

Intel® Quark™ Microcontroller Developer Kit D2000

Temperature Sensor: Lab 4 Guide

Contents

[Intel® Quark™ Microcontroller Developer Kit D2000 Temperature Sensor: Lab 4 Guide](#)

[Overview](#)

[Prerequisites](#)

[Lab Workflow](#)

[Lab Instructions](#)

[Step 1: Connect the Intel® Quark™ Microcontroller Developer Kit D2000 board](#)

[Step 2: Check the USB drivers for COM port and JTAG adapter](#)

[Step 3: Run PuTTY](#)

[Step 4: Run Intel® System Studio™ for Microcontrollers](#)

[Step 5: Update target ROM](#)

[Step 6: Create a new project](#)

[Step 7: Replace BMCs](#)

[Step 8: Build the project](#)

[Step 9: Flash and run the project](#)

[Step 10: Observe Results](#)

Overview

The purpose of this lab is to show how to program the temperature sensor integrated in Intel® Quark™ Microcontroller Developer Kit D2000 board using Quark Microcontroller Software Interface (QMSI) and Intel® System Studio 2016 for Microcontrollers. The sample project used in this lab reads the temperature and prints it to serial port.

Prerequisites

- Host machine:
 - PC running Microsoft Windows 7 or later with two available USB ports.
- Software:
 - Intel® System Studio for Microcontrollers 2016.
 - PuTTY terminal emulation software
- Hardware:
 - Intel® Quark™ Microcontroller Developer Kit D2000 board.
 - USB Type A to Micro USB Type B cable. This cable is supplied with the kit.
 - FTDI TTL-232R-3V3 USB to Serial cable
 - 3 male to male jumper wires

Lab Workflow

1. Connect the D2000 board to the host PC; connect FTDI cable
2. Check that USB drivers are properly installed
3. Run Putty
4. Run Intel® Systems Studio for Microcontrollers
5. Update Boot ROM (once per board)
6. Create a new project using accel template and replace the main.c using provided file
7. Replace the bmc150.c and bmc.h from template using provided files
8. Build the project
9. Flash and run the project using Run <project_name> (flashing)
10. Observe the result: current room temperature should be outputted in the ISSM terminal

Lab Instructions

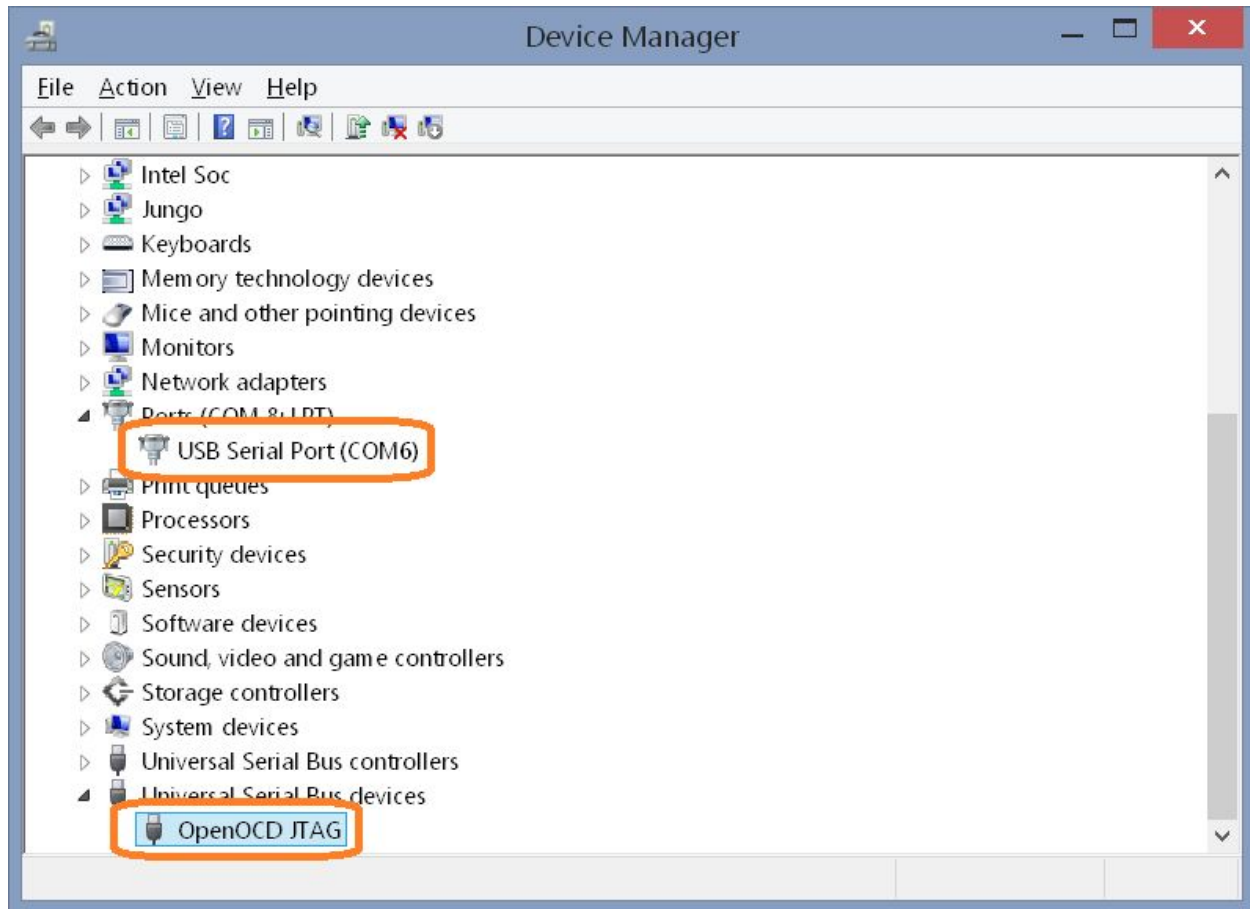
Step 1: Connect the Intel® Quark™ Microcontroller Developer Kit D2000 board

