# EECS 151/251A Discussion 8

Daniel Grubb 10/19, 10/20, 10/25

# Administrivia

- Lab 6 due Friday 10/22
  - Cutoff for checkoffs and submissions is the end of next week's lab sessions
- Project starts this week
  - Do your best to finish up labs and don't procrastinate!
- Homework 6 due Friday 10/22
- Midterm 1 grades release
  - Regrades due Friday 10/22
- Midterm 2 on the horizon (tentatively November 4th)
- What are you having trouble with?  Feedback?

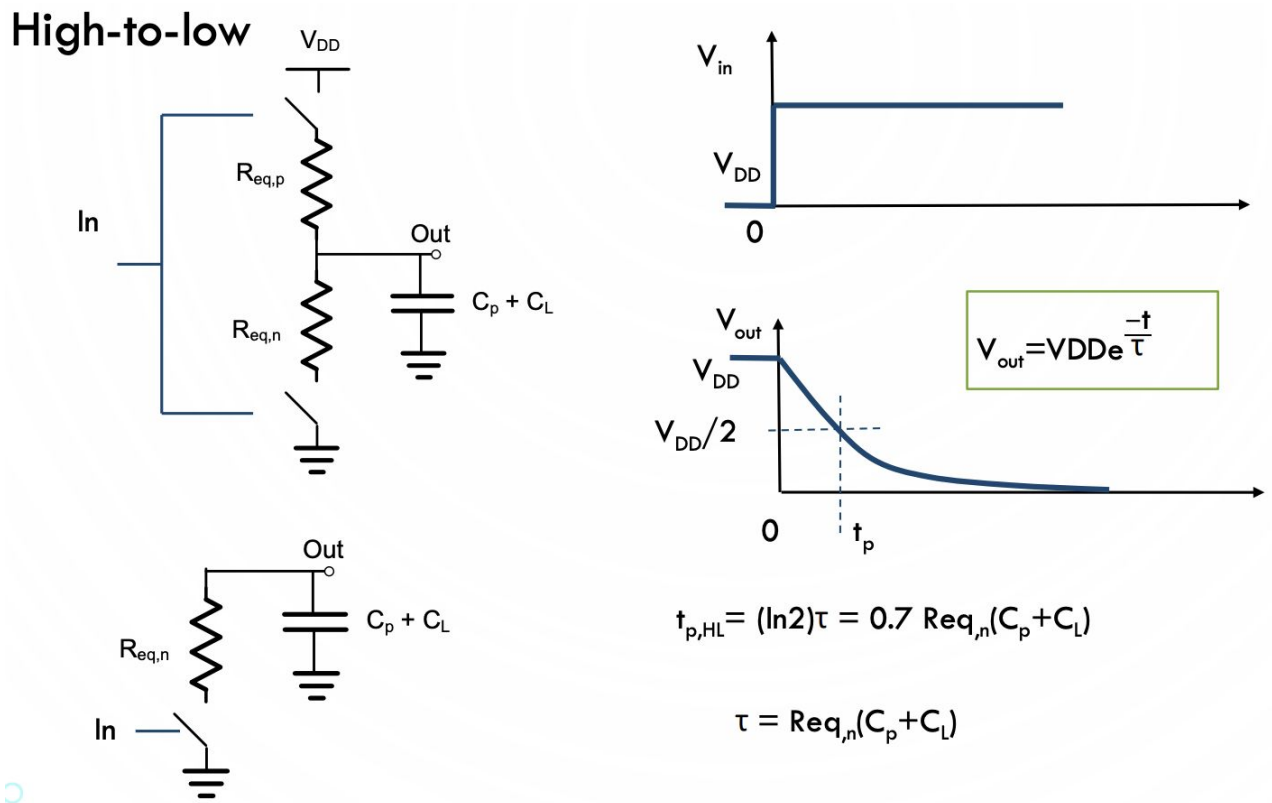
# Agenda

- Topics
  - Inverter delay and sizing
  - Logical effort
  - Path effort
  - Elmore delay
- How familiar/comfortable are you with:
  - transistor switch model?
  - inverter delay?
  - inverter delay optimization?
  - logical effort?
  - parasitic delay?
  - path effort/optimization?

