

Digital Integrated Circuits Workshop

Week 3:
**CMOS Logic, Pass-Transistor Logic,
and Gate Sizing**



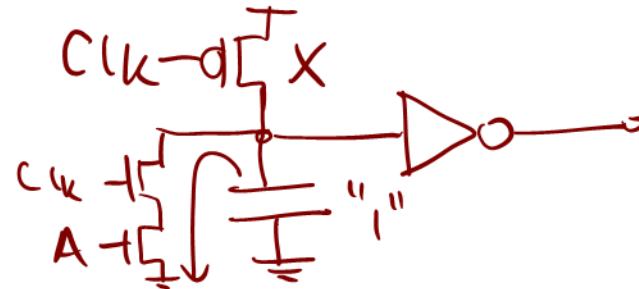
**Prof. Dejan Markovic
UCLA**

Week 3 Agenda

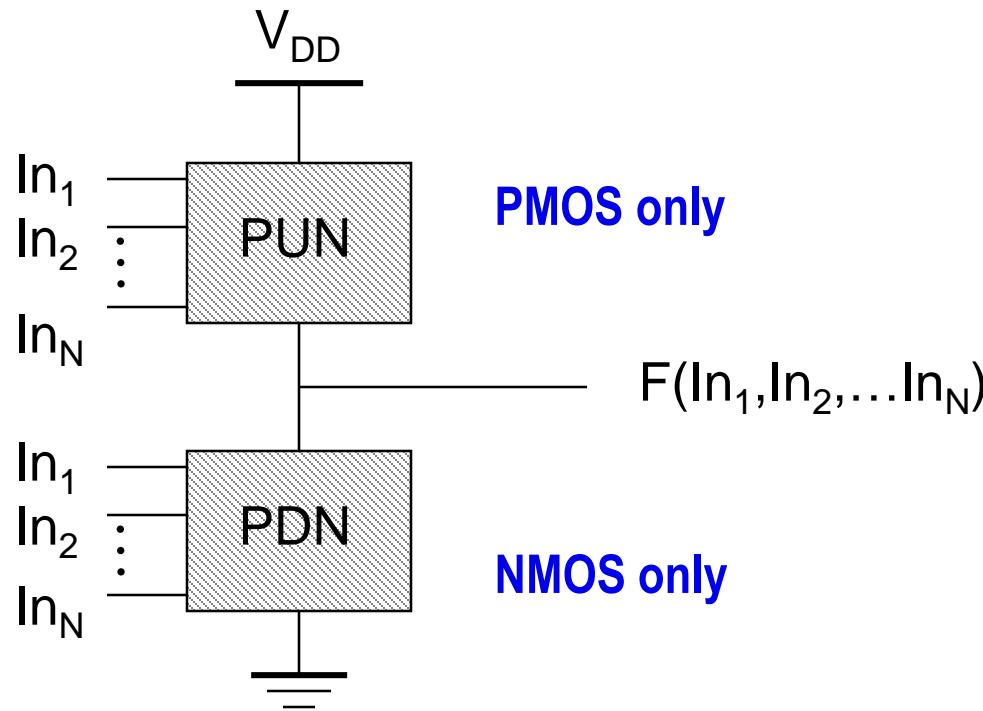
- ◆ CMOS Logic
- ◆ Pass-Transistor Logic
- ◆ Sizing of Logic Gates

Static CMOS Circuit

- ◆ At every point in time (except during the switching transients) each **gate output is connected to either V_{DD} or V_{SS}** via a low-resistive path
- ◆ The outputs of the gates **assume at all times the value of the Boolean function**, implemented by the circuit (ignoring, once again, the transient effects during switching periods)
- ◆ This is in contrast to the **dynamic** circuit class, which relies on temporary storage of signal values on the capacitance of high impedance circuit nodes.



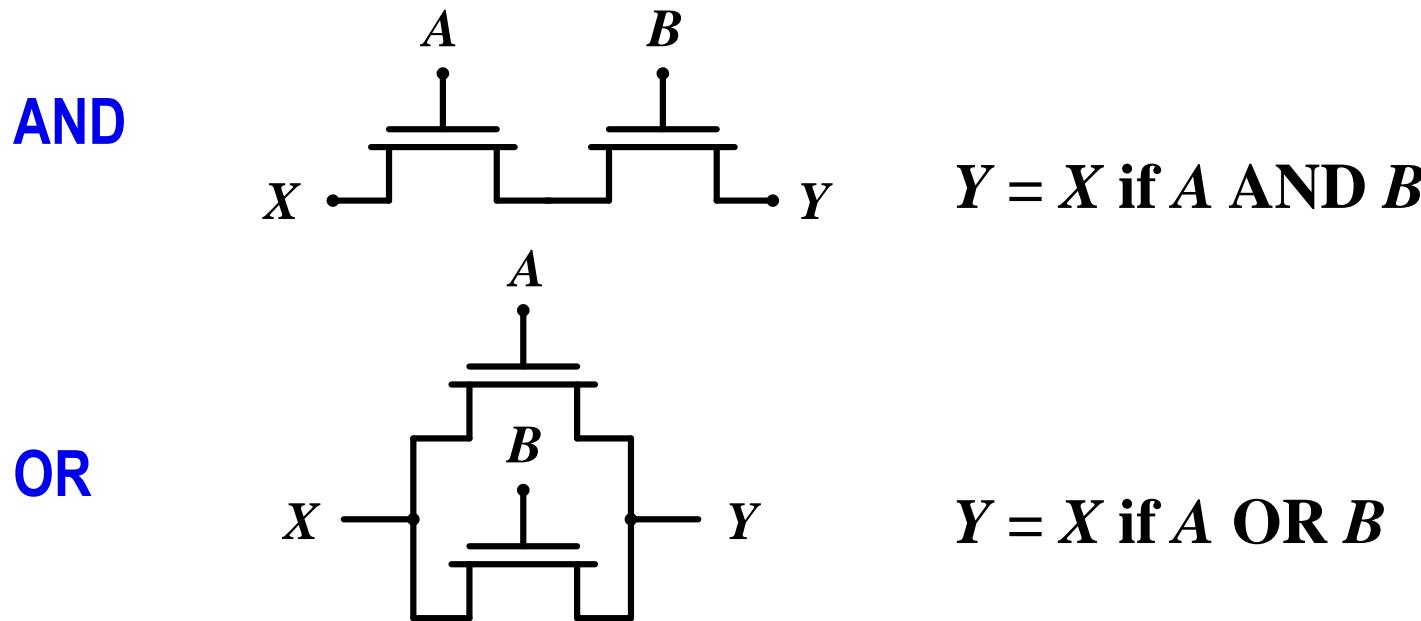
Static Complementary CMOS



- ◆ PUN and PDN are **dual** logic networks
- ◆ PUN and PDN functions are **complementary**

NMOS Transistors in Series/Parallel

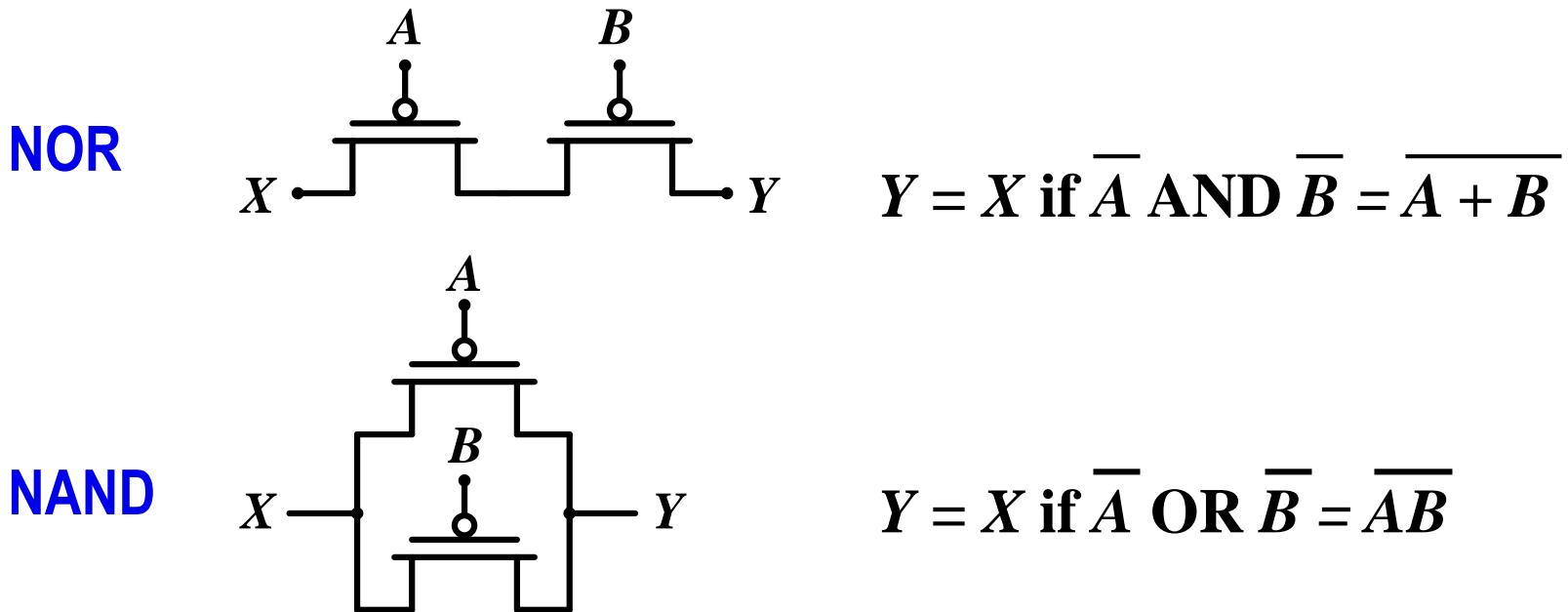
- ◆ Transistor \leftrightarrow switch controlled by its gate signal
 - NMOS switch closes when switch control input is high



- ◆ NMOS transistors pass a “strong” 0 but a “weak” 1

PMOS Transistors in Series/Parallel

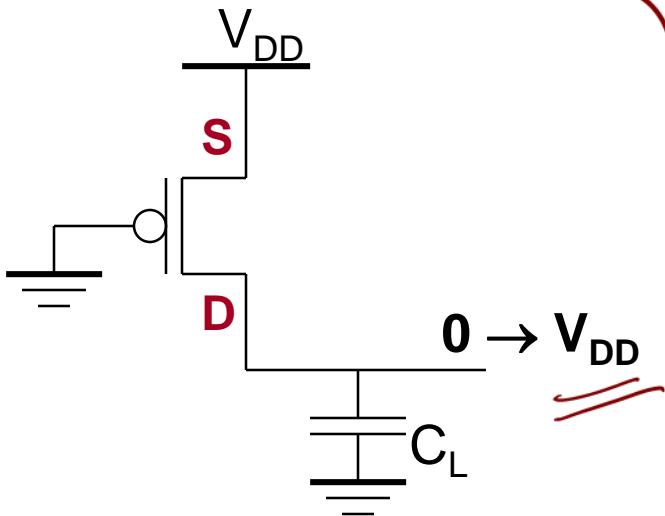
- ◆ PMOS switch closes when switch control is low



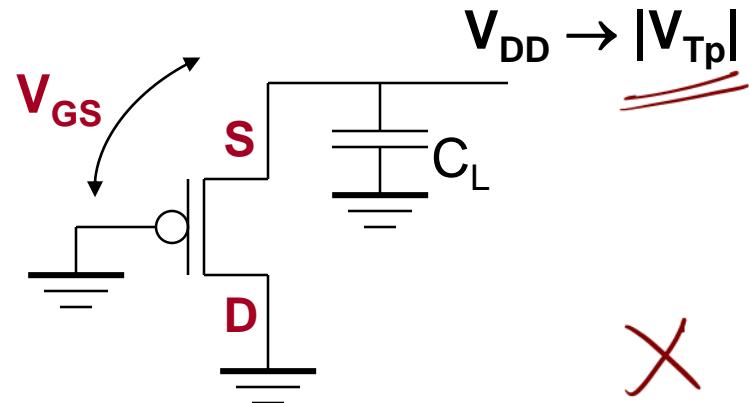
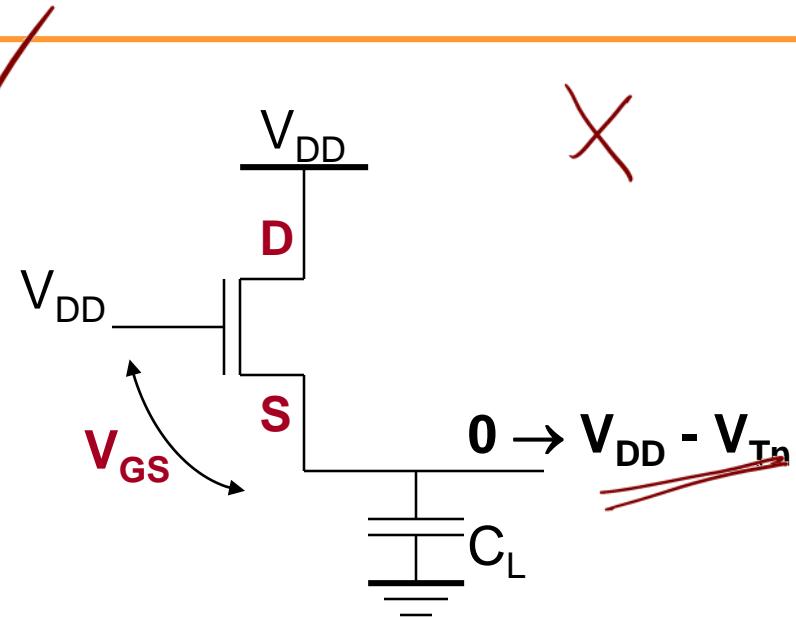
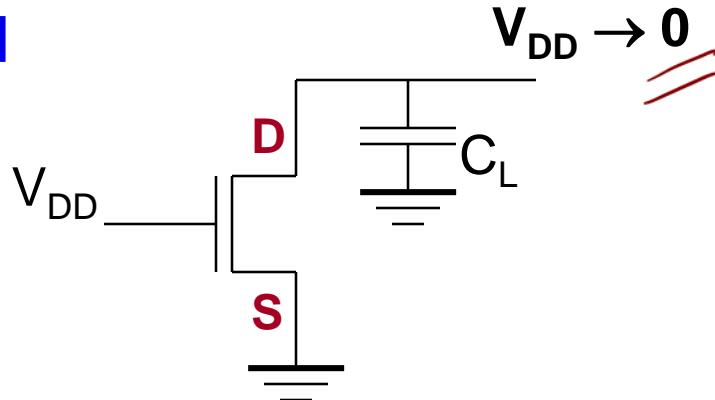
- ◆ PMOS transistors pass a “strong” 1 but a “weak” 0

Watch Out for Threshold Voltage Drops

PUN



PDN



Complementary CMOS Logic Style

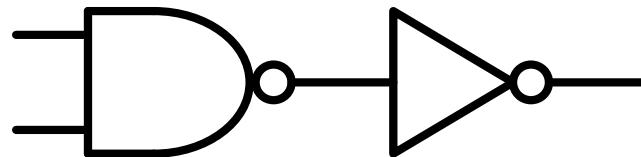
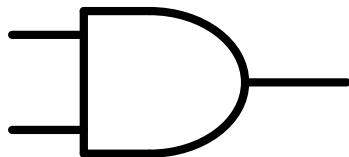
- ◆ PUP is dual to PDN

(can be shown using DeMorgan's Theorems)

