EECS 427
Lecture 5: Logical Effort

Reading: handout

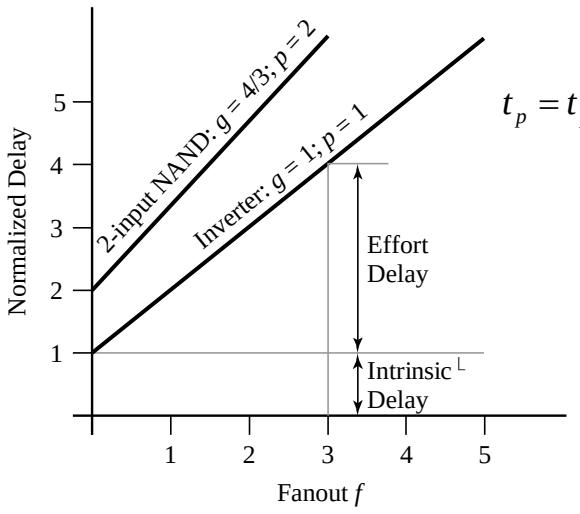
Reminders

- Seminar announcement: Dr. Michael McCorquodale, CTO and founder, Mobius Microsystems
  - Topic: Straight Down the Crooked Path – The Dynamic Process of Commercializing Research
  - Friday 9/25, 2:30-3:30 pm, 1200 EECS
- CAD3 will be done in teams. CAD3 is due next Wednesday
  - Start early
  - Try not to skip lecture
- Looking ahead:
  - HW3 – Project proposal due Wednesday 10/7, 2 weeks away
  - Quiz1 Wednesday 10/14, 3 weeks away

Last Time

- Intro to logical effort
  - g: Logical effort
  - f: Electrical effort
  - p: Parasitic delay

Delay components review

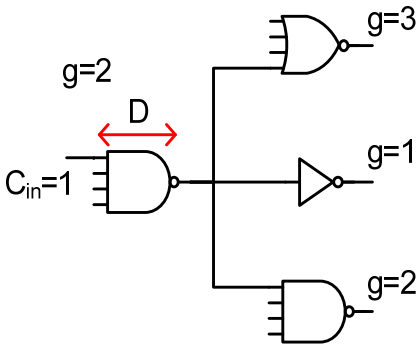


$$t_p = t_{p0}(p + g \cdot f)$$

I. Sutherland, *et al*, Logical Effort, Academic Press, 1999

More complex circuit

Every gate is sized according to a 2:1 min-sized inverter



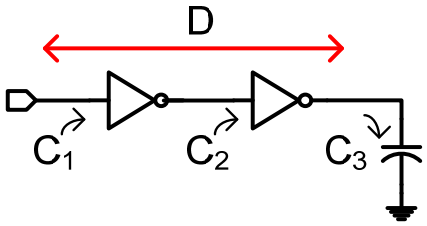
$$f = \frac{C_{gate,nor4} + C_{gate,inv} + C_{gate,nand4}}{C_{gate,nand4}} = 3$$

$$g_{nand4} = 2$$

$$p_{nand4} = 4$$

$$t_p = t_{p0}(p_{nand4} + g_{nand4} f) = 10 t_{p0}$$

Multistage effort



Path effective fan-out F

$$F = \frac{C_3}{C_1}$$

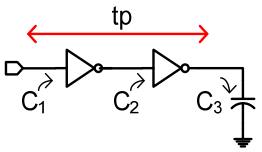
$$f_1 = \frac{C_2}{C_1}$$

$$f_2 = \frac{C_3}{C_2}$$

$$\Rightarrow \quad F = f_1 f_2$$

Minimum delay

Total delay $t_p = p_1 + g_1 f_1 + p_2 + g_2 f_2$ (tp_1, tp_2)



Substitute F $t_p = p_1 + g_1 f_1 + p_2 + g_2 \frac{F}{f_1}$

$$\min\{t_p\} \Rightarrow \frac{\partial t_p}{\partial f_1} = g_1 - g_2 \frac{F}{f_1^2} = 0$$ Minimize the delay

