# Overview

## Overview

This manual explains the display driver module (this module) that controls the Display Unit and VSPD on R-Car H3 / M3 / M3N / E3 / D3 / V3U / V3H.

## Function

This module controls VSPD to be equipped with R-Car H3 / M3 / M3N / E3 / D3 / V3U / V3H and blends an image.

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

* Multi plane

It's possible to blend at maximum five plane per VSPD 1ch. one plane is used a desktop, and another of four plane can be used overlays.

* Multi display

It's possible to be displayed in independence per channel.

* Pixel format

RGB332, RGB565, ARGB4444, XRGB4444, ARGB1555, XRGB1555, BGR888, RGB888, BGRA8888, BGRX8888, ARGB8888, XRGB8888, UYVY, YUYV, YVYU, NV12, NV21, NV16, NV61, YUV420, YVU420, YUV422, YVU422, YUV444, YVU444, RGBX1010102\*1, RGBA1010102\*1, ARGB2101010\*1

Note: 1. RGBX1010102, RGBA1010102, ARGB2101010 formats are supported only in R-Car V3U

* Alpha blend

This is the function to change the transmitted color of plane. There is plane alpha (layer uniform transmission) and pixel alpha (transmission in pixels).

Plane alpha: RGB332, RGB565, XRGB4444, XRGB1555, BGR888, RGB888, BGRX8888, XRGB8888, RGBX1010102\*1, RGBA1010102\*1, ARGB2101010\*1

Pixel alpha: ARGB1555, ARGB4444, BGRA8888, ARGB8888, RGBA1010102\*1, ARGB2101010\*1

Plane alpha and pixel alpha: ARGB4444, BGRA8888, ARGB8888, RGBA1010102\*1, ARGB2101010\*1

Note: 1. RGBX1010102, RGBA1010102, ARGB2101010 formats are supported only in R-Car V3U

* Clipping

This is the function to clip an image of a frame buffer area.

* Plane Order

This is the function to change the display priority of the plane.

* VSPD0, VSPD1 and VSPD2 are supported (Please refer to 1.6 Section for more details)
* Hot plug (HDMI connection)
* Display List support

This function automatically downloads the register settings without CPU intervention from external memory.

* Screen shot (Write back function) support
* Vmute function
* Color key support

Note: 1. RGBX1010102, RGBA1010102, ARGB2101010 formats are not supported

* The LVDS-IF supports the Dual-link output by using vertical stripe output function.(R-Car E3 only)
* DSI/CSI-2-TX-IF Connection (R-Car V3U only)
* Safe Rendering Support

### Display Resolution

Supported display resolution of this module is as follows.

If there is no explanation below, each resolution is an output in progressive mode.

If the monitor supports interlaced mode, interlaced mode can also be displayed.

Refer to "4.1.6 Resolution Change" and “5.2.3 Kernel Parameters” to change resolution.

Table 1.1 Supported resolution (R-Car H3 / M3 / M3N system evaluation board)

| Display resolution | HDMI \*1 | Analog RGB \*2 | LVDS |
| --- | --- | --- | --- |
| VGA (640x480) | yes | yes | no \*5 |
| WVGA (800x480) | yes | yes \*4 | no \*5 |
| SVGA (800x600) | yes | yes | no \*5 |
| WSVGA (1024x600) | yes | yes \*4 | no \*5 |
| XGA (1024x768) | yes | yes | yes |
| FWXGA (1280x720) | yes | yes \*4 | no \*5 |
| WXGA (1280x768) | yes | yes\*4 | no \*5 |
| 1080i (1920x1080i) | yes | yes \*4 | no \*5 |
| 1080p (1920x1080) | yes | no | no \*5 |
| WUXGA (1920x1200) | yes | no | no |
| 4k (3840x2160) \*3 | yes | no | no |

Table 1.2 Supported resolution (R-Car E3 / D3 system evaluation board)

| Display resolution | HDMI \*1 | Analog RGB \*2 | LVDS |
| --- | --- | --- | --- |
| VGA (640x480) | yes | yes | no \*5 |
| WVGA (800x480) | yes | yes \*4 | no \*5 |
| SVGA (800x600) | yes | yes | no \*5 |
| WSVGA (1024x600) | yes | yes\*4 | no \*5 |
| XGA (1024x768) | yes | yes | yes |
| WXGA (1280x768) | yes | no | no \*5 |
| SXGA (1280x1024) | yes | no | no \*5 |
| (1920x720) | yes | no | no \*5 |

Interlaced mode is not supported in R-Car D3 / E3.

Table 1.3 Supported resolution (R-Car V3U system evaluation board)

| Display resolution | DSI/CSI-2-TX |
| --- | --- |
| VGA (640x480) | yes |
| SVGA (800x600) | yes |
| XGA (1024x768) | yes |
| FWXGA (1280x720) | yes |
| WXGA (1280x768) | yes |
| 1080p (1920x1080) | yes |
| WUXGA (1920x1200) | yes |

Interlaced mode is not supported in R-Car V3U.

Table 1.4 Supported resolution (R-Car V3H system evaluation board – Condor & Condor-I)

| Display resolution | HDMI \*1 |
| --- | --- |
| VGA (640x 480) | yes |
| WVGA (800x 480) | yes |
| SVGA (800x 600) | yes |
| WSVGA (1024x 600) | yes |
| XGA (1024x 768) | yes |
| SXGA (1280x1024) | yes |
| 1080p (1920x 1080) | yes |

Interlaced mode is not supported in R-Car V3H.

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.

2. The initial value is XGA (1024x768). When you want to display more than XGA size, please specify resolution size as a kernel parameter and boot kernel. Please refer to 5.2.3 Kernel Parameters in detail.

3. The refresh rate of 4k is supported 30Hz only.

4. About the default resolution of Analog RGB, these resolutions can be displayed by modifying the driver source code or bootargs. Please refer to [Analog RGB] in “4.1.7 Add Resolution Setting” and “5.2.3 Kernel Parameters”.

5. About the resolution of LVDS, these resolutions other than XGA is not supported in this driver. However, these resolutions may be possible to display in LVDS panel dependent, please refer to [LVDS output] in “4.1.7 Add Resolution Setting”.

### Pixel Format

Supported pixel format of this module is as follows.

Refer to "4.1.8 Pixel Format Change" to change pixel format in DRM access.

Table 1.5 Supported pixel format

| Pixel format | DRM FourCC macro name | Support |
| --- | --- | --- |
| RGB332 | DRM\_FORMAT\_RGB332 | yes |
| ARGB4444 | DRM\_FORMAT\_ARGB4444 | yes |
| XRGB4444 | DRM\_FORMAT\_XRGB4444 | yes |
| ARGB1555 | DRM\_FORMAT\_ARGB1555 | yes |
| XRGB1555 | DRM\_FORMAT\_XRGB1555 | yes |
| RGB565 | DRM\_FORMAT\_RGB565 | yes |
| BGR888 | DRM\_FORMAT\_BGR888 | yes |
| RGB888 | DRM\_FORMAT\_RGB888 | yes |
| BGRA8888 | DRM\_FORMAT\_BGRA8888 | yes |
| BGRX8888 | DRM\_FORMAT\_BGRX8888 | yes |
| ARGB8888 | DRM\_FORMAT\_ARGB8888 | yes |
| XRGB8888 | DRM\_FORMAT\_XRGB8888 | yes |
| RGBX1010102 | DRM\_FORMAT\_RGBX1010102 | yes \*1 |
| RGBA1010102 | DRM\_FORMAT\_RGBA1010102 | yes \*1 |
| ARGB2101010 | DRM\_FORMAT\_ARGB2101010 | yes \*1 |
| UYVY | DRM\_FORMAT\_UYVY | yes |
| YUYV | DRM\_FORMAT\_YUYV | yes |
| YVYU | DRM\_FORMAT\_YVYU | yes |
| NV12 | DRM\_FORMAT\_NV12 | yes |
| NV21 | DRM\_FORMAT\_NV21 | yes |
| NV16 | DRM\_FORMAT\_NV16 | yes |
| NV61 | DRM\_FORMAT\_NV61 | yes |
| YUV420 | DRM\_FORMAT\_YUV420 | yes |
| YVU420 | DRM\_FORMAT\_YVU420 | yes |
| YUV422 | DRM\_FORMAT\_YUV422 | yes |
| YVU422 | DRM\_FORMAT\_YVU422 | yes |
| YUV444 | DRM\_FORMAT\_YUV444 | yes |
| YVU444 | DRM\_FORMAT\_YVU444 | yes |

Notes: 1. RGBX1010102, RGBA1010102, ARGB2101010 formats are supported only in R-Car V3U.

