

NS-RX231를 위한 소프트웨어가이드

<블루투스>

목차

| | |
|-------------------|---|
| 1 프로젝트 가져오기 | 2 |
| 2 개요 | 3 |
| 3 소스코드 | 4 |
| 3.1 송신 | 4 |
| 3.2 수신 | 5 |
| 4 디버깅 | 6 |
| 5 실행 | 7 |
| 6 회로도 | 9 |

1 프로젝트 가져오기

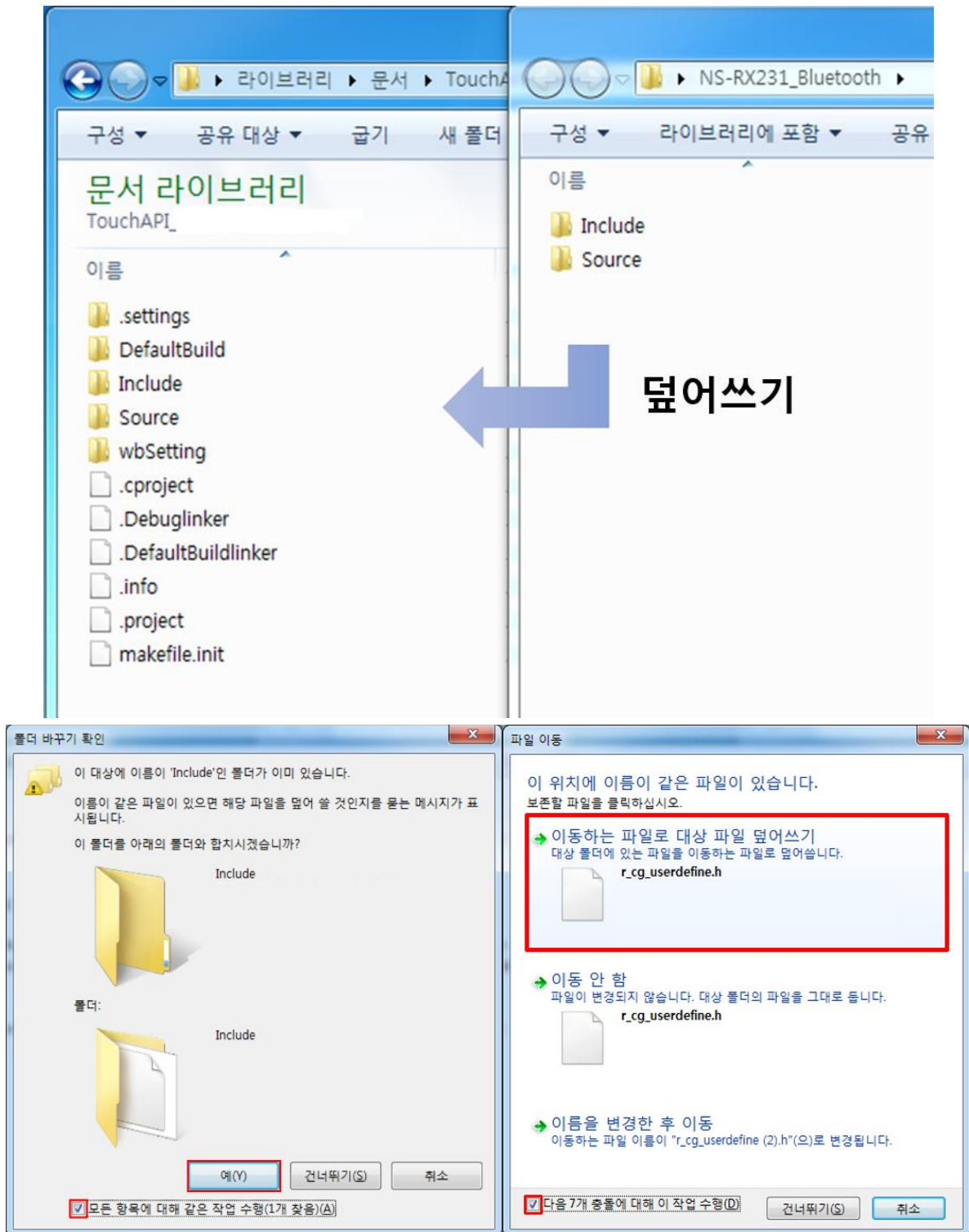


그림 1-1 소스파일 덮어쓰기

첨부된 소스파일을 Workbench6 First step guide 마법사로 만든 프로젝트에 덮어쓰기 한 후, e2studio에서 실행합니다.

