Original document by Filip Maksimovic & Mike Lorek, Spring 2015, derived from earlier EE141 lab manuals Revisions for IC6 by David Burnett & Thaibao Phan, Spring 2016 Revisions made by Nandish Mehta to incorporate new 45nm PDK,Spring 2018

EE 140/240A - Full IC Design Flow Tutorial

1. Server Login

For server access, we recommend you use a remote desktop client called x2go [http://wiki.x2go.org/doku.php]. This program uses accelerated X11 tunneling to reduce the latency associated with using applications over SSH. To use it, download the client, and click the 'New Session' button in the top-left corner. For the host name, use one of the hpse (hpse-9 to hpse-15) or c125m (c125m-6 to c125m-14) servers. Eg. hpse-10.eecs.berkeley.edu or c125m-6.eecs.berkeley.edu

Login should be the login of your class account. The desktop environment can be changed with the 'Session Type' option; GNOME seems to work the best with Cadence. Leave everything else at the default setting and type in your credentials to start a remote connection. Note: if you have a Mac, you will need to install XQuartz [http://www.xquartz.org/] to use x2go.

If, for whatever reason, you do not want to use x2go, or x2go is not working, you can directly SSH from a console (use Putty [http://www.putty.org/] for Windows, or terminal in Mac OS X). This is done with the following command:

ssh -X username@hpse-10.eecs.berkeley.edu

hpse-10 can be replaced with any of the other hpse or c125m servers. From there, you will have access to a terminal from which you can proceed with the lab.

2. Cadence Setup & Launch

Run the following command to setup and start virtuoso:

```
cp -r /home/ff/ee140/sp18/cadence .
cd cadence
bash
source cadence_setup
virtuoso &
```

This will open Cadence!

Every time you want to start it, just navigate into the ee140 directory and type virtuoso &. There's no need to run the setup script again. For Cadence documentation,

https://inst.eecs.berkeley.edu/~inst/pub/

Click the "help" button in Cadence, search the web (especially hits on cadence.com and edaboard.com), or ask your GSI(s).

NOTE: if you have more than one session running Cadence on the servers, you will likely experience very slow performance. When closing the remote desktop window, x2go will, by default, suspend your session. Next time you connect using x2go, you should be presented with the option to resume the suspended session. This is a nice feature because you can leave your design environment open. To terminate a session, please exit Cadence by closing the virtuoso console window, and log out of your session from your GNOME/other remote desktop window.

3. Creating a Schematic View

If the Library Manager doesn't open automatically with the virtuoso console window, open it by clicking Tools → Library Manager. You should see a window pop up that looks like this:

🏶 Applications Places System 🥘		Q
•	Library Manager: Directory0/sp18/staff/ee140-laa	cadence
File Edit View Design Manager Help		
Show Categories Show Files Loray Show Categories Show Files Loray Show Categories Show Files Show Files Show Files Show Files Show Files Show Files Show F		
Messages		

If the setup is correct then you would see gpdk45, giolib045, GSCLIB045, and GSCLIB045_SVT. These are the 45nm PDK related libraries which will be extensively used later in this lab and also during project.

In Cadence, there is a relatively straightforward hierarchical organizational structure. Libraries (the furthest left column in the library manager) are collections of Cells, and Cells are collections of Views.

The first thing that you will need is a new library. To do this, go to the library manager and click $File \rightarrow New \rightarrow Library$. A new window will pop up. Type lab1 in the name field and press OK.

				,			
Library —							
Name	lab1						
Directory	e1 4	0/sp16/staff	/ee140-la	/ee140/ <mark>-</mark>	(= E	r (## ##
 cds.lit displa libMar	y.drf nager.log nager.log	.cdslck					
File <u>t</u> ype:	Directo	ries					×
File <u>t</u> ype: Design Ma	Directo	ries					
File <u>t</u> ype: Design Ma	Directo nager ONE	ries					
File <u>type</u> : Design Ma Use N Use N	Directo nager ONE o DM	ries					

At this point, Cadence will prompt you for something called a Technology File. The technology file is a library or group of libraries that all cell views inside of your new library will automatically reference. This basically means that you will not have to import models every time you run a new simulation.

To import the technology file for this class, click the "Attach to an existing techfile" button, press ok, and use the drop down menu to select "gpdk045". This stands for generic process design kit, 45nm. 45nm refers to the minimum feature size available.



Now that you have a library, you can create your first schematic. Select your new "lab1" library in the library manager, and click File \rightarrow New \rightarrow Cell View. When prompted, name the new cell view inverter and select "schematic" from the drop-down menu and "schematic" should auto-fill under View. Click OK, and the schematic editor should open up. Now we can build a virtual representation of our circuit at the transistor level.

