

EECS 151/251A SP2022 Discussion 7

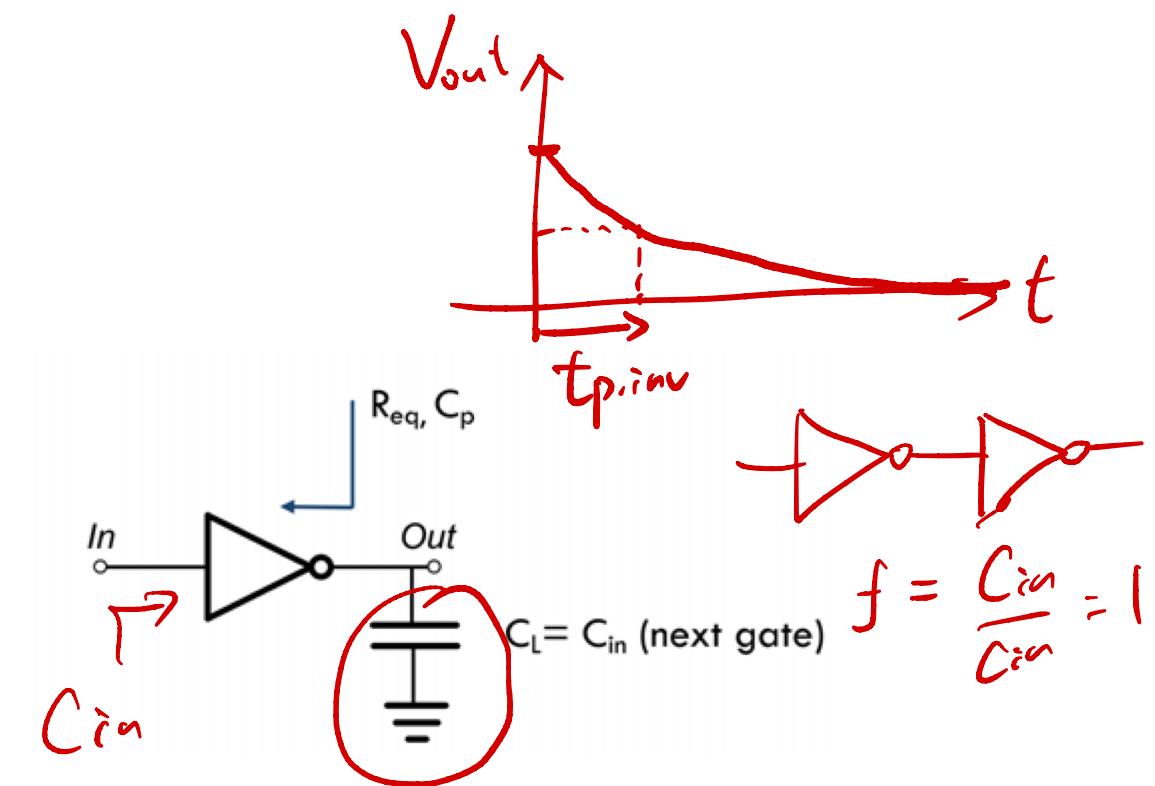
GSI: Yikuan Chen, Dima Nikiforov

Agenda

- Inverter Delay – continue from last week's
- Logical Effort
- Elmore Delay

Loaded Inverter Delay

$$\begin{aligned}
 t_{p,inv} &= \ln 2 \cdot R_{eq} (C_{p,tot} + C_L) \\
 &= \ln 2 \cdot R_{eq} C_{p,tot} \left(1 + \frac{C_L}{C_{p,tot}} \right) \\
 &= \ln 2 \cdot R_{eq} C_{p,tot} \left(1 + \frac{C_L}{\gamma C_{in}} \right) \\
 &= \ln 2 \cdot R_{eq} C_{p,tot} \left(1 + \frac{f}{\gamma} \right) \\
 &= \ln 2 \cdot \underline{\tau_{inv}} \left(1 + \frac{f}{\gamma} \right)
 \end{aligned}$$

