

NS-RX231를 위한 소프트웨어가이드

<Gadget>

목차

1 개요.....	2
2 NS-RX231.....	3
2.1 프로젝트 가져오기.....	3
2.2 소스코드	4
3 GR-SAKURA.....	5
3.1 웹 컴파일러	5
3.2 프로젝트 가져오기.....	6
3.3 USB드라이버 설치.....	7
3.4 소스코드	10
4 빌드 및 업로드.....	11
5 실행.....	13

1 개요

르네사스의 가젯 프로젝트인 GR시리즈들(SAKURA, PEACH, KURUMI...)은 다른 모듈, 보드(셸드)들과 결합하여 사용할 수 있도록 제작되어 있습니다.

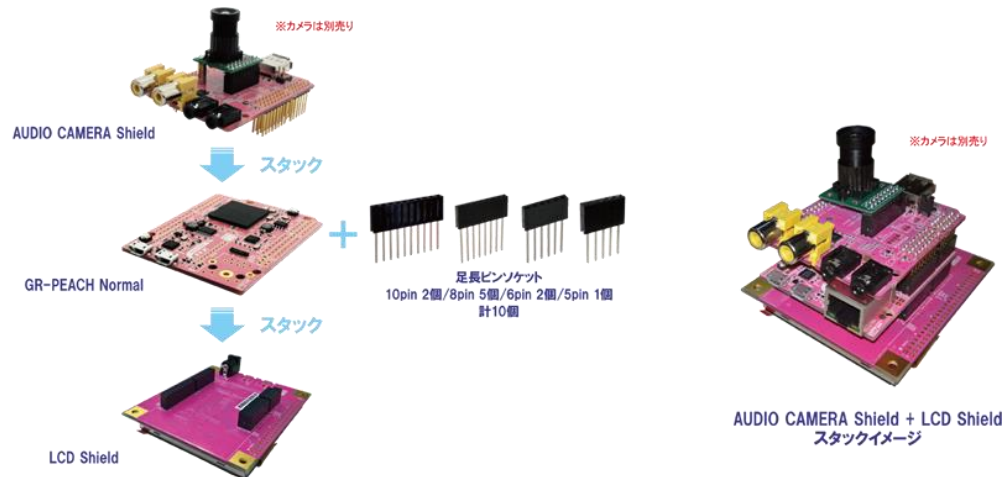


그림 1-1 GR-PEACH + 카메라 모듈 + LCD 모듈

그리고 NS-RX231과 GR-SAKURA는 핀 헤더 간격, 나사 구멍의 크기, 위치 등 Arduino UNO의 규격에 맞게 설계되어 있기 때문에 함께 결합하여 사용할 수 있습니다.

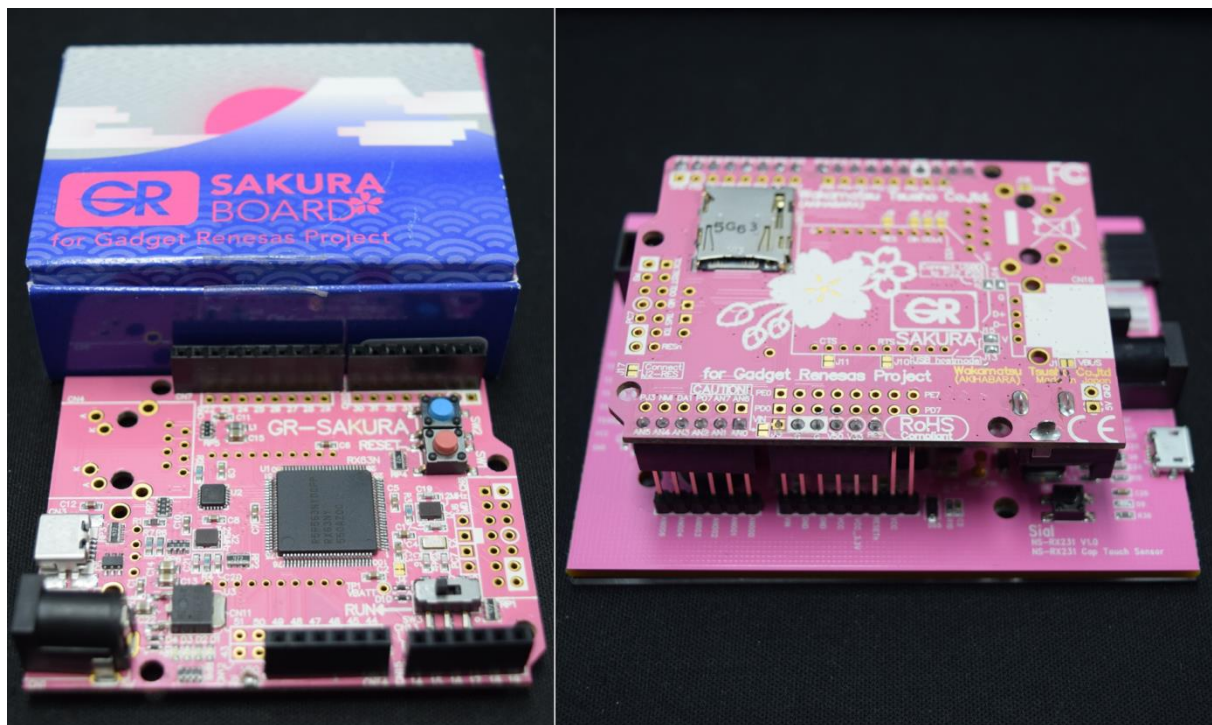


그림 1-2 GR-SAKURA 보드(왼쪽)과 NS-RX231보드의 결합 사진(오른쪽)

