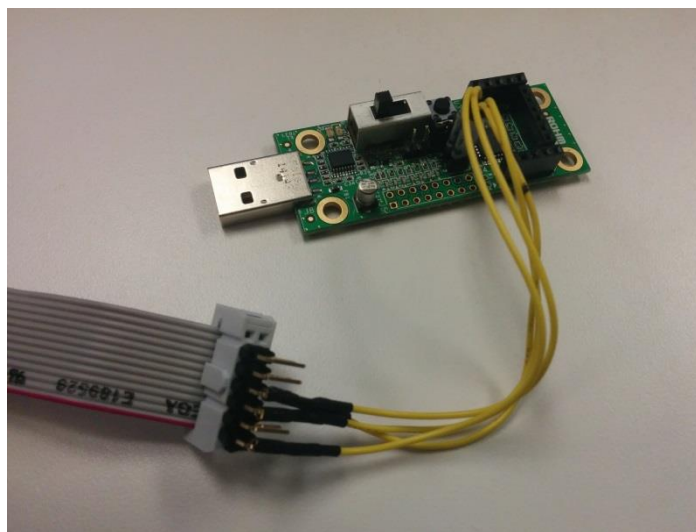
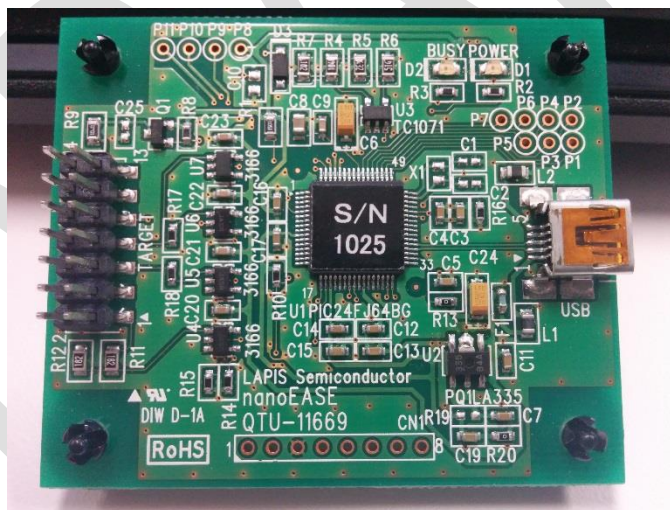


Firmware Flashing and Initial Hardware Check Guide for ROHM's Sensor Platform Kit



Above: ROHM Sensor Platform base board connected ROHM nanoEase Cable for Firmware Flash



Above: nanoEase Programmer Board required for flashing the LAPIS Microcontroller Boards

Table of Contents

Revision History	2
Introduction	3
Tools Required for Getting Started	3
Loading the HEX firmware files to Sensor Platform Base Kit	4
Programming and Debugging the Sensor Platform Base Kit	8
Performing a Functional Check for After Assembly	12
Checking the Base Board Power Connection via USB and the Standalone LED Output	12
Checking the Checking the Reset Button	12
Checking Base Board Communication over USB	13
Checking Each Individual Sensor Output	14

Revision History

Date	Description	Revision ID
27 January 2015	First Draft	A

Introduction

The following document was written to provide an explanation to how to flash the firmware for these sensor platform boards with the most recent firmware. Also, this guide will go into the details of how to check the hardware assemblies of these boards.

For other guides, such as operation manuals and other platform connection tutorials, please visit our public repository at the following link...

ROHM's Sensor Platform Kit: <https://github.com/ROHMUSDC/ROHMSensorPlatformEVK>

- ROHM BH1721 – Digital Ambient Light Sensor
- ROHM BH1620 – Analog Ambient Light Sensor
- ROHM BU52011HFV – Hall Switch Sensor
- ROHM BDE0600G – Analog Temperature Sensor
- LAPIS ML8511 – Analog UV Sensor
- Kionix KMX061 – Digital Accelerometer and Magnetometer

Tools Required for Getting Started

The Base board for the Sensor Platform Kit uses a LAPIS **ML610Q112** Microcontroller as the main application processor for this kit. Please note, that this board does not come with any on-board programming chip; therefore, in order to successfully flash this MCU, you will need a **“uEase” or “nanoEase” programmer/debugger board**. These can be found at the following link:

uEase MCU programmer and Debugger Hardware:

<http://www.digikey.com/product-search/en?x=0&y=0&lang=en&site=us&keywords=uEase>

Please note that you will also need to register and login to the Lapis Support Site in order to download the software tools required for flashing this device.



LAPIS Support Site Login/Registration Page:

<https://www.lapis-semi.com/cgi-bin/MyLAPIS/regi/login.cgi>

Once you've logged into the support suite, click the “Development Support System” link on the left side of the screen.

