ECEN 474/704 Lab 1: Introduction to Cadence and MOS Device Characterization

Access to ECE Linux System

Cadence is the integrated circuit (IC) design tool that will be used for simulation and layout preparation of CMOS circuits throughout this course, which is available on the ECE Linux system. Assuming you already have your TAMU NetID and password (otherwise visit https://gateway.tamu.edu), and you are enrolled in Duo NetID two-factor authentication (otherwise visit https://duo.tamu.edu), follow the steps below to prepare your Linux home directory:

- Connect to TAMU VPN at https://connect.tamu.edu
- Complete your Linux home-directory setup following the instructions at http://bit.ly/ecenhomedoc

Once your home directory setup is completed, you can access ECE Linux system with your **TAMU NetID and password** using any of the following four methods (**"Host Name"** can be substituted with **olympus.ece.tamu.edu**).

1. Login to Linux workstations in ZACH 127
    - Open a terminal using *Applications → Favorites → Terminal*
2. Remote login to ECE Linux servers through TAMU VOAL
    - Using a WEB browser, go to https://connect.voal.tamu.edu, and click on **VMware Horizon HTML Access** (alternatively, install VMware Horizon client)
    - After login, double click on **VOAL**
    - In the browser, from the Start menu, click on *Start → MobaXterm Personal Edition*
    - Type the **"Host Name"** into the box labeled **"Find existing session or server name…"** and press **Enter**. If asked to choose a session type, click on **SSH**, then **OK**.
3. Remote login to ECE Linux servers through COE VOAL
    - Connect to VPN unless you are on campus network
    - Using a WEB browser, go to https://coe-connect.engr.tamu.edu, and click on **VMware Horizon HTML Access** (alternatively, install VMware Horizon client)
    - After login, double click on **CoE General Desktop**
    - In the browser, from the Start menu, click on *Start → MobaXterm*
    - Type the **"Host Name"** into the box labeled **"Find existing session or server name…"** and press **Enter**. If asked to choose a session type, click on **SSH**, then **OK**.
4. Remote login to ECE Linux servers directly from your computer
    - Download and install MobaXterm (https://mobaxterm.mobatek.net/download.html)
    - Connect to VPN unless you are on campus network
    - Start MobaXterm to connect to ECE Linux servers as described in (2) or (3).

After you log into **olympus.ece.tamu.edu**, you need to type **load-ecen-704** and press **Enter** immediately after login.
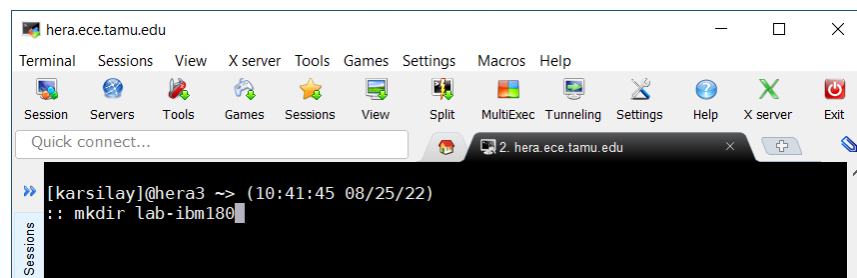

Figure 1-1: Creating a directory in a Linux shell

Linux operating system commands are issued by typing them into a shell, terminal or xterm, which is where you will be after completing one of the four access methods above. Figure 1-1 shows an example for creating a new directory. Table 1-1 shows some basic Linux commands.

If you type "`ps -ef | grep <NetID>`", it will list your processes running on the system. If your session crashes or freezes, the processes could still be running and slowing down the server without doing anything. Using the "ps" command above you can find the process ID (PID), which is the number in the column next to NetID. Now if you type "`kill -9 <PID>`", you can kill the process. You can also issue the command "`kill -1 -1`" to kill all your processes.

Table 1-1: Common UNIX Commands

ls –al	List all files in current directory, including hidden files
ll	List all files in current directory with time stamps and permissions
mkdir X	Create a new directory titled "X"
cd X	Change the current directory to "X"
cd ..	Change the current directory back one level
cd ~	Change the current directory back to the home directory
cp X Y	Copy file "X" to file "Y"
mv X Y	Move (or rename) file "X" to file "Y"
rm X	Delete file "X"
gedit X &	Open file "X" using the text editor "gedit", "&" returns the prompt to the shell
clear	Clear the screen (alternatively, you can press Ctrl-L)
top	Show the process/memory table (press q to exit)
rm -rf X	Remove the file or directory "X" without any confirmation

Running Cadence

The Cadence Development System consists of a bundle of software packages such as schematic editors, simulators and layout editors. All the Cadence design tools are managed by a software package called the Design Framework II. This program supervises a common database which holds all circuit information including schematics, layouts, and simulation data. From the Design Framework II, also known as the "framework", we can invoke a program called the "Library Manager" which governs the storage of circuit data. We can access libraries and the components of the libraries called cells. Also, from the framework we can invoke the schematic entry editor called "Composer", which is used to draw circuit diagrams and draw circuit symbols. A program called "Virtuoso Layout Suite" is used for creating integrated circuit layouts. The layout is used to create the masks which are used in the integrated circuit fabrication process. Circuit simulation is handled through an interface called Analog Design Environment (ADE). This interface can be used to invoke various simulators including HSPICE, Spectre, UltraSim, and Verilog. We will be using the Spectre simulator in this course.

To start Cadence, the first step is to create a new directory (see Figure 1-1) that will keep all Cadence designs for the ECEN 474/704 lab. This directory can be used only for one technology (for this course, IBM 180nm). If you need to use another process for a different project or course, you need to create another directory. **Never start Cadence directly from your home directory**, always change directory (cd) into the subdirectory corresponding to the technology you will use, then start Cadence as follows:

```
cd lab-ibm180
/disk/amsc/bin/ibm180
```

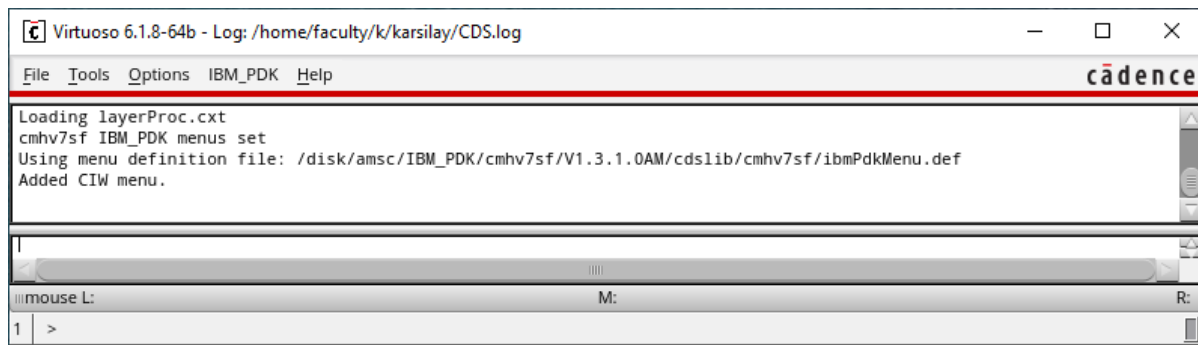The Command Interpreter Window (CIW) will now load as shown in Figure 1-2.

