

# Resumen-COMPLETO-Segundo-Parcial...



**Alvaroo\_04**



**Tecnología de Computadores**



**1º Grado en Ingeniería Informática**



**Escuela Técnica Superior de Ingeniería Informática  
Universidad Politécnica de Valencia**

**WUOLAH + BBVA**

**Llévate**

**15€**

Al abrir tu Cuenta Online Sin Comisiones y hacer una compra superior a 15€.

**1/6**

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

Cuéntame más



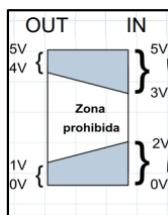
¿Cómo? →



### SEGUNDO PARCIAL TCO – RESUMEN COMPLETO

#### TEMA 3 – FAMILIAS LÓGICAS INTEGRADAS

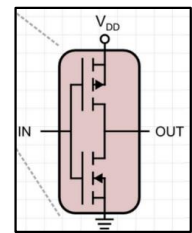
##### INTRODUCCIÓN



La puerta lógica es un circuito electrónico que funciona a modo de operador booleano. Los circuitos más básicos son: *AND*, *OR*, y *NOT*, que corresponden con los operadores *producto*, *suma* y *negación*.

La lógica booleana está definida únicamente para dos elementos, representados en electrónica digital como "0" y "1" lógicos.

Se establece una zona prohibida alrededor del límite para poder discriminar entre el "0" o el "1", la cual se corresponde con un nivel lógico erróneo, es decir, ni "0" ni "1".



Ahora bien, debemos saber que son las **familias lógicas**, las cuales son un conjunto de elementos funcionales con el mismo circuito base y la misma tecnología de fabricación. En ellas, identificamos tres familias lógicas principales: *bipolares*, *MOS*, y *BiCMOS*.

A su vez, debemos tener en cuenta el concepto de las **escalas de integración (SI)**, el cual sigue la secuencia Small, Medium, Large, Very Large, Ultra Large y Giga e indica el número de puertas lógicas por chip.

<b>SSI</b>	<10p
<b>MSI</b>	10-100p
<b>LSI</b>	100-1000p
<b>VLSI</b>	1000-100.000p
<b>ULSI</b>	100.000-1.000.000p
<b>GSI</b>	>1.000.000p

A fin de usar los circuitos integrados, debemos usar un encapsulado, el cual sirve para proteger el chip, disipar el calor y establecer interconexiones.

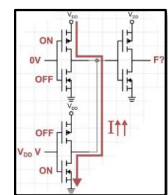
##### TIPOS DE SALIDAS

Existen diferentes tipos de salida en función de los posibles valores de salida:

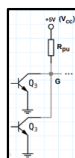
##### ESTÁNDAR: "0" Y "1"

Las salidas normales **no** se pueden conectar entre sí. En caso de tener valores lógicos diferentes, se provoca un conflicto lógico que genera tensiones indeterminadas y corrientes excesivas.

Este hecho ocurre en cualquier tecnología cuyas salidas se establezcan con baja impedancia.



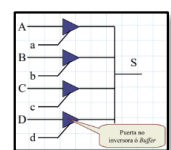
##### COLECTOR/DRENADOR ABIERTO: "0" Y Z



1. Permite el cableado lógico directo.
2. Se requiere una resistencia externa (denominada de *pull-up*) para obtener el nivel alto.
3. Las salidas forman una AND cableada.

##### TRIESTADO: "0", "1" Y Z

1. Permite la conexión directa de las salidas.
2. La salida puede tener tres posibles estados lógicos:
  - a. '0' y '1' de baja impedancia
  - b. Alta impedancia (H.Z. o Z\*)
3. Aplicación de buses de datos.



## TENSIÓN DE ALIMENTACIÓN

Un parámetro muy importante es la tensión de alimentación. Esta es necesaria para hacer funcionar a los transistores (y otros dispositivos) que constituyen los circuitos lógicos. Tiene nombres diferentes en función de la tecnología, los más habituales son: ( $V_{cc}$ , GND), ( $V_{DD}$ ,  $V_{SS}$ ) y ( $V+$ ,  $V-$ ).

Los valores típicos en chips SSI son:

- **TTL:** 5V
- **CMOS:** rango variable, típicamente entre 2-6V. También entre 3-15V.
- **VLSI:** 1-2.5V

## RANGOS DE TENSIONES

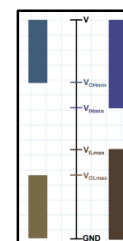
### SALIDA

- $V_O$ : Voltaje en una salida.
- $V_{OH}$ : Voltaje en una salida para un '1' lógico.
- $V_{OH_{min}}$ : Voltaje mínimo en una salida para un '1' lógico.
- $V_{OL}$ : Voltaje en una salida para un '0' lógico.
- $V_{OL_{max}}$ : Voltaje máximo en una salida para un '0' lógico.



### ENTRADA

- $V_I$ : Voltaje en una entrada.
- $V_{IH_{min}}$ : Voltaje mínimo necesario en una entrada para ser considerado un '1' lógico.
- $V_{IL_{max}}$ : Voltaje máximo necesario en una entrada para ser considerado un '0' lógico.



La separación entre los valores de entrada y de salida nos proporciona información sobre la inmunidad al ruido eléctrico.

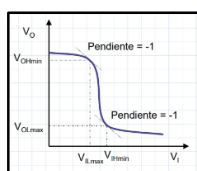
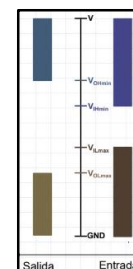
## MARGEN DE RUIDO

Es la diferencia entre los voltajes de entrada y salida. Es por ello que podemos calcularlo de la siguiente forma:

$$NM_H = V_{OH_{min}} - V_{IH_{min}} \quad NM_L = V_{IL_{max}} - V_{OL_{max}}$$

Se define el margen de ruido de una familia lógica como el mínimo entre los dos niveles. Siempre, tendremos interés en obtener un valor de NM alto.

A veces, la señal de salida se verá perturbada por una interferencia electromagnética. Algunas variaciones serán absorbidas por la propia inmunidad al ruido y no provocarán fallas, llegando a regenerarse incluso después de haber invadido la zona de indeterminación.



Los puntos de la curva de transferencia con pendiente = -1 marcan los límites  $V_{IL_{max}}$  y  $V_{IH_{min}}$ .

Si proyectamos estos en el eje vertical, obtendremos  $V_{OH_{min}}$  y  $V_{OL_{min}}$ .



