美國原裝 DevPack for SimpleLink SensorTag Debugger (CC-DEVPACK-DEBUG)

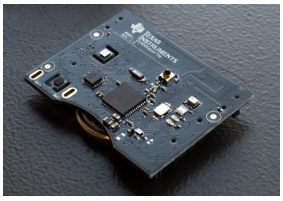
Description

The Debugger DevPack adds debug capability to your SensorTag. Plug it into the SensorTag DevPack expansion header and debug your SensorTag with Code Composer Studio or IAR ARM development environments. The Debugger DevPack includes a FREE license of Code Composer Studio that is tied to the DevPack.

The Debugger DevPack includes a USB power connection, making it easy to power your SensorTag during debugging. Alternatively, simply connect it to a USB power supply if your sensor application requires permanent power. It also includes traces for three Grove connectors to make it easy to add support for any of the 100s of Grove sensors and actuators.

The Debugger DevPack is a small form-factor XDS110 debugger. It can an be used with IAR and CCS.

How To Setup A Development Environment For The CC2650STK SensorTag



There is a lot of information on TI's website on how to get your CC2650 Sensortag ready to connect to a development environment which is quite overwhelming. There are still a few bugs in the software which the SensorTag team is working on. In the mean time, this guide will help you get setup with a development environment for the CC2650STK.

Warning: The binary/hex file that is generated with TI's code used in this write-up has a bug and it will not work with TI's Android application, due to a versioning error. Read the end of this article for a workaround.

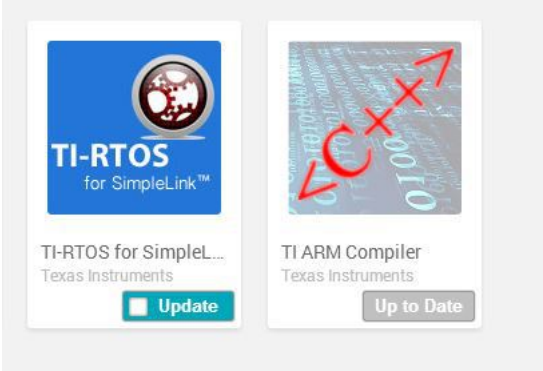
---

1. Required Hardware:

- CC2650STK Sensortag
- SensorTag DevPack(CC-DEVPACK-DEBUG)

---

2. Required Software

- The latest Code Composer Studio
  - If you already have CCS, make sure you update to the latest version via Help->Check for updates.
  - CCS switches to a free version once you have the Debug DevPack connected.
- TI-RTOS for SimpleLink and TI-ARM compiler
  - The above two can be downloaded from TI's application center from within CCS.
  - Goto Help->CCS App Center
  - Under Code Composer Studio Add-Ons, you should see the following. Install them:
  - 

- BLE-STACK2
  - Remember to download BLE-STACK2, not BLE-STACK.
  - Install the stack using recommended settings. On my machine it defaults to c:\ti\simplelink

