Analog Integrated Circuit

Project: I

Task 01-Characteristic of a MOS transistor

a Start the Orcad Capture CIS program and create a new project named "Project01". Draw the circuit shown in Fig. 1 with Capture. Use the PSpice MOS model "MbreakN" from the BREAKOUT library for the MOS transistor. This is a modifiable NMOS model. Modify the name of transistor from "MbreakN" to "nmos_tsmc".[1] Change also the names and values of voltage sources as shown in the schematic.

b Change the width and the length of the transistor by double clicking on the transistor symbol. In the popup window you can enter the values for with (W=1u) and length (L=250n) (W=1 μm and L=250 nm).

c Change the value of the DC-voltage source VGDC to 0.5V ($V_{GS,DC}$=0.5V) and the value of the DC-voltage source VDDC to 0.9V ($V_{DS,DC}$=0.5V).

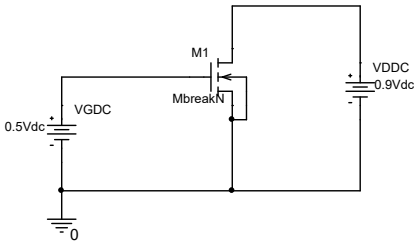

Figure 1: Circuit for simulating the DC characteristics of the transistor.

Configuration of the PSpice Model Library

PSpice searches model libraries for the model names specified by the MODEL implementation for parts in your design. These are the model definitions that PSpice uses to simulate your circuit. For PSpice to locate these model definitions, you must configure the libraries.

In this lab we use the model file `tsmc_250nm.lib`. Please download the model file (available under: TSMC CMOS 250nm Model) and store it in on your computer.

This is a text file which contains the model parameters to describe NMOS transistors in a 250nm CMOS Technology of TSMC. The content of this text file is

[1] The names of devices can be modified by double clicking the on the textbox showing the name of each device (e.g. for the transistor "MBreakN"). In the popped up menu ("Display Properties"), the name can be modified.

shown for convenience in Fig. 2. The model is the so called "Level 3" MOSFET model. More details regarding the definitions of model parameters can be found e.g. under Comparison of Level 1, 2 and 3 MOSFET's.

```
*********************************************************************
*************     NMOS  MODEL

.MODEL nmos_tsmc NMOS ( LEVEL  = 3
+ TOX     = 5.7E-9          NSUB    = 1E17            GAMMA   = 0.4317311
+ PHI     = 0.7             VTO     = 0.4238252       DELTA   = 0
+ UO      = 425.6466519     ETA     = 0               THETA   = 0.1754054
+ KP      = 2.501048E-4     VMAX    = 8.287851E4      KAPPA   = 0.1686779
+ RSH     = 4.062439E-3     NFS     = 1E12            TPG     = 1
+ XJ      = 3E-7            LD      = 3.162278E-11    WD      = 1.232881E-8
+ CGDO    = 6.2E-10         CGSO    = 6.2E-10         CGBO    = 1E-10
+ CJ      = 1.81211E-3      PB      = 0.5             MJ      = 0.3282553
+ CJSW    = 5.341337E-10    MJSW    = 0.5             )
*********************************************************************
*********************************************************************

*********************************************************************
*************     PMOS  MODEL

.MODEL pmos_tsmc PMOS (                                 LEVEL   = 3
+ TOX     = 5.7E-9          NSUB    = 1E17            GAMMA   = 0.6348369
+ PHI     = 0.7             VTO     = -0.5536085      DELTA   = 0
+ UO      = 250             ETA     = 0               THETA   = 0.1573195
+ KP      = 5.194153E-5     VMAX    = 2.295325E5      KAPPA   = 0.7448494
+ RSH     = 30.0776952      NFS     = 1E12            TPG     = -1
+ XJ      = 2E-7            LD      = 9.968346E-13    WD      = 5.475113E-9
+ CGDO    = 6.66E-10        CGSO    = 6.66E-10        CGBO    = 1E-10
+ CJ      = 1.893569E-3     PB      = 0.9906013       MJ      = 0.4664287
+ CJSW    = 3.625544E-10    MJSW    = 0.5             )
*********************************************************************
```

Figure 2: The content of the he model file `tsmc_250nm.lib`. The model file contains one NMOS model "nmos_tsmc" and one PMOS model "pmos_tsmc".

It is recommended to define the device models by adding the model library to the entire PSpice project. To add the library, you have to first create an arbitrary simulation profile: Open the window New Simulation ("Pspice" → "New Simulation Profile").

Enter an arbitrary name for the simulation profile (e.g. "DC Characteristic").[2]

In the "Simulation Setting" window, click on the tab "Configuration Files" and under the tab "Category", select "Library" and then click on "Browse..." browse and select the model file. Then click on the "Add to Design" tab at the right side to add the model file to the current PSpice project (See Fig.3).

d Change the model name of the transistor to "nmos_tsmc" (By double clicking on the model name "MbreakN" a pop up window appears. Replace "MbreakN" in the Value field by "nmos_tsmc".) The modified circuit is shown in Fig.4.

