

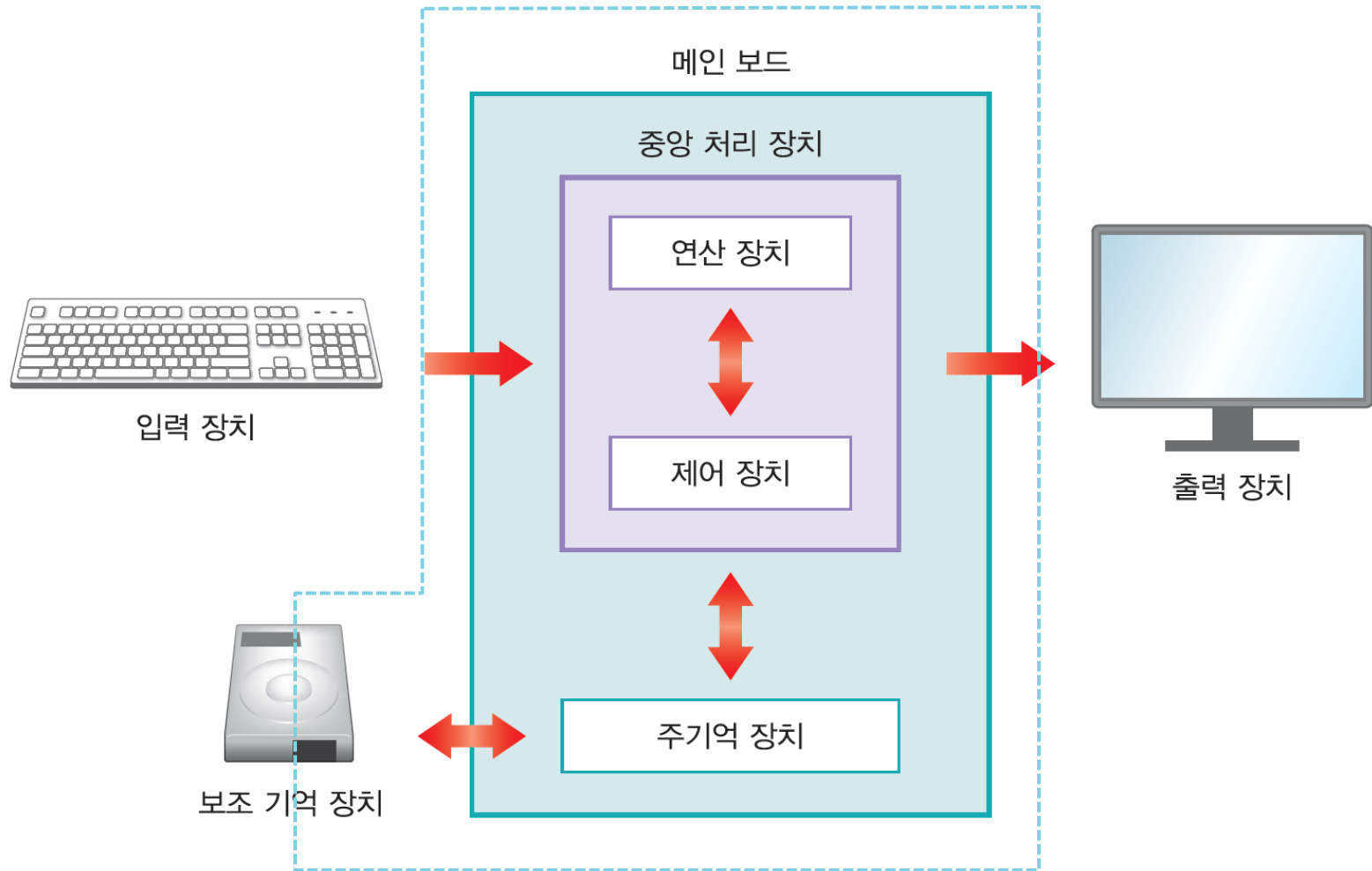
# 마이크로컨트롤러



부산대학교 정보·의생명공학대학  
정보컴퓨터공학부



# 컴퓨터 하드웨어의 구성 요소

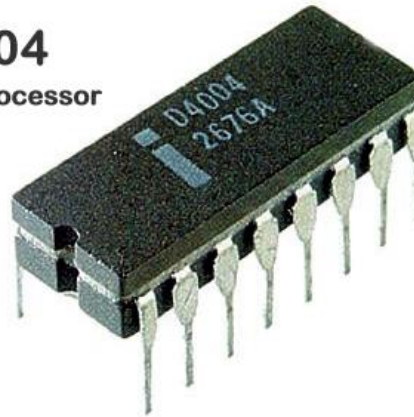


# 마이크로프로세서

## ❖ 마이크로프로세서(microprocessor)

- 컴퓨터의 중앙처리장치(**CPU : Central Processing Unit**)를 단일 IC 칩에 집적해 만든 반도체 소자
- 단순히 프로세서 또는 MPU(Micro Processing Unit)라고도 부름
- 1971년 미국 Intel사, 세계 최초로 마이크로프로세서 개발

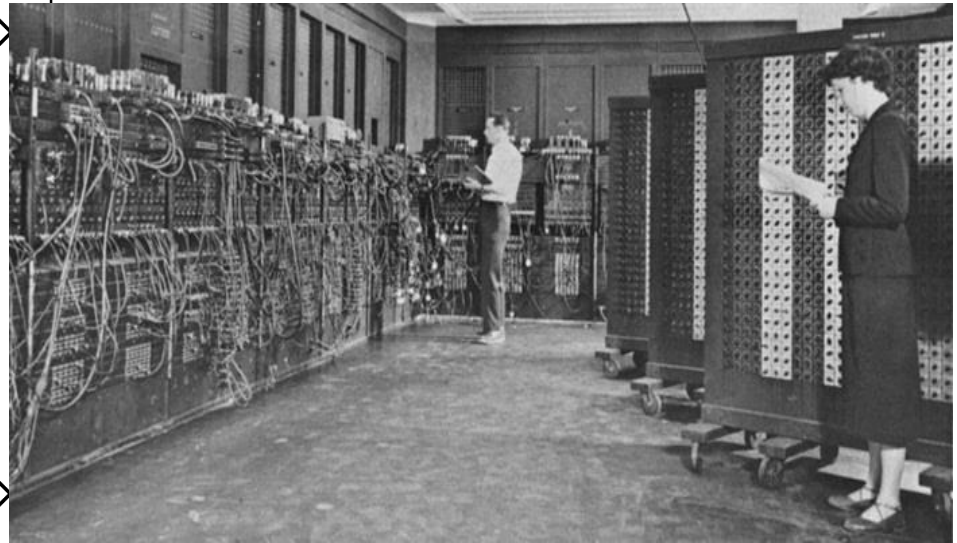
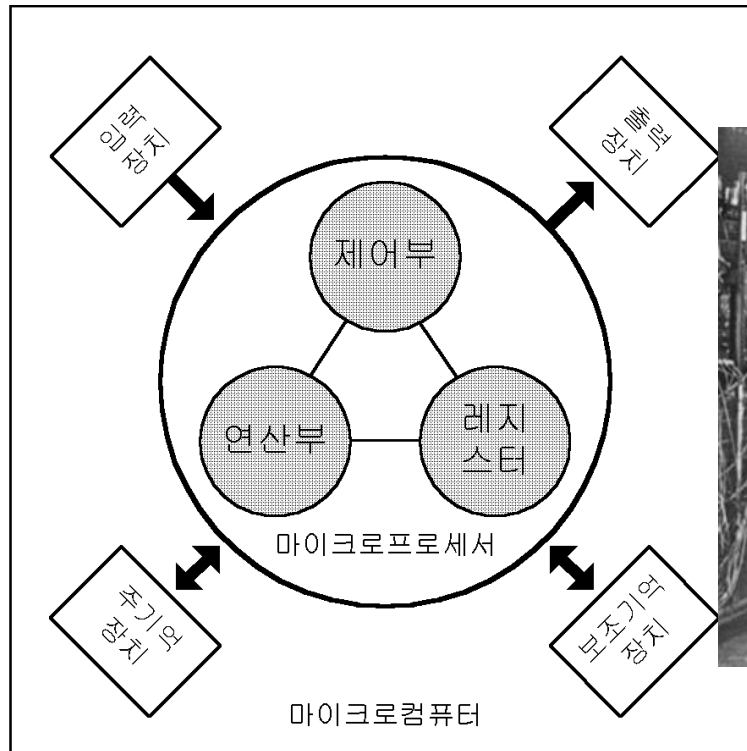
**INTEL 4004**  
The first microprocessor



