

## Unidad 5.1 – MEMORIAS DE ESTADO SÓLIDO

# MEMORIAS DE ESTADO SÓLIDO

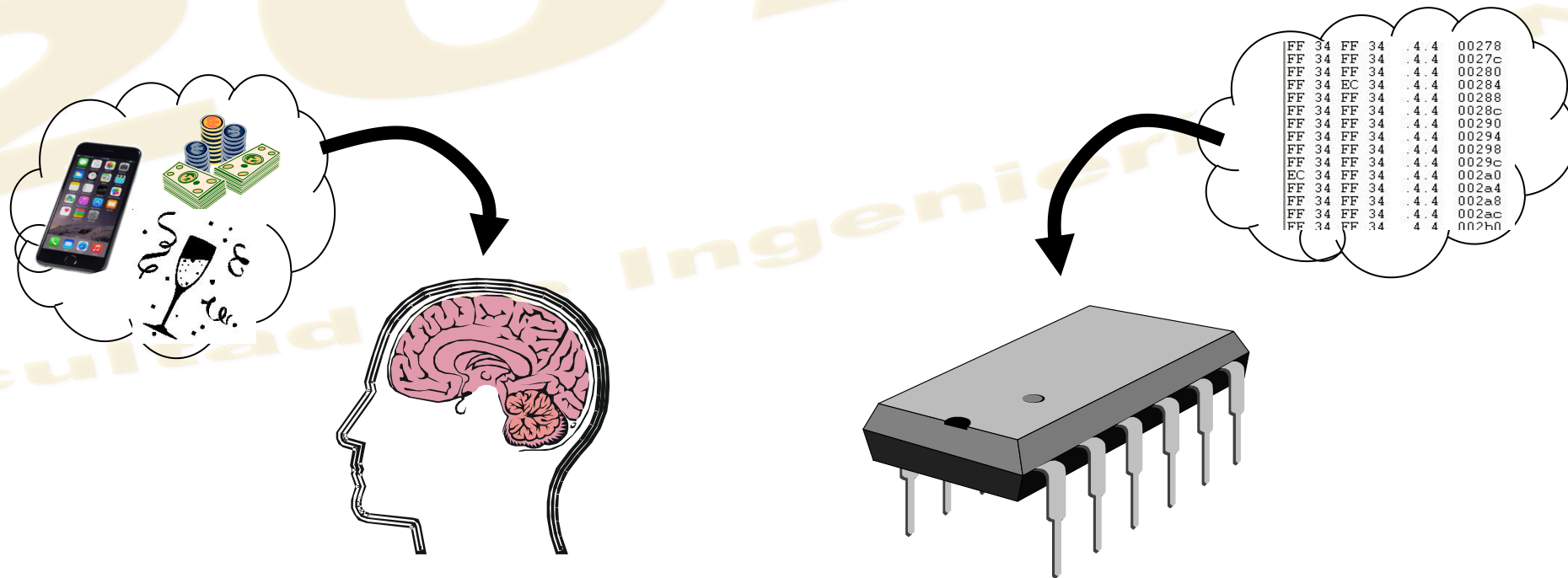
- **Memorias**
  - **Concepto**
  - **Parámetros Característicos**
- **Memorias de estado sólido**
  - **Memoria ROM**
  - **Memoria RAM**
- **Memoria en el sistema**
- **Posicionamiento de memorias.**

- > **Floyd T. (2006). FUNDAMENTOS DE SISTEMAS DIGITALES. Capítulo 10: Memorias y almacenamiento.**
- > **Tocci R. (2007). SISTEMAS DIGITALES PRINCIPIOS Y APLICACIONES. Capítulo 12: Dispositivos de memoria.**
- > **Mano M. (2003). FUNDAMENTOS DE DISEÑO LÓGICO Y DE COMPUTADORAS. Capítulo 9: Memorias.**
- > **Florez H. (2010). DISEÑO LÓGICO - FUNDAMENTOS EN ELECTRÓNICA DIGITAL. Capítulo 8: Memorias.**

## Concepto

**General:** es un sistema que permite almacenar datos e información para ser utilizados luego a requerimiento del propietario.

**Informático:** es un dispositivo de hardware que puede almacenar una determinada cantidad de datos en un formato predefinido, para ser accedidos con posterioridad bajo esquemas de lectura, modificación o eliminación.



# MEMORIAS: PARÁMETROS CARACTERÍSTICOS

**Capacidad:** cantidad de datos que puede almacenar.

**1 byte = 8 bits**

**1 Kbyte =  $2^{10}$  bytes = 1.024 bytes** → (1 mil ...)

(KILO Byte)

**1 Mbyte = 1 Kbyte x 1 Kbyte =  $2^{20}$  bytes = 1.048.576 bytes** → (1 millón ...)

(MEGA Byte)

**1 Gbyte = 1 Kbyte x 1 Mbyte =  $2^{30}$  bytes = 1.073.741.824 bytes** → (1 millardo ...)

(GIGA Byte)

**1 Tbyte = 1 Kbyte x 1 Gbyte =  $2^{40}$  bytes = 1.099.511.627.776 bytes** → (1 billón ...)

(TERA Byte)

**1 Pbyte = 1 Kbyte x 1 Tbyte =  $2^{50}$  bytes = 1.125.899.906.842.624 bytes** → (1 billardo ...)

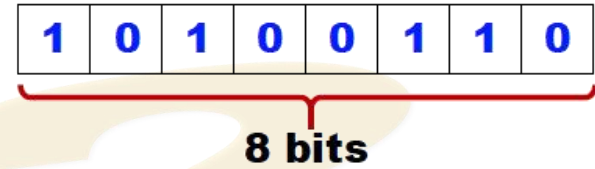
(PETA Byte)

**1 Ebyte = 1 Kbyte x 1 Pbyte =  $2^{60}$  bytes = 1.152.921.504.606.846.976 bytes** → (1 trillón ...)

(EXA Byte)

**1 Zbyte = 1 Kbyte x 1 Ebyte =  $2^{70}$  bytes = 1.180.591.620.717.411.303.424 bytes** → (1 trillardo ...)

(ZETTA Byte)



**Densidad de almacenamiento:** [bits/mm<sup>2</sup>] cantidad de bits almacenados por unidad de superficie.

**Tiempo de acceso:** [ns] tiempo requerido para almacenar o extraer un dato.

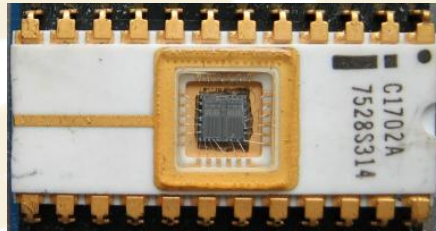
**Costo de almacenamiento:** [\$/bit] relación de costo por capacidad del dispositivo.

**Confiabilidad:** [%] [N° errores/N° accesos] errores que pueden ocurrir durante la operación del dispositivo. Un error cada  $10^{10}$  accesos genera una confiabilidad del 99,99999999%.

