

Linux Interface Specification LCDC

User's Manual: Software

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

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

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

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

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

(Rev. 1.12 Jul. 22, 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/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/V2N Group User's Manual: Hardware	---
		RZ/G3E Group User's Manual: Hardware	---
User's manual for Software	Description of LCDC Linux interface Specification	Linux interface Specification Device Driver LCDC	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
DU	Display Unit on RZ/G Series, 2nd Generation
VSPD	Video signal processing for Display
DSI	Display Serial Interface
FBDev	Framebuffer Device
DRM	Direct Rendering Manager
KMS	Kernel Mode Setting
DRI	Direct Rendering Infrastructure
FB	Framebuffer
LIF	LCDC Interface (VSP-DU connect mode)
RPF	Read Pixel Formatter
WPF	Write Pixel Formatter
BRS	Blend ROP Sub-Unit
CRTC	Cathode Ray Tube Controller
CVT	Coordinated Video Timings
GTF	General Timing Formula

Table of Contents

1. Overview	1
1.1 Overview	1
1.2 Function	1
1.2.1 Display Resolution	2
1.2.2 Pixel format	3
1.2.3 Connector	3
1.3 Related Document	3
1.4 Restrictions	3
1.5 Notice	3
2. Terminology	4
3. Operating Environment	5
3.1 Hardware Environment	5
3.2 Module configuration	6
3.3 State Transition Diagram	9
4. External Interface	10
4.1 External Interface for DRM/KMS Driver	10
4.2 DRM resource information	10
4.3 Hot plug Operation	11
4.4 Multiple Display Support	12
5. Integration	13
5.1 Directory Configuration	13
5.2 Integration Procedure	14
5.2.1 Kernel Configuration	14
5.2.2 Devicetree definition	15
5.2.3 Kernel parameters	26

1. Overview

1.1 Overview

This manual explains the LCD controller that controls the Frame Compression Processor (FCPVD), Video Signal Processor (VSPD), and Display Unit (DU) in Linux Solution for RZ/G2L Group, RZ/V2L Group, RZ/V2N Group, and RZ/G3E Group.

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

- v5.10: RZ/G2L Group and RZ/V2L Group.
- v6.1: RZ/G2L, RZ/G2LC, RZ/V2N and RZ/G3E.

1.2 Function

This module controls VSPD to be equipped with RZ/G2L Group, RZ/V2L Group, RZ/V2N and RZ/G3E Group and blends an image.

A blended image is output through Display module. A function list supported to a Display driver is as follows:

- Support Display Parallel Interface (DPI), MIPI DSI Link Video interface and Low-Voltage Differential Signaling (LVDS) (*)
- Display timing master:
 - Generate video timings (Front porch, Back porch, Sync active, Active video area).
 - Selecting the polarity of output DCLK, HSYNC, VSYNC and DE.
- Input pixel format (from VSPD): RGB888, RGB666, RGB565, YCbCr444.
- Output pixel format: same as Input data format.
- Support Progressive (non-interlace) and not support Interlace.
- Support Full HD (1920 pixels x 1080 lines).
- Support up to 2 display channels for RZ/G3E.
- Hot plug detection.

(*) MIPI DSI Link Video interface is not supported in RZ/G2UL.

Display Parallel Interface is not supported in RZ/V2N.

Low-Voltage Differential Signaling is only supported in RZ/G3E.

1.2.1 Display Resolution

The supported display resolution of this module is as follows.

Table 1.1 Supported resolution for each board

Display resolution	RZ/G2L RZ/G2LC RZ/V2L (HDMI)	RZ/G2UL (HDMI)	RZ/V2N (HDMI)	RZ/G3E		
				MIPI-DSI (HDMI)	Parallel (HDMI)	LVDS (HDMI)
VGA (640x480)	yes	yes	yes	yes	yes	yes
SVGA (800x600)	yes	yes	yes	yes	yes	yes
WSVGA (1024x600)	yes	yes	yes	yes	yes	yes
XGA (1024x768)	yes	yes	yes	yes	yes	yes
720p (1280x720)	yes	yes	yes	yes	yes	yes
SXGA (1280x1024)	yes	no	yes	yes	no	yes
1080@30p (1920x1080)	yes	yes	yes	yes	yes	yes
1080@60p (1920x1080)	yes	no	yes	yes	no	yes

Notes: 1. The initial value is set as the recommendation resolution which a display monitors requires (The start-up in HDMI cable connection). the resolution of HDMI cannot be displayed about resolution that a display monitor does not support. The resolution of HDMI can set the supported resolution to the display monitor.

1.2.2 Pixel format

Supported pixel format of this module is as follows.

Table 1.2 Supported pixel format

Pixel format	DRM FourCC macro name	Support
XRGB8888	DRM_FORMAT_XRGB8888	yes
ARGB8888	DRM_FORMAT_ARGB8888	yes
RGB888	DRM_FORMAT_RGB888	yes

1.2.3 Connector

Supported connector of this module is as follows.

Table 1.3 Supported connector for each board

Output signal	Connector Name	Evaluation Board Kit
HDMI (MIPI DSI IF)	CN13 (Micro HDMI port)	RZ/G2L, RZ/G2LC and RZ/V2L
HDMI (MIPI DSI IF)	CN4 (HDMI port)	RZ/V2N
HDMI (Digital Parallel IF)	CN5	RZ/G2UL
HDMI (MIPI DSI IF)	Micro HDMI port on carrier	RZ/G3E
HDMI (Digital Parallel IF)	CN2	
HDMI (LVDS)	Dual LVDS-HDMI	

1.3 Related Document

Table 1.4 Related documents

Number	Issue	Title	Edition	Data
-	-	-	-	-

1.4 Restrictions

There is no restriction.

1.5 Notice

- FBDev access is not supported. However, it supports only to draw the image to the framebuffer with FBDev. Other FBDev access control does not support.

2. Terminology

The following table shows the terminology related to this module.

Table 2-1 Terminology

Terms	Explanation
DU	Display Unit on RZ/G Series, 2 nd Generation
VSPD	Video signal processing for Display
DSI	Display Serial Interface
FBDev	Framebuffer Device
DRM	Direct Rendering Manager
KMS	Kernel Mode Setting
DRI	Direct Rendering Infrastructure
FB	Framebuffer
LIF	LCDC Interface (VSP-DU connect mode)
RPF	Read Pixel Formatter
WPF	Write Pixel Formatter
BRS	Blend ROP Sub-Unit
CRTC	Cathode Ray Tube Controller
CVT	Coordinated Video Timings
GTF	General Timing Formula

3. Operating Environment

3.1 Hardware Environment

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

Table 3-1 Hardware specification (RZ/G2L and RZ/V2L)

Name	Version	Manufacturer
RZ/G2L Evaluation Board Kit	SMARC module: v01 Carrier board: v03	Renesas Electronics Europe GmbH
RZ/G2LC Evaluation Board Kit	SMARC module: v01 Carrier board: v03	Renesas Electronics Europe GmbH
RZ/G2UL Evaluation Board Kit	SMARC module: v01 Carrier board: v03	Renesas Electronics Europe GmbH
RZ/V2L Evaluation Board Kits	SMARC module: v01 Carrier board: v03	Renesas Electronics Europe GmbH

Table 3-2 Hardware specification (RZ/V2N)

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

3.2 Module configuration

The following figure shows the configuration of this module.

