

Introduction

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 Data Operation Circuit (DOC) HAL module is a high-level API for DOC applications and is implemented on `r_doc`. The DOC HAL module uses the DOC peripherals on the Renesas Synergy MCU device. A user-defined callback can be created to inform the CPU when an event occurs.

Contents

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

1. DOC HAL Module Features

The DOC HAL module peripheral is used to compare 16-bit data and can detect the following events:

- A match or mismatch between data values
- Overflow of an addition operation
- Underflow of a subtraction operation.

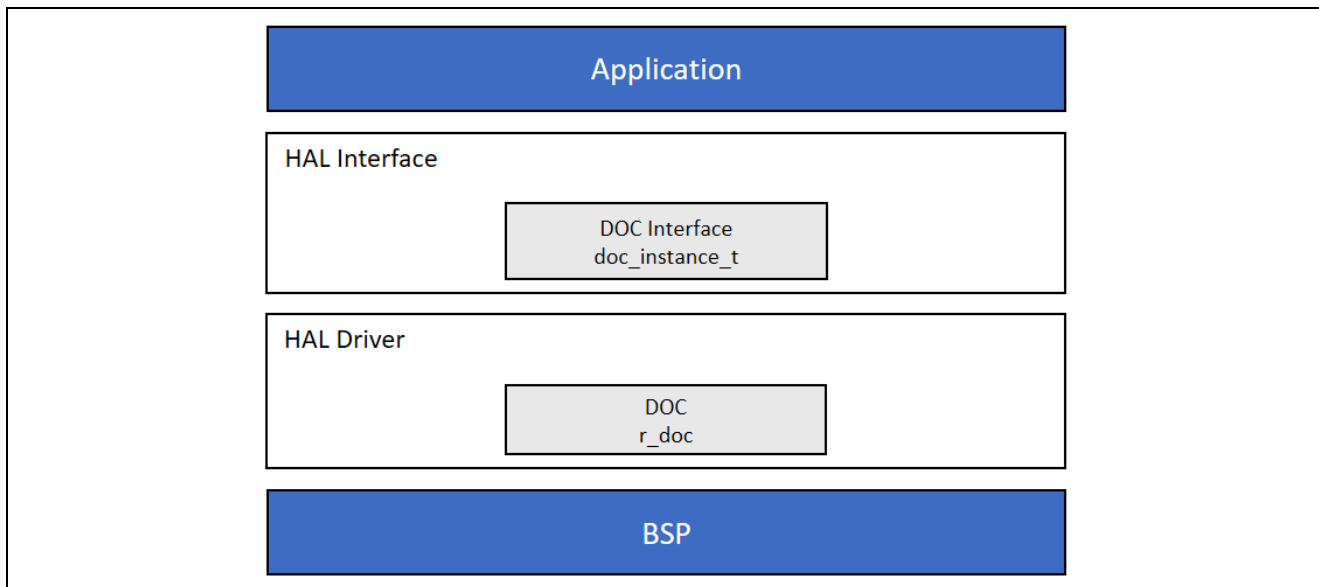


Figure 1 DOC HAL Module Block Diagram

2. DOC HAL Module APIs Overview

The DOC HAL module defines APIs for opening, closing, checking the status, and writing data to the data operation circuit. The DOC HAL module uses the DOC peripheral on the Synergy MCU. 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 DOC HAL Module API Summary

Function Name	Example API Call and Description
.open	g_doc.p_api->open(g_doc.p_ctrl, g_doc.p_cfg) Initial configuration.
.close	g_doc.p_api->close(g_doc.p_ctrl) Allow the driver to be reconfigured. Will reduce power consumption.
.statusGet	g_doc.p_api->statusGet(g_doc.p_ctrl, &my_Status) Get the DOC status and stores it in the provided pointer p_status.
.statusClear	g_doc.p_api->statusClear(g_doc.p_ctrl) Clear DOPCF status flag.
.write	g_doc.p_api->write(g_doc.p_ctrl, value) Write to the DODIR and DODSR registers.
.inputRegisterWrite	g_doc.p_api->inputRegisterWrite(g_doc.p_ctrl, &doc_values) Write to the DODIR register.
.versionGet	g_doc.p_api->versionGet(g_doc.p_ctrl, &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	DOC successfully configured.
SSP_ERR_IN_USE	Module already open.
SSP_ERR_ASSERTION	One or more pointers point to NULL.
SSP_ERR_INVALID_ARGUMENT	ISR is not enabled. Enable the ISR in <code>bsp_irq_cfg.h</code> .
SSP_ERR_HW_LOCKED	DOC resource is locked.
SSP_ERR_NOT_OPEN	Driver not open.

