

소프트웨어 프로젝트 2

아두이노 기초 (2)

2024년 2학기

국민대학교
소프트웨어학부/인공지능학부
주용수, 최진우, 한재섭, 허대영
{ysjoo, jaeseob, jnwochoi, dyheo}@kookmin.ac.kr

아두이노 IDE 설치

- 다운로드: Legacy IDE (1.8.X) 설치
 - 주의: 2.x 설치시 추후 plotter를 통한 프로젝트 평가에 어려움이 있음)
 - <https://www.arduino.cc/en/software>
- 설치 동영상 참조
 - 운영체제별 설치 영상: <https://youtu.be/4ygr8vtUmal>
 - 버전, 운영체제 에 따라 한글 입력(C&P) 문제 있음
 - Linux의 경우 한글 폰트 설정, root 권한 등 필요

Legacy IDE (1.8.X)



 **Arduino IDE 1.8.19**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Arduino IDE 1.x documentation](#) for installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file
Windows app Win 8.1 or 10 

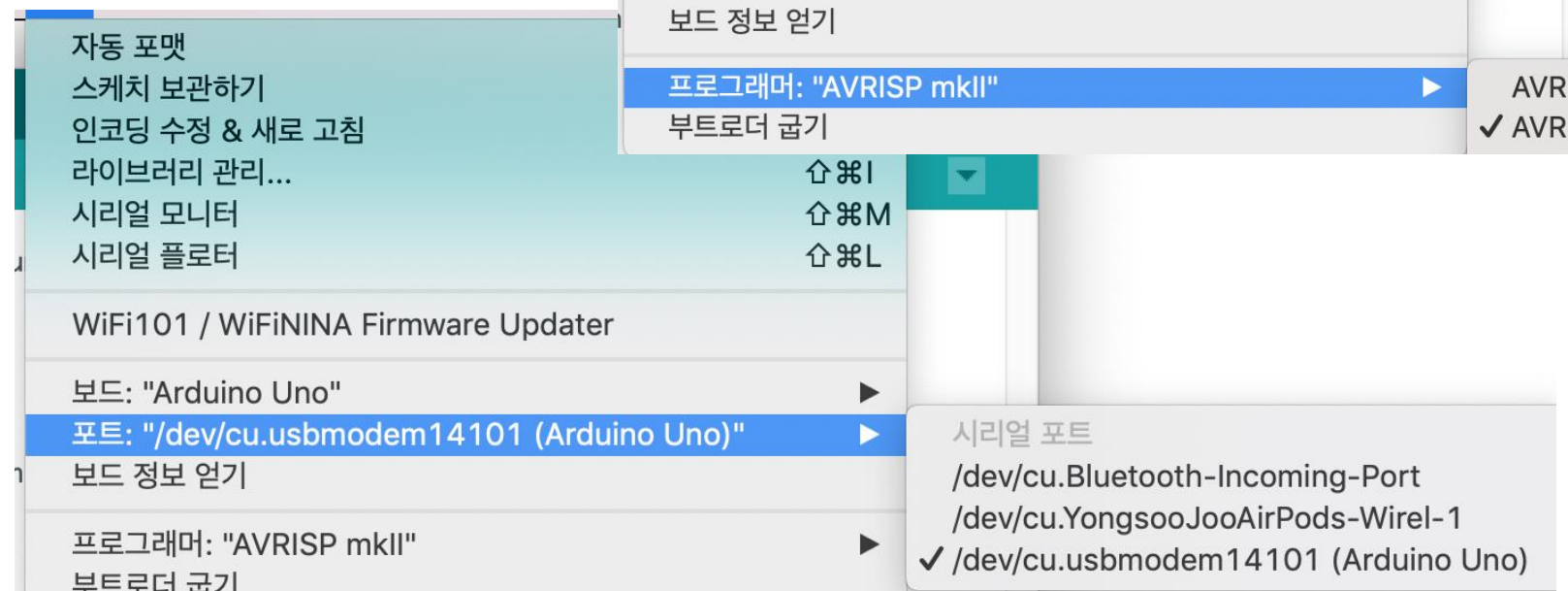
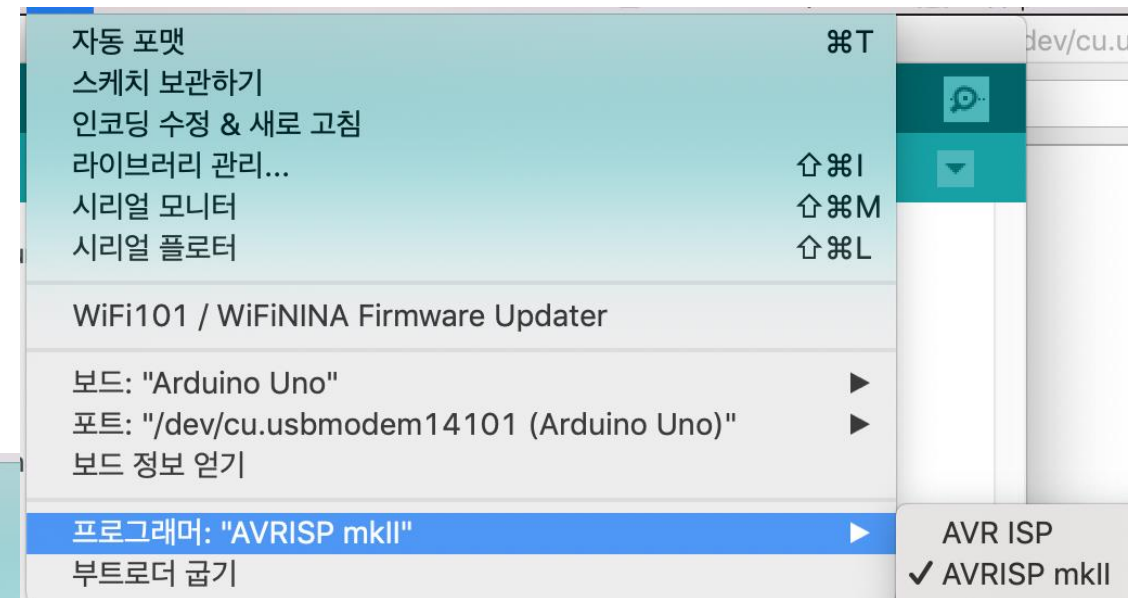
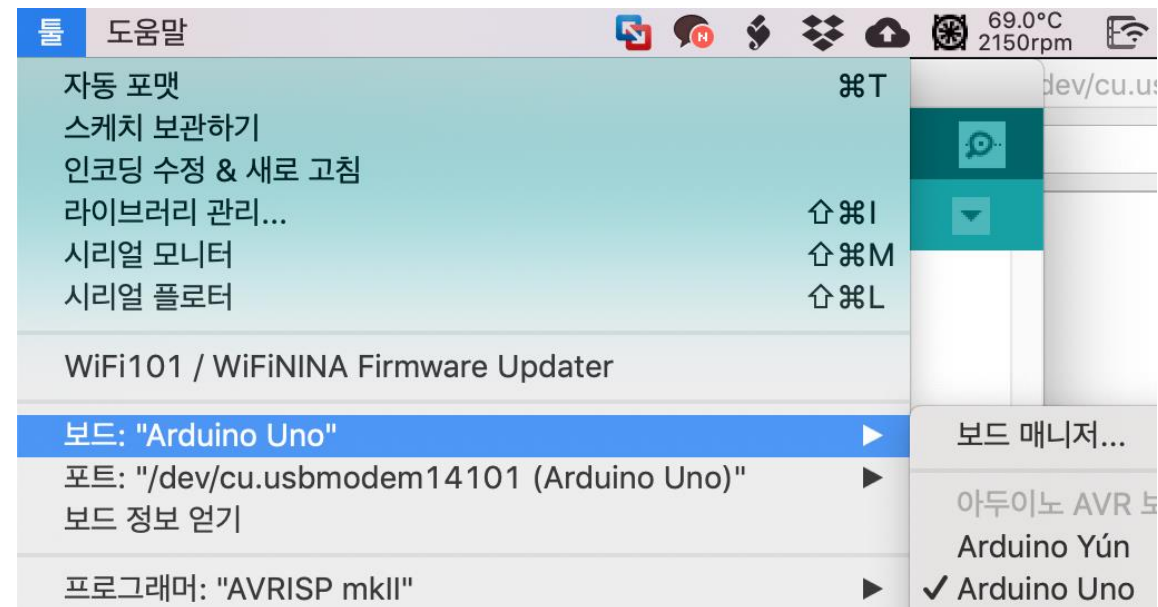
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)
[Checksums \(sha512\)](#)