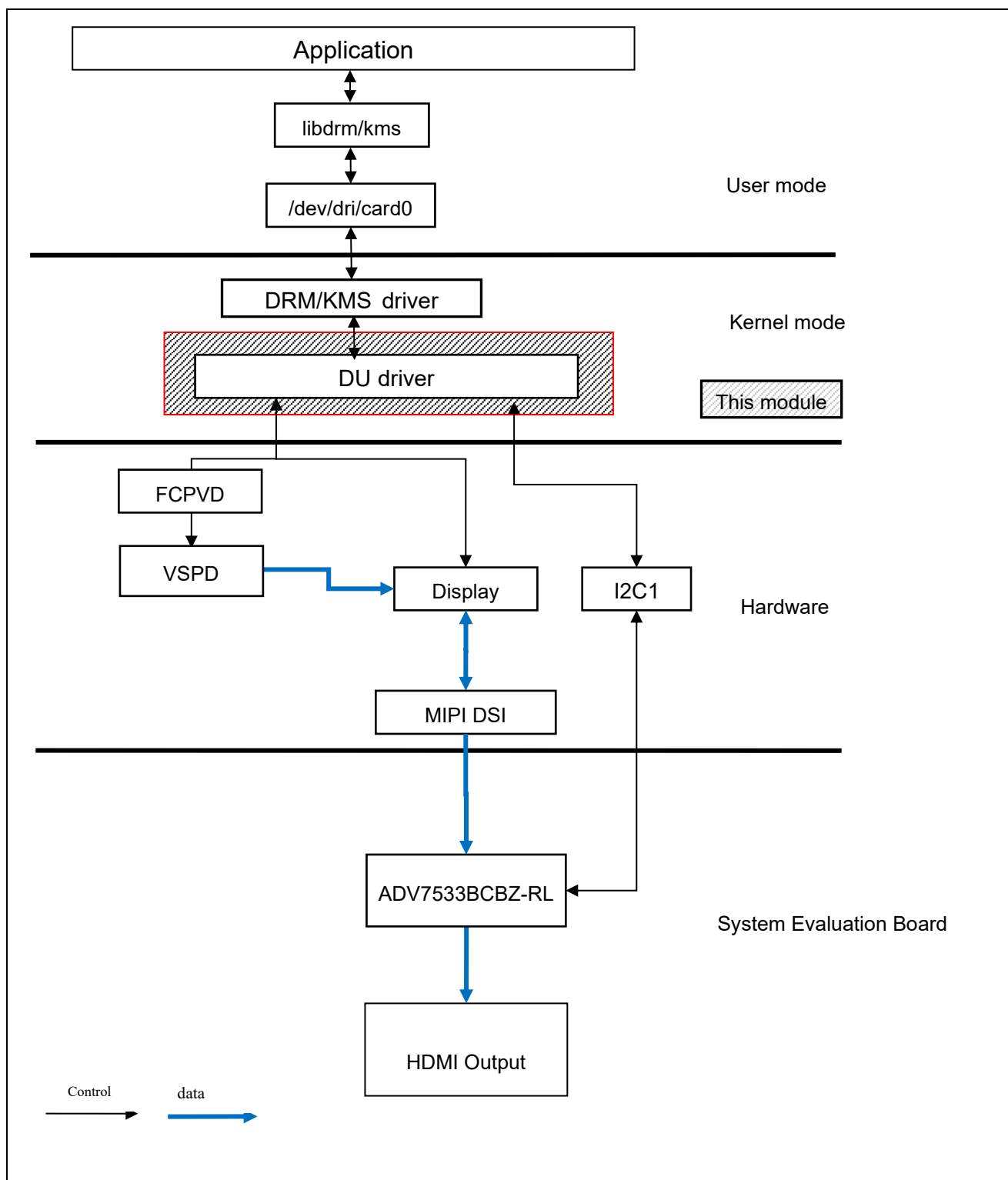
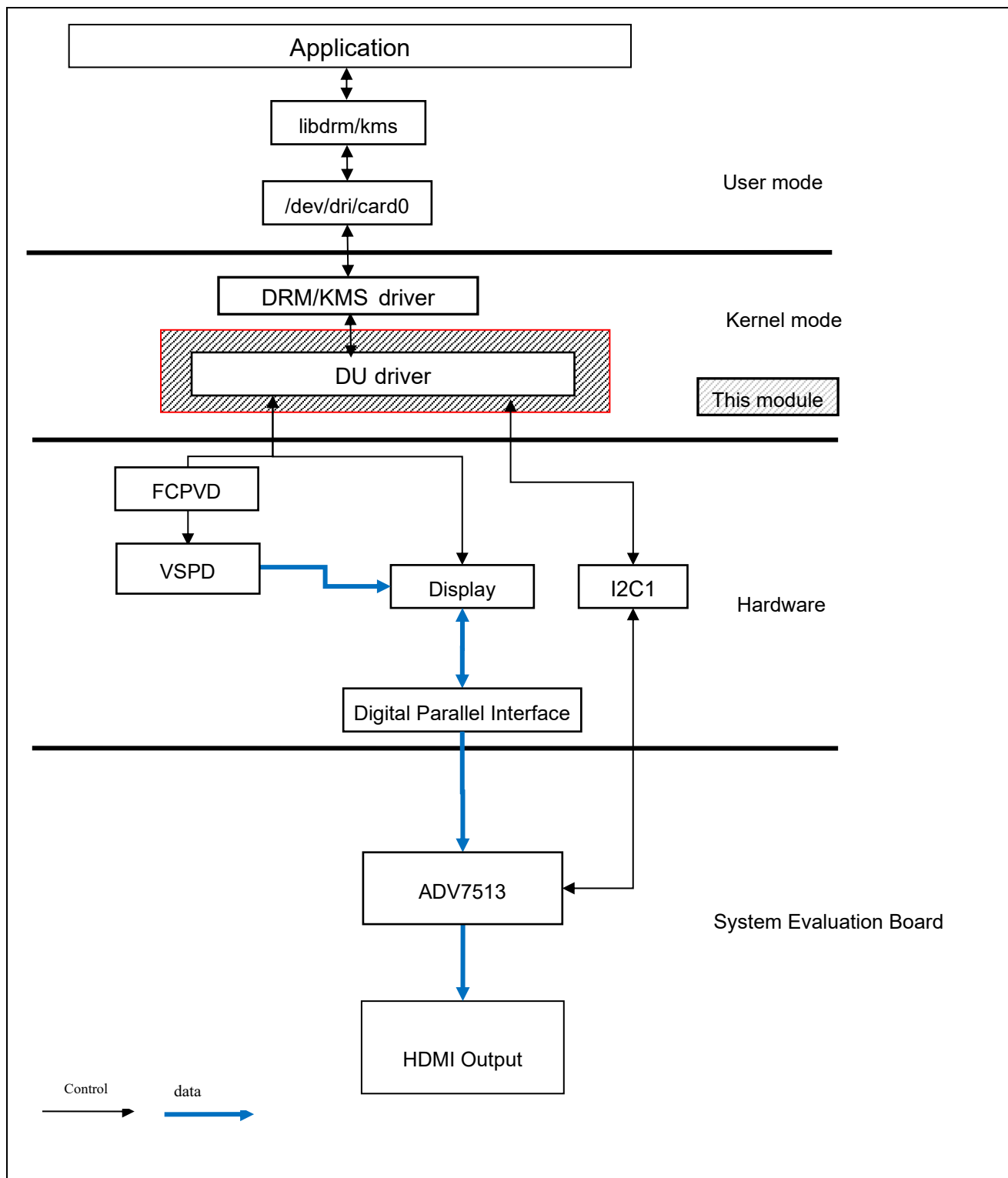