2 개요

블루투스 통신을 위해, HC-06모듈을 사용했습니다.

그리고, 다른 모듈과 혼선을 막기 위해 터미널프로그램에 AT 명령어를 이용해 블루투스 모듈의 설정을 바꿔보았습니다.

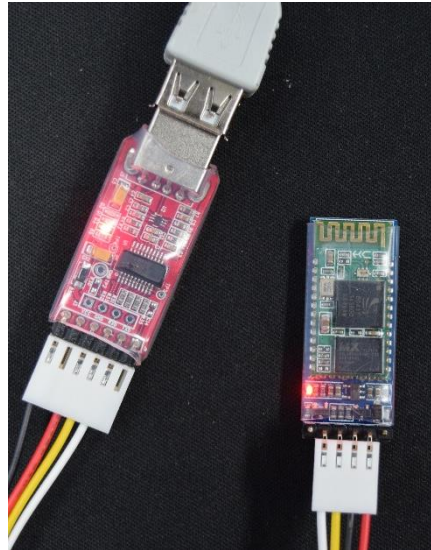


그림 2-1 HC-06 블루투스 모듈과 컴퓨터 연결

<https://sites.google.com/site/terminalbpp/terminal20141030.zip?attredirects=0&d=1>

Feedback

| Command | Feedback |
|--|-----------|
| Communication Test AT | OK |
| Change Baud rate AT+BAUD4 | OK9600 |
| Change Bluetooth device name AT+NAMEname | Oksetname |
| Change Pincode(password) AT+PINxxxx | OKsetPIN |

Input

Command

```

AT
AT+BAUD4
AT+NAMEY231
AT+PIN1234
    
```

Connected Rx: 25 Tx: 39 Rx OK

그림 2-2 터미널 프로그램

3 소스코드

3.1 송신

```

if ( _0_SUCCESS == R_Set_Cap_Touch_Result_Create( method ) )
{
    ts_result = R_Get_Cap_Touch_Result( method );

    if ( 1 == (ts_result.button[0] >> 15))
    {
        sci1_print(&send_data[0],1);
        PORTB.PODR.BYTE = 0x01;
    }
    else if(1 == (ts_result.button[1] >> 0))
    {
        sci1_print(&send_data[1],1);
        PORTB.PODR.BYTE = 0x02;
    }
    else if(1 == (ts_result.button[1] >> 6))
    {
        sci1_print(&send_data[2],1);
        PORTB.PODR.BYTE = 0x08;
    }
    else if(1 == (ts_result.button[1] >> 7))
    {
        sci1_print(&send_data[3],1);
        PORTB.PODR.BYTE = 0x20;
    }
    else if(1 == (ts_result.button[1] >> 11))
    {
        sci1_print(&send_data[4],1);
        PORTB.PODR.BYTE = 0x40;
    }
    else if(1 == (ts_result.button[1] >> 14))
    {
        sci1_print(&send_data[5],1);
        PORTB.PODR.BYTE = 0x80;
    }
}
    
```