아두이노 USB 연결 및 설정 확인

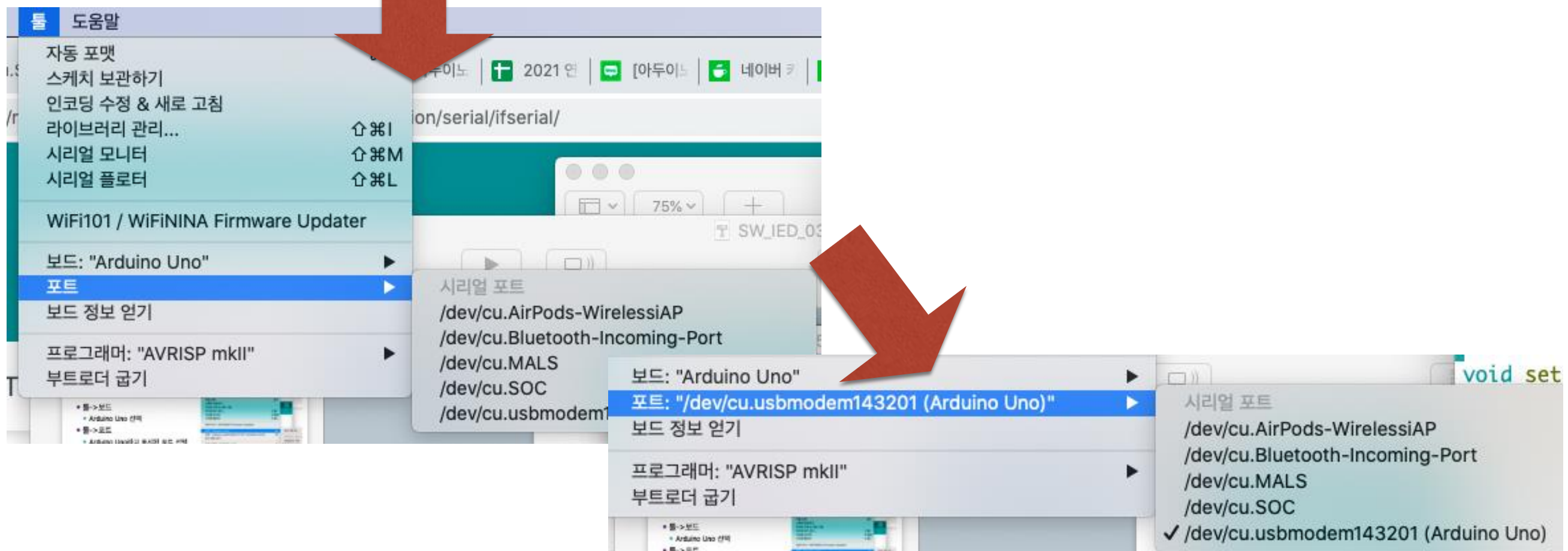
- 아두이노와 컴퓨터를 USB 케이블로 연결 (USB 전원만으로 구동 가능)
 - 아두이노 아랫 면이 전도성 물체에 접촉되지 않도록 유의 (플라스틱 케이스 사용)
- 툴->보드
 - Arduino Uno 선택
- 툴->포트 (OS 마다 다르게 표시됨)
 - Arduino Uno라고 표시된 포트 선택
- 툴->프로그래머
 - AVRISP mkII 선택



