

# Tema 5

## Tecnología de las Memorias semiconductoras

# Contenidos (I)

## 1. Organización básica de las memorias semiconductoras.

- 1.1. Capacidad de almacenamiento. Tamaños de la información.
- 1.2. Conexión UCP-Memoria.
- 1.3. Estructura básica de las memorias semiconductoras

## 2. Tipos de memorias semiconductoras.

### 2.1. No volátiles:

- 2.1.1. ROM, PROM, EEPROM, FLASH . Celdillas elementales. Ejemplos

### 2.2. Volátiles de acceso aleatorio (RAM):

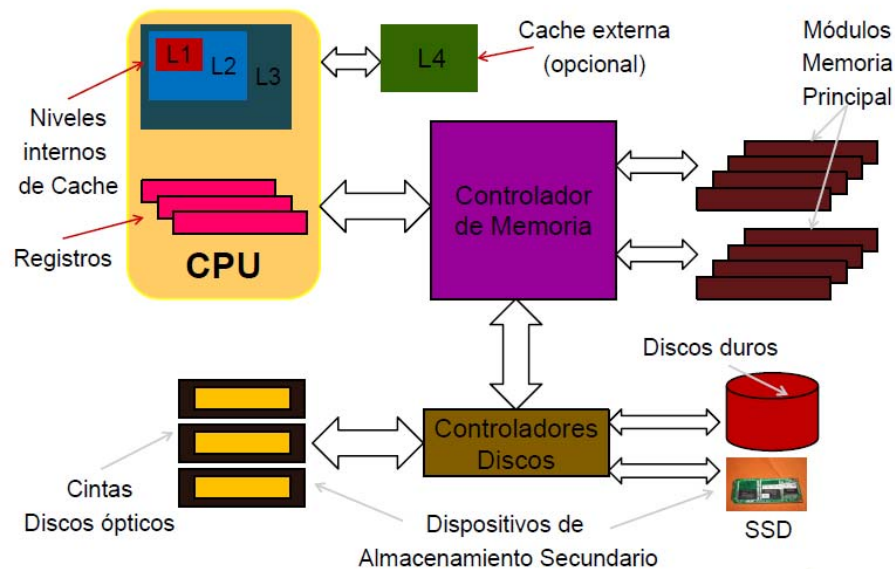
- 2.2.1. RAM estática (SRAM). Celdillas elementales. Estructura. Ejemplos.  
Cronogramas básicos de lectura y escritura.
- 2.2.2. RAM dinámica (DRAM). Celdilla elemental.

## Bibliografía del tema

- **Bibliografía**

- \* Floyd, T.L. Fundamentos de sistemas digitales. Ed. Prentice Hall. 1997. Caps. 12 y 14.
- \* “Diseño Digital”. Wakerly. Ed. Prentice-Hall. 2006.
- \* Prince, B. Semiconductor memories. A handbook of design manufacture and application. Ed. Wiley. 1991
- \* Electrónica digital, principios y aplicaciones (Roger Tokheim). 2010. Cap. 11: Memorias.
- \* Jan M. Rabaey. Circuitos Integrados Digitales. Pearson Prentice Hall. 2004.

## Sistema de memoria de un computador



Fuente: Estructura de Computadores, T5: La memoria principal

Nos centraremos en las memorias semiconductoras, y más concretamente en:

- SRAM, usadas en la Cache, con una estructura de celda basada en la de los latches

- DRAM, usadas en los módulos de memoria principal

- No volátiles, sobre todo FLASH, usada en discos SSD, ROM BIOS, pendrives, etc.

Aparte de su uso en dispositivos con baterías, como teléfonos móviles, cámaras digitales, reproductores de audio, etc.

En segundo curso (asignatura ETC), se profundizará en la memoria DRAM y en los discos magnéticos.

# 1. Organización Básica.

## 1.1. Capacidad de almacenamiento

- Cantidad de información almacenada: bits o bytes
  - \* Nomenclatura: B = 1 byte, b = 1 bit
- ¿Cómo se expresa la capacidad?
  - \* Capacidad total: en bytes o múltiplos de byte
- Prefijos
  - \* Según el contexto, son de tipo  $2^n$  ó del tipo  $10^n$ 
    - Ejemplo: la capacidad de la memoria principal siempre se expresa en unidades del tipo  $2^n$

Prefijo	Valor ( $2^n$ )	Valor ( $10^n$ )
Kilo (K)	$2^{10}$	$10^3$
Mega (M)	$2^{20}$	$10^6$
Giga (G)	$2^{30}$	$10^9$
Tera (T)	$2^{40}$	$10^{12}$
Peta (P)	$2^{50}$	$10^{15}$

5

1Byte (B) = 8 bits (b)

La CAPACIDAD de la memoria se expresa en potencias de 2, al contrario de otras magnitudes de la memoria, que se expresan en potencias de 10 (frecuencia, ancho de banda, ...)

## 1.1. Ejemplos de capacidad de almacenamiento

- Capacidad total expresada en bytes
  - \*  $1024 \text{ bytes} = 2^{10} \text{ bytes} = 1 \text{ KB}$
- La memoria tiene 128K posiciones de 16 bits cada una
  - \*  $128\text{K} \times 16 \text{ bits} = 128\text{K} \times 2^4 \text{ bits} = 128\text{K} \times 2^1 \text{ bytes} = 256 \text{ KB}$
- La memoria tiene 8 MB en palabras de 32 bits
  - \*  $8 \text{ MB} = 2^3 \times 2^{20} \times 2^3 \text{ bits} = 2^1 \times 2^{20} \times 2^5 \text{ bits} = 2\text{M} \times 32 \text{ bits}$
- Otros ejemplos
  - \*  $64 \text{ Kbits} = 64\text{K} \times 1 \text{ bits} = 2^{16} \text{ bits} = 2^{13} \times 2^3 \text{ bits} = 2^{13} \text{ bytes}$
  - \*  $256 \text{ Mbits} = 2^8 \times 2^{20} \text{ bits} = 2^5 \times 2^{20} \times 2^3 \text{ bits} = 32 \text{ MB}$

6

Capacidad = Número de palabras x Tamaño de la palabra

Se puede dar en múltiplos de bit (b) o de Byte (B)

## 1.1. Tamaños de la información

- Palabra

- \* Unidad máxima de transferencia en un acceso
- \* La longitud en bytes suele ser una potencia entera de 2 (1, 2, 4, 8, 16, ...)
- \* Ejemplo
  - Transferencias entre memoria principal y procesador sin memoria cache

- Bloque

- \* Conjunto de palabras al que se accede en una misma petición
- \* Ejemplos
  - Transferencias entre discos y memoria principal
  - Transferencias entre memoria cache y memoria principal

7

Tamaños típicos de palabra: 1B, 2B, 4B, 8B

En los procesadores actuales el tamaño más usado es 8B = 64 bits

Los bloques son conjuntos de palabras consecutivas, que se acceden en una misma petición de lectura o escritura

Ejemplo: entre la memoria principal y la cache se intercambia información por bloques de 4 o 8 palabras consecutivas

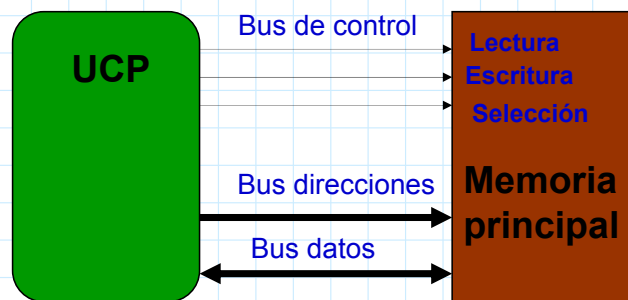
La CPU accede a la cache por palabras individuales, no por bloques.

El tema de la cache se verá en la asignatura ETC de segundo.

