

User Manual

DA14580/581/583 Bluetooth Smart Development Kit – Pro

UM-B-034

Abstract

This document describes the Bluetooth Smart Development Kit - Pro based on DA14580/581/583.

It helps users to set up the hardware development environment, install required software and quickly start product development with help of example source code.

Contents

Abstract	1
Contents	2
Tables	3
Figures	3
1 Terms and Definitions	4
2 References	4
3 Introduction	5
3.1 Hardware Content	6
3.2 Web Content	8
3.2.1 Software Development Kit Content	8
3.3 DA14580 - DA14581 Pro Kit	12
3.3.1 Differences Between the DA14580, DA14581 and DA14583	12
3.4 Pinning	14
3.4.1 PCB Design and Functionalities	15
3.4.2 Configuring the Pro Kit-Board by Jumper Settings.....	16
3.4.3 Block diagram	18
4 Installation of Tools and Drivers	19
4.1 Keil	19
4.2 SEGGER Jlink Driver	20
4.3 FTDI Driver.....	21
4.4 Tera Term	21
5 Memory and Tools	22
6 Using the Demo Kit	23
6.1 Run an Example on DA14580/581/583	23
Appendix A Layout	28
Appendix B Connections of J7 and J8	28
Appendix C Schematics Motherboard and Daughterboards	29
C.1 Peripherals	29
C.2 SPI and Serial	30
C.3 Current Measurement	31
C.4 JTAG Interface	32
C.5 PCI-e	33
C.6 Power Supplies	34
C.7 USB HUB Controller.....	35
C.8 DA14580 WLCSP Daughter board	36
C.9 DA14580 QFN40 Daughterboard	37
C.10 DA14580 QFN48 Daughterboard	38
C.11 DA14581 WLCSP Daughterboard	39
C.12 DA14581 QFN40 Daughterboard	40
C.13 DA14583 QFN40 Daughterboard	41
Appendix D Using the SmartSnippets CLI	42

DA14580/581/583 Bluetooth Smart Development Kit – Pro

Appendix E Opening Your Project for the First Time	43
Appendix F Latency Timer of FTDI Cable	44
Appendix G RF-Testing DEVKT - Pro	45
Appendix H Battery Connection “Workaround”	48
7 Web-Link	49
8 Revision History	49

Tables

Table 1: Content of the DA14580/581/583 Pro Kit	7
Table 2: DA14580	12
Table 3: DA14581	12
Table 4: DA14583	13
Table 5: SPI Connections	14
Table 6: Installation Tools and Drivers	19
Table 7: Run an Example on DA14580/581/583	23

Figures

Figure 1: DA14580/581 - Pro kit	6
Figure 2: DA14583 - Pro Kit	6
Figure 3: QFN40 Pin Assignment from Datasheet	14
Figure 4: DA14580 Development Kit – Pro (PCB topview)	15
Figure 5: DA14580/581/583 (Fabrication Default) UART Boot Settings (T_TxD : (P0_5) and T_RxD : (P0_4))	16
Figure 6: DA14580/581 Boot from External SPI Memory	16
Figure 7: DA14583 Boot from Internal SPI Memory	17
Figure 8: Layout of headers J5 and J6	17
Figure 9: UART Data Direction of UART within J5	17
Figure 10: Block Diagram of Total System	18
Figure 11: DA14580/581/583 Block Diagram	22
Figure 12: Board Layout	28
Figure 13: IO breakout available on QFN40 and QFN48	28
Figure 14: IO Breakout Available on QFN48	28
Figure 15: Peripherals Schematic of Motherboard	29
Figure 16: SPI and Serial Schematic of Motherboard	30
Figure 17: Current Measurement Schematic of Motherboard	31
Figure 18: JTAG Interface Schematic of Motherboard	32
Figure 19: PCI-e Schematic of Motherboard	33
Figure 20: Power Supplies Schematics of Motherboard	34
Figure 21: USB HUB Controller Schematic of Motherboard	35
Figure 22: DA14580 WLCSP Daughter Board	36
Figure 23: DA14580 QFN40 Daughterboard	37
Figure 24: DA14580 QFN48 Daughterboard	38
Figure 25: DA14581 WLCSP Daughterboard	39
Figure 26: DA14581 QFN40 Daughterboard	40
Figure 27: DA14583 QFN40 Daughterboard	41
Figure 28: Open the CLI of SmartSnippets	42
Figure 29: Smart Snippet CMD Window	42
Figure 30: FTDI Latency Timer	44
Figure 31: Reset Circuitry. R84 to be Dismounted	48
Figure 32: Location of R84	48

1 Terms and Definitions

BLE	Bluetooth Low Energy
CS	Chip Select
DK	Development Kit
EEPROM	Electrically Erasable Programmable Memory
FTDI	Brand name of USB – UART interface
GPIO	General Purpose Input Output
OTP	One Time Programmable
PCB	printed circuit board
QFN	Quad-Flat No-leads
SDK	Software Development Kit
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SWD	Serial Wire Debug
USB	Universal Serial Bus
UART	Universal Asynchronous Receiver/Transceiver
WLCSP	Wafer Level Chip Scale Packaging
WoW	Way of Working

2 References

1. DA14580, Datasheet, Dialog Semiconductor
2. DA14581, Datasheet, Dialog Semiconductor
3. UM-B-015, DA14580_581_583 Software Architecture, Dialog Semiconductor
4. DA14580_CB PXI QFN40 layout, Dialog Semiconductor
5. DA14580_CB_PXI_QFNP40, Dialog Semiconductor
6. DA14580_CB_PXI_WLCSP, Dialog Semiconductor
7. DA14580_CB_PXI_WLCSP_layout, Dialog Semiconductor
8. DA14580_MB_VB_layout, Dialog Semiconductor
9. DA14580_CB PXI_QFN48, Dialog Semiconductor
10. UM-B-005, DA14580_581_583 Peripheral Examples, Dialog Semiconductor
11. UM-B-010, DA14580_581_583 Proximity application, Dialog Semiconductor
12. AN-B-015, DA14580 Supply current measurement, Dialog Semiconductor

DA14580/581/583 Bluetooth Smart Development Kit – Pro

3 Introduction

DA14580/581/583 is a Bluetooth Smart chip, working with extremely low power while providing world-class RF performance, a small footprint and flexible peripheral configurations for a wide range of applications. The development kit includes a set of hardware (e.g. development board with on-board debugger), a Software Development Kit (SDK) (e.g. development tools, source code examples documents and so on) along with documentation. This document helps users to set up hardware/software development environment, install required software and quickly start product development with the help of example source code.

Web content can be downloaded at:

www.dialog-semiconductor.com/support.

Product information about the DA14580/581/583 can be found at:

<http://www.dialog-semiconductor.com/products/bluetooth-smart>

Product information about the DA14580/581/583 Development Kit - Pro can be found at:

<http://www.dialog-semiconductor.com/products/bluetooth-smart/smartbond-development-tools/da14580-development-kit-pro>

The DA14583 is DA14580 plus 1Mbit SPI memory in same package.

UM-B-034

DA14580/581/583 Bluetooth Smart Development Kit – Pro

3.1 Hardware Content

In [Figure 1](#) and [Figure 2](#), the kit components are shown and [Table 1](#) contains an overview the parts.



Figure 1: DA14580/581 - Pro kit



Figure 2: DA14583 - Pro Kit

DA14580/581/583 Bluetooth Smart Development Kit – Pro
Table 1: Content of the DA14580/581/583 Pro Kit

DA14580/581/583 DEVKT – PRO:	Included in the kit	Separate option for the 580 kit*	Separate option for the 581 kit*	Separate option for the 583 kit*
Battery CR2032	X			
Mini USB Cable	X			
DA14580A3DB-P (QFN48)		X		
DA14580ATDB-P (QFN40)		X		
DA14580UNDB-P (WL-CSP)		X		
DA14581UNDB-P (WL-CSP)			X	
DA14581ATDB-P (QFN40)			X	
DA14583ATDB-P (QFN40)				X
DA14580DEVKT-P_VB (Main board)**	X			
USB Dongle	X			

Note 1 * Not included in the kit, must be bought separately.

Note 2 ** The mainboard is compatible with all boards.

3.2 Web Content

3.2.1 Software Development Kit Content

3.2.1.1 Tools

Web-link: www.dialog-semiconductor.com/support

SmartSnippets

SmartSnippets is a framework of PC based tools to control DA14580/581/583 development kit, consisting of:

- Power Profiler : Real time current consumption measurement to for the DA14580/581/583 motherboard
- OTP Programmer: Tool for OTP memory programming
- UART/JTAG booter: Tool for downloading hex files to DA14580/581/583 SRAM over UART or JTAG
- SPI & EEPROM programmer: A tool for SPI & EEPROM flash programming
- Sleep Mode Advisor: Calculation tool to determine most optimal sleep modes

ConnectionManager

ConnectionManager is a PC based software tool to control the link layer of the DA14580/581/583 with the following capabilities:

- Functional in Peripheral and Central role
- Set advertising parameters
- Set connection parameters
- Reading from Attribute database
- Perform production test commands

3.2.1.2 SDK documents

- UM-B-003, DA14580_581_583 Software development guide
- UM-B-004, DA14580_581_583 Peripheral drivers
- UM-B-005, DA14580_581_583 Peripheral examples
- UM-B-006, DA14580 Sleep mode configuration
- UM-B-007, DA14580 Software Patching over the Air (SPOTA)
- UM-B-008, DA14580 Production test tool
- UM-B-010, DA14580_581_583 Proximity application
- UM-B-011, DA14580 Memory map – scatter file
- UM-B-012, DA14580 Secondary boot loader
- UM-B-013, DA14580 External Processor Interface over SPI
- UM-B-014, DA14580 Bluetooth Smart Development Kit - Expert
- UM-B-015, DA14580_581_583 Software architecture
- UM-B-016, DA14580 Software Porting Guide
- UM-B-017, DA14580 GTL interface Integrated Processor Application

3.2.1.3 SDK Source Code Examples (Created in Keil)

- **dk_apps.** This folder holds all the necessary folders needed for DA14580/581/583 application development.
 - **dk_apps\keil_projects\proximity:**
The folder contains the following subfolders and in each one of them resides the respective project file:

DA14580/581/583 Bluetooth Smart Development Kit – Pro

Table 1: SDK Examples

Folder	Project File	Description
prox_monitor_ext\Keil_4 prox_monitor_ext\Keil_5	prox_monitor_ext.uvproj prox_monitor_ext_581.uvproj prox_monitor_ext_583.uvproj prox_monitor_ext.uvprojx prox_monitor_ext_581.uvprojx prox_monitor_ext_583.uvprojx	Proximity Monitor (External processor configuration)
prox_reporter_ext\Keil_4 prox_reporter_ext\Keil_5	prox_reporter_ext.uvproj prox_reporter_ext_581.uvproj prox_reporter_ext_583.uvproj prox_reporter_ext.uvprojx prox_reporter_ext_581.uvprojx prox_reporter_ext_583.uvprojx	Proximity Reporter (External processor configuration)
prox_reporter\Keil_4 prox_reporter\Keil_5	prox_reporter.uvproj prox_reporter_581.uvproj prox_reporter_583.uvproj prox_reporter.uvprojx prox_reporter_581.uvprojx prox_reporter_583.uvprojx	Proximity Reporter (Integrated processor configuration)
prox_monitor_ext_usb\Keil_4 prox_monitor_ext_usb\Keil_5	prox_monitor_ext_usb.uvproj prox_monitor_ext_usb_581.uvproj prox_monitor_ext_usb_583.uvproj prox_monitor_ext_usb.uvprojx prox_monitor_ext_usb_581.uvprojx prox_monitor_ext_usb_583.uvprojx	Proximity Monitor (External processor configuration) Version for USB dongle
prox_reporter_ext_usb\Keil_4 prox_reporter_ext_usb\Keil_5	prox_reporter_ext_usb.uvproj prox_reporter_ext_usb_581.uvproj prox_reporter_ext_usb_583.uvproj prox_reporter_ext_usb.uvprojx prox_reporter_ext_usb_581.uvprojx prox_reporter_ext_usb_583.uvprojx	Proximity Reporter (External processor configuration) Version for USB dongle
prox_reporter_ext_spi\Keil_4 prox_reporter_ext_spi\Keil_5	prox_reporter_ext_spi.uvproj prox_reporter_ext_spi_581.uvproj prox_reporter_ext_spi_583.uvproj prox_reporter_ext_spi.uvprojx prox_reporter_ext_spi_581.uvprojx prox_reporter_ext_spi_583.uvprojx	Proximity Reporter (External processor) SPI version

- **dk_apps\keil_projects\prod_test:** These folders include the source code of the production test firmware. Refer to DA14580_581_583_Production_Test_Tool.docx for more information how to build and use it.
- **dk_apps\keil_projects\template:** These folders include a template as a starting point of a new application.

