

C h a p t e r

02



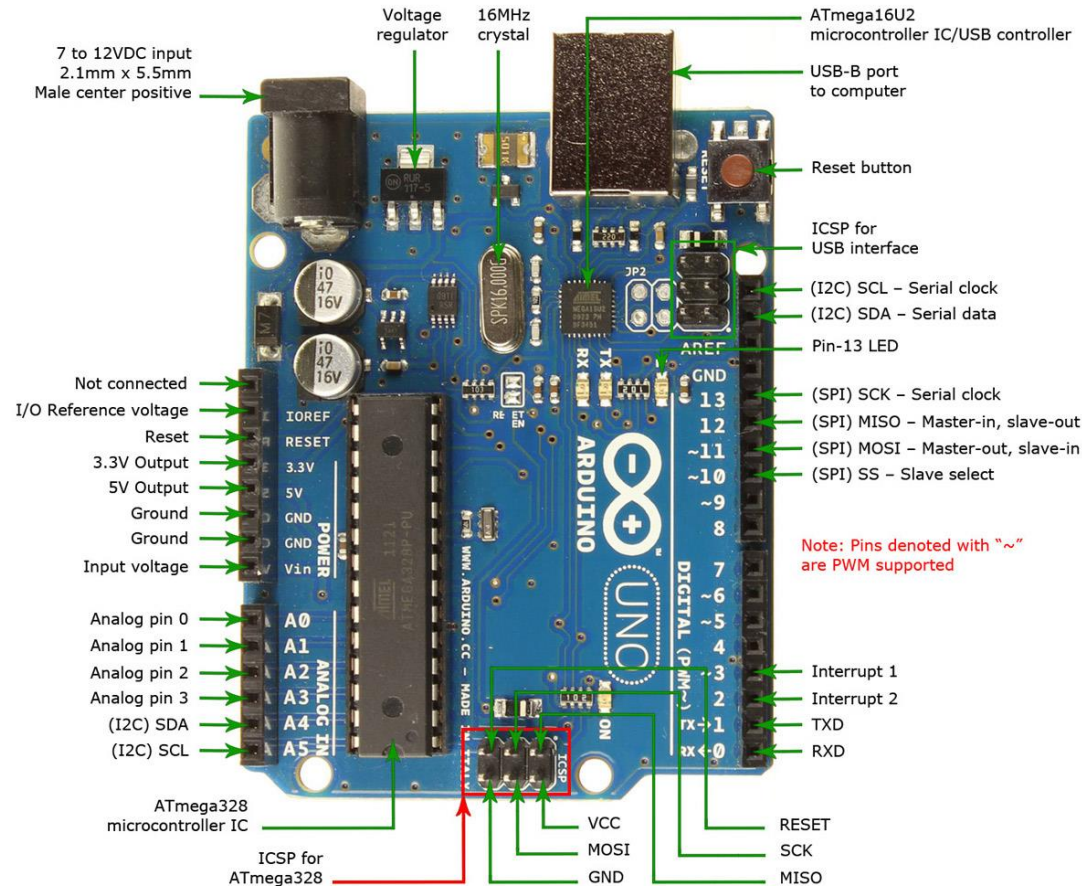
Arduino 기초

03

Arduino의 하드웨어

하드웨어 구조를 알아보고, 각 명칭과 기능을 살펴봅니다.

■ Arduino의 하드웨어



▪ 공두이노의 기본 사양

항목	세부사항
MCU	ATmega328P-PU
동작클럭	16MHz 크리스탈
메모리	플래시 메모리 32KBytes, EEPROM 1KBytes SRAM 2KBytes
USB 단자	USB-MINI5핀 타입
USB 인터페이스 칩	ATMEGA16U2
전원	USB 전원 또는 외부 DC 5-6V 전원 잭 3.3 V 레귤레이터 내장
기타	펌웨어 업데이트를 위한 MINI ISP 포트 전원 선택 스위치 (USB 전원 또는 외부 전원) 블루투스 확장 쉴드의 설정을 위한 통신 선택 스위치 (NC 또는 USB)
확장기능	<ul style="list-style-type: none"> - SCL/SDA 핀 기능 - ADC6/ADC7 핀 기능 - 외부 전원 출력 단자 - 통신 선택 스위치 - 보드 고정용 홀 다수

04

아두이노(Arduino)의 확장 실드

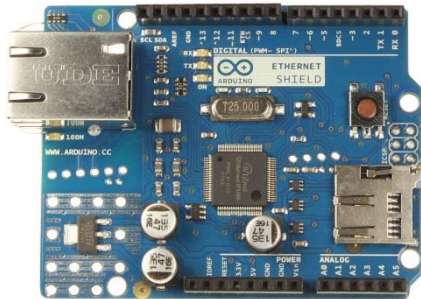
아두이노 보드에 기능을 확장하기 위해 제작된 실드의 종류와 기능에 대해서 알아봅니다.

