System Architecture Design Guide

RENESAS

CONFIDENTIAL

Security
Board Support Package

System Architecture Design Guide: Software

R-Car Series, 3rd Generation

— Preliminary —
Specifications common to R-Car Series Products
R-Car H3
R-Car M3
R-Car M3N
R-Car E3
R-Car D3

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (http://www.renesas.com).

Renesas Electronics
www.renesas.com

Rev.3.0.3 Aug. 2021

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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. 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.
6. 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.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. 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.
9. 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.
10. 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.
11. 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.
12. 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.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

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: www.renesas.com/contact/.

Trademark

- Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
- Windows and Windows Media are registered trademarks of Microsoft Corporation in the United States and other countries.
- Other company names and product names mentioned herein are registered trademarks or trademarks of their respective owners.
- Registered trademark and trademark symbols (® and ™) are omitted in this document

How to Use This Manual

- **[Readers]**
  This manual is intended for engineers who develop products which use the R-Car H3/M3/M3N/E3/D3 processor.

- **[Purpose]**
  This manual is intended to give users an understanding of the functions of the R-Car H3/M3/M3N/E3/D3 processor device driver and to serve as a reference for developing hardware and software for systems that use this driver.

- **[How to Read This Manual]**
  It is assumed that the readers of this manual have general knowledge in the fields of electrical
  — engineering, logic circuits, microcontrollers, and Linux.
  → Read this manual in the order of the CONTENTS.
  — To understand the functions of a multimedia processor for R-Car H3/M3/M3N/E3/D3
  → See the R-Car H3/M3/M3N/E3/D3 User’s Manual.
  — To know the electrical specifications of the multimedia processor for R-Car H3/M3/M3N/E3/D3
  → See the R-Car H3/M3/M3N/E3/D3 Data Sheet.

- **[Conventions]**
  The following symbols are used in this manual.
  Data significance: Higher digits on the left and lower digits on the right
  Note: Footnote for item marked with Note in the text
  Caution: Information requiring particular attention
  Remark: Supplementary information
  Numeric representation: Binary ... ××××, 0b××××, or ××××B
  Decimal ... ××××
  Hexadecimal ... 0x×××× or ××××H
  Data type: Double word … 64 bits
  Word … 32 bits
  Half word ... 16 bits
  Byte ... 8 bits

Table of Contents

1. Overview 1

1.1 Overview 1

1.2 References 1

1.3 Terminology 2

2. Architecture 3

2.1 System structure 3

2.1.1 Secure World 4

2.1.1.1 Monitor (Trusted Firmware-A BL31) 4

2.1.1.2 Secure OS (OP-TEE OS) 4

2.1.1.3 TA (Trusted Application) 4

2.1.1.4 Device drivers 5

2.1.2 Normal world 5

2.1.2.1 CA (Client Application) 5

2.1.2.2 OP-TEE Client 5

2.1.2.3 Rich OS 5

2.1.2.4 Device driver (OP-TEE driver) 5

2.2 Secure World control 6

2.2.1 OP-TEE 6

2.2.1.1 TA (Trusted Application) 6

2.2.1.2 Request to TA 7

2.2.1.3 TA binary image 8

2.2.1.4 Cryptographic Function 9

2.2.1.5 Logging 9

2.2.1.6 Remote Procedure Call 10

2.2.2 MMU and Cache 11

2.2.3 Interrupt control 13

2.2.4 Access control 14

2.2.4.1 Access control of Built-in IP (Life Cycle) 14

2.2.4.2 Access control of CPG(SRST, MSTP) 16

2.2.4.3 RAM Protection 17

2.3 Power Management 18

2.3.1.1 PSCI (Power State Coordinate Interface) 18

2.4 Secure Storage 25

2.4.1 Secure Storage Key(SSK) 26

2.4.2 Secure Storage in OP-TEE 26

1. Overview

1.1 Overview

This document describes the architecture of the Security BSP with TrustZone implemented on R-Car H3/M3/M3N/E3/D3/H3e/M3e/M3Ne/E3e/D3e series SoCs. TrustZone is a system-wide approach which is designed by Arm for secure system. Security BSP is standard package for R-Car H3/M3/M3N/E3/D3/H3e/M3e/M3Ne/E3e/D3e series.

1.2 References

The following table shows the document related to this document.

Number	Issue	Title	Edition
1	Renesas Electronics	R-Car Series, 3rd Generation User’s Manual: Hardware	#0
2	Renesas Electronics	Security Board Support Package User's Manual: Software R-Car H3/M3/M3N/E3/D3 Series	#0
3	Renesas Electronics	Initial Program Loader User’s Manual: Software	#0
4	Arm	Arm® Architecture Reference Manual Armv8, for Armv8-A architecture profile	A.h
5	Arm	Power State Coordination Interface (PSCI)	Version 1.1
6	Arm	Arm® Generic Interrupt Controller	Architecture version 2.0
7	GlobalPlatform Device Technology	TEE Client API Specification	Version 1.1
8	GlobalPlatform Device Technology	TEE Internal API Specification	Version 1.1
9	GitHub	OP-TEE document https://optee.readthedocs.io/en/3.13.0/	Tag3.13.0

#0 : This manual refers to the latest edition.

1.3 Terminology

The following table shows the terminology related to this packages.

Table 1.1 Terminology

Terms	Explanation
Exception Levels (EL0/EL1/EL3)	The Armv8-A architecture defines a set of Exception levels, EL0 to EL3, where: ♦ If ELn is the Exception level, increased values of *n* indicate increased software execution privilege. ♦ Execution at EL0 is called unprivileged execution ♦ EL2 provides support for virtualization of Non-secure operation. ♦ EL3 provides support for switching between two Security states, Secure state and Non-secure state.
PSCI	Power State Coordination Interface It is defines a Standard interface for power management that can be used by OS vendors for supervisory software working at different levels of privilege on an Arm device.
Rich Execution Environment (REE)	An environment that is provided and governed by a Non-secure OS, potentially in conjunction with other supporting operating systems. It is outside of the TEE.
Trusted Execution Environment (TEE)	An execution environment that runs alongside but isolated from an REE. A TEE has security capabilities and meets certain security-related requirements: It protects TEE assets from general software attacks, defines rigid safeguards as to data and functions that a program can access, and resists a set of defined threats.

2. Architecture

TrustZone is an Arm function that enables security feature on Arm CPU and its peripherals. R-Car H3/M3/M3N/E3/D3 SoC is designed to work aligned with TrustZone.

2.1 System structure

Figure 2.1 shows the system structure of TrustZone system on R-Car H3/M3/M3N/E3/D3.

The shaded portion of the gray is the subject of this guide of the software.

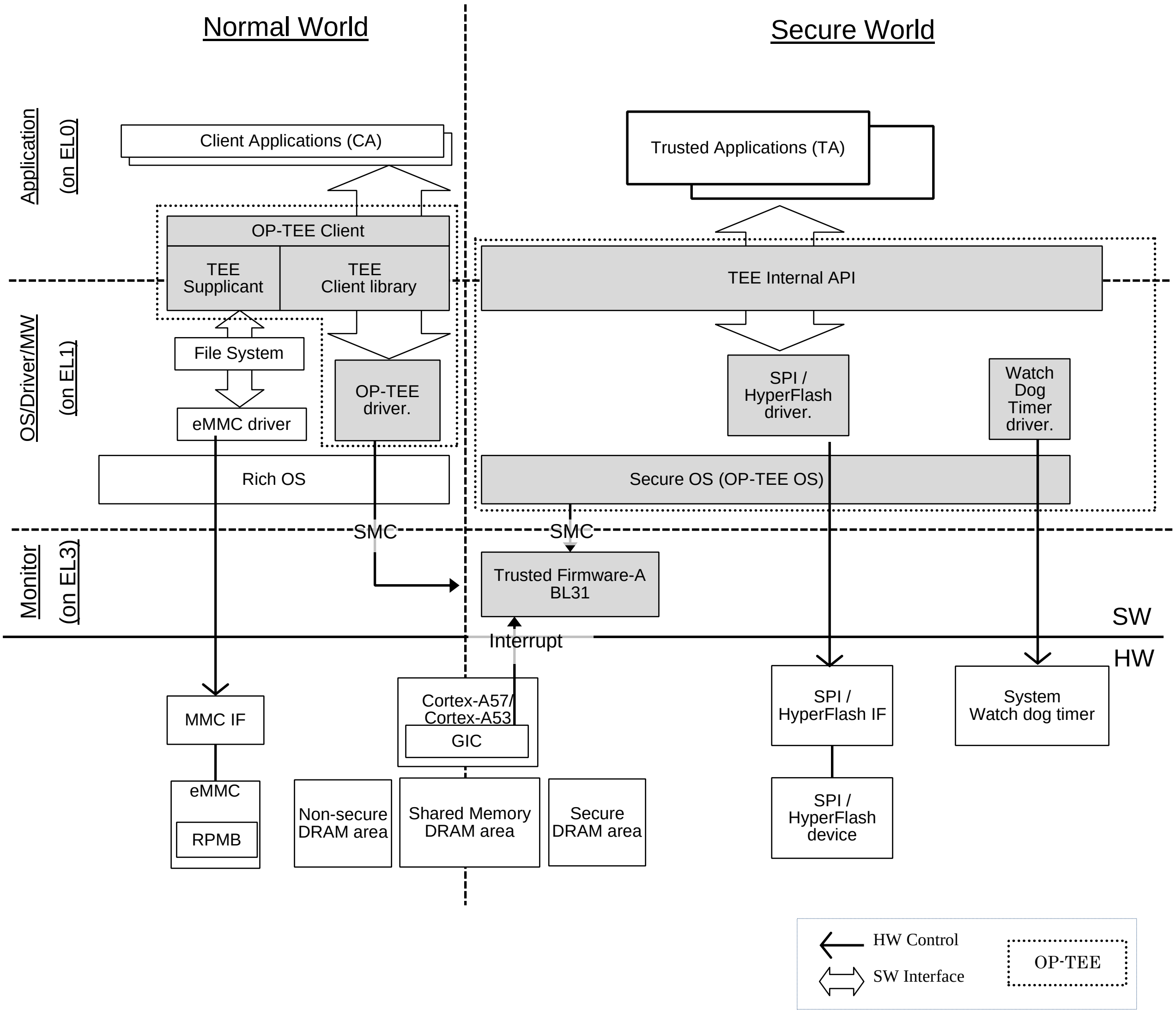

Figure 2.1 System structure of TrustZone system on R-Car H3/M3/M3N/E3/D3.

