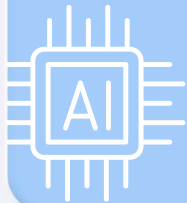




디지털논리회로 [Digital Logic Circuits]

1강.

# 디지털 논리회로와 데이터 표현



컴퓨터과학과 **강지훈** 교수

# 1강. 디지털논리회로와 데이터 표현

## ➡ 제1장.

# 컴퓨터와 디지털 논리회로



## 제1장 | 컴퓨터와 디지털 논리회로

contents

# 학습 목차

## ▶ 1 강

### 01 디지털 시스템

- 시스템의 정의
- 아날로그와 디지털
- 디지털 시스템의 설계 및 논리회로

### 02 컴퓨터 구성

- 컴퓨터 시스템
- 컴퓨터 하드웨어의 구성

### 03 집적회로

- 집적회로의 개요
- 집적도
- 디지털 논리게열
- 양논리와 음논리

# 1강. 디지털논리회로와 데이터 표현

## ➡ 제1장. 컴퓨터와 디지털논리회로

### 1.1 디지털 시스템



# 1.1.1 시스템



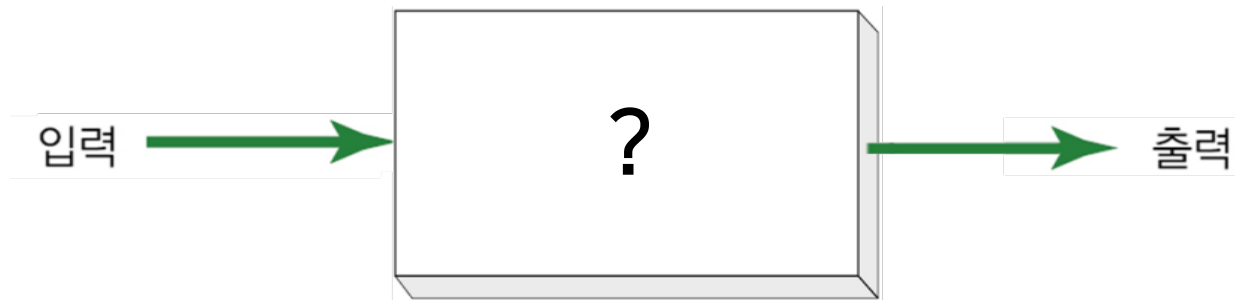
디지털논리로

> 01.



## • 시스템의 정의

블랙 박스형 시스템





# 1.1.1 시스템

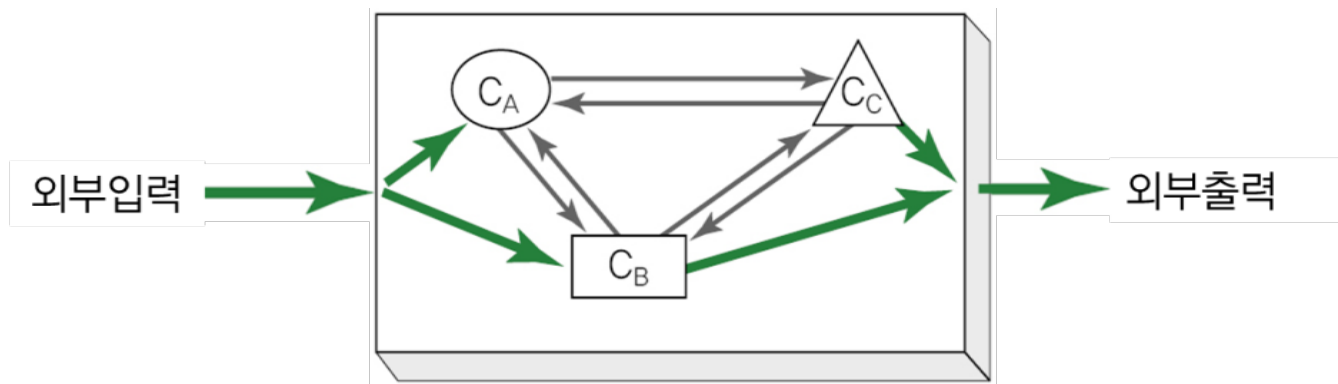


디지털논리회로

> 01.



구성요소의 집합으로서의 시스템



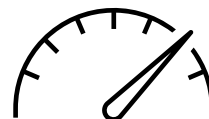
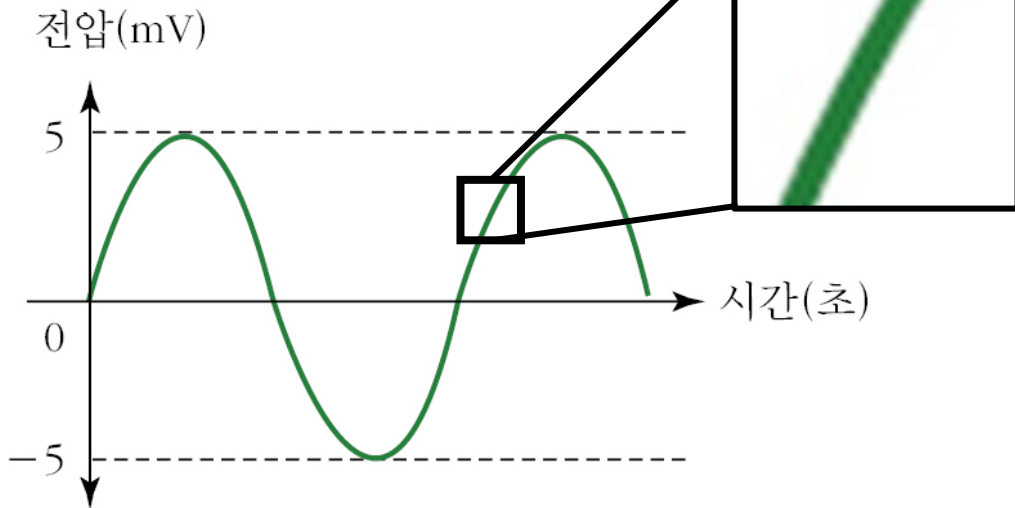