• • •	X New File
- File	
Library	lab1 🧧
Cell	inverter
View	schematic
Туре	schematic 🔽
Application	
Open with	Schematics L
🔲 Always use thi	s application for this type of file
Library path file	
c/ee140/sp16/s	taff/ee140-la/ee140/cds.lib
	OK Cancel Help

To instantiate circuit elements in the schematic, press the `I' key. This will bring up the Add Instance window and the Component Browser. To put an NMOS transistor into your schematic, go to Component Browser, change Library to gpdk045, tick the "Flatten" box, and scroll down to nmos1v. This selection should auto-fill in the Add Instance window. You can also type these references into the fields manually. As soon as you do this, you will be prompted with lots of new options. What you are actually doing at this point is instantiating a parameterized cell, or P-cell for short. You can parameterize the transistor gate width, length, and fingers. Your choices will be reflected in the schematic. For now, let's stick with a minimum-sized transistor with a Length of 45nm and a Width of 120nm. Press the "hide" button or hit the Enter key in the Add Instance menu and click down the transistor into your schematic. Press the `Esc' key to stop adding components. If you ever want to change a p-cell's properties after placing it in the schematic, click on it and press the `q' key.

	A	\dd Instance ×
Library	gpdk045	Browse
Cell	nmos1v	· · · · · · · · · · · · · · · · · · ·
View	symbol	
Names	M1	
🗹 Add Wir	e Stubs at: 🔾 all terminals 🧕	registered terminals only
Array	Rows	1 Columns 1
	Rotate	Sideways 🛛 🚭 Upside Down
Model Nan	ne	g45n1svt
Multiplier		1
Length		45n M
Finger Wid	th	120n M
Total Width	ı	120n M
Fingers		1
Folding Thr	reshold	10u M
Diff Cont		✓
Gate Conn	ection	None
Use DFM R	ules	Minimum 🔽
S/D Metal V	Width	60n M
1		Hide Cancel Defaults Help

Following those same instructions, instantiate a PMOS transistor (pmoslv) with a 45nm length and a 120nm width. In addition, instantiate global symbols "vdda" and "gnd" from the analogLib library.

Now that all of our components are in the schematic, you will need to connect them together. Press the `w' key to open up the ``Add Wire" window. This will allow you to make connections between nodes. Simply click on the red squares (the contacts) in the schematic view, and make the connections needed for an inverter circuit. Remember to connect the bulk terminals of the NMOS and PMOS devices to gnd and vdda respectively. You can use `s' to snap the wire to the nearest contact, indicated by a diamond on the schematic.

To navigate the schematic, use the arrow keys on the keyboard to pan around. You can zoom in/out by clicking the magnifying glass buttons on the toolbar, or with the hotkeys control-Z/shift-Z. You can also zoom to a specific area by click-dragging a square with the right mouse button around the region of interest. If the circuit ever disappears, or if you want to take a look at the whole circuit, press the `f' key. The last thing that you need is two pins to indicate the input and output of your inverter. This way you cell view can interface with higher level schematics (more on this later). Press the 'p' key to open up the "Add Pins" window, type the names of your pins with spaces between them (I used input and output here, you can use whatever you like). Make sure that you select "InputOutput" in the direction drop-down menu, click "Hide", place your pins in the schematic, and wire them up.

• • •	X	Add Pin	
Pin Names	input output		
Direction	inputOutput 🔽	Bus Expansion	🖲 off 🔾 on
Usage	schematic 🔽	Placement	🖲 single 🔾 multiple
Signal Type	signal 🧧		
Attach Net Expres	sion: 💿 No 🔾	Yes	
Property Name			
Default Net Name			
Font Height	0.0625	Font Style	stick
🕼 🕹 Rotate	▲ Sideways		Show Sensitivity >>
		Hide Canc	el Defaults Help

If you followed the instructions carefully, your schematic should look something like this:



When it does, press the "Check and Save" button (it looks like a box with a check mark). Leave the schematic editor open for now; we'll need it for the later portions of the lab.

One more quick thing to note before you create a symbol. Sometimes it is useful to label nodes instead of directly connecting them with a wire. To connect two nodes, just label them with the same name! To create a label, press the `L' key, type in the desired node name, and click on the wire in the schematic.

4. Creating a Symbol View

Now that your schematic is finished, we need create a symbol for it. Click Create \rightarrow Cellview \rightarrow From Cellview

A pop to create symbol appears. After you press OK, a new pop-up title "Symbol Generation Options" shows up. Press OK here too.



Notice that vdda and gnd do not show up here. As mentioned earlier, those are global pins that are shared throughout the design automatically.

After you press OK, you will have the opportunity to place the missing pins in the symbol editor. Click to place the pins, rotating by clicking the right mouse button

if needed. Note that the red dot is the actual point where a wire will connect, so put it towards the edge of the symbol. Draw various shapes using the shape buttons at the top to make your symbol look like an actual inverter. Save your symbol view when finished.



5. Circuit Simulation with ADE

Before you simulate anything, you will need to build a test bench for your inverter. To do this, create a schematic called "inverter_tb" in the lab1 library. Once you do that, instantiate a cell called "vpulse" from the analogLib library. When you instantiate the voltage source, enter the string "Vin" for the "DC Voltage" property. Assigning parameters to variables allows you to change them easily and sweep them in the simulator. Set the two vpulse voltage levels to 0V and 1V, and set the rise and fall times to 10ps. The pulse width should be 10ns and the period should be 20ns. Now, instantiate the inverter symbol that you created only moments ago. Don't forget to connect vdda to a 1V supply! The "vdc" source in the analogLib library should be used here. Finally, connect all the circuit components together properly with wires. You should also label the input and output nets to easily identify them when viewing the simulation results (again, press 'l'). Your testbench schematic should look something like this:



Frequency name for 1/per.	
Noise file name	
Number of noise/freq paim	0
DC voltage	Vin V
AC magnitude	
AC phase	
XF magnitude	
PAC magnitude	
PAC phase	
Voltage 1	0 V
Voltage 2	1 V
Period	20n s
Delay time	
Rise time	10p s
Fall time	10p s
Pulse width	10n s

Hit check and save like before. You will get a few warnings that you can safely ignore.

