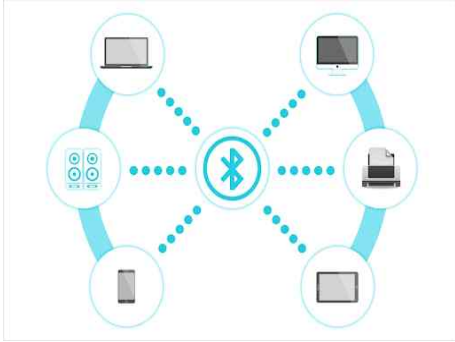





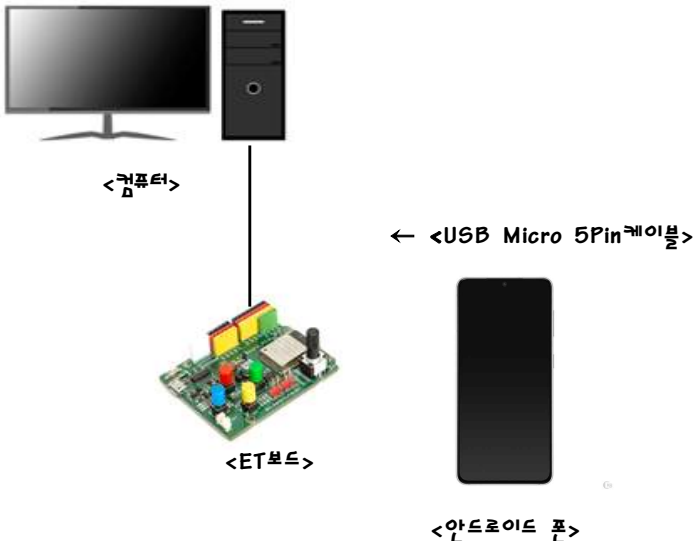


ET보드 level2 콘텐츠

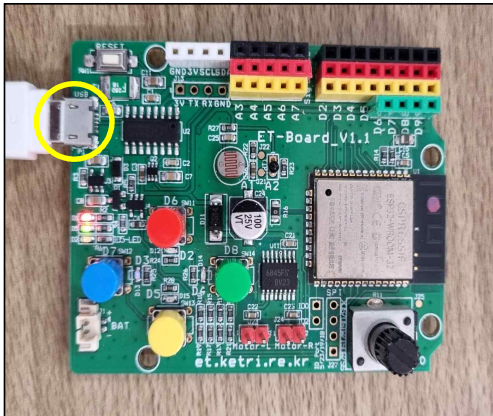
211201 최수범

학습내용	블루투스 연결 해보기
소스	01._Bluetooth_connect.ino
개념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> ← <USB Micro 5Pin 케이블></p> <p><ET보드> <안드로이드 폰></p>

회로 구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스 코드

// 참고사항 @항목의 블루투스 테스트앱을 설치 후 진행합니다.

```
#include "BluetoothSerial.h"
```

```
BluetoothSerial SerialBT;
```

```
char names[] = "your_name"; // 블루투스의 이름 입력
```

```
void setup(){
```

```
  Serial.begin(115200); // 통신속도 설정
```

```
  SerialBT.begin(names); // 자신의 이름으로 된 블루투스 생성
```

```
  while(!SerialBT.connected(1000)){
```

```
    Serial.println("연결되지 않았습니다.");
```

```
  }
```

```
  delay(1000); // 제대로 연결 하기 위해 잠시 대기
```

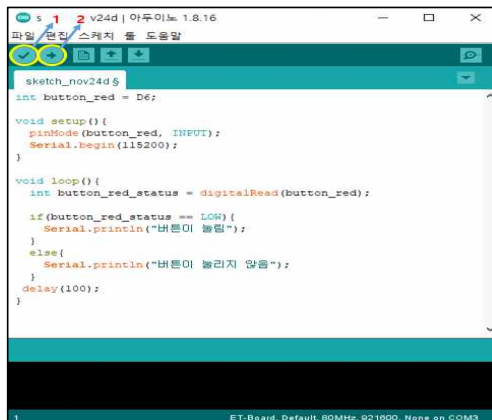
```
  String message = names + String(" : 연결에 성공했습니다!");
```

```
Serial.println(message);
SerialBT.println(message.c_str());
}

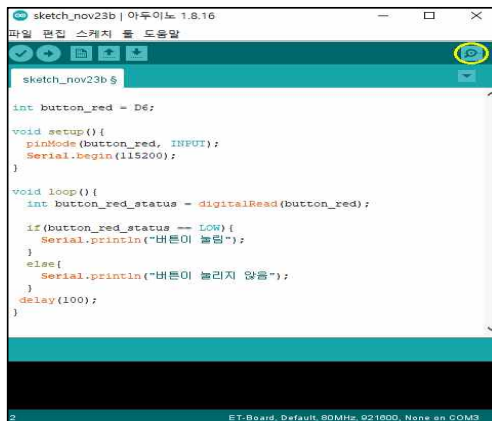
void loop(){
}
```

// 시리얼에 연결에 성공했다고 출력
// 연결된 블루투스에 성공했다고 전송

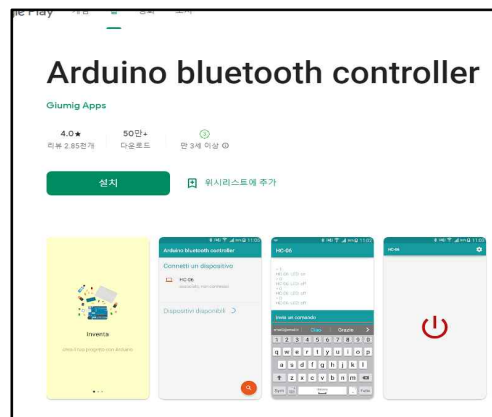
동작 과정



① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드 합니다.



② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해줍니다.



③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다.
(참고사항 ② 소스코드 테스트앱 확인)

Available devices



5F:F2:0C:26:61:D1



4A:5D:6C:1D:15:C2



5C:3C:F8:8F:9F:A3



68:11:69:A5:A3:69



your-name

- ④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

Connect in



Controller mode



Switch mode



Dimmer mode



Terminal mode

- ⑤ **Terminal mode**에 연결합니다.

your-name

your-name: your-name : 연결에 성공했습니다!

