Full Custom Design

Warning!

While not mentioned explicitly, it is advised to ensure after every step that your design is free of errors. This will help you debug the problem at the initial stage. For schematic, the F8 shortcut will work. For layouts, you will have to select Verify > DRC ...

1 Introduction

In this lab, we will characterize an NMOS transistor and find its parameters. Specifically, we will find the following parameters

- λ (Channel Length Modulation Parameter)
- Threshold Voltage
- Transcoductance Parameter
- Body Effect Parameter

2 Transfer Characteristics

Start virtuoso by typing the following commands

```
cd  /vlsi
virtuoso &
```

Using the Library Manager, copy the library we created for an NMOS IV simulation to a new library 'NMOS_Char'

Open the schematic for the NMOS_IV_cell.

Launch ADE L for simulation. (If lucky, you might reuse the saved state by opting for Load State)
Go to Analyses > Choose ...

Choose a dc sweep from 0 to 5V with 1mV increment (like we did before).

Choose 'Netlist and Run'

You will see something like below

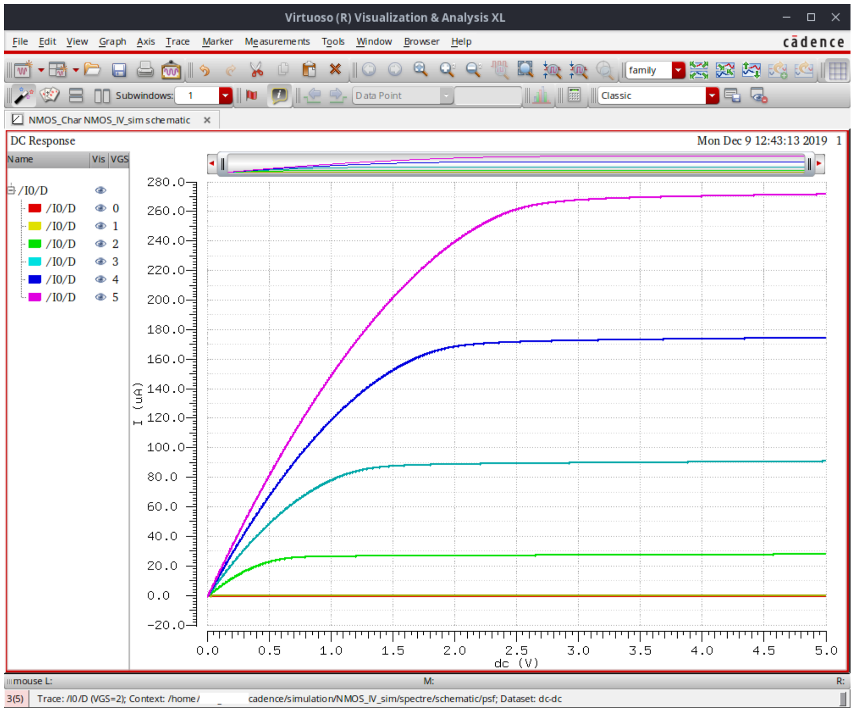

Figure 1: Transfer Curve of the Inverter

Select any specific curve. Launch calculator by going to Tools › Calculator ... Make sure the wave option is selected in the last toolbar

The window must show something like **leafValue(getData("/I0/D" ?result "dc") "VGS" 4)**. If not, you need to select the waveform..

