

임베디드컴퓨팅

Embedded Computing
(0009488)

Arduino Overview

2022년 2학기

정보기술대학 정보통신공학과

김 영 필

ypkim@inu.ac.kr

Recap: Embedded system

- Embedded?

- **Including** something with anything for a reason, or **integrated or attached** with another thing.

- Embedded Systems?

- An integrated system which is formed as an combination of computer hardware and software **for a specific function.**
- A **dedicated computer system** which has been developed for some **particular reason.**
- Can **work independently or attached to a larger system** to work on few specific functions.
- Can work **without** human intervention or **with** a little **human intervention**

Recap: General characteristics of an embedded system

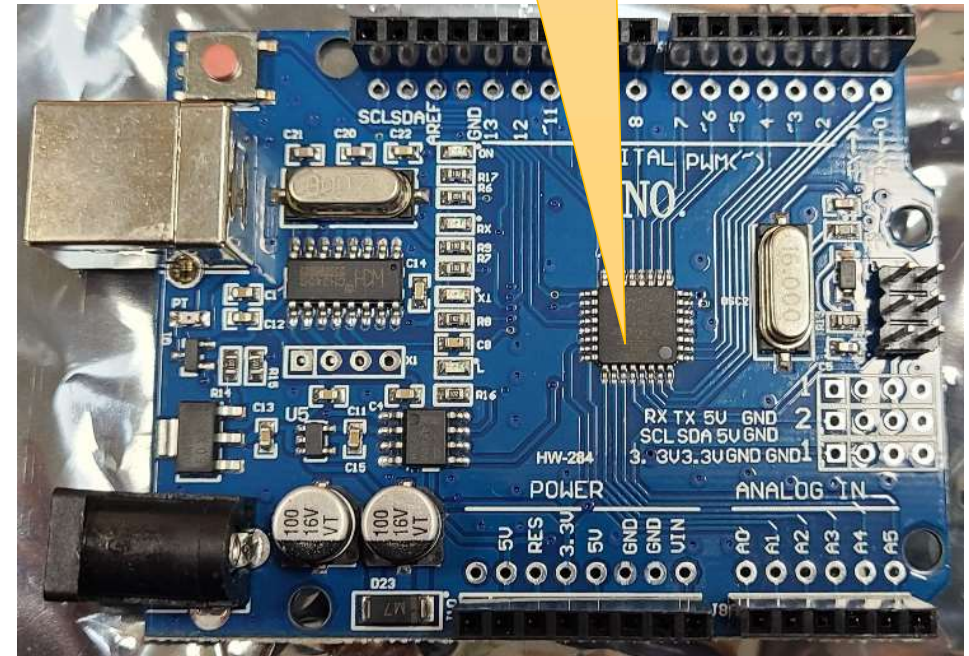
- Performs specific task
- Low Cost
- Time Specific
- Low Power
- High Efficiency
- Minimal User interface
- Less Human intervention
- Highly Stable
- High Reliability
- Use microprocessors or micro controllers

