ECE520 – VLSI Design

Lecture 11: Combinational Static Logic

Prof. Payman Zarkesh-Ha

Office: ECE Bldg. 230B
Office hours: Wednesday 2:00-3:00PM or by appointment
E-mail: pzarkesh@unm.edu

Review of Last Lecture

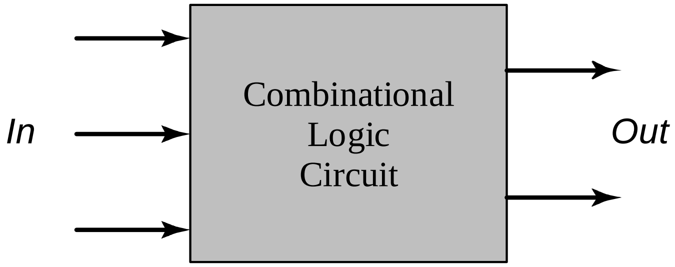
- ❑ **L-Edit Demo**
- ❑ **Layout Techniques**
  - **Design for density**
  - **Design for performance**
  - **Design for reliability**

Today’s Lecture

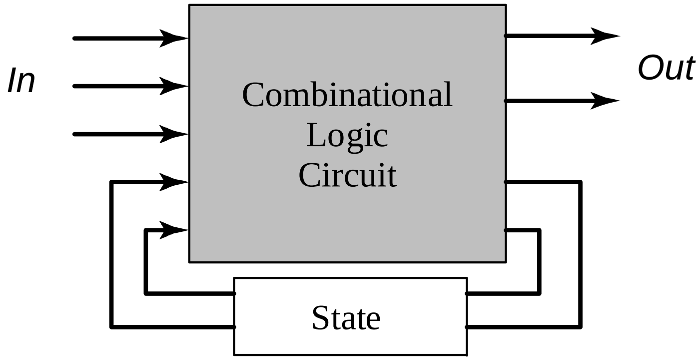
- ❑ **Combinational Logic**
  - **Logic Design**
  - **Transistor Sizing**
  - **Delay Analysis**

Combinational versus Sequential Logic

Combinational Logic: Out=f(in)

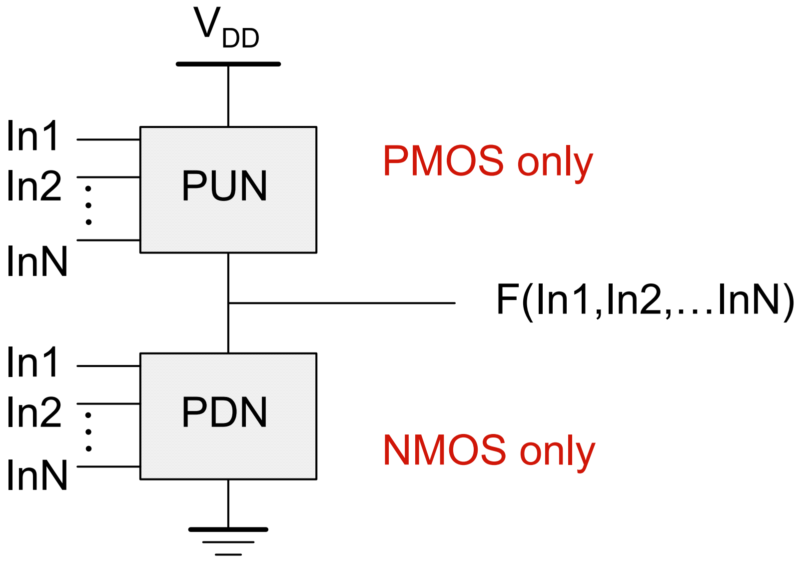


sequential Logic: Out=f(in,previous_in)



- **Combinational Logic:**
  - **Output is a function of present inputs (delayed by the propagation delay) i.e., do not contain memory**
  - **Implements logic functions like NAND, NOR, XOR, Multiplex or any complex functions such as Decoder, adder, shifter**
- **Sequential Logic:**
  - **Implement memory**
  - **Stores past values**
  - **Edge sensitive: Flip-flops**
  - **Level sensitive: Latches**