Figure 3-1 Module configuration (RZ/G2L, RZ/G2LC and RZ/V2L)

**Figure 3-2 Module configuration (RZ/G2UL)**

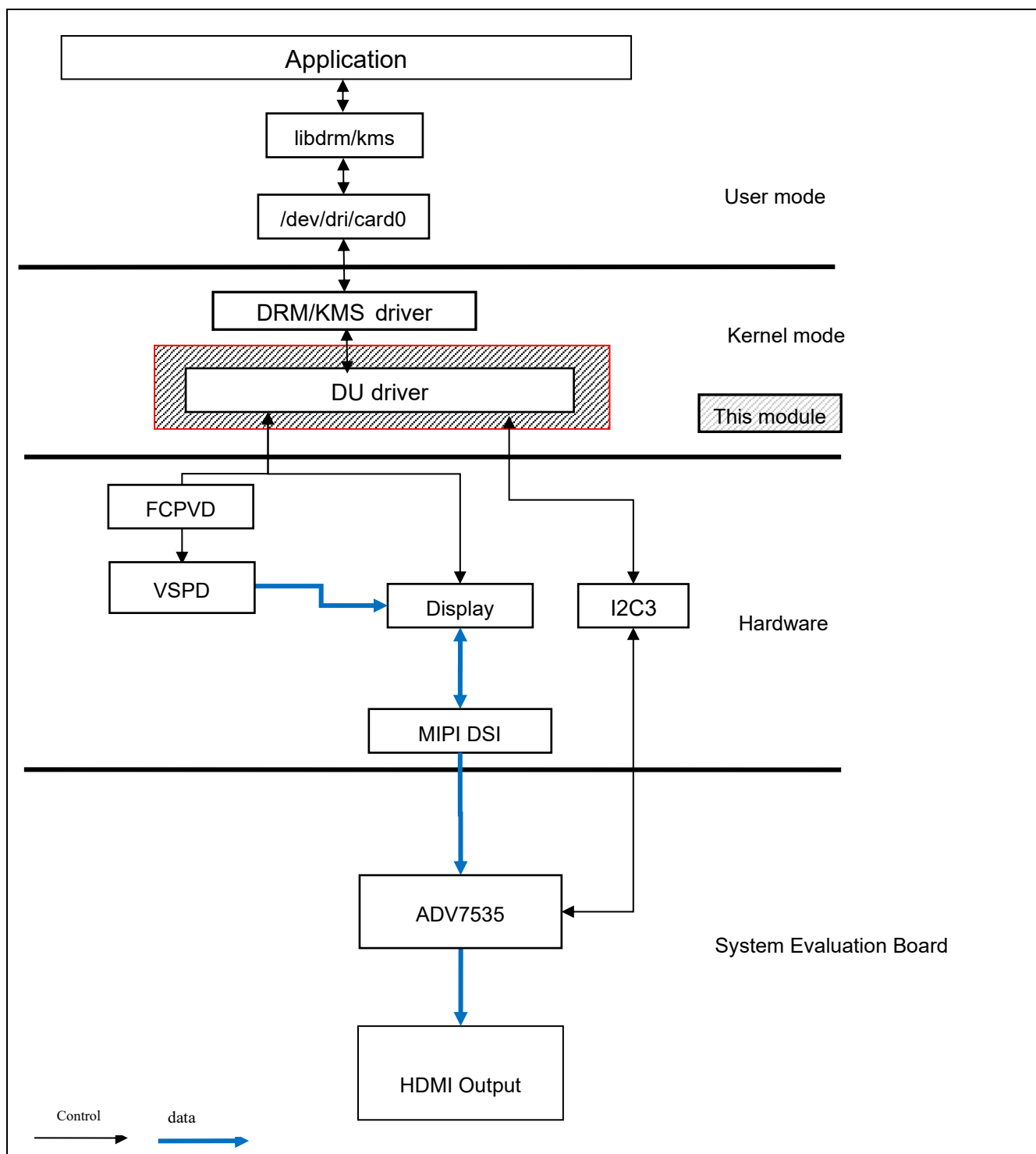


Figure 3-3 Module configuration (RZ/V2N)