[2]This simulation profile will be used again in part c) to simulate the output characteristic of the transistor, therefore we chose the name "DC Characteristic". The configuration of model library is, however, valid for all other circuits and simulation profiles defined afterwards in this PSpice project.

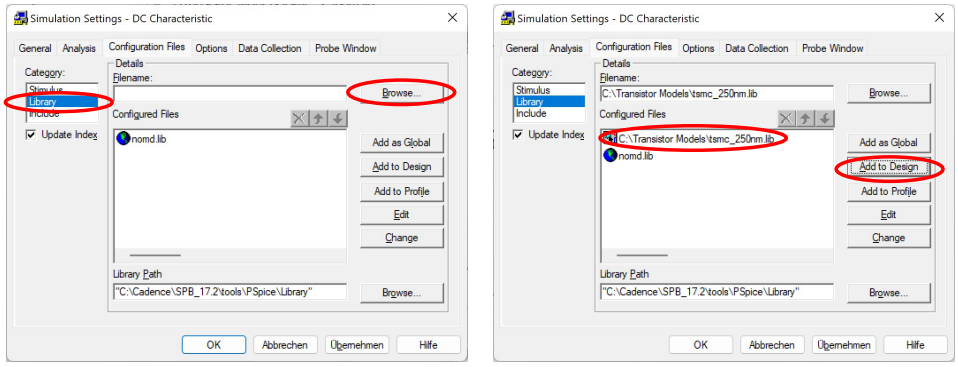

Figure 3: Configuration of the library file in PSpice.

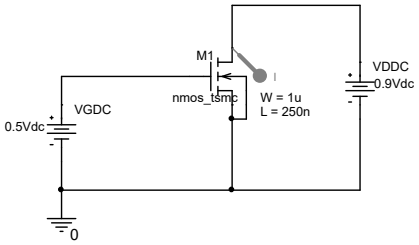

Figure 4: Simulation setup to plot the output and input characteristic.

Simulation of the input characteristics

e Select a current marker in the capture window via PSpice → Markers→ Current into Pin and place the marker at the drain contact of the transistor. This current marker will cause the current flowing into the drain terminal to be automatically displayed in the probe window.

Open the Simulation Settings window for the "DC Characteristic" simulation profile ((PSpice → Edit Simulation Profile) and select "DC-Sweep" under Analysis Type.

Under the Primary Sweep select "Voltage Source" and under the Name specify the attribute name of the DC voltage source (VGDC). Under Sweep Type specify Linear, Start Value=0, End Value=1, Increment=0.01 and click on the button "OK" and run the simulation.

The value of the DC voltage source is varied with this setting from 0V to 1V in 0.01V steps. The drain current versus the gate-source voltage ($I_D = f(V_{GS})$) for a constant drain-source voltage ($V_{DS,DC}$=0.5V) is automatically shown in the Probe window. This is referred to as *input characteristic of the transistor*.

f Determine the transconductance g_m of the transistor at $V_{GS} = 0.5$ V by calculating the slope of this curve:

$$g_m = \frac{\partial I_D}{\partial V_{GS}}\bigg|_{V_{GS}=0.5V}$$

Hint: In Probe window activate the cursors by "Trace→Cursor →Display". Use the left and right mouse button to move the cursors A and B around the operating point and determine the changes in drain current ΔI_D for a "small" change of gate-source voltage ΔV_{GS} around the DC-Operating point (in this case $V_{GS} = 0.5V$) and then determine the slope of the curve $\frac{\Delta I_D}{\Delta V_{GS}}$.

Alternatively, you can use the derivative function "D()" in the Probe window (under Trance→Add Trace→Choose under Analog Operators and Functions "D()" (derivative) and then choose ID(M1): The trace Expression should look like D(I(IM1). PSpice calculates the derivative of the drain current with respect to the swept variable (x-Axis). The value of the derivative at $V_{GS} = 0.5V$ is the corresponding transconductance value.)

Extraction of the threshold voltage from the input characteristics.

g One method to experimentally "extract" the threshold voltage for a transistor is to sketch the tangent line in the input characteristic field at the point of maximum rise and determine the intersection point of the tangent line with the $U_{GS,DC}$ axis (see Fig. 5). Use this method to approximately determine the threshold voltage V_{TH} of the transistor.

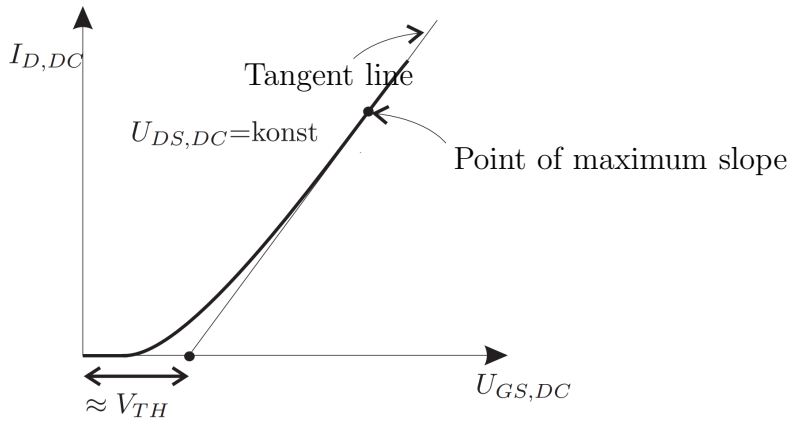

Figure 5: Graphical method for approximating the threshold voltage V_{TH}.

