FM0+ S6E1B-Series

Starter Kit Guide

Doc. # 002-10654 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Copyrights

Contents

1. Introduction

Thank you for your interest in the FM0-100L-S6E1B8 FM0+ S6E1B-Series Starter Kit .The FM0+ S6E1B-Series Starter Kit is a low-cost hardware platform to enable design and debug of the S6E1B8 microcontroller. The S6E1B series is designed for meters, sensor networks and handheld devices. There are multiple series of device families in this portfolio. S6E1A-Series, S6E1B-Series and S6E1C-Series are few of the prominent series of device families. This kit uses a device from the S6E1B-Series.

Devices in the S6E1B-Series are highly integrated 32-bit microcontrollers with high performance and competitive cost. This series is based on the ARM® Cortex®-M0+ 40 MHz processor with on-chip 560KB flash memory, 64KB SRAM and 82 GPIOs. This series has peripherals such as motor control timers, A/D converters, LCD controller and communications interfaces like USB, UART, CSIO (SPI), I2C and LIN.

The FM0+ S6E1B-Series Starter Kit offers footprint compatibility with Arduino™ shields, which provides many options for application development. This board features a rich set of peripherals such as USB host, USB device, stereo codec, potentiometer and accelerometer.

1.1 Kit Contents

The FM0+ S6E1B-Series Starter Kit contains the following, as shown in Figure 1-1.

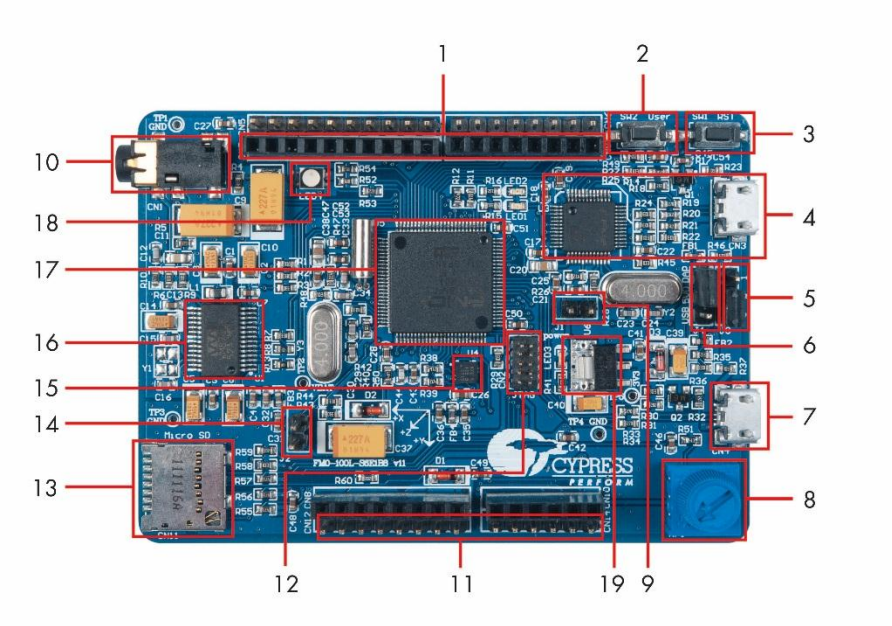
- FM0+ S6E1B-Series Starter board
- USB Standard-A to Micro-B cable
- Quick Start Guide

Figure 1-1: Kit Contents

Inspect the contents of the kit; if you find any part missing, contact your nearest Cypress sales office for help: www.cypress.com/support.

1.2 Board Details

Figure 1-2: FM0+ S6E1B-Series Starter Kit board Markup



1. Arduino interface (CN7-CN10)
2. User button (SW2)
3. Reset button (SW1)
4. MB9AF312K Programmer and debugger (CMSIS-DAP)
5. Serial programming mode select (J3)
6. Power supply select (J4)
7. USB device connector (CN4)
8. Potentiometer
9. Programming mode jumper of MB9AF312K (J1)
10. Headphone and microphone jack (CN1)
11. Additional GPIO headers (CN5, CN6, CN12, CN14)
12. 10-pin JTAG connector (CN2)
13. Micro SD card connector (CN11)
14. Programming mode jumper of FM0+ (J2)
15. Accelerometer
16. Stereo codec
17. Cypress FM0+ S6E1B86F0A MCU
18. RGB LED (LED4)
19. 3.3V Voltage Regulator

1.3 Jumpers and Connectors

Table 1-1: Jumper Description

Jumper	Function	Setting	Default
J1	Sets the programming mode pin (MD0) of MB9AF312K (CMSIS-DAP)	Open: run mode	Open
		Closed: serial programming mode	
J2	Sets the programming mode pin (MD0) of S6E1B8	Open: run mode	Open
		Closed: serial programming mode	
J3	Serial programming mode select of S6E1B8	Pin 2 to Pin 1: UART programming mode	Pin 2 to Pin 3
		Pin 2 to Pin 3: USB programming mode	
J4	Power supply select	Pin 2 to Pin 1: power from USB port of CMSIS-DAP (CN3)	Pin 2 to Pin 1
		Pin 2 to Pin 3: power from USB port of FM0+ (CN4)	

Table 1-2: Connector Description

Number	Connector	Description
1	CN1	3.5mm headphone and microphone jack
2	CN2	10-pin JTAG interface
3	CN3	USB port of CMSIS-DAP (MB9AF312K)
4	CN4	USB port of FM0+ MCU (S6E1B8)
5	CN5,CN6,CN12,CN14	Additional GPIO headers
6	CN7,CN8,CN9,CN10	Arduino compatible headers
7	CN11	Micro SD card slot

1.4 Getting Started

This guide will help you get started with the FM0+ S6E1B-Series Starter Kit:

- The Installation and Test Operation chapter describes the installation of the kit, and the test procedures for testing the board.
- The Hardware chapter describes the major features of the FM0+ S6E1B-Series Starter Kit and functionalities such as CMSIS-DAP debugger, micro SD card, USB, stereo codec, potentiometer and sensor.
- The Software Development chapter describes how to open and run an example project in the IAR Embedded Workbench or Keil µVision IDE; it also describes how to use the example projects and how to program the devices using the Flash USB DIRECT Programmer.
- The Appendix provides the kit schematics, and the bill of materials (BOM).

1.5 Additional Learning Resources

Cypress provides a wealth of data at www.cypress.com to help you to select the right MCU device for your design, and to help you to quickly and effectively integrate the device into your design.

Visit www.cypress.com/FM0-100L-S6E1B8 for the following documents on S6E1B-Series MCU:

- **Overview**: S6E1B8-Series fact sheet
- **Device Selector** : Microcontroller Select Guide
- **Datasheets**: S6E1B8-Series datasheet and handling precautions.
- **FM0+ Peripheral Manual**: Main Section, Communication Macro section, Analog Macro section, and Timer section

1.6 Technical Support

For assistance, visit Cypress Support or contact customer support at +1(800) 541-4736 Ext. 2 (in the USA) or +1(408) 943-2600 Ext. 2 (International).

1.7 Acronyms

Table 1-3. Acronyms Used in this Document

Acronym	Description
ADC	Analog-to-Digital Converter
CMSIS-DAP	Debug Access Port
ICE	In-Circuit Emulator
IDE	Integrated Development Environment
I2C	Inter-Integrated Circuit
I2S	Inter-IC Sound
INT	Interrupt
JTAG	Joint Test Action Group
LDO	Low Drop Out (voltage regulator)
LED	Light-Emitting Diode
LIN	Local Interconnect Network
MFS	Multi Function Serial
MFT	Multi Function Timer
PDL	Peripheral Driver Library
PPG	Programmable Pulse Generator
RGB	Red Green Blue
SPI	Serial Peripheral Interface
SWD	Serial Wire Debug
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus

2. Installation and Test Operation

This chapter describes the steps to install the software tools and drivers on a PC for using the FM0+ S6E1B-Series Starter Kit. After a successful installation, the user can run the test code that is pre-programmed on the device.

2.1 Install Software

Follow the steps below to install the FM0+ S6E1B-Series Starter Kit software:

1. Download the FM0+ S6E1B-Series Starter Kit installer from the webpage:

   www.cypress.com/FM0-100L-S6E1B8. The Kit software is available for download in three formats.

   - **FM0+ S6E1B-Series Starter Kit Complete Setup**: This installation package contains the files related to the kit, including the Documentation, Hardware, Firmware, Software tools and drivers. However, it does not include the Windows Installer or Microsoft .NET framework packages. If these packages are not on your computer, the installer directs you to download and install them from the Internet.
   - **FM0+ S6E1B-Series Starter Kit Only Package:** This executable file installs only the kit contents, which include kit code examples, hardware files, and user documents. This package can be used if all the software prerequisites (listed in step 7) are installed on your PC.
   - **FM0+ S6E1B-Series Starter Kit DVD ISO**: This file is a complete package, stored in a DVD-ROM image format, which you can use to create a DVD or extract using an ISO extraction program such as WinZip® or WinRAR. The file can also be mounted like a virtual CD/DVD using virtual drive programs such as Virtual CloneDrive and MagicISO. This file includes all the required software, utilities, drivers, hardware files, and user documents.

2. If you have downloaded the ISO file, mount it like a virtual drive. Extract the ISO contents if you do not have a virtual drive to mount. Double-click **cyautorun.exe** in the root directory of the extracted content or the mounted ISO if “Autorun from CD/DVD” is not enabled on the PC. The installation window will appear automatically.

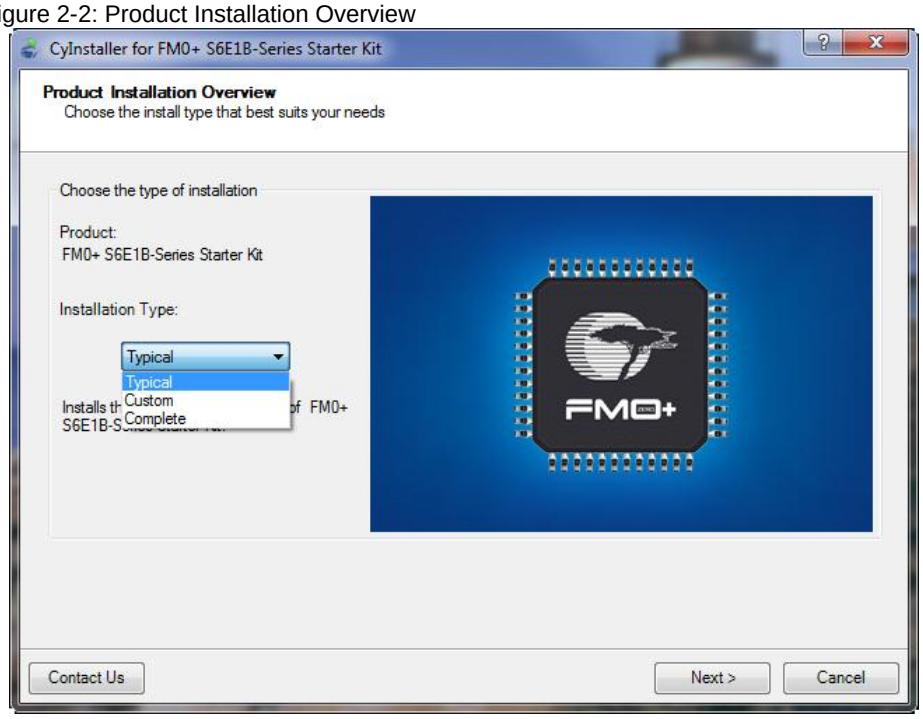
   Note: If you are using the “Kit Complete Setup” or “Kit Only Package”. Then go to step 4 for installation.

3. Click **Install FM0-100L-S6E1B8** to start the kit installation, shown as Figure 2-1.

Figure 2-1: Kit Installation Window

4. Select the folder in which you want to install this package or use the default folder and click **Next**.
5. Choose the **Typical**, **Custom**, or **Complete** installation type (select **'Typical'** if you do not know which one to select) in the Product Installation Overview window, as shown in Figure 2-2. Click **Next** after you select the installation type.

Figure 2-2: Product Installation Overview

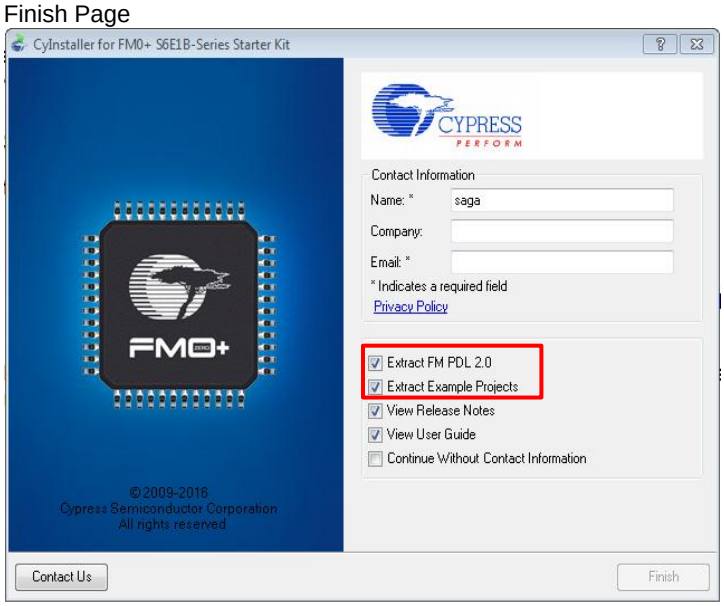


6. Read the License agreement and select **I accept the terms in the license agreement** to continue with installation.

7. When you click **Next**, the FM0+ S6E1B-Series Starter Kit installer automatically installs the required software, if it is not present on your PC. Following are the required software and drivers:
   - FM Universal Peripheral Driver Library (PDL)
   - Serial Port Viewer
   - FLASH USB DIRECT Programmer
   - FLASH MCU Programmer
   - CMSIS-DAP driver
8. When the installation begins, a list of packages appears on the installation page. A green check mark appears next to each package after successful installation.
9. If you are an un-registered user either enter your contact information or select the check box **Continue without Contact Information**. If you are a registered user, then the installation procedure will not request you to enter the contact information. Click **Finish** to complete the kit installation.

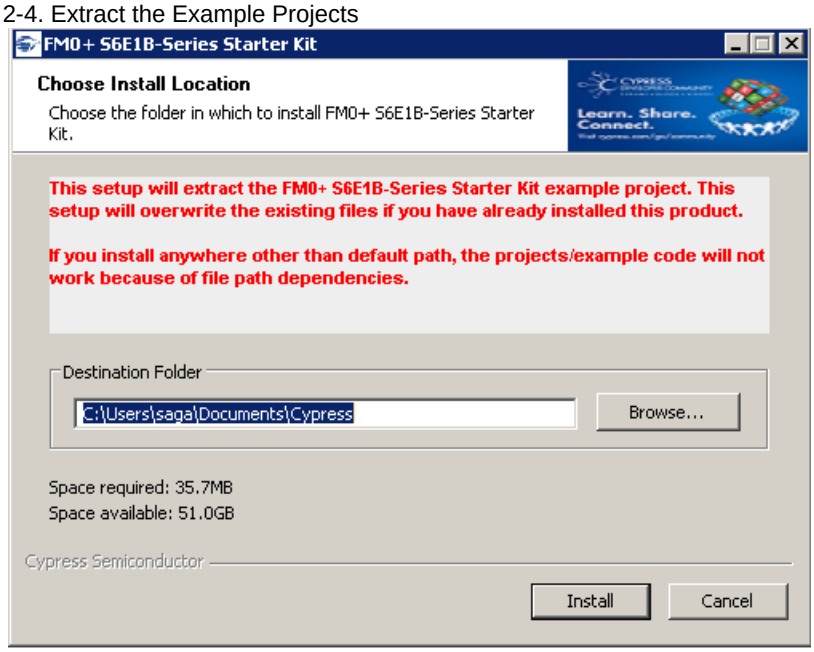
   Note: Be sure to select the check boxes Extract Example Projects and Extract FM PDL 2.0.

Figure 2-3. Finish Page



10. Click **Install**.

Figure 2-4. Extract the Example Projects



11. Click **Close** to finish the extraction.

After the installation is complete, the kit documentations and hardware files are available at the following location:

```
Windows 7 (64-bit): C:\Program Files (x86)\Cypress
\FM0+ S6E1B-Series Starter Kit
Windows 7 (32-bit): C:\Program Files\Cypress
\FM0+ S6E1B-Series Starter Kit
```

The Peripheral Driver Library (PDL) will be extracted to this default directory:

```
C:\Users\<User Name>\My Documents\Cypress\FM_PDL_2.0.1
```

And, the example projects will be extracted to the following default directory:

```
C:\Users\<User Name>\My Documents\Cypress
\FM0+ S6E1B-Series Starter Kit_Ver01
```

In the rest of this document, the following directory is termed as <User_Directory>:

```
C:\Users\<User Name>\My Documents\Cypress
```

2.2 Un-install Software

The software can be uninstalled using one of the following methods:

- Go to **Start** > **All Programs** > **Cypress** > **Cypress Update Manager** and select the **Uninstall** button that corresponds to the kit software.
- Go to **Start** > **Control Panel** > **Programs and Features for Windows 7** or **Add/Remove Programs for Windows XP**; select the **Uninstall/Change** button.