## 1.1.2 아날로그와 디지털

### • 데이터의 표현 방법(아날로그 방식)

- 연속적인 값으로 표현

아날로그 방식



아날로그 측정기



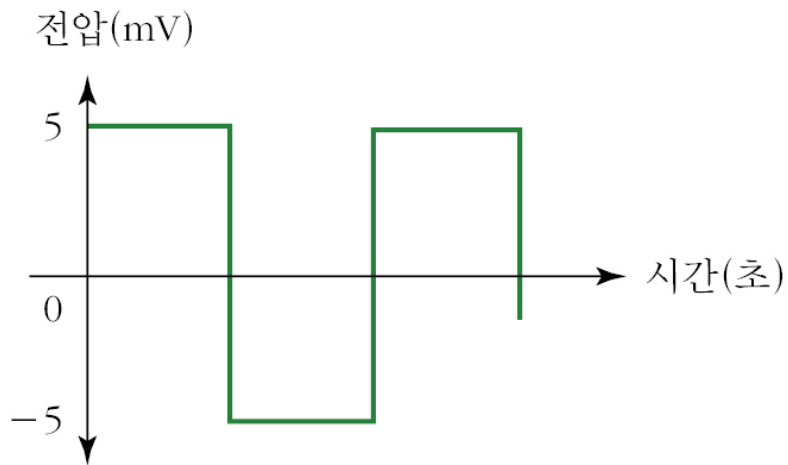
## 1.1.2 아날로그와 디지털



### • 데이터의 표현 방법(디지털 방식)

- 연속적이지 않고 구분된 상태로 명확히 나뉘는 값인 이산 값(Discrete Value)으로 표현

#### 디지털 방식



디지털 측정기





## 1.1.2 아날로그와 디지털



### • 아날로그 시스템과 디지털 시스템

#### 아날로그 시스템

- 입력과 출력이 아날로그 데이터인 시스템
- 예시: 눈금과 바늘로 값을 확인하는 계기 시스템 등

#### 디지털 시스템

- 입력과 출력이 디지털 데이터인 시스템
- 예시: 정확한 값을 확인할 수 있는 전자시계, 디지털 온도계 등



## 1.1.2 아날로그와 디지털

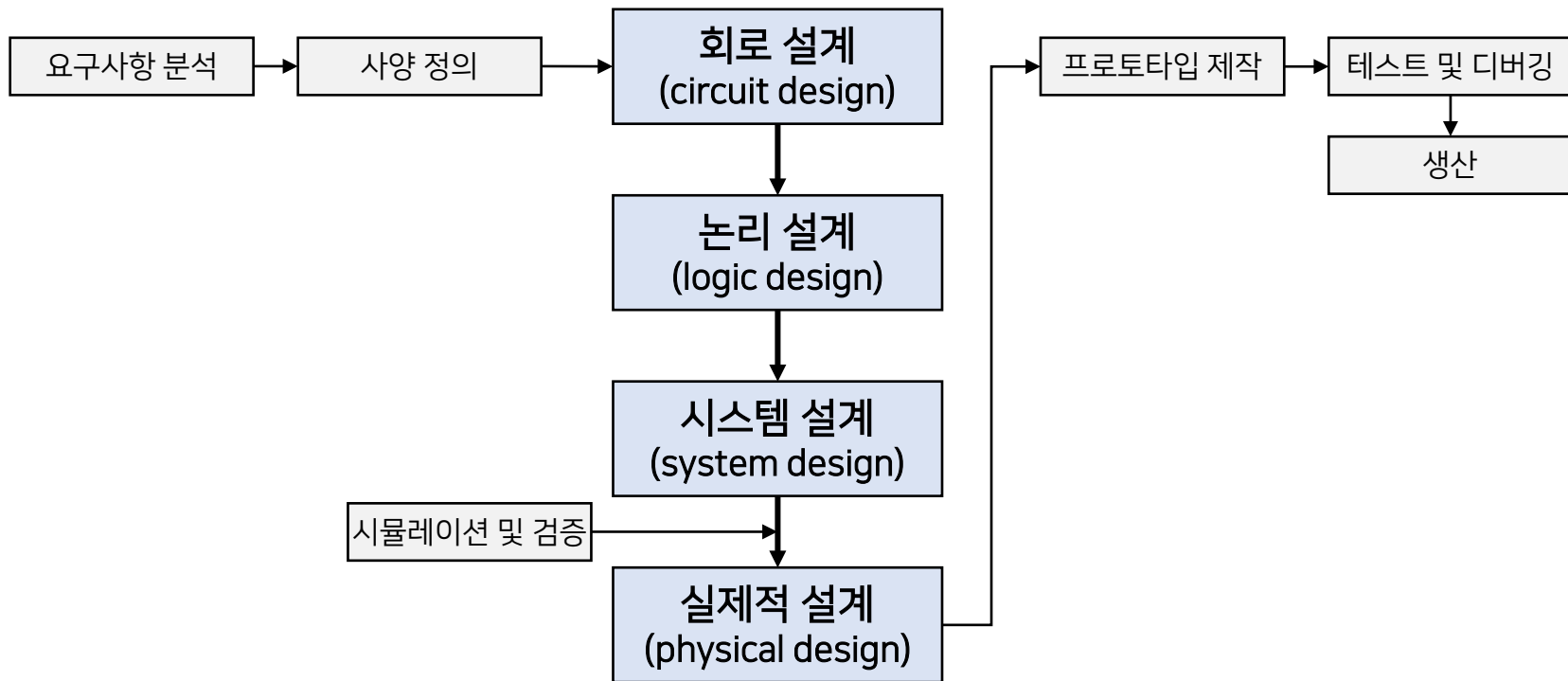