2 NS-RX231

2.1 프로젝트 가져오기

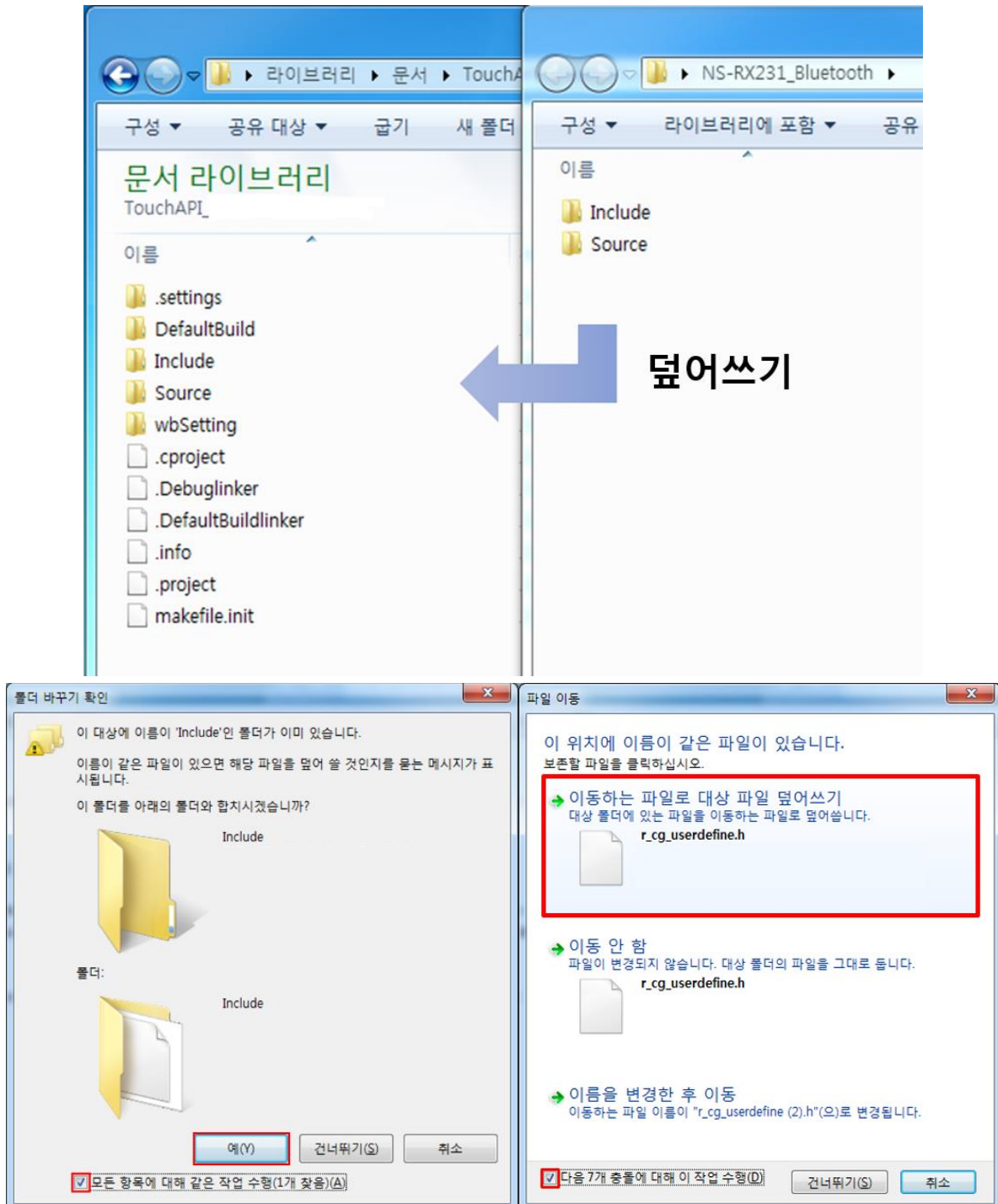


그림 2-1 소스파일 덮어쓰기

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

2.2 소스코드

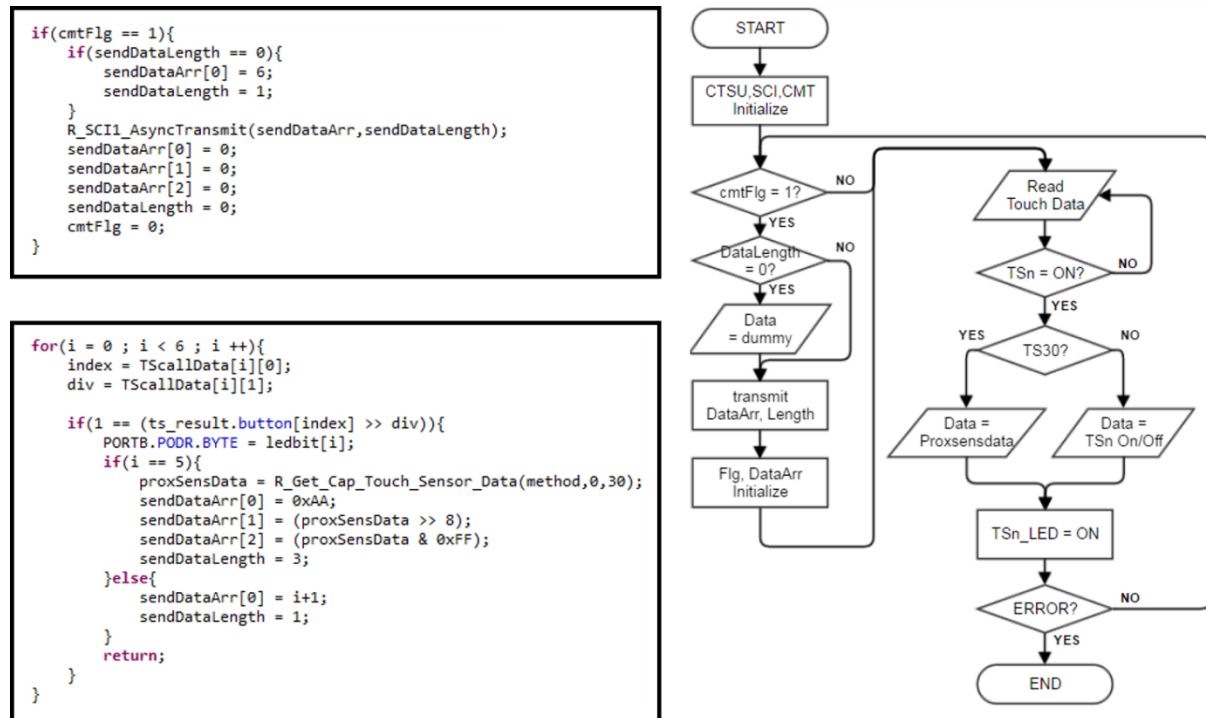


그림 2-2 소스코드 및 순서도

비교 일치 타이머(CMT)는 초기화 후 20ms마다 인터럽트를 발생하고, 인터럽트 처리 문에서 cmtFlg를 1로 만듭니다.

