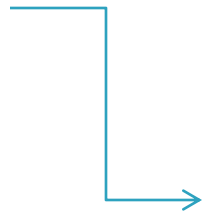
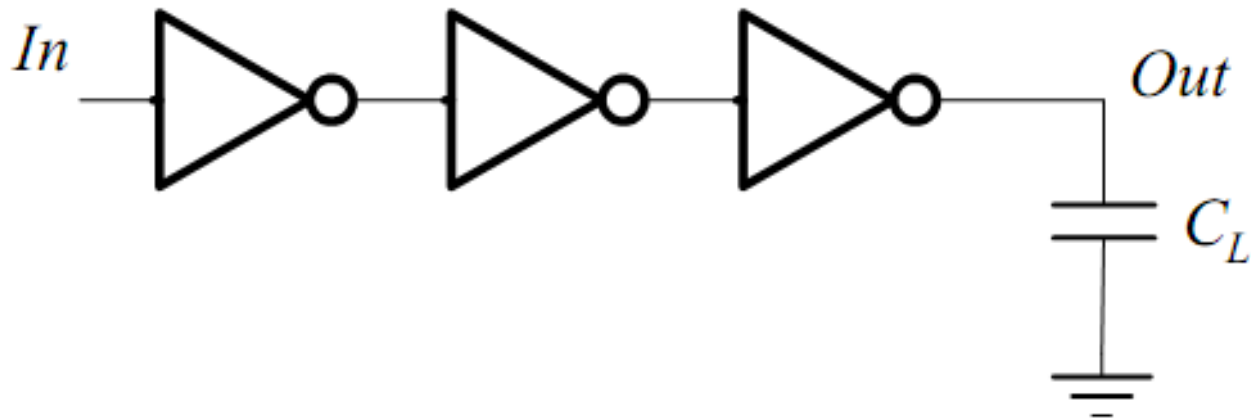


Inverter Delay Optimization



The Next Question: Inverter Chain

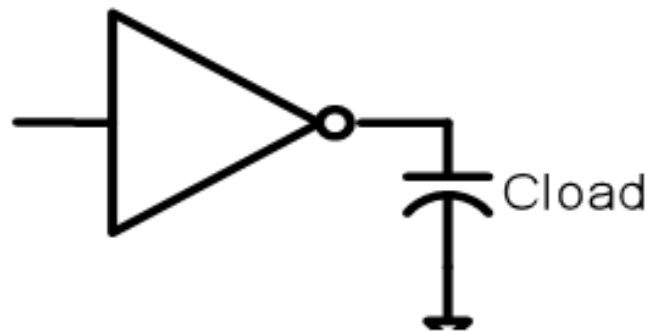


- For some given C_L :
 - How many stages are needed to minimize delay?
 - How to size the inverters?

- Anyone want to guess the solution?

Careful about Optimization Problems

- Get fastest delay if build one **very** big inverter
 - So big that delay is set only by self-loading



- Likely not the problem you're interested in
 - Someone has to drive this inverter...

Engineering Optimization Problems in General

- Need to have a set of constraints
- Constraints key to:
 - Making the result useful
 - Making the problem have a 'clean' solution
- For sizing problem:
 - Need to constrain size of first inverter

