

# Linux Interface Specification DMAC

User's Manual: Software

RZ/G2L Group, RZ/V2L Group, RZ/V2N Group,  
RZ/V2H Group, RZ/G3E Group, RZ/G3S Group  
and RZ/Five

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 1.16 Mar 2026)

## Corporate Headquarters

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

[www.renesas.com](http://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/](http://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.. It is intended for users developing software incorporating the processors. A basic knowledge of software development and Linux systems is necessary in order to use this document.

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/G2L Group, RZ/V2L Group, RZ/V2N Group, RZ/V2H Group, RZ/G3E Group, RZ/G3S Group and RZ/Five 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/G2L Group User's Manual: Hardware	---
		RZ/V2L Group User's Manual: Hardware	---
		RZ/V2N Group User's Manual: Hardware	---
		RZ/V2H Group User's Manual: Hardware	---
		RZ/Five Group User's Manual: Hardware	---
		RZ/G3E Group User's Manual: Hardware	---
		RZ/G3S Group User's Manual: Hardware	---
User's manual for Software	Description of DMAC Linux interface Specification	Linux interface Specification Device Driver DMAC	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
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller

## Table of Contents

1. Overview.....	1
1.1 Overview .....	1
1.2 DMAC HW support .....	1
1.3 Function .....	1
1.4 Restrictions .....	1
2. Terminology.....	2
3. Operating Environment.....	3
3.1 Module Configuration.....	3
3.2 State Transition Diagram .....	3
4. Integration.....	4
4.1 Hardware Environment .....	4
4.2 Directory Configuration .....	4
4.3 Integration Procedure .....	4
5. Device Tree Setting .....	5
5.1 DMA node .....	5
5.1.1 On RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, RZ/G3S and RZ/Five .....	5
5.1.2 On RZ/V2N, RZ/V2H and RZ/G3E.....	8
5.2 DMA setting for client.....	12
5.2.1 On RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, RZ/G3S and RZ/Five .....	12
5.2.2 On RZ/V2N, RZ/V2H and RZ/G3E.....	14
6. Notes .....	16
6.1 Configuration for DMA Mem to Mem transaction.....	16

# 1. Overview

## 1.1 Overview

This manual explains the DMA Engine device driver in Linux Solution for RZ/G2L Group, RZ/V2L Group, RZ/Five, RZ/V2N Group, RZ/V2H Group, RZ/G3S Group and RZ/G3E Group.

Note: Currently, this device is supported in two kernel versions v5.10 and v6.1 with the information below:

- v5.10: RZ/G2L Group, RZ/V2L Group, RZ/G3S Group and RZ/Five Group.
- v6.1: RZ/G2L Group, RZ/V2L Group, RZ/V2N Group, RZ/V2H Group, RZ/G3S Group and RZ/G3E Group.

## 1.2 DMAC HW support

**Table 1-1 Support DMAC in RZ/G2L Group, RZ/V2L Group, RZ/G3S Group and RZ/Five Group**

	RZ/G2L group, RZ/V2L Group, RZ/G3S Group and RZ/Five Group
SYS-DMAC	16 channels

**Table 1-2 Support DMAC in RZ/V2N Group, RZ/V2H Group and RZ/G3E Group**

	RZ/V2N Group, RZ/V2H Group and RZ/G3E Group
SYS-DMAC	5 DMAC modules, each has 16 channels

## 1.3 Function

This module controls DMAC by using DMA Engine framework and provides the following functions:

- DMA device driver module management with DMA Engine framework.
- Control each channel of support DMAC. (Refer to **Table 1-1** and **Table 1-2** in details)
- Call the registered callback function when DMA transfer is completed.
- Control the data transfer between the memory and the peripheral (such as SSI, RSPI, SCIF...), memory to memory (SYS-DMAC).
- Forced to terminate during transmission.
- Control the descriptor transfer.

## 1.4 Restrictions

There is no restriction in this module.



## 2. Terminology

The following table shows the terminology related to this module.

**Table 2-1 Terminology**

Terms	Explanation
DMA	Direct Memory Access
DMAC	DMA Controller
SYS-DMAC	System DMAC

### 3. Operating Environment

#### 3.1 Module Configuration

The following figure shows the configuration of this module.