Figure 1-2: The Cadence Command Interpreter Window (CIW)

Creating a New Library

From the CIW (Figure 1-2) select *Tools → Library Manager* to load the library manager (Figure 1-3). The Library Manager stores all designs in a hierarchal manner. A library is a collection of cells. For example, if you had a digital circuit library named Digital, it will have several cells including inverters, nand gates, nor gates, multiplexers, etc. Each cell has different views such as symbol, schematic, or layout.
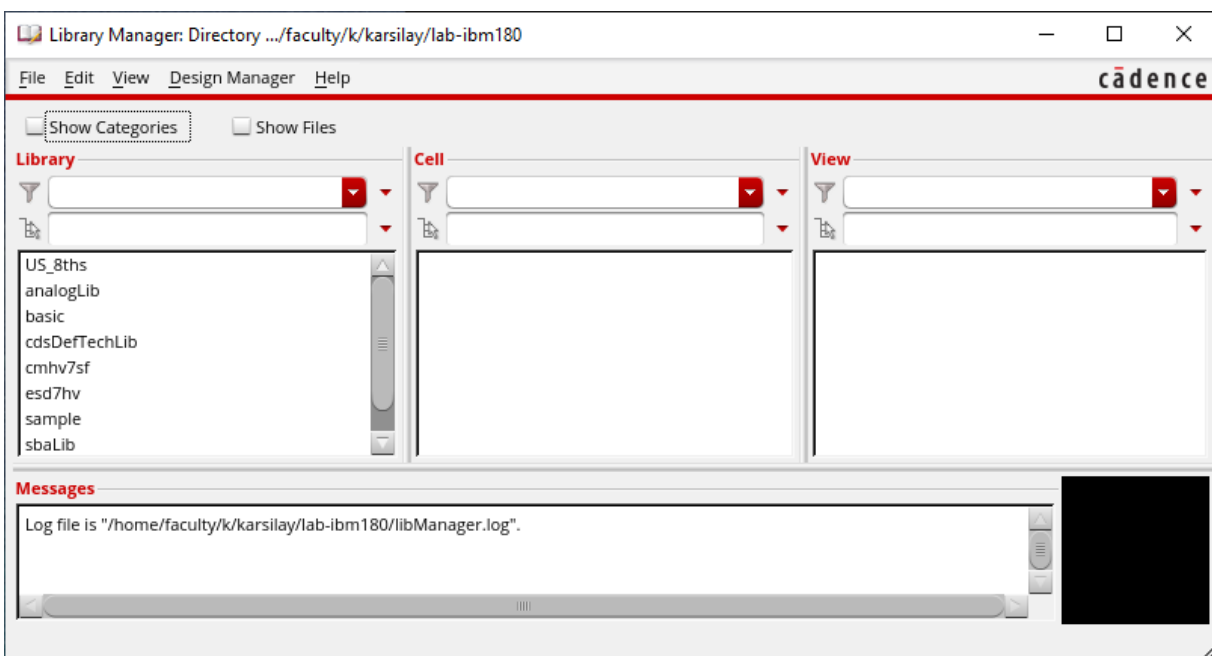

Figure 1-3: The Library Manager Window

The first step to start a design is to create a library to store the cells. From the Library Manager window select *File → New → Library*. Name the library "Lab1" (without the quotes) and select OK. In the popup window that appears select "Attach to an existing technology library" (Figure 1-4) and select OK. In the next window make sure the **cmhv7sf** technology library is selected and select OK. Every library that you create should be attached to the technology library that you are working with.
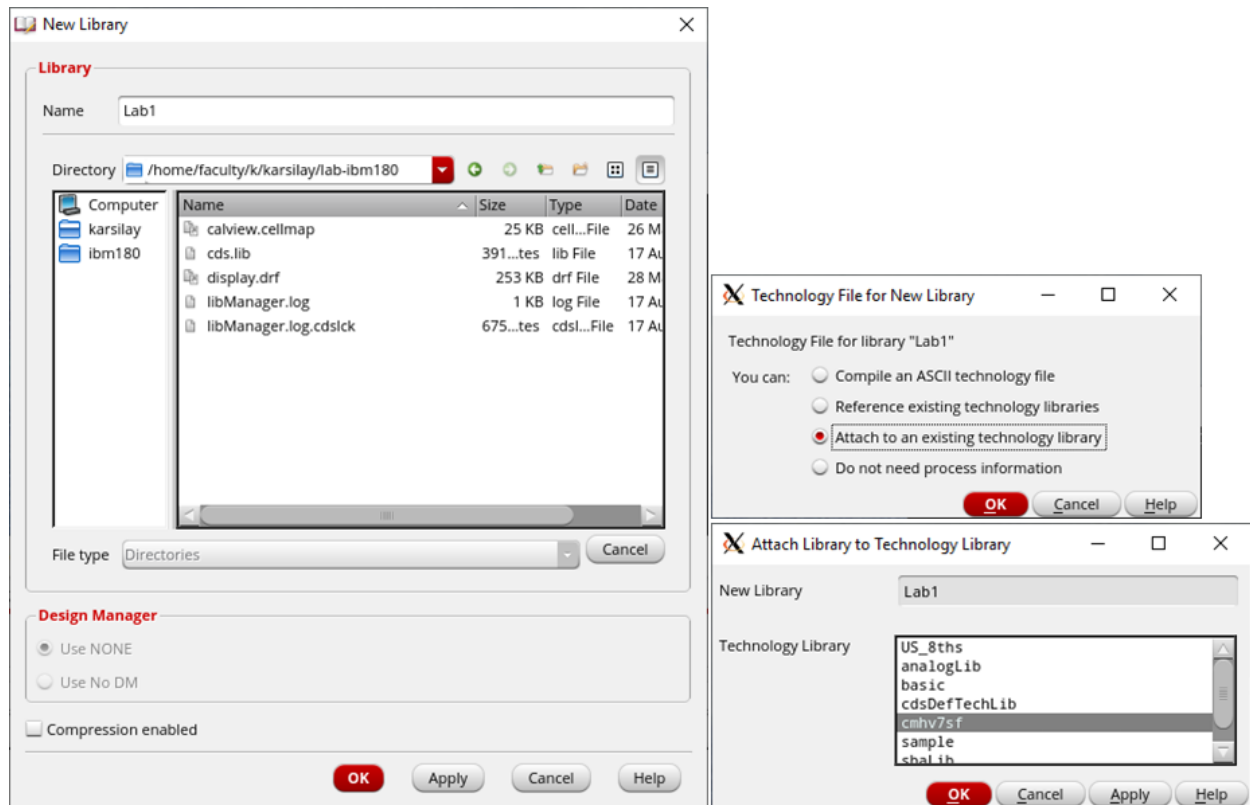
Figure 1-4: Creating a Library and Attaching a Technology Library

