

QC-BEKITPOC2Z

QuickConnect Beginners Kit V2.0

Introduction

This application note provides step-by-step guidance using the QuickConnect Beginners Kit V2.0 for three projects: Blinky, Smart Temperature Data Logger, and BLE-based Asset Tracker. Each project includes detailed instructions for hardware setup, software configuration, and result analysis. The document also suggests possible customizations to help users modify and expand the project outcomes.

All applications are developed using the QuickConnect Studio alongside the Beginners Kit V2.0 hardware enabling rapid and flexible prototype development. To ensure proper setup, follow the Quick Start Procedure at the beginning of this document.

These labs offer hands-on experience that guides users through both hardware assembly and software development for embedded system prototyping. The QuickConnect platform streamlines the design process allowing users to move from concept to prototype in just minutes. Code generation, compilation, customization, and debugging are all seamlessly managed through a single, unified interface.

Note: To ensure the projects are set up correctly, complete the steps in the order listed in this document.

Contents

1.	Kit Information2				
	1.1	Kit Content	is	2	
2.	Hardy	vare Descri	ption	3	
	2.1	BGK-RA6E	2	3	
		2.1.1 E	Block Diagram	4	
		2.1.2 E	Board Specifications	4	
	2.2	DA16600 F	PMOD	7	
	2.3	DA14531 F	PMOD	8	
	2.4	Humidity a	nd Temperature Sensor PMOD	8	
3.	Quick	Connect St	udio	9	
4.	Prerequisites9				
5.	Gettir	ng Started		11	
	5.1	Launch Qu	ickConnect Studio Workspace	11	
	5.2	Create a N	ew Project	12	
	Project 1 – Blinky				
6.	Proje	ct 1 – Blinky	y	13	
6.	Proje 6.1	ct 1 – Blink y Create the	y Project	 13 13	
6.	Proje 6.1 6.2	ct 1 – Blinky Create the Programmi	y Project ing Hardware and Viewing Results	 13 13 16	
6. 7.	Proje 6.1 6.2 Proje	ct 1 – Blinky Create the Programmi ct 2 - Smart	y Project ng Hardware and Viewing Results Temperature Data Logger	 13 13 16 17	
6. 7.	Proje 6.1 6.2 Proje 7.1	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr	y Project ing Hardware and Viewing Results Temperature Data Logger reate the Project	 13 13 16 17 17	
6. 7.	Project 6.1 6.2 Project 7.1 7.2	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi	y Project ng Hardware and Viewing Results Temperature Data Logger reate the Project ng Hardware and Viewing Results	13 13 16 17 17 23	
6. 7. 8.	Project 6.1 6.2 Project 7.1 7.2 Project	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b	y Project Ing Hardware and Viewing Results reate the Project Ing Hardware and Viewing Results ased Asset Tracker	13 13 16 17 17 23 27	
6. 7. 8.	Project 6.1 6.2 Project 7.1 7.2 Project 8.1	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr	y Project Ing Hardware and Viewing Results Temperature Data Logger reate the Project Ing Hardware and Viewing Results Ing Hardware	13 13 16 17 17 23 27	
6. 7. 8.	Project 6.1 6.2 Project 7.1 7.2 Project 8.1 8.2	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr Programmi	y Project Project Temperature Data Logger reate the Project Ing Hardware and Viewing Results Pased Asset Tracker Project Ing Hardware and Viewing Results Project	13 13 16 17 23 27 27 30	
6. 7. 8. 9.	Project 6.1 6.2 Project 7.1 7.2 Project 8.1 8.2 Debug	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr Programmi gging on Qu	y Project Ing Hardware and Viewing Results Temperature Data Logger Ing Hardware and Viewing Results Pased Asset Tracker Ing Hardware and Viewing Results Ing Hardware and Viewing Results Ing Hardware and Viewing Results Ing Hardware Studio Ing Kardware Studio Ing Hardware	13 13 16 17 23 27 27 30 34	
6. 7. 8. 9.	Project 6.1 6.2 Project 7.1 7.2 Project 8.1 8.2 Debut 9.1	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr Programmi gging on Qu Remote De	y Project Project Temperature Data Logger reate the Project Ing Hardware and Viewing Results Ing Hardware and Viewing Results Pased Asset Tracker Preate the Project Ing Hardware and Viewing Results Ing Hardware And Viewi	13 13 16 17 23 27 27 30 34 34	
6. 7. 8. 9.	Projec 6.1 6.2 Projec 7.1 7.2 Projec 8.1 8.2 Debut 9.1 Appe	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr Programmi gging on Qu Remote De ndix	y Project Ing Hardware and Viewing Results reate the Project Ing Hardware and Viewing Results Ing Ha	13 13 16 17 23 27 27 30 34 34 37	
 6. 7. 8. 9. 10. 	Project 6.1 6.2 Project 7.1 7.2 Project 8.1 8.2 Debut 9.1 Apper 10.1	ct 1 – Blinky Create the Programmi ct 2 - Smart Steps to Cr Programmi ct 3 - BLE-b Steps to Cr Programmi gging on Qu Remote De ndix	y Project Ing Hardware and Viewing Results Temperature Data Logger reate the Project Ing Hardware and Viewing Results Pased Asset Tracker Ing Hardware and Viewing Results Ing Hardware and Viewing Results Ing Hardware using Results Debugging Ode to the Hardware using SEGGER J-Flash Lite	13 13 16 17 23 27 27 30 34 34 37 37	



	10.3	Customiz	zation	41
		10.3.1	Foundation for Customization	41
		10.3.2	Customize Blinky application	44
		10.3.3	Customize Sensor Data to AWS Cloud application	82
		10.3.4	Customize Sensor Data to BLE application	93
11.	Refer	ences		110
12.	Revis	.3 Customization 41 10.3.1 Foundation for Customization 41 10.3.2 Customize Blinky application 44 10.3.3 Customize Sensor Data to AWS Cloud application 82 10.3.4 Customize Sensor Data to BLE application 93 eferences 110 evision History 110		

1. Kit Information

1.1 Kit Contents

Orderable Part Number – QC-BEKITPOC2Z

Hardware Components - BGK-RA6E2 (R7FA6E2BB3CFM) MCU board

- PMOD Board with Ultra-Low Power Wi-Fi + Bluetooth[®] Low Energy Combo Module, DA16600MOD
- PMOD Board with Low Power Bluetooth, DA14531MOD
- PMOD Board with Relative Humidity and Temperature Sensor (HS4001)

Note: While the sensor used on the board (HS4001) is marked as NRND (Not Recommended for New Designs), users can still run the examples in this document and fully experience QuickConnect Studio without any issues.

2. Hardware Description

2.1 BGK-RA6E2

The BGK-RA6E2 Board provides an entry point for evaluation, prototyping, and development with the RA6E2 MCU. Also, because this board incorporates an emulator circuit, it can be used for designing user applications without the requirement of making further investments in tools. This product includes through-holes for pin headers that allow access to all MCU signal pins, which allows easy prototyping with a breadboard.

Item	Specification
Evaluation MCU	Part No: R7FA6E2BB3CFM; package: 64-pin LQFP
	On-chip memory: 256KB code flash, 40KB SRAM, 4KB data flash memory
Board size	Size: 53mm × 85mm; thickness: 1.6mm
Power-supply voltage	Board supply: 5V. VCC: 3.3V. MCU operation voltage range 2.7V to 3.6V
Power-supply circuit	USB connector: VBUS (5V input); VBUS is converted to 3.3V by LDO
	2-pin external power-supply header*1
Push switch	Reset switch x 1, user switch x 1
LED	Power indicator: green x 1, user: green x 2, On-board debugger ACT: yellow x 1
USB connector	Connector: micro-USB type B
PMOD [™] connector	Connector: angle type, 12-pin x 2
Arduino [™] connector	Connector: 6-pin x 1, 8-pin x 2, 10-pin x 1
	The interface is compatible with Arduino [™] Uno R3
MCU header ¹	Header: 32 pins x 2
Emulator	J-Link on-board programmer / debugger

Table	1.	Board	Specifications
Iable		Duaru	opecifications

^{1.} This part is not mounted.

2.1.1 Block Diagram



2.1.2 Board Specifications



Figure 1. BGK-RA6E2 MCU Board Layout

2.1.2.1 Arduino Interface



Figure 2. BGK-RA6E2 MCU Board Arduino Interface

2.1.2.2 PMOD Interface

The QC Beginner Kit RA6E2 MCU Board has two PMOD connectors. PMOD 1 supports the Type 2A interfaces and can also be configured for Type 3A. PMOD 2 supports Type 6A interfaces and can also be configured for Type 2A, Type 3A, and Type 7A.

The default setting for PMOD 1 is SPI/UART interfaces.



(Configure the indicated jumpers as shown in the table)

Figure 3. PMOD 1 Interface

The default setting for PMOD 2 is I2C interfaces.



NOTE1: P213/P212 option can not be selected at the same time as the 24MHz crystal



2.1.2.3 Copper Jumpers

Two types of copper jumpers are provided on the RA6E2 MCU board, designated trace-cut and solder-bridge with E.

A trace-cut jumper is provided with a narrow copper trace connecting its pads. The silkscreen overlay printing around a trace-cut jumper is a solid box. To isolate the pads, cut the trace between pads adjacent to each pad, and remove the connecting copper foil either mechanically or with the assistance of heat. When the etched copper trace is removed, the trace-cut jumper is turned into a solder-bridge jumper for any later changes.

A solder-bridge jumper is provided with two isolated pads can be joined together by either of the three methods:

- Solder can be applied to both pads to develop a bulge on each and the bulges joined by touching a soldering iron across the two pads.
- A small wire can be placed across the two pads and soldered in place.
- A SMD resistor, size 0805, 0603, or 0402, can be placed across the two pads and soldered in place. A 0Ω resistor shorts the pads together.

For any copper jumper, the connection is considered closed if there is an electrical connection between the pads (default for trace-cut jumpers.) The connection is considered open if there is no electrical connection between the pads (default for the solder-bridge jumpers).

Trace-cut Jumper

Solder Bridge



Figure 5. Copper Jumpers

2.2 DA16600 PMOD

The <u>US159-DA16600EVZ</u> is a PMOD board for the DA16600MOD. It is a Wi-Fi BLE module (802.11 b/g/n and Bluetooth V5.1). The DA16600 addresses the requirement of battery-use devices that require minimal power consumption and reliable operation. It can be used with a MCU board that supports Type 3A PMOD. In the following projects, this module is used for Wi-Fi communication.



Figure 6. US159-DA16600EVZ PMOD Board



Figure 7. Evaluation Kit Details

2.3 DA14531 PMOD

The <u>US159-DA14531EVZ</u> is a PMOD board with DA14531MOD. It is a low power Bluetooth 5.1 SOC module. In this document, this module is used for BLE based projects.



Figure 8. DA14531 Evaluation Board (Top)



Figure 9. DA14531 Evaluation Board (Bottom)

2.4 Humidity and Temperature Sensor PMOD

The humidity and temperature PMOD board with a digital I²C humidity and temperature sensor is used in the following labs.



Connector to MCU board

Relative Humidity Sensor

Figure 10. Humidity and Temperature Sensor Board Layout

Connector for additional sensor PMOD boards



Figure 11. Sensor PMOD Application Schematic

3. QuickConnect Studio

The QuickConnect Studio (QCStudio) is an online, cloud-based embedded system design platform that enables users to graphically drag-and-drop eval boards to build custom prototypes. Users can generate, compile, customize, and debug the base system code on a single browser window without installing multiple applications. This reduces the complexity to build and test embedded system prototypes and increases the time to market.

3.1 Key Features

The following list highlights key features supported on QuickConnect platform.

- QuickConnect supports a broad portfolio of Renesas and Partner boards and devices
- Real-time code customization
- QuickConnect debugging by connecting to remote board farms deployed globally
- Multi-region deployment to increase connection speed from anywhere in the world
- Support for multiple concurrent users globally
- Real-time monitoring for cyber security threats

For more details, refer to the <u>QuickConnect Studio</u> landing page.

4. Prerequisites

Before you begin this lab, ensure you have the following items:

- QuickConnect Beginners Kit V2.0 The kit includes all the hardware components needed for this lab.
- Windows PC A computer running Windows operating system. Renesas recommends using a system running Windows 10 or later versions, equipped with at least one available USB port.

- Web Browser Ensure a web browser is installed on a Windows PC. For optimal performance, it is advised to use Google Chrome.
- MyRenesas Registration An active MyRenesas account must be obtained to access QuickConnect. If you
 do not have one, register on the <u>MyRenesas</u> page.
- Internet Connection: A stable internet connection is required to access online resources and download necessary files.
- Segger J-link flasher <u>SEGGER The Embedded Experts Downloads J-Link / J-Trace</u>. To flash the boards, Renesas recommends downloading and installing the 64-bit installer, ensuring it is the latest version compatible with the user's device.
- QuickConnect Sandbox application For applications based on Bluetooth connectivity, download and install the QuickConnect Mobile Sandbox application for <u>Android</u> or <u>iOS</u>. It can support Android versions up to 14 and iOS 10.0 or later versions.
- Wi-Fi security requirements The DA16xxx series Wi-Fi modules support the following WiFi security types: 0 (OPEN), 1 (WEP), 2 (WPA), 3 (WPA2), 4 (WPA+WPA2), 5 (WPA3 OWE), 6 (WPA3 SAE), 7 (WPA2 RSN and WPA3 SAE). The "Sensor Data to AWS IoT" application used in this lab uses WPA2 security type by default. Verify that the security setting on the Wi-Fi access point(AP)/router matches this requirement. If not, update the settings accordingly either in the application or on the Wi-Fi AP/router.
- RTT Viewer Download and install the RTT Viewer application (version 8.24) which is used to view the debug logs from the Renesas365 Dev Kit. Download link: <u>About the RTT Viewer</u>
- MQTT Explorer Download and install the MQTT Explorer application, which is used to view the MQTT messages published by the Renesas365 Dev Kit to AWS IoT. Download link: <u>MQTT Explorer | An all-round</u> <u>MQTT client that provides a structured topic overview</u>

5. Getting Started

5.1 Launch QuickConnect Studio Workspace

- 1. Launch the QuickConnect Studio platform in a PC browser window.
 - a. To launch a QuickConnect Studio user workspace, visit the QuickConnect Studio.
 - b. Click on the Launch QuickConnect Studio button to launch a unique workspace in a browser window.



Redefining System Design: Co-Develop Hardware and Software with QuickConnect

QuickConnect simplifies the development of electronic systems by using standardized hardware and software building blocks.

QuickConnect Studio, a cloud-based design platform, facilitates visual construction of hardware and software, accelerating prototype validation and product development. Engineers can drag and drop hardware device blocks, and the platform auto-generates code. Additionally, it supports real-time code customization and remote debugging, enabling iterative testing before deploying physical boards.

QuickConnect Boards are the physical boards with standardized connectors that enable rapid prototyping and testing by eliminating compatibility issues.



Rapid Prototyping

Launch QuickConnect Studio

Hardware Compatibility

Access a fast, modular system of sensors, connectivity and MCU/MPU evaluation boards for rapid prototyping. Enjoy hardware compatibility with standardized PMOD, mikroBUS™ and Arduino interfaces that eliminate compatibility issues and simplify the design process.

Reduce Code Development

Save time and effort with pre-integrated sensor libraries and automatic code generation.



	Sign in to your account	
Username o	r email	
Password		
	Sign In	
	Sign In Or sign in with	

5.2 Create a New Project

1. Click on Create QuickConnect Studio Configuration Project.



2. Enter the project name in the field that appears and press the Enter key.

R	File Edit Diagram Selection \	/iew Go Run	Help			
← -	>│□ 📮 ∥ ※▼ 🗔					1
Û	EXPLORER Create QCStudio Configuration Project	1			GETTING STARTED	23
	✓ WORKSPACE		Rene	SAS	Getting Started With QUICK- CONNECT STUDIO Please see the sections below to get an overview of the available features and use the blue links to directly see them in action. Atternatively, please use the toolbar on top, right below the main menu bar. The following feature sections describe the needed steps and	
			🗐 Output X	File Downloader 🛛 🗸 🚍 🛆	also offer an action to execute those steps automatically, they are indicated by the play icon (③).	
£63					Pa Create QCStudio	

6. Project 1 – Blinky

6.1 Create the Project

1. Follow the steps in Section 5 to launch and create a new project.



2. Drag and drop the MCU board from the QCC tool palette. Hover the cursor over the MCU to view its features. In this example, BGK-RA6E2 is used.

R	🖪 File Edit Diagram Selection View Go Run Help					
← -	→ □ ⊑ ル % ▼ □				1:	
ධ	EXPLORER	& Blinky.qcc ×		QCC TOOL PALETTE	23	
م ھ	OPEN EDITORS WORKSPACE D Blinky/configuration Blinky.qcc	Blinky > configuration > [2] Blinky.qcc		Search tool palette items SHOWING ALL AVAILABLE DEVICES V RENESAS	3 (1)	
s¤⊳				aik-RA4E1 o Aik-RA6M3		
₿		89 A		BGK-RA6E2		
A		▲ Proble	BEGINNERS KIT FOR RAFE2 MCU GROUP 200MHz ARM® Cortex ®-M33 256K8 Code Flash, 40K8 SRAM Samples of use cases for AVMS connectivity and Renesas AI applications Mobile app available for Wi-Fi provisioning on Android and iOS Multiple sensor and wireless interface Arduino Uno and Pmod™ interfaces	図 CK-RA6M5 図 EK-RA2A2 図 EK-RA2E1 図 EK-RA2E2 図 EK-RA2L1 図 EK-RA4E2 図 EK-RA4M1 図 EK-RA4M2		
5				F EK-RA4M3		
⊗ 0	∆0				0 🗆	

3. Select the MCU and then choose Application Configuration.



4. In the Application Configuration window, select IoT > Other > Blink LEDs.

₿ В	linky.qcc ×		Ж в	linky.qcc ×
Blir	nky > configuration > 📴 Blinky.qcc		Blir	iky > configuration > '쪒 Blinky.qcc
>	APPLICATION CONFIGURATION		>	APPLICATION CONFIGURATION
	Search application			Search application
	Please select an application			
	imes Market Segment			< End Equipment
	Industrial	>		Other >
	юТ	>		
	Please select an application from the	list		Please select an application from the list above.
№ Blink Blink >	hky.qcc × y > configuration > 図 Blinky.qcc APPLICATION CONFIGURATION Search application IoT / Other < Application Blink LEDs ~		Beneroo	
	FreeRTOS Sensor Data to AdafruitIO (MQTT	Application n	a me: Blir	nk LEDs
	onchip) Sensor Data to AWS Cloud (MQTT onchip)	Description: 1 to verify hardw configurations	This appli ware and s, GPIO fu	cation toggles the board LEDs in sequence at regular intervals software setup. Enabling verification of microcontroller inctionality, and timing control
	Sensor Data to AWS Cloud (HTTPS	Market Segm	ent: IoT	
	onchip) Sensor Data over BLE	OS: Baremetal	nt: Other	
	Baremetal		-	MAC COOR
	Sensor Data to AWS Cloud (MQTT onchip)	1	10	
A Pro	Sensor Data over BLE			File Downloader 🛛 🗸 🚍 🔓
	Blink LEDs			



5. Click on Build Project. The build progress is displayed in the output log.



6. After the project is built, a notification appears stating "QCStudio project build finished successfully". The **Readme.md** file opens for further instructions.

R	File Edit Diagram Selection \	riew Go Run Help		
$\leftarrow \rightarrow$	→│□□│ᡛᠯ᠈᠀᠉ᢦ᠊□			7:
ß	EXPLORER ····	🖁 Blinky.qcc 🖸 Preview README.md 🗴 🖒	QCC TOOL PALETTE	93
G	> OPEN EDITORS	Blinky > bgk_ra6e2_baremetal > ^{BB} README.md	No active QCC GLSP Editor	~
い い い の い の の の の の の の の の の の の の の	 > WORKSPACE > Dubut > Dubug > Drag > Package > Package<th>Overview This application project blinks onboard LEDs in sequence. Instructions For Running the Application 1. Build your QCStudio project. Click on spanner icon (3rd icon from top left). 2. After successful build, download the srec file from the project debug folder to your local PC. 3. Open J-Flash Lite application in your local PC, choose the MCU that matches in your QCStudio project. 4. Select the srec file and program the kit. 5. User can configure Segger RTT Viewer to view the project log. 6. Run the application. References 1. J-Link / J-Trace Downloads: https://www.segger.com/downloads/jlink/ A Problems ① Output X File Downloader v 冨 合</th><th>NO active QCC GCSP confor</th><th>?</th>	Overview This application project blinks onboard LEDs in sequence. Instructions For Running the Application 1. Build your QCStudio project. Click on spanner icon (3rd icon from top left). 2. After successful build, download the srec file from the project debug folder to your local PC. 3. Open J-Flash Lite application in your local PC, choose the MCU that matches in your QCStudio project. 4. Select the srec file and program the kit. 5. User can configure Segger RTT Viewer to view the project log. 6. Run the application. References 1. J-Link / J-Trace Downloads: https://www.segger.com/downloads/jlink/ A Problems ① Output X File Downloader v 冨 合	NO active QCC GCSP confor	?
	README.md	2025-02-05109:52:22:4402 All QCS processes completed. 2025-02-05T09:32:22.478Z Generation and build finished successfully.		
	> 🖻 build-workspace	2025-02-05T09:32:22.4802 Write generated configuration to lock file /home/qcstudio/workspace/Bli 2025-02-05T09:32:22.4827 OCStudio project build finished successfully!	r	
553	Blinky.qcc			
⊗ 0	A 0		¢.	1 🗖

7. Go to Explorer, expand the project, expand **Debug**, select the .srec file, right-click, and select **Download**.

🔒 File Edit Die					
$\leftarrow \rightarrow \square \square_{+}$	P x - 🗆				7
		Physical Control Control Preview READMErnd Blinky > bgk_rasie2_baremetal > Debug > >	D bgk_ra6e2_baremetaLirec × D bgk_ra6e2_baremetaLirec		occ 1000, MALETTE
	lata	Open Open With_	**17265606574616C2E73726563CA 300601107060074 700601107060087A 300601107060003		- 9
🖁 🗸 🖾 bgk_ra	6e2_baremetal	Select for Compare	700000100000009		
→ En settin En → En build → En Debu	nge	Copy Ctrl+C Paster	303811847704761 3006006480749A9 303D0044808819E		
	pen .	Copy Path Alt+Shift+C Copy Relative Path Ctrl+K, Ctrl+Shift+C Copy Download Link	300200300002000 34380FFF708FFF 901232370108055 3000000865453157		
A ∎ bgk ■ bgk D bgi	raõe2 beremetal0.build.st ;raõe2_baremetal.elf K raõe2_baremetal.elf.in k raõe2_baremetal.elf.in	Upload Files Download	000F0008553252577 300709C0A0020075 700EFT2C00002300A 31068ELA4B1A0EAE	-	
D bgk D bgi	Cra6e2_baremetaLrpd &_ra6e2_baremetaLsbd	Duplicate F2	384F803E01A68F2 34FEA1022144933 322F9811222F4A6	/etc-an	
D bgk Br mak D mak II mer Br obje	r_raGe2_baremetal.srec sehle kefile.init mory.regions.id ects.mk rec.s.mk	 53150000012040420C8F42F095024 531500000130EFF31082072781F31 5315000001400131040FF80808 53150000015082F31088106000E8 53150000015082608104FF47A710 531500000150802608104FF47A710 531500000150802608104FF47A710 	2F 0640241F 8202092 08831964118119053 08904062787045 0800402000006068083 06F 00500000681083 06F 00500002306055AE 77 060F 060F 180100AT		
עמיים היים (היים (היים (ices.mk g en	A Problems	processes completed. on and build finished successfully.	File Downloader v	= 6
) 면 script 양) 면 src		2025-02-05109132:22.4092 Mrite ge 2025-02-05T09:32:22.482Z QCStudio	project build finished successfully!	ace/dlinky/.qcc.com.igurac	100.10

6.2 **Programming Hardware and Viewing Results**

1. Connect the BGK-RA6E2 to a laptop using a USB cable. Refer to the section, Flashing Code to the Hardware using SEGGER J-Flash Lite, to flash the code.



2. Observe the output - LED 1 and 2 blinking in a pattern.



7. Project 2 - Smart Temperature Data Logger

7.1 Steps to Create the Project

1. Follow the steps in the Getting Started section to create a new workspace and a project.



2. Drag and drop the MCU board from the QCC tool palette. In this example, BGK-RA6E2 is used.

FR	File Edit Diagram Selection View Go	Run Help		
← -	> □ ¤ » » - □			<i>'i</i> :
n l	EXPLORER ····	🖧 Smart_Temperature_Data_Logger.qcc 🗙	QCC TOOL PALETTE	
6	> OPEN EDITORS	Smart_Temperature_Data_Logger > configuration > 🖾 Smart_Temperature_Data_Logger.qcc		
9	WORKSPACE	APPLICATION CONFIGURATION		50
0.	Smart_Temperature_Data_Logger.qcc		SHOWING ALL AVAILABLE DEVICES	
Pr.			V RENESAS	
		Please select an application	CONNECTIVITY MODULES	
~		✓ Market Segment	C DA14531 Bluetooth	
B			🛆 DA16200 Wi-Fi	
π		от > 1999292 8 8 0 1117 1999295	🖒 DA16600 Wi-Fi	
A			V SENSORS	
			FS1015 Vertical mount Air velocity	
		Please select an application from the list above.	6 FS3000 Horizontal mount Air velocity	
			HS3001 Temperature / Humidity	
			B HS4001 Temperature / Humidity	
			Ø ZMOD4410 Indoor Air Quality	
			O ZMOD4450 Refrigeration Air Quality	
			⑦ ZMOD4510 Outdoor Air Quality	
		A Problems	✓ PARTNER	
		2025-02-05109:32:22.440Z All QCS processes completed.	✓ SENSORS	
		2025-02-05109:32:22.4782 Generation and build finished successfully. 2025-02-05T09:32:22.4802 Write generated configuration to lock file /home/qcstudio/workspace/Blinky	AS7331 Spectral UV	
-		2025-02-05T09:32:22.4822 QCStudio project build finished successfully!	AS7343 Spectral	
⊗0	Δ.0			Ø1 🖬

3. In the Application Configuration window, select market segment as **IoT**, then **Other**, and finally **Sensor Data to AWS Cloud (MQTT onchip)**.