# 마이크로컴퓨터

## ❖ 마이크로컴퓨터

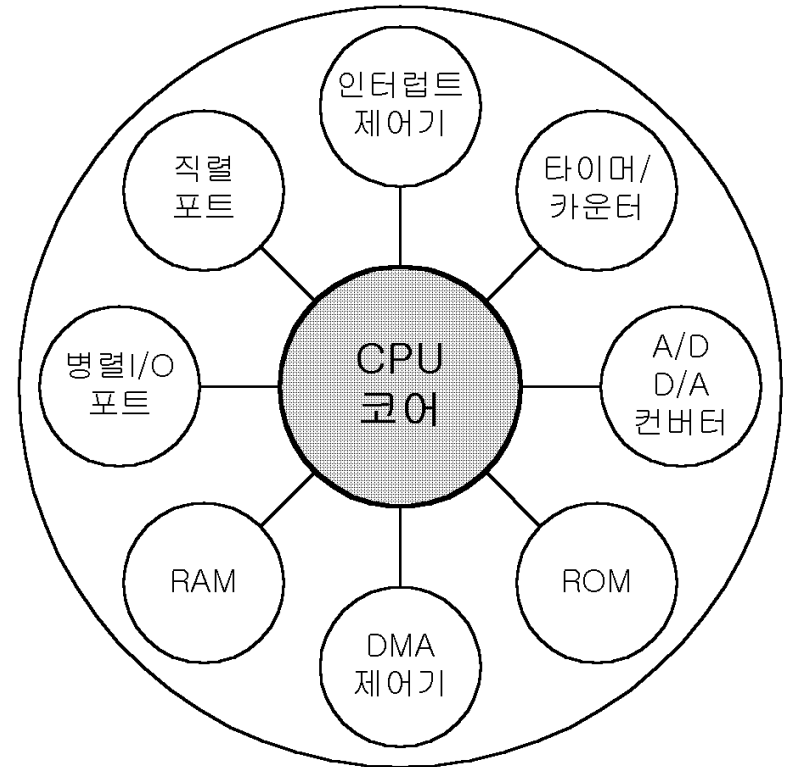
- 마이크로프로세서와 기억장치, 입출력 장치 등을 모아 하나의 시스템으로 구성한 것
- 소형 경량화, 저 소비전력, 저가, 부품 수 감소로 인한 신뢰성 향상



# 마이크로컨트롤러

## ❖ 마이크로컨트롤러(microcontroller)

- 단일 칩 내 마이크로프로세서의 **CPU 코어(core)**와 함께 한정된 용량의 **주기억장치(ROM, RAM)**와 **입출력 인터페이스**를 내장
- 칩 하나만으로 기본적인 컴퓨터의 기능 수행
- 최소 부품으로 간단한 컴퓨터를 구성할 수 있어 시스템 단순화 및 저가격화 가능
- 범용 목적보다 기기제어용으로 주로 사용
- IC 소자 1개로 완전한 형태의 컴퓨터가 구성되므로 단일-칩 마이크로컴퓨터(**single-chip microcomputer**)



# $\mu$ P – Microprocessor, $\mu$ C – Microcontroller

## ❖ 마이크로프로세서 ( $\mu$ P – Micro-Processor)

- 컴퓨터의 중앙 처리 장치(CPU)를 하나의 칩으로 구현한 반도체 소자
- 마이크로프로세서를 중심으로 ROM, RAM, 입출력장치 등을 추가하여 마이크로컴퓨터(microcomputer)를 구성함
  - 이러한 마이크로프로세서를 CPU형 마이크로프로세서라고 함

## ❖ 마이크로컨트롤러 ( $\mu$ C – Micro-Controller)

- 마이크로컴퓨터에 들어가는 모든 부품들을 하나의 반도체 칩에 집적한 것으로 one chip microcomputer 라고도 함
- 작고 가벼운(마이크로, micro) 제어장치(컨트롤러, controller) 구성을 위해 사용되는 특화된 마이크로프로세서의 일종으로 CPU형 마이크로프로세서와 구별
- 연산용과 제어용으로 분류하며, 제어용 단일 칩 마이크로컴퓨터를 마이크로컨트롤러(microcontroller)라고 하고 8051, AVR, PIC 등이 있음

# μP와 μC 사용 시스템 비교

항목	아두이노 메가2560	데스크톱 컴퓨터
CPU	ATmega2560	Intel Core i7
비트	8	64
메모리	256 KByte	8 GByte 32GB
클록	16 MHz (싱글 코어)	3.4 GHz (쿼드 코어)
시스템 가격	약 25,000원 (ATmega2560) US\$ 38.5	약 1,000,000원

# 마이크로컨트롤러의 특징

## ❖ 소형, 경량화가 가능하다

- 다양한 프로그램으로 응용범위와 주변 소자수를 대폭 줄일 수 있어 회로가 간단함

## ❖ 가격이 저렴하다

- 하나의 IC 안에 입/출력포트, 직/병렬통신, 기억소자, 카운터/타이머 등의 기능을 내장시켜 별개의 IC로 시스템을 구현할 때 보다 훨씬 저렴한 시스템의 구축이 가능함

## ❖ 타 시스템과의 이식성(Portability)이 뛰어나다

- 대부분의 제어 기능이 하나의 IC에 내장되어 있으므로, 시스템을 변경하고자 할 경우에는 약간의 기능 변경과 추가를 통해 쉽게 달성할 수 있어 다양한 용도로 활용이 가능함