■ 아두이노(Arduino)의 확장 실드

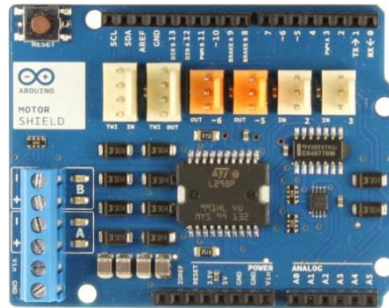
아두이노는 기본 제어 보드 외에 기능을 확장하기 위한 다양한 종류의 실드 보드가 존재합니다.

이더넷 실드(Ethernet Shield)

LAN 선을 연결하여 인터넷에 접속이 가능하며, 이를 이용한 웹 서버 및 네트워크 제어를 위해 사용합니다.

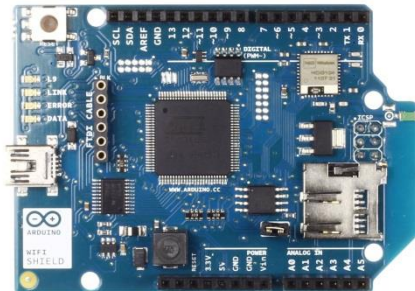


모터 실드 (MOTOR Shield)

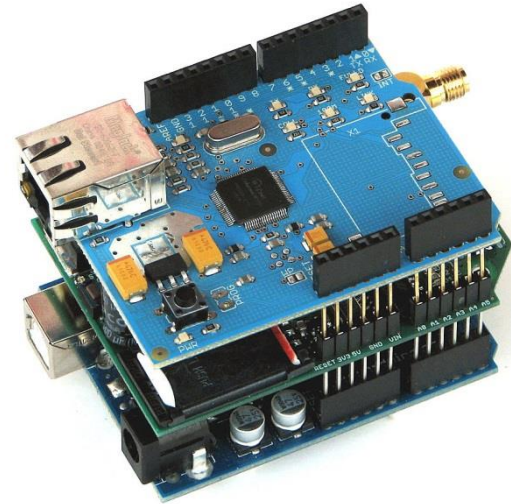


와이파이 실드(WIFI Shield)

무선으로 네트워크에 접속하여 네트워크 상의 제어를 위해 사용합니다.



GPS 실드 (GPS Shield)



적층형 구조

아두이노의 확장 핀을 통해 기본 제어 보드와 확장 실드를 적층(위로 계속 쌓아 올림) 해서 연결합니다.

05

개발환경 (Development Environment)

여러 어플리케이션에 응용하기 위해 개발 환경을 구축하는 방법에 대해서 알아본다.

■ 하드웨어의 준비

공두이노는 기본 제어 보드와 USB 케이블 만으로도 최소 개발하기 위한 하드웨어 준비가 완료됩니다.



공두이노 제어 보드
(Arduino Diecimila Compatible)



USB 미니5P 케이블

■ 소프트웨어의 설치

공두이노는 사용자가 PC를 이용하여 제어 프로그램을 작성하고, 이를 제어 보드에 다운로드하여 동작합니다. PC에서 제어 프로그램을 작성하기 위해서는 2개의 소프트웨어를 설치하여야 하며, 본 교재에서는 MS 윈도우즈를 기준으로 설명합니다.