Contents

- Microcontroller
- Arduino Board
- Arduino Interfaces
- Arduino IDE

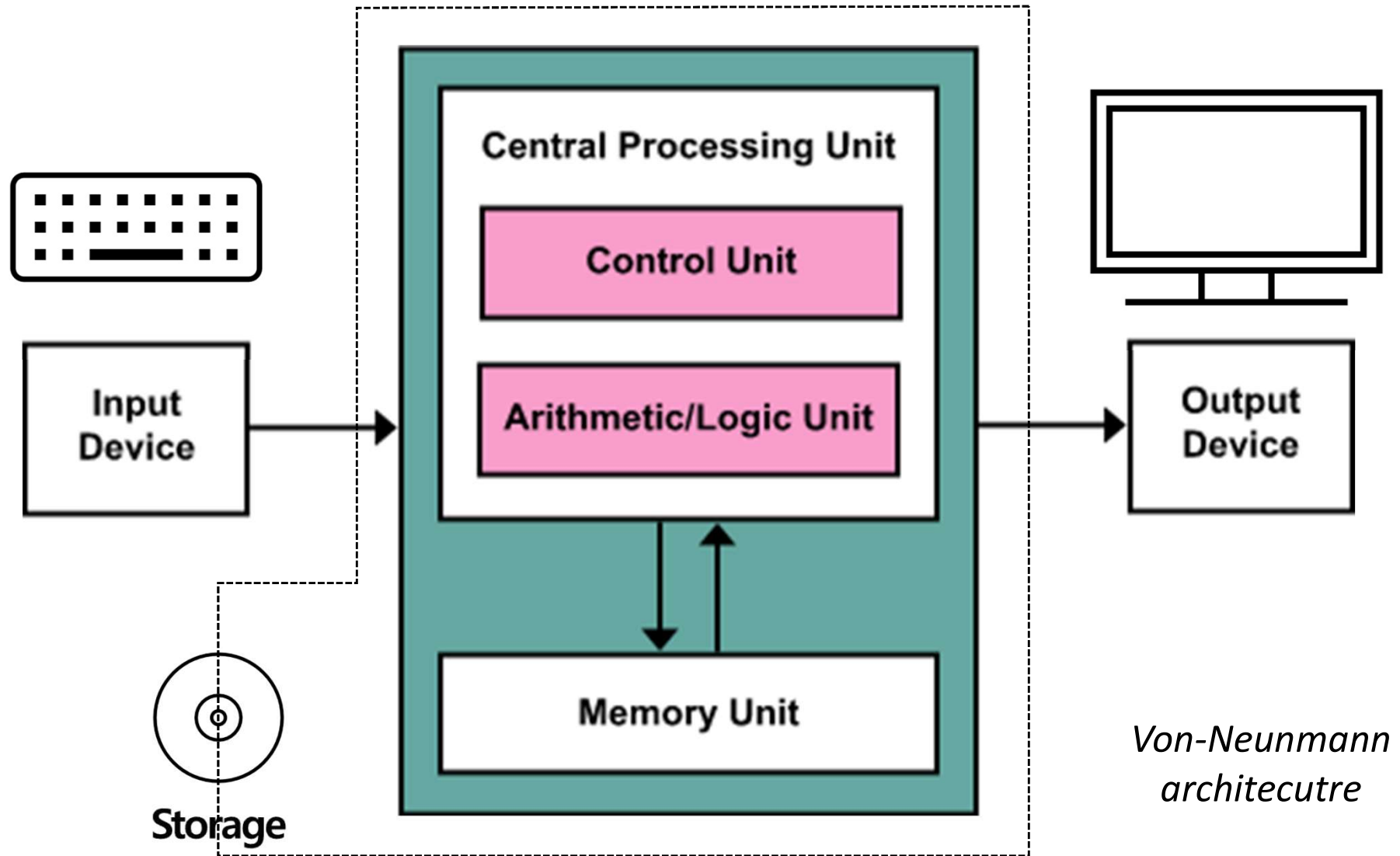
Microcontroller (uC)

- One-chip based **small computer**
 - Microcontroller Unit (MCU)
- CPU + Memory + HDD + I/O interface + α
- Basically the same structure as the main body of a desktop computer and performs the same operation
- As long as power is supplied, it can operate as a computer.
- By connecting an input device and an output device, it works like a computer.



Microcontroller board
(Arduino Uno R3)

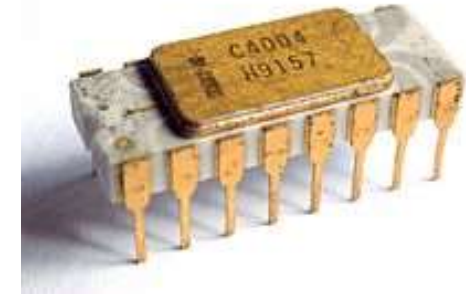
Basic computer structure



Microcontroller = Mainboard + A part of storage

Microprocessor (uP)

- A semiconductor device that implements the central processing unit (CPU) of a computer in a single chip
- First microprocessor
 - Intel's 4004
 - 4-bit microprocessor (1969).



