

Linux Interface Specification GPIO

User's Manual: Software

RZ/G2L Group, RZ/V2L Group, RZ/V2N Group,
RZ/G3E 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. 1.12 Jul 2025)

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.. 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/Five Group, RZ/V2N Group and RZ/G3E 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/Five Group User's Manual: Hardware	---
		RZ/V2N Group User's Manual: Hardware	---
		RZ/G3E Group User's Manual: Hardware	---
User's manual for Software	Description of GPIO Linux interface Specification	Linux interface Specification Device Driver GPIO	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
GPIO	General Purpose Input/Output

Table of Contents

1. Overview.....	1
1.1 Overview	1
1.2 Function	1
1.2.1 General Purpose Input Output Port (GPIO)	1
1.2.2 Port Function Control	4
1.2.3 Special Purpose Port Function Control	5
1.2.3.1 RZ/G2L, RZ/V2L and RZ/Five support settings of the following specific ports:	5
1.2.3.2 RZ/V2N support settings of the following specific ports:	5
1.2.3.3 RZ/G3E support settings of the following specific ports:	5
1.3 Reference.....	5
1.3.1 Standard.....	5
1.3.2 Related document	5
1.4 Restrictions	6
2. Terminology.....	7
3. Operating Environment.....	8
3.1 Hardware Environment	8
3.2 Module Configuration.....	8
3.3 State Transition Diagram	9
4. External Interface.....	10
4.1 sysfs interface	10
4.2 Available attributes for sysfs interface	10
5. Integration.....	11
5.1 Directory Configuration	11
5.2 Kernel configuration	11
5.3 Option Setting	12
5.3.1 Module Parameters	12
5.3.2 Kernel Parameters	12
5.4 Devicetree settings.	13
5.4.1 Device node definition	13
5.4.2 GPIO definition	14
5.4.3 Pin control definition	16

1. Overview

1.1 Overview

This manual explains the Linux GPIO device driver in the RZ/G2L Group, RZ/V2L Group, RZ/Five Group, RZ/V2N Group and RZ/G3E Group.

Note: Currently, the devices that support this function are RZ/G2{L, LC, UL}, RZ/V2L, RZ/Five, RZ/V2N and RZ/G3E.

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 and RZ/Five.
- v6.1: RZ/G2L, RZ/G2LC, RZ/V2N and RZ/G3E.

1.2 Function

Each port of the general-purpose I/O port is multiplexed with the terminal of the peripheral function, and either the general-purpose I/O port function or the peripheral function. It has the following features:

1.2.1 General Purpose Input Output Port (GPIO)

General-purpose input/output ports provide general-purpose ports that:

- Support selection of input/output in GPIO pin.
- Support reading state of high/low in Input pin.
- Support setting high/low value in Output pin.
- Support detection of interrupts (rising/falling edge and low/high level).

RZ/G2L and RZ/V2L can support up to 123 general-purpose I/O pins from 49 ports in the following table:

Table 1-1 GPIO supported pins in RZ/G2L and RZ/V2L.

Port name	External Terminal Name					
	Bit7-5	Bit4	Bit3	Bit2	Bit1	Bit0
PORT 10	-	-	-	-	P0_1	P0_0
PORT 11	-	-	-	-	P1_1	P1_0
PORT 12	-	-	-	-	P2_1	P2_0
PORT 13	-	-	-	-	P3_1	P3_0
PORT 14	-	-	-	-	P4_1	P4_0
PORT 15	-	-	-	P5_2	P5_1	P5_0
PORT 16	-	-	-	-	P6_1	P6_0
PORT 17	-	-	-	P7_2	P7_1	P7_0
PORT 18	-	-	-	P8_2	P8_1	P8_0
PORT 19	-	-	-	-	P9_1	P9_0
PORT 1A	-	-	-	-	P10_1	P10_0
PORT 1B	-	-	-	-	P11_1	P11_0
PORT 1C	-	-	-	-	P12_1	P12_0
PORT 1D	-	-	-	P13_2	P13_1	P13_0
PORT 1E	-	-	-	-	P14_1	P14_0
PORT 1F	-	-	-	-	P15_1	P15_0
PORT 20	-	-	-	-	P16_1	P16_0
PORT 21	-	-	-	P17_2	P17_1	P17_0
PORT 22	-	-	-	-	P18_1	P18_0
PORT 23	-	-	-	-	P19_1	P19_0
PORT 24	-	-	-	P20_2	P20_1	P20_0
PORT 25	-	-	-	-	P21_1	P21_0
PORT 26	-	-	-	-	P22_1	P22_0
PORT 27	-	-	-	-	P23_1	P23_0
PORT 28	-	-	-	-	P24_1	P24_0
PORT 29	-	-	-	-	P25_1	P25_0
PORT 2A	-	-	-	-	P26_1	P26_0
PORT 2B	-	-	-	-	P27_1	P27_0

