

Práctica 7: Circuitos lógicos CMOS

Indice

1. Objetivos
2. Material necesario
3. Desarrollo práctico
 - 3.1 Puerta NAND de 2 entradas
 - 3.2 Función lógica general
 - 3.3 Multiplexor 2x1 (apartado opcional)

1. Objetivos

- ☐ Diseñar y simular el funcionamiento de circuitos CMOS, empleando lógica complementaria.

2. Material necesario

- PC y programa de simulación PSpice para Windows. Existe una versión de estudiante en PoliformaT. También está accesible a través de PoliLabs (<https://polilabs.upv.es>).

3. Desarrollo práctico

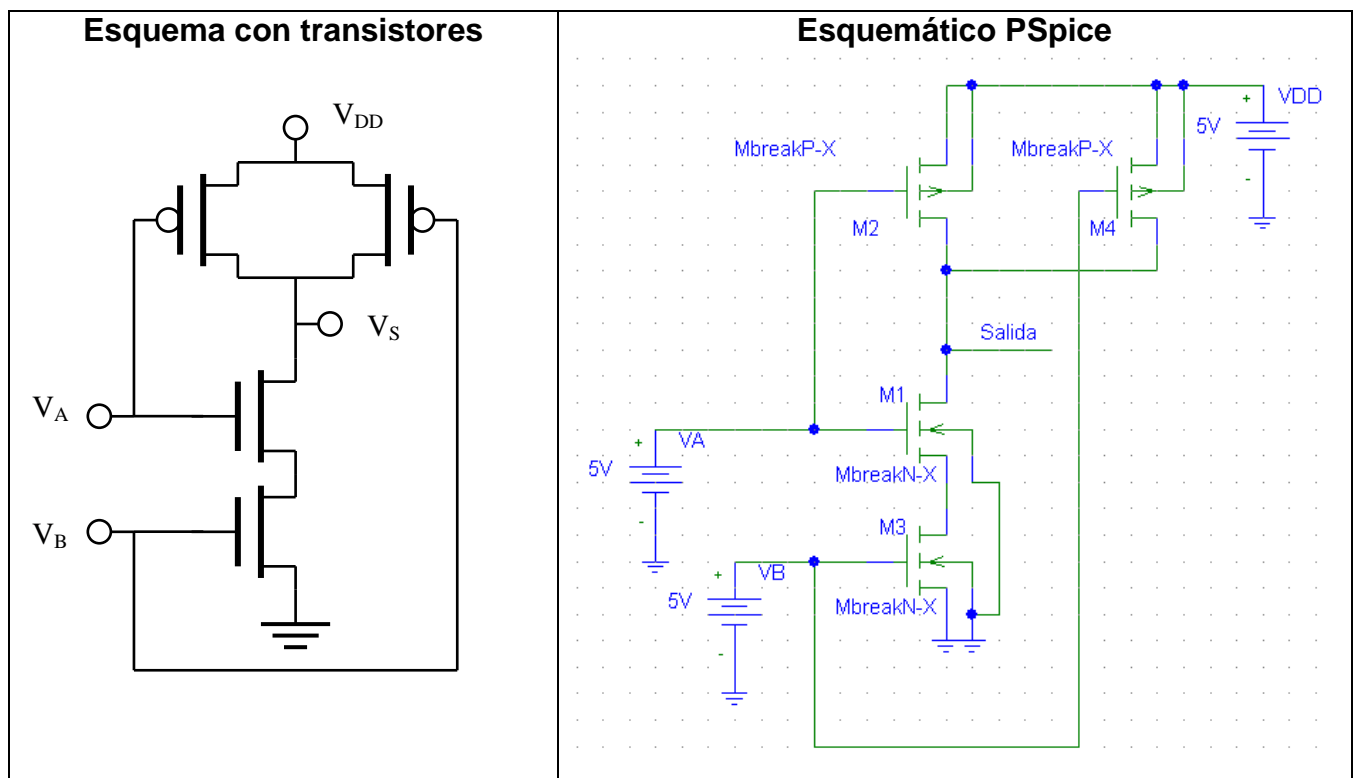
Inicie PSpice haciendo doble clic sobre el icono PSPICE del escritorio y guarde los archivos en W:\TCO\Prac7.