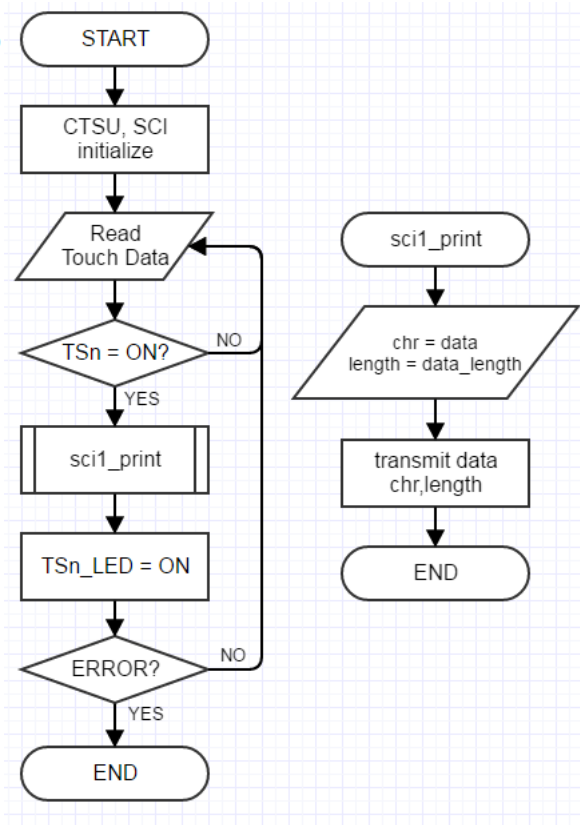


그림 3-1 소스코드 및 순서도

위 소스코드는 조건 식에 따라 검출된 터치버튼에 대응하는 LED 를 점등하고, 해당하는 터치버튼의 번호(ex:TS15 = 15)를 sci1(UART 통신)를 통해 전송하는 동작을 수행합니다.

3.2 수신

```

static void r_sci1_callback_receiveend(void) {
    /* Modifiable code [start] */
    switch (g_rx_char[0]) {
        case 1:
            PORTB.PODR.BYTE = 0x00; //reset
            break;
        case 15:
            PORTB.PODR.BYTE = 0x01; //TS15
            break;
        case 16:
            PORTB.PODR.BYTE = 0x02; //TS16
            break;
        case 22:
            PORTB.PODR.BYTE = 0x08; //TS22
            break;
        case 23:
            PORTB.PODR.BYTE = 0x20; //TS23
            break;
        case 27:
            PORTB.PODR.BYTE = 0x40; //TS27
            break;
        case 30:
            PORTB.PODR.BYTE = 0x80; //TS30
            break;
        default:
            break;
    }
    /* Modifiable Code [end] */

    R_SCI1_Serial_Receive((uint8_t *)&g_rx_char, 1);
    sci1_rxdone = 0;
}

```

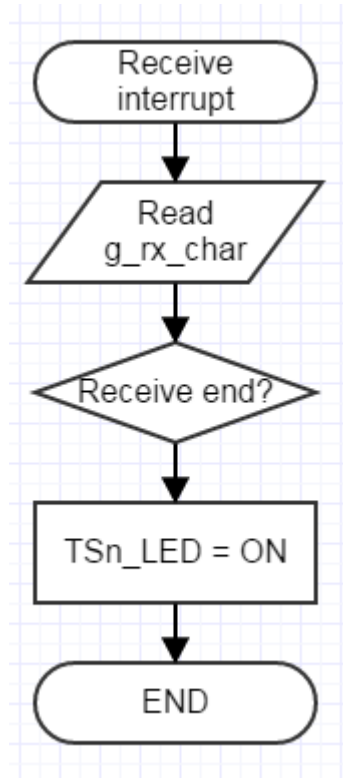


그림 3-2 소스코드 및 순서도

수신 인터럽트가 발생하여 지정된 수의 데이터 수신이 완료되면, 수신받은 데이터의 처리를 위해 그림 3-2의 프로그램이 실행됩니다.

데이터 1을 수신 한 경우에는 LED를 모두 OFF하고 15을 수신했을 때에는 PortB의 0번을 제어해 TS15에 대응하는 LED만 점등하고, 23을 수신했을 때에는 PortB의 5번을 제어해 TS23에 해당하는 LED만 점등합니다.

