

Linux Interface Specification Device Driver GPT

User's Manual: Software

RZ/G2L Group, RZ/V2L Group, RZ/G3S Group,
RZ/G3E Group, RZ/V2H Group and RZ/V2N 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 depend 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. 1.13 Mar 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 Micro processing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Micro processing 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 micro processing 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/V2H Group, RZ/V2N Group, RZ/G3E Group and RZ/G3S 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/V2H Group User's Manual: Hardware	---
		RZ/V2N Group User's Manual: Hardware	---
		RZ/G3E Group User's Manual: Hardware	---
		RZ/G3S Group User's Manual: Hardware	---
User's manual for Software	Software specifications (basic function of Linux kernel, memory maps, interrupt, clock, pins mux, device tree)	Linux interface Specification - Device Driver GPT	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
Hi-Z	High Impedance
I/O	Input/Output
LSB	Least Significant Bit
MSB	Most Significant Bit
PLL	Phase Locked Loop
PWM	Pulse Width Modulation

Table of Contents

1. Overview.....	1
1.1 Overview	1
1.2 Function	1
1.3 Connected Port	2
1.4 Reference.....	3
1.4.1 Standard.....	3
1.4.2 Related document	3
1.5 Restrictions	3
2. Terminology.....	4
3. Operating Environment.....	5
3.1 Hardware Environment	5
3.2 Module Configuration.....	6
4. External Interface.....	10
4.1 sysfs interface	10
5. Integration.....	16
5.1 Directory Configuration	16
5.2 Integration Procedure	16
5.2.1 Choosing output channel	16
5.2.2 Choosing POEG group	20
5.2.3 Kernel Configuration	25
5.3 Device Tree Setting	25
5.3.1 GPT node	25
5.3.2 POEG node	28
5.3.3 Documentation file in the kernel	29
5.4 Option Setting	29
5.4.1 Module Parameters	29
5.4.2 Kernel Parameters	29

1. Overview

1.1 Overview

This manual explains the driver module (this module) that controls the GPT timer on RZ/G2L Group, RZ/V2L Group, RZ/V2H Group, RZ/V2N Group, RZ/G3E Group and RZ/G3S Group. Detailed explanation is skipped because the interface of this module is based on Linux.

Note: Currently, the devices that support this function are RZ/G2L, RZ/G2LC, RZ/V2L, RZ/G3S, RZ/V2H, RZ/V2N and RZ/G3E. RZ/G2UL does not have the GPT module.

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

1.2 Function

This module controls the GPT timer on RZ/G2L Group, RZ/V2L Group, RZ/V2H Group, RZ/V2N Group, RZ/G3E Group, RZ/G3S Group. The following functionality is supported:

- Output continuous high-level width of signal.
- Single buffer output operation.
- Double buffer output operation.
- Dead time output operation.
- Input capture operation.
- Counting input operation.
- The settable range of periods is from 30ns to 2.15 seconds. This limitation is due to sysfs framework.
- The settable range of duty cycles is from 30ns to the value of period.
- Change polarity and idle state.
- Double output on single channel.
- Output pulse number can be specified.
- Port Output Enable for GPT (This function can enable via sysfs)
 1. Disable output via software.
 2. Disable output via input level detection.
 3. Disable output request from GPT.
 4. POEG pin can be used to reset counter when GPT operates in counting input.

Note: RZ/V2H and RZ/G3E currently supports 3 modes:

- Output continuous high-level width of signal.
- Single buffer output operation.
- Double buffer output operation.

1.3 Connected Port

The supported connector of this module is as follows.

Table 1.1 Supported connector.

Channel	Connector	Support status	Remark
GPT 0 (RZ/G3S)	Carrier board: GPIO8 & GPIO9	Yes	Output as PMD01_6A_PIN7 (Channel B) PMD01_6A_PIN8 (Channel A)
GPT 1	-	No	-
GPT 2	-	No	-
GPT 3 (RZ/G2LC)	SMARC module : CN7	Yes	Output as PMD01_PIN1 (channel A) PMD01_PIN4 (channel B)
GPT 4 (RZ/G2L)	SMARC module: CN4 Carrier board: J1	Yes	Output as PMD00_PIN7 (channel A) PMD00_PIN10 (channel B)
GPT 5	-	No	-
GPT 6	-	No	-
GPT 7	-	No	-
POEGA (RZ/G3S)	Carrier board: GPIO4	Yes	PMD00_2A_PIN7
POEGB	-	No	-
POEGC	-	No	-
POEGD	-	No	-

Table 1.2 Supported connector (RZ/V2H, RZ/V2N and RZ/G3E).

Channel	Connector	Support status	Remark
GPT0_0	-		
GPT0_1	-	Yes	Output as PMOD1_A6_PIN3(channel A) and PMOD1_A6_PIN4(channel B)
GPT0_2 (RZ/V2H and RZ/V2N)	-	Yes	Output as CN6_PIN8(channel A) and CN3_PIN8(channel B)
GPT0_3	-		
GPT0_4	-		
GPT0_5	-		
GPT0_6	-		
GPT0_7	-		
GPT1_0	-		
GPT1_1	-		
GPT1_2	-		
GPT1_3	-		
GPT1_4	-		
GPT1_5	-		
GPT1_6	-		
GPT1_7	-		
POEGA	-	Yes	
POEGB	-		
POEGC (RZ/V2N)	-	Yes	PMOD 6A PIN4
POEGD	-		

1.4 Reference

1.4.1 Standard

There is no supported standard in this module.

1.4.2 Related document

There is no related document.

1.5 Restrictions

There are no restrictions.

2. Terminology

The following table shows the terminology related to this module.

Table 2.1 Terminology

Terms	Explanation
PWM	Pulse width modulation
GPT	General PWM Timer
LVDS	Low voltage differential signaling
BKLT	Back light
POEG	Port Output Enable for GPT

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/G2L Evaluation Board Kit	RTK9744L23S01000BE
RZ/G2LC Evaluation Board Kit	RTK9744C22S01000BE
RZ/V2L Evaluation Board Kit	RTK9754L23S01000BE
RZ/G3S Evaluation Board Kit	RTK9845S33C01000BE
RZ/V2H Evaluation Board Kit	RTK0EF0168C04000BJ
RZ/V2N Evaluation Board Kit V1.0	RTK0EF0186C03000BJ
RZ/V2N Evaluation Board Kit V2.0	RTK0EF0186C03001BJ
RZ/G3E Evaluation Board Kit.	RTK9947E57S01000BE

3.2 Module Configuration

The following figure shows the configuration of this module.

