Mixed-Signal Multi-Level Circuit Simulation with a Dynamic Delay Model

Dr Zhou Xing

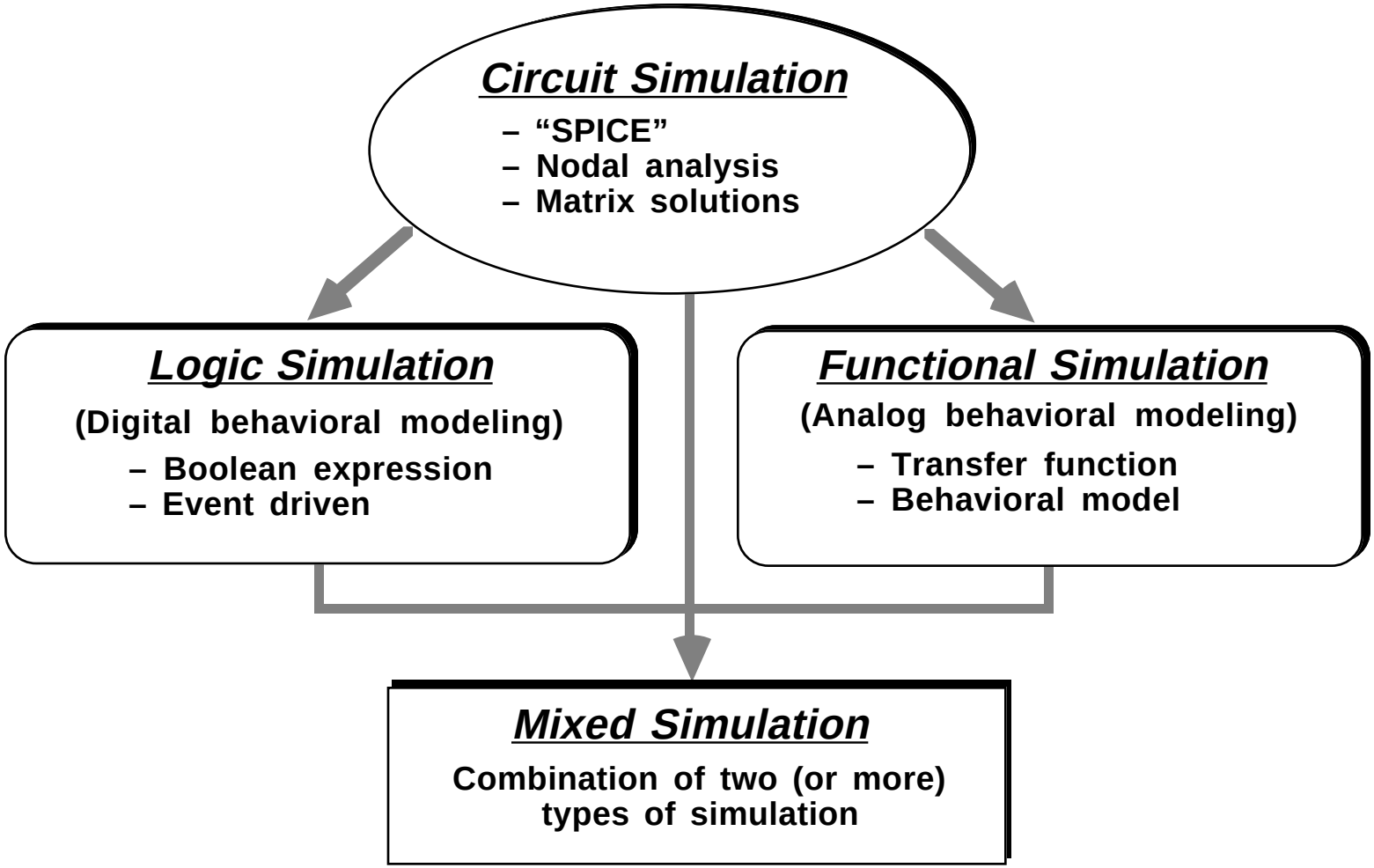
School of Electrical and Electronic Engineering
Nanyang Technological University

October 17, 1996

Outline

- **Motivation and Objective**
  - Develop a truly implicit mixed-signal multi-level simulator with automatic circuit partitioning and dynamic mode switching
  - Develop a dynamic delay model for accurate timing simulation of large digital circuits with gate-level speed and circuit-level accuracy
- **Approach**
  - Two representations of a logic gate based on subcircuit expansion approach
  - Gate-level logic model parameters are directly extracted from the circuit-level electrical simulation
- **Dynamic delay model**
  - Nonlinear loading effect and the effects of input transition time and multiple-input triggering
  - Automatic fanout extraction based on the user netlist
- **Demonstration**

Types of Circuit Simulation



Xsim at a Glance

❑ Types of simulation

- Electrical (circuit-level)
- Digital (gate-level)
- Analog (behavioral-level)
- Mixed signal (analog & digital)
- Mixed level (gate←circuit→behavioral)
- Mixed technology

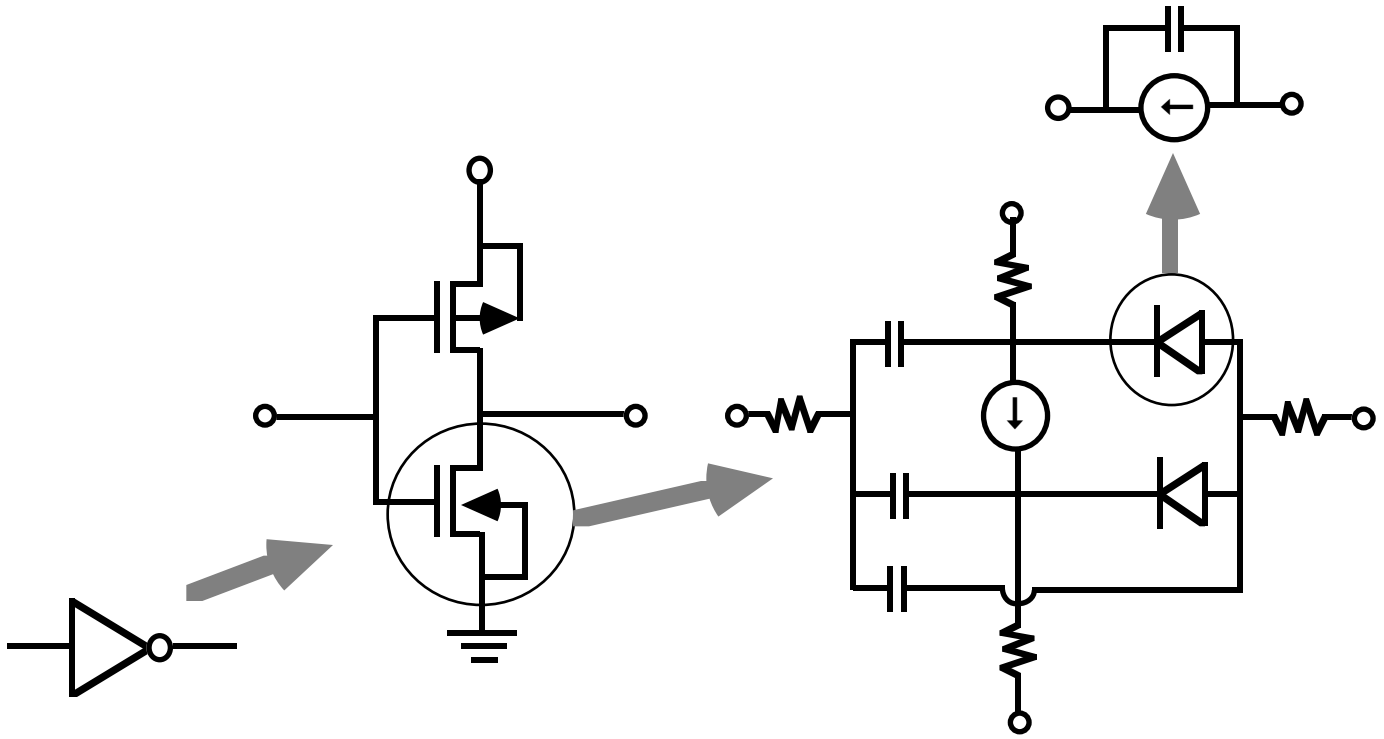
❑ Implemented models

- Device models: diode; MOS level 1, 2, 3; BJT Gummel–Poon; AIM HFET, MESFET, HBT, BiCMOS
- Dynamic delay model
- Built-in behavioral functionals (SPICE netlist format)

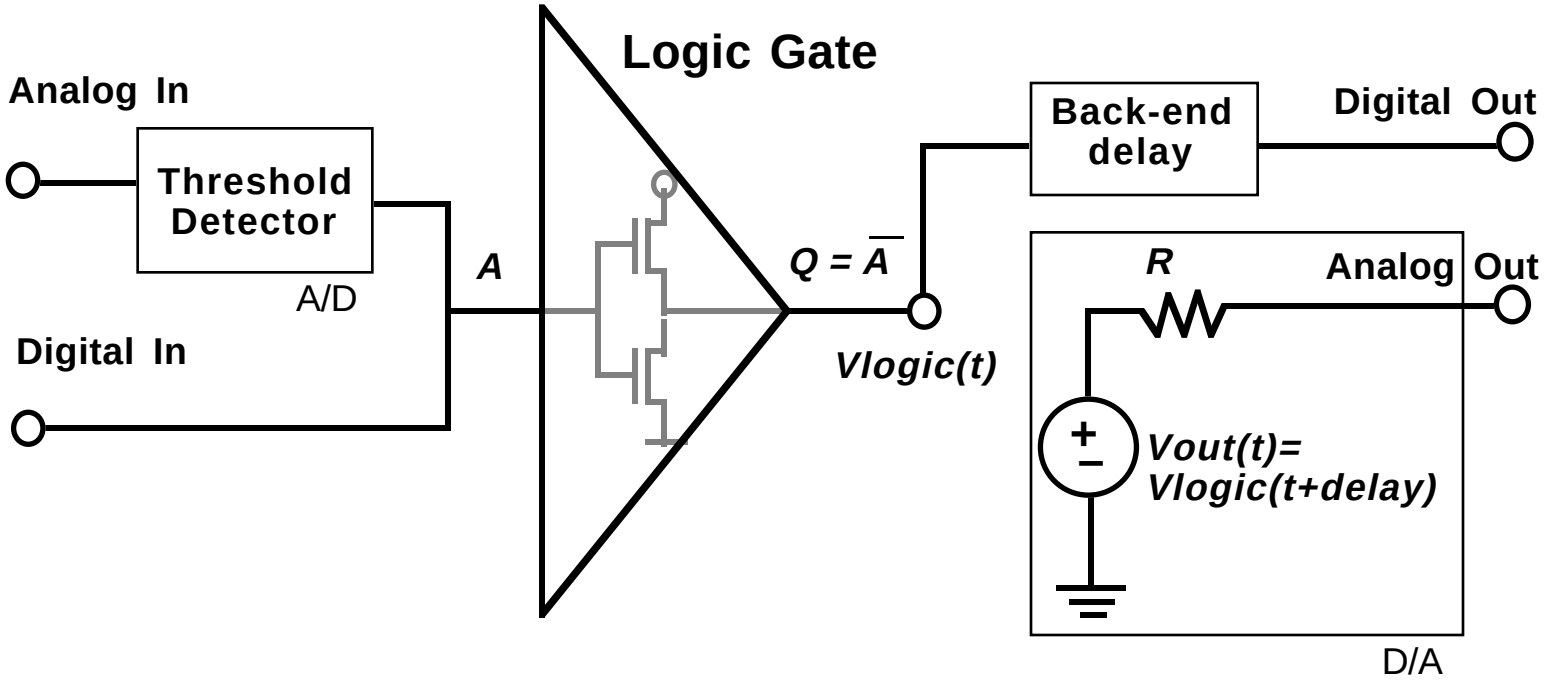
❑ Key features

- Automatic circuit partitioning
- Dynamic mode switching
- Hierarchical signal probing