The following list shows the functions provided by this system.

Table 2.1 Functions provided by this system

Function	Summary
Secure World control	For operating within the Secure World and the resources controlled by the Secure World side.(See 2.2)
Power Management	For power control using the Arm PSCI interface.(See 2.3)
Secure Storage	For storing confidential data in an external storage.(See 2.4)

The following sections describe each SW component on Secure World and Normal World.

2.1.1 Secure World

Secure World represents Secure state referred to in the Armv8 architecture.

SW components shown in the right side of vertical dashed line in Figure 2.1 works on Secure World. That means the code is placed on protected memory (Secure DRAM area) and executed by CPU with secure state. Refer to 2.2.4 for access control.

2.1.1.1 Monitor (Trusted Firmware-A BL31)

Monitor is a software running in EL3 defined Armv8 architecture.

Monitor manages the switches between the Secure World and the Normal World. If it need to switch the world, the Secure Monitor saves register data and restore register for the next world. When a SMC, FIQ and IRQ are generated, the Exception Handler decides to need to switch the world (see 2.2.3). Also, Monitor controls power supply control that conforms to the Power State Coordination Interface (PSCI) (see 2.3.1.1).

In this system, Trusted Firmware-A BL31 is used as Monitor software.

2.1.1.2 Secure OS (OP-TEE OS)

Secure OS is OS which provides basic functions (resource management, IO access etc.) for Secure World.

In this system, OP-TEE OS is used as Secure OS which is running in the Secure World. This package provides the TEE Internal API defined by the GlobalPlatform TEE standard to the Trusted Applications that it executes and accesses secure resources.

2.1.1.3 TA (Trusted Application)

Applications running on the OP-TEE OS are called TA.
TA is a passive type of application.
TA receives the request command from CA or another TA to execute it. Then, it returns the results to them (See 2.2.1.2).

2.1.1.4 Device drivers

Secure drivers controls devices working on Secure World side.

- Watchdog timer driver
  OP-TEE OS provides driver for System Watchdog timer. This timer watches Secure World side.
  When Secure World side freezes, it is detected by this timer expired and accepted the detection in FIQ.
  This driver can be controlled by a pseudo TA.
- SPI/HyperFlash driver
  By using this driver, TA can accesses SPI Flash/HyperFlash connected to the SoC (See 2.4).

2.1.2 Normal world

Normal world represents Non-secure state referred to in the Armv8 architecture.

Normal world is also referred to as a Non-Secure World for the Secure World.
SW components shown in the left side of vertical dashed line in Figure 2.1 work on Normal World.

2.1.2.1 CA (Client Application)

CA running inside of the Normal World

CA makes use of the TEE Client API to access facilities provided by TA inside the Secure World.

2.1.2.2 OP-TEE Client

This package provides the TEE Client library and the TEE Supplicant.

TEE Client library is the library that contains APIs defined by the GlobalPlatform TEE standard. This library called by CA are executed on Non-secure OS to communicate with the OP-TEE OS and TAs.

TEE Supplicant is client for OP-TEE OS in Secure World to use Rich OS resource.

2.1.2.3 Rich OS

Rich OS is OS which works in Normal World and also referred to as Non-Secure OS.

It includes the following differences from original one.

- Add driver (OP-TEE driver) which calls SMC to switch to Secure World (see 2.1.2.4).
- Modification for PM which is the component for power management. It controls the CPU core in conjunction with the Secure World (see 2.3.1.1).

2.1.2.4 Device driver (OP-TEE driver)

OP-TEE Driver for Non-secure OS allows communication between the Non-secure OS and the OP-TEE OS via Trusted Firmware-A BL31 as Monitor.

2.2 Secure World control

2.2.1 OP-TEE

This system adopts the OP-TEE OS as Secure OS.
This section describes about functions provided by OP-TEE OS.

2.2.1.1 TA (Trusted Application)

TA running on the OP-TEE is an application to have control of security.
TA can use the functions provided by the OP-TEE OS by using interface defined by the TEE Internal API.

Table 2.2 Functions list for TA

No.	Function	Summary
1	Encryption function	For encryption function which TA uses.
2	Storage function	For function of storing data like confidential data and key info which TA uses.
3	Memory management function	For memory function for Secure World such as alloc, memmove, memcmp, and so on.
4	Timer function	For function for operation of time information for Secure World such as gettimeofday, wait, and so on.

2.2.1.2 Request to TA

As a way for requesting the processing to the TA, there are two routes of the following.

1. Root of request from Client Application in Normal World side.
2. Root of request from another Trusted Application which is already working.

The following Figure 2.2 shows the two routes.

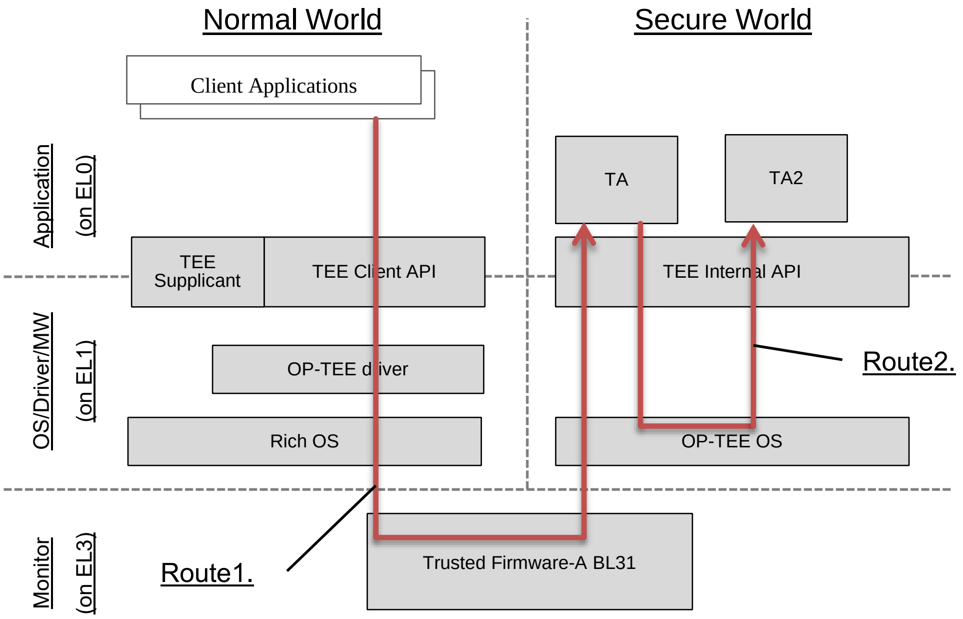

Figure 2.2 Routes that requires the process to TA

TEE Client API is used to request to TA from Normal World.
TEE Internal API is used to request to TA from another TA.
About these APIs which this system supports, please refer to External Interface in References [2] for details.

2.2.1.3 TA binary image

The following Table 2.3 shows TA types.

Table 2.3 TA type

Type	Summary
Dynamic TA	･The Dynamic TA executes each processing through TEE Internal API. ･Store the execute file of TA in filesystem of the Rich OS. ･The execution image of the Dynamic TA is not included in OP-TEE OS image. It is necessary to build individually. Please refer to "How to implement a Dynamic TA" in References [2] for implementation and build details.
pseudo TA	･The pseudo TA must call the OP-TEE OS function directly instead of calling the TEE Internal API, because it runs in kernel context. ･The execution image of the pseudo TA is included in OP-TEE OS image.
early TA	･The early TA executes each processing through TEE Internal API. ･The execution image of the early TA is included in OP-TEE OS image.

Figure 2.3 below is a flow that Dynamic TA is loaded.