**WUOLAH + BBVA**

**1/6**

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

# Llévate

# 15€

1

Abre tu Cuenta Online sin comisiones ni condiciones

2

Haz una compra igual o superior a 15€ con tu nueva tarjeta

3

BBVA te devuelve 15€

Cuéntame más



¿Cómo? ➔

**WUOLAH + BBVA**

## CORRIENTES

Por convenio, siempre dibujaremos las corrientes dirigidas hacia la puerta, tanto en las entradas como en las salidas. Definiciones:

- $I_I$ : Corriente en la rama conectada a una entrada, cada entrada se considera por separado.
- $I_O$ : Corriente en la rama conectada a una salida de una puerta.
- $I_{IL_{max}}$ : Corriente máxima "exigida" por la rama de entrada cuando la tensión en la misma es considerada un '0'.
- $I_{IH_{max}}$ : Corriente máxima "proporcionada" por la rama de entrada cuando la tensión en la misma es considerada un '1'.
- $I_{OL_{max}}$ : Corriente por debajo de la cual en la salida se garantiza un '0' lógico.

$$\uparrow |I_{OL}| \rightarrow \uparrow V_{OL} \quad |I_{OL}| \leq |I_{OL_{max}}| \quad V_{OL} \leq V_{OL_{max}}$$

- $I_{OH_{max}}$ : Corriente por debajo de la cual en la salida se garantiza un '1' lógico.

$$\uparrow |I_{OH}| \rightarrow \downarrow V_{OH} \quad |I_{OH}| \leq |I_{OH_{max}}| \quad V_{OH} \geq V_{OH_{max}}$$

## FAN-OUT/FAN-IN

### FAN-OUT

El *fan-out* es un parámetro de una familia lógica que establece el número de entradas de la puerta lógica que pueden conectarse a una salida. Para su cálculo, se considera que únicamente hay entradas de otras puertas de la **misma** familia.

A su vez, se calcula como la relación entre las corrientes de salida y de entrada:

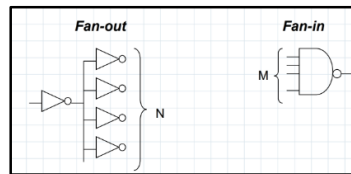
$$\bullet \quad \text{Fan-out}(L) = \frac{|I_{OL_{max}}|}{|I_{IL_{max}}|} \quad \bullet \quad \text{Fan-out}(H) = \frac{|I_{OH_{max}}|}{|I_{IH_{max}}|}$$

El fan-out de la familia es el mínimo de ambos.

### FAN-IN

El *fan-in* es un parámetro de una familia lógica que establece el número de entradas que puede tener un componente sin que se desvirtúen los márgenes lógicos.

Cada entrada supone una capacidad parásita. A mayor el número de entradas, menor velocidad de funcionamiento.



## PARÁMETROS TEMPORALES

### VOLTAGE SWING

Diferencia entre los valores de tensión máximo y mínimo en la salida:  $V_{OH_{max}} - V_{OL_{min}}$

### PROPAGATION DELAY

Tiempo transcurrido desde el paso de un flanco en la entrada por el 50% de su recorrido hasta el correspondiente paso por el 50% del recorrido en el flanco en la salida. Se definen:

- $t_{p_{HL}}$ : transición de salida de '0' a '1'.
- $t_{p_{LH}}$ : transición de salida de '1' a '0'.

### RETARDO DE PROPAGACIÓN MEDIO

$$t_p = \frac{t_{p_{LH}} + t_{p_{HL}}}{2}$$

### TIEMPO DE TRANSICIÓN

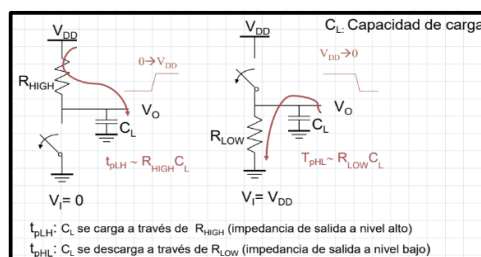
Refleja la velocidad de conmutación entre el 10% y el 90% del recorrido de una señal. Se define: *tiempo de subida* ( $t_r$ ) y *tiempo de bajada* ( $t_f$ ).



## MODELO DE RETARDO DE PROPAGACIÓN

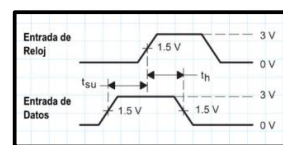
Interesa que el producto RC (Constante de Tiempo) sea el menor posible. Eso se cumple con:

- Baja impedancia de salida.
- Bajas capacidades parásitas.



## PARÁMETROS TEMPORALES SECUENCIALES

- $f_{\max}$ : Frecuencia máxima de funcionamiento. Superarla supone problemas de temporización y disipación.
- $T_w$ : Ancho de pulso mínimo. Si en la entrada se inyecta un pulso, éste ha de reflejarse en la salida.
- $t_{su}$ : Tiempo de establecimiento (*setup time*). Tiempo que debe permanecer estable una entrada **antes** de la aparición del flanco.
- $t_h$ : Tiempo de mantenimiento (*hold time*). Tiempo que debe permanecer estable una señal **después** de la aparición del flanco.



## POTENCIA

### POTENCIA ESTÁTICA

Potencia disipada cuando la salida está estabilizada. TTL y NMOS tienen consumo estático, mientras que CMOS consigue eliminar la potencia estática.

$$P_{est} = \frac{P_L + P_H}{2} = \frac{V_{cc} \cdot I_{ccl} + V_{cc} \cdot I_{cch}}{2}$$

### POTENCIA DINÁMICA

Potencia disipada durante las transiciones de la salida. En CMOS:

$$P_{din} = V_{cc}^2 \cdot f \cdot C_L$$

## PRODUCTO RETARDO-POTENCIA

Potencia y retardo son parámetros contrapuestos: familias con retardos pequeños tienen consumo elevado y viceversa.

El *PDP* (*Power-Delay Product*) es el factor de mérito que indica cómo se combinan el consumo y el retardo en una familia lógica.

$$PDP = P(mW) \cdot t(ns)$$

### INTERCONEXIÓN DE COMPONENTES ELÉCTRICAS O ESTÁTICAS (DC)

#### ESPECIFICACIONES ESTÁTICAS

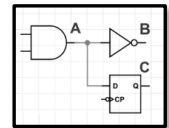
Para saber si es posible realizar una conexión, debemos fijarnos en:

- Tipo de circuito de la etapa de salida de los componentes digitales. Conexión entre salidas:
  - Colector/Drenador abierto + R<sub>pu</sub>
  - Triestado
- La compatibilidad de las tensiones.
- La compatibilidad de las corrientes.

#### COMPATIBILIDAD DE TENSIONES Y CORRIENTES

La comprobación de la compatibilidad de tensiones consiste en verificar que los rangos de las tensiones de salida "caben" en los rangos de las tensiones de entrada. Debemos realizar las siguientes preguntas:

- $\dot{V}_{OLmax}(A) \leq V_{ILmax}(B)?$  y  $\dot{V}_{OLmax}(A) \leq V_{ILmax}(C)?$
- $\dot{V}_{OHmin}(A) \geq V_{IHmin}(B)?$  y  $\dot{V}_{OHmin}(A) \geq V_{IHmin}(C)?$



Si se cumplen las dos condiciones, entonces se cumple la compatibilidad en tensiones.

Por otro lado, la comprobación de la compatibilidad de corrientes consiste en verificar que las intensidades de salida son "mayores" que las que requieren las entradas.

- $\dot{I}_{OLmax}(A) \geq |I_{ILmax}(B)| + |I_{ILmax}(C)|?$
- $\dot{I}_{OHmax}(A) \geq |I_{IHmax}(B)| + |I_{IHmax}(C)|?$

Si se cumplen las dos condiciones, entonces se cumple la compatibilidad en corrientes.

