Renesas Synergy™ Platform

External IRQ HAL Module Guide

R11AN0089EU0101
Rev.1.01
Aug 23, 2017

This module guide will enable you to effectively use a module in your own design. Upon completion of this guide, you will be able to add this module to your own design, configure it correctly for the target application and write code using the included application project code as a reference and efficient starting point. References to more detailed API descriptions and suggestions of other application projects that illustrate more advanced uses of the module are available in the Renesas Synergy Knowledge Base (as described in the References section at the end of this document), and should be valuable resources for creating more complex designs.

The External IRQ HAL module is an API for configuring and using external IRQ pins on Synergy MCUs. The External IRQ HAL module is implemented on r_icu and uses the Interrupt Controller Unit (ICU) of the Synergy MCU.

Contents

1. External IRQ HAL Module Features 2
2. External IRQ HAL Module APIs Overview 2
3. External IRQ HAL Module Operational Overview 3
3.1 External IRQ HAL Module Important Operational Notes and Limitations 3
3.1.1 External IRQ HAL Module Operational Notes 3
3.1.2 External IRQ HAL Module Limitations 4
4. Including the External IRQ HAL Module in an Application 4
5. Configuring the External IRQ HAL Module 4
5.1 External IRQ HAL Module Clock Configuration 5
5.2 External IRQ HAL Module Pin Configuration 5
6. Using the External IRQ HAL Module in an Application 6
7. The External IRQ HAL Module Application Project 7
8. Customizing the External IRQ HAL Module for a Target Application 10
9. Running the External IRQ HAL Module Application Project 10
10. External IRQ HAL Module Conclusion 10
11. External IRQ HAL Module Next Steps 11
12. External IRQ HAL Module Reference Information 11

1. External IRQ HAL Module Features

- Supports the external interrupt pins available on the target Synergy MCU
- Supports multiple function options:
  - Enabling and disabling generation of an interrupt
  - Enabling and disabling the IRQ noise filter
  - Setting external pin IRQ trigger (Rising edge, falling edge or low level on the IRQ pin)
- Supports configuring a user callback function, which will be invoked by the HAL module when an external pin interrupt is generated.