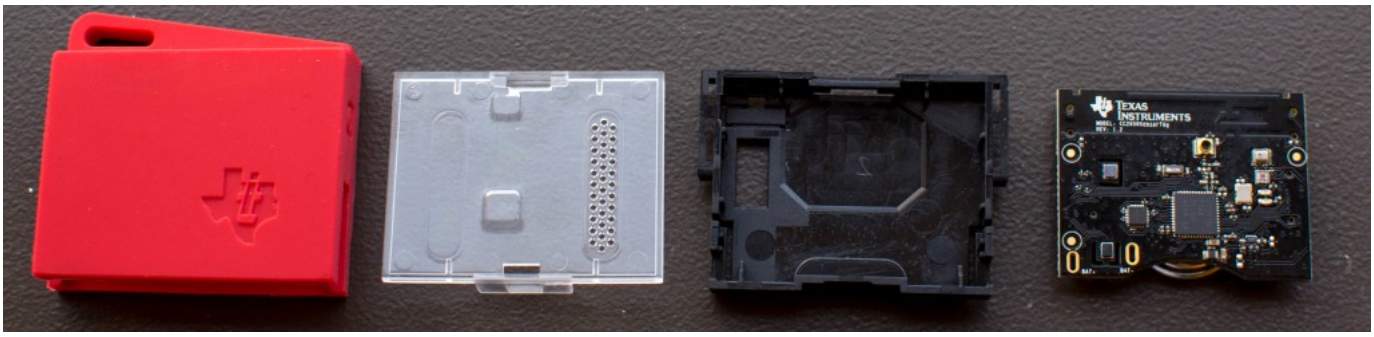
---

3. Connecting Your SensorTag Hardware

This is kind of tricky. There are two ways to go about this.

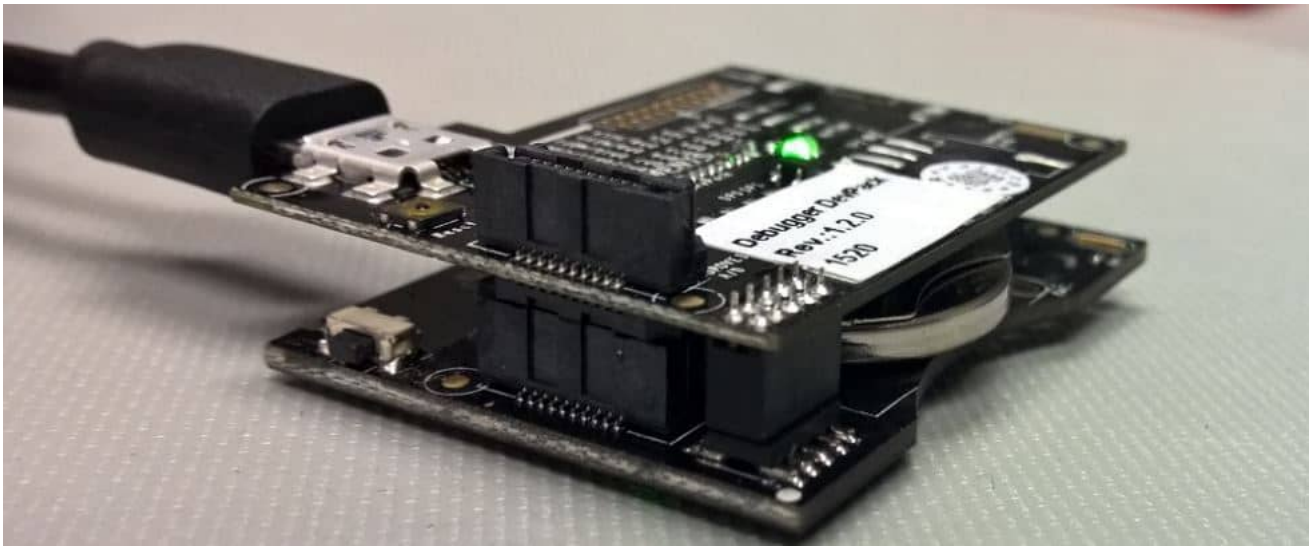
3.1 Connecting the Tag directly to the Debug Devpack

Separate your SensorTag into its different parts, so that you only have the main Tag PCB. Now align the SensorTag 10 pin connector and the 20-pin SKIN connector as shown below. Do NOT remove the battery! The LED should stay on when you connect the USB cable to the PC.



Warning: It is highly recommended not to leave the Debug DevPack connected to the SensorTag without the USB cable as it can damage the Debug Devpack pins, with the coin cell inserted.

