Software Manual

Linux Interface Specification GStreamer

Software Manual

RZ/V2H Group and RZ/V2N Group

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

Renesas Electronics

www.renesas.com

Rev.2.03 Oct. 30, 2026

Credit

These following platforms and projects were used to build this document.

1. Platforms
    - Architecture: `x86_64`
    - Docker: `Docker version 28.1.1, build 4eba377`
    - Container OS: `Ubuntu 22.04.5 LTS`
2. Projects
    - Python: `3.12.13`
    - pip: `26.2.1`
    - PDFtk:

```
pdftk port to java 3.2.2 a Handy Tool for Manipulating PDF Documents
Copyright (c) 2017-2018 Marc Vinyals - https://gitlab.com/pdftk-java/pdftk
Copyright (c) 2003-2013 Steward and Lee, LLC.
pdftk includes a modified version of the iText library.
Copyright (c) 1999-2009 Bruno Lowagie, Paulo Soares, et al.
This is free software; see the source code for copying conditions. There is
NO warranty, not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

    - pdfgrep:

```
This is pdfgrep version 2.1.2.

Using poppler version 22.02.0
Using libpcre version 8.39 2016-06-14
```

    - Table Generator: https://www.tablesgenerator.com/
    - Kroki!: https://kroki.io/
3. Python Libraries (pip)

> **NOTE**
>
> The table below was created using `pip-licenses`, combined with information gathered manually from each project's website.
>
> Each license is provided as a link to where it is stored.

Index	Python Libraries (pip)

1	`Markdown` | 3.9 | BSD3 License
	https://Python-Markdown.github.io/
	
	`Copyright 2007, 2008 The Python Markdown Project (v. 1.7 and later)`
	`Copyright 2004, 2005, 2006 Yuri Takhteyev (v. 0.2-1.6b)`
	`Copyright 2004 Manfred Stienstra (the original version)`

Index	Python Libraries (pip)

2	`PyYAML` | 6.0.3 | MIT License
	https://pyyaml.org/
	
	`Copyright (c) 2017-2021 Ingy döt Net`
	`Copyright (c) 2006-2016 Kirill Simonov`

3	`beautifulsoup4` | 4.14.2 | MIT License
	https://www.crummy.com/software/BeautifulSoup/bs4/
	
	`Beautiful Soup is made available under the MIT license:`
	
	`Copyright (c) Leonard Richardson`
	
	`Beautiful Soup incorporates code from the html5lib library, which is also made available under the MIT license. Copyright (c) James Graham and other contributors`
	
	`Beautiful Soup depends on the soupsieve library, which is also made available under the MIT license. Copyright (c) Isaac Muse`

4	`markdown-exec` | 1.11.0 | ISC License https://pawamoy.github.io/markdown-exec `Copyright (c) 2022, Timothée Mazzucotelli`
5	`markdown-include` | 0.8.1 | GNU General Public License v3 (GPLv3) https://github.com/cmacmackin/markdown-include `Copyright 2015 Christopher MacMackin`
6	`markdown_grid_tables` | 0.6.0 | GNU General Public License v3 (GPLv3) https://gitlab.com/WillDaSilva/markdown_grid_tables

7	`mike` | 2.1.3 | BSD License
	https://github.com/jimporter/mike
	
	`Copyright (c) 2017-2024, Jim Porter`
	`All rights reserved.`

8	`mkdocs` | 1.6.1 | BSD License https://github.com/mkdocs/mkdocs `Copyright © 2014, Tom Christie. All rights reserved.`
9	`mkdocs-awesome-pages-plugin` | 2.10.1 | MIT License https://github.com/lukasgeiter/mkdocs-awesome-pages-plugin/ `Copyright (c) 2018 Lukas Geiter`

Index	Python Libraries (pip)
10	`mkdocs-caption` | 1.3.0 | MIT License https://github.com/tobiasah/mkdocs-caption `Copyright (c) 2023 Tobias Ahrens`
11	`mkdocs-exclude` | 1.0.2 | Apache Software License https://github.com/apenwarr/mkdocs-exclude `Copyright [yyyy] [name of copyright owner]`
12	`mkdocs-glightbox` | 0.5.1 | MIT License https://blueswen.github.io/mkdocs-glightbox/ `Copyright (c) 2022 Blueswen`
13	`mkdocs-include-markdown-plugin` | 7.2.0 | Apache Software License https://github.com/mondeja/mkdocs-include-markdown-plugin `Copyright 2017-2025 Joe Rickerby and contributors`
14	`mkdocs-kroki-plugin` | 0.7.1 | MIT License https://github.com/AVATEAM-IT-SYSTEMHAUS/mkdocs-kroki-plugin `Copyright (c) 2018 Benjamin Bittner`
15	`mkdocs-macros-plugin` | 1.4.0 | MIT License https://github.com/fralau/mkdocs_macros_plugin `Copyright (C) 2018-2024 Laurent Franceschetti`
16	`mkdocs-material` | 9.6.21 | MIT License https://github.com/squidfunk/mkdocs-material `Copyright (c) 2016-2025 Martin Donath`
17	`mkdocs-material-extensions` | 1.3.1 | MIT License https://github.com/facelessuser/mkdocs-material-extensions `Copyright (c) 2021 Isaac Muse`
18	`mkdocs-minify-html-plugin` | 0.3.4 | MIT License https://github.com/monosans/mkdocs-minify-html-plugin `Copyright (c) 2023 monosans`

Index	Python Libraries (pip)
19	`mkdocs-minify-plugin` | 0.8.0 | MIT License https://github.com/byrnereese/mkdocs-minify-plugin `Copyright (c) 2018 Your Name Here`
20	`mkdocs-table-reader-plugin` | 3.1.0 | MIT License https://github.com/timvink/mkdocs-table-reader-plugin `Copyright (c) 2019 Tim Vink`
21	`mkdocs-with-pdf` | 0.9.3 | MIT License https://github.com/orzih/mkdocs-with-pdf `Copyright (c) 2020, orzih`

22	`pandas` | 2.3.3 | BSD License
	https://pandas.pydata.org
	
	`Copyright (c) 2008-2011, AQR Capital Management, LLC, Lambda Foundry, Inc. and PyData Development Team`
	`All rights reserved.`
	
	`Copyright (c) 2011-2023, Open source contributors.`

23	`pillow` | 11.3.0 | MIT-CMU License
	https://python-pillow.github.io
	
	`The Python Imaging Library (PIL) is`
	
	`    Copyright © 1997-2011 by Secret Labs AB`
	`    Copyright © 1995-2011 by Fredrik Lundh and contributors`
	
	`Pillow is the friendly PIL fork. It is`
	
	`    Copyright © 2010 by Jeffrey A. Clark and contributors.`

24	`pip-licenses` | 5.0.0 | MIT License https://github.com/raimon49/pip-licenses `Copyright (c) 2018 raimon`

25	`weasyprint` | 64.1 | BSD License
	https://weasyprint.org/
	
	`Copyright (c) 2011-2021, Simon Sapin and contributors.`
	`All rights reserved.`

Trademarks

- Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
- Other company names and product names mentioned herein are registered trademarks or trademarks of their respective owners.
- Registered trademark and trademark symbols (® and ™) are omitted in this document.

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.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

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/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2026 Renesas Electronics Corporation. All right reserved.

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 VIL (Max.) and VIH (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 VIL (Max.) and VIH (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 document is designed to provide the user with an understanding of the software development environment for RZ/V2H Group and RZ/V2N Group MPU(s). It is intended for users developing software incorporating the MPU(s). A basic knowledge of software development and Linux systems is necessary 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/V2H Group and RZ/V2N Group MPU(s). 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.
Hardware Manual	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description	RZ/V2H Group Hardware Manual	-
		RZ/V2N Group Hardware Manual	-
Renesas Technical Update	Describes all basic steps to use Yocto build environment with AI SDK for RZ/V2H and RZ/V2N	Available from AI SDK GitHub page	-

2. List of Abbreviations and Acronyms

Abbreviation	Description
ALSA	Advanced Linux Sound Architecture
API	Application Programming Interface
BSP	Board Support Package
CRU	Camera Data Receive Unit
CSI	Camera Serial Interface
DU	Display Unit on RZ/V Series
EVK	Evaluation Kit
FB	Framebuffer
FPS	Frames per second
ISU	Image Scaling Unit
MCU	Micro Controller Unit
MIPI	Mobile Industry Processor Interface

Abbreviation	Description
MPU	Micro Processor Unit
OSS	Open-Source Software
RTP	Real-time Transport Protocol
SDK	Software Development Kit
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
V4L2	Video for Linux 2
VLP	Verified Linux Package
VSP	Video Signal Processor
VSPD	VSP for DU

3. Conventions

   Command line run on Linux host PC will be shown as below:

```
$    echo "This is command line run on x86-64 Linux PC"
```

   Command line run on target board's linux kernel will be shown as below:

```
#    echo "This is command line run on ARM board"
```

   Command line run on target board's bootloader will be shown as below:

```
=>    echo "This is command line run on ARM board"
```

   File content will be shown as below:

FILENAME

```
#!/bin/bash
echo "This is content in a file"
```

Table of Contents

1. Overview 1

1.1 Related Documents 1

1.2 Terminology 1

2. Operating Environment 3

2.1 Hardware and Software Environment 3

2.2 Software Requirement 4

2.3 Module Configuration 5

2.3.1 Video Decode and Display 5

2.3.2 Video Capture and Display 6

2.3.3 Video Capture and Encode 7

3. GStreamer Tool, Plug-in and Element 8

3.1 GStreamer Tool 8

3.1.1 `gst-launch-1.0` 8

3.1.2 `gst-inspect-1.0` 8

3.1.3 `gst-discoverer-1.0` 8

3.2 GStreamer Plug-in and Element 9

3.2.1 `playbin` and `playbin3` 10

3.2.2 Video Decode Plugin 10

3.2.3 Video Capture Plugin 11

3.2.4 Video Conversion Plugin 11

3.2.5 Video Encode Plugin 11

3.2.6 Video Sink Plugin 11

3.2.7 Demux Plugin 12

3.2.8 Mux Plugin 12

3.2.9 Audio Plugin 13

3.2.10 Network Protocol Plugin 13

3.2.11 Payload/Depayload Plugin 14

3.3 Renesas Extended Element 15

3.3.1 Video H.264 Decode Element: `omxh264dec` 15

3.3.2 Video H.265 Decode Element: `omxh265dec` 18

3.3.3 Video H.264 Encode Element: `omxh264enc` 21

3.3.4 Video H.265 Encode Element: `omxh265enc` 25

3.3.5 Source Element: `v4l2src` 29

3.3.6 Video Conversion Element: `vspmfilter` 32

3.3.7 Video Sink Element: `waylandsink` 34

4. Example Pipeline 36
4.1 playbin and playbin3 36
4.1.1 playbin 36
4.1.2 playbin3 36
4.2 Video Decode Pipeline 37
4.2.1 decodebin 37
4.2.2 omxh264dec (Hardware Accelerated Video Decoder) 37
4.2.3 avdec_h264 (Software Decoder) 38
4.2.4 omxh265dec (Hardware Accelerated Video Decoder) 38
4.2.5 avdec_h265 (Software Decoder) 40
4.3 Video Conversion Pipeline 41
4.3.1 vspmfilter 41
4.3.2 videoconvert 41
4.3.3 videoscale 41
4.4 Video Encode Pipeline 42
4.4.1 omxh264enc (Hardware Accelerated Video Encoder) 42
4.4.2 omxh265enc (Hardware Accelerated Video Encoder) 43
4.5 Video Sink Pipeline 45
4.5.1 waylandsink 45
4.5.2 fbdevsink 47
4.5.3 fpsdisplaysink 47
4.5.4 glimagesink 47
4.6 Audio Pipeline 49
4.6.1 Decode MP3 49
4.6.2 Decode Vorbis stream 49
4.6.3 Decode AAC 49
4.6.4 Encode Vorbis stream 49
4.7 Network Protocol Pipeline 50
4.7.1 UDP 50
4.7.2 TCP (tcpserversink to tcpclientsrc) 50
4.7.3 TCP (tcpclientsink to tcpserversrc) 51
5. Debugging Tool 52
5.1 GST_DEBUG 52
5.2 Pipeline Graph 53
6. Performance Comparison 55
6.1 Renesas Hardware Accelerated Decoder vs Software Decoder 55
6.1.1 omxh265dec vs avdec_h265 55
6.1.2 omxh264dec vs avdec_h264 56

6.2 Running Mode Comparison: `use-dmabuf` vs `no-copy` 59
7. Limitation 60
7.1 `vspmfilter` 60
Appendix 61
A How to Write Output to Raw File Instead of Displaying Using `filesink` 61
B How to Change the Default Video Filter In `playbin` and `playbin3` Elements 62
C How to Change the Default Mode In OMX Video Decoder Element 63
D How to Detect Active Camera 64
