IN3170, spring 2025, mandatory laboratory exercise 2: bandwidth and stability of a source follower (deadline 31-March-2025, 10:15!)

P. Häfliger & Andreas Nungpau Engøy
Institute of Informatics
University of Oslo
e-mail: hafliger@ifi.uio.no

March 20, 2025

change log	
3-March-2025	First release
10-March-2025	Some more Matlab hints: semilogx, loglog
20-March-2025	Implemented function MDO_TimebaseScale.m to solve automated sweep issue and updated matlab code in this text for task 2.

Abstract

In this lab we will simulate and measure a source follower, aka common drain stage. Out of the three single transistor gain stages it is the only one that can show instability. We use it here as a vehicle to derive and study frequency response.

This is the second in a series of three lab tasks that will be graded and will count 40% towards the final grade. The first lab is only be 'pass' or 'fail' and the students are required to pass this lab assignment to be admitted to the exam. The second and third labs will get a score of between 0% and 100% and will each count 20% towards the course score. 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 fro labelling 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 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 labelled with how many points it will contribute towards the score.

2 Lab Rules

2.1 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!

2.2 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.

3 Task

3.1 Tools

The tools are the same as for lab 1. As an additional device, you will also us the waveform generator. Check the webpage that describes those lab devices and their MATLAB interfaces for an update!

- **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/v25/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.

  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/v25/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/v25/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/v25/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 to be found here (if you want to investigate advanced features): https://www.uio.no/studier/emner/matnat/ifi/IN3170/v25/materiale/Lab%20Instrument%20Documentation/

3.2 Tasks

Task 1 (8p): In short:

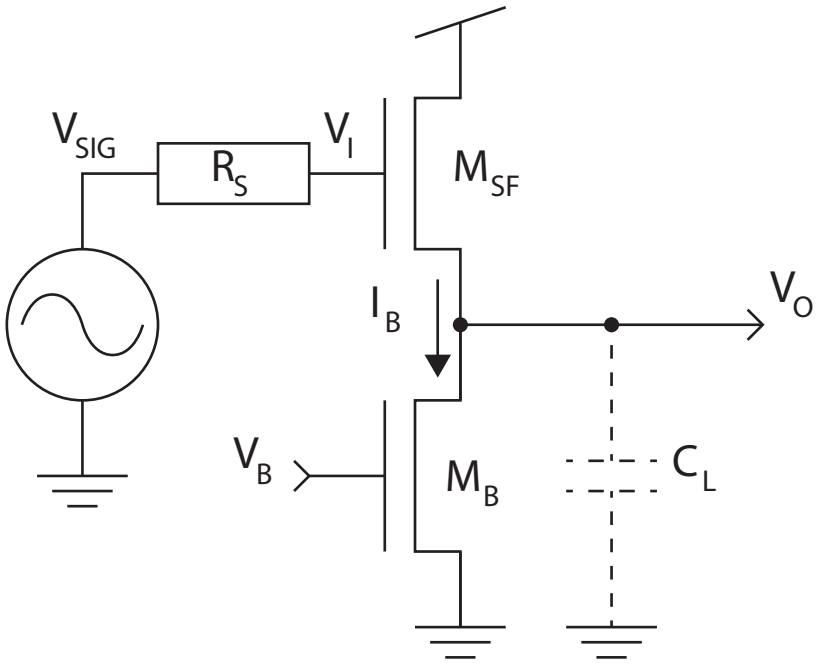


Figure 1: A source follower. M_B serves as a close to ideal current source, as long as it stays in saturation. As long as M_{SF} also is in saturation, its V_{GS} is dictated by the current provided by M_B. Thus the output voltage at the source of M_{FS} follows the input at its gate with a fixed offset V_{GS}. Note that the inputs point of operation cannot fall under $V_{GS} + V_{sat}$ to fulfil the requirement that both FETs be in saturation! In strong inversion that is $V_t + 2V_{OV}$ and in weak inversion $4U_T + V_{GS} = 4U_T + V_B$.

simulate a source follower/CD stage. With two different parameter settings it will first be unstable and then stable. Interpret the bode plots of the output.