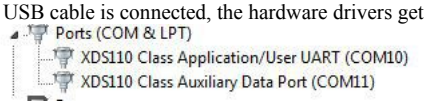
> *It is not recommended to use the Debug DevPack without the USB power attached due to the risk of powering the Debugger from the coin cell battery on the SensorTag. The SensorTag battery voltage will be disconnected from the Debug DevPack power supply but there is a risk that the I/O from the CC2650 will power the MCU on the debugger. The following I/O DevPack signals are connected to the MCU on the DevPack*



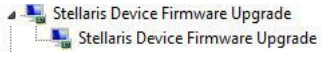
3.2Breaking the rear cover tab

Another way to connect the SensorTag2 while it is in it's casing is by breaking the tab on the rear cover. There seems to be a few places where you can cut the tab off. For my setup, I went the above route.

Once the USB cable is connected, the hardware drivers get installed if Step 2 was followed correctly. The Device Manager should show the following XDS110 devices:

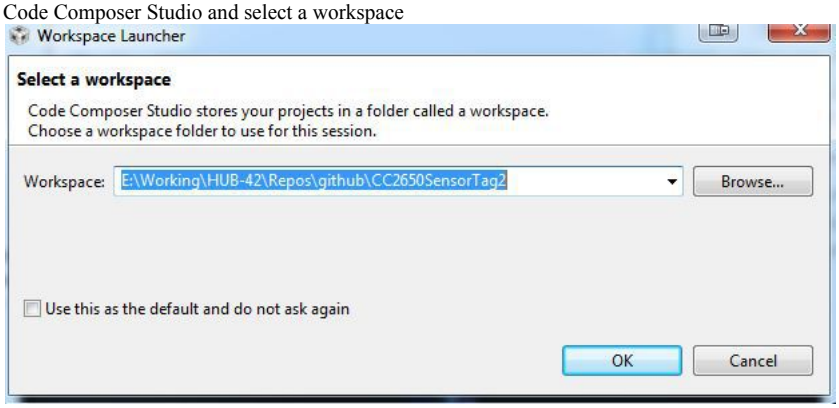


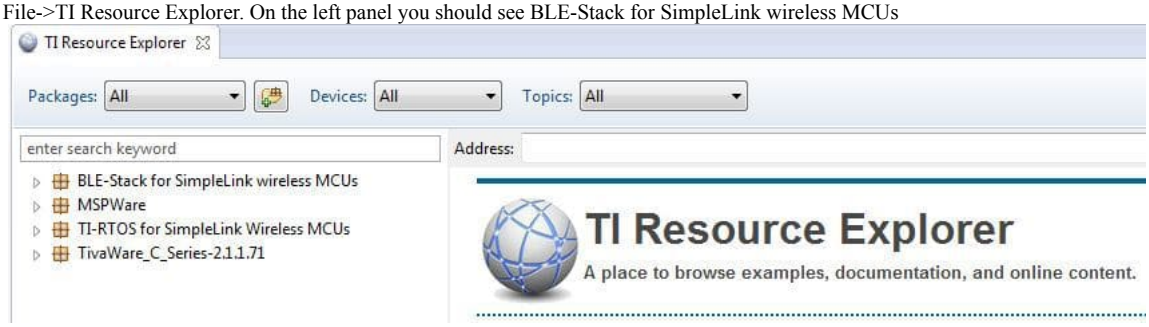
Note that if you do not have the coin cell inserted into the SensorTag, the Debug DevPack comes up as a DFU or “Stellaris Device Firmware Upgrade” for upgrading the TivaC on the Debug DevPack



4.Loading Demo Code On The SensorTag

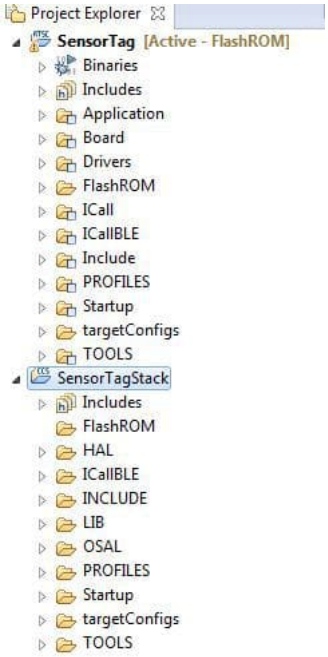
The fun part!

