ECE 2300
Digital Logic & Computer Organization
Spring 2025

CMOS Logic

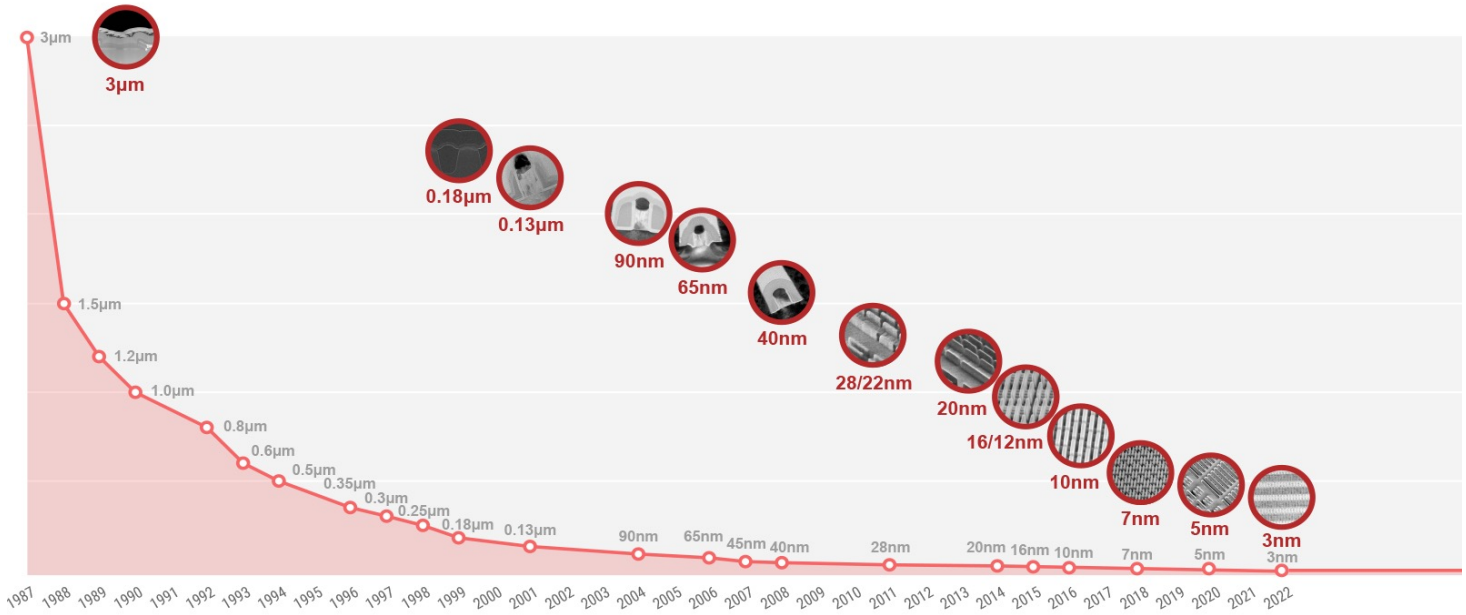
Cornell University

Announcements

- Links to lecture recordings & previous quiz questions are in the pinned post on Ed

A Little Bit of History

- **Transistors**
  - Invented by John Bardeen, Walter Brattain, and William Shockley at Bell Labs in 1947
- **Integrated circuits (IC)**
  - Independently developed by Jack Kilby (at TI) and Robert Noyce (at Fairchild) in the 1950s
  - Robert Noyce and Gordon Moore founded Intel in 1968

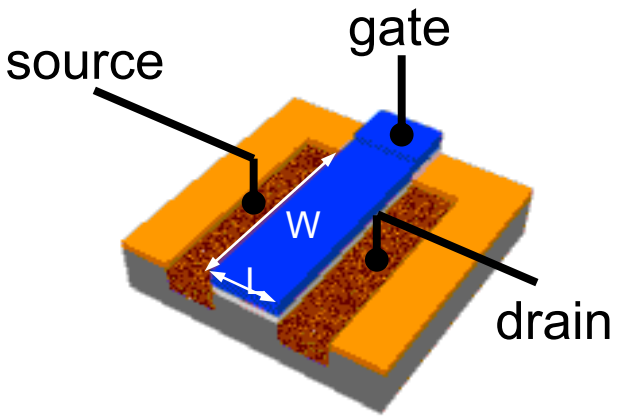


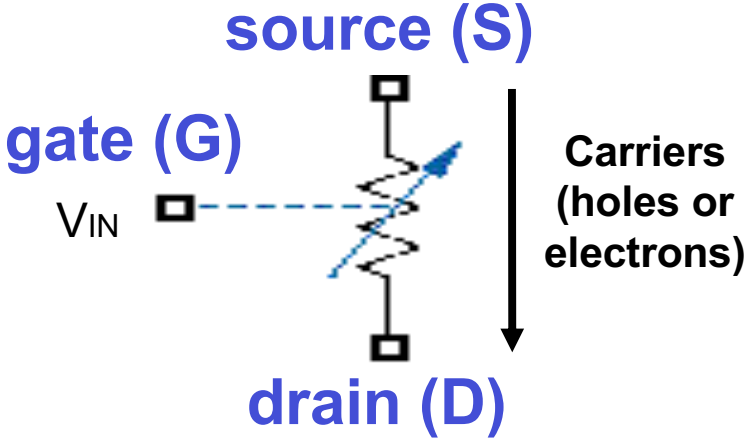
Technology scaling of CMOS transistors

https://www.tsmc.com/english/dedicatedFoundry/technology/logic

MOS Transistors

- **Metal-Oxide Semiconductor Field-Effect Transistors (MOSFETs)**
  - **MOS for short**

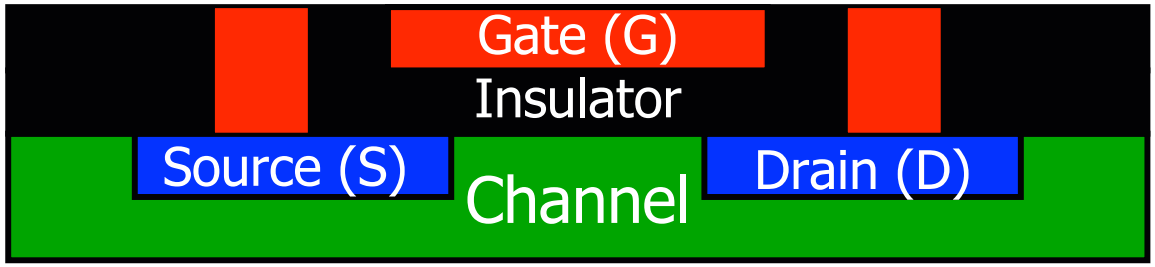




A 3-terminal device that changes its conductance (or resistance) based on the voltage applied to the gate terminal

- **Extreme changes in resistance (0 to ∞) make transistors act like switches**

MOSFET



(cross-section view of a MOSFET)