PORT 2C	-	-	-	-	P28_1	P28_0
PORT 2D	-	-	-	-	P29_1	P29_0
PORT 2E	-	-	-	-	P30_1	P30_0
PORT 2F	-	-	-	-	P31_1	P31_0
PORT 30	-	-	-	-	P32_1	P32_0
PORT 31	-	-	-	-	P33_1	P33_0
PORT 32	-	-	-	-	P34_1	P34_0
PORT 33	-	-	-	-	P35_1	P35_0
PORT 34	-	-	-	-	P36_1	P36_0
PORT 35	-	-	-	P37_2	P37_1	P37_0
PORT 36	-	-	-	-	P38_1	P38_0
PORT 37	-	-	-	P39_2	P39_1	P39_0
PORT 38	-	-	-	P40_2	P40_1	P40_0
PORT 39	-	-	-	-	P41_1	P41_0
PORT 3A	-	P42_4	P42_3	P42_2	P42_1	P42_0
PORT 3B	-	-	P43_3	P43_2	P43_1	P43_0
PORT 3C	-	-	P44_3	P44_2	P44_1	P44_0
PORT 3D	-	-	P45_3	P45_2	P45_1	P45_0
PORT 3E	-	-	P46_3	P46_2	P46_1	P46_0
PORT 3F	-	-	P47_3	P47_2	P47_1	P47_0
PORT 40	-	P48_4	P48_3	P48_2	P48_1	P48_0

:- unused pins

Note: RZ/G2LC will support same amount of GPIO ports but fewer amount of GPIO pins than RZ/G2L.

RZ/G2UL can support up to 82 general-purpose I/O pins from 19 ports in the following table:

Table 1-2 GPIO supported pins in RZ/G2UL.

Port name	External Terminal Name					
	Bit7-5	Bit4	Bit3	Bit2	Bit1	Bit0
PORT 10	-	-	P0_3	P0_2	P0_1	P0_0
PORT 11	-	P1_4	P1_3	P1_2	P1_1	P1_0
PORT 12	-	-	P2_3	P2_2	P2_1	P2_0
PORT 13	-	-	P3_3	P3_2	P3_1	P3_0
PORT 14	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PORT 15	-	P5_4	P5_3	P5_2	P5_1	P5_0
PORT 16	-	P6_4	P6_3	P6_2	P6_1	P6_0
PORT 17	-	P7_4	P7_3	P7_2	P7_1	P7_0
PORT 18	-	P8_4	P8_3	P8_2	P8_1	P8_0
PORT 19	-	-	P9_3	P9_2	P9_1	P9_0
PORT 1A	-	P10_4	P10_3	P10_2	P10_1	P10_0
PORT 1B	-	-	P11_3	P11_2	P11_1	P11_0
PORT 1C	-	-	-	-	P12_1	P12_0
PORT 1D	-	P13_4	P13_3	P13_2	P13_1	P13_0
PORT 1E	-	-	-	P14_2	P14_1	P14_0
PORT 1F	-	-	P15_3	P15_2	P15_1	P15_0
PORT 20	-	-	-	-	P16_1	P16_0
PORT 21	-	-	P17_3	P17_2	P17_1	P17_0
PORT 22	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0

:- unused pins

RZ/Five can support up to 114 general-purpose I/O pins from 27 ports in the following table:

Table 1-3 GPIO supported pins in RZ/Five.

Port name	External Terminal Name							
	Bit 7	Bit 6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PORT 06	-	-	-	-	-	0 (*)	P19_1	P19_0
PORT 07	P20_7	P20_6	P20_5	P20_4	P20_3	P20_2	P20_1	P20_0
PORT 08	-	-	-	-	-	-	P21_1	0 (*)
PORT 09	-	-	-	-	P22_3	P22_2	P22_1	P22_0
PORT 0A	-	-	P23_5	P23_4	P23_3	P23_2	P23_1	0 (*)
PORT 0B	-	-	P24_5	P24_4	P24_3	P24_2	P24_1	P24_0
PORT 0C	-	-	-	-	-	-	P25_1	0 (*)
PORT 0F	-	-	P28_5	P28_4	P28_3	P28_2	P28_1	P28_0
PORT 10	-	-	-	-	P0_3	P0_2	P0_1	P0_0
PORT 11	-	-	-	P1_4	P1_3	P1_2	P1_1	P1_0
PORT 12	-	-	-	-	P2_3	P2_2	P2_1	P2_0
PORT 13	-	-	-	-	P3_3	P3_2	P3_1	P3_0
PORT 14	-	-	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PORT 15	-	-	-	P5_4	P5_3	P5_2	P5_1	P5_0
PORT 16	-	-	-	P6_4	P6_3	P6_2	P6_1	P6_0
PORT 17	-	-	-	P7_4	P7_3	P7_2	P7_1	P7_0
PORT 18	-	-	-	P8_4	P8_3	P8_2	P8_1	P8_0
PORT 19	-	-	-	-	P9_3	P9_2	P9_1	P9_0
PORT 1A	-	-	-	P10_4	P10_3	P10_2	P10_1	P10_0
PORT 1B	-	-	-	-	P11_3	P11_2	P11_1	P11_0
PORT 1C	-	-	-	-	-	-	P12_1	P12_0
PORT 1D	-	-	-	P13_4	P13_3	P13_2	P13_1	P13_0
PORT 1E	-	-	-	-	-	P14_2	P14_1	P14_0
PORT 1F	-	-	-	-	P15_3	P15_2	P15_1	P15_0
PORT 20	-	-	-	-	-	-	P16_1	P16_0
PORT 21	-	-	-	-	P17_3	P17_2	P17_1	P17_0
PORT 22	-	-	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0

:- unused pins

(*) when read, the initial value is read. The written value will be ignored.

