CC Debugger

User's Guide

TEXAS
INSTRUMENTS

Contents

List of Figures

List of Tables

CC Debugger

1 Introduction

The CC Debugger is primarily used for Flash programming and debugging software running on CCxxxx 8051-based System-on-Chip (SoC) devices from Texas Instruments. The PC tools available for these purposes are the SmartRF™ Flash Programmer [9] from Texas Instruments and IAR Embedded Workbench® for 8051 from IAR Systems [15].

When connected to the debugger, the SoC devices can be controlled directly from SmartRF™ Studio [8]. SmartRF Studio is also able to control supported CCxxxx RF transceivers (CC2520, CC2500, CC110x, CC11xL, CC112x, CC120x) when they are connected to the debugger as explained in Section 6.3.

In addition, the CC Debugger is used for configuring the CC85xx devices with the PurePath Wireless Configurator [12] and controlling them with the PurePath Wireless Commander [13].

2   Abbreviations and Acronyms

CSn	Chip Select (active low)
DC	Debug Clock
DD	Debug Data
DUT	Device Under Test
GND	Ground
LED	Light Emitting Diode
MISO	Master In Slave Out
MOSI	Master Out Slave In
RF	Radio Frequency
SCLK	Serial Clock
SoC	System-on-Chip
SPI	Serial Peripheral Interface
USB	Universal Serial Bus
V_{DD}	Positive voltage on target

3   Box Contents

- 1 x CC Debugger
- 1 x USB-A to Mini-B USB cable
- 1 x 10-pin flat cable with 2x5 2.54 mm connector
- 1 x 10-pin flat cable with 2x5 1.27 mm connector
- 1 x Converter board 2.54 mm – 1.27 mm connector
- Documentation

4   Operating Conditions of the CC Debugger

Minimum target voltage:	1.2 V	
Maximum target voltage:	3.6 V	
Operating temperature:	0°C to 85°C	
Regulated voltage on CC Debugger:	3.3 V	
Maximum target current: [1]	200 mA [1]	
Supported Operating Systems:	Microsoft® Windows® 2000 operating system	
	Windows XP SP2/SP3 operating system	(32-bit versions)
	Windows Vista® operating system	(32 and 64 bit)
	Windows 7 operating system	(32 and 64 bit)
	Windows 8 operating system	(32 and 64 bit)

[1]   Only applicable if the target is powered from the CC Debugger.

5 Initial Steps

5.1 Installing the USB Driver

To get the required USB driver for the CC Debugger, it is necessary to install one of the tools listed below:

- SmartRF Studio [8]
- SmartRF Flash Programmer [9]
- SmartRF Packet Sniffer [10]
- PurePath Wireless Configurator [12]
- PurePath Wireless Commander [13]

Alternatively, you can download *Cebal – CCxxxx Development Tools USB Driver for Windows x86 and x64* [4], which is a standalone installer including only the device driver.

After the driver is installed, connect the CC Debugger to the PC. The USB driver is installed automatically. You can quickly check that the debugger has been associated correctly with the USB device driver by opening the Windows Device Manager. The debugger should appear as a "Cebal controlled device".

Figure 1. Verify Correct Driver Installation

For further details or troubleshooting the driver installation process, see the *Cebal – CCxxxx Development Tools USB Driver Installation Guide* [5].

5.2 Supported PC Tools

Currently, the CC Debugger can be used together with the following PC Tools:

- **IAR Embedded Workbench for 8051**: In circuit debugging of system-on-chips
- **SmartRF Flash Programmer**: Flash programming of system-on-chips
- **SmartRF Studio**: RF testing of radio devices (transceivers and SoCs)
- **SmartRF Packet Sniffer**: Packet sniffing with selected radio devices
- **PurePath Wireless Configurator**: Programming of CC85xx devices
- **PurePath Wireless Commander**: Advanced control of CC85xx devices

The debugger operates as the interface between the RF device and the tools listed above. Ensure that the correct connection is made between the device and CC Debugger before starting to use the tools.

The connection of the device to the CC Debugger is covered in Section 6.

6   Connecting the CC Debugger to the Device

6.1   Target Connector Details

The target connector, located on the lateral side of the debugger, is a 10-pin 2x5 2.54 mm pitch connector with a direction coded plastic guide. Suggested matching (male) surface mounted headers would be 95278-101A10LF from FCI or BB02-HP from GradConn.
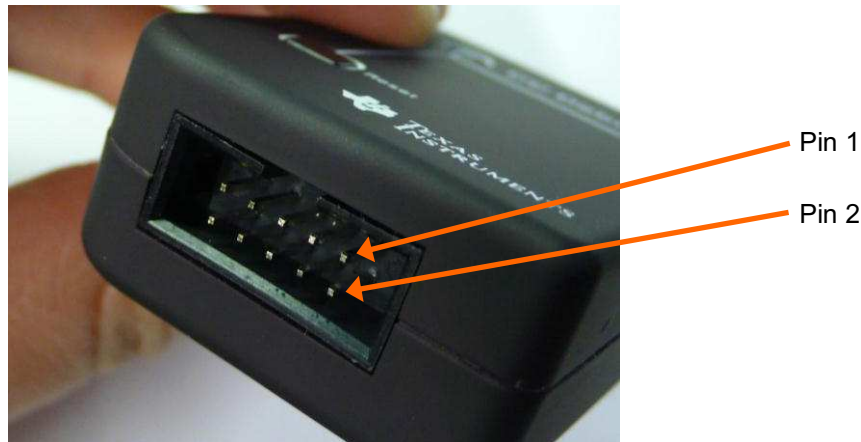

Figure 2. Placement of Target Connector Pins

The adapter board, which has a 10-pin 2x5 1.27 mm pitch connector, has the same pin placement. Suggested matching (male) surface mounted headers would be 20021121-000-10C4LF from FCI or FTS-105-01-F-DV from Samtec.