## 1.2. Conexión UCP-Memoria

- Líneas del bus

- \* **Control:** La UCP selecciona la memoria y determina la operación (lectura o escritura)
- \* **Direcciones:** La UCP determina la dirección
- \* **Datos:** Según la operación, los datos van:
  - En el caso de la lectura: de la memoria a la UCP
  - En el caso de la escritura: de la UCP a la memoria



8

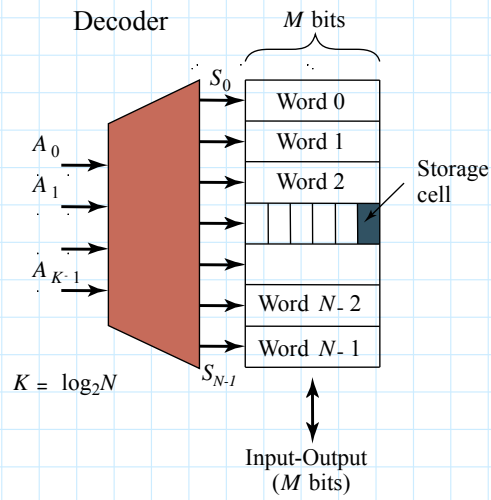
UCP: Unidad Central de Proceso (CPU), es el microprocesador

Observe que las señales de control y dirección son unidireccionales

Las señales del bus de datos son bidireccionales (lectura o escritura)



### 1.3. Estructura básica de las memorias semiconductoras



El decodificador reduce el número de señales de selección  
 $K = \log_2 N$

9

Estructura 2D, array de  $N$  palabras

Selección de las palabras por filas, mediante un decodificador.

Dependiendo del valor del código de direcciones ( $A_0: A_{K-1}$ ), el decodificador selecciona una de las  $N$  palabras.

Cada palabra contiene  $M$  bits, que son leídos o escritos simultáneamente.

O sea, se leen o escriben al mismo tiempo las  $M$  celdas de una palabra.

## 2. Tipos de memorias semiconductoras.

Memoria de lectura/escritura		Memoria de Lectura/escrit. No volátil	Memoria de sólo Lectura No volátil
Acceso aleatorio	Acceso no aleatorio	EPROM E <sup>2</sup> PROM FLASH	Programada por máscara ROM ROM Programable (PROM)
SRAM DRAM	FIFO LIFO Shift Register CAM		

10

La transparencia muestra una clasificación de las memorias semiconductoras, dependiendo de si son de lectura y/o escritura, o si son volátiles/no volátiles

FIFO: First In First Out (Queue)

LIFO: Last In First Out (Stack)

CAM: Content-Addressable Memory, o memoria de contenido direccionable

## 2.1. Memorias no volátiles: ROM/PROM/EPROM/FLASH

Memoria de lectura/escritura		No Volátiles	
Acceso aleatorio	Acceso no aleatorio	Memoria de Lectura/escrit. No volátil	Memoria de sólo Lectura No volátil
SRAM	FIFO	EPROM	Programada por máscara ROM
DRAM	LIFO	E <sup>2</sup> PROM	ROM Programable (PROM)
	Shift Register	FLASH	
	CAM		

11

Se han resaltado a la derecha las memorias no volátiles. Mantienen los datos aunque se desconecte la alimentación.

Unas son de Sólo-lectura: ROM (Read-Only Memories) y PROM (ROM programables)

Otras son de lectura-escritura: EPROM, EEPROM, FLASH

### 2.1.1. Memorias de sólo lectura (ROM)

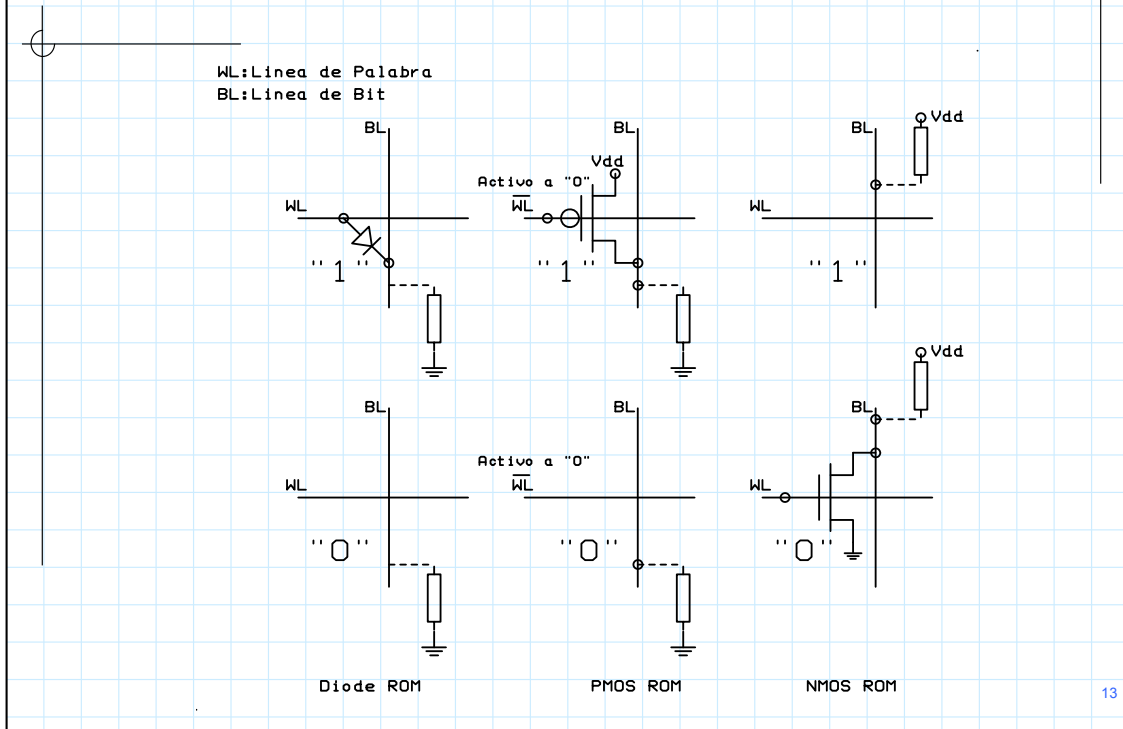
- Las ROM programadas por máscara usan un transistor por bit
  - \* Se programan durante la fabricación.
  - \* La presencia o ausencia del transistor determina el '1' ó '0'
- Son de acceso aleatorio
- Son no volátiles
  - \* Retienen sus contenidos cuando se elimina la alimentación

12

Las memorias de solo lectura almacenan la información como parte de la topología del circuito, por ejemplo añadiendo o eliminando transistores. Puesto que esta topología está prefijada en hardware en tiempo de fabricación (*mask-programmed*), los datos no pueden modificarse, sino que solo pueden ser leídos.

Además la desconexión de la tensión de alimentación no provoca la pérdida de los datos almacenados.

### 2.1.1. Celdas ROM



En la figura se representan diferentes tipos de celdas de memoria ROM, dependiendo del componente que se coloque en la unión fila/columna.

La información de la celda dependerá de la existencia o no del elemento en la unión.

En la fila de arriba, la información almacenada en las celdas es un "1". Al seleccionar la fila con WL, en la línea de bit BL se leerá una tensión alta ("1").

En la fila de abajo, la información almacenada en las celdas es un "0". Al seleccionar la fila con WL, en la columna BL se obtendrá una tensión baja ("0").

La celda más simple es la basada en un diodo situado en la intersección entre la línea de palabra (WL) y la línea de bit (BL).

Suponiendo que BL se conecta con la masa a través de una resistencia, la ausencia del diodo pone un nivel bajo en BL, independientemente del valor de WL. Es un 0 almacenado.