## Memorias de estado sólido



**ROM**



**EPROM**



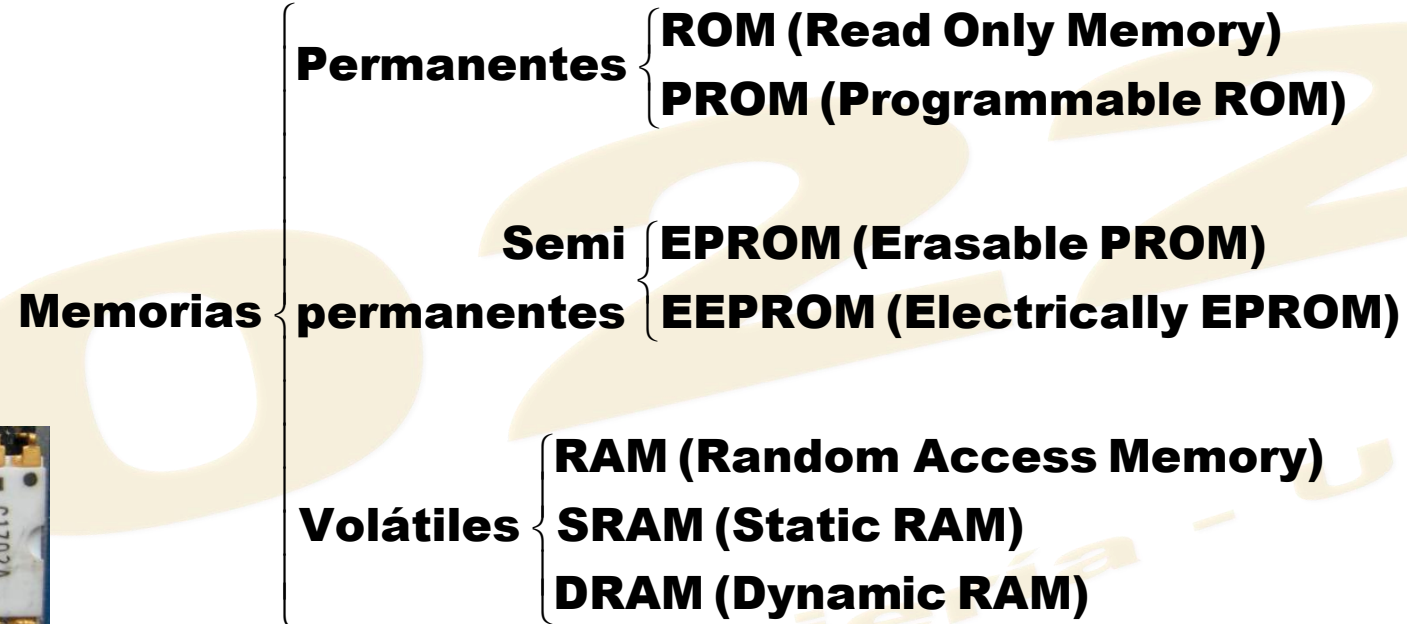
**EEPROM**



**USB  
Flash Drive  
(EEPROM)**

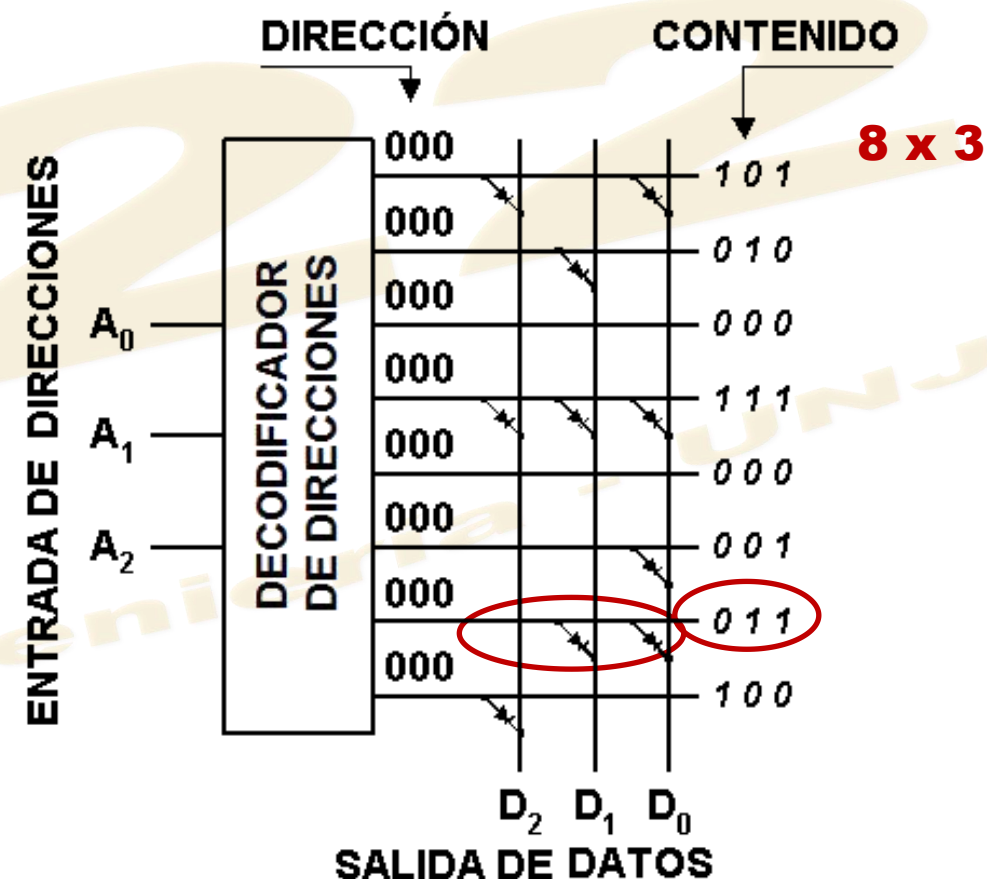
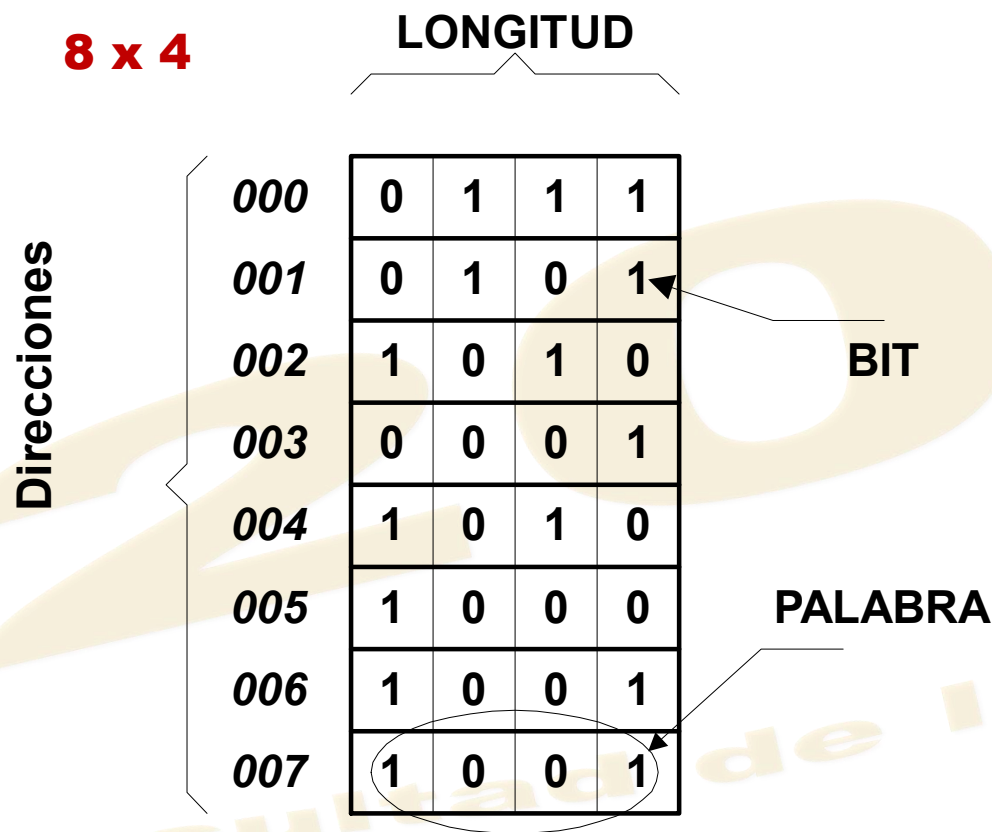


**DRAM SIMM**



## Memoria ROM

Memoria permanente grabada en fábrica o por el usuario, por única vez.



Se interpreta como pequeños diodos en las intersecciones de la matriz de conexiones, que aportan un 1 lógico. Al grabarse, si se “funde” representa un 0 lógico.