- Open Code Composer Studio and select a workspace
  - 

- Goto File->TI Resource Explorer. On the left panel you should see BLE-Stack for SimpleLink wireless MCUs
  - 

- Expanding “BLE-Stack for SimpleLink wireless MCUs” should give you a list as shown below. Select SensorTagApp and click ” Import the example project into CCS”
  - 

- Perform the same step for the SensorTagStack project. On import you should see the two projects on the left panel.Now, all that is needed is the SensorTagApp project. The SensorTagStack project is provided if you are brave enough to make changes to the Bluetooth Low Energy stack. TI separated these two to make it modular and ease development.



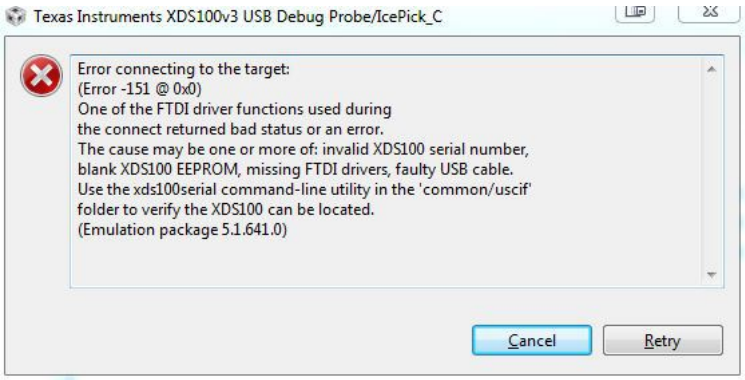
- Right click the SensorTagApp project and click build. It will take some time to build the first time. On completion, you should see a “Build Finished” output.

5.Debugging the SensorTag

Debug can be started by clicking the “bug” drop down and selecting Debug as->Code Composer Debug Session.

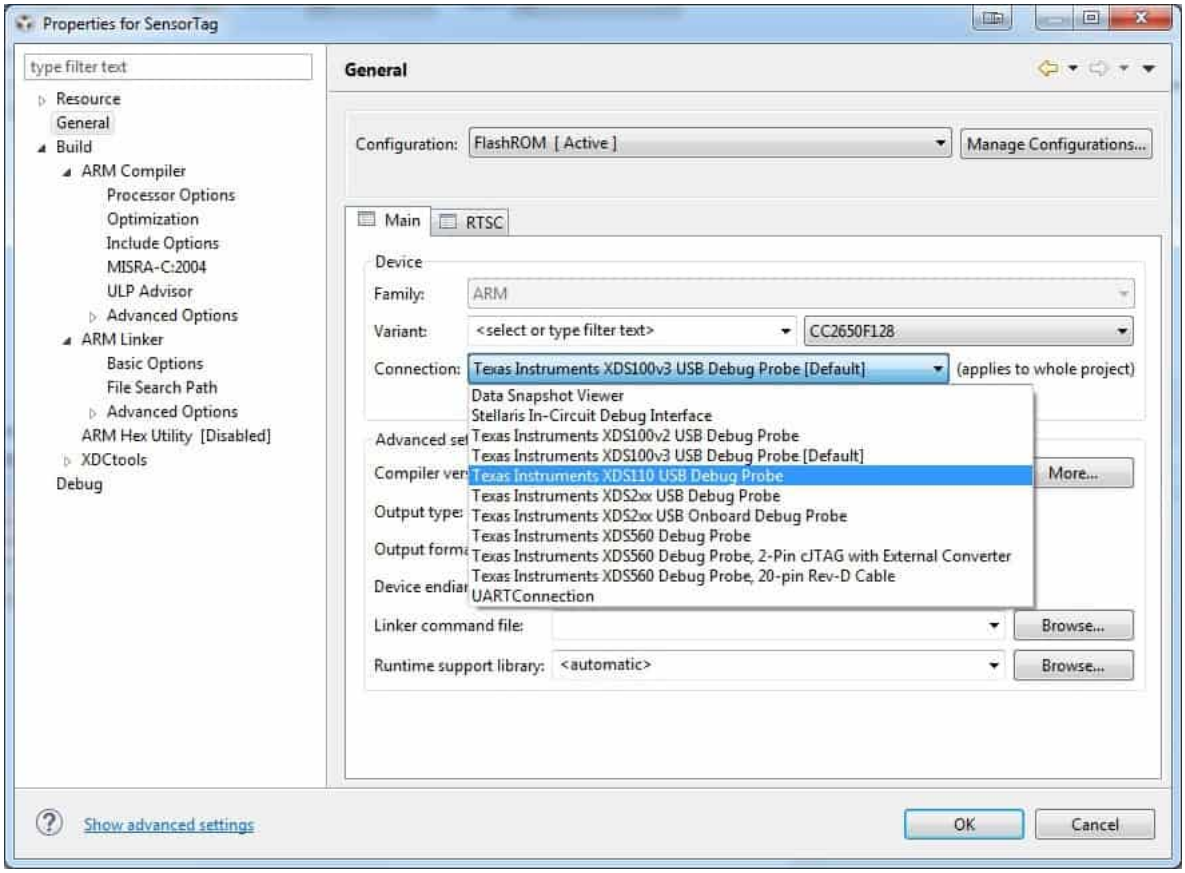
The first time you try to debug the SensorTag, a XDS100 error will pop up.

