

Lecture 10

SPICE Simulation

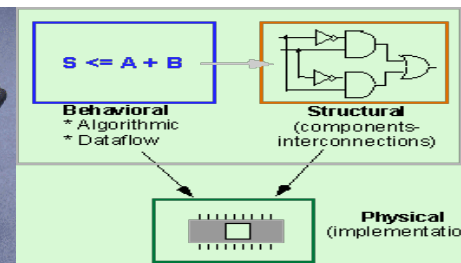
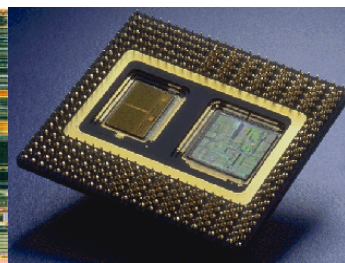
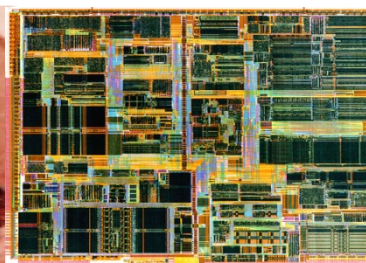
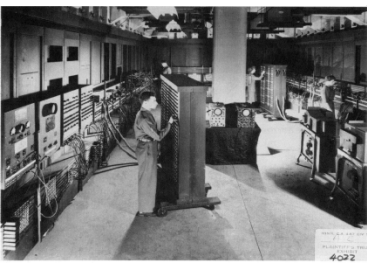
Bryan Ackland

Department of Electrical and Computer Engineering

Stevens Institute of Technology

Hoboken, NJ 07030

Adapted from Lecture Notes, David Mahoney Harris CMOS VLSI Design



Simulation of VLSI Circuits

- Complex circuits cannot be analyzed using analytic expressions for transistor behavior
 - simple models useful for estimation to compare competing approaches
 - modern transistors are highly non-linear devices that deviate significantly from first-order models (e.g. Shockley)
 - need simulation to confidently verify correct operation and performance
 - over range of process, temperature & voltage variations
- Simulation at different levels of abstraction:
 - Process simulation e.g. SUPREME
 - Device simulation e.g. PROPHET
 - Circuit simulation e.g. SPICE
 - Logic Simulation e.g. ModelSim, Isim (VHDL, Verilog)
 - Architectural Simulation (processor specific)
 - Algorithmic Simulation (MATLAB)

Introduction to SPICE

- **S**imulation **P**rogram with **I**ntegrated **C**ircuit **E**mphasis
 - Developed in 1970's at Berkeley
 - Many commercial versions are available
 - HSPICE is a robust industry standard
 - PSpICE is PC based – also widely used
 - LTSpICE free version from Linear Technology
 - WinSpice is another Windows based version
- Originally written in FORTRAN for punch-card machines
 - Circuits elements are called *cards*
 - Complete description is called a SPICE *deck*

Structure of a SPICE Deck

A SPICE deck may contain:

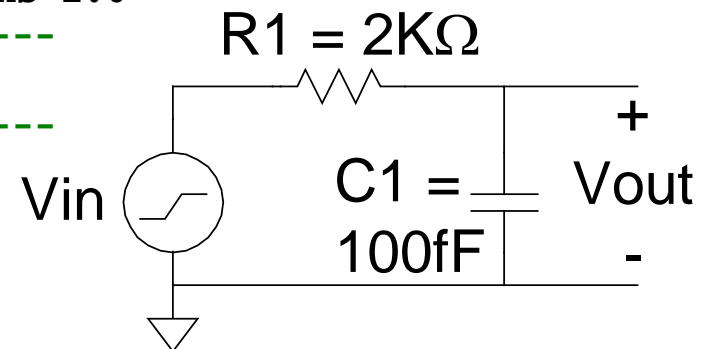
- Circuit components
 - netlist of resistors, capacitors, transistors, subcircuits etc.
 - netlist can be
 - handwritten
 - extracted from layout
 - derived from schematic
- Sources
 - voltage sources, current sources etc. to drive circuit
- Device models
 - transistor models, resistor models, diode models etc.
- SPICE commands
 - initial conditions, type of analysis, plot specification etc.