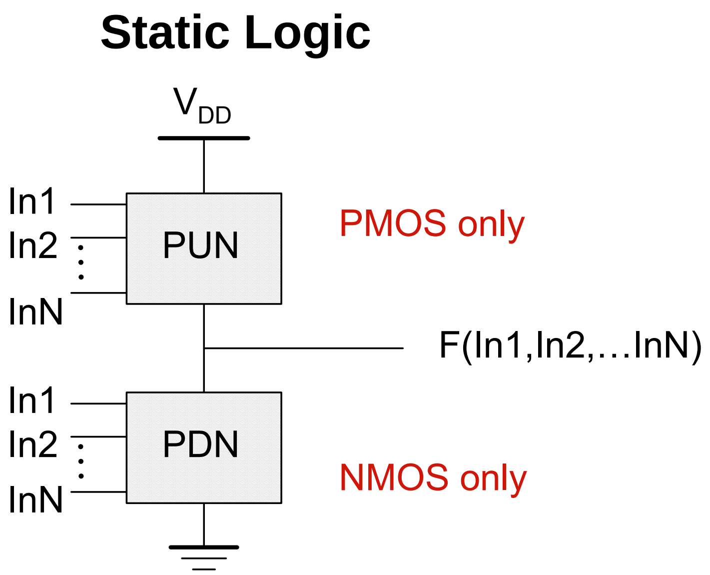
Static CMOS Logic Gate

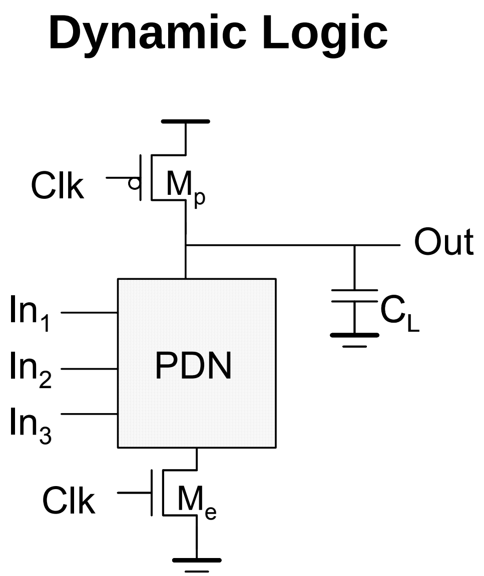


PUN and PDN are dual logic networks

- ❑ **Static Logic is a gate where the output is maintained at 0 or 1 as long as power is applied**
- ❑ **PUN and PDN are dual (Complimentary) to drive the output from 0 to 1 and 1 to 0 full rail (VSS to VDD)**

