



LED 디바이스 제어

조 진 성
경희대학교 컴퓨터공학과
Mobile & Embedded System Lab.



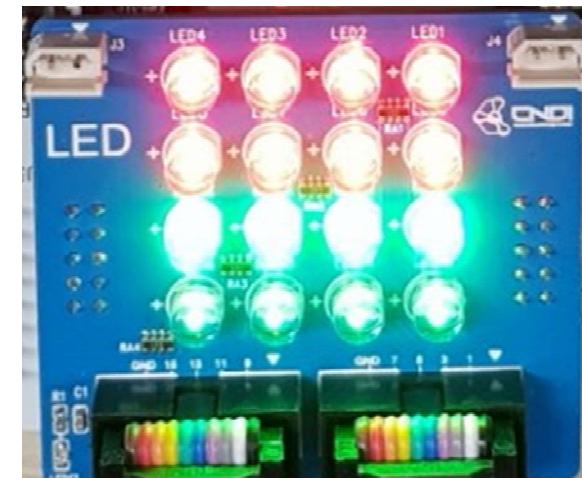
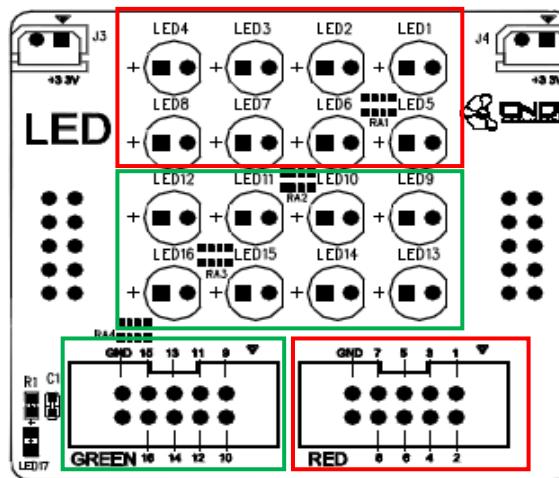
하드웨어 구성



❖ Arduino MEGA ADK



❖ LED 모듈

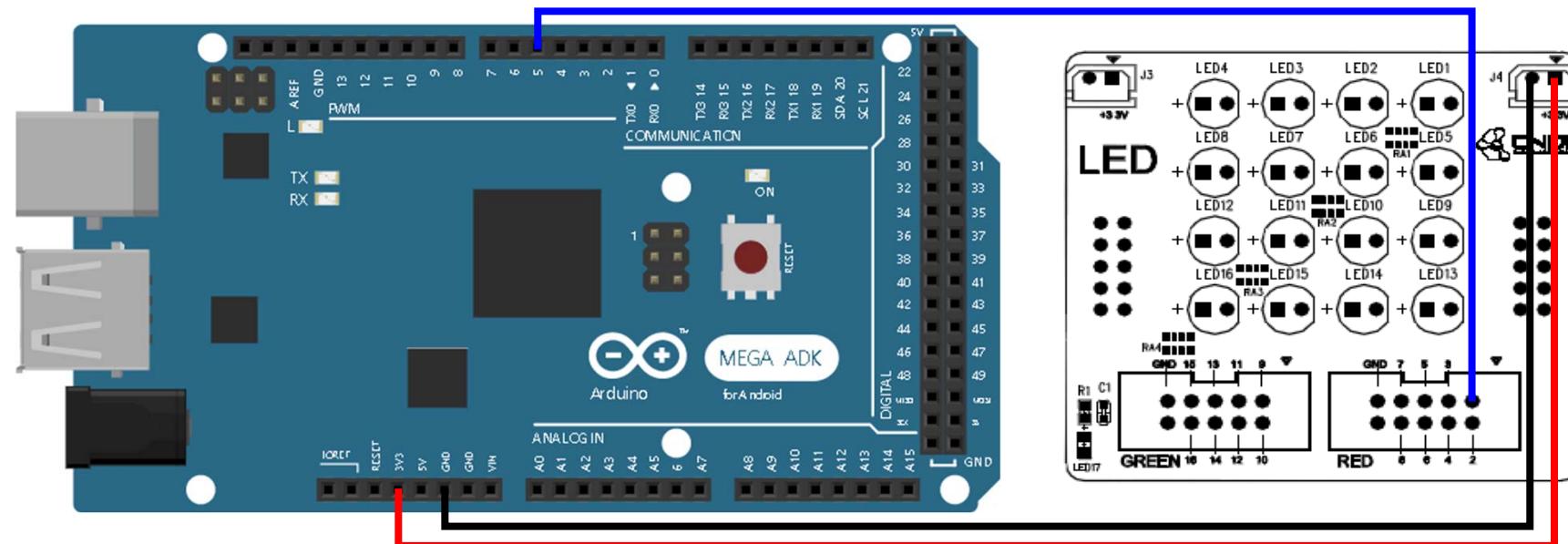


하드웨어 구성 (1)



❖ LED 1개 ON/OFF

- 500ms 간격으로 LED 1개를 ON/OFF
- 실습을 위한 Arduino MEGA ADK와 LED 결선 방법
 - Arduino MEGA ADK의 DIGITAL 5와 LED 모듈의 RED 포트의 핀 1을 연결
 - Arduino MEGA ADK의 3.3V와 LED 모듈의 J4 포트의 +3.3V와 연결
 - Arduino MEGA ADK의 GND(0V)와 LED 모듈의 J4 포트의 좌측에 연결
- 결선 그림



실습 예제 (1)