E How to Configure and Utilize MIPI Camera 65
E.1 Setup MIPI Camera 65
E.2 Setup MIPI Camera without Device IP 66
E.3 Configuring **e-CAM22_CURZH** Camera Capture FPS 66
F How to Configure and Utilize Audio 67
Index 68

List of Figures

Figure 2-1: Video Playback 5

Figure 2-2: Video Playback (scaling and format conversion) 6

Figure 2-3: Video Capture and Display 6

Figure 2-4: Video Capture and Display (scaling and format conversion) 7

Figure 2-5: Video Capture and Encode 7

Figure 4-1: Video on Weston Screen With Setting Position 45

Figure 4-2: Fullscreen Video on Weston Screen 46

Figure 4-3: Video With Setting Size on Weston Screen 46

Figure 4-4: `fpsdisplaysink` 47

Figure 5-1: Pipeline Graph 54

List of Tables

Table 1-1 Related Documents 1

Table 1-2 Terminology 1

Table 2-1 Hardware and Software Environment 3

Table 2-2 Software Requirement 4

Table 3-1 GStreamer and GStreamer Plugins 9

Table 3-2 `playbin` and `playbin3` 10

Table 3-3 Video Decode Plugins 10

Table 3-4 Video Capture Plugin 11

Table 3-5 Video Conversion Plugins 11

Table 3-6 Video Encode Plugins 11

Table 3-7 Video Sink Plugins 11

Table 3-8 Demux Plugins 12

Table 3-9 Mux Plugins 12

Table 3-10 Audio Plugins 13

Table 3-11 Network Protocol Plugins 13

Table 3-12 Payload/Depayload Plugins 14

Table 3-13 `omxh264dec` Pad Templates 15

Table 3-14 `omxh264dec` Expansion Properties 16

Table 3-15 `omxh265dec` Pad Templates 18

Table 3-16 `omxh265dec` Expansion Properties 19

Table 3-17 `omxh264enc` Pad Templates 21

Table 3-18 `omxh264enc` Expansion Properties 22

Table 3-19 `omxh265enc` Pad Templates 25

Table 3-20 `omxh265enc` Expansion Properties 26

Table 3-21 `v4l2src` Pad Templates 29

Table 3-22 `v4l2src` Expansion Properties 30

Table 3-23 `vspmfilter` Pad Templates 32

Table 3-24 `vspmfilter` Expansion Properties 32

Table 3-25 `waylandsink` Pad Templates 34

Table 3-26 `waylandsink` Expansion Properties 35

Table 5-1 Debug Levels 52

Table 6-1 Decode H.265 Input File Information 55

Table 6-2 `omxh265dec` and `avdec_h265` Performance Comparison for RZ/V2H 56

Table 6-3 Decode H.264 Input File Information 56

Table 6-4 `omxh264dec` and `avdec_h264` Performance Comparison for RZ/V2H 57

Table 6-5 `use-dmabuf` and `no-copy` Performance Comparison for RZ/V2H 59
Table 8-1 Dip Switch 2 (DSW2) for RZ/V2H EVK and RZ/V2N EVK v1.0 67
Table 8-2 Dip Switch 2 (DSW2) for RZ/V2N EVK v2.0 67

1. Overview

This document explains the expanded specification of GStreamer Plug-in and explains how to run the GStreamer Plug-in and GStreamer on Linux on RZ/V2H Group and RZ/V2N Group MPU(s).

1.1 Related Documents

The following table shows the document related to this module.

Table 1-1 Related Documents

Index	Document Name	Remarks
1	Getting Started of RZ/V AI SDK	See: • RZ/V2H AI SDK v5.20 and RZ/V2N AI SDK v5.00 https://renesas-rz.github.io/rzv_ai_sdk/5.20/getting_started.html • RZ/V2H AI SDK v6.00 and RZ/V2N AI SDK v6.30 https://renesas-rz.github.io/rzv_ai_sdk/7.10/getting_started.html

1.2 Terminology

The following table shows the terminology related to this module.

Table 1-2 Terminology

