UNIVERSITY OF CALIFORNIA
College of Engineering
Department of Electrical Engineering and Computer Sciences
Last modified on October 9, 2005 by Dejan Markovic (dejan@eecs.berkeley.edu)

Dejan Markovic

EECS 141
Fall 2005

Project 1 Background & HSPICE Tips

Problem 1. Delay and Energy Capacitance

In this problem we have to determine linear equivalent of the inverter input capacitance using HSPICE. Simulation setup is shown in Fig. 1. The transistor sizes in microns are marked on schematics. All transistors are minimum length, L=0.25µm. Dotted inverter at the output is used to suppress excessive Miller kick-back effect on the input capacitance of the previous stage.

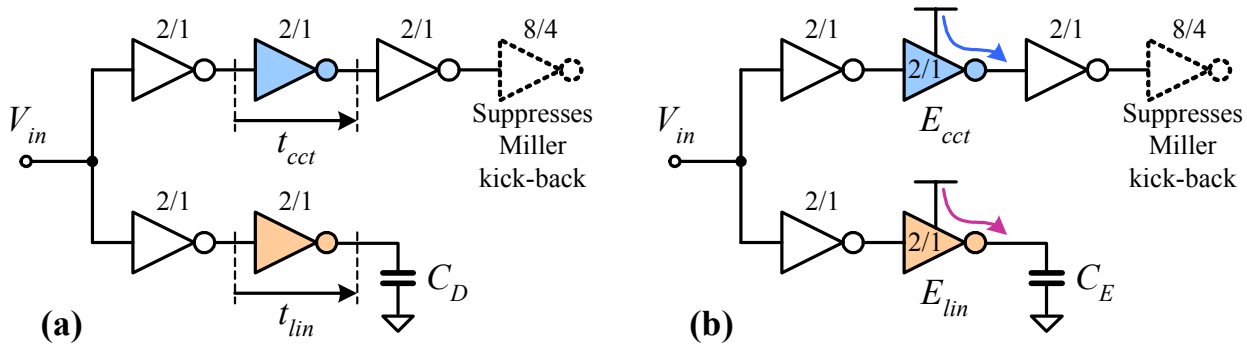


Figure 1: Equivalent delay and energy capacitances

a) Determine capacitance C_D from Fig. 1a in such a way as to match propagation delays of the inverter loaded with another inverter, delay $t_{cct} = (t_{LHcct} + t_{HLcct})/2$, and inverter loaded with equivalent linear capacitance, delay $t_{lin} = (t_{LHlin} + t_{HLlin})/2$.

b) Determine capacitance C_E in such a way as to match energy of the two shaded inverters in Fig. 1b over the time interval from 10ns to 30ns. Calculate energy by integrating current through V_{DD} (see HSPICE tips below).

c) Are C_D and C_E different? Comment your results.