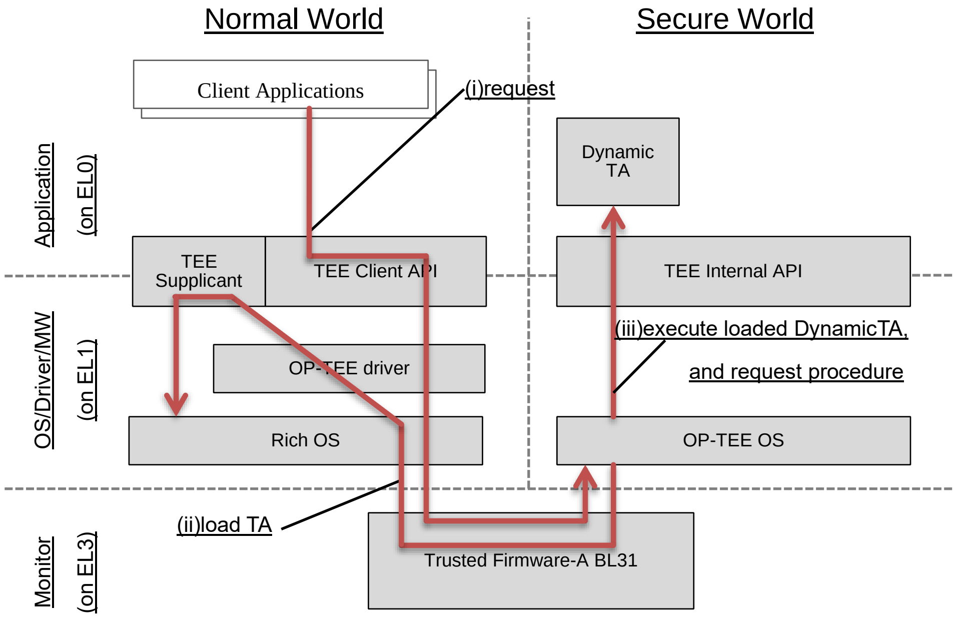

Figure 2.3 Flow of procedure that Dynamic TA is loaded.

It is the explanation of each number of Figure 2.3 as follows.

(i) OP-TEE OS checks the request from application whether pseudo TA, early TA or Dynamic TA.

(ii) In case of the Dynamic TA, OP-TEE OS requests loading a Dynamic TA to TEE Supplicant in the Normal World side. TEE Supplicant accesses Rich OS filesystem for getting a Dynamic TA image, and transfers the image to OP-TEE OS side.

(iii) OP-TEE OS received Dynamic TA image executes Dynamic TA, and requests processing to it.

2.2.1.4 Cryptographic Function

When TA executes encrypt processing, TA uses TEE Internal API provided from OP-TEE.

Cryptographic Function under TEE Internal API do encryption, decryption, creating key, and creating random number.

This system uses LibTomCrypt software as Cryptographic Function. LibTomCrypt is a software engine that is installed in a standard in the OP-TEE.

About TEE Internal APIs which this system supports, please refer to External Interface in References [2] for details.

2.2.1.5 Logging

OP-TEE OS standard outputs logging data in Secure World side (OP-TEE OS/TA) by using UART. But in this system, OP-TEE OS outputs logging data in Secure World side to Secure DRAM area.

As well as, Trusted Firmware-A BL31 standard outputs logging data by using UART. But in this system, Trusted Firmware-A BL31 outputs logging data to Secure DRAM area.

Also, in this system, logging data(*) in Secure World side is outputted to Normal World side by using Remote Procedure Call (see 2.2.1.6) which transfers to Non-Secure DRAM area. This function is default not enable. So, it needs to change option settings to enable at the build.

(*) The log of OP-TEE OS and TA applies, but the log of Trusted Firmware-A BL31 does not apply.

Please refer to References [2] for this system logging function details.

2.2.1.6 Remote Procedure Call

There is a case that OP-TEE requests function such as filesystem, time management, and resource management to OS of Normal World side.

In that case, OP-TEE calls Remote Procedure Call, transits the state of TA to 'suspend', and switches control to Normal World side.

The table below shows the function of the Remote Procedure Call.
“OP-TEE log output to Normal World side” function is provided by this system. The others is provided by OP-TEE standard.

Table 2.4 Remote Procedure Call Function list

Function	Summary
MUTEX WAIT	OP-TEE OS calls this function when the wait has occurred by using MUTEX in it.
WAIT_QUEUE	OP-TEE OS calls this function when the wait has occurred by using WAIT_QUEUE in it.
WAIT	OP-TEE OS has no time management function, and is not preemptive OS. Therefore, OP-TEE OS calls this function if it requires a wait operation.
Shared Memory **Management**	OP-TEE OS calls this function when it allocates (or releases) memory which is used for data passing between OP-TEE and Normal World. (Management of this memory has implemented on the OS side of the Normal World.)
TEE Supplicant	OP-TEE OS calls this function if the following feature to work ・load and free Dynamic of TA ・getting Time(date/time) ・accessing Rich OS filesystem
IRQ event delivery	OP-TEE OS calls this function if the IRQ occurs in the Secure World. By this function, occurring of IRQ is notified to the Rich OS side.
OP-TEE log output to Normal World side	OP-TEE OS calls this function if there is logging data of Secure World side (OP-TEE OS/TA).

2.2.2 MMU and Cache

This section describes about MMU and Cache settings in the Secure World side.

MMU and Cache settings of Trusted Firmware-A BL31 and OP-TEE OS for this system are set when they each start up.

Following each tables shows that each MMU and Cache settings.

Table 2.5 MMU settings of Trusted Firmware-A BL31

Area	Address	MMU Page Table Settings		
		size	NS	Memory type
image including heap & stack	0x44000000 – 0x44010FFF	0x00011000	0	Write-Back Cacheable
	0x44011000 – 0x44011FFF	0x00001000	0	device nGnRE
shared memory between BL2 and BL31	0x4403E000 – 0x4403EFFF	0x00001000	0	Write-Back Cacheable
Stack area for system crash	0x4403F000 – 0x4403FFFF	0x00001000	0	Write-Back Cacheable
log	0x44040000 – 0x44053FFF	0x00014000	0	device nGnRE
hardware register(*)	0xE6000000 – 0xE62FFFFF	0x00300000	0	device nGnRE
System RAM for driver of I2C for DVFS	0xE6300000 – 0xE6301FFF	0x00002000	0	Write-Back Cacheable
stack area for driver of I2C for DVFS	0xE6302000 – 0xE6302FFF	0x00001000	0	device nGnRE
hardware register(*)	0xE6360000 – 0xFFFFFFFF	0x19CA0000	0	device nGnRE

(*)Trusted Firmware-A BL31 maps hardware registers which are used by it.

Table 2.6 MMU settings of OP-TEE OS

Area	Address	MMU Page Table Settings		
		size	NS	Memory type
image including heap & stack	0x44100000 – 0x443FFFFF	0x00300000	0	Write-Back Cacheable
TA image(*1) including heap & stack	0x44400000 – 0x461FFFFF	0x01E00000	0	Write-Back Cacheable
TA area for verification	0x46200000 – 0x462FFFFF	0x00100000	0	device nGnRE
log (for Secure World)	0x46400000 – 0x46413FFF	0x00014000	0	device nGnRE
Stand-alone FS Work area	0x46432000 – 0x464B1FFF	0x00080000	0	device nGnRE
Non-cache stack area	0x46500000 – 0x465FFFFF	0x00100000	0	device nGnRE
Crypto Engine Work area	0x46600000 – 0x466FFFFF	0x00100000	0	Write-Back Cacheable
shared memory between Secure World and Normal World ~~(The latter half 1 MB is used for the log (for Normal World) area.)~~	0x47E00000 – 0x47EFFFFF	0x00100000	1	Write-Back Cacheable
log (for Normal World) area.	0x47FEC000 – 0x47FFFFFF	0x00014000	1	Write-Back Cacheable
hardware register(*2)	0xE6000000 – 0xE62FFFFF	0x00300000	0	device nGnRE
hardware register(*2)	0xE6300000 – 0xE63FFFFF	0x00100000	0	device nGnRE
hardware register(*2)	0xE6600000 – 0xE67FFFFF	0x00200000	0	device nGnRE
hardware register(*2)	0xE6A00000 – 0xE73FFFFF	0x00A00000	0	device nGnRE
MaskROM API(*2)	0xEB100000 – 0xEB1FFFFF	0x00100000	0	device nGnRE

hardware register(*2)	0xEE200000 – 0xEE3FFFFF	0x00200000	0	device nGnRE
hardware register(*2)	0xF1000000 – 0xF11FFFFF	0x00200000	0	device nGnRE
hardware register(*2)	0xFFE00000 – 0xFFFFFFFF	0x00200000	0	device nGnRE

(*1)MMU & Cache settings of TA conforms OP-TEE standard.

(*2)OP-TEE OS maps hardware registers which are used by it.

NS: represents NS field in Attribute fields VMSAv8-64 Block and Page descriptors.
NS == 0: Access the Secure physical address space.
NS == 1: Access the Non-secure physical address space.

MemoryType: represents memory type which is controlled by AttrIndx [2:0] in Attribute fields VMSAv8-64 Block and Page descriptors.
device nGnRE : Non-Cacheable

Please refer to References [4] for above Arm register contents details.

2.2.3 Interrupt control

As GIC interrupt, SGI (Software-generated interrupt), PPI (Private Peripheral Interrupt), and SPI (Shared Peripheral Interrupt) types are defined.

