

## C8051F35x-DK DEVELOPMENT KIT USER'S GUIDE

### 1. Kit Contents

The C8051F35x-DK Development Kit contains the following items:

- C8051F350 Target Board
- C8051Fxxx Development Kit Quick-Start Guide
- AC to DC Power Adapter
- USB Debug Adapter (USB to Debug Interface)
- USB Cable
- CD-ROM

### 2. Hardware Setup using a USB Debug Adapter

The target board is connected to a PC running the Simplicity Studio via the USB Debug Adapter as shown in Figure 1.

1. Connect the USB Debug Adapter to the **DEBUG** connector on the target board with the 10-pin ribbon cable.
2. Connect one end of the USB cable to the USB connector on the USB Debug Adapter.
3. Connect the other end of the USB cable to a USB Port on the PC.
4. Connect the ac/dc power adapter to power jack **P1** on the target board.

#### Notes:

- Use the **Reset** button in the IDE to reset the target when connected using a USB Debug Adapter.
- Remove power from the target board and the USB Debug Adapter before connecting or disconnecting the ribbon cable from the target board. Connecting or disconnecting the cable when the devices have power can damage the device and/or the USB Debug Adapter.

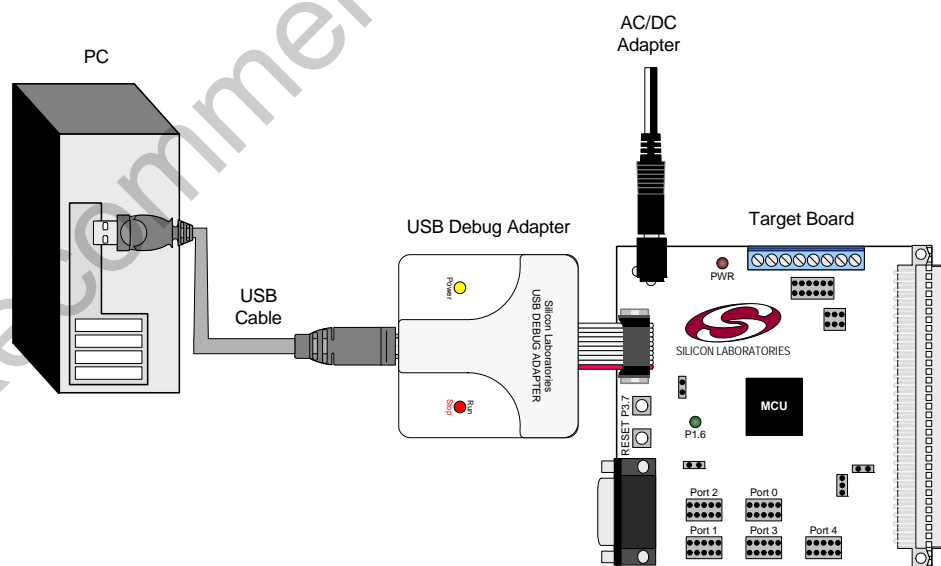
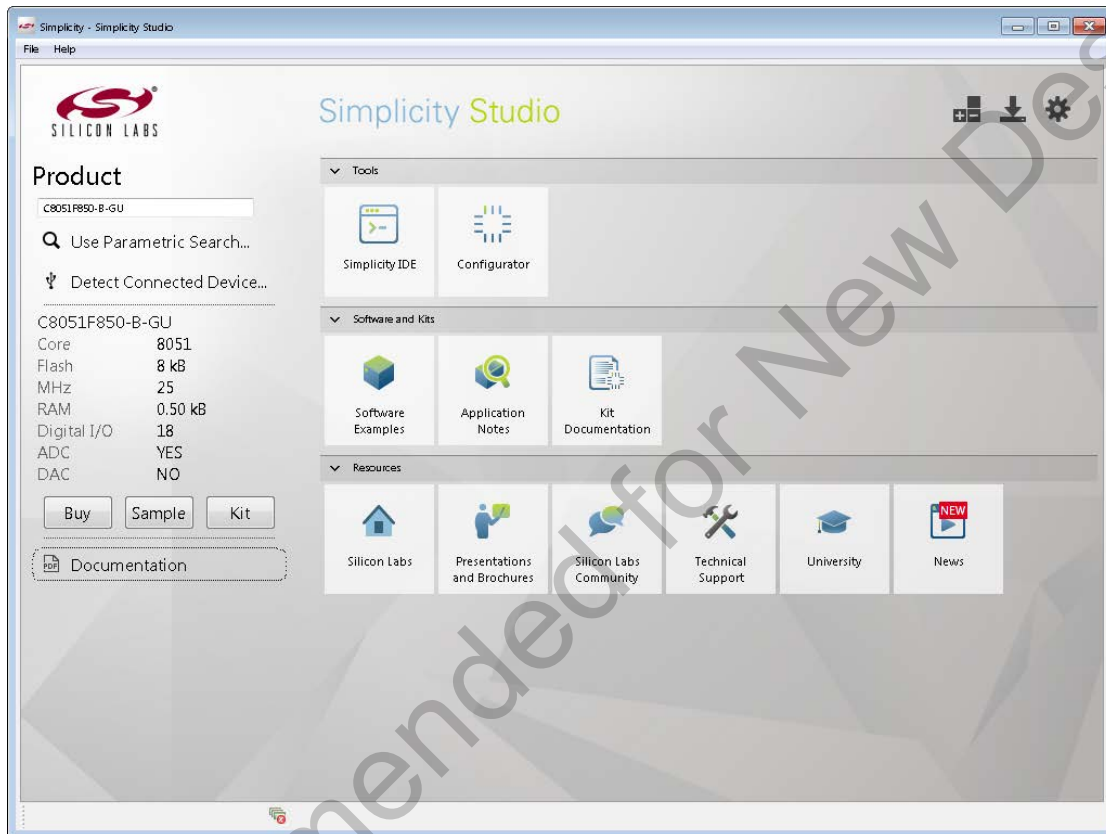