Creating a Cell

In the library manager window, click on "Lab1", then *File → New → Cell View*. Name your cell MosChar. The application you want to use here is Schematics L as seen in Figure 1-5. After selecting OK on the "New File" tab and OK on the next popup window, the schematic window opens. **If you see any popup windows asking to use the "XL" version of a license or a similar prompt, select "Always" in the window.**
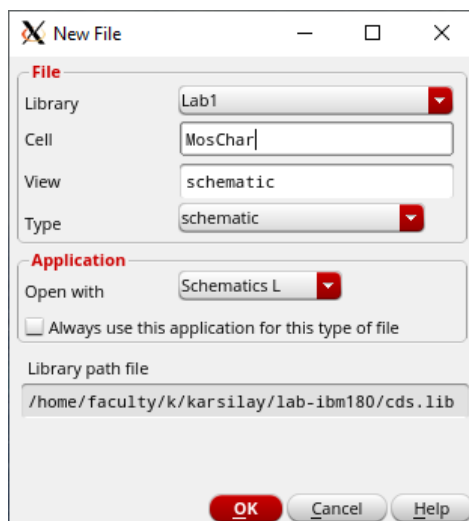

Figure 1-5: Creating a New Cell View

Working with the Schematic Editor

Transistors can be inserted into the schematic using the menu item *Create* → *Instance* or by pressing the "I" hotkey on the keyboard. In the "Component Browser" window, click on the "Browse" button, and make sure the library **cmhv7sf** is selected. Select **nfetx** and make sure **symbol** is highlighted. A window with all the FET parameters should pop up. The parameters may be edited if needed. Next, click "hide" and place the NFET on the schematic. You can go back to the Component Browser and select **pfetx** to insert a PMOS transistor. Voltage sources and grounds should be inserted from the "analogLib" library (the "vdc" and "gnd" components can be used for the purposes of this lab). The components can be connected with wires by selecting *Add* → *Wire* or using the "W" hotkey. To change the parameters of any component, you can use *Edit* → *Properties* → *Objects* or use the "Q" hotkey after selecting the device. A summary of common schematic editor hotkeys and commands is shown in Table 1-2.

Table 1-2: Common Schematic Editor Commands

W	Draw wire (double-click to leave the end of a wire floating)
I	Add instance
Delete	Delete anything that is clicked
Esc	Return the cursor to its normal state
R	Rotate element
M	Move element
F	Zoom fit
C	Copy element
U	Undo
Shift-U	Redo
L	Create wire label
Q	Open Property Editor for an instance
Shift-R	Flip element across y-axis (while moving)
Ctrl-R	Flip element across x-axis (while moving)
Ctrl-Z	Zoom out
Ctrl-D	Deselect everything
Right-click and drag	Will zoom in on drawn region
Shift-scroll	Move left and right in the schematic viewer window
Ctrl-scroll	Move up and down in the schematic viewer window
Arrow keys	Move up/down or left/right, according to pressed key, in schematic viewer window

It is generally good practice to label any wires that will be a potential signal of interest by using the "L" hotkey, mentioned in Table 1-2. Labeling wires provides descriptive names to more clearly indicate what signal is examined and keeps the signal names consistent in the event that changes are made to the schematic (default net names are given to unlabeled nets, but these names may change as the schematic is modified). In the label window, multiple wire labels can be created by separating each label with a space (as such, a wire label itself cannot contain spaces, but underscores are acceptable characters). Avoid using other special characters as part of a wire label as these can lead to errors in the netlist generation. Wire labels must also be unique, otherwise two wires sharing the same label will be connected together. Additionally, while netlists generated from the schematic editor are generally case-sensitive to wire labels, it is good practice to treat wire labels as case-insensitive (e.g. act as if "VDD" and "vdd" would be treated as the same net) due to later stages of the circuit design and verification process which will be explored in later labs.

Simulating a Schematic

Before simulating a schematic, you must ensure the schematic is updated by clicking the "Check and Save" button (). From the schematic window, select *Launch → ADE L*, click OK on the popup window, then the ADE L window should appear. To set up a DC analysis, select *Analysis → Choose → <select dc> → <check box "save dc operating point"> → <check box "component parameter"> → Select Component → <click on the source to be swept> → <double click the "DC" option> → <set start and stop> → <set sweep type> → <click OK>*. This method configures a parameter sweep because of the "component parameter" option. Other sweep options also exist, like a design variable sweep which is described below. Before running a simulation, signals should be saved to the list of outputs in the ADE L window. Select *Outputs → To be saved... → Select on Design* and then go to the schematic and click on wires to save voltage signals or click on pins (the red dot or square on a symbol) to save current signals. Once you are finished selecting signals to be saved, press ESC to return the cursor to its normal state. The "Outputs" section of the ADE L window should be populated with the voltages and/or currents that were selected. Checking the box in the "Plot" column associated with an output will cause the signal to be plotted automatically when the simulation is finished. To run the simulation, select *Simulation → Netlist and Run,* or click on the green play button along the right side of the ADE L window. Once finished, a new window should appear with plots of the selected voltages and currents. While in graph mode, you can use the "M" key to insert a marker. The "A" and "B" keys will insert markers as well, but they will give dx, dy and the slope between the two points (pairs of markers that display this kind of information are called "delta markers"). The "H" key will generate a horizontal bar, and the "V" key will generate a vertical bar. A summary of common hotkeys and commands for the waveform viewer is shown in Table 1-3. Additional information about the waveform viewer is provided in Appendix B of this document.