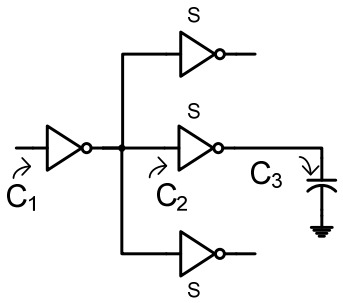
$$g_1 - g_2 \frac{f_2}{f_1} = 0$$ Solve for the minimum delay

$$g_1 f_1 = g_2 f_2$$

Gate effort $h_1 = h_2$

Delay is optimal when stage efforts are equal

Branching



Define **path effort** H

$$H = h_1 h_2 = g_1 f_1 \cdot g_2 f_2$$
$$= (g_1 g_2) \cdot (f_1 f_2)$$
$$= G \frac{3C_2}{C_1} \cdot \frac{C_3}{C_2} = 3GF$$

G: **path logical effort**
F: **path effective fan-out**

Branching effort

$$b_i = \frac{C_{on-path} + C_{off-path}}{C_{on-path}}$$

$$C_{on-path} = C_2$$
$$C_{off-path} = 2C_2$$
$$b_2 = \frac{3C_2}{C_2} = 3$$

Equivalent Path Efforts

$$F = \frac{C_{out}}{C_{in}}$$

$$B = \prod b_i$$

$$FB = \prod f_i$$

$$G = \prod g_i$$

Path Effort

$$H = GFB = \prod g_i f_i$$

Path Effort:

- Does not change with added inverters
- Does not depend on sizes, but on topology

Minimum Delay

$$H = GFB = \prod g_i f_i$$

Stage effort of stage *i*: $h_i = g_i f_i$

Optimal stage effort: $\hat{h} = h_i$

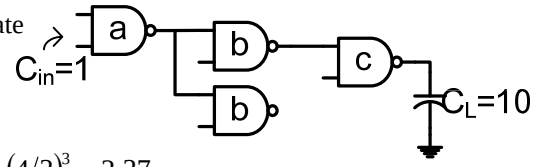
For N stages: $H = \hat{h}^N \Rightarrow \boxed{\hat{h} = H^{1/N}}$

Minimum Delay: $t_p = t_{p0} \sum_i (p_i + g_i f_i)$

$$t_p = \boxed{t_{p0} (\sum_i p_i + N H^{1/N})}$$

Example to compute min. delay

Unit sized nand2 gate



$G = g_a g_b g_c = (4/3)^3 = 2.37$

$F = C_L / C_{in} = 10$

$B = 2$

$H = GFB = 47.4$

$\hat{h} = H^{1/3} = 3.62$

$t_p = t_{p0}(3p_{nand2} + 3\hat{h})$

$t_p = 16.9 t_{p0}$

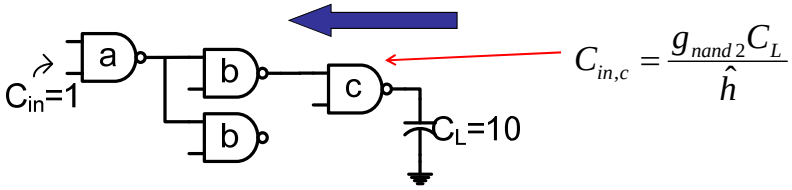
Stage sizing

After computing $\hat{h}$

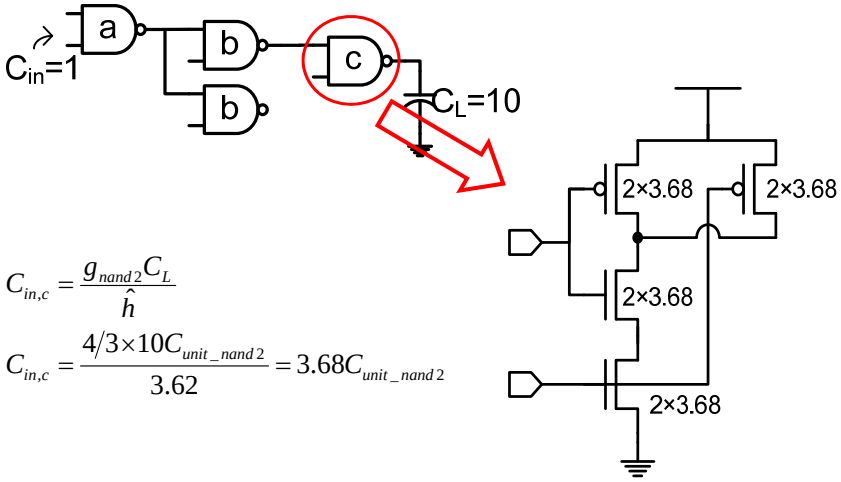
$f_i = \hat{h} / g_i = C_{out} / C_{in}$