-					_
R					
← -	→ 🖽 ⊑‡ ∥ 👷 🖵				;
Ø	EXPLORER ····	Smart_Temperature_Data_Logger.qcc	X Marvation)問 Smart Temperature Data Longer.com	GETTING STARTED	
Q	V WORKSPACE V El Smart_Temperature_Data_Logger/co	APPLICATION CONFIGURATION		Getting Started With	
fo	2 Smart_Temperature_Data_Logger.qoc	Search application		QUICK-CONNECT STUDIO	C
a⊳		< Application		Please see the sections below to get an overview of the available features and use the blue links to directly see them in action.	
₿		EreapTOS		Alternatively, please use the toolbar on top, right below the main menu bar. e following feature sections describe the needed	
Δ		Sensor Data to AWS Cloud (MC onchip) Sensor Data over BLE	Application name: Sensor Date to AWS Cloud (MQIT onchip) Description: This application collects real-time sensor data and transmits it to AWS 1 Core using the DA16200/DA16600 on-chip MQIT stack over WiFi for monitoring and	ps and also offer an action to execute those steps tomatically, they are indicated by the play icon (of	
		Baremetal Sensor Data to AWS Cloud (MC	Visualization. Market Segment: 107 End Equipment: Other	Create QCStudio Configuration Project	
		Sensor Data over BLE Blink LEDs	OS: FreeRTOS	Sibar icon E.4 or click the action item below. This I open a quick input field to define a project name. be a name, e.g. "Test123", into the window and hit The action of the state of t	
				separate project folder in the currently opened workspace. Upon successful creation, the diagram edites will be opened automatically.	
		A Problems 🗐 Output X	File Downloader 🛛 🗸 🛱 🔓	 Create QCStudio Configuration Project 	
~~				፼ Diagram Editor	
563				The diagram editor allows you to configure a board	
80					φE

4. Drag and drop the connectivity module DA16600

$\leftarrow \rightarrow \square \sqcap \mathscr{P} \rtimes \blacktriangledown \square$	17:
EXPLORER ···· & Smart_Temperature_Data_Logger.qcc × QCC.TOOL PALETTE > OPEN EDITORS Smart_Temperature_Data_Logger > configuration > [2] Smart_Temperature_Data_Logger.qcc	45
V WORKSPACE V Di Smart, Temperature, Data_Logger, Cc. Smart, Temp	9 ()
C DA16200 Wi-Fi C DA16600 Wi-Fi ↓ SENSORS ↓	
Composition of the state o	ocity velocity lity lity Quality ity
	0.8

a. Based on the chosen board and application, the QuickConnect tool automatically suggests the PMOD connector compatible with the wi-fi module board.



5. Drag and drop the humidity and temperature sensor board.

ER	File Edit Diagram Selection	View Go Ru	ın Help					
← -	→ □ □ □ □ ₽ ₽ ₩ - □							<i>'i</i> :
G	EXPLORER	ጸ	Smart_Temperature_Data_Lo	ogger.qcc X			QCC TOOL PALETTE	23
	> OPEN EDITORS V WORKSPACE V CI Smart_Temperature_Data_1 Smart_Temperature_Data	1 2 2 2 2 2		RELATIVE H • Prm • RH to - 0.3j	UMIDITY SENSOR PMOD [®] BOAI d ¹¹¹ ports type 6A (12C) accuracy: 1-15/8KH ±0.2°C, -80 °C) Very low current cons LA average (8-bit resolution,	RD typical (-10 °C sumption: 3.3V supply)	Search tool palette items RHOWING ALL AVAILABLE DEVICES V RENESAS Ø FS1015 Vertical mount Air velocity Ø FS3000 Horizontal mount Air velocity Ø KE2001 Transporture / Laumidite	^م ٩
д							HS4001 Temperature / Humidity	
			****				 ZMOD4410 Indoor Air Quality ZMOD4450 Refrigeration Air Quality ZMOD4510 Outdoor Air Quality PARTNER 	
			Problems 🗐 Output		File Downloader	~ ≡6	✓ SENSORS	
							AS7331 Spectral UV	
							Ø AS7343 Spectral	
563							FECS43 SO2 Gas Sensor	
.⊗0	▲ 0							Q 🗖

6. To generate and build project, click on the Generate/Build QCS Project icon on the top left-hand side corner. QuickConnect Studio automatically generates the required software package including drivers, middleware, and network stacks required for the user-created system solution.



7. To run the application project, refer to the instructions in the **README.md** file located in the project directory. In this example, the Wi-Fi SSID and password must be entered, along with the username and AWS endpoint, should be edited in user.h.



8. Additionally, certificate.h must be updated by adding the device certificate, private key, and Amazon root CA. Refer to the workflow below for setting up an MQTT Thing in AWS IoT Core.



 The user.h file must be updated with Wi-Fi SSID, Wi-Fi password, IO Username, and MQTT host (AWS data endpoint). For AWS endpoint, Login to the AWS account > select AWS IOT > Go to Settings > Expand Domain configuration > Copy domain Name.

FR	File Edit Diagram Selection '	View Go Run Help	0
← -	→ □ □ □ ₽ ₽ ₩▼ □		7:
Ch	EXPLORER ···	& Smart_Temperature_data_Logger.qcc IIII Preview README.md Course A 3 X QCC TOOL PALETTE	23
-	> OPEN EDITORS	Smart_Temperature_data_Logger > bgk_ra6e2_freertos > src > C user.h	
Q		1 /* Convright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates	5
-0	✓ ➡ Smart_Temperature_d ●		
20	> 🗂 .metadata		$\widehat{\mathbf{O}}$
•	✓ ➡ bgk_ra6e2_freertos		\sim
N	> 🖸 .settings	b #linder User_H	
æ~	> 🖸 build		
- 0	> 🖸 Debug	9 #define WIFI_SSID ("MyWiFiName"	
田		10 #define WIFI_PWD "Name@123"	
	> En ra_crg	11 12 Hdofine TO LISERNAME "Moderate"	
A	> P1 erriet	12 wdefile to_oschwere Pyrtoject	
		14 #define EXAMPLE_NQTT_HOST "a2a04u7example-ats.iot.us-east-1.amazonaws.com"	
	> P1 oc-middleware		
	> 🖻 sensor	16 #define USER_LED_TOPIC "/feeds/led"	
	C certificate.h	1/ 18 #endif /* USER H */	
	C hal_entry.c		
	C main_thread_entry.c		
	c sensor_thread_entry.c	A Problems ③	
	🗅 sm_define_sensors.i	$\omega \omega \omega$ of Antalognetic sector of a caloring out concentration of a concentration of the sector of	
	C user.h 3	2025-04-24112:54:47.5632 2025-04-24 12:54:47.302 - ease create projects py - INFO - Logs available for later	
	• .api_xml	2025-04-24T12:54:49.8102 2025-04-24 12:54:49,771 - ease_create_projects.py - INFO - All projects processed wi	
	cproject	2025-04-24T12:54:52.3012 Number of projects: 1	
	• .generated	2025-04-24112:54:52.3112 NUMBER OF ELTS : 1	
220	project	2025-04-24112:54:52:3962 [Generation and build finished successfully.	
563	 secure_azone 	2025-04-24T12:54:52.398Z Write generated configuration to lock file /home/qcstudio/workspace/Smart_Temperatur	
	secure_xml	2025-04-20112-54-52 1007 OCStudio project huild finished successfully	

Note: Renesas recommends not to edit the User_LED_Topic or define the sensor topic, as the tool automatically takes care of it.

10. Update the downloaded certificates in the **certificate.h** file. Make sure to download the Root CA, Client Certificate (that is, device certificate), and Private Key file. Additionally, ensure that the macros are edited before uploading.



Note: The certificate values or lines provided in above screenshot are just for sample. Refer to the attached screenshot only to ensure the correct format.

11. After making changes in the corresponding .c and .h files, rebuild the application project by selecting **Build QCStudio Project** from the drop-down menu.



7.2 Programming Hardware and Viewing Results

- 1. Follow the steps in the section, Steps to Create the Project, to create the solution.
- 2. The application project output files can be found under the bgk_ra6e2_freertos > Debug folder.
- 3. Right-click on the .srec file and download it to the local PC.

EXPLORER	··· D bgk_ra6e2_freertos.srec ×		GETTING STARTED
> OPEN EDITORS WORKSPACE Smart, Temperature_Data_Log Entradata Elbgk_ra6e2_freentos Elbuid Debug Elbuid Debug Elbuid Bigk_ra6e2_freentos.0buik Bigk_ra6e	Select for Compare Copy Copy Ctrl+C Paste Copy Relative Path Alt+Shift+C Copy Relative Path Cuplcad Files Doubleate Doubleate Delete Delete Delete Paname F2	+ bgk_ra6e2_freetos >> Debug >> D bgk_ra6e2_freetos.stec ***13665325F6672656572746F7322755630E \$\$55480000514600006148000016 \$\$300000000000000000000000000000000000	Getting Started With QUICK- CONNECT STUDIO Please see the sections below to get an overview of the available features and use the blue links to directly see them in action. Alternatively, please use the toolbar on top, right below the main menu bar. The following feature sections describe the needed steps and also offer an action to execute those steps automatically, they are indicated by the

4. Connect the BGK-RA6E2 to the laptop using a USB cable. Refer to the section, Flashing Code to the Hardware using SEGGER J-Flash Lite, to flash the code.



- 5. Use Jlink Flash programmer to program the **.srec** file into the required MCU kit (QuickConnect Beginner kit). Reference section, Flashing Code to the Hardware using SEGGER J-Flash Lite, to flash the code.
- 6. View the output on the AWS MQTT test client by subscribing and publishing the data.

Steps to be followed:

- a. Log in to the AWS account and open <u>AWS IoT Core</u>. Follow the instructions according to **README.md** to create a thing.
- b. In the left console, under **Test**, click on **MQTT test client**. A window appears for subscribing and publishing a topic.



Note: Ensure by subscribing to the topic # and check the connection details.

c. In the topic filter, enter "UserName/feeds/humidity" and click **Subscribe** to see the humidity values.

C Search	[Alt+S]	▶ 🕹 ⑦ 🔞 United States (N. Virginia) ▼	Vin
WS IoT ×	AWS IoT > MQTT test client		
onitor	MQTT test client Info		
ionnect	You can use the MQTT text client to monitor the MQTT messages being passed in your AWS account. Devices publish MQTT messages that are idd messages to inform devices and apps of changes and events. You can subscribe to MQTT message topics and publish MQTT messages to topics b	sentified by topics to communicate their state to AWS IoT. AWS IoT also publishes MQTT by using the MQTT test client.	
Connect one device Connect many devices Domain configurations Updated	Connection details Connected To disconnect from the MQITT test client, choose Disconnect. To re-establish a connection, you can update the details and choose Connect.	Disconnect]
set	Subscribe to a topic Publish to a topic		
Device Advisor MQTT test client Device Location New Query connectivity status New	Topic filter : info The trajke filter can include MQTT wildcard characters. (MyProject/redsr/humidity) > Additional configuration Subscrite		
anage All devices Greengrass devices	Subscriptions Topic		
LPWAN devices Software packages <u>New</u>	You have no topic subscriptions. Subscriptions.	incoming messages.	
Remote actions Message routing			
Retained messages Security			
Fleet Hub			
CloudShell Feedback		© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Co	ookie pref

d. Similarly, enter "UserName/feeds/temperature" and click **Subscribe** to see the temperature values.

aws Services Q Search	[Alt+S	a	Þ.	¢	0	0	United States (N. Virginia) 🔻 Vinutha 🔻
Monitor	You successfully deleted 4 things.							× ©
Connect Connect one device Connect many devices Domain configurations Updated	AWS IoT > MQTT test client MQTT test client Info You can use the MQTT test client to monitor the MQTT me communicate their state to AWS IoT. AWS IoT also publish and publish MQTT messages to topics by using the MQTT to	ssages being passed in your AWS account. Devices put s MQTT messages to inform devices and apps of char est client.	blish MQ nges and	TT mes events	sages tl . You ca	nat are io n subscri	dentified by topics to ibe to MQTT message to	pics
Test ▶ Device Advisor MQTT test client	Connection details Connected To disconnect from the MQTT test client, choose Disconnect.	To re-establish a connection, you can update the details and d	hoose Con	inect.			Disconne	ct
Device Location <u>New</u> Query connectivity status <u>New</u>	Subscribe to a topic Publish to a topic Topic fitter Info The topic fitter describes the topicfol to which we want to unknown	ibu. The tool-filter can include MATT wildcard characters						
Manage ♥ All devices Thinos	MyProject/feeds/temperature Additional configuration	and the topic medical minable right indeals dranetera.]			
Thing groups Thing types	Subscribe							
Greengrass devices I PWAN devices	Subscriptions Topic							
Software packages New Remote actions Message routing	You have no topic subscriptions.	Subscribe or select a topic to view	w incomi	ng mes	sages.			
Security		@ 2025, Am	nazon Web	Service	s, Inc. or i	ts affiliate	is. Privacy Terms	Cookie preferences

Note: The username should be the same as what was entered in the **user.h** file in the QuickConnect Studio.

e. Observe the output humidity and temperature values.

aws iii Services Q. Search		[Alt+S]	🔁 💠 🧭 🗐 United States (N. Virginia) 🕶 V	/inutha 🔻
AWS Int Y	The topic filter describes the topic(s) to which yo	wwant to subscribe. The topic filter can include MQTT wildcard characters.		0
	MyProject/feeds/temperature			
Monitor	Additional configuration			
Connect	Subscribe			
Connect one device				
Connect many devices	Subscriptions	MyProject/feeds/humidity	Pause Clear Export Edit	
Domain configurations Updated	MyProject/feeds/humidity	Message payload		
	MyProject/feeds/temperature 🌣 🗙	["message": "Helio from AWS IoT console"		
Test)		
Device Advisor		 Additional configuration 		
Device Location New		P Additional configuration		
Query connectivity		Publish		
status New				
Manane		 MyProject/feeds/humidity 	February 19, 2025, 16:45:54 (UTC+0530)	
All devices		_		
Greengrass devices		50.01		
LPWAN devices				
Software packages New		Properties		
Remote actions				
Message routing		 MyProject/feeds/humidity 	February 19, 2025, 16:45:24 (UTC+0530)	
Retained messages				
Security				
▶ Fleet Hub		50.24		

WS III Services Q S	earch		[Alt+S]	ව 🗘 🖉 🕲 United States (N. Virginia) ▼ Vinutha
AWS IoT	×	Subscriptions	MyProject/feeds/temperature	Pause Clear Export Edit
Monitor		MyProject/feeds/humidity 🗢 🗙	Message payload	
Connect		MyProject/reeds/temperature	"message": "Hello from AWS IoT console" }	
Connect one device			Additional configuration	
Domain configurations Updated			Publish	
Test			 MyProject/feeds/temperature 	February 19, 2025, 16:46:24 (UTC+0530)
Device Advisor MQTT test client			26.61	
Device Location New Query connectivity status New			▶ Properties	
fanage			MyProject/feeds/temperature	February 19, 2025, 16:45:54 (UTC+0530)
All devices Greengrass devices			26.60	
Software packages New Remote actions			Properties	
Message routing Retained messages			MyProject/feeds/temperature	February 19, 2025, 16:45:24 (UTC+0530)
Security Fleet Hub			26.58	

f. The feature to publish to a topic can also be used by giving the topic name as "UserName/feeds/led" with the message payload set to 0 to turn off the LED and 1 to turn on the LED.

Services Q Search	cient.	[Alt+S]		Inited States (N. Virginia) 🔻 Vinutha 🔻
 Connect many devices Domain configurations Updated 	► Connection details To disconnect from the MQTT te	Connected st client, choose Disconnect. To re-establish a	connection, you can update the details	Disconnect and choose Connect.
Test Device Advisor	Subscribe to a topic	lish to a topic		
MQTT test client Device Location New	Topic name The topic name identifies the messa	ge. The message payload will be published to	this topic with a Quality of Service (Qo	S) of 0.
Query connectivity	Q MyProject/feeds/led		×	
Status Inter	Message payload			
Manage	1			
All devices			4	
Greengrass devices	Additional configuration			
LPWAN devices Software packages New	Publish			
Remote actions				
Message routing				
Retained messages	Subscriptions	MyProject/feeds/te	emperature	

8. Project 3 - BLE-based Asset Tracker

8.1 Steps to Create the BLE-based Project

1. Follow the Steps to Create the Project section to create a new project.



2. Drag and drop the MCU from the QCC tool palette. In this example, BGK-RA6E2 is used.



3. Choose the Application Configuration. In the **Application Configuration** window, first select **IoT**, **Other**, and finally **Sensor Data over BLE**.



4. Choose the Bluetooth module DA14531.

ER	File Edit Diagram Selection View Go Run	Help			
\leftarrow	→ Ш ⊑ 10 % ▼ 🖓				7.
Ω	EXPLORER ····	℅ BLE_based_Asset_Tracker.qcc ×		QCC TOOL PALETTE	23
_	> OPEN EDITORS	BLE_based_Asset_Tracker > configuration > 🔡 B	LE_based_Asset_Tracker.qcc	Search tool palette items	
9	✓ WORKSPACE ✓ ☐ BLE_based_Asset_Tracker / configuration	APPLICATION CONFIGURATION		SHOWING ALL AVAILABLE DEVICES	59
zo	BLE_based_Asset_Tracker.qcc			∽ RENESAS	1
N		loT / Other		✓ CONNECTIVITY MODU	
\$ ^{>}				△ DA14531 Blue	
<u>д</u> р				SENSORS	
ц.,			 Pmod[™] ports type 3A (UART) 	Ø FS1015 Vertica	
A			 Integrated chip antenna Bluetooth 5.1 core gualified IoTMark[™] - BLE score of 18300 	@ FS3000 Horizo	
		o" #	+2.2dBm maximum output power -93dBm	@ HS3001 Temp	
			sensitivity Optional battery operation with onboard 	@ HS4001 Temp	
			CR1220 coin cell	@ ZMOD4410 In	
				@ ZMOD4450 Re	
				@ ZMOD4510 O	
				✓ SENSORS	
53				@ AS7331 Spectr	
00					

a. Based on the chosen board, the QuickConnect tool automatically suggests the PMOD connection compatibility



5. Drag and drop the Humidity and Temperature sensor in the reference application (HS4001 is used).



6. To generate and build project, click on the Generate/Build QCS Project icon on the top left-hand side corner. QuickConnect Studio automatically generates the required software package including drivers, middleware, and network stacks required for the user-created system solution.



8.2 **Programming Hardware and Viewing Results**

- 1. Follow the steps in the Steps to Create the Project section to create the solution.
- 2. The application project output files is found under the **Debug** folder. Right-click on the **.srec** file and download it to a local PC.

R	File Edit Diagram Selection V	/iew Go Run Help						
← -	→ 🔟 🛱 🖉 🎘 🖵 🗔							<i>7</i> :
Ch	EXPLORER		based_Asset_Tracker.qcc	Preview README.md ×		ъ	GETTING STARTED	90
	> OPEN EDITORS	BLE_b	ased_Asset_Tracker > bg	k_ra6e2_freertos > 🛤 README.md				
く や 日 日	 > OFFNE DOTORS > WORKSPACE > BLE Dassed_Asset_Tracker > Bok_ra6e2_freetros > Bok_ra6e2_freetros > Dobug > Dobug > Dasset > D	Deen Open With Select for Compare Copy Paste Copy Path Copy Relative Path Copy Relative Path Copy Download Link Upload Files Download Delete Rename	sed_Asset_Tracker > bg Ctrl+C Alt+Shift+C Ctrl+K, Ctrl+Shift+C Ctrl+K, Ctrl+Shift+C Delete F2 5. User can configure Set items	K_na6e2_treertos > ™ READMErnd cases BLE connectivity use case. In user's cellphone/tablet will act as and sensor data through 12C interf e. e Application oject. Click on spanner icon (3rd i download the srec file from the pr cation in your local PC, choose the program the kit. kit will start BLE Advertising packe gger RTT Viewer to view the project	this application, the RA MCU kit BLE Central. In this application ace and publish the sensor data con from top left). oject debug folder to your local MCU that matches in your t and the green LED will blink. t og.	16	Getting Started With QUICK- CONNECT STUDIO Please see the sections below to get an overview of the available features and use the blue links to directly see them in action. Alternatively, please use the toolkar or top, right below the main menu bar. The following feature sections describe the needed steps and also offer an action to secucit	(i) (i)
603 603 00	iii memory_regions.ld ♥ objects.mk ♥ sources.mk > 户 ra						those steps automatically, they are indicated by the play icon (🕑).	

3. Use the J-link Flash programmer to program the **.srec** file into the desired MCU kit. Refer to the Flashing Code to the Hardware using SEGGER J-Flash Lite section.



- 4. View the output on QC Sandbox by downloading the QC Sandbox app.
- 5. As soon as the application is open, it asks you to select either **Bluetooth Low Energy (BLE)** or **Wi-Fi Provisioning**. For this reference example, select **Bluetooth Low Energy (BLE)**.

17:43 🕥 🎘 🛊 🌾 🗃	() a "a	il CD
RENES	ΔS	i
🛞 Bluetooth Low Ene	ergy (BLE)	
Wi-Fi Provision	oning	
	\leq	

- 6. To view the output:
 - b. Turn on Bluetooth on the device.



7. Scan to connect and click on the **Connect** button.

17:44 🕥 🏞 🕈 🤄 🖻	() * # ⁵ 1	ıl 🕼		
RENESAS i				
(
US000-QCS				
RSSI: -40	Connect			
Adv Timestamp	17:44:08:198			
Local Name	US000-QCS			
\equiv				

8. Select the LED option on the QC Sandbox app and toggle the LED switch to turn it on and off. Observe the output.



9. Select the sensor **HS4001** on the mobile app to see the humidity and temperature sensor values.



9. Debugging on QuickConnect Studio

9.1 Remote Debugging

Projects can be debugged remotely when it is preferred to validate the generated reference application onto the target hardware before obtaining the boards, and remote board farms deployed globally can be used. The remote debugging feature is supported for the QuickConnect beginners Kit for Blinky, sensor data over BLE, and sensor data over AWS applications.

At this stage, it is assumed the user followed the steps to successfully build a project on QuickConnect.

Follow the proceeding steps for remote debugging.

- 1. Build the application.
- 2. To remotely connect to the remote boards, click on the Debug icon as follows:



3. Enable the debug console window to capture the debug log message when connecting to the remote board farm. Click the **View** option and select **Debug Console**.

R	File Edit Diagram Selection	View Go Run Terminal Help	
< -} 		Command Palette Open View	Ctrl+Shift+P
G		Appearance	>
0	test_ckra6m5 - Renesas Cl	Editor Layout	>
\sim		Call Hierarchy	Ctrl+Shift+F1
የያ	> THREADS	Debug	Ctrl+Shift+D
6	> CALL STACK	Debug Console	Ctrl+Shift+Y
	> VARIABLES	Explorer	Ctrl+Shift+E
£	> WATCH	Extensions	Ctrl+Shift+X
	> BREAKPOINTS	History	Alt+H
Ш		Outline	Ctrl+Shift+I
_		Output	Ctrl+Shift+U
Å		Plugins	
		Problems	Ctrl+Shift+M



4. It is important to select the project and configuration that requires debugging.



5. The QCStudio platform automatically generates the launch.json with the debug configuration settings for the targeted hardware used in the design. To check the launch.json file, click on the settings icon. *Note*: Ensure that the selected configuration is correct in launch.json. If there is any old project remnant data, clear the launch.json, rebuild the project, and select the correct configuration.



6. Click on the green play button as to initiate the debug connection to the remote board farm. The QCStudio platform automatically routes the user to the nearest board farms that have the requested board configurations.

Note: If the board is not available or cannot establish a connection, it might be because the board is in use by others and currently unavailable. Try using it at a later time.



- 7. On successful connection to the remote board, the QCStudio platform programs the board with the application image and halts on the **main.c** file.
- 8. After a debug session starts, the Debug toolbar appears supporting the following actions.

= ▶ ? ¥ 1 Ð ■

Action	lcon	Explanation
Continue/Pause		Continue: Resume normal program/script execution (up to the next breakpoint). Pause: Inspect code executing at the current line and debug line-by-line.
Step Over	7.	Execute the next instruction as a single command without inspecting or following its component steps.
Step Into	¥	Enter the next instruction to follow its execution line-by-line.
Step Out	1	When inside a method or subroutine, return to the earlier execution context by completing the remaining lines of the current method as though it were a single command.
Restart	୯	Terminate the current program execution and start debugging again using the current run configuration.
Stop		Terminate the current program execution

After debugging the reference application, the user must terminate the debug connection by pressing the **Stop** button from the **Debug** toolbar.
10. Appendix

10.1 Flashing Code to the Hardware using SEGGER J-Flash Lite

Download the Segger software to the local PC if it is not already installed. According to the device compatibility, choose the installer under J-Link Software and Documentation Pack, Renesas recommends installing the 64-bit installer.

- 1. Open SEGGER J-Flash Lite:
 - a. Navigate to the **Program Files** on the PC.
 - b. Open the SEGGER Jlink folder.
 - c. Launch JFlashLite.exe.
- 2. Select Target Device:
 - a. In the J-Flash Lite window, click on the (...) button next to the Target Device field.
 - b. A new window appears. Here, the user can select the manufacturer and device.
 - c. For this project, the RA6E2 MCU is used, search for the part number R7FA6E2BB.
 - d. Select the target device and click OK.
 - e. Ensure the target interface is set to SWD.
 - f. Click OK.

			Manufacturer	Device	Core	NumCores	Flash Size
Farget device		~R7FA6		~ Filter		Filter	
			Renesas	R7FA6E2B9_RAMLess	Cortex-M33	1	512 Bytes + 128 KB + 4 KB + 6
R7FA6E2BB		Renesas	R7FA6E2D9	Cortex-M33	1	512 Bytes + 128 KB + 4 KB +	
		Renosas	R7FA6E2DB RAMLocs	Cortex-M22	1	512 Bytes + 256 KB + 4 KB + 6	
larget Interface Speed			Ren sas	R7fA6E268	Cortex-M33		512 Bytes + 256 KB + 4 KB + 6
			Renesas	R7FA6E10F	Cortex-M33	1	512 Bytes + 1 MB + 8 KB + 64
SWD • 4000	~	kHz	Renesas	R7FA6E10F_RAMLess	Cortex-M33	1	512 Bytes + 1 MB + 8 KB + 64
			Renesas	R7FA6E10D	Cortex-M33	1	512 Bytes + 512 KB + 8 KB + 6
Elach hanks		Renesas	R7FA6E10D_RAMLess	Cortex-M33	1	512 Bytes + 512 KB + 8 KB + 6	
Idshi Ddiliks			Renesas	R7FA6M1AD	Cortex-M4	1	32 Bytes + 512 KB + 8 KB + 64
Dess & dds Manus	Landar		Renesas	R7FA6M2AD	Cortex-M4	1	32 Bytes + 512 KB + 32 KB + 6
BaseAddr Name	Loader	~	Renesas	R7FA6M2AF	Cortex-M4	1	32 Bytes + 1 MB + 32 KB + 64
Co.01000100 Teterral antian astrian memory	Default	21	Renesas	R7FA6M3AF	Cortex-M4	1	32 Bytes + 1 MB + 64 KB + 64
✓ uxu1uuA100 Internal option-second memory	Delauit		Renesas	R7FA6M3AH	Cortex-M4	1	32 Bytes + 2 MB + 64 KB + 64
✓ 0x00000000 Internal program flash	Default	*	Renesas	R7FA6M4AD	Cortex-M33	1	512 Bytes + 512 KB + 8 KB + 6
			Renesas	R7FA6M4AD_RAMLess	Cortex-M33	1	512 Bytes + 512 KB + 8 KB + 6
✓ 0x08000000 Internal data flash	Default	•	Renesas	R7FA6M4AE	Cortex-M33	1	512 Bytes + 768 KB + 8 KB + 6
0v6000000 External OSPI flach	CI K@D100 m		Renesas	R7FA6M4AE_RAMLess	Cortex-M33	1	512 Bytes + 768 KB + 8 KB + 6
OXOOOOOOO EXternal Q3P1 hash	CDK@P100_IF		Renesas	R7FA6M4AF	Cortex-M33	1	512 Bytes + 1 MB + 8 KB + 64
			Renesas	R7FA6M4AF RAMLess	Cortex-M33	1	512 Bytes + 1 MB + 8 KB + 64

3. Import the .srec File:

- a. In the main J-Flash Lite window, locate the Data File (bin / Hex / mot / srec / ...) section.
- b. Click on the (...) button to import the .srec file.
- c. Select the .srec file that was downloaded by following the steps in the Quick Start procedure.