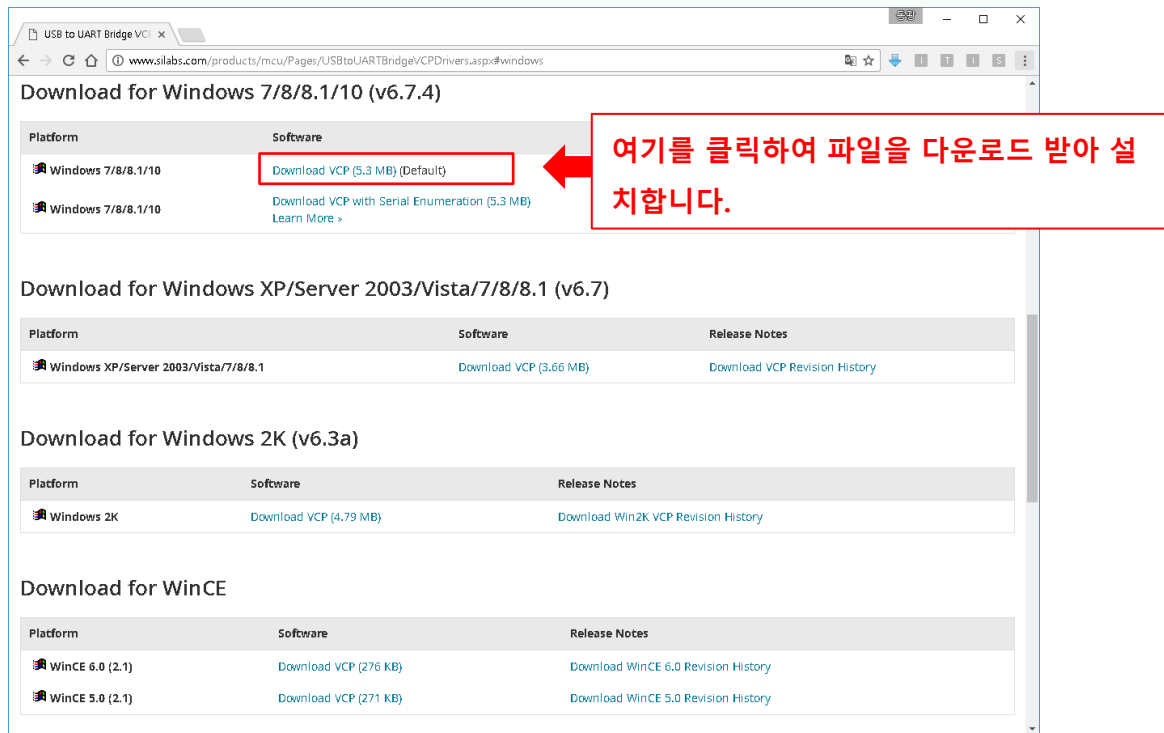
1. USB 드라이버의 설치

프로그램을 작성하는 PC와 공두이노 제어 보드는 USB 케이블을 이용해서 연결됩니다.

공두이노 제어보드를 USB 케이블에 연결하기 전에 먼저 제어보드의 USB 인터페이스 칩인 Silicon Labs 사의 CP210x에 대한 드라이버 파일을 설치하여야 합니다.

<http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx>

홈페이지를 접속합니다. (UNO사용자는 해당 아두이노IDE설치 디렉토리 하단에 /Drivers 밑에 위치)



Download for Windows 7/8/8.1/10 (v6.7.4)

Platform	Software
Windows 7/8/8.1/10	Download VCP (5.3 MB) (Default)
Windows 7/8/8.1/10	Download VCP with Serial Enumeration (5.3 MB) Learn More >

Download for Windows XP/Server 2003/Vista/7/8/8.1 (v6.7)

Platform	Software	Release Notes
Windows XP/Server 2003/Vista/7/8/8.1	Download VCP (3.66 MB)	Download VCP Revision History

Download for Windows 2K (v6.3a)

Platform	Software	Release Notes
Windows 2K	Download VCP (4.79 MB)	Download Win2K VCP Revision History

Download for WinCE

Platform	Software	Release Notes
WinCE 6.0 (2.1)	Download VCP (276 KB)	Download WinCE 6.0 Revision History
WinCE 5.0 (2.1)	Download VCP (271 KB)	Download WinCE 5.0 Revision History

※ 카페 게시글 252번에 있는 드라이버 파일을 다운받으셔도 됩니다.

2. 아두이노 IDE 의 설치

아두이노 공식 홈페이지 (<http://www.Aduino.cc>)에 접속(구글 크롬추천)하여 공개하여 배포하고 있는 아두이노 IDE 소프트웨어를 다운로드 받습니다.

<http://arduino.cc/en/Main/Software>

Support the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



압축파일(ZIP)을 다운로드 받아 저장합니다.



JUST DOWNLOAD

CONTRIBUTE & DOWNLOAD

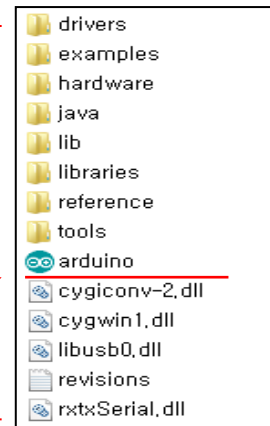
다운로드한 압축 파일을 본인이 원하는 경로에 (예: c:\ Arduino 등)에 풀어 둡니다.

아두이노 IDE 소프트웨어는 자바로 작성된 파일로
별도로 설치할 필요가 없습니다.

C:\Arduino 폴더 밑에 아두이노 관련 파일
들이 있게 합니다.