Now you can finally set up the simulation for your inverter. Click Launch \rightarrow ADE L to open up the legendary Virtuoso Analog Design Environment (ADE). This tool is a little bit obtuse but extremely powerful. Let's start with two simple simulations: a dc simulation to determine our inverter's voltage transfer characteristic, and a transient simulation to determine its propagation delay.

Let's first begin by setting the design variables. Click on Variables \rightarrow Edit to open up the Design Variable Editor. Then click the "Copy From" button at the bottom. Remember how you set the DC voltage of the vpulse source to a variable? Here is where you can use it in your simulation. Set the default value to 1V so that your variable window looks like this:

🔳 Editing 🛙	Design Variables Virtuos	o® Ana	log Desi	gn Environm(×	
	Selected Variable		Design Variables		
	11.	- Vin	Nane	Yalue	
Name	V1n	1 ***			
Value (Expr)	1				
Add Dele Next Cle	ete Change Find				
Cellview Var.	iable Copy From Copy To				
	OK Cancel Apply	Apply	∣& Run Si	mulation (Help)	

Now we can set up the simulation. Click on Analyses \rightarrow Choose to open the analysis window. Select "dc", set the sweep variable to "Design Variable", and use "vin" as the variable name. Sweep vin from 0V to 1V with a step size of 10mV. Click OK. Follow the same procedure to create a transient simulation with a duration of 100ns.

DC Analysis	Analysis 🖲 tran 🔾 dc 🔾 ac 📿 noise
Save DC Operating Point 🔛	🔾 xf 🛛 Sens 🔾 dcmatch 🔾 stb
Hysteresis Sweep 📃	🔾 pz 🛛 Sp 🔾 envlp 🔾 pss
	🔾 pac 🛛 🔾 pstb 🔾 pnoise 🔾 pxf
Sweep Variable	🔾 psp 🛛 dpss 🔾 dpac 💭 dpnoise
	🔾 qpxf 🛛 qpsp 🔾 hb 💭 hbac
Uniphie Name Vin	🔾 hbnoise
Design Variable Variable Name Vin Component Parameter Select Design Variable	Transient Analysis
Model Parameter	Stop Time 100n
	Accuracy Defaults (errpreset)
Sweep Range	🗹 conservative 🔛 moderate 🔛 liberal
• Start-Stop Start 0 Stop 1	
Center-Span	🔲 Transient Noise
Sweep Type	
Step Size	🔛 Dynamic Parameter
Ullear 🖉 🖉 Number of Steps	Enabled 🕑 Options

To make the simulator automatically plot the desired waveforms, click <code>Outputs</code> \rightarrow To Be Plotted \rightarrow Select on Schematic, and click on the input and output nodes that you labeled earlier. Remember to press <code>`Esc'</code> after you are done selecting the wires. Afterwards, your ADE window should look like this:

🖀 🛛 Virtuoso® Analog Design Environmer	nt (1) - lab1 inverter_tb schematic 🛛 💶 🗙
Launch Session Setup Analyses Variables Out	tputs <u>S</u> imulation <u>R</u> esults <u>I</u> ools <u>H</u> elp cādence
🎼 🔊 🧊 27 🛛 👌 🎾 🖆 🗹 🗁	
Design Variables Analyses	? 🗗 🗙 🚃
Name Value I do	inable Arguments Orconstant
1 Vin 1 latran	0 1 10m Einear Step Size Start Stop
	¦-⇔→
	×
	0
Outputs	? • × 🔾
Name/S	ignal/Expr Value Plot Save Save Options
o output	
	Outer Replace Replace
> Select on Schematic Outputs to : Plot after	SIMULATION Plotting motheprace
mouse L:	M: R:
4(6) Choose Analyses Status: Selecting out	outs to be plotted T=27 C Simulator:s

To run the simulation, click the green "play" icon on the right side. Close the "What's New" popup box and the simulation will run. The input and output waveforms should be plotted automatically.



From the DC sweep (shown on the right side, above), find the gain by estimating the slope of the curve where the output is at 0.5 V. One option is to go to the "Markers" menu and placing two markers, manually calculating the slope.



The other option is to use the calculator under the "Tools" menu. Using the "wave" selection option, click the DC response on the graph. You should have a wordy expression that represents the output voltage. To calculate the derivative, use the expression as the argument for the function "deriv". You can plot the result in a new window and evaluate the gain from there. Inverters are normally associated with digital circuits, but they can be used as amplifiers as well.



Using these two methods, record the DC gain for the lab report.

The propagation delay of the inverter is defined as the time between the input crossing Vdd/2 and the output transitioning to Vdd/2. Use the cursors on the transient simulation to measure the high-to-low and low-to-high propagation delays. **Provide a plot and record these values for the lab report.**

6. Creating a Layout View

Go back to the library manager and create a new inverter cell view as you did previously. This time, select the "layout" option as the tool. This will create an empty layout editing view.

