Chapter 1 :: From Zero to One

***Digital Design and Computer Architecture*, 2nd Edition**

David Money Harris and Sarah L. Harris 저

Chapter 1 :: Topics

- Background
- The Game Plan
- The Art of Managing Complexity
- The Digital Abstraction
- Number Systems
- Logic Gates
- Logic Levels
- CMOS Transistors
- Power Consumption

Review

- What did you learn at last lesson?
  - Number Systems(수의 체계)
  - Number Conversion
  - Hexadecimal Numbers (16진수)
  - Hexadecimal to Binary Conversion
  - Bits, Bytes, Nibbles…
  - Powers of Two (거듭제곱의 계산)
  - Overflow (오버플로우)
  - Sign/Magnitude Numbers (부호/크기 표기법)
  - Two's Complement Numbers
  - Sign-extension / Zero-extension

1.5 Logic Gates

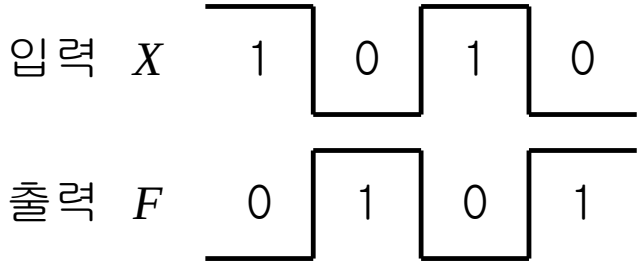
- Perform logic functions:
  - inversion (NOT), AND, OR, NAND, NOR, etc.
- Single-input:
  - NOT gate, buffer
- Two-input:
  - AND, OR, XOR, NAND, NOR, XNOR
- Multiple-input

Single-Input Logic Gates (NOT)

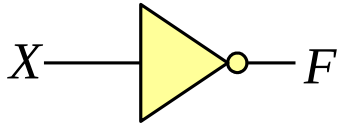
- 한 개의 입력과 한 개의 출력을 갖는 게이트로서 논리 부정을 나타낸다.

X	F
0	1
1	0

진리표

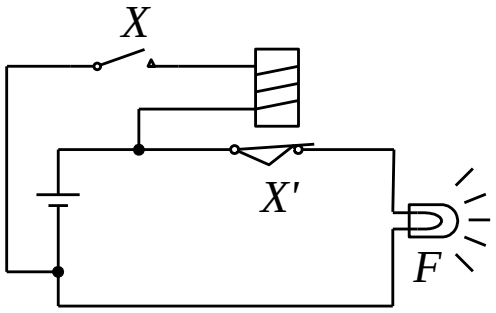


동작파형

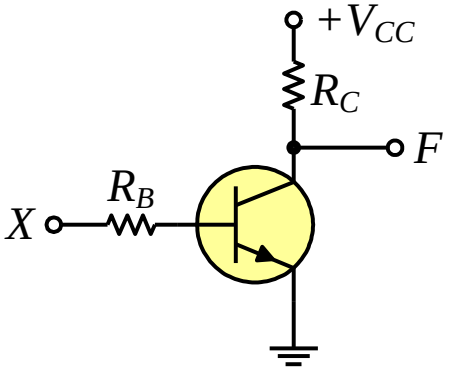


논리회로 기호

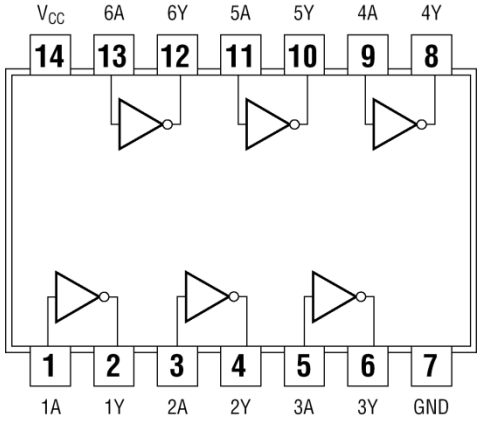
논리식 $F = \overline{X} = X'$



스위칭 회로



트랜지스터 회로



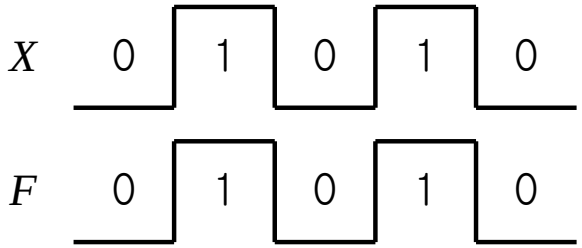
IC 7404

Single-Input Logic Gates (BUF)

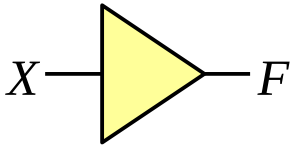
- 버퍼(buffer)는 입력된 신호 그대로를 출력하는 게이트로서 단순한 전송을 의미한다.

X	F
0	0
1	1

진리표



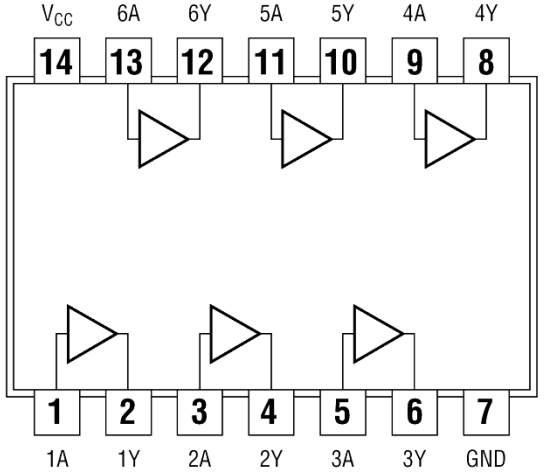
동작파형



논리 기호

$$F = X$$

논리식



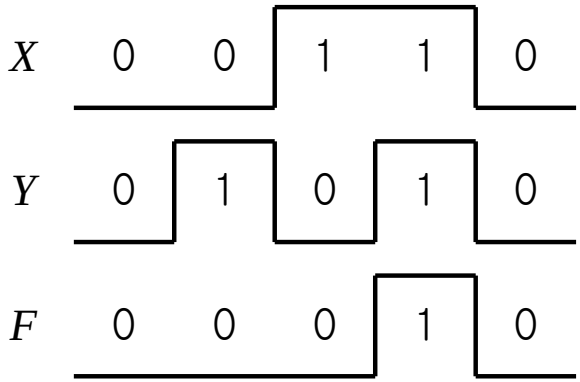
IC 7407 핀 배치도

Two-Input Logic Gates (AND)

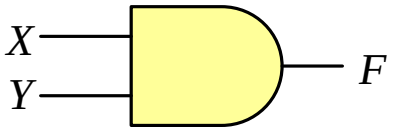
- 입력이 모두 1(on)인 경우에만 출력은 1(on)이 되고,
- 입력 중에 0(off)인 것이 하나라도 있을 경우에는 출력은 0(off)이 된다.