MOS: three materials needed to make a transistor

- **Metal: strong conductor of current**
- **Oxide: insulator (does not conduct)**
- **Semiconductor: conduction can be controlled**

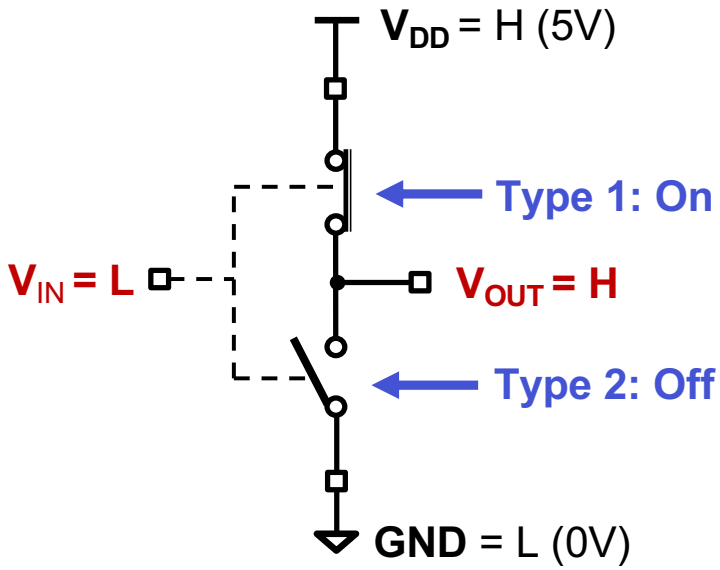
FET: Field Effect Transistor

- **Acts as a voltage-controlled switch: gate voltage creates electric field that turns on/off connection between source and drain**
  - **Two types of switches: active low and active high**
    (active = connection is on)

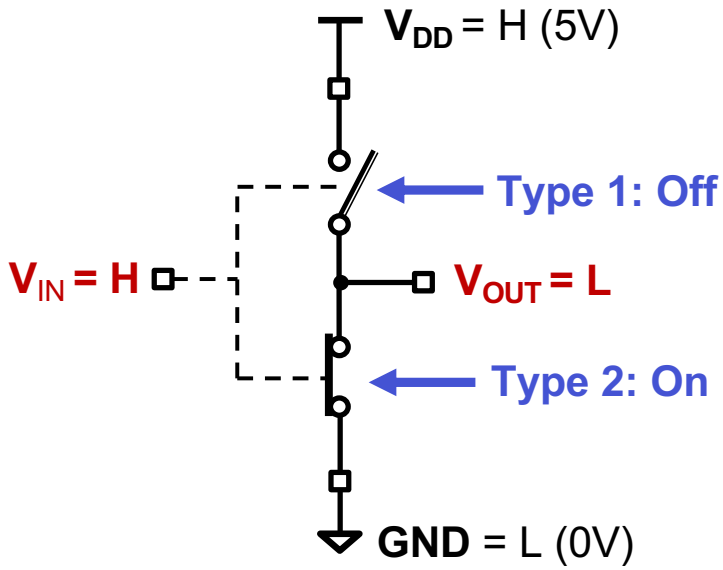
Optional readings: [1] "How Does a Transistor Work?" by Veritasium; [2] "Working of Transistors" by Lesics

NOT Gate Using Switches

- **Can build an inverter using two types of complementary switches**
  - **Type 1 (active low):** ON when input = 0, OFF when input = 1
  - **Type 2 (active high):** ON when input = 1, OFF when input = 0



When input voltage is low, output is connected to voltage supply (V_{DD})

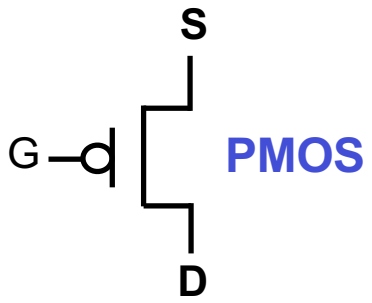


When input voltage is high, output is connected to ground (GND)

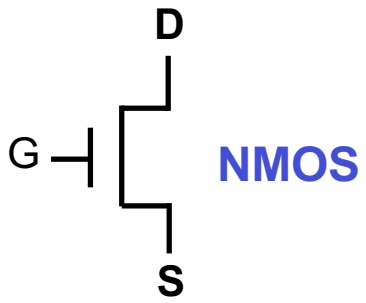
Two Types of MOS Transistors

- PMOS: Type 1 switch; NMOS: Type 2 switch

- Current flows when ON (*conducting*)
- No current flows when OFF (*not conducting*)



Type 1 (active low)



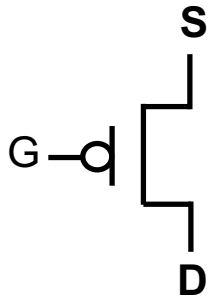
Type 2 (active high)

G: Gate S: Source D: Drain

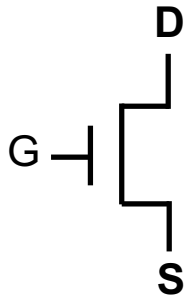
○ Bubble indicates active low

Two Types of MOS Transistors

- **PMOS**
  - **Closed when input is low** [1]
  - **Open when input is high**
  - **Passes a good one (but a poor zero)** [2]



- **NMOS**
  - **Closed when input is high** [1]
  - **Open when input is low**
  - **Passes a good zero (but a poor one)** [2]