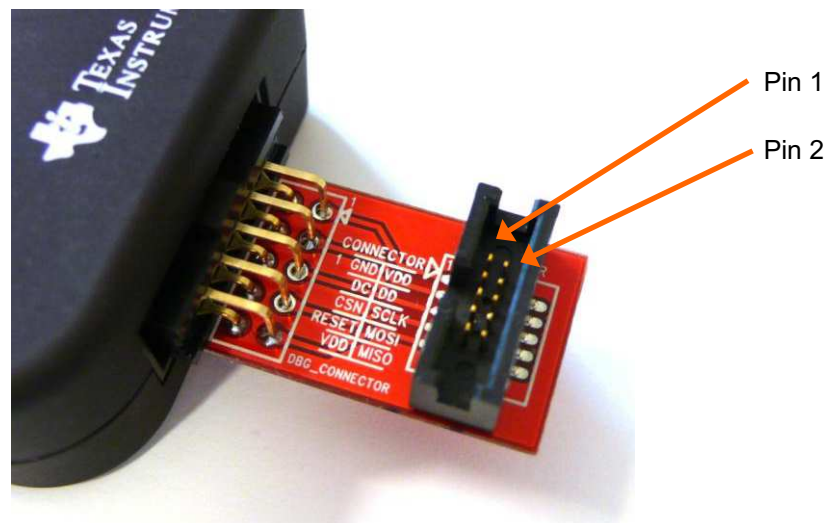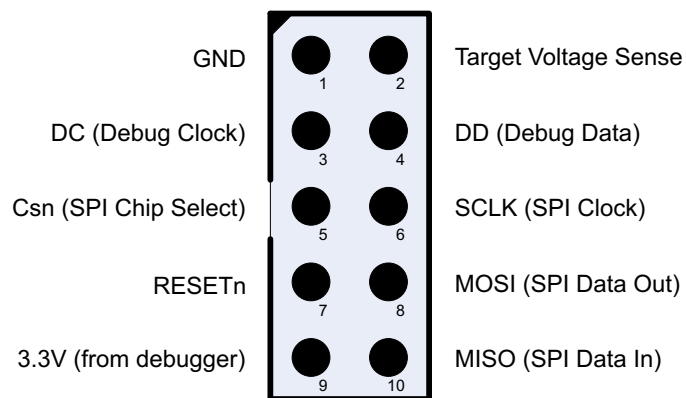**Figure 3. Placement of Target Connector Pins on Adapter Board**

The pin-out of the target connector is shown in Figure 4. Note that not all of these pins need to be connected to the target device for programming and debugging. Only VDD, GND, DD, DC and RESET are required for the SoC. The other pins are optional or for special features.

Figure 4. Target Connector Pin-Out

Note the concept with the target voltage sense signal. This signal is used by the level converters on the CC Debugger to handle different voltage levels on the target board and the debugger. Pin 2 on the target connector must be connected to VDD on the target board.
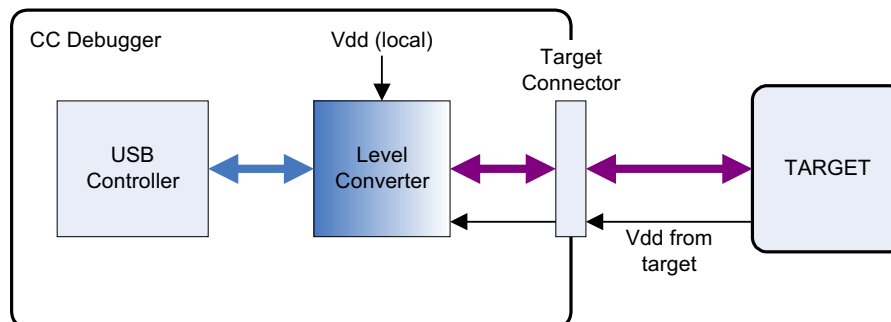
Figure 5. Voltage From Target to CC Debugger

Alternatively, it is possible to power the target by connecting pin 9 to VDD on the target. In that case, the CC Debugger supplies 3.3 V to the target.

6.2 Connecting the CC Debugger to a System-on-Chip

6.2.1 Minimum Connection for Debugging

For successful debugging of a TI 8051-based RF SoC, connect the two debug signals Debug Data (DD) and Debug Clock (DC) and the reset signal RESETn to the device. Note that DD is a bidirectional signal. In addition, the CC Debugger must be connected to GND and VDD on the board. VDD is used as an input to the level shifters on the CC Debugger, thus allowing a different operating voltage on the target than internally on the debugger.

For CC111x, CC251x, CC243x, CC253x and CC254x, except CC2544 and CC2545, connect the DD signal to pin P2.1 and DC to pin P2.2.

For CC2544, connect the DD signal to P1.3 and DC to P1.2.

For CC2545, connect the DD signal to P1.3 and DC to P1.4.

Note that it is possible to power the target board from the debugger by connecting the 3.3 V signal on pin 9 on the connector to the target board.
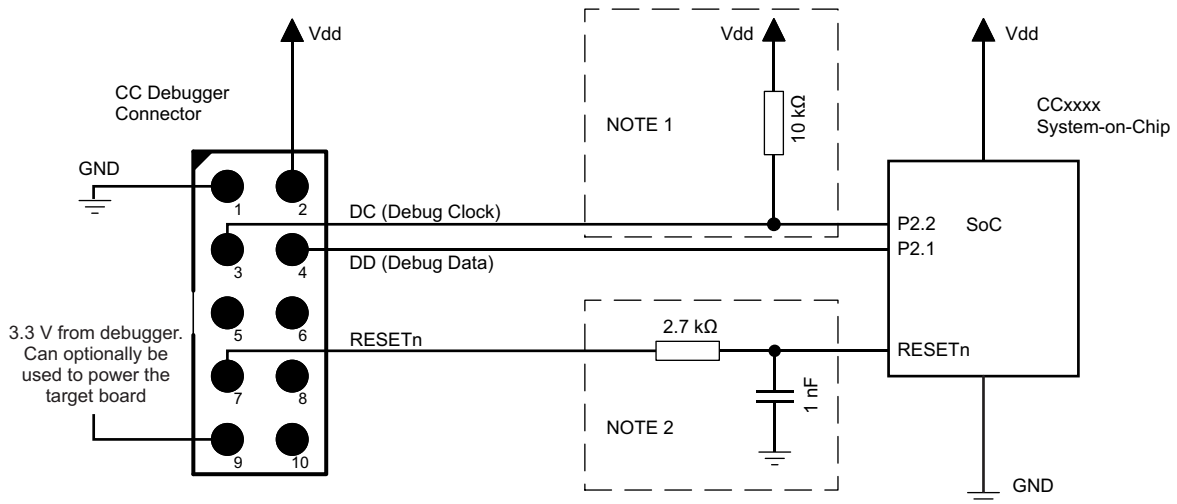
Figure 6. Minimum Connection for Debugging of 8051 SoC

> **NOTE:** Some early revisions of certain SoCs (CC2430, CC2510 and CC1110) needed an external pull-up to avoid unwanted transitions on the debug clock line during chip reset, inadvertently setting the device in debug mode. All new revisions of all SoCs now have an internal pull-up on P2.2, so this external component is not required.

> **NOTE:** The RESETn pin is sensitive to noise and can cause unintended reset of the chip. For reset lines susceptible to noise, it is recommended to add an external RC filter. For recommended RESET circuitry, see the device-specific SoC data sheet and reference designs. The CC Debugger supports slow transitions on the reset line, using a 2 ms delay between any transition on the RESET line and other transitions on the DC and DD lines.

6.2.2   Minimum Connection for SmartRF Studio

Use the same connection as for debugging the SoC.

