

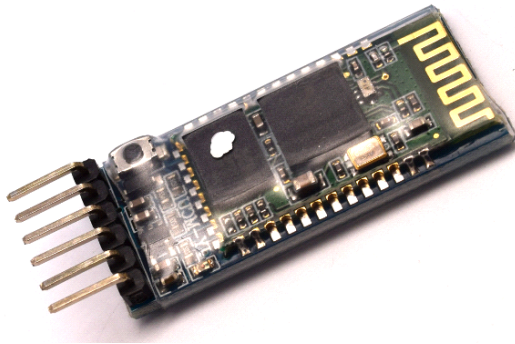
Chapter 20. 아두이노-아두이노 블루투스 통신

1. 블루투스

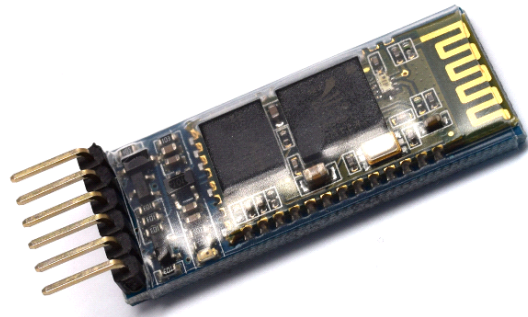
블루투스(Bluetooth)는 1994년 에릭슨(Ericsson)이 개발한 개인 근거리 무선 통신 (Personal Area Network, PAN)을 위한 표준으로 RS-232를 대체하기 위한 저가격, 저전력의 무선 기술로 개발되었다. 블루투스는 기본적으로 10m 이내에서의 통신을 목표로 하지만 통신 범위는 100m까지 확장이 가능하다. 블루투스는 2.4GHz 대역인 ISM (Industrial, Scientific, Medical) 대역을 사용한다. ISM 대역은 산업, 과학 및 의료 목적으로 할당된 대역으로 전파 사용에 대한 허가를 받을 필요가 없어 저전력 개인 무선기기에 많이 사용되고 있으며 무선랜 (WiFi)과 지그비(ZigBee) 역시 ISM 대역을 사용한다.

블루투스는 기기 간의 통신 방식을 정의하기 위해 프로파일(profile)을 사용한다. 프로파일은 블루투스를 이용한 어플리케이션을 구현할 때 특정 어플리케이션에서 사용해야 하는 프로토콜의 종류와 사용 방법을 정의하고 있다. 블루투스 표준을 관리하는 Bluetooth SIG(Special Interest Group)에서는 다양한 프로파일을 정의하고 있으며 이 장에서 사용하는 프로파일은 시리얼 포트 프로파일(Serial Port Profile, SPP)로 시리얼 통신을 에뮬레이션하기 위해 사용되는 프로파일이다. 마이크로컨트롤러에서 SPP를 지원하는 블루투스 모듈을 사용하는 경우 UART 통신 코드를 약간만 수정하고 블루투스 모듈을 연결하는 것만으로 블루투스 통신이 가능하므로 손쉽게 유선 통신을 블루투스를 이용한 무선 통신으로 바꿀 수 있다.

SPP 프로파일을 지원하는 블루투스 모듈 중 이 장에서는 HC-06 모듈을 사용한다. HC-06 모듈은 마스터와 슬레이브 모듈이 별도로 만들어져 있다. 외형상으로는 큰 차이가 없으며 마스터 모듈에는 슬레이브 모듈과는 다른 표시가 있어 구별된다. 블루투스 모듈은 6핀으로 구성되어 있지만 가운데 4핀만을 연결하면 되고 나머지 2핀은 사용하지 않는다.



(a) 마스터 모듈



(b) 슬레이브 모듈

그림 1. 블루투스 모듈¹⁾

아두이노 메가2560에는 디지털 18번과 19번 핀을 사용하는 Serial1을 통해 블루투스 모듈을 연결하며, 아두이노 우노에는 디지털 4 번과 5번 핀을 사용하는 SoftwareSerial을 통해 블루투스 모듈을 연결한다.