$$C_{in} = \frac{g_i C_{out}}{\hat{h}}$$

Work backwards to size each gate



Stage sizing example



$$C_{in,c} = \frac{g_{nand2} C_L}{\hat{h}}$$

$$C_{in,c} = \frac{4/3 \times 10 C_{unit\_nand2}}{3.62} = 3.68 C_{unit\_nand2}$$

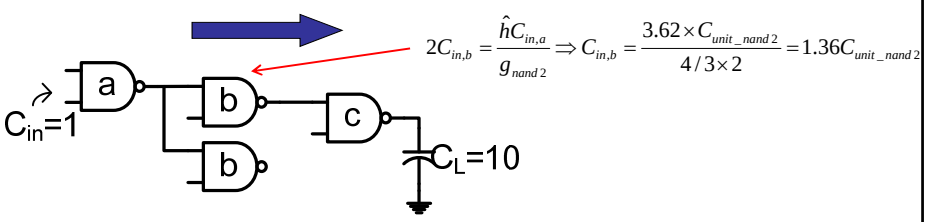
Stage sizing

After computing $\hat{h}$

$$f_i = \hat{h} / g_i = C_{out} / C_{in}$$

$$C_{out} = \frac{\hat{h} C_{in}}{g_i}$$

Can also work forward to size each gate



$$2C_{in,b} = \frac{\hat{h} C_{in,a}}{g_{nand2}} \Rightarrow C_{in,b} = \frac{3.62 \times C_{unit\_nand2}}{4/3 \times 2} = 1.36 C_{unit\_nand2}$$

Calculating optimal # of stages

- Path effort *H* can be used to determine the optimal number of stages
  - Assuming we add n_2 inverters
    - New number of stages $N=n_1+n_2$
    - G, F, B don't change – H is fixed
    - But intrinsic delay increases

$$t_p = (n_1 + n_2)F^{1/(n_1+n_2)} + \sum_i^{n_1} p_i + n_2 p_{inv}$$

Optimum is technology dependent

Summary of the terminology

Gate level

Parasitic delay	p
Logical effort	g
Electrical effort	$f = \frac{C_{in}}{C_{out}}$
Stage effort	$h = gf$
Stage delay	$t_p = h + p$

Path

Path electrical effort	$F = \frac{C_{out-path}}{C_{in-path}}$
Path logical effort	$G = \prod g_i$
Branch effort	$B = \prod b_i$
	$\prod f_i = BF$
Branching factor	$b_i = \frac{C_{on-path} + C_{off-path}}{C_{on-path}}$

Method

Path effort $H = GFB$

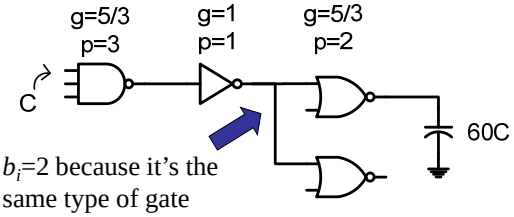
Optimal stage effort $\hat{h} = H^{1/N}$

Optimal path delay $t_p = \sum p_i + NH^{1/N}$

Stage sizing $C_{out,i} = f_i C_{in,i} = \frac{\hat{h}}{g_i} C_{in,i}$

1. Compute path effort
2. Compute optimal stage effort
3. Add buffers (determine optimal number of stages)
4. Compute fan-out *f* of each stage
5. Size individual gates (working backward or forward)

Method example 1/3



b_i=2 because it's the same type of gate

$G = \frac{5}{3} \cdot 1 \cdot \frac{5}{3} = \frac{25}{9}$

$B = 1 \cdot 2 \cdot 1 = 2$

$F = 60$

$\sum p_i = 3 + 1 + 2 = 6$

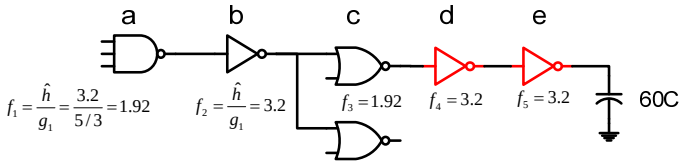
$H = GFB = 333.33 \rightarrow \hat{N} = 5$