### Connector

Supported connector of this module is as follows.

Table 1.6 Supported connector (R-Car H3 / M3 / M3N system evaluation board)

| Output signal | Connector’s number |
| --- | --- |
| Analog RGB | CN15 |
| HDMI0 | CN16 |
| HDMI1 (R-Car H3 only) | CN17 |
| LVDS \*1 | CN18/CN19 |

Note: 1. The backlight of LVDS is controlled by a GPIO terminal (GPIO6\_07). The backlight is ON when kernel starting. If you dynamically want to control, please use the GPIO of API. Please refer to Linux Interface Specification Device Driver GPIO user's manual for details.

Table 1.7 Supported connector (R-Car E3 system evaluation board)

| Output signal | Connector’s number |
| --- | --- |
| Analog RGB | CN15 |
| HDMI | CN37 |
| LVDS0\*2 | CN18/CN19, CN40/CN50(TFT) |
| LVDS1\*2 | CN38/CN39 |

Note: 2. The backlight of LVDS is directly connected to 3.3V DC and cannot be controlled using GPIOs.

Table 1.8 Supported connector (R-Car V3U system evaluation board)

| Output signal | Connector’s number |
| --- | --- |
| DSI-TX-IF0 | CN5 |

Table 1.9 Supported connector (R-Car V3H system evaluation board – Condor & Condor-I)

| Output signal | Number of connector |
| --- | --- |
| HDMI\*3 | CN5 |

Note: 3. The LVDS interface is connected to a HDMI output. The LVDS signals from the R-CarV3H are converted to digital RGB signals by the THC63LVD1024 LVDS receiver, and these in turn are converted to HDMI signals by the ADV7511WBSWZ HDMI transmitter.

Table 1.10 Supported connector (R-Car D3 system evaluation board)

| Output signal | Connector’s number |
| --- | --- |
| Analog RGB | CN15 |
| HDMI | CN37 |
| LVDS0\*4 | CN18/CN19 |
| LVDS1\*4 | CN38/CN39 |

Note: 4. The backlight of LVDS is controlled by a GPIO terminal (D3:GP2\_31 (LVDS0)/GP4\_00 (LVDS1)) . The backlight is ON when kernel starting. If you dynamically want to control, please use the GPIO of API. Please refer to Linux Interface Specification Device Driver GPIO user's manual for details.

## Related Document

The related document to this module is as follows.

Table 1.10 Related document (R-Car H3 / M3 / M3N / E3 / D3 / V3U / V3H)

| **Number** | **Issue** | **Title** | **Edition** | **Date** |
| --- | --- | --- | --- | --- |
| - | Renesas Electronics | R-Car Series, 3rd Generation User’s Manual: Hardware | Rev.2.20 | Jun. 30, 2020 |
|  | Renesas Electronics | R-Car V3U Series User's Manual | Rev.0.5 | Jul. 31, 2020 |
| - | Renesas Electronics | R-CarH3-SiP System Evaluation Board Salvator-X RTP0RC7795SIPB0011S | Rev.1.09 | May. 11, 2017 |
| - | Renesas Electronics | R-CarM3-SiP System Evaluation Board Salvator-X RTP0RC7796SIPB0011S | Rev.0.04 | Oct. 3, 2016 |
| - | Renesas Electronics | R-CarH3-SiP/M3-SiP/M3N-SiP System Evaluation Board Salvator-XS Hardware Manual | Rev.2.04 | Jul. 17, 2018 |
| - | Renesas Electronics | R-CarE3 System Evaluation Board Ebisu Hardware Manual RTP0RC77990SEB0010S | Rev.0.03 | Apr. 11, 2018 |
| - | Renesas Electronics | R-CarE3 System Evaluation Board  Ebisu-4D (E3 board 4xDRAM) Hardware Manual | Rev.1.01 | Jul. 19, 2018 |
|  | Renesas Electronics | R-CarV3U System Evaluation Board Falcon Hardware Manual | Rev.0.01 | Sep. 11, 2020 |
| - | Renesas Electronics | R-Car V3H\_2: Additional Document for User’s Manual: Hardware | Rev.0.50 | Jul 31, 2020 |
| - | Renesas Electronics | R-CarV3H System Evaluation Board Condor-I Hardware Manual | Rev.0.02 | Nov 11, 2020 |
| - | Renesas Electronics | R-CarD3 System Evaluation Board Hardware Manual RTP0RC77995SEB0010S | Rev.1.20 | Jun. 25, 2017 |

## Restrictions

There is no restriction.

## 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. If you do not want to use the FBDev access, please do the setting of the configuration regarding 5.2.1 Kernel Configuration.
* A lengthwise parameter is corrected irrespective of the format at the time of interlaced mode. The clipping starting height, the clipping lengthwise width, the display starting height and the display lengthwise width will be 2 pixels aligning by pixel format. (It's because the progressive picture input to VSPD is changed to an interlace picture, and it's outputting.)
* In R-Car E3, TVM1 bit in DSYSR1 register is Master Mode only by H/W specification, so the sync signal of DU1 can not be stopped.

## Plane access

An access related figure of the plane by each device is indicated below.

Please refer to “4.1 External Interface for DRM/KMS Driver ” about method control of plane. Plane1 and Plane3 will be common H/W plane of DU0 / DU1 or DU2 / DU3. VSPD uses one of H/W plane of DU and displays a blended image through BRU/BRS and WPF0/WPF1. The number of RPF is multi-plane number.

VSPDL (VSPD0)

VSPD1

Multi plane:

One plane is assigned to desktop plane.

Remaining planes are assigned as overlays.

RPF0-RPF4 of VSPD0 is shared in DU0 and DU3

Two planes (One of them is desktop plane) of the RPF 0-4 are assigned to DU 3.

Plane1

Plane3

RPF3

RPF2

RPF1

RPF0

Plane1

LVDS

HDMI 0

HDMI 1

RPF4

Plane3

Analog RGB

RPF3

RPF2

RPF1

RPF0

RPF4

VSPD2

RPF3

RPF2

RPF1

RPF0

RPF4

BRU

BRU

BRU

WPF0

WPF0

WPF0

DU0 / DU1

DU2 / DU3

DU1

DU0

DU2

DU3

BRS\*

WPF1

Figure 1‑1 accesses of layers (R-Car H3)

\* The number of RPFs used in BRS can be selected with dtsi file. Please refer to 4.5 BRS number settingfor details.

VSPD1

Multi plane:

One plane (RPF0) is assigned to desktop plane.

Remaining four planes (RPF1-4) are assigned as overlays.