X	Y	F
0	0	0
0	1	0
1	0	0
1	1	1

진리표



동작파형



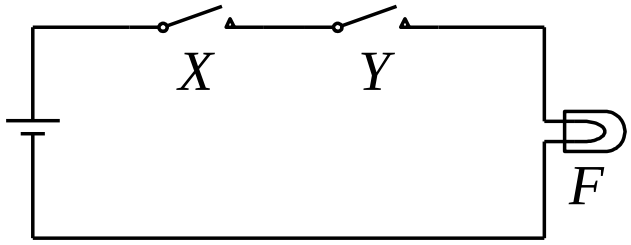
논리회로 기호

$$F = XY = X \cdot Y$$

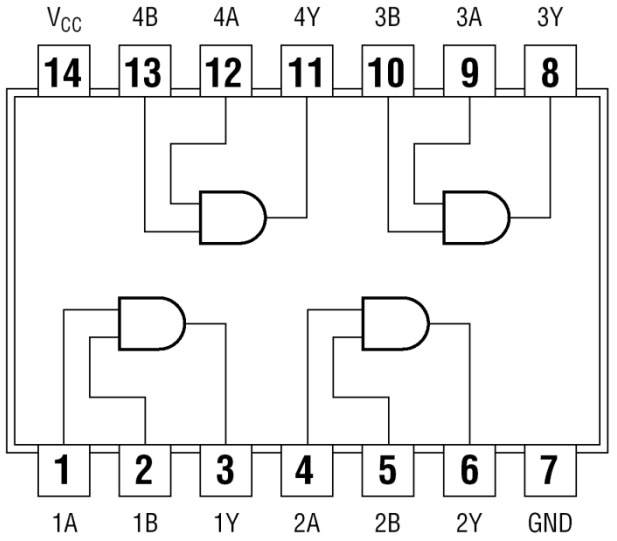
논리식

Two-Input Logic Gates (AND)

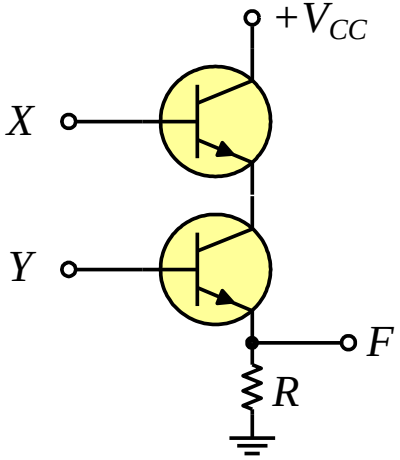
- AND 게이트의 회로 표현과 IC



스위칭 회로



IC 7408



트랜지스터 회로

3-Input Logic Gates (AND)

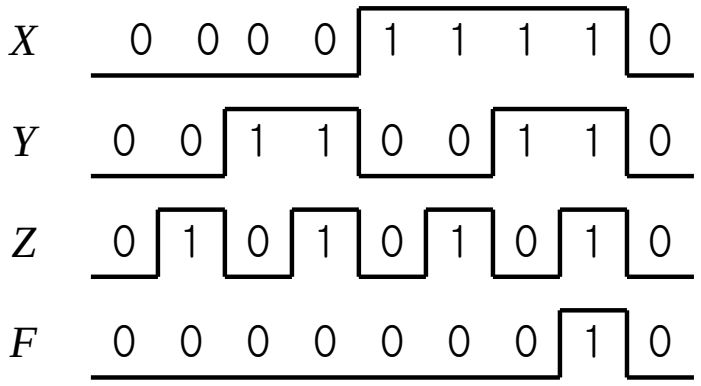
- AND 게이트의 기본 개념(3입력)

X	Y	Z	F
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

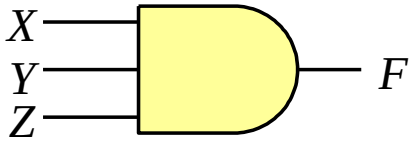
진리표

$$F = XYZ = X \cdot Y \cdot Z$$

논리식



동작파형



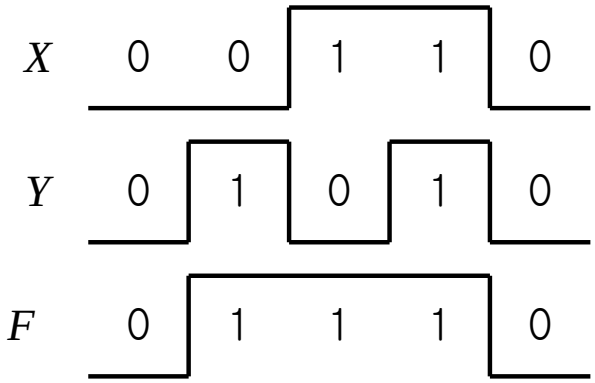
논리회로 기호

Two-Input Logic Gates (OR)

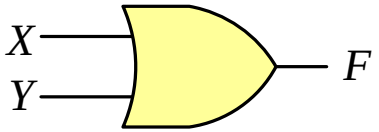
- 입력이 모두 0인 경우에만 출력은 0 이 되고,
- 입력 중에 1이 하나라도 있으면, 출력은 1이 된다.

X	Y	F
0	0	0
0	1	1
1	0	1
1	1	1

진리표



동작파형



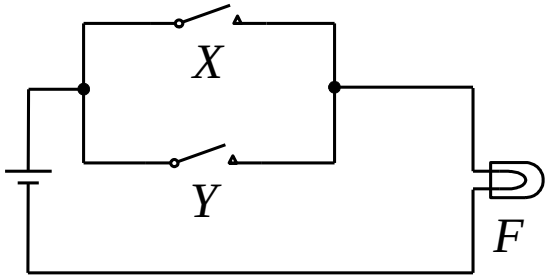
논리회로 기호

$$F = X + Y$$

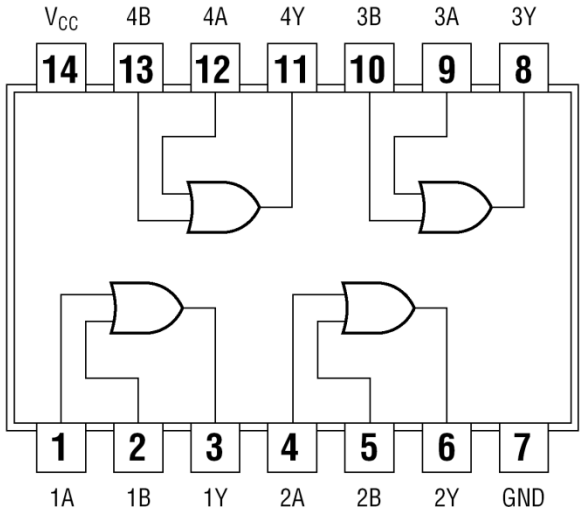
논리식

Two-Input Logic Gates (OR)