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. DOC HAL Module Operational Overview

The DOC HAL module controls the DOC peripheral on a Synergy MCU. It is used to compare 16-bit data and can detect a mismatch between data values, an overflow of an addition value or an underflow of a subtraction operation. If a callback is available and the associated interrupt is enabled, the callback function will be called in response to a DOC event.

The DOC uses two data registers to perform operations: the DOC Data Input Register (DOCDIR) holds the data to be operated on, and the DOC Data Setting Register (DOCDSR) holds the value that is used to operate on the input data. In addition and subtraction modes, this register stores the results of data operations. (Both registers are 16-bits wide.)

3.1 DOC HAL Module Important Operational Notes and Limitations

3.1.1 DOC HAL Module Operational Notes

The initial setting of comparison data is written to the DOC by calling the `write` API. The `write` API writes to the DOC DODSR and DODIR registers. The `write` API uses a variable of type `doc_data_t` as illustrated below:

```
doc_data_t g_doc_values;
    g_doc_values.dodir = 0x1000;
    g_doc_values.dodsr = 0x1000;
    g_doc.p_api->write(g_doc.p_ctrl, &g_doc_values)
```

If the data to be compared does not change, there is no need to re-write it each time a comparison is required. The input data value can be written to the DOC by using the `inputRegisterWrite` API. The `inputRegisterWrite` API writes only to the DOC data-input register.

3.1.2 DOC HAL Module Limitations

Refer to the latest *SSP Release Note* for any additional operational limitations for this module.

4. Including the DOC HAL Module in an Application

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

Note: This section assumes that 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 DOC Driver to an application, simply add it to a thread using the stacks selection sequence provided in the following table. (The default name for the DOC Driver is `g_doc0`. This name can be changed in the associated Properties window.)

Table 3 DOC HAL Module Selection Sequence

Resource	ISDE Tab	Stacks Selection Sequence
<code>g_doc0</code> Data Operation Circuit Driver on <code>r_doc</code>	Threads	New Stack > Driver > Monitoring > Data Operation Circuit Driver on <code>r_doc</code>

When the DOC HAL module on `r_doc` 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 in the text box highlighted in **Red**.

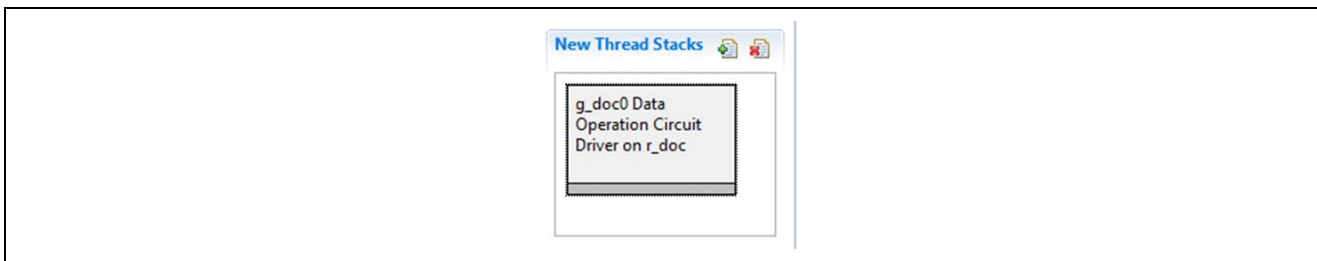


Figure 2 DOC HAL Module Stack

5. Configuring the DOC HAL Module

The DOC 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. 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 they are identified with a lock icon for the ‘locked’ property in the Properties 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 indicates 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 it is easily visible within 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 following configuration table settings. This helps to orient you and can be a useful approach for learning the processes of developing with SSP.