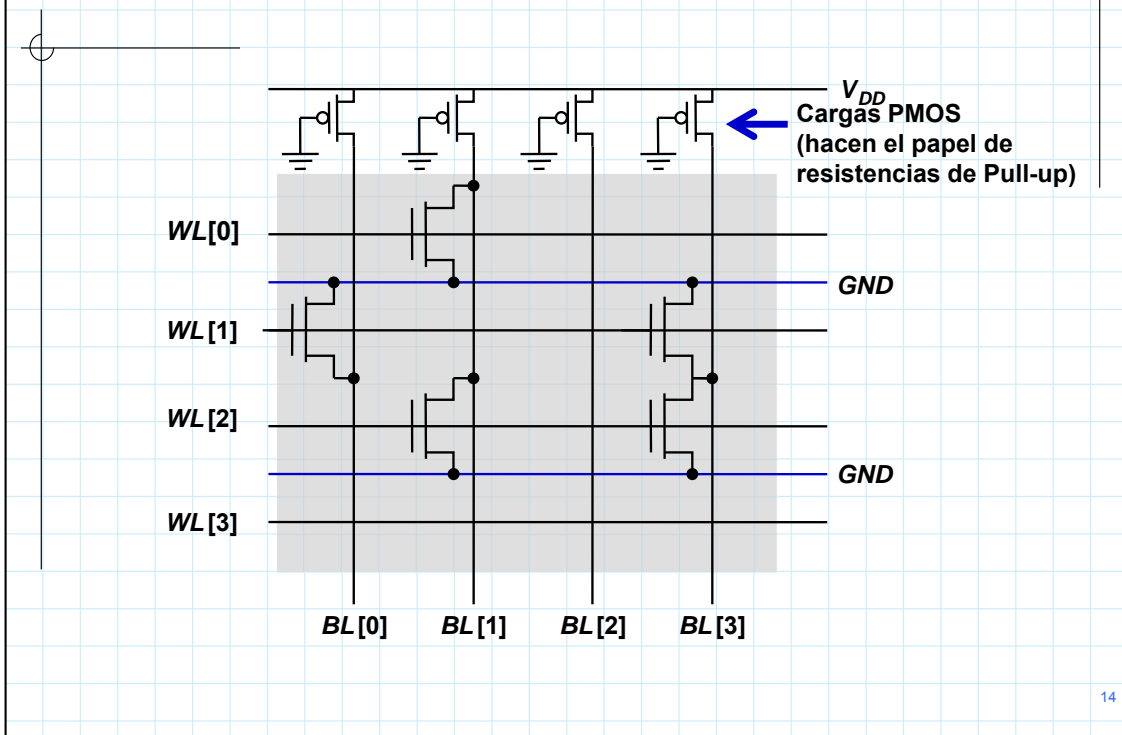
Por el contrario, la presencia del diodo hace que, cuando WL="1" ( $V_{WL}$  es un nivel alto de tensión), el diodo conduzca y en la línea de bit aparezca un nivel alto igual a  $V_{WL} - V_{D(on)}$ , es decir, un "1" lógico. Si WL="0" el diodo no conduce y no afecta a BL.

La alternativa es usar transistores MOSFET en vez de diodos. La operación es idéntica a la de la celda basada en diodo, pero ahora el transistor proporciona mayor corriente de salida para cargar la línea de bit. De este modo se aísla la línea de bit de la línea de palabra y se pueden diseñar ROM de mayor capacidad.

El precio a pagar es que la celda es más compleja y ocupa mayor espacio, debido al contacto adicional con V<sub>DD</sub> o GND.

La figura muestra 2 posibilidades (PMOS ROM o NMOS ROM), dependiendo de si se usan transistores PMOS o NMOS como elementos de conexión. (En el caso de PMOS ROM, la línea de palabra se activaría con nivel bajo.)

### 2.1.1. ROM NMOS con estructura NOR



La figura muestra una estructura NOR, usando en las celdas transistores MOS conectados a masa (NMOS ROM en la transparencia anterior).

Cuando  $WL = "1"$ , los transistores de la palabra conducen y sus correspondientes líneas de bit se conectan a GND ("0"). En caso de que en la celda no exista el transistor, la línea de bit permanece a "1" por defecto, gracias a las cargas PMOS conectadas a VDD. Estas funcionan como resistencias "activas", que tienen un valor mayor que la  $R_{on}$  de los transistores NMOS, para garantizar un nivel bajo adecuado (ya que forman un divisor resistivo). En condiciones normales de operación, sólo una de las líneas de palabra estará a "1".

Cuando  $WL = "0"$ , todos los transistores de la palabra están cortados y no influyen en las líneas de bit.

La estructura NOR se refiere al conjunto de transistores conectados en paralelo a una misma línea de bit, junto con la carga pull-up. Funciona como una puerta NOR NMOS con las líneas de palabra como entradas.

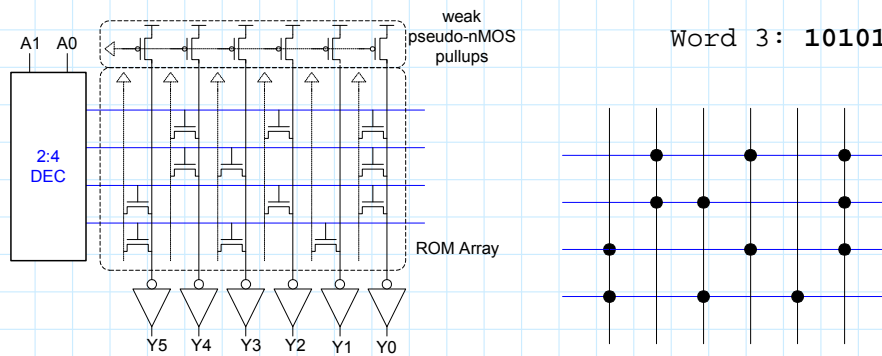
Obsérvese que las conexiones de GND se comparten entre celdas vecinas, reflejando las celdas de las palabras impares respecto al eje horizontal. Así se reduce el cableado ligado a las conexiones metálicas con GND.

### 2.1.1. ROM-Ejemplo

- 4-word x 6-bit ROM

- \* Representación con diagrama de puntos
- \* Los puntos indican “1’s” en ROM

Word 0: 010101  
Word 1: 011001  
Word 2: 100101  
Word 3: 101010



15

Un ejemplo de ROM NOR, formada por 4 palabras de 6 bits cada una. Las palabras están numeradas desde la fila de arriba (word 0) a la de abajo (word 3).

El diagrama de puntos indica dónde se sitúan los transistores en cada intersección.

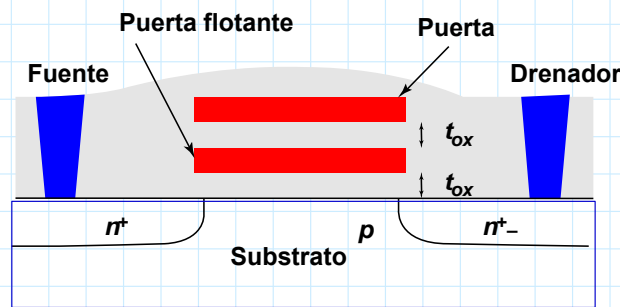
La salida se invierte mediante los inversores Y5-Y0

Obsérvese la No-volatilidad de la ROM: si se desconectara la alimentación, el contenido de la memoria no se perdería. El contenido viene dado por la presencia (“1”) o ausencia (“0”) de los transistores de intersección. Y esto está fijado desde el momento de la fabricación, y no depende de la alimentación.

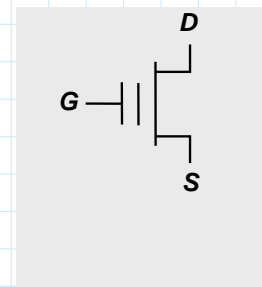
### 2.1.1. Memorias no volátiles reprogramables

#### • Erasable Programmable ROMs

- \* Utilizan una puerta flotante para poner al corte los transistores no deseados
- \* EPROM, EEPROM, Flash



Transistor FAMOS



Símbolo

16

Recordemos que las PROM, PLA y PAL sólo se pueden programar 1 vez.

Para poder REPROGRAMAR varias veces, sin perder la no-volatilidad, surgen nuevas tecnologías:

EPROM, EEPROM, FLASH. Son memorias no volátiles de lectura-escritura (NVRWM), pues además de leerse pueden programarse (escribirse) tras un borrado previo. La programación de la memoria es normalmente más lenta que la operación de lectura.

La celda es una variante del transistor MOS: un transistor MOS de puerta flotante (FAMOS). Además de la puerta de control (gate) habitual, tiene otra puerta rodeada de aislante. Almacenando electrones en la puerta flotante se impide la formación del canal y el transistor está en corte independientemente de la tensión de la puerta normal. Eliminando los electrones de la puerta flotante, el transistor funciona normalmente.

La celda se puede reprogramar efectuando un borrado seguido de una escritura. Esto lo veremos con más detalle a continuación.

La memoria es No volátil porque la carga se almacena en la puerta flotante, rodeada de aislante. Aunque se elimine la alimentación, la carga no se pierde.

Dependiendo de la forma en que se efectúa el borrado (eliminación de la carga de la puerta flotante), hay diferentes tipos de memorias.

La EPROM (Erasable PROM) se borra haciendo incidir luz ultravioleta sobre las celdas de memoria a través de una ventana transparente integrada en el encapsulado. Se generan pares electrón-hueco que descargan la puerta flotante. El proceso es lento, pudiendo tardar entre unos segundos y varios minutos, dependiendo de la intensidad de la fuente UV. El borrado es global. La programación requiere una tensión elevada (12V aprox.) para atraer los electrones a la puerta flotante. La programación es selectiva, por palabras. Tanto el borrado como la programación se realizan fuera del funcionamiento normal, mediante un dispositivo programador especial.