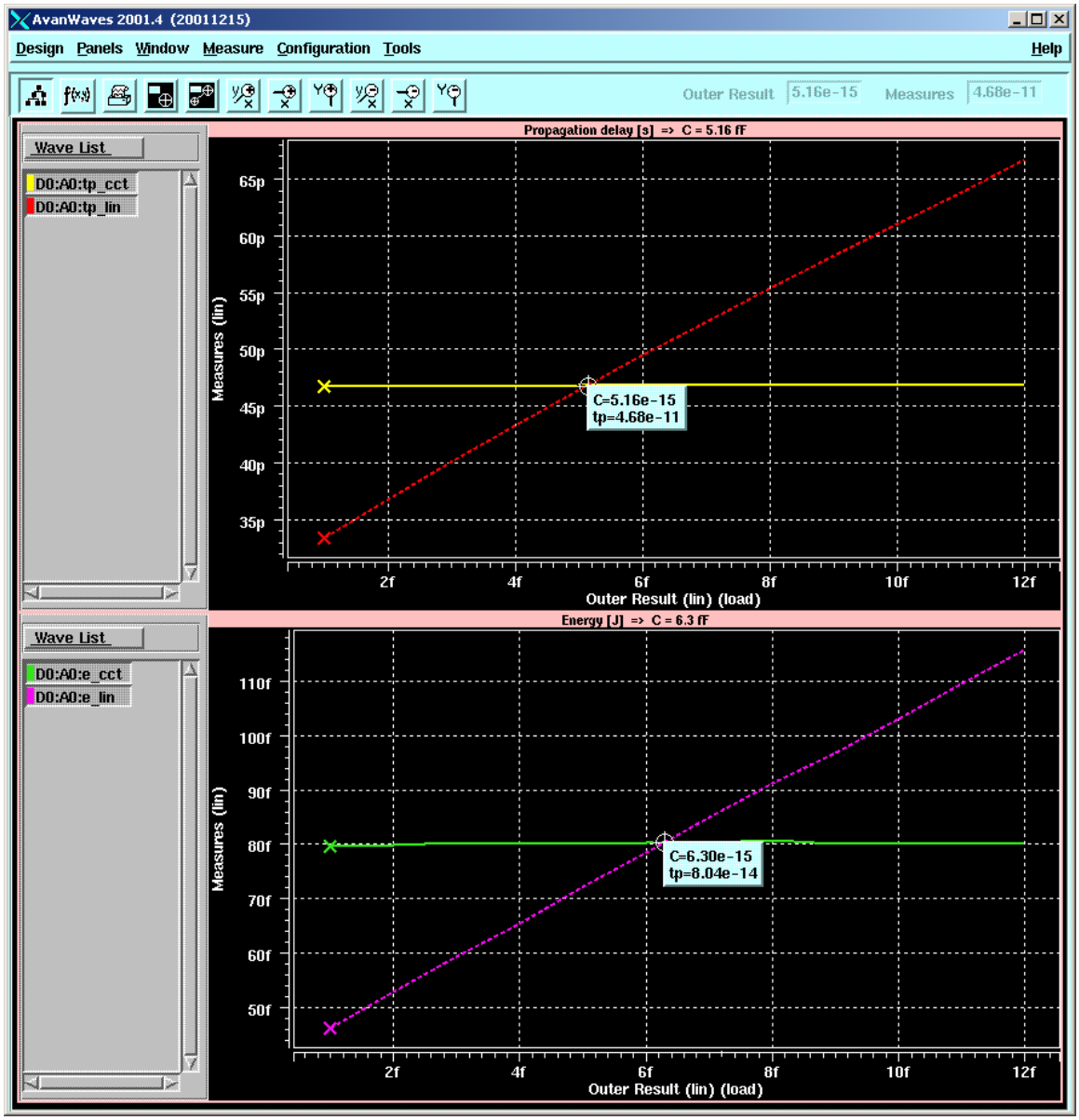
Assumptions: V_{DD}=2.5V, V_{in} is a pulse voltage taking values 0 and V_{DD} with rise and fall times of 100ps, duty cycle of 10ns and period of 20ns. For exact analysis, include AD=AS='0.66u*w', PD=PS='2*0.66u+w' in your transistor models (see HSPICE manual). Use **TT** process corner.

Problem 1. Delay and Power Capacitances (Solution)

The result of HSPICE simulations is shown below:

a) Equivalent delay capacitance $C_D = 5.2$ fF

b) Equivalent energy capacitance $C_E = 6.3$ fF



c) **Equivalent energy and delay capacitances are not equal due to nonlinear capacitance-voltage relationship. Therefore, for accurate hand analysis of delay and energy, we need to extract two capacitances to work with.**

Appendix: Linear capacitance that balances t_{pLH}: $C_{DLH} = 5.47$ fF
Linear capacitance that balances t_{pHL}: $C_{DHL} = 4.80$ fF

HSPICE Netlist for Problem 1

```
*****************************
*    D. Markovic, ED Caps    *
*     EECS141, Fall 2005     *
*****************************

**** Parameters ****
.global Vdd 0 Vss
.param vdd=2.5
.param vss=0
.param lambda='0.125u'
.param def_w='2*lambda'
.param def_l='2*lambda'
.param load='1fF'
.temp 25

**** Voltage sources ****
Vdd   Vdd   0     'vdd'
Vcct  Vcct  0     'vdd'
Vlin  Vlin  0     'vdd'
Vss   Vss   0     'vss'
Vin   in    0     Pulse (0 vdd 5n 100p 100p 10n 20n)

Cload t1    0     'load'

**** Devices ****
.subckt NTRAN d g s b wn='def_w' ln='def_l'
m1 d g s b nmos w='wn' l='ln'
+AD='0.66u*wn' AS='0.66u*wn' PD='2*0.66u+wn' PS='2*0.66u+wn'
.ends

.subckt PTRAN d g s b wp='def_w' lp='def_l'
m1 d g s b pmos w='wp' l='lp'
+AD='0.66u*wp' AS='0.66u*wp' PD='2*0.66u+wp' PS='2*0.66u+wp'
.ends

.subckt INV in out vgnd vpwr wni='def_w' wpi='wni*ratio'
x1 out in vgnd vgnd NTRAN wn='wni'
x2 out in vpwr vpwr PTRAN wp='wpi'
.ends INV

**** Device models ****
.lib '/home/aa/grad/dejan/EE141/Models/g25.mod' TT

**** Test circuit ****
Xinv1   in  d0  Vss  Vdd  INV  wni=1u wpi=2u
Xinv2   d0  d1  Vss  Vcct  INV  wni=1u wpi=2u
Xinv3   d1  d2  Vss  Vdd  INV  wni=1u wpi=2u
Xinv4   d2  d3  Vss  Vdd  INV  wni=4u wpi=8u

Xinvt1  in  t0  Vss  Vdd  INV  wni=1u wpi=2u
Xinvt2  t0  t1  Vss  Vlin  INV  wni=1u wpi=2u

**** Control Section ****
.Options POST=2 Accurate nomod
```

Define "load" as parameter. See TRAN statement below on how to SWEEP the load.

