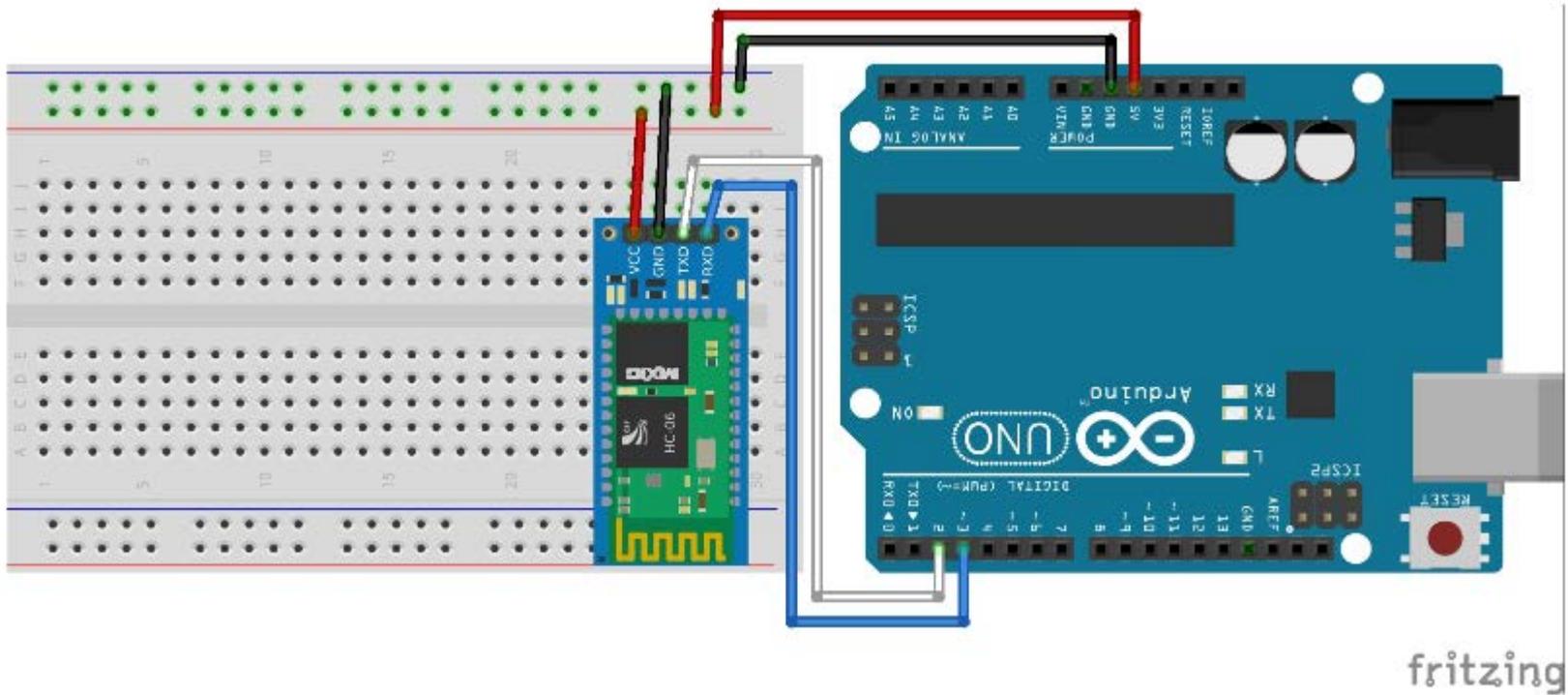


# 아두이노에 블루투스 모듈 연결하기



# 아두이노에 블루투스 모듈 연결 스케치

```
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2,3); // 아두이노의 RX, TX에 블루투스 모듈의 TX, RX를 연결

void setup() {
  // 아두이노의 시리얼 전송속도(BAUD Rate)와 블루투스의 데이터 전송속도를 같게 해 줌
  Serial.begin(9600);
  BTSerial.begin(9600);
}

void loop() {
  // 블루투스 시리얼이 연결되어 사용가능하다면, 블루투스를 통해 스마트폰에서 보내온(읽은) 데이터를 아두이노의 시리얼
  모니터에 보여줌
  if (BTSerial.available()) {
    Serial.write(BTSerial.read());
  }
  // 아두이노의 시리얼 모니터가 연결되어 있다면, 시리얼 모니터에서 보내온(읽은) 데이터를 블루투스를 통해 스마트폰에
  보여줌
  if (Serial.available()) {
    BTSerial.write(Serial.read());
  }
}
```

# 블루투스 통신

## SoftwareSerial 사용 예

```
#include <SoftwareSerial.h> //시리얼통신 라이브러리 호출

int blueTX=2; //Tx (보내는핀 설정)at
int blueRX=3; //Rx (받는핀 설정)
SoftwareSerial BTSerial(blueTX, blueRX); //시리얼 통신을 위한 객체선언

void setup()
{
  Serial.begin(9600); //시리얼모니터
  BTSerial.begin(9600); //블루투스 시리얼
}
void loop()
{
  if (BTSerial.available()) {
    Serial.write(BTSerial.read()); //블루투스측 내용을 시리얼모니터에 출력
  }
  if (Serial.available()) {
    BTSerial.write(Serial.read()); //시리얼 모니터 내용을 블루투스 측에 WRITE
  }
}
```

# 아두이노에 블루투스 모듈 연결 테스트 환경 설정

스케치에서 시리얼 모니터를 열고 AT 를 입력하고 전송을 누르면 밑에 OK 라는 글자가 나오면 블루투스 연결이 잘 된 것입니다.