#### COMPATIBILIDAD ENTRE FAMILIAS

Pueden existir problemas de acoplamiento entre componentes de familias distintas, debidos a: *diferentes tensiones de alimentación, diferentes niveles lógicos, corrientes de entrada y salida incompatibles*. Requisitos:

$$V_{OLmax} \leq V_{ILmax} \quad V_{OHmin} \geq V_{IHmin} \quad |I_{OLmax}| \geq n \cdot |I_{ILmax}| \quad |I_{OHmax}| \geq n \cdot |I_{IHmax}|$$

Se considera el caso general de una salida de una determinada familia, conectada a "n" entradas de otra familia.

#### DISEÑO DEL CIRCUITO DE PULL-UP

La tensión de alimentación debe ser SIEMPRE la misma que la de los receptores ( $V_B$ ). De este modo se podrá hacer que los receptores B puedan interpretar correctamente la tensión de sus entradas.

- Nivel alto:** la resistencia de pull-up debe ser inferior a un límite máximo para respetar el margen de tensión a nivel alto en las entradas ( $B_j$ )
  - $I_{RH} = m \cdot I_{OHmax(A)} + n \cdot I_{IHmax(B)}$
  - $V_o = V_B - I_{RH} \cdot R_{pu} \geq V_{OHmin}(B)$
- Nivel bajo:** la resistencia de pull-up debe ser superior a un mínimo para no superar la corriente máxima a nivel bajo en una entrada ( $A_i$ )
  - $I_{RL} = (V_B - V_{OLmax(A)})/R_{pu}$
  - $I_{OL} = I_{RL} + n \cdot |I_{ILmax}(B)| \leq I_{OLmax}(A)$

$$R_{pu} \leq \frac{V_B - V_{OHmin}(B)}{m \cdot I_{OHmax}(A) + n \cdot I_{IHmax}(B)}$$

$$R_{pu} \geq \frac{V_B - V_{OLmax}(A)}{I_{OLmax}(A) - n \cdot |I_{ILmax}(B)|}$$



# SEGUNDO PARCIAL TCO – RESUMEN COMPLETO

## TEMA 4 – FUNDAMENTOS DE LA TECNOLOGÍA CMOS

### INTRODUCCIÓN – LA LEY DE MOORE

Esta dice que la cantidad de transistores por chip se duplica cada aproximadamente 2 años, mientras que la longitud del canal, o el “*feature size*” disminuye el 30% cada aproximadamente 2 o 3 años.

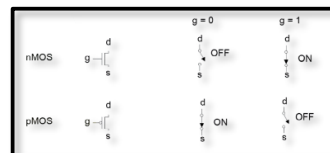
En lo que a la frecuencia del reloj respecta, el crecimiento también ha sido exponencial, pero no tan rápido.

La potencia consumida por los chips también ha sufrido un crecimiento exponencial, desde los primeros chips de micros que no llegaban a 1W, hasta los actuales con más de 100W.

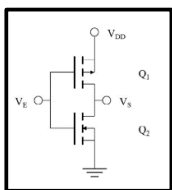
Donde más se observa el problema de la potencia es en su valor relativo respecto a la unidad de área de chip, es decir, la *densidad de potencia*. La disipación de potencia es uno de los problemas más importantes en el diseño y fabricación de los chips VLSI actuales, a medida que aumenta la densidad de integración, reduce fiabilidad en los transistores y circuitos.

### CIRCUITOS COMBINACIONALES

Los transistores MOS se pueden ver como interruptores controlados por tensión. La tensión en la puerta controla la conexión entre los terminales de drenador y fuente.



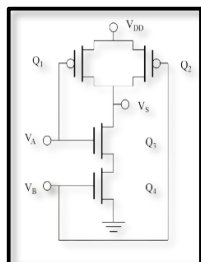
### EL INVERSOR CMOS



El inversor es la puerta básica de la familia CMOS, a partir de la cual se diseñan las demás.

- Pareja de transistores, PMOS y NMOS.
- PMOS conectado a  $V_{DD}$  y NMOS conectado a masa.
- Las puertas de los transistores conectadas entre sí y a la entrada.
- Los drenadores de los transistores conectados entre sí y a la salida.

### LA PUERTA NAND CMOS



1 **0V | 0V:** Los dos transistores NMOS están cortados mientras que los dos PMOS conducen.

- Salida:  $V_{DD}$

2 **0V |  $V_{DD}$ :** La tensión de nivel bajo aplicada a  $V_A$  hace conducir a Q1 (PMOS) y deja cortado a Q3 (NMOS), mientras que la tensión de nivel alto aplicada a  $V_B$  hace conducir a Q4 (NMOS) y deja cortado a Q2 (PMOS).

- Salida:  $V_{DD}$

3  **$V_{DD}$  | 0V:** La tensión de nivel bajo aplicada a  $V_B$  hace conducir a Q2 (PMOS) y deja cortado a Q4 (NMOS), mientras que la tensión de nivel alto aplicada a  $V_A$  hace conducir a Q3 (NMOS) y deja cortado a Q1 (PMOS).

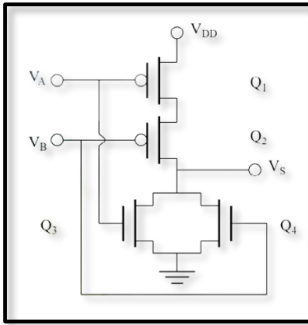
- Salida:  $V_{DD}$

4  **$V_{DD}$  |  $V_{DD}$ :** Los dos transistores PMOS están cortados, mientras que los dos NMOS conducen.

- Salida: 0V



## LA PUERTA NOR CMOS



**1 0V | 0V:** Los dos transistores NMOS están cortados mientras que los dos PMOS conducen.

- Salida:  $V_{DD}$

**2 0V |  $V_{DD}$ :** La tensión de nivel bajo aplicada a  $V_A$  hace conducir a Q1 (PMOS) y deja cortado a Q3 (NMOS), mientras que la tensión de nivel alto aplicada a  $V_B$  hace conducir a Q4 (NMOS) y deja cortado a Q2 (PMOS).

- Salida: 0V

**3  $V_{DD}$  | 0V:** La tensión de nivel bajo aplicada a  $V_B$  hace conducir a Q2 (PMOS) y deja cortado a Q4 (NMOS), mientras que la tensión de nivel alto aplicada a  $V_A$  hace conducir a Q3 (NMOS) y deja cortado a Q1 (PMOS).

- Salida:  $V_{DD}$

**4  $V_{DD}$  |  $V_{DD}$ :** Los dos transistores PMOS están cortados, mientras que los dos NMOS conducen.

- Salida: 0V

Existen otras puertas básicas como, por ejemplo:

$$\text{Buffer} = \text{NOT} + \text{NOT}$$

$$\text{AND} = \text{NAND} + \text{NOT}$$

$$\text{OR} = \text{NOR} + \text{NOT}$$

## DISEÑO DE FUNCIONES GENERALES EN LÓGICA CMOS COMPLEMENTARIA

Podemos obtener la función analizando el bloque NMOS o el bloque PMOS.