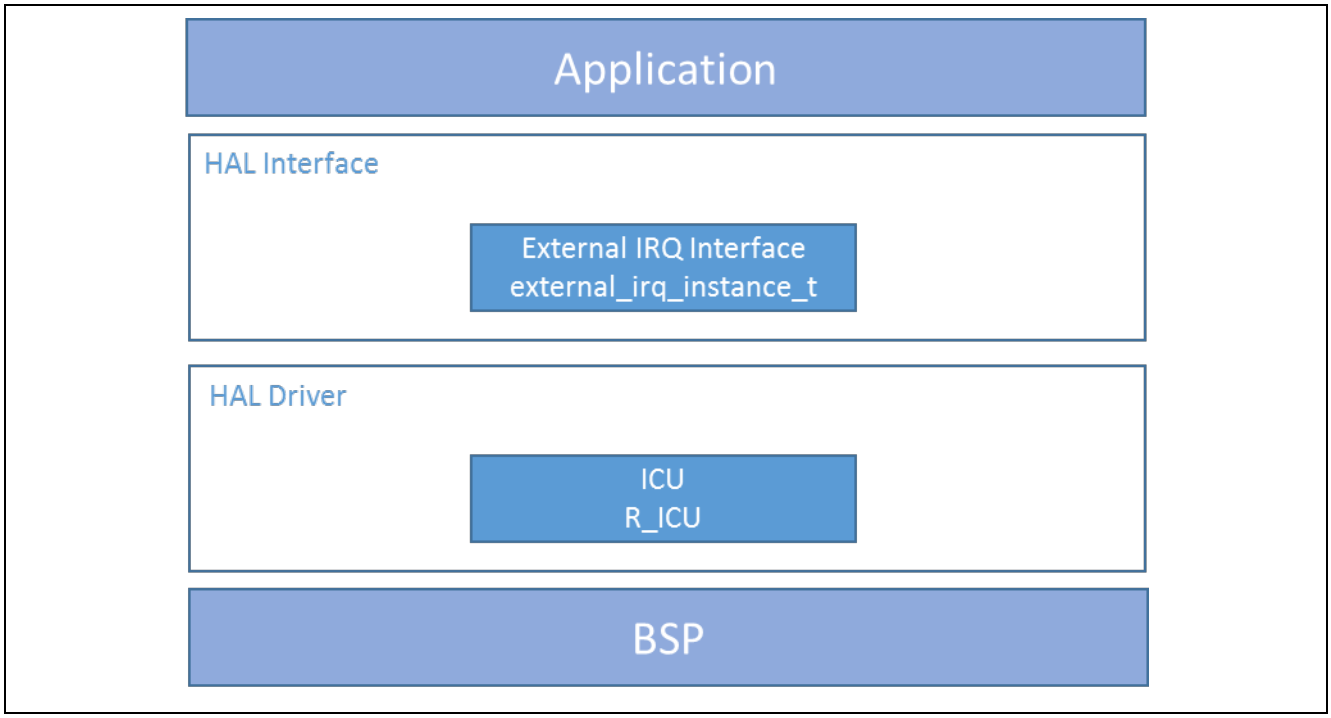


Figure 1 External IRQ HAL Module Block Diagram

2. External IRQ HAL Module APIs Overview

The External IRQ HAL module defines APIs for opening, closing, and waiting for interrupt events from external pins. A complete list of the available APIs, an example API call and a short description of each can be found in the following table. A table of status return values follows the API summary table.

Table 1 External IRQ HAL Module API Summary

Function Name	Example API Call and Description
.open	`g_external_irq.p_api->open(g_external_irq.p_ctrl, g_external_irq.p_cfg)` Open instance and initialize.
.enable	`g_external_irq.p_api->enable(g_external_irq.p_ctrl)` Enable callback when IRQ occurs.
.disable	`g_external_irq.p_api->disable(g_external_irq.p_ctrl)` Disable callback when IRQ occurs.
.triggerSet	`g_external_irq.p_api->triggerSet(g_external_irq.p_ctrl, trigger)` Set trigger.

.filterEnable	`g_external_irq.p_api->filterEnable(g_external_irq.p_ctrl)` Enable noise filter.
.filterDisable	`g_external_irq.p_api->filterDisable(g_external_irq.p_ctrl)` Disable noise filter.
.close	`g_external_irq.p_api->close(g_external_irq.p_ctrl);` Close instance.
.versionGet	`g_external_irq.p_api->wait(&version);` Retrieve the API version with the version pointer.

Note: For more complete descriptions of operation and definitions for the function data structures, typedefs, defines, API data, API structures and function variables, review the SSP User's Manual API References for the associated module.

Table 2 Status Return Values

Name	Description
SSP_SUCCESS	Function successful.
SSP_ERR_ASSERTION	Assertion error.
SSP_ERR_INVALID_ARGUMENT	Callback is not NULL but ISR is not enabled.
SSP_ERR_IN_USE	Device in use.
SSP_ERR_NOT_OPEN	Device unopened.

Note: Lower-level drivers may return common error codes. Refer to the SSP User's Manual API references for the associated module for a definition of all relevant status return values.

3. External IRQ HAL Module Operational Overview

The External IRQ HAL module provides a set of APIs for controlling external interrupts. Interrupts can be triggered on rising edge, falling edge, both edges or low level of the input signal on the external IRQ pin. A digital-filtering function can be enabled to eliminate some noise on the input signal. A user-callback function is supported and is triggered each time an IRQ event occurs.

To trigger a transfer of data using the DMAC or DTC peripheral when the configured external IRQ event occurs, configure the DMAC or DTC transfer with the activation source set to `ELC_EVENT_PORTn_IRQ` (where n is the IRQ channel number.)

Other peripherals can be triggered to start from an external interrupt using the Event Link Controller (ELC.) Refer to the SSP User Manual User Guide for the ELC HAL module for more information.

3.1 External IRQ HAL Module Important Operational Notes and Limitations

3.1.1 External IRQ HAL Module Operational Notes

- Refer to the datasheet for the Synergy device to be programmed to find the port pins which support the external interrupt functions and to obtain the external IRQ number for a given port pin.
- The external IRQ number corresponds to the channel setting in the ISDE Properties window for the External IRQ HAL module.
- The PORTn (where n is the IRQ number) interrupt must be enabled in the BSP to notify the module that the anticipated hardware event has occurred.
- A user-callback function can be registered in the `open` API. If this callback function is provided, it will be called from the interrupt service routine (ISR) each time the IRQn triggers. NOTE: Since the callback is called from an ISR, care should be taken not to use blocking calls or lengthy processing. Spending excessive time in an ISR can adversely affect the responsiveness of the system.

3.1.2 External IRQ HAL Module Limitations

- Refer to the most recent SSP Release Notes for any additional operational limitations for this module.