아두이노 USB 연결 및 설정 확인

- 포트 미설정 오류 예
 - 툴->포트에서 Arduino 포트 선택되어 있는지 확인

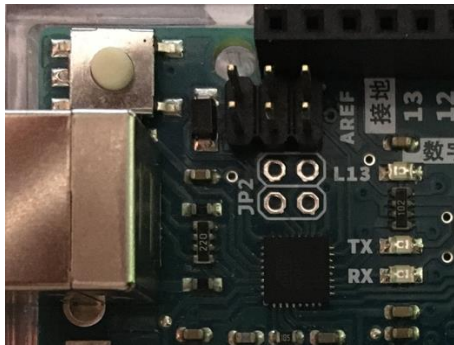
```
스케치를 업로드 하는 동안 에러가 발생하였습니다. 오류 메시지 복사
전역 변수는 동적 메모리 200바이트(9%)를 사용, 1848바이트의 지역변수가 남음. 최대는 2048 바이트.
스케치를 업로드 하는 동안 에러가 발생하였습니다.
avrdude: ser_open(): can't open device "/dev/cu.usbmodem14601": No such file or directory
12 Arduino Uno on /dev/cu.usbmodem14601
```



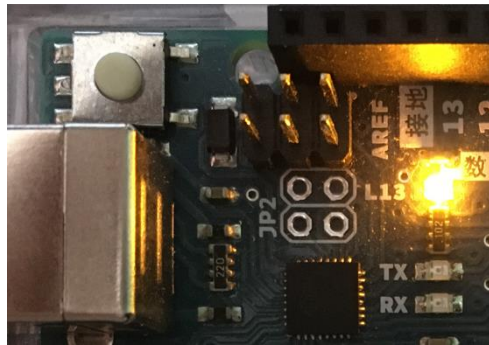
컴파일 및 업로드 테스트

- 작업 폴더 생성
 - 예: ~/IED_src/
- 우측 하단 예제 코드를 입력 후 “04_example_1”로 저장
 - ~/IED_src/04_example_1/04_example_1.ino 로 저장됨
- 좌측 상단 두번째 버튼을 눌러 컴파일 및 업로드
- PIN_LED 출력 0, 1에 대해 동작 확인

