Transistor Sizing
Logical Effort

Lecture 7

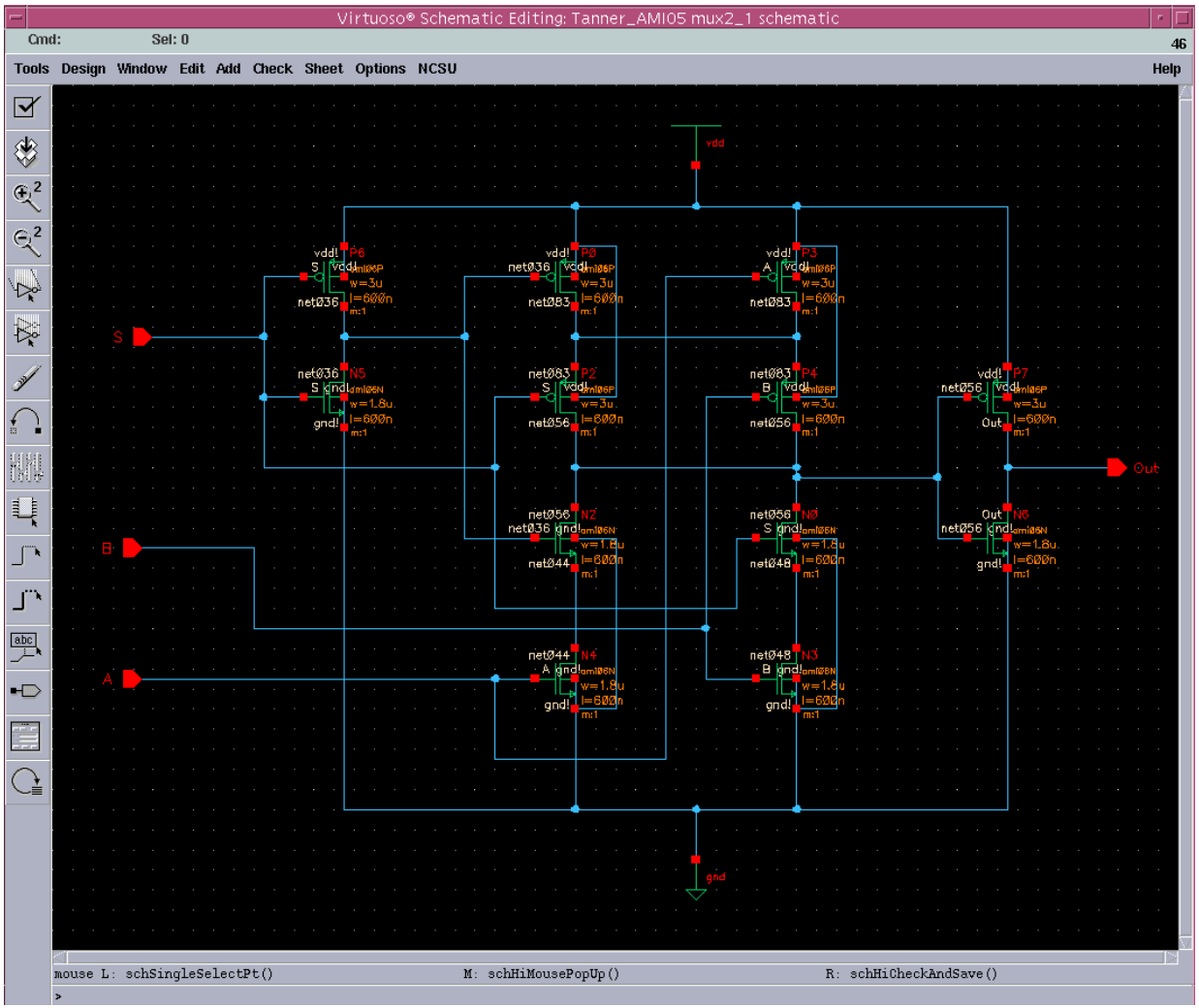
18-322 Fall 2003

Textbook: [5.1, 5.2, 6.1, 6.2-6.2.1]

Overview

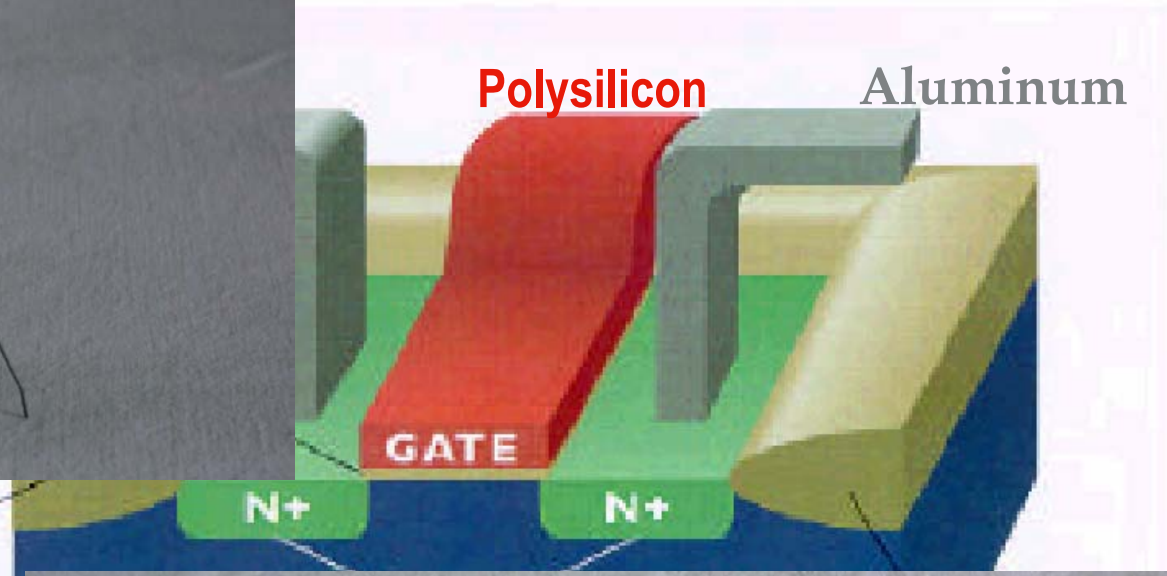
- Static CMOS circuit design
  - Transistor sizing
    - For symmetrical response
    - For performance
  - Large Fanin gates
  - Chains of logic gates
- Logical effort introduction

Transistors Everywhere…



- Node centric perspective
  - Focus on devices and their properties (today)
- Network centric perspective
  - Focus on interconnects (next lectures)

The MOS Transistor



It is very diffic
modern transi
calculations w

What is a Transistor?

$$R_n = \frac{1}{\mu_n C_{ox}(V_{GS} - V_{Tn})}\left(\frac{L}{W}\right)$$

Increasing W decreases the resistance which allows more current to flow!

