

Linux Interface Specification Device Driver PCIe

User's Manual: Software

RZ/V2N Group, RZ/G3E Group, RZ/G3S Group,
and RZ/V2H Group

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>).

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. 1.04 Mar. 27, 2026)

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/.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MPU. It is intended for users designing application systems incorporating the MPU. A basic knowledge of electric circuits, logical circuits, and MPUs is necessary in order to use this manual.

The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and electrical characteristics; and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the RZ/V2N Group, RZ/G3E Group, RZ/G3S Group, and RZ/V2H Group. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

Document Type	Description	Document Title	Document No.
User's manual for Hardware	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description Note: Refer to the application notes for details on using peripheral functions.	RZ/V2N Group User's Manual: Hardware	---
		RZ/G3E Group User's Manual: Hardware	---
		RZ/G3S Group User's Manual: Hardware	---
		RZ/V2H Group User's Manual: Hardware	---
User's manual for Software	Description of PCIe Linux interface Specification	Linux Interface Specification Device Driver PCIe	This user's manual
Application Note	Information on using peripheral functions and application examples Sample programs Information on writing programs in assembly language and C	Available from Renesas Electronics Web site.	
Renesas Technical Update	Product specifications, updates on documents, etc.		

2. Notation of Numbers and Symbols

3. Register Notation

4. List of Abbreviations and Acronyms

Abbreviation	Full Form
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect Express
MSI	Message Signaled Interrupt
EP	Endpoint
RC	Root Complex
BAR	Base Address Register
DMA	Direct Memory Access

Table of Contents

1. Overview.....	1
1.1 Overview	1
1.2 Function	1
1.2.1 Root Complex.....	1
1.2.2 Endpoint	1
1.3 Connected Port	1
1.4 Related Document	2
1.4.1 Standard.....	2
1.4.2 Related document	2
1.5 Restrictions	2
2. Terminology.....	3
3. Operating Environment.....	4
3.1 Hardware Environment	4
3.2 Module Configuration	5
3.2.1 Root Complex Mode	5
3.2.2 Endpoint Mode	6
3.3 PCIe Address Map space	7
3.4 Device ID (Root Complex - Endpoint).....	7
4. External Interface.....	8
4.1 Device access	8
4.2 Definitions	8
4.2.1 Root Complex.....	8
4.2.1.1 SoC Device-tree	8
4.2.1.2 Board Device-tree	20
4.2.2 Endpoint	22
4.2.2.1 SoC Device-tree	22
5. Integration.....	25
5.1 Directory Configuration	25
5.2 Integration Procedure	25
5.2.1 Kernel Configuration for Root Complex	25
5.2.2 Kernel Configuration for Endpoint.....	26
5.3 Option Setting	27
5.3.1 Module Parameters	27
5.3.2 Kernel Parameters	27

1. Overview

1.1 Overview

This manual explains the driver module (this module) that controls the PCIe 3.0 Controller on RZ/V2N Group, RZ/G3E Group, and RZ/V2H Group, and the PCIe 2.0 controller on RZ/G3S Group.

Note: Currently, the device can support the following Linux kernel version:

- rz-6.1-cip28: RZ/G3E Group.
- rz-6.1-cip43: RZ/G3S Group, RZ/V2H Group and RZ/V2N Group.

1.2 Function

1.2.1 Root Complex

This module controls the PCIe Controller on RZ/V2N Group, RZ/G3E Group, RZ/G3S Group and RZ/V2H Group acting as a PCIe Host Controller. It allows PCIe Endpoint devices to be used with RZ/V2N, RZ/G3E, RZ/G3S and RZ/V2H Board. The following functionality is supported:

- Configuration read and writes.
- Host initiated memory reads and writes.
- Slave initiated memory reading and writing (bus mastering).
- MSI interrupts.

1.2.2 Endpoint

This module controls the PCIe Controller on RZ/G3E Group, RZ/V2H Group and RZ/V2N acting as a PCIe Endpoint Controller. It allows RZ/G3E, RZ/V2H and RZ/V2N Board to be used as PCIe Endpoint devices. The following functionality is supported:

- Configuration Base Address Register (BAR)
- Read/Write/Copy data.
- MSI Interrupt.
- DMA Support.

1.3 Connected Port

The supported connectors of this module are listed below.

Table 1-1 Connected Port

Target boards	Channel	Connected to	Support status	Remark
RZ/V2N-EVK	PCIE0	PCIE1	YES	
RZ/V2H-EVK	PCIE0	PCIE1	YES	

PCIe RC drivers can control the Endpoint device through 3 interfaces connectors on RZ/G3E SMARC.

Table 1-2 Connected Port (RZ/G3E SMARC Board)

Channel	Connected to	Support status	Remark
PCIE0	PCI SLOT	YES	
PCIE0	M2 KEY-B	YES	
PCIE0	M2 KEY-E	NO	

Table 1-3 Connected Port (RZ/G3S SMARC Board)

Channel	Connected to	Support status	Remark
PCIE0	PCI SLOT	YES	Configure SW_PCIE_MUX[0:2] = 100

1.4 Related Document

The data relevant to this module is as follows.

1.4.1 Standard

Table 1-4 Standard

Number	Issue	Title	Edition	Date
-	PCI-SIG	PCI Express Base Specification	Rev.3.0	Nov 10, 2010

1.4.2 Related document

There is no related document for this module.

1.5 Restrictions

This module only uses legacy memory area, because IPMMU is not implemented.

2. Terminology

The following table shows the terminology related to this module.

Table 2-1 Terminology

Terms	Explanation
MSI	Message Signaled Interrupt
PCI	Peripheral Component Interconnect
PCIe	PCI Express
RC	Root Complex
EP	Endpoint
BAR	Base Address Register
DMA	Direct Memory Access

3. Operating Environment

3.1 Hardware Environment

The following table lists the hardware needed to use this module.