Una vez se ha programado la EPROM, se utiliza como memoria de sólo lectura en el funcionamiento normal, hasta el siguiente borrado-reprogramación.

Las memorias EPROM fueron bastante usadas en aplicaciones que no requerían una reprogramación regular. Posteriormente se substituyeron por las EEPROM y las Flash, que permiten programar eléctricamente durante el funcionamiento normal, sin necesidad de un dispositivo externo.



### 2.1.1. Memorias Flash

- Memorias **no volátiles** de lectura/escritura, semiconductoras
- Aplicaciones
  - \* Pendrives
  - \* Tarjetas de memoria en cámaras digitales
  - \* Reproductores portátiles de audio (MP3)
  - \* Teléfonos móviles
  - \* Discos duros de estado sólido
  - \* ...
- Características
  - \* Pequeñas, baratas, bajo consumo y flexibles
  - \* Basadas en las EEPROM, pero permite el borrado bloque a bloque
  - \* Número limitado de escrituras y borrados
  - \* Tipos: NAND Flash, NOR Flash

17

Toshiba inventó la memoria Flash en 1980, como una nueva tecnología de memoria no volátil semiconductoras

Desde entonces, se ha convertido en una tecnología de almacenamiento muy usada en diversos dispositivos de consumo e industriales.

Son sucesoras de las EEPROM

Consiguen más densidad de integración (bits/chip)

Permiten el borrado por bloques (más rápido, de ahí el nombre de Flash)

Sin embargo hay un problema de fiabilidad: la escritura y el borrado provocan

la degradación progresiva del aislante de puerta, apareciendo corrientes de fuga y

modificándose la tensión  $V_T$  de los transistores. Esto limita el número de escrituras/borrados (entre 10.000 y 100.000) dependiendo de

la precisión del proceso de fabricación y del voltaje necesario para su borrado.

Otras ventajas:

-Al ser memorias semiconductoras, no tienen partes móviles y en consecuencia no tienen problemas de averías en elementos mecánicos, como ocurre en los discos duros mecánicos

-Al ser no volátiles, no requieren alimentación para mantener los datos. El bajo consumo aumenta la vida de las baterías.

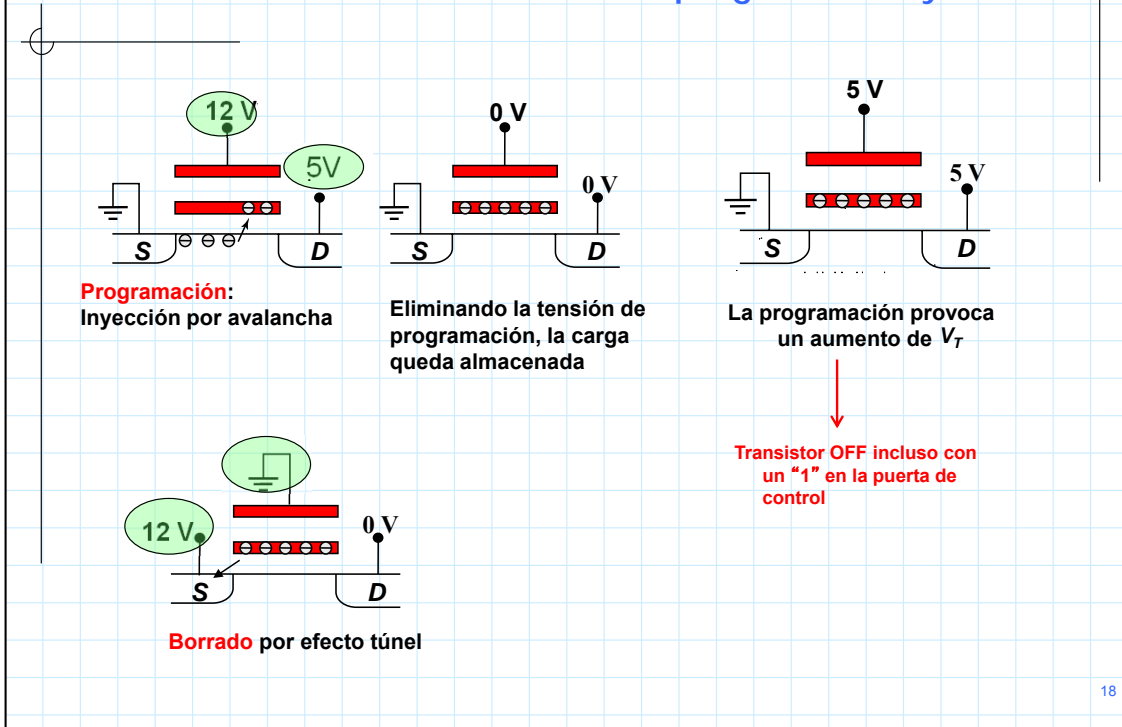
-Más rápidas (tiempo de acceso del orden de microsegundos) que los discos magnéticos (tiempo de acceso del orden de milisegundos).

Tipos:

-NAND Flash: las celdas se disponen con una estructura NAND. Tienen mayor capacidad, y se usan para almacenar datos no volátiles. Ej: pendrive, datos en teléfonos móviles y cámaras digitales, discos de estado sólido (SSD)...

-NOR Flash: las celdas se disponen con una estructura NOR. Tienen menor capacidad, y se acceden por palabras. Ej: S.O en móviles, ROM BIOS en los computadores...

### 2.1.1. Celda Flash: mecanismos de programación y borrado



Celda básica = **transistor FAMOS (Floating-gate Avalanche-injection MOS)**

Tiene una **puerta flotante**, entre la puerta de control y el sustrato

Ambas puertas rodeadas por aislante muy fino

La idea es almacenar electrones en la puerta flotante, aplicando una tensión positiva elevada (mayor de lo normal, típicamente unos 12V) en la puerta de control.

Esto, junto a la estrechez de la capa de aislante,

genera un campo eléctrico muy intenso que consigue que los electrones atraviesen

la fina capa de aislante. Al recuperar las tensiones normales, la carga almacenada

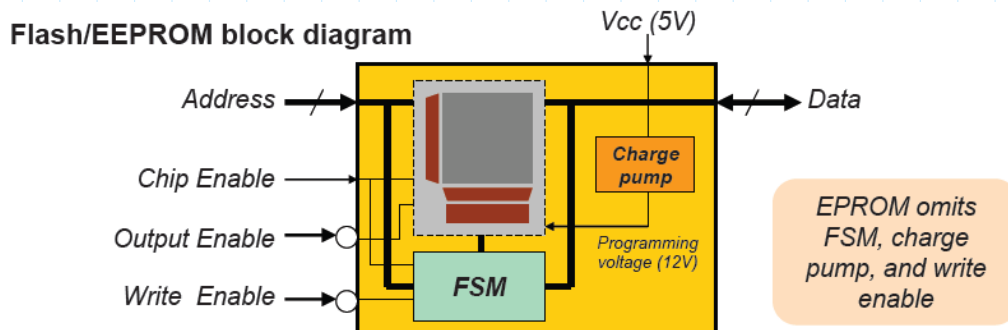
no se pierde, pues la puerta flotante está rodeada de aislante. De ahí la **no-volatilidad**.

Si se quiere descargar la puerta flotante, se aplica la tensión con la polaridad inversa, lo que provoca el vaciado de la puerta flotante.

2 mecanismos físicos parecidos:

- Escritura (programación) mediante *hot-electron injection*: los e- pasan desde el sustrato a la puerta flotante.
- Borrado mediante *efecto túnel Fowler-Nordheim*: los e- abandonan la puerta flotante hacia la zona de fuente del transistor.

### 2.1.1. Estructura interna chips memoria Flash



19

- La lectura de flash (E)EPROM o Flash es igual que la de SRAM
- Vpp: entrada para el voltaje de programación (12V)
- EPROM: Vpp es proporcionada por una máquina externa de programación
- Los chips flash/EEPROM generan 12V mediante un circuito especial interno (charge pump-bomba de carga), encargado de generar tensiones superiores a la tensión de alimentación, sin consumir mucha corriente.
- EPROM no tiene write enable: se borra y se programa entera en programador externo.
- En la flash y EEPROM, la secuencia de escritura es controlada por un FSM (Finite State Machine-Autómata de Estados Finitos) interno
- En las escrituras se envían señales al FSM
- Aunque se usan las mismas señales, el cronograma de escritura en flash/EEPROM es diferente a SRAM. En flash/EEPROM, un ciclo de programación (escritura) incluye un ciclo de borrado previo, que puede incorporar a su vez varios pasos de monitorización para comprobar que todas las celdas del bloque han sido borradas. Esto es necesario porque los diferentes  $V_T$  iniciales de las celdas, así como las variaciones en el espesor del óxido, pueden provocar diferencias en el  $V_T$  de las celdas al final del borrado.