6.2.3 Minimum Connection for SmartRF Packet Sniffer

In order to use the packet sniffer capabilities of the CC Debugger, it is also necessary to connect the SPI bus to the SoC. The SPI interface is used by the CC Debugger for reading the captured RF packets from the SoC (see Figure 7).
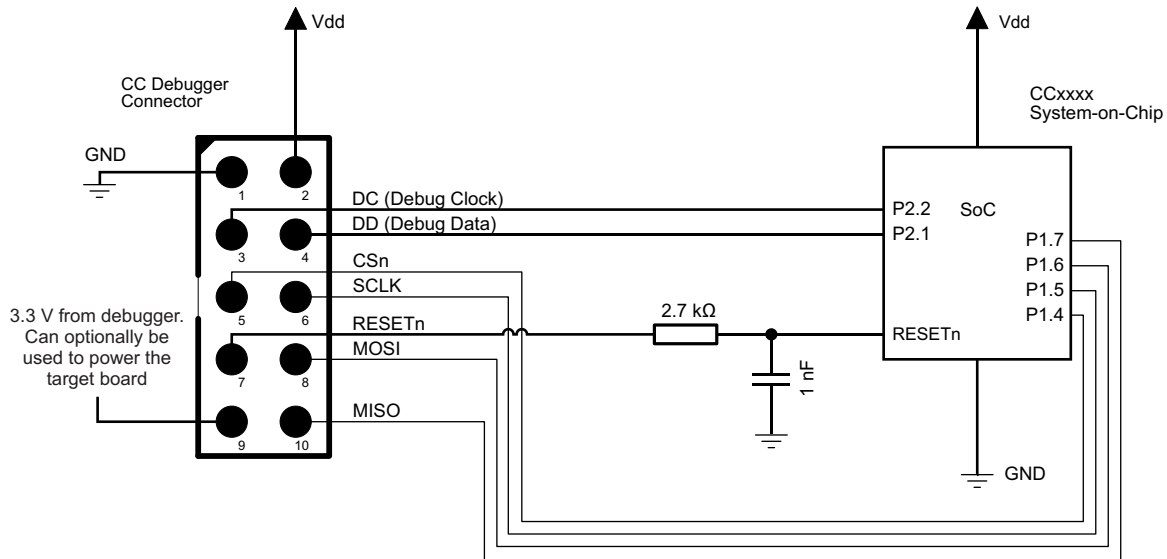

Figure 7. Connection to SoC to Enable Packet Sniffing

Note that the packet sniffer overwrites the Flash on the SoC with special packet capture firmware.

Note concerning the SPI interface to the SoC used for packet sniffing

All of the current TI RF SoCs can be configured to operate as SPI slaves, with the SPI signals (CS, SCLK, MISO and MOSI) going to one of the USART peripherals. The packet sniffer application programs the SoC with firmware that configures one of the USART peripherals in order to communicate with the CC Debugger. The firmware can use any of the four possible pin configurations (USART 0 or 1, pin out alternative 1 or 2). However, only a subset is currently supported (see Table 1).

Table 1. Supported SPI Connections (marked OK)

	USART0, alt 1	USART0, alt 2	USART1, alt 1	USART1, alt 2
CC243x	-	-	-	OK
CC253x/CC254x	-	-	-	OK
CC111x	OK	-	-	OK
CC251x	OK	-	-	OK

Table 2. USART Pin Out Details

	USART0, alt 1	USART1, alt 2
SCLK	P0.5	P1.5
CS	P0.4	P1.4
MOSI	P0.3	P1.6
MISO	P0.2	P1.7

In case of multiple supported interfaces, the Packet Sniffer application lets you choose which interface to use.

6.3 *Connecting the CC Debugger to a Transceiver*

The SPI interface on the CC Debugger can be used to interface many of the CCxxxx transceivers and control them from SmartRF Studio. The transceivers, transmitters, and receivers currently supported are:

- CC1100
- CC1101
- CC1120
- CC1121
- CC1125
- CC1175
- CC110L
- CC113L
- CC115L
- CC1200
- CC1201
- CC2500
- CC2520

Note that the CC Debugger operates as the SPI Master. In a multi-master system, it is necessary to make sure the debugger output signals (DC, DD, CSn, SCLK, MOSI and RESETn) do not interfere with the other SPI master on the board. The other SPI master would typically be the microcontroller on the board.

Figure 8 through Figure 10 show the interconnection between the debugger and the various supported transceivers.
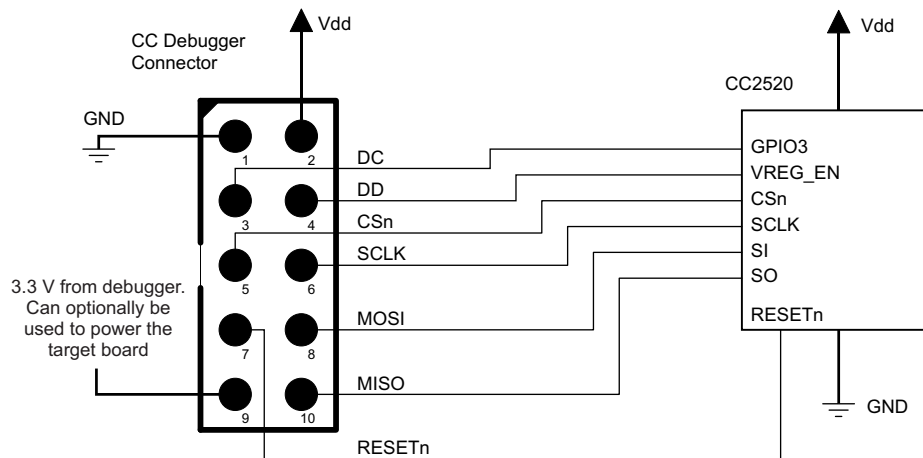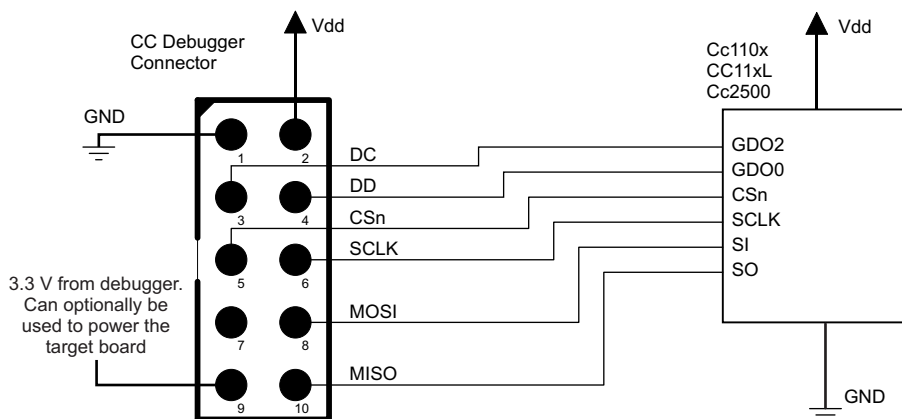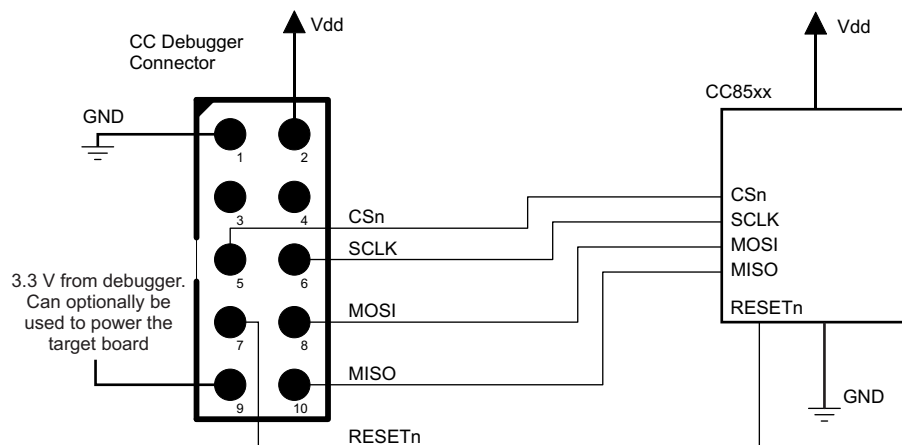

