



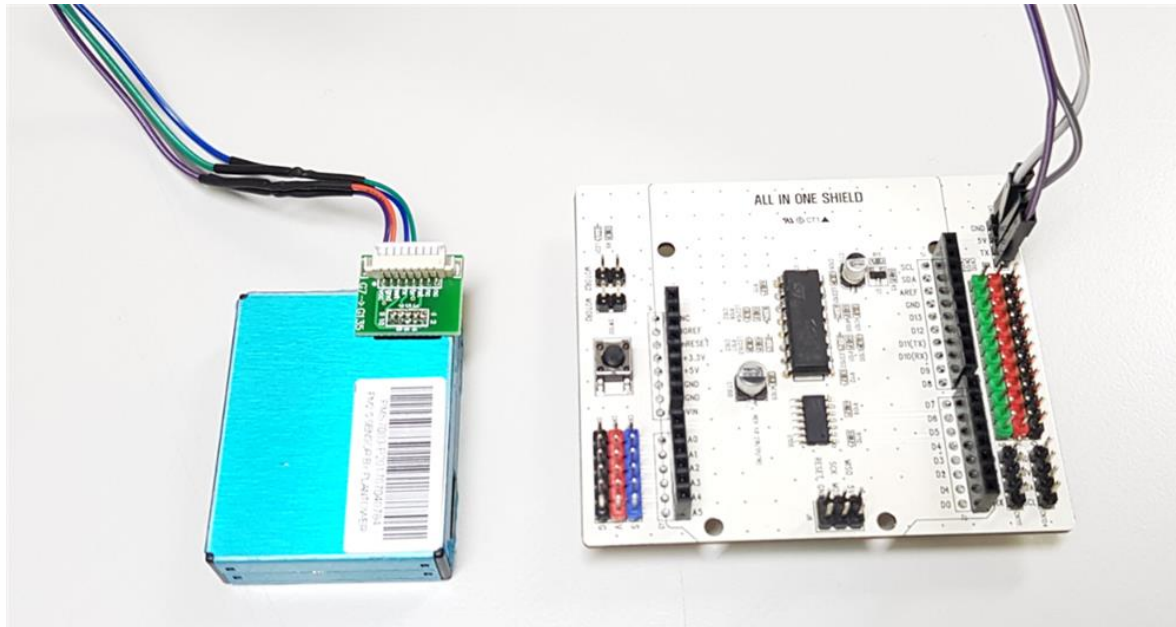
12

IoT 응용 실습

- 12-1 초미세먼지 센서 연결하기
- 12-2 미세먼지 센서값 측정하기
- 12-3 스마트팜 제어 실습

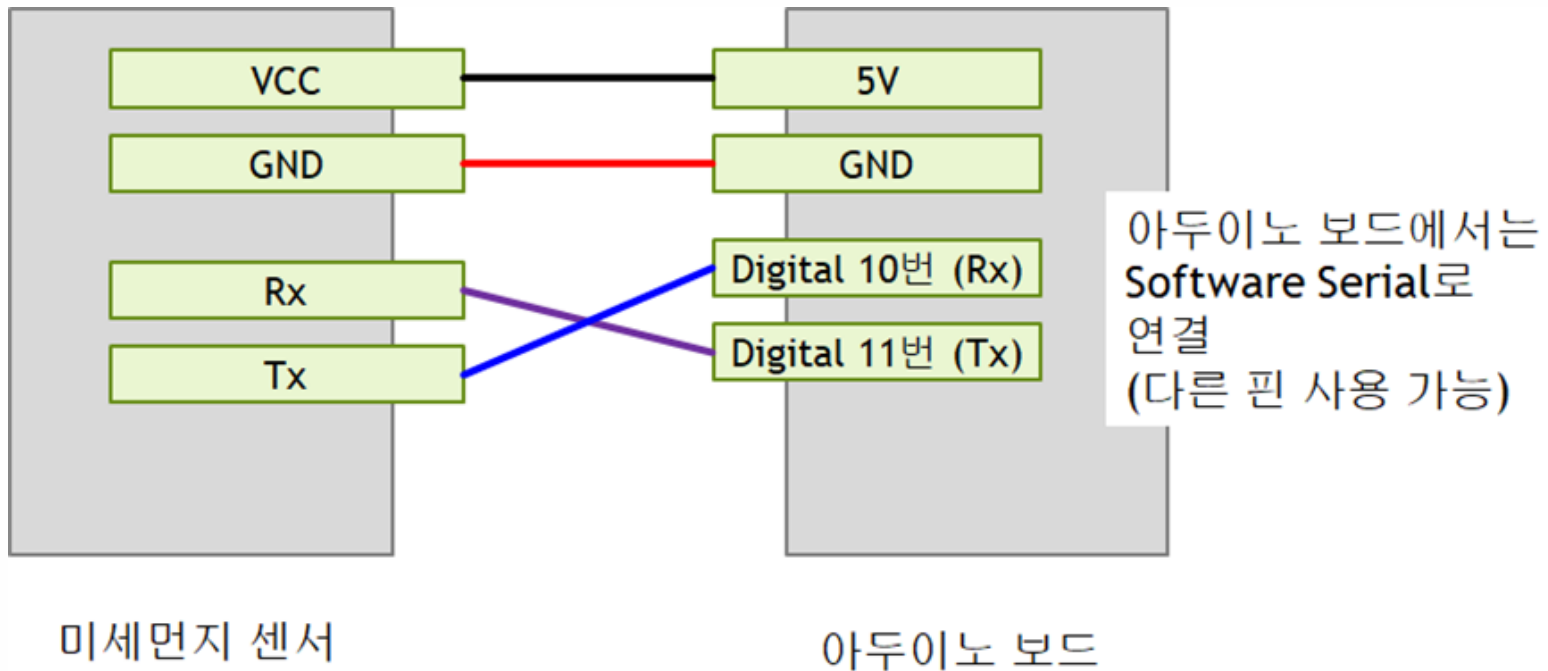


1. 초미세먼지 측정 센서 준비하기

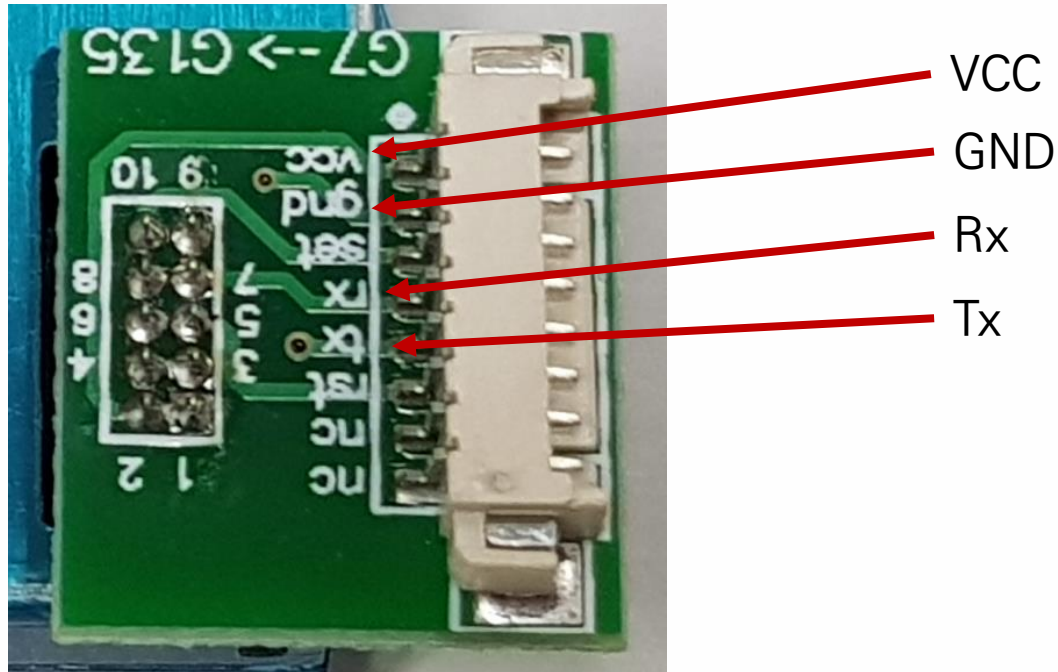


1마이크론, 2.5마이크론, 10마이크론 측정 센서

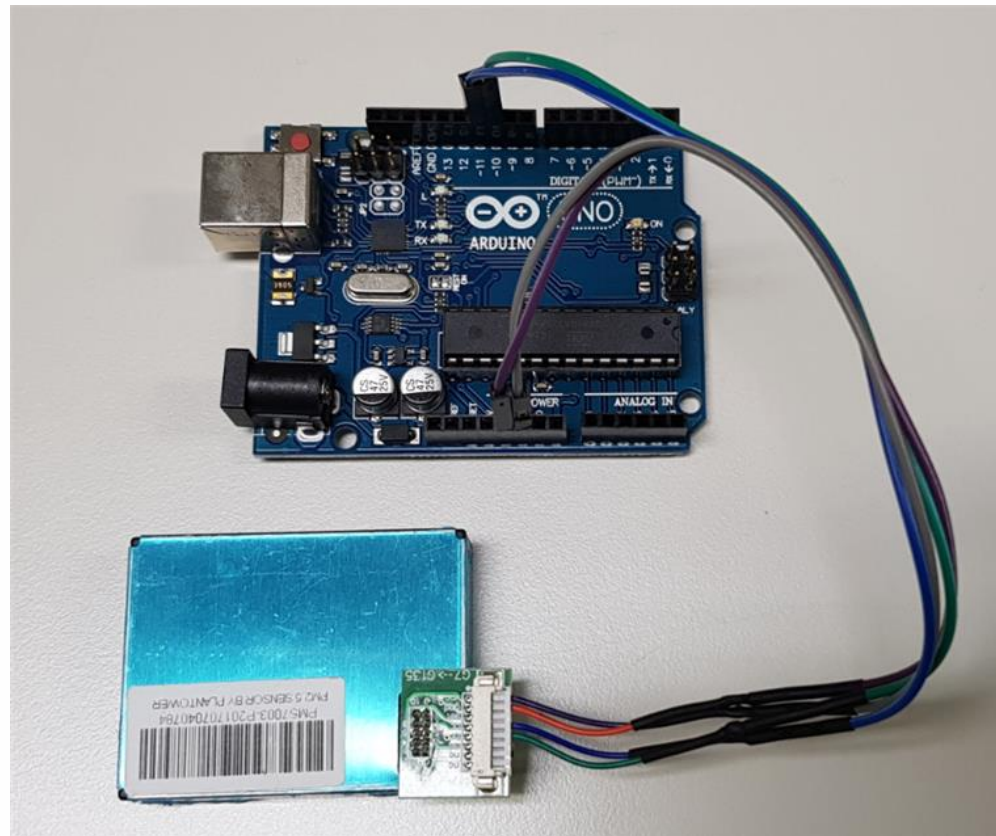
2. 시리얼 4핀 케이블을 이용한 센서 연결



2. 시리얼 4핀 케이블을 이용한 센서 연결



2. 시리얼 4핀 케이블을 이용한 센서 연결



1. 기본 측정 코드 (콘솔창 표시 예제)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
#include <SoftwareSerial.h>

//미세먼지 센서의 Tx는 10번, Rx는 11번에 연결합니다.
SoftwareSerial Serial1(10, 11);

//컬러 LED바는 디지털 2번에 연결합니다.
//LCD는 I2C 핀에 연결합니다.
//선풍기는 디지털 3번에 연결합니다.
//LED 소자는 디지털 13번에 연결합니다.

int incomingByte = 0;

const int MAX_FRAME_LEN = 64;
char frameBuf[MAX_FRAME_LEN];
int detectOff = 0;
int frameLen = MAX_FRAME_LEN;
bool inFrame = false;
char printbuf[256];

const bool DEBUG = false;