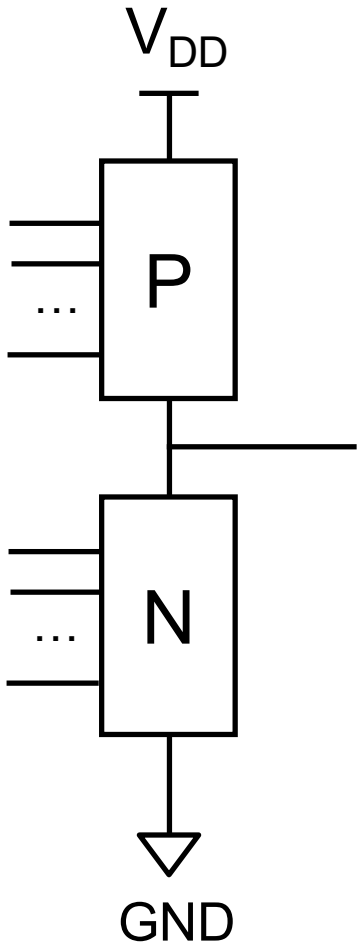


PMOS and NMOS have *complementary* properties

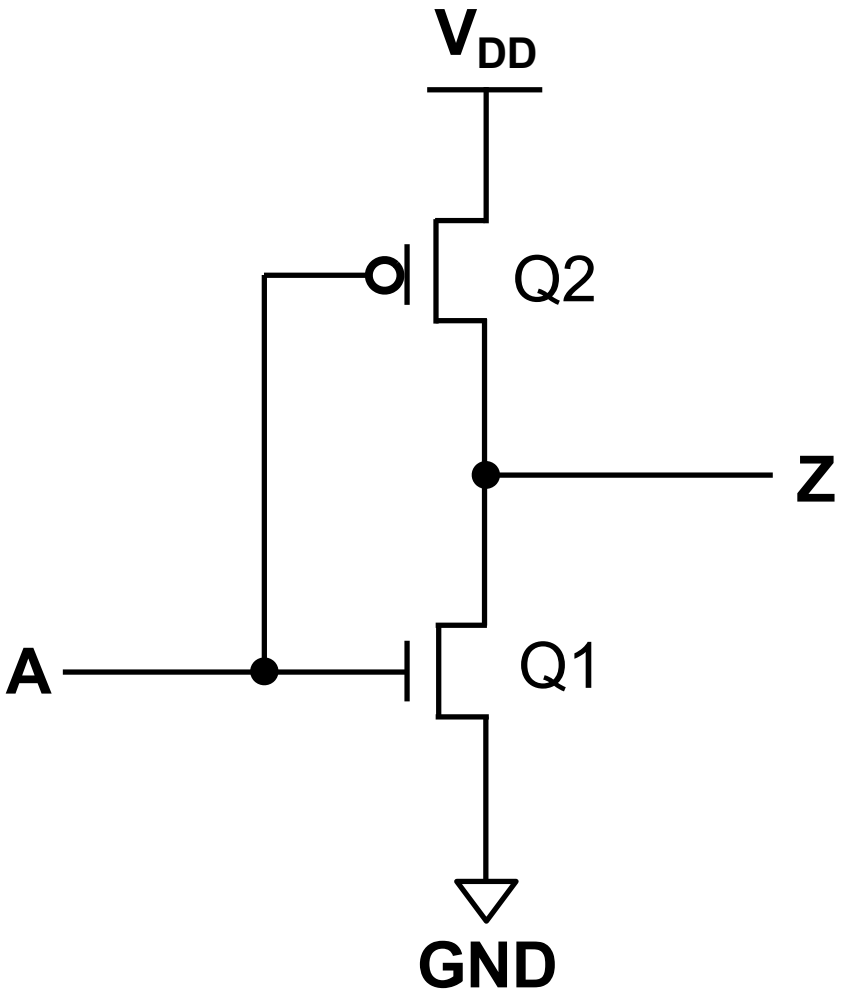
[1] In both cases, the voltage difference between gate and source must exceed a certain threshold voltage before the the transistor starts having any effect.
[2] Optional reading: vlsimsee.blogspot.com/2013/05/why-cant-nmos-pass-1-and-pmos-pass-0.html

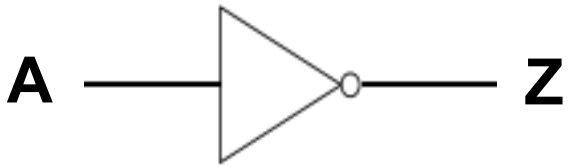
CMOS Logic Gates

- **Complementary MOS (CMOS)**
  - CMOS dominates the digital IC market
- **Uses both NMOS and PMOS devices such that there is no direct supply-ground path**
  - Dissipates little power when the inputs don't change
- **Our focus: Static CMOS gates**
  - Other types exist as well (pseudo-NMOS, domino, ...)



CMOS Inverter



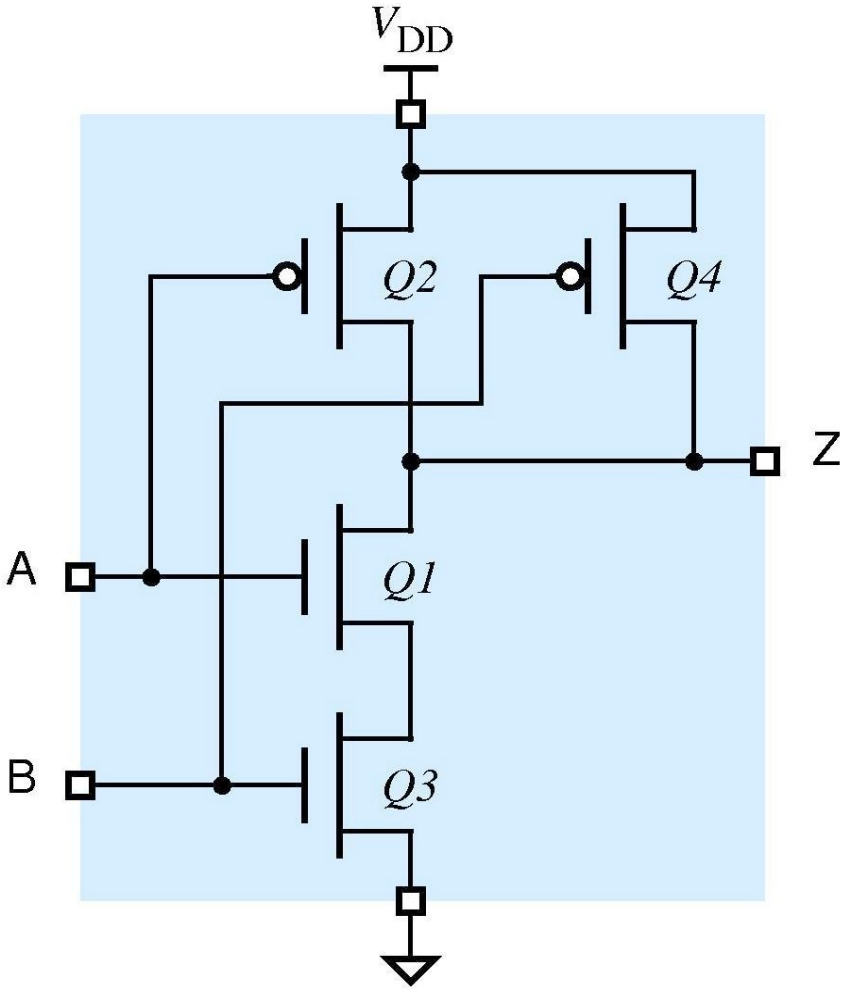


A is high
- Q1 is on
- Q2 is off
- Z is low

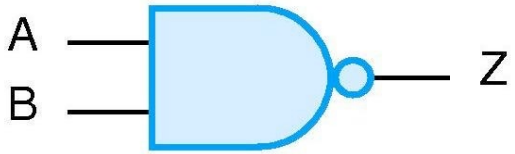
A is low
- Q1 is off
- Q2 is on
- Z is high