For details, please read [9].

DA14580/581/583 Bluetooth Smart Development Kit – Pro

- **host_apps:** This folder holds the DA14580/581/583 PC applications:
 - **host_apps\windows\proximity:** The folder includes two Windows C++ applications, with each one acting as part of a proximity monitor and a proximity reporter application. They are placed in subfolders *monitor* and *reporter* respectively. For details, please read the *DA14580_581_583 Proximity Application Guide*.
 - **host_binaries\windows\proximity:** The folder includes two pre-compiled Windows executables which correspond to the C++ applications described right above and are included for user convenience.
 - **peripheral_examples:** The folder includes sample code of how to use peripheral blocks of the DA14580/581/583 (e.g. UART, SPI, I2C etc.) bundled to a demo-kit. For details, please refer to [10].
- Tools:
 - **tools\prod_test\prod_test_cmds:** This folder includes the source code of the production test tool. Refer to DA14580/DA14581_Production_Test_Tool.docx for more information how to build and use it.

DA14580/581/583 Bluetooth Smart Development Kit – Pro
3.3 DA14580 - DA14581 Pro Kit
3.3.1 Differences Between the DA14580, DA14581 and DA14583

The only hardware difference between the daughter boards of the DEVKT-Pro 14580 and the DEFKT-Pro 14581, is the design in of QFN48 (580) and the QFN40 (581). The silkscreen may have small textual differences.

The DA14581 uses a dedicated ROM which offers optimisations **targeting A4WP and HCI**.

Table 2: DA14580

Product	Memory size	General Purpose IO's	Package	Key Features	Applications
WLCSP34	ROM 84kBytes OTP 32kBytes RAM 50kBytes	12	2.5x2.5x0.5mm, pitch 0.4mm	<ul style="list-style-type: none"> Bluetooth 4.0 + 4.1 Cortex M0 application processor Power supply 0.9 -3.3V Single pin RF I/O Rich set of analog and digital peripherals 	Beacon & Proximity Health & Fitness HID Smart Home
QFN40		24	5x5x0.9mm, pitch 0.4mm		
QFN48		32	6x6x0.9mm, pitch 0.4mm		

Table 3: DA14581

Product	Memory size	General Purpose IO's	Package	Key Features	Applications
WLCSP34	ROM 84kBytes OTP 32kBytes RAM 50kBytes	12	2.5x2.5x0.5mm, pitch 0.4mm	<ul style="list-style-type: none"> Bluetooth 4.0 + 4.1 Cortex M0 application processor Power supply 0.9 -3.3V Single pin RF I/O Rich set of analog and digital peripherals 8 connections Optimized boot time 	Wireless charging (A4WP) HCI
QFN40		24	5x5x0.9mm, pitch 0.4mm		

DA14580/581/583 Bluetooth Smart Development Kit – Pro

Table 4: DA14583

Product	Memory size	General Purpose IO's	Package	Key Features	Applications
QFN40	ROM 84kBytes OTP 32kBytes RAM 50kBytes	24	5x5x0.9mm, pitch 0.4mm	<ul style="list-style-type: none"> Bluetooth 4.0 + 4.1 Cortex M0 application processor Power supply 2.35 -3.3V Single pin RF I/O Rich set of analog and digital peripherals 	Beacon & Proximity Health & Fitness HID Smart Home

- A4WP wireless charging features:
 - Fast boot time for Power Receiving Unit (PRU)
 - 8 connections for Power Transmitting Unit (PTU)
- HCI features:
 - Optimized code for HCI which fits into the OTP
 - This enables customers/modules makers to provide a pre-programmed HCI module

Remark: DA14583 will run in BUCK mode only!

DA14580/581/583 Bluetooth Smart Development Kit – Pro

3.4 Pinning

In Figure 3 the pinout of the DA14583 is shown. New, compared to the DA14580/581, are the connections to the internal SPI flash memory.

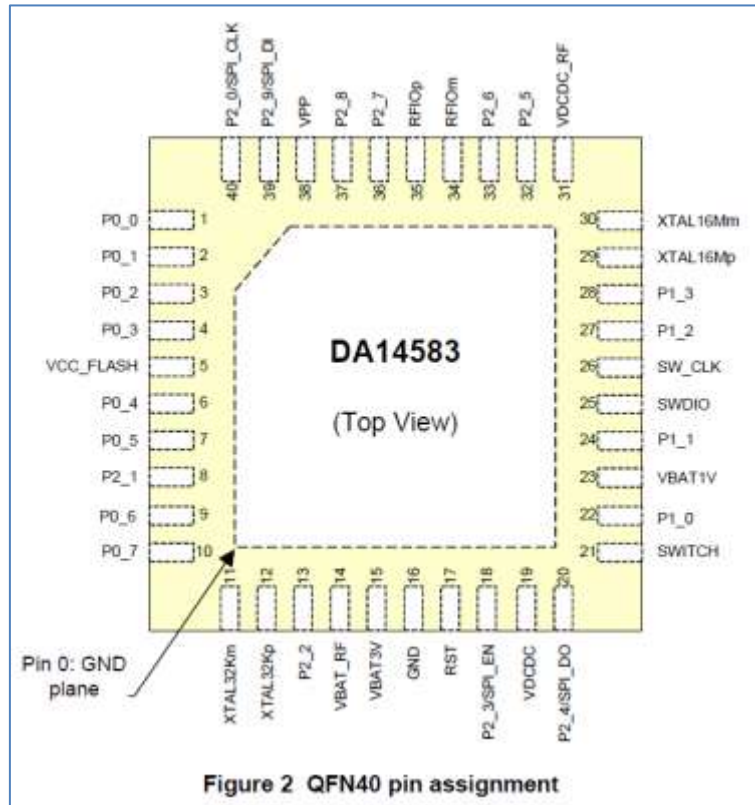


Figure 3: QFN40 Pin Assignment from Datasheet

Table 5: SPI Connections

Port DA14583	Function	Remark
P2_0	SPI_CLK	SCLK (Note 1)
P2_9	SPI_DI	MOSI (Note 1)
P2_4	SPI_DO	MISO (Note 1)
P2_3	SPI_EN	not to be used for external SPI (!)
	VCC_FLASH	power for internal Flash Memory
	GND	

Note 1 shared with internal flash memory

When external SPI components are used, SPI_EN is occupied for internal use. Another pin should be chosen for SPI_EN of the external component.

By using a Secondary Bootloader the proper pins are programmed to load the booting software from the SPI-memory at startup.

DA14580/581/583 Bluetooth Smart Development Kit – Pro

3.4.1 PCB Design and Functionalities

The top-screen layer of the pro kit PCB is shown below in Figure 4.

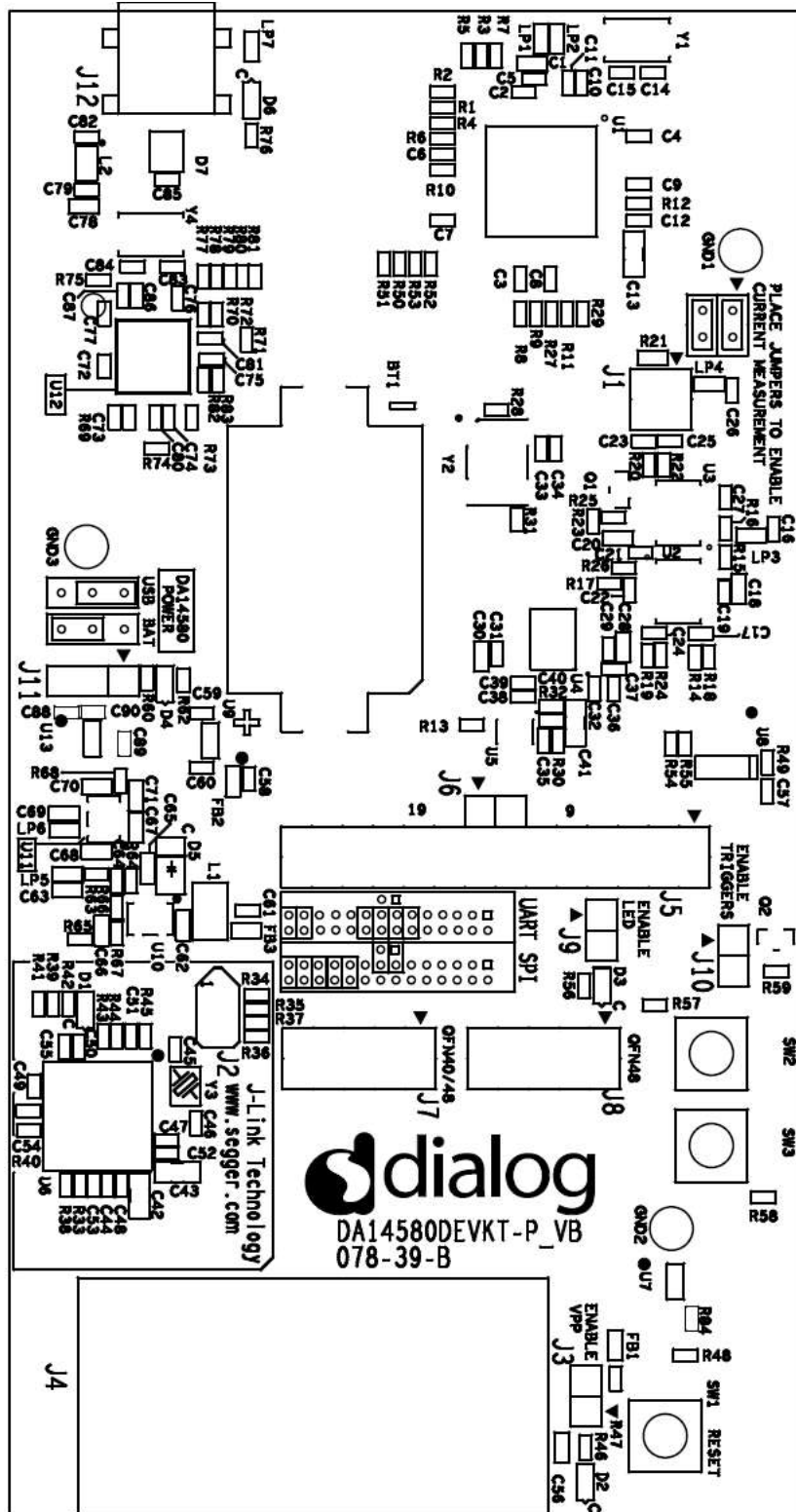


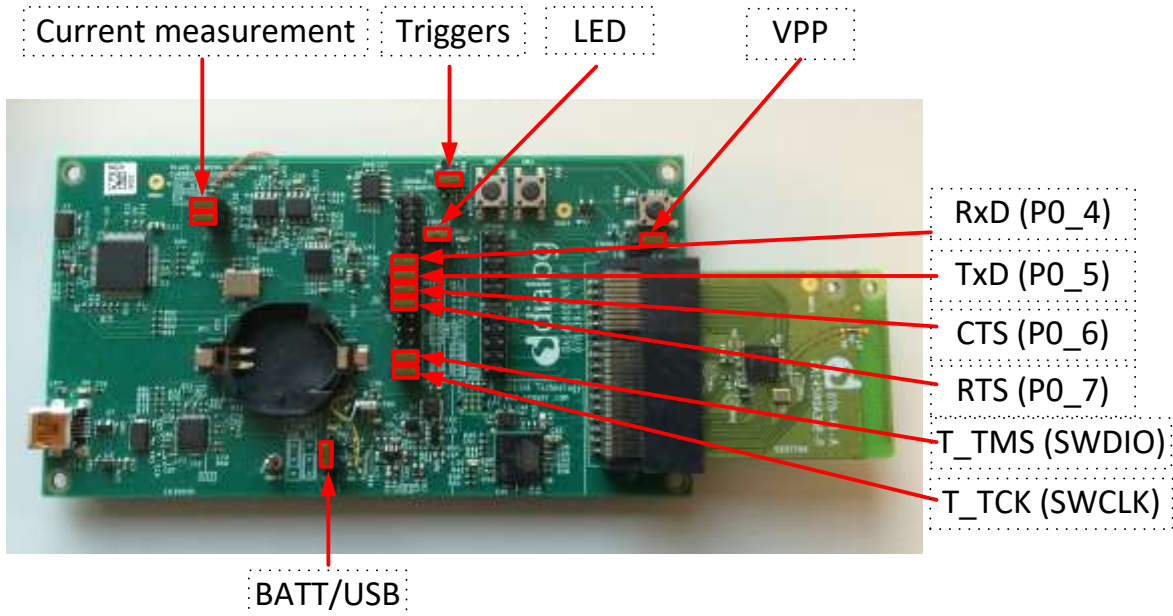
Figure 4: DA14580 Development Kit – Pro (PCB topview)

DA14580/581/583 Bluetooth Smart Development Kit – Pro

3.4.2 Configuring the Pro Kit-Board by Jumper Settings

There are two configurations that can be switched; the default configuration that supports the boot from UART or the configuration that supports boot from an external SPI flash memory.