(Floyd, pg. 623; Tocci, pg. 795)

## Memoria ROM - Pin-out característico

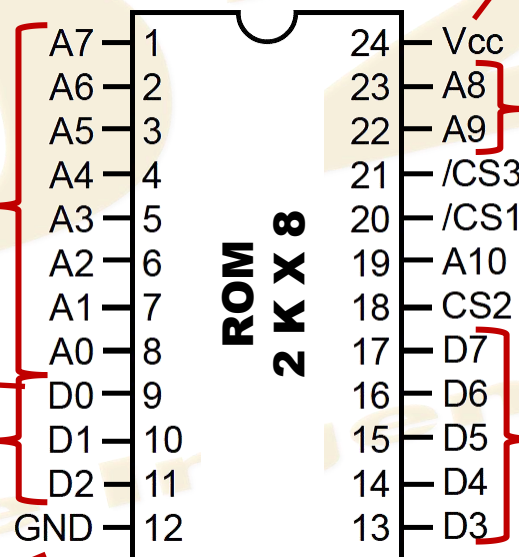
**2 KByte  
(2048 Byte)  
x 8 bits**

**Volt corriente  
continua: Conexión  
eléctrica positiva.**

**Líneas de direcciones:**  
La cantidad depende de  
la capacidad de la  
memoria. Se conectan al  
bus de direcciones y sólo  
reciben datos del  $\mu P$ .

**Líneas de datos  
(unidireccionales):** La cantidad  
depende de la longitud de  
palabra de la memoria. Se  
conectan al bus de datos y  
entregan datos al  $\mu P$ .

**Selección de chip (CS1!-CS2-  
CS3!):** Cuando estas líneas son  
activadas simultáneamente  
por el  $\mu P$ , se permite lectura o  
escritura en el chip.



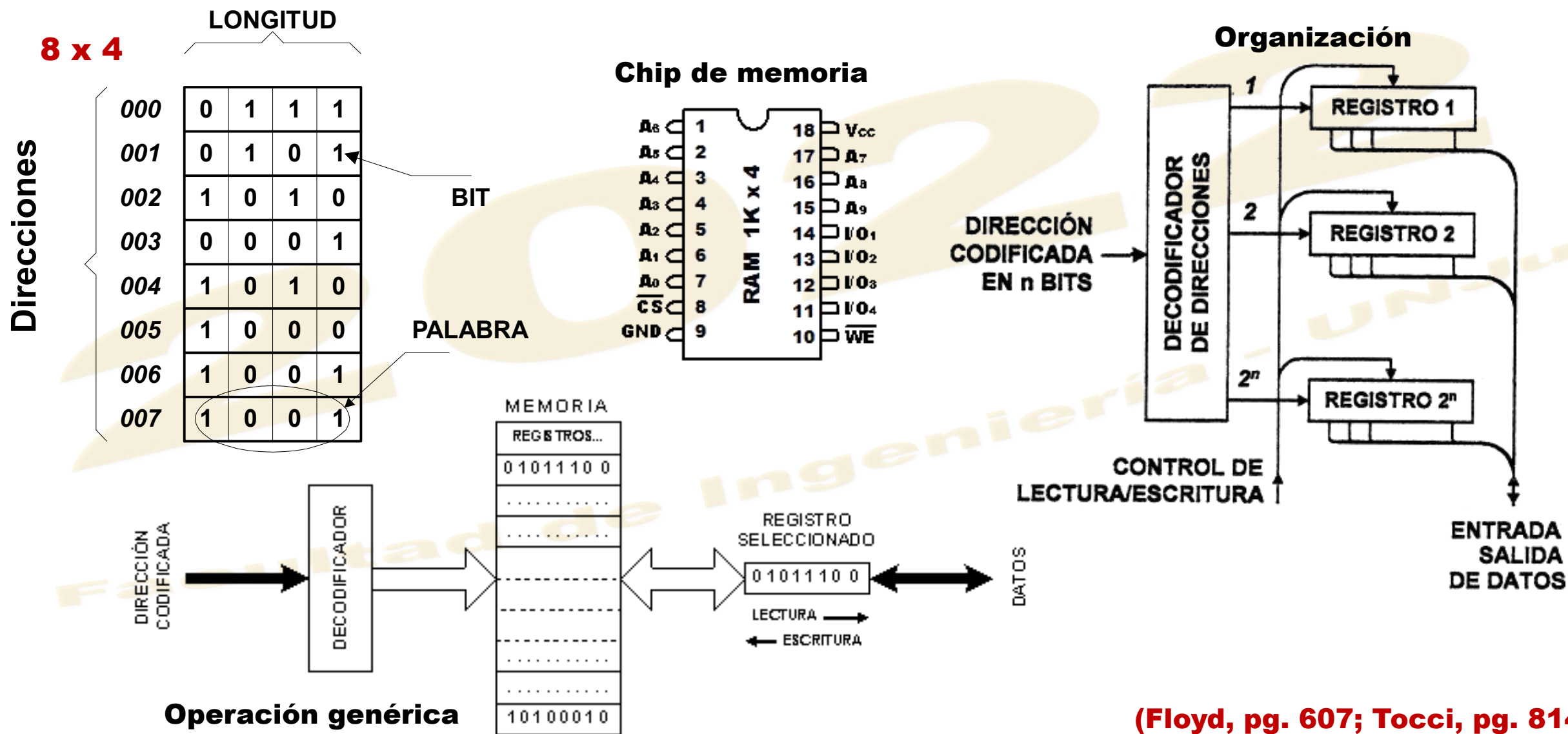
**Ground (tierra):** Conexión  
eléctrica negativa.

**Líneas de datos  
(unidireccionales):** La cantidad  
depende de la longitud de  
palabra de la memoria. Se  
conectan al bus de datos y  
entregan datos al  $\mu P$ .

**Vista esquemática  
superior de un chip  
de memoria**

# MEMORIAS DE ESTADO SÓLIDO

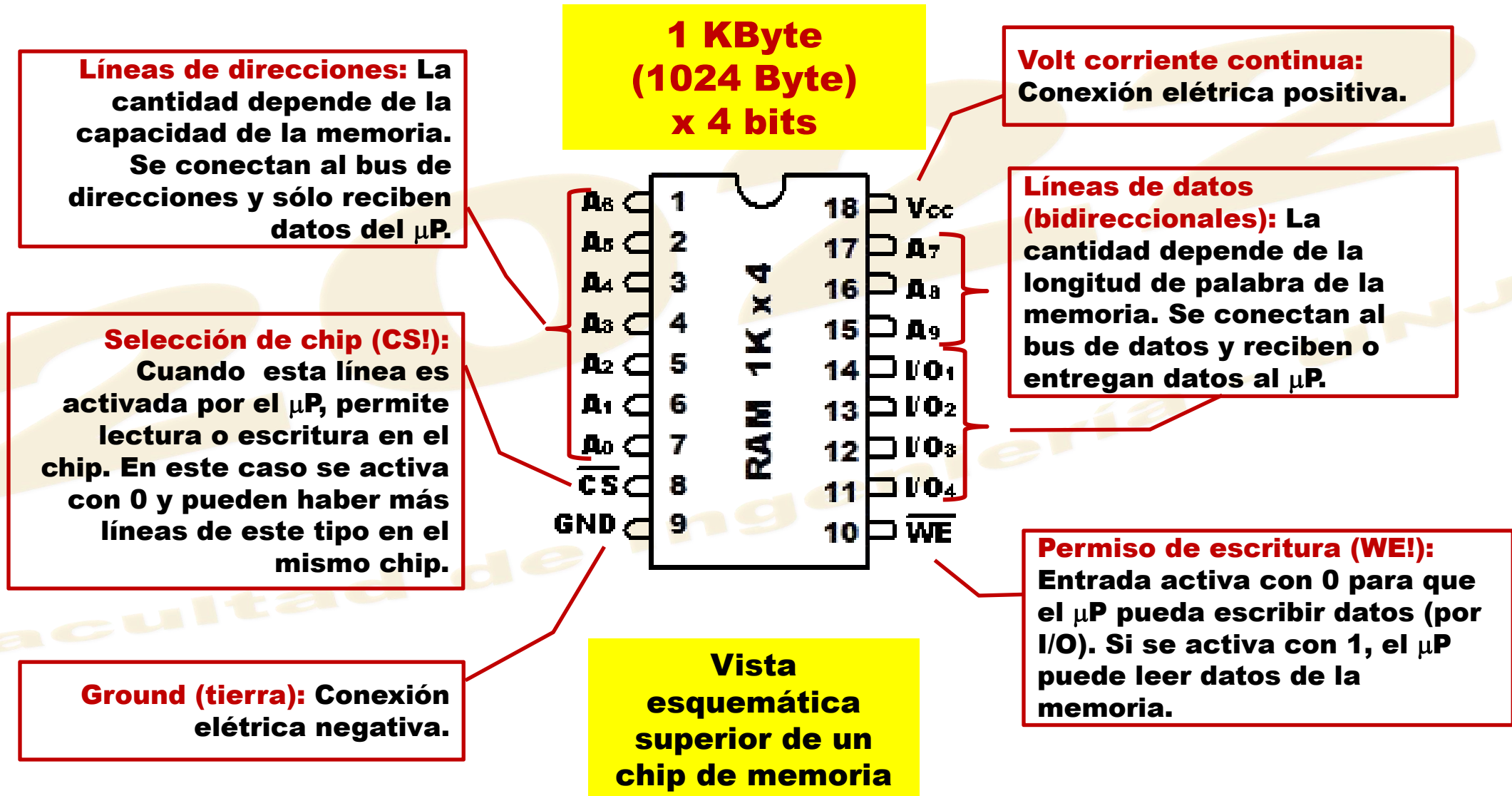
## Memoria RAM Memoria volátil de lectura/escritura con acceso aleatorio.