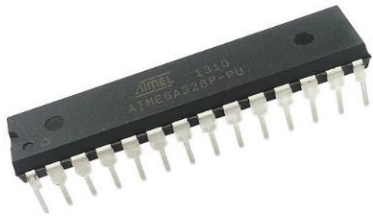
## ❖ 신뢰성이 높다

- 시스템 구성 소자수가 적어 획기적으로 줄어들어 제품의 신뢰성이 높음

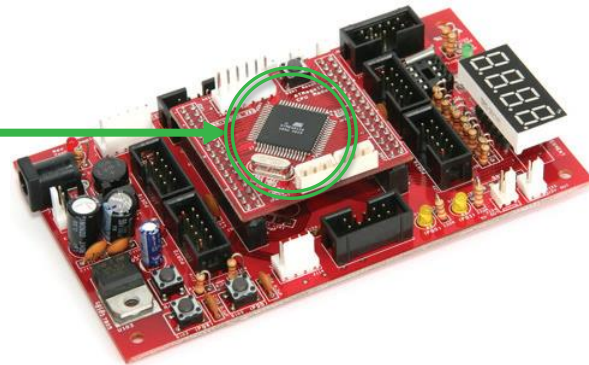
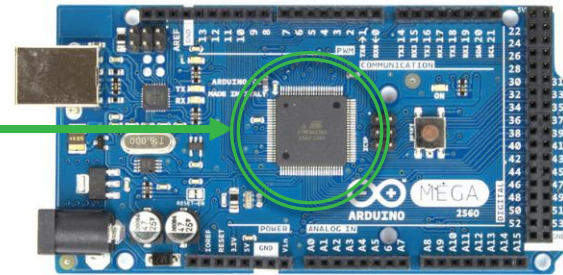


# 마이크로컨트롤러 보드

❖ 마이크로컨트롤러 보드 = 마이크로컨트롤러 +  $\alpha$



마이크로컨트롤러



마이크로컨트롤러 보드

# 마이크로컨트롤러 사용 예

분야	사용 예
의료	의료기 제어, 자동 심박계
교통	신호등 제어, 주차장 관리
감시	출입자/침입자 감시, 산불 감시
가전	에어컨, 세탁기, 전자레인지
음향	CD 플레이어, 전자 타이머
사무	복사기, 무선 전화기
자동차	엔진 제어, 충돌 방지
기타	게임기, 차고 개폐 장치

# μC를 위한 프로그램 개발 과정

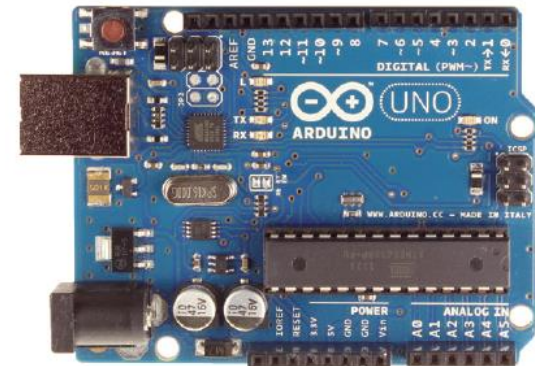
## ❖ 교차 개발 환경

- 프로그램 개발은 **개발 시스템**(데스크톱 컴퓨터)에서 진행
- 개발 시스템은 **목적 시스템**(마이크로컨트롤러)에서 실행되는 기계어 파일을 생성할 수 있는 **교차 컴파일러 (Cross Compiler)** 사용
- 개발된 기계어 파일은 전용 장치를 사용하여 목적 시스템으로 업로드



