AN4691

Getting Started with MCC and Soteria-G3

Author: *Shreyas Kannan*
Microchip Technology Inc.

1.0 INTRODUCTION

Soteria-G3 is a firmware design executed on the CEC173x family of devices. It can be used in conjunction with any application processor (AP) that boots out of an external SPI Flash device to extend the Root-of-Trust (RoT) and enforce a secure boot process in the system.

Soteria-G3 uses the CEC173x immutable secure bootloader, implemented in ROM, as the system RoT. The CEC173x secure bootloader loads, decrypts, and authenticates the embedded controller firmware from the external (or internal) SPI Flash device. The validated Soteria-G3 that runs on the CEC173x is designed to subsequently authenticate the application processor firmware (AP_FW) located in the same SPI Flash component and up to three additional SPI Flash components.

Soteria-G3 prevents the system from booting unless the AP_FW stored in the external SPI Flash device is authentic code signed by the original equipment manufacturer (OEM). It offers security features to authenticate the SPI Flash image in the external SPI Flash device.

The validated AP_FW that runs on the application processor can utilize crypto resources in the CEC173x to authenticate other code in the system, thereby extending the Chain-of-Trust (CoT) to ensure that all code running in the system is authorized.

Soteria-G3 also supports secure firmware updates, which can authenticate updates to both AP_FW and Soteria-G3 in the system.

This application note provides details on how to use MPLAB® Code Configurator (MCC) with the CEC173x part and use Soteria-G3 secure-boot solution.

This document is limited to providing the user with a high-level overview of MCC, Soteria-G3, and getting started with using Soteria-G3 in CEC173x part.

1.1 Sections

This document includes the following topics:

1.2 References

Consult the following references for details on the specific parts referred to in this document:

- MPLAB® Code Configurator (MCC) Getting Started: https://microchipdeveloper.com/mcc:start

1.3 Pre-Requisites

- IDE – MPLABX IDE v6.00 or higher
- DFP – v1.5.142 or higher
- Debugger (only in case of debugging) – ICD4 or PICKit4
- Compiler – XC32 v4.00
- Board – CEC1736 development board with, (a) CEC173x internal Flash pre-programmed binary, and (b) external Flash modules with pre-programmed AP_FW binaries

1.4 Assumptions and Dependencies

The user is expected to have a fair idea of using MCC with any other Microchip microcontrollers.

1.5 Terms and Abbreviations

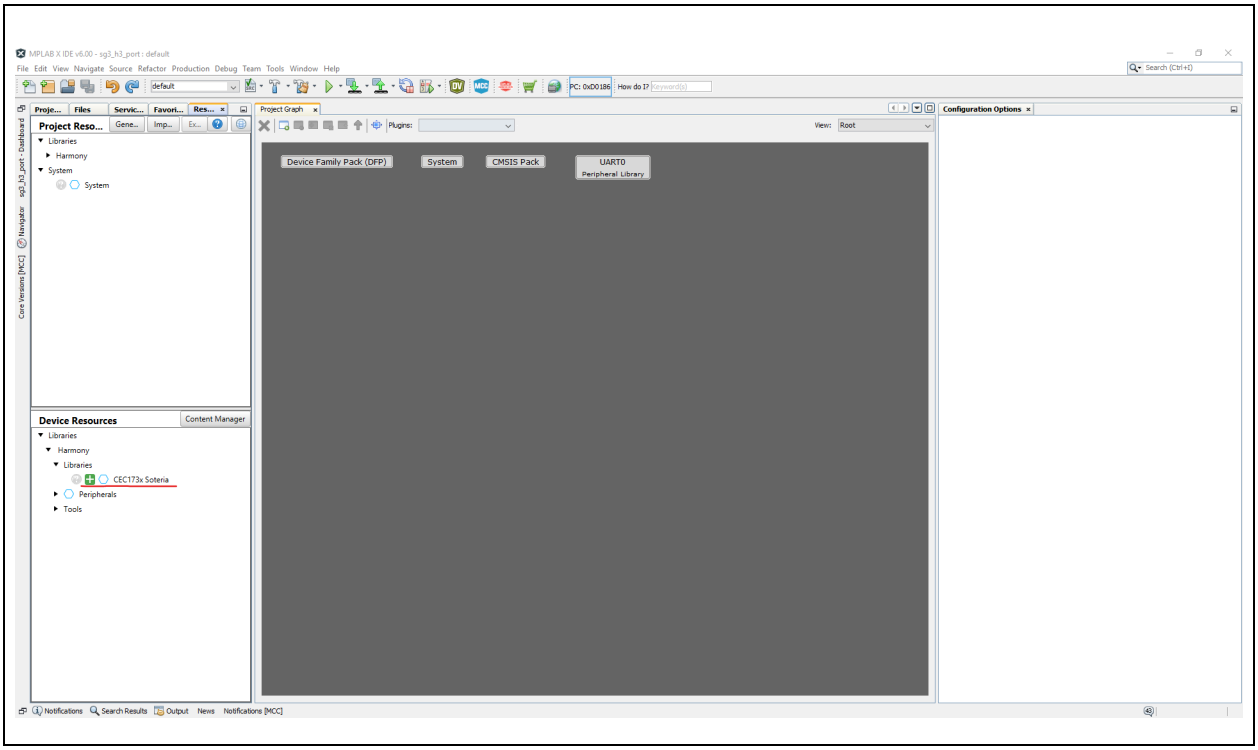
- **AP**: Application Processor
- **API**: Application Programming Interface
- **BSP**: Board Support Package
- **CoT**: Chain-of-Trust
- **ECIA**: Embedded Controller Interrupt Aggregator
- **GPIO**: General Purpose Input Output
- **HAL**: Hardware Abstraction Layer
- **Hex**: Hexadecimal
- **IRQ**: Interrupt Request
- **MCC**: Microchip MPLAB Code Configurator
- **OEM**: Original Equipment Manufacturer
- **PLIB**: Peripheral Library
- **RoT**: Root-of-Trust
- **SPI**: Serial Peripheral Interface
- **UART**: Universal Asynchronous Receiver and Transmitter

2.0 SETTING UP AN MCC PROJECT WITH SOTERIA-G3 LIBRARY

To set up an MCC project with Soteria-G3 library:

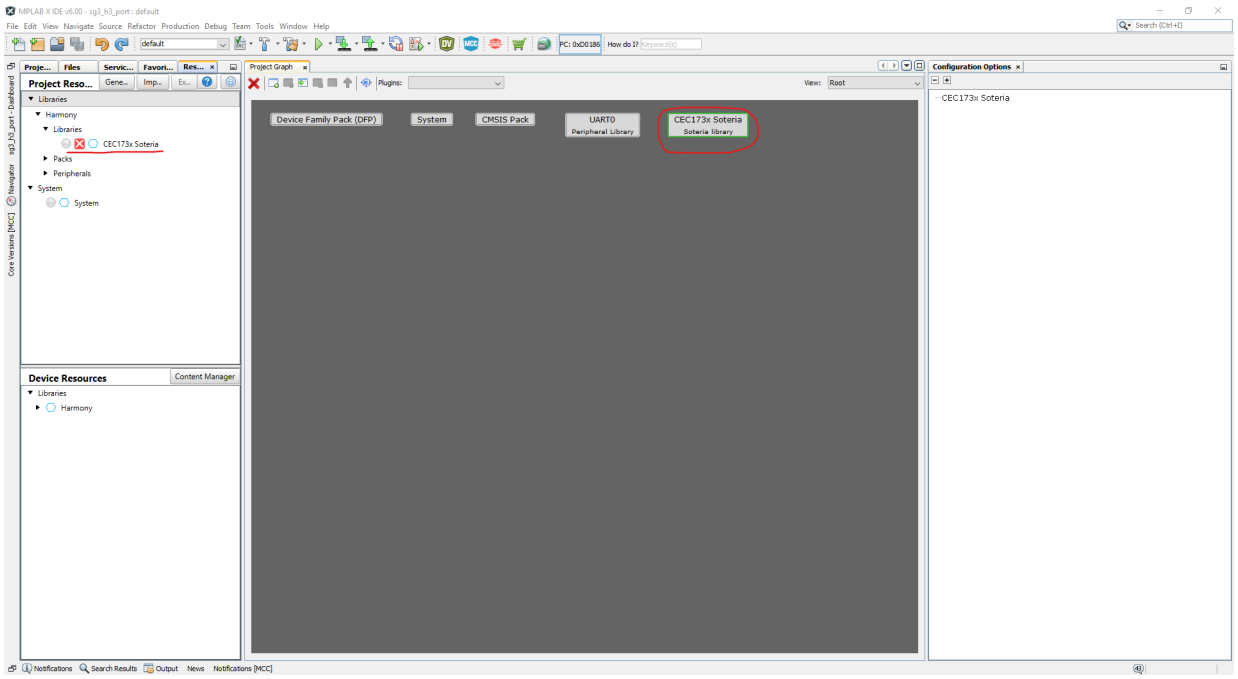
1. Create a new 32-bit MCC Harmony Project and select **CEC1736_S0_2ZW** as the target device.
2. Select and download the `cec173x_soteria_lib` component from the MCC content manager.
3. Add Soteria-G3 as a library into the created application project. Double-click the **CEC173x Soteria** component that can be found under *Device Resources>Libraries>Harmony>Libraries>CEC173x Soteria* (Figure 1).

FIGURE 1: ADDING SOTERIA-G3 AS LIBRARY



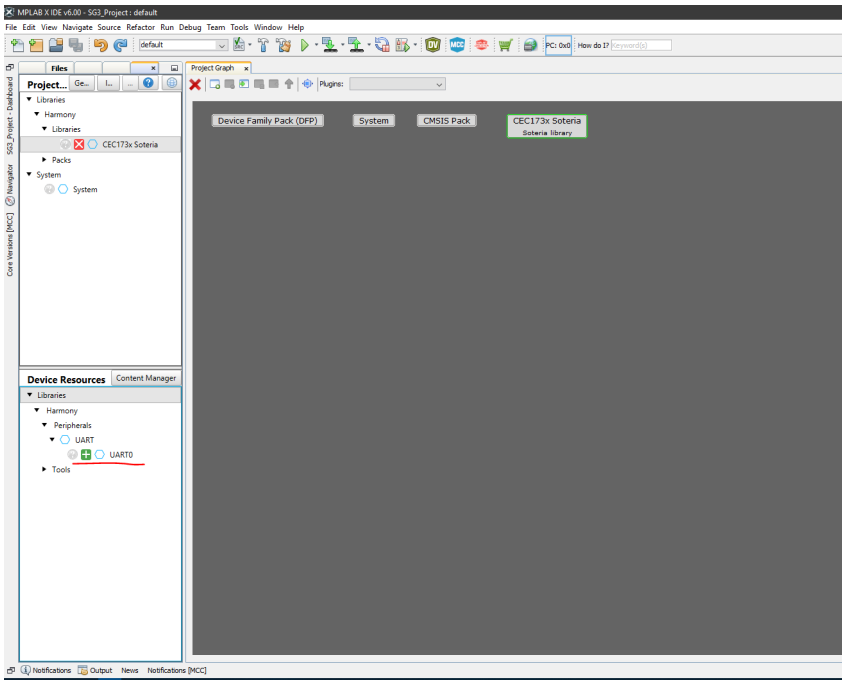
The Soteria-G3 library component is then added in the **Project Graph** and **Project Resources** tabs as shown in Figure 2.

FIGURE 2: SOTERIA-G3 LIBRARY ADDED IN PROJECT GRAPH AND PROJECT RESOURCES



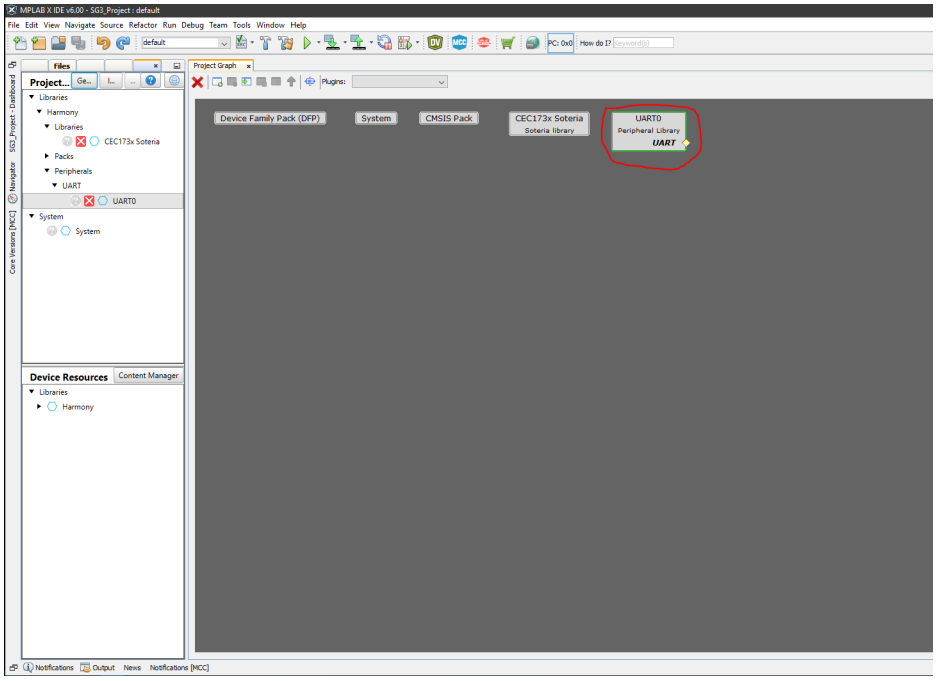
4. Add UART peripheral into the created application project. Double-click the **UART0** component that can be found under *Device Resources>Peripherals>UART>UART0* (Figure 3).

FIGURE 3: ADDING UART PERIPHERAL IN APPLICATION PROJECT



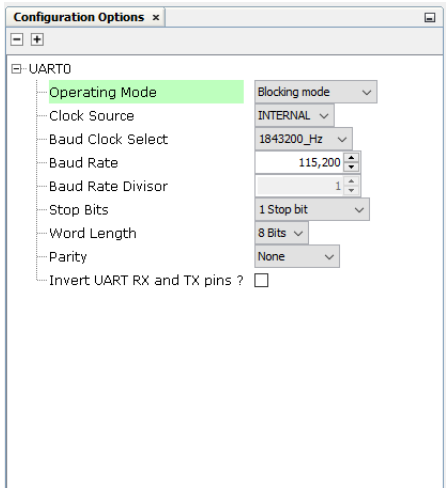
The UART peripheral component is then added in the **Project Graph** and **Project Resources** tabs as shown in Figure 4.