RZ/V2N can support up to 86 general-purpose I/O pins from 12 ports in the following table:

Table 1-4 GPIO supported pins in RZ/V2N.

Port name	External Terminal Name							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PORT 20	P0_7	P0_6	P0_5	P0_4	P0_3	P0_2	P0_1	P0_0
PORT 21	-	-	P1_5	P1_4	P1_3	P1_2	P1_1	P1_0
PORT 22	-	-	-	-	-	-	P2_1	P2_0
PORT 23	P3_7	P3_6	P3_5	P3_4	P3_3	P3_2	P3_1	P3_0
PORT 24	P4_7	P4_6	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PORT 25	P5_7	P5_6	P5_5	P5_4	P5_3	P5_2	P5_1	P5_0
PORT 26	P6_7	P6_6	P6_5	P6_4	P6_3	P6_2	P6_1	P6_0
PORT 27	P7_7	P7_6	P7_5	P7_4	P7_3	P7_2	P7_1	P7_0
PORT 28	P8_7	P8_6	P8_5	P8_4	P8_3	P8_2	P8_1	P8_0
PORT 29	P9_7	P9_6	P9_5	P9_4	P9_3	P9_2	P9_1	P9_0
PORT 2A	PA_7	PA_6	PA_5	PA_4	PA_3	PA_2	PA_1	PA_0
PORT 2B	-	-	PB_5	PB_4	PB_3	PB_2	PB_1	PB_0

:- unused pins

(*) when reading, the initial value is read. The written value will be ignored

RZ/G3E can support up to 140 general-purpose I/O pins from 22 ports in the following table:

Table 1-5 GPIO supported pins in RZ/G3E.

Port name	External Terminal Name							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PORT 20	P0_7	P0_6	P0_5	P0_4	P0_3	P0_2	P0_1	P0_0
PORT 21	P1_7	P1_6	P1_5	P1_4	P1_3	P1_2	P1_1	P1_0
PORT 22	-	-	-	-	-	-	P2_1	P2_0
PORT 23	P3_7	P3_6	P3_5	P3_4	P3_3	P3_2	P3_1	P3_0
PORT 24	-	-	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PORT 25	-	P5_6	P5_5	P5_4	P5_3	P5_2	P5_1	P5_0
PORT 26	-	P6_6	P6_5	P6_4	P6_3	P6_2	P6_1	P6_0
PORT 27	P7_7	P7_6	P7_5	P7_4	P7_3	P7_2	P7_1	P7_0
PORT 28	-	-	P8_5	P8_4	P8_3	P8_2	P8_1	P8_0
PORT 2A	PA_7	PA_6	PA_5	PA_4	PA_3	PA_2	PA_1	PA_0
PORT 2B	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1	PB_0
PORT 2C	-	-	-	-	-	PC_2	PC_1	PC_0
PORT 2D	PD_7	PD_6	PD_5	PD_4	PD_3	PD_2	PD_1	PD_0
PORT 2E	PE_7	PE_6	PE_5	PE_4	PE_3	PE_2	PE_1	PE_0
PORT 2F	-	-	-	-	-	PF_2	PF_1	PF_0
PORT 30	PG_7	PG_6	PG_5	PG_4	PG_3	PG_2	PG_1	PG_0
PORT 31	-	-	PH_5	PH_4	PH_3	PH_2	PH_1	PH_0
PORT 33	-	-	-	PJ_4	PJ_3	PJ_2	PJ_1	PJ_0
PORT 34	-	-	-	-	PK_3	PK_2	PK_1	PK_0
PORT 35	PL_7	PL_6	PL_5	PL_4	PL_3	PL_2	PL_1	PL_0
PORT 36	PM_7	PM_6	PM_5	PM_4	PM_3	PM_2	PM_1	PM_0
PORT 3C	-	-	-	-	PS_3	PS_2	PS_1	PS_0

-: unused pins

(*) when reading, the initial value is read. The written value will be ignored

1.2.2 Port Function Control

This module supports switching of each port function:

- GPIO function and peripheral function selection.
- Selection of each peripheral function.

1.2.3 Special Purpose Port Function Control

1.2.3.1 RZ/G2L, RZ/V2L and RZ/Five support settings of the following specific ports:

- IO voltage mode control: Ethernet channel 0/1, SDHI channel 0/1, QSPI.
- Buffer drive ability control: input enable for RIIC channel 0/1, ...
- Slew rate control.
- Pull up, pull down control.
- Ethernet MII/RGMII mode control.