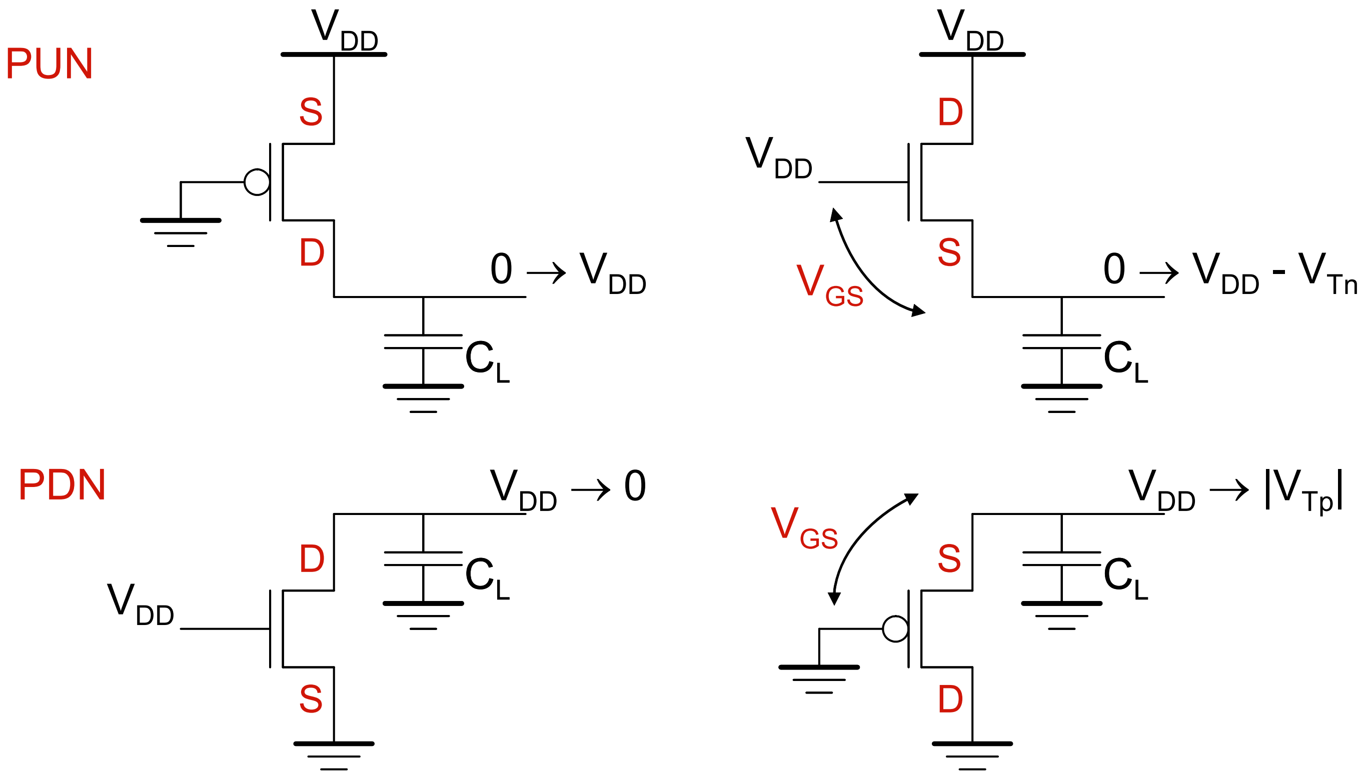
Static versus Dynamic Logic Gate





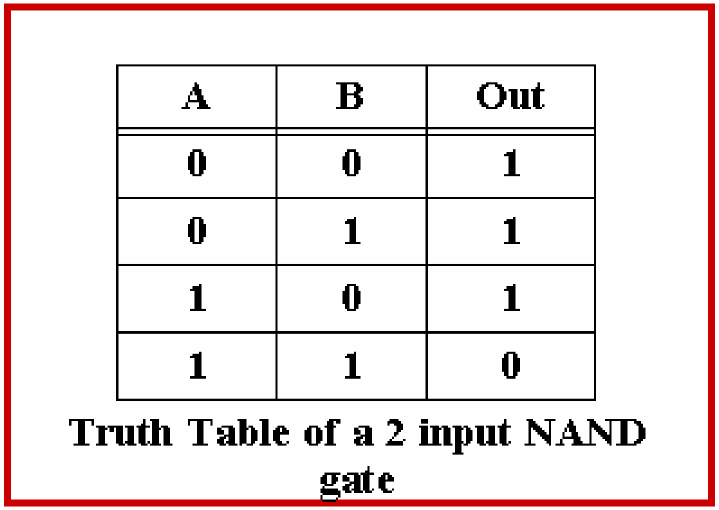
- ❑ **Unlike static logic, in dynamic logic the output may leak away**
- ❑ **The output is valid only after pre-charge when clock applies**