Table 3-1 Hardware Environment

Name	Product number
RZ/G3S Evaluation Board Kit	RTK9845S33C01000BE
RZ/G3E Evaluation Board Kit	RTK9947E57S01000BE
RZ/V2H Evaluation Board Kit	RTK0EF0168C04000BJ
RZ/V2N Evaluation Board Kit V1.0	RTK0EF0186C03000BJ
RZ/V2N Evaluation Board Kit V2.0	RTK0EF0186C03001BJ

3.2 Module Configuration

The following figure shows the configuration of this module.

3.2.1 Root Complex Mode.

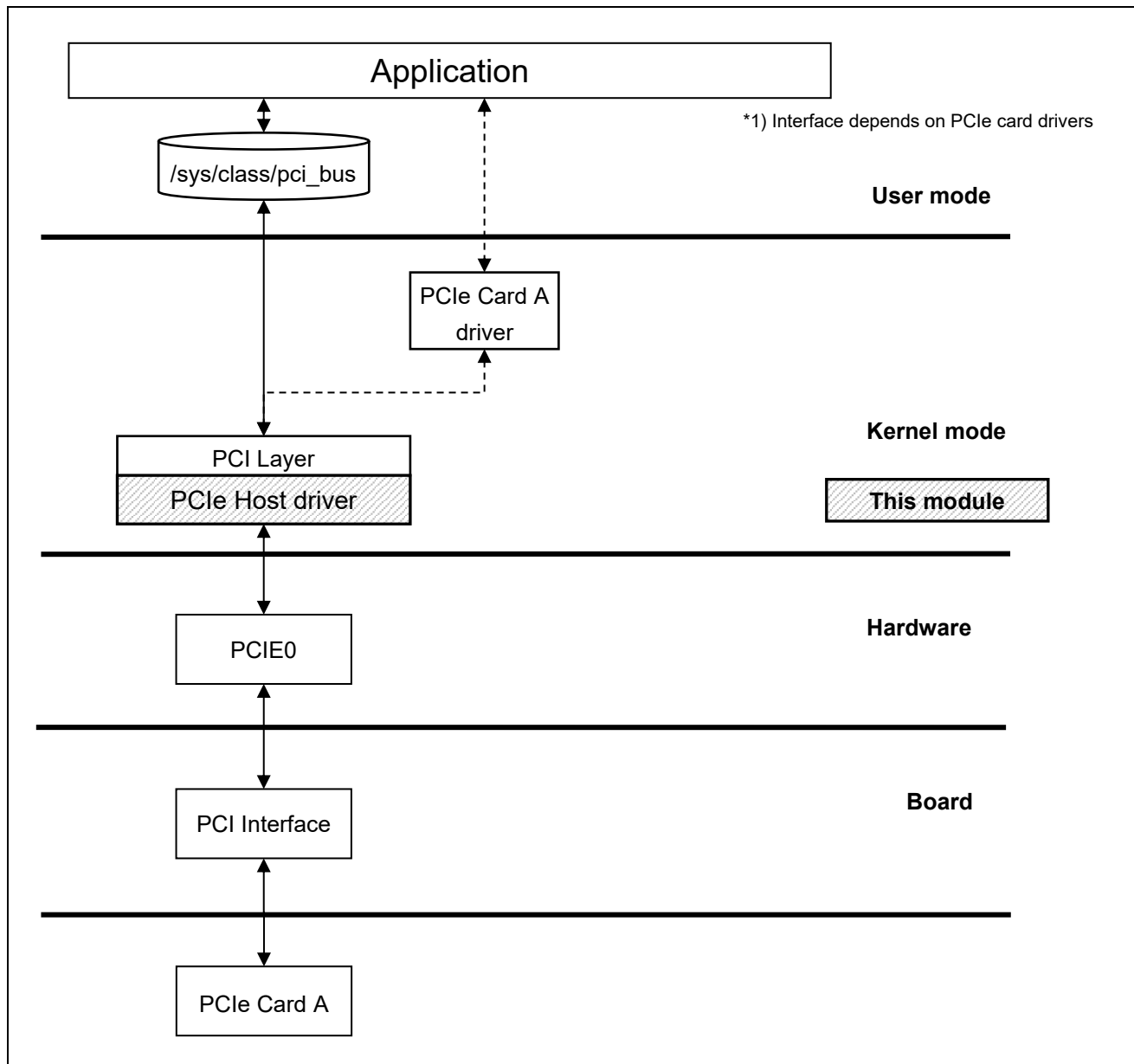


Figure 3-1 Module configuration

- **Target boards:** RZ/G3E SMARC, RZ/G3S SMARC, RZ/V2H and RZ/V2N EVK-VER1/VER2 Boards