> *Error connecting to the target: (Error -151 @ 0x0) One of the FTDI driver functions used during the connect returned bad status or an error. The cause may be one or more of: invalid XDS100 serial number, blank XDS100 EEPROM, missing FTDI drivers, faulty USB cable. Use the xds100serial command-in the ‘common/uscif’ folder to verify the XDS100 can be located. (Emulation package 5.1.641.0)*

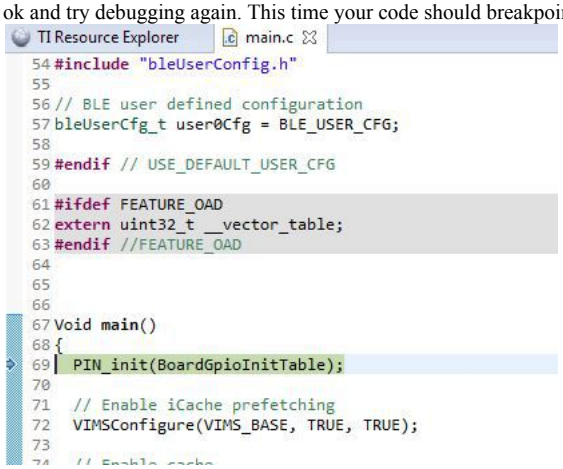


This is due to the fact the stock SensorTagApp project was setup to use the XDS100 debugger, but the debug DevPack comes with the XDS110. This will need to be changed as shown below.

- Right click the SensorTag project and click properties.
- Select “General” and change the Connection from “Texas Instruments XDS100v3 USB Debug Probe” to “Texas Instruments XDS110 USB Debug Probe”