## 2.2. Memorias volátiles de acceso aleatorio: RAM

Memoria de lectura/escritura		Memoria de Lectura/escrit. No volátil	Memoria de sólo Lectura <del>No volátil</del>
Acceso aleatorio	Acceso no aleatorio	EPROM E <sup>2</sup> PROM FLASH	Programada por máscara ROM ROM Programable (PROM)
SRAM DRAM	FIFO LIFO Shift Register CAM		
Volátiles RAM			

20

Se han resaltado a la izquierda las memorias volátiles. Pierden los datos tras desconectar la alimentación.

## 2.2. Características de las memorias RAM

### Memoria RAM: (*Random Access Memory*)

- Memorias volátiles
- Operaciones de lectura y escritura
- Acceso aleatorio

Dependiendo de la celda básica

{ RAM estáticas (**SRAM**)  
RAM dinámicas (**DRAM**)

## 2.2. Clasificación: Memorias SRAM y DRAM

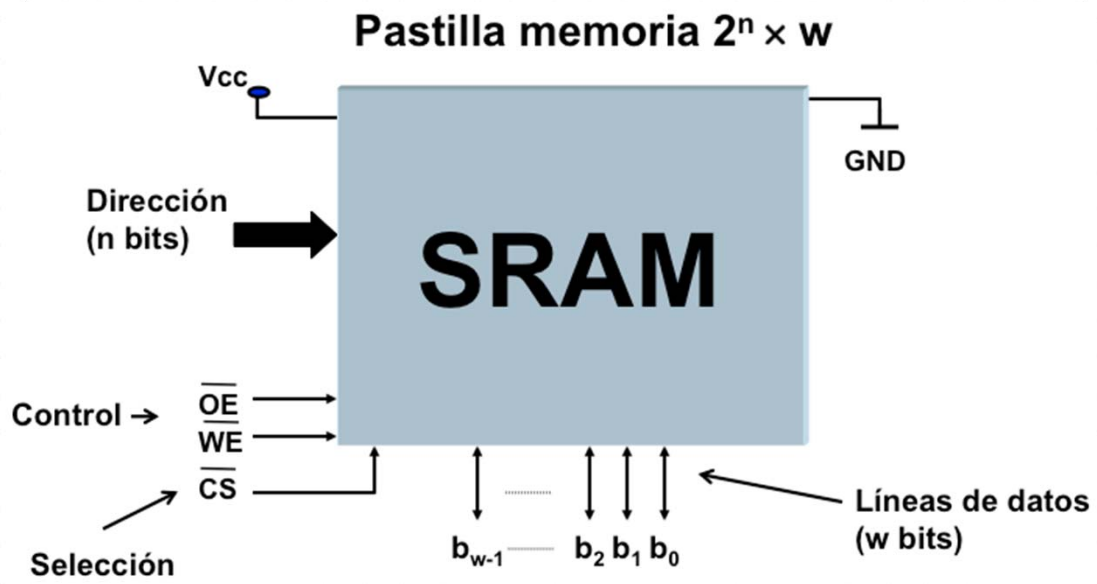
- Estáticas (Static RAM = SRAM)
  - \* Basadas en biestables
  - \* Datos almacenados mientras hay alimentación
  - \* Tamaño de celda grande (6 transistores/celda)
  - \* Rápidas (caches)
- Dinámicas (Dynamic RAM = DRAM)
  - \* Basadas en la carga/descarga de capacidad estructural
  - \* Requieren un refresco periódico
  - \* Tamaño de celda pequeño (1 a 3 transistores)
    - Más bits/chip
  - \* Más lentas (memoria principal)

22

SRAM se usan en la Cache, que suele estar integrada en la CPU. Son más rápidas, pero tienen menor capacidad (del orden de MB). La celda es un biestable (latch).

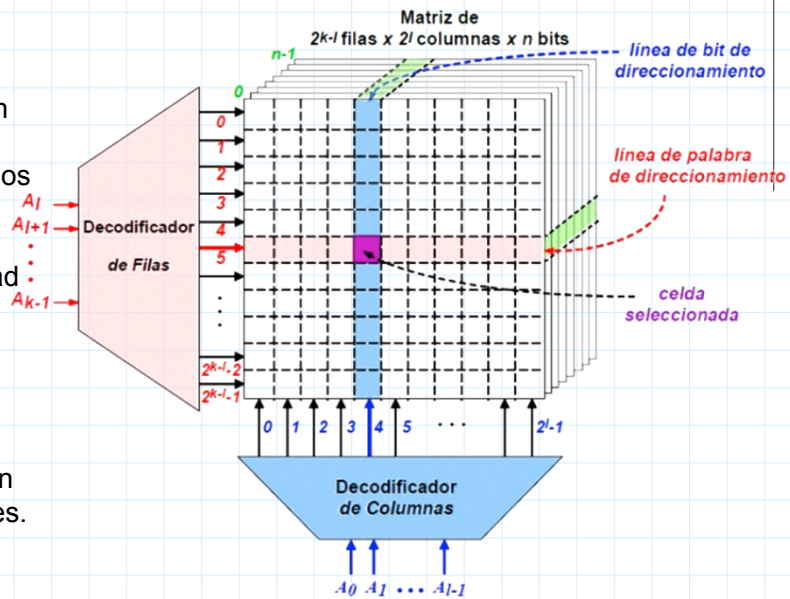
DRAM se usa en la Memoria Principal, externa a la CPU. Son más lentas, pero tienen mayor capacidad (del orden de GB). La celda es un condensador integrado y un transistor de paso. Necesita refresco periódico (del orden de ms), pues en caso contrario el dato se pierde por la descarga del condensador debido a corrientes de fuga. Por ello se llaman dinámicas.

### 2.2.1. RAM estática. Líneas externas de una SRAM



## 2.2.1. RAM estática. Estructura en ARRAY 2D

- Las líneas de dirección se dividen aprox. en dos mitades, para que los decodificadores de fila y de columna sean de complejidad similar.
- Cada dirección selecciona una palabra completa, que se distribuye en  $n$  planos semejantes.
- La memoria es un array de  $2^{K-I}$  filas x  $2^I$  columnas x  $n$  bits (cada bit está en un plano diferente)



(Imagen con licencia de creative commons. Obtenida de: <http://es.slideshare.net/manuelbarcell/tema08-37160685> )

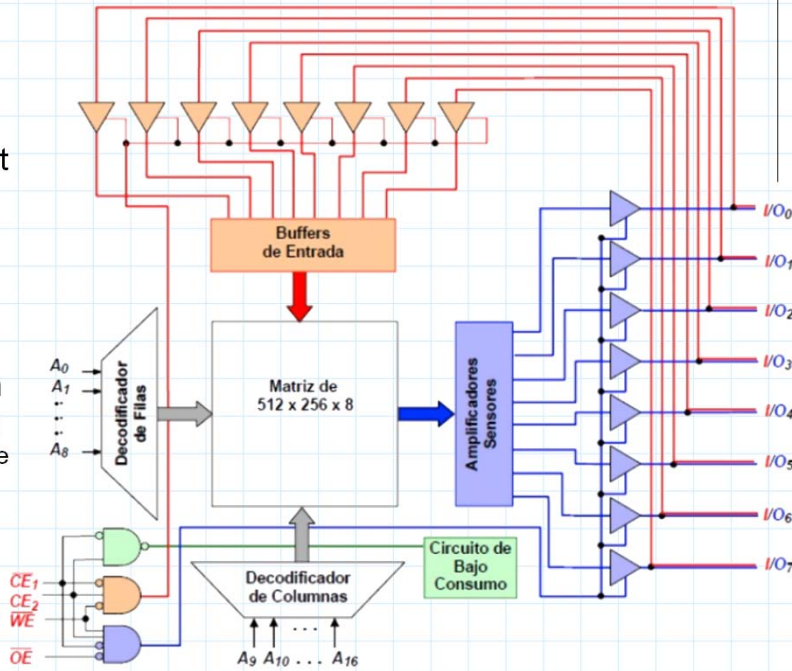
24

Selección de la palabra por filas y columnas. ( $K-I$  líneas para las filas,  $I$  líneas para las columnas) Permite disminuir el tamaño de los decodificadores, cuando el número de palabras es muy grande. El decodificador de filas activa una de sus salidas. Lo mismo hace el decodificador de columnas. Se selecciona aquella palabra correspondiente a la intersección de las dos salidas de los decodificadores.