Summary of each interrupt shown in Table 2.7 below.

Table 2.7 Type of interrupt

Type	Interrupt ID	Summary
SGI	0～15	This type is interrupt to be used between the CPU, and it is issued by the software.
PPI	16～31	This type is interrupt with a unique within the CPU(Generic Timer etc)
SPI	32～480	This type is interrupt from the built-in IP in the SoC.

Please refer to References [6] for GIC details.

In accordance with Arm Security Technology, This system classifies the interrupt GIC issued as follows.

- FIQ :Interrupt for Secure World
- IRQ :Interrupt for Normal World

Interrupt to be set as the FIQ is shown in Table 2.8 below.

Table 2.8 Interrupt to be set as the FIQ

Type	Interrupt ID	Summary
SGI	8～15	Reserved as a software interrupt used between CPUs in Secure OS.
PPI	29	Reserved as a interrupt for Secure physical timer
SPI	70	Reserved as a interrupt for RPC
	97	Reserved as a interrupt for Crypto Engine sec
	102	Reserved as a interrupt for Crypto Engine sec
	166	Reserved as a interrupt for System Timer
	171	Reserved as a interrupt for System Up Timer
	173	Interrupt for System WDT

These settings are set to GIC at initialize of Trusted Firmware-A BL31.

For the mechanism to interrupt handling, the operation of the OP- TEE standard.
Please refer to “Entry and exit of secure world” in References [8] for OP-TEE standard interrupt handling details.

2.2.4 Access control

This section describes about access control in this system.

2.2.4.1 Access control of Built-in IP (Life Cycle)

In R-Car H3/M3/M3N/E3/D3, there is an IP called Life Cycle that controls access among built-in IP. This system can protect IP used by Secure World side from the application in Normal World side and IP belonging Non-Secure.

The concept of access protection by the Life Cycle is shown in Figure 2.4 below.

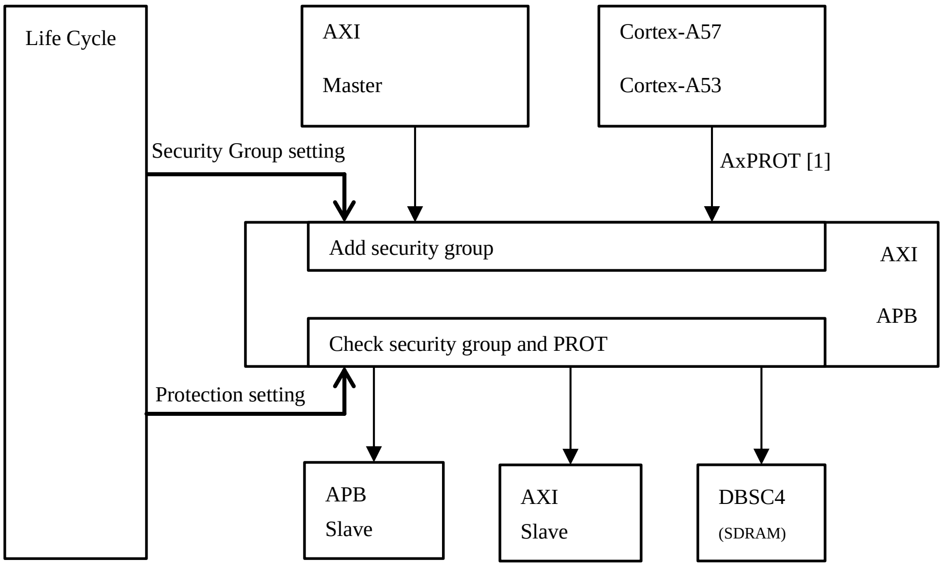

Figure 2.4 Access protection by Life Cycle

(1)Bus Master side

When bus access from Bus Master (such as AXI Master, Cortex-A57, Cortex-A53) to AXI, an attribute in accordance with Security Group setting set to Life Cycle is added to the access. Security Group can be set 4 levels.

Also, when bus access occurs from Cortex-A57/Cortex-A53, AxPROT [1] in accordance with the World where CPU is running is added to the access. Bus Master not set AxPROT [1] is treated as setting NS to AxPROT [1].

The following table shows relationship of PROT and security groups

Table 2.9 Relationship of AxPROT[1] and security groups.

AxPROT[1]	Security Group	Summary
S (0)	-	Access from Secure World of Cortex-A57/Cortex-A53
NS (1)	11 (secure)	Access from secure attribute of AXI Master
	10 (group2)	Not used in this system
	01 (group1)	Not used in this system
	00 (public)	Access from non-secure attribute of AXI Master

(2)Bus Slave side

When access to Bus Slave such as APB Slave, AXI Slave and DBSC occurs, access protection in accordance with Protection setting set to Life Cycle works.

Protection setting is set by the following two settings in addition to Security Group.

- Slave Security : Attribute setting of the IP is Secure (0) or non-Secure (1).
- Write Protect : Attribute setting of the IP is Write Protect (Read Only) (1) or not (0).

For accessing IPs in the following Table 2.10, this system sets Protection setting to Bus Master side.

Table 2.10 IP set Slave Security=0 and Security Group=11(Secure) to Bus Slave side

IP	Summary
DBSC4	External Bus Controller for SDRAM
AXI-bus(Main Memory domainAXI)	for memory protection etc
MaskROM	ROM area including boot program
RST	For CPU reset control
SCEG Secure Core	Secure Engine Core for Secure World side
SCEG PKA(secure APB)	Secure Engine PKA for Secure World side
RPC	SPI Multi I/O Bus Controller for accessing HyperFlash/QSPI
SWDT	For System Watch Dog Timer
SCMT	For System Timer
SUCMT	For System Up Time Clock Timer
Life Cycle	For controlling access protection at on-chip bus system ports etc

About combination of Security Group setting and Protection setting, please refer to section of Life Cycle in the References [1].

Security Group setting and Protection setting are set by IPL at boot.

2.2.4.2 Access control of CPG(SRST, MSTP)

CPG has an access protection function for the input from APB Bus. However, apart from that, CPG has an access protection function about the control of the SRST register. SRST register(SRCRn) can prohibit access to each of its bit. SRST register(SRCRn) is assigned built-in IP for each bit, and can reset specified built-in IP by bit control.

The following Figure 2.5 shows access protection of CPG.

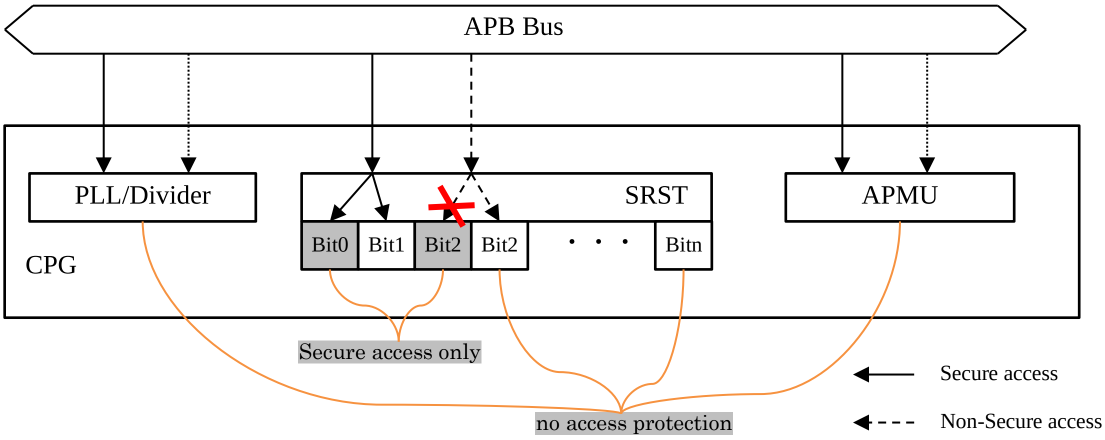

Figure 2.5 CPG access protection

This system protects control of built-in IP used by secure side by using access control function of SRST.

The following Table 2.11 shows that bit allocation of SRST register corresponds to built-in IP used by secure side.

Table 2.11 Built-in IP used by Secure World side and bit allocation of SRST register

Built-in IP	SRST register SRCR[bit]
SCEG(PKA Core)	SRCR2[26]
SCEG(Secure Core)	SRCR2[28]
SCMT	SRCR4[0]
SWDT	SRCR4[1]
SUCMT	SRCR4[31]
RPC	SRCR9[17]

Please refer to section of Software Reset Register in the References [1] for details.

For MSTP, secure dedicated register is used. Please refer to section of Software Reset Register in the References [1] for details.

2.2.4.3 RAM Protection

This system protects access to the SDRAM in the Secure World side from Normal World side by using the memory protection function of AXI-bus.

This system allocates SDRAM for Secure World side 0x43F00000 - 0x47DFFFFF

Above area includes following contents mainly:

- image(Trusted Firmware-A BL31 /OP-TEE OS /TA)
- above image’s work area
- above image’s log area

These protections are set by IPL at boot.
Please refer to References [3] for setting details.

2.3 Power Management

2.3.1.1 PSCI (Power State Coordinate Interface)

PSCI is standard interface to be related to power supply control of CPU among the software(such as Rich OS, Trusted Firmware-A BL31, and OP-TEE OS) running on each different Exception Level.