1.2.3.2 RZ/V2N support settings of the following specific ports:

- IO voltage mode control: Ethernet channel 0/1, SDHI channel 0/1, XSPI, I3C.
- Buffer drive ability control: input enable for RIIC channel 0/1, I3C, ...
- Drive strength (renesas, output-impedance)
- Slew rate control.
- Pull up, pull down control.
- Open Drain Control.
- Schmitt Control.
- Ethernet MII/RGMII mode control.

1.2.3.3 RZ/G3E support settings of the following specific ports:

- Buffer drive ability control: input enable for SDHI, ...
- Drive strength (renesas, output-impedance)
- Slew rate control.
- Pull up, pull down control.
- Open Drain Control.
- Schmitt Control.
- Ethernet MII/RGMII mode control.

1.3 Reference

1.3.1 Standard

There is no supported standard in this module.

1.3.2 Related document

There is no related document in this module.

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
GPIO	General Purpose Input/Output

3. Operating Environment

3.1 Hardware Environment

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

Table 3-1 Hardware Environment (RZ/G2L, RZ/V2L, RZ/Five)

Name	Version	Manufacture
RZ/G2L Evaluation Board Kit	SMARC module: v01 Carrier board: v04	Renesas Electronics Europe GmbH
RZ/G2LC Evaluation Board Kit	SMARC module: v01 Carrier board: v04	Renesas Electronics Europe GmbH
RZ/G2UL Evaluation Board Kit	SMARC module: v01 Carrier board: v04	Renesas Electronics Europe GmbH
RZ/V2L Evaluation Board Kit	SMARC module: v01 Carrier board: v04	Renesas Electronics Europe GmbH
RZ/Five Evaluation Board Kit	SMARC module: v01 Carrier board: v04	Renesas Electronics Europe GmbH

Table 3-2 Hardware Environment (RZ/V2N, RZ/G3E)

Name	Product number
RZ/V2N Evaluation Kit	RTK0EF0186C03000BJ
RZ/G3E SMARC Evaluation Board Kit	RTK9947E57S01000BE

3.2 Module Configuration

The following figure shows the configuration of this module.

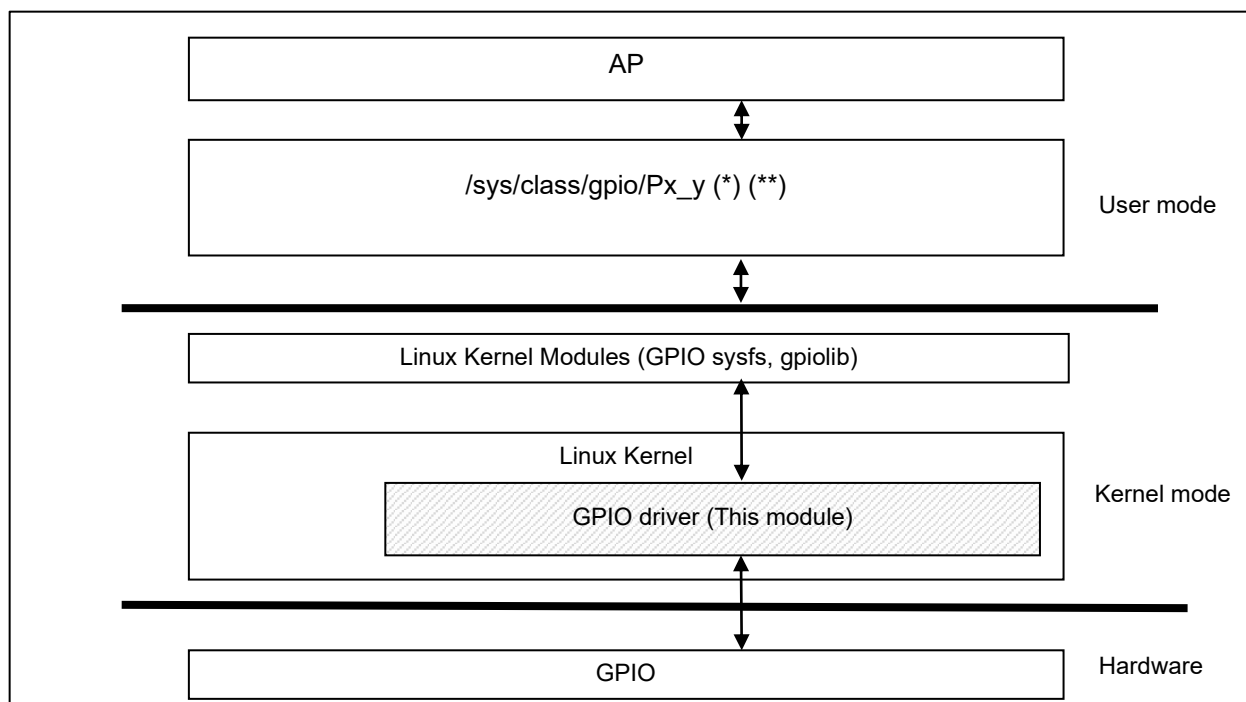


Figure 3-2 Module Configuration (RZ/G2L, RZ/V2L, RZ/V2N and RZ/G3E)