Note: Uninstalling the Kit software will not remove the FM PDL 2.0.1 and FM0+ S6E1B-Series Starter Kit Example Projects from <User_Directory>.

2.3 Test Operation

The FM0+ S6E1B-Series Starter Kit has been pre-programmed with a test demo code, which helps to test all the on-board features. The Motorola-S format file, *tp_fm-100l-s6e1b8.srec*, is provided in the following directory.

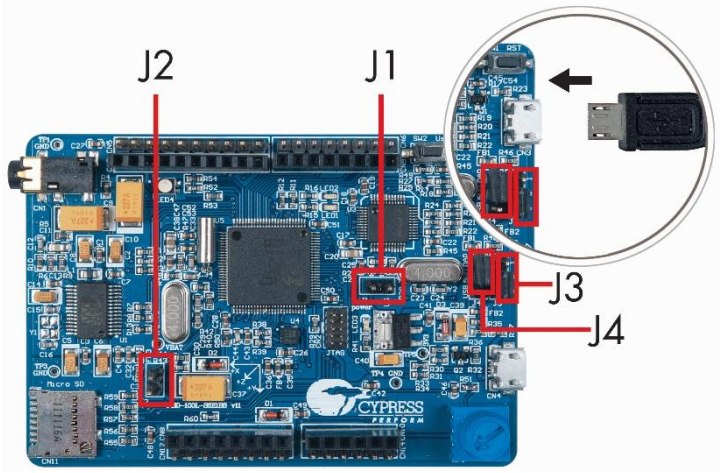
```
<User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
\Demo Projects\Test_Demo_Code
```

2.3.1 Run the Test Demo

Follow the instructions below to run the test code.

1. Ensure the jumpers J1 and J2 are open. Close Pin 2 and Pin 3 of J3 (default). Close Pin 1 and Pin 2 of J4 (default) and connect the CN3 connector to a PC using the USB cable provided.

Figure 2-5. Jumper Settings for Test Code

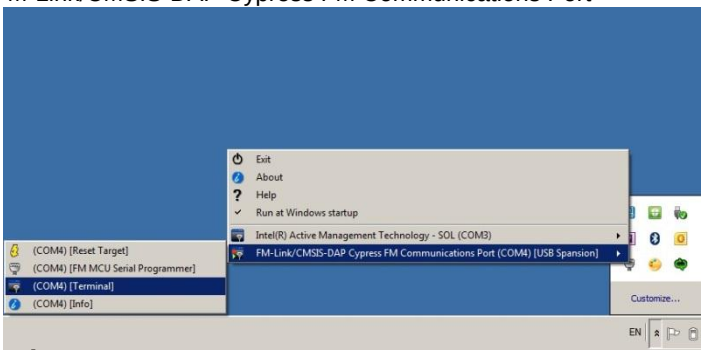


2. If not launched already, then launch the Serial Port Viewer from the start menu.

   Start Menu > **All Programs** > **Cypress** > **Serial Port Viewer Tool**

3. Click on the Serial Port Viewer icon in the task bar and select FM-Link/CMSIS-DAP Cypress FM Communications Port.

Figure 2-6. FM-Link/CMSIS-DAP Cypress FM Communications Port

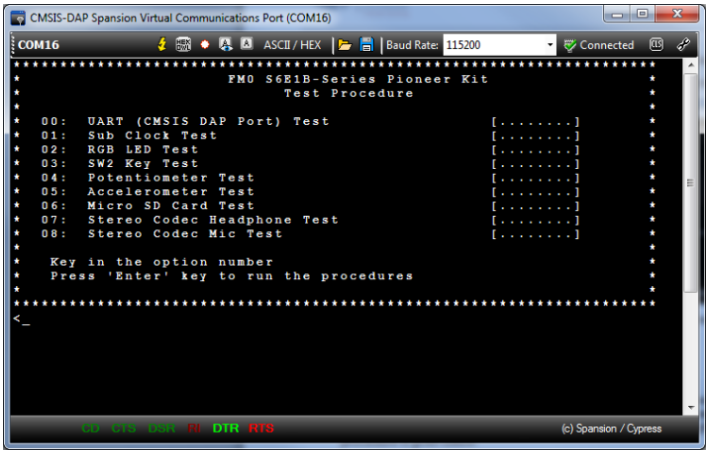


4. Select the baud rate of **115200**, and click the **Disconnected** button to connect to the board.

Figure 2-7: Select the Baud Rate

5. Press the **Enter** key on your keyboard to run the test procedure. Key in the option number and press the **Enter** key to run any of the functions.

Figure 2-8: Test Procedure



6. For example, key in **2** and press the **Enter** key on the PC keyboard.

Figure 2-9: RGB LED Test

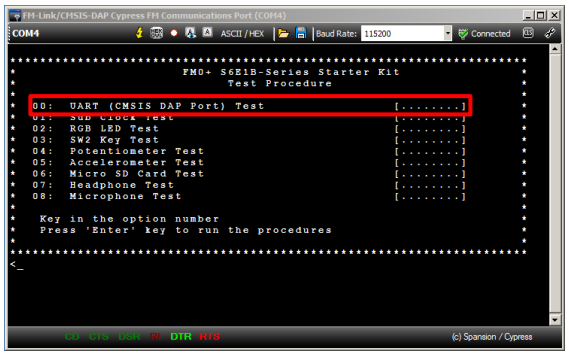
7. Watch the RGB LED (LED4), its color will change from red to green to blue. Then press the **Enter** key on the PC to end the test.

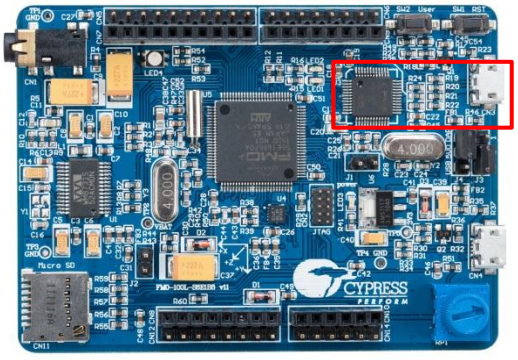
2.3.2 Test Procedure Explanation

This section explains the test procedures. This test procedure is based on the Serial Port Viewer Tool. The user has to key-in the test procedure number displayed on the menu and then press the **Enter** key on the PC. The firmware on the board will run the test procedure and display the results. There are eight test procedures and the section below provides a brief description of each test procedure.

- **UART Test:** This procedure will test the UART communication between the CMSIS-DAP device and the S6E1B8 MCU. Key in **0** and press the **Enter** key.

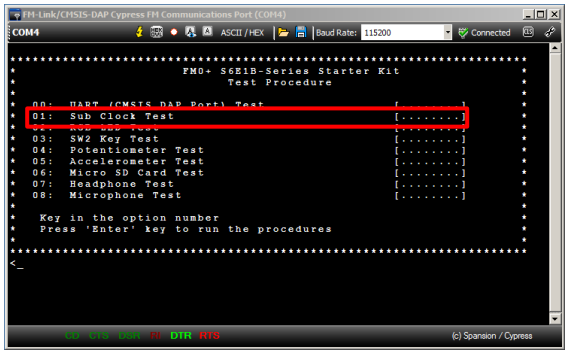
Figure 2-10: UART Test

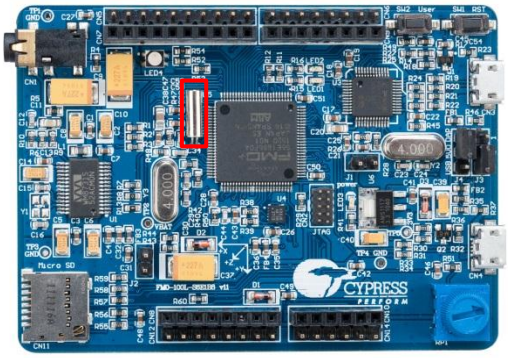




- **Sub Clock Test**: This procedure will test whether the sub clock is at 32.768 KHz or not. Key in **1** and press the **Enter** key, the main routine will shift the system clock to the sub clock, and shift back to the main clock after the sub clock is confirmed to be running at 32.768 KHz.

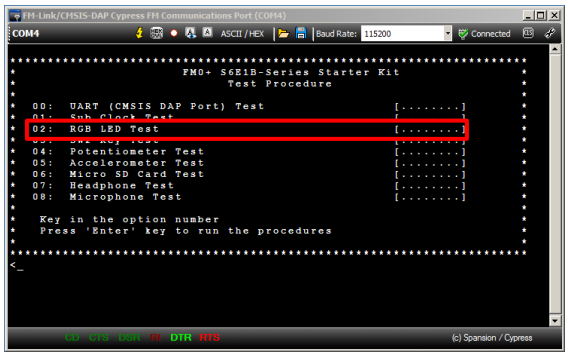
Figure 2-11: Sub Clock Test

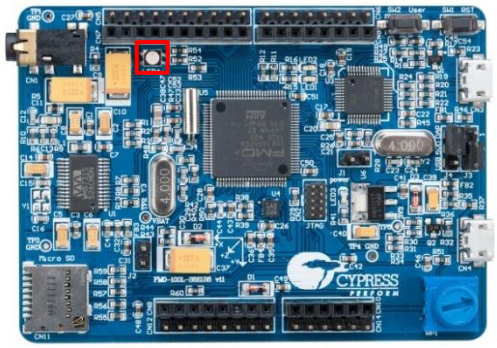




- **RGB LED Test**: This procedure is to test the RGB LED. Key in **2** and press the **Enter** key, the RGB LED color will change from red to green to blue.

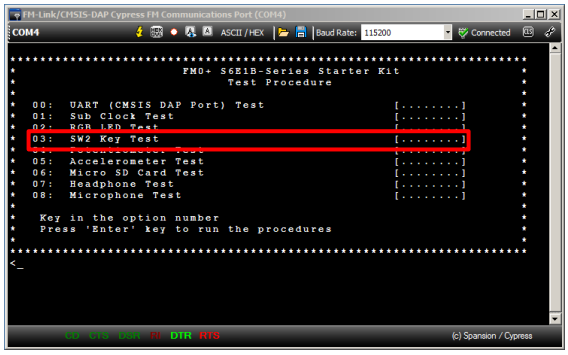
Figure 2-12: RGB LED Test

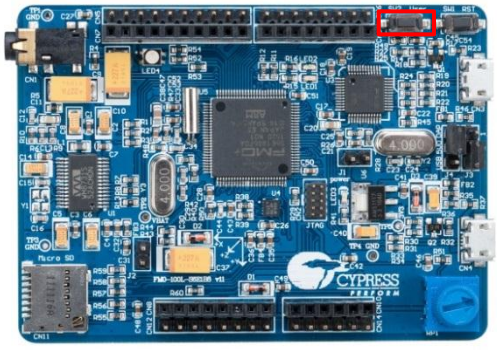




- **SW2 Key Test**: This procedure tests the SW2 button. Key in **3** and press the **Enter** key, the test routine will detect the press and release status of the User Button SW2.

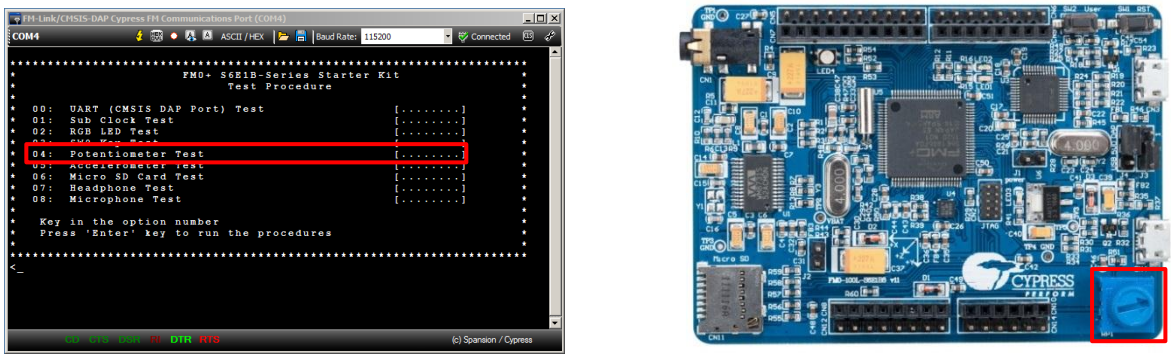
Figure 2-13: SW2 Button Test





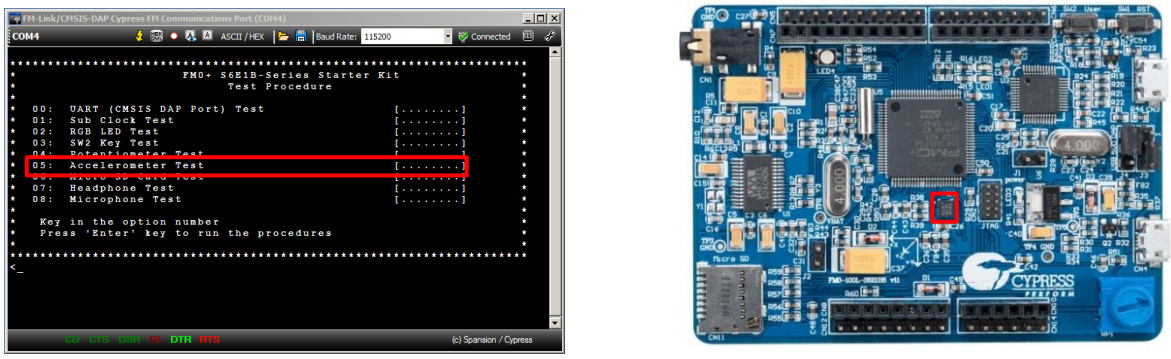
- **Potentiometer Test**: This procedure is to test the potentiometer. Key in **4** and press the **Enter** key. The ADC value will be displayed on the terminal. Turn the potentiometer (RP1), the ADC value will change accordingly.

Figure 2-14: Potentiometer Test



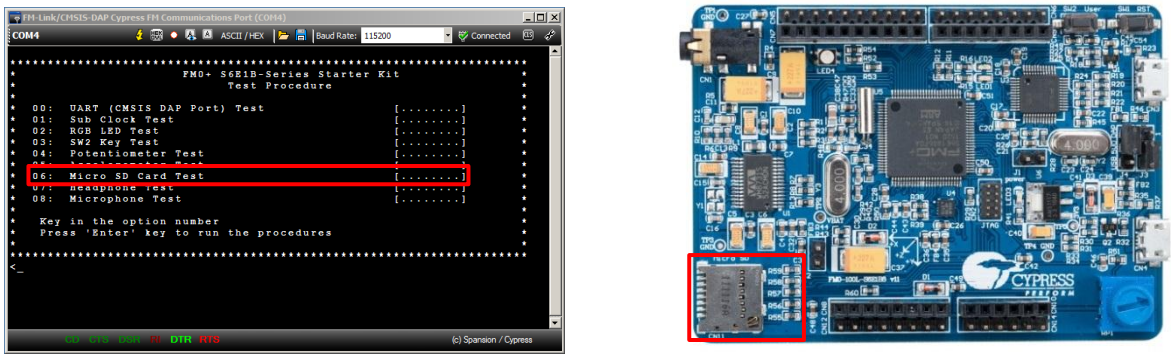
- **Accelerometer Test**: This procedure is to evaluate the accelerometer. Key in **5** and press the **Enter** key. The terminal will show the data of each of axes. This data will change if there is any change in the board position.

Figure 2-15: Accelerometer Test



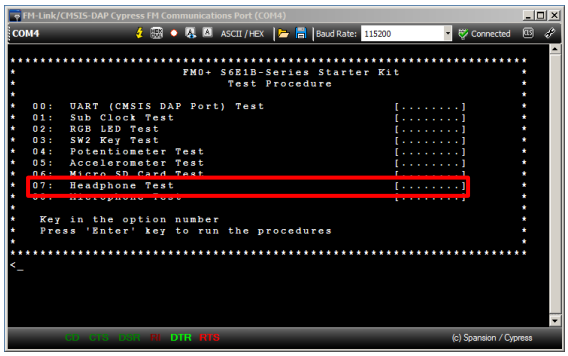
- **Micro SD Card Test**: This procedure is to evaluate the Micro SD card interface. Insert a Micro SD card into the socket, CN11, and then key in **6** and press the **Enter** key, the test result will be displayed on the terminal.

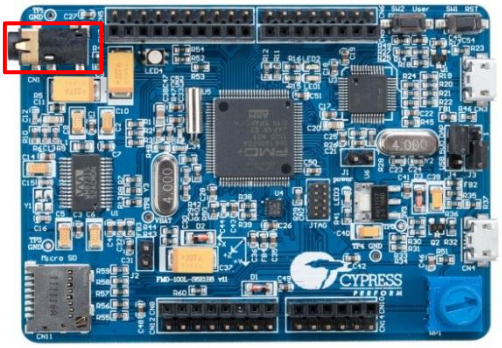
Figure 2-16: Micro SD Card Test



- **Headphone Test**: This procedure is to evaluate the headphone channel of the stereo codec. Key in **7** and press the **Enter** key, a pre-defined calling bell sound will be heard on the headphone connected to **CN1**.

