Lab 7: CMOS logic circuits

Contents

1. Objectives

- ❑ Design and simulate CMOS circuits, using complementary logic.

2. Required Materials

- Simulation Environment:
  - PSpice program for Windows, from Orcad. There is a student version in PoliformaT.

3. Practical development

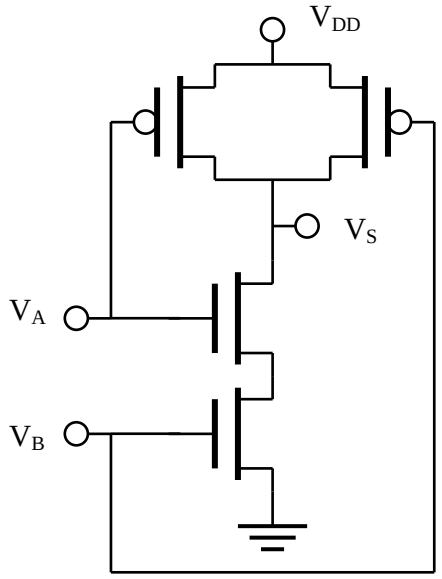
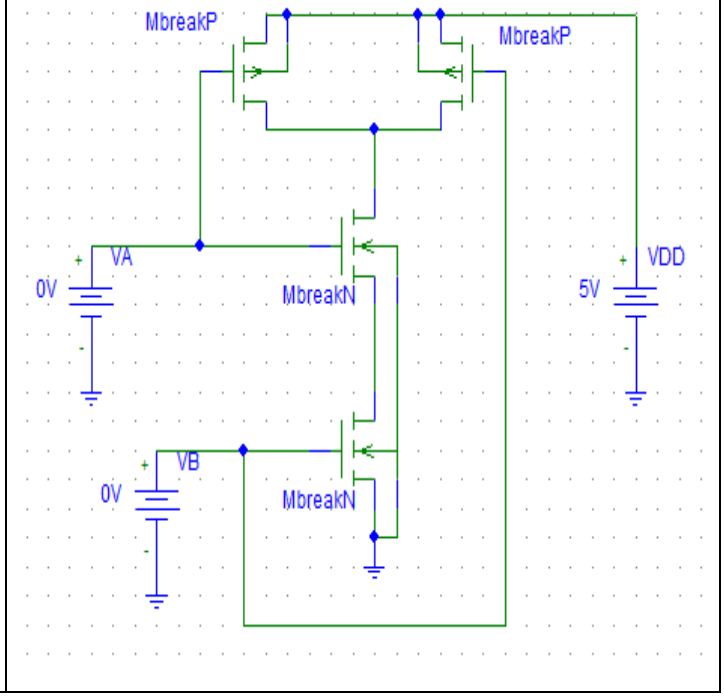
Log in with your user of the ALUMNO domain and start PSpice by double clicking on the PSPICE icon on the desktop. Save the created files in the folder W:\TCO\ Prac7.

VERY IMPORTANT: when you save the circuit, put a different file name to schematic1, schematic2, etc.

3.1 2 input NAND gate

- Edit the circuit using *Schematics* for a CMOS NAND gate with 2 inputs, as shown in the figure below. **Use a power supply of VDD=+5 V**.
    - To import the components, follow the commands *Draw→ Get New Part→Part Name* in the Edit menu.
        - MbreakP (PMOS transistor)
        - MbreakN (NMOS transistor)
        - EGND (ground)
        - VDC (DC voltage, for 2 inputs and power supply)

    You can rotate (CTRL-R) and flip (CTRL-F) the components in order to place them correctly in the schematic.

Schematic with trasistors	Pspice schematic
	

Note that the NMOS transistors are connected in series, and the PMOS in parallel.

- Edit the MOSFET transistor model:
  - Using *Edit→Model→Edit Instance Model (Text)*), and adjust Kp (transconductance) and Vto (threshold voltage V_T)

model MbreakP PMOS vto=-1 kp=15u *$	model MbreakN NMOS vto=1 kp=15u *$

- Specify the width (W) and lenght (L) of the channel by clicking on the symbol of the transistor and changing attributes: W = L = 1u (1u = 1 micro = 10^{-6})

Note that, to simplify the analysis and design, the same values of Kp and (W/L) to 2 transistors have been assigned.

Important: when replicating a single transistor several times, once should edit the transistor model only one time, specifying their parameters. After editing the model, select it (by clicking on it) and copy it (*Edit→Copy*) on the schematic. If doing so, we ensure that all models are identical.

