CET 141: Day 8

Dr. Noori KIM

Discipline

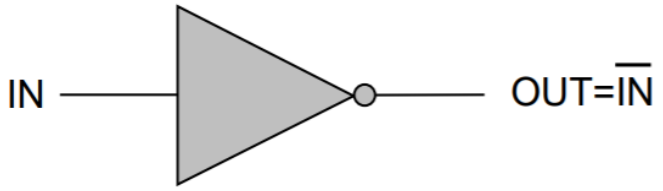
- Intentionally restrict design choices
- Example: Digital discipline
  - Discrete voltages instead of continuous
  - Simpler to design than analog circuits – can build more sophisticated systems
  - Digital systems replacing analog predecessors: i.e., digital cameras, digital television, cell phones, CDs

Digital Discipline → Binary Values

- **Two discrete values:**
  - 1's and 0's
  - 1, TRUE, HIGH
  - 0, FALSE, LOW
- **1 and 0:** voltage levels, rotating gears, fluid levels, etc.
- Digital circuits use **voltage** levels to represent 1 and 0
- ***Bit*****:** *B*inary dig*it*

Digital Discipline

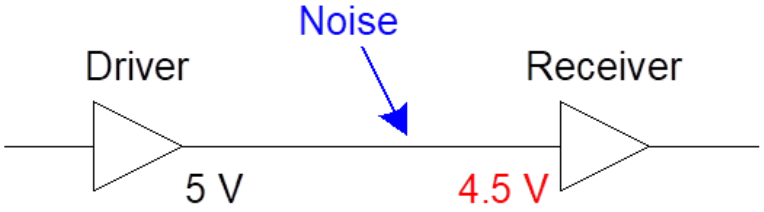
Digital	Analog
Logic level 1	V_{max} (i.e., 5V, 3.3V)
Logic level 0	V_{min} (i.e., 0V)



IN	OUT
0	1
1	0

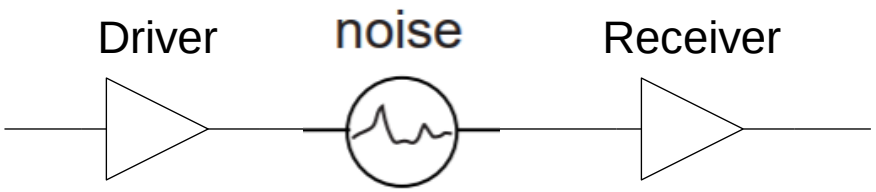
NOISE?

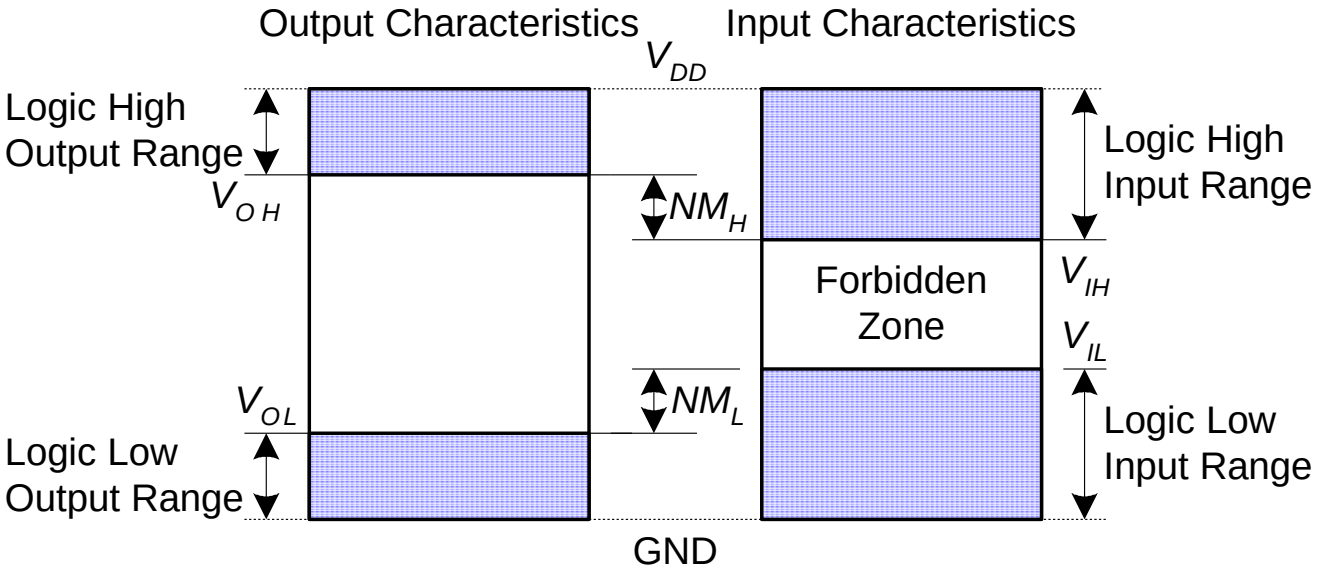
- Anything that degrades the signal
  - E.g., resistance, power supply noise, coupling to neighboring wires, etc.
  - Example: a gate (driver) outputs 5 V but, due to a long wire, receiver gets 4.5 V



- If range of V_{OL} to V_{OH} is wider than the range of V_{IL} to V_{IH}, then the buffer exhibits some "**noise immunity**".

A chain of two buffers





High Noise Margin: $NM_H = V_{OH} - V_{IH}$

Low Noise Margin: $NM_L = V_{IL} - V_{OL}$

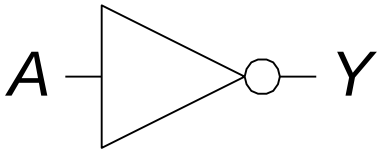
Logic gates and 74 series chips

Types of Logic Gates

- Perform logic functions:
  - inversion (NOT), AND, OR, NAND, NOR, etc.
- Single-input:
  - **NOT gate, buffer**
- Two-input:
  - AND, OR, XOR, NAND, NOR, XNOR
- Multiple-input

Single-Input Logic Gates

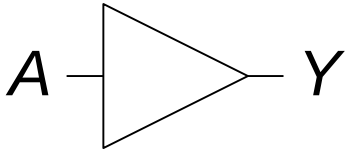
NOT



$Y = \overline{A}$

A	Y
0	1
1	0

BUF

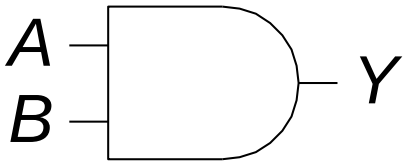


$Y = A$

A	Y
0	0
1	1

Two-Input Logic Gates

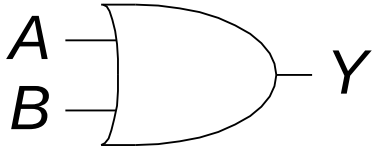
AND



$Y = AB$

A	B	Y
0	0	
0	1	
1	0	
1	1	

OR

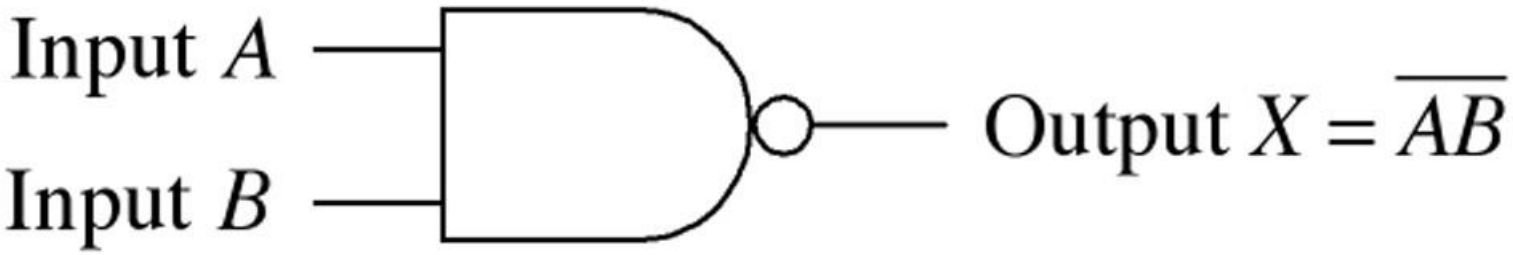


$Y = A + B$

A	B	Y
0	0	
0	1	
1	0	
1	1	

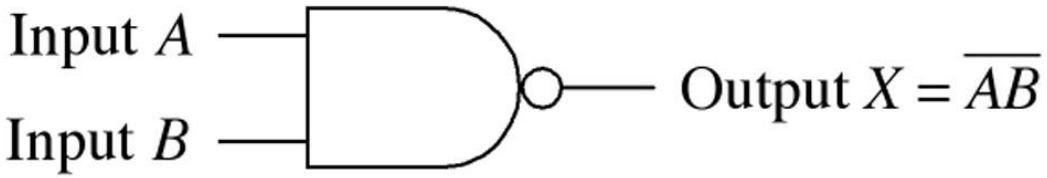
The NAND Gate

- Same as the AND gate except that its output is inverted
  - If A = 1 and B = 1, X = 0
  - If A = 0 or B = 0, X = 1



The NAND Gate

- Truth Table
- Boolean Equation: $X = \overline{AB}$
- Multiple inputs - the output is always HIGH unless all inputs go HIGH

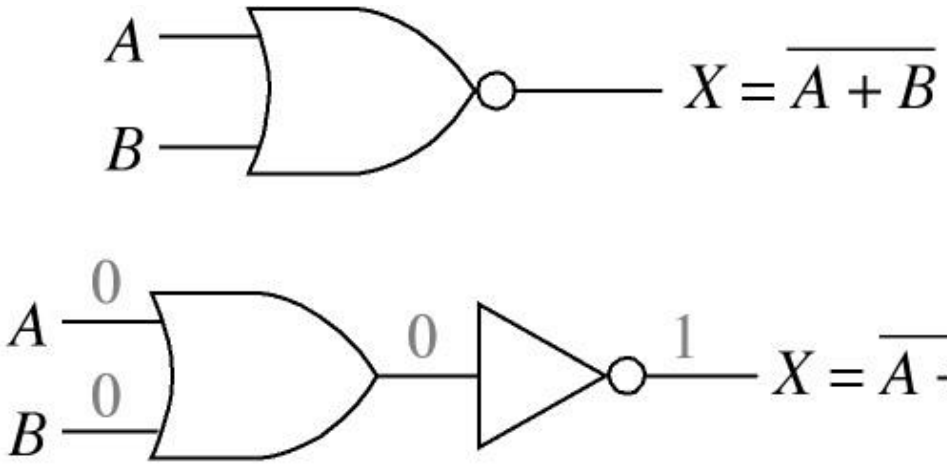


A	B	$X = \overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

Output is always HIGH unless both inputs are HIGH.