### • 디지털 시스템의 장점

- 편리성 ➡ 데이터가 숫자로 바로 입출력
- 융통성 ➡ 외부 조건의 변화에 따라 실행 순서 조절이 가능
- 단순성 ➡ 0과 1, 두 가지 상태의 신호만 취급
- 안정성 ➡ 0과 1, 2개의 상태 신호만 유지하면 됨
- 견고성 ➡ 0과 1의 상태로 전송되어 신호가 왜곡되어도 잡음에 강함
- 정확성 ➡ 신호를 이산신호로 변환하여 처리

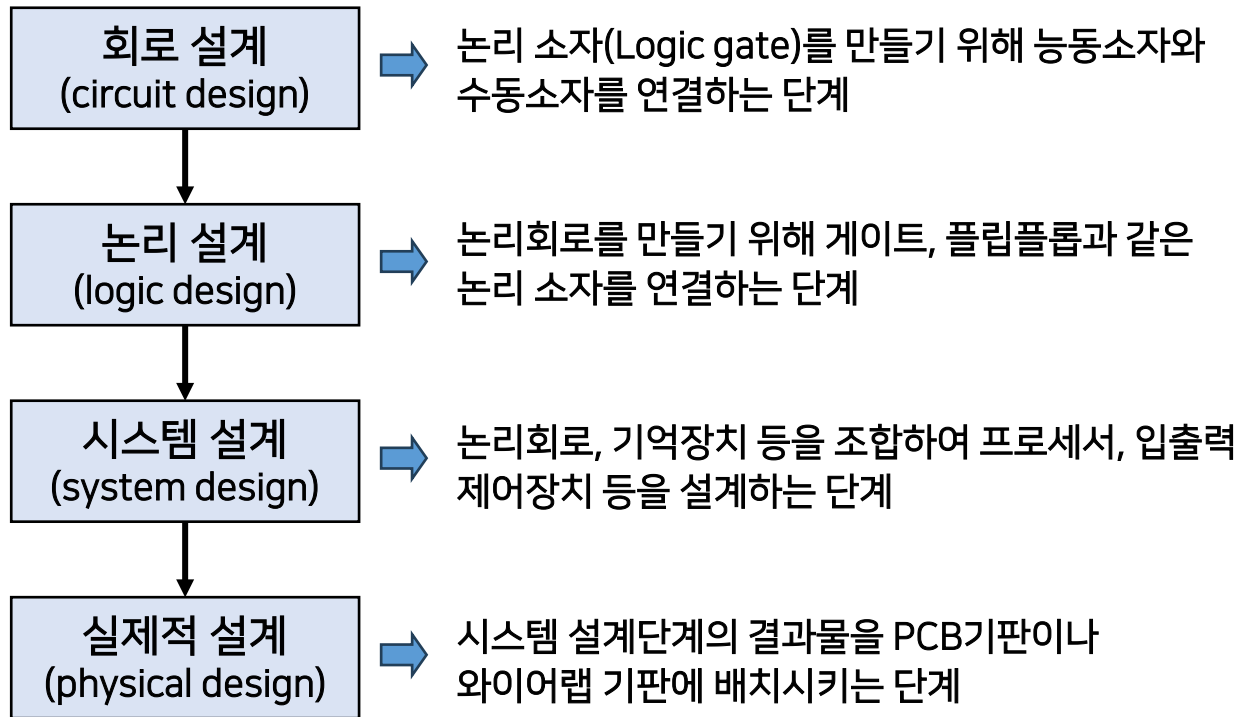


# 1.1.3 디지털 시스템의 설계 및 논리회로





## 1.1.3 디지털 시스템의 설계 및 논리회로





# 1.1.3 디지털 시스템의 설계 및 논리회로



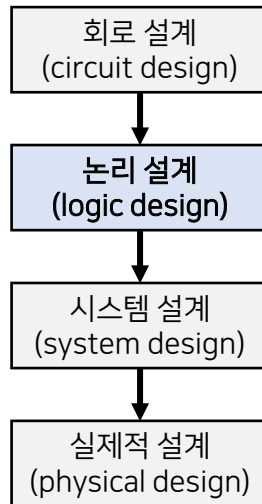
## • 디지털 논리회로의 개요

### • 디지털 논리 회로란?

- 2진 디지털 논리를 논리 게이트(Logic gate)로 구현한 것
  - 하나 이상의 논리 게이트가 결합된 것이 논리회로
- 디지털 시스템을 구현하는 기본 요소
- 디지털 시스템의 논리 설계 단계에서 설계됨