First, press the `E' key to get to the display options menu. We will need to change the snap spacing so we can move our cursor with sufficient resolution. Make sure your grid matches 5nm requirement. A layout that does not comply grid will not pass DRC checks and will require re-layout. You can change display levels to stop at 10 so that we don't run into any problems.

Grid Controls		
Type 🔾 none 🖲 dots 🔾 lines		
🔲 Dim Major Dots		
Minor Spacing 1		
Major Spacing 5	Array Display	Display Levels
X Snap Spacing 0.005	• Full	Start 0
Y Snap Spacing 0.005	Source	Stop 10

First, we'll draw rulers to specify the outline of the inverter. Press the k' key to draw rulers, and make a 3.0 x 1.6 micron rectangle.

Then, we'll make metal lines for the power supply. To specify the material, we can go to the "Layers" toolbox on the left side. Find "Metall" and click on the layer whose purpose is drawing. Now press the 'p' key to create a path. A "create path" dialog box should pop up; if it doesn't, hit 'F3' on your keyboard. Change the path width to 0.6 microns. Now draw the paths so that they are flush with the top and bottom of your rulers. Click once to start a path, double click to end a path.

Layers	? • ×		0 0,5 3	1 1.6
AV NV AS	NS			
Metal1 drawing	-		2.5	2,5
🍸 All Valid Layer	-s 🔽 🐠		-	
🔲 Used Layers Onl	у		2 -	
Q Metal1	× •			
△ Layer Purpose	VS		1.5 -	1.5 -
■ Me slot	<u>v</u> <u>v</u>			
Me label	✓ ✓			
Me fill	<u> </u>	Path Options Sin		
Me pin	 ✓ ✓ 			0.9
🛛 Me net	✓ ✓	Width 0.6	ā 0.5	1.6

Press 'e' to change the display options, and set the x and y snaps to 0.005 microns. Now press the 'i' key to instantiate some transistors. Find the layout view of the NMOS and PMOS you drew in schematic, set the length and width to match, and place the devices in your layout.

This may be the first time you've seen a MOSFET laid out! Figure out what the different layers mean. Where are the source, gate, and drain? Where is the channel?

Draw 0.06 micron Metall traces to connect the drains of the transistors together, and to connect the sources to the power rails. Draw a 0.045 micron "Poly" trace to connect the transistor gates together.

Polysilicon is quite resistive; we'd want to switch to metal traces as soon as possible. To do this, there needs to be a via between polysilicon and metal lines. To do this, go to the menu and click $Create \rightarrow Via$. Make sure the "via definition" is "M1_PO". Rotate it and place it onto the layout. After you've finished the steps, your layout should look like this:



Remember that MOSFETs are 4-terminal devices. We've connected the source, drain, and gates – what happens to the bulk terminal?

In an NMOS, the bulk is the substrate, which is implicitly shown in black. A PMOS must be placed in an N-well to operate properly. The N-well is automatically placed with the PMOS, but is too small to make any connections to.

To fix this, we need to draw a bigger N-well. Find the "Nwell" drawing layer and press the 'r' key to draw a rectangle that includes the PMOS and the top metal line.

To make the bulk connections, we must connect the top power rail to the N-well, and the bottom power rail to the substrate. Unfortunately, this design library does not provide a utility for connections to the substrate, so we'll have to do it manually. In a MOSFET, a connection to the substrate requires four things:

- 1) Initial thinning down of the oxide (layer "Oxide").
- 2) Cutting a hole through the oxide (layer "Cont").
- 3) Doping the substrate to reduce resistance (layer "Nimp"/"Pimp").
- 4) Depositing metal for a connection (layer "Metal1").



Easiest way to make contact to the terminal is to use N-Tap and P-Tap. To do so Create -> Multipartpath and immediately press F3 without making a mouse click or any other action. Select N-Tap or P-Tap from the pop-up.

-	Create M	ultipart	: Pa	ath	×
Template Name	N-Тар	MPPTemplat	e	N-Tap	
		Technology Li	ibrary	gpdk045	
	🗹 Choppable				
Width	0.18	End Type	trunca	ite 🔽	
Offset	0	Begin Extension	0		
Justification	center 🔽	End Extension	0		
Connectivity	None 🔽				
	Load Template Sa	ve Template	Su	ubpart	
			<u>H</u> ide	<u>C</u> ancel	Help

Finally, we draw 0.12 micron metal paths extending the inverter input and output connections to the far left and far right of the cell boundary. If you wish, you may delete the rulers used to specify the distances between contacts for the substrate connections.



7. Design Rule Checking (DRC)

The DRC checker verifies that your drawn layers obey all of the design rules. These design rules are provided by the IC foundry to ensure that the IC devices perform to specification. The design rules document are uploaded on bcourses under gPDK45nm page.