Figure 1. Hardware Setup using a USB Debug Adapter

## 3. Software Setup

Simplicity Studio greatly reduces development time and complexity with Silicon Labs EFM32 and 8051 MCU products by providing a high-powered IDE, tools for hardware configuration, and links to helpful resources, all in one place.

Once Simplicity Studio is installed, the application itself can be used to install additional software and documentation components to aid in the development and evaluation process.



**Figure 2. Simplicity Studio**

The following Simplicity Studio components are required for the C8051F350 Development Kit:

- 8051 Products Part Support
- Simplicity Developer Platform

Download and install Simplicity Studio from [www.silabs.com/8bit-software](http://www.silabs.com/8bit-software) or [www.silabs.com/simplicity-studio](http://www.silabs.com/simplicity-studio). Once installed, run Simplicity Studio by selecting **Start**→**Silicon Labs**→**Simplicity Studio**→**Simplicity Studio** from the start menu or clicking the **Simplicity Studio** shortcut on the desktop. Follow the instructions to install the software and click **Simplicity IDE** to launch the IDE.

The first time the project creation wizard runs, the **Setup Environment** wizard will guide the user through the process of configuring the build tools and SDK selection.

In the **Part Selection** step of the wizard, select from the list of installed parts only the parts to use during development. Choosing parts and families in this step affects the displayed or filtered parts in the later device selection menus. Choose the C8051F35x family by checking the **C8051F35x** check box. Modify the part selection at any time by accessing the **Part Management** dialog from the **Window**→**Preferences**→**Simplicity Studio**→**Part Management** menu item.

Simplicity Studio can detect if certain toolchains are not activated. If the **Licensing Helper** is displayed after completing the **Setup Environment** wizard, follow the instructions to activate the toolchain.