Manually Writing a SPICE Deck

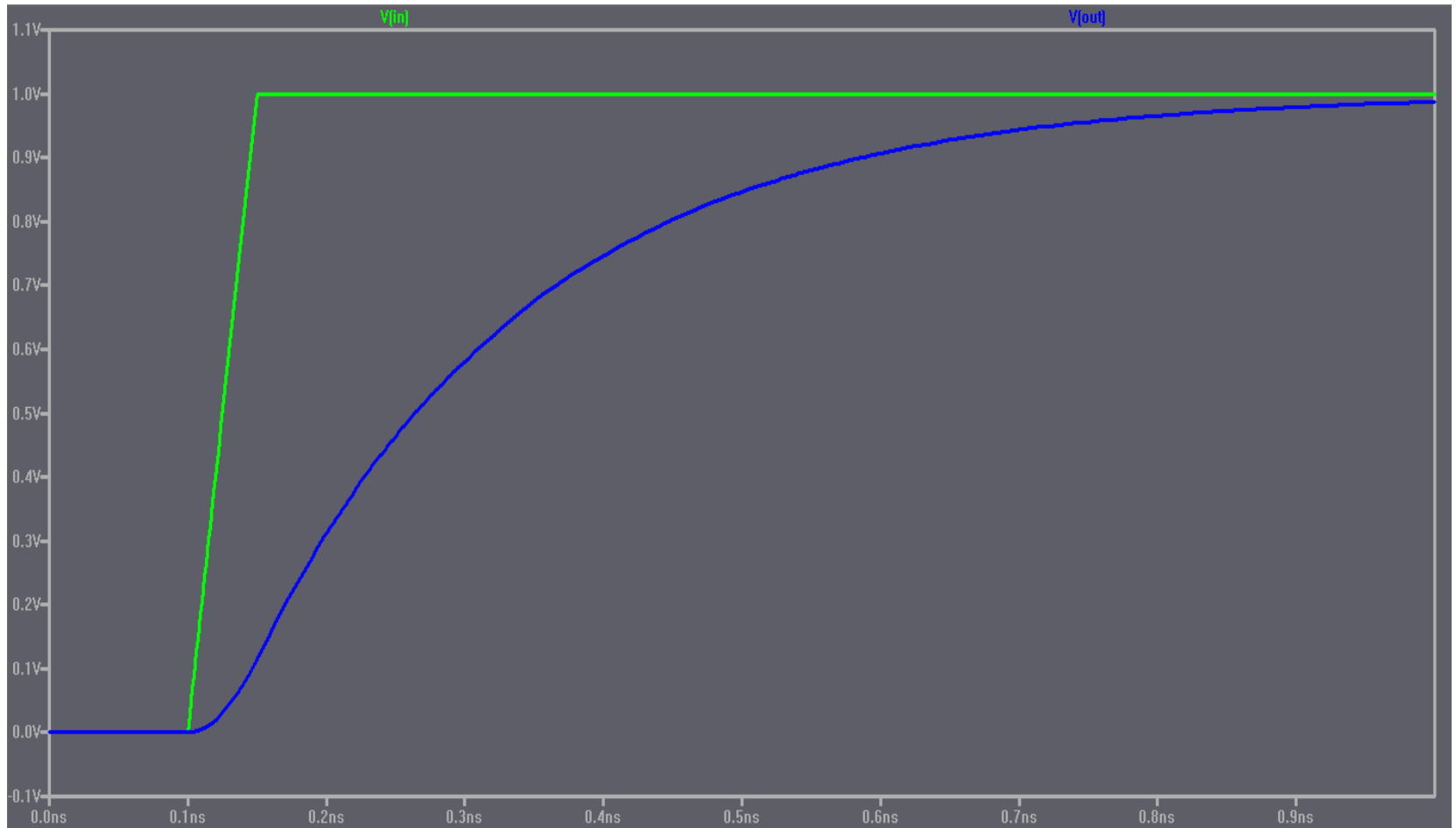
- Writing a SPICE deck is like writing a good program
 - Plan: sketch schematic on paper or in editor
 - Modify existing decks whenever possible
 - Code: strive for clarity
 - Start with name, email, date, purpose
 - Generously comment
 - Test:
 - Predict what results should be
 - Compare with actual
 - Understand any differences before you proceed!