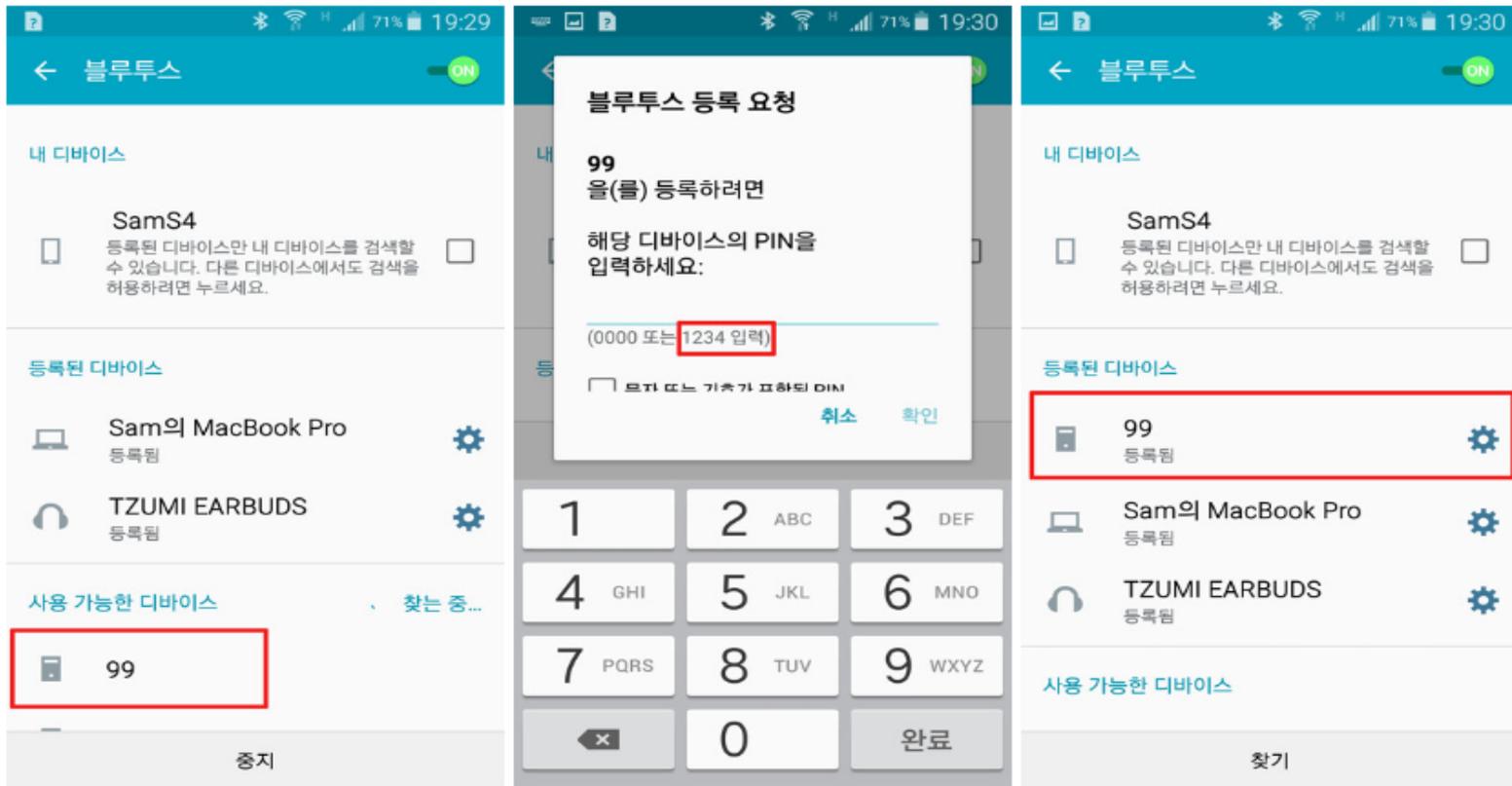
AT+NAME99 이렇게 입력하면 블루투스 모듈의 이름이 99로 변경됩니다.

이제 스마트폰을 준비하고 설정에 가서 블루투스 환경설정을 엽니다.

아래쪽 찾기 버튼을 클릭하면 방금 설정한 99번 블루투스가 검색이 됩니다.

99번을 선택하면 PIN 번호를 넣으라는 화면이 나옵니다.

여기에 1234를 입력하면 이제 이 스마트폰에 블루투스 모듈이 등록이 됩니다.