VSPD0

Plane1

Plane3

RPF3

RPF2

RPF1

RPF0

Plane1

LVDS

HDMI 0

RPF4

Analog RGB

RPF3

RPF2

RPF1

RPF0

RPF4

VSPD2

RPF3

RPF2

RPF1

RPF0

RPF4

BRU

BRU

BRU

WPF0

WPF0

WPF0

DU0 / DU1

DU2

DU1

DU0

DU2

Figure 1‑2 accesses of layers (R-Car M3)

VSPDL (VSPD0)

VSPD1

Plane1

Plane3

RPF3

RPF2

RPF1

RPF0

LVDS

HDMI 0

RPF4

Plane3

Analog RGB

RPF3

RPF2

RPF1

RPF0

RPF4

BRU

BRU

WPF0

WPF0

DU0 / DU1

DU3

DU1

DU0

DU3

BRS\*

WPF1

Multi plane:

One planes is assigned to desktop plane.Remaining planes are assigned as overlays.

RPF0-RPF4 of VSPD0 is shared in DU0 and DU3

Two planes (One of them is desktop plane) of the RPF 0-4 are assigned to DU 3.

Figure 1‑3 accesses of layers (R-Car M3N)

\* The number of RPFs used in BRS can be selected with dtsi file. Please refer to 4.5 BRS number settingfor details.

|  |
| --- |
| VSPD1  Multi plane:  One plane (RPF0) is assigned to desktop plane.  Remaining four planes (RPF1-4) are assigned as overlays.Multi plane:  Multi plane:  One plane (RPF0) is assigned to desktop plane.  Remaining four planes (RPF1-4) are assigned as overlays.  VSPD0  Plane1  RPF3      RPF2  RPF1      RPF0  RPF4      RPF3      RPF2  RPF1      RPF0  RPF4      BRU  BRU  WPF0  WPF0  DU0 / DU1  DU1  DU0  Plane2  LVDS1  HDMI  LVDS0  Analog RGB  ADV7511W |

Figure 1‑4 accesses of layers (R-Car E3 / D3)

|  |
| --- |
| Multi plane:  One plane (RPF0) is assigned to desktop plane.  Remaining four planes (RPF1-4) are assigned as overlays.Multi plane:  α blending only work with RPF0 and RPF1  VSPD1  VSPD0  Plane1  RPF3      RPF2  RPF1      RPF0  RPF4      RPF3      RPF2  RPF1      RPF0  RPF4      BRU  BRU  WPF0  WPF0  DU0 / DU1  DU1  DU0  Plane2  DSI/CSI-2-TX 0 |

Figure 1‑6 accesses of layers (R-Car V3U)

|  |
| --- |
| VSPD0  Multi plane:  One plane (RPF0) is assigned to desktop plane.  Remaining four planes (RPF1-4) are assigned as overlays.Multi plane:  Multi plane:  One plane (RPF0) is assigned to desktop plane.  Remaining four planes (RPF1-4) are assigned as overlays.  Plane1  RPF3      RPF2  RPF1      RPF0  RPF4      BRU  BRS  WPF0  DU0  DU0  HDMI  Analog RGB (EXIO)  ADV7511W |

Figure 1‑7 accesses of layers (R-Car V3H)

## Progressive / Interlace (P/I) conversion

One input progressive image is divided into two image. One is an even field and other is an odd field at the time of interlaced mode display. A schematic of P/I conversion is indicated below.



Time

Input image size

Input image size

30p

(Full screen)

Progressive Frame on DDR

i

Input image size

Input image size

Input image size

Input image size

Input image size

Display

P/I convert

i

60i

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

Input image size

VSPDU unit

：Odd field

：Even field

Input image size

Figure 1‑6 P/I conversion

## LVDS Dual-Link (R-Car E3/D3)

Supports the Dual-link output by using vertical stripe output function.

During dual-link operation, the PLL1 in the LVDS1-IF can output the dotclock to DU1.

|  |
| --- |
| LVDS0  DU0  DU1  LVDS1  960x720  54.455 MHz x 7  960x720  54.455 MHz x 7  800x480  29.581 MHz  RGB  Vertical  Stripe  1920x720  108.91 MHz x 7  29.581 MHz x 7  dotclock  EVEN  ODD  dotclock |

Figure 1‑7 LVDS Dual-Link

# Terminology

The following table shows the terminology related to this module.

Table 2.1 Terminology

| **Terms** | **Explanation** |
| --- | --- |
| DU | Display Unit on R-Car Series, 3rd Generation |
| VSP2 | Video Signal Processing |
| VSPD | VSP2 for Display |
| 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 |
| BRU | Blend ROP Unit |
| BRS | Blend ROP Sub Unit |
| ROP | Raster Operation |
| CRTC | Cathode Ray Tube Controller |
| VESA | Video Electronics Standards Association |
| CVT | Coordinated Video Timings |
| GTF | General Timing Formula |
| 5P49V5923A/  5P49V6901A | Programmable PLL Clock Generator (IDT Co., Ltd.) |
| DSI | Display Serial Interface |
| CSI | Camera Serial Interface |

# Operating Environment

## Hardware Environment

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

Table 3.1 Hardware specification (R-Car H3 / M3 / M3N / E3 / D3 / V3U / V3H)

| **Name** | **Version** | **Manufacture** |
| --- | --- | --- |
| R-CarH3-SiP System Evaluation Board Salvator-X | - | Renesas Electronics |
| R-CarM3-SiP System Evaluation Board Salvator-X | - | Renesas Electronics |
| R-CarH3-SiP/M3-SiP/M3N-SiP System Evaluation Board Salvator-XS | - | Renesas Electronics |
| R-CarE3 System Evaluation Board Ebisu | - | Renesas Electronics |
| R-CarE3 System Evaluation Board Ebisu-4D | - | Renesas Electronics |
| R-CarV3U System Evaluation Board Falcon | - | Renesas Electronics |
| R-CarV3H System Evaluation Board Condor-I | - | Renesas Electronics |
| R-CarD3 System Evaluation Board Draak | - | Renesas Electronics |

## Module Configuration

The following figure shows the configuration of this module.

VSP2

Data

Control

Application

User mode

Analog RGB

Output

LVDS

Output

HDMI 1

Output\*2

DRM/KMS driver

/dev/dri/card0

Display

libdrm/kms

HDMI 0

Output

DU driver

This module

HDMI

LVDS

Kernel mode

Hardware

System Evaluation Board

VSPD

5P49V5923A/

5P49V6901A\*1

Clock driver

I2C4

\*1 5P49V5923A on Salvator-X and 5P49V6901A on Salvator-XS creates various dot clock to Analog RGB and LVDS

\*2 HDMI1 can be used in R-Car H3 only. R-Car M3 / M3N cannot use HDMI1.

FCPVD

Figure 3‑1 Module configuration (R-Car H3 / M3 / M3N)

|  |
| --- |
| VSP2  Application  User mode  Analog RGB  Output  LVDS0  Output  LVDS1  Output  DRM/KMS driver  /dev/dri/card0  Display  libdrm/kms  HDMI  Output  DU driver  This module  LVDS\*1  LVDS\*1  Kernel mode  Hardware  System Evaluation Board  VSPD  I2C0  FCPVD  ADV7511W  Control  data  \*1 In R-Car E3/D3, dot clock can be created by PLL in LVDS. |

Figure 3‑2 Module configuration (R-Car E3/D3)

**Figure 3‑3 Module configuration (R-Car V3U)**

VSP2

Application

User mode

SN65DSI86

mini DiplayPort

DRM/KMS driver

/dev/dri/card0

Display

libdrm/kms

DU driver

This module

DSI/CSI-2-TX\*1

Kernel mode

Hardware

System Evaluation Board

VSPD

I2C1

FCPVD

Control

data

\*1 In R-Car V3U, dot clock can be created by PLL in DSI/CSI-2 TX.

|  |
| --- |
| VSP2  Data  Control  Application  User mode  Analog RGB  Output (EXIO)  DRM/KMS driver  /dev/dri/card0  Display  libdrm/kms  HDMI  Output  DU driver  This module  LVDS  Kernel mode  Hardware  System Evaluation Board  VSPD  I2C0  FCPVD  ADV7511W |

Figure 3‑4 Module configuration (R-Car V3H)

## State Transition Diagram

There is no state transition diagram for this module.