3.2.2 Endpoint Mode.

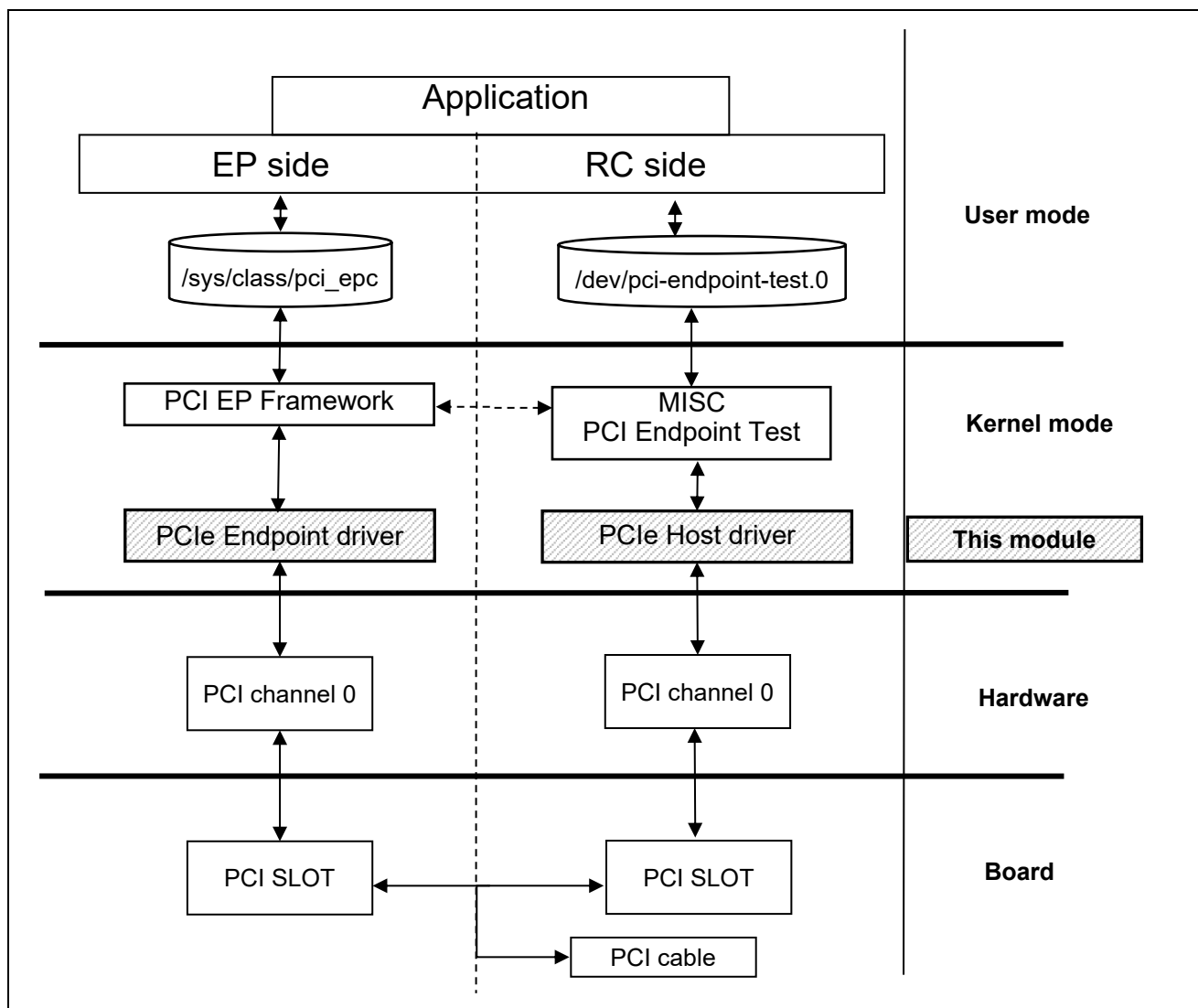


Figure 3-2 Module configuration

3.3 PCIe Address Map space

The Linux PCI sub-system will assign local addresses within ranges provided by the PCIe host controller (e.g. PCIEC) driver. The card can be access by “Window” memory. For the RZ/V2N, RZ/G3E, RZ/G3S, and RZ/V2H Board the following table details these windows.

Table 3-2 Hardware Environment

Target boards	Local Address	Channel	PCle access type
RZ/V2N-EVK- VER1/VER2	0x3000 0000 to 0x37FF FFFF	PCIE0	32-bit Memory
RZ/G3E SMARC	0x0_3000_0000 to 0x0_37FF_FFFF 0x4_4000_0000 to 0xA_4000_0000	PCIE0	32-bit Memory 64-bit Memory
RZ/G3S SMARC	0x3000 0000 to 0x37FF FFFF	PCIE0	32-bit Memory
RZ/V2H-EVK- VER1/VER2	0x3000 0000 to 0x37FF FFFF	PCIE0	32-bit Memory

3.4 Device ID (Root Complex - Endpoint)

The PCI device ID, vendor ID for RZ/V2N Group, RZ/G3E Group, RZ/G3S Group and RZ/V2H Group:

Table 3-3 Hardware Configuration

Board	Vendor ID	RC device ID	EP device ID
RZ/V2N	0x1912	0x003b	0x003a
RZ/G3E	0x1912	0x0039	0x0038
RZ/G3S	0x1912	0x0033	--
RZ/V2H	0x1912	0x003b	0x003a

4. External Interface

4.1 Device access

The external interface of this module is based on Linux.

The PCI sub-system provides a number of controls that can be accessed via /sys/class/pci_bus. Please see <https://www.kernel.org/doc/Documentation/filesystems/sysfs-pci.txt>.

4.2 Definitions

4.2.1 Root Complex

4.2.1.1 SoC Device-tree

(1) RZ/G3E SoC

Supported Kernel: rz-6.1-cip28

The PCIe RC device properties for RZ/G3E SoC are shown below.

```
pcie0: pcie@13400000 {
    compatible = "renesas,rzg3e-pcie";
    device_type = "pci";
    reg = <0 0x13400000 0 0x10000>;
    clocks = <&cpg CPG_MOD 0xc4>,
            <&cpg CPG_MOD 0xc5>;
    power-domains = <&cpg>;
    resets = <&cpg 0xb2>;
    ranges = <0x02000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>,
            <0x42000000 0x4 0x48000000 0x4 0x48000000 0x1 0x00000000>;
    dma-ranges = <0x42000000 0 0x48000000 0 0x48000000 0 0xf8000000>;
    interrupts = <GIC_SPI 796 IRQ_TYPE_LEVEL_HIGH>;
    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &gic GIC_SPI 796 IRQ_TYPE_LEVEL_HIGH>;
    bus-range = <0x0 0xff>;
    linux,pci-domain = <0x0>;
    #address-cells = <3>;
    #size-cells = <2>;
    pcie,channel = <0>;
    renesas,pcie-sys = <&sys>;
    status = "disabled";
};
```

Figure 4-1 Example PCIe channel 0 Root Complex (RZ/G3E)

Note: All the information in the above device tree example is commonly used for RZ/G3E boards except for “compatible”, “clock”, and “reset” as described below.

