IN3170, spring 2026, mandatory laboratory exercise 1: Lab equipment and circuit simulation to charactarize a pFET (deadline 3-March-2026, 10:15!)

P. Häfliger & Halvor Klepsvik
Institute of Informatics
University of Oslo
e-mail: hafliger@ifi.uio.no

February 5, 2026

change log	
22-Jan-2026	First release
28-Jan-2026	Corrected some URL-paths from 'v25' to 'v26'.
5-Feb-2026	Corrected for task 2: "Keep V_D at **Gnd**, corrected schematic to use Gnd as reference for all sources. Also the sweep matlab functions have been changed to wait only 0.5 seconds instead of 2 seconds between sourcing and measuring (should be long enough and makes things much faster)

Abstract

For the following labs we are going to use electronics lab equipment as well as the advanced integrated circuits simulation tools in the software package Cadence. This first lab task is mostly to get acquainted with those tools and to use them for the first time for some circuit analysis.

In summary we will obtain the V/I curves of single transistors in simulation and from physical devices and try to extract the three main independent EKV model parameters I_S, V_t, and λ from those curves.

This is the first in a series of three lab tasks that will be graded and will count 40% towards the final grade. The first lab, however, will only be 'pass' or 'fail'. The students are required to pass this lab assignment to be admitted to the exam. The deadline in the title is a hard deadline! Do not miss it! Plan to submit well ahead of the deadline! We will use devilry.ifi.uio.no for submission of you lab report.

1 Report and Group Assignments

1.1 Requirements for the Lab Report (read carefully!)

You are required to execute the tasks and answer all the questions posed below and to submit a report on your work. The report needs to be explaining clearly what you have done, how you have done it, what the results were and what you conclude from them. A good basic rule for any scientific writing is that a document should provide sufficient detail for someone reading it to be able to replicate the results that are presented. Make sure to answer all questions! Supply the report with drawings of the circuits (including the values of the components and parameters you used where appropriate, e.g. bias voltages/currents, component sizes etc.) and measurement setups! Show your measurements in graphs! Use labels in the schematics that you draw, such as M_1, M_2 (M is often used for labeling CMOS transistors), $opamp_1$, I_1, V_1 etc. You should then use those labels in your text, since it is much easier to write: 'transistor M_1 in figure 1' than 'the transistor third from the top and second from the left in the righthand side circuit in figure 1'. **MANDATORY when assembling a circuit in the lab**: Include a photograph of your circuit into the report!

1.2 Graded Mandatory Group Assignments

Note that this is part of the course's exam and strict rules apply as described in the document https://www.uio.no/english/studies/examinations/compulsory-activities/mn-ifi-mandatory.html . The page explains the significance of mandatory assignments in a course and in particular group assignments. It also specifies your responsibility to not plagiarize anybody else's work and that you are required to conduct and understand your own experiments and obtain your own results, while you are still allowed and actually very much encouraged to exchange advice and experiences also between groups.

Each group must deliver a written lab report using the Devilry online submission system **before** the **hard** deadline indicated in the title. Note that you can submit multiple times and the last submission before the deadline will be graded, so it might be a good idea to plan to submit preliminary versions well before the deadline. The points given for this lab assignment will determine if the lab assignment is accepted or rejected. You will need to pass this lab assignment in order to be admitted to the exam. The next two lab assignments will be weighted as 20% of the total score of the course, i.e. your final grade.

Each task is labeled with how many points it will contribute towards the score.

1.3 Safety

Voltages over 40 Volt can in some cases be harmful, even though it usually requires more than that. The lab equipment is thus not able to provide voltages higher than 36V. Do not use equipment other than that provided in the lab! If

a part of the skin is covered with a conductive fluid or is pierced and exposed to such voltages, a current could bypass the "insulator" of the skin and run through the body. If this current passes through the heart it can cause fibrillations or even cardiac arrest. Even higher more extreme currents could also give rise to internal burns. If this happens to anybody or something else happens in the lab, seek medical help immediately: heart fibrillations can last and cause trouble long after the incident. Also notify the person in charge of the lab.

Some electronic components can explode if they are exposed to high currents. This is important to remember when working with electrolytic capacitors. However, none of the capacitors provided in the lab are electrolytic capacitors. Never bring your own electronic components into the lab!

1.4 Conduct

Good routines are necessary to make the work in the lab effective and safe:

- Food and drinks are prohibited from every lab.
- In general everybody is responsible for keeping the lab tidy.
- Always turn off the power supply before you start adding and/or removing components.
- Use an ESD protection wrist strap when handling ICs and other sensitive components.(ESD: electrostatic discharge)
- Always clean up after using equipment and tools:
  - Turn off all equipment, except for lab computer.
  - Throw away cutoffs and vacuum clean the desk, chair, and floor if necessary.
  - Place all components you have used back to their respective places. (Do this while you work, if you have a component you don't use anymore, put it back.)
- When you leave; the desk should be clean and ready for the next group.
- Read the information posters in the lab describing what to do in case of fire or medical emergency.

