

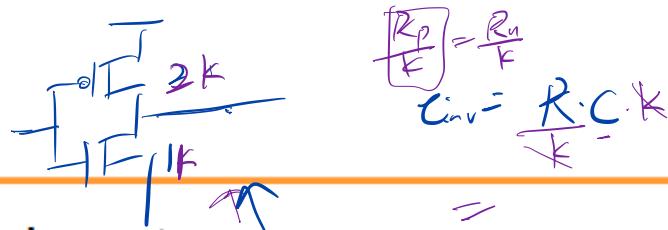
GEC Workshop | Summer 2022
Discussion 4

Digital Integrated Circuits



Mentor: Sida (Star) Li
UCLA Electrical & Computer Eng.

Normalized Delay



Strategy: normalize to a time constant of an inverter

- ◆ Approach 1: normalize to fictitious “technology time constant”

$$\frac{\text{Delay}}{\tau_{INV}} = \frac{\tau_{gate}}{\tau_{INV}} \left(\gamma_{gate} + \frac{C_{out}}{C_{in,gate}} \right)$$

- ◆ Normalized delay: $d = g \cdot (\gamma_{gate} + h)$

$$d = gh + p$$

gh - load
p - parasitic $\neq f(w)$

Logical effort terms

- ◆ Logical effort (g)

$$g = \frac{R_{gate} \cdot C_{in,gate}}{R_{INV} \cdot C_{in,INV}}$$

- ◆ Electrical fanout (h)

$$h = \frac{C_{out}}{C_{in,gate}}$$

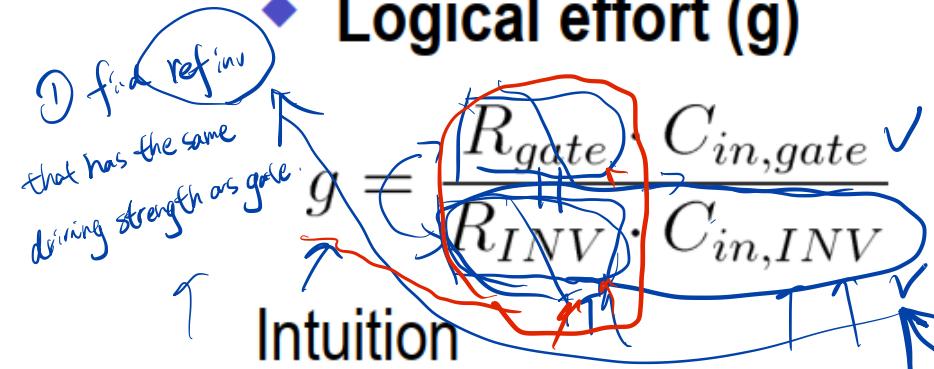
- ◆ Parasitic delay (p)

$$p = \frac{C_{par,gate}}{C_{par,INV}} \cdot \gamma_e$$

The Meaning of Logical Effort Terms: Summary

Logical effort terms

◆ Logical effort (g)



Intuition

Electrical fanout (h)

$$h = \frac{C_{out}}{C_{in,gate}}$$

Parasitic delay (p)

$$p = \frac{C_{par,gate}}{C_{par,INV}} \cdot \gamma_{INV}$$

◆ Logical effort (g)

- R_{on} ratio for equal C_{in} (or, equivalently, C_{in} ratio for equal R_{on})

◆ Electrical fanout (h)

- C_{out} / C_{in} ratio (gate cap only, diffusion counts in the p term)

◆ Parasitic delay (p)

- Ratio of parasitic capacitances for equal R_{on}

equal R_{on} :

$$g = \frac{C_{gate}}{C_{INV}}$$

Calculating Logical Effort – Simple Example

DEF: Logical effort is the ratio of the input capacitance to the input capacitance of an inverter delivering the same output current

