For:sanjay
Printed on:Wed, Sep 2, 1998 13:24:11

Document:lab1
Last saved on:Wed, Sep 2, 1998 13:22:18

Lab1: CMOS Inverter

Introduction

The purpose of this lab is to introduce the student to the CMOS inverter. This lab will also demonstrate how to use HSPICE simulations for analysis

Setup

Read the HSPICE/MWAVES tutorial and modify your environment so that you can use HSPICE and MWAVES. The URL for the tutorial is:

http://www.erc.msstate.edu/~reese/EE4253/mwaves.html

To modify your environment to use the ecad tools make sure the swsetup line contains the following information:

ecad:meta:cadence (i.e. swsetup ecad:meta:cadence)

Then type '*source .cshrc*' at the command prompt.

Make a directory called 'lab1', and copy all files from ~sr2/EE4253/lab1 into it.

Part 1 – DC Analysis of a CMOS inverter

Look at the file 'inverter_dc.sp'. This file demonstrates how to perform a DC analysis on a CMOS circuit. Notice the section of the file labeled 'corner information'. You will see two '.include' statements calling the files. 'tt.cor' and 'trtc.cor'. These files contain the model definitions that you will use throughout the lab. There are some different model files that can be used, but for now just use these.

Look at the file 'inv_sub.sp'. This file contains the circuit elements used to create the inverter. Fig. 1 illustrates the CMOS inverter defined there.

Run the simulation using hspice and observe the measured results. Use MWAVES to produce a plot of the input voltage versus the output voltage.

The initial VDD value is 3.3 volts. Add '.alter' statements to the spice file so that you can measure the switching point for VDD values of 2.5, 2.0, and 1.5 volts. Do not make plots of these results. Just turn in the measured values. Hspice lists the output in files with the .ms* extension (eg .ms0, .ms1 etc) for the DC analysis. Group these files and edit them for turning in the results.

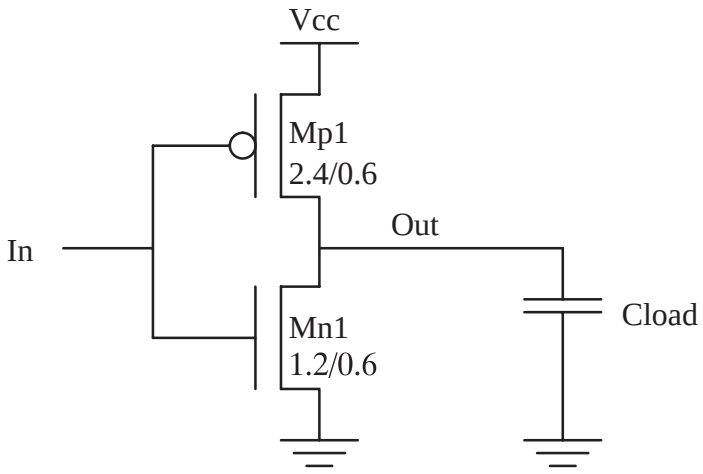

Figure 1. CMOS Inverter

Part2 – AC Analysis of a CMOS inverter

One of the files you copied is called 'inverter_ac.sp'. This file does an AC analysis on a CMOS inverter and measures the high–to–low propagation delay.

Run the simulation using hspice and observe the measured results. Use MWAVES to produce a single plot which shows the input waveform versus time, and output waveform versus time.

Edit the spice file so that the low–to–high propagation delay is measured. This measurement value should be called 'tplh'. You will need to edit the input PWL source so that it produces a falling input waveform. You will also need to edit the .measure statement.

The initial VDD value is 3.3 volts. Add '.alter' statements to the spice file so as to measure 'tplh' for Cload values of 60fF and 150fF for a VDD value of 3.3 v. Do not make plots for these. Just turn in the measured values. Hspice lists the output in files with the .mt* extension (eg. .mt0, .mt1 etc) for the AC analysis. Group these files and edit them for turning in the results.

Part 3 – Another inverter

Modify the file 'inv_sub.sp' such that the pmos transistor pulls the output low, and the nmos pulls the output high. Run the spice file and use MWAVES to obtain the transfer characteristic Vout vs Vin, for the modified inverter. You can use the same stimulus file as used for the DC analysis, but without the '.measure' statements.

To Turn In

1. DC analysis plot
2. New DC analysis spice file (after adding the .alter statements)
3. DC measured results (group all the results and present in a neat tabular form)
4. AC analysis plot
5. New AC analysis spice file (for the 'tplh' measurement)
6. AC measured results (group all results and present in a neat tabular form)
7. Plot of the transfer curve (Vout vs Vin) for Part –3.
8. Answers to the following questions (your answers must be brief and to the point):

    1. In the DC analysis, what is the trend of the switching point as a percentage of VDD as the voltage is scaled down. Would you say it stays 'mostly the same', 'steadily & significantly decreasing', or 'steadily & significantly increasing'?

    2. What is the trend of the propagation delay as the output capacitive load is scaled up? Why does this happen?

    3. Compare the maximum high and minimum low logic output levels obtained for the transfer curves in Part 1 and Part 3. Is there any difference. If so, why?