Terms	Explanation
GStreamer	It provides functionality based on multimedia applications such as Audio/Video in multimedia framework open source URL: http://gstreamer.freedesktop.org/
GStreamer Plug-in	It is various Plug-in of GStreamer framework, provides other functionality and various codec
OpenMAX	It is general multimedia API for embedded devices such as image, audio,…, it consists of three layers as below: • OpenMAX AL (Application Layer) • OpenMAX IL (Integration Layer) • OpenMAX DL (Development Layer) URL: http://www.khronos.org/openmax/
gst-omx	GStreamer Plug-in that allows communication with OpenMAX IL component

Terms	Explanation
Wayland	Wayland is a protocol that specifies the communication between the display server (called Wayland compositor) and its clients, as a replacement for the X Window System
Weston	Weston is the reference implementation of a Wayland compositor (typical elements of a window)

2. Operating Environment

2.1 Hardware and Software Environment

The following table lists the hardware (EVK) and software (BSP/VLP/SDK) needed to use this module.

Table 2-1 Hardware and Software Environment

Index	EVK	BSP/VLP/SDK	Remarks
1	RZ/V2H Evaluation Board Kit	RZ/V2H AI SDK v5.20 (or later) *Kernel 5.10-CIP, Yocto-3.1 (Dunfell)*	-
		RZ/V2H AI SDK v6.00 (or later) *Kernel 6.1-CIP, Yocto-5.0 (Scarthgap)*	
2	RZ/V2N Evaluation Board Kit v1.0 and v2.0[1]	RZ/V2N AI SDK v5.00 (or later) *Kernel 5.10-CIP, Yocto-3.1 (Dunfell)*	-
		RZ/V2N AI SDK v6.00 (or later) *Kernel 6.1-CIP, Yocto-5.0 (Scarthgap)*	

1. See https://www.renesas.com/design-resources/boards-kits/rz-v2n-evk.

2.2 Software Requirement

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

Table 2-2 Software Requirement

Index	Name	Remarks
1	RZ/V2H AI SDK RZ/V2N AI SDK	• Memory Manager MMNGR • UVCS Driver • Audio Codec (`faac` / `faad`) • Video Signal Processor Manager VSPM • Wayland/Weston • Video Codec Library (OpenMAX)[1]

1. In some older revisions of the AI SDK, the "Video Codec Library (OpenMAX)" (previously written as "OMX (Video)") was provided as a separate package "RZ MPU Video Codec Library".

2.3 Module Configuration

This section shows the software configuration in which GStreamer is used.

2.3.1 Video Decode and Display

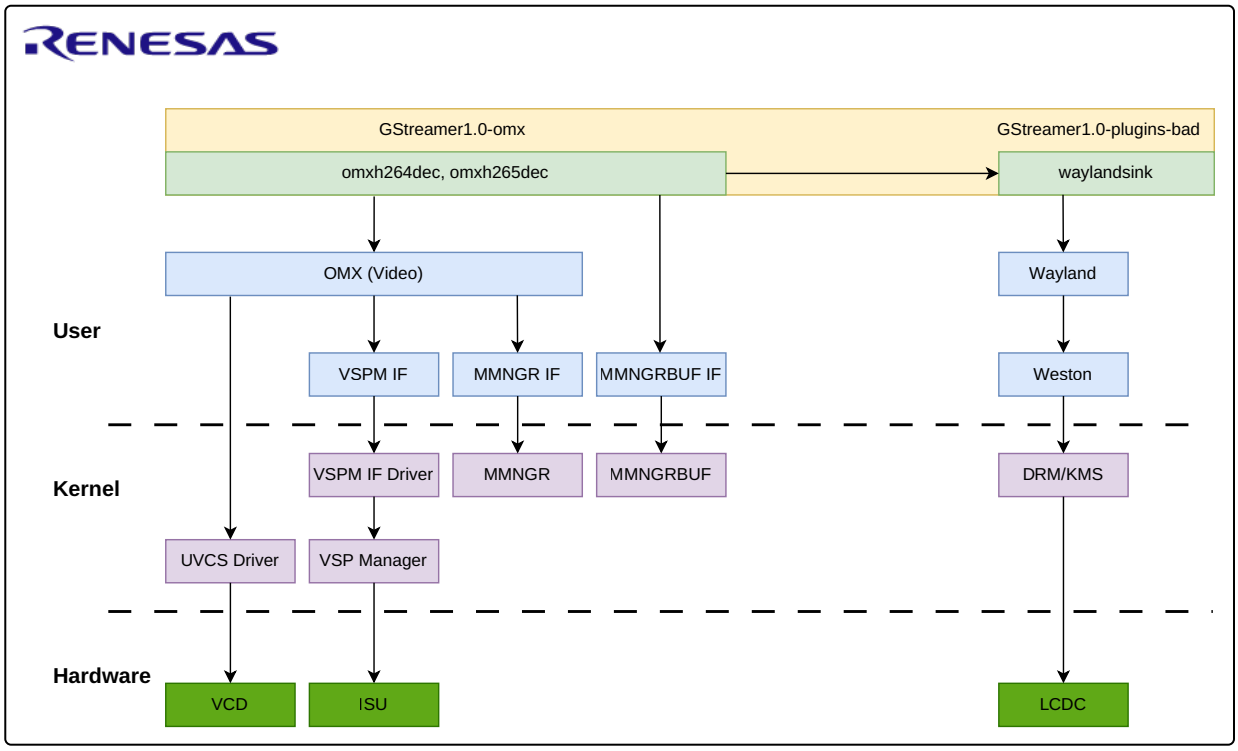


Figure 2-1: Video Playback

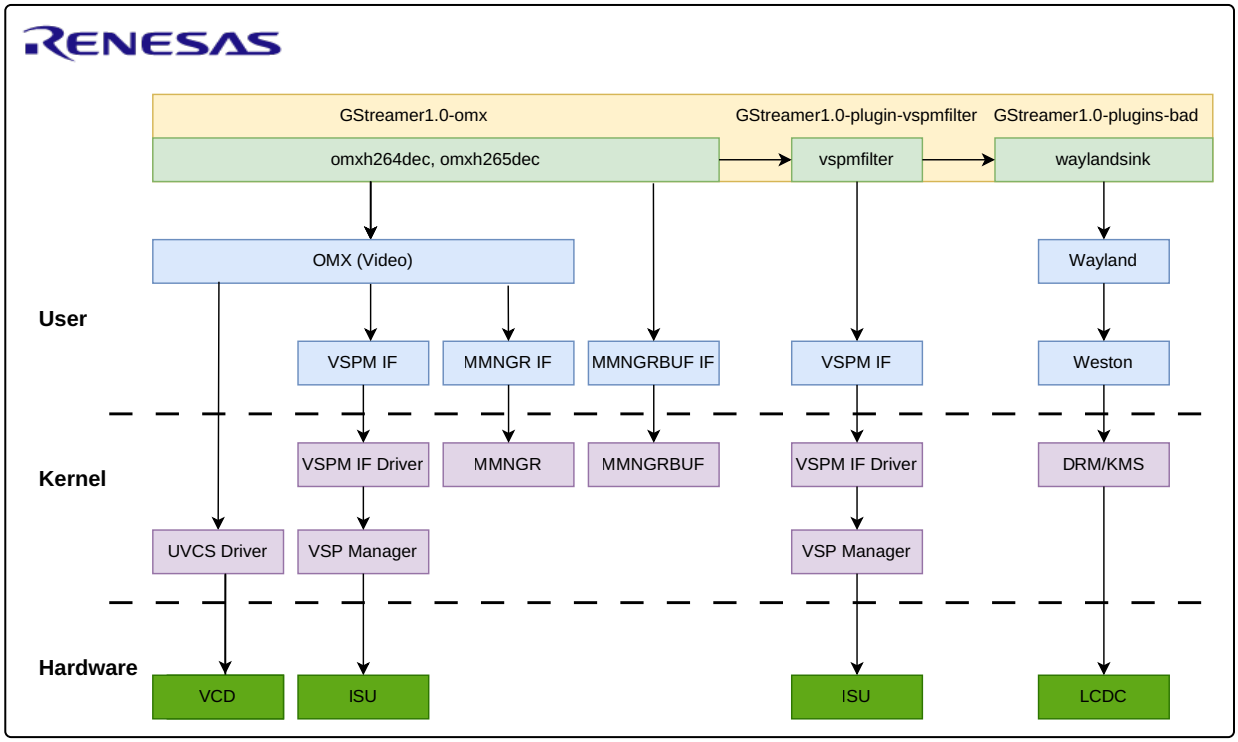


Figure 2-2: Video Playback (scaling and format conversion)

2.3.2 Video Capture and Display

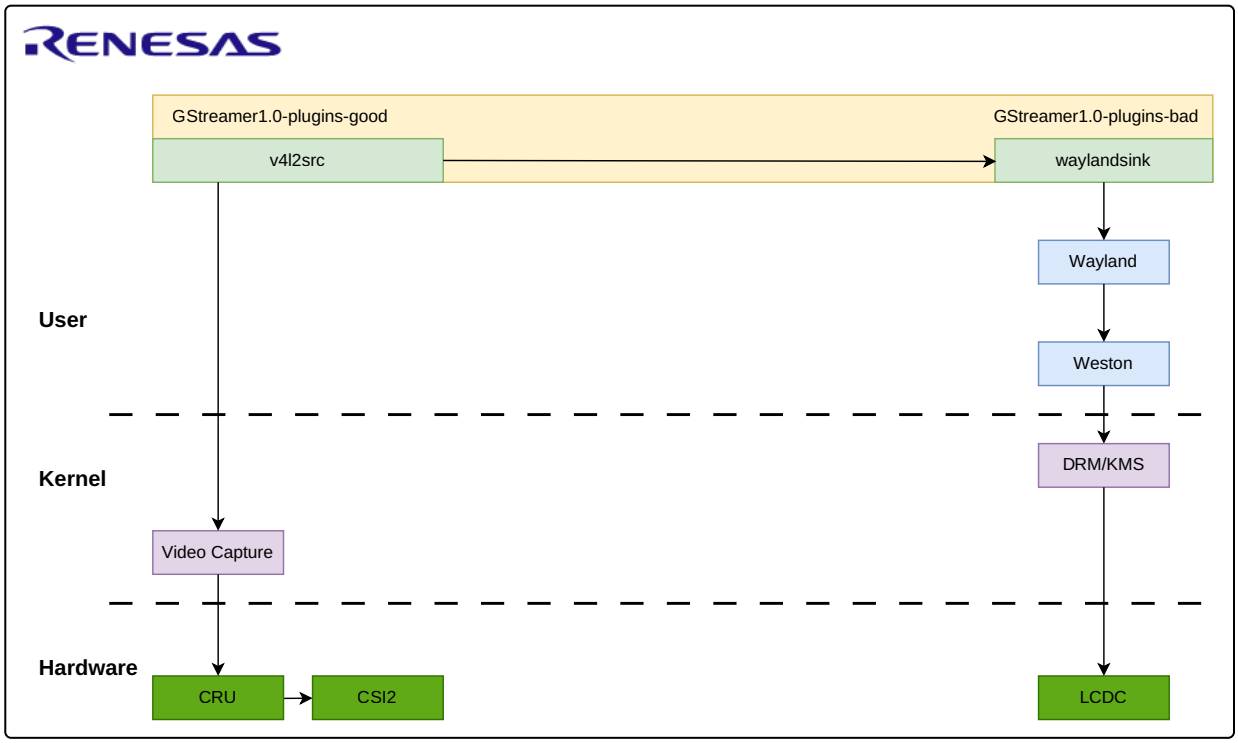


Figure 2-3: Video Capture and Display

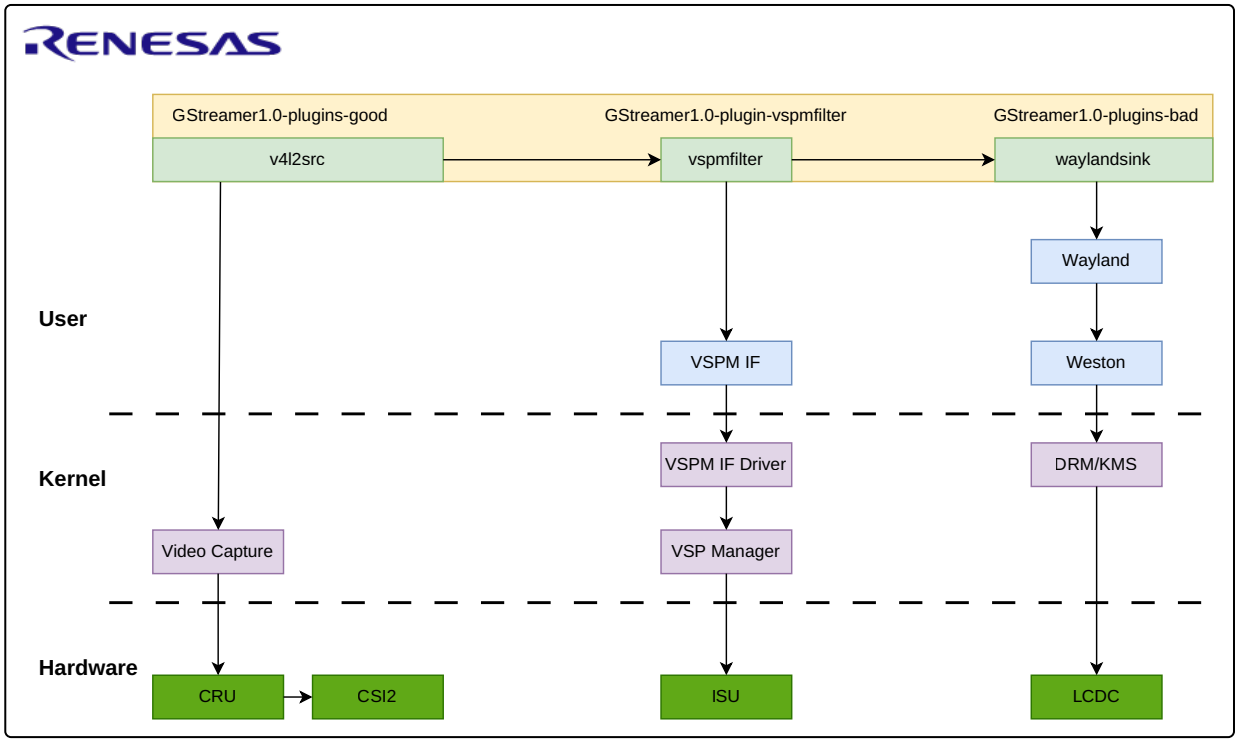


Figure 2-4: Video Capture and Display (scaling and format conversion)

2.3.3 Video Capture and Encode

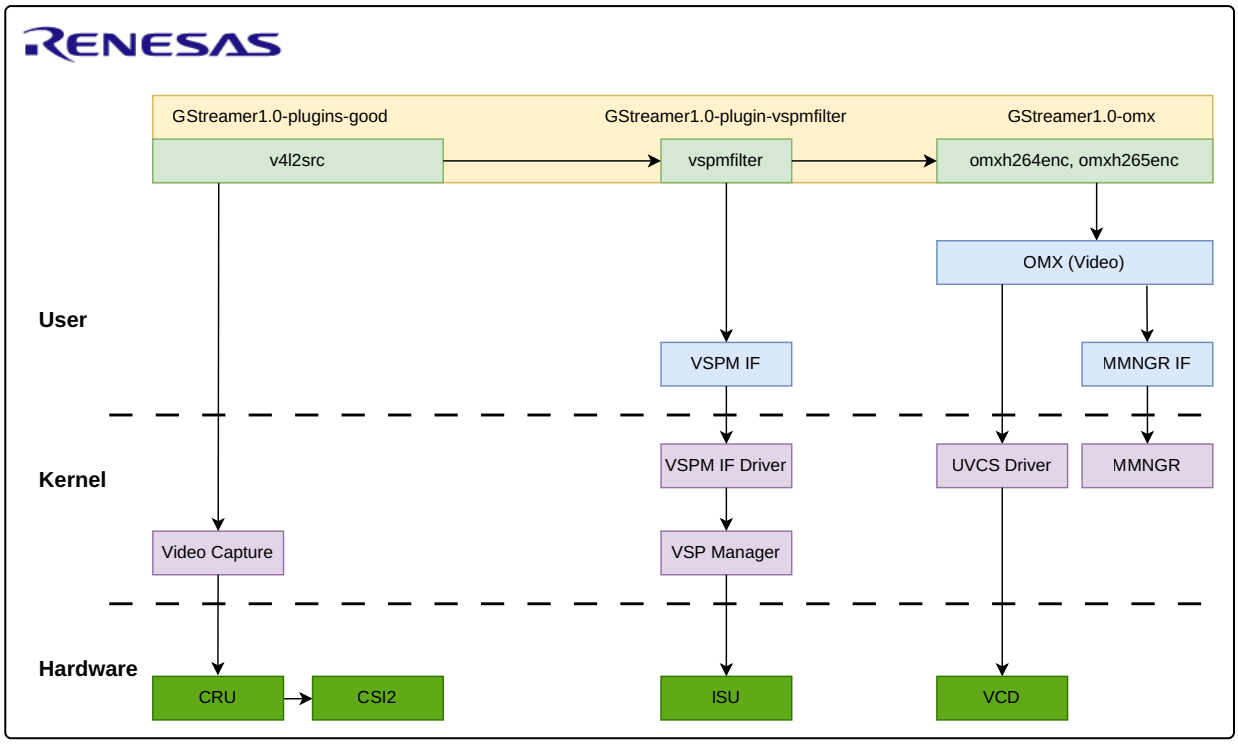


Figure 2-5: Video Capture and Encode

3. GStreamer Tool, Plug-in and Element

3.1 GStreamer Tool

3.1.1 `gst-launch-1.0`

`gst-launch-1.0` - build and run a GStreamer pipeline.

This tool accepts a textual description of a pipeline, instantiates it, and sets it to the PLAYING state. It allows you to quickly check if a given pipeline works, before going through the actual implementation using GStreamer API calls.

