CNRLAB

CIM&ROBOTICS LABORATORY

로봇공학입문설계

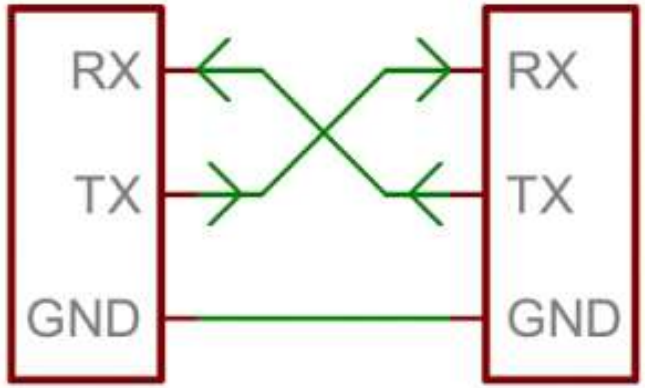
10주차 모바일 로봇(4)

로봇공학과

시리얼 통신

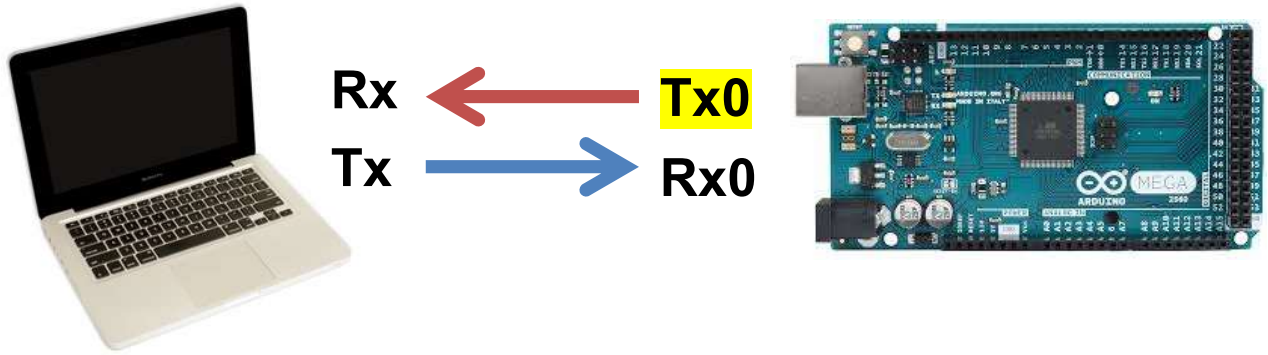
❑ 연결방법

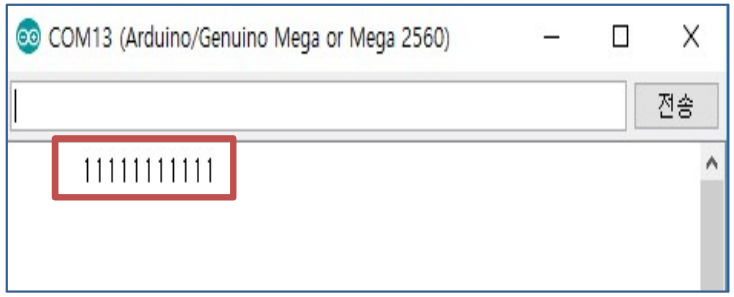
- ❍ 시리얼 모듈은 각각 Tx(송신), RX(수신), GND의 세가지 라인으로 구성
- ❍ RX와 TX라인을 이용하여 두 모듈사이에 통신이 이루어짐
- ❍ RX, TX라는 이름은 각 장치 자신의 입장에서 바라봤을 때 라인이 담당하는 역할이므로 두 모듈을 연결할 때에는 RX와 TX를 서로 엇갈리게 연결



시리얼 통신

❑ Arduino ↔ PC

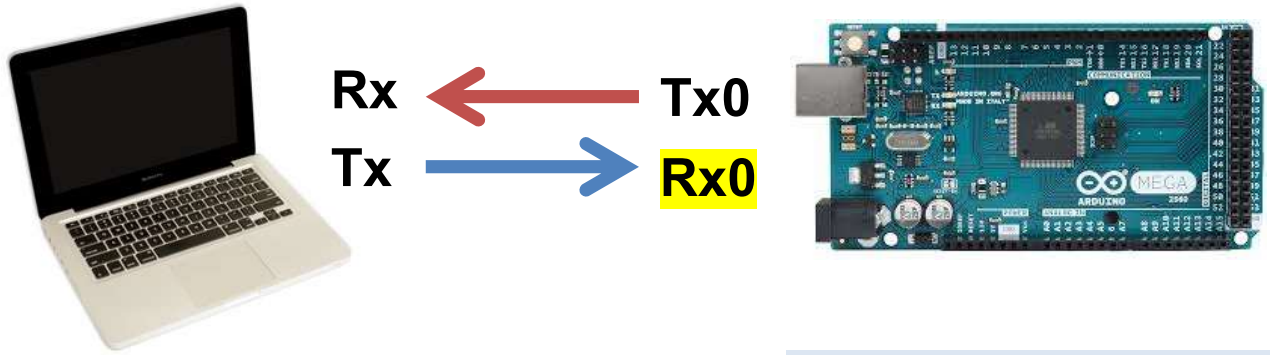


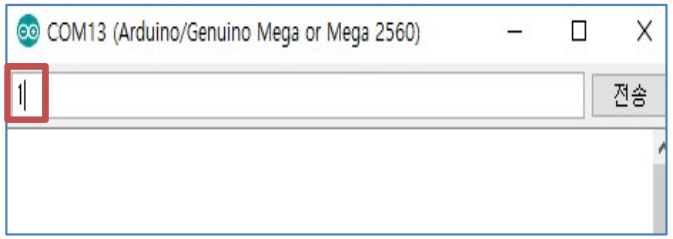



```
void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.print(1);
  delay(1000);
}
```

