User Manual

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

UM-B-049

Abstract

This guide is intended to help customers setup the hardware development environment, install required software, download, and run an example application on the DA14585/586 Pro Development Kit development platform.

The current revision of this guide (V2.0) covers only the **DA14585** and **DA14586** devices. For the **DA148580**, **DA14581** and **DA14583** devices, please refer to revision 1.2

Contents

Figures

Tables

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

1   Terms and Definitions

BLE	Bluetooth® Low Energy
COM	Communication Port
CS	Chip Select
DBG	Debug
EEPROM	Electrically Erasable Programmable Memory
FTDI	Future Technology Devices International
GPIO	General Purpose Input/Output
HW	Hardware
IDE	Integrated Development Environment
IRQ	Interrupt Request
LED	Light Emitting Diode
OS	Operating System
OTP	One Time Programmable
ProDev Kit	Professional Development Kit
RF	Radio frequency
SDK	Software Development Kit
SoC	System on Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SWD	Serial Wire Debug
TCK	JTAG Test Clock
TMS	JTAG Test Mode Select
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

2   Prerequisites

- Minimum hardware requirements:
  - Windows Operating System, 1 GHz, 32-bit or 64-bit processor
  - 1 GB of system memory (RAM)
  - 2 GB of available disk space
- Pro Development Kit DA14585/586 and accessories
- Serial-port terminal software (for example, Tera Term)
- A USB connection supporting USB-Serial (FTDI)

3   Introduction

The DA14585/586 is a family of Bluetooth Smart SoC devices that operate at very low power levels while providing world-class RF performance.

The devices are available in a small footprint while still offering a large flexibility in configuration of the on-chip interfaces making them suitable for a wide range of applications. More information can be found in the DA14585 Datasheet.

The development kit includes a set of hardware (for example, a development board with an on-board debugger) and is supported by a Software Development Kit (SDK) (that is, a development toolchain, source code examples, and so on), and documentation.

The purpose of this guide is to provide an overview of the DA14585/586 ProDev Kit Development Board and describe the setup of the hardware and installation of the software tools to fully use the board.

The following hardware and software elements are required to use the DA14585/586 Development Kit:

- The Pro Development kit and Software Development Kit (SDK)

- SmartSnippets™ Studio which can be installed on Windows, see **Section 8**.

- Windows users should download and install terminal software such as RealTerm, Putty or Tera Term. This document uses Tera Term for its examples.

The rest of guide is organized as follows:

- **Section 6** describes the hardware components and their initial installation and setup.

- **Sections 7** and **8** describe the installation of the SmartSnippets™ DA14585/586 SDK software, along with all necessary tools.

- **Section 9** contains all steps for downloading and executing your first DA14585/586 Applications.

3.1   Kit Content

The ProDev Kit parts can be ordered via various distributors. The Pro Development Kit contains the following:

- Bluetooth low energy motherboard

- The DA14585 daughterboard

- Mini USB-cable

- Coin cell battery (CR2032)

4   The DA14585/586 Hardware

The Pro Development Kit (ProDev Kit) consists of a motherboard (MB-PRO) and a daughterboard featuring the DA14585/586 SoC of the desired package, WLCSP34 or QFN40.

4.1   The ProDev Kit Motherboard

Figure 1 illustrates the physical layout of the ProDev Kit motherboard. The daughter-board containing DA14585/586 device is shown in **Figure 4**.

The ProDev Kit Motherboard provides the necessary hardware to enable:

- Full digital connectivity with external hardware using UART, SPI, GPIO, and I^{2}C

- USB based debugging capabilities using the SEGGER J-Link on-board debugger

- USB based UART communication with the host PC using a Future Technology Devices International (FTDI) chipset which converts UART to USB.

Note: The DA14580 and the DA14585/586 share the same ProDev Kit motherboard (DA14580 DEVKT).

Figure 1: The DA14585/586 ProDev Kit Motherboard

4.2   The Pro Kit daughterboard

Figure 2 presents the DA14585/586 Daughterboard.
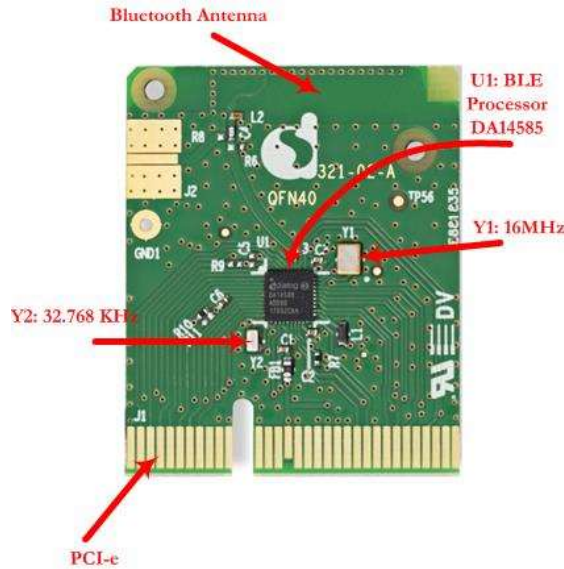
Figure 2: DA14585/586 Daughterboard

4.3   Configuring the Pro Kit Board by Jumper Settings

The DA14585/586 supports booting from UART. The jumper settings are displayed below:
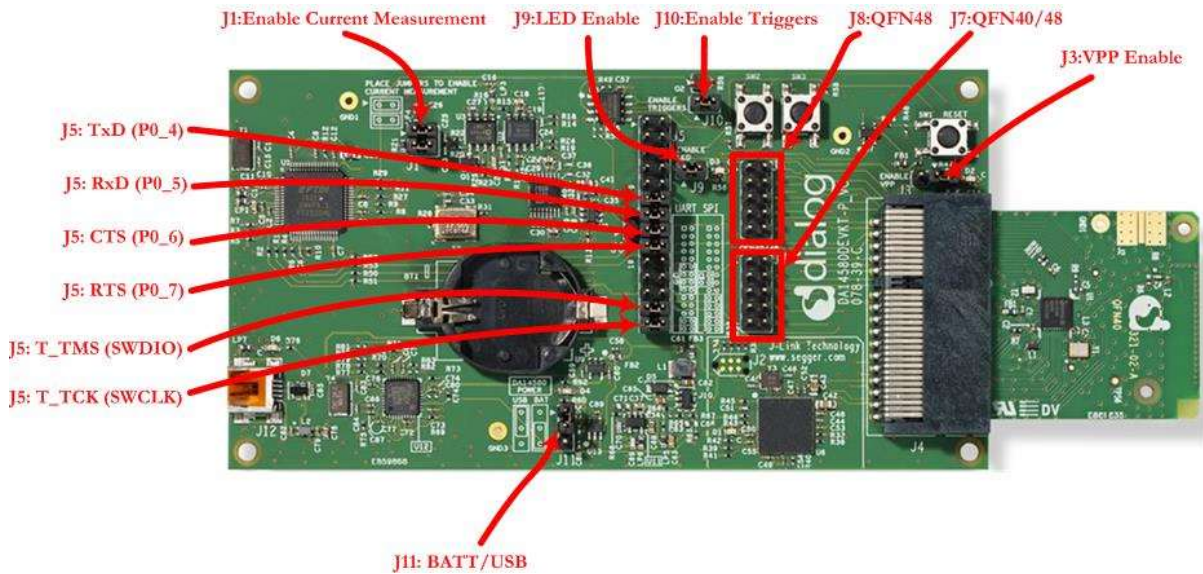
Figure 3: DA14585/586 Jumper Settings

Figure 4 shows the layout of the header jumper, J5.