```
#    gst-launch-1.0 [OPTIONS] PIPELINE-DESCRIPTION
```

3.1.2 `gst-inspect-1.0`

`gst-inspect-1.0` - print info about a GStreamer plugin or element.

`gst-inspect-1.0` is a tool that prints out information on available GStreamer plugins, information about a particular plugin, or information about a particular element.

When executed with no `PLUGIN` or `ELEMENT` argument, `gst-inspect-1.0` will print a list of all plugins and elements together with a summary.

```
#    gst-inspect-1.0
```

When executed with a `PLUGIN` or `ELEMENT` argument, `gst-inspect-1.0` will print information about that plug-in or element.

```
#    gst-inspect-1.0 omxh264dec
```

3.1.3 `gst-discoverer-1.0`

`gst-discoverer-1.0` - Display file metadata and stream information.

`gst-discoverer-1.0` is a tool that can be used to print basic metadata and stream information about a media file. It can be run on individual files or whole directories. It will recurse into sub-directories in that case.

```
#    gst-discoverer-1.0 MP4_file.mp4
```

3.2 GStreamer Plug-in and Element

Table 3-1 GStreamer and GStreamer Plugins

Index	Name	Yocto Version	License	Version	Remarks
1	GStreamer core plugins	Dunfell	GPL-2.0	1.16.3	Downloaded by Yocto [1]
		Scarthgap	LGPL-2.1	1.22.12	
2	GStreamer Base plugins	Dunfell	GPL-2.0	1.16.3	Downloaded by Yocto [1]
		Scarthgap	LGPL-2.1	1.22.12	
3	GStreamer Good plugins (Include change for `v4l2src`)	Dunfell	LGPL-2.1	1.16.3	Downloaded by Yocto [1]
		Scarthgap		1.22.12	
4	GStreamer Bad plugins (Include change for `waylandsink`, `bayersink`, `bayerconvert`)	Dunfell	GPL-2.0	1.16.3	Downloaded by Yocto [1]
		Scarthgap	LGPL-2.1	1.22.12	
5	GStreamer Ugly plugins	Dunfell	LGPL-2.1	1.16.3	Downloaded by Yocto
		Scarthgap		1.22.12	
6	GStreamer gst-omx plugins (Include change for `omxh264dec`, `omxh265dec`, `omxh264enc`, `omxh265enc`)	Dunfell	LGPL-2.1	1.16.3	Downloaded by Yocto [1]
		Scarthgap		1.22.12	
7	gst-plugin-vspmfilter (Include `vspmfilter`)	Dunfell	GPL-2.0	-	Downloaded by Yocto [2]
		Scarthgap			

1. URL: https://github.com/renesas-rcar for AI SDK v5.20 and older.
   URL: https://github.com/renesas-rz for AI SDK v6.00 and later.
2. URL: https://github.com/renesas-rz/rzg_gstreamer_vspmfilter/tree/rz_g2l.

3.2.1 `playbin` and `playbin3`

`playbin` and `playbin3` provide a standalone, all-in-one abstraction for audio and/or video playback. They can handle both audio and video streams as well as related features.

- Automatic file type recognition and based on that automatic selection and usage of the right audio/video/subtitle demuxers/decoders.
- Visualisations for audio files.
- Subtitle support for video files. Subtitles can be stored in external files.
- Stream selection between different video/audio/subtitles streams.
- Meta info (tag) extraction.
- Easy access to the last video sample.
- Buffering when playing streams over a network.
- Volume control with mute option.

Table 3-2 `playbin` and `playbin3`

Element Name	Package	Description
`playbin`	`gst-plugins-base`	Various playback elements
`playbin3`	`gst-plugins-base`	Various playback elements

NOTE

The default video filter of the `playbin` and `playbin3` for RZ/V2H and RZ/V2N AI SDK(s) is set by `/etc/gstpbfilter.conf`, for changing default video filter please see How to Change the Default Video Filter In `playbin` and `playbin3` Element.

3.2.2 Video Decode Plugin

Video decoders are utilized to convert encoded video into raw video and establish a connection with a destination sink, such as a display or file storage.

Table 3-3 Video Decode Plugins

Element Name	Package	Description
`decodebin`	`gst-plugins-base`	Autoplug and decode video
`avdec_h264`	GStreamer FFmpeg Plugins	libav H.264 decoder
`avdec_h265`	GStreamer FFmpeg Plugins	libav H.265 decoder
`omxh264dec`	`gst-omx`	OpenMAX H.264 Video Decoder
`omxh265dec`	`gst-omx`	OpenMAX H.265 Video Decoder

CAUTION

In AI SDK v6.00 or later, `avdec_h264` and `avdec_h265` are no longer included in the default installation. Users who wish to use `avdec_h264` and/or `avdec_h265` must install them manually and assume all associated risks and responsibilities.

3.2.3 Video Capture Plugin

Table 3-4 Video Capture Plugin

Element Name	Package	Description
`v4l2src`	`gst-plugins-good`	Reads frames from a Video4Linux2 device

3.2.4 Video Conversion Plugin

Table 3-5 Video Conversion Plugins

Element Name	Package	Description
`videoconvert`	`gst-plugins-base`	Converts video from one colorspace to another
`videoscale`	`gst-plugins-base`	This element resizes video frames
`vspmfilter`	`gst-plugin-vspmfilter`	This is a GStreamer plugin that can use the color conversion and scaling down with the hardware acceleration on RZ/V2H and RZ/V2N MPU(s)

3.2.5 Video Encode Plugin

Table 3-6 Video Encode Plugins

Element Name	Package	Description
`omxh264enc`	`gst-omx`	OpenMAX H.264 Video Encoder
`omxh265enc`	`gst-omx`	OpenMAX H.265 Video Encoder

3.2.6 Video Sink Plugin

Table 3-7 Video Sink Plugins

Element Name	Package	Description
`autovideosink`	`gst-plugins-good`	A video sink that automatically detects an appropriate video sink to use
`fbdevsink`	`gst-plugins-bad`	Linux framebuffer videosink
`glimagesink`	`gst-plugins-base`	glimagesink renders video frames to a drawable on a local or remote display using OpenGL
`fpsdisplaysink`	`gst-plugins-bad`	Can display the current and average framerate as a `testoverlay` or on `stdout`
`waylandsink`	`gst-plugins-bad`	The waylandsink is creating its own window and renders the decoded video frames to that

NOTE

RZ/V2H and RZ/V2N AI SDK(s) enable `waylandsink` as the highest rank video sink. This causes the `autovideosink`, `playbin` and/or `playbin3` to utilize the `waylandsink`. To use another video sink, you need to specify it with the `video-sink=<video_sink_element>` options.

3.2.7 Demux Plugin

Table 3-8 Demux Plugins

Element Name	Package	Description
`qtdemux`	`gst-plugins-good`	Demuxes a .mov/mp4 file into raw or compressed audio and/or video streams
`matroskademux`	`gst-plugins-good`	Demuxes a Matroska file into the different contained streams
`flvdemux`	`gst-plugins-good`	Demuxes a FLV file into the different contained streams
`avidemux`	`gst-plugins-good`	Demuxes an AVI file into raw or compressed audio and/or video streams

NOTE

As a limitation, `qtdemux` element in GStreamer of the RZ/V2N AI SDK v6.00 cannot be used in combination with `omxh265dec`. Please use other demux plugins.

3.2.8 Mux Plugin

Table 3-9 Mux Plugins

Element Name	Package	Description
`qtmux`	`gst-plugins-good`	Merges streams (audio and video) into QuickTime(.mov) file
`matroskamux`	`gst-plugins-good`	Muxes different input streams into a Matroska file
`flvmux`	`gst-plugins-good`	Muxes different streams into a FLV file
`avimux`	`gst-plugins-good`	Muxes raw or compressed audio and/or video streams into an AVI file
`mp4mux`	`gst-plugins-good`	Merges streams (audio and video) into ISO MPEG-4 (.mp4) file

NOTE