## 3.1. Running Blinky

Each project has its own source files, target configuration, SDK configuration, and build configurations such as the **Debug** and **Release** build configurations. The IDE can be used to manage multiple projects in a collection called a workspace. Workspace settings are applied globally to all projects within the workspace. This can include settings such as key bindings, window preferences, and code style and formatting options. Project actions, such as build and debug are context sensitive. For example, the user must select a project in the **Project Explorer** view in order to build that project.

To create a project based on the Blinky example:

1. Click the **Software Examples** tile from the Simplicity Studio home screen.
2. In the **Kit** drop-down, select **C8051F350 Development Kit**, in the **Part** drop-down, select **C8051F350**, and in the **SDK** drop-down, select the desired SDK. Click **Next**.
3. Under **C8051F350 Development Kit**, select **F35x Blinky**, click **Next**, and click **Finish**.
4. Click on the project in the **Project Explorer** and click **Build**, the hammer icon in the top bar. Alternatively, go to **Project**→**Build Project**.
5. Click **Debug** to download the project to the hardware and start a debug session.
6. Press the **Resume** button to start the code running. The LED should blink.



7. Press the **Suspend** button to stop the code.



8. Press the **Reset the device** button to reset the target MCU.



9. Press the **Disconnect** button to return to the development perspective.



## 3.2. Simplicity Studio Help

Simplicity Studio includes detailed help information and device documentation within the tool. The help contains descriptions for each dialog window. To view the documentation for a dialog, click the question mark icon in the window:



This will open a pane specific to the dialog with additional details.

The documentation within the tool can also be viewed by going to **Help**→**Help Contents** or **Help**→**Search**.

## 4. Target Board

The C8051F35x Development Kit includes a target board with a C8051F350 device pre-installed for evaluation and preliminary software development. Numerous input/output (I/O) connections are provided to facilitate prototyping using the target board. Refer to Figure 3 for the locations of the various I/O connectors.

- P1 Power connector (accepts input from 7 to 15 VDC unregulated power adapter)
- J1 22-pin Expansion I/O connector
- J3 Port I/O Configuration Jumper Block
- J4 DEBUG connector for Debug Adapter interface
- J5 DB-9 connector for UART0 RS232 interface
- J6 Analog I/O terminal block
- J7 Connector for IDAC0 voltage circuit
- J8 USB Debug Adapter target board power connector
- J9, J10 External crystal enable connectors
- J11 Connector for IDAC1 voltage circuit
- J12 Connector block for Thermistor circuitry
- J13, J14 ADC external voltage reference connectors