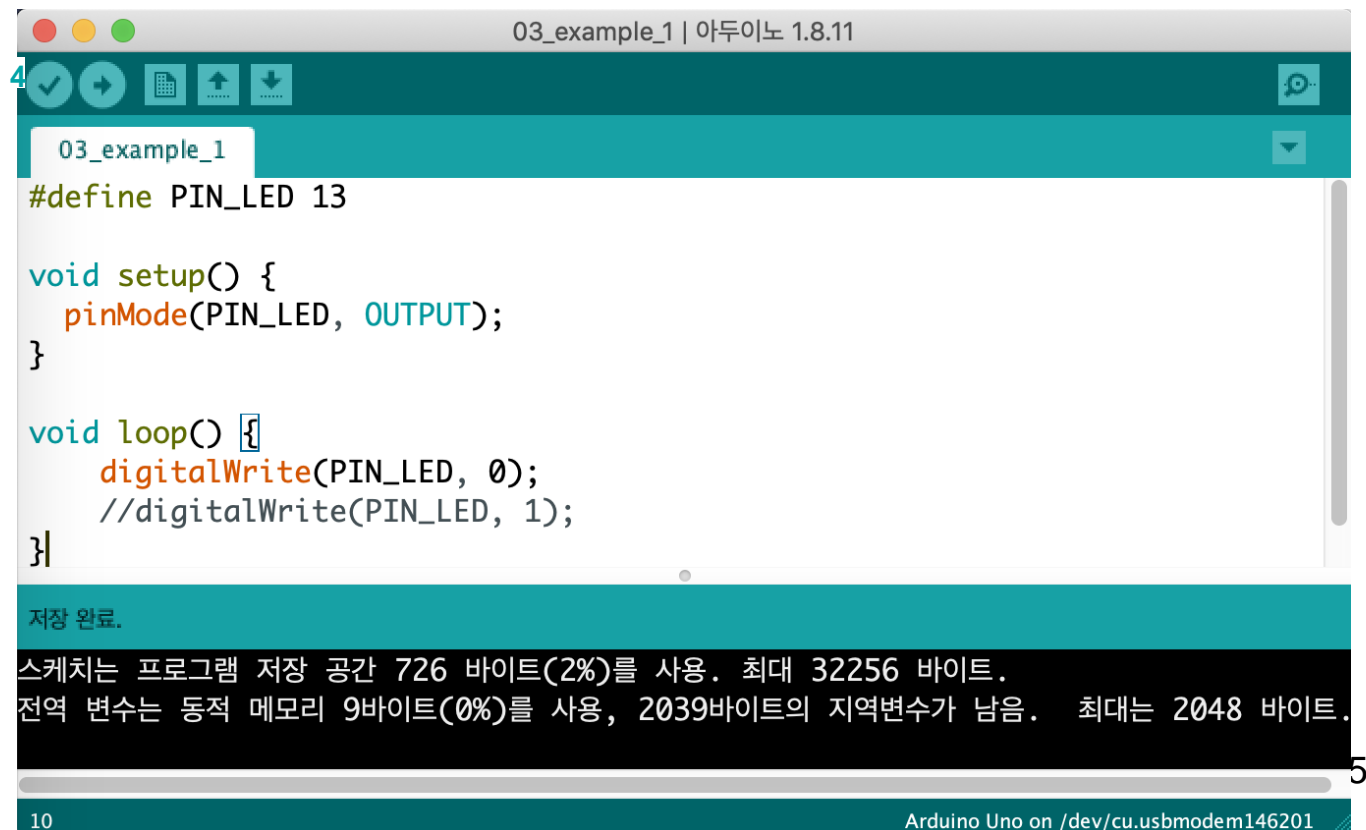
4



출력 0



출력 1



```
03_example_1 | 아두이노 1.8.11
4
03_example_1
#define PIN_LED 13

void setup() {
  pinMode(PIN_LED, OUTPUT);
}

void loop() {
  digitalWrite(PIN_LED, 0);
  //digitalWrite(PIN_LED, 1);
}

저장 완료.
스케치는 프로그램 저장 공간 726 바이트(2%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 9바이트(0%)를 사용, 2039바이트의 지역변수가 남음. 최대는 2048 바이트.
```