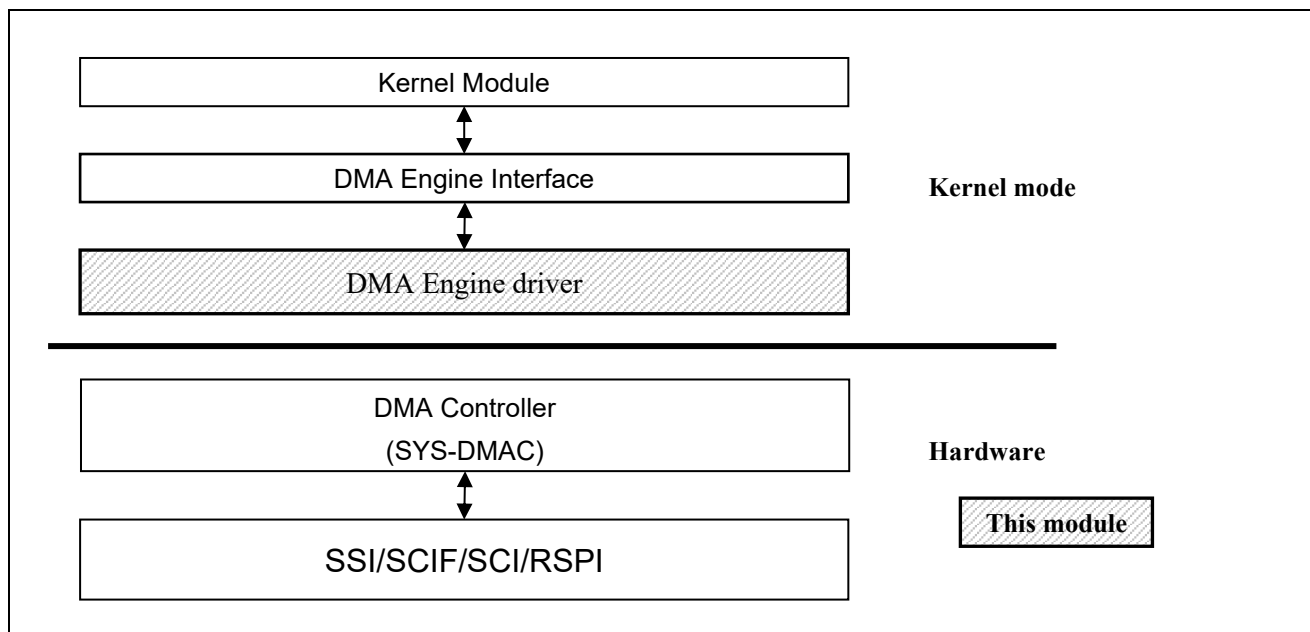


Figure 3-1 DMA Engine Driver configuration

#### 3.2 State Transition Diagram

The state transition managed by this module has "DMA stop" state and "During DMA transmission" state. This module does not support the suspend state.