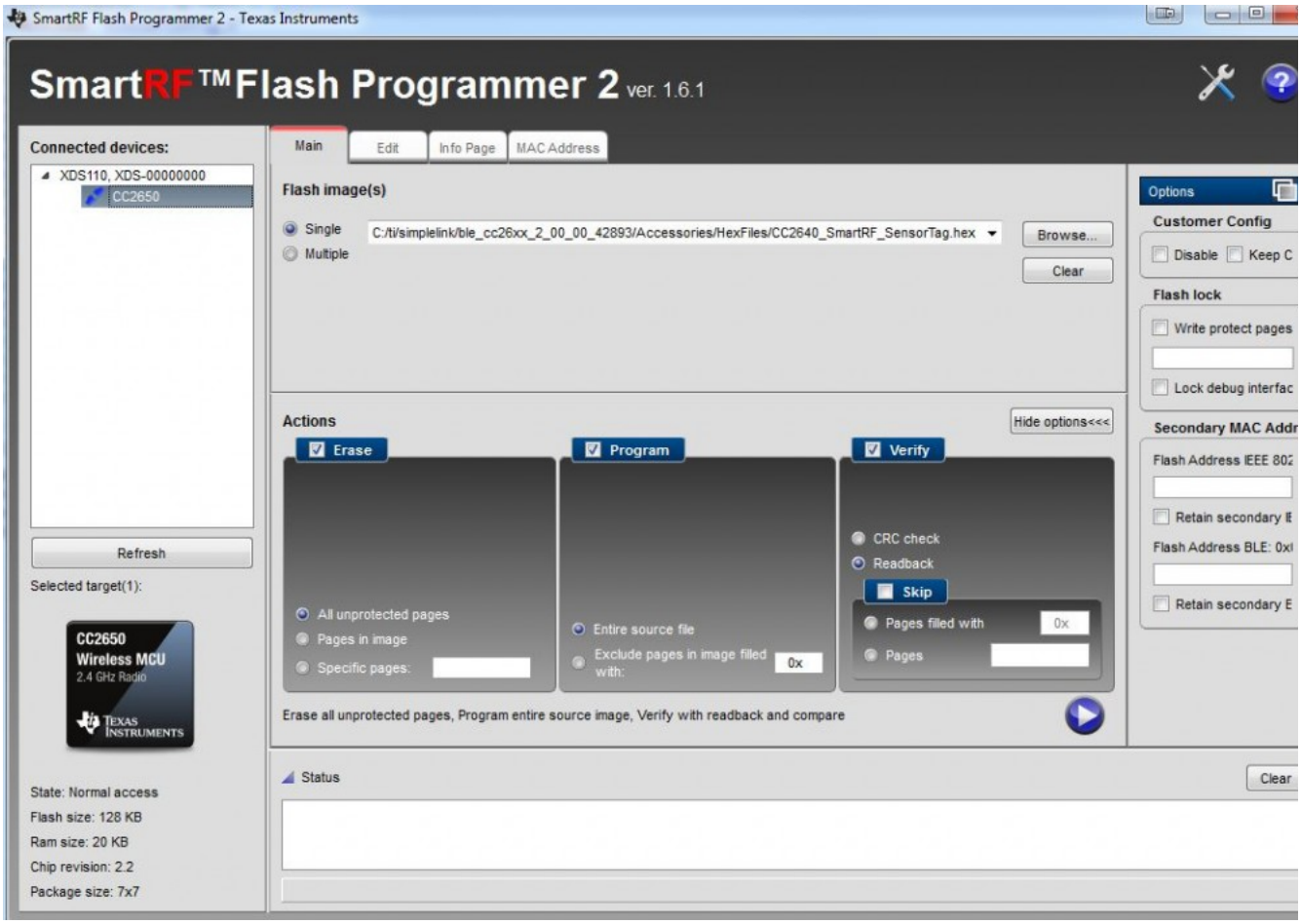


- Click ok and try debugging again. This time your code should breakpoint at *main()*.



Please note that although you have code burned onto the CC2650 SensorTag and are excited to try it with TI's Android application, the application will error out. This is due to a version check bug as described here. The good news is that you can restore the SensorTag to factory by either of the following two ways:

- Pressing the two user buttons for six seconds and release to hear a beep…OR
- Programming the CC2640_SmartRF_SensorTag.hex file which can be found under: C:\ti\simplelink\ble_cc26xx_2_00_00_42893\Accessories\HexFiles
  - This firmware can be downloaded via the SmartRF Flash programmer-2



Hope you liked this how-to. If you have any questions