EE140: Lab 5, Project Week 2

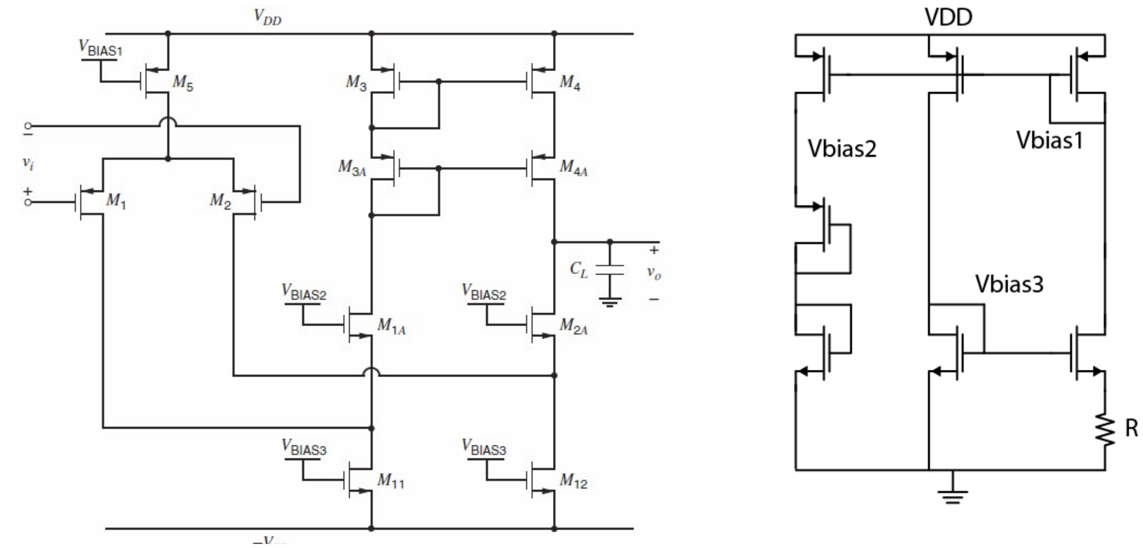
VGA Op-amp

Introduction

For this lab, you will be developing the background and circuits that you will need to get your final project to work. You should do this with your project group (ie, each group collectively makes one set of slides this time). The results should go into your powerpoint design document, and will be presented to your GSI next week at the beginning of lab.

Objective

The goal is to design and simulate a PMOS input folded cascode for your Variable Gain Amplifier (VGA). The figures below show the topology and bias generation circuitry of a PMOS input folded cascode. **You can copy them directly if you like, or use other sources, or design your own from scratch.** In any case, you might want to start with longer channel lengths (e.g. L = 1um) to keep the devices in quadratic mode where they are easier to analyze. There are many ways to bias this circuit, this is only one of many possibilities.



Left: PMOS Input Folded Cascode

Right: Constant-gm Bias for Folded Cascode

Specification	**Requirement**
V_{DD}	1.2 V
Settling Accuracy (f=8)	< 0.4 %
Settling Time (f=8)	< 5 µs
C_L	400 fF
Input Common Mode Range	Includes ground
Temperature	-40 C < T < +85 C

Recommended Design Strategy

1) Calculate the requirements for open loop gain and unity gain bandwidth.

2) Choose device lengths (L = 1 µm is a good place to start) and overdrive voltages.

3) Calculate the transconductance required for the bandwidth and the current given your choice of overdrive voltage.

4) Calculate widths for all transistors. Start with the current mirror for the input diff pair and work your way toward the output. The table below should give you a reasonable place to start with hand calculations using the quadratic model.

	Vth	λ (@L=1µm)	µCox
nmos1v	0.4 V	0.14 1/V	300 $\mu A/V^2$
pmos1v	0.35 V	0.12 1/V	230 $\mu A/V^2$

5) After sizing all transistors run a DC sim and check all the voltages and currents (Don’t forget to apply a DC bias at the input). If the simulated values are very far off from what your hand calculations predicted, STOP! Go back through systematically and look for the discrepancy. If your DC bias is wrong, your amplifier is never going to work as expected.

6) Once your DC bias is working, use the test bench (see below) to check the AC small signal frequency response of your amplifier. If you fail to meet your gain or bandwidth specs, think about what you need to change in order to increase the gain or bandwidth and make those

changes. Keep track of what changes you make and what their effects were (making copies of your schematics with revision numbers is recommended). Continue this strategy until you have met all specs.

7) You'll likely find that one temperature condition is more difficult to meet than the others. It is easiest to focus on getting your amplifier working under this most difficult constraint, then the other temperatures should achieve the specs with ease.

Test Bench

We often use components from analogLib to test our circuits (things like voltage sources, etc) but we want our schematics to only include real devices that we can fabricate (this becomes important for layout). The solution is to create a test wrapper which turns our schematic with real devices into a component with input and output pins. We can then create a new schematic, instantiate the symbol of our circuit, and attach analogLib parts to the pins for testing.