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit
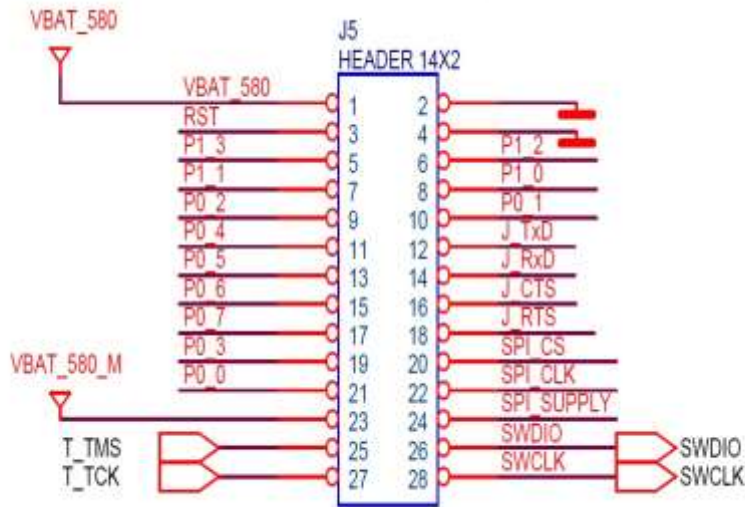
Figure 4: Jumper J5 Layout

4.4 Board Layout

The layout of the boards is displayed in **Figure 5**.
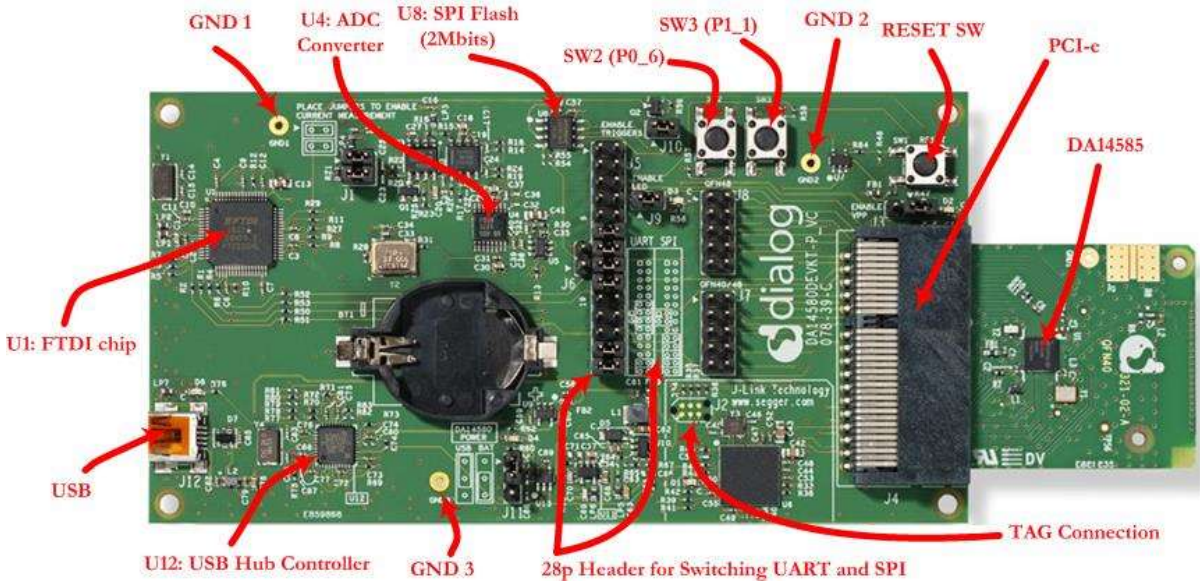
Figure 5: DA14585/586 Board Layout

Note: The schematic diagrams, BOM, and Gerber files are illustrated in detail in the Development Kit Documentation**.**

4.5   Connecting the ProDev Kit to the host PC

The ProDev Kit motherboard includes FTDI chipset **U1** FT2232HLwhich allow easy communication with the development host over USB. The FT2232HL implements the USB to UART function and ensure the connectivity of PC to the DA14585/586 SoC UART port and to current sense circuitry through SPI connection with ADC **U4**. The chip is supplied with a 3.3V from **U13**. A 12MHz crystal **Y1** is required for the chip operation.

The ProDev Kit motherboard also includes an embedded J-Link debugger **U6** SAM3U2CA which ensures the USB to JTAG function by loading the software from Segger to the ROM of U6. The chip is supplied with a 3.3V from **U13** which is enabled by **PWR_ENABLE** signal.

Before connecting the ProDev Kit to the host PC, make sure that the motherboard and the desired daughterboard module are properly connected.

A choice can be made between 3.3 V (via USB: J11 1-2) or VDD (a coin cell: J11 2-3). No battery is needed when running via the mini-USB cable.

Note:

- **The ProDev Kit is connected to the host PC over the connector marked as USB (J12), as shown in Figure 5 using a standard mini-USB cable.**

- **When mounting a battery on the daughterboard, Jumper J11 should be fitted on 2-3 position.**

- **For proper battery functionality, a small modification should be made. See the workaround in Appendix C.**

5   DA14585/586 Software Installation

This section describes the installation procedure for the drivers, the configuration of the serial port, and all necessary steps to verify the connection with the PC. It also provides solutions to problems that may occur.

5.1   Requirements of the Development PC

For proper evaluation and application development using the DA14585/586 SoC and the ProDev Kit, an external host is required. This external host must have a Microsoft Windows already installed and USB ports as described in **Section 6**.

5.2   Driver Installation

On first connection to a host PC running Microsoft Windows, the system will detect several devices and will automatically install all necessary drivers. If the system is configured to use Microsoft Windows Update, this may take several minutes to complete.

When the driver installation is complete, the system displays a Microsoft Windows similar to the one presented in **Figure 6**.