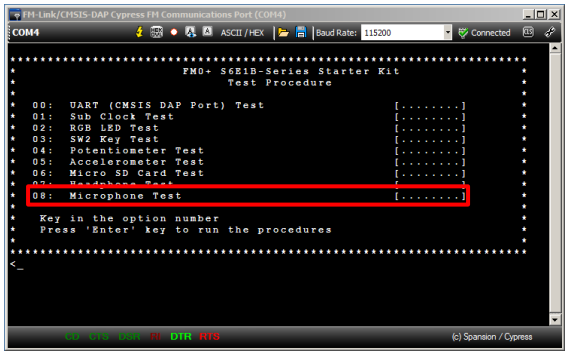
Figure 2-17: Headphone Test

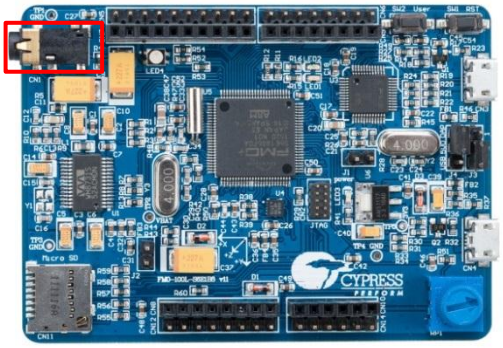




- **Microphone Test**: This procedure tests the microphone channel of the stereo codec. Key in **8** and press the **Enter** key, you can hear your voice from the microphone on the headphone connected to **CN1**.

Figure 2-18: Microhone Test





Note: Connect a headset with a 4-conductor phone plug (American Headset Jack) to CN1 for this test to function.

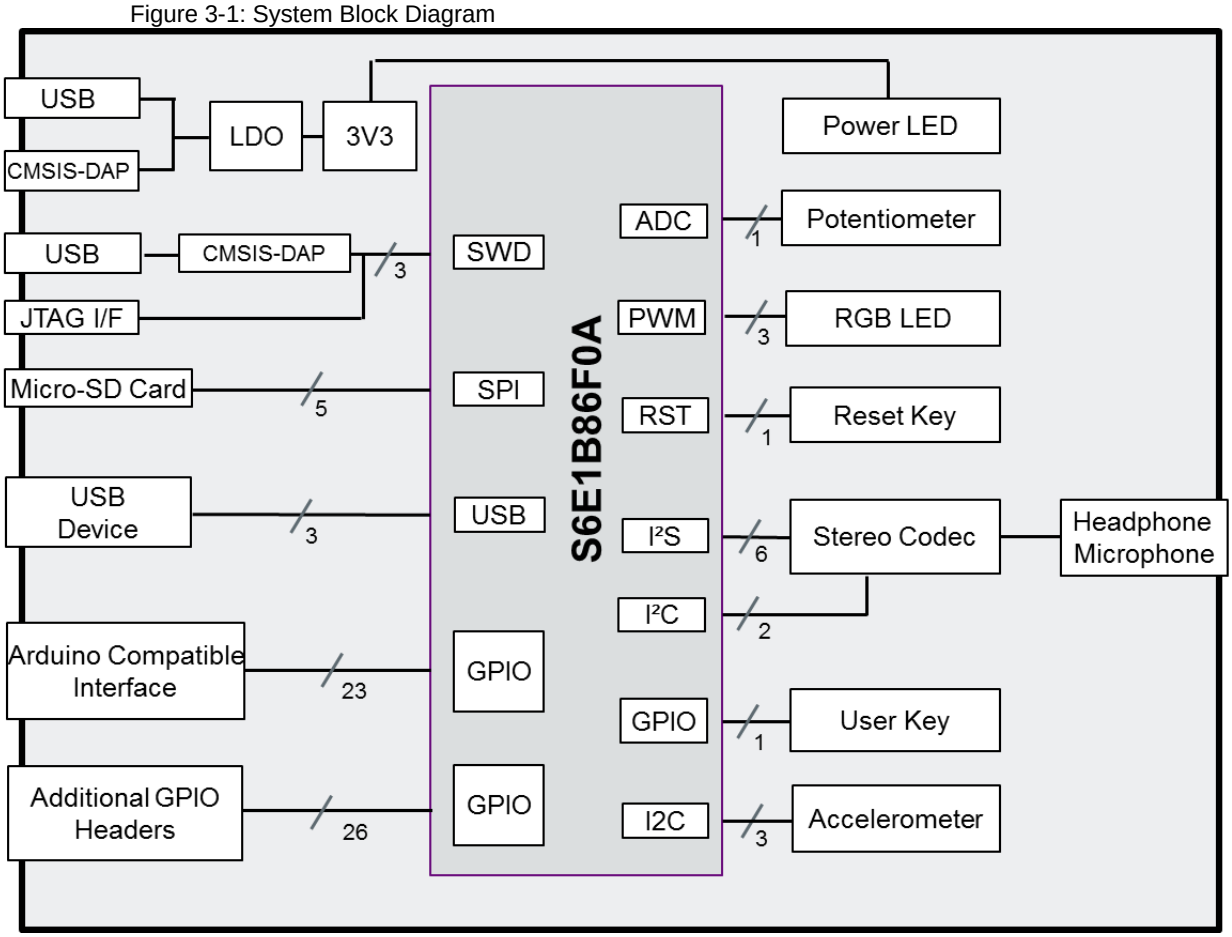
3. Hardware

This chapter describes the features and hardware details of the FM0+ S6E1B-Series Starter Kit.

3.1 System Block Diagram

Figure 3-1 shows the block diagram of the FM0+ S6E1B-Series Starter Kit.

Figure 3-1: System Block Diagram



3.2 Hardware Features

- Cypress FM0+ S6E1B-Series MCU
- On-board ICE (CMSIS-DAP compatible)
- 10-pin JTAG interface
- USB device interface
- Micro SD card connector
- Stereo codec
- Accelerometer
- Potentiometer
- RGB LED
- User button
- Arduino compatible headers
- Additional GPIO headers

3.3 Hardware Details

3.3.1 FM0+ Series MCU

The FM0+ S6E1B-Series Starter Kit features an ultra-low-power, highly integrated S6E1B8 MCU, a 32-bit ARM® Cortex®-M0+ MCU.

The S6E1B8 MCU is a member of the S6E1B-Series device family with 40MHz CPU, 560KB flash, 64KB SRAM and 82 GPIOs. The S6E1B8 MCU features a wide variety of peripherals such as multi-function timers (MFT), programmable pulse generators (PPG), 12-bit SAR ADC, LCD Controller, and communication interfaces like USB (host and device), DMA, CSIO (SPI), I2S, I2C and UART.

3.3.2 User Button and LED

The FM0+ S6E1B-Series Starter Kit features a user button and a 3-color LED. The switch and the LED are connected to the S6E1B8 MCU device via pins listed in Table 3-1.

Table 3-1: Button and LED

Pin No.	Pin Name	External Device
85	P08/AN23/TIOA0_2/CTS4_2/INT16_0/SEG06	SW2
22	P3D/RTO03_0/TIOA3_1/IC1_VPEN_0/SEG36	LED4 – red
23	P3E/RTO04_0/TIOA4_1/INT19_2/IC1_VCC_0/SEG35	LED4 – green
24	P3F/RTO05_0/TIOA5_1/IC1_CLK_0/SEG34	LED4 – blue

The port P3D/P3E/P3F pins are also assigned as the PWM output pins, the user can dim the LED by configuring the base timers in PWM mode to output PWM signals.

3.3.3 Arduino Compatible Interface

The FM0+ S6E1B-Series Starter Kit provides footprint compatibility with the Arduino interface. These headers expand the possibility for the user to develop more applications based on this development kit using Arduino compatible shields. Figure 3-2 shows the pins.

Figure 3-2: Pins of Arduino Compatible Interface

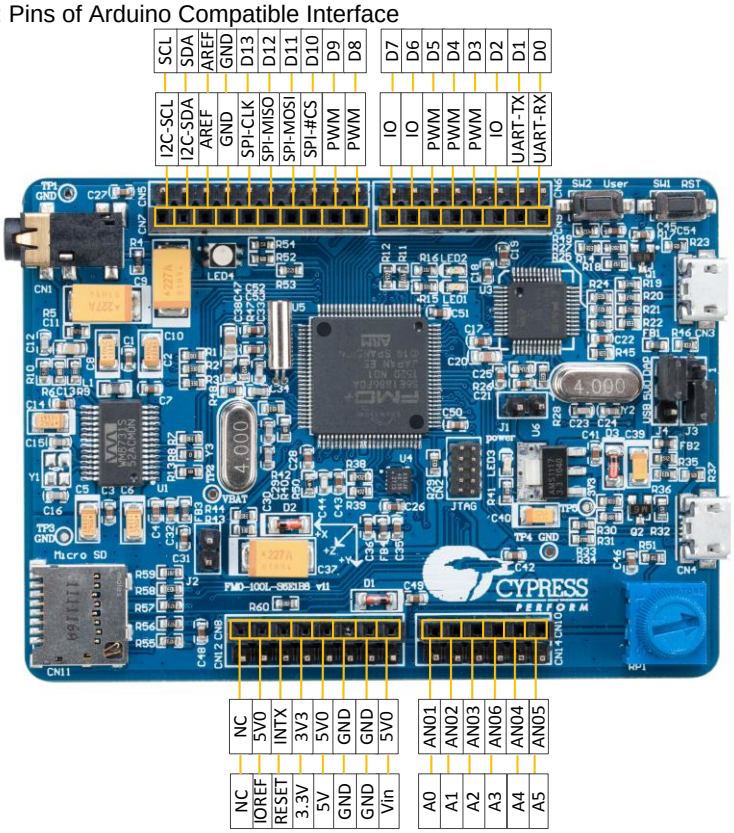


Table 3-2 shows full functions of the pins connected with the Arduino headers.

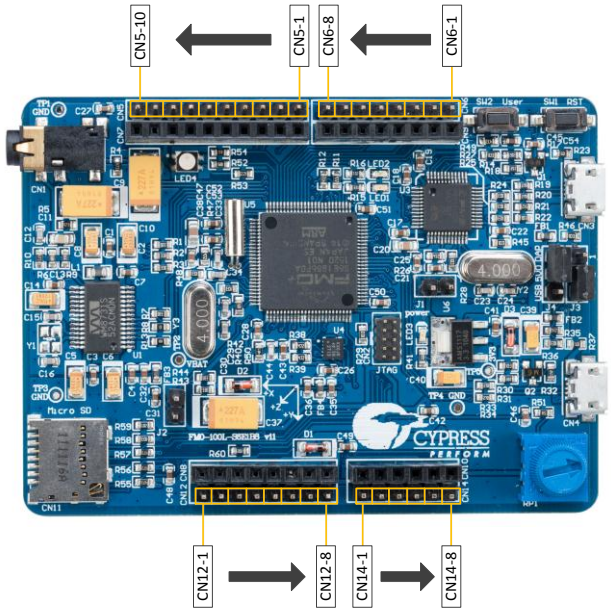
Table 3-2: Pins of Arduino Compatible Interface

Pin No.	Pin Name	Arduino designation	Function (part)
15	P36/IC02_0/SIN5_2/INT09_1/WKUP11	D0	UART – RX
16	P37/IC01_0/SOT5_2/INT10_1	D1	UART – TX
17	P38/IC00_0/SCK5_2/INT11_1	D2	IO
19	P3A/RTO00_0/TIOA0_1/INT07_0/RTCCO_2/SUBOUT_2 /IC1_CIN_0/COM2	D3	PWM
18	P39/DTTI0X_0/ADTG_2/TIOB4_0/INT06_0/COM3	D4	IO
20	P3B/RTO01_0/TIOA1_1/IC1_DATA_0/COM1	D5	PWM
21	P3C/RTO02_0/TIOA2_1/INT18_2/IC1_RST_0/COM0	D6	PWM
14	P35/SCS62_1/IC03_0/TIOB5_1/INT08_1/SEG37	D7	IO
32	P45/LVDI/TIOA5_0/SEG32/IC0_CIN_1	D8	PWM
31	P44/TIOA4_0/INT10_0/SEG33/RTS1_2/IC0_DATA_1	D9	PWM
30	P43/TIOA3_0/INT09_0/ADTG_7/CTS1_2/IC0_RST_1	D10	SPI -- #CS
28	P41/TIOA1_0/INT13_1/SOT1_2/IC0_VCC_1	D11	SPI -- SOT
27	P40/TIOA0_0/INT12_1/SIN1_2/IC0_CLK_1	D12	SPI -- SIN
29	P42/TIOA2_0/INT08_0/SCK1_2/IC0_VPEN_1	D13	SPI -- SCK
44	P4C/TIOB2_0/SOT7_1/INT12_0/SEG29/CEC0_0	D14	I2C -- SDA
45	P4D/TIOB3_0/INT13_0/SCK7_1/WKUP6/SEG28	D15	I2C -- SCL
69	P1D/AN13/CTS4_1/DTTI0X_1/INT22_2/SEG14	A0	AN13
70	P1E/AN14/RTS4_1/ADTG_5/FRCK0_1/INT23_2	A1	AN14
71	P23/AN16/SCK0_0/TIOA7_1/RTO00_1/SEG13	A2	AN16
66	P1A/AN10/SIN4_1/IC01_1/INT05_1/SEG17	A3	AN10
67	P1B/AN11/SOT4_1/IC02_1/INT20_2/SEG16	A4	AN11/I2C--SDA
68	P1C/AN12/SCK4_1/IC03_1/INT21_2/SEG15	A5	AN12/I2C--SCL
36	INITX	RESET	RESET

3.3.4 Additional GPIO Headers

The unused pins of the S6E1B8 MCU are routed to the CN5, CN6, CN12 and CN14 I/O headers.

Figure 3-3: Additional GPIO Pins



These additional GPIO interfaces make it easy for the user to access more GPIOs and peripherals, like the MFS (Multi-Function Serial), PWM and I2S. Table 3-3 shows details of the pins.

Table 3-3: Additional GPIO

Pin No.	Pin Name	Designation	Functions (part)
N/A	N/A	CN6-1	GND
N/A	N/A	CN6-2	3V3
94	P62/SCK5_0/I2SCK5_0/ADTG_3/INT07_1/SEG01/TIOA6_1/ IC0_RST_0	CN6-3	MFS,I2S,PWM,INT
95	P61/SOT5_0/I2SDO5_0/TIOB2_2/DTTI0X_2/SEG00	CN6-4	MFS,I2S
86	P09/TIOB0_2/RTS4_2/INT17_0/SEG05	CN6-5	IO,INT
97	P80/SIN7_2/INT20_1/C0	CN6-6	MFS,INT
98	P81/SOT7_2/INT11_0/C1	CN6-7	MFS,INT
99	P82/SCK7_2/MD2	CN6-8	MFS
2	P50/INT00_0/SIN3_1/VV4	CN5-1	MFS,INT
5	P53/SIN6_0/TIOA1_2/INT07_2/VV1/WKUP8	CN5-2	MFS,PWM,INT
6	P54/SOT6_0/TIOB1_2/INT18_1/VV0	CN5-3	MFS,PWM
7	P55/SCK6_0/ADTG_1/INT19_1/SEG39	CN5-4	MFS,INT
13	P34/SCS61_1/FRCK0_0/TIOB4_1	CN5-5	IO, Base Timer

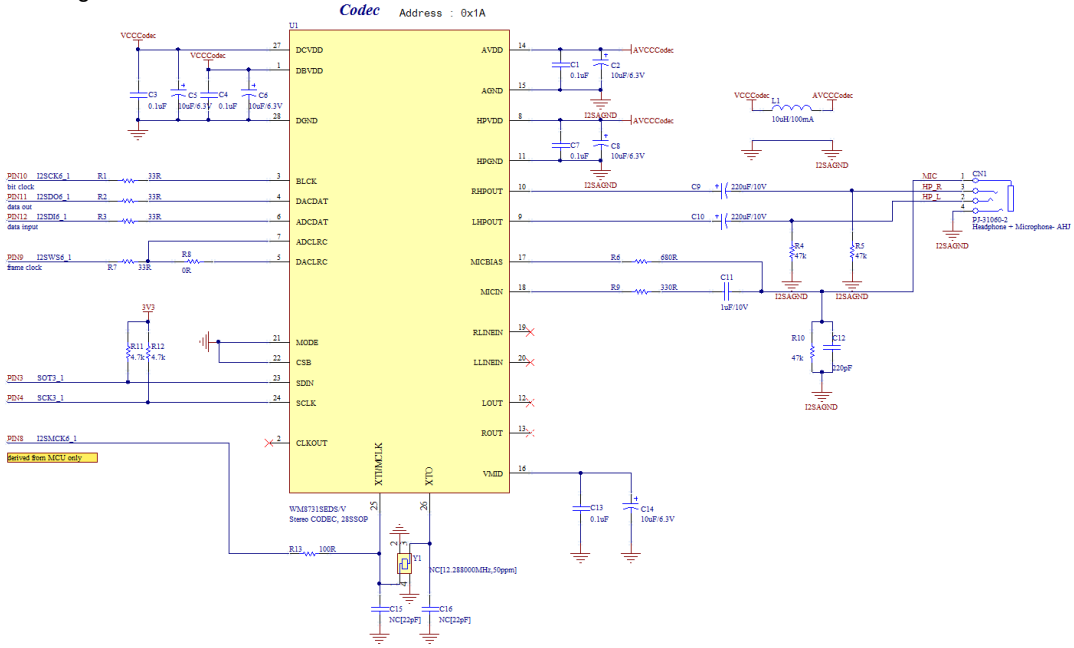
Pin No.	Pin Name	Designation	Functions (part)
39	P48/VREGCTL	CN5-6	IO
40	P49/VWAKEUP	CN5-7	IO
42	P4A/TIOB0_0/SCS70_1/INT21_1/SEG31	CN5-8	IO,INT, Base Timer
N/A	N/A	CN5-9	3V3
N/A	N/A	CN5-10	GND
90	P0D/RTS4_0/TIOA3_2/INT20_0/SEG04/IC0_VPEN_0	CN14-1	PWM,INT
91	P0E/CTS4_0/TIOB3_2/INT21_0/SEG03/IC0_VCC_0	CN14-2	INT, Base Timer
92	P0F/NMIX/CROUT_1/RTCCO_0/SUBOUT_0/I2SMCK5_0/WKUP0 /IC0_CLK_0/SCK4_0	CN14-3	NMIX,I2S
93	P63/SIN5_1/I2SWS5_0/INT03_0/SEG02/TIOB6_1/IC0_DATA_0	CN14-4	MFS,I2S,INT,ISO7816
79	P02/SIN3_2/TIOB5_0	CN14-5	MFS, Base Timer
59	P17/AN07/SIN2_2/INT04_1/SEG20	CN14-6	ADC,MFS,INT
52	P10/AN00/IC1_CLK_1/CTS1_1/SEG27	CN12-8	ADC,ISO7816
54	P12/AN02/IC1_VPEN_1/SOT1_1/IC00_2/SEG25	CN12-7	ADC,MFS,ISO7816
55	P13/AN03/IC1_RST_1/SCK1_1/RTCCO_1/IC01_2/SUBOUT_1/SE G24	CN12-6	ADC,MFS,ISO7816
56	P14/AN04/IC1_DATA_1/RTS1_1/SIN0_1/INT03_1/IC02_2/SEG23	CN12-5	ADC,INT,ISO7816
57	P15/AN05/IC1_CIN_1/SOT0_1/IC03_2/INT14_0/SEG22	CN12-4	ADC,MFS,INT,ISO7816
58	P16/AN06/SCK0_1/INT15_0/SEG21	CN12-3	ADC,MFS,INT
N/A	N/A	CN12-2	3V3
N/A	N/A	CN12-1	GND

3.3.5 Stereo Codec

The WM8731 is a low power stereo codec with an integrated headphone driver. On the digital side, it has an Inter-IC Sound interface (I2S) which is connected with the MCU's I2S macro. Figure 3-4 shows the details of the connections. The codec also has an I2C interface for configuring the device. The I2C address for configuring the stereo codec is 0x1A.