Since we will use test benches to evaluate your final project performance, you will get familiar with the process in this lab. The test bench has already been created for you as an example. The overview of the steps to use the test bench are outlined below, and then explained in detail:

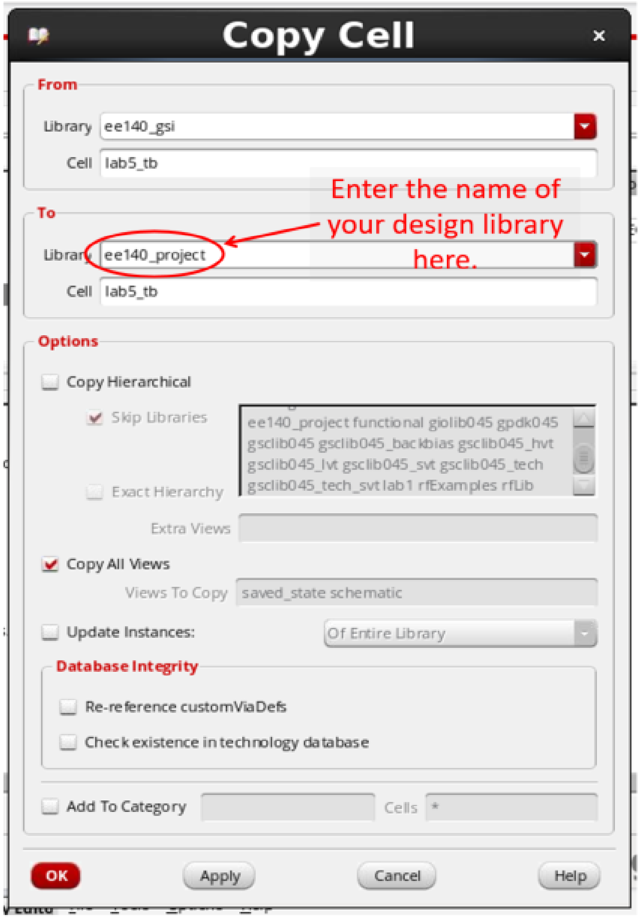
1) Copy over the test bench file, lab5_tb from library:
   ee140_gsi
2) Copy your schematic to the test bench library
3) Load saved state and run simulation

The Detailed Steps:

1) Copy over the test bench file, lab5_tb from library: ee140_gsi

- From within your cadence folder (where you launch cadence from), open cds.lib using vi or your favorite text editor. Make sure you point to ee140_gsi library by looking for following line:

  DEFINE ee140_gsi /home/ff/ee140/sp18/cadence/ee140_gsi
- Once this is ensured, copy "lab5_tb" and "lab5_amp" by right Click --> Copy.
- In the pop-up menu make sure to provide your library name under "To". By default it would be "ee140_gsi" which you need to replace it with name of your design library.
- Follow the remaining instructions as shown in the snapshot below.
- Make sure to do exactly the same steps while copying both "lab5_tb" and "lab5_amp"



Copy Cell

From

Library ee140_gsi

Cell lab5_tb

To

Library ee140_project

Cell lab5_tb

Enter the name of your design library here.

Options

□ Copy Hierarchical

☑ Skip Libraries

ee140_project functional giolib045 gpdk045 gsclib045 gsclib045_backbias gsclib045_hvt gsclib045_lvt gsclib045_svt gsclib045_tech gsclib045_tech_svt lab1 rfExamples rfLib