❖ LED 1개 ON/OFF

▪ 스케치 작성

- lab1-2_1.ino

```
#define LED 5 // Arduino PIN 정의

void setup() {
    pinMode(LED, OUTPUT); // 5번 PIN을 출력으로 설정
}

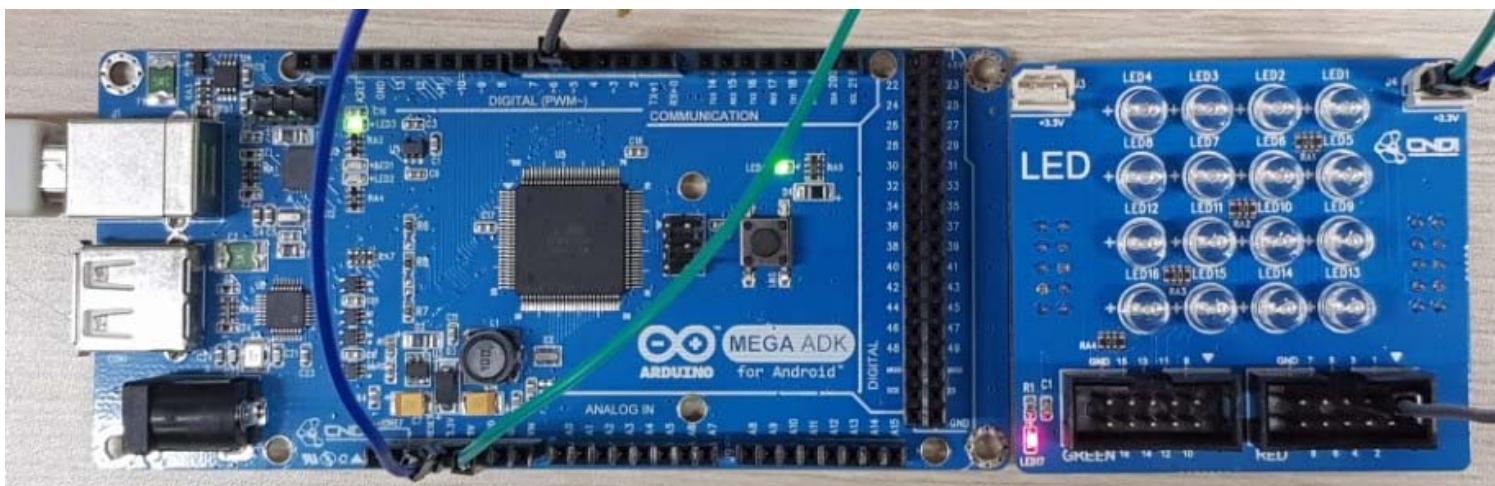
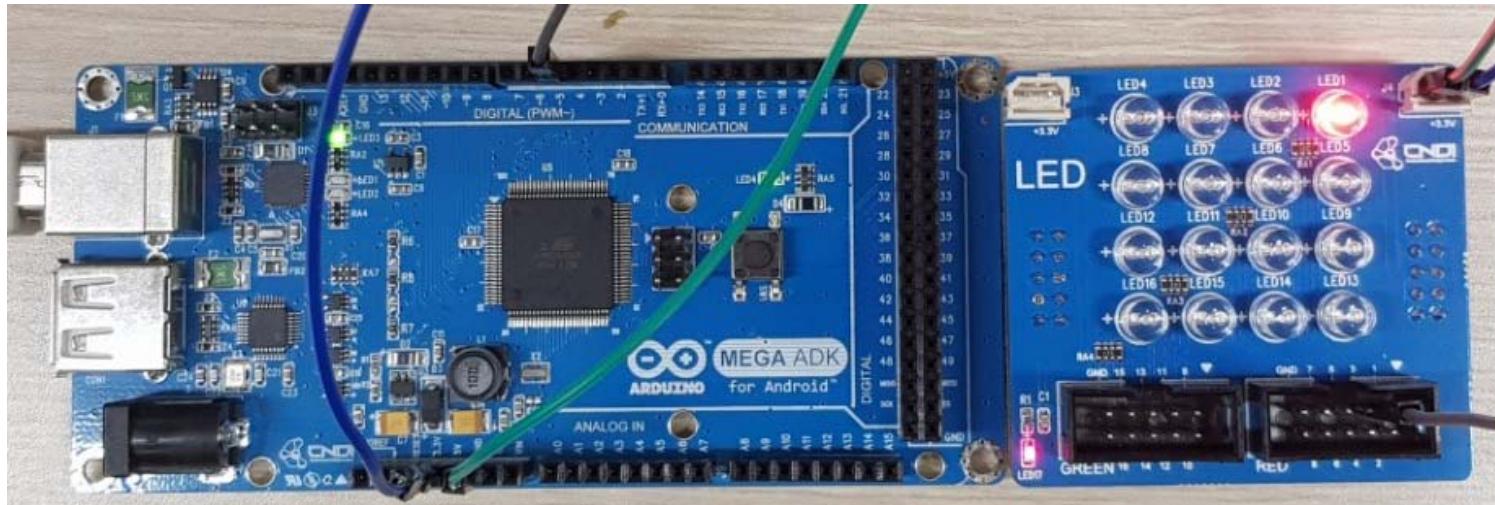
void loop() {
    digitalWrite(LED, HIGH); // 5번 PIN에 HIGH 출력(RED LED 1 ON)
    delay(500); // 500ms 지연
    digitalWrite(LED, LOW); // 5번 PIN에 LOW 출력(RED LED 1 OFF)
    delay(500); // 500ms 지연
}
```

실습 예제 (1)



❖ LED 1개 ON/OFF

▪ 동작 화면



하드웨어 구성 (2)