The jumper settings are displayed below.



**Figure 5: DA14580/581/583 (Fabrication Default)
UART Boot Settings (T_TxD : (P0_5) and T_RxD : (P0_4))**

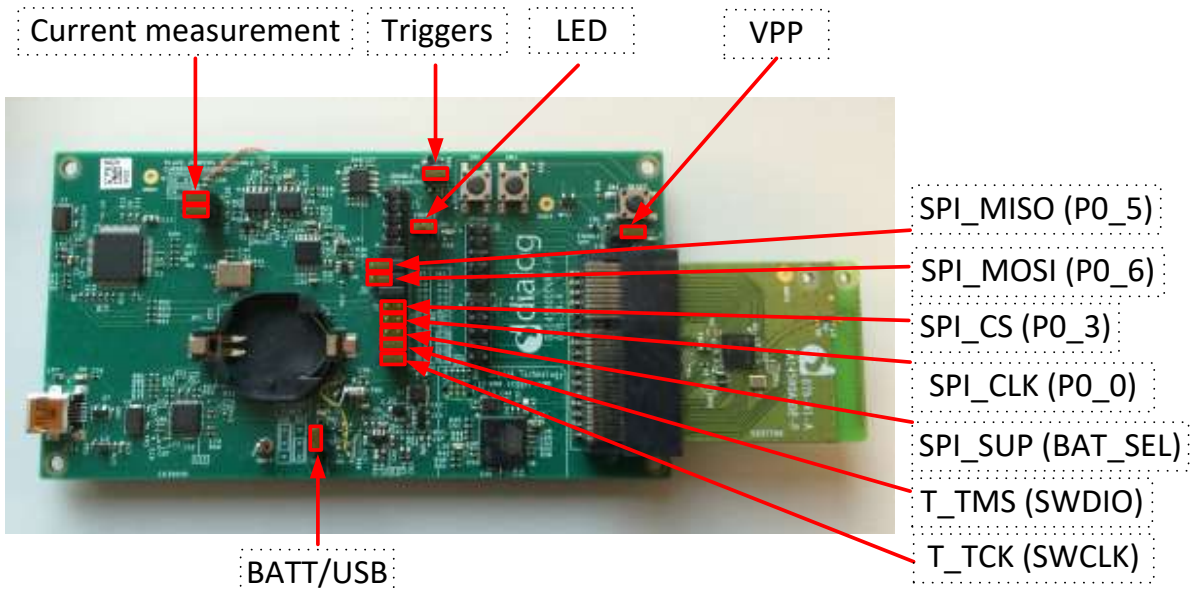


Figure 6: DA14580/581 Boot from External SPI Memory

Note 2 These functionalities are shown in detail in [Appendix A](#)

DA14580/581/583 Bluetooth Smart Development Kit – Pro

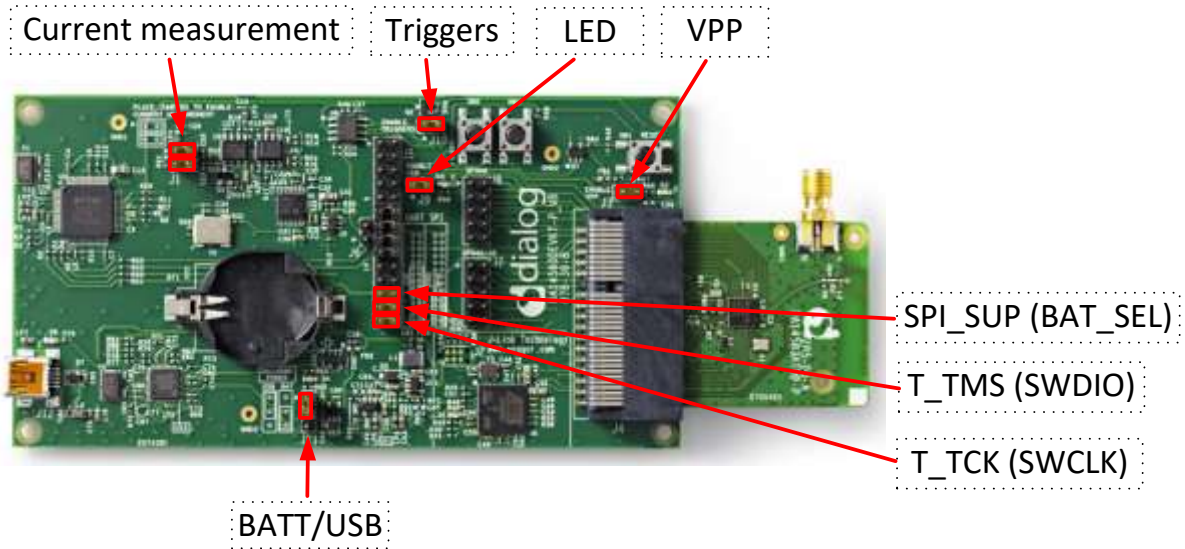


Figure 7: DA14583 Boot from Internal SPI Memory

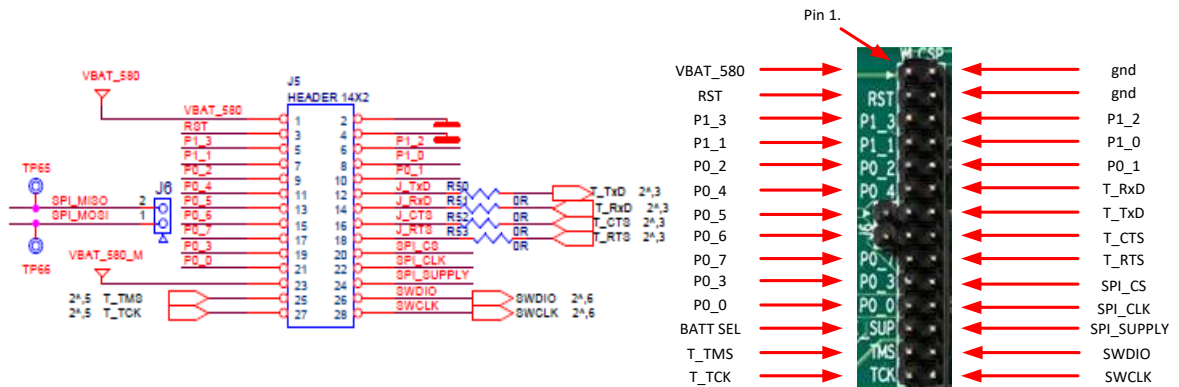


Figure 8: Layout of headers J5 and J6

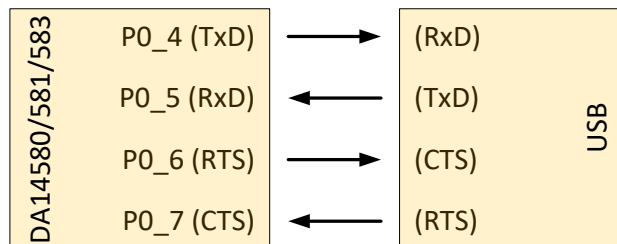


Figure 9: UART Data Direction of UART within J5

Example: when jumper J5 (27-28) is placed, connection 'T_TCK = SWCLK' is made.

In [Figure 5](#) and [Figure 6](#) the connections are added next to the arrows.

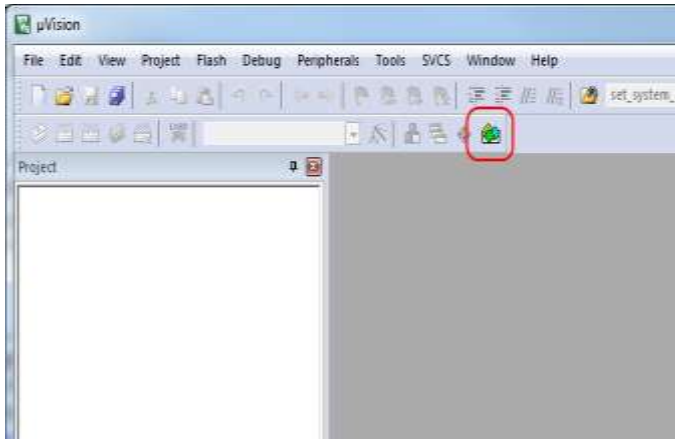


On this board only the buck mode is used. A choice can be made between 3V3 (via USB: J11 1-2) or Vdd (a coin cell: J11 2-3). No battery is needed when running via the USB-mini-cable.




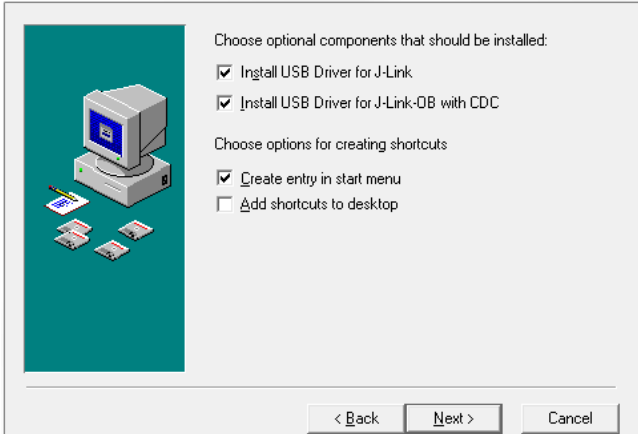
Remark: For proper battery functionality a small modification should be made. For the details see the workaround in [Appendix H](#).

4 Installation of Tools and Drivers

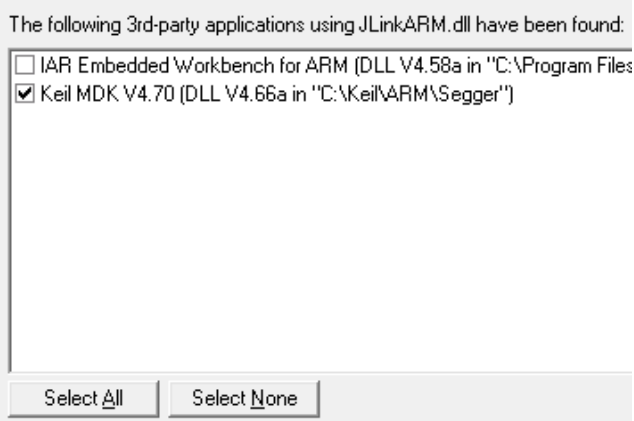
To install the Software development environment, please follow the steps as shown in [Table 6](#).

Table 6: Installation Tools and Drivers

4.1 Keil		
4.1.1	Download and install the Keil MDK-ARM uVision IDE. Both uVision ver. 4.0 and ver. 5.0 are supported, but version 5 is preferred.	<p>Keil: https://www.keil.com/download/product/</p> <p>Keil MDK-ARM Version 5 – Installation: http://www2.keil.com/mdk5/install</p> <p>Starters Guide: http://www.keil.com/uvision/ide_ov_starting.asp</p>
4.1.2	For uVision Version 5 you have to check if the generic software pack ARM::CMSIS is up to date. Press the packet installer option.	
4.1.3	You should see a list of packs as shown on the right. If you do not see this list, please click the “Packs” menu item and select the “Check for Updates” option to download an updated list. Click on the “Install” or the “Update” button to the right of “ARM::CMSIS” package if not up to date.	
4.1.4	If the installation is successful, the pack installer window should look like this.	