Using jumper wires connect the FTDI TTL-232R-3V3 cable to the microcontroller as shown on the picture below.

Make sure that pin 1 of the cable (black wire, also identified by a small arrow on the connector) is connected to the GND pin of the board. Pin 4 of the cable (orange wire) is connected to the RX pin (0) of the board. And pin 5 of the cable (yellow wire) is connected to the TX pin (1) of the board.

Connect the USB end of the FTDI cable to your PC.

Connect the board to your PC using the USB Type A to Micro USB Type B cable provided in the kit. The green P3V3 LED on the board should light up.



If the **USB Serial Port** device does not appear or the **Ports (COM & LPT)** category is missing, the FTDI driver needs to be installed. It can be downloaded from the [FTDI web site](#).

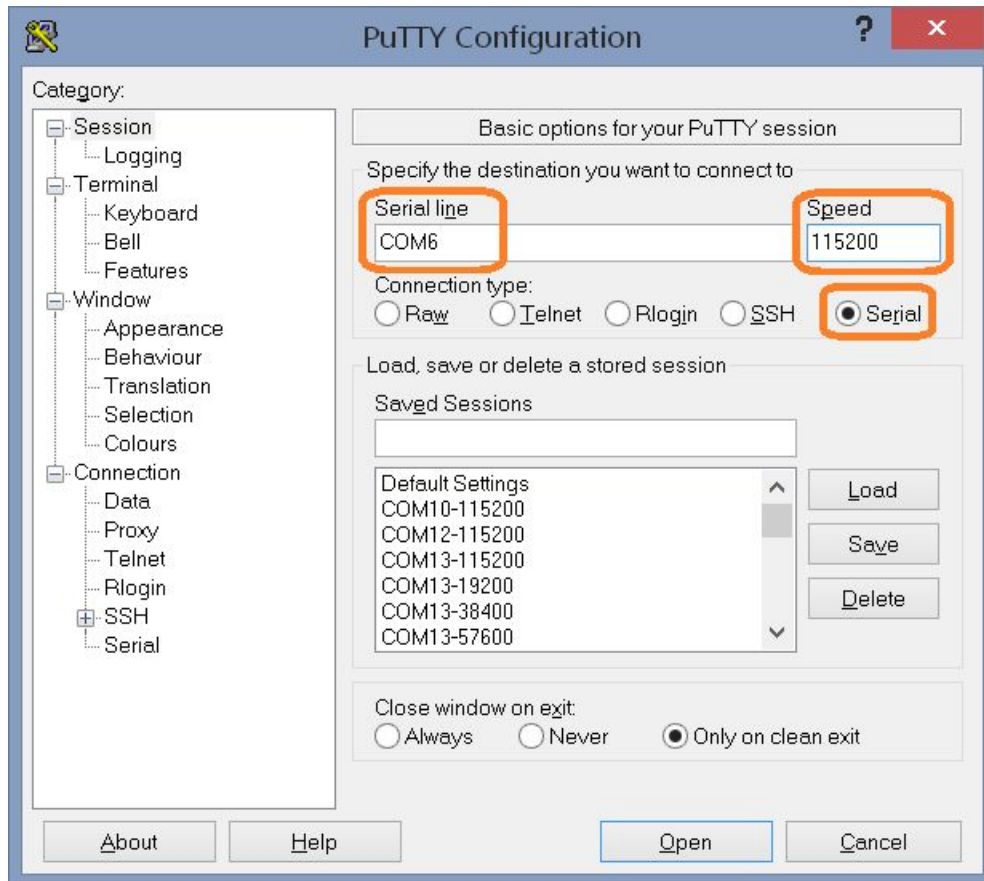
If the **OpenOCD JTAG** device does not appear or the **Universal Serial Bus devices** category is missing, the USB driver needs to be installed. To install the driver navigate to [C:\IntelSWTools\ISSM_2016.0.027\tools\debugger\driver](#) directory using **File Explorer**, double click on the *install.bat*, and answer 'y' in the **OpenOCD JTAG Driver** window that will appear.