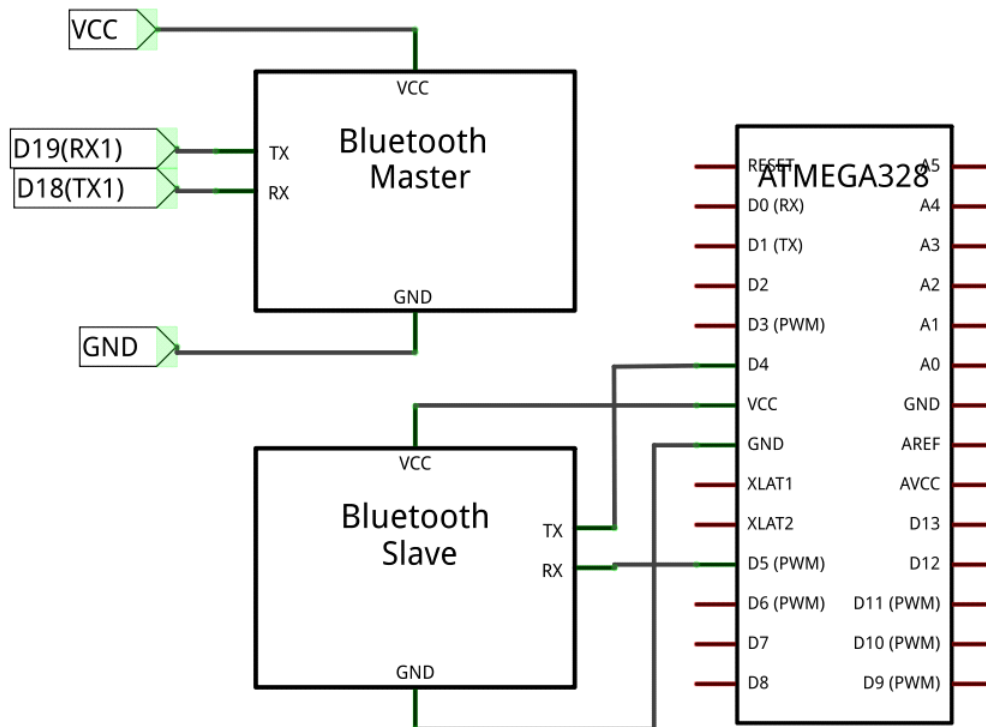


그림 2. 블루투스 모듈 연결 회로도

1) 이미지 출처: <http://www.newtc.co.kr/>

이전 장에서 동일한 핀을 UART 시리얼 통신을 위해 사용하였던 점을 기억할 것이다. 앞에서도 언급한 바와 같이 SPP는 시리얼 통신을 에뮬레이션 하는 프로파일로, 유선의 UART 시리얼 통신이 무선의 블루투스 통신으로 바뀐 점을 제외하면 UART 시리얼 통신과 블루투스 통신은 동일하다고 볼 수 있다.

2. 블루투스 모듈 설정

먼저 슬레이브 모듈을 우노에 연결하고 스케치 1을 우노에 업로드 하자. 우노는 하나의 하드웨어 시리얼 포트만을 제공하므로 블루투스 연결을 위해서는 소프트웨어적으로 UART 시리얼 통신을 구현한 SoftwareSerial 클래스를 사용하였다.

스케치 1. 블루투스 슬레이브 모듈 설정

```
#include <SoftwareSerial.h>

// SoftwareSerial(RX, TX) 형식으로 블루투스 모듈과 교차하여 연결
SoftwareSerial BTSerial(4, 5);

void setup()
{
    Serial.begin(9600);           // 컴퓨터와의 시리얼 통신 초기화
    BTSerial.begin(9600);         // 블루투스 모듈 초기화
}

void loop()
{
    // 블루투스 모듈 → 아두이노 → 시리얼 모니터
    if (BTSerial.available())
        Serial.write(BTSerial.read());

    // 시리얼 모니터 → 아두이노 → 블루투스 모듈
    if (Serial.available())
        BTSerial.write(Serial.read());
}
```

