

# 마이크로컴퓨터구조 개요

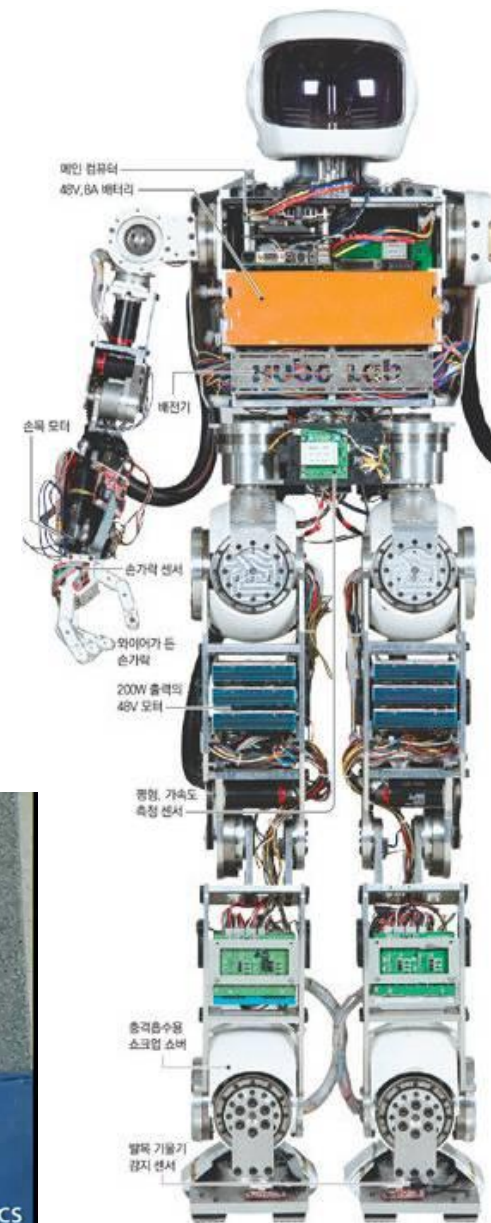
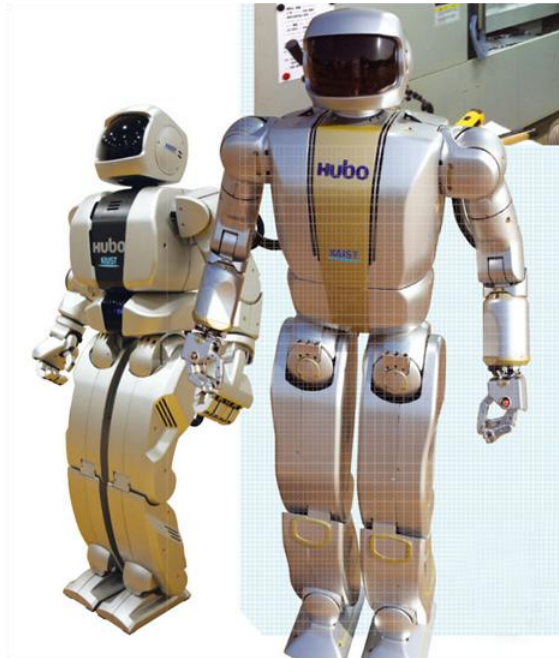


# 1. 마이크로컴퓨터구조 교과 소개

# 마이크로컴퓨터 내장(Embedded) 시스템



# 마이크로컴퓨터 내장 시스템



## 교과목 목표

1. 마이크로컴퓨터 H/W 구조?
2. 마이크로컴퓨터 동작 원리?
3. 마이크로컴퓨터(임베디드 제어기) H/W 설계방법?
4. 마이크로컴퓨터 (임베디드 제어기) S/W 프로그래밍?

### \* 임베디드 제어기(Embedded Controller)

: 기기(장치/기계)안에 내장되어 그 기기의 모든 동작을 제어하는 컴퓨터 기반 모듈

### \* 임베디드 시스템(Embedded System)

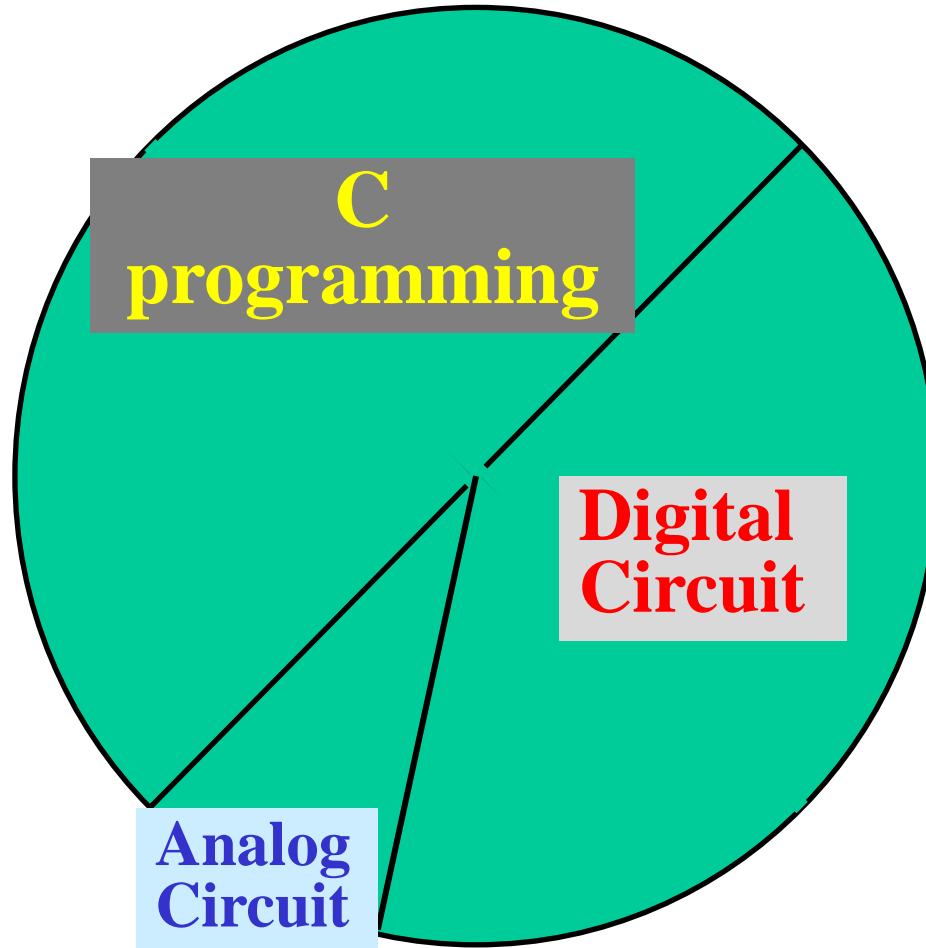
: 임베디드 제어기를 내장한 기기(장치/기계)

# 교과목 구성

기간	1학기			
	3주	4주	6주	2주
내용	컴퓨터구조	Cortex-M4구조, 프로그래밍환경구축	<b>GPIO/INT/LCD</b>	Project
	이론	이론/실습	이론/실습	실습

기간	2학기			
	5주	3주	3주	4주
내용	TIIMER(CNT, PWM). <b>Motor</b>	ADC,DAC, <b>Sensor</b>	TCP/IP,USART, <b>Communication</b>	Project
	이론/실습	이론/실습	이론/실습	실습

# 마이크로컴퓨터구조의 선수교과목들





# 마이크로컴퓨터구조 교과의 선수지식

## C programming

대분류	기초	연산자	제어문	응용
소분류	전처리기/변수/상수/내장함수	산술/대입/비트/관계/논리/조건	if/switch/for/while	array/pointer/function

## Digital circuit

대분류	Gate	Combinational circuit	Sequential circuit	Complex circuit
소분류	AND/OR/NOT/XOR	Adder/Encoder/Decoder/Multiplexer/Demultiplexer/Comparator	Flipflop/Register/Counter	ADC/DAC



## II. 마이크로컴퓨터의 출현(개발) 배경 및 동작 원리

# 일반디지털회로와 컴퓨터회로와의 차이

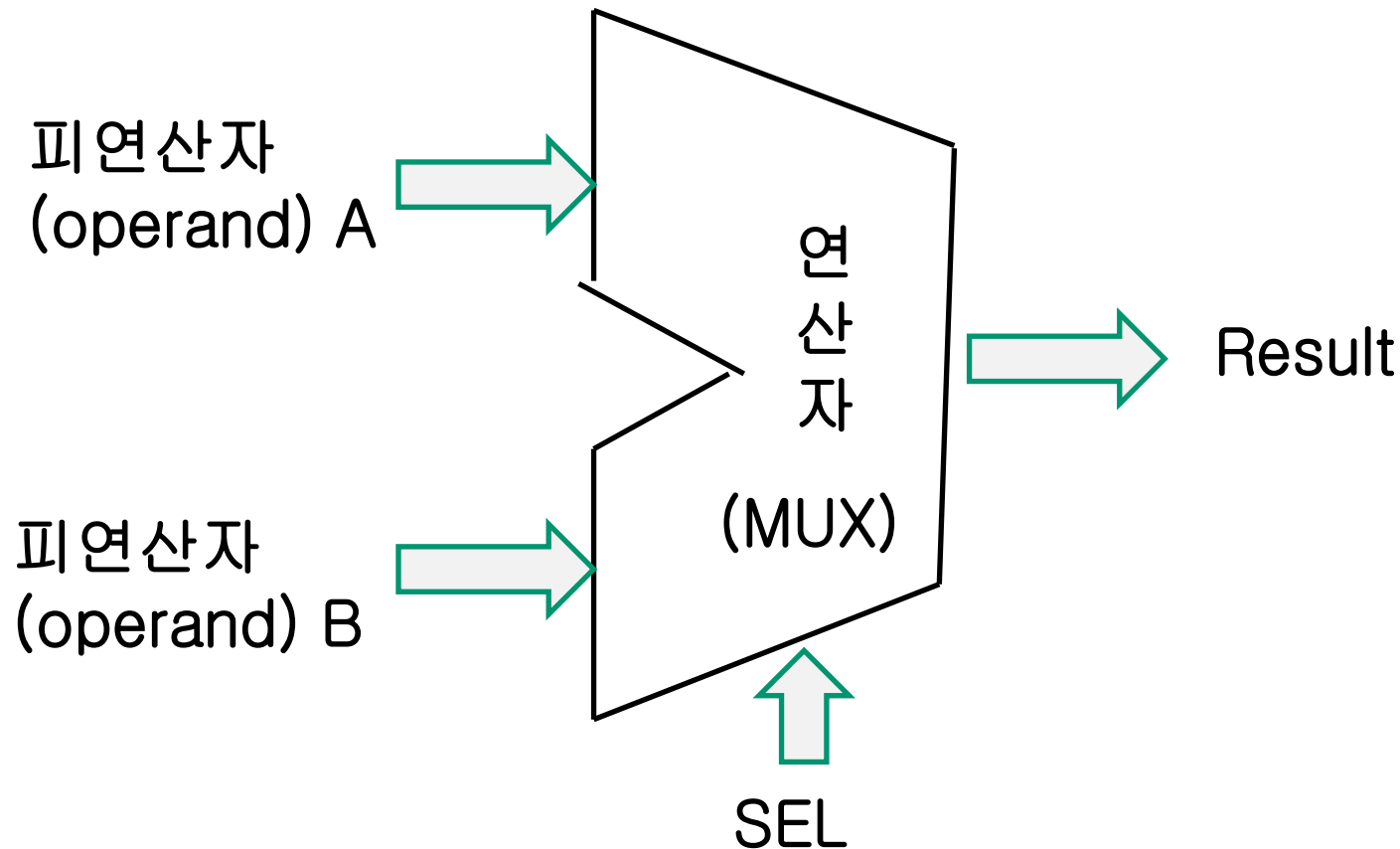
## : 컴퓨터의 출현(개발)배경

### ● History of Electronic circuit

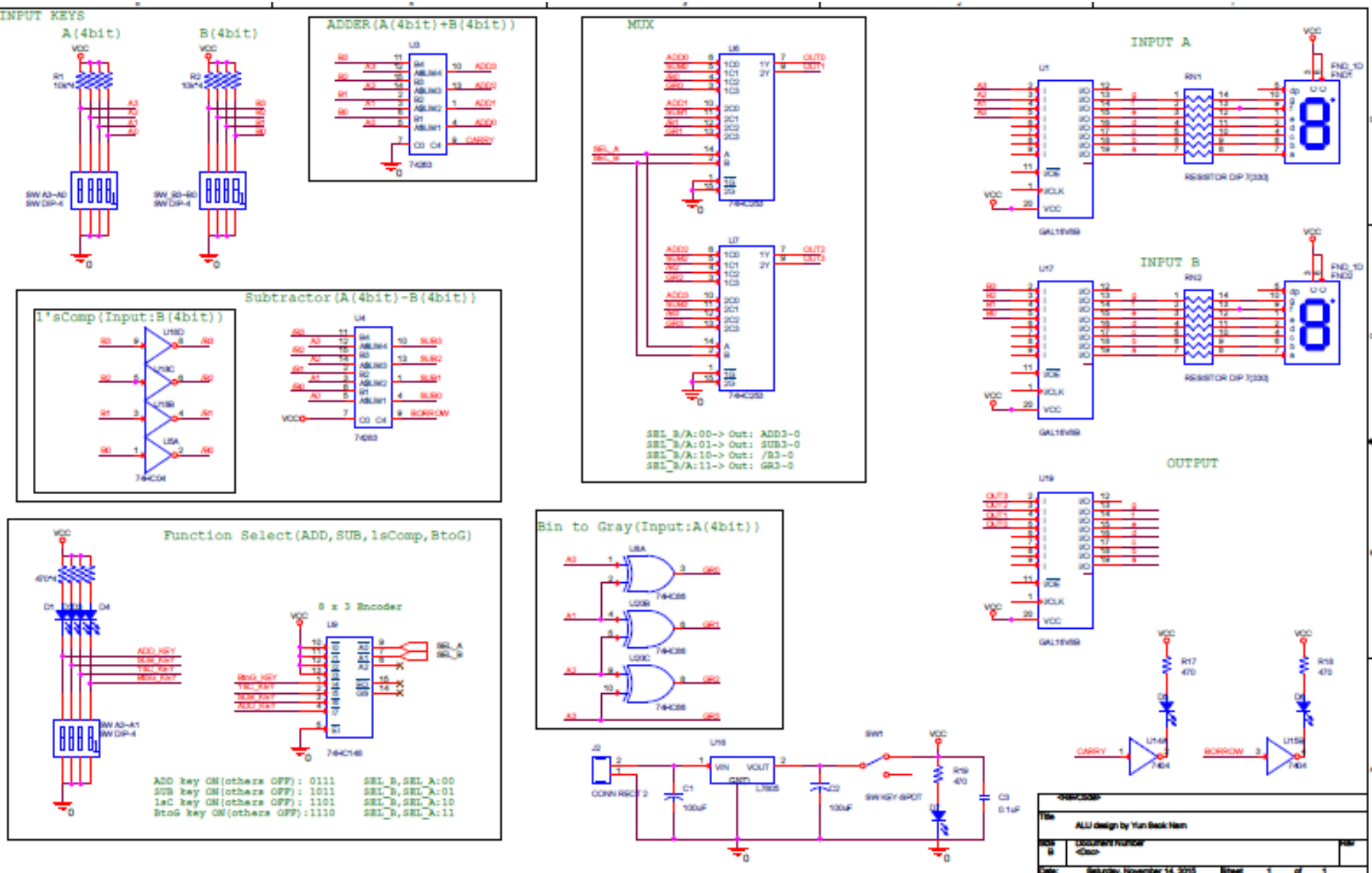
1. Vacuum tube : 'Analog era'
2. Semiconductor
3. Transistor(TR) : Amplification & Switching
4. Switch and Binary number: 'Digital era'
5. Binary operation을 위한 Digital circuit:
  - Basic Gate circuit : AND/OR/NOT/XOR etc.
  - Combinational(조합) circuit : Decoder/MUX/Adder etc.
  - Sequential(순차) circuit : Flipflop/Latch/Register/Counter etc.
6. Digital circuit의 한계:” Hard” or “Non-Flexible”
7. Micro-processor(Computer)의 필요성 : “Soft” & “Flexible”  
'Computer era'