### • 저장요소의 유무에 따라 분류됨

- 조합논리회로: 가산기, 디코더 등
- 순서논리회로: 레지스터, 카운터 등



# 1강. 디지털논리회로와 데이터 표현

## ▶ 제1장. 컴퓨터와 디지털논리회로

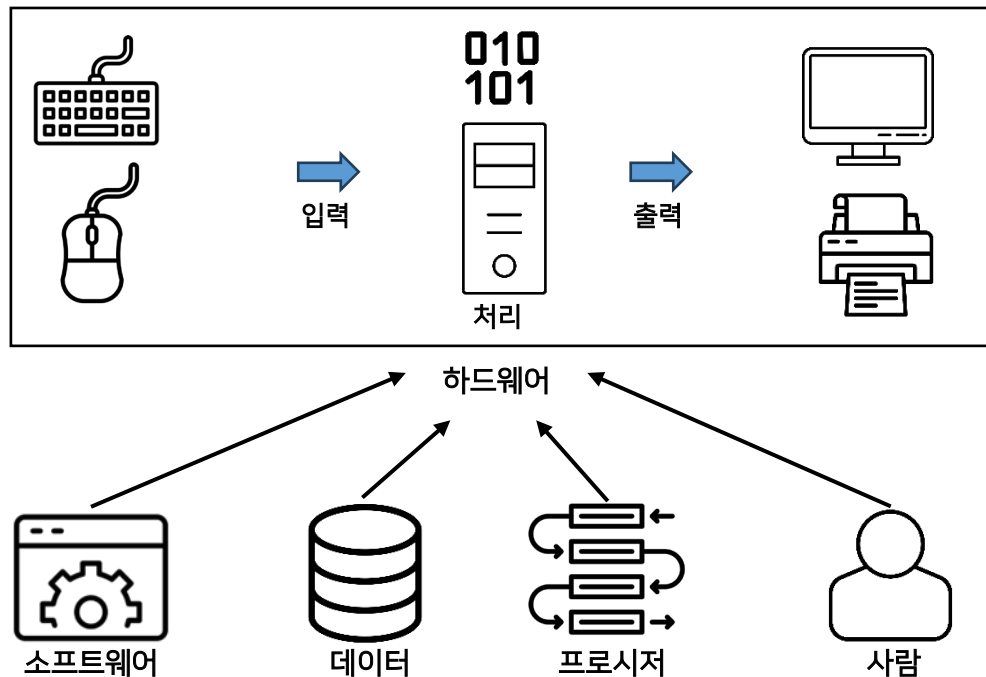
### 1.2 컴퓨터 구성



## 1.2.1 컴퓨터 시스템



### • 전자식 데이터 처리 시스템(Electronic Data Processing System)





## 1.2.1 컴퓨터 시스템



디지털논리회로

> 01.



- 컴퓨터를 시스템 측면에서 분석

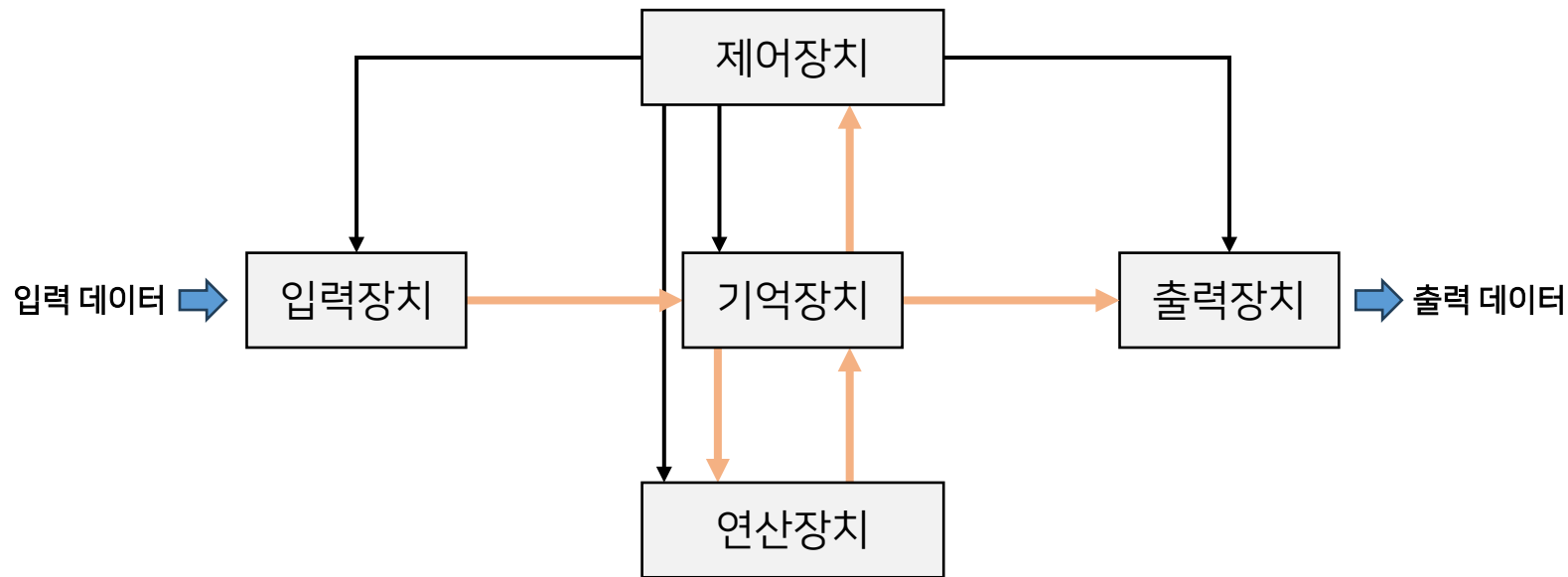
- 구성 요소는 무엇인가?
- 각 구성 요소는 어떤 기능을 갖는가?
- 입출력은 무엇인가?