PUN and PDN Networks



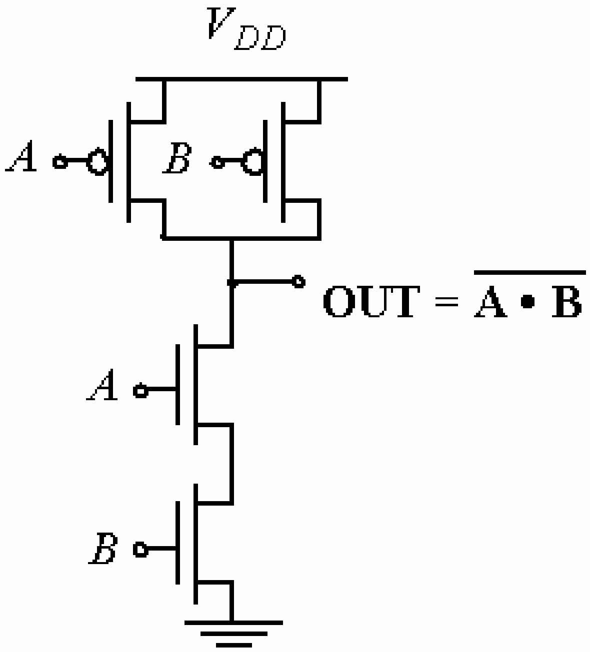
- **NMOS is a good pull down device (PDN)**
- **PMOS is a good pull up device (PUN)**

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

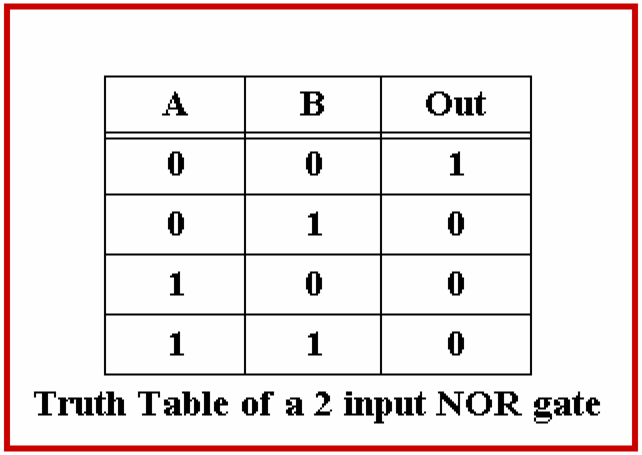


PUN: $\bar{A}+\bar{B}=\overline{AB}$ **(Conduction to VDD)**

PDN: **AB** **(Conduction to GND)**

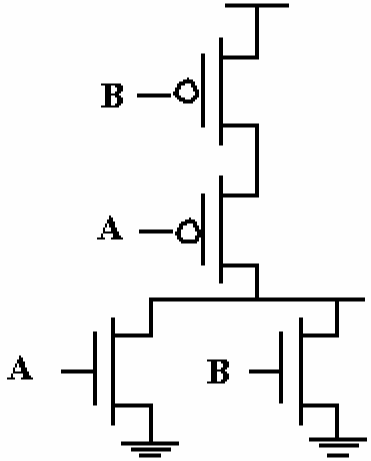
- **How do you size transistors to have approximately the same delay as an inverter?**

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



$OUT = \overline{A+B}$

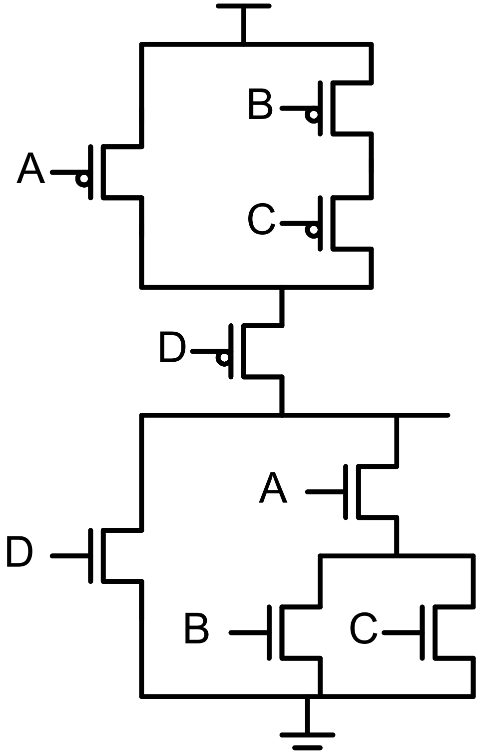
PUN: $\overline{A}\cdot\overline{B} = \overline{A+B}$ (Conduction to VDD)

PDN: $A+B$ (Conduction to GND)

- ❑ How do you size transistors to have approximately the same delay as an inverter?
- ❑ Which one is better to use often: NOR or NAND?