More exhaustive:

Simulate a source follower as depicted in figure 1. The circuit will be discussed exhaustively later in the course. For now you will receive a brief description of its function and will then just analyse its frequency response.

The source follower (SF) is not really a gain stage, insofar as it does not amplify its input. As its name suggests it will merely move its output (almost) as much as the input moves. In other words its gain is slightly smaller than one.

Set up a simulation of the circuit depicted in figure 1. For both transistors, use dimensions.

$$W = 10\mu\text{m} \quad (1)$$
$$L = 1\mu\text{m} \quad (2)$$

For both FETs, connect the bulks to the source terminals!

Set the signal source output resistance

$$R_S = 10\text{M}\Omega \quad (3)$$

Use a bias voltage

$$V_B = 250\text{mV} \quad (4)$$

Do not use an explicit load capacitor C_L yet.

Do not forget a 'vdc' to provide the circuit supply of 1.2V for this technology.

Use a 'vpulse' as signal source. Selecting it and hitting 'Q', set its 'ac magnitude' to 1 and its 'dc value' to 995mV for the 'ac' small signal simulation. For a 'trans' simulation, set its 'period' to 4μs, its 'pulse width' to 2μs and its rise and fall times to 100fs. Configure a small step from voltag 1 of 990mV to voltage 2 of 1.000V.

Open 'ADE explore' to simulate. Set up 1) an 'ac' simulation where you sweep the frequency from 1Hz to 10GHz, and 2) a trans simulation to observe the step response. If you choose appropriate outputs to be plotted and run the simulation, you will see the step response in the trans simulation and the magnitude response (on a linear Y-scale!) in the ac simulation result.

One way of getting a Bode plot (i.e. the magnitude in dB as well as the phase) directly from Cadence is to go to the meastro cell view editor window and to choose in top bar menu:

Results → Direct Plots → AC Magnitude & Phase

Then select the nodes that are labelled V_{SIG}, V_I, and V_O in the figure (give them appropriate cable names in the simulation!) and hit the 'run' button. Unfortunately you have to go through the direct plot menu and the selection on the schematic each time for this!

Then to export results from this 'direct plot' use the label in the result plot window for the individual traces, e.g. for the phase of a signal labelled 'out', the appropriate command is:

```
ocnPrint(phase(VF("/out")) ?numberNotation 'none ?output "outPhase.m" )
```

And in Matlab use the utility function (found on the course's 'resources' web-page):

```
m = readCadenceOcnPrint( "outPhase.m" )
```

Note that the dB20 signals imported from Cadence already are in the logarithmic dB scale! So if you want them on a linear scale, you'll have to do the appropriate conversion. However, you do not really need that: ploting it on a linear y-scale will show the correct values in dB. To get a logarithmic x-scale you can use the Matlab function 'semilogx'. Type 'help semilogx' for the syntax and parameters. Another useful Matlab command might be 'hold on'. It makes it sop that a new plot or semilogx command will not erase whatever has already been plotted. So that way you can plot two graphs in the same axes. To rescind the command type 'hold off'.

Make a nice Bode plot, and also a plot of the step response! Try to draw conclusions from the Bode plot concerning DC gain, cut off frequency, poles, zeroes and stability! Does the step response reflect whether the circuit is stable or unstable (also compare with the next simulation!)?

Then set the bias voltage to

$$V_B = 550\text{mV} \tag{5}$$

repeat the experiment, as well as the analysis and interpretation for this new setting!

Task 2 (8p): In short:

repeat the same experiment doing measurements in the lab with the SF assembled using two transistors from the IC CD4007UBE.