교차 컴파일러가 설치된  
개발 시스템

개발된 기계어 파일을  
업로드



개발된 프로그램이 실행될  
목적 시스템

# 프로그램 업로드 방식

## ❖ ISP (In System Programming) 방식

- SPI 시리얼 통신 사용
- $\mu$ C의 메모리에 직접 프로그램 기록
- AVR 마이크로컨트롤러에서 주로 사용하는 방식



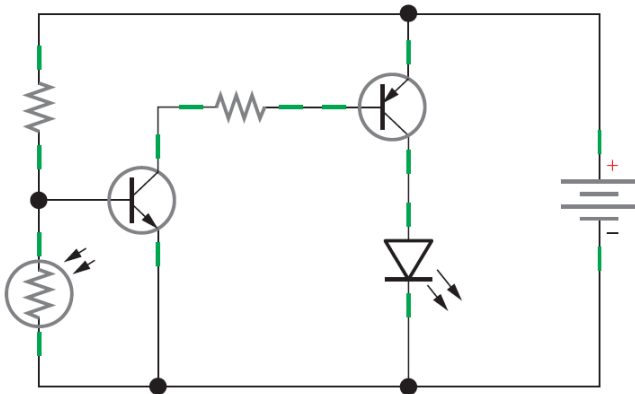
## ❖ 시리얼(Serial) 방식

- UART 시리얼 통신 사용
- 부트로더를 통해  $\mu$ C의 메모리에 프로그램 기록
- 아두이노에서 주로 사용하는 방식



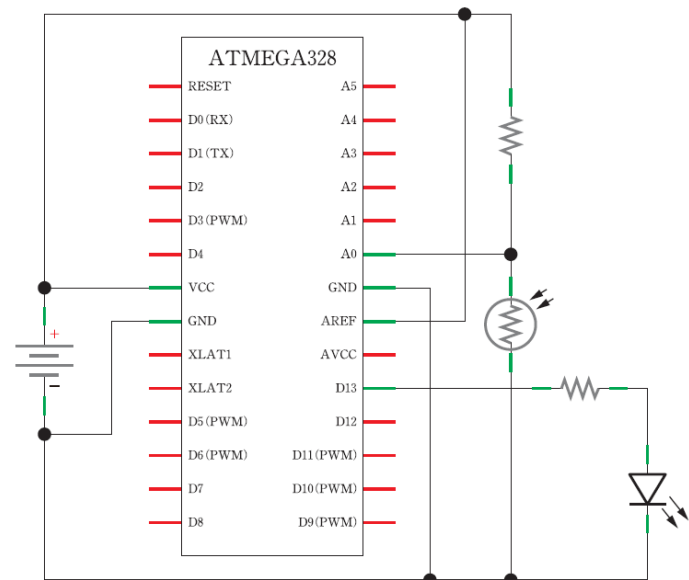
# μC는 정말 필요한가? - 자동 점등 회로

## μC 없이 '하드웨어'로 구현



## μC 사용

### '하드웨어 + 소프트웨어'로 구현



# μC 사용의 장단점

## ❖ 단점

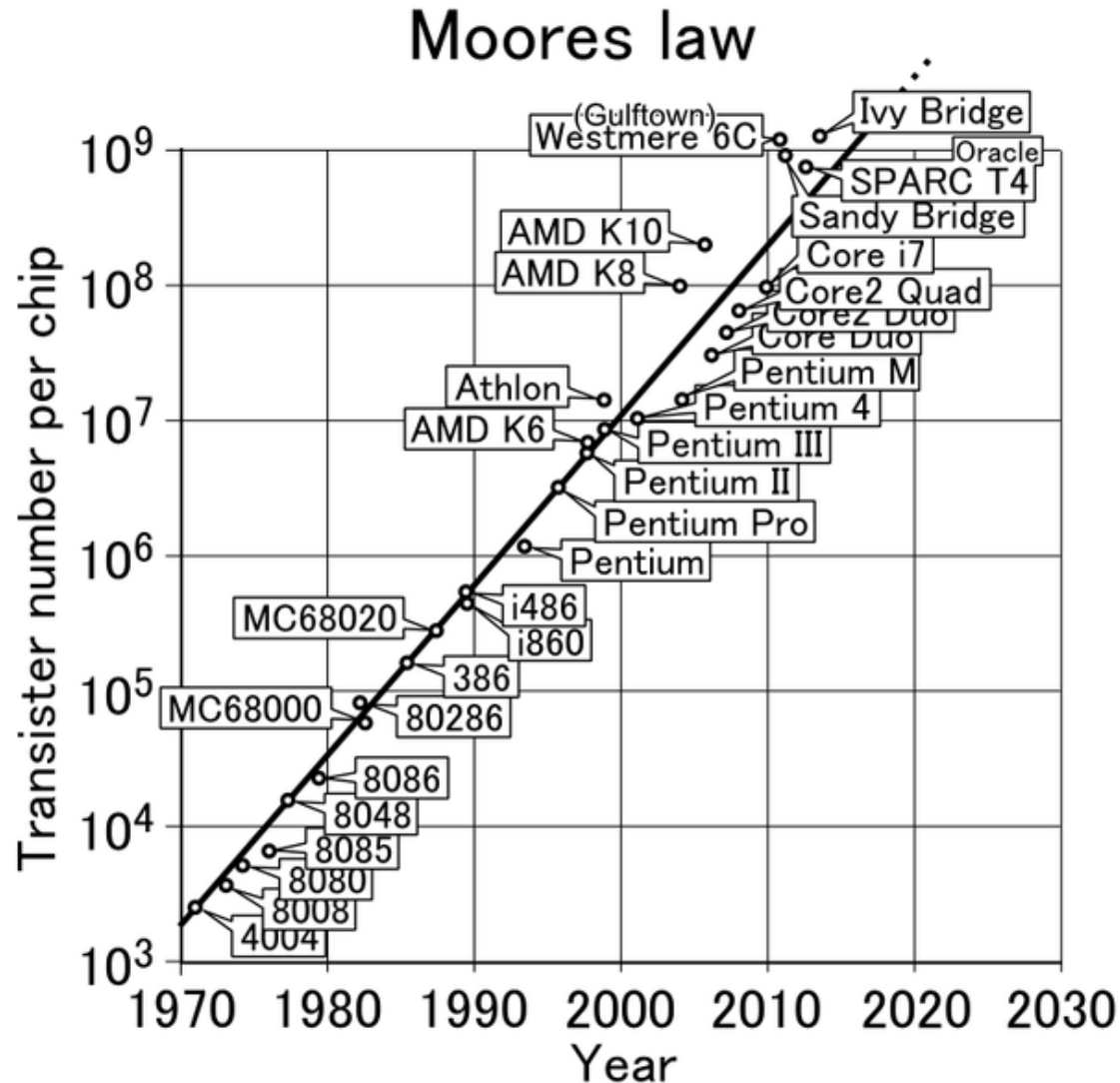
- 점등 회로의 경우 하드웨어로만 구현하는 방식에 비해 장점 없음
- 마이크로컨트롤러 기능을 100% 사용하지 않음
- 별도의 소프트웨어(펌웨어)를 구현해야 함

## ❖ 장점

- 기능을 변경 또는 추가하는 경우, 동일한 입출력을 사용한다면 소프트웨어 변경 만으로 가능
- 무어의 법칙에 따라 마이크로컨트롤러의 성능은 계속 발전하고  
가격은 지속적으로 하락하고 있어 기능을 100% 사용하지 않는 경우에도 소형화, 경량화,  
초기 개발 시간 단축, 유지보수 비용 절감 등의 장점으로 경쟁력 확보 가능

# 무어의 법칙(Moore's Law)

❖ 24개월마다 칩(chip)안의 트랜지스터 개수가 2배가 된다



# Internet of Things

❖ 사물 인터넷(Internet of Things, 약어로 IoT)은 각종 사물에 **센서와 통신** 기능을 내장하여 인터넷에 연결하는 기술을 의미

- 사물이란 가전제품, 모바일 장비, 웨어러블 컴퓨터 등 다양한 임베디드 시스템



## 01 Device connection

IoT devices  
IoT connectivity  
Embedded intelligence

## 02 Data sensing

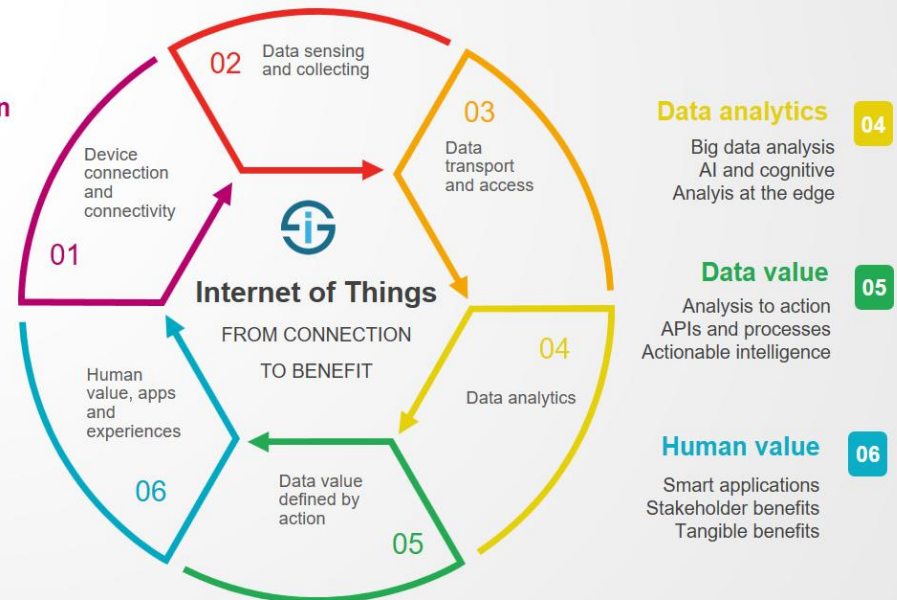
Capture data  
Sensors and tags  
Storage

## 01 Communication

Focus on access  
Networks, cloud, edge  
Data transport

## The Internet of Things

From connecting devices to human value





## ❖ 마이크로컨트롤러는

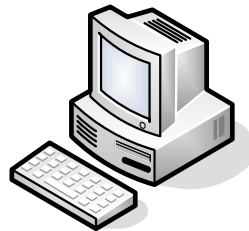
- 작고 단순한 컴퓨터
- 컴퓨터의 본체에 해당하는 기능을 하나의 칩으로 구현 (**single chip computer**)
- 전원만 주어진다면 동작
  - 단, 컴퓨터와 마찬가지로 프로그램이 설치되어 있어야 함
- 입출력 장치를 연결함으로써 완전한 컴퓨터로 동작 가능
- 간단한 제어 장치를 위해 특화된 컴퓨터
  - 낮은 사양으로 데스크톱 컴퓨터와 비교할 수는 없음
- 사물 인터넷을 실현을 위한 핵심 기반 요소

# 컴퓨터 시스템의 이해

컴퓨터 시스템은 사용자 요구에 따른 **임의의 다양한 계산을** 수행할 수 있는 장치로 개발되었다.

## 컴퓨터 시스템

H/W



S/W

- 컴퓨터 H/W로 수행할 수 있는 계산은 **명령어 (Instruction)**라 불리는 **한정된 기초 연산**에 불과하다.
- H/W만으로는 사용자가 요구하는 **임의의 다양한 계산을 수행할 수 없다.**

- **명령어를 조합하여 순차적으로 실행**하면 사용자가 요구하는 임의의 다양한 계산을 실현할 수 있다.
- 컴퓨터 시스템에서 S/W란 특정 계산을 위해 수행하는 명령어 조합과 관련 데이터 모음이다.