Before running DRC create a symbolic link:

ln -s /home/ff/ee140/sp18/gpdk gpdk

To run DRC, go to the menu bar and click Verify → DRC. Uncheck the "Rules Library" box. Enter the following for "Rules File":

	Run	Assura L	DRC	×
Layout Design Source	DRI Compare	e two layouts	Generate LvL Comp	pare Rules
Library 1ab1	Cell inverte	r View	layout	Browse
Save Extracted View	View Name drc_ex	tracted		
Area To Be Checked	Full			
Run Name	Run Director	y ./DRC_LVS_sim	n	
Run Location	local			
View Rules Files	Z Technology	undefined-	Rule Set default	
Rules File	ee140-1aa/gpdk/gpdk045	/assura/assuraDRC.r	ul View	Edit Reload
Switch Names			Se	t Switches
RSF Indude				Vie w Edit
Variable	Value Defa	ult Descriptio	on	
None 🔽				
View avParameters	Modify avPa	rameters	1 avParameter i	s set.
View Additional Functio	ins 🗌		No additional functions	are set.
Enable limitDrcCheck				
-	OK Cancel Apply	Defaults	d State Save State	View DSE Helm

Now click OK to run the DRC checker! The results should be displayed in the Virtuoso window. You'll probably have some problems – the errors may be really difficult to understand. Try to decode what needs to be fixed (usually has to do with spacing between components and paths not flush with each other). Ask your GSI if you're stuck. When there are no errors, your Virtuoso window should look like this:



8. Adding Pins to the Layout View

Remember that, in the schematic and symbol, we had pins for the input and output of the inverter. Select the "Metal1" layer, then on the menu bar, click Create \rightarrow Pin. Type "input output" for the terminal names.

	Create Shape Pin x
Connectivity	🖲 strong 🔾 weak
Terminal Names	input output Physical Only
🔲 Keep First N	ame X Pitch 0 Y Pitch 0
🔲 Display Term	inal Name splay Terminal Name Option.
🔛 Create as RO	D Object
Name	rect0
Mode	● rectangle 🥥 dot 🔾 polygon 🔾 circle 🔾 auto pin
I/O Type	◯ input 🔾 output .● inputOutput ◯ switch
	🔾 jumper 🔾 unused 🔾 tristate
Snap Mode	orthogonal 🔽
Access Direction	🗹 Top 🗹 Bottom 🗹 Left 🗹 Right
	🗹 Any 🛄 None
	Hide Cancel Help

Now you can draw two metal squares onto the input and output of the inverter.



We also need to label the pins and global nodes. Press the `l' key to create labels. In the text box, type in "input output vdda! gnd!", change the height to 0.1, and change the layer to Metall. Now you can label the terminals.

	Create Label	×
Mode 💌 Manı	ual 🔾 Auto	
Label (Pattern)	vdda! gnd!	
Label Layer/Pur	pose	
🔾 Use currenter	try layer	
🔾 Use same laye	r as shape, select purpose drawing	
Auto (Using st.	ampLabelLayers section from technology file)	
Select layer	Metal1 label	
Label Options		
Font stick	Height 0.1 Justification centerCenter	
⊻ Drafting	🗹 Attach	
Overbar	🗹 Expand	
⊻ Keep Label		
Scan Line		
Label Selected	I Objects 🗹 Auto Zoom Out	
Auto Step		
Step Custon	1 Х Ө Ү Ө	
Snap Mode	orthogonal 🔽	
E R	otate 🖉 🖉 Sideways 🖉 🧲 Upside Down	
	Hide Cancel H	elp



9. Layout versus Schematic Verification

Now, we need to check to see whether the transistors we placed here actually match up with the transistors we placed in schematic. To do this, we run a layout vs. schematic (LVS) checker.

From the Virtuoso layout menu bar, click Assura → Technology. In the "Assura Technology File" dialog box, enter the following:

/home/ff/ee140/sp18/gpdk/gpdk045/assura_tech.lib

Click OK. In the Virtuoso menu bar, click Assura → Run LVS. Switch the "Technology" option to "gpdk045". Your "Run Assura LVS" menu should look like this:

	DRI	Use Existing Netlist		Netlisting O	ptions
Library lab1	Cell	inverter	View	schematic	Browse
Layout Design Source	DRI	Use Existing Extracted	d Netlist		
Library lab1	Cell	inverter	View	layout	Browse
Run Name	R	un Directory ./DRC.	_LVS_sim		
Run Location	local 🔽				
View Rules Files Extract Rules Compare Rules	✓ Technolog /home/cc/ee140 sp18/staff/ee14	y -undefined-)/sp18/staff/ee140-1a 40-1aa/gpdk/gpdk045/	aa/gpdk/ assura/c	Rule Set defau gpd View. ompare.rul	Edit Reload
Switch Names Binding File(s)					Set Switches
RSF Include	3/staff/ee140-]	Laa/gpdk/gpdk045/ass	ura/LVSi	nclude.rsf	View Edit
Varia ble	Value	Default	Desa	iption	
View avParameters		lodify avParameters		1 avPara meter is	set
View avCompareRules		odify avCompareRules		No avCompare n	ules are set.
View Additional Function	s 🗌			No additional functio	ns are set.

Click OK to run LVS. If your layout matches your schematic, you will see the following message:



If you have any LVS errors, please correct them by making the appropriate adjustments to your layout. Remember to run DRC check again to verify that the update doesn't violate any design rules.

At this point, your layout now passes DRC and LVS – a huge milestone in the IC design process. Next, we will examine how the physical placement of transistors (layout) impacts circuit performance.

10. Parasitic Extraction with Assura

This tool allows us to extract parasitic resistances and capacitances from the layout view. From the menu, click Assura → Open Run, and make sure "Run Name" lists the current cell you are working on ("inverter"). Click OK.