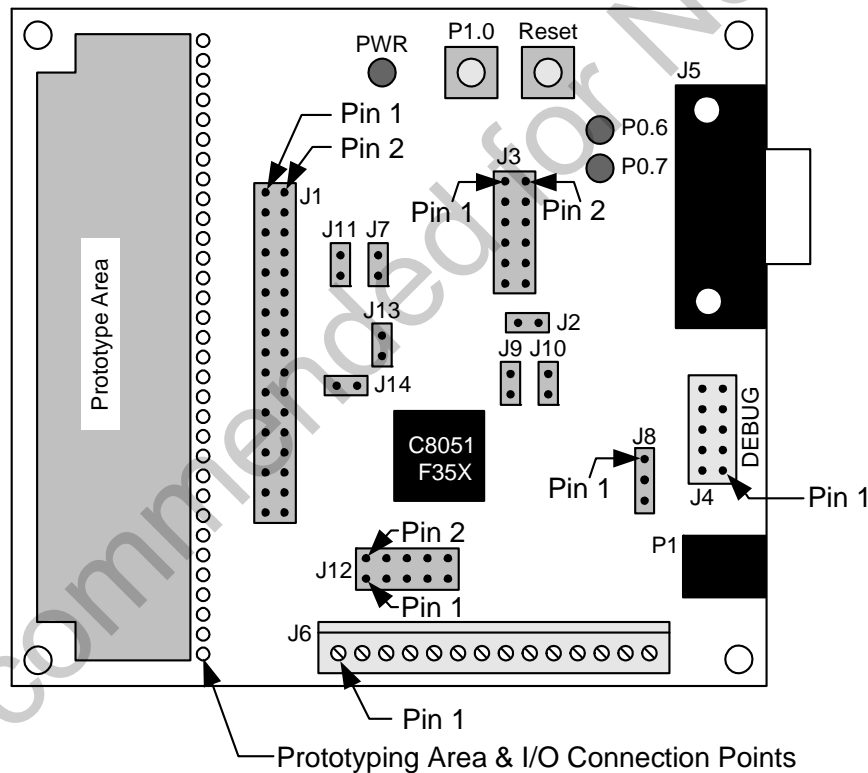


Figure 3. C8051F350 Target Board

## 4.1. System Clock Sources

The C8051F350 device installed on the target board features a calibrated programmable internal oscillator which is enabled as the system clock source on reset. After reset, the internal oscillator operates at a frequency of 3.0625 MHz ( $\pm 2\%$ ) by default but may be configured by software to operate at other frequencies. Therefore, in many applications an external oscillator is not required. However, if you wish to operate the C8051F350 device at a frequency not available with the internal oscillator, an external crystal may be used. Refer to the C8051F35x data sheet for more information on configuring the system clock source.

The target board is designed to facilitate the installation of an external crystal. Remove shorting blocks at headers J9 and J10 and install the crystal at the pads marked Y1. Install a 10 M $\Omega$  resistor at R9 and install capacitors at C14 and C15 using values appropriate for the crystal you select. Refer to the C8051F35x data sheet for more information on the use of external oscillators.

## 4.2. Switches and LEDs

Two switches are provided on the target board. Switch SW1 is connected to the RESET pin of the C8051F350. Pressing SW1 puts the device into its hardware-reset state. Switch SW2 is connected to the C8051F350's general purpose I/O (GPIO) pin through headers. Pressing SW2 generates a logic low signal on the port pin. Remove the shorting block from the jumper to disconnect SW2 from the port pins. The port pin signal is also routed to a pin on the J1 I/O connector. See Table 1 for the port pins and headers corresponding to each switch.

Three LEDs are also provided on the target board. The red LED labeled PWR is used to indicate a power connection to the target board. The green LEDs labeled with port pin names are connected to the C8051F350's GPIO pins through headers. Remove the shorting blocks from the headers to disconnect the LEDs from the port pins. The port pin signals are also routed to pins on the J1 I/O connector. See Table 1 for the port pins and headers corresponding to each LED.

**Table 1. Target Board I/O Descriptions**

Description	I/O	Jumper
SW1	Reset	none
SW2	P1.0	J3[5–6]
Green LED D2	P0.6	J3[1–2]
Green LED D1	P0.7	J3[3–4]
Red LED	PWR	none

## 4.3. Expansion I/O Connector (J1)

The 34-pin Expansion I/O connector J1 provides access to all signal pins of the C8051F350 device. Pins for  $V_{DD}$  and GND as well as pins for  $V_{DDA}$  and AGND are also available. A small through-hole prototyping area is also provided. All I/O signals routed to connector J1 are also routed to through-hole connection points between J1 and the prototyping area (see Figure 3 on page 4). Each connection point is labeled indicating the signal available at the connection point. See Table 2 for a list of pin descriptions for J1.

**Table 2. J1 Pin Descriptions**

Pin #	Description	Pin #	Description	Pin #	Description
1	$V_{DD}$	13	P1.2	25	AIN3
2	GND	14	P1.3	26	AIN4
3	P0.0	15	P1.4	27	AIN5
4	P0.1	16	P1.5	28	AIN6
5	P0.2	17	P1.6/IDAC0	29	AIN7
6	P0.3	18	P1.7/IDAC1	30	VREF+
7	P0.4	19	P2.0	31	/RST
8	P0.5	20	AGND	32	VREF-
9	P0.6	21	AGND	33	VDDA
10	P0.7	22	AIN0	34	AGND
11	P1.0	23	AIN1	-	-
	P1.1	24	AIN2	-	-

## 4.4. Target Board DEBUG Interface (J4)

The DEBUG connector (J4) provides access to the DEBUG (C2) pins of the C8051F350. It is used to connect the Serial Adapter or the USB Debug Adapter to the target board for in-circuit debugging and Flash programming. Table 3 shows the DEBUG pin definitions.