아두이노
관련 파일과 폴더

아두이노 IDE
실행 파일



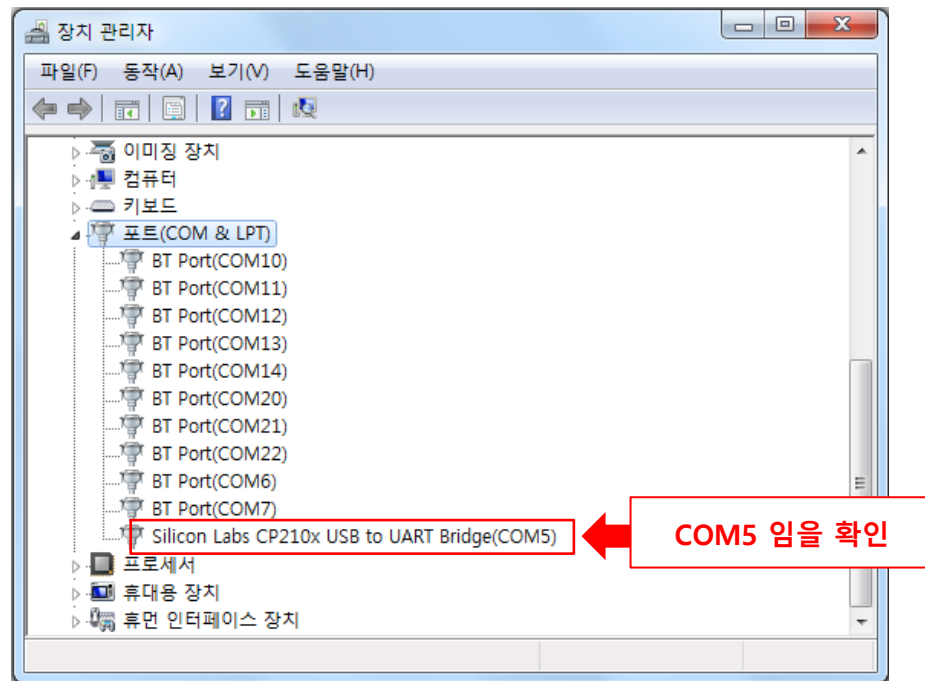
3. 하드웨어 연결

하드웨어를 다음 순서로 연결합니다.

- ① 공두이노 제어 보드에 USB 케이블을 연결합니다.

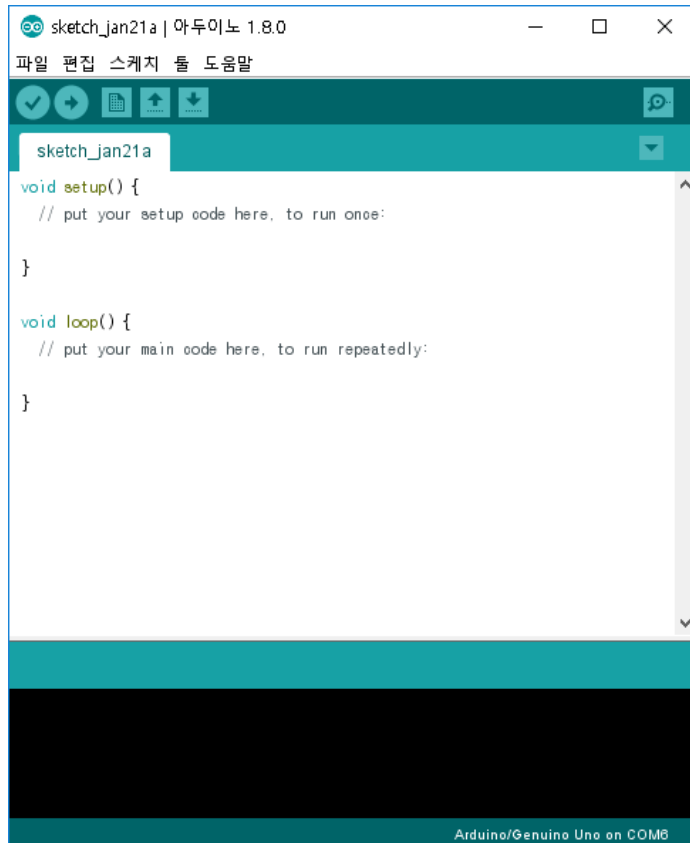
이 때, 제어보드의 전원 스위치가 USB로, 통신 선택 스위치가 USB(default)로 되어 있는지 확인합니다.

- ① USB의 다른 쪽을 PC에 연결합니다. 이 때, USB 드라이버 소프트웨어가 제대로 설치되어 있으면 공두이노를 자동으로 인식하며, 추후 재설치 불필요합니다.
- ② 연결된 공두이노의 USB 포트를 윈도우의 장치 관리자에서 확인합니다.(예:COM5)