❖ LED 4개 ON/OFF

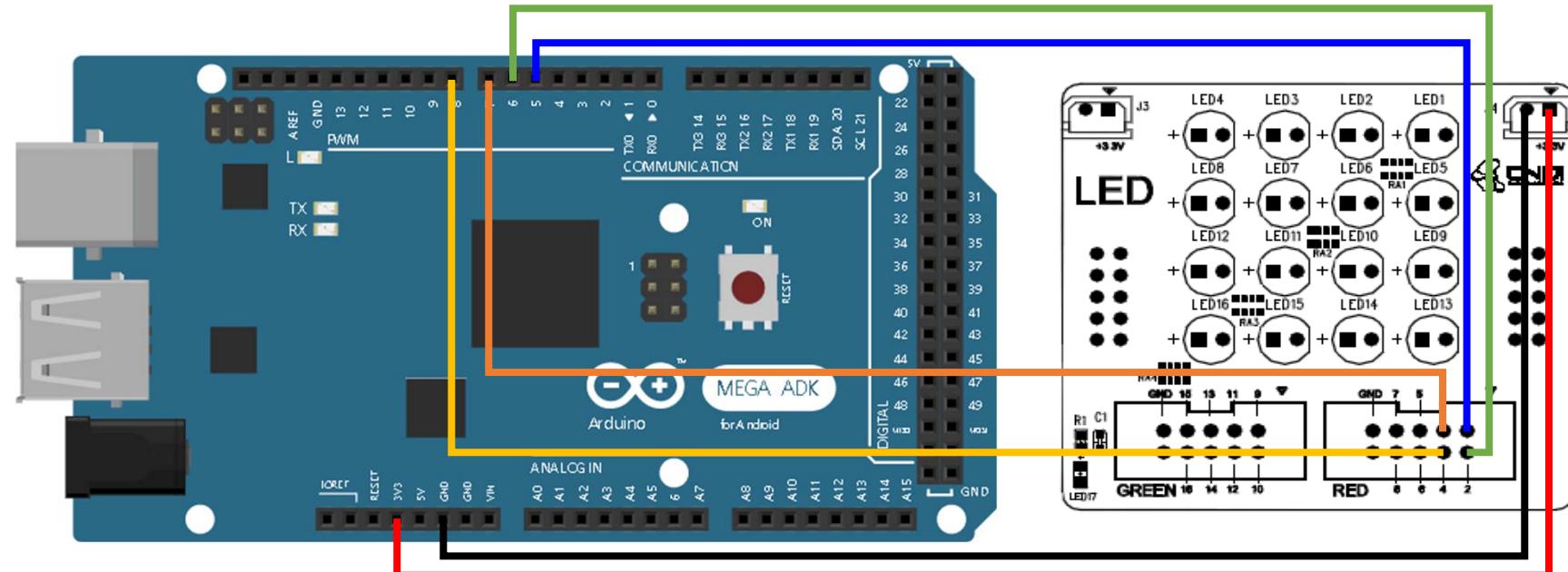
- 500ms 간격으로 LED 4개를 순차적으로 ON/OFF
- 실습을 위한 Arduino MEGA ADK와 LED 결선 방법
 - Arduino MEGA ADK의 DIGITAL 5와 LED 모듈의 RED 포트의 핀 1을 연결
 - Arduino MEGA ADK의 DIGITAL 6과 LED 모듈의 RED 포트의 핀 2를 연결
 - Arduino MEGA ADK의 DIGITAL 7과 LED 모듈의 RED 포트의 핀 3을 연결
 - Arduino MEGA ADK의 DIGITAL 8과 LED 모듈의 RED 포트의 핀 4를 연결
 - Arduino MEGA ADK의 3.3V와 LED 모듈의 J4 포트의 +3.3V와 연결
 - Arduino MEGA ADK의 GND(0V)와 LED 모듈의 J4 포트의 좌측에 연결

하드웨어 구성 (2)



❖ LED 4개 ON/OFF

▪ 결선 그림



실습 예제 (2)



❖ LED 4개 ON/OFF

▪ 스케치 작성

- lab1-2_2.ino

```
int Led[4] = {5, 6, 7, 8}; // Arduino PIN 정의

void setup() {
    int i;
    for(i=0; i<4; i++)
        pinMode(Led[i], OUTPUT); // 각 PIN을 출력으로 설정
}

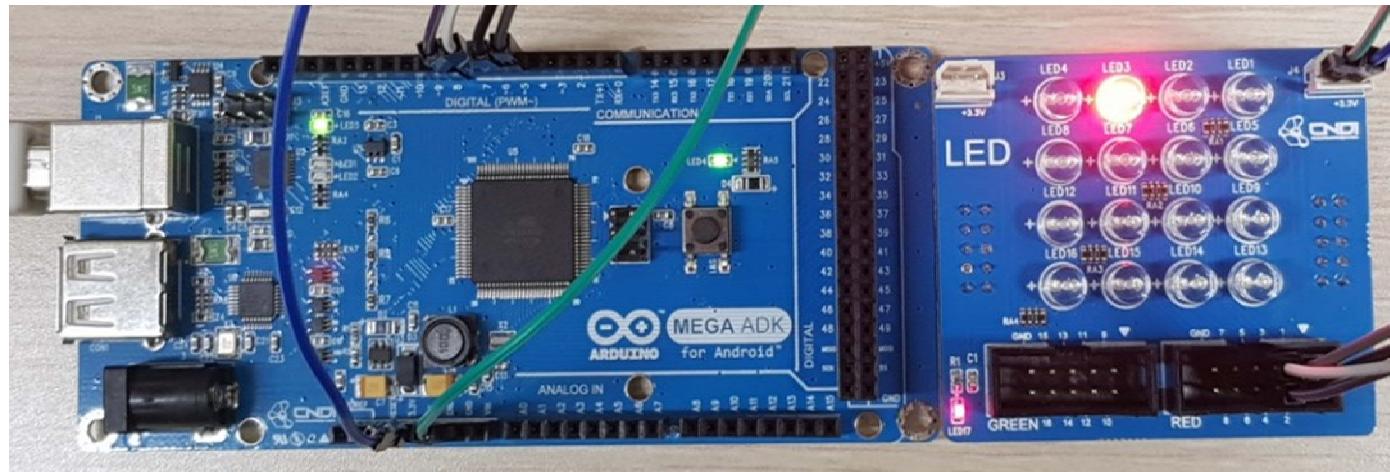
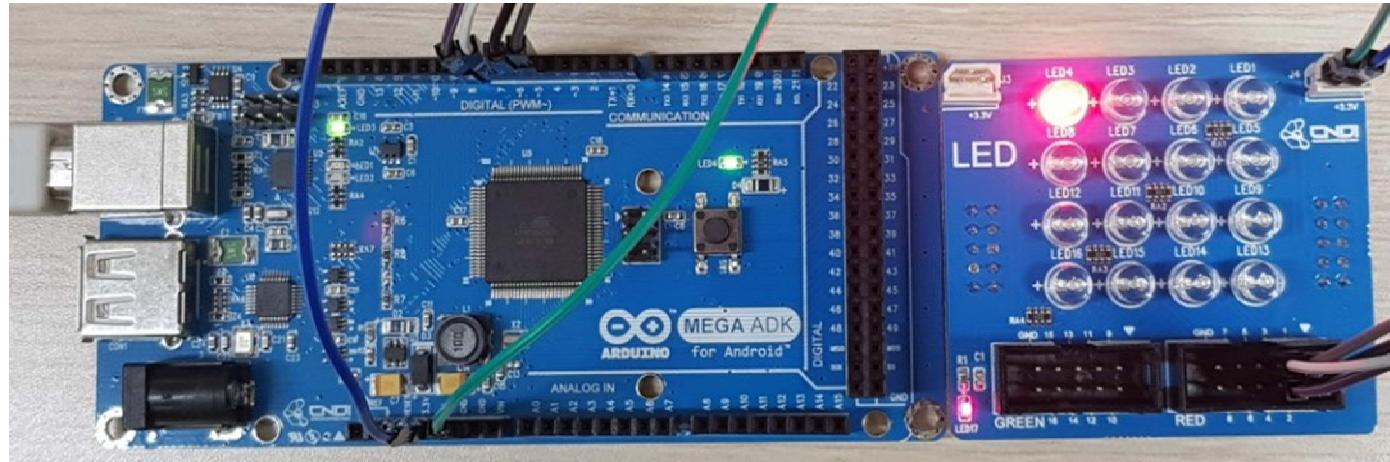
void loop() {
    int i;
    for(i=0; i<4; i++) {
        digitalWrite(Led[i], HIGH); // 순차적으로 PIN에 HIGH 출력(RED LED ON)
        delay(500); // 500ms 지연
        digitalWrite(Led[i], LOW); // 순차적으로 PIN에 LOW 출력(RED LED OFF)
        delay(500); // 500ms 지연
    }
}
```