Example: RC Filter

```
* RC_Filter.cir
* backland@stevens.edu 4/2/12
* Find the response of RC circuit to rising input
*
*-----
* Parameters and options
*-----
.options reltol=.01 gshunt=1e-9
*-----
* Simulation netlist
*-----
R1      in out 2k
C1      out gnd 100f
*-----
* Sources
*-----
Vin     in gnd  pw1 0ps 0 100ps 0 150ps 1.0 1ns 1.0
*-----
* Commands
*-----
.tran 20ps 1ns
.plot v(in) v(out) *ignored by LTspice
.end
```



RC_Filter LTspice Result



Sources

❑ DC Source

vdd vdd gnd 2.5

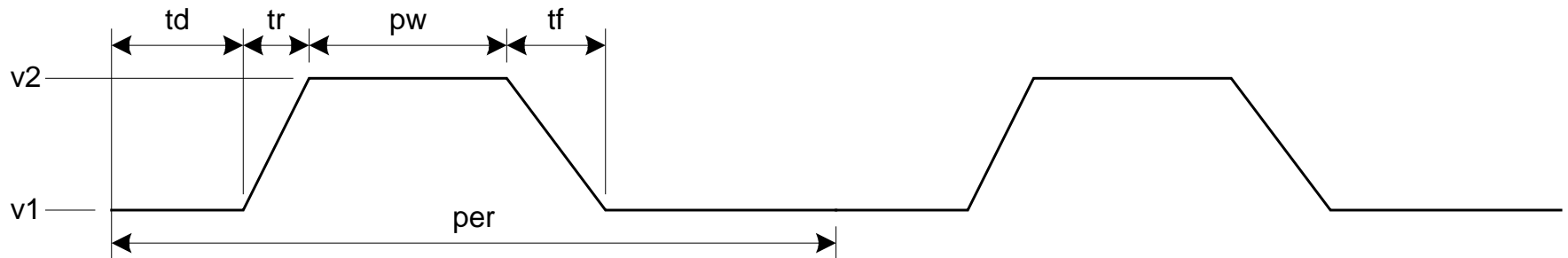
❑ Piecewise Linear Source

vin in gnd pw1 0ps 0 100ps 0 150ps 1.0 1ns 1.0

❑ Pulsed Source

vck clk gnd PULSE 0 1.0 0ps 100ps 100ps 300ps 800ps

PULSE v1 v2 td tr tf pw per



Circuit Elements

Letter	Element
R	Resistor
C	Capacitor
L	Inductor
K	Mutual Inductor
V	Independent voltage source
I	Independent current source
M	MOSFET
D	Diode
Q	Bipolar transistor
W	Lossy transmission line
X	Subcircuit
E	Voltage-controlled voltage source
G	Voltage-controlled current source
H	Current-controlled voltage source
F	Current-controlled current source

Units

Letter	Unit	Magnitude
a	atto	10^{-18}
f	femto	10^{-15}
p	pico	10^{-12}
n	nano	10^{-9}
u	micro	10^{-6}
m	milli	10^{-3}
k	kilo	10^3
x	mega	10^6
g	giga	10^9

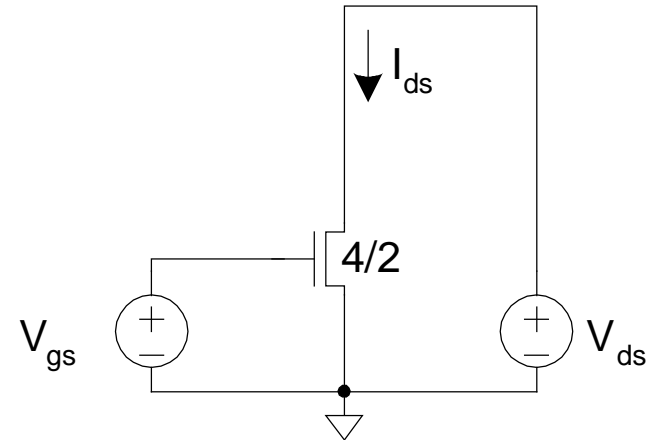
Ex: 100 femptofarad capacitor = 100fF, 100f, 100e-15