10 Arduino Uno on /dev/cu.usbmodem146201

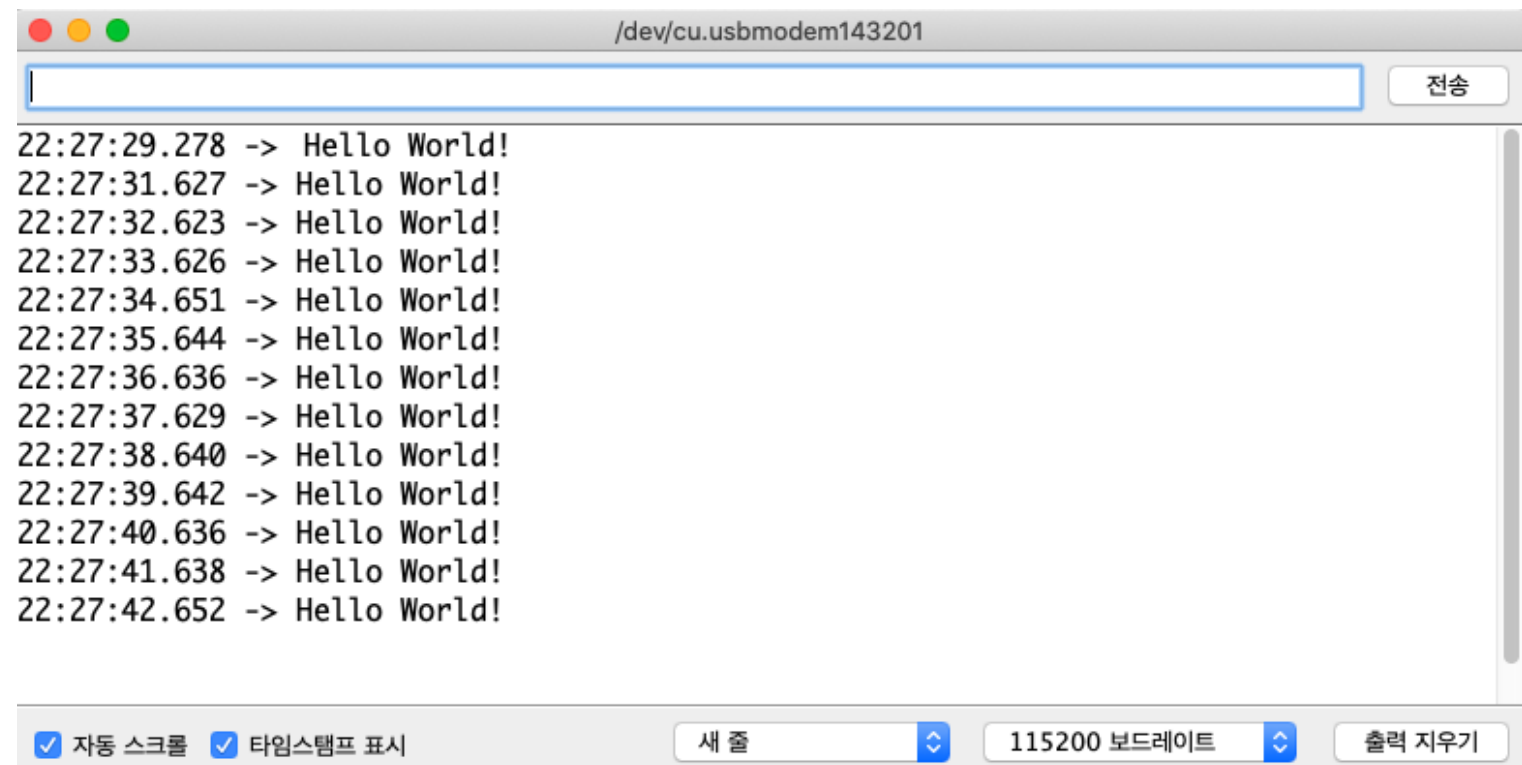
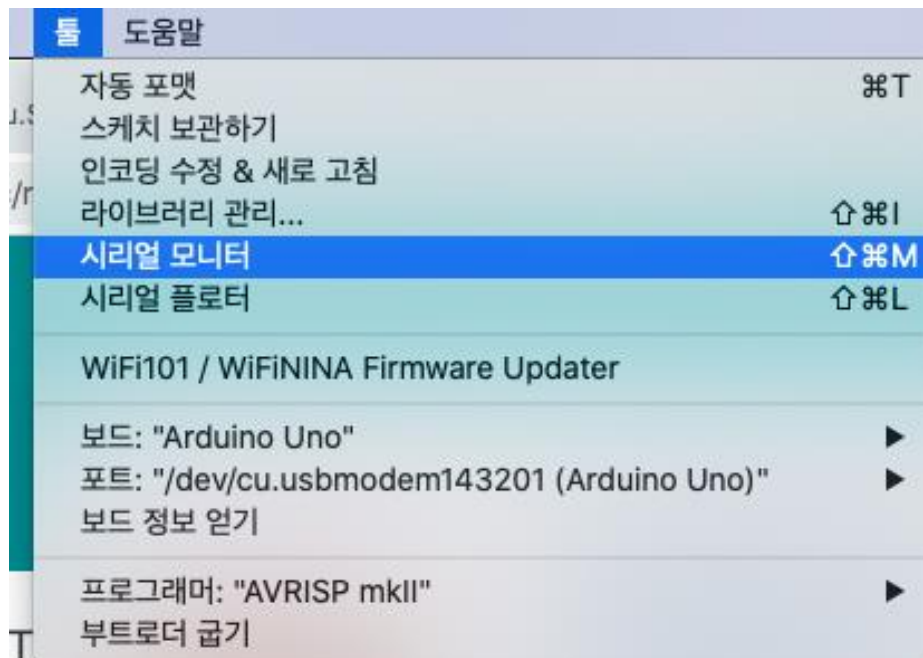
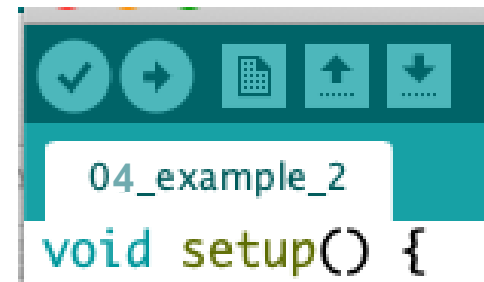
시리얼 모니터 테스트