2 Task

2.1 Introduction

First you'll acquaint yourself with the lab equipment that can be interfaced via MATLAB functions, via the LAN in the lab. Furthermore, you will be using the schematic design tool as well as the simulation tool of the circuit design software Cadence. And last but not least we'll be analysing parameters of FETs both

in the lab and in simulation, i.e. for a physical IC in the lab, and simulated transistors in a 65nm CMOS technology in Cadence. You will also learn some soldering skills to be able to connect the IC under test properly on a printed circuit board (PCB).

2.2 Tools

- **Linux command line interface:** To run Cadence you'll have to start a remote session on the Linux sever rh7login.ifi.uio.no. The whole process is described on the web-page: https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html. In Linux you'll be working with a command line interface to start the relevant software (Cadence). The command line interface is called 'terminal' or 'xterm'. On rh7login.ifi.uio.no you'll open a 'Terminal' by choosing 'Activities' from the top bar menu and then click on the 'Terminal' icon. On the lab machines there is a 'console' terminal open when you log on that you can use. Else you can also choose 'Applications' in the top bar menu and then 'System Tools' and then 'Terminal'.

  When you open a terminal, your 'home directory' is the active directory by default. The active directory is the default directory where a program that is started stores any files it creates and where it is looking for files it needs first. Your 'home directory' is a directory on a file server where you can store your files and that is accessible from any Linux machine on campus. From Windows machines on Campus you access it as the disk 'm:'. It also contains a directory called 'pc' which is the directory containing your typical 'Desktop' and 'Documents' and similar Windows folders. You can, for instance, create a sub-directory in your home directory called 'IN3170' to work in with your labs, and possibly more subdirectories for individual labs to keep things organized. To make a directory use the command 'mkdir <directory_name>' and to change the active directory 'cd <directory_name>' or 'cd ..' to go 'up' to a parent directory.

- **MATLAB** is a mathematics calculation tool and script language. it is also very useful for making nice graphs, e.g. of measurement results, in a lab report (This is ment as a hint!).

  We will be using MATLAB in the digital exam (!!!) to express symbolic results where a variable is expressed as a function of other variables and parameters, and as the calculation tool for numeric results. So you will need some basic knowledge of Matlab syntax to be able to make simple functions and scripts. Check out some of the exercises in the compendium for examples!

  Please refer to the online html-document https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html for instruction on how to get a copy of matlab installed on your home computer. It IS also installed on rh7login.ifi.uio.no, as well as on the Windows computers

in the course lab OJD5419, and the virtual machines 'Lab Net Workstation' from which you will control the lab instruments. Under Linux, you simply type 'matlab' in a terminal to start it.

One useful function in MATLAB is 'help'. 'help <command_name>' will display an explanation on how to use '<command_name>'. Another help function that helps you find functions that you do not know the exact name of is 'lookfor'. Type 'help lookfor' to learn more.

- **Cadence** is an industrial SW tool for integrated circuit design. When using it for simulation, its models are much more detailed than the hand calculation models you learn about in the course. These models include a lot of secondary effects and have a lot more parameters and are able to predict with much higher accuracy how a circuit will really behave when coming back from production. Still you'll see that the hand calculation models are a useful approximation in many cases.

  We will be using the Cadence modules "Schematics Editor" and "Analog Design Environment explorer" in this course.

  In order to use Cadence (and potentially other SW that runs on a Linux server at IFI/UiO) you will need the means of logging onto such a server and have a user interface to it presented on your home/UiO computer. I have made a HTML document that explains how to do that for Windows, Linux, and Mac PCs here: https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html The document also explains how to set up Cadence and how to use its tools.

- **Soldering on a PCB** You will need the skill of soldering components and cables to a printed circuit board (PCB) for this and future labs.

  There are pipes with air vents at each work place that suck in air: turn them on (green button on the desk) and suck in the fumes that occur during soldering. Small doses of those fumes are unproblematic so there is no real reason for concern, but substantial continuous exposure over a long time (e.g. if someone continuously works with PCB assembly as their job) to such fumes can be unhealthy.

  When soldering any kind of conductor to a PCB pad you want to establish an electric contact between that conductor and the pad on the PCB surface. To make a proper contact you need to heat BOTH the pad AND the conductor, so try to touch BOTH with the soldering iron. The solder tin only need to touch one of those two and will melt automatically if that surface becomes hot enough. So I usually hold the soldering iron to the conductor and try to place the tip on the pad on the PCB, while I hold the solder to the conductor from the opposite side. Then the solder will liquefy automatically as the conductor and pad become hot enough and the connection between those two is thus guaranteed to be good. If the pad becomes hot enough as well you will notice the liquid solder 'flowing out' onto the pad. In contrast, if you do not heat the pad enough the

solder may look like it touches the pad as a bobble 'tucking in' at the contact point instead of flowing out. It may still touch and may establish a finicky contact, but it is not solidly 'glued' to the pad, may be unreliable and can get loose.