On the analog side, the codec has headphone, microphone, line-in and line-out channels. On this kit, only the headphone and microphone are routed to the jack, CN1.

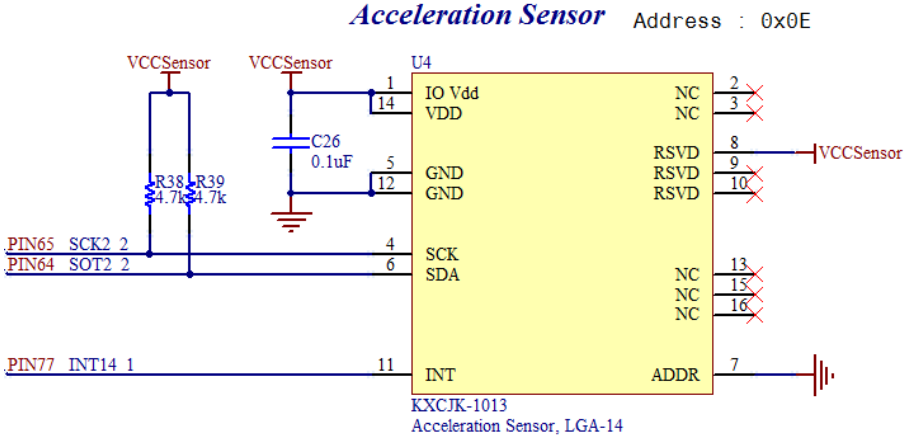
Figure 3-4: Stereo Codec Circuit



3.3.6 Accelerometer

The KXCJK-1013 is a 3-axis ±2g, ±4g or ±8g silicon micro-machined accelerometer. This sensor communicates with the MCU via an I2C interface. The I2C address for configuring the accelerometer is 0x0E. The accelerometer can also interrupt the FM0+ S6E1B8 MCU when a change is detected in either of the three axes.

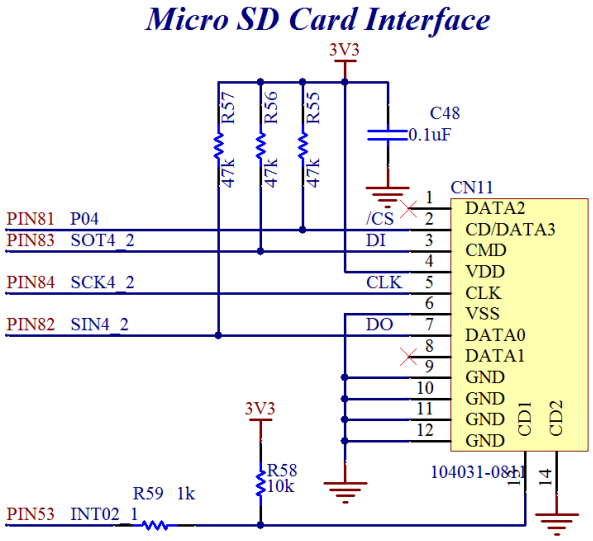
Figure 3-5: Accelerometer



3.3.7 Micro SD Card

The FM0+ S6E1B-Series Starter Kit features a Micro SD card interface, CN11. The Micro SD card is connected to the FM0+ S6E1B8 MCU via SPI. Figure 3-6 shows the connection details in the SPI mode.

Figure 3-6: Micro SD Card in SPI Mode



The pin arrangement of the Micro SD card is shown as below:

Figure 3-7: Pin arrangement of Micro SD Card

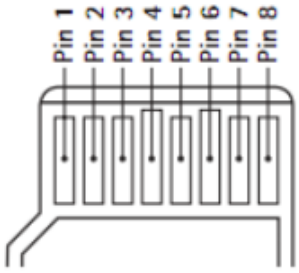


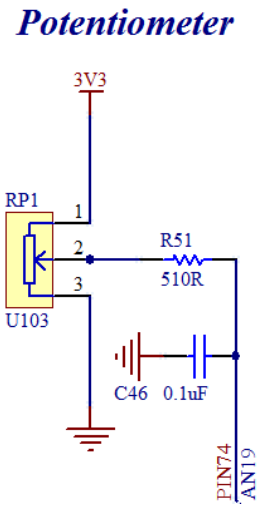
Table 3-4: Micro SD Card Signals

Number	Name	Type	Description
1	RSV	-	Reserved in SPI mode
2	CS	I	Chip select
3	DI	I	Data in
4	VDD	power	Power supply
5	CLK	I	Clock
6	VSS	power	Power supply ground
7	DO	O	Data out
8	RSV	-	Reserved in SPI mode

3.3.8 Potentiometer

The FM0+ S6E1B-Series Starter Kit provides a potentiometer (RP1) with resistance value ranging from 0 to 10kΩ. The middle terminal is connected to the ADC channel AN19 (Pin 74).

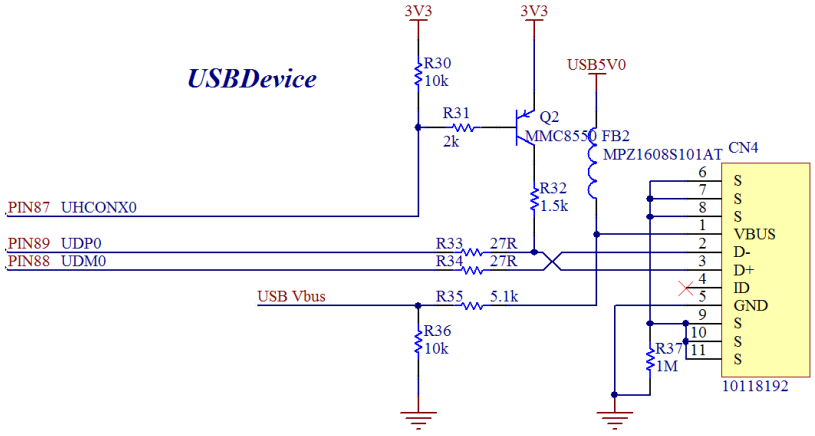
Figure 3-8: Potentiometer



3.3.9 USB Interface

The FM0+ S6E1B8 MCU has a USB unit that can work as a host or a device. In the FM0+ S6E1B-Series Starter Kit, the USB port is configured as a USB device and is connected to CN4, a micro USB type B connector.

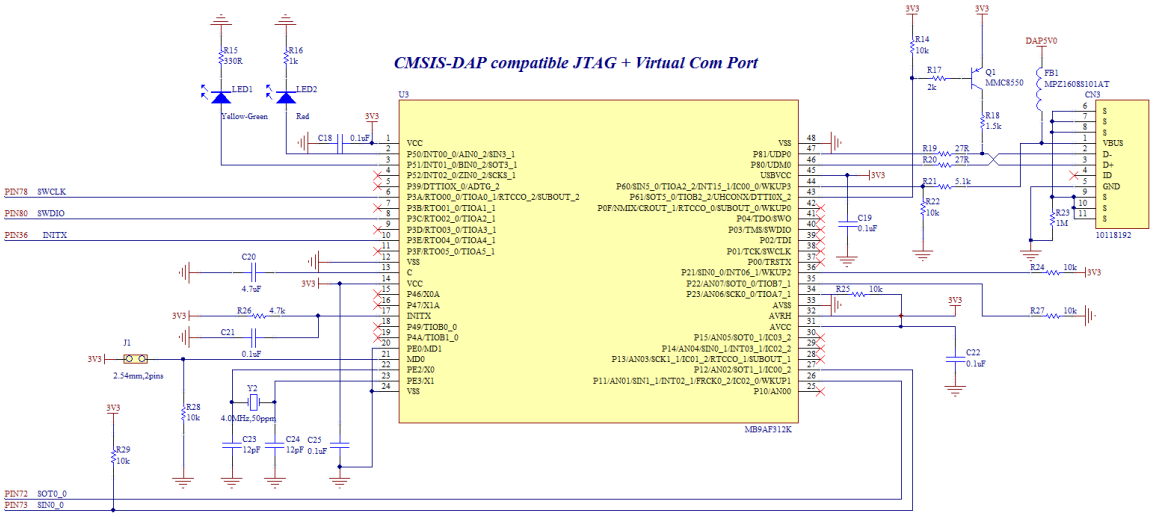
Figure 3-9: USB Device



3.3.10 CMSIS-DAP

The FM0+ S6E1B-Series Starter Kit features an on-board CMSIS-DAP module to enable programming and debugging of the FM0+ S6E1B8 MCU. The CMSIS-DAP firmware solution supports full JTAG configuration and a two-wire Serial Wire Debug (SWD) interface. The CMSIS-DAP module can also power the FM0+ S6E1B-Series Starter Kit via the CN3 connector, when Pin 1 and Pin 2 of Jumper J4 are shorted.

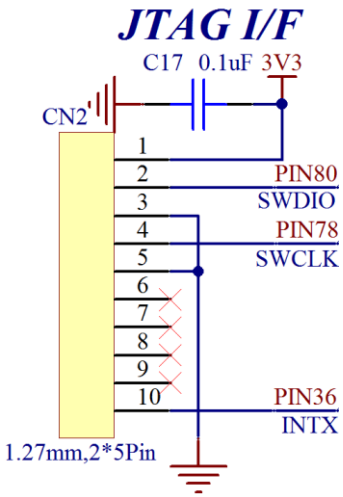
Figure 3-10: CMSIS-DAP Circuit



3.3.11 JTAG

The FM0+ S6E1B-Series Starter Kit provides an interface, CN2, to connect an external programmer for programming the FM0+ S6E1B8 MCU. CN2 is a standard ARM 0.05'' 10-pin Cortex debug header.

Figure 3-11: 10-pin JTAG I/F



4. Software Development

4.1 Tool Options

The FM0+ S6E1B-Series device is supported by several third party tools/IDEs, and the user can choose their preferred tool for development. The example projects can be opened and compiled in either of following IDEs:

- IAR Embedded Workbench for ARM
- Keil ARM RealView® Microcontroller Development System

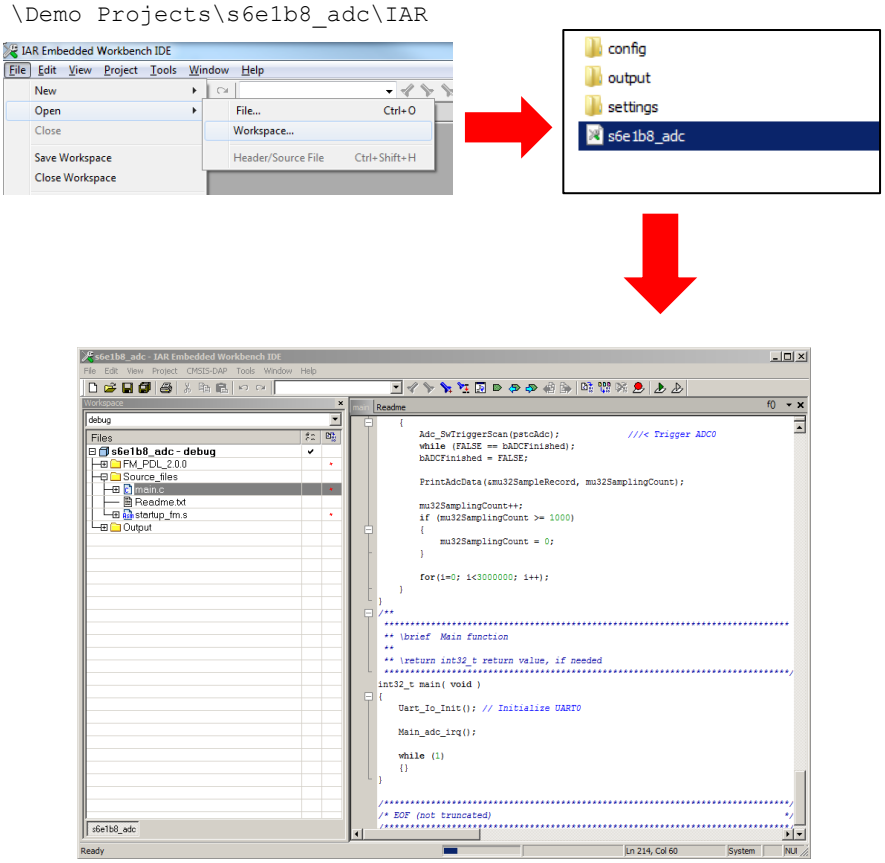
Download evaluation versions of these tools from the vendor's website. A full license may be required to build or debug some of the examples. For detailed information on using the tools, see the documentation in the **Help** section of the tool chain or the website of the tools supplier.

4.1.1 Open the Example Projects in IAR IDE

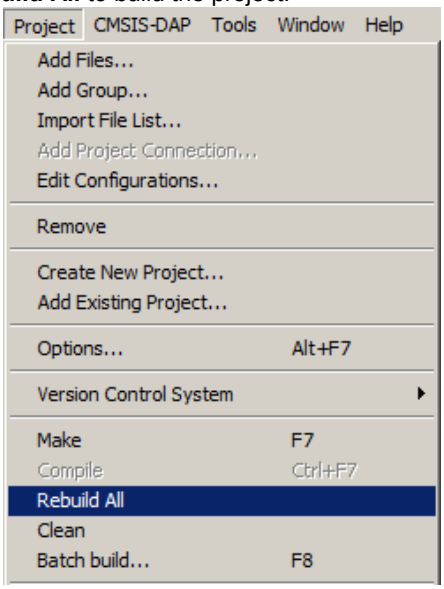
The following steps describe how to open, build, and run an example project in the IAR IDE. The *s6e1b8_adc* is one example project that is available after a successful installation

1. Launch IAR Embedded Workbench IDE V7.40.5.9739 (or later).

2. Click **File** > **Open** > **Workspace** and select the workspace file *s6e1b8_adc.eww* from `<User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware \Demo Projects\s6e1b8_adc\IAR`



3. Click **Project** > **Rebuild All** to build the project.



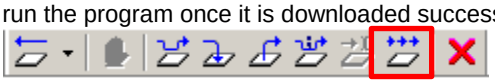
4. Make sure the jumpers on the FM0+ S6E1B-Series Starter board are placed according to Table 4-1.

Table 4-1: Debugging Jumper Setting

Jumper	Position	Description
J1	Open	Sets MB9AF312K (CMSIS-DAP) in run mode.
J2	Open	Sets S6E1B8 in run mode.
J4	Pin 1 to Pin 2	Power from USB port of CMSIS-DAP (CN3)

5. Connect the USB cable to CN3 port.
6. Observe that Power LED (LED3) is glowing green.
7. Click the **Debug** icon in the tool bar; use Shortcut **Crtl+D** or choose **Project** > **Download and Debug** to start downloading and debugging.