As a limitation, `qtmux` element in GStreamer of the RZ/V2N AI SDK v6.00 cannot be used in combination with `omxh265enc` to pack the H.265 encoded stream to Media Container file. Please store the H.265 encoded stream with other container.

3.2.9 Audio Plugin

Audio plugins are used to convert or translate different formats of audio data such as wav, ogg, aac, mp3, etc.

Table 3-10 Audio Plugins

Element Name	Package	Description
`mpg123audiodec`	`gst-plugins-good`	Audio decoder for MPEG-1 layer 1/2/3 audio data
`vorbisdec`	`gst-plugins-base`	Decodes a Vorbis stream to raw float audio
`vorbisenc`	`gst-plugins-base`	Encodes raw float audio into a Vorbis stream
`faac`	`gst-plugins-bad`	Encodes raw audio to AAC (MPEG-4 part 3) streams
`faad`	`gst-plugins-bad`	Decodes AAC (MPEG-4 part 3) stream
`alsasink`	`gst-plugins-base`	Renders audio samples using the ALSA audio API

NOTE To utilize an audio device, refer to Section How To Configure and Utilize Audio.

3.2.10 Network Protocol Plugin

Table 3-11 Network Protocol Plugins

Element Name	Package	Description
`udpsink`	`gst-plugins-good`	`udpsink` is a network sink that sends UDP packets to the network. It can be combined with RTP payloaders to implement RTP streaming
`multiudpsink`	`gst-plugins-good`	`multiudpsink` is a network sink that sends UDP packets to multiple clients
`udpsrc`	`gst-plugins-good`	`udpsrc` is a network source that reads UDP packets from the network
`tcpserversink`	`gst-plugins-base`	A sink element that sends data as a server over the network via TCP
`tcpclientsrc`	`gst-plugins-base`	A source element that receives data as a client over the network via TCP
`tcpclientsink`	`gst-plugins-base`	A sink element that send data as a client over the network via TCP
`tcpserversrc`	`gst-plugins-base`	A source element that receive data as a server over the network via TCP

3.2.11 Payload/Depayload Plugin

Table 3-12 Payload/Depayload Plugins

Element Name	Package	Description
`rtph264pay`	`gst-plugins-good`	Payload-encode H.264 video into RTP packets
`rtph264depay`	`gst-plugins-good`	Extracts H.264 video from RTP packets
`rtpmpapay`	`gst-plugins-good`	Payload MPEG audio as RTP packets
`rtpmpadepay`	`gst-plugins-good`	Extracts MPEG audio from RTP packets

3.3 Renesas Extended Element

The main elements customized by Renesas are explained below. The properties added by Renesas are explained as “Expansion Properties”.

3.3.1 Video H.264 Decode Element: `omxh264dec`

Table 3-13 `omxh264dec` Pad Templates

SINK	SRC
Capabilities: video/x-h264 alignment: Au stream-format: byte-stream width: [224, 1920] height: [96, 1080]	Capabilities: video/x-raw format: { NV12, NV16, YUY2, UYVY, RGB16, RGB, BGR, ARGB, ABGR, RGBA, BGRA, GRAY8 } width: [1, 1920] height: [1, 1080]

Table 3-14 `omxh264dec` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="4"><code>omxh264dec</code></td><td rowspan="4">• Support NV12, NV16, YUY2, UYVY, RGB16, RGB, BGR, ARGB, RGB16, RGB, BGR, ARGB as the output formats
• Support video downscaling[1] and cropping (if video has crop information[2])
• Supported resolution:
  • sink: 224x96 to 1920x1080
  • src: 1x1 to 1920x1080</td><td><code>no-copy</code> [3]</td><td>Do not copy output data in gst-omx
Share output buffer with element of downstream
Export userptr to downstream for buffer sharing
<table><tr><td>TRUE</td><td>Copyless</td></tr><tr><td>FALSE (default[4])</td><td>Copy (general purpose)</td></tr></table></td></tr>
<tr><td><code>use-dmabuf</code> [3]</td><td>Do not copy output data in gst-omx
Use DMA buffer sharing framework (DMABUF)
Export DMABUF to downstream for buffer sharing
<table><tr><td>TRUE (default[4])</td><td>Use DMABUF</td></tr><tr><td>FALSE</td><td>Does not use DMABUF (Same as <code>no-copy=FALSE</code>)</td></tr></table></td></tr>
<tr><td><code>sizebytes</code></td><td>Change the size of allocation buffer for the input port. (0 - 4294967295)
Default: 0</td></tr>
<tr><td><code>num-outbufs</code> [5]</td><td>Number of buffers that are required on output port. (0 - 32)
Default: 4</td></tr>
</table>

1. The maximum supported downscale factor is less than 16.
2. Crop information can be set via the omxh264enc crop property (see Video H.264 Encode Element).
3. Select one, either `no-copy` or `use-dmabuf` option. When you specify neither `no-copy` nor `use-dmabuf` option, `omxh264dec` works as `use-dmabuf=TRUE` .
4. The default value can be changed by updating gstomx.conf (see How to Change the Default Mode In OMX Video Decoder Element).
5. If `num-outbufs` is set smaller than the default value, it will be round up to default value.

NOTE

Renesas `omxh264dec` decoder with RZ/V2H AI SDK v5.20 (or older) and/or RZ/V2N AI SDK v6.00 (or older) only supports decoding 1 video stream (decoding 2 or more video simultaneously is not supported regardless pipelines number).

Example:
Using 2 `omxh264dec` on separate GStreamer pipelines is not supported.

NOTE

Renesas `omxh264dec` for RZ/V2H cannot be used on same time with OpenCV Accelerator.
Renesas `omxh264dec` for RZ/V2N cannot be used on same time with DRP-AI TVM and/or OpenCV Accelerator.

3.3.2 Video H.265 Decode Element: `omxh265dec`

Table 3-15 `omxh265dec` Pad Templates

SINK	SRC
Capabilities: video/x-h265 alignment: Au stream-format: byte-stream width: [256, 4096] height: [96, 4096]	Capabilities: video/x-raw format: { NV12, NV16, YUY2, UYVY, RGB16, RGB, BGR, ARGB, ABGR, RGBA, BGRA, GRAY8 } width: [1, 4096] height: [1, 4096]

Table 3-16 `omxh265dec` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="4"><code>omxh265dec</code></td><td rowspan="4">• Support NV12, NV16, YUY2, UYVY, RGB16, RGB, BGR, ARGB, RGB16, RGB, BGR, ARGB as the output formats
• Support video downscaling[1] and cropping (if video has crop information[2])
• Supported resolution:
  • sink: 256x96 to 4096x4096
  • src: 1x1 to 4096x4096</td><td><code>no-copy</code> [3]</td><td>Do not copy output data in gst-omx
Share output buffer with element of downstream
Export userptr to downstream for buffer sharing
<table><tr><td>TRUE</td><td>Copyless</td></tr><tr><td>FALSE (default[4])</td><td>Copy (general purpose)</td></tr></table></td></tr>
<tr><td><code>use-dmabuf</code> [3]</td><td>Do not copy output data in gst-omx
Use DMA buffer sharing framework (DMABUF)
Export DMABUF to downstream for buffer sharing
<table><tr><td>TRUE (default[4])</td><td>Use DMABUF</td></tr><tr><td>FALSE</td><td>Does not use DMABUF (Same as <code>no-copy=FALSE</code>)</td></tr></table></td></tr>
<tr><td><code>sizebytes</code></td><td>Change the size of allocation buffer for the input port. (0 - 4294967295)
Default: 0</td></tr>
<tr><td><code>num-outbufs</code> [5]</td><td>Number of buffers that are required on output port. (0 - 32)
Default: 4</td></tr>
</table>

1. The maximum supported downscale factor is less than 16.
2. Crop information can be set via the omxh265enc crop property (see Video H.265 Encode Element).
3. Select one, either `no-copy` or `use-dmabuf` option. When you specify neither `no-copy` nor `use-dmabuf` option, `omxh265dec` works as `use-dmabuf=TRUE` .
