Lecture 8: SPICE Simulation
Outline

- Introduction to SPICE
- DC Analysis
- Transient Analysis
- Subcircuits
- Optimization
- Power Measurement
- Logical Effort Characterization
Introduction to SPICE

- **Simulation Program with Integrated Circuit Emphasis**
  - Developed in 1970’s at Berkeley
  - Many commercial versions are available
  - HSPICE is a robust industry standard
    - Has many enhancements that we will use
- Written in FORTRAN for punch-card machines
  - Circuits elements are called *cards*
  - Complete description is called a SPICE *deck*
Writing Spice Decks

- Writing a SPICE deck is like writing a good program
  - Plan: sketch schematic on paper or in editor
    - Modify existing decks whenever possible
  - Code: strive for clarity
    - Start with name, email, date, purpose
    - Generously comment
  - Test:
    - Predict what results should be
    - Compare with actual
    - Garbage In, Garbage Out!
Example: RC Circuit

* rc.sp
* David_Harris@hmc.edu 2/2/03
* Find the response of RC circuit to rising input

* Parameters and models
*-------------------------------------------------
.option post
*-------------------------------------------------

* Simulation netlist
*-------------------------------------------------
Vin in gnd pw1 0ps 0 100ps 0 150ps 1.0 1ns 1.0
R1 in out 2k
C1 out gnd 100f

* Stimulus
*-------------------------------------------------
.tran 20ps 1ns
.plot v(in) v(out)
.end

R1 = 2KΩ
R1 = 2KΩ

C1 = 100fF
C1 = 100fF

Vin
Vin

Vout
Vout

8: SPICE Simulation CMOS VLSI Design 4th Ed.
Result (Graphical)
Sources

- **DC Source**
  
  \[
  \text{Vdd vdd gnd 2.5}
  \]

- **Piecewise Linear Source**
  
  \[
  \text{Vin in gnd pw1 0ps 0 100ps 0 150ps 1.0 1ns 1.0}
  \]

- **Pulsed Source**
  
  \[
  \text{Vck clk gnd PULSE 0 1.0 0ps 100ps 100ps 300ps 800ps}
  \]

\[
\text{PULSE v1 v2 td tr tf pw per}
\]
## SPICE Elements

<table>
<thead>
<tr>
<th>Letter</th>
<th>Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Resistor</td>
</tr>
<tr>
<td>C</td>
<td>Capacitor</td>
</tr>
<tr>
<td>L</td>
<td>Inductor</td>
</tr>
<tr>
<td>K</td>
<td>Mutual Inductor</td>
</tr>
<tr>
<td>V</td>
<td>Independent voltage source</td>
</tr>
<tr>
<td>I</td>
<td>Independent current source</td>
</tr>
<tr>
<td>M</td>
<td>MOSFET</td>
</tr>
<tr>
<td>D</td>
<td>Diode</td>
</tr>
<tr>
<td>Q</td>
<td>Bipolar transistor</td>
</tr>
<tr>
<td>W</td>
<td>Lossy transmission line</td>
</tr>
<tr>
<td>X</td>
<td>Subcircuit</td>
</tr>
<tr>
<td>E</td>
<td>Voltage-controlled voltage source</td>
</tr>
<tr>
<td>G</td>
<td>Voltage-controlled current source</td>
</tr>
<tr>
<td>H</td>
<td>Current-controlled voltage source</td>
</tr>
<tr>
<td>F</td>
<td>Current-controlled current source</td>
</tr>
</tbody>
</table>
# Units

<table>
<thead>
<tr>
<th>Letter</th>
<th>Unit</th>
<th>Magnitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>atto</td>
<td>$10^{-18}$</td>
</tr>
<tr>
<td>f</td>
<td>fempto</td>
<td>$10^{-15}$</td>
</tr>
<tr>
<td>p</td>
<td>pico</td>
<td>$10^{-12}$</td>
</tr>
<tr>
<td>n</td>
<td>nano</td>
<td>$10^{-9}$</td>
</tr>
<tr>
<td>u</td>
<td>micro</td>
<td>$10^{-6}$</td>
</tr>
<tr>
<td>m</td>
<td>milli</td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>k</td>
<td>kilo</td>
<td>$10^{3}$</td>
</tr>
<tr>
<td>x</td>
<td>mega</td>
<td>$10^{6}$</td>
</tr>
<tr>
<td>g</td>
<td>giga</td>
<td>$10^{9}$</td>
</tr>
</tbody>
</table>

Ex: 100 femtofarad capacitor = 100fF, 100f, 100e-15
DC Analysis

* mosiv.sp

*---------------------------------------------------
* Parameters and models
*---------------------------------------------------
.include '../models/ibm065/models.sp'
.temp 70
.option post