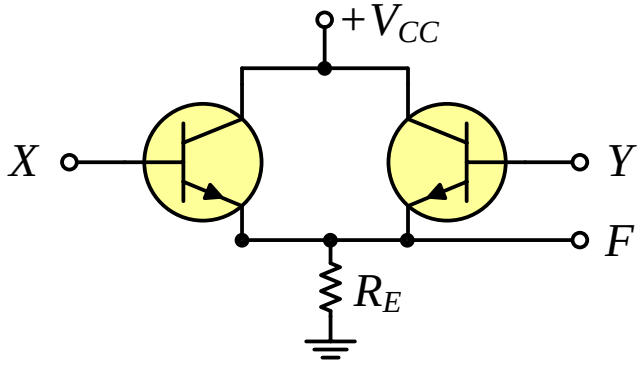
- OR 게이트의 회로 표현과 IC



스위칭 회로



IC 7432



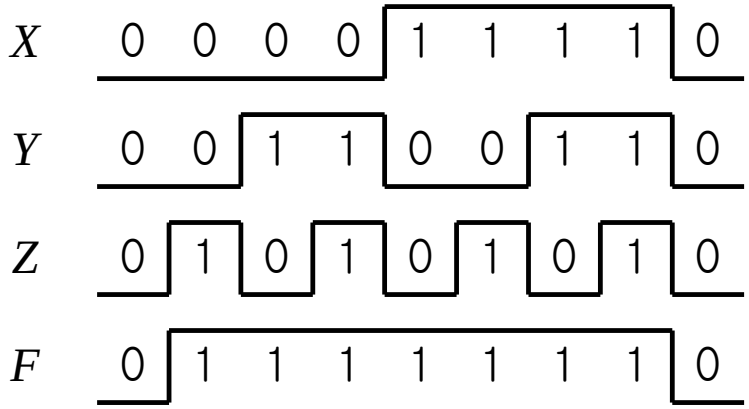
트랜지스터 회로

3-Input Logic Gates (OR)

- OR 게이트의 기본 개념(3입력)

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

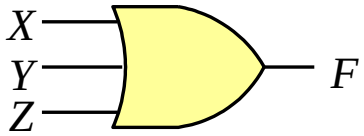
진리표



동작파형

$$F = X + Y + Z$$

논리식



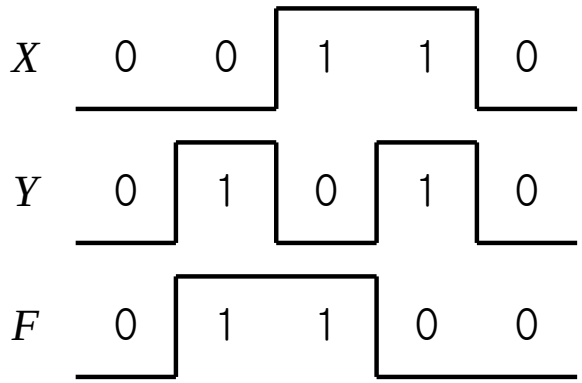
논리회로 기호

Two-Input Logic Gates (XOR)

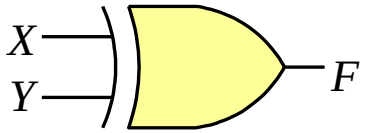
- XOR 게이트의 기본 개념(2입력)
  - 입력 중 홀수 개의 1이 입력된 경우에 출력은 1이 되고
  - 그렇지 않은 경우에는 출력은 0 이 된다.

X	Y	F
0	0	0
0	1	1
1	0	1
1	1	0

진리표



동작파형

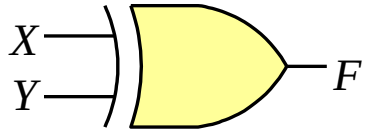


논리회로 기호

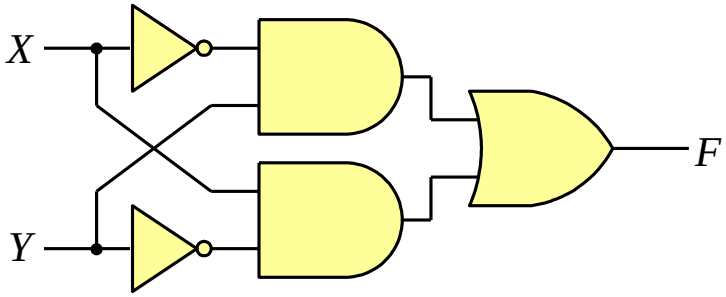
$$F = X \oplus Y = \overline{X}Y + X\overline{Y}$$

논리식

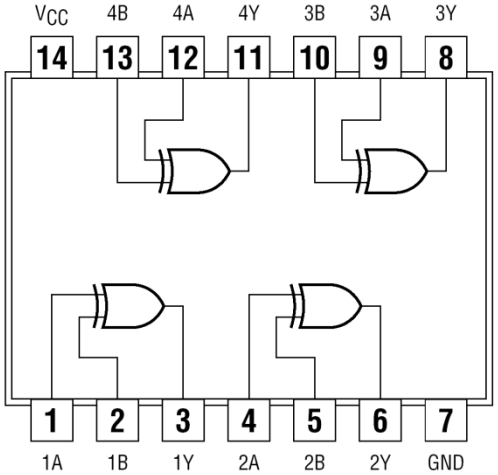
Two-Input Logic Gates (XOR)



논리회로 기호



AND-OR 게이트로 표현



IC 7486

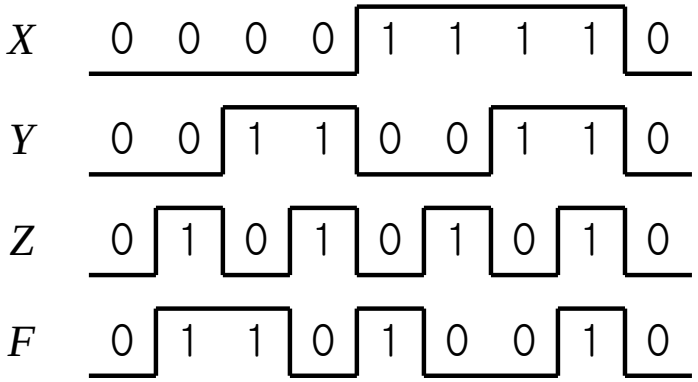
$$F = X \oplus Y = \overline{X}Y + X\overline{Y}$$

논리식

3-Input Logic Gates (XOR)

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

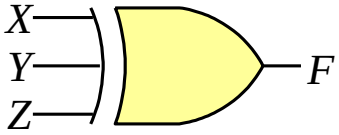
진리표



동작파형

$$F = X \oplus Y \oplus Z$$

논리식

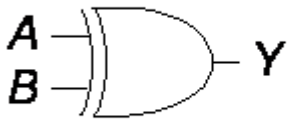


논리회로 기호

Two-Input Logic Gates (XNOR)

- 출력값은 XOR 게이트에 NOT 게이트를 연결한 것이므로
- XOR 게이트와 반대이다.