MUY IMPORTANTE: Cuando guarde un circuito, dé al archivo un nombre corto, y preferiblemente con caracteres alfanuméricos (y guiones); evite acentos, espacios y caracteres especiales. Recuerde también que los nombres por defecto utilizados por PSpice (Schematic1, Schematic2, etc.) pueden dar problemas.

3.1 Puerta NAND de 2 entradas

- Edite en el *Schematics* el circuito correspondiente a una puerta NAND CMOS de 2 entradas, tal como muestra la figura adjunta. **Aliméntela a VDD= +5V.**
 - * Para importar los componentes, siga el camino *Draw*→*Get New Part*→*Part Name* en el menú *Edit*.
 - MbreakP (transistor PMOS)
 - MbreakN (transistor NMOS)
 - EGND (masa)
 - VDC (tensión continua, para las 2 entradas y la tensión de alimentación)

Para orientarlos correctamente en el esquema, tenga en cuenta que se pueden rotar (CTRL-R) y voltear (CTRL-F).



Observe que los transistores NMOS están en serie, y los PMOS en paralelo.

Edite el modelo de los transistores MOSFET:

- Con *Edit→Model→Edit Instance Model (Text)*), ajuste Kp (transconductancia o ganancia) y Vto (tensión umbral V_T)

model MbreakP PMOS Vto = -1 Kp = 15u *\$	model MbreakN NMOS Vto = 1 Kp = 15u *\$
--	---

- Especifique el ancho (W) y largo (L) del canal pulsando dos veces sobre el símbolo del transistor y cambiando los atributos: W = L = 1u (1u = 1 micro = 10^{-6})

Observe que se han asignado los mismos valores de Kp y (W/L) a los 2 transistores, para simplificar.

Importante: cuando hay que replicar un mismo transistor varias veces, conviene **editar una sola vez el modelo del transistor**, especificando sus parámetros. Después de haber editado el modelo, se selecciona (con un clic sobre él), y **se van dejando copias** (Edit→Copy) sobre el esquema. Al hacerlo así, nos aseguramos de que todos los modelos son idénticos.

Otro detalle a recordar: los **substratos** de los transistores **NMOS** deben conectarse todos a **masa**, y los **substratos** de los transistores **PMOS** deben conectarse todos a **V_{DD}**.

Simule el funcionamiento de la puerta para cada una de las valoraciones de su **tabla de verdad**:

A	B	Vs (voltaje)	Vs (valor lógico)	Consumo estático (I)
0V	0V			
0V	5V			
5V	0V			
5V	5V			

Para ello:

- * Habilite el tipo de análisis como *Bias Point Analysis*. (*Analysis → Setup → Bias point detail*).
- * Varíe el valor lógico de las entradas ("0" = 0V, "1" = 5V).
- * Ejecute la simulación (tecla F11 ó *análisis→Simulate*).
- * Verifique el voltaje en la salida, mediante el **botón V** de la pantalla del *Schematics*.



- Compruebe que los **valores de tensión de salida son muy extremos**, aproximadamente V_{DD} para el “1” lógico o GND para el “0” lógico.
- * Como hay 4 posibles combinaciones para 2 entradas, deberá efectuar 4 simulaciones.
 - Compruebe (usando el **botón I**) que en todas ellas **el consumo estático es despreciable**. El consumo estático es la corriente que sale de la fuente hacia el circuito.

Nota: las abreviaturas que se utilizan son:

u = micro = 10^{-6}

n = nano = 10^{-9}

p = pico = 10^{-12}

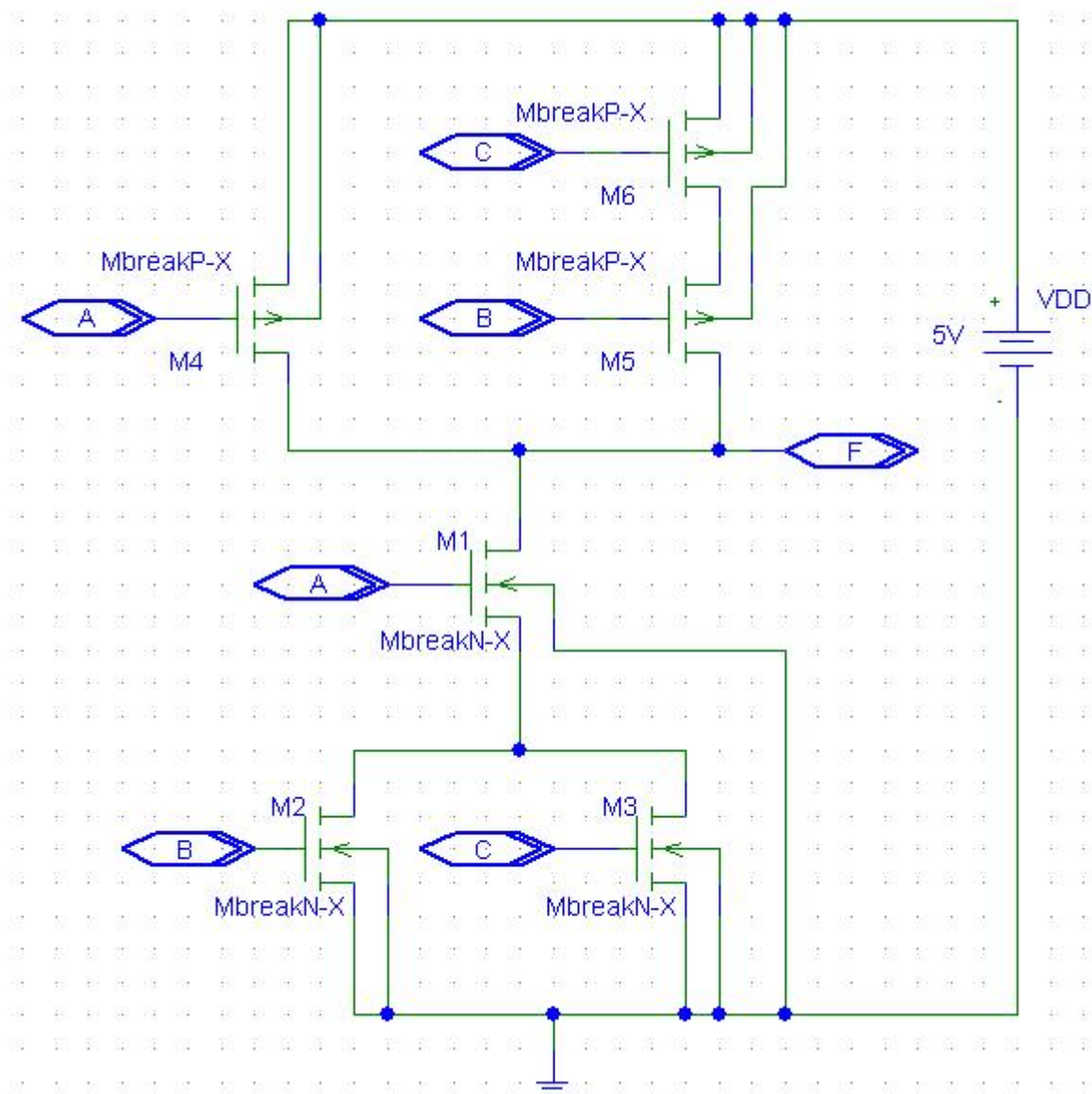
3.2 Función lógica general

- Edite en el **Schematics** el circuito correspondiente a la función lógica:

$$F = \overline{A(B + C)}$$

De la expresión lógica se deduce que la función del bloque NMOS es: $G = A.(B + C)$

Asociando el operador AND a una agrupación serie de transistores NMOS, y el operador OR a una agrupación paralela de transistores NMOS, obtenemos el circuito del bloque NMOS, que situamos entre la salida F y la masa, tal como muestra la figura adjunta. Por su parte el circuito del bloque PMOS se obtiene con una estructura dual, cambiando las agrupaciones serie por paralelo, y viceversa. El bloque PMOS se sitúa entre V_{DD} y la salida, tal como muestra la figura.