Table 1-3: Common Waveform Viewer Commands

M	Place marker on a trace
A	Place first marker for a delta marker pair
B	Place second marker for a delta marker pair
D	If a marker is selected when this is pressed, creates a delta marker pair (can create multiple sets of delta markers)
Shift-scroll	Zoom in or out along x-axis
Ctrl-scroll	Zoom in or out along y-axis
F	Zoom fit for all traces
U	Undo
Shift-U	Redo
V	Place vertical line marker
H	Place horizontal line marker
Right-click and drag	Will zoom in on drawn region
Ctrl-E	Clears plot of all markers
Q	Open Property Editor for selected marker, axis, trace, or window
N	Move marker to next data point on the trace
P	Move marker to previous data point on the trace
Arrow keys	Pan around the plot area when zoomed in on the x- and/or y-axes

In addition to the parameter sweep described above, many parameter fields for a component will allow you to input a design variable to be used for simulations. For example, if a DC voltage source (the "vdc" component from the "analogLib" library) is used as a positive supply voltage with different values for different simulations, then the DC voltage parameter field of the source could be given a value of "Vdd" which will create a design variable of the same name. Variables created in the schematic can be transferred to the ADE L window using the ADE L menu item *Variables → Copy from Cellview*. To use the variable

for a DC analysis, select *Analysis* → *Choose* → *<select dc>* → *<check box "save dc operating point">* → *<check box "design variable">* → *<enter the name of your design variable>* → *<set start and stop>* → *<set sweep type>* → *<click OK>*. Once set up, run the simulation as above. **Note: do not try to use a unit prefix (e.g. "n" for "nano" or "u" for "micro") as part of a variable name. This will not be correctly interpreted when importing design variables from the schematic. Instead, include the unit prefix when defining the variable value in the ADE L window.** If a simulation does not run while attempting to use a design variable in a parameter field, you may be forced to use a parameter sweep (as described earlier) instead while providing a default value in the schematic for the parameter.

Once the simulation results are plotted, data can be extracted from the waveforms and/or new waveforms can be created from the simulation data using the calculator. The calculator can be accessed from the *Tools* menu in either the ADE L window or the waveform viewer window. For additional information regarding the calculator, see Appendix A at the end of this document.

In addition to DC simulations, AC simulations will also be needed for the characterizations that will be performed in this lab. From the ADE L window, select *Analysis* → *Choose*, or click the button along the right side of the ADE L window. From the analysis setup window, select "ac" as the analysis type, make sure that "Frequency" is chosen as the sweep variable, use the "Start-Stop" sweep range, set the "start" as 1 and the "stop" as 100G (i.e. 100GHz), set the "Sweep Type" to "Logarithmic" with 20 points per decade, and then click "OK" to save the analysis setup to the ADE L state. Before beginning a simulation, you must set the "AC magnitude" parameter of one of the voltage sources in the schematic to be a non-zero value. Otherwise, the AC magnitude field must be assigned a design variable name, and the value of that design variable must be non-zero before performing the AC simulation. **Note that all sources with a non-zero AC magnitude field will be active during an AC simulation. Therefore, only one source (the input source for the corresponding test) should have a non-zero AC magnitude parameter for a given simulation.**

After the AC magnitude parameter is properly set, perform the AC simulation. **Note that the AC simulation is a small-signal simulation (i.e. a linearization of the circuit) based on the DC operating point. If the DC operating point is incorrect, then the circuit's performance will not be close to what is expected.** The DC operating point is determined by any default design variable values in the ADE L state and/or any fixed parameter values in the schematic. See Appendix A for one method of verifying that the circuit's DC operating point is set appropriately. Once the AC simulation is finished, the results will be plotted in the waveform viewer window, much like the DC simulations performed previously. However, the x-axis should be the frequency of the signal (in Hz) because of the "Sweep Variable" setting in the AC analysis. Signals are assumed to be sinusoidal, and the data for a signal is stored as a complex variable (a "phasor") to store magnitude and phase shift information. When the AC signal is plotted, the default plotting behavior is to plot the magnitude of the signal as a function of frequency. This lab will focus on the use of signal magnitudes, but future labs will also utilize the phase information. Additionally, **DC offsets in the signals are ignored when plotting results from AC simulations. Only the AC components of voltages and currents are displayed in the AC simulation results.**

Additionally, for future reference, the AC magnitude field is treated separately from the amplitude field of other transient sources (like sinusoidal or square wave sources). For sources that provide a transient waveform, there are multiple amplitude fields. One of them is the "AC magnitude" field we have been discussing. The other field may refer to an amplitude or different voltage levels for the min and max values of the waveforms. If the other fields do not mention "AC" (or some other analysis type) in the parameter name, then they are generally for transient simulations. **In other words, a source's magnitude in AC simulations and its magnitude in transient simulations are completely independent from each other.**

MOS Transistor Operation

MOS transistors are the fundamental devices of CMOS integrated circuits. The schematic symbols for an NMOS and PMOS transistor are illustrated in Figure 1-6 and Figure 1-7, respectively.
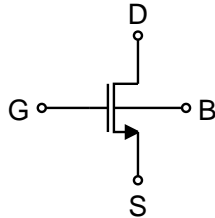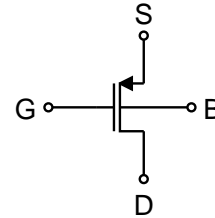
Figure 1-6: NMOS Transistor   **Figure 1-7:** PMOS Transistor

A cross sectional view of an NMOS transistor is shown in Figure 1-8. When the potential difference between the source (S) and the Drain (D) is small (~0 V), and a large potential (> V_t) is applied between the gate (G) and source, the transistor will be operating in the triode (or linear) region. The positive gate potential causes electrons to gather below the surface of the substrate near the gate in a process called "inversion". This region of mobile charge forms a "channel" between the source and drain. The amount of charge is a function of the gate capacitance (C_{ox}) and the gate-to-source overdrive voltage:
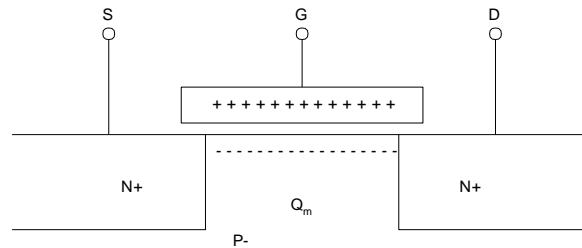
$$Q_m = C_{ox}(V_{GS} - V_t)$$

Figure 1-8: Cross-Sectional View of an NMOS Transistor

The term V_t is the threshold voltage. When the gate-to-source voltage (V_{GS}) exceeds this value, an inversion region is formed. Before reaching the inversion region, as the gate-to-source voltage is increased, the transistor passes through the accumulation region where holes are repelled from and electrons are attracted to the substrate region under the gate. Immediately before inversion, the transistor reaches the depletion region (weak-inversion) when the gate to source voltage is approximately equal to the threshold voltage. In this region a very small current flows.