메인 루프에서 cmtFlg를 확인하고 cmtFlg가 1일때에만 UART를 통해 데이터를 전송합니다. 만약 터치 데이터가 존재하지 않는 경우 더미데이터를 전송하여 OFF상태를 표시합니다.

터치를 감지했을 때, 다섯개의 터치버튼(TS15, TS16, TS22, TS23, TS27)의 데이터는 ON/OFF를 나타내는 1바이트의 데이터를 배열에 넣어 보내게되고, TS30은 데이터를 구별하기 위해 구분 코드 1바이트와 2바이트의 센싱데이터를 한 바이트씩 나누어 총 3바이트의 데이터를 배열에 넣어 보내게 됩니다.

데이터 전송은 SCI1을 사용하며, NS-RX231보드와 GR-SAKURA보드가 연결되어 있을 때 TXD1, RXD1핀을 사용하여 UART통신을 이용해 데이터를 보내게 됩니다.

3 GR-SAKURA

3.1 웹 컴파일러

<http://tool-cloud2.renesas.com/>

GR-SAKURA는 GadgetRenesas의 웹 컴파일러가 지원되는데, 웹 컴파일러는 인터넷 웹사이트에서 코딩하고, 간단하게 업로드 할 수 있습니다.

웹 컴파일러는 아두이노 라이브러리를 사용해 아두이노와 비슷한 환경에서 개발을 쉽게 할 수 있습니다.

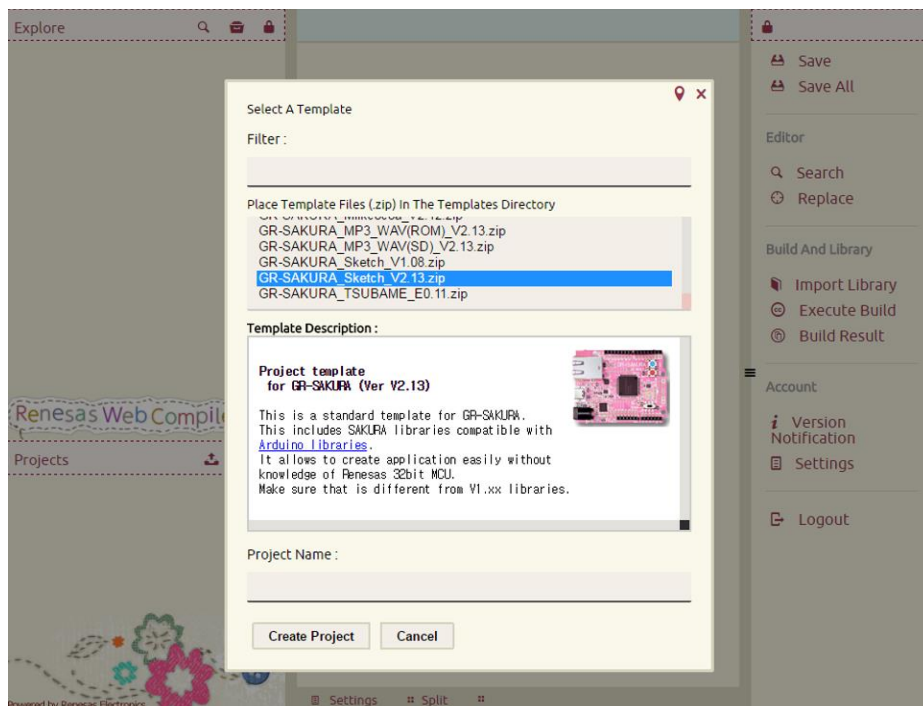


그림 3-1 웹 컴파일러 시작

※이 작업은 프로젝트를 새로 생성하여 처음부터 만들 때 사용하는 방법입니다

Template는 개발 보드의 목적에 맞게 선택하여 빌드 및 디버깅에 도움을 주는 역할을 합니다.

GR-SAKURA_Sketch_V2.13.zip으로 선택해줍시다.

1.08버전이 아닌 2.13버전을 선택한 이유는 USB드라이버 설치가 간단하고, 라이브러리의 지원이 더 좋기 때문입니다.