4. The default value can be changed by updating gstomx.conf (see How to Change the Default Mode In OMX Video Decoder Element).
5. If `num-outbufs` is set smaller than the default value, it will be round up to default value.

NOTE

Renesas `omxh265dec` decoder with RZ/V2H AI SDK v5.20 (or older) and/or RZ/V2N AI SDK v6.00 (or older) only supports decoding 1 video stream (decoding 2 or more video simultaneously is not supported regardless pipelines number).

Example:
Using 2 `omxh265dec` on separate GStreamer pipelines is not supported.

NOTE

Renesas `omxh265dec` for RZ/V2H cannot be used on same time with OpenCV Accelerator.
Renesas `omxh265dec` for RZ/V2N cannot be used on same time with DRP-AI TVM and/or OpenCV Accelerator.

3.3.3 Video H.264 Encode Element: `omxh264enc`

Table 3-17 `omxh264enc` Pad Templates

SINK	SRC
Capabilities: video/x-raw format: { NV12 } width: [224, 1920] height: [96, 1920]	Capabilities: video/x-h264 stream-format: byte-stream width: [224, 1920] height: [96, 1920]

Table 3-18 `omxh264enc` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="8"><code>omxh264enc</code></td><td rowspan="8">• Support NV12 as the input format
• Supported resolutions:
• 224x96 to 1920x1080 (1080x1920 for vertical encoding)</td><td><code>no-copy</code> [1]</td><td>Do not copy input data in gst-omx
Share input buffer with element of upstream
Export userptr to upstream for buffer sharing
<table><tr><td>TRUE</td><td>Copyless</td></tr><tr><td>FALSE (default)</td><td>Copy (general purpose)</td></tr></table></td></tr>
<tr><td><code>use-dmabuf</code> [1]</td><td>Do not copy input data in gst-omx
Use DMA buffer sharing framework (DMABUF)
Import DMABUF from upstream for buffer sharing
<table><tr><td>TRUE</td><td>Use DMABUF</td></tr><tr><td>FALSE (default)</td><td>Does not use DMABUF
(Same as <code>no-copy=FALSE</code>)</td></tr></table></td></tr>
<tr><td><code>scan-type</code></td><td>Set encode scan type
<table><tr><td>0</td><td>Progressive</td></tr><tr><td>4</td><td>Interlaced Top Field First</td></tr><tr><td>5</td><td>Interlaced Bottom Field First</td></tr><tr><td>-1</td><td>OMX default (Progressive)</td></tr></table></td></tr>
<tr><td><code>crop</code> [2]</td><td>Set output crop information. The encoder does not perform cropping as pre-processing. The crop information is added to SPS header of output video
<code>crop=crop_left:crop_right:crop_top:crop_bottom</code>
String. Default: "0:0:0:0"</td></tr>
<tr><td><code>send-eos</code> [3]</td><td>Send EOS/EOF stream data downstream</td></tr>
<tr><td><code>ref-frames</code> [3]</td><td>Number of reference frames</td></tr>
<tr><td><code>b-frames</code></td><td>Specifies the number of B-frames inserted between each pair of reference frames (i.e., between P–P or I–P frames). (0-3)
Default: 0</td></tr>
<tr><td><code>sizebytes</code></td><td>Change the size of allocation buffer for the output port. (0 - 4294967295)
Default: 0</td></tr>
</table>

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="4"></td><td rowspan="4"></td><td>interval_intraframes [4]</td><td>The sum of P-frames between 2 I-frames and b-frames setting. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
<tr><td>control-rate</td><td>Bitrate control method
<table><tr><td>(0): disable</td><td>Disable</td></tr><tr><td>(1): variable</td><td>Variable</td></tr><tr><td>(2): constant</td><td>Constant</td></tr><tr><td>(3): variable-skip-frame</td><td>Variable Skip Frames</td></tr><tr><td>(4): constant-skip-frames</td><td>Constant Skip Frames</td></tr><tr><td>(-1): default</td><td>Component Default</td></tr></table></td></tr>
<tr><td>target-bitrate</td><td>Target bitrate in bits per second. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
<tr><td>periodicity-idr</td><td>Periodicity of IDR frames. Specifies how many GOPs are encoded between consecutive IDR frames. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
</table>

1. Select one, either `no-copy` or `use-dmabuf` option. When you specify neither `no-copy` nor `use-dmabuf` option, `omxh264enc` works as `use-dmabuf=TRUE`.
2. To set crop information in SPS header of output video, you need to add option `crop=crop_left:crop_right:crop_top:crop_bottom` to the pipeline. These parameters must satisfy the following conditions:
   - `crop_left`, `crop_right`, `crop_top`, `crop_bottom`, must be even numbers greater than or equal to 0;
   - The sum of `crop_left` and `crop_right` must be less that the width of video size;
   - The sum of `crop_top` and `crop_bottom` must be less that the height of video size.

   For example pipeline, please see Video Encode Pipeline.
3. This property is not supported.
4. When `interval_intraframes` is set to 255 and `b-frames` is 0, the frame number in the GOP becomes unlimited. In this case, the first frame is encoded as an IDR frame, and all subsequent frames are encoded as P-frames.

NOTE

When using the Renesas `omxh264enc` encoder with RZ/V2N AI SDK v6.00 or RZ/V2H AI SDK v5.20 or RZ/V2N AI SDK v5.00 (or older versions), and the input source is `videotestsrc`, the height of the input resolution must be a multiple of 32.

Example:
Full HD (FHD) resolution is typically 1920x1080. However, for this encoder, you need to adjust the height to be a multiple of 32. Therefore, instead of 1920x1080, you should use 1920x1088.

3.3.4 Video H.265 Encode Element: `omxh265enc`

Table 3-19 `omxh265enc` Pad Templates

SINK	SRC
Capabilities: video/x-raw format: { NV12 } width: [256, 4096] height: [96, 4096]	Capabilities: video/x-h265 stream-format: byte-stream width: [256, 4096] height: [96, 4096]

Table 3-20 `omxh265enc` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="7"><code>omxh265enc</code></td><td rowspan="7">• Support NV12 as the input format
• Supported resolutions:
• 256x96 to 3840x2160 (2160x3840 for vertical encoding)</td><td><code>no-copy</code> [1]</td><td>Do not copy input data in gst-omx
Share input buffer with element of upstream
Export userptr to upstream for buffer sharing
<table><tr><td>TRUE</td><td>Copyless</td></tr><tr><td>FALSE (default)</td><td>Copy (general purpose)</td></tr></table></td></tr>
<tr><td><code>use-dmabuf</code> [1]</td><td>Do not copy input data in gst-omx
Use DMA buffer sharing framework (DMABUF)
Import DMABUF from upstream for buffer sharing
<table><tr><td>TRUE</td><td>Use DMABUF</td></tr><tr><td>FALSE (default)</td><td>Does not use DMABUF
(Same as <code>no-copy=FALSE</code>)</td></tr></table></td></tr>
<tr><td><code>scan-type</code></td><td>Set encode scan type
<table><tr><td>0</td><td>Progressive</td></tr><tr><td>4</td><td>Interlaced Top Field First</td></tr><tr><td>5</td><td>Interlaced Bottom Field First</td></tr><tr><td>-1</td><td>OMX default (Progressive)</td></tr></table></td></tr>
<tr><td><code>crop</code> [2]</td><td>Set output crop information. The encoder does not perform cropping as pre-processing. The crop information is added to SPS header of output video
<code>crop=crop_left:crop_right:crop_top:crop_bottom</code>
String. Default: "0:0:0:0"</td></tr>