(Floyd, pg. 607; Tocci, pg. 814)

## Memoria RAM - Pin-out característico

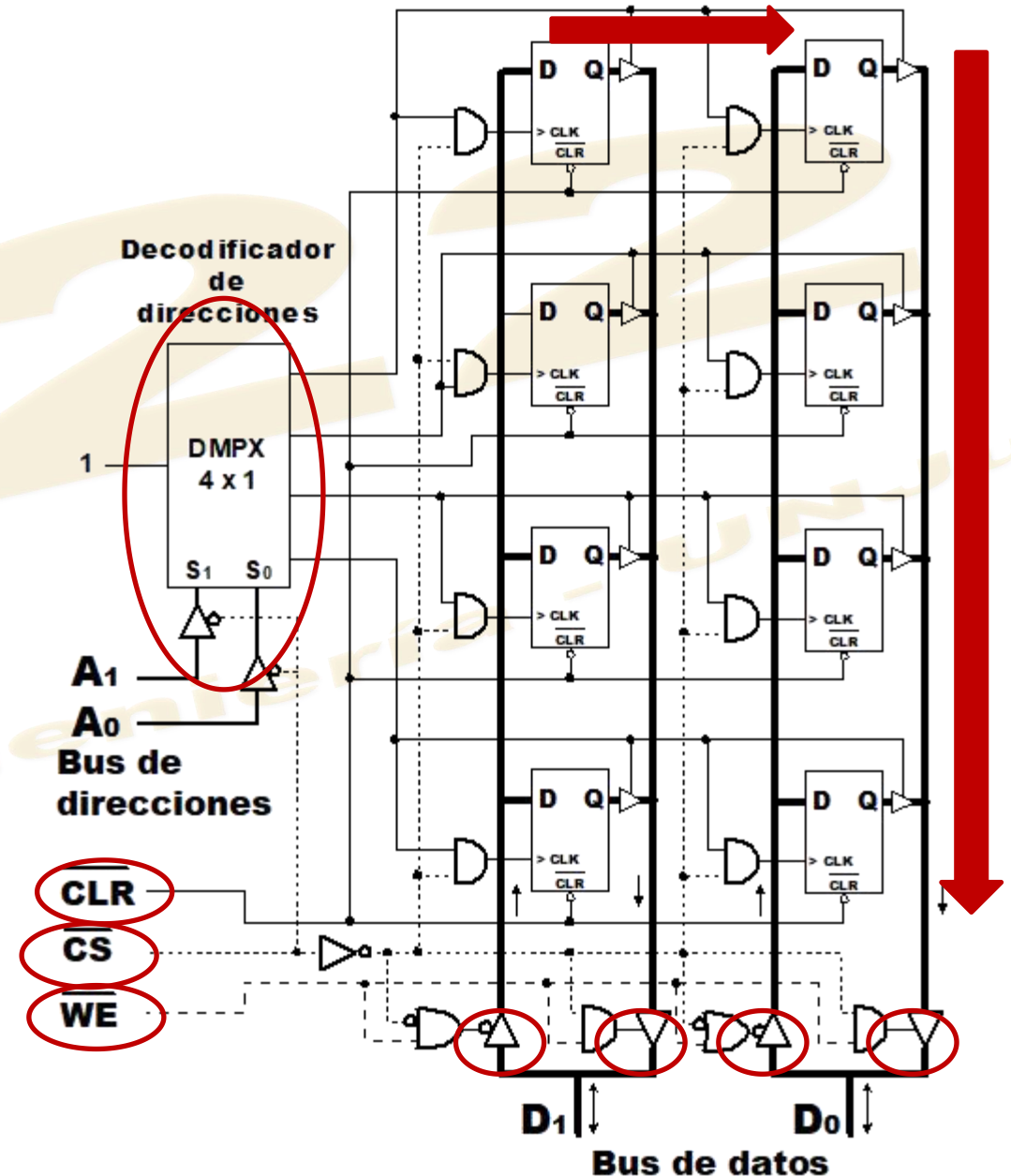
(Tocci, pg. 790)





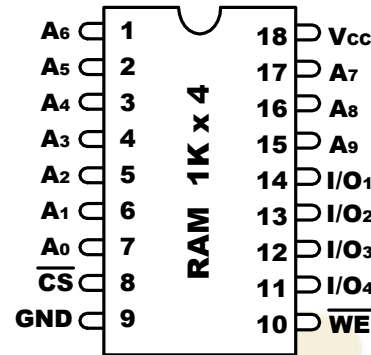
## Ejemplo: Memoria RAM 4 x 2

- 4 registros con 2 bits por registro.
- La unidad se selecciona con 0 (CS). El chip deseleccionado desconecta todos los buses y bloquea el CLK.
- La habilitación de escritura se hace con 0 (WE!). También necesita de CS!.
- El bus de datos (interno) se unifica (hacia afuera), por cada bit, con compuertas tri-state.
- El sistema dispone de una línea de borrado general (no habitual).
- La decodificación de direcciones se hace con un demultiplexor 4 x 1.