The NOR Gate

- Same as the OR gate except that its output is inverted
  - If A = 1 or B = 1, X = 0
  - If A = 0 and B = 0, X = 1



A	B	$X = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

XOR

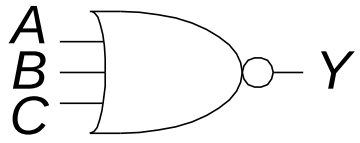
A	B	Y
0	0	
0	1	
1	0	
1	1	

XNOR

A	B	Y
0	0	
0	1	
1	0	
1	1	

Multiple-Input Logic Gates

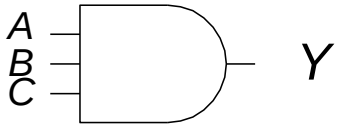
NOR3



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

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

AND3



$$Y = ABC$$

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Logic Family Examples

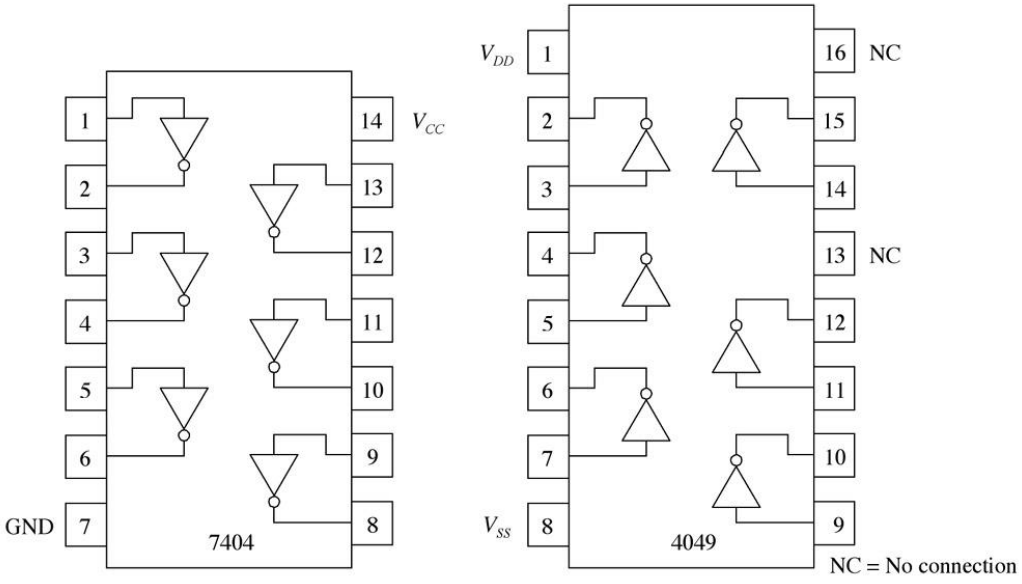
Logic Family	V_{DD}	V_{IL}	V_{IH}	V_{OL}	V_{OH}
TTL	5 (4.75 - 5.25)	0.8	2.0	0.4	2.4
CMOS	5 (4.5 - 6)	1.35	3.15	0.33	3.84
LVTTL	3.3 (3 - 3.6)	0.8	2.0	0.4	2.4
LVCMOS	3.3 (3 - 3.6)	0.9	1.8	0.36	2.7

A side bar: V_{DD} Scaling

- In 1970's and 1980's, V_{DD} = 5 V
- V_{DD} has dropped
  - Avoid frying tiny transistors
  - Save power
- 3.3 V, 2.5 V, 1.8 V, 1.5 V, 1.2 V, 1.0 V, …
  - Be careful connecting chips with different supply voltages

Using IC Logic Gates

- Hex - six gates



Notes on the most popular two families in the 7400 series gates:

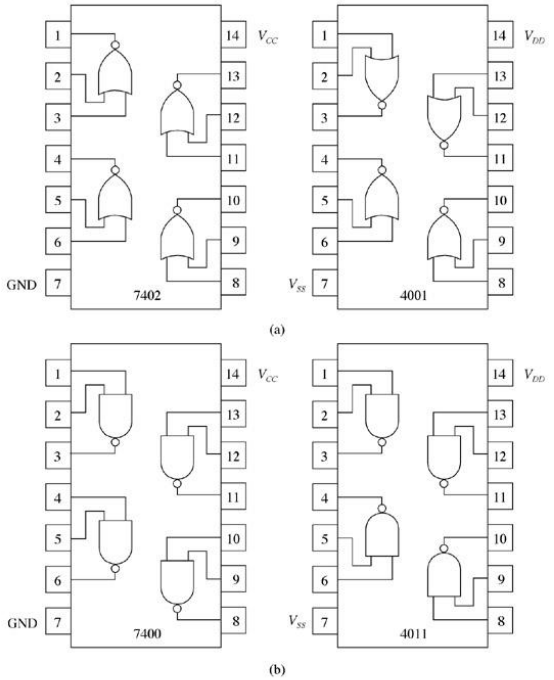
- 74LS: Low power Shottky, based on BJT/TTL transistors.
- 74HC: High speed CMOS, based on CMOS transistors.

Recommend to use the same family gates in your circuit!

- An HC device can drive a TTL device, but not vice-versa (V_{OH} of TTL is less than V_{IH} of CMOS. So if LS drives HC, the input falls down on the forbidden region)

Using IC Logic Gates

- Quad - four gates
- Three-, four-, and eight-input configurations



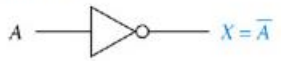
Common IC Logic Gates

TABLE 3–9 Common IC Gates in the TTL and CMOS Families

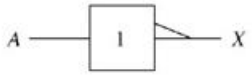
Gate Name	Number of Inputs per Gate	Number of Gates per Chip	Part Number			
			Basic TTL	LS TTL	HC CMOS	4000 CMOS
Inverter	1	6	7404	74LS04	74HC04	4009
AND	2	4	7408	74LS08	74HC08	4081
	3	3	7411	74LS11	74HC11	4073
	4	2	7421	74LS21	—	4082
OR	2	4	7432	74LS32	74HC32	4071
	3	3	—	—	74HC4075	4075
	4	2	—	—	—	4072
NAND	2	4	7400	74LS00	74HC00	4011
	3	3	7410	74LS10	74HC10	4013
	4	2	7420	74LS20	74HC20	4012
	8	1	7430	74LS30	—	4068
	12	1	74134	74LS134	—	—
	13	1	74133	74LS133	—	—
NOR	2	4	7402	74LS02	74HC02	4001
	3	3	7427	74LS27	74HC27	4025
	4	2	7425	74LS25	74HC4002	4002
	5	2	74260	74LS260	—	—
	8	1	—	—	—	4078

IEEE/IEC Standard Logic Symbols

Inverter:

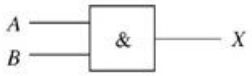


A	X
0	1
1	0

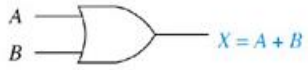


AND:

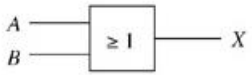
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



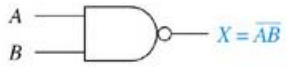
OR:



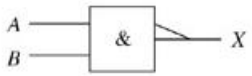
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1



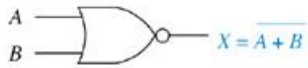
NAND:



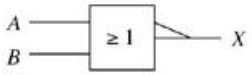
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0



NOR:



A	B	X
0	0	1
0	1	0
1	0	0
1	1	0



Diodes and Transistors

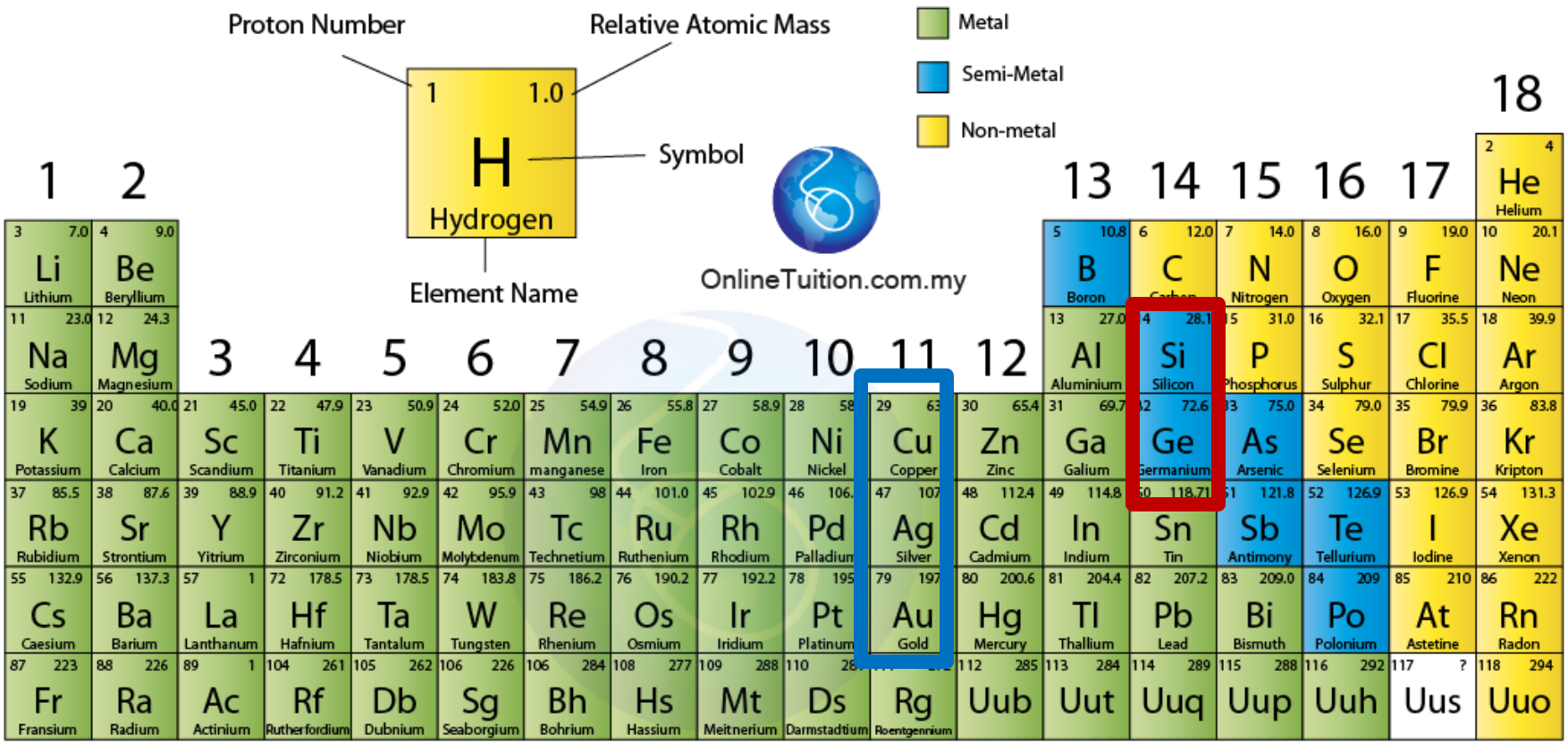
(Analog and Digital circuit elements)

Linear circuit elements

- Resistor, Inductor, Capacitor

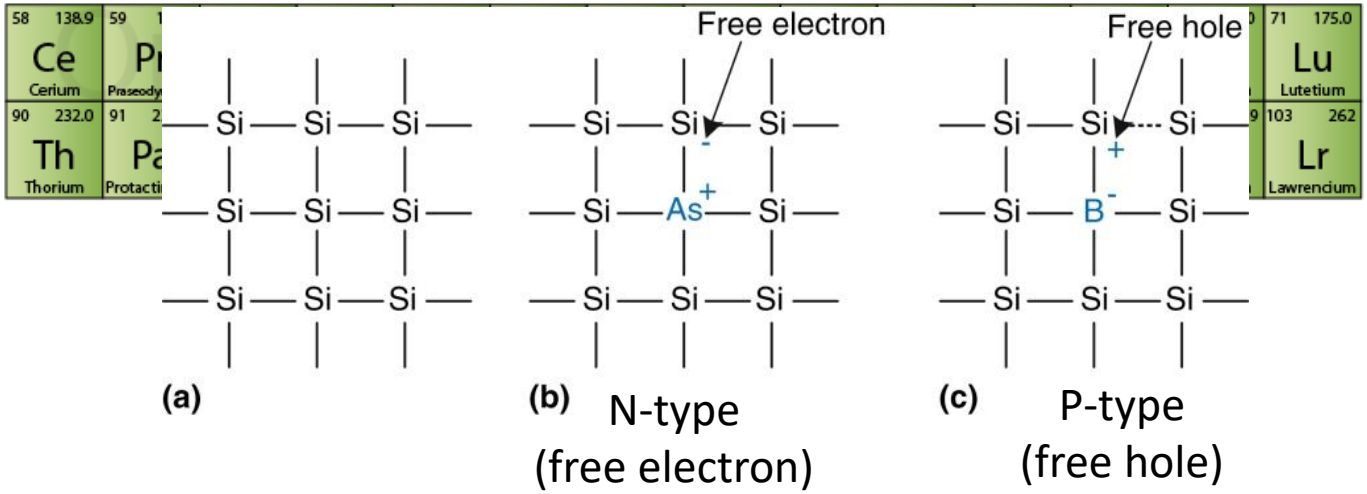
Nonlinear circuit elements

- Semiconductors:
  - conduct a little bit: carrying electric current, but not as easily as a normal conductor
  - i.e., silicon (Si, sand?) and Germanium (Ge)



Lanthanide

Actinide

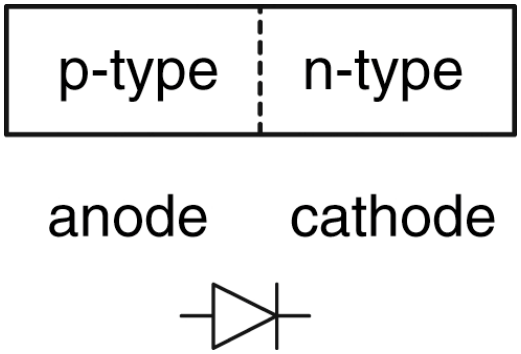


N-type
(free electron)

P-type
(free hole)

Non-Linear circuit elements

- **A diode**: the junction between p-type and n-type **silicon**
- Blocking current in one direction while letting current flow in another direction
  - Protecting devices if batteries are reversed: simply blocks any backward-biased current
- LED (Light Emitter Diode) is a type of diodes



Current – Voltage relationship $I = f(V)$ for two terminal devices.

Linear elements

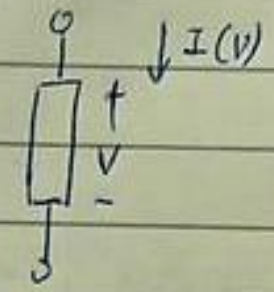
$I = V/R$

$I = C \cdot dV/dt$

$I = \frac{1}{L} \int V dt$

Nonlinear elements

$I = I_s [e^{V/nV_T} - 1]$



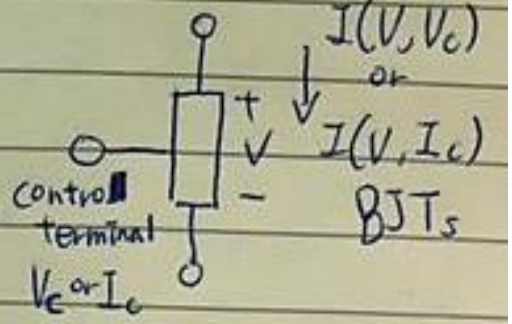
Transistors ⇒ adding a third terminal

① Type 1 : BJT

⇒ Current – controlled current flow.

② Type 2 : FETs

⇒ Voltage – controlled current flow



Transistors

- The word “transistor” is derived from the words “Transfer” and “Resistor”
- Three terminal component
  - i.e., Gate Source Drain / Collector Base Emitter
  - Transistor as a switch: Push button, + and – terminals
  - Transistor as an amplifier

Inventors of the Transistor

Born on three different continents (Brattain in Amoy, China; Bardeen in Madison, Wisconsin, USA; and Shockley in London, England), they all grew up in the United States and invented the transistor in 1947–1948 at Bell Telephone Laboratories. Brattain was an experimentalist while Bardeen and Shockley contributed more to the concepts and theories. Their reflections on that historic event:

"... after fourteen years of work, I was beginning to give up ..."

—Walter H. Brattain (1902–1987)

"Experiments that led to the invention of the point-contact transistor by Walter Brattain and me were done in November and December, 1947, followed closely by the invention of the junction transistor by Shockley."

—John Bardeen (1908–1991)

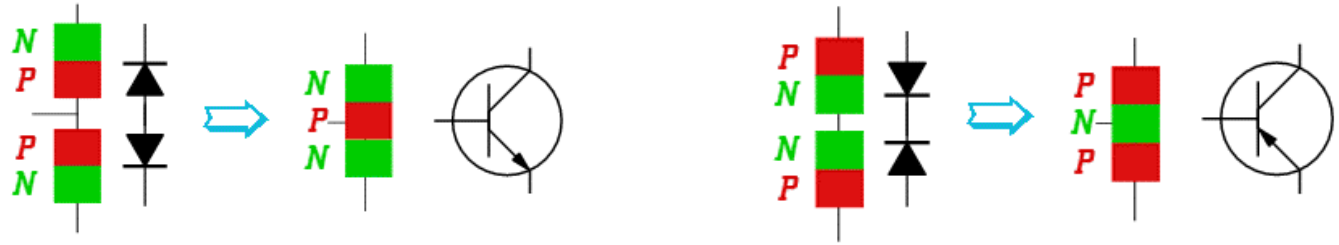
"All of us who were involved had no doubt that we had opened a door to a new important technology. "

—William B. Shockley (1910–1988)