Step 3: Run PuTTY

Click on the Windows start button, type **putty**, and press the **Enter** key to the application.

The **PuTTY Configuration** dialog will appear. Using this dialog configure PuTTY as follows (also refer to the screenshot below):

- Select **Serial** radio button
- Type **115200** in the **Speed** field. 115200 bits per second (bps) is the default speed of the UART
- Type the COM port number (e.g. **COM6**) in the **Serial Line** field.

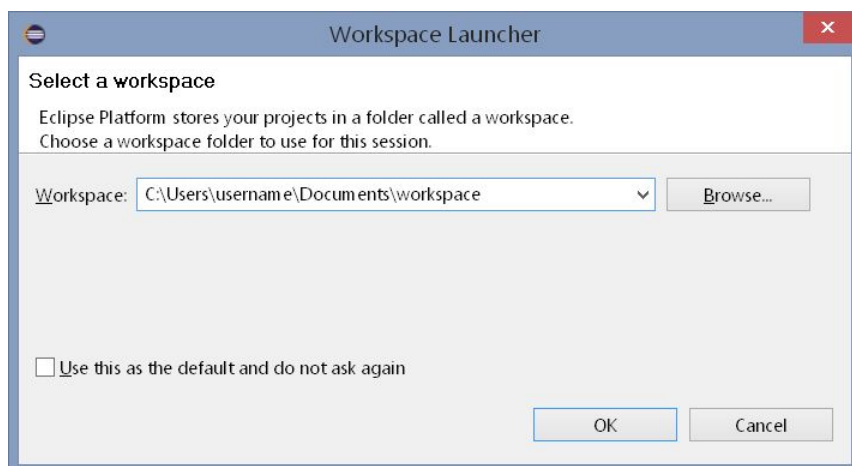


Click **Open** button to open the serial terminal

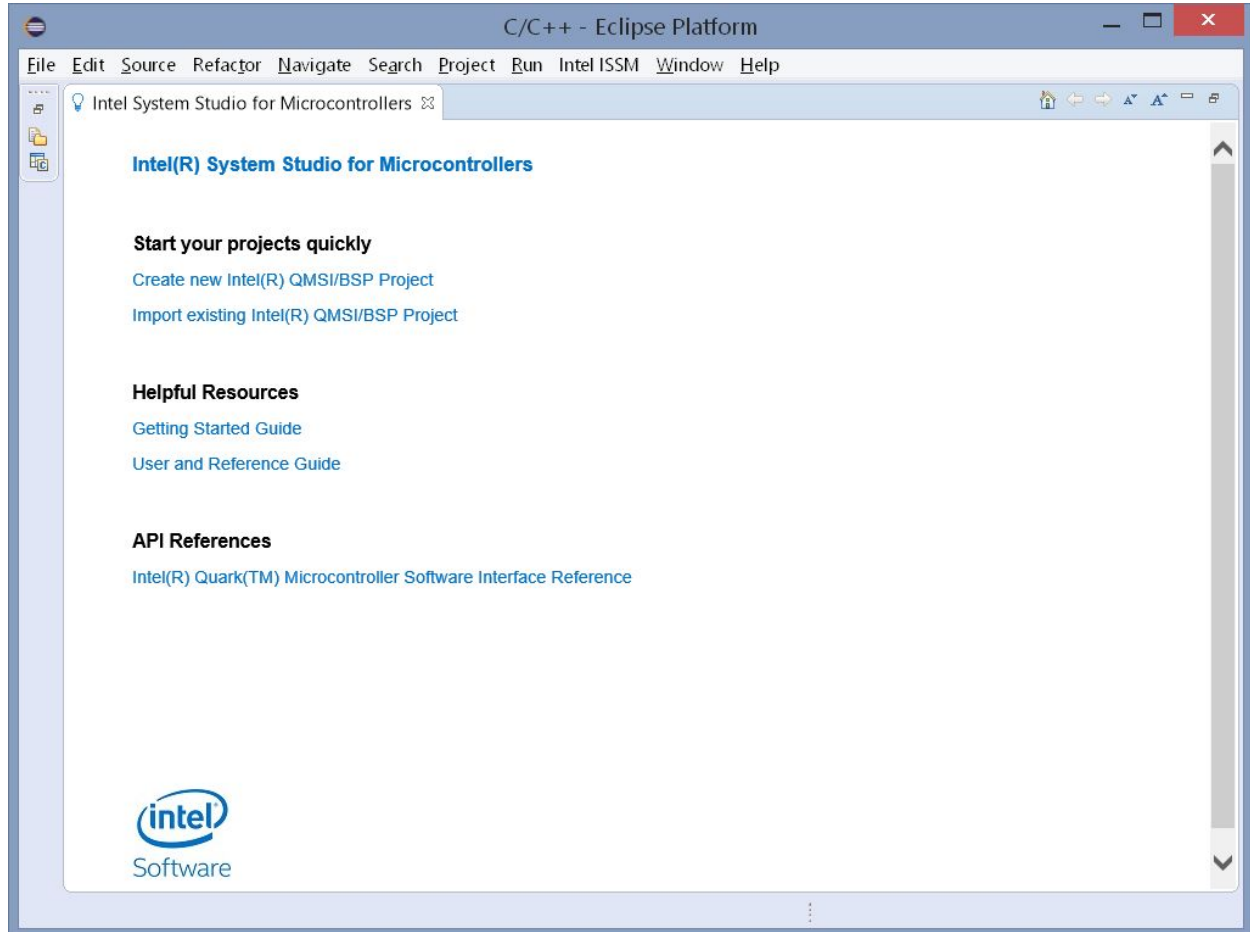
Step 4: Run Intel® System Studio™ for Microcontrollers

Using **File Explorer** navigate to C:\IntelSWTools\ISSM_2016.0.027 directory, and double click on the `iss_mcu_ide_eclipse-launcher.bat`.

The Intel® System Studio™ for Microcontrollers will prompt for the workspace location:



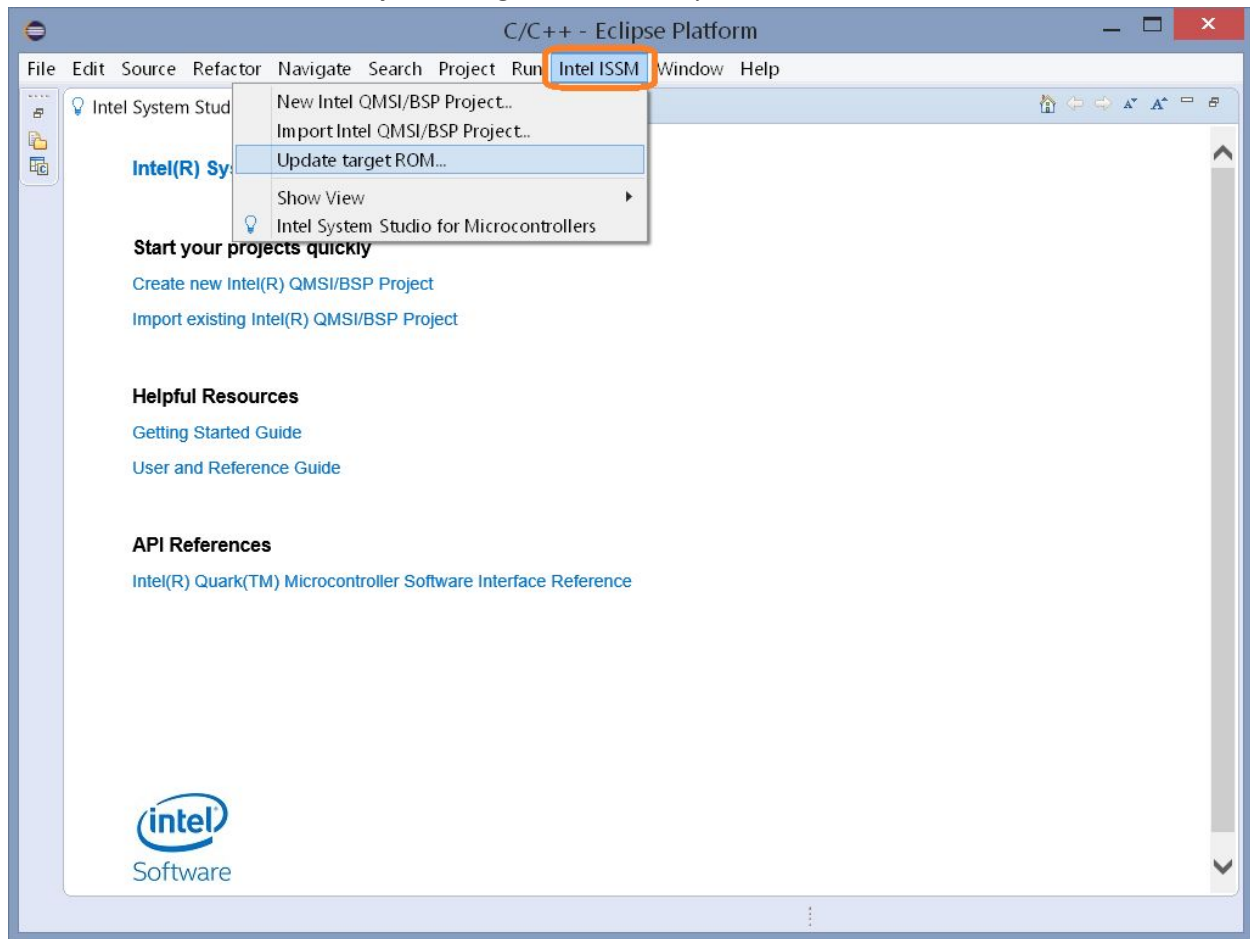
It is recommended to click **Browse...** button here and change the workspace location to your user directory, for example **C:\Users\username\Documents\workspace**. Click **OK** button to confirm workspace selection. The Intel® System Studio for Microcontrollers windows will appear as shown on the screenshot below.



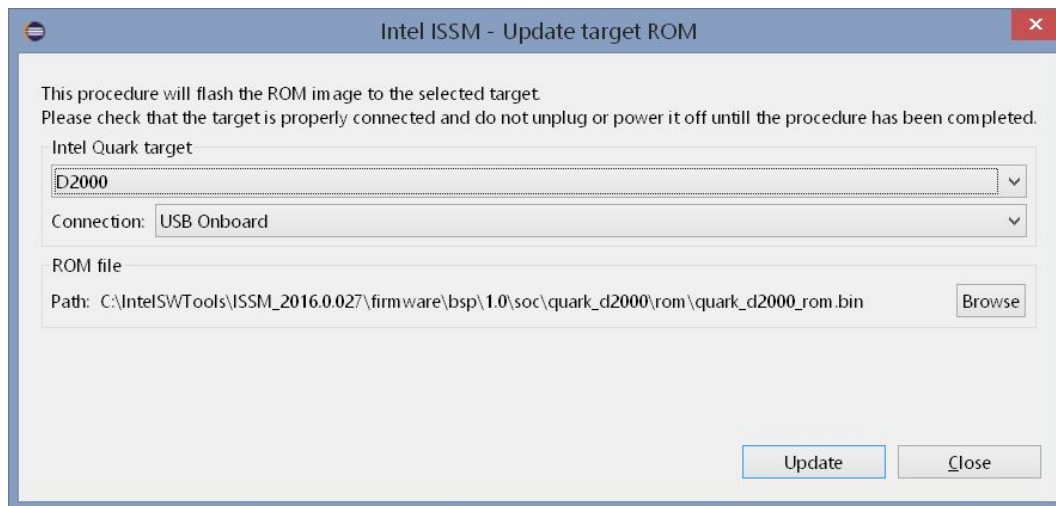
Step 5: Update target ROM

Note: This step needs to be performed only once per board.

