

마이크로컨트롤러 기초(#514112)

#3. 컴퓨터 구조 기초

한림대학교
전자공학과 이선우

주요 학습 내용

▶ 컴퓨터 시스템 구조 기초

- ▶ 디지털 논리 회로 관련 용어, 장치 기초
- ▶ 컴퓨터 시스템 구조 관련 기초 내용

▶ 참고문헌

- ▶ “논리회로 이론,실습, 시뮬레이션”, 임석구, 홍경호, 한빛미디어, 2007
- ▶ “알기 쉽게 해설한 컴퓨터 구조 완성”, 조경산 저, 이한 출판사

Typical digital signal

- ▶ TTL (transistor-transistor logic) logic-level range
 - ▶ Logic “1”, “High”: voltage +2 ~ +5V
 - ▶ Logic “0”, “Low”: voltage 0 ~ 0.8V

- ▶ Positive logic(정 논리)
 - ▶ HIGH=1 (+5V)
 - ▶ LOW=0 (0V)

- ▶ Negative logic(부 논리)
 - ▶ HIGH=0 (+5V)
 - ▶ LOW=1 (0V)

Logic Gates & Bitwise operator in C language

- ▶ **AND: $x=AB$ or $A \bullet B$ (&)**
 - ▶ & operator in C language.
 - ▶ Ex. `a = a & 0x0f` (before `a=0x35`, after `a=0x05`)
- ▶ **OR: $x=A+B$ (|)**
 - ▶ Ex. `a = a | 0xf0` (before `a=0x05`, after `a=0xf5`)
- ▶ **XOR: $x=A \oplus B$ (^)**
 - ▶ Ex. `a = a ^ 0xff` (before `a=0xa5`, after `a=0x5a`)
- ▶ **NOT: $x=A'$ (~)**
 - ▶ Ex. `a = ~a` (before `a=0x0f`, after `a=0xf0`)

조합 논리 (Combinatorial logic)

▶ Combinatorial logic이란?

- ▶ 조합논리(combinational logic): 게이트의 조합으로 이루어진 회로로 출력이 입력으로 귀환되지 않는 회로
- ▶ 인/디코더, 가/감산기, 승산기
- ▶ $Y = F(X)$, where X/Y =입/출력변수의 집합
- ▶ **순서논리(sequential logic):** 메모리를 포함

▶ 조합논리 설계 단계

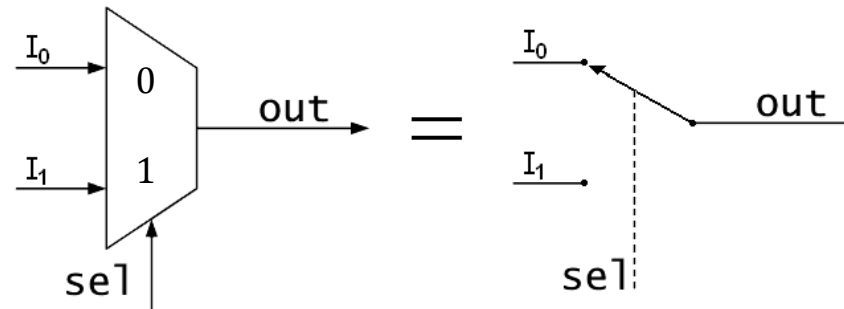
1. 문제의 기술
2. 진리표 작성(문제의 기술에 근거)
3. 출력식의 간략화(카노맵, McClusky)
4. 논리식의 재전개 - bubble, NAND, NOR
5. 최종 논리도 작성
6. 설계의 문서화 - 변수와 작동레벨 표시

Digital Multiplex (Mux)

- ▶ 여러 개의 아날로그 혹은 디지털 입력 신호들 중 하나를 선택하는 기능을 하는 장치

- ▶ 2-to-1 Mux

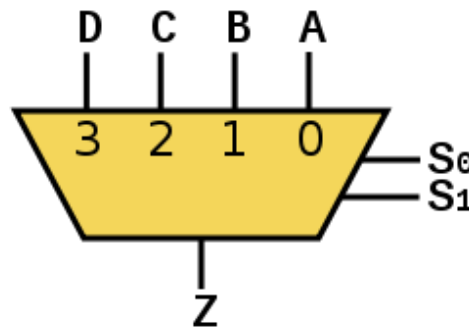
- ▶ If $sel=0$, then $out=I_0$
- ▶ If $sel=1$, then $out=I_1$