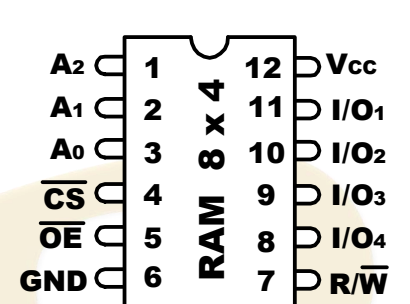


## Ejemplo: Memoria RAM de 8x4.

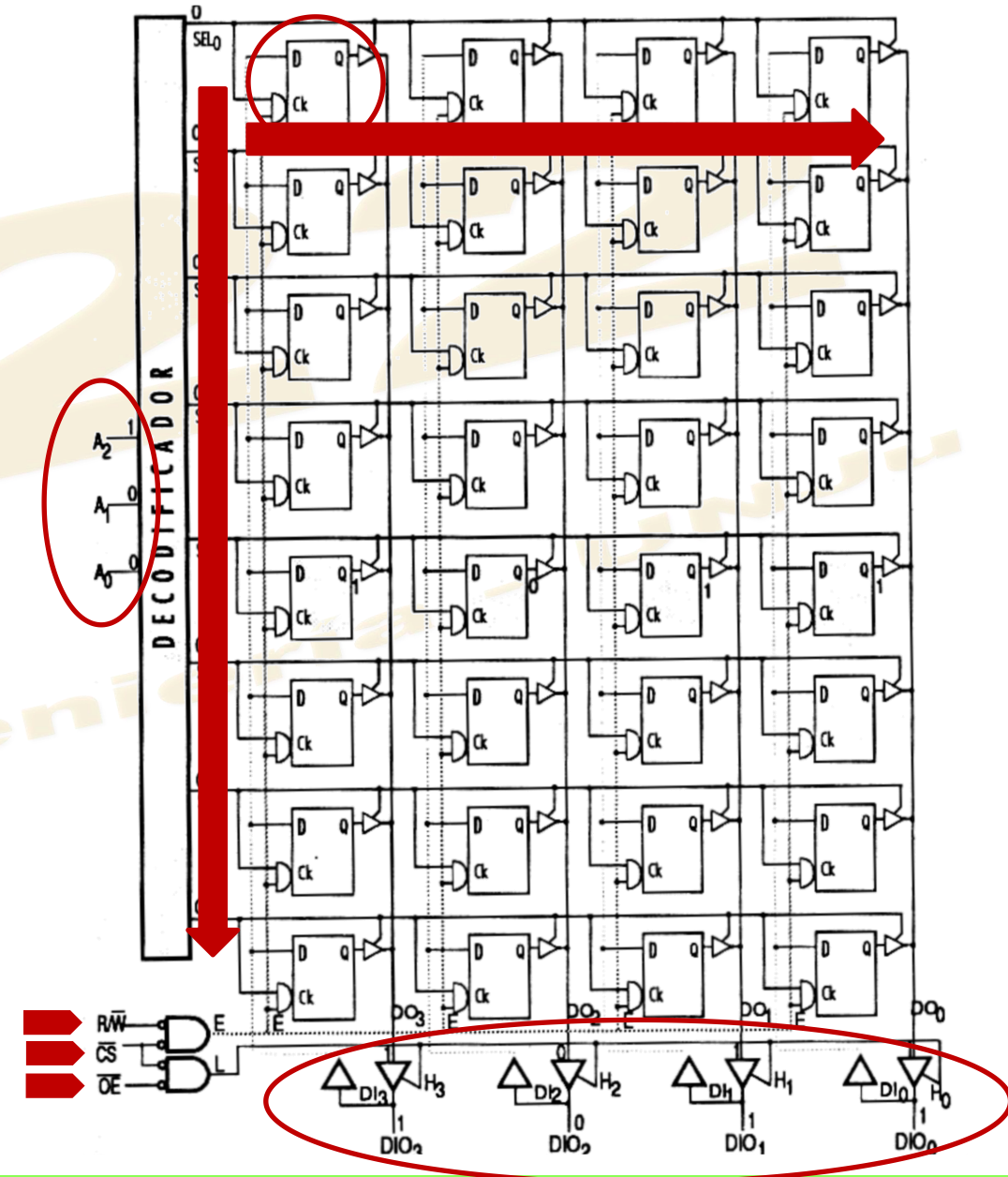
Esquema real



Esquema ficticio



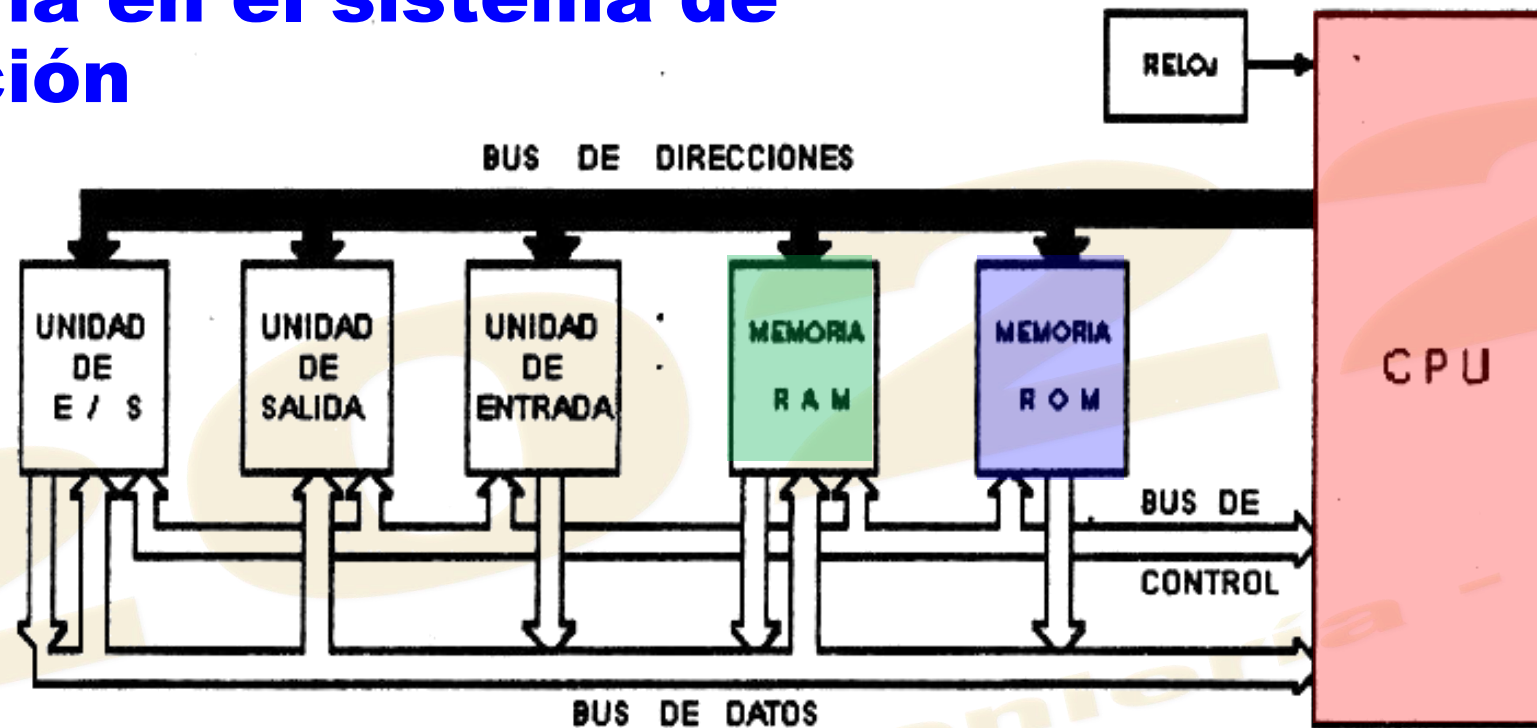
- Memoria RAM con flip-flop tipo D.
- 8 registros de 4 bits.
- Bus de direcciones de 3 bits  $A_0 A_1 A_2 \rightarrow 2^3 = 8$  direcciones
- Bus de datos bidireccional  $DIO_0 DIO_1 DIO_2 DIO_3$
- Lectura/escritura R/W!
- Selección de chip CS!
- Habilitación de salida OE!



## La memoria en el sistema de computación

(Tocci, pg. 793)