*---------------------------------------------------
* Simulation netlist
*---------------------------------------------------
*nmos
Vgs  g       gnd       0
Vds  d       gnd       0
M1   d       g       gnd       gnd       NMOS       W=100n   L=50n

*---------------------------------------------------
* Stimulus
*---------------------------------------------------
.dc Vds 0 1.0 0.05 SWEEP Vgs 0 1.0 0.2
.end
I-V Characteristics

- nMOS I-V
  - $V_{gs}$ dependence
  - Saturation

![Graph showing I-V characteristics of nMOS with different $V_{gs}$ values]
MOSFET Elements

M element for MOSFET

Mname drain gate source body type
+ W=<width> L=<length>
+ AS=<area source> AD = <area drain>
+ PS=<perimeter source> PD=<perimeter drain>
* inv.sp

* Parameters and models
*-------------------------------------------------
.param SUPPLY=1.0
.option scale=25n
.include './models/ibm065/models.sp'
.temp 70
.option post

* Simulation netlist
*-------------------------------------------------
Vdd   vdd   gnd    'SUPPLY'
Vin   a     gnd    PULSE  0 'SUPPLY'  50ps 0ps 0ps 100ps 200ps
M1    y     a     gnd    gnd    NMOS    W=4    L=2
+ AS=20 PS=18 AD=20 PD=18
M2    y     a     vdd    vdd    PMOS    W=8    L=2
+ AS=40 PS=26 AD=40 PD=26

* Stimulus
*-------------------------------------------------
.tran 0.1ps 80ps
.end
Transient Results

- Unloaded inverter
  - Overshoot
  - Very fast edges

![Graph showing transient results with key parameters]

- $t_f = 2.5 \text{ ps}$
- $t_{pdf} = 3.1 \text{ ps}$
- $t_{pdr} = 3.6 \text{ ps}$
- $t_r = 3.5 \text{ ps}$
Subcircuits

- Declare common elements as subcircuits

```
.subckt inv a y N=4 P=8
M1 y a gnd gnd NMOS W='N' L=2
  + AS='N*5' PS='2*N+10' AD='N*5' PD='2*N+10'
M2 y a vdd vdd PMOS W='P' L=2
  + AS='P*5' PS='2*P+10' AD='P*5' PD='2*P+10'
.ends
```

- Ex: Fanout-of-4 Inverter Delay
  - Reuse inv
  - Shaping
  - Loading

![Diagram of the circuit with nodes X1 to X5 and load resistors 1, 4, 16, 64, 256, 512, 128, and 32, with a shape input and device under test.]
FO4 Inverter Delay

* fo4.sp

* Parameters and models

.PARAM SUPPLY=1.0
.PARAM H=4
.OPTION SCALE=25n
.INCLUDE '../models/ibm065/models.sp'
.TEMP 70
.OPTION POST

* Subcircuits

.GLOBAL VDD GND
.INCLUDE '../lib/inv.sp'

* Simulation netlist

VDD VDD GND 'SUPPLY'
VIN A GND PULSE 0 'SUPPLY' 0PS 20PS 10PS 20PS 120PS 280PS
X1 A B INV * shape input waveform
X2 B C INV M='H' * reshape input waveform
FO4 Inverter Delay Cont.

X3  c  d  inv  M='H**2'  * device under test
X4  d  e  inv  M='H**3'  * load
x5  e  f  inv  M='H**4'  * load on load

* Stimulus

*---------------------------------------------------------------*
.tran 0.1ps 280ps
.measure tpdr
+    TRIG v(c) VAL='SUPPLY/2' FALL=1
+    TARG v(d) VAL='SUPPLY/2' RISE=1
.measure tpdf
+    TRIG v(c) VAL='SUPPLY/2' RISE=1
+    TARG v(d) VAL='SUPPLY/2' FALL=1
.measure tpd param='(tpdr+tpdf)/2'
.measure trise
+    TRIG v(d) VAL='0.2*SUPPLY' RISE=1
+    TARG v(d) VAL='0.8*SUPPLY' RISE=1
.measure tfall
+    TRIG v(d) VAL='0.8*SUPPLY' FALL=1
+    TARG v(d) VAL='0.2*SUPPLY' FALL=1
.end
FO4 Results

\[ V(t) \] vs. \[ t \] graph showing waveforms labeled a through f with timing parameters: \[ t_{pdf} = 16 \text{ ps} \] and \[ t_{pdr} = 18 \text{ ps} \].
Optimization

- HSPICE can automatically adjust parameters
  - Seek value that optimizes some measurement
- Example: Best P/N ratio
  - We’ve assumed 2:1 gives equal rise/fall delays
  - But we see rise is actually slower than fall
  - What P/N ratio gives equal delays?
- Strategies
  - (1) run a bunch of sims with different P size
  - (2) let HSPICE optimizer do it for us
P/N Optimization