8. Click **Run** icon to run the program once it is downloaded successfully.



9. Click **Stop** icon to stop the program.

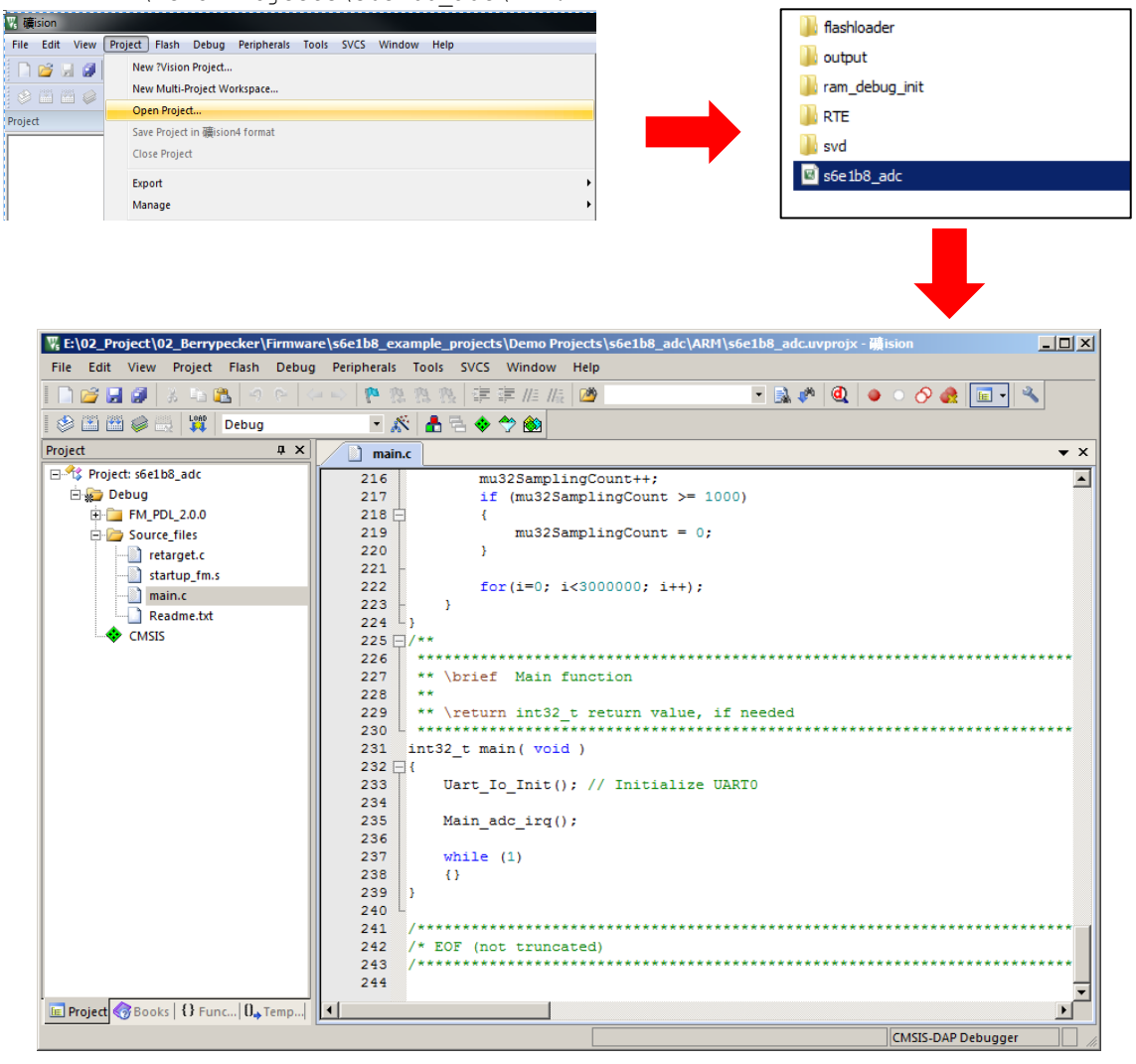
For more information about the IAR Embedded Workbench IDE, please click **Help**.

4.1.2 Open the Example Projects in the Keil µVision IDE

The following steps describe how to open, build and run an example project in Keil µVision IDE.

The *s6e1b8_adc* is one example project that is available after a successful installation.

1. Launch Keil µVision IDE v5.16a (or later).
2. Click **Project** > **Open Project** and select the workspace file *s6e1b8_dac.uvprojx* from `<User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware \Demo Projects\s6e1b8_adc\ARM`.



3. Click the **Build** icon to build the project.

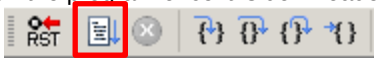
4. Make sure the jumpers on the FM0+ S6E1B-Series Starter board are placed according to Table 4-2.

Table 4-2: Debugging Jumper Setting

Jumper	Position	Description
J1	Open	Sets MB9AF312K (CMSIS-DAP) in run mode.
J2	Open	Sets S6E1B8 in run mode.
J4	Pin 1 to Pin 2	Power from USB port of CMSIS-DAP (CN3)

5. Connect the USB cable to the CN3 port.
6. Observe that Power LED (LED3) is glowing green.
7. Click the **Debug** icon, use shortcut **Crtl+F5** or choose **Debug** > **Start/Stop Debug Session** to start downloading and debugging.

8. Click the **Run** icon to run the program once it is downloaded successfully.



9. Click the **Stop** icon to stop the program when you want.

For more information about the Keil µVision IDE, please click **Help**.

4.2 Example Projects

The FM0+ S6E1B-Series Starter Kit provides twelve example projects to help the user get a quick start with the S6E1B8 device. They are included in the directory:

```
<User_Directory>:\FM0+ S6E1B-Series Starter
Kit_Ver01\Firmware\Demo Projects
```

These examples listed in Table 4-3 are based on the Peripheral Driver Library (PDL). The PDL provides the API for initializing and operating on-chip peripherals.

Use IAR Embedded Workbench v7.40.5.9739 (or later) or Keil µVision IDE v5.16a (or later) to open the example projects.

Table 4-3: Example Projects

#	Projects	Title/Description
1	S6E1B8_adc	Title: Analog-to-Digital Converter Description: This project enables ADC channel AN19 to measure the voltage of the potentiometer output. It is converted to a decimal value and sent out using UART0. Refer to section AD Converter for details.
2	S6E1B8_bt_pwm	Title: Base Timer Description: This project demonstrates the base timer operation of the S6E1B8 device. The program configures base timer 4 in PWM mode to generate a PWM sequence. The sequence outputs from the TIOA4_1 pin to drive the green LED of LED4. The PWM duty cycle is updated every 2ms by base timer 2 which produces a breathing LED effect.Refer to section Base Timer for details.
3	S6E1B8_dstc	Title: Descriptor System Data Transfer Controller (DSTC)

#	Projects	Title/Description
		Description: This project demonstrates the Descriptor System Data Transfer Controller (DSTC) operation of the S6E1B8 device. The program configures DSTC to move the data in au32SourceData (source array) to au32DestinationData (destination array), and then compares the content of the arrays to verify the data. Refer to section Descriptor System Data Transfer Controller (DSTC) for details.
4	S6E1B8_ext_int	Title: External Interrupt Description: This project demonstrates the external interrupt operation of the S6E1B8 device. SW2 is connected to the non-maskable external interrupt (INT16_0). Pressing the SW2 key on the board will change the color of the RGB LED (LED4) from red to green to blue. Refer to section External Interrupt for details.
5	S6E1B8_flash	Title: Flash Write Description: This project demonstrates the flash writing operation of the S6E1B8 device. A specific set of four values each of four bytes in size will be written into a specific address location in the flash memory. Refer to section Flash for details.
6	S6E1B8_gpio	Title: GPIO Description: This project demonstrates the GPIO operations of the S6E1B8 device by driving an LED. Pin P3E drives the green LED of LED4. When program runs, the P3E outputs a sequence to drive the green LED of LED4. The RGB LED (LED4) will blink with green color accordingly. Refer to section GPIO for details.
7	S6E1B8_mfs_uart	Title: Multi-function Serial Interface Description: This project demonstrates UART communication of the S6E1B8 device. This program enables the MFS0 as UART mode to communicate with the CMSIS-DAP device. The CMSIS-DAP device serves as the bridge between the MCU and PC. Refer to section UART Communication for details.
8	S6E1B8_mft_frt	Title: Multi-function Timer Description: This project demonstrates the multi-function timer (MFT) operation of the S6E1B8 device. The project configures the multi-function timer unit0 in free-run timer mode. An interrupt will occur at the peak point and zero point of the counter respectively. The state of P3E (which drives the green LED of LED4) will be changed in the interrupt service routines. Refer to section Multi-function Timer for details.
9	S6E1B8_rtc	Title: Real Time Clock Description: This project demonstrates the RTC operation of the S6E1B8 device. The program enables the RTC in calendar mode, and sends out the current calendar data through UART0.The calendar starts from 2015/9/13 23:59:01 Wednesday. Refer to section RTC calendar for details.
10	S6E1B8_sleep_mode	Title: Sleep Mode Description: This project demonstrates the sleep mode operation of the S6E1B8 device. The MCU will enter sleep mode after blinking the RGB LED (LED4) 5 times (green color). It can be awakened up by pressing SW2 key, and then the RGB LED (LED4) will glow green. Refer to section Sleep Mode for details.
11	S6E1B8_sw_wdt	Title: Software Watchdog Description: This project demonstrates the software watchdog operation of the

#	Projects	Title/Description
		S6E1B8 device. This program will show the influence of feeding the software watchdog or not feeding the software watchdog. If the software watchdog is enabled and fed, the program will run normally, and the RGB LED (LED4) will blink with green color. If the software watchdog is enabled but not fed, the S6E1B8 device will reset, and the RGB LED (LED4) will be glow green after reset. Refer to section Software Watchdog for details.
12	S6E1B8_wc	Title: Watch Timer Description: This project demonstrates the watch timer operation of the S6E1B8 device. The watch timer will generate an interrupt once per second. In the interrupt service routine, P3E will be toggled which makes the green LED of the RGB LED (LED4) blink. Refer to section Watch Timer for details.

4.2.1 AD Converter

4.2.1.1 Project Description

This project demonstrates the AD conversion of the S6E1B8 device. The project enables ADC channel AN19 to measure the voltage from a potentiometer. It is converted to a decimal value and sent out using UART0.

4.2.1.2 Hardware Connection

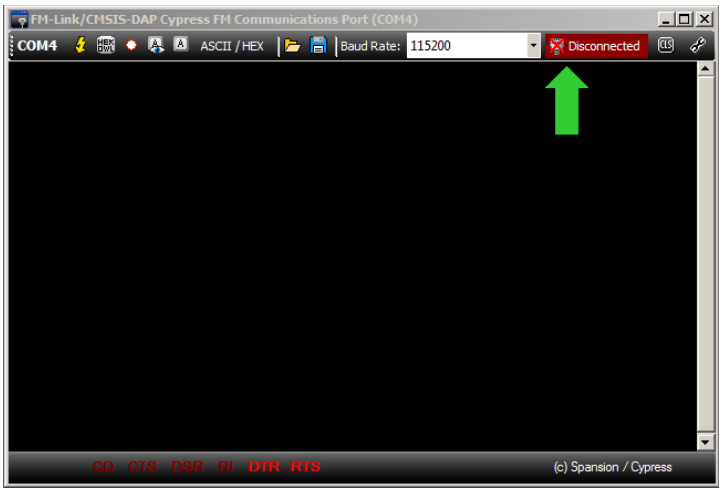
No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.1.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using USB a cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   `IAR project: <User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_adc\IAR\s6e1b8_adc.eww.`
   `Keil project: <User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_adc\ARM \s6e1b8_adc.uvprojx.`
3. Build the project and download the code into the S6E1B8 device.

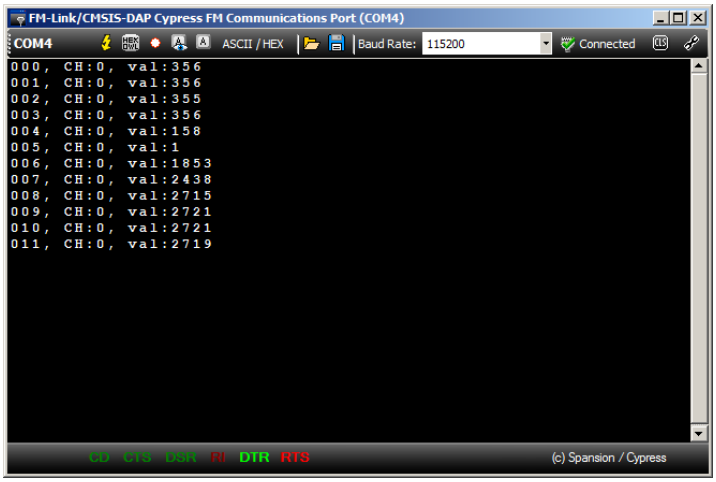
4. Run the Serial Port Viewer, set the baud rate as **115200**, and click the **Disconnected** button to connect the board with PC, refer to section Run the Test Demo**.**

Figure 4-1: Select the Baud Rate



5. Run the program and the ADC value will display in the Serial Port Viewer window.

Figure 4-2: ADC value



6. Turn the potentiometer (RP1), the ADC value will change accordingly.

4.2.2 Base Timer

4.2.2.1 Project Description

This project demonstrates the base timer operation of the S6E1B8 device. The program configures base timer 4 in PWM mode to generate a PWM sequence. The sequence outputs from the TIOA4_1 pin to drive the green LED of the RGB LED (LED4). The duty cycle of the PWM is varied every 2 ms by base timer 2 which results in a breathing effect on the LED.

4.2.2.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.2.3 Verify Output

1. Power the FM0+ S6E1B-Series Pioneer board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   `IAR project: <User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_bt\IAR\s6e1b8_bt.eww.`
   `Keil project: <User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_bt\ARM \s6e1b8_bt.uvprojx.`
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will glow green with a breathing effect.

4.2.3 Descriptor System Data Transfer Controller (DSTC)

4.2.3.1 Project Description

This project demonstrates the Descriptor System Data Transfer Controller (DSTC) operation of the S6E1B8 device. The program configures DSTC to move the data in *au32SourceData* (source array) to *au32DestinationData* (destination array), and then compares the content of the arrays to verify the data.

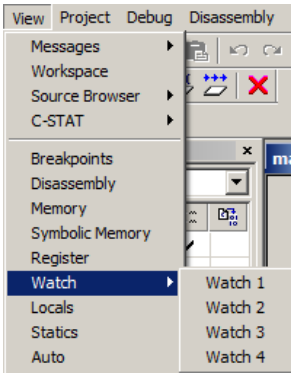
4.2.3.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.3.3 Verify Output

4.2.3.3.1 Using the project in IAR Embedded Workbench

1. Power the FM0+ S6E1B8-Series Pioneer board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_dstc\IAR\s6e1b8_dstc.eww.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Open Watch1 window from **View** > **Watch**.



5. Add the arrays m_au32SourceDataand m_au32DestinationData in Watch1 window.

Watch 1

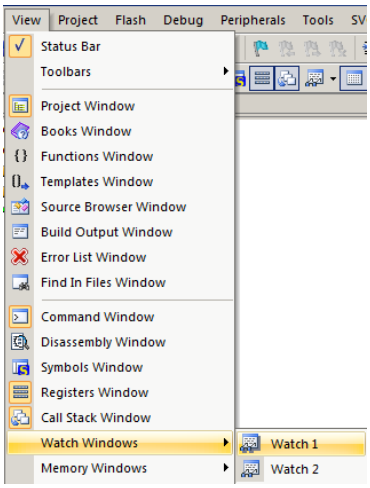
Expression	Value	Location	Type
⊞ m_au32SourceD...	<array>	0x20000000	uint32_t[16]
⊞ m_au32Destinati...	<array>	0x20000040	uint32_t[16]
<click to edit>			

6. Run the program for a while (>10 seconds).
7. Stop the program and check the arrays mentioned above. The Program Counter (PC) will stop at the routine as shown below which means the content of the arrays are the same.

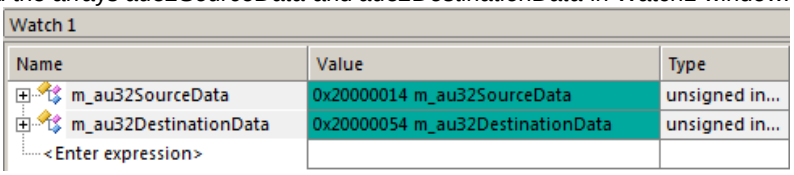
```
    if (TRUE == bCompareError) // Should never happen ...
    {
        while(1)
        {}
    }

    while(1)
    {}
}
```

4.2.3.3.2 Using the project in Keil µVision IDE

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in Keil µVision IDE from the following directory on your PC:
   `Keil project: <User_Directory>:\FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_dstc\ARM \s6e1b8_dstc.uvprojx.`
3. Build the project and download the code into the S6E1B8 device.
4. Open Watch1 window from **View** > **Watch Windows**.



5. Add the arrays *au32SourceData* and *au32DestinationData* in Watch1 window.



6. Run the program for a while (>10 seconds).
7. Stop the program and check the arrays mentioned above. The Program Counter (PC) will stop at the routine as shown below which means the content of the arrays are the same.

```
206     if (TRUE == bCompareError) // Should never happen ...
207     {
208         while(1)
209         {}
210     }
211
212     while(1)
213     {}
214 }
```

4.2.4 External Interrupt

4.2.4.1 Project Description

This project demonstrates the external interrupt operation of the S6E1B8 device. SW2 is connected to the non-maskable external interrupt (INT16_0). Pressing the SW2 button on the board will change the color of the RGB LED (LED4) from red to green to blue.