The PCIe device required properties:

compatible:

Must be set to "renesas,rzg3e-pcie" for R9A09G047 (RZ/G3E).

device_type:

Must be set to “pci”

reg:

Base address and length of the memory resource used by the PCIe CH0.

clocks:

slot 1:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to GPG_MOD clock index

slot 2:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to GPG_MOD clock index

reset:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to GPG_MOD reset index

Ranges: support 32 bit area and 64 bit area.

For RZ/G3E, must be set:

<0x02000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>,

<0x42000000 0x4 0x40000000 0x4 0x48000000 0x1 0x00000000>,

dma-ranges: a single range for the inbound memory region.

For RZ/G3E, must be set: <0x42000000 0 0x48000000 0 0x48000000 0 0xF8000000>;

pcie,channel:

Must be set to “0” to choose channel 0 in probe.

interrupt:

interrupt for PCIE device

#interrupt-cells: must be 1.

interrupt-map: standard property used to define the mapping of the PCI interrupts to the GIC interrupts.

interrupt-map-mask: standard property that helps to define the interrupt mapping.

bus-range:

The PCI bus number range; as this is a single bus, the range should be specified as the same value twice.

renesas,pcie-sys:

For RZ/G3E only, must be set <&sys> to system control setting for PCI.

```
sys: system-controller@10430000 {  
    compatible = "renesas,r9a09g047-sys", "syscon";  
    reg = <0 0x10430000 0 0x10000>;  
    clocks = <&cpg CPG_CORE R9A09G047_SYS_0_PCLK>;  
    resets = <&cpg 48>;  
    #renesas,sysc-signal-cells = <2>;  
};
```

Figure 4-2 Example SYS node (RZ/G3E SMARC)

(2) RZ/G3S SoC

The PCIe RC device properties for RZ/G3S are shown below. (**supported kernel: rz-6.1-cip43**)

```
pcie: pcie@11e40000 {
    compatible = "renesas,r9a08g045s33-pcie";
    reg = <0 0x11e40000 0 0x10000>;
    ranges = <0x02000000 0 0x30000000 0 0x30000000 0 0x80000000>;
    /* Map all possible DRAM ranges (4 GB). */
    dma-ranges = <0x42000000 0 0x40000000 0 0x40000000 0x1 0x0>;
    bus-range = <0x0 0xff>;
    clocks = <&cpg CPG_MOD R9A08G045_PCI_ACLK>,
            <&cpg CPG_MOD R9A08G045_PCI_CLKL1PM>;
    clock-names = "aclk", "pm";
    resets = <&cpg R9A08G045_PCI_ARESETN>,
            <&cpg R9A08G045_PCI_RST_B>,
            <&cpg R9A08G045_PCI_RST_GP_B>,
            <&cpg R9A08G045_PCI_RST_PS_B>,
            <&cpg R9A08G045_PCI_RST_RSM_B>,
            <&cpg R9A08G045_PCI_RST_CFG_B>,
            <&cpg R9A08G045_PCI_RST_LOAD_B>;
    reset-names = "aresetn", "rst_b", "rst_gp_b", "rst_ps_b",
                  "rst_rsm_b", "rst_cfg_b", "rst_load_b";
    interrupts = <GIC_SPI 395 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 396 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 397 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 398 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 399 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 400 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 401 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 402 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 403 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 404 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 405 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 406 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 407 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 408 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 409 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 410 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "serr", "serr_cor", "serr_nonfatal",
                      "serr_fatal", "axi_err", "inta",
                      "intb", "intc", "intd", "msi",
                      "link_bandwidth", "pm_pme", "dma",
                      "pcie_evt", "msg", "all";
    #interrupt-cells = <1>;
}
```

```

        interrupt-controller;
        interrupt-map-mask = <0 0 0 7>;
        interrupt-map = <0 0 0 1 &pcie 0 0 0 0>, /* INT A */
                        <0 0 0 2 &pcie 0 0 0 1>, /* INT B */
                        <0 0 0 3 &pcie 0 0 0 2>, /* INT C */
                        <0 0 0 4 &pcie 0 0 0 3>; /* INT D */

        device_type = "pci";
        num-lanes = <1>;
        #address-cells = <3>;
        #size-cells = <2>;
        power-domains = <&cpg>;
        vendor-id = <0x1912>;
        device-id = <0x0033>;
        renesas,sysc = <&sysc>;
        status = "disabled";

    };
};

```

Figure 4-3 Example PCIe Root Complex (RZ/G3S SMARC)

The PCIe device required properties:

compatible:

Must be set to "renesas,r9a08g045s33-pcie" for R9A08G045 (RZ/G3S).

device_type:

Must be set to "pci"

reg:

Base address and length of the memory resource used by the PCIe CH0.

clocks:

slot 1:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to R9A08G045_PCI_ACLK

slot 2:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to R9A08G045_PCI_CLKL1PM

reset:

slot 1:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_ARESETN

slot 2:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_B

slot 3:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_GP_B

slot 4:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_PS_B

slot 5:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_RSM_B

slot 6:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_CFG_B

slot 7:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to R9A08G045_PCI_RST_LOAD_B

Ranges: support 32 bit area

For RZ/G3S, must be set: <0x02000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>

dma-ranges: a single range for the inbound memory region.