**Table 3. DEBUG Connector Pin Descriptions**

Pin #	Description
1	+3 VD (+3.3 VDC)
2, 3, 9	GND (Ground)
4	C2D
5	/RST (Reset)
6	P3.0
7	C2CK
8	Not Connected
10	USB Power

## 4.5. Serial Interface (J5)

A RS232 transceiver circuit and DB-9 (J5) connector are provided on the target board to facilitate serial connections to UART0 of the C8051F350. The TX, RX, RTS and CTS signals of UART0 may be connected to the DB-9 connector and transceiver by installing shorting blocks on header J3.

- J3[7–8] - Install shorting block to connect UART0 TX (P0.4) to transceiver.
- J3[9–10] - Install shorting block to connect UART0 RX (P0.5) to transceiver.
- J3[11–12] - Install shorting block to connect UART0 RTS (P1.4) to transceiver.
- J3[13–14] - Install shorting block to connect UART0 CTS (P1.5) to transceiver.

#### 4.6. Analog I/O (J6)

Many of the C8051F350 target device's port pins are connected to the J6 terminal block. Connections for VDDA, AGND, ADC external voltage references, IDAC outputs and ADC inputs are available. Refer to Table 4 for the J6 terminal block connections.

**Table 4. J6 Terminal Block Pin Descriptions**

Pin #	Description	Pin #	Description
1	IDAC0	8	AIN3
2	IDAC1	9	AIN4
3	VREF+	10	AIN5
4	VREF-	11	AIN6
5	AIN0	12	AIN7
6	AIN1	13	AGND
7	AIN2	14	VDDA

#### 4.7. IDAC Connectors (J7, J11)

The C8051F350 target board also features two Current-to-Voltage 1 K $\Omega$  load resistors that may be connected to the 10-bit current-mode Digital-to-Analog Converters (IDACs) on port pins P1.6 and P1.7. Install a shorting block on J7 to connect the IDAC0/P1.6 pin of the target device to a load resistor. Install a second shorting block on J11 to connect the IDAC1/P1.7 pin of the target device to a load resistor. The IDAC signals are then routed to the J1 and J6 connectors.

#### 4.8. USB Debug Adapter Target Board Power Connector (J8)

The USB Debug Adapter includes a connection to provide power to the target board. This connection is routed from J4[10] to J8[1]. Place a shorting block at header J8[2-3] to power the board directly from an ac/dc power adapter. Place a shorting block at header J8[1-2] to power the board from the USB Debug Adapter. Please note that the second option is not supported with either the EC1 or EC2 Serial Adapters.

#### 4.9. Thermistor Connector Block (J12)

A thermistor circuit is included on the board which allows for a ratiometric measurement of the included thermistor. To use the circuit, remove the shorting blocks on the IDAC headers (J7, J11) and the External Voltage Reference headers (J13, J14). Next, connect all of the signals on the thermistor connector block (J12). Software on the chip can be written to use IDAC0 as a reference current source for the circuit. The current creates a voltage across R27 which is used as an external Voltage Reference input, while the voltage across the thermistor (R26) can be measured by connecting AIN0.0 to the AIN+ ADC input, and AIN0.1 to the AIN- ADC input.

#### 4.10. External Voltage Reference (J13, J14)

A precision 2.5V voltage reference source is also included on the C8051F350 target board. To use this circuit as a reference source for the ADC, place shorting block on headers J13 and J14. Note that software on the device should be configured to use an external reference when using this configuration. When using the internal (on-chip) reference source, headers J13 and J14 should be left open.