## 2.2.1. RAM estática. Estructura en ARRAY 2D (II)

- La entrada y la salida de cada bit utilizan amplificadores y buffers con triestado.
- Las líneas de control gestionan la entrada/salida:
  - CS: selección de chip.
  - OE: habilitación de la salida (lectura).
  - WE: habilitación de escritura.



(Imagen con licencia de creative commons. Obtenida de: <http://es.slideshare.net/manuelbarcell/tema08-37160685> )

25

La gestión de la entrada o la salida de la información utiliza buffers con tri-estado que permiten habilitar la salida para lectura (OE), o la entrada, para la escritura(WE).

El ejemplo de la figura corresponde a una memoria de 128KBytes.

Se distribuye en 9 líneas para filas, 8 líneas para columnas, y 8 planos de bit.

Observe los circuitos de salida y las señales de control típicas:

-Buffers triestado de entrada/salida para conectarse al bus de datos bidireccional común, compartido por otros chips (memoria, E/S, etc.)

-Lógica de control de escritura que activa los buffers de entrada

-Lógica de control de lectura que activa los buffers de salida

-Varias señales de control típicas:

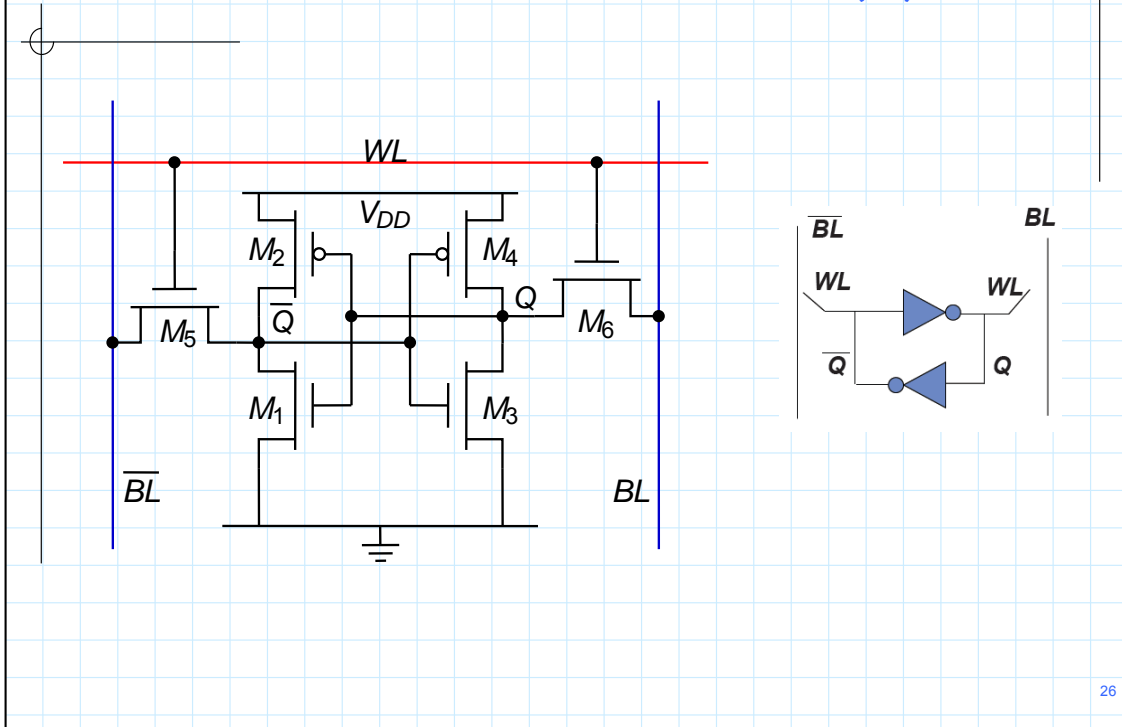
(/CE<sub>1</sub> y CE<sub>2</sub>):Chip enable: activan o desactivan los buffers.

En caso de desactivación, la salida queda en alta impedancia y el chip está desconectado del bus de datos y entra en modo de bajo consumo.

(/WE):Write enable: da paso a los datos externos para que sean escritos.

(OE): Output enable: da paso a los datos internos para que salgan al exterior

### 2.2.1. RAM estática. Celda de memoria SRAM (6T)



#### Estructura:

M5 y M6 son dos puertas de transmisión NMOS para acceder (en lectura o escritura) al ***latch (flip-flop) interno***.

M1-M2, M3-M4 son dos inversores CMOS realimentados para formar el ***latch***, que almacena el estado de la celda.

BL y /BL son las **líneas de bit complementarias**, donde se sitúan los datos para escribir, o se leen los datos.

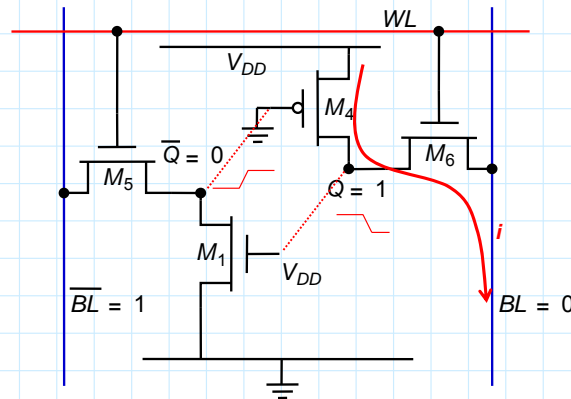
WL es la **línea de selección de palabra**. Su cometido es seleccionar las celdas de una misma palabra para lectura o escritura.

Cuando WL="1" los transistores de paso M5 y M6 se cierran y se accede a la celda.

Cuando WL="0", M5 y M6 se abren y la celda está aislada, manteniendo el estado mientras se mantenga la alimentación, gracias a la **realimentación** entre los dos inversores.

Obsérvese que la celda es un **latch asíncrono**. No se usa un Flip-flop porque ocuparía el doble de transistores. La sincronización con una señal de reloj se puede hacer externamente a las celdas, introduciendo registros síncronos para las direcciones, datos y control. Así se consigue una mayor velocidad en el acceso por parte de la CPU. Se trata de las SSRAM (SRAM síncronas).

## 2.2.1. RAM estática. Celda de memoria SRAM (6T). Operación de Escritura



Supongamos que  $Q = "1"$ ,  $/Q = "0"$ , y queremos escribir un "0"

1) Se sitúan los datos complementarios en BL y /BL

2) Se selecciona la celda:  $WL = "1"$

3) El biestable cambia de estado :

Es suficiente con que Q quede por debajo de  $V_{DD}/2$

La realimentación hace que /Q pase a "1" y Q a "0"

27

### Escritura:

Se sitúan los datos complementarios en BL y /BL, y se activa  $WL = "1"$ . El estado de los transistores comienza a fluctuar y la realimentación positiva del latch hace el resto.

Por ejemplo, si  $Q = "1"$  y  $/Q = "0"$ , y queremos escribir un "0", pondremos  $BL = "0"$ ,  $/BL = "1"$  y activaremos  $WL = "1"$ . Es suficiente con que Q baje de  $V_{DD}/2$  (debido a la conducción de M4 hacia BL) para que el umbral de conmutación del FF se sobrepase, /Q sobrepase  $V_{DD}/2$  y la realimentación conmute el estado.

Q pasa a 0, M1 se corta y M2 conduce, con lo que /Q pasa a "1". Queda almacenado el nuevo dato.

Es importante el tamaño relativo de los transistores, que determina la resistencia relativa de los mismos. Así, debe cumplirse que:

$R_{M6} < R_{M4} \rightarrow (W/L)_{M6} > (W/L)_{M4}$  para asegurar que Q pase a "0"

$R_{M5} > R_{M1} \rightarrow (W/L)_{M5} < (W/L)_{M1}$  para asegurar que /BL="1" no afecte al estado bajo inicial de M1

## Operación de Lectura



- 1) BL y /BL se precargan a VDD
- 2) Se selecciona la celda: WL = "1"
- 3) /BL se descarga por M5 y M1 y pasa a "0"

28

BL y /BL se **precargan** a  $V_{DD}$  (las dos a la misma tensión!).