실습 예제 (2)



❖ LED 4개 ON/OFF

- 동작 화면(일부)

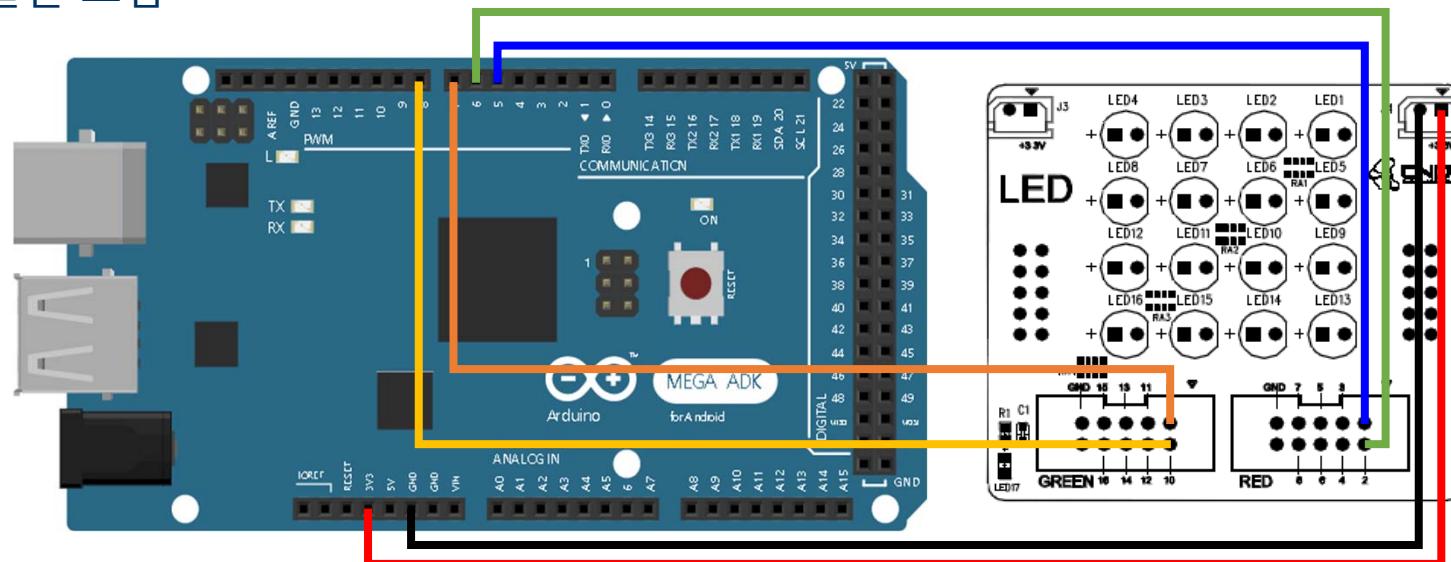


하드웨어 구성 (3)



❖ 타이머를 활용한 LED 제어

- 500ms 간격으로 LED 4개를 순차적으로 ON한 후 순차적으로 OFF
- 실습을 위한 Arduino MEGA ADK와 LED 결선 방법
 - Arduino MEGA ADK의 DIGITAL 5와 LED 모듈의 RED 포트의 핀 1을 연결
 - Arduino MEGA ADK의 DIGITAL 6과 LED 모듈의 RED 포트의 핀 2를 연결
 - Arduino MEGA ADK의 DIGITAL 7과 LED 모듈의 GREEN 포트의 핀 9를 연결
 - Arduino MEGA ADK의 DIGITAL 8과 LED 모듈의 GREEN 포트의 핀 10을 연결
 - Arduino MEGA ADK의 3.3V와 LED 모듈의 J4 포트의 +3.3V와 연결
 - Arduino MEGA ADK의 GND(OV)와 LED 모듈의 J4 포트의 좌측에 연결
- 결선 그림



실습 예제 (3)



❖ Arduino 라이브러리

▪ 사용자 라이브러리

- Arduino 사용자가 직접 생성한 라이브러리

▪ 표준 라이브러리

- Arduino.cc에서 제공하는 라이브러리로 Arduino IDE에 내장
- 표준 라이브러리 위치
 - Arduino IDE 실행 파일 폴더 내 'libraries'
 - arduino-1.8.5/libraries

이름	수정한 날짜	유형
Bridge	2018-02-08 오후...	파일 폴더
Esplora	2018-02-08 오후...	파일 폴더
Ethernet	2018-02-08 오후...	파일 폴더
Firmata	2018-02-08 오후...	파일 폴더
Keyboard	2018-02-08 오후...	파일 폴더
LiquidCrystal	2018-02-08 오후...	파일 폴더
Mouse	2018-02-08 오후...	파일 폴더