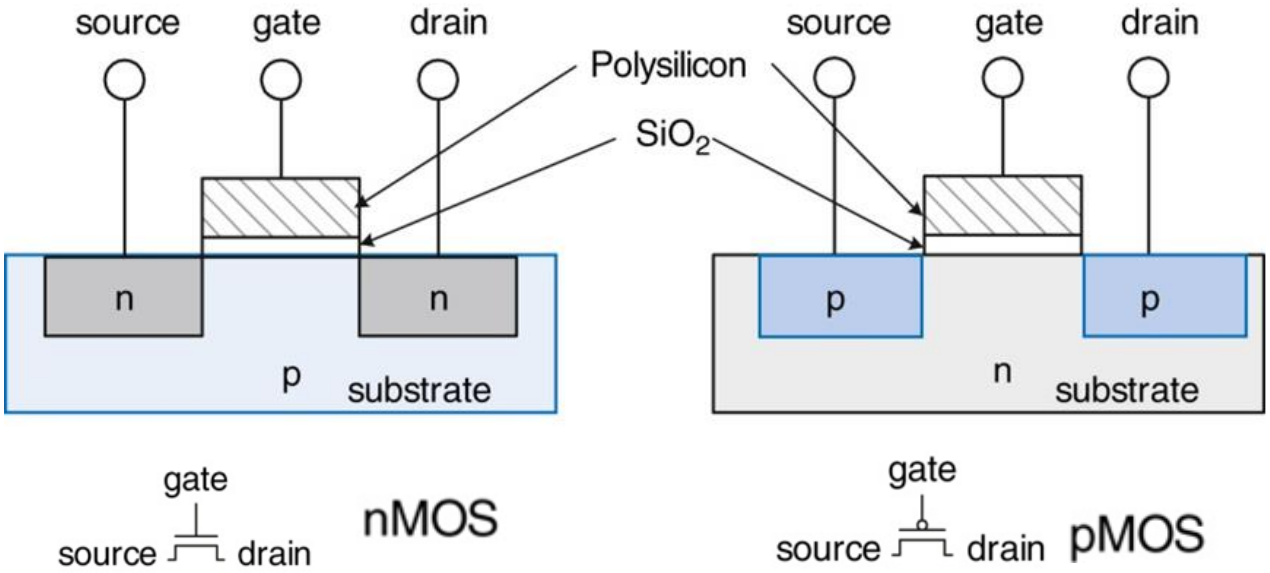
FIGURE 1–1 Transistor inventors John Bardeen, William Shockley, and Walter Brattain (left to right) at Bell Telephone Laboratories. (Courtesy of Corbis/Bettmann.)

More about semiconductor basic: holes and electrons:
http://www.allaboutcircuits.com/textbook/semiconductors/chpt-2/electrons-and-holes/

- BJT (Bipolar Junction Transistors)
  - Back to back diodes
  - https://youtu.be/7ukDKVHnac4 (0:43 - 5:53)



- MOSFET (Metal Oxide Silicon Field Effect Transistor)
  - https://youtu.be/stM8dgcY1CA (0:33 ~)



Logic Family Examples

Logic Family	V_{DD}	V_{IL}	V_{IH}	V_{OL}	V_{OH}
TTL	5 (4.75 - 5.25)	0.8	2.0	0.4	2.4
CMOS	5 (4.5 - 6)	1.35	3.15	0.33	3.84
LVTTL	3.3 (3 - 3.6)	0.8	2.0	0.4	2.4
LVCMOS	3.3 (3 - 3.6)	0.9	1.8	0.36	2.7

LS series, BJT

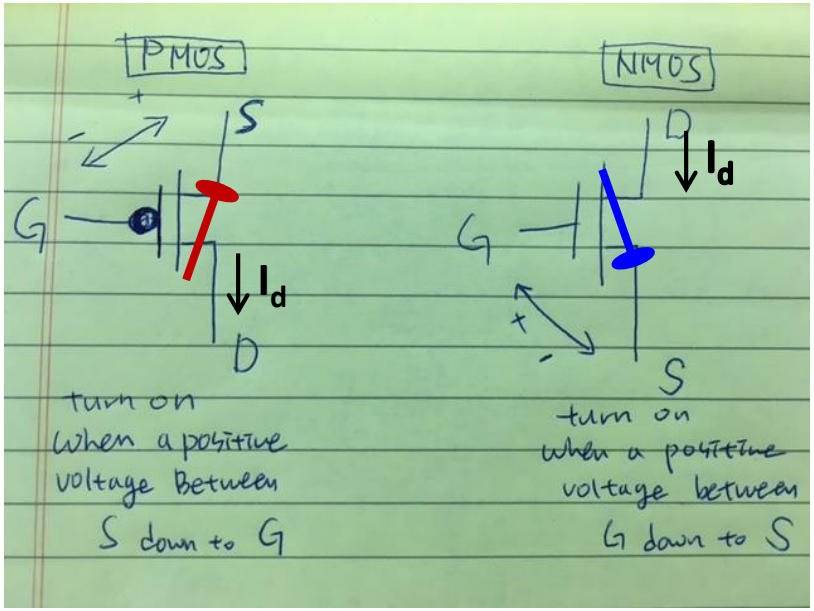
HC series, MOSFET, CMOS

Compatibility of Logic Families

		Receiver			
		TTL	**CMOS**	**LVTTL**	**LVCMOS**
Driver (Sender)	**TTL**	OK	NO ($V_{OH}<V_{IH}$)	Maybe	Maybe
	CMOS	OK	OK	Maybe	Maybe
	LVTTL	OK	NO ($V_{OH}<V_{IH}$)	OK	OK
	LVCMOS	OK	NO ($V_{OH}<V_{IH}$)	OK	OK

More on MOSFETs

- Today, most transistors are of the MOSFET type
- Gate/Source/Drain : P channel, N channel
- In terms of analysis, a voltage controlled device
  - A voltage input to the gate controls the flow of current from source to drain (V_{gs} is used to control I_d)



What is a drain?