<http://en.wikipedia.org/wiki/File:Multiplexer2.png>

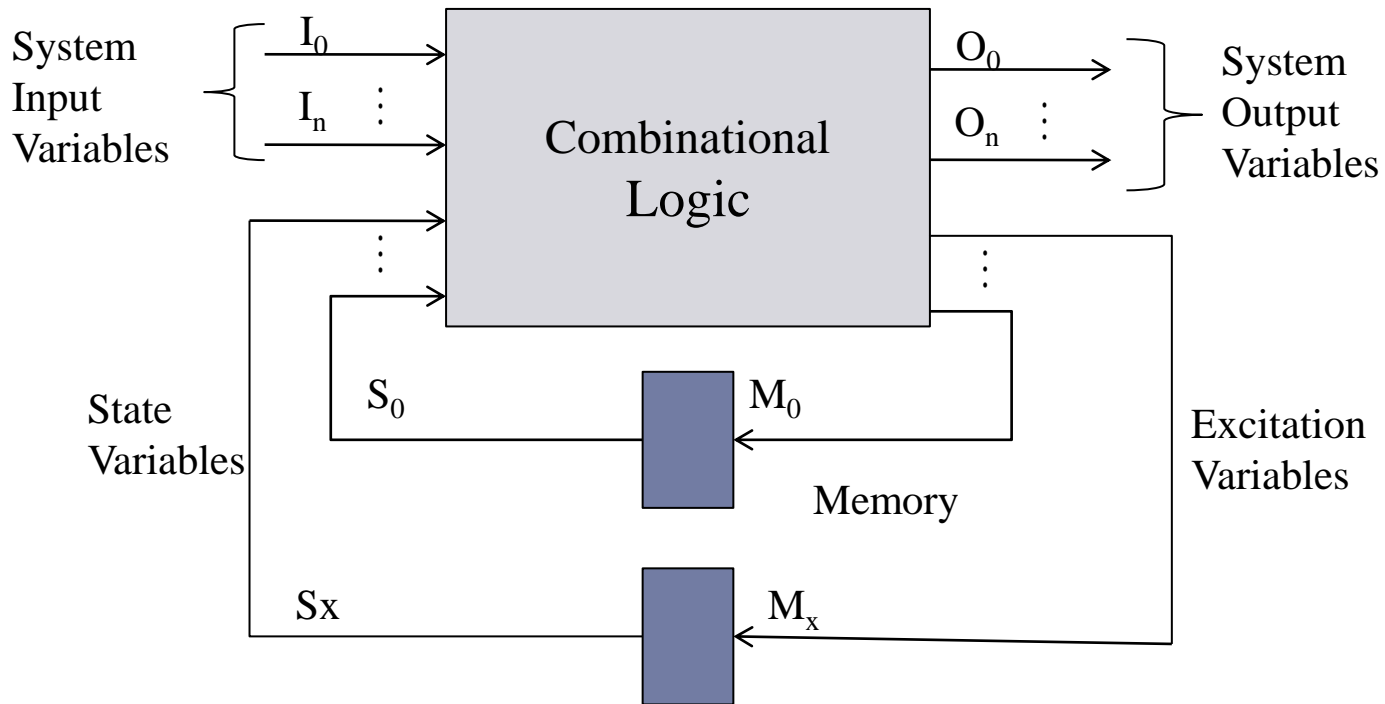
- ▶ 4-to-1 Mux

- ▶ $S_1:S_0=00 \rightarrow A$
- ▶ $S_1:S_0=01 \rightarrow B$



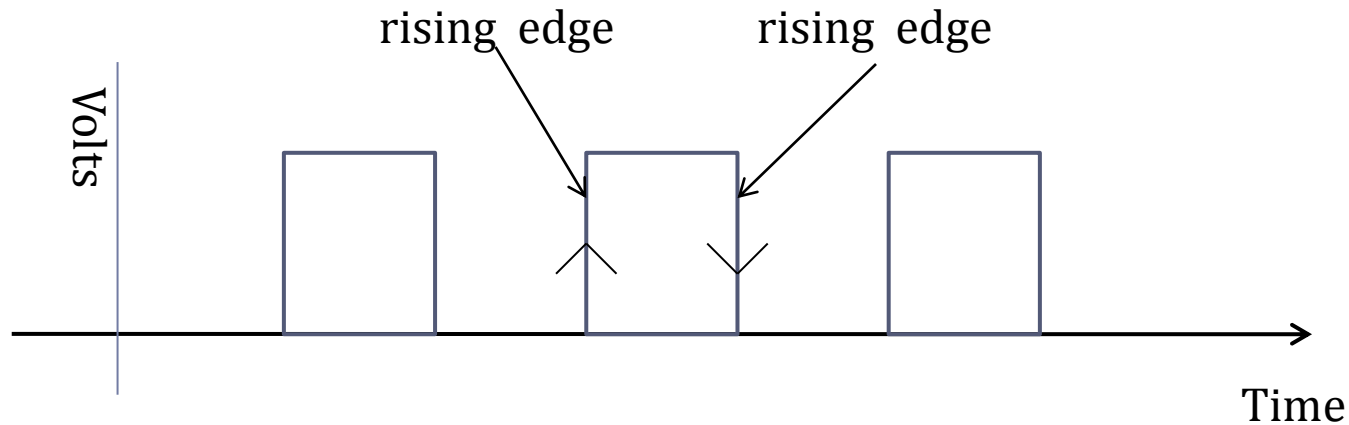
순서회로 (Sequential Logic)

- ▶ 조합논리 + 저장장치(메모리)