Table 4 Configuration Settings for the DOC HAL Module on r_doc

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled (Default: BSP)	Enable or disable the parameter error checking
Name	g_doc0	Module name
Event	Comparison mismatch, Comparison match, Addition overflow, Subtraction underflow (Default: Comparison mismatch)	Specify the event which will trigger the DOC interrupt
Callback	NULL	A user callback function can be defined here. If this callback function is provided, it is called from the interrupt service routine (ISR) when the configured DOC event occurs. 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.
DOC Interrupt Priority	Priority 0 (highest), 1,2,3,4,5,6,7,8,9,10,11,12,13,14, 15 (lowest, not valid if using Thread X), Disabled (Default: Disabled)	Use the pull down to set the DOC interrupt priority

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

In some cases, settings other than the defaults can be desirable. For example, it might be useful to select different events depending on the operation desired.

5.1 DOC HAL Module Clock Configuration

The DOC HAL module does not require any specific clock configuration.

5.2 DOC HAL Module Pin Configuration

The DOC HAL module does not require any specific pin configurations.

6. Using the DOC HAL Module in an Application

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

1. Initialize the DOC using the `open` API.
2. Set register values in DODIR and DODSR using the `write` API.
3. Stream data to the DOC using the `inputRegisterWrite` API.
4. Read the status of the comparison using the `statusGet` API or in the callback if enabled.
5. Clear status flags using the `statusClear` API.
6. Close the module using the `close` API.

The following figure illustrates these common steps in a typical operational flow diagram:

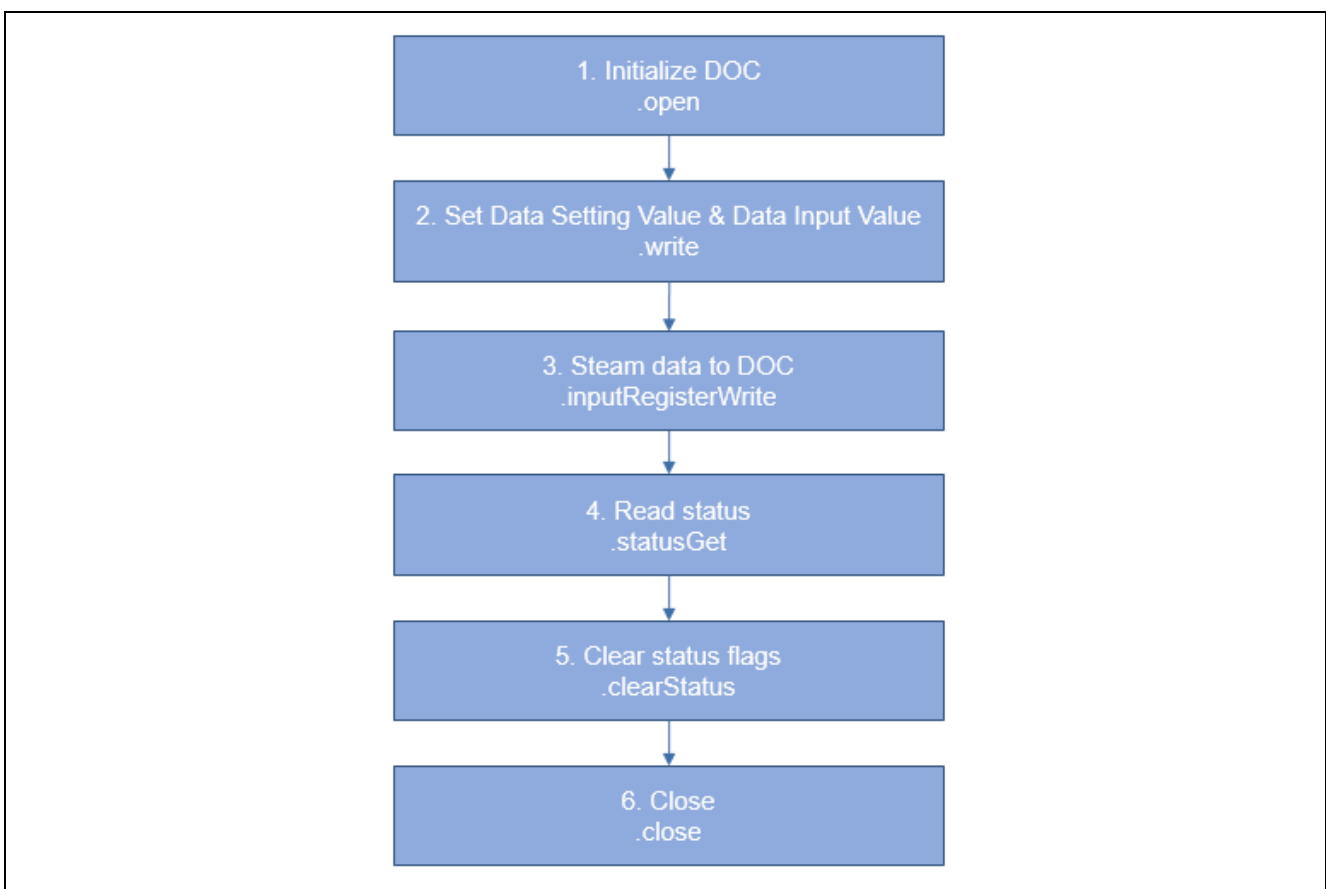


Figure 3 Flow Diagram of a Typical DOC HAL Module Application

7. The DOC HAL Module Application Project

The application project demonstrates the typical use of the DOC HAL module APIs. The application project initializes two DOC HAL module instances to demonstrate the use of the DOC in both the match/mismatch mode and the data addition mode. (The application project does not demonstrate data subtraction mode.) The data used for match and data addition modes are defined in the application project file `doc_hal.h`. It is a simple task for the user to change the data values being used and observe the behavioral effect. The application project performs two data comparisons for each driver instance created with the results displayed on LEDs, if the debugger is connected through the debug console window.

The following table identifies the target versions for the associated software and hardware used by the application project.

Table 5 Software and Hardware Resources Used by the Application Project

Resource	Revision	Description
e ² studio	5.3.1 or later	Integrated Solution Development Environment
SSP	1.2.0 or later	Synergy Software Platform
IAR EW for 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.