<p>4.1.5</p>	<p>This screen may occur, is so: Select 'yes' and in the next window Select 'MO'.</p>	
<h2>4.2 SEGGER Jlink Driver</h2>		
<p>4.2.1</p>	<p>Download and install the Jlink software & documentation pack for Windows. <i>Please note that your SEGGER Jlink serial number is required for downloading. Use the number on the sticker located on the backside of board.</i></p>	<p> J-Link software & documentation pack for Windows</p> <p>Installing the software will automatically install the J-Link USB drivers. It also software can be installed on the same PC without problems; they will co-exist.</p> <p>The package contains:</p> <ul style="list-style-type: none"> • GDB Server - Support for GDB and other debuggers using the same • J-Link Configurator - Free utility to manage a various number of J-Link • J-Link Commander - Simple command line utility, primarily for diagnosis • J-Link Remote Server - Free utility which provides the possibility to • SWO Viewer - Free tool which shows terminal output of the target • J-Mem - Memory viewer. • J-Link DLL Updater - Allows to update 3rd party applications which • Free flash programming utilities - Simple command line utilities which • USB driver (Includes driver for J-Links with CDC functionality). • Manuals: UM08001 (J-Link User Guide), UM08003 (J-Flash User Guide) (Flasher ARM User Guide). • Release notes for J-Link DLL, J-Flash, J-Link RDI DLL. • J-Flash, including sample projects for most popular eval boards. • J-Link RDI - Support for ARM RDI standard. Makes J-Link compatible <p> Download Software and documentation pack for Windows</p> <p>http://www.segger.com/jlink-software.html</p>
<p>4.2.2</p>	<p>In order to for the USB controller to be properly recognized by Windows as a J-Link device, you have to install the driver with the settings shown in the figure on the right.</p>	

DA14580/581/583 Bluetooth Smart Development Kit – Pro

4.2.3	At the end of the installation, please tick the IDE (Keil MDK Vxx) that you are using.	
<h3>4.3 FTDI Driver</h3>		
4.3.1	<p>The Development Kit uses the D2xx driver.</p> <p>For Windows, this driver is part of the Combined Driver Model (CDM) driver.</p> <p>(It is recommended that the latest driver available from the FTDI page is used.)</p> <p>Latency Timer: see Appendix F</p>	<p>USB Drivers: http://www.ftdichip.com/Drivers/D2XX.htm</p> <p>Installation Guide: http://www.ftdichip.com/Support/Documents/InstallGuides.htm</p>
<h3>4.4 Tera Term</h3>		
4.3.1	Download and install Tera Term on your PC.	<p>Tera Term: http://en.sourceforge.jp/projects/ttssh2/releases/</p>

5 Memory and Tools

The DA14580/DA14581DEVKT is equipped with: (on the chip) SRAM (50k) and OTP (32k). Mounted on the board is external SPI flash memory (2Mbit).

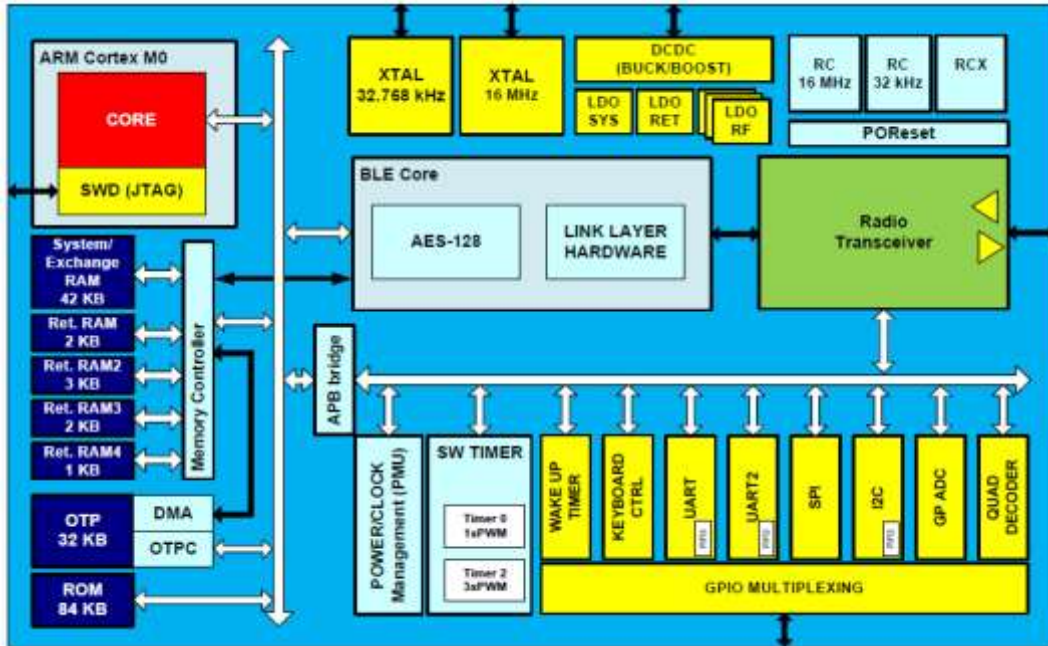


Figure 11: DA14580/581/583 Block Diagram

Software can be downloaded to:

- SRAM
 - Keil IDE
 - SmartSnippets
 - Command Line Interface (CLI)
 - Connection Manager
- OTP
 - SmartSnippets
 - CLI
- SPI (flash)
 - SmartSnippets
 - CLI

Example: loading software (hex-file) by using SmartSnippets

- PC → UART → DA14580/581/583
- PC → UART → DA14580/581/583 → SPI (flash)
- PC → UART → DA14580/581/583 → OTP

For the settings of the jumpers see [Figure 5](#).

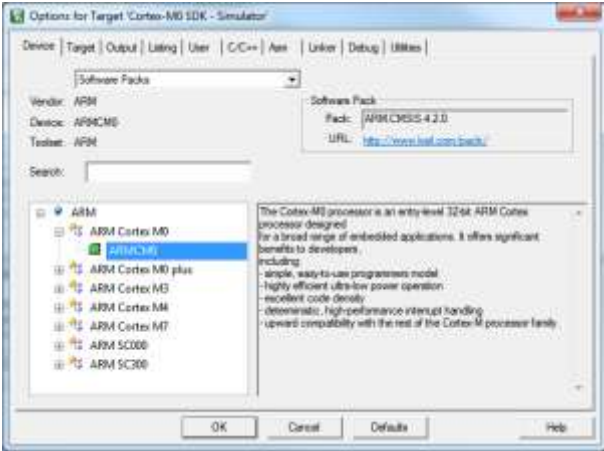


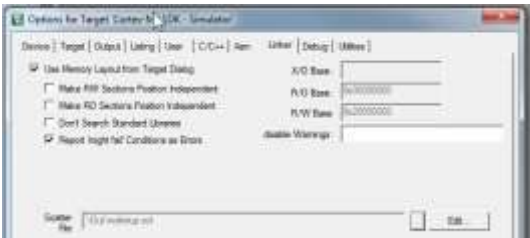
An example of the **CLI** is shown in [Appendix D](#)

6 Using the Demo Kit


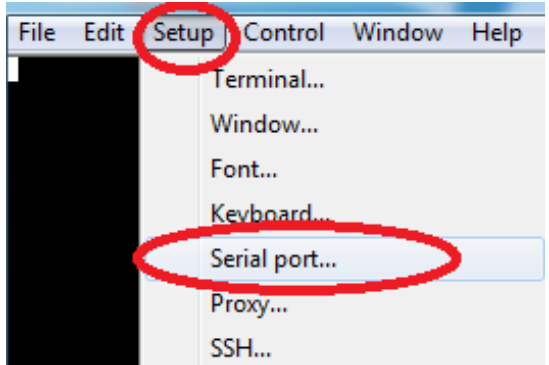
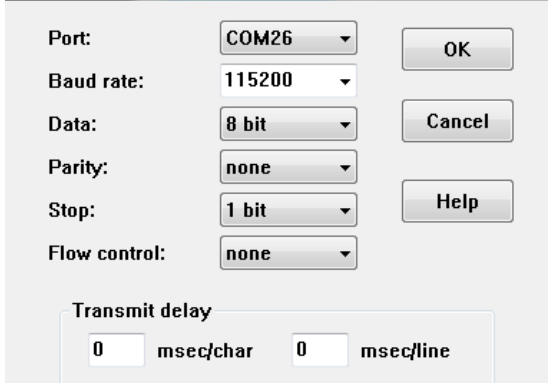
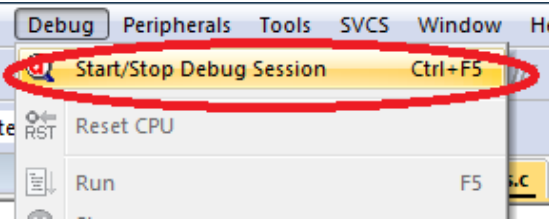
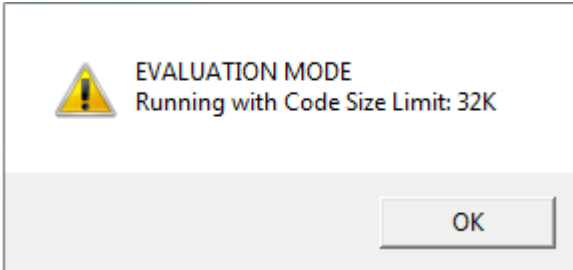
Follow the steps shown in [Table 7](#) to easily create a working demo kit.

Table 7: Run an Example on DA14580/581/583

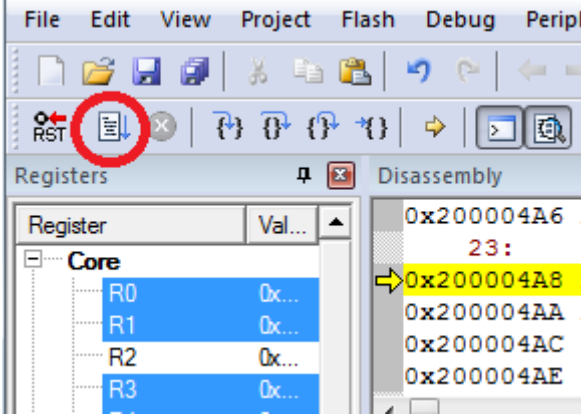
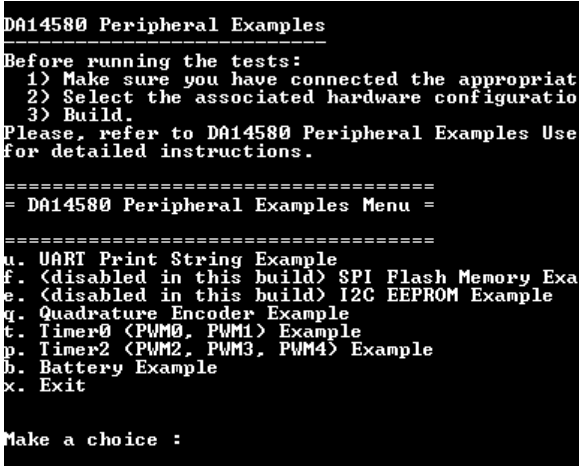
6.1 Run an Example on DA14580/581/583		
<p>6.1.1</p>	<p>After you download the SDK at http://support.dialog-semiconductor.com/software-downloads/index</p> <p>You can find a source code example in example directory called (SDK xxx): “<i>peripheral_examples\peripheral_setup</i>”.</p> <p>Double click “DA14580_peripheral_setup.uvproj”, as shown in the image to the right.</p>	
<p>6.1.2</p>	<p>The development environment should look like this when the project is opened with Keil.</p>	
<p>6.1.3</p>	<p>Click on the ‘Options for Target’ button.</p>	

<p>6.1.4</p>	<p>'Options for Target' → 'Device' -screen should look like this.</p>	
<p>6.1.5</p>	<p>'Options for Target' → 'Target' - screen looks like this. Mind the 'Memory' areas.</p>	<p>580:</p>  <p>581:</p>  <p>scatter file selection in 'Options for Target'</p> 

<p>6.1.6</p>	<p>Make sure “J-LINK/J-Trace Cortex” is selected as shown and the initialization file field is set correctly to “.\sysram.ini”.</p>	
<p>6.1.7</p>	<p>Click on the “Setting” button above to make sure the SW Device has been detected correctly.</p> <p><i>Important note: Under certain circumstances, uVision IDE may crash upon pressing the “settings” button. A workaround is found in Appendix E</i></p>	
<p>6.1.8</p>	<p>Click “OK” to save the settings.</p>	<p>All settings have been saved properly now, and you can continue to build the example.</p>
<p>6.1.9</p>	<p>Build the project by pressing “F7” key, or click the build button as shown in following picture</p>	
<p>6.1.10</p>	<p>Make sure you have a UART connection between your PC and a mother board, as shown in 6.1.5. Check the “COM” number on you PC.</p>	<p>Go to the Windows Control Panel →Administrative Tools →Computer Management → Device Manager → Ports → USB Serial Port # (connect or disconnect to see the COM port of that module)</p>
<p>6.1.11</p>	<p>Open the Tera Term serial terminal on you PC.</p>	

<p>6.1.12</p>	<p>Open Tera Term and choose a COM port, which you have found in step 3, and click OK</p>	
<p>6.1.13</p>	<p>Choose Setup->Serial port to configure the Baud rate etc.</p>	
<p>6.1.14</p>	<p>Set "Baud rate" to 115200, "Data" to 8 bit, "Parity" to None, "Stop" to 1 bit and "Flow control" to none. Click OK. Now we have a properly configured UART terminal on our PC.</p>	
<p>6.1.15</p>	<p>Go back to Keil Project. In the menu bar, select Debug->Start/Stop Debug Session.</p>	
<p>6.1.16</p>	<p>A dialog window pops up, like the one on the right. Please click "OK".</p>	