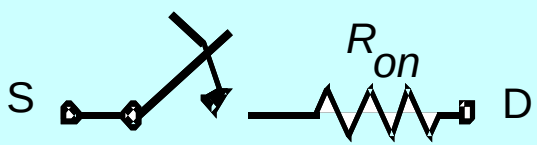
$C_{ox} = \varepsilon_{ox}/t_{ox}$ oxide capacitance [F/cm^2]

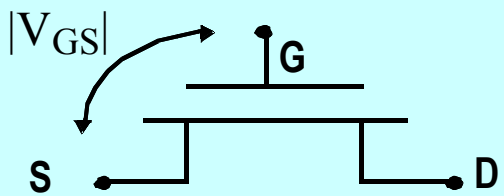
$$\beta_n = \mu_n C_{ox}\left(\frac{W}{L}\right) = k_n'\left(\frac{W}{L}\right)$$ device transconductance [A/V^2]

$$C_G = C_{ox}(WL)$$ gate capacitance [F]

Switch! $\longleftrightarrow$ MOS Transistor

$V_{GS} \geq V_T$





What is Different Between nFET and pFET?

$$R_n = \frac{1}{\beta_n (V_{DD} - V_{Tn})}$$

$$\beta_n = \mu_n C_{ox} \left(\frac{W}{L} \right)_n$$

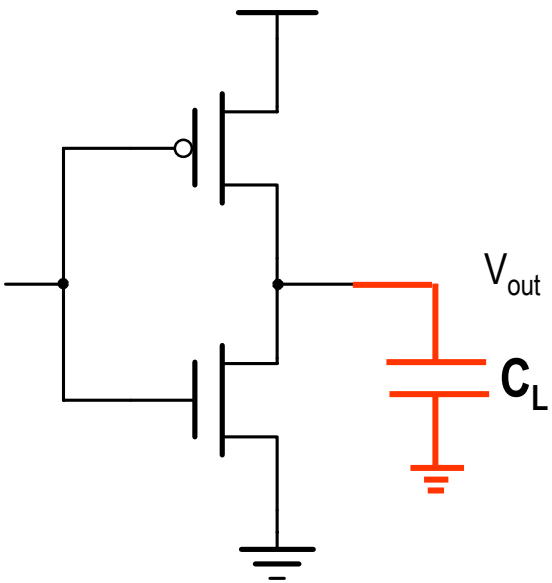
$$\frac{\mu_n}{\mu_p} = r \quad \text{typically } (2 \ldots 3)$$

$$R_p = \frac{1}{\beta_p (V_{DD} - |V_{Tp}|)}$$

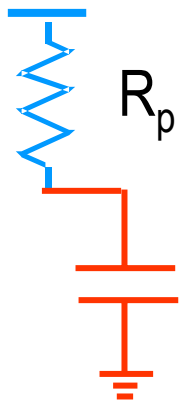
$$\beta_p = \mu_p C_{ox} \left(\frac{W}{L} \right)_p$$

Balancing Rise and Fall Time

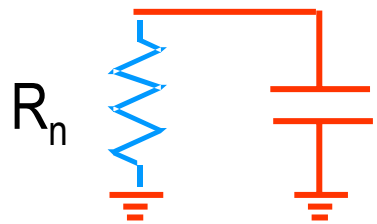
- Inverter



charging
V_{out} rising



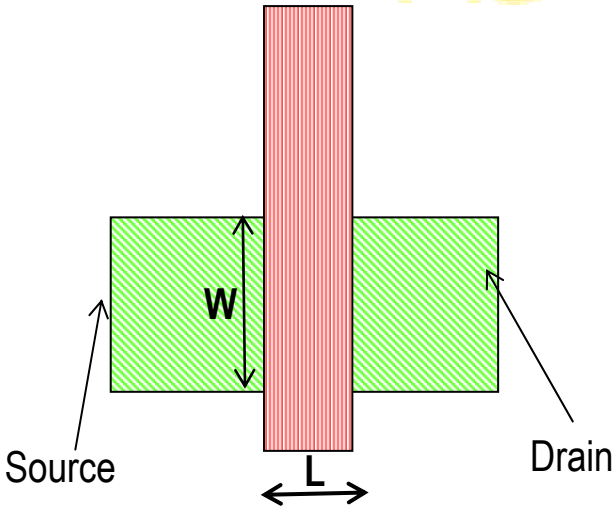
discharging
V_{out} falling



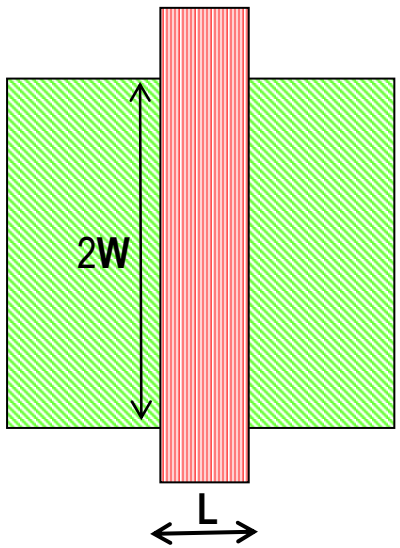
If $(W/L)_p = r\,(W/L)_n$ then $\beta_n = \beta_p$ $(R_n = R_p) \rightarrow$ symmetrical inverter

Make PMOS bigger (wider) by r times

FET Sizing and the Unit Transistor

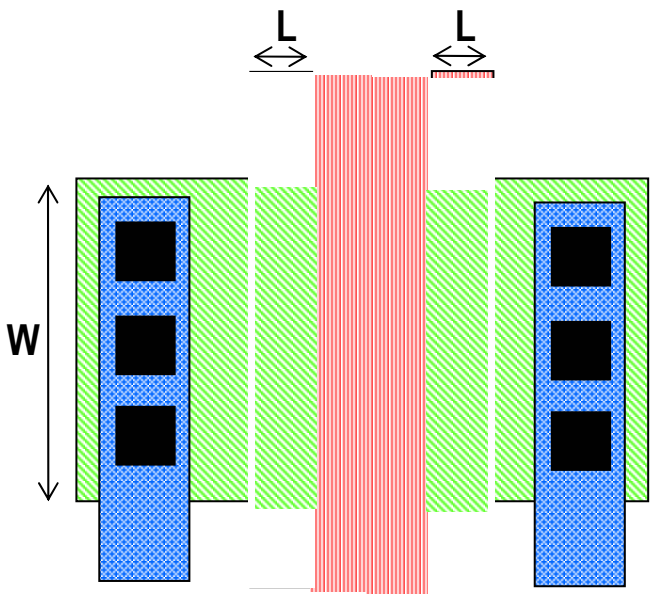


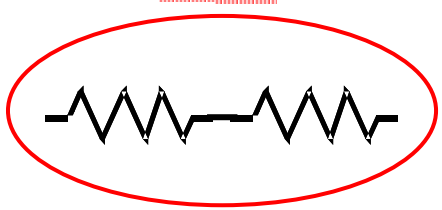
- The electrical characteristics of transistors determine the switching speed of a circuit
  - Need to select the aspect ratios $(W/L)_n$ and $(W/L)_p$ of *every* FET in the circuit



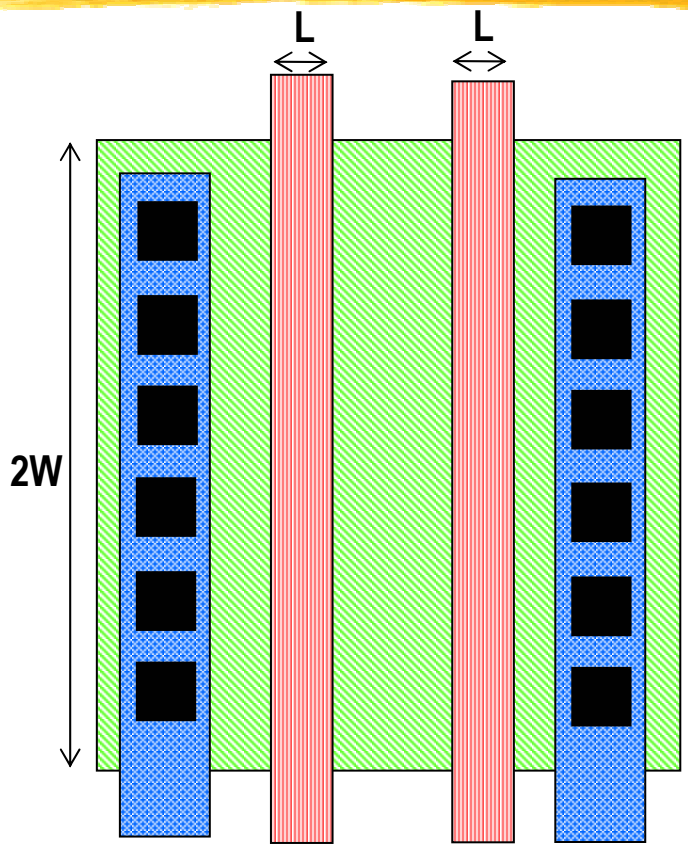
- Define the *unit transistor* (R_1, C_1)
  - L/W_{min} -> highest resistance (needs scaling)
  - $R_2 = R_1/2$ and $C_2 = 2C_1$
  - Separate nFET and pFET unit transistors
  - Unit devices are *not* restricted ti individual transistors (see next example)