4. 아두이노 IDE의 실행

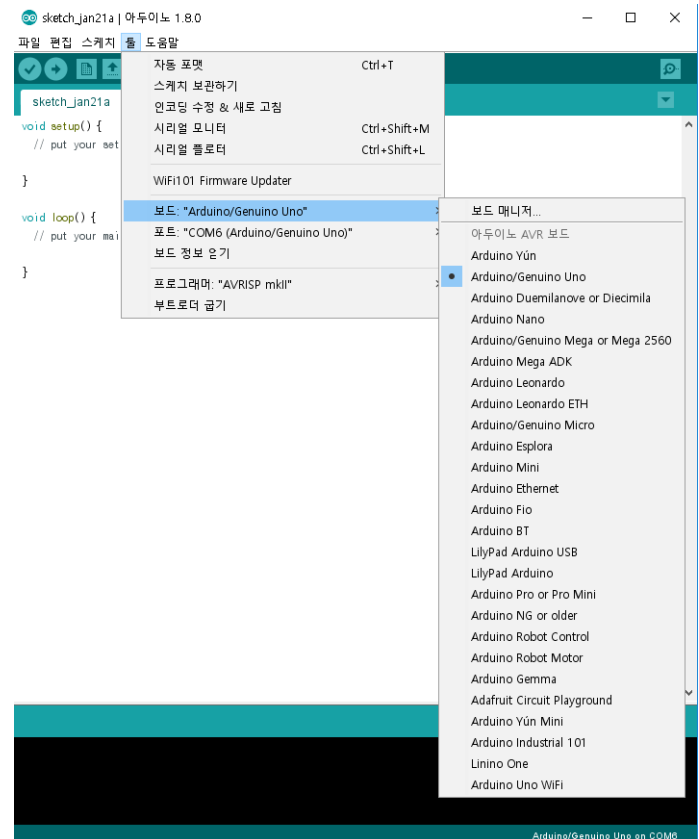
아두이노 설치된 해당 폴더인 C:\Arduino에서 arduino.exe 실행 파일을 더블 클릭하여 실행합니다. PC의 화면에 아래와 같은 아두이노 IDE 프로그램이 실행됩니다.



5. 아두이노 설정

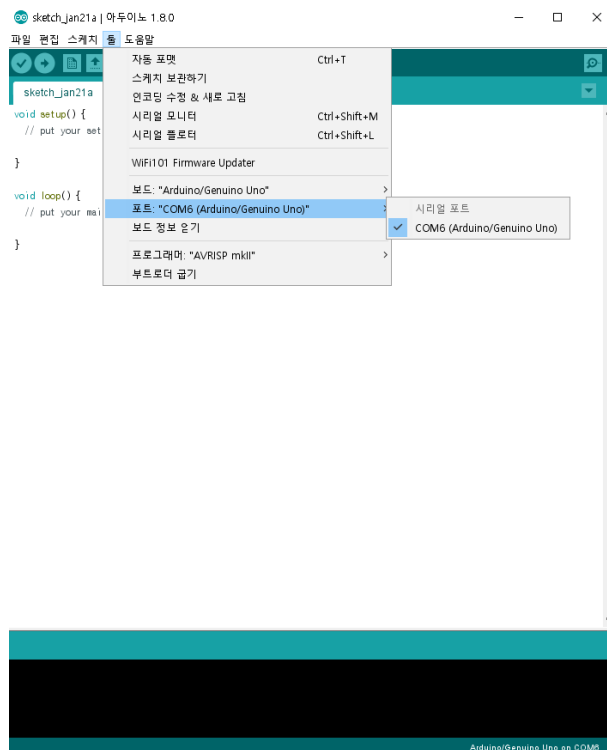
아두이노 IDE를 처음 실행하면, 사용하고자 하는 아두이노 보드에 대한 설정을 합니다. 현재 사용되는 공두이노(Gongduino) 설정을 위해 다음과 같이 합니다.(Uno사용자는 [Arduino Uno](#)선택)

① 도구 → 보드 → “Aduino Duemilanove or Diecimila” 선택



② 도구 → 시리얼 포트 → “COM5” 선택

시리얼 포트는 연결된 PC마다 달라지므로, 앞의 “3. 하드웨어 연결” 에서 장치관리자를 통해 확인이 가능합니다.



③ 모든 설정이 완료되면 아두이노 IDE의 하단 상태바에 아래와 같이 설정된 상태가 표시됩니다.