🔝 SEGGER J-Flash Lite V7.98c — 🗌 🗙	🔜 SEGGER J-Flash Lite V7.98c — 🗌 🗙
File Help	File Help
Target Speed Device Interface Speed R7FA6E2BB SWD 4000 kHz	Target Speed Device Interface Speed [R7FA6E2BB [SWD [4000 kHz
Data File (bin / hex / mot / srec /) Erase Chip	Data File (bin / hex / mot / srec /) 7681\Downloads\bgk_ra6e2_freertos.srec
Program Device	Program Device

- 4. Program the Device:
 - a. Click on Program Device.
 - b. A prompt can appear asking if an update to the latest firmware version is required. Select No.
 - c. The code now flashes to the MCU.

d. After the process is complete, the log section of the screen displays Done.

🔜 SEGGER J-Flash Lite V7.98c —				\times			
File Help							
Target Device Interface SWD		Speed 4000 kHz	:				
Data File (bin / hex / mot / srec /) Frase /) 7681\Downloads\bgk_ra6e2_freertos.srec							
Program De	vice						
Log							
Log #3: 0x0100A134 - 0x0100A137 (4 Bytes) #4: 0x0100A200 - 0x0100A2CB (204 Bytes) Connecting to J-Link Bank selection: BankAddr=0x0100A100 Enabled Bank selection: BankAddr=0x0000000 Enabled Bank selection: BankAddr=0x0000000 Disabled Loader selection: BankAddr=0x0100A100 Loader=Default Loader selection: BankAddr=0x0000000 Loader=Default Bank Selection: BankAddr=0x0000000 Loader=Default Bank Selection: BankAddr=0x0000000 Loader=Default Bank Selection: BankAddr=0x0000000 Loader=Default Bank Selection: Bank Bank Bank Bank Bank Bank Bank Bank							
<				>			
Ready							

10.2 Enable Data log

RTT debug logging is a debugging method that is already available in the project generated from the QCS. Enabling this will provide some predefined debugging points through which mistakes in the project can be identified. This is basically some serial messages printed from the application. Complete the following steps to enable this:

1. Go to the **src/main_thread_entry.c** file and make the following changes in the header file inclusion section.



This essentially enables the debug messages on this file. The same can also be followed for sensor files (src/sensor/hs4001_sensor.c).

2. Re-compile the project by clicking **Build QCStudio Project**.



- 3. Go to **Debug** folder in the repository and open the ***.map** file. This file has the same name as the **srec** file, but the extension is **.map**.
- 4. Inside the file, search for the word **_segger** and the following appears:

R	File Edit Diagram Selection View Go Ru								
← -	> Ⅲ ⊑, ∥ % ◄ ♫								
Ð	EXPLORER		Preview README.md	Cocertificate.h 3	Couser.h 3	Comain_thread_entry.c 24	bgk_ra6e2_freerto	is.map X	
の。予問日	 > oren tomors > oren tomors > Wokspace > Charles and the set of the s	Wi-Filest1 > bgk, s_act 2991 bss.act 2992 bss.act 2993 bss.st 2996 bss.gen 2996 bss.gen 2996 bss.gen 2997 2998 2999 bss.sub1 3003 3003 bss.bus_ 3004 bss.bus_ 3006 bss.bus_ 3006 bss.bus_ 3007 bss.bus_ 3008 bss.bus_ 3010 at ss.bus_ 3014 bss.bus_ 3014 bss.bus_ 3014 bss.bus_ 3015 bss.bus_ 3016 bss.bss.bus_ 3016 bss.bss.bss.bss.bss.bss.bss.bss.bss.bss	abc2_freetos > Debug > Deffer 0x2000048c 0x2000048c 0x2000048c 0x2000048c 0x2000048c 0x2000048c 0x2000088c 0x2000088c 0x2000088c 0x2000088c 0x2000088c 0x2000084c 0x2000084c 0x2000044c 0x200004c 0x200004c	D bgirn62_freetosn 0x400 ./src/qc=did 0x40 ./src/qc=did 0x60 ./src/qc=did 0x60 ./src/msin_t 0x60 ./src/msin_t 0x6 ./src/msin_t 0x8 ./ra_gen/com bus_liz_ cing_semptore_handl 0x4 ./ra_gen/com bus_liz_ sive_mutet_memory 0x4 ./ra_gen/com bus_liz_ sive_mutet_memory 0x4 ./ra_gen/com bus_liz_	map dileware/St66ER, Hileware/St66ER, RTT Hrread_entry.o t_onchip_dalox: hread_entry.o s mon_data.o on_sci_lzc0 mon_data.o on_sci_lzc0 mon_data.o on_sci_lzc0 mon_data.o on_sci_lzc0 mon_data.o on_sci_lzc0 mon_data.o	RTT/SEGGER_RTT.o RTT/SEGGER_RTT.o xx_instance ocking_semaphore_memory ocking_semaphore_handl cursive_mutex_memory cursive_mutex_handle			

- 5. A similar address appears in the map file. Copy that address.
- 6. Flash the .srec file to the board.
- 7. Search the PC for the JLink RTT Viewer application and open it.
- 8. Select the three dots by the Target Device selection.

L J-Link RTT Viewer V8.26	_		\times
File Terminals Input Logging Help			
All Terminals Terminal 0			
J-Link RTT Viewer V8.26 Configuration ×			
Connection to J-Link			
USB Serial No			
ОТСР/ІР			
O Existing Session			
Specify Target Device			
R7FA6E28B			
Force go on connect			
Script file (optional)			
Target Interface & Speed			
SWD • 4000 kHz •			
RTT Control Block			
Enter the address of the RTT Control block.			
0x2000088d			
OK Cancel	Enter	Clear	r.

9. On the next window, under Device type A6E2, four devices are listed. Select R7FA6E2BB and click OK.

Terminals Termin	nal 0			
🔝 Target Device	Settings			×
Manufacturer	Device	Core	NumCores	Flash Size
	~ A6E2		∼ Filter	Filter
Renesas	R7FA6E2B9_RAMLess	Cortex-M33	1	512 Bytes + 128 KB + 4 KB + 64 MB
Renesas	R7FA6E2B9	Cortex-M33	1	512 Bytes + 128 KB + 4 KB + 64 MB
Renesas	R7FA6E2BB RAMLess	Cortex-M33	1	512 Bytes + 256 KB + 4 KB + 64 MB
Renesas	R7FA6E2BB	Cortex-M33		512 Bytes + 256 KB + 4 KB + 64 MB

- 10. R7FA6E2BB is now selected as the device in first window (see step 8).
- 11. Under RTT Control Block, select Address and enter the address copied from step 5.

🔜 J-Link RTT Viewer V8	8.26 Configura	tion $ imes$			
Connection to J-Link	Corial No				
Existing Session					
Specify Target Device					
R7FA6E2BB		×			
Force go on connect					
Script file (optional)					
Target Interface & Speed					
SWD	•	4000 kHz 🔹			
RTT Control Block					
Auto Detection O	ddress 🛛	Search Range			
Enter the address of the RTT Control block. Example: 0x20000000					
0x2000088c					
	OK	Cancel			

- 12. Click **OK** and press No if asked for software update.
- 13. A window appears:



10.3 Customization

Customization refers to making modifications to the applications. To perform this exercise, it is assumed that the user has built the basic blink, smart temperature data logger, and BLE-based asset tracker.

10.3.1 Foundation for Customization

To update the **QCS project** with modification within the scope of the application, follow these steps:

- 1. Create the necessary folders in the project to organize and upload the new files.
- 2. Upload the required files to their corresponding paths within the project structure.
- 3. Replace any existing files that need to be updated with the new versions.

10.3.1.1 Create the Required Folders/files

To organize and upload the new files, the user needs to create folders/files as required. For example, the **gpt_timer.c** file should be placed at the path: **bgk_ra6e2_baremetal/src/gpt_timer.c**

However, in the existing Blinky project, this folder structure does currently exist. Therefore, the user must create the **gpt_timer.c file** under the specified path.

A new folder/file can be created by following these steps:

- 1. Click on the project folder (highlighted in the green box).
- 2. Navigate to the target path folder (highlighted in the blue box).
- 3. Right-click on the folder where the new folder needs to be created.
- 4. Select the New Folder/New File option from the context menu (highlighted in the red box in the image).
- 5. Edit the folder/File Name as required.
- 6. After creating the required file at specified path, a user can paste or edit the content in the file as required.





OK

10.3.1.2 Upload the Required Files

C main application.

To upload the required files to the **QCS project**, complete the following steps:

- 1. Open the QCS project in the browser.
- 2. Click on the generated project folder (highlighted with a green box in the image below).

E Output X

3. Right-click on the folder where the files need to be uploaded (Blue box), select the **Upload Files** option (highlighted in red box in the image).

This opens a file browser window on the local PC, allowing for selecting and uploading the required files, as shown in the second image.

e the

QC-BEKITPOC2Z Manual

R	File Edit Diagram Selection View	New File						
← ÷		New Folder_						7.
ſh	EXPLORER	Select for Compare					GETTING STARTED	40
5	> OPEN EDITORS	Find in Folder	Alt+Shift+F	gpt_timer.c				
Q%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%	vontestands vontesta	Copy Paste Copy Path Copy Relative Path Copy Relative Path Upload Files Download End Download End Download End Duplicate Rename	Ctrl+C Alt+Shift+C Ctrl+K Ctrl+Shift+C Delete F2	Ctrl+C Shift+C Shift+C Delete			Getting Started With QUICK- CONNECT STUDIO Please see the sections below to get an overview of the available features and use the blue links to	0
	reaction y application c main_application c main_application h apj_yml cproject generated project secure_azone secure_zml bgk_rafee_baremetal Debug_Flatlau. contention ymm	A Problems 3	E Output X		clangd	S = 6	directly see them in action. Alternatively, please use the toolbar on top, right below the main menu bar. The following feature sectors describe the needed steps and also offer an action to execute those steps automatically, they are indicated by the play icon (\bigotimes).	



10.3.1.3 Replace Any Existing Files

Similar to how new files are uploaded, existing files can require updates.

While uploading an existing file, a pop-up window appears with the message, "Do you want to replace it?". In this case, click on **Replace** to overwrite the old version with the updated one.

n · · · ↑ · · · My F ize · New folder Attachments · ·	C: IND-5CG434189N > Downloads > bgk_ra6e2	_baremetal (1)		v 0	Sauth hale refera h		×
ize • New folder Attachments	C IND-5CG434189N > Downloads > bgk_ra6e2	_baremetal (1)		v 0	Canada hali mital k		
ize • New folder Attachments					Search bgk Jabez L	aremetal (1)	P
Attachments					10	•	0
Desktop Documents Microsoft Copilo Microsoft Teams Pictures Recordings ty PC: IND-5CG4 3D Objects Desktop Documents Documents	Name suproject generated project soure zone soure zmi bgk_nde2_baremetal Debug_Flat_link bgk_rafe2_baremetal Debug_Flat_lunch Configuration.xml Illinktoglog Illinktoglog UCENSE.md project_infojson RADUET RSADUET md	Date modified 11 04 2022 15:43 02-04-2025 15:43 11-04-2025 13:43 11-04-2025 13:49 11-04-2025 13:49 11-04-2025 13:42 08-04-2025 15:42 08-04-2025 15:42 17-03-2025 15:42 17-03-2025 15:43 02-04-2025 15:43 03-04-2025 15:43 03-04-2025 15:43	Type GENERATED File PROJECT File SECURE AZONE File SECURE AZONE File JUNK File LAUNCH File Microsoft Edge HTM., LOG File Markdown Source File TXT File Markdown Source File	Size 1 K8 2 K8 3 K8 7 K3 1 K8 10 K8 40 K8 17,165 K8 2 K8 3 K8 2 K8 3 K8 1 K8			
Ebug Flat.lav	Configuration and	17-53-2023 17:36	clangd	ad from mobile	All files (**) Open Alternatively, please the toolbar on top, below the main mer below the main mer sections describe th needed steps and a fer an action to est those steps automat they are indicated b low icon (6) 1	Cancel use ight tu bar. e e so ecute sically, y the	
	ficrosoft Teams Istures lecordings (PC IND-SCG4 D Objects Documents Downloads × File nam	ficrosoft Teams Scrures ecordings CPC IND-SCG4 DO Objects Do Objects Do currents Desktop Do currents Desktop Des	fitrosoft Teams fitrosoft Teams fitros	fikrosoft Teams Kitures ecordings ecordings PC: IND-SCG4 D Objects becktop Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents Documents D Objects D Obje	fikrosoft Teams Kitures kordings cordings	fikrosoft leans Kitures ecorelings ecorelings poly, rafed, baremetal Debug, Flat,lank 11:04:2025 13:49 bgk, rafed, baremetal Debug, Flat,lank 11:04:2025 13:42 LUNK File 10:05 Configuration.ml 08:04:2025 13:42 PC: IND-SCGL D Objects beckop bocuments boundbads v RADME nd 17:03:2025 17:38 Markdown Source File 2KB project, infolgon 02:04:2025 13:49 DXT File 26:KB Vulplead from mobile File name: configuration.xml VI file (*) Uplead from mobile Problems I Coutput X clangd V I file (*) D Open Alternatively, please the tooblat on top, 1 below the main mer The following feature sections describe th needed steps and al offer an action to e play icon (@).	fikrosoft Teams Kitures konnensa konne vml 00-04-2025 13:49 SECURE_XALFIe 7 K8 bigk_rafe2_baremetal Debug_FlatJanch 11-04-2025 13:49 ALMK File 1 K8 ecordings ecordings ecordings computation.mml 00-04-2025 13:49 Microsoft Edge H1M 01 K8 PC: IND-SCGL D Objects bektop bounents boundbads v RADME and 17-03-2025 17:08 Markdown Scurce File 2 K8 boundbads v RADME and 17-03-2025 17:08 Markdown Scurce File 2 K8 boundbads v RADME and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 17:08 Markdown Scurce File 1 K8 File name: configuration.xml V RadMe and 17-03-2025 10-00 Cancel File name: configuration.xml V RadMe and 17-03-2025 10-00 Cancel File name: configuration.xml V RadMe and 17-03-2025 10-00 Cancel File name: configuration.xml V File 20 File name: configuration.xml V File 100 File name: configuration.xml V File 100 File name: configuration.xml V File 100 File n



10.3.2 Customize Blinky application

This section includes some optional customization users can implement on the Blinky Project.

10.3.2.1 Update the Blinky Application to Code the Desired LED to Blink.

If the user wants to control which LED should be enabled in the code, they can do so by defining macros in the **main_application.h** file:

To enable or disable a specific LED, simply comment or uncomment the corresponding macro.

For example, to disable LED1, comment out the **ENABLE_LED1** macro. To enable it again, uncomment the line.



Code Block 1 main_application.h

After defining the macros, the user should update the **main_application.c** file accordingly. Lines marked with a **+** symbol indicate the additions made to support this configuration.

```
void main_application(void) {
// Start of autogenerated code
  while(true)
  {
    #ifdef ENABLE LED1
+
          utils_set_LED(LED1_LED, BSP_IO_LEVEL_HIGH);
          utils_delay_ms(250);
          log_debug("** LED1 ON ** \r\n");
+
    #endif
+
    #ifdef ENABLE_LED2
+
          utils_set_LED(LED2_LED, BSP_IO_LEVEL_HIGH);
          utils_delay_ms(250);
          log_debug("** LED2 ON ** \r\n");
+
    #endif
+
    #ifdef ENABLE_LED1
+
          utils_set_LED(LED1_LED, BSP_IO_LEVEL_LOW);
          utils_delay_ms(250);
          log_debug("** LED1 OFF ** \r\n");
+
    #endif
+
    #ifdef ENABLE_LED2
+
          utils_set_LED(LED2_LED, BSP_IO_LEVEL_LOW);
          utils_delay_ms(250);
          log_debug("** LED2 OFF ** \r\n");
    #endif
  }
// End of autogenerated code
}
```

Expected Output RTT Logs:

Check the following logs for the expected behavior and also observe LED toggle on Board.

J-Link RTT Viewer V8.10h						
File Terminals Input Logging Help						
All Terminals Terminal 0						
00> DEBUG:/src/main_application.c ** LED1 ON **						
00> 00 PERUS, //set/set setting a ** LED ON **						
AA>						
00> DEBUG:/src/main_application.c ** LED1 OFF **						
00>						
00> DEBUG:/src/main_application.c ** LED2 OFF **						
00> DEBUG:/src/main application.c ** LED1 ON **						
00>						
00> DEBUG:/src/main_application.c ** LED2 ON **						
00> 00> DEUK: /spc/main application c ** LED1 OFE **						
00> Debus/src/mail_application.c of Lebi orr mail						
00> DEBUG:/src/main_application.c ** LED2 OFF **						
de> DEBUG:/src/main_application.c ** LEDI ON **						
00> DEBUG:/src/main application.c ** LED2 ON **						
66>						
00> DEBUG:/src/main_application.c ** LED1 OFF **						

10.3.2.2 Toggle LED using GPIO button

If the user wants to control the on-board LEDs (that is LED1 and LED2) using the GPIO switch S1 (connected to pin **P304** on the **BGK-RA6E2** board), the first step is to enable the **External Interrupt (ICU) stack** in the code by editing the **configuration.xml** file.

In the updated file:

- Lines starting with + indicate new additions.
- Lines starting with **#** indicate modifications to the existing default configuration.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<raConfiguration version="9">
   <generalSettings>
       <option key="#Board#" value="board.bgk_ra6e2"/>
       <option key="CPU" value="RA6E2"/>
       <option key="Core" value="CM33"/>
       <option key="#TargetName#" value="R7FA6E2BB3CFM"/>
       <option key="#TargetARCHITECTURE#" value="cortex-m33"/>
       <option key="#DeviceCommand#" value="R7FA6E2BB"/>
       <option key="#RTOS#" value="_none"/>
       <option key="#pinconfiguration#" value="R7FA6E2BB3CFM.pincfg"/>
       <option key="#FSPVersion#" value="5.5.0"/>
       <option key="#ConfigurationFragments#" value="Renesas##BSP##Board##ra6e2_bgk##"/>
       <option key="#SELECTED_TOOLCHAIN#" value="gcc-arm-embedded"/>
       <option key="#ToolchainVersion#" value="13.2.1.arm-13-7"/>
   </generalSettings>
   <raBspConfiguration>
       <config id="config.bsp.ra6e2.R7FA6E2BB3CFM">
          <property id="config.bsp.part_number" value="config.bsp.part_number.value"/>
          <property id="config.bsp.rom_size_bytes" value="config.bsp.rom_size_bytes.value"/></property id="config.bsp.rom_size_bytes.value"/>
          <property id="config.bsp.rom_size_bytes_hidden" value="262144"/>
          <property id="config.bsp.ram_size_bytes" value="config.bsp.ram_size_bytes.value"/>
          <property id="config.bsp.data_flash_size_bytes"</pre>
value="config.bsp.data_flash_size_bytes.value"/>
          <property id="config.bsp.package_style" value="config.bsp.package_style.value"/></property id="config.bsp.package_style" value="config.bsp.package_style" value="conf
          <property id="config.bsp.package_pins" value="config.bsp.package_pins.value"/>
          <property id="config.bsp.irq_count_hidden" value="96"/>
       </config>
       <config id="config.bsp.ra6e2">
          <property id="config.bsp.series" value="config.bsp.series.value"/>
       </config>
       <config id="config.bsp.ra6e2.fsp">
          <property id="config.bsp.fsp.inline_irq_functions"</pre>
value="config.bsp.common.inline_irg_functions.enabled"/>
          <property id="config.bsp.fsp.tz.exception_response"</pre>
value="config.bsp.fsp.tz.exception_response.nmi"/>
          <property id="config.bsp.fsp.tz.cmsis.bfhfnmins"</pre>
value="config.bsp.fsp.tz.cmsis.bfhfnmins.secure"/>
          <property id="config.bsp.fsp.tz.cmsis.sysresetreqs"</pre>
value="config.bsp.fsp.tz.cmsis.sysresetreqs.secure_only"/>
          <property id="config.bsp.fsp.tz.cmsis.s_priority_boost"</pre>
value="config.bsp.fsp.tz.cmsis.s_priority_boost.disabled"/>
          <property id="config.bsp.fsp.tz.csar" value="config.bsp.fsp.tz.csar.both"/>
          <property id="config.bsp.fsp.tz.rstsar" value="config.bsp.fsp.tz.rstsar.both"/>
          <property id="config.bsp.fsp.tz.bbfsar" value="config.bsp.fsp.tz.bbfsar.both"/>
          <property id="config.bsp.fsp.tz.sramsar.sramprcr"</pre>
value="config.bsp.fsp.tz.sramsar.sramprcr.both"/>
          <property id="config.bsp.fsp.tz.sramsar.sramecc"</pre>
value="config.bsp.fsp.tz.sramsar.sramecc.both"/>
          <property id="config.bsp.fsp.tz.stbramsar"</pre>
value="config.bsp.fsp.tz.stbramsar.both"/>
          <property id="config.bsp.fsp.tz.bussara" value="config.bsp.fsp.tz.bussara.both"/>
          <property id="config.bsp.fsp.tz.bussarb" value="config.bsp.fsp.tz.bussarb.both"/></property id="config.bsp.fsp.tz.bussarb.both"/>
          <property id="config.bsp.fsp.tz.uninitialized_ns_application_fallback"</pre>
value="config.bsp.fsp.tz.uninitialized_ns_application_fallback.enabled"/>
          <property id="config.bsp.fsp.cache_line_size"</pre>
value="config.bsp.fsp.cache_line_size.32"/>
          <property id="config.bsp.fsp.OFS0.iwdt_start_mode"</pre>
value="config.bsp.fsp.OFS0.iwdt_start_mode.disabled"/>
          <property id="config.bsp.fsp.OFS0.iwdt_timeout"</pre>
value="config.bsp.fsp.OFS0.iwdt_timeout.2048"/>
          <property id="config.bsp.fsp.OFS0.iwdt_divisor"</pre>
value="config.bsp.fsp.OFS0.iwdt_divisor.128"/>
          <property id="config.bsp.fsp.OFS0.iwdt_window_end"</pre>
value="config.bsp.fsp.OFS0.iwdt_window_end.0"/>
```

```
<property id="config.bsp.fsp.OFS0.iwdt_window_start"</pre>
value="config.bsp.fsp.OFS0.iwdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.iwdt_reset_interrupt"</pre>
value="config.bsp.fsp.OFS0.iwdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.iwdt_stop_control"</pre>
value="config.bsp.fsp.OFS0.iwdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS0.wdt_start_mode"</pre>
value="config.bsp.fsp.OFS0.wdt_start_mode.register"/>
      <property id="config.bsp.fsp.OFS0.wdt_timeout"</pre>
value="config.bsp.fsp.OFS0.wdt_timeout.16384"/>
      <property id="config.bsp.fsp.OFS0.wdt_divisor"</pre>
value="config.bsp.fsp.OFS0.wdt_divisor.128"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_end"</pre>
value="config.bsp.fsp.OFS0.wdt_window_end.0"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_start"</pre>
value="config.bsp.fsp.OFS0.wdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.wdt_reset_interrupt"</pre>
value="config.bsp.fsp.OFS0.wdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.wdt_stop_control"</pre>
value="config.bsp.fsp.OFS0.wdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0.start"</pre>
value="config.bsp.fsp.OFS1.voltage_detection0.start.disabled"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0_level"</pre>
value="config.bsp.fsp.OFS1.voltage_detection0_level.280"/>
      <property id="config.bsp.fsp.OFS1.hoco_osc"</pre>
value="config.bsp.fsp.OFS1.hoco_osc.disabled"/>
      <property id="config.bsp.fsp.BPS.BPS0" value=""/>
      <property id="config.bsp.fsp.PBPS.PBPS0" value=""/>
      <property id="config.bsp.fsp.hoco_fll" value="config.bsp.fsp.hoco_fll.disabled"/>
      <property id="config.bsp.common.main_osc_wait"</pre>
value="config.bsp.common.main_osc_wait.wait_8163"/>
      <property id="config.bsp.fsp.mcu.adc.max_freq_hz" value="50000000"/>
      <property id="config.bsp.fsp.mcu.sci_uart.max_baud" value="166666666"/>
      roperty id="config.bsp.fsp.mcu.adc.sample_and_hold" value="1"/>
      <property id="config.bsp.fsp.mcu.sci_spi.max_bitrate" value="25000000"/>
      <property id="config.bsp.fsp.mcu.spi.max_bitrate" value="50000000"/>
      <property id="config.bsp.fsp.mcu.iic_master.rate_fastplus" value="1"/></pro>
      <property id="config.bsp.fsp.mcu.canfd.num_channels" value="1"/>
      <property id="config.bsp.fsp.mcu.canfd.rx_fifos" value="2"/>
      <property id="config.bsp.fsp.mcu.canfd.buffer_ram" value="1216"/>
      <property id="config.bsp.fsp.mcu.canfd.afl_rules" value="32"/>
      <property id="config.bsp.fsp.mcu.canfd.afl_rules_each_chnl" value="32"/>
      <property id="config.bsp.fsp.mcu.canfd.max_data_rate_hz" value="5"/>
      <property id="config.bsp.fsp.mcu.sci_uart.cstpen_channels" value="0x0201"/>
      <property id="config.bsp.fsp.mcu.gpt.pin_count_source_channels" value="0xFFFF"/>
      <property id="config.bsp.common.id_mode"</pre>
value="config.bsp.common.id_mode.unlocked"/>
      <property id="config.bsp.common.id_code"</pre>
<property id="config.bsp.common.id1" value=""/>
      <property id="config.bsp.common.id2" value=""/>
      <property id="config.bsp.common.id3" value=""/>
      <property id="config.bsp.common.id4" value=""/>
      <property id="config.bsp.common.id_fixed" value=""/>
      <property id="config.bsp.fsp.mcu.adc_dmac.samples_per_channel" value="32767"/>
    </config>
    <config id="config.bsp.ra">
      <property id="config.bsp.common.main" value="1024"/>
      <property id="config.bsp.common.heap" value="1024"/>
      <property id="config.bsp.common.vcc" value="3300"/>
      <property id="config.bsp.common.checking"</pre>
value="config.bsp.common.checking.disabled"/>
      <property id="config.bsp.common.assert" value="config.bsp.common.assert.none"/>
      <property id="config.bsp.common.error_log"</pre>
value="config.bsp.common.error_log.none"/>
```