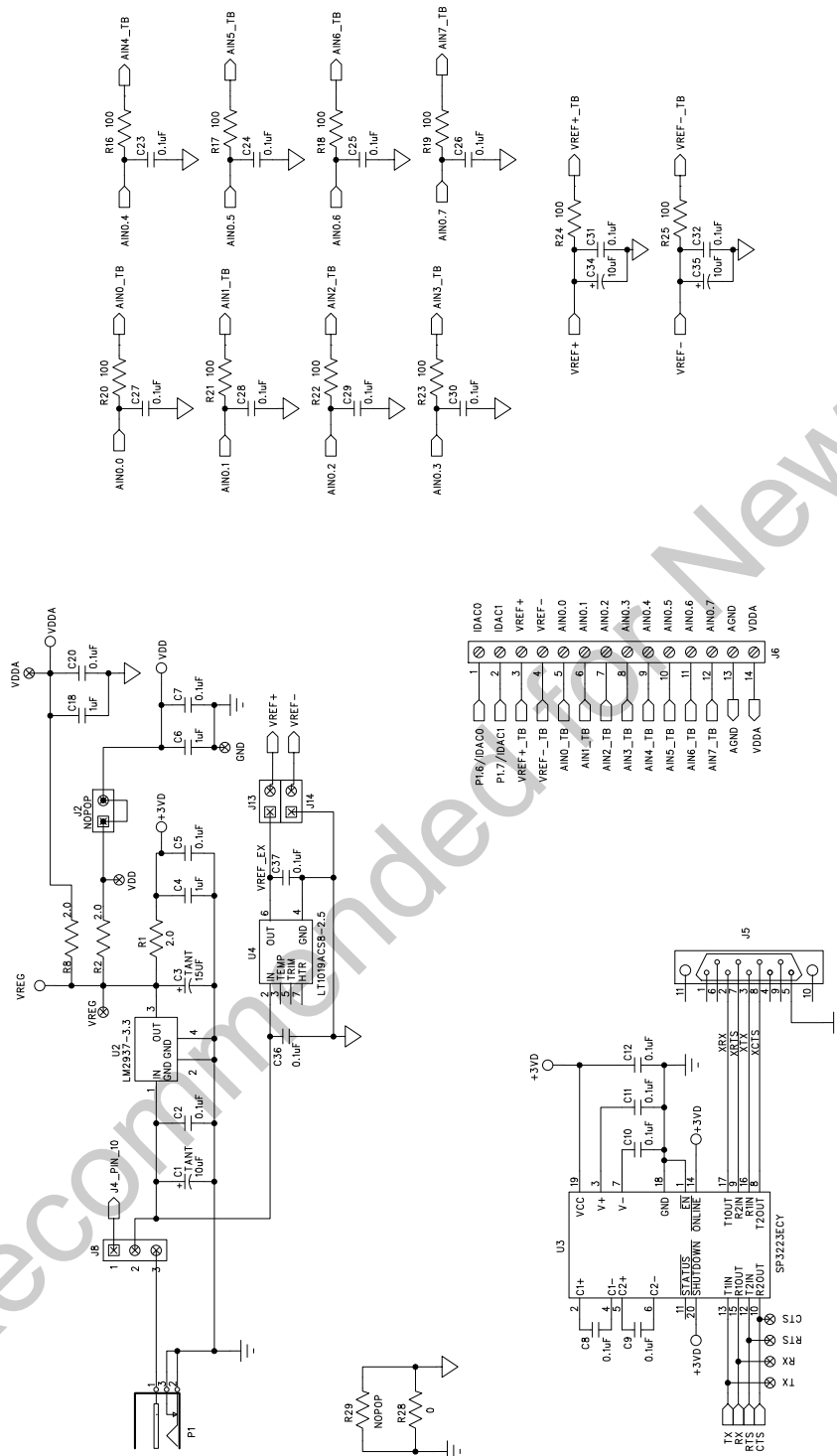


Figure 5. C8051F350 Target Board Schematic - Page 2

## DOCUMENT CHANGE LIST

### Revision 0.1 to Revision 0.2

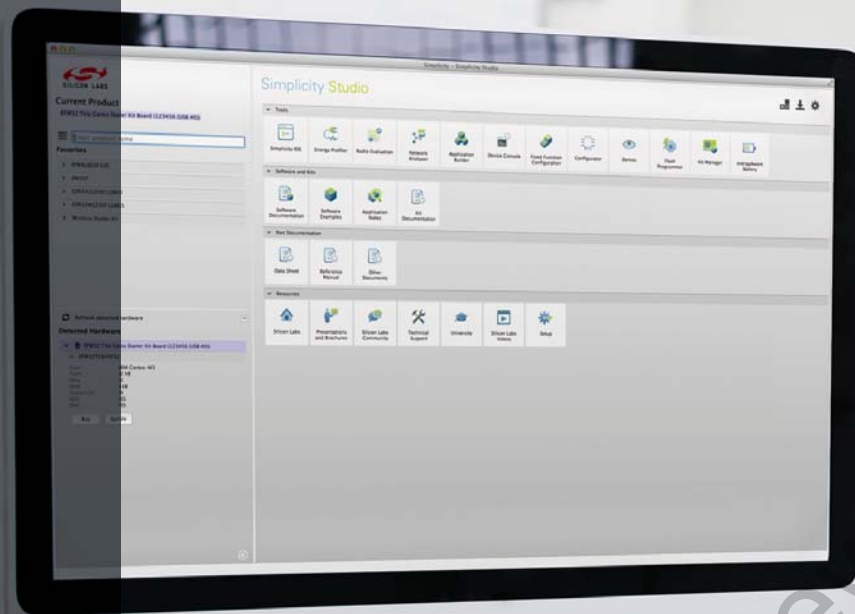
- Section 1, added USB Debug Adapter and USB Cable.
- Section 2, changed name from "Hardware Setup" to "Hardware Setup using an EC2 Serial Adapter".
- Section 2, added 2 Notes bullets.
- Section 2, removed Note from bottom of page.
- Added Section 3, "Hardware Setup using a USB Debug Adapter".
- Section 5.4.2, changed step 2 to include new instructions.
- Section 7, J4, changed "Serial Adapter" to "Debug Adapter".
- Target Board DEBUG Interface Section, added USB Debug Adapter.
- DEBUG Connector Pin Descriptions Table, changed pin 4 to C2D.
- Changed "jumper" to "header".
- EC2 Serial Adapter section, added EC2 to the section title, table title and figure title.
- EC2 Serial Adapter section, changed "JTAG" to "DEBUG".
- Added "USB Debug Adapter" section.
- Section 7, J8, changed "Serial Adapter" to "USB Debug Adapter"
- DEBUG Connector Pin Descriptions Table, changed pin 10 to USB Power
- USB Debug Adapter Target Board Power Connector (J8) Section, changed "Serial" to "USB Debug"

### Revision 0.2 to Revision 0.3

- Removed EC2 Serial Adapter from Kit Contents.
- Removed Section 2. Hardware Setup using an EC2 Serial Adapter. See RS232 Serial Adapter (EC2) User's Guide.
- Removed Section 8. EC2 Serial Adapter. See RS232 Serial Adapter (EC2) User's Guide.
- Removed Section 9. USB Debug Adapter. See USB Debug Adapter User's Guide.

### Revision 0.3 to Revision 0.4

- Updated "Software Setup," on page 2.



## Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



**IoT Portfolio**  
[www.silabs.com/IoT](http://www.silabs.com/IoT)



**SW/HW**  
[www.silabs.com/simplicity](http://www.silabs.com/simplicity)



**Quality**  
[www.silabs.com/quality](http://www.silabs.com/quality)



**Support and Community**  
[community.silabs.com](http://community.silabs.com)

### Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

### Trademark Information

Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOmodem®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.  
400 West Cesar Chavez  
Austin, TX 78701  
USA

<http://www.silabs.com>