### ANALIZANDO EL BLOQUE NMOS

Obtener una función intermedia G a partir de las asociaciones serie/paralelo de los transistores NMOS:

- **Serie:** AND ( $\cdot$ ) de las variables correspondientes.
- **Paralelo:** OR (+) de las variables correspondientes.

Una vez obtenida G,  $F = \bar{G}$ , pues si G = '1' hay una conexión entre F y GND.

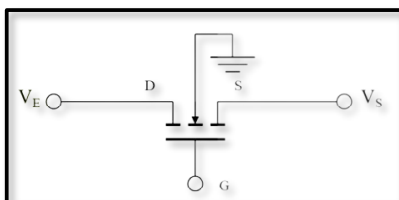
### ANALIZANDO EL BLOQUE PMOS

Obtener una función intermedia G a partir de las asociaciones serie/paralelo de los transistores PMOS:

- **Serie:** AND ( $\cdot$ ) de las variables correspondientes, pero *negadas*, dado que PMOS conduce con '0' en la entrada.
- **Paralelo:** OR (+) de las variables correspondientes, pero *negadas*.

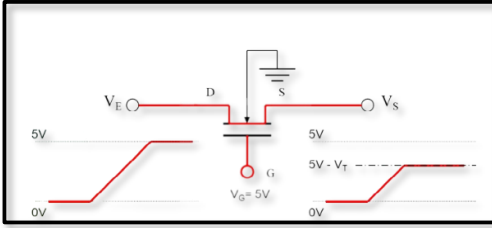
Una vez obtenida G,  $F = G$ , pues si G = '1' hay una conexión entre F y VDD. Estas funciones F obtenidas por cualquiera de los 2 métodos deben ser equivalentes. Si aplicamos el teorema de *De Morgan* a una de ellas, se obtiene la otra.

## PUERTAS DE TRANSMISIÓN



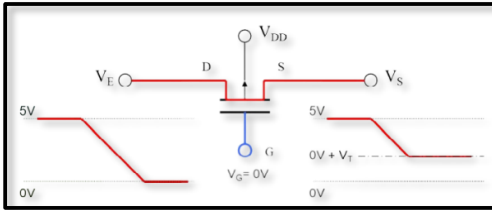
Las puertas de transmisión permiten diseñar circuitos CMOS de manera más compacta. Se aplican a determinados circuitos CMOS, como multiplexores, XOR y biestables.

La idea es tener un interruptor gobernado por  $V_G$ , que conecte la entrada  $V_E$  con la salida  $V_S$ ,



## NMOS

Si la tensión de entrada pasa a  $V_{DD}$  (5V), un nivel alto en  $V_G$  de 5V hace que la puerta de transmisión se cierre. La tensión de entrada se transmite a la salida de la puerta de transmisión pero, cuando alcanza el valor  $V_{DD} - V_T$ , el transistor deja de conducir, pues no se cumple la condición de formación del canal. Por tanto no transmite más allá de  $5V - V_T$ , y se dice que el '1' se degrada en una cantidad =  $V_T$ .



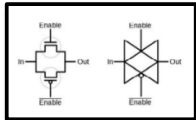
## PMOS

En este caso, la puerta conduce con un '0' en G, y la entrada E se transmite a la salida S. Si  $G = '1'$ , el transistor está en corte y no se transmite la señal.

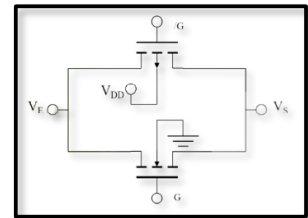
No se transmite el '0' íntegro,  $V_S$  se queda en el valor  $0 + V_T$ . A partir de ahí,  $V_{GS} \geq -V_T$ , y el transistor se corta.

## CMOS

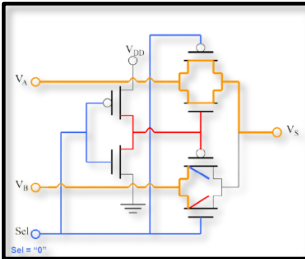
Reúne las características de paso de las dos puertas, no degrada la salida. Obsérvese que el nivel lógico aplicado al terminal de puerta ( $\bar{G}$ ) del transistor PMOS (*arriba*) es el resultado de invertir el aplicado al terminal de puerta (G) del NMOS (*abajo*).



La idea es tener dos puertas de transmisión en paralelo, una NMOS y otra PMOS. Así, solucionamos el problema de la degradación, pues siempre habrá una de las dos puertas de transmisión que transmita la señal.



## MULTIPLEXOR ANALÓGICO



En la figura de la derecha podemos observar el circuito inversor que hace que las entradas a los terminales de control (G) de las puertas de transmisión tengan valores complementarios. Cuando la entrada de selección es un nivel bajo ('0'), se tiene:

- Dos transistores de la puerta de transmisión inferior cortados: '0' a NMOS y '1' a PMOS.
- Dos transistores de la puerta de transmisión superior conduciendo: '1' a NMOS y '0' a PMOS.

Por tanto, el valor de la tensión  $V_A$  se transmite a la salida.

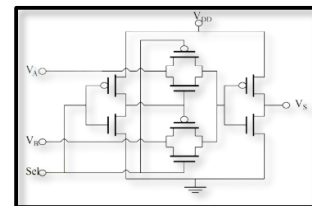
En cambio, cuando la entrada de selección es un nivel alto ('1'), se tiene el caso contrario:

- Dos transistores de la puerta de transmisión inferior conduciendo: '1' a NMOS y '0' a PMOS.
- Dos transistores de la puerta de transmisión superior cortados: '0' a NMOS y '1' a PMOS.

Por ende, el valor de la tensión  $V_B$  se transmite a la salida.

## MULTIPLEXOR DIGITAL

El inversor de salida restaura la señal digital. Sin este, el problema es el siguiente: *si la entrada es una señal con ruido o degradada, la salida recibirá el mismo ruido. Por ello, después de varias etapas como esta, la señal podría resultar demasiado degradada y perder el nivel lógico.*



### LOS BIESTABLES

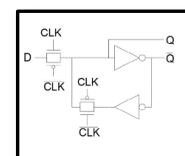
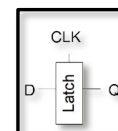
#### LATCH D

Aquí, podemos tener dos opciones distintas:

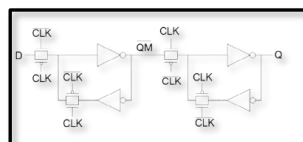
- **CLK = 1:** el Latch es *transparente* → D se transmite a Q como en un buffer.
- **CLK = 0:** el Latch es *opaco* → Q mantiene el valor anterior independientemente de D.

Cuando la señal CLK toma el valor de '1', los dos transistores de la puerta de transmisión CMOS de arriba a la izquierda conducen y, por tanto, se transmite el valor lógico de D. Mientras tanto, la otra puerta de transmisión está abierta, pues los valores aplicados a los terminales de control cortan a los respectivos transistores. El bucle de realimentación está abierto y la salida Q es D.

Sin embargo, cuando la señal CLK toma el valor de '0' se produce el caso contrario, siendo que los dos transistores de arriba a la izquierda se cortan y los de abajo se cierran, por lo que el bucle de realimentación se cierra y mantiene el valor de Q.