4 디버깅

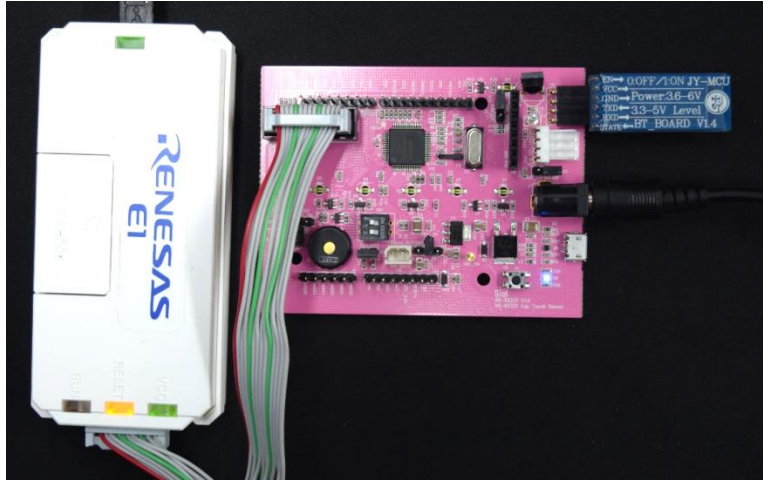


그림 4-1 NS-RX231을 전원 어댑터와 E1디버거를 연결한 모습.