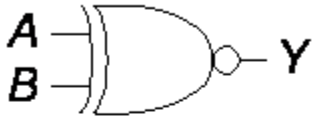
XOR



$Y = A \oplus B$

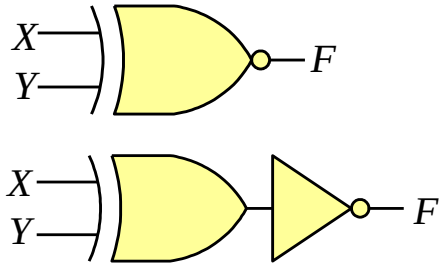
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

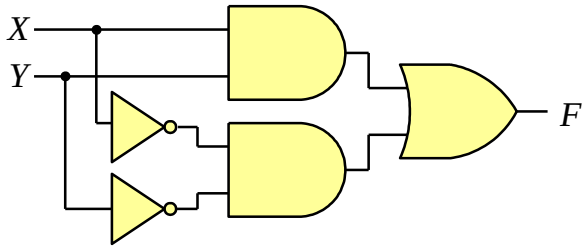
XNOR



$Y = \overline{A \oplus B}$

A	B	Y
0	0	
0	1	
1	0	
1	1	





논리회로 기호

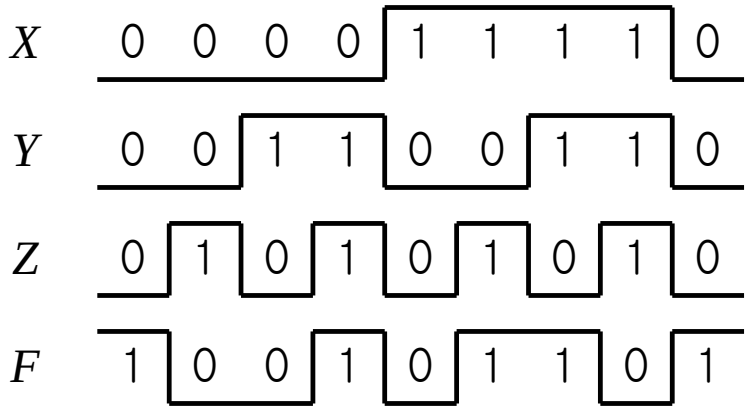
$$F = \overline{X}\,\overline{Y} + XY = \overline{X \oplus Y} = X \odot Y$$

논리식

3-Input Logic Gates (XNOR)

X	Y	Z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

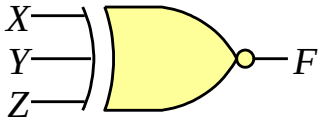
진리표



동작파형

$$F = \overline{X \oplus Y \oplus Z} = X \odot Y \odot Z$$

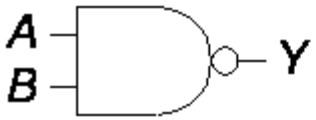
논리식



논리회로 기호

More Two-Input Logic Gates

NAND

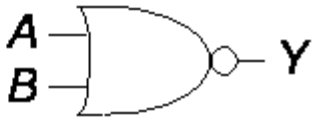


$Y = \overline{AB}$

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

Y=(AB)’

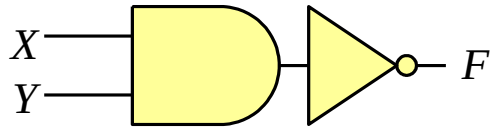
NOR

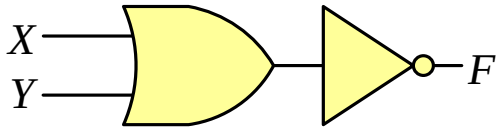


$Y = \overline{A + B}$

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

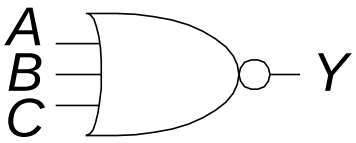
Y=(A+B)’





Multiple-Input Logic Gates

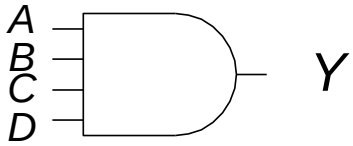
NOR3



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

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

AND4



$Y = ABCD$

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

Logic Levels

전압레벨	정 논리	
+5V	High=1	
0V	Low=0	

X	*Y*	*F*
L	*L*	*L*
L	*H*	*L*
H	*L*	*L*
H	*H*	*H*

전압레벨

X	*Y*	*F*
0	0	0
0	1	0
1	0	0
1	1	1

정논리 AND

Logic Levels

- Define discrete voltages to represent 1 and 0
- For example, we could define:
  - 0 to be *ground* or 0 volts
  - 1 to be V_{DD} or 5 volts

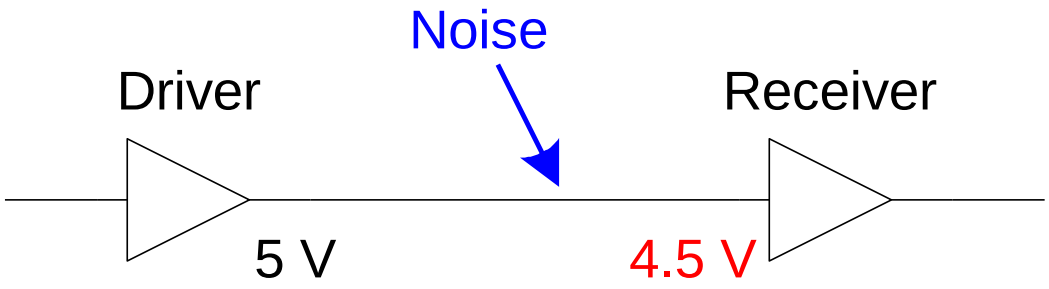
- What about 4.99 volts? Is that a 0 or a 1?
- What about 3.2 volts?

- Define a *range* of voltages to represent 1 and 0
- Define different ranges for outputs and inputs to allow for *noise* in the system

What is Noise?

- Noise is anything that degrades the signal
  - E.g., resistance, power supply noise, coupling to neighboring wires, etc.