PSCI is mainly related to the following function.

- Core idle management
- Addition and separation of dynamic CPU core such as the CPU Hot Plug.
- Secondary CPU boot
- big.LITTLE migration
- System shutdown and reset

About the interface that PSCI provides for the Rich OS, Please refer to the References [5] for details.

It is representative control using PSCI in this system as follows.

1. CPU_ON

CPU_ON is used when CPU(Master) which has been already got up starts CPU(Slave). A use case includes secondary boot and Hot Plug which adds CPU's dynamically during normal operation.

Figure 2.6 shows the summary of procedure of PSCI CPU_ON.

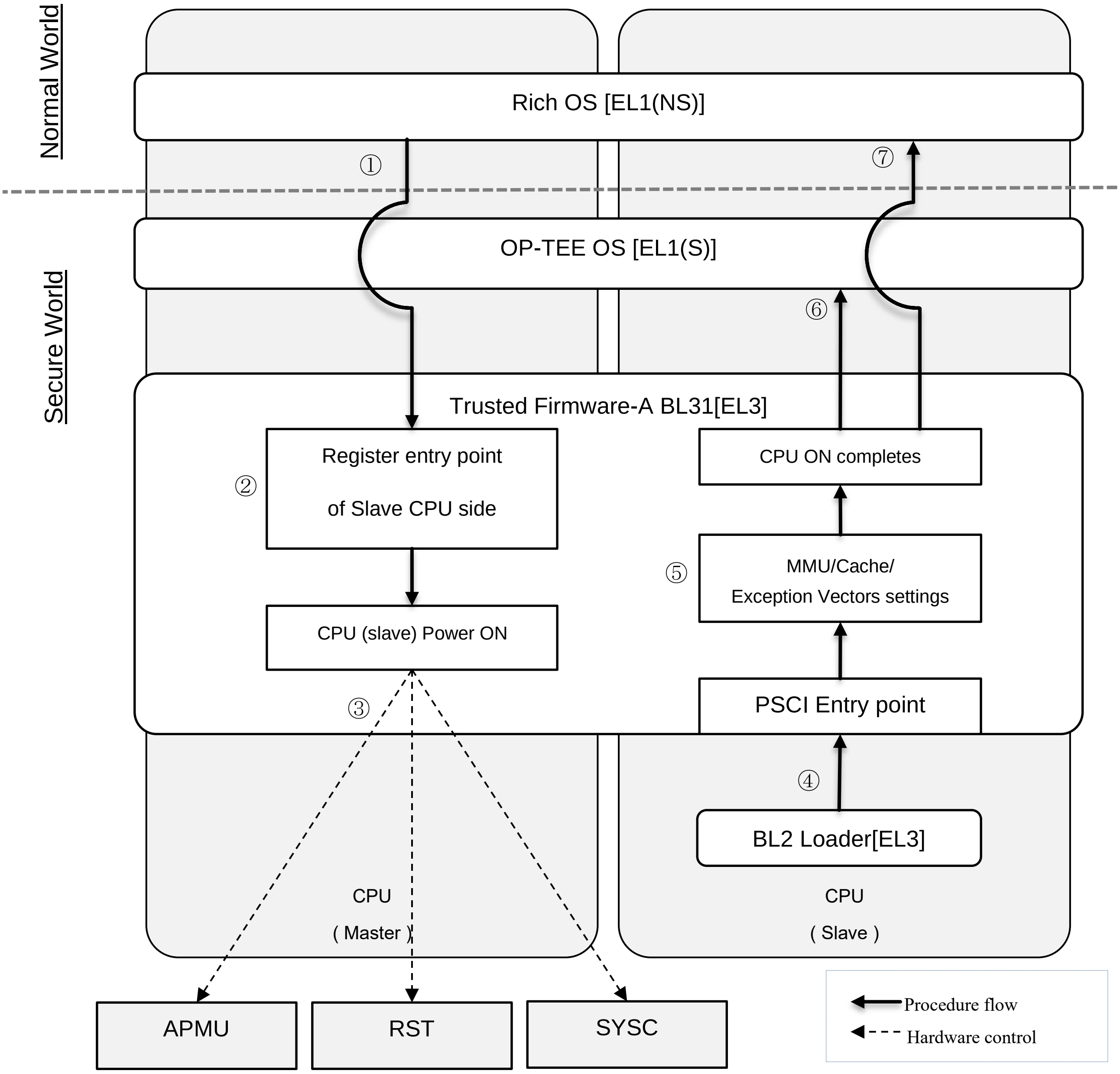

Figure 2.6 Summary of procedure of PSCI CPU_ON

It is the explanation of each number of Figure 2.6 as follows.

① Rich OS running on Master CPU requests PSCI CPU_ON to Trusted Firmware-A BL31. At this time, the Trusted Firmware-A BL31 is passed Entry point that Rich OS of the Slave CPU side is used when the power is ON.

② Trusted Firmware-A BL31 saves this Entry point in inside.

③ Master CPU controls APMU/RST/SYSC, and turns on Slave CPU.

④ BL2 Loader starts up in Slave CPU and kicks Trusted Firmware-A BL31. Then, the processing jumps to Entry point of PSCI. Since the primary CPU boot is not a cold boot, Trusted Firmware-A BL31 moves processing to process the warm reset processing.

⑤ Trusted Firmware-A BL31 sets MMU/Cache/Exception Vectors of Slave CPU.

⑥ OP-TEE OS in Slave CPU is initialized.

⑦ The processing jumps to Entry point for Slave CPU power on which is saved at ②, then Rich OS on Slave CPU starts up.

2. CPU_OFF

CPU_OFF is used to turn OFF the power of its own CPU.

Figure 2.7 shows the summary of procedure of PSCI CPU_OFF.

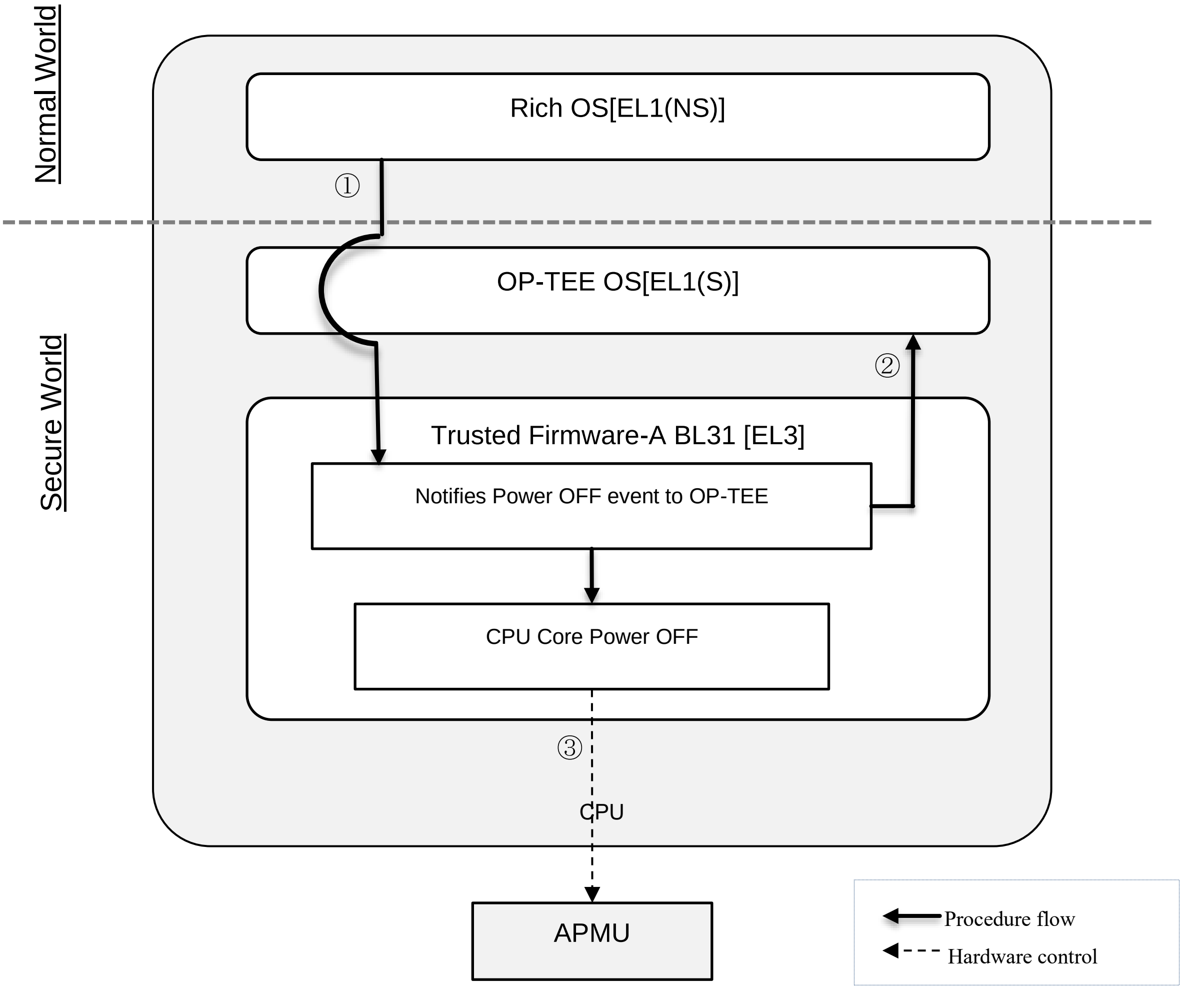

Figure 2.7 Summary of procedure of PSCI CPU_OFF

It is the explanation of each number of Figure 2.7 as follows.