## 1.2.2 컴퓨터 하드웨어 구성



# 1강. 디지털논리회로와 데이터 표현

## ▶ 제1장. 컴퓨터와 디지털논리회로

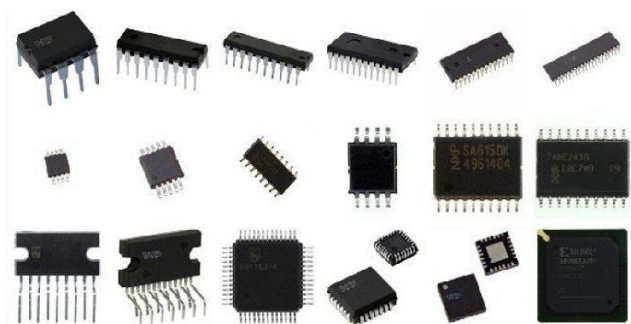
### 1.3 집적 회로



## 1.3.1 집적회로 개요

### • 집적회로(IC : Integrated Circuits)

- 디지털 회로의 구성 요소
- 디지털 게이트의 기능을 수행하는 전자 소자를 포함한 실리콘 반도체 크리스탈 (silicon semiconductor crystal), 칩(Chip)이라고 함
- 내부에 여러 게이트들이 목적에 부합되도록 상호 연결되어 있음





## 1.3.2 집적회로의 집적도



### • 집적도(level of integration)

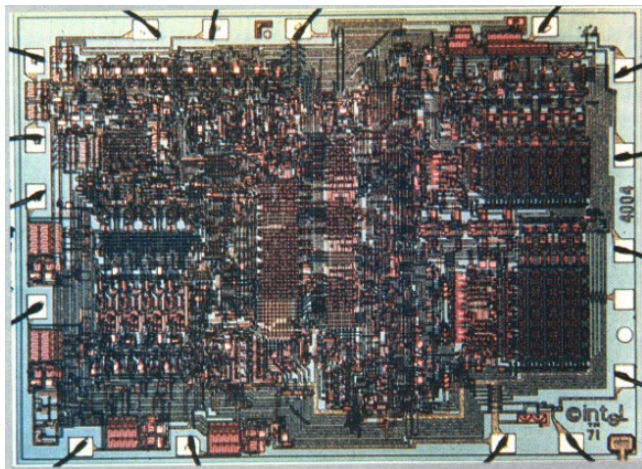
- 실리콘 칩의 단위 면적당 집적할 수 있는 전자 소자(gate)의 수

집적도	소자 수
소규모 집적 (SSI: Small Scale Integration)	몇 개 수준 (계산기, 전자시계)
중규모 집적 (MSI: Medium Scale Integration)	수십 개에서 수백 개 정도 (레지스터, ALU, 디코더, 멀티플렉스)
대규모 집적 (LSI: Large Scale Integration)	수천 개에서 수만 개 정도 (초기 마이크로프로세서, 메모리 칩)
초대규모 집적 (VLSI: Very Large-Scale Integration)	수백만 개에서 수억 개 이상 (현대 마이크로프로세서, GPU, SoC)



## 1.3.2 집적회로의 집적도

### • 집적회로





## 1.3.2 집적회로의 집적도



### • 디지털 논리계열(Digital Logic Families)

- 디지털 논리회로 설계에 사용되는 논리 소자의 집합
- 논리 게이트를 구현하기 위한 반도체 기술이나 전압, 전류 특성에 따라 분류됨
- 각 논리계열의 기본적인 회로는 NAND gate, NOR gate, 인버터

디지털 논리계열	응용 분야	대표 회로
TTL(Transistor-Transistor Logic)	초기 디지털 컴퓨터	NAND 게이트
ECL(Emitter-Coupled Logic)	고속 통신 시스템	NOR 게이트
NMOS (N-channel Metal-Oxide-Semiconductor)	초기 마이크로프로세서	NAND 게이트
CMOS(Complementary Metal-Oxide Semiconductor)	현대 마이크로프로세서	인버터





## 1.3.4 양논리와 음논리

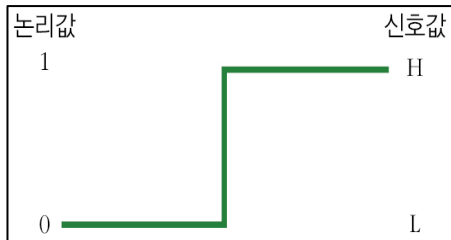


### • 양논리와 음논리

- 게이트의 입력과 출력에서 2진신호는 두 값 중에 한 값을 가짐
- 두 논리 값 0과 1에 대한 두 신호 값 H(High-높은 값), L(Low-낮은 값)을 할당하는 방법
  - 논리 1(참) 과 논리 0(거짓)을 어떻게 정의 하느냐에 따라 분류

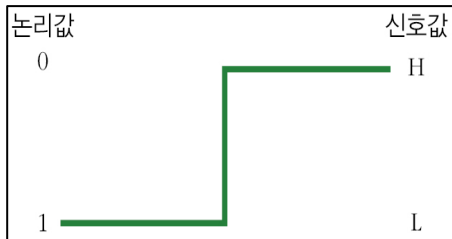
#### 양논리 시스템

- 높은 값을 논리 1
- 낮은 값을 논리 0



#### 음논리 시스템

- 높은 값을 논리 0
- 낮은 값을 논리 1



# 1강. 디지털논리회로와 데이터 표현

## ➡ 제2장.

# 데이터 표현





# 학습 목차

## ▶ 1강

### 01 수치 데이터

- 진법
- 보수

### 02 디지털 코드

- 10진 코드
- 영 숫자 코드

# 1강. 디지털논리회로와 데이터 표현

## ➡ 제2장. 데이터 표현

### 2.1 수치 데이터



## 2.1.1 진법

### • 수를 특정 기수(base)를 기반으로 숫자의 조합으로 표현하는 체계

- 기수(base)는 진법에서 자리 값의 기준이 되는 값, 한 자리에서 표현할 수 있는 숫자의 총 개수를 의미함(기수는 2 이상의 양의 정수)
- 숫자의 위치에 따라 각 숫자가 해당 자리의 값(가중치)을 결정함
- 가중치는 기수의 승수(거듭제곱)으로 계산됨