시리얼 통신

❑ Arduino ↔ PC

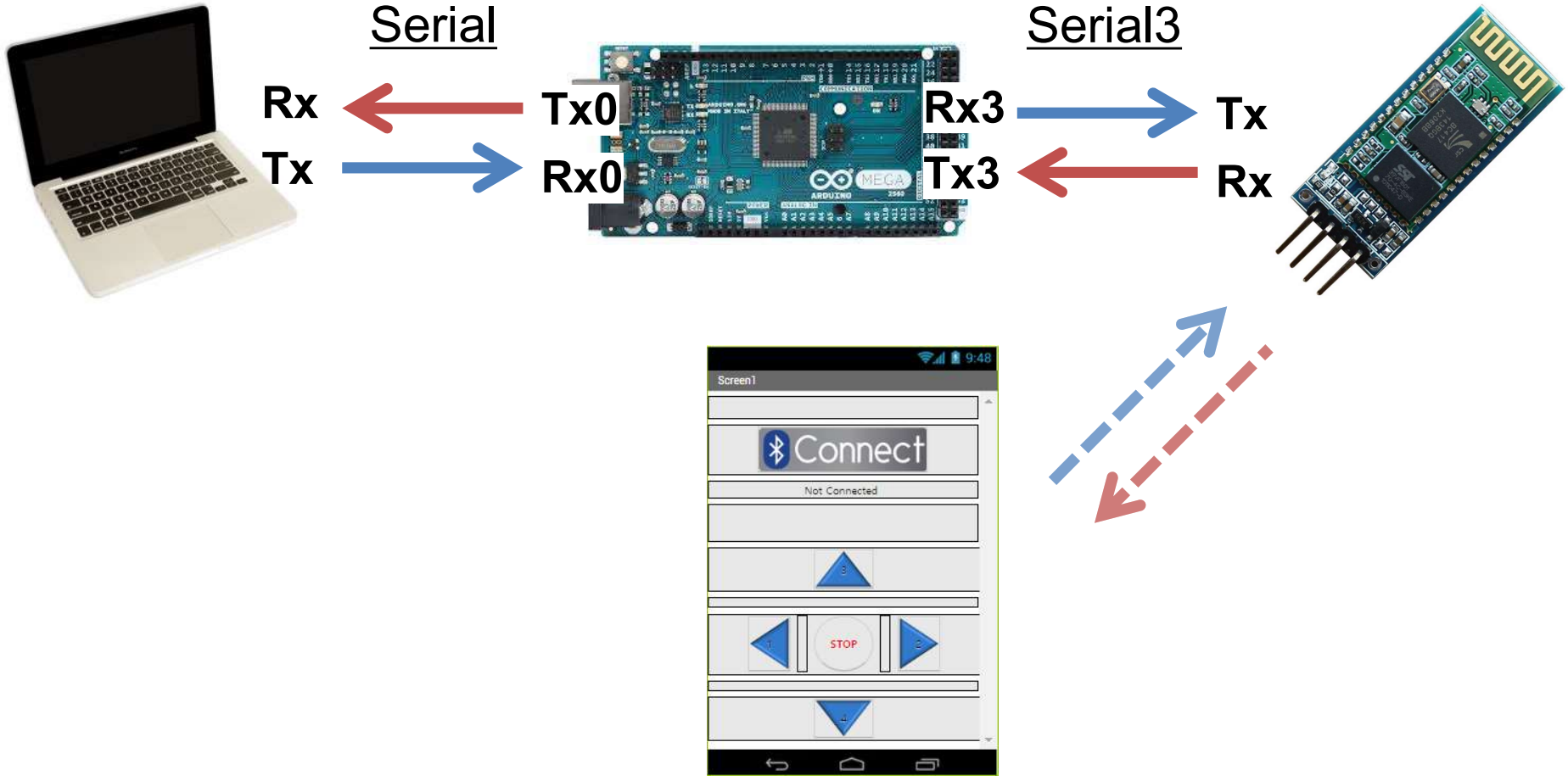





```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int val = Serial.read();
}
```