Simulation of the output characteristics.

h Modify the simulation setup to plot the output characteristics (I_D vs. U_{DS}) in the range $0V \leq U_{DS} \leq 2V$ for $V_{GS} = 0.5V$.

i Use the value of V_{TH} you obtained in part g to determine in the output characteristic field where the transistor is working in linear (triode) region or in the saturation region.

j Determine the small signal output resistance of the transistor $r_{ds} = r_o$ at $V_{DS} =$ 0V, at $V_{DS} = 50$ mV, and at $V_{DS} = 0.9$V.

Note: $r_{ds} = r_o$ is the inverse of the slope of the curve I_D vs. U_{DS}:

$$r_o = \left(\frac{\partial I_D}{\partial V_{DS}} \Big|_{V_{DS}=0.9V} \right)^{-1}$$

k Change the channel length of the transistor (parameter L in the property editor) from $L = 0.25\mu m$ to a larger value (e.g. $L = 1\mu m$) and adjust the width of the transistor such that approximately the same drain current flows through the transistor at $V_{DS} = 0.9$V. Repeat i and j.
How does the output characteristic field of the MOS transistor and r_o change? Explain why this behavior is expected.

Extraction of small signal parameters using DC Bias Point Analysis

When you connect a battery or a power supply to a circuit, the circuit voltages and currents effectively settle down to what is known as a DC steady-state condition. This is also known as the operating point or Bias Point of a circuit under steady-state conditions. In PSpice, the Bias Point analysis calculates the node voltages and currents through the devices in the circuit.

l Create a new *Bias Point* simulation simulation profile for the circuit shown in Fig. 4. Run the simulation.
On the Capture/PSpice toolbar, click the "Enable Bias Voltage Display", and the "Enable Bias Current Display" buttons to display the results of the dc bias point analysis directly on the schematic. Determine I_D and ensure that you get consistent values as being determined by the output and input characteristics in previous steps.

m Now modify the simulation setup (PSpice →Edit Simulation Profile) as shown below (see also Fig. 6): Check the "Include detailed bias point information ...". Using this setup PSpice PSpice calculates the small signal parameters of all non-linear devices in the circuit '(transistors, diodes, etc.)
After running the simulation, the PSpice A/D Demo window opens. In the upper toolbar click in "View" → "Output File" to view output simulation file.
Scroll to the bottom of the output file to find the results of the small-signal analysis below the line `**** SMALL-SIGNAL CHARACTERISTICS` and determine the values of g_m, r_o (See Fig. 7). Compare the results with the results in parts f and j.

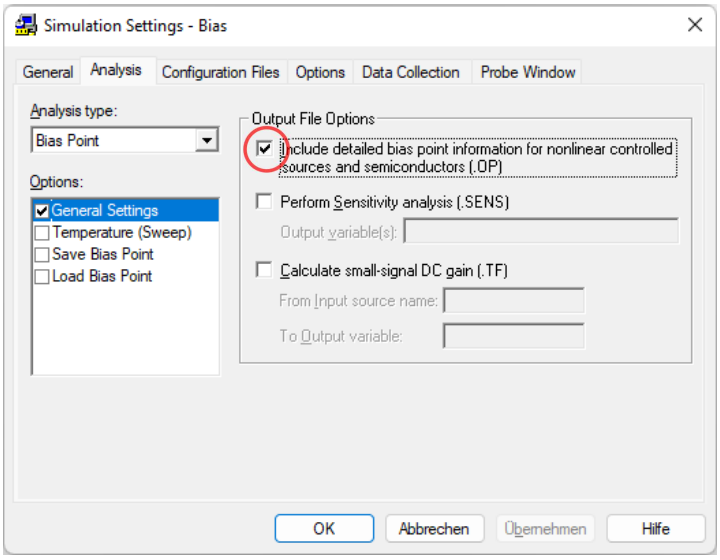

Figure 6: *Bias Point* simulation setup to determine the small signal parameters of the transistor.

**** MOSFETS

NAME	M M1
MODEL	nmos_tsmc
ID	$I_{D,DC}$
VGS	VGS
VDS	VDS
VBS	VBS
VTH	V_{TH}
VDSAT	VDSAT
Lin0/Sat1	Lin0/Sat1
if	if
ir	ir
TAU	TAU
GM	g_m
GDS	$g_{ds} = r_{ds}^{-1}$
GMB	GMB
CBD	CBD
CBS	CBS
CGSOV	CGSOV
CGDOV	CGDOV
CGBOV	CGBOV
CGS	CGS
CGD	CGD
CGB	CGB

Figure 7: Simulation output file, showing the calculated small signal parameters of the transistor.