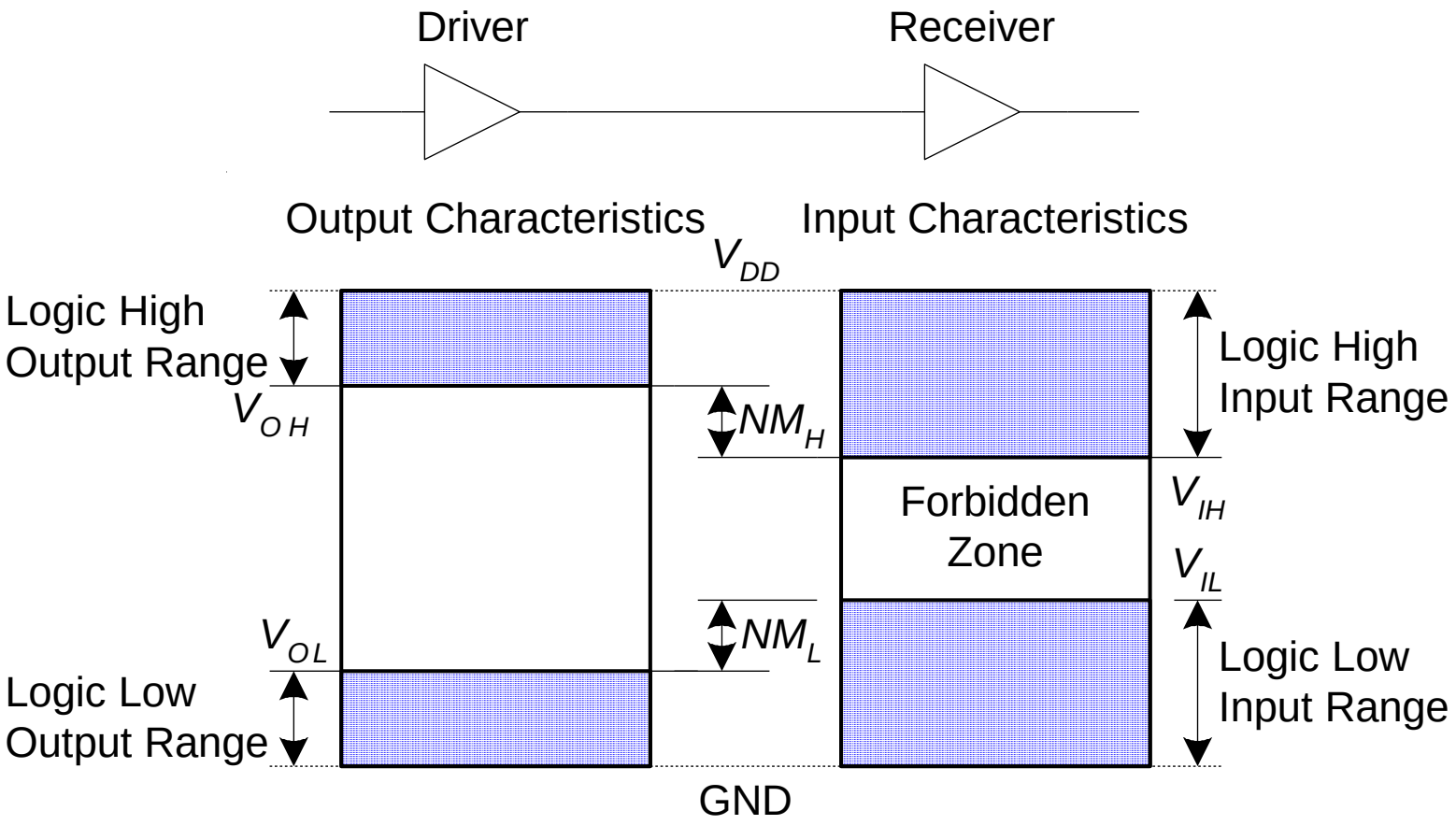
- Example : a gate (driver) could output a 5 volt signal but, because of losses in the wire and other noise, the signal could arrive at the receiver with a degraded value, for example, 4.5 volts



The Static Discipline

- With logically valid inputs, every circuit element must produce logically valid outputs
- Use limited ranges of voltages to represent discrete values

Noise Margins (잡음여유도)

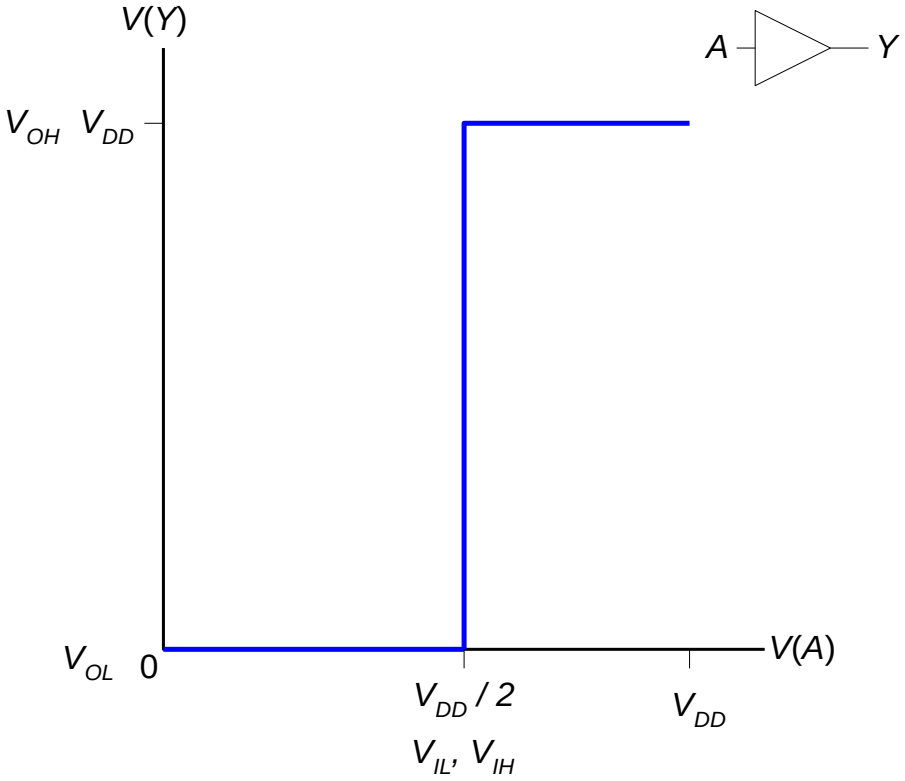


$$NM_H = V_{OH} - V_{IH}$$

$$NM_L = V_{IL} - V_{OL}$$

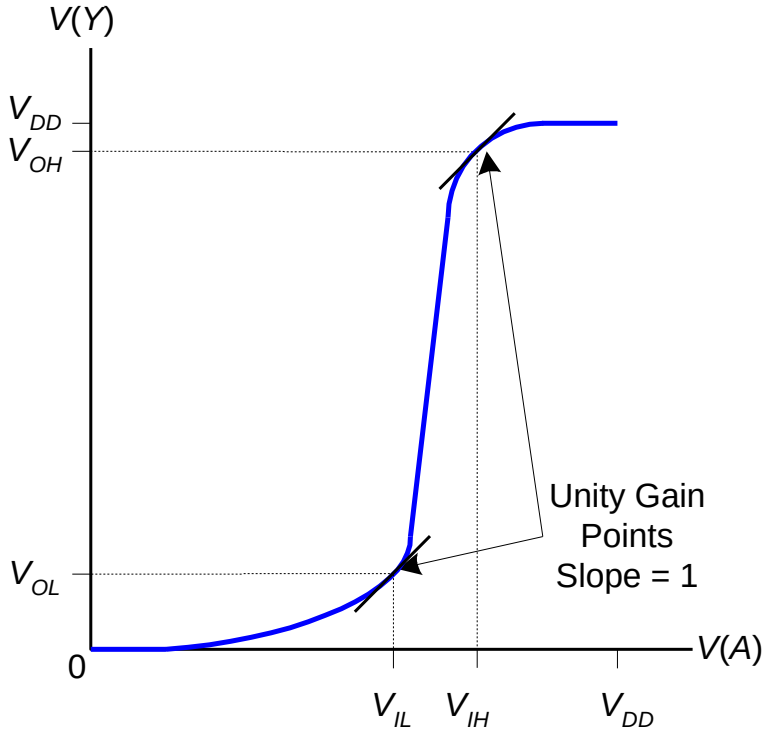
DC Transfer Characteristics

Ideal Buffer:



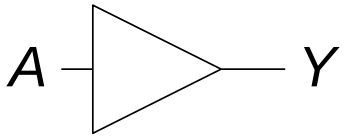
$NM_H = NM_L = V_{DD}/2$

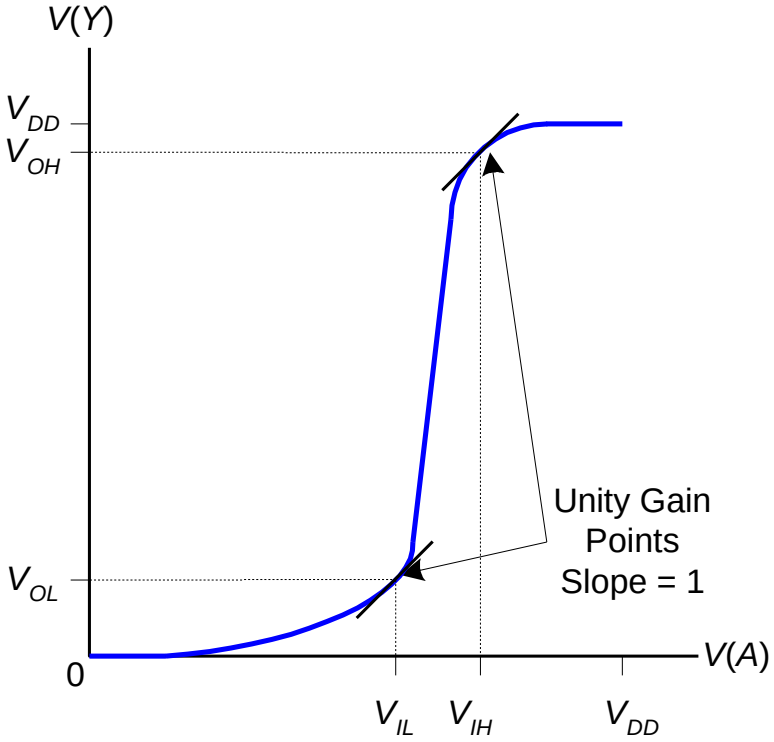
Real Buffer:

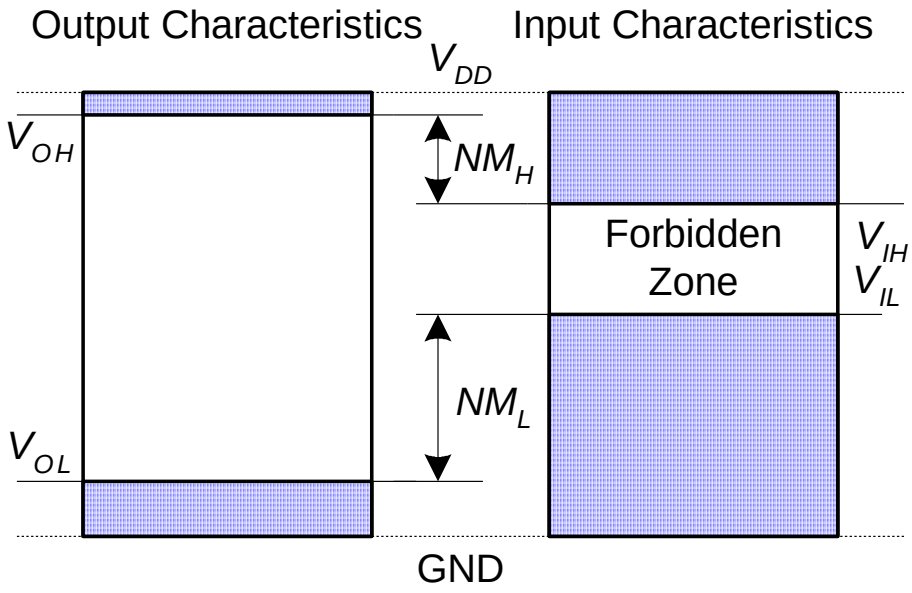


$NM_H , NM_L < V_{DD}/2$

DC Transfer Characteristics







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

Chips operate because they contain magic smoke

Proof:

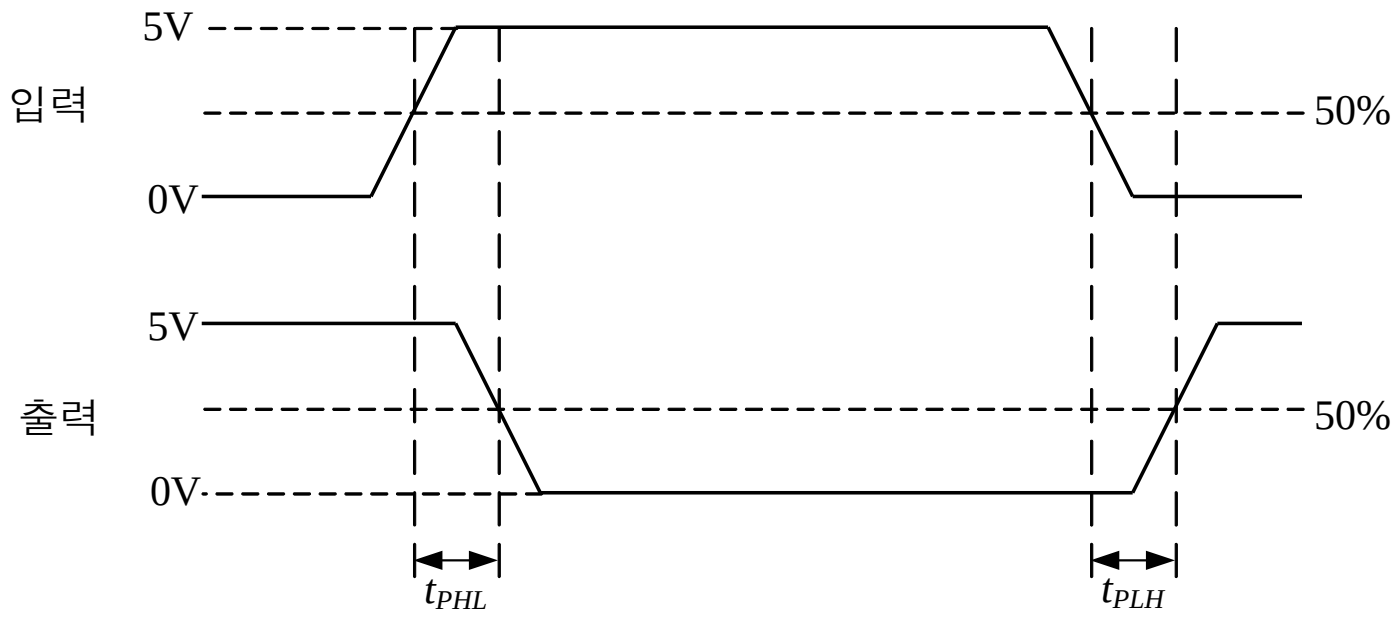
- if the magic smoke is let out, the chip stops working

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

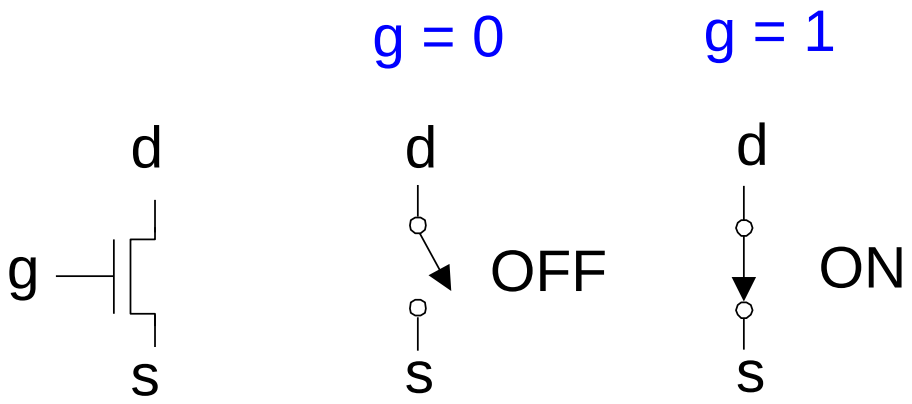
Gate Propagation Delay (전파지연시간)

- 신호가 입력되어서 출력될 때까지의 시간을 말하며, 게이트의 동작 속도를 나타낸다.