Analyses

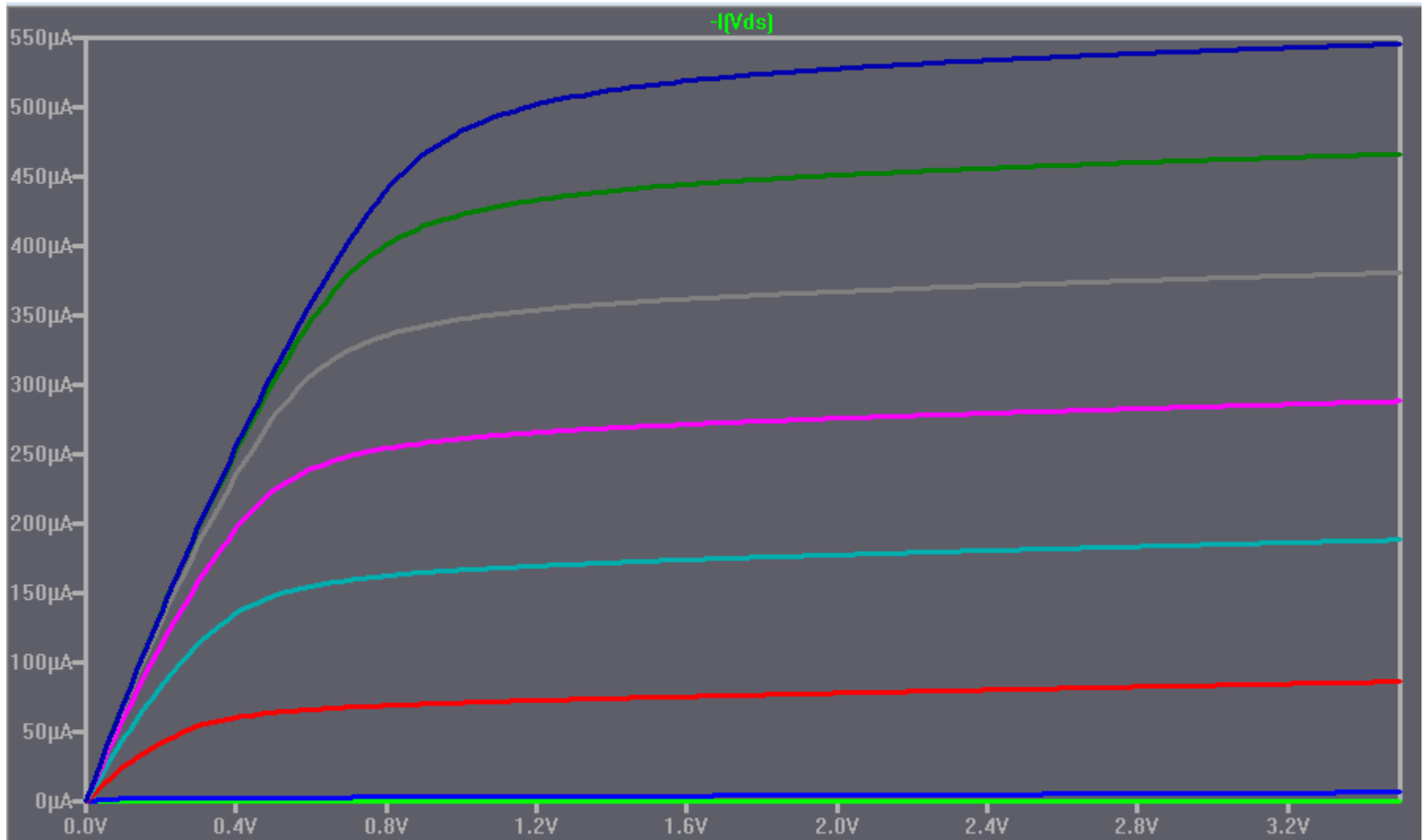
.ac	Small signal AC (linear)
.dc	DC transfer function
.disto	Distortion
.noise	Noise (linear)
.op	Operating Point
.pz	Pole-zero (linear)
.sens	DC or AC Sensitivity
.temp	Temperature sweep
.tf	Transfer function (linear)
.tran	Transient

DC Analysis

```
* MOS_IV.cir
* backland@stevens.edu 4/2/12
* Plot IV characteristics of NMOS device
*-----
* Parameters and options
*-----
.options reltol=.01 gshunt=1e-9
.include Models_mosis_tsmc_180nm_18.ftr
.temp 70
*-----
* Simulation netlist
*-----
M1    d g gnd gnd NMOS W=0.4u L=0.2u
*-----
* Sources
*-----
Vgs   g gnd 0
Vds   d gnd 0
*-----
* Commands
*-----
.dc Vds 0 3.5 0.1 Vgs 0 3.5 0.5
.plot -i(vds)    *ignored by LTspice
.end
```



LTspice: I-V Characteristics



MOSFET Elements

M element for MOSFET:

Mname drain gate source body type

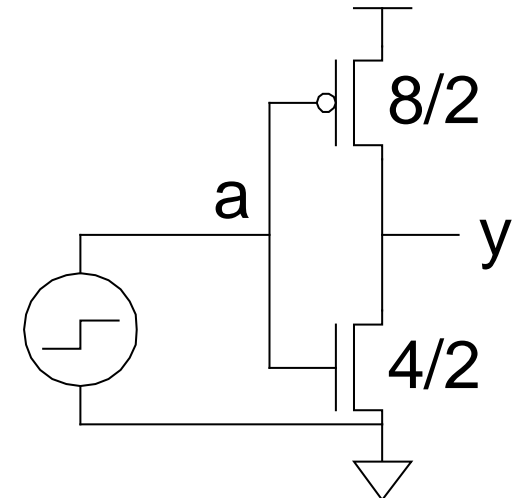
+ W=<width> L=<length>

+ AS=<area source> AD = <area drain>

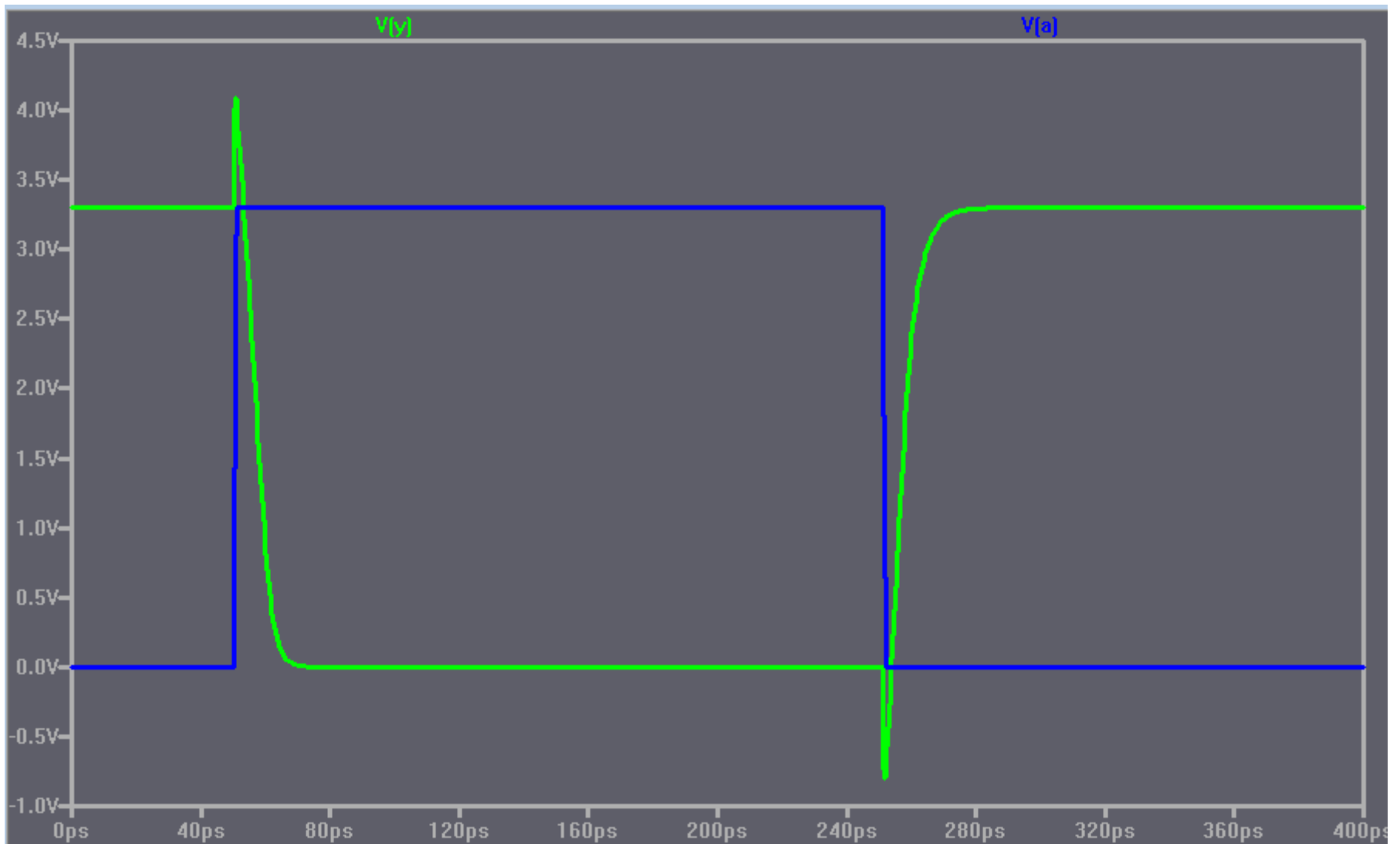
+ PS=<perimeter source> PD=<perimeter drain>