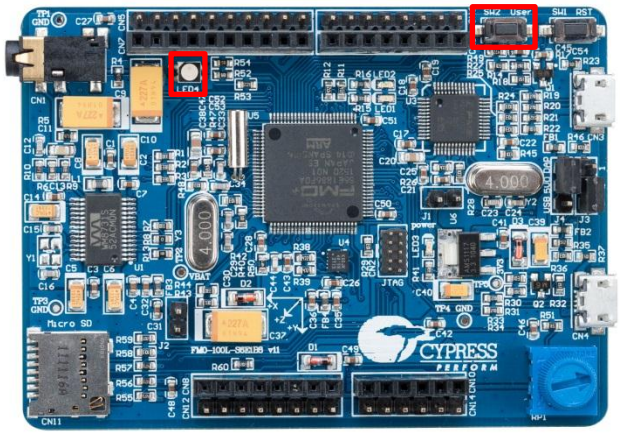
4.2.4.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.4.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_ext_int\IAR\s6e1b8_ext_int.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_ext_int\ARM\s6e1b8_ext_int.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. Press the SW2 button to change the color of the RGB LED (LED4).

Figure 4-3: SW2 and RGB LED



4.2.5 Flash

4.2.5.1 Project Description

This project demonstrates the flash writing operation of the S6E1B8 device. A specific set of four values each of four bytes in size will be written into a specific address location 0x0007C000 in the flash memory.

4.2.5.2 Hardware Connection

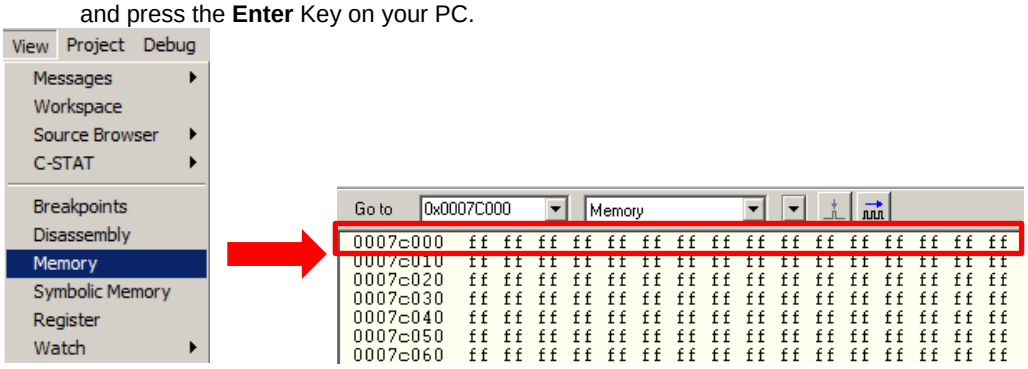
No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.5.3 Verify Output

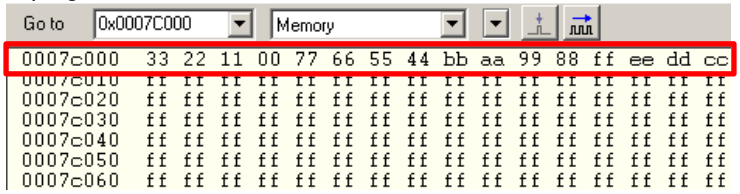
4.2.5.3.1 Using the project in IAR Embedded Workbench

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.

2. Open the project file in IAR Embedded Workbench from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_flash\IAR\s6e1b8_flash.eww.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Open the memory window from **View** > **Memory**. Enter 0x0007C000 in the **Go to** table and press the **Enter** Key on your PC.

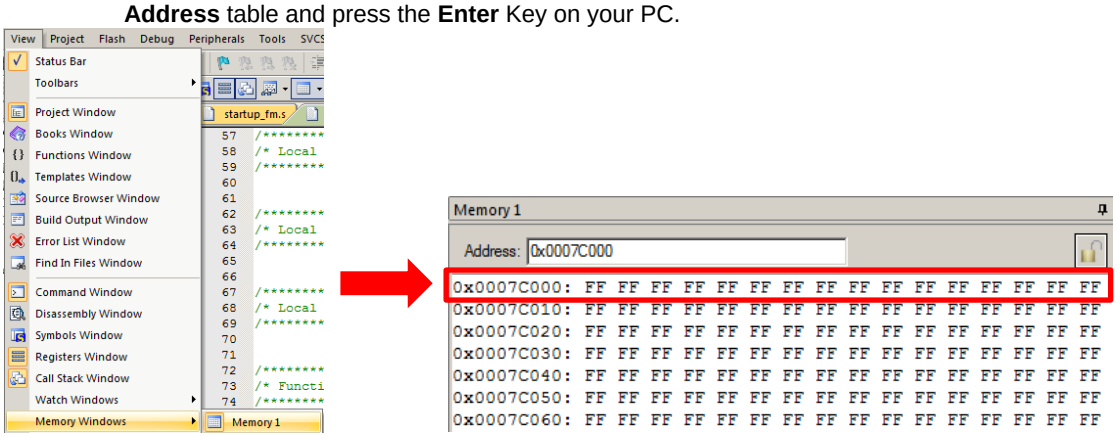


5. Run the program for a while (>10 seconds).
6. Stop the program and check the content of 0x0007C000 in flash.



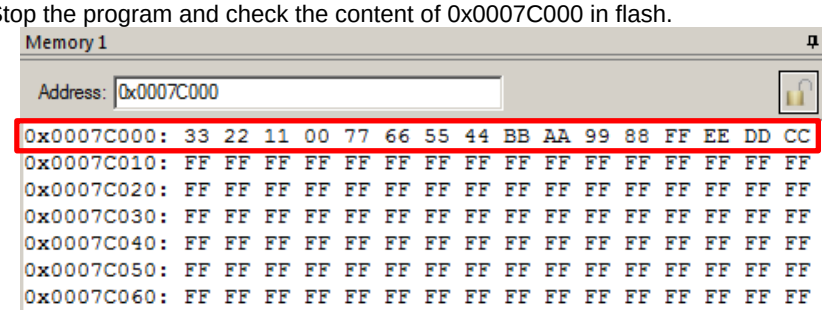
4.2.5.3.2 *Using the project in Keil µVision IDE*

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in Keil µVision IDE from the following directory on your PC:
   ```
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_flash\ARM\s6e1b8_flash.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Open the Memory1 window from **View** > **Memory Windows**. Enter 0x0007C000 in the **Address** table and press the **Enter** Key on your PC.



5. Run the program for a while (>10 seconds).

6. Stop the program and check the content of 0x0007C000 in flash.



4.2.6 GPIO

4.2.6.1 Project Description

This project demonstrates the GPIO operations of the S6E1B8 device by blinking an LED. The P3E pin drives the green LED of LED4. When program runs, the P3E pin outputs a sequence to drive the green LED of LED4. The RGB LED (LED4) will blink with green color accordingly.

4.2.6.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.6.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_gpio\IAR\s6e1b8_gpio.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_gpio\ARM\s6e1b8_gpio.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will blink with green color.

4.2.7 UART Communication

4.2.7.1 Project Description

This project demonstrates the UART communication of the S6E1B8 device. This program enables the MFS0 as UART mode to communicate with the CMSIS-DAP device. The CMSIS-DAP device serves as the bridge between the MCU and PC.

4.2.7.2 Hardware Connection

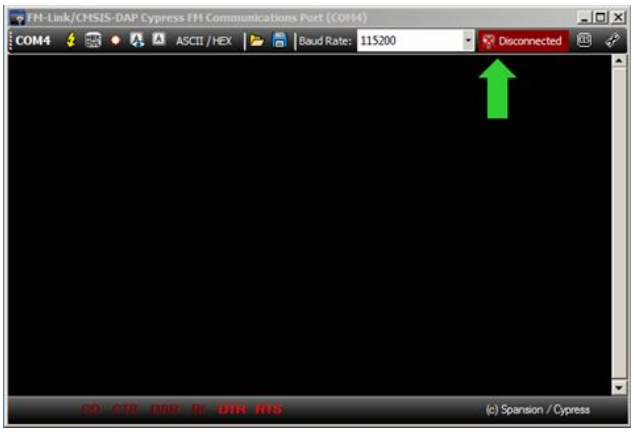
No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.7.3 Verify Output

1. Power the FM0+ S6E1B-Series Pioneer board from CN3 using a USB cable, refer to Figure 2-5.

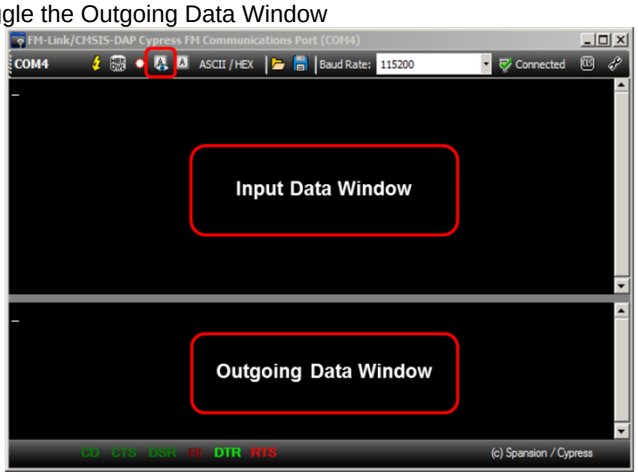
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_mfs_uart\IAR\s6e1b8_mfs_uart.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_mfs_uart\ARM\s6e1b8_mfs_uart.uvprojx.
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. Run the Serial Port Viewer tool, set the baud rate as **115200**, and click the **Disconnected** button to connect the board with PC, refer to section Run the Test Demo.

Figure 4-4: Select the Baud Rate



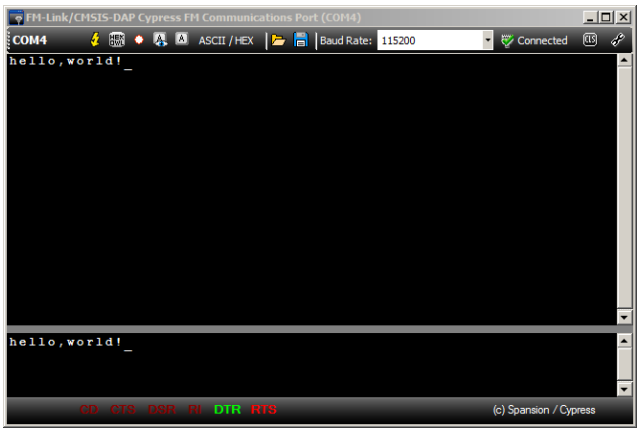
6. Click the Toggle Outgoing Data Window icon.

Figure 4-5: Toggle the Outgoing Data Window



7. Key in any characters in the Outgoing Data Window, the same characters will be echoed in the Input Data Window.

Figure 4-6: Echo Test

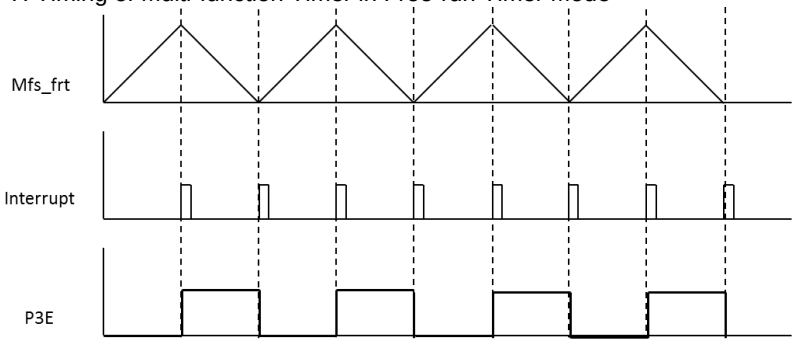


4.2.8 Multi-function Timer

4.2.8.1 Project Description

This project demonstrates the multi-function timer (MFT) operation of the S6E1B8 device. This project configures the multi-function timer unit0 in free-run timer mode. An interrupt will occur at the peak point and zero point of the counter respectively. The state of pin P3E will be changed in the interrupt service routines.

Figure 4-7: Timing of Multi-function Timer in Free-run Timer Mode



4.2.8.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.8.3 Verify Output

1. Power the FM0+ S6E1C-Series Pioneer board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_mft_frt\IAR\s6e1b8_mft_frt.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_mft_frt\ARM\s6e1b8_mft_frt.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will blink with green color.

4.2.9 RTC calendar

4.2.9.1 Project Description

This project demonstrates the RTC operation of the S6E1B8 device. The program enables the RTC in calendar mode, and sends out the current calendar data through UART0.The calendar starts from 2015/9/13 23:59:01 Wednesday.

4.2.9.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

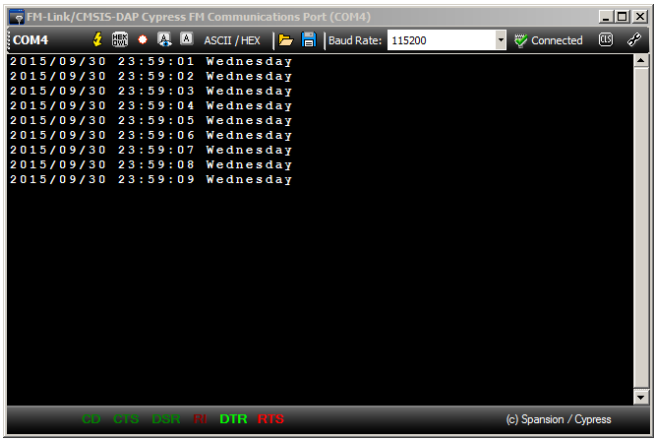
4.2.9.3 Verify Output

1. Power the FM0+ S6E1B-Series Pioneer board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_rtc\IAR\s6e1b8_rtc.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo
   Projects\s6e1b8_rtc\ARM\s6e1b8_rtc.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Run the Serial Port Viewer, set the baud rate as **115200**, and click the **Disconnected** button to connect the board with PC, refer to section Run the Test Demo.

Figure 4-8: Select the Baud Rate

5. Run the program.
6. The calendar data will be displayed in the Serial Port Viewer window.

Figure 4-9: Calendar



4.2.10 Sleep Mode

4.2.10.1 Project Description

This project demonstrates the sleep mode operation of the S6E1B8 device. The MCU will enter sleep mode after blinking the RGB LED (LED4) 5 times (green color). It can be woken up by pressing SW2 key, and then the RGB LED (LED4) will glow with green color.

4.2.10.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.10.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   ```
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo
   Projects\s6e1b8_sleep_mode\IAR\s6e1b8_sleep_mode.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware
   \Demo Projects\s6e1b8_sleep_mode\ARM
   \s6e1b8_sleep_mode.uvprojx.
   ```
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will blink 5 times (green color), and then the MCU enters sleep mode.
6. Press the SW2 key to wake up the MCU.
7. The RGB LED (LED4) will glow green.

4.2.11 Software Watchdog

4.2.11.1 Project Description

This project demonstrates the software watchdog operation of the S6E1B8 device. This program will show the effect of feeding the software watchdog or not feeding the software watchdog when it is enabled.

If the software watchdog is enabled and fed in time, the program will run normally, and the RGB LED (LED4) will blink green.

If the software watchdog was enabled but not fed in time, the S6E1B8 device will reset, and the RGB LED (LED4) will be glow green after reset.

4.2.11.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.11.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.

2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_st_wdt\IAR\s6e1b8_st_wdt.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_st_wdt\ARM\s6e1b8_st_wdt.uvprojx.
3. Build the project and download the code into S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will blink with green color.
6. Stop the program, comment out the line *Swwdg_Feed();* in main.c, and click **File** > **Save** .

```
static void WdgSwCallback(void)
{
    //Shield "Swwdg_Feed()"
    //Swwdg_Feed();   // Clear Irq and Reset Timer
    ++u8CountWdg;
    if (TRUE != FM_SWWDT->WDOGRIS_f.RIS) // If the watchdog interrupt flag had
    {
        SetLed(u8CountWdg);
    }
}
```

7. Repeat steps 3 and 4.
8. The RGB LED (LED4) will glow with green color.

4.2.12 Watch Timer

4.2.12.1 Project Description

This project demonstrates the watch timer operation of the S6E1B8 device. The watch timer will generate an interrupt once per second. In the interrupt service routine, P3E will be toggled which makes the RGB LED (LED4) blink with green color.

4.2.12.2 Hardware Connection

No specific hardware connections are required for this project. All connections are hardwired on the board.

4.2.12.3 Verify Output

1. Power the FM0+ S6E1B-Series Starter board from CN3 using a USB cable, refer to Figure 2-5.
2. Open the project file in IAR Embedded Workbench or Keil µVision IDE from the following directory on your PC:
   IAR project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_wc\IAR\s6e1b8_wc.eww.
   Keil project: <User_Directory>:
   \FM0+ S6E1B-Series Starter Kit_Ver01\Firmware\Demo Projects\s6e1b8_wc\ARM\s6e1b8_wc.uvprojx.
3. Build the project and download the code into the S6E1B8 device.
4. Run the program.
5. The RGB LED (LED4) will blink with green color.

4.3 Flash Programming

Most IDEs, including IAR and Keil µVision, are capable of programming the embedded flash. If that option is not desirable for some reason, the following section shows you how to program the embedded flash using either a serial or a USB connection. It also shows how to re-program the firmware in the CMSIS-DAP device should it be necessary.