06

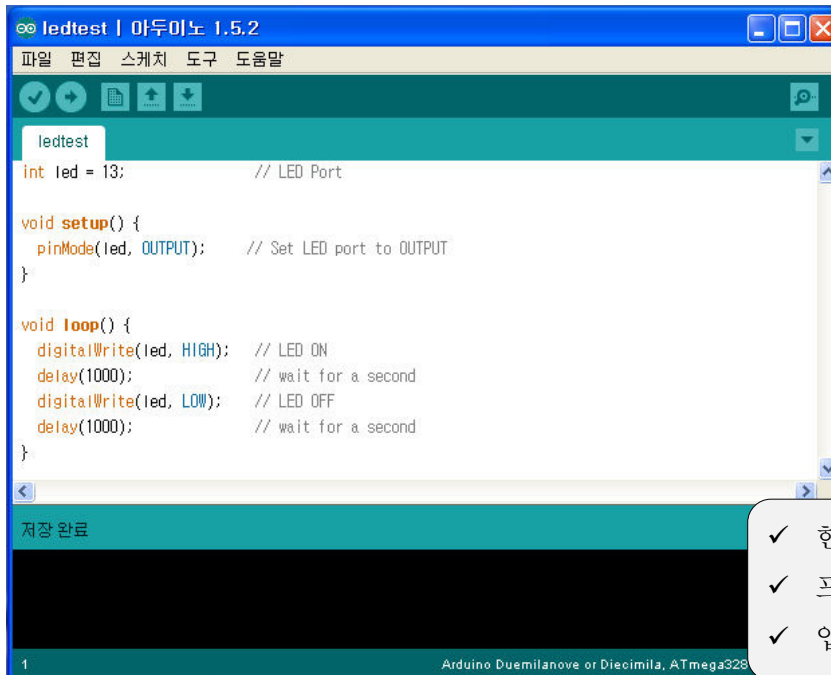
무조건 따라하기

공두이노 제어장치와 아두이노 IDE를 사용하여 간단한 프로그램을 작성하여 동작하여 보도록 하겠습니다.

■ 첫 프로그램의 작성

공두이노의 기본 사용법을 익히기 위해 아래 프로그램을 작성하고 동작시켜 보도록 하겠습니다.

1. 프로그램의 입력



The screenshot shows the Arduino IDE interface with a file named 'ledtest'. The code is as follows:

```
int led = 13;           // LED Port

void setup() {
  pinMode(led, OUTPUT); // Set LED port to OUTPUT
}

void loop() {
  digitalWrite(led, HIGH); // LED ON
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // LED OFF
  delay(1000);             // wait for a second
}
```

At the bottom of the IDE, the status bar indicates 'Arduino Duemilanove or Diecimila, ATmega328P'.

- ✓ 한글 입력은 지원하지 않음.
- ✓ 프로그램의 “// 부분”은 설명이므로 입력하지 않아도 됨.
- ✓ 입력이 정상적이면 텍스트의 색상이 위 화면과 같이 자동으로 바뀜.

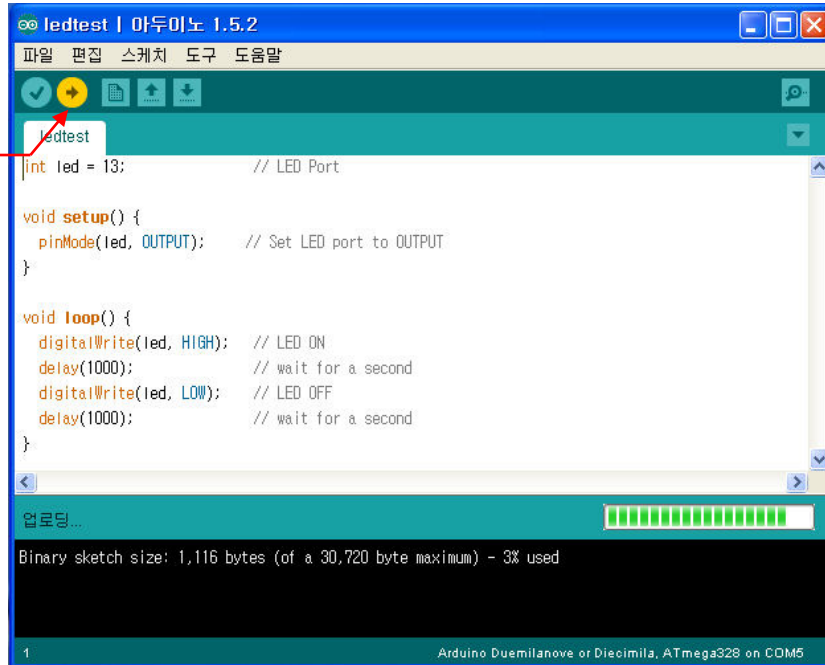
위 화면과 같이 편집 창에 프로그램을 입력하고 “ledtest” 라는 이름으로 저장합니다.

2. 프로그램의 업로드 (Upload)

작성한 프로그램을 공두이노 보드에 업로드(Upload) 합니다.

<파일> → <업로드> 를 선택하거나, Ctrl + U 단축키를 누르거나, 아래 그림과 같이 업로드 단축 버튼을 누릅니다.

업로드
단축버튼



정상적인 프로그램이 작성되었으면(오류가 없고, 공두이노 연결이 잘 되어 있는 상태), 프로그램 컴파일(Compile) 과정을 거쳐 공두이노 보드에 업로드(Upload) 되어, 바로 동작됨을 확인할 수 있습니다.