FIGURE 4: UART PERIPHERAL ADDED IN PROJECT GRAPH AND PROJECT RESOURCES



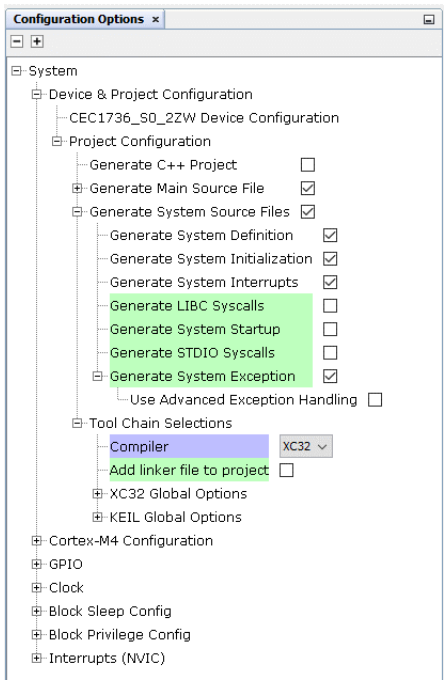
5. Change the UART0 configuration as shown in Figure 5.

FIGURE 5: CHANGING THE UART0 CONFIGURATION



6. Select the project configuration as shown in Figure 6.

FIGURE 6: SELECTING THE PROJECT CONFIGURATION



7. Go to *Plugins>Pin Configuration* located in the **Project Graph** tab (Figure 7) and change the pin configurations as shown in Figure 8.

FIGURE 7: PROJECT GRAPH>PLUGINS>PIN CONFIGURATION

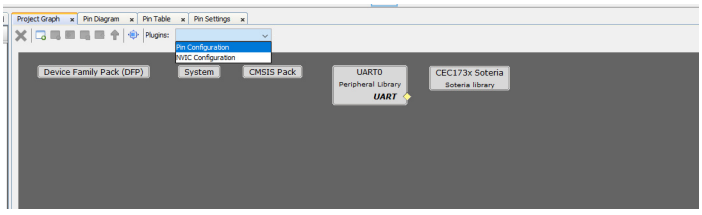


FIGURE 8: PIN CONFIGURATION

Pin Number	Pin ID	Custom Name	Function	Direction	Latch	Output Buffer	Polarity	PU/PD	Interrupt	Drive Strength	Slew Rate
A1	GPIO063	GPIO_GPIO063	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
A2	GPIO113	GPIO_GPIO113	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
A6	GPIO107	GPIO_GPIO107	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
A7	GPIO046	GPIO_GPIO046	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
B2	GPIO050	GPIO_GPIO050	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
B3	GPIO015	GPIO_GPIO015	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
B7	GPIO140	GPIO_GPIO140	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
C2	GPIO047	GPIO_GPIO047	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
F2	GPIO013	GPIO_GPIO013	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
F3	GPIO127	GPIO_GPIO127	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow
G2	GPIO201	GPIO_GPIO201	GPIO	In	n/a	Push Pull	Non-Inverted	None	FALLING_EDGE	Level0	Slow

8. Go to *Plugins>NVIC Configuration* located in the **Project Graph** tab (Figure 9) and change the interrupt configurations as shown in Figure 10.

FIGURE 9: PROJECT GRAPH>PLUGINS>NVIC CONFIGURATION

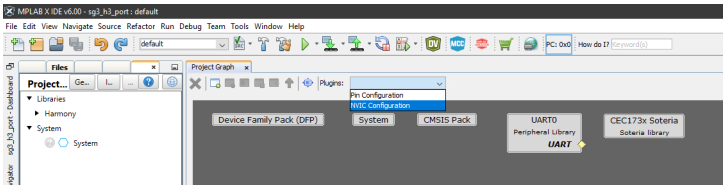
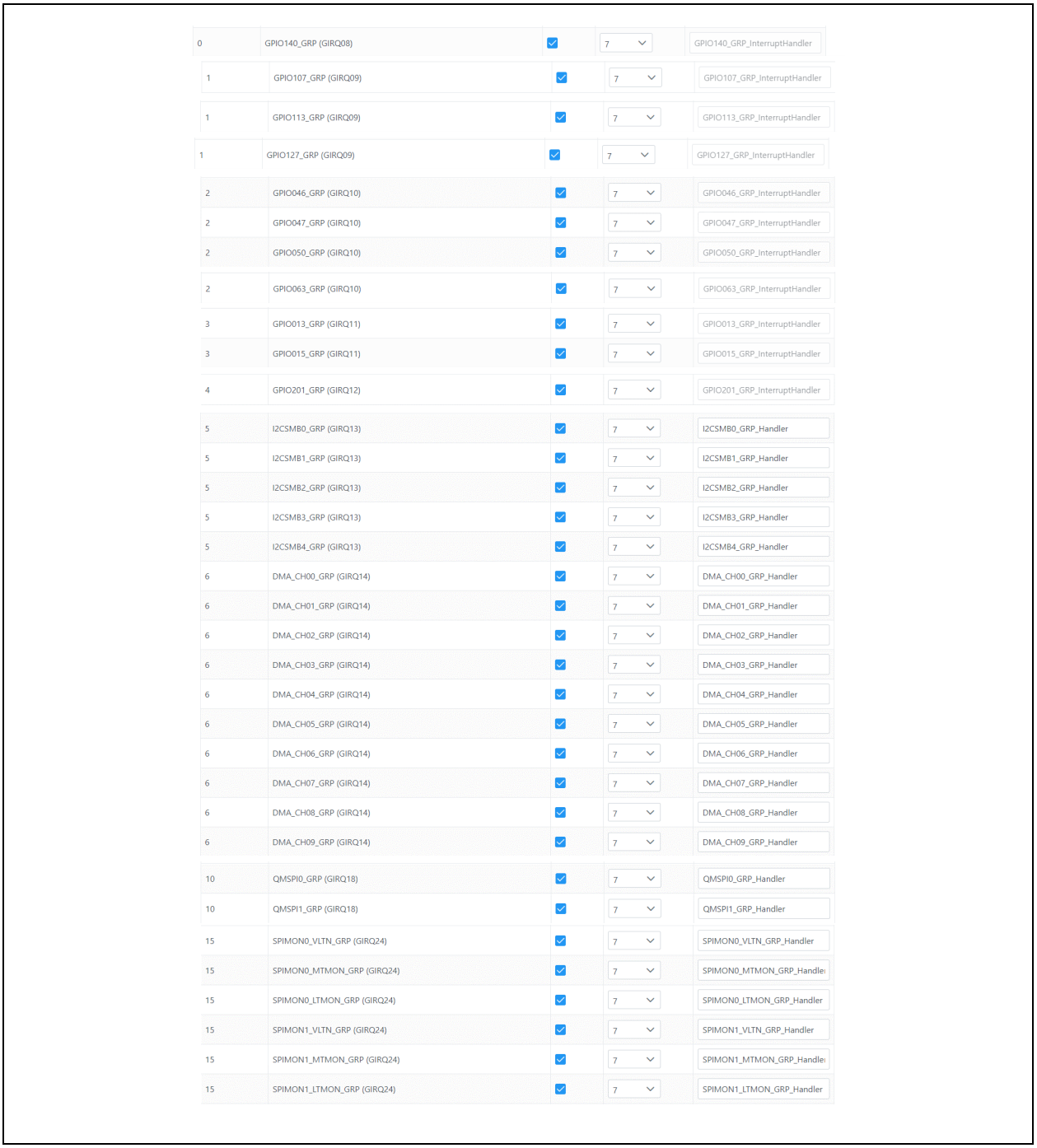