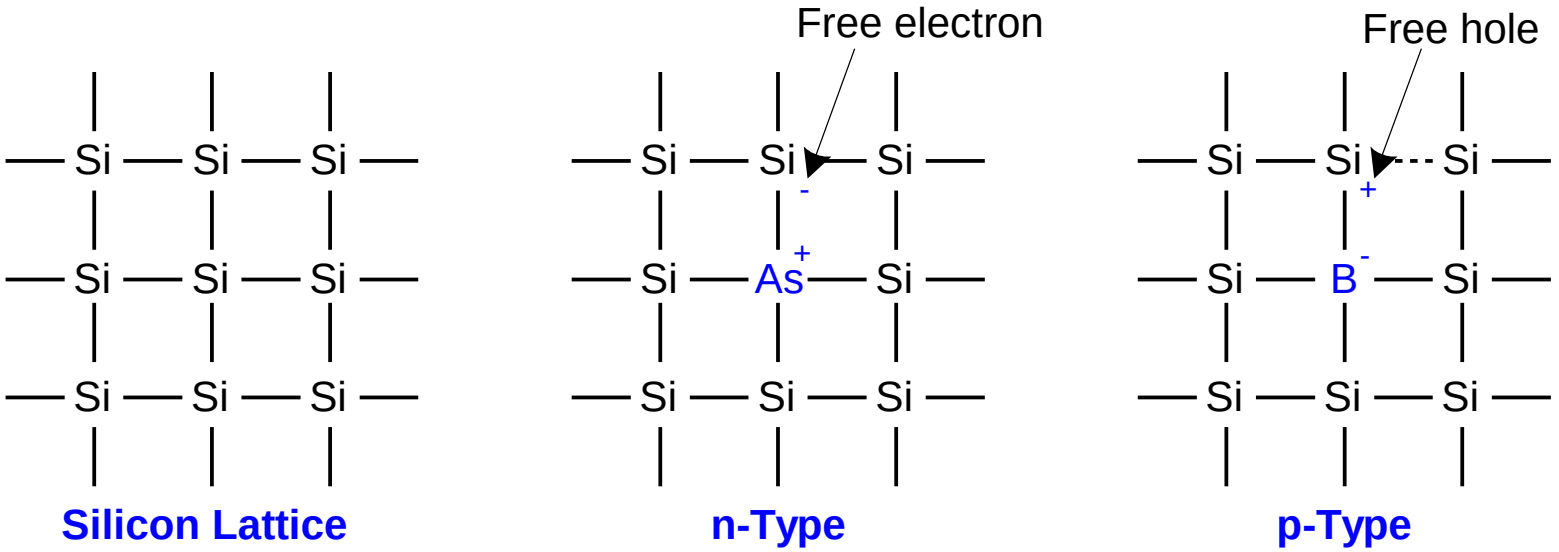
Transistor Basics

- Transistor is a three-ported voltage-controlled switch
  - Two of the ports are connected depending on the voltage on the third port
  - For example, in the switch below the two terminals (d and s) are connected (ON) only when the third terminal (g) is 1



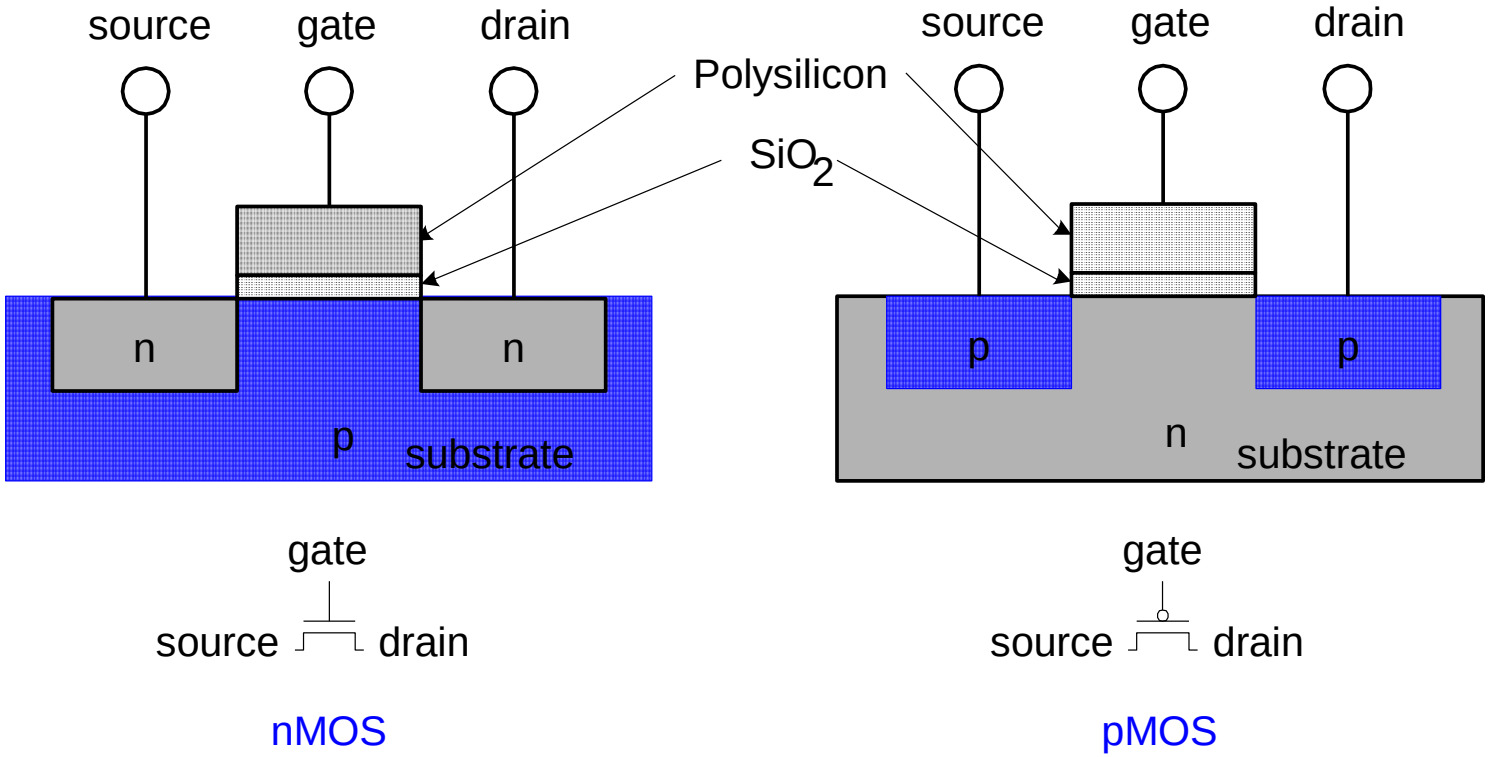
Silicon

- Transistors are built out of silicon, a semiconductor
- Silicon is not a conductor (no free charges)
- Doped silicon is a conductor (free charges)
  - n-type (free *n*egative charges, electrons)
  - p-type (free *p*ositive charges, holes)