For RZ/G3S, must be set: <0x42000000 0 0x40000000 0 0x40000000 0x1 0x0>;

interrupt:

interrupt for PCIE device

#interrupt-cells: must be 1.

interrupt-map: standard property used to define the mapping of the PCI interrupts to the GIC interrupts.

interrupt-map-mask: standard property that helps to define the interrupt mapping.

bus-range:

The PCI bus number range; as this is a single bus, the range should be specified as the same value.

renesas,sysc:

Specific property for RZ/G3S; must be set to <&sysc> which is reference to the System control node

```
sysc: system-controller@11020000 {
    compatible = "renesas,r9a08g045-sysc", "syscon";
    reg = <0 0x11020000 0 0x10000>;
    interrupts = <GIC_SPI 39 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 40 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 41 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 42 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "lpm_int", "ca55stbydone_int",
                    "cm33stbyr_int", "ca55_deny";
    #renesas,sysc-signal-cells = <2>;
    status = "disabled";
};
```

Figure 4-4 Example SYS node (RZ/G3S SMARC)

(3) RZ/V2H SoC

The PCIe RC device properties for RZ/V2H are shown below. (**supported kernel: rz-6.1-cip43**)

```
pcie0: pcie@13400000 {
    compatible = "renesas,r9a09g057-pcie";
    reg = <0 0x13400000 0 0x10000>;
    ranges = <0x03000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>;
    /* Map all possible DRAM ranges (16 GB). */
    dma-ranges = <0x42000000 0 0x40000000 0 0x40000000 0x4 0x00000000>;
    bus-range = <0x0 0xff>;
    clocks = <&cpg CPG_MOD 0xc4>,
            <&cpg CPG_MOD 0xc5>;
    clock-names = "aclk", "pmu";
    resets = <&cpg 0xb2>;
    reset-names = "aresetn";
    interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 792 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 793 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 794 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 795 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 796 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 797 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 799 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 800 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 801 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 802 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 803 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 806 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 807 IRQ_TYPE_LEVEL_HIGH>;
}
```

```

        interrupt-names = "all", "inta", "intb", "intc",
                           "intd", "msi", "link_bandwidth",
                           "pm_pme", "serr", "serr_cor",
                           "serr_nonfatal", "serr_fatal", "dma",
                           "pcie_evt", "axi_err", "msg";

    #interrupt-cells = <1>;

    interrupt-controller;

    interrupt-map-mask = <0 0 0 7>;

    interrupt-map = <0 0 0 1 &pcie0 0 0 0 0>, /* INT A */
                   <0 0 0 2 &pcie0 0 0 0 1>, /* INT B */
                   <0 0 0 3 &pcie0 0 0 0 2>, /* INT C */
                   <0 0 0 4 &pcie0 0 0 0 3>; /* INT D */

    device_type = "pci";

    num-lanes = <4>;

    #address-cells = <3>;

    #size-cells = <2>;

    power-domains = <&cpg>;

    vendor-id = <0x1912>;

    device-id = <0x003b>;

    pcie,channel = <0>;

    renesas,sysc = <&sys>;

    status = "disabled";

};

```

Figure 4-5 Example PCIe Root Complex (RZ/V2H EVK)

The PCIe device required properties:

compatible:

Must be set to "renesas,r9a09g057-pcie" for R9A09G057 (RZ/V2H Group).

device_type:

Must be set to "pci"

reg:

Base address and length of the memory resource used by the PCIe0.

clocks:

slot 1:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to PCIE_0_ACLK

slot 2:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to PCIE_0_CLK_PMU

reset:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to PCIE_0_ARESETN

ranges: support 32-bit area

For RZ/V2H, must be set: <0x30000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>

dma-ranges: a single range for the inbound memory region.

For RZ/V2H, must be set: <0x42000000 0 0x40000000 0 0x40000000 0x4 0x00000000>

interrupt:

interrupt for PCIe device

#interrupt-cells: must be 1.

interrupt-map: standard property used to define the mapping of the PCI interrupts to the GIC interrupts.

interrupt-map-mask: standard property that helps to define the interrupt mapping.

bus-range:

The PCI bus number range; as this is a single bus, the range should be specified as the same value.

renesas,sysc:

Specific property for RZ/V2H; must be set to <&sys> which is reference to the System control node

```

sys: system-controller@10430000 {
    compatible = "renesas,r9a09g057-sys", "syscon";
    reg = <0 0x10430000 0 0x10000>;
    clocks = <&cpg CPG_CORE R9A09G057_SYS_0_PCLK>;
    resets = <&cpg 0x30>;
};

```

Figure 4-6 Example SYS node (RZ/V2H-EVK)

(4) RZ/V2N SoC

The PCIe RC device properties for RZ/V2N are shown below. (**supported kernel: rz-6.1-cip43**)

```

pcie0: pcie@13400000 {
    compatible = "renesas,r9a09g056-pcie";
    reg = <0 0x13400000 0 0x10000>;
    ranges = <0x03000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>;
    /* Map all possible DRAM ranges (8 GB). */
    dma-ranges = <0x42000000 0 0x40000000 0 0x40000000 0x2 0x00000000>;
    bus-range = <0x0 0xff>;
    clocks = <&cpg CPG_MOD 0xc4>,
            <&cpg CPG_MOD 0xc5>;
    clock-names = "aclk", "pmu";
    resets = <&cpg 0xb2>;
    reset-names = "aresetn";
    interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 792 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 793 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 794 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 795 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 796 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 797 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 799 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 800 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 801 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 802 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 803 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 806 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 807 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "all", "inta", "intb", "intc",
                    "intd", "msi", "link_bandwidth",
                    "pm_pme", "serr", "serr_cor",
                    "serr_nonfatal", "serr_fatal", "dma",
                    "pcie_evt", "axi_err", "msg";
    #interrupt-cells = <1>;
    interrupt-controller;
    interrupt-map-mask = <0 0 0 7>;
    interrupt-map = <0 0 0 1 &pcie0 0 0 0 0>, /* INT A */
                  <0 0 0 2 &pcie0 0 0 0 1>, /* INT B */
                  <0 0 0 3 &pcie0 0 0 0 2>, /* INT C */
                  <0 0 0 4 &pcie0 0 0 0 3>; /* INT D */
    device_type = "pci";
    num-lanes = <4>;
};