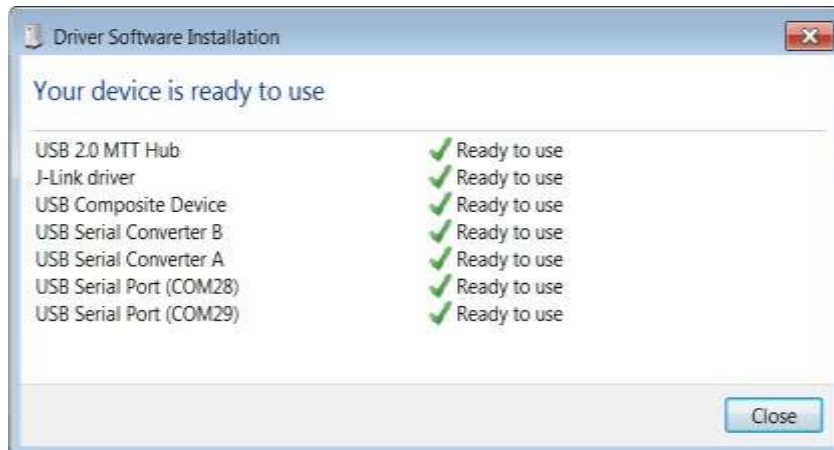## DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

Figure 6: Windows Driver Installation

There are two virtual COM ports created by the Windows driver. The first COM port (lower number, COM28 in this example) provides a UART interface between the PC and the DA14585/586 device. The second (higher number, COM29 in this example) is used to export measurement data from the current sense circuitry on the ProDev Kit to the Power Profiler tool. For more information on the Power Profiler, see the SmartSnippets Toolbox User Manual (UM-B-083).

Note: The COM port numbers assigned to the ProDev Kit motherboard might be different to the ones shown in Figure 6.

Note: If Your PC has a serial port on it, then that's the COM1 you're seeing. Hence you can't make a communication between the ProDev Kit board and the Windows on that port. But you can do it only by using the other lowest COM port, COM28 in this case.

The COM port numbers can be found in the Windows Device Manager (Control Panel > Device Manager > Ports (COM & LPT)) as shown in **Figure 7**.

Figure 7: Device Manager Ports

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

5.3   Configuring the Serial Port for UART

Several development tools require UART to be routed to the FTDI serial port. Refer to **Figure 31** in **Appendix B** for the ProDev Kit board connection verification and on how to properly configure the specific port.

5.3.1   Using Serial Port

To make sure that the communication between the ProDev Kit board and the Windows development host is properly established, it is necessary to verify the UART connection between the two nodes.

On the Windows host, Tera Term can be used to validate the connection to the ProDev Kit by following these steps:

1. Connect the ProDev Kit board to the PC via USB cable. The location of the board's USB port is shown in **Figure 5**.

2. Use the Windows Device Manager to verify that the host discovered two serial ports – the first of which is connected to UART, see **Figure 7**.

3. Open Tera Term from the Windows Start menu as shown in **Figure 8**.

Figure 8: Start Tera Term

4. In the Tera Term: New connection dialog, select Serial, then select the COM Port to use, and click OK.

5. Select Setup > Serial Port and configure your UART port using the parameters shown in **Table 1** (you need to be an administrator on your local machine), see **Figure 9**.

Table 1: Parameters for Connecting to UART

Settings	Values
Baud rate	**115200**
Data bits	**8**
Parity	**None**
Stop bits	**1**
Flow Control	**None**

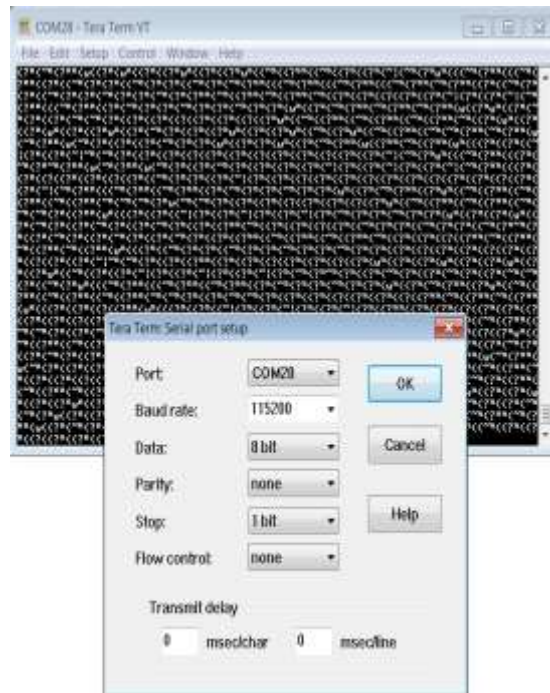**DA14585/DA14586 Getting Started Guide with the PRO-Development Kit**

Figure 9: Setting Up the Port and Testing Connectivity via Tera Term

Note: P0_4 and P0_5 ports are used during the boot sequence. That's why you can see these printed characters which coming out when opening the Tera term.

5.4   Troubleshooting

If there any problems with the ProDev Kit connection to PC some possible solutions might be:

•   Make sure that your FTDI drivers are up to date. Drivers are available from the FTDI website.

•   Check for possible cabling issues by using a different USB cable.

•   Connect the two elements using a different USB port on the host PC.

•   Make sure that the hardware configuration is set for Pro development Kit, see **Figure 30** in **Appendix A**.

Note: If none of these actions resolve the issue, please contact Dialog Tools Forums.

6 Software Development Tools

Dialog SmartSnippets™ Studio is a royalty-free software development platform for Smartbond™ devices. It fully supports the DA14585/586 family of devices.

For the DA14585/586, KEIL must be used for compilation and debug instead of using the supplied SmartSnippets GNU ARM GCC. KEIL needs to be installed separately (it is not included).

SmartSnippets™ Studio contains:

- SmartSnippets™ Toolbox which covers all software development requirements, including:
    - Programming and loading of firmware into SRAM, OTP, and Flash
    - Power profiling
- SmartSnippets™ documentation

The SmartSnippets™ IDE is enabled by an on-board J-Link debugger from SEGGER. This offers standard debug capabilities such as single stepping, setting breakpoints, software download and many more. For more details on the debugger capabilities, visit https://www.segger.com/.

6.1 SmartSnippets™ Installation

The installation procedure for SmartSnippets™ Studio is described in detail in UM-B-057 SmartSnippets Studio User Manual.

A summary of the steps is given here.