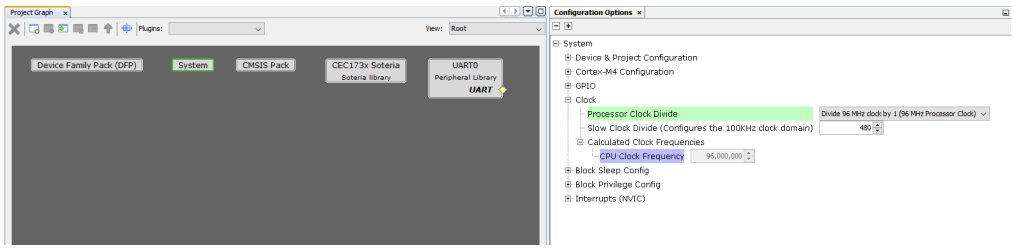
FIGURE 10: INTERRUPT CONFIGURATION



0	GPIO140_GRP (GIRQ08)	☑	7	GPIO140_GRP_InterruptHandler
1	GPIO107_GRP (GIRQ09)	☑	7	GPIO107_GRP_InterruptHandler
1	GPIO113_GRP (GIRQ09)	☑	7	GPIO113_GRP_InterruptHandler
1	GPIO127_GRP (GIRQ09)	☑	7	GPIO127_GRP_InterruptHandler
2	GPIO046_GRP (GIRQ10)	☑	7	GPIO046_GRP_InterruptHandler
2	GPIO047_GRP (GIRQ10)	☑	7	GPIO047_GRP_InterruptHandler
2	GPIO050_GRP (GIRQ10)	☑	7	GPIO050_GRP_InterruptHandler
2	GPIO063_GRP (GIRQ10)	☑	7	GPIO063_GRP_InterruptHandler
3	GPIO013_GRP (GIRQ11)	☑	7	GPIO013_GRP_InterruptHandler
3	GPIO015_GRP (GIRQ11)	☑	7	GPIO015_GRP_InterruptHandler
4	GPIO201_GRP (GIRQ12)	☑	7	GPIO201_GRP_InterruptHandler
5	I2CSMB0_GRP (GIRQ13)	☑	7	I2CSMB0_GRP_Handler
5	I2CSMB1_GRP (GIRQ13)	☑	7	I2CSMB1_GRP_Handler
5	I2CSMB2_GRP (GIRQ13)	☑	7	I2CSMB2_GRP_Handler
5	I2CSMB3_GRP (GIRQ13)	☑	7	I2CSMB3_GRP_Handler
5	I2CSMB4_GRP (GIRQ13)	☑	7	I2CSMB4_GRP_Handler
6	DMA_CH00_GRP (GIRQ14)	☑	7	DMA_CH00_GRP_Handler
6	DMA_CH01_GRP (GIRQ14)	☑	7	DMA_CH01_GRP_Handler
6	DMA_CH02_GRP (GIRQ14)	☑	7	DMA_CH02_GRP_Handler
6	DMA_CH03_GRP (GIRQ14)	☑	7	DMA_CH03_GRP_Handler
6	DMA_CH04_GRP (GIRQ14)	☑	7	DMA_CH04_GRP_Handler
6	DMA_CH05_GRP (GIRQ14)	☑	7	DMA_CH05_GRP_Handler
6	DMA_CH06_GRP (GIRQ14)	☑	7	DMA_CH06_GRP_Handler
6	DMA_CH07_GRP (GIRQ14)	☑	7	DMA_CH07_GRP_Handler
6	DMA_CH08_GRP (GIRQ14)	☑	7	DMA_CH08_GRP_Handler
6	DMA_CH09_GRP (GIRQ14)	☑	7	DMA_CH09_GRP_Handler
10	QMSPI0_GRP (GIRQ18)	☑	7	QMSPI0_GRP_Handler
10	QMSPI1_GRP (GIRQ18)	☑	7	QMSPI1_GRP_Handler
15	SPIMON0_VLTN_GRP (GIRQ24)	☑	7	SPIMON0_VLTN_GRP_Handler
15	SPIMON0_MTMON_GRP (GIRQ24)	☑	7	SPIMON0_MTMON_GRP_Handler
15	SPIMON0_LTMON_GRP (GIRQ24)	☑	7	SPIMON0_LTMON_GRP_Handler
15	SPIMON1_VLTN_GRP (GIRQ24)	☑	7	SPIMON1_VLTN_GRP_Handler
15	SPIMON1_MTMON_GRP (GIRQ24)	☑	7	SPIMON1_MTMON_GRP_Handler
15	SPIMON1_LTMON_GRP (GIRQ24)	☑	7	SPIMON1_LTMON_GRP_Handler

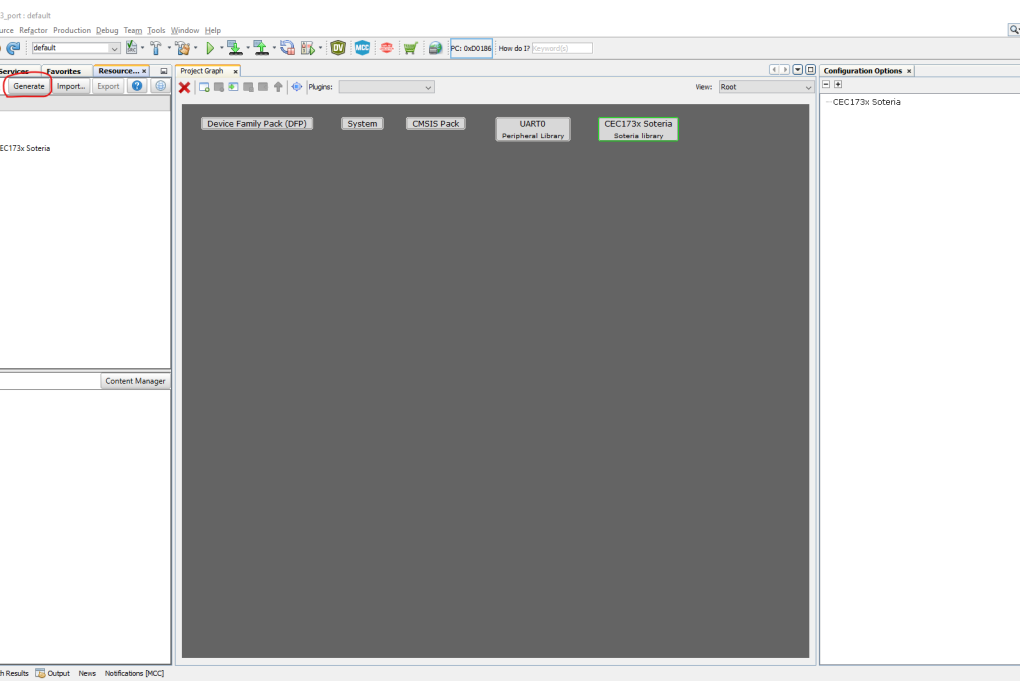
9. Change the core clock settings as shown in Figure 11.

FIGURE 11: CLOCK CONFIGURATION



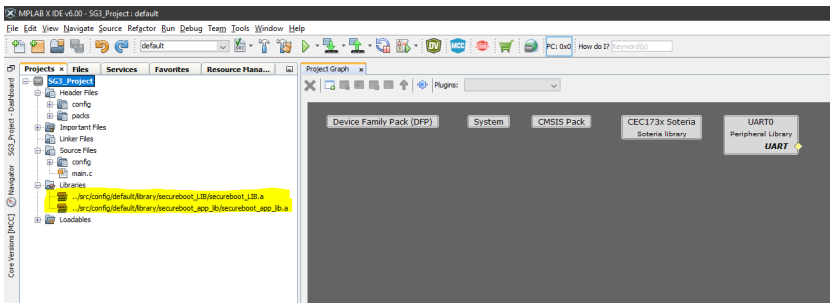
10. Click the **Generate** button located in the **Project Resources** tab and wait for the code generation to complete (Figure 12).

FIGURE 12: GENERATING CODE



Once the code generation is complete, the Soteria-G3 can be located under the “Libraries” folder of the current project (Figure 13).

