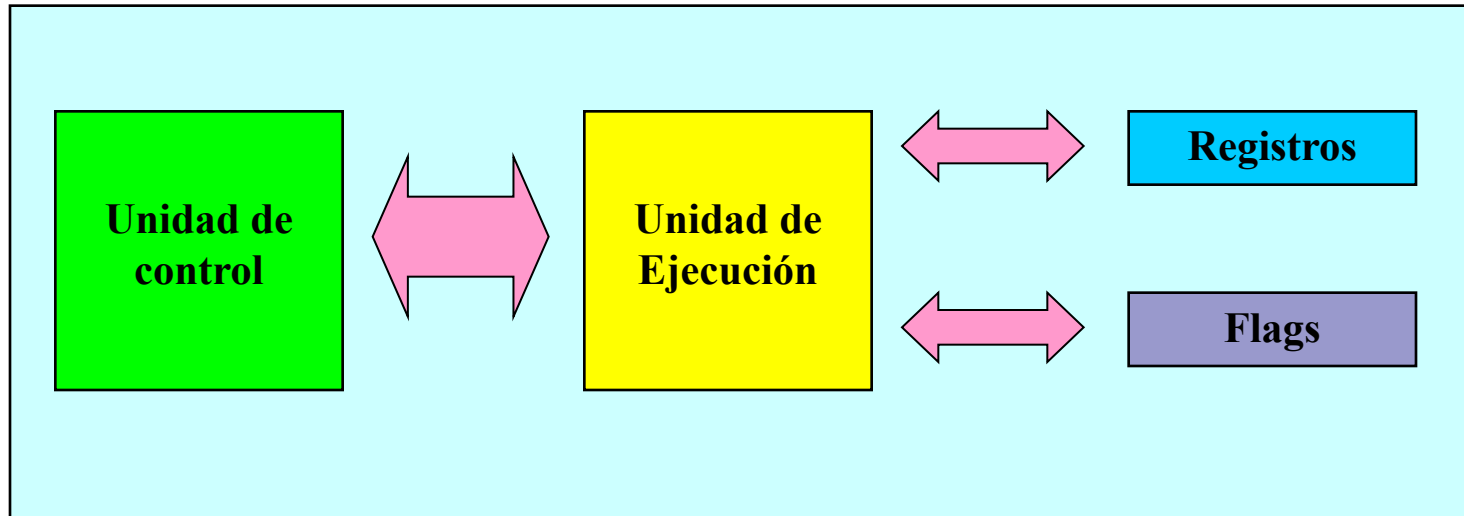
Sistema de Entrada y Salida



CPU



CPU (Resumen)



Unidad de control: Recupera Instrucciones de memoria, las decodifica, escribe en memoria