컴퓨터 시스템은 하드웨어와 소프트웨어로 이루어진다

## ❖ 컴퓨터 프로그램(Program)이란?

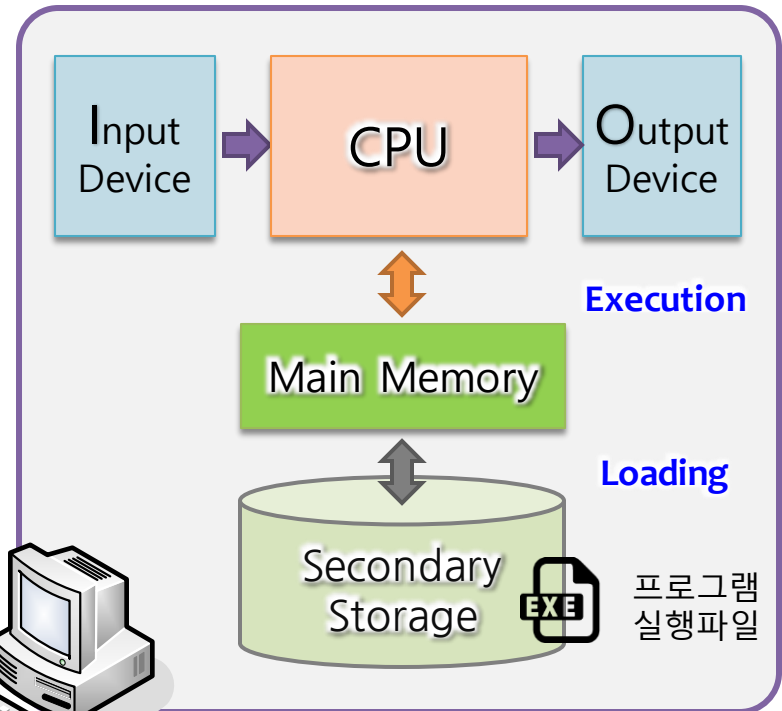
- 컴퓨터가 수행할 명령어들의 순차 조합 (Sequence of Instruction)과 관련 데이터
- 명령어는 기계가 직접 해독하고 실행할 수 있는 이진 값으로 표현되는 기계어 (Machine Code)
- 실행 파일(Executable File)의 형태로 보조기억장치에 저장/설치됨

## ❖ 프로그램의 실행 과정

- 보조기억장치에 저장된 프로그램 실행 파일을 Main Memory로 적재(Load)하고
- CPU는 Main Memory로부터 프로그램의 명령어들을 순차적으로 하나씩 읽어 들여 해석하고 수행하는 방식으로 프로그램을 실행(Execution)
- 참고) 프로그램 적재/실행을 수행하는 SW → 운영체제 (OS : Operating System)

### 컴퓨터 H/W 구성 요소

- 중앙처리장치 - CPU (Central Processing Unit)  
명령어를 해석하여 실행하는 핵심 구성 요소
- 주기억장치 - Main Memory  
프로그램과 데이터를 저장하는 휘발성 기억장치  
(참고) 전원이 꺼져도 저장 정보를 유지하는 기억 장치를 비휘발성 (Non-Volatile), 그렇지 못한 장치를 휘발성(Volatile) 기억장치라고 함
- 보조기억장치 - Secondary Storage  
프로그램과 데이터를 파일의 형태로 저장하는 비휘발성(Non Volatile) 기억장치로 HDD, SSD 등이 있음
- 입출력 장치 - Input / Output Device  
인간과 컴퓨터의 상호 작용을 도와주는 키보드, 마우스, 모니터와 같은 장치



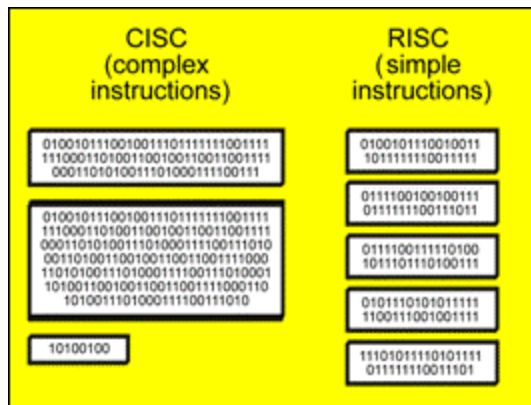
# 명령어 집합에 의한 CPU 분류 (CISC vs. RISC)

## ❖ CISC (Complex Instructions Set Computers)

- CISC 구조의 대표 CPU/ISA로 Intel의 x86 Architecture – PC/Notebook 등에 널리 쓰임

## ❖ RISC (Reduced Instructions Set Computers)

- RISC 구조의 대표 CPU/ISA로 ARM의 ARM Architecture – 다양한 Embedded 장치 및 모바일 기기에 널리 쓰임



CISC	RISC
1) CISC architecture gives more importance to hardware	1) RISC architecture gives more importance to Software
2) Complex instructions.	2) Reduced instructions.
3) It access memory directly	3) It requires registers.
4) Coding in CISC processor is simple.	4) Coding in RISC processor requires more number of lines.
5) As it consists of complex instructions, it take multiple cycles to execute.	5) It consists of simple instructions that take single cycle to execute.
6) Complexity lies in microporgram	6) Complexity lies in compiler.

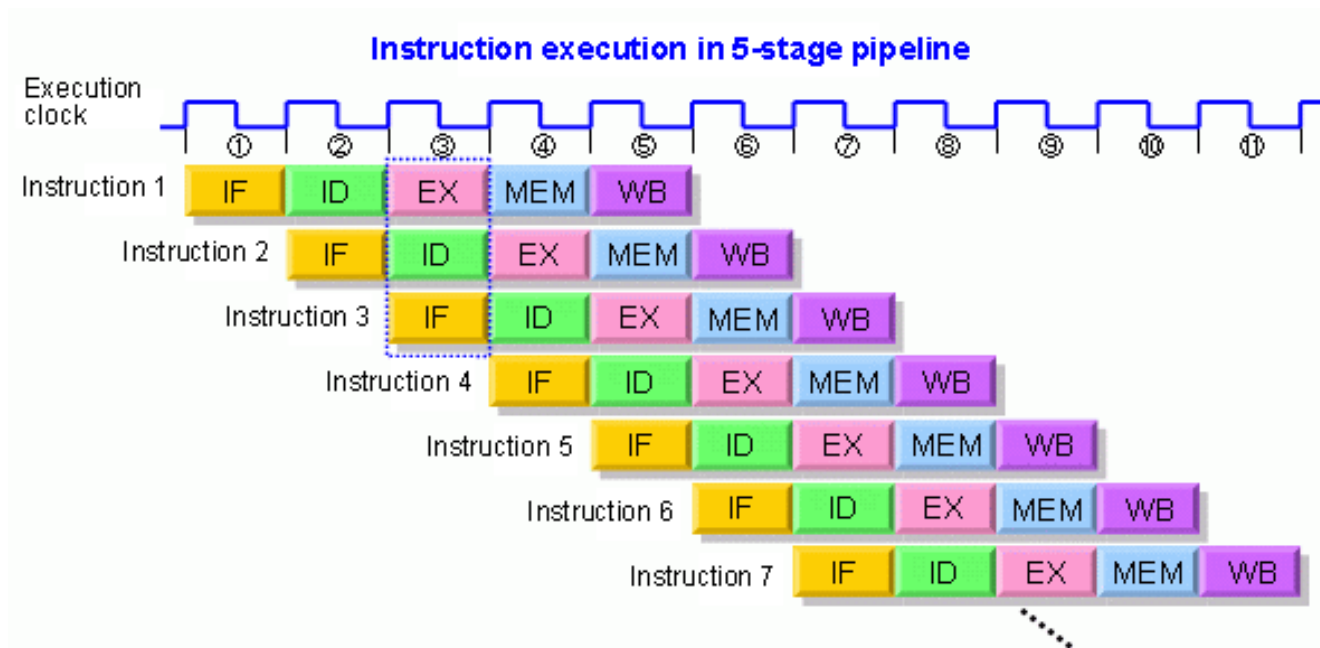
# RISC and Instruction pipeline

## ❖ CPU의 Instruction 처리 과정

- IF (Instruction Fetch)
- ID (Instruction Decode)
- EX (EXECUTE)
- MEM(MEMORY ACCESS)
- WB (WRITE BACK)