Subcircuit Expansion Approach



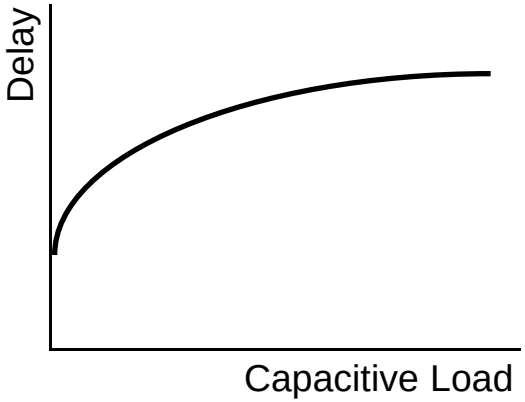
Logic Block Equivalent Circuit



The Need for a Dynamic Delay Model

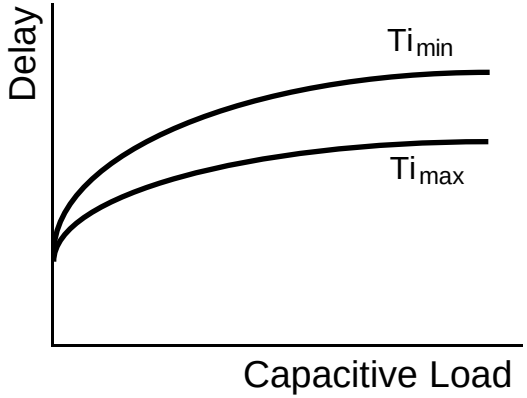
❑ Fanout

- Delay depends on the load, which is non–linear for submicron circuits
- User-netlist dependent



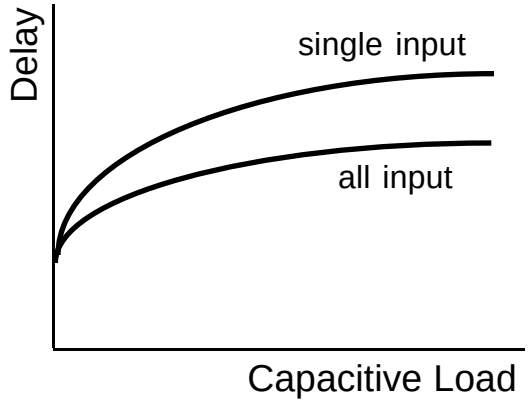
❑ Input slope

- Delay depends on the input transition time (slope)
- Run-time dependent



❑ Fanin

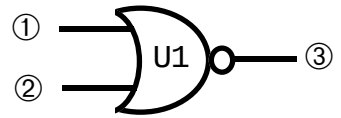
- For multiple input gate, delay depends on which input is triggered
- Run-time dependent



Logic Gate Model

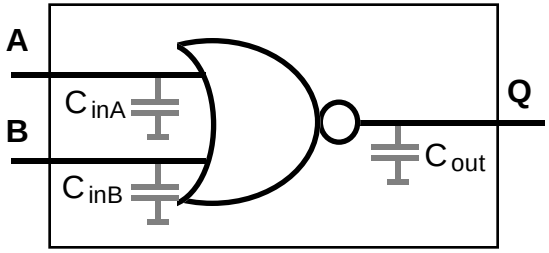
- **Logic element**

```
U1 0 3 2 1 hfet nor2 hfet-nor2
```



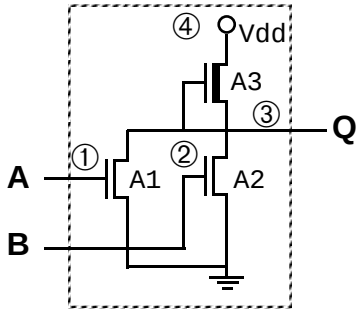
- **Logic (digital, behavioral) representation**

```
.model hfetnor2 logic vmax=1.0
+ vmin=0.1 thh=0.8 thl=0.2 mr=5
+ mf=5 over=0.1 cap1=… cap2=…
```



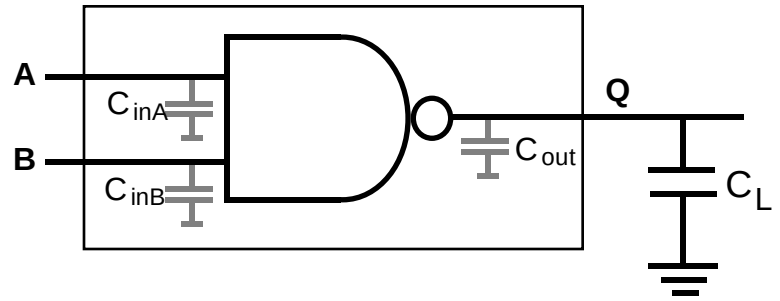
- **Circuit (analog, structural) representation**

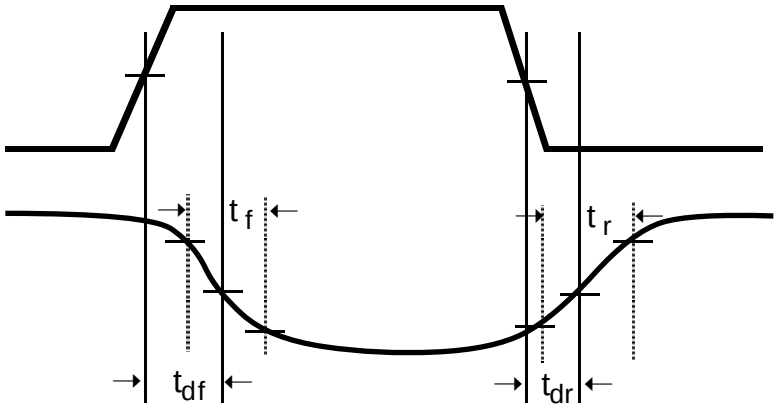
```
.subckt hfet-nor2 0 3 2 1
A1 3 1 0 nhfet l=0.5u w=5u
A2 3 2 0 nhfet l=0.5u w=5u
A3 4 3 3 nhfet l=0.5u w=4u
Vdd 4 0 1.0
.ends
```



Basic Logic Gate Parameters

- **Input capacitance**
  - $C_{in} = \left\langle \frac{I(V)}{dV/dt} \right\rangle_V$
- **Intrinsic delay**
  - $t_{di} = t_d \big|_{C_L = 0}$
- **Effective resistance**
  - $R_{eff} = \frac{t_d - t_{di}}{C_L}$
- **Output capacitance**
  - $C_{out} = \left\langle \frac{t_{di}}{R_{eff}} \right\rangle_{\text{rise,fall}}$
- **Rise and fall times**
  - $t_r = t_{ri} + R_{tr}C_L$
  - $t_f = t_{fi} + R_{tf}C_L$

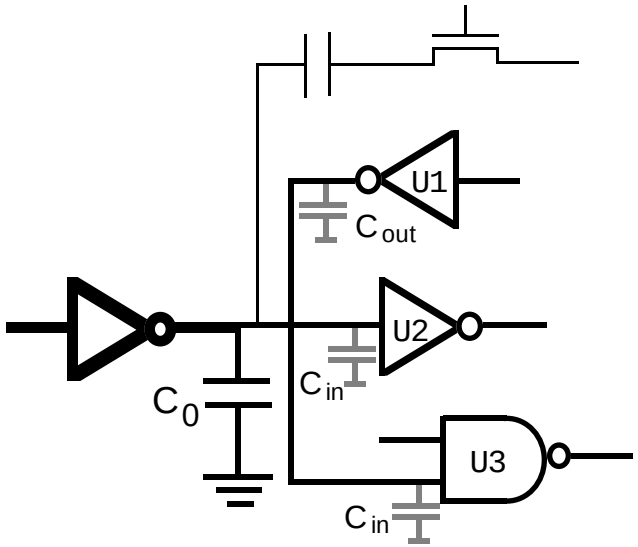




$$t_d = t_{di} + R_{eff}C_L$$

Fanout (Loading Capacitance) Extraction

Trace all the logic gates and grounding capacitances

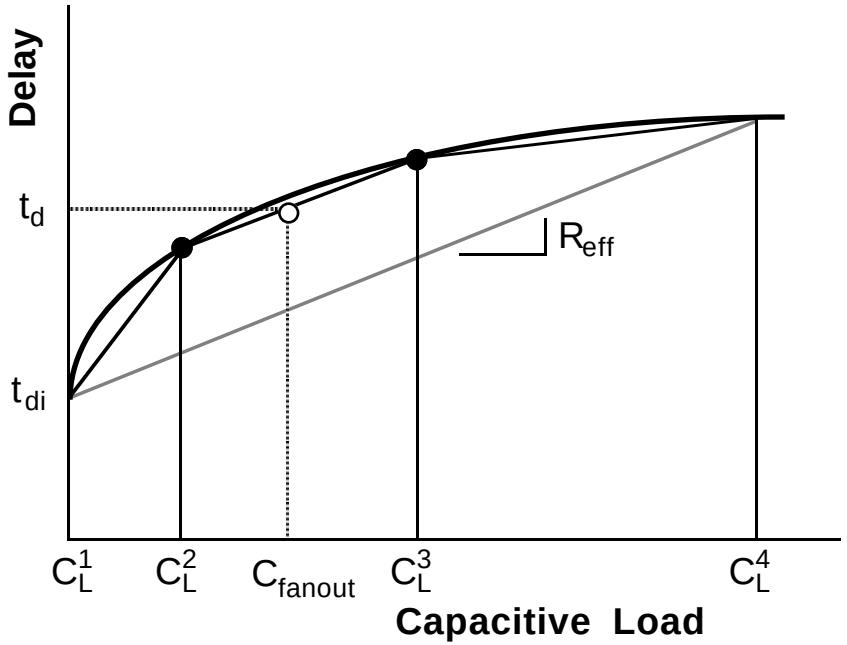


$$C_{fanout} = C_0 + \sum_U C_{in} + \sum_U C_{out}$$

How to Handle Nonlinear Load Dependence

Piece-wise linear approximation

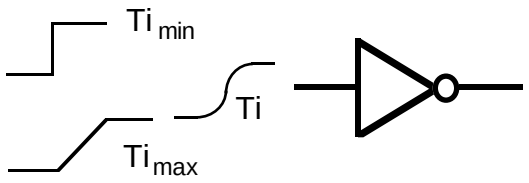
- **_N_-breakpoint algorithm**
  - Run full-subcircuit for *N* loading capacitances to extract the delays
  - Store the delay parameters in the logic model
- **Linear interpolation (or extrapolation) between neighboring breakpoints**