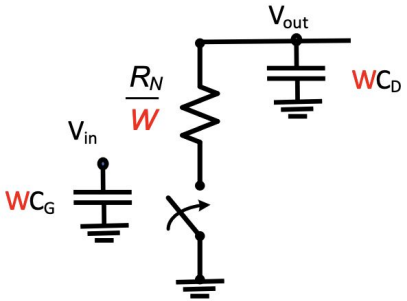
# A Note on Notation and Values

- PMOS / NMOS ratios
  - Old technologies/textbooks give 2:1, this class generally uses 1:1
- Ratio of Cd to Cg = $\gamma$ (gamma) = 1
  - Set by process technology
  - ~1.2 in recent processes, but still keep it 1
- Logical effort
  - **LE** or **g** both used
- Fanout or electrical effort
  - **F** for total fanout
  - **f** for stage fanout
  - **h** also used

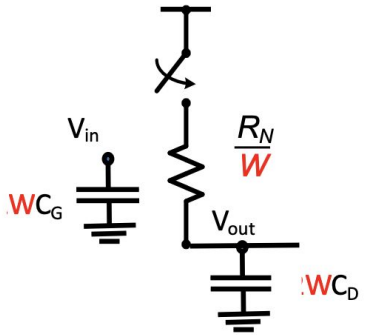
# Inverter RC Delay Model



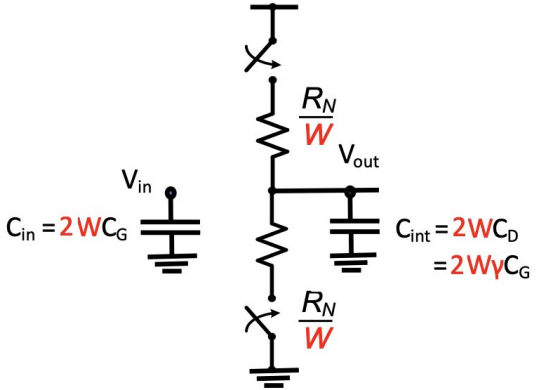
# Inverter Delay




NMOS with size W




PMOS with same R as NMOS



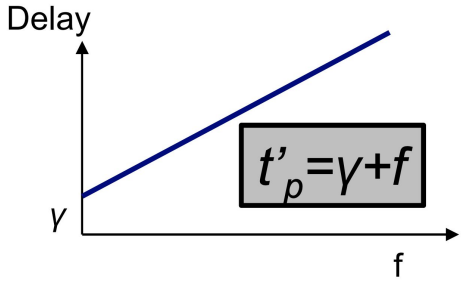

$$t_p = 0.69\left(\frac{R_N}{W}\right)(3W\gamma C_G) = 0.69(3\gamma)R_N C_G$$

Intrinsic inverter delay

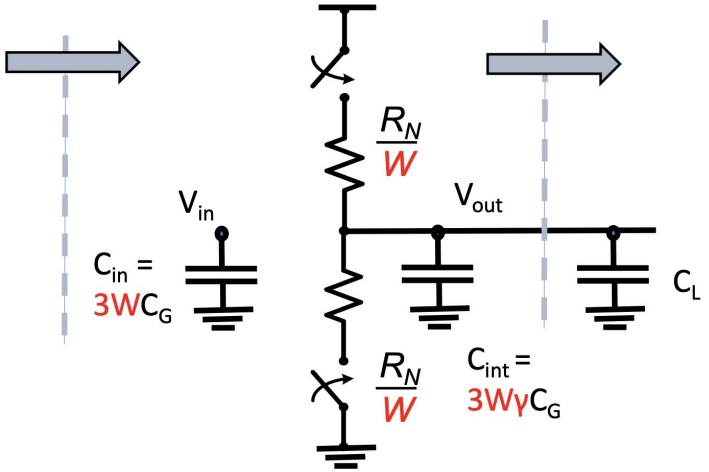
What if we make the inverter bigger?

# Inverter Delay

- Add a load capacitance now
- Two components
  - Intrinsic delay
  - Fanout, f = $C_L/C_{in}$
- FO4 delay: how many unit delays?
  - Why is this a useful quantity?