More exhaustive:

for R_S you can use one of the 100kΩ (or similar) resistors in the lab. No need to do the step response. You shall only use pure sine waves from the mfg function generator to analyse the frequency response and do a Bode plot. Use the Matlab functions from 'resources' → 'MatlabScripts':

```
function [ok] = MFG_sine(h,freq,ampl,offs,chan)
function [ok] = MDO_TimebaseScale(h,scale)
function [ok] = MDO_SingleTrigger(h)
function [waveform,time,ok] = MDO_GetWaveform(h,channel)
function [mag,ph] = GainAndPhase(x,y,t,freq)
```

Note that the maximal sine wave frequency of the mfg waveform generator is 20MHz. See https://www.uio.no/studier/emner/matnat/ifi/IN3170/v25/materiale/LabAssignmentGuides.html for more info on the functions.

Supply sine waves to the circuit and record both the signal source signal V_{SIG} and output signal V_O to deduce both gain and phase for a good spectrum of frequencies. Note that by connecting the scope probes you actually add about 15pF of capacitive load, so here you actually have a C_L. On V_{SIG} the influence should be small as the waveform generator is a strong voltage source with a small output resistance. Use a sufficiently small V_{SIG} amplitude close to Vdd such that the output signal is not clipped, but also as large as possible, to get clear sinusoidal waveforms from the oscilloscope. Use a Vdd of 5V. Make a Bode plot. Is the range of the MFG function generator sufficient to reach the cut off frequency? Compare as best you can with the results from the simulation.

To get the magnitude and gain of the transfer function at a particular frequency you can feed the input and output waveforms to the Matlab function GainAndPhase.m that you find under 'resources' → 'matlab scripts' on the course webpage. The header is:

```
[mag,ph] = GainAndPhase(x,y,t,freq)
```

Some Matlab hints:

```
f=10.^(0:0.1:8)
```

makes a vector of logarithmically distributed numbers from 1 to 10'000'000 with a factor $10^{0.1}$ between two consecutive numbers, and

```
for f=10.^(0:0.1:8)
  f
end;
```

cycles through these numbers. Inside the loop f is a scalar, i.e. not a vector.

```
pause(s);
```

pauses execution for 's' seconds. This is sometimes necessary between sourcing a value to an instrument and then taking a measurement!

So your function to go through a frequency spectrum and measure a Bode plot may contain a loop something like that:

```
phase=[];
magnitude=[];
for f=10.^(0:0.1:8)
[ok] = MFG_SetSine(hmfg,f,ampl,offs,chan);
scale=1/f;
% scale sets the time per divisionon the scope and there are 10 divisions.
% This ensures that you have a recording of about 10 cycles to compute the phase
[ok] = MDO_TimebaseScale(hmdo,scale)
pause(2);
[ok] = MDO_SingleTrigger(hmdo);
pause(max(1,25/f)); % need to wait long enough for the MDO buffer to fill
[x,t,ok] = MDO_GetWaveform(hmdo,1);
[y,t,ok] = MDO_GetWaveform(hmdo,2);
[mag,ph] = GainAndPhase(x,y,t,f);
% appending the results to vectors phase and magnitude:
ph
mag
phase=[phase ph];
magnitude=[magnitude mag];
end; %for
```

Note the internal workings of the 'GainAndPhase' function, to understand when it may give you wrong results: If x and t are row vectors representing a sine wave graph x(t)

```
max(x)-min(x)
```

is the maximal peak to peak amplitude. 'GainAndPhase' uses that to determine 'mag'.

```
upcrossings=find((x(2:length(x))>=thresh)&(x(1:(length(x)-1))<thresh))
```

gives you the vector indices where x is about to cross the threshold 'thresh' upwards and thus

```
t(upcrossings)
```

are the times of these crossings. This is used internally in 'GainAndPhase'. However, if x is a noisy sine wave, there may be multiple false hits. The function 'GainAndPhase' uses some clean up algorithm to remove false hits like that, but if the signal is too noisy, it may still fail. So check whether the return values 'mag' and 'ph' make sense by visual inspection of the traces too.