In the linear region, the MOSFET acts a voltage-controlled resistor. Resistance is determined by V_{GS}, transistor size, and process parameters.

When the drain-to-source voltage (V_{DS}) is increased, the quantity and distribution of mobile charge carriers becomes a function of V_{DS} as well. Now the total charge is given by:

$$Q_m = C_{ox}(V_{GS} - V_t - V_{DS})$$

Distribution of this charge is such that Q_m is greater near the source and less near the drain. To find the channel conductance, the charge must be recast as a function of position $Q_m(y)$ and integrated from the source to drain. Since the charge is a function of V_{DS}, the conductance depends on V_{DS}. The channel current becomes:

$$I_D = \mu_0 C_{OX} \frac{W}{L}(V_{GS} - V_t)V_{DS}$$

or

$$I_D = KP \frac{W}{L}(V_{GS} - V_t)V_{DS}$$

As V_{DS} increases, eventually the drain current saturates, that is, an increase in V_{DS} does not cause an increase in current. The saturation voltage depends on V_{GS} and is given by:

$$V_{DS(sat)} = V_{GS} - V_t$$

The equation of the drain current becomes:

$$I_D = \frac{1}{2}KP \frac{W}{L}(V_{GS} - V_t)^2$$

At this point the transistor is operating in the saturation region. This region is commonly used for amplification applications. In saturation, I_D actually depends weakly on V_{DS} with the parameter λ. Also, the threshold voltage depends on the bulk-to-source voltage (V_{BS}) through the parameter γ. A better equation for the MOSFET (that includes the effects of V_{BS}) in saturation is given by:

$$I_D = \frac{1}{2}KP \frac{W}{L}(V_{GS} - V_t)^2(1 + \lambda V_{DS})$$

When V_{GS} is less than the threshold voltage, the channel also conducts current. This region of operation is called weak-inversion or sub-threshold conduction. It is characterized by an exponential relationship between V_{GS} and I_D. Also, when V_{GS} becomes very large the charge carrier's velocity no longer increases with the applied voltage. This region is known as velocity saturation and has an I_D that depends linearly on V_{GS} as opposed to the quadratic relation shown above.

Figure 1-9 is a three-dimensional cross-sectional view of a MOSFET. Notice in the figure the overlap between the gate region and the active regions. The overlap forms parasitic capacitors C_{GS} and C_{GD}.
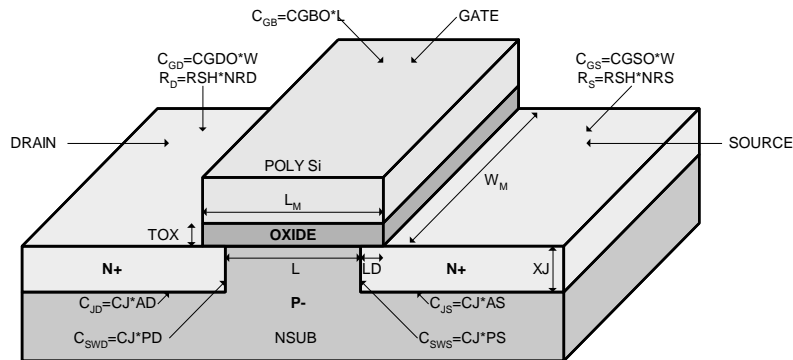

Figure 1-9: Physical Structure of a MOSFET

The reverse-biased junctions between the active regions and the bulk form the parasitic capacitors C_{DB} and C_{SB}. The conductivity of the active regions forms the parasitic resistors R_D and R_S. A schematic symbol with these parasitic elements is illustrated in Figure 1-10.
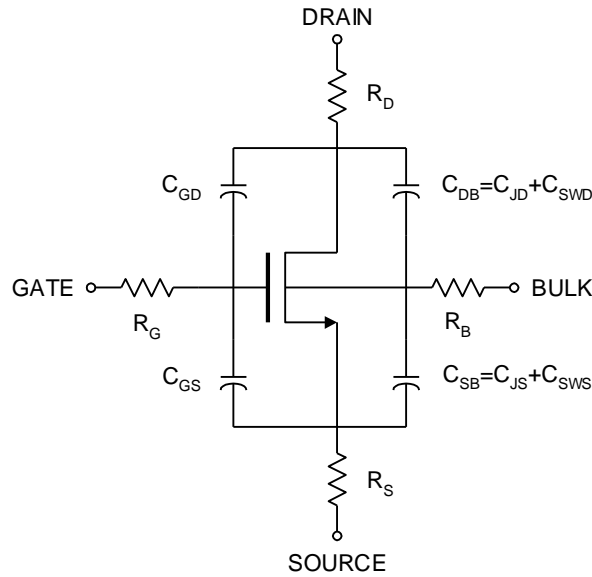

Figure 1-10: MOSFET Parasitic Resistors and Capacitors

MOS Device Characterization

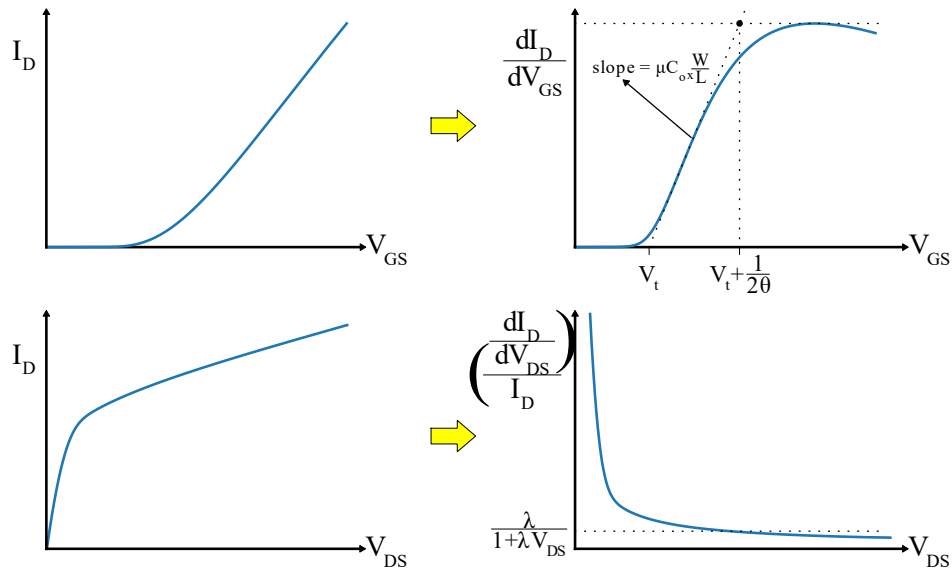The following plots illustrate extraction of the main parameters from a MOS transistor:

Figure 1-11: Example DC sweep plots for extraction of key process parameters

To obtain g_m, the derivative of I_D needs to be plotted as a function of V_{GS}. The resulting plot can be used to extract V_t, μC_{ox} and θ, as shown above. The channel length modulation parameter, λ, can be extracted by taking the derivative of I_D as a function of V_{DS} and further dividing it by I_D. From the square-law model equations, this yields a plot of $\lambda/(1+\lambda V_{DS})$, from which λ can be determined at a particular value of V_{DS} as shown above.