3.2 프로젝트 가져오기

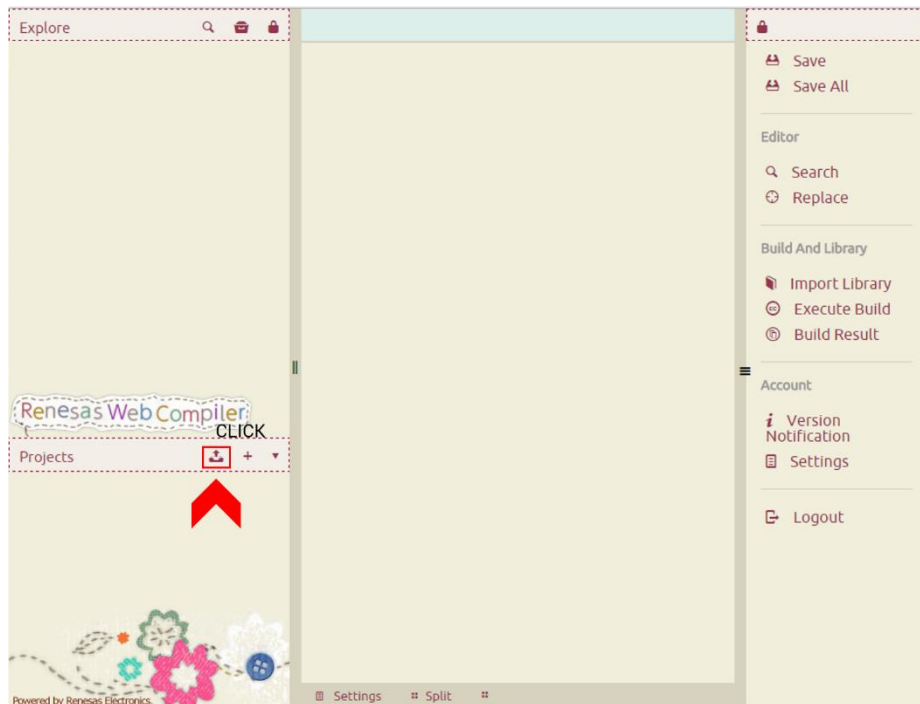


그림 3-2 프로젝트 불러오기 - 1

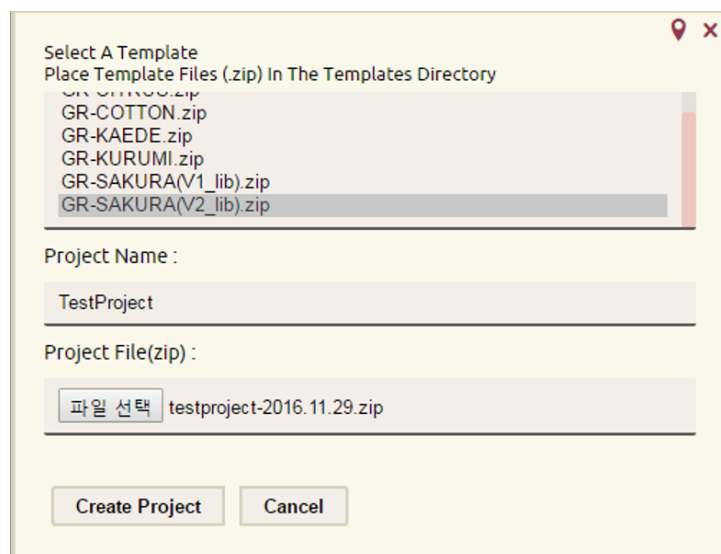


그림 3-3 프로젝트 가져오기 - 2

Template는 **GR-SAKURA(V2_lib).zip** 으로 선택 후, 밑의 Project Name은 사용자 마음대로 적어도 괜찮습니다. 그리고 그 아래의 Project File(zip)은 첨부된 파일을 열어줍니다.

3.3 USB드라이버 설치

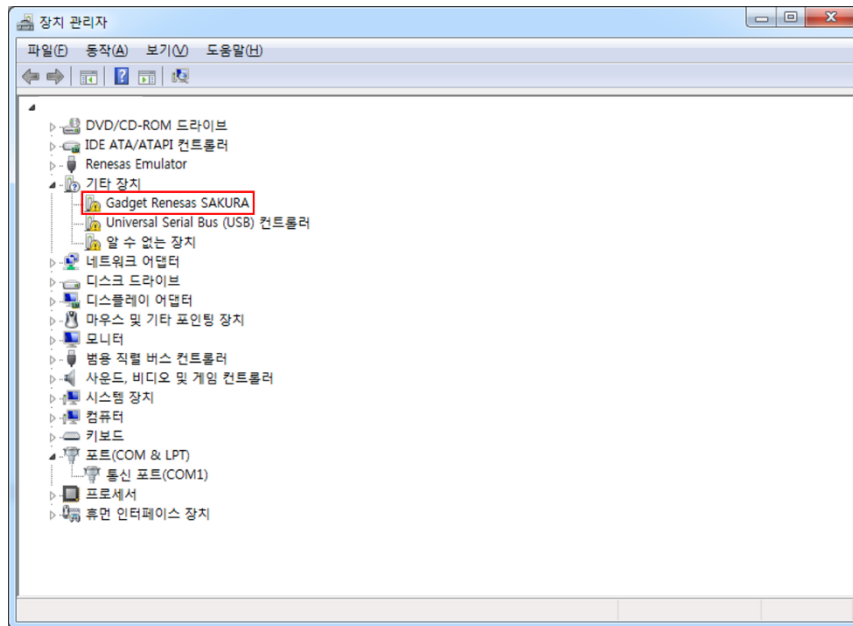


그림 3-4 장치 관리자

GR-SAKURA를 컴퓨터와 연결했을 때 위의 사진과 같이 인식할 수 없을 때 밑의 절차를 따라 드라이버를 설치해주시십시오.

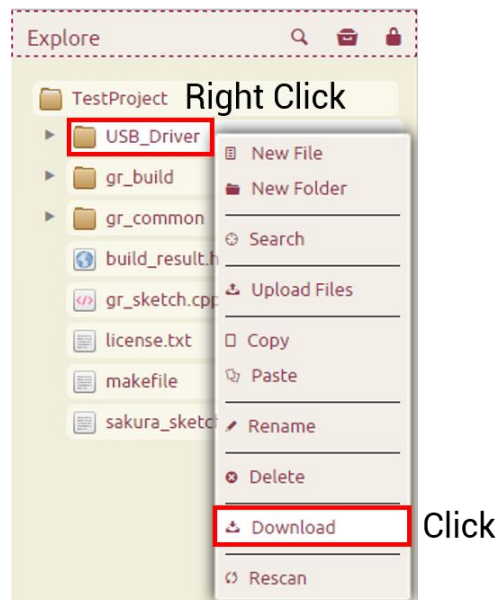


그림 3-5 USB 드라이버 설치 - 1

웹 컴파일러에서 SAKURA 프로젝트를 생성 한 후 USB_Driver 폴더를 다운로드합니다.