블루투스 통신

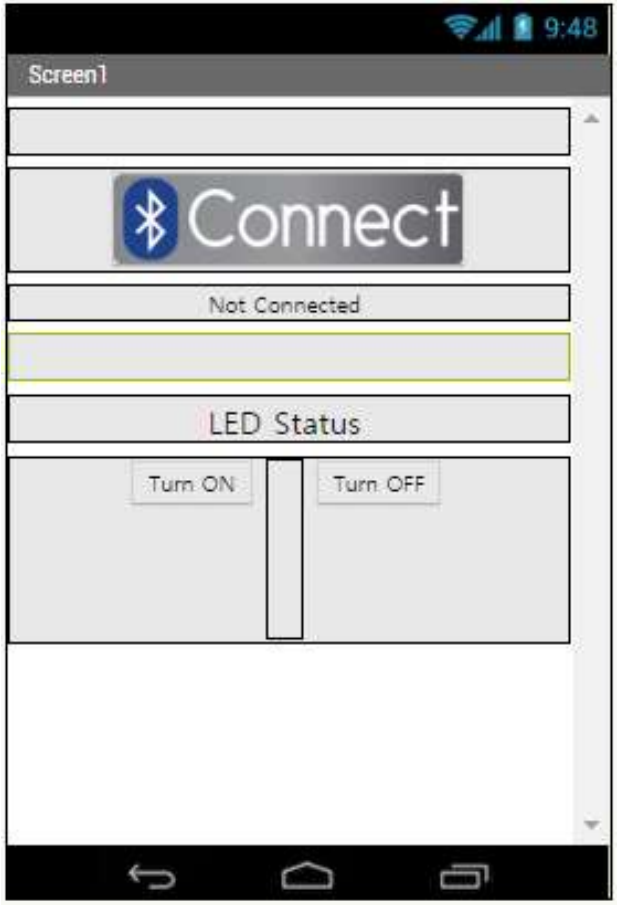
❑ 블루투스 통신 구조



MIT App Inventor

❑ 프로그램 구성

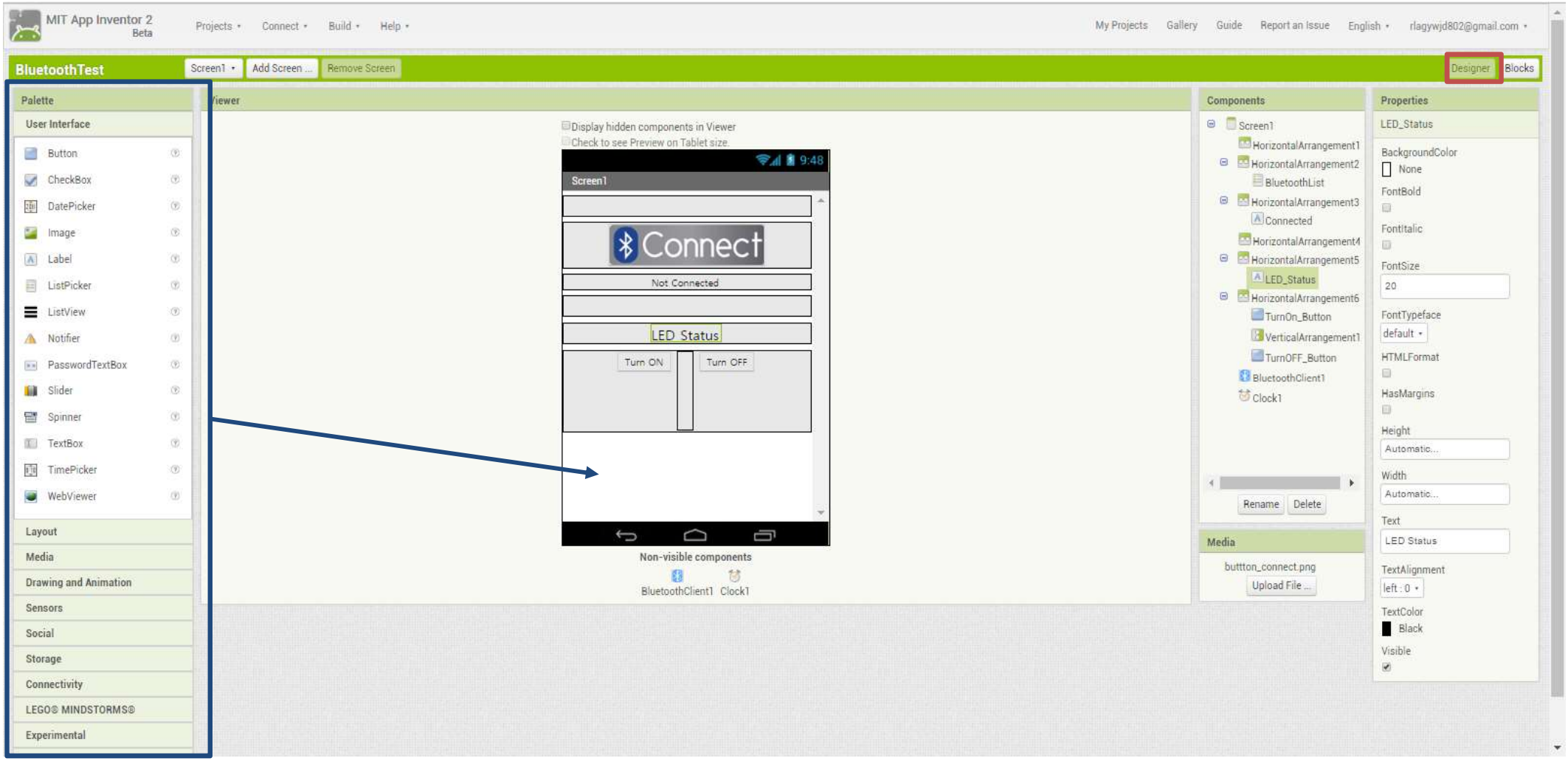
Designer



Blocks

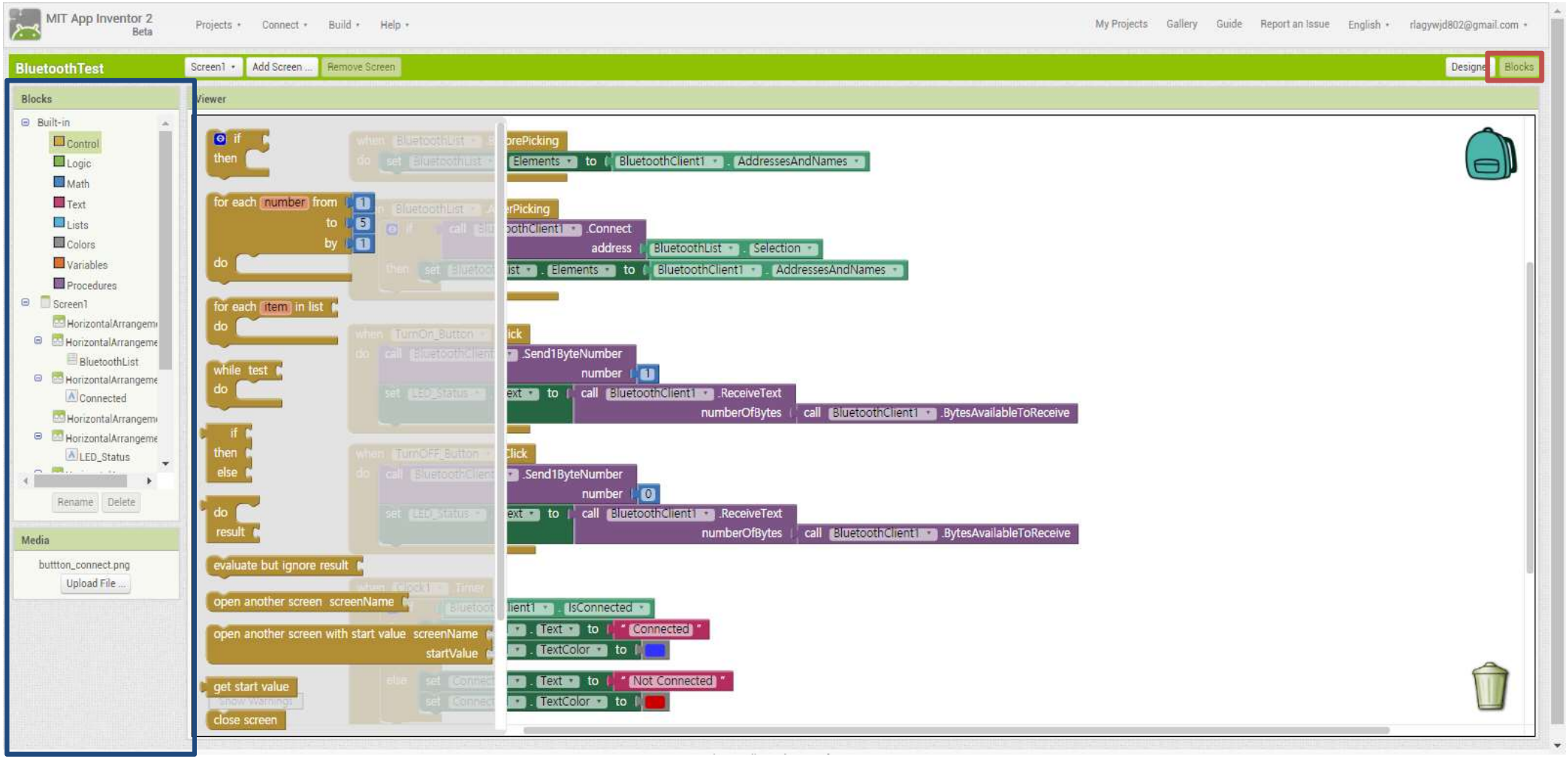
MIT App Inventor