□ Exact Hierarchy

Extra Views

☑ Copy All Views

Views To Copy saved_state schematic

□ Update Instances: Of Entire Library

Database Integrity

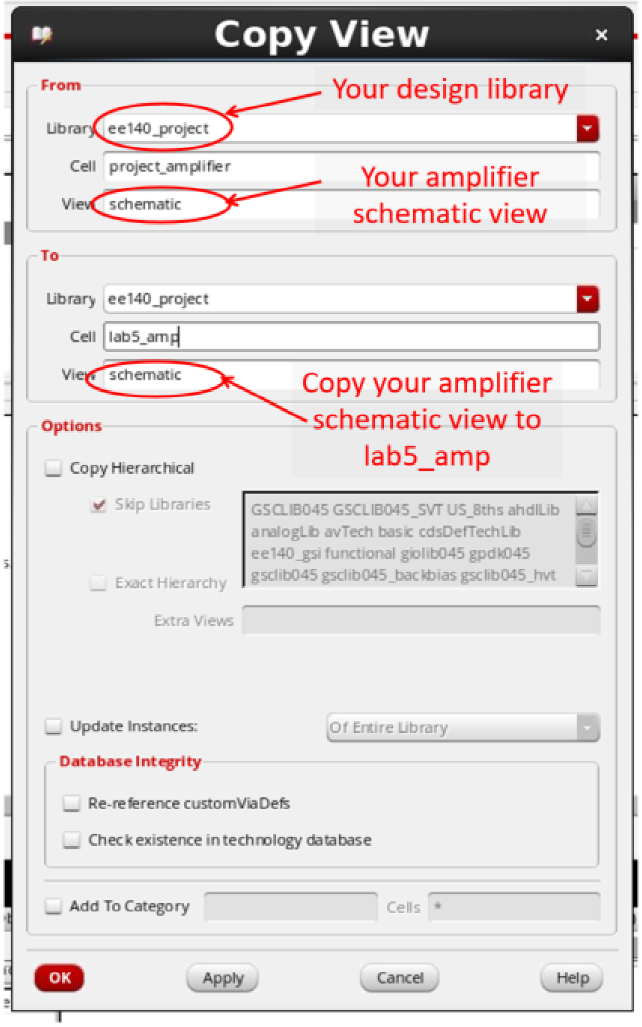
□ Re-reference customViaDefs

□ Check existence in technology database

□ Add To Category Cells *

2) Copy your schematic to the test bench library

- Inside “lab5_tb” you will two cells:
  - “lab5_amp” contains the symbol view only. You should copy your schematic view into this location by right clicking your schematic, choosing copy, and filling out the form as shown below.
  - Your schematic needs to have the same pin names and types (inputOutput) as the symbol.
  - If you get warnings about data.dm, they are safe to ignore and overwrite.



From

Library ee140_project

Cell project_amplifier

View schematic

To

Library ee140_project

Cell lab5_amp

View schematic

Options

□ Copy Hierarchical

☑ Skip Libraries

GSCLIB045 GSCLIB045_SVT US_8ths ahdlLib analogLib avTech basic cdsDefTechLib ee140_gsi functional giolib045 gpdk045 gsclib045 gsclib045_backbias gsclib045_hvt

□ Exact Hierarchy

Extra Views

□ Update Instances: Of Entire Library

Database Integrity

□ Re-reference customViaDefs

□ Check existence in technology database

□ Add To Category Cells *

Your design library

Your amplifier schematic view

Copy your amplifier schematic view to lab5_amp

3) Load saved state and run simulation

- The other cell inside lab5_tb has two views. One is the test bench schematic which instantiates your amplifier symbol along with analogLib parts for testing. The other is a saved ADE state called “saved state”. Saved states allow you to save a particular simulation configuration so that you can easily run it again later.
- Double click on “saved state”. This will open ADE with DC and AC simulations already set up. There are also variables for VDD and the common mode input voltage which default to 1.2 V and 0 V respectively. Click run. If you have set everything up correctly a plot should open showing the magnitude and phase response of your amplifier.

Debugging

It’s a good idea to build things in pieces and test as you go. You could start with the bias network, building up the circuit one leg at a time and verifying that you get the expected gate bias voltages, and then adding in the transistors in the signal path. Or you could build the signal path first, with ideal sources to set biases, and verify that it operates the way that you expect before adding in the bias network.

“Test as you go” means “do a hand analysis to estimate bias point voltages and currents, small-signal model parameters, gain, BW, etc., and then check with simulation to make sure that is what your circuit is doing”. If SPICE and hand analysis don’t match, stop! Go back and figure out if your analysis is wrong, or you built the circuit wrong, or what. A DC plot of Vout vs. V+ will tell you gain and output swing (take a derivative and see how close you can get to the rails before the gain drops off). That same plot with several different values for V- will give you an idea of what your input common mode range is, and how gain and swing vary with input common mode. Estimate phase margin, and then put the amplifier in unity gain feedback and see if the response to a step input looks like you expect.

Deliverables (via PowerPoint)

- Show how you calculated gain and bandwidth requirements.
- Schematic of opamp and bias generation with sizes and currents annotated.
- Bode plot of amplifier frequency response (mag and phase)
- The following table:

Temperature	DC Gain	Unity Gain Bandwidth	Phase Margin
-40 C			
25 C			
85 C			

And a reminder….

A Note on Plots in Cadence

- The default plotting options in Cadence are garbage for presentations.
- The pro-level thing to do is export the data and plot with MATLAB, Excel, etc.
- Cadence plots are only acceptable for submission if at a minimum the following steps have been performed:
  - Graph > Properties > Set background to white > OK
  - Graph > Properties > Graph Options > Font > Fixed [Sony] > Size 18
  - Right click on traces > Width > Extra Thick
- Even with these adjustments, the plots are still not great for presentation. You can experiment with other settings, but they will never look as good as if you had exported.

A Note on Schematics in Cadence

- Like with default plots, schematics in Cadence are poor.
- Drawing your circuits neatly goes a long way in making them understandable.
- If you want really nice schematics, use Adobe Illustrator.
- Otherwise to get decent schematics from Cadence, follow these steps:
- File > Export Image
  - > Background > White
  - > Foreground > Black
  - Bi-color
  - Selected Area > Select > Draw a box around your circuit (It is not click'n'drag)
  - Choose a name, then Save to File
- Annotate sizes and nodes onto schematics by adding labels in post-processing.