2.1 Channel Length Modulation Parameter

In the Function Panel, select Special Functions and value. Type as follows

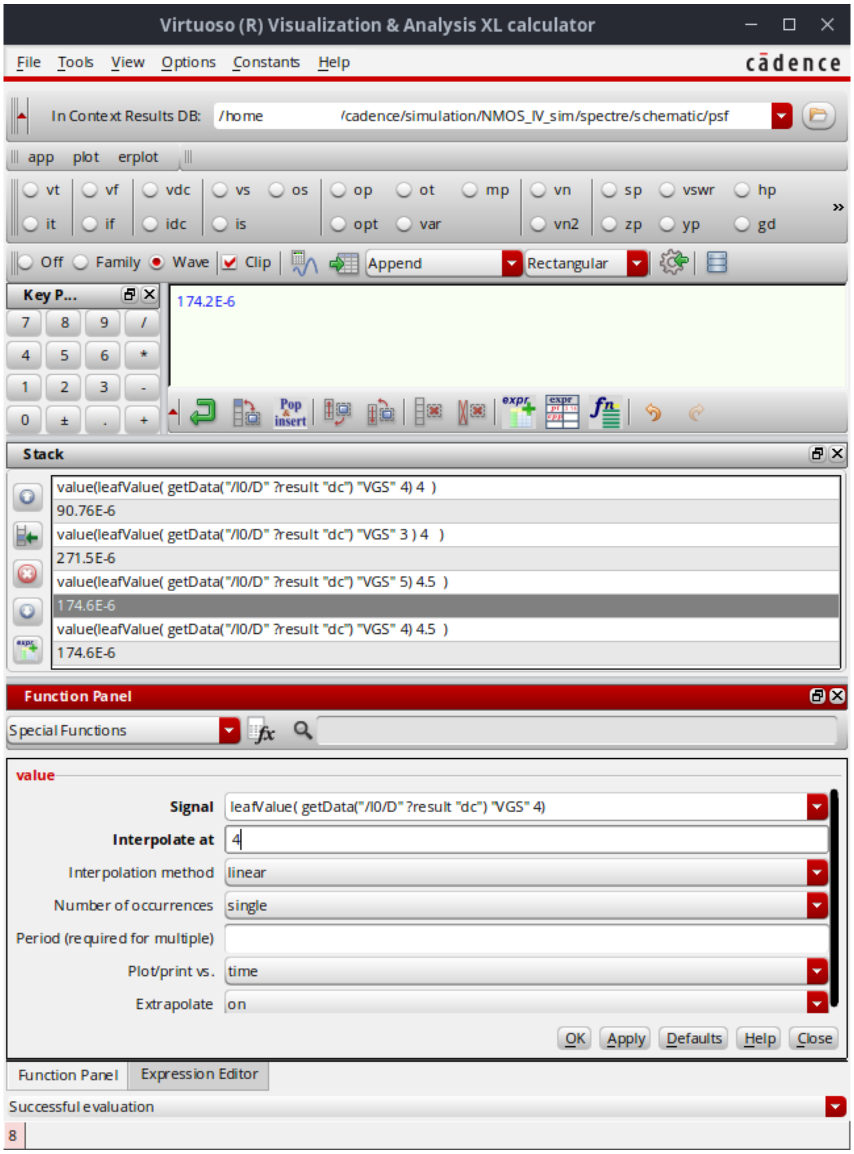

Figure 2: Buffer Windows -Derivative of Output Curve

Choose the 'Evaluate the buffer' icon to get the value (My answer was 174.2E-6)

Repeat for current at V_{DS}=4.5 V. Choose the 'Evaluate the buffer' icon to get the value (My answer was 174.6E-6)

Type the values of the following expression in the buffer window (or use MATLAB etc.)

$$\lambda = \frac{I_{D2} - I_{D1}}{I_{D1}V_{DS2} - I_{D2}V_{DS1}}$$

Choose the 'Evaluate the buffer' icon to get the value (My answer was 4.68E-3)

2.2 Threshold Voltage

$$V_{T0} = \frac{V_{GS1} - V_{GS2}\sqrt{\frac{I_{D1}}{I_{D2}}}}{1 - \sqrt{\frac{I_{D1}}{I_{D2}}}}$$

Using similar steps in section 2.1, find I_D for two different V_{GS}, but same V_{DS}

(For V_{DS} = 4, I got I_D= 90.76E-6, V_{GS}=3 and I_D= 174.2E-6, V_{GS}=4)

Type the values of the expression in the buffer window. Choose the 'Evaluate the buffer' icon to get the value (My answer was 0.4053)

3 ON Resistance

$$R = \frac{1}{V_{DD}/2} \int_{V_{DD}/2}^{V_{DD}} \frac{V}{I_{d,sat}(1+\lambda V)} dV$$

Using the Library Manager, copy the library we created for an NMOS IV simulation to a new library 'NMOS_RON'

(You may also start with a blank cell. A schematic will suffice)

First, assign a value 5V to the voltage source at the gate. Plot both the Drain voltage and the drain current as follows.