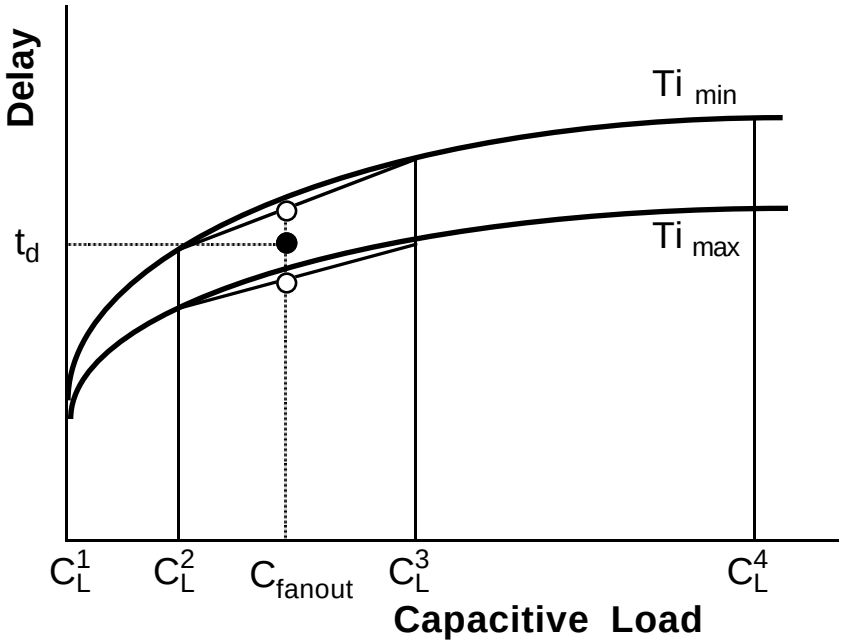


How to Handle Input Slope Dependence

Linear interpolation/extrapolation

- **Max/min slope pre-calculation**
  - Run full-subcircuit for two "extreme" input transition times to extract the delays
  - Store the delay parameters in the logic model
- **Linear interpolation/ extrapolation**
  - Trace the input transition time at each time step for each gate
  - Use linear interpolation based on the actual C_{fanout} and its interpolated values between neighboring breakpoints

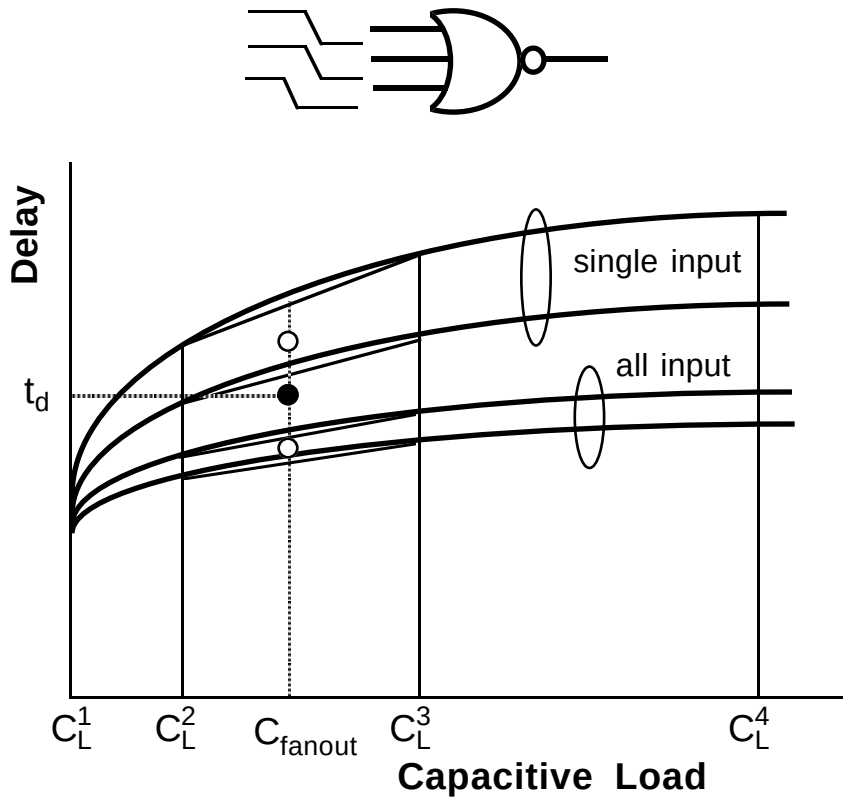




How to Handle Multiple Input Triggering

Best/worst-case scenario

- **Single/all-input triggering pre-calculation**
  - Run full-subcircuit for single- and all-input triggering to extract the delays
  - Store the delay parameters in the logic model
- **Linear interpolation**
  - Trace the number of effective triggering inputs at each time step for each gate
  - Use linear interpolation based on the actual C_{fanout} and its interpolated values between neighboring breakpoints



Logic Library Configuration (1)

❑ Logic configuration ***file*.cfg**

```
U1 0 1 2 3 cmos nor2 cmos-nor2
.subckt cmos-nor2 1 2 3
M1 3 2 1 0 mname l=1u w=10u [subcircuit]
.ends
.model mname nmos [parameters]
.model cmos-nor2 logic (tunit= n=4 cmin= cmax= timin= timax= )
.probe tran
```

❑ Library generation ***file*.lib**

```
Xsim=> extract file                    (use file.cfg)
```

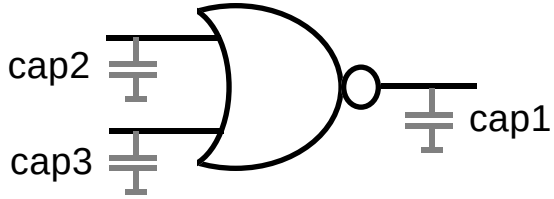
```
.model cmos-nor2 logic ( [logic parameters] cap1= cap2= cap3=
+ drms1= drxs1= drma1= drxa1= […] drms4= drxs4= drma4= drxa4=
+ dfms1= dfxs1= dfma1= dfxa1= […] dfms4= dfxs4= dfma4= dfxa4=
+ trms1= trxs1= trma1= trxa1= […] tfms4= tfxs4= tfma4= tfxa4=
```

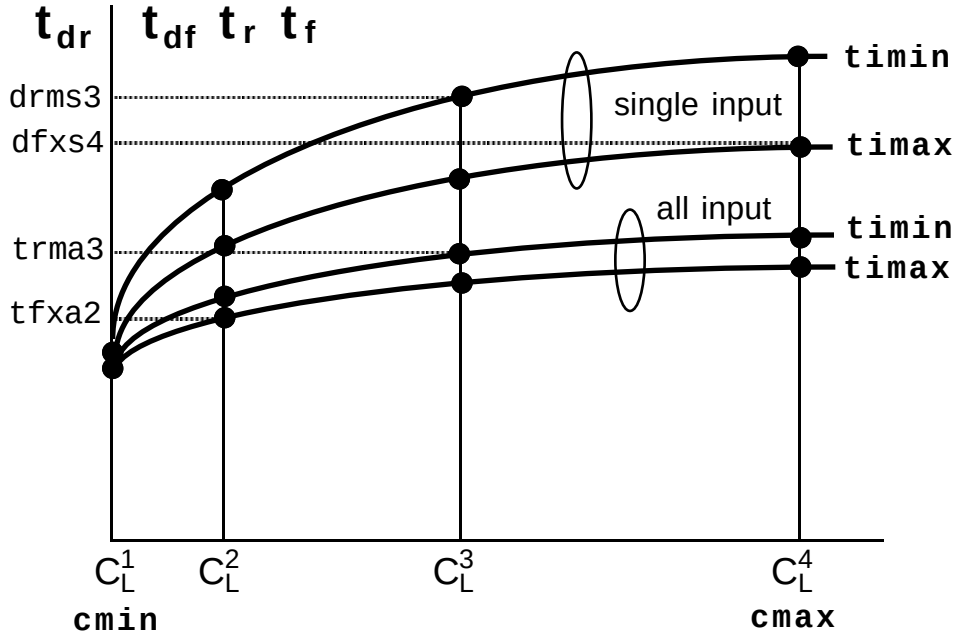
❑ Library inclusion ***file*.ckt**

```
.library file                          (include file.lib)
[user netlist]
```

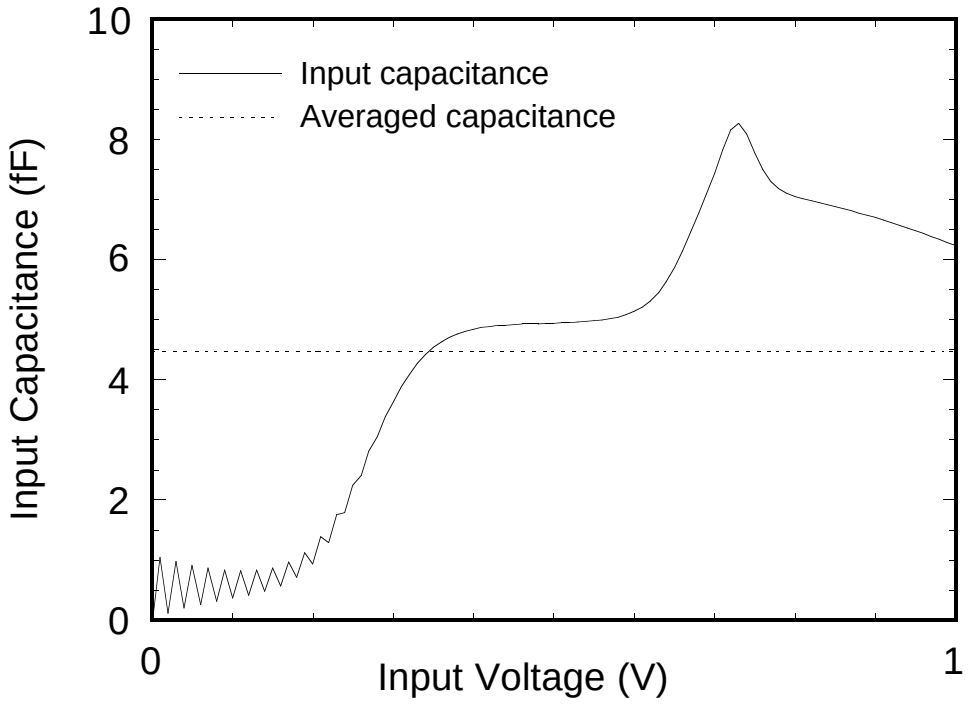
```
xsim file                              (run file.ckt)
```

Logic Library Configuration (2)