< 표준 라이브러리 위치 >

Arduino 라이브러리

Bridge

EEPROM

Esplora

Ethernet

Firmata

HID

Keyboard

LiquidCrystal

Mouse

Robot Control

Robot IR Remote

Robot Motor

SD

SPI

Servo

SoftwareSerial

SpacebrewYun

Ternboo

WiFi101

Wire

< 표준 라이브러리 >

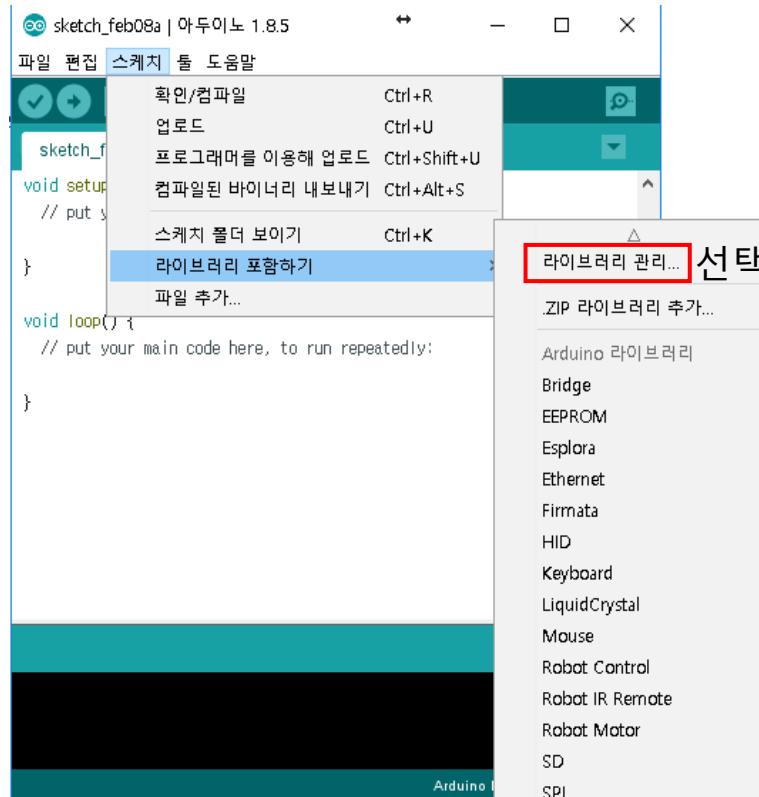
실습 예제 (3)



❖ Arduino 라이브러리

▪ Arduino 표준 라이브러리 추가

- 라이브러리 매니저를 통해 간편 설치 가능
- Arduino IDE 실행 – ‘스케치’ – ‘라이브러리 포함하기’ – ‘라이브러리 관리...’ 선택



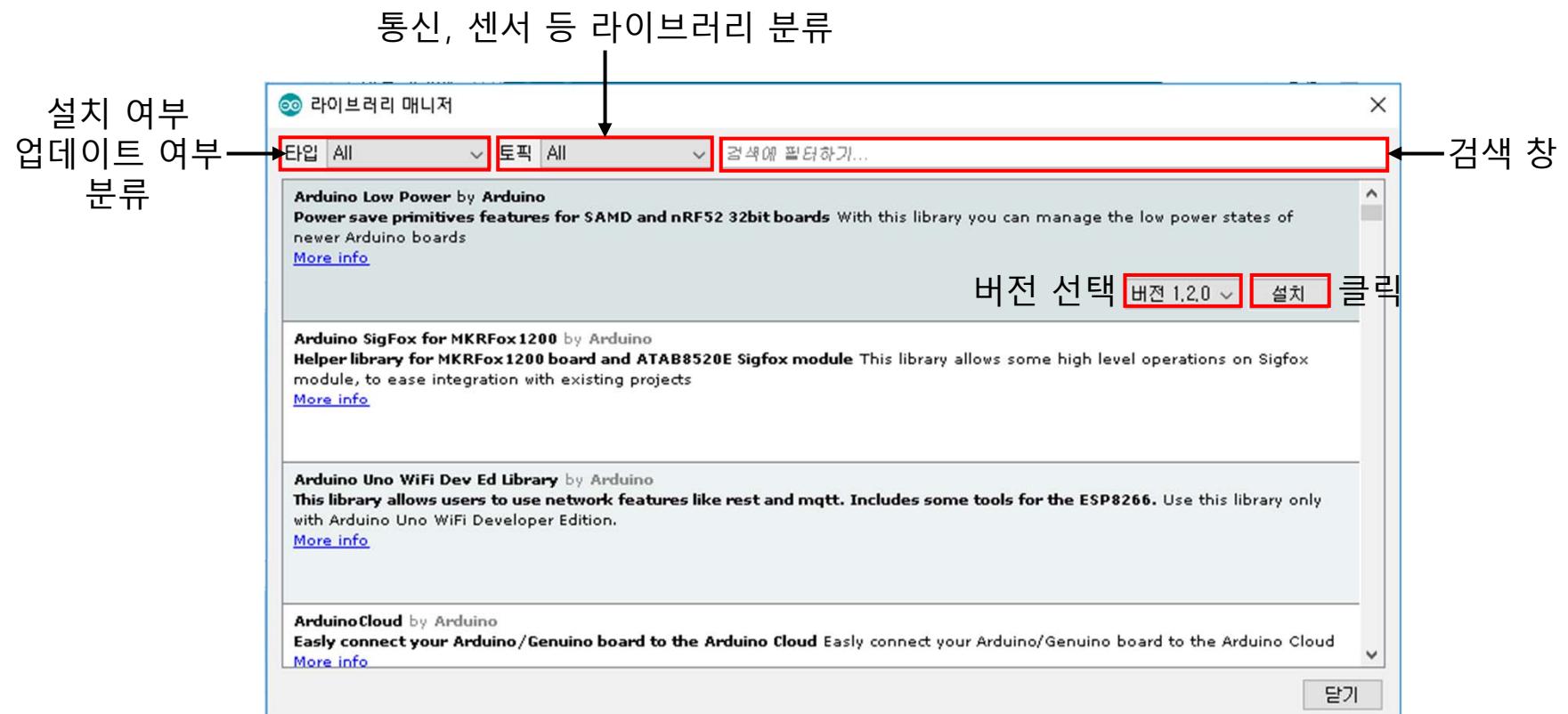
실습 예제 (3)