- **Aspectos a tener en cuenta en la edición:**
 - * Utilice los mismos parámetros para los transistores que en el apartado 3.1.
 - Especifique un transistor NMOS (con sus parámetros) y replíquelo después las veces que haga falta.
 - Haga lo mismo con el transistor PMOS.
 - * **Alimente el circuito a +5V.** Utilice el elemento VDC para V_{DD} .
 - * Recuerde conectar los **substratos** de los transistores **NMOS** a **masa** y los **substratos** de los transistores **PMOS** a V_{DD} .
 - * Para conseguir una simulación más ágil, se van a usar **entradas digitales**, distintas de las de tipo VDC anteriores. Vaya a *Draw* → *Get New Part* → *Part Name* y seleccione el componente **IF_IN (entradas digitales)**. Sitúelo en las 3 entradas. Pulsando 2 veces sobre el símbolo, asócielo su nombre (A, B, C).
 - * Del mismo modo, seleccione el componente **IF_OUT (salida digital)**. Sitúelo en la salida y asócielo el nombre F.
- **Cree un fichero de estímulos**, para definir las combinaciones lógicas de las señales de entrada:
 - * Seleccione *Analysis* → *Edit Stimuli*. Aparecerá la ventana del editor de estímulos (*Stimulus Editor*).
 - * Especifique las señales de entrada (A, B, C) como ondas cuadradas digitales de distinta frecuencia. Para cada señal de entrada:
 - *Stimulus* → *New*
 - *Name*
 - *Digital* → *Clock*
 - Especifique la frecuencia de la señal. Para que aparezcan las 8 combinaciones posibles de A, B, y C, duplique las frecuencias. Por ejemplo:
 - **C: 1Meg** (para especificar 1MHz, que sería equivalente a un Período $T = 1\mu$ (microsegundo)). **Duty cycle** = 0.5 indica que la onda es simétrica, es decir, la duración del estado alto es igual a la duración del estado bajo. Déjelo así.
 - **B: 2Meg** (2MHz) → Período $T = 0.5\mu$
 - **A: 4Meg** (4MHz) → Período $T = 0.25\mu$

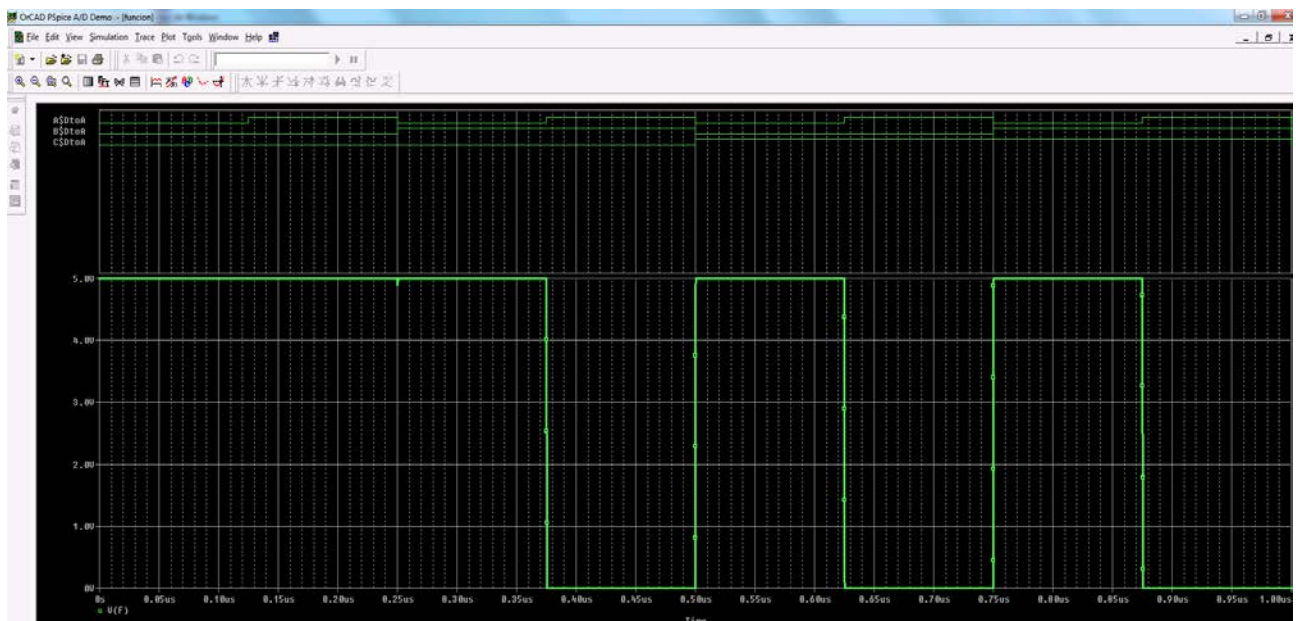
Atención: fíjese que las unidades Mega (10^6), en Pspice se indican como Meg. La M se reserva para mili (10^{-3}).