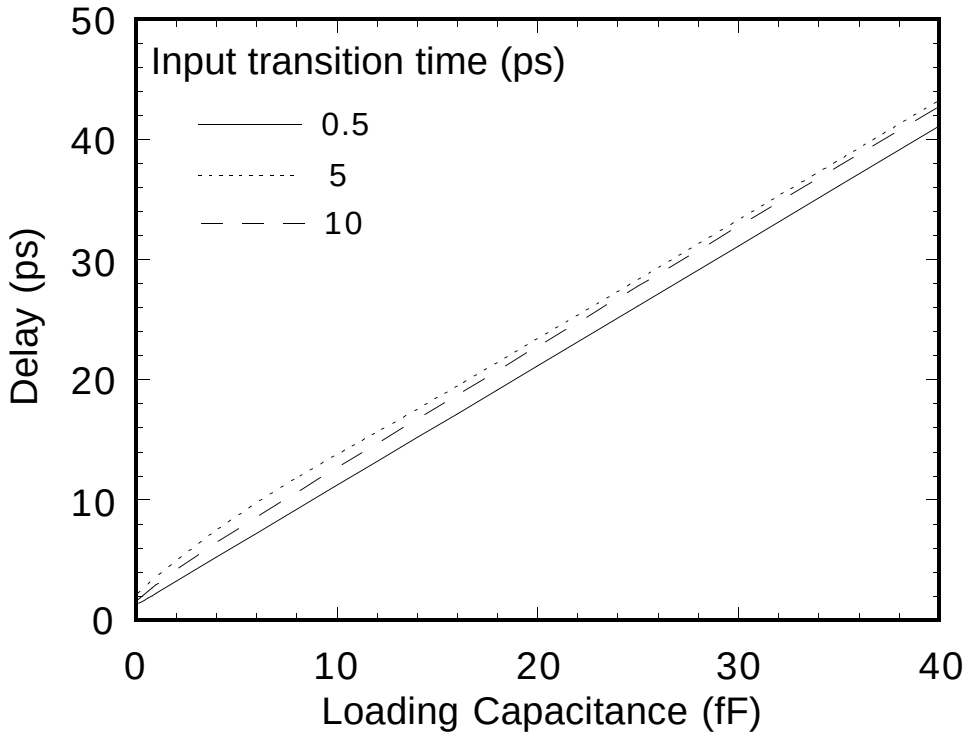




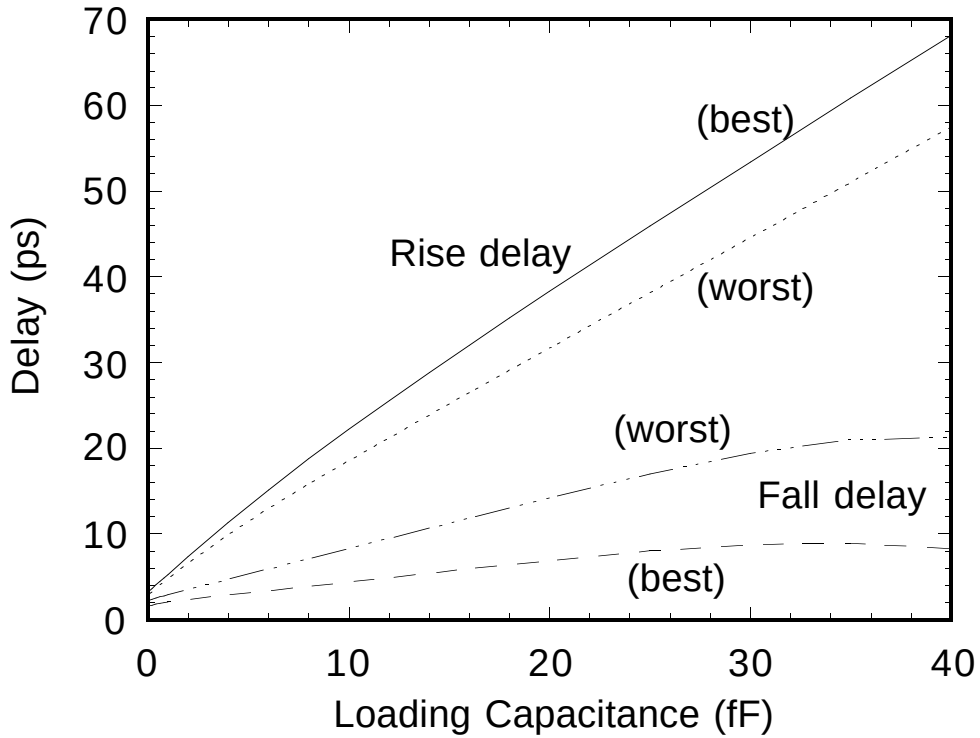
Input Capacitance Determination



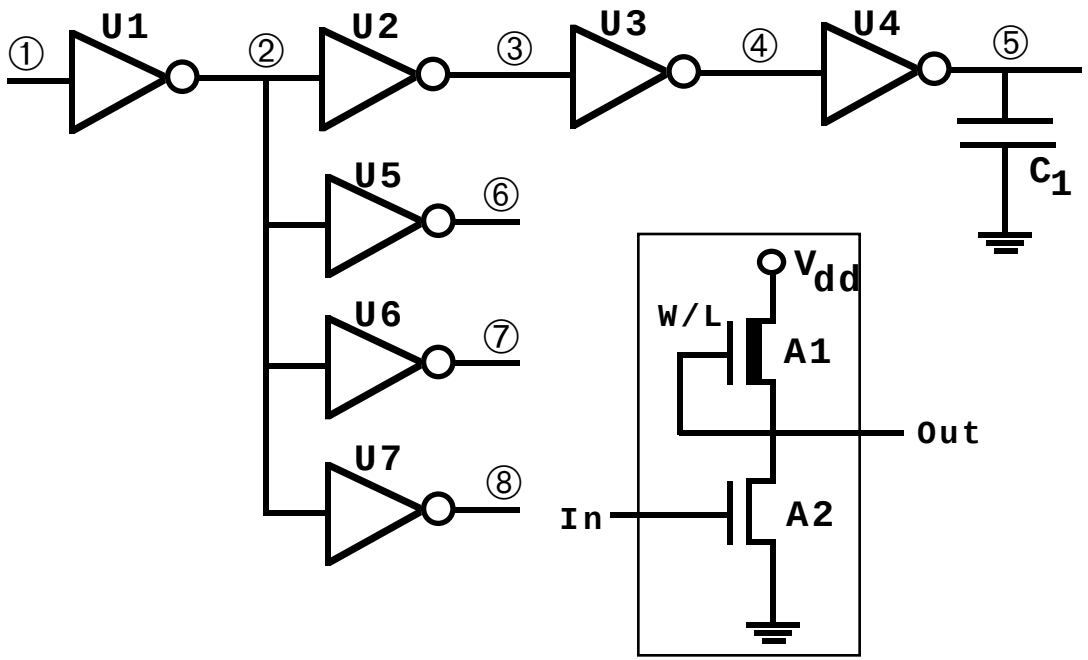
Delay with Different Input Transition Times



Delay with Multiple Input Triggering



Fanout Extraction and Subcircuit Tracing (1)



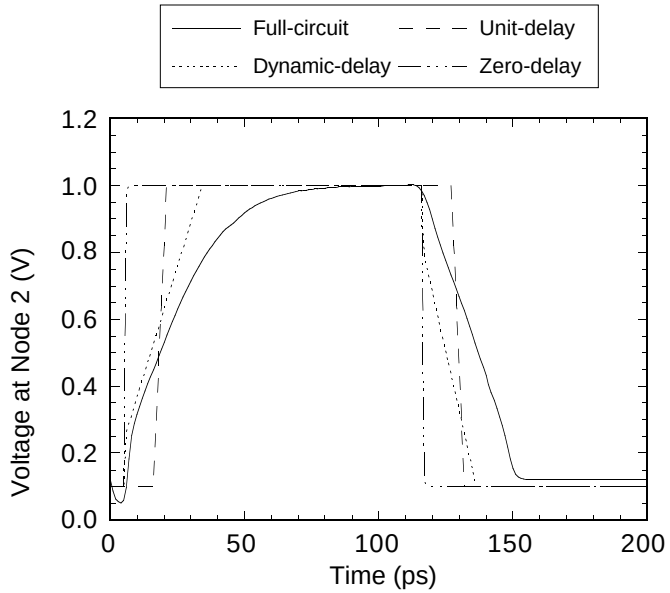

```
Xsim netlist
.lib hfet
U1 0 2 1 hfet inv1 hfet-inv1
U2 0 3 2 hfet inv1 hfet-inv1
U3 0 4 3 hfet inv1 hfet-inv1
U4 0 5 4 hfet inv1 hfet-inv1
U5 0 6 2 hfet inv2 hfet-inv2
U6 0 7 2 hfet inv3 hfet-inv3
U7 0 8 2 hfet inv4 hfet-inv4
C1 0 5 5e-15
V1 1 0 pulse(1. 0. 0. 10p 10p
+ 100p 200p)
.ic v(5)=0
.set mode=analog
*.set mode=digital
.probe tran v(nodes)
.plot tran v 2 5
.tran 0 400p 1p
```

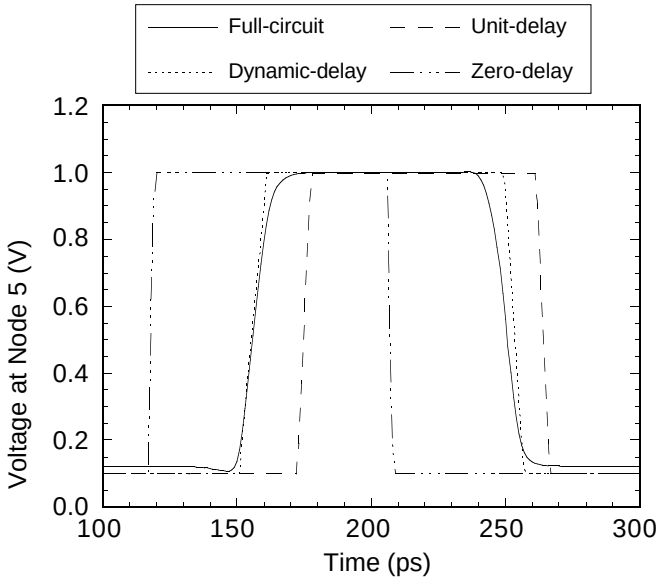
Fanout Extraction and Subcircuit Tracing (2)


```
Xsim=> fanout
 U1: c_fanout = 1.93372e-14
 U2: c_fanout = 6.1576e-15
 U3: c_fanout = 6.1576e-15
 U4: c_fanout = 6.67485e-15
 U5: c_fanout = 2.01527e-15
 U6: c_fanout = 2.11913e-15
 U7: c_fanout = 1.63416e-15
Xsim=> network u
Node: Branches:
   0: U1 U2 U3 U4 U5 U6 U7
   1: U1
   2: U1 U2 U5 U6 U7
   3: U2 U3
   4: U3 U4
   5: U4
   6: U5
   7: U6
   8: U7
Xsim=> network a
Node: Branches:
   0: A1.U1 A1.U2 A1.U3 A1.U4 A1.U5 A1.U6 A1.U7
   1: A1.U1
   2: A2.U1 A2.U1 A1.U1 A1.U2 A1.U5 A1.U6 A1.U7
   3: A2.U2 A2.U2 A1.U2 A1.U3
   4: A2.U3 A2.U3 A1.U3 A1.U4
   5: A2.U4 A2.U4 A1.U4
   6: A2.U5 A2.U5 A1.U5
   7: A2.U6 A2.U6 A1.U6
   8: A2.U7 A2.U7 A1.U7
```

```
Xsim=> network c
Node: Branches:
   0: C1
   1: Cgd.A1.U1 Cgs.A1.U1
   2: Cgd.A2.U1 Cgs.A2.U1 Cgd.A1.U2
+ Cgs.A1.U2 Cgd.A1.U5 Cgs.A1.U5
+ Cgd.A1.U6 Cgs.A1.U6 Cgd.A1.U7
+ Cgs.A1.U7
   3: Cgd.A2.U2 Cgs.A2.U2 Cgd.A1.U3
+ Cgs.A1.U3
   4: Cgd.A2.U3 Cgs.A2.U3 Cgd.A1.U4
+ Cgs.A1.U4
   5: Cgd.A2.U4 Cgs.A2.U4 C1
   6: Cgd.A2.U5 Cgs.A2.U5
   7: Cgd.A2.U6 Cgs.A2.U6
   8: Cgd.A2.U7 Cgs.A2.U7
Xsim=> network r
Node: Branches:
   0: Rs.A1.U1 Rs.A1.U2 Rs.A1.U3
+ Rs.A1.U4 Rs.A1.U5 Rs.A1.U6 Rs.A1.U7
   1:
   2: Rs.A2.U1 Rd.A1.U1
   3: Rs.A2.U2 Rd.A1.U2
   4: Rs.A2.U3 Rd.A1.U3
   5: Rs.A2.U4 Rd.A1.U4
   6: Rs.A2.U5 Rd.A1.U5
   7: Rs.A2.U6 Rd.A1.U6
   8: Rs.A2.U7 Rd.A1.U7
```