오류 증상

1. 컴파일 오류

프로그램을 작성할 때, 잘못된 명령어나 구문 오류 등이 발생되어 컴파일 과정(C언어를 기계어 코드로 변환)을 수행하지 못함. ⇒ 프로그램을 잘 살펴보고 잘못된 부분을 수정

2. not in sync 오류

하드웨어의 연결이 잘못되어 있거나, 시리얼 포트 설정이 잘못되어 연결이 되지 않아 공두이노 보드에 업로드 하지 못함. ⇒ 하드웨어 확인, 케이블과 포트 확인하여 다시 연결함

C h a p t e r

03



프로그램 기초

아두이노 IDE에서 사용하는 프로그램의 기본 구조에 대해서 이해합니다.

■ 프로그램의 기본 구조

아두이노 IDE의 프로그램은 다음 2가지 함수를 기본으로 사용합니다.

1. void setup() { }

이 함수는 프로그램이 실행하기 전에 설정하여야 할 사항들 (연결된 소자에 따른 입출력 포트, 시리얼 통신, 라이브러리 등)을 명시합니다.

2. void loop() { }

아두이노 보드가 반복적으로 실행하는 프로그램 부분을 기술합니다. 대부분의 제어보드는 전원이 ON 되면서 OFF 될 때까지 지속적인 반복 동작을 하게 됩니다. C언어에서는

```
void main (void)
{
    while (1){
        // 반복되는 프로그램
    }
}
```

와 같은 동작을 수행합니다.

- ✓ 프로그램의 처음 시작은 항상 setup() 함수와 loop() 함수가 있어야 함.
- ✓ 아두이노 IDE의 <파일> → <새 파일>을 선택하면 자동으로 setup() 함수와 loop() 함수가 생성됨.

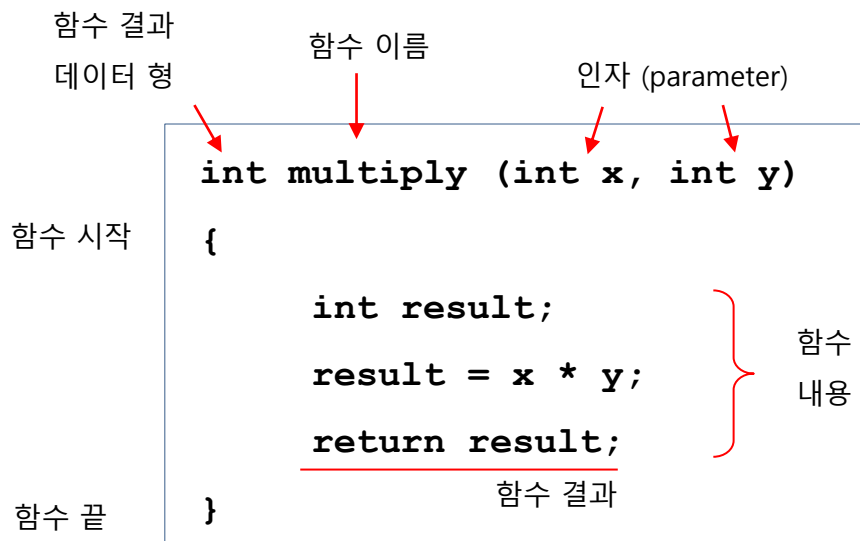
■ 선언문

1. 변수 선언 (Variable Declaration)

아두이노는 다음의 변수 데이터 형태를 지원합니다.

boolean, char, unsigned char, byte,
int, unsigned int, word, long, unsigned long,
short, float, double, string, array

2. 함수 선언 (Function Declaration)



- ✓ 결과를 사용하지 않는 함수의 경우에는 void 로 함수 데이터 형을 선언합니다.
- ✓ 함수 내부에 넘겨줄 인자가 없는 경우에도 void 를 사용합니다.

```
void 함수이름 (void) {
    // 무조건 수행만 하는 프로그램을 기술
}
```

▪ 제어문 (Control)

- if
- if ... else
- for
- switch ... case
- while
- do ... while
- break
- continue
- return

▪ 산술연산자 (Arithmetic)

- + (덧셈)
- - (뺄셈)
- * (곱셈)
- / (나눗셈)
- % (나머지)

▪ 논리연산자 (Boolean)

- && (그리고)
- || (또는)
- ! (부정)

▪ 기타

- ; (세미콜론)
- { }
- //
- /* */

▪ 비교연산자 (Comparison)

- ==
- !=
- <
- >
- <=
- >=

▪ 조합연산자 (Compound)

- ++
- --
- +=
- -=
- *=
- /=

✓ 제어문과 연산자는 기본적인 C 언어와 같으므로, 여기에서는 문법적인 구조에 대한 설명을 생략합니다.