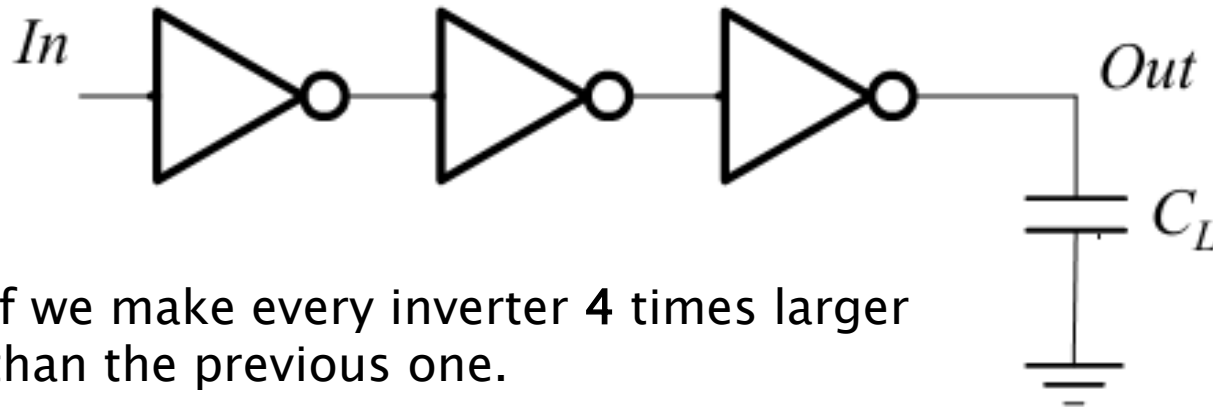
Delay Optimization Problem #1

- You are given:
 - A fixed number of inverters
 - The size of the first inverter
 - The size of the load that needs to be driven

- Your goal:
 - Minimize the delay of the inverter chain

- Need model for inverter delay vs. size

Inverter Chain



If we make every inverter 4 times larger than the previous one.

- For some given C_L :
 - How many stages are needed to minimize delay?
 - How to size the inverters?

- Anyone want to guess the solution?

Inverter Delay

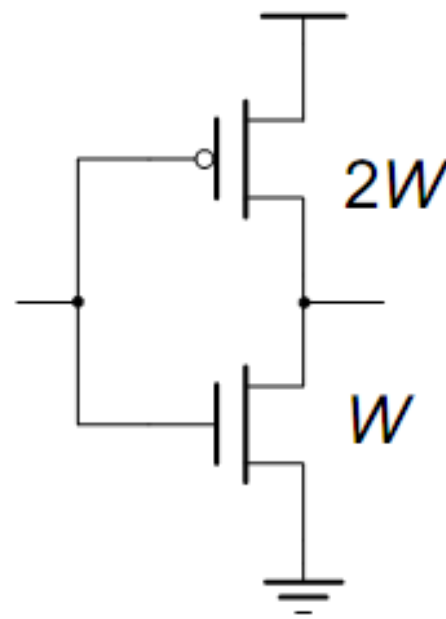
- Minimum length devices, $L = 0.09\mu\text{m}$
- Assume that for $W_P = 2W_N = 2W$
 - approximately equal resistances, $R_N = R_P$
 - approx. equal rise and fall delays, $t_{pHL} = t_{pLH}$

- Analyze as an RC network:

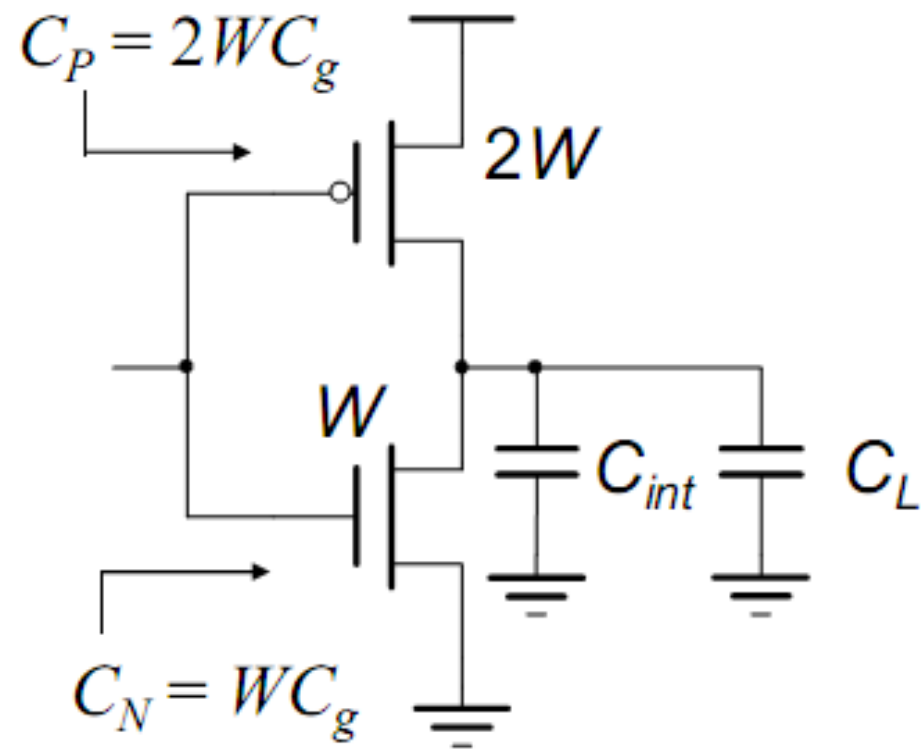
$$R_P = R_{sq,p} \left(\frac{L}{W_P} \right) \approx R_N = R_{sq,n} \left(\frac{L}{W_N} \right) = R_W$$

$$\text{Delay: } t_{pHL} = (\ln 2) R_N C_{tot} = t_{pLH} = (\ln 2) R_p C_{tot}$$

$$\text{Loading on the previous stage: } C_{in} = 3WC_g$$



Inverter Delay



$$R_W = R_{sq,n} \left(\frac{L}{W} \right)$$

$$C_{int} = 3WC_d$$

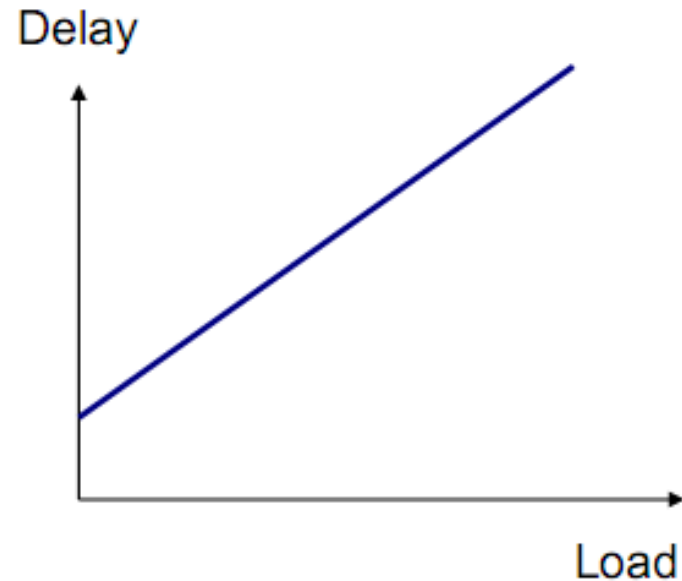
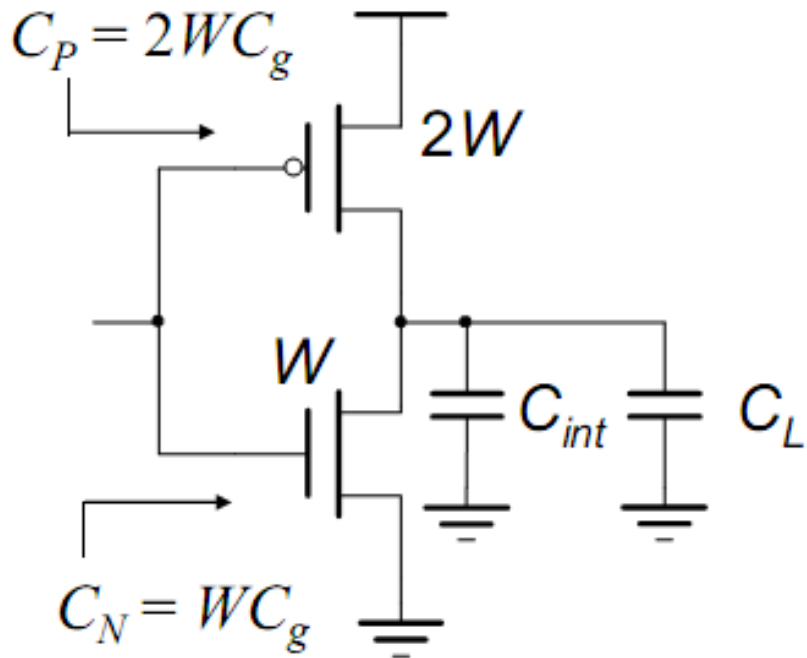
$$C_{in} = 3WC_g$$