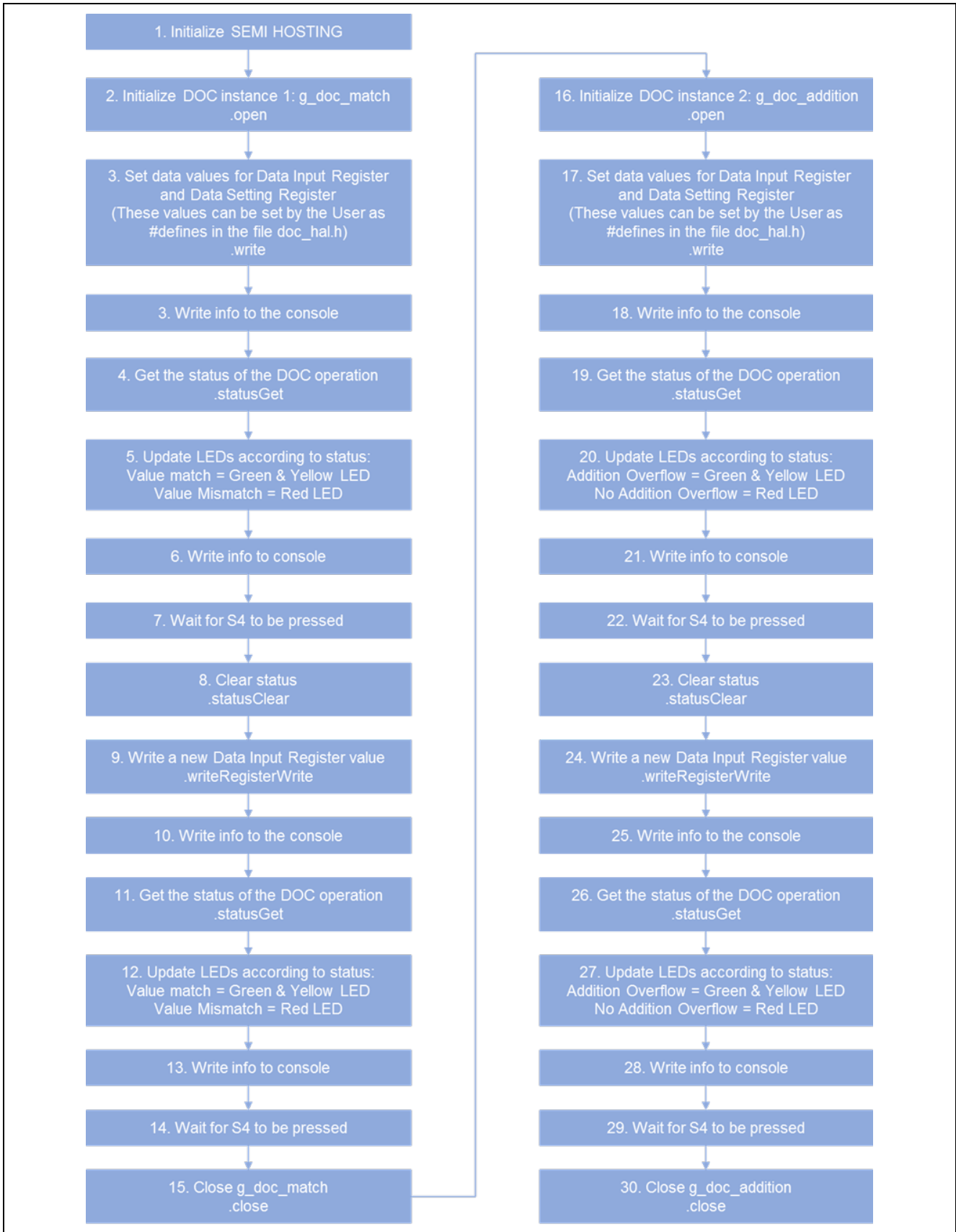


Figure 4 Data Operation Circuit HAL Module Application Project Flow Diagram

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

The first section of `doc_hal.c` has the header files, global variables and function prototypes.

The next section is the application project entry function. This section initializes semi-hosting (if defined) and sends debug strings to the debug console. After initializing semi-hosting, the first DOC driver instance is initialized: `g_doc_match`. Initial data values for reference and input data are set and written to the DOC. Writing the data causes the DOC to perform a comparison. If semi-hosting is enabled, the values of the data being compared get written to the console. The `statusGet` API determines the comparison result. LEDs illuminate to indicate a data match or mismatch. The application waits for the user push button, S4, to be pressed before performing the next comparison.

In the second comparison, the reference data is left the same as before and only the input data is changed using the `inputRegisterWrite` API. Again, the result of the comparison is determined by the `statusGet` API, and LEDs illuminate to indicate a data match or mismatch. The application waits for the user push button, S4, to be pressed before performing the next comparison.

At this point, the DOC HAL module instance `g_doc_match` is closed, while the second instance, `g_doc_addition`, is opened. Initial data values for reference and input data are set and written to the DOC. The process of writing the data causes the DOC to perform a comparison. If semi-hosting is enabled, the values of data being compared get written to the console. The result of comparison is determined by the `statusGet` API. LEDs illuminate to indicate a data match or mismatch. The application waits for the user push button, S4, to be pressed before performing the next comparison.

The last section of the `doc_hal.c` file contains both a function that processes the DOC operation result and a function that facilitates LEDs control.

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

Table 6 DOC HAL Module Configuration Settings for the Application Project

ISDE Property	Value Set
Name	<code>g_doc_match</code>
Event	Comparison match
Callback	NULL
DOC Interrupt Priority	Disabled

Table 7 Data Operation Circuit HAL Module (instance 2) Configuration Settings for the Application

ISDE Property	Value Set
Name	<code>g_doc_addition</code>
Event	Addition overflow
Callback	Null
DOC Interrupt Priority	Disabled

Note: The second DOC HAL module instance has the interrupt priority set to Disabled. Disabling is necessary because the DOC interrupts are already enabled by the first instance configuration.