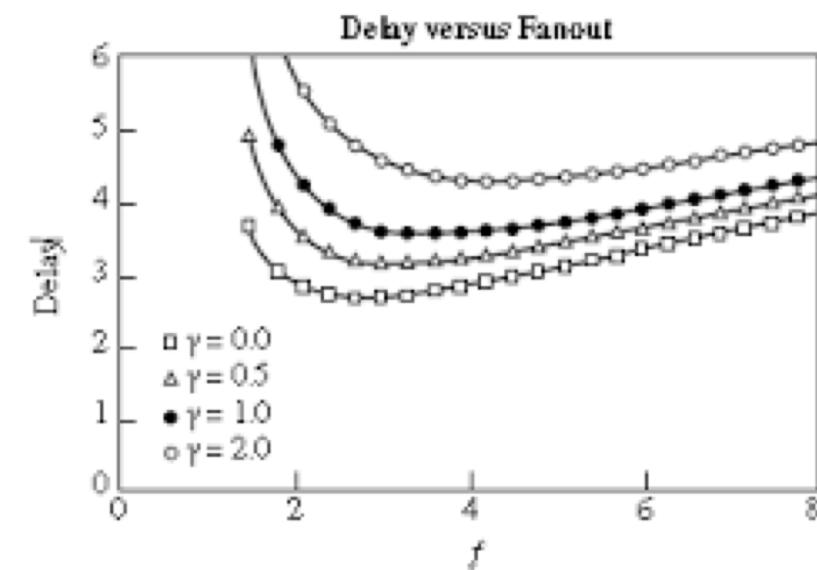
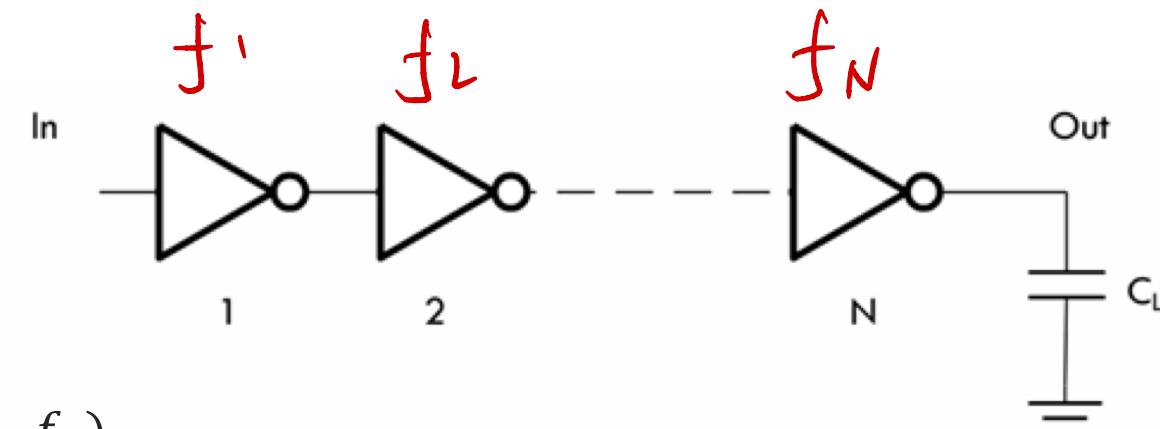


- Intrinsic vs. Extrinsic delay
 - Intrinsic τ_{inv} independent of sizing
- Fanout $\underline{f = C_L/C_{in}}$
- Generalizable to any CMOS gate

t : time it take for output to halve.

Inverter Chain Sizing

- Goal: minimize path delay
 - Assume 1st inverter has unit size ($C_{in,1} = 1$)
- Path delay:
$$D = t_{p1} + \dots + t_{pN} = (1 + f_1) + \dots + (1 + f_N)$$
- Path fanout $F = \frac{C_L}{C_{in}}$
- Solution
 - Take partial derivatives w.r.t C_2, \dots, C_N
 - Get $f_1 = f_2 = \dots = f_N$
 - Min. path delay = $\underline{Nf} + N = \underline{\underline{N\sqrt{F}}} + \underline{\underline{N}}$
 - Size backwards