① Rich OS requests PSCI CPU_OFF to Trusted Firmware-A BL31.

② Trusted Firmware-A BL31 notifies power supply OFF of the CPU of own to OP-TEE OS.

③ Trusted Firmware-A BL31 lets core standby (*) change in a power supply state of the CPU of own by controlling APMU.

(*)For the core standby, please refer to section of AP-System Core in the References [1] for details.

3. CPU_SUSPEND

CPU_SUSPEND is used when power state of its own CPU changes to power off state which can accept interrupts.

Figure 2.8 shows the summary of procedure of PSCI CPU_SUSPEND.

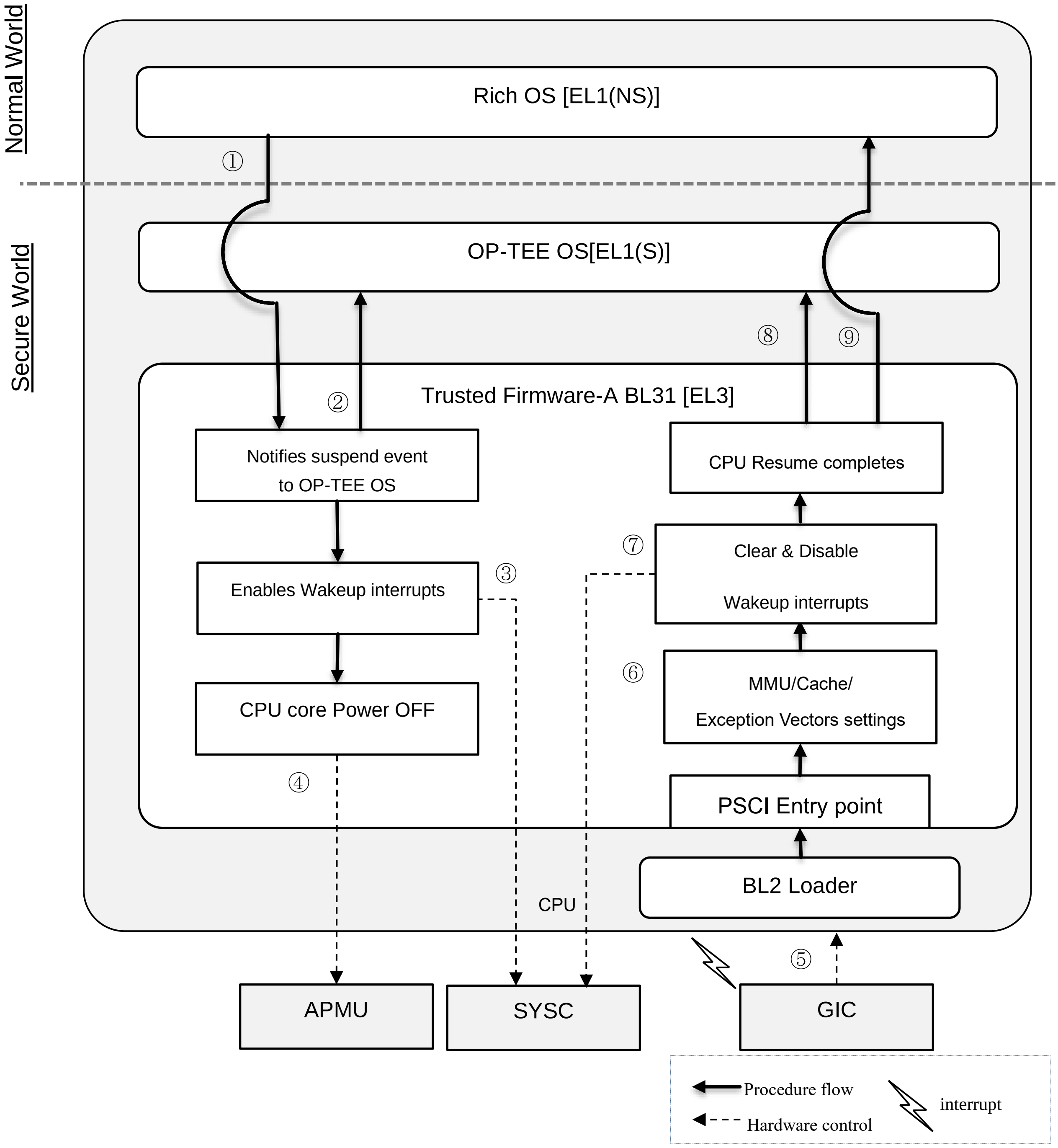

Figure 2.8 Summary of procedure of PSCI CPU_SUSPEND

It is the explanation of each number of Figure 2.8 as follows.

① Rich OS requests PSCI CPU_SUSPEND to Trusted Firmware-A BL31.

② Trusted Firmware-A BL31 notifies suspend event to OP-TEE OS.

③ Trusted Firmware-A BL31 enables an interrupt by controlling SYSC after OP-TEE OS response.

④ Trusted Firmware-A BL31 lets core standby (*) change in a power supply state of the CPU of own by controlling APMU.

The subsequent steps are the procedure at the time of the suspend release.

⑤ When an interrupt occurs, BL2 Loader starts up and kicks Trusted Firmware-A BL31. Then, the processing jumps to Entry point of PSCI.

⑥ Trusted Firmware-A BL31 sets MMU/Cache/Exception Vectors.

⑦ By controlling the SYSC, Trusted Firmware-A BL31 sets an invalid state to the interrupt state and clears the interrupt.

⑧ Trusted Firmware-A BL31 notifies resume event to OP-TEE. Then, OP-TEE starts resuming.

⑨ The processing jumps to Entry point of Rich OS, and then, Rich OS starts resuming.

(*)For the core standby, please refer to section of AP-System Core in the References [1] for details.

4. SYSTEM_OFF/SYSTEM_RESET

SYSTEM_OFF is used to shut down the system. Also, SYSTEM_RESET is used to reset the system. Since these procedure is almost the same, here describes them together.

Figure 2.9 shows the summary of procedure of PSCI SYSTEM_OFF/SYSTEM_RESET.

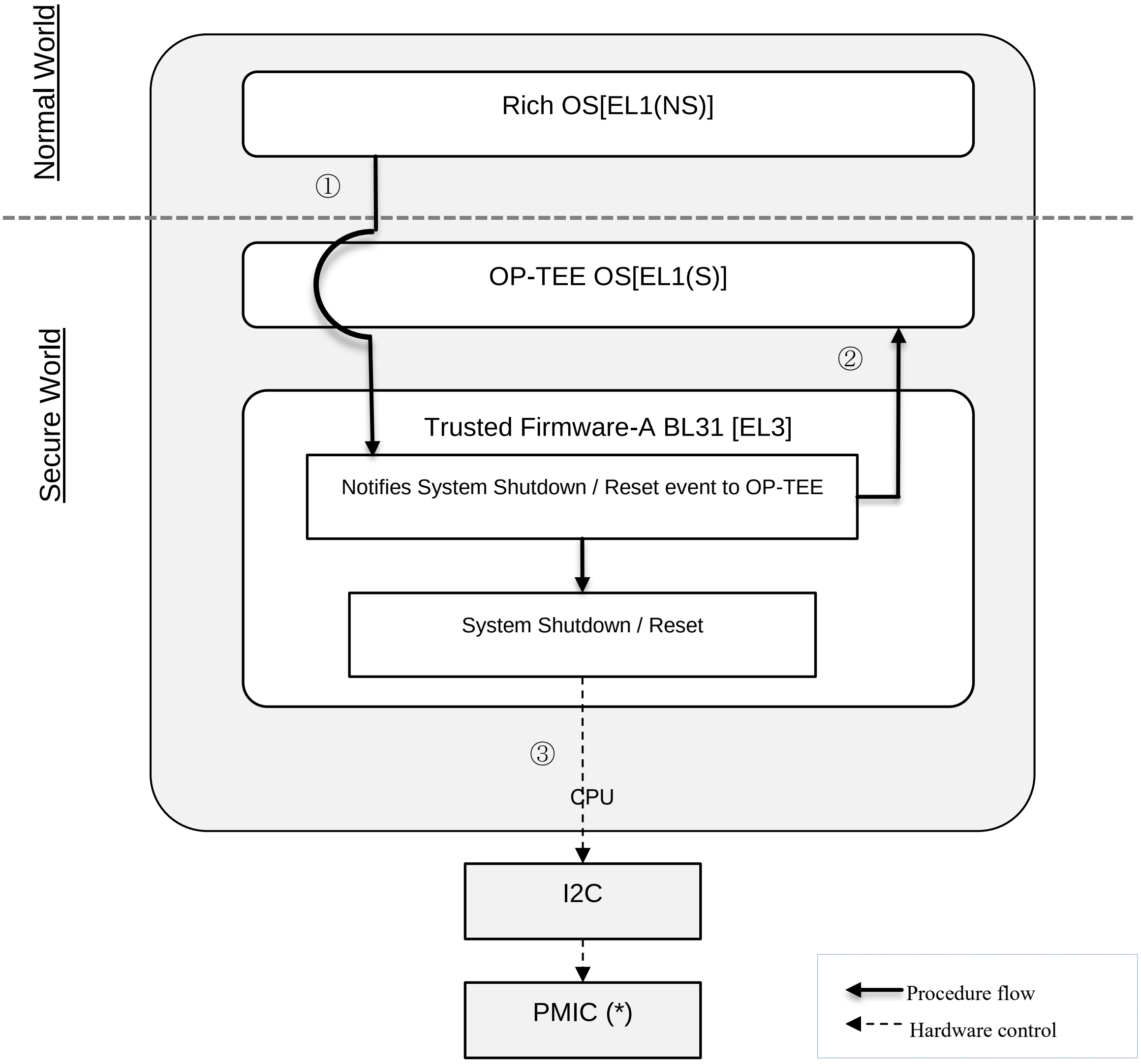

Figure 2.9 Summary of procedure of SYSTEM_OFF/SYSTEM_RESET

It is the explanation of each number of Figure 2.9 as follows.