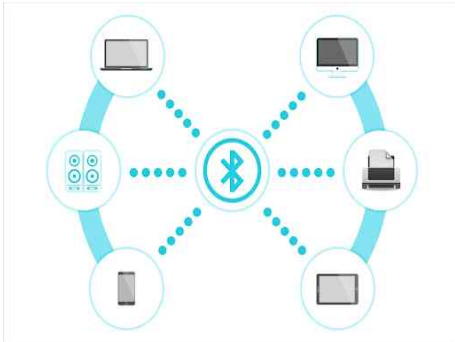





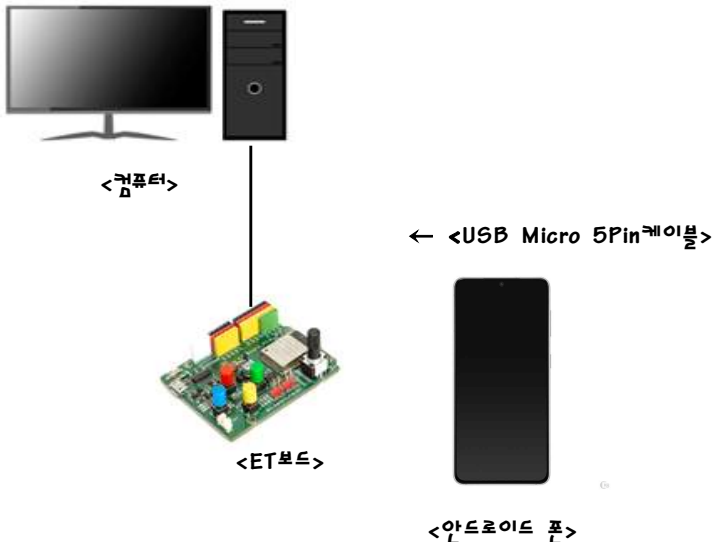
- ⑥ 연결 상태를 확인할 수 있습니다.

COM3

```
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
```

- ⑦ 시리얼 모니터에서
연결 상태를 확인할 수 있습니다.

참 고 사 항	<p>① 회로</p> <ul style="list-style-type: none"> ○ 블루투스에 대한 참고사항 링크 https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4 https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124 <p>② 소스코드</p> <ul style="list-style-type: none"> ○ 블루투스 테스트 앱 https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal ○ 블루투스를 사용하는 방법 https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/ ○ esp32 블루투스 라이브러리 https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/

학습내용	시리얼 통신으로 수신한 데이터를 블루투스 통신으로 송신 해보기
소스	02._Bluetooth_one_way_communication.ino
개념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> ← <USB Micro 5Pin 케이블></p> <p><ET보드> <안드로이드 폰></p>

<div data-bbox="143 609 220 687" data-label="Section-Header"> <h2>회로 구성</h2> </div>	<div data-bbox="304 215 799 631" data-label="Image"> </div> <div data-bbox="865 383 1353 461" data-label="Text"> <p>① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.</p> </div> <div data-bbox="304 667 799 1084" data-label="Image"> </div> <div data-bbox="865 855 1441 891" data-label="Text"> <p>② 다른 한쪽 케이블을 ET보드에 연결합니다.</p> </div>
<div data-bbox="143 1523 220 1601" data-label="Section-Header"> <h2>소스 코드</h2> </div>	<div data-bbox="250 1120 1099 1158" data-label="Text"> <p>// 참고사항 @항목의 블루투스 테스트앱을 설치 후 진행합니다.</p> </div> <div data-bbox="250 1209 1418 2007" data-label="Text"> <pre> #include "BluetoothSerial.h" BluetoothSerial SerialBT; char names[] = "your_name"; // 블루투스의 이름 입력 void setup(){ Serial.begin(115200); // 통신속도 설정 SerialBT.begin(names); // 자신의 이름으로 된 블루투스 생성 while(!SerialBT.connected(1000)){ Serial.println("연결되지 않았습니다."); } delay(1000); // 제대로 연결 하기위해 잠시 대기 String message = names + String(" : 연결에 성공했습니다!"); Serial.println(message); // 시리얼에 연결에 성공했다고 출력 SerialBT.println(message.c_str()); // 연결된 블루투스에 성공했다고 전송 </pre> </div>

	<pre> } void loop(){ if(Serial.available()){ SerialBT.write(Serial.read()); } delay(20); } </pre> <p>// 시리얼에 읽을 데이터가 있다면 // 블루투스로 시리얼 값 넘겨주기</p> <p>// 0.02초 대기</p>
<p>동작 과정</p>	<div data-bbox="300 658 796 1075"> </div> <p>① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드합니다.</p> <div data-bbox="300 1111 796 1527"> </div> <p>② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해 줍니다.</p> <div data-bbox="300 1563 796 1980"> </div> <p>③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다. (참고사항 ② 소스코드 테스트앱 확인)</p>

Available devices



5F:F2:0C:26:61:D1



4A:5D:6C:1D:15:C2



5C:3C:F8:8F:9F:A3



68:11:69:A5:A3:69



your-name

- ④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

Connect in



Controller mode



Switch mode



Dimmer mode



Terminal mode

- ⑤ **Terminal mode**에 연결합니다.

your-name

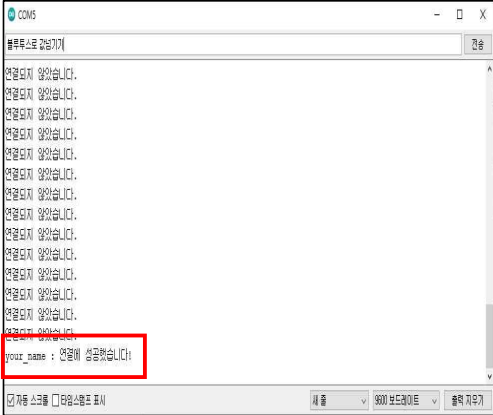
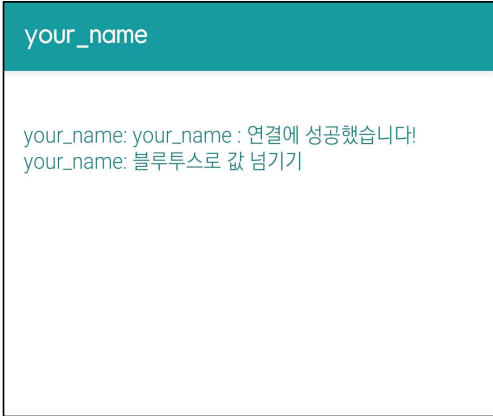
your-name: your-name : 연결에 성공했습니다!

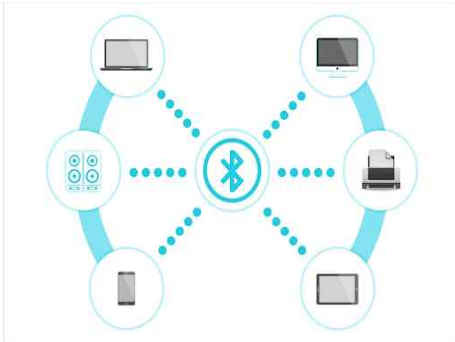





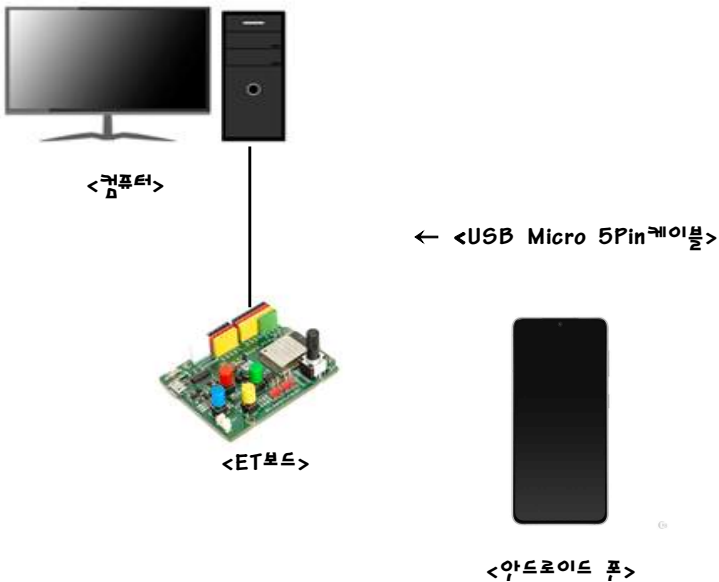
- ⑥ 연결 상태를 확인할 수 있습니다.