#### FLIP-FLOP D



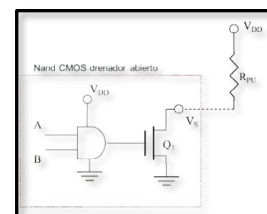
El maestro y el esclavo son 2 latches D como los descritos anteriormente. La salida del maestro se conecta a la entrada del esclavo. La activación por reloj del maestro y del esclavo son complementarias, es decir, cuando uno es '1', el otro es '0'.

### SALIDAS ESPECIALES EN CMOS

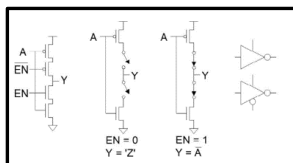
#### PRIMER CASO

En este primer caso, es necesaria una salida en drenador abierto, la cual requiere de una  $R_{PU}$  externa para poner el '1' de salida.

Obsérvese que en el caso de esta estructura NAND, se ha planteado con una estructura AND seguida de un inversor en colector abierto.



#### SEGUNDO CASO



Para este segundo caso, necesitaremos una salida triestado y un inversor triestado. La clave es poder dejar los 2 transistores de salida simultáneamente. Para ello, es necesaria la señal de habilitación (EN) adicional.

### PARÁMETROS CARACTERÍSTICOS – CMOS ESTÁNDAR

#### TENSIÓN DE ALIMENTACIÓN

La tensión de alimentación ( $V_{DD}$ ) es variable, no como en TTL que era fija a 5V. En SSI y MSI se halla típicamente entre 3V y 15V, mientras que en los chips VLSI se encuentra entre los siguientes valores: 3.3V, 2.5V, 1.8V, 1.5V, 1.2V, 1V...



## CONSUMO

Pese a que el consumo **estático** es nominalmente 0, en realidad hay corrientes de fuga muy pequeñas, del orden de nA o pA. Las principales causas del consumo estático son: *corrientes de fuga en uniones PN polarizadas en inversa, corrientes de fuga en puerta aislada por "efecto túnel" o "Sub-threshold currents", cuando  $V_{GS} < V_T$ , siendo  $V_T$  muy pequeño.*

$$P_{leakage} = V_{DD} \cdot I_{leakage}$$

Por otra parte, en el régimen **dinámico**, el cual es el más importante en CMOS, hallamos las siguientes fórmulas:

### Energía Suministrada por la Fuente

$$E_{fuente} = C_L \cdot V_{DD}^2$$

### Potencia Consumida

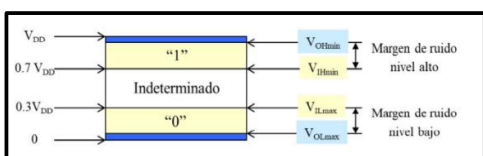
$$P_{dynamic} = (V_{DD})^2 \cdot C_L \cdot f$$

La *capacidad parásita* ( $C_L$ ) es debida a las capacidades de: *drenador en el inversor, cableado y puerta de los circuitos conectados a la salida.* Esta es del orden picofaradios (pF) o femtofaradios (fF). Por otra parte, el consumo de  $V_{DD}$  corresponde a la carga de  $C_L$ , mientras que la descarga no exige ningún consumo de  $V_{DD}$ .

Estos dos consumos se relacionan para obtener el consumo total, el cual se calcula de la siguiente manera:

$$P_{total} = P_{leakage} + P_{dynamic}$$

## NIVELES LÓGICOS – INMUNIDAD AL RUIDO



La curva de transferencia y la inmunidad al ruido es bastante mejor que en TTL.

En CMOS, la inmunidad al ruido es directamente proporcional a  $V_{DD}$ .

## NIVELES DE CORRIENTE – FAN-OUT

Con el fin de no incrementar los tiempos de retardo y el consumo dinámico, el fabricante suele recomendar un Fan-out no superior a 50 entradas.

## RETARDOS DE PROPAGACIÓN

Según la ecuación, el retardo es directamente proporcional a la capacidad de carga e inversamente proporcional a la tensión de alimentación.

$$t_p = \frac{1}{2} \cdot (t_{p_{hl}} + t_{p_{lh}}) \approx \frac{C_L}{4 \cdot (V_{DD} - V_T)} \cdot \left( \frac{1}{K_p} + \frac{1}{K_n} \right)$$

El valor del retardo depende de  $V_{DD}$ ,  $C_L$ ,  $V_T$  y de la  $K$  de los transistores.

- A mayor  $V_{DD}$  menor retardo.
- A mayor  $C_L$  mayor retardo.
- A menor  $V_T$ , menor retardo
- A menor  $K$ , menor retardo

A su vez, podemos aumentar la velocidad de los circuitos. Esto se logra *reduciendo  $C_L$ , incrementando  $K_p$  y  $K_n$  o disminuyendo  $V_T$ .*

## PROCESO DE FABRICACIÓN DE LOS CHIPS

Consiste de tres etapas:

1. **Diseño y verificación:** se diseña y verifica el *layout* mediante herramientas CAD.
2. **Fabricación de las máscaras:** que son patrones poligonales que representan las diferentes capas de material para la fabricación de los circuitos. Estas se fabrican mediante métodos ópticos.
3. **Proceso fotolitográfico:** las máscaras son proyectadas sobre la oblea. La deposición de las máscaras sobre la oblea se realiza en paralelo, fabricando varios chips a la vez.

## PROCESO FABRICACIÓN INVERSOR CMOS

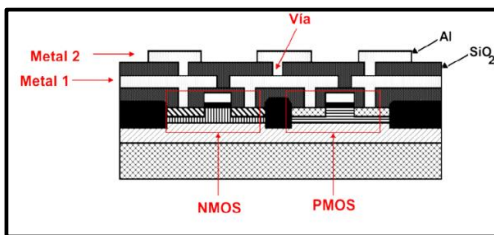
### DISEÑO Y VERIFICACIÓN

En esta ilustración, podemos encontrar las diferentes capas que presenta el inversor CMOS, de arriba abajo, siendo A la entrada e Y la salida del inversor:

- **SiO<sub>2</sub>:** aislante de puerta
- **N<sup>+</sup> diffusion:** zonas semiconductor N altamente dopado de fuente y drenador del transistor NMOS.
- **P<sup>+</sup> diffusion:** zonas semiconductor P altamente dopado de fuente y drenador del transistor PMOS.
- **Polysilicon:** puerta de los transistores.
- **Metal 1:** para las interconexiones VDD y GND.

Para evitar corrientes parásitas debidas a las uniones PN parásitas, hace falta añadir los contactos de sustrato y pozo indicados a la derecha.

Normalmente hay varias capas de metal: metal1, metal2... dispuestas en tres dimensiones para facilitar las interconexiones entre los circuitos integrados en el chip. Las más internas corresponden a conexiones locales dentro de un mismo circuito.