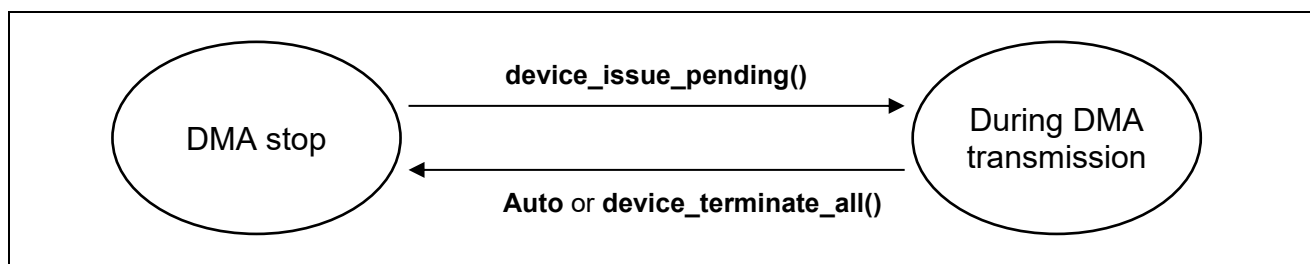


Figure 3-2 DMA Engine Driver State Transition Diagram

## 4. Integration

### 4.1 Hardware Environment

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

**Table 4-1 Hardware specification**

Name	Product number
RZ/G2L Evaluation Board Kit	RTK9744L23S01000BE
RZ/G2LC Evaluation Board Kit	RTK9744C22S01000BE
RZ/G2UL Evaluation Board Kit	RTK9743U11S01000BE
RZ/V2L Evaluation Board Kit	RTK9754L23S01000BE
RZ/V2N Evaluation Board Kit V1.0	RTK0EF0186C03000BJ
RZ/V2N Evaluation Board Kit V2.0	RTK0EF0186C03001BJ
RZ/V2H Evaluation Board Kit	RTK0EF0168C04000BJ
RZ/G3S Evaluation Board Kit	RTK9845S33C01000BE
RZ/G3E Evaluation Board Kit	RTK9947E57S01000BE
RZ/Five Evaluation Board Kit	RTK9743F01S01000BE

### 4.2 Directory Configuration

The directory configuration is shown below.

—— drivers/dma/sh      —— rz-dmac.c      : SYS-DMAC driver source file

**Figure 4-1 Directory configuration**

### 4.3 Integration Procedure

To enable the function of this module, make the following setting with Kernel Configuration.

```
Device Drivers  --->
  [*] DMA Engine support  --->
    <*> Renesas RZ/G2L DMA Controller
```

**Figure 4-2 Kernel configuration**

## 5. Device Tree Setting

### 5.1 DMA node

Basic configuration information of SYS-DMAC is defined at device tree file (r9a07g044.dtsi for RZ/G2L, RZ/G2LC, r9a07g043.dtsi RZ/G2UL, RZ/Five or r9a07g054.dtsi for RZ/V2L, r9a09g056.dtsi for RZ/V2N, r9a09g057.dtsi for RZ/V2H, r9a09g047.dtsi for RZ/G3E and r9a08g045.dtsi for RZ/G3S). It exists at **arch/arm64/boot/dts/renesas** directory. This module uses these described contents in the device tree file.

#### 5.1.1 On RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, RZ/G3S and RZ/Five

Figure 5–1 shows details about the initial reference information for DMAC node of RZ/G2L, RZ/G2LC, and RZ/V2L.

```

dma-controller@11820000 {
    compatible = "renesas,r9a07g044-dmac",
                 "renesas,rz-dmac";
    reg = <0 0x11820000 0 0x10000>,
          <0 0x11830000 0 0x10000>;
    interrupts = <GIC_SPI 141 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 125 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 126 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 127 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 128 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 129 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 130 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 131 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 132 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 133 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 134 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 135 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 136 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 137 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 138 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 139 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 140 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error",
                     "ch0", "ch1", "ch2", "ch3",
                     "ch4", "ch5", "ch6", "ch7",
                     "ch8", "ch9", "ch10", "ch11",
                     "ch12", "ch13", "ch14", "ch15";
    clocks = <&cpg CPG_MOD R9A07G044_DMAC_ACLK>,
             <&cpg CPG_MOD R9A07G044_DMAC_PCLK>;
    clock-names = "main", "register";
    power-domains = <&cpg>;
    resets = <&cpg R9A07G044_DMAC_ARESETN>,
             <&cpg R9A07G044_DMAC_RST_ASYNC>;
    reset-names = "arst", "rst_async";
    #dma-cells = <1>;
    dma-channels = <16>;
};

```

Figure 5–1 Device tree initial references information of DMAC (RZ/G2L, RZ/G2LC and RZ/V2L)

Figure 5–2 Shows details about the initial reference information for DMAC node of RZ/G3S

```

dmac: dma-controller@11820000 {
    compatible = "renesas,r9a08g045-dmac",
                 "renesas,rz-dmac";
    reg = <0 0x11820000 0 0x10000>,
          <0 0x11830000 0 0x10000>;
    interrupts = <GIC_SPI 111 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 112 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 113 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 114 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 115 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 116 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 117 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 118 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 119 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 120 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 121 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 122 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 123 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 124 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 125 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 126 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 127 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error",
                     "ch0", "ch1", "ch2", "ch3",
                     "ch4", "ch5", "ch6", "ch7",
                     "ch8", "ch9", "ch10", "ch11",
                     "ch12", "ch13", "ch14", "ch15";
    clocks = <&cpg CPG_MOD R9A08G045_DMAC_ACLK>,
             <&cpg CPG_MOD R9A08G045_DMAC_PCLK>;
    clock-names = "main", "register";
    power-domains = <&cpg>;
    resets = <&cpg R9A08G045_DMAC_ARESETN>,
             <&cpg R9A08G045_DMAC_RST_ASYNC>;
    reset-names = "arst", "rst_async";
    #dma-cells = <1>;
    dma-channels = <16>;
};

```

**Figure 5–2 Device tree initial references information of RZ/G3S DMAC**

Note: All the information in the above device tree is commonly used for RZ/G2L, RZ/G2LC, RZ/V2L and RZ/G3S except clocks and resets.