4. Including the External IRQ HAL Module in an Application

This section describes how to include the External IRQ HAL module in an application using the SSP configurator.

Note: This section assumes you are familiar with creating a project, adding threads, adding a stack to a thread and configuring a block within the stack. If you are unfamiliar with any of these items, refer to the first few chapters of the SSP User's Manual to learn how to manage each of these important steps in creating SSP-based applications.

To add the External IRQ HAL driver to an application, simply add it to a thread using the stacks selection sequence given in the following table. (The default name for the module is g_external_irq0. This name can be changed in the associated Properties window.)

Table 3 External IRQ HAL Driver Selection Sequence

Resource	ISDE Tab	Stacks Selection Sequence
r_icu0 External IRQ Driver on r_icu	Threads	New Stack> Driver> Input> External IRQ Driver on r_icu

When the External IRQ HAL module on r_icu is added to the thread stack as shown in the following figure, the configurator automatically adds any needed lower-level modules. Any drivers that need additional configuration information will be box text highlighted in Red. Modules with a Gray band are individual modules that stand alone.

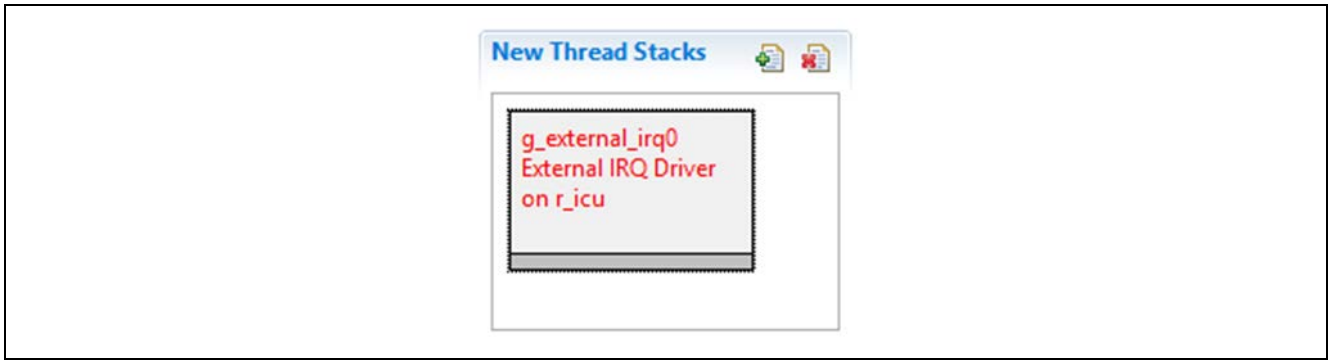


Figure 2 External IRQ HAL Module Stack

5. Configuring the External IRQ HAL Module

The External IRQ HAL module must be configured by the user for the desired operation. The SSP configuration window will automatically identify (by highlighting the block in red) any required configuration selections, such as interrupts or operating modes, which must be configured for lower-level modules for successful operation. Furthermore, only those properties that can be changed without causing conflicts are available for modification. Other properties are 'locked' and are not available for changes, and are identified with a lock icon for the 'locked' property in the property window in the ISDE. This approach simplifies the configuration process and makes it much less error prone than previous 'manual' approaches to configuration. The available configuration settings and defaults for all the user-accessible properties are given in the Properties tab within the SSP configurator and are shown in the following tables for easy reference.

One of the properties most often identified as requiring a change is the interrupt priority; this configuration setting is available within the Properties window of the associated module. Simply select the indicated module and then view the Properties window; the interrupt settings are often toward the bottom of the properties list, so scroll down until they become available. Also, note that the interrupt priorities listed in the Properties window in the ISDE will include an indication as to the validity of the setting based on the targeted MCU (CM4 or CM0+). This level of detail is not included in the following configuration properties tables, but is easily visible with the ISDE when configuring interrupt-priority levels.

Note: You may want to open your ISDE, create the module and explore the property settings in parallel with looking over the configuration table settings provided in the following table. This will help orient you and can be a useful 'hands-on' approach to learning the ins and outs of developing with SSP.