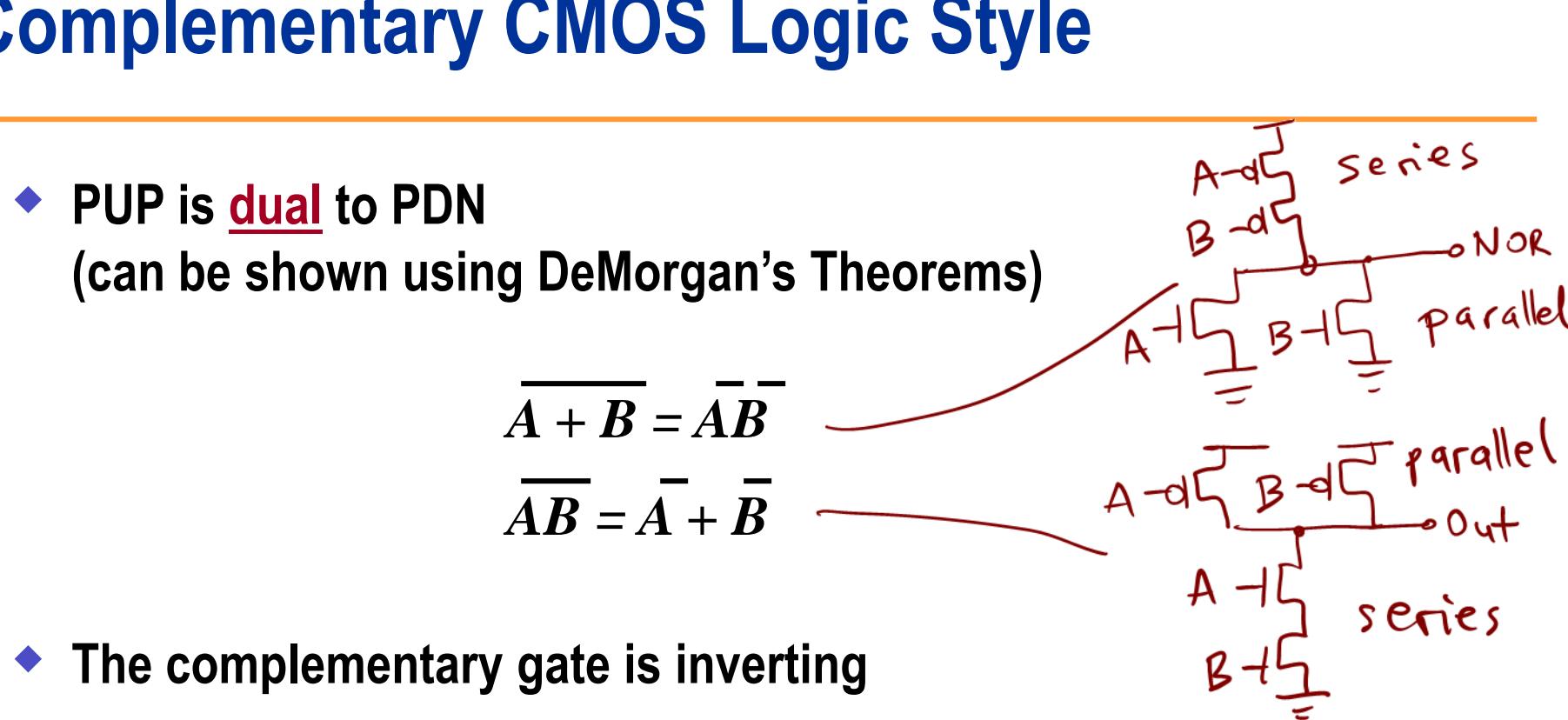
$$\overline{A + B} = \overline{A}\overline{B}$$

$$\overline{AB} = \overline{A} + \overline{B}$$

- ◆ The complementary gate is inverting



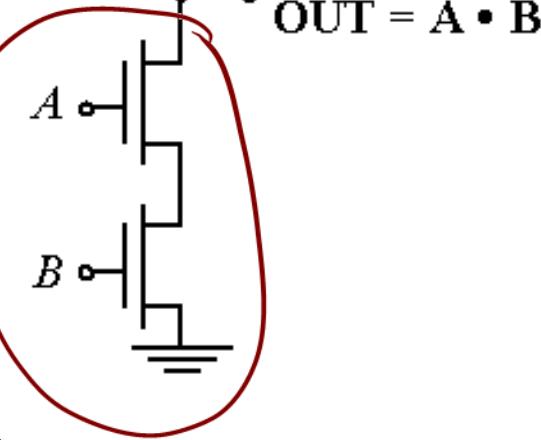
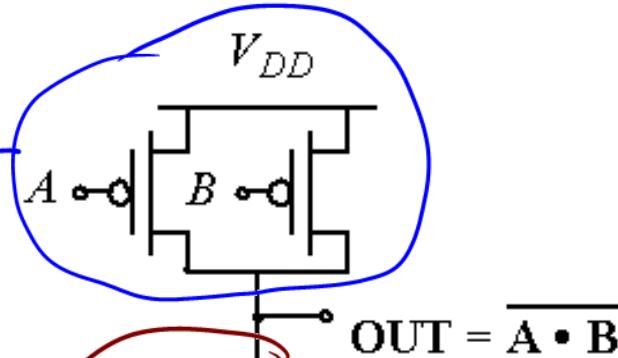
AND = NAND + INV



Example: NAND Gate

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

Truth Table of a 2 input NAND gate



- ◆ PDN: $G = AB \Rightarrow$ Conduction to GND
- ◆ PUN: $F = \overline{A} + \overline{B} = \overline{AB} \Rightarrow$ Conduction to V_{DD}

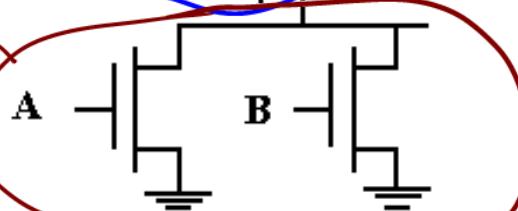
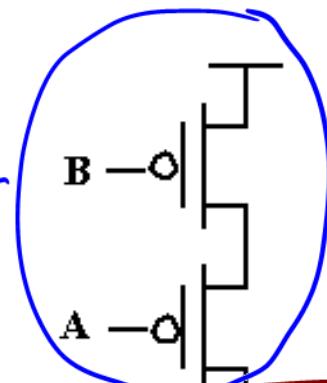
$$\overline{G(\overline{In_1}, \overline{In_2}, \overline{In_3}, \dots)} \equiv F(\overline{In_1}, \overline{In_2}, \overline{In_3}, \dots)$$

$\overline{\overline{G}} \quad \overline{V_{DD}}$

Example: NOR Gate

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

Truth Table of a 2 input NOR gate



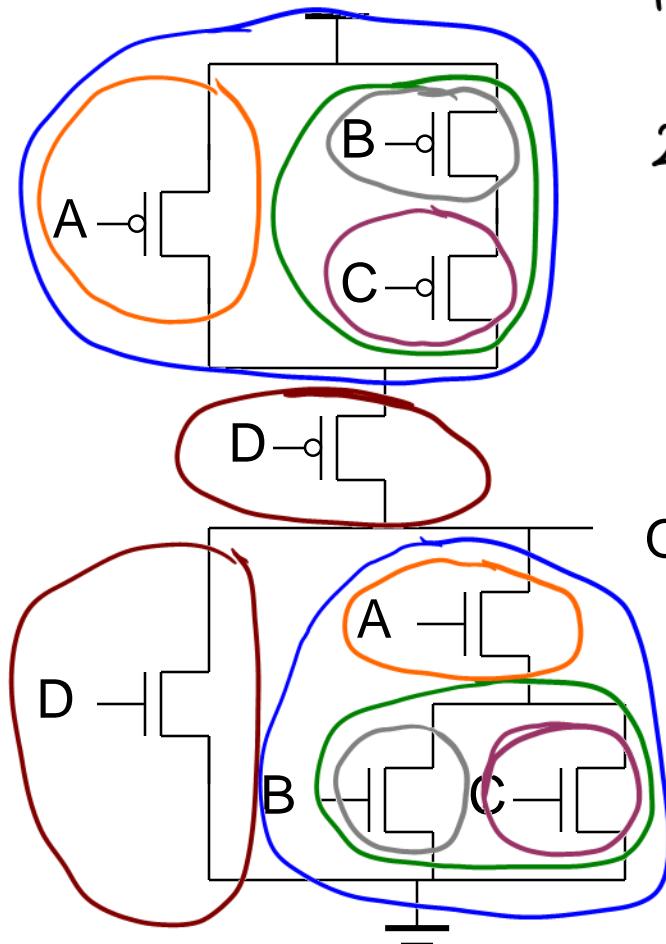
$$OUT = \overline{A+B}$$

Complex CMOS Gate

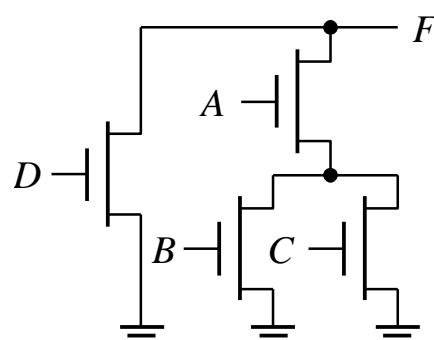
typical questions:

- 1) write logic expression based on schematic
 - 2) Construct PUN from PDN
(or vice versa)
 - parallel / series
 - sequence / order
 - stacked MOS

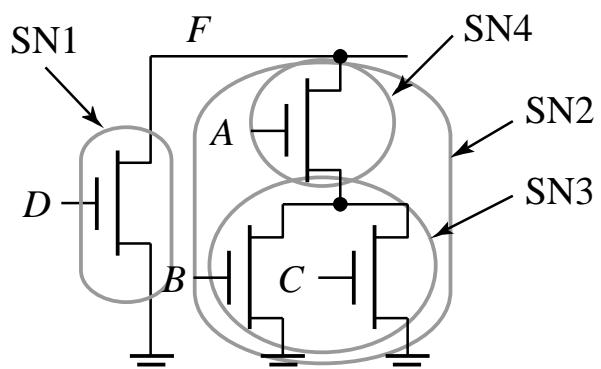
$$\text{OUT} = D + A \cdot (B + C)$$



Constructing a Complex Gate



(a) pull-down network

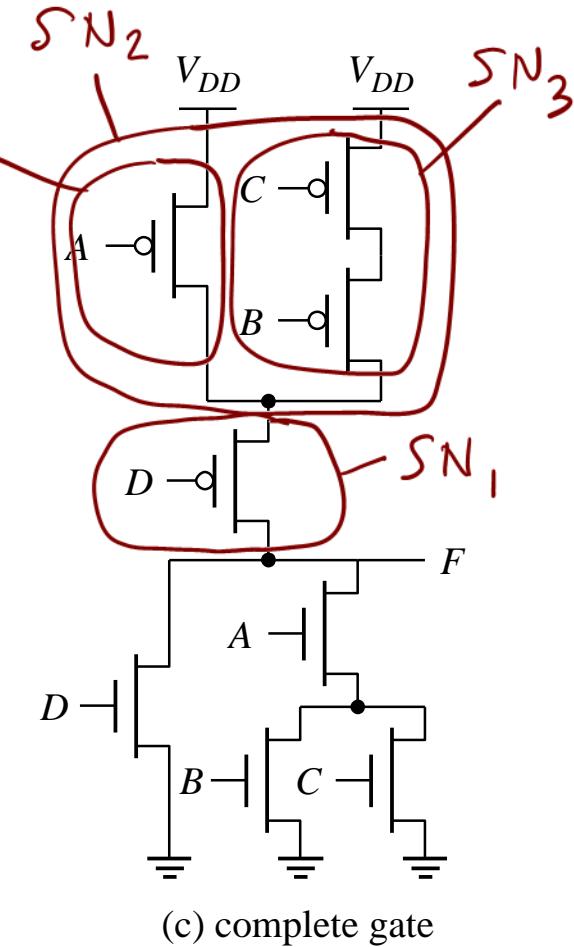


(b) Deriving the pull-up network hierarchically by identifying sub-nets

$$F = \overline{D} + A \cdot (\overline{B} + C)$$

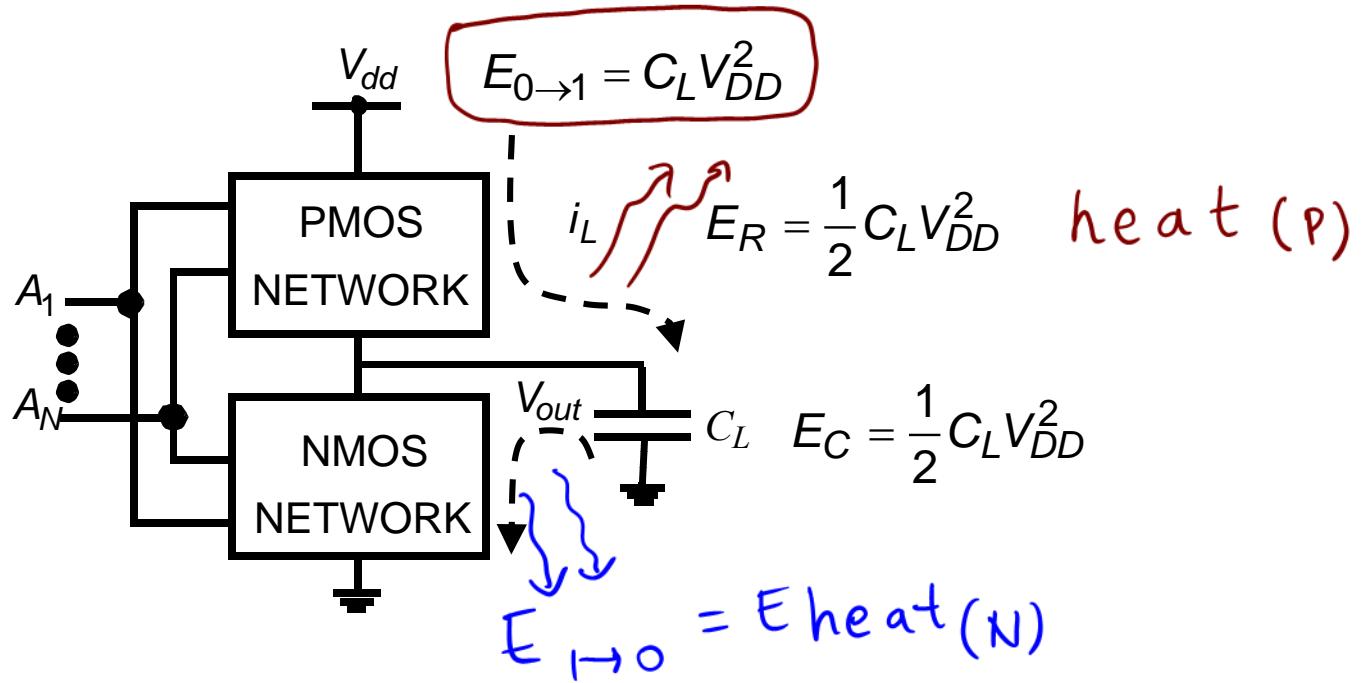
SN_1

SN_2



(c) complete gate

Dynamic Power Consumption in CMOS



- ◆ One half of the power from the supply is consumed in the pull-up network and one half is stored on C_L
- ◆ Charge from C_L is dumped during the $1 \rightarrow 0$ transition

Dynamic Power Consumption (Cont.)

Power = Energy/transition • Transition rate

$$= C_L V_{DD}^2 \cdot f_{0 \rightarrow 1}$$

$P_{0 \rightarrow 1}$: probability of $0 \rightarrow 1$ trans.
 $P_{0 \rightarrow 1} = \alpha_{0 \rightarrow 1}$

$$= C_L V_{DD}^2 \cdot f \cdot P_{0 \rightarrow 1}$$

$\alpha_{0 \rightarrow 1} \cdot f_{CLK}$

$$= C_{switched} V_{DD}^2 \cdot f$$

$C_{SW} = \alpha_{0 \rightarrow 1} \cdot C_L$

- ◆ Power dissipation is **data dependent** – depends on the switching probability
- ◆ Switched capacitance $C_{switched} = C_L \cdot P_{0 \rightarrow 1}$

Transition Activity and Power (Review)

- ◆ Energy consumed in N cycles, E_N :

$$E_N = C_L \cdot V_{DD}^2 \cdot n_{0 \rightarrow 1}$$

$n_{0 \rightarrow 1}$ – number of $0 \rightarrow 1$ transitions in N cycles

$$P_{avg} = \lim_{N \rightarrow \infty} \frac{E_N}{N} \cdot f_{clk} = \left(\lim_{N \rightarrow \infty} \frac{n_{0 \rightarrow 1}}{N} \right) \cdot \boxed{C_L \cdot V_{DD}^2} \cdot f_{clk}$$

$$\alpha_{0 \rightarrow 1} = \lim_{N \rightarrow \infty} \frac{n_{0 \rightarrow 1}}{N}$$

$$P_{avg} = \alpha_{0 \rightarrow 1} \cdot C_L \cdot V_{DD}^2 \cdot f_{clk}$$

Factors Affecting Switching Activity

- ◆ “Static” component (does not account for timing)
 - Type of logic function (NOR vs. XOR)
 - Type of logic style (Static vs. Dynamic)
 - Signal statistics
 - Inter-signal correlations

- ◆ “Dynamic” or timing dependent component
 - Circuit topology
 - Signal statistics and correlations

Type of Logic Function: NOR vs. XOR

Example: Static 2-input NOR Gate

$$P_{out=0} = 1 - \frac{1}{4} = \frac{3}{4}$$

$$P_{out=1} = P_{A=0} \times P_{B=0} \\ = (1 - 1/2) \times (1 - 1/2) = 1/4$$

A	B	Out
0	0	1
0	1	0 1/4
1	0	0 1/4
1	1	0 1/4

$$P_{out=0} = \frac{1}{4} + \frac{1}{4} + \frac{1}{4} = 3/4$$

If inputs switch every cycle

$$\neg a_{0 \rightarrow 1} = 3/16$$

Assume signal probabilities

$$p_{A=1} = 1/2$$

$$p_{B=1} = 1/2$$

? Then transition probability

$$p_{0 \rightarrow 1} = p_{out=0} \times p_{out=1} \\ = 3/4 \times 1/4 = 3/16$$

$$P_{X=0} + P_{X=1} = 1 \quad (1)$$

$$P_{X:0 \rightarrow 1} = P_{X=0} \times P_{X=1} \quad (2)$$

Type of Logic Function: NOR vs. XOR

Example: Static 2-input XOR Gate

A	B	Out
0	0	0 $\frac{1}{4}$
0	1	1
1	0	1
1	1	0 $\frac{1}{4}$

Assume signal probabilities

$$p_{A=1} = 1/2$$

$$p_{B=1} = 1/2$$

? Then transition probability

$$\begin{aligned} p_{0 \rightarrow 1} &= p_{\text{Out}=0} \times p_{\text{Out}=1} \\ &= 1/2 \times 1/2 = 1/4 \end{aligned}$$

If inputs switch in every cycle

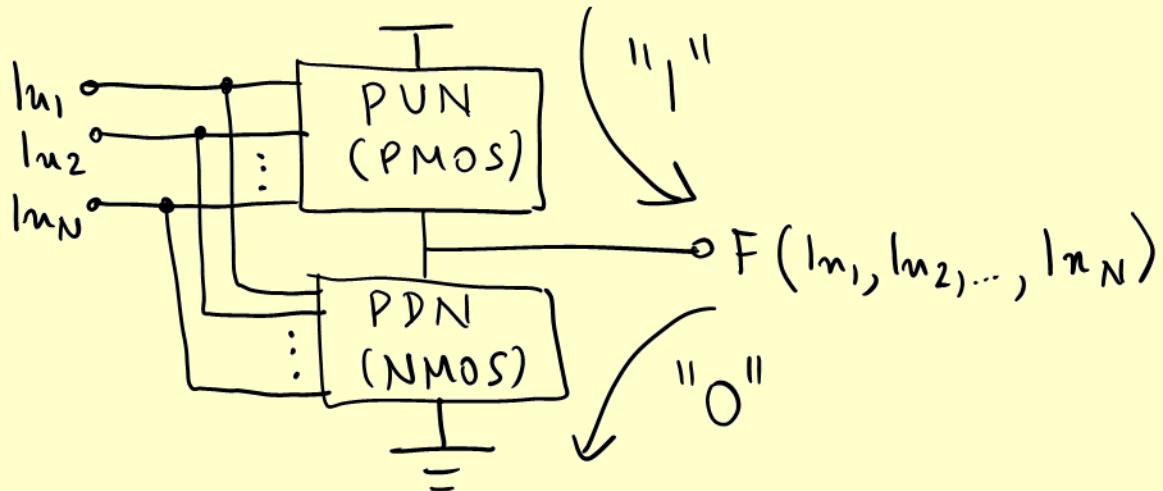
$$a_{0 \rightarrow 1} = 1/4$$

$$\begin{array}{c} \text{XOR} \\ = \frac{4}{16} > \frac{3}{16} \\ \text{NOR} \end{array}$$