① Rich OS requests SYSTEM_OFF/SYSTEM_RESET to Trusted Firmware-A BL31.

② Trusted Firmware-A BL31 notifies SYSTEM_OFF/SYSTEM_RESET to OP-TEE OS.

③ Trusted Firmware-A BL31 controls PMIC (*) via I2C for shutdown or reset system.

(*)This PMIC is for Salvator-X/Salvator-XS/Ebisu/Ebisu-4D which is RENESAS evaluation board.

2.4 Secure Storage

Secure Storage is a function to ensure confidentiality of data by encrypting confidential data and then writing to external storage device such as flash memory.

Secure Storage uses Secure Storage Key (SSK) and File Encryption Key (FEK).
SSK is per-device key and protecting FEK.
FEK is generated by PRNG (pseudo random number generator) and used for encrypting / decrypting the confidential data.
OP-TEE manages these key materials.
Please refer to "Secure Storage in OP-TEE" in References [9] for each key details

The following Figure 2.10 shows simplified image of Secure Storage.

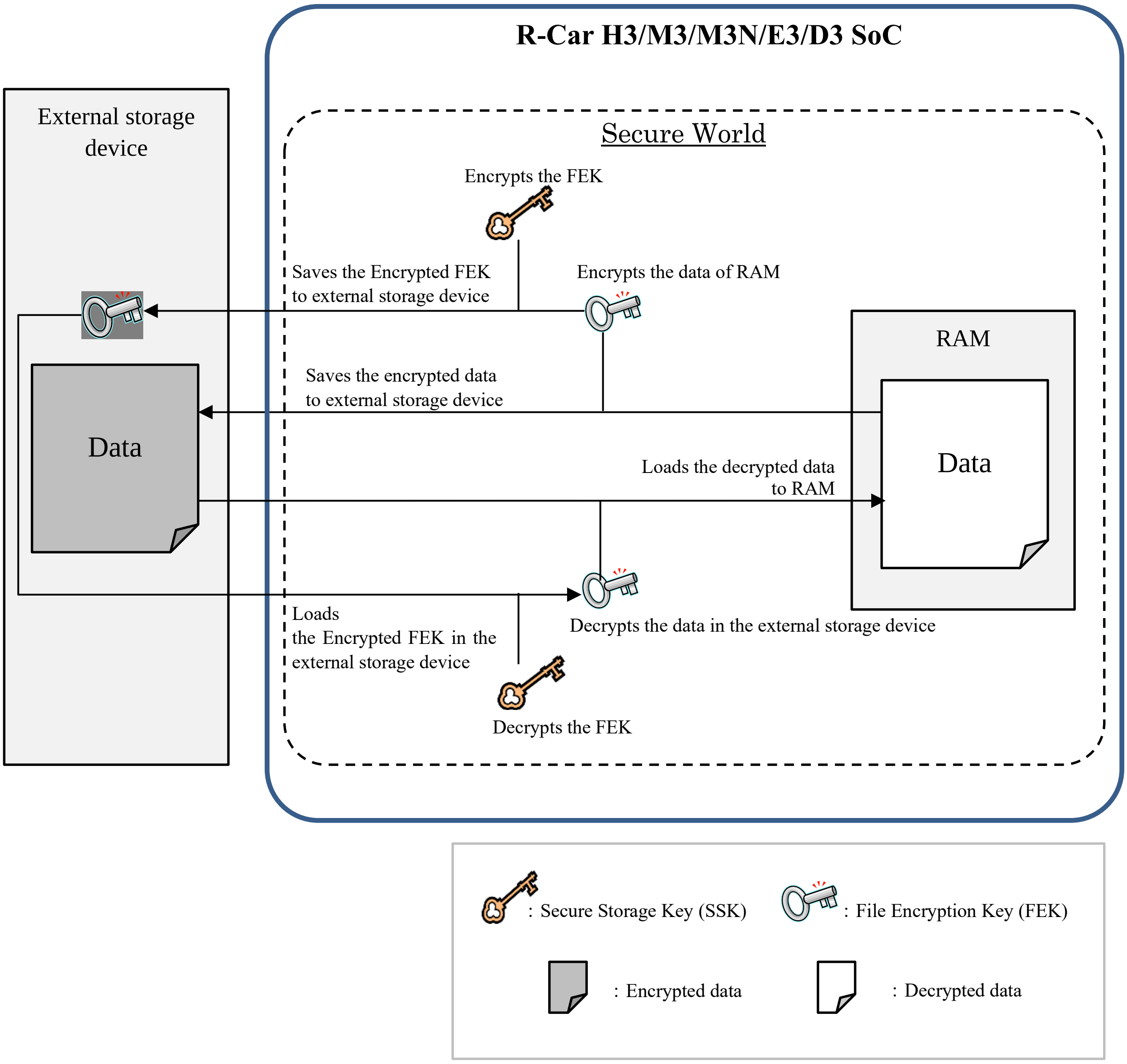
R-Car H3/M3/M3N/E3/D3 SoC
External storage device
Secure World
Encrypts the FEK
Saves the Encrypted FEK to external storage device
Encrypts the data of RAM
RAM
Saves the encrypted data to external storage device
Data
Loads the decrypted data to RAM
Data
Loads the Encrypted FEK in the external storage device
Decrypts the data in the external storage device
Decrypts the FEK
: Secure Storage Key (SSK)
: File Encryption Key (FEK)
: Encrypted data
: Decrypted data

Figure 2.10 Summary of processing of Secure Storage

For Secure Storage Key, see 2.4.1.
For external storage devices and their control to be handled in this system, see 2.4.2

2.4.1 Secure Storage Key(SSK)

Secure Storage Key(SSK) is a per-device key and managed in Secure World only.
However, currently OP-TEE standard generates this key by using fixed value.
Please refer to "Secure Storage in OP-TEE" in References [9] for details

2.4.2 Secure Storage in OP-TEE

There are following three Secure Storage implementations in OP-TEE in this system.

- The first one relies on the Rich OS file system. It is the default implementation.
- The second one makes use of the Replay Protected Memory Block (RPMB) partition of an eMMC device.
- The third one makes use of the SPI/HyperFlash device.

Only one storage location in the above is used as Secure Storage, so these cannot coexist.
The first and second one are provided by OP-TEE standard.
Please refer to "Secure Storage in OP-TEE" in References [9] for the first one details.
Please refer to "RPMB Secure Storage" in References [9] for the second one details.
Finally, the third one is provided by this system.

By request from CA, TA calls Trusted Storage API to use Secure Storage.
If Secure Storage is the first or second one, TA can access the storage via TEE Supplicant in Normal World side.
If Secure Storage is the third one, TA can accessed the storage via SPI/HyperFlash driver in Secure World side.

The following two figures show the flow of processing from requests from CA to access the Secure Storage.
Figure 2.11 shows the case of first and second Secure Storage. Then, Figure 2.12 shows the case of third Secure Storage.