Si Q="1", /Q="0":

BL permanece sin cambios, en su valor de precarga “1”, pues M3 está cortado.

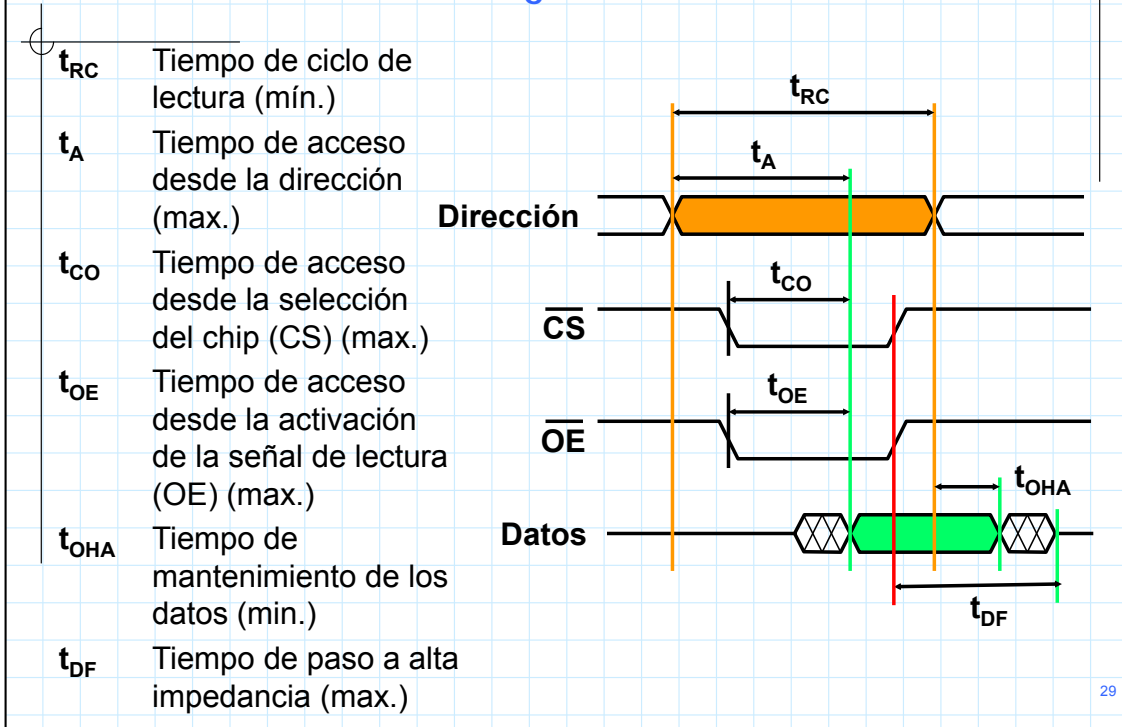
BL se descarga a través de M3-M6, y pasa a “0”

/BL permanece sin cambios, en su valor de precarga “1”, pues M1 está cortado.

Tamaño relativo de los transistores para asegurar lecturas y escrituras correctas:

Ejemplo:  $(W/L)_{NMOS} = 4$ ,  $(W/L)_{PASO} = 2$ ,  $(W/L)_{PMOS} = 1$

## 2.2.1. RAM estática. Cronograma de lectura de una SRAM



Antes de empezar a ver cronogramas conviene explicar los tipos de líneas que aparecen en ellos.

Líneas individuales: Puede estar a nivel alto, nivel bajo, (CS o OE) o alta impedancia (Datos)

Bus: líneas paralelas que indican que contiene más de una línea y que cada una puede estar a uno o a cero. (Dirección o Datos)

Líneas que pueden cambiar en cualquier momento (CS o Datos).

Niveles estables en buses indicados por un color opaco.

Niveles cambiando en buses (Aspas en Dirección o en datos)

Ahora ya podemos explicar el cronograma de lectura simplificado de una SRAM. Para ello conviene primero indicar que es una comunicación asíncrona entre dos elementos,

CPU y Memoria, siendo la CPU la que manda. La CPU inicia la petición de lectura dejando la dirección de donde quiere leer en el bus de direcciones. Acto seguido la CPU activa las líneas CS y OE en cualquier orden. Cuando la línea CS sea bajada por la CPU, la memoria se dará cuenta de que se está realizando una operación sobre ella y mirará en qué estado está la línea OE y WE. La línea que ese momento esté a 0 indicará el tipo de operación.

Tipos de tiempos:

Tiempo de ciclo: tiempo mínimo que ha de pasar entre una operación de lectura y la siguiente.

Tiempo de acceso: tiempo máximo que tardará la memoria en proporcionar los datos a partir de:

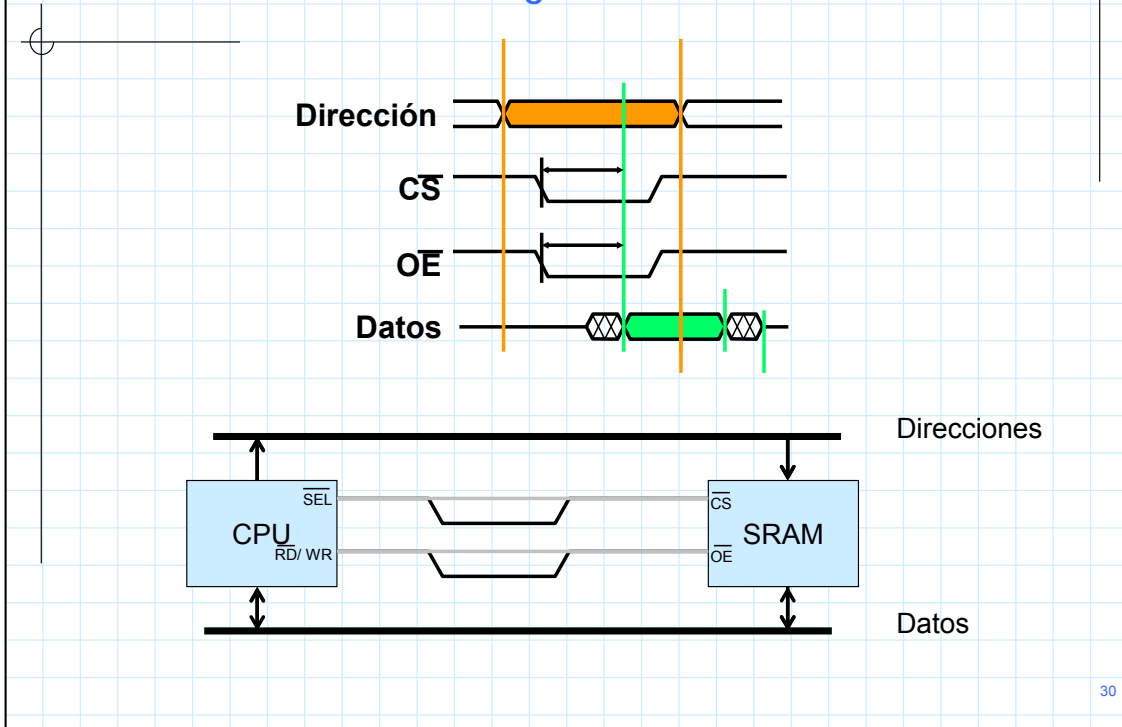
direcciones estables:  $t_A$

activación de CS:  $t_{CO}$

activación de lectura OE  $t_{OE}$

Tiempo de Mantenimiento: tiempo que se mantendrán los datos estables en el bus después de que cambien las direcciones para iniciar una nueva operación.