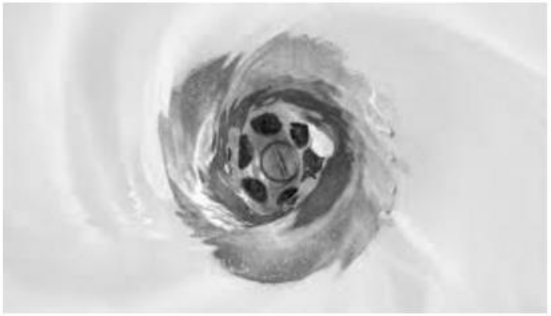








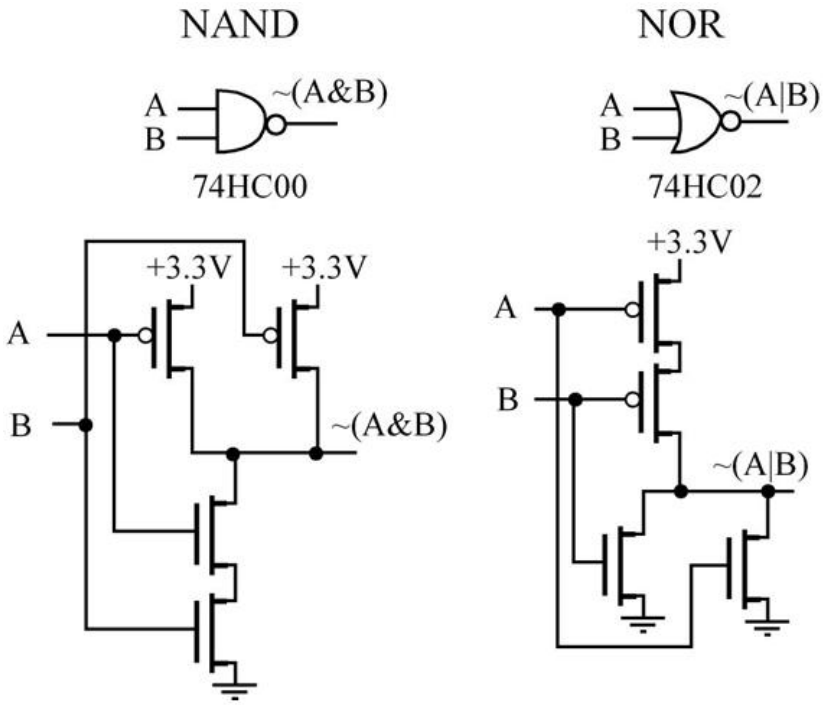


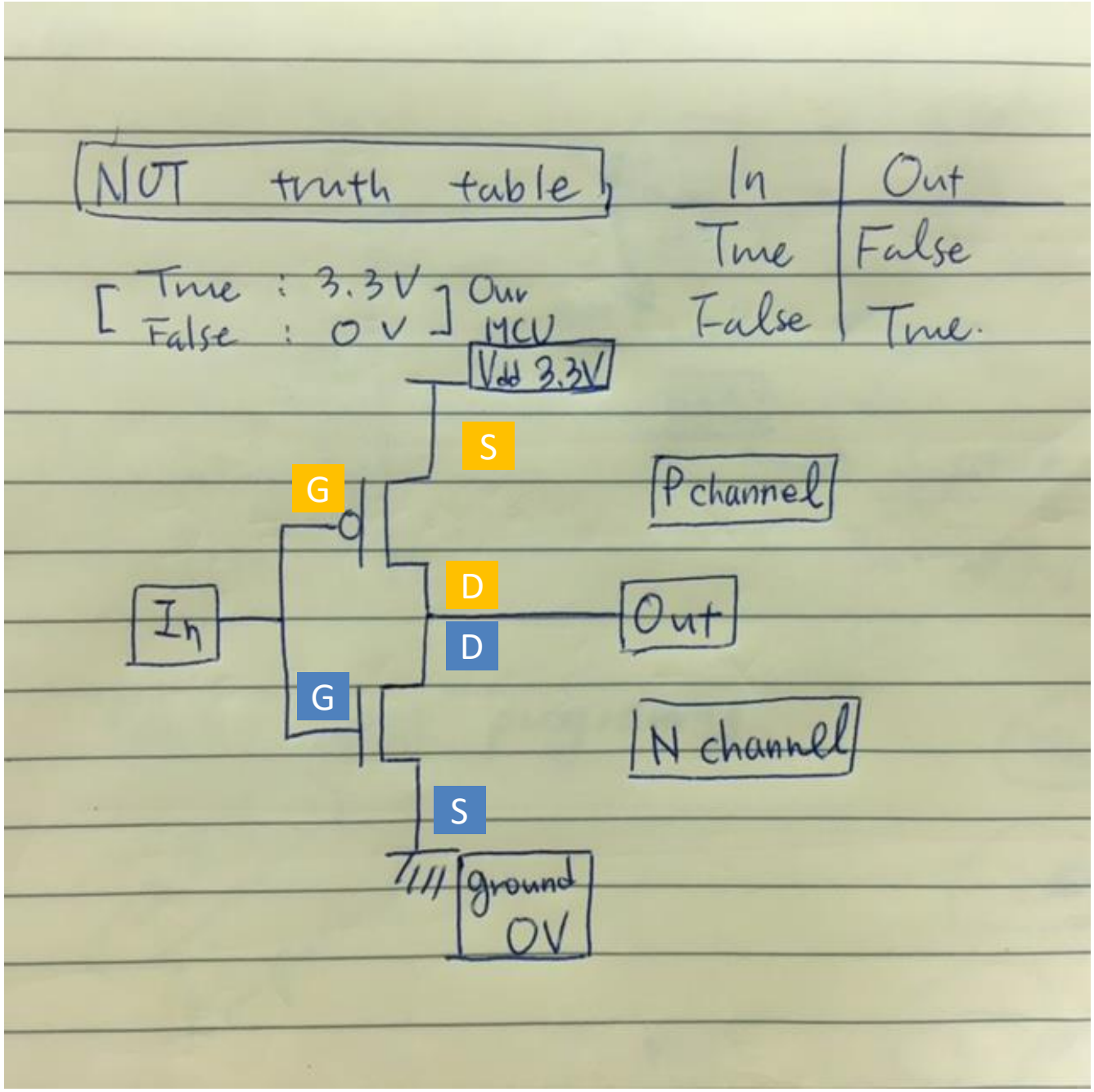


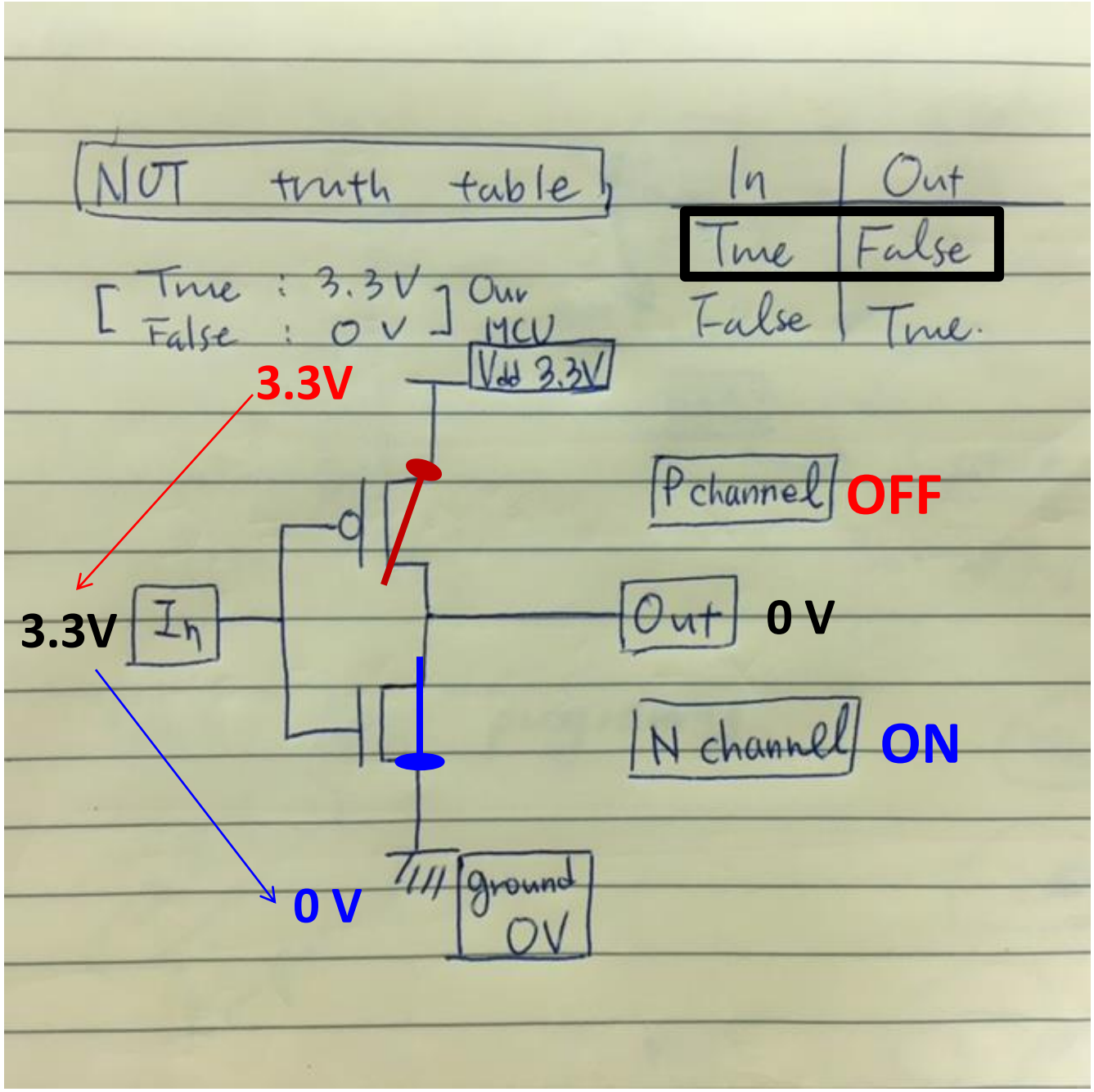

Logic gate

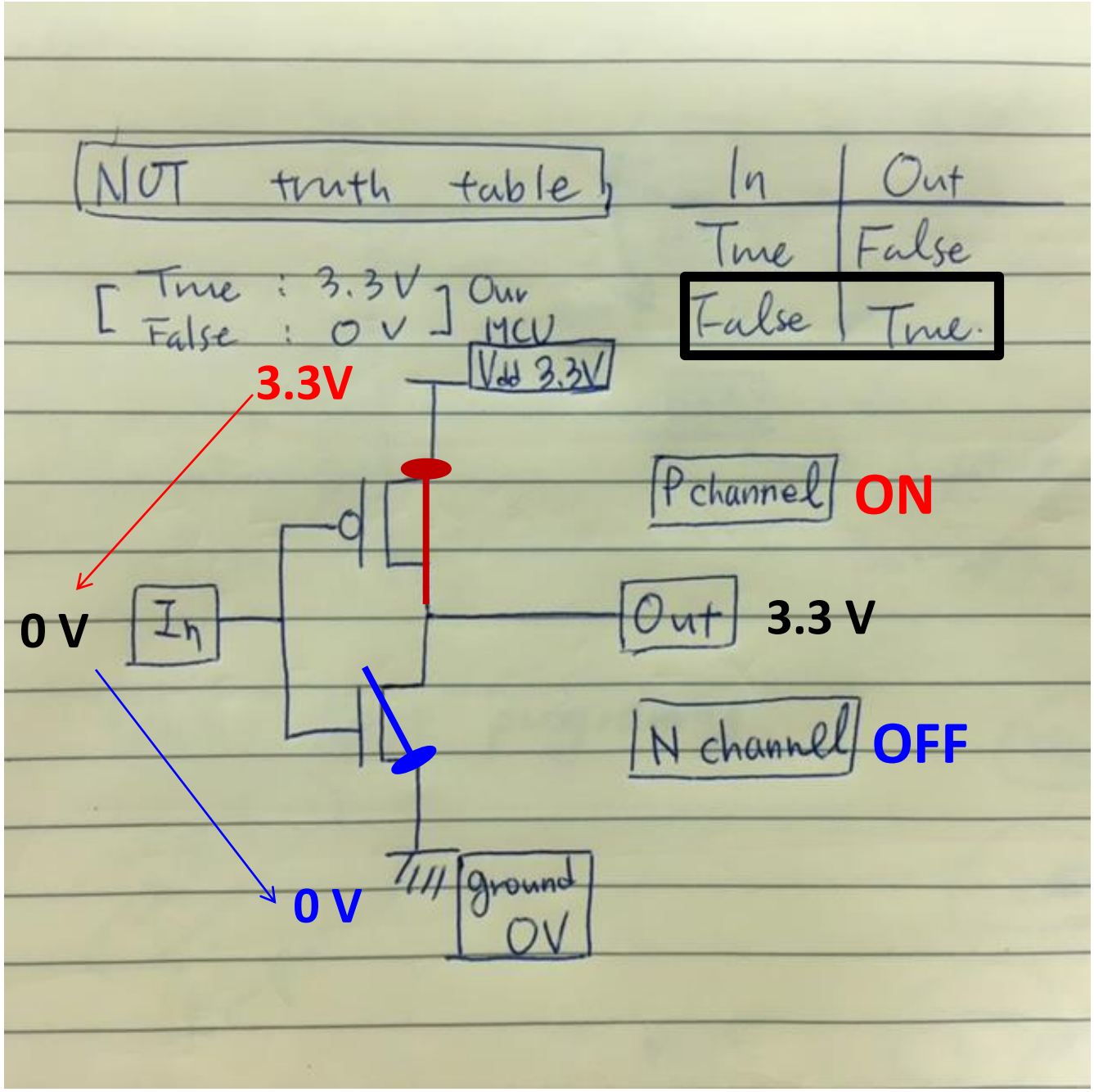
- An idealized or physical device implementing a Boolean function
  - Performs a logical operation on one or more logical inputs
  - Produces a single logical output
- **Basic building elements: transistors**
  - In modern practice, most gates are made from field-effect transistors (FETs), particularly **MOSFETs**.