Lab Report

The device width mentioned below refers to the transistor's total width (the "width all fingers" parameter).

1. Construct the circuit below using **nfetx** transistor with W/L=0.5μ/0.18μ using 1 finger.
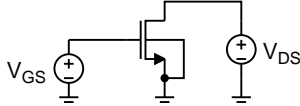
   a. V_{DS}=1V. Sweep V_{GS} from 0 to 1.8V, plot (dI_D/dV_{GS}) vs. V_{GS}, and extract V_{tn}, $\mu_n C_{ox}$ and θ.
   b. $V_{GS}=V_{tn}$+50mV+1/(4θ). Sweep V_{DS} from 0 to 1.8V, plot (dI_D/dV_{DS})/I_D vs. V_{DS} and extract λ at V_{DS}=1V. Additionally, find the unity-gain frequency (f_t) using AC simulations.
   c. Reconfigure the circuit if necessary and find the intrinsic gain (A_i) using AC simulations when V_{DS}=1V and $V_{GS}=V_{tn}$+50mV+1/(4θ).
2. Repeat (1) for the **nfetx** transistor with W/L=5μ/0.18μ using 1 finger.
3. Repeat (1) for the **nfetx** transistor with W/L=5μ/0.18μ using 10 fingers (i.e. "width all fingers"=5μ while "width single finger"=0.5μ).
4. Repeat (1) for the **nfetx** transistor with W/L=5μ/0.27μ using 1 finger.
5. Construct the circuit below using **pfetx** transistor with W/L=0.5μ/0.18μ using 1 finger.
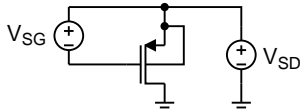
   a. V_{SD}=1V. Sweep V_{SG} from 0 to 1.8V, plot dI_D/dV_{SG} vs. V_{SG}, and extract V_{tp}, $\mu_p C_{ox}$ and θ.
   b. $V_{SG}=|V_{tp}|$+50mV+1/(4θ). Sweep V_{SD} from 0 to 1.8V, plot (dI_D/dV_{SD})/I_D vs. V_{SD} and extract λ at V_{SD}=1V. Additionally, find the unity-gain frequency (f_t) using AC simulations.
   c. Reconfigure the circuit if necessary and find the intrinsic gain (A_i) using AC simulations when V_{SD}=1V and $V_{SG}=|V_{tp}|$+50mV+1/(4θ).
6. Repeat (5) for the **pfetx** transistor with W/L=5μ/0.18μ using 1 finger.
7. Repeat (5) for the **pfetx** transistor with W/L=5μ/0.18μ using 10 fingers.
8. Repeat (5) for the **pfetx** transistor with W/L=5μ/0.27μ using 1 finger.

You may find it useful for this lab to create multiple schematics and/or cell views for some of the tests. Likewise, it may also be helpful to utilize multiple ADE L states for the above tests, rather than attempting to keep all sweeps and outputs in a single ADE L state, due to the variety of sweeps and parameters that need to be evaluated. See Appendix A for information about building and saving an ADE L state.

When sweeping V_{DS} for the transistors (part b in the above characterizations), begin the DC sweep of V_{DS} at a non-zero value and choose the "linear" sweep type rather than "automatic" for properly scaling the plots needed to extract λ.

The results of this characterization will be used for design calculations in future labs, so use any method that you believe will give you the most accurate results. Include all schematics, plots, annotations, and calculations. Summarize the characterization results in a single table. Comment on the results. **Note:** separate images are not needed for duplicate schematics where only the dimensions of the transistor are changed.

Appendix A: Cadence Calculator and ADE L Outputs

This section provides more information regarding the ADE L test setup window, particularly with regard to setting up output expressions. An overview of the calculator tool in Cadence is also presented along with mention of a few functions that will be particularly useful for the purposes of this lab but will also be useful in future labs.

Within the list of outputs in the ADE L window, user-defined expressions can be created to calculate waveforms or values of interest from a plot after a set of simulations is performed. Output expressions can be created by going to *Outputs* → *Setup* or by clicking the [icon] button on the right side of the ADE L window. If the function(s) and syntax of the output expression are already known, these can be typed in the output editor window directly to create the new expression. After defining the expression, click the "Add" button at the bottom of the outputs editor window to add the expression to the list of outputs, and then click "OK" to close the window. Otherwise, the expressions can be defined using the calculator and then the expressions can be imported into the ADE L outputs list.

The calculator can be accessed from the *Tools* menu in either the ADE L window or in the waveform viewer window. It can also be accessed by clicking the "Open" button next to "Calculator" in the output editor window. **Note that signals cannot be used directly when defining expressions.** This is because when a signal is saved in the ADE L outputs list, the data from every kind of analysis is stored and associated with the signal. Instead, the signal data associated with a specific kind of analysis (e.g. transient, DC, AC, etc.) must be used to create a user-defined expression. This analysis-specific data is returned using the functions associated with the radio buttons near the top of the calculator window (VT, IT, VDC, VS, OS, etc.) as shown in Figure 1-12. Hovering your mouse over the radio button will provide a short description about what kind of simulation data is returned by the function. Click on one of the radio buttons and then go to the schematic and click the appropriate net/pin/instance (as appropriate for the function) on the schematic. **There is an important distinction between the VDC or IDC functions and the VS or IS functions.** The VDC/IDC functions will return a <u>single value</u> of a DC voltage/current for a <u>single DC operating point</u>. This single point is defined by any fixed parameters in the schematic along with any default values of design variables in the ADE L window. The VS/IS functions return the <u>waveform</u> associated with a DC voltage/current resulting from a <u>DC sweep of a variable or parameter</u>. The sweep variable is configured in the DC analysis setup, as discussed in the "Simulating a Schematic" section.