Figure 8. CC Debugger Connected to CC2520
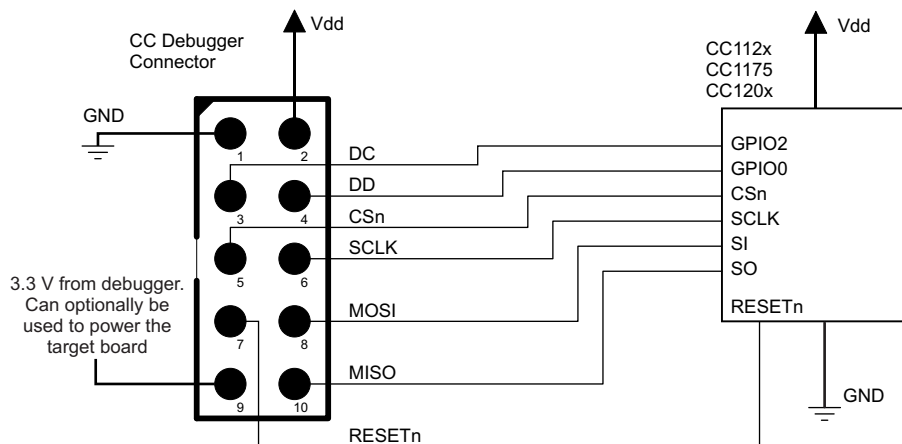
Figure 9. CC Debugger Connected to CC112x/CC1175/CC120x

Figure 10. CC Debugger Connected to CC110x/CC11xL/CC2500

6.4 *Connecting the CC Debugger to a CC85xx*

In order to configure the CC85xx devices (program the Flash on the device) with PurePath Wireless Configurator, the device's SPI interface must be connected to the CC Debugger as shown in Figure 11.

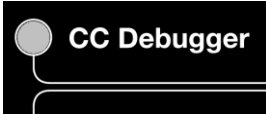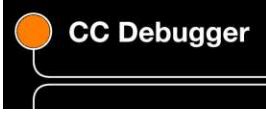**Figure 11. CC Debugger Connected to CC85XX**
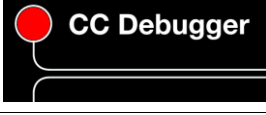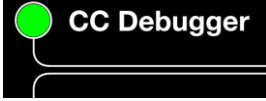
7 Using the CC Debugger

After having connected the debugger to the target device, the debugger can be powered up by plugging in the USB cable.

The debugger immediately starts a device detection process, looking for all known devices. If no devices are detected, the LED will be RED. If a device is detected, the LED will be GREEN.

If the LED is GREEN, it is possible to start using the debugger together with one of the supported PC tools.

7.1 Understanding the LED

Table 3. Understanding the LED

OFF	The debugger has no power or there is no valid firmware on the debugger. Make sure the debugger is properly powered via the USB cable or try to resurrect the debugger using the method described in Section 8.4.
AMBER (BOTH LEDS ON)	The debugger is powered, but there is no valid firmware. Try to resurrect the debugger using the method described in Section 8.4.
RED LED BLINKING	The Debugger is in Boot Recovery Mode. The debugger briefly enters this state while the firmware is being upgraded (see Section 8). The board might also enter this state if the firmware is corrupt or if the user has manually forced the board to start up in the special "boot recovery mode" (see Section 8.3). To go out of the state, reset the debugger by pressing the *Reset* button or by power-cycling the device. If the LED is still blinking, reprogram the unit by using the Flash Programmer Application.
RED LED ON	No device detected. This might be due to old firmware on the CC Debugger. New devices might not be supported with the current firmware on the debugger. For the firmware upgrade procedure, see Section 8. There might also be a problem with the hardware connection. Check the connection to device and make sure the target board is properly powered and that VDD on the target board is connected to pin 2 on the debug connector. Press and release the reset button to retry the target device detection.
GREEN LED ON	The target device has been properly detected. It is possible to start using the supported tools (see Section 5.2).

8   Updating the Firmware

In order to make sure the CC Debugger works seamlessly with your device, it is important that it has the latest and greatest firmware. This section describes how you can upgrade the firmware automatically from SmartRF Studio or manually from the SmartRF Flash Programmer. The section also describes how to resurrect a seemingly broken debugger.

8.1   *Updating the Firmware Automatically in SmartRF Studio*

Updating the firmware on the CC Debugger can be done automatically by SmartRF Studio. Follow the steps described below:

1. Start SmartRF Studio.
2. Disconnect the debugger from any target board and connect it to the PC via the USB cable. The debugger will appear in the list of connected devices in the lower part of the SmartRF Studio startup panel (see Figure 12).
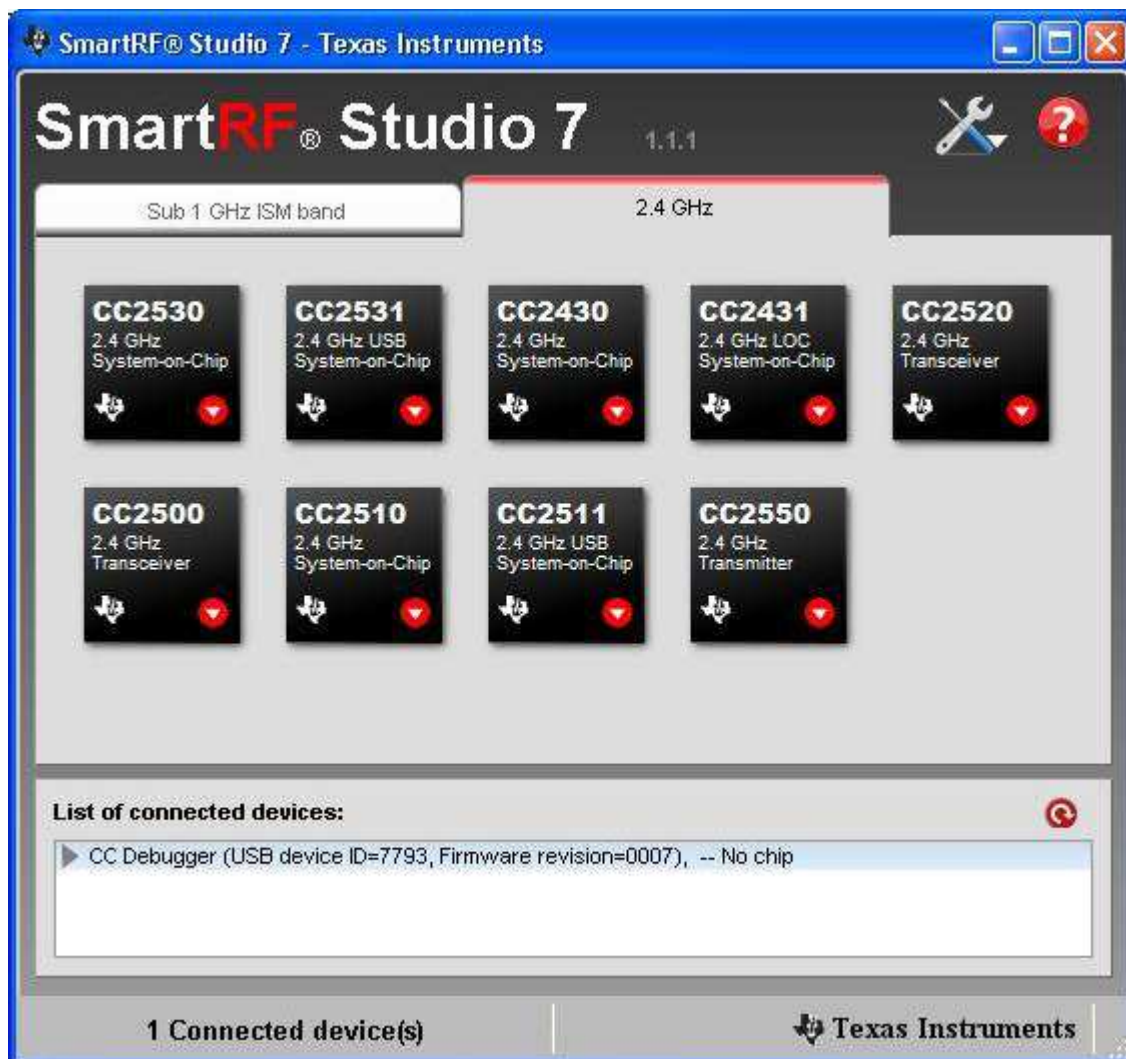
Figure 12. Auto Firmware Upgrade

3.   Double click on the item in the list and a new window will appear (see Figure 13).

Figure 13. Auto Firmware Update

4.   Click *Yes* (see Figure 14) and let SmartRF Studio do the rest.

Figure 14. Firmware Successfully Updated

5.   Click *Done* (see Figure 14). The device should appear in the list of connected devices, now showing the new firmware revision.

8.2 Updating the Firmware Manually in SmartRF Flash Programmer

You can also update the firmware manually using the SmartRF Flash Programmer (illustrated in Figure 15). You can use this method if you like to have full control of the firmware image to be programmed on the controller of the debugger (for example, programming custom firmware or old firmware revisions).

1. Start the SmartRF Flash Programmer and select the *EB application (USB)* tab. This tab lets you program compatible firmware on the CC Debugger (or evaluation boards) via the USB interface (no external programming device required).

2. Disconnect the debugger from any target board and connect it to the PC via the USB cable. The debugger appears in the list of connected devices. The chip type is listed as N/A.

3. Select the Flash image you want to program on the debugger. Normally, you would select: C:\Program Files (x86)\Texas Instruments\SmartRF Tools\Firmware\CC Debugger\cebal_fw_srf05dbg.hex. [1]

4. Select *Erase, program and verify* in the Actions box.

5. Click the *Perform actions* button. The programming procedure will start. Note that this takes several seconds.

6. The CC Debugger reappears in the list of connected devices, now showing the new firmware revision in the device list.
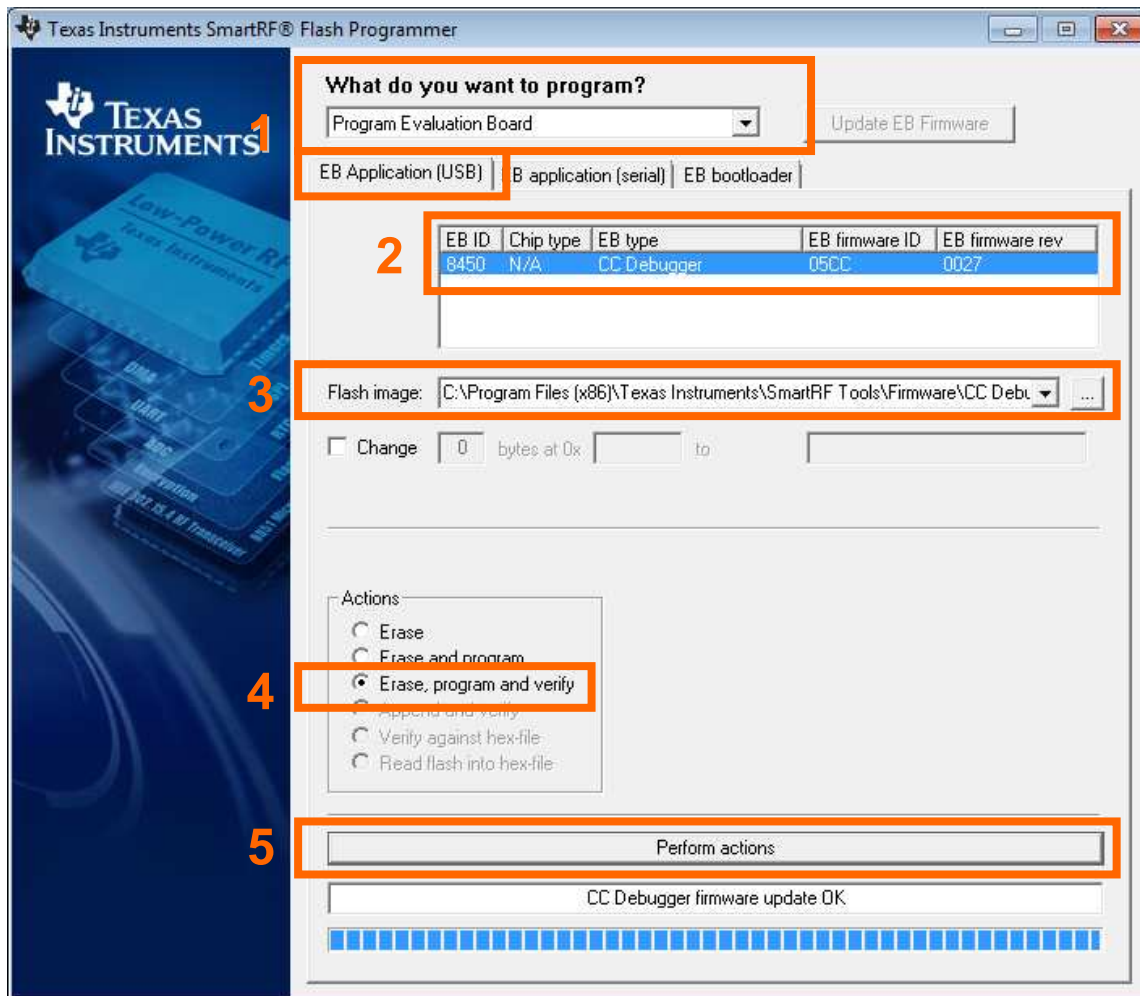
7. Done!