그리고 다운로드 한 압축 파일을 경로 설정하기 쉬운 장소에 압축을 풉니다.



그림 3-6 USB 드라이버 설치 - 2

Gadget Renesas SAKURA 항목을 우클릭 해 드라이버 소프트웨어 업데이트 화면으로 넘어갑니다.

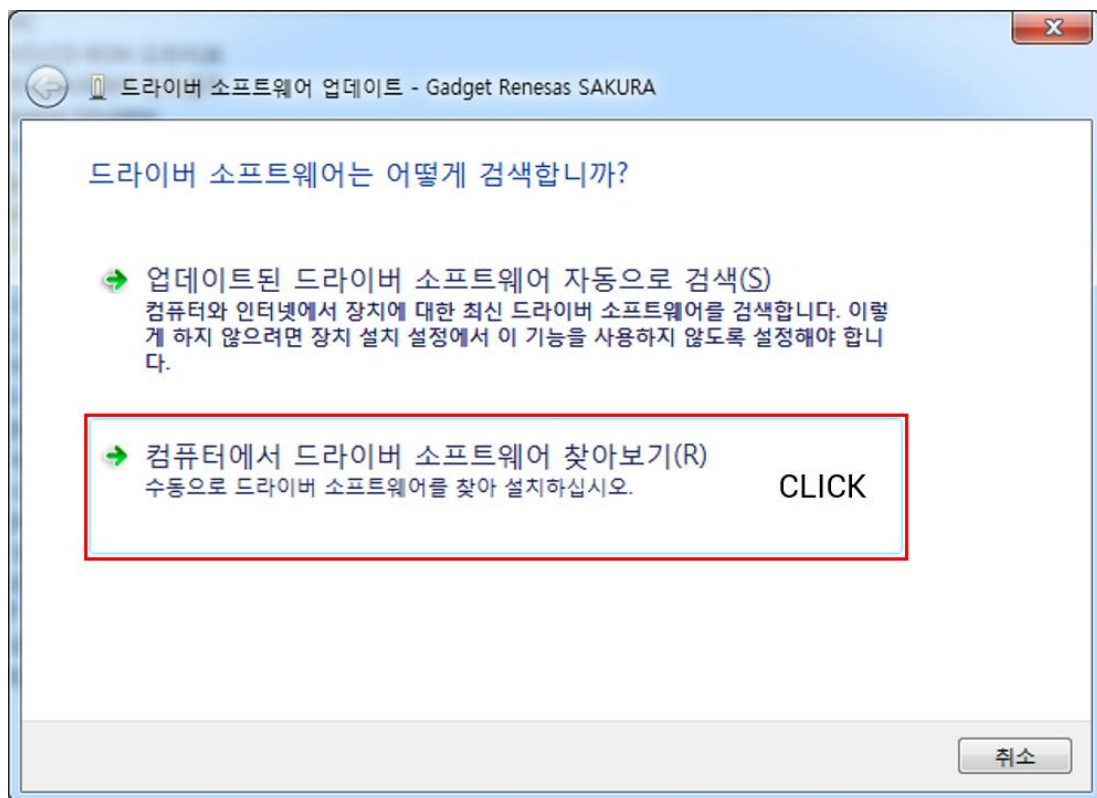


그림 3-7 USB 드라이버 설치 - 3

컴퓨터에서 드라이버 소프트웨어 찾아보기를 선택하여 계속 진행합니다.

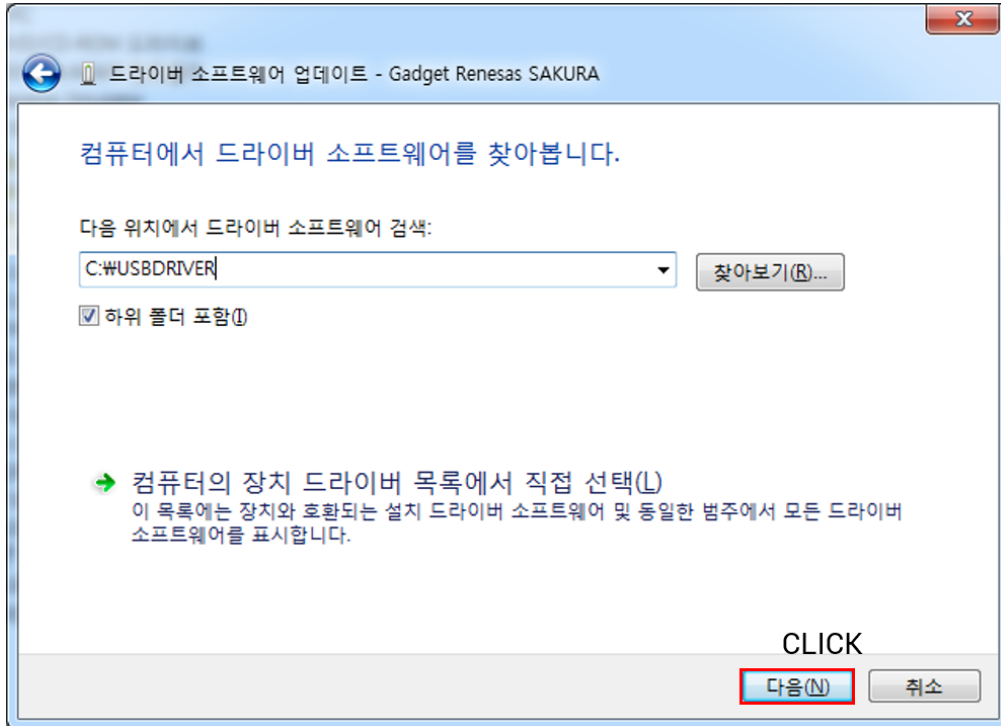


그림 3-8 USB 드라이버 설치 - 4

다운로드 받은 파일의 경로를 설정하여 진행합니다. 사진에서는 로컬디스크 C:에 폴더째로 압축을 풀었기 때문에 USB_Driver라는 폴더로 경로를 설정해 주었습니다.