Unidad de Ejecución: Lleva a cabo la ejecución de la instrucción

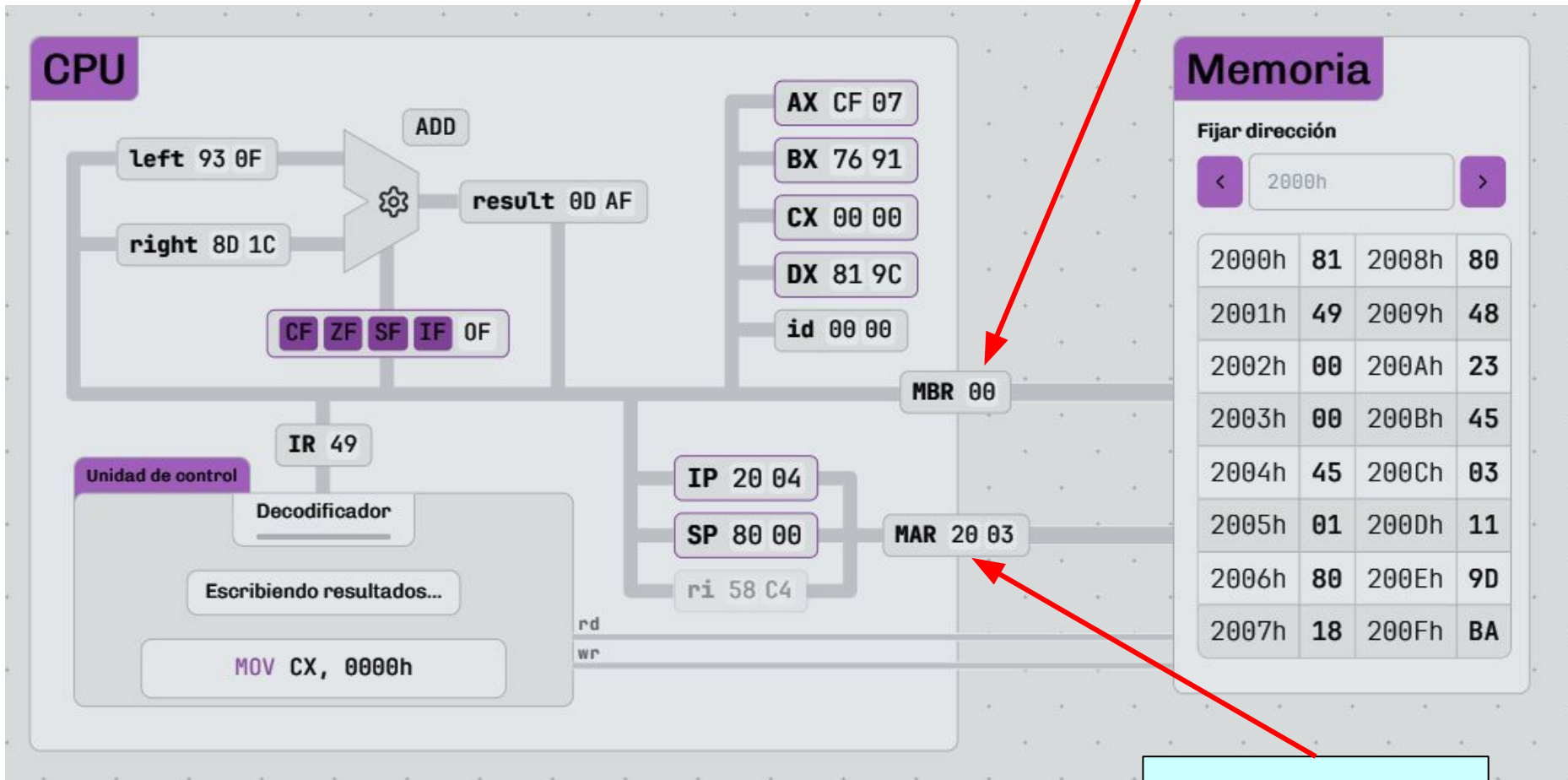
Registros: Memoria interna utilizada como variable.

Flags: Indican eventos luego de ejecutar las instrucciones

Veamos el simulador

<https://vonsim.github.io/>

Bus de Datos
(con buffer)



Registros internos de la CPU

IR: Registro de instrucción - Almacena la instrucción a analizar

ri: Almacena dirección temporal

id: Almacena dato temporal

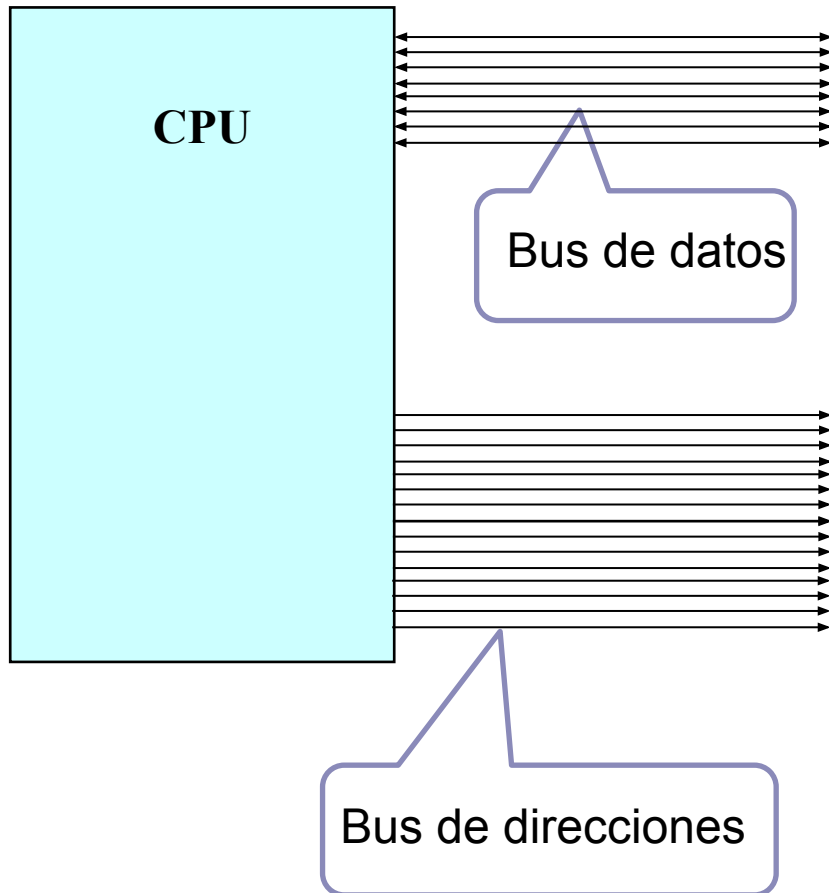
Bus de
Direcciones
(con buffer)