Figure 15. SmartRF Flash Programmer - Manually Updating the Bootloader

[1] Assuming default installation path of SmartRF Flash Programmer.

8.3 Forced Boot Recovery Mode

If, for some reason, the firmware update fails and the CC Debugger appears to be non-responsive, there is a way to force the board to only run the bootloader and stop all further execution. In this mode, no attempts are made to start the firmware, and the board will only allow the user to perform a new firmware upgrade over USB.

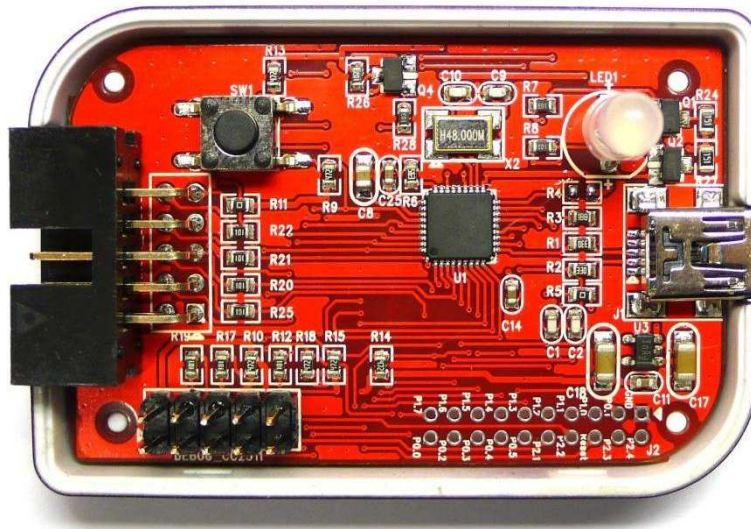Disconnect the debugger from any power source and open the plastic enclosure.

Figure 16. Internal View of CC Debugger

Short circuit the pins as depicted in Figure 17: P1.6 on the CC2511 must be connected to GND during the power-on reset to enter boot recovery mode.
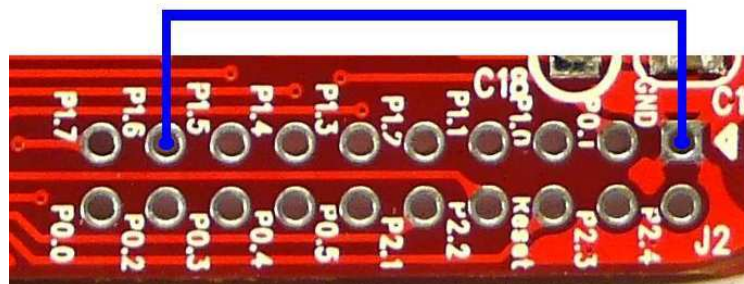
Figure 17. Short-Circuit Pins for Boot Recovery Mode

When reconnecting the USB cable, the LED starts to blink with a RED light. This indicates that the bootloader is running and that the debugger is in boot recovery mode.

At this point, follow the same firmware programming steps as described at the beginning of this section.

Also note that the boot recovery mode can be used as a check to verify that the bootloader on the debugger is working.

8.4 Resurrecting the CC Debugger

If the CC Debugger appears to be completely dead when applying power, there is a way to "unbrick" the board. The method consists of reprogramming the bootloader on the debugger using the debug connector inside the box. This requires an extra programming device.

When opening the box, locate the debug connector header next to the target connector. Connect this header to another CC Debugger (see Figure 18) or to a SmartRF05EB (see Figure 19). When using SmartRF05EB, connect a 10-pin flat cable from the "Ext SoC Debug" plug (P3) on the EB to the "USB Debug" plug (P2) on the CC Debugger. The dead debugger needs power, so connect the USB cable. Turn on the SmartRF05EB or debugger - it should detect the USB Controller (CC2511) on the debugger.

Figure 18. Programming the Bootloader on the CC Debugger Using Another CC Debugger
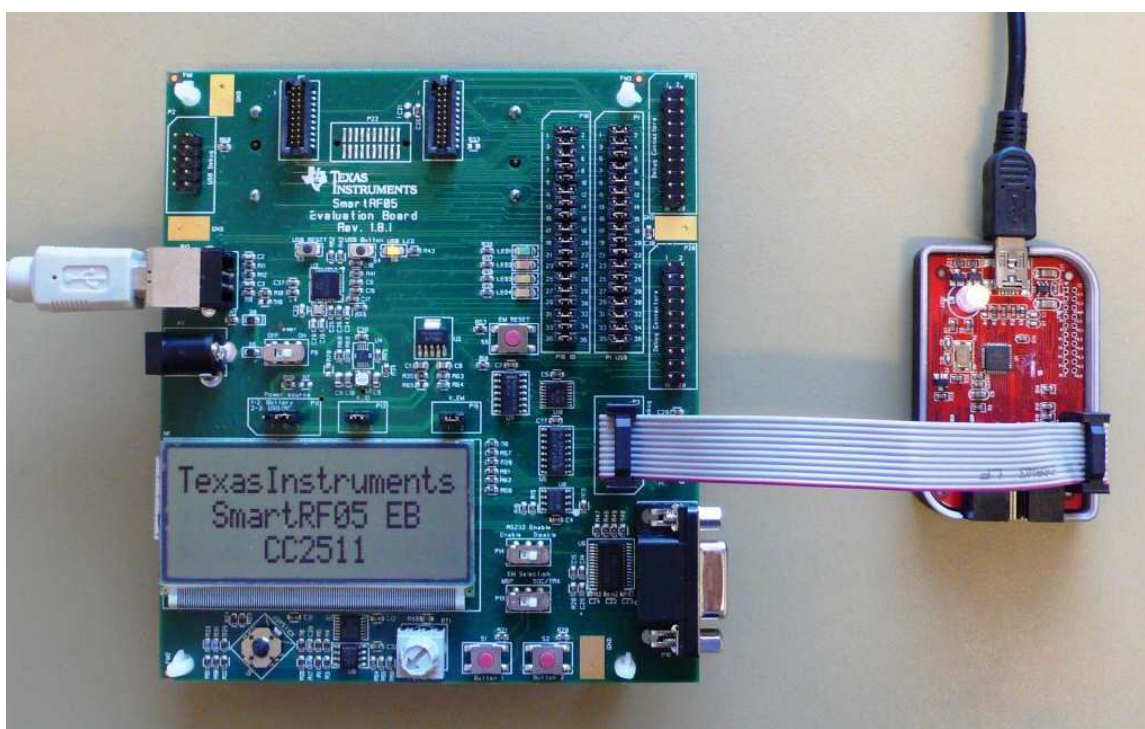
Figure 19. Programming the Bootloader on the CC Debugger Using SmartRF05EB