Figure 1-12: Calculator buttons for simulation-specific signal data

The following calculator functions may also be useful for this lab and future labs:
- *ymax* – returns the maximum value of a waveform
- *value* – provide a waveform and an x-coordinate and the function returns the waveform value at the provided x-coordinate
- *deriv* – takes the (discrete) derivative of the provided waveform. The derivative is taken with respect to the x-axis variable.
- *cross* – provided a waveform and a y-axis value, the function returns the x-coordinate at which the desired y-axis value occurs. Additional arguments are needed to determine whether the crossing point is a rising or a falling edge or to return the x-coordinate associated with a different occurrence of the same y-axis value (e.g. returning the time at which the second or third edge of a clock occurs rather than the first edge)

- *tangent* (as opposed to *tan*, which is the trigonometric function) – provided a waveform, x-coordinate, y-coordinate, and a slope, the function will draw a tangent line in the current plotting window that is associated with the same axis as the provided waveform and passes through the specified point with the specified slope.

When creating expressions that utilize the above functions or other pre-defined functions in the calculator, clicking on the function will bring up a tool for building the function, similar to what is shown in Figure 1-13 for the *cross* function. In this example, the *cross* function would return the x-coordinate of the first instance for which the "dc_gmn" waveform (a named output expression) reaches its maximum value. Clicking "OK" at the bottom of the sub-window will load the completed expression with the user-defined arguments into the calculator buffer and close the function builder window (if the "OK" button appears to be hidden, you may need to vertically expand the calculator window in order to see it). For simpler functions that only take in a single argument (such as *deriv*, or *ymax*) the tool will not appear and instead the function will encapsulate any variables and/or expressions that are currently in the calculator buffer.
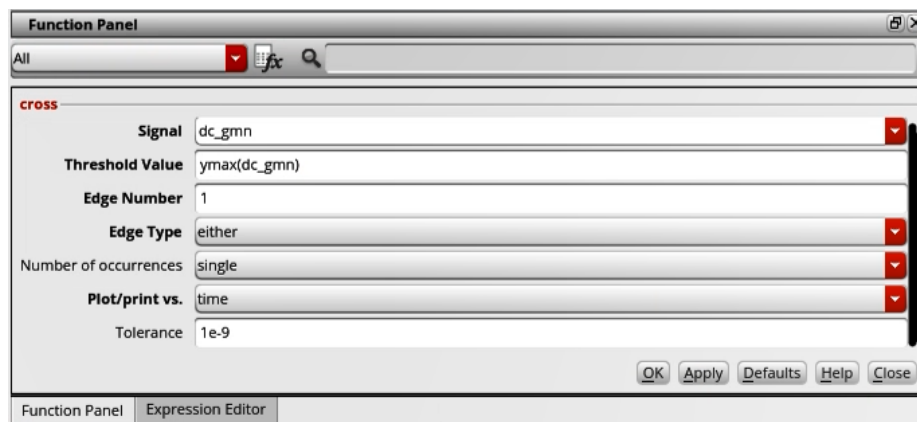

Figure 1-13: Example of the function builder tool using the cross function

After creating an expression in the calculator buffer, click the button to plot the expression if it returns a waveform or to display the expression result if it only returns a single value. If nothing occurs when clicking the evaluation button, there will likely be an error message at the bottom of the calculator window and/or in the CIW indicating that there is a problem with the expression definition. An important note regarding calculator expressions: **expressions will not be plotted or evaluated if they contain unbalanced parentheses or if they contain more parentheses than are necessary**. This is due to the programming language from which Cadence is constructed. **Therefore, when defining expressions, use only the minimum number of parentheses that are needed for preserving the correct order of operations**. The calculator will add any pairs of parentheses above this minimum that are expected by the programming language, but extra parentheses beyond this will be interpreted incorrectly.

An output expression can be created in the calculator tool and then exported to the ADE L outputs list using the button at the top of the calculator buffer. When the expression is exported, the definition will appear in the list of outputs, but underlined: user-defined expressions can be renamed in the output editor window to be more meaningful and easier to reference later. Named output expressions can be used in the definitions of other expressions to define more complex outputs. **When naming expressions, use underscores instead of spaces to separate portions of the output name so that the output can be easily referenced when creating other expressions.** If an output expression returns another waveform, that waveform can be plotted along with the other outputs in the ADE L window. If an expression returns only a single value, the returned value will be displayed in the "Value" column of the ADE L outputs list.

When new outputs are added to an ADE L state, they must be reevaluated before being plotted. Output expressions can be reevaluated without running a new simulation by pressing the ![icon] button at the right of the ADE L window. Note, however, that if an expression utilizes signals or other data that was not saved in the previous simulation, a new simulation may be necessary. For example, current signals are not saved by default unless explicitly added to the ADE L outputs list. If a new current signal is added to the outputs list and then a user-defined expression is created that utilizes that current signal, a new simulation will be necessary.

After building the output expressions or making any changes to the ADE L state configuration, you can save the ADE L state for later use so that the analyses and outputs do not need to be built from scratch again. Click *Session* → *Save State…* or click the ![icon] button near the top of the ADE L window. In the window that appears, it is recommended to change the "Save State Option" at the top of the window to "Cellview" instead of "Directory." Saving the state as a cell view allows you to associate the ADE L state with its corresponding test bench schematic. Saving the ADE L state in this way also causes it to appear in the view list associated with a cell when viewed from the Library Manager, making it easier to identify any states that have already been created rather than looking through your home directory structure for the ADE L state that was used with a particular schematic. Similarly, a previously created ADE L state can be loaded by clicking *Session* → *Load State…* or by clicking the ![icon] button near the top of the ADE L window. Changing the "Load State Option" to "Cellview" at the top of the window that appears will allow you to load a previously created ADE L cell view associated with a schematic.
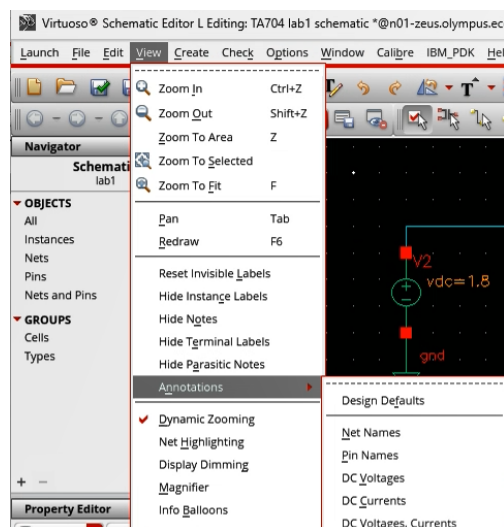

Figure 1-14: Schematic annotation menu

Finally, although not a function associated with the calculator, there is an option to display DC operating point properties directly in the schematic window after a DC simulation is performed. To do this, from the schematic window go to *View* → *Annotations*, and toggle one of the options for the DC voltages, currents, or operating points, like what is shown in Figure 1-14. These annotations can be used to quickly verify that a circuit's DC operating point has been appropriately configured before further testing. In this lab, for example, the test conditions for parts (b) and (c) of the transistor characterization procedures can be verified with these annotations before utilizing the results from the AC simulations.