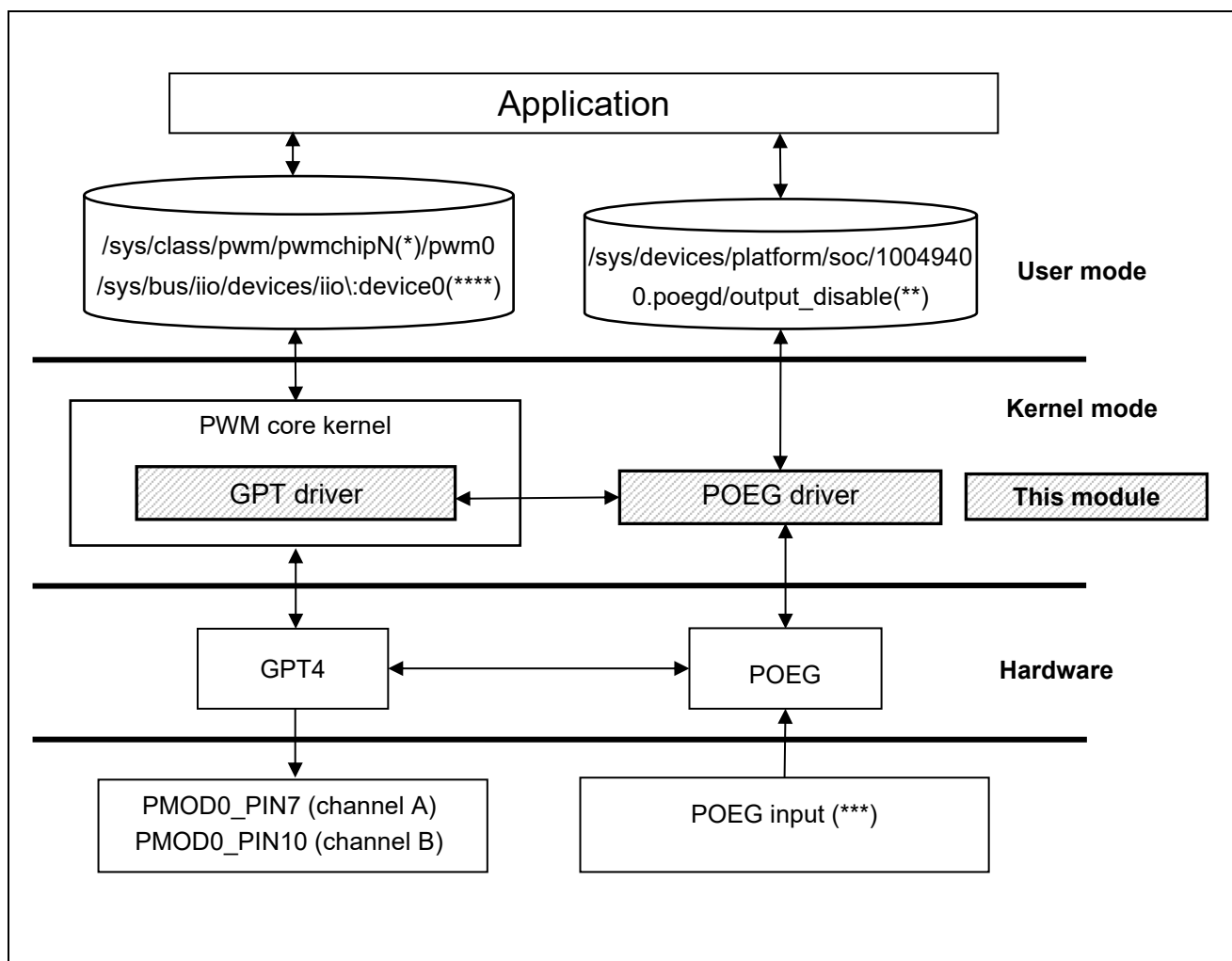


Figure 3-1 Module configuration (RZ/G2L, RZ/G2LC, RZ/V2L, RZ/G3S)

Note: (*) Each probed PWM controller/chip will be exported as pwmchipN, where N is the base of the PWM chip.

(**) Setting to this device sysfs depends on which POEG user is using. In this case, POEGD is using, user can see others poeg and address if others poeg is used.

(***) RZ/G2L Evaluation Board Kit don't have input pin for POEG.

(****) When enable without others iio device, `/sys/bus/iio/devices/iio\:deviceX` is device0, when enable with other iio device (like MTU), it is device1, user should check whether it is iio device for GPT, as picture it is enable without others device.