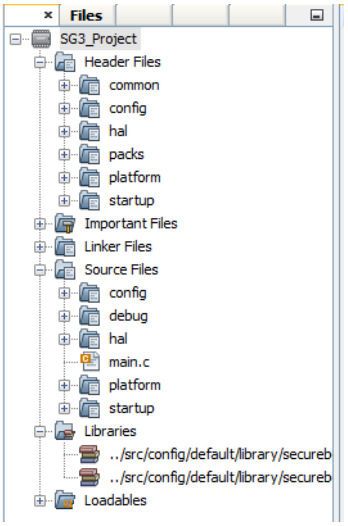
FIGURE 13: SOTERIA-G3 PROJECT UNDER LIBRARIES FOLDER



11. Use the “hal” folder provided in the `cec173x_soteria_lib/apps/sg3_h3_port` application project (refer to Section 3.1, "Opening Soteria-G3 Sample Library Project") in this new project. Make sure to add this folder into the XC32 compiler, including the path in your project settings.
12. Use the “startup” folder provided in `cec173x_soteria_lib/apps/sg3_h3_port` application project (refer to Section 3.1, "Opening Soteria-G3 Sample Library Project") in this new project (include only `startup_CEC173x.S` in the project).
13. Use the “common” folder provided in `cec173x_soteria_lib/apps/sg3_h3_port` application project (refer to Section 3.1, "Opening Soteria-G3 Sample Library Project") in this new project. Make sure to add this folder into the XC32 compiler, including the path in your project settings.
14. Use the linker script `secureboot_app.ld` provided in step 13 (refer to Section 3.1, "Opening Soteria-G3 Sample Library Project") in this new project (can be found under `common/include/`).
15. Use the “platform” folder provided in `cec173x_soteria_lib/apps/sg3_h3_port` application project (refer to Section 3.1, "Opening Soteria-G3 Sample Library Project") in this new project. Make sure to add this folder into the XC32 compiler, including the path in your project settings.

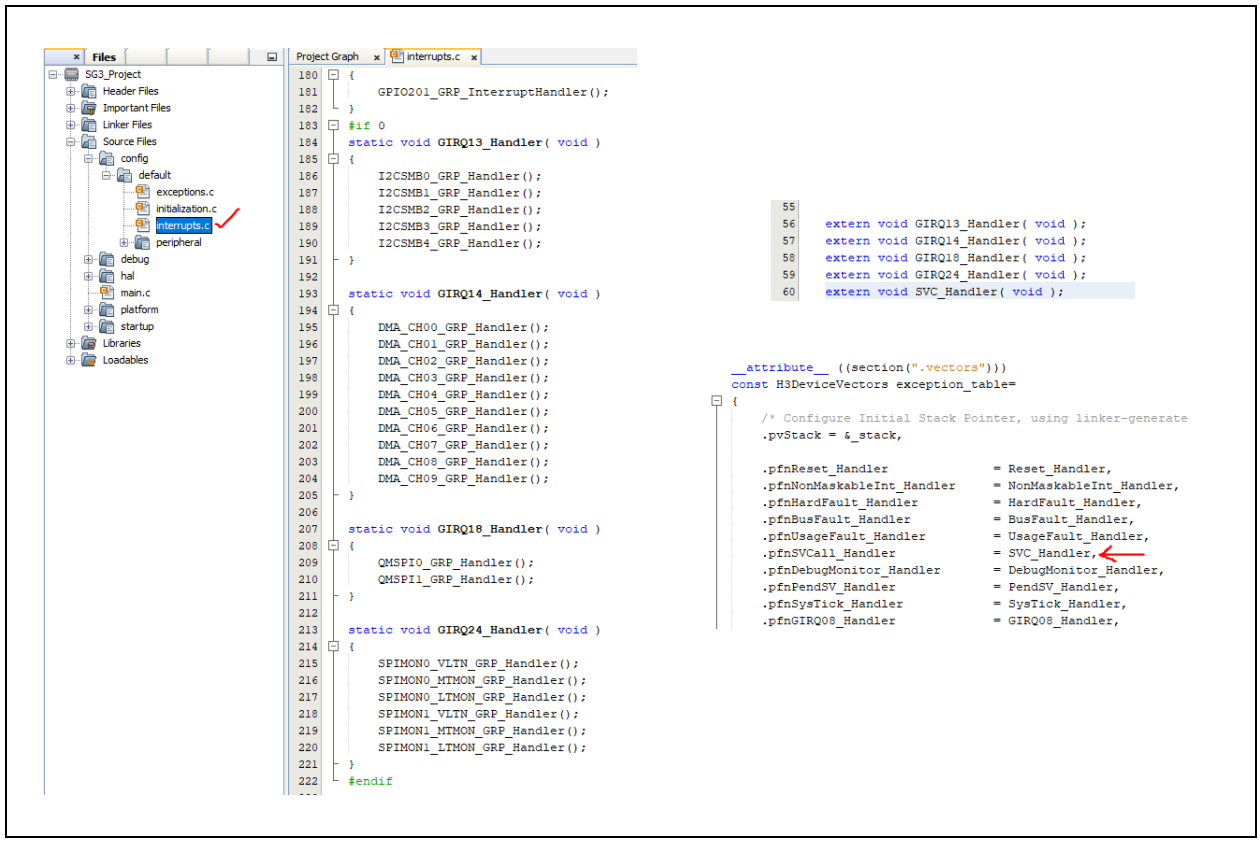
The project structure should then display as in Figure 14.

FIGURE 14: SOTERIA-G3 PROJECT STRUCTURE



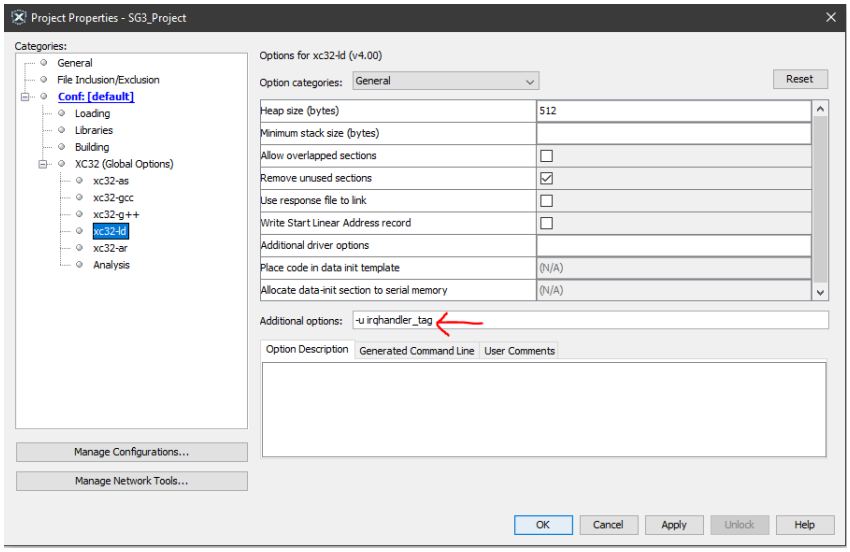
16. Navigate to *Source Files>config>default>interrupts.c* in the project and carry out the following changes as shown in Figure 15.
    a) Disable the “GIRQ13_Handler”, “GIRQ14_Handler”, “GIRQ18_Handler”, and “GIRQ24_Handler” functions and add their external declarations.
    b) Change the name of Supervisor Call handler function from “SVCall_Handler” to “SVC_Handler”.