Mapa de memoria

Mapa de memoria

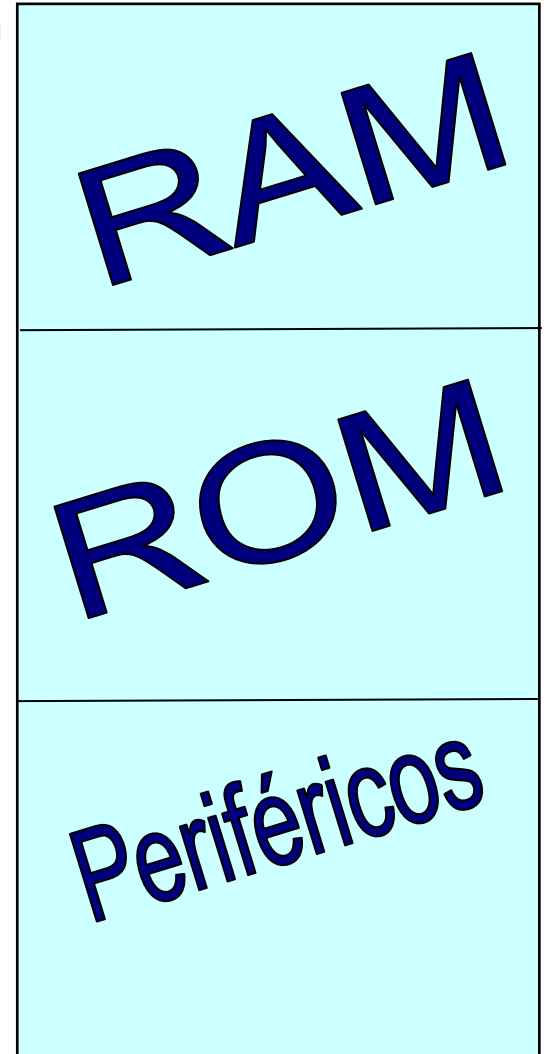
Todas las direcciones que
puede acceder una CPU



También se lo llama “Espacio de Direcciones”