❑ Designer



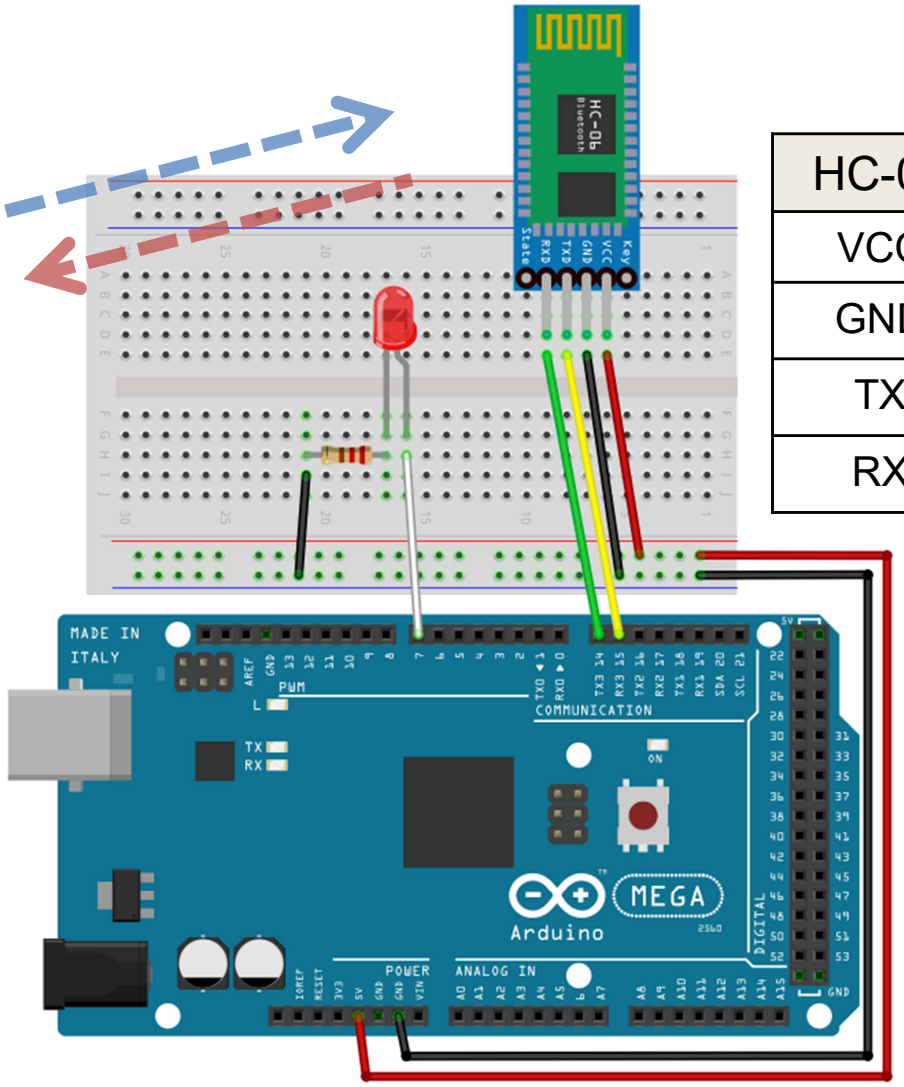
MIT App Inventor

❑ Blocks



블루투스 통신

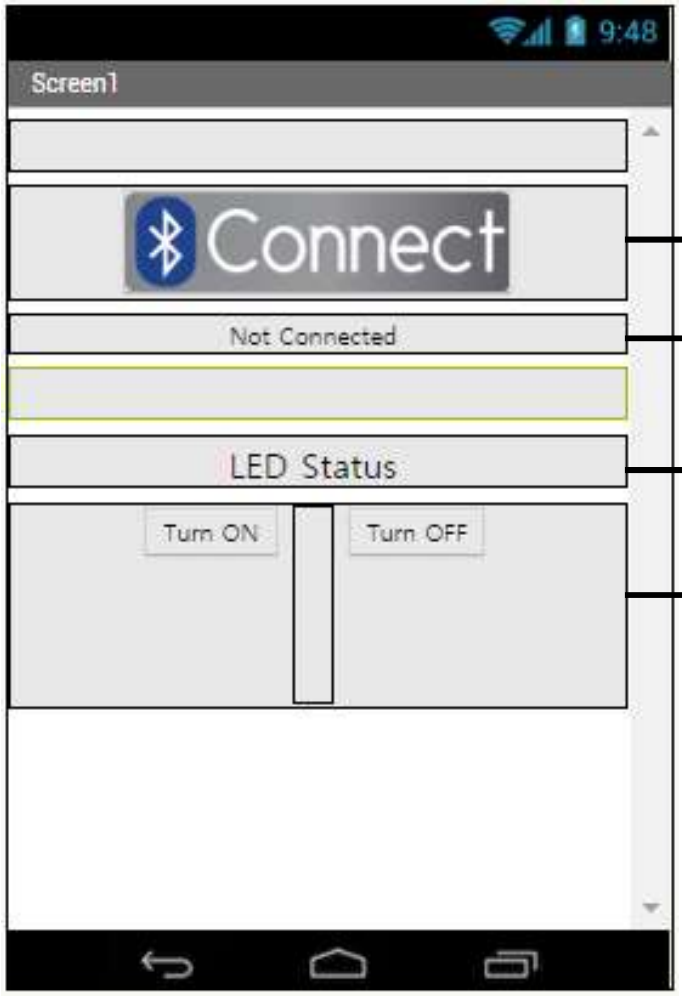
[예제1] Bluetooth로 LED키기



HC-06	Arduino
VCC	5V
GND	GND
TX	RX3(15)
RX	TX3(16)