- ① find ref inv

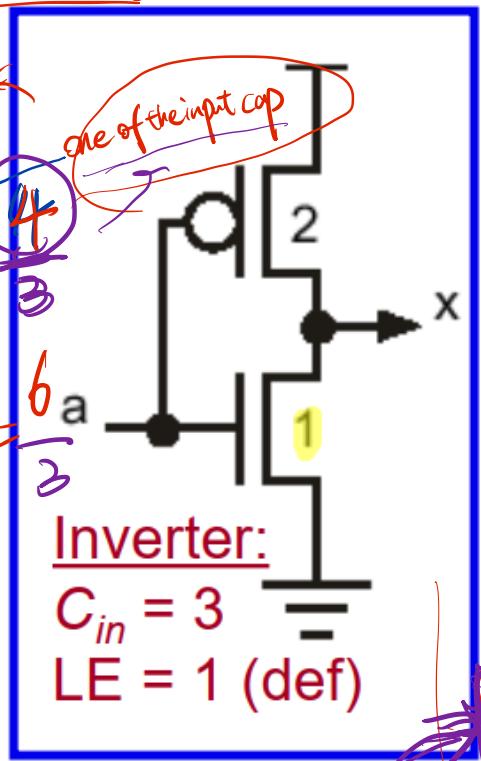
that has the same driving strength

as logic gate

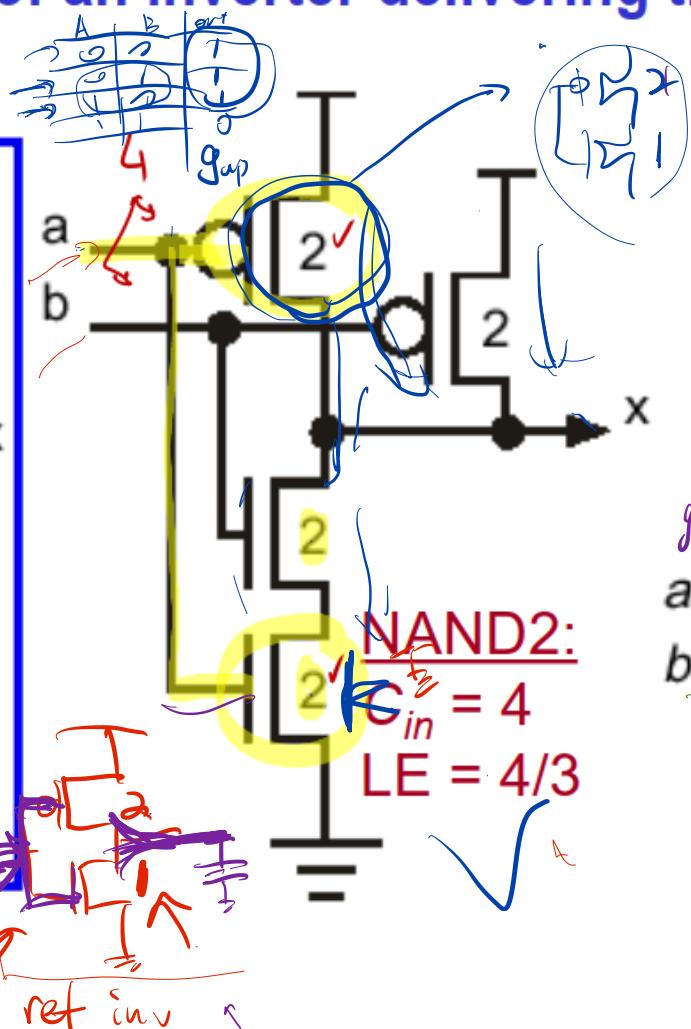
$$R_{\text{refine}} = R_{\text{rigid}}$$

$$② g = \frac{\text{Circate}}{\text{Cirfinv}}$$

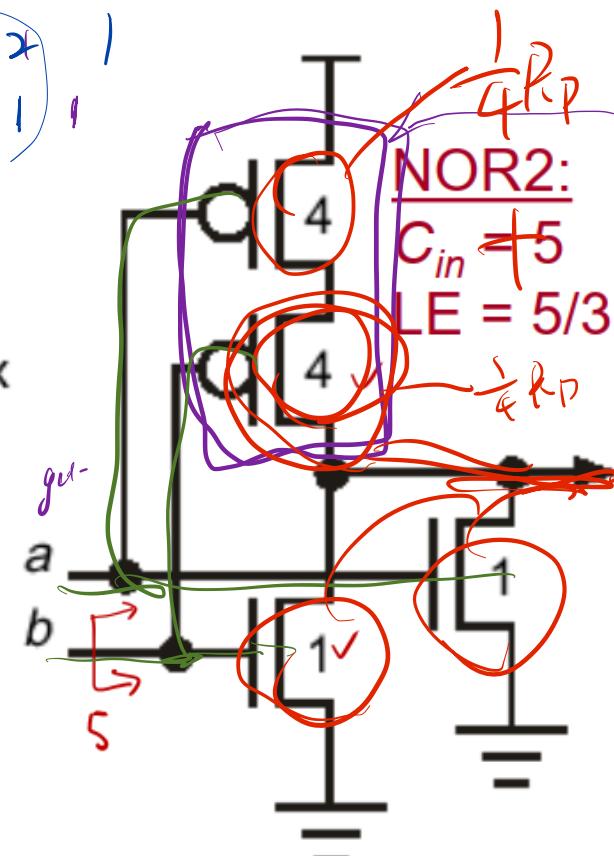
$$\textcircled{3} \quad P = \frac{\text{Cap, got}}{\text{Cap in}}$$