FIGURE 15: SOURCE FILES>CONFIG>DEFAULT>INTERRUPTS.C



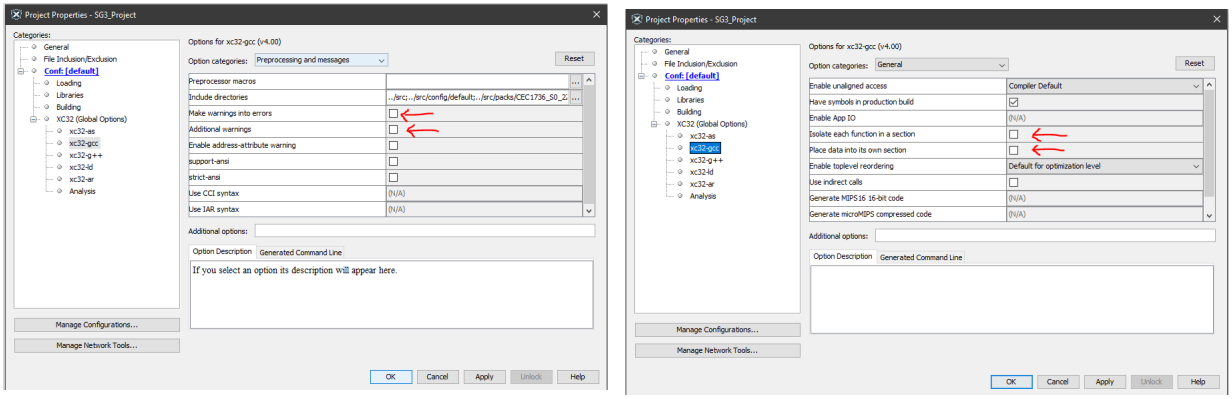
17. Enter *-u irqhandler_tag* in the “Additional Options” text box section of the XC32 linker options as shown Figure 16.

FIGURE 16: XC32 LINKER OPTIONS



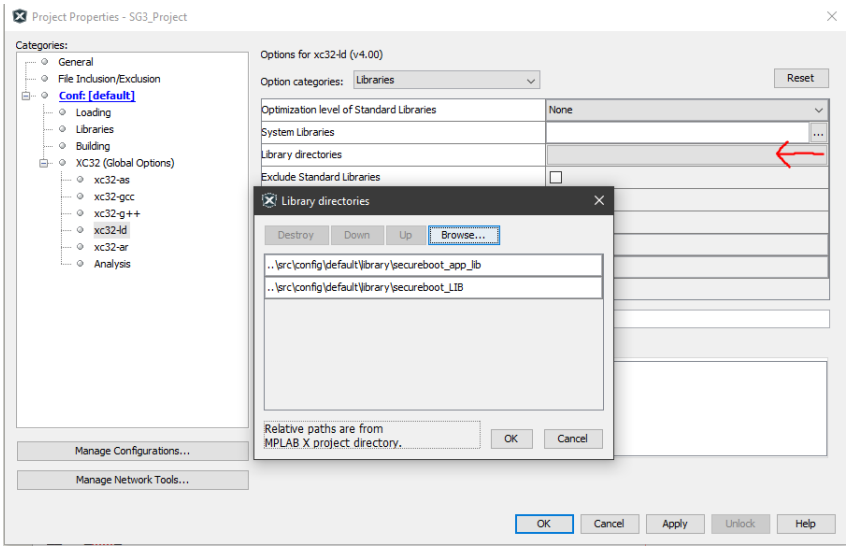
18. Disable the following options in the XC32 compiler settings as shown in Figure 17.
    - “Make warnings into errors”
    - “Additional warnings”
    - “Isolate each function in a section”
    - “Place data into its own section”

FIGURE 17: OPTIONS TO BE DISABLED IN XC32 COMPILER



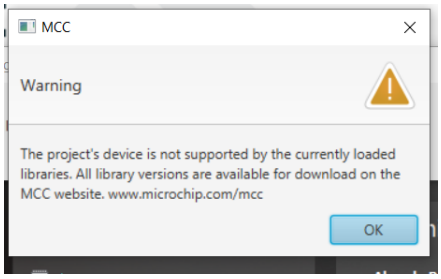
19. Add the path to the Soteria-G3 libraries into the XC32 linker options as shown in Figure 18.

FIGURE 18: ADDING PATH TO SOTERIA-G3 LIBRARIES INTO XC32 LINKER OPTIONS



20. If you get the following error during the project creation process, navigate to *Tools>Options>Plugins Tab>MPLAB Code Configurator x.x* as shown in step 2 in Section 3.1, "Opening Soteria-G3 Sample Library Project" and reset the path to the Harmony Framework with the same value again.

FIGURE 19: PROJECT CREATION ERROR



21. Include the `common.h` file in the `main.c` file of this project.
22. To run the Soteria-G3 application, the application's main function should call the functions described in Section 5.2, "Soteria-G3-Specific APIs".

Note: Refer to Section 5.0, "Soteria-G3 Library APIs" and Section 7.0, "Application Tasks for Debugging" to understand the usage of the available API functions and OEM tasks.

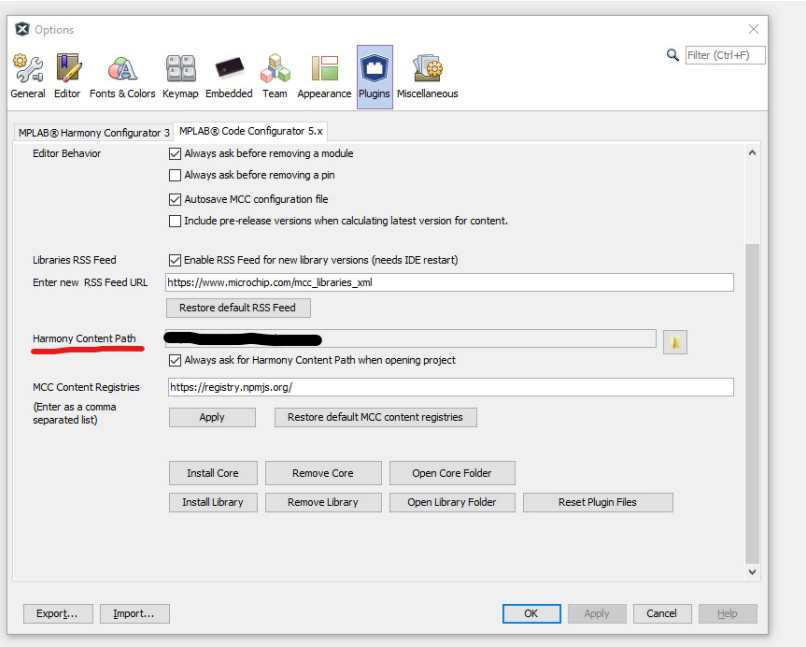
3.0 SOTERIA-G3 SAMPLE LIBRARY PROJECT

To ease the process of creating a Soteria-G3 project from scratch, a sample project has already been created, which can be found under `HarmonyFrameworkPath/cec173x_soteria_lib/apps/sg3_h3_port/`.

3.1 Opening Soteria-G3 Sample Library Project

1. From the MCC content manger, select the `cec173x_soteria_lib` component and download it.
2. Locate the “MCC Content Path” by navigating to *Tools>Options>Plugins Tab>MPLAB Code Configurator x.x* as shown in Figure 20.