Scaling of Series-Connected FET Chains



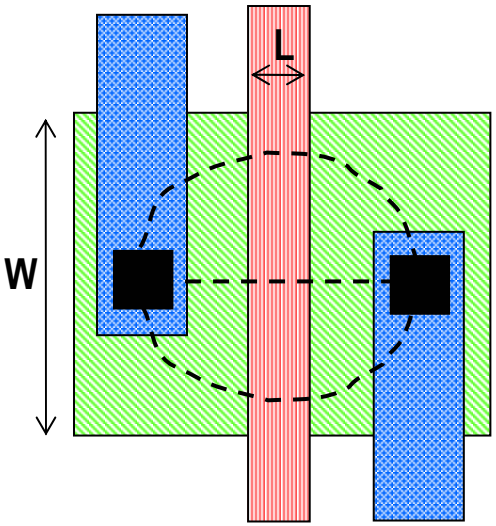


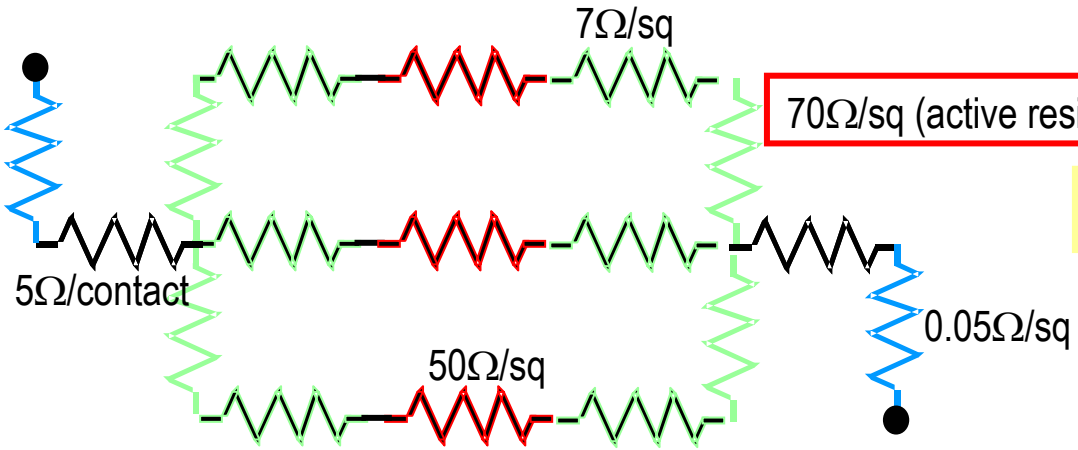
R_1



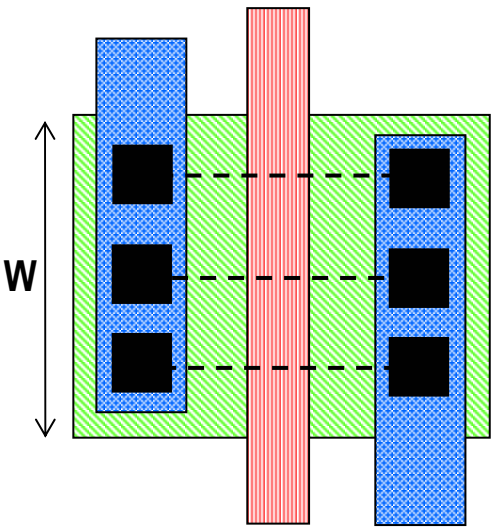
$R_2 = ½ R_1$

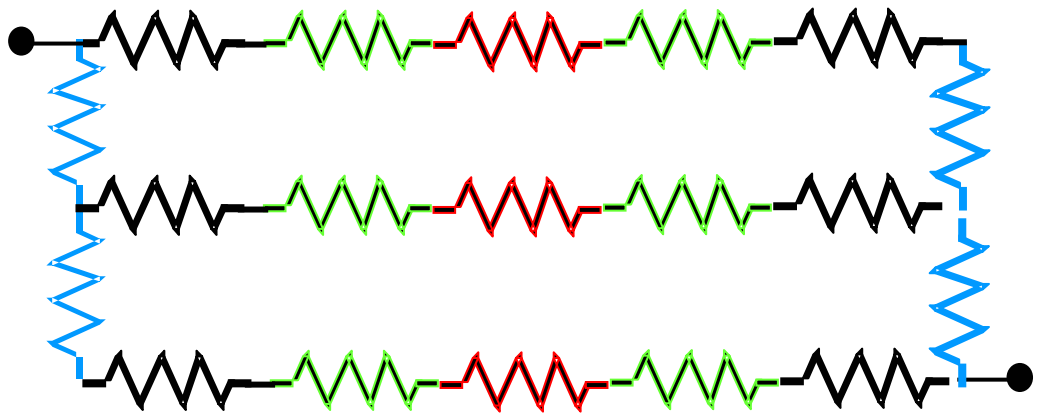
Transistor Resistance Model





~56Ω

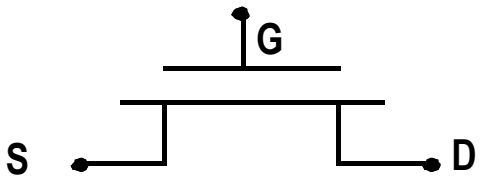


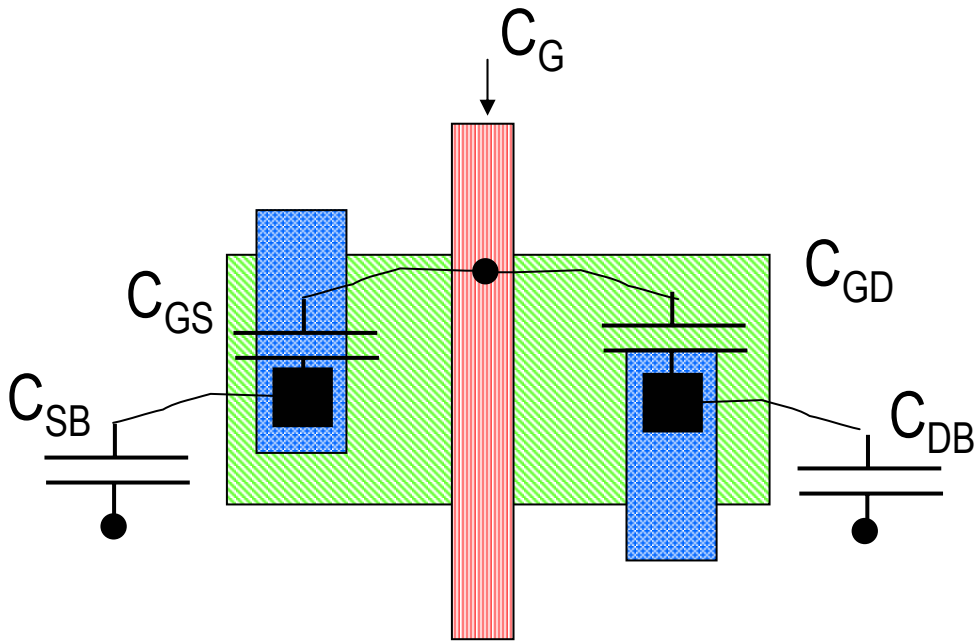


~25Ω

Better reliability

Transistor Capacitance Model



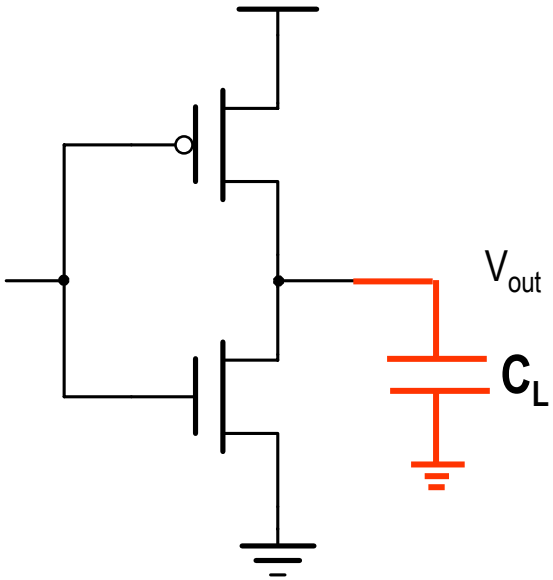


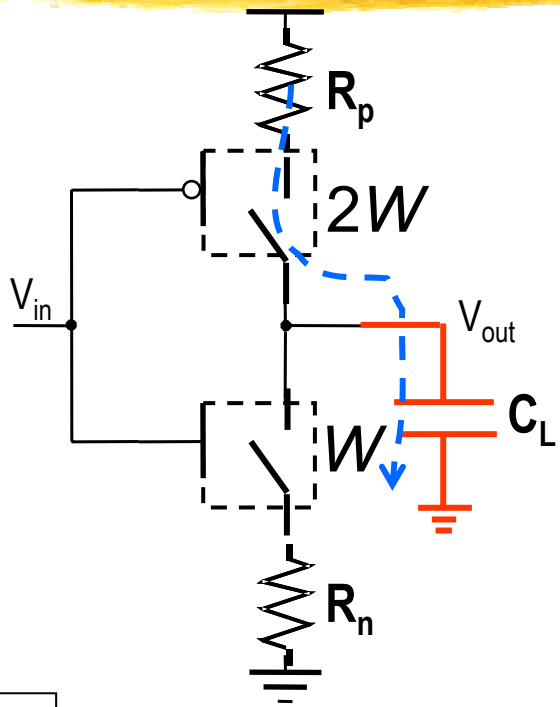
$$C_S = C_{GS} + C_{SB}$$

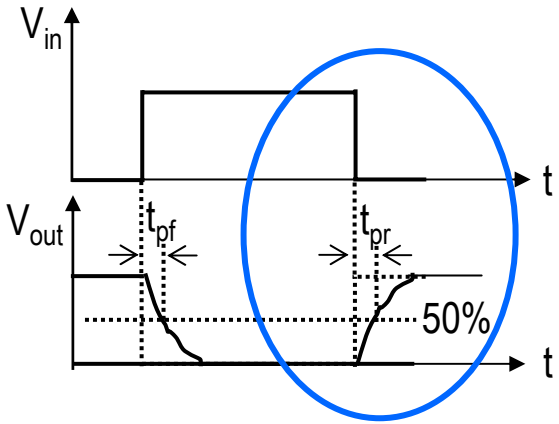