# 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 (R-Car H3 / M3 / M3N / E3 / D3 / V3U / V3H)

| **Device node** | **Major number** | **Minor number** |
| --- | --- | --- |
| /dev/dri/card0 | 226 | 0 |

## External Interface for DRM/KMS Driver

This driver corresponds to the v2.4.104 of libdrm/libkms library.

This driver supports 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

<https://www.kernel.org/doc/html/v5.10/index.html>

### Driver name to use libdrm interface

drmOpen() is called when using a libdrm interface.

Please specify the argument of drmOpen() as follows.

drmOpen

| \*name | rcar-du |
| --- | --- |
| busid | NULL |

**Note in using Display driver:**

If user cannot execute the API with master authority, please use drmDropMaster API.

### External Interface supported Function for DRM/KMS Driver

It will describe only the necessary interface to the control of the VSPD and DU.

libkms library is supported all.

Table 4.2 List of external interfaces supported function for DRM/KMS driver

| **Support interface name** | **summary** |
| --- | --- |
| drmOpen / drmClose | File Descriptor with a master authorization is acquired / released |
| drmSetMaster / drmDropMaster | Master authority is set / released |
| drmModeGetResources / drmModeFreeResources | DRM resource information is acquired / released |
| drmModeGetConnector / drmModeFreeConnector | Connector information is acquired / released |
| drmModeGetEncoder / drmModeFreeEncoder | Encoder information is acquired / released |
| drmModeGetPlaneResources / drmModeFreePlaneResources | Plane resource information is acquired / released |
| drmModeGetPlane / drmModeFreePlane | Plane information is acquired / released |
| drmModeGetCrtc / drmModeFreeCrtc | CRTC information is acquired / released |
| drmModeAddFB2 / drmModeRmFB | FB object is created / released |
| drmModeSetPlane | Overlay display |
| drmModeSetCrtc | Setting and displaying of desktop |
| drmModePageFlip | Page flipping |
| drmModeAtomicCommit | Update display by atomic |
| drmModeAtomicAlloc / drmModeAtomicFree | Atomic object is allocated / released |
| drmModeAtomicAddProperty | Add property by atomic |
| drmModeObjectGetProperties / drmModeFreeObjectProperties | Object property information is acquired / released |
| drmModeGetProperty / drmModeFreeProperty | property information is acquired / released |
| drmModeObjectSetProperty | Object property setting |
| drmModeConnectorSetProperty | Connector property setting |
| kms\_create / kms\_destroy | KMS is created / released |
| kms\_bo\_create / kms\_bo\_destroy | Buffer object is created / released |
| kms\_bo\_map / kms\_bo\_unmap | Buffer is mapped in the user space / unmapped |
| kms\_bo\_get\_prop / kms\_get\_prop | Handle and property of buffer object and is acquired |

### External Interface Unsupported Function for DRM/KMS Driver

The libdrm/libkms library in which this driver is not supported is described.

Please do not use the interface of the following contained to a libdrm library.

Table 4.3 List of external interface unsupported function for DRM/KMS driver

| **Function name** |
| --- |
| drmModeCrtcGetGamma, drmModeCrtcSetGamma |
| drmModeSetCursor, drmModeSetCursor2 |
| drmSetContextFlags, drmGetContextFlags |
| drmCreateDrawable, drmDestroyDrawable |
| drmUpdateDrawableInfo |
| drmAgp\* |
| drmFinish |
| drmGetInterruptFromBusID |

### The display method example of overlay by DRM access

Please confirm modetest.c with libdrm/kms library for more information.

drmOpen()

drmModeGetResources

Resource information is acquired

Member of drmModePlane structure is acquired

drmModeGetPlane

Plane information is acquired

Device file open

Member of drmModeRes structure is acquired

drmModeGetEncoder

drmModeGetConnector

Connector information is acquired

Encoder information is acquired

Member of drmModeEncoder structure is acquired

Member of drmModeConnector structure is acquired

drmModeGetCrtc

CRTC information is acquired

Member of drmModeCrtc structure is acquired

kms\_create

KMS is created

kms\_bo\_create

kms\_bo\_map

kms\_bo\_get\_prop

drmModeAddFB2

drmModeSetPlane

Buffer object is created

Buffer is mapped in the user space

Handle / offset / pitch is set

FB Object is created for CRTC or Planes

Plane (overlay) is displayed and updated

Setting and allocation of frame buffer

Set number of CRTC and plane use

drmModeObjectSetProperty

Plane property is set

αblend or order is set

Each resource of DRM is acquired

drmModeSetPlane()

int drmModeSetPlane( int fd, uint32\_t plane\_id, uint32\_t crtc\_id,

uint32\_t fb\_id, uint32\_t flags,

uint32\_t crtc\_x, uint32\_t crtc\_y,

uint32\_t crtc\_w, uint32\_t crtc\_h,

uint32\_t src\_x, uint32\_t src\_y,

uint32\_t src\_w, uint32\_t src\_h);

The overlay is decided at the value indicated below. A scaling can't be performed with a Display driver.

crtc\_x: start x position on monitor

crtc\_y: start y position on monitor

crtc\_w: display width on monitor

crtc\_h: display height on monitor

src\_x: start x clipping position on source image

src\_y: start y clipping position on source image

src\_w: clipping width on source image

src\_h: clipping height on source image

The clipping is decided at the value of “src\_x/src\_y/src\_w/src\_h”

[note]

・”src\_x/src\_y/src\_w/src\_h”must to use the value to which the 16bit left shifted.

drmModeSetCrtc

Set CRTC (Create desktop plane)

### Setting of plane property

Please use drmModeObjectSetProperty() as API for setting of layer order and alpha blend which are the optional functions of overlay.

Please refer to "tests/modetest/modetest.c :set\_property ()" of libdrm for detail procedure. The notes on use are explained as follows.

[The setting method about the value specified as the 5th argument of a drmModeObjectSetProperty() function]

When alpha blend setting

|  |  |
| --- | --- |
| symbol name | alpha |

A layer alpha blending (uniformly alpha blending of plane whole) rate is set up at the time of RGB332, XRGB4444, XRGB1555, RGB565, BGR888, RGB888, XRGB8888, BGRX8888, XRGB1555, RGBX1010102, RGBA1010102 and ARGB2101010 format specification.

The pixel alpha blend is performed at the time of ARGB1555, ARGB4444, BGRA8888, ARGB8888, RGBA1010102 and ARGB2101010 format specification.

Please use ARGB4444, BGRA8888, ARGB8888, RGBA1010102 and ARGB2101010 when you use a pixel transparent. The pixel value bit on a drawing buffer can perform alpha blend per pixel. In addition, while using the pixel alpha, ARGB4444, BGRA8888, ARGB8888, RGBA1010102 and ARGB2101010 can be used at the same time of plane alpha and pixel alpha by setting alpha in drmModeObjectSetProperty().