Principles for Power Reduction

- ◆ Prime choice: Reduce voltage! ✓
 - Recent years have seen an acceleration in supply voltage reduction
 - Design at very low voltages still open question (0.6 ... 0.9 V by 2010!)
 - Reducing thresholds to improve performance increases leakage
- ◆ Reduce switching activity
- ◆ Reduce physical capacitance

Review of CMOS Logic



Duality

PDN is dual to PUN

$$\overline{A+B} (P) \leftrightarrow \overline{\bar{A} \cdot \bar{B}} (S)$$

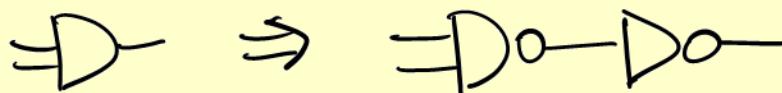
$$\overline{A \cdot B} (S) \leftrightarrow \overline{\bar{A} + \bar{B}} (P)$$

NMOS

PMOS

(P) parallel, (S) series

- N-input CMOS gate requires $2N$ transistors
- Only inverting functions are implemented in each stage



Review of Switching Activity



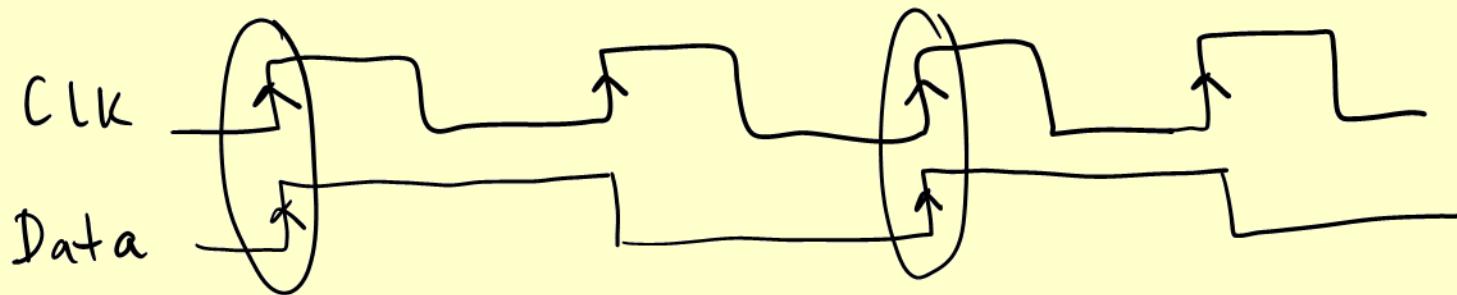
✓ given $P(A=1), P(B=1)$
? calculate $\alpha_{0 \rightarrow 1}$ (out : 0 → 1)

$$P_{X=0} = 1 - P_{X=1}$$

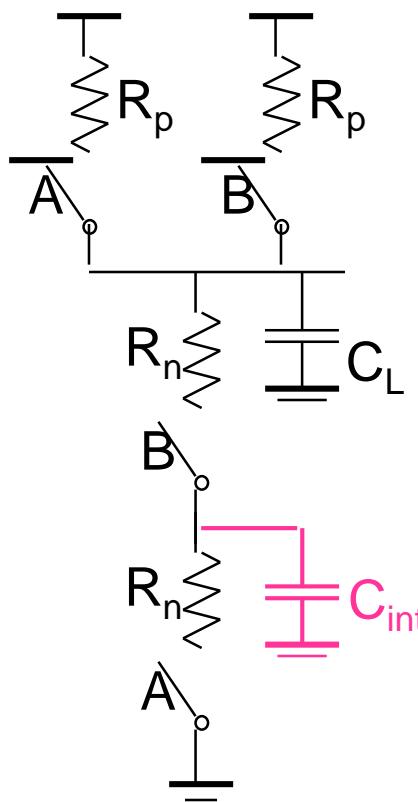
$$P_{X:0 \rightarrow 1} = P_{X=0} \times P_{X=1}$$

$$\alpha_{0 \rightarrow 1}^{\max} = \frac{1}{4} \text{ for XOR } P_A=1 = P_B=1 = \frac{1}{2}$$

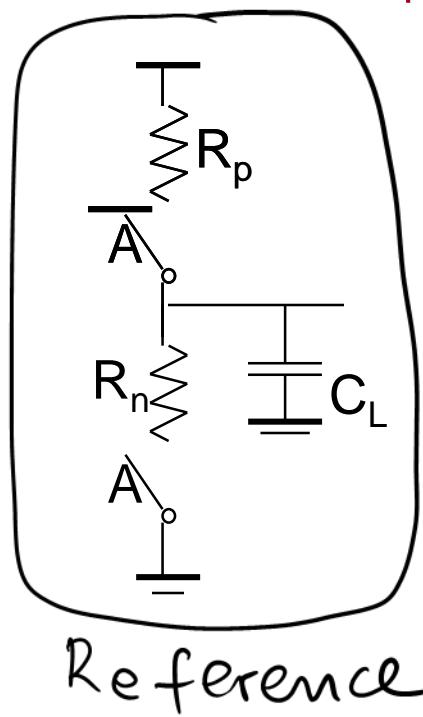
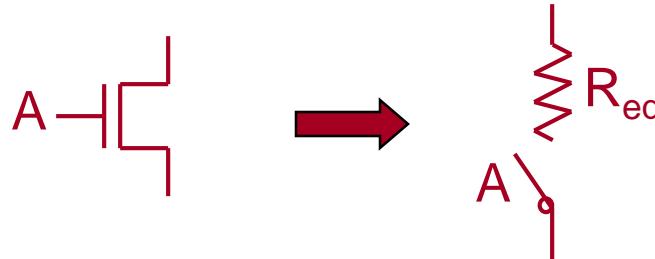
$$\alpha_{\text{clk}} = 1$$
$$\alpha_{\text{Data}}^{\max} = \frac{1}{2}$$



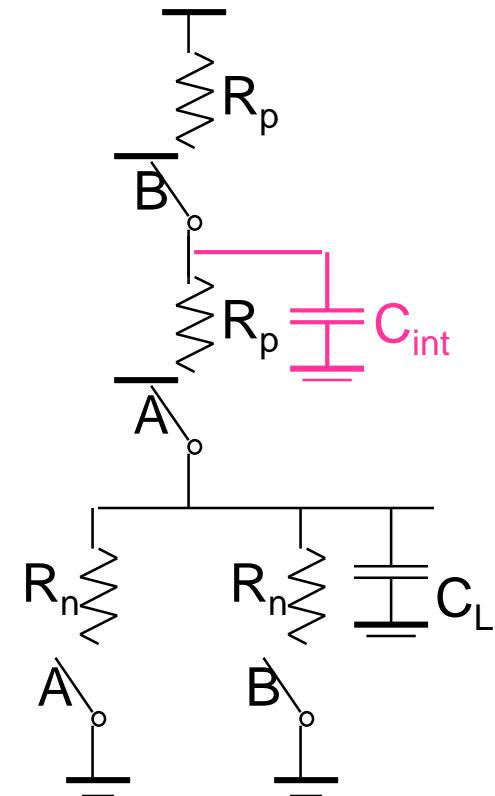
Switch Delay Model



NAND-2

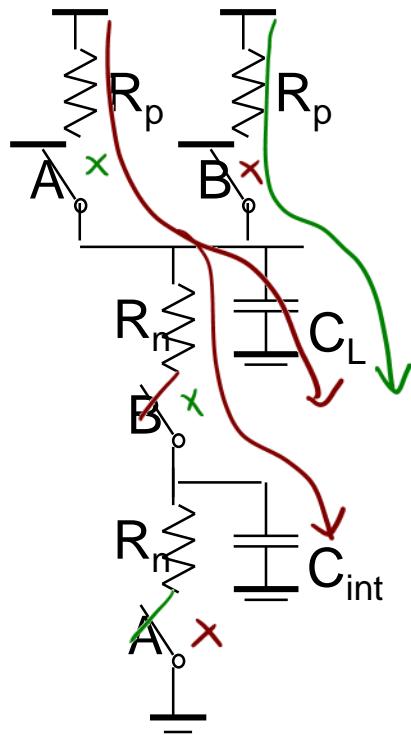


Inverter



NOR-2

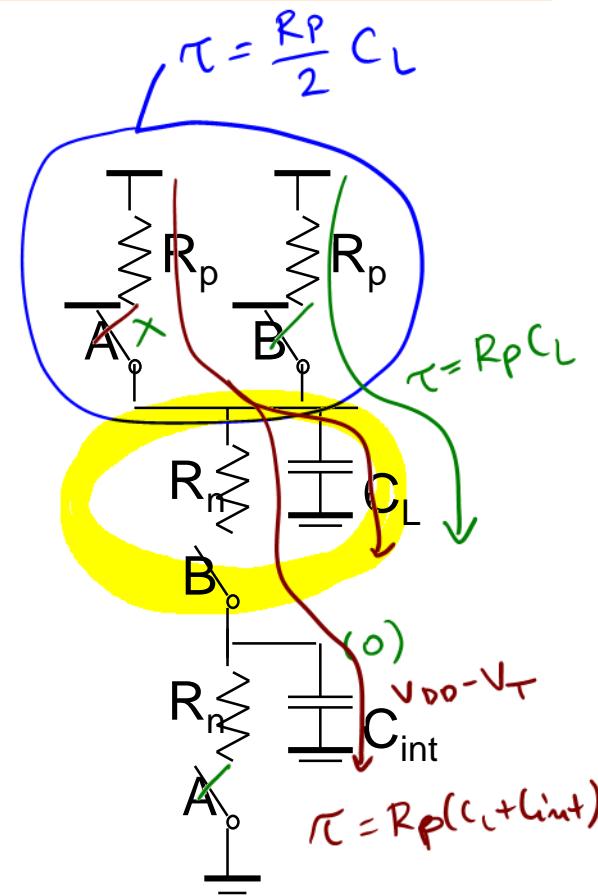
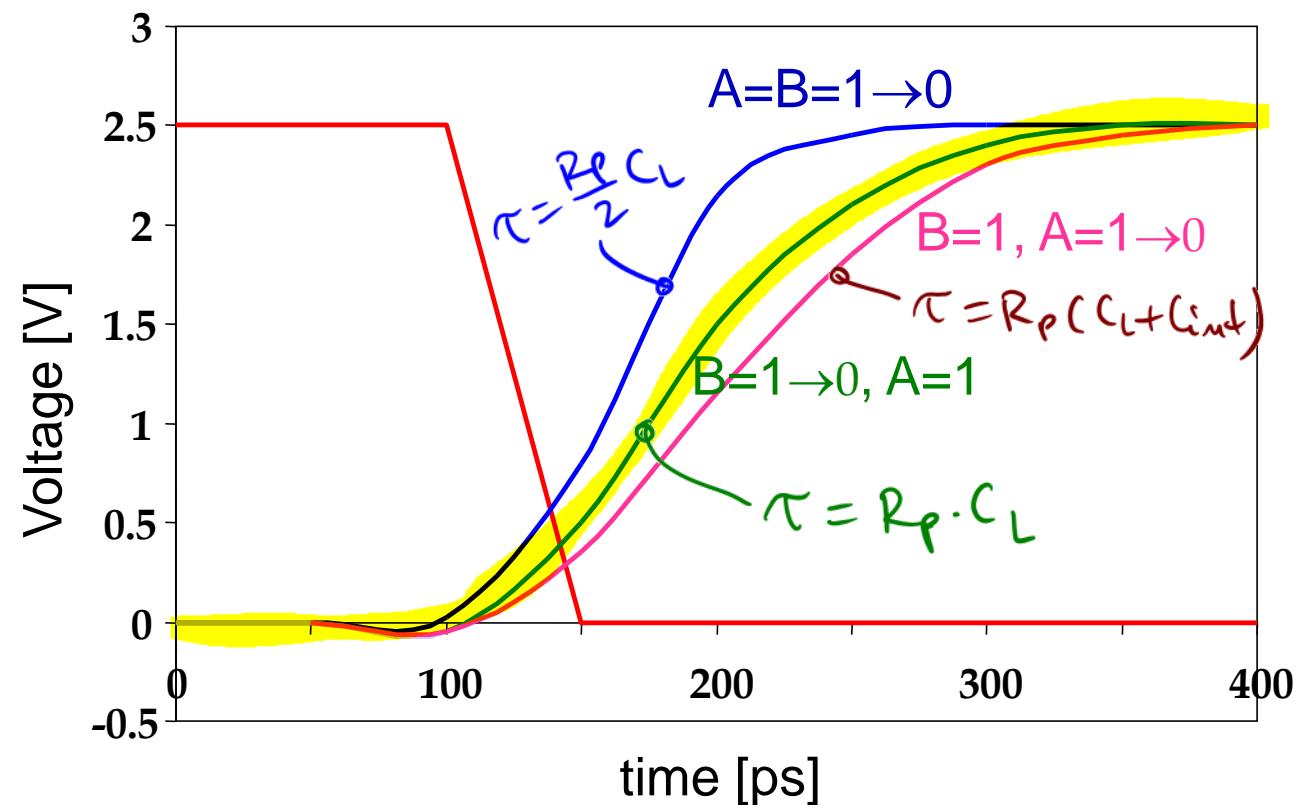
Input Pattern Effects on Delay



- ◆ Delay is dependent on the **pattern** of inputs
- ◆ Low-to-high transition
 - both inputs go low
 - delay is $0.69 R_p / 2 C_L$
 - one input goes low
 - delay is $0.69 R_p C_L$
- ◆ High-to-low transition
 - both inputs go high
 - delay is $0.69 2R_n C_L$

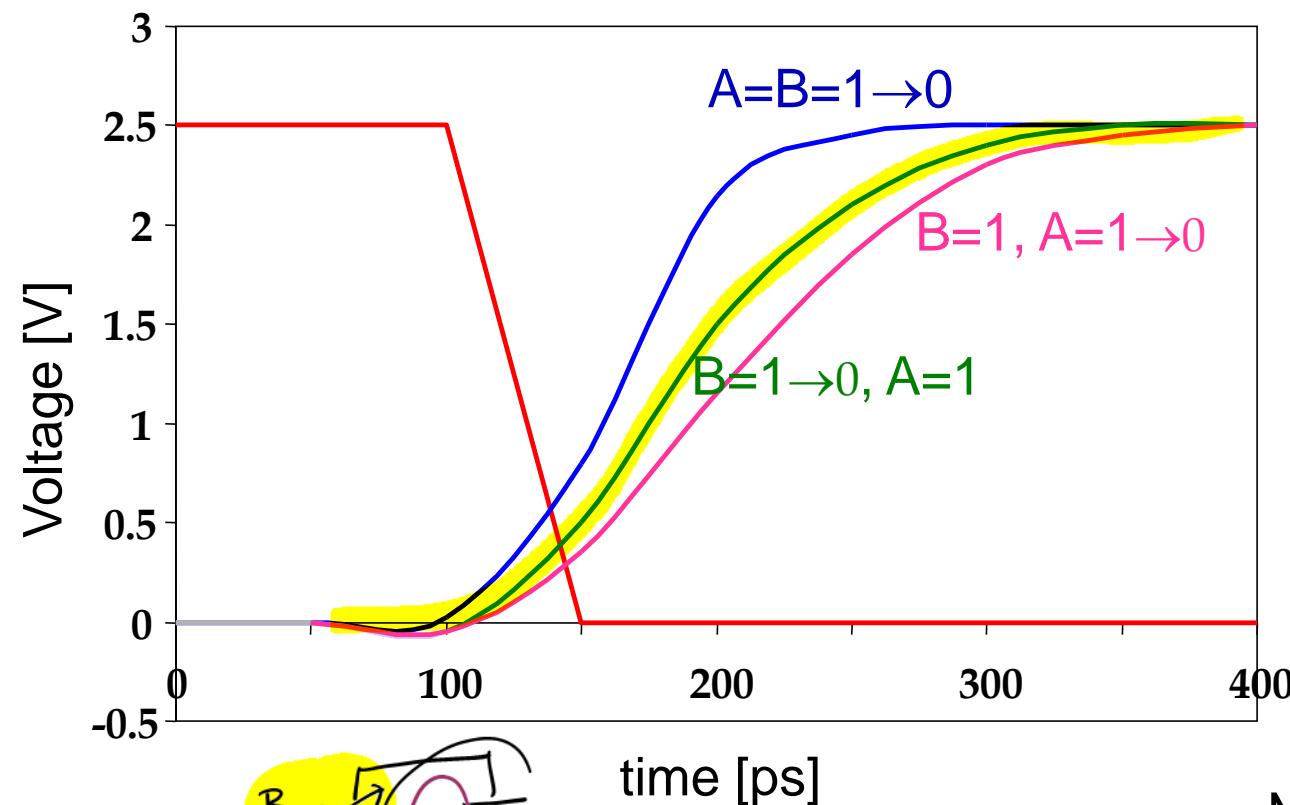


Delay Dependence on Input Patterns

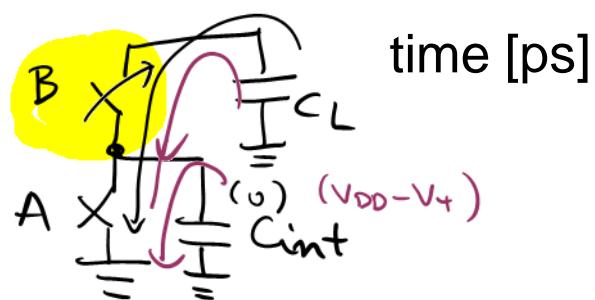


Last input to change should be wired to the top of the stack (array from $\frac{1}{2}$ or V_{DD})