- 예제코드 설명: 04_example_2.ino (코드는 직접 입력하세요)
 - Line 3: serial 포트 초기화
 - Line 4-6: serial 포트 초기화 완료까지 대기 (Uno 버전은 불필요)
 - <https://www.arduino.cc/reference/en/language/functions/communication/serial/ifserial/>
 - Line 10: serial 포트에 메시지 출력 후 줄바꿈 (println)
 - Line 11: 1000 ms, 즉 1초간 대기

```
1 void setup() {
2   //Initialize serial and wait for port to open:
3   Serial.begin(115200);
4   while (!Serial) {
5     ; // wait for serial port to connect. Needed for native USB
6   }
7 }
8
9 void loop() {
10  Serial.println("Hello World!");
11  delay(1000);
12 }
```

시리얼 모니터 테스트

- 테스트 절차
 - 컴파일 및 업로드
 - 보드 위의 TX LED 점멸 확인
 - 툴->시리얼 모니터 선택
 - 메시지가 약 1초 간격으로 출력됨을 확인



LED 점멸제어 테스트

- 04_example_3.ino (코드는 직접 입력하세요)

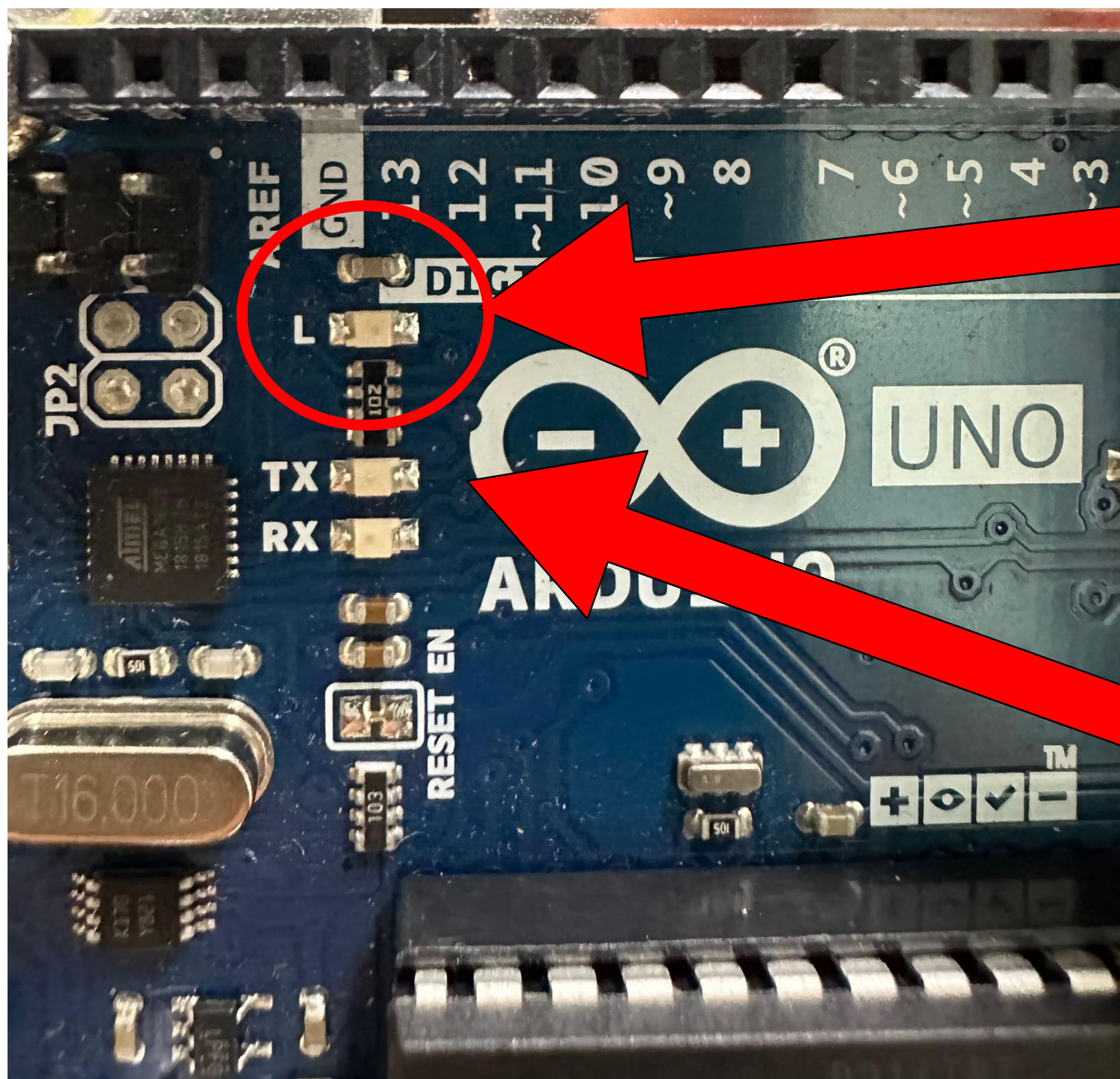
- 코드 설명은 아두이노 기초
Part I 슬라이드 참조

- 실습 과제

- **옆 코드는 에러를 여럿 포함하고 있음
(수정해서 정상 실행시켜야 함)**

- 컴파일 에러 디버깅
- 1초마다 LED가 켜졌다 꺼졌다
하는 동작을 반복하도록 코드 완성

```
1 #define PIN_LED 13