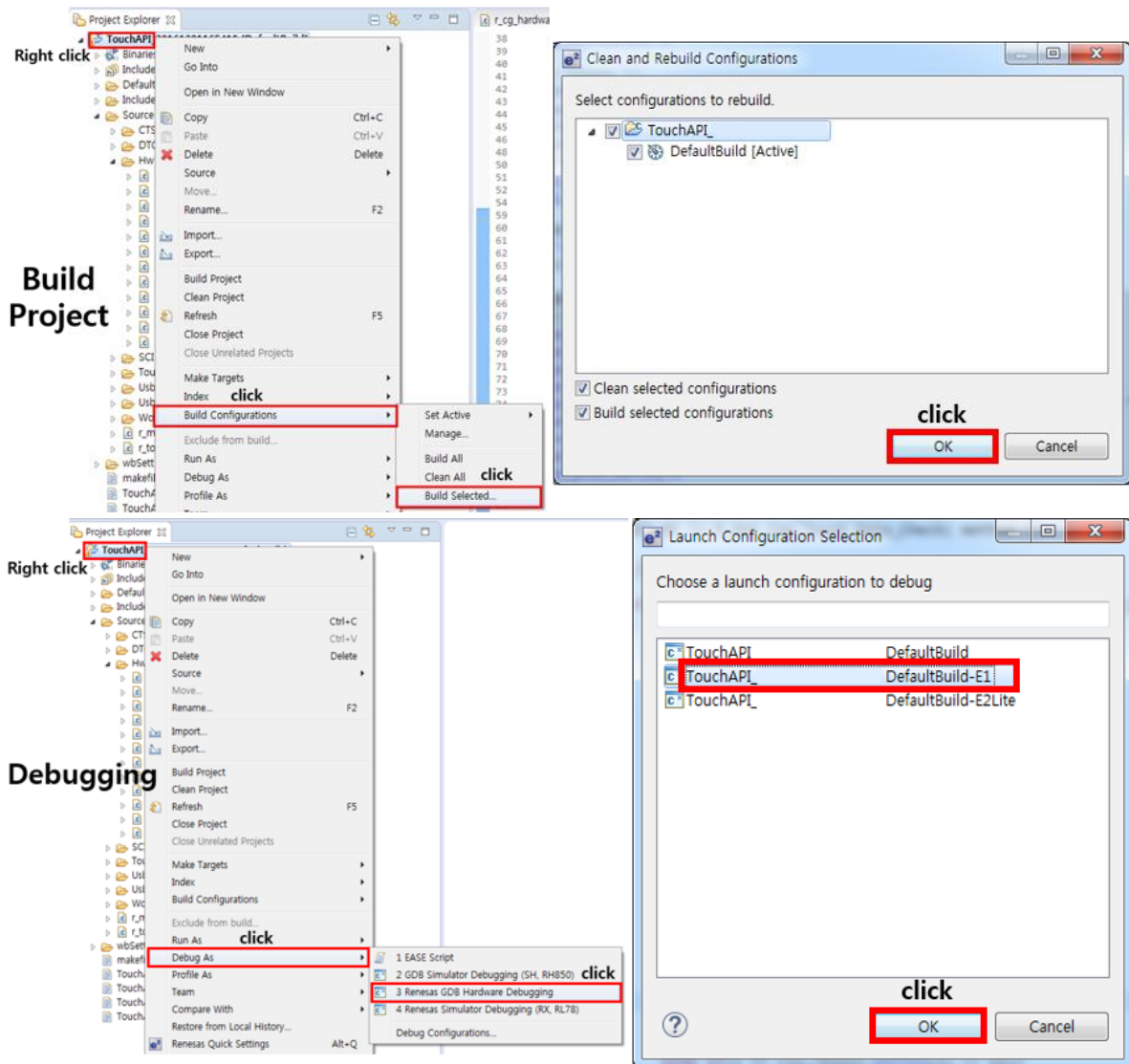


그림 4-2 프로젝트 빌드 및 디버깅

5 실행

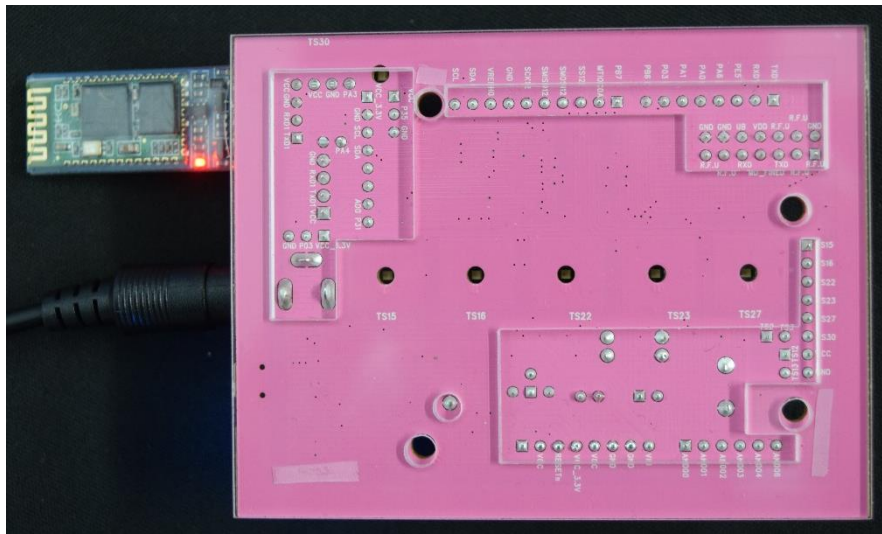


그림 5-1 블루투스 모듈을 연결한 NS-RX231



그림 5-2 안드로이드 애플리케이션 화면

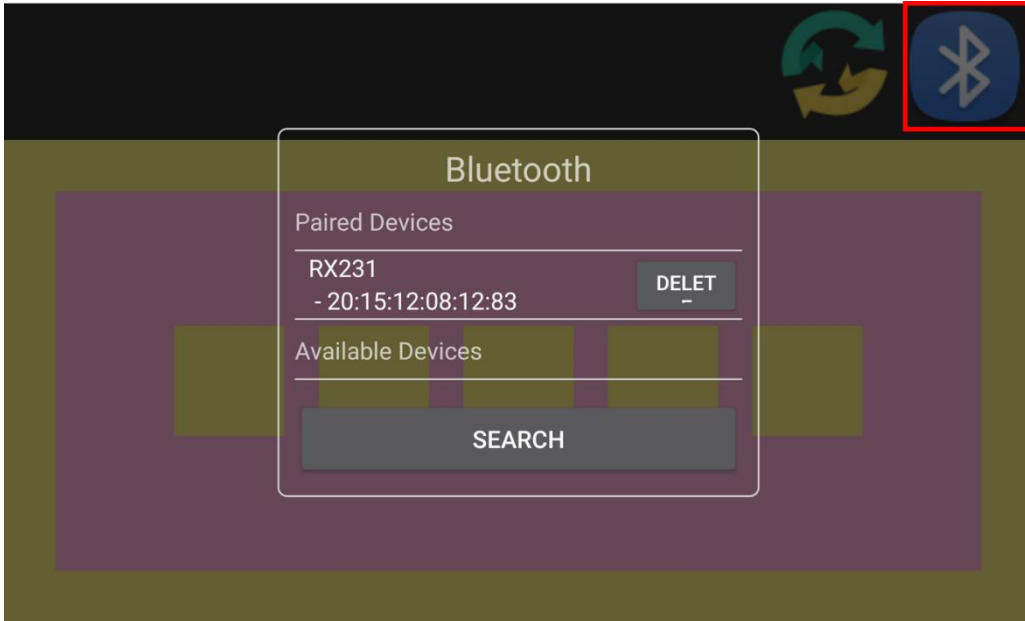


그림 5-3 안드로이드 애플리케이션

상단 오른쪽 버튼을 눌러 블루투스 모듈과 연결을 시도합니다. SEARCH버튼을 눌러 페어링할 모듈을 찾은 뒤, 비밀번호를 입력하여 연결합니다.