COM3

```
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
```

- ⑦ 시리얼 모니터에서
연결 상태를 확인할 수 있습니다.

		<p>⑧ 시리얼 모니터에서 값을 입력하고 블루투스를 통해 넘깁니다</p>
<p>동작 과정</p>		<p>⑨ 연결된 기기를 통해 값이 넘어온 걸 확인할 수 있습니다.</p>
<p>참고 사항</p>	<p>① 회로</p> <ul style="list-style-type: none"> ○ 블루투스에 대한 참고사항 링크 https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4 https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124 <p>② 소스코드</p> <ul style="list-style-type: none"> ○ 블루투스 테스트 앱 https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal ○ 블루투스를 사용하는 방법 https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/ ○ esp32 블루투스 라이브러리 https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/ 	

학습내용	블루투스를 이용하여 시리얼 통신하기
소스	03._Bluetooth_two_way_communication.ino
개 념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>

<div data-bbox="143 649 220 725" data-label="Text"> <p>외 로 구 성</p> </div>	<div data-bbox="304 253 799 669" data-label="Image"> </div> <div data-bbox="873 421 1361 501" data-label="Text"> <p>① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.</p> </div> <div data-bbox="304 703 799 1122" data-label="Image"> </div> <div data-bbox="873 891 1450 929" data-label="Text"> <p>② 다른 한쪽 케이블을 ET보드에 연결합니다.</p> </div>
<div data-bbox="143 1545 220 1621" data-label="Text"> <p>소 스 코 드</p> </div>	<div data-bbox="248 1140 1099 1176" data-label="Text"> <p>// 참고사항 @항목의 블루투스 테스트 앱을 설치 후 진행합니다.</p> </div> <div data-bbox="248 1232 1345 2024" data-label="Text"> <pre> #include "BluetoothSerial.h" BluetoothSerial SerialBT; char names[] = "your_name"; // 블루투스의 이름 입력 void setup(){ Serial.begin(115200); // 통신속도 설정 SerialBT.begin(names); // 자신의 이름으로 된 블루투스 생성 while(!SerialBT.connected(1000)){ Serial.println("연결되지 않았습니다."); } delay(1000); // 제대로 연결이 되어야하니 잠시 대기 String message = names + String(" : 연결에 성공했습니다!"); Serial.println(message); // 시리얼에 연결에 성공했다고 출력 SerialBT.println(message.c_str()); // 연결된 블루투스에 성공했다고 전송 </pre> </div>

```

}

void loop(){
  if(Serial.available()){
    SerialBT.write(Serial.read());
  }

  if(SerialBT.available()){
    Serial.write(SerialBT.read());
  }

  delay(20);
}

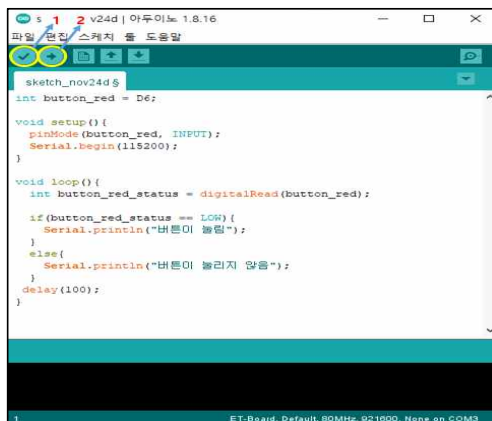
```

// 시리얼에 읽을 데이터가 있다면
// 블루투스로 시리얼 값 넘겨주기

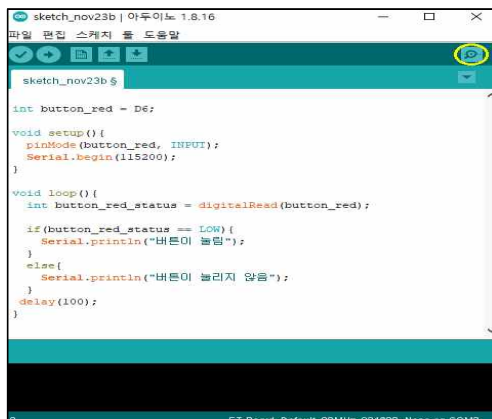
// 블루투스에 읽을 데이터가 있다면
// 시리얼 모니터에 출력

// 0.02초 대기

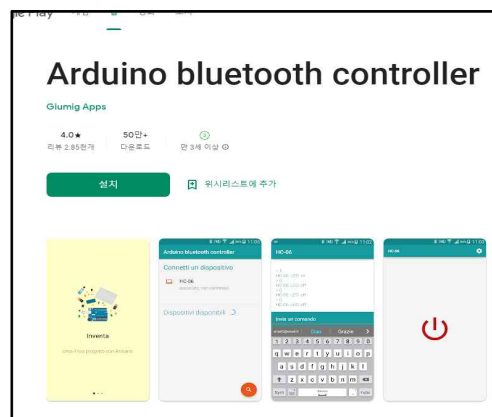
동작과정



① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드합니다.



② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해 줍니다.