Mapa de memoria

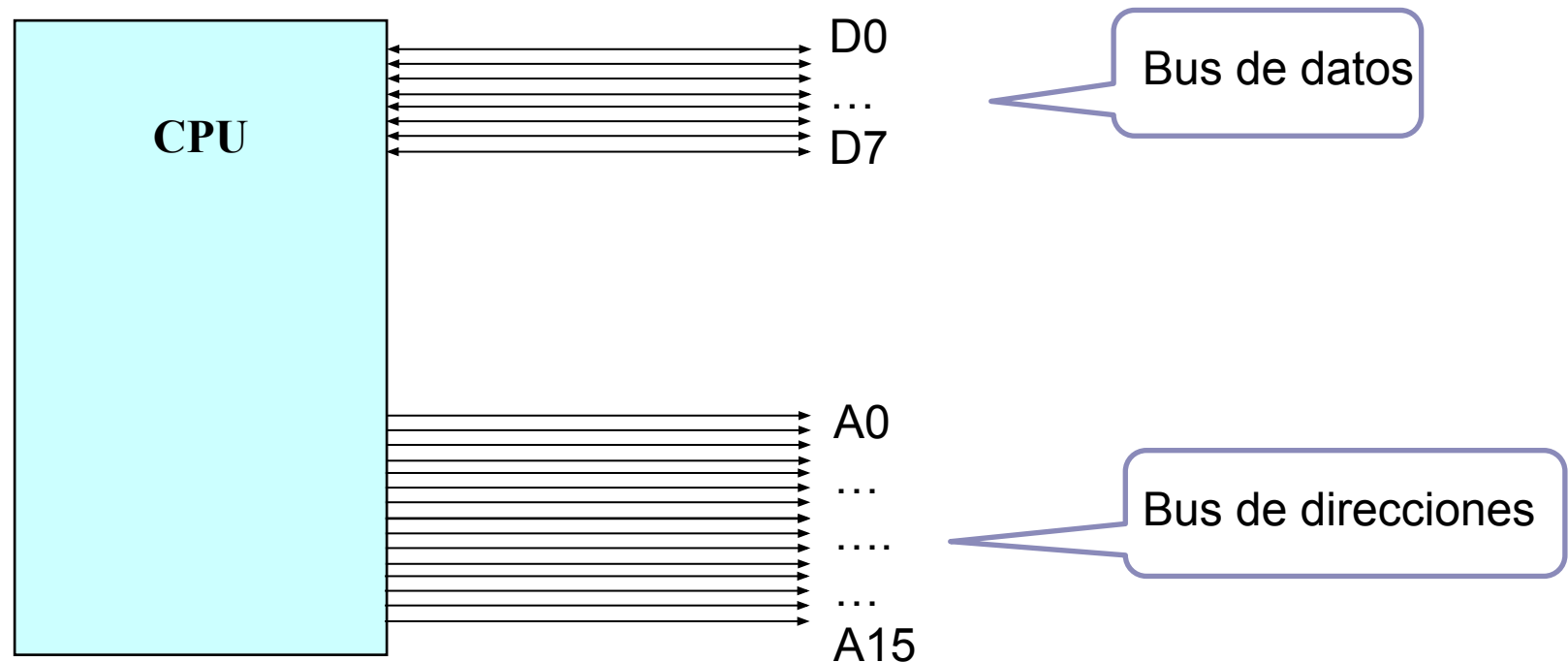
0000.....000h



FFF.....FFh

Mapa de memoria

- ❑ Supongamos un procesador que tiene 16 líneas (ó patitas) de bus de direcciones y 8 líneas de bus de datos.
- ❑ ¿ Que cantidad de información puede acceder ?



Mapa de memoria (2)

- ☐ ¿Y un procesador que tiene 16 líneas de bus de direcciones y 16 líneas de bus de datos ?
- ☐ ¿Y un procesador con 32 líneas de datos y 32 líneas de Direcciones ?

Para pensar

- **IP:** Puntero a instrucción.

¿ Cual es la primera instrucción que ejecuta el micro al encenderse ?

Pista: Pensar en la demo de VonSim

¿Qué valores tiene una memoria al energizarse ?



Memorias



Memorias - Clasificación

- ☐ Por el **modo** en que se accede a los datos
- ☐ Por las **operaciones** que aceptan
- ☐ Por la **duración** de los datos

Memoria – Tipo ROM

ROM (Read Only Memory)

PROM (Programmable ROM)

EPROM (Erasable PROM)

Flash, EEPROM (Electric Erasable Programmable ROM)

- Mantienen su información **sin energía** (no volátil)
- La escritura es **más lenta** que la RAM.



Memoria – Tipo RAM

RAM (Random Access Memory)

DRAM (Dinamic RAM)

SRAM (Static RAM)

- Pierde su información sin energía. (volátil)

- # DRAM

Memorias – Tiempo de Acceso

Es el tiempo que le toma a una memoria RAM para completar un acceso después de otro.

Se compone de:

- **Latencia** (tiempo que tarda en devolver el valor la memoria)
- **Transferencia**

Las DRAM suelen tener tiempos entre 50 y 150 ns.

Las SRAM menores a 10 ns.