uint16_t calcChecksum = 0;
```

```
struct PMS7003_framestruct {
    uint8_t frameHeader[2];
    uint16_t frameLen = MAX_FRAME_LEN;
    uint16_t concPM1_0_CF1;
    uint16_t concPM2_5_CF1;
    uint16_t concPM10_0_CF1;
    uint16_t concPM1_0_amb;
    uint16_t concPM2_5_amb;
    uint16_t concPM10_0_amb;
    uint16_t rawGt0_3um;
    uint16_t rawGt0_5um;
    uint16_t rawGt1_0um;
    uint16_t rawGt2_5um;
    uint16_t rawGt5_0um;
    uint16_t rawGt10_0um;
    uint8_t version;
    uint8_t errorCode;
    uint16_t checksum;
} thisFrame;
```


1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
void setup() {  
  Serial.begin(115200);  
  delay(1000);  
  Serial.println("-- Initializing...");  
  
  LcdBackLight();  
}
```

```
bool pms7003_read() {  
  
  Serial.println("-- Reading PMS7003");  
  Serial1.begin(9600);  
  bool packetReceived = false;  
  while (!packetReceived) {  
    if (Serial1.available() > 32) {  
      int drain = Serial1.available();  
      if (DEBUG) {  
        Serial.print("-- Draining buffer: ");  
        Serial.println(Serial1.available(), DEC);  
      }  
      for (int i = drain; i > 0; i--) {  
        Serial1.read();  
      }  
    }  
    if (Serial1.available() > 0) {  
      if (DEBUG) {  
        Serial.print("-- Available: ");  
        Serial.println(Serial1.available(), DEC);  
      }  
    }  
  }  
}
```

1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
incomingByte = Serial1.read();
if (DEBUG) {
  Serial.print("-- READ: ");
  Serial.println(incomingByte, HEX);
}
if (!inFrame) {
  if (incomingByte == 0x42 && detectOff == 0) {
    frameBuf[detectOff] = incomingByte;
    thisFrame.frameHeader[0] = incomingByte;
    calcChecksum = incomingByte;
    detectOff++;
  }
  else if (incomingByte == 0x4D && detectOff == 1) {
    frameBuf[detectOff] = incomingByte;
    thisFrame.frameHeader[1] = incomingByte;
    calcChecksum += incomingByte;
    inFrame = true;
    detectOff++;
  }
  else {
    Serial.print("-- Frame syncing... ");
    Serial.print(incomingByte, HEX);
    if (DEBUG) {
      Serial.println();
    }
  }
}
}
```

```
else {
  frameBuf[detectOff] = incomingByte;
  calcChecksum += incomingByte;
  detectOff++;
  uint16_t val = frameBuf[detectOff-1]+(frameBuf[detectOff-2]<<8);
  switch (detectOff) {
    case 4:
      thisFrame.frameLen = val;
      frameLen = val + detectOff;
      break;
    case 6:
      thisFrame.concPM1_0_CF1 = val;
      break;
    case 8:
      thisFrame.concPM2_5_CF1 = val;
      break;
    case 10:
      thisFrame.concPM10_0_CF1 = val;
      break;
  }
}
```


1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
case 12:
    thisFrame.concPM1_0_amb = val;
    break;
case 14:
    thisFrame.concPM2_5_amb = val;
    break;
case 16:
    thisFrame.concPM10_0_amb = val;
    break;
case 18:
    thisFrame.rawGt0_3um = val;
    break;
case 20:
    thisFrame.rawGt0_5um = val;
    break;
case 22:
    thisFrame.rawGt1_0um = val;
    break;
case 24:
    thisFrame.rawGt2_5um = val;
    break;
case 26:
    thisFrame.rawGt5_0um = val;
    break;
case 28:
    thisFrame.rawGt10_0um = val;
    break;
```

```
case 29:
    val = frameBuf[detectOff-1];
    thisFrame.version = val;
    break;
case 30:
    val = frameBuf[detectOff-1];
    thisFrame.errorCode = val;
    break;
case 32:
    thisFrame.checksum = val;
    calcChecksum -= ((val)>>8)+(val&0xFF));
    break;
default:
    break;
}
```