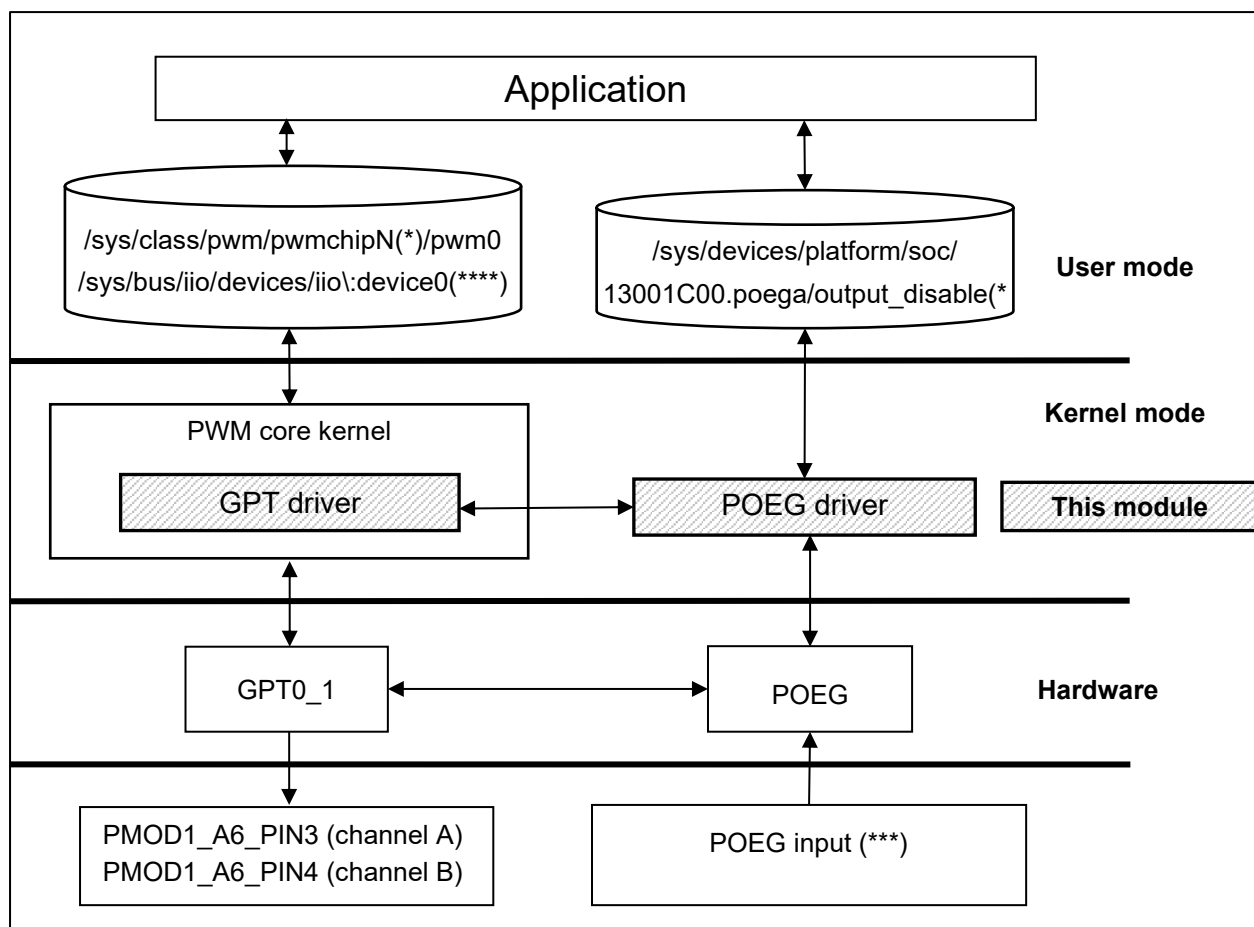


Figure 3-2 Module configuration RZ/G3E

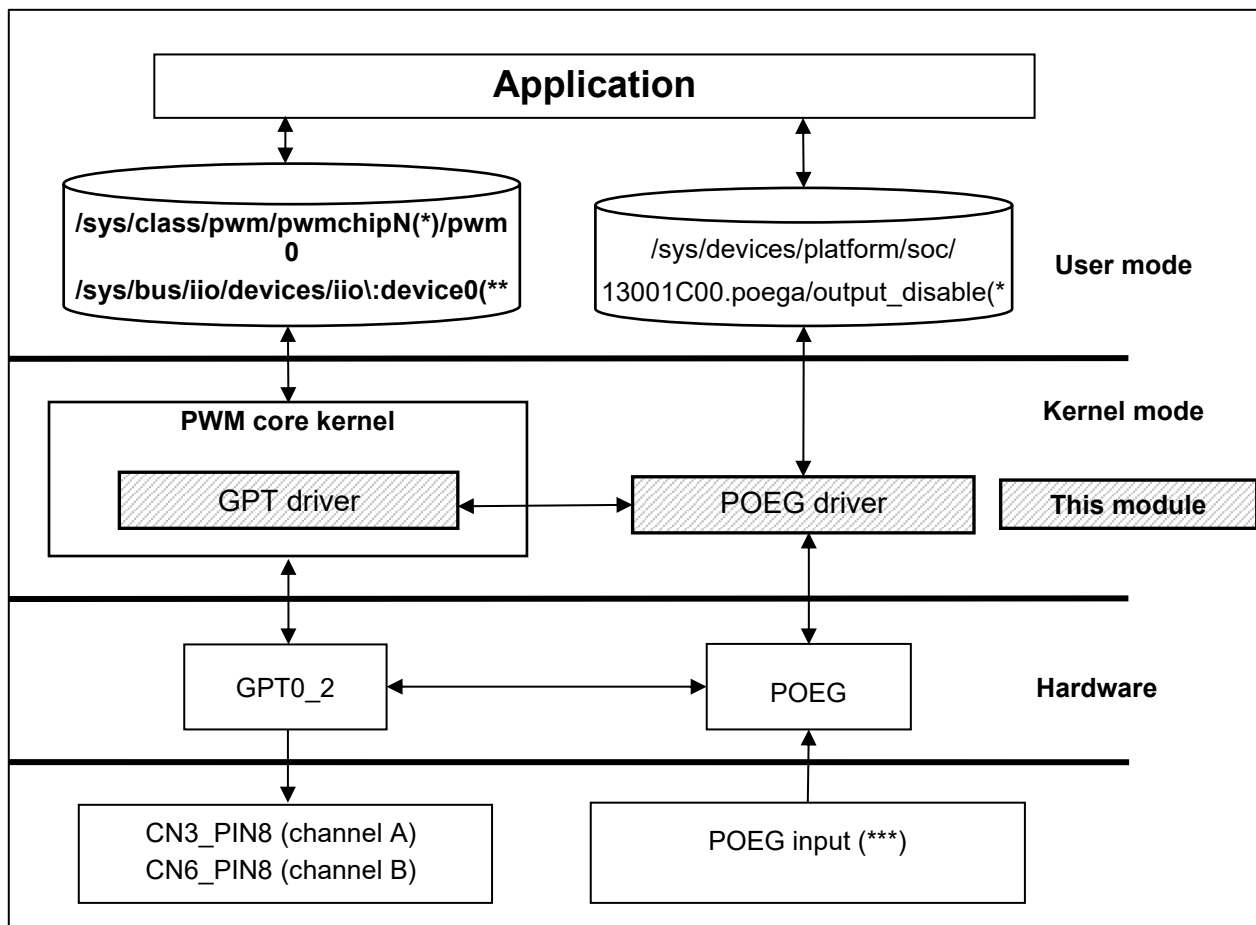


Figure 3-3 Module configuration RZ/V2H

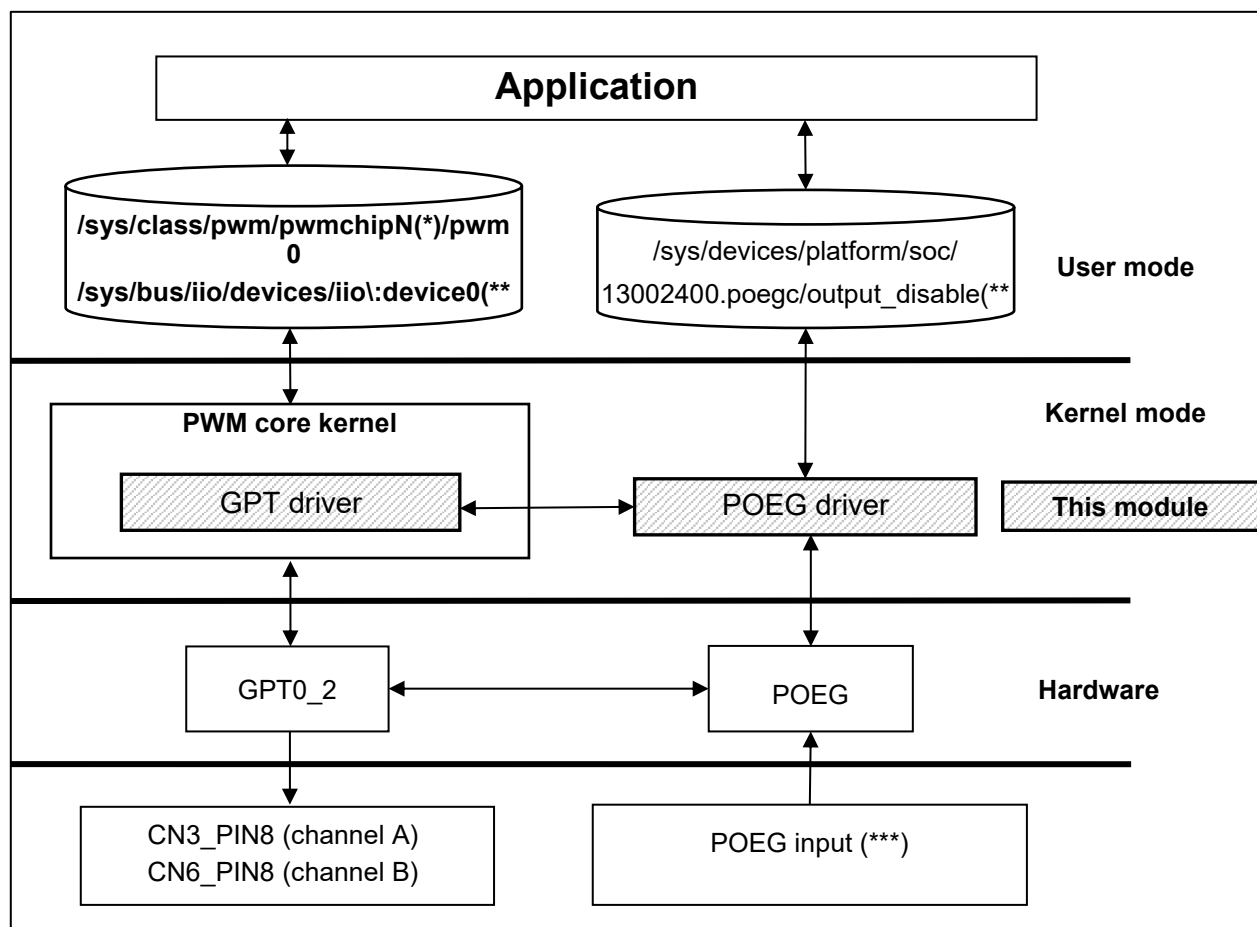


Figure 3-4 Module configuration RZ/V2N

4. External Interface

This section explains in the following format about the functions this module supplies.

4.1 sysfs interface

The external interface of this module is based on Linux. The interface for operating GPT from a user space is PWM sysfs for PWM operation and IIO sysfs for counting operation. Device node of this module is shown below.

Table 4.1 GPT device file

GPT	Device Node
GPT	/sys/class/pwm/pwmchip0/pwm0 /sys/bus/iio/devices/iio\:device0 (*)

The interface for operating POEG from a user space is created in sysfs. Device node of this module is shown below. Use this device node to control POEG via software.

(*) When enable without others iio device, /sys/bus/iio/devices/iio\:deviceX is device0, when enable with other iio device (like MTU), it is device1, user should check whether it is iio device for GPT, as picture it is enable without others device.

Table 4.2 POEG device file

POEG	Device Node
POEGA	/sys/devices/platform/soc/10048800.poega/output_disable
POEGB	/sys/devices/platform/soc/10048c00.poeqb/output_disable
POEGC	/sys/devices/platform/soc/10049000.poegc/output_disable
POEGD	/sys/devices/platform/soc/10049400.poegd/output_disable

The GPT sub-system provides a number of controls that can be accessed via /sys/class/pwm and /sys/bus/iio/devices.

Table 4.3 shows GPT specification in RZ/G2L and RZ/G3S. If you want to confirm the other interface, please see: <https://www.kernel.org/doc/Documentation/ABI/testing/sysfs-class-pwm>

Table 4.4 shows POEG specification in RZ/G2L and RZ/G3S.

Table 4.3 GPT sysfs device file

GPT Sysfs Interface	Description	Support status	Notes
/sys/class/pwm/pwmchip0/device/gpt_operation_available	Show all available GPT operation	Yes	This config is read-only.
/sys/class/pwm/pwmchip0/device/gpt_operation	Sets GPT operation	Yes	Setting GPT operation before setting others config. (*1)
period	Sets the GPT signal period in nanoseconds	Yes	Need to set over 30. (*2)
duty_cycle	Sets the GPT signal duty cycle in nanoseconds.	Yes	Need to set period, before it is set. (*3) Need to set over 30. (*2)
enable	Enable/ disable the GPT signal.	Yes	Set period and duty_cycle, before it is set 1.
polarity	Invert output signal	Yes	echo normal > /sys/class/pwm/pwmchip0/pwm0/polarity change polarity to high end low when compare match. echo inverse > /sys/class/pwm/pwmchip0/pwm0/polarity change polarity to low end high when compare match.
capture	Capture pulse	Yes	cat /sys/class/pwm/pwmchip0/pwm0/capture and then generate pulse to input pin to capture
/sys/class/pwm/pwmchip0/device/buffA0 /sys/class/pwm/pwmchip0/device/buffB0	Set first buffer value in single buffer operation mode. Set first buffer value in double buffer operation mode.	Yes	Must set period before set buffer (*3) (*4) buffA0 set for channel A. buffB0 set for channel B.