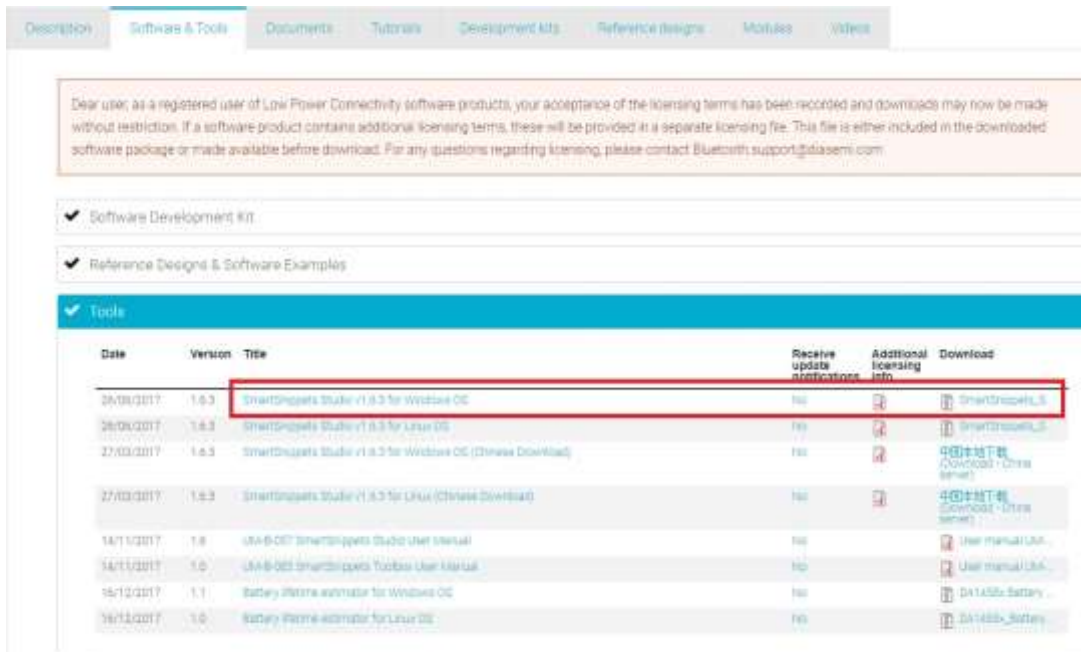1. Download the latest version of SmartSnippets™ Studio from Software and tools, as shown in **Figure 10**.
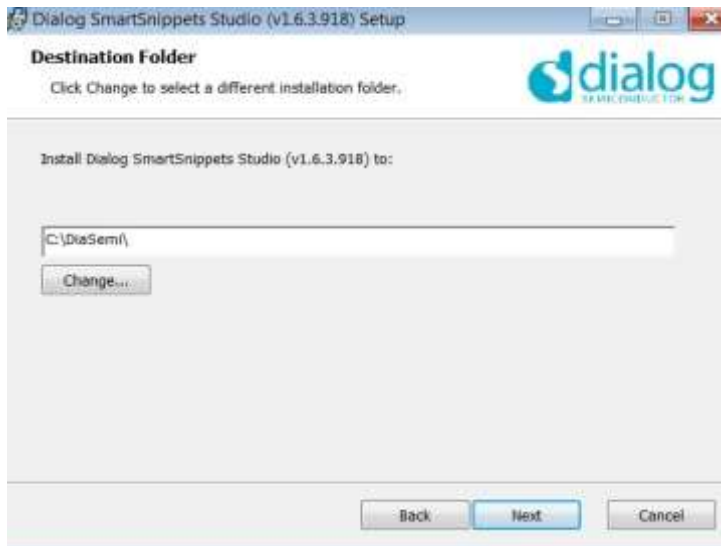

Figure 10: SmartSnippets Studio Install Link

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

2.   Run the SmartSnippets™ Studio installer (.msi). Several of the required tools are automatically installed, others need to be manually downloaded and installed.

3.   Select to install the latest version of SEGGER J-Link GDB server and click **Next**.
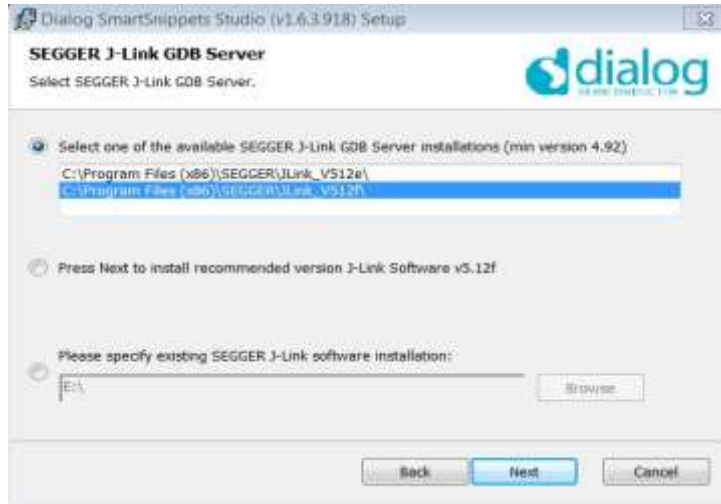
Figure 11: Automatically Install J-Link

4.   Select the destination folder for the SmartSnippets™ Studio and click **Next**.

Figure 12: Select SmartSnippets™ Studio Install Directory

The SmartSnippets™ Studio is installed.

Note:

- **The default Windows installation folder for SmartSnippets™ Studio is C:\DiaSemi.**

- **When Windows Defender is turned ON it will prevent the file SmartSnippets™ Studio installer (.msi) from automatically running, click Run Anyway when prompted.**

- **Administrator access is required to install SmartSnippets Studio software components.**

6.2   KEIL Installation

KEIL µVision IDE must be downloaded and installed separately, registration is required.

1.   Download and install the Keil tools from https://www.keil.com/demo/eval/arm.htm.

Note: The Keil development tools can be run as a Lite/Evaluation version without a product license, see Figure 13 (File > License Management). This Lite/Evaluation version has a 32 kB of code (in ROM) limitation.
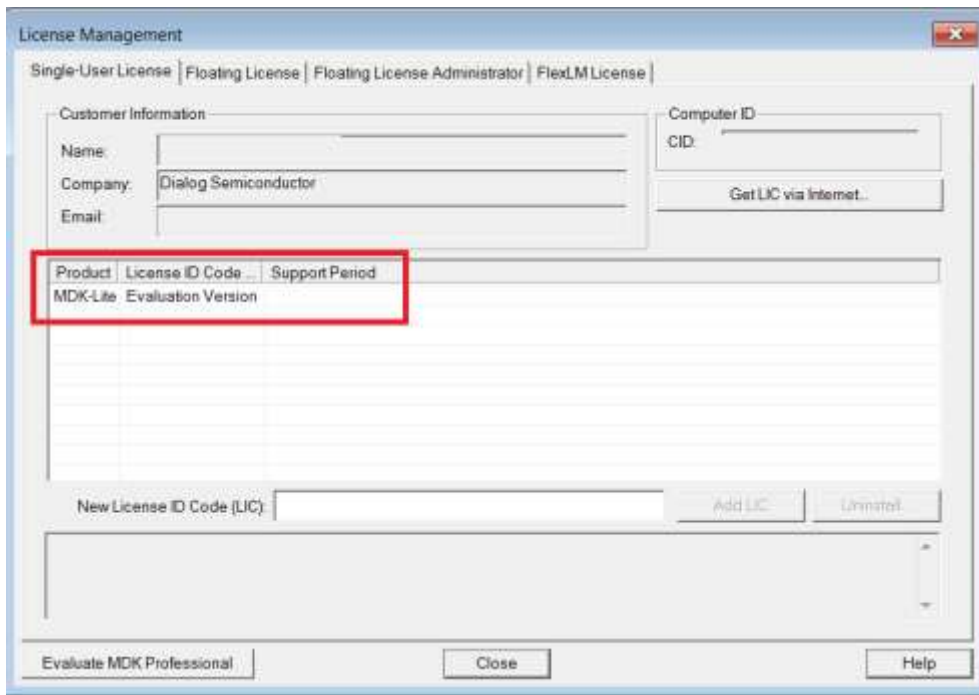

Figure 13: Keil Lite version

2.   When the install is complete, a list of packs is shown, see **Figure 14**. If you do not see this list:

   a.   Click the **Packs Installer** button (or select **Project > Manage > Pack Installer**).

   b.   Click the **Check for Updates** button to show an updated list of packs.