Delay Dependence on Input Patterns



Input Data Pattern	Delay (ps)
$A=B=0 \rightarrow 1$	69
$B=1, A=0 \rightarrow 1$ $\tau = R_L C_{int} + 2 R_L C_L$	62
$B=0 \rightarrow 1, A=1$	50
$A=B=1 \rightarrow 0$	35
$B=1, A=1 \rightarrow 0$	76
$B=1 \rightarrow 0, A=1$	57

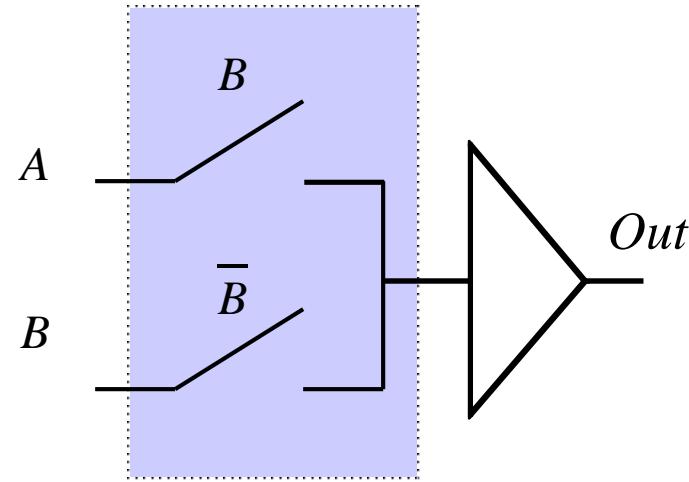
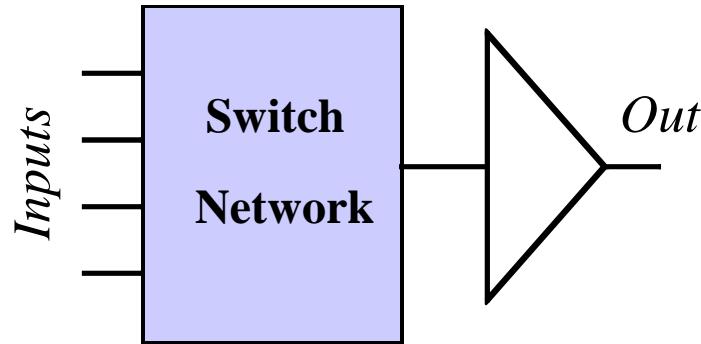


NMOS = $0.5\mu\text{m}/0.25\mu\text{m}$
PMOS = $0.75\mu\text{m}/0.25\mu\text{m}$
 $C_L = 100 \text{ fF}$

Week 3 Agenda

- ◆ CMOS Logic
- ◆ Pass-Transistor Logic
- ◆ Sizing of Logic Gates

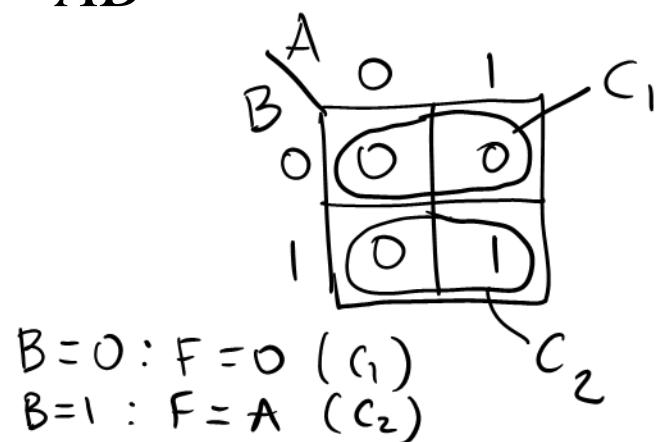
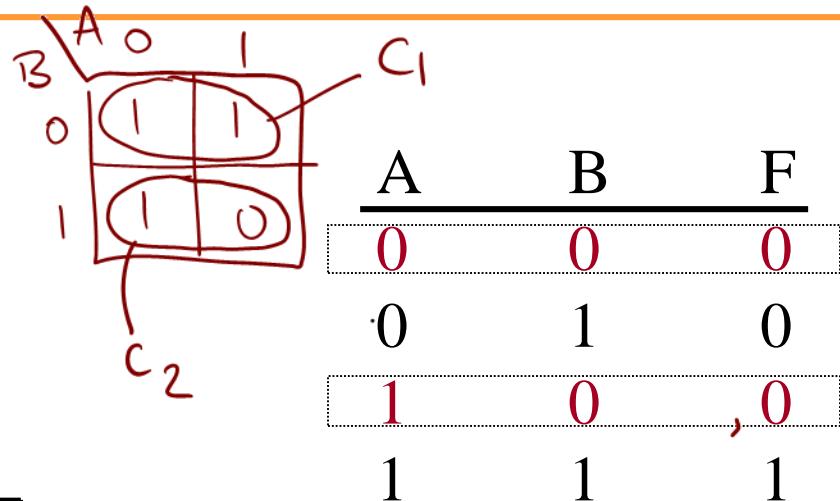
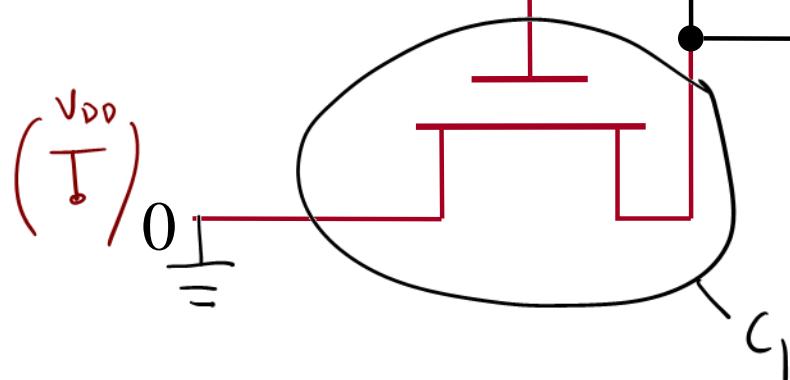
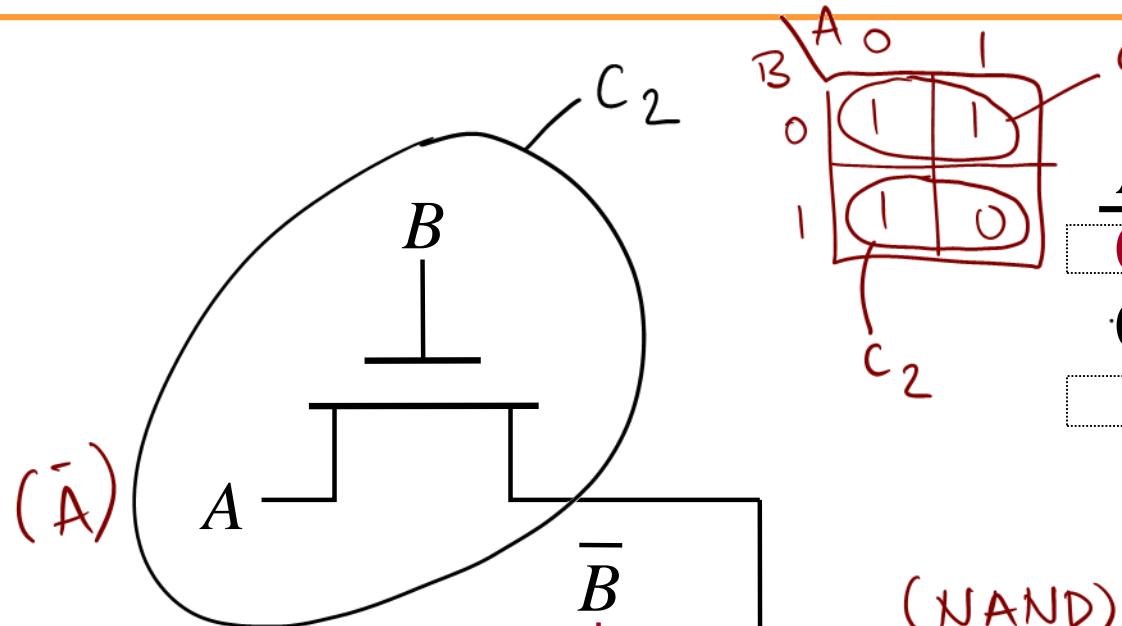
Pass Transistor Logic (PTL)



- ◆ N transistors vs. $2N$ transistors in CMOS
- ◆ No static power consumption

Allows primary inputs to drive S and D terminals!
(idea: reduce the number of transistors)

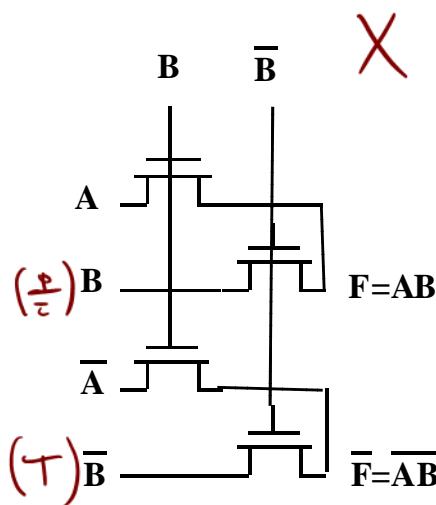
Example: AND Gate



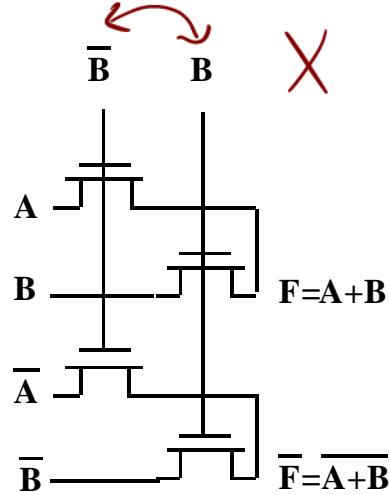
Complementary Pass Transistor Logic (CPL)



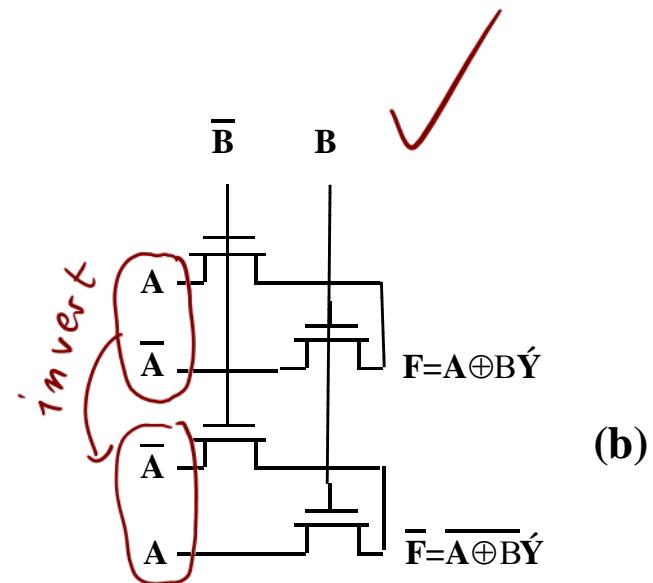
(a) Same topology & control signals
inverted inputs



AND/NAND

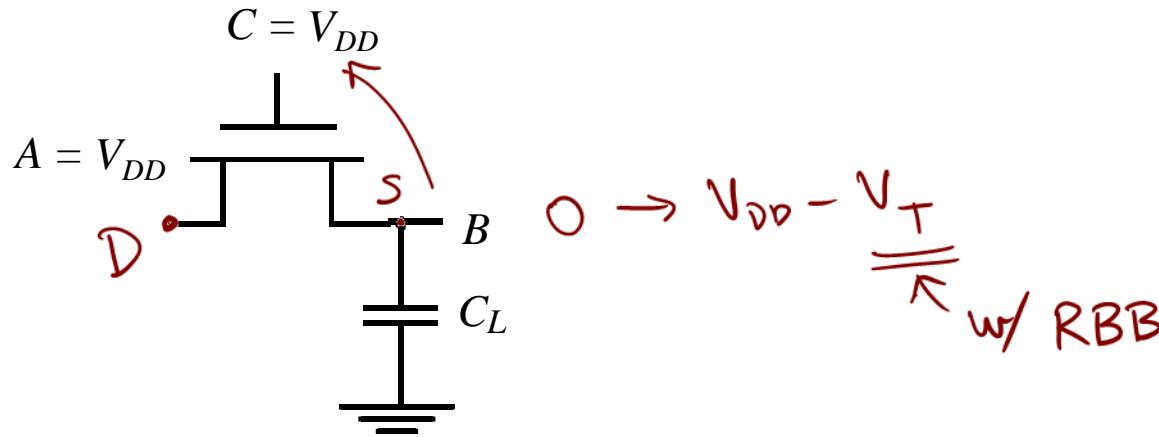


OR/NOR



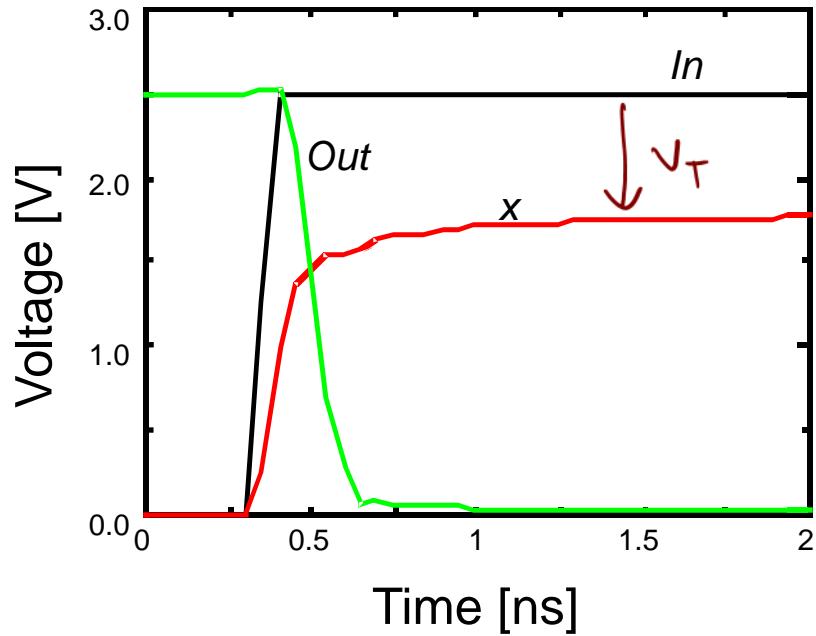
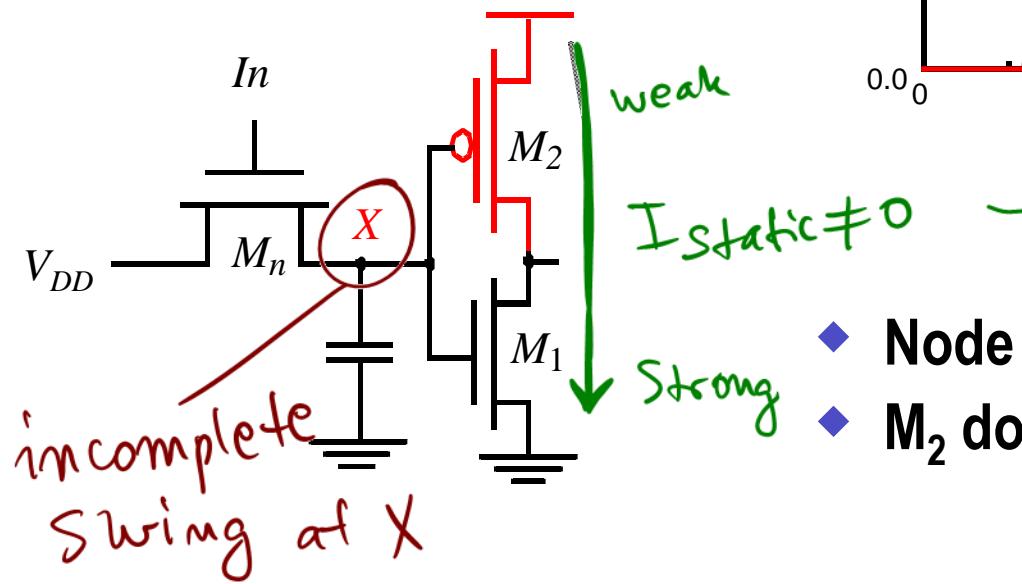
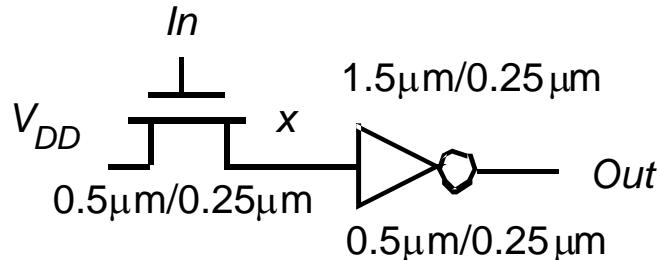
EXOR/NEXOR