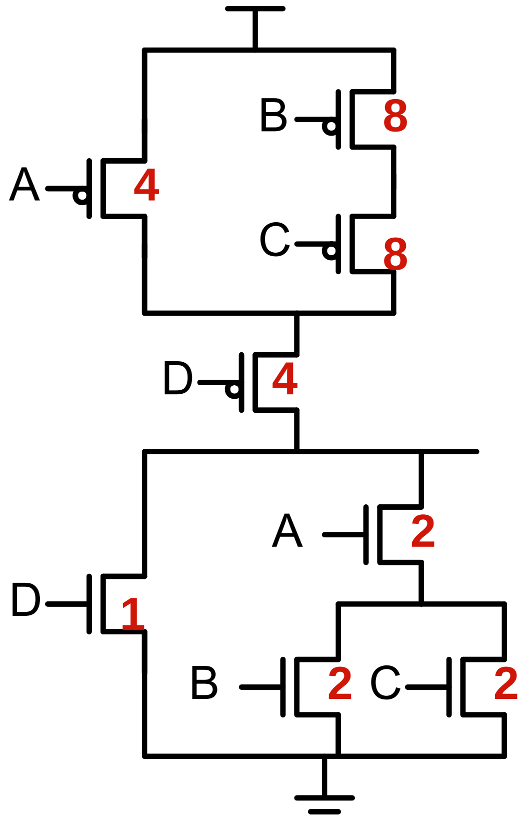
Example: More Complicated Gate

- **Consider F=D+A(B+C)**
  - **Stacks give the AND function**
  - **Parallel gates give the OR function**
- **Derive the PDN in hierarchical fashion**
- **Build the complement for the PUN**
  - **Convert stacks to parallel**
  - **Convert parallel to stacks**



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

Transistor Sizing of a Complex CMOS Gate



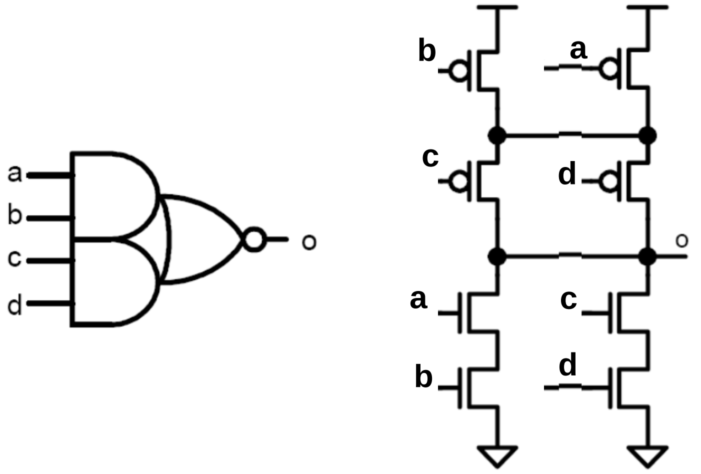
$$OUT = \overline{D + A \cdot (B + C)}$$

- ☐ **Size of transistors to have the same delay of an inverter when $(W/L)_P=2$ and $(W/L)_N=1$**

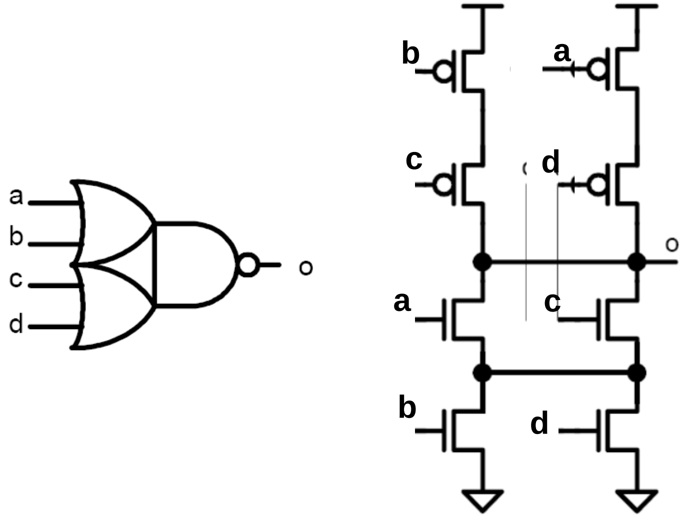
Example: AOI and OAI Gates

- ❑ **AND-OR-Invert**
  - Are used quite often in VLSI design
  - Relatively complex function in one stage
  - Can also be used as multiplexer