Replace $\ln(2)$ with k (a constant):

$$\text{Delay} = kR_W C_{int} + kR_W C_L$$

$$\text{Delay} = kR_{sq,n} (L/W) (3WC_d) + kR_{sq,n} (L/W) C_L$$

Inverter with Load



$$\begin{aligned}
 \text{Delay} &= kR_W C_{in} (C_{int}/C_{in} + C_L/C_{in}) \\
 &= 3kLR_{sq,n} C_g [C_d/C_g + C_L/(3WC_g)] \\
 &= \text{Delay (Internal)} + \text{Delay (Load)}
 \end{aligned}$$

Delay Formula

$$\text{Delay} \sim R_W (C_{int} + C_L)$$

$$t_p = kR_W C_{in} (C_{int}/C_{in} + C_L/C_{in}) = t_{inv} (\gamma + f)$$

$$C_{int} = \gamma C_{in} (\gamma \approx 1 \text{ for inverter})$$

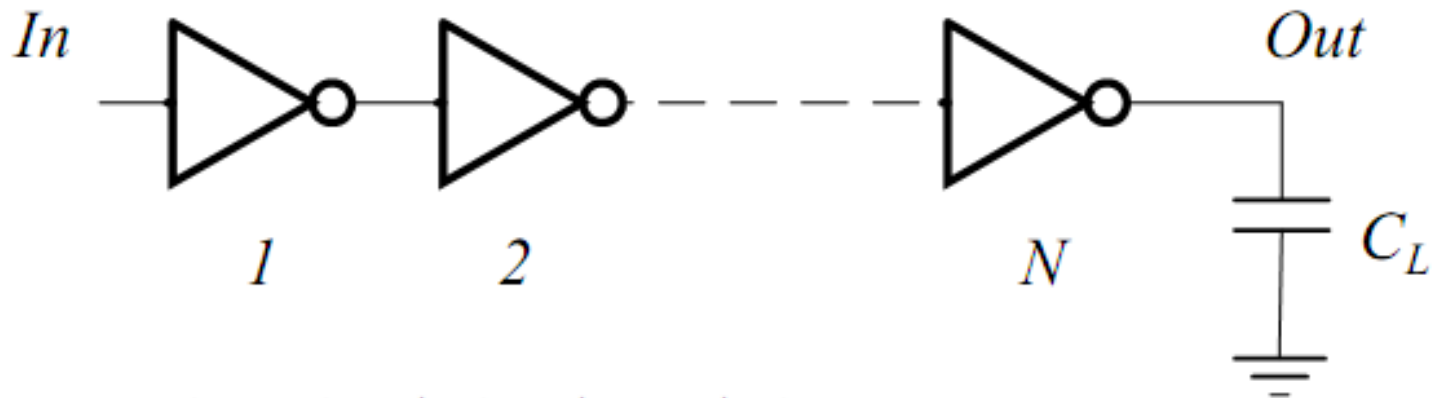
$$f = C_L/C_{in} - \text{electrical fanout}$$

$$R_W = R_{sq}(L/W); C_{in} = 3WC_g$$

$$t_{inv} = 3 \cdot \ln(2) \cdot L \cdot R_{sq} C_g$$

t_{inv} is independent of sizing of the gate!!!