$$C_D = C_{GD} + C_{DB}$$

Capacitances increase with channel width!

Inverter Propagation Delay







$$t_{pf} = \ln 2\,(R_n C_L) = 0.69\,(R_n C_L)$$

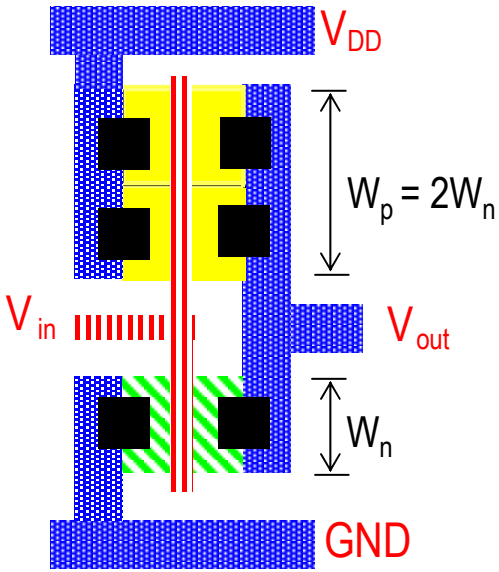
$$t_{pr} = \ln 2\,(R_p C_L)$$

$$t_p = \tfrac{1}{2}\,(t_{pf} + t_{pr})$$

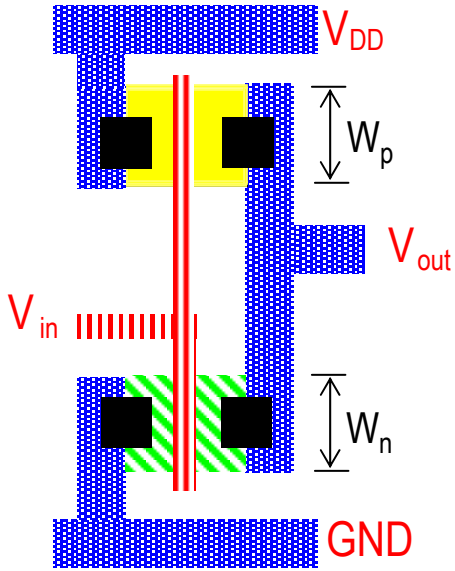
- $W_P = 2W_N$ *(symmetrical design)*
  - approx. equal resistances $R_n = R_p$
  - approx. equal rise and fall delays

- $t_p \sim 1/(\beta V_{DD})\, C_L$ -> minimize load, increase width of the driving transistor, increase V_{DD}???