Memorias - Operación

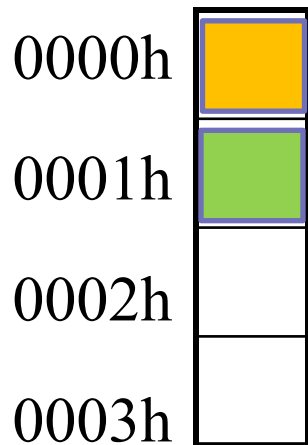
Las memorias para operar utilizan:

- ☐ Acción a realizar (lectura o escritura)
- ☐ Dirección de la palabra a acceder.
- ☐ Dato (entrante o saliente según acción)

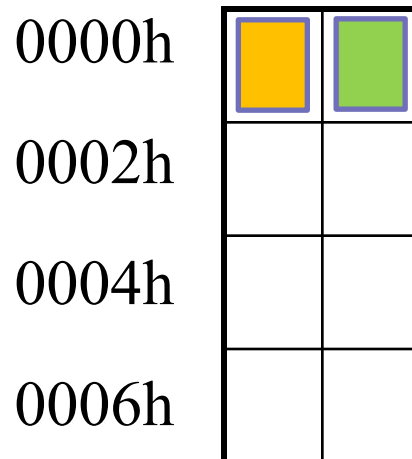
Memorias – Estructura

Si el procesador, como es el caso de **Intel**, quiere mantener compatibilidad hacia atrás, permite acceder a la memoria a **nivel byte**.

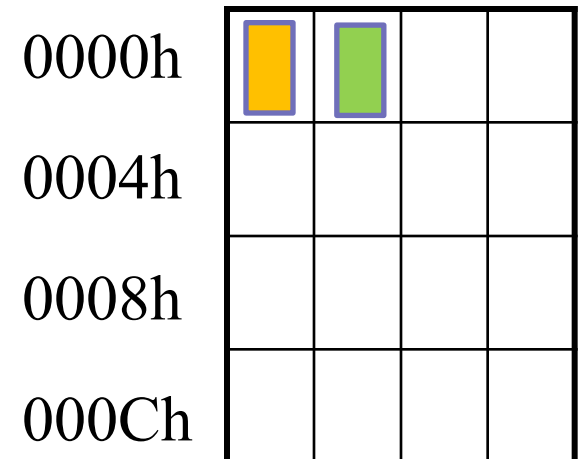
Por lo tanto la decodificación cambia según el tipo de memoria



Bus de 8 líneas



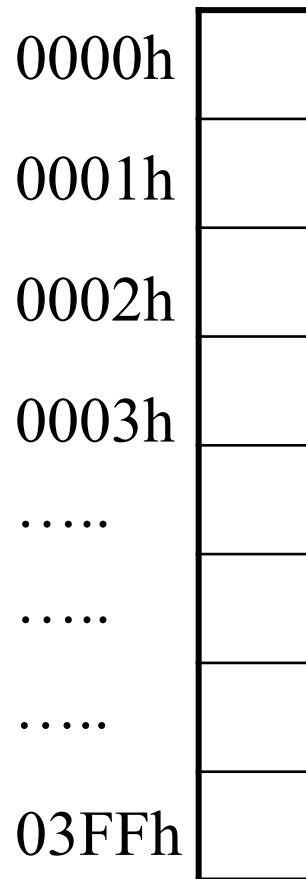
Bus de 16 líneas



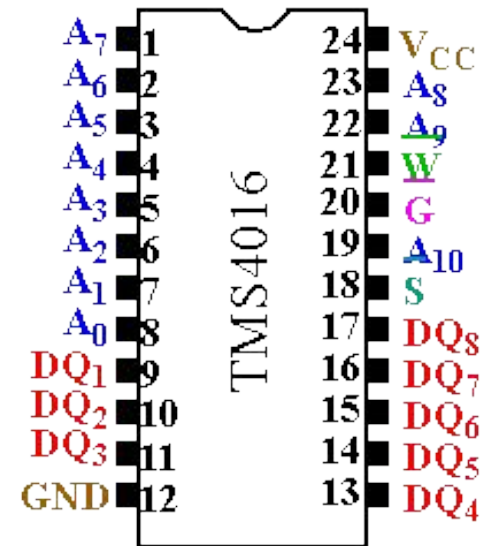
Bus de 32 líneas

Memorias – Estructura

Ejemplo de memoria de 1k x 8



Memoria Comercial (1)



2K x 8 SRAM

Pin(s)	Function
A_0-A_{10}	Address
DQ_0-DQ_7	Data In/Data Out
\overline{S} (CS)	Chip Select
\overline{G} (OE)	Read Enable
\overline{W} (WE)	Write Enable

Memoria Comercial (2)



- SDRAM 4GB
- DDR3
- 240 pines
 - 64 pines de bus de datos
 - Bus de Address dividido en buses de 16 lineas