Fill-up the form as below and press. Make sure you are pointing to you local *gpdk* and not to the one in the staff folder. You do not have permission to access it.

Now, click <code>Assura</code> \rightarrow <code>Quantus</code> <code>QRC</code>. If you don;t see this tab make sure you <code>cadence_setup</code> described on page-1. Adjust the settings in the setup, extraction, and netlisting tabs to match the screenshots below.

		Open Run	×
Run Directory	./DRC_LVS_sim		-
Run Name	inverter 🔽	LVS De bug Environment LVS Error Report	ERC Browser
			<u>Cancel H</u> elp

	Assura T	ech	nology Lib Select (on hpse-10.EECS.Bei	rkeley.EDU)	×
Assura	a Technology	File	/home/ff/ee140/sp18/gpdk/gpdk045/assura_tech.lib		lit
			OK Cancel App	ly Defaults H	lelp

IP Extraction Filtering N	Netlisting Run Details	Substrate			
Technology gpdk045_av		RuleSet rcx_typical			
p2lvsSet		UseMultRuleSets)	
Setup Dir /home/ff/	/ee140/sp18/gpdk/gpdł	k045/assura//qr	c/typica)	
Include Command File	e140/sp18/gpdk/gpdk04	45/assura/LVSinclu	ude.rsf	View Edit	
Rule Command File Includ	e			View Edit	
Tech Cmd File				View Edit	≣
Layer Setup File				View Edit	
LPE Config File				View Edit	
🔲 Library Cell Mapping File				View Edit	
Library Definitions File	/home/ff/ee140/sp18	/cadence/cds.lib		View Edit	
Probe Text File				View Edit	
Output Extracted View	Lib lab1	Cell inverter	View av.	_extracted	
Enable CellView Check					
Parasitic Res Component	presistor		Prop Id	r	
Parasitic Cap Component	pcapacitor		Prop Id	c	
Parasitic Ind Component	pinductor		Prop Id	1	
Parasitic M Component	pmind		Prop Id	k	
Inductance L1 Prop Id	ind1	Inductance L2	Prop Id	ind2	

Quantus QRC (Assura) Parasitic Extraction Run Form (on hpse	-10.E ×
Setup Extraction Filtering Netlisting Run Details Substrate	Â
Extraction Type RC 🔽 Name Space Schematic Names 🔽	
Max fracture length infinite microns Temperature 25.0 C Edit	
Max fracture length by layer Edit	
From File	
Cap Coupling Mode Decoupled Ref Node gnd!	=
Mult Factor 1.0 Extend Checking Distance default Diffusion Equation R	1
Ladder Network	
Select User Region View Edit	
Net Selection Type Full Chip All Nets FS Extraction Mode NONE	Π.
Layer Setup Customization 🔲 Edit Quantus FS mode	
Field Solver High	
Resistance Mesh	
Automatic Mesh Size R Mesh Accuracy Moderate Options	
Auto Mesh Size Scale Factor 1.0 Error Tolerance 1.0	%
Select R Mesh User Region View Edit	
Non-Manhattan Resistance Accuracy default	
From File	
Cancel Apply Defaults Load State Save State View Command File	Help

Make sure you have entered gnd! as the REF node. Else the extraction will result in dangling nodes.

🖬 Quantus QRC (Ass	ura) Parasiti	c Extraction Run For	n (on hpse-	10.E ×
Setup Extraction Filtering	Netlisting Run Deta	ails Substrate		\wedge
Design Capacitor Models	Do Not Include			
Parasitic Capacitor Models	Do Not Include			
Design Resistor Models	Do Not Include			
Parasitic Resistor Models	Include As Comme	nt 🔽		
Netlist With Names From	Schematic 🕞	EM Analysis		
Enable Metal Fill	virtual ·	Save Fill Shapes		= =
Sub Node Character	#	Bus Bit	[]	
Input Hierarchy Delimiter	/	Device Finger Delimiter	@	
Output Hierarchy Delimiter	/	Force Globals		
Import Globals		Parasitic Resistance Length		
Parasitic Resistance Width	Off			
Parasitic Resistance Model B	y Sub Conductor			
Parasitic Resistance Temper	ature Coefficient	Do Not Include		
XY Coordinates	R 🗌 C 🛄	r C D M	Q 🗌 🛛 X 🛄	
Ignore Vias	Layers	Nets		
Auto Substrate Stamping Off				
Add Explicit Vias				
Via Effect Off				
OK Cancel App	ly Defaults	Load State Save State Vie	ew Command File	Help

Click OK to run the parasitic extractor. A successful extraction should give you this message:



Return to the library manager, and your inverter cell should have a new view named "av_extracted". You can zoom in and see all the little parasitics!

The extracted view looks something like below:



Once you zoom-in you will find tiny capacitors as below:



11. Parasitic Extracted Circuit Simulation

We would like to simulate the extracted layout. First, create a new cell view for your "inverter_tb" cell. This time, select the "config" tool, making sure the option says "Hierarchy Editor".

In the "New Configuration" window, click the "Use Template" button and select "spectre" as the template. Change the "Top Cell View" field from "myView" to "schematic". Now click OK. In the resultant window's menu bar, click View \rightarrow Update. Your windows should look like the screenshots below.