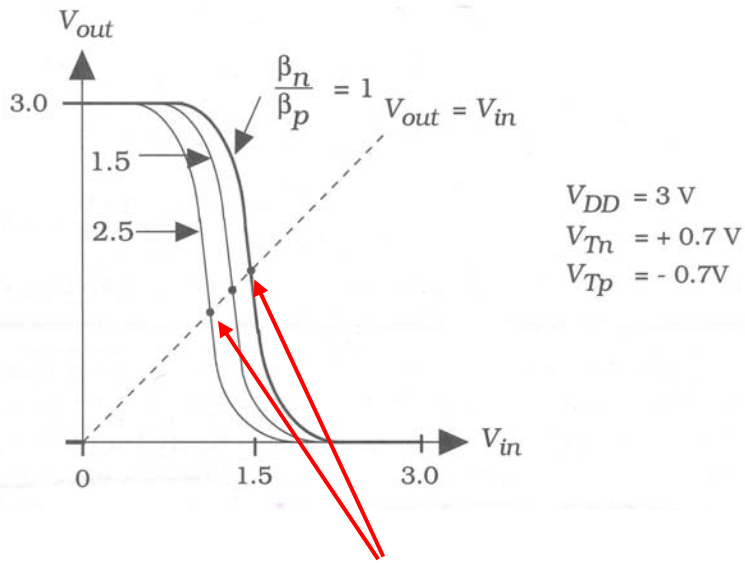
Symmetrical Inverter



Larger pFET design

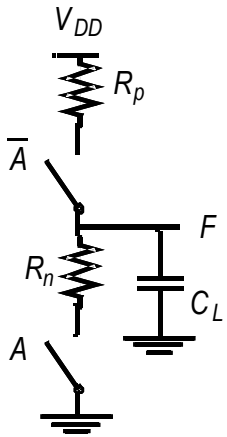


Equal aspect ratios

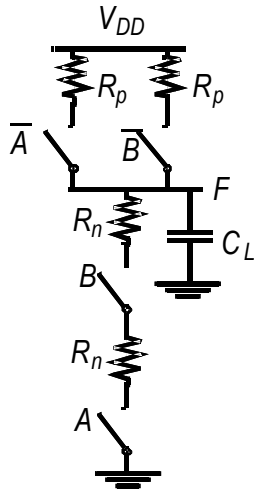


Dependence of V_M on the device ratio

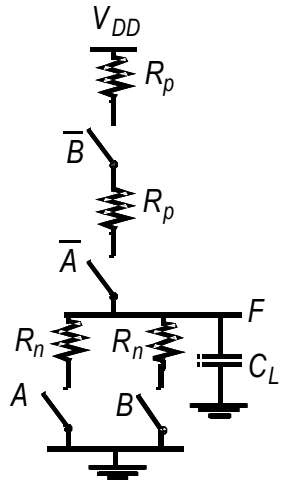
Propagation Delay Analysis - The Switch Model



(a) Inverter



(b) 2-inputNAND

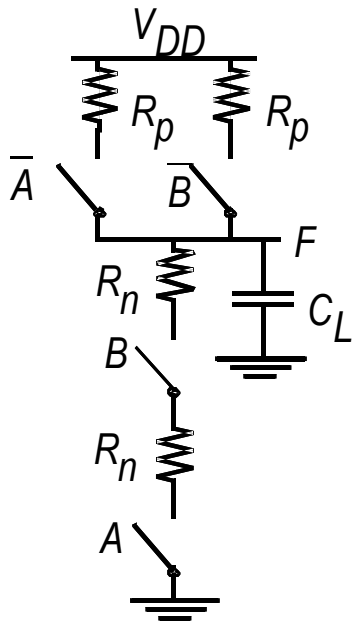


(c) 2-input NOR

$t_p = 0.69\ R_{on}\ C_L$

(assuming that C_L dominates!)

Analysis of Propagation Delay



2-input NAND

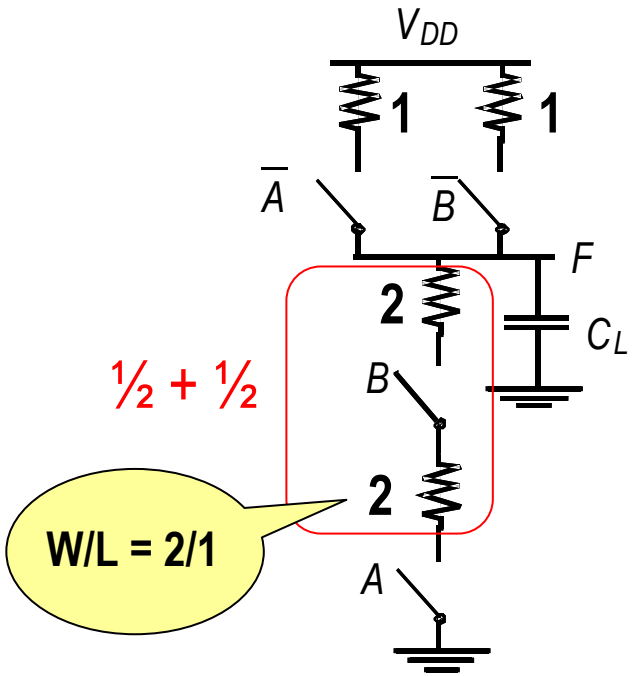
1. Assume R_n=R_p= resistance of minimum sized NMOS inverter
2. Determine "Worst Case Input" transition (Delay depends on input values)
3. Example: t_{pLH} for 2input NAND
   - Worst case when only ONE PMOS Pulls up the output node
   - For 2 PMOS devices in parallel, the resistance is lower