Transient Analysis (LTspice)

```
* Inverter.cir
*-----
* Parameters and options
*-----
.options reltol=.01 gshunt=1e-9
.include Models_mosis_tsmc_180nm_18.ftr
.param SUPPLY=3.3
*-----
* Simulation netlist
*-----
M1  y a gnd gnd NMOS W=0.4u L=0.2u AD=1.6E-13 PD=1.6u AS=1.6E-13 PS=1.6u
M2  y a vdd vdd PMOS W=0.8u L=0.2u AD=3.2E-13 PD=2.4u AS=3.2E-13 PS=2.4u
*-----
* Sources
*-----
Vvdd  vdd gnd {SUPPLY}
Vvin  a gnd PULSE 0 {SUPPLY} 50p 1p 1p 200p 400p
*-----
* Commands
*-----
.tran 5p 400p
.plot v(a) v(y)      *ignored by LTspice
.end
```



Inverter Transient Results

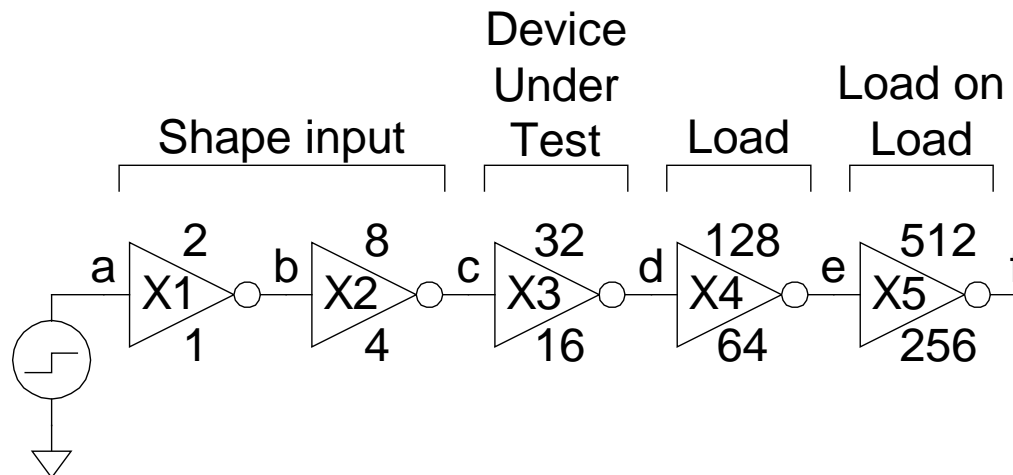


Subcircuits

- Build hierarchy by declaring common elements as subcircuits

```
.subckt inv a y N=1 P=2
M1 y a gnd gnd NMOS W='N*0.4u' L=0.2u AS='N*0.4u*0.2u'
+ PS='2*N*0.4u+0.4u' AD='N*0.4u*0.2u' PD='2*N*0.4u+0.4u'
M2 y a vdd vdd PMOS W='P*0.4u' L=0.2u AS='P*0.4u*0.2u'
+ PS='2*P*0.4u+0.4u' AD='P*0.4u*0.2u' PD='2*P*0.4u+0.4u'
.ends
```

- Ex: Fanout-of-4 Inverter Delay



FO4 Inverter Delay (LTspice)