1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
if (detectOff >= frameLen) {
    sprintf(printbuf, "PMS7003 ");
    sprintf(printbuf, "%s[%02x %02x] (%04x) ", printbuf,
        thisFrame.frameHeader[0], thisFrame.frameHeader[1], thisFrame.frameLen);
    sprintf(printbuf, "%sCF1=[%04x %04x %04x] ", printbuf,
        thisFrame.concPM1_0_CF1, thisFrame.concPM2_5_CF1, thisFrame.concPM10_0_CF1);
    sprintf(printbuf, "%samb=[%04x %04x %04x] ", printbuf,
        thisFrame.concPM1_0_amb, thisFrame.concPM2_5_amb, thisFrame.concPM10_0_amb);
    sprintf(printbuf, "%sraw=[%04x %04x %04x %04x %04x %04x] ", printbuf,
        thisFrame.rawGt0_3um, thisFrame.rawGt0_5um, thisFrame.rawGt1_0um,
        thisFrame.rawGt2_5um, thisFrame.rawGt5_0um, thisFrame.rawGt10_0um);
    sprintf(printbuf, "%sver=%02x err=%02x ", printbuf,
        thisFrame.version, thisFrame.errorCode);
    sprintf(printbuf, "%scsum=%04x %s xsum=%04x", printbuf,
        thisFrame.checksum, (calcChecksum == thisFrame.checksum ? "==" : "!="), calcChecksum);
```

1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
//미세먼지 1.0마이크론 값은 아래 변수에 저장됩니다.  
//thisFrame.concPM1_0_CF1  
  
//미세먼지 2.5마이크론 값은 아래 변수에 저장됩니다.  
//thisFrame.concPM2_5_CF1  
  
//미세먼지 10.0마이크론 값은 아래 변수에 저장됩니다.  
//thisFrame.concPM10_0_CF1  
  
//시리얼 모니터링 창에 값을 출력합니다.  
Serial.print("PM1.0:");  
Serial.print(thisFrame.concPM1_0_CF1, DEC);  
Serial.print(" PM2.5:");  
Serial.print(thisFrame.concPM2_5_CF1, DEC);  
Serial.print(" PM10:");  
Serial.println(thisFrame.concPM10_0_CF1, DEC);
```

1. 기본 측정 코드 (콘솔창 표시 예제 - 계속)

1마이크론, 2.5마이크론, 10마이크론 크기를 동시에 측정하여 콘솔창에 표시

```
packetReceived = true;
    detectOff = 0;
    inFrame = false;
    }
    }
    }
    Serial1.end();
    return (calcChecksum == thisFrame.checksum);
}

void loop () {
    if (!pms7003_read()) {
        delay(4000);
    }

    //5초마다 측정합니다.
    delay(5000);
}
```

(최종 코드)

<http://www.helloapps.co.kr/download/>

초미세먼지 측정 센서

✓ 초미세먼지 측정 센서 활용 가이드.hwp

2. 기타 시나리오 소스 코드 참고

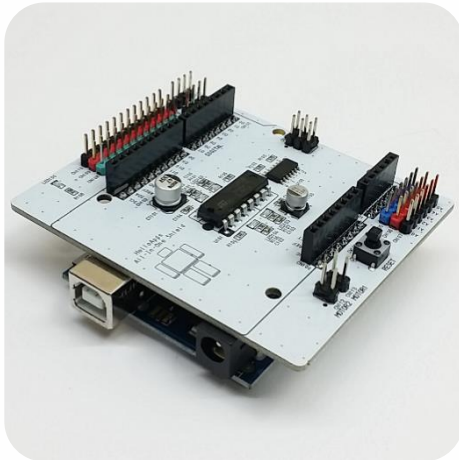
- 측정 값 콘솔창에 출력하기
- 측정 값 LCD에 출력하기
- 컬러 LED바로 상태 표시하기
- DC 팬 제어하기