* fo4opt.sp

* Parameters and models

.PARAM SUPPLY=1.0
.OPTION SCALE=25n
.INCLUDE '/models/ibm065/models.sp'
.TEMP 70
.OPTION POST

* Subcircuits

.GLOBAL VDD GND
.INCLUDE '/lib/inv.sp'

* Simulation netlist

<table>
<thead>
<tr>
<th>Vdd</th>
<th>vdd</th>
<th>gnd</th>
<th>'SUPPLY'</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vin</td>
<td>a</td>
<td>gnd</td>
<td>PULSE</td>
</tr>
<tr>
<td>X1</td>
<td>a</td>
<td>b</td>
<td>inv</td>
</tr>
<tr>
<td>X2</td>
<td>b</td>
<td>c</td>
<td>inv</td>
</tr>
<tr>
<td>X3</td>
<td>c</td>
<td>d</td>
<td>inv</td>
</tr>
</tbody>
</table>
P/N Optimization

X4  d  e  inv  P='P1'  M=64  * load
X5  e  f  inv  P='P1'  M=256  * load on load

* Optimization setup
*------------------------------------------------------------
.param P1=optrange(8,4,16)  * search from 4 to 16, guess 8
.model optmod opt itropt=30  * maximum of 30 iterations
.measure bestratio param='P1/4'  * compute best P/N ratio

* Stimulus
*------------------------------------------------------------
.tran 0.1ps 280ps SWEEP OPTIMIZE=optrange RESULTS=diff MODEL=optmod
.measure tpdr  * rising propagation delay
+    TRIG v(c) VAL='SUPPLY/2' FALL=1
+    TARG v(d)    VAL='SUPPLY/2' RISE=1
.measure tpdf  * falling propagation delay
+    TRIG v(c)    VAL='SUPPLY/2' RISE=1
+    TARG v(d)    VAL='SUPPLY/2' FALL=1
.measure tpd param='(tpdr+tpdf)/2' goal=0  * average prop delay
.measure diff param='tpdr-tpdf' goal = 0  * diff between delays
.end
P/N Results

- P/N ratio for equal delay is 2.9:1
  - $t_{pd} = t_{pdr} = t_{pdf} = 17.9$ ps (slower than 2:1 ratio)
  - Big pMOS transistors waste power too
  - Seldom design for exactly equal delays
- What ratio gives lowest average delay?

```plaintext
.tran 1ps 1000ps SWEEP OPTIMIZE=optrange RESULTS=tpd MODEL=optmod
```

- P/N ratio of 1.8:1
  - $t_{pdr} = 18.8$ ps, $t_{pdf} = 15.2$ ps, $t_{pd} = 17.0$ ps
- P/N ratios of 1.5:1 – 2.2:1 gives $t_{pd} < 17.2$ ps
Power Measurement

- HSPICE can measure power
  - Instantaneous $P(t)$
  - Or average $P$ over some interval

```plaintext
.print P(vdd)
.measure pwr AVG P(vdd) FROM=0ns TO=10ns
```

- Power in single gate
  - Connect to separate $V_{DD}$ supply
  - Be careful about input power
Logical Effort

- Logical effort can be measured from simulation
  - As with FO4 inverter, shape input, load output

![Logical Effort Diagram]

- Shape input
- Device Under Test
- Load
- Load on Load

X1
- M=1
- M=h

X2
- M=h

X3
- M=h^2

X4
- M=h^3

X5
- M=h^4

a
b
M=1
M=h
M=h
M=h^3
M=h^4
Logical Effort Plots

- Plot $t_{pd}$ vs. $h$
  - Normalize by $\tau$
  - $y$-intercept is parasitic delay
  - Slope is logical effort
- Delay fits straight line very well in any process as long as input slope is consistent
Logical Effort Data

For NAND gates in IBM 65 nm process:

<table>
<thead>
<tr>
<th># of inputs</th>
<th>Input</th>
<th>Rising Logical Effort $g_u$</th>
<th>Falling Logical Effort $g_d$</th>
<th>Average Logical Effort $g$</th>
<th>Rising Parasitic Delay $p_u$</th>
<th>Falling Parasitic Delay $p_d$</th>
<th>Average Parasitic Delay $p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>A</td>
<td>1.40</td>
<td>1.12</td>
<td>1.26</td>
<td>2.46</td>
<td>2.48</td>
<td>2.47</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>1.31</td>
<td>1.16</td>
<td>1.24</td>
<td>1.97</td>
<td>1.82</td>
<td>1.89</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>1.59</td>
<td>1.38</td>
<td>1.48</td>
<td>3.05</td>
<td>2.43</td>
<td>2.74</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>1.90</td>
<td>1.59</td>
<td>1.75</td>
<td>4.04</td>
<td>2.93</td>
<td>3.49</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>2.15</td>
<td>1.42</td>
<td>1.78</td>
<td>7.63</td>
<td>5.94</td>
<td>6.79</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>2.09</td>
<td>1.48</td>
<td>1.78</td>
<td>6.67</td>
<td>5.37</td>
<td>6.02</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.08</td>
<td>1.53</td>
<td>1.80</td>
<td>5.32</td>
<td>4.51</td>
<td>4.91</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>1.90</td>
<td>1.59</td>
<td>1.75</td>
<td>4.04</td>
<td>2.93</td>
<td>3.49</td>
</tr>
</tbody>
</table>

Notes:

- Parasitic delay is greater for outer input
- Average logical effort is better than estimated
## Comparison

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Orbit</th>
<th>HP</th>
<th>AMI</th>
<th>AMI</th>
<th>TSMC</th>
<th>TSMC</th>
<th>TSMC</th>
<th>IBM</th>
<th>IBM</th>
<th>IBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>MOSIS</td>
<td>IBM</td>
<td>IBM</td>
<td>IBM</td>
</tr>
<tr>
<td>Feature Size $f$</td>
<td>nm</td>
<td>2000</td>
<td>800</td>
<td>600</td>
<td>600</td>
<td>350</td>
<td>250</td>
<td>180</td>
<td>130</td>
<td>90</td>
</tr>
<tr>
<td>$V_{DD}$</td>
<td>V</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3.3</td>
<td>3.3</td>
<td>2.5</td>
<td>1.8</td>
<td>1.2</td>
<td>1.0</td>
</tr>
<tr>
<td>FO4 Delay</td>
<td>ps</td>
<td>856</td>
<td>297</td>
<td>230</td>
<td>312</td>
<td>210</td>
<td>153</td>
<td>75.6</td>
<td>46.0</td>
<td>37.3</td>
</tr>
<tr>
<td>$\tau$</td>
<td>ps</td>
<td>170</td>
<td>59</td>
<td>45</td>
<td>60</td>
<td>40</td>
<td>30</td>
<td>15</td>
<td>9.0</td>
<td>7.4</td>
</tr>
</tbody>
</table>

### Logical Effort

| Inverter | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| NAND2    | 1.13 | 1.07 | 1.05 | 1.08 | 1.12 | 1.12 | 1.14 | 1.16 | 1.20 | 1.26 |
| NAND3    | 1.32 | 1.21 | 1.19 | 1.24 | 1.29 | 1.29 | 1.31 | 1.35 | 1.41 | 1.51 |
| NAND4    | 1.53 | 1.37 | 1.36 | 1.42 | 1.47 | 1.47 | 1.50 | 1.55 | 1.62 | 1.78 |
| NOR2     | 1.57 | 1.59 | 1.58 | 1.60 | 1.52 | 1.50 | 1.57 | 1.56 | 1.56 | 1.50 |
| NOR3     | 2.16 | 2.23 | 2.23 | 2.30 | 2.07 | 2.02 | 2.00 | 2.12 | 2.08 | 1.96 |
| NOR4     | 2.76 | 2.92 | 2.96 | 3.09 | 2.62 | 2.52 | 2.53 | 2.70 | 2.60 | 2.43 |

### Parasitic Delay

| Inverter | 1.08 | 1.05 | 1.18 | 1.25 | 1.33 | 1.18 | 1.03 | 1.16 | 1.07 | 1.20 |
| NAND2    | 1.87 | 1.85 | 1.92 | 2.10 | 2.28 | 2.07 | 1.90 | 2.29 | 2.25 | 2.47 |
| NAND3    | 3.34 | 3.30 | 3.40 | 3.79 | 4.15 | 3.65 | 3.51 | 4.14 | 4.10 | 4.44 |
| NAND4    | 4.98 | 5.12 | 5.22 | 5.78 | 6.30 | 5.47 | 5.52 | 6.39 | 6.39 | 6.79 |
| NOR2     | 2.86 | 2.91 | 3.29 | 3.56 | 3.52 | 2.95 | 2.85 | 3.35 | 3.01 | 3.29 |
| NOR3     | 5.65 | 6.05 | 7.02 | 7.70 | 6.89 | 5.61 | 5.57 | 6.59 | 5.76 | 6.35 |
| NOR4     | 9.11 | 10.3 | 12.4 | 13.9 | 11.0 | 8.76 | 8.95 | 10.54 | 9.11 | 10.16 |