- AND, NOR, OR NOT and etc
- An important logic to understand registers: a NOT gate so called an invertor
  - Basic element to represent a bit




NOT truth table
[True : 3.3V
False : 0 V] Our MCU
In | Out
True | False
False | True.
Vdd 3.3V
S
G
Pchannel
D
In
Out
D
G
N channel
S
ground
0V


NOT truth table
In | Out
True | False
False | True.
[True : 3.3V
False : 0 V] Our MCU
Vdd 3.3V
3.3V
Pchannel
OFF
3.3V
In
Out
0 V
N channel
ON
0 V
ground
0V



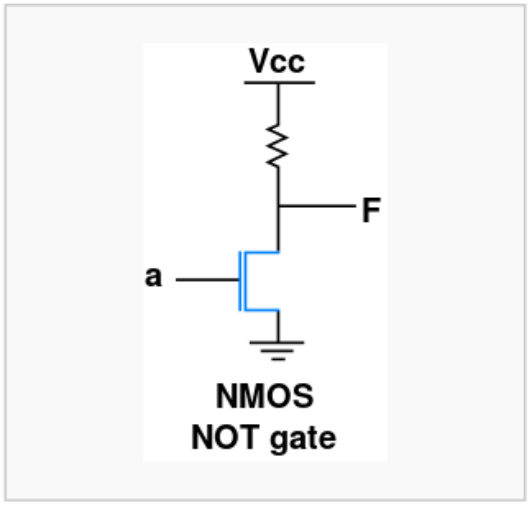
A basic element to represent a bit

This is a NOT symbol

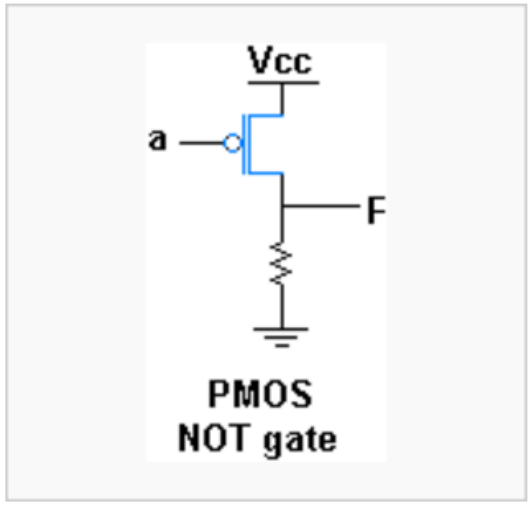
Or an inverter

other NOT gate implementations

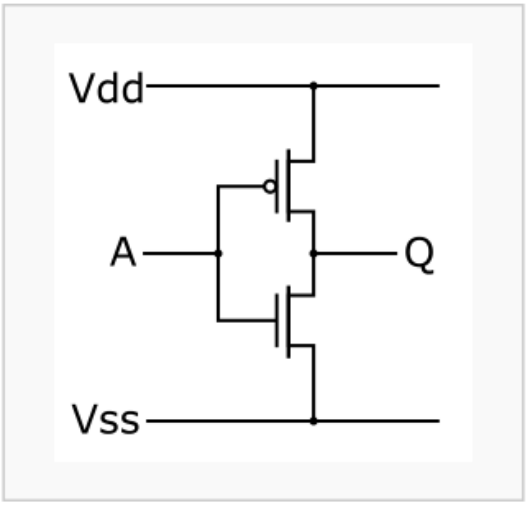
Electronic implementation [edit]