## (예1)일반디지털회로: ALU(Arithmetic & Logic Unit:CPU 핵심요소)

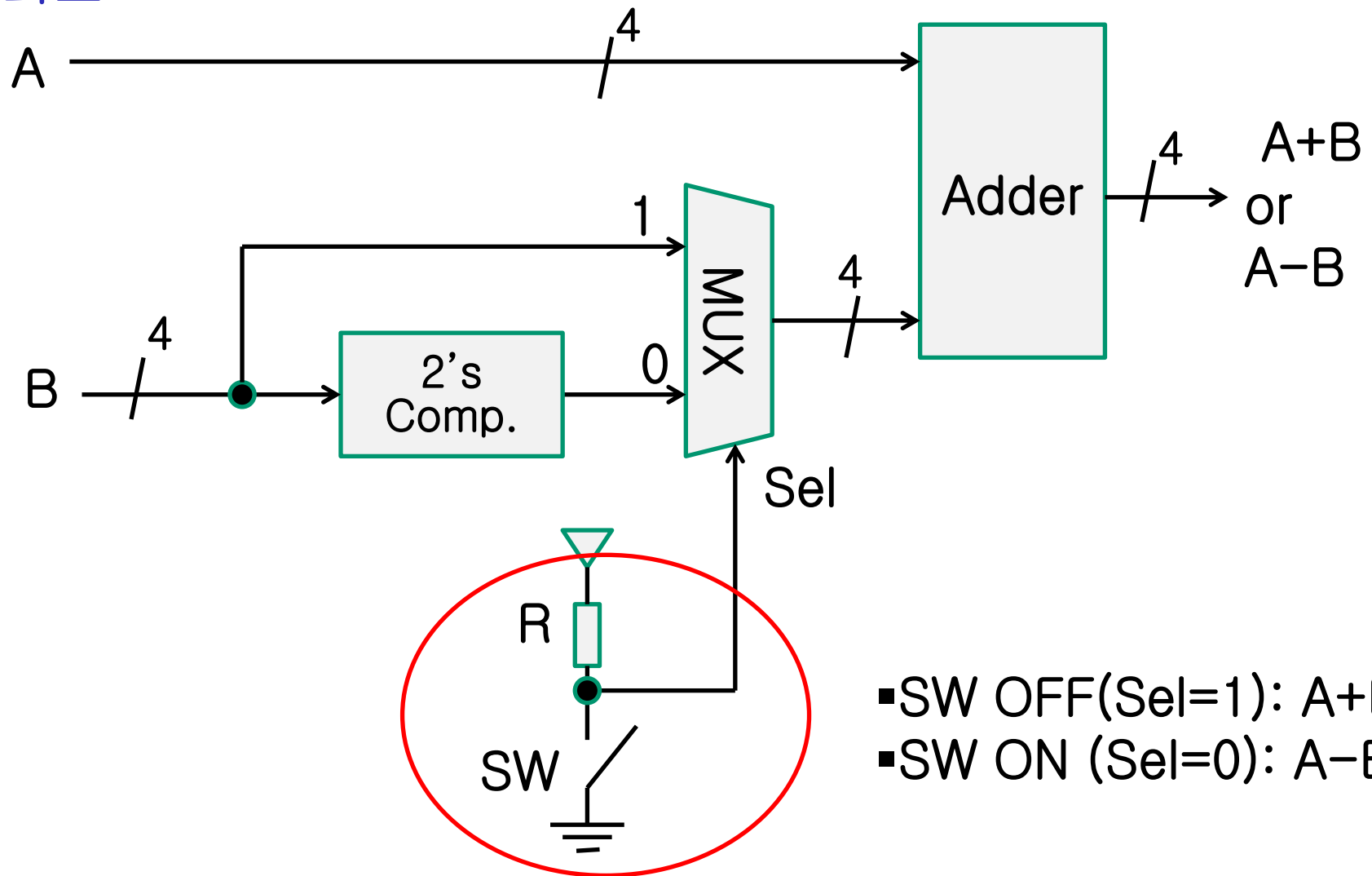
- CPU내의 데이터연산(산술 및 논리)을 담당하는 핵심요소(MUX)
- 연산자의 종류: 산술연산(ADD,SUB,MUL,DIV), 논리연산(NOT, AND, OR, XOR 등), 비트연산(SHIFT,BIT-AND, BIT-OR 등) 등
- \* SEL 신호(제어신호)에 연산자 선택



(예1)일반디지털회로: ALU(Arithmetic & Logic Unit:CPU 핵심요소)

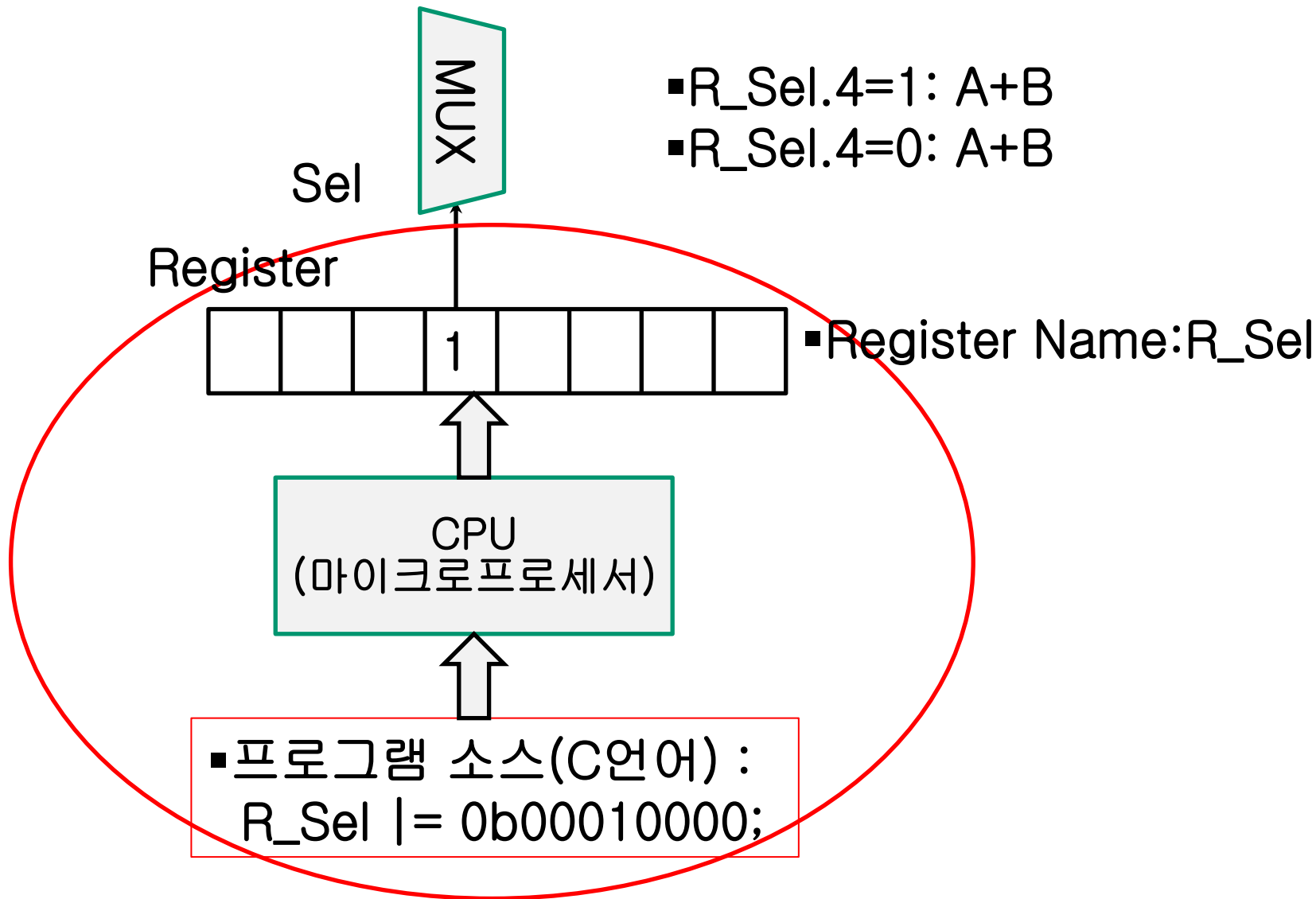


# (예1) 일반디지털회로(Non-CPU)의 기능변화의 한계: $A+B$ or $A-B$ 연산 회로



➤ 한계: 회로의 기능을 바꾸려면 하드웨어 조작이 필요

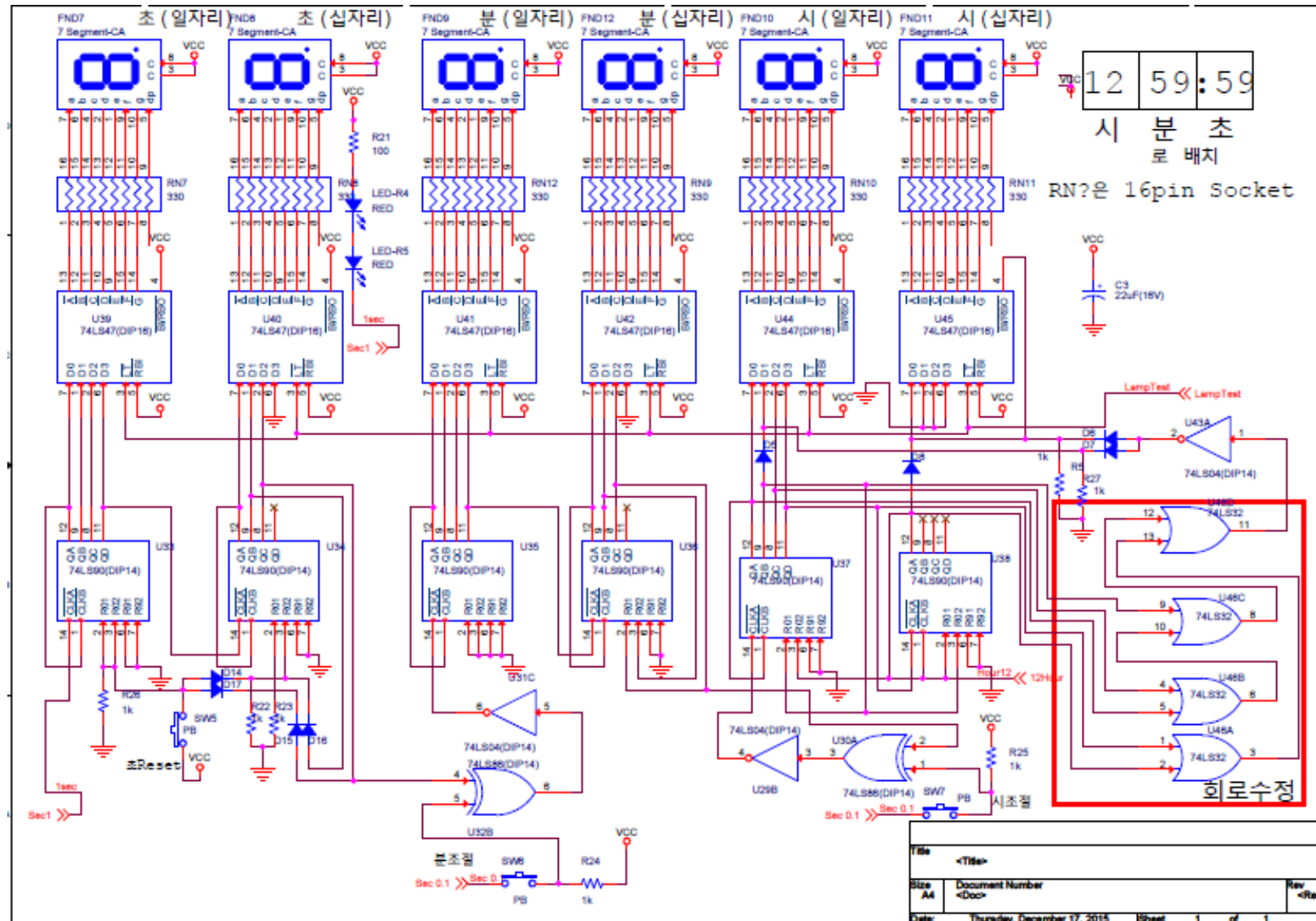
## (예1) (마이크로)프로세서회로(CPU)의 도입:A+B / A-B 연산회로



➤ 한계극복: 회로의 기능을 **소프트웨어**만으로 변경 가능

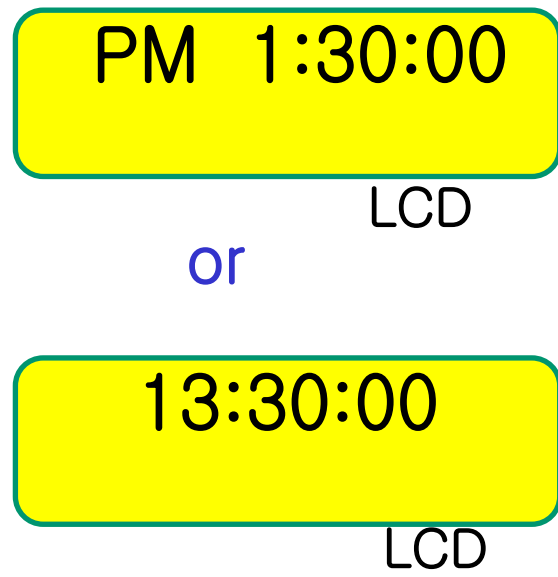
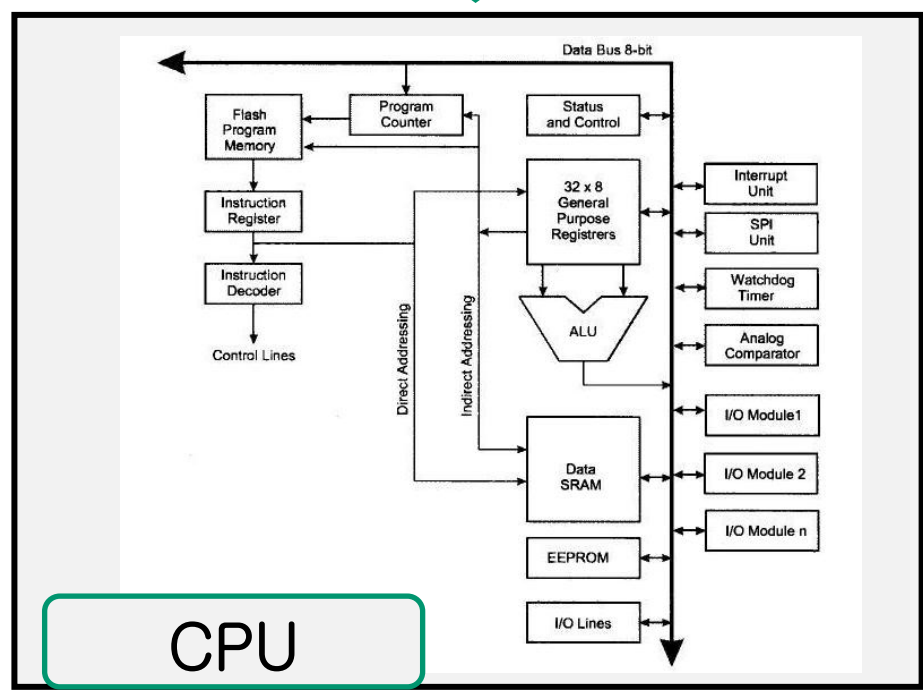
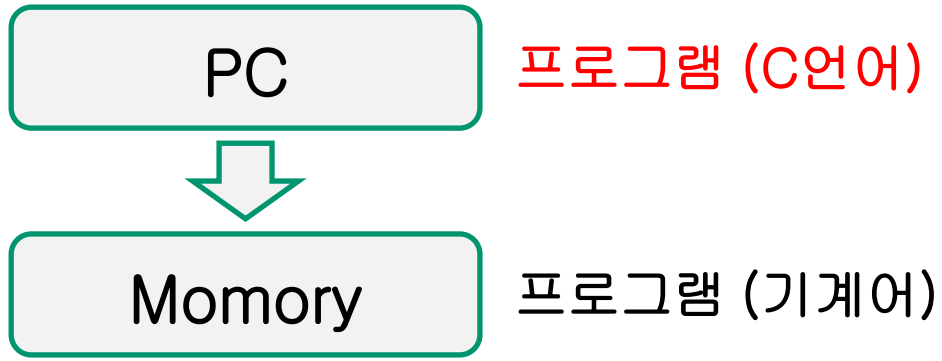
## (예2) 일반디지털회로: Digital Clock