/sys/class/pwm/pwmchip0/device/buffA1 /sys/class/pwm/pwmchip0/device/buffB1	Set second buffer value in single buffer operation mode. Set second buffer value in double buffer operation mode.	Yes	Must set period before set buffer (*3) (*4) buffA1 set for channel A. buffB1 set for channel B.
/sys/class/pwm/pwmchip0/device/buffA2 /sys/class/pwm/pwmchip0/device/buffB2	Set third buffer value in double buffer operation mode	Yes	Must set period before set buffer (*3) (*4) buffA2 set for channel A. buffB2 set for channel B.

/sys/class/pwm/pwmchip0/device/deadtime_first	Set first half dead time	Yes	(*5)
/sys/class/pwm/pwmchip0/device/deadtime_second	Set second half dead time	Yes	(*5)
/sys/class/pwm/pwmchip0/device/polarityA /sys/class/pwm/pwmchip0/device/polarityB	Set polarity and idle state	Yes	Set polarity and idle state for channel A and channel B.
/sys/class/pwm/pwmchip0/device/pulse_number	Set output pulse number	Yes	Set number of GPT output pulse.
/sys/bus/iio/devices/iio\:device0/in_count_counter_mode_available	Show all available mode count for counting input operation	Yes	This config is read-only.
/sys/bus/iio/devices/iio\:device0/in_count0_counter_mode	Set counter mode	Yes	See hardware manual for more detail in each counter mode
/sys/bus/iio/devices/iio\:device0/in_count0_counter_preset	Set max counter value	Yes	Counter will operate in range 0 -> in_count0_counter_preset
/sys/bus/iio/devices/iio\:device0/in_count0_raw	Current value of counter	Yes	
/sys/bus/iio/devices/iio\:device0/in_count0_en	Enable counter	Yes	Enable before count

Note:

*1) Setting GPT operation first to use GPT, GPT has 6 operation, **default value** is normal_output:

normal_output, single_buffer_output, double_buffer_output, deadtime_output, input_capture, counting_input.

- normal_output: When the counter value matches the value set in duty cycle, the output from the associated pin GTIOCA or GTIOCB can be changed. At present, the normal_output operates in single-buffer mode, allowing the duty cycle to be updated sequentially while the channel is enabled.
- single_buffer_output: the buffer transfer is performed at an overflow (when counter reach period setting).
Ex: buffA1 will transfer to duty cycle buffA0, buffA0 will transfer to buffA1 => buffA0, buffA1 will become duty cycle sequentially.
When the counter value matches the current value in duty cycle, the output from the associated pin GTIOCA or GTIOCB can be changed.
- double_buffer_output: the buffer transfer is performed at an overflow (when counter reach period setting).
Ex: buffA2 will transfer to buffA1, buffA1 will transfer to duty cycle buffA0, buffA0 will transfer to buffA2 => buffA0, buffA1, buffA2 will become duty cycle sequentially.

When the counter value matches the current value in duty cycle, the output from the associated pin GTIOCA or GTIOCB can be changed.