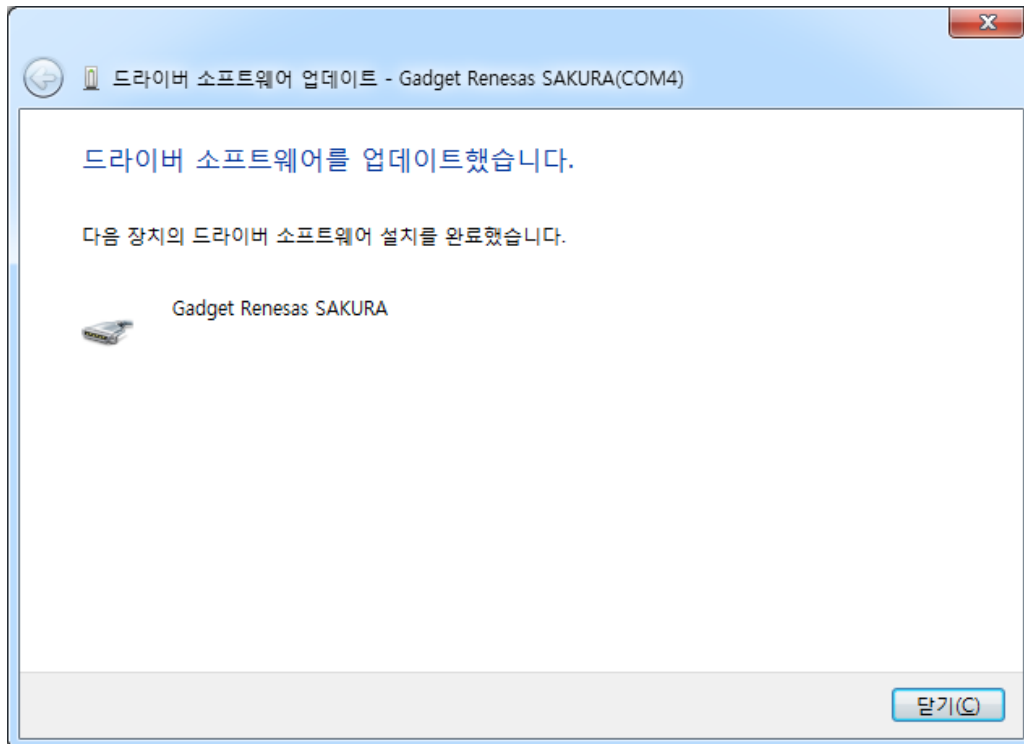


그림 3-9 USB 드라이버 설치 - 5

설치가 완료되었습니다.

3.4 소스코드

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

  pinMode(PIN_LED0, OUTPUT);
  digitalWrite(PIN_LED0, 1);
}

void loop() {
  if(blth.available()){
    data = blth.read();

    if(proxSensFlg != true && data == 170){
      proxSensFlg = true;
    }

    if(proxSensFlg == true){
      switch(proxSensCnt){
        case 1:
          proxSensData = (data<<8);
        case 2:
          proxSensData += data;
          proxSensCnt = -1;
          proxSensFlg = false;
          break;
      }
      proxSensCnt++;
    }

    if(proxSensData != 0 || (0 < data && data < 7)){
      if(data > 0 && data < 6){
        onoff[data-1] = 1;
      }
    }

    Serial.println("      TS      ");
    Serial.println(" 15  16  22  23  27 ");
    Serial.println("      ");
    Serial.print(" ");
    for(int i = 0 ; i < 5 ; i ++){
      Serial.print(onoff[i] == 1 ? " ON | " : " OFF|");
      onoff[i] = 0;
    }
    Serial.println();