From **Intel ISSM** menu select **Update target ROM...** entry as shown on the screenshot below.

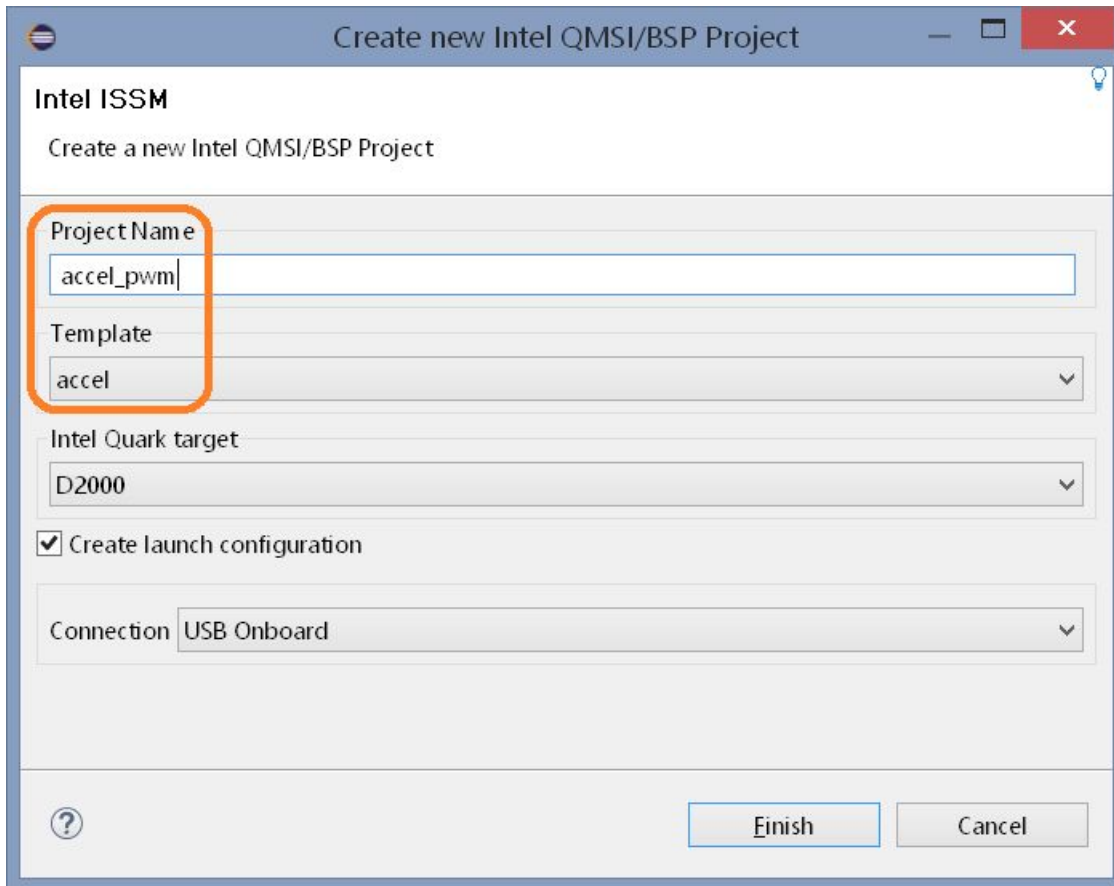


The **Update target ROM** dialog will appear. Make sure that the selected *Intel Quark target* is **D2000**, and the selected *Connection* is **USB Onboard**. Click **Update** button to program the ROM to the microcontroller.



Step 6: Create a new project

From the **Intel ISSM** menu select **New Intel QMSI/BSP Project...** The **Create new Intel QMSI/BSP Project** will appear. Type an appropriate name in the Project Name field. Make sure that other settings match to the settings on the screenshot below.



Click the **Finish** button to create the new project.

In the **Project Explorer** window on the left side of Intel® System Studio for Microcontrollers click on the <project_name> project. Find *main.c* file and double click on it to open it the editor. Replace the *main.c* content with the code given from the link below. Save the file by pressing Ctrl-S key combination.