2 unsigned int count, toggle;
3
4 void setup() {
5     pinMode(PIN_LED, OUTPUT)
6     Serial.begin(115200); // Initialize serial port
7     while (!Serial) {
8         ; // wait for serial port to connect.
9     }
10    Serial.println("Hello World!");
11    count = toggle = 0;
12    digitalWrite(PIN_LED, toggle); // turn off LED.
13 }
14
15 void loop() {
16     Serial.println(++count);
17     toggle = toggle_state(toggle); toggle LED value.
18     digitalWrite(PIN_LED, toggle); // update LED status.
19     delay(1000); // wait for 1,000 milliseconds
20 }
21
22 int toggle_state(int toggle) {
23     return toggle;
24 }
```

이 'L' 을
깜박이는 것이
목표

‘TX’는
알아서 깜박임.
(Serial.println()
때문에)

LED 점멸제어 테스트

- 제출방법
 - 완성한 코드를 각자 github에 업로드하고 작동 영상 촬영
 - GitHub 사용법 문의: 각 분반 헬퍼 (slack)
 - 제출물
 - 제출 소스 코드의 github URL
(파일명은 04_example_3.ino 그대로 사용)
 - Youtube 링크 (영상 설정은 일부공개)
 - 제출 마감: 4주 2강 수업 전까지
- 과제 제출 설문지 링크
 - 수업 홈페이지 참조
 - 과제코드: 04P9