- ▶ 입력변수(I) -> 출력변수(O)
- ▶ 내부변수 : 상태변수(S) + 여기변수(E)



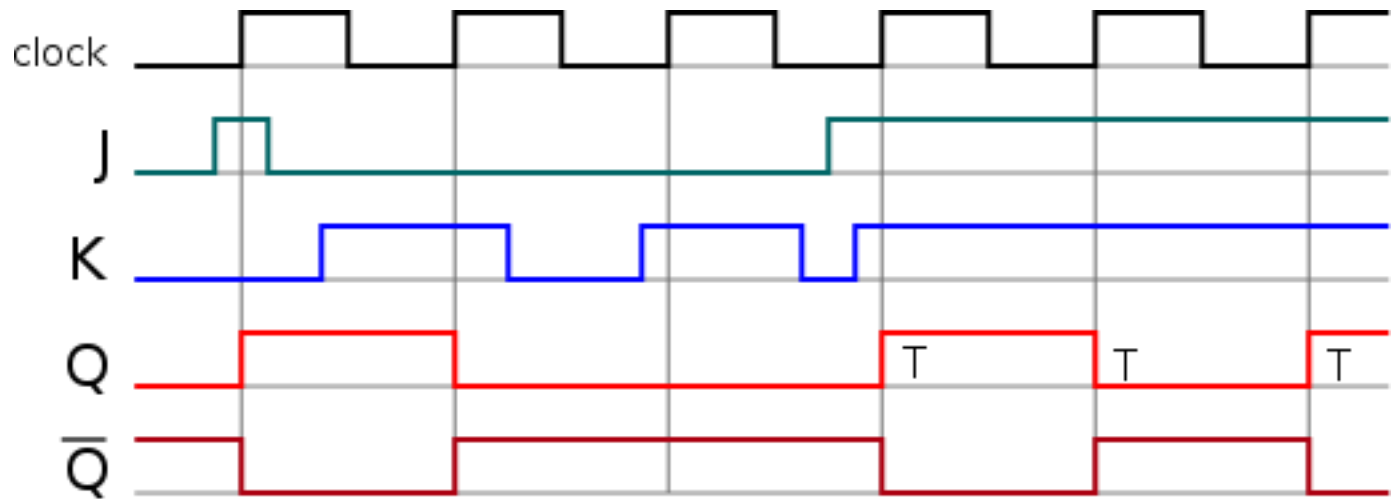
Clock signals

- ▶ 대부분의 디지털 시스템은 “동기 순차 시스템 (synchronous sequential systems)”임
⇒ 시스템이 주 클럭 신호(master clock signal)에 따라 동작함을 의미.
- ▶ 대개 다음과 같이 구형파(square wave; 50% duty cycle)