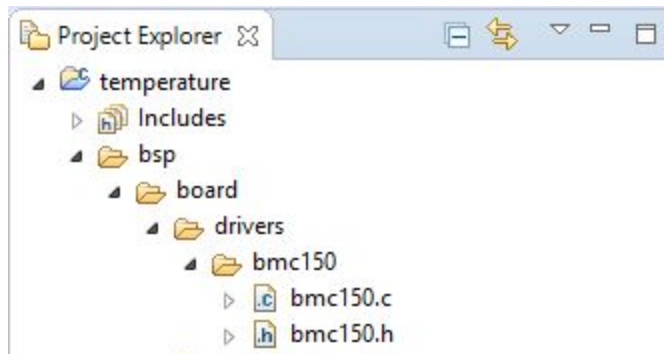
<https://github.com/Salv95/Intel-Quark-D2000-Combined-Mag-Accel-and-Temp-code-with-PWM/blob/master/main%20for%20temperature%20sensor%20without%20PWM/main.c>

Step 7: Replace the bmc150.c and bmc.h from template using provided files


Clone/download the bmc150.c and bmc.h files from the link below:

<https://github.com/Salv95/Intel-Quark-D2000-Combined-Mag-Accel-and-Temp-code-with-PWM>

Once cloned/downloaded, replace the bmc150.c and bmc150.h with the new files. Below is a screenshot of the location of the files.



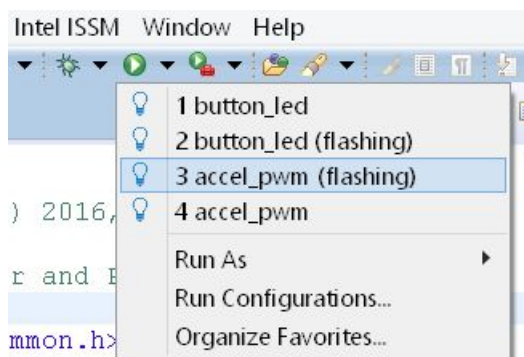
Step 8: Build the Project

Click on the **Build** button (“hammer”  icon) on the toolbar to build the project. Alternatively it is possible to select the build configuration using the drop down menu next to that button. Intel® Systems Studio for Microcontrollers provides two configurations: **debug** and **release**. The **debug** configuration includes symbols in the generated binary files to facilitate debugging. It is the default configuration. The **release** configuration optimizes code for deployment.

Step 9: Flash and run the project

Click on the drop down menu next to the **Run** button (“play” icon). Select **<project_name> (flashing)** from the menu. The Intel® Systems Studio for Microcontrollers will recompile the code, and flash it to the microcontroller.

*Note: Intel® System Studio for Microcontrollers offers two ways to run the code: The first – **<project_name>** option assumes that the code had been already programmed to the microcontroller. The second – **<project_name> (flashing)** option will recompile and re-flash the code. Use this option if the source code had been changed since the last time microcontroller was programmed. This is the option used in this lab step.*

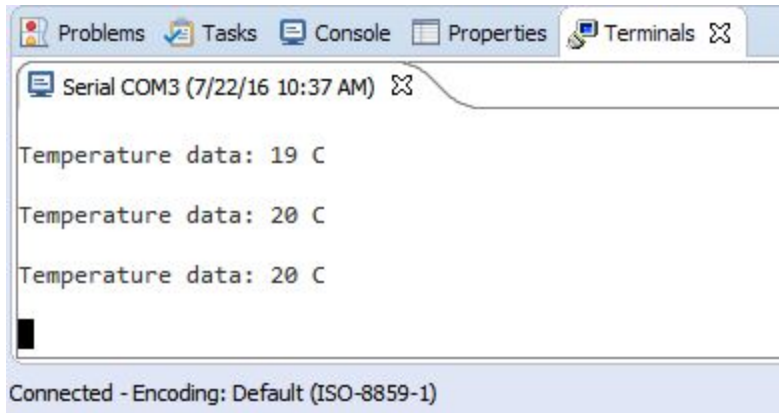


Note: The flashing process takes some time. Look for the progress bar at the bottom right corner of the Intel® Systems Studio for Microcontrollers window:



Step 10: Observe the Result

Switch to the PuTTY window. You should be seeing temperature readings being printed out in the terminal.



*Note that the screenshot is from the Eclipse terminal; however, it will still work for Putty

**For a guide on how the code works please refer to master branch of the code (link below)

<https://github.com/Salv95/Intel-Quark-D2000-Combined-Mag-Accel-and-Temp-code-with-PWM>