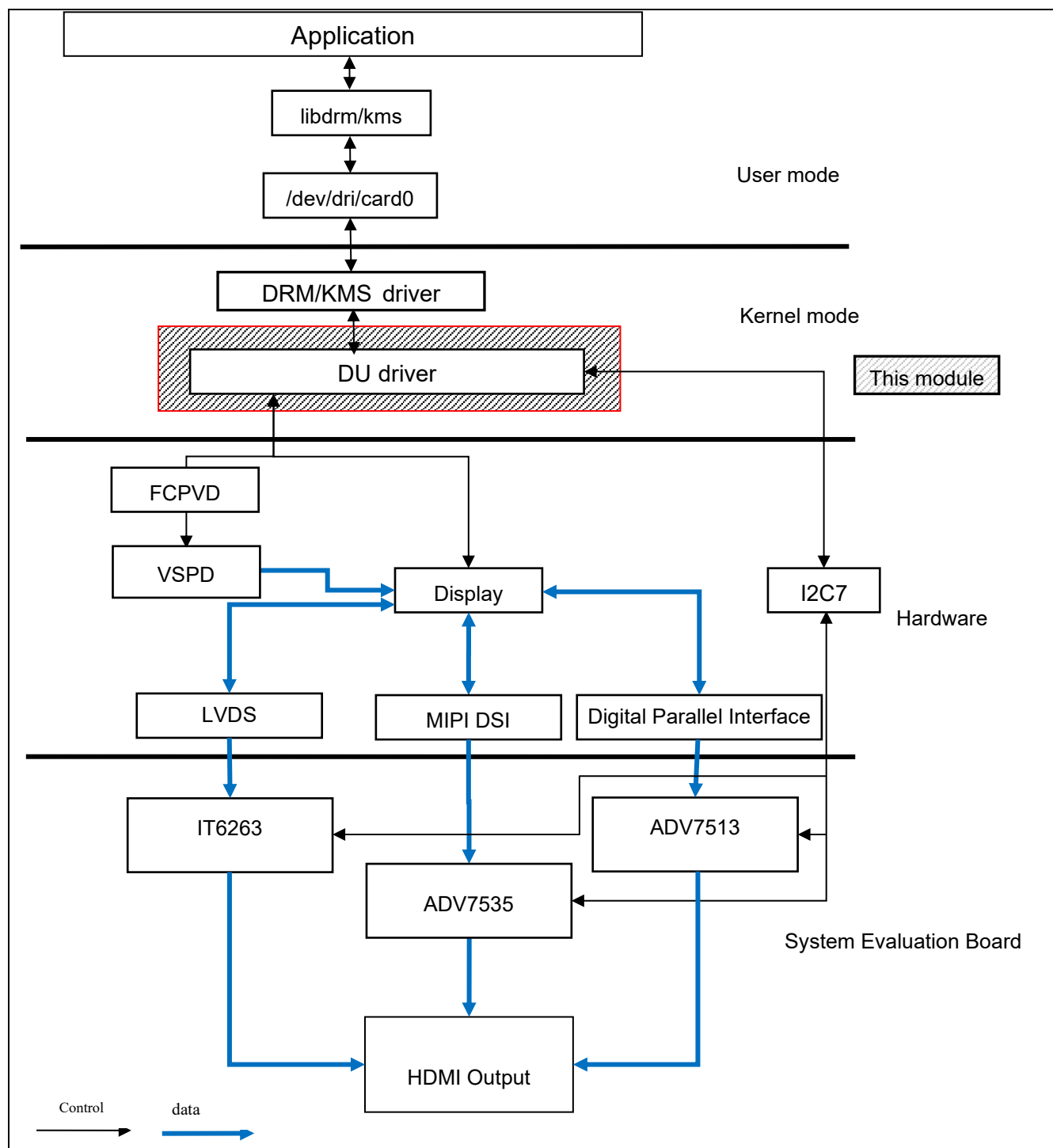


Figure 3-4 Module configuration (RZ/G3E)

3.3 State Transition Diagram

There is no state transition diagram for this module.

4. External Interface

The external interface of this module is based on Linux.

Device node of this module is shown below.

Table 4-1 DRM device node (RZ/G2L, RZ/V2L, RZ/V2N and RZ/G3E)

Device node	Major number	Minor number
/dev/dri/card0	226	0

4.1 External Interface for DRM/KMS Driver

This driver corresponds to the v2.4.101 of libdrm/libkms library.

For details, please refer to the following:

- libdrm library (libdrm/libkms library download site.)
<http://cgit.freedesktop.org/mesa/drm/>
 "tests/modetest/modetest.c" attached to libdrm/libkms library is a sample test program which becomes reference of how to call libdrm/libkms interface.
- DRI Wiki (Information of DRI. Documentation and build information of libdrm/libkms library.)
<http://dri.freedesktop.org/wiki/>
- Linux GPU Driver Developer's Guide
 Kernel v5.10: <https://www.kernel.org/doc/html/v5.10/gpu/index.html>
 Kernel v6.1: <https://www.kernel.org/doc/html/v6.1/gpu/index.html>

4.2 DRM resource information

The DRM resource information on Connectors ID and CRTC's ID at the time of a default configuration is indicated. If you want to change the configuration, there are cases when the value of the ID does not match the following.

Table 4-2 List of DRM resources ID

Platform		Connectors ID	CRTC's ID
RZ/G2L and RZ/V2L		HDMI: 37	DU: 35
RZ/V2N		HDMI: 37	DU: 35
RZ/G3E	MIPI-DSI	HDMI: 54	DU: 52
	Parallel/LVDS	HDMI: 56	DU: 51

4.3 Hot plug Operation

This driver supports hot plug operation.

The state of a HDMI cable can be checked. A sample for getting state of connected or disconnected via DRM is follows.

```
{
    int i, fd;
    drmModeRes *resources;
    drmModeConnector *connector;

    fd = drmOpen("rzg2l-du", NULL);
    resources = drmModeGetResources(fd);

    for (i = 0; i < resources->count_connectors; i++) {
        connector = drmModeGetConnector(fd, resources->connectors[i]);
        if (connector->connector_type == DRM_MODE_CONNECTOR_HDMIA) {
            if (connector->connection == DRM_MODE_CONNECTED)
                printf("connected\n");
            else if (connector->connection == DRM_MODE_DISCONNECTED)
                printf("disconnected\n");
            else if (connector->connection == DRM_MODE_UNKNOWNCONNECTION)
                printf("unknown\n");
        }
    }
    return 0;
}
```

Figure 4.1 Acquisition method of getting HDMI cable status via DRM

Moreover, HDMI connection state can also be checked via sysfs filesystem as below:

```
HDMI :

# cat /sys/class/drm/card0-HDMI-A-1/status
```

The following information can get.

```
connected

or

disconnected
```

4.4 Multiple Display Support

This function is only available on RZ/G3E Group.

RZ/G3E supports 3 types of display interfaces, MIPI-DSI, Digital Parallel Interface and Low-Voltage Differential Signaling (LVDS). But due to hardware, only 2 types of display interfaces can be enabled at the same time.

Default, RZ/G3E is enabled MIPI-DSI and LVDS.

To switch/enable/disable these, please modify in device tree **r9a09g047e57-smarc.dts** based on Table 4-3

Table 4-3 Macro enable multiple display

Interface	Macro	Switch on Board	MIPI-DSI and LVDS		MIPI-DSI and Parallel		Parallel and LVDS	
			Macro	Switch	Macro	Switch	Macro	Switch
MIPI-DSI	RZG3E_SMARC_MIPI_DSI_ENABLE	--	1	--	1	--	0	--
	SW_HDMI_EN	SW4_2	0	OFF	0	OFF	0	OFF
Parallel	SW_LCD_EN	SW4_5	0	OFF	1	ON	1	ON
LVDS	RZG3E_SMARC_LVDS_ENABLE	--	1	--	0	--	1	--

5. Integration

5.1 Directory Configuration

The directory configuration is shown below.