- 한계: 한가지 기능(only H/W), 시간표시 변경 한계





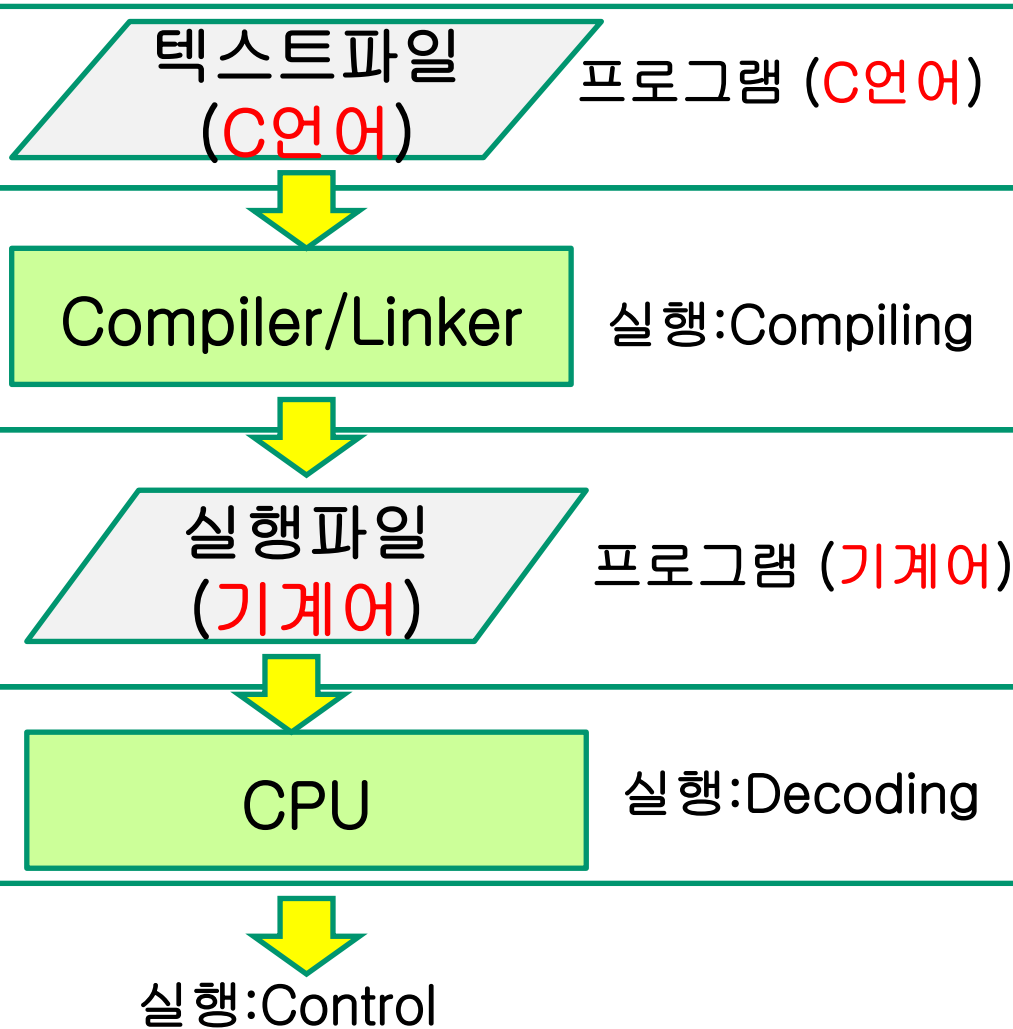
(예2) (마이크로)프로세서회로(CPU)의 도입  
 : 무한 기능(H/W + S/W), 시간표시 자유로운 변경 가능



## ● 컴퓨터 주요 용어

- **컴퓨터(Computer)** : S/W에 의해 H/W를 변화시키는 (H/W 회로 또는 H/W 동작 변환) 전자 기기
- **컴퓨터의 3가지 구성요소**: CPU, Memory, IO-peripherals
- **CPU**(Central Processing Unit)  
: 여러 명령어들로 이루어진 프로그램내의 명령어들을 한개씩 읽어 해독하고, 명령에 해당하는 컴퓨터 각 요소들을 제어하는 신호를 발생하는 컴퓨터의 핵심 모듈(회로)
- **프로세서**(Processor): CPU
- **마이크로프로세서**(Micro-Processor or u-Processor)  
: 원칩(One-chip:IC)으로 만들어진 프로세서 또는 CPU
- **마이크로컴퓨터**: 마이크로프로세서로 구성된 컴퓨터
- **원칩 마이크로컴퓨터**: 원칩으로 된 마이크로컴퓨터
- **MCU**(Micro Controller Unit): 임베디드 제어를 목적으로 제조된 원칩 마이크로컴퓨터

# ● 컴퓨터 프로그래밍



	PC 프로그래밍	uC 프로그래밍
파일 위치	DRAM	DRAM(PC)
파일 위치	DRAM	DRAM(PC)
파일 위치	DRAM	DRAM(PC) → Flash(uC)
CPU 위치	PC	uC
제어 대상	PC	uC

\*uC: 마이크로컴퓨터

## ● 컴퓨터 프로그래밍

### • C 언어

(예) `char A,B,C; A=A+B; C=A;`

### • 컴파일러: C언어 문장을 기계어 문장으로 변환하는 프로그램

–(Self) compiler: C 파일이 존재하는 컴퓨터의 CPU가 인식(해석) 가능한 기계어 파일로 변환하는 컴파일러 (예: PC에서 실행하는 파일을 생성하는 컴파일러, Visual studio내장 compiler)

–Cross compiler: 타 컴퓨터의 CPU가 인식(해석) 가능한 기계어 파일로 변환하는 컴파일러 (예: 모든 임베디드 컴퓨터용 실행파일을 생성하는 컴파일러, IAR workbench내장 compiler(Cortex용 컴파일러))

### • 기계어(Machine language): CPU가 인식하는 이진수 조합으로 된 약속된 코드(8,16,32,64 bit)

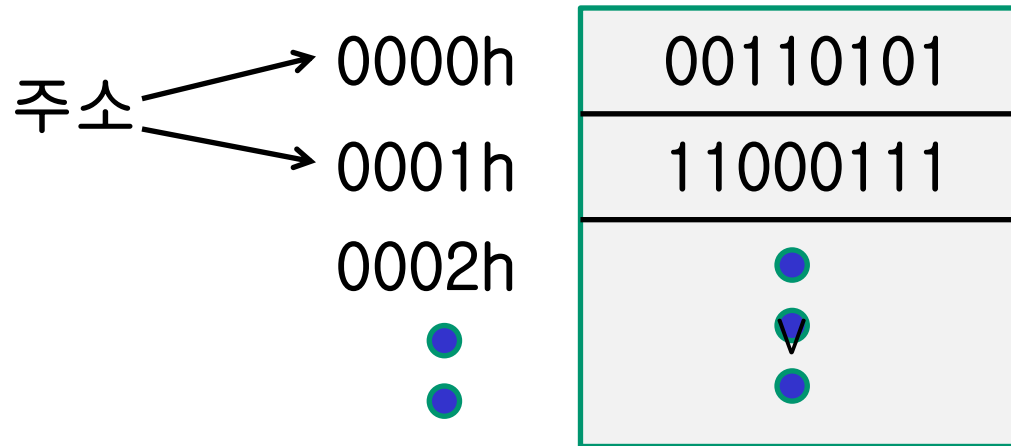
(예) **0**b00110101(or 00110101**B**): “ADD” (A=A+B, A,B: Register)

0b11000111(or 11000111B): “MOV” (C←A, A,C: Register)

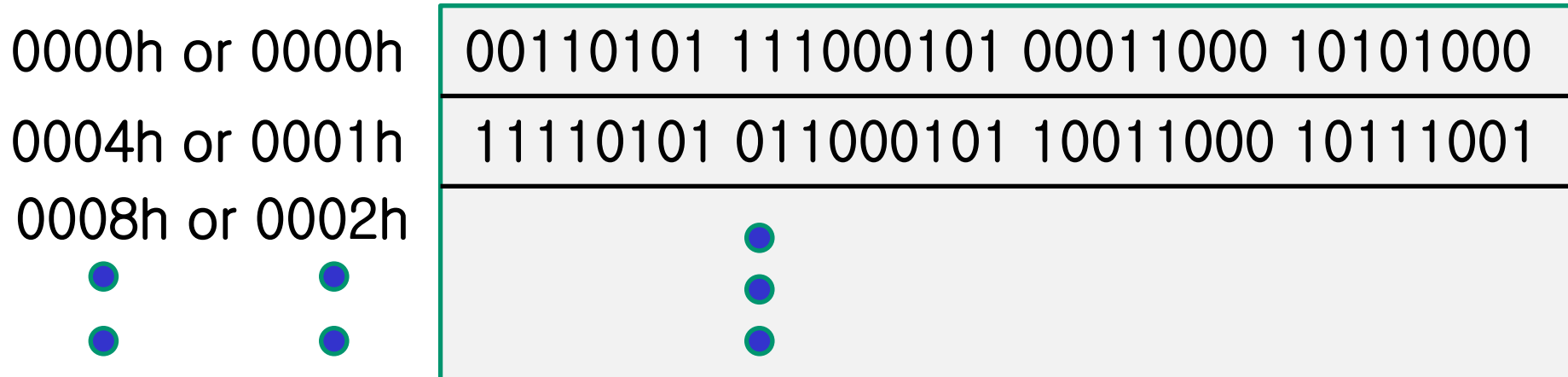
## ● 컴퓨터 프로그래밍

### • 기계어 파일의 저장 형태

-8bit 경우 : 명령어  $2^8=256$ 가지



-32bit 경우 : 명령어  $2^{32}=4G$



## ● 컴퓨터 프로그래밍

- 기계어 파일의 저장 매체 : RAM, ROM, Flash(반도체 메모리)
  - \* PC 경우: HD/SSD에 저장했다가 실행할 때 DRAM에 load된 후 실행됨
  - \* 임베디드컴퓨터 경우: Flash/ROM에 저장되었다가 바로 실행됨
- 메모리 특성
  - 기본단위: 1-byte(8bit)
  - 기본저장단위: 1-byte(8), 2-bytes(16), 4-bytes(32), 8-bytes(64)
  - 각 저장단위마다 주소(Address) 배정되어 관리
- \*진법표기
  - 2진수: 0b..., ...B
  - 10진수: ...
  - 16진수: 0x..., ...h

## ● 컴퓨터 프로그래밍

- 메모리용량과 주소사이 관계

(Q) 0x0000~0xFFFF 의 주소를 가진 메모리 용량은?

(A) 하나의 16진수는 4개의 2진수(4bit)를 가진다. 경우의 수는  $2^4=16$ . 10bit, 즉 0x000~0x3FF(0b0011 1111 1111)는  $2^{10}=1024=1K$

그러므로 0x0000~0xFFFF 주소영역의 메모리용량은  $2^{16}=64K$   
(즉,  $2^{16}=2^6 * 2^{10} = 64K$ )

(Q) 0x0000~0xFFFFFFFF : 24bit

(A)  $2^4 * 2^{10} * 2^{10} = 16 * K * K = 16M$

(Q) 64M의 용량을 가진 메모리의 가능한 주소영역은?

(A)  $64 * M = 2^6 * 2^{20}$

$= 0b \underline{0011\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111} = 0x3FFFFFFF$

즉 주소영역은 0x00000000 ~ 0x3FFFFFFF

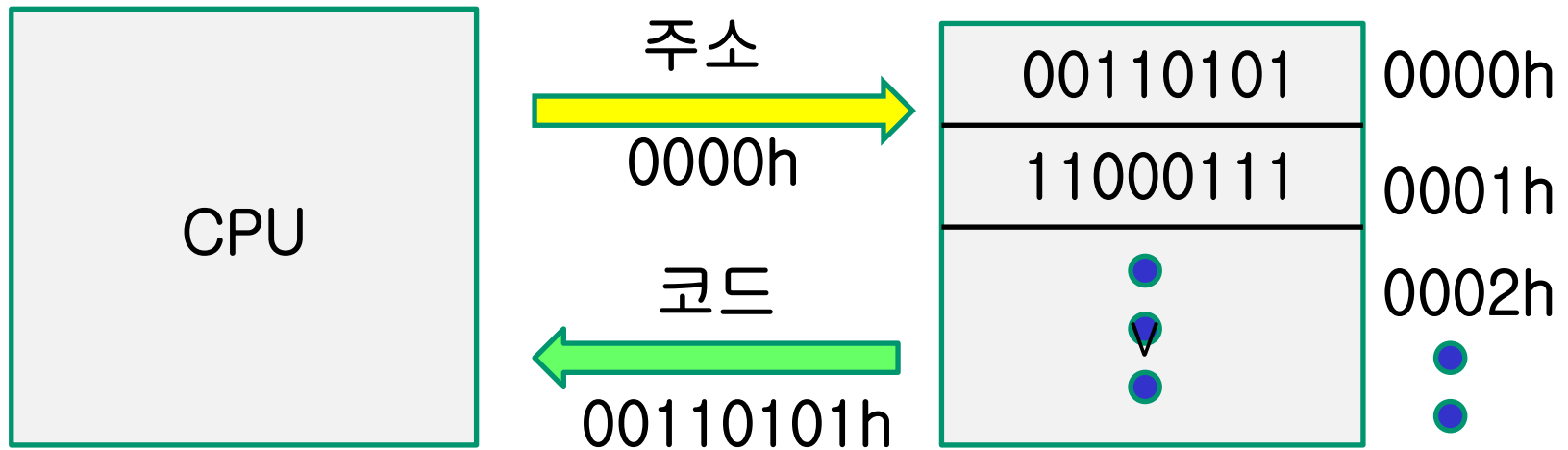
(Q) 4G?



## ● 컴퓨터 프로그래밍

• 기계어 코드(문장)을 CPU로 읽어오기

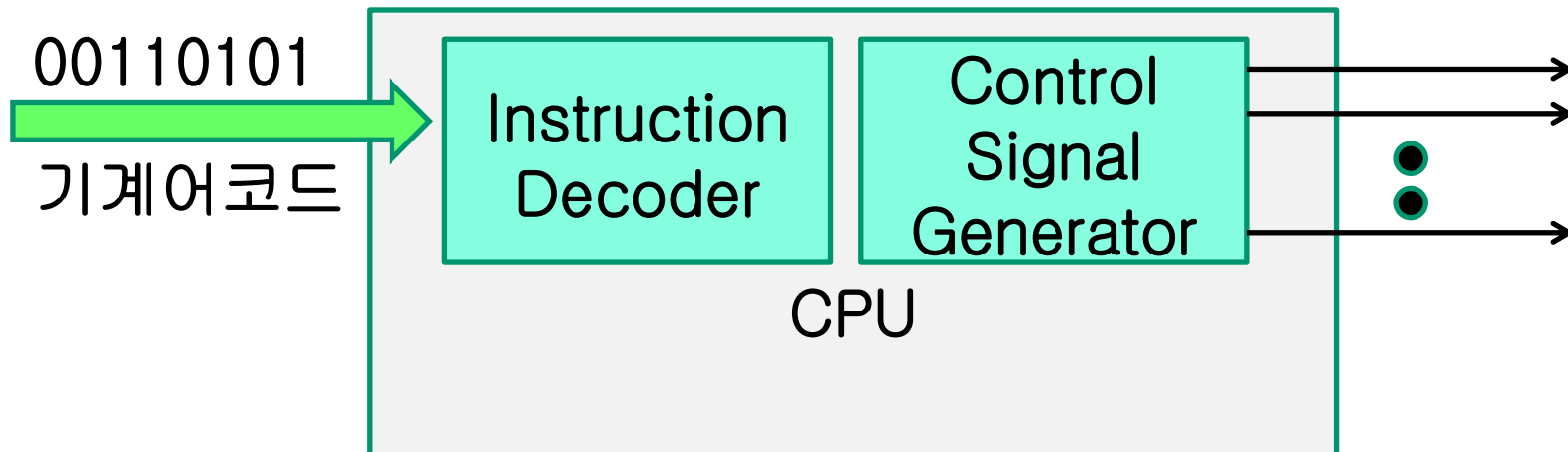
: CPU는 읽어야 할 기계어코드가 있는 주소를 메모리에 전송하고  
메모리는 주소를 받고 해당 주소의 내용물(기계어 코드)를 CPU에  
전송



## ● 컴퓨터 프로그래밍

- 기계어 파일의 해독 및 실행(제어신호 발생)