```



```

        #address-cells = <3>;
        #size-cells = <2>;
        power-domains = <&cpg>;
        vendor-id = <0x1912>;
        device-id = <0x003b>;
        pcie,channel = <0>;
        renesas,sysc = <&sys>;
        status = "disabled";
};

```

Figure 4-7 Example PCIe Root Complex (RZ/V2N EVK)

The PCIe device required properties:

compatible:

Must be set to "renesas,r9a09g056-pcie" for R9A09G056 (RZ/V2N Group).

device_type:

Must be set to "pci"

reg:

Base address and length of the memory resource used by the PCIe0.

clocks:

slot 1:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to 0xc4

slot 2:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to CPG_MOD.

The 3rd cell must be set to PCIE_0_CLK_PMU

reset:

The 1st cell is a node or label of CPG clock to be used.

The 2nd cell must be set to PCIE_0_ARESETN

ranges: support 32-bit area

For RZ/V2N, must be set: <0x3000000 0x0 0x30000000 0x0 0x30000000 0x0 0x08000000>

dma-ranges: a single range for the inbound memory region.

For RZ/V2N, must be set: <0x42000000 0 0x40000000 0 0x40000000 0x2 0x00000000>

interrupt:

interrupt for PCIE device

#interrupt-cells: must be 1.

interrupt-map: standard property used to define the mapping of the PCI interrupts to the GIC interrupts.

interrupt-map-mask: standard property that helps to define the interrupt mapping.

bus-range:

The PCI bus number range; as this is a single bus, the range should be specified as the same value.

renesas,sysc:

Specific property for RZ/V2N; must be set to <&sys> which is reference to the System control node

```
sys: system-controller@10430000 {
    compatible = "renesas,r9a09g056-sys", "syscon";
    reg = <0 0x10430000 0 0x10000>;
    clocks = <&cpg CPG_CORE R9A09G056_SYS_0_PCLK>;
    resets = <&cpg 0x30>;
};
```

Figure 4-8 Example SYS node (RZ/V2N-EVK)

4.2.1.2 Board Device-tree

(1) RZ/G3E Board

PCIe RC device on RZ/G3E Smarc EVK can be enabled in devicetree file “**rzg3e-smarc.dtsi**” in directory “**arch/arm64/boot/dts/renesas/**”. The device’s properties are shown below

Note: PCIe-RC mode can be set manually by macro **SEL_PCIE_MODE=0**

```
* SEL_PCIE_MODE:
* - 0: Root Complex (default)
* - 1: Endpoint
*/
#define SEL_PCIE_MODE 0
&pinctrl {
    pcie0_pins: pcie0 {
        pinmux = <RZV2H_PORT_PINMUX(G, 7, 1)>; /* PCIE_SLOT_RST */
    };
};

#if (!SEL_PCIE_MODE)
&pcie0 {
    status = "okay";
    pinctrl-0 = <&pcie0_pins>;
    pinctrl-names = "default";
};
#else
&pcie0_ep {
    status = "okay";
    pinctrl-0 = <&pcie0_pins>;
    pinctrl-names = "default";
};
#endif
```

Figure 4-9 Example of enabling PCIe RC channel 0 (RZ/G3E SMARC)

(2) RZ/G3S Board

PCIe RC device can be enabled in devicetree file “**rzg3s-smarc.dtsi**” in directory

“**arch/arm64/boot/dts/renesas/**”. The RZ/G3S PCIe device’s properties are shown below:

```
&pinctrl {
    pcie_pins: pcie {
        pinmux = <RZG2L_PORT_PINMUX(13, 2, 2)>, /* PCIE_RST_OUT_B */
                <RZG2L_PORT_PINMUX(13, 3, 2)>; /* PCIE_CLKREQ_B */
    };
};
&pcie {
    pinctrl-0 = <&pcie_pins>;
    pinctrl-names = "default";
    status = "okay";
};
```

Figure 4-10 Example of enabling PCIe RC (RZ/G3S)

(3) RZ/V2H Board

RZ/V2H-EVK have the first 128MB of memory which is reserved on this board for secure area. Update the PCIe dma-ranges property in:

arch/arm64/boot/dts/renesas/rzv2h-evk-common.dtsi

```
&pcie0 {
    /* First 128MB is reserved for secure area. */

    dma-ranges = <0x42000000 0 0x48000000 0 0x48000000 0x3 0xF8000000>;

    status = "okay";
};
```

Figure 4-11 Example of updating PCIe DMA mapping

PCIe RC device on RZ/V2H-EVK can be enabled by manual setting **SEL_PCIE_MODE=0** in devicetree in directory **“arch/arm64/boot/dts/renesas/”**.