After this page has loaded, please download the latest version of the “*U8 Development Tools*”. This Software package includes the application named “*FWuEASE Flash Writer*” and “*IDEU8 GUI Environment*”. Either application can be used to flash the sensor platform base boards. IDEU8 would be used to develop code and program/debug. The FWuEASE flash writer can be used to program HEX files

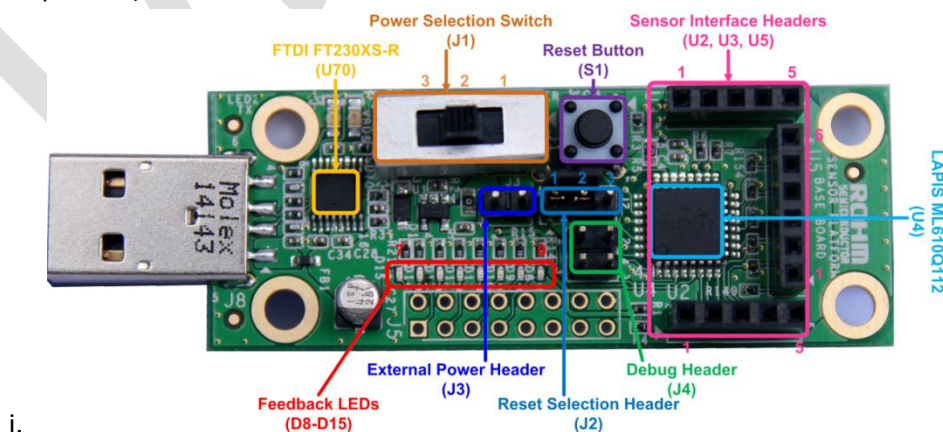
direct to the MCU Base board. See the picture below for the software package that should be downloaded after registering for the LAPIS Development Support Site:

Software package	Version	Explanation	File	Release note
U8 Development Tools	NEW 1.12.0	<p>Project manage tool, Build tool, Support tools for LCD Control, Flash programming tools(FwuEASE),ROM Code Generator Tool for Code Entry, U8 Device Information Files in 'Low Power Microcontroller Development Support system Tabale' and etc are included in this package.</p> <p>-Notice- [Notice about upgrade of CCU8 and RLU8] Those who are using the following versions need to download PDF form here with the software "CCU8 compiler" and "RLU8 linker" contained in the package before U8 Development Tools Release 1.8.2. -CCU8 V3.21 or older version. -RLU8 V1.40 or older version.</p> <p>[Reference] There is the "U8 development tools tutorial" explaining how to use this U8 development tools. Please download this tutorial from this page.</p>	 Download (321MB)	 Download
MWuEASE Multiple Flash Writer	2.2.0	<p>This soft-ware Supports simultaneous Flash programming of multiple boards of the same type</p> <p>** The software included to "U8 Development Tools" **</p>		

Loading the HEX firmware files to Sensor Platform Base Kit

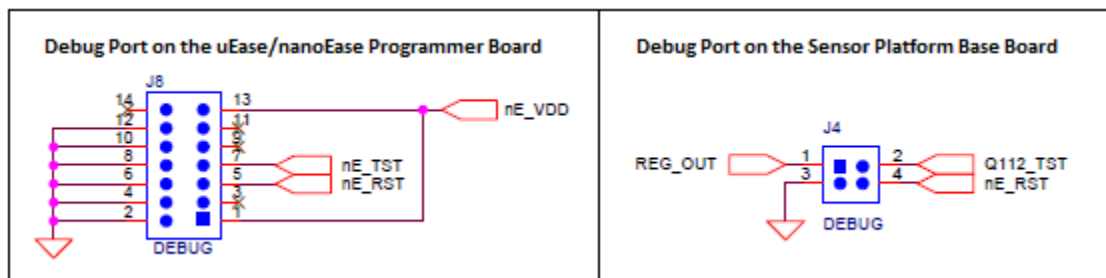
This section should be referred to if the user only wants to load an already generated HEX file into the existing base board. If the user wants to adjust source code and test/debug their code, please skip forward to the section entitled, "Programming and Debugging the Sensor Platform Base Kit".

1. Acquire a "uEase" or "nanoEase" programmer/debugger board and install the FWuEASE Flash Writer as explained in the previous section.
2. Connect the "uEase" or "nanoEase" programmer/debugger board to the sensor platform base board
 - a. On the Sensor Platform Base board, we will want to connect to pin header J4 (2x2 header pin row)

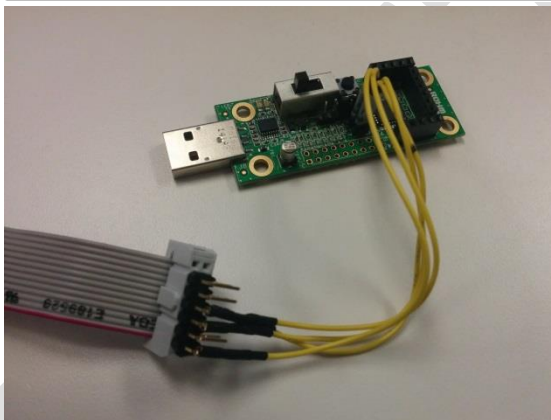


i.