NMOS inverter



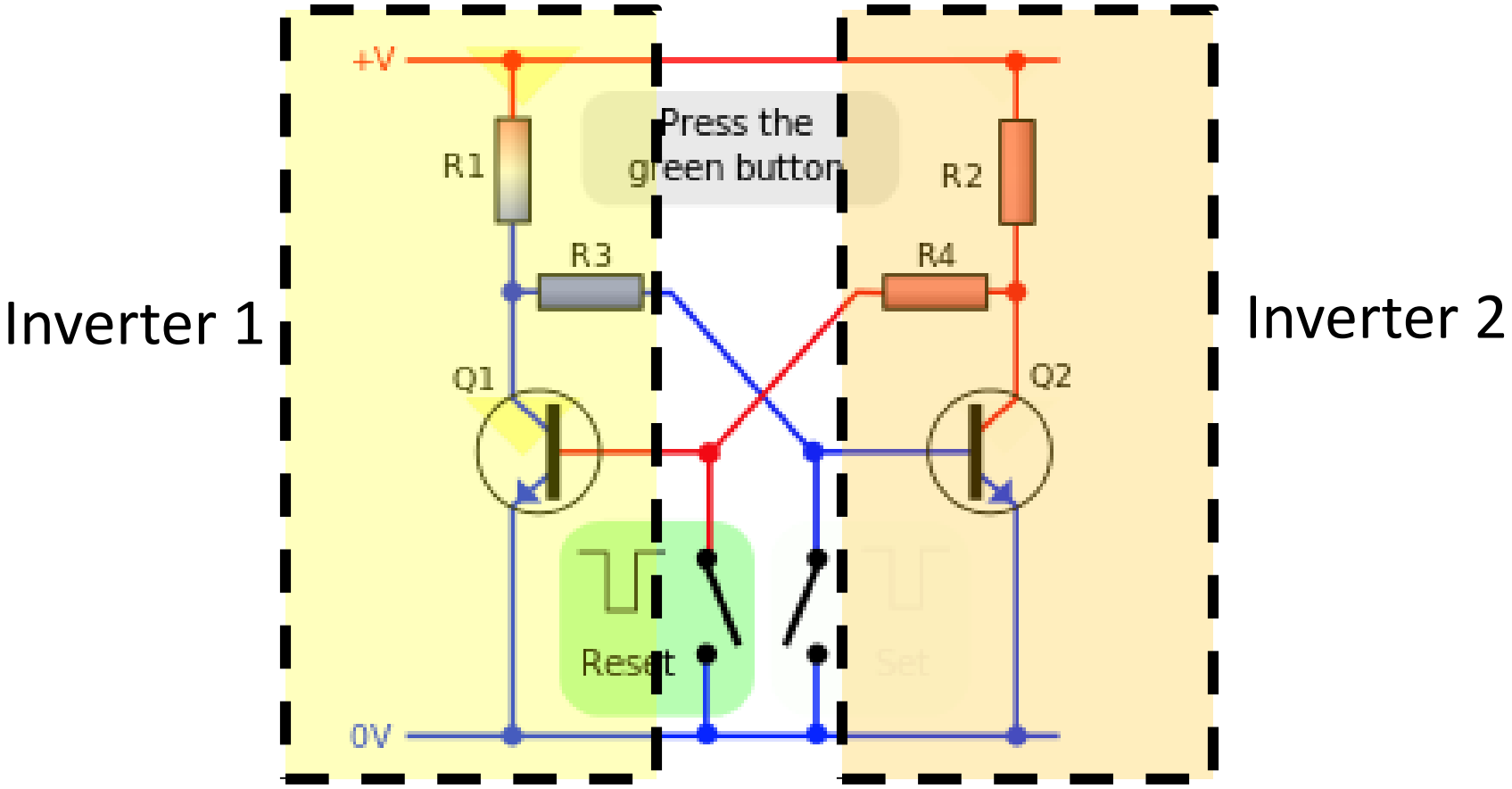
PMOS inverter



Static CMOS inverter

Latch

- Basic element: Two inverters

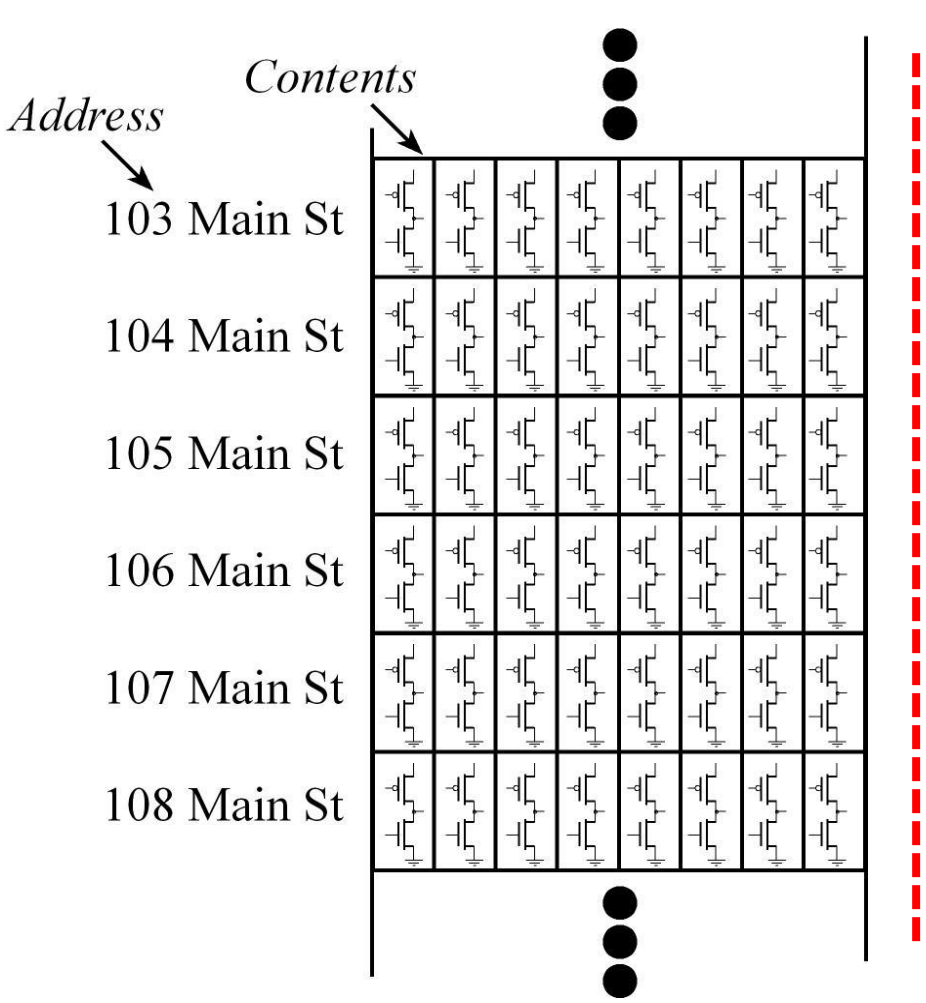


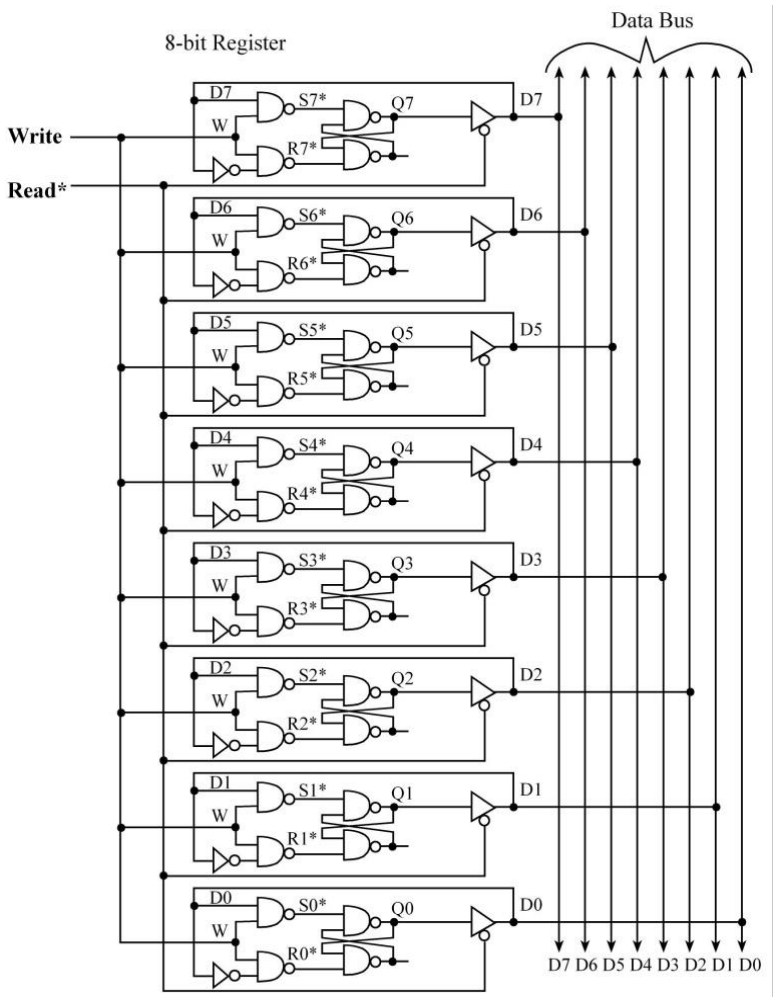
- A bistable multivibrator

Flip flop

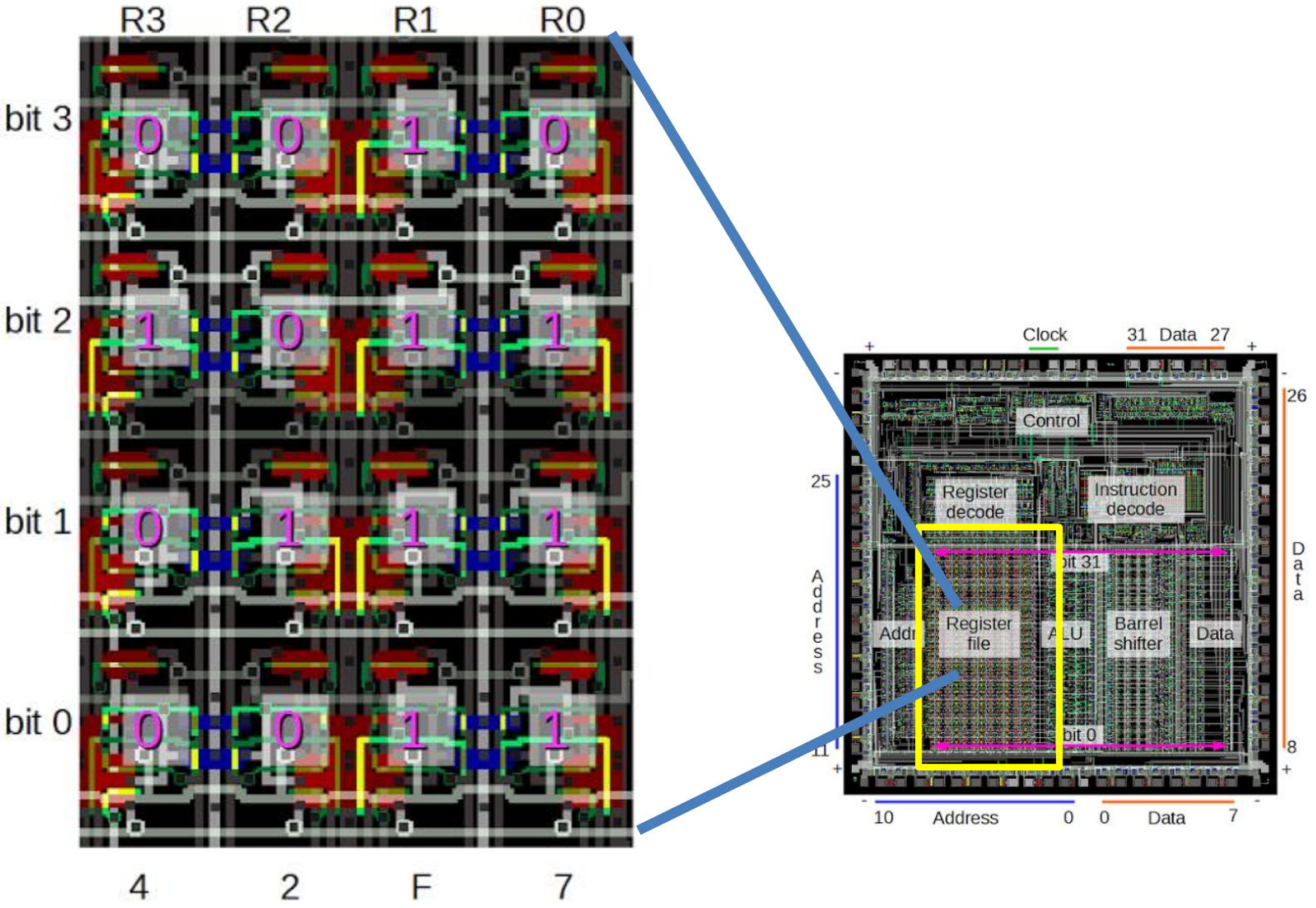
- Basic storage elements for todays registers and memories
  - bistable mutivibrator
  - feedback path
- Basic element: two latches (master and slave) with a forcing switch (i.e., timer)
  - Synchronous devices
  - operate on clock edges
- Example: D flip flop, SR flip flop

Logic implementation of memories and registers (not exact, but ST like this)



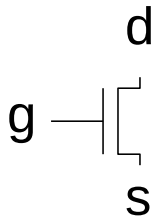


- To create the register file, the register cell above is repeated 32 times vertically for each bit

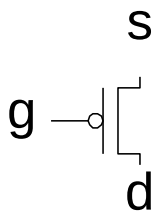


Transistor Function

nMOS

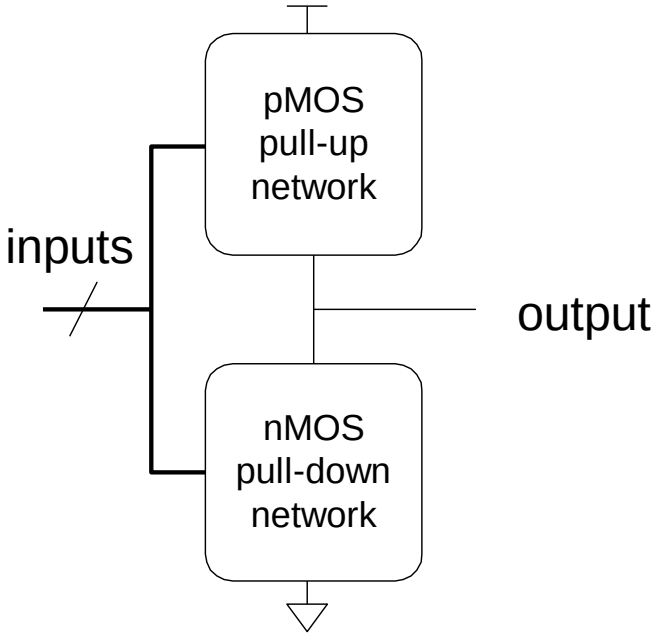


pMOS



MOSFET Functions

- **nMOS**: pass good **0**'s, so connect source to GND
  (not good at passing low voltage)
- **pMOS**: pass good **1**'s, so connect source to V_{DD}
  (not good at passing high voltage)