## ❖ Instruction Pipeline

- 한 번에 하나의 명령어만 실행하는 것이 아니라 하나의 명령어가 실행되는 도중에 다른 명령어 실행을 시작하는 식으로 동시에 여러 개의 명령어를 실행하는 기법

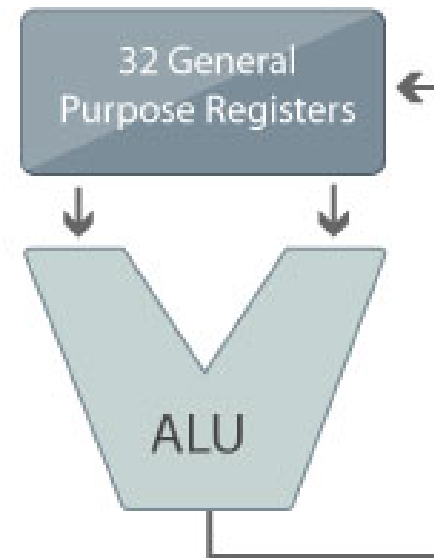


# Arduino Mega2560의 마이크로컨트롤러는?



- ❖ AVR2560 microcontroller
- ❖ AVR stands for **A**lf (Egil Bogen) and **V**egard (Wollan)'s **R**ISC processor
- ❖ **RISC architecture with CISC instruction set**
  - Easy to learn and powerful instruction set for C and Assembly
- ❖ **Single cycle execution**
  - one instruction per external clock
  - Low power consumption
- ❖ **32 working Register**
  - All register are directly connected to ALU
- ❖ **Very efficient core**
  - New design using new technology
  - Fully scalable for future products

Vegard Wollan      Alf-Egil Bogen

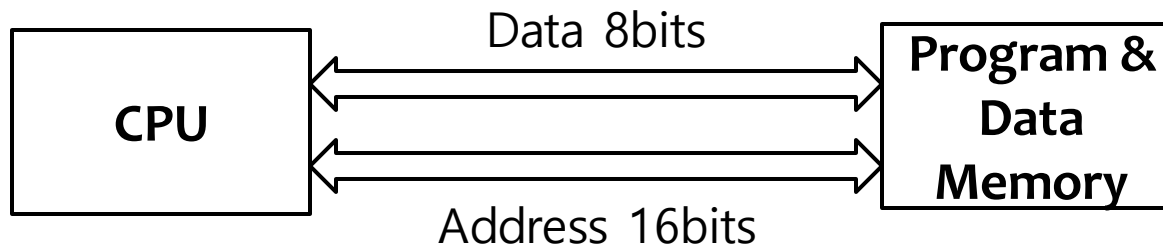


# 메모리 구조에 의한 CPU분류



## ❖ Von Neumann 구조

- 프로그램과 데이터를 하나의 메모리에 저장하여 데이터는 메모리에서 읽거나 메모리에 쓰기도 하는 반면, 명령어는 메모리에서 읽기만 하는 구조.
- 인텔의 x86 계열의 CPU 와 AMD의 인텔 호환 CPU 등이 해당됨.