3.   Click on the **Install** or **Update** button next to the "ARM::CMSIS" package if it is not up to date.

Figure 14: CMSIS Package Installation successful

If the installation was successful, the Pack Installer window will show the summary displayed in **Figure 15**.
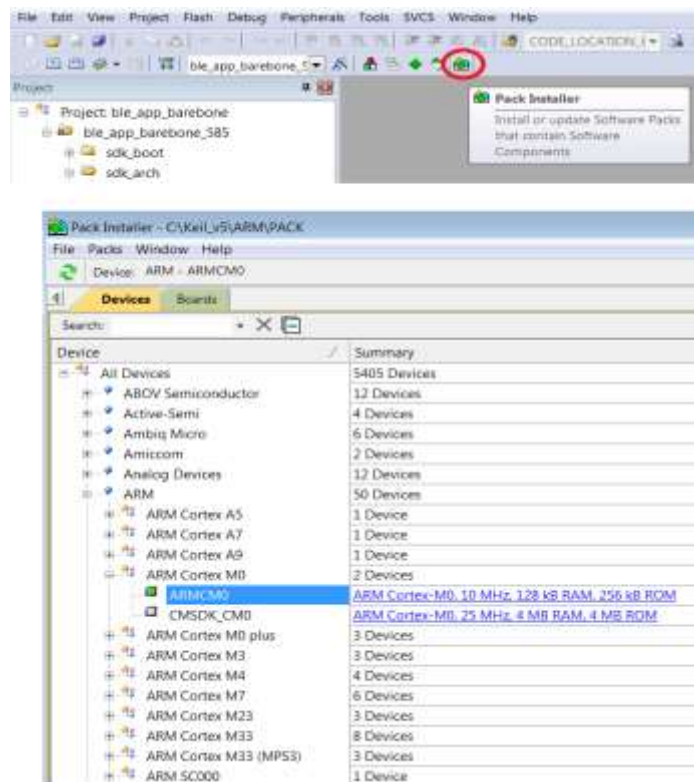

Figure 15: CMSIS Package Installation Successful

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

7 Build a DA14585/586 Application

This section explains how the user can build, program, and run a simple software application on the ProDev Kit development board.

It provides step-by-step instructions for loading the example as a project in the Keil environment, setting it up and building it, and finally executing it via the debug environment on any of the DA14585/586 devices.

Make sure that you have all required tools installed as described in **Section 7** and **Section 8**.

The DA14585/586 Software Development Kit (SDK) includes examples running on the Pro and the Basic Development Kits. The examples are provided with preconfigured KEIL projects.

More details are provided in the DA14585 & DA14586 Software Developer's Guide (UM-B-080).

> To run Blinky or BLE Barebone examples you don't need to install Dialog SmartSnippets™ Studio tool but it is a part of the full Development Environment.

7.1 The Blinky Peripheral Example Application

Blinky is a simple application which demonstrates basic initialization of DA14585/586 and LED blinking. After downloading the SDK, the blinky source code example can be found in the directory called peripheral_examples.

1. Open the folder containing the SDK files. This is the folder where you extracted the SDK zip file.

2. In *<sdk_root_directory>\projects\target_apps\peripheral_examples\blinky\Keil_5*, double-click **blinky.uvproj** to open the project in Keil.
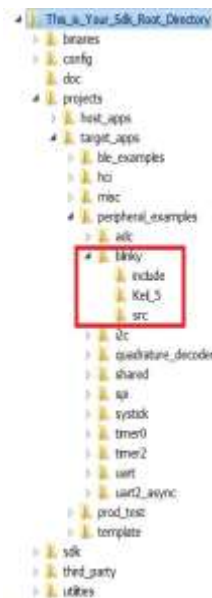
Figure 16: Blinky Project directory

The development environment should look like this when the project is opened with Keil:
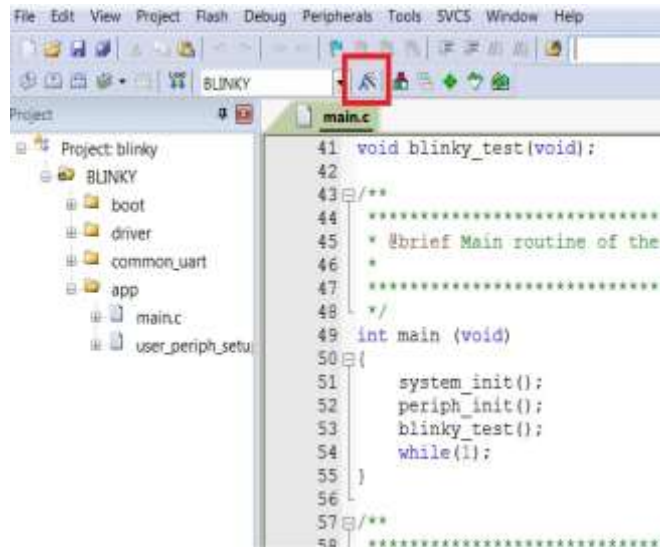
Figure 17: Blinky Project Keil Workspace

3.   Click on the **Target Options** button, then click on the **Device** tab. The dialog should look like this.
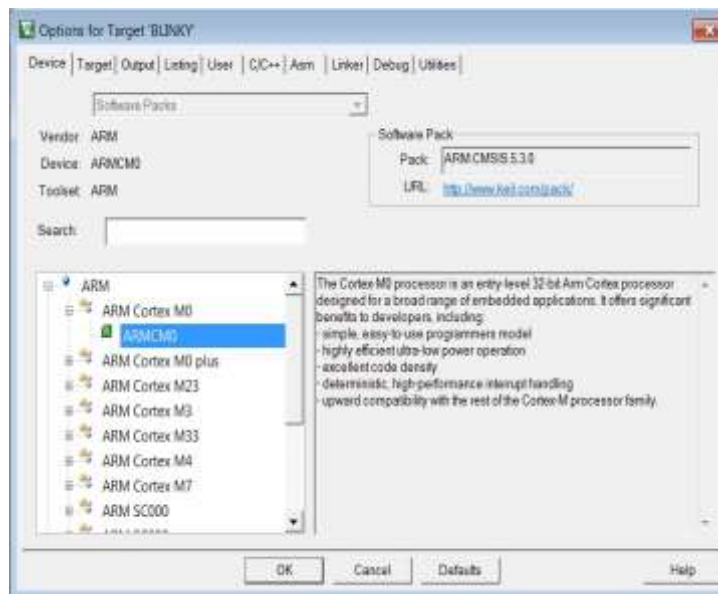
Figure 18: Blinky Project Options

4.   Click on **Linker**. Scatter files (.sct) are used for selecting memory areas.

Figure 19: Blinky Project Scatter File

5. Click on **Debug** and ensure **J-LINK/J-TRACE Cortex** is selected and that the **Initialization File** is set correctly to **.sysram.ini**.