DA14580/581/583 Bluetooth Smart Development Kit – Pro

<p>6.1.17</p>	<p>Press F5 key or click execution button as shown in following picture, to start code execution.</p>	
<p>6.1.18</p>	<p>Then you can see a <i>hello</i> message on your UART terminal screen. That means you have successfully programmed and started the peripheral program on DA14580/581/583 demo board.</p> <p>The <i>peripheral_setup</i> demo consists of a small suite of tests that encompasses some of the most commonly used peripherals such as I2C EEPROM, SPI Flash, Rotary Encoder, audio buzzer etc. For more detailed info and technical details please refer to the <i>UM-B-005</i>: <i>DA14580/581/583Peripheral Examples</i> as well as the source code of the <i>peripheral_setup</i> demo.</p> <p>Remark: this software is originally written for the 'SmartBond™ DA14580/581/583-- Development Kit – Expert'.</p> <p>See: http://support.dialog-semiconductor.com/resources</p>	

Appendix A Layout

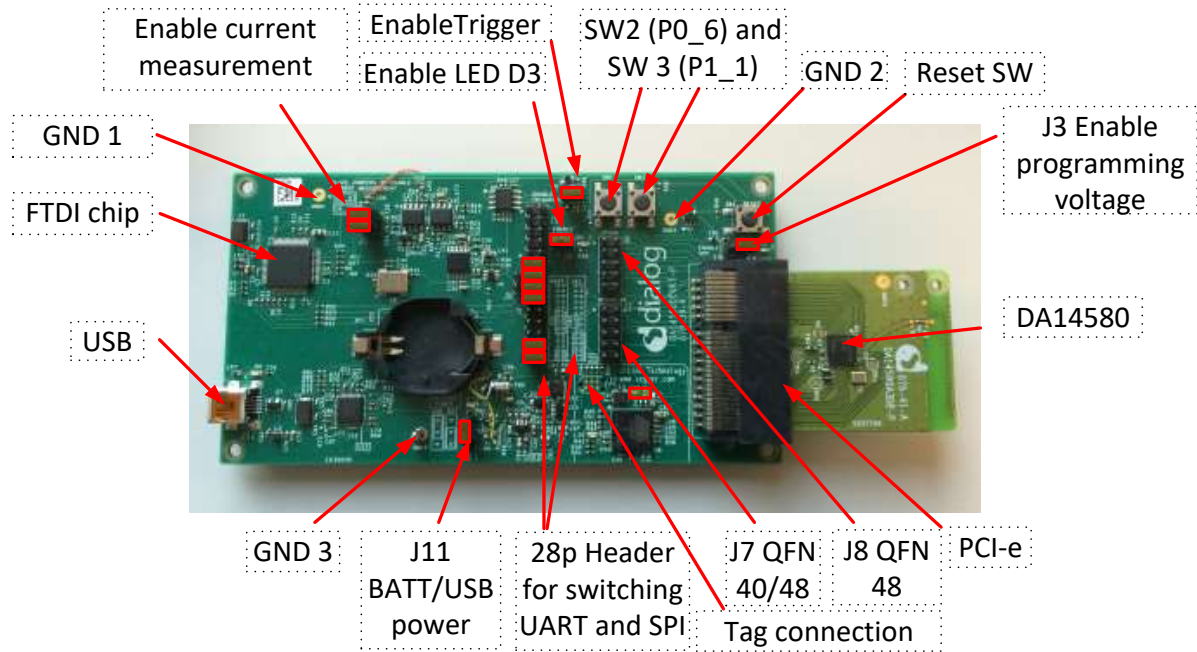


Figure 12: Board Layout

Appendix B Connections of J7 and J8

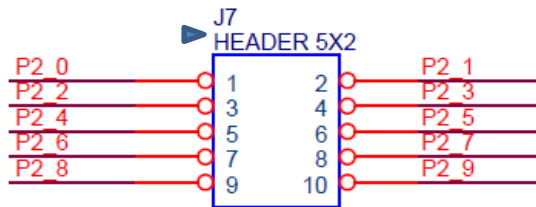


Figure 13: IO breakout available on QFN40 and QFN48

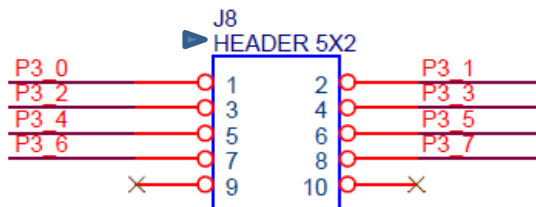


Figure 14: IO Breakout Available on QFN48

DA14580/581/583 Bluetooth Smart Development Kit – Pro

C.2 SPI and Serial

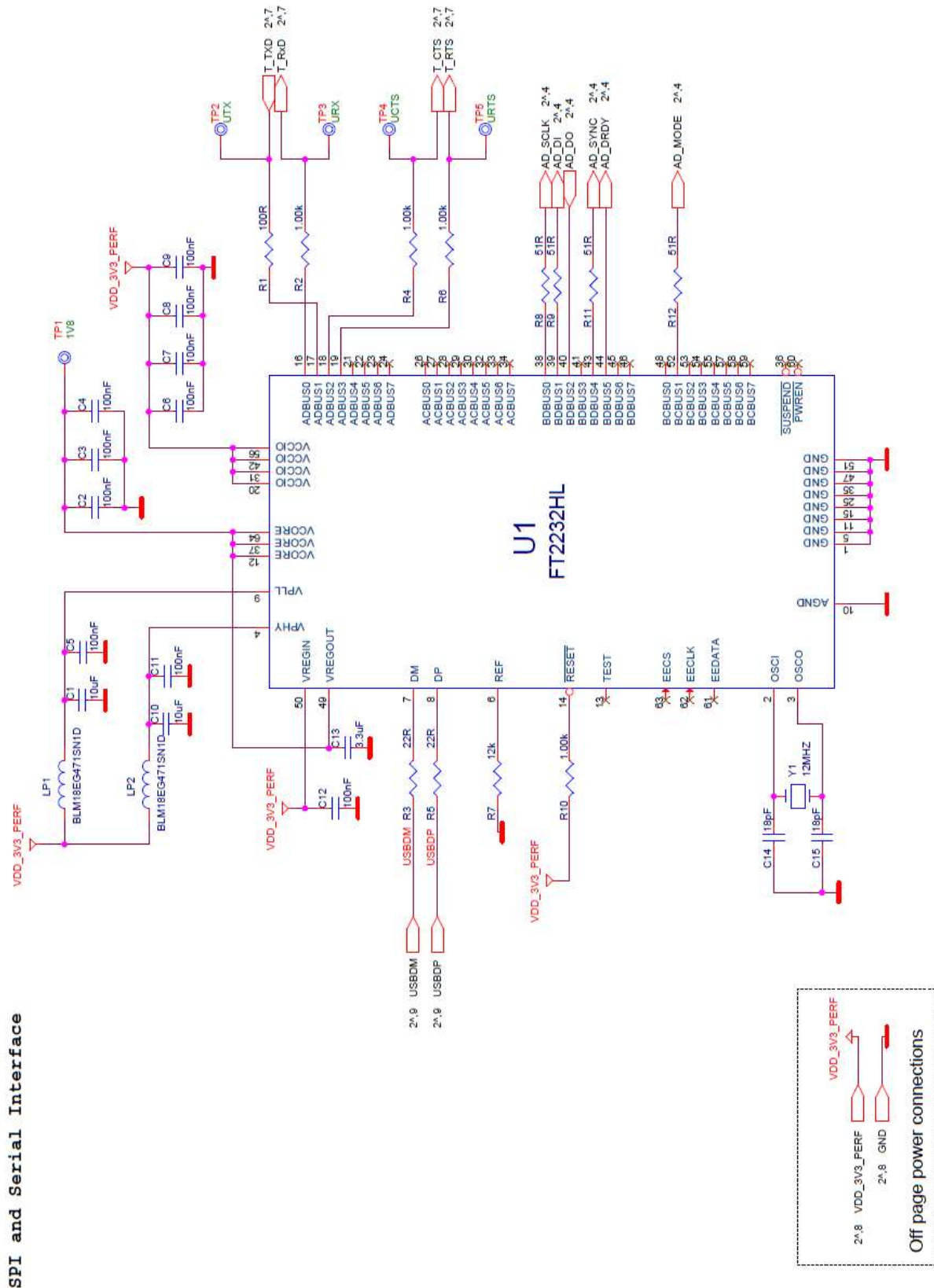


Figure 16: SPI and Serial Schematic of Motherboard

C.3 Current Measurement

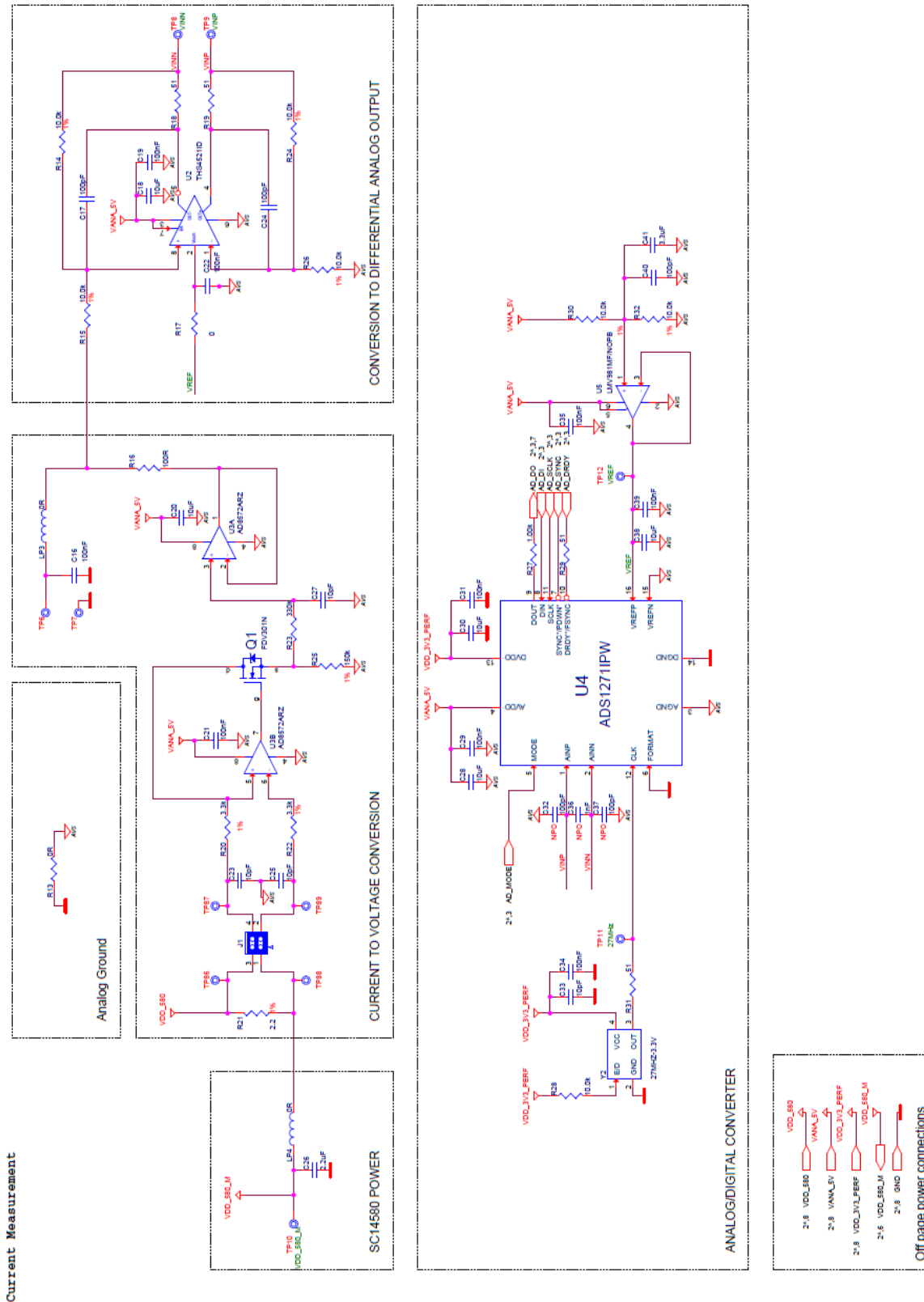


Figure 17: Current Measurement Schematic of Motherboard

C.6 Power Supplies

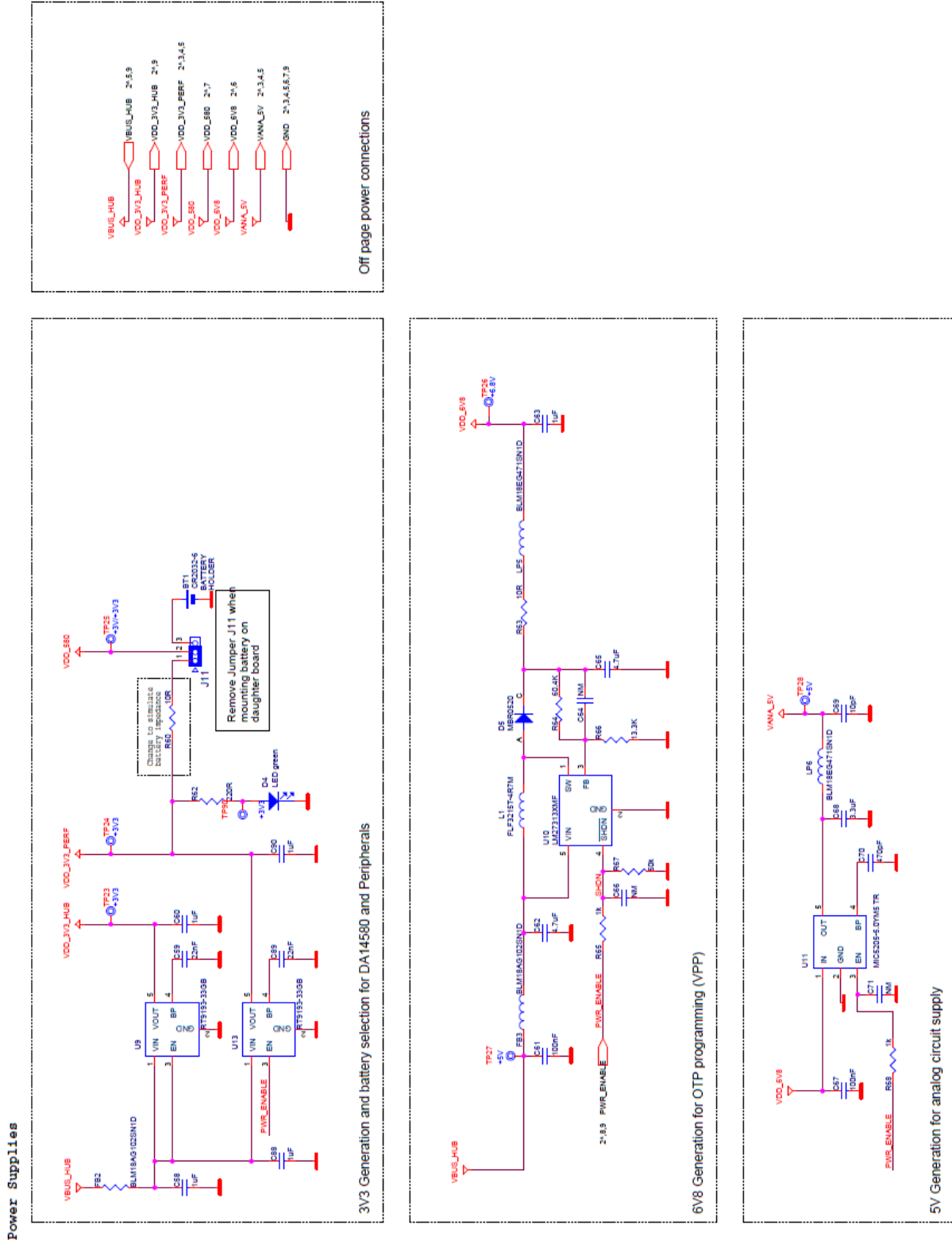


Figure 20: Power Supplies Schematics of Motherboard

C.8 DA14580 WLCSP Daughter board

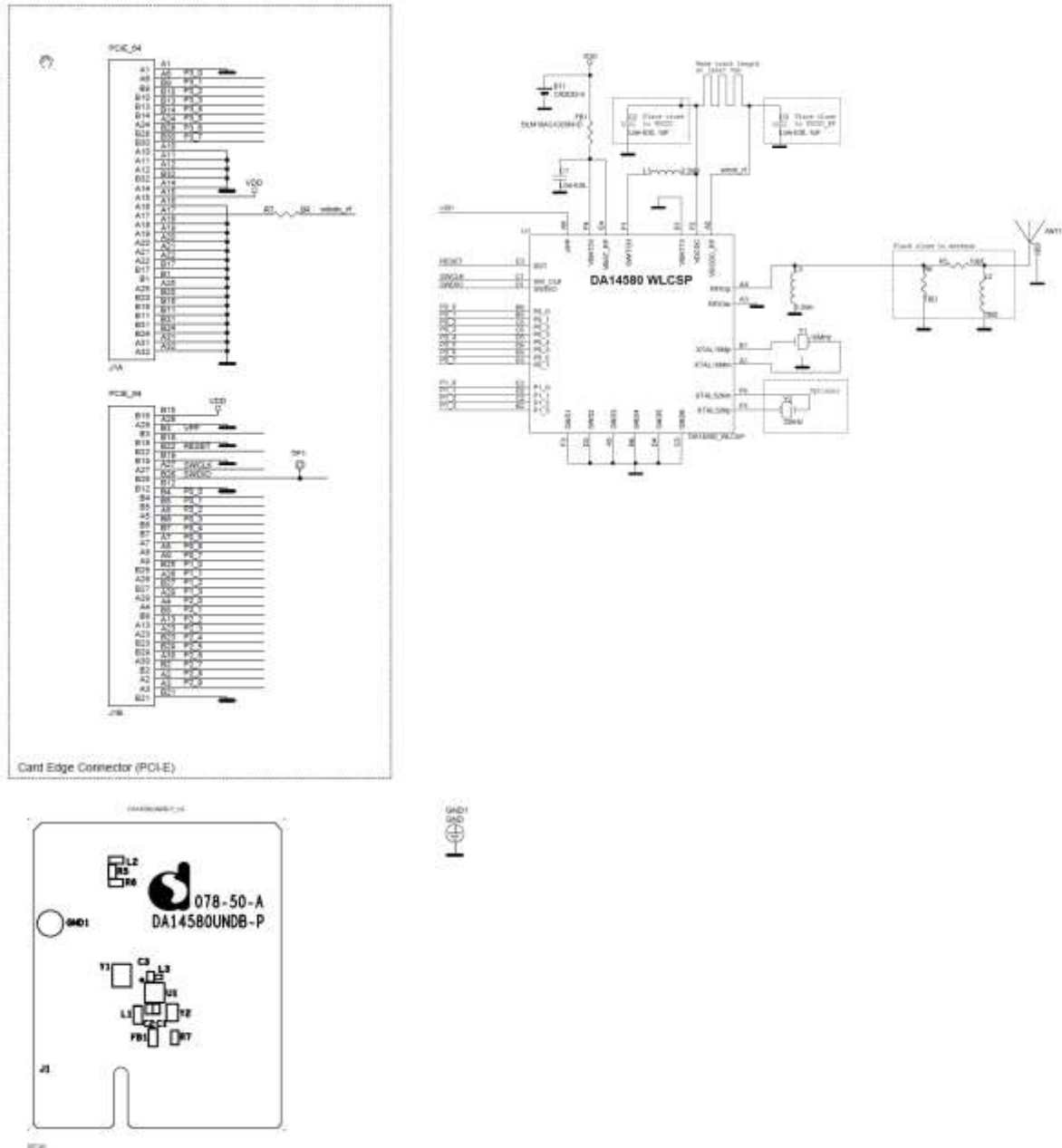


Figure 22: DA14580 WLCSP Daughter Board

UM-B-034

DA14580/581/583 Bluetooth Smart Development Kit – Pro

C.9 DA14580 QFN40 Daughterboard

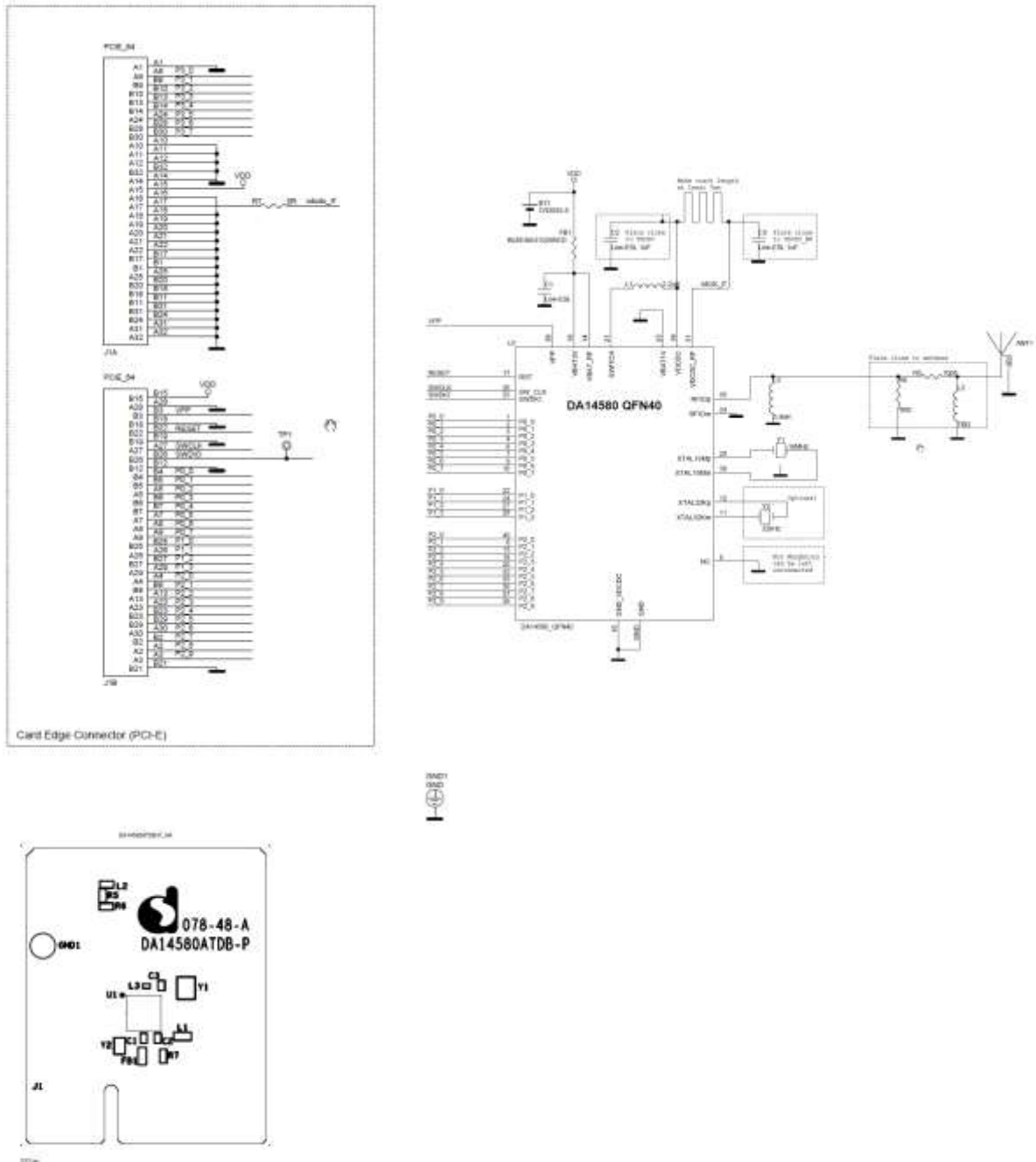


Figure 23: DA14580 QFN40 Daughterboard

UM-B-034

DA14580/581/583 Bluetooth Smart Development Kit – Pro

C.11 DA14581 WLCSP Daughterboard

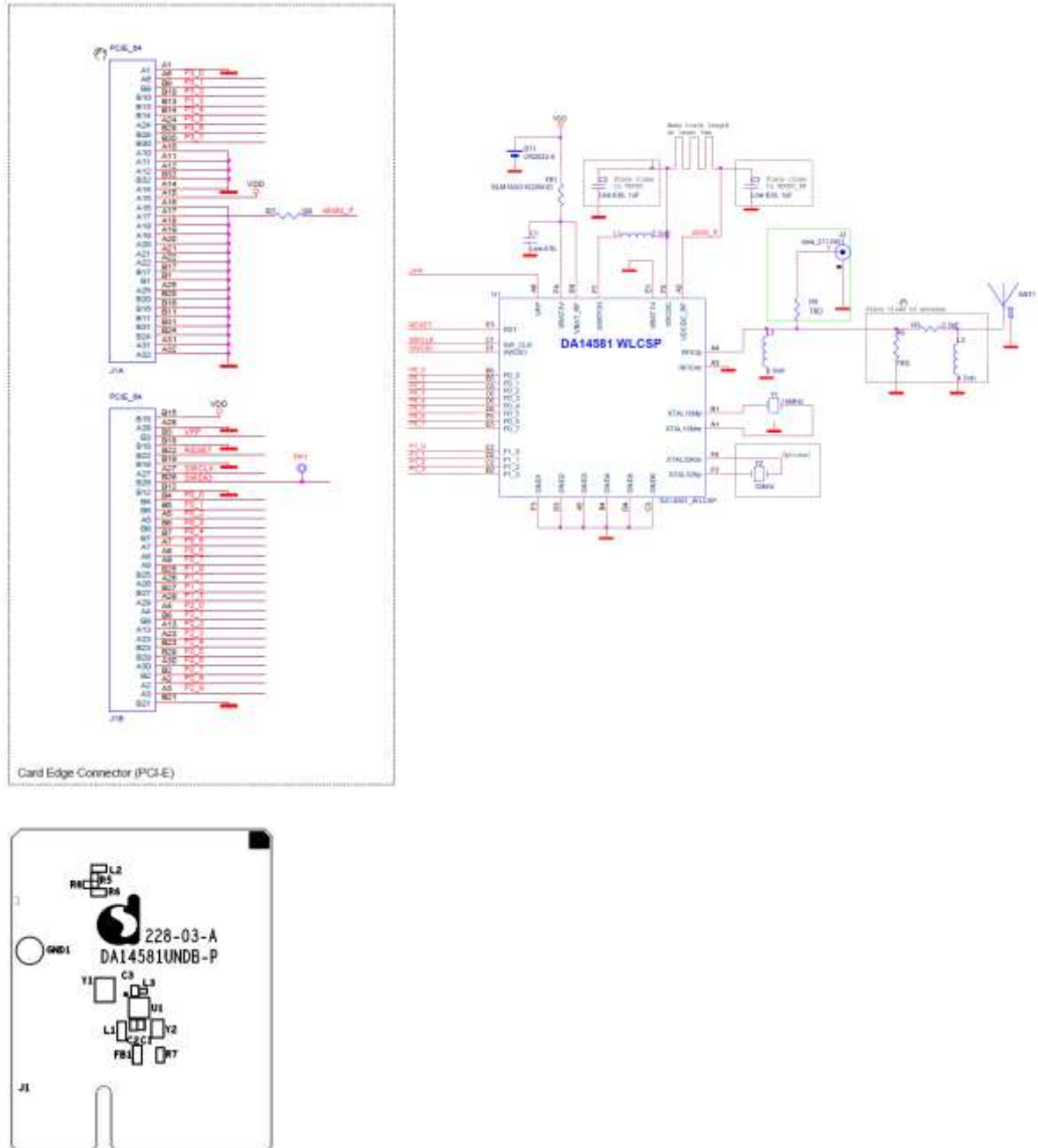