    Serial.println("      |      ");
    Serial.print(" | Prox. Sensor | ");
    if(proxSensData != 0){
      Serial.print(" " + String(proxSensData, DEC));
      Serial.println("  | ");
    }
    else{
      Serial.println("  OFF  | ");
    }
    Serial.println("      |      ");
    proxSensData = 0;
  }
}
```

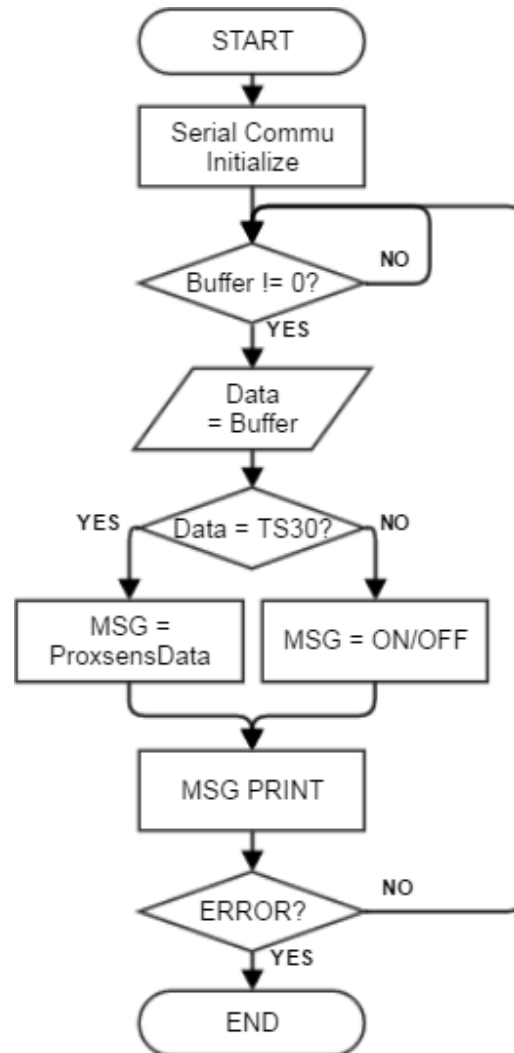


그림 3-10 소스코드 및 순서도

TXD1 RXD1핀을 통해 데이터가 교환하게 됩니다. 데이터가 일정 값이면 TS30의 데이터라고 판단해 다음에 뒤따라오는 후속 데이터를 OR연산으로 결합하여 결과값을 산출해 냅니다.

데이터가 TS30이 아닐 경우에는, On / Off를 표시하는 메시지를 만들어 표시하게 됩니다.

4 빌드 및 업로드

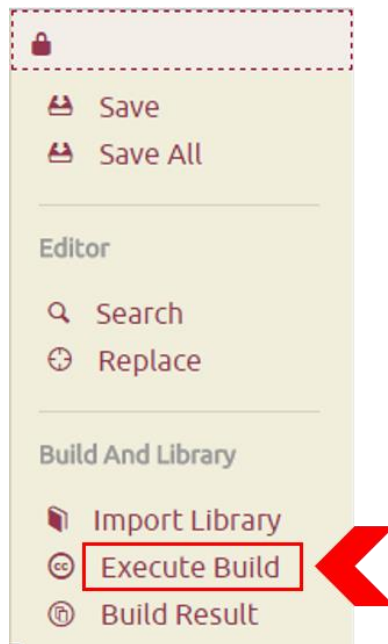


그림 4-1 프로젝트 빌드

오른쪽의 메뉴 바에서 Execute Build 를 선택하여 프로젝트를 빌드합니다.

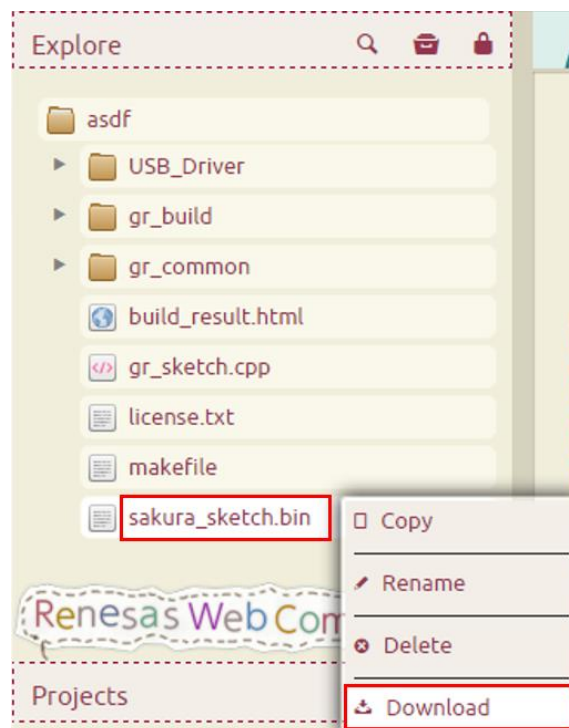


그림 4-2 sakura_sketch.bin 다운로드