블루투스 통신

❑ MIT App Inventor



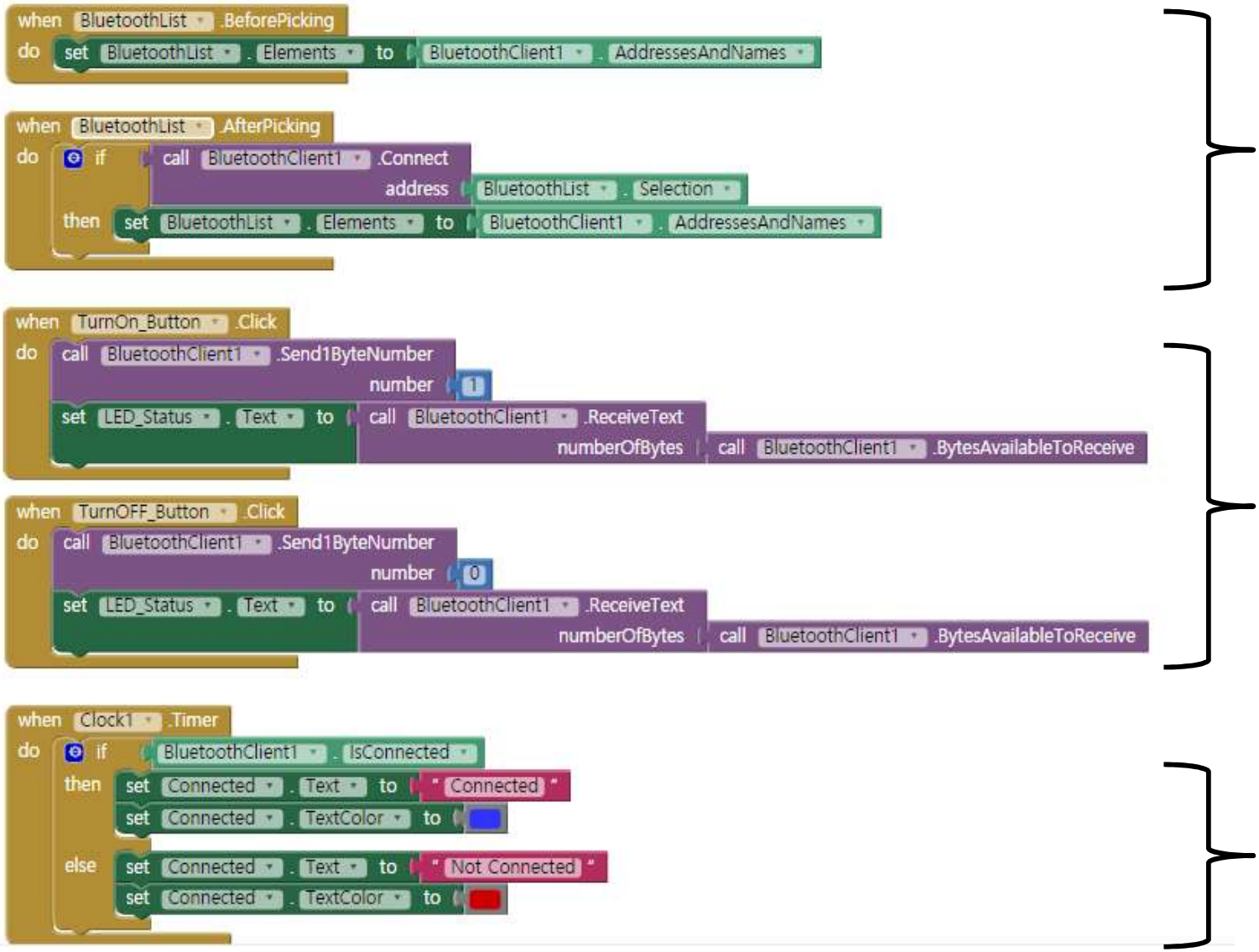
→ 블루투스 연결부

→ 블루투스 통신 연결상태 표시

→ LED 상태 표시

→ LED 점멸 조작 버튼

블루투스 통신

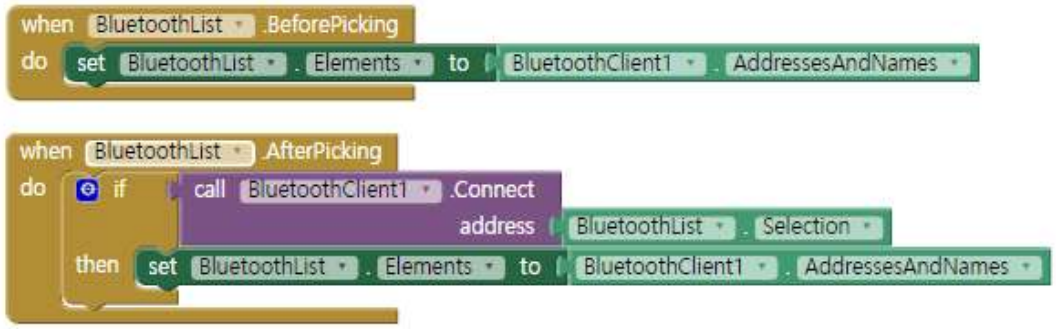


블루투스 연결부

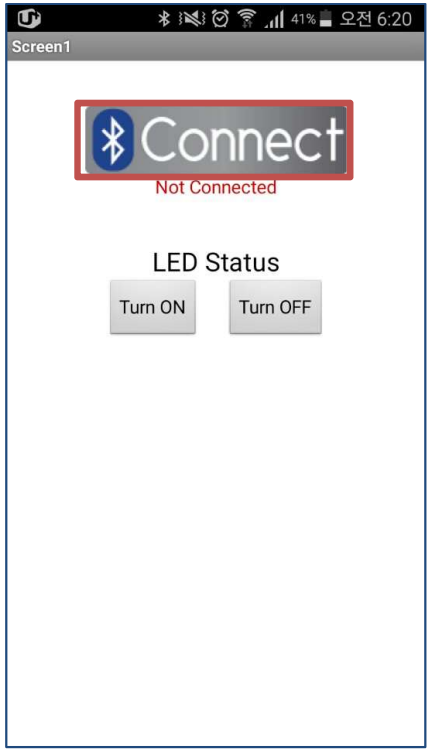
버튼 조작부

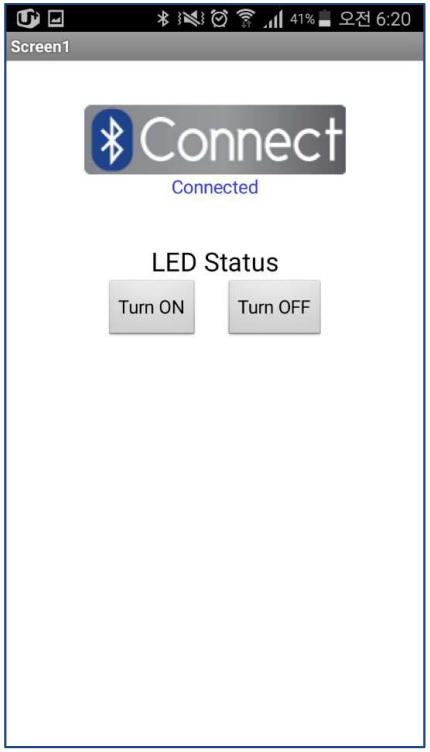
타이머

블루투스 통신

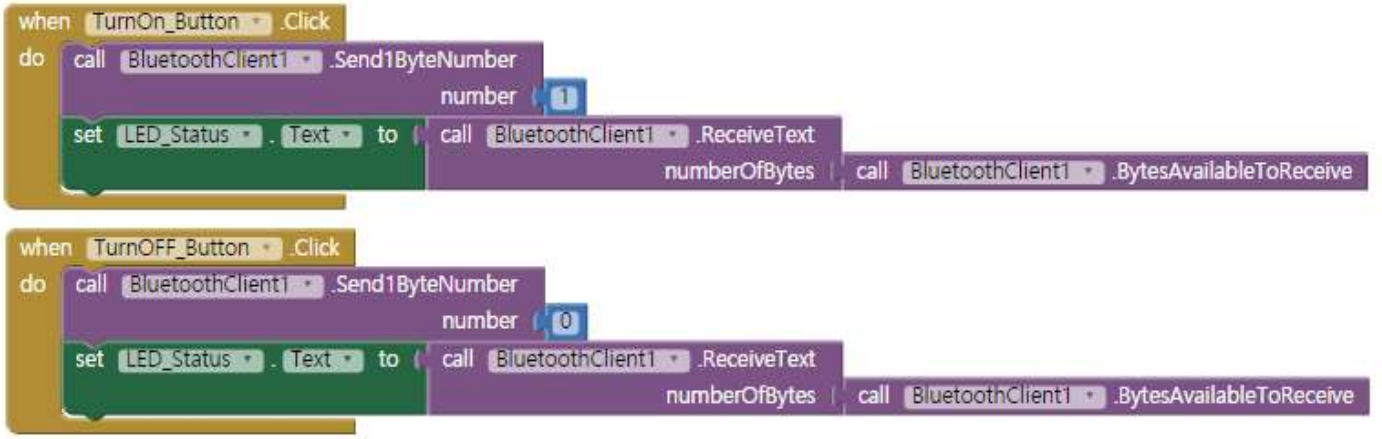


블루투스 연결부





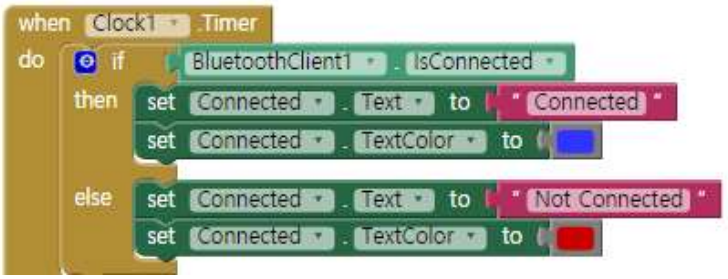
블루투스 통신



버튼 조작부


```
void loop() {
  if(Serial3.available()){
    indata = Serial3.read();

    if(indata == 0){
      digitalWrite(ledPin, LOW);
      Serial.println("LED:OFF");
    }
    else if(indata == 1){
      digitalWrite(ledPin, HIGH);
      Serial.println("LED:ON");
    }
    else{
      Serial3.println("Error");
      Serial.println(indata);
    }
}
```

블루투스 통신


when Clock1 .Timer
do if BluetoothClient1 . IsConnected
then set Connected . Text to " Connected "