NMOS-only Switch



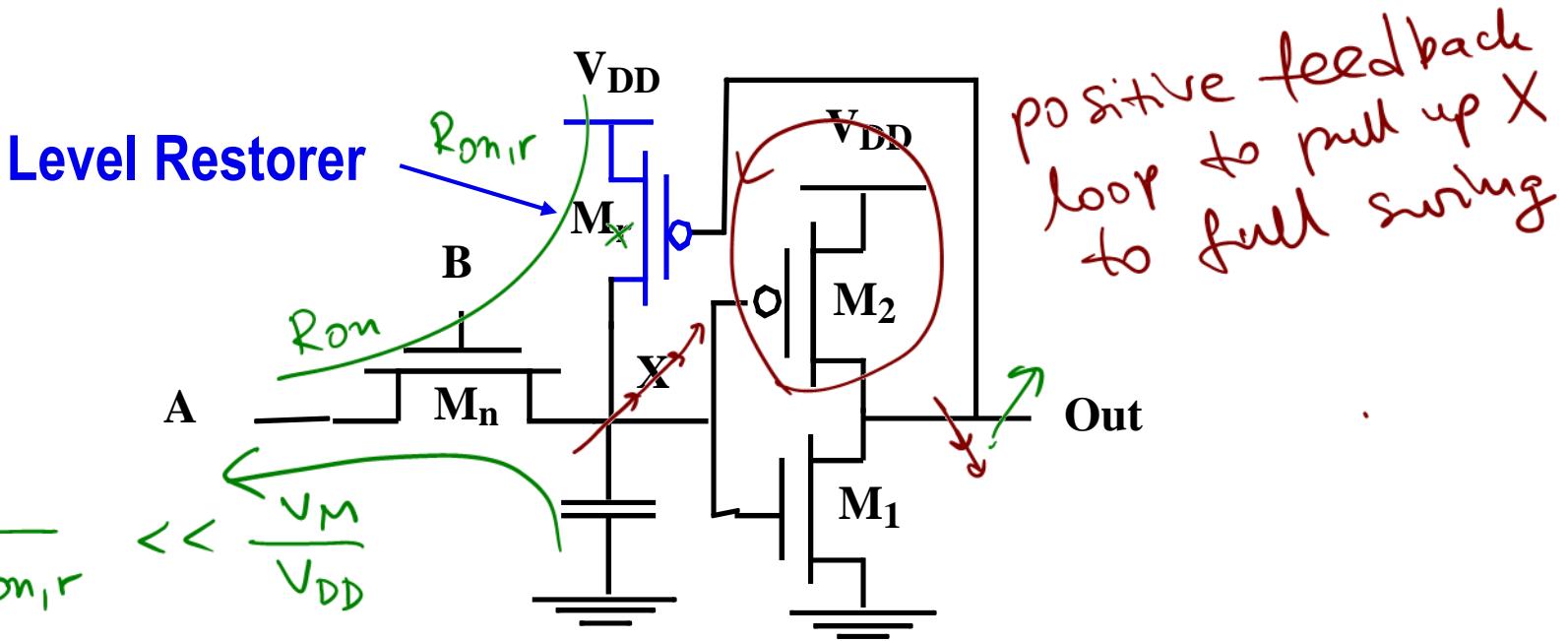
- ◆ **V_B does not pull up to V_{DD} , but to $V_{DD} - V_{Tn}$**
 - Threshold voltage loss causes static power consumption
 - NMOS has higher threshold than PMOS (body effect)

NMOS-Only Logic



- ◆ Node X is not full-swing
- ◆ M_2 does not fully switch off

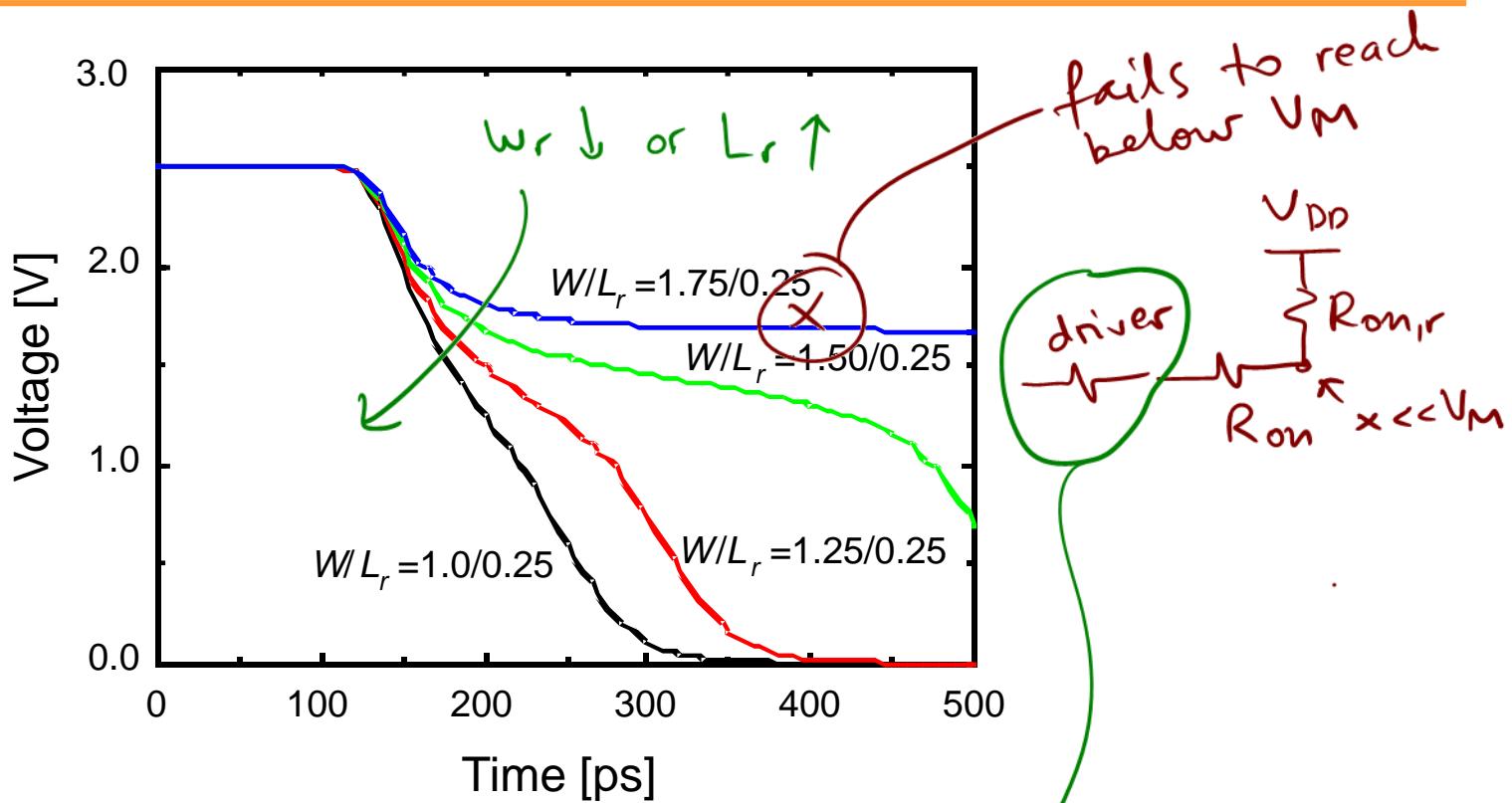
Solution #1: Level Restoring Transistor



- ◆ Advantage: Full swing
- ◆ Restorer adds capacitance, takes away PDN current at X
- ◆ **Ratio problem**

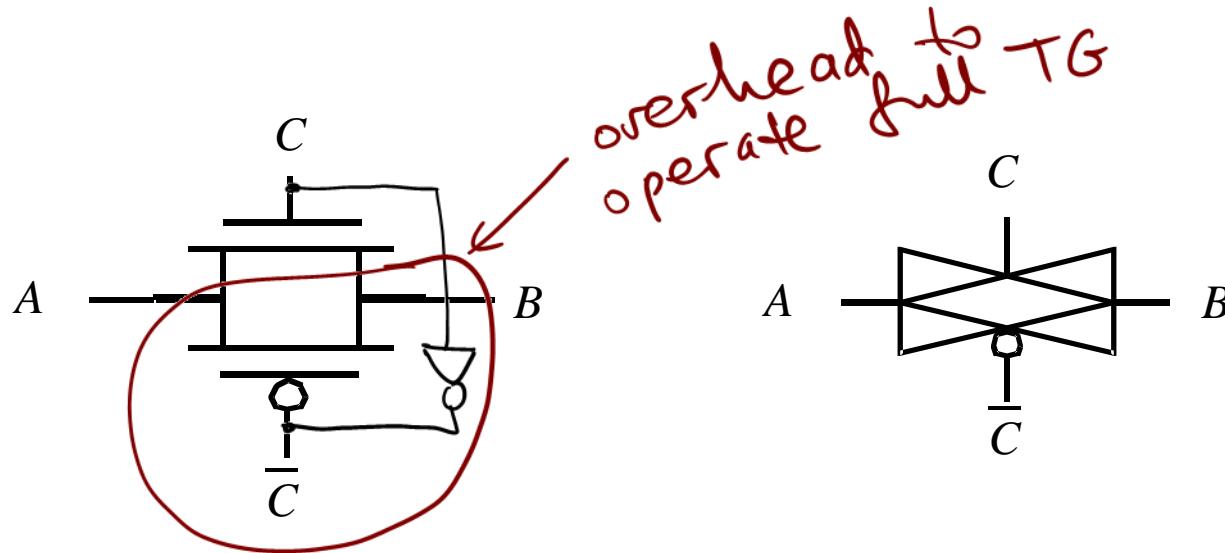
make $R_{on,r} \gg R_{on}$ (make L_r long & min W_r)

Restorer Sizing



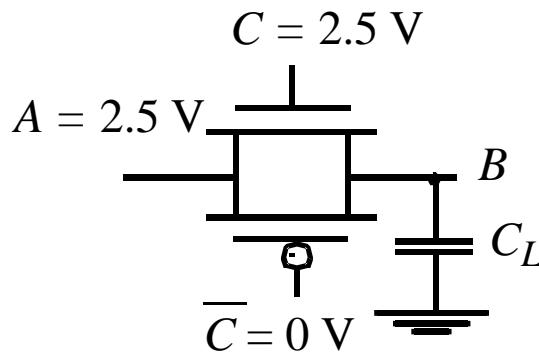
- Upper limit on restorer size
- Pass-transistor PDN can have several transistors in stack

Solution #2: Transmission Gate



Cons:

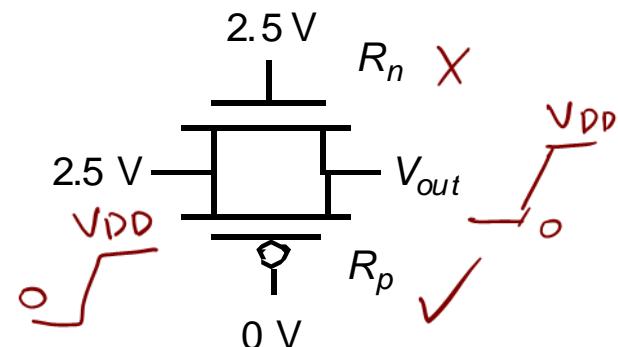
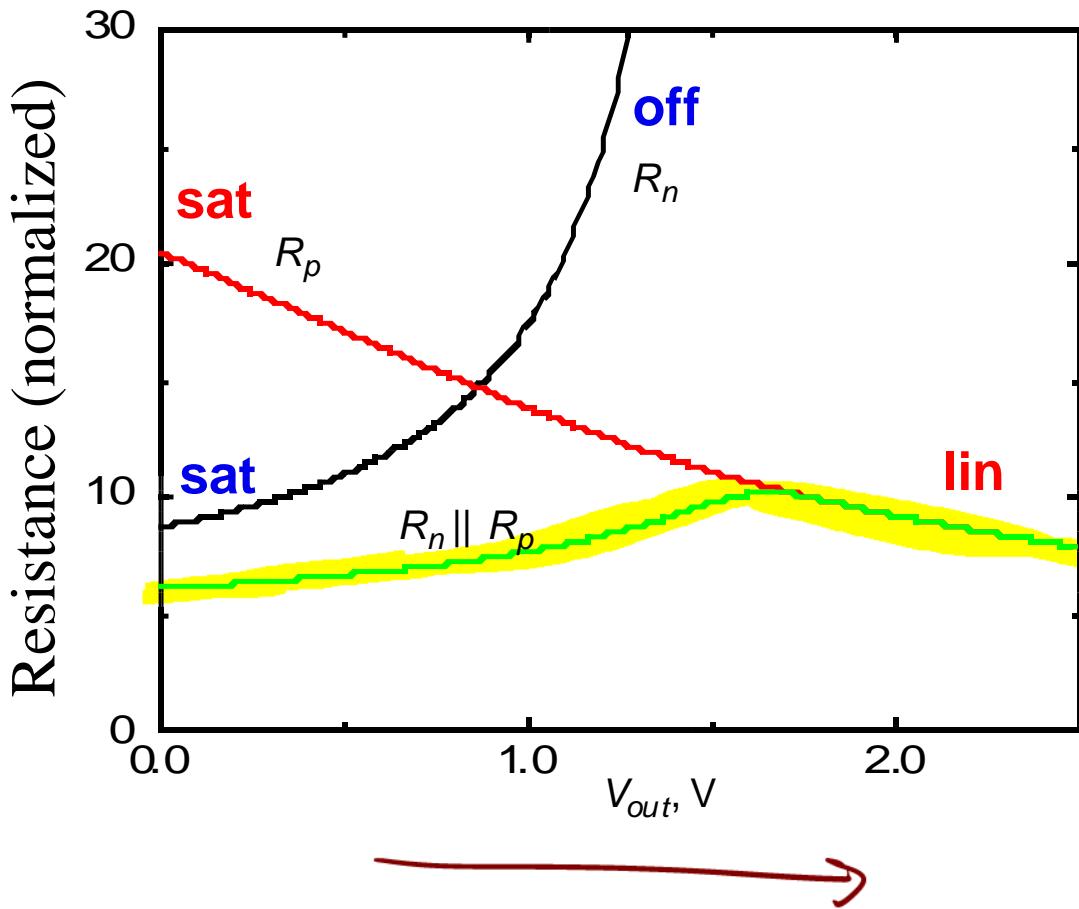
- Requires two transistors
- More control signals needed



Pros:

- Bidirectional switch
- Rail-to-rail switching

Resistance of Transmission Gate

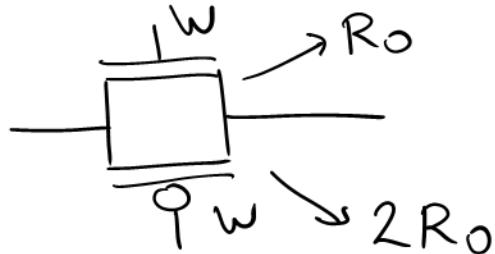


$V_{out}: 0 \rightarrow 1$

Pass Transistor Sizing

◆ Equal PMOS and NMOS

- Compact layout, approximately equal R_{PUP} and R_{PDN}



$$R_{on} = R_p \parallel R_n$$

- if a transistor supports "weak" transition (N:up, P:dn), apply penalty (e.g. 2x) on its resistance

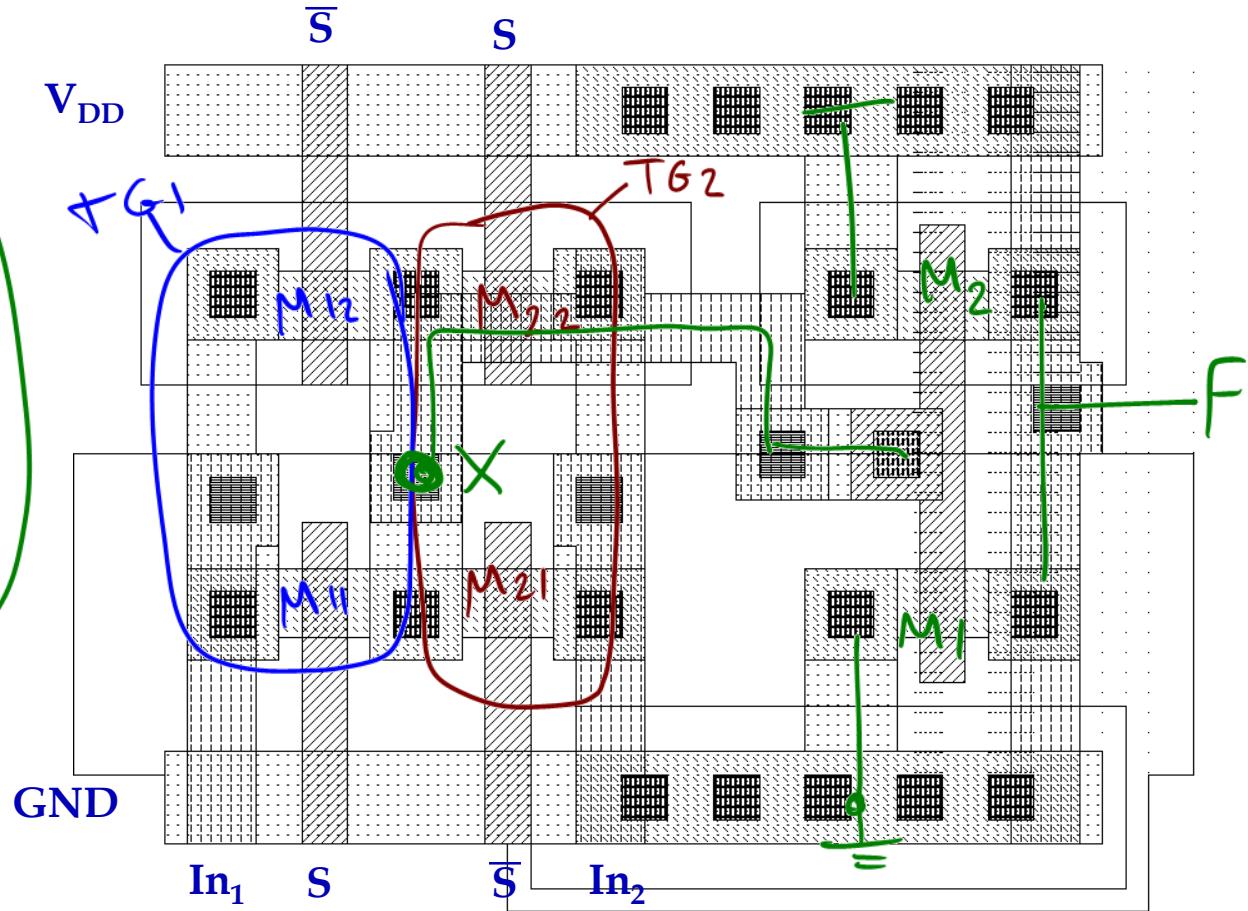
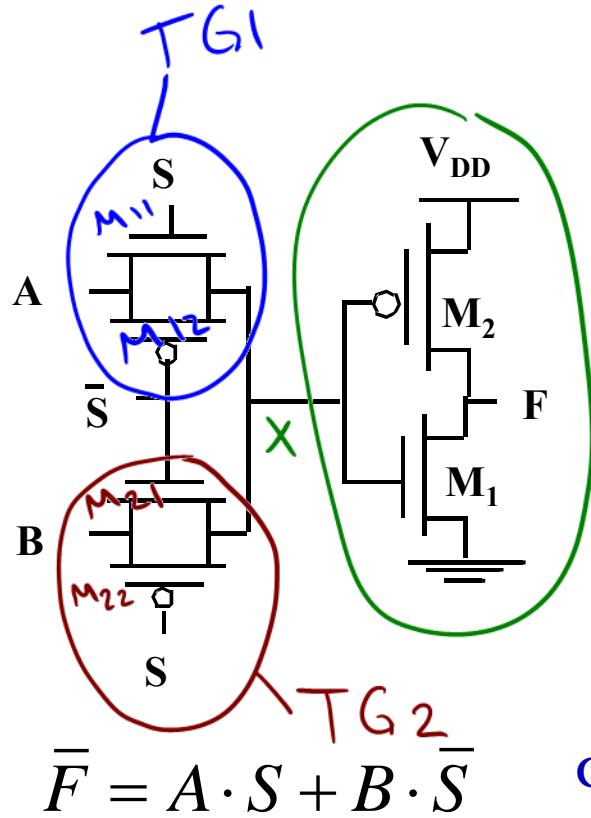
$$PUP: 2R_o \parallel 2R_o = R_o$$