Ref - Intel 4004:

https://en.wikipedia.org/wiki/Intel_4004

Performance

Max. CPU clock rate: 740-750 kHz
Data width: 4 bits

Item	Arduino Uno R3	Desktop PC (i7)
CPU	ATmega328p	Intel Core i7-11 th Gen.
Bit	8	64
Main Memory size	2 KByte (SRAM)	16 GByte (DDR4 DRAM)
Clock speed	16 MHz (single core)	2.5 ~ 4.9 GHz (quad core)
Total Cost	30,000 ~ 40,000 Won	900,000 ~ 1,000,000 Won

Key differences: uC vs. uP

- **Cost**

- Generally, $uC < uP$
- uP are typically manufactured for use with more expensive devices, more complex and computational tasks
- uC usually perform a dedicated function intended for the specific application
- uC internally houses all of the necessary computing features and components

- **Speed**

- uC needs just the right amount of speed and power to get the job done – no more and no less.
- clocking speeds of uP are up to 4 GHz, while uC can operate with 200 MHz or less

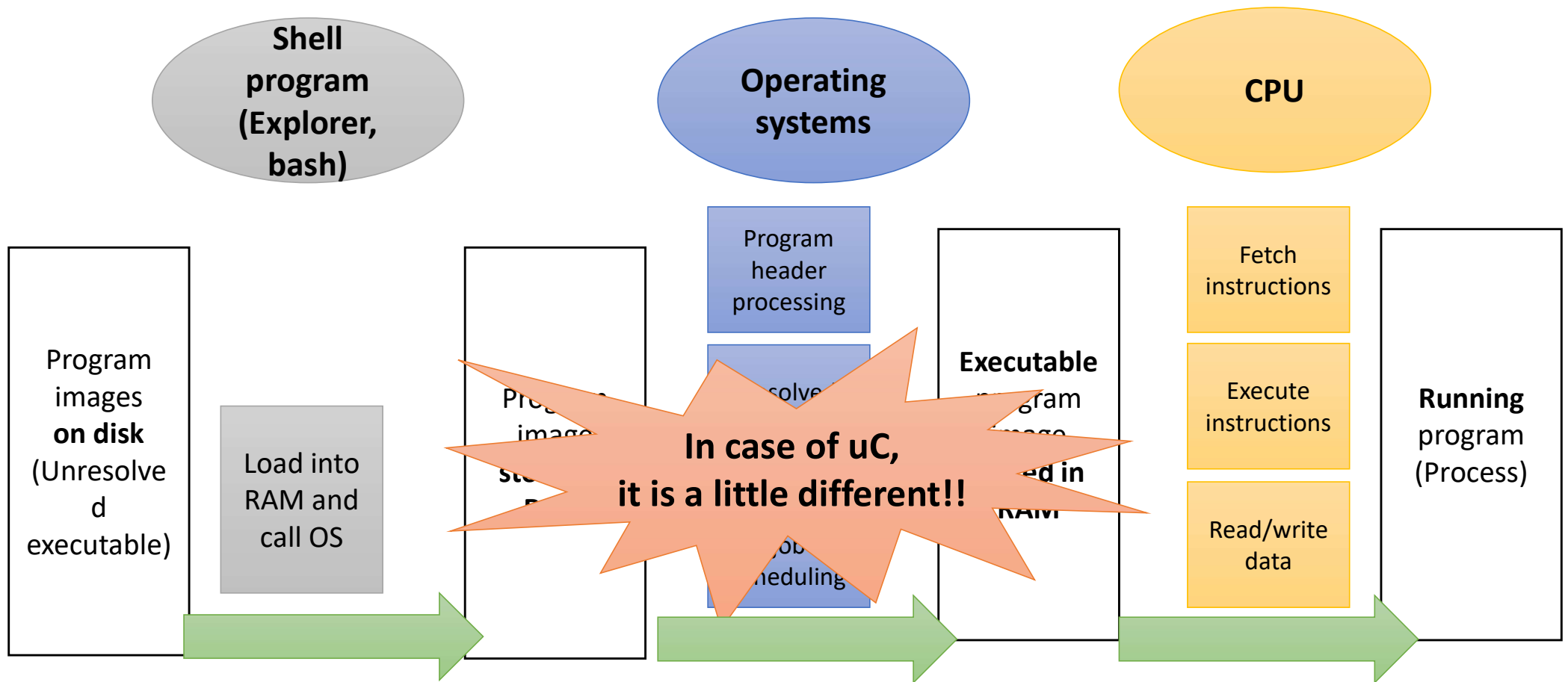
- **Power Consumption**

- uP consumes a lot of power and need external power supply
- uC consumes limited power and could be powered for a long time by just a small battery