$\hat{h} = (333.33)^{1/5} = 3.2$

$t_p = t_{p0}(5 \times 3.2 + 6 + 2) = 24 t_{p0}$

- Choose # of stages such that stage effort is close to 4.
- Add 2 inverters to the existing 3 stages

Method example: Sizing 2/3



$$C_{in,e} = \frac{C_{out,e}}{f_5} = \frac{60C}{3.2} = 18.75C$$

$$C_{in,d} = \frac{18.75C}{3.2} = 5.86C$$

Branching

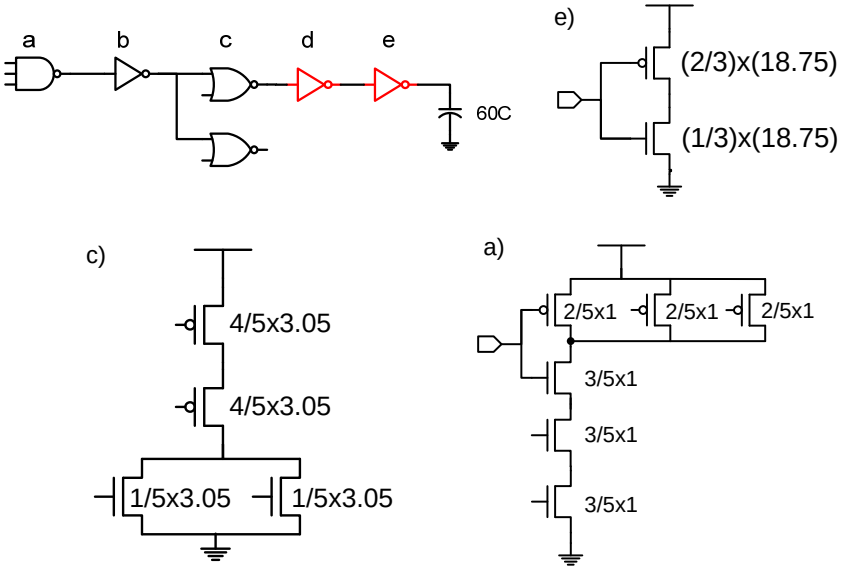
$$C_{in,c} = \frac{5.86C}{1.92} = 3.05C$$

$$C_{in,b} = \frac{3.05C \times 2}{3.2} = 1.9C$$

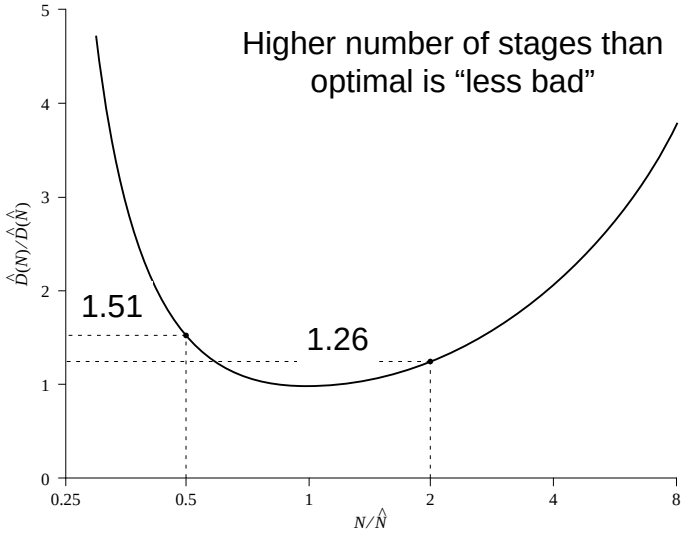
$$C_{in,a} = \frac{1.9C}{1.92} = 0.989C \approx C$$