Table 4 Configuration Settings for the External IRQ HAL Module on r_icu

ISDE Property	Value	Description
Parameter Checking	Default, Enabled, Disabled (Default: Default)	Parameter Checking
Name	g_external_irq0	Driver name.
Channel	0	Specifies the hardware IRQ channel used.
Trigger condition	Falling, Rising, Both Edges, Low Level (Default: Rising)	Trigger selection.
Digital Filtering	Enabled, Disabled (Default: Disabled)	Digital filter enable/disable.
Digital Filtering Sample Clock (Only valid when Digital Filtering is Enabled)	PCLK/1, PLCK/8, PLCK/32, PCLK/64 (Default: PCKL/64)	Sets noise filter sampling period.
Interrupt enabled after initialization	True, False (Default: True)	Interrupt enable selection.
Callback	NULL	A user callback function can be registered using the open API. If this callback function is provided, it is called from the interrupt service routine (ISR) each time the IRQn triggers. Note: Since the callback is called from an ISR, care should be taken not to use blocking calls or lengthy processing. Spending excessive time in an ISR can affect the responsiveness of the system.
Interrupt Priority	Enabled, Disabled (Default: Disabled)	Interrupt priority setting

Note: The example values and defaults are for a project using the Synergy S7G2. Other MCUs may have different default values and available configuration settings.

5.1 External IRQ HAL Module Clock Configuration

The IRQ peripheral module doesn't require any specific clock settings.

5.2 External IRQ HAL Module Pin Configuration

The External IRQ peripheral module uses pins on the MCU to communicate to external devices. I/O pins must be selected and configured as required by the external device. The following table illustrates the method for selecting the pins within the SSP configuration window and the subsequent table illustrates an example selection for the IRQ pins.

Table 5: Selection Sequence for the External IRQ HAL Driver on r_icu

Resource	ISDE Tab	Pin selection Sequence
IRQ	Pins	Select Peripherals > Input: IRQ > IRQ0

Note: The selection sequence assumes IRQ0 is the desired hardware target for the driver.

Table 6: Pin Configuration Settings for External IRQ Driver on r_icu

Pin Configuration Property	Value	Description
Operation Mode	Disabled, Enabled (Default: Disabled)	Select Enabled to enable interrupts
NMI	None, P200 (Default: None)	Non-maskable interrupt Pin
IRQ00:14	None, Pnn, Pmm (Default: None)	Interrupt request Pin

Note: The example values are for a project using the Synergy S7G2 and the SK-S7G2 Kit. Other Synergy Kits and other Synergy MCUs may have different available pin configuration settings.

6. Using the External IRQ HAL Module in an Application

The typical steps in using the External IRQ HAL module in an application are:

1. Initialize the External IRQ HAL module using the external_irq_api_t::`open` API
2. Enable the IRQ (if needed) with the externalirq_api_t::`enable` API
3. Enable the noise filter (if needed) with the external_irq_api_t::`filterEnable` API
4. Change the trigger condition (only if the module is closed previously to avoid any false events) with the external_irq_api_t::`triggerSet` API
5. Disable the noise filter (if enabled) with external_irq_api_t::`filterDisable` API
6. Close the module (if needed) with the external_irq_api_t::`close` API

These common steps are illustrated in a typical operational flow diagram in the following figure:

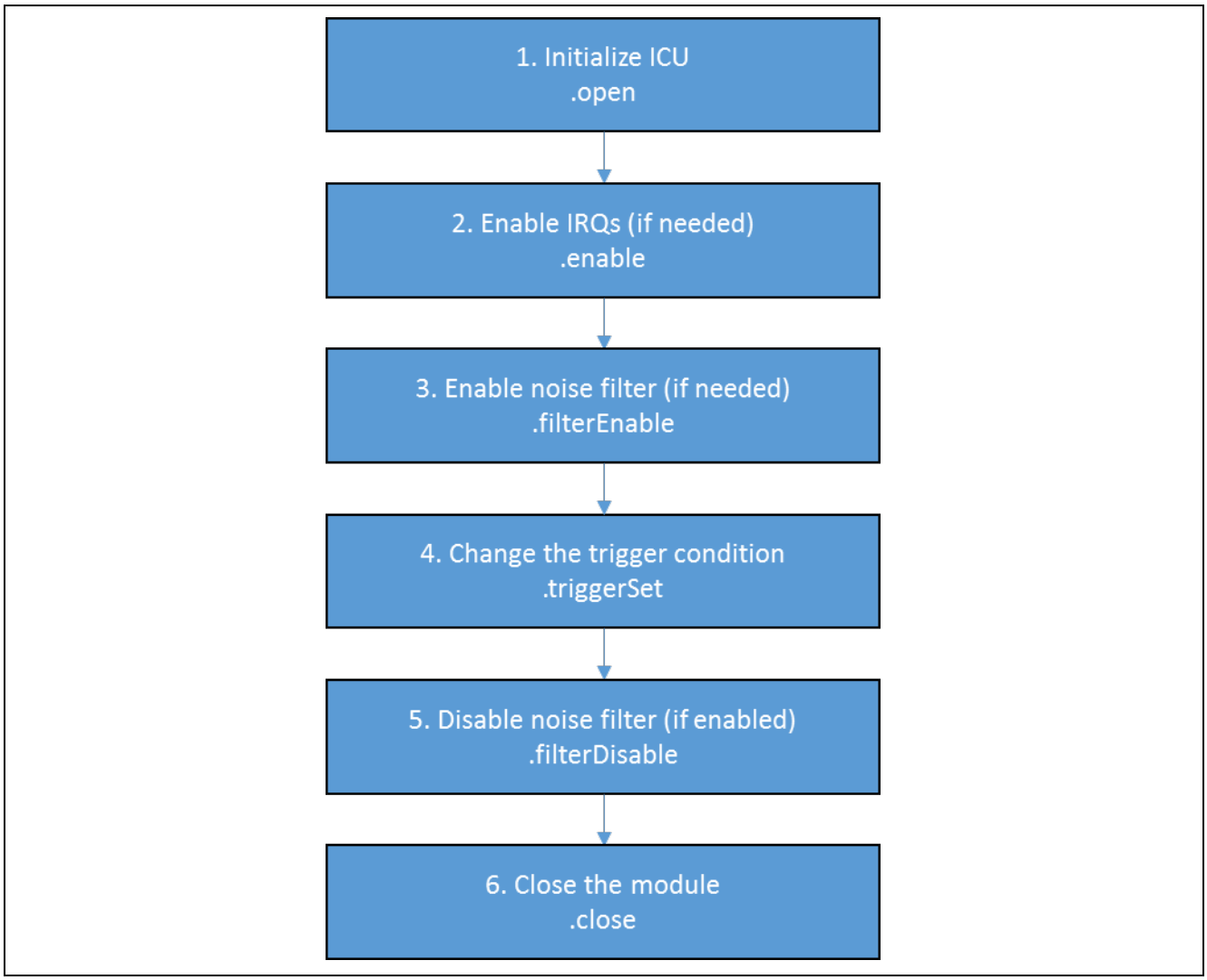


Figure 3 Typical External IRQ HAL Module Application Flow Chart

7. The External IRQ HAL Module Application Project

The application project associated with this module guide demonstrates the steps in an example application. You may want to import and open the application project with the ISDE and view the configuration settings for the IRQ HAL modules. You can also read over the code in `extirq_hal_module_guide_project.c` used to illustrate the IRQ HAL module APIs in a complete design.

The application project demonstrates the typical use of the IRQ HAL module APIs. The application project initializes two instances of IRQ HAL modules, one for IRQ10 (push button S5 of the SK-S7G2) and IRQ11 (push button S4 of the SK-S7G2). IRQ10 is configured to detect falling edge events on the IRQ pin and to be enabled when the module is opened. Additionally, noise filtering is enabled and a callback function is specified.

IRQ11 is also configured to detect falling edge events on the IRQ pin but is set to be disabled when the module is opened. Additionally, noise filtering is disabled and a callback function is specified.

When a falling edge on IRQ10 occurs, the callback function is called. In this callback the Green LED is toggled and IRQ11 is either enabled or disabled. Green LED ON means IRQ11 is enabled and when it is OFF, IRQ11 is disabled.

When enabled, a falling edge on IRQ11 will result in the callback function being called. The Amber LED will turned on and the IRQ11 will be reconfigured to detect rising edges. When a rising edge on IRQ11 occurs, the callback function is called. The Red LED will turned on and the IRQ11 will be reconfigured to falling rising edges.

Therefore, when IRQ11 is enabled (Green LED on), pressing and holding S4 the Amber LED will be on. When released, the Red LED will be on. However, disabling IRQ11 whilst the Amber LED is on, (Green LED off), on releasing S4 the Amber LED will remain on. Pressing S4 will have no effect until it is enabled by pressing S5.

The state of the IRQs is also reported to the debug console via printf statements.