Nontransparent value = 255, Semitransparent value = 128, Full transparent value = 0

This value is common to each plane. The default value of each planes is "255". The value can be specified from 0 to 255.

Notice: Once a value sets up, a value will be kept within a Display driver. When you display multiplane, please be sure to check a value. Please perform a re-setup, if required.

Moreover, a setup of alpha bit function of ARGB1555 can be set according to a kernel configuration. The function transparent by either alpha bit = 0 or alpha bit = 1. Please refer to chapter 5.2.1 in detail.

When plane order setting

|  |  |
| --- | --- |
| symbol name | zpos |

A value of each layer is compared with specified value. A layer with a large value turns into a high priority layer.

When the value of "zpos" is the same, the plane which a plane id is larger becomes a high priority plane. The value can be specified from 1 to 4.

A related figure is shown below.

**When zpos value of each layer is the same**

Priority

High

Low

Priority

High

**When zpos value of each layer is differ**

Plane2 id: 23

Plane4 id: 25

Plane3 id: 24

zpos=1

zpos=1

zpos=1

zpos=3

zpos=2

zpos=4

Plane1 id: 22

Plane3 id: 24

Plane2 id: 23

Ex)

Plane1 id: 22

zpos=1

Low

zpos=1

Plane4 id: 25

Figure 4‑1 Figure of plane priority relation

When color key and color key alpha setting

|  |  |
| --- | --- |
| symbol name | colorkey |
| symbol name | colorkey\_alpha |

The 24th bit of value is bit of colorkey ON/OFF. 1'b24: ON, 0'b24: OFF

23 to 0 bit of value specifies colors. (R [23:16], G [15:8], B [7:0])

In case of 24BPP, all of 8 bits is used, but in case of 16BPP, upper side bits are used.

The YUV format compares Y values and enables color key.

For RGBX1010102, RGBA1010102 and ARGB2101010 pixel format in V3U, the colokey function does not support on those pixel format

colorkey\_alpha can be set value from 0 to 255.

(Nontransparent value = 255, Semitransparent value = 128, Full transparent value = 0)

If colorkey alpha and pixel alpha are to be used at the same time, colorkey alpha takes precedence.

[C: colorkey ON/OFF, R: Red, G: Green, B: Blue, Y: Luminance]

[RGB 32bpp]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | C | R | R | R | R | R | R | R | R | G | G | G | G | G | G | G | G | B | B | B | B | B | B | B | B |

[RGB565]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | C | R | R | R | R | R |  |  |  | G | G | G | G | G | G |  |  | B | B | B | B | B |  |  |  |

[ARGB1555]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | C | R | R | R | R | R |  |  |  | G | G | G | G | G |  |  |  | B | B | B | B | B |  |  |  |

[YUV]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | C |  |  |  |  |  |  |  |  | Y | Y | Y | Y | Y | Y | Y | Y |  |  |  |  |  |  |  |  |

Ex) RGB565 value = 0x0100FC00 (remove green color key), XRGB888 value = 0x0100FF00 (remove green color key)

The default value of each layer is "0". The value can be specified from 0 to 0x1FFFFFF.

### Resolution Change

In order to change resolution, drmModeAddFB2() and drmModeSetCrtc() that are defined in xf86drmMode.c are used. xf86drmMode.c is contained in libdrm/libkms library. About the flow and the setting method of processing, please refer to set\_mode() of tests/modetest/modetest.c in libdrm/libkms library.

Please refer to “How to set video mode” in Wayland user’s manual if you want to change the resolution when starting Wayland.

**[Notice about resolution]**

When drmModeSetCrtc is called, a reset occurs once (blackout) to update a display parameter by the specification of H/W. When updating (When the resolution changes) DU0 or DU1, DU0 and DU1 are reset together. Then, the image of screen is flickered momentarily. The combination of DU2 and DU3 is similar.

The divided external clock or internal clock is used to generate the dot clock. However, accuracy of a dot clock may be unable to be fulfilled to the recommended resolution. In this case

* All the resolution displayed in DRM resources may be unable to display.
* The value of a refresh rate displayed in DRM resources may differ from an actual value.

### Add Resolution Setting

This chapter describes how to change a resolution in LVDS and Analog output.

**[Analog RGB output]**

Please refer to 5.2.3 Kernel Parameters.

If user want to add resolution, please add the resolution parameters to the resolution table

drm\_dmt\_modes[] = { ) of drivers/gpu/drm/drm\_edid.c.

static const struct drm\_display\_mode drm\_dmt\_modes[] = {

･･･

+/\* 1920x1080i@60Hz \*/

+{ DRM\_MODE("1920x1080i", DRM\_MODE\_TYPE\_DRIVER, 74250, 1920, 2008,

+ 2052, 2200, 0, 1080, 1084, 1094, 1125, 0,

+ DRM\_MODE\_FLAG\_PHSYNC | DRM\_MODE\_FLAG\_PVSYNC |

+ DRM\_MODE\_FLAG\_INTERLACE) },

･･･

In case of interlace mode support

drivers/gpu/drm/bridge/dumb-vga-dac.c

static int dumb\_vga\_attach(struct drm\_bridge \*bridge)

…

+ vga->connector.interlace\_allowed = 1;

…

drm\_connector\_helper\_add(&vga->connector,

&dumb\_vga\_con\_helper\_funcs);

Ex) 1080i

**[LVDS output]**

If you want to change the resolution in LVDS, please modify the following parameters.

arch/arm64/boot/dts/renesas/salvator-common.dtsi for Salvator board (H3/M3/M3N)

arch/arm64/boot/dts/renesas/r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts for Ebisu board (E3)

arch/arm64/boot/dts/renesas/r8a77995-draak.dts for Draak board (D3)

Please refer to Documentation/devicetree/bindings/display/panel/panel-lvds.txt in detail.

lvds { // lvds0 and lvds1 in case of r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts and r8a77995-draak.dts

compatible = "panel-lvds";

width-mm = <210>; // monitor width size (mm)

height-mm = <158>; // monitor height size (mm)

data-mapping = "jeida-24"; // LVDS-IF mode

or "jeida-18" or "vesa-24”

data-mirror; // LVDS-IF mode

…

panel-timing {

/\* 1024x768 @60Hz \*/

clock-frequency = <65000000>; // Dotclock (Hz)

hactive = <1024>; // display width size

vactive = <768>; // display height size

hsync-len = <136>; // Hsync lengh

hfront-porch = <20>; // Hsync front porch

hback-porch = <160>; // Hsync back porch

vfront-porch = <3>; // Vsync front porch

vback-porch = <29>; // Vsync back porch

vsync-len = <6>; // Vsync length

hsync-active = <1>; \*1 // Hsync polarity: low<0> / high<1>

vsync-active = <0>; \*1 // Vsync polarity: low<0> / high<1>

};

･････

}

\*1 There is no specification in default. When not specified, the setting value is low.

### Pixel Format Change

In order to change pixel format, drmModeAddFB2() and drmModeSetPlane() that are defined in xf86drmMode.c are used. xf86drmMode.c is contained in libdrm/libkms library.

About the flow and the setting method of processing, please refer to set\_plane() of tests/modetest/modetest.c in libdrm/libkms library.

### Vmute function (DRM)

It describes the expansion API specifications of the Vmute (VSPD mute) function.