set Connected . TextColor to
else set Connected . Text to " Not Connected "
set Connected . TextColor to

타이머

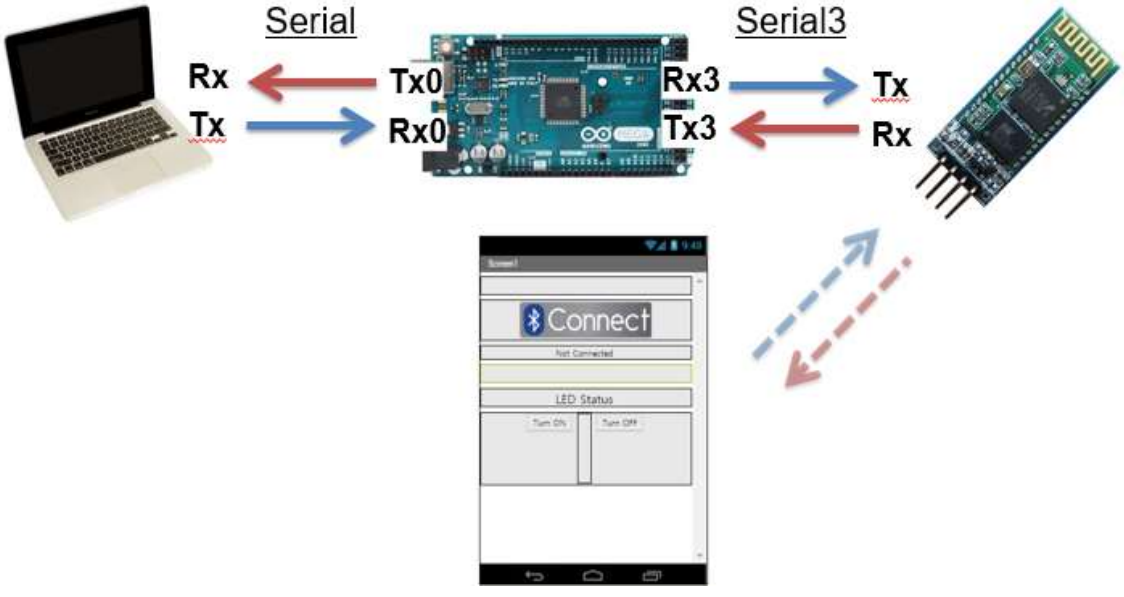

Screen1
Connect
Not Connected
LED Status
Turn ON
Turn OFF


Screen1
Connect
Connected
LED Status
Turn ON
Turn OFF

블루투스 통신

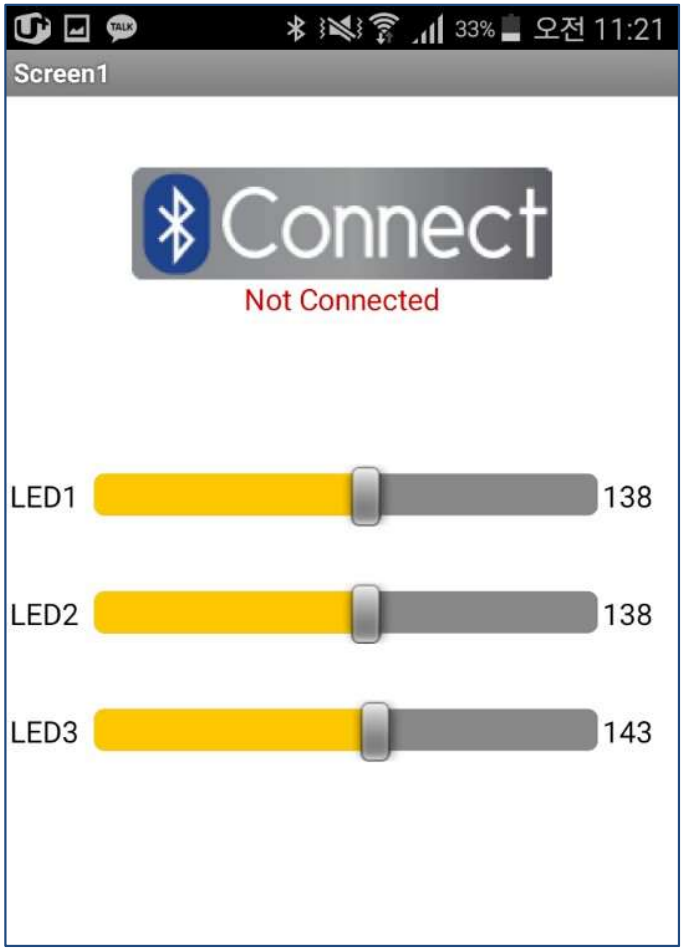
❑ Arduino

```
#define ledPin  7
unsigned char indata; // 1Byte Number

void setup() {
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);
  Serial.begin(9600);
  Serial3.begin(9600); // Default baud rate of the Bluetooth
}

void loop() {
  if(Serial3.available()){
    indata = Serial3.read();

    if(indata == 0){
      digitalWrite(ledPin, LOW);
      Serial.println("LED:OFF");
    }
    else if(indata == 1){
      digitalWrite(ledPin, HIGH);
      Serial.println("LED:ON");
    }
    else{
      Serial3.println("Error");
      Serial.println(indata);
    }
  }
}
```