③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다.
(참고사항 ② 소스코드 테스트앱 확인)

Available devices



5F:F2:0C:26:61:D1



4A:5D:6C:1D:15:C2



5C:3C:F8:8F:9F:A3



68:11:69:A5:A3:69



your-name

- ④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

Connect in



Controller mode



Switch mode



Dimmer mode



Terminal mode

- ⑤ **Terminal mode**에 연결합니다.

your-name

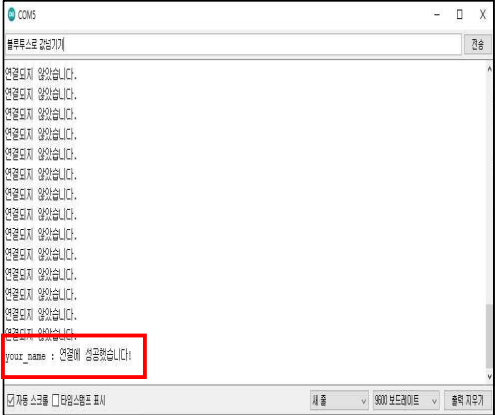
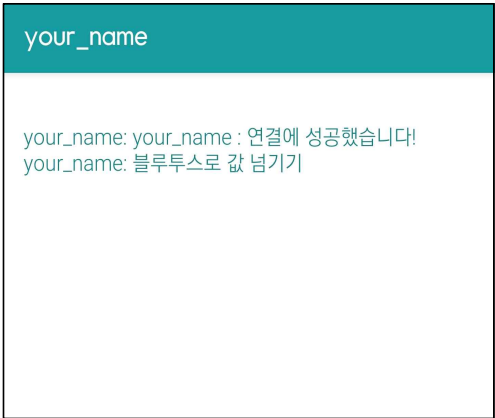
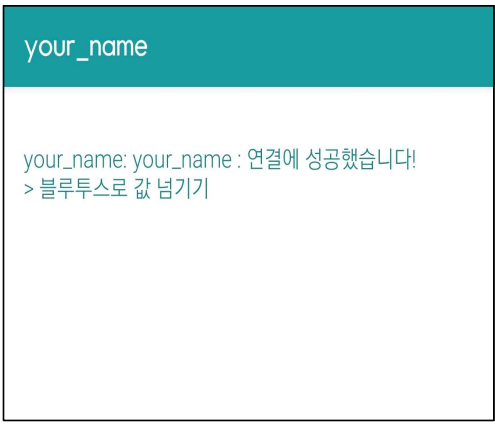
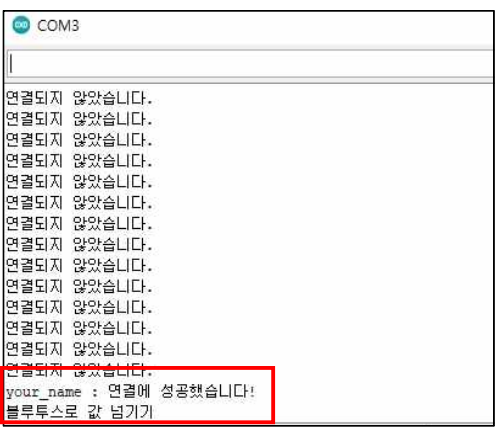
your-name: your-name : 연결에 성공했습니다!

- ⑥ 연결 상태를 확인할 수 있습니다.

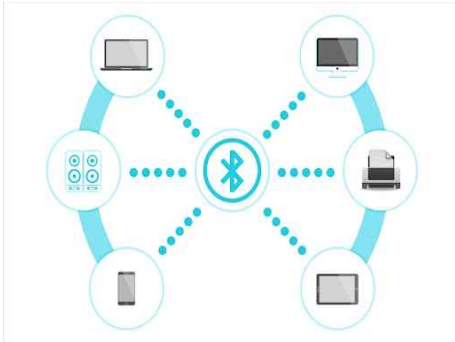





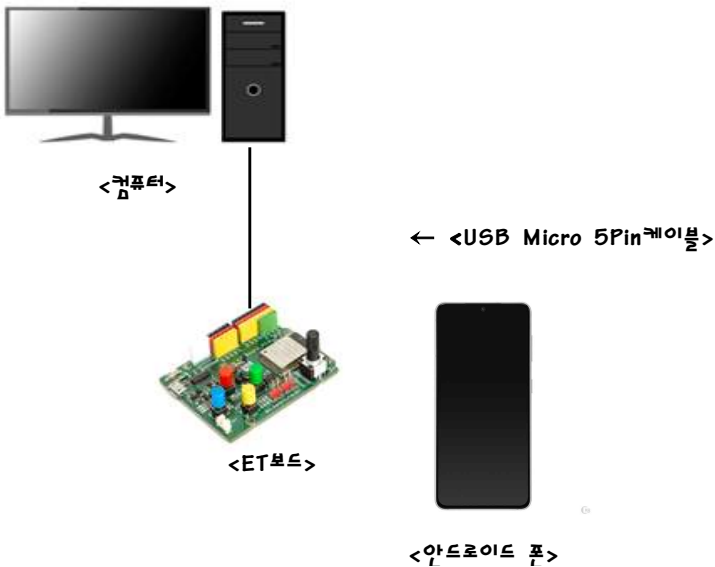
COM3

```
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
```

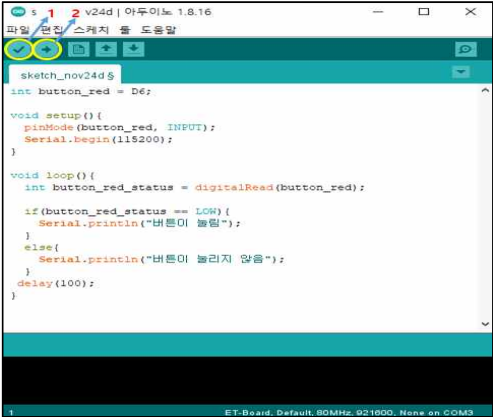
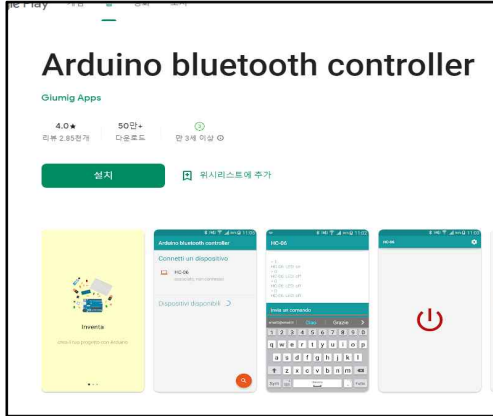
- ② **시리얼 모니터**에서
연결상태를 확인할 수 있습니다.

		<p>③ 시리얼 모니터에서 값을 입력하고 블루투스를 통해 넘깁니다</p>
<p>동 작 과 정</p>		<p>④ 블루투스에 연결된 기기를 통해 값이 넘어온걸 확인할 수 있습니다.</p>
		<p>⑤ 연결된 기기로 값을 입력하고 블루투스를 통해 다시 보냅니다.</p>
		<p>⑥ 최종적으로 시리얼모니터에 값이 넘어온걸 확인할 수 있습니다.</p>

<p>참 고 사 항</p>	<p>① 회로</p> <ul style="list-style-type: none"> ○ 블루투스에 대한 참고사항 링크 https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4 https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124 <p>② 소스코드</p> <ul style="list-style-type: none"> ○ 블루투스 테스트 앱 https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal ○ 블루투스를 사용하는 방법 https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/ ○ esp32 블루투스 라이브러리 https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/
----------------------------	--

학습내용	블루투스를 이용하여 가변저항의 값을 앱에 보내기
소스	04._Bluetooth_variable_resistance.ino
개념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> ← <USB Micro 5Pin 케이블></p> <p><ET보드> <안드로이드 폰></p>