You'll find some good video on this on Youtube, for example: https://www.youtube.com/watch?v=vAx89WhpZ3k

- **IC CD4007UBE with 6 transistors** There will be generic PCBs in the lab where you shall solder on an IC labelled 'CD4007UBE' and later also other components and where necessary some cables. The IC contains 6 transistors and some interconnections. Check the data sheet 'cd4007ub.pdf' under 'resources' on the course webpage. We will only need a single nFET from it for lab 1, so pay attention to the interconnections already on the IC and pick the most suitable nFET, where the hard ired connections do not interfere.

- **Lab instruments** will be used to measure and characterise some experimental circuits in the lab. You can control them locally manually for single measurements and to set up the right instrument parameters or remotely via MATLAB functions, also to execute series of automated measurements and to write results into MATLAB readable files for making illustrations. Check out this HTML page for a list and descriptions of the MATLAB functions: https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html

  The four instruments are:

  Oscilloscope: This instrument is for recording voltage traces over time. Note that scope probes also have some input impedance (capacitance and resistance) that might influence the node that you measure! Use the identifier "mdo" in LabInst_GetHandle.

  DC power supply: This instrument provides multiple programmable stable voltage outputs on corresponding output channels. Use the identifier "gpp" in LabInst_GetHandle.

  Multi-meter: This instrument measures a number of electrical units, typically a voltage or a current. Note that you may connect different input ports to measure either voltage or current! Use the identifier "gdm" in LabInst_GetHandle.

  Function generator: This instrument produces voltage traces over time. Typically from a selection of standard waveforms, such as sine waves or square waves. Use the identifier "gfm" in LabInst_GetHandle.

There are extensive user manuals for the instruments to be found on drive 'L:' on the LabNet server machines that you'll use in the labs, if you want to investigate advanced features and implement more functions than the above. You may have to mount drive L: on those machines first according to instructions: https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html

2.3 Lab Tasks

task 1 (2p), set up lab equipment: In short, use the command line in MATLAB and verify the proper function and control of the power supply and multimeter.

Follow instructions on https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuides.html to log onto the 'Lab Net Workstation' and to start MATLAB. Set up a connection from your power supply output channel of choice to the voltage measurement terminal on the multi meter. Use 'LabInst_GetHandle' in matlab to get object handles to those two instruments. Use GPP_SetVoltage to set some output voltages and verify that they indeed appear on the display as the current output voltages. Check that they also appear on the multi meter. Use GDM_GetVoltage to verify that you can actually read that measurement into MATLAB. Also find an arbitrary resistor (report the value used) and hook it up to the power supply and current (!) measure input of the multimeter. Note that to measure current you need the current meter IN SERIES with the device you want to measure. Manually verify that you measure the corresponding current for different voltages. Pay attention to having the current limit of the power supply high enough so it does NOT limit the current you want to measure, i.e. it is able to supply the voltage that you specify.

a) Make a voltage sweep and measure that voltage and make a plot (which should show the 'identity' function, with a bit of noise) for the report.

b) Make a voltage sweep and measure the resistor current and make a plot for the report. It should show the function $I = \frac{V}{R}$.

There is a function for a voltage sweep and current measurement already among the Lab instrument functions that are provided on the courses web page, called 'LabInst_IvsV'. You may adapted it to create another function to measure voltage as well, or Philipp may do it before the lab starts.

task 2 (4p), measure a PFET and compare to EKV: In short, Measure the I/V curves of a single PFET on the IC CD4007UBE and compare the result with the EKV model by figuring out the parameters I_S, V_{tp}, and λ.

Use the general purpose PCB provided in the lab and solder on the IC CD4007UBE. Check the data sheet and make appropriate connections to be able to measure a single PFET's I_D vs V_{GS} and I_D vs V_{DS} curves. Make sure to connect the power supplies correctly. These discrete transistors