[API specification]

|  |  |  |  |
| --- | --- | --- | --- |
| [Function] | drmCommandWrite (int fd, unsigned long drmCommandIndex, struct rcar\_du\_vmute arg, unsigned long size) | | |
|  |
| [Argument] | fd | File descriptor | |
|  | drmCommandIndex | DRM\_RCAR\_DU\_SET\_VMUTE (value is 0) | |
|  | arg | Pointer of rcar\_du\_vmute structure  　　　　　　　(Please refer to [Structure] in detail) | |
|  | size | Data size of rcar\_du\_vmute structure | |
| [Header file] | xf86drm.h | | |
| [Library file] | libdrm.so | | |
| [Returns] | 0 | Success | |
|  | -1 | Error | |
| [Error value] | EINVAL | Invalid argument | |
| [Structure] | struct rcar\_du\_vmute | | |
|  | int | crtc\_id; | [CRTCs ID] Refer to Table 4.4 about CRTCs ID. |
|  | int | on; | [1: Vmute ON, 0: Vmute OFF] |
| [Description] | Vmute (VSPD mute) function can be executed per VSPD channel. | | |
| [Remark] |  | | |

### Write back function (DRM)

It describes the expansion API specifications and use method of the write back function.

[API specification]

|  |  |  |  |
| --- | --- | --- | --- |
| [Function] | drmCommandWrite (int fd, unsigned long drmCommandIndex, struct rcar\_du\_screen\_shot arg, unsigned long size) | | |
| [Argument] | fd | File descriptor | |
|  | drmCommandIndex | DRM\_RCAR\_DU\_SCRSHOT (value is 4) | |
|  | arg | Pointer of rcar\_du\_screen\_shot structure  　　　　　　　(Please refer to [Structure] in detail) | |
|  | size | Data size of rcar\_du\_screen\_shot structure | |
| [Header file] | xf86drm.h | | |
| [Library file] | libdrm.so | | |
| [Returns] | 0 | Success | |
|  | -1 | Error | |
| [Error value] | EINVAL | Invalid argument | |
| [Description] | Screen capture of one shot is get | | |
| [Remark] | This API takes at least more than 2 frame end period to perform. (It takes maximum 3 frame end period. In addition, when other display update is executed continuously, this API may become larger than 3 vsync.). Interlaced mode is prohibited by the specification of the H/W. This API is a synchronous API. | | |
| [Structure] | struct rcar\_du\_screen\_shot | | |
|  | [member] | [type] | [summary] |
|  | buff | unsigned long | Physical address of output buffer |
|  | buff\_len | unsigned int | Size of output buffer (byte) |
|  | crtc\_id | unsigned int | CRTC Object ID |
|  | fmt | unsigned int | Output format  - DRM\_FORMAT\_RGB565  - DRM\_FORMAT\_ARGB1555  - DRM\_FORMAT\_ARGB8888 |
|  | width | unsigned int | Width of output buffer (pixel) |
|  | height | unsigned int | Height of output buffer (pixel) |

[Use method]

1. Capture buffer allocation. Please allocate a buffer by using Memory Manager (please refer to “Memory Manager User’s manual” in detail)
2. Please set the rcar\_du\_screen\_shot structure

|  |  |
| --- | --- |
| buff | Set the physical address of the buffer that was allocated in section 1. |
| buff\_len | Set the size of the buffer that was allocated in section 1. |
| crtc\_id | Set the number of CRTC Object ID to get screen shot |
| fmt | Set output format (Set ARGB8888 or ARGB1555 or RGB565) |
| width | Set capture width (Pixel size) |
| height | Set capture height (Pixel size) |

1. Execute API
2. After the capture run, please confirm the capture data of buffer that was allocated in section 1.

## DRM resource information

The DRM resource information on Connectors ID and CRTCs ID at the time of a default configuration is indicated. If you want to change the configuration (R-Car DU Gen3 HDMI Encoder Support or R-Car DU LVDS Encoder Support), there are case when the value of the ID does not match the following.

Please refer to the Table 4.4 to specify the 2nd argument (CRTCs ID) and the 6th argument (Connectors ID) of the drmModeSetCrtc() function of libdrm API. Please use reference at the specification of the 3rd argument (CRTCs ID) of drmModeSetPlane() function of libdrm API.

Table 4.4 List of DRM resources ID

|  |  |  |
| --- | --- | --- |
|  | Connectors ID | CRTCs ID |
| R-Car H3 | |  |  | | --- | --- | | Analog: | 69 | | HDMI0: | 71 | | HDMI1: | 74 | | LVDS: | 77 | | |  |  | | --- | --- | | DU0: | 64 | | DU1: | 65 | | DU2: | 66 | | DU3: | 67 | |
| R-Car M3 | |  |  | | --- | --- | | Analog: | 68 | | HDMI0: | 70 | | LVDS: | 73 | | |  |  | | --- | --- | | DU0: | 64 | | DU1: | 65 | | DU2: | 66 | |
| R-Car M3N | |  |  | | --- | --- | | Analog: | 58 | | HDMI0: | 60 | | LVDS: | 63 | | |  |  | | --- | --- | | DU0: | 54 | | DU1: | 55 | | DU3: | 56 | |
| R-Car E3 | |  |  | | --- | --- | | Analog: | 57 | | HDMI0:  (LVDS) | 59 |  | | |  |  | | --- | --- | | DU0: | 54 | | DU1: | 55 | |
| R-Car D3 | |  |  | | --- | --- | | Analog: | 57 | | HDMI0:  (LVDS) | 59 |  | | |  |  | | --- | --- | | DU0: | 54 | | DU1: | 55 | |
| R-Car V3U | |  |  | | --- | --- | | DSI/CSI-2-TX 0: | 57 | | DSI/CSI-2-TX 1 | no support |  | | |  |  | | --- | --- | | DU0: | 54 | | DU1: | no support | |
| R-Car V3H (HDMI0 x1) | |  |  | | --- | --- | | HDMI0: | 45 | | |  |  | | --- | --- | | DU0: | 43 | |

## DRM connector selection

You can control the output selection of the connector by modifying the DT (Device Tree) file

**Note**: In case of modifying DT file, Connectors ID and CRTCs ID will be changed from the default.

[In case of Salvator board]

Ex)

arch/arm64/boot/dts/renesas/r8a7795-salvator-xs.dts

&lvds0 {

#if 0 // LVDS is not output

ports {

port@1 {

lvds0\_out: endpoint {

remote-endpoint = <&lvds\_in>;

};

};

};

#endif

};

arch/arm64/boot/dts/renesas/r8a7795-salvator-xs.dts

&hdmi0 {

- status = "okay"; // HDMI0 is not output

･･･････

};

･･･････

&hdmi1 {

- status = "okay"; // HDMI1 is not output

･･･････

};

arch/arm64/boot/dts/renesas/salvator-common.dtsi

&du {

･･･････

#if 0 // Analog RGB is not output

ports {

port@0 {

endpoint {

remote-endpoint = <&adv7123\_in>;

};

};

};

#endif

};

[In case of Ebisu / Draak board]  **\*Other than the patterns below is unsupported in this driver**

Ex 1）(HDMI x 1 (Single-link), Analog RGB x 1)

There is no change in the device tree (arch/arm64/boot/dts/renesas/r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts for Ebisu board and arch/arm64/boot/dts/renesas/r8a77995-draak.dts for Draak board), it is the default setting.

