

Chapter 23. 적외선 통신

1. 적외선

흔히 IR이라 불리는 적외선(Infrared)은 가정에서 리모컨에 흔히 사용된다. 아두이노와 함께 사용하는 경우 싼 가격으로 간단하게 무선 조정 장치를 만드는 것이 가능하므로 자동차 등의 컨트롤러로 사용된 경우도 쉽게 찾아볼 수 있다.

리모컨의 버튼을 누르면 리모컨은 초당 38,000회 깜빡거리면서 데이터를 전송한다. 적외선은 자연환경에서도 흔히 볼 수 있지만 38KHz 주파수 대역의 적외선은 찾아보기 어려우므로 데이터 전달을 위해 흔히 사용된다. 대부분의 리모컨은 38KHz 대역을 사용하지만 일부 다른 주파수 대역을 사용하는 경우도 있다.

적외선 통신에서 적외선 신호를 보내기 위해서는 적외선 LED를 사용할 수 있다. 적외선 LED는 일반 LED와 동일한 형태를 가지므로 외형만으로는 일반 LED와 구별하기 어렵다. 적외선 LED도 일반 LED와 마찬가지로 다리가 긴 쪽이 (+), 짧은 쪽이 (-)이다.



그림 1. 적외선 LED

적외선 수신기는 다양한 종류가 판매되고 있으며 학습보드에 포함되어 있는 적외선 수신기는 PL-IRM0101이다. PL-IRM0101은 3개의 핀을 가지고 있으며 2번과 3번 핀을 GND와 VCC에 연결하고 1번 핀은 데이터 수신 용도로 사용한다.

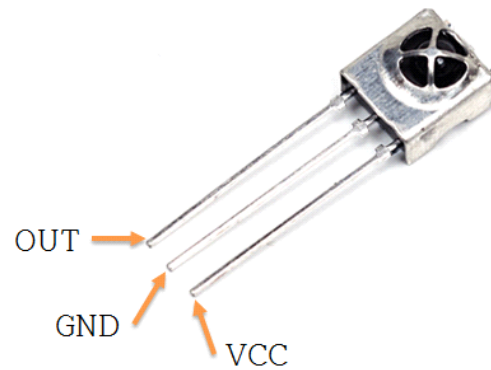


그림 2. PL-IRM0101 적외선 수신기

아두이노 우노의 디지털 3번 핀에 적외선 LED를 연결하고, 아두이노 메가2560의 디지털 12번 핀에 적외선 수신기를 연결하자.

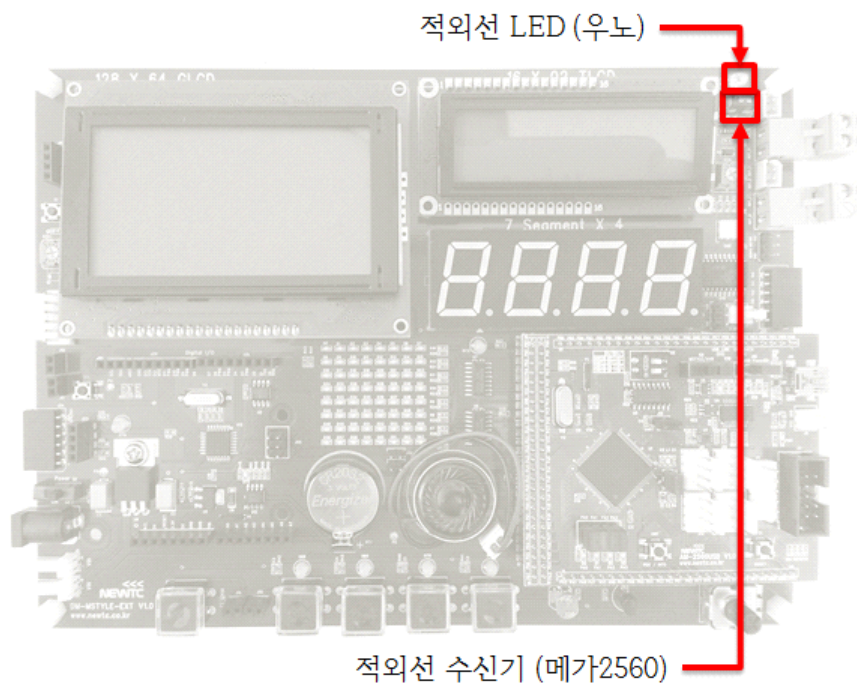


그림 3. 학습보드의 적외선 LED와 수신기

2. 적외선 송수신 라이브러리

아두이노에서 적외선 송수신을 위해서는 IRremote 라이브러리를 사용할 수 있다. IRremote 라이브러리를 다운 받아¹⁾ 설치하자. 먼저 적외선으로 데이터를 송신하는 스케치를 작성해 보

자. 데이터를 전송하는 방법은 간단하다. 먼저 ‘스케치 → Include Library → IRremote’ 메뉴 항목을 선택하여 필요한 헤더 파일을 포함시킨다.

```
#include <IRremote.h>
#include <IRremoteInt.h>
```

다음은 데이터 송신을 위한 IRsend 클래스의 객체를 생성한다.

```
IRsend irsend;
```

객체가 생성되면 IRsend 클래스의 send 함수를 사용하여 데이터를 전송할 수 있다. IRremote 라이브러리를 살펴보면 send 함수는 제조 회사에 따라 다양한 함수가 존재함을 알 수 있다. 즉, 데이터 형식에서 약간씩의 차이가 있다. 여기서는 흔히 사용되는 NEC 방식으로 4바이트의 데이터를 전송하도록 한다. 스케치 1은 적외선 LED를 통해 3초 간격으로 4바이트의 데이터를 NEC 방식으로 인코딩하여 전송하는 스케치의 예이다.