- `deadtime_output`: This operation make channel A and channel B can not change at the same time, user can set time between channel A changing and channel B changing (deadtime)
- `input_capture`: This operation measure duty and period of input pulse
- `counting_input`: This operation uses GPT timer as a counter that count base on input pulse. Usually use for counting encoder.

Ex:

```
echo counting_input > /sys/class/pwm/pwmchip0/device/gpt_operation
```

*2) The value of period and `duty_cycle` have some errors due to round up between calculations, we should set over this value to reduce effect of error calculation (± 1 tick in register equivalent ± 10 ns).

*3) When setting period, driver calculate prescale for GPT clock and detect frequency operating range for GPT module. `Duty_cycle`, buffer 0 and buffer 1 based on this range and cannot set prescale by itself.

*4) Must set greater than 1000ns because if not do that, next buffer value will not affect immediately, it will affect after a few cycles.

Must set buffer in order period -> buff0 -> buff1....

Ex:

```
echo 50000000 > /sys/class/pwm/pwmchip0/pwm0/period
```

```
echo 20000000 > /sys/class/pwm/pwmchip0/device/buff0
```

```
echo 40000000 > /sys/class/pwm/pwmchip0/device/buff1
```

*5) In this mode buffA1 is first half compare match and buffA2 is second half compare match

When switch in/out this mode, you should stop pwm to prevent unexpected output

Channel B is set automatically, you can read that value via `cat /sys/class/pwm/pwmchip0/device/buffB0`

Example using dead time mode:

```
echo deadtime_output > /sys/class/pwm/pwmchip0/device/gpt_operation (Enable dead time mode)
```

```
echo 50000 > /sys/class/pwm/pwmchip0/pwm0/period (Set period)
```

```
echo 15000 > /sys/class/pwm/pwmchip0/device/buffA1 (Set first half compare match)
```

```
echo 35000 > /sys/class/pwm/pwmchip0/device/buffA2 (Set second half compare match)
```

```
echo 5000 > /sys/class/pwm/pwmchip0/device/deadtime_first (Set dead time for first half)
```

```
echo 3000 > /sys/class/pwm/pwmchip0/device/deadtime_second (Set dead time for second half)
```

Table 4.4 POEG sysfs device file

POEG Sysfs Interface	Description	Support status	Notes
/sys/class/pwm/pwmchip0/device/POEG_available	Show all available POEG can be used	Yes	This config is read-only (*1)
/sys/class/pwm/pwmchip0/device/POEG	Set which POEG to use	Yes	Ex: echo POEGA > /sys/class/pwm/pwmchip0/device/POEG (*2)
/sys/devices/platform/soc/10048800.poeга/output_disable	Disable GPT output via software	Yes	Echo 1 > /sys/devices/platform/soc/10048800.poeга/output_disable disable GPT output Echo 0 > /sys/devices/platform/soc/10048800.poeга/output_disable enable GPT output
/sys/devices/platform/soc/10048c00.poeгb/output_disable	Disable GPT output via software	Yes	Echo 1 > /sys/devices/platform/soc/10048c00.poeгb/output_disable disable GPT output Echo 0 > /sys/devices/platform/soc/10048c00.poeгb/output_disable enable GPT output
/sys/devices/platform/soc/10049000.poeгc/output_disable	Disable GPT output via software	Yes	Echo 1 > /sys/devices/platform/soc/10049000.poeгc/output_disable disable GPT output Echo 0 > /sys/devices/platform/soc/10049000.poeгc/output_disable enable GPT output
/sys/devices/platform/soc/10049400.poeгd/output_disable	Disable GPT output via software	Yes	Echo 1 > /sys/devices/platform/soc/10049400.poeгd/output_disable disable GPT output Echo 0 > /sys/devices/platform/soc/10049400.poeгd/output_disable enable GPT output
/sys/bus/iio/devices/iio\:device0/in_count_reset_counter_available	Show all available reset counter can be used	Yes	This config is read-only (*3)
/sys/bus/iio/devices/iio\:device0/in_count0_reset_counter	Set which pin to use as reset counter	Yes	Ex: echo GTETRGA > /sys/bus/iio/devices/iio\:device0/in_count0_reset_counter (*4)

Note:

*1) POEG now can be choose to use or not via sysfs.

User can check which POEG can be used by enter command:

```
dmesg | grep poeg
```

```
dmesg | grep gpt
```

Output as below: (In below picture poega, poegb, poegc, poegd can be used)

```

root@smarc-rzv2l:~# dmesg | grep poeg
[ 0.201992] poeg-rzg2l 10048800.poega: RZ/G2L POEG Driver probed
[ 0.202307] poeg-rzg2l 10048c00.poegb: RZ/G2L POEG Driver probed
[ 0.202584] poeg-rzg2l 10049000.poegc: RZ/G2L POEG Driver probed
[ 0.202860] poeg-rzg2l 10049400.poegd: RZ/G2L POEG Driver probed
root@smarc-rzv2l:~# dmesg | grep gpt
[ 0.200712] gpt-rzg2l 10048400.gpt: Can use GTETRGA as POEG, reset_counter
[ 0.200755] gpt-rzg2l 10048400.gpt: Can use GTETRGA as POEG, reset_counter
[ 0.200775] gpt-rzg2l 10048400.gpt: Can use GTETRC as POEG, reset_counter
[ 0.200795] gpt-rzg2l 10048400.gpt: Can use GTETRGD as POEG, reset_counter

```

Or can be checked by command: `cat /sys/class/pwm/pwmchip0/device/POEG_available`

POEG has 3 function:

1. Disable output via software:

Control via `/sys/devices/platform/soc/10048800.poega/output_disable` if POEGA is chosen

Control via `/sys/devices/platform/soc/10048c00.poegb/output_disable` if POEGB is chosen

Control via `/sys/devices/platform/soc/10049000.poegc/output_disable` if POEGC is chosen

Control via `/sys/devices/platform/soc/10049400.poegd/output_disable` if POEGD is chosen

Note: Currently RZ/V2H and RZ/G3E only support this mode.

2. Disable output via input level detection:

When input pin of POEG receive high level signal, GPT output is disabled

But RZG2L Evaluation Board Kit **don't have pin** for this function.

3. Disable output request from GPT (this mode only work in when GPT in double output mode):

When 2 output pin of GPT have the same level, GPT output is disabled

In dead time mode, POEG not disable output have the same output logic signal, it only disable when there is deadtime error

*2) If user don't want to use POEG:

`echo NOT_USE > /sys/class/pwm/pwmchip0/device/POEG`

*3) reset counter is used for clear counter when GPT operate in counting input operation

This input pin is using the same POEG pin.

User can check which reset counter pin can be used by command:

`cat /sys/bus/iio/devices/iio\:device0/in_count_reset_counter_available`

(In below picture poega, poegb, poegc, poegd pin can be used as reset counter)

```

root@smarc-rzv2l:~# cat /sys/bus/iio/devices/iio\:device1/in_count_reset_counter_available
NOT_USE GTETRGA GTETRGA GTETRC GTETRGD

```

*4) If user don't want to use reset counter:

`echo NOT_USE > /sys/bus/iio/devices/iio\:device0/in_count0_reset_counter`

5. Integration

5.1 Directory Configuration

The directory configuration is shown below.

