Intel® System Studio 2016 for Microcontrollers
Update 2

Release Notes – Windows* and Linux* host

16 May 2017

Contents
1 Introduction .................................................................................................................................................. 2
2 Intended Audience ..................................................................................................................................... 2
3 What’s New .................................................................................................................................................. 2
4 Product Contents ....................................................................................................................................... 3
5 Minimum System Requirements ............................................................................................................... 3
  5.1 Hardware Requirements .......................................................................................................................... 3
  5.2 Software requirements ............................................................................................................................ 4
    5.2.1 Host Operating System ...................................................................................................................... 4
    5.2.2 Target Operating System ................................................................................................................... 4
6 Installation .................................................................................................................................................. 4
  6.1 Windows* Host ...................................................................................................................................... 4
    6.1.1 JTAG Driver Installation ...................................................................................................................... 5
  6.2 Linux* Host ........................................................................................................................................... 6
    6.2.1 Root Installation ................................................................................................................................ 6
    6.2.2 Non-root Installation .......................................................................................................................... 6
7 Intel® Integrated Performance Primitives for Microcontrollers ................................................................. 7
8 Documentation .............................................................................................................................................. 7
9 Known Limitations ...................................................................................................................................... 8
10 Technical Support .................................................................................................................................... 10
  10.1 Getting Support .................................................................................................................................... 10
11 Disclaimer and Legal Information .............................................................................................................. 10
1 Introduction
Intel® System Studio for Microcontrollers is an integrated tool suite for developing, optimizing, and debugging systems and firmware for the Intel® Quark™ microcontrollers (including the Intel® Quark™ Microcontroller D2000 and Intel® Quark™ SE Microcontroller C1000 development boards), which offers a microcontroller core to enable applications from device control to edge sensing IoT solutions. Further in this document, we will refer to the Intel® System Studio for Microcontrollers as the “suite”.

2 Intended Audience
This document is intended for the system and application developers who develop for Intel architecture-based microcontroller systems and devices.

3 What's New
This section highlights important changes in the product release.

- Now supporting Zephyr* OS and code samples for version 1.7
- Now supporting Intel® Quark™ Microcontroller Software Interface (Intel® QMSI) and code samples for version 1.4
- Added support for MRAA IO Communication Layer / UPM Sensor and Actuator Library
- New feature added for performing energy analysis on target microcontrollers
- Added Install new Firmware feature
  - Note that Intel® System Studio for Microcontrollers has been tested with the Zephyr* OS version listed above. Keep in mind that the Zephyr* repository is a work in progress, and it is not being validated on all the supported platforms on regular basis. Please roll back to the version Zephyr* OS version 1.7 or the last known good version in case of any issues
- New support for Arrow Panther* board and tinyTILE* boards
  - It is needed to update the firmware to the Zephyr* master branch using Install new Firmware feature to get the latest bits for Arrow Panther* board
- Updated Intel® Integrated Performance Primitives for Microcontrollers library, including support for Sensor Subsystem for Intel® Quark™ SE Microcontroller C1000, additional functions and code samples
- New user interface designs with workflow optimizations
More detailed information about new features and changes may be found in the respective component release notes.

4 Product Contents

The software development tool suite consists of the following components:

- Board Support Package (BSP) for Intel® QMSI
- Eclipse* Neon 4.6 including Intel® System Studio for Microcontrollers integration
- Floating Point Emulation library
- GCC* Version 5.2.1 for Intel® architecture
  - Linker
  - Assembler
  - C run-time Libraries
- GCC* Version 4.8.5 for the Sensor Subsystem in Intel® Quark™ SE Microcontroller C1000
  - Linker
  - Assembler
  - C run-time Libraries
- GNU GDB* 7.9
- Java* 64 bit Standard Edition Runtime Environment 1.8
- Intel® Compiler for Intel® Quark™ Microcontrollers - 2016.0
- Energy Analysis for Microcontrollers
- Intel® Integrated Performance Primitives for Microcontrollers 1.2.0
- Intel® QMSI 1.4.0
- MRAA IO Communication Layer / UPM Sensor and Actuator Library
- OpenOCD* 0.8.0 JTAG Debugger
- Python* 2.7.10 and 3.6.0
- Sample Applications
- Standard and optimized math library [libm]
- TinyCrypt* 0.2.6 (Internet connection required during installation)
- WinUSB* driver for Intel® Quark™ microcontrollers
- Zephyr* OS 1.7.0

Note: Refer to the file “third-party-programs.txt” included with the package for more information.