<div data-bbox="143 609 220 687" data-label="Section-Header"> <h2>회로 구성</h2> </div>	<div data-bbox="304 215 799 631" data-label="Image"> </div> <div data-bbox="865 383 1353 461" data-label="List-Group"> <ol style="list-style-type: none"> ① 컴퓨터 본체의 USB포트에 케이블을 연결합니다. </div> <div data-bbox="304 667 799 1084" data-label="Image"> </div> <div data-bbox="865 855 1441 893" data-label="List-Group"> <ol style="list-style-type: none"> ② 다른 한쪽 케이블을 ET보드에 연결합니다. </div>
<div data-bbox="143 1525 220 1603" data-label="Section-Header"> <h2>소스 코드</h2> </div>	<div data-bbox="250 1120 1099 1158" data-label="Text"> <p>// 참고사항 @항목의 블루투스 테스트앱을 설치 후 진행합니다.</p> </div> <div data-bbox="250 1211 687 1247" data-label="Text"> <pre>#include "BluetoothSerial.h"</pre> </div> <div data-bbox="250 1299 641 1337" data-label="Text"> <pre>BluetoothSerial SerialBT;</pre> </div> <div data-bbox="250 1388 670 1426" data-label="Text"> <pre>char names[] = "your_name";</pre> </div> <div data-bbox="901 1388 1249 1426" data-label="Text"> <p>// 블루투스의 이름 입력</p> </div> <div data-bbox="250 1433 501 1469" data-label="Text"> <pre>int sensor = A0;</pre> </div> <div data-bbox="901 1433 1141 1469" data-label="Text"> <p>// 가변저항센서</p> </div> <div data-bbox="250 1523 459 1561" data-label="Text"> <pre>void setup(){</pre> </div> <div data-bbox="279 1568 611 1606" data-label="Text"> <pre> Serial.begin(115200);</pre> </div> <div data-bbox="901 1568 1149 1606" data-label="Text"> <p>// 통신속도 설정</p> </div> <div data-bbox="279 1612 625 1648" data-label="Text"> <pre> SerialBT.begin(names);</pre> </div> <div data-bbox="901 1612 1425 1648" data-label="Text"> <p>// 자신의 이름으로 된 블루투스 생성</p> </div> <div data-bbox="279 1702 799 1738" data-label="Text"> <pre> while(!SerialBT.connected(1000)){</pre> </div> <div data-bbox="306 1744 890 1783" data-label="Text"> <pre> Serial.println("연결되지 않았습니다.");</pre> </div> <div data-bbox="279 1792 306 1827" data-label="Text"> <pre> }</pre> </div> <div data-bbox="279 1834 472 1872" data-label="Text"> <pre> delay(1000);</pre> </div> <div data-bbox="901 1834 1404 1872" data-label="Text"> <p>// 제대로 연결 하기위해 잠시 대기</p> </div> <div data-bbox="279 1921 1161 1962" data-label="Text"> <pre> String message = names + String(" : 연결에 성공했습니다!");</pre> </div> <div data-bbox="279 1966 657 2004" data-label="Text"> <pre> Serial.println(message);</pre> </div> <div data-bbox="901 1966 1414 2004" data-label="Text"> <p>// 시리얼에 연결에 성공했다고 출력</p> </div>

	<pre> SerialBT.println(message.c_str()); // 연결된 블루투스에 성공했다고 전송 } void loop(){ int sensor_result = analogRead(sensor); // 센서가 측정한 값 저장 String sensor_data = String(sensor_result, DEC); SerialBT.println(sensor_data); delay(1000); } </pre>
<div>동작과정</div>	<div> <div>  <p>① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드합니다.</p> </div> <div>  <p>② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해 줍니다.</p> </div> <div>  <p>③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다. (참고사항 ② 소스코드 테스트앱 확인)</p> </div> </div>

Available devices



5F:F2:0C:26:61:D1



4A:5D:6C:1D:15:C2



5C:3C:F8:8F:9F:A3



68:11:69:A5:A3:69



your-name

- ④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

Connect in



Controller mode



Switch mode



Dimmer mode



Terminal mode

- ⑤ **Terminal mode**에 연결합니다.

your-name

your-name: your-name : 연결에 성공했습니다!

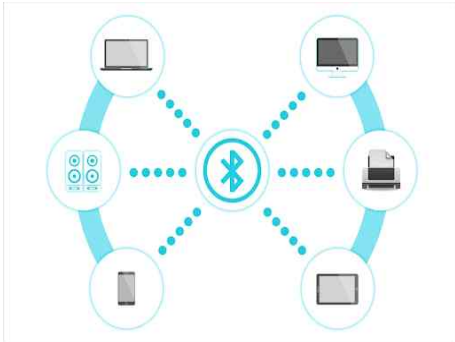





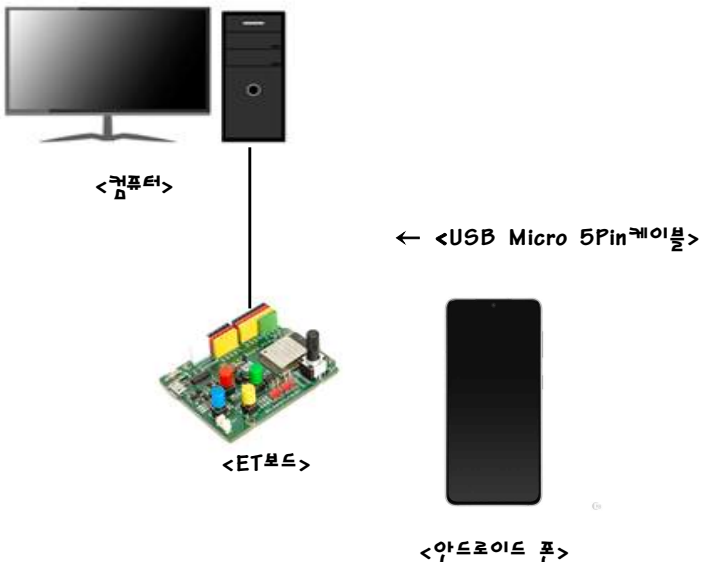
- ⑥ 연결 상태를 확인할 수 있습니다.

COM3

```
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
```

- ② 시리얼 모니터에서
연결상태를 확인할 수 있습니다.