**arquitectura Von Neumann:**  
datos e instrucciones por el mismo bus



**CPU:** Unidad Central de Proceso (microprocesador).

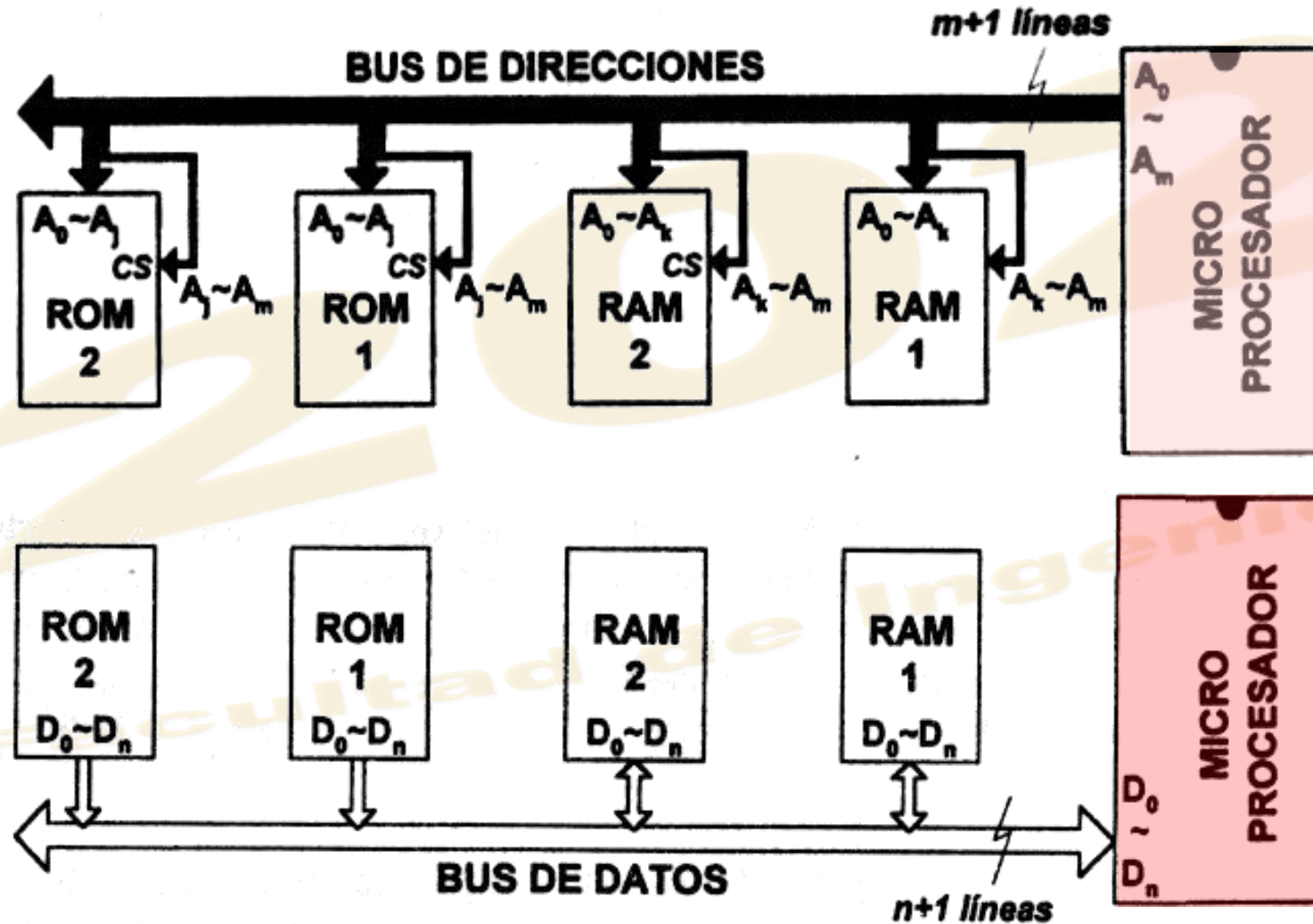
**Bus de direcciones:** conjunto de líneas que permiten individualizar un registro particular (unidireccional).

**Bus de datos:** conducen los datos desde y hacia la CPU (bidireccional).

**Bus de control:** conjunto de líneas que transportan señales de control. Activan las unidades y gestionan las interrupciones (bidireccional).

## La memoria en el sistema de computación

(Floyd, pg. 605)



El bus de direcciones conecta a todas las unidades. Selecciona chips y registros. Es unidireccional.

El bus de datos llega a todas las unidades. Es unidireccional para las ROM y bidireccional para las RAM.

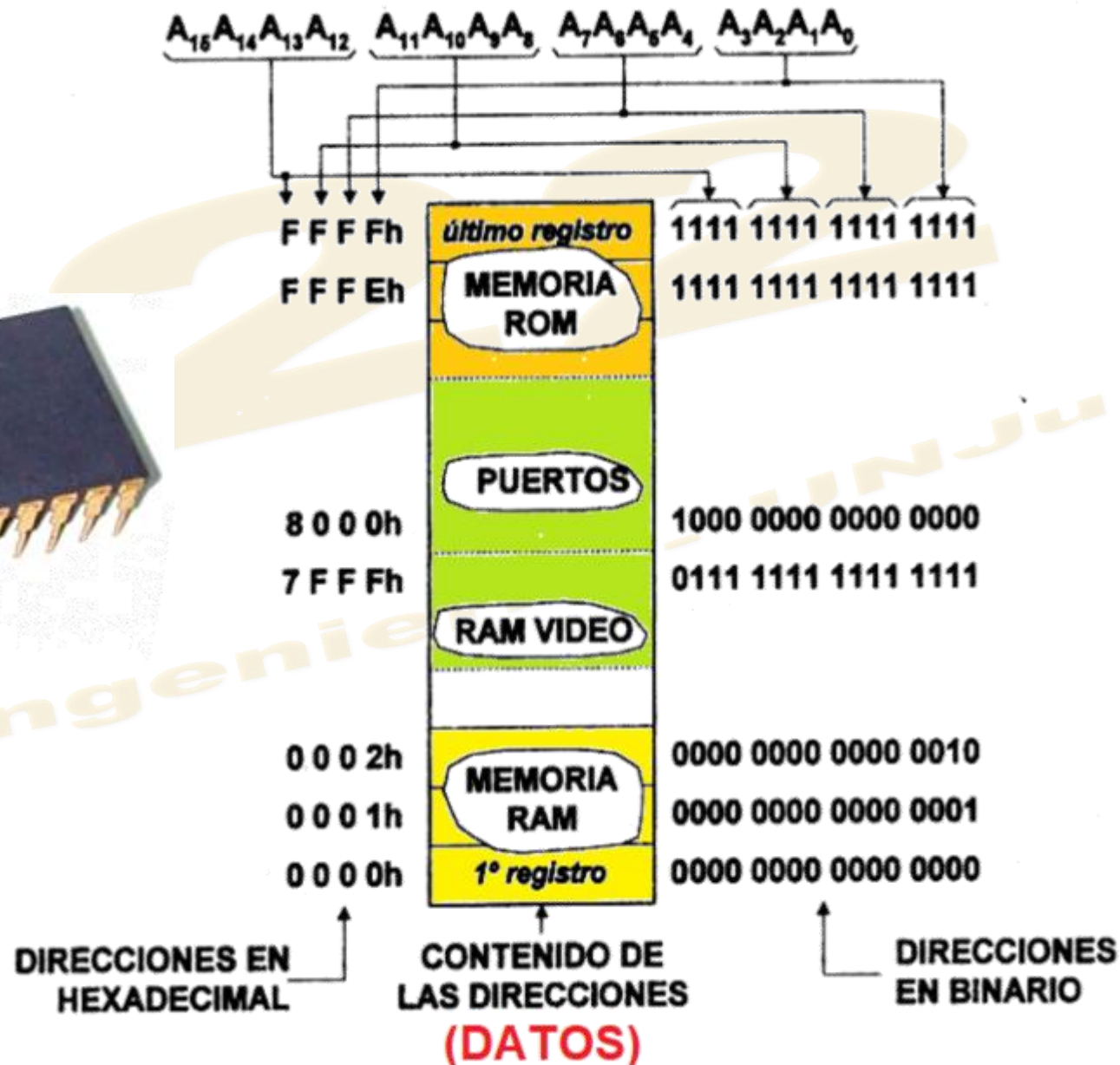
## Mapa de memoria (o mapa de direcciones)