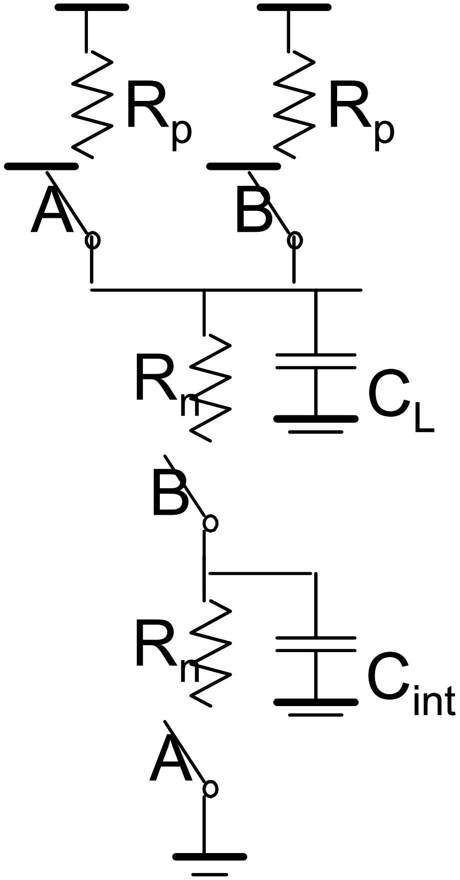
AOI Logic Gate



OAI Logic Gate

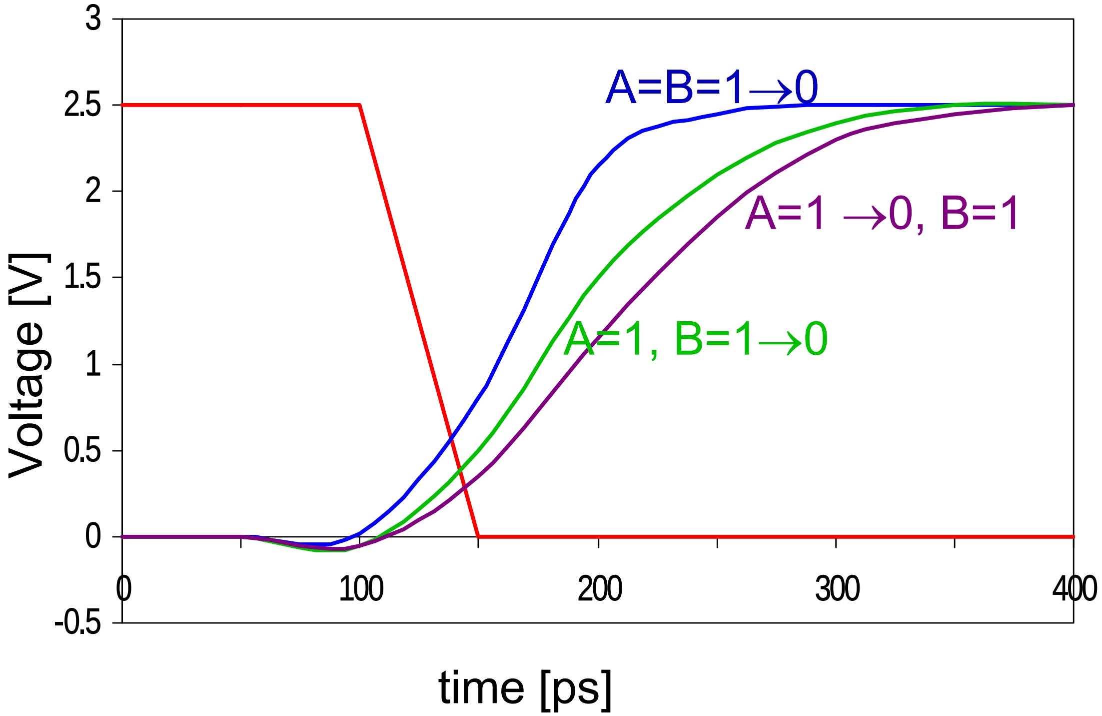


Gate Dynamic Behavior (NAND)