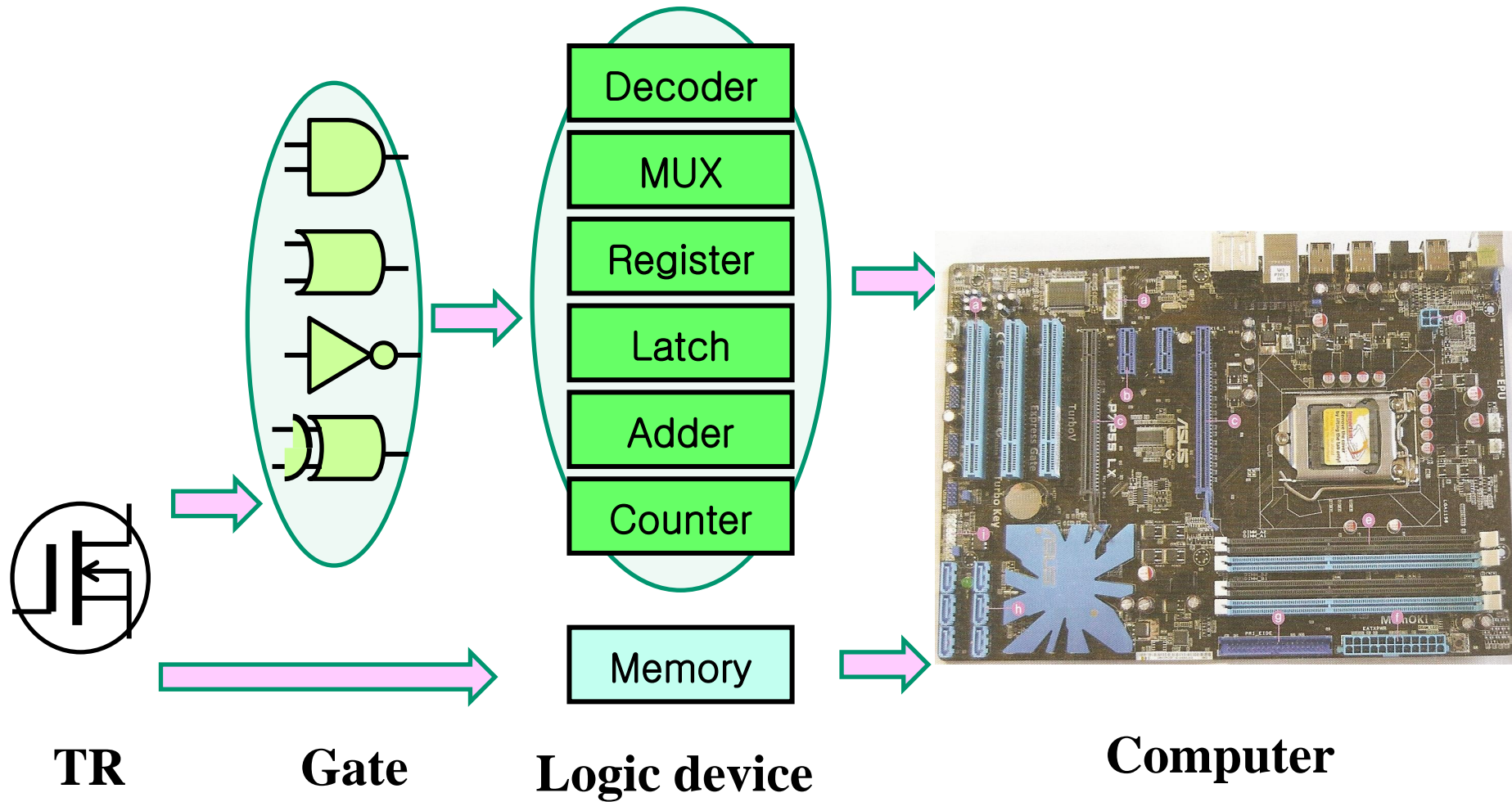
: CPU 내부의 해독기(Instruction Decoder)에서 기계어코드를 해독(decoding)하고 해독된 명령어에 따라 실행해야 할 동작을 위해 순차적으로 제어신호를 CPU는 컴퓨터 내부의 각 모듈(회로)로 전송.



# III. 마이크로컴퓨터구조 요약

# 1. 컴퓨터

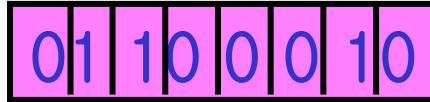
## ■ 컴퓨터 H/W 계층구조



## ■ 컴퓨터의 H/W 요소

-**레지스터**(Register): 데이터를 저장하는 기본소자 (크기:8,16,32,64 bits)

\* 예) 8bit Reg.



-**ALU**(Arithmetic & Logic Unit): CPU내에 위치하고 연산(사칙연산/논리연산/비트연산 등)을 수행

-**Controller**(Command Decoder & Control Signal Generator): CPU내에 위치하고, 명령어(프로그램)를 해독하여 컴퓨터를 동작시키는 각종 제어 신호를 발생

-**Memory**: 데이터나 프로그램을 저장

(Flash memory, ROM, RAM (DRAM, SRAM, SDRAM, DDR SDRAM) )

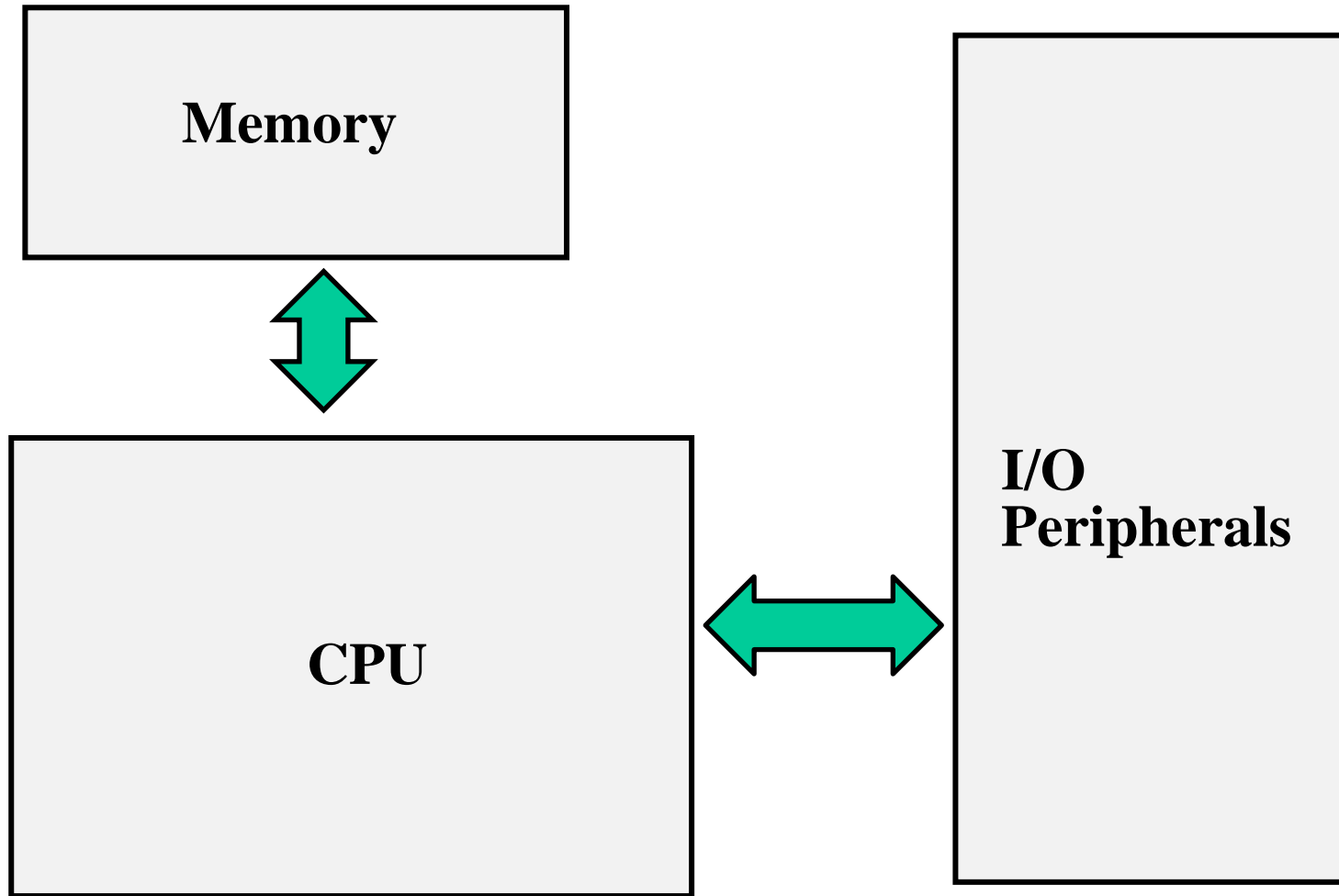
-**I/O peripheral**(입출력 주변장치): 사용자나 다른 컴퓨터와의 인터페이스를 위한 장치