Reference



$$\text{NAND2: } C_{in} = 4 \quad LF = 4/3$$



$$P = \frac{\text{Oper goat e}}{\text{Oper inv}} ?$$

$$= \frac{6}{3} \geq 2$$

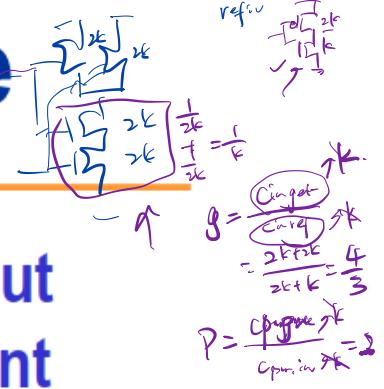
$$= \frac{6}{3} \geq 2$$



$$g = \frac{\text{Cingote}}{\text{Cinrefus}}$$

$$P = \frac{\text{Gpar grade}}{\text{Gpar inv}} ?$$

$$= \frac{6}{3} \geq 2$$



Calculating p Including Series Stacking

12 C diff

What about the intermediate nodes?

- One way to account for them is to use an “effective” p .
- For example: NOR pull up of B input

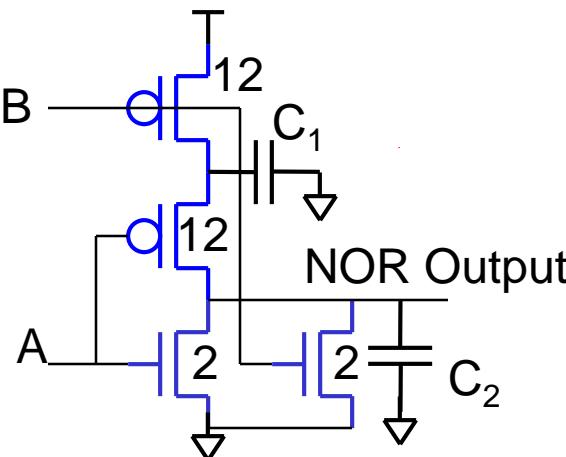
- $R_{NOR} = 2 * R_{PMOS}$
- Delay = $(R_{NOR}/2) * C_1 + R_{NOR} * C_2 + R_{NOR} * C_{load}$
- $p_{BUP} = [(R_{NOR}/2) * C_1 + R_{NOR} * C_2] / (R_{inv} * C_{inv})$ (where $R_{inv} = R_{gate}$)
- $p_{BUP} = (C_1/2 + C_2) / C_{inv}$

Using $C_{S/D} = 0.5C_G$

$C_1 = 6C_o$ (shared)