```
<property id="config.bsp.common.soft_reset"</pre>
value="config.bsp.common.soft_reset.disabled"/>
      <property id="config.bsp.common.main_osc_populated"</pre>
value="config.bsp.common.main_osc_populated.disabled"/>
      <property id="config.bsp.common.pfs_protect"</pre>
value="config.bsp.common.pfs_protect.enabled"/>
      <property id="config.bsp.common.c_runtime_init"</pre>
value="config.bsp.common.c_runtime_init.enabled"/>
      <property id="config.bsp.common.early_init"</pre>
value="config.bsp.common.early_init.disabled"/>
      <property id="config.bsp.common.main_osc_clock_source"</pre>
value="config.bsp.common.main_osc_clock_source.crystal"/>
      <property id="config.bsp.common.subclock_populated"</pre>
value="config.bsp.common.subclock_populated.enabled"/>
      <property id="config.bsp.common.subclock_drive"</pre>
value="config.bsp.common.subclock_drive.standard"/>
      <property id="config.bsp.common.subclock_stabilization_ms" value="1000"/>
    </config>
  </raBspConfiguration>
  <raClockConfiguration>
    <node id="board.clock.xtal.freq" mul="20000000" option="_edit"/>
    <node id="board.clock.hoco.freq" option="board.clock.hoco.freq.20m"/>
    <node id="board.clock.loco.freq" option="board.clock.loco.freq.32768"/>
    <node id="board.clock.moco.freq" option="board.clock.moco.freq.8m"/>
    <node id="board.clock.subclk.freq" option="board.clock.subclk.freq.32768"/><node id="board.clock.pll.source" option="board.clock.pll.source.hoco"/>
    <node id="board.clock.pll.div" option="board.clock.pll.div.1"/>
    <node id="board.clock.pll.mul" option="board.clock.pll.mul.100"/>
    <node id="board.clock.pll.display" option="board.clock.pll.display.value"/>
    <node id="board.clock.clock.source" option="board.clock.clock.source.pll"/>
    <node id="board.clock.clkout.source" option="board.clock.clkout.source.disabled"/>
    <node id="board.clock.uclk.source" option="board.clock.uclk.source.disabled"/>
    <node id="board.clock.canfdclk.source"
option="board.clock.canfdclk.source.disabled"/>
    <node id="board.clock.cecclk.source" option="board.clock.cecclk.source.disabled"/> <node id="board.clock.i3cclk.source" option="board.clock.i3cclk.source.disabled"/>
    <node id="board.clock.iclk.div" option="board.clock.iclk.div.1"/>
    <node id="board.clock.pclka.div" option="board.clock.pclka.div.2"/>
    <node id="board.clock.pclkb.div" option="board.clock.pclkb.div.4"/>
    <node id="board.clock.pclkc.div" option="board.clock.pclkc.div.4"/>
    <node id="board.clock.pclkd.div" option="board.clock.pclkd.div.2"/>
    <node id="board.clock.fclk.div" option="board.clock.fclk.div.4"/>
    <node id="board.clock.clkout.div" option="board.clock.clkout.div.1"/>
    <node id="board.clock.uclk.div" option="board.clock.uclk.div.5"/>
    <node id="board.clock.canfdclk.div" option="board.clock.canfdclk.div.6"/>
    <node id="board.clock.cecclk.div" option="board.clock.cecclk.div.1"/>
    <node id="board.clock.i3cclk.div" option="board.clock.i3cclk.div.1"/>
    <node id="board.clock.iclk.display" option="board.clock.iclk.display.value"/>
    <node id="board.clock.pclka.display" option="board.clock.pclka.display.value"/>
    <node id="board.clock.pclkb.display" option="board.clock.pclkb.display.value"/>
    <node id="board.clock.pclkc.display" option="board.clock.pclkc.display.value"/>
    <node id="board.clock.pclkd.display" option="board.clock.pclkd.display.value"/>
<node id="board.clock.fclk.display" option="board.clock.fclk.display.value"/>
    <node id="board.clock.clkout.display" option="board.clock.clkout.display.value"/>
    <node id="board.clock.uclk.display" option="board.clock.uclk.display.value"/>
    <node id="board.clock.canfdclk.display"
option="board.clock.canfdclk.display.value"/>
    <node id="board.clock.cecclk.display" option="board.clock.cecclk.display.value"/>
    <node id="board.clock.i3cclk.display" option="board.clock.i3cclk.display.value"/>
  </raClockConfiguration>
  <raComponentSelection>
    <component apiversion="" class="Common" condition="" group="all"
subgroup="fsp_common" variant="" vendor="Renesas" version="5.5.0">
      <description>Board Support Package Common Files</description>
      <originalPack>Renesas.RA.5.5.0.pack</originalPack>
    </component>
```

```
<component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_icu" variant="" vendor="Renesas" version="5.5.0">
       <description>External Interrupt</description>
+
       <originalPack>Renesas.RA.5.5.0.pack</originalPack>
+
     </component>
+
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_ioport" variant="" vendor="Renesas" version="5.5.0">
      <description>I/O Port</description>
      <originalPack>Renesas.RA.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="CMSIS" condition="" group="CMSIS5" subgroup="CoreM"
variant="" vendor="Arm" version="6.1.0+fsp.5.5.0">
      <description>Arm CMSIS Version 6 - Core (M)</description>
      <originalPack>Arm.CMSIS6.6.1.0+fsp.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="Board" subgroup="ra6e2_bgk"
variant="" vendor="Renesas" version="5.5.0">
      <description>RA6E2-BGK Board Support Files</description>
      <originalPack>Renesas.RA_board_ra6e2_bgk.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="device"
variant="R7FA6E2BB3CFM" vendor="Renesas" version="5.5.0">
      <description>Board support package for R7FA6E2BB3CFM</description>
      <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="device"
variant="" vendor="Renesas" version="5.5.0">
      <description>Board support package for RA6E2</description>
      <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="fsp"
variant="" vendor="Renesas" version="5.5.0">
      <description>Board support package for RA6E2 - FSP Data</description>
      <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="events"
variant="" vendor="Renesas" version="5.5.0">
      <description>Board support package for RA6E2 - Events</description>
      <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack>
    </component>
  </raComponentSelection>
  <raElcConfiguration/>
  <raIcuConfiguration/>
  <raModuleConfiguration>
    <module id="module.driver.ioport_on_ioport.0">
      <property id="module.driver.ioport.name" value="g_ioport"/>
      <property id="module.driver.ioport.elc_trigger_ioport1" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioport2" value="_disabled"/></property
      <property id="module.driver.ioport.elc_trigger_ioport3" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioport4" value="_disabled"/>
      <property id="module.driver.ioport.pincfg" value="g_bsp_pin_cfg"/>
    </module>
     <module id="module.driver.external_irq_on_icu.431231040">
+
       <property id="module.driver.external_irq.name" value="g_external_irq"/>
+
       <property id="module.driver.external_irq.channel" value="9"/>
+
       <property id="module.driver.external_irq.trigger"</pre>
value="module.driver.external_irq.trigger.trig_falling"/>
       <property id="module.driver.external_irq.filter_enable"</pre>
value="module.driver.external_irq.filter_enable.false"/>
       property id="module.driver.external_irq.clock_source_div"
value="module.driver.external_irq.clock_source_div.clock_source_div_by_64"/>
      <property id="module.driver.external_irq.p_callback"</pre>
+
value="user_button_callback"/>
       <property id="module.driver.external_irq.ipl"</pre>
value="board.icu.common.irg.priority12"/>
     </module>
```

```
<context id="_hal.0">
       <stack module="module.driver.ioport_on_ioport.0"/>
         <stack module="module.driver.external_irg_on_icu.431231040"/>
     </context>
     <config id="config.driver.ioport">
       <property id="config.driver.ioport.checking"</pre>
value="config.driver.ioport.checking.system"/>
      </config>
+
+
      <config id="config.driver.icu">
+
      <property id="config.driver.icu.param_checking_enable"</pre>
value="config.driver.icu.param_checking_enable.bsp"/>
     </config>
  </raModuleConfiguration>
  <raPinConfiguration>
     <symbolicName propertyId="p000.symbolic_name" value="ARDUINO_A0"/>
     <symbolicName propertyId="p001.symbolic_name" value="ARDUINO_A1"/>
     <symbolicName propertyId="p002.symbolic_name" value="ARDUINO_A2"/>
     <symbolicName propertyId="p003.symbolic_name" value="ARDUINO_A4"/>
     <symbolicName propertyId="p004.symbolic_name" value="ARDUINO_A3"/>
     <symbolicName propertyId="p005.symbolic_name" value="PMOD1_RESET"/><symbolicName propertyId="p006.symbolic_name" value="ARDUINO_D8"/>
     <symbolicName propertyId="p008.symbolic_name" value="ARDUINO_D7"/>
     <symbolicName propertyId="p013.symbolic_name" value="ARDUINO_A5"/>
     <symbolicName propertyId="p014.symbolic_name" value="PMOD2_GPI01"/>
     <symbolicName propertyId="p015.symbolic_name" value="PMOD2_GPI02"/>
<symbolicName propertyId="p100.symbolic_name" value="PMOD2_SCL0_ARDUIN0_SCL"/>
     <symbolicName propertyId="p101.symbolic_name" value="PMOD2_SDA0_ARDUINO_SDA"/>
     <symbolicName propertyId="p104.symbolic_name" value="LED1"/>
     <symbolicName propertyId="p105.symbolic_name" value="ARDUINO_D2"/>
     <symbolicName propertyId="p105.symbolic_name" value="PMOD1_GPI01"/>
<symbolicName propertyId="p107.symbolic_name" value="PMOD1_GPI02"/>
<symbolicName propertyId="p108.symbolic_name" value="DEBUG_SWDI0"/>
     <symbolicName propertyId="p109.symbolic_name" value="PMOD1 TX ARDUINO D11"/>
     <symbolicName propertyId="p110.symbolic_name" value="PMOD1_RX_ARDUINO_D12"/>
     <symbolicName propertyId="pl10.symbolic_name" value="PMOD1_RSPCK_ARDUINO_D13"/>
<symbolicName propertyId="pl12.symbolic_name" value="LED2"/>
<symbolicName propertyId="pl13.symbolic_name" value="ARDUINO_D6"/>
     <symbolicName propertyId="p205.symbolic_name" value="PMOD2_IRQ"/>
     <symbolicName propertyId="p206.symbolic_name" value="PMOD2_SSIDATA"/>
     <symbolicName propertyId="p207.symbolic_name" value="PMOD2_SSICLK"/>
     <symbolicName propertyId="p208.symbolic_name" value="ARDUINO_RESET"/>
<symbolicName propertyId="p300.symbolic_name" value="DEBUG_SWDCLK"/>
     <symbolicName propertyId="p301.symbolic_name" value="ARDUINO_D10"/>
     <symbolicName propertyId="p402.symbolic_name" value="PMOD1_IRQ"/>
     <symbolicName propertyId="p403.symbolic_name" value="ARDUINO_D9"/>
     <symbolicName propertyId="p407.symbolic_name" value="PMOD2_SSIBCK"/>
<symbolicName propertyId="p408.symbolic_name" value="ARDUINO_D3"/>
<symbolicName propertyId="p409.symbolic_name" value="ARDUINO_D5"/>
     <symbolicName propertyId="p410.symbolic_name" value="PMOD2_MISO_ARDUINO_D0"/>
     <symbolicName propertyId="p411.symbolic_name"
value="PMOD2_SENSOR_RESET_ARDUINO_D1"/>
     <symbolicName propertyId="p500.symbolic_name" value="ARDUINO_D4"/>
     <pincfg active="true" name="BGK_RA6E2.pincfg" selected="true"</pre>
symbol="g_bsp_pin_cfg">
       <configSetting altId="adc0.an000.p000" configurationId="adc0.an000"/>
       <configSetting altId="adc0.an001.p001" configurationId="adc0.an001"/>
       <configSetting altId="adc0.an002.p002" configurationId="adc0.an002"/>
<configSetting altId="adc0.an004.p004" configurationId="adc0.an004"/>
<configSetting altId="adc0.an007.p003" configurationId="adc0.an007"/>
       <configSetting altId="adc0.an011.p013" configurationId="adc0.an011"/>
       <configSetting altId="adc0.mode.custom.free" configurationId="adc0.mode"/>
       <configSetting altId="irq.irq9.p304" configurationId="irq.irq9"/>
+
       <configSetting altId="irq.mode.custom.free" configurationId="irq.mode"/>
```

```
<configSetting altId="jtag_fslash_swd.mode.swd.free"
configurationId="jtag_fslash_swd.mode"/>
      <configSetting altId="jtag_fslash_swd.swclk.p300"
configurationId="jtag_fslash_swd.swclk"/>
      <configSetting altId="jtag_fslash_swd.swdio.p108"
configurationId="jtag_fslash_swd.swdio"/>
      <configSetting altId="p000.adc0.an000" configurationId="p000"/>
      <configSetting altId="p000.gpio_mode.gpio_mode_an"
configurationId="p000.gpio_mode"/>
      <configSetting altId="p001.adc0.an001" configurationId="p001"/>
      <configSetting altId="p001.gpio_mode.gpio_mode_an"
configurationId="p001.gpio_mode"/>
      <configSetting altId="p002.adc0.an002" configurationId="p002"/>
      <configSetting altId="p002.gpio_mode.gpio_mode_an"
configurationId="p002.gpio_mode"/>
      <configSetting altId="p003.adc0.an007" configurationId="p003"/>
      <configSetting altId="p003.gpio_mode.gpio_mode_an"
configurationId="p003.gpio_mode"/>
      <configSetting altId="p004.adc0.an004" configurationId="p004"/>
      <configSetting altId="p004.gpio_mode.gpio_mode_an"
configurationId="p004.gpio_mode"/>
      <configSetting altId="p005.output.high" configurationId="p005"/>
      <configSetting altId="p005.gpio_mode.gpio_mode_out.high"
configurationId="p005.gpio_mode"/>
      <configSetting altId="p006.input" configurationId="p006"/>
      <configSetting altId="p006.gpio_mode.gpio_mode_in"
configurationId="p006.gpio_mode"/>
      <configSetting altId="p008.input" configurationId="p008"/>
      <configSetting altId="p008.gpio_mode.gpio_mode_in"
configurationId="p008.gpio_mode"/>
      <configSetting altId="p013.adc0.an011" configurationId="p013"/>
      <configSetting altId="p013.gpio_mode.gpio_mode_an"
configurationId="p013.gpio_mode"/>
      <configSetting altId="p014.input" configurationId="p014"/>
      <configSetting altId="p014.gpio_mode.gpio_mode_in"
configurationId="p014.gpio_mode"/>
      <configSetting altId="p015.input" configurationId="p015"/>
      <configSetting altId="p015.gpio_mode.gpio_mode_in"
configurationId="p015.gpio_mode"/>
      <configSetting altId="p100.sci0.rxd0" configurationId="p100"/>
      <configSetting altId="p100.gpio_mode.gpio_mode_peripheral"
configurationId="p100.gpio_mode"/>
      <configSetting altId="p100.gpio_otype.gpio_otype_n_ch_od"
configurationId="p100.gpio_otype"/>
      <configSetting altId="p101.sci0.txd0" configurationId="p101"/>
      <configSetting altId="p101.gpio_mode.gpio_mode_peripheral"
configurationId="p101.gpio_mode"/>
      <configSetting altId="p101.gpio_otype.gpio_otype_n_ch_od"
configurationId="p101.gpio_otype"/>
      <configSetting altId="p104.output.low" configurationId="p104"/>
      <configSetting altId="p104.gpio_mode.gpio_mode_out.low"
configurationId="p104.gpio_mode"/>
      <configSetting altId="p105.input" configurationId="p105"/>
      <configSetting altId="p105.gpio_mode.gpio_mode_in"
configurationId="p105.gpio_mode"/>
      <configSetting altId="p106.input" configurationId="p106"/>
      <configSetting altId="p106.gpio_mode.gpio_mode_in"
configurationId="p106.gpio_mode"/>
      <configSetting altId="p107.input" configurationId="p107"/>
      <configSetting altId="p107.gpio_mode.gpio_mode_in"
configurationId="p107.gpio_mode"/>
      <configSetting altId="p108.jtag_fslash_swd.swdio" configurationId="p108"/>
      <configSetting altId="p108.gpio_mode.gpio_mode_peripheral"
configurationId="p108.gpio_mode"/>
      <configSetting altId="p109.sci9.txd9" configurationId="p109"/>
```

```
<configSetting altId="p109.gpio_mode.gpio_mode_peripheral"
configurationId="p109.gpio_mode"/>
      <configSetting altId="p110.sci9.rxd9" configurationId="p110"/>
      <configSetting altId="p110.gpio_mode.gpio_mode_peripheral"
configurationId="p110.gpio_mode"/>
      <configSetting altId="pll1.input" configurationId="pll1"/>
      <configSetting altId="p111.gpio_mode.gpio_mode_in"
configurationId="pll1.gpio_mode"/>
      <configSetting altId="p112.output.low" configurationId="p112"/>
      <configSetting altId="p112.gpio_mode.gpio_mode_out.low"
configurationId="p112.gpio_mode"/>
      <configSetting altId="p113.input" configurationId="p113"/>
      <configSetting altId="p113.gpio_mode.gpio_mode_in"
configurationId="p113.gpio_mode"/>
      <configSetting altId="p205.input" configurationId="p205"/>
      <configSetting altId="p205.gpio_mode.gpio_mode_in"
configurationId="p205.gpio_mode"/>
      <configSetting altId="p206.ssie0.ssidata0" configurationId="p206"/>
      <configSetting altId="p206.gpio_mode.gpio_mode_peripheral"
configurationId="p206.gpio_mode"/>
      <configSetting altId="p207.ssie0.ssilrck0" configurationId="p207"/>
      <configSetting altId="p207.gpio_mode.gpio_mode_peripheral"
configurationId="p207.gpio_mode"/>
      <configSetting altId="p208.input" configurationId="p208"/>
      <configSetting altId="p208.gpio_mode.gpio_mode_in"
configurationId="p208.gpio_mode"/>
      <configSetting altId="p300.jtag_fslash_swd.swclk" configurationId="p300"/>
      <configSetting altId="p300.gpio_mode.gpio_mode_peripheral"
configurationId="p300.gpio_mode"/>
      <configSetting altId="p301.input" configurationId="p301"/>
      <configSetting altId="p301.gpio_mode.gpio_mode_in"
configurationId="p301.gpio_mode"/>
       <configSetting altId="p304.irg.irg9" configurationId="p304"/>
       <configSetting altId="p304.gpio_irq.gpio_irq_enabled"
configurationId="p304.gpio_irg"/>
       <configSetting altId="p304.gpio_mode.gpio_mode_irq"
configurationId="p304.gpio_mode"/>
       <configSetting altId="p402.input" configurationId="p402"/>
      <configSetting altId="p402.gpio_mode.gpio_mode_in"
configurationId="p402.gpio_mode"/>
      <configSetting altId="p403.input" configurationId="p403"/>
      <configSetting altId="p403.gpio_mode.gpio_mode_in"
configurationId="p403.gpio_mode"/>
      <configSetting altId="p407.ssie0.ssibck0" configurationId="p407"/>
      <configSetting altId="p407.gpio_mode.gpio_mode_peripheral"
configurationId="p407.gpio_mode"/>
      <configSetting altId="p408.input" configurationId="p408"/>
      <configSetting altId="p408.gpio_mode.gpio_mode_in"
configurationId="p408.gpio mode"/>
      <configSetting altId="p409.input" configurationId="p409"/>
      <configSetting altId="p409.gpio_mode.gpio_mode_in"
configurationId="p409.gpio_mode"/>
      <configSetting altId="p410.input" configurationId="p410"/>
      <configSetting altId="p410.gpio_mode.gpio_mode_in"
configurationId="p410.gpio_mode"/>
      <configSetting altId="p411.output.high" configurationId="p411"/>
      <configSetting altId="p411.gpio_mode.gpio_mode_out.high"
configurationId="p411.gpio_mode"/>
      <configSetting altId="p500.input" configurationId="p500"/>
      <configSetting altId="p500.gpio_mode.gpio_mode_in"
configurationId="p500.gpio_mode"/>
      <configSetting altId="sci0.mode.simplei2c.free" configurationId="sci0.mode"/>
      <configSetting altId="sci0.rxd0.pl00" configurationId="sci0.rxd0"/>
      <configSetting altId="sci0.txd0.p101" configurationId="sci0.txd0"/>
      <configSetting altId="sci9.mode.asynchronousuart.free"
configurationId="sci9.mode"/>
```

```
<configSetting altId="sci9.rxd9.p110" configurationId="sci9.rxd9"/>
      <configSetting altId="sci9.txd9.p109" configurationId="sci9.txd9"/>
      <configSetting altId="ssie0.mode.half_dash_duplex.c"
configurationId="ssie0.mode"/>
      <configSetting altId="ssie0.pairing.c" configurationId="ssie0.pairing"/>
      <configSetting altId="ssie0.ssibck0.p407" configurationId="ssie0.ssibck0"/>
      <configSetting altId="ssie0.ssidata0.p206" configurationId="ssie0.ssidata0"/>
      <configSetting altId="ssie0.ssilrck0.p207" configurationId="ssie0.ssilrck0"/>
    </pincfg>
    <pincfg active="false" name="R7FA6E2BB3CFM.pincfg" selected="false" symbol="">
      <configSetting altId="jtag_fslash_swd.mode.swd.free"
configurationId="jtag_fslash_swd.mode"/>
      <configSetting altId="jtag_fslash_swd.swclk.p300"
configurationId="jtag_fslash_swd.swclk"/>
      <configSetting altId="jtag_fslash_swd.swdio.p108"
configurationId="jtag_fslash_swd.swdio"/>
      <configSetting altId="p108.jtag_fslash_swd.swdio" configurationId="p108"/>
      <configSetting altId="p108.gpio_mode.gpio_mode_peripheral"
configurationId="p108.gpio mode"/>
      <configSetting altId="p300.jtag_fslash_swd.swclk" configurationId="p300"/>
      <configSetting altId="p300.gpio_mode.gpio_mode_peripheral"
configurationId="p300.gpio_mode"/>
    </pincfg>
  </raPinConfiguration>
</raConfiguration>
```

Code Block 2 configuration.xml

Files to be added:

To initialize the external interrupt stack (ICU), middleware API support is required. The following are the files responsible for initializing the external interrupt stack. These functions are called in **main_application.c** to initialize the stack.

```
bgk_ra6e2_baremetal/src /icu_ep.c
bgk_ra6e2_baremetal/src /icu_ep.h
```

The following is the content of the file that should be added to the project.

```
* File Name : icu_ep.c
* Description : Contains function definition.
***********************************
* Copyright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates
* SPDX-License-Identifier: BSD-3-Clause
                      **********************************
#include "common_utils.h"
#include "icu_ep.h"
// Uncomment the desired debug level
#include "log_disabled.h"
//#include "log_error.h"
//#include "log_warning.h"
//#include "log_info.h"
//#include "log_debug.h"
* @addtogroup icu_ep
* @{
***********************************
******************************//**
* @addtogroup icu_ep
* @{
,
********************************//**
* @brief This functions initializes ICU module.
* @param[IN] None
* @retval FSP_SUCCESS Upon successful open of ICU module
* @retval Any Other Error code apart from FSP_SUCCESS Unsuccessful open
                         Upon successful open of ICU module
***********************************
fsp_err_t icu_init(void)
ł
  fsp_err_t err = FSP_SUCCESS;
  /* Open ICU module */
  err = R_ICU_ExternalIrqOpen(&g_external_irq_ctrl, &g_external_irq_cfg);
  /* Handle error */
  if (FSP_SUCCESS != err)
  ł
    /* ICU Open failure message */
    log_error ("\r\n**R_ICU_ExternalIrqOpen API FAILED**\r\n");
  return err;
}
```

QC-BEKITPOC2Z Manual

```
*******************************//**
 * @brief
                         This function enables external interrupt for specified channel.