$$t_{pLH} = 0.69 R_p C_L$$

4. Example: t_{pHL} for 2input NAND
   - Worst case : TWO NMOS in series

$$t_{pHL} = 0.69(2R_n) C_L$$

Transistor Sizing: NAND2

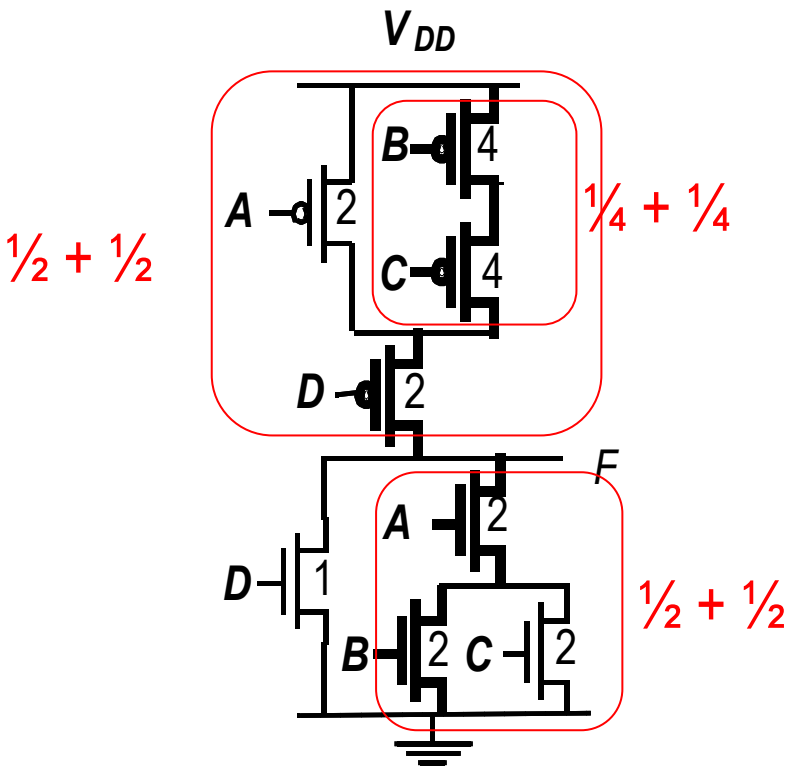


Input Dependent

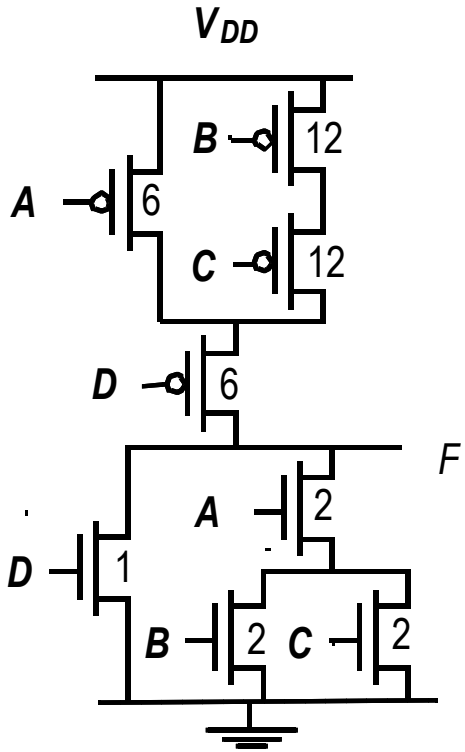
Focus on worst-case

Here it is assumed that $R_p = R_n$

Designing for Worst-Case

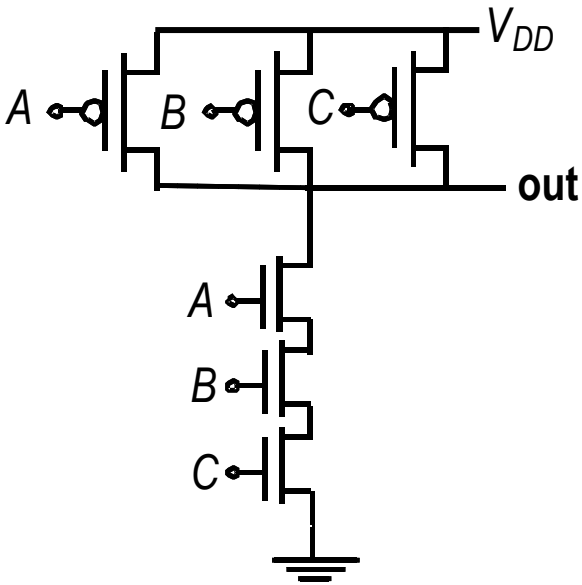


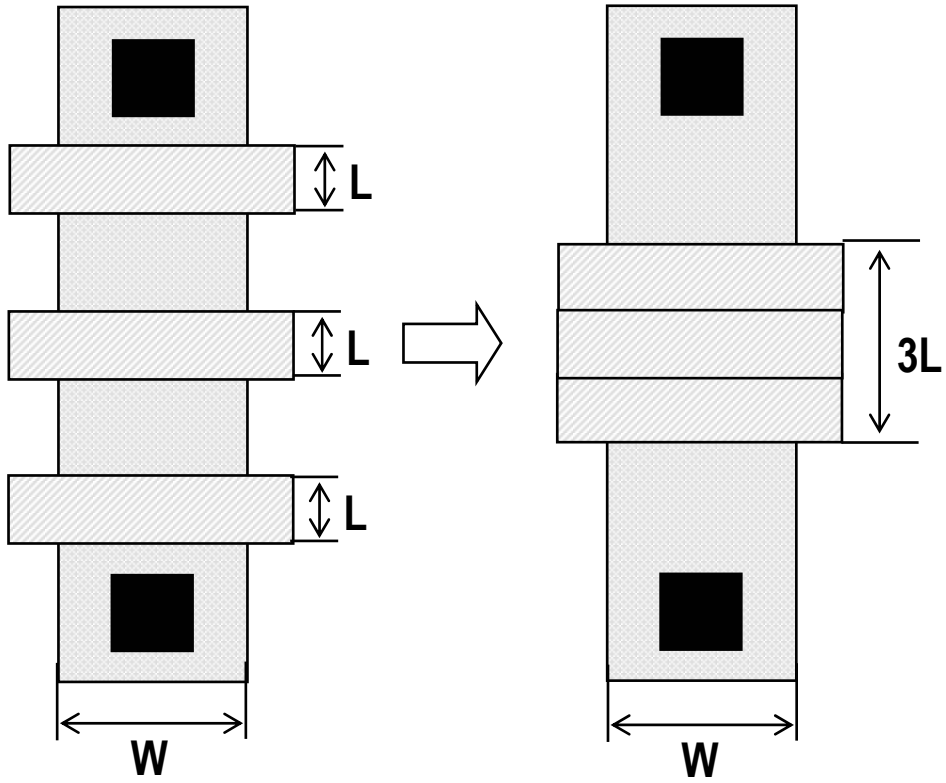
Here it is assumed that $R_p = R_n$



(Wp = 3Wn assumed)

Equivalent Inverter





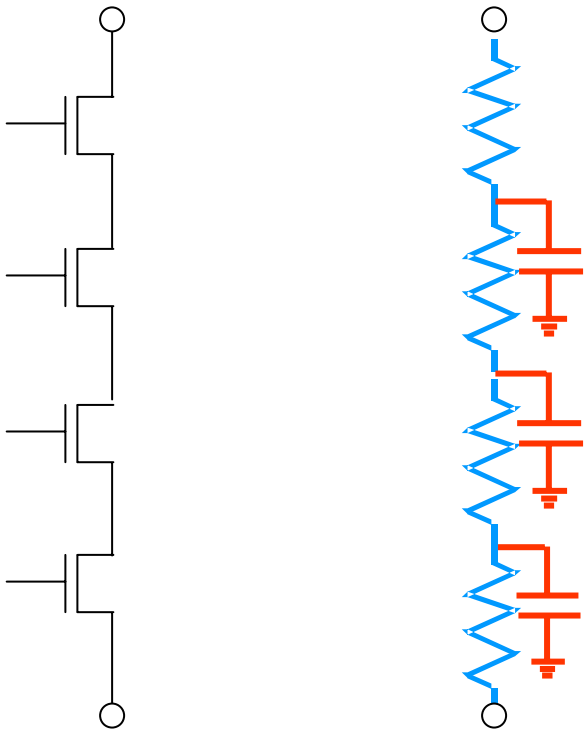
$$\beta_{series} = \beta_n / 3$$

$$t_{series} = k \frac{C_L}{(\beta_n / 3) V_{DD}}$$

Big Fanin Gates

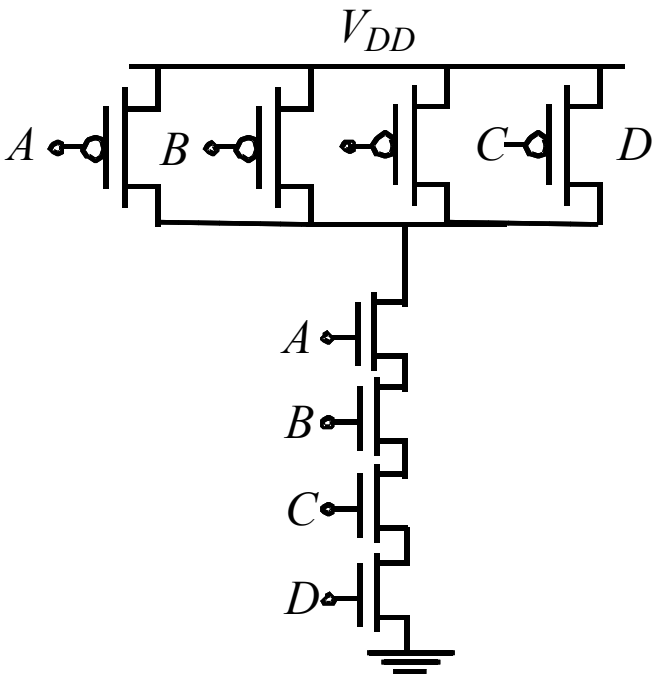
- So, how would NAND scale with fanin N
  - Assume all transistors are 1x
  - Rise time: best case $C_L R_p/N$, worst case: $C_L R_p$
  - Fall time: best/worst case $NC_L R_n$
  - Problem 1: Big fanin gates have big difference for rise and fall
  - Problem 2: The truth is actually worse than that