5 Minimum System Requirements

5.1 Hardware Requirements

- One of the following development boards:
- Intel® Quark™ Microcontroller D2000 development board
- Intel® Quark™ SE Microcontroller C1000 development board
- Arduino*/Genuino* 101 board
- tinyTILE* board
- Arrow Panther board*

- Host system with 2 GB RAM or greater (See software requirements for supported OS versions)
- USB cable (Type A to Micro-B if using Customer Reference Board)
- Optional: FTDI-cable (e.g. model TTL-232R-3V3 works well) is used to send and receive serial data.
  Find out more about, or purchase, an FTDI cable here: http://www.ftdichip.com/Products/Cables/USBTTLSerial.htm

5.2 Software requirements

5.2.1 Host Operating System
Supported host OS:
- Linux (64bit: Ubuntu* 16.04 LTS and Fedora* 25)
- Windows* (64bit: Versions 7, 8.1, and 10)

5.2.2 Target Operating System
Supported target environments:
- Bare Metal with Intel® Quark Microcontroller Software Interface 1.4.0
- Zephyr* OS 1.7.0

6 Installation

6.1 Windows* Host
1. Ensure you have met the prerequisites:
   a. The package supports x64 host architecture only.
   b. The Intel® Quark™ microcontroller development board should be plugged in the host PC in order to properly install the Microsoft* WinUSB driver for JTAG (alternatively, reinstall the driver later – see section 6.1.1)
   c. Internet connection is required (otherwise the installation of the TinyCrypt component of the package will not be successful).

2. After downloading the package double click on the installer to run it:
   w_cembd_mv_p_1.0.y.xxx.exe
6.1.1 JTAG Driver Installation

If you have any problems flashing or debugging your development board, it might be related to the JTAG driver.

The installer should automatically install the correct JTAG driver for your board if it is connected during setup. **If you didn't connect your board before install or if you change to a new board after install you may need to manually install the driver.** On some systems this may fail due to existing drivers with the same identifiers (VID&PID). This is often the case where Windows has automatically installed drivers from FTDI either from Windows Update or from users installing the FTDI package. These drivers assume the device is to be used in serial mode where OpenOCD requires them to be in JTAG (WinUSB) mode.

**Warning: These steps may interfere with other FTDI devices in use.**

JTAG Driver Manual Installation Steps:

a. Connect your board (micro USB) to your host machine (USB) and run the driver installer to set up the Microsoft* WinUSB driver for JTAG:

   ```
   C:\IntelSWTools\ISSM_2016.y.xxx\tools\debugger\driver\install.bat
   ```

   This will attempt to install the correct drivers for the debugger device.

b. If OpenOCD can still not find the device, removal of existing drivers is required:
   i. Open Device Manager (WIN + X).
   ii. Locate the driver either within the “Universal Serial Bus controllers” group, “USB Serial Converter”, “USB Serial Converter A” or “USB Serial Converter B”.
   iii. Right click on the device and select Uninstall. Make sure to tick the “Delete the driver software for this device”. Replug the device.
   iv. Repeat this process until Windows recognizes the device as Unknown, at which point run the driver “install.bat” shown in step “a”.
   v. You should now see the device installed as “OpenOCD JTAG”

c. If OpenOCD fails to detect the device even with the “OpenOCD JTAG” device visible in the device manager here are some troubleshooting steps you can try:
   i. Re-plug your target/debugger.
   ii. Run the install.bat described in step “a”. (Whilst device is connected)
   iii. Remove existing FTDI drivers as described in step “b”.
   iv. Reboot your machine

The default installation path **ISSM_ROOT** is:
Then you can launch the suite by running the launcher script in **ISSM_ROOT**:

`iss_mcu_ide_eclipse-launcher.bat`

### 6.2 Linux* Host

1. Ensure you have met the prerequisites:
   a. The package supports x64 host architecture only.
   b. The Intel® Quark™ microcontroller development board should be plugged in the host PC in order to properly install the driver for JTAG installs correctly.
   c. Internet connection is required (otherwise the installation of the TinyCrypt component of the package will not be successful).