      * @param[IN]
      None

      * @retval
      FSP_SUCCESS
      Upon successful enable of ICU modulation of the second 
                                                                                              Upon successful enable of ICU module
*****
fsp_err_t icu_enable(void)
ł
        fsp_err_t err = FSP_SUCCESS;
        /* Enable ICU module */
        err = R_ICU_ExternalIrqEnable(&g_external_irq_ctrl);
        /* Handle error */
        if (FSP_SUCCESS != err)
        ł
                 /* ICU Enable failure message */
                log_error ("\r\n**R_ICU_ExternalIrqEnable API FAILED**\r\n");
        }
       return err;
}
/***********************
                                                                  *******************************//**
 * @brief
                          This function closes opened ICU module before the project ends up in an
Error Trap.
  * @param[IN] None
 * @retval
                                 None
void icu_deinit(void)
{
        fsp_err_t err = FSP_SUCCESS;
        /* Close ICU module */
        err = R_ICU_ExternalIrqClose(&g_external_irq_ctrl);
        /* Handle error */
        if (FSP_SUCCESS != err)
        ł
                 /* ICU Close failure message */
                log_error("\r\n**R_ICU_ExternalIrqClose API FAILED**\r\n");
        }
}
```

Code Block 3 bgk_ra6e2_baremetal/src /icu_ep.c

```
* File Name : icu_ep.h
* Description : Contains Macros and function declarations.
* Copyright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates
* SPDX-License-Identifier: BSD-3-Clause
                  #ifndef ICU_EP_H_
#define ICU_EP_H_
#define USER_SW1_IRQ_NUMBER (9)
/* Function declaration */
fsp_err_t icu_init(void);
fsp_err_t icu_enable(void);
void icu_deinit(void);
#endif /* ICU_EP_H_ */
```

Code Block 4 bgk_ra6e2_baremetal/src /icu_ep.h

The user can enable or disable control for each LED based on the macros defined in main_application.h. The process for enabling or disabling these macros has been explained in the previous section.

Files to be updated:

```
* Copyright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates
* SPDX-License-Identifier: BSD-3-Clause
*/
#include <stdio.h>
#include "common_utils.h"
#include "main_application.h"
+ #include "icu_ep.h"
// Uncomment the desired debug level
#include "log_disabled.h"
//#include "log_error.h"
//#include "log_warning.h"
//#include "log_info.h"
//#include "log_debug.h"
+ volatile bool g_sw_press = false;
void main_application(void) {
// Start of autogenerated code
  fsp_err_t err = FSP_SUCCESS;
+
  uint8_t led_state = 1;
+
+ /* Initialize External IRQ driver*/
  log_debug("** ICU ICU Stack ** \r\n");
+
  err = icu_init();
+
  /* Handle error */
+
  if(FSP_SUCCESS != err)
+
+
  {
+
     log_error("** ICU INIT FAILED ** \r\n");
  }
+
+ /* Enable External IRQ driver*/
+ err = icu_enable();
  /* Handle error */
+
  if(FSP_SUCCESS != err)
+
+
   {
    log_error("** ICU ENABLE FAILED ** \r\n");
+
+
    /* Close External IRQ module.*/
+
    icu_deinit();
+
  }
  while(true)
  {
    if(g_sw_press == true)
+
+
+ #ifdef ENABLE_LED1
       if(led_state)
+
+
       {
           utils_set_LED(LED1_LED, BSP_IO_LEVEL_HIGH);
+
           log_debug("** LED1 ON ** \r\n");
+
       }
+
+
       else
       {
+
           utils_set_LED(LED1_LED, BSP_IO_LEVEL_LOW);
+
           log_debug("** LED1 OFF ** \r\n");
+
       }
+
+ #endif
+ #ifdef ENABLE_LED2
       if(led_state)
+
+
       {
+
           utils_set_LED(LED2_LED, BSP_IO_LEVEL_HIGH);
```

```
log_debug("** LED2 ON ** \r\n");
+
       }
+
       else
+
+
       {
           utils_set_LED(LED2_LED, BSP_IO_LEVEL_LOW);
+
           log_debug("** LED2 OFF ** \r\n");
+
       }
+
+ #endif
+
       led_state= !led_state;
+
       g_sw_press = false;
     }
+
 }
// End of autogenerated code
}
// Start of autogenerated code
+ void user_button_callback( external_irq_callback_args_t * p_args)
+ {
+
     if (p_args->channel == USER_SW1_IRQ_NUMBER)
+
     {
       g_sw_press = true;
+
       log_debug("** S1 Switch Pressed ** \r\n");
+
     }
+
+ }
// End of autogenerated code
```

Code Block 5 main_application.c

After all the changes are complete, compile and test the code.

Expected Output RTT Logs:

Check the following logs for the expected behavior and also observe LED toggle on board.

	J Link	-Link RTT Viewer V8.10h	—	\times
	File	Terminals Input Logging Help		
	All	Terminals Terminal 0		
	00>	<pre>DEBUG:/src/main_application.c ** ICU ICU Stack **</pre>		
	00> 00> 00>	DEBUG:/src/main_application.c ** S1 Switch Pressed **		
	00>	<pre>DEBUG:/src/main_application.c ** LED1 ON **</pre>		
	00> 00>	DEBUG:/src/main_application.c ** LED2 ON **		
	00>	<pre>DEBUG:/src/main_application.c ** S1 Switch Pressed **</pre>		
	00> 00>	<pre>DEBUG:/src/main_application.c ** LED1 OFF **</pre>		
t	00> 00>	<pre>DEBUG:/src/main_application.c ** LED2 OFF **</pre>		
	00> 00>	<pre>DEBUG:/src/main_application.c ** S1 Switch Pressed **</pre>		
I	00> 00>	<pre>DEBUG:/src/main_application.c ** LED1 ON **</pre>		
	00>	DEBUG:/src/main_application.c ** LED2 ON **		
	00>	DEBUG:/src/main_application.c ** S1 Switch Pressed **		
	00>	DEBUG:/src/main_application.c ** LED1 OFF **		
	00> 00> 00>	DEBUG:/src/main_application.c ** LED2 OFF **		
l				

10.3.2.3 Control the LED using PWM timer.

If controlling the LED is required, enable the Timer PWM feature on the corresponding LED pins. The first step is to enable the Timer PWM Stack in the code by modifying the **configuration.xml** file.

In the updated file:

- Lines starting with + indicate new additions.
- Lines starting with **#** indicate modifications to the existing default configuration.
- Lines starting with indicate deletion.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<raConfiguration version="9">
   <generalSettings>
       <option key="#Board#" value="board.bgk_ra6e2"/>
       <option key="CPU" value="RA6E2"/>
       <option key="Core" value="CM33"/>
       <option key="#TargetName#" value="R7FA6E2BB3CFM"/>
       <option key="#TargetARCHITECTURE#" value="cortex-m33"/>
       <option key="#DeviceCommand#" value="R7FA6E2BB"/>
       <option key="#RTOS#" value="_none"/>
       <option key="#pinconfiguration#" value="R7FA6E2BB3CFM.pincfg"/>
       <option key="#FSPVersion#" value="5.5.0"/>
       <option key="#ConfigurationFragments#" value="Renesas##BSP##Board##ra6e2_bgk##"/>
       <option key="#SELECTED_TOOLCHAIN#" value="gcc-arm-embedded"/>
       <option key="#ToolchainVersion#" value="13.2.1.arm-13-7"/>
   </generalSettings>
   <raBspConfiguration>
       <config id="config.bsp.ra6e2.R7FA6E2BB3CFM">
          <property id="config.bsp.part_number" value="config.bsp.part_number.value"/>
          <property id="config.bsp.rom_size_bytes" value="config.bsp.rom_size_bytes.value"/></property id="config.bsp.rom_size_bytes.value"/>
          <property id="config.bsp.rom_size_bytes_hidden" value="262144"/>
          <property id="config.bsp.ram_size_bytes" value="config.bsp.ram_size_bytes.value"/></property id="config.bsp.ram_size_bytes" value="config.bsp.ram_size_bytes.value"/>
          <property id="config.bsp.data_flash_size_bytes"</pre>
value="config.bsp.data_flash_size_bytes.value"/>
          <property id="config.bsp.package_style" value="config.bsp.package_style.value"/></property id="config.bsp.package_style" value="config.bsp.package_style" value="conf
          <property id="config.bsp.package_pins" value="config.bsp.package_pins.value"/>
          <property id="config.bsp.irq_count_hidden" value="96"/>
       </config>
       <config id="config.bsp.ra6e2">
          <property id="config.bsp.series" value="config.bsp.series.value"/>
       </config>
       <config id="config.bsp.ra6e2.fsp">
          <property id="config.bsp.fsp.inline_irq_functions"</pre>
value="config.bsp.common.inline_irg_functions.enabled"/>
          <property id="config.bsp.fsp.tz.exception_response"</pre>
value="config.bsp.fsp.tz.exception_response.nmi"/>
          <property id="config.bsp.fsp.tz.cmsis.bfhfnmins"</pre>
value="config.bsp.fsp.tz.cmsis.bfhfnmins.secure"/>
          <property id="config.bsp.fsp.tz.cmsis.sysresetreqs"</pre>
value="config.bsp.fsp.tz.cmsis.sysresetreqs.secure_only"/>
          <property id="config.bsp.fsp.tz.cmsis.s_priority_boost"</pre>
value="config.bsp.fsp.tz.cmsis.s_priority_boost.disabled"/>
          <property id="config.bsp.fsp.tz.csar" value="config.bsp.fsp.tz.csar.both"/>
          <property id="config.bsp.fsp.tz.rstsar" value="config.bsp.fsp.tz.rstsar.both"/>
          <property id="config.bsp.fsp.tz.bbfsar" value="config.bsp.fsp.tz.bbfsar.both"/>
          <property id="config.bsp.fsp.tz.sramsar.sramprcr"</pre>
value="config.bsp.fsp.tz.sramsar.sramprcr.both"/>
          <property id="config.bsp.fsp.tz.sramsar.sramecc"</pre>
value="config.bsp.fsp.tz.sramsar.sramecc.both"/>
          <property id="config.bsp.fsp.tz.stbramsar"</pre>
value="config.bsp.fsp.tz.stbramsar.both"/>
          <property id="config.bsp.fsp.tz.bussara" value="config.bsp.fsp.tz.bussara.both"/>
          <property id="config.bsp.fsp.tz.bussarb" value="config.bsp.fsp.tz.bussarb.both"/></property id="config.bsp.fsp.tz.bussarb.both"/>
          <property id="config.bsp.fsp.tz.uninitialized_ns_application_fallback"</pre>
value="config.bsp.fsp.tz.uninitialized_ns_application_fallback.enabled"/>
          <property id="config.bsp.fsp.cache_line_size"</pre>
value="config.bsp.fsp.cache_line_size.32"/>
          <property id="config.bsp.fsp.OFS0.iwdt_start_mode"</pre>
value="config.bsp.fsp.OFS0.iwdt_start_mode.disabled"/>
          <property id="config.bsp.fsp.OFS0.iwdt_timeout"</pre>
value="config.bsp.fsp.OFS0.iwdt_timeout.2048"/>
          <property id="config.bsp.fsp.OFS0.iwdt_divisor"</pre>
value="config.bsp.fsp.OFS0.iwdt_divisor.128"/>
          <property id="config.bsp.fsp.OFS0.iwdt_window_end"</pre>
value="config.bsp.fsp.OFS0.iwdt_window_end.0"/>
```

```
<property id="config.bsp.fsp.OFS0.iwdt_window_start"</pre>
value="config.bsp.fsp.OFS0.iwdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.iwdt_reset_interrupt"</pre>
value="config.bsp.fsp.OFS0.iwdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.iwdt_stop_control"</pre>
value="config.bsp.fsp.OFS0.iwdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS0.wdt_start_mode"</pre>
value="config.bsp.fsp.OFS0.wdt_start_mode.register"/>
      <property id="config.bsp.fsp.OFS0.wdt_timeout"</pre>
value="config.bsp.fsp.OFS0.wdt_timeout.16384"/>
      <property id="config.bsp.fsp.OFS0.wdt_divisor"</pre>
value="config.bsp.fsp.OFS0.wdt_divisor.128"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_end"</pre>
value="config.bsp.fsp.OFS0.wdt_window_end.0"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_start"</pre>
value="config.bsp.fsp.OFS0.wdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.wdt_reset_interrupt"</pre>
value="config.bsp.fsp.OFS0.wdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.wdt_stop_control"</pre>
value="config.bsp.fsp.OFS0.wdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0.start"</pre>
value="config.bsp.fsp.OFS1.voltage_detection0.start.disabled"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0_level"</pre>
value="config.bsp.fsp.OFS1.voltage_detection0_level.280"/>
      <property id="config.bsp.fsp.OFS1.hoco_osc"</pre>
value="config.bsp.fsp.OFS1.hoco_osc.disabled"/>
      <property id="config.bsp.fsp.BPS.BPS0" value=""/>
      <property id="config.bsp.fsp.PBPS.PBPS0" value=""/>
      <property id="config.bsp.fsp.hoco_fll" value="config.bsp.fsp.hoco_fll.disabled"/>
      <property id="config.bsp.common.main_osc_wait"</pre>
value="config.bsp.common.main_osc_wait.wait_8163"/>
      <property id="config.bsp.fsp.mcu.adc.max_freq_hz" value="50000000"/>
      <property id="config.bsp.fsp.mcu.sci_uart.max_baud" value="166666666"/>
      roperty id="config.bsp.fsp.mcu.adc.sample_and_hold" value="1"/>
      <property id="config.bsp.fsp.mcu.sci_spi.max_bitrate" value="25000000"/>
      <property id="config.bsp.fsp.mcu.spi.max_bitrate" value="50000000"/>
      <property id="config.bsp.fsp.mcu.iic_master.rate_fastplus" value="1"/></pro>
      <property id="config.bsp.fsp.mcu.canfd.num_channels" value="1"/>
      <property id="config.bsp.fsp.mcu.canfd.rx_fifos" value="2"/>
      <property id="config.bsp.fsp.mcu.canfd.buffer_ram" value="1216"/>
      <property id="config.bsp.fsp.mcu.canfd.afl_rules" value="32"/>
      <property id="config.bsp.fsp.mcu.canfd.afl_rules_each_chnl" value="32"/>
      <property id="config.bsp.fsp.mcu.canfd.max_data_rate_hz" value="5"/>
      <property id="config.bsp.fsp.mcu.sci_uart.cstpen_channels" value="0x0201"/>
      <property id="config.bsp.fsp.mcu.gpt.pin_count_source_channels" value="0xFFFF"/>
      <property id="config.bsp.common.id_mode"</pre>
value="config.bsp.common.id_mode.unlocked"/>
      <property id="config.bsp.common.id_code"</pre>
<property id="config.bsp.common.id1" value=""/>
      <property id="config.bsp.common.id2" value=""/>
      <property id="config.bsp.common.id3" value=""/>
      <property id="config.bsp.common.id4" value=""/>
      <property id="config.bsp.common.id_fixed" value=""/>
      <property id="config.bsp.fsp.mcu.adc_dmac.samples_per_channel" value="32767"/>
    </config>
    <config id="config.bsp.ra">
      <property id="config.bsp.common.main" value="1024"/>
      <property id="config.bsp.common.heap" value="1024"/>
      <property id="config.bsp.common.vcc" value="3300"/>
      <property id="config.bsp.common.checking"</pre>
value="config.bsp.common.checking.disabled"/>
      <property id="config.bsp.common.assert" value="config.bsp.common.assert.none"/>
      <property id="config.bsp.common.error_log"</pre>
value="config.bsp.common.error_log.none"/>
```

```
<property id="config.bsp.common.soft_reset"</pre>
value="config.bsp.common.soft_reset.disabled"/>
      <property id="config.bsp.common.main_osc_populated"</pre>
value="config.bsp.common.main_osc_populated.disabled"/>
      <property id="config.bsp.common.pfs_protect"</pre>
value="config.bsp.common.pfs_protect.enabled"/>
      <property id="config.bsp.common.c_runtime_init"</pre>
value="config.bsp.common.c_runtime_init.enabled"/>
      <property id="config.bsp.common.early_init"</pre>
value="config.bsp.common.early_init.disabled"/>
      <property id="config.bsp.common.main_osc_clock_source"</pre>
value="config.bsp.common.main_osc_clock_source.crystal"/>
      <property id="config.bsp.common.subclock_populated"</pre>
value="config.bsp.common.subclock_populated.enabled"/>
      <property id="config.bsp.common.subclock_drive"</pre>
value="config.bsp.common.subclock_drive.standard"/>
      <property id="config.bsp.common.subclock_stabilization_ms" value="1000"/>
    </config>
  </raBspConfiguration>
  <raClockConfiguration>
    <node id="board.clock.xtal.freq" mul="20000000" option="_edit"/>
    <node id="board.clock.hoco.freq" option="board.clock.hoco.freq.20m"/>
    <node id="board.clock.loco.freq" option="board.clock.loco.freq.32768"/>
    <node id="board.clock.moco.freq" option="board.clock.moco.freq.8m"/>
    <node id="board.clock.subclk.freq" option="board.clock.subclk.freq.32768"/><node id="board.clock.pll.source" option="board.clock.pll.source.hoco"/>
    <node id="board.clock.pll.div" option="board.clock.pll.div.1"/>
    <node id="board.clock.pll.mul" option="board.clock.pll.mul.100"/>
    <node id="board.clock.pll.display" option="board.clock.pll.display.value"/>
    <node id="board.clock.clock.source" option="board.clock.clock.source.pll"/>
    <node id="board.clock.clkout.source" option="board.clock.clkout.source.disabled"/>
    <node id="board.clock.uclk.source" option="board.clock.uclk.source.disabled"/>
    <node id="board.clock.canfdclk.source"
option="board.clock.canfdclk.source.disabled"/>
    <node id="board.clock.cecclk.source" option="board.clock.cecclk.source.disabled"/> <node id="board.clock.i3cclk.source" option="board.clock.i3cclk.source.disabled"/>
    <node id="board.clock.iclk.div" option="board.clock.iclk.div.1"/>
    <node id="board.clock.pclka.div" option="board.clock.pclka.div.2"/>
    <node id="board.clock.pclkb.div" option="board.clock.pclkb.div.4"/>
    <node id="board.clock.pclkc.div" option="board.clock.pclkc.div.4"/>
    <node id="board.clock.pclkd.div" option="board.clock.pclkd.div.2"/>
    <node id="board.clock.fclk.div" option="board.clock.fclk.div.4"/>
    <node id="board.clock.clkout.div" option="board.clock.clkout.div.1"/>
    <node id="board.clock.uclk.div" option="board.clock.uclk.div.5"/>
    <node id="board.clock.canfdclk.div" option="board.clock.canfdclk.div.6"/>
    <node id="board.clock.cecclk.div" option="board.clock.cecclk.div.1"/>
    <node id="board.clock.i3cclk.div" option="board.clock.i3cclk.div.1"/>
    <node id="board.clock.iclk.display" option="board.clock.iclk.display.value"/>
    <node id="board.clock.pclka.display" option="board.clock.pclka.display.value"/>
    <node id="board.clock.pclkb.display" option="board.clock.pclkb.display.value"/>
    <node id="board.clock.pclkc.display" option="board.clock.pclkc.display.value"/>
    <node id="board.clock.pclkd.display" option="board.clock.pclkd.display.value"/>
<node id="board.clock.fclk.display" option="board.clock.fclk.display.value"/>
    <node id="board.clock.clkout.display" option="board.clock.clkout.display.value"/>
    <node id="board.clock.uclk.display" option="board.clock.uclk.display.value"/>
    <node id="board.clock.canfdclk.display"
option="board.clock.canfdclk.display.value"/>
    <node id="board.clock.cecclk.display" option="board.clock.cecclk.display.value"/>
    <node id="board.clock.i3cclk.display" option="board.clock.i3cclk.display.value"/>
  </raClockConfiguration>
  <raComponentSelection>
    <component apiversion="" class="Common" condition="" group="all"
subgroup="fsp_common" variant="" vendor="Renesas" version="5.5.0">
      <description>Board Support Package Common Files</description>
      <originalPack>Renesas.RA.5.5.0.pack</originalPack>
    </component>
```