- b. On the uEase or nanoEase programmer boards, we will want to isolate 4 pins (VDD, GND, TST, and RST). Then, we recommend building a jumper cable to make the connection easier to re-connect. See below for schematics for this debug port and an example debug port connector we used to flash these boards

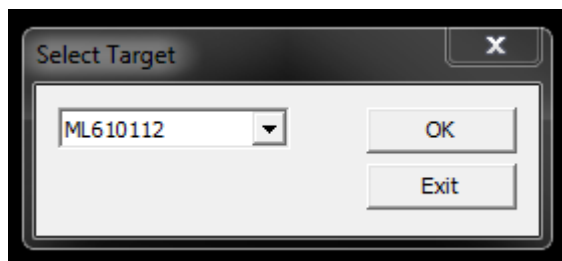


i.



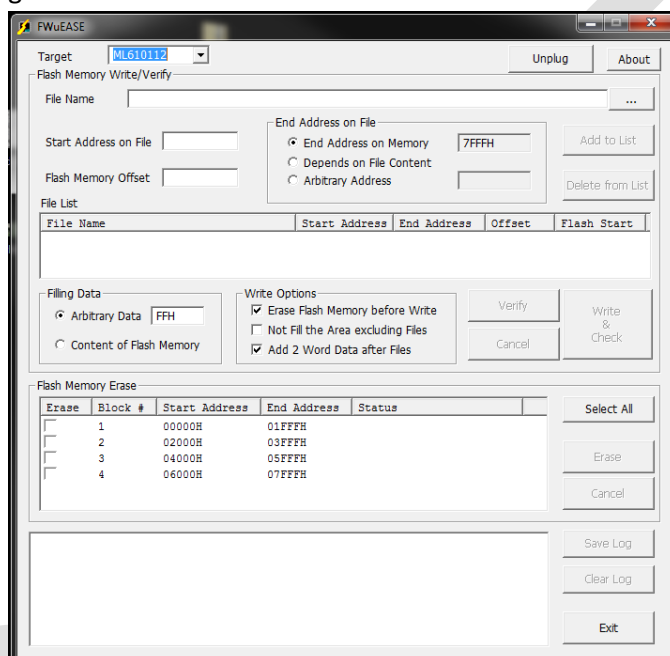
ii.

3. Move the Jumper on the J2 connector to connect the top two pins (looking at the base board with the sensor interface connector on top).
 - a. This will change the connection of the reset pin on the board from the PB (S1) to the programmer.
4. Connect the uEase/nanoEase board to the PC using a mini USB cable.
5. Download the latest HEX programming file for the latest revision firmware
 - a. This can be found at the GitHub repository page at the following link
 - i. <https://github.com/ROHMUSDC/ROHMSensorPlatformEVK>
 - b. Please note that this can be found in the following folder hierarchy
 - i. ...ROHMSensorPlatformEVK/Firmware - Base Board ML610Q112 Code/_output/_hex
 - c. Name of the file should be as follows
 - i. "Q112-SENSORPLATFORMEVK.HEX"
 - d. Be sure to save this file somewhere where you can find it later as it will be the main file used to program these boards.
6. Next, open the newly installed "FWuEASE" software. When opening this application for the first time, an error message may occur that asks you to choose a target device. Choose "ML610112" then click OK



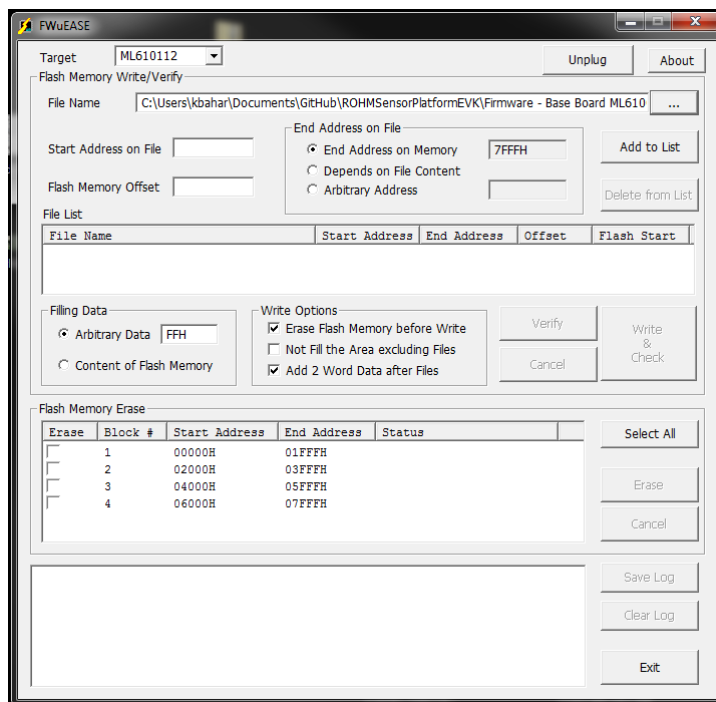
a.