		New Configuration	
Top Cell			
Libraru*	lab1		
Cell:	invert	er th	
View:	schema	tic	
Global Bin	dings		
Library Li	st:	myLib	
View List:		_sch cmos.sch schematic veriloga ahdl	
Stop List:		spectre	
Constraint	List:		
Descriptio	n		
Default f Note: Plea View fie.	templat ase rem lds wit	e for spectre ember to replace Top Cell Library, Cel h the actual names used by your design	ll, and 1.
	OK	Cancel Use Template (Help)	

Now, right click on the inverter cell and select Set Cell View \rightarrow av_extracted. Update the hierarchy again, then save the configuration.

	7 40- 9 (C (
Lell		? 8 X	Global Bindings	7
orary: lab1			Library List:	nyLib
ll: inverte	r_tb		View List:	veriloga ahdl i
ew: schemat	ic			
			Stop List:	spectre
pen Edit			Constraint List	:
Table View	Tree View			
ell Bindings				
Library	Cell	View Found	View To Use	Inherited View Lis
analogLib	vdc	spectre		spectre cmos
analogLib	vpulse	spectre		spectre cmos
spdk045	nmos1v	spectre		spectre cNos
(pdk))45	prosiv	spectre		spectre chos
lab1	inverter th	schematic		spectre cmos
Namespace CDB#	a Filters: OFF Virtuoso® Hierard	:hy Editor: (lab1 inv	erter_tb config)	_ □ X
Namespace CDB/	A Filters: OFF Virtuoso® Hierard <u>Plugins H</u> elp	:hy Editor: (lab1 inv	erter_tb config)	_□× cādence
Namespace CDB > ile Edit Yiew	A Filters: OFF Virtuoso® Hierard Plugins Help 2 &	chy Editor: (lab1 inv	erter_tb config)	– □ × cādence
Namespace: CDBs > ile Edit View p Cell	A Filters: OFF Virtuoso® Hierard Plugins Help 2 &	:hy Editor: (lab1 inv) - 도 (lab1 inv (기용 곳)	erter_tb config) Global Bindings	– □ × cādence ? 5×
Namespace CDB/) Filters: OFF Virtuoso® Hierard Plugins Help) &	chy Editor: (lab1 inv	erter_tb config) Global Bindings Library List;	_ □ × cādence ?ē× ₩JLib
Namespace CDBs	Filters: OFF Virtuoso® Hierard Plugins Help 2 &	chy Editor: (lab1 inv) C C C C X	erter_tb config) Global Bindings Library List:	_ □ × cādence ? 8 × mgLib
Namespace CDBs	Filters: OFF Virtuoso@ Hierard Plugins Help 2	:hy Editor: (lab1 inv) C (lab1 inv)	erter_tb config) Global Bindings Library List: View List:	_ □ × cādence ?#/× mgLib weriloga ahdl
Namespace CDBs	A Filters: OFF Virtuoso@ Hieraro Plugins Help D &	chy Editor: (lab1 inv) 🖻 👔 🚱 🥹 (2)6, X,	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List:	_ □ X cādence ?ē× wgLib yeriloga ahdl spectre
Namespace: CDB ile Edit View p Cell Library: labi Cell: invert View: schema Open Edit.	A Filters: OFF Virtuoso® Hierard Plugins Help 2 &	chy Editor: (lab1 inv	erter_tb config) <u>Global Bindings</u> Library List: Vieu List: Stop List: Constraint List:	_ □ X cādence ?@x mgLib spectre
Namespace CDBs ile Edit Yiew p Cell Library: labi Cell: invert View: schema Open Edit.	A Filters: OFF Virtuoso@ Hierard Plugins Help 2	chy Editor: (lab1 inv) C C C C C C C C C C C C C C C C C C C	Global Bindings Library List: View List: Stop List: Constraint List:	- C X cādence ? 5 X mgLib spectre
Namespace CDB/ ile Edit Yiew p Cell Library: lab1 Cell: invert View: scheme Open Edit. Table View	 Filters: OFF Virtuoso@ Hierard Plugins Help Plugins Help P- % (er_tb) tic Tree View 	thy Editor: (lab1 inv	Global Bindings Library List: View List: Stop List: Constraint List:	- C X cādence ? # X WyLib weriloga ahdl spectre
Namespace CDB/	A Filters: OFF Virtuoso@ Hieraro Plugins Help D D O O O O cer_tb stic Tree View	chy Editor: (lab1 inv)	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List:	<pre>_ C × cādence ?@ × myLib spectre</pre>
Namespace CDBs ile Edit View p Cell Library: labi Cell: invert View: schems Coon Edit. Table View Cell Bindings	 Filters: DFF Virtuoso@ Hierarc Plugins Help @	chy Editor: (lab1 inv) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use	- C X cādence ? @ x myLib spectre [inherited View Lis]
Namespace CDBs ile Edit View p Cell Library: labi Cell: invert View: schems Coon Edit. Table View Cell Bindings Library analogLib	 Filters: OFF Virtuoso@ Hierarc Plugins Help Q- > Q- ><!--</td--><td>thy Editor: (lab1 inv</td><td>erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use</td><td>- C X cādence ? @ x mgLib spectre inherited View List spectre cmos</td>	thy Editor: (lab1 inv	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use	- C X cādence ? @ x mgLib spectre inherited View List spectre cmos
Namespace CDBs ile Edit View p Cell Library: labi Cell: invert View: schema Coon Edit. Table View Cell Bindings Library analogLib analogLib	 Filters: DFF Virtuoso@ Hierard Plugins Help @- % @- %<!--</td--><td>thy Editor: (lab1 inv Crexit</td><td>erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use</td><td>_ C X cādence ? 5 × myLib spectre inherited View Lis spectre cnos spectre cnos</td>	thy Editor: (lab1 inv Crexit	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use	_ C X cādence ? 5 × myLib spectre inherited View Lis spectre cnos spectre cnos