10진수 123 → 각 위치에 따라 100, 10, 1의 가중치를 가짐

    ↙    ↑    ↖  
백의 자리  십의 자리  일의 자리



## 2.1.1 진법

### • $r$ 진법

- 기수가  $r$  ( $r \geq 2$ ) 인 경우  $r$ 진법이라고 하며,  $r$ 진법으로 표현된 수를  $r$ 진수라고 함
- 각 자리는  $r$  개의 숫자( $0 \sim r - 1$ )로 표현
- $r$ 진수 오른쪽 아래에 기수  $r$ 을 표기함

10진수  $123_{10}$

2진수  $1001_2$



## 2.1.1 진법



### • $r$ 진법으로 표현된 $r$ 진수를 10진법으로 변환할 때

- 정수 부분이  $n$  자리, 소수점 이하 부분을  $m$  자리로 표시할 때

$$N = (a_{n-1}a_{n-2}\cdots a_0.a_{-1}a_{-2}\cdots a_{-m})_r$$

$$= a_{n-1}r^{n-1} + a_{n-2}r^{n-2} + \cdots + a_0r^0 + a_{-1}r^{-1} + a_{-2}r^{-2} +$$

$$\cdots + a_{-m}r^{-m}$$

$$= \sum_{k=-m}^{n-1} a_k r^k$$

10진수  $178.5_{10}$  는

$$178.5_{10} = 1 \times 10^2 + 7 \times 10^1 + 8 \times 10^0 + 5 \times 10^{-1}$$

2진수  $1010.01_2$  는

$$\begin{aligned} 1010.01_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} \\ &= 8 + 0 + 2 + 0 + 0 + 0.25 \\ &= 10.25_{10} \end{aligned}$$





## 2.1.1 진법

### • 여러가지 진법에 따른 수 표현

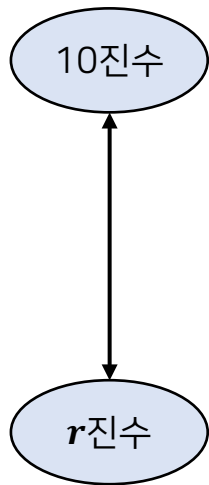
10진수	2진수	3진수	4진수	8진수	16진수
0	0	0	0	0	0
1	1	1	1	1	1
2	10	2	2	2	2
3	11	10	3	3	3
4	100	11	10	4	4
5	101	12	11	5	5
6	110	20	12	6	6
:	:	:	:	:	:
15	1111	120	33	17	F
16	10000	121	100	20	10



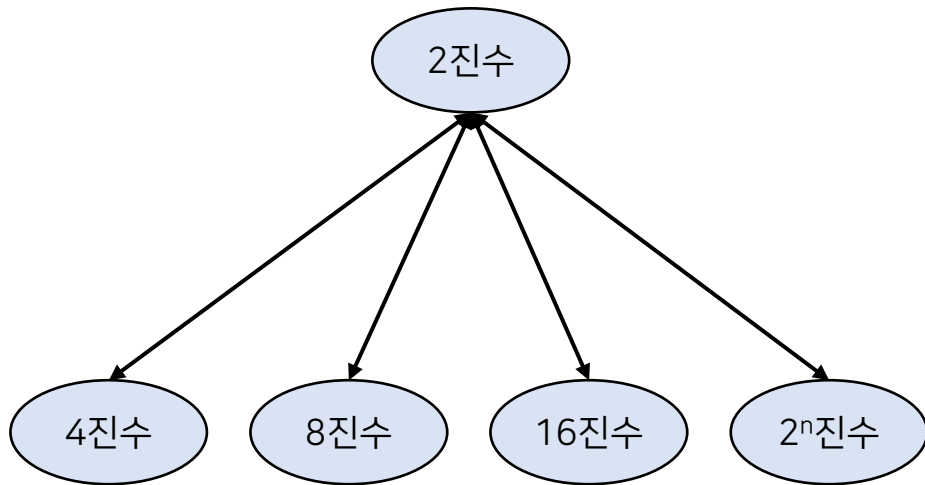
## 2.1.1 진법

### • 진수 변환

- 진법 체계의 공통 기반을 기준으로 변환이 가능함



10진수와  $r$ 진수의 변환



$2^n$  진수 상호간의 변환



## 2.1.1 진법

### • 컴퓨터에서의 대표적인 진법

- 2진수(Binary)
  - 전기 신호의 ON/OFF를 기반으로 표현되는 가장 기본적인 숫자 체계
- 8진수(Octal)
  - 2진수를 더 간결하게 표현하기 위해 사용됨, 현재는 사용 빈도 낮음
  - 리눅스 OS에서 파일 권한 정보를 3비트로 표현하며, 8진수로 요약됨
- 16진수(Hexadecimal)
  - 헥사코드(Hexa code)
  - 메모리 주소, 색상 코드 등 현대 컴퓨터에서 널리 사용됨





## 2.1.1 진법



### • 진수 변환

#### • $r$ 진수의 10진수 변환

2진수 1011.101을 10진수로 변환

$$\begin{aligned} 1011.101_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125 \\ &= 11.625 \end{aligned}$$

8진수 262.3을 10진수로 변환

$$\begin{aligned} 262.3_8 &= 2 \times 8^2 + 6 \times 8^1 + 2 \times 8^0 + 3 \times 8^{-1} \\ &= 128 + 48 + 2 + 0.375 \\ &= 178.375 \end{aligned}$$