7. After clicking OK above, the main application window will open. It should look like the following...



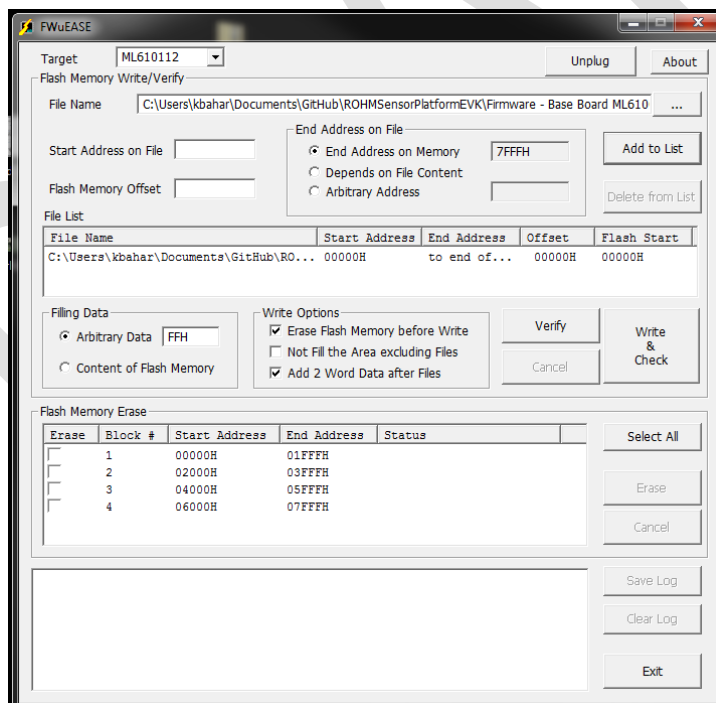
a.

8. Next, click the "..." button to the right of the "File Name" field. Browse and load the HEX file saved from step 5. Click OK and the window should look like the following



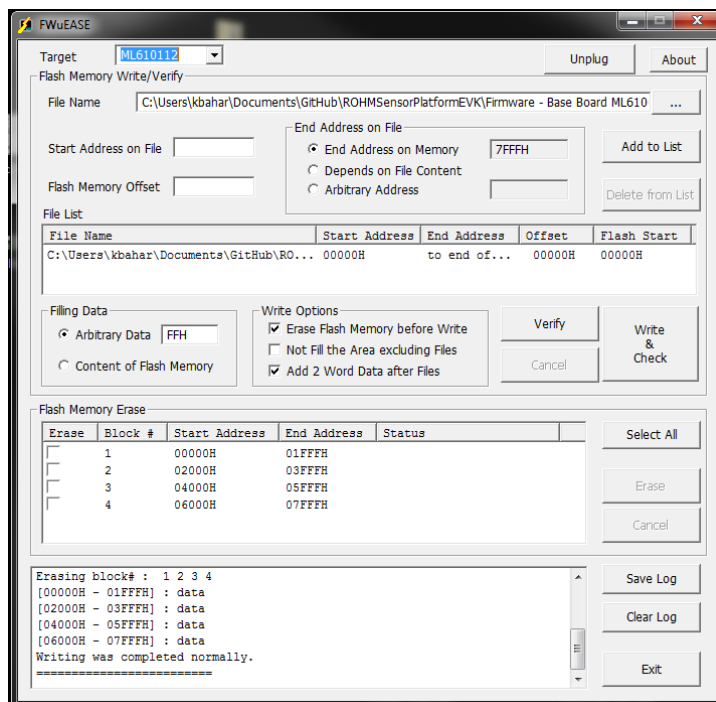
a.

9. Next, click the “Add to List” button. Also, confirm that the other setting options match the window below.



a.

10. Finally, when ready to program the device, click the “Write & Check Button. Writing will take a couple seconds to complete. Upon completion, the status output will look like the following...

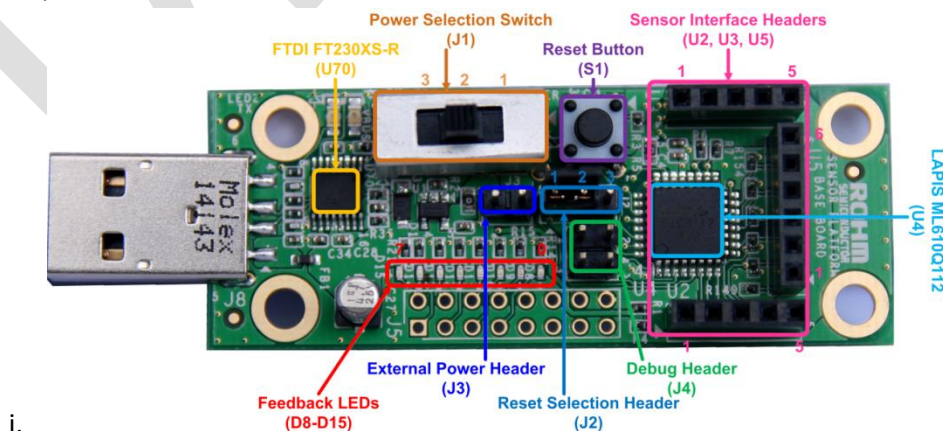


a.