## ■ 컴퓨터의 S/W 요소

-**데이터**: 컴퓨터 동작의 대상

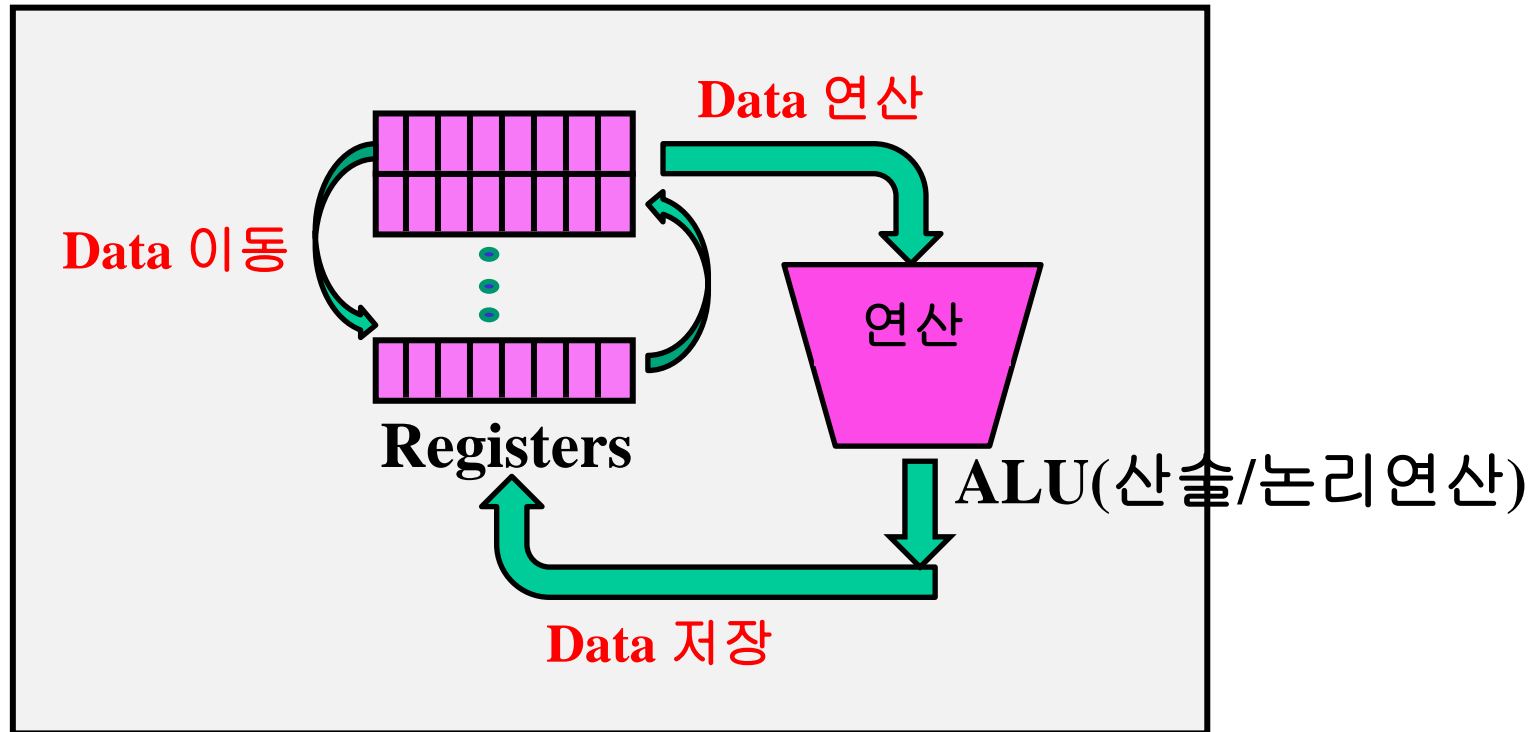
-**프로그램**: 컴퓨터 동작을 실현

## 2. 컴퓨터 구조



**F1. CPU, Memory and I/O peripheral devices**

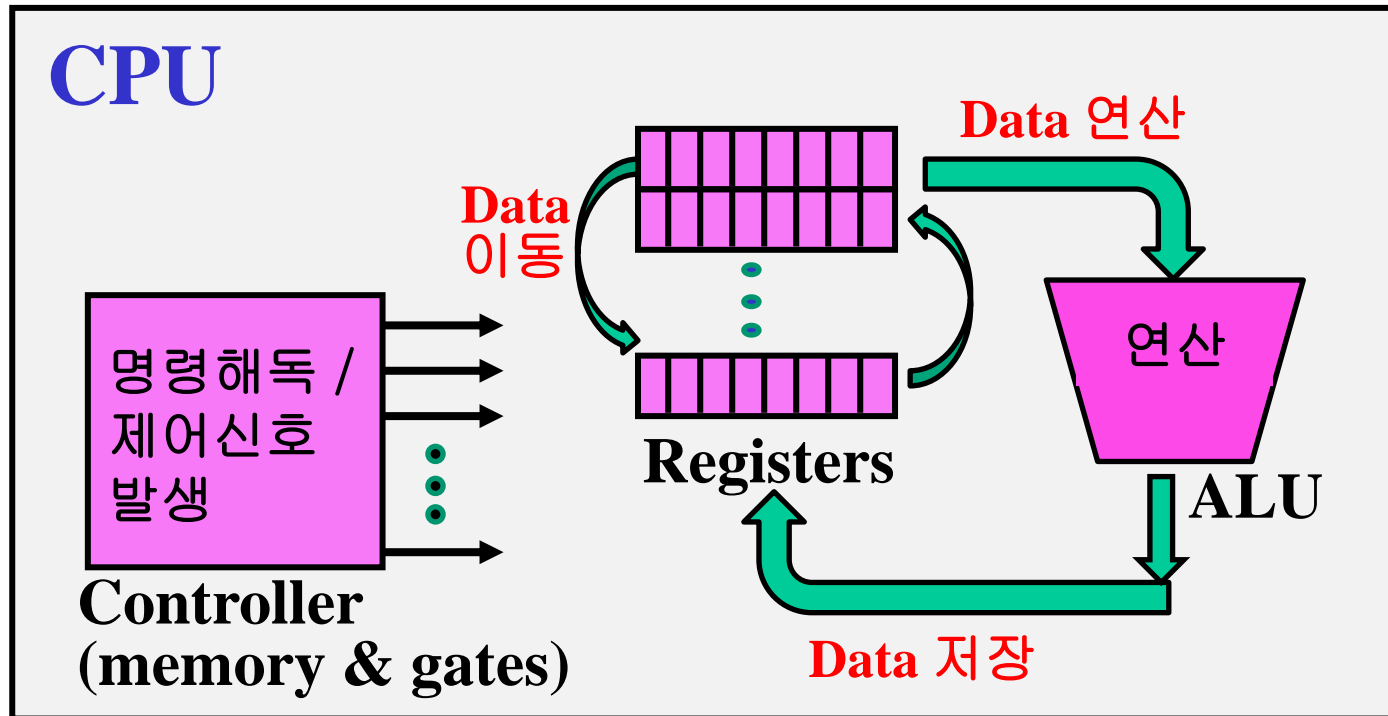
## 2. 컴퓨터 동작-1



**F2. Register와 ALU(산술/논리연산, shift 등)**

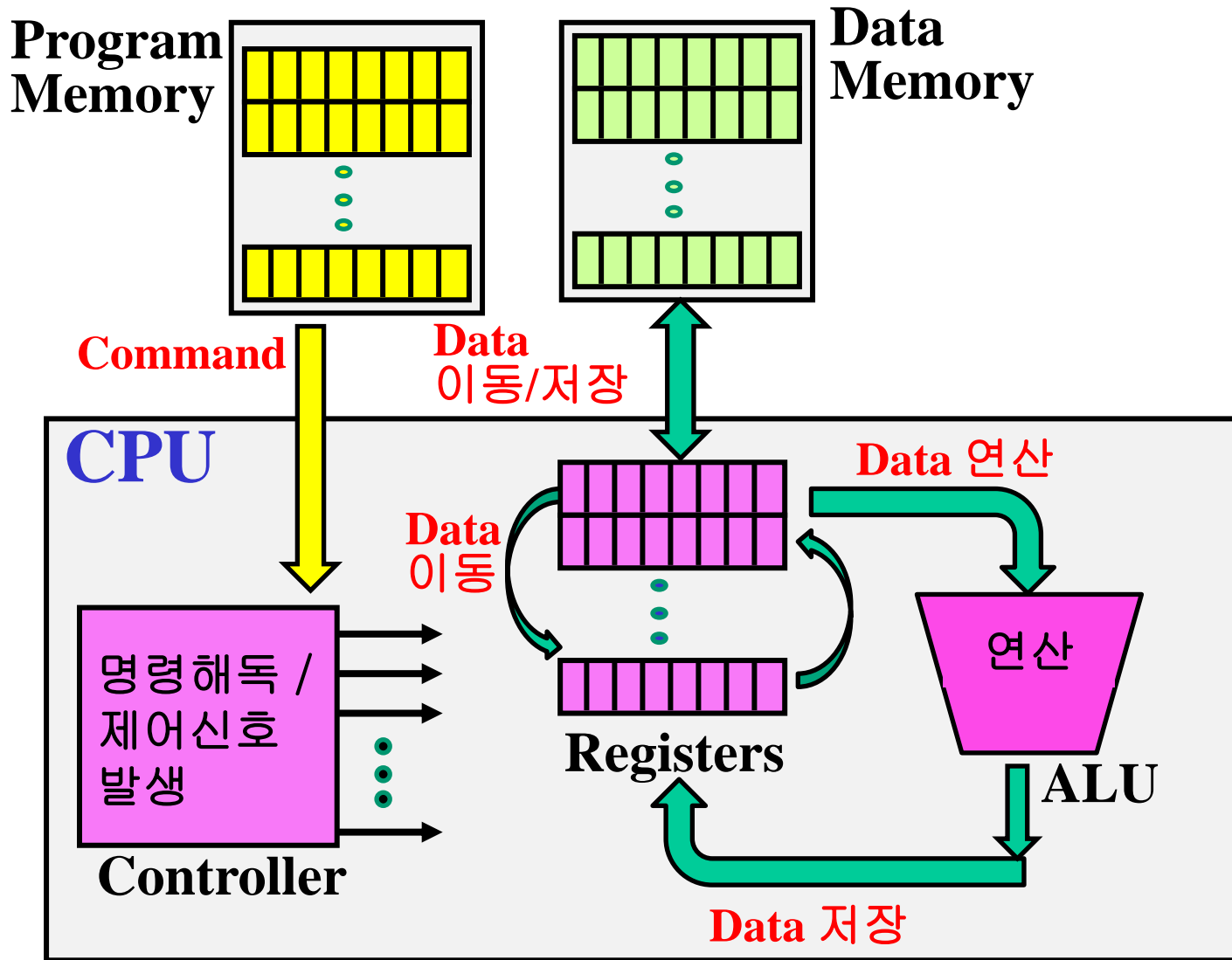


## 2. 컴퓨터 동작-2



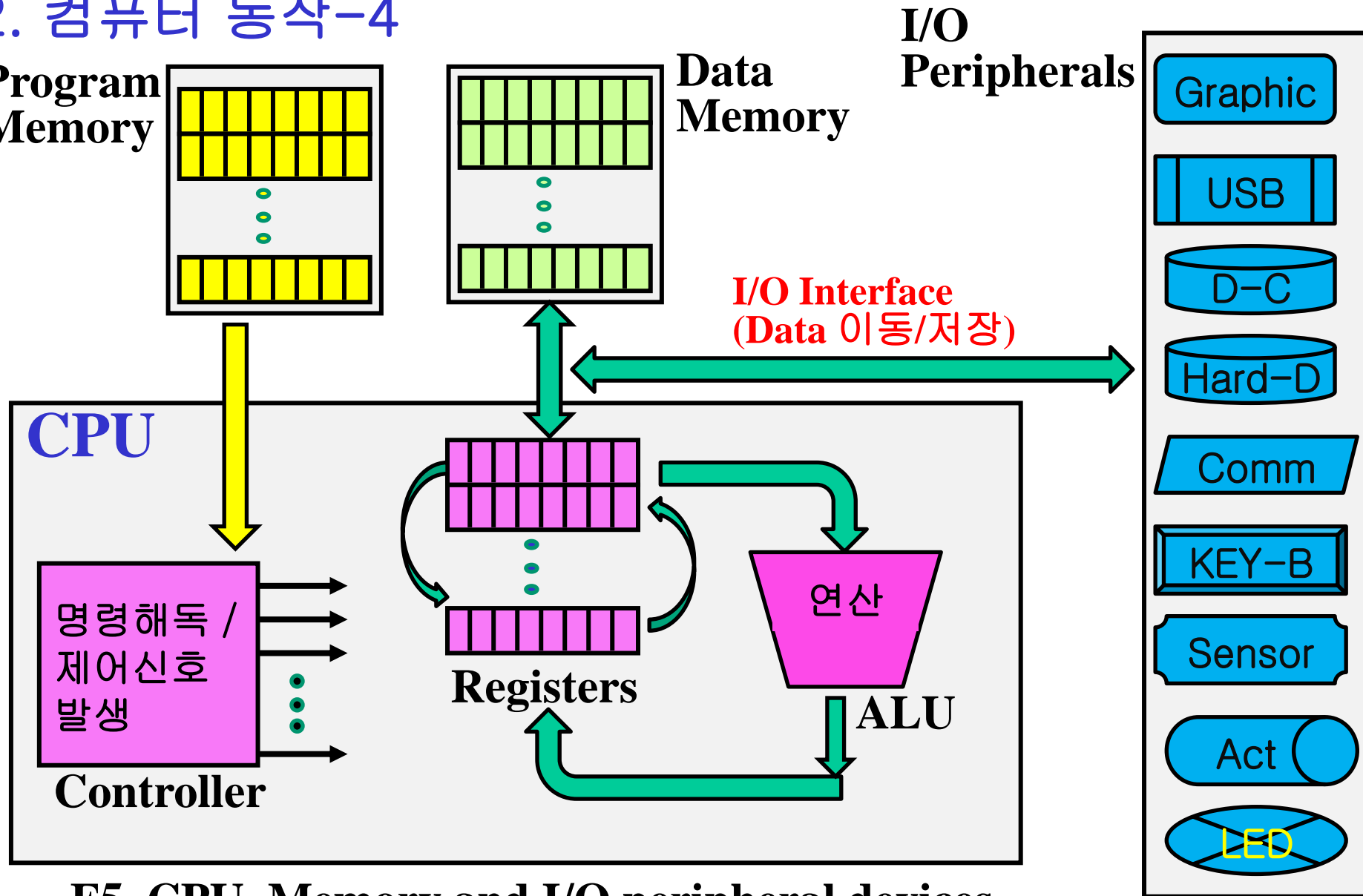
**F3. CPU(Register, ALU, Controller)**

## 2. 컴퓨터 동작-3



F4. CPU and Memory

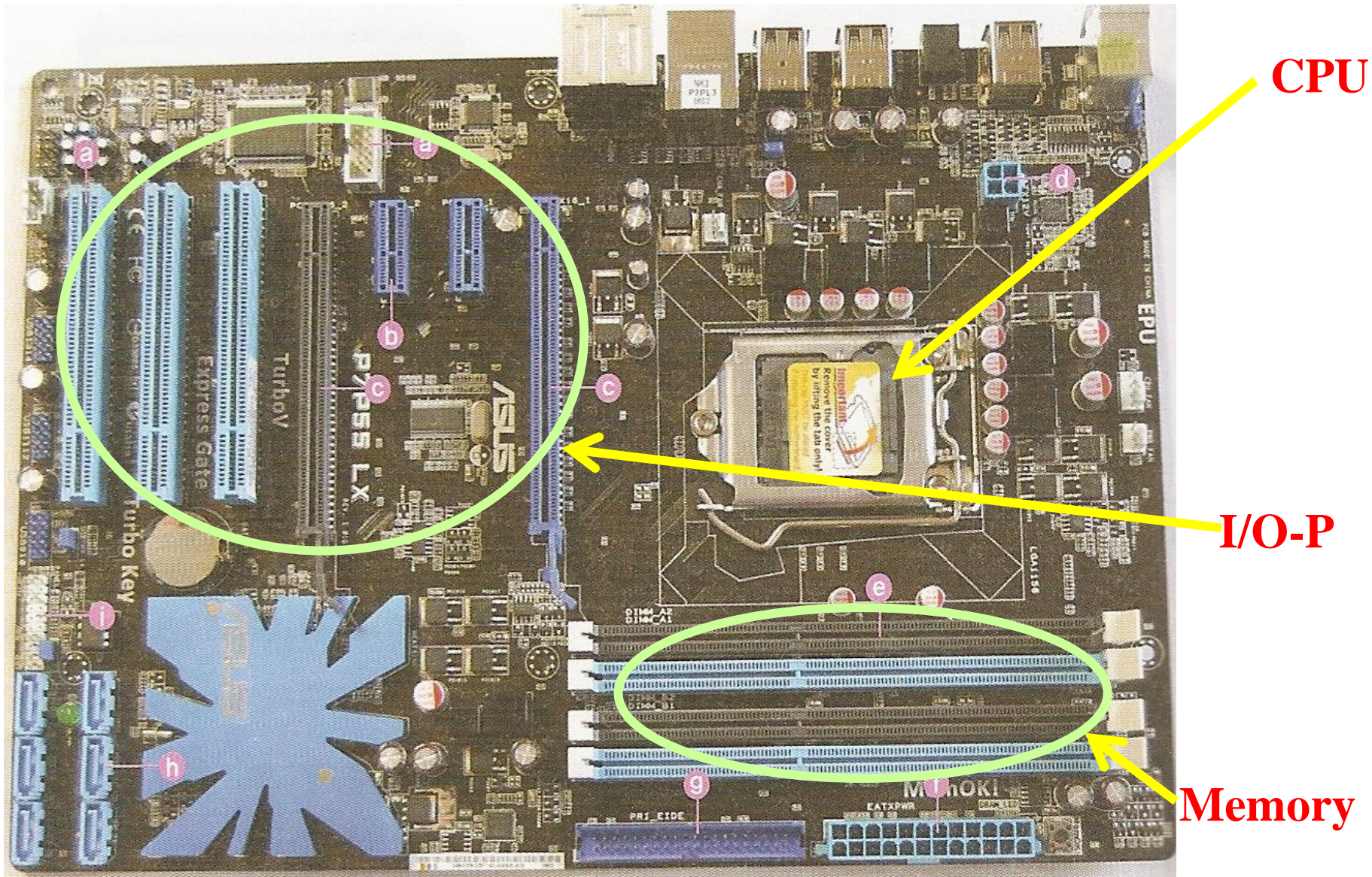