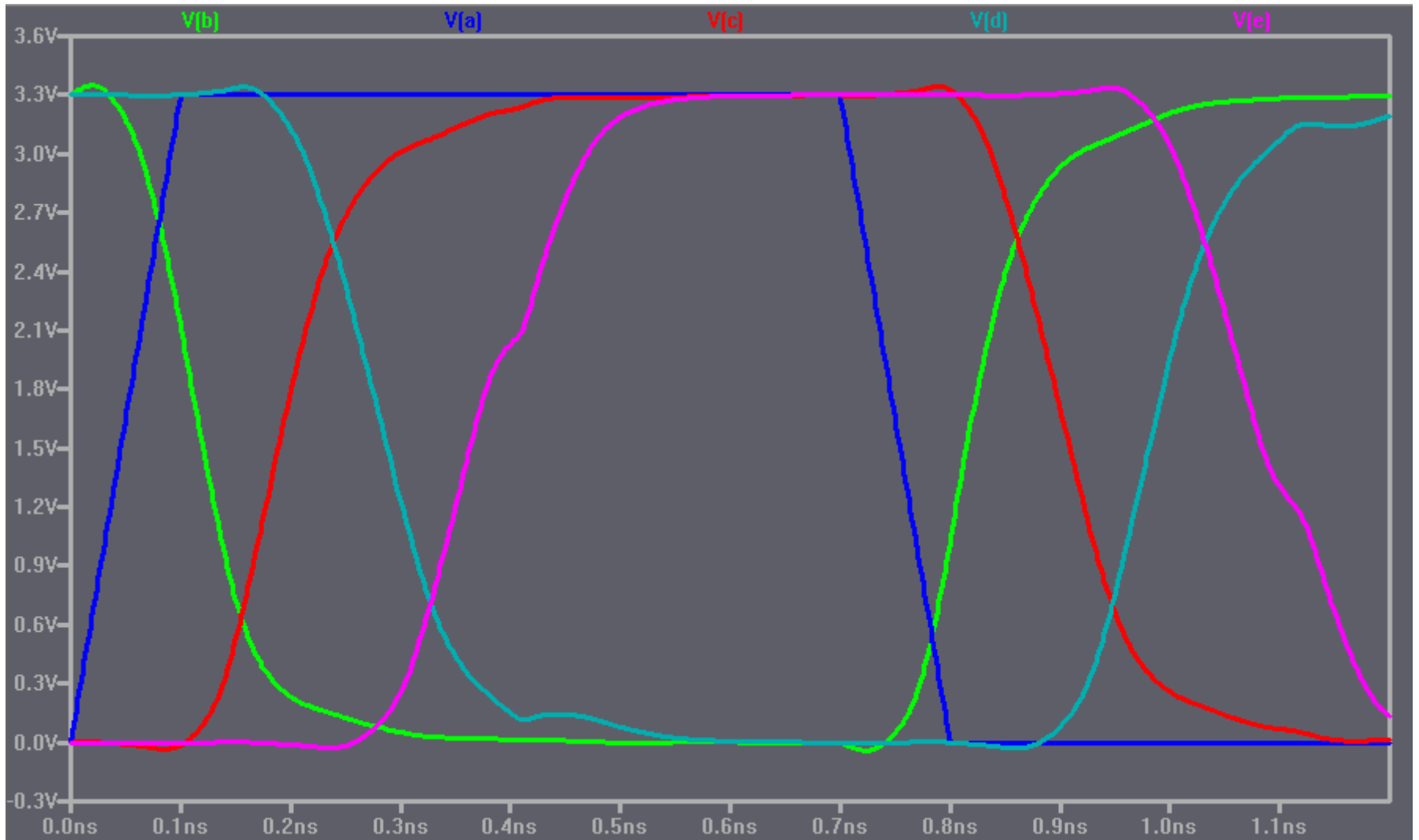
```
* Inv_FO4_delay.cir
*-----
* Parameters and options
*-----
.options reltol=.01 gshunt=1e-9
.include Models_mosis_tsmc_180nm_18.ftr
.param SUPPLY=3.3
.global vdd gnd
*-----
* Simulation netlist
*-----
.subckt inv a y N=1 P=2
M1 y a gnd gnd NMOS W='N*0.4u' L=0.2u AS='N*0.4u*0.2u'
+ PS='2*N*0.4u+0.4u' AD='N*0.4u*0.2u' PD='2*N*0.4u+0.4u'
M2 y a vdd vdd PMOS W='P*0.4u' L=0.2u AS='P*0.4u*0.2u'
+ PS='2*P*0.4u+0.4u' AD='P*0.4u*0.2u' PD='2*P*0.4u+0.4u'
.ends

X1 a b inv N=1 P=2           *shape input
X2 b c inv N=4 P=8          *shape input
X3 c d inv N=16 P=32        *device under test
X4 d e inv N=64 P=128       *load
X5 e f inv N=256 P=512      *load on load
```