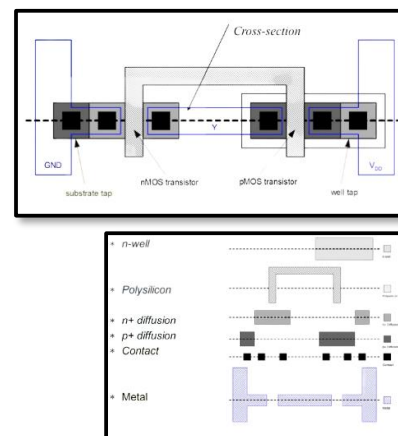
- Las capas de metal más externas suelen ser las de alimentación y las de reloj.
- Las capas de metal diferentes se pueden conectar entre sí mediante contactos llamados Vías.
- Normalmente las capas de metal más externas son más anchas, pues por ellas pasa más corriente.
- Los metales más usados son Aluminio (Al) y Cobre (Cu).

### MÁSCARAS

Las máscaras son polígonos-patrón que representan la forma de las diferentes capas a depositar sobre el sustrato. Son proyectadas sobre la oblea para establecer la forma, tamaño y ubicación de las diferentes capas de material.

Máscaras en el orden en que se depositan:

1. Pozo-n
2. Polisilicio, es decir, la puerta del transistor.
3. Difusión n<sup>+</sup>
4. Difusión p<sup>+</sup>
5. Contactos metálicos
6. Capas de metal de VDD y GND

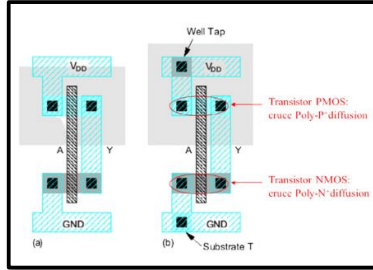




## FOTOLITOGRAFÍA

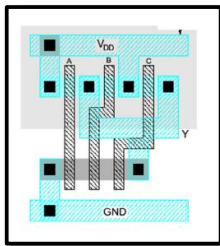
Una vez se han definido las máscaras, hay que proyectarlas sobre la oblea. Las máscaras se fabrican con los siguientes materiales:

- **Glass plate:** cuarzo transparente, deja pasar la luz.
- **Chromium:** opaco, no deja pasar la luz



El *layout* es el esquema previo a la generación de las máscaras. Se realiza con herramientas CAD, o con un editor de *layout*.

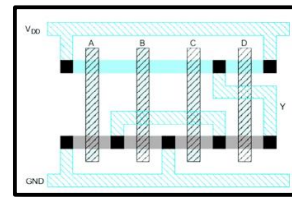
El *layout* de la izquierda muestra el esquema del inversor con el tamaño exacto de las diferentes máscaras. A la figura de la derecha se le han añadido los contactos de sustrato y pozo.



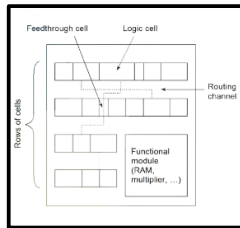
El *layout* que podemos encontrar a la izquierda corresponde a una puerta NAND de tres entradas.

Por otra parte, el *layout* situado a la derecha cumple la siguiente función lógica:

$$Y = \overline{(A + B + C)} \cdot D$$



## CELDAS ESTÁNDAR



Las celdas suelen ser circuitos de complejidad limitada. Sus *layout* son almacenados en librerías y reutilizados en el diseño de circuitos más complejos.

Las celdas se disponen por filas. Cada fila de celdas comparte una línea horizontal de GND (inferior) y una línea horizontal de VDD (superior). De esta forma, se compacta más el diseño.



## SEGUNDO PARCIAL TCO – RESUMEN COMPLETO

### TEMA 5 – TECNOLOGÍA DE LAS MEMORIAS SEMICONDUCTORAS

#### ORGANIZACIÓN BÁSICA

##### CAPACIDAD DE ALMACENAMIENTO

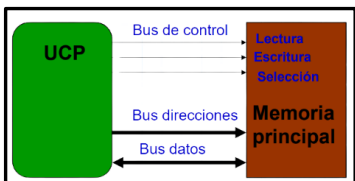
La capacidad se expresa en bytes o múltiplos de bytes. Desde Kilobytes (1024 bytes) hasta Petabytes ( $2^{50}$  bytes). También, debemos tener en cuenta la siguiente igualdad: 1 Byte (B) = 8 bits (b).

Ejemplos de capacidad de almacenamiento:

- **Memoria de 128K posiciones de 16 bits cada una:**  $128K \cdot 16 \text{ bits} = 128K \cdot 2^4 \text{ bits} = 128K \cdot 2^1 \text{ Bytes} = 256K$
- **Memoria de 8MB en palabras de 32 bits:**  $8MB = 2^3 \cdot 2^{20} \cdot 2^3 \text{ bits} = 2^1 \cdot 2^{20} \cdot 2^5 \text{ bits} = 2M \cdot 32 \text{ bits}$

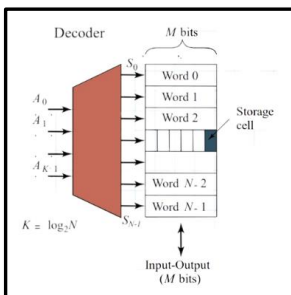
Por ello, la capacidad = Número de palabras · Tamaño de la palabra. Esta se puede dar en múltiplos de bit (b) o de Byte (B).

##### CONEXIÓN CPU – MEMORIA



- **Control:** la CPU selecciona la memoria y determina la operación (lectura o escritura).
- **Direcciones:** la CPU determina la dirección.
- **Datos:** según la operación, los datos van:
  - **Lectura:** de la memoria a la CPU.
  - **Escritura:** de la CPU a la memoria.

##### MEMORIAS SEMICONDUCTORAS



La figura que se encuentra a la izquierda representa la estructura básica de las memorias semiconductoras. En esta, se realiza una selección de las palabras por filas, mediante un decodificador.

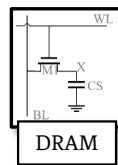
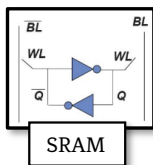
Dependiendo del valor del código de direcciones ( $A_0:A_{k-1}$ ), el decodificador selecciona una de las N palabras. Cada palabra contiene M bits, que son leídos o escritos simultáneamente.

En función de su tiempo de acceso, tenemos:  $SRAM < DRAM < Flash < Disco Magnético$ , siendo la SRAM la que cuenta con un tiempo de acceso menor. Si analizamos en función del coste de acceso, tenemos:  $SRAM > DRAM > Flash > Disco Magnético$ .

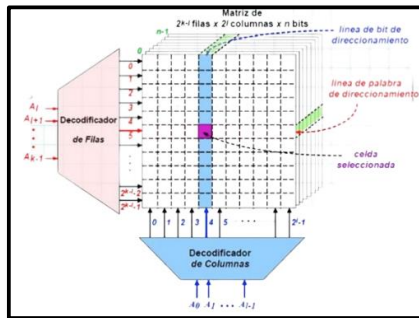
#### MEMORIAS VOLÁTILES DE ACCESO ALEATORIO: RAM

Las memorias RAM (*Random Access Memory*) pierden los datos si se desconecta la alimentación, es por eso que se dice que son memorias *volátiles*. También, realizan operaciones de lectura y escritura. Dependiendo de la celda básica, pueden ser SRAM (*Static RAM*) o DRAM (*Dynamic RAM*).