Note:

(*) **Px_y** with:

x: port number

y: pin position

Example:

- **P42_4** of RZ/G2L:
 - **Port:** 42
 - **Pin:** 4
- **P9_7** of RZ/V2N:
 - **Port:** 9
 - **Pin:** 7

(**) “120” is a GPIO based number created by GPIO Linux common framework for RZ/G2{L, LC} and RZ/V2L.

“360” is a GPIO based number created by GPIO Linux common framework for RZ/G2UL and RZ/Five.

“416” is a GPIO based number created by GPIO Linux common framework for RZ/V2N.

“280” is a GPIO based number created by GPIO Linux common framework for RZ/G3E.

GPIO number is determined by formula:

$$\text{GPIO_ID} = \text{GPIO_port} * 8 + \text{GPIO_pin} + \text{GPIO_based_number}.$$

Example:

- **P42_4** of RZ/G2L has its id **460** with above formula (**42 * 8 + 4 + 120**).
- **P2_1** of RZ/G2UL has its id **377** with above formula (**2 * 8 + 1 + 360**).
- **P9_7** of RZ/V2N has its id **495** with above formula (**9 * 8 + 7 + 416**).
- **P6_6** of RZ/G3E has its id **334** with above formula (**6 * 8 + 6 + 280**).

It is used to export a GPIO pin in userspace.

3.3 State Transition Diagram

There is no state transition diagram for this module.

4. External Interface

4.1 sysfs interface

The external interface of this module is based on Linux. The interface for operating GPIO pin from a user land is GPIO sysfs. Device node of this module is shown below.

Table 4-1 Example GPIO device nodes of RZ/G2{L, LC} and RZ/V2L

Name of pins	Device node
P0_0	/sys/class/gpio/P0_0
P0_1	/sys/class/gpio/P0_1
Unused pin	/sys/class/gpio/P0_2
:	:
P48_3	/sys/class/gpio/P48_3
P48_4	/sys/class/gpio/P48_4
:	:
Unused pin	/sys/class/gpio/P48_7

Table 4-2 Example GPIO device nodes of RZ/V2N

Name of pins	Device node
P0_0	/sys/class/gpio/P0_0
P0_1	/sys/class/gpio/P0_1
:	:
PA_0	/sys/class/gpio/PA_0
PB_5	/sys/class/gpio/PB_4
:	:
Unused pin	/sys/class/gpio/PB_6

Table 4-2 Example GPIO device nodes of RZ/G3E

Name of pins	Device node
P0_0	/sys/class/gpio/P0_0
P0_1	/sys/class/gpio/P0_1
:	:
PA_0	/sys/class/gpio/PA_0
PB_5	/sys/class/gpio/PB_5
:	:
Unused pin	/sys/class/gpio/PS_7

All GPIO pins are defined in **Table 4-1**, **Table 4-2** and **Table 4-3**.

They include available pins and unused pins which are described in **Table 1-1** and **Table 1-5**.

4.2 Available attributes for sysfs interface

Please refer <https://www.kernel.org/doc/Documentation/gpio/sysfs.txt> to know how to control GPIO correctly in userspace.

5. Integration

5.1 Directory Configuration

The directory configuration is shown below:

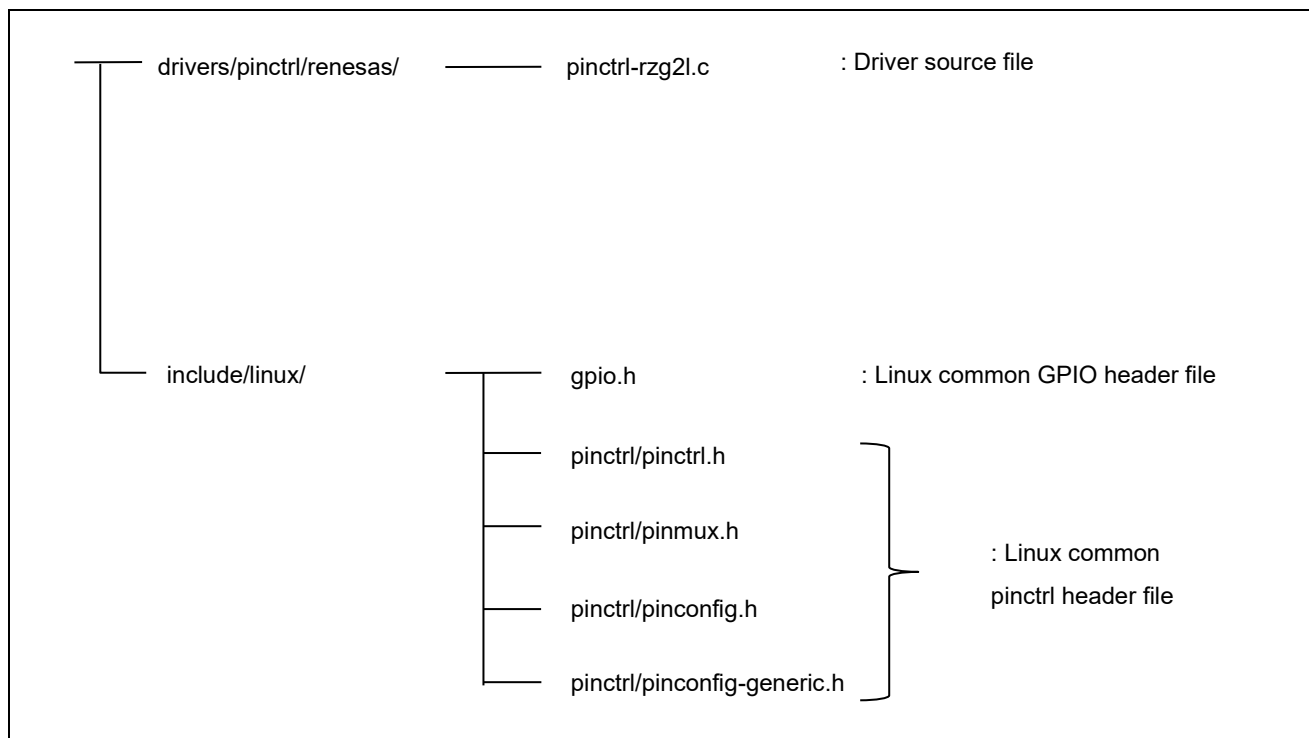


Figure 5-1 Directory configuration

5.2 Kernel configuration

To enable the function of this module, make the following setting in Kernel configuration:

```
Device Drivers --->
  <*> Pin controllers
```

Figure 5-2 Kernel configuration

Figure 5-3 lists the kernel config symbols to support GPIO for RZ/G2L, RZ/V2L Group, RZ/Five, RZ/V2N and RZ/G3E Group.

```
CONFIG_PINCTRL_PFC_RZG2L=y
CONFIG_GPIOLIB=y
CONFIG_GENERIC_PINCTRL_GROUPS=y
CONFIG_GENERIC_PINMUX_FUNCTIONS=y
CONFIG_GENERIC_PINCONF=y
```

Figure 5-3 Kernel config symbols

To use GPIO sysfs, make the following setting with Kernel Configuration:

```
-*- GPIO Support --->
[*] /sys/class/gpio/... (sysfs interface)
```

Figure 5-4 Kernel configuration for GPIO sysfs

5.3 Option Setting

5.3.1 Module Parameters

There are no module parameters.

5.3.2 Kernel Parameters

There are no kernel parameters.

5.4 Devicetree settings.

5.4.1 Device node definition

The device node of this module is defined in LSI devicetree file.

Example in **r9a07g044.dtsi for RZ/G2L** and **r9a07g054.dtsi for RZ/V2L**:

```
pinctrl: pin-controller@11030000 {
    compatible = "renesas,r9a07g044-pinctrl";
    reg = <0 0x11030000 0 0x10000>, <0 0x110a0020 0 0x30>; // 1st area is pinctrl registers, 2nd are is used for
                                                    setting gpio interrupt
    gpio-controller; // Marks the device node as a gpio controller.
    #gpio-cells = <2>; // 1st cell is GPIO number and the 2nd cell specifies GPIO flags.
    gpio-ranges = <&pinctrl 0 0 392>; // Range of pins managed by the GPIO controller.
    clocks = <&cpg CPG_MOD R9A07G044_GPIO_HCLK>;
}
```

Figure 5-5 Example of device node.

Example in **r9a09g056.dtsi for RZ/V2N**:

```
pinctrl: pinctrl@10410000 {
    compatible = "renesas,r9a09g056-pinctrl";
    reg = <0 0x10410000 0 0x10000>; // 1st area is pinctrl registers.
    gpio-controller; // Marks the device node as a gpio controller.
    #gpio-cells = <2>; // 1st cell is GPIO number and the 2nd cell specifies GPIO flags.
    gpio-ranges = <&pinctrl 0 0 96>; // Range of pins managed by the GPIO controller.
}
```

Figure 5-6 Example of device node.

Example in **r9a09g047.dtsi for RZ/G3E**:

```
pinctrl: pinctrl@10410000 {
    compatible = "renesas,r9a09g047-pinctrl";
    reg = <0 0x10410000 0 0x10000>; // 1st area is pinctrl registers.
    gpio-controller; // Marks the device node as a gpio controller.
    #gpio-cells = <2>; // 1st cell is GPIO number and the 2nd cell specifies GPIO flags.
    gpio-ranges = <&pinctrl 0 0 232>; // Range of pins managed by the GPIO controller.
}
```

Figure 5-7 Example of device node.