Required properties:

- compatible: must be  
"renesas, r9a07g044-dmac" for R9A07G044L, R9A07G044C (RZ/G2L, RZ/G2LC), "renesas, r9a07g043-dmac" for R9A07G043 (RZ/G2UL), and "renesas, r9a07g054-dmac" for R9A07G054 (RZ/V2L) and "renesas, r9a08g045-dmac" for R9A08G045 (RZ/G3S).
- reg:  
need two resources, first one mapped to register base address, second one mapped to extension resource registers base address.
- interrupts:  
interrupt specifier for the DMAC, one for each channel.
- interrupt-names:  
mapped with interrupts and must be in format: "ch%u" (%u: channel number 0->15), "error".
- dma-channels:  
number of dma channel. Maximum is 16.
- #dma-cells:  
must be 1.
- clocks, resets: CPG\_MOD R9A07G044\_DMAC\_X is used for RZ/G2L, RZ/G2LC, CPG\_MOD R9A07G043\_DMAC\_X is used for RZ/G2UL, RZ/Five, CPG\_MOD R9A07G054\_DMAC\_X is used for RZ/V2L and CPG\_MODE R9A08G045\_DMAC\_X is used for RZ/G3S  
phandle + clock/reset specifier pairs.

- RZ/Five: DMA node: arch/arm64/boot/dts/renesas/r9a07g043.dtsi

```

dmac: dma-controller@11820000 {
    compatible = "renesas,r9a07g043-dmac",
                 "renesas,rz-dmac";
    reg = <0 0x11820000 0 0x10000>,
          <0 0x11830000 0 0x10000>;
    interrupts = <SOC_PERIPHERAL_IRQ(141) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(125) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(126) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(127) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(128) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(129) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(130) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(131) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(132) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(133) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(134) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(135) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(136) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(137) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(138) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(139) IRQ_TYPE_EDGE_RISING>,
                 <SOC_PERIPHERAL_IRQ(140) IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error",
                     "ch0", "ch1", "ch2", "ch3",
                     "ch4", "ch5", "ch6", "ch7",
                     "ch8", "ch9", "ch10", "ch11",
                     "ch12", "ch13", "ch14", "ch15";
    clocks = <&cpg CPG_MOD R9A07G043_DMAC_ACLK>,
             <&cpg CPG_MOD R9A07G043_DMAC_PCLK>;
    clock-names = "main", "register";
    power-domains = <&cpg>;
    resets = <&cpg R9A07G043_DMAC_ARESETN>,
             <&cpg R9A07G043_DMAC_RST_ASYNC>;
    reset-names = "arst", "rst_async";
    #dma-cells = <1>;
    dma-channels = <16>;
};

```

### 5.1.2 On RZ/V2N, RZ/V2H and RZ/G3E

Figure 5-3 shows details about the initial reference information for DMAC node of RZ/V2N, RZ/V2H and RZ/G3E.

```

dmac1: dma-controller@14830000 {
    compatible = "renesas,r9a09g057-dmac";
    reg = <0 0x14830000 0 0x10000>;
    interrupts = <GIC_SPI 495 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 25 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 26 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 27 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 28 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 29 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 30 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 31 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 32 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 33 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 34 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 35 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 36 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 37 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 38 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 39 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 40 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error",
        "ch0", "ch1", "ch2", "ch3",
        "ch4", "ch5", "ch6", "ch7",
        "ch8", "ch9", "ch10", "ch11",
        "ch12", "ch13", "ch14", "ch15";
    clocks = <&cpg CPG_MOD 0x1>;
    power-domains = <&cpg>;
    resets = <&cpg 0x32>;
    #dma-cells = <1>;
    dma-channels = <16>;
    renesas,icu = <&icu 0>;
    status = "okay";
};