$$\Rightarrow R_{on} \approx R_o$$

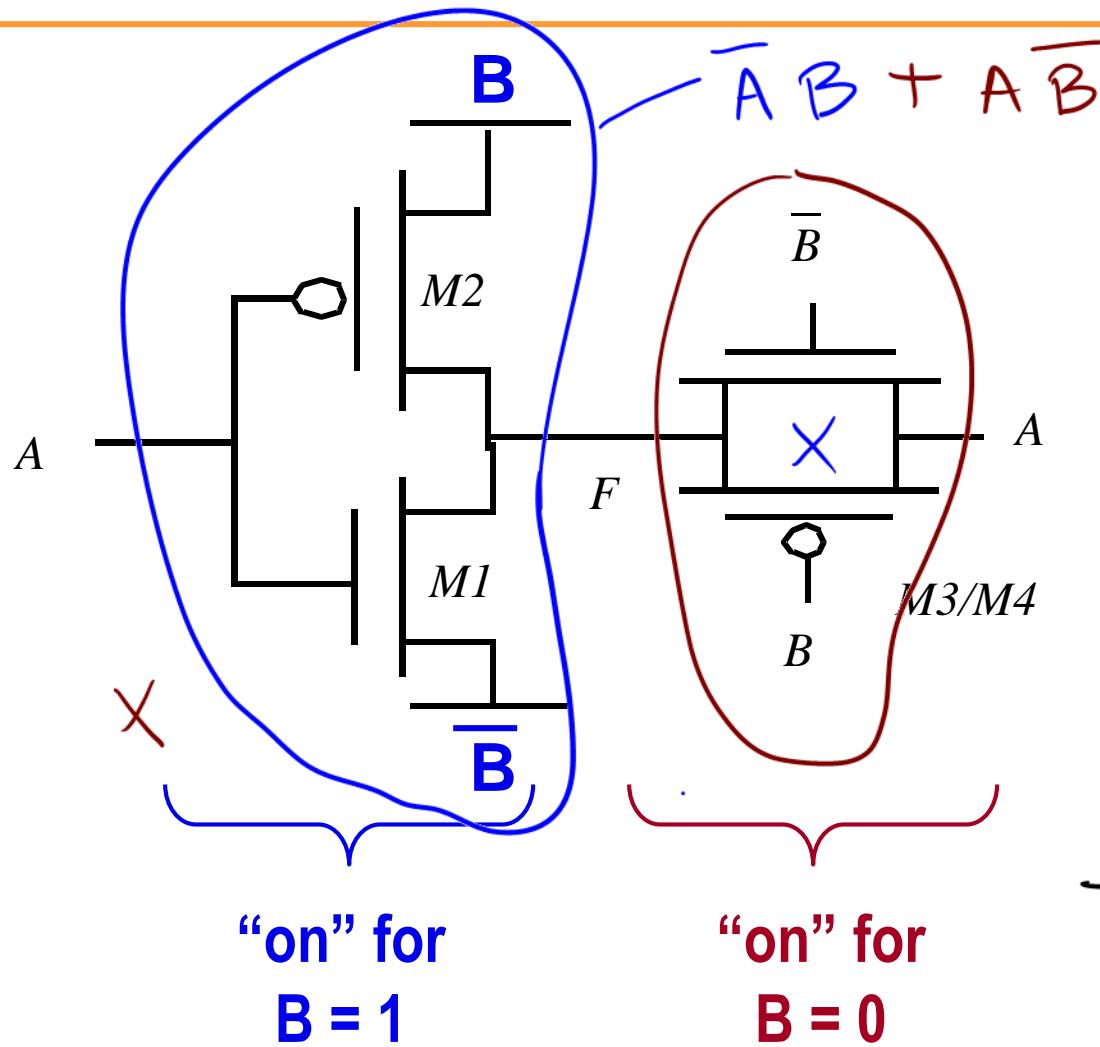
$$PDN: 2 \cdot 2R_o \parallel R_o = 0.8R_o$$

2 is the "penalty" factor used here

Pass Transistor based Multiplexer



Transmission Gate XOR

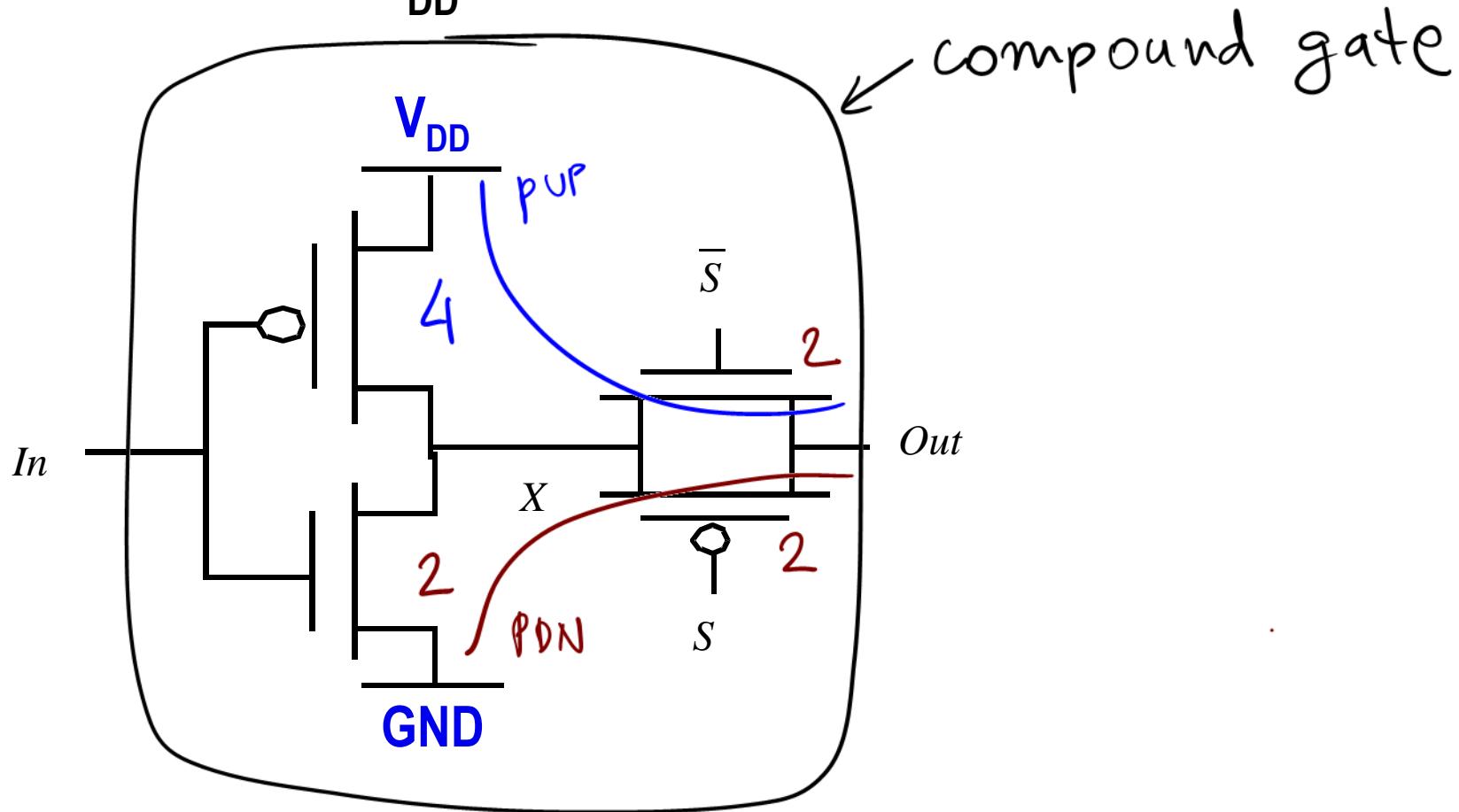


- $B \rightarrow \overline{B}$ including
- 6 transistors only
 - 12 transistors in CMOS

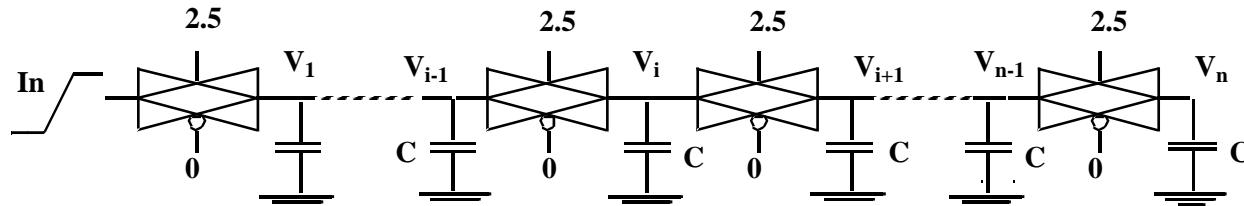
wired OR method
→ one section of the circuit is ON depending on B

PTL Gate is a Compound Gate

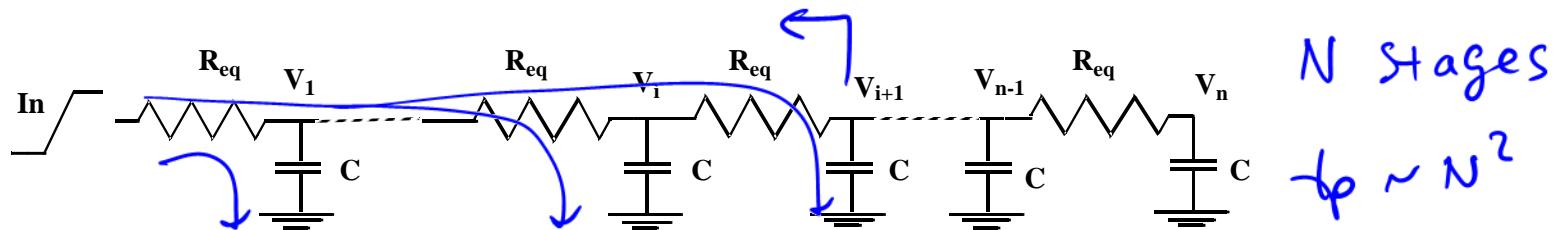
- ◆ Back-trace to V_{DD} or GND



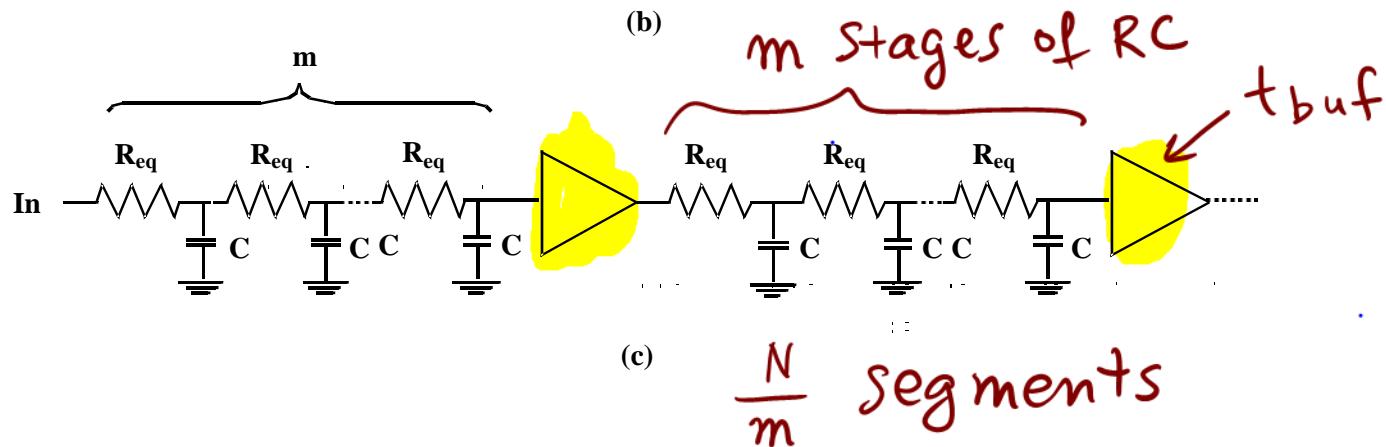
Delay in Transmission Gate Networks



(a)



N Stages
 $\rightarrow \sim N^2$



Delay Optimization

- ◆ Delay of RC chain

$$t_p = \sum_{k=0}^n CR_{eq} k = 0.69 CR_{eq} \frac{n(n+1)}{2}$$

- ◆ Delay of buffered chain

$$t_p = 0.69 \left\lfloor \frac{n}{m} CR_{eq} \frac{m(m+1)}{2} \right\rfloor + \left(\frac{n}{m} - \cancel{m} \right) t_{buf}$$

$$= 0.69 \left\lfloor CR_{eq} \frac{n(m+1)}{2} \right\rfloor + \left(\frac{n}{m} - \cancel{m} \right) t_{buf}$$

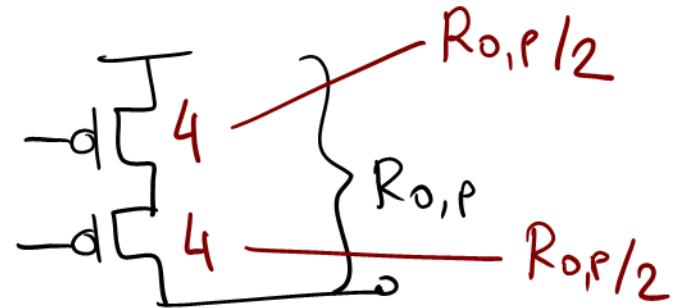
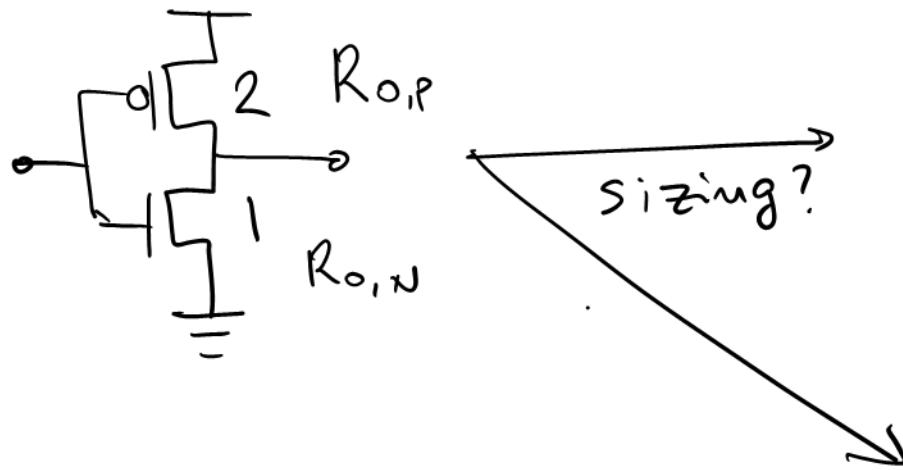
$$\frac{\partial t_p}{\partial m} \Rightarrow \boxed{m_{opt} = 1.7 \sqrt{\frac{t_{buf}}{CR_{eq}}}} \Rightarrow \underline{\underline{m \leq 3}}$$