블루투스 모듈의 설정을 위해서는 AT 명령어를 블루투스 모듈로 전달하여야 한다. 시리얼 모니터에서 'AT'를 입력해 보자. 정상적으로 연결이 되었다면 'AT'는 블루투스 모듈로 전달되고, 'AT' 명령을 전달받은 블루투스 모듈은 현재 상태로 'OK' 문자열을 반환한다. HC-06 블루투스 모듈은 디폴트값으로 9600 보율의 속도로 설정되어 있으므로 스케치 1에서도 9600 보율을 사용하고 있다. 또한 시리얼 모니터에서 'No line ending'이 선택되어 있어야 한다는 점도 주의하여야 한다. 블루투스 모듈 설정을 위해 표 1의 명령어를 순서대로 실행하자.

명령어	사용 방법	반환값	비고
AT	AT	OK	
AT+NAME	AT+NAME <u>megaSlave</u>	OKsetname	이름 설정
AT+PIN	AT+PIN <u>1234</u>	OKsetPIN	PIN 설정
AT+BAUD	AT+BAUD <u>4</u>	OK9600	통신 속도 설정

표 1. 블루투스 슬레이브 모듈 설정 AT 명령어

AT+NAME은 모듈의 이름을 설정하기 위해 사용된다. AT+PIN은 연결 코드, 패스 코드, 핀 코드 등으로도 불리는 일종의 비밀번호 설정을 위해 사용된다. AT+BAUD는 통신 속도를 설정하기 위해 사용되며 숫자에 따른 통신 속도는 표 2와 같다.

숫자	전송 속도(Baud)	숫자	전송 속도(Baud)
1	1200	5	19200
2	2400	6	38400
3	4800	7	57600
4	9600	8	115200

표 2. 전송 속도

유사하게 마스터 모듈을 설정해 보자. 마스터 모듈 설정을 위해서는 스케치 2를 업로드 한다. 스케치 2는 메가2560에서 제공하는 하드웨어 시리얼 포트 중 하나인 Serial1을 사용한다는 점을 제외하면 스케치 1과 동일하다.

스케치 2. 블루투스 마스터 모듈 설정

```
void setup()
{
  Serial.begin(9600);          // 컴퓨터와의 시리얼 통신 초기화
  Serial1.begin(9600);         // 블루투스 모듈 초기화
}

void loop()
{
  // 블루투스 모듈 → 아두이노 → 시리얼 모니터
  if (Serial1.available())
    Serial.write(Serial1.read());

  // 시리얼 모니터 → 아두이노 → 블루투스 모듈
  if (Serial.available())
    Serial1.write(Serial.read());
}
```

스케치 2를 업로드 하고 'AT+PIN1234'와 'AT+BAUD4' 만을 시리얼 모니터를 통해 실행한다. 마스터 모듈은 슬레이브 모듈과 달리 이름을 지정할 수 없다.

설정이 끝났으면 두 모듈을 모두 연결해 보자. 블루투스는 1:n 연결을 지원하지만 HC-06 모듈은 1:1 연결만을 지원한다. 슬레이브 모듈과 마스터 모듈이 동일한 PIN과 동일한 속도로 설정되어 있으면 마스터와 슬레이브 모듈은 자동으로 연결된다. 두 모듈이 서로 연결되면 마스터 모듈에는 슬레이브 모듈의 정보가 저장되고 다른 슬레이브 모듈은 연결될 수 없다. 다른 슬레이브 모듈을 연결하고자 한다면 모듈의 KEY 핀에 연결된 리셋 버튼을 눌러 슬레이브 정보를 삭제하여야 한다. 슬레이브 모듈은 연결되지 않은 상태에서는 LED가 깜빡거리다가 연결되면 항상 켜진 상태로 바뀐다. 마스터 모듈은 슬레이브 정보가 저장되어 있지만 연결되지 않은 상태라면 느린 속도로 깜빡거리지만, 슬레이브 정보 없이 연결되지 않은 상태에서는 빠른 속도로 깜빡거린다. 연결이 된 후에는 켜진 상태로 바뀌는 것은 슬레이브 모듈과 동일하다.