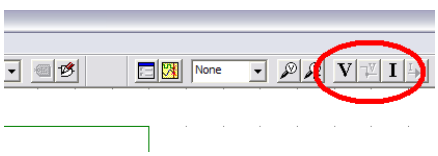
Another thing to remember: the substrates of **all** NMOS transistors must be connected to ground (GND), and substrates of the **all** PMOS transistors must be connected to V_{DD}.

Simulate the gate operation, writing its truth table:

A	B	Vs (voltage)	Vs (logical, value)	Static consumption
0V	**0V**			
0V	**5V**			
5V	**0V**			
5V	**5V**			

To do this:

- Enable the type of analysis as Bias Point Analysis. (*Analysis→Setup→**Bias point detail***).
- Modify the logical value of the inputs ("0" = 0V, "1" = 5V).
- Run the simulation (F11 or *Analysis→Simulate*).
- Check the voltage at the output selecting the **V button** of *Schematics* screen.



- Verify that the **output voltage values are very close to ideal ones**: V_{DD} for "1" lógic and GND for "0" logic.
- As there are 4 possible combinations for 2 inputs, you must do 4 simulations. Check (using the **I button**) that the static current consumption is negligible in all of them. The static consumption is the current coming from the power supply to the circuit.

Note: the abbreviations:

u = micro = 10^{-6}

n = nano = 10^{-9}

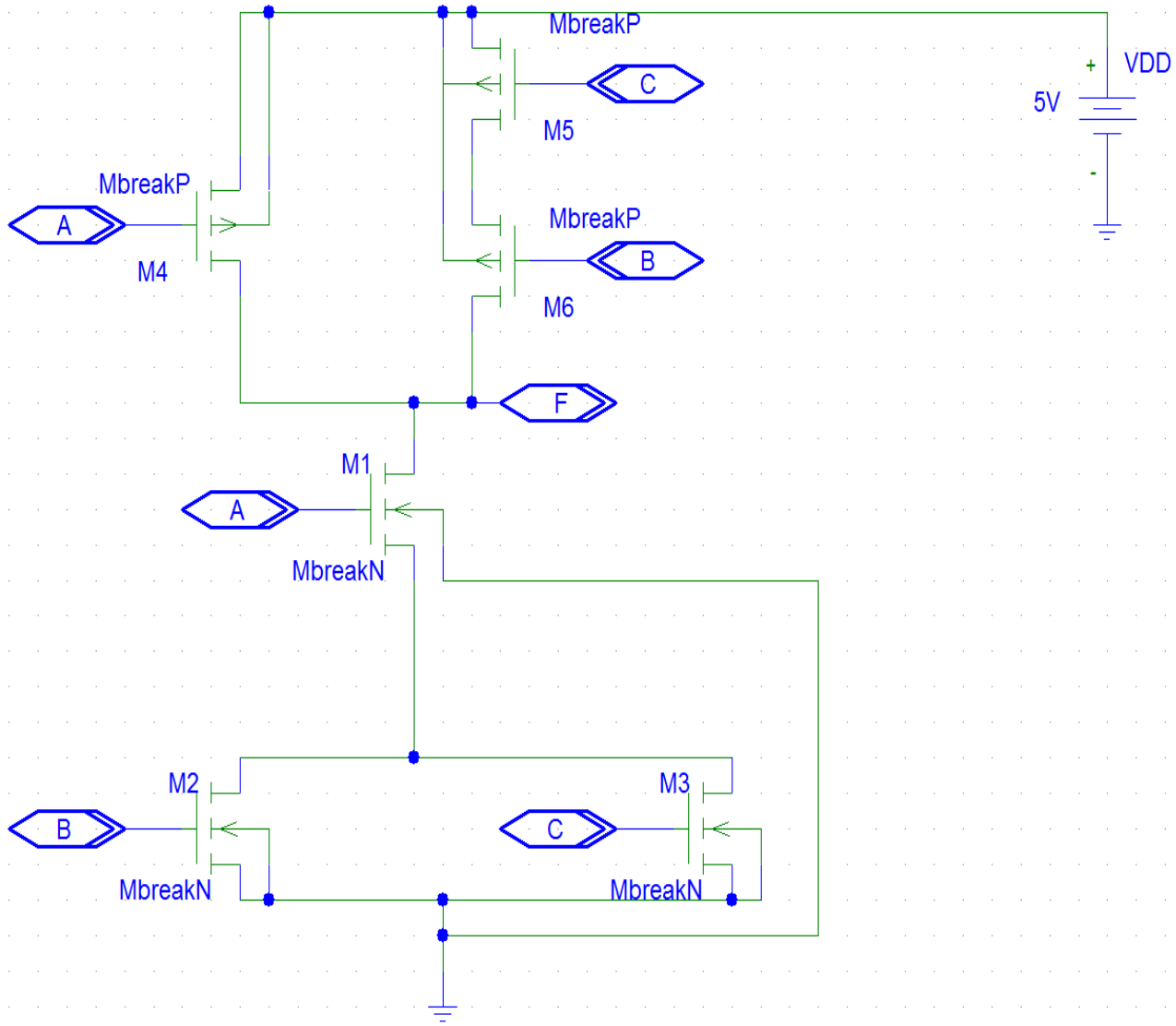
p = pico = 10^{-12}

3.2. Generic logic function

- **Edit in Schematics** the CMOS circuit with transistors for the following logic function:

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

From this logic equation it follows that the NMOS block function is: $G = A \cdot (B + C)$ AND operator corresponds to a series association of NMOS transistors, and the OR operator to a parallel association. We can so draw the NMOS block, placed between the output and ground, as shown in the following figure. Meanwhile, PMOS circuit block is obtained with a dual structure, changing parallel by serial associations, and vice versa. PMOS block is between VDD and the output, as shown in figure.



- **Aspects to take into account when editing the circuit**:
  - Use the same parameters for the transistors than in paragraph 3.1.
    - Specify a NMOS transistor and replicate it as many times as needed.
    - Do the same for the PMOS transistor.
  - Connect the circuit to a supply of +5 V. Use the part VDC to V_{DD}.
  - Remember to connect the substrates of NMOS transistors to ground and the substrates of PMOS transistors to V_{DD}.
  - To achieve a faster simulation, go to *Draw→Get New Part→Part Name,* and set the inputs as ***IF_IN (digital inputs).*** Clicking 2 times on the symbol, attach its name (A, B, C, or D).
  - Define F as ***IF_OUT output (digital output).*** Place it in the output ans assign the name as F.

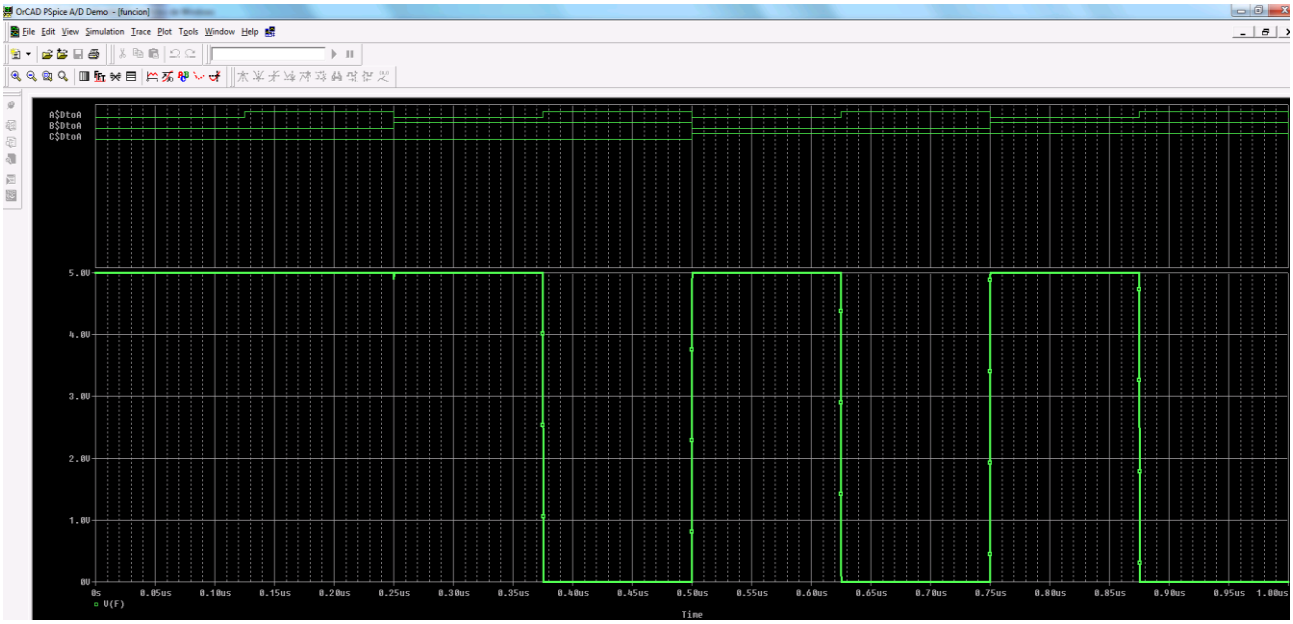
- **Create a file of stimuli**, to define the logical combinations of input signals:
  - Select *Analysis →Edit Stimuli*. In the editor window appears stimuli (*Stimulus Editor*).
  - Specify input signals (A, B, C, D) as digital square waves of different frequencies. For each input signal:
    - *Stimulus → New*
    - *Name*
    - *Digital→ Clock*
    - Specify the frequency of the signal. To display the 8 possible combinations of A, B and C, duplicate the frequencies. For example:
      - C: 1M (1MHz → Period T = 1u)
      - B: 2M (2MHz → Period T = 0.5u)
      - A: 4M (4MHz) → Period T = 0.25u)