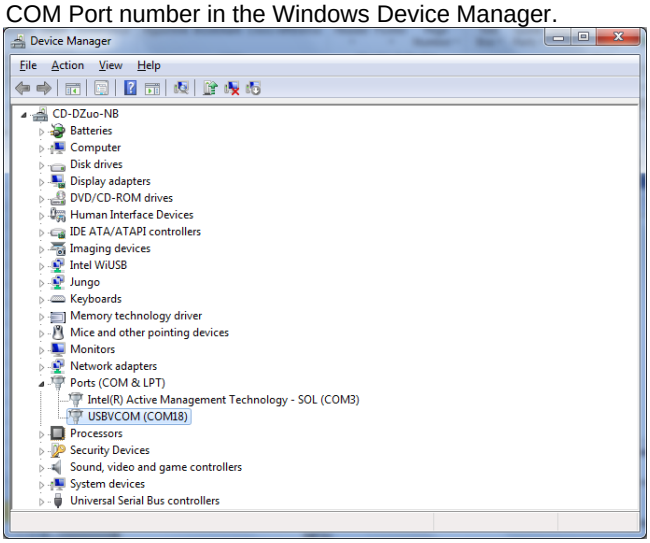
4.3.1 Programming S6E1B8 Using FLASH USB DIRECT Programmer

1. Install the FM0+ S6E1B-Series Starter Kit installer as per section Install Software. The FLASH USB DIRECT Programmer gets installed in your PC as part of kit installer.
2. Make sure the jumpers on the FM0+ S6E1B-Series Starter board are placed according to Table 4-4:

Table 4-4: Programming Jumper Settings

Jumper	Position	Description
J1	Open	Sets MB9AF312K (CMSIS-DAP) in run mode.
J2	Closed	Sets S6E1B8 in programming mode
J3	Pin 2 to Pin 3	Sets USB programming mode
J4	Pin 2 to Pin 3	Power from USB port of S6E1B8 (CN4)

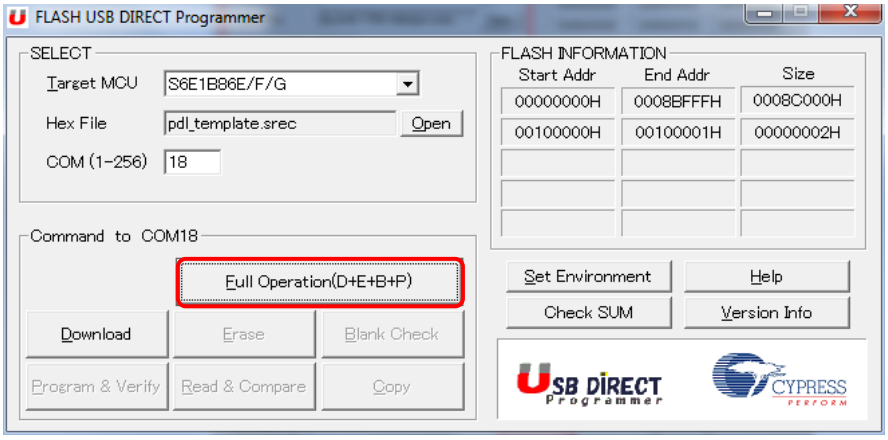
3. Connect the USB cable to the CN4 port.
4. Observe that Power LED (LED3) is glowing green.
5. Check the COM Port number in the Windows Device Manager.



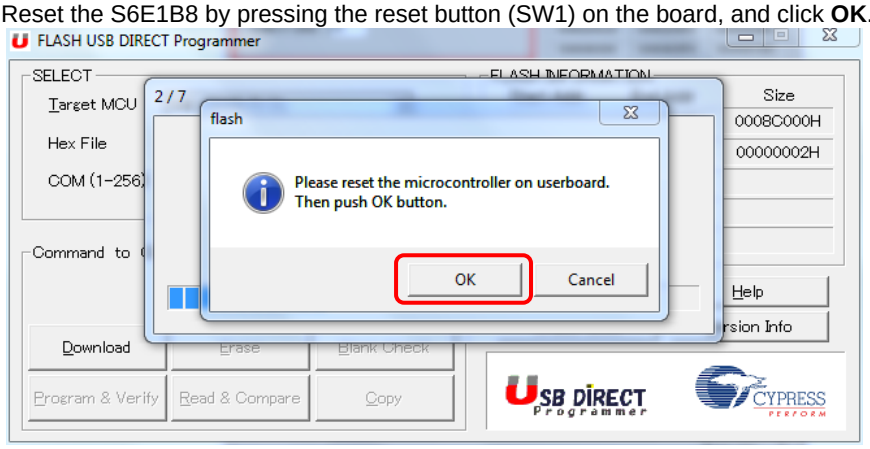
6. Launch the FLASH USB DIRECT Programmer from Windows

   Start Menu > **All Programs** > **Cypress** > **FLASH USB DIRECT Programmer** > **USBDirect**
7. Select “Target MCU” to S6E1B86E/F/G.
8. Select the Motorola-S format file or Intel-HEX format file to be programmed to the FLASH memory in the MCU.

   Note: The **Test Demo firmware** HEX File selected in this example.
9. Enter the Virtual COM Port listed in the Ports from the Device Manager in the “COM” box.
10. Click on **Full Operation (D+E+B+P)** button to start programming.



11. Reset the S6E1B8 by pressing the reset button (SW1) on the board, and click **OK**.



Note: Please click on **Help** for any issues or errors encountered during programming.

4.3.2 Programming S6E1B8 Using FLASH MCU Programmer

1. Install the FM0+ S6E1B8-Series Starter Kit installer per section Install Software.
2. Make sure the jumpers on the FM0+ S6E1B8-Series Pioneer board are placed according to the Table 4-5.

Table 4-5: Jumper Settings for S6E1B8 programming using FLASH MCU Programmer

Jumper	Position	Reasoning
J1	Open	Sets MB9AF312K (CMSIS-DAP) in run mode.
J2	Closed	Sets S6E1B8 in programming mode.
J3	Pin 1 to Pin 2	Sets for UART programming mode.
J4	Pin 1 to Pin 2	Power from USB port of CMSIS-DAP

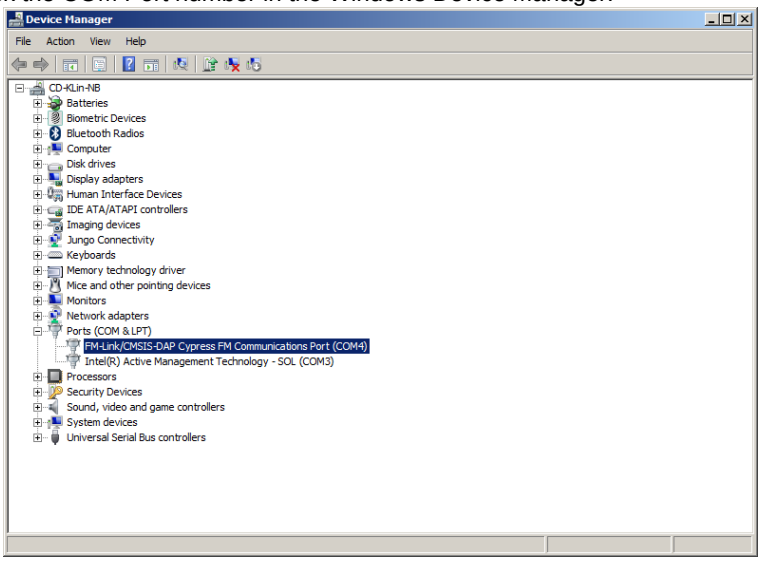
3. Connect the USB cable to the CN3 port.
4. Observe that Power LED (LED3) is glowing green.
5. Launch the FLASH MCU Programmer from Windows

   Start Menu > **All Programs** > **Cypress** > **FLASH MCU Programmer** > **FM0+ FM3 FM4**
6. Select “Target MCU” to **S6E1B86E/F/J**.
7. Select “Crystal Frequency” to 4 MHz.

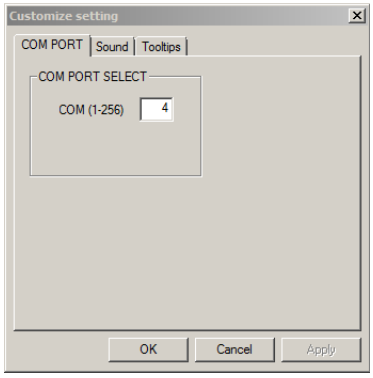
8. Select the Motorola-S format file or Intel-HEX format file to be programmed to FLASH memory in the MCU.

   Note: The HEX File selected in this example is the Test Demo firmware.

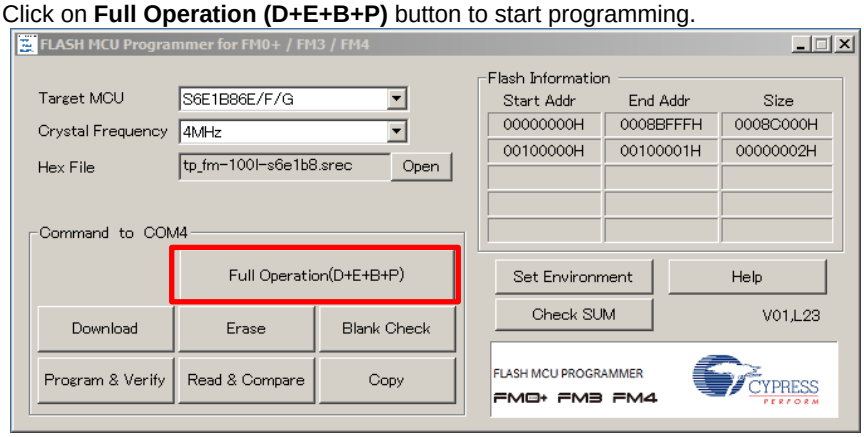
9. Check the COM Port number in the Windows Device Manager.



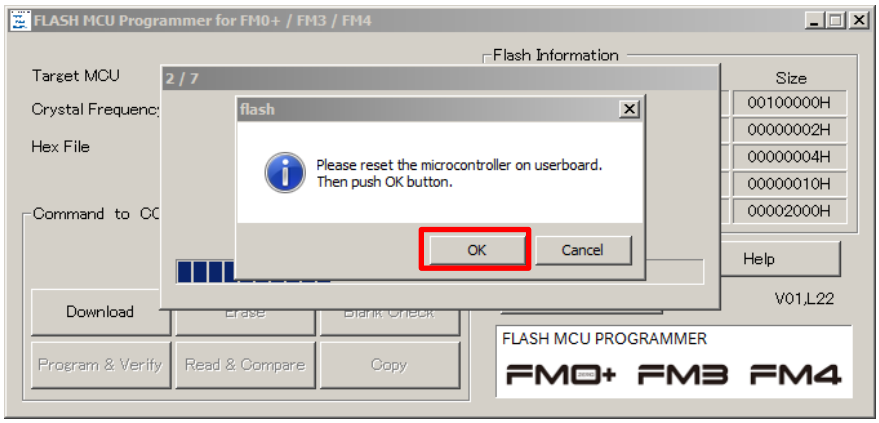
10. Click the “Set Environment” button and enter the Virtual COM Port listed in the Ports from the Device Manager in the “COM” box.



11. Click on **Full Operation (D+E+B+P)** button to start programming.



12. Reset the S6E1B8 by pressing the reset button (SW1) on the board, and click **OK**.



Note: Please click on **Help** for any issues or errors encountered during programming.

4.3.3 Programming CMSIS-DAP Using FLASH USB DIRECT Programmer

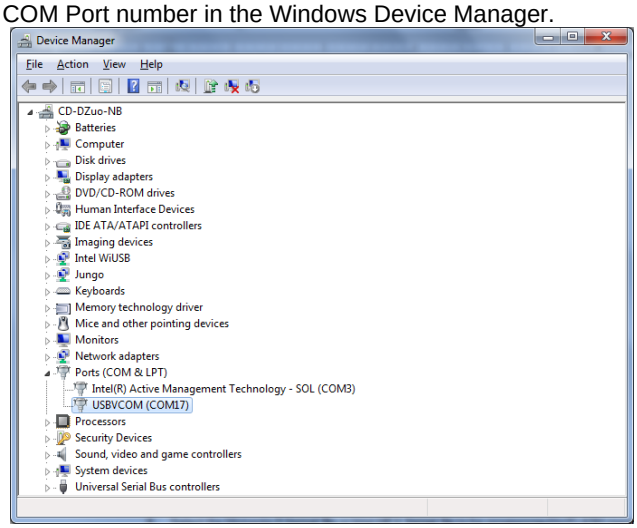
By default, the latest CMSIS-DAP firmware is programmed on the MB9AF312K. It is not normally required for the user to re-program this device before running the CMSIS-DAP debugger. Follow the steps below to update the firmware if needed.

1. Install the FM0+ S6E1B-Series Starter Kit installer per section Install Software.
2. Make sure the jumpers on the FM0+ S6E1B-Series Starter board is placed according to the Table 4-6.

Table 4-6: Programming Jumper Settings

Jumper	Position	Description
J1	Closed	Sets MB9AF312K (CMSIS-DAP) in programming mode.
J2	Open	Sets S6E1B8 in run mode.
J4	Pin 1 to Pin 2	Power from CMSIS-DAP (CN3)

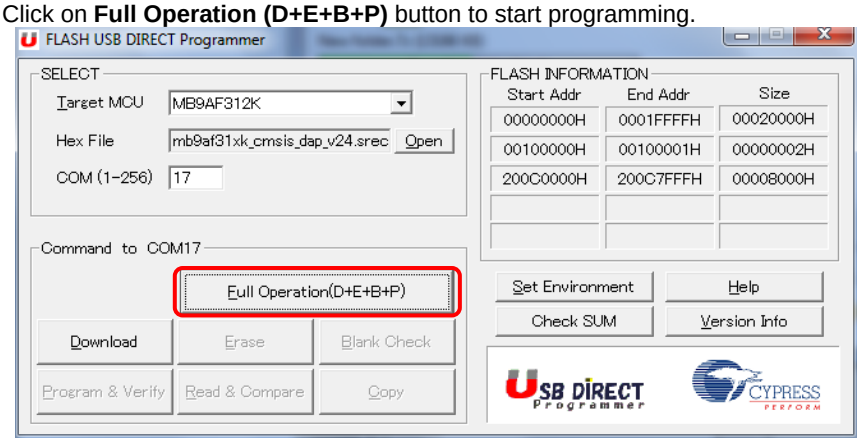
3. Connect the USB cable to the CN3 port.
4. Observe that Power LED (LED3) is glowing green.
5. Check the COM Port number in the Windows Device Manager.



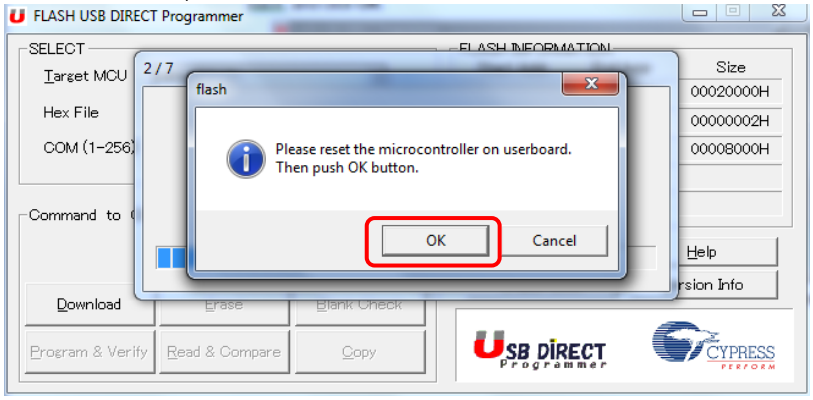
6. Launch the FLASH USB DIRECT Programmer from Windows

Start Menu > **All Programs** > **Cypress** > **FLASH USB DIRECT Programmer** > **USBDirect**

7. Select “Target MCU” to **MB9AF312K**.
8. Select the Motorola-S format file or Intel-HEX format file to be programmed to FLASH memory in the MCU. The HEX file is included in the following directory:
   `<Install_Directory>:\FM0+ S6E1B-Series Starter Kit \<version>\Firmware\CMSIS-DAP`
9. Enter the Virtual COM Port listed in the Ports from the Device Manager in **COM** box.
10. Click on **Full Operation (D+E+B+P)** button to start programming.



11. Reset the CMSIS-DAP microcontroller by removing the USB cable and reconnecting the USB cable back, and click **OK**.



Note: Please click **Help** for any issues or errors encountered during programming.