Always check this

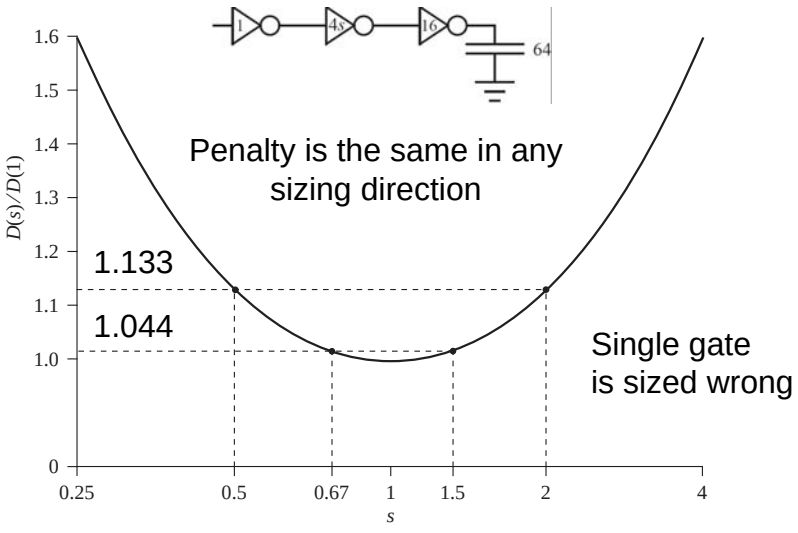
Method example: Gate sizing 3/3



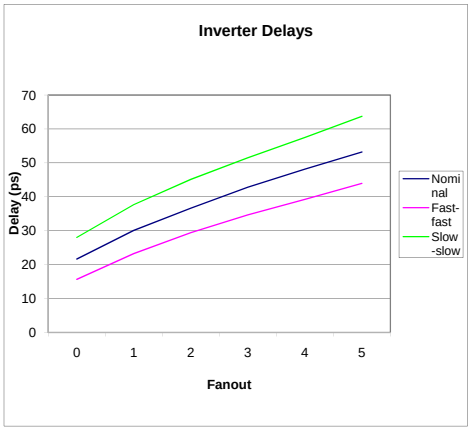
Wrong number of stages

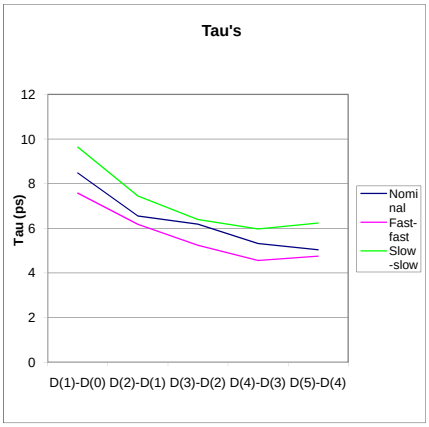


Wrong gate size

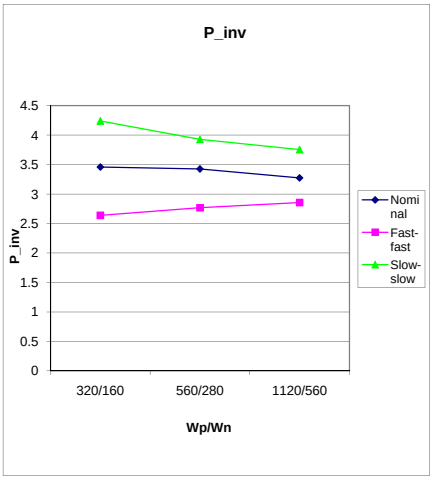


tp0 for our 130nm technology





Tau and Intrinsic Delay (Pinv) for 130nm Tech



	Wp/Wn	Nom	Fast	slow
Average Tau value	320/160	6.26	5.74	6.98
	560/280	6.31	5.66	7.14
	1120/560	6.37	5.55	7.14
Average Pinv	320/160	3.46	2.64	4.24
	560/280	3.27	2.86	3.75
	1120/560	3.42	2.77	3.93

Best number of stages $p_{inv} = 3.38\tau$

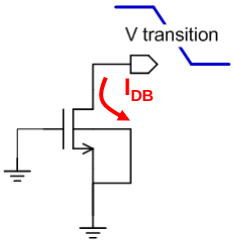
Path effort H	Best number of stages	Min. delay t_p	Stage effort h
0		3.38	
	1		0-9.57
9.57		12.9	
	2		3.09-7.38
54.4		21.5	
	3		3.79-6.65
294		30.1	
	4		4.14-6.29
1563		38.7	
	5		4.35-6.07
8246		47.3	
	6		4.49-5.93
43327		55.8	
	7		