Figure 20: Blinky Project: Debug Option

6. Click on **Settings** (on the right: Debugger) and check that the SW Device has been detected correctly.

Figure 21: Blinky Project: Jlink setup

7.   Click **OK** to save the settings in both windows. All settings have now been saved and you can continue to build the example.

8.   Build the project by pressing **F7** key, or clicking the **Build** button, see **Figure 22**.
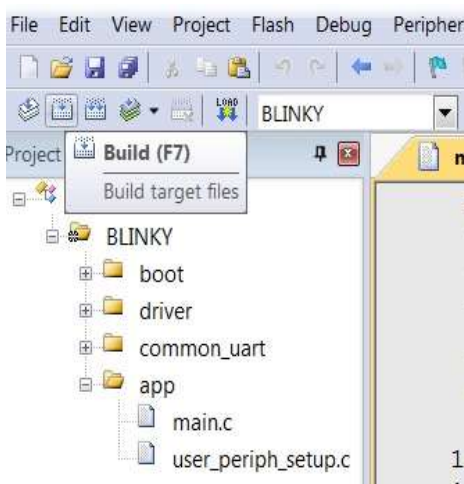
Figure 22: Blinky Project: Project Building

9.   Make sure you have a UART connection between your PC and the motherboard, as shown in **Figure 7**. Check the COM number on your PC.

10.   Open Tera Term, choose the COM port from step 9, and click **OK**.

11.   In Tera Term, Select **Setup > Serial Port** to configure the port as shown in **Table 1**.

12.   In Keil, select **Debug > Start/Stop Debug Session**.
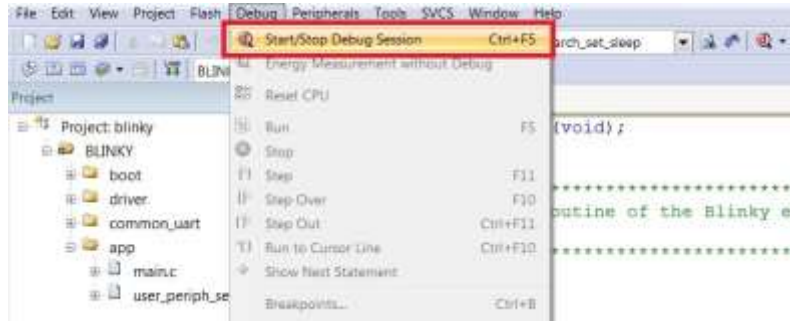
Figure 23: Blinky Project: Start Debug Session

13. If a non-licensed version of Keil is used, the following dialog is displayed. Click **OK**.
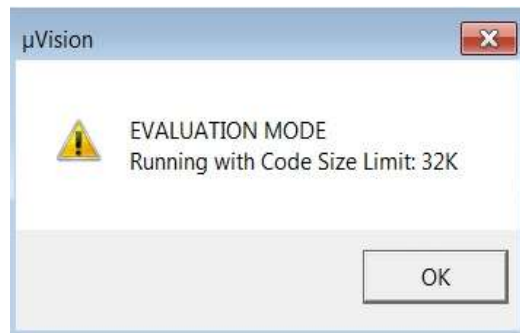
Figure 24: Keil Lite Pop Up Window

14. Press **F5** or click the **Run** button to start code execution.
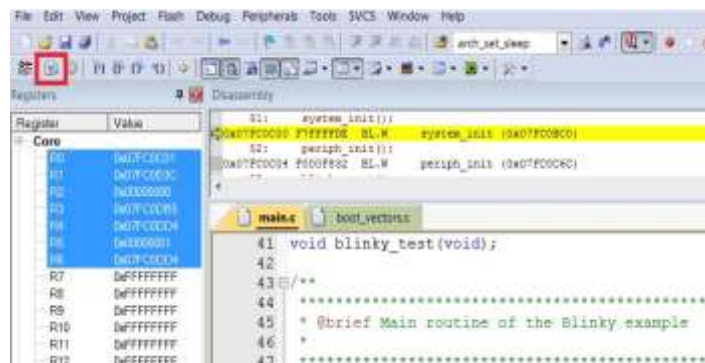
Figure 25: Blinky Project: Code Execution

When the blinky message is displayed on your UART terminal screen and the green LED is blinking you have successfully programmed and started the blinky program on DA14585/586 Demo board.

Figure 26: Blinky Project: Blinky message on COM28 terminal

**Note: The whole application is placed/executed in RAM. For more details about RAM
management you can refer** to How to change the RAM size Tutorial**.**

**Note: SPI Flash (U8: See Figure 5) can also be used for downloading an image file (a .hex or
.bin). for SPI flash programming steps using SmartSnippets Toolbox you can refer to** UM-B-
080: Section 6.8: SPI Flash Memory Example **and for SPI flash programming Keil example you
can refer to** UM-B-083: Section 11: SPI Flash Programmer**. The Figure 27 shows the Jumper
setting for the J5 when using SPI configuration.**

Figure 27: J5 Jumper setting for SPI flash programming

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

7.2 The Barebone BLE Example Application

The Barebone BLE example application demonstrates basic BLE procedures such as advertising, connection, updating connection parameters, and implementation of the Device Information Service Server (DISS). See the Bluetooth SIG for more details.

1. Open the folder containing the SDK files. This is the folder where you extracted the SDK zip file.

2. In *<sdk_root_directory>\projects\target_apps\ble_examples\ble_app_barebone\Keil_5*, double-click **ble_app_barebone.uvproj** to open the project in Keil.

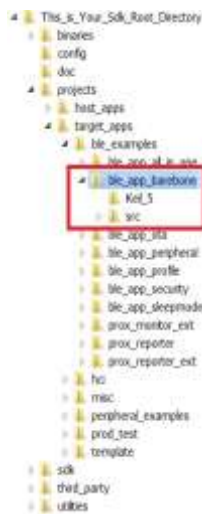**Figure 28: Barebone BLE Project directory**

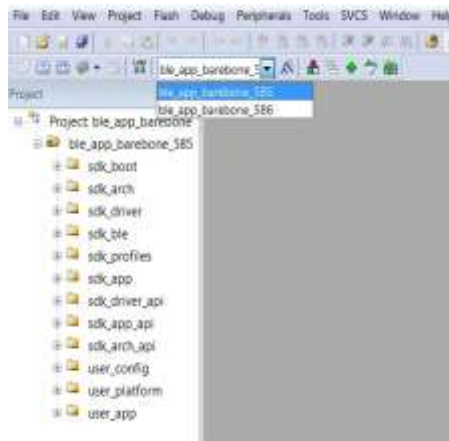**Figure 29** shows the Keil project layout.

Figure 29: Barebone BLE: KEIL Project Layout

3. To execute the code and run the example, repeat steps 3 to 14 in The Blinky Peripheral Example Application.

   The barebone application does not use the UART and so the only jumpers required are those for the debugger interface as noted in **Table 3**.