A. Appendix

A.1 Schematic

Figure A-1. MCU

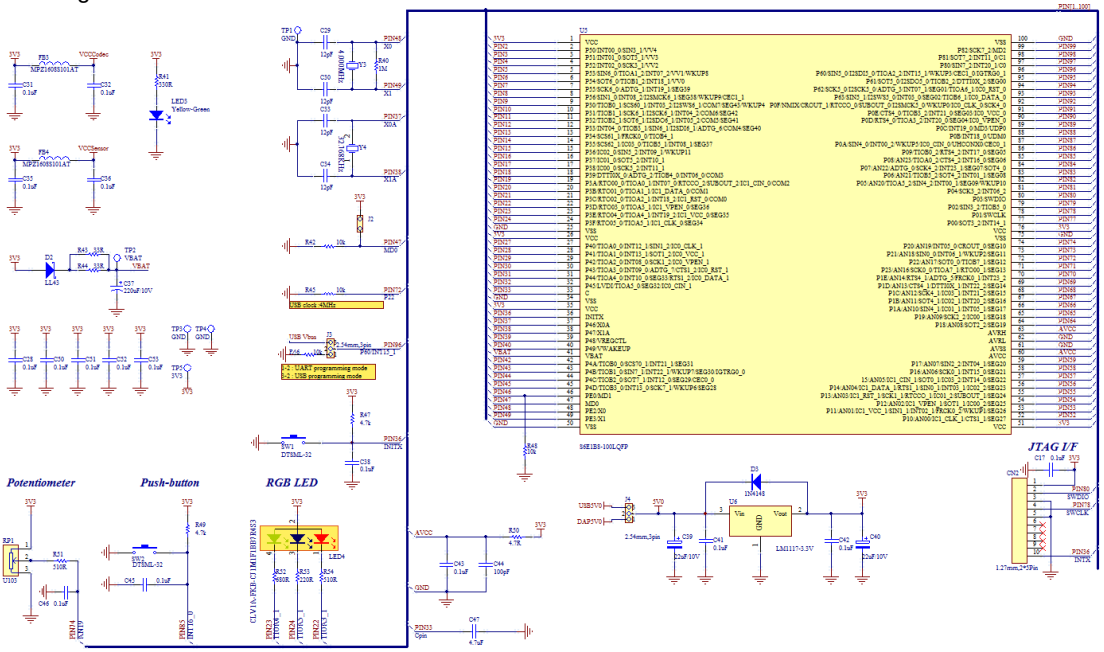


Figure A-2. CMSIS-DAP&USB Device

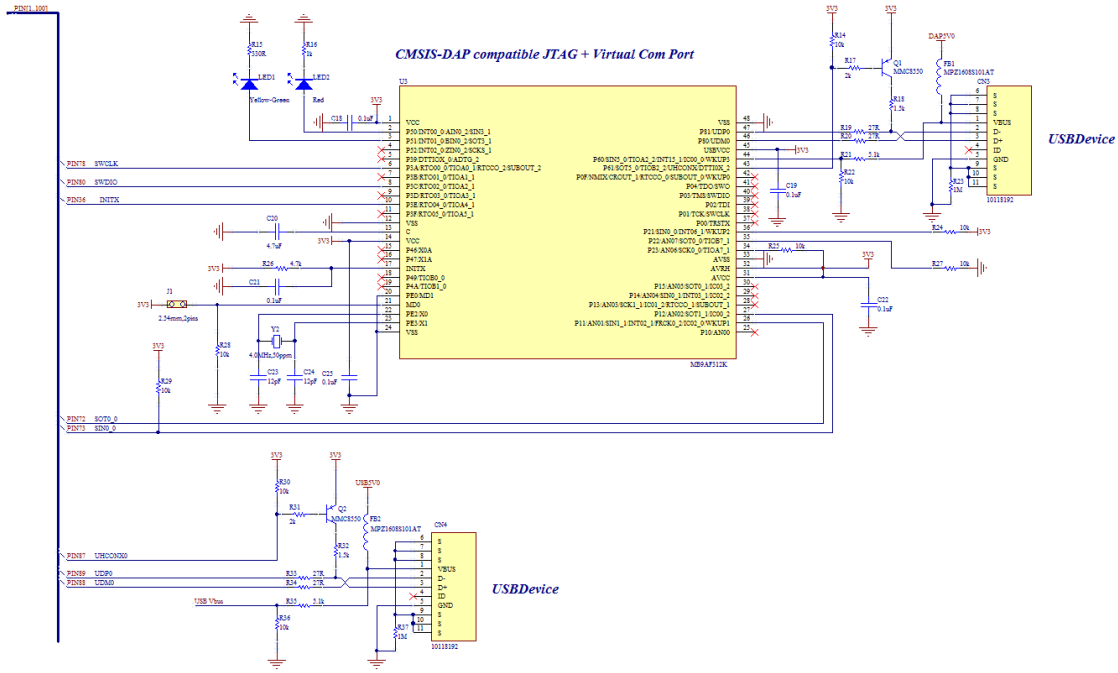


Figure A-3. Stereo Codec

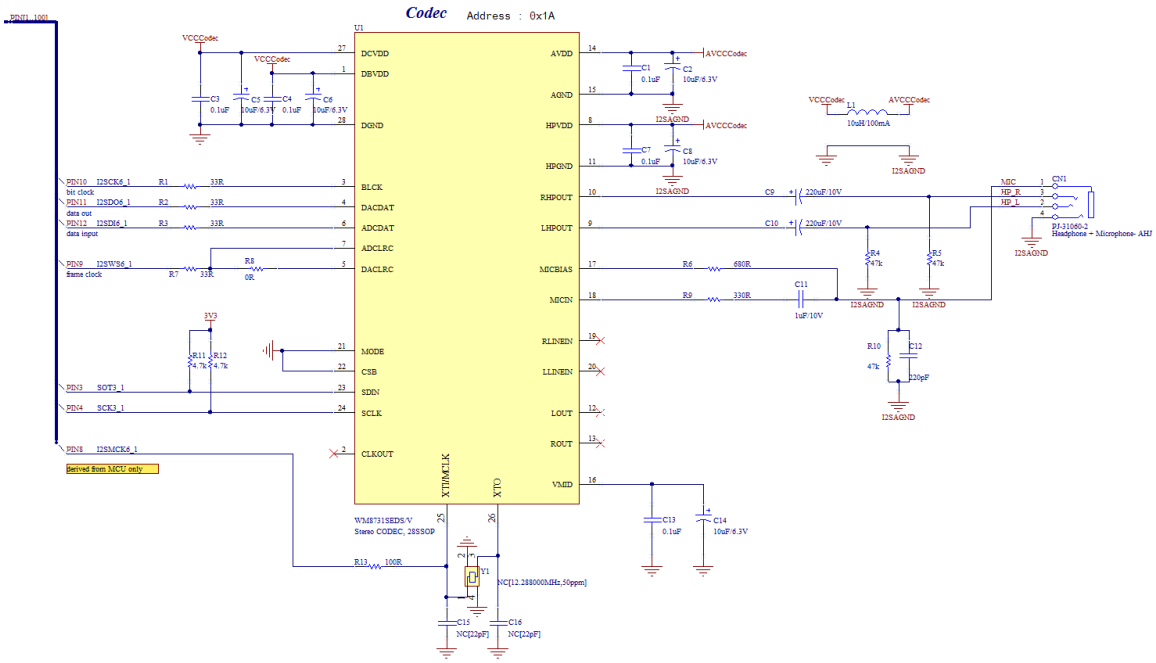
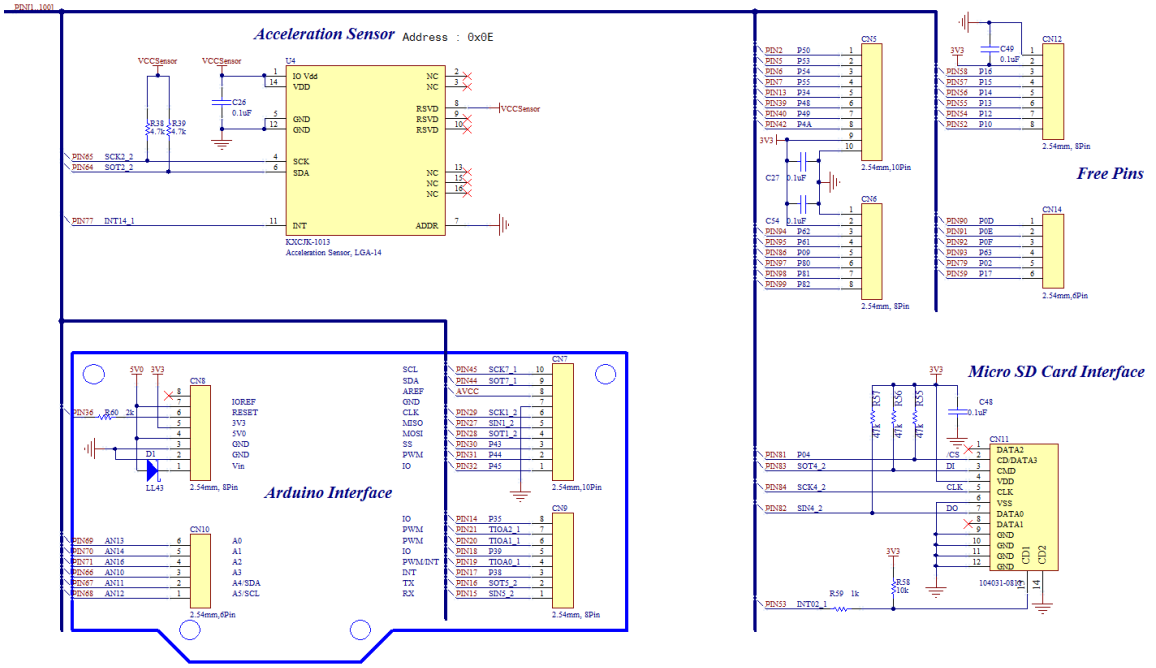


Figure A-4. Sensor & Interfaces



A.2 Bill of Materials

Item	Qty	Reference	Value	Description	Mfg	Mfg part number
1	31	C1, C3, C4, C7, C13, C17, C18, C19, C21, C22, C25, C26, C27, C28, C31, C32, C35, C36, C38, C41, C42, C43, C45, C46, C48, C49, C50, C51, C52, C53, C54	0.1uF	Ceramic Capacitor	YAGEO	CC0603KRX5R8BB104
2	2	C20, C47	4.7uF	Ceramic Capacitor	YAGEO	CC0805KKX5R8BB475
3	6	C23, C24, C29, C30, C33, C34	12pF	Ceramic Capacitor	YAGEO	CC0603JRNPO9BN120
4	2	C39, C40	22uF/10V	Tan Capacitor	AVX	TAJA226K010RNJ
5	5	C2, C5, C6, C8, C14	10uF/6.3V	Tan Capacitor	AVX	TAJA106K006RNJ
6	3	C9, C10, C37	220uF/10V	Tan Capacitor	AVX	TAJD227K010R
7	1	C11	1uF/10V	Ceramic Capacitor	Murata	GRM21BR71A105KA01L
8	1	C12	220pF	Ceramic Capacitor	'YAGEO	CC0603JRNPO9BN221
9	1	C44	100pF	Ceramic Capacitor	'YAGEO	CC0603JRNPO9BN101
10	1	CN2	1.27mm,2*5Pin	JTAG I/F	AIMO	1415-1205CNGOS3.01.52.301
11	2	CN3, CN4	10118192AC	Micro USB-B type, FCI,	FCI	10118192-0001LF
12	1	CN1	PJ-31060-2	3.5mm, line in jack	AIMO	PJ-31060-2
13	1	CN7	2.54mm,10Pin	CONNECTOR,wafer,1*10 PIN,Pitch=2.54	AIMO	2285-0110ANGO01
14	2	CN8, CN9	2.54mm, 8Pin	CONNECTOR,wafer,1*8IN ,Pitch=2.54	AIMO	2285-0108ANGO01
15	1	CN10	2.54mm,6Pin	CONNECTOR,wafer,1*6PI N,Pitch=2.54	AIMO	2285-0106ANGO01
16	1	CN5	2.54mm,10Pin	CON,header,1*6PIN,180°, DIP,MALE	AIMO	1125-1110ANGOS11.5001
17	2	CN6,CN12	2.54mm, 8Pin	CON,header,1*6PIN,180°, DIP,MALE	AIMO	1125-1108ANGOS11.5001
18	1	CN14	2.54mm,6Pin	CON,header,1*6PIN,180°, DIP,MALE	AIMO	1125-1106ANGOS11.5001
19	1	CN11	104031-0811	Micro-SD card	Molex	104031-0811

Item	Qty	Reference	Value	Description	Mfg	Mfg part number
20	1	D3	DL4148	Recifier diode	MCC	DL4148
21	2	D1, D2	LL43	Socktety diode	Vishay	LL43-GS08
22	4	FB1, FB2, FB3, FB4	MPZ1608S101AT	Ferrite bead, 3A,100R@100MHz,TDK	TDK	MPZ1608S101AT
23	2	J1, J2	2.54mm,2pins	CONN,PIN HEADER, 1*2PIN,DIP,180°	AIMO	1225-1102ANGOS11.501
24	2	J3, J4	2.54mm,3pin	CONN,PIN HEADER, 1*3PIN,DIP,180°,	AIMO	1225-1103ANGOS11.501
25	1	L1	MLZ2012N100LT	Ferrite Bead,TDK	TDK	MLZ2012N100LT
26	2	LED1, LED3	Yellow-Green	LED,Yellow-Green	Everlight	19-21SYGC/S530-E3/TR8
27	1	LED2	Red	LED, red	Everlight	19-21SURC/S530-A5/TR8
28	1	LED4	CLV1A-FKB-CJ1M1F1BB7R4S3	RGB LED,SMD,4pin	Cree	CLV1A-FKB-CJ1M1F1BB7R4S3
29	2	Q1, Q2	MMC8550	PNP transistor	MCC	MMS8550-H-TP
30	14	R14, R22, R24, R25, R27, R28, R29, R30, R36, R42, R45, R46, R48, R58	10k	Resistor	YAGEO	RC0603FR-0710KL
31	3	R9,R41,R15	330R	Resistor	YAGEO	RC0603FR-07330RL
32	2	R16, R59	1k	Resistor	YAGEO	RC0603FR-071KL
33	3	R17, R31,R60	2k	Resistor	YAGEO	RC0603FR-072KL
34	2	R18, R32	1.5k	Resistor	YAGEO	RC0603FR-071K5L
35	4	R19, R20, R33, R34	27R	Resistor	YAGEO	RC0603FR-0727RL
36	2	R21, R35	5.1k	Resistor	YAGEO	RC0603FR-075K1L
37	3	R23, R37, R40	1M	Resistor	YAGEO	RC0603FR-071ML
38	7	R11, R12, R26, R38, R39, R47, R49	4.7k	Resistor	YAGEO	RC0603FR-074K7L
39	2	R51, R54	510R	Resistor	YAGEO	RC0603FR-07510RL
40	6	R1, R2, R3, R7,R43, R44	33R	Resistor	YAGEO	RC0603FR-0733RL
41	6	R4, R5,R10, R55, R56, R57	47k	Resistor	YAGEO	RC0603FR-0747KL
42	2	R6, R52	680R	Resistor	YAGEO	RC0603FR-07680RL
43	1	R8	0R	Resistor	YAGEO	RC0603FR-070RL

Item	Qty	Reference	Value	Description	Mfg	Mfg part number
44	1	R13	100R	Resistor	YAGEO	RC0603FR-07100RL
45	1	R50	4.7R	Resistor	YAGEO	RC0603FR-074R7L
46	1	R53	220R	Resistor	YAGEO	RC0603FR-07220RL
47	2	RP1	3386U-1-103T	Potentiometer	BURANS	3386U-1-103T
48	2	SW1, SW2	K2-1101ST-C4SA-01	6*35mm, Button, 2pin , SMT	HANRO	K2-1101ST-C4SA-01
49	1	U3	MB9AF312K	MB9AF314KPMC	Cypress	MB9AF314KPMC
50	1	U1	WM8731SEDS/V	Stereo codec	Wolfson	WM8731SEDS/RV
51	1	U4	KXCJK-1013	Acceleration Sensor, Rohm, LGA-14	Rohm	KXCJK-1013
52	1	U5	S6E1B86F0A	MCU, Cypress	Cypress	S6E1B86F0AGV20000
53	1	U6	LM1117-3.3V	LDO	TI	LM1117IMPX-3.3/NOPB
54	1	Y2, Y3	4 MHz,50ppm	Crystal Oscillator	Wisdom	QRS-4M00A5020B
55	1	Y4	32.768KHz	Crystal Oscillator	Wisdom	QRA-32768A20125B

Revision History

Document Revision History

Document Title: FM0+ S6E1B-Series Starter Kit Guide Document Number: 002-10654				
Revision	**ECN Number**	**Issue Date**	**Origin of Change**	**Description of Change**
**	5152938	03/09/2016	CCTA	Initial revision.
*A	5179521	03/24/2016	CCTA	Updated Document Title to “FM0+ S6E1B-Series Starter Kit Guide”. Replaced phototransistor with potentiometer across the document. Updated Introduction: Updated description. Updated Installation and Test Operation: Updated Install Software: Updated description. Added Figure 2-3. Updated Test Operation: Updated Run the Test Demo (Updated description). Updated Test Procedure Explanation (Updated description). Updated Hardware: Updated Hardware Details: Updated User Button and LED: Updated Table 3-1 (Renamed “Port” with “Pin Name” in column heading and updated details in the same column). Updated Micro SD Card (Updated description). Updated CMSIS-DAP (Updated description). Updated Software Development: Updated Tool Options (Updated description). Updated Example Project: Updated Table 4-3 (Updated details in “Title/Description” column of S6E1B8_bt_pwm and S6E1B8_flash). Updated Flash: Updated Project Description (Updated description). Updated Flash Programming: Updated Programming S6E1B8 Using FLASH USB DIRECT Programmer (Updated description).
*B	5713151	04/26/2017	SHEA	Updated logo and copyright