Defining a subcircuit

Subcircuit call

```
**** Analysis ****
.TRAN   0.5n  35n   SWEEP  load  1fF 12fF 1fF
.op

**** Measurements ****
.MEASURE TRAN tpLH_cct trig V(d0) val='vdd/2' fall=2 targ V(d1) val='vdd/2' rise=2
.MEASURE TRAN tpHL_cct trig V(d0) val='vdd/2' rise=1 targ V(d1) val='vdd/2' fall=1
.MEASURE tp_cct Param='(tpLH_cct+tpHL_cct)/2'

.MEASURE TRAN tpLH_lin trig V(t0) val='vdd/2' fall=2 targ V(t1) val='vdd/2' rise=2
.MEASURE TRAN tpHL_lin trig V(t0) val='vdd/2' rise=1 targ V(t1) val='vdd/2' fall=1
.MEASURE tp_lin Param='(tpLH_lin+tpHL_lin)/2'

*.MEASURE TRAN   Pcct_bad  AVG  P(Xinv2)  FROM=10n TO=30n
*.MEASURE TRAN   Plin_bad  AVG  P(Xinvt2)  FROM=10n TO=30n

.MEASURE TRAN   Icct  INTEG  I(Vcct)  FROM=10n TO=30n
.MEASURE TRAN   Ilin  INTEG  I(Vlin)  From=10n TO=30n

.MEASURE   E_cct  Param='-Icct*vdd'
.MEASURE   E_lin  Param='-Ilin*vdd'

.END
```

SWEEPING the load: (one TRAN analysis for each value of the load)

Do not use Power command in HSPICE. It is inaccurate!

Integrate current instead. Better accuracy!

Problem 2. Delay Model Calibration

You would like to find the self-loaded (i.e. no external capacitive load, just the internal capacitances) delay of an inverter fabricated in our not-so-technologically-advanced 2.5V 0.25μm process. However, since you are an EE141 expert, you know that you cannot simply measure this delay in HSPICE with a single unloaded inverter due to problems with Miller multiplication and capacitive coupling. Luckily, you are also an HSPICE expert. So first, you set up a SPICE deck to simulate the following multi-stage inverter chain (with fanout ***f***):

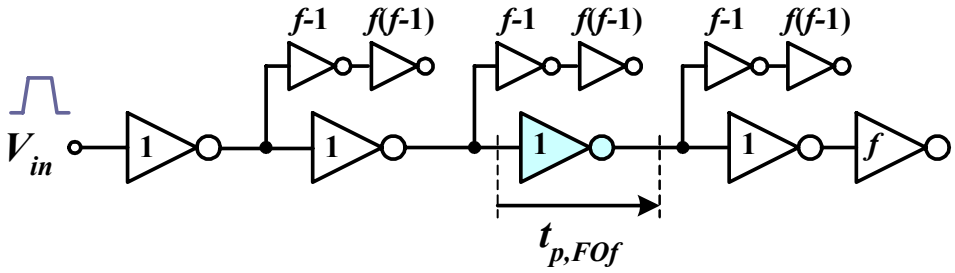


Figure 2: Test setup for delay versus fanout simulation.
Assume pulse waveform at V_{in} (e.g. $t_{rise/fall}$ ~10ps).

Using this circuit, you can get an accurate measurement of the real delay in a circuit by measuring the delay of the third inverter in the chain. The first two stages create a realistic input signal slope to the third inverter. The fourth stage provides an appropriate load for the third stage. You put the last stage just for good measure, to make sure that the Miller effect for the fourth stage is reasonable. Of course, you don't forget the fanout of the load inverter stages either to ensure proper Miller effect on the loading gates.

*Hint: Since there are so many inverters, you know that using the ".SUBCKT" command would make life really easy. You also remember that using the "**m**" tag (multiplicity parameter) when calling a sub-circuit could be helpful.*

In order to get the self-loaded delay, you have to do the following:

a) Use HSPICE to find the *average* propagation delay $t_p = (t_{pLH} + t_{pHL})/2$ for an inverter in this process for a fanout of 2 to 10 in increments of 1. Simply measure the delay of the third inverter in this chain. **Plot the propagation delay as a function of the fanout.**

b) In your plot, the points should fall in a straight line. Find the best-fit line through the data. This allows you to extrapolate the delay for a fanout of 0 (intercept with the ***x***-axis). **From the intercept, find self-loaded delay t_{p0}.**

c) In your plot, the slope of the line tells you about the additional delay per fanout. Using this information, **determine parameter γ in the formula $t_p = t_{p0}(1+f/\gamma)$.**

Problem 2. Delay Model Calibration (Solution)

a) **Propagation delay as a function of fanout** is shown in Fig. 2.