- **SRAM:** están basadas en biestables, y almacenan datos mientras haya alimentación. El tamaño de la celda es grande (de 6 transistores por celda) y se caracterizan por ser rápidas (caché). La celda es un biestable Latch.
- **DRAM:** se basan en la carga y descarga de capacidad estructural. Estas requieren de un refresco periódico, tienen un tamaño de celda menor (de 1 a 3 transistores por celda) y son más lentas que las SRAM (memoria principal).



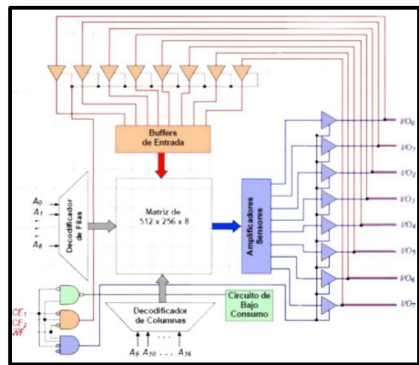
## SRAM: ESTRUCTURA EN ARRAY CON 2 DECODIFICADORES



Las líneas de dirección se dividen 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  $2^{K-1}$  filas  $\cdot$   $2^l$  columnas  $\cdot$   $n$  bits.

La figura de la izquierda representa un esquema para poder entenderlo, pero también tenemos que tener en cuenta el esquema real:

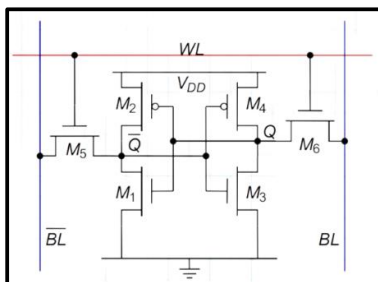


Ahora, podemos encontrar a la izquierda el esquema de la memoria SRAM, la cual utiliza buffers triestado para la gestión de la entrada o las salida de la información.

Las líneas de control gestionan la entrada/salida:

- **CE:** "chip-enable", selección de chip.
- **OE:** "output enable", habilitación de la salida (lectura).
- **WE:** "write enable", habilitación de escritura.

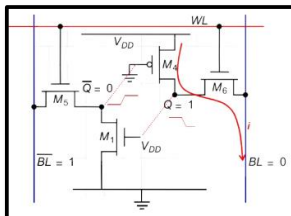
## SRAM: CELDA DE MEMORIA



Como sabemos de antes, la celda de la memoria SRAM es un biestable Latch. En la figura de la izquierda, tenemos:

- **M<sub>5</sub> y M<sub>6</sub>:** dos puertas de transmisión NMOS para acceder al Latch interno.
- **M<sub>1</sub> ↔ M<sub>2</sub> y M<sub>3</sub> ↔ M<sub>4</sub>:** dos inversores CMOS para formar el Latch, los cuales almacenan el estado de la celda.
- **BL y BL-bar:** líneas de bit complementarias, donde se sitúan los datos para escribir, o se leen los datos.
- **WL:** es la línea de selección de palabra. Cuando WL = '1', los transistores M<sub>5</sub> y M<sub>6</sub> se cierran y se accede a la celda, mientras que cuando WL = '0', dichos transistores se abren y la celda está aislada, de forma que se mantiene el estado.

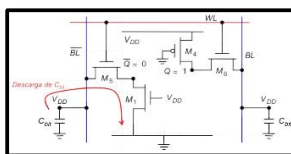
### PROCESO DE ESCRITURA



Supongamos que Q = '1' y Q-bar = '0', y queremos escribir un '0':

1. Se sitúan los datos complementarios en BL y BL-bar.
2. Se selecciona la celda: WL = '1'.
3. El biestable cambia de estado:
  - a. Es suficiente con que Q quede por debajo de VDD/2.
  - b. La realimentación hace que Q-bar pase a '1' y Q a '0'.

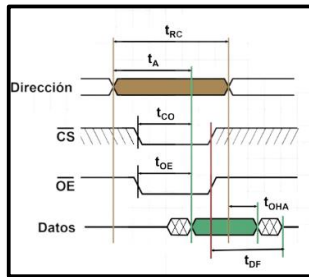
### PROCESO DE LECTURA



Supongamos que Q = '1' y Q-bar = '0'.

1. BL y BL-bar se precargan a VDD.
2. Se selecciona la celda WL = '1'.
3. BL-bar se descarga por M<sub>5</sub> y M<sub>1</sub> y pasa a '0'.

## SRAM – CRONOGRAMA



Antes de poder leer el cronograma, debemos conocer 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 cero (Dirección o Datos).

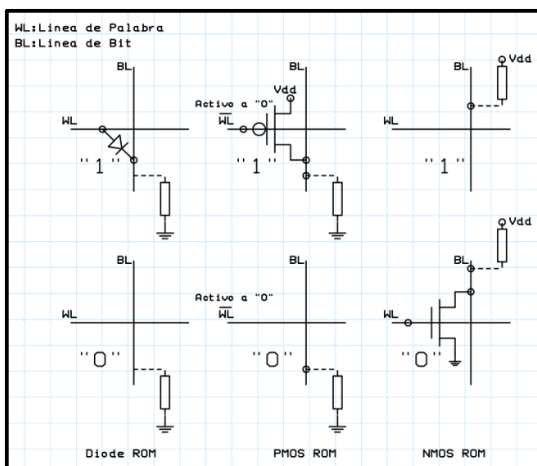
Dentro de estas, tenemos líneas que pueden cambiar en cualquier momento (CS o Datos), niveles estables en buses, los cuales se hallan indicado en color opaco, y niveles cambiando en buses, indicados por las Aspas visibles en Dirección o Datos.

Con esto, ya podemos entender el cronograma de lectura de una SRAM. CPU y Memoria se comunican de forma *asíncrona*, que es cuando la CPU inicia la petición de lectura dejando la dirección de donde quiere leer en el bus de direcciones, para luego activar las líneas CS y OE en cualquier orden. Cuando 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 esté a 0 en ese momento indicará el tipo de operación. Existen diversos tipo de tiempos que debemos conocer:

- **Tiempo de ciclo ( $t_{RC}$ ):** tiempo mínimo que ha de pasar entre una operación de lectura y la siguiente.
  - **Tiempo de ciclo de escritura:**  $t_{WC}$
  - **Tiempo de ancho del pulso de escritura:**  $t_{WP}$
  - **Tiempo de establecimiento de los datos:**  $t_{DW}$
  - **Tiempo de mantenimiento de los datos:**  $t_{DH}$
- **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 ( $t_{OHA}$ ):** tiempo que se mantendrán los datos estables en el bus después de que cambien las direcciones para iniciar una nueva operación.
- **Tiempo de paso a alta impedancia ( $t_{DF}$ )**

## MEMORIAS DE SOLO LECTURA – ROM (READ-ONLY MEMORY)

Las memorias de solo lectura almacenan la información como parte de la topología del circuito. Puesto que esta topología está prefijada en hardware en tiempo de fabricación, los datos no pueden modificarse, sino que solo pueden ser leídos. A su vez, la desconexión de la tensión de alimentación no provoca la pérdida de los datos almacenados.