Appendix B: Waveform Viewer

When a new waveform window is created, typically all of the waveforms will be collected into a single window (if some of the waveforms are user-defined expressions, these may appear in separate sub-windows within a new tab). A single waveform line is referred to as a "trace," and a collection of traces that share a single y-axis within a sub-window is called a "strip." Near the top of the waveform viewer window is the following set of buttons: . In order, these buttons will (1) separate all traces into individual strips, (2) group together all traces into a single strip, (3) split the waveforms in the currently selected strip, (4) copy the selected trace to a new strip, or (5) move the selected trace to a new strip. Strips can be moved around the sub-window by clicking on blank space in the plot grid and then dragging the strip. You can also click-and-drag traces to move them between strips and/or sub-windows.

Along with taking screenshots of the waveform viewer window (which is acceptable for lab reports), the window (or individual sub-windows within a tab) can be saved using the button near the top of the window. Saving images in this way allows for more options regarding the appearance of the window (e.g. the plot background color can be changed or certain components of the plot can be ignored) and allows a broader choice of file formats compared with taking a screenshot, such as PNG, JPG, PDF, or scalable vector graphic (SVG) formats.

Selecting the x or y axis and pressing the Q hotkey (or right-clicking on the axis and selecting "Axis Properties…") opens the axis properties. From here, the min and max values, grid line spacing, and the type of scale (linear or logarithmic) can all be controlled. The axis can also be changed to linear or logarithmic scale by right-clicking on the axis and enabling or disabling the "Log Scale" option. **Note**: if you select the "automatic" option for the "Sweep Type" parameter when setting up a simulation in an ADE L state, the sweep type will default to a linear sweep if the starting point is zero but will default to a logarithmic sweep if the starting point is non-zero. Therefore, explicitly specifying the Sweep Type parameter and the step size or number of steps may be preferable in several cases.

Right-clicking on a trace will allow you to adjust the color of the trace, the style (solid, dotted, etc.), and the thickness of the trace for better contrasts and visibility if needed. These options are selected near the bottom of the right-click menu or can be accessed in the trace properties window (using the Q hotkey again with the trace selected, or selecting "Trace Properties" from the right-click menu). Additionally, the typical copy and paste hotkeys may not function properly in the waveform viewer, so you may need to right-click on a trace and select "copy" from the menu and then right-click in another legend space and select "paste" instead of using hotkeys. Within a given sub-window, traces can be dragged and dropped between different strips. Holding Ctrl while left-clicking on traces will allow selection of multiple traces for a single action.

You can also send a trace to the calculator if you wish to perform further operations on it. To do this, right click on a trace in the legend and in the menu that appears go to *Send To → Calculator*. If the calculator window has not already been initialized, then it will appear and a calculator expression corresponding to the selected trace will appear in the calculator buffer. If the calculator window has already been initialized, then the calculator buffer will be updated with the expression (note that the calculator window may not automatically come into focus when sending a trace to the calculator).

Figure 1-15: Waveform x-axis selection box

Along with the shortcuts recorded in Table 1-3 that can be used to zoom in or out along the x-axis, the x-axis selection box, like what appears in Figure 1-15, which appears at the top of a waveform viewer sub-

window can be used to limit the view to a specific region of the x-axis. Clicking and dragging the left or right end of the selection box will change the min or max x-axis value (respectively). If the box does not take up the entire length of the x axis, then you can click inside the box and drag it to shift the box's position along the x-axis range while keeping the width of the box constant. Additionally, right-clicking on the selection box and selecting "Zoom to..." will allow you enter specific min and max values that you want to zoom to within the x-axis.

Labels or annotations can be created on a plot sub-window by right-clicking the plot area and selecting "Create Graph Label" from the menu. Clicking outside of the label's text box will set the text, and double-clicking the label will allow you to edit the text again. The font size of the label can be edited by selecting the label, pressing the Q hotkey to bring up the properties, and clicking the button in the "Font/Color" section of the properties window.

When a marker is created, the "N" and "P" hotkeys will allow you to scroll to the next or previous data point, respectively, in the trace data. This scrolling can also be controlled with the following buttons and menu: ⬅️ ➡️ Data Point ▾. These buttons and the drop-down menu appear near the top of the waveform viewer window. The buttons perform the same function as the N and P hotkeys. The drop-down menu controls what kinds of points the markers will jump to when the buttons (or the N and P hotkeys) are pressed. The default option from the drop-down menu is "Data Point," but this can be changed to various maxima or minima of the trace or to specific x- or y-axis values. In the last case of jumping to specific values along one of the axes, the box next to the drop-down will allow you to specify the value that you want to jump to. Similarly, selecting a marker and opening its properties (the Q hotkey) will enable you to change the "Position" setting. The default is "byXMode," but changing this to "byYMode" will enable you to specify a y-axis value to which the marker should move.

For horizontal and vertical markers, the points intercepting the markers will normally be displayed when the marker is selected or your mouse hovers over the marker. To keep the intercepts displayed when the marker is not selected, right-click on the marker and select *Intercepts* → *On*. This option can also be changed from the properties window of the marker.

Lastly, you can find additional simulation parameters and data from the results browser sub-window. To open the results browser, in the waveform viewer window go to *Browser* → *Results* → *Open Results...* using the menus at the top of the window, as shown in Figure 1-16(a). A file browser window will appear that should automatically open to the directory where the simulation data is stored. The data is stored in a file called "psf," as shown in Figure 1-16(b). Open this file and a new sub-window should appear at the left of the waveform viewer window. Within this sub-window you can navigate through the available simulation data to plot additional simulation data that is saved but which may be harder to access directly from the calculator.
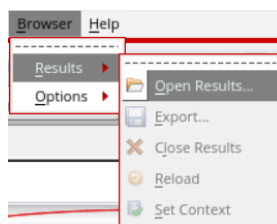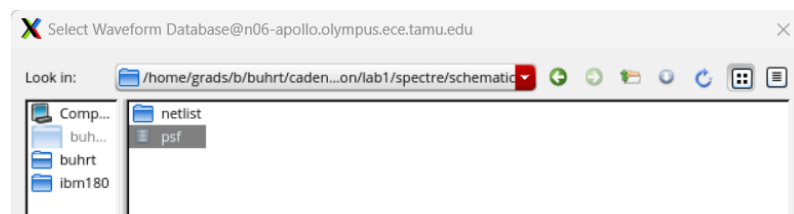
(a)   (b)

Figure 1-16: (a) Accessing the results browser sub-window and (b) selection of the "psf" file that stores simulation data