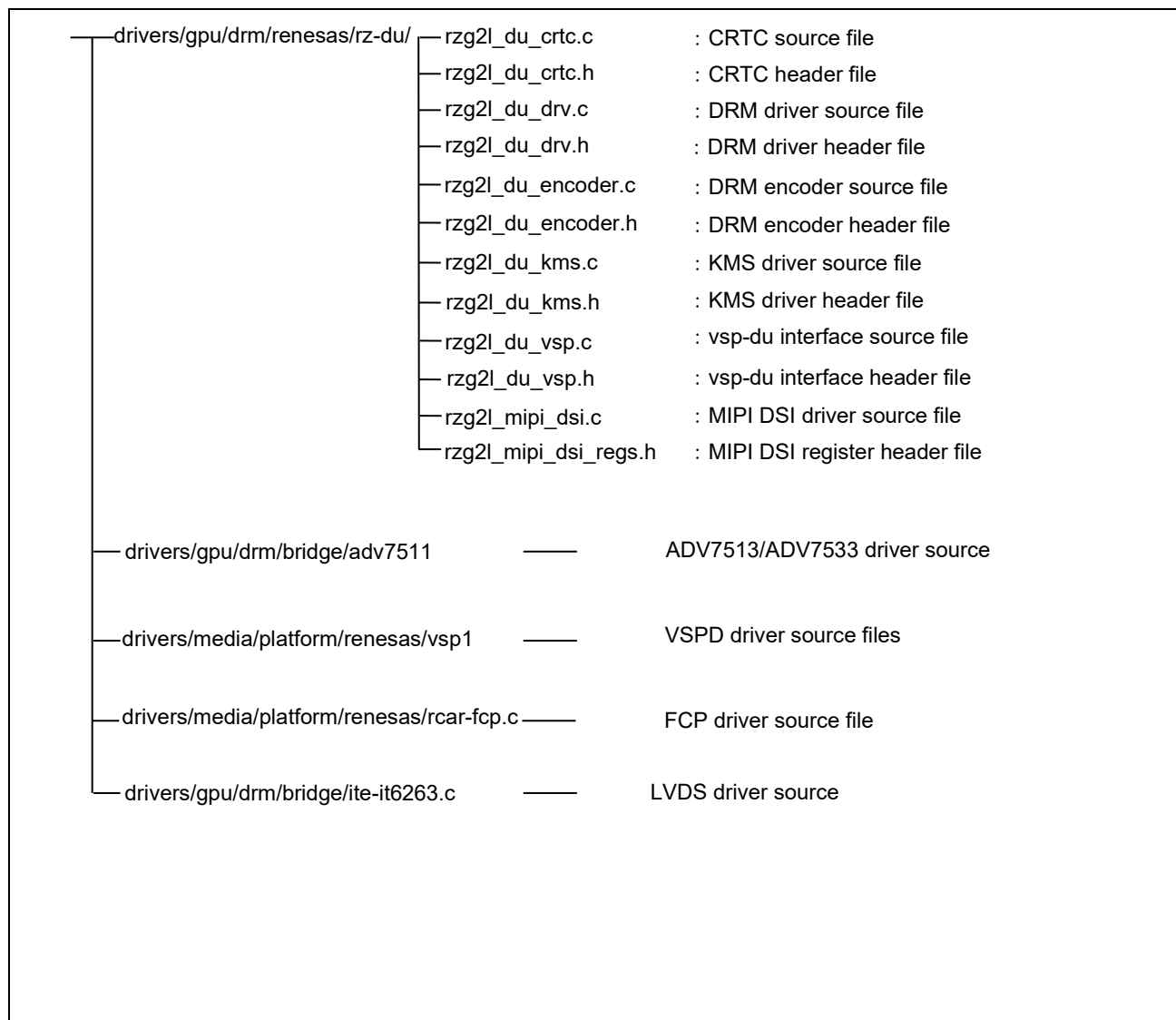


Figure 5-1 Directory configuration

5.2 Integration Procedure

5.2.1 Kernel Configuration

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

```
Device Drivers --->
  I2C support --->
    I2C Hardware Bus support --->
      <*> Renesas R-Car I2C Controller
  <*> Multimedia support --->
    [*] Media Controller API
    [*] Memory-to-memory multimedia devices --->
      <*> Renesas Frame Compression Processor
      <*> Renesas VSP1 Video Processing Engine
  Graphics support --->
    <*> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      [*] Enable legacy fbdev support for your modesetting driver (*1)
    <*> DRM Support for RZ/G2L Display Unit
    <*> RZ/G2L and RZ/V2L MIPI DSI Encoder Support
    <*> ITE IT6263 LVDS/HDMI bridge
    [*] RZ/G3E DU LVDS Encoder Support
  Display Interface Bridges --->
    <*> ADV7511 encoder (*2)
```

Figure 5-2 Kernel configuration

***1**

Please remove the check if you do not support the legacy FBDev support.
Note that this support also provides the linux console support on top of your modesetting driver.
The default setting is ON. The console image is drawn to DU.

***2**

The ADV7513/ADV7533/ADV7535 and IT6263(only for RZ/G3E) HDMI transmitters are used to output HDMI in RZ/G2{L, LC, UL}, RZ/V2L, RZ/V2N Evaluation Board Kit and RZ/G3E Smarc Evaluation Board Kit.

5.2.2 Devicetree definition

Below figure lists the necessary properties to support LCDC in RZ/G2L, RZ/V2L, RZ/V2N Evaluation Board Kit and RZ/G3E Smarc Evaluation Board Kit.

Example:

- **r9a07g044.dtsi:**

```
vspd: vsp@10870000 {
    compatible = "renesas,r9a07g044-vsp2";
    reg = <0 0x10870000 0 0x10000>;
    interrupts = <GIC_SPI 149 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD R9A07G044_LCDC_CLK_A>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_P>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_D>;
    clock-names = "aclk", "pclk", "vclk";
    power-domains = <&cpg>;
    resets = <&cpg R9A07G044_LCDC_RESET_N>;
    renesas,fcv = <&fcv>; // phandle to fcv node
};

fcv: fcv@10880000 {
    compatible = "renesas,r9a07g044-fcv",
               "renesas,fcv";
    reg = <0 0x10880000 0 0x10000>;
    clocks = <&cpg CPG_MOD R9A07G044_LCDC_CLK_A>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_P>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_D>;
    clock-names = "aclk", "pclk", "vclk";
    power-domains = <&cpg>;
    resets = <&cpg R9A07G044_LCDC_RESET_N>;
};
```

Figure 5-3 Device nodes for FCP and VSPD

```
du: display@10890000 {
    compatible = "renesas,r9a07g044-du";
    reg = <0 0x10890000 0 0x10000>;
    interrupts = <GIC_SPI 152 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD R9A07G044_LCDC_CLK_A>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_P>,
            <&cpg CPG_MOD R9A07G044_LCDC_CLK_D>;
    clock-names = "aclk", "pclk", "vclk";
    resets = <&cpg R9A07G044_LCDC_RESET_N>;
    vsps = <&vspd 0>; // phandle to vspd node
    power-domains = <&cpg>;
    status = "disabled";
    ports {
        // output ports endpoint (0: parallel; 1: MIPI DSI)
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            du_out_rgb: endpoint {
            };
        };
        port@1 {
            reg = <1>;
            du_out_dsi: endpoint {
                remote-endpoint = <&dsi0_in>;
            };
        };
    };
};
```

Figure 5-4 Device node for Display Unit (DU)

```

dsi0: dsi@10850000 {
    compatible = "renesas,r9a07g044-mipi-dsi",
        "renesas,rzg2l-mipi-dsi";
    reg = <0 0x10850000 0 0x10000>; /* DPHY */
    interrupts = <GIC_SPI 142 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 143 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 144 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 145 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 146 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 147 IRQ_TYPE_LEVEL_HIGH>,
        <GIC_SPI 148 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "seq0", "seq1", "vin1", "rcv",
        "ferr", "ppi", "debug";
    clocks = <&cpg CPG_MOD R9A07G044_MIPI_DSI_PLLCLK>,
        <&cpg CPG_MOD R9A07G044_MIPI_DSI_SYSCLK>,
        <&cpg CPG_MOD R9A07G044_MIPI_DSI_ACLK>,
        <&cpg CPG_MOD R9A07G044_MIPI_DSI_PCLK>,
        <&cpg CPG_MOD R9A07G044_MIPI_DSI_VCLK>,
        <&cpg CPG_MOD R9A07G044_MIPI_DSI_LPCLK>;
    clock-names = "pllclk", "sysclk", "aclk", "pclk", "vclk", "lpclk";
    resets = <&cpg R9A07G044_MIPI_DSI_CMN_RSTB>,
        <&cpg R9A07G044_MIPI_DSI_ARESET_N>,
        <&cpg R9A07G044_MIPI_DSI_PRESET_N>;
    reset-names = "rst", "arst", "prst";
    power-domains = <&cpg>;
    status = "disabled";

    ports {
        // input/output ports endpoint (0: input (point to DU DSI
output port); 1: output
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            dsi0_in: endpoint {
                remote-endpoint = <&du_out_dsi>;
            };
        };
        port@1 {
            reg = <1>;
        };
    };
};

```

Figure 5-5 Device node for MIPI DSI

Note: All the information in the above device tree is used for RZ/G2L, also the same as RZ/V2L. To config for RZ/V2L, only replace r9a07g044 with r9a07g054.