- ☐ **Delay is dependent on the pattern of inputs**
- ☐ **Low to high transition**
  - **both inputs go low then delay is 0.69 $R_p/2$ C_L**
  - **one input goes low then delay is 0.69 R_p C_L**
- ☐ **High to low transition**
  - **both inputs go high then delay is 0.69 $2R_n$ C_L**
- ☐ **More accurate delay estimation is very complicated**
  - **Internal capacitances (C_{int})**
  - **Effective load capacitance (depends on ON transistors and C_{int})**
  - **Body effect of stacked transistors**

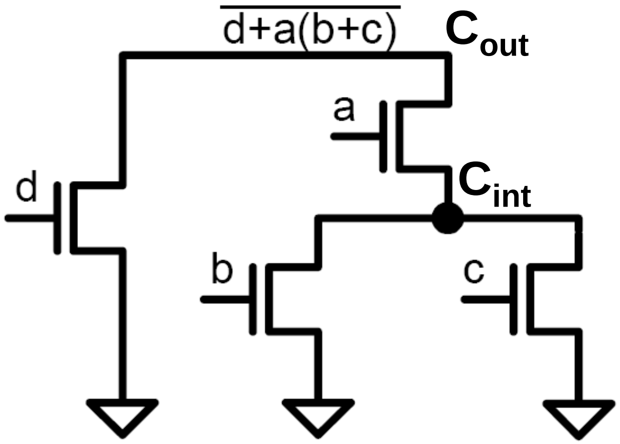
Gate Dynamic Behavior (NAND)



Input Data Pattern	**Delay (psec)**
A=B=0→1	67
A=1, B=0→1	64
A= 0→1, B=1	61
A=B=1→0	45
A=1, B=1→0	80
A= 1→0, B=1	81

NMOS = 0.5μm/0.25 μm
PMOS = 0.75μm/0.25 μm
C_L = 100 fF

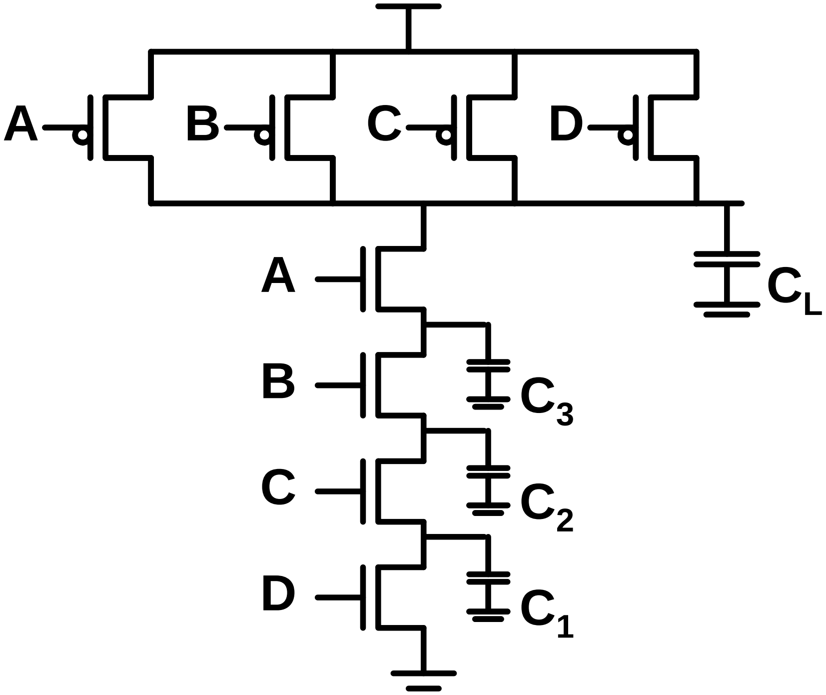
Delay Estimation Using Elmore Delay Model