Apply to Inverter Chain



$$t_p = t_{p1} + t_{p2} + \dots + t_{pN}$$

$$t_{pj} = t_{inv} \left(\gamma + \frac{C_{in,j+1}}{C_{in,j}} \right)$$

$$t_p = \sum_{j=1}^N t_{p,j} = t_{inv} \sum_{i=1}^N \left(\gamma + \frac{C_{in,j+1}}{C_{in,j}} \right), \quad C_{in,N+1} = C_L$$

Optimal Tapering for Given N

- Delay equation has $N-1$ unknowns, $C_{in,2} \dots C_{in,N}$
- **To minimize the delay**, find $N-1$ partial derivatives:

$$t_p = \dots + t_{inv} \frac{C_{in,j}}{C_{in,j-1}} + t_{inv} \frac{C_{in,j+1}}{C_{in,j}} + \dots$$

$$\frac{dt_p}{dC_{in,j}} = t_{inv} \frac{1}{C_{in,j-1}} - t_{inv} \frac{C_{in,j+1}}{C_{in,j}^2} = 0$$

Optimal Tapering for Given N (cont'd)

- Result: every stage has equal fanout:

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

- In other words, size of each stage is geometric mean of two neighbors:

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

- Equal fanout \rightarrow every stage will have same delay

Optimum Delay and Number of Stages

- When each stage has same fanout f :

$$f^N = F = C_L / C_{in,1}$$

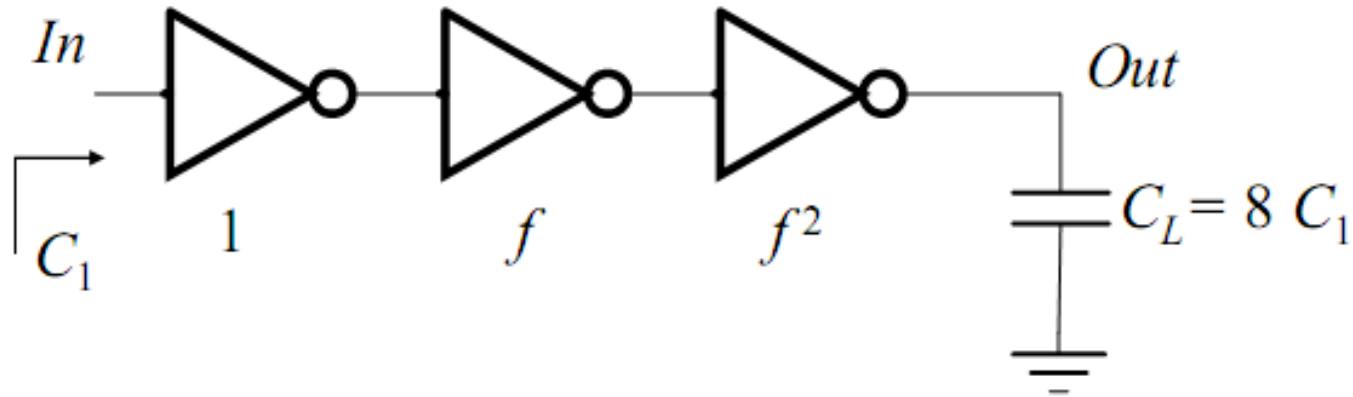
- Effective fanout of each stage:

$$f = \sqrt[N]{F}$$

- Minimum path delay:

$$t_p = Nt_{inv} \left(\gamma + \sqrt[N]{F} \right)$$

Example



C_L/C_1 has to be evenly distributed across $N = 3$ stages:

$$f = \sqrt[3]{8} = 2$$

Delay Optimization Problem #2

- You are given:
 - The size of the first inverter
 - The size of the load that needs to be driven
- Your goal:
 - Minimize delay by finding optimal number and sizes of gates
- So, need to find N that minimizes:

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

Solving the Optimization

- Rewrite N in terms of fanout/stage f :

$$f^N = C_L / C_{in} \rightarrow N = \frac{\ln(C_L / C_{in})}{\ln f}$$

$$t_p = N t_{inv} \left((C_L / C_{in})^{1/N} + \gamma \right) = t_{inv} \ln(C_L / C_{in}) \left(\frac{f + \gamma}{\ln f} \right)$$

$$\frac{\partial t_p}{\partial f} = t_{inv} \ln(C_L / C_{in}) \cdot \frac{\ln f - 1 - \gamma/f}{\ln^2 f} = 0$$

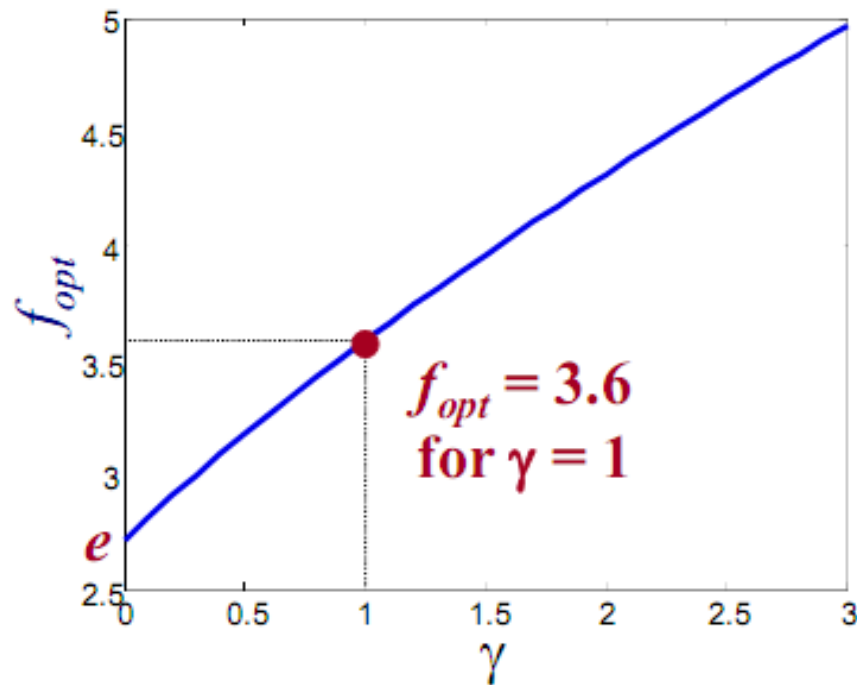