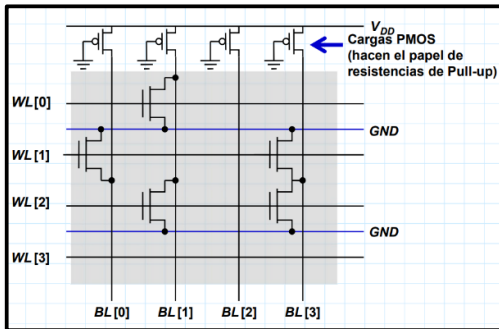


En la figura de la izquierda se representan tipos de celdas de memoria ROM. En la fila de arriba, la información almacenada en las celdas es un '1'.

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). La presencia de este diodo hace que, cuando WL = '1', el diodo conduzca y en la BL 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. Si este diodo no estuviese, dicha ausencia pone un nivel bajo en BL, almacenando un '0'.

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 pueden diseñar ROM de más capacidad. Esta alternativa es más compleja y ocupa un mayor espacio.

## ROM NMOS CON ESTRUCTURA NOR

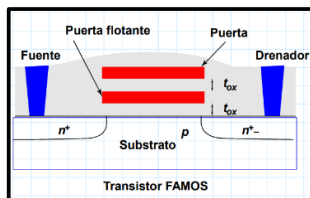


Cuando  $WL = '1'$ , los transistores de la palabra conducen y sus correspondientes BL se conectan a GND. En caso de que en la celda no exista el transistor, la BL permanece a '1' por defecto, gracias a las cargas PMOS conectadas a VDD.

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

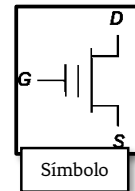
La estructura NOR se refiere al conjunto de transistores conectados en paralelo a una misma línea de bit, junto con la carga pull-up.

## MEMORIAS NO VOLÁTILES REPROGRAMABLES



Para poder reprogramar varias veces sin perder la no-volatilidad, surgen nuevas tecnologías, véase la EPROM, EEPROM o FLASH. Estas memorias pueden leerse y 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 (Floating-gate Avalanche-injection MOS).



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. Es un '0' almacenado. Por otra parte, si eliminamos los electrones de la puerta flotante, el transistor funciona correctamente. Es un '1' almacenado.

La memoria es no-volátil porque la carga se almacena en la puerta flotante, rodeada de aislante. Aunque se elimine la alimentación, no pierde la carga. Dependiendo de la forma en que se efectúa el borrado, hay diferentes tipos de memoria:

- **EPROM (Erasable PROM):** se borra haciendo incidir luz ultravioleta sobre las celdas de memoria a través de una ventana de cuarzo 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. El borrado es global, y la programación requiere una tensión elevada, aproximada a 12V. La programación es selectiva por palabras, mientras que el borrado y la programación se realizan fuera del funcionamiento normal.
- **EEPROM y Flash:** estas son versiones mejoradas de la anterior, las cuales permiten programar eléctricamente durante el funcionamiento normal, sin necesidad de un dispositivo externo.

## MEMORIAS FLASH

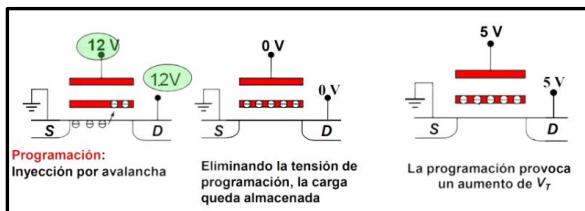
Estas son sucesoras de las EEPROM, y consiguen más densidad de integración, dado que asignan un transistor por celda, en vez de dos. Permiten el borrado por bloques, el cual es mucho más rápido (de ahí el nombre *flash*).

Existe un problema de fiabilidad, donde 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 y borrados (100.000 – 1.000.000), dependiendo de la precisión del proceso de fabricación y del voltaje necesario para su borrado. Sin embargo, presentan más ventajas:

- Al ser memorias semiconductoras, no tienen partes móviles y en consecuencia no tienen problemas de averías.
- Al ser no-volátiles, no requieren alimentación para mantener los datos.
- Son más rápidas que los discos magnéticos ( $\mu s$  vs  $ms$ ).



### CELDA FLASH



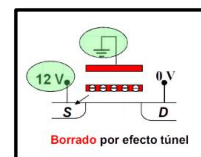
La celda básica está compuesta por un transistor FAMOS, el cual tiene una puerta flotante entre la puerta de control y el sustrato.

Ahora, explicaremos el proceso de programación y de borrado:

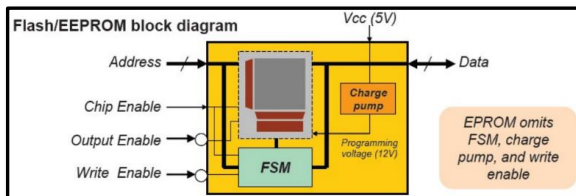
- **Programación:** La idea es almacenar electrones en la puerta flotante, aplicando una tensión positiva elevada (típicamente 12V) en la puerta de control. Esto, junto a la fina capa de aislante, genera un campo eléctrico muy intenso que consigue que los electrones atraviesen el aislante. Al recuperar las tensiones normales, la carga almacenada no se pierde, pues la puerta flotante está rodeada de aislante. La programación pone la celda a '0'.
- **Borrado:** Si se quiere descargar la puerta flotante, se aplica la tensión con la polaridad inversa, lo que provoca el vaciado de la puerta flotante. El borrado pone la celda a '1'.

Se producen dos mecanismos físicos muy parecidos:

1. **Escritura mediante hot-electron:** los electrones pasan desde el sustrato a la puerta flotante.
2. **Borrado mediante efecto túnel Fowler-Nordheim:** los electrones abandonan la puerta flotante hacia la zona de fuente del transistor.

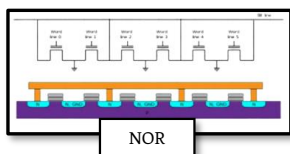


### ESTRUCTURA INTERNA CHIPS DE MEMORIA FLASH

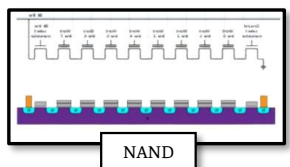


- Lectura de (E)EPROM o Flash igual a SRAM.
- $V_{pp}$ : entrada voltaje programación (12V)
- EPROM:  $V_{pp}$  proporcionada por máquina externa
- Flash/EEPROM generan 12V internamente.
- EPROM no tiene Write-Enable
- Flash/EEPROM con secuencia de escritura controlada por un FSM.

### NOR FLASH VS NAND FLASH



**NOR Flash:** las celdas se disponen con estructura NOR. Tienen menor capacidad y se leen por palabras. Ej: BIOS



**NAND Flash:** las celdas se disponen con estructura NAND. Tienen mayor capacidad, se usan para almacenar datos no volátiles y se leen por páginas. Ejemplos: *pendrive*, *datos en móviles*, *SSDs*, etc.



participa  
aquí