Figure 25: DA14581 WLCSP Daughterboard

UM-B-034

DA14580/581/583 Bluetooth Smart Development Kit – Pro

C.13 DA14583 QFN40 Daughterboard

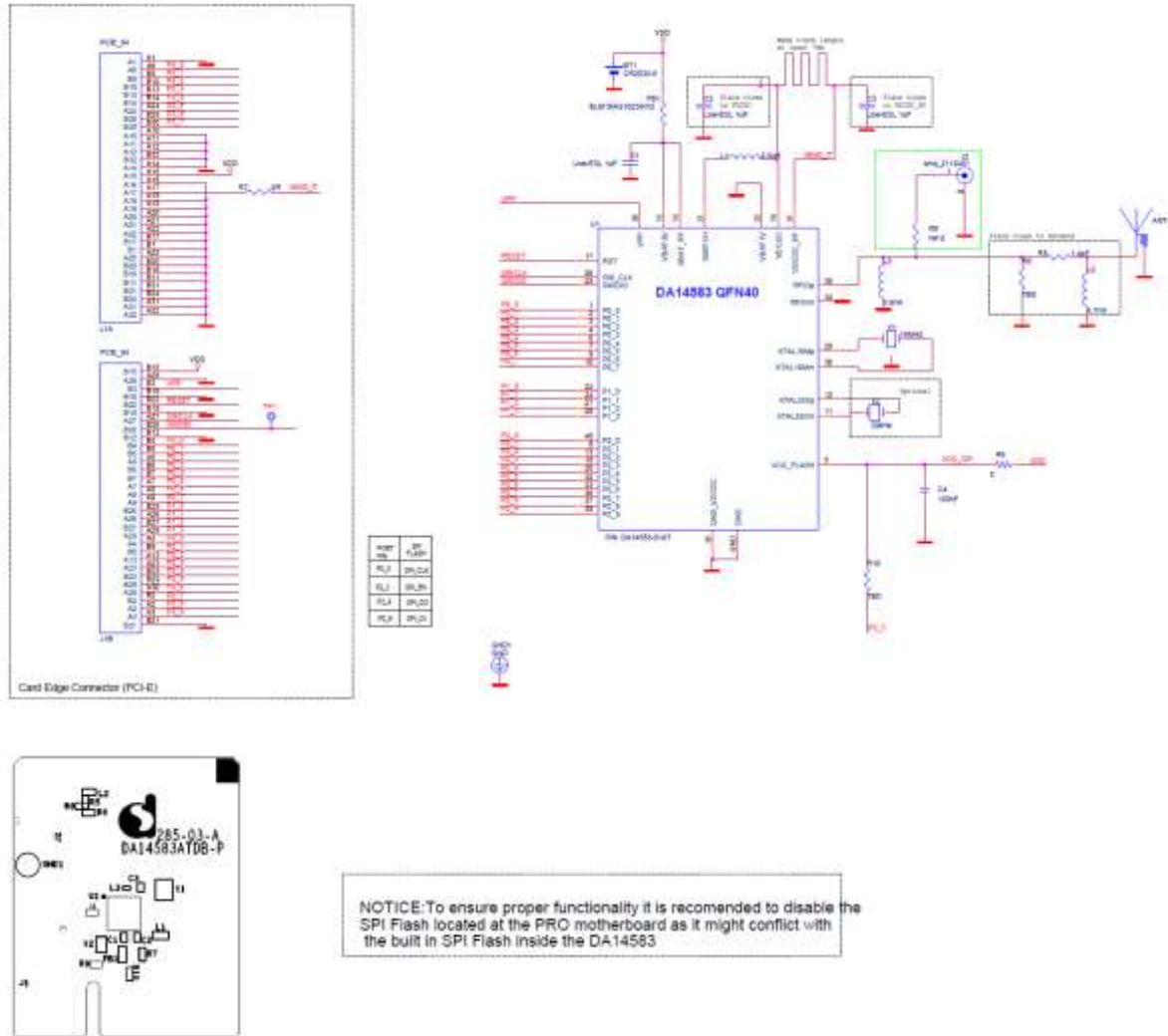


Figure 27: DA14583 QFN40 Daughterboard

Appendix D Using the SmartSnippets CLI

All the information/syntaxes about the CLI can be found from the **HELP** tab in the SmartSnippets GUI or by written **Smartsnippets –help** in the CLI.

In this example, it is supposed that the SPI memory is using P0_0 as SCK, P0_3 as CS, P0_5 as MISO and P0_6 as MOSI.

First of all, the CLI can send the commands either via UART or JTAG according to the binary file which has to be loaded.

On the one hand, if the commands are going to be sent via UART, the following binary file which can be found from the resources folder of SmartSnippets has to be downloaded into the DA14580/DA14581:

- *flash_programmer.bin*

On the other hand, if the commands are going to be sent via JTAG, the following binary file which can be found from the resources folder of SmartSnippets has to be downloaded into the DA14580/DA14581:

- *jtag_programmer.bin*

Secondly, open the CLI by pushing the Shift button and right click on the **'bin'** folder of the SmartSnippet and select **'Open command window here'** as follow:

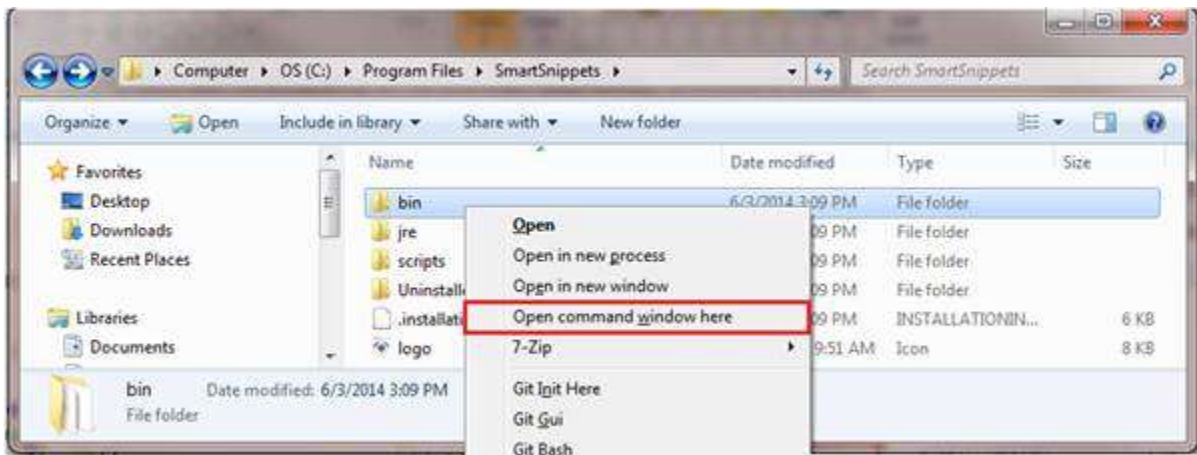


Figure 28: Open the CLI of SmartSnippets

Finally, in order to write a value 0x1347 (example of a bluetooth device address) at the address 0x93 for instance, the following command line can be written:

```
SmartSnippets.exe -type spi -chip DA14580/DA14581-01 -jtag 228202458 -cmd write_field -offset 0x93 -data 1347 -firmware "D:\SmartSnippets\resources\jtag_programmer.bin"
```

The answers should be as shown below:

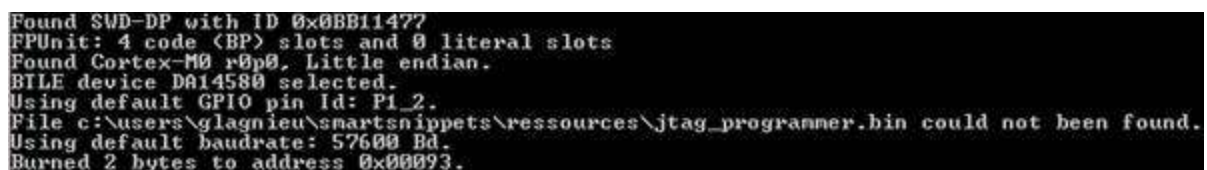


Figure 29: Smart Snippet CMD Window

Appendix E Opening Your Project for the First Time

Issue description

When, on a Keil uVision project, some entries in file .uvopt is missing or the file is missing, then, when the user clicks on the button 'settings' (options{debug tag}) with the{J-LINK/J-TRACE Cortex} selected, uVision crashes.

