# Basis and Theory for Digital Circuits

↳ 디지털 회로를 구성하는 기본이론

Before learning to design digital circuits, two questions…

- Q1: What's '*fundamental basis*' of digital circuit? → 디지털 회로의 기반 이론
  - CMOS (Complementary Metal Oxide Semiconductor)

    ↳ MOS를 어떻게 구성하는가

- Q2: What's '*base design theory*' for implementing digital circuit ? → 디지털 회로의 디자인 이론
  - Boolean Algebra ⇒ "불 대수"

더 이전에는 "전기 스위치"(electric switches) 였음.

binary digital ⇒ 0, 1

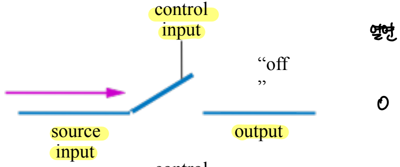
# Fundamental Basis

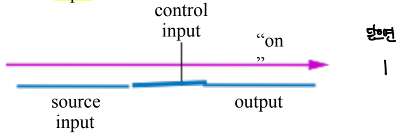
↳ 전자 switch

Electronic switches are the basis of binary digital circuits

A switch has three parts

- Source input, and output
  - Current wants to flow from source input to output.
- Control input
  - Controls whether that current can flow.




(b)

The amazing shrinking switch $\boxed{\text{제어장치소자}}$

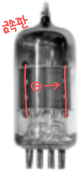
- 전기 스위치 – 1930s: Relays (릴레이) – 느림, 크기가 큼, 전력 큼
- 전자 스위치 – 1940s: Vacuum tubes (진공관) - 빨라짐, 크기가 큼
- 〃 – 1950s: Discrete transistor (반도체 - 트랜지스터)
- 〃 – 1960s: Integrated circuits (ICs)
  - Initially just a few transistors on IC
  - Then tens, hundreds, thousands of transistors

코일 – 자기장

relay

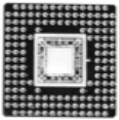
십 ~ 수백 voltage

금속판



vacuum tube

discrete transistor



IC

quarter (to see the relative size)

전자 vs 전기
(약전) (강전)

"CMOS"로 인해 가능해짐.

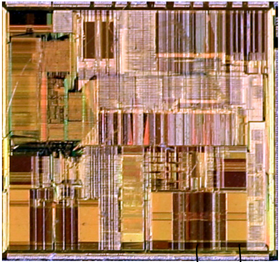
# Fundamental Basis

⇑ 집적도의 향상

→ 18개월마다 IC의 집적도가 2배씩 늘어날 것이다.

→ 실제로는 더 빠른속도로 발전함.

- IC (Integrated Circuit) capacity doubling about every 18 months for several decades (예언)
  - Known as "Moore's Law" after Gordon Moore, (co-founder of Intel 초기 창업자)
    - Predicted (in 1965) that components per IC would double roughly every year or so.
    - Enables incredibly powerful computation in incredibly tiny devices
  - Today's ICs hold *billions* of transistors
    - The first Pentium processor (early 1990s) needed only 3 million.



dye 포토 ⇐ 색으로 구분함. ⇐ ↳ 메모리 ↳ 논리게이트

*An Intel Pentium processor IC having millions of transistors*

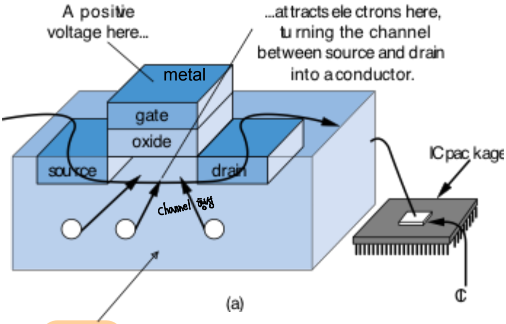
⇒ 3백만개 트랜지스터

# Fundamental Basis

집적도 향상 → 같은 면적에 더 많은 트랜지스터를 넣는 것.
→ S,D,G 의 size를 줄이는 방식 (MOS의 특징)

NMOS와 PMOS가 같이 쓰임. (in IC)

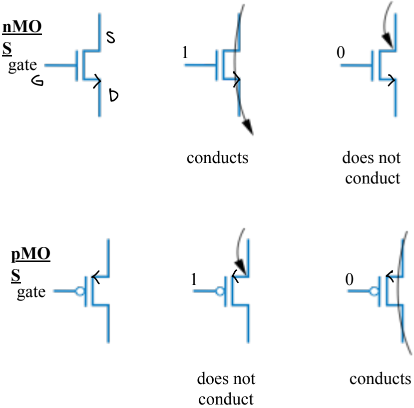
- CMOS (Complementary Metal Oxide Semiconductor) transistor
  - Basic switch in modern ICs




Silicon -- not quite a conductor or insulator:
도체 부도체

*Semiconductor*

power 공급 : 도체
미공급 : 부도체




$I_D = \frac{1}{2}\mu_n C_{ox}(V_{GS}-V_{th})^2$

# Fundamental Basis

ㄴ? 강대원 박사님.

Dr. Dae Won Kang (1931-1992)

- an Korean electronic engineer/ physicist
- B.S. from Seoul National Univ.
- Ph.D from Ohio state Univ.
- He developed the world's first MOS 1960.
- When, as a 29-year-old engineer at AT&T Bell Lab.
- He is also credited with inventing nonvolatile memory that is a basis of flash memory.
- He has been the most important one who is leading IT Revolution.

# Base Design Theory

정보이론 아버지

"불 대수"를 디지털 회로에 적용시킨사람.