In case of this case, also change DIP switch. (R-Car E3/D3)

SW44: All ON, SW 45: ON, SW47: OFF

Ex 2）(HDMI x 1 (Dual-link), Analog RGB x 1)

arch/arm64/boot/dts/renesas/r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts for Ebisu board

arch/arm64/boot/dts/renesas/r8a77995-draak.dts for Draak board

lvds-decoder　{

……

port@1 {

reg = <1>;

thc63lvd1024\_in\_dual\_link: endpoint {

+ remote-endpoint = <&lvds1\_out>;

};

};

……

};

vga-encoder {

……

port@0 {

reg = <0>;

adv7123\_in: endpoint {

- remote-endpoint = <&lvds1\_out>;

};

};

……

};

&lvds1 {

……

port@1 {

lvds1\_out: endpoint {

- remote-endpoint = <&adv7123\_in>;

+ remote-endpoint = <& thc63lvd1024\_in\_dual\_link>;

};

};

……

};

In case of this case, also change DIP switch. (R-Car E3/D3)

SW44: All ON, SW 45: OFF, SW47: ON

Ex 3）(LVDS x 2 (Single-link))

arch/arm64/boot/dts/renesas/r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts

arch/arm64/boot/dts/renesas/r8a77995-draak.dts for Draak board

lvds-decoder {

･･･････

port@0 {

thc63lvd1024\_in: endpoint {

* remote-endpoint = <&lvds0\_out>;

};

};

･･･････

};

･･･････

lvds0 {

･･･････

port {

lvds0\_panel\_in: endpoint {

+ remote-endpoint =<&lvds0\_out >

};

};

};

lvds1 {

･･･････

port {

lvds1\_panel\_in: endpoint {

+ remote-endpoint =<&lvds1\_out >

};

};

};

･･･････

vga-encoder {

･･･････

port@0 {

adv7123\_in: endpoint {

- remote-endpoint = <&lvds1\_out>;

};

};

･･･････

};

･･･････

};

･･･････

&lvds0 {

･･･････

ports {

port@1 {

lvds0\_out: endpoint {

- remote-endpoint = <&thc63lvd1024\_in>;

+ remote-endpoint = <&lvds0\_panel\_in>;

};

};

};

&lvds1 {

･･･････

ports {

port@1 {

lvds1\_out: endpoint {

- remote-endpoint = <&adv7123\_in>;

+ remote-endpoint = <&lvds1\_panel\_in>;

};

};

};

};

In case of this case, also change DIP switch. (R-Car E3/D3)

SW44: All OFF, SW 45: ON, SW47: OFF

Ex 4）(Analog RGB x 1, LVDS x 1 (Single-link))

arch/arm64/boot/dts/renesas/r8a77990-ebisu.dts, r8a77990-es10-ebisu.dts

arch/arm64/boot/dts/renesas/r8a77995-draak.dts for Draak board

lvds-decoder {

･･･････

port@0 {

thc63lvd1024\_in: endpoint {

* remote-endpoint = <&lvds0\_out>;

};

};

･･･････

};

･･･････

lvds1 {

･･･････

port {

lvds1\_panel\_in: endpoint {

+ remote-endpoint =<&lvds1\_out >

};

};

};

･･･････

vga-encoder {

･･･････

port@0 {

adv7123\_in: endpoint {

- remote-endpoint = <&lvds1\_out>;

+ remote-endpoint = <&lvds0\_out>;

};

};

･･･････

};

･･･････

};

･･･････

&lvds0 {

･･･････

ports {

port@1 {

lvds0\_out: endpoint {

- remote-endpoint = <&thc63lvd1024\_in>;

+ remote-endpoint = <&adv7123\_in>;

};

};

};

};

&lvds1 {

･･･････

ports {

port@1 {

lvds1\_out: endpoint {

- remote-endpoint = <&adv7123\_in>;

+ remote-endpoint = <&lvds1\_panel\_in>;

};

};

};

};

In case of this case, also change DIP switch. (R-Car E3/D3)

SW44: All OFF, SW 45: ON, SW47: OFF

[In case of Condor and Condor-I board]

Ex)

arch/arm64/boot/dts/renesas/r8a77980-condor.dts

arch/arm64/boot/dts/renesas/r8a77980-condor-i.dts

lvds-decoder {

･･･････

port@0 {

thc63lvd1024\_in: endpoint {

remote-endpoint = <&lvds0\_out>;

};

};

port@2 {

thc63lvd1024\_out: endpoint {

remote-endpoint = <&adv7511\_in>;

};

};

･･･････

};

&lvds0 {

ports {

port@1 {

lvds0\_out: endpoint {

remote-endpoint = <&thc63lvd1024\_in>;

};

};

};

};

&du {

status = "okay";

･･･････

};

## Hot plug Operation

This driver supports hot plug operation.

The state of a HDMI cable or a Display Port (DP) cable can be checked.

The following sample code is for getting state of connected or disconnected of HDMI and DP cabble via DRM.

{

int i, fd;

drmModeRes \*resources;

drmModeConnector \*connector;

fd = drmOpen("rcar-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");

}

if (connector->connector\_type == DRM\_MODE\_CONNECTOR\_DisplayPort) {

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 4‑2 Acquisition method of getting HDMI cable status via DRM

Moreover, the following commands are executed on target.

HDMI0

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

HDMI1 (R-Car H3 only)

# cat /sys/class/drm/card0-HDMI-A-2/status

Display Port (R-Car V3U only)

# cat /sys/class/drm/card0-DP-1/status

The following information can get.

connected

or

disconnected

### Notice about hot plug

If you start the kernel with connecting the HDMI or DP cable, the monitor displays the recommended resolution.

If the HDMI or DP cable was inserted after starting the kernel, the monitor would be displayed in XGA resolution.

Basically, please do not change the output destination display. Also, please start up the kernel with the HDMI or DP cable.

## BRS number setting

Shows how to set brs option. By adding the setting, determine the number of planes to use in BRS.

arch/arm64/boot/dts/renesas/r8a7795.dtsi or r8a77965.dtsi

vspd0: vsp@fea20000 {

compatible = "renesas,vsp2";

reg = <0 0xfea20000 0 0x4000>;

･･･

renesas,#brs = <2>; // please change number <0> or <1> or <2>.

// plane assignment is not fixed if it commented out.

renesas,fcp = <&fcpvd0>;

};

# Integration

## Directory Configuration

The directory configuration is shown below.