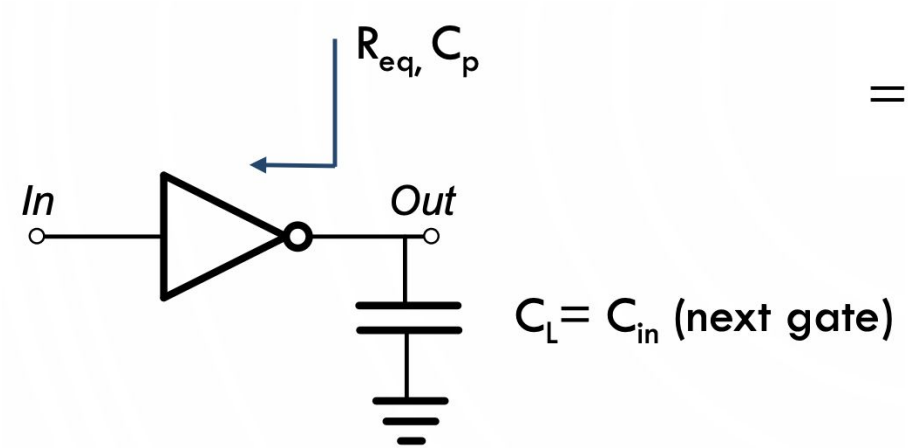

$$
\begin{aligned}
t_p &= 0.69\left(\frac{R_N}{W}\right)(C_{\text{int}} + C_L) \\
&= 0.69\left(\frac{R_N}{W}\right)(3W\gamma C_G + C_L) \\
&= 0.69(3C_G R_N)(\gamma + \frac{C_L}{C_{in}}) \\
&= t_{inv}(\gamma + \frac{C_L}{C_{in}}) = t_0(\gamma + f)
\end{aligned}
$$



# Inverter Delay Scratch Area

# Inverter Sizing Thought Experiment

- Say we just have 1 inverter driving a load capacitance
  - We want to minimize delay
  - We can size it anyway we want
  - How should we size it?
- Be careful with this solution!



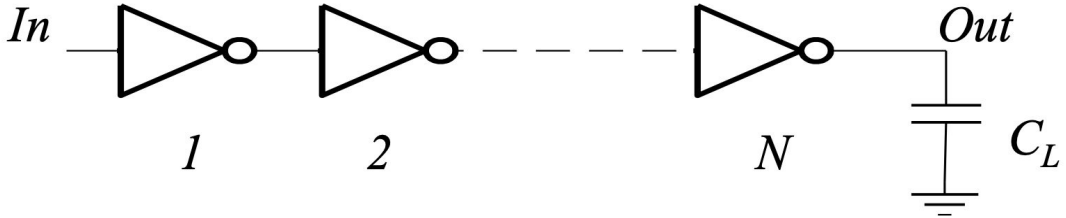

$$= t_{inv}(\gamma + \frac{C_L}{C_{in}}) = t_0(\gamma + f)$$

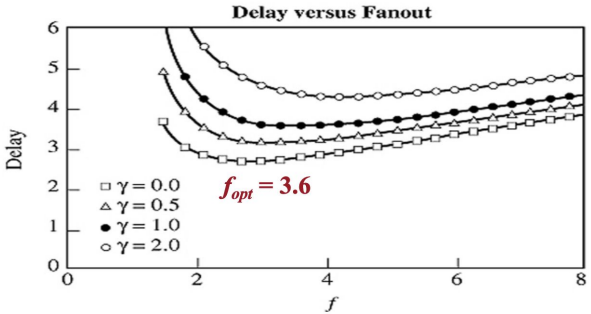
# Inverter Chain Sizing

$$\frac{C_{in,j}}{C_{in,j-1}} = \frac{C_{in,j+1}}{C_{in,j}}$$

- Goal: minimize the delay of the path
  - Constrain the size of the first inverter + load capacitance in each case
  - What's a first pass solution?
- (1) Size a chain given N inverters
  - Key result: size stages to have identical fanout (minimize the total chain delay expression)
    - What does this say about each stage's delay?
  - Find f: f^N=F=$C_L/C_{in,1}$
  - Start from beginning or end of chain
- (2) Determine optimal N and size the chain
  - Assume fanout, f=4 (from graph)
  - Find number of stages, N (integer): ?
  - Same steps as (1)