Possible causes

Some important information concerning the j-link driver is missing. Calling the driver's dll probably causes the crash.

Versions of Keil uVision found to be affected

At least versions 5.11.1.0 and 5.10.0.2 are affected.

Under which circumstances user will encounter this error

When a local GIT repository is first created, this file (.uvopt) does not exist, since it is not included in the remote repository. When the user opens the project for the first time, this file is created, but some keys/values are missing.

A proposed solution:

- 1) Ensure that the .uvopt file does not exist in the folder of your project. If it exists and crash has been identified to happen, delete the .uvopt file.
- 2) Open the Keil project and close it. The .uvopt file is created automatically in the project folder (where the .uvproj is located).
- 3) Open the .uvopt file, using your favourite text editor.
- 4) Under the key <TargetOption> add the following lines:


```
<TargetDriverDllRegistry>
<SetRegEntry>
<Number>0</Number>
<Key>JL2CM3</Key>
<Name>-U228202424 -O78 -S0 -A0 -C0 -JU1 -JI127.0.0.1 -JP0 -RST0 -N00 ("ARM
CoreSight SW-DP") -D00 (0BB11477) -L00 (0) -TO18 -TC10000000 -TP21 -TDS8007 -TDT0 -
TDC1F -TIEFFFFFFFF -TIP8 -TB1 -TFE0 -FO7 -FD20000000 -FC800 -FN0</Name>
</SetRegEntry>
</TargetDriverDllRegistry>
```
- 5) Save the .uvopt file and close the text editor.
- 6) Open the Keil project in uVision.
- 7) Click on *Project→Options for Project 'XXX'*.
- 8) On the 'Debug' Tab, select J-Link / J-TRACE Cortex debugger and click on the 'Settings' button for the debugger (not the simulator). This is the instance where the crash would happen.
- 9) The 'Cortex JLink/JTrace Target Driver Setup' Dialog opens. *Select your debugger as normally.*
- 10) Close the dialog windows clicking ok.
- 11) *Now, normal operation of j-link debugger is resumed. After you have finished your work, close the Keil uVision IDE to allow for updates to the .uvopt file to be saved.*

Appendix F Latency Timer of 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 15ms to <10ms.