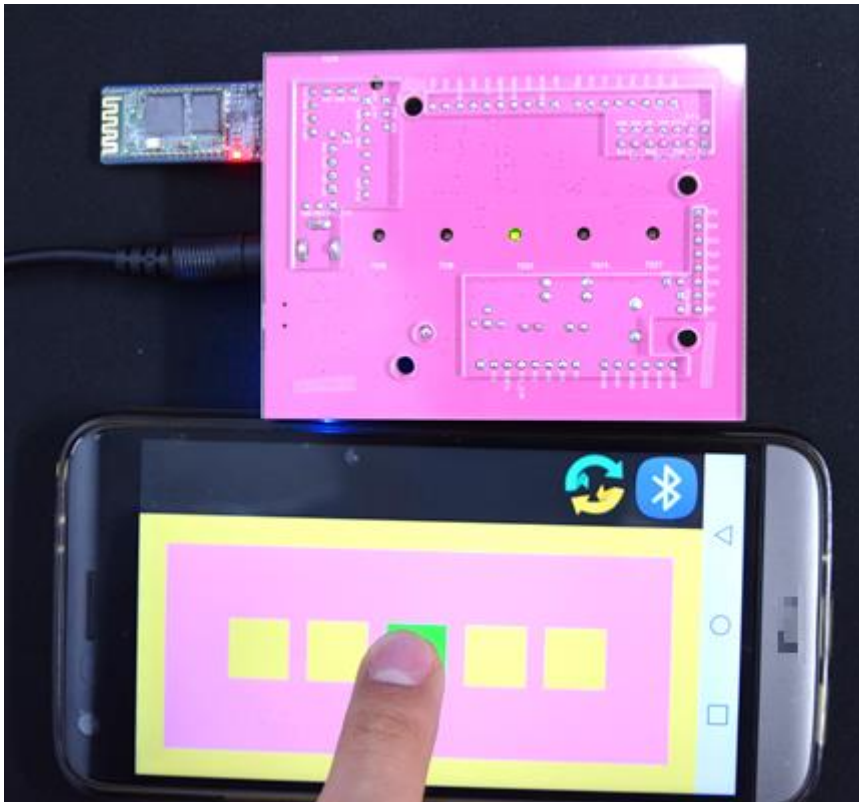


그림 5-4 안드로이드 애플리케이션 테스트

애플리케이션의 버튼을 누르면 NS-RX231의 Led가 점등되고, 터치버튼을 누르면 애플리케이션의 해당 버튼이 점등됩니다.

6 회로도

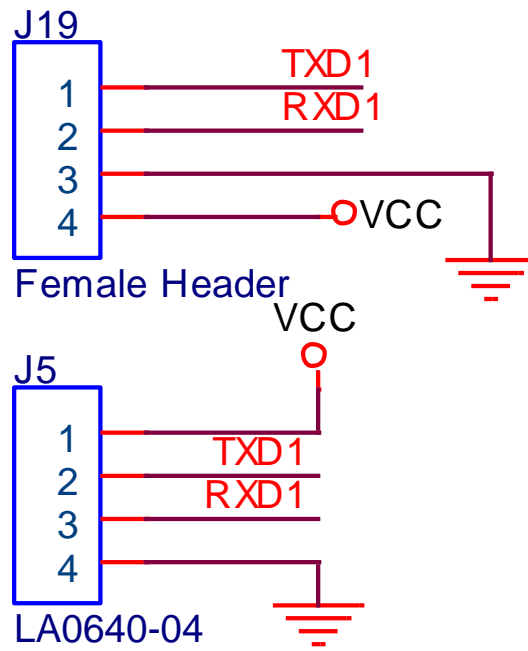


그림 6-1 NS-RX231의 UART 통신 회로도