❖ Arduino 라이브러리

▪ Arduino 표준 라이브러리 추가

- 설치하고자 하는 라이브러리를 검색
- 검색된 라이브러리 클릭 시 버전 및 설치 버튼 활성화
- 버전을 선택하고 설치 버튼을 클릭하여 라이브러리 설치



실습 예제 (3)



❖ Arduino 라이브러리

▪ Arduino 사용자 라이브러리 추가

- Google 또는 Github 검색을 통해 다양한 Arduino 사용자 라이브러리를 찾을 수 있음
- 검색한 Arduino 사용자 라이브러리를 다운로드
- 사용자 라이브러리 추가 방법은 2가지가 존재
 - .ZIP 라이브러리 추가
 - 압축되지 않은 라이브러리 추가
- 다음 페이지에서 Timer 관련 라이브러리를 통해 자세하게 설명

실습 예제 (3)



❖ MsTimer2

- Arduino에서 타이머를 쉽게 사용하기 위한 라이브러리
- 다운로드 경로: <http://playground.arduino.cc/Main/MsTimer2>
 - 해당 URL 접속 후 중간 부분에 위치한 MsTimer2.zip 다운로드
 - 강의 홈페이지(<http://mesl.khu.ac.kr>)에서도 다운로드 가능

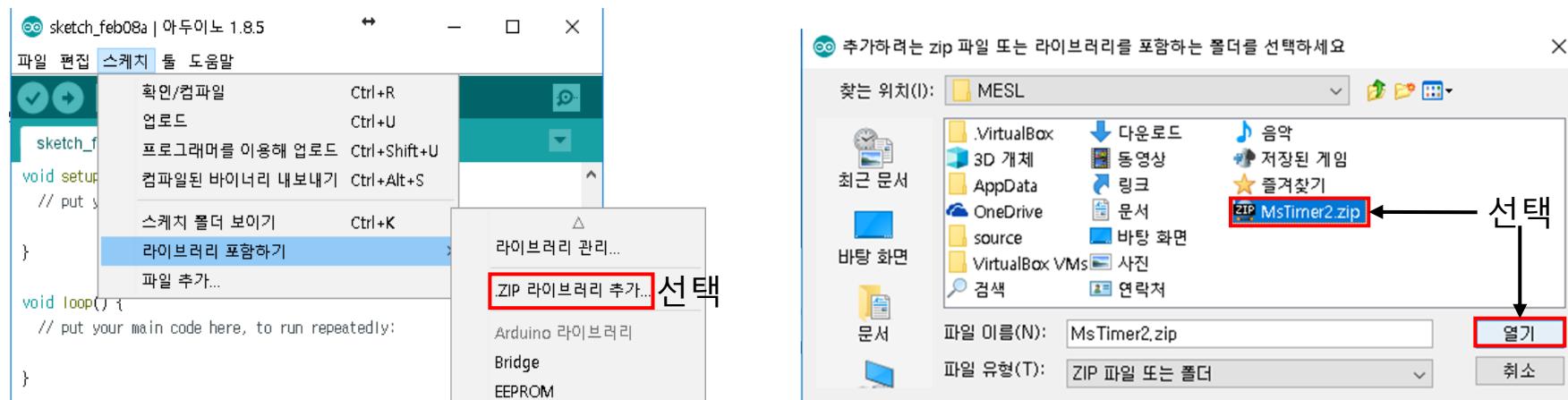
The screenshot shows a web browser window with the URL playground.arduino.cc/Main/MsTimer2 in the address bar. The page has a teal header with the Arduino logo. Below the header, the text "Source code" is displayed in orange. Underneath it, "License: LGPL" is shown. A red box highlights the link "MsTimer2.zip". To the right of the link, the text "선택" (Select) is written. At the bottom of the page, there is a note: "Install it on {arduino-path}/libraries/".

실습 예제 (3)



❖ MsTimer2

- 다운로드 받은 MsTimer2 라이브러리 사용을 위해 '라이브러리 추가' 진행
- .ZIP 라이브러리 추가 방법
 - Arduino IDE 실행 – '스케치' – '라이브러리 포함하기' – '.ZIP 라이브러리 추가 ...' 선택
 - 이후 MsTimer2.zip를 선택



실습 예제 (3)



❖ MsTimer2

▪ .ZIP 라이브러리 추가 방법

- MsTimer2.zip 열기 이후 '스케치' – '라이브러리 포함하기' 메뉴 확인
 - MStimer2 라이브러리가 추가되었음을 확인

```
sketch_feb09a | 아두이노 1.8.5
파일 편집 스케치 툴 도움말
sketch_feb09a
void setup() {
    // put your setup code here, to run once:
}

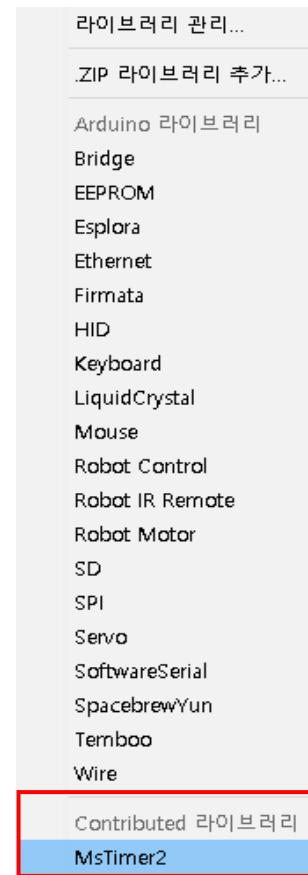
void loop() {
    // put your main code here, to run repeatedly:
}
```