<component apiversion="" class="HAL Drivers" condition="" group="all" subgroup="r_gpt" variant="" vendor="Renesas" version="5.5.0"> <description>General PWM Timer</description> + <originalPack>Renesas.RA.5.5.0.pack</originalPack> + </component> + <component apiversion="" class="HAL Drivers" condition="" group="all" subgroup="r_ioport" variant="" vendor="Renesas" version="5.5.0"> <description>I/O Port</description> <originalPack>Renesas.RA.5.5.0.pack</originalPack> </component> <component apiversion="" class="CMSIS" condition="" group="CMSIS5" subgroup="CoreM" variant="" vendor="Arm" version="6.1.0+fsp.5.5.0"> <description>Arm CMSIS Version 6 - Core (M)</description> <originalPack>Arm.CMSIS6.6.1.0+fsp.5.5.0.pack</originalPack> </component> <component apiversion="" class="BSP" condition="" group="Board" subgroup="ra6e2_bgk" variant="" vendor="Renesas" version="5.5.0"> <description>RA6E2-BGK Board Support Files</description> <originalPack>Renesas.RA_board_ra6e2_bgk.5.5.0.pack</originalPack> </component> <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="device" variant="R7FA6E2BB3CFM" vendor="Renesas" version="5.5.0"> <description>Board support package for R7FA6E2BB3CFM</description> <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack> </component> <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="device" variant="" vendor="Renesas" version="5.5.0"> <description>Board support package for RA6E2</description> <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack> </component> <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="fsp" variant="" vendor="Renesas" version="5.5.0"> <description>Board support package for RA6E2 - FSP Data</description> <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack> </component> <component apiversion="" class="BSP" condition="" group="ra6e2" subgroup="events" variant="" vendor="Renesas" version="5.5.0"> <description>Board support package for RA6E2 - Events</description> <originalPack>Renesas.RA_mcu_ra6e2.5.5.0.pack</originalPack> </component> </raComponentSelection> <raElcConfiguration/> <raIcuConfiguration/> <raModuleConfiguration> <module id="module.driver.ioport_on_ioport.0"> <property id="module.driver.ioport.name" value="g_ioport"/> <property id="module.driver.ioport.elc_trigger_ioport1" value="_disabled"/> <property id="module.driver.ioport.elc_trigger_ioport2" value="_disabled"/></property <property id="module.driver.ioport.elc_trigger_ioport3" value="_disabled"/> <property id="module.driver.ioport.elc_trigger_ioport4" value="_disabled"/> <property id="module.driver.ioport.pincfg" value="g_bsp_pin_cfg"/> </module> <module id="module.driver.timer_on_gpt.1083164079"> <property id="module.driver.timer.name" value="g_timer_pwm_led1"/> <property id="module.driver.timer.channel" value="1"/> <property id="module.driver.timer.mode"</pre> value="module.driver.timer.mode.mode_pwm"/> <property id="module.driver.timer.period" value="1"/> <property id="module.driver.timer.compare_match.a.status"</pre> value="module.driver.timer.compare_match.a.status.disabled"/> <property id="module.driver.timer.compare_match.a.value" value="0"/> <property id="module.driver.timer.compare_match.b.status"</pre> value="module.driver.timer.compare_match.b.status.disabled"/> <property id="module.driver.timer.compare_match.b.value" value="0"/> <property id="module.driver.timer.unit"</pre> value="module.driver.timer.unit.unit_period_sec"/>

```
<property id="module.driver.timer.gtior.gtioa.initial_output_level"</pre>
value="module.driver.timer.gtior.gtioa.initial_output_level.low"/>
      <property id="module.driver.timer.gtior.gtioa.cycle_end_output_level"</pre>
value="module.driver.timer.gtior.gtioa.cycle_end_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtioa.compare_match_output_level"</pre>
value="module.driver.timer.gtior.gtioa.compare_match_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtioa.count_stop_retain"</pre>
value="module.driver.timer.gtior.gtioa.count_stop_retain.disabled"/>
      <property id="module.driver.timer.gtior.gtiob.initial_output_level"</pre>
value="module.driver.timer.gtior.gtiob.initial_output_level.low"/>
      <property id="module.driver.timer.gtior.gtiob.cycle_end_output_level"</pre>
value="module.driver.timer.gtior.gtiob.cycle_end_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtiob.compare_match_output_level"</pre>
value="module.driver.timer.gtior.gtiob.compare_match_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtiob.count_stop_retain"</pre>
value="module.driver.timer.gtior.gtiob.count_stop_retain.disabled"/>
      <property id="module.driver.timer.gtior.custom_waveform_enable"</pre>
value="module.driver.timer.gtior.custom_waveform_enable.disabled"/>
      <property id="module.driver.timer.duty_cycle" value="50"/>
      <property id="module.driver.timer.gtioca_output_enabled"</pre>
value="module.driver.timer.gtioca_output_enabled.false"/>
      <property id="module.driver.timer.gtioca_stop_level"</pre>
value="module.driver.timer.gtioca_stop_level.pin_level_low"/>
      <property id="module.driver.timer.gtiocb_output_enabled"</pre>
value="module.driver.timer.gtiocb_output_enabled.true"/>
      <property id="module.driver.timer.gtiocb_stop_level"</pre>
value="module.driver.timer.gtiocb_stop_level.pin_level_low"/>
      <property id="module.driver.timer.count_up_source" value=""/>
      <property id="module.driver.timer.count_down_source" value=""/>
      <property id="module.driver.timer.start_source" value=""/>
      <property id="module.driver.timer.stop_source" value=""/>
      <property id="module.driver.timer.clear_source" value=""/>
      <property id="module.driver.timer.capture_a_source" value=""/>
      <property id="module.driver.timer.capture_b_source" value=""/>
      <property id="module.driver.timer.gtioca_filter"</pre>
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
      <property id="module.driver.timer.gtiocb_filter"</pre>
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
      <property id="module.driver.timer.p_callback" value="NULL"/>
      <property id="module.driver.timer.ipl" value="_disabled"/>
      <property id="module.driver.timer.capture_a_ipl" value="_disabled"/><property id="module.driver.timer.capture_b_ipl" value="_disabled"/>
      <property id="module.driver.timer.trough_ipl" value="_disabled"/>
      <property id="module.driver.timer.extra"</pre>
value="module.driver.timer.extra.disabled"/>
      <property id="module.driver.timer.poeg_link"</pre>
value="module.driver.timer.poeg_link.poeg_link_poeg0"/>
      <property id="module.driver.timer.output_disable" value=""/>
      <property id="module.driver.timer.adc_trigger" value=""/>
      <property id="module.driver.timer.adc_a_compare_match" value="0"/>
      <property id="module.driver.timer.adc_b_compare_match" value="0"/>
      <property id="module.driver.timer.dead_time_count_up" value="0"/>
      <property id="module.driver.timer.dead_time_count_down" value="0"/>
      <property id="module.driver.timer.interrupt_skip.source"</pre>
value="module.driver.timer.interrupt_skip.source.none"/>
      property id="module.driver.timer.interrupt_skip.count"
value="module.driver.timer.interrupt_skip.count.count_0"/>
      <property id="module.driver.timer.interrupt_skip.adc"</pre>
value="module.driver.timer.interrupt_skip.skip_sources.interrupt_skip.adc.none"/>
      <property id="module.driver.timer.gtioca_disable_setting"</pre>
value="module.driver.timer.gtioca_disable_setting.gtioc_disable_prohibited"/>
      <property id="module.driver.timer.gtiocb_disable_setting"</pre>
value="module.driver.timer.gtiocb_disable_setting.gtioc_disable_prohibited"/>
    </module>
     <module id="module.driver.timer_on_gpt.230584519">
+
       <property id="module.driver.timer.name" value="g_timer_pwm_led2"/>
```

```
<property id="module.driver.timer.channel" value="3"/>
       <property id="module.driver.timer.mode"</pre>
+
value="module.driver.timer.mode.mode_pwm"/>
       <property id="module.driver.timer.period" value="1"/>
       <property id="module.driver.timer.compare_match.a.status"</pre>
value="module.driver.timer.compare_match.a.status.disabled"/>
       <property id="module.driver.timer.compare_match.a.value" value="0"/>
+
       <property id="module.driver.timer.compare_match.b.status"</pre>
value="module.driver.timer.compare_match.b.status.disabled"/>
       <property id="module.driver.timer.compare_match.b.value" value="0"/>
       <property id="module.driver.timer.unit"</pre>
value="module.driver.timer.unit.unit_period_sec"/>
       <property id="module.driver.timer.gtior.gtioa.initial_output_level"</pre>
+
value="module.driver.timer.gtior.gtioa.initial_output_level.low"/>
       <property id="module.driver.timer.gtior.gtioa.cycle_end_output_level"</pre>
value="module.driver.timer.gtior.gtioa.cycle_end_output_level.retain"/>
       <property id="module.driver.timer.gtior.gtioa.compare_match_output_level"</pre>
value="module.driver.timer.gtior.gtioa.compare_match_output_level.retain"/>
       <property id="module.driver.timer.gtior.gtioa.count_stop_retain"</pre>
value="module.driver.timer.gtior.gtioa.count_stop_retain.disabled"/>
       <property id="module.driver.timer.gtior.gtiob.initial_output_level"</pre>
value="module.driver.timer.gtior.gtiob.initial_output_level.low"/>
       <property id="module.driver.timer.gtior.gtiob.cycle_end_output_level"</pre>
value="module.driver.timer.gtior.gtiob.cycle_end_output_level.retain"/>
       <property id="module.driver.timer.gtior.gtiob.compare_match_output_level"</pre>
+
value="module.driver.timer.gtior.gtiob.compare_match_output_level.retain"/>
       <property id="module.driver.timer.gtior.gtiob.count_stop_retain"</pre>
value="module.driver.timer.gtior.gtiob.count_stop_retain.disabled"/>
       <property id="module.driver.timer.gtior.custom_waveform_enable"</pre>
+
value="module.driver.timer.gtior.custom_waveform_enable.disabled"/>
       <property id="module.driver.timer.duty_cycle" value="50"/>
       <property id="module.driver.timer.gtioca_output_enabled"</pre>
value="module.driver.timer.gtioca_output_enabled.false"/>
       <property id="module.driver.timer.gtioca_stop_level"</pre>
+
value="module.driver.timer.gtioca_stop_level.pin_level_low"/>
       <property id="module.driver.timer.gtiocb_output_enabled"</pre>
+
value="module.driver.timer.gtiocb_output_enabled.true"/>
       <property id="module.driver.timer.gtiocb_stop_level"</pre>
value="module.driver.timer.gtiocb_stop_level.pin_level_low"/>
       <property id="module.driver.timer.count_up_source" value=""/></pro>
+
       <property id="module.driver.timer.count_down_source" value=""/>
+
+
       <property id="module.driver.timer.start_source" value=""/></property
       <property id="module.driver.timer.stop_source" value=""/>
+
       <property id="module.driver.timer.clear_source" value=""/>
+
       <property id="module.driver.timer.capture_a_source" value=""/></pro>
+
       <property id="module.driver.timer.capture_b_source" value=""/>
+
       <property id="module.driver.timer.gtioca_filter"</pre>
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
       <property id="module.driver.timer.gtiocb_filter"</pre>
+
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
       <property id="module.driver.timer.p_callback" value="NULL"/></property id="module.driver.timer.p_callback" value="NULL"/>
+
+
       <property id="module.driver.timer.ipl" value="_disabled"/>
       <property id="module.driver.timer.capture_a_ipl" value="_disabled"/><property id="module.driver.timer.capture_b_ipl" value="_disabled"/>
+
+
       <property id="module.driver.timer.trough_ipl" value="_disabled"/>
+
       <property id="module.driver.timer.extra"</pre>
+
value="module.driver.timer.extra.disabled"/>
       <property id="module.driver.timer.poeg_link"</pre>
+
value="module.driver.timer.poeg_link.poeg_link_poeg0"/>
       property id="module.driver.timer.output_disable" value=""/>
+
       <property id="module.driver.timer.adc_trigger" value=""/>
+
       <property id="module.driver.timer.adc_a_compare_match" value="0"/>
+
+
       <property id="module.driver.timer.adc_b_compare_match" value="0"/>
       property id="module.driver.timer.dead_time_count_up" value="0"/>
+
       <property id="module.driver.timer.dead_time_count_down" value="0"/>
```

```
property id="module.driver.timer.interrupt_skip.source"
value="module.driver.timer.interrupt_skip.source.none"/>
        <property id="module.driver.timer.interrupt_skip.count"</pre>
+
value="module.driver.timer.interrupt_skip.count.count_0"/>
        <property id="module.driver.timer.interrupt_skip.adc"</pre>
+
value="module.driver.timer.interrupt_skip.skip_sources.interrupt_skip.adc.none"/>
       <property id="module.driver.timer.gtioca_disable_setting"</pre>
+
value="module.driver.timer.gtioca_disable_setting.gtioc_disable_prohibited"/>
        <property id="module.driver.timer.gtiocb_disable_setting"</pre>
value="module.driver.timer.gtiocb_disable_setting.gtioc_disable_prohibited"/>
     </module>
     <context id="_hal.0">
       <stack module="module.driver.ioport_on_ioport.0"/>
        <stack module="module.driver.timer_on_gpt.1083164079"/>
+
        <stack module="module.driver.timer_on_gpt.230584519"/>
+
     </context>
      <config id="config.driver.gpt">
+
        <property id="config.driver.gpt.param_checking_enable"</pre>
value="config.driver.gpt.param_checking_enable.bsp"/>
        <property id="config.driver.gpt.output_support_enable"</pre>
value="config.driver.gpt.output_support_enable.enabled"/>
       <property id="config.driver.gpt.write_protect_enable"</pre>
value="config.driver.gpt.write_protect_enable.disabled"/>
     </config>
     <config id="config.driver.ioport">
       <property id="config.driver.ioport.checking"</pre>
value="config.driver.ioport.checking.system"/>
     </config>
  </raModuleConfiguration>
  <raPinConfiguration>
     <symbolicName propertyId="p000.symbolic_name" value="ARDUINO_A0"/>
     <symbolicName propertyId="p001.symbolic_name" value="ARDUINO_A1"/>
     <symbolicName propertyId="p002.symbolic name" value="ARDUINO_A2"/>
     <symbolicName propertyId="p003.symbolic_name" value="ARDUINO_A4"/>
    <symbolicName propertyId="p005.symbolic_name" value="ARDUINO_A3"/>
<symbolicName propertyId="p005.symbolic_name" value="PMOD1_RESET"/>
<symbolicName propertyId="p006.symbolic_name" value="ARDUINO_D8"/>
     <symbolicName propertyId="p008.symbolic_name" value="ARDUINO_D7"/>
     <symbolicName propertyId="p013.symbolic_name" value="ARDUINO_A5"/>
     <symbolicName propertyId="p014.symbolic_name" value="PMOD2_GPI01"/>
     <symbolicName propertyId="p015.symbolic_name" value="PMOD2_GPI02"/>
<symbolicName propertyId="p100.symbolic_name" value="PMOD2_SCL0_ARDUIN0_SCL"/>
     <symbolicName propertyId="p101.symbolic_name" value="PMOD2_SDA0_ARDUINO_SDA"/>
     <symbolicName propertyId="p104.symbolic_name" value="LED1"/>
     <symbolicName propertyId="p105.symbolic_name" value="ARDUINO_D2"/>
    <symbolicName propertyId="p100.symbolic_name" value="PMOD1_GPI01"/>
<symbolicName propertyId="p107.symbolic_name" value="PMOD1_GPI02"/>
<symbolicName propertyId="p108.symbolic_name" value="DEBUG_SWDI0"/>
     <symbolicName propertyId="p109.symbolic name" value="PMOD1 TX ARDUINO D11"/>
     <symbolicName propertyId="p110.symbolic_name" value="PMOD1_RX_ARDUINO_D12"/>
     <symbolicName propertyId="p111.symbolic_name" value="PMOD1_RSPCK_ARDUINO_D13"/>
     <symbolicName propertyId="p112.symbolic_name" value="LED2"/>
<symbolicName propertyId="p113.symbolic_name" value="ARDUINO_D6"/>
     <symbolicName propertyId="p205.symbolic_name" value="PMOD2_IRQ"/>
     <symbolicName propertyId="p206.symbolic_name" value="PMOD2_SSIDATA"/>
     <symbolicName propertyId="p207.symbolic_name" value="PMOD2_SSICLK"/>
    <symbolicName propertyId="p208.symbolic_name" value="ARDUINO_RESET"/>
<symbolicName propertyId="p300.symbolic_name" value="DEBUG_SWDCLK"/>
<symbolicName propertyId="p301.symbolic_name" value="ARDUINO_D10"/>
     <symbolicName propertyId="p402.symbolic_name" value="PMOD1_IRO"/>
     <symbolicName propertyId="p403.symbolic_name" value="ARDUINO_D9"/>
     <symbolicName propertyId="p407.symbolic_name" value="PMOD2_SSIBCK"/>
     <symbolicName propertyId="p408.symbolic_name" value="ARDUINO_D3"/>
<symbolicName propertyId="p409.symbolic_name" value="ARDUINO_D5"/>
     <symbolicName propertyId="p410.symbolic_name" value="PMOD2_MISO_ARDUINO_D0"/>
```

```
<symbolicName propertyId="p411.symbolic_name"
value="PMOD2_SENSOR_RESET_ARDUINO_D1"/>
    <symbolicName propertyId="p500.symbolic_name" value="ARDUINO_D4"/>
    <pincfg active="true" name="BGK_RA6E2.pincfg" selected="true"</pre>
symbol="g_bsp_pin_cfg">
      <configSetting altId="adc0.an000.p000" configurationId="adc0.an000"/>
      <configSetting altId="adc0.an001.p001" configurationId="adc0.an001"/>
      <configSetting altId="adc0.an002.p002" configurationId="adc0.an002"/>
<configSetting altId="adc0.an004.p004" configurationId="adc0.an004"/>
<configSetting altId="adc0.an007.p003" configurationId="adc0.an007"/>
      <configSetting altId="adc0.an011.p013" configurationId="adc0.an011"/>
      <configSetting altId="adc0.mode.custom.free" configurationId="adc0.mode"/>
       <configSetting altId="gpt1.gtioclb.p104" configurationId="gpt1.gtioclb"/>
+
       <configSetting altId="gpt1.mode.gtiocaorgtiocb.free"
configurationId="gpt1.mode"/>
       <configSetting altId="gpt3.gtioc3b.p112" configurationId="gpt3.gtioc3b"/>
       <configSetting altId="gpt3.mode.gtiocaorgtiocb.free"
configurationId="gpt3.mode"/>
      <configSetting altId="jtag_fslash_swd.mode.swd.free"
configurationId="jtag_fslash_swd.mode"/>
      <configSetting altId="jtag_fslash_swd.swclk.p300"
configurationId="jtag_fslash_swd.swclk"/>
      <configSetting altId="jtag_fslash_swd.swdio.p108"
configurationId="jtag_fslash_swd.swdio"/>
      <configSetting altId="p000.adc0.an000" configurationId="p000"/>
      <configSetting altId="p000.gpio_mode.gpio_mode_an"
configurationId="p000.gpio_mode"/>
      <configSetting altId="p001.adc0.an001" configurationId="p001"/>
      <configSetting altId="p001.gpio_mode.gpio_mode_an"
configurationId="p001.gpio_mode"/>
      <configSetting altId="p002.adc0.an002" configurationId="p002"/>
      <configSetting altId="p002.gpio_mode.gpio_mode_an"
configurationId="p002.gpio_mode"/>
      <configSetting altId="p003.adc0.an007" configurationId="p003"/>
      <configSetting altId="p003.gpio_mode.gpio_mode_an"
configurationId="p003.gpio_mode"/>
      <configSetting altId="p004.adc0.an004" configurationId="p004"/>
      <configSetting altId="p004.gpio_mode.gpio_mode_an"
configurationId="p004.gpio_mode"/>
      <configSetting altId="p005.output.high" configurationId="p005"/>
      <configSetting altId="p005.gpio_mode.gpio_mode_out.high"
configurationId="p005.gpio_mode"/>
      <configSetting altId="p006.input" configurationId="p006"/>
      <configSetting altId="p006.gpio_mode.gpio_mode_in"
configurationId="p006.gpio_mode"/>
      <configSetting altId="p008.input" configurationId="p008"/>
      <configSetting altId="p008.gpio_mode.gpio_mode_in"
configurationId="p008.gpio_mode"/>
      <configSetting altId="p013.adc0.an011" configurationId="p013"/>
      <configSetting altId="p013.gpio_mode.gpio_mode_an"
configurationId="p013.gpio_mode"/>
      <configSetting altId="p014.input" configurationId="p014"/>
      <configSetting altId="p014.gpio_mode.gpio_mode_in"
configurationId="p014.gpio_mode"/>
      <configSetting altId="p015.input" configurationId="p015"/>
      <configSetting altId="p015.gpio_mode.gpio_mode_in"
configurationId="p015.gpio_mode"/>
      <configSetting altId="p100.sci0.rxd0" configurationId="p100"/>
      <configSetting altId="p100.gpio_mode.gpio_mode_peripheral"
configurationId="p100.gpio_mode"/>
      <configSetting altId="p100.gpio_otype.gpio_otype_n_ch_od"
configurationId="p100.gpio_otype"/>
      <configSetting altId="p101.sci0.txd0" configurationId="p101"/>
      <configSetting altId="p101.gpio_mode.gpio_mode_peripheral"
configurationId="p101.gpio_mode"/>
```

```
<configSetting altId="p101.gpio_otype.gpio_otype_n_ch_od"
configurationId="p101.gpio_otype"/>
       <configSetting altId="pl04.gpt1.gtioclb" configurationId="pl04"/>
#
       <configSetting altId="p104.gpio_mode.gpio_mode_peripheral"
#
configurationId="p104.gpio_mode"/>
      <configSetting altId="p105.input" configurationId="p105"/>
      <configSetting altId="p105.gpio_mode.gpio_mode_in"
configurationId="p105.gpio_mode"/>
      <configSetting altId="p106.input" configurationId="p106"/>
      <configSetting altId="p106.gpio_mode.gpio_mode_in"
configurationId="p106.gpio_mode"/>
      <configSetting altId="p107.input" configurationId="p107"/>
      <configSetting altId="p107.gpio_mode.gpio_mode_in"
configurationId="p107.gpio_mode"/>
      <configSetting altId="p108.jtag_fslash_swd.swdio" configurationId="p108"/>
      <configSetting altId="p108.gpio_mode.gpio_mode_peripheral"
configurationId="p108.gpio_mode"/>
      <configSetting altId="p109.sci9.txd9" configurationId="p109"/>
      <configSetting altId="p109.gpio_mode.gpio_mode_peripheral"
configurationId="p109.gpio_mode"/>
      <configSetting altId="p110.sci9.rxd9" configurationId="p110"/>
      <configSetting altId="p110.gpio_mode.gpio_mode_peripheral"
configurationId="p110.gpio_mode"/>
      <configSetting altId="pll1.input" configurationId="pll1"/>
      <configSetting altId="p111.gpio_mode.gpio_mode_in"
configurationId="p111.gpio_mode"/>
       <configSetting altId="pll2.gpt3.gtioc3b" configurationId="pll2"/>
#
       <configSetting altId="p112.gpio_mode.gpio_mode_peripheral"
#
configurationId="p112.gpio_mode"/>
      <configSetting altId="p113.input" configurationId="p113"/>
      <configSetting altId="p113.gpio_mode.gpio_mode_in"
configurationId="p113.gpio_mode"/>
      <configSetting altId="p205.input" configurationId="p205"/>
      <configSetting altId="p205.gpio_mode.gpio_mode_in"
configurationId="p205.gpio_mode"/>
      <configSetting altId="p206.ssie0.ssidata0" configurationId="p206"/>
      <configSetting altId="p206.gpio_mode.gpio_mode_peripheral"
configurationId="p206.gpio_mode"/>
      <configSetting altId="p207.ssie0.ssilrck0" configurationId="p207"/>
      <configSetting altId="p207.gpio_mode.gpio_mode_peripheral"
configurationId="p207.gpio_mode"/>
      <configSetting altId="p208.input" configurationId="p208"/>
      <configSetting altId="p208.gpio_mode.gpio_mode_in"
configurationId="p208.gpio_mode"/>
      <configSetting altId="p300.jtag_fslash_swd.swclk" configurationId="p300"/>
      <configSetting altId="p300.gpio_mode.gpio_mode_peripheral"
configurationId="p300.gpio_mode"/>
      <configSetting altId="p301.input" configurationId="p301"/>
      <configSetting altId="p301.gpio_mode.gpio_mode_in"
configurationId="p301.gpio_mode"/>
      <configSetting altId="p402.input" configurationId="p402"/>
      <configSetting altId="p402.gpio_mode.gpio_mode_in"
configurationId="p402.gpio_mode"/>
      <configSetting altId="p403.input" configurationId="p403"/>
      <configSetting altId="p403.gpio_mode.gpio_mode_in"
configurationId="p403.gpio_mode"/>
      <configSetting altId="p407.ssie0.ssibck0" configurationId="p407"/>
      <configSetting altId="p407.gpio_mode.gpio_mode_peripheral"
configurationId="p407.gpio_mode"/>
      <configSetting altId="p408.input" configurationId="p408"/>
      <configSetting altId="p408.gpio_mode.gpio_mode_in"
configurationId="p408.gpio_mode"/>
      <configSetting altId="p409.input" configurationId="p409"/>
      <configSetting altId="p409.gpio_mode.gpio_mode_in"
configurationId="p409.gpio_mode"/>
      <configSetting altId="p410.input" configurationId="p410"/>
```

```
<configSetting altId="p410.gpio_mode.gpio_mode_in"
configurationId="p410.gpio_mode"/>
      <configSetting altId="p411.output.high" configurationId="p411"/>
      <configSetting altId="p411.gpio_mode.gpio_mode_out.high"
configurationId="p411.gpio_mode"/>
      <configSetting altId="p500.input" configurationId="p500"/>
      <configSetting altId="p500.gpio_mode.gpio_mode_in"
configurationId="p500.gpio_mode"/>
      <configSetting altId="sci0.mode.simplei2c.free" configurationId="sci0.mode"/>
      <configSetting altId="sci0.rxd0.pl00" configurationId="sci0.rxd0"/>
      <configSetting altId="sci0.txd0.pl01" configurationId="sci0.txd0"/>
      <configSetting altId="sci9.mode.asynchronousuart.free"
configurationId="sci9.mode"/>
      <configSetting altId="sci9.rxd9.pl10" configurationId="sci9.rxd9"/>
      <configSetting altId="sci9.txd9.p109" configurationId="sci9.txd9"/>
      <configSetting altId="ssie0.mode.half_dash_duplex.c"
configurationId="ssie0.mode"/>
      <configSetting altId="ssie0.pairing.c" configurationId="ssie0.pairing"/>
      <configSetting altId="ssie0.ssibck0.p407" configurationId="ssie0.ssibck0"/>
      <configSetting altId="ssie0.ssidata0.p206" configurationId="ssie0.ssidata0"/>
<configSetting altId="ssie0.ssilrck0.p207" configurationId="ssie0.ssilrck0"/>
      <lockSetting id="gpt1.gtioc1b" lock="true"/>
+
    </pincfg>
    <pincfg active="false" name="R7FA6E2BB3CFM.pincfg" selected="false" symbol="">
       <configSetting altId="gpt1.gtioclb.p104" configurationId="gpt1.gtioclb"/>
+
       <configSetting altId="gpt1.mode.gtiocaorgtiocb.free"
configurationId="gpt1.mode"/>
      <configSetting altId="jtag_fslash_swd.mode.swd.free"
configurationId="jtag_fslash_swd.mode"/>
      <configSetting altId="jtag_fslash_swd.swclk.p300"
configurationId="jtag_fslash_swd.swclk"/>
      <configSetting altId="jtag_fslash_swd.swdio.pl08"
configurationId="jtag_fslash_swd.swdio"/>
       <configSetting altId="p104.gpt1.gtioc1b" configurationId="p104"/>
       <configSetting altId="p104.gpio_mode.gpio_mode_peripheral"
+
configurationId="p104.gpio_mode"/>
      <configSetting altId="p108.jtag_fslash_swd.swdio" configurationId="p108"/>
      <configSetting altId="p108.gpio_mode.gpio_mode_peripheral"
configurationId="p108.gpio_mode"/>
      <configSetting altId="p300.jtag_fslash_swd.swclk" configurationId="p300"/>
      <configSetting altId="p300.gpio_mode.gpio_mode_peripheral"
configurationId="p300.gpio_mode"/>
    </pincfg>
  </raPinConfiguration>
</raConfiguration>
```

Code Block 6 configuration.xml

Files to be added:

To initialize the timer PWM stack, middleware API support is required. The following are the files responsible for initializing the external interrupt stack. This function is called in **main_application.c** to initialize the stack.

```
bgk_ra6e2_baremetal/src/gpt_timer.c
bgk_ra6e2_baremetal/src/gpt_timer.h
```

The following is the content of the file that should be added to the project.

QC-BEKITPOC2Z Manual

```
* File Name : gpt_timer.c
* Description : Contains function definition.
* Copyright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates
* SPDX-License-Identifier: BSD-3-Clause
                        **********************************
#include "common_utils.h"
#include "gpt_timer.h"
#include "log_disabled.h"
//#include "log_error.h"
//#include "log_warning.h"
//#include "log_info.h"
//#include "log_debug.h"
/**********************
                   * @addtogroup r_gpt_ep
* @{
/* Boolean flag to determine one-shot mode timer is expired or not.*/
/* Store Timer open state*/
* @brief Initialize GPT timer.
* @param[in] p_timer_ctl Timer instance control structure
* @param[in] p_timer_cfg Timer instance Configuration structure
* @param[in] timer_mode Mode of GPT Timer
* @retval FSP_SUCCESS Upon successful op
                   Upon successful open of timer.
* @retval
         Any Other Error code apart from FSP_SUCCES on Unsuccessful open .
fsp_err_t init_gpt_timer(timer_ctrl_t * const p_timer_ctl, timer_cfg_t const * const
p_timer_cfg)
  fsp err t err = FSP SUCCESS;
  /* Initialize GPT Timer */
  err = R_GPT_Open(p_timer_ctl, p_timer_cfg);
  if (FSP_SUCCESS != err)
  {
    log_error ("\r\n ** R_GPT_TimerOpen FAILED ** \r\n");
    return err;
  }
  return err;
}
* @brief
         Start GPT timers in periodic, one shot, PWM mode.
* @param[in] p_timer_ctl Timer instance control structure
* @retval FSP_SUCCESS Upon successful start of timer.
```

```
* @retval
             Any Other Error code apart from FSP_SUCCES on Unsuccessful start .
fsp_err_t start_gpt_timer (timer_ctrl_t * const p_timer_ctl)
ł
   fsp_err_t err = FSP_SUCCESS;
   /* Starts GPT timer */
   err = R_GPT_Start(p_timer_ctl);
   if (FSP_SUCCESS != err)
   ł
       /* In case of GPT_open is successful and start fails, requires a immediate
cleanup.
        * Since, cleanup for GPT open is done in start_gpt_timer,Hence cleanup is not
required */
      log_error ("\r\n ** R_GPT_Start API failed ** \r\n");
   return err;
}
* @brief set duty cycle of PWM timer.
* @param[in] duty_cycle_percent.

    * @retval FSP_SUCCESS on correct duty cycle set.
    * @retval FSP_INVALID_ARGUMENT on invalid info.

fsp_err_t set_timer_duty_cycle(uint8_t duty_cycle_percent ,timer_ctrl_t * const
p_timer_ctl)
ł
   fsp_err_t err
                                      = FSP_SUCCESS;
   uint32_t duty_cycle_counts
uint32_t current_period_counts
                                     = RESET_VALUE;
                                    = RESET_VALUE;
   timer_info_t info
                                     = {(timer_direction_t)RESET_VALUE,
RESET_VALUE, RESET_VALUE };
   /* Get the current period setting. */
   err = R_GPT_InfoGet(p_timer_ctl, &info);
   if (FSP_SUCCESS != err)
   {
       /* GPT Timer InfoGet Failure message */
      log_error ("\r\n ** R_GPT_InfoGet API failed ** \r\n");
   }
   else
   {
       /* update period counts locally. */
      current_period_counts = info.period_counts;
       /\ast Calculate the desired duty cycle based on the current period. Note that if
the period could be larger than
       * UINT32_MAX / 100, this calculation could overflow. A cast to uint64_t is used
to prevent this. The cast is
       * not required for 16-bit timers. */
       duty_cycle_counts =(uint32_t) ((uint64_t) (current_period_counts *
duty_cycle_percent) /
             GPT_MAX_PERCENT);
#if defined(BOARD_RA4W1_EK) || defined (BOARD_RA6T1_RSSK) ||defined (BOARD_RA6T3_MCK) ||
defined (BOARD_RA4T1_MCK)
      duty_cycle_counts = (current_period_counts - duty_cycle_counts);
#endif
```
```
/* Duty Cycle Set API set the desired intensity on the on-board LED */
       err = R_GPT_DutyCycleSet(p_timer_ctl, duty_cycle_counts, TIMER_PIN);
       if(FSP_SUCCESS != err)
           /* GPT Timer DutyCycleSet Failure message */
           /* In case of GPT_open is successful and DutyCycleSet fails, requires a
immediate cleanup.
            * Since, cleanup for GPT open is done in timer_duty_cycle_set,Hence cleanup
is not required */
           log_error ("\r\n ** R_GPT_DutyCycleSet API failed ** \r\n");
        }
    }
   return err;
}
@brief set duty cycle of PWM timer.
@param[in] duty_cycle_percent.
@retval FSP_SUCCESS on correct duty cycle set.
* @brief
  eretvalFSP_SUCCESS on correct and<br/>oretval@retvalFSP_INVALID_ARGUMENT on invalid info.
* @retval
fsp_err_t set_timer_Period_and_Dutycycle(uint32_t period_counts ,uint8_t
duty_cycle_percent, timer_ctrl_t * const p_timer_ctl)
{
                                          = FSP_SUCCESS;
   fsp_err_t err
   uint32_t duty_cycle_counts
                                          = RESET_VALUE;
   uint32_t duty_cycle_counts
uint32_t current_period_counts
                                          = RESET_VALUE;
                                         = {(timer_direction_t)RESET_VALUE,
   timer_info_t info
RESET_VALUE, RESET_VALUE };
   gpt_instance_ctrl_t * p_instance_ctrl = (gpt_instance_ctrl_t *) p_timer_ctl;
    /* PeriodSet Set API set the desired intensity on the on-board LED */
   err = R_GPT_PeriodSet(p_timer_ctl, period_counts);
   if(FSP_SUCCESS != err)
        /* GPT Timer DutyCycleSet Failure message */
       /* In case of GPT_open is successful and \ensuremath{\mathsf{DutyCycleSet}} fails, requires a
immediate cleanup.
        * Since, cleanup for GPT open is done in timer_duty_cycle_set,Hence cleanup is
not required */
       log_error ("\r\n ** R_GPT_PeriodSet API failed ** \r\n");
    }
   current_period_counts = period_counts;
    /* Calculate the desired duty cycle based on the current period. Note that if the
period could be larger than
    * UINT32_MAX / 100, this calculation could overflow. A cast to uint64_t is used to
prevent this. The cast is
    * not required for 16-bit timers. */
   duty_cycle_counts =(uint32_t) ((uint64_t) (current_period_counts *
duty_cycle_percent) /
           GPT_MAX_PERCENT);
#if defined(BOARD_RA4W1_EK) || defined (BOARD_RA6T1_RSSK) ||defined (BOARD_RA6T3_MCK) ||
defined (BOARD_RA4T1_MCK)
   duty_cycle_counts = (current_period_counts - duty_cycle_counts);
#endif
    /* Duty Cycle Set API set the desired intensity on the on-board LED */
    err = R_GPT_DutyCycleSet(p_timer_ctl, duty_cycle_counts, TIMER_PIN);
   if(FSP_SUCCESS != err)
    {
```

```
/* GPT Timer DutyCycleSet Failure message */
     /* In case of GPT_open is successful and DutyCycleSet fails, requires a
immediate cleanup.
     * Since, cleanup for GPT open is done in timer_duty_cycle_set,Hence cleanup is
not required */
     log_error ("\r\n ** R_GPT_DutyCycleSet API failed ** \r\n");
  }
  return err;
}
******
* @brief Close the GPT HAL driver.
* @param[in] p_timer_ctl Timer instance control structure
* @retval None
**********
void deinit_gpt_timer(timer_ctrl_t * const p_timer_ctl)
{
  fsp_err_t err = FSP_SUCCESS;
  /* Timer Close API call*/
  err = R_GPT_Close(p_timer_ctl);
  if (FSP_SUCCESS != err)
  {
     /* GPT Close failure message */
     log_error("\r\n ** R_GPT_Close FAILED ** \r\n");
  }
}
* @} (end addtogroup r_gpt_ep)
```