타이밍 다이어그램 (Timing diagram)

- ▶ 타이밍 다이어그램: 입력과 출력의 변화와 서로 간의 관계를 시간의 흐름 순으로 가로로 배치해서 나타낸 그래프.
- ▶ Ex. JK flip-flop
 - ▶ $J=1, K=0 \rightarrow \text{Set } Q=1, /Q=0$
 - ▶ $J=0, K=1 \rightarrow \text{Reset } Q=0$
 - ▶ $J=1, K=1 \rightarrow \text{Toggle}$



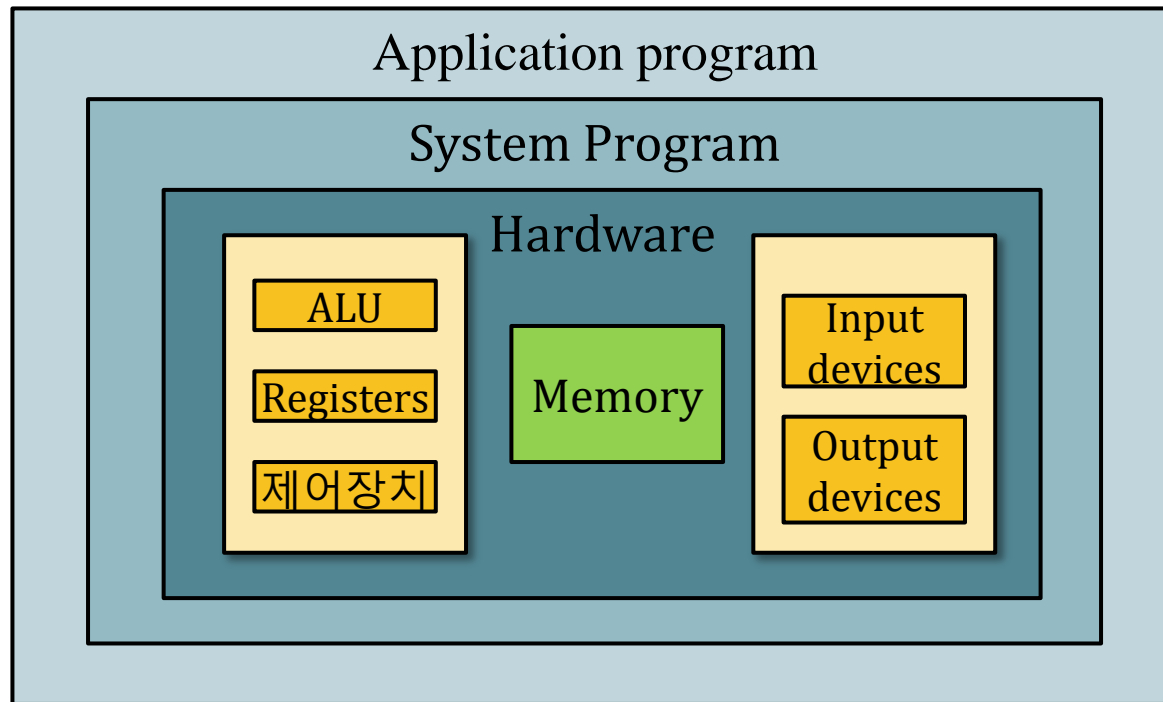
http://en.wikipedia.org/wiki/File:JK_timing_diagram.svg