11. If you see the above message, then you have successfully flashed the hardware of this device!

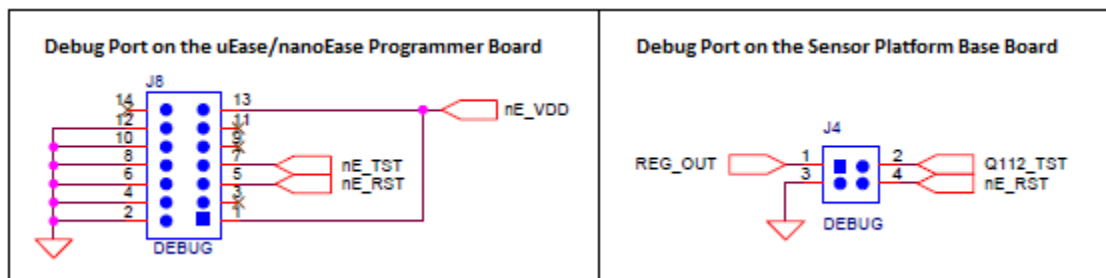
Programming and Debugging the Sensor Platform Base Kit

1. Acquire a “uEase” or “nanoEase” programmer/debugger board and install the IDEU8 GUI Development Environment as explained in the “Tools Required for Getting Started” section above.
2. Connect the “uEase” or “nanoEase” programmer/debugger board to the sensor platform base board
 - a. On the Sensor Platform Base board, we will want to connect to pin header J4 (2x2 header pin row)

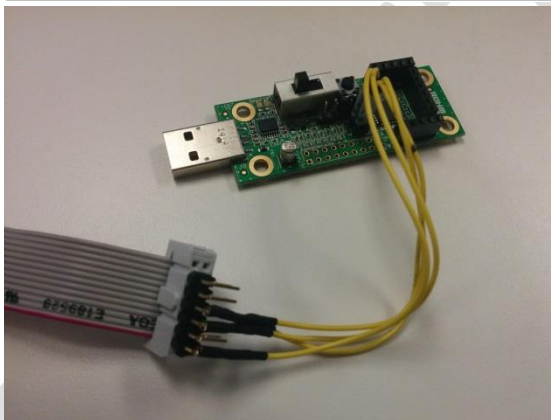


i.

- b. On the uEase or nanoEase programmer boards, we will want to isolate 4 pins (VDD, GND, TST, and RST). Then, we recommend building a jumper cable to make the connection easier to re-connect. See below for schematics for this debug port and an example debug port connector we used to flash these boards

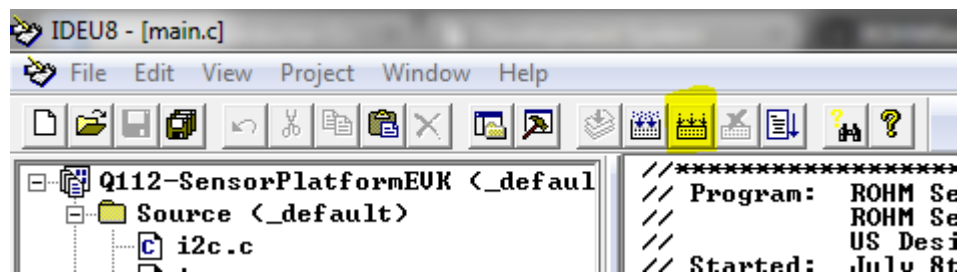


i.

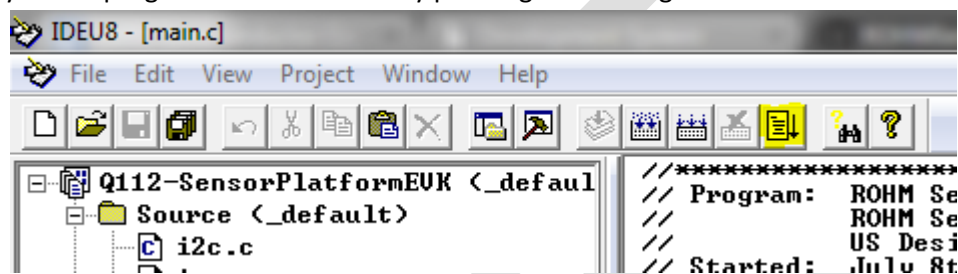


ii.

3. Move the Jumper on the J2 connector to connect the top two pins (looking at the base board with the sensor interface connector on top).
 - a. This will change the connection of the reset pin on the board from the PB (S1) to the programmer.
4. Connect the uEase/nanoEase board to the PC using a mini USB cable.
5. Next, open the newly installed *IDEU8 GUI Development Environment* and open up your target MCU's source code
 - a. If you're just getting started you can view and download the base revision of the code by downloading the firmware example from the following repository page:
 - i. <https://github.com/ROHMUSDC/ROHMSensorPlatformEVK>
 - ii. .../ROHMSensorPlatformEVK/Firmware - Base Board ML610Q112 Code
6. Next, adjust the code as required for your application and then confirm that the application complies properly by using the "Rebuild" button



- a. **OR... Project -> Rebuild**
7. Finally, you can program the base board by pressing the “debug” button



- a. **OR... Project -> Debug**
- c. **OR... CTRL+F5**
8. After the IDEU8 completes building successfully, you will see the following message in the console and a new target settings window will appear. Adjust these to the following:

- a. Successful Build Message:

```
Generating HEX file...
/H q112-sensorplatformevk.abs _output\_hex\q112-sensorplatformevk.hex

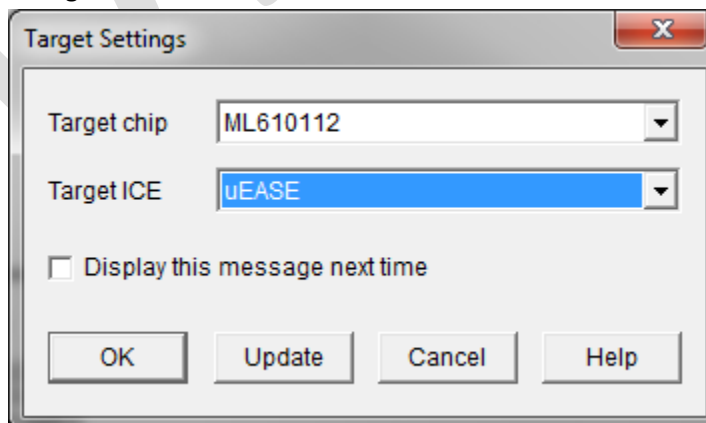
OHU8 Object Converter, Ver.1.20
Copyright (C) 2008-2011 LAPIS Semiconductor Co., Ltd.

Object was created by RL08

Convert End.

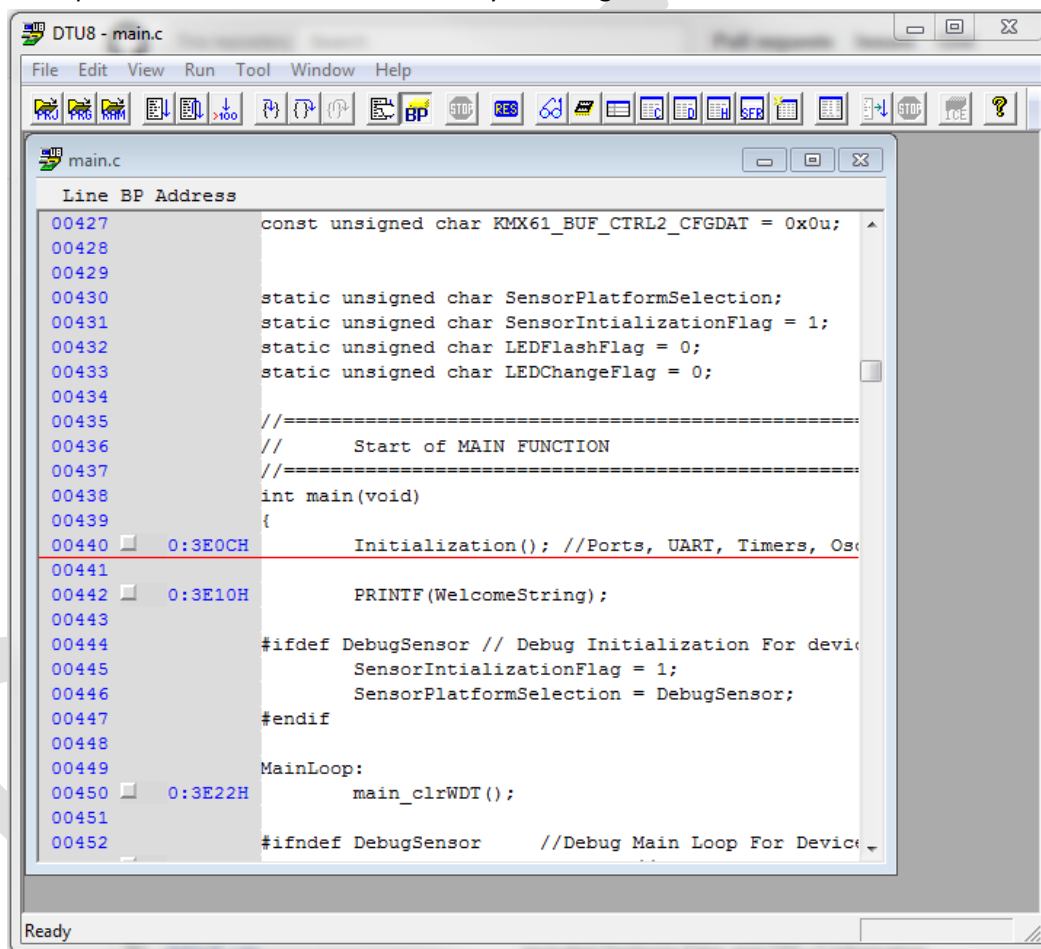
Finished building
```

- b. Target Settings to use for the base board:



i.

- ii. Note: uEASE can be chosen if a uEASE programmer is being used. However, nanoEase should be chosen if a nanoEASE programmer is being used.
- 9. After setting the target settings, the DTU8 debugger will be open with the code used to program the base board with.
 - a. Run the code using the “F5” key
 - b. Additional run commands can be found in the “Run” menu (Pause, stop, step in, step out, etc...)
 - c. Breakpoints can also be added to code by checking the “BP” box in the actual code itself



- d.
- 10. When you unplug from the debugger board, the last code that was debugged will run on the base board platform. So, feel free to disconnect and connect again to test and build the application up to your liking!