IBM 0.13um Characteristics

Char / Unit distance	Parameter
NMOS C_j	1.06fF/um (W)
PMOS C_j	1.03fF/um (W)
NMOS C_g	1.57fF/um (W)
PMOS C_g	1.44fF/um (W)
NMOS R_{eq}	43.24 kΩ/um (W)
PMOS R_{eq}	86.95 kΩ/um (W)

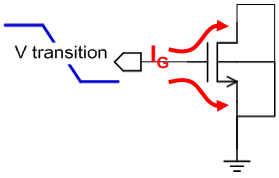
* All values obtained from device simulation

Device Simulations



Junction Capacitance

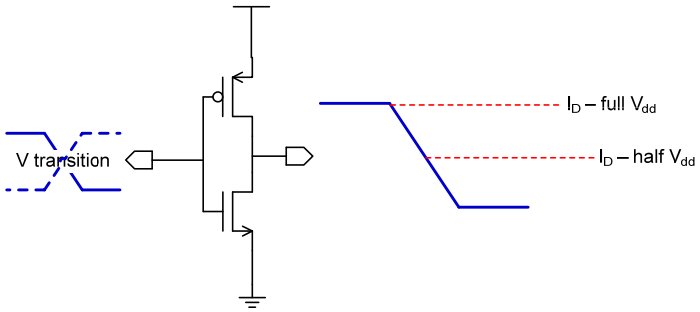
$$\int I\,dt = Q = C_j V$$



Gate Capacitance

$$\int I\,dt = Q = C_g V$$

Device Simulations



$$R_{eq} = \frac{R_{eq}^{full} + R_{eq}^{half}}{2}$$

$$R_{eq}^{full} = \frac{V_{DD}}{I_D^{full}} \qquad R_{eq}^{half} = \frac{V_{DD}}{I_D^{half}}$$

IBM 0.13um Characteristics

Char / Unit distance	Parameter	
Poly Resistance	58.33Ω/um (ℓ)	
M1 Resistance	0.44Ω/um (ℓ)	
M2-M6 Resistance	0.32Ω/um (ℓ)	
Upper layer Metals	0.09Ω/um (ℓ)	
	Plate Above, Below	**Isolated**
Poly Capacitance	0.1691 fF/um	0.1219 fF/um
M1 Capacitance	0.2619 fF/um	0.1565 fF/um
M2-M6 Capacitance	0.2341 fF/um	0.1592 fF/um
Upper layer Metals	0.1970 fF/um	0.1150 fF/um

* All values obtained from design manual

IBM 130nm Characteristics

MOS Type	Parameter	Calculated	Simulated	% Error
NMOS	C_g	2.03fF/um	1.57fF/um	22.7%
PMOS	C_g	1.94fF/um	1.44fF/um	25.8%
NMOS	C_j	1.209fF/um	1.06fF/um	12.3%
PMOS	C_j	1.244fF/um	1.03fF/um	17.2%
NMOS	t_p (FO4)	42.8ps	35.5ps	21%
PMOS	t_p (FO4)	86.1ps	68.2ps	26%

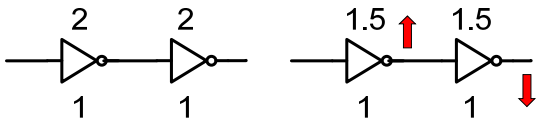
* Calculated $t_p = 0.69 * R_{eq} * C_L$

$C_L = 4C_{g,P} + 4C_{g,N} + C_{j,N} + C_{j,P}$

* Simulated t_p = 50% - 50% transition

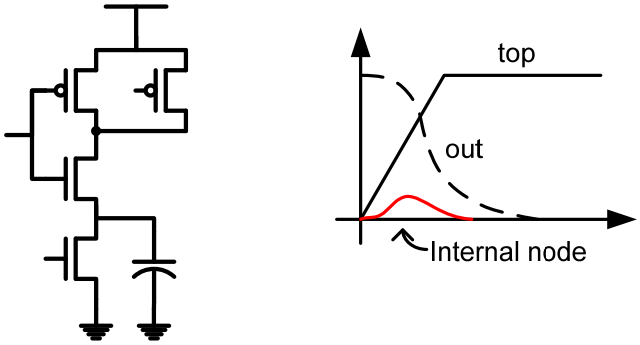
P/N ratio

- Why use P/N = 2?
  - Noise margins are balanced
  - Equal slopes
- How about P/N = 1.5?