A	Z
L	H
H	L

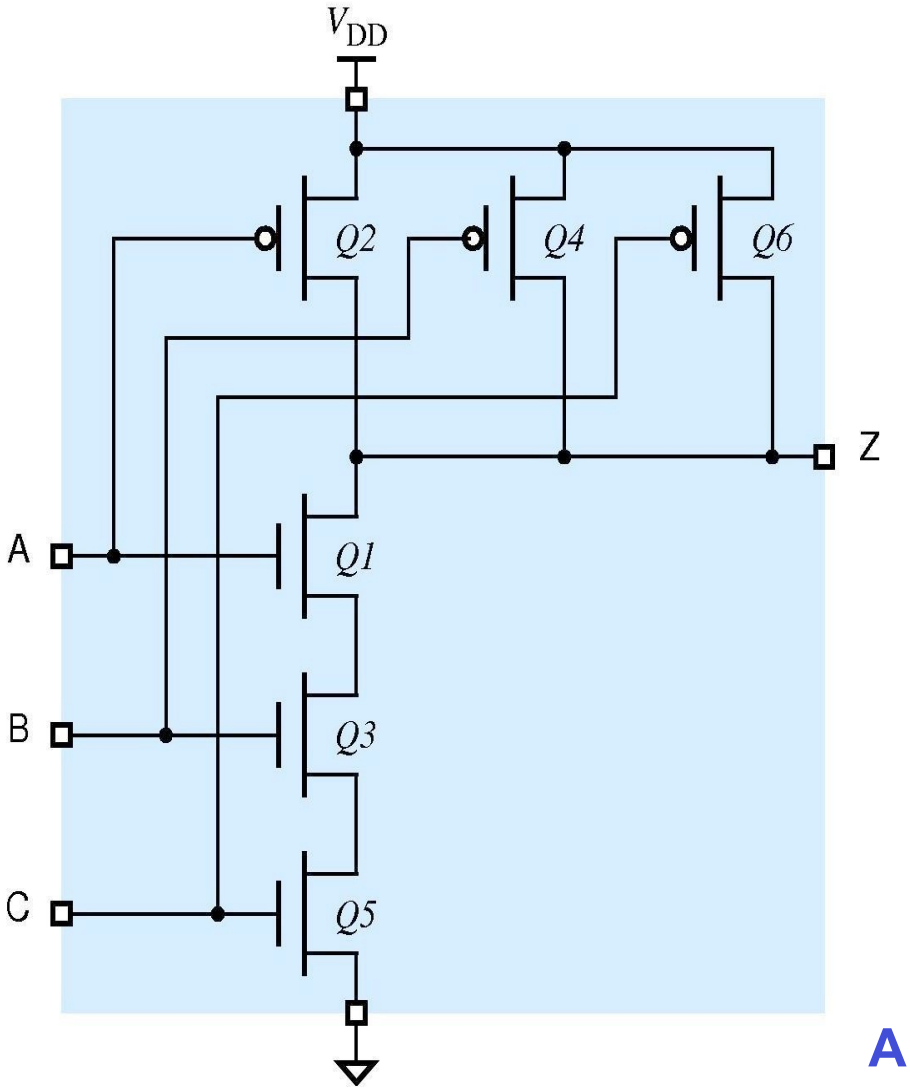
CMOS NAND Gate



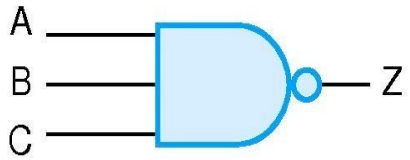
A	B	*Q1*	*Q2*	*Q3*	*Q4*	Z
L	L	off	on	off	on	H
L	H	off	on	on	off	H
H	L	on	off	off	on	H
H	H	on	off	on	off	L



3 Input CMOS NAND Gate

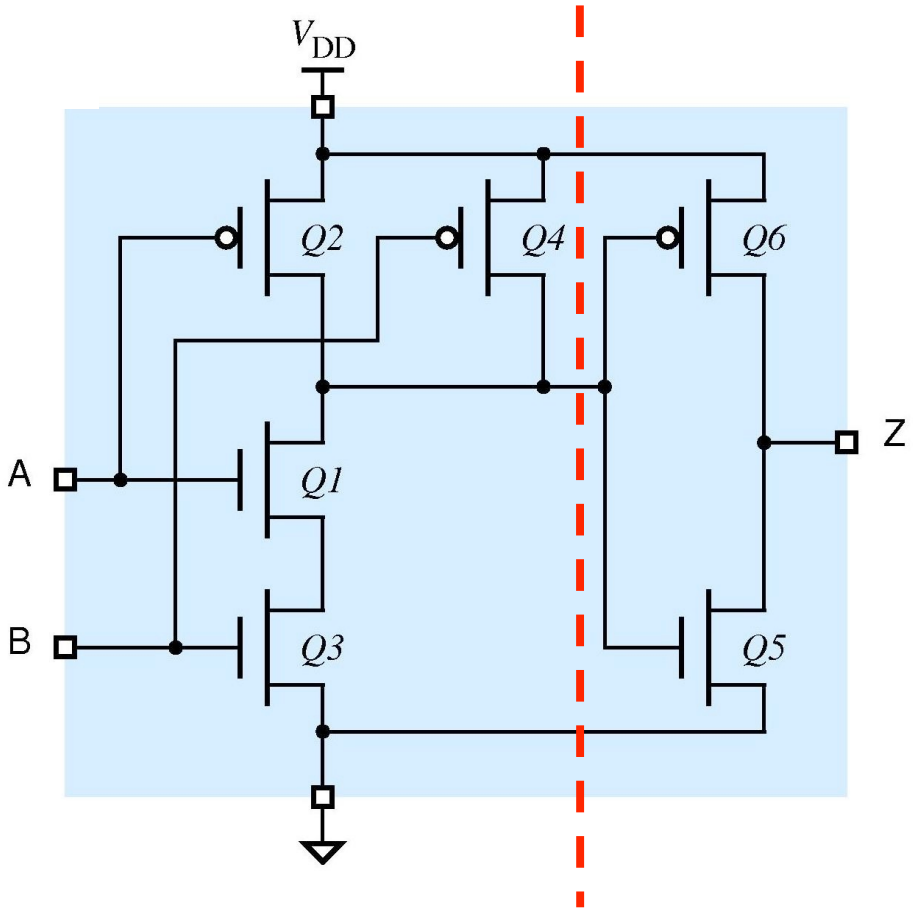


A	B	C	*Q1*	*Q2*	*Q3*	*Q4*	*Q5*	*Q6*	Z
L	L	L	off	on	off	on	off	on	H
L	L	H	off	on	off	on	on	off	H
L	H	L	off	on	on	off	off	on	H
L	H	H	off	on	on	off	on	off	H
H	L	L	on	off	off	on	off	on	H
H	L	H	on	off	off	on	on	off	H
H	H	L	on	off	on	off	off	on	H
H	H	H	on	off	on	off	on	off	L