For RZ/V2N and RZ/G3E, please use the below device tree information in **r9a09g056.dtsi** and **r9a09g047.dtsi**.

```

fcvpvd: fcp@16470000 {
    compatible = "renesas,r9a09g056-fcpvd",
        "renesas,fcpv";
    reg = <0 0x10880000 0 0x10000>;
    clocks = <&cpg CPG_MOD 0xed>,
        <&cpg CPG_MOD 0xee>,
        <&cpg CPG_MOD 0xef>;
    clock-names = "aclk", "pclk", "vclk";
    resets = <&cpg 0xdc>;
    power-domains = <&cpg>;
};

vspd: vsp@16480000 {
    compatible = "renesas,r9a07g044-vsp2";
    reg = <0 0x16480000 0 0x10000>;
    interrupts = <GIC_SPI 881 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD 0xed>,
        <&cpg CPG_MOD 0xee>,

```

```

        <&cpg CPG_MOD 0xef>;
        clock-names = "aclk", "pclk", "vclk";
        resets = <&cpg 0xdc>;
        power-domains = <&cpg>;
        renesas,fcv = <&fcvvd>;
};

```

Figure 5-6 Device nodes for FCP and VSPD

```

du: display@16460000 {
    compatible = "renesas,r9a09g056-du";
    reg = <0 0x16460000 0 0x10000>;
    interrupts = <GIC_SPI 882 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD 0xed>,
            <&cpg CPG_MOD 0xee>,
            <&cpg CPG_MOD 0xef>;
    clock-names = "aclk", "pclk", "vclk";
    power-domains = <&cpg>;
    resets = <&cpg 0xdc>;
    renesas,vsp = <&vspd 0>;
    status = "disabled";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            reg = <0>;
            du_out_dsi0: endpoint {
                remote-endpoint = <&dsi0_in>;
            };
        };
    };
};

```

Figure 5-7 Device node for Display Unit (DU)

```

dsi0: dsi@16430000 {
    compatible = "renesas,r9a09g056-mipi-dsi",
               "renesas,rzv2n-mipi-dsi";
    reg = <0 0x16430000 0 0x20000>;
    clocks = <&cpg CPG_MOD 0xe8>,
            <&cpg CPG_MOD 0xe9>,
            <&cpg CPG_MOD 0xea>,
            <&cpg CPG_MOD 0xeb>,
            <&cpg CPG_MOD 0xec>;
    clock-names = "pclk", "aclk",
                  "vclk", "lpclk", "pllclk";
    resets = <&cpg 0xd7>,
            <&cpg 0xd8>;
    reset-names = "prst", "arst";
    power-domains = <&cpg>;
    status = "disabled";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            reg = <0>;
            dsi0_in: endpoint {
                remote-endpoint = <&du_out_dsi0>;
            };
        };

        port@1 {
            reg = <1>;

```



```

                                dsi0_out: endpoint {
                                    };
                                };
};

```

Figure 5-8 Device node for MIPI DSI

For RZ/G3E, please use the below device tree information in **r9a09g047.dtsi**

```

du: display@16460000 {
    compatible = "renesas,r9a09g047-du";
    reg = <0 0x16460000 0 0x10000>,
        <0 0x16490000 0 0x10000>;
    reg-names = "du0", "du1";
    interrupts = <GIC_SPI 882 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&cpg CPG_MOD 237>,
        <&cpg CPG_MOD 238>,
        <&cpg CPG_MOD 239>,
        <&cpg CPG_MOD 424>,
        <&cpg CPG_MOD 425>,
        <&cpg CPG_MOD 426>;
    clock-names = "aclk0", "pclk0", "vclk0",
        "aclk1", "pclk1", "vclk1";
    power-domains = <&cpg>;
    resets = <&cpg 220>,
        <&cpg 286>;
    renesas,vsp = <&vspd0 0>, <&vspd1 0>;
    status = "disabled";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            reg = <0>;
            du_out_dsi0: endpoint {
                remote-endpoint = <&dsi0_in>;
            };
        };
        port@1 {
            reg = <1>;
            du_out_rgb: endpoint {
            };
        };
        port@2 {
            reg = <2>;
            du_out_lvds0: endpoint {
                remote-endpoint = <&lvds0_in>;
            };
        };
        port@3 {
            reg = <3>;
            du_out_lvds1: endpoint {
                remote-endpoint = <&lvds1_in>;
            };
        };
    };
};

lvds: lvds@164c0000 {
    compatible = "renesas,rzg3e-lvds";
    reg = <0x0 0x164c0000 0x0 0x10>;
    clocks = <&cpg CPG_MOD 420>;
    resets = <&cpg 282>;
    power-domains = <&cpg>;
    status = "disabled";
    ranges;
}

```

```

#address-cells = <2>;
#size-cells = <2>;

lvds0: lvds0-encoder@164c0010 {
    compatible = "renesas,rzg3e-lvds-link";
    reg = <0x0 0x164c0010 0x0 0x8>;
    clocks = <&cpg CPG_MOD 416>,
            <&cpg CPG_MOD 418>;
    clock-names = "phyclk", "dotclk";
    power-domains = <&cpg>;
    renesas,id = <0>;
    renesas,companion = <&lvds1>;
    status = "disabled";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            lvds0_in: endpoint {
                remote-endpoint = <&du_out_lvds0>;
            };
        };
        port@1 {
            reg = <1>;
        };
    };
};

lvds1: lvds1-encoder@164c0018 {
    compatible = "renesas,rzg3e-lvds-link";
    reg = <0x0 0x164c0018 0x0 0x8>;
    clocks = <&cpg CPG_MOD 417>,
            <&cpg CPG_MOD 419>;
    clock-names = "phyclk", "dotclk";
    power-domains = <&cpg>;
    renesas,id = <1>;
    status = "disabled";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            lvds1_in: endpoint {
                remote-endpoint = <&du_out_lvds1>;
            };
        };
        port@1 {
            reg = <1>;
        };
    };
};
};

```

Figure 5-9 Device nodes for Display Unit (MIPI-DSI, Parallel and LVDS) of RZ/G3E

- **rzg2l-smarc.dtsi:**

```

        osc1: cec-clock {
            compatible = "fixed-clock";
            #clock-cells = <0>;
            clock-frequency = <12000000>;
        };
        hdmi-out {
            compatible = "hdmi-connector";    // Compatible string for kind of
connector
            type = "d";
            port {
                hdmi_con_out: endpoint {
                    remote-endpoint = <&adv7535_out>;    // Output port
from HDMI transmitter
                };
            };
        };