▪ 변수와 상수

1. 상수(Constants)

- HIGH / LOW
- INPUT / OUTPUT
- true / false

3. 변수 범위 및 한정

- static
- volatile
- const
- PROGMEM

4. 기타

- cast
- sizeof()

2. 데이터 형

- boolean
- char
- byte
- int
- unsigned int
- long
- unsigned long
- float
- double
- string
- array

▪ 아두이노 제공 함수

1. 디지털 신호 입출력

- pinMode (pin, mode)
- digitalWrite (pin, value)
- int digitalRead (pin)

3. 향상된 입출력

- shiftOut (dataPin, clockPin, bitOrder, value)
- unsigned long pulseIn (pin, value)

5. 수학

- min (x, y)
- max (x, y)
- abs (x, y)
- constrain (x, a, b)
- map (value, fromLow, fromHigh, toLow, toHigh)
- pow (base, exponent)
- sqrt (x)

2. 아날로그 신호 입출력

- int analogRead (pin)
- analogWrite (pin, value) – PWM

4. 시간 (Time)

- unsigned long millis()
- delay (ms)
- delayMicroseconds(us)

6. 삼각 함수

- `sin (rad)`
- `cos (rad)`
- `tan (rad)`

8. 외부 인터럽트

- `attachInterrupt (interrupt, function, mode)`
- `detachInterrupt (interrupt)`

9. 인터럽트

- `interrupts()`
- `noInterrupts()`

7. 무작위 수 (random)

- `randomSpeed (seed)`
- `long random (max)`
- `long random (min, max)`

10. 시리얼 통신

- `Serial.begin (speed)`
- `int Serial.available()`
- `int Serial.read()`
- `Serial.flush()`
- `Serial.print (data)`
- `Serial.println (data)`

프로그램에 많이 사용되는 ASCII 코드표

DEC	HEX	OCT	Char	DEC	HEX	OCT	Char	DEC	HEX	OCT	Char
0	00	000	Ctrl-@ NUL	43	2B	053	+	86	56	126	V
1	01	001	Ctrl-A SOH	44	2C	054	,	87	57	127	W
2	02	002	Ctrl-B STX	45	2D	055	-	88	58	130	X
3	03	003	Ctrl-C ETX	46	2E	056	.	89	59	131	Y
4	04	004	Ctrl-D EOT	47	2F	057	/	90	5A	132	Z
5	05	005	Ctrl-E ENQ	48	30	060	0	91	5B	133	[
6	06	006	Ctrl-F ACK	49	31	061	1	92	5C	134	\
7	07	007	Ctrl-G BEL	50	32	062	2	93	5D	135]
8	08	010	Ctrl-H BS	51	33	063	3	94	5E	136	^
9	09	011	Ctrl-I HT	52	34	064	4	95	5F	137	_
10	0A	012	Ctrl-J LF	53	35	065	5	96	60	140	`
11	0B	013	Ctrl-K VT	54	36	066	6	97	61	141	a
12	0C	014	Ctrl-L FF	55	37	067	7	98	62	142	b
13	0D	015	Ctrl-M CR	56	38	070	8	99	63	143	c
14	0E	016	Ctrl-N SO	57	39	071	9	100	64	144	d
15	0F	017	Ctrl-O SI	58	3A	072	:	101	65	145	e
16	10	020	Ctrl-P DLE	59	3B	073	:	102	66	146	f
17	11	021	Ctrl-Q DC1	60	3C	074	<	103	67	147	g
18	12	022	Ctrl-R DC2	61	3D	075	=	104	68	150	h
19	13	023	Ctrl-S DC3	62	3E	076	>	105	69	151	i
20	14	024	Ctrl-T DC4	63	3F	077	?	106	6A	152	j
21	15	025	Ctrl-U NAK	64	40	100	@	107	6B	153	k
22	16	026	Ctrl-V SYN	65	41	101	A	108	6C	154	l
23	17	027	Ctrl-W ETB	66	42	102	B	109	6D	155	m
24	18	030	Ctrl-X CAN	67	43	103	C	110	6E	156	n
25	19	031	Ctrl-Y EM	68	44	104	D	111	6F	157	o
26	1A	032	Ctrl-Z SUB	69	45	105	E	112	70	160	p
27	1B	033	Ctrl-[ESC	70	46	106	F	113	71	161	q
28	1C	034	Ctrl-\ FS	71	47	107	G	114	72	162	r
29	1D	035	Ctrl-] GS	72	48	110	H	115	73	163	s
30	1E	036	Ctrl-^ RS	73	49	111	I	116	74	164	t
31	1F	037	Ctrl_ US	74	4A	112	J	117	75	165	u
32	20	040	Space	75	4B	113	K	118	76	166	v
33	21	041	!	76	4C	114	L	119	77	167	w
34	22	042	"	77	4D	115	M	120	78	170	x
35	23	043	#	78	4E	116	N	121	79	171	y
36	24	044	\$	79	4F	117	O	122	7A	172	z
37	25	045	%	80	50	120	P	123	7B	173	{
38	26	046	&	81	51	121	Q	124	7C	174	
39	27	047	'	82	52	122	R	125	7D	175	}
40	28	050	(83	53	123	S	126	7E	176	