To change the Latency Timer:

Device Manager → COM port → Right click on the COM port chosen → Properties → Port Settings → Advanced → Latency Timer: set it <10ms.

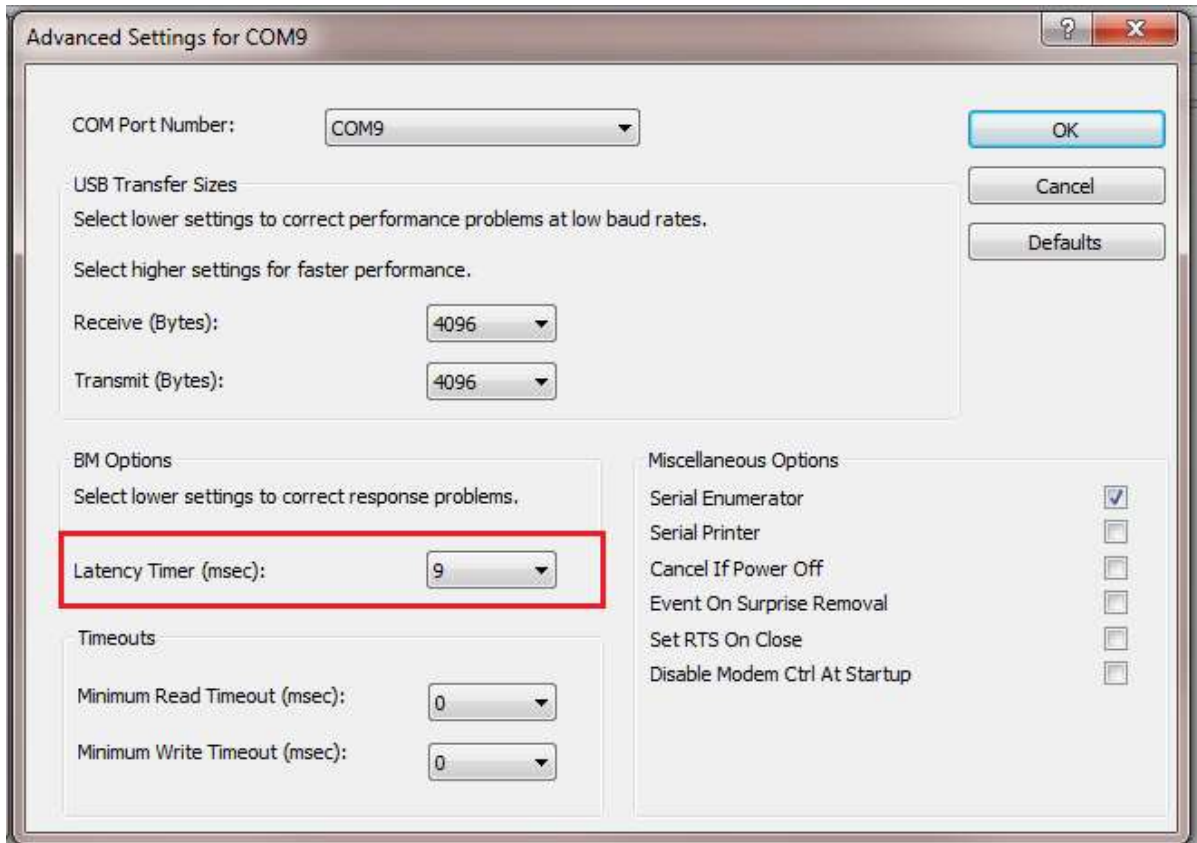


Figure 30: FTDI Latency Timer

DA14580/581/583 Bluetooth Smart Development Kit – Pro

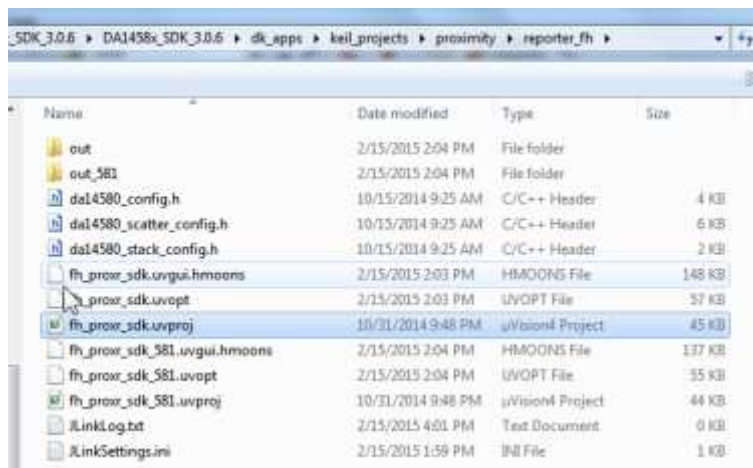
Appendix G RF-Testing DEVKT - Pro

Follow steps to start example from SDK:

- Step 0. connect DEVKT- Pro to USB-port.
- Step 1a. for DA14580: start KEIL via double click on project name:
... dk_apps\keil_projects\proximity\prox_reporter\prox_reporter.uvproj

or

- step 1b. for DA14581: start KEIL via double click on SDK 3.0.6. object name:
... DA1458x_SDK_3.0.6\DA1458x_SDK_3.0.6\dk_apps\keil_projects\proximity\prox_reporter\prox_reporter_581.uvproj



- Step 2. #define HW_CONFIG_PRO_DK // Pro DK
see: DA14580_config.h

```

99  /*
100  * HW configuration
101  */
102  // #define HW_CONFIG_BASIC_DK // Basic DK (DA14580DEVKT-B)
103  #define HW_CONFIG_PRO_DK // Pro DK (DA14580DEVKT-P)
104
105  #endif // DA14580_CONFIG_H
106

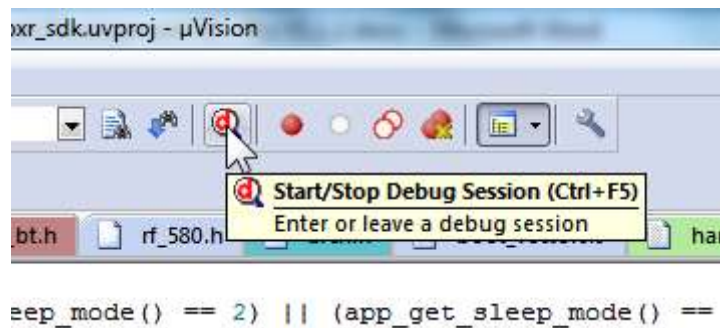
```

- Step 3. KEIL is started and press F7 for 'Building' the software. When the build-result is 'no errors', then got the next step.

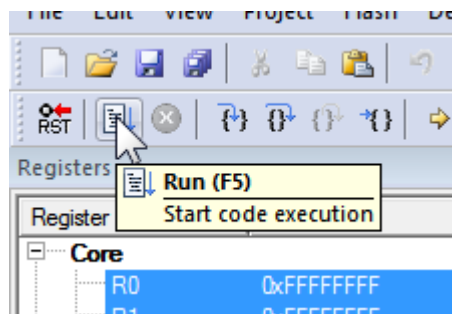
DA14580/581/583 Bluetooth Smart Development Kit – Pro



- [Step 4.](#) Start Debug Session (Ctrl + F5)

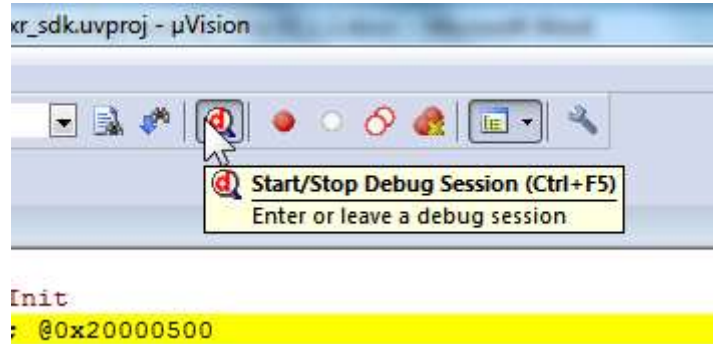


- [Step 5.](#) Run the software by pressing 'F5'



DA14580/581/583 Bluetooth Smart Development Kit – Pro

- **Step 6.** Stop Debug Session by pressing 'Ctrl + F5'
Software will start running now!



- **Step 7.** Check via 'Bluetooth scanning software' whether the RF-part of the DEVKT - Pro is working.
This software is available for iPhone and Android phone as an App.

Appendix H 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, then T_RESET becomes low. After inverter U7 RST is high. In this case the board will not operate.

Workaround is the dismounting of R84.

In [Figure 31](#) the Reset circuitry is shown.

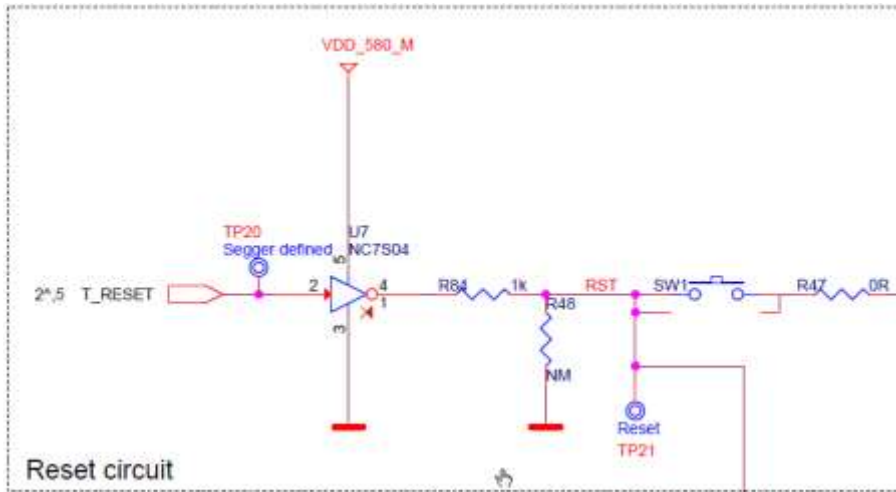


Figure 31: Reset Circuitry. R84 to be Dismounted

In a DEVKT – Pro update the hardware problem will be solved.

In [Figure 32](#) the location of R84 is shown.

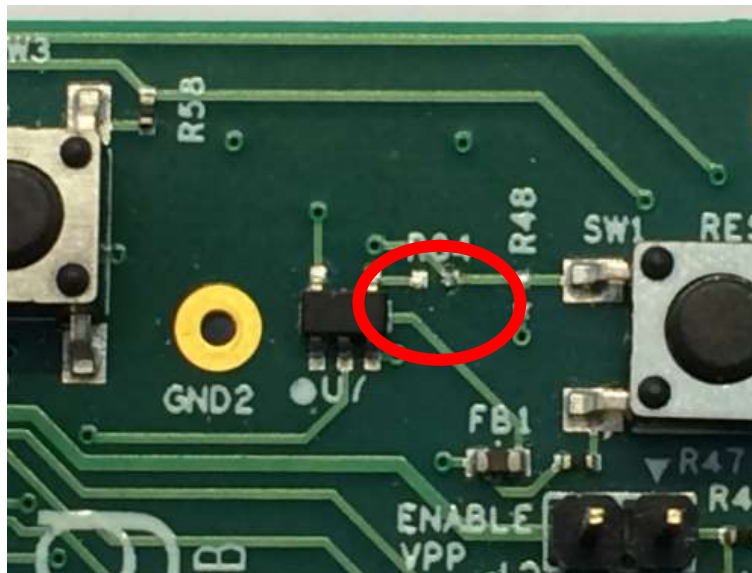


Figure 32: Location of R84

DA14580/581/583 Bluetooth Smart Development Kit – Pro

7 Web-Link

All support info:

- <http://support.dialog-semiconductor.com>

8 Revision History

Revision	Date	Description
1.5	22-Dec-2021	Updated logo, disclaimer, copyright
1.4	20-Oct-2015	Minor updates in Keil installation
1.3	03-Jun-2015	Updated for SDK 3.0.10 added DA14583-pictures
1.2	18-05-2015	Updated for SDK 3.0.8 added Appendix I Battery connection workaround
1.1	25-02-2015	Some textual changes Added FTDI-links Added changes to convert to DA14580/581/583
1.0	11-02-2015	Initial version for DA14580

DA14580/581/583 Bluetooth Smart Development Kit – Pro**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 complies to European Directive 2001/95/EC and from 2 January 2013 onwards to European Directive 2011/65/EU concerning Restriction of Hazardous Substances (RoHS/RoHS2).
Dialog Semiconductor's statement on RoHS can be found on the customer portal <https://support.diasemi.com/>. RoHS certificates from our suppliers are available on request.