  - Save the stimuli file (.stl)

- **Perform an Transient analysis** to see the response (F) over time:
  - Activate *Analysis→ Setup→Transient* in the *Schematics* window.
  - Specify the time duration of the analysis: As the lowest frequency is 1MHz, the period is 1us (1 micro-second). Then , indicate as *Final Time→*1u.
  - Run the simulation (F11 or *analysis→Simulate*).
  - Once completed, the *Probe* window will be automatically opened, which allows signals to be graphically represented. To make the selection of the signal, activate the *Trace→ Add Trace*. Select the 4 digital input signals and the output signal, V(F).

CAUTION: to view correctly the inputs, you should disable the analog voltages label. In this way, only digital signals are shown.
To view correctly the output, select the label of analog voltages. Thus, the output signal can be seen in more detail.

- **Check for proper operation of the circuit**, analyzing the chronogram. As you can see, in a single simulation all possible logical combinations are shown. If inputs were defined as VDC (as in the case of NAND gate of paragraph 3.1), it would have been required as many *Byas point detail* simulations as input combinations (8 simulations). Therefore, if you have a relatively large number of logic inputs, it is recommended to define a file of stimuli and to run a *Transient analysis*

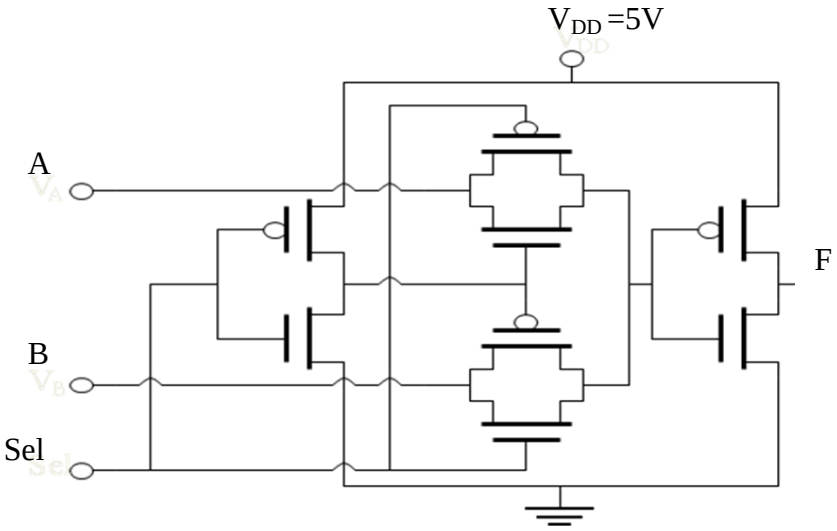


- **Verify that the static consumption is zero for all input combinations**. For this, display the current supplied by the source, I (VDD). You need to create a new graph using Add_Plot command.
-

  Current peaks can be seen in the transitions of the inputs, due to the dynamic short circuit current consumption.

3.3. OPTIONAL Paragraph: 2x1inverter Multiplexer

- Edit the schematic of the figure below, corresponding to a 2x1 Mux with inverting output. Remember to connect substrates to GND for NMOS transistors and to V_{DD} for PMOS transistors.



- Use the VDC component for the inputs of channels A and B. Fix the values to: A= “1”, B= “0”.

- Use the IF_IN component (digital input) for the selection input (*Sel)*.

- Create a stimulus file with Sel signal, defining a square wave of 1MHz.

- Simulate (Transient) and observe F in function of *Sel*. If Sel=”1” channel B must be selected. If Sel=”0” channel A must be selected. Remember that the output is inverted.

In conclusion, the inverted behaves like this:

Sel = “1” → $F = \overline{B}$

Sel = “0” → $F = \overline{A}$