$$f = \exp(1 + \gamma/f)$$

$$\text{For } \gamma = 0, f = e, N = \ln(C_L / C_{in})$$

Optimum Effective Fanout f

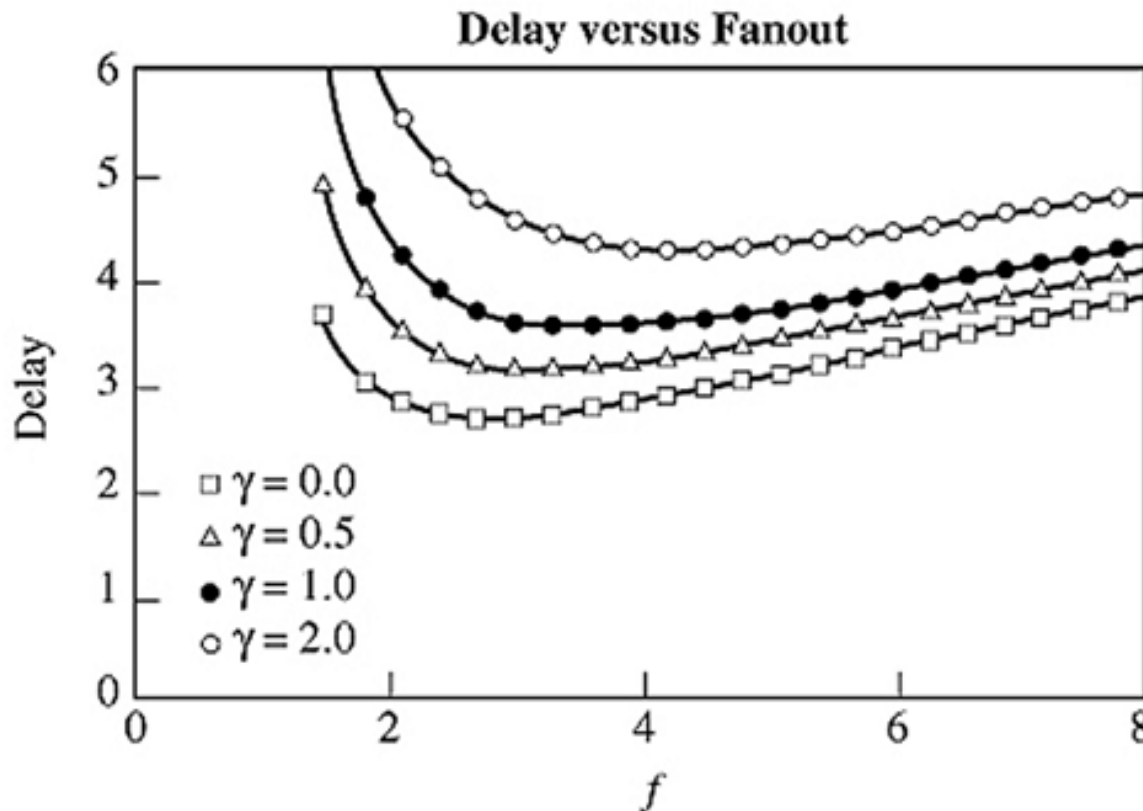
- Optimum f for given process defined by γ

$$f = \exp(1 + \gamma / f)$$



- Intuition: why does f go up with γ ?

In Practice: Plot of Total Delay

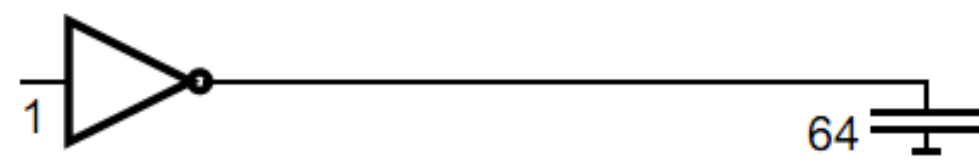
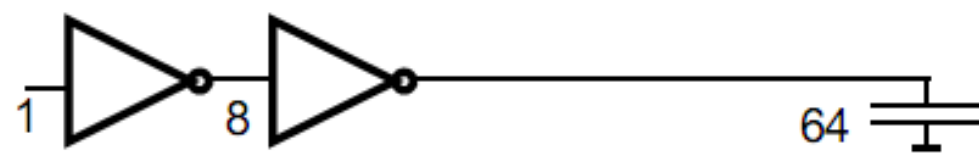
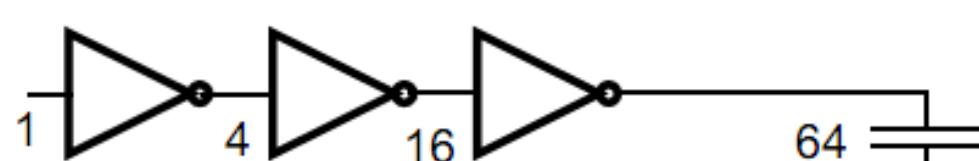
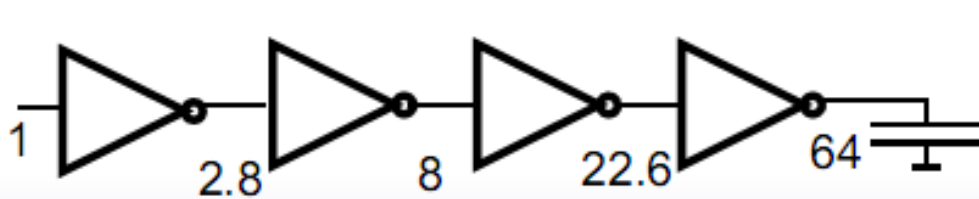