Note: All information in above device tree is used for both RZ/G2L, RZ/V2L, RZ/V2N and RZ/G3E except:

compatible: "renesas,r9a07g044-pinctrl" is used for R9A07G044L (RZ/G2L), "renesas,r9a07g054-pinctrl" is used for R9A07G054L (RZ/V2L), "renesas,r9a09g056-pinctrl" " is used for R9A09G056 (RZ/V2N) and "renesas,r9a09g047-pinctrl" is used for R9A09G047 (RZ/G3E).

clocks: R9A07G044_GPIO_HCLK is used for RZ/G2L, R9A07G054_GPIO_HCLK is used for RZ/V2L, R9A09G056_IOTOP_0_SHCLK is used for RZ/V2N and R9A09G047_IOTOP_0_SHCLK is used for RZ/G3E.

5.4.2 GPIO definition

A definition of the GPIO Pins is described on device tree. The examples of device tree are as follows:

- Define an output pin (**P42_2**) with high value.

```
canfd0_en {
    gpio-hog;
    gpios = <RZG2L_GPIO(42, 2) GPIO_ACTIVE_LOW>;
    output-high;
    line-name = "canfd0_en";
};
```

- Define an output pin (**P39_1**) to control regulator voltage based on its value

```
vccq_sdhi1: regulator-vccq-sdhi1 {
    compatible = "regulator-gpio";

    regulator-name = "SDHI1 VccQ";
    regulator-min-microvolt = <1800000>;
    regulator-max-microvolt = <3300000>;

    gpios = <&pinctrl RZG2L_GPIO(39, 1) GPIO_ACTIVE_HIGH>;

    gpios-states = <1>;
    states = <3300000 1 1800000 0>;
};
```

The format of a "gpios" property is as follows:

- The 1st cell is a node or label of GPIO device to be used. Here is "pinctrl".
- The 2nd cell contains the identifying number for the GPIO Pin in the node. Here is defined by a macro "**RZG2L_GPIO(port, pin)**" in included file "**include/dt-bindings/pinctrl/pinctrl-rzg2l.h**".
- The 3rd cell is the GPIO flags as defined in "**include/dt-bindings/gpio/gpio.h**", encoded as follows:
 - GPIO_ACTIVE_HIGH** = active high level-sensitive
 - GPIO_ACTIVE_LOW** = active low level-sensitive
- Define a pin (**P1_0**) used for interrupt detection RZ/G2L, RZ/V2L and RZ/Five.

```
&eth0 {
    ...

    phy0: ethernet-phy@7 {
        ...
        interrupt-parent = <&pinctrl>;
        interrupts = <RZG2L_GPIO(1, 0) IRQ_TYPE_EDGE_FALLING>;
    };
};
```

- Define a pin (**P7_1**) used for interrupt detection RZ/V2N.

```
&i2c3 {
    ...

    adv7535: hdmi@3d {
        ...
        interrupt-parent = <&pinctrl>;
        interrupts = <RZG2L_GPIO(7, 1) IRQ_TYPE_LEVEL_LOW >;
    };
};
```

The format of a "interrupts" property is as follows:

- The "interrupt-parent" property points to GPIO's node to be used,
- The 1st cell contains the identifying number for the GPIO Pin in the node. Here is defined by a macro "**RZG2L_GPIO(port, pin)**" in included file "**include/dt-bindings/pinctrl/pinctrl-rzg2l.h**".
- The 2nd cell is the INTERRUPT flags as defined in "**include/dt-bindings/interrupt-controller/irq.h**", encoded as follows:
 - **IRQ_TYPE_EDGE_FALLING** = active when falling edge-sensitive
 - **IRQ_TYPE_EDGE_RISING** = active when rising edge-sensitive
 - **IRQ_TYPE_LEVEL_LOW** = active when low level triggered
 - **IRQ_TYPE_LEVEL_HIGH** = active when high level triggered

5.4.3 Pin control definition

Below figure shows an example of the Pin Control definition in device tree.

- **Pin configuration definition**

```
&pinctrl {
    i2c1_pins: i2c1 {
        pins = "RIIC1_SDA", "RIIC1_SCL";
        input-enable;
    };

    i2c3_pins: i2c3 {
        pinmux = <RZG2L_PORT_PINMUX(18, 0, 3)>, /* SDA */
                <RZG2L_PORT_PINMUX(18, 1, 3)>; /* SCL */
    };
};
```

Figure 5-6 Example of device tree for pin configuration definitions.

The pin configuration required properties:

- Pin is multiplexed with several peripheral functions:
 - pinmux: A list of macros describes the port, bit and function of the pins.
Example: RZG2L_PORT_PINMUX(18, 0, 3) means 18th port, 0th bit and 3rd function.
- Pin has a unique function:
 - pins: A list of strings describes the name of pins.