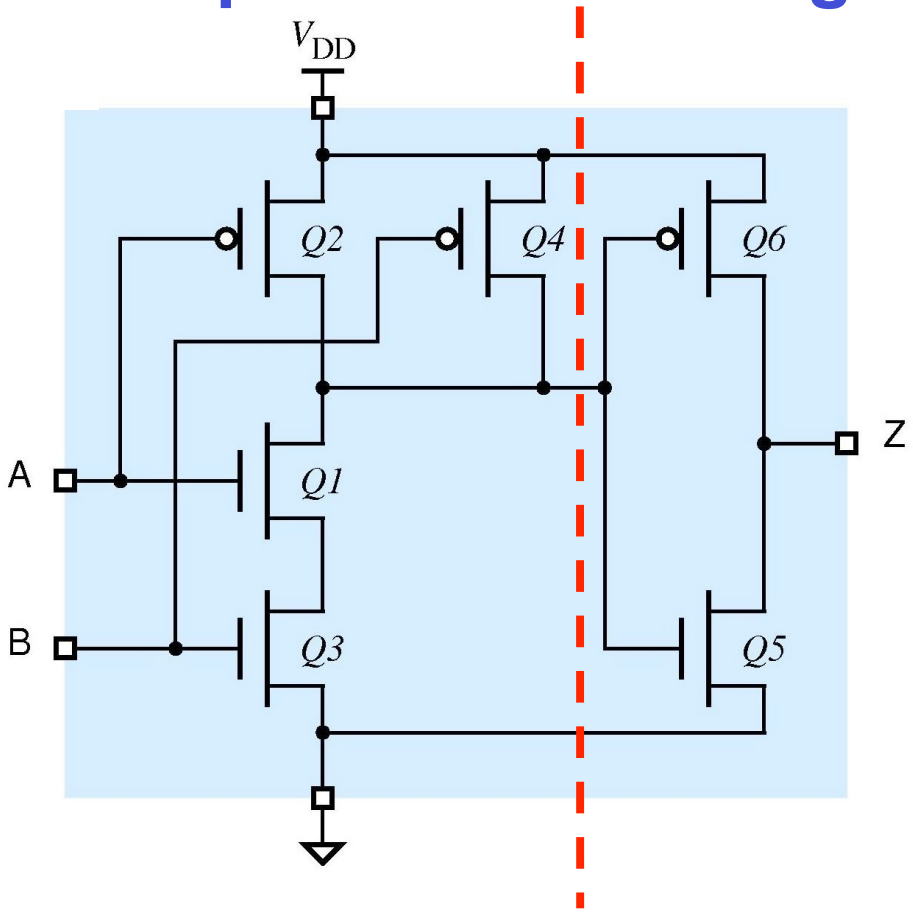
An *n*-input NAND uses 2*n* transistors

Another "Mystery" Gate

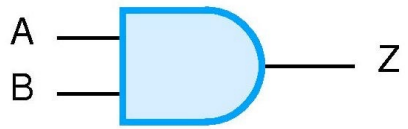


2-Input AND Gate in CMOS

- CMOS gates produce inherent inversion
- Need to invert to the output of NAND to implement an AND gate

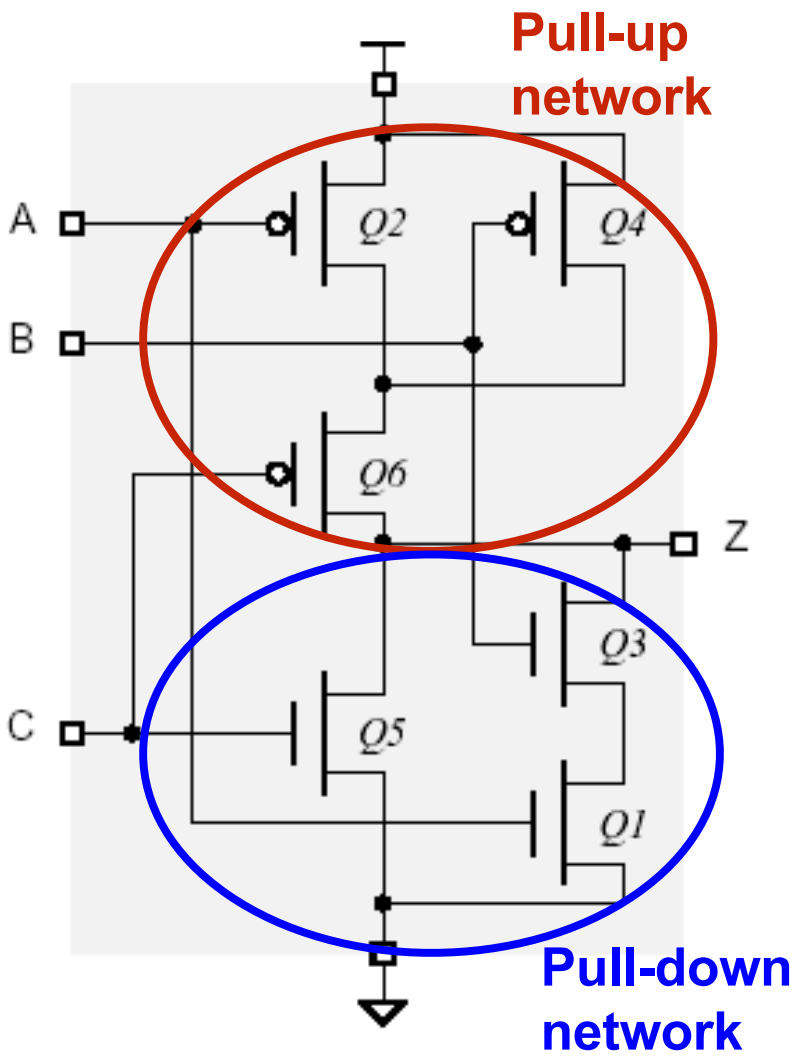


A	B	Q1	Q2	Q3	Q4	Q5	Q6	Z
L	L	off	on	off	on	on	off	L
L	H	off	on	on	off	on	off	L
H	L	on	off	off	on	on	off	L
H	H	on	off	on	off	off	on	H