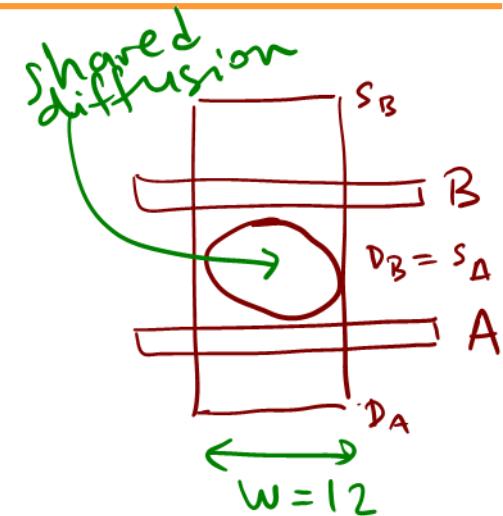
$C_2 = 8C_o$

$p_{BUP} = 11/4$

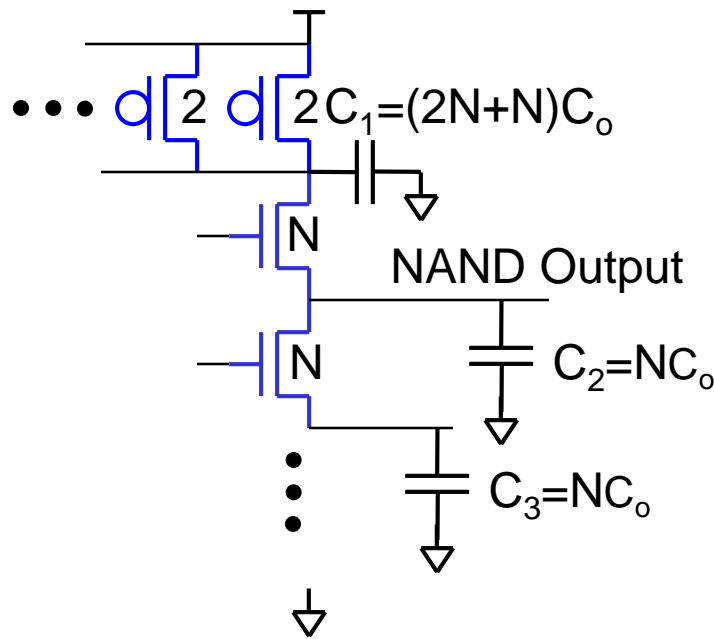


Note: this increased accuracy requires different p 's for different input AND pull up/down.

This class: simplify by ignoring these nodes



Generalize N-input NAND



- ◆ **Output load = 3N**
 - N size-2 PMOS=2N
 - 1 size-N NMOS = N
- ◆ **Intermediate load = N (shared)**
- ◆ **Total pull down delay**
 - $T = R(3NC_o) + \sum_{i=1-N-1} \{(iR/N)*NC_o\}$
 - $d (\text{norm}) = 3N + (N^2/2 - N/2)$
 - $p = (N^2/2 - N/2)$
 - Proportional to $N^2!!!$
- ◆ **This is bad news for long stacks**
 - Even worse for PMOS (NOR)
 - Reality is even worse since C_{GS} makes each intermediate node capacitance $> NC_o$

A Catalog of Gates

Gate Type	g for Different number of inputs					
	1	2	3	4	5	n
Inverter	1					
NAND		4/3	5/3	6/3	7/3	(n+2)/3
NOR		5/3	7/3	9/3	11/3	(2n+1)/3
Multiplexer	2	2	2	2	2	2
XOR,XNOR	4	12	32			

Gate Type	Parasitic delay
Inverter	p_{inv}
n-input NAND	np_{inv}
n-input NOR	np_{inv}
n-way Multiplexer	$2np_{inv}$
2-input XOR,XNOR (sym)	$n2^{n-1}p_{inv}$

$$\beta = \mu = 2$$

Mux is tri-state inverters shorted together.

XOR assumes that input is bundled (a,a')