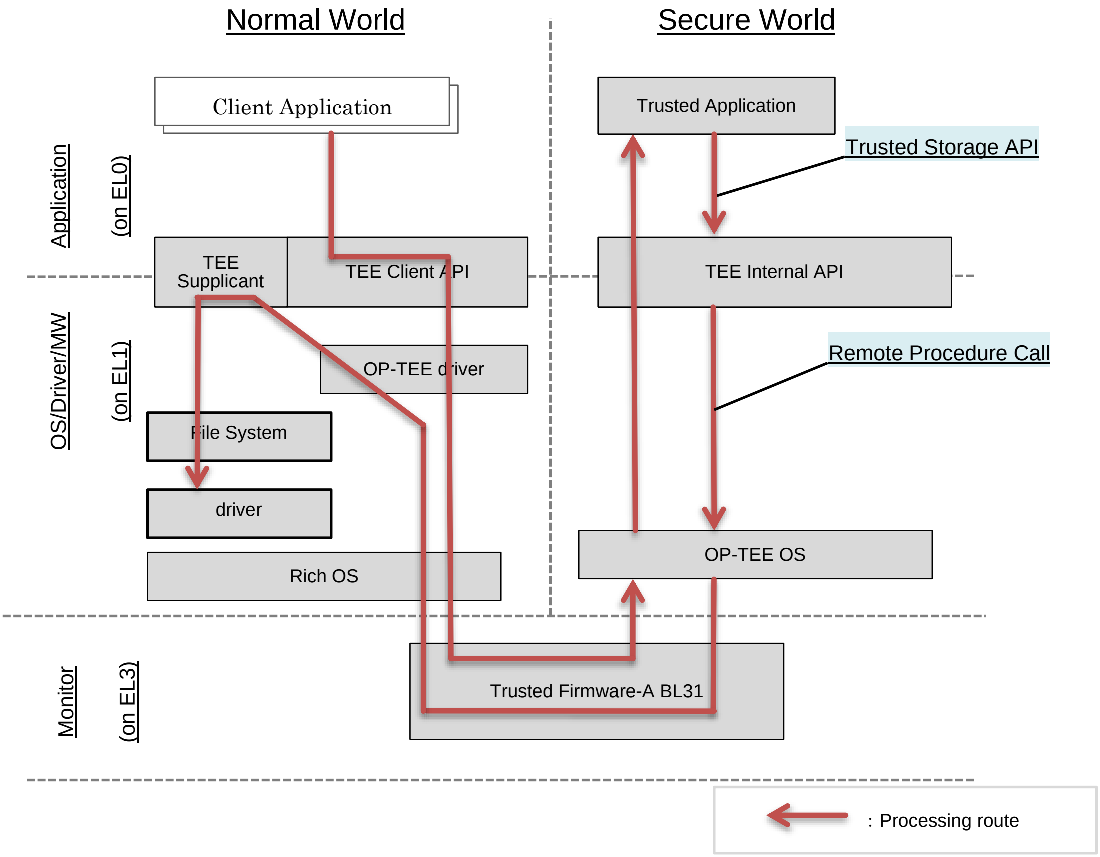

Figure 2.11 Flow of processing of Secure Storage (File System/RPMB)

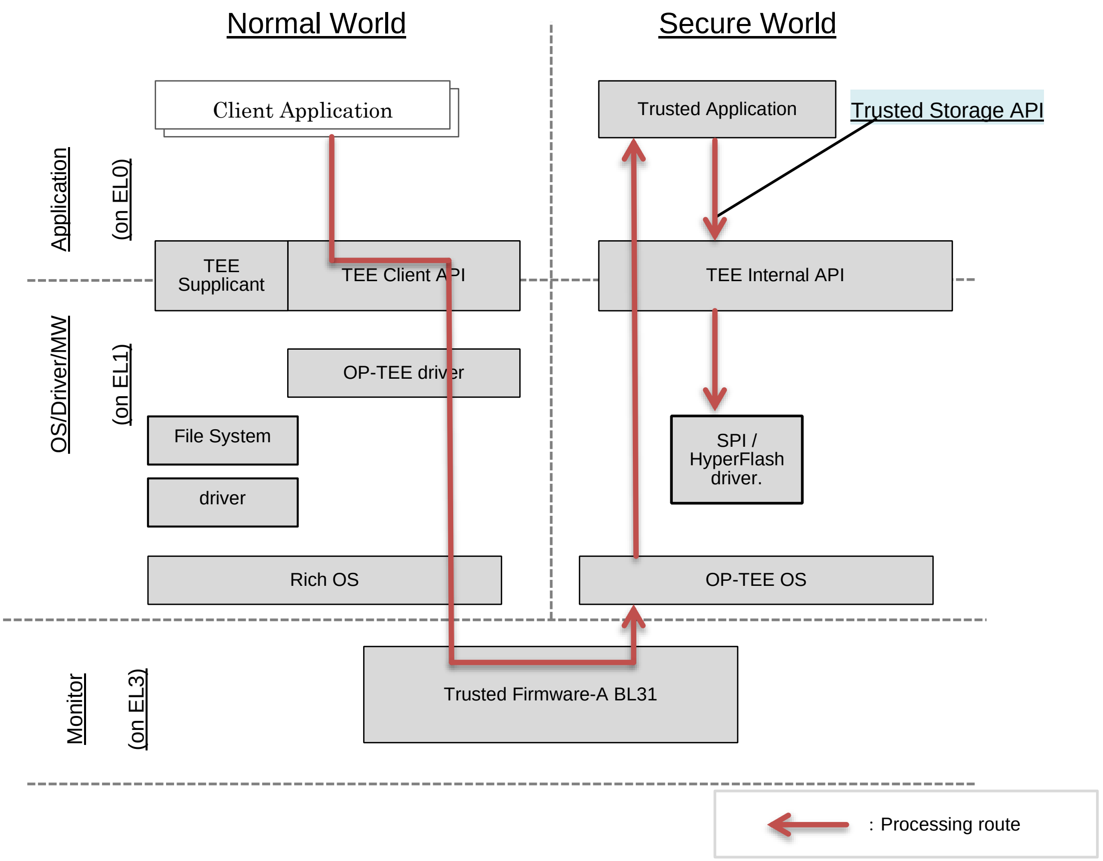

Figure 2.12 Flow of processing of Secure Storage (SPI/HyperFlash)

Also, in each devices, there is feature and restriction.
Following Table 2.12 shows features and restriction of device used as Secure Storage.

Table 2.12 Features and Restrictions of device used as Secure Storage

Device	Feature	Restriction
File system	relies on applied file system	relies on applied file system
RPMB	implements countermeasure of replay attack	can not be specified data size of more than communication buffer between Normal World and Secure World.
SPI/HyperFlash	implements rollback function for countermeasure of power off during writing data.	can not be specified data size more than sector size of HyperFlash.

REVISION HISTORY	Security Board Support Package System Architecture Design Guide: Software

Rev.	Date	Description	
		Page	Summary
1.0.0	Apr. 27, 2016	—	New creation.
1.0.1	Jan. 13, 2017	11	Modified MMU settings.
1.0.2	Mar. 15, 2017	11	Change the address and size of the following area of Table 2.6. ・TA image including heap & stack ・log (for Secure World) ・Stand-alone FS Work area ・log (for Normal World) ・MaskROM API
		15	Changed the abbreviation and summary of the following IP in Table 2.10. ・SWDT ・SCMT ・SUCMT
1.0.3	Jul. 12, 2017	－	Fixed the format of the document (trademark, etc.)
1.0.4	Jan. 12, 2018	－	Fixed the format of the document (trademark, etc.)
		－	R-Car M3N support.
		5	Change the Static TA to the pseudo TA.
		8	Change the Static TA to the pseudo TA by a name of TA Type of Table2.3. Add early TA to Table2.3.
		11	Change the address and size of the following area of Table 2.5. ・System RAM for driver of I2C for DVFS ・stack area for driver of I2C for DVFS Change the address and size of the following area of Table 2.6. ・image including heap & stack ・TA image including heap & stack
1.0.5	Apr. 11, 2018	－	Change the format of Cover.
		－	R-Car E3 support.
1.0.6	Jun. 11, 2018	11	Change the address and size of the following area of Table 2.6. ・shared memory between Secure World and Normal World Delete the following area of Table 2.6. ・log (for Normal World)
1.0.7	Oct. 12, 2018	－	Ebisu-4D support.
		11	Change the address of the following area of Table 2.5. ・stack area for driver of I2C for DVFS
2.0.0	Nov. 26, 2018	－	Fixed the format of the document (Address List.)
		11	Change the address of the following area of Table 2.5. ・System RAM for driver of I2C for DVFS ・stack area for driver of I2C for DVFS Change the following area of Table 2.6. ・shared memory between Secure World and Normal World Add the following area of Table 2.6. ・log (for Normal World)
3.0.2	Apr. 6, 2021	－	R-Car D3 support.
		－	Changed the software name from ARM Trusted Firmware to Trusted Firmware-A.
		－	Changed ARM notation to Arm.
3.0.3	Aug. 16, 2021	－	Fixed the format of the document (Notice, Address List.)
		－	Add information of Gen3e.
		1	Changed the URL and Edition of OP-TEE document.

Security Board Support Package
System Architecture Design Guide: Software

Publication Date: Rev.1.0.0 Apr. 27, 2016
Rev.3.0.3 Aug. 16, 2021

Published by: Renesas Electronics Corporation

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

Renesas Electronics America Inc. San Jose Campus
6024 Silver Creek Valley Road, San Jose, CA 95138, USA
Tel: +1-408-284-8200, Fax: +1-408-284-2775

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

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 101-T01, Floor 1, Building 7, Yard No. 7, 8th Street, Shangdi, Haidian District, Beijing 100085, 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, 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, #06-02 Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit No 3A-1 Level 3A Tower 8 UOA Business Park, No 1 Jalan Pengaturcara U1/51A, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia
Tel: +60-3-5022-1288, Fax: +60-3-5022-1290

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700

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

Colophon 7.0

ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

http://www.renesas.com

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社　〒135-0061　東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：https://www.renesas.com/contact/

Colophon 3.1

Security Board Support Package
System Architecture Design Guide

Renesas Electronics Corporation