Microcontroller Usages

Area	Usage
Medical	Medical device controller, automatic heart rate monitor
Traffic control	Traffic controller, parking lot management
Monitoring	Intruder alarm, forest fire monitoring
Consumer Electronics	Airconditioner, Washing machine, Microwave
Music and Sound	CD player, alarm timer
Office	Copy machine, Office phone
Automobiles	Engine control, Forward collision avoidance
Etc.	Game console, Garage door control

Simple program execution in your PC



Cross-platform development

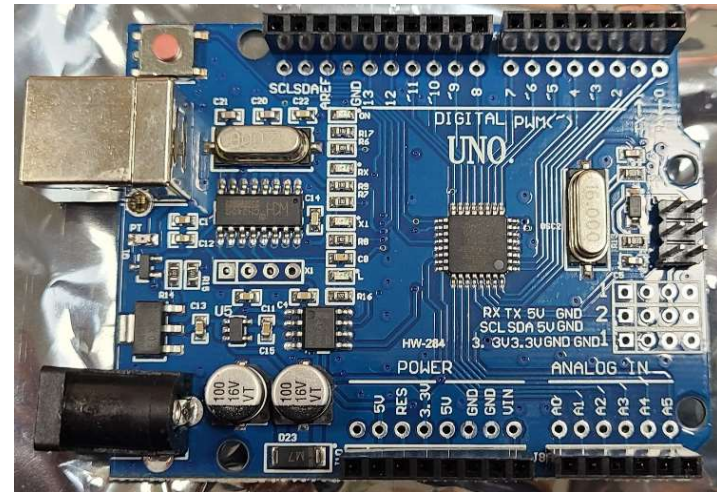
- Development system = a desktop computer
- Target system = a board with uC
- How to make program?
 - Uses a **cross-compiler** that can generate machine language object files (**program image**) that run on the target system (uC).
- How to load program?
 - The **program image** is uploaded to the target system using a **special device**.



Development system
(w/ cross-compiler)



Upload
program image



Target system
(w/ uC)

How to upload our program?

- **ISP (In System Programming)**

- Upload by SPI serial communication
- Write program into memory of uC
- Commonly, used in AVR uC