T = toggle

컴퓨터 시스템 구성

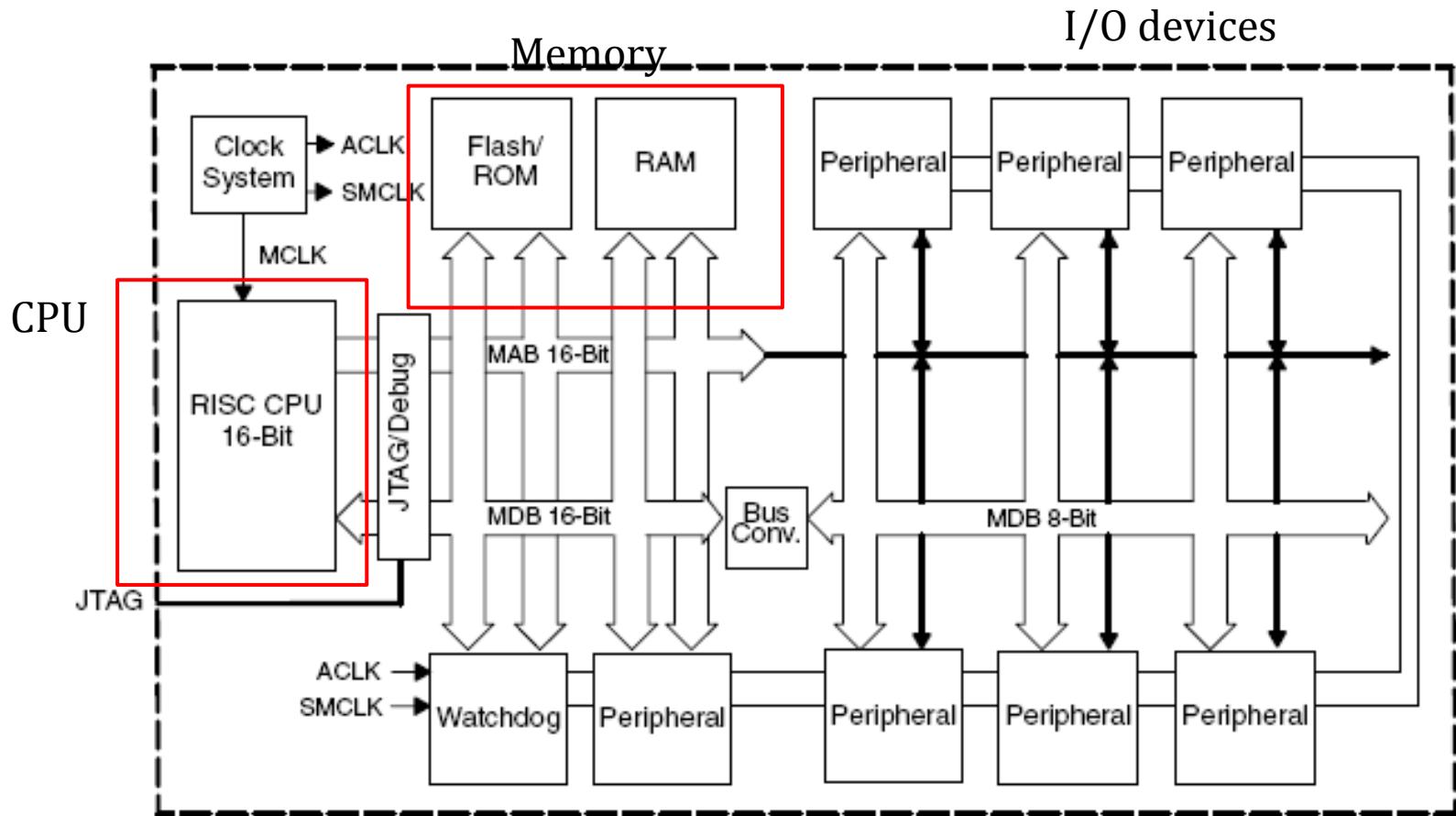
- ▶ 컴퓨터 시스템 : 컴퓨터의 하드웨어 및 관련 소프트웨어 까지 포함한 용어

Computer System



MSP430 Architecture

Figure 1-1. MSP430 Architecture



[1] 1-3 page

Central Processing Unit (CPU)

▶ CPU

- ▶ Computer의 기능을 수행하는 핵심 요소로 프로그램의 명령어를 수행하는 장치
- ▶ CPU는 컴퓨터의 여러 기능을 수행하기 위해 일정한 순서로 짜여진 명령어 집합(이를 컴퓨터 프로그램이라 함)을 수행함.
- ▶ CPU의 기본적인 동작은 물리적 메모리(program memory)에 기록되어 있는 machine code(명령어; Instruction)를 읽어와서(fetch) 해석하고(decode) 실행(execute)하는 동작을 반복한다. (오직 이것만 한다!!)
- ▶ CPU내부 주요 구성
 - ▶ ALU, 레지스터, 제어회로, 내부 연결 장치(버스)