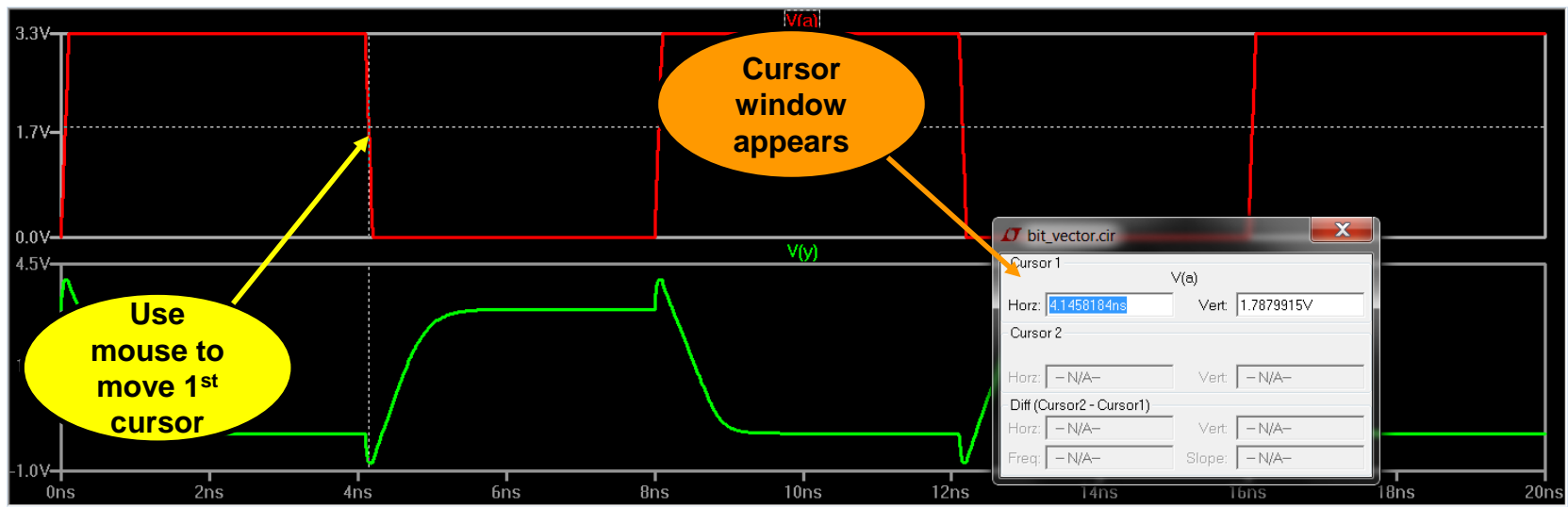
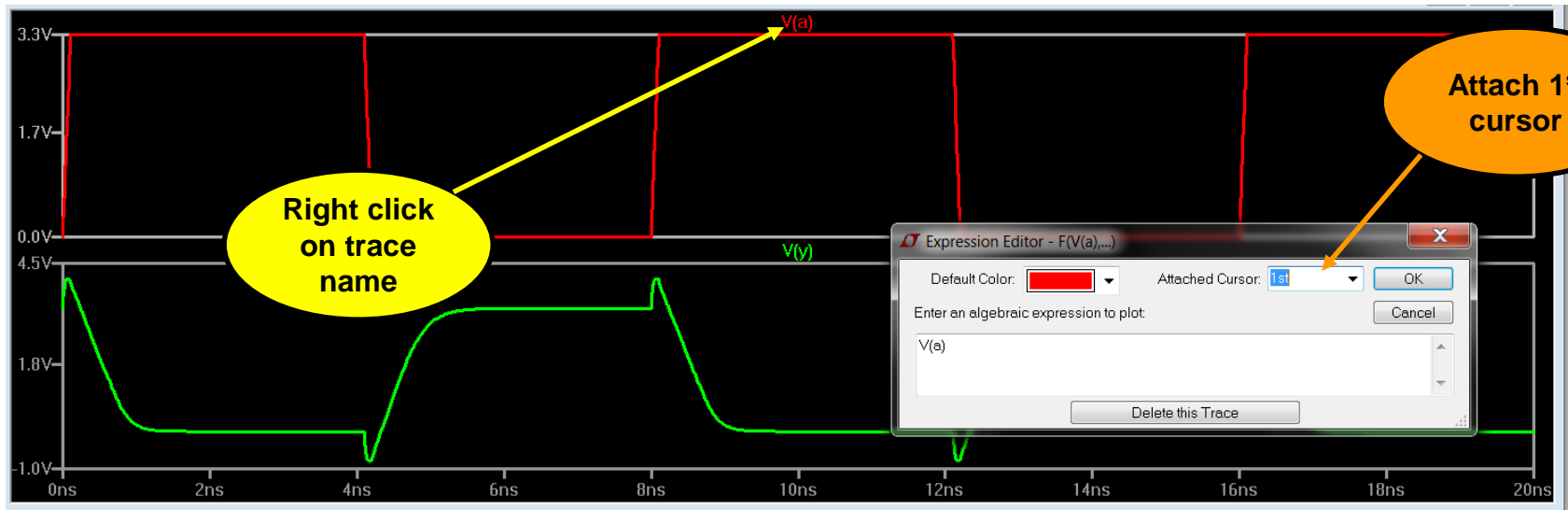
FO4 Inverter Delay (cont.)

```
*-----  
* Sources  
*-----  
Vvdd  vdd gnd {SUPPLY}  
Vvin  a gnd PULSE 0 {SUPPLY} 0p 100p 100p 600p 1200p  
*-----  
* Commands  
*-----  
.tran 1p 1200p  
.plot v(a) v(b) v(c) v(d) v(e)      *ignored by LTspice  
.end
```

FO4 Results



Measuring Delays in LTSpice



Measuring Delays in LTSpice (2)