스케치 1. 적외선 데이터 전송 - 우노

```
#include <IRremote.h>
#include <IRremoteInt.h>

IRsend irsend;

long data = 0x18E758A7;           // 전송 데이터

void setup()
{
```

1) <https://github.com/shirriff/Arduino-IRremote>

아두이노의 기본 라이브러리 중 하나에는 RobotIRremote 라이브러리가 포함되어 있다. IRremote 라이브러리를 설치하여 이 장의 스케치를 컴파일 하는 경우 두 라이브러리의 충돌로 컴파일 시에 오류 메시지가 나타날 수 있다. 이러한 경우라면 아두이노 설치 디렉터리 아래 ‘libraries’ 디렉터리에 있는 RobotIRremote 디렉터를 다른 위치로 옮긴 후 사용하면 된다.

```

    Serial.begin(9600);                // 시리얼 통신 초기화
}

void loop()
{
    Serial.print("** Data Sending... ");
    Serial.println(data, HEX);
    irsend.sendNEC(data, 32);          // NEC 방식의 4바이트 데이터 전송
    delay(3000);
}

```

스케치 1에서 한 가지 주의할 점은 적외선 LED가 연결된 핀을 설정하지 않고 있다는 점으로 IRsend 클래스는 디지털 3번 핀을 디폴트로 사용한다. 적외선 수신을 위해서는 IRrecv 클래스가 사용된다. 먼저 적외선 수신기의 데이터 핀이 연결된 핀을 지정하여 객체를 생성한다.

```

int RECV_PIN = 12;
IRrecv irrecv(RECV_PIN);

```

생성된 객체는 enableIRIn 멤버함수로 초기화하면 데이터를 받을 준비는 끝난다.

```

irrecv.enableIRIn();

```

실제 데이터 수신은 decode 함수로 가능하며, 수신되어 해석된 데이터는 전용 클래스인 decode_results 클래스에 저장된다.

```

decode_results results;
irrecv.decode(&results);

```

decode_results 클래스에서 데이터는 unsigned long 형식의 4바이트 크기를 갖는 value 멤버변수에 저장된다. 스케치 2는 적외선 수신기를 통해 데이터를 수신하는 스케치의 예이다.

스케치 2. 적외선 데이터 수신 - 메가2560

```
#include <IRremote.h>
#include <IRremoteInt.h>

int RECV_PIN = 12;                // 적외선 수신기의 DATA 핀 연결

IRrecv irrecv(RECV_PIN);          // 적외선 수신기 초기화
decode_results results;            // 수신 데이터

void setup()
{
  Serial.begin(9600);              // 시리얼 통신 초기화
  irrecv.enableIRIn();             // 적외선 수신기 시작
}

void loop()
{
  if(irrecv.decode(&results)) {    // 디코딩 결과
    Serial.print("Received data : 0x");
    Serial.println(results.value, HEX); // 수신 데이터 출력

    irrecv.resume();               // 다음 데이터 수신
  }

  delay(200);
}
```

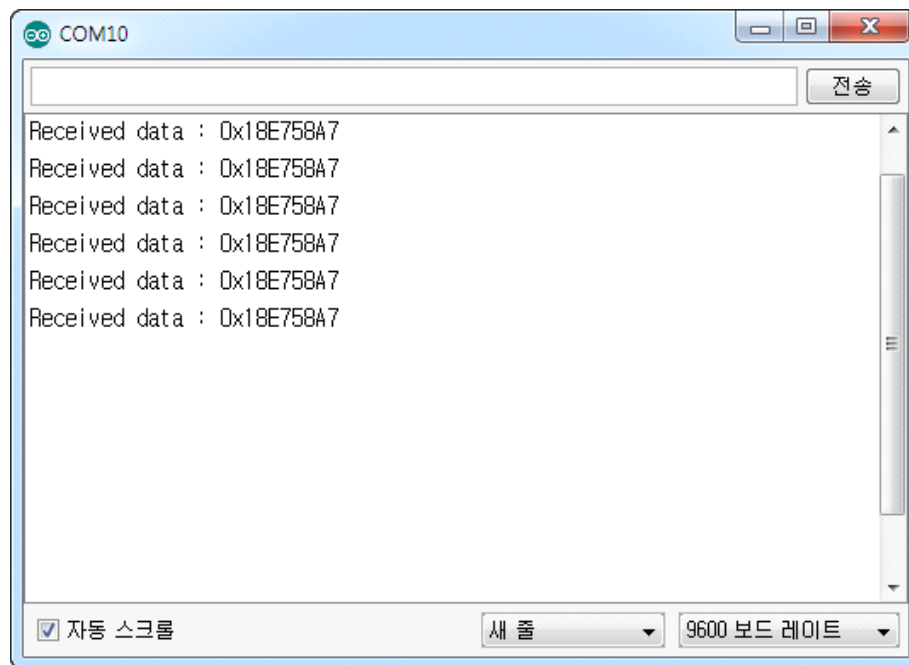


그림 4. 스케치 1과 스케치 2 실행 결과