Registers

- ▶ 프로세서 레지스터(범용 레지스터)는 작은 양의 정보를 저장하는 장치 (8bit, 16bit, 32bit-register로 표현하며 이 값은 저장하는 데이터의 크기를 나타냄)
- ▶ CPU 내부에 있고 CPU 동작을 제어하고 작동된 결과 등도 저장된다.
- ▶ MSP430 CPU
 - ▶ 16개의 범용 레지스터 가짐: R0~R15
 - ▶ Size: 16bit

Program Counter	PC/R0
Stack Pointer	SP/R1
Status Register	SR/CG1/R2
Constant Generator	CG2/R3
General-Purpose Register	R4
General-Purpose Register	R5
General-Purpose Register	R6
General-Purpose Register	R7
General-Purpose Register	R8
General-Purpose Register	R9
General-Purpose Register	R10
General-Purpose Register	R11
General-Purpose Register	R12
General-Purpose Register	R13
General-Purpose Register	R14
General-Purpose Register	R15

[2] 10 page

Memory

- ▶ CPU가 동작하기 위해선 반드시 메모리가 필요함. Why?
- ▶ Program memory
 - ▶ Program이란?
 - ▶ Definition: A set of instructions, 즉 명령어들이 일정 순서로 잘 작성된 것이 프로그램
 - ▶ 명령어(Instruction)이란? CPU가 할 수 있는 단위 기능들..
 - ▶ 메모리에 저장되어 있는 프로그램은 순서대로 CPU가 읽어 실행함.
 - ▶ ROM 사용
- ▶ Data memory
 - ▶ 프로그램이 실행하면서 사용하는 메모리로 명령어 수행 결과 등을 저장하는데 사용
 - ▶ RAM 사용

컴퓨터 메모리 종류: RAM

- ▶ Volatile(휘발성) memory
 - ▶ 저장된 정보를 유지하기 위해 전력(전원) 필요한 메모리
 - ▶ Static RAM
 - ▶ 1 bit 기억을 위해 여러 개의 반도체 소자가 필요 → 비쌈
 - ▶ Cache memory, MCU internal data memory etc.
 - ▶ Dynamic RAM
 - ▶ 기억 비트당 한 개 소자를 사용하므로 대용량 저가 구현 가능
 - ▶ 전원이 있어도 일정 시간 내에 refresh를 해주어야 함.
 - ▶ 메모리 외부에 refresh를 위한 별도 회로 필요
 - ▶ PC의 주기억장치
- ▶ Upcoming
 - ▶ T-RAM (Tyristro RAM) : <http://en.wikipedia.org/wiki/T-RAM>
 - ▶ Z-RAM (Zero capacitor RAM): <http://en.wikipedia.org/wiki/Z-RAM>
 - ▶ TTRAM (Twin Transistor RAM):
http://en.wikipedia.org/wiki/Twin_Transistor_RAM

컴퓨터 메모리 종류: ROM

▶ Non-volatile (비휘발성) memory

▶ ROM

▶ Mask ROM

- ROM 제조시에 프로그램 코드를 이용하여 내부 회로 고정시켜 제조
- 지우기, 쓰기 불가

▶ PROM(Programmable ROM)

- OTP(One Time Programmable)

▶ EPROM (Erasable PROM)

- 전기적으로 쓰고 광학식(UV light 사용)으로 지움

▶ EEPROM(Electrically EPROM)

- 동작 전압보다 큰 전압 이용 지움.
- Byte 단위로 지움.
- Flash 보다 제조 단가 비쌈



컴퓨터 메모리 종류: ROM

- ▶ Non-volatile (비휘발성) memory
 - ▶ 초기 시대: Paper tape, punched cards
 - ▶ Storage devices
 - ▶ Magnetic devices: hard discs, floppy discs
 - ▶ Optical discs: CDRom, DVD, Blu-ray
 - ▶ Flash memory (http://en.wikipedia.org/wiki/Flash_memory)
 - ▶ EEPROM의 한 종류
 - ▶ 제조단가 저렴하여 대용량 가능
 - ▶ Block 단위로 지움.
 - ▶ Write endurance : erase/rewrite 할 수 있는 최대 횟수, 10K~1M
 - ▶ 구현 방식에 따라 NOR, NAND flash로 구별