- **UART Serial communication**

- Write program into memory of uC by a bootloader
- Commonly, used in Arduino

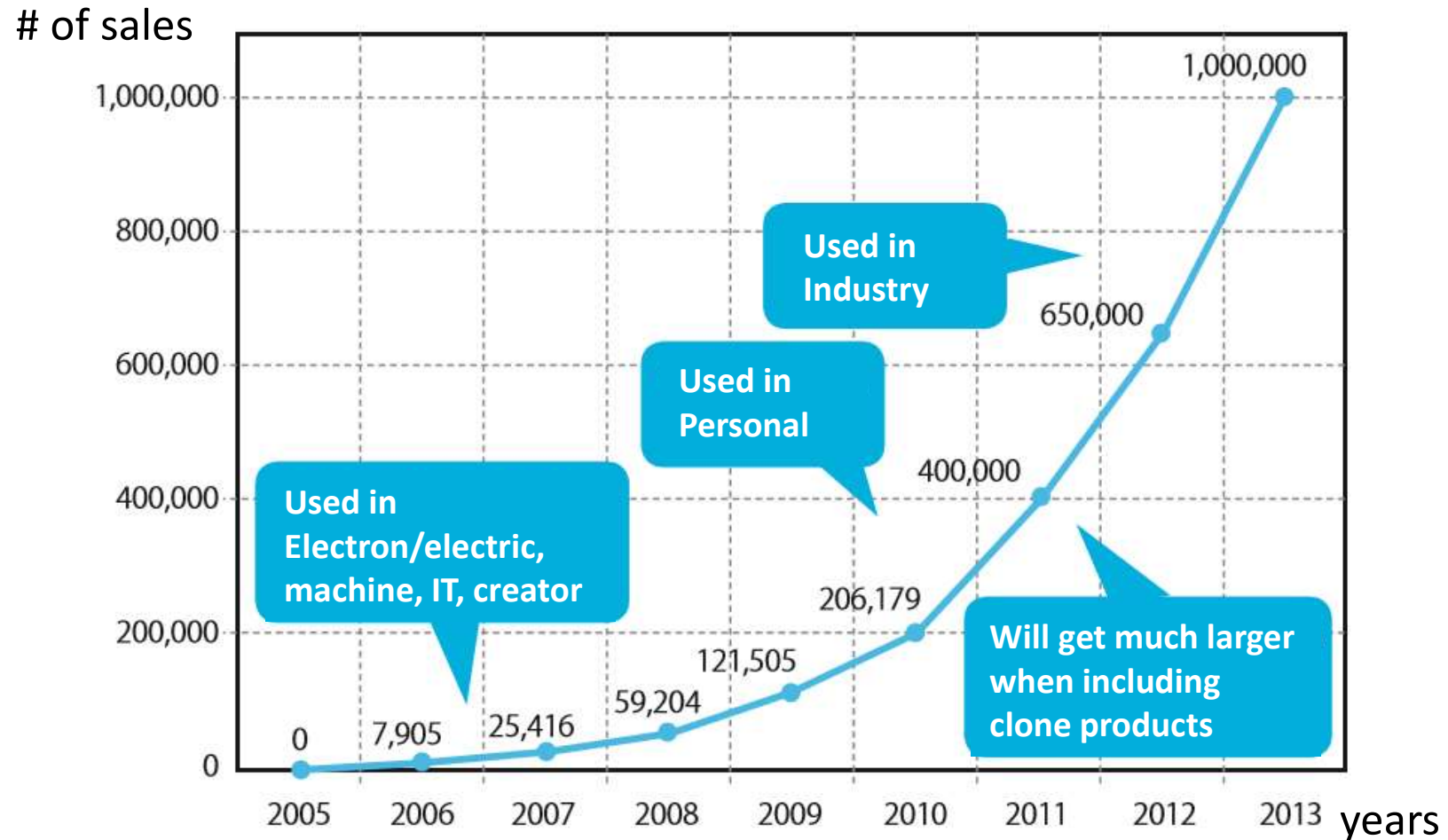
uC with S/W

- When changing or adding a simple function
 - Without additional H/W facility, it is possible by changing the software with the same I/O.
- Price of microcontrollers is continuously down
- 100% of the functions are not always used
- S/W approach has competitiveness in terms of small size, light weight, reduction of initial development time and maintenance costs

Arduino in brief

- Arduino was born in Italy at the end of 2005.
- Anyone can use it easily and cheaply, and other users can use open source materials on the Internet for free.
- Arduino is highly appreciated in that it is not too difficult for beginners, and even professional engineers can use it with skills outside of their field of expertise.
- Arduino adopts the open source hardware concept

Number of Arduino Product Sales (accumulation)



Source: 길벗, "모두의 아두이노"

Why so popular?

- **Easy & Cheap**

- Arduino was developed as a uC board for cheap textbooks so that students majoring in electricity and electronics could study easily.

- **Low technical barriers**

- Arduino is now used by a variety of students, regardless of the arts and sciences, such as information processing, machines, design, and creators.

- **World-wide and general usages**

- Arduino is used by students, instructors, general public, and industry technicians.