—— drivers/pwm	—— gpt-rzg2l.c	: GPT timer driver source file
	—— poeg-rzg2l.c	: POEG driver source file

Figure 5-1 Directory configuration

5.2 Integration Procedure

5.2.1 Choosing output channel

Note*: This feature is supported starting from version v6.1-cip43 and after.

The following description explains how to choose GPT output channel in RZ/G2L, RZ/V2L, RZ/G2LC, RZ/V2H and RZ/V2N.

The following description indicates a difference from the default setting. ("−": Delete a description (default setting), "+": Setting after the modification))

file: arch/arm64/boot/dts/renesas/rzg2l-smarc.dtsi for RZ/G2L

arch/arm64/boot/dts/renesas/rzg2l-smarc.dtsi for RZ/G2LC

arch/arm64/boot/dts/renesas/rzv2h-evk-common.dtsi for RZ/V2H

arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dtsi for RZ/V2N

```
&gpt4 {
    pinctrl-0 = <&gpt4_pins>;
    pinctrl-names = "default";
    #if (GPT4_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT4_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
};
```

Figure 5-2 Choosing output channel (RZ/G2L)

```

#if (SCIF1_GPT3)
&gpt3 {
    pinctrl-0 = <&gpt3_pins>;
    pinctrl-names = "default";
    #if (GPT3_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT3_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
    #if (POEGD_SUPPORT)
        poeg = <&poega &poegb &poegc &poegd>;
    #endif
    status = "okay";
};
#endif

```

Figure 5-3 Choosing output channel (RZ/G2LC)

```

&gpt0_2 {
    pinctrl-0 = <&gpt0_2_pins>;
    pinctrl-names = "default";
    #if (GPT0_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT0_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
    #if (I2C7_POEG_SEL)
        poeg = <&poega0>;
    #endif
    status = "okay";
};

```

Figure 5-4 Choosing output channel (RZ/V2H)

```

&gpt0_2 {
    pinctrl-0 = <&gpt0_2_pins>;
    pinctrl-names = "default";
    #if (GPT0_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT0_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
    #if (I2C2_I3C_POEG_SEL == POEG_SEL)
        poeg = <&poegc0>;
    #endif
    status = "okay";
};

```

Figure 5-5 Choosing output channel (RZ/V2N)


```

* SW1-3 : SW_SCIF_CAN          (1: CAN1; 0: SCIF1)*
#define SW_SCIF_CAN            0

#if (SW_SCIF_CAN)

/* Due to HW routing, SW_RSPI_CAN is always 0 when SW_SCIF_CAN is set to 1 */
#define SW_RSPI_CAN            0

/* Enable GPT3 when SCIF1 disabled */
#define SCIF1_GPT3              1

#else

/* Please set SW_RSPI_CAN. Default value is 1 */
#define SW_RSPI_CAN            1
#define SCIF1_GPT3              0

#endif

```

Figure 5-6 Define marco enable GPT3 for RZ/G2LC

Note : This SCIF1_GPT3 depends on SW_SCIF_CAN. Configure SW_SCIF_CAN to enable GPT3.

```

/*
* GPT4_CHANNEL_SEL:
*
*     GPT_CHANNEL_A
*
*     GPT_CHANNEL_B
*
*     BOTH_CHANNEL (default)
*/
#define GPT4_CHANNEL_SEL        BOTH_CHANNEL

```

Figure 5-7 Define marco switch GPT4 channel (RZ/G2L)

Note: This is an example of configuring the channel for RZ/G2L. Apply the same approach for the other boards using their respective DTS files:

RZ/G2L: arch/arm64/boot/dts/renesas/r9a07g044l2-smarc.dts

RZ/V2L: arch/arm64/boot/dts/renesas/r9a07g054l2-smarc.dts

RZ/G2LC: arch/arm64/boot/dts/renesas/r9a07g044c2-smarc.dts

RZ/V2N: arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dts

RZ/V2H: arch/arm64/boot/dts/renesas/r9a09g057h44-rzv2h-evk-ver2.dts

file: arch/arm64/boot/dts/renesas/rzg3e-smarc.dtsi for RZ/G3E

```
#if (I2C1_GPT_SEL)

&gpt0_1 {

    pinctrl-0 = <&gpt0_1_pins>;

    pinctrl-names = "default";

    channel = "both_AB";

    poeg = <&poega>;

    status = "okay";

};
```

Figure 5-8 Choosing output channel (RZ/G3E)

Note*: Hardware design of GPT0_1 for RZ/G3E interface have some pins function control conflict with others function. Define switch macro to choose GPT0_1 function for pins (arch/arm64/boot/dts/renesas/r9a09g047e57-smarc.dts)

```
/*
 * I2C1 and GPT (PMOD) selection
 * 0 - I2C1
 * 1 - GPT0_1 on PMOD0
 */
#define I2C1_GPT_SEL 0
```

Figure 5-9 Define macro switch for GPT0_1 (RZ/G3E)

file: arch/arm64/boot/dts/renesas/rzg3s-smarc.dtsi for RZ/G3S

```
&gpt0 {
    pinctrl-0 = <&gpt0_pins>;

    pinctrl-names = "default";

    channel = "both_AB";

    poeg = <&poega>;

    status = "okay";
};
```

Figure 5-10 Choosing output channel (RZ/G3S)

- In single output mode, GPT on boards only output **one** pin port so you must choose which channel corresponding to that pin (channel A or channel B).

Ex: For RZG2L Evaluation Board Kit, choose channel A as below:

```
channel = "channel_A";
```

For choosing channel B:

```
channel = "channel_B";
```

- In double output mode, GPT on boards output **two** pin port. Choose this mode as below:
channel = "both_AB";
- Remember to choose 2 correspond pin in pinctrl-0 when using double output mode.
- Remember when using GPT as backlight, setting channel = "channel_A" or channel = "channel_B" because in "both_AB" mode, must set duty cycle of gpt via /sys/class/pwm/pwmchip0/device/buffA0 or /sys/class/pwm/pwmchip0/device/buffB0, but in backlight driver it set duty cycle via /sys/class/pwm/pwmchip0/pwm0/duty_cycle.

5.2.2 Choosing POEG group

The following description explains how to choose POEG group in RZ/G2L, RZ/V2L, RZ/V2H, RZ/V2N, RZ/G3E and RZ/G3S.