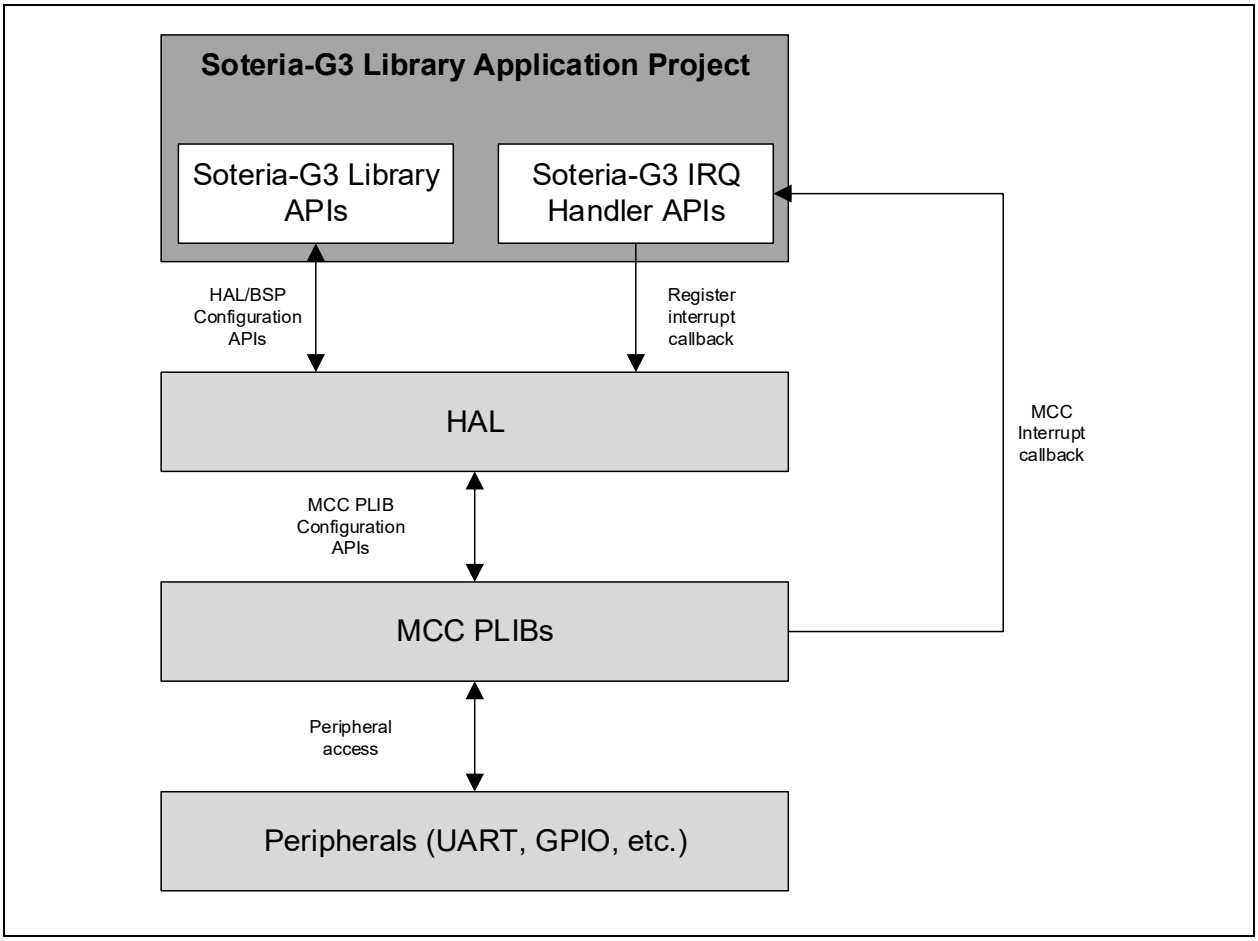
FIGURE 20: TOOLS>OPTIONS>PLUGINS TAB>MPLAB CODE CONFIGURATOR X.X



3. Navigate to this location to find the `cec173x_soteria_lib/apps/sg3_h3_port/firmware/` folder that contains the Soteria-G3 application project for this device.
4. Open the `sg3_h3_port` sample application project in MPLABX.
5. Use the application task functions of this project as mentioned in Section 7.0, "Application Tasks for Debugging" to get started with developing an application.

3.2 High-Level Design

FIGURE 21: SOTERIA-G3 HIGH-LEVEL DESIGN



4.0 SOTERIA-G3 LIBRARY PROJECT STRUCTURE

TABLE 1: SOTERIA-G3 LIBRARY PROJECT STRUCTURE

Project Level	Description
common/debug/	APIs for UART debugging
common/include/	APIs for working with GPIO and ECIA blocks Common file inclusions for use by application Linker script
config/	MCC-generated PLIB files
hal/	Hardware Abstraction Layer APIs (not to be used unless an API is not present in ahb_api_mpu.h)
kernel/	Soteria-G3 APIs for application use
oem/	Functions and definitions for adding user code
packs/	MCC- generated device-specific files (not for application use)
platform/	Application-specific configurations Interrupt handling routines
startup/	Device startup file

5.0 SOTERIA-G3 LIBRARY APIS

5.1 UART Debugging

5.1.1 FORMATTED PRINTING TO UART

Function prototype:

```
void tracex(const char *fmt, ...);
```

Description:

The function usage is like the `printf` function of stdio

Inputs:

Same as `printf` function of stdio

Outputs:

None

5.1.2 ISR SAFE FORMATTED PRINTING TO UART

Function prototype:

```
void tracex_from_ISR(const char *fmt, ...);
```

Description:

This function is an ISR safe equivalent of `tracex`

Inputs:

Same as `printf` function of stdio

Outputs:

None

5.1.3 HEX DUMP TO UART

Function prototype:

```
void print_buf(uint8_t *buf, uint32_t len);
```

Description:

Prints hexadecimal values inside a buffer of user defined length

Inputs:

Input Parameter	Description
buf	Pointer to a user defined allocated buffer which contains
len	Length of the user defined allocated buffer

Outputs:

None

5.2 Soteria-G3-Specific APIs

5.2.1 SOTERIA-G3 FIRMWARE INITIALIZATION

Function prototype:

```
int sg3_init(void)
```

Description:

Initializes the Soteria-G3 firmware application

Inputs:

Input Parameter	Description
0	Soteria-G3 initialization succeeded
–1	Soteria-G3 initialization failed

Outputs

None

5.2.2 START SOTERIA-G3 FIRMWARE OPERATION

Function prototype:

```
void sg3_start(void)
```

Description:

Runs the Soteria-G3 firmware application

Inputs:

None

Outputs:

None

5.3 GPIO and ECIA Peripheral Access

To configure the GPIO and ECIA peripherals from OEM functions, please refer to the `ahb_api_mpu.h` file present in the `cec173x_soteria_lib /apps/sg3_h3_port` sample Soteria-G3 project. Accessing these peripherals directly using MCC-generated APIs is not allowed because of software design constraints.

5.4 Interrupts

The following interrupts are already defined in the Soteria-G3 application library, hence redefining these interrupt handlers will cause a build error:

- GIRQ13_Handler
- GIRQ14_Handler
- GIRQ18_Handler
- GIRQ24_Handler
- SVC_Handler

For use in your custom Soteria-G3 project, it is enough to declare the prototypes for these handlers as follows:

- extern void GIRQ13_Handler (void);
- extern void GIRQ14_Handler (void);
- extern void GIRQ18_Handler (void);
- extern void GIRQ24_Handler (void);
- extern void SVC_Handler (void);