4. Use the Android application BLE Scanner to scan for, and connect to, the **DIALOG-BRBN** app, see **Figure 30**.
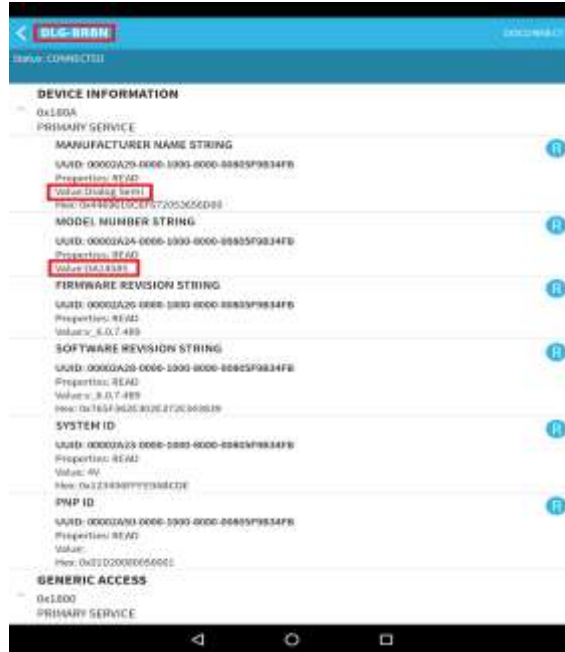

Figure 30: Barebone BLE: Interacting with BLE Application

Your BLE application is now running and advertising using the default advertiser string: "DIALOG-BRBN".

Note: In this example, we used an Android application but you can also use the LightBlue iOS application to connect an iPad/iPod/iPhone device to the application.

Note: When you change the advertising name of your device and you don't want your iPhone/iPad... to connect to it anymore, it's easy to make your device forget about it. Follow easy steps mentioned in the Apple Support Communities **and your iPhone won't remember your Bluetooth device. If you change your mind, no worries, you can pair the device as though it were new.**

When multiple device use the same BD address in the same room test, it is highly recommended to changes manually your BD address or use the static random address (advanced) to avoid any conflict, You can refer to Tutorial 1: Advertising contents

DA14585/DA14586 Getting Started Guide with the PRO-Development Kit

7.2.1   Troubleshooting

If you can't see any blinky messages in your UART terminal and the LED is not blinking, check the UART and LED jumper configuration, see **Table 2**.

Table 2: Blinky Example UART and LED Jumper Configuration

GPIO	Function	Jumper
P0_4	UART2 TX	Connect J5.11 - J5.12
P0_5	UART2 RX	Connect J5.13 - J5.14
P1_0	LED	Connect J9.1 – J9.2

If you can't start a debug session, check the SWD debugger jumper configuration, see **Table 3**.

Table 3: Blinky Example SWD Debugger Jumper Configuration

JTAG	Function	Jumper
TMS	SWD IO	Connect J5.25 - J5.26
TCK	SWD CLK	Connect J5.27 - J5.28

If you have trouble with the J-link firmware or any issue with SDK operation, refer to the DA1458x/68x Development Kit J-Link Interface Application Note AN-B-052.

Note: If none of these actions resolved the issue, please contact Dialog Software Forum.

7.2.2   What Next ?

This tutorial does not cover all the topics relevant to software development environments, it describes the first steps necessary to get started with the Pro Development Kit. The following links provide more information on DA14585/586:

• DA14585 Product Brief: To know more about the SmartBond™ DA14585 SoC.

• DA14586 Product Brief: To know more about the SmartBond™ DA14586 SoC.

• UM-B-079: DA14585/586 SDK 6 Software Platform Reference: To know more about software architecture.

• UM-B-080: DA14585/586 SDK 6 Software Developer's Guide: To build more BLE and peripheral examples.

• Codeless Tutorial: To quick start your IoT application and learn more about AT commands.

• Sleep mode configurations and power measurement: To learn more DA14585/586 Power management.

• Application Hardware Design Guideline : To learn how to prepare schematics and PCB layouts for your application.

8 Appendices

8.1 Appendix A: Using the Pro Development Kit

The user_periph_setup.h file contains hardware settings related to the Development Kit used. The default is the ProDev Kit.
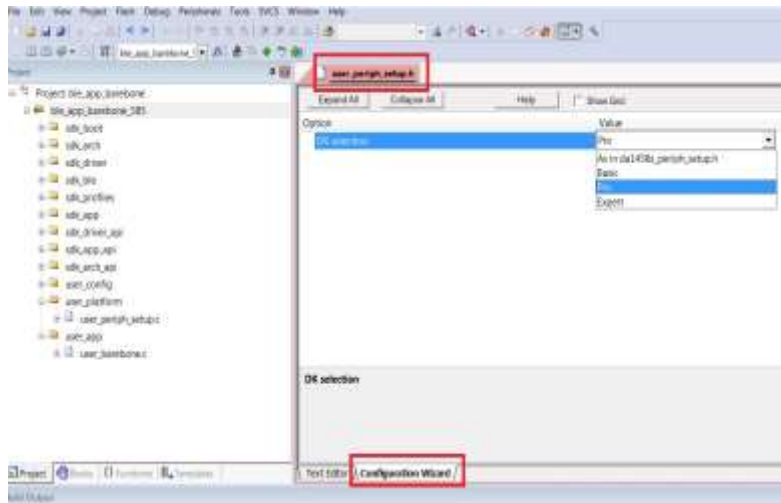

Figure 31: Hardware Settings for Pro Development Kit

8.2 Appendix B: Latency Timer of the FTDI cable

If an external FTDI cable is used to burn the OTP (or to download the image into the external memory), the latency timer of the FTDI cable has to be changed from 15 ms to <10 ms.

To change the latency timer:

1. In the Windows **Device Manager**, expand the list of COM ports.

2. Right click on the required COM port and select **Properties**.

3. In the **Port Settings** tab, click **Advanced**.

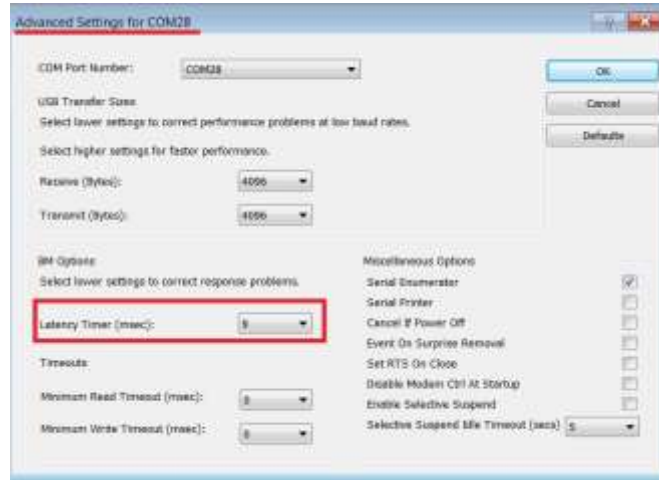4. Set the **Latency Timer** to < 10 ms.