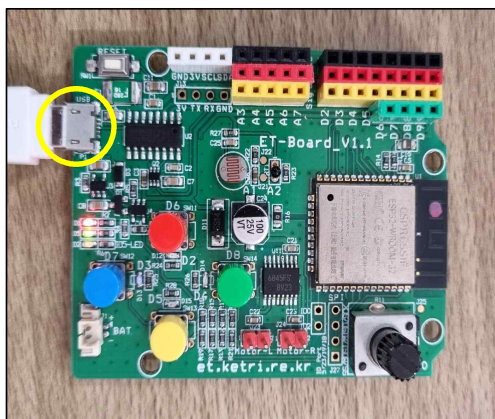
	<div data-bbox="306 210 799 633"> <div>your_name</div> <pre> your_name: 0 your_name: 0 your_name: 0 your_name: 0 your_name: 0 your_name: 0 your_name: 611 your_name: 829 your_name: 1194 your_name: 1351 your_name: 1582 your_name: 1674 your_name: 1783 </pre> </div> <div data-bbox="874 360 1469 483"> <p>③ 1초마다 블루투스로 가변저항의 값을 넘기는 걸 연결된 기기에서 확인할 수 있습니다. ※ 가변저항을 움직여 확인 가능</p> </div>
참 고 사 항	<p>① 회로</p> <ul style="list-style-type: none"> ○ 블루투스에 대한 참고사항 링크 https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4 https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124 <p>② 소스코드</p> <ul style="list-style-type: none"> ○ 블루투스 테스트 앱 https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal ○ 블루투스를 사용하는 방법 https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/ ○ esp32 블루투스 라이브러리 https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/

학습내용	블루투스를 이용하여 조도센서의 값을 앱에 보내기
소스	05._Bluetooth_communication_photoresistor_sensor.ino
개념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>

회로 구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스 코드

// 참고사항 @항목의 블루투스 테스트앱을 설치 후 진행합니다.

```
#include "BluetoothSerial.h"
```

```
BluetoothSerial SerialBT;
```

```
int sensor = A1;
```

// 조도센서

```
char names[] = "your_name";
```

// 블루투스의 이름 입력

```
void setup(){
```

```
  Serial.begin(115200);
```

// 통신속도 설정

```
  SerialBT.begin(names);
```

// 자신의 이름으로 된 블루투스 생성

```
  while(!SerialBT.connected(1000)){
```

```
    Serial.println("연결되지 않았습니다.");
```

```
  }
```

```
  delay(1000);
```

// 제대로 연결 하기위해 잠시 대기

```
  String message = names + String(" : 연결에 성공했습니다!");
```

```
  Serial.println(message);
```

// 시리얼에 연결에 성공했다고 출력

```

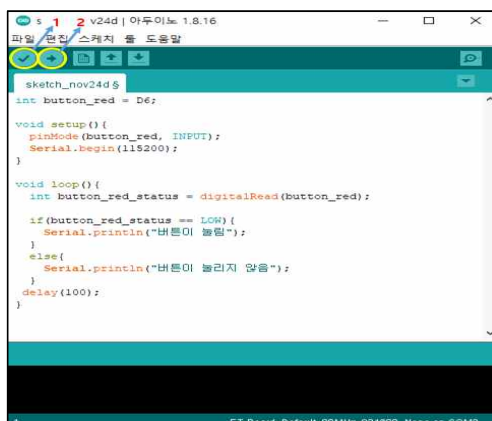
SerialBT.println(message.c_str());    // 연결된 블루투스에 성공했다고 전송
}

void loop(){
  int sensor_result = analogRead(sensor); // 센서가 측정한 값 저장
  String sensor_data = String(sensor_result, DEC);

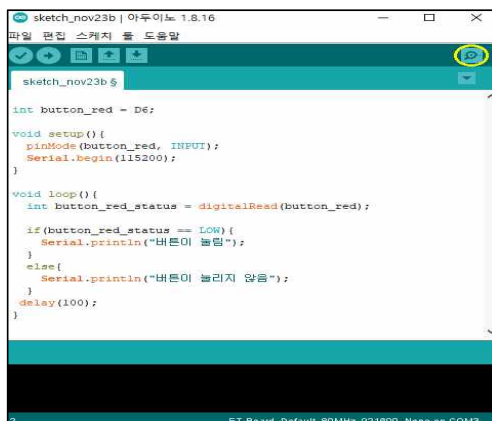
  SerialBT.println(sensor_data);
  delay(1000);
}

```

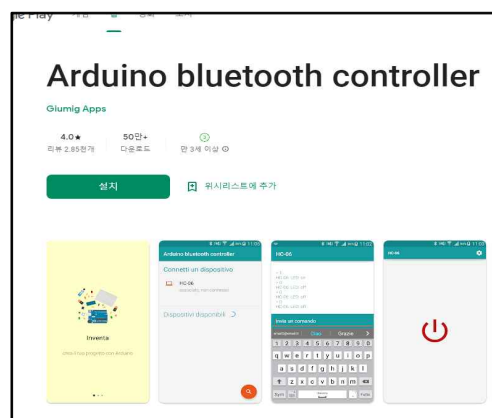
동작과정



① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드합니다.



② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해 줍니다.



③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다.
(참고사항 ② 소스코드 테스트앱 확인)

Available devices



5F:F2:0C:26:61:D1



4A:5D:6C:1D:15:C2



5C:3C:F8:8F:9F:A3



68:11:69:A5:A3:69



your-name

- ④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

Connect in



Controller mode



Switch mode



Dimmer mode



Terminal mode

- ⑤ **Terminal mode**에 연결합니다.

your-name

your-name: your-name : 연결에 성공했습니다!

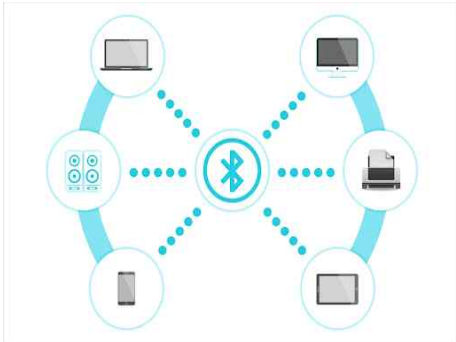





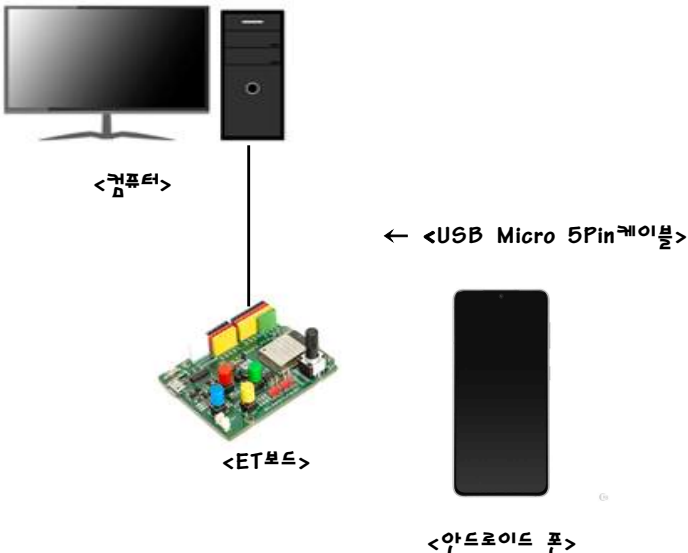
- ⑥ 연결 상태를 확인할 수 있습니다.

COM3

```
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
```

- ② 시리얼 모니터에서
연결상태를 확인할 수 있습니다.