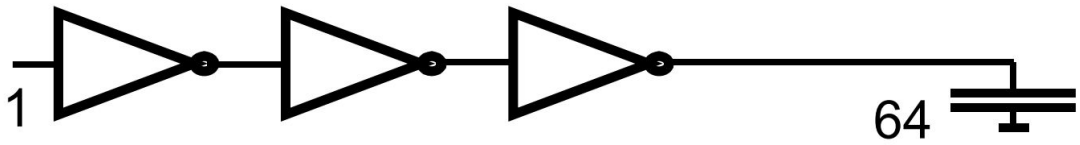
$$t_p = Nt_{inv}\left(\gamma + \sqrt[N]{F}\right),\ F = C_L/C_{in}$$



# Inverter Chain Sizing Examples

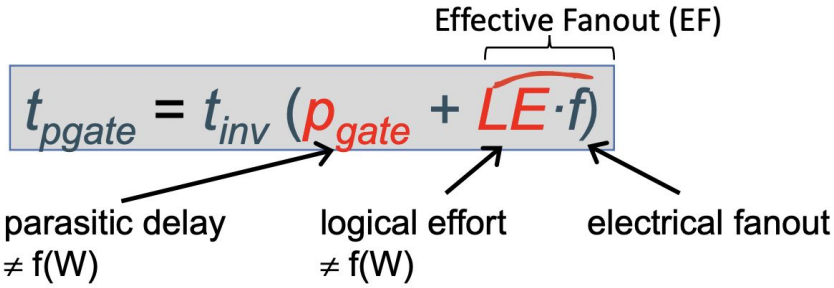
- Chain of 3 with load cap of 64:
  - What if we add a capacitance in the middle?

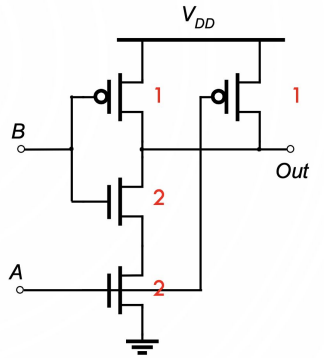



- Load cap. of 1024, give N and sizings:
  - What if non-integer?

# Logical Effort and Parasitic Delay

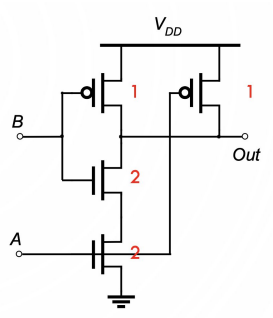
- How much worse is a gate at driving a load capacitance than an inverter with the same input capacitance? (other ways to look at it)
  - Looks like an inverter, but the fanout seems larger
  - Easy way: size gate to deliver same as inverter current, take cap. Ratio
- LE = $(R_{eq,gate}C_{in,gate})/(R_{eq,inv}C_{in,inv})$
- Parasitic delay: delay of gate driving no load (set by internal cap.)
  - Find gate RC delay, then extract $t_{inv}$ term; ratio of internal cap to inverter

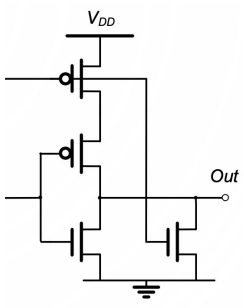


# Logical Effort/Parasitic Delay Examples

- Also, consider if the PMOS/NMOS resistances are different than nominal?