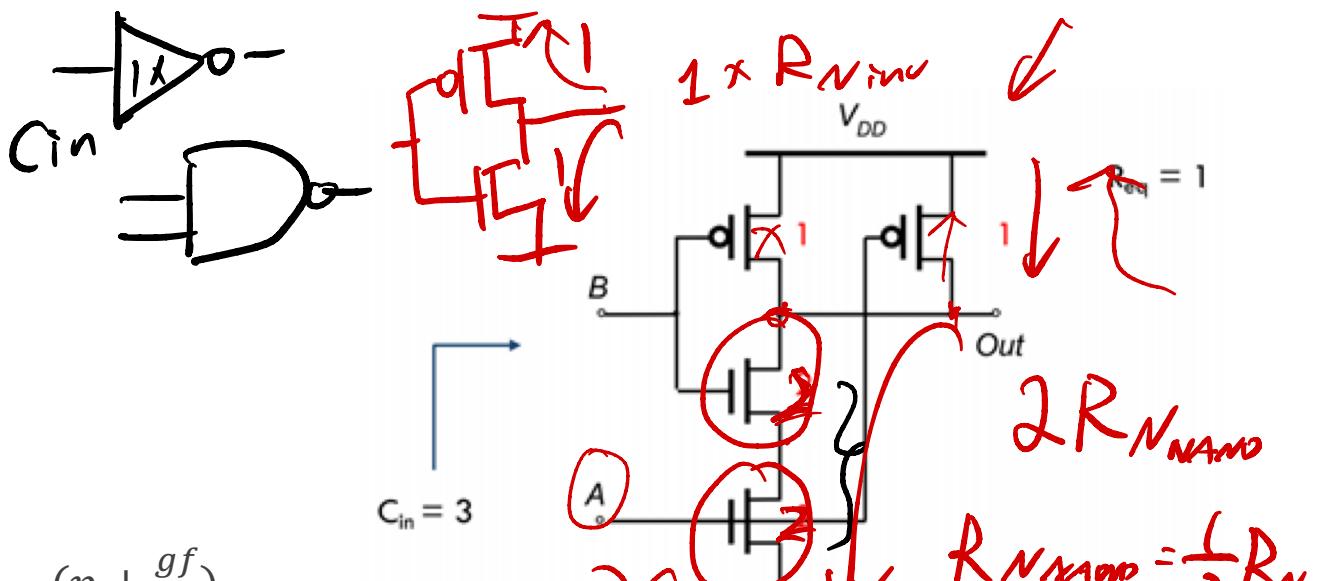


Logical Effort



Logical Effort, Parasitic Delay

- For any gate, $D(\text{gate}) = \text{LE} \cdot \text{FO} + P$



- Logical Effort

- LE , Also defined as g in $t_{p,gate} = \tau_{\text{inv}}(p + \frac{gf}{\gamma})$

- $\text{LE} \equiv \frac{R_{\text{eq},\text{gate}} C_{\text{in},\text{gate}}}{R_{\text{eq},\text{equiv}} C_{\text{in},\text{inv}}}$, i.e. ratio of $C_{\text{in},\text{gate}}$ to $C_{\text{in},\text{inv}}$ of a unit inverter with the same output current

- Key to calculating LE: size the transistors to have the same R_{eq} as a unit inverter

- Parasitic Delay

- Also defined as p in $t_{p,gate} = \tau_{\text{inv}}(p + \frac{gf}{\gamma})$

- Only difference from LE is that p looks at the ratio of $C_{\text{out},\text{gate}}$ to $C_{\text{out},\text{inv}}$ (intrinsic capacitance)

$$\text{LE}_{\text{NAND}} = \frac{1 \cdot (1+2)}{1 \cdot (1+1)} = \frac{3}{2}$$

\uparrow
 inv, min

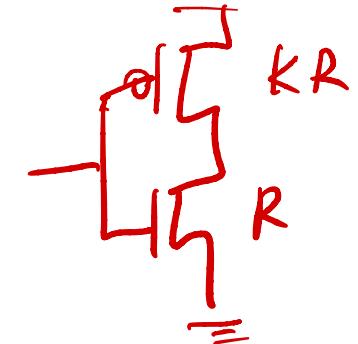
$$P \equiv \frac{C_{\text{out},\text{gate}}}{C_{\text{out},\text{inv}}}$$