Code Block 7 bgk_ra6e2_baremetal/src/gpt_timer.c

* File Name : gpt_timer.h * Description : Contains Macros and function declarations. ********************************** ***** * Copyright (c) 2020 - 2024 Renesas Electronics Corporation and/or its affiliates * SPDX-License-Identifier: BSD-3-Clause ***** #ifndef GPT_TIMER_H_ #define GPT_TIMER_H_ /* Max Duty Cycle percentage */ /* Macros definitions */ (100U) (16U) #define GPT_MAX_PERCENT #define BUF_SIZE /* Size of buffer for RTT input data */ /* To perform GPT Timer in Periodic #define PERIODIC_MODE_TIMER (1U) mode */ #define PWM_MODE_TIMER (2U) /* To perform GPT Timer in PWM mode */ #define ONE_SHOT_MODE_TIMER (3U) /* To perform GPT Timer in ONE-SHOT mode */ #define INITIAL_VALUE ('\0') /* timer unit in millisecond */ (1ULL) #define CLOCK_TYPE_SPECIFIER /* type specifier */ /* GPT Timer Pin for boards */ #define TIMER_PIN (GPT_IO_PIN_GTIOCB) #if defined (BOARD_RA2A1_EK) || defined (BOARD_RA4W1_EK) || defined (BOARD_RA4T1_MCK) || defined (BOARD_RA4E2_EK) #define GPT_MAX_PERIOD_COUNT (OXFFFF) /* Max Period Count for 16-bit Timer*/ #else #define GPT_MAX_PERIOD_COUNT (0XFFFFFFF) /* Max Period Count for 32-bit Timer*/ #endif #define PERIODIC_MODE (1U) /* To check status of GPT Timer in Periodic mode */ #define PWM_MODE (2U) /* To check status of GPT Timer in PWM mode */ #define ONE_SHOT_MODE /* To check status of GPT Timer in oNE-(3U) SHOT mode */ "\r\nThe project initializes GPT module in Periodic, PWM or One-shot #define EP INFO mode based on user input " \setminus "from the displayed menu options." \ "\r\nIn periodic mode, user can enter the time period within the permitted ranges to change " $\$ "the frequency of the user LED." \backslash "\r\nIn PWM mode, user can enter the duty cycle within the specified range to adjust the " $\$ "intensity of the user LED." \ "\r\nIn ONE SHOT mode, Output will be displayed on JlinkRTTViewer when timer expires.\r\n " /* Function declaration */ fsp_err_t init_gpt_timer(timer_ctrl_t * const p_timer_ctl, timer_cfg_t const * const p_timer_cfg); fsp_err_t start_gpt_timer (timer_ctrl_t * const p_timer_ctl); fsp err t set timer duty cycle(uint8 t duty cycle percent , timer ctrl t * const p_timer_ctl); uint32_t process_input_data(void); void deinit_gpt_timer(timer_ctrl_t * const p_timer_ctl);

```
void print_timer_menu(void);
fsp_err_t set_timer_Period_and_Dutycycle(uint32_t period_counts ,uint8_t
duty_cycle_percent, timer_ctrl_t * const p_timer_ctl);
#endif /* GPT_TIMER_H_ */
```

Code Block 8 bgk_ra6e2_baremetal/src/gpt_timer.h

Next, edit the main_application.c file as the following.

```
#include <stdio.h>
#include "common_utils.h"
#include "main_application.h"
+ #include "gpt_timer.h"
// Uncomment the desired debug level
#include "log_disabled.h"
//#include "log_error.h"
//#include "log_warning.h"
//#include "log_info.h"
//#include "log_debug.h"
+ #ifdef ENABLE_PWM_LED1
+ void enable_Led1_PWM(void)
+ {
   fsp_err_t err = FSP_SUCCESS;
   uint32_t gpt_desired_duty_cycle_percent = RESET_VALUE;
+
   uint32_t gpt_desired_period_ms
                                            = RESET VALUE;
+
+
   uint64_t period_counts
                                             = RESET_VALUE;
                                             = RESET_VALUE;
   uint32_t pclkd_freq_hz
+
   gpt_desired_duty_cycle_percent = LED1_DEFAULT_DUTY_CYCLE ; // edit here to change
duty cycle
   log_debug( "LED1:Default Period:%d ms Default Duty Cycle:%d \r\n",
+
((LED1_DEFAULT_PERIOD* 1000) /SET_PERIOD_1_SECS),LED1_DEFAULT_DUTY_CYCLE);
    /* Validate Duty cycle percentage */
+
   if (GPT_MAX_PERCENT < gpt_desired_duty_cycle_percent)</pre>
+
+
   {
        log_error ("\r\n ** LED1:INVALID INPUT, DESIRED DUTY CYCLE IS OUT OF RANGE. **
+
r^n);
+
    }
+
    else
+
    {
            /* we got valid input, Initialize PWM timer */
+
            err = init_gpt_timer(&g_timer_pwm_led1_ctrl, &g_timer_pwm_led1_cfg);
+
            if(FSP_SUCCESS != err)
+
+
            {
+
                log_error("** LED1:GPT TIMER INIT FAILED ** \r\n");
+
            }
            log_debug("LED1:Opened Timer in PWM Mode\r\n");
+
+
            /* Start PWM Timer*/
+
+
            err = start_gpt_timer(&g_timer_pwm_led1_ctrl);
+
            if(FSP_SUCCESS != err)
+
            {
                log_error("** LED1:GPT TIMER START FAILED ** \r\n");
+
                /*Close PWM Timer instance */
+
                deinit_gpt_timer(&g_timer_pwm_led1_ctrl);
+
+
            log_debug("LED1:Started Timer in PWM Mode\r\n");
+
+
        /* Set DutyCycle and period of PWM timer */
+
            err =
+
set_timer_Period_and_Dutycycle(LED1_DEFAULT_PERIOD,gpt_desired_duty_cycle_percent,&g_tim
er_pwm_led1_ctrl);
            if(FSP_SUCCESS != err)
+
+
            {
                 /* GPT Timer DutyCycleSet Failure message */
+
                log_error ("\r\n ** LED1:Timer Period and Dutycycle SET FAILED **
r^n);
                /*Close PWM Timer instance */
+
+
                deinit_gpt_timer(&set_timer_Period_and_Dutycycle);
            }
+
    }
+
+ }
```

```
+#endif
+ #ifdef ENABLE_PWM_LED2
+ void enable_Led2_PWM(void)
+ {
    fsp_err_t err = FSP_SUCCESS;
+
    uint32_t gpt_desired_duty_cycle_percent = RESET_VALUE;
+
    uint32_t gpt_desired_period_ms
                                       = RESET_VALUE;
+
+
    uint64_t period_counts
                                             = RESET_VALUE;
+
   uint32_t pclkd_freq_hz
                                             = RESET_VALUE;
   gpt_desired_duty_cycle_percent = LED2_DEFAULT_DUTY_CYCLE ; // edit here to change
+
duty cycle
    log_debug( "LED2:Default Period:%d ms Default Duty Cycle:%d \r\n",
+
((LED2_DEFAULT_PERIOD* 1000) /SET_PERIOD_1_SECS),LED2_DEFAULT_DUTY_CYCLE);
    /* Validate Duty cycle percentage */
+
   if (GPT_MAX_PERCENT < gpt_desired_duty_cycle_percent)
+
+
   {
        log_error ("\r\n ** LED2:INVALID INPUT, DESIRED DUTY CYCLE IS OUT OF RANGE. **
+
r^n);
+
    }
    else
+
+
    {
            /* we got valid input, Initialize PWM timer */
+
+
            err = init_gpt_timer(&g_timer_pwm_led2_ctrl, &g_timer_pwm_led2_cfg);
            if(FSP_SUCCESS != err)
+
+
            {
                log_error("** LED2:GPT TIMER INIT FAILED ** \r\n");
+
+
+
            log_debug("LED2:Opened Timer in PWM Mode\r\n");
+
            /* Start PWM Timer*/
+
            err = start_gpt_timer(&g_timer_pwm_led2_ctrl);
+
+
            if(FSP_SUCCESS != err)
+
            {
                log_error("** LED2:GPT TIMER START FAILED ** \r\n");
+
                /*Close PWM Timer instance */
+
                deinit_gpt_timer(&g_timer_pwm_led2_ctrl);
+
            }
+
            log_debug("Started Timer in PWM Mode\r\n");
+
+
        err =
set_timer_Period_and_Dutycycle(LED2_DEFAULT_PERIOD,gpt_desired_duty_cycle_percent,&g_tim
er_pwm_led2_ctrl);
        if(FSP_SUCCESS != err)
+
+
        ł
             /* GPT Timer DutyCycleSet Failure message */
+
            log_error ("\r\n ** LED2:Timer Period and Dutycycle SET FAILED ** \r\n");
+
            /*Close PWM Timer instance */
+
            deinit_gpt_timer(&set_timer_Period_and_Dutycycle);
        }
+
+
    }
+ }
+ #endif
void main_application(void) {
// Start of autogenerated code
+ #if defined(ENABLE_LED1) && defined(ENABLE_PWM_LED1)
+ enable_Led1_PWM();
+ #endif
+ #if defined(ENABLE_LED2) && defined(ENABLE_PWM_LED2)
+ enable_Led2_PWM();
+ #endif
```

while(true)
{
 // do nothing
 }
// End of autogenerated code
}

Code Block 9 main_application.c

Updated main_application.h file.

```
#ifndef __MAIN_APPLICATION_H
#define ___MAIN_APPLICATION_H
#include "common_utils.h"
+ #define SET_PERIOD_1_SECS
                                 0x10000 // 1 second period
+ #define SET_PERIOD_500_MSECS
                                 0x8000 // 500 ms period
+ #define SET_PERIOD_250_MSECS
                                 0x4000 // 250 ms period
+ #define SET_PERIOD_125_MSECS
                                 0x2000 // 125 ms period
                                          // 75 ms period
+ #define SET_PERIOD_75_MSECS
                                 0x1000
                                          // 37.5 ms period
+ #define SET_PERIOD_37_MSECS
                                 0x800
+ #define SET_PERIOD_16_MSECS
                                        // 16 ms period
                                 0x400
                                 0x200 // 8 ms period
+ #define SET_PERIOD_8_MSECS
+ #define SET_PERIOD_4_MSECS
                                 0x100 // 4 ms period
                                        // 2 ms period
+ #define SET_PERIOD_2_MSECS
                                 0x80
                                         // 1 ms period
+ #define SET_PERIOD_1_MSECS
                                 0x40
+ #define ENABLE_LED1
                       // Enable or disable LED1
+ #define ENABLE_LED2 // Enable or disable LED2
+ #ifdef ENABLE_LED1
+ #define ENABLE_PWM_LED1
+ #endif
+ #ifdef ENABLE_LED2
+ #define ENABLE_PWM_LED2
+ #endif
+ #ifdef ENABLE_PWM_LED1
+ #define LED1_DEFAULT_DUTY_CYCLE 50 // 50 %
+ #define LED1_DEFAULT_PERIOD
                                     SET_PERIOD_1_SECS
+ #endif
+ #ifdef ENABLE_PWM_LED2 // 50 %
+ #define LED2_DEFAULT_PERIOD
                                    SET_PERIOD_1_SECS
+ #define LED2_DEFAULT_DUTY_CYCLE 50
+ #endif
/* Function declaration */
void main_application(void);
#endif /* __MAIN_APPLICATION_H */
```

Code Block 10 main_application.h

The user can enable or disable the PWM code for individual LEDs using the macros provided below. To use this feature, the user must make the following changes in **main_application.h**:

Enable the Macro:

#define ENABLE_LED1	// Enable or disable LED1
#define ENABLE_LED2	// Enable or disable LED2

Expected Output RTT Logs:

Check the following logs for the expected behavior and also observe LEDs on Board to co-relate with the feature.

🔜 J-Link RTT Viewer V8.10h	_	\times
File Terminals Input Logging Help		
All Terminals Terminal 0		
<pre>00> DEBUG:/src/main_application.c LED1:Default Period:1000 ms Default Duty Cycle: 00> DEBUG:/src/main_application.c LED1:Opened Timer in PWM Mode 00> DEBUG:/src/main_application.c LED1:Started Timer in PWM Mode 00> 00> DEBUG:/src/main_application.c LED2:Default Period:1000 ms Default Duty Cycle:S 00> DEBUG:/src/main_application.c LED2: Opened Timer in PWM Mode 00> 00> DEBUG:/src/main_application.c LED2: Opened Timer in PWM Mode 00> 00> DEBUG:/src/main_application.c LED2: Started Timer in PWM Mode 00> 00> DEBUG:/src/main_application.c LED2:Started Timer in PWM Mode 00></pre>	50 0	

10.3.2.4 Adjust the LED Period using Macro

By default, both LEDs are configured with a 1s period. However, the blink interval can be modified by changing the macro values in **main_application.h**:

- For LED1: #define LED1_DEFAULT_PERIOD
- For LED2: #define LED2_DEFAULT_PERIOD

Any of the previous defined values can be selected and set with LED periods like this:

#define	LED1_DEFAULT_PERIOD	SET_PERIOD_1_SECS
#define	LED2_DEFAULT_PERIOD	SET_PERIOD_1_SECS

These macros can be adjusted to achieve the required LED blink intervals.

Expected Output RTT Logs:

Check the following logs for the expected behavior and also observe LEDs on board to co-relate with the feature.

Lak J-Link RTT Viewer V8.10h —	\times
File Terminals Input Logging Help	
All Terminals Terminal 0	
00> DEBUG:/src/main_application.c LED1:Default Period:1000 ms Default Duty Cycle:50 00>	
00> DEBUG:/src/main_application.c LED1:Opened Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED1:Started Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED2:Default Period:1000 ms Default Duty Cycle:50 00>	
00> DEBUG:/src/main_application.c LED2: Opened Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED2:Started Timer in PWM Mode 00>	

10.3.2.5 Adjust the LED ON/OFF duration using macro

Users can adjust the On/OFF duration (that is the duty cycle) using the predefined macro in **main_application.h** file.

These macros as by default set a value of 50% for a duty cycle but they can be set as per required from the range of 1 to 100%.

- #define LED1_DEFAULT_DUTY_CYCLE 50 // duty cycle macro for LED 1
- #define LED2_DEFAULT_DUTY_CYCLE 50 // duty cycle macro for LED 2

The LEDs are configured with a 1-second period, allowing to be set the duty cycle from 1% to 100%.

- A 1% duty cycle means the LED will be ON for 10ms and OFF for 990ms.
- A 100% duty cycle means the LED stays ON continuously.

This way, users can adjust the ON/OFF duration anywhere between 10ms (minimum ON time) and 990ms (maximum OFF-time), depending on the required brightness or behavior.

Note: To control the LED brightness using PWM, the PWM period should be reduced to a level where the LED's on/off toggling is not visually perceptible. This allows for smooth brightness control through the duty cycle.

In this example, set the PWM period to 37ms by using the macro SET_PERIOD_37_MSECS for LED1_DEFAULT_PERIOD, and then vary the duty cycle between 1% to 100% to adjust brightness.

Example:

Maximum brightness

```
#define LED1_DEFAULT_DUTY_CYCLE
#define LED1_DEFAULT_PERIOD
```

100 // 100 % SET_PERIOD_37_MSECS

Minimum brightness

```
#define LED1_DEFAULT_DUTY_CYCLE
#define LED1_DEFAULT_PERIOD
```

1 // 1 % SET_PERIOD_37_MSECS

Expected Output RTT Logs:

Check the following logs for the expected behavior and also observe LEDs on the board to correlate with the feature.

File Terminals Input Logging Help	
All Terminals Terminal 0	
00> DEBUG:/src/main_application.c LED1:Default Period:1000 ms Default Duty Cycle:50	
00> DEBUG:/src/main_application.c LED1:Opened Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED1:Started Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED2:Default Period:1000 ms Default Duty Cycle:50 00>	
00> DEBUG:/src/main_application.c LED2: Opened Timer in PWM Mode 00>	
00> DEBUG:/src/main_application.c LED2:Started Timer in PWM Mode 00>	

10.3.3 Customize Sensor Data to AWS Cloud application

As a first step, the user must create a project in the QuickConnect Studio by selecting the appropriate MCU kit (for example, BGK-RA6E2), sensor (for example, HS4001 Temperature / Humidity), connectivity module (for example, DA16600 Wi-Fi) and Sensor Data to AWS Cloud (MQTT onchip) application

10.3.3.1 Exercise 1: Change Sensor Units (C to F or F to C)

This exercise is to update the temperature units between C and F. The default unit for temperature on QCS is Celsius. This exercise publishes temperature data into AWS in Fahrenheit. Files to be modified:

\src\main_thread_entry.c

Modifications:

Modify the following code in the main_thread_entry() function to convert the sensor data if the parameter is temperature only.

```
void main_thread_entry(void *pvParameters) {
FSP_PARAMETER_NOT_USED (pvParameters);
uint8_t timeout = 10;
while (1) {
    sm_handle sensor_handle;
    uint8_t sensor_count = 0;
        if (pdTRUE == xQueueReceive(g_sensor_queue,
   &sensor_data, PUBLISHING_INTERVAL_MS)) {
            // Store the received data into the right slot
            for (index = 0; index < NUM_SENSORS; index++) {</pre>
                if (0 == sensor_slots[index].handle.value ||
   sensor_slots[index].handle.value ==
   sensor_data.handle.value) {
                     // Found a slot
                    sensor_slots[index].handle.value =
   sensor_data.handle.value;
                     sensor_slots[index].data =
   sensor_data.data;
                    sensor_slots[index].data = (TEMPERATURE
+
   ==
   sm_get_sensor_type_by_handle(sensor_slots[index].handle))
   ? \
+
   ((sensor_data.data * 9) / 5) + 3200 : \
+
   sensor_data.data;
                     num_sensors = index + 1;
                    break;
                 }
            }
        }
    }
}
                  Code Block 11 main_thread_entry.c changes
```

Expected output:

- AWS IOT MQTT test client
 - » <user_device>/feeds/temperature
 - 93.06
 - » Properties
 - » <user_device>/feeds/humidity
 - 57.13
 - » Properties

March 23, 2025, 16:26:51 (UTC+0530)

March 23, 2025, 16:26:51 (UTC+0530)

RTT Viewer

INFO:../src/sensor/hs4001_sensor.c Sensor channel 0 open success
INFO:../src/sensor/hs4001_sensor.c Sensor channel 1 open success
DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
INFO:../src/main_thread_entry.c Starting Wi-Fi connection...
INFO:../src/main_thread_entry.c Wi-Fi connection successful!
INFO:../src/main_thread_entry.c MQTT setup successful!
INFO:../src/main_thread_entry.c MQTT Subscribe to user_device/feeds/led
DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 1
DEBUG:../src/main_thread_entry.c Topic: <user_device>/feeds/humidity data: 57.13
DEBUG:../src/main_thread_entry.c Topic: <user_device>/feeds/temperature data: 93.06

10.3.3.2 Change Frequency of Sensor Update

This exercise is to change the sensor update time from 10 seconds to one minute. The default time for the sensor update on QCS is 30 seconds. This exercise changes the sensor data publish interval the into AWS cloud.

Files to be modified:

\src\main_thread_entry.c

Modifications: Define the following mentioned enum in the User Macros section and assign the variable to PUBLISHING_INTERVAL_M:

```
* * * * * * * * * * * * * * * * * * *
User Macros
* * * * * * * * * * * * * * * * * /
+// Define sensor update periods (in milliseconds)
+typedef enum {
    PERIOD_{10SEC} = 10000,
+
+
    PERIOD_20SEC = 10000,
   PERIOD_30SEC = 30000,
+
   PERIOD_45SEC = 45000,
   PERIOD_1MIN = 60000
+} SensorUpdatePeriod_t;
+SensorUpdatePeriod_t sensor_update_period = PERIOD_30SEC;
  // Default setting
#define QUEUE_RECEIVE_TIMEOUT
                                          (100)
                                                 //
  this is the interval we also receive MQTT messages
-#define PUBLISHING_INTERVAL_MS
                                          (30000)
+#define PUBLISHING_INTERVAL_MS
  sensor_update_period
```

Code Block 12 main_thread_entry.c changes

10.3.3.3 Modify the MQTT Topic Name as <sensor name>/<parameter>

This exercise is to modify the **aws mqqt** publish topic name includes sensor name. The default string in the topic name in QCS is feeds. This is modified by the sensor name (for example, hs4001_sensor).

Files to be modified:

Isrc\main_thread_entry.c

Modifications: Define the following macro mentioned in User Macros section and change the mentioned code in main_thread_entry() function:

```
* * * * * * * * * * * * * * * * * * *
User Macros
* * * * * * * * * * * * * * * * * /
+// create automatic headers for each sensor
+#define DEFINE_SENSOR_DRIVER(DRIVER) char sensor_name[] =
  #DRIVER;
+#include "sm_define_sensors.inc"
****
User function implementations
                        * * * * * * * * * * * * * * * * * * /
void main_thread_entry(void *pvParameters) {
FSP_PARAMETER_NOT_USED (pvParameters);
uint8_t timeout = 10;
while (1) {
   sm_handle sensor_handle;
   uint8_t sensor_count = 0;
            char pub_message[WIDTH_64];
            char pub_topic[WIDTH_64];
            snprintf(pub_topic, WIDTH_64,
  IO_USERNAME"/feeds/%s",\
            snprintf(pub_topic, WIDTH_64,
  IO_USERNAME"/%s/%s",\
                   sensor_name, \
   sm_get_sensor_path_by_handle(sensor_slots[index2].handle)
   , \setminus
                  );
   }
}
```

Code Block 13 main_thread_entry.c changes

Expected output:

AWS IoT MQTT test client	
» <user_device>/hs4001_sensor/temperature</user_device>	March 23, 2025, 16:26:51 (UTC+0530)
34.08	
» Properties	
» <user_device>/hs4001_sensor/humidity</user_device>	March 23, 2025, 16:26:51 (UTC+0530)
52.86	
» Properties	
RTT Viewer	

- 00> INFO:../src/sensor/hs4001_sensor.c Sensor channel 0 open success
- 00> INFO:../src/sensor/hs4001_sensor.c Sensor channel 1 open success
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
- 00> INFO:../src/main_thread_entry.c Starting Wi-Fi connection...
- 00> INFO:../src/main_thread_entry.c Wi-Fi connection successful!
- 00> INFO:../src/main_thread_entry.c MQTT setup successful!
- 00> INFO:../src/main_thread_entry.c MQTT Subscribe to <user_device>/feeds/led
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 1
- 00> DEBUG:../src/main_thread_entry.c Topic: <user_device>/hs4001_sensor/humidity data: 52.86
- 00> DEBUG:../src/main_thread_entry.c Topic: <user_device>/hs4001_sensor/temperature data: 34.08

10.3.3.4 Identify Min, Max, and Averaging of Sensor Values (Send Only when Absolute Change Observed)

This exercise is to calculate the statistics (min, max average, and absolute change) of a sensor parameter and publish the statistics.