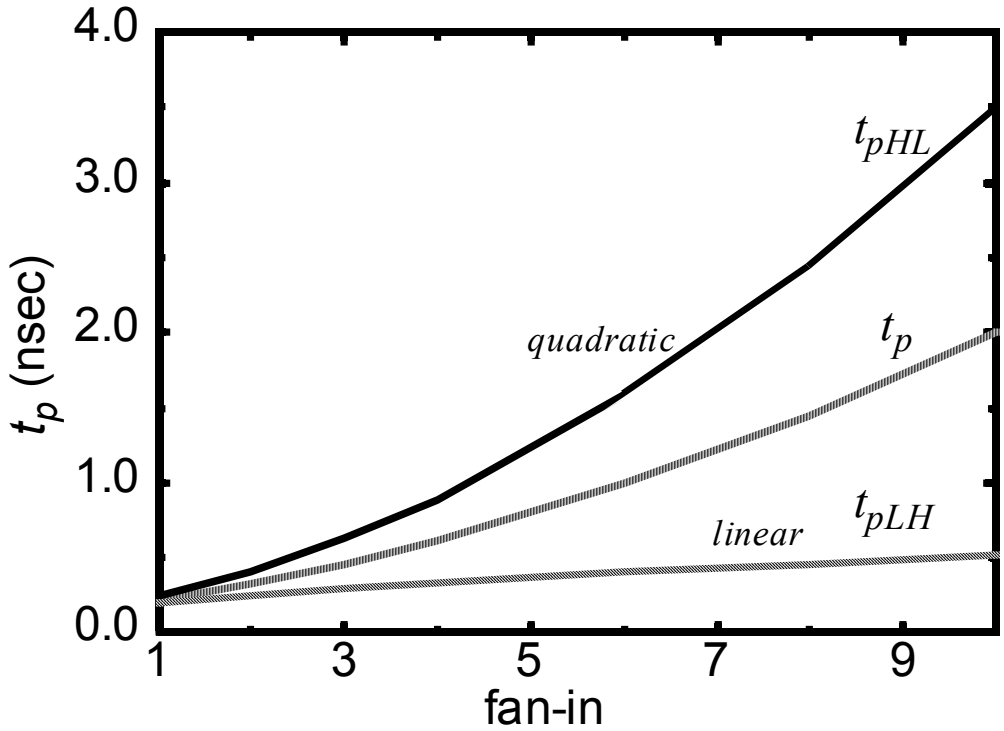
The Truth about Transistor Chains



- Series Transistors add series resistance
- Series Transistors also add Capacitance
- We'll talk about estimating delay of distributed RC on Thursday
- Result: Quadratic, not linear relationship of delay and fanin

t_p as a function of Fan-In

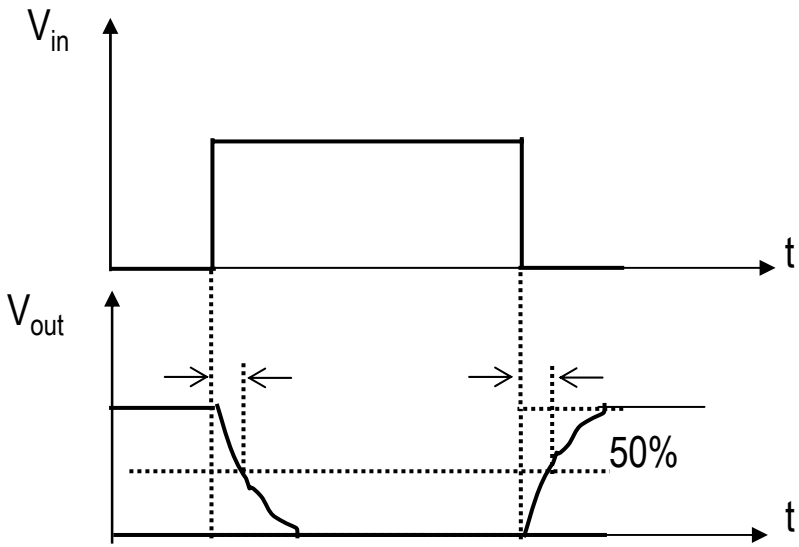




AVOID LARGE FAN-IN GATES! (Typically not more than FI < 4)

Chains of Gates

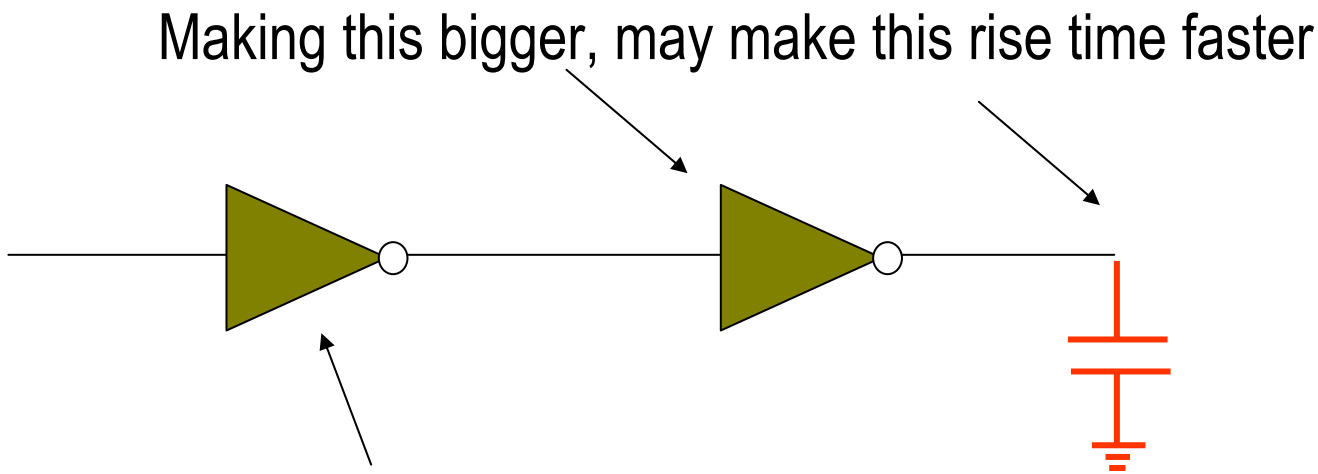
- Making W/L bigger decreases R_{on}
- Decreases t_{HL} or t_{LH}



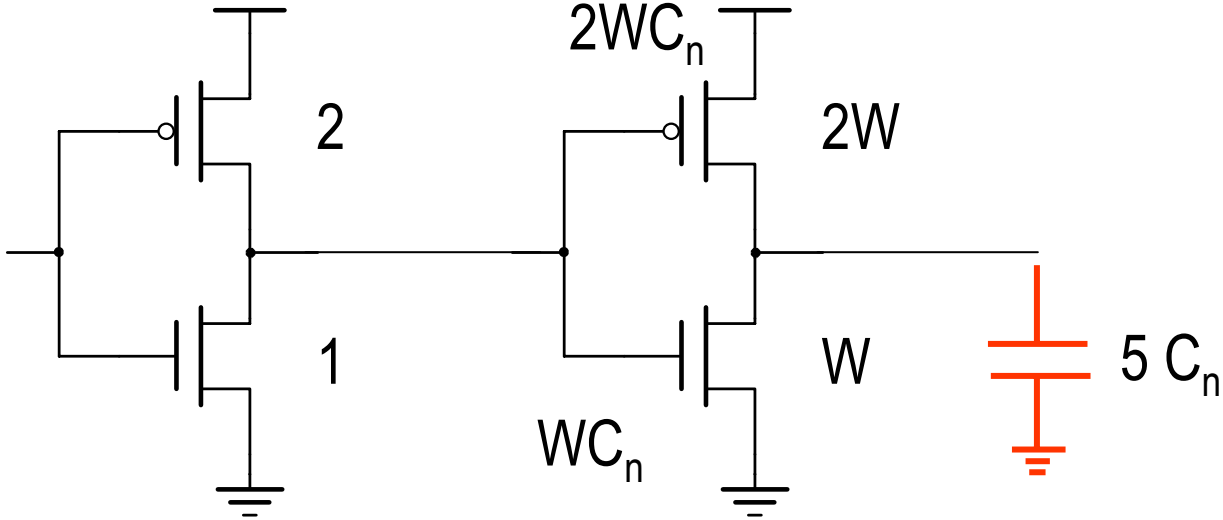
- But wait! Doesn't increased W increase capacitance

The Big Trade-off

- Making a gate bigger increases it output drive
- But also increases its input capacitance



How to Optimize?