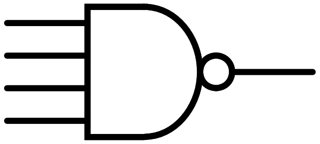


$$\tau_{DN} = R_B C_{int} + (R_A + R_B) C_{out}$$

- Elmore delay model can be used to take C_{int} into account
- The total capacitance seen when discharging via the transistor controlled by input D depends on the voltage applied to input A
- This is that “side branch” capacitance

Another Example





- Using Elmore delay model the worst case delay is:

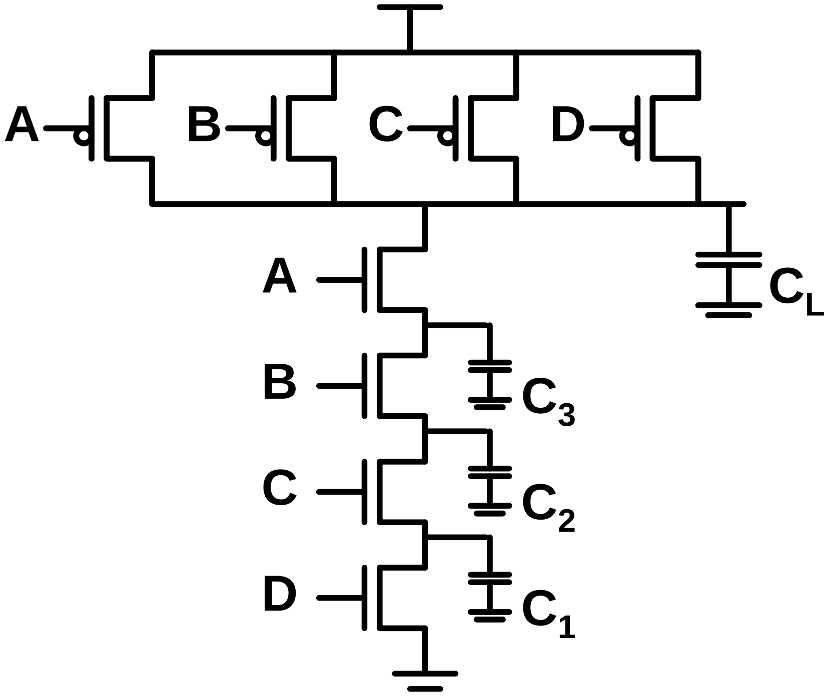
$$t_{PHL} = 0.69\left[R_D C_1 + (R_D + R_C)C_2 + (R_D + R_C + R_B)C_3 + (R_D + R_C + R_B + R_A)C_L\right]$$

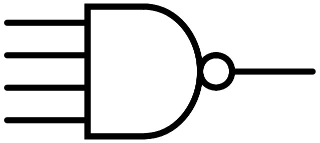
- Note that R_D appears in all terms

Some Gate Inputs are Faster than Others

- ☐ **Transistor with input D is the slowest transistor**
  - It is also the furthest to the output
  - Put the earliest arriving input there
  - This gives time for the other series transistors to discharge the intermediate nodes before the latest input arrives
  - Gate symbols should be labeled to show the fast vs. slow inputs
  - This can be as simple as order, e.g., a, b, c, d…
  - Note that this means a is the bottom on a NOR, top on NAND
- ☐ **Example**
  - 2 input NAND gate with inputs a and b
  - If input b arrives at 5.1ns and a arrives at 5.5ns the RC is just $R_A C_{OUT}$ rather than $R_B C_{INT} + (R_A + R_B)\, C_{INT}$
  - If the time to discharge the intermediate node is < 400 ps
  - Don’t forget the upper transistor does not have appreciable V_{GS}-V_T until the source node drops

More on Delay Dependency to Input Pattern





- a=0, b=1, c=1, d=1 leaves the intermediate nodes pre-discharged
- But a=1, b=1, c=1, d=0 leaves them at VDD - V_{Tn}
- Therefore, when all inputs go to “one” the speed may vary