**MICROPROCESADOR**  
**de 64 KB x 8 bits**  
**65.536 direcciones**

**Motorola**  
**6800**



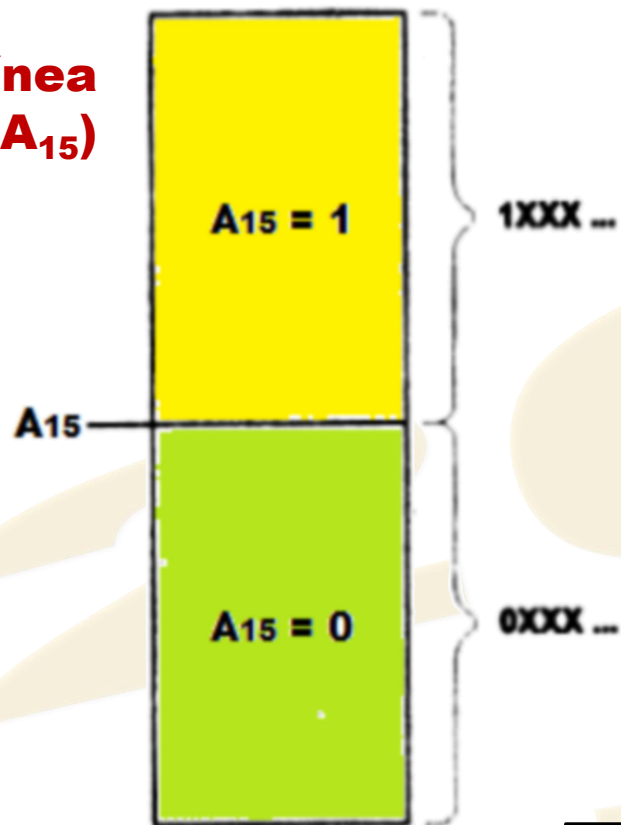
Las líneas del bus de direcciones se utilizan para ubicar los chips de memoria (y otros dispositivos) en el mapa, seleccionar el chip e identificar cada registro individualmente.



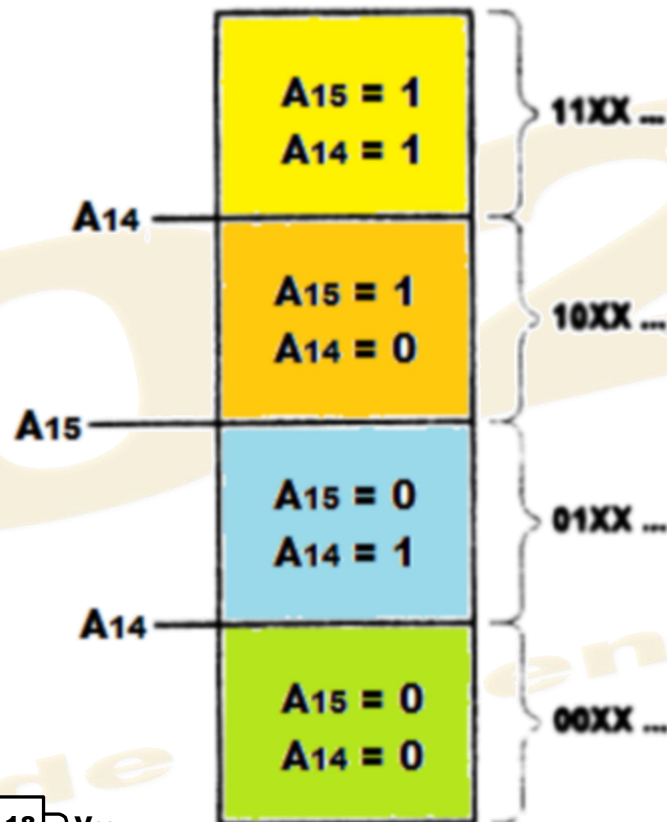
# POSICIONAMIENTO DE MEMORIAS

## Partición del mapa

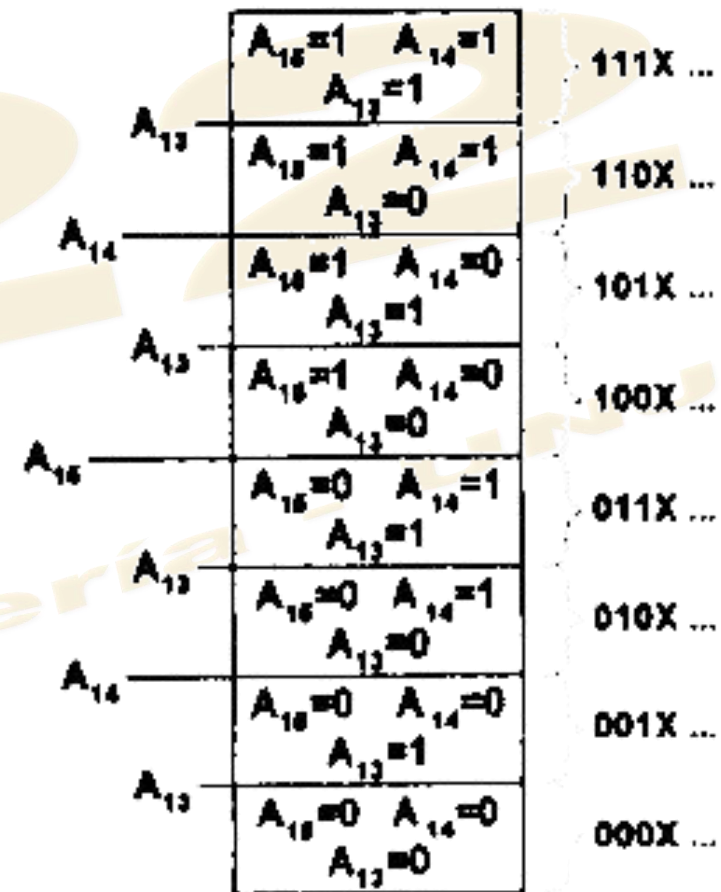
Una línea  
( $A_{15}$ )



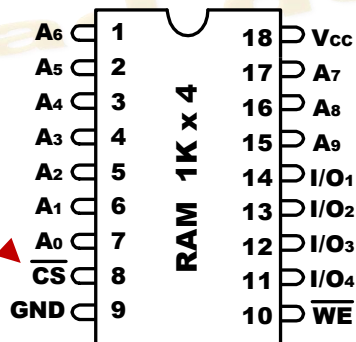
Dos líneas  
( $A_{15} | A_{14}$ )



Tres líneas  
( $A_{15} | A_{14} | A_{13}$ )

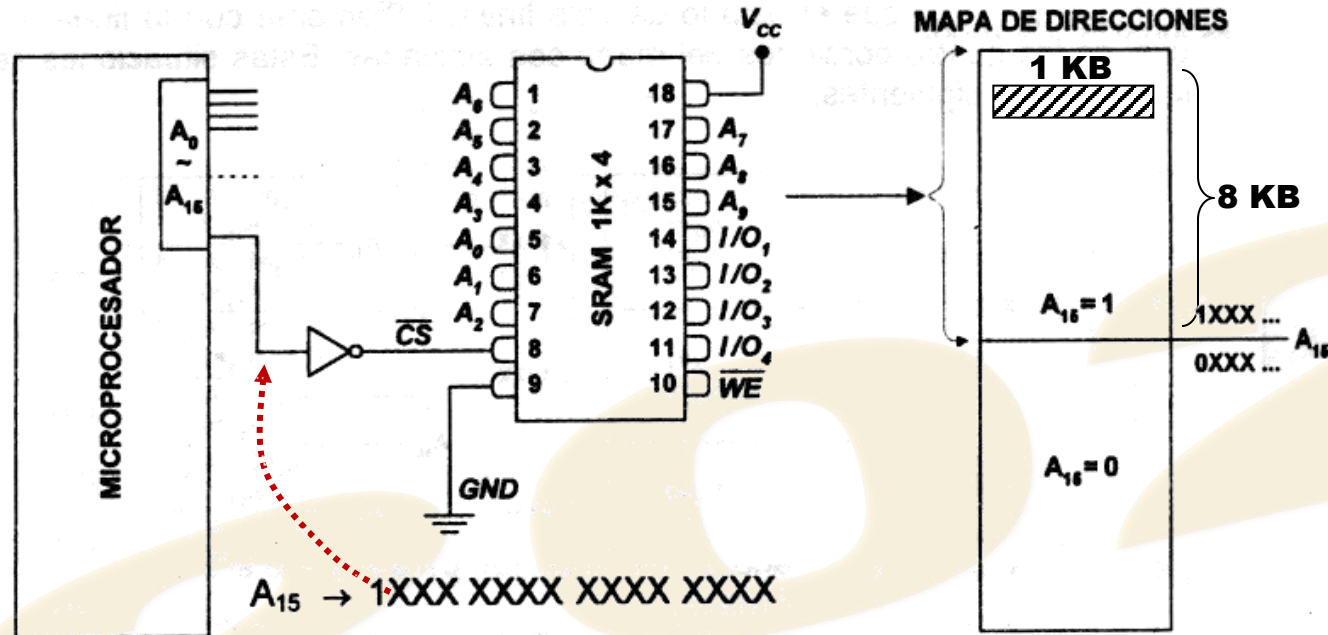


Selección  
de chip CS!  
con 0



La selección y conexión de las líneas más altas al CS! permite ubicar cada chip de memoria en una zona determinada.