Table 7 Software and Hardware Resources Used by the Application Project

Resource	Revision	Description
e² studio	5.3.1 or later	Integrated Solution Development Environment
SSP	1.2.0 or later	Synergy Software Platform
IAR EW for Renesas Synergy	7.71.2 or later	IAR Embedded Workbench for Renesas Synergy
SSC	5.3.1 or later	Synergy Standalone Configurator
SK-S7G2	v3.0 to v3.1	Starter Kit

A simple flow diagram of the application project is given in the following figure:

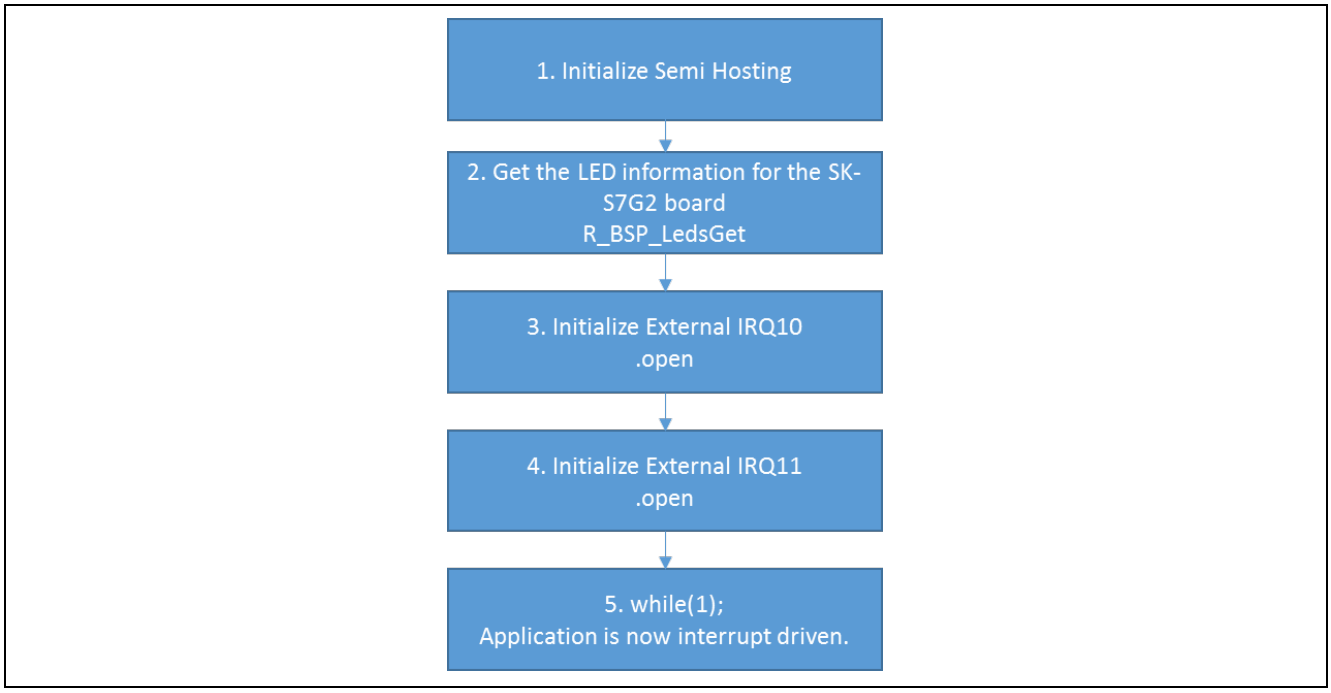


Figure 4 External IRQ HAL Module Application Project Flow Diagram

The `extirq_hal_module_guide_project.c` file is located in the project once is has been imported into the ISDE. You can open the file within the ISDE and follow along with the description to identify key uses of the APIs.

The first section of `extirq_hal_module_guide_project.c` includes the header files for the project. `stdio.h` is required for the usage of Semi Hosting printf to the debug console. `hal_data.h` is a system created file that has include files to all of the other modules of the Synergy project.

Following this are declarations for enabling semi-hosting and a debug trap function for reporting if an API returns an error code other than SSP_SUCCESS.

Following this is a user created enum that is used to identify which IRQ edge is going to be detected. This is used in the IRQ11 callback.

Next is the module guide project entry point, `void ExtIRQ_hal_module_guide_project(void)`.

This function starts by, if required, by calling a function to enable semi hosting printf. Note that this is only called if using the GCC compiler and if the debugger is connected.

Then a function to determine the number of LEDs the development board has is called.

Then the 2 external IRQ instances are opened. If the open call is unsuccessful, the error trap function will catch this and execution of the project will send a message to the debug console and wait in a while(1) loop.

Once the external IRQ instances are opened, the project enters an empty while(1) loop. This is as all further processing is performed in the IRQ callback functions.

The ext_irq10_callback, which is called when S5 is pressed, is next. The first section toggles the state of a static variable from 1 to 0, or 0 to 1. This is used next to set the state of LED[0], the green LED.