- In other words, Arduino is **the standard for uC boards**

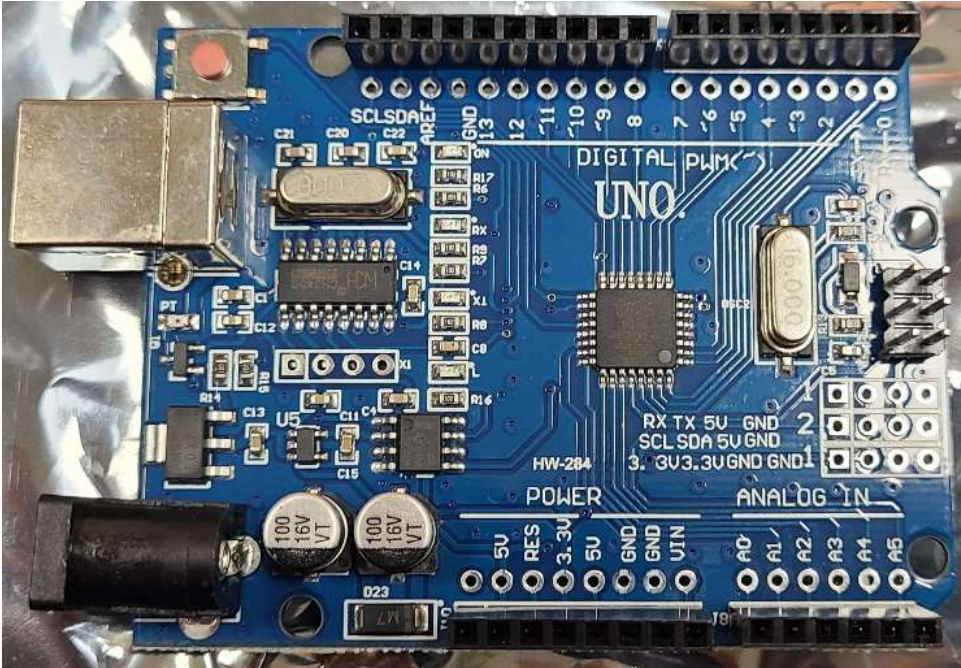
Open-source H/W?

- Developed as open source H/W, circuit diagrams and board diagrams of Arduino have been released.
- By referring to this, anyone can simply develop and sell cloned products.
- S/W development environment (**IDE, integrated development environment**) for programming the uC board can also be **downloaded free of charge** from the Internet.
- Many users **can freely upload** their own examples or programs to the Internet.

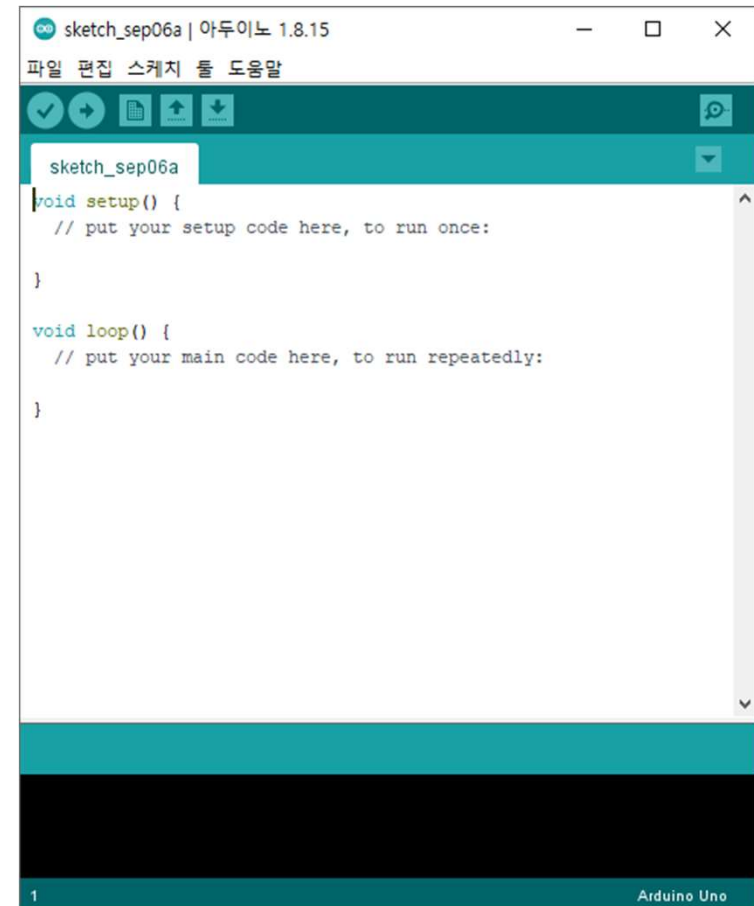
Arduino and IDE

- **Arduino = uC board (H/W) + IDE (S/W)**
- **uC board**
 - Electronic components are connected (sensors, motors, ..)
- **IDE**
 - Written programs are stored in IDE-installed computer
 - Used for uploading developed program via serial communication or UART communication
- From now, the Arduino uC board is called Arduino, and the integrated development environment is called IDE.

Arduino and IDE



**Arduino:
Microcontroller**



**Arduino:
Integrated Development
Environment (IDE)**

General steps for Arduino development

4) Execute and evaluate in Arduino



1) Compose Arduino + H/W components

How to design
I/O circuit



Connect via USB cable
(Serial comm.)