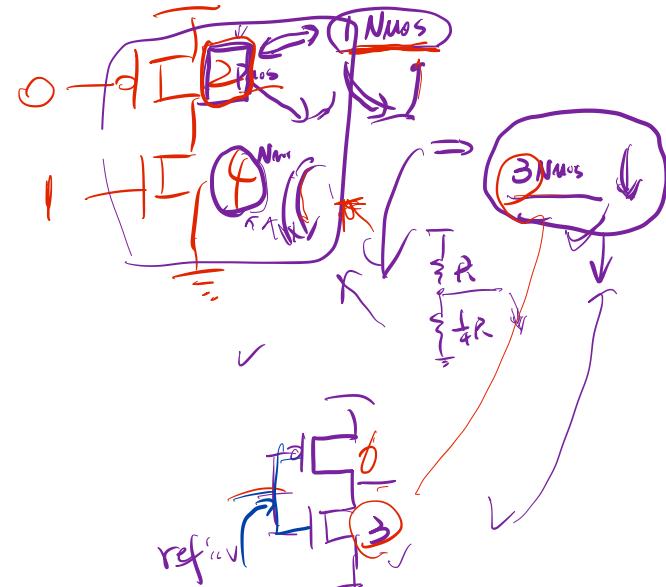
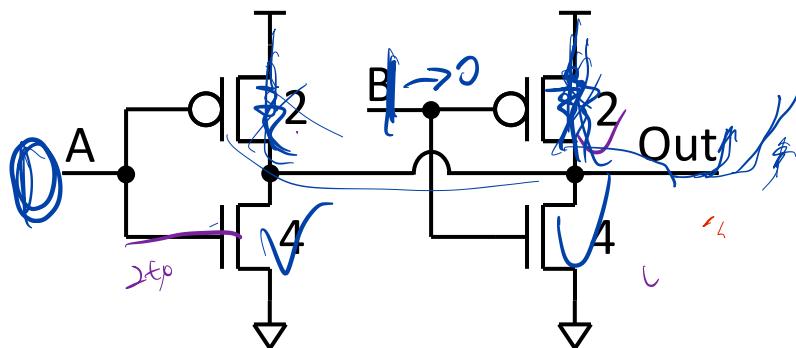
$$p_{INV} \sim 1$$

p_{GATE} in this table does not include intermediate nodes.

PROBLEM 1: Logical Effort (10 pts)

Consider the following "Ganged" CMOS gate. Assume $k = m_n/m_p = 2$.

$$P \sum_{i=1}^2 \frac{1}{m_i}$$



a. What is the logic function that this gate implements? (4p)

A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1

$$g_{DN} = \frac{C_{igate}}{C_{invref}} = \frac{2+4}{6+3} = \frac{6}{9} = \frac{2}{3}$$

NoR

$$\text{Out} = \overline{A+B} \Leftrightarrow \overline{A} \cdot \overline{B}$$

b. Calculate g_{UP} and g_{DN} . (6p)

g_{UP}

$$\frac{1}{2} \left(\frac{1}{2} \right) = \frac{1}{4} R_p \rightarrow \frac{1}{4} \frac{1}{4} = \frac{1}{16}$$

$$g_{UP} = \frac{C_{igate}}{C_{invref}} = \frac{6}{4+2} = 1$$

g_{DN}

1st Case 2, 3	2nd case, 4, both $A=B=1$
------------------	---------------------------------