[Hodges, p.281]

□ Curves very flat for $f > 2$

- Simplest/most common choice: $f = 4$

Buffer Design

	N	f	t_p
	1	64	65
	2	8	18
	3	4	15
	4	2.8	15.3

Normalized Delay As a Function of F

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

<i>F</i>	Unbuffered	Two Stage	Inverter Chain
10	11	8.3	8.3
100	101	22	16.5
1000	1001	65	24.8
10,000	10,001	202	33.1

$$4^N = F = \frac{C_L}{C_{in}}$$

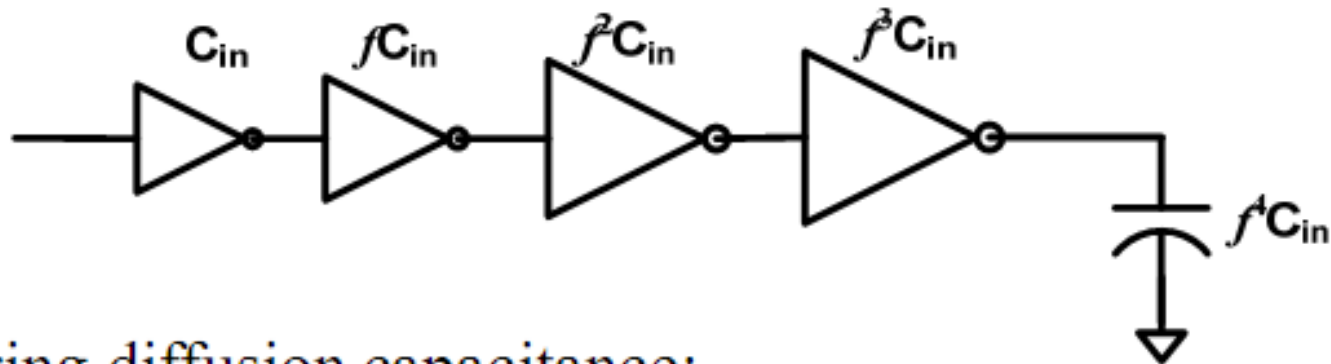
$$N = \frac{\ln(F)}{\ln(4)}$$

($\gamma = 1$)

For $F=10000$ $N=6.64$ take $N=7$

$$t_p = 7 \cdot t_{inv} (1 + \sqrt[7]{10000}) = 33.1 \cdot t_{inv}$$

What About Energy (and Area)?



Ignoring diffusion capacitance:

$$\begin{aligned} C_{tot} &= C_{in} + f \cdot C_{in} + \dots + f^N \cdot C_{in} \\ &= C_{in} \cdot (1 + f + \dots + f^N) \\ &= C_{in} + C_{in} \cdot f^N + C_{in} \cdot f \cdot (1 + f + \dots + f^{N-1}) \end{aligned}$$

2)

Overhead !!! $f(f^{N-1}-1) / (f-1)$

Example ($\gamma=0$): $C_L = 20\text{pF}$; $C_i = 50\text{fF} \rightarrow N = 6$

Fixed: 20pF

Overhead: 11.66pF !!!

Example Overhead Numbers

Example: $C_L = 20\text{pF}$; $C_{in} = 50\text{fF}$