Comparison of Different Delay Models

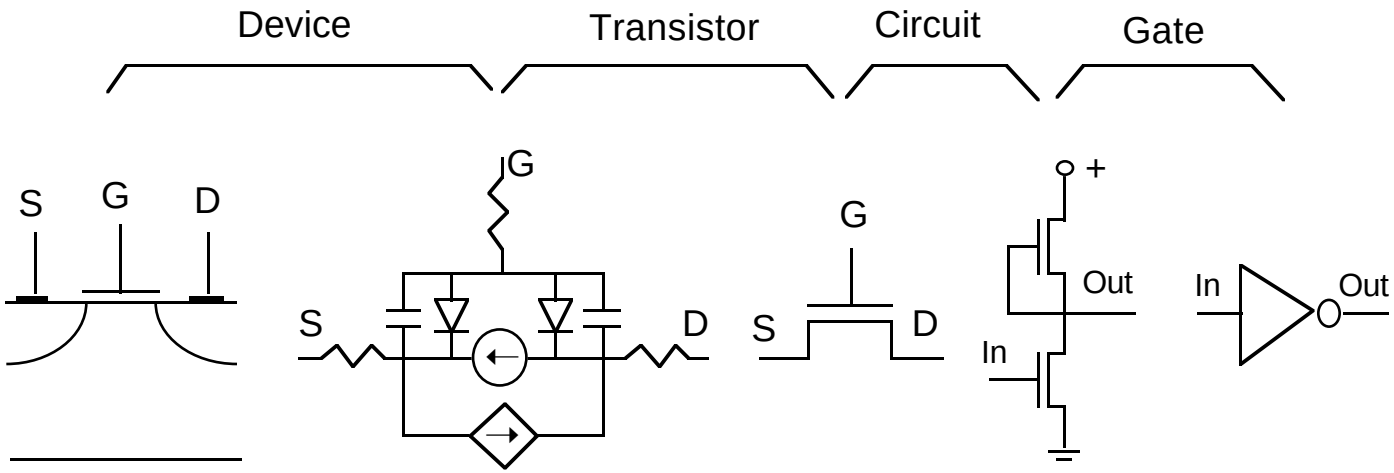




- **Circuit parameters:** W_{1-4} = 5 μm, W_5 = 4 μm, W_6 = 6 μm, W_7 = 7 μm
- **Input capacitances:** C_{in1-4} = 4.48 fF, C_{in5} = 4.64 fF, C_{in6} = 4.33 fF, C_{in7} = 4.20 fF
- **CPU time (sec):** Full-circuit = 21.86, dynamic = 3.89, unit = 3.61, zero = 3.54

Multi-Level Modeling of Logic Gates

Multi-level representation of an HEMT



Device Level

```
.model <Mname> nhfet
+ [param-list]
```

Transistor Level

```
Axxx <Nd> <Ng> <Ns>
+ <Mname> <L=val> <W=val>
```

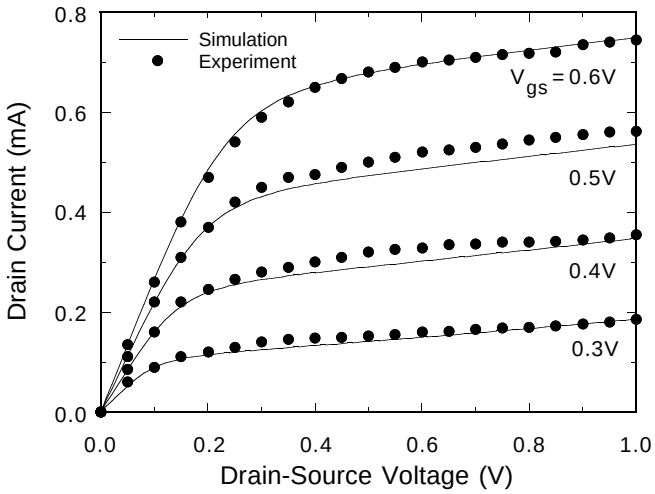
Gate Level

```
Uxxx <Nref> <Nout> <Nin1>
+ [<Nin2> …] <Family>
+ <Type><n> <Cktname>
```

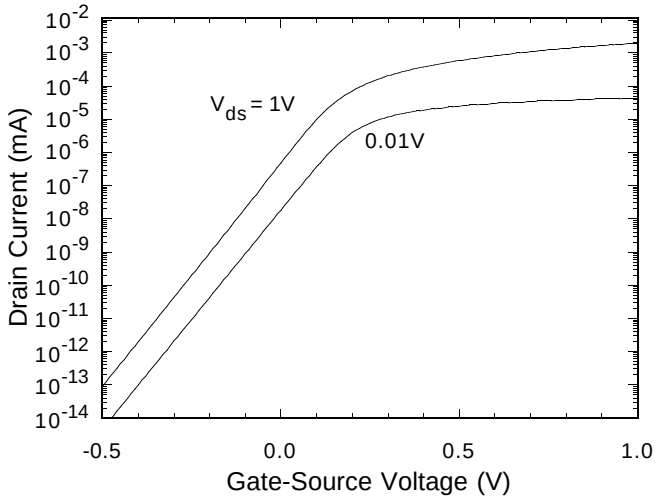
Transistor-Level DC Characteristics

HEMT device with L = 1 μm, W = 10 μm

Output Characteristic



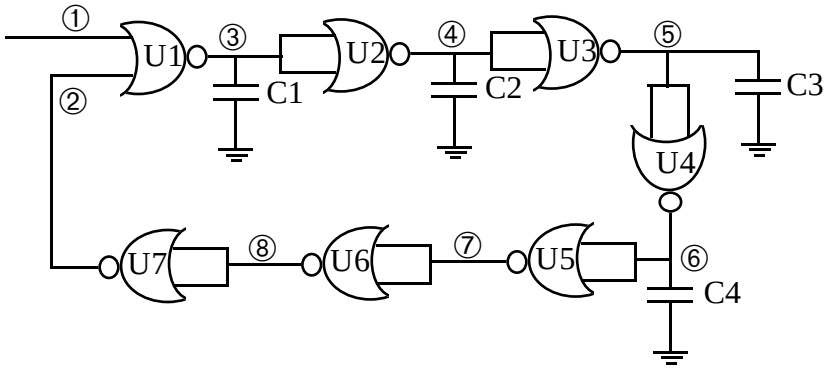
Input Characteristic



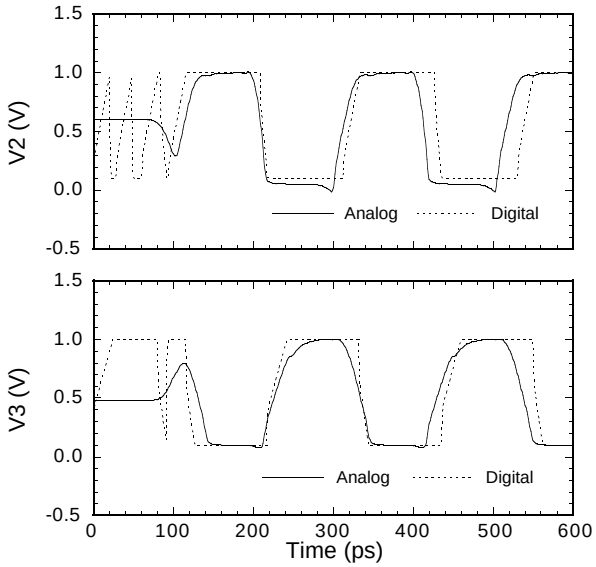
Gate-Level Timing Simulation

Seven-stage NOR-gate ring oscillator using E/D HEMT device

Circuit



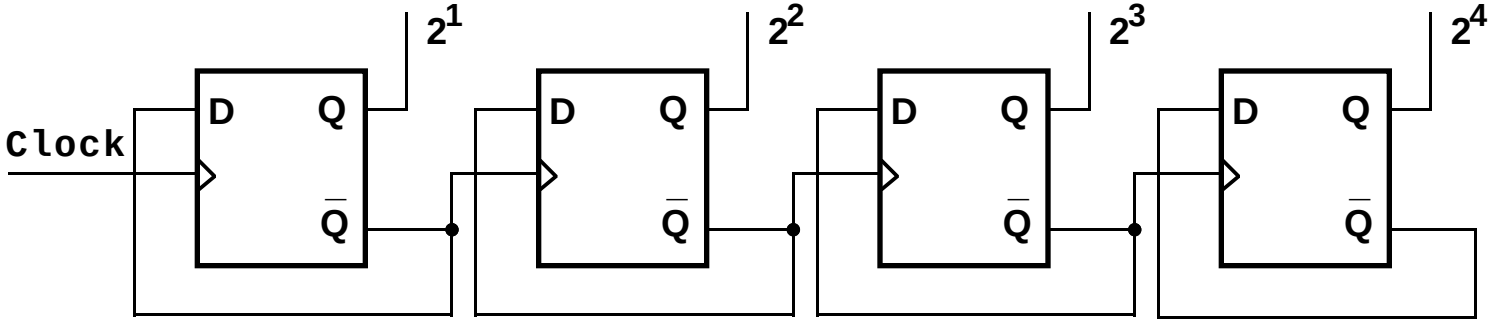
Waveform

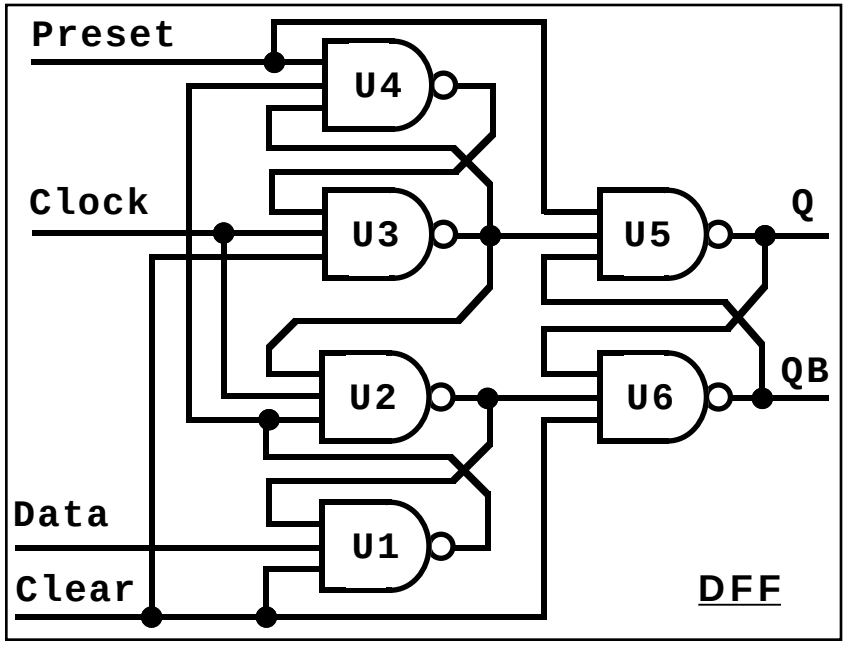


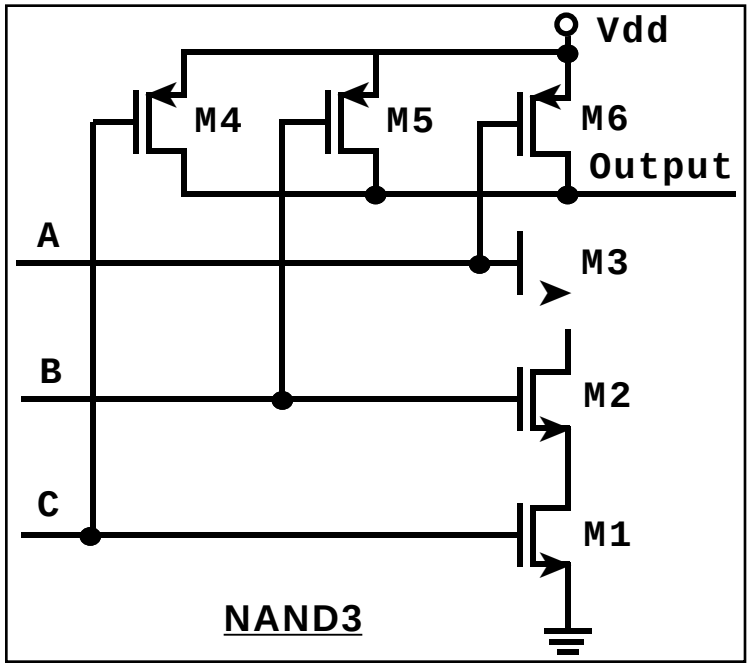
CPU Time

Digital mode:	12.85 sec
Analog mode:	57.43 sec

Multi-Level Digital Simulation (1)