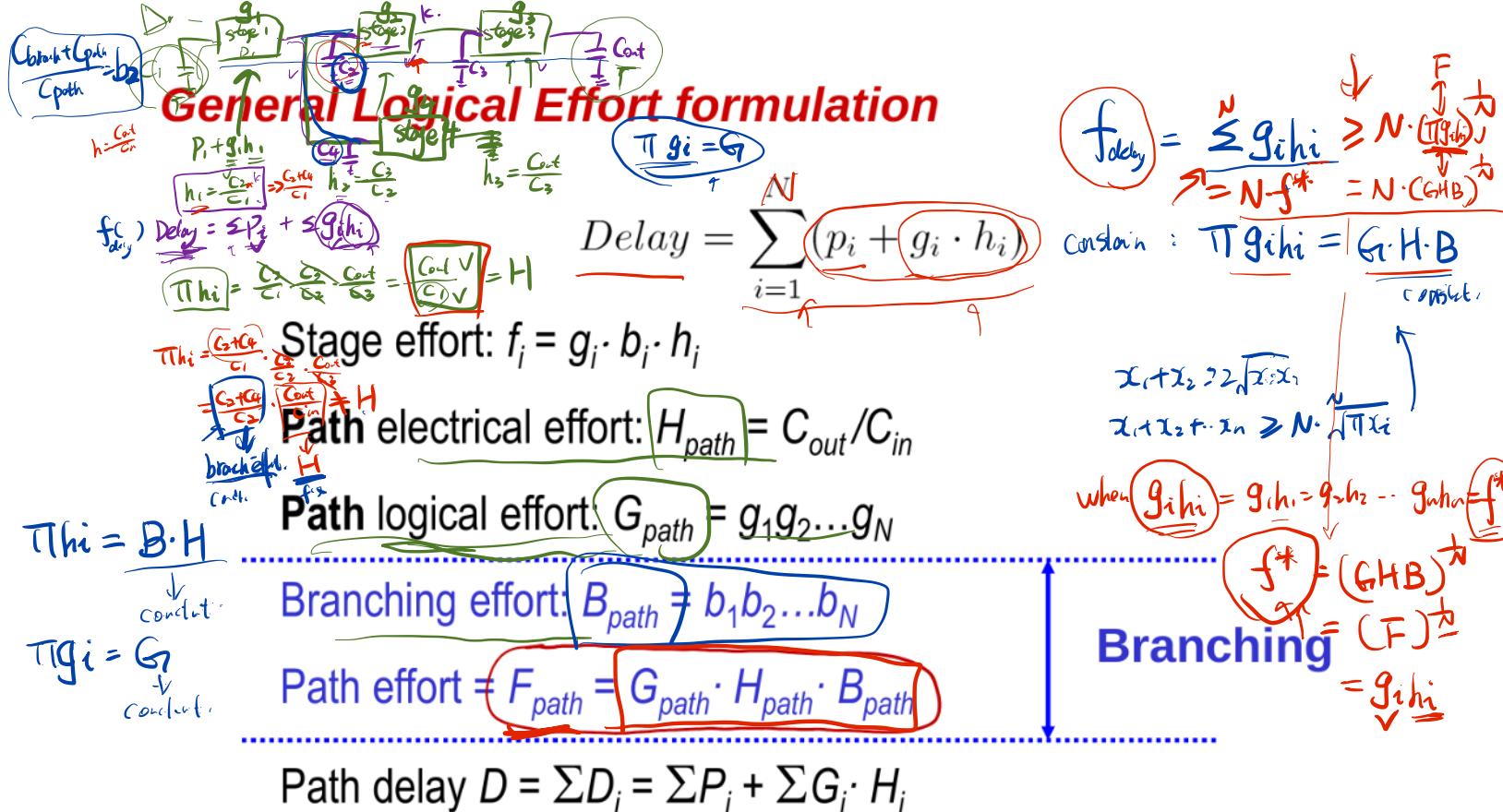
$$g_{DN} = \frac{2}{3}$$

Worst case

$$\frac{1}{2} \left(\frac{1}{2} \right) = \frac{1}{4} R_p \rightarrow \frac{1}{4} \frac{1}{4} = \frac{1}{16}$$

$$g_{DN} = \frac{C_{igate}}{C_{invref}} = \frac{6}{16+8} = \frac{6}{24} = \frac{1}{4}$$

$g_{UP} = 1$	$g_{DN} = \frac{2}{3}$
--------------	------------------------



Optimum Effort per Stage

When each stage bears the same effort:

$$f^N = \prod g \cdot h = \text{PathEffort}$$

$$f = \sqrt[N]{\text{PathEffort}}$$

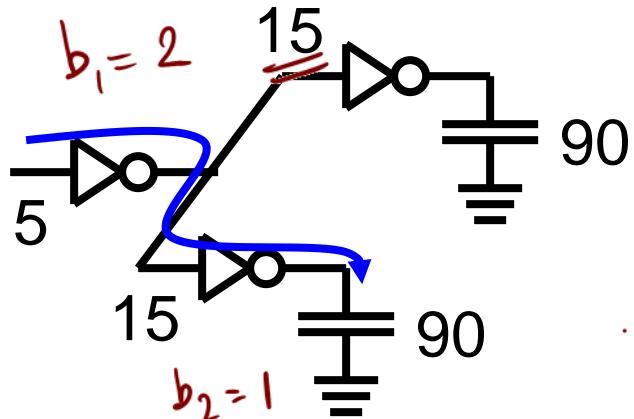
Fanout of each stage: $h_i = f/g_i$

Complex gates should drive smaller load!!!