Performing a Functional Check for After Assembly

When performing a functional check of the sensor platform kit boards, we will want to check each of the following aspects of the board

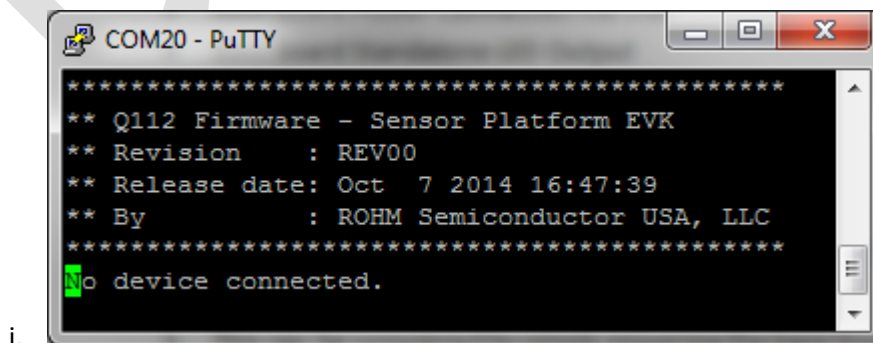
- Base Board Power Connection via USB
- Base Board Standalone LED Output
- Reset Button
- Base Board Data Communication over USB
- Each individual Sensor Output

Checking the Base Board Power Connection via USB and the Standalone LED Output

1. This can be completed by simply powering the base board through any standard USB connector
 - a. The included ROHM battery will work for this, but we recommend connecting this to a PC's USB port in order to save some time since we will want to check data communication to the board as well
2. Pass Condition
 - a. When no sensor breakout boards are connected to this board, the board will flash the onboard LEDs repeatedly.

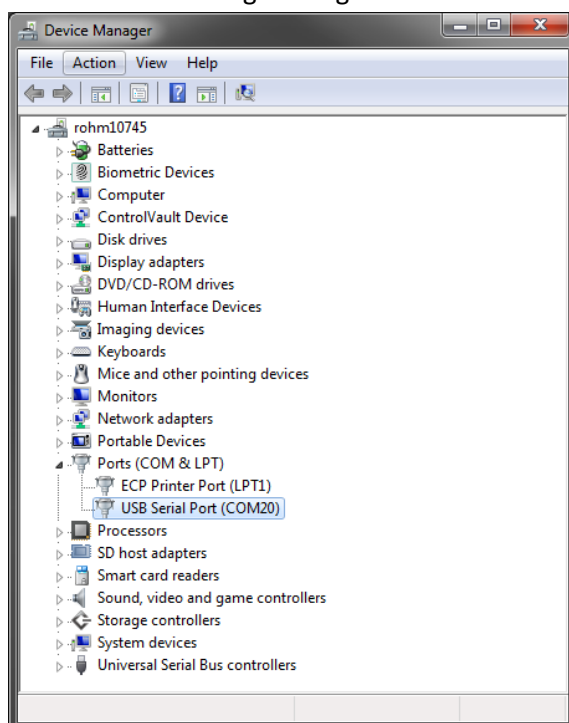
Checking the Checking the Reset Button

1. Move the Jumper on the J2 connector to connect the bottom two pins (looking at the base board with the sensor interface connector on top). Please note, that connector J2 is a three pin connector (not to be confused with the additional 2 pin connector, J3, below connector J2).
2. Pass Condition (perform **either** step a or b below)
 - a. If looking at the output LEDs, if the PB (S1) is pressed, the LEDs will stop flashing for the duration of the press. When this switch is let go, the LEDs should begin to flash again
 - b. If looking at the debug console, if the PB (S1) is pressed, all previous information provided will be cleared and the firmware information should be displayed like the picture below

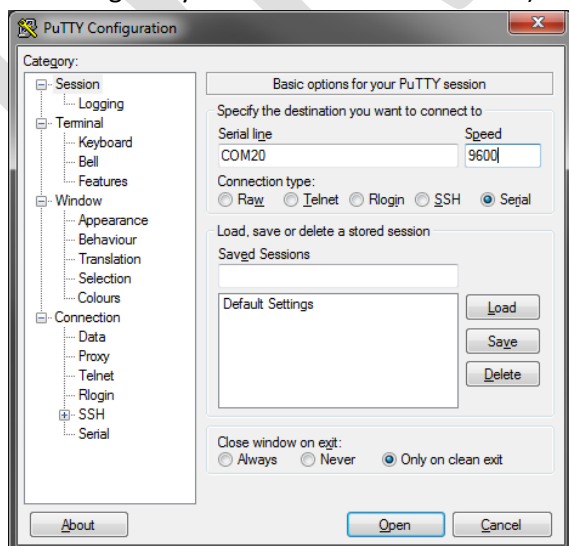


Checking Base Board Communication over USB

1. NOTE: if the reset button was checked using the PC debug console, then this section is not necessary
2. Connect the Sensor Platform Base board to the PC
3. Check the PC's device manager to figure out which COM port this device is connected to