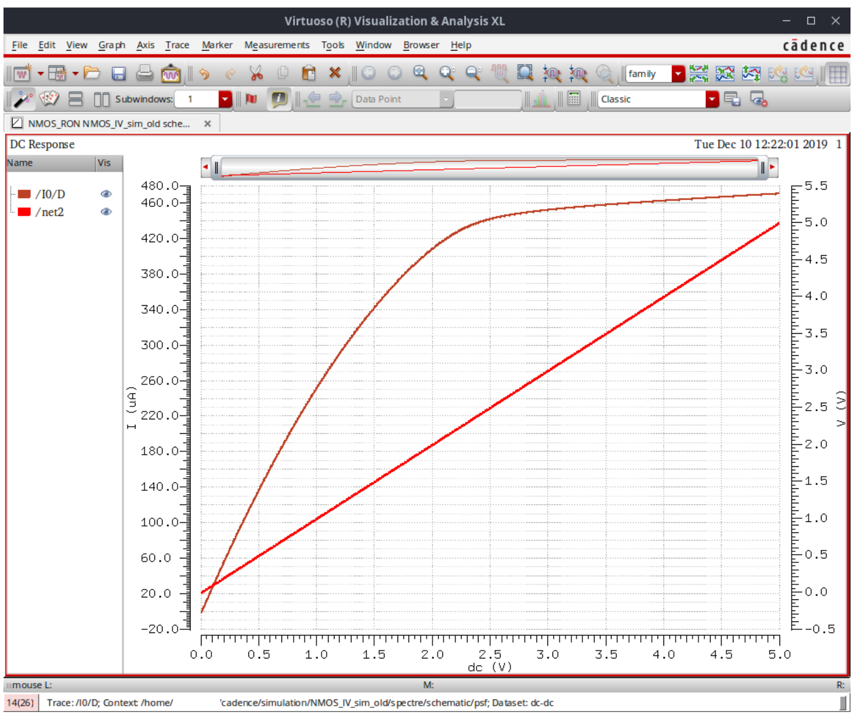

Figure 3: Buffer Windows -Derivative of Output Curve

Now integrate V/I from 2.5 to 5 and

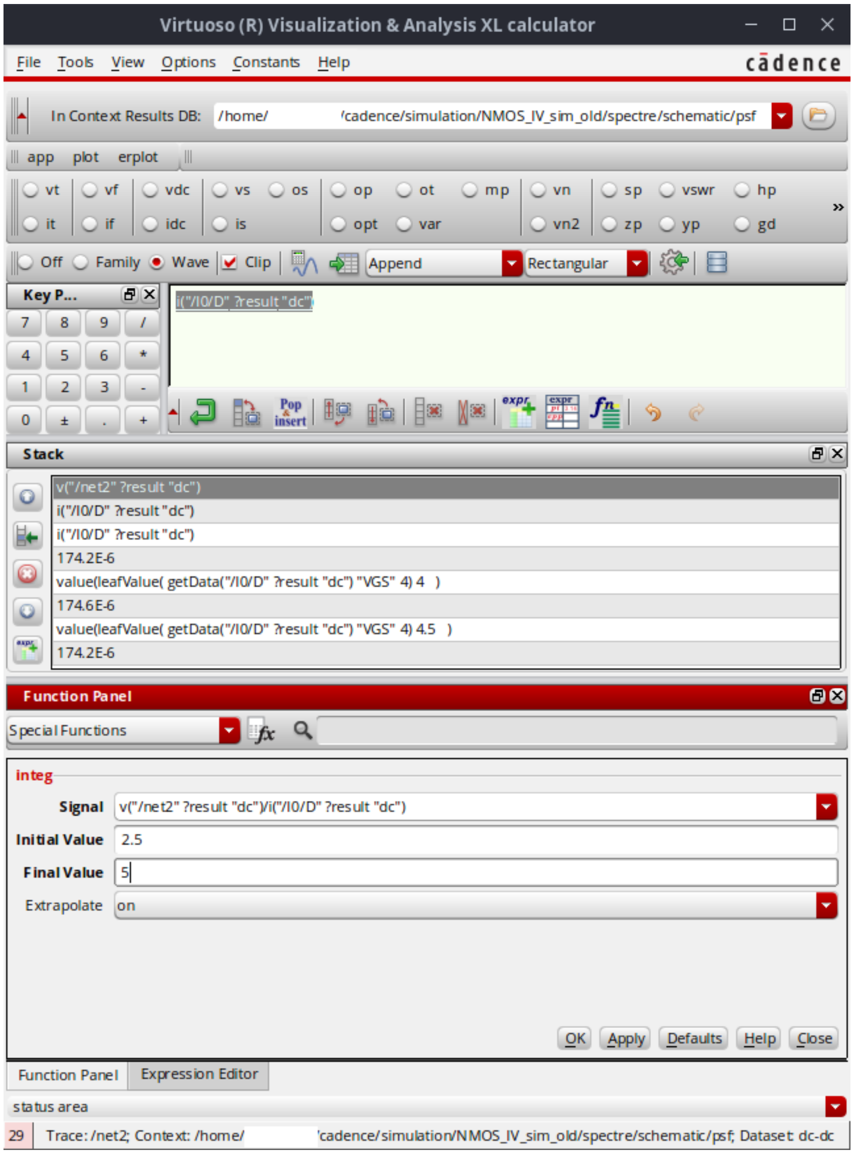

Figure 4: Buffer Windows -Derivative of Output Curve

Then, divide by 2.5(=VDD/2) and evaluate the buffer. (I got R = 8.123E3)

How can we verify this value? We can estimate the delay associated with charging/discharging a capacitor!