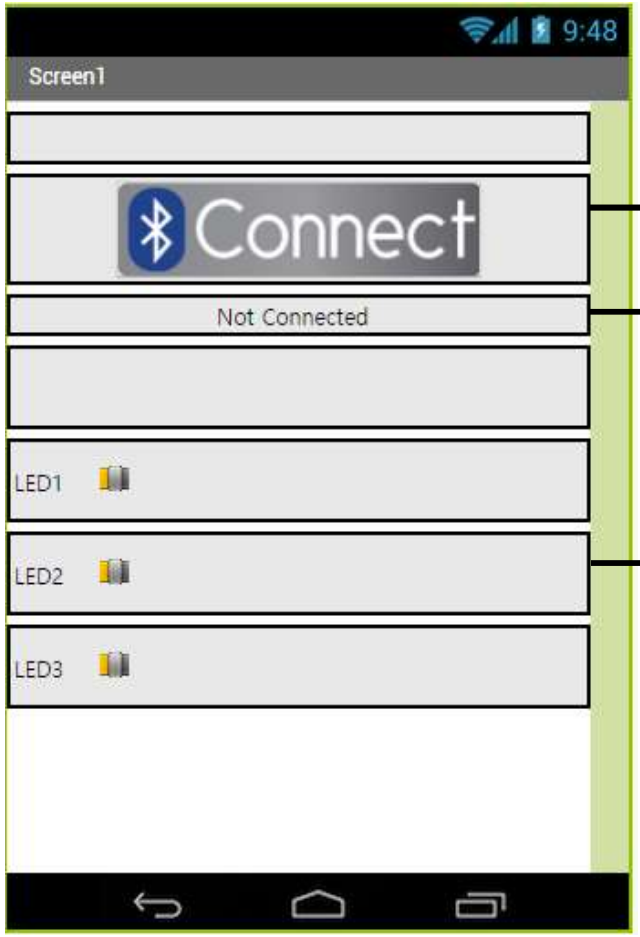
블루투스 통신

[예제2] Bluetooth로 여러 개 LED 밝기조절



블루투스 통신

❑ MIT App Inventor



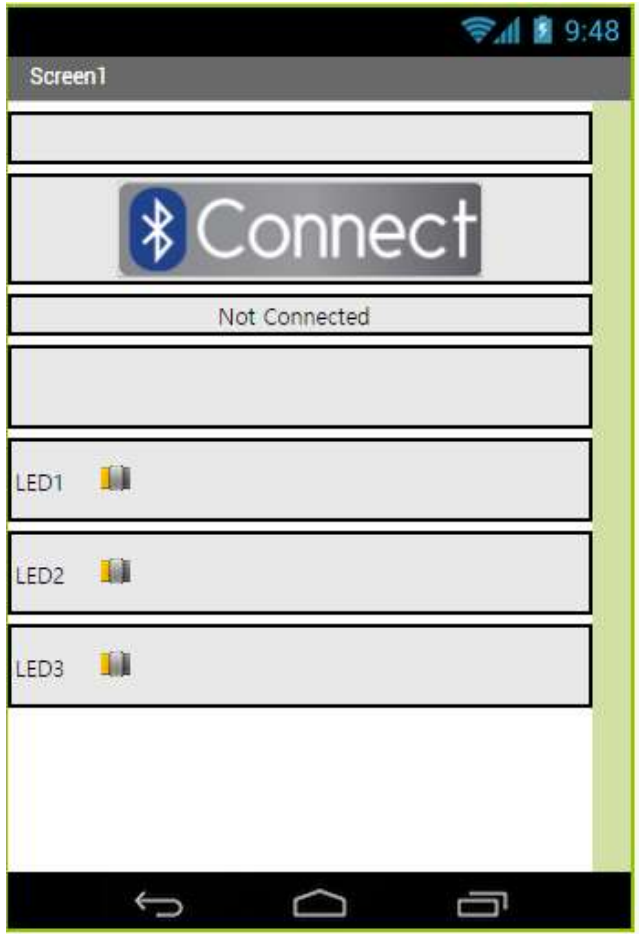
→ 블루투스 연결부

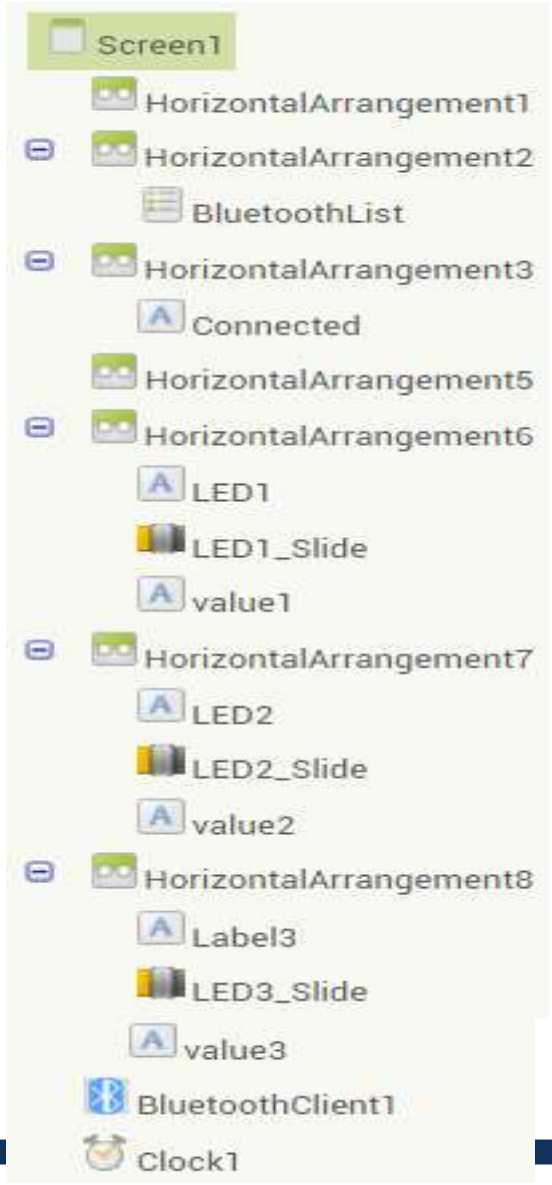
→ 블루투스 통신 연결상태 표시

→ LED 밝기 조절 및 표시

블루투스 통신

❑ MIT App Inventor



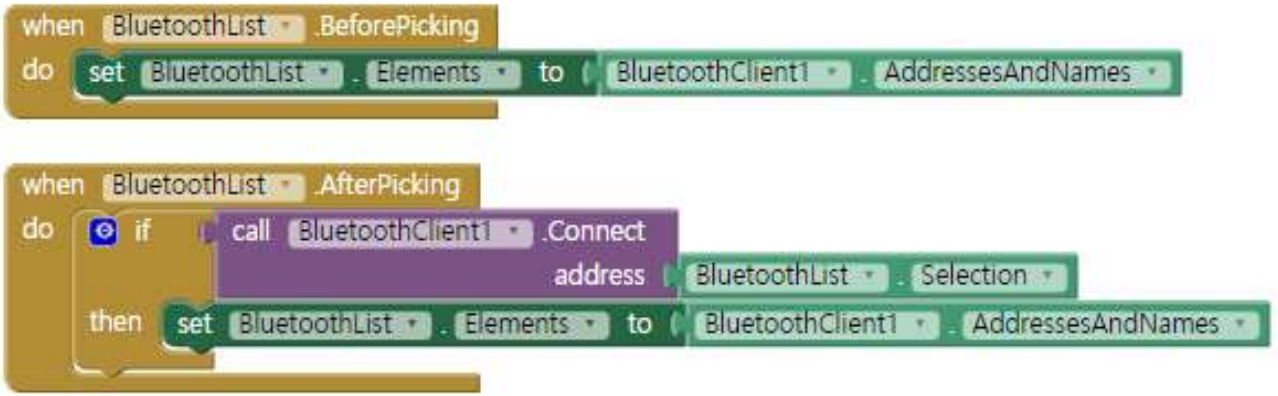


- Components 부분을 사진과 같이 구성

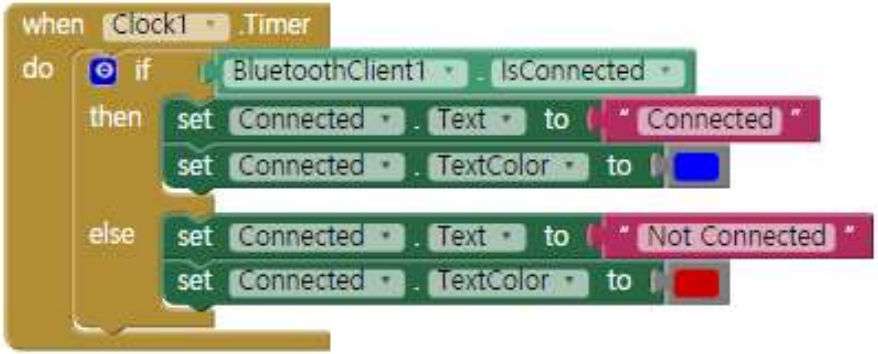
- Properties 부분은 임의로 구성

블루투스 통신

❑ MIT App Inventor

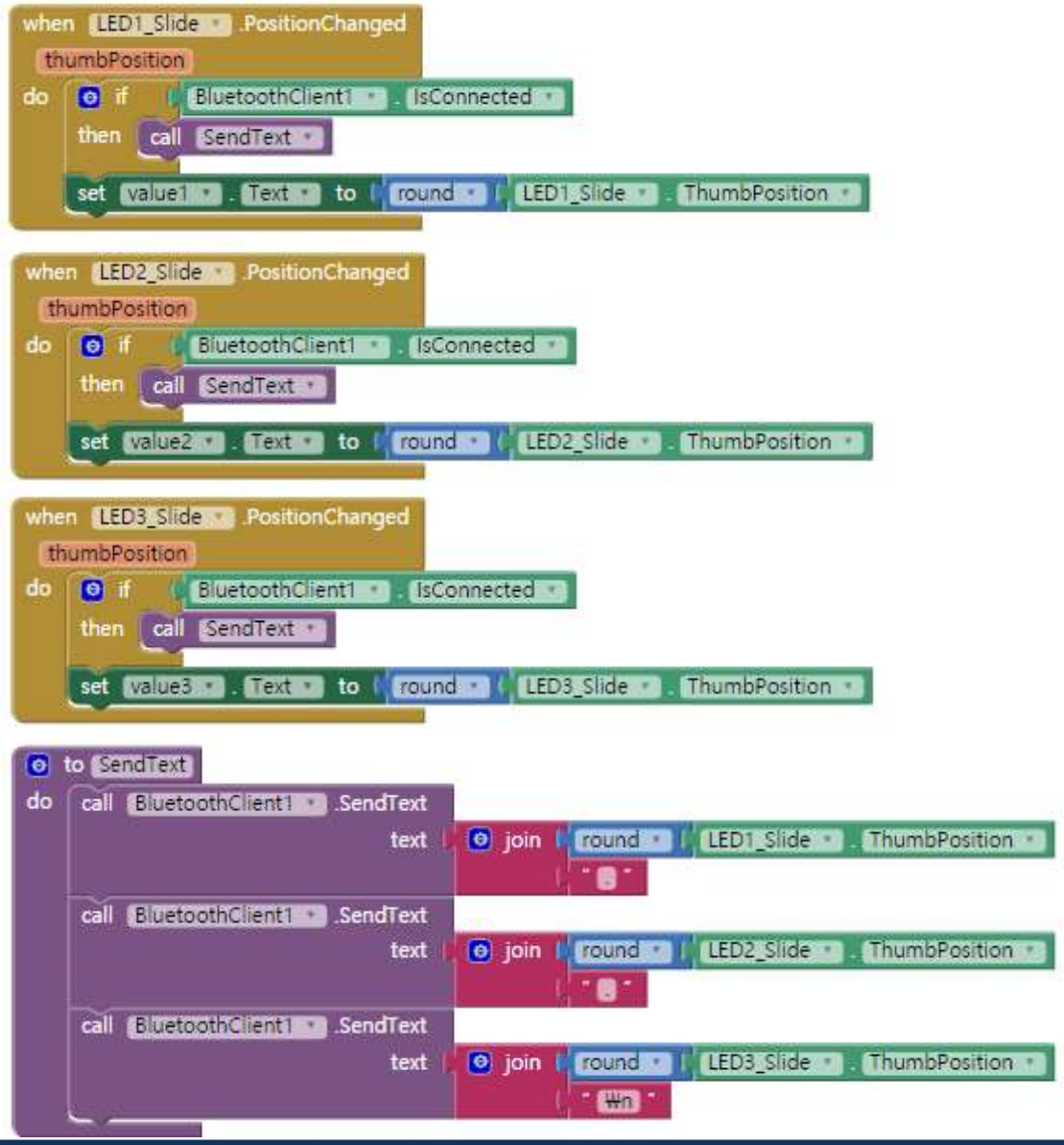


블루투스 연결부



타이머

블루투스 통신



슬라이드 조작부

함수부

블루투스 통신

❑ Arduino

```
#define ledPin1  7
#define ledPin2  8
#define ledPin3  9

void setup() {
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(ledPin3, OUTPUT);
  Serial3.begin(9600);
}

void loop() {
  if(Serial3.available()){
    int val1 = Serial3.parseInt();
    int val2 = Serial3.parseInt();
    int val3 = Serial3.parseInt();
    if(Serial3.read() == '\n'){
      analogWrite(ledPin1, val1);
      analogWrite(ledPin2, val2);
      analogWrite(ledPin3, val3);
    }
  }
}
```