# 앱인벤터로 블루투스 모듈 이용 아두이노와 데이터 주고받기

Display hidden components in Viewer  
 Check to see Preview on Tablet size.



Screen1

Select Bluetooth Client    Disconnect

Red    Yellow    Green

BluetoothClient1    Clock1    Notifier1    ActivityStarter1

### Components

- Screen1
  - HorizontalArrangement1
    - ListPicker1
    - Disconnect
  - HorizontalArrangement2
    - RedButton
    - YellowButton
    - GreenButton
  - BluetoothClient1
  - Clock1
  - Notifier1
  - ActivityStarter1

Rename    Delete

### Media

Upload File ...

### Properties

Screen1

AboutScreen

AlignHorizontal  
Left : 1 ▾

AlignVertical  
Top : 1 ▾

AppName

BackgroundColor  
 White

BackgroundImage

CloseScreenAnimation  
Default ▾

Icon

OpenScreenAnimation  
Default ▾

ScreenOrientation  
Unspecified ▾

Scrollable

ShowStatusBar

```
when Screen1.Initialize
do
  if not BluetoothClient1.Enabled
  then
    set ActivityStarter1.Action to "android.bluetooth.adapter.action.REQUEST_ENABLE"
    call ActivityStarter1.StartActivity
```

```
when ListPicker1.BeforePicking
do
  set ListPicker1.Elements to BluetoothClient1.AddressesAndNames
```

```
when ListPicker1.AfterPicking
do
  evaluate but ignore result call BluetoothClient1.Connect
  address ListPicker1.Selection
```

```
when Clock1.Timer
do
  if BluetoothClient1.IsConnected
  then
    set ListPicker1.BackgroundColor to #00FF00
    set ListPicker1.Text to "Connected"
  else
    set ListPicker1.BackgroundColor to #808080
    set ListPicker1.Text to "Disconnected"
```

```
when Disconnect.Click
do
  if BluetoothClient1.IsConnected
  then
    call BluetoothClient1.Disconnect
```

```
when Screen1.ErrorOccurred
  component functionName errorNumber message
do
  call Notifier1.ShowAlert
  notice
    join "에러발생"
    get component
    ","
    get functionName
    ","
    get errorNumber
    ","
    get message
```

```
when RedButton .Click
do
  if BluetoothClient1 .IsConnected
  then
    call BluetoothClient1 .SendText
      text "r"
    set RedButton .BackgroundColor to red
    set YellowButton .BackgroundColor to gray
    set GreenButton .BackgroundColor to gray
```

```
when YellowButton .Click
do
  if BluetoothClient1 .IsConnected
  then
    call BluetoothClient1 .SendText
      text "y"
    set RedButton .BackgroundColor to gray
    set YellowButton .BackgroundColor to yellow
    set GreenButton .BackgroundColor to gray
```

```
when GreenButton .Click
do
  if BluetoothClient1 .IsConnected
  then
    call BluetoothClient1 .SendText
      text "g"
    set RedButton .BackgroundColor to gray
    set YellowButton .BackgroundColor to gray
    set GreenButton .BackgroundColor to green
```

```
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2, 3);

int redPin = 10;
int yellowPin = 9;
int greenPin = 8;

void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(yellowPin, OUTPUT);
  pinMode(greenPin, OUTPUT);

  Serial.begin(9600);
  BTSerial.begin(9600);
}
```

```
void loop() {
  char data;
  if(BTSerial.available()) {
    data = BTSerial.read();
    Serial.println(data);
    delay(1);

    if(data=='r') {
      digitalWrite(redPin, HIGH);
      digitalWrite(yellowPin, LOW);
      digitalWrite(greenPin, LOW);
      delay(150);
    }
    if(data=='y') {
      digitalWrite(redPin, LOW);
      digitalWrite(yellowPin, HIGH);
      digitalWrite(greenPin, LOW);
      delay(150);
    }
    if(data=='g') {
      digitalWrite(redPin, LOW);
      digitalWrite(yellowPin, LOW);
      digitalWrite(greenPin, HIGH);
      delay(150);
    }
  }
}
```

initialize global sensorValue to 0

when ListPicker1 .BeforePicking

do set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when ListPicker1 .AfterPicking

do set ListPicker1 . Selection to call BluetoothClient1 .Connect  
address ListPicker1 . Selection

if BluetoothClient1 . IsConnected  
then

when Clock1 .Timer

do if BluetoothClient1 . IsConnected  
then if call BluetoothClient1 .BytesAvailableToReceive  $\geq$  4  
then set global sensorValue to call BluetoothClient1 .ReceiveSignedBytes  
numberOfBytes 4  
set Label2 . Text to get global sensorValue

when Disconnect .Click

do if BluetoothClient1 . IsConnected  
then call BluetoothClient1 .Disconnect

```
#include <SoftwareSerial.h>
SoftwareSerial btSerial(2,3);
int sensor = A0;
int sensorValue;

void setup() {
  Serial.begin(9600);
  btSerial.begin(9600);
}

void loop() {
  sensorValue = analogRead(sensor);
  Serial.println(sensorValue);
  btSerial.println(sensorValue);
  delay(5000);
}
```