Next, if the LED is off, then the IRQ11 is disabled.

If the LED is on, then the IRQ11 is enabled. However, prior to enabling the IRQ11, the IRQ11 interrupt status bit is cleared. This is because when the IRQ11 instance is opened, even though the interrupt is disabled, an edge on the IRQ pin will cause the IR bit in the ICU (interrupt control unit) to be set and held pending. Enabling the IRQ would cause this IRQ to be taken. This type of behavior of taking pending interrupts may not be acceptable, so the code clears any pending interrupts prior to enabling the IRQ11.

Any errors in the APIs will be caught be the error trap function.

The occurrence of IRQ10 and the enabling / disabling of IRQ11 is output to the debug console.

The ext_irq11_callback, which is called when S4 is pressed, is next.

At reset IRQ11 is configured to detect a falling edge interrupt. A static variable is declared to be waiting for a falling edge. When a falling edge IRQ occurs the static variable is changed to be waiting for a rising edge and the LEDs are set to amber LED on, red LED off and the IRQ trigger is set to detect rising edges.

When a rising edge IRQ occurs the static variable is changed to be waiting for a falling edge and the LEDs are set to amber LED off, red LED on and the IRQ trigger is set to detect falling edges.

If any errors in the APIs will be caught be the error trap function.

The occurrence of IRQ11 is output to the debug console.

Note: The previous description assumes you are familiar with using `printf()` with the Debug Console in the Synergy Software Package. If you are unfamiliar with this, refer to the "*How do I Use Printf() with the Debug Console in the Synergy Software Package*" Knowledge Base article, available as described in the References section at the end of this document. Alternatively, the user can see results via the watch variables in the debug mode.

A few key properties are configured in this application project to support the required operations and the physical properties of the target board and MCU. The properties with the values set for this specific project are listed in the following table. You can also open the application project and view these settings in the Properties window as a hands-on exercise.

Table 8: External IRQ HAL Module Configuration Settings for the Application Project

Resource	ISDE Property	Value Set
g_external_irq10 External IRQ Driver on r_icu	Parameter Checking	Enabled
	Name	g_external_irq10
	Channel	10
	Trigger	Falling
	Digital Filtering	Enabled
	Digital Filtering Sample Clock	PCLK/64
	Interrupt Enable after Initialization	True
	Callback	ext_irq10_callback
	Interrupt Priority	Priority 2
Pins tab > Peripherals > Input:IRQ > Enabled	IRQ10	P005
g_external_irq10 External IRQ Driver on r_icu	Parameter Checking	Enabled
	Name	g_external_irq11
	Channel	11
	Trigger	Falling
	Digital Filtering	Disabled
	Digital Filtering Sample Clock	PCLK/64
	Interrupt Enable after Initialization	False
	Callback	ext_irq11_callback
	Interrupt Priority	Priority 2
Pins tab > Peripherals > Input:IRQ > Enabled	IRQ11	P006

8. Customizing the External IRQ HAL Module for a Target Application

Some configuration settings will normally be changed by the developer from those shown in the application project. For example, the user can easily change the configuration settings for the External IRQ Digital Filtering Sample clock by updating the PCLKC in the Clocks tab. The user can also change the external IRQ port pins to select the desired pin as an interrupt trigger. This can be done using the "Pins" tab in the configurator. The External IRQs used in this application project are channel 10 & channel 11 which are connected to the SK-S7G2 User Pushbuttons Button S5 and S4.

9. Running the External IRQ HAL Module Application Project

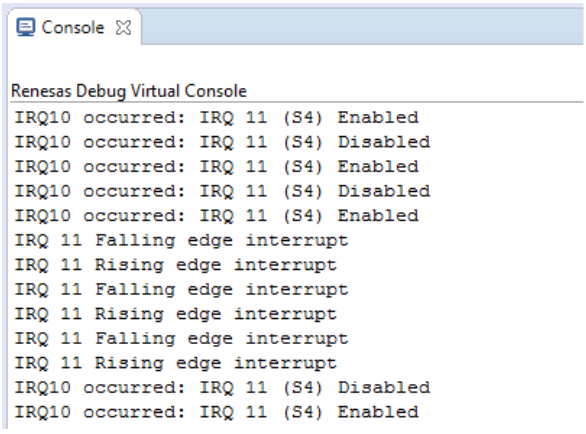
To run the External IRQ HAL module guide project and to see it executed on a target kit, you can simply import it into your ISDE, compile and run debug. Refer to the Synergy Project Import Guide (r11an0023eu0116-synergy-ssp-import-guide.pdf, included in this package) for instructions on importing the project into e^2 studio or IAR embedded workbench and building/running the application.