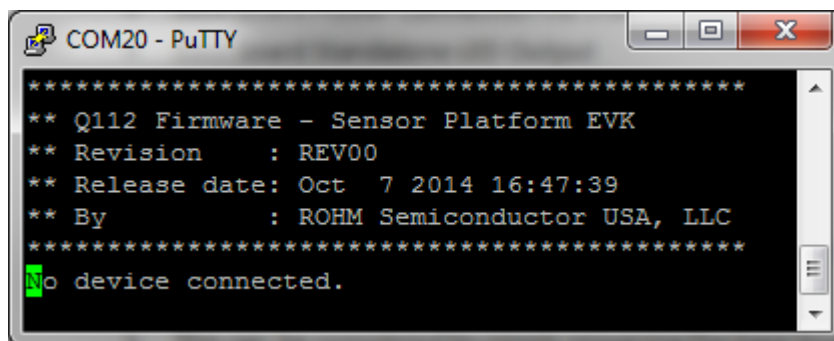


- a.
4. Open Putty (or your preferred COM terminal application). If you choose to use putty, simply check the “Serial” radio button and adjust the “serial line” to **COM20** (or whatever COM port the device manager says this device is connected to) and adjust the “speed” field to **9600**



a.

5. Upon clicking the “Open” Button, the putty configuration screen will disappear and the terminal output will be shown. Click the Reset button to show the firmware details.
6. Pass condition
 - a. This stage passes our checks if we successfully see the firmware details when pressing the reset button



i.

Checking Each Individual Sensor Output

Since this kit includes 6 sensor breakout boards, we will want to test each of the 6 included sensors. For this, we will also want to test both the standalone output (via output LEDs) and the PC output (via COM terminal). Please note, the below will assume the board is connected to a base board.

- Recommended Tools for Testing
 - Flashlight
 - Magnet
 - “Quik-Freeze” canned air
 - UV Flashlight
- ROHM BH1721 – Digital Ambient Light Sensor
 - Passing conditions
 1. Low Light condition – Cover the Sensor
 - Standalone LED’s binary output should be close to 0-1. Essentially, will be lower than ambient and high lighting conditions
 2. Ambient Light condition – use the current level of lighting
 - Standalone LED’s binary output should be higher than the Low light condition, but lower than the high light condition
 3. High Light condition – Shine a Flashlight onto the sensor
 - Standalone LED’s binary output should be higher than ambient light and low light conditions
 - If checking on the PC, the output should follow in the same fashion as the standalone LED’s binary output.
- ROHM BH1620 – Analog Ambient Light Sensor
 - Follow the same check as the BH1721 digital ALS above

- ROHM BU52011HFV – Hall Switch Sensor
 - Passing Conditions
 - 1. No Mag Field Condition
 - Standalone LED's binary output should be off (no LEDs ON)
 - The PC Terminal output should also state "Hall – No Mag Fields Detected"
 - 2. N and S polarity Mag Field condition
 - Standalone LED's binary output should show binary "1". (AKA, right most LED will be on)
 - The PC Terminal Output should also state "Hall – Mag Field Detected"
 - Note: This is an Omnipolar Hall Switch IC, thus it will trigger one output when a magnetic field is present. However, this is not a dual output hall sensor, thus it cannot determine the polarity of the magnetic field with this sensor device.
- ROHM BDE0600G – Analog Temperature Sensor
 - Passing Conditions
 - 1. Room Temperature Condition
 - Standalone LED's binary output should show a constant value. Also, this value should be higher than the low temp condition and lower than the high temp condition
 - The PC terminal will output the temperature. This should be around the temperature of the room (~25C)
 - 2. Warm Temperature Condition
 - A warm temperature condition can be seen by placing a finger on the sensor; however this should only increase the sensor output by 1-2 degrees
 - The Standalone LED's binary output and PC output should show a value higher than the room temperature value. However, since the change is not drastic, this may not be seen; thus, it is best to confirm this on the PC terminal.
 - 3. Cold Temperature Condition
 - A cold temperature condition can be seen by spraying the sensor with canned air. In our test, we used "Quik-Freeze", which dramatically changed the sensor output.
 - The Standalone LED's binary output and PC output should show a value lower than the room temperature value.
- LAPIS ML8511 – Analog UV Sensor
 - Passing Conditions
 - 1. Nominal Lighting Condition

- The nominal lighting condition can be used in the normal environment without any changes to lighting
 - This should show a constant value and passes if alternative lighting sources do not change this value
 - 2. UV lighting condition
 - The UV lighting condition can be induced using a UV LED or UV flashlight
 - The Standalone LED's binary output and the PC output should show values higher than the nominal condition when the UV light is shined upon the sensor
- Kionix KMX061 – Digital Accelerometer and Magnetometer
 - Passing Conditions - Accel
 1. Using the Standalone LEDs, we can quickly check the operation of the accelerometer portion of the IC. This can be done by tilting the board in the either longward side. The LED output should follow the direction of the tilt.
 2. Also, this can be checked using the PC terminal. When the board is flat, the "AccelZ_scaled" value should be close to 1 while the X and Y components are close to 0.
 - Passing Conditions – Mag
 1. The Mag portion cannot be tested using the standalone LEDs
 2. When looking at the PC terminal output, we can quickly test by placing the magnet over the top of the sensor. This will affect the Z output and should dramatically increase the magnitude of this value. Flipping the magnet will flip the sign.