### 2.2.1. RAM estática. Cronograma de lectura de una SRAM

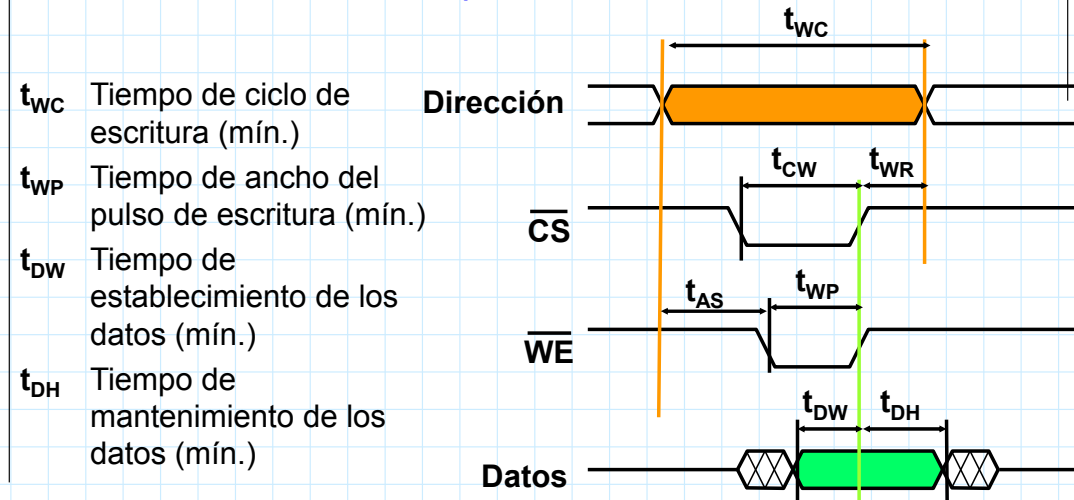


En esta figura se puede observar la relación maestro-esclavo siendo el maestro la CPU, ya que manda de la operación, y el esclavo la SRAM, ya que realiza la operación que la CPU le indica.

- 1) se dejan las direcciones
- 2) se activa CS
- 3) se activa el tipo de operación (lectura  $\rightarrow$  OE = 0)
- 4) la memoria deja los datos correspondientes a la dirección indicada en el bus de datos

## 2.2.1. RAM estática. Cronograma de escritura de una SRAM

### - Escritura controlada por WE



31

En el caso de la lectura, quien dejaba los datos en el bus de datos era la memoria, pero en esta ocasión quien deja los datos en el bus es la CPU para que la memoria los escriba en las celdas. Puede haber dos temporizaciones diferentes:

Escritura controlada por la línea WE o por CS. En esta ocasión veremos como ejemplo una escritura controlada por WE:

- 1) la CPU coloca las direcciones
- 2) la CPU activa CS
- 3) la CPU activa WE. A partir de este momento la memoria se prepara para la escritura, necesitando un tiempo mínimo para almacenar correctamente el dato ( $t_{WP}$ ,  $t_{CW}$ ).
- 4) cuando WE vuelve a subir la memoria empieza la escritura realmente.

Tiempo de ciclo:  $t_{WC}$  tiempo mínimo entre una operación de escritura y la siguiente operación

Tiempo de set up: tiempo que tienen que estar estables los datos en el bus antes de ...

$t_{AS}$ : direcciones estables antes de activación de CS y WE

$t_{DW}$ : datos estables antes del fin de escritura (subida de WE)

Tiempo de mantenimiento: tiempo que tienen que estar estables los datos en el bus después de ...

$t_{WR}$ : direcciones estables después del fin de escritura (subida de WE) o CS

$t_{DH}$ : datos estables después del fin de escritura (subida de WE)

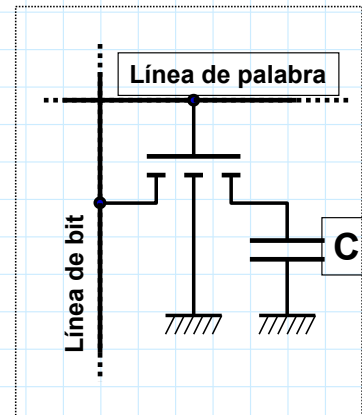
La escritura controlada por CS es igual pero invirtiendo los pasos segundo y tercero y tomando como referencia los flancos de CS para los tiempos de mantenimiento y de set up

### 2.2.2. RAM dinámica. Celdilla básica DRAM-1T.

**Celdilla básica: Condensador** →

#### **Tecnología:**

- Requieren refresco (ms)
- Menor velocidad
- Muy alta densidad de integración
- Baratas (coste/bit almacenado)



**Capacidad chips comerciales: N M x 4 bit**

**N M x 8 bit**

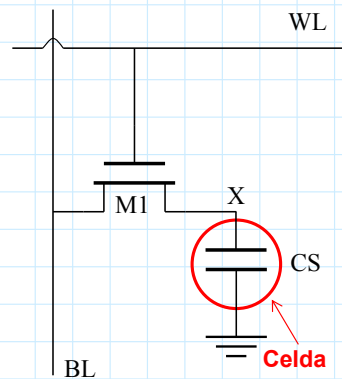
**N M x 16 bit**

**N= 1 ... 1024**

DRAMs son de 4 a 8 veces más capaces que las SRAMs, si bien estas últimas son de 8 a 16 veces más rápidas, pero también de 8 a 16 veces más caras.



### 2.2.2. RAM dinámica. Celdilla básica DRAM-1T.



Dato almacenado como carga de una capacidad  $C_s$   
Gran densidad de integración (1T y  $C_s$  muy pequeña)  
Necesidad de refresco periódico

33

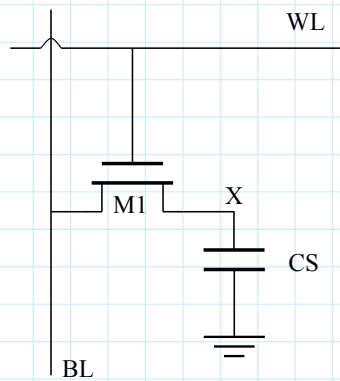
Máxima reducción de la complejidad de la celda.

El dato se almacena en una **capacidad explícitamente incluida en el diseño del transistor M1** ( $C_s$  en la transparencia).  $C_s$  del orden de fF, muy pequeña.

Necesidad de **refresco** para **evitar la pérdida de carga debido a las corrientes de fuga**.  
**Refresco periódico (cada pocos milisegundos) = lectura + escritura del contenido de las celdas.**

La denominación de **dinámica** se debe a que las celdas se basan en la **carga almacenada en una capacidad y que necesitan refresco**.

## DRAM 1T- Escritura



### Escritura:

- 1) Dato en BL
- 2) Se activa WL:

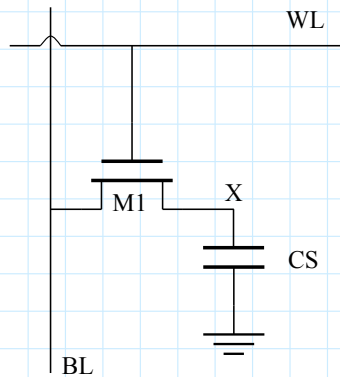
Cs se carga o descarga, dependiendo del dato en BL

34

### Escritura

El dato es situado en BL, y WL se pone a "1" ( $V_{DD}$ ), cerrando M1. Dependiendo del valor, Cs se carga o descarga.

## DRAM 1T- Lectura



### Lectura:

- 1) Precarga de BL a  $V_{DD}/2$
- 2) Se activa WL:

Redistribución de carga entre  $C_{BL}$  y  $C_S$   
 $\Delta V$  en BL, que debe amplificarse

Lectura destructiva → se necesita reescritura

35

### Lectura

BL se **precarga** a  $V_{DD}/2$ . WL se pone a “1”. Se produce una **redistribución de carga entre  $C_S$  y  $C_{BL}$**  (la capacidad parásita asociada a BL). Esto resulta en un **cambio de tensión en BL**, que será **positivo o negativo** dependiendo de la tensión que había en  $C_S$ . Este cambio es muy pequeño (típicamente 250mV) y es amplificado en la salida mediante una circuitería de amplificación (*sense amplifier*).

$$\Delta V = V_{BL} - V_{PRE} = V_{BIT} - V_{PRE} \frac{C_S}{C_S + C_{BL}}$$

$C_{BL}$  es del orden de 1pF y  $C_S$  del orden de 30fF, o sea que  $C_S \ll C_{BL}$   
 (1pF=10<sup>-12</sup>F, 1fF=10<sup>-15</sup>F)

La lectura en DRAM 1T es **destructiva**. El valor leído debe ser reescrito de nuevo. La lectura y el refresco están intrínsecamente entrelazados.

## Resumen

En el tema 5 se ha estudiado:

- Memorias no volátiles, centradas en ROM y Flash
- Memorias SRAM, que se utilizan en la memoria cache de los microprocesadores
- Celda básica DRAM. Las memorias DRAM se utilizan en la memoria principal de los microprocesadores.

La memoria DRAM será estudiada en detalle en la asignatura ETC de segundo curso