## 2. 컴퓨터 동작-4



F5. CPU, Memory and I/O peripheral devices



### 3. 마이크로컴퓨터 구조-1( PC main board 실물)



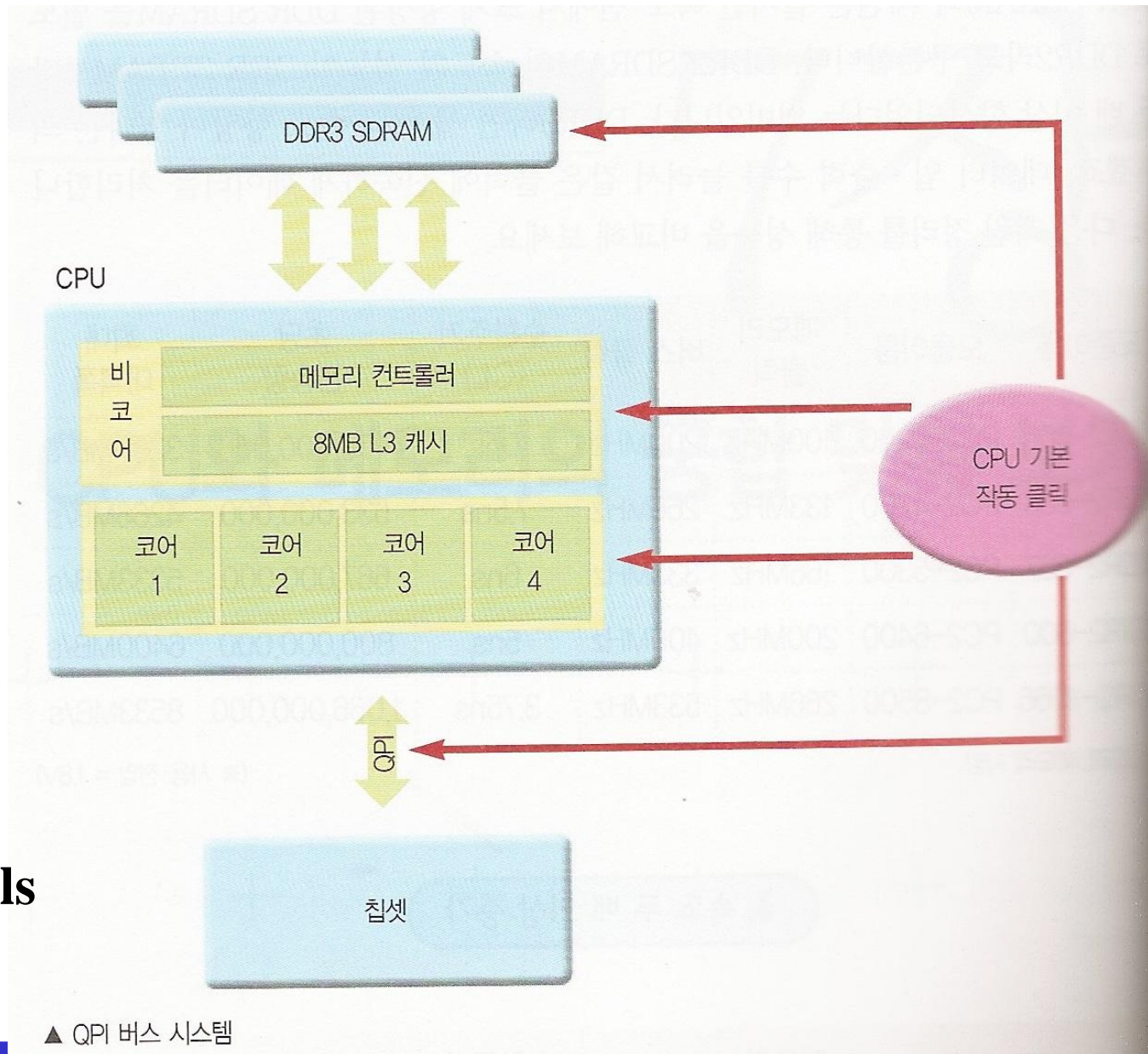


# 3.마이크로컴퓨터 구조-2(PC Main board H/W 블록도)

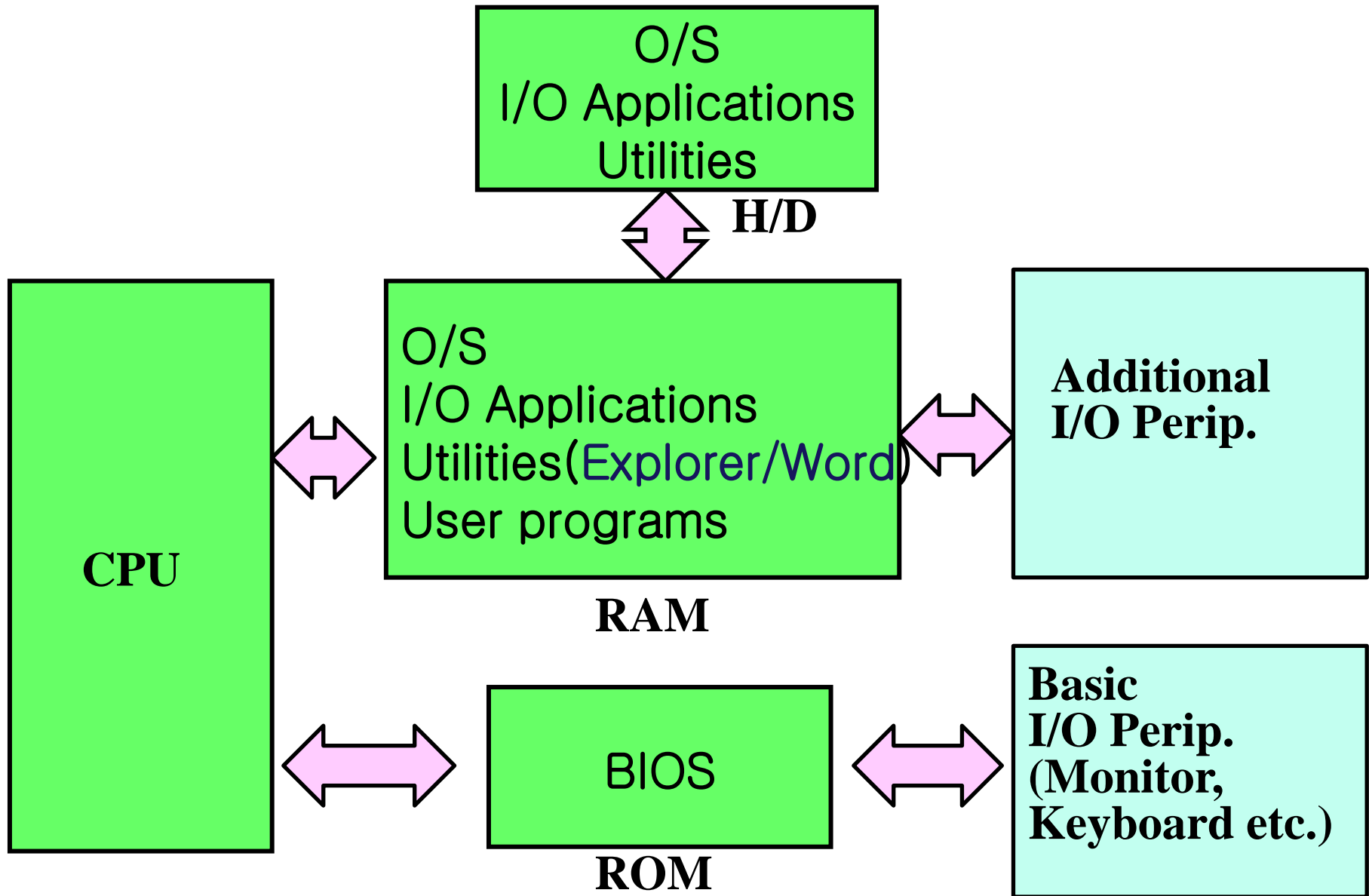
Memory

CPU

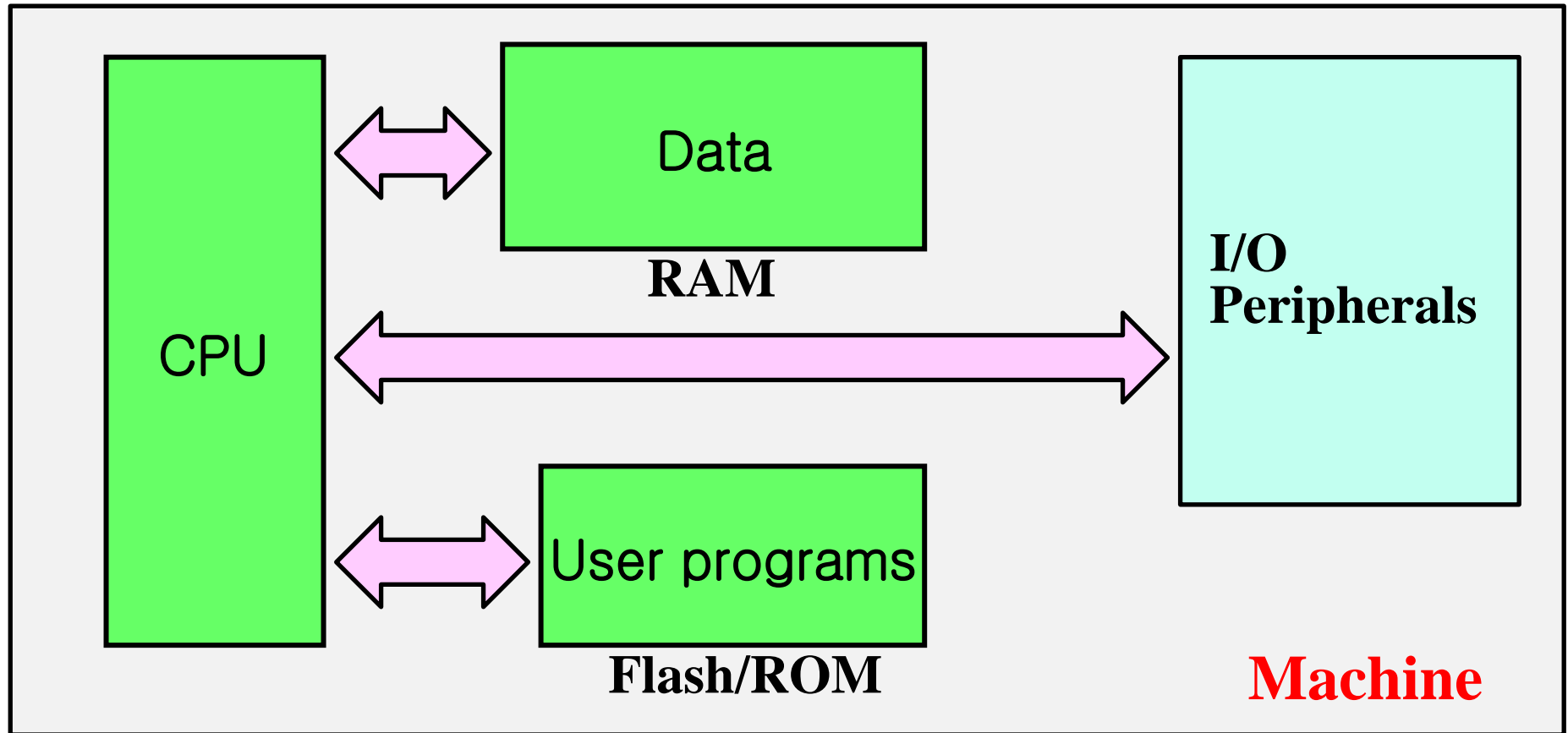
I/O  
Peripherals



### 3. 마이크로컴퓨터 구조-3(PC Main board S/W 블록도 )



### 3.마이크로컴퓨터 구조-4(Embedded Microcomputer)

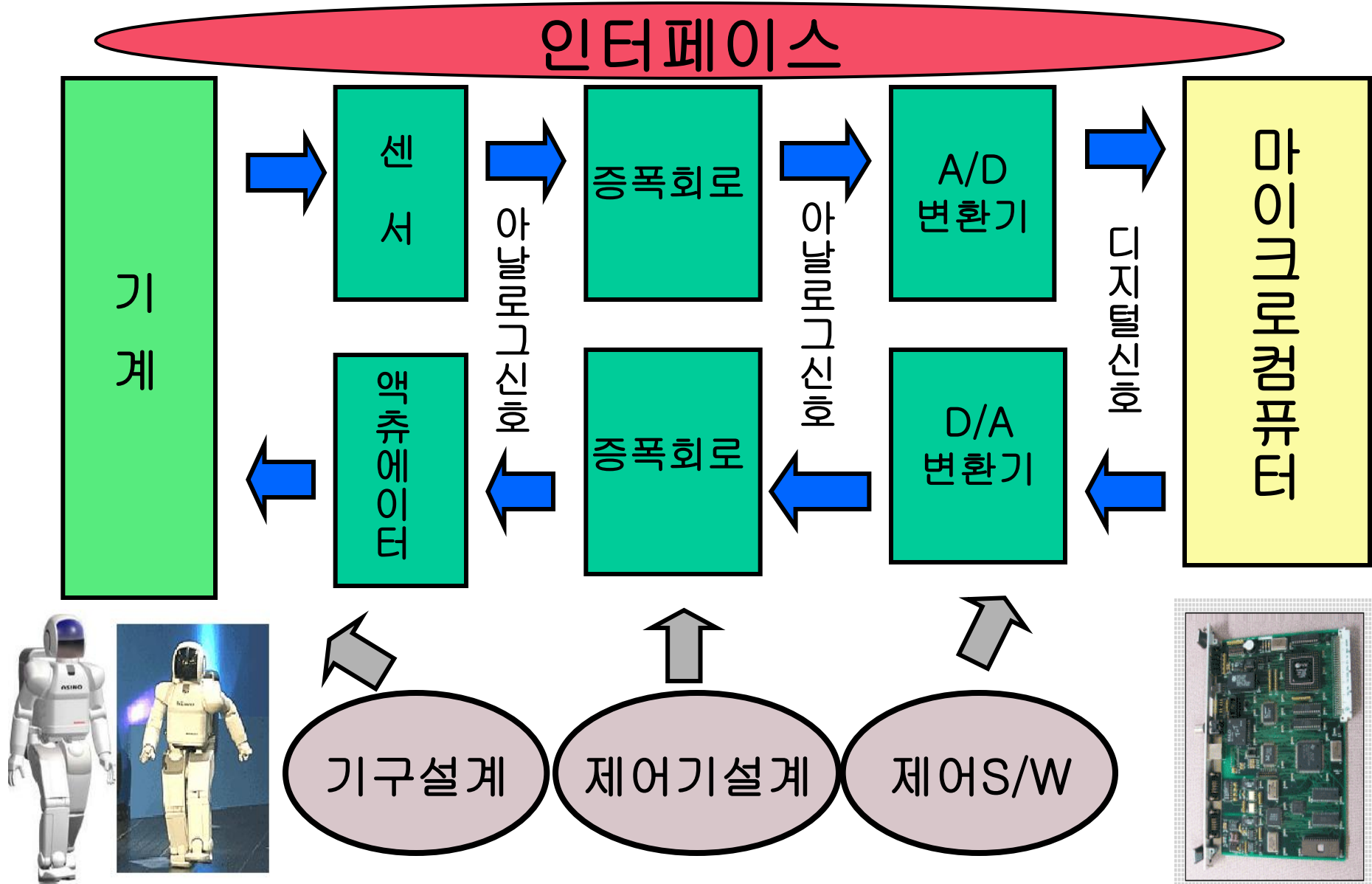


- Small user program: with BIOS
- Big user program: with small O/S(Linux/Android etc)