3) Compile Sketch and
upload it into Arduino



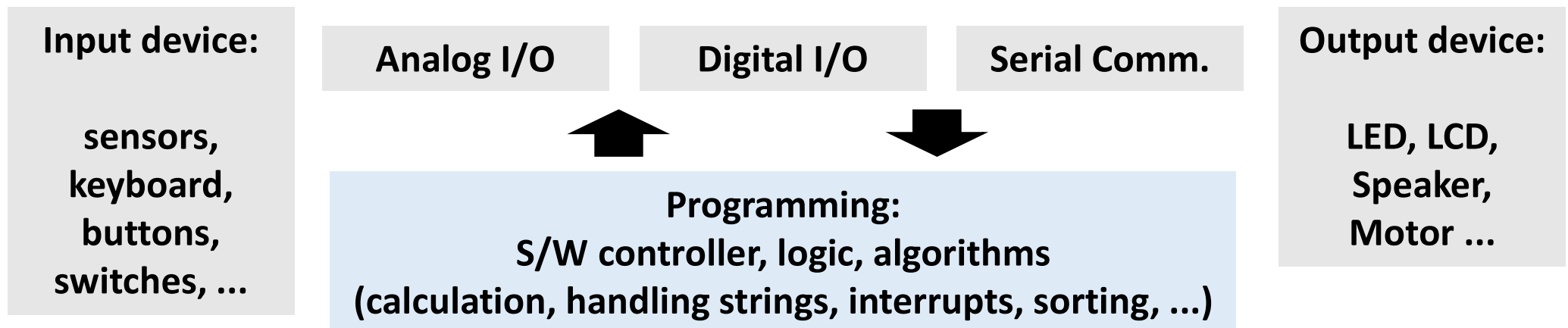
2) Writing and
debugging Sketch in IDE

How to write
Sketch
(programming)

Source: 길벗, “모두의 아두이노”

What can we do using Arduino?




- The uC board is a **miniature computer**
- You can upload programs to make them work in a variety of ways or to assign tasks to them.
- If you connect electronic components such as LED, speaker, sensor, etc. to the microcomputer board and upload a program, you can make the LED light or make a sound from the speaker.
- If you connect a temperature sensor or light sensor and upload a program, you can read the temperature or illuminance value.



Arduino terminology

- In Arduino,
 - Program == Sketch
 - Extended Arduino board = Shield
 - Control flow and process = Recipe
- By Shield, Arduino can extend the functionality
 - To make automatic control devices, robots, 3D printers, etc.
 - **Shield with USB port** can store large amounts of data using SD memory
 - **Shields with wireless capability** can form wireless networks
 - **Shield with 3G** supports the communication of mobile phones, so it can be expanded to the Internet of Things.
 - **Shield with GPS or camera** can also be applied to crime prevention, disaster prevention, and environmental protection.

Shield examples

Ethernet shield	XBee SD shield	3G shield
		
Accessible to wired LAN	Can storage data into SD memory with local sensor network	Accessible to 3G-vendors' mobile network

Source: 길벗, “모두의 아두이노”

uC in Arduino

- General characteristics
 - CPU: Atmel CPU (8 to 32 bit)
 - Memory: Flash-memory (non-volatile storage for Sketch), SRAM (volatile working memory), EEPROM (non-volatile ROM)
 - Many products are available as CPU speed, memory capacity, etc.
- Uno
 - Small, popular uC board for Arduino beginners (**recommended in this course**)
- Mega, Due
 - Support more analog and digital I/O pins, so they have enhanced scalability
- Basically, pin locations or usages are similar
 - As the kinds of pins, the number and basic voltage can be different (3.3V or 5V)

Assignment of this week

- Visit the Arduino main site
(<https://www.arduino.cc/en/Main/Products>)
- Search and compare tech specs for three entry-level H/W products
 - Uno, Leonardo, Nano
- Summarize in a single table in a Word (or HWP) document in terms of the following aspects
 - microcontroller, operating voltage, I/O pins number (analog, digital), flash memory size, SRAM size, EEPROM size, clock speed, weight
- Figure out your recommendations for each uC board in terms of usage