Figure 32: FTDI Latency Timer

8.3 Appendix C: Battery Connection 'Workaround'

There is a small hardware malfunction; when the board operates on battery (J11 jumper on 2-3), the RST signal is permanently high.

When USB is disconnected, T_RESET becomes low and the invertor U7 RST is high. This prevents the board from operating.
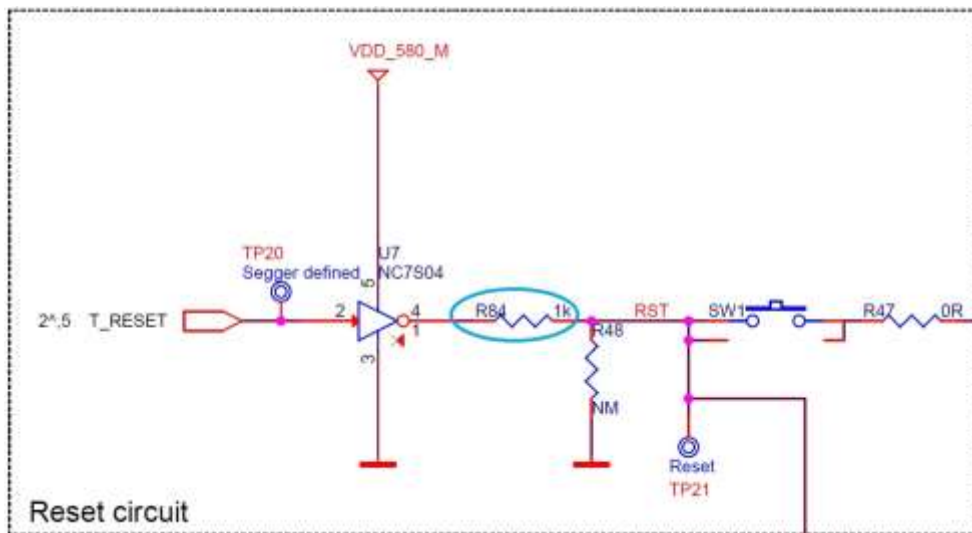
The workaround is to dismount R84, see **Figure 33**.

Figure 33: Reset Circuitry to Dismount R84

Figure 34: R84 Location

8.4 Appendix D: Power Bouncing Workaround

DA14585/586 daughterboards are supplied from the motherboard through a jumper in header J11. See the **3V3 Generation and battery selection for DA14580 and Peripherals** section in the DA14585/586 DEVKT-Pro MAINBOARD Electrical Schematic.

Inserting this jumper when the daughterboard is mounted on the motherboard may cause bouncing issues. A hardware workaround is implemented on DA14585 and DA14586 QFN40 daughterboards.
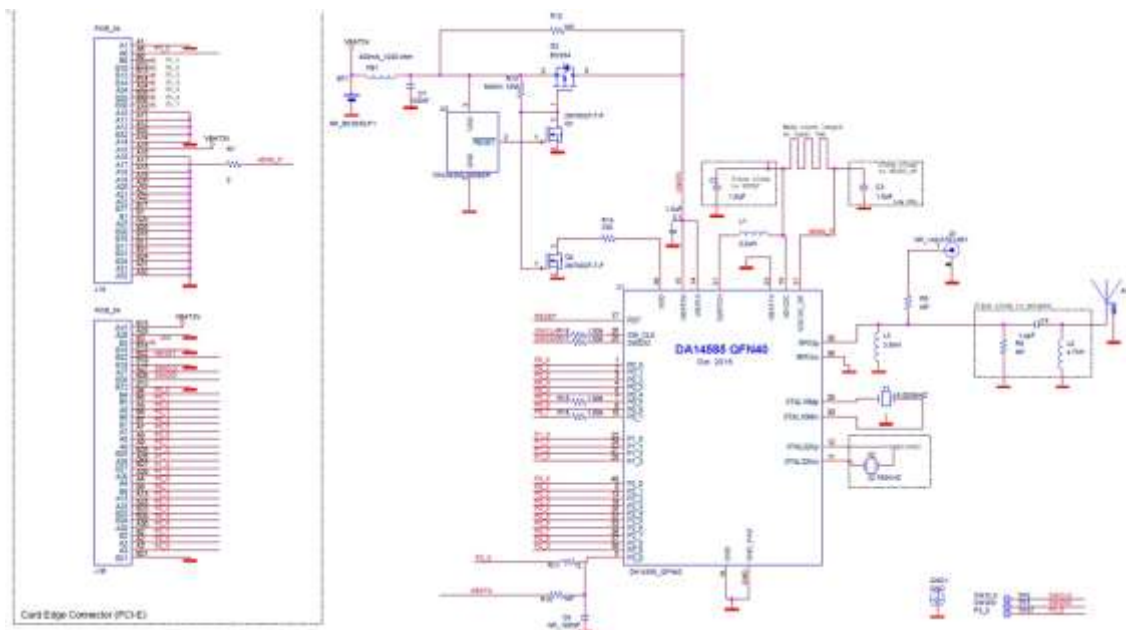

Figure 35: DA14585/586 QFN40 Daughterboard (321-10-x)

Revision history

Revision	Date	Description
1.0	27-Aug-2015	Initial version for DA1458x family with SDK 5.
1.1	20-Oct-2015	Minor Update in Keil installation instructions.
1.2	30-Mar-2017	Add DA14585/586.
2.0	31-May-2018	First HTML released version. Keep only DA14585/586.
2.1	02-Nov-2018	Changes implemented: ● Minor text revisions
2.2	20-Jan-2022	

Status Definitions

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

RoHS Compliance

Dialog Semiconductor's suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Development Boards & Kits - Wireless *category:*

Click to view products by Dialog Semiconductor *manufacturer:*

Other Similar products are found below :

TWR-13237-KIT K32W-001-T10 C METER CLICK HALL CURRENT CLICK COLOR 3 CLICK COMPASS 2 CLICK HDC1000 CLICK RELAY CLICK RFID CLICK RIVERDI CLICK CUGSM113#UFL MATRIX G CLICK MATRIX RGB CLICK 3D MOTION CLICK 4-20MA R CLICK 4-20MA T CLICK THERMO CLICK MCP2003B CLICK EXPAND 3 CLICK MCP2542 CLICK MCP25625 CLICK ATA663211 CLICK TOUCHKEY CLICK MICROSD CLICK TRF CLICK BUTTON R CLICK I2C ISOLATOR CLICK UNIQUE ID CLICK USB SPI CLICK CAN-SPI CLICK 3.3V CAN-SPI CLICK 5V CAP EXTEND CLICK CAPSENSE CLICK 13DOF 2 CLICK 13DOF CLICK DC MOTOR CLICK WIFI2 CLICK 4X4 KEY CLICK RS232 CLICK RS485 CLICK 5V RTC2 CLICK EM3588-MLR-AN-C CC3100 CLICK FLASH 3 CLICK NFC CLICK NFC TAG CLICK FLICKER CLICK NRF C CLICK NRF S CLICK SPEAKUP CLICK