The following description indicates a difference from the default setting. ("-" : Delete a description (default setting), "+" : Setting after the modification))

file: arch/arm64/boot/dts/renesas/rzg2l-smarc-som.dtsi for RZG2L, arch/arm64/boot/dts/renesas/rzg2l-smarc.dtsi for RZG2LC, arch/arm64/boot/dts/renesas/rzg3e-smarc.dtsi for RZ/G3E, arch/arm64/boot/dts/renesas/r9a09g057h44-rzv2h-evk-ver2.dts for RZ/V2H and arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dts for RZ/V2N.

```

+&poega {
+    status = "okay";
+};

+&poegb {
+    status = "okay";
+};

+&poegc {
+    status = "okay";
+};

+&poegd {
+    status = "okay";
+};

&gpt4 {
    pinctrl-0 = <&gpt4_pins>;
    pinctrl-names = "default";
    channel = "channel_A";
+    poeg = <&poega &poegb &poegc &poegd >;
    status = "okay";
};

```

Figure 5-11 Choosing POEG group (RZ/G2L)

file: arch/arm64/boot/dts/renesas/rzg3e-smarc.dtsi for RZ/G3E

```

#if (I2C1_GPT_SEL)
&gpt0_1 {
    pinctrl-0 = <&gpt0_1_pins>;
    pinctrl-names = "default";
    channel = "both_AB";
    poeg = <&poega>;
    status = "okay";
};

&poega {
    pinctrl-0 = <&poega_pins>;
    pinctrl-names = "default";
    status = "okay";
};

#endif

```

Figure 5-12 Choosing POEG group (RZ/G3E)

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

```

#if (I2C7_POEG_SEL)
&poega0 {
    pinctrl-0 = <&poega0_pins>;
    pinctrl-names = "default";
    status = "okay";
};
#endif

&gpt0_2 {
    pinctrl-0 = <&gpt0_2_pins>;
    pinctrl-names = "default";
    #if (GPT0_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT0_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
    #if (I2C7_POEG_SEL)
        poeg = <&poega0>;
    #endif
    status = "okay";
};

```

Figure 5-13 Choosing POEG group (RZ/V2H)

file arch/arm64/boot/dts/renesas/r9a09g057h44-rzv2h-evk-ver2.dts

```

#define I2C7_POEG_SEL                0
* I2C7 and POEG selection
*      1 - POEG                      0 - I2C7 (default)

```

Figure 5-14 Define macro switch for POEGA (RZ/V2H)

file: arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dtsi

```

#if (I2C2_I3C_POEG_SEL == POEG_SEL)
&poegc0 {
    pinctrl-0 = <&poegc0_pins>;
    pinctrl-names = "default";
    status = "okay";
};
#endif

&gpt0_2 {
    pinctrl-0 = <&gpt0_2_pins>;
    pinctrl-names = "default";
    #if (GPT0_CHANNEL_SEL == GPT_CHANNEL_A)
        channel = "channel_A";
    #elif (GPT0_CHANNEL_SEL == GPT_CHANNEL_B)
        channel = "channel_B";
    #else
        channel = "both_AB";
    #endif
    #if (I2C2_I3C_POEG_SEL == POEG_SEL)
        poeg = <&poegc0>;
    #endif
    status = "okay";
};

```

Figure 5-15 Choosing POEG group (RZ/V2N)

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

```
/*
 * Enable I2C2, I3C or POEG on pins 20, 21.
 * I2C2_I3C_POEG_SEL
 * - POEG    0
 * - I2C2    1 (default)
 * - I3C     2
 */
#define POEG_SEL        0
#define I2C2_SEL        1
#define I3C_SEL         2
#define I2C2_I3C_POEG_SEL  I2C2_SEL
```

Figure 5-16 Define macro switch for POEGC (RZ/V2N)

file: arch/arm64/boot/dts/renesas/rzg3s-smarc.dtsi

```

&gpt0 {
    pinctrl-0 = <&gpt0_pins>;
    pinctrl-names = "default";
    channel = "both_AB";
    poeg = <&poega>;
    status = "okay";
};

#if POEGA_SEL
&poega {
    pinctrl-0 = <&poega_pins>;
    pinctrl-names = "default";
    status = "okay";
};
#endif

```

Figure 5-17 Choosing POEG group (RZ/G3S)

file: arch/arm64/boot/dts/renesas/r9a08g045s33-smarc.dts

```

/*
 * POEGA selection states:
 * @POEGA_SEL:
 *     0 - Select USER_SW1 (default)
 *     1 - Select POEGA for gpt0 function
 */
#define POEGA_SEL    0

```

Figure 5-18 Define macro switch for POEGA (RZ/G3S)

There is 4 type of poeg : poega, poegb, poegc, poegd user can chose which POEG, reset counter to use via sysfs.

- Always set poeg = <&poega &poegb &poegc &poegd > in gpt node , if don't want use what poeg, set status = "disabled" in that poeg node.
- For RZG2L Evaluation Board Kit, there is no input pin for poeg, so user can add as Figure 5.2

- For RZG2L Evaluation Board Kit, for easier using, user can enable poeg by set POEGD_SUPPORT to 1 in arch/arm64/boot/dts/renesas/rzg2l-smarc.dtsi
- For RZG3E SMARC Evaluation Board Kit, there is no input pin for poeg, so user can add as Figure 5.12.
- For RZV2H Evaluation Board Kit, the POEG has an input pin, so user can add as Figure 5.13.
- For RZV2H Evaluation Board Kit, for easier using, user can enable poeg by set I2C7_POEG_SEL to 1 in arch/arm64/boot/dts/renesas/r9a09g057h44-rzv2h-evk-ver2.dts.
- For RZV2N Evaluation Board Kit, the POEG has an input pin, so user can add as Figure 5.15.
- For RZV2N Evaluation Board Kit, for easier using, user can enable poeg by set I2C2_I3C_POEG_SEL to POEG_SEL in arch/arm64/boot/dts/renesas/r9a09g056n48-rzv2n-evk.dts.
- For RZG3S SMARC Evaluation Board Kit, the POEG has an input pin, so the user can add it as Figure 5.17.
- For RZG3S SMARC Evaluation Board Kit, for easier using, user can enable poeg by set POEGA_SEL to 1 in arch/arm64/boot/dts/renesas/r9a08g045s33-smarc.dts.

5.2.3 Kernel Configuration

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

```
Device Drivers --->
  [*] Pulse-Width Modulation (PWM) Support ----
    <*> GPT RZG2L
    <*> POEG RZG2L
```

Figure 5-19 Kernel configuration

5.3 Device Tree Setting

5.3.1 GPT node

Basic configuration information of GPT is defined at device tree file. It exists at arch/arm64/boot/dts/renesas directory. This module uses these described contents in device tree file.

- r9a07g044.dtsi for RZ/G2L.
- r9a07g054.dtsi for RZ/V2L.
- r9a09g047.dtsi for RZ/G3E.
- r9a09g057.dtsi for RZ/V2H
- r9a09g056.dtsi for RZ/V2N
- r9a08g045.dtsi for RZ/G3S.