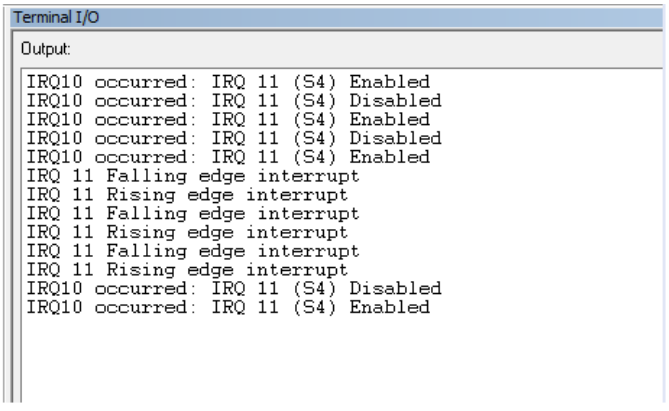
To implement the External IRQ HAL module guide project in a new Synergy project, follow the steps for defining, configuring, auto-generating files, adding code, compiling and debugging on the target kit. Following these steps is a hands-on approach that can help make the development process with SSP more practical, while just reading over this guide will tend to be more theoretical.

Note: The following steps are described in sufficient detail for someone experienced with the basic flow through the Synergy development process. If these steps are not familiar, refer to the first few chapters of the SSP User's Manual for a description of how to accomplish these steps.

To create and run the External IRQ application project, simply follow these steps:

1. Create a new Renesas Synergy project for the SK-S7G2 called ExtIRQ_HAL_MG_AP.
2. Select the **Threads** tab
3. Add two instances of `External IRQ Driver on r_icu`, one called `g_external_irq10`, the other `g_external_irq11`.
4. Configure the IRQ instances and I/O ports as detailed in table 8.
5. Click the **Generate Project Content**
6. Add the code from the supplied files `hal_entry.c` and `extirq_hal_module_guide_project.c`.
7. Build the project
8. Connect to the USB micro cable at J19 on SK-S7G2 board and connect other end of USB cable to the Host.
9. Start to debug the application.
10. Pressing S5 will toggle the state of the Green LED. When ON, pressing S5 will change the state of the Red and Amber LEDs. When pressed and held, the Amber LED will be ON and when released the Red LED will be on.
11. If debugging, the state IRQs being generated is shown in the debug console (E2 Studio) / Terminal IO (IAR EW)





10. External IRQ HAL Module Conclusion

This module guide has provided all the background information needed to select, add, configure and use the module in an example project. Many of these steps were time consuming and error-prone activities in previous generations of embedded systems. The Renesas Synergy Platform makes these steps much less time consuming and removes the common errors, like conflicting configuration settings or the incorrect selection of lower-level drivers. The use of high-level APIs (as demonstrated in the application project) illustrates additional development time savings by allowing work

to begin at a high level and avoiding the time required in older development environments to use or, in some cases, create, lower-level drivers.

11. External IRQ HAL Module Next Steps

After you have mastered a simple External IRQ HAL module project, you may want to review a more complex example. You may find that the External IRQ Framework module is a better fit for your target application. The External IRQ Framework Module Guide illustrates the use of the external IRQ within a ThreadX-based implementation. This guide is available in the References section at the end of this document.

Other application projects and application notes that demonstrate External IRQ use are available in the References section at the end of this document.

12. External IRQ HAL Module Reference Information

SSP User Manual: Available in html format in the SSP distribution package and as a pdf from the Synergy Gallery.

Links to all the most up-to-date r_icu module reference materials and resources are available on the Synergy Knowledge Base: https://en-us.knowledgebase.renesas.com/English_Content/Renesas_Synergy%E2%84%A2_Platform/Renesas_Synergy_Knowledge_Base/R_ICU_Module_Guide_Resources.

Website and Support

Support: https://synergygallery.renesas.com/support

Technical Contact Details:

- America: https://www.renesas.com/en-us/support/contact.html
- Europe: https://www.renesas.com/en-eu/support/contact.html
- Japan: https://www.renesas.com/ja-jp/support/contact.html

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	May 15, 2017		Initial Release
1.01	Aug 23, 2017		Update to Hardware and Software Resources Table

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other disputes involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawing, chart, program, algorithm, application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics products.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (space and undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of failure or malfunction of Renesas Electronics products, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.3.0-1 November 2016)

RENESAS

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 6.0