```

Figure 5–3 Example of device tree of DMAC on RZ/V2N



```

dmac1: dma-controller@14830000 {
    compatible = "renesas,r9a09g057-dmac";
    reg = <0 0x14830000 0 0x10000>;
    interrupts = <GIC_SPI 495 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 25 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 26 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 27 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 28 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 29 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 30 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 31 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 32 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 33 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 34 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 35 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 36 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 37 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 38 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 39 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 40 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error",
        "ch0", "ch1", "ch2", "ch3",
        "ch4", "ch5", "ch6", "ch7",
        "ch8", "ch9", "ch10", "ch11",
        "ch12", "ch13", "ch14", "ch15";
    clocks = <&cpg CPG_MOD 0x1>;
    power-domains = <&cpg>;
    resets = <&cpg 0x32>;
    #dma-cells = <1>;
    dma-channels = <16>;
    renesas,icu = <&icu 0>;
};

```

Figure 5–4 Example of device tree of DMAC on RZ/V2H

**Note:** Figure 5–3 shows an example of device tree settings of DMAC for RZ/V2N. Please use corresponding properties as below.

Required properties:

- compatible:  
must be “renesas, r9a09g057-dmac” for RZ/V2H and RZ/V2N  
must be “renesas, r9a09g047-dmac” for RZ/G3E
- reg:  
need two resources, first one mapped to register base address, second one mapped to extension resource registers base address.
- interrupts:  
interrupt specifier for the DMAC, one for each channel.
- interrupt-names:  
mapped with interrupts and must be in format: “error”, “ch%u” (%u: channel number 0->15).
- dma-channels:  
number of dma channel. Maximum is 16.
- #dma-cells:  
must be 1.
- clocks, resets: phandle + clock/reset specifier pairs.  
0x32 is used for RZ/V2N, RZ/V2H.
- peripheral-request used for RZ/V2N, RZ/V2H and RZ/G3E  
renesas,icu is used. The number icu specifies for dmac, user can check **DMACx Factor Selection Register y (DMxSELy) (x = 0 to 4, y = 0 to 7)** of hardware manual for this number.  
Ex:       dmac1:renesas,icu = <&icu 0>;  
          dmac2:renesas,icu = <&icu 1>;

## 5.2 DMA setting for client

### 5.2.1 On RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, RZ/G3S and RZ/Five

```

ssi0: ssi@10049c00 {
    compatible = "renesas,r9a07g044-ssi",
                 "renesas,rz-ssi";
    reg = <0 0x10049c00 0 0x400>;
    interrupts = <GIC_SPI 326 IRQ_TYPE_LEVEL_HIGH>,
                 <GIC_SPI 327 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 328 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 329 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "int_req", "dma_rx", "dma_tx", "dma_rt";
    clocks = <&cpg CPG_MOD R9A07G044_SSI0_PCLK2>,
             <&cpg CPG_MOD R9A07G044_SSI0_PCLK_SFR>,
             <&audio_clk1>, <&audio_clk2>;
    clock-names = "ssi", "ssi_sfr", "audio_clk1", "audio_clk2";
    resets = <&cpg R9A07G044_SSI0_RST_M2_REG>;
    dmas = <&dmac 0x2655>, <&dmac 0x2656>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    #sound-dai-cells = <0>;
    status = "disabled";
};

spi1: spi@1004b000 {
    compatible = "renesas,r9a07g044-rspi", "renesas,rspi-rz";
    reg = <0 0x1004b000 0 0x400>;
    interrupts = <GIC_SPI 418 IRQ_TYPE_LEVEL_HIGH>,
                 <GIC_SPI 416 IRQ_TYPE_LEVEL_HIGH>,
                 <GIC_SPI 417 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "error", "rx", "tx";
    clocks = <&cpg CPG_MOD R9A07G044_RSPI1_CLKB>;
    resets = <&cpg R9A07G044_RSPI1_RST>;
    dmas = <&dmac 0x2e99>, <&dmac 0x2e9a>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    num-cs = <1>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};