RZ/V2H-EVK-VER1: **“r9a09g057h44-rzv2h-evk.dts”**

RZ/V2H-EVK-VER2: **“r9a09g057h44-rzv2h-evk-ver2.dts”**

Note: Default is PCIe-RC mode

(4) RZ/V2N Board

RZ/V2N-EVK have the first 128MB of memory which is reserved on this board for secure area. Update the PCIe dma-ranges property in:

arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dts

```
&pcie0 {
    /* First 128MB is reserved for secure area. */

    dma-ranges = <0x42000000 0 0x48000000 0 0x48000000 0x1 0xF8000000>;

    status = "okay";
};
```

Figure 4-12 Example of updating PCIe DMA mapping

PCIe RC device on RZ/V2N-EVK can be enabled by manual setting **PCIE_MODE_SEL=0** in devicetree in directory **“arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dts”**.

Note: Default is PCIe-RC mode

4.2.2 Endpoint

4.2.2.1 SoC Device-tree

(1) RZ/G3E SoC

Supported kernel: rz-6.1-cip28

```
pcie0_ep: pcie-ep@13400000 {
    compatible = "renesas,rzg3e-pcie-ep";
    reg = <0x0 0x13400000 0x0 0x10000>,
        <0x0 0x30000000 0x0 0x04000000>,
        <0x0 0x34000000 0x0 0x04000000>;
    reg-names = "apb-base", "memory0", "memory1";
    interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD 0xc4>,
        <&cpg CPG_MOD 0xc5>;
    power-domains = <&cpg>;
    resets = <&cpg 0xb2>;
    pcie,channel = <0>;
    renesas,pcie-sys = <&sys>;
    status = "disabled";
};
```

Figure 4-13 Example PCIe channel 0 Endpoint (RZ/G3E SMARC)

Note: All the information in the above device tree example is used for RZ/G3E boards. For others, please use corresponding properties as below.

compatible:

Must be set to "renesas,rzg3e-pcie-ep" for R9A09G047 (RZ/G3E).

reg:

For RZ/G3E must be set to:

```
<0x0 0x13400000 0x0 0x10000>,
<0x0 0x30000000 0x0 0x04000000>,
<0x0 0x34000000 0x0 0x04000000>;
```

Other parameters, please set the same as Root Complex device tree.

(2) RZ/V2H SoC**Supported kernel: rz-6.1-cip43**

```

pcie0_ep: pcie-ep@13400000 {
    compatible = "renesas,rzv2h-pcie-ep";
    reg = <0x0 0x13400000 0x0 0x10000>,
        <0x0 0x30000000 0x0 0x04000000>,
        <0x0 0x34000000 0x0 0x04000000>;
    reg-names = "apb-base", "memory0", "memory1";
    interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>;
        <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "all", "dma", "pcie_evt";
    clocks = <&cpg CPG_MOD 0xc4>,
        <&cpg CPG_MOD 0xc5>;
    power-domains = <&cpg>;
    vendor-id = <0x1912>;
    device-id = <0x003a>;
    resets = <&cpg 0xb2>;
    pcie,channel = <0>;
    renesas,pcie-sys = <&sys>;
    status = "disabled";
}

```

Figure 4-14 Example PCIe channel 0 Endpoint (RZ/V2H EVK)

Note: All the information in the above device tree example is used for RZ/V2H boards. For others, please use corresponding properties as below.

compatible:

Must be set to "renesas,rzv2h-pcie-ep" for R9A09G057 (RZ/V2H).

reg:

For RZ/V2H must be set to:

```

<0x0 0x13400000 0x0 0x10000>,
<0x0 0x30000000 0x0 0x04000000>,
<0x0 0x34000000 0x0 0x04000000>;

```

interrupts:

For RZ/V2H, must be set to:

```

interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
    <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>,
    <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>;
interrupt-names = "all", "dma", "pcie_evt";

```

Other parameters, please set the same as Root Complex device tree.

(3) RZ/V2N SoC**Supported kernel: rz-6.1-cip43**

```

pcie0_ep: pcie-ep@13400000 {
    compatible = "renesas,rzv2n-pcie-ep";
    reg = <0x0 0x13400000 0x0 0x10000>,
        <0x0 0x30000000 0x0 0x04000000>,
        <0x0 0x34000000 0x0 0x04000000>;
    reg-names = "apb-base", "memory0", "memory1";
    interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "all", "dma", "pcie_evt";
    clocks = <&cpg CPG_MOD 0xc4>,
        <&cpg CPG_MOD 0xc5>;
    power-domains = <&cpg>;
    vendor-id = <0x1912>;
    device-id = <0x003a>;
    resets = <&cpg 0xb2>;
    pcie,channel = <0>;
    renesas,pcie-sys = <&sys>;
    status = "disabled";
};

```

Figure 4-15 Example PCIe channel 0 Endpoint (RZ/V2N EVK)

Note: All the information in the above device tree example is used for RZ/V2N boards. For others, please use corresponding properties as below.

compatible:

Must be set to "renesas,rzv2n-pcie-ep" for R9A09G056 (RZ/V2N).

reg:

For RZ/V2N must be set to:

```

<0x0 0x13400000 0x0 0x10000>,
<0x0 0x30000000 0x0 0x04000000>,
<0x0 0x34000000 0x0 0x04000000>;

```

interrupts:

For RZ/V2N, must be set to:

```

interrupts = <GIC_SPI 791 IRQ_TYPE_LEVEL_HIGH>,
    <GIC_SPI 804 IRQ_TYPE_LEVEL_HIGH>,
    <GIC_SPI 805 IRQ_TYPE_LEVEL_HIGH>;
interrupt-names = "all", "dma", "pcie_evt";

```

Other parameters, please set the same as Root Complex device tree.