# Path Delay

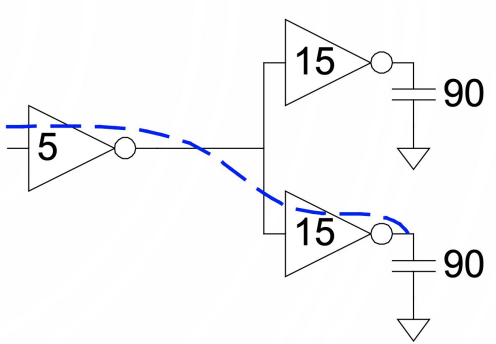
- How do we minimize delay more generally?
  - Delay = $t_{inv}\Sigma(p_i + LE_i \cdot f_i)$
- Path fanout: $F=C_L/C_{in}$
- Path LE: $G=g_1g_2...g_N$
- Branching effort
- Use similar result to inverter chain
  - Best EF is still around 4
- Design process:
  - Calculate Path Effort: PE=GFB
  - Estimate best number of stages: $N=\log_4 F$
  - Calculate Effective Fanout per stage: EF=PE^(1/N)
  - Size the gates from beginning or end of chain
- Can always add inverters to the end of the chain

$$b = \frac{C_{onpath} + C_{offpath}}{C_{onpath}}$$

$$G = \prod g_i$$

$$F = \prod f_i = \prod g_i h_i$$

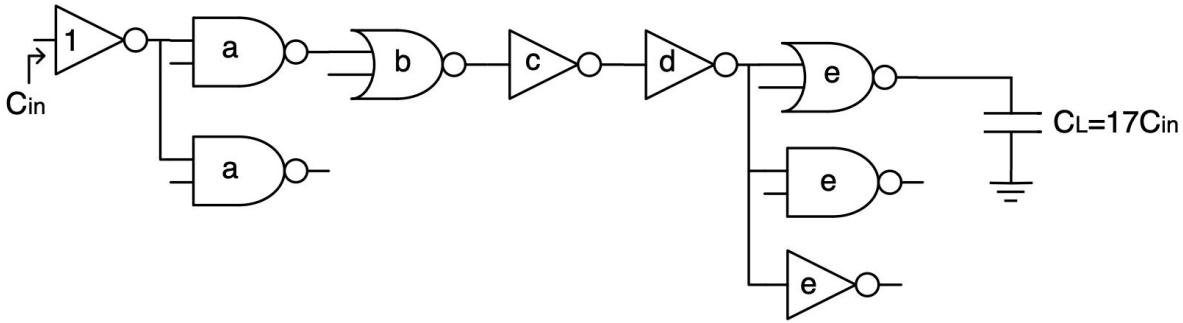
$$B = \prod b_i$$



# Path Delay Scratch Area

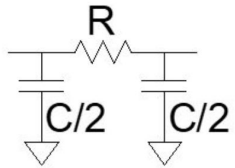
# Path Delay Example

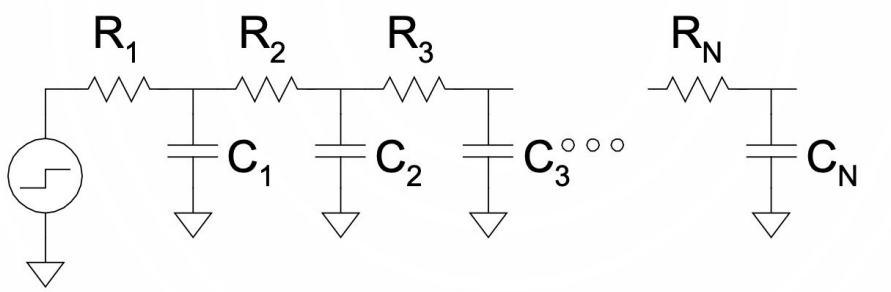
- Remember, $EF_i = LE_i \cdot f_i = LE_i \cdot C_{L,i}/C_{in,i}$



# Elmore Delay

- Approximate RC time constant of a tree network
- R's and C's include both transistors/parasistics and wires
  - Wire π model (where does the R and C come from?)
- Can think of it as:
  - ∑[$C_i$ x (sum or R's charging $C_i$)]
  - ∑[$R_i$ x (sum of C's that $R_i$ charges)]




$$t_{pd} \approx \sum_{\text{nodes } i} R_{i-to-source} C_i$$
$$= R_1 C_1 + (R_1 + R_2) C_2 + \ldots + (R_1 + R_2 + \ldots + R_N) C_N$$

# Elmore Delay Examples

- What information do we need?
- Draw the equivalent model
- Calculate delay to outputs