확인

라이브러리가 추가되었습니다. "라이브러리 포함하기" 메뉴를 확인하세요

Arduino Mega ADK on COM4

< 라이브러리 추가 완료 화면 >



< 추가된 라이브러리 확인 화면 >

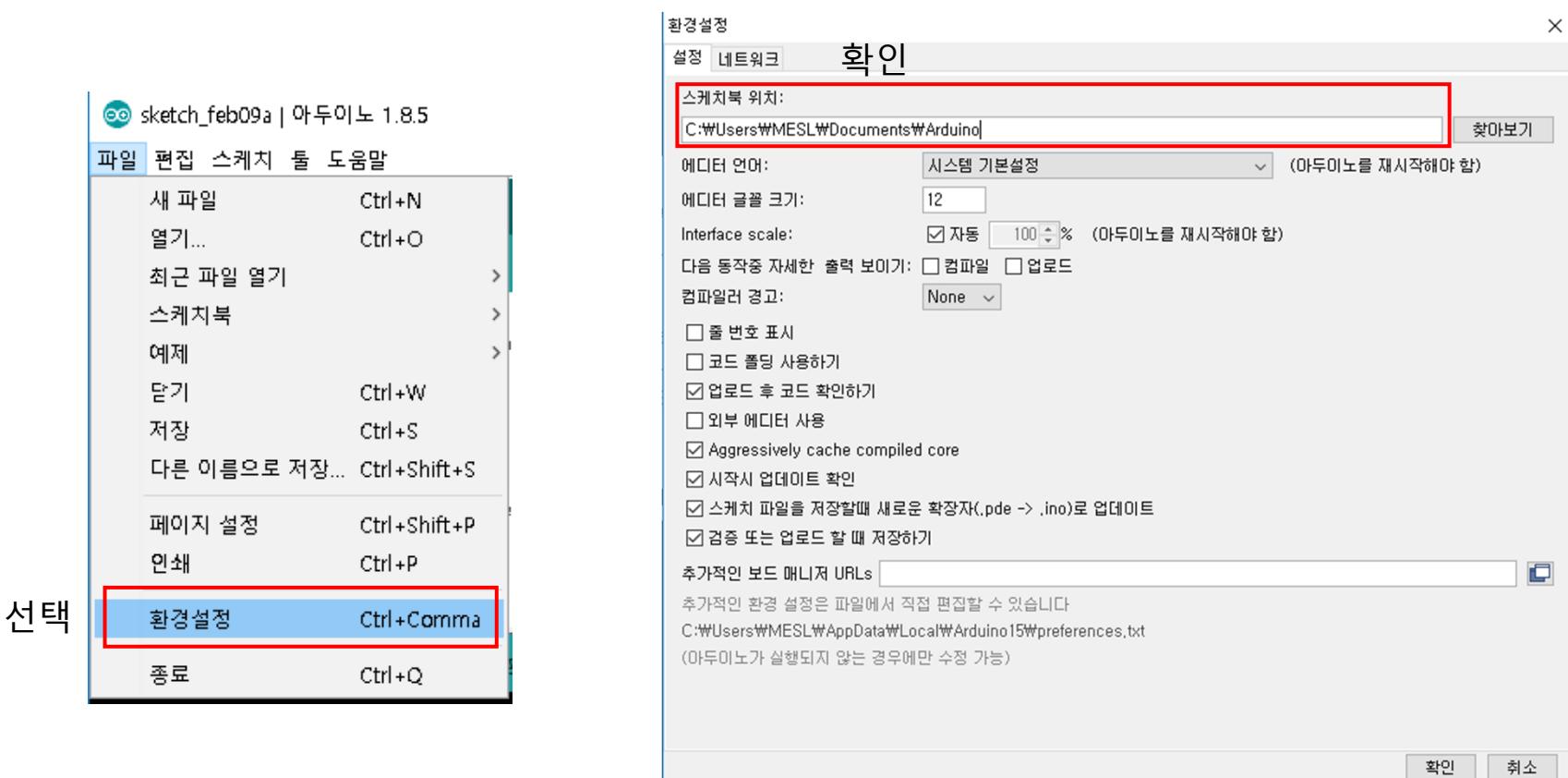
실습 예제 (3)



❖ MsTimer2

▪ .ZIP 라이브러리 추가 방법

- 라이브러리가 추가된 위치
 - Arduino IDE – ‘파일’ – ‘환경설정’ 선택
 - ‘스케치북 위치’의 폴더 주소를 찾아가면 libraries 폴더를 찾을 수 있음



실습 예제 (3)



❖ MsTimer2

▪ .ZIP 라이브러리 추가 방법

- 라이브러리가 추가된 위치

내 PC > 문서 > Arduino		
이름	수정한 날짜	유형
libraries	2018-02-09 오후...	파일 폴더

- Arduino 사용자 라이브러리는 libraries 폴더에 존재

내 PC > 문서 > Arduino > libraries		
이름	수정한 날짜	유형
MsTimer2	2018-02-09 오후...	파일 폴더
readme.txt	2017-09-29 오전...	텍스트 문서

- MsTimer2 라이브러리 구성요소

내 PC > 문서 > Arduino > libraries > MsTimer2		
이름	수정한 날짜	유형
examples	2018-02-09 오후...	파일 폴더
keywords.txt	2018-02-09 오후...	텍스트 문서
* MsTimer2.cpp	2018-02-09 오후...	C++ Source
MsTimer2.h	2018-02-09 오후...	C/C++ Header
README.rnd	2018-02-09 오후...	MD 파일

실습 예제 (3)



❖ MsTimer2

▪ 압축되지 않은 라이브러리 추가(참고)

- 다운로드 받은 'MsTimer2.zip'의 압축을 해제
- 앞서 확인한 '스케치북 위치' 아래 libraries에 압축 해제 된 MsTimer2 폴더를 복사(또는 이동)
- MsTimer2.zip 열기 이후 '스케치' – '라이브러리 포함하기' 메뉴 확인
 - 만약, MsTimer2가 보이지 않는다면 라이브러리 매니저를 한 번 실행한 후 다시 확인
 - 라이브러리 매니저 실행 방법: '스케치' – '라이브러리 포함하기' – '라이브러리 관리...' 선택