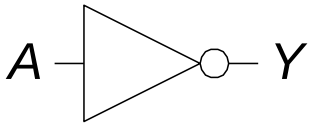


➔CMOS (Complementary metal oxide semiconductor) Gate Structure

1. First, understand CMOS principle by means of “Switch models” of circuits
2. Then, apply tricks to solve questions.

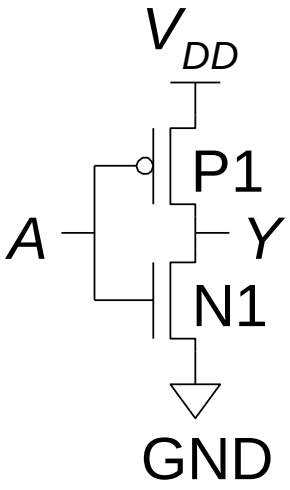
CMOS Gates: NOT Gate

NOT



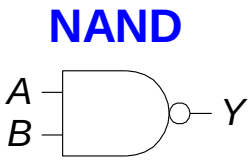
$Y = \overline{A}$

A	Y
0	1
1	0



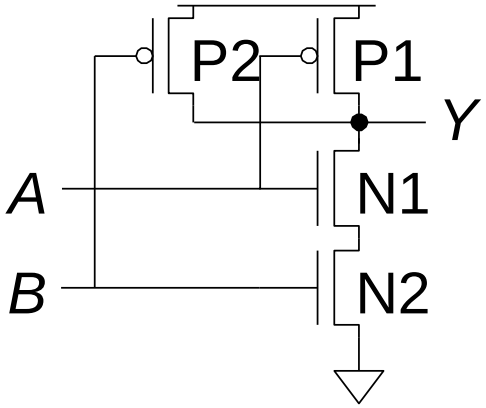
A	P1	N1	Y
0	On or Off		1
1			0

CMOS Gates: NAND Gate



$Y = \overline{AB}$

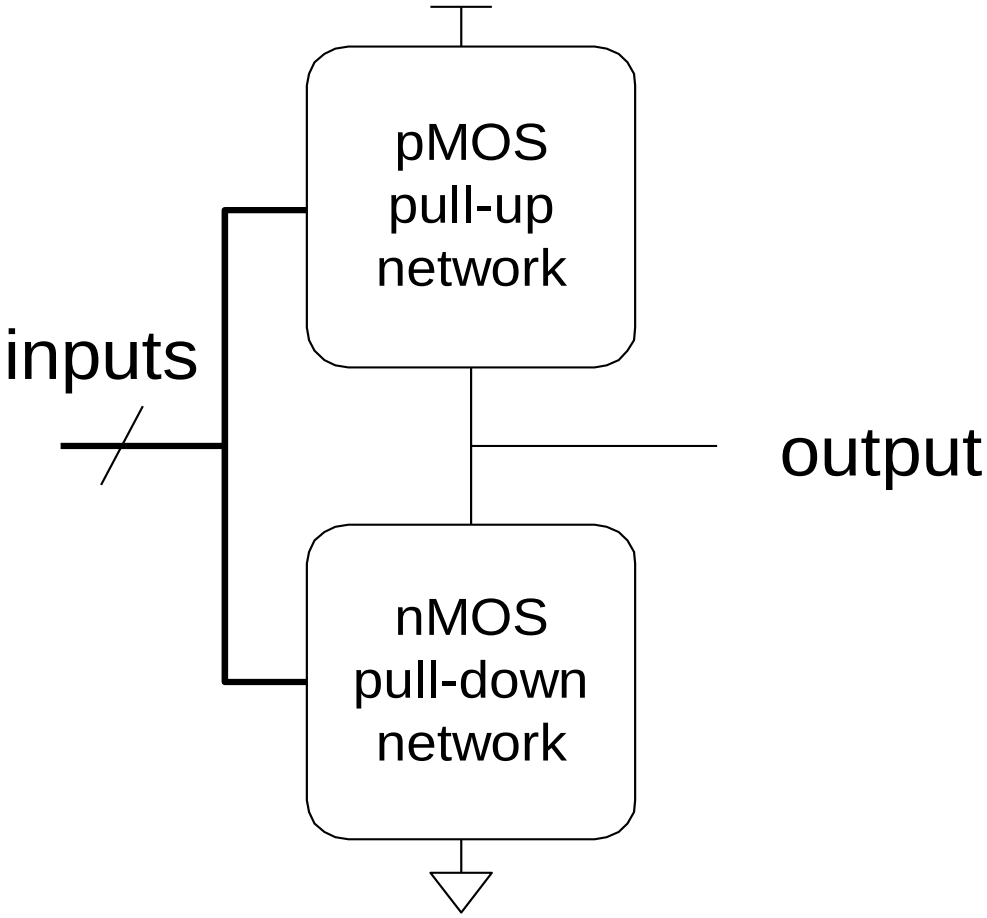
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



A	B	P1	P2	N1	N2	Y
0	0	On or Off				1
0	1					1
1	0					1
1	1					0

More EXAMPLES for your try

Keep in your mind:
CMOS Gate Structure



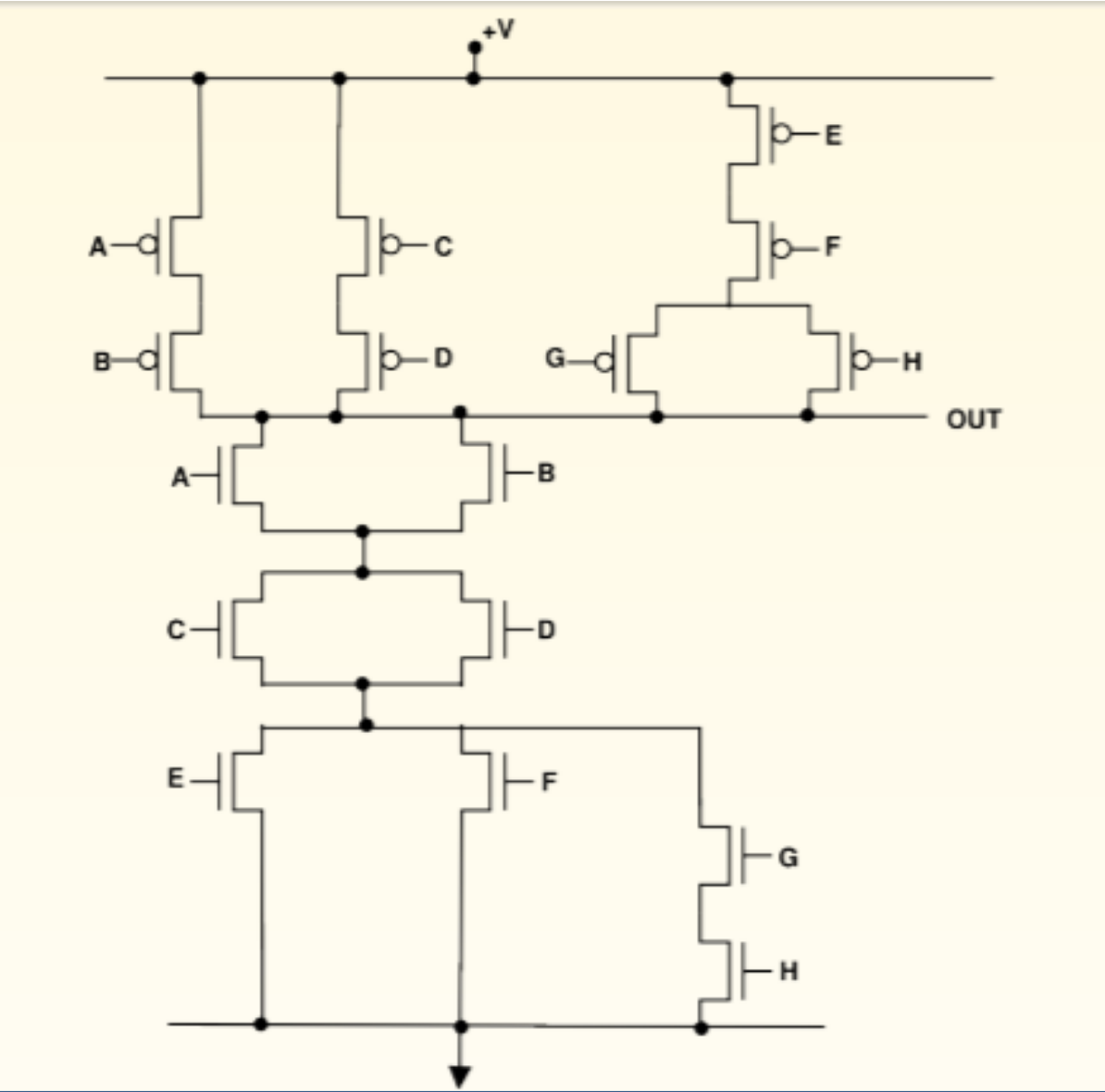
Example: $F = \overline{(A \cdot B) + (C \cdot D)}$

1. Take un-inverted function $F = (A \cdot B) + (C \cdot D)$ and derive N-network
2. Identify **AND (series), OR (parallel)** components: F is OR of AB, CD
3. Make connections of transistors

4. Construct P-network by taking complement of N-expression (AB + CD), which gives the expression, $(A + B) \cdot (C + D)$
5. Combine P and N circuits

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

You Try!!!



You Try!!!

How do you build a three-input NOR gate?