2. Download the installation package and follow these steps to start the installation:

#### 6.2.1 Root Installation

1. Decompress the package
   
   `tar -xvf ./l_cembd_mv_p_1.0.y.xxx.tgz`

2. Choose to install via GUI (option “a”) or command line (option “b”)
   a. Start the installer in GUI mode
      
      `./l_cembd_mv_p_1.0.y.xxx/install_GUI.sh`
   b. Alternatively force the installer to install in text mode
      
      `./l_cembd_mv_p_1.0.y.xxx/install.sh --cli-mode`

#### 6.2.2 Non-root Installation

If you installed as user (nonroot), then complete these two additional steps:

1. Set the USB rules separately, to allow you to flash and debug your target:
   a. Run this script to install the driver:
      
      `~/intel/issm_2016.y.xxx/tools/utils/install_driver.sh`
   b. Then the board should be unplugged and plugged back in, after the driver is installed by root, so that the rules in .rules file can apply.

2. Set user permissions for serial port usage
a. To reliably access serial devices, it is best to add your user to the "dialout" Linux* group. The following will work on most Linux* distributions:

```bash
sudo usermod -aG dialout <your-username>
```

Finally, the default installation path ISSM_ROOT is:

For sudo/root install: `/opt/intel/issm_2016.y.XXX`
For user install: `~/intel/issm_2016.y.XXX`

Note: To refresh the IDE: Close all GDB sessions and restart the JTAG OpenOCD session using the "double arrows" button.

Then you can launch the suite by running the launcher script in ISSM_ROOT:

```bash
iss_mcu_ide_eclipse-launcher
```

7 Intel® Integrated Performance Primitives for Microcontrollers

To enable the use of Intel® Integrated Performance Primitives (Intel® IPP) library in your project the following must be added to the project Makefile.

```bash
CFLAGS += -I$(IPP_LIBRARY_PATH)/include -I$(IPP_LIBRARY_PATH)/examples -D__IPP_ENABLED__=1
LDLIBS += -L$(IPP_LIBRARY_PATH)/lib/$(BUILD)/$(SOC)/$(TARGET) -lippsq
```

This adds the correct Compiler and Linker flags for using the included Intel® IPP installation. Please see the Intel® IPP documentation for details regarding Intel® IPP API usage.

8 Documentation

After completing the download of Intel® System Studio for Microcontrollers documentation can be found in the “docs” under the installation path.

Additionally we provide websites with guides to set up and use this tool suite with the Intel® Quark™ Microcontroller D2000 and the Intel® Quark™ SE Microcontroller C1000. For further information on using the Arduino*/Genuino* 101 board see:


To learn more about the Intel® Quark™ microcontroller Development Platforms we recommend you visit [this page](https://www.arduino.cc/en/Guide/Arduino101).


The [User and Reference Guide](https://www.arduino.cc/en/Guide/Arduino101) gives more detailed information about this tool suite and the
command line options to build applications with Intel® Quark™ Microcontroller D2000 and Intel® Quark™ SE Microcontroller C1000.