Next, use the SmartRF Flash Programmer to program the bootloader on the debugger. Follow these five steps (illustrated in Figure 20):

1. Start the application and select *Program Evaluation Board* in the *What do you want to program?* drop down box, then select the *EB Bootloader* tab.

2. In the Device box, select *SmartRF05EB* regardless of the device being used to program the debugger. For example, select *SmartRF05EB* both when you are using a CC Debugger and when you are using a SmartRF05EB for the resurrection.

3. Select which Flash image to program. The bootloader image is included when installing the Flash programmer and it is usually located at "C:\Program Files (x86)\Texas Instruments\SmartRF Tools\Firmware\CC Debugger".

4. In the Board identification box, give the debugger a unique ID number; any 4 digit number will work. This number is used by the driver on the PC to uniquely identify devices if more than one debugger is connected at the same time.

5. Select *Erase, program and verify* in the Actions box

6. Click the *Perform Actions* button. The firmware upgrade takes a few seconds.
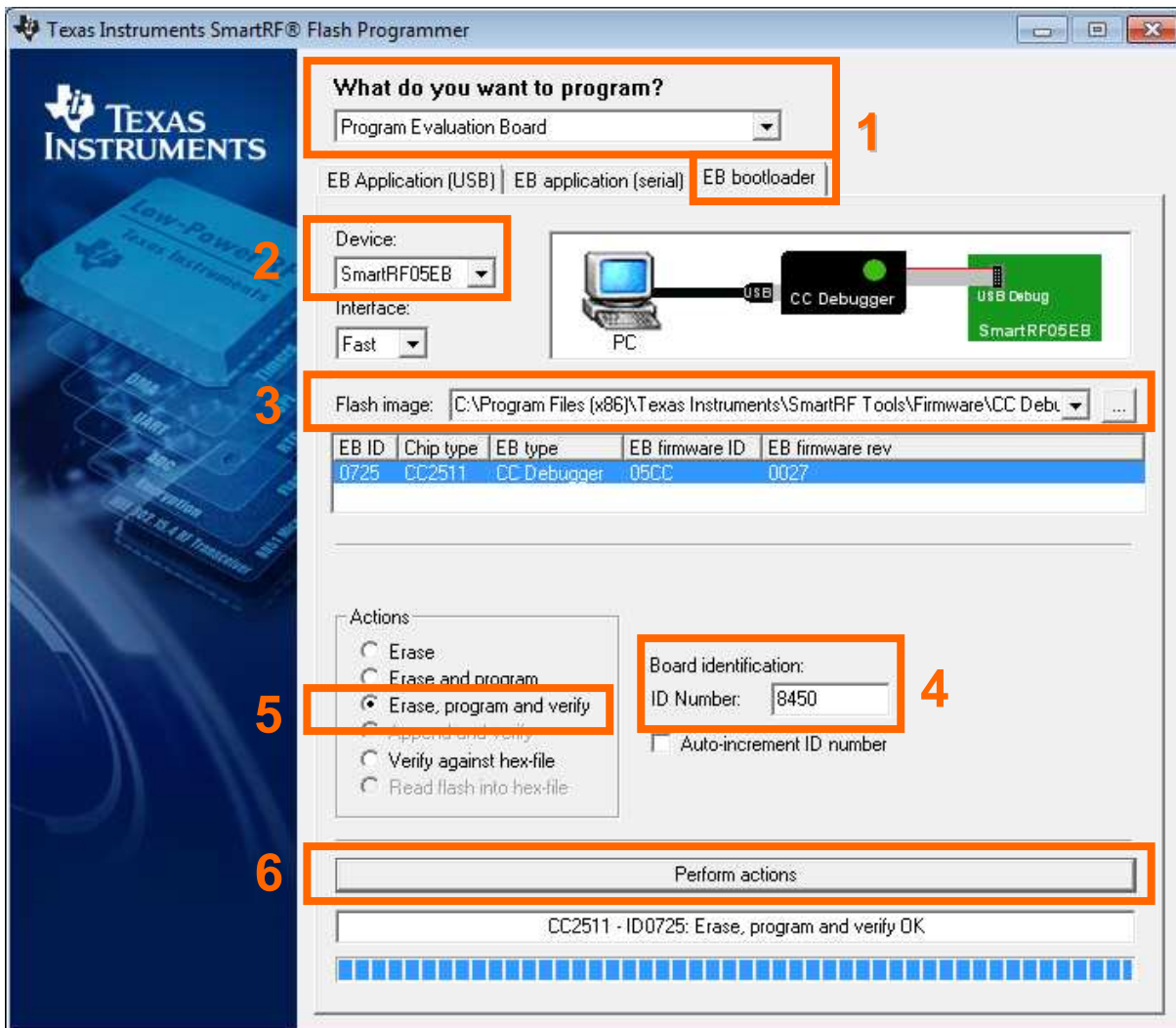
Figure 20. SmartRF Flash Programmer - Updating the Bootloader

Once the bootloader is programmed, you might be asked to install a USB driver on the PC. Follow the same procedure as when the debugger was connected to the PC the first time (see Section 5).

The RED LED on the debugger should now be blinking, indicating that the bootloader is running but that no application has been loaded. If the RED LED is off, there is probably something wrong with the hardware. The debugger firmware can now be programmed directly over USB by following the procedure in either Section 8.1 or Section 8.2.