5. Integration

5.1 Directory Configuration

The directory configuration for RZ/G3E is shown below. (**supported kernel: rz-6.1-cip28**)

—— drivers/pci/controller	—— pcie-rzv2h.h	: PCIe header file
	—— pcie-rzv2h-window.c	: PCIe common instruction support file
	—— pcie-rzv2h-host.c	: PCIe Host Controller source file
	—— pcie-rzv2h-ep.c	: PCIe Endpoint Controller source file

Figure 5-1 Directory configuration (RZ/G3E)

The directory configuration for RZ/G3S, RZ/V2H and RZ/V2N is shown below. (**supported kernel: rz-6.1-cip43**)

Note: Not support Endpoint function and DMA function for RZ/G3S Group.

—— drivers/pci/controller	—— pcie-rzg3s-regs.h	: PCIe header file
	—— pcie-rzg3s-host.c	: PCIe Host Controller source file
	—— pcie-rzg3s-dma.c	: PCIe DMA source file
	—— pcie-rz-ep.c	: PCIe Endpoint Controller source file

Figure 5-2 Directory configuration (RZ/G3S, RZ/V2H and RZ/V2N)

5.2 Integration Procedure

5.2.1 Kernel Configuration for Root Complex

(1) RZ/G3E Board

Supported kernel: rz-6.1-cip28

To enable the function of this module, make the following setting with Kernel Configuration. This module is selected only as a build-in module.

```
Device Drivers --->
  [*] PCI support
    PCI controller drivers --->
      <*> Renesas RZ/V2H Series PCIe controller
      [*] Message Signaled Interrupts (MSI)
```

Figure 5-3 Kernel configuration for RC (RZ/G3E)

Configuration symbol of PCIe-RC for RZ/G3E as below:

```
CONFIG_RZV2H_PCIE=y
```


Figure 5-4 Kernel configuration symbol for RC (RZ/G3E)**(2) RZ/G3S, RZ/V2H and RZ/V2N Board****Supported kernel: rz-6.1-cip43**

To enable the function of this module, make the following setting with Kernel Configuration. This module is selected only as a build-in module

```
Device Drivers --->
  [*] PCI support
    PCI controller drivers --->
      <*> Renesas RZ/G3S PCIe host controller
    [*] Message Signaled Interrupts (MSI)
```

Figure 5-5 Kernel configuration for RC (RZ/G3S, RZ/V2H and RZ/V2N)

Configuration symbol of PCIe-RC for RZ/G3S, RZ/V2H and RZ/V2N Board as below:

```
CONFIG_PCIE_RENESAS_RZG3S_HOST=y
```

Figure 5-6 Kernel configuration symbol for RC (RZ/G3S, RZ/V2H and RZ/V2N)**5.2.2 Kernel Configuration for Endpoint**

To enable the function of this module, make the following setting with Kernel Configuration. This module is selected only as a built-in module.

(1) RZ/G3E Board**Supported kernel: rz-6.1-cip28**

```
Device Drivers --->
  [*] PCI support
    PCI controller drivers --->
      <*> Renesas RZ/V2H Series PCIe endpoint controller
    PCI Endpoint support --->
      [*] PCI Endpoint Configfs Support
      <*> PCI Endpoint Test driver
    Misc devices --->
      <*> PCI Endpoint Test driver
    [*] Message Signaled Interrupts (MSI)
```

Figure 5-7 Kernel configuration for Endpoint (RZ/G3E)

```
CONFIG_RZV2H_PCIE_EP=y
CONFIG_PCI_ENDPOINT=y
CONFIG_PCI_EPF_TEST=y
CONFIG_PCI_ENDPOINT_TEST=y
```

Figure 5-8 Kernel configuration symbol for Endpoint (RZ/G3E)**(2) RZ/V2H and RZ/V2N Board****Supported kernel: rz-6.1-cip43**

```
Device Drivers --->
  [*] PCI support
    PCI controller drivers --->
      [*] Renesas RZ Series PCIe endpoint controller
      [*] Renesas RZ/G3S PCIe internal DMA
    PCI Endpoint support --->
      [*] PCI Endpoint Configfs Support
      <*> PCI Endpoint Test driver
    Misc devices --->
      <*> PCI Endpoint Test driver
```

[*] Message Signaled Interrupts (MSI)

Figure 5-9 Kernel configuration for Endpoint (RZ/V2H and RZ/V2N)

```
CONFIG_RZ_PCIE_EP=y
CONFIG_PCI_ENDPOINT=y
CONFIG_PCI_EPF_TEST=y
CONFIG_PCI_ENDPOINT_TEST=y
CONFIG_PCIE_RENESAS_RZG3S_DMA=y
```

Figure 5-10 Kernel configuration symbol for Endpoint (RZ/V2H and RZ/V2N)

5.3 Option Setting

5.3.1 Module Parameters

There are no module parameters

5.3.2 Kernel Parameters

There are no kernel parameters

REVISION HISTORY	Linux Interface Specification Device Driver PCIe User's Manual: Software
------------------	---

Rev.	Date	Description	
		Page	Summary
1.00	Jun. 30, 2025	—	First Edition issued Add RZ/V2N support information
1.01	Jul. 22, 2025	—	Add RZ/G3E support information
1.02	Nov. 30, 2025	—	Add RZ/G3S support information
1.03	Dec. 19, 2025	—	Add RZ/V2H support information. Remove 4.2.2.2 Board Device-tree for Endpoint function.
1.04	Mar. 27, 2026	—	Update RZ/V2N kernel version to rz-6.1-cip43

Linux Interface Specification Device Driver PCIe
User's Manual: Software

Publication Date: Rev. 1.04 Mar. 27, 2026

Published by: Renesas Electronics Corporation

RZ/V2N Group, RZ/G3E Group, RZ/G3S Group
and RZ/V2H Group



Renesas Electronics Corporation