9 Known Limitations

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
<th>Implication</th>
<th>Workaround</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eclipse IDE</td>
<td>Source level debugging cannot be performed when using the build_release configuration.</td>
<td>No source code will be displayed if build_release is selected for debug.</td>
<td>Select build_debug configuration before performing a debug session.</td>
</tr>
<tr>
<td>Eclipse IDE</td>
<td>Message about a missing source file appears in red in the GDB console when launching a debug session if multiple projects are setup and the project being debugged is not the 1st listed.</td>
<td>No impact on actual debugging.</td>
<td>To avoid the message only setup a single project per workspace.</td>
</tr>
<tr>
<td>Eclipse IDE</td>
<td>Serial Terminal might not release the lock for the used port when closed on Linux.</td>
<td>It won’t be possible to open a new Terminal on a locked port.</td>
<td>Remove the lock file in /var/lock manually.</td>
</tr>
<tr>
<td>Intel® QMSI Power SoC Sample</td>
<td>Error when running or debugging ‘power_soc’ sample application in ISSM Eclipse IDE (Warning: Cannot insert breakpoint 5. Cannot access memory at address 0x40030175)</td>
<td>Sample will flash to board, but debug will not be available.</td>
<td>Add a delay of 2 or more seconds to the start of the code sample before putting the system to sleep.</td>
</tr>
<tr>
<td>Intel® Compiler for Intel® Quark™ Microcontrollers</td>
<td>When using Intel® Compiler for Intel® Quark™ Microcontrollers to build Zephyr 1.7 samples may see the warnings turned into errors for sample:</td>
<td>This sample will fail to build with default configuration.</td>
<td>To suppress the warnings and avoid the errors you can add this flag to the build command: -Wno-pointer-sign</td>
</tr>
<tr>
<td>Component</td>
<td>Description</td>
<td>Implication</td>
<td>Workaround</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>Energy Analysis for Microcontrollers</td>
<td>Running Energy Analysis on applications with SOC Sleep transitions in Intel® Quark™ SE Microcontroller C1000 platform cause application to freeze.</td>
<td>Energy Analysis cannot trace SOC Sleep transitions.</td>
<td>User has to disable SOC sleep transitions in application while running energy analysis. In Zephyr OS 1.7 the CONFIG entry CONFIG_SYS_POWER_DEEP_SLEEP need to be set to &quot;n&quot; to disable SOC Sleep transitions.</td>
</tr>
<tr>
<td>dfu-util</td>
<td>Normally dfu-util package includes the SHA-256 checksums for the pre-built binaries. However it is not included in this release package.</td>
<td>Impossible to verify pre-build binaries checksums.</td>
<td>The SHA-256 checksums for the dfu-util binaries provided below.</td>
</tr>
</tbody>
</table>
| ZUPM/ZMRAA Sensors Library | Projects for the Intel® Quark™ SE C1000 Developer Board do not support the ZUPM/ZMRAA examples and sensor view by default. | No easy method to create ZUPM/ZMRAA based projects for Intel® Quark™ SE Microcontroller C1000. | Create a ZUPM/ZMRAA based project for Arduino* 101, and follow the steps below to change the platform to Intel® Quark™ SE C1000 Developer Board:  
- Right click on the project  
- Click on Properties  
- Select “Platform” from the left pane  
- Select “Intel® Quark™ SE C1000 Developer Board” from the developer board drop down  
- Click OK  
- Build the project. |

For specific silicon errata visit: [http://www.intel.com/quark](http://www.intel.com/quark) and in the Documentation section for your platform select the Specification Update document.

dfu-util SHA-256 checksums:

<table>
<thead>
<tr>
<th>checksum</th>
<th>dfu-util-qda.exe</th>
<th>checksum</th>
<th>dfu-util-qda.exe</th>
</tr>
</thead>
<tbody>
<tr>
<td>029521c573d420343c718ebac89619de5a7b6703ec22ff06353fe3472927fd08</td>
<td>a7c39d985d817ef4384e4fd8dcfdba01d32e33c54ffdc1f1f76a92a315a342fb</td>
<td>dfu-util-qda.exe</td>
<td>dfu-util-qda.exe</td>
</tr>
</tbody>
</table>
10 Technical Support

Your feedback is very important to us. To learn more about our tools or to ask questions you can visit our main product page, our product support page, our FAQ page or visit the software forum. For hardware questions we recommend visiting http://www.intel.com/quark.

10.1 Getting Support

To received technical support for the tools provided in this product and technical information please file a support request at Online Service Center (http://www.intel.com/supporttickets).

NOTE: Registering for support varies for release product or pre-release products (alpha, beta, etc.) – only released software products have support web pages at http://software.intel.com/sites/support/.

11 Disclaimer and Legal Information

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at Intel.com, or from the OEM or retailer.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel, the Intel logo, Quark, and Curie are trademarks of Intel Corporation in the U.S. and/or other countries.
Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice Revision #20110804

*Other names and brands may be claimed as the property of others

© Intel Corporation.