9   Troubleshooting

Table 4. Troubleshooting FAQs

|	
Q1	Help! The debugger does not detect the SoC. What should I do?
A1	There are several things to check. Upgrade the firmware. Many CC Debuggers have old firmware that will not automatically detect newer devices, like CC2543/44/45. For further instructions, see Section 8. Check that the cable is oriented correctly and that the pins are connected to the right signals on the debugger. Check that the debugger gets power from the target (proper connection of the Target Voltage Sense signal). This is required in order for the level converters on the debugger to work. Check that ground on the target is connected to ground on the debugger. This is normally achieved through the target connector. Note that since the ground planes are the same, be aware of any adverse effects caused by different ground planes on the target and on the PC (grounded via USB cable). Check that the cable and the cable adapter board are not broken. Especially the small flat cable is prone to stop working if handled a lot or being bent and stretched beyond normal operating conditions.
Q2	Does IAR EW8051 support the CC Debugger as debugging device?
A2	Yes, but make sure you have an up-to-date version of IAR with the new debug driver plug-in from Texas Instruments. You will need version 7.51A or higher.
Q3	Can the debugger be used as an interface to the RF device for packet sniffing?
A3	Yes, this is supported for selected devices. Use the same interconnection as in the diagrams in Section 6.
Q4	Is there a way to remove the plastic casing without damaging it?
A4	Yes, there is. Hold the bottom piece of the plastic in one hand. With your other hand, take a firm grip on the long lateral sides of the upper part of the plastic and squeeze while moving the upper part away from the bottom. The two parts should separate from each other. To reassemble the plastic, just click the two pieces together.
Q5	Is this a Mini or a Micro USB plug?
A5	Mini USB type A.
Q6	I have two CC Debuggers with the same EB ID, and I'm unable to use them together. What do I do?
A6	Two EBs with the same EB ID cause a driver conflict. The solution is to resurrect one of the CC Debuggers and give it a new EB ID. 1. Connect one CC Debugger to your computer. 2.Connect the CC Debugger you want to resurrect to a separate power source (for example, another computer or a USB charger). 3. Follow the steps for resurrecting the CC Debugger described in Section 8.4.

10 Schematics

For more information, see Appendix A or the complete bundle including gerber files, schematics and layout [3].

11 References

1. CC-Debugger product web site: http://www.ti.com/tool/cc-debugger
2. *CC Debugger Quick Start Guide* (SWRU196)
3. CC-Debugger Layout and Schematics www.ti.com/lit/zip/swrr105
4. Cebal – CCxxxx Development Tools USB Driver for Windows x86 and x64 www.ti.com/lit/zip/swrc212
5. *Cebal – CCxxxx Development Tools USB Driver Installation Guide* (SWRA366)
6. Texas Instruments Support: support.ti.com
7. Texas Instruments Low Power RF Online Community: www.ti.com/lprf-forum
8. SmartRF Studio: www.ti.com/tool/smartrftm-studio
9. SmartRF Flash Programmer: www.ti.com/tool/flash-programmer
10. SmartRF Packet Sniffer: www.ti.com/tool/packet-sniffer
11. *SmartRF Flash Programmer User Manual* (SWRU069)
12. PurePath Wireless Configurator: www.ti.com/tool/purepath-wl-cfg
13. PurePath Wireless Commander: www.ti.com/tool/purepath-wl-cmd
14. SoC Battery Board product web site: www.ti.com/tool/soc-bb
15. IAR Embedded Workbench for 8051: www.iar.com/ew8051

Appendix A Schematic
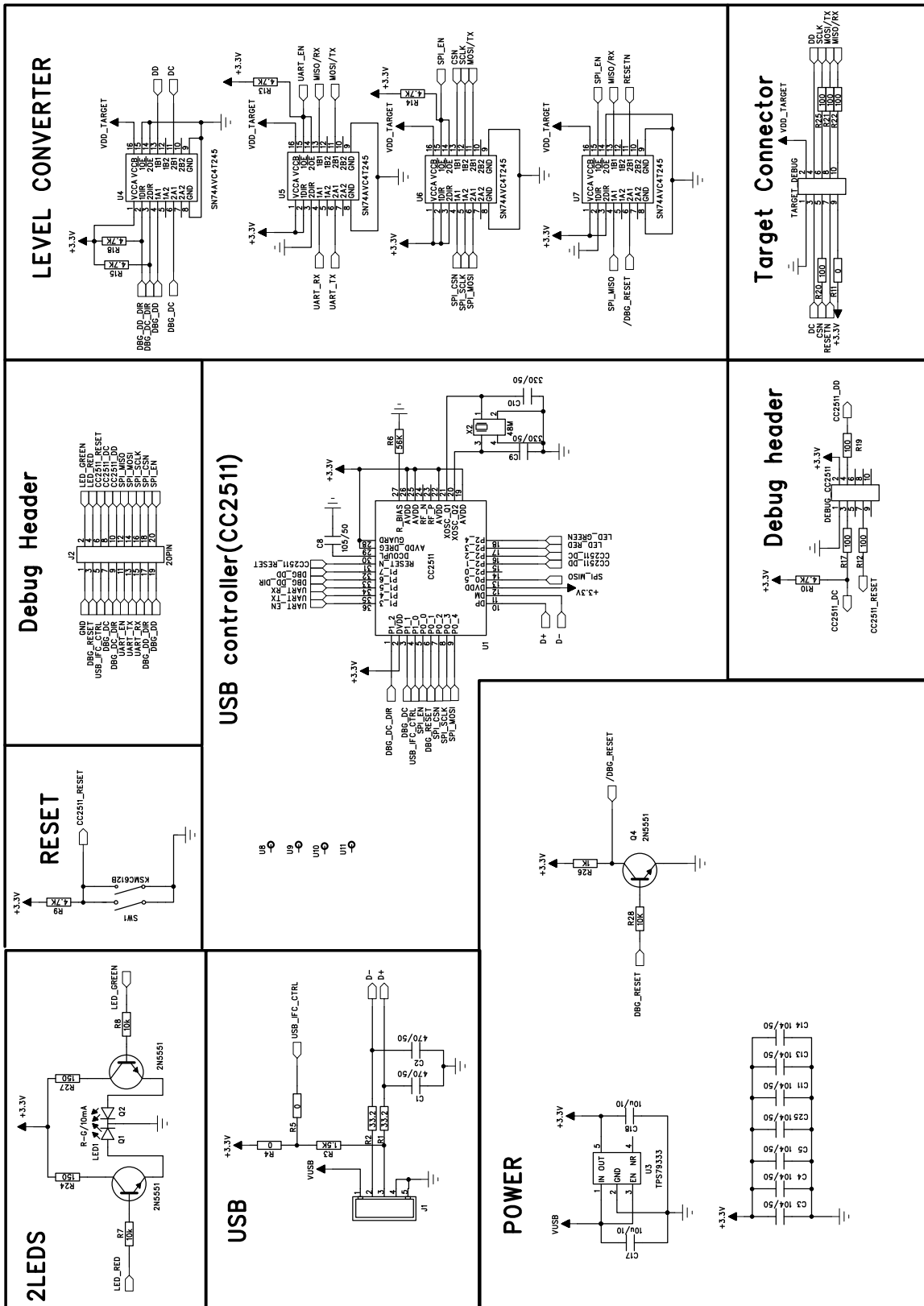
Revision History

Changes from G Revision (January 2014) to H Revision	Page

- Added Windows 8 as a supported operating system in Section 4. 5

IMPORTANT NOTICE

Products		Applications	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		