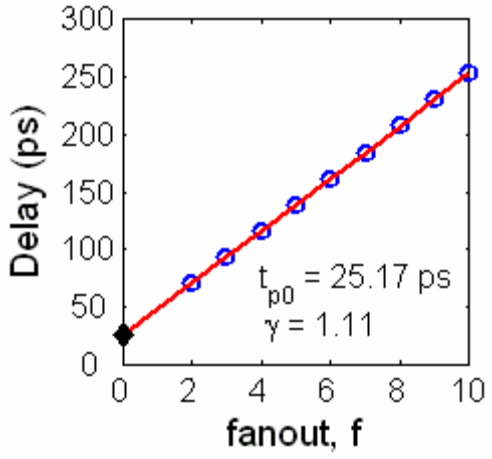


Figure 2. Inverter propagation delay vs. fanout.

b) **Self-loaded delay t_{p0}.** This parameter is found by extrapolating $t_p(f)$ curve to f = 0.

t_{p0} = 25.2ps

c) **Parameter γ in the formula $t_p = t_{p0}(1+f/\gamma)$.** Slope of $t_p(f)$ curve is $1/\gamma$.

γ = 1.11

HSPICE Netlist:

```
************************************
*    D. Markovic, Delay Calibration   *
*        Project 1, Problem 2         *
*         EECS141, Fall 2005          *
************************************

**** Parameters ****
.global Vdd 0 Vss
.param vdd=2.5
.param vss=0
.param lambda='0.125u'
.param pnratio=2
.param def_w='2*lambda'
.param def_l='2*lambda'
.param f='2'
.temp 25

**** Voltage sources ****
Vdd     Vdd     0       'vdd'
Vss     Vss     0       'vss'
Vin in     0      Pulse (0 vdd 5n 10p 10p 5n 10n)

**** Devices ****
.subckt NTRAN d g s b wn='def_w' ln='def_l'
m1 d g s b nmos w='wn' l='ln'
```

```
+ AD='0.66u*wn' AS='0.66u*wn'
+ PD='2*0.66u+wn' PS='2*0.66u+wn'
.ends

.subckt PTRAN d g s b wp='def_w' lp='def_l'
m1 d g s b pmos w='wp' l='lp'
+ AD='0.66u*wp' AS='0.66u*wp'
+ PD='2*0.66u+wp' PS='2*0.66u+wp'
.ends

.subckt INV in out vgnd vpwr wni='def_w' wpi='def_w*pnratio'
x1 out in vgnd vgnd NTRAN wn='wni'
x2 out in vpwr vpwr PTRAN wp='wpi'
.ends INV

**** Device models ****
.lib '/home/aa/grad/dejan/EE141/Models/g25.mod' TT

**** Test circuit ****
Xinv1     in  d1  Vss  Vdd  INV  wni=1u wpi=2u m=1
Xinv2     d1  d2  Vss  Vdd  INV  wni=1u wpi=2u m=1
Xinv3   d2  d3  Vss  Vdd  INV  wni=1u wpi=2u m=1
Xinv4   d3  d4  Vss  Vdd  INV  wni=1u wpi=2u m=1
Xinv5   d4  d5  Vss  Vdd  INV  wni=1u wpi=2u m=1

Xinv11  d1  d11  Vss  Vdd  INV  wni=1u wpi=2u  m='f-1'
Xinv12  d11 d12  Vss  Vdd  INV  wni=1u wpi=2u  m='f*(f-1)'
Xinv21  d2  d21  Vss  Vdd  INV  wni=1u wpi=2u  m='f-1'
Xinv22  d21 d22  Vss  Vdd  INV  wni=1u wpi=2u  m='f*(f-1)'
Xinv31  d3  d31  Vss  Vdd  INV  wni=1u wpi=2u  m='f-1'
Xinv32  d31 d32  Vss  Vdd  INV  wni=1u wpi=2u  m='f*(f-1)'

**** Control Section ****
.Options POST=2 Accurate nomod

**** Analysis ****
.TRAN   0.5n  35n   SWEEP  f  2 10 1

**** Measurements ****
.MEASURE TRAN tpHL trig V(d2) val='vdd/2' fall=2 targ V(d3) val='vdd/2' rise=2
.MEASURE TRAN tpLH trig V(d2) val='vdd/2' rise=2 targ V(d3) val='vdd/2' fall=2
.MEASURE tp Param='(tpLH+tpHL)/2'

.END
```