Minimum path delay

$$\underline{D_{min}} = \sum (g_i \cdot h_i + p_i) = \underline{N \cdot f} + \underline{P}$$

Branching Example #1



$$\begin{array}{lcl} G & = & 1 \\ h & = & 90/5 = 18 \\ F & = & 18 \text{ (wrong!) } \times \\ \hline f_1 & = & (15+15)/5 = 6 \\ f_2 & = & 90/15 = 6 \\ F & = & 36, \text{ not } 18! \quad b/c \quad B=2 \end{array}$$

Introduce new kind of effort to account for branching:

- **Branching Effort:**

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

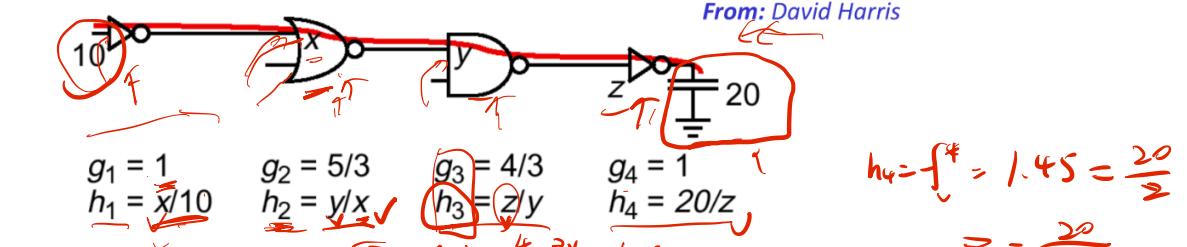
- Path Branching Effort:

$$B = \prod b_i$$

Now we can compute the path effort:

- **Path Effort:** $H = \prod g \cdot h \cdot b$

Gate Sizing Example



- First Compute Path Effort

$$\text{Path Effort } P = \prod g \cdot h = \frac{1}{10} \cdot \frac{5}{3} \cdot \frac{4}{3} \cdot 1 = \frac{20}{9}$$

- The optimal stage effort is: $N=4$

$$f^* = g \cdot h = \left(\frac{40}{9}\right)^{1/4} = 1.45$$

- We can now size the gates:

$$z = 1 \cdot \frac{20}{1.45} = 13.8$$

$$x = \frac{5}{3} \cdot \frac{y}{1.45} = 14.5$$

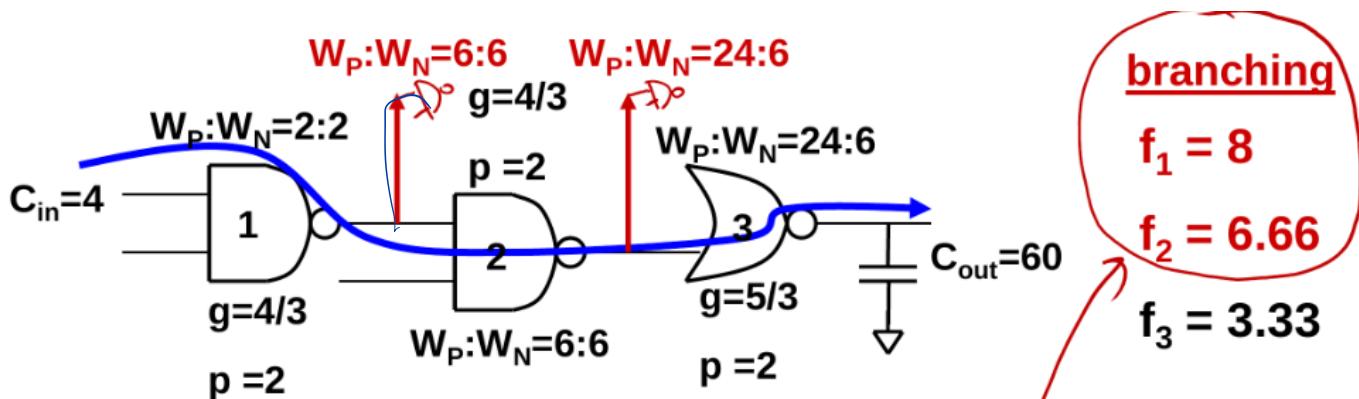
$$y = \frac{4}{3} \cdot \frac{z}{1.45} = 12.7$$

$$C_{in} = 1 \cdot \frac{x}{1.45} = 10$$

$$C_{in} = g \cdot \frac{C_{out}}{f^*}$$

- The total normalized delay is (assuming $P_{inv}=1$):

$$D = 4SE + \sum P = 4 \cdot 1.45 + (1+2+2+1) = 11.8$$



- For circuits with branching:

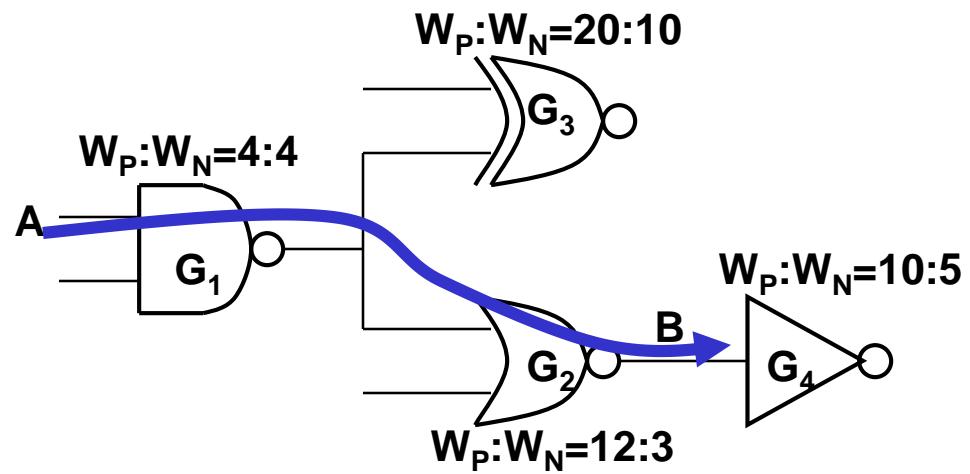
- G_{PATH} is the same = 3
- H_{PATH} is the same = 15
- F_{PATH} differs
 - $h_1 = 24/4 = 6, h_2 = 60/12 = 5, h_3 = 2$
 - $F_{PATH} = 4/3 * 6 * 4/3 * 5 * 5/3 * 2 = 177.8$

- F is no longer GH

- New F_{PATH} with branching: $F_{PATH} = GBH$

Branching Example #3: Gate Delays

- Delay of the path from A to B where $\beta = \mu = 2$ and $p_{INV}=1$
 - $g_{G1} = 4/3, p_{G1}=2, C_{IN_G1} = 8$
 - $g_{G2} = 5/3, p_{G2}=2, C_{IN_G2} = 15$
 - $g_{G3} = 4, p_{G3}=4, C_{IN_G3} = 30$
 - $C_{IN_G4} = 15$
 - $h_{G1} = (C_{IN_G2}+C_{IN_G3})/C_{IN_G1} = 5.625, h_{G2} = C_{IN_G4}/C_{IN_G2} = 1$
 - $d_{G1} = g_{G1}h_{G1}+p_{G1} = 9.5$
 - $d_{G2} = g_{G2}h_{G2}+p_{G2} = 3.66$
 - Delay = 13.16
 - Normalized



Branching Example #4

Select gate sizes y and z to minimize delay from A to B

Logical Effort:

$$G = (4/3)^3$$

Electrical Effort:

$$H = C_{\text{out}} / C_{\text{in}} = 9$$

Branching Effort:

$$B = 2 \cdot 3 = 6$$

Path Effort:

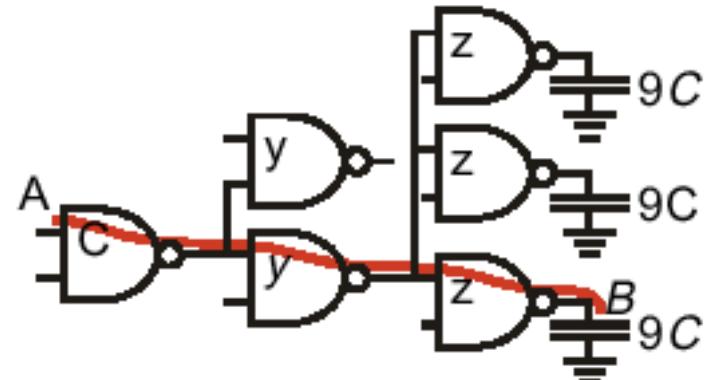
$$PE = G \cdot H \cdot B = 128$$

Best Stage Effort:

$$f = PE^{1/3} \approx 5$$

Delay:

$$D = 3 \cdot 5 + 3 \cdot 2 = 21$$



Work backward for sizes:

$$z = \frac{9C \cdot (4/3)}{5} = 2.4C$$

$$y = \frac{3z \cdot (4/3)}{5} = 1.9C$$