```

Figure 5-10 Device node for HDMI connector port

```

        &pinctrl {
            i2c1_pins: i2c1 {
                pins = "RIIC1_SDA", "RIIC1_SCL";
                input-enable;
            };
        };
        &i2c1 {
            // I2C channel support HDMI transmitter.
            pinctrl-0 = <&i2c1_pins>; // Set I2C mode for these pins
            pinctrl-names = "default";
            status = "okay";
            adv7535: hdmi@3d {
                compatible = "adi,adv7535";
                reg = <0x3d>;    // I2C address of HDMI transmitter
                interrupt-parent = <&pinctrl>;
                interrupts = <RZG2L_GPIO(2, 1) IRQ_TYPE_EDGE_FALLING>;
                clocks = <&osc1>;
                clock-names = "cec";
                avdd-supply = <&reg_1p8v>;
                dvdd-supply = <&reg_1p8v>;
                pvdd-supply = <&reg_1p8v>;
                a2vdd-supply = <&reg_1p8v>;
                v3p3-supply = <&reg_3p3v>;
                v1p2-supply = <&reg_1p8v>;
                adi,dsi-lanes = <4>;    // Number of available supported DSI
data lanes
            };
            ports {
                // Input/Output endpoints (0: input from
RZ/G2L DSI output; 1: output for HDMI connector)
                #address-cells = <1>;
                #size-cells = <0>;
                port@0 {
                    reg = <0>;
                    adv7535_in: endpoint@0 {
                        remote-endpoint = <&dsi0_out>;
                    };
                port@1 {
                    reg = <1>;
                    adv7535_out: endpoint@1 {
                        remote-endpoint = <&hdmi_con_out>;
                    };
                };
            };
        };

```

Figure 5-11 Device node for HDMI transmitter ADV7533

```

        &dsi {
            status = "okay";

```

```

        ports {
            port@1 {
                dsi0_out: endpoint {
                    remote-endpoint = <&adv7535_in>; // Phandle to
HDMI transmitter.
                    data-lanes = <1 2 3 4>; // Maximum DSI data
lanes can be supported.
                };
            };
        };

        &du {
            status = "okay";
        };

```

Figure 5-12 Enable Display Unit (DU) and MIPI DSI nodes

Note: All the information in the above device tree is used for RZ/G2L and RZ/V2L. To config for RZ/V2N, please add the above device tree information in **r9a09g056n44-evk.dts** instead of **rzg2l-smarc.dtsi**. The information in Figure 5-9 and Figure 5-11 are the same as RZ/G2L, RZ/V2L, RZ/V2N and RZ/G3E, about the information in Figure 5-10 please use the below information:

```

        &pinctrl {
            i2c3_pins: i2c3 {
                pinmux = <RZG2L_PORT_PINMUX(3, 6, 1)>, /* I2C3_SDA */
                    <RZG2L_PORT_PINMUX(3, 7, 1)>; /* I2C3_SCL */
            };
        };

        &i2c3 {
            // I2C channel support HDMI transmitter.
            pinctrl-0 = <&i2c3_pins>; // Set I2C mode for these pins
            pinctrl-names = "default";
            status = "okay";

            adv7535: hdmi@3d {
                compatible = "adi,adv7535";
                reg = <0x3d>; // I2C address of HDMI transmitter

                clocks = <&oscl>;
                clock-names = "cec";
                avdd-supply = <&reg_1p8v>;
                dvdd-supply = <&reg_1p8v>;
                pvdd-supply = <&reg_1p8v>;
                a2vdd-supply = <&reg_1p8v>;
                v3p3-supply = <&reg_3p3v>;
                v1p2-supply = <&reg_1p8v>;

                interrupt-parent = <&pinctrl>;
                interrupts = <RZG2L_GPIO(7, 1) IRQ_TYPE_LEVEL_LOW>;

                adi,dsi-lanes = <4>; // Number of available supported DSI
data lanes

                ports {
                    // Input/Output endpoints (0: input from
RZ/V2N DSI output; 1: output for HDMI connector)
                    #address-cells = <1>;
                    #size-cells = <0>;
                    port@0 {
                        reg = <0>;
                        adv7535_in: endpoint@0 {
                            remote-endpoint = <&dsi0_out>;
                        };
                    };
                    port@1 {
                        reg = <1>;
                        adv7535_out: endpoint@1 {

```

```
};  
};  
};  
};  
};
```

Figure 5-13 Device node for HDMI transmitter ADV7535 (RZ/V2N)

```

&pinctrl {
    i2c7_pins: i2c7 {

        pinmux = <RZV2H_PORT_PINMUX(A, 4, 4)>, /* SCL */
                <RZV2H_PORT_PINMUX(A, 5, 4)>; /* SDA */
    };
};

&i2c7 {

    pinctrl-0 = <&i2c7_pins>;
    pinctrl-names = "default";

    status = "okay";

    adv7535: hdmi@3d {
        compatible = "adi,adv7535";
        reg = <0x3d>;

        clocks = <&oscl>;
        clock-names = "cec";
        avdd-supply = <&reg_1p8v>;
        dvdd-supply = <&reg_1p8v>;
        pvdd-supply = <&reg_1p8v>;
        a2vdd-supply = <&reg_1p8v>;
        v3p3-supply = <&reg_3p3v>;
        v1p2-supply = <&reg_1p8v>;
        interrupt-parent = <&pinctrl>;
        interrupts = <RZV2H_GPIO(L, 4) IRQ_TYPE_EDGE_FALLING>;

        adi,dsi-lanes = <4>;

        ports {

            #address-cells = <1>;
            #size-cells = <0>;

            port@0 {
                reg = <0>;
                adv7535_in: endpoint@0 {
                    remote-endpoint = <&dsi0_out>;
                };
            };

            port@1 {
                reg = <1>;
                adv7535_out: endpoint@1 {
                    remote-endpoint = <&hdmi_con_out>;
                };
            };

        };
    };
};
};