$$[P_{\text{NAND}} = \frac{(1+1)+2}{1+1} = 2]$$

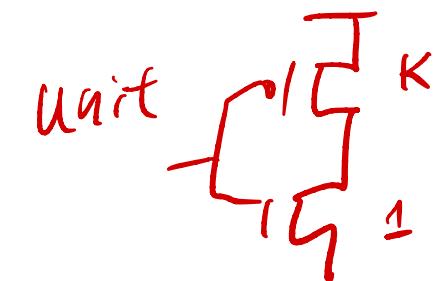
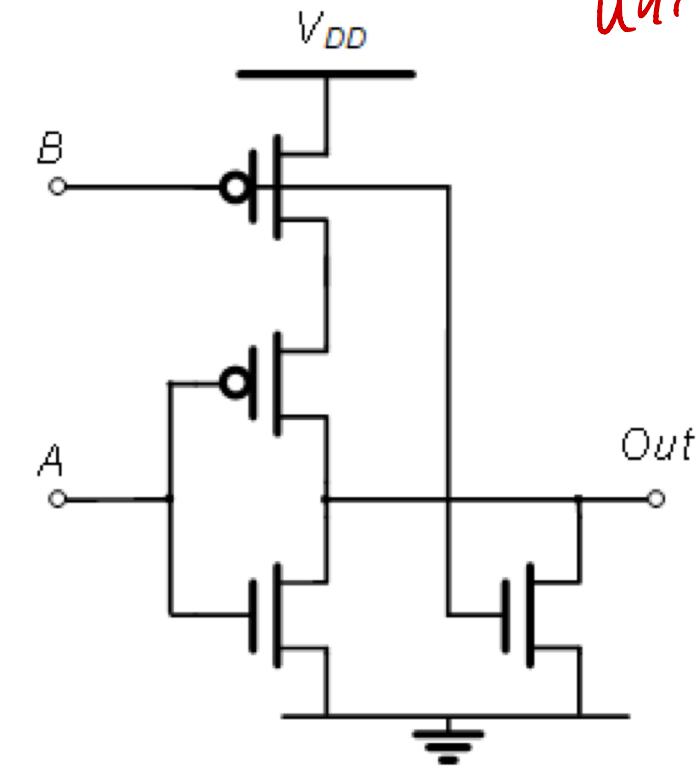
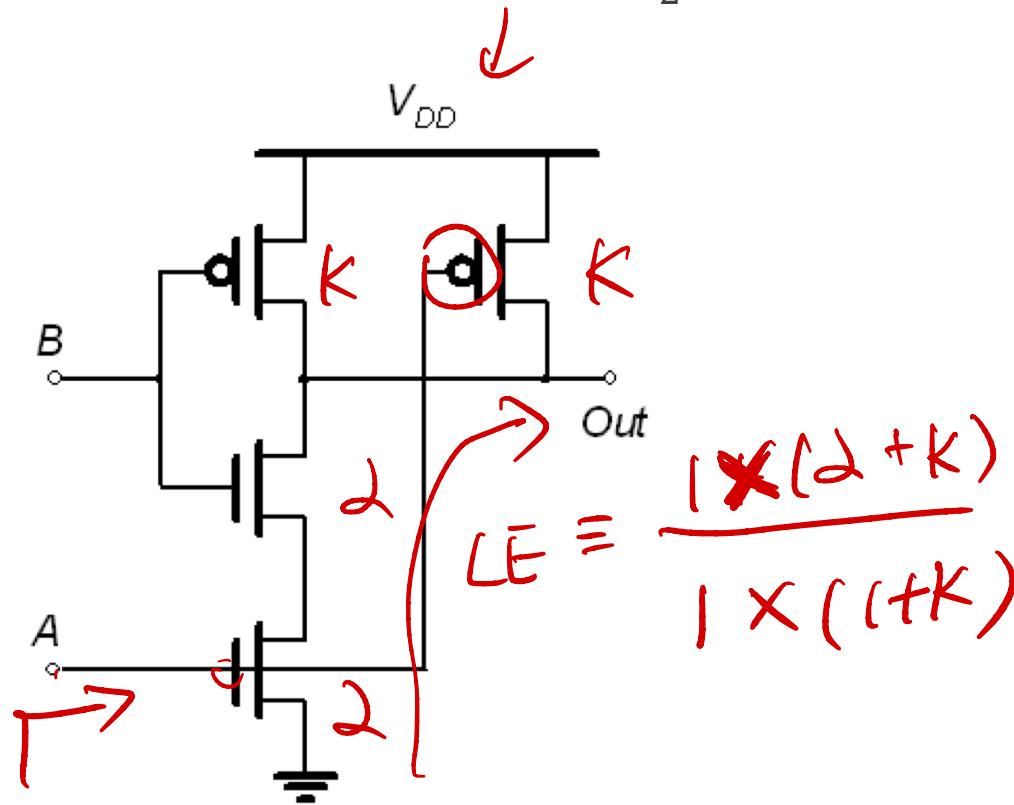
PMOS/NMOS Resistance