 - * Guarde el fichero de estímulos (.sti). **Importante:** guarde por defecto (SAVE), sin modificar ni el nombre ni el directorio destino.
- **Efectúe un análisis de tipo *Transient***, para ver la respuesta (F) en el tiempo:
 - * *Active Analysis* → *Setup* → *Transient* en la ventana de *Schematics*.
 - * Especifique la duración temporal del análisis

- Como la frecuencia menor es 1MHz, su período es 1 μ s (1 micro-segundo). Indique en *Final Time* \rightarrow 1 μ
- * Ejecute la simulación (tecla F11 o *análisis* \rightarrow *Simulate*).
- * Una vez terminada, se abrirá automáticamente la ventana de *Probe*, que permite representar señales gráficamente. Para efectuar la selección de la señal, active la opción *Trace* \rightarrow *Add Trace*. Escoja las 3 señales digitales de entrada (V(A), V(B), V(C)) y la señal de salida, V(F).

OJO: para ver las entradas se debe deshabilitar la pestaña de voltajes analógicos. De esta forma se muestran solo las señales digitales. Para ver la salida, se selecciona la pestaña de voltajes analógicos. De esta forma se ve con mayor detalle la señal de salida.

- **Verifique el correcto funcionamiento del circuito**, a partir del cronograma. Como puede observar, en una única simulación aparecen todas las combinaciones lógicas posibles. Si hubiera definido las entradas como VDC (como en el caso de la puerta NAND del apartado 3.1), hubiera necesitado tantas simulaciones *Bias point detail* como combinaciones de entrada (8 simulaciones). Por ello, en caso de tener un número relativamente grande de entradas lógicas, es aconsejable definir un fichero de estímulos y realizar un análisis de tipo *Transient*.

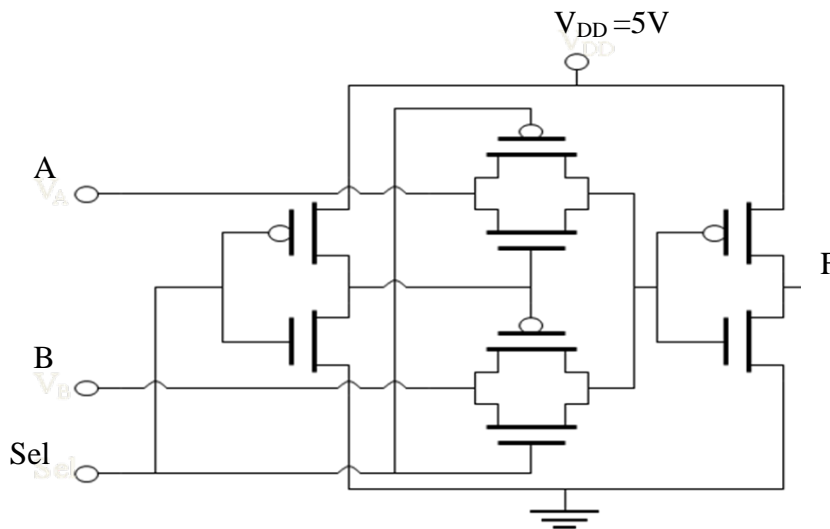


- **Verifique que el consumo estático es nulo para todas las combinaciones de entrada.** Para ello visualice la corriente que suministra la fuente, $I(V_{DD})$. Será necesario crear una gráfica nueva con *Add_Plot*.

Pueden apreciarse picos de corriente en las transiciones de las entradas, debidos al consumo dinámico de cortocircuito.

3.3 APARTADO OPCIONAL: Multiplexor 2x1 con salida inversora

- Edite el esquema de la figura adjunta, correspondiente a un Mux 2x1 con salida inversora. Recuerde conectar los sustratos de los transistores NMOS a GND y los sustratos de los transistores PMOS a V_{DD} .



- Utilice el componente VDC para las entradas de los canales A y B. Fije su valor a: A= "1", B= "0".
- Utilice el componente IF_IN (entrada digital) para la entrada de selección Sel.
- Cree un fichero de estímulos con la señal Sel, definiendo una onda cuadrada de frecuencia 1MHz.
- Simule (Transient) y observe el valor de F en función de Sel. Si Sel="1" debe seleccionar el canal B. Si Sel="0" debe seleccionar el canal A. Pero recuerde que el inversor de la salida invertirá el valor del canal seleccionado.

O sea, el multiplexor actúa así:

$$\text{Sel} = "1" \rightarrow F = \overline{B}$$

$$\text{Sel} = "0" \rightarrow F = \overline{A}$$