메모리 장치 개념

▶ 메모리 장치

▶ 입력 1: Address

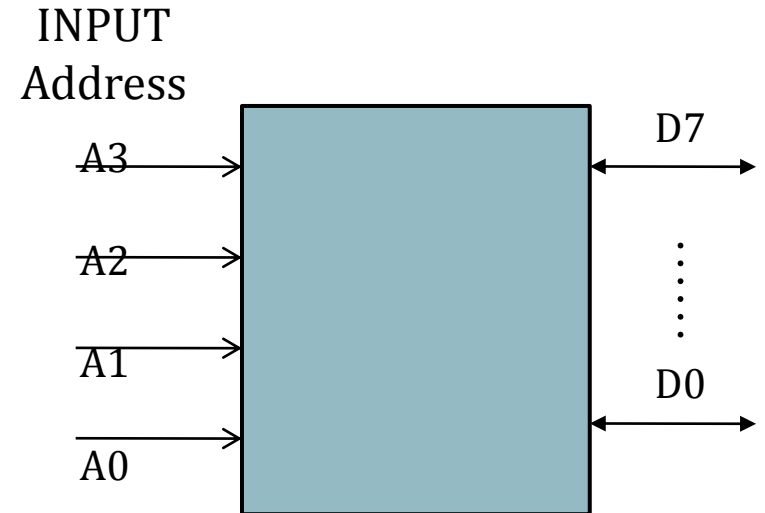
- ▶ 각 데이터를 저장하고 있는 박스의 번호
- ▶ 만약 address line이 모두 4bit이라면, 모두 16개의 박스 존재

▶ 입력 2: control line

- ▶ Chip enable, Output Enable, Write Enable(RAM의 경우) 등 동작 제어를 위한 입력

▶ 입/출력: Data

- ▶ 대개 8bit
- ▶ 저장하기 위한 데이터를 입력하기 위해 혹은 저장된 데이터를 출력하



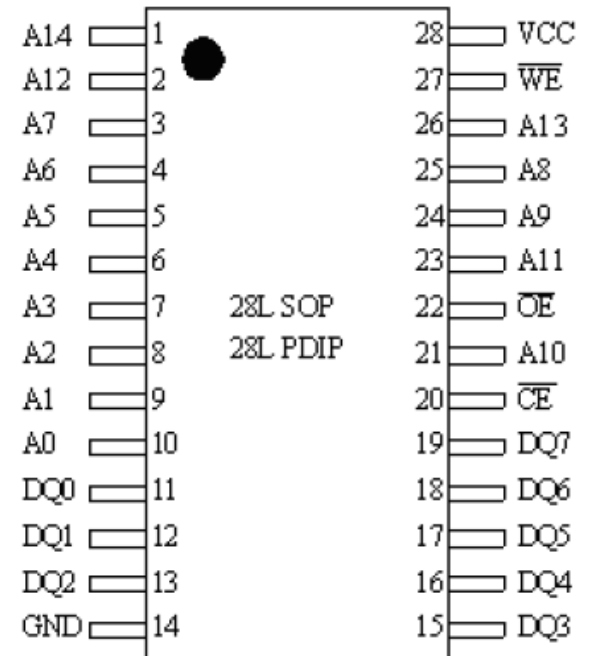
예: 32K bytes SRAM chip

▶ CS18LV02565

- ▶ 32K bytes SRAM
- ▶ Address inputs: A14~A0 (15bit)
- ▶ Data : 8bit
- ▶ Control inputs: /CE, /WE, /OE

■ TRUTH TABLE

Mode	/CE	/WE	/OE	DQ0~7	Vcc Current
Standby	H	X	X	High Z	I_{CCSB}, I_{CCSB1}
Output Disabled	L	H	H	High Z	I_{CC}
Read	L	H	L	D_{OUT}	I_{CC}
Write	L	L	X	D_{IN}	I_{CC}



Copied from Datasheet