Multi-Level Digital Simulation (2)

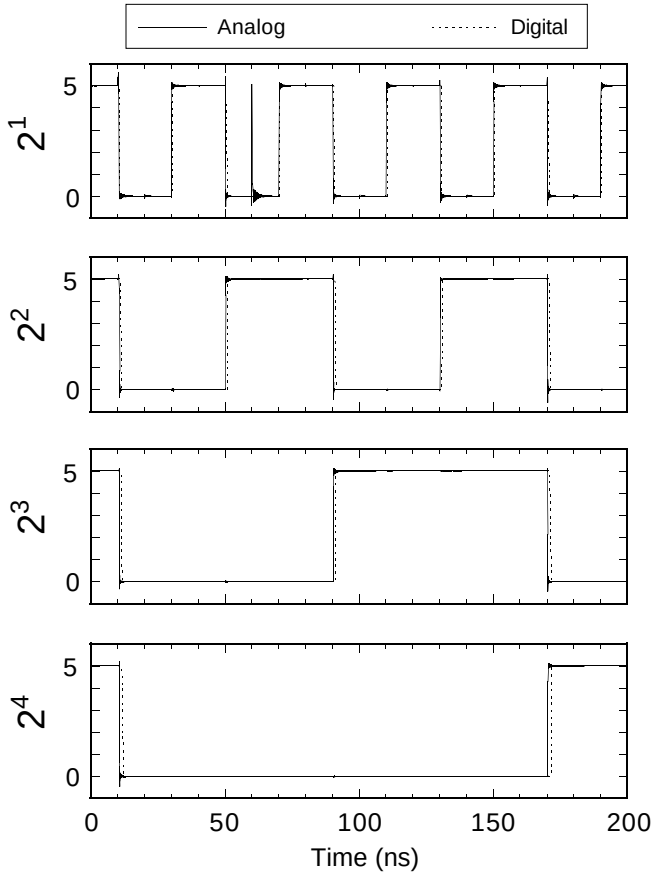
Netlist

```
Xsim netlist for the shift register
.lib mos12u
X1 1 2 2 3 dff
X2 2 4 4 5 dff
X3 4 6 6 7 dff
X4 6 8 8 9 dff
.subckt dff 1 3 9 10
U1 0 5 6 3 4 cmos nand3 cmos-nand3
U2 0 6 7 1 5 cmos nand3 cmos-nand3
U3 0 7 8 1 4 cmos nand3 cmos-nand3
U4 0 8 2 5 7 cmos nand3 cmos-nand3
U5 0 10 2 7 9 cmos nand3 cmos-nand3
U6 0 9 10 6 4 cmos nand3 cmos-nand3
V_clear 4 0 5
V_preset 2 0 5
.ends dff
V_clock 1 0 pulse(5 0 0 0 0 10n 20n)
.options mode=digital
*.options mode=analog
.tran 0 200n 0.1n
```

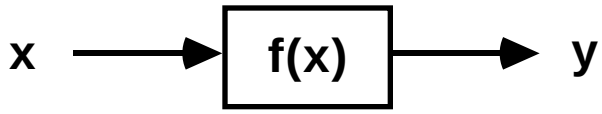
CPU Time

Digital mode:	344 sec
Analog mode:	1622 sec

Output Waveforms



Circuit-Level Behavioral Modeling



- **Idea** — model circuit elements using functions

- **Netlist syntax**
  - ***SPICE format***: *Label connection value/function*

    e.g., **`R1 1 2 10K`** **`Vin 3 0 sin(2.5 0.1 10meg)`**

  - ***Xsim extension***: *Label connection expression*

    where *expressions*: value; function + conditionals; sum of functions

Built-in Behavioral Modeling Functions and Conditionals

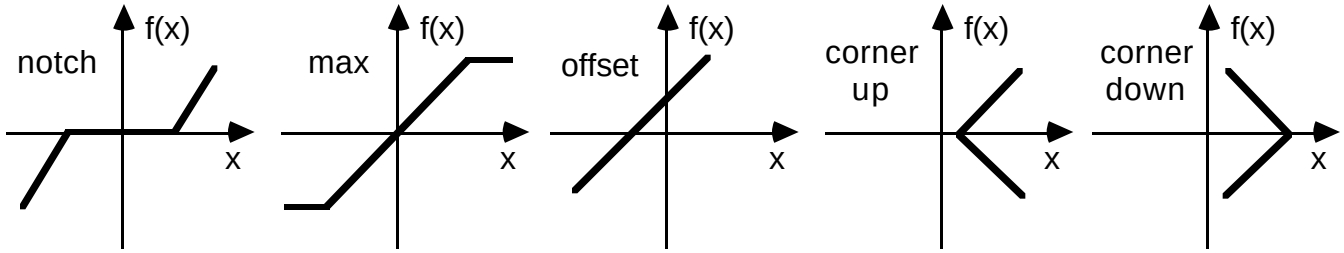
❑ Built-in functions

exp	cornerdown	max
pulse	cornerup	netfunction
pwl	complex	notch
sffm	delay	offset
sin	expterm	polar
tanh	generator	polyterm

❑ Conditionals

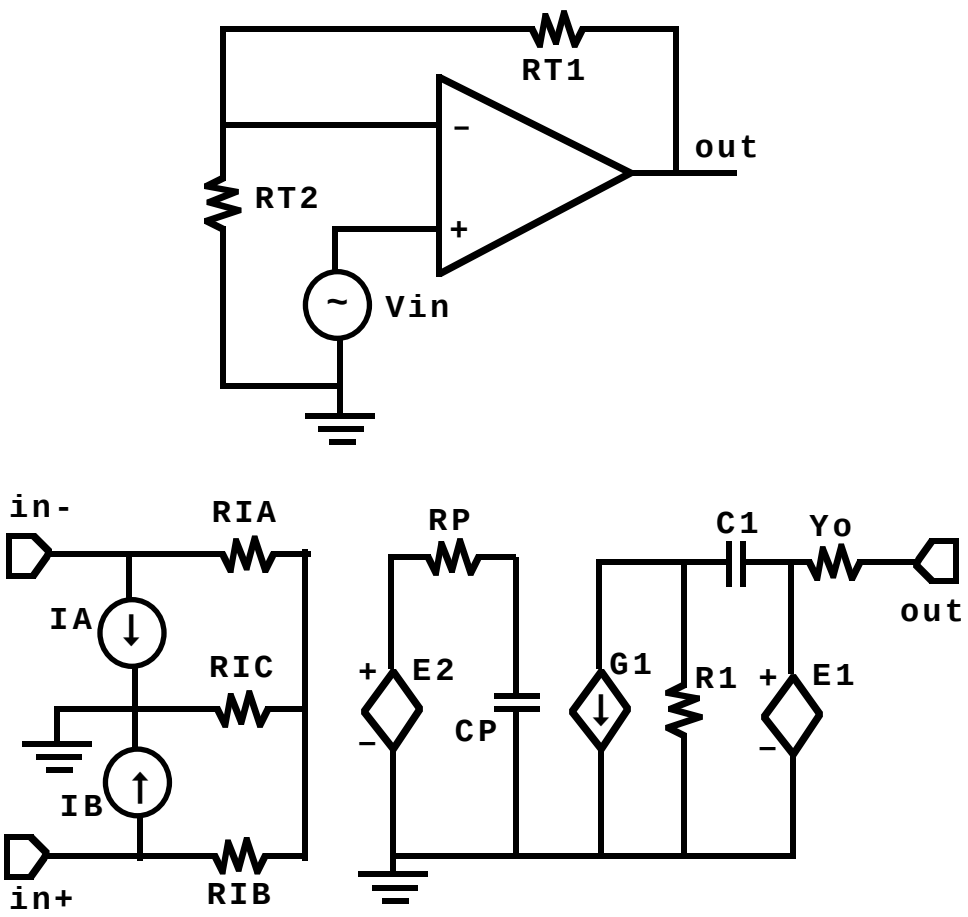
ac	tran	frequency
dc	period	time
dctran		

❑ Some examples



Op-Amp Macromodel (1)

Schematic

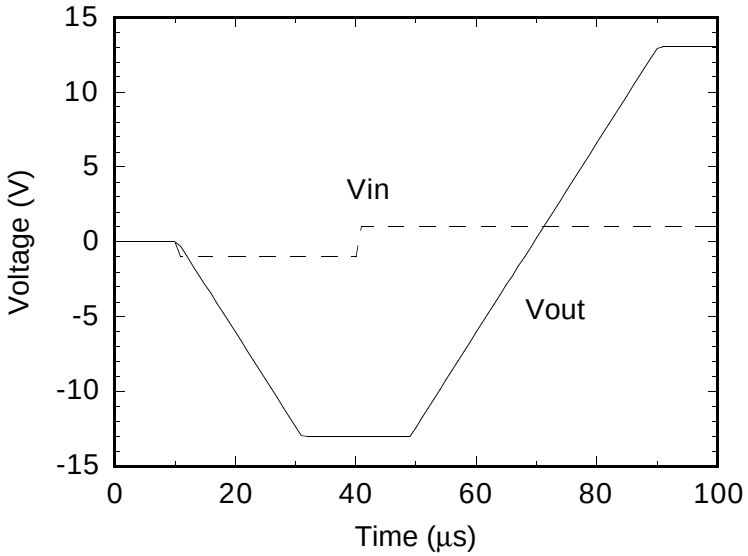


Netlist

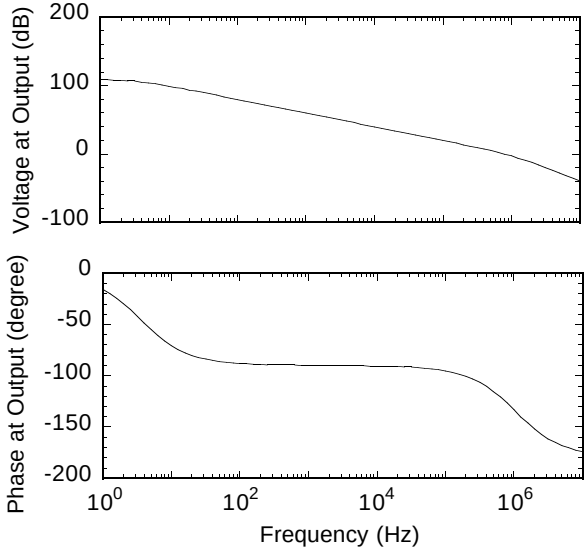
```
* Xsim netlist for op-amp macromodel
Xop_amp3 5 1 3 op_amp
.subckt op_amp 9 8 1
Yo 3 1 max(0.02 0.02)
E1 3 0 0 4 max(505 13)
C1 4 3 30p
R1 4 0 3meg
G1 4 0 0 6 tanh(185u 19u)
CP 6 0 1p
RP 7 6 150k
E2 7 0 8 9 1
RIC 0 10 100meg
RIB 9 10 1meg
RIA 8 10 1meg
IB 9 0 80n
IA 8 0 80n
.ends op_amp
Vin 5 0 pwl(0 0 10u 0 10.1u -1 40u
+ -1 40.1u 1 100u 1) ac (1)
RT2 0 1 1k
RT1 1 3 ac (100G) tran (100k)
.plot ac vdb 3
.plot ac vp 3
.plot tr v 5
.plot tr v 3
.tran 0 100u 1u
.ac 1 1e7 dec 10
```