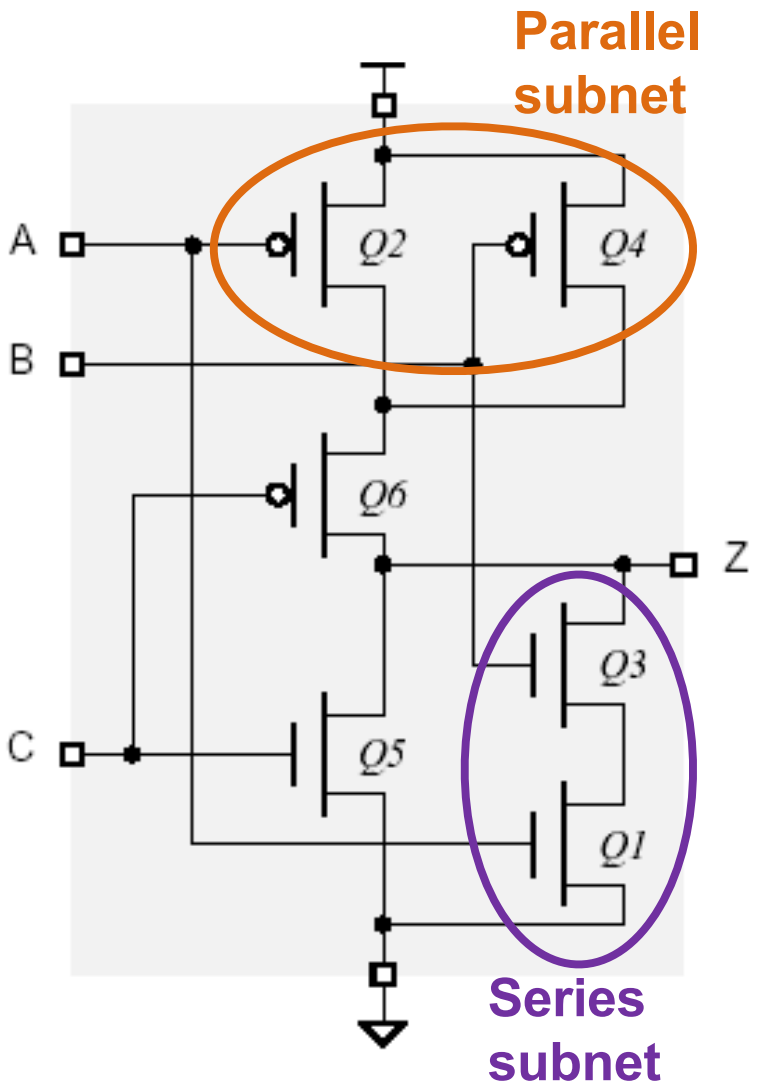
Structure of Transistor Networks

- **Two complementary networks**
  - **A pull-up network composed of PMOS, with sources tied to voltage supply**
  - **A pull-down network composed of NMOS, with sources tied to ground**
  - **Equal number of NMOS and PMOS transistors**



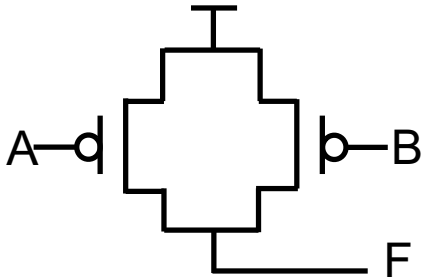
Structure of Transistor Networks

- The pull-up and pull-down networks are always duals
- To construct the dual of a network:
  - Exchange NMOS for PMOS (and vice versa)
  - Exchange series subnets for parallel subnets (and vice versa)
    - This transformation applies to hierarchical structures



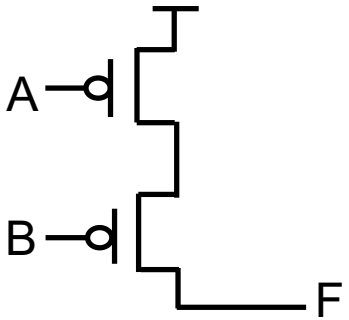
Series and Parallel Subnets

- **Parallel**
  - ***At least one*** **input must be 0 (p-type) or 1 (n-type) to make the connection**



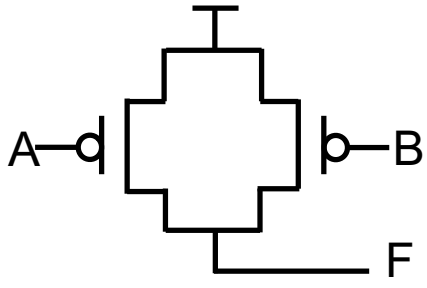
F = A'+B'

- **Series**
  - ***All*** **inputs must be 0 (p-type) or 1 (n-type) to make the connection**

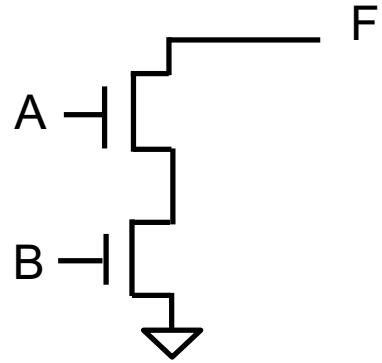


F = A'·B'

Duality of Parallel/Series Subnets

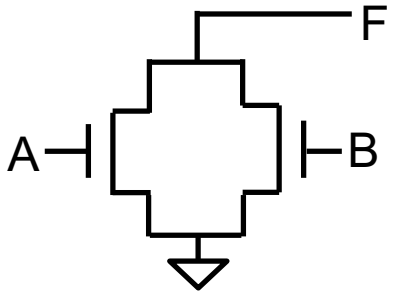


=

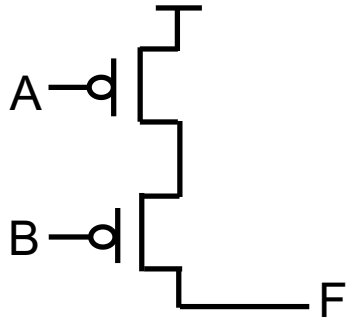


p-type parallel subnet
F pulls up to 1 when A or B is low =>
F = A'+B' = (A•B)'

n-type series subnet
F pulls down to 0 when A and B are high =>
F = (A•B)'



=

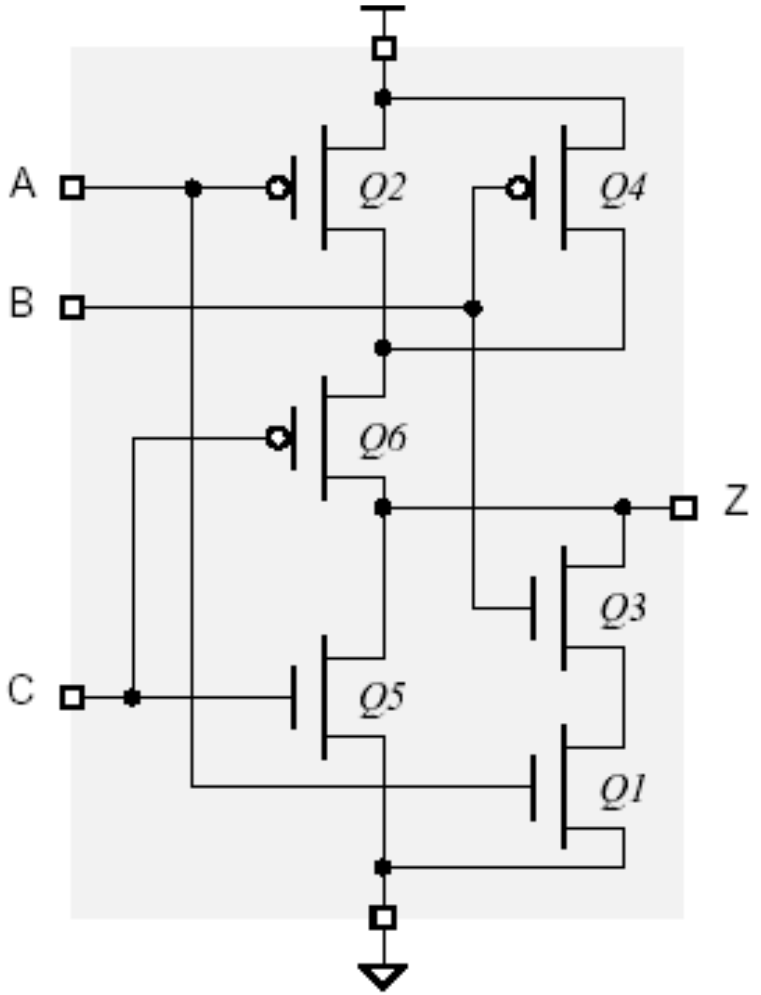


n-type parallel subnet
F pulls down to 0 when A or B is high =>
F = (A+B)'