## Claude Shannon (1916-2001)

- an American mathematician, electronic engineer, cryptographer
- Famous for having founded information theory : Known as “the father of information theory”
- He is also credited with founding both **digital computer** and **digital circuit** design theory in 1937.
- When, as a 21-year-old master’s student at MIT.
- He wrote a thesis about **how the Boolean algebra could be used to design digital circuits.**
  - The circuit in electronic devices have inputs, each of which is either 0 or 1 and produce outputs that are also 0 or 1.

→ 석사논문 중 최고의 논문임.

- **The most important master’s thesis of all time.**

"불 대수" 체계만 이용하면 0,1 로된 어떠한 input 이든 output으로 만들 수 있다.

# Base Design Theory

- Boolean Algebra (불 대수)
  - as developed in 1854 by George Boole in his book '*An Investigation of the Laws of Thought*'.
  - is a variant (변형) of ordinary elementary (일반 대수학의) algebra
    - Instead of the usual algebra of numbers, Boolean algebra is the algebra of truth values 0 and 1.
    - Variables represent 0 or 1 only.
    - Operators return 0 or 1 only.
  - Basic operators
    - AND ( $\cdot$ ) : $a \cdot b$ returns 1 only when both a=1 and b=1
    - OR (+) : $a + b$ returns 1 if either (or both) a=1 or b=1
    - NOT ( ' ) : $a'$ returns the opposite of a (1 if a=0, 0 if a=1)

↳ 기호 주의하기

# Base Design Theory

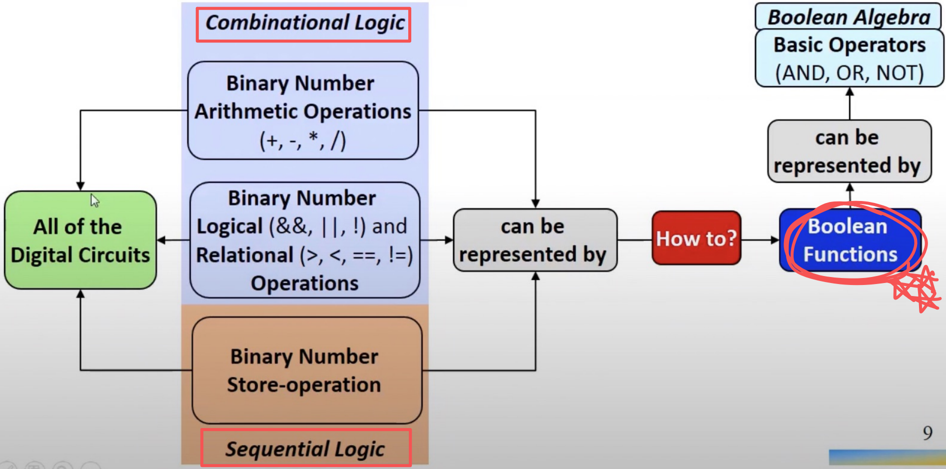
- Boolean Algebra ☆☆☆
  - Boolean Identities $\boxed{\rightarrow \text{10개의 법칙}}$ H.W/SW ⇐ 논리적 수식으로 구성됨.

| | | | |
|---|---|---|---|
| $x + 0 = x$<br>$x \cdot 1 = x$ | Identity Law | $x + (y + z) = (x + y) + z$<br>$x(yz) = (xy)z$ | Associative Law |
| $x + 1 = 1$<br>$x \cdot 0 = 0$ | Domination law | $x(y + z) = xy + xz$<br>$x + yz = (x + y)(x + z)$ | Distributive Law |
| $x + x = x$<br>$x \cdot x = x$ | Idempotent Law | $(x y)' = x' + y'$<br>$(x + y)' = x' y'$ | (드모르간의 법칙)<br>De Morgan's Law |
| $(x')' = x$ | Complementation Law | $x + xy = x$<br>$x(x + y) = x$ | Absorption Law |
| $x + y = y + x$<br>$xy = yx$ | Commutative Law | $x + x' = 1$<br>$x x' = 0$ | Complement Law |

# Base Design Theory

## Boolean Algebra and its Relation to Digital Circuits

- Claude Shannon's observation
  - all of the digital circuits can be implemented by only using 3 operations in boolean algebra no matter how the circuits are very complex.

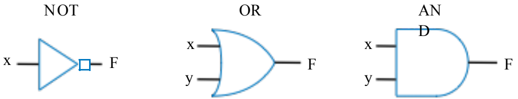


# Base Design Theory

## Boolean Algebra and its Relation to Digital Circuits

- Claude Shannon observed that all of the digital circuits can be implemented by only using 3 operations in boolean algebra no matter how the circuits are very complex. ↳ 불 대수의 3가지 operation으로 모든 디지털 회로 표현이 가능하다!
- In this reason, he introduced Boolean Logic Gates that are basic building blocks for digital circuits.
- Boolean logic gates corresponds to 3 basic operators in boolean algebra.
  - AND, OR, NOT
- Engineers can easily analyze and design digital circuits by using boolean logic gates.

**Symbol**




| a | NOT |
|---|---|
| 0 | 1 |
| 1 | 0 |

| a | b | OR |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

| a | b | AND |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

# Base Design Theory

## Boolean Logic Gates

– are eventually composed of CMOS transistor.

2개 4개 4개

NOT OR AND

**Symbol**


**Truth table**

NOT

| x | F |
|---|---|
| 0 | 1 |
| 1 | 0 |

OR

| x | y | F |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

AND

| x | y | F |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

**CMOS Transistor circuit**