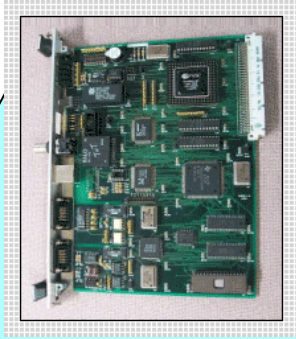


## IV. 마이크로컴퓨터 응용

# 1. 마이크로컴퓨터 응용 - 메카트로닉스 시스템



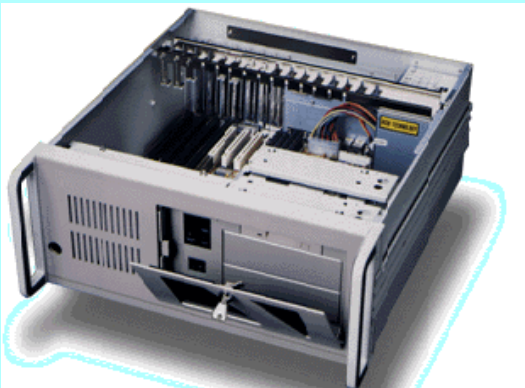
## 2. 마이크로컴퓨터 응용-Robot1



마이크로프로세서  
내장형 보드

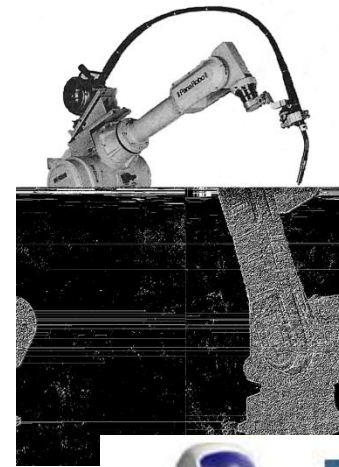


Panel PC



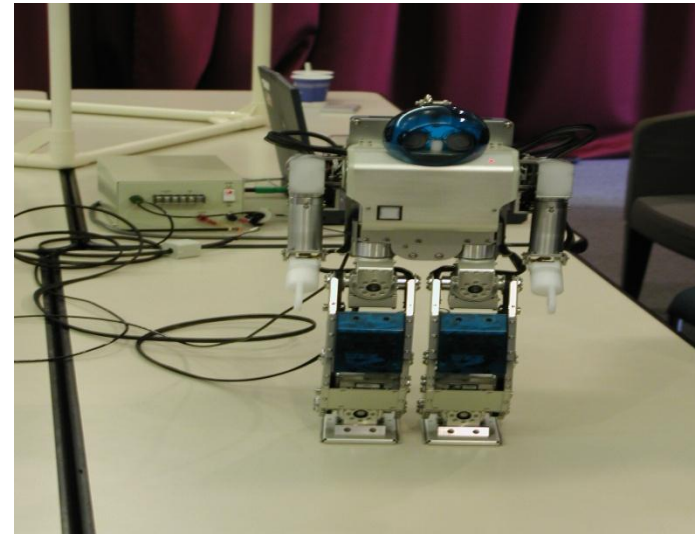
산업용 PC

로봇 제어 및  
사용자 인터페이스

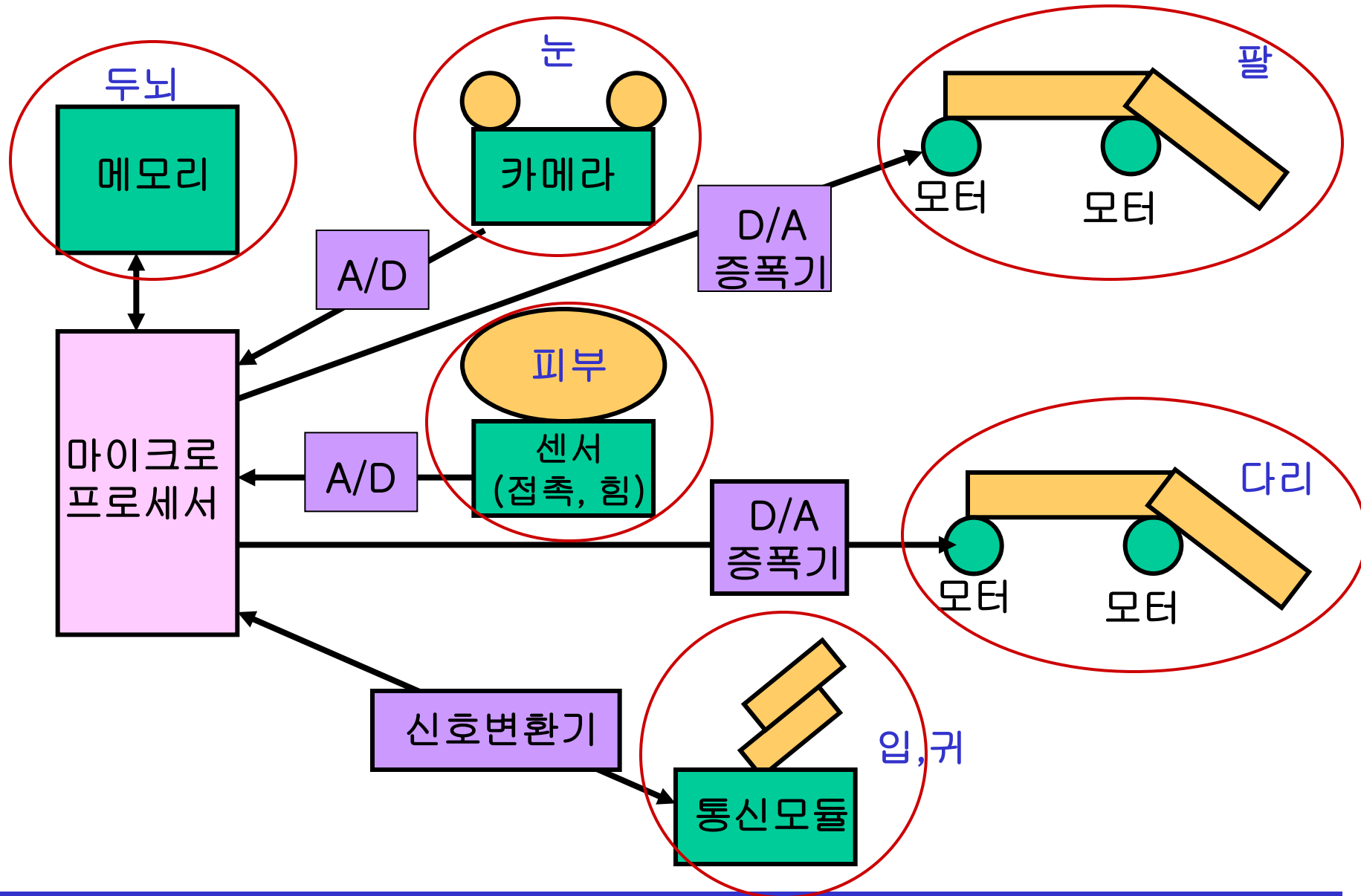


마이크로프로세서 기반 시스템

## 2. 마이크로컴퓨터 응용- Robot2



## 2. 마이크로컴퓨터 응용-Robot(내부구조)

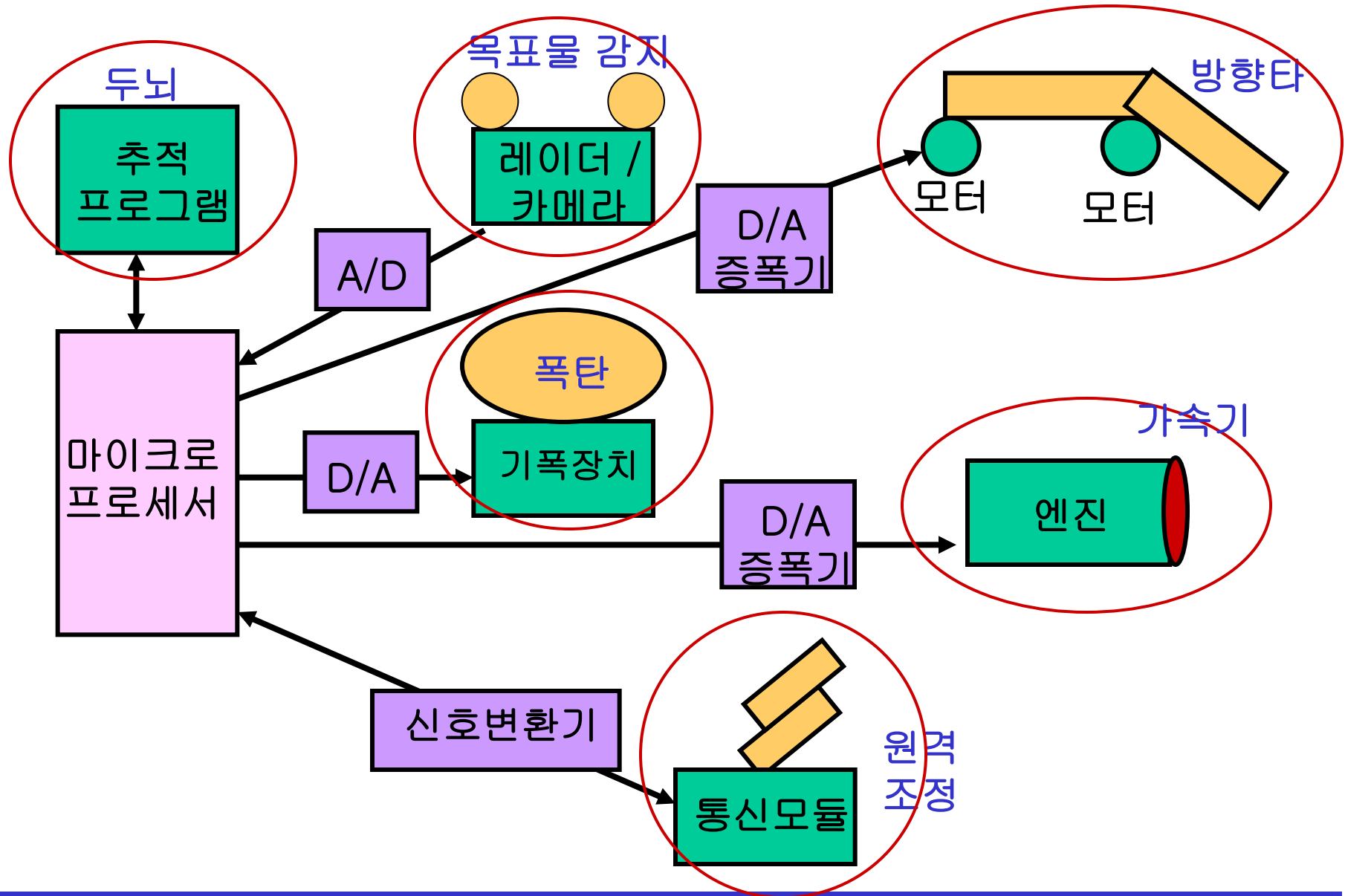




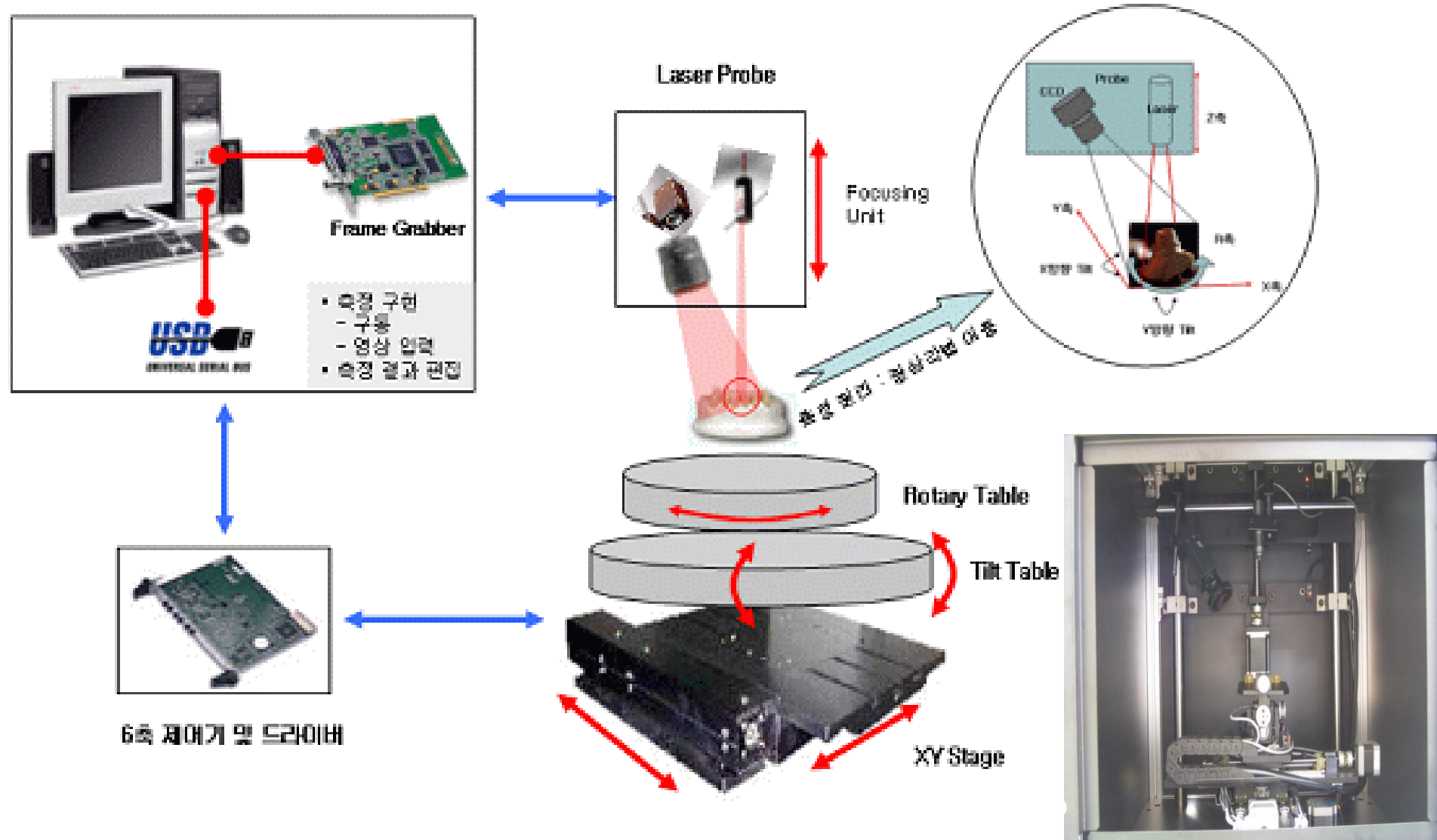
### 3. 마이크로컴퓨터 응용-Missile



### 3. 마이크로컴퓨터 응용-Missile(내부구조)

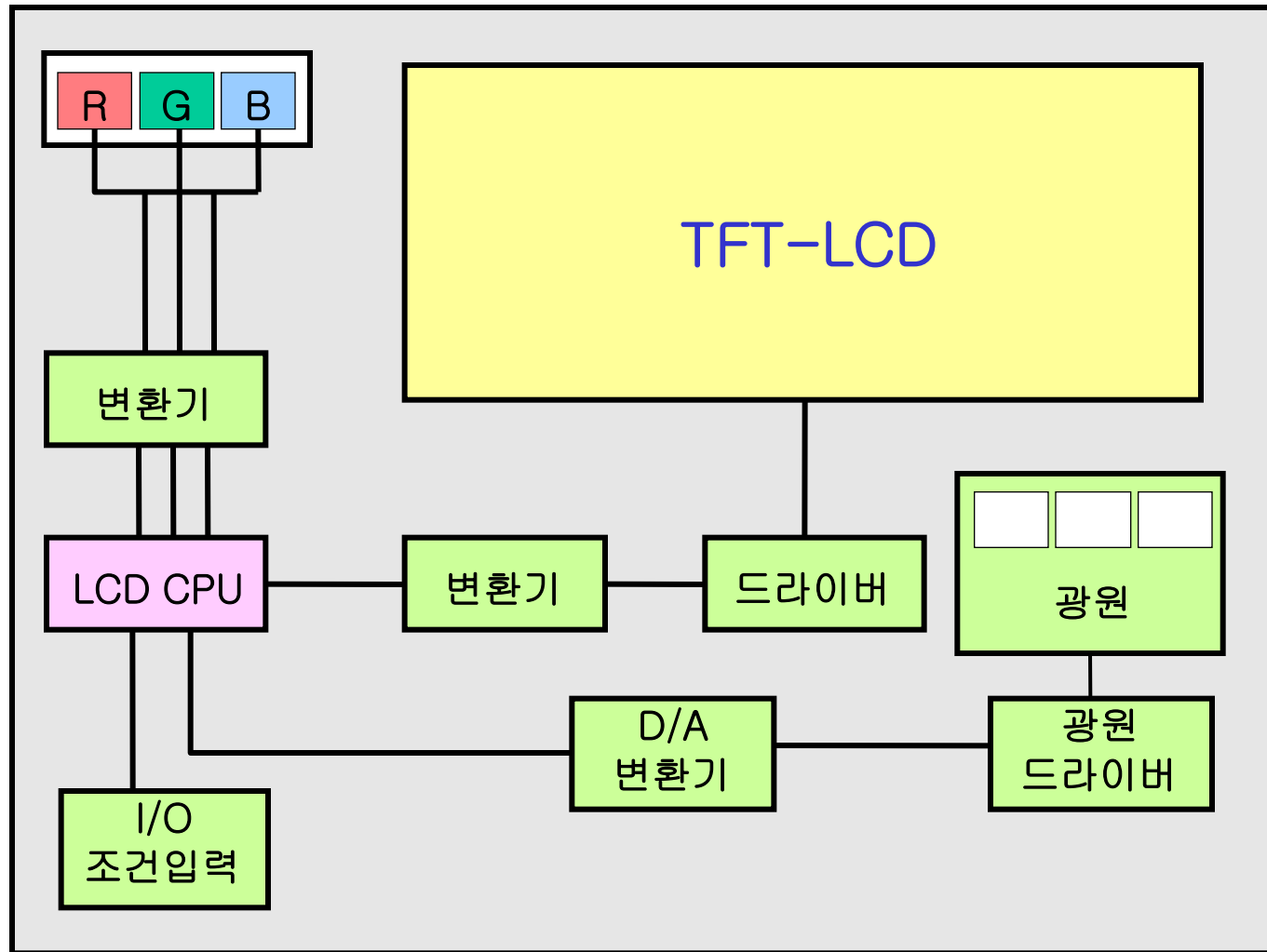


# 4. 마이크로컴퓨터 응용-레이저이용 3차원 물체측정기





## 5. 마이크로컴퓨터 응용-Mobile phone :LCD 모듈



## 6. 마이크로컴퓨터 응용-반도체 제조장비

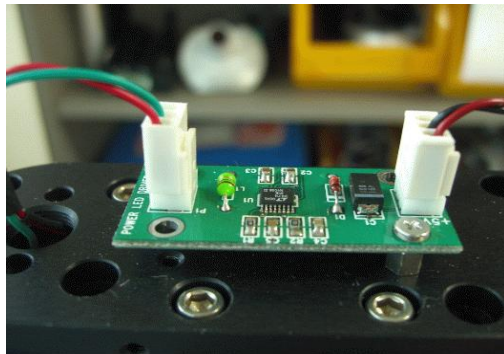
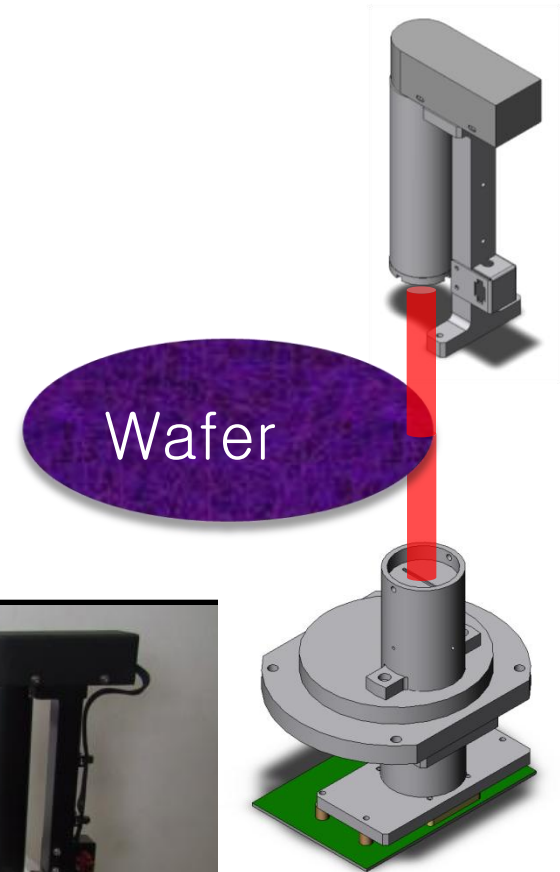
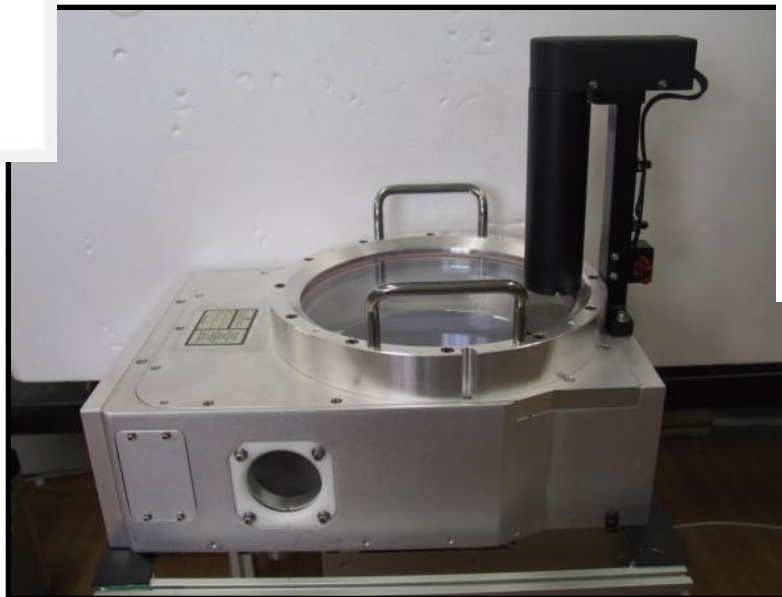
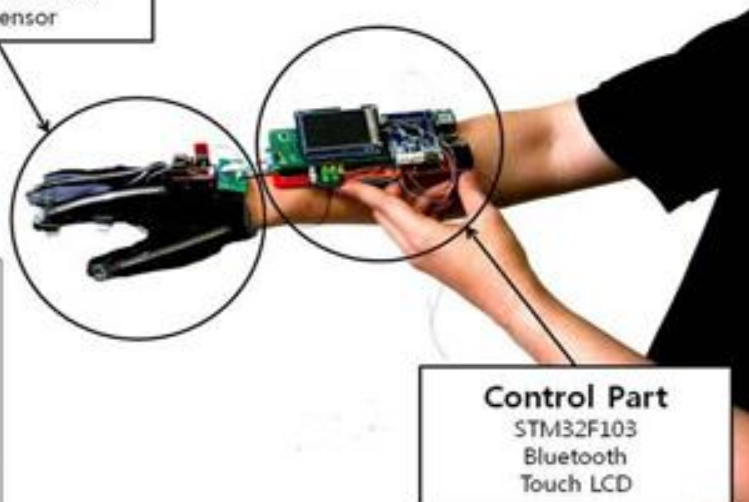