Delay ~ 3 W C_n * R_n + 5C_n * R_n / W

How to optimize?

Summary

- Switching delays increase with the external load
- The layout geometries affect the transient response of logic gates
- Switching delays increase with the fan-in
- Increasing the “drive” of a gate increases the load to be driven by the previous gate

Overview

- ✓ Static CMOS circuit design
  - Transistor sizing
    - For symmetrical response
    - For performance
- Logical effort

Logical Effort

- A way of thinking about delay in MOS circuits. It seeks to determine quickly a circuit's *maximum* possible speed and *how* to achieve it.

- Book: "*Logical effort: Designing fast CMOS Circuits*" by I. Sutherland, B. Sproull and D. Harris

Formula for Gate Delay (inverter)

$$t_p = 0.69 R_{eq}\left(C_{int} + C_{ext}\right)$$

$$t_p = t_{p0}\left(1 + \frac{C_{ext}}{C_{\text{int}}}\right)$$

$$t_p = t_{p0}\left(1 + \frac{C_{ext}}{\gamma C_g}\right)$$

"intrinsic": if Cext = 0 C_{int} is linear with gate size

$$f = \frac{C_{ext}}{C_g}$$

f is "effective fanout"

$$t_p = t_{p0}\left(1 + \frac{f}{\gamma}\right)$$

Definitions

- The logical effort of a logical gate is defined as the ratio of its input capacitance to that of an *inverter* that delivers equal output current.
  - How much worse a gate is at producing output current than an inverter, assuming inverter and gate have same input capacitance
  - How much more input capacitance a gate presents to deliver the same output current as an inverter
- Use inverter as the reference gate

Delay Formula for Complex Gates

$$t_p = t_{p0}\left(1 + \frac{f}{\gamma}\right) \Rightarrow t_p = t_{p0}\left(p + \frac{gf}{\gamma}\right)$$

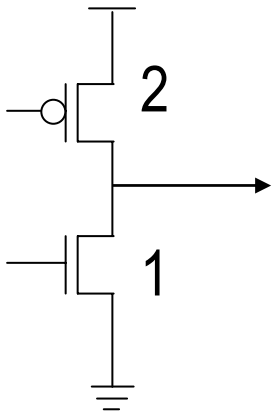
g is logical effort

Assume PMOS 2x wider than NMOS in inverter gates

Rise time == Fall time

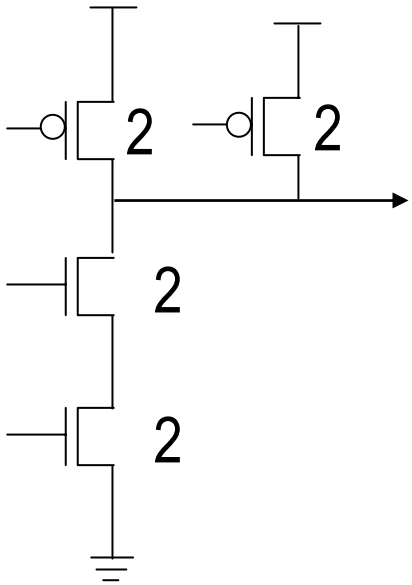
Gate	1 inp	2 inp	3 inp
INV	1		
NAND		4/3	5/3
NOR		5/3	7/3
XOR		4	12

Determining Logical Effort



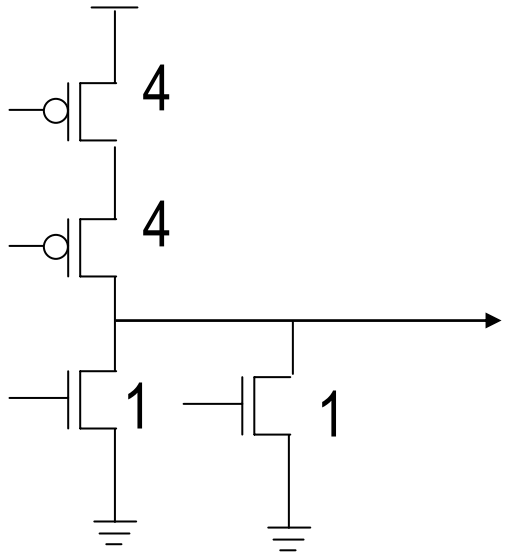
$C_{in} = 3$

$g = 1$



$C_{in} = 4$

$g = 4/3$



$C_{in} = 5$

$g = 5/3$

Delay thru a Path of Gates

$$t_p = t_{p0} \sum_{i=1}^{N} \left(p_i + \frac{g_i f_i}{\gamma} \right)$$

to Optimize:

$$f_1 g_1 = f_2 g_2 = \ldots = f_N g_N$$

Logical Effort (cont'd)

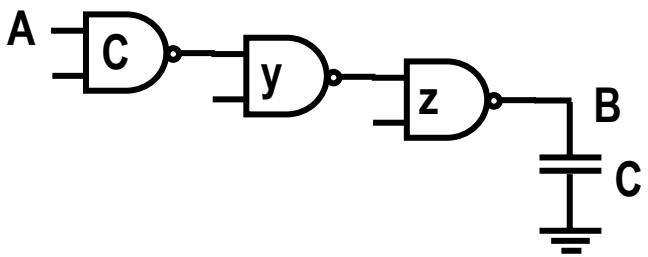
- Type of efforts
  - logical path effort ($G = \Pi g_i$)
  - electrical path effort ($F = C_{out}/C_{in}$)
  - branching effort ($B = \Pi b_i$)

- Path effort
  - $H = GFB$

Optimization

- N-stage logic network
- Idea: *The path delay is least when each stage in the path bears the same stage effort*
  - $h_i = g_i f_i = (H)^{1/N}$
- Main result: minimum delay achievable along a path
  - $\mathcal{D} = N\,(H)^{1/N} + P$ (where $P = \sum p_i$)
  - $C_{ini} = (1/f)\, g_i\, C_{outi}$ (used for transistor sizing!)
- The method of logical effort achieves an *approximate optimum*!

Example



Gate	1 inp	2 inp	3 inp
INV	1		
NAND		4/3	5/3
NOR		5/3	7/3
XOR		4	12

Gate	P
Inv	$P_{inv} = 1$
n-NAND	np_{inv}
n-NOR	np_{inv}
XOR	$4p_{inv}$

$G = (4/3)^3 = 2.37$
$B = 1$
$F = C/C = 1$

$H = 2.37$
$\mathcal{D} = 3(2.37)^{1/3} + 3(2p_{inv}) = 10$ delay units (min delay)

$f = (2.37)^{1/3} = 4/3$ (this is the stage effort)
$z = C\ (4/3) / (4/3) = C$
$y = z\ (4/3) / (4/3) = C$
(all 3 gates should have the same input capacitance)