```

**Figure 5–5 Device tree setting references for SSI and RSPI (RZ/G2L, RZ/V2L, and RZ/Five)**

```

ssi0: ssi@100a8000 {
    compatible = "renesas,r9a08g045-ssi",
                 "renesas,rz-ssi";
    reg = <0 0x100a8000 0 0x400>;
    interrupts = <GIC_SPI 240 IRQ_TYPE_LEVEL_HIGH>,
                 <GIC_SPI 241 IRQ_TYPE_EDGE_RISING>,
                 <GIC_SPI 242 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "int_req", "dma_rx", "dma_tx";
    clocks = <&cpg CPG_MOD R9A08G045_SSI0_PCLK2>,
             <&cpg CPG_MOD R9A08G045_SSI0_PCLK_SFR>,
             <&audio_clk1>, <&audio_clk2>;
    clock-names = "ssi", "ssi_sfr", "audio_clk1", "audio_clk2";
    resets = <&cpg R9A08G045_SSI0_RST_M2_REG>;
    dmas = <&dmac 0x2665>, <&dmac 0x2666>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    #sound-dai-cells = <0>;
    status = "disabled";
};

```

**Figure 5–6 Device tree setting references for SSI (RZ/G3S)**

In the client driver node:

<pre>dmac = &lt;&amp;dmac 0x2e99&gt;, &lt;&amp;dmac 0x2e9a&gt;; dma-names = "tx", "rx";</pre>
---

With:

dmac specifies the encoded MID/RID values of the DMAC port connected to the DMA client and the slave channel configuration parameters.

- bits[0:9] - Specifies MID/RID value
- bit[10] - Specifies DMA request high enable (HIEN)
- bit[11] - Specifies DMA request detection type (LVL)
- bits[12:14] - Specifies DMAACK output mode (AM)
- bit[15] - Specifies Transfer Mode (TM)

dma-name name of dmac channel

- tx - dma transfer channel
- rx - dma receive channel

Please follow guidelines in Documentation/devicetree/bindings/dma/renesas,rz-dmac.yaml and refer to

**Table 14.3 On-Chip Module Requests** in the RZ/G2L Group, RZ/G2LC Group, RZ/V2L, RZ/G3S Group and RZ/Five Hardware Manual.

## 5.2.2 On RZ/V2N, RZ/V2H and RZ/G3E

```

rspi2: spi@12800800 {
    compatible = "renesas,r9a09g056-rspi",
                "renesas,r9a09g057-rspi";
    reg = <0x0 0x12800800 0x0 0x400>;
    interrupts = <GIC_SPI 111 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 112 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 113 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 504 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 505 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "idle", "error", "end", "rx", "tx";
    clocks = <&cpg CPG_MOD 0x5a>,
            <&cpg CPG_MOD 0x5b>,
            <&cpg CPG_MOD 0x5c>;
    clock-names = "pclk", "pclk_sfr", "tclk";
    resets = <&cpg 0x7f>, <&cpg 0x80>;
    reset-names = "presetrn", "tresetrn";
    dmas = <&dmac1 0x7F4491>, <&dmac1 0x7F4490>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};

```

Figure 5–7 Device tree setting references for RSPI2 on RZ/V2N

```

spi0: spi@12800000 {
    compatible = "renesas,rspi-g3e";
    reg = <0 0x12800000 0 0x400>;
    interrupts = <GIC_SPI 106 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 107 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 500 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 501 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "error", "cend", "rx", "tx";
    clocks = <&cpg CPG_MOD 84>,
            <&cpg CPG_MOD 85>,
            <&cpg CPG_MOD 86>;
    clock-names = "pclk", "pclk_sfr", "tclk";
    resets = <&cpg 123>,
            <&cpg 124>;
    reset-names = "presetrn", "tresetrn";
    dmas = <&dmac1 0x7F448D>, <&dmac1 0x7F448C>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    num-cs = <1>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};

```

Figure 5–8 Device tree setting references for RSPI0 on RZ/G3E

```

rspi0: spi@12800000 {
    compatible = "renesas,r9a09g057-rspi";
    reg = <0x0 0x12800000 0x0 0x400>;
    interrupts = <GIC_SPI 105 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 106 IRQ_TYPE_LEVEL_HIGH>,
                <GIC_SPI 107 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 500 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 501 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "idle", "error", "end", "rx", "tx";
    clocks = <&cpg CPG_MOD 0x54>,
            <&cpg CPG_MOD 0x55>,
            <&cpg CPG_MOD 0x56>;
    clock-names = "pclk", "pclk_sfr", "tclk";
    resets = <&cpg 0x7b>, <&cpg 0x7c>;
    reset-names = "presetrn", "tresetrn";
    dmas = <&dmac1 0x7F448D>, <&dmac1 0x7F448C>;
    dma-names = "tx", "rx";
    power-domains = <&cpg>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};