drivers/gpu/drm/rcar-du/

rcar\_du\_crtc.c

rcar\_du\_crtc.h

rcar\_du\_drv.c

rcar\_du\_drv.h

rcar\_du\_encoder.c

rcar\_du\_encoder.h

rcar\_du\_group.c

rcar\_du\_group.h

rcar\_du\_kms.c

rcar\_du\_kms.h

rcar\_du\_lvds.c

rcar\_du\_plane.c

rcar\_du\_plane.h

rcar\_du\_regs.h

rcar\_lvds\_regs.h

rcar\_du\_vsp.c

：LVDS driver source file

：CRTC source file

：CRTC header file

：DRM driver source file

：DRM driver header file

：DRM encoder source file

：DRM encoder header file

：DRM group source file

：DRM group header file

：KMS driver source file

：KMS driver header file

：Plane operation source file

：Plane operation header file

：DU register header file

：LVDS register header file

：vsp-du interface source file

rcar\_du\_vsp.h

：vsp-du interface header file

dw-hdmi.c

dw-hdmi.h

：dw hdmi source file

：dw hdmi header file

drivers/gpu/drm/bridge/synopsys/

clk-versaclock5.c

：5p49x source file

drivers/clk/

rcar\_dw\_hdmi.c

：dw-hdmi source file

panel-lvds.c

：LVDS panel source file

drivers/gpu/drm/panel/

drivers/gpu/drm/bridge/

dumb-vga-dac.c

：VGA bridge source file

thc63lvd1024.c

：thine thc63lvd1024 bridge source file

rcar\_du\_lvds.h

：LVDS driver header file

rcar\_mipi\_dsi.c

：DSI driver source file

rcar\_mipi\_dsi.h

：DSI driver header file

rcar\_mipi\_dsi\_regs.h

：DSI registers header file

ti-sn65dsi86.c

：ti sn65dsi86 bridge source file

vsp1\_brx.c

：VSP BRX source file

vsp1\_wpf.c

vsp1\_rwpf.h

vsp1\_rwpf.c

vsp1\_rpf.c

vsp1\_pipe.c

vsp1\_lif.h

vsp1\_lif.c

vsp1\_entity.h

vsp1\_entity.c

vsp1\_drv.c

vsp1\_drm.h

vsp1\_drm.c

vsp1\_dl.h

：VSP WPF source file

：VSP RWPF header file

：VSP RWPF source file

：VSP RPF source file

：VSP pipe source file

：VSP LIF header file

：VSP LIF source file

：VSP entity header file

：VSP entity source file

：VSP driver source file

：VSP DRM header file

：VSP DRM header file

：VSP Display List header file

vsp1\_dl.c

：VSP Display List source file

vsp1\_brx.h

：VSP BRX header file

vsp1\_regs.h

vsp1\_pipe.h

：VSP registers file

：VSP pipe header file

drivers/media/platform/vsp1/

vsp1.h

：VSP header file

include/uapi/drm/

include/drm/bridge/

dw\_hdmi.h

：dw hdmi header file

include/media/

vsp1.h

：VSP header file

rcar\_du\_drm.h

：rcar-du user application header file

drivers/media/platform/

rcar-fcp.c

：FCP source file

Figure 5‑1 Directory configuration

## Integration Procedure

### 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 device types --->

[\*] Cameras and video grabbers

Media core support --->

[\*] Media Controller API \*7

Video4Linux options --->

[\*] V4L2 sub-device userspace API

Media drivers --->

[\*] Memory-to-memory multimedia devices --->

<\*> Renesas Frame Compression Processor

<\*> Renesas VSP1 Video Processing Engine

[ ] Renesas VSP1 underrun debug messages **\*1**

(1) Renesas VSP alpha bit function of ARGB1555 **\*2**

Graphics support --->

<\*> Direct Rendering Manager(XFree86 4.1.0 and higher DRI support) --->

[\*] Enable legacy fbdev support for your modesetting driver **\*3**

<\*> DRM Support for R-Car Display Unit

<\*> R-Car DU Color Management Module (CMM) Support

<\*> R-Car Gen3 and RZ/G2 DU HDMI Encoder Support \*4

<\*> R-Car DU LVDS Encoder Support

<\*> R-Car DU MIPI DSI Encoder Support

Display Panels --->

<\*> Generic LVDS panel driver

Display Interface Bridges --->

<\*> Display connector support

<\*> Simple DRM bridge support

<\*> Thine THC63LVD1024 LVDS decoder bridge \*5

<\*> ADV7511 encoder \*6

<\*> TI SN65DSI86 DSI to eDP bridge

Common Clock Framework --->

<\*> Clock driver for IDT VersaClock 5,6 devices

Figure 5‑2 Kernel configuration

**\***1

This configuration is enabled debug message when VSP underrun occurring.

In addition, please step on the following steps after kernel starting.

1. # echo 1 > /sys/module/vsp1/parameters/debug (debug message enable and VSP underrun count starts)
2. # cat /sys/module/vsp1/parameters/underrun\_vspd (Check VSP underrun count)
3. # 0,0,0,0 (Left most indicates VSPD0): VSP underrun count) \* Please confirm vspd channel for each device.

**\***2

It sets up about the function of alpha bit of ARGB1555 format.

If 0 is specified by configuration, pixel alpha blending will be performed when the alpha bit is 0.

If 1 is specified by configuration, pixel alpha blending will be performed when the alpha bit is 1.

The default value is 0.

**\***3

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 all DU channel.

\*4

This configuration is not applicable for R-Car E3 / D3.

\*5

This configuration is applicable for R-Car E3 (Ebisu board) / R-Car D3 (Draak board).

\*6

The ADV7511 encoder can be used to output HDMI in R-Car E3/D3.

\*7

The [Media core support] option is only visible when [Filter media driver] option is unmark, so please uncheck [Filter media driver] option if you want access to Media Control API

### Size of CMA Change

About CMA change method, please refer to 7. Memory map of “Yocto recipe Start-Up Guide User’s Manual: Software “.

In DRM access, when using one overlay in Full HD size, CMA is used [1920x1080x4(32bpp) = About 8MB].

Moreover, since a CMA area may be used with other drivers, please set the CMA size in consideration of a system.

### 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 (VGA output is XGA. LVDS output is XGA). The resolution must set the resolution size to support of the monitor.

Please add to bootargs the command which underline drawn

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

|  |  |
| --- | --- |
| connector name | HDMI-A-1 (HDMI0 output), HDMI-A-2 (HDMI1 output),VGA-1(VGA output), LVDS-1(LVDS output), DP-1 (DSI-TX-IF 0 output) |
| [width x height] | Please specify resolution which monitor is supported. |
| bpp | [16] RGB565 / [32] ARGB8888 |
| refresh rate | Please specify refresh rate. |
| R option | If 'R' is specified, do a 'reduced blanking' calculation for digital displays. |
| i option | If 'i' is specified, calculate for an interlaced mode. |

Notes:

1. If the configuration of “Enable legacy fbdev support for your modesetting driver” is disable, bootargs option is not available.
2. When 'R' option is specified, even if the monitor does not support, the Display driver outputs forcibly.
3. When ‘R’ option is added, specification of a bits/pixel cannot be performed. It becomes an output of ARGB8888.
4. HDMI-A-2 (HDMI1 output) can be used in R-Car H3 only.
5. The resolution parameter is calculated by CVT algorithm or GTF algorithm if the resolution is not in the EDID information.
6. Please refer to “How to set video mode” in Wayland user’s manual if you want to change the resolution when starting Wayland.

Example) Please add the underlined part in the boot command.

[R-Car H3 / M3 / M3N]

| [XGA]:RGB565 (HDMI0 connector) |
| --- |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-1:1024x768-16@60 |
| [VGA]:ARGB8888 (Analog RGB connector) |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=VGA-1:640x480-32@60 |
| [1080i]:ARGB8888 (Analog RGB connector) \* If 4k monitor is connected to HDMI1 monitor, the picture stride at bootup may be disturbed. |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=VGA-1:1920x1080-32@60i |
| [1080i]: ARGB8888 (HDMI0 connector) \* If 4k monitor is connected to HDMI1(HDMI-A-2), the picture stride at bootup may be disturbed.  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@60i |
| [1080p]: ARGB8888 (HDMI1 connector) |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-2:1920x1080-32@60 |

[R-Car E3/D3]

| [XGA]: ARGB8888 (HDMI connector) |
| --- |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-1:1024x768-32@60 |
| [VGA]:ARGB8888 (Analog RGB connector) |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=VGA-1:800x600-32@60 |

[R-Car V3U]

| [XGA]: ARGB8888 (DSI-TX connector) |
| --- |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=DP-1:1920x1080-32@60 |

[R-Car V3H]

| [XGA]:RGB565 (HDMI0 connector) |
| --- |
| bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-1:1024x768-16@60 |
| [SXGA]: ARGB8888 (HDMI0 connector)  bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-1:1280x1024-32@60 |
| [1080p]: RGB565 (HDMI0 connector)  bootargs=console=ttySC0,115200 rw root=/dev/nfs nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20 video=HDMI-A-1:1920x1080-16@60 |

[Example 'R' option]

| video=HDMI-A-1:1920x720R@60 |
| --- |
| Reduce blanking. Resolution of 1920x720 is output to HDMI connection |