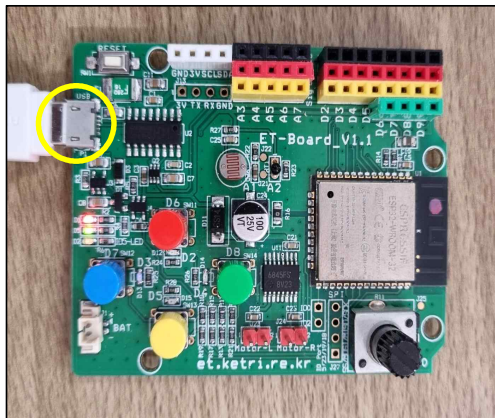
	<div data-bbox="306 215 799 629"> <div>your_name</div> <pre> your_name: 2621 your_name: 2513 your_name: 2539 your_name: 2174 your_name: 971 your_name: 571 your_name: 299 your_name: 319 your_name: 323 your_name: 213 your_name: 198 your_name: 141 </pre> </div> <div data-bbox="884 320 1477 524"> <p>③ 1초마다 블루투스로 조도센서의 값을 넘깁니다. 블루투스로 연결된 기기에서 확인할 수 있습니다. (※ 조도센서에 손가락을 붙였다 떼다 하면서 값 확인)</p> </div>
참 고 사 항	<p>① 회로</p> <ul style="list-style-type: none"> ○ 블루투스에 대한 참고사항 링크 https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4 https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124 <p>② 소스코드</p> <ul style="list-style-type: none"> ○ 블루투스 테스트 앱 https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal ○ 블루투스를 사용하는 방법 https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/ ○ esp32 블루투스 라이브러리 https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/

학습내용	블루투스를 이용해서 LED를 켜다 켜다 해보기
소스	06._Bluetooth_led_control.ino
개념	  <p>블루투스는 일상생활에서 흔하게 볼 수 있습니다.</p>
준비물	    <p><컴퓨터> <USB Micro 5Pin 케이블> <ET보드> <안드로이드 폰></p>
회로구성	 <p><컴퓨터> ← <USB Micro 5Pin 케이블> <안드로이드 폰></p> <p><ET보드></p>

회로 구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스 코드

// 참고사항 @항목의 블루투스 테스트앱을 설치 후 진행합니다.

// 시리얼 통신으로 제어

```
#include "BluetoothSerial.h"
```

```
#define led_red D2 // 빨강 LED
```

```
#define led_blue D3 // 파랑 LED
```

```
#define led_green D4 // 초록 LED
```

```
#define led_yellow D5 // 노랑 LED
```

```
BluetoothSerial SerialBT;
```

```
char names[] = "your_name"; // 블루투스의 이름 입력
```

```
void setup(){
```

```
    Serial.begin(115200); // 통신 속도
```

```
    pinMode(led_red, OUTPUT); // 핀 모드 설정
```

```
    pinMode(led_blue, OUTPUT); // 핀 모드 설정
```

```
    pinMode(led_green, OUTPUT); // 핀 모드 설정
```

```
    pinMode(led_yellow, OUTPUT); // 핀 모드 설정
```

```

SerialBT.begin(names);           // 자신의 이름을 적어주세요 !

while(!SerialBT.connected(1000)){
    Serial.println("연결되지 않았습니다.");
}
delay(1000);                     // 제대로 연결 하기위해 잠시 대기

String message = names + String(" : 연결에 성공했습니다!");
Serial.println(message);         // 시리얼에 연결에 성공했다고 출력
SerialBT.println(message.c_str()); // 연결된 블루투스에 성공했다고 전송
}

void loop(){
    char cmd = -1;

    if(SerialBT.available()){    // 입력받은 값을 저장하고 입력 받은 값 출력
        cmd = (char)SerialBT.read();
        Serial.print("입력값 = ");
        Serial.println(cmd);
    }

    if(cmd == '1'){              // 1 입력되었을 때 빨강 LED 켜기
        Serial.println ("빨강 온 !!!!");
        digitalWrite(led_red, HIGH);
    }
    else if(cmd == '2'){         // 2 입력되었을 때 빨강 LED 끄기
        Serial.println("빨강 오프 !!");
        digitalWrite(led_red, LOW);
    }
    else if(cmd == '3'){         // 3 입력되었을 때 파랑 LED 켜기
        Serial.println("파랑 온 !!!!");
        digitalWrite(led_blue, HIGH);
    }
    else if(cmd == '4'){         // 4 입력되었을 때 파랑 LED 끄기
        Serial.println("파랑 오프 !!");
        digitalWrite(led_blue, LOW);
    }
    else if(cmd == '5'){         // 5 입력되었을 때 초록 LED 켜기
        Serial.println("초록 온 !!!!");
    }
}

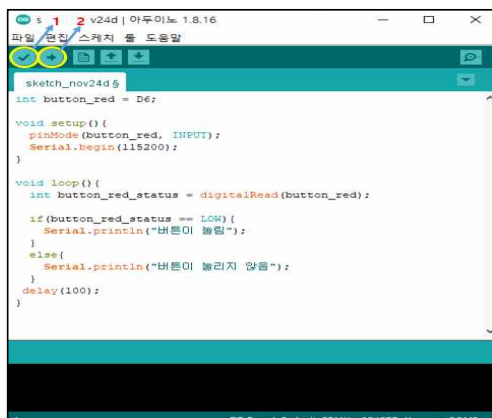
```

```

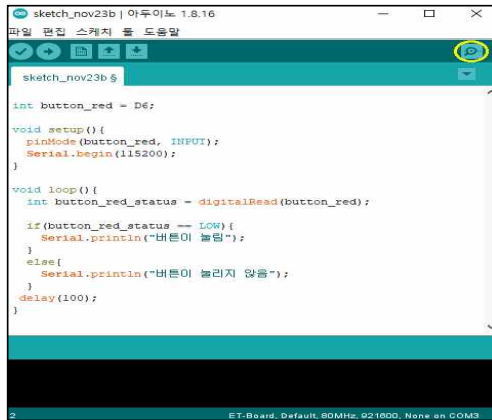
digitalWrite(led_green, HIGH);
}
else if(cmd == '6'){ // 6 입력되었을 때 초록 LED 끄기
    Serial.println("초록 오프 !!");
    digitalWrite(led_green, LOW);
}
else if(cmd == '7'){ // 7 입력되었을 때 노랑 LED 켜기
    Serial.println("노랑 온 !!!!");
    digitalWrite(led_yellow, HIGH);
}
else if(cmd == '8'){ // 9 입력되었을 때 노랑 LED 끄기
    Serial.println("노랑 오프 !!");
    digitalWrite(led_yellow, LOW);
}
else if(cmd == '9')
{
    Serial.println("모두 오프 !!");
    digitalWrite(led_red, LOW);
    digitalWrite(led_blue, LOW);
    digitalWrite(led_green, LOW);
    digitalWrite(led_yellow, LOW);
}
else
{
    //그 외에는 아무것도 하지 않음
}
delay(20); // 0.02초 대기
}