Delete the voltage source labeled vds. Place a capacitor instead. Choose the capacitance as 10fF and 5V as the initial condition.

Replace the dc voltage source at the gate with a pulse voltage

Add labels for Vin and Vout. Your sim cell will look like as follows.

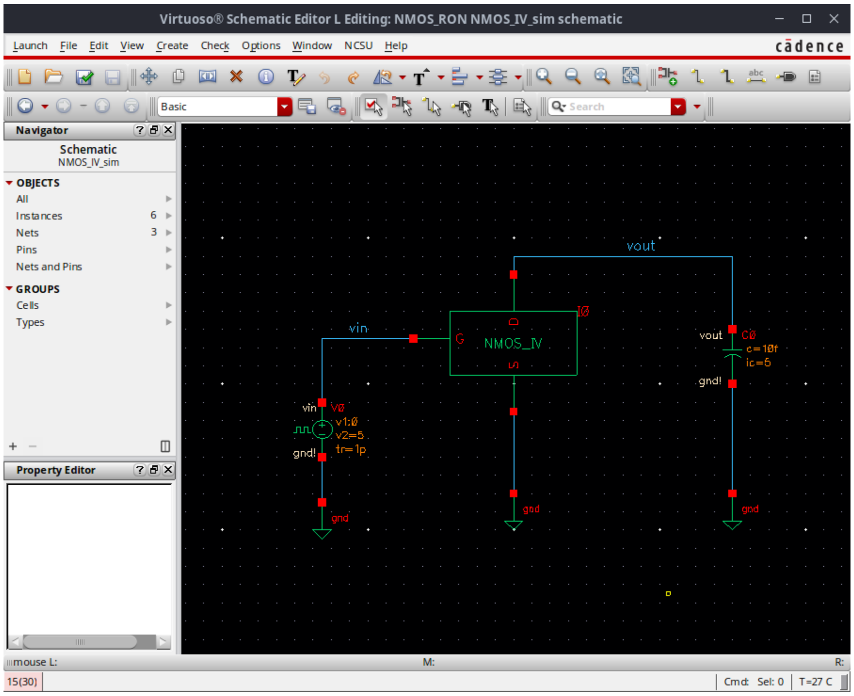

Figure 5: Buffer Windows -Derivative of Output Curve

Open ADE L. Perform a transient analysis for 10 ns. Plot both vin and vout.