- **Using the pin configuration definitions**

```
&i2c3 {
    pinctrl-0 = <&i2c3_pins>;
    pinctrl-names = "default";
    ...
};
```

- **Figure 5-7 Examples of device tree for pin configuration definitions**

Supported pin configuration parameters

The pin configuration parameters use the generic pinconf bindings defined in

“Documentation/devicetree/bindings/pinctrl/pinctrl-bindings.txt”. The supported parameters are power-source and input-enable. For pins that have a configurable I/O voltage, the power-source value should be the nominal I/O voltage in millivolts.

- **Power-source parameter.**

```
&pinctrl {
    sdhi1_pins: sd1 {
        sd1_data {
            pins = "SD1_DATA0", "SD1_DATA1", "SD1_DATA2", "SD1_DATA3";
            power-source = <3300>;
        };

        sd1_ctrl {
            pins = "SD1_CLK", "SD1_CMD";
            power-source = <3300>;
        };
    };

    sdhi1_pins_uhs: sd1_uhs {
        sd1_data_uhs {
            pins = "SD1_DATA0", "SD1_DATA1", "SD1_DATA2", "SD1_DATA3";
        };
    };
};
```

```

        power-source = <1800>;
    };

    sd1_ctrl_uhs {
        pins = "SD1_CLK", "SD1_CMD";
        power-source = <1800>;
    };
};

```

Figure 5-8 Examples of device tree for pin power-source parameter

- pins:
Select pins to specify I/O voltage:
- power-source:
Set a value to specify the I/O voltage:
 - 1800: set 1.8 voltage.
 - 3300: set 3.3 voltage.

- **Input-enable parameter.**

```

&pinctrl {

    i2c1_pins: i2c1 {
        pins = "RIIC1_SDA", "RIIC1_SCL";
        input-enable;
    };
};

```

Figure 5-9 Examples of device tree for pin input-enable parameter

- pins:
Select pins to specify input buffer capability:
- input-enable:
Select pin to enable input buffer capability.

- **Pull up parameter:**

```

&pinctrl {

    i2c1_pins: i2c1 {
        pinmux = <RZG2L_PORT_PINMUX(3, 2, 1)>, /* SDA */
                <RZG2L_PORT_PINMUX(3, 3, 1)>; /* SCL */
        bias-pull-up;
    };
};

```

Figure 5-10 Examples of device tree for pin bias-pull-up parameter

- pins:
Select pins to specify pull-up capability.
- bias-pull-up:
Select pin to enable pull-up capability.

- **Open-Drain parameter (RZ/G3E):**

```
rsci_i2c9_pins: rsci_i2c9 {
    pinmux = <RZV2H_PORT_PINMUX(6, 6, 5)>,
            <RZV2H_PORT_PINMUX(6, 5, 5)>;
    drive-open-drain = <1>;
};
```

Figure 5-11 Examples of device tree for pin drive-open-drain parameter

- **Drive strength parameter (renesas, output-impedance for RZ/V2N, RZ/G3E)**

```
sdhi0_pins: sd0emmc {
    sd0-emmc-ctrl {
        pins = "SD0_CLK", "SD0_CMD";
        renesas,output-impedance = <3>;
    };

    sd0-iovs {
        pins = "QSD0_IOVS";
        renesas,output-impedance = <3>;
    };

    sd0-emmc-data {
        pins = "QSD0_DAT0", "QSD0_DAT1", "QSD0_DAT2", "QSD0_DAT3",
              "QSD0_DAT4", "QSD0_DAT5", "QSD0_DAT6", "QSD0_DAT7";
        renesas,output-impedance = <3>;
    };

    sd0-emmc-rst {
        pins = "SD0_RSTN";
        renesas,output-impedance = <3>;
    };
};
```

Figure 5-12 Examples of device tree for pin renesas,output-impedance parameter

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

Rev.	Date	Description	
		Page	Summary
0.50	Apr. 30, 2021	—	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 information about RZ/G2LC and RZ/G2UL
1.3	Mar. 31, 2022	—	Update document for BSP-3.0.0
1.4	May. 31, 2022	—	Correct information about RZ/G2UL GPIO support
1.5	Jun. 24, 2022	—	Add information about RZ/Five
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	Mar. 31, 2024	3	Add the latest GPIO pins information table for RZ/Five
1.10	May. 30, 2025	1	Add MPU information support for both kernel versions v5.10 and v6.1.
1.11	Jun. 30, 2025	—	Add RZ/V2N information
1.12	Jul. 22, 2025	1	Add RZ/G3E overview
		4	Add RZ/G3E pins table
		5	Add RZ/G3E port settings
		8	Add RZ/G3E hardware environment Update module configuration
		10	Add example GPIO device nodes of RZ/G3E
		13	Add example of pinctrl node of RZ/G3E
		18	Add Open-Drain parameter for RZ/G3E Add drive strength parameter for RZ/G3E. RZ/V2N

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

Publication Date: Rev. 1.12 Jul 22, 2025

Published by: Renesas Electronics Corporation

RZ/G2L Group, RZ/V2L Group, RZ/V2N Group,
RZG3E Group and RZ/Five



Renesas Electronics Corporation