Files to be modified:

\src\main_thread_entry.c

Modifications: Define the following mentioned enum in the User Macro section, add the variables to the User Global Variables section, and assign the variable to PUBLISHING_INTERVAL_MS as mentioned:

```
* * * * * * * * * * * * * * * * * *
User Macros
********************
+// Define statistics offsets
+typedef enum {
  VALUE = 0,
+
+
  MIN,
+
  MAX,
+
  AVG,
  ABS,
+
  COUNT,
+
+
  STATS_MAX_PARAMS
+}SensorStatsParams_t;
* * * * * * * * * * * * * * * * * * *
User global variables
* * * * * * * * * * * * * * * * * /
+static const char* stats_names[] = { "value", "minimum",
  "maximum", "average", "absolute_change"};
+#define STATS_NAMES_COUNT (sizeof(stats_names) /
  sizeof(stats_names[0]))
+static int32_t stats_values[NUM_SENSORS][STATS_MAX_PARAMS];
* * * * * * * * * * * * * * * * * * *
User function prototype declarations
                  ******
+void update_sensor_stats(int32_t new_value, uint32_t
  senor_index);
****
User function implementations
* * * * * * * * * * * * * * * * * /
+// Function to update sensor statistics
+void update_sensor_stats(int32_t new_value, uint32_t
  senor_index) {
   if(0 == stats_values[senor_index][COUNT])
+
+
   {
      stats values[senor index][MIN]
                                  = new value;
+
      stats_values[senor_index][MAX] = new_value;
+
      stats_values[senor_index][AVG]
+
                                 = new_value;
                                 = new_value;
+
      stats_values[senor_index][VALUE]
+
      stats_values[senor_index][ABS]
                                  = 0;
+
      stats_values[senor_index][COUNT]++;
+
      return;
+
+
   if (new_value < stats_values[senor_index][MIN]) {</pre>
      stats_values[senor_index][MIN] = new_value;
+
   }
+
+
   if (new_value > stats_values[senor_index][MAX]) {
+
      stats_values[senor_index][MAX] = new_value;
+
   }
+
   stats_values[senor_index][AVG] =
+
  (stats_values[senor_index][AVG] *
```

RENESAS

```
stats_values[senor_index][COUNT] + new_value) /
   (stats_values[senor_index][COUNT] + 1);
     stats_values[senor_index][COUNT]++;
+
     stats_values[senor_index][ABS] =
                                         (new_value >
+
   stats_values[senor_index][VALUE]) ? \
                                          (new_value -
+
   stats_values[senor_index][VALUE]) : \
+
   (stats_values[senor_index][VALUE] - new_value);
     stats_values[senor_index][VALUE] = new_value;
+
+ }
+
void main_thread_entry(void *pvParameters) {
FSP_PARAMETER_NOT_USED (pvParameters);
uint8_t timeout = 10;
while (1) {
    sm handle sensor handle;
    uint8_t sensor_count = 0;
               char pub_message[WIDTH_64];
               char pub_topic[WIDTH_64];
               snprintf(pub_topic, WIDTH_64,
   IO_USERNAME"/feeds/%s",\
   sm get sensor path by handle(sensor slots[index2].handle)
                          );
_
                  snprintf((char*)pub_message, WIDTH_64,
_
   "%ld.%02ld",\
                          sensor_slots[index2].data/100, \
                          sensor_slots[index2].data%100);
_
+
   update_sensor_stats(sensor_slots[index2].data, index2);
+
               for(SensorStatsParams_t stats_index = 0;
   stats_index < STATS_NAMES_COUNT; stats_index++){</pre>
+
                      snprintf(pub_topic, WIDTH_64,
   IO_USERNAME"/feeds/%s/%s",\
+
   sm_get_sensor_path_by_handle(sensor_slots[index2].handle)
   , \
+
                             stats_names[stats_index]\
+
                              );
                      snprintf((char*)pub_message, WIDTH_64,
+
   "%ld.%02ld",\
+
   stats_values[index2][stats_index]/100, \
+
   stats_values[index2][stats_index]%100);
                    mqtt_onchip_dal6xxx_pub_info_t pubData;
                     pubData.qos = MQTT_ONCHIP_DA16XXX_QOS_0,
                    pubData.p_topic_name = pub_topic;
                    pubData.topic_name_Length =
   (uint16_t)strlen(pub_topic);
                    pubData.p_payload = pub_message;
                    pubData.payload_length =
   strlen(pubData.p_payload);
                     log_debug("Topic: %s data:
   %s",pub_topic, pub_message);
                     /* Publish data to the MQTT Broker */
```



Code Block 14 main_thread_entry.c changes

Expected output:

•	AWS IoT MQTT test client	
	» <user_device>/feeds/temperature/absolute_change</user_device>	April 04, 2025, 17:46:12 (UTC+0530)
	0.04	
	» Properties	
	» <user_device>/feeds/temperature/average</user_device>	April 04, 2025, 17:46:11 (UTC+0530)
	34.32	
	» Properties	
	» <user_device>/feeds/temperature/maximum</user_device>	April 04, 2025, 17:46:11 (UTC+0530)
	34.37	
	» Properties	
	» <user_device>/feeds/temperature/minimum</user_device>	April 04, 2025, 17:46:11 (UTC+0530)
	34.31	
	» Properties	
	» <user_device>/feeds/temperature/value</user_device>	April 04, 2025, 17:46:10 (UTC+0530)
	34.33	
	» Properties	

RTT Viewer

- 00> INFO:../src/sensor/hs4001_sensor.c Sensor channel 0 open success
- 00> INFO:../src/sensor/hs4001_sensor.c Sensor channel 1 open success
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
- 00> INFO:../src/main_thread_entry.c Starting Wi-Fi connection...
- 00> INFO:../src/main_thread_entry.c Wi-Fi connection successful!
- 00> INFO:../src/main_thread_entry.c MQTT setup successful!
- 00> INFO:../src/main_thread_entry.c MQTT Subscribe to user_device/feeds/led
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 0
- 00> DEBUG:../src/sensor/hs4001_sensor.c Sensor read channel 1
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/humidity/value data: 56.86
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/humidity/minimum data: 56.86
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/humidity/maximum data: 57.35
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/humidity/average data: 57.17
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/humidity/absolute_change data: 0.27
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/temperature/value data: 34.33
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/temperature/minimum data: 34.31
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/temperature/maximum data: 34.33
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/temperature/average data: 34.32
- 00> DEBUG:../src/main_thread_entry.c Topic:<user_device>/feeds/temperature/absolute_change data: 0.02

10.3.3.5 Add Widget (Gauge / Box)

QCS can only publish the sensor data to the AWS cloud. It does not display like a widget. This can be implemented by installing a third-Party dashboard like AWS cloud watch, Grafana, Node-RED, or AWS Amplify with IoT Dashboard (Custom UI). For ease of exercise, use a direct third-party executable <u>MQTT explorer</u> to show the sensor value in widget and graph. Follow these steps to configure MQTT explorer.

1. Open MQTT Explorer

a. Launch the **.exe** file and click + to add a new connection.

	MQTT Connection	Topic matt://matt.eclipse.org.1883/	
matt.eclipse.org	Name mqtt.eclipse.org	Validate certificate	Encryption (tis)
met Heat monorite ang 1985	Protocol Host <u>mqtt:// +</u> mqtt.eclipse.	org	Port 1883
	Username	Password	8
		DVANCED	

- 2. Set the Connection Settings and input the basic information:
 - a. Name: AWS IoT
 - b. Protocol: mqtts:// (use secure MQTT over TLS)

- c. Host: Enter the AWS IoT endpoint, for example: a3k7example8-ats.iot.us-west-2.amazonaws.com
- d. Port: 8883 (TLS secured MQTT)

rr rr te	+ Connections nqtt.eclipse.org nqt//a1nu1ssmczosrp-ats.iot est.mosquitto.org nqt//rest.mosquitto.org1883/	MQTT Connection mqtt://a1r Name AWS	nu1*******-ats.tot.us-east-1.amazon	aws.com: Encryption (tls)
A	WS	Protocol Host mqtt://	-east-1.amazonaws.com	Port 8883
		Username youraws_rootuser_mail@gmail.com	Password aws_account_password	٥
		DELETE TADVANCED	SAVE	() CONNECT
 Enable TLS/SSL a. Toggle Enable 	TLS			
	Encrypt	tion (tls)		
b. Click on Adva	nced.			
	DELETE	ADVANCED	SAVE	() CONNECT
c. Click on Certif	icates.			
	MQTT Client IE	0		CATES ACP

- 4. Upload the certificate files:
 - a. Client Certificate: The device-certificate.pem.crt
 - b. Client Key: The private.pem.key
 - c. CA Certificate: Amazon root CA (for example, AmazonRootCA1.pem) <u>Download CA Cert here</u>

+ Connections	MQTT Connection mqtt://a1nu1ssmxzosrp-ats.ioLus-east-1.amazonaws.co
mqtt.eclipse.org mqtt.//a1nu1ssmxzosrp-ats.iot	
test.mosquitto.org mqtt.//test.mosquitto.org.1883/	× AmazonRootCA1 (9).pem
AWS mgtt://a1nu1ssmucrosrp_ats.iot	CLIENT CERTIFICATE x 7a9ea8b368ec3f1378cc8fed51a84fabc56304a1e6fb4b226bbd842fa565a863-certificate.pem.crt
new connection	
	★ / 8998003006C31137 0CC019031804180C3030481e01040220000042183038003-private, peritikey
	K BACK

- 5. Client ID
 - a. Optional, but recommended: set a unique client ID, for example, mqtt-explorer-client.
- 6. Click Back and Enter topic
 - a. Retain or Clean Session can be enabled if required.
 - b. Add Topic name (for example, <user_device>/feeds/temperature)

MOTT Connection

c. Go back and click **Save**.

nti_de	vice/feeds/temperature	0 👻	TAD
	Торіс		QoS
Î	Vinutha/feeds/temperature		0
Î	kranti_device/feeds/temperature		0

mott://a1nu1esmyzosrn-ats jot us-past-1 amazonaws co

- 7. Connect!
 - a. Click Connect.
 - b. If everything is set up right, the MQTT topics appear and start to populate. (Wait for updated interval period that the user has chosen.)
- 8. MQTT Explorer
 - a. Expand the Username > feeds, to see the real-time value.



b. Click on History to see the history and add charts to observe the real-time values graphically.



Note: <user_device> is replaced by IO_USERNAME from the src/user.h in all output logs.

10.3.4 Customize Sensor Data to BLE application

As a first step, the user must create a project in the QuickConnect Studio by selecting the appropriate MCU kit (BGK-RA6E2), sensor (HS4001 Temperature / Humidity), connectivity module (DA14531 Bluetooth) and Sensor Data over BLE FreeRTOS application.

10.3.4.1 Change Temperature Sensor Units (C to F and F to C)

This exercise is to update the temperature units between Celsius (C) and Fahrenheit (F). The default unit for temperature on QCS is Celsius. This exercise will convert the value to Fahrenheit and display it on the QC sandbox mobile application. Since the QC sandbox application needs to display the correctly converted data as well as the correct unit, Both changes need to be made to achieve the correct result.

Complete the following steps:

- 1. Generate the Sensor Data over BLE application on QCS using BGK-RA6E2, HS4001 Sensor, and DA14531 BLE module for FreeRTOS.
- 2. Modify the **ble_app.c** and **gui_cfg.json** files as mentioned above.
- 3. Rebuild the project by pressing the **Build QCStudio Project** option in QCS.
- 4. Download the *.srec file from the **Debug/** path and flash the BGK-RA6E2 board.
- 5. Pair the QC Sandbox mobile application with the board and verify the data.

Files modified:

- / src / qe_gen / ble / ble_app.c
- / gui_cfg.json

Modifications:

ble_app.c: The ble_app_run() function performs the BLE stack execution and obtains the data from FreeRTOS queue whenever it is available. This data is pushed upon receiving the request from mobile application. This function must be modified to convert the temperature value to Fahrenheit. Because this function is generic and is called for both temperature and humidity, the conversion should only take place in the case of temperature. The modification is made to convert only temperature data. (The complete function is not included below, the + symbol indicates you must add those lines inside the original function for the **ble_app.c** file).

```
void ble_app_run(void) {
    /* Process BLE Event */
   R_BLE_Execute();
    if (pdTRUE == xQueueReceive(g_sensor_queue, &sensor_data, 0)) {
        uint32_t index = 2;
        while (index < (uint32_t)sm_get_total_sensor_count()+2) {</pre>
            if (qc_sv_req_handlers[index].sensor_handler.value ==
sensor_data.handle.value) {
                sm_scaling scaling;
                sm_get_sensor_scaling(sensor_data.handle, &scaling);
                float floatData = ((float)sensor_data.data *
(float)scaling.multiplier)/(float)scaling.divider + (float)scaling.offset;
                if
(sm_get_sensor_type_by_handle(qc_sv_req_handlers[index].sensor_handler) == TEMPERATURE)
+
                ł
+
                    floatData = ((floatData * 18) / 10) + 32;
                }
                sensor_data_array[index-2] = floatData;
                break;
            index++;
        }
    }
}
```

Code Block 15 ble_app.c

gui_cfg.json: Parameter units for TEMPERATURE indicates the unit that should be shown against the value in mobile application. Default value for this field is C indicating the value is in Celsius, which should be modified to F because the application is pushing temperature data in Fahrenheit.

```
{
    "gui":
    {
         "id": "0000",
         "title": "US000 QC Example",
        "version:": "0.0.1",
        "tab":
         [
             {
                  "id": "0001",
                  "name": "LED",
                  "toggle":
                  {
                      "id": "0100",
                      "type": "uint8_t",
"name": "LED 1",
                      "default": 0
                 }
             },
{
                  "id": "0002",
                  "name": "HS4001",
                  "parameters":
                  {
                      "id": "0200",
                      "type": "blob",
                      "access": "r",
                      "list":
                       [
                           {
                               "id": "0201",
                               "name": "TEMPERATURE",
"units": "F",
                               "type": "float",
                               "access": "r",
                               "auto_read": 1,
                               "value": null
                           },
{
                               "id": "0202",
                               "name": "HUMIDITY",
"units": "%",
                               "type": "float",
                               "access": "r",
                               "auto_read": 1,
                               "value": null
                           }
                      ]
                 }
             },
{
                  "id": "0003",
                  "name": "About",
                  "parameters":
                  {
                      "id": "0300",
                      "type": "blob",
                      "access": "r",
                      "list":
                      [
                           {
                               "id": "0301",
                               "name": "Test Version",
                               "type": "string",
                               "access": "const",
                               "value": null
```



Code Block 16 gui_cfg.json

Outcome for this exercise:

After flashing the updated application into BGK-RA6E2 board, open the QC Sandbox mobile application and navigate to HS4001 page. There the user can see the temperature value in Fahrenheit format and the updated unit as well.

4:04	al S	26)		
REI	NESAS	i		
LED	S4001 About			
TEMPERATURE	78.26000	F		
HUMIDITY	61.68000	%		
C Read All				

10.3.4.2 Change Frequency of Sensor Value Update

This exercise involves updating the time interval at which data is read from the sensor. It adds a custom time interval value for sensor reads, which is reflected on the QC Sandbox mobile application. This update does not modify the interval at which the mobile application requests data from the board.

Complete the following steps:

- 1. Generate the Sensor Data over BLE application on QCS using BGK-RA6E2, HS4001 Sensor, and DA14531 BLE module for FreeRTOS.
- 2. Modify the **sensor_thread_entry.c** files as mentioned above.
- 3. Rebuild the project by pressing the Build QCStudio Project option in QCS.
- 4. Download the ***.srec** file from **Debug/** path and flash the BGK-RA6E2 board.
- 5. Pair the QC Sandbox mobile application with the board and verify the data.

Files modified:

/ src / sensor_thread_entry.c

Modification:

Sensor thread is responsible for the sensor related activities inside this application. That is initializing the sensor manager and reading and parsing the sensor data. To introduce the custom time interval for a sensor data read, create a static function **static void sensor_configure_interval(uint32_t interval)** that accepts the time interval value as parameter and configures the same as a sensor acquisition interval.

```
static void sensor_configure_interval(uint32_t interval)
{
    sm_handle handle;
    uint16_t index;
    sm_get_sensor_handle(SENSOR_ANY_TYPE, &handle, &index);
    sm_set_sensor_attribute(handle, SM_ACQUISITION_INTERVAL, interval);
}
```

Code Block 17 sensor_thread_entry.c

After this is added, modify the existing **void sensor_thread_entry(void *pvParameters)** by calling the above function with the required time interval value. This way, the sensor manager can be configured with the time interval between sensor reads. Here the function is called with 6000, which means the sensor acquisition interval is 6 seconds. The tested maximum supported value is 8 seconds.

```
/* Sensor entry function */
/* pvParameters contains TaskHandle_t */
void sensor_thread_entry(void *pvParameters) {
   FSP_PARAMETER_NOT_USED (pvParameters);
    sensor_configure_interval(6000);
   /* TODO: add your own code here */
   sm_init();
   while (1) {
      sm_run();
   }
}
```

Code Block 18 sensor_thread_entry.c

Outcome for this exercise:

After flashing the .srec file, connect the board with QC Sandbox mobile application over BLE and navigate to the HS4001 page. There, the data update should be according to the configured time interval only. *Note*: The mobile application has separate update interval. The observation is linked with that parameter and, that is, if the sensor update interval is configured for a lesser value than mobile app update, the user does not see any visible change after this exercise.

10.3.4.3 Add Widget for Sensor Value Visualization (Gauges / Box)

This exercise is on adding extra widget on the mobile application for the sensor data visualization. Default configuration for the mobile application is box widget in which the sensor data will be displayed. Additional configuration must be performed on the application to make the gauge widget available on the mobile application.

Complete the following steps:

- 1. Generate the **Sensor Data over BLE** application on QCS using the BGK-RA6E2, HS4001 Sensor, and DA14531 BLE module for FreeRTOS.
- 2. Modify the **gui_cfg.json** file as previously mentioned.
- 3. Rebuild the project by pressing the **Build QCStudio Project** option in QCS.
- 4. Download the *.srec file from the Debug/ path and flash the BGK-RA6E2 board.
- 5. Pair the QC Sandbox mobile application with the board and verify the data.

Files modified:

/gui_cfg.json

Modifications:

```
{
    "gui": {
    "id": "0000",
    "title": "US000 QC Example",
    "version:": "0.0.1",
    "tab": [
    {
        "id": "0001",
"name": "LED",
        "toggle": {
            "id": "0100",
            "type": "uint8_t",
            "name": "LED 1",
            "default": 0
        }
    },
    {
    "id": "0002",
    "name": "HS4001",
    "parameters": {
        "id": "0200",
        "type": "blob",
        "access": "r",
        "list": [
        {
            "id": "0201",
            "name": "TEMPERATURE",
            "units": "C",
            "type": "float",
            "access": "r",
            "auto_read": 1,
            "value": null
        },
        {
            "id": "0202",
            "name": "HUMIDITY",
            "units": "%",
            "type": "float",
            "access": "r",
            "auto_read": 1,
            "value": null
        }
        1
        },
        "gauge": [
        {
            "id": "0201",
            "name": "TEMPERATURE",
            "units": "C",
            "type": "float",
            "minimum": 0,
            "maximum": 100,
            "showinmax": true,
            "default": 0,
            "value": null
        },
{
            "id": "0202",
            "name": "HUMIDITY",
            "units": "%",
            "type": "float",
            "minimum": 0,
            "maximum": 100,
            "showinmax": true,
            "default": 0,
            "value": null
```

```
}
           j
     },
{
           "id": "0003",
           "name": "About",
           "parameters": {
                  "id": "0300",
"type": "blob",
"access": "r",
                  "list": [
                  {
                        "id": "0301",
                        "name": "Test Version",
                        "type": "string",
"access": "const",
                        "value": null
                  }
]
     }
}
     ]
}
}
```

Code Block 19 gui_cfg.json

Outcome for this exercise:

After this change, the QC Sandbox mobile application should show gauge widget as well for the sensor data.



10.3.4.4 Update Frequency of Sensor Data Update on the BLE Mobile Application

This exercise is to update the time interval at which data is read by the QC Sandbox mobile application. This is independent of the time interval at which the sensor data is read by the firmware (discussed on REQ - 2); that is, this exercise defines the rate at which function **handle_read_data()** is invoked inside the application.

Complete the following steps:

- 1. Generate Sensor Data over BLE application on QCS using BGK-RA6E2, HS4001 Sensor and DA14531 BLE module for FreeRTOS.
- 2. Modify the **gui_cfg.json** file as mentioned above.
- 3. Rebuild the project by pressing the **Build QCStudio Project** option in QCS.
- 4. Download the ***.srec** file from Debug/ path and flash the BGK-RA6E2 board.
- 5. Pair QC Sandbox mobile application with the board and verify the data.

Files to be modified:

/ gui_cfg.json

Modifications:

The auto_read parameter in the **gui_cfg.json** file mentions the time interval (in seconds) at which the request comes from mobile application. Value of 1 (Default) in this field indicates for every one second, the request is made, and the sensor data is collected.

```
{
    "gui": {
        "id": "0000",
        "title": "US000 QC Example",
        "version:": "0.0.1",
        "tab": [
        {
             "id": "0001",
"name": "LED",
             "toggle": {
                 "id": "0100",
                 "type": "uint8_t",
                 "name": "LED 1",
                 "default": 0
             }
        },
{
             "id": "0002",
             "name": "HS4001",
             "parameters": {
                 "id": "0200",
                 "type": "blob",
                 "access": "r",
                 "list": [
                 {
                      "id": "0201",
                      "name": "TEMPERATURE",
                      "units": "C",
                      "type": "float",
                      "access": "r",
                      "auto_read": 5,
                      "value": null
                 },
{
                      "id": "0202",
                      "name": "HUMIDITY",
"units": "%",
                      "type": "float",
                      "access": "r",
                      "auto_read": 5,
                      "value": null
                 }
                 1
             }
        },
{
             "id": "0003",
             "name": "About",
             "parameters": {
                 "id": "0300",
                 "type": "blob",
                 "access": "r",
                 "list": [
                 {
                      "id": "0301",
                      "name": "Test Version",
                      "type": "string",
                      "access": "const",
                      "value": null
                 }
                 ]
             }
        }
]
    }
}
```

Code Block 20 gui_cfg.json

Here it is configured for a 5s interval and the mobile application is expected to send the read request for sensor data every 5s. After everything is done, the user should see the data update for every 5s inside the mobile application.

Outcome for this exercise:

After flashing the **.srec** file, connect the board with QC Sandbox mobile application over BLE and navigate to the HS4001 page. The user should see the data update according to the configured time interval only.

18:31 🐥 🕓 🗢 🖲 🛎	0 \$ 2 %	1 63	18:32 🐥 🖸 🛊 🖗 📇	0 ≮ ≊ '‰i ©®
< REN	IESAS	i	< REN	IESAS i
LED	S4001 About	D	LED	S4001 About
TEMPERATURE	27.28000068¢	C	TEMPERATURE	27.270000457 C
HUMIDITY	61.990001678	%	HUMIDITY	62.240001678 %
æ	Read All		٩	Read All

10.3.4.5 Send the Sensor Value Only if the Read Button Pressed on the BLE Mobile Application

This requirement is to disable the auto_read functionality of the QC Sandbox mobile application. The user is expected to initiate the read process from mobile application to get the updated data.

Complete the following steps:

- 1. Generate Sensor Data over BLE application on QCS using the BGK-RA6E2, HS4001 Sensor, and DA14531 BLE module for FreeRTOS.
- 2. Modify the **gui_cfg.json** file as mentioned above.
- 3. Rebuild the project by pressing the Build QCStudio Project option in QCS.
- 4. Download the ***.srec** file from **Debug/** path and flash the BGK-RA6E2 board.
- 5. Pair the QC Sandbox mobile application with the board and verify the data.

Files to be modified:

/ gui_cfg.json

Modifications:

The auto_read parameter in the **gui_cfg.json** file mentions the time interval (in seconds) at which the request comes from the mobile application. Configuring this value to 0 disables the auto read functionality completely and the user must initiate the data read manually.

```
{
    "gui": {
        "id": "0000",
        "title": "US000 QC Example",
        "version:": "0.0.1",
        "tab": [
        {
             "id": "0001",
"name": "LED",
             "toggle": {
                 "id": "0100",
                 "type": "uint8_t",
                 "name": "LED 1",
                 "default": 0
             }
        },
{
             "id": "0002",
             "name": "HS4001",
             "parameters": {
                 "id": "0200",
                 "type": "blob",
                 "access": "r",
                 "list": [
                 {
                      "id": "0201",
                      "name": "TEMPERATURE",
                      "units": "C",
                      "type": "float",
                      "access": "r",
                      "auto_read": 0,
                      "value": null
                 },
{
                      "id": "0202",
                      "name": "HUMIDITY",
"units": "%",
                      "type": "float",
                      "access": "r",
                      "auto_read": 0,
                      "value": null
                 }
                 1
             }
        },
{
             "id": "0003",
             "name": "About",
             "parameters": {
                 "id": "0300",
                 "type": "blob",
                 "access": "r",
                 "list": [
                 {
                      "id": "0301",
                      "name": "Test Version",
                      "type": "string",
                      "access": "const",
                      "value": null
                 }
                 ]
             }
        }
]
    }
}
```

Code Block 21 gui_cfg.json

Outcome for this exercise:

After flashing the **.srec** file, connect the board with QC Sandbox mobile application over BLE and navigate to the HS4001 page. The user should not see the automatic update for the sensor data. The sensor data should get updated only after pressing the **Read All** button.

10.3.4.6 Using the BLE Mobile Application, Toggle the LED which is Currently ON and OFF

This exercise is to update the onboard LED response for the mobile application action. After this exercise, the LED should toggle for each LED button press on the mobile application. Currently, the button is a simple slider button that can be either at OFF position or ON position. This needs to be modified into a simple button so that with each press, the LED toggle can be visible.

Complete the following steps:

- 1. Generate the **Sensor Data over BLE** application on QCS using BGK-RA6E2, HS4001 Sensor, and DA14531 BLE module for FreeRTOS.
- 2. Modify the gui_cfg.json and ble_app.c files as mentioned above.
- 3. Rebuild the project by pressing the Build QCStudio Project option in QCS.
- 4. Download the *.srec file from Debug/ path and flash the BGK-RA6E2 board.
- 5. Pair the QC Sandbox mobile application with the board and verify the data.

Files to be modified:

- / src / qe_gen / ble / ble_app.c
- /gui_cfg.json

Modifications:

ble_app.c: The handle_write_led() function is responsible for defining the LED state based on requests from the mobile application and data received through BLE. The default code turns OFF the LED if the data received over BLE is 0x00; otherwise, it turns the LED on. This function should be modified to toggle the LED state whenever the handler is invoked.

```
static void handle_write_led(uint16_t id, uint8_t const * const data)
{
    FSP_PARAMETER_NOT_USED(id);
    static bool state = false;
    state = !state;
    if (state == false)
    {
        utils_set_LED(BLUE_LED,BSP_IO_LEVEL_LOW);
    }
    else
    {
        utils_set_LED(BLUE_LED,BSP_IO_LEVEL_HIGH);
    }
}
```

Code Block 22 ble_app.c

gui_cfg.json: This file should be modified to render button for the LED instead of ON and OFF switch.

```
{
    "gui":
    {
        "id": "0000",
        "title": "US000 QC Example",
        "version:": "0.0.1",
        "tab":
        [
             {
                 "id": "0001",
                 "name": "LED",
                 "button":
                 {
                     "id": "0100",
                     "name": "LED",
                     "ack": false,
                     "timeout": 0,
                     "read": false
                 }
            },
{
                 "id": "0002",
                 "name": "HS4001",
                 "parameters":
                 {
                     "id": "0200",
                     "type": "blob",
                     "access": "r",
                     "list":
                     [
                          {
                              "id": "0201",
                              "name": "TEMPERATURE",
                              "units": "C",
                              "type": "float",
                              "access": "r",
                              "auto_read": 1,
                              "value": null
                          },
                          {
                              "id": "0202",
                              "name": "HUMIDITY",
"units": "%",
                              "type": "float",
                              "access": "r",
                              "auto_read": 1,
                              "value": null
                         }
                     ]
                 }
            },
{
                 "id": "0003",
                 "name": "About",
                 "parameters":
                 {
                     "id": "0300",
                     "type": "blob",
                     "access": "r",
                     "list":
                     [
                          {
                              "id": "0301",
                              "name": "Test Version",
                              "type": "string",
                              "access": "const",
```


Code Block 23 gui_cfg.json

Outcome for this exercise:

The modifications made to **gui_cfg.json** file render buttons on the LED page of QC sandbox application. When ***.srec** is flashed on the BGK-RA6E2 and paired with the QC Sandbox mobile application, on the LED page, button is rendered as follows.



Press this button and the user can see the on-board LED (LED2) toggling for each button press.

11. References

- <u>RA6E2 Entry-Line 200MHz Arm® Cortex®-M33 General Purpose Microcontroller | Renesas</u>
- DA16600MOD Ultra-Low Power Wi-Fi + Bluetooth® Low Energy Combo Modules for Battery Powered IoT Devices | Renesas
- HS4001 Relative Humidity and Temperature Sensor, Digital Output, ±1.5% RH | Renesas
- US159-DA14531EVZ Low Power Bluetooth PMOD Board (Renesas QuickConnect IoT)

Technical Updates/Technical News

- The latest information can be downloaded from the Renesas Electronics Website.

Website and Support

Renesas Electronics Website - <u>https://www.renesas.com/</u> Inquiries - <u>https://www.renesas.com/contact/</u>

12. Ordering Information

Part Number	Description	
QC-BEKITPOC2Z	QuickConnect Beginners Kit V2.0	

13. Revision History

Revision	Date	Description
1.00	May 8, 2025	Initial release.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES ("RENESAS") PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit <u>www.renesas.com/contact-us/</u>.