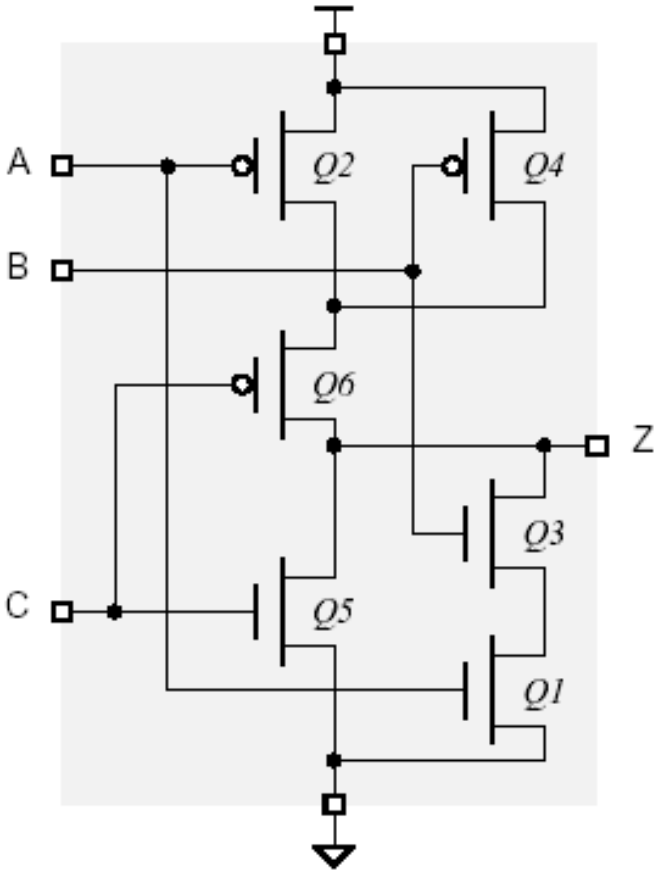
p-type series subnet
F pulls up to 1 when A and B are low =>
F = A'•B' = (A+B)'

Analysis of Transistor Networks

- **Transistor states**
  - Determine all possible input combinations
  - Figure out the state of each transistor
  - Determine final output
- ***or* by inspection**
  - Figure out what input combinations cause a 1 (or a 0) output



Analysis of Transistor Networks

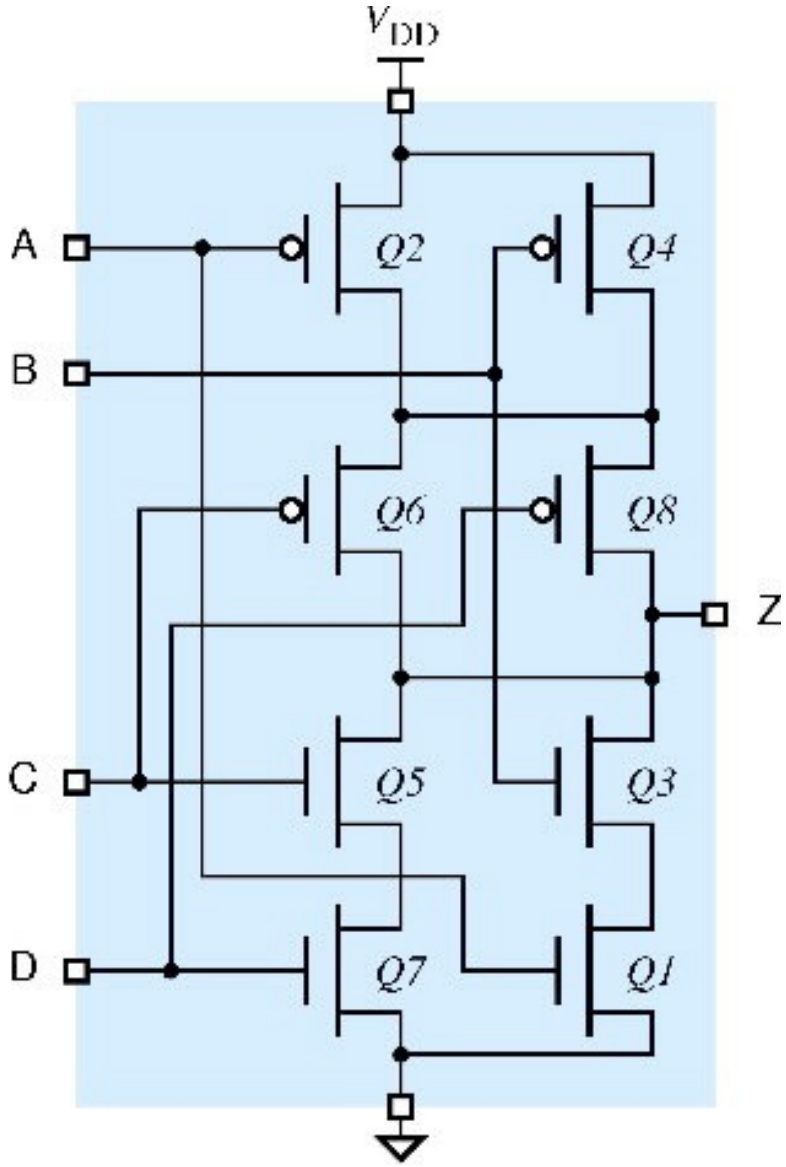


- **By inspection**
  - **Inspect *either* pull-up (PMOS) *or* pull-down (NMOS) network**
  - **Translate the series (parallel) subnets into product (sum) terms**
  - **For pull-down network, negate the combined expression**