빌드가 완료되면 sakura_sketch.bin 파일이 생성되며, 이 파일을 GR-SAKURA 보드에 업로드하여 프로젝트를 실행합니다.

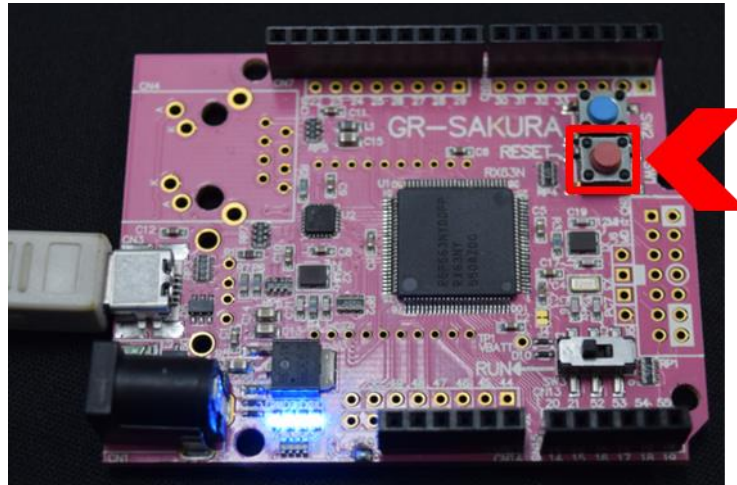


그림 4-3 GR-SAKURA와 PC연결

USB Mini-B케이블을 사용해 GR-SAKURA와 PC를 연결합니다.

연결을 했음에도 불구하고 저장소가 뜨지 않는다면 RESET스위치를 눌러 진행해 주십시오.

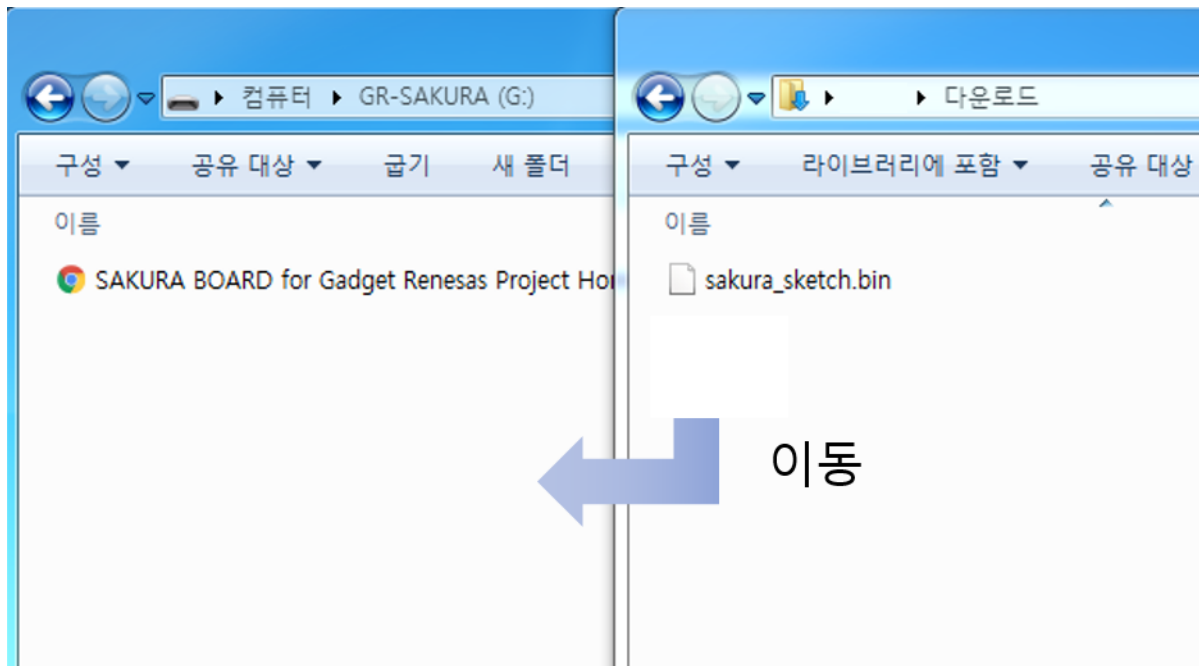


그림 4-4 GR-SAKURA 프로젝트 업로드

Web Compiler에서 다운받은 sakura_sketch.bin파일을 GR-SAKURA에 업로드합니다.

업로드가 진행되는 동안 GR-SAKURA보드의 LED가 점멸하며 움직이기 때문에 육안으로도 확인할 수 있습니다.

5 실행

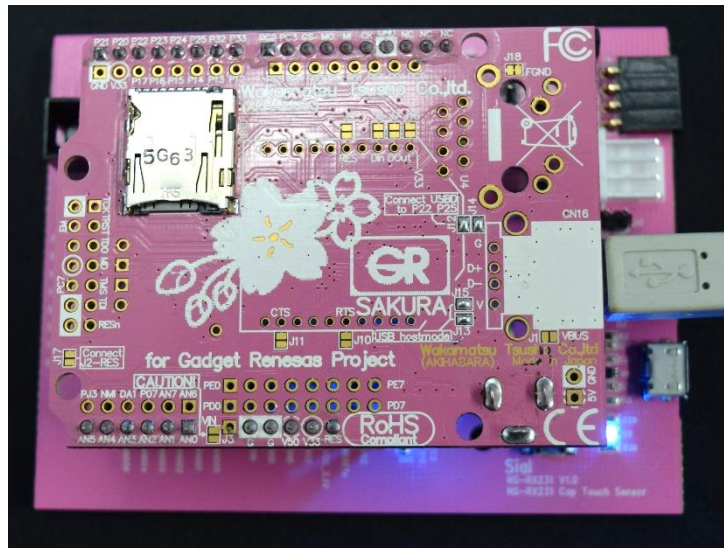


그림 5-1 NS-RX231과 GR-SAKURA의 결합 및 USB 연결 사진

GR-SAKURA와 NS-RX231보드를 결합 후, SAKURA의 mini-B포트에 케이블을 꽂아 컴퓨터와 연결했습니다.

터미널 프로그램은 Tera Term을 사용했으며, 다운로드 링크는 다음과 같습니다.

<https://osdn.net/projects/ttssh2/releases/>

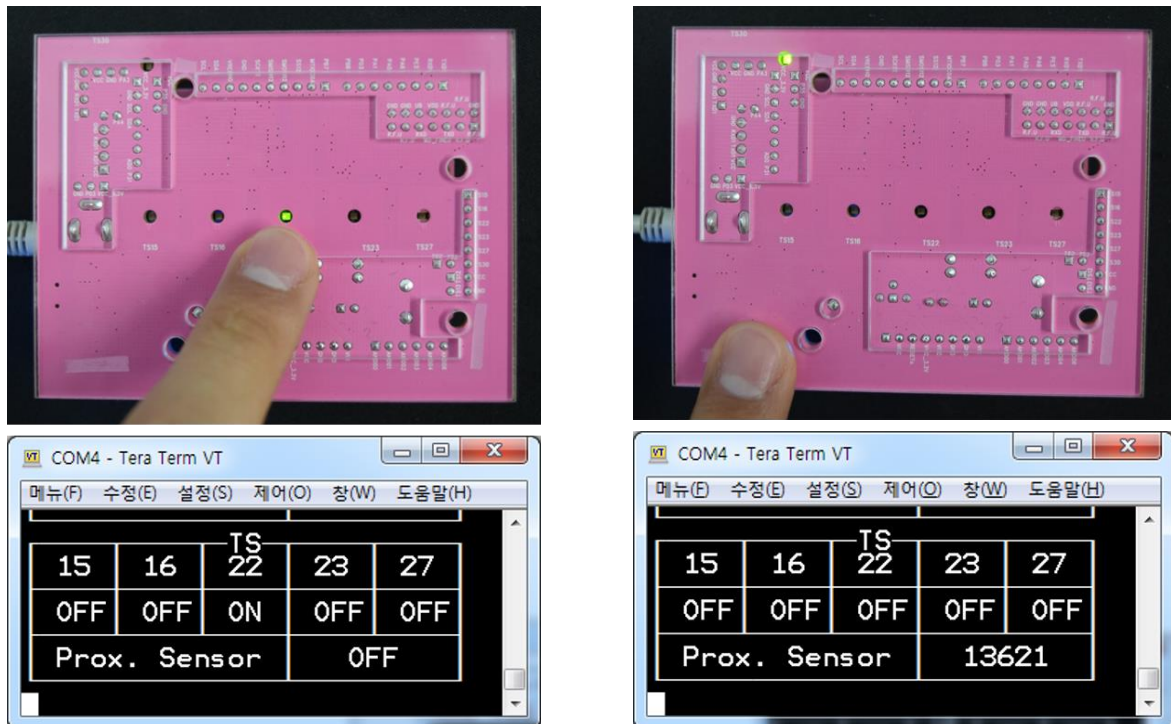


그림 5-2 터미널의 반응