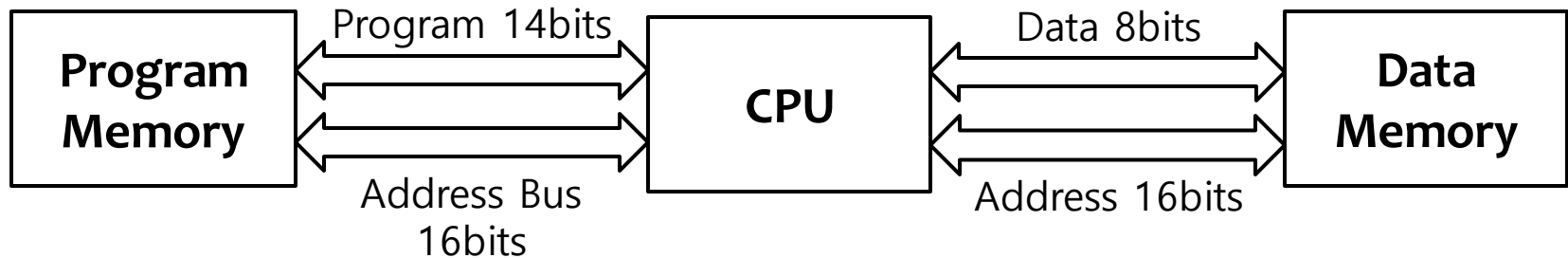


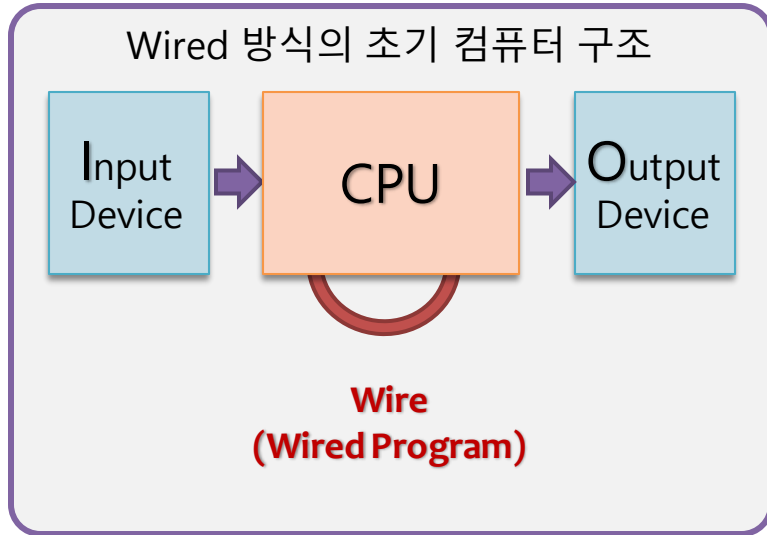


# 메모리 구조에 의한 CPU분류

## ❖ Harvard 구조

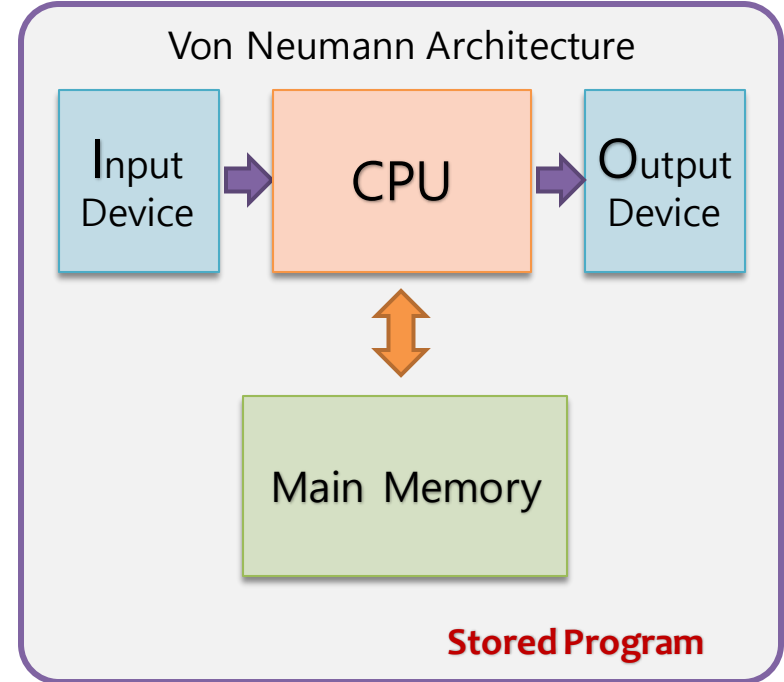
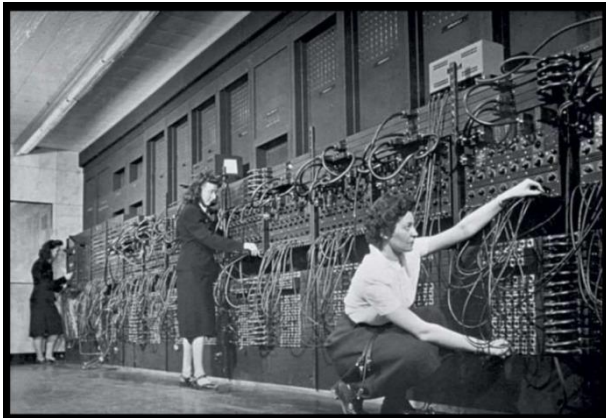
- 프로그램과 데이터를 물리적으로 구분하여 각각 다른 메모리에 저장하는 구조
- 명령 구조상으로 RISC(Reduced Instruction Set Computer)구조임
- 애플의 PowerPC 등이 해당됨.