Op-Amp Macromodel (2)

Transient Response

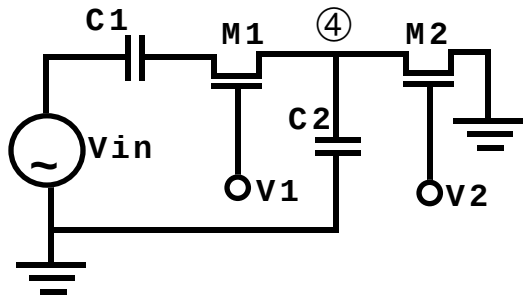


AC Response



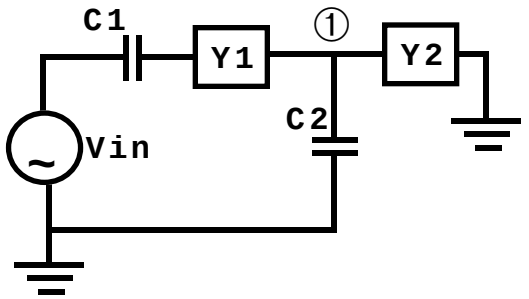
Switched-Capacitor Filter (1)

Complete Circuit




```
* Xsim netlist for complete circuit
Vin 6 0 sin(0 1 100k 0 0)
V1 1 0 pulse(5 0 0 0 0 500n 1u)
V2 7 0 pulse(0 5 0 0 0 500n 1u)
C1 6 5 1p
C2 4 0 1p
M1 5 1 4 4 nnn l=100u w=100u
M2 4 7 0 0 nnn l=100u w=100u
.model nnn nmos (level=2 vto=1)
.plot tr v 4
.tran 0 20u 0.1u
.status
.clear
```

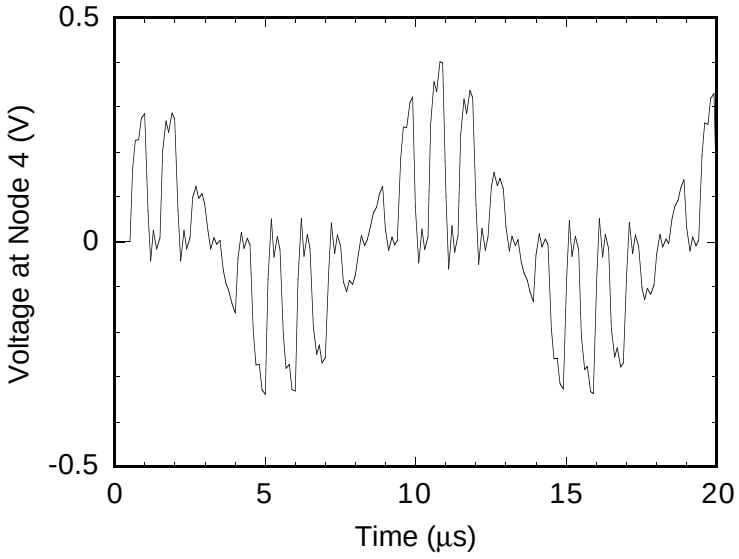
Behavioral Model




```
* Xsim netlist for behavioral model
Vin 7 0 sin(0 1 100k 0 0)
C1 7 3 1p
C2 1 0 1p
Y1 3 1 period(1u) (20u) time(500n) (0)
Y2 1 0 period(1u) (0) time(500n) (20u)
.generator freq=100k
.plot tr v 1
.tran 0 20u 0.1u
.status
.end
```

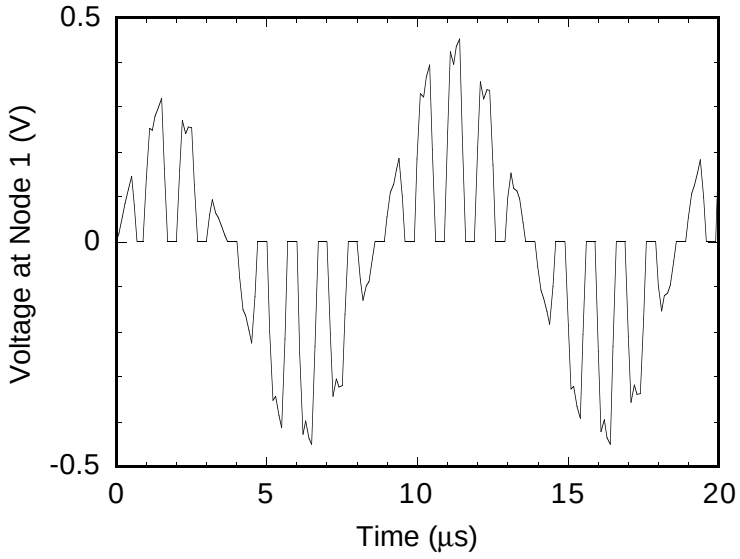
Switched-Capacitor Filter (2)

Complete Circuit



CPU: 0.69 sec.

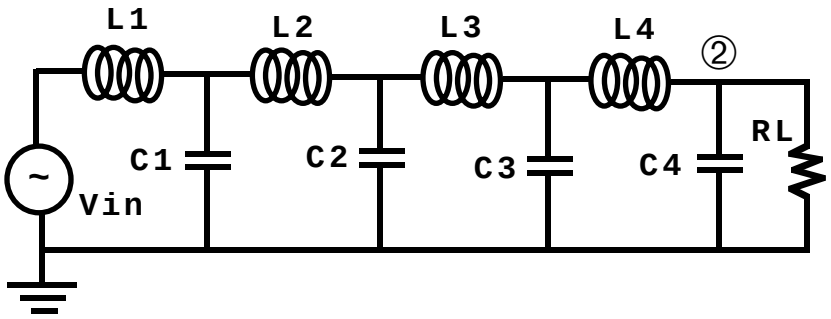
Behavioral Model



CPU: 0.18 sec.

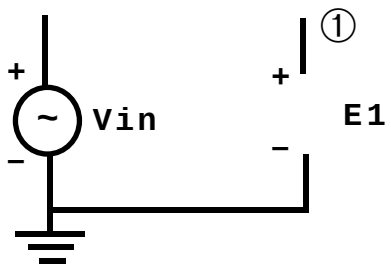
8th-Order Chebyshev Filter (1dB Ripple) (1)

Lumped Circuit




```
* Xsim netlist for lumped circuit
RL 0 2 1k
C4 2 0 173.033n
C3 3 0 326.857n
C2 4 0 341.435n
C1 5 0 332.984n
L4 2 3 234.769m
L3 3 4 268.176m
L2 4 5 270.899m
L1 5 6 217.899m
Vin 6 0 ac 1
.plot ac vdb 2
.plot ac vp 2
.ac 100 1000 dec 100
```

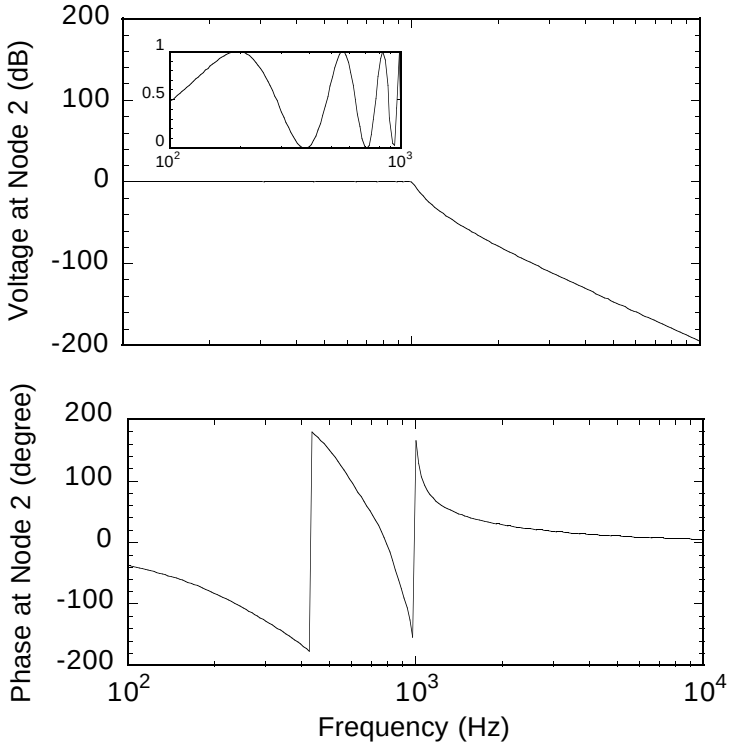
Behavioral Model



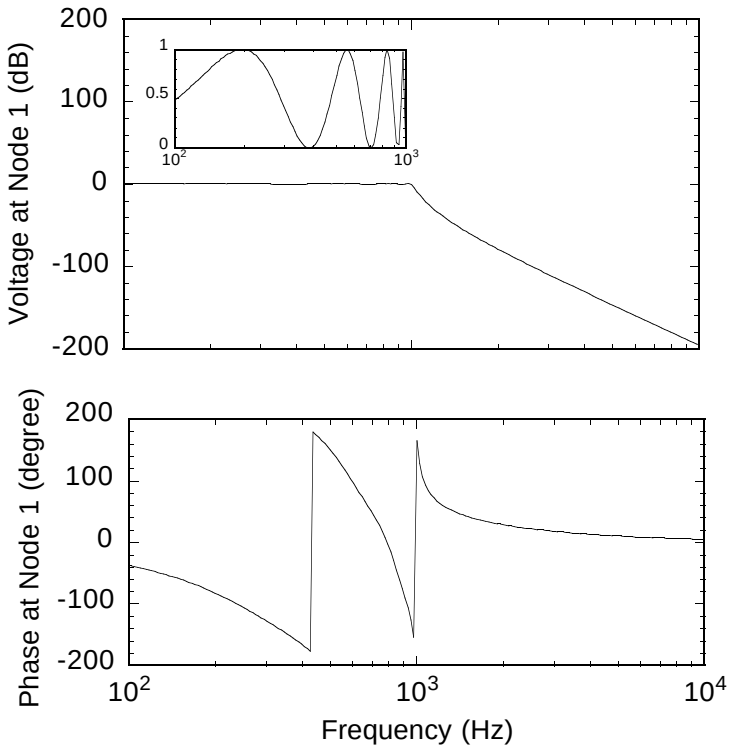

```
* Xsim netlist for behavioral model
E1 1 0 3 0 netfunction(1 1k 8 0.017227
+ 0.107345 0.447826 0.846824 1.836902
+ 1.655156 2.423026 0.919811 1
+ 0.017227)
Vin 3 0 ac 1
.plot ac vdb 1
.plot ac vp 1
.ac 100 1000 dec 100
```

8th-Order Chebyshev Filter (1dB Ripple) (2)