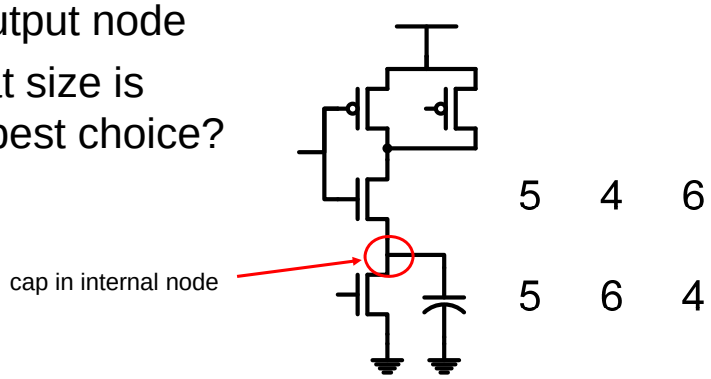
Limitations – Internal capacitance

- Capacitance in internal nodes
- Body effect



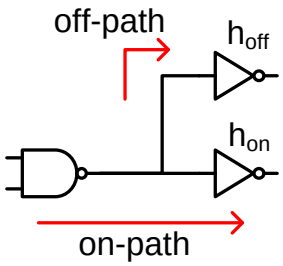
Limitations - Tapering

- Transistor sizes in stack are different
- The latest arriving input should be closest to output node
- What size is the best choice?



Limitations – Branching

- Assume that the size of the off-path gate tracks the size of the gate on-path



- Sizing one “critical” path of a branch may make the other paths worse

Limitations – Series Devices

- Models 2 series devices of size 2 to be equivalent in drive strength to a single device of size 1
  - Not very accurate
- Solution – simulate gates to directly find logical efforts

Limitations – Slope & Interconnect

- Ignores impact of input slope on stage delay
- Interconnect capacitances are side loads. They do not necessary scale with gates and cannot be lumped in the logical effort formulation

Limitations - Scaling

- Scaling is not linear with width

$$t_p \approx 0.69RC$$

$$C = k_c S \Rightarrow C = k_c S + k_p$$ Perimeter

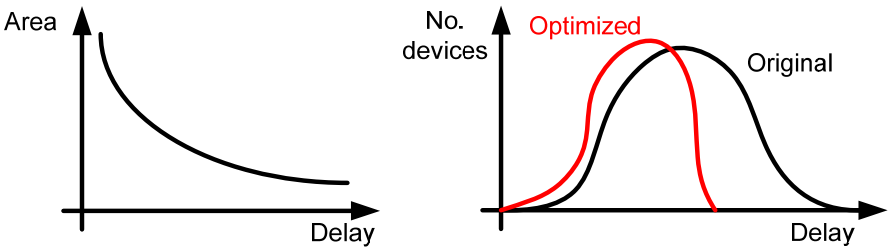
$$R = \frac{k_R}{S} R \Rightarrow \frac{k_R}{S} + k_v$$ Narrow width effects, Process variations

- Parasitic delay of multi-input gates usually much less than simple model predicts
  - Diffusion sharing, input dependencies

Sizing tool

- Tool: TILOS [Dunlop 89]
  - Start with all transistors of min. size
  - Find critical path (Optimize path)
  - Compute delays
  - Increase size of transistors in “critical path”
  - Size path with best sensitivity
  - Repeat
  - Goal of path distribution → All paths equal in length…

Area - Delay



But the impact of process variations can be worse for the optimized paths – more on this later in the course

Summary

- Logical effort is useful for thinking of delay in circuits
  - NANDs are faster than NORs in CMOS
  - Paths are fastest when effort delays are ~4
  - Path delay is weakly sensitive to stages, sizes
  - But using fewer stages doesn't mean faster paths
  - Delay of path is about $\log_4 F$ FO4 inverter delays
  - Inverters and NAND2 best for driving large caps
- Provides language for discussing fast circuits
  - But requires practice to master