```

**Figure 5–9 Device tree setting references for RSPI0 on RZ/V2H**

RZ/V2N required the properties as below for the dma client:

```

dmas = <&dmac1 0x7F4491>, <&dmac1 0x7F4490>;
dma-names = "tx", "rx";

```

RZ/V2H, RZ/G3E required the properties as below for the dma client:

```

dmas = <&dmac1 0x7F448D>, <&dmac1 0x7F448C>;
dma-names = "tx", "rx";

```

Where:

dmas specifies the DMAC REQ values of the DMAC channel configuration connected to ICU parameters.

- bits[0:9] - Specifies DMAC request number value
- bit[10] - Specifies DMA request high enable (HIEN)
- bit[11] - Specifies DMA request detection type (LVL)
- bits[12:14] - Specifies DMAACK output mode (AM)
- bit[15] - Specifies Transfer Mode (TM)
- bits[22:16] - Specifies DMAACK output number if not use define as 7F (all 1)

dma-name of dma channel

- tx - dma transfer channel
- rx - dma receive channel

Please follow guidelines in Documentation/devicetree/bindings/dma/renesas,rz-dmac.yaml and refer to

**Table 4.7-22 DMA Transfer Request Detection Operation Setting Table** in the RZ/G3E Group, RZ/V2N Group, RZ/V2H Group Hardware Manual.

## 6. Notes

### 6.1 Configuration for DMA Mem to Mem transaction.

For DMA mem to mem transaction, we need to care about the transfer data size (destination/source) and data alignment of the transaction.

Currently, in theory, DMAC can support up to **1024** bits transfer data size based on Hardware Manual.

**Table 6-1 Channel Configuration Register n/nS (CHCFG\_n/nS):**

Bit	Bit Name	Description																														
19 to 16	DDS [3:0]	Destination Data Size  Sets the DMA transfer size of the transfer destination:																														
		<table><tr><th>Value</th><th>Size</th><th>Data Alignment</th></tr><tr><td>0000</td><td>8 bits</td><td>1 byte</td></tr><tr><td>0001</td><td>16 bits</td><td>2 bytes</td></tr><tr><td>0010</td><td>32 bits</td><td>4 bytes</td></tr><tr><td>0011</td><td>64 bits</td><td>8 bytes</td></tr><tr><td>0100</td><td>128 bits</td><td>16 bytes</td></tr><tr><td>0101</td><td>256 bits</td><td>32 bytes</td></tr><tr><td>0110</td><td>512 bits</td><td>64 bytes</td></tr><tr><td>0110</td><td>1024 bits</td><td>128 bytes</td></tr><tr><td>Other than the above</td><td>-</td><td>Setting prohibited</td></tr></table>	Value	Size	Data Alignment	0000	8 bits	1 byte	0001	16 bits	2 bytes	0010	32 bits	4 bytes	0011	64 bits	8 bytes	0100	128 bits	16 bytes	0101	256 bits	32 bytes	0110	512 bits	64 bytes	0110	1024 bits	128 bytes	Other than the above	-	Setting prohibited
		Value	Size	Data Alignment																												
		0000	8 bits	1 byte																												
		0001	16 bits	2 bytes																												
		0010	32 bits	4 bytes																												
		0011	64 bits	8 bytes																												
		0100	128 bits	16 bytes																												
		0101	256 bits	32 bytes																												
		0110	512 bits	64 bytes																												
		0110	1024 bits	128 bytes																												
		Other than the above	-	Setting prohibited																												
15 to 12	SDS [3:0]	Source Data Size  Sets the DMA transfer size of the transfer source:																														
		<table><tr><th>Value</th><th>Size</th><th>Data Alignment</th></tr><tr><td>0000</td><td>8 bits</td><td>1 byte</td></tr><tr><td>0001</td><td>16 bits</td><td>2 bytes</td></tr><tr><td>0010</td><td>32 bits</td><td>4 bytes</td></tr><tr><td>0011</td><td>64 bits</td><td>8 bytes</td></tr><tr><td>0100</td><td>128 bits</td><td>16 bytes</td></tr><tr><td>0101</td><td>256 bits</td><td>32 bytes</td></tr></table>	Value	Size	Data Alignment	0000	8 bits	1 byte	0001	16 bits	2 bytes	0010	32 bits	4 bytes	0011	64 bits	8 bytes	0100	128 bits	16 bytes	0101	256 bits	32 bytes									
		Value	Size	Data Alignment																												
		0000	8 bits	1 byte																												
		0001	16 bits	2 bytes																												
		0010	32 bits	4 bytes																												
		0011	64 bits	8 bytes																												
		0100	128 bits	16 bytes																												
		0101	256 bits	32 bytes																												