1943: ENIAC

- 최초의 범용 전자 컴퓨터
- **Hard Wired Program** - 다이얼과 스위치 조작



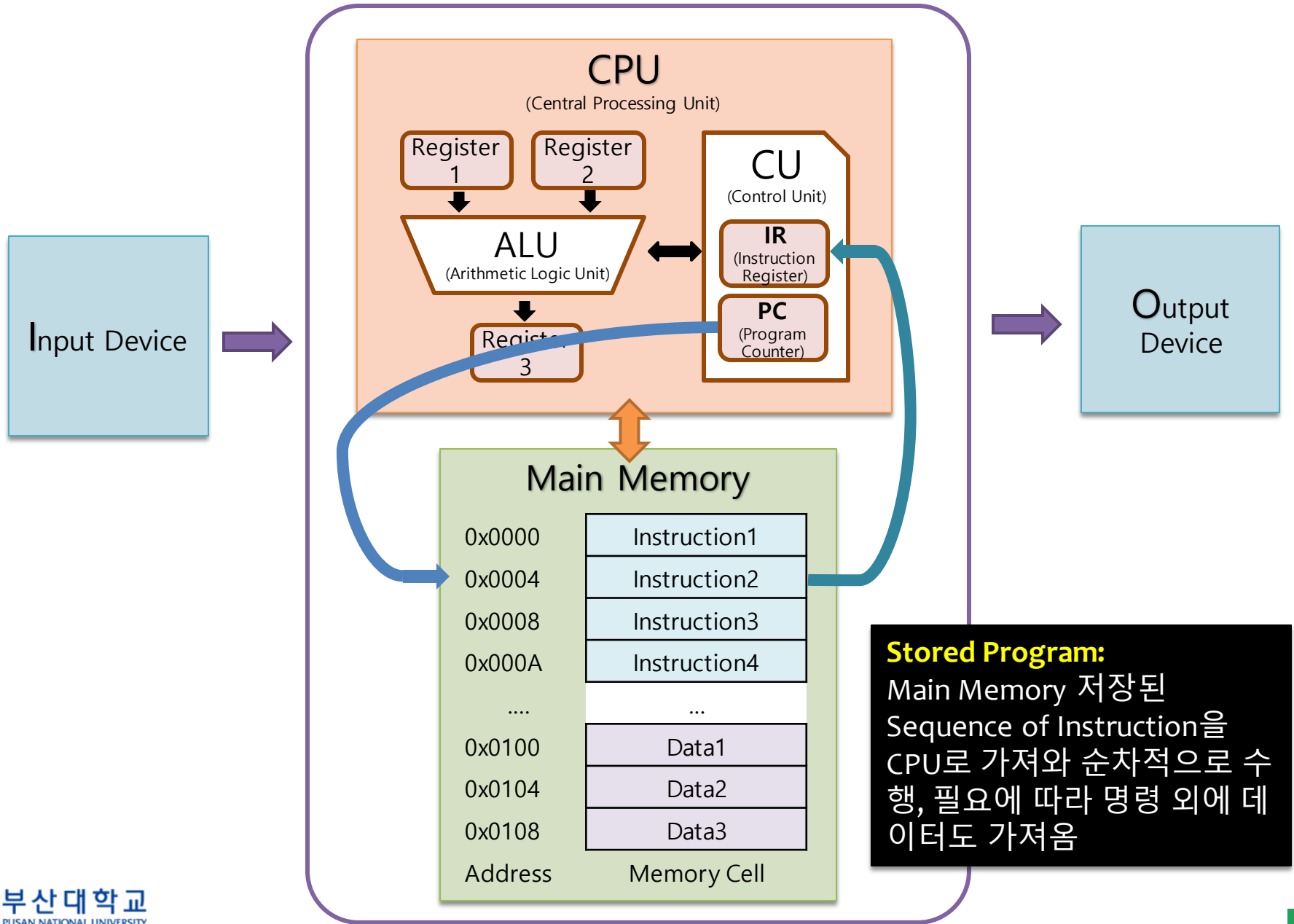
1944: Beginnings of EDVAC

- **프로그램을 메모리에 저장**할 수 있는 기능 개선 이루어짐

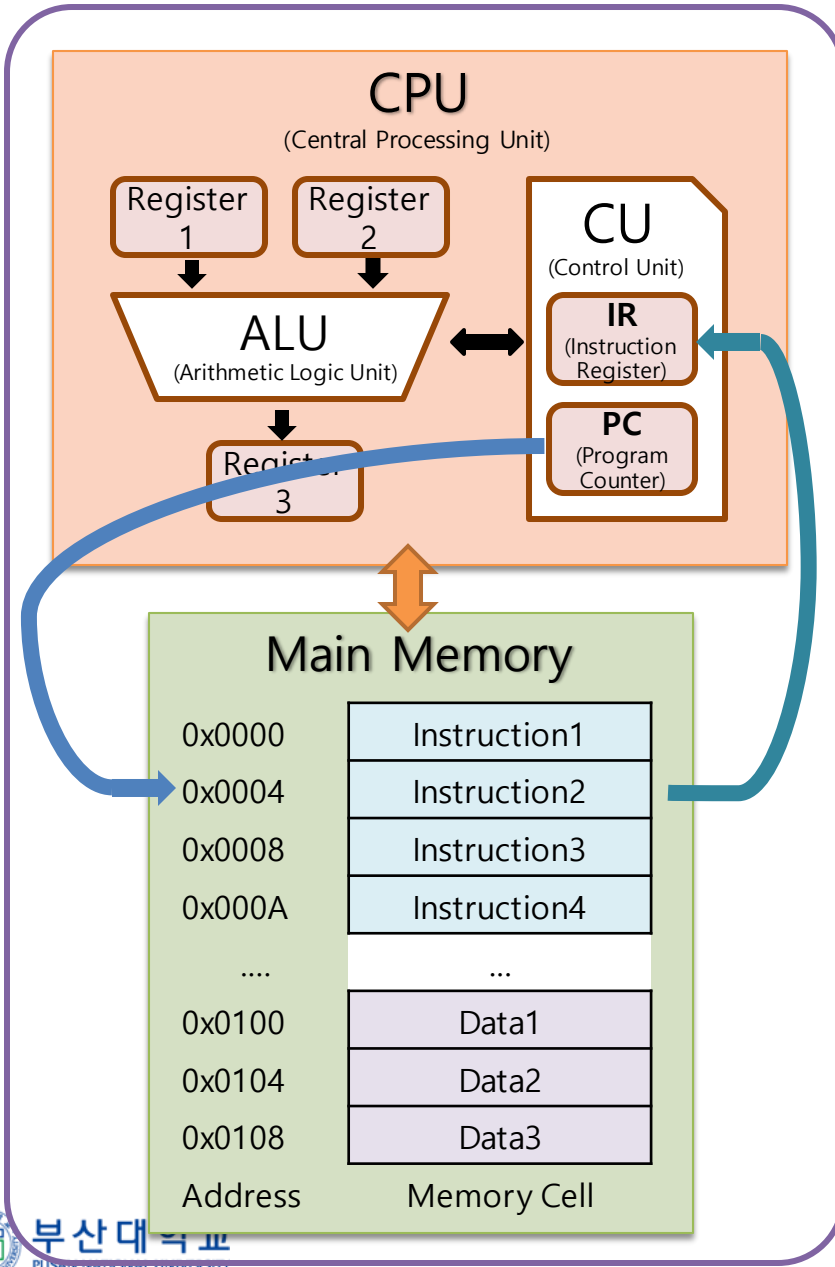
1945: John von Neumann

- “The First Draft of a Report on EDVAC”이라는 문서를 통해 **“Stored Program”** 개념 발표
- Draft에서 설명한 기본 구조 → Von Neumann Machine (or Model)로 불림

# Von Neumann/Stored Program Architecture



# CPU: Central Processing Unit



❖ 컴퓨터의 두뇌에 해당하는 핵심부분으로 명령어의 인식, 해독과 실행을 제어하며 연산처리를 수행하는 장치

❖ CPU : ALU + CU + Registers

❖ ALU (Arithmetic Logic Unit)

- 산술논리장치 : 연산을 수행하는 Unit

❖ Register

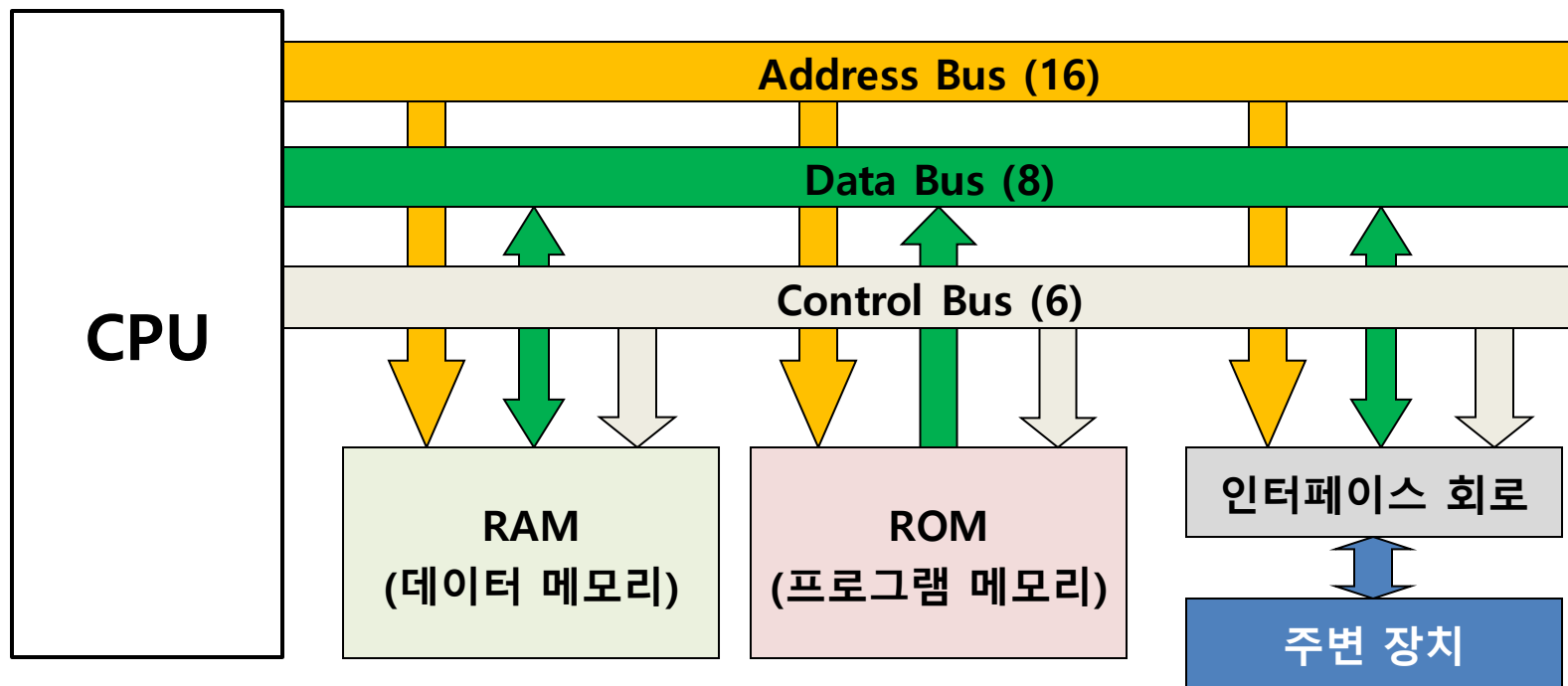
- ALU의 연산 작업 등에 쓰이는 CPU 내의 적은 용량의 고속 Memory

❖ CU (Control Unit)

- Instruction을 Main Memory에서 읽어와 해석하여 ALU와 Register를 활용하여 명령을 수행하는 과정을 제어하는 Unit
- 현재 읽어 들여 수행할 명령이 저장된 메모리 주소 값을 가지는 **PC (Program Counter)**라고 불리는 Register가 존재함
  - PC 값을 이용해 수행할 명령을 읽어 온다
  - Instruction을 수행하면 일반적으로  $PC = PC + 1$ 의 값을 가짐, 즉 다음 Instruction이 저장된 주소를 가짐
- 읽어 드린 명령을 저장하는 **IR (Instruction Register)**라고 불리는 Register가 존재

## ❖ BUS in Computing

- a **Communication system** that transfers data between components inside a computer



✚ 내부 버스 : 주소버스, 데이터 버스, 제어버스

✚ 메모리 : ROM과 RAM

✚ 주변장치 : 직/병렬 입출력 장치, 타이머/카운터, 인터럽트, ADC 및 DAC

# CPU의 동작