## 2.1.1 진법

### • 진수 변환

#### • 10진수의 $r$ 진수 변환

10진수 28.125를 2진수로 변환

정수부분 변환

$$\begin{array}{r}
 2 \overline{) 28} \\
 2 \overline{) 14} \dots\dots 0 \\
 2 \overline{) 7} \dots\dots 0 \\
 2 \overline{) 3} \dots\dots 1 \\
 2 \overline{) 1} \dots\dots 1 \\
 0 \dots\dots 1
 \end{array}$$

$$(28)_{10} = (11100)_2$$

소수부분 변환

$$\begin{array}{r}
 0 \quad .125 \\
 \times 2 \\
 \hline
 0 \quad .250 \\
 \times 2 \\
 \hline
 0 \quad .500 \\
 \times 2 \\
 \hline
 1 \quad .000
 \end{array}$$

$$(0.125)_{10} = (0.001)_2$$

$$(28.125)_{10} = (11100.001)_2$$

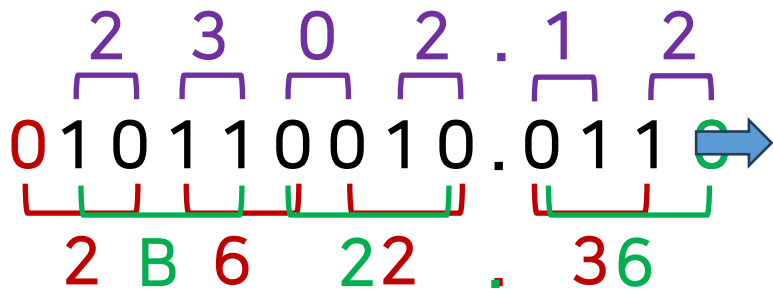


## 2.1.1 진법



### • 2진수를 $2^n$ 진수로 변환

2진수 10110010.011을 4진수, 8진수, 16진수로 변환



4진수	2302.12 <sub>4</sub>
8진수	262.3 <sub>8</sub>
16진수	B2.6 <sub>16</sub>



## 2.1.1 진법



### • $2^n$ 진수를 2진수로 변환

8진수 436.71을 2진수로 변환

$$\begin{array}{ccccccc} 4 & 3 & 6 & . & 7 & 1 & \\ \hline 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & . & 1 & 1 & 1 & 0 & 0 & 1 \end{array}$$
$$436.71_8 = 1001110.111001_2$$



### • 기타 변환

- 5진수를 8진수로 바꾸거나 2진수를 7진수로 바꾸려면 10진수로 변환한 후 해당 진수로 변환해야 함
- 몇몇 진수는 서로 직접적인 비트 그룹화 관계가 없음
  - 홀수(3, 5...) 진수에서 짝수(2, 4...) 진수로의 변환, 또는 그 반대
  - 6진수를 4진수로 변환하는 등의 공통 기반 진수의 거듭제곱과 관계가 없을 때
- 따라서, 모든 진법의 공통 기반인 10진수를 중간 변환으로 사용함



## 2.1.1 진법

### • 산술 연산

#### 가산

$$\begin{array}{r}
 \text{피가수} \rightarrow \quad 11001 \\
 \text{가 수} \rightarrow \quad + 11100 \\
 \hline
 \text{합} \rightarrow \quad 110101
 \end{array}$$

#### 감산

$$\begin{array}{r}
 \text{피감수} \rightarrow \quad 11001 \\
 \text{감 수} \rightarrow \quad - 01100 \\
 \hline
 \text{차} \rightarrow \quad 01101
 \end{array}$$

#### 승산

$$\begin{array}{r}
 \text{피승수} \rightarrow \quad 1101 \\
 \text{승 수} \rightarrow \quad \times \quad 11 \\
 \hline
 \quad 1101 \\
 1101 \\
 \hline
 \text{곱} \rightarrow \quad 100111
 \end{array}$$

#### 제산

$$\begin{array}{r}
 \quad \quad \quad 1000 \leftarrow \text{몫} \\
 \text{제 수} \rightarrow 111 \overline{) 111101} \leftarrow \text{피제수} \\
 \quad \underline{111} \\
 \quad \quad 101 \leftarrow \text{나머지}
 \end{array}$$



## 2.1.1 진법

### • 산술 연산

가산  $(27E)_{16} + (3F4)_{16}$ 의 계산  $\Rightarrow$  각 자리 수를 10진수로 변환 후 계산

2 7 E $\rightarrow$	$2_{10}$	$7_{10}$	$14_{10}$
+ 3 F 4 $\rightarrow$	$3_{10}$	$15_{10}$	$4_{10}$
6 7 2 <sub>16</sub>	$1_{10}$	$1_{10}$	$18_{10} = 16_{10} + 2_{10}$
	$6_{10}$	$23_{10} = 16_{10} + 7_{10}$	$= 12_{16}$
	$6_{16}$	$= 17_{16}$	

Diagram showing carry propagation: Green arrows indicate carries from the units place to the tens place, and from the tens place to the hundreds place. Red boxes highlight the final hexadecimal digits: 6, 7, and 2.