The initial reference information is shown below.


```

gpt4: gpt@10048400 {
    compatible = "renesas,gpt-r9a07g044";
    reg = <0 0x10048400 0 0xa4>;
    #pwm-cells = <2>;
    interrupts = <GIC_SPI 270 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 271 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 272 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 273 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 274 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 275 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 276 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 277 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 278 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 279 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 280 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 281 IRQ_TYPE_EDGE_RISING>,
        <GIC_SPI 282 IRQ_TYPE_EDGE_RISING>;
    interrupt-names = "gtcia", "gtcib", "gtcic", "gtcid",
        "gtcie", "gtcif", "gtciada", "gtciadb",
        "gtciv", "gtciu", "gtcih", "gtcil",
        "gtdei";
    clocks = <&cpg CPG_MOD R9A07G044_GPT_PCLK>;
    power-domains = <&cpg>;
    resets = <&cpg R9A07G044_GPT_RST_C>;
    status = "disabled";
};

```

Figure 5-20 Device tree initial references information of GPT (RZ/G2L, RZ/V2L and RZ/G3S)

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

Required properties:

- **compatible:**
must be "renesas,gpt-r9a07g044" for R9A07G044 (RZ/G2L), "renesas,gpt-r9a07g054" for R9A07G054 (RZ/V2L) and "renesas,gpt-r9a08g045" 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 GPT
- **interrupt-names:**
mapped with interrupts and must be in the order with interrupts number.
- **clocks:** R9A07G044_GPT_PCLK is used for RZ/G2L, R9A07G054_GPT_PCLK is used for RZ/V2L and R9A08G045_GPT_PCLK is used for RZ/G3S
- **reset:** R9A07G044_GPT_RST_C is used for RZ/G2L, R9A07G054_GPT_RST_C is used for RZ/V2L and R9A08G045_GPT_RST_C is used for RZ/G3S phandle + clock/reset specifier pairs

```

gpt0_1: gpt@13010100 {
    compatible = "renesas,gpt-r9a09g047";
    reg = <0 0x13010100 0 0xa4>;
    #pwm-cells = <2>;
    interrupts = <GIC_SPI 539 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 547 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 555 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 563 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 571 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 579 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 587 IRQ_TYPE_EDGE_RISING>,
                <GIC_SPI 595 IRQ_TYPE_EDGE_RISING>;
    interrupt-names =
        "gtcia", "gtcib", "gtcic", "gtcid",
        "gtcie", "gtcif", "gtcih", "gtcil";
    clocks = <&cpg CPG_MOD 49>;
    power-domains = <&cpg>;
    resets = <&cpg 89>,
            <&cpg 90>;
    status = "disabled";
};

```

Figure 5-21 Device tree initial references information of GPT (RZ/V2H, RZ/V2N and RZ/G3E)

Note: All of the information in above device tree is used for RZ/G3E.

Required properties:

- compatible:
must be "renesas,gpt-r9a09g047" for R9A09G047 (RZ/G3E and RZ/V2H) and "renesas, gpt-r9a09g056" for R9A09G056 (RZ/V2N).
- 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 GPT
- interrupt-names:
mapped with interrupts and must be in the order with interrupts number.

- clocks: GPT_PCLK_SFR index 49 is used for RZ/G3E (GPT0_1), 0x31 for RZ/V2H and RZ/V2N (GPT0_2).
- reset: GPT0_RST_P_REG index 89 is used for RZ/G3E (GPT0_1), 0x59 for RZ/V2H and RZ/V2N (GPT0_2).
reset: GPT0_RST_S_REG index 90 is used for RZ/G3E (GPT0_1), 0x5A for RZ/V2H and RZ/V2N (GPT0_2).
phandle + clock/reset specifier pairs

5.3.2 POEG node

Basic configuration information of GPT is defined at device tree file (r9a07g044.dtsi for RZ/G2L, r9a07054.dtsi for RZ/V2L, r9a09g056.dtsi for RZ/V2N, 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 device tree file.

The initial reference information is shown below.

```
poegd: poegd@10049400 {
    compatible = "renesas,poeg-r9a07g044";
    reg = <0 0x10049400 0 0x04>;
    interrupts = <GIC_SPI 325 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD R9A07G044_POEG_D_CLKP>;
    power-domains = <&cpg>;
    resets = <&cpg R9A07G044_POEG_D_RST>;
    status = "disabled";
};
```

Figure 5-22 Device tree initial references information of POEG (RZ/G2L)

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

Required properties:

- compatible:
must be "renesas,poeg-r9a07g044" for R9A07G044 (RZ/G2L), "renesas,poeg-r9a07g054" for R9A07G054 (RZ/V2L) and "renesas,poeg-r9a08g045" 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 GPT
- interrupt-names:
mapped with interrupts and must be in the order with interrupts number.
- clocks: R9A07G044_POEG_D_CLKP is used for RZ/G2L, R9A07G054_POEG_D_CLKP is used for RZ/V2L and R9A08G045_POEG_A_CLKP is used for POEGA of RZ/G3S; other channels must use corresponding clock index.
- reset: R9A07G044_POEG_D_RST is used for RZ/G2L, R9A07G054_POEG_D_RST is used for RZ/V2L and R9A08G045_POEG_A_RST is used for POEGA of RZ/G3S, other channels must use corresponding reset index
phandle + clock/reset specifier pairs

```

poega: poega@13001C00 {
    compatible = "renesas,poeg-r9a09g047";
    reg = <0 0x13001C00 0 0x04>;
    interrupts = <GIC_SPI 666 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD 51>;
    power-domains = <&cpg>;
    resets = <&cpg 93>;
    status = "disabled";
};

```

Figure 5-23 Device tree initial references information of POEG (RZ/V2H, RZ/V2N and RZ/G3E)

Note: All of the information in above device tree is used for RZ/G3E, RZ/V2H and RZ/V2N except compatible, clocks and resets..

Required properties:

- **compatible:**
must be "renesas,poeg-r9a09g047" for R9A09G047 (RZ/G3E), "renesas,poeg-r9a09g057" for R9A09G057 (RZ/V2H) and "renesas,poeg-r9a09g056" for R9A09G056 (RZ/V2N).
- **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 GPT
- **interrupt-names:**
mapped with interrupts and must be in the order with interrupts number.
- **clocks:** R9A07G044_POEG_A_CLKP index 51 is used for RZ/G3E, 0x33 for RZ/V2H and RZ/V2N.
- **reset:** R9A07G044_POEG_A_RST index 93 is used for RZ/G3E, 0x5D for RZ/V2H and RZ/V2N.
phandle + clock/reset specifier pairs

5.3.3 Documentation file in the kernel

Please follow guideline in Documentation/devicetree/bindings/pwm/pwm.txt

Please follow guideline in Documentation/devicetree/bindings/pwm/renesas,poeg-rzg2l.txt

5.4 Option Setting

5.4.1 Module Parameters

There are no module parameters.

5.4.2 Kernel Parameters

There are no parameters.

REVISION HISTORY	Linux Interface Specification Device Driver GPT RZ/G2L Group 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 RZ/G2LC device
1.3	Mar. 31, 2022	—	Update information for device tree.
1.4	May. 31, 2022	—	No modification, keep version to keep consistent with other documents
1.5	Jun. 24, 2022	—	No modification, keep version to keep consistent with other documents
1.6	Sep. 15, 2022	—	Add AB phase counting, choosing GPT operation to driver, update device tree
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	May. 30, 2025	1	- Add MPU information support for both kernel versions v5.10 and v6.1.
1.10	Jul. 22, 2025	--	- Add RZ/G3E support information
1.11	Nov. 28, 2025	1	- Add information of RZ/G2UL and RZ/V2L support for kernel v6.1
		10	- Update information for normal output mode
		15	- Update define marco for GPT channel select
		--	- Add RZ/G3S support information
1.12	Dec. 19, 2025	--	- Add RZ/V2H support information
1.13	Mar. 27, 2026	--	- Add RZ/V2N support information
		5	- Update product number for RZ/V2N EVK boards

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

Publication Date: Rev. 1.13 Mar. 27, 2026

Published by: Renesas Electronics Corporation

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



Renesas Electronics Corporation