		0110	512 bits	64 bytes
		0110	1024 bits	128 bytes
		Other than the above	-	Setting prohibited

Currently, in Linux driver (**drivers/dma/sh/rz-dmac.c**), the DMA transfer size for destination and source are set to 8 bits via **CHCFG\_MEM\_COPY** macro.

```
#define CHCFG_MEM_COPY (0x80400008)
```

This default setting can cause low performance when doing the DMA mem-to-mem transaction.

To increase the transfer speed, please change the value of this macro to your considered DMA transfer size (compatible with data alignment) based on the above table.

Example: set the DMA transfer size to 32 bits

```
#define CHCFG_MEM_COPY (0x80422008)
```

Below is the experimental DMA mem to mem transfer speed for 16MB data on RZ/G2L Smarc Board.

Transfer data size (bits)	Average Time (ms)	Average Transfer Speed (KB/s)
8	7220	2269
16	3610	4537
32	1807	9064
64	905	18092
128	415	39464
256	229	71410
512	130	125450
1024	80	203996



Revision History	Linux Interface Specification Device Driver DMAC User's Manual: Software
------------------	---

Rev.	Date	Description	
		Page	Summary
0.50	Apr. 30, 21	—	First Edition issued
1.0	Jul. 15, 2021	—	No modification, keep version to keep consistent with other documents
1.1	Sep. 15, 2021	—	Merge RZ/G2L driver manual with RZ/V2L
1.2	Feb. 15, 2022	—	Add RZ/G2UL, RZ/G2LC device
1.3	Mar. 31, 2022	—	Change to rz-dmac driver when moving to kernel 5.10
1.4	May. 31, 2022	—	Update version and add RZ/V2L
1.5	Jun. 24, 2022	—	Add RZ/Five device
1.6	Sep. 15, 2022	—	No modification, keep version to keep consistent with other documents
1.7	Dec. 15, 2022	—	No modification, keep version to keep consistent with other documents
1.8	Mar. 15, 2023	—	No modification, keep version to keep consistent with other documents
1.9	Jul. 31, 2024	10, 11	Add note about DMA mem to mem transaction
1.10	Mar. 31, 2025	1, 4, 7, 8	Correct device tree node for RZ/G2L series, RZ/V2L and RZ/Five, correct DMA function and update DMA Engine Driver configuration
1.11	May. 30, 2025	1	Add MPU information support for both kernel versions v5.10 and v6.1.
1.12	Jun. 30, 2025	—	Merge the document with RZ/V2N document
1.13	Jul. 22, 2025	—	Merge the document with RZ/G3E document
1.14	Nov. 28, 2025	—	Add information of RZ/G2UL and RZ/V2L support for kernel v6.1 Update reference Table for setting dmas property for for RZ/G2L series, RZ/V2L, RZ/Five, RZ/V2N and RZ/G3E Add information of RZ/G3S support for kernel v5.10 and v6.1 Update hardware specification table
1.15	Dec. 19, 2025	—	Merge the document with RZ/V2H document
1.16	Mar. 27, 2026	9, 11	Update the properties of the device tree node for the RZ/V2N board.

---

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

Publication Date: Rev. 1.16 Mar. 27, 2026

Published by: Renesas Electronics Corporation

---

RZ/G2L Group, RZ/V2L Group, RZ/V2N Group,  
RZ/V2H Group, RZ/G3E Group, RZ/G3S Group  
and RZ/Five



Renesas Electronics Corporation