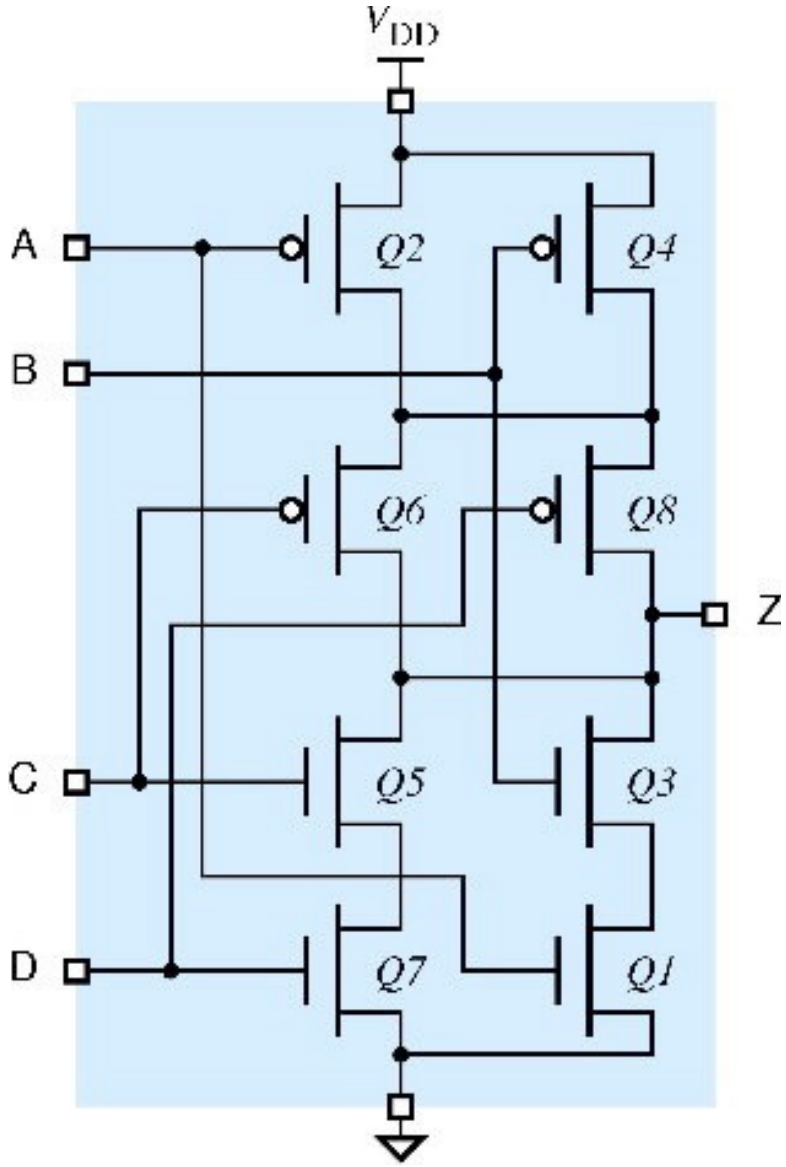
Pull-up: **(A'+B')C'**

Pull-down: **(A•B + C)'**

A More Complicated Circuit



A More Complicated Circuit



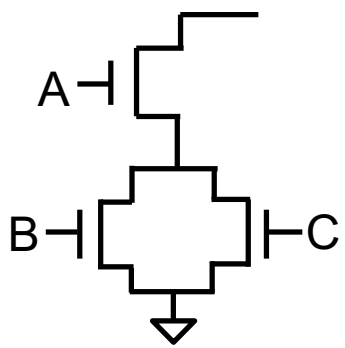
Pull-up: (A'+B')•(C'+D')

Pull-down: (A•B + C•D)'

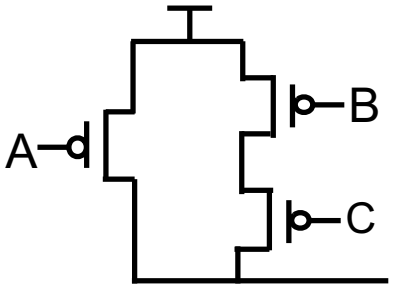
Constructing CMOS Gate from Boolean Expression

Example: F = (A•(B+C))'

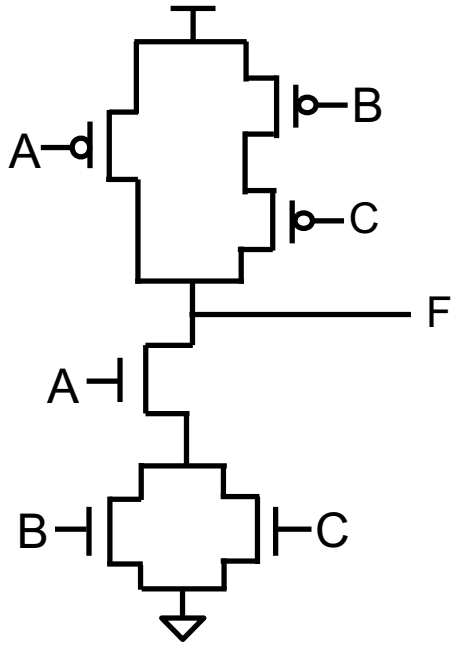
Step 1. Figure out pull-down network that does what you want (e.g., what combination of inputs generates a low output)



Step 2. Walk the hierarchy replacing NMOS with PMOS, series subnets with parallel subnets, and parallel subnets with series subnets

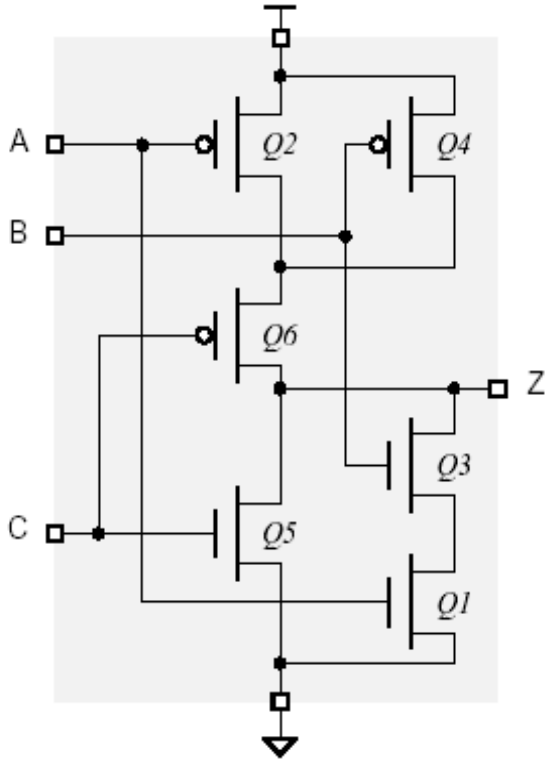


Step 3. Combine PMOS pull-up network (from Step 2) with NMOS pull-down network (from Step 1) to form a fully-complementary CMOS gate



CMOS Sanity Checks

- Equal number of NMOS and PMOS
- NMOS sources tied to ground or to drain of another NMOS
- PMOS sources tied to Vdd or drain of another PMOS
- Inputs tied to pairs of PMOS and NMOS transistors



Next Class

Sequential Logic: Clocks, Latches, Flip-Flops (H&H 3.1-3.2)