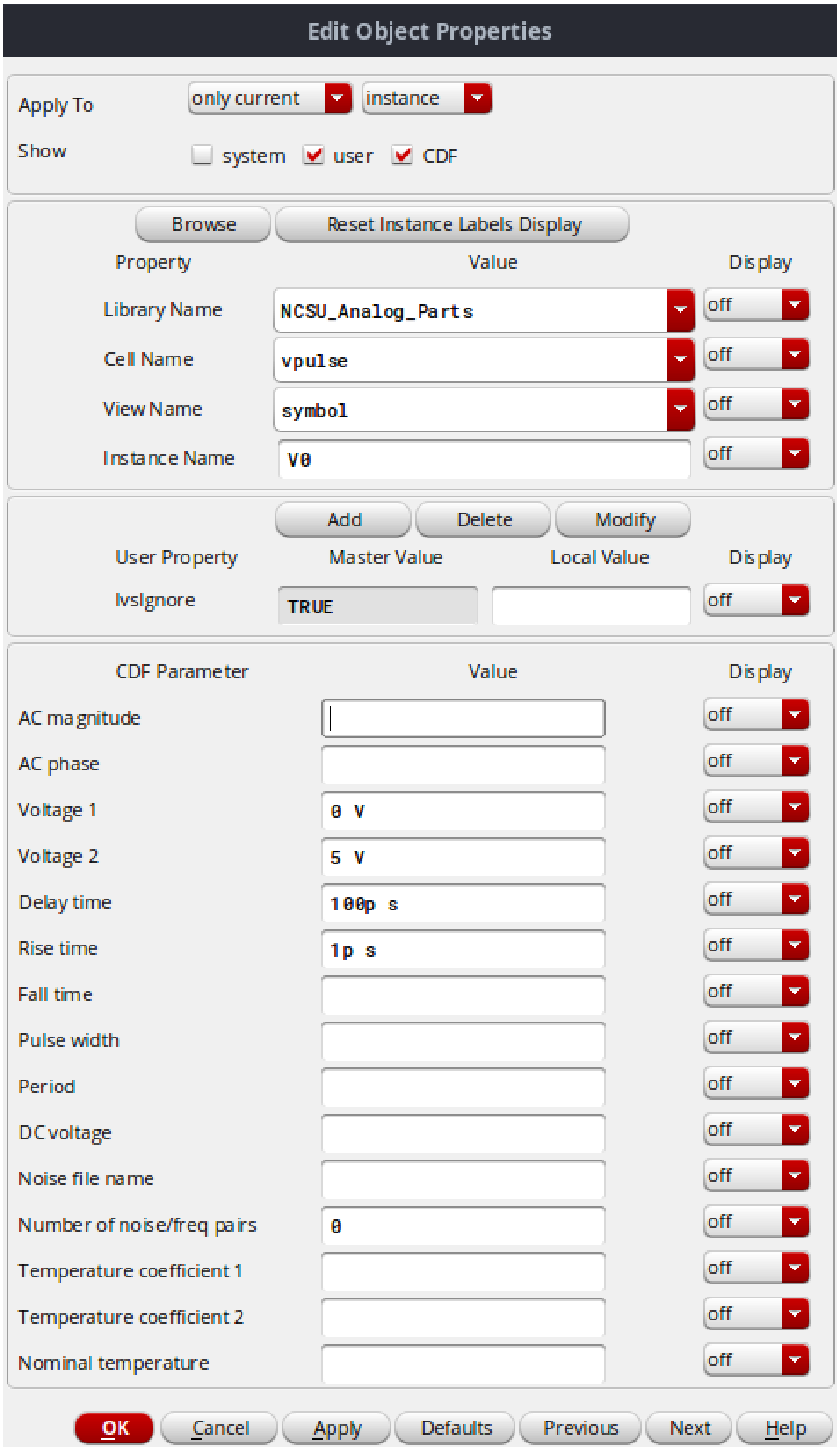

Figure 6: Buffer Windows -Derivative of Output Curve

Locate the points where vin and vout are equal to 2.5 V. Press a and b to place markers. The software will indicate the difference in x and y values. (For me dx= 73.5ps)

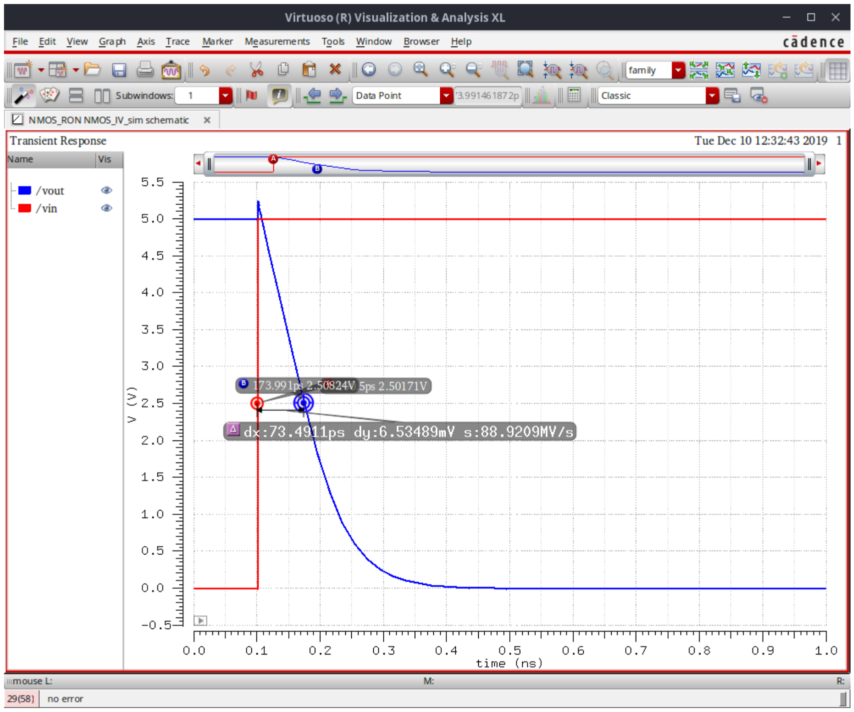

Figure 7: Buffer Windows -Derivative of Output Curve

Theoretically, it should have been

$$t_d = 0.69 * 8.12k * 10f = 56ps$$

To improve accuracy, include the diffusion capacitance as well.