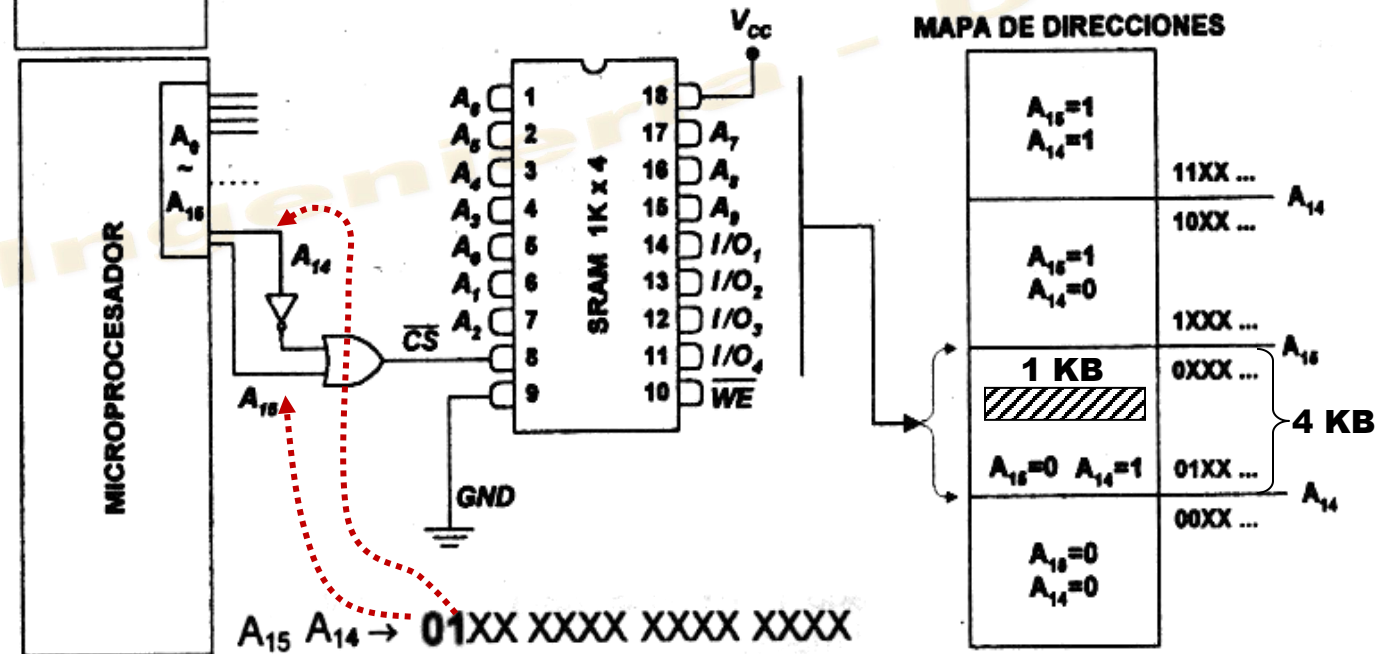
# POSICIONAMIENTO DE MEMORIAS



La línea  $A_{15} = 1$  (sola), ubica al chip en algún lugar de la mitad superior del mapa de direcciones. Con el resto de las líneas se precisa el lugar exacto.

Las líneas  $A_{15} = 0$  y  $A_{14} = 1$ , ubica al chip en algún lugar del cuarto central inferior del mapa de direcciones.

Con el resto de las líneas se precisa el lugar exacto.



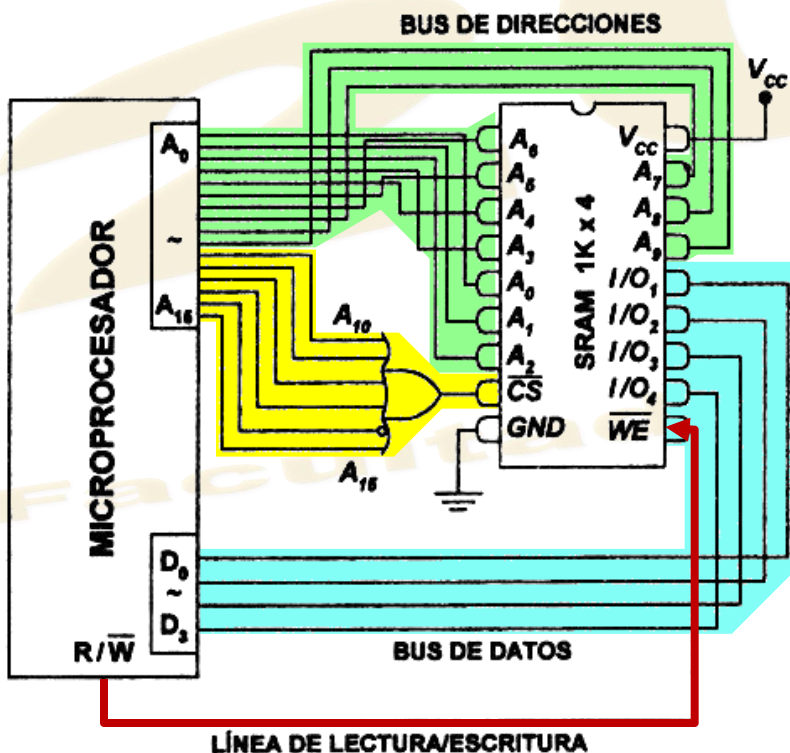
# POSICIONAMIENTO DE MEMORIAS

**EJEMPLO:** Ubicación de chip RAM de 1 Kbyte x 4 bits, a partir de dirección  $4\ 000_{16}$  ( $16\ 384_{10}$ ) en un sistema de 64 KB (16 líneas de address).

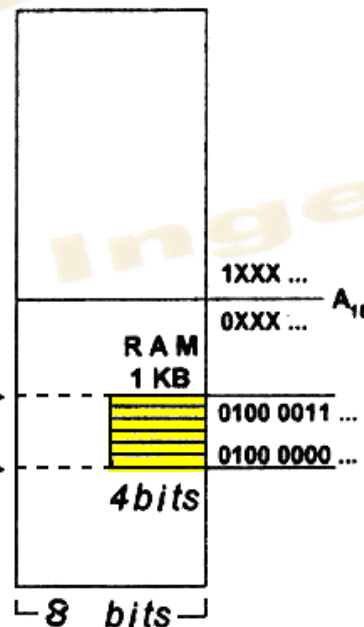
Identificación →		A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	
Dirección inicial →	Hexa	4				0				0				0h				
	Bin	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		selección de chip ←							→ direccionamiento interno									
Dirección final →	Bin	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	
	Hexa	4				3				F				Fh				

líneas para posicionamiento externo

registros internos de la memoria (= 1 KB)



MAPA DE DIRECCIONES



Las líneas A<sub>15</sub> a A<sub>10</sub> seleccionan el chip (CS!).

Las líneas A<sub>9</sub> a A<sub>0</sub> seleccionan los 1024 registros internos.

Las líneas D<sub>3</sub> a D<sub>0</sub> conectan la mitad de los datos del bus de datos.

R/W! del  $\mu$ P habilita lectura/escritura sobre WE! (write enable).



# POSICIONAMIENTO DE MEMORIAS

**EJEMPLO:** Ubicación de chip RAM de 1 Kbyte x 4 bits, a partir de dirección  $4\ 000_{16}$  ( $16\ 384_{10}$ ) en un sistema de 64 KB (16 líneas de address).