```

Figure 5-14 Device node for HDMI transmitter ADV7535 (RZ/G3E)

```

/ {
    rgb-to-hdmi-out {
        compatible = "hdmi-connector";
        type = "d";

        port {
            rgb_to_hdmi_out: endpoint {
                remote-endpoint = <&adv7513_out>;
            };
        };
    };
};

&pinctrl {
    du_pins: du {
        clk {
            pinmux = <RZV2H_PORT_PINMUX(5, 0, 1)>; /* LCD_CLK */
            renesas,output-impedance = <3>;
        };

        sync {
            pinmux = <RZV2H_PORT_PINMUX(5, 1, 1)>, /* LCD_HSYNC */
                    <RZV2H_PORT_PINMUX(5, 2, 1)>; /* LCD_VSYNC */
        };

        de {
            pinmux = <RZV2H_PORT_PINMUX(5, 3, 1)>; /* LCD_DE */
        };

        data {
            pinmux = <RZV2H_PORT_PINMUX(5, 4, 1)>, /* LCD_G7 */
                    <RZV2H_PORT_PINMUX(5, 5, 1)>, /* LCD_B7 */
                    <RZV2H_PORT_PINMUX(5, 6, 1)>, /* LCD_R7 */
                    <RZV2H_PORT_PINMUX(6, 0, 1)>, /* LCD_G6 */
                    <RZV2H_PORT_PINMUX(6, 1, 1)>, /* LCD_B6 */
                    <RZV2H_PORT_PINMUX(6, 2, 1)>, /* LCD_R6 */
                    <RZV2H_PORT_PINMUX(6, 3, 1)>, /* LCD_G5 */
                    <RZV2H_PORT_PINMUX(6, 4, 1)>, /* LCD_B5 */
                    <RZV2H_PORT_PINMUX(6, 5, 1)>, /* LCD_R5 */
                    <RZV2H_PORT_PINMUX(6, 6, 1)>, /* LCD_G4 */
                    <RZV2H_PORT_PINMUX(7, 0, 1)>, /* LCD_B4 */
                    <RZV2H_PORT_PINMUX(7, 1, 1)>, /* LCD_R4 */
                    <RZV2H_PORT_PINMUX(7, 2, 1)>, /* LCD_G3 */
                    <RZV2H_PORT_PINMUX(7, 3, 1)>, /* LCD_B3 */
                    <RZV2H_PORT_PINMUX(7, 4, 1)>, /* LCD_R3 */
                    <RZV2H_PORT_PINMUX(7, 5, 1)>, /* LCD_G2 */
                    <RZV2H_PORT_PINMUX(7, 6, 1)>, /* LCD_B2 */
                    <RZV2H_PORT_PINMUX(7, 7, 1)>, /* LCD_R2 */
                    <RZV2H_PORT_PINMUX(8, 0, 1)>, /* LCD_G1 */
                    <RZV2H_PORT_PINMUX(8, 1, 1)>, /* LCD_B1 */
                    <RZV2H_PORT_PINMUX(8, 2, 1)>, /* LCD_R1 */
                    <RZV2H_PORT_PINMUX(8, 3, 1)>, /* LCD_G0 */
                    <RZV2H_PORT_PINMUX(8, 4, 1)>, /* LCD_B0 */
                    <RZV2H_PORT_PINMUX(8, 5, 1)>; /* LCD_R0 */
        };
    };
};

&du {
    pinctrl-0 = <&du_pins>;
    pinctrl-names = "default";

    ports {
        port@1 {
            du_out_rgb: endpoint {

```

```

remote-endpoint = <&adv7513_in>;
    };
};

&i2c7 {
    adv7513: hdmi@39 {
        compatible = "adi,adv7513";
        reg = <0x39>, <0x29>, <0x49>, <0x59>;
        reg-names = "main", "cec", "edid", "packet";

        clocks = <&osc1>;
        clock-names = "cec";

        avdd-supply = <&reg_1p8v>;
        dvdd-supply = <&reg_1p8v>;
        pvdd-supply = <&reg_1p8v>;
        dvdd-3v-supply = <&reg_3p3v>;
        bgvdd-supply = <&reg_1p8v>;

        adi,input-depth = <8>;
        adi,input-colorspace = "rgb";
        adi,input-clock = "1x";

        ports {
            #address-cells = <1>;
            #size-cells = <0>;

            port@0 {
                reg = <0>;
                adv7513_in: endpoint {
                    remote-endpoint = <&du_out_rgb>;
                };
            };

            port@1 {
                reg = <1>;
                adv7513_out: endpoint {
                    remote-endpoint = <&rgb_to_hdmi_out>;
                };
            };
        };
    };
};

```

Figure 5-15 Device node for HDMI transmitter ADV7513 (RZ/G3E Parallel Interface)

```

/ {
    lvds-to-hdmi-out {
        compatible = "hdmi-connector";
        type = "d";

        port {
            lvds_to_hdmi_con_out: endpoint {
                remote-endpoint = <&it6263_out>;
            };
        };
    };
};

&i2c7 {
    it6263: it6263@4c {
        compatible = "ite,it6263";
        reg = <0x4c>;
        ite,dual-link;

        ports {
            #address-cells = <1>;

```

```

        #size-cells = <0>;

        port@0 {
            reg = <0>;
            dual-lvds-odd-pixels;
            lvds0_in0: endpoint@0 {
                remote-endpoint = <&lvds0_out>;
            };
        };
        port@1 {
            reg = <1>;
            dual-lvds-even-pixels;
            lvds0_in1: endpoint@1 {
                remote-endpoint = <&lvds1_out>;
            };
        };
        port@2 {
            reg = <2>;
            it6263_out: endpoint@2 {
                remote-endpoint = <&lvds_to_hdmi_con_out>;
            };
        };
    };
};
&lvds {
    status = "okay";
};
&lvds0 {
    status = "okay";

    ports {
        port@1 {
            lvds0_out: endpoint {
                remote-endpoint = <&lvds0_in0>;
            };
        };
    };
};
&lvds1 {
    clocks = <&cpg CPG_MOD 416>,
            <&cpg CPG_MOD 418>;

    status = "okay";

    ports {
        port@1 {
            lvds1_out: endpoint {
                remote-endpoint = <&lvds0_in1>;
            };
        };
    };
};
};

```

Figure 5-16 Device node for HDMI transmitter IT6263 (RZ/G3E LVDS)

5.2.3 Kernel parameters

By adding the following parameters to “bootargs” which is an environment variable of u-boot, the resolution at a kernel start-up and a setup of a pixel format can be changed. When not setting up, it becomes the value of 32bpp and recommended resolution in HDMI output. User must set the resolution size to support the monitor.

Please add to “bootargs” the command which underline drawn

```
video=[name of connector]:[width x height][-<bpp>][@<refresh rate>]
```

Table 5-1 Video Mode Input Parameters

connector name	HDMI-A-1 (HDMI output)
[width x height]	Please specify resolution which monitor is supported.
bpp	[16] RGB565 / [32] ARGB8888
refresh rate	Please specify refresh rate.

Notes:

1. If the configuration of “Enable legacy fbdev support for your modesetting driver” is disable, bootargs option is not available.s
2. The resolution parameter is calculated by CVT algorithm or GTF algorithm if the resolution is not in the EDID information.

Example: Set FullHD 1080p to “bootargs”:

```
bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20
video=HDMI-A-1:1920x1080-32@60
```

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

Rev.	Date	Description	
		Page	Summary
0.50	Apr. 30, 2021	—	First Edition issued
1.0	Jul. 15, 2021	—	No change. Keep revision 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 devicetree nodes for BSP-3.0.0
1.4	May. 31, 2022	—	No change. Keep revision to keep consistent with other documents
1.5	Jun. 24, 2022	—	No change. Keep revision to keep consistent with other documents
1.6	Sep. 15, 2022	—	No change. Keep revision to keep consistent with other documents
1.7	Dec. 15, 2022	—	No change. Keep revision to keep consistent with other documents
1.8	Mar. 15, 2023	—	No change. Keep revision to keep consistent with other documents
1.9	Mar. 31, 2025	3,9, 10, 11,12, 13,14, 15	Develop driver rzg2l-du based on driver rcar-du, update pixel format that supported. Update devicetree nodes for BSP-3.0.7
1.10	May. 30, 2025	1 10	- Add MPU information support for both kernel versions v5.10 and v6.1. - Correct the path of vspd and fcp.
1.11	Jun. 30, 2025	—	Add information for RZ/V2N
1.12	Jul. 22, 2025	— 12	Add information for RZ/G3E Add "4.4 Multiple Display Support"

Linux Interface Specification Device Driver LCDC
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 and RZ/G3E
Group



Renesas Electronics Corporation