▪ MsTimer2는 3개의 함수로 구성되어 동작

- MsTimer2::set(millisecond, function)
 - 타이머 사용 시 필요한 타이머 시간 및 타이머 처리 함수를 설정
 - millisecond: 타이머 시간으로 설정된 시간 이후 타이머 처리 함수를 실행(단위: ms)
 - function: 설정된 타이머 시간 이후 실행되는 타이머 처리 함수
- MsTimer2::start()
 - 타이머 실행
- MsTimer2::stop()
 - 타이머 정지

실습 예제 (3)



❖ MsTimer2

▪ 라이브러리 사용 방법

- Arduino IDE 실행 – ‘스케치’ – ‘라이브러리 포함하기’ – ‘MsTimer2’ 선택
- 스케치 파일에 '#include <MsTimer2.h>'가 추가됨
 - 그림과 같은 과정을 거치지 않고 사용되는 헤더 파일을 직접 타이핑해서 사용해도 됨

The image consists of two side-by-side screenshots of the Arduino IDE. Both screenshots show a sketch named 'sketch_feb12a' with the following code:void setup() {
 // put your setup code here, to run once:
}

void loop() {
 // put your main code here, to run repeatedly:
}In the left screenshot, the 'Sketch' menu is open, and the 'Include Library...' option is highlighted. A dropdown menu shows various libraries, and 'MsTimer2' is visible at the bottom, with a red box around it and the word '선택' (Selected) written below it. In the right screenshot, the 'Sketch' menu is also open, but the 'Include Library...' option is now highlighted. The dropdown menu shows the same list of libraries, but 'MsTimer2' is now highlighted with a red box, and the word '확인' (Confirm) is written below it.

실습 예제 (3)



❖ 타이머를 활용한 LED 제어

▪ 스케치 작성

- lab1-2_3.ino

```
#include <MsTimer2.h>

int Led[4] = {5, 6, 7, 8}; // Arduino PIN 정의

void LedControl() {
    static int counter = 0;
    static byte output = HIGH; // LED ON/OFF를 결정할 변수

    digitalWrite(Led[counter++], output); // LED ON/OFF 수행
    if (counter > 3) {
        counter = 0; // 배열 인덱스 초기화
        output = !output; // 현재와 반대 상태로 전환
    }
}

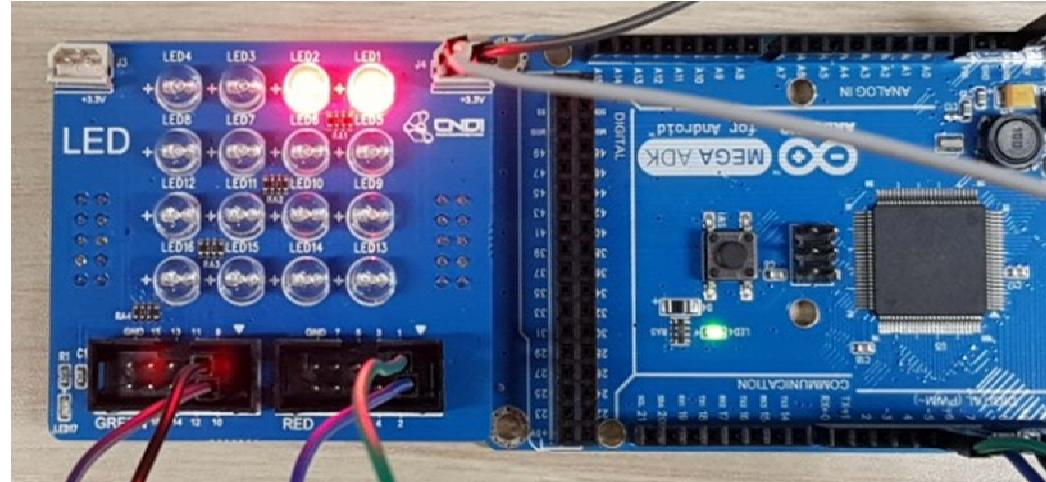
void setup() {
    int i;
    for (i=0; i<4; i++)
        pinMode(Led[i], OUTPUT); // 각 PIN을 출력으로 설정
    MsTimer2::set(500, LedControl); // Timer 설정
    MsTimer2::start(); // Timer 시작
}

void loop() {
```

실습 예제 (3)



- ❖ 타이머를 활용한 LED 제어
 - 동작 화면(일부)



실습 과제



- ❖ 1) RED/GREEN LED 번갈아 500ms 마다 ON/OFF
 - LED1 – LED9 – LED2 – LED10의 순서로 ON/OFF
- ❖ 2) RED/GREEN LED 쌍으로 500ms 마다 ON/OFF
 - LED1+LED9 – LED2+LED10의 순서로 ON/OFF
- ❖ 3) RED/GREEN LED ON/OFF 속도 조절
 - LED1 – LED9 – LED2 – LED10의 순서로 ON/OFF
 - LED1가 켜진 후부터 LED10이 꺼진 순간을 1회 수행으로 정의
 - 1회 수행 이후 지연 시간이 500ms씩 증가
 - 첫 번째 수행 시 지연 시간 500ms
 - 두 번째 수행 시 지연 시간 1000ms
 - 세 번째 수행 시 지연 시간 1500ms
 - 4번째 수행(지연 시간 2000ms) 이후 지연 시간 초기화
 - 지연 시간 500ms부터 다시 동작
- ❖ 4) delay()와 MsTimer2() 비교
 - lab1-2_2.ino와 lab1-2_3.ino 스케치 모두 LED 16을 1초마다 점등하는 코드 추가
 - 각 코드 작성의 이점 확인

실습 과제



- ❖ 5) 'is-3.arduino.pdf'의 p.15 실습
- ❖ 6) 실습과제 5)의 방법으로 실습예제 2) 수행

참고



❖ 시리얼 모니터를 이용한 디버깅 메시지 출력

```
void setup() {
    // 시리얼 모니터 사용
    Serial.begin( 9600);

    Serial.println("Serial monitor messages...");
}

void loop() {
    char input;
    Serial.print('A');
    Serial.println(100);
    Serial.print(99.999);

    Serial.print("Debug: ");
    Serial.println("Here");

    while (1) {
        if (Serial.available() > 0) {
            input = Serial.read();
            Serial.print(input);
        }
    }
}
```



Q & A



<http://mesl.khu.ac.kr>