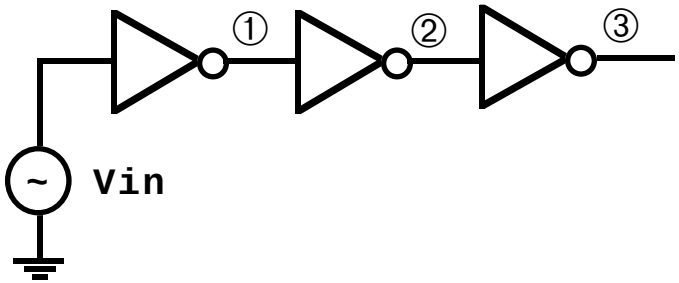
Lumped Circuit



Behavioral Model



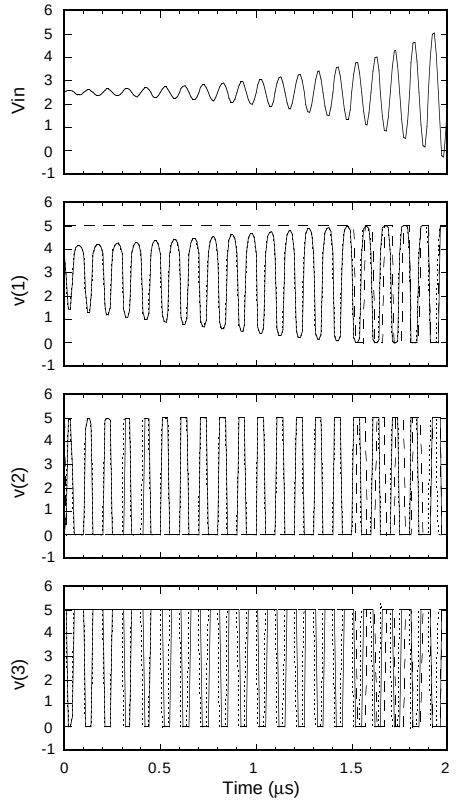
Implicit Mixed-Mode Simulation



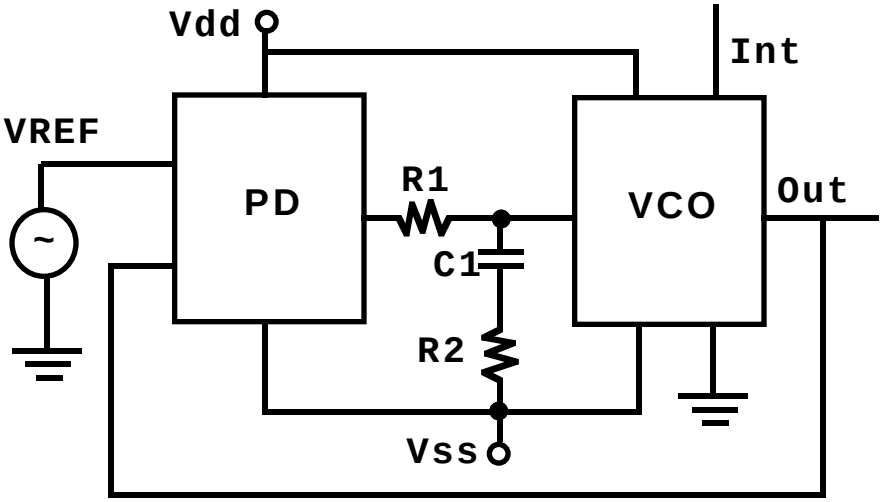
Mode	**CPU (sec.)**
Analog	3.57
Digital	0.82
Mixed	2.45 (30% speedup)

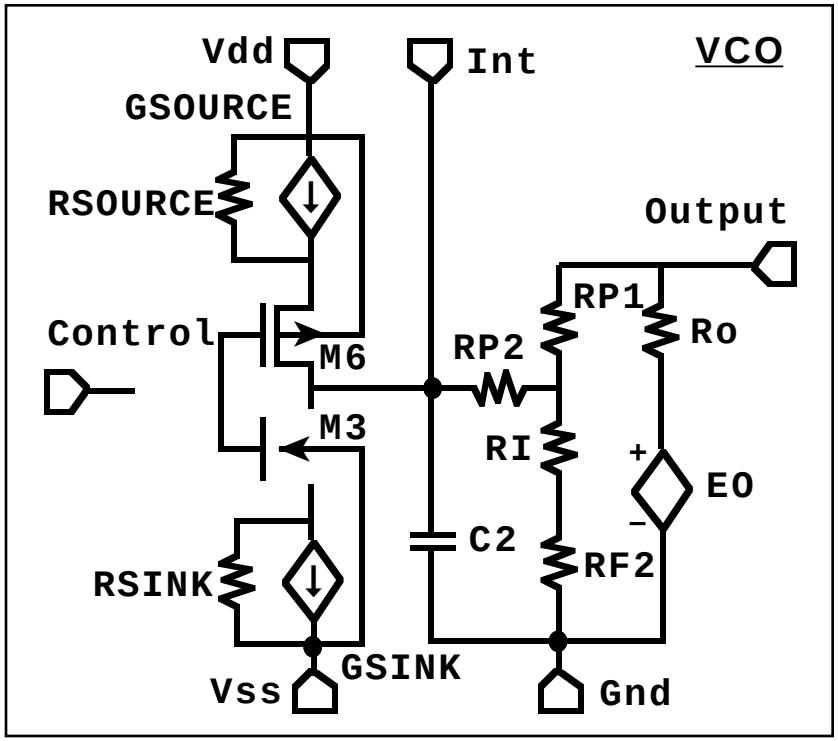
***In mixed mode*:**

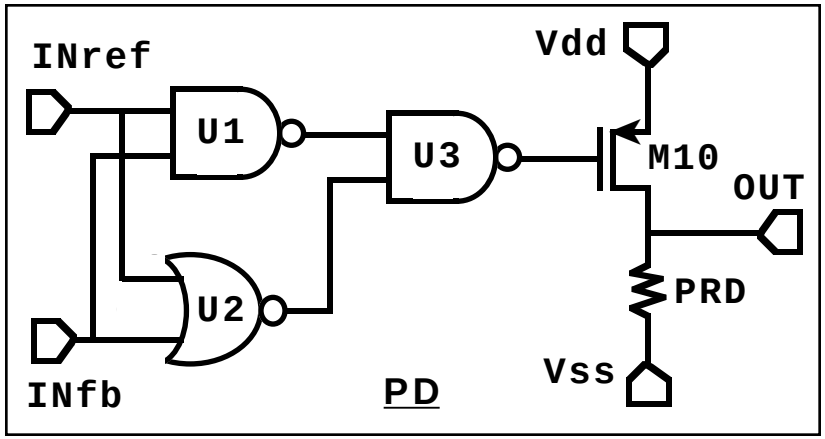
- U1 switched to digital after 1.86 µs
- U2 and U3 switched to digital after 0.45 µs



Mixed-Mode Phase-Locked Loop (1)







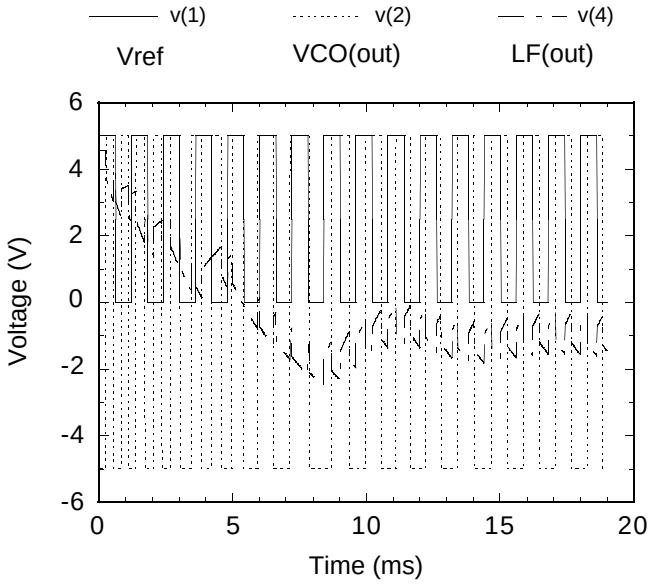
Mixed-Mode Phase-Locked Loop (2)


```
Xsim PLL netlist
* Phase detector
X2   1  2  3   31  32  PD
* VCO
X1   4  2  31  32  0  40  VCO
* Low pass filter
R1   3  4  1Meg
C1   4  9  3n
R2   9  32  100K
* Clock reference
V1REF 1 0 pulse(0 5 0 20u 20u 60u 0.0012)
* Power supplies
VDD   31 0 dc ( 5)
VSS   32 0 dc (-5)
*** Phase Detector SubCkt ***
* Input(1) Input(2) Output(6) VDD(7) VSS(8)
.subckt PD 1 2 6 7 8
U1  0 3 1 2 cmos nand2 cmos-nand2
U2  0 4 1 2 cmos or2 cmos-or2
U3  0 5 3 4 cmos nand2 cmos-nand2
M10 6 5 7 7 pmos l=2u w=50u ad=20p as=20p pd=24u
+ ps=24u
RPD 6 8 10K
.ends
*** VCO Subckt ***
* Control(33) Output(5) VDD(31) VSS(32)
* Gnd(10) Int(6)
.subckt VCO  33  5  31  32  10  6
* Comparator with hysteresis
* -input(1) +input(6) output(5) gnd(10)
* Op-amp model: AVo~277K RI=2Meg RO=50 Vo=+/-5V
```

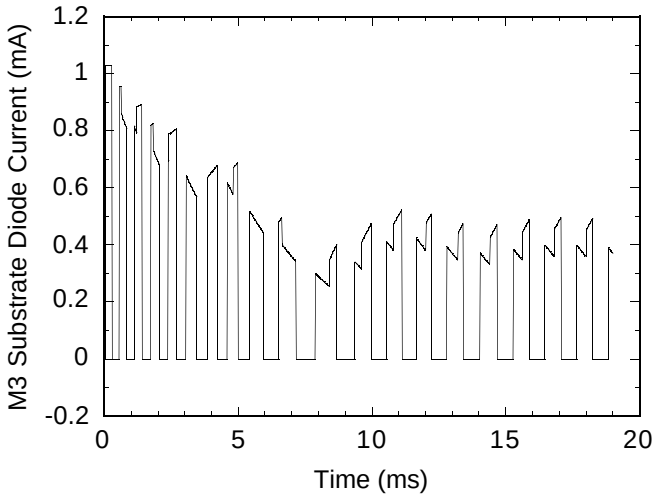
```
RI  1 2 20Meg
Ro  4 5 50
E1  4 10 2 1 max(25K 5)
* Feedback elements for hysteresis
RF2 1 10 1
RP1 5 2 4Meg
RP2 2 6 1Meg
* Charge element for current
C2  6 10 100n
* Transconductance and current steering stage
RSINK  32 35 10Meg
RSORCE 36 31 10Meg
GSORCE 31 36 33 32 110u
GSINK  35 32 33 32 110u
M3  6 5 35 32 nmos l=2u w=20u ad=20p as=20p pd=24u
+ ps=24u
M6  6 5 6  31 pmos l=2u w=40u ad=20p as=20u pd=24u
+ ps=24u
.ends
* Analysis
.lib pll
.IC v(40)=-1
.options mode=analog itl4=500
.plot tran v(1) v(2) v(3) v(4) v(40)
.probe tran vdsb M6.X1
.probe tran idsb M6.X1
.probe tran vdsb M3.X1
.probe tran idsb M3.X1
.tran 0 19m 0.01m
.status
.end
```

Mixed-Mode Phase-Locked Loop (3)

Transient Response



Internal Probe



Xsim: Applications

- **Large digital circuits**
  - Using full “digital” mode for logic synthesis with vendor-supplied delay parameters, and then, subcircuit model for back-annotated timing verification

- **Mixed-signal interface circuits**
  - Using “mixed” mode for data-converter systems with large ratio of digital/analog circuitry

- **PCB-level system simulation**
  - Large systems with mixed digital (including mixed logic families) and analog functional blocks