$$I = I_{Nmos} : Pmos$$

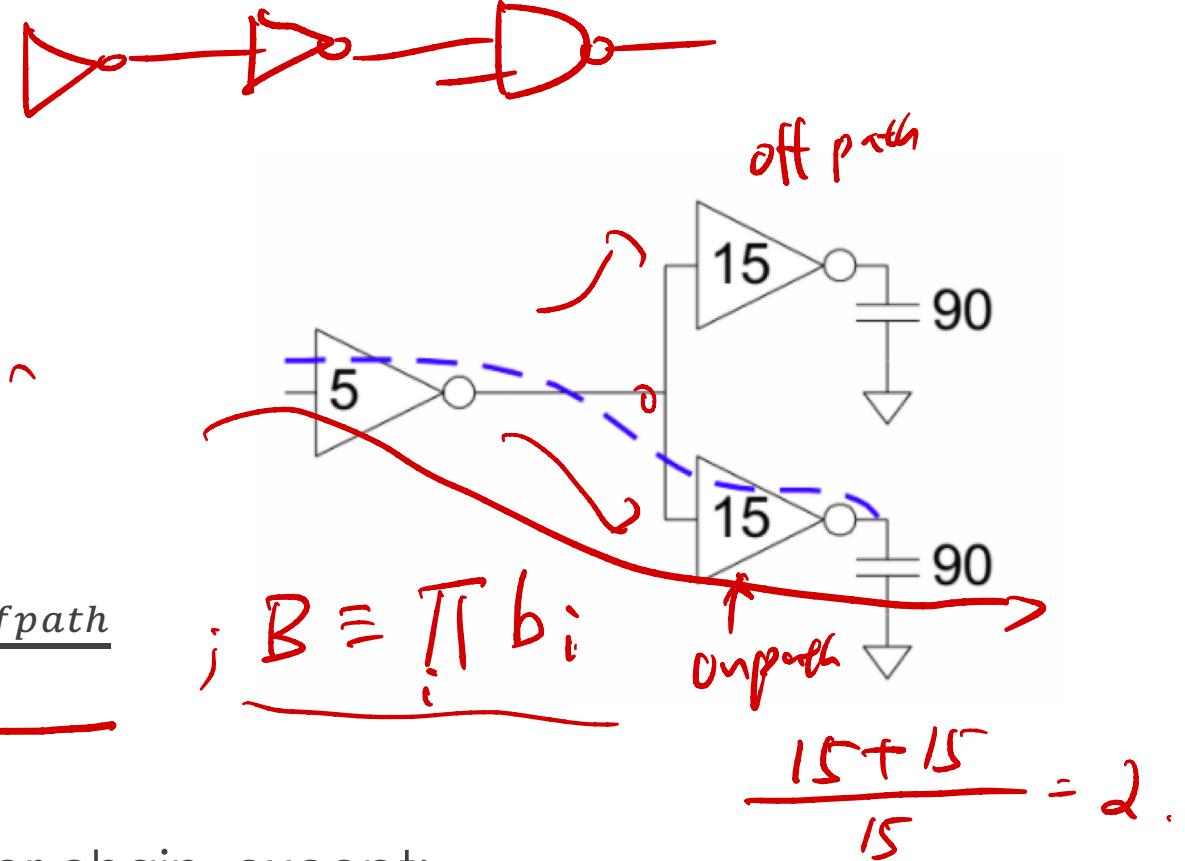
$$r = K$$



- In lecture, $LE_{NAND2} = LE_{NOR2} = \frac{3}{2}$. What if $R_{on,p} = K \cdot R_{on,n}$?