## 2.1.2 보수



- 보수(補數, complement – 보충을 해주는 수)

- 주어진 숫자가 특정 기준 값에서 얼마나 떨어져 있는지를 나타내는 값
  - 10진수 7에 대한 10의 보수는 3
- 기수가  $r$ 인 수 체계에서는 진보수인  $r$ 의 보수와  $(r - 1)$ 의 보수가 존재함
  - 10진수에서는 진 보수 10의 보수와  $(r - 1)$ 의 보수인 9의 보수가 있음
  - 2진수의 경우 진 보수인 2의 보수와  $(r - 1)$ 의 보수인 1의 보수가 있음





## 2.1.2 보수



### • $r$ 의 보수

- $N$ 에 대한  $r$ 의 보수(정수 부분이  $n$ 자리 일 때)

$$\text{공식} : r^n - N$$

10진수 35.34 의 10의 보수는

$$10^2 - 35.34 = 64.66$$

2진수 110111 의 2의 보수는

$$\begin{aligned} 2^6 - 110111_2 &= 64_{10} - 110111_2 \\ &= 1000000_2 - 110111_2 = 001001_2 \end{aligned}$$





### • $(r - 1)$ 의 보수

- $N$ 에 대한  $(r - 1)$ 의 보수(정수 부분이  $n$ 자리, 소수 부분이  $m$ 자리 일 때)

$$\text{공식} : r^n - r^{-m} - N$$

10진수 35.34 의 9의 보수는

$$10^2 - 10^{-2} - 35.34 = 99.99 - 35.34 = 64.65$$

2진수 1101.11 의 1의 보수는

$$\begin{aligned} 2^4 - 2^{-2} - 1101.11 &= 16_{10} - 0.25_{10} - 1101.11_2 \\ &= 10000_2 - 0.01_2 - 1101.11_2 \\ &= 1111.11_2 - 1101.11_2 \\ &= 0010.00_2 \end{aligned}$$





## 2.1.2 보수



### • $r$ 의 보수와 $(r - 1)$ 의 보수의 관계

	$r$ 의 보수	$(r - 1)$ 의 보수
10진수 678	322	321
2진수 1110	0010	0001

$r$ 의 보수 =  $(r - 1)$ 의 보수 값의 가장 낮은 자리에 +1



### • 2진수의 보수

- 보수는 주어진 숫자가 특정 기준 값에서 얼마나 떨어져 있는지를 나타내는 값
- 2진수에서 1의 보수는 주어진 값의 자리 수에서 표현할 수 있는 가장 큰 숫자를 기준 값으로 사용함
  - 2진수 10010에 대한 1의 보수에서 기준 값은 11111이 됨
  - 10010의 1의 보수는 기준 값 11111과 얼마나 떨어져 있는지를 나타내는 값임

$$11111 - 10010 = 01101$$



## 2.1.2 보수



+



+

디지털논리로

&gt; 01.



### • 보수를 이용한 감산

$$\pm A - (\pm B) = \pm A + (\mp B)$$

#### • 10진수 감산 $923 - 678 = 245$

- (-)는 보수를 이용해 가산할 수 있음
- 즉,  $923 + (678\text{의 } 10\text{의 보수 } 322) = 1245$ 가 되며, 올림 수 1을 제외하여 245가 됨

$$\begin{array}{r} 1011 \\ - 0101 \\ \hline 0110 \end{array}$$



$$\begin{array}{r} 1011 \\ + 1011 \\ \hline 10110 \end{array}$$

← 0101의 2의 보수





## 2.1.3 부호 있는 2진수

### • 정수의 표현 방법

10진수	부호 있는 절대치	부호 있는 1의 보수	부호 있는 2의 보수
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
...	...	...	...
+0	0000	0000	0000
-0	1000	1111	-
...	...	...	...
-6	1110	1001	1010
-7	1111	1000	1001
-8	-	-	1000

# 1강. 디지털논리회로와 데이터 표현

## ➡ 제2장. 데이터 표현

### 2.2 디지털 코드



## 2.2.1 10진 코드

### • 10진수를 나타내는 2진 코드

10진수	BCD8421	BCD2421	BCD84-2-1	Excess-3	Gray
0	0000	0000	0000	0011	0000
1	0001	0001	0111	0100	0001
2	0010	0011	0110	0101	0011
3	0011	0011	0101	0110	0010
4	0100	0100	0100	0111	0110
5	0101	1011	1011	1000	0111
6	0110	1100	1010	1001	0101
7	0111	1101	1001	1010	0100
8	1000	1110	1000	1011	1100
9	1001	1111	1111	1100	1101
사용되지 않는 비트조합	1010	0101	0001	0000	1111
	1011	0110	0010	0001	1110
	1100	0111	0011	0010	1010
	1101	1000	1100	1101	1011
	1110	1001	1101	1110	1001
	1111	1010	1110	1111	1000





## 2.2.2 영 숫자 코드

- **ASCII 코드**(American Standard Code for Information Interchange)
  - ASCII하나의 영 숫자 코드가 7비트로 구성, 전체 수는  $128(=2^7)$ 개 +1비트 패리티 비트
- **EBCDIC**(Extended Binary Coded Decimal Interchange Code)
  - 하나의 영 숫자 코드가 8비트로 구성
  - 4비트인 BCD 코드를 8비트로 확장하여 문자까지 포함
- **유니코드**
  - 아스키 코드는 알파벳을 제외한 다른 언어를 표현하는데 제한적임
  - 유니코드는 세계의 모든 문자를 컴퓨터에서 일관되게 표현하고 다룰 수 있도록 개발된 표준 방식임(UTF-8, UTF-16, UTF-32)



# 내용 정리

Summary

Contents



## 1강 | 디지털 논리회로와 데이터 표현



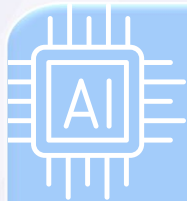
디지털 +  
논리회로



- 01 디지털 시스템 및 집적회로의 개념
- 02 수치 데이터 표현 방식
- 03 보수와 보수를 이용한 감산
- 04 디지털 코드

Digital Logic Circuits





다음시간에는

## 2 강. 논리게이트와 부울대수 (1)