Week 3 Agenda

- ◆ CMOS Logic
- ◆ Pass-Transistor Logic
- ◆ Sizing of Logic Gates

Transistor Sizing: Reference Inverter

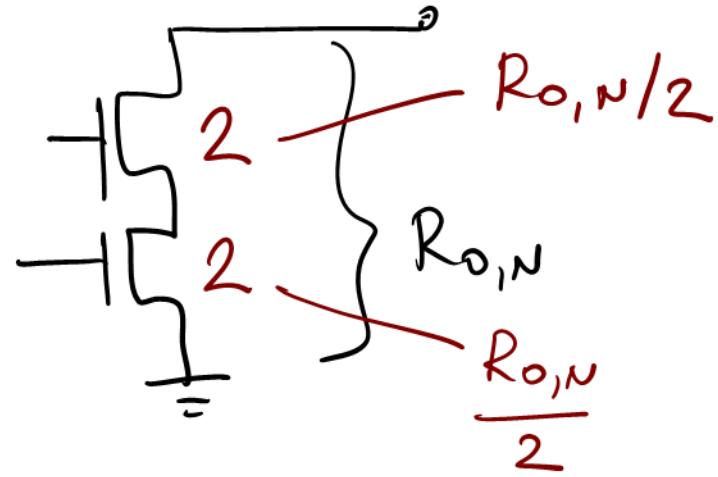
N-Stack \Rightarrow increase W by N for same total R_o



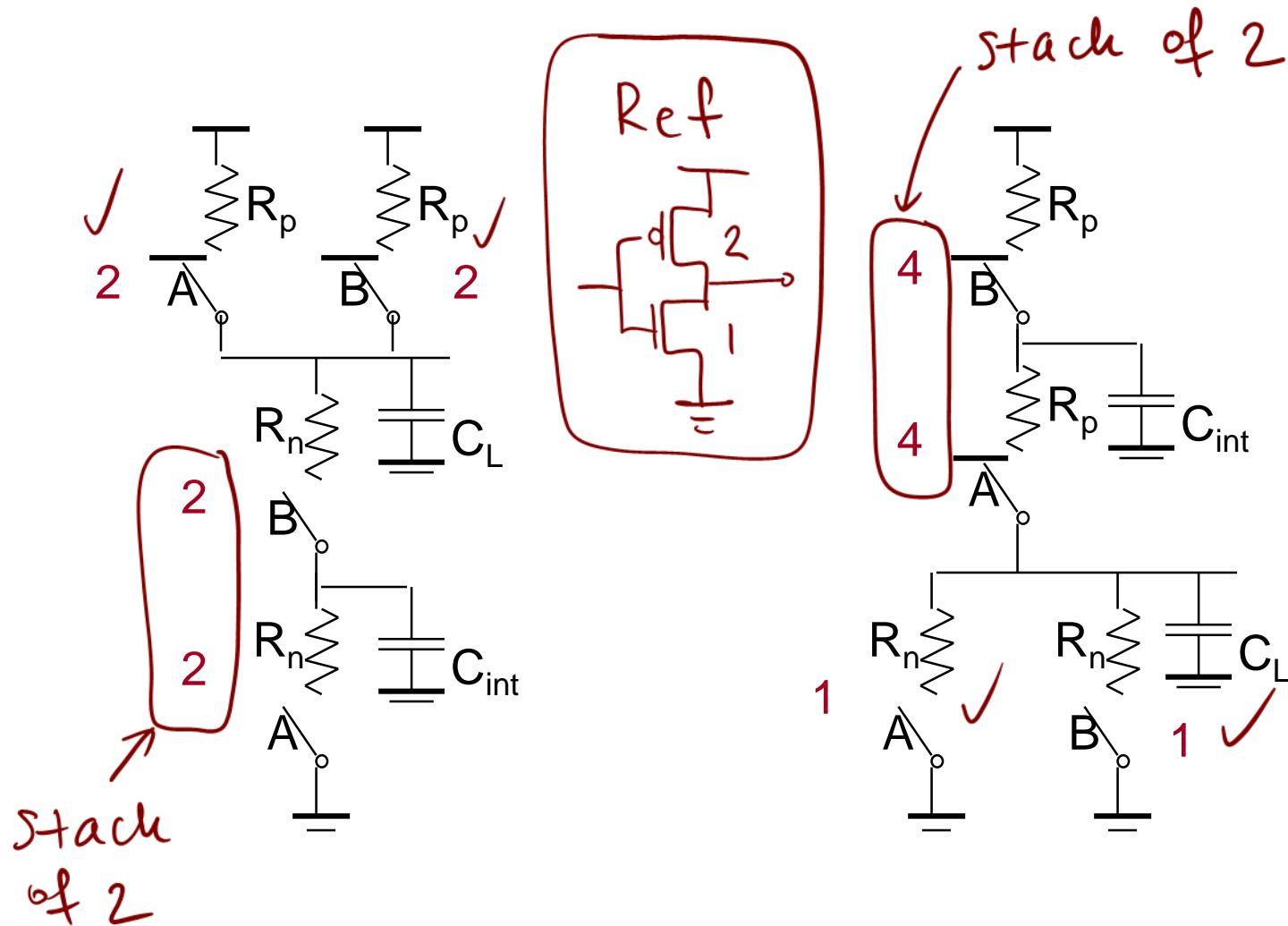
Sizing for equal R_o

$$R \sim \frac{1}{W} \quad R_o \sim \frac{1}{W_0}$$

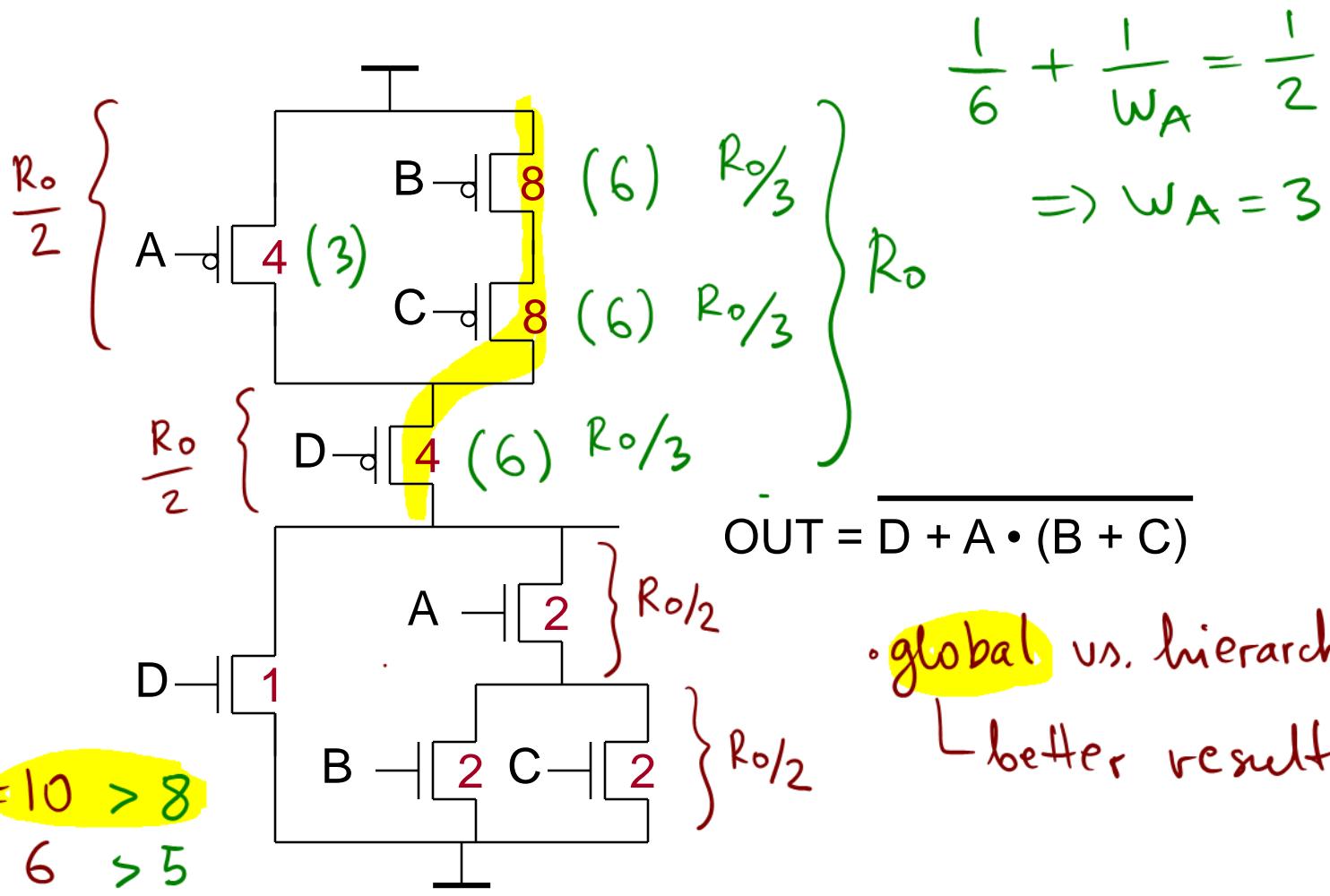
$$R = R_o \frac{W_0}{W} \quad W = 2W_0$$



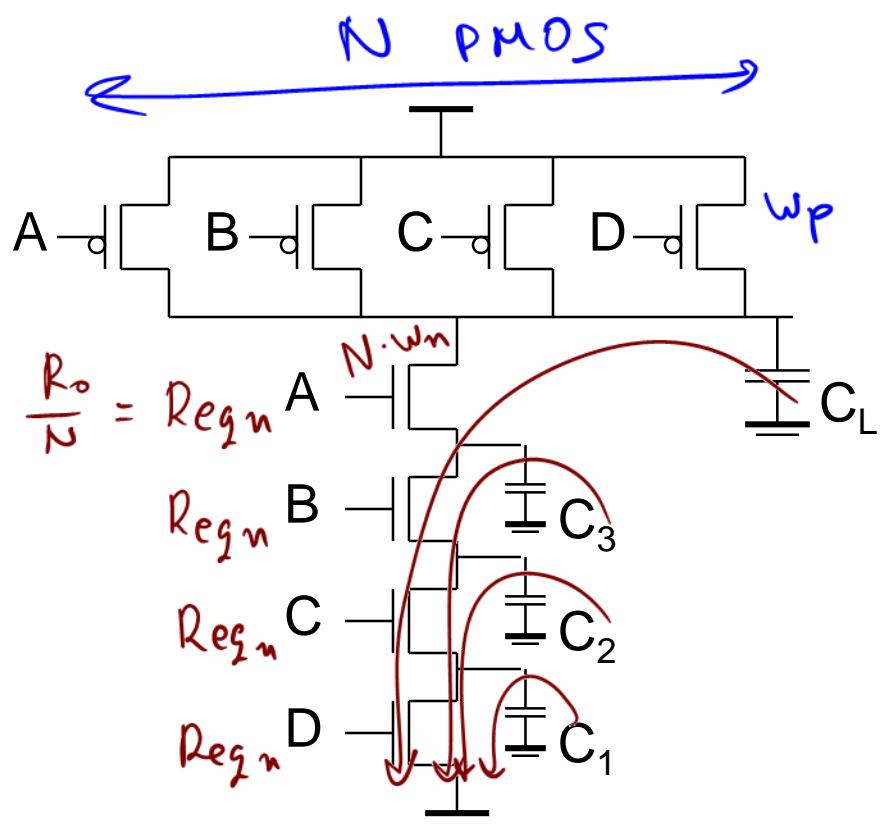
Transistor Sizing



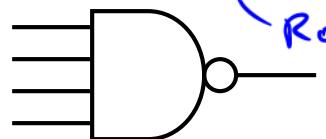
Sizing of a Complex CMOS Gate



Fan-In Considerations



$$t_{PLH} = 0.69 R_{eqn} (N C_{par} + C_L) \underset{R_{o,P}}{\sim} N$$

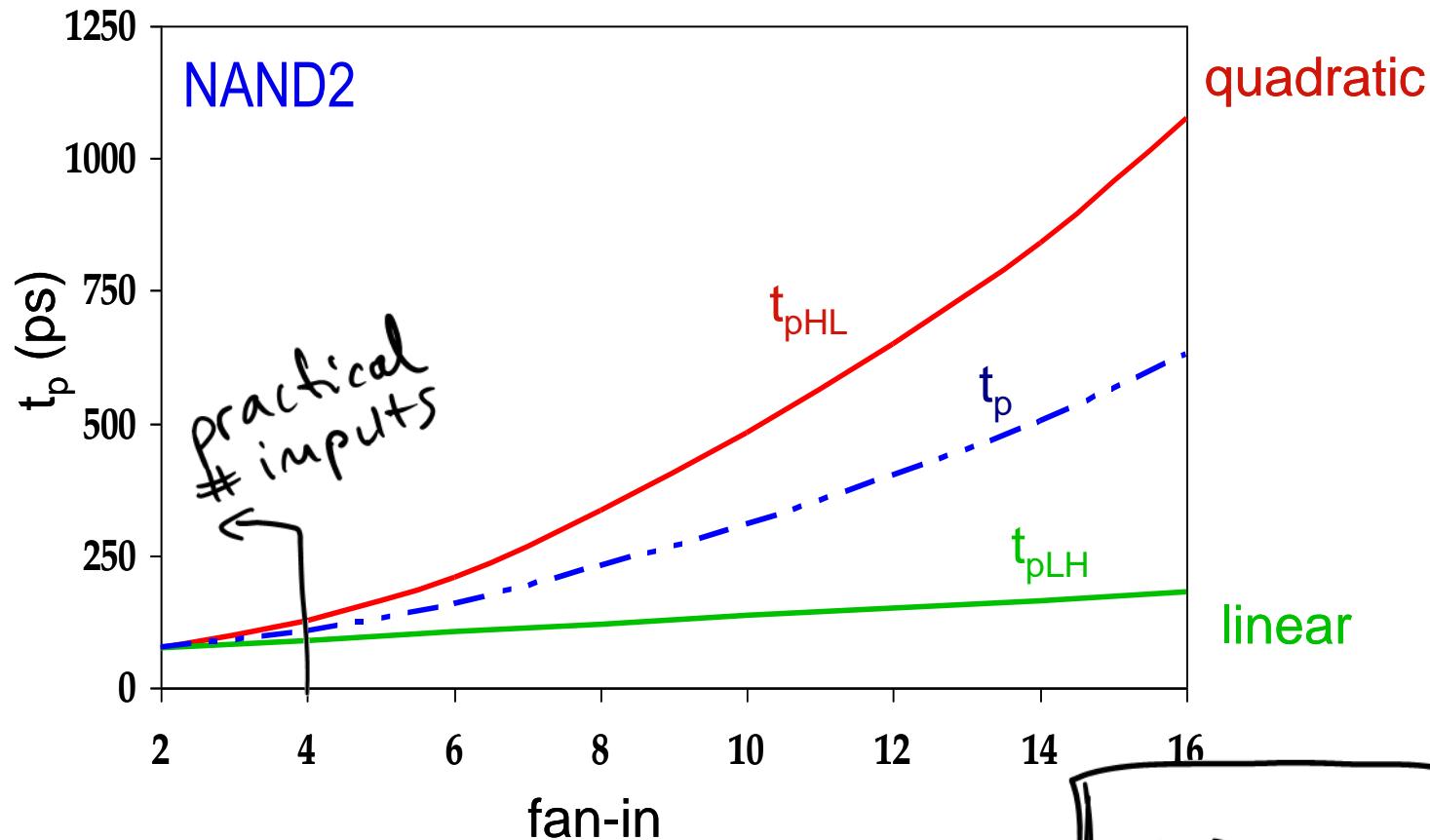


Distributed RC model
(Elmore delay)

$$t_{PHL} = 0.69 R_{eqn} (C_1 + 2C_2 + 3C_3 + 4C_L) \underset{C_1, 2C_2, 3C_3}{\sim} N^2$$

Propagation delay deteriorates rapidly
as a function of fan-in – **quadratically**
in the worst case.