3. 블루투스 통신

SPP 프로파일을 사용한 블루투스 통신은 UART 시리얼 통신과 동일하다. 아두이노 우노에는 스케치 3을 업로드 하자. 스케치 3은 수신된 문자열에 슬레이브로부터의 에코 백임을 표시하

는 문자열을 추가하여 전송하는 스케치의 예이다.

스케치 3. 슬레이브의 에코 백 - 우노

```
#include <SoftwareSerial.h>
// SoftwareSerial(RX, TX) 형식으로 블루투스 모듈과 교차하여 연결된다.
SoftwareSerial BTSerial(4, 5);
char buffer[100]; // 수신 문자열 버퍼

void setup(){
  BTSerial.begin(9600);
}

void loop(){
  if(BTSerial.available()){ // 마스터로부터 데이터 수신
    int no = BTSerial.readBytes(buffer, 99);
    buffer[no - 1] = '\0'; // 개행문자 제거
    BTSerial.print(" (From Slave : "); // 마스터로 echo back
    BTSerial.print(buffer);
    BTSerial.println(")");
  }
}
```

아두이노 메가2560에는 스케치 4를 업로드 하자. 스케치 4는 시리얼 모니터에 입력된 데이터를 슬레이브로 전송하고, 슬레이브로부터 수신된 데이터를 시리얼 모니터로 출력하는 스케치의 예이다. 스케치를 업로드 한 후 시리얼 모니터를 실행시키고 문자열을 입력하면 동일한 내용이 시리얼 모니터로 두 번 출력되는 것을 확인할 수 있다.

스케치 4. 마스터의 데이터 전송 - 메가2560

```
void setup()
{
  Serial.begin(9600); // 시리얼 통신 초기화
  Serial1.begin(9600); // 블루투스 모듈 초기화
}
```

```
void loop()
{
  // 블루투스로 전달된 데이터를 시리얼 모니터로 출력
  while(Serial1.available())
    Serial.write(Serial1.read());

  // 시리얼 모니터로 입력된 데이터를 시리얼 모니터와 블루투스로 출력
  while(Serial.available()){
    byte data = Serial.read();
    Serial1.write(data);
    Serial.write(data);
  }
}
```

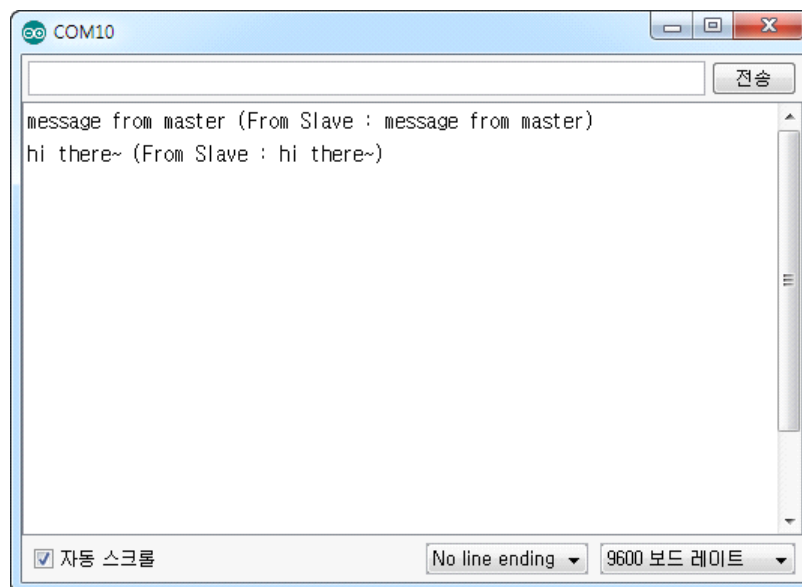


그림 3. 스케치 3과 스케치 4 실행 결과 - 메가2560