그림 14. Flat Zone Aligner LED Driver



# 7. 마이크로컴퓨터 응용-졸작1:Wearable Input Device

**Sensor Part**  
Zyro Sensor  
Acceleration Sensor  
Flex Sensor



**Control Part**  
STM32F103  
Bluetooth  
Touch LCD

MCU(Micro Controller Unit)  
RC10F450



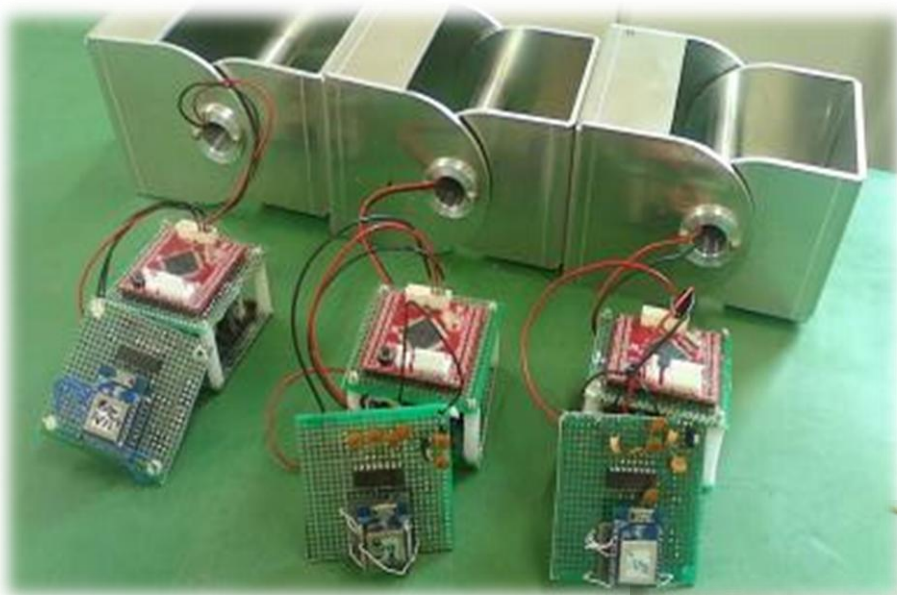
USB Connector

Wireless Communication  
(Bluetooth)

**Transmission and Reception Part**

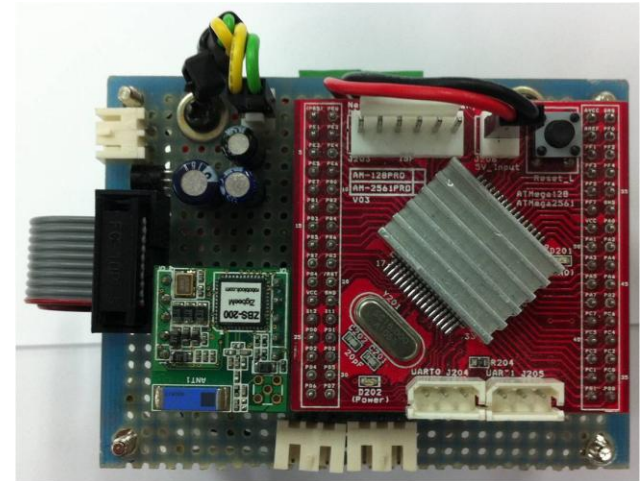
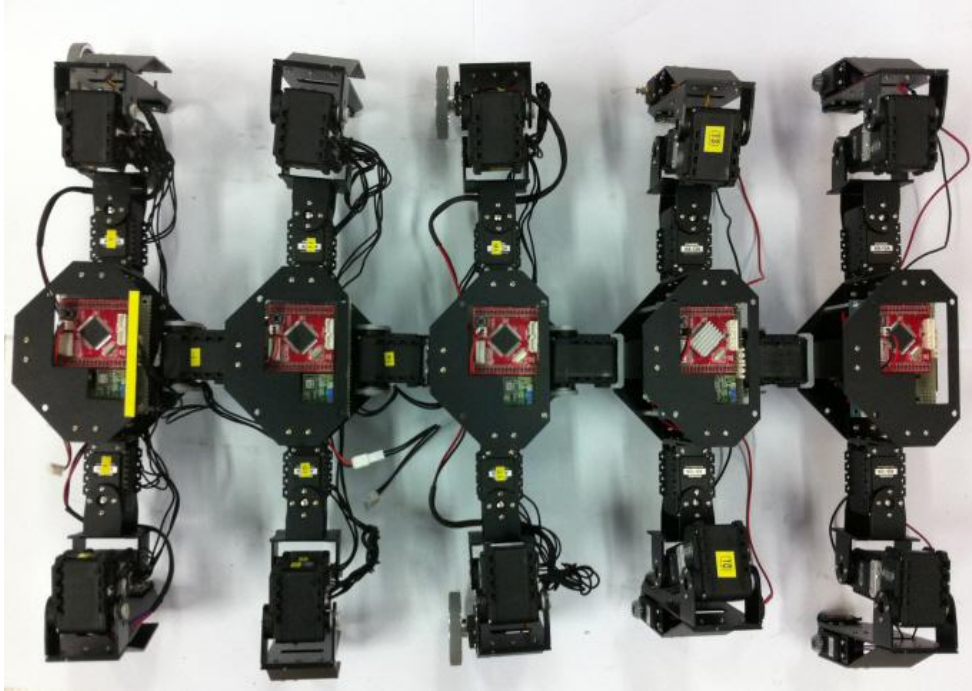


## 7. 마이크로컴퓨터 응용-졸작2:LEBOT(Lego+Robot)

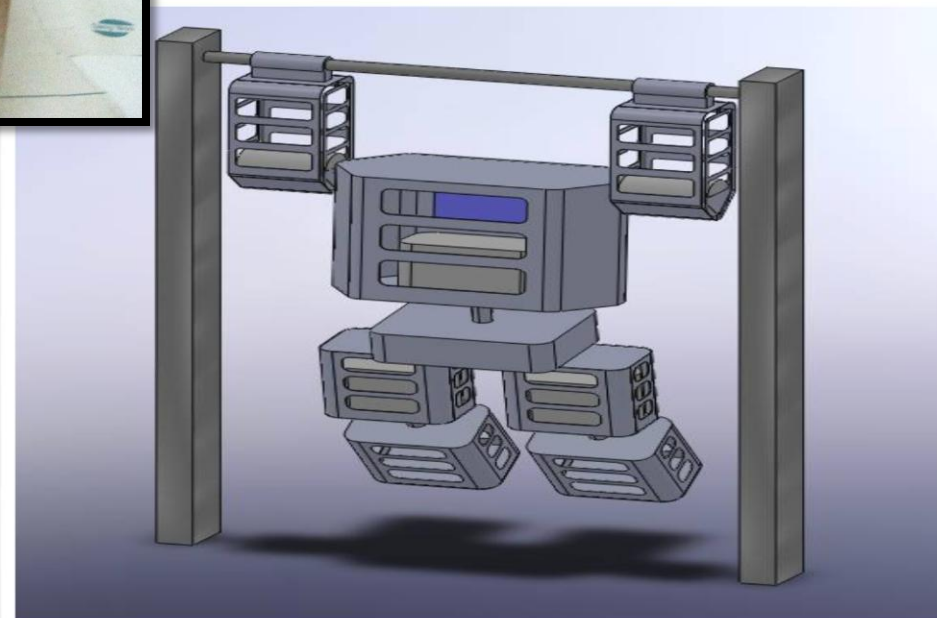




# 7. 마이크로컴퓨터 응용-졸작3:T-BUG(지네형 변신로봇)



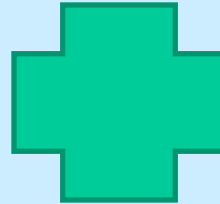
## 7. 마이크로컴퓨터 응용-졸작4: 철봉 로봇



## V. 마이크로컴퓨터 개발환경

# 1. 마이크로컴퓨터 개발

MCU

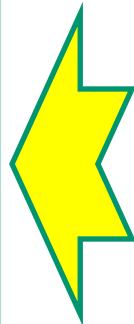


H/W  
(Digital, Analog)



S/W(F/W)

: C언어  
+ MCU정보  
+ H/W정보



개발환경

–H/W: PC, Emulator or ISP  
–S/W: IDE(Integrated Development Environment)  
(Editor/Compiler/Downloader/Debugger)



## 2. 마이크로컴퓨터 개발

개발

Down load

실행



USB

or

Emulator



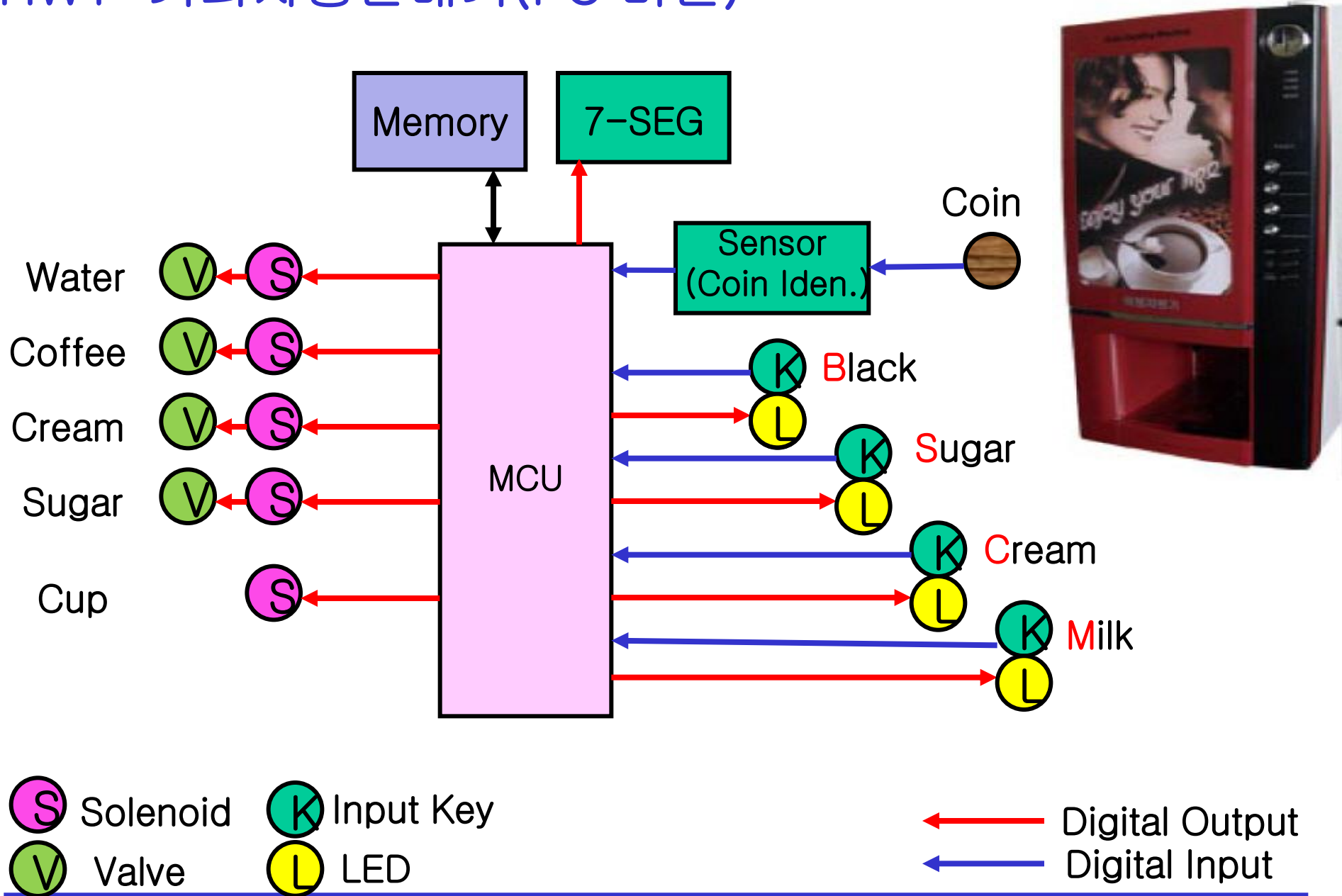
Host System(PC)

:IAR Workbench  
(Cross-Compiler  
/Debugger  
/Project Management

Programmer  
/Downloader

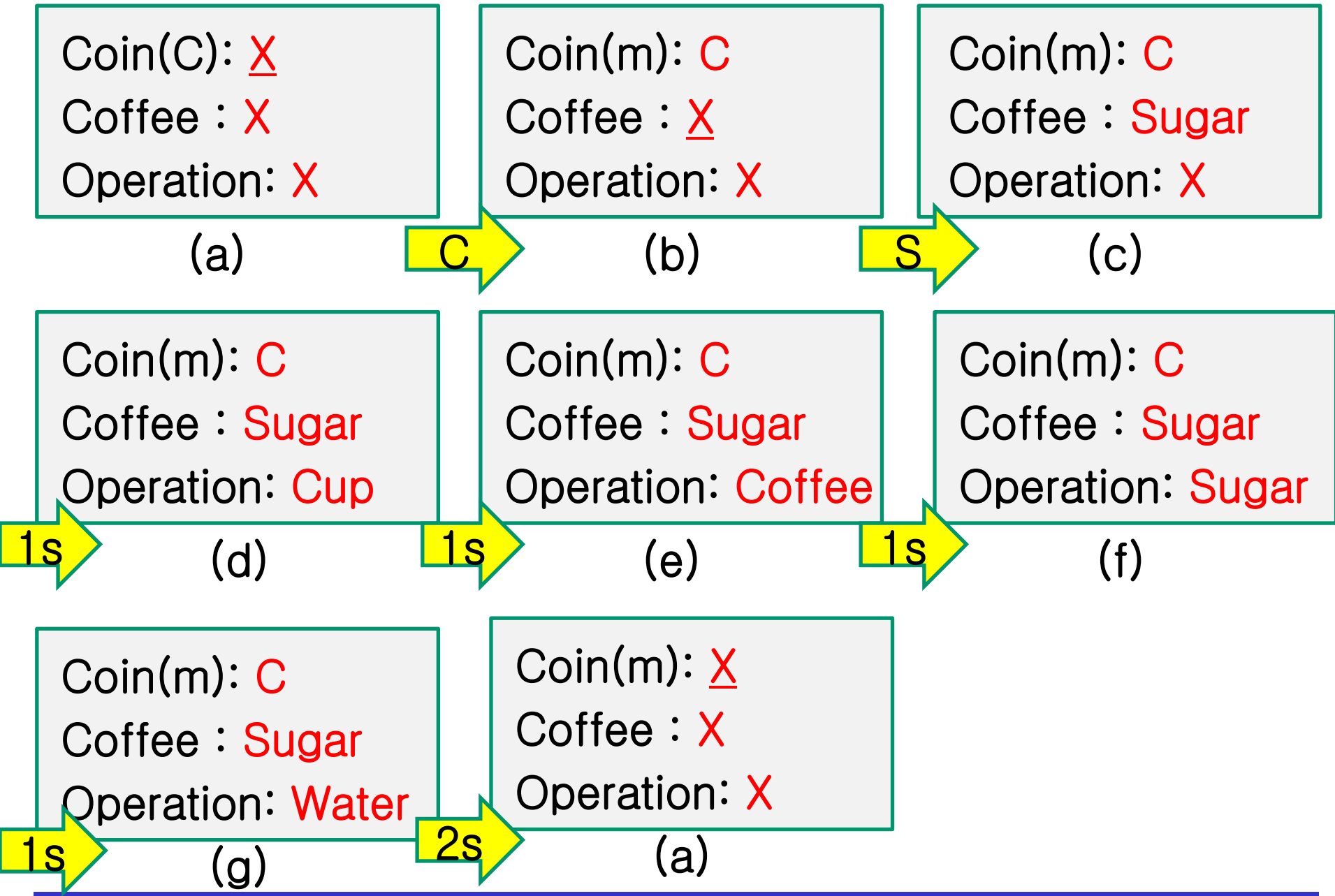
Target System  
(Cortex-M4)

# HW1: 커피자동판매기(PC 버전)



- 모니터에 아래 그림과 같은 화면이 표시되도록 함 (그림 (a))
  - \* X 밑에 언더라인이 있으면 입력을 기다리고 있음을 의미
- Coin 입력: 키 'C or c'를 누르면 X → C로 표시 (그림 (b))  
다음 라인 Coffee 선택메뉴에 X에 언더라인 표시
- Coffee 종류: Black(B), Sugar(S), Cream(C), Milk(M) 중 선택  
그림(c) : Sugar coffee 선택 결과
- 커피에 따라 동작과정을 표시(각 동작사이의 시간간격은 1sec)
- 예: Sugar Coffee: Cup → Sugar → Coffee → Water  
그림 (d) ~ (g)
- 다시 처음으로 (그림 (a))

\* 주의 화면에 표시되는 문자들은 고정위치



## ➤ Homework/Project 제출 표지 및 파일 이름

### ■ On-line 제출시

-파일 이름: 1\_HW1\_2014xxxxx\_HGD.c(txt,hwp)

\* 1:화요일반, 2:수, 3:목

\* HW1: 1<sup>st</sup> Homework

\* 2014xxxxx: 학번

\* HGD: 이름 이니셜(성 먼저)

\* 파일 형식: \*.c 또는 \*.txt( source만 제출시),  
\*.hwp 또는 \*.doc(텍스트 포함한 프로그램 제출시)

표지 (On/OFF-line 제출시)

- On/OFF-line 제출시
  - \* 표지작성하여 제출

과목명: 마이크로컴퓨터 구조  
과제명: C언어로 커피자판기 구현

제출일: 2018.3.13.  
제출자: 메카트로닉스공학과  
화요일반 2014XXXXX  
홍길동