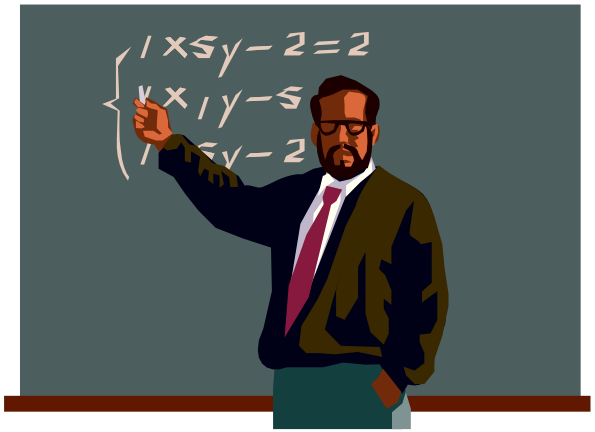


MOS Transistors

- Metal oxide silicon (MOS) transistors:
  - Polysilicon (used to be **metal**) gate
  - **Oxide** (silicon dioxide) insulator
  - Doped **silicon** substrate and wells

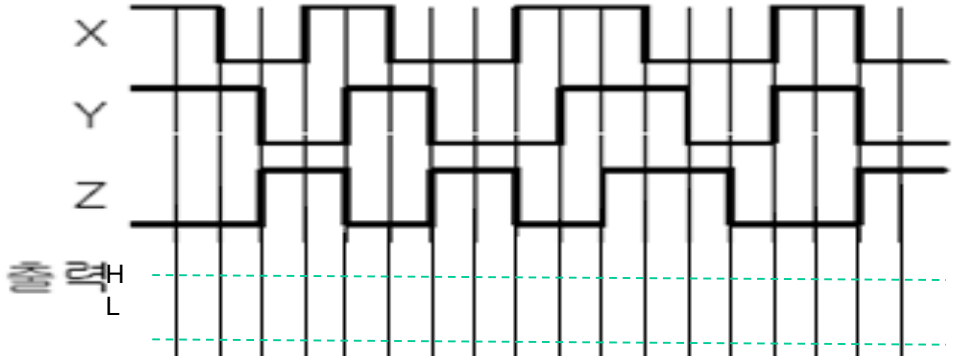


Q & A

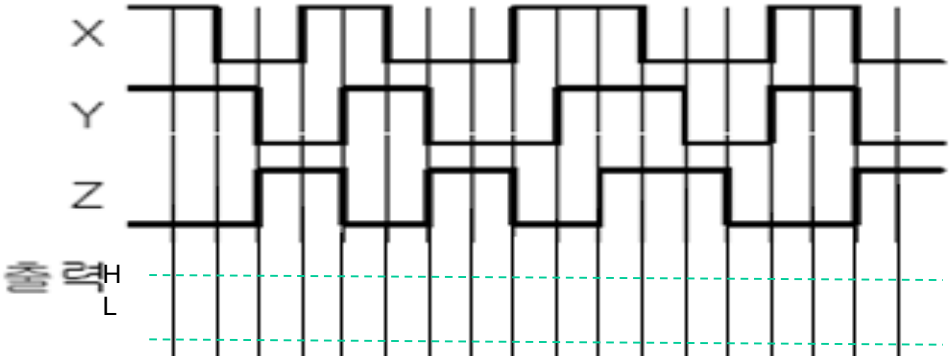


[실습] Chapter 1.5~1.6 :: 연습문제

1. [기본게이트] 3입력 NAND 게이트의 심벌, 부울식, 진리표, 출력파형을 작성하시오. (관련연습문제 p49 – 71, 72)



2. [기본게이트] 3입력 XOR 게이트의 심벌, 부울식, 진리표, 출력파형을 작성하시오. (관련연습문제 p49 – 71, 72)



[실습] Chapter 1.5~1.6 :: 연습문제

3. 여러 개의 스위치를 가진 회로에서 어느 한 스위치만 닫았더니, 램프가 켜졌다. 이것은 어떤 유형의 스위치 회로인가**?**

4. 여러 개의 스위치를 가진 회로에서 한 개 이외의 모든 스위치들을 닫았으나 램프는 켜지지 않았다**.** 이것은 어떤 유형의 스위치 회로인가**?**

5. p54 인터뷰질문 **1.3**

6. p49 연습문제 **73** 진리표 작성**:**

7. p49 연습문제 **75** 진리표 와 논리식 작성**:**

[실습] 응용문제

- [응용문제] 어떤 의사결정 회의에서 각 위원들은 안건에 대한 의견을 표명하기 위한 스위치를 한 개씩 가지고 있다. 각 위원은 안건에 찬성하면 스위치를 누르고, 반대한다면 스위치를 누르지 않는다. 두 위원(A,B)의 의견이 같은 경우에 한해 C 위원의 찬성이 있을 경우 출력(F)을 '1 ' 로 발생하는 회로를 구성하시오. 스위치는 누르면 '1'을 발생하고, 누르지 않으면 '0'을 발생한다.

- 위 [응용문제]를 다음페이지에서 설명하는 쿼터스를 이용하여 구현하고 시뮬레이션 결과가 예상한 진리표와 일치하는지 확인하시오.

[실습] Altera Quartus 사용법1

0. Altera Quartus, ModelSim 다운로드 설치
   - http://www.altera.com
   - support - download - Quartus II Web Edition

1. Quartus 소프트웨어 수행

2. Project 만들기
   - File -> New Project Wizard -> next -> 프로젝트 폴더이름
   - 프로젝트 이름을 줄 때는 모듈 명과 같게 해야 한다.
   - Family Device 셋팅 : Family를 사이클론 2 로 설정
   - available : ep2c35f672C8 -> next -> next -> finish

[실습] Altera Quartus 사용법2

3. 블럭 다이어그램 설계

- File 메뉴 -> new -> Block Diagram/Schematic File -> 설계에 들어간다

4. 컴파일

- Processing 메뉴 -> Start compilation

5. Vector Waveform

- File 메뉴 -> new -> Vector Waveform
- vwf 파일이름을 프로젝트 이름과 동일하게 변경하여 저장

-> Name/Value 부분 더블클릭 -> Node Finder -> Filter(Pins all) -> List -> Input과 Output 핀 선택 -> OK

-> 컨트롤 하려는 핀을 선택 -> Overwrite Clock을 선택하고 input 값들의 Period을 조정

[실습] Altera Quartus 사용법3

6. simulation

- processing 메뉴 -> start simulation

* Simulation이 성공적으로 실행되면 Output핀의 Value가 기록된다. 유의할 점은 Gate 마다 딜레이가 있기 때문에 결과값이 밀려 보이는 것이 이상한 것이 아니다.

7. Delay 조정

- delay를 없애기 위해서 Simulator tool에서 Simulation mode를 timing에서 functional로 바꾸어, delay를 나타내지 않을 수도 있다.