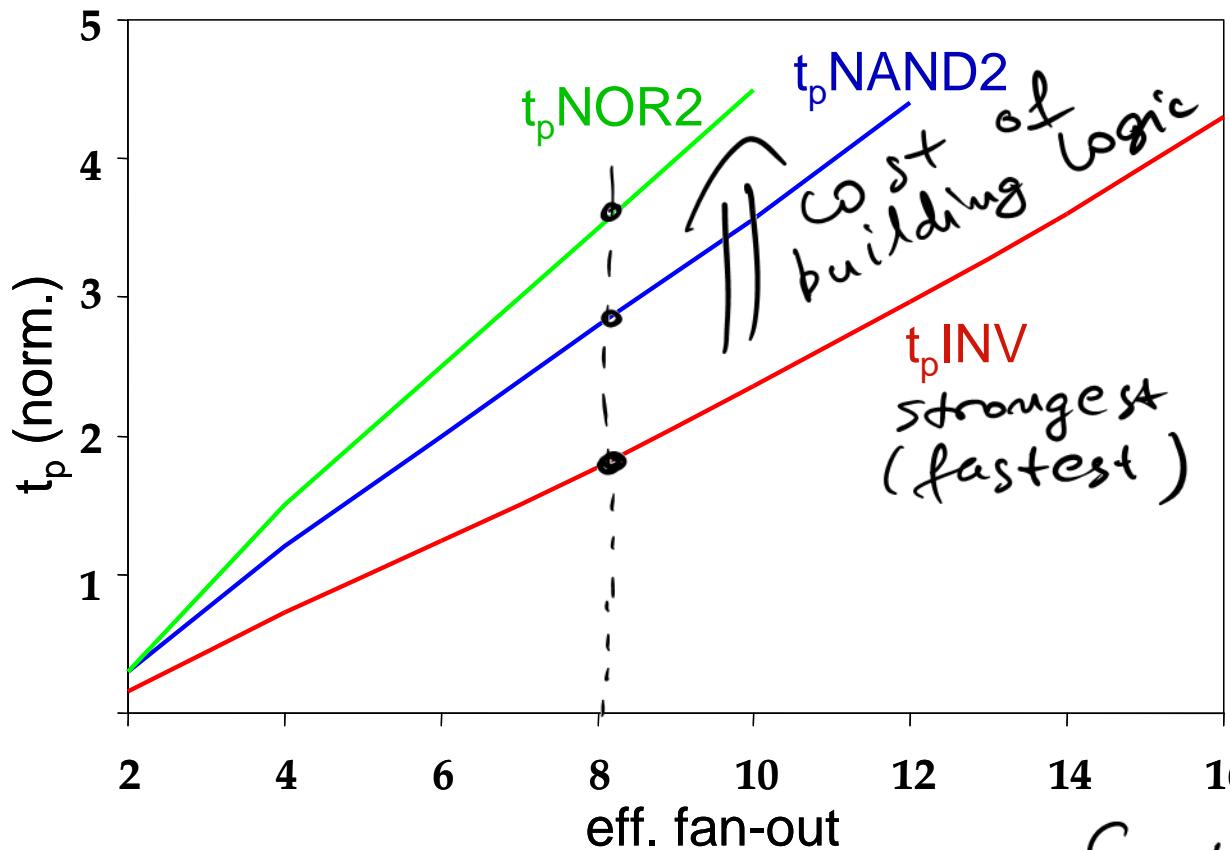
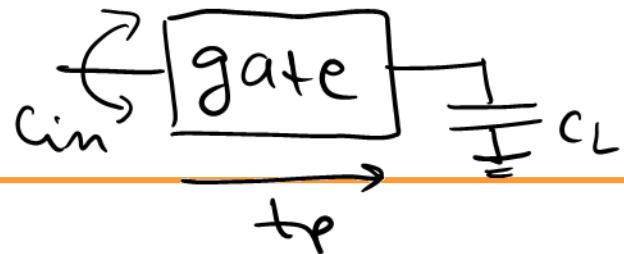
t_p as a Function of Fan-In



- ◆ Gates with $\text{fan-in} > 4$ should be avoided

FI ≤ 3

t_p as a Function of Fan-Out



- ◆ All gates have the same drive current
- ◆ Slope is a function of “driving strength”

C_L / C_{in} ($C_L \uparrow$)

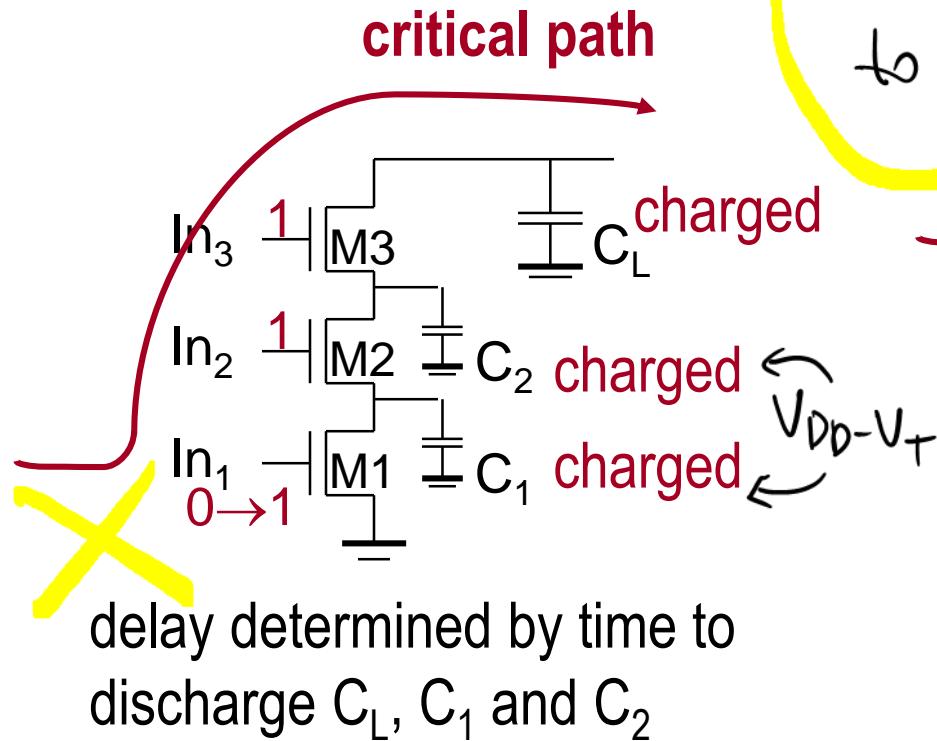
t_p as a Function of Fan-In and Fan-Out

- ◆ Fan-in: **quadratic** due to increasing R and C
- ◆ Fan-out: each additional fan-out gate adds **two** gate capacitances to C_L

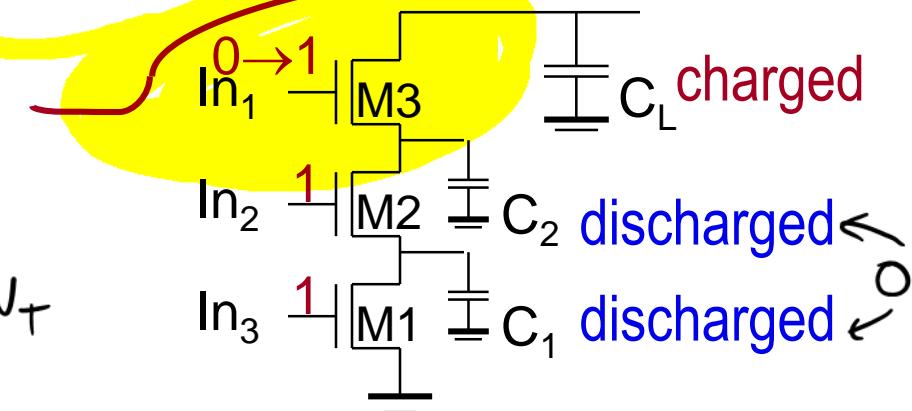
$$t_p = a_1 FI + a_2 FI^2 + a_3 FO$$

Fast Complex Gates: Design Technique #1 – Transistor Ordering

- Transistor ordering



wire up slowest input to
the top of the stack (away
from \overline{I}_N/N or V_{DD}/P)
to reduce to **critical path**

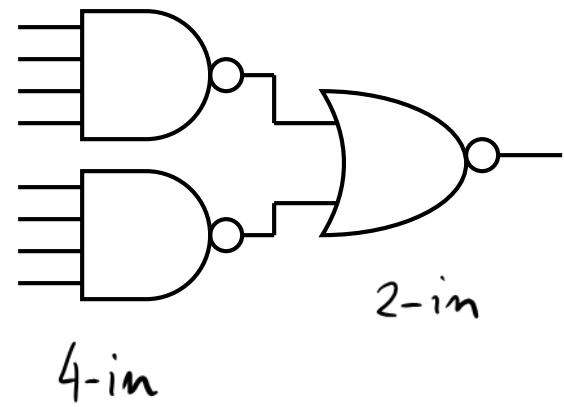
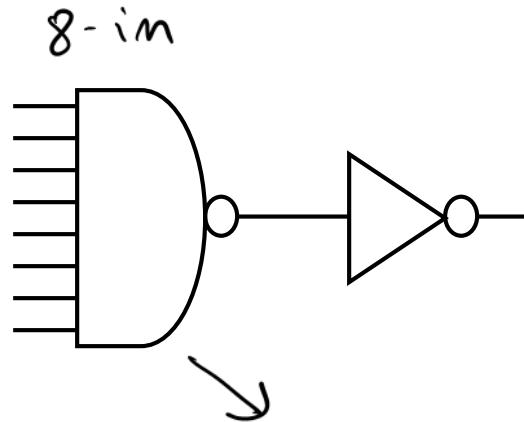
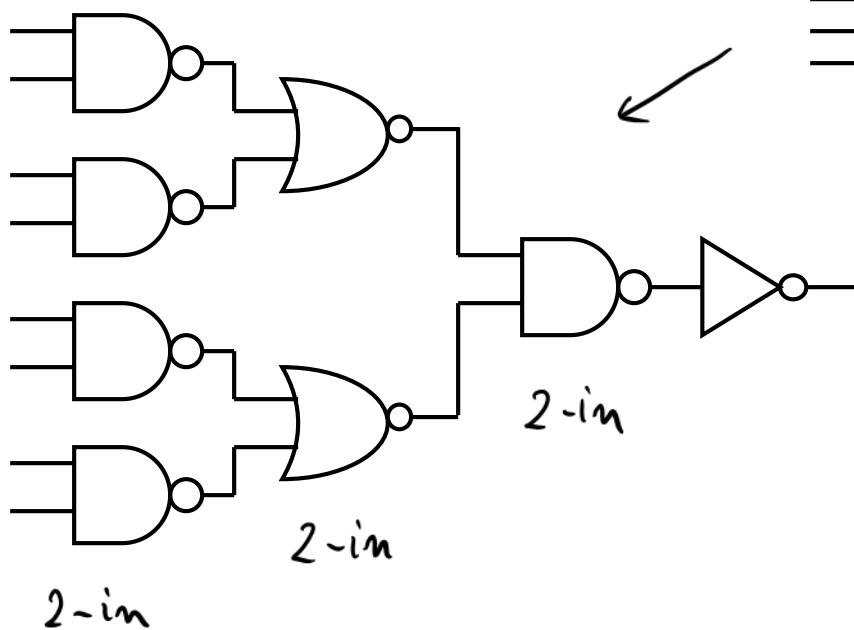


delay determined by time to
discharge C_L

Fast Complex Gates: Design Technique #2 – Alternative Logic Structures

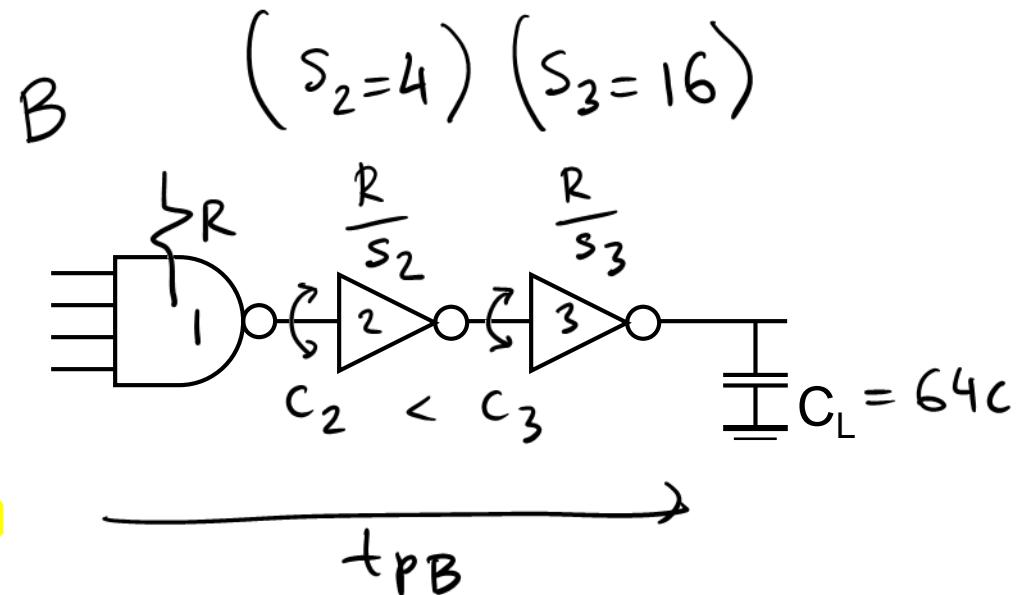
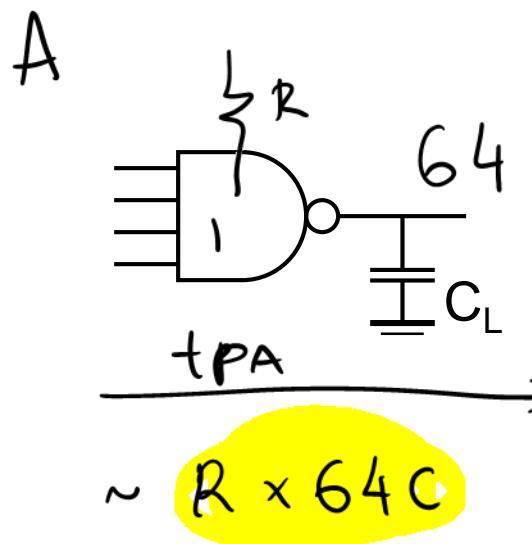
- ◆ Alternative logic structures

$$F = ABCDEFGH$$

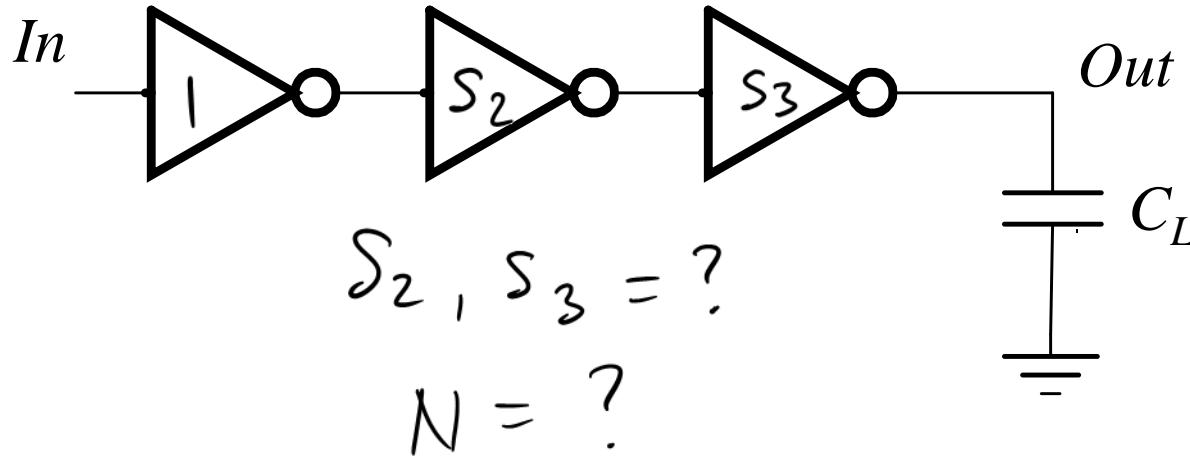


Fast Complex Gates: Design Technique #3 – Buffer Insertion

- Isolating fan-in from fan-out using buffer insertion

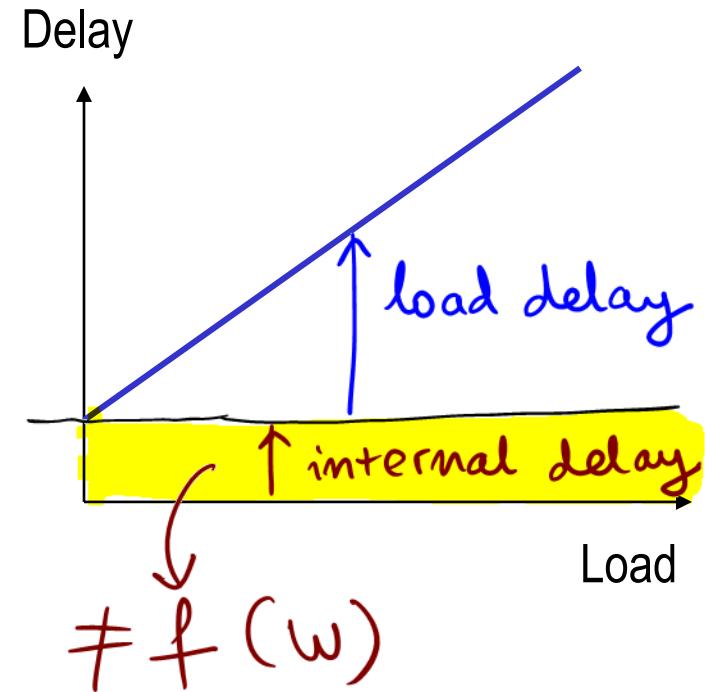
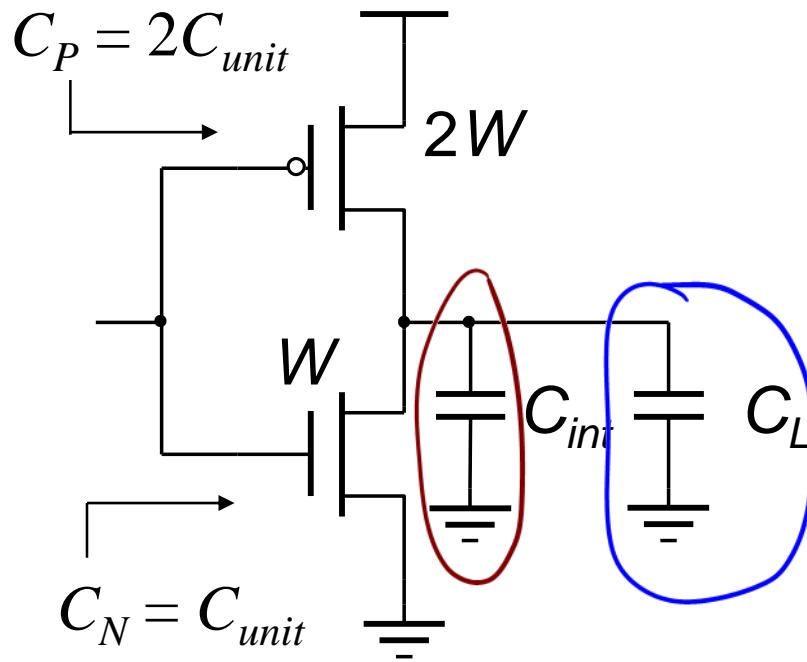


Example: Inverter Chain



- ◆ If C_L is given:
 - How many stages are needed to minimize delay?
 - How to size the inverters?
- ◆ May need some additional constraints

Inverter: Complete Delay Model



$$\text{Delay} = kR_W(C_{int} + C_L) = kR_WC_{int} + kR_WC_L = \mathbf{kR_W C_{int}(1 + C_L / C_{int})}$$

= Delay (Internal) + Delay (Load)

constant

variable

\rightarrow sizing (w) dependent