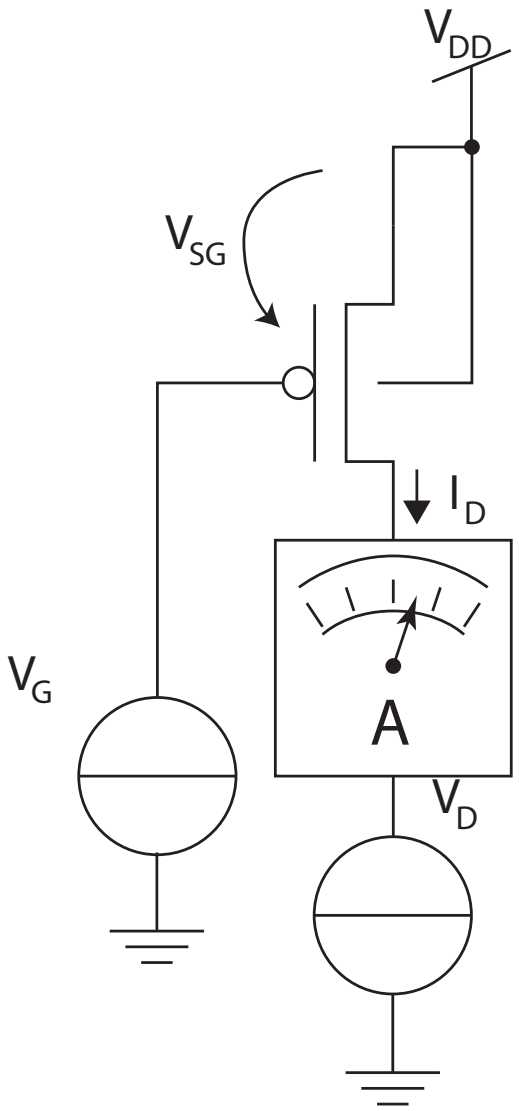


Figure 1: Measurement setup task 2 (and 3)

operate well with a 5V supply, so let us use 5V for power from channel 3 of the "gpp" power supply. Pay attention to supply the bulk of the pFETs with these 5V, and the bulk for all nFETs to Gnd. Make it so that I_D is measured on the multi meter (needs to be connected IN SERIES to measure current!!!) and use two channels on the power supply to control V_D and V_G, and the third to supply 5V to the pFET bulk. Measure and plot both the I_D vs V_{SG} and I_D vs V_{SD} curves. For the I_D vs V_{SG}, keep V_D at Gnd. For the I_D vs V_{SD}, plot three curves with three different V_{SG}, one clearly above the threshold voltage (strong inversion), one roughly equal to the threshold voltage (moderate inversion) and one below the threshold (weak inversion). (You might have to redo some of these AFTER you have determined the threshold voltage V_t.)

Now give it some thought on how you can get the model parameters I_S, V_t and λ from those plots. Assume $n = 1$. Document well how you do it! Hint on V_t: try a method to fit a quadratic function to the strong inversion part of your curve. V_t is where that quadratic function has its minimum.

Enter the parameters into the EKV Matlab function (You'll find it under 'resources' on the course page) and plot the result together with your measurement to verify your match. Show that first result! Then, if you see some discrepancy, try to reason on how to tweak your parameters further to improve the match and report again. How good can you make it? Are there any deviations that indicate that there are some effects that the EKV model does not model?

task 3 (4p), simulate a PFET and compare to EKV: In short, simulate a minimum size PFET with in the TSMC 65nm technology and again produce the same set of plots and derive the model parameters as before for the real life transistor in the lab. One significant difference will be that the supply voltage should be 1.2V only. Do the curves look otherwise qualitatively similar? What are the similarities and differences to the physical experiment in the lab?

The normal transistor models to use in this technology's process development kit (PDK) are called 'nch' and 'pch' respectively and are in library 'tsmcN65'. You'll want to conduct a 'dc' simulation and choose once drain and once the gate voltage as sweep parameter. Use the cell 'vdc' in library 'analogLib' to provide signals and supplies. The power supply is 1.2V for this 65nm technology.

Check the online guide https://www.uio.no/studier/emner/matnat/ifi/IN3170/v26/materiale/LabAssignmentGuide.html! After a simulation run, in Cadence's main command line window (!), use the function

```
ocnPrint(i("/M0/D" ?result 'dc) ?numberNotation 'none ?output "out.ocn" )
```

to export an output current labelled "/M0/D" which is the current at the

drain of device "M0" from the current dc simulation. Then in MATLAB use

```
m=readCadenceOcnPrint("out.ocn")
```

. This gives you a 2 column matrix m where the first column is the time and the second the current. So you can plot it in MATLAB with

```
plot(m(:,1),m(:,2))
```

, where $m(x, y)$ refers to the element in the x^{th} row and y^{th} column, and a ':' indicates the entire range of an index, so $m(:, y)$ refers to the entire y^{th} column and is thus a column vector. So the command plots the second column of the matrix vs the first.

As before make one I_D vs V_{GS} plot and three I_D vs V_{DS}, the latter with V_{GS} to set the operation ranges to weak-, moderate-, and strong inversion. Using the data in MATLAB, figure out the three parameters I_S, V_t and λ again. Once again, plot both the EKV curve and the measured curve together. You will most likely see some effects not modelled by EKV: comment!