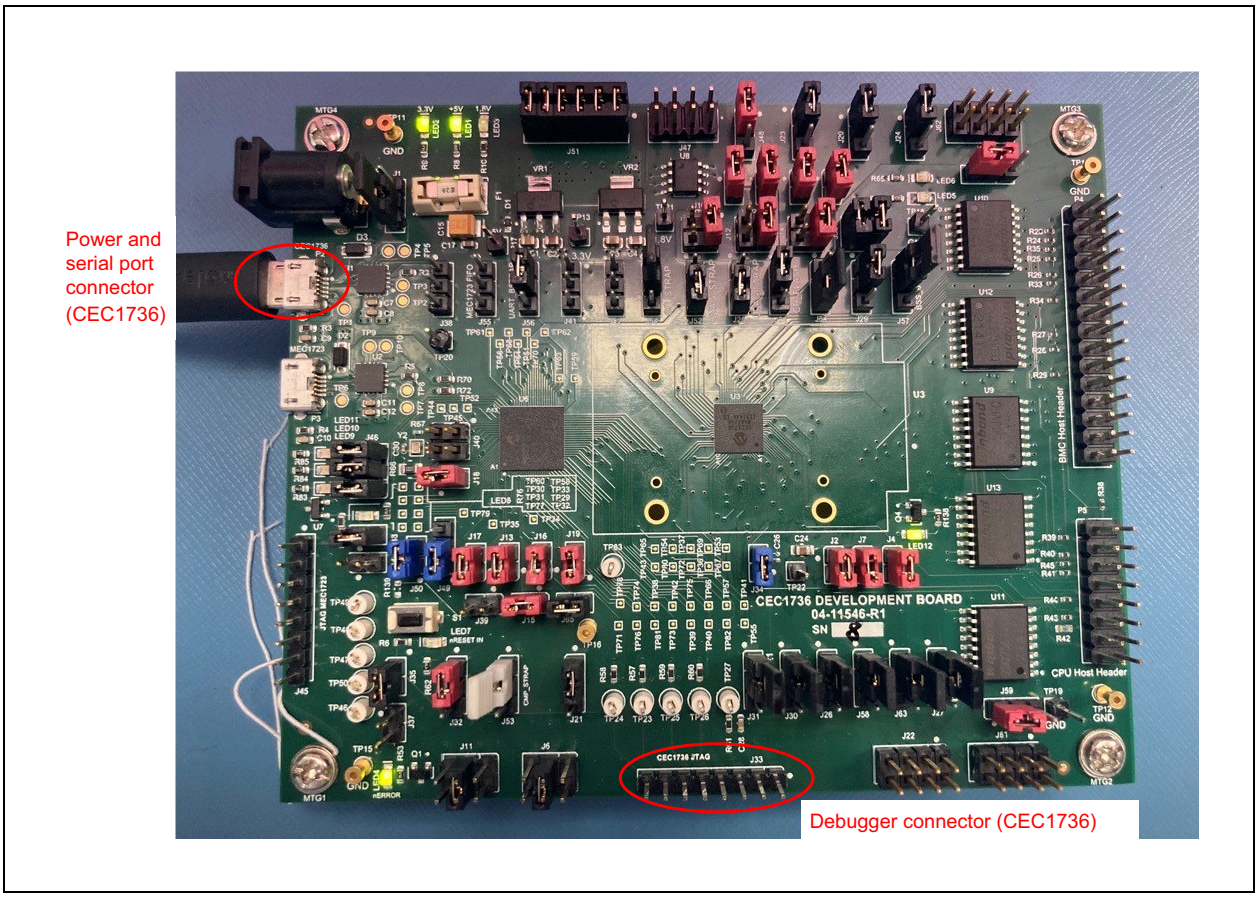
Note: Make sure that the names of the ISRs above match with those in the vector table generated by MCC (located in `config/default/interrupts.c`).

6.0 SOTERIA-G3 USER INTERACTION AND FEEDBACK

6.1 Debugging

1. Connect a micro-USB cable to the P2 connector on the development board.
2. Connect the debugger to the J33 connector on the development board.

FIGURE 22: CEC173X DEVELOPMENT BOARD



3. Open the `sg3_h3_port` sample Soteria-G3 project using MPLABX IDE. (Refer to Section 3.1, "Opening Soteria-G3 Sample Library Project".)
4. Clean and build the project by selecting the “Clean and Build” option from the project context menu.
5. Start a debug session of this project by selecting the “Debug” option from the project context menu.
6. Click **Run** from the “Debug” context menu.
7. Open PuTTY or any other serial port application with the following settings:
   - Baud rate: 115200
   - Stop bits: 1
   - Flow control: Off
   - Parity: None

The UART output from Soteria-G3 can be observed on the serial port application.

6.2 On Board LEDs

TABLE 2: LED12 BEHAVIOR

State	Observation
Authenticating AP images	Blink rate = 2 Hz, Pattern = None
Authentication completed and no error detected	Blink rate = 0.5 Hz, Pattern = None
Authentication completed and non-fatal error detected	Blink rate = 1 Hz, Pattern = 2
Authentication completed and fatal error detected	Blink rate = 1 Hz, Pattern = 1
Executing recovery sequence	Blink rate = 4 Hz, Pattern = None
Authentication completed post recovery and no error detected	Blink rate = 1 Hz, Pattern = None

Note 1: Blink Pattern 1: Blink–Blink–Off–Off <repeat>
2: Blink Pattern 2: Blink–Off–Off <repeat>

TABLE 3: LED5 AND LED6 BEHAVIOR

State	AP0 Critical Image	AP1 Critical Image	LED5	LED6
Authenticating AP images	No failure	No failure	Off	Off
	Image failure	No failure	Blink rate = 1 Hz Pattern = None	Off
	No failure	Image failure	Off	Blink rate = 1 Hz Pattern = None
	Image failure	Image failure	Blink rate = 1 Hz Pattern = None	Blink rate = 1 Hz Pattern = None
Executing recovery sequence	Recover image	No recovery	Blink rate = 4 Hz Pattern = None	Off
	No recovery	Recover image	Off	Blink rate = 4 Hz Pattern = None
	Recover image	Recover image	Blink rate = 4 Hz Pattern = None	Blink rate = 4 Hz Pattern = None
Authentication completed and error detected	Non-fatal error	No failure	Blink rate = 1 Hz Pattern = None	Off
	No Failure	Non-fatal error	Off	Blink rate = 1 Hz Pattern = None
	Non-fatal error	Non-fatal error	Blink rate = 1 Hz Pattern = None	Blink rate = 1 Hz Pattern = None
	No failure	Fatal error	Off	Blink rate = 1 Hz Pattern = 2
	Non-fatal error	Fatal error	Blink rate = 1 Hz Pattern = None	Blink rate = 1 Hz Pattern = 2
	Fatal error	X	Blink rate = 1 Hz Pattern = 1	Blink rate = 1 Hz Pattern = 1
Authentication completed and no error detected	Pass	Pass	Off	Off
Authentication completed post recovery	Image recovered	No image recovered	Blink rate = 1 Hz Pattern = None	Off
	No image recovered	Image recovered	Off	Blink rate = 1 Hz Pattern = None
	Image recovered	Image recovered	Blink rate = 1 Hz Pattern = None	Blink rate = 1 Hz Pattern = None

Note 1: Blink Pattern 1: Blink–Blink–Off–Off <repeat>
2: Blink Pattern 2: Blink–Off–Off <repeat>

7.0 APPLICATION TASKS FOR DEBUGGING

Soteria-G3 provides OEM task functions for user to play around with various features of the application project.

There are three functions provided to the user to get started with Soteria-G3:

- oem_task1_function ()
- oem_task2_function ()
- oem_task3_function ()

The user can add his own code inside these functions to evaluate the capabilities and features of Soteria-G3 and CEC173x secure-boot controller.

Note: Refer to the sample Soteria-G3 application project present in `cec173x_soteria_lib/apps/sg3_h3_port` for reference. The OEM task functions can be located under `src/oem/oem_task1`, `src/oem/oem_task2`, and `src/oem/oem_task3` directories.

APPENDIX A: APPLICATION NOTE REVISION HISTORY

TABLE A-1: REVISION HISTORY

Revision Level & Date	Section/Figure/Entry	Correction
DS00004691A (08-09-22)	Initial release	

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable" Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

ISBN: 978-1-6683-1048-9

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820