Minimizing Path Delay

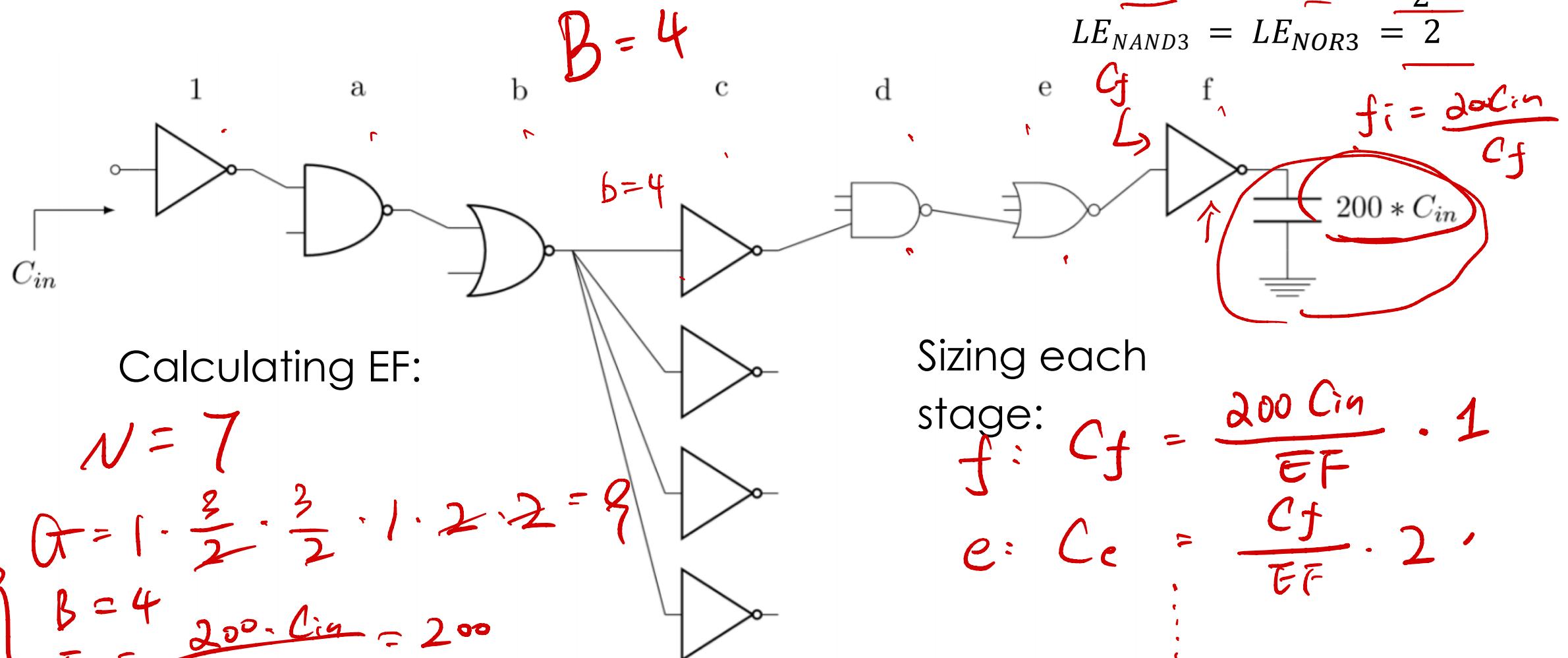


- Path Logical Effort: $G = g_1 \cdot g_2 \cdot \dots \cdot g_N$
- Path fanout $F = C_L / C_{in}$
- Branching, b
 - Extra loading factor $b = \frac{C_{onpath} + C_{offpath}}{C_{onpath}}$
 - Add b to total path effort $H = GFB$
- Solution is similar to single-branch inverter chain, except:
 - Calculate an “effective fanout” for each stage: $EF = \sqrt[N]{H}$
 - Size each stage such that $EF = g_i \cdot f_i$. You can control f_i by sizing!
 - Min. path delay = $N * \sqrt[N]{H} + \sum p_i$

$\sqrt[N]{H}$
to size

$$EF = g_i \cdot f_i \quad g_i \in LE_i$$

Logic Path with Branch - Example



Sizing each stage:

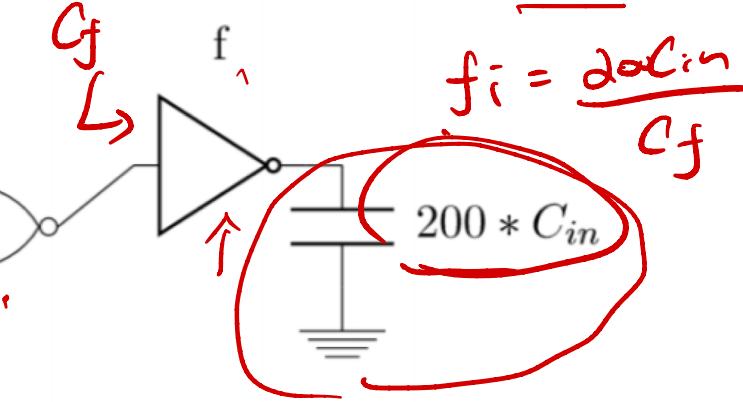
$$f: C_f = \frac{200 C_{in}}{EF} \cdot 1$$

$$e: C_e = \frac{C_f}{EF} \cdot 2 \cdot \dots$$

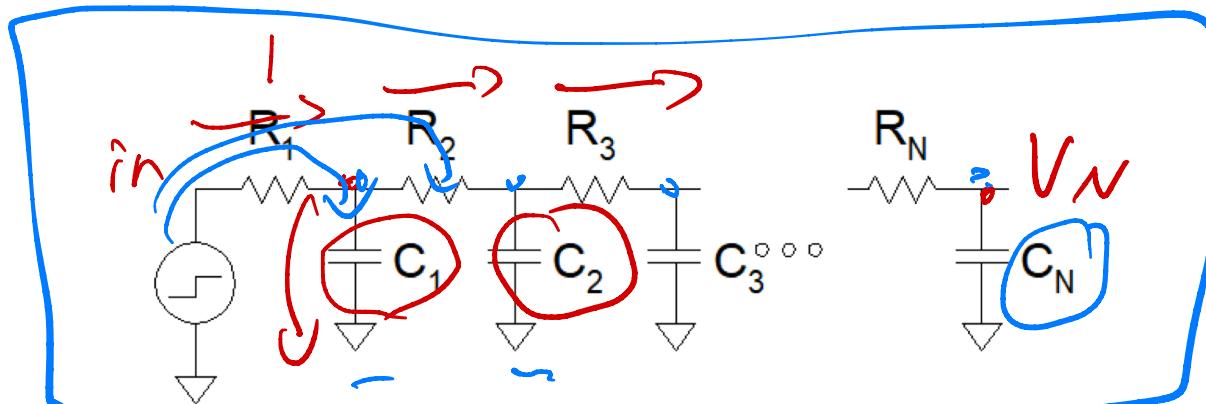
a:

$$\underline{LE_{NAND2}} = \underline{LE_{NOR2}} = \frac{3}{2}$$

$$\underline{LE_{NAND3}} = \underline{LE_{NOR3}} = \frac{1}{2}$$



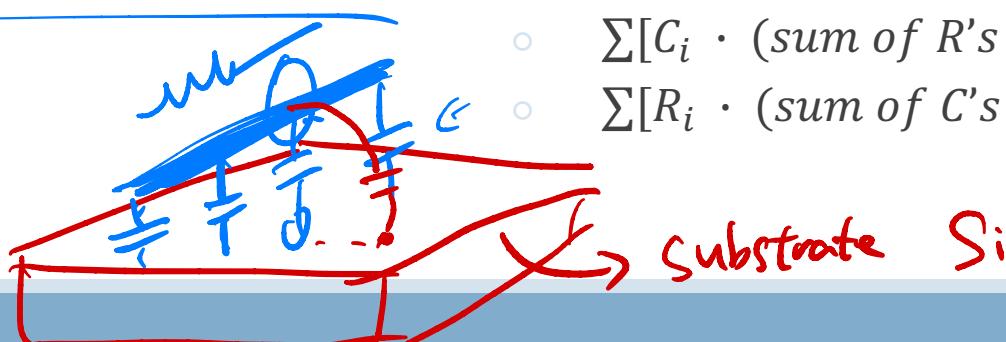
Elmore Delay – “All the R’s for that C” or “All the C’s by that R”



$$t_{pd} \approx \sum_{\text{nodes } i} R_{i-\text{to-source}} C_i$$

$$= R_1 C_1 + (R_1 + R_2) C_2 + \dots + (R_1 + R_2 + \dots + R_N) C_N$$

$$R_1 (C_1 + C_2 + \dots + C_N) \\ + R_2 (C_2 + \dots + C_N) \\ + \dots$$



Approximately the dominant RC time constant of a network

R's can be:

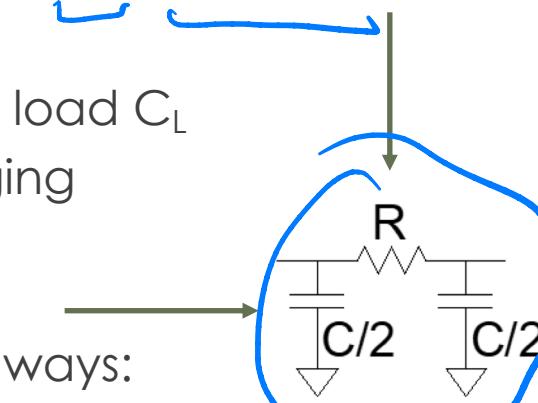
- Gate equivalent resistance R_{eq}
- Wire resistance $R = R_{\square} \cdot \# \text{ squares}$



C's can be:

- Gate C_{in} and C_p , load C_L
- Wire plate & fringing capacitance

■ $\pi - model$



Think of it 2 equivalent ways:

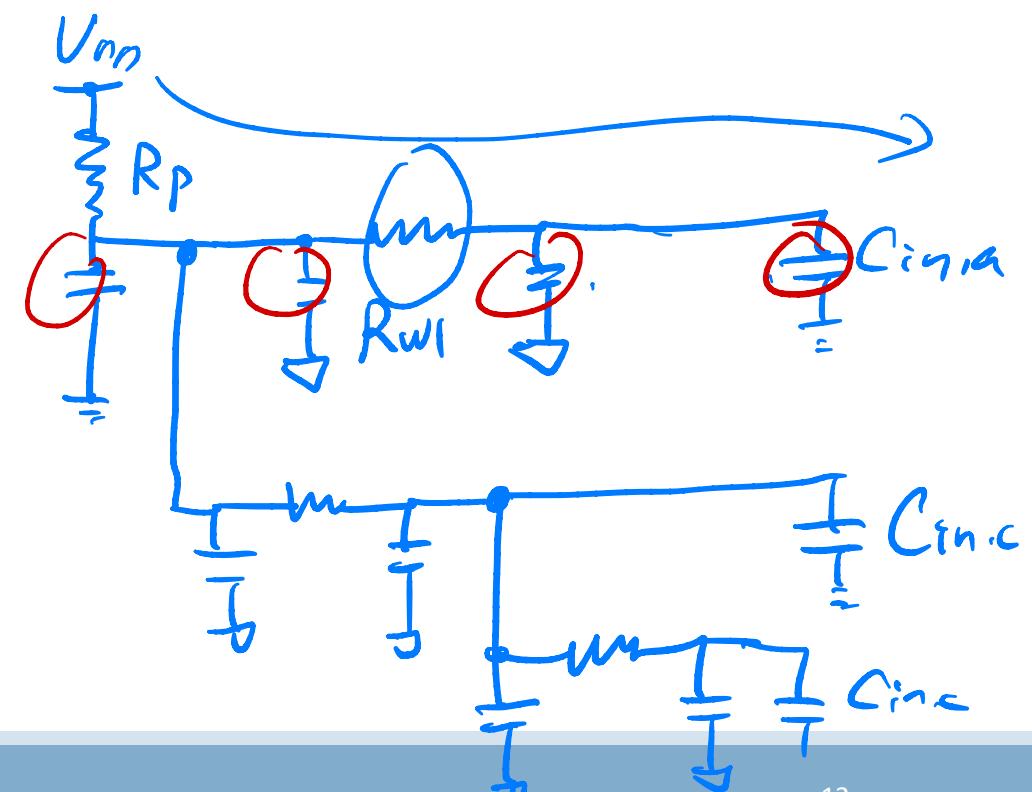
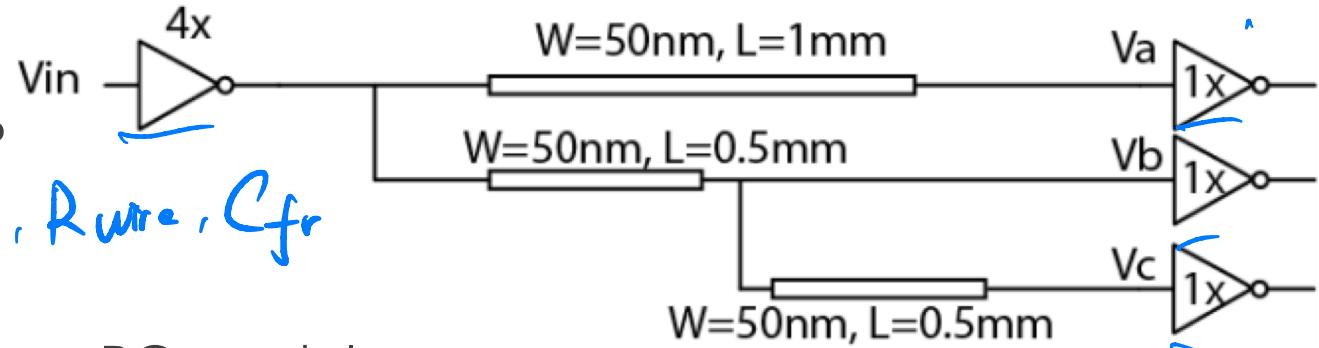
- $\sum [C_i \cdot (\text{sum of } R's \text{ charging } C_i)]$
- $\sum [R_i \cdot (\text{sum of } C's \text{ that } R_i \text{ charges})]$

Elmore Example

- What parameters do we need?
 $C_{inu}, C_p, R_{onp}, R_{onn}, C_{pp}, R_{wire}, C_{fr}$
- Draw the equivalent circuit Elmore RC model
- Calculate delay to V_a

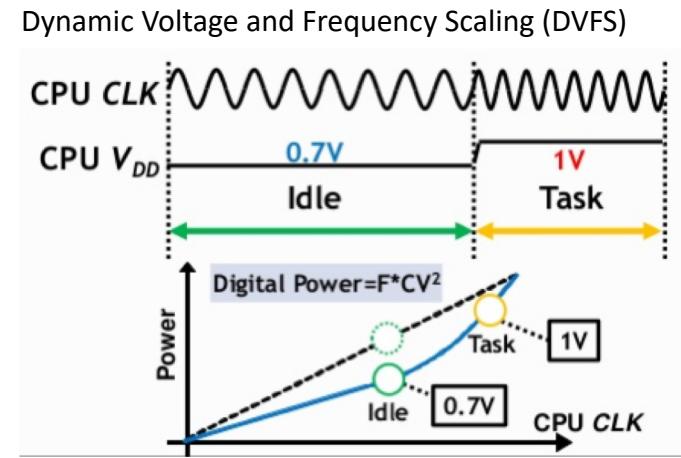
$$t_p = \ln 2 \cdot (R_p (\sum C_{\text{charged by } R_p}))$$

$$+ \ln 2 \cdot (R_{wi} \cdot \sum C_{\text{charged by } R_{wi}})$$



Power/Energy in Digital Circuits

- Fundamentally, charging/discharging capacitors (gate, parasitic, load) through resistances (PMOS, NMOS, wires)
 - Capacitors draw CV^2 joules from supply over 1 charge/discharge cycle
 - $\frac{1}{2}CV^2$ dissipated in PMOS as heat when charging
 - $\frac{1}{2}CV^2$ stored on capacitor, then dissipated in NMOS when discharging
- Dynamic power = $P_{switching} = \alpha CV^2 f$
 - How to minimize each term?
 - Minimizing which terms reduces total energy consumed?
- Static power = leakage → wasted energy!



Energy Example

- Initially: A = 1, C = 1, Out = 0
- Energy pulled from supply when B = 1 \rightarrow 0?
- Then, how much energy dissipated when C = 1 \rightarrow 0?