<http://www.helloapps.co.kr/download/>

초미세먼지 측정 센서

- ✓ 초미세먼지 측정 센서 활용 가이드.hwp

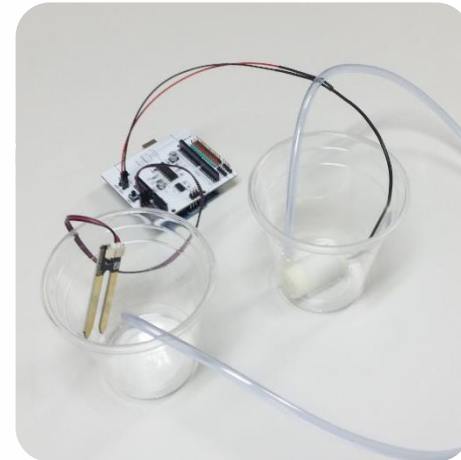
1. 부품 연결하기



아두이노 우노 보드 및
올인원 센서셴드



토양습도 센서 및
물공급용 모터 펌프



토양습도 센서 및
물공급용 모터 펌프

토양 습도를 측정하고 물을 공급할 수 있는 모듈

모터 펌프를 바로 연결할 수 있는 모터 구동 회로 및 전용 핀 내장

1. 부품 연결하기



모터펌프는 MOTOR1에 연결

토양습도 센서는 아날로그 0번에 연결

2. 토양습도 센서값 측정하기

```
void setup()
{
}

void loop()
{
    a = AnalogRead(0)
    PrintLine(a)
    Delay(100)
}
```



토양 습도 센서는 흙이 건조할 수록 값이 작고, 수분이 높으면 값이 커집니다.

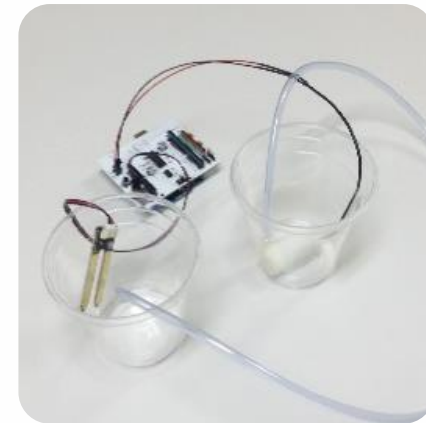
3. 모터 펌프 작동시키기

```
void setup()
{

}

void loop()
{
    //작동
    AnalogWrite(5, 255)
    Delay(3000)

    //중지
    AnalogWrite(5, 0)
    Delay(3000)
}
```



AnalogWrite(5, 255) 명령어
모터 작동
(PWM 방식입니다)

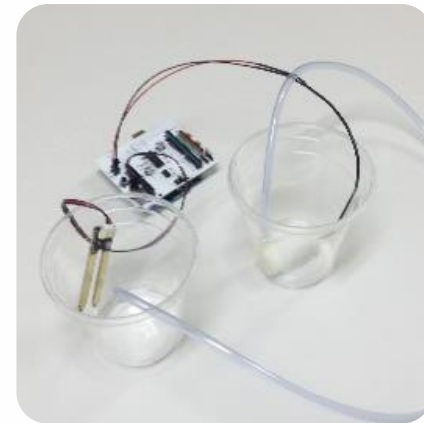
작동시 건전지 홀더를 연결하고
작동시키면 모터가 더 강하게
작동합니다.

4. 자동으로 모터 펌프 작동시키기

```
void loop()
{
    a = AnalogRead(0)

    if (a < 300)
        AnalogWrite(5, 255)
    else
        AnalogWrite(5, 0)

    Delay(100)
}
```



토양습도 센서를 컵 밖으로 꺼내면
모터 펌프가 작동

건조하면 모터 펌프를 자동으로
작동시킴

5. 버튼으로 모터 펌프 3초간 작동시키기

```
void loop()
{
    d = DigitalRead(2)

    if (d == HIGH)
    {
        AnalogWrite(5, 255)
        Delay(3000)

        AnalogWrite(5, 0)
    }

    Delay(100)
}
```

6. 스마트폰 블루투스로 펌프 제어하기 (3초간 작동시키기)

```
void loop()
{
    s = SerialRead()
    char c = (char)s

    if (s == 1 || c == '1')
    {
        AnalogWrite(5, 255)
        Delay(3000)

        AnalogWrite(5, 0)
    }

    Delay(100)
}
```


7. 프로젝트 실습

- 1) LCD에 작동 상태를 표시하시오
- 2) 컬러 LED바로 식물이 잘 자라도록 초록색 조명을 제공하시오
- 3) WiFi 보드 또는 블루투스를 이용하여 스마트폰으로 제어하는 기능을 구현하시오
- 4) 토양 습도가 낮으면 경고음을 발생 시키시오
- 5) 물의 수위가 일정 높이가 넘으면 경고음을 발생시키시오
- 6) 물의 수위가 일정 높이가 넘으면 펌프 작동을 중단시키는 기능을 구현하시오