```

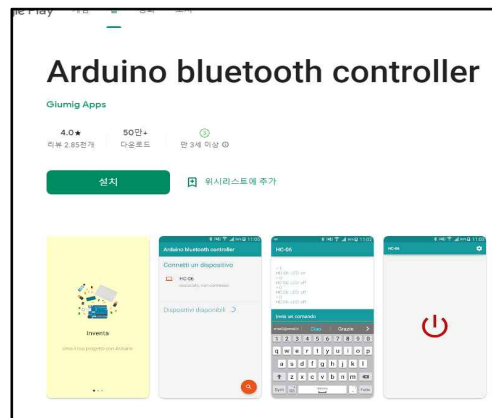
동작 과정



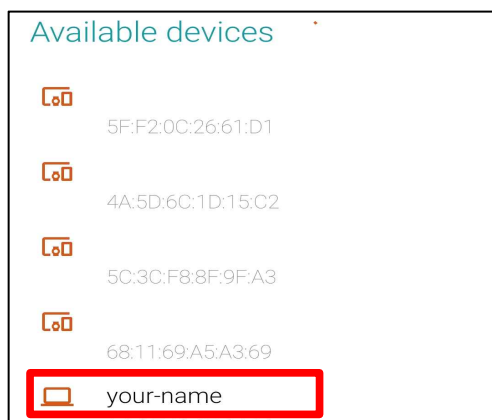
- ① 아두이노 코드입력 후 상단의 1[확인]아이콘을 클릭하고 완료되면 2[업로드] 아이콘도 클릭해서 펌웨어를 ET보드에 업로드합니다.



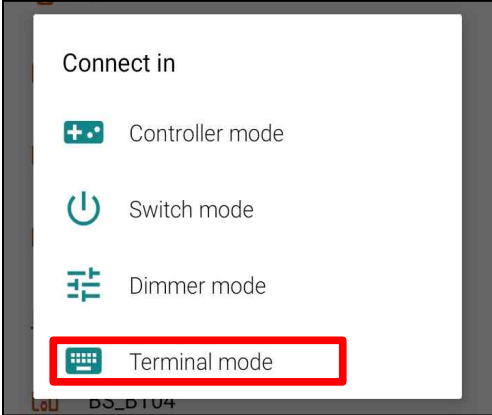
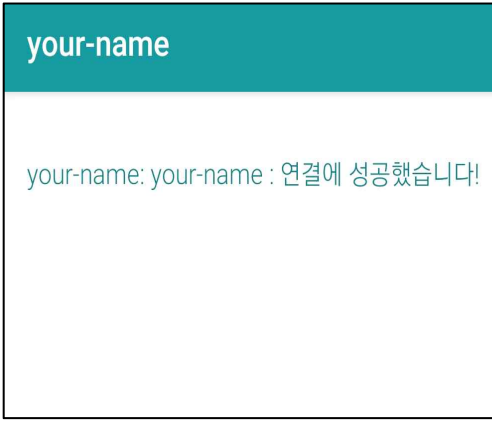
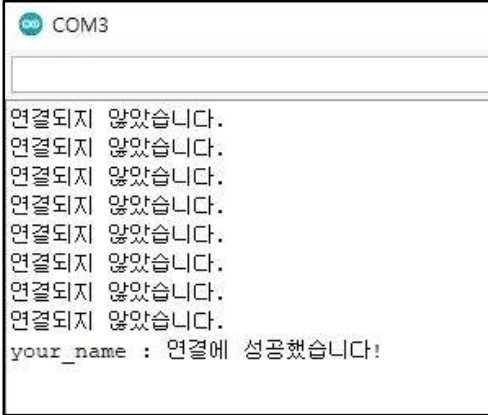
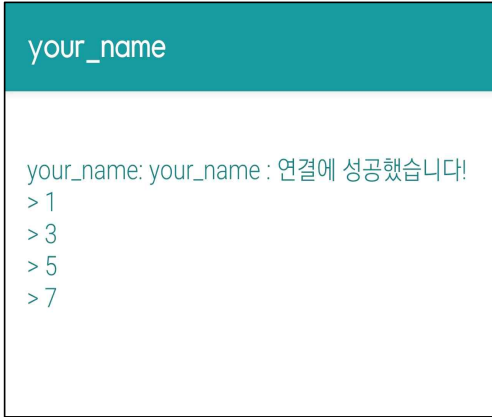
② 업로드까지 마치면 우측상단 돋보기 모양의 “시리얼모니터”를 클릭 해 줍니다.



③ 안드로이드 핸드폰 아두이노 블루투스 controller 앱 설치합니다.
(참고사항 ② 소스코드 테스트앱 확인)



④ **your-name**을 선택합니다.
(입력한 블루투스 이름 선택)

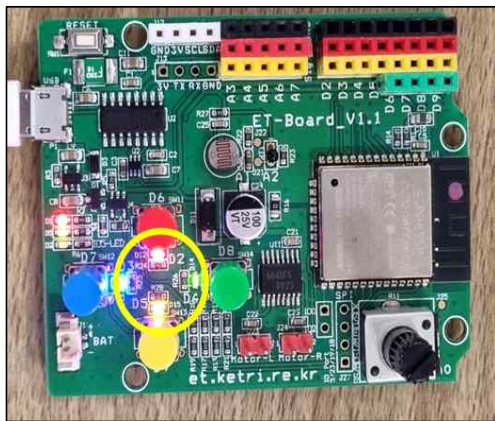
		<p>⑤ Terminal mode에 연결합니다.</p>
동작 과정		<p>⑥ 연결 상태를 확인할 수 있습니다.</p>
		<p>② 시리얼 모니터에서 연결상태를 확인할 수 있습니다.</p>
		<p>③ 블루투스로 값을 보내면 코드가 작동합니다.</p> <ul style="list-style-type: none"> 1 = 빨간 LED On 2 = 빨간 LED Off 3 = 파란 LED On 4 = 파란 LED Off 5 = 초록 LED On 6 = 초록 LED Off 7 = 노란 LED On 8 = 노란 LED Off 9 = 모든 LED Off


```

COM3

연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
연결되지 않았습니다.
your_name : 연결에 성공했습니다!
입력값 = 1
빨강 온 !!!!
입력값 = 3
파랑 온 !!!!
입력값 = 5
초록 온 !!!!
입력값 = 7
노랑 온 !!!!

```



① 회로

- 블루투스에 대한 참고사항 링크

<https://ko.wikipedia.org/wiki/%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4>

<https://m.post.naver.com/viewer/postView.nhn?volumeNo=18826538&memberNo=44155124>

② 소스코드

- 블루투스 테스트 앱

<https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor>

https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal

- 블루투스를 사용하는 방법

<https://randomnerdtutorials.com/esp32-bluetooth-classic-arduino-ide/>

- esp32 블루투스 라이브러리

<https://www.arduino.cc/reference/en/libraries/esp32-ble-arduino/>

참 고
사 항