Namespace CDBs ile Edit View p Cell Library: labi Cell: invert View: schema Coon Edit. Table View Cell Bindings Library analogLib analogLib analogLib	 Filters: DFF Virtuoso@ Hierard Plugins Help Q- > Q- ><!--</td--><td>thy Editor: (lab1 inv</td><td>erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use</td><td>_ C X cādence ? 5 × mgLib spectre spectre cnos spectre cnos spectre cnos</td>	thy Editor: (lab1 inv	erter_tb config) <u>Global Bindings</u> Library List: View List: Stop List: Constraint List: View To Use	_ C X cādence ? 5 × mgLib spectre spectre cnos spectre cnos spectre cnos
Namespace CDBs ile Edit View op Cell Library: lab1 Cell: invert View: schema Con Edit. Table View Cell Bindings Library analogLib analogLib analogLib analogLib gpdk045	 Filters: DFF Virtuoso@ Hierard Plugins Help Q-	thy Editor: (lab1 inv	erter_tb config) Global Bindings Library List: View List: Stop List: Constraint List: View To Use	_ C X cādence ? 8 × mgLib meriloga ahdl spectre spectre cmos spectre cmos spectre cmos spectre cmos
Namespace CDBs ile Edit View op Cell Library: lab1 Cell: invert View: schema Coor Edit. Table View Cell Bindings Library analogLib analogLib analogLib gpdk045 gpdk045	 Filters: OFF Virtuoso Hierard Plugins Help Image: State of the state Virtuoso Tree View Cell pcapacitor presistor vdc vpulse nmoslv pmoslv 	thy Editor: (lab1 inv	erter_tb config) Global Bindings Library List: View List: Stop List: Constraint List: View To Use	- C X cādence ? 8 × mgLib spectre spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos
Namespace CDBs ile Edit View op Cell Library: lab1 Cell: invert View: schema Con Edit. Table View Cell Bindings Library analogLib analogLib analogLib analogLib godk045 godk045 lab1	 Filters: OFF Virtuoso@ Hierard Elugins Help Plugins Help Plugins Plugins Help Plugi	thy Editor: (lab1 inv	erter_tb config) Global Bindings Library List: View List: Stop List: Constraint List: View To Use View To Use	Inherited View List spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos
Namespace CDBs ile Edit View op Cell Library: lab1 Cell: invert View: schema Coor Edit. Table View Cell Bindings Library analogLib analogLib analogLib analogLib gpdk045 gpdk045 lab1 lab1	 Filters: OFF Virtuoso Hierard Elugins Help Image: Second Secon	thy Editor: (lab1 inv	erter_tb config) Global Bindings Library List: View List: Stop List: Constraint List: View To Use View To Use	- C X cādence ? 8 × mgLib spectre spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos
Namespace CDBs ile Edit View op Cell Library: lab1 Cell: invert View: schema Coor Edit. Table View Cell Bindings Library analogLib analogLib analogLib gpdk045 gpdk045 lab1 lab1	 Filters: OFF Virtuoso Hierard Elugins Help Image: Ser_tb Ser_tb Ser_tb Tree View Cell pcapacitor presistor vdc vpulse nmoslv pmoslv inverter inverter_tb 	thy Editor: (lab1 inv	erter_tb config) Global Bindings Library List: View List: Stop List: Constraint List: View To Use View To Use	- C X cādence ? 8 × mgLib spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos spectre cmos

Open up the Library Manager window and double click the inverter_tb "config" cell view. This prompts you to open the hierarchy configuration (which is what we just edited) and/or the top cell view. If you kept the configuration window open, just open the top cell view.

You should now see your "inverter_tb" schematic. Click on the inverter symbol, then press the 'x' key to descend into the inverter cell. Your regular layout has now been replaced with the extracted one. Press the 'b' key to ascend one level back to the top level schematic.

Launch "ADE L" again. Click Simulation \rightarrow Netlist \rightarrow Create. The netlist should be simulated with all the parasitics included. Look at the parasitics to get a feel for their values. Save a copy of the netlist (either text or screenshot) to include in your lab report.

Configure ADE to run the DC sweep and transient simulation again. Find the DC gain and the propagation delays. Are the simulation results different now?

12. Deliverables

- A. Schematic simulation
 - a. DC Gain
 - b. High-Low Propagation Delay
 - c. Low-High Propagation Delay
- B. Screenshot of finished inverter layout
- C. Netlist with extracted parasitics
- D. Extracted Simulation
 - a. DC Gain
 - b. High-Low Propagation Delay
 - c. Low-High Propagation Delay