8. Customizing the DOC HAL Module for a Target Application

The application project demonstrates the use of the DOC polling for status using the `statusGet` API. The DOC can be configured to generate an interrupt on one of four conditions: data match, data mismatch, data overflow or data underflow. The ability to generate an interrupt means data checking can be done without the need for any CPU intervention. For example, a data-transfer peripheral, such as the DMA or DTC, could be configured to transfer data to the DOC. With the DOC configured in addition mode, an interrupt is generated as the data crosses a threshold.

9. Running the DOC HAL Module Application Project

To run the DOC HAL Module application project and to see it executed on a target kit, you can simply import it into your ISDE, compile, and run debug.

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

Note: The following steps are 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 DOC HAL Module application project, follow these steps:

1. Create a new Renesas Synergy project for the SK-S7G2 board called DOC_HAL_MG_AP.
2. Select the **Threads** tab.
3. Add the DOC HAL module to the HAL/Common thread.
4. Add the second DOC HAL module to the HAL/Common thread.
5. Click on the **Generate Project Content** button.
6. Add the code from the supplied project file `doc_hal.c` or copy over the generated `doc_hal.c` file.
7. Connect to the host PC through a micro USB cable to J19 on SK-S7G2.
8. Start to debug the application.
9. The output can be viewed on the LEDs.

10. DOC 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 in previous generations of embedded systems. The Renesas Synergy™ Platform makes these steps much less time consuming and removes common errors like conflicting configuration settings or the incorrect selection of lower-level drivers. By allowing work to begin at a high level, the use of high-level APIs (as demonstrated in the application project) illustrates reductions in both development time and the time required to use or create lower-level drivers in older development environments.

11. DOC HAL Module Next Steps

After you have mastered a simple DOC HAL module project, you may want to review a more complex example. As the DOC utilizes an interrupt service routine (ISR), it can be used as a trigger for some interrupt-based actions or ELC events in an application.

12. DOC 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_doc` module reference materials and resources are available on the Synergy Knowledge Base: [https://en-](https://en-us.knowledgebase.renesas.com/English_Content/Renesas_Synergy%E2%84%A2_Platform/Renesas_Synergy_Knowledge_Base/Module_Guide_References%3Ar_doc)

[us.knowledgebase.renesas.com/English_Content/Renesas_Synergy%E2%84%A2_Platform/Renesas_Synergy_Knowledge_Base/Module_Guide_References%3Ar_doc](https://en-us.knowledgebase.renesas.com/English_Content/Renesas_Synergy%E2%84%A2_Platform/Renesas_Synergy_Knowledge_Base/Module_Guide_References%3Ar_doc).

Website and Support

Visit the following vanity URLs to learn about key elements of the Synergy Platform, download components and related documentation, and get support.

Synergy Software	renesassynergy.com/software
Synergy Software Package	renesassynergy.com/ssp
Software add-ons	renesassynergy.com/addons
Software glossary	renesassynergy.com/softwareglossary
Development tools	renesassynergy.com/tools
Synergy Hardware	renesassynergy.com/hardware
Microcontrollers	renesassynergy.com/mcus
MCU glossary	renesassynergy.com/mcuglossary
Parametric search	renesassynergy.com/parametric
Kits	renesassynergy.com/kits
Synergy Solutions Gallery	renesassynergy.com/solutionsgallery
Partner projects	renesassynergy.com/partnerprojects
Application projects	renesassynergy.com/applicationprojects
Self-service support resources:	
Documentation	renesassynergy.com/docs
Knowledgebase	renesassynergy.com/knowledgebase
Forums	renesassynergy.com/forum
Training	renesassynergy.com/training
Videos	renesassynergy.com/videos
Chat and web ticket	renesassynergy.com/support

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Aug 2, 2017	-	Initial version
1.10	Nov 6, 2018	-	Edits to tables 6 and 7 and text between

All trademarks and registered trademarks are the property of their respective owners.

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 claims 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, drawings, charts, programs, algorithms, and 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 reverse engineer 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, copying or reverse engineering.
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; 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.
Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or 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 damage (space system; 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 any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
6. When using 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 dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of 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. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a 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. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating 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. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using 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 comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
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 directly or indirectly controlled subsidiaries.

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

(Rev.4.0-1 November 2017)



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

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

Renesas Electronics Corporation

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc.

1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

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: +44-1628-651-700

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, 200333 P. R. China
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 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.

17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338