<tr><td><code>b-frames</code></td><td>Specifies the number of B-frames inserted between each pair of reference frames (i.e., between P–P or I–P frames). (0-3)
Default: 0</td></tr>
<tr><td><code>sizebytes</code></td><td>Change the size of allocation buffer for the output port. (0 - 4294967295)
Default: 0</td></tr>
<tr><td><code>interval_intraframes</code> [3]</td><td>The sum of P-frames between 2 I-frames and b-frames setting. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
</table>

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="3"></td><td rowspan="3"></td><td><code>control-rate</code></td><td>Bitrate control method
<table><tr><td>(0): disable</td><td>Disable</td></tr><tr><td>(1): variable</td><td>Variable</td></tr><tr><td>(2): constant</td><td>Constant</td></tr><tr><td>(3): variable-skip-frame</td><td>Variable Skip Frames</td></tr><tr><td>(4): constant-skip-frames</td><td>Constant Skip Frames</td></tr><tr><td>(-1): default</td><td>Component Default</td></tr></table></td></tr>
<tr><td><code>target-bitrate</code></td><td>Target bitrate in bits per second. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
<tr><td><code>periodicity-idr</code></td><td>Periodicity of IDR frames. Specifies how many GOPs are encoded between consecutive IDR frames. (0 - 4294967295)
Default: 4294967295 (use OMX encoder default setting)</td></tr>
</table>

1. Select one, either `no-copy` or `use-dmabuf` option. When you specify neither `no-copy` nor `use-dmabuf` option, `omxh265enc` works as `use-dmabuf=TRUE`.
2. To set crop information in SPS header of output video, you need to add option `crop=crop_left:crop_right:crop_top:crop_bottom` to the pipeline. These parameters must satisfy the following conditions:
   - `crop_left`, `crop_right`, `crop_top`, `crop_bottom`, must be even numbers greater than or equal to 0;
   - The sum of `crop_left` and `crop_right` must be less that the width of video size;
   - The sum of `crop_top` and `crop_bottom` must be less that the height of video size.

   For example pipeline, please see Video Encode Pipeline.
3. When `interval_intraframes` is set to 255 and `b-frames` is 0, the frame number in the GOP becomes unlimited. In this case, the first frame is encoded as an IDR frame, and all subsequent frames are encoded as P-frames.

NOTE

When using the Renesas `omxh265enc` encoder with RZ/V2N AI SDK v6.00 or RZ/V2H AI SDK v5.20 or RZ/V2N AI SDK v5.00 (or older versions), and the input source is `videotestsrc`, the height of the input resolution must be a multiple of 32.

Example:
Full HD (FHD) resolution is typically 1920x1080. However, for this encoder, you need to adjust the height to be a multiple of 32. Therefore, instead of 1920x1080, you should use 1920x1088.

3.3.5 Source Element: `v4l2src`

Table 3-21 `v4l2src` Pad Templates

SRC
Capabilities: video/x-raw format: { BGR, GRAY8, YUY2, UYVY, NV16, ARGB, BGRA, BGRx } width: [80, 1920] height: [80, 1080] framerate: [0/1, 60/1]

Table 3-22 `v4l2src` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th rowspan="2">Element Explanation</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="4"><code>v4l2src</code></td><td rowspan="4">• <code>v4l2src</code> can be used to capture video from v4l2 devices, like webcams and tv cards
• Support BGR, GRAY8, YUY2, UYVY, NV16, ARGB, BGRA, BGRx as the output formats</td><td><code>io-mode</code></td><td>This property is standard implementation
The available settings are as follows
<table><tr><td>auto (default)</td><td>Auto</td></tr><tr><td><code>mmap</code> [1]</td><td>Use mmap</td></tr><tr><td><code>dmabuf</code> [2]</td><td>Use dmabuf</td></tr><tr><td><code>userptr</code> [3]</td><td>Use userptr</td></tr><tr><td><code>dmabuf-import</code></td><td>Use dmabuf-import</td></tr><tr><td><code>rw</code></td><td>Use rw</td></tr></table></td></tr>
<tr><td><code>no-resurect-buf</code> [4]</td><td>This property is that skip resurrect buffer when all buffers in queue used up
Please set true when sharing buffers with downstream such as encoder
<table><tr><td>TRUE</td><td>Enable</td></tr><tr><td>FALSE (default)</td><td>Disable</td></tr></table></td></tr>
<tr><td><code>num-alloc-buffer</code></td><td>Change number of buffers in <code>v4l2src</code>
It is used for debugging purpose</td></tr>
<tr><td><code>disable-dynamic-buffer-alloc</code></td><td>Flag for handling dynamic buffer allocation
<table><tr><td>TRUE</td><td>Enable</td></tr><tr><td>FALSE (default)</td><td>Disable</td></tr></table></td></tr>
</table>

1. If device does not support DMA-BUF, it selects `mmap` mode. This mode is the preferred mode for most applications that require fast data processing and minimal latency.
2. If device supports DMA-BUF, it selects `dmabuf` mode. This mode is highly efficient for systems that support DMA and can significantly reduce CPU load and increase throughput.
3. This mode is unsupported.
4. `no-resurect-buf` is unused and deprecated option in AI-SDK v6.00 (or later). Please use `disable-dynamic-buffer-alloc` instead.

NOTE

To identify active cameras and obtain their details, refer to Section How to Detect Active Camera While USB cameras can be directly integrated with a GStreamer pipeline, MIPI cameras necessitate the preliminary configuration of the CRU/CSI2 driver. For more details, please refer to the manual of the CRU/CSI2 driver. Example for e-CAM22_CURZH MIPI camera is at Section How to Configure and Utilize MIPI Camera.

3.3.6 Video Conversion Element: `vspmfilter`

Table 3-23 `vspmfilter` Pad Templates

SINK	SRC
Capabilities: video/x-raw format: { NV12, RGB16, RGB, BGR, RGBx, BGRx, xRGB, xBGR, BGRA, ARGB, ABGR, UYVY, YUY2, NV16, GRAY10_LE64, GRAY8 } width: [1, 4096] height: [1, 4096]	Capabilities: video/x-raw format: { NV12, RGB16, RGB, BGR, RGBx, BGRx, xRGB, xBGR, BGRA, ARGB, ABGR, UYVY, YUY2, NV16, GRAY10_LE64, GRAY8 } width: [1, 4096] height: [1, 4096]

Table 3-24 `vspmfilter` Expansion Properties

Element Name	Element Explanation	Expansion Properties	
		Property	Property Explanation
`vspmfilter`	• `vspmfilter` controls Colorspace conversion and Video down scaling with VSPM • Support NV12, RGB16, RGB, BGR, RGBx, BGRx, xRGB, xBGR, RGBA, BGRA, ARGB, ABGR, UYVY, YUY2, NV16, GRAY8, GRAY10_LE64 as the input format • Support NV12, RGB16, RGB, BGR, RGBx, BGRx, xRGB, xBGR, RGBA, BGRA, ARGB, ABGR, UYVY, YUY2, NV16, GRAY8, GRAY10_LE64 as the output format	`outbuf-alloc`	Whether or not to use `dmabuf` for output buffer TRUE: vspmfilter internally allocates memory for the output buffers FALSE (default): downstream element allocates memory for output buffers
		`dmabuf-use`	Whether or not to use `dmabuf` for output buffer TRUE: `vspmfilter` allocates DMA-BUF (Direct Memory Access Buffer Sharing Framework) for the output buffers FALSE (default): downstream element allocates memory for output buffers

NOTE

GRAY10_LE64 is Renesas' own proprietary format and is supported starting from AI SDK v6.00 (Codec 4.3.3.1).

GRAY10_LE64 packs 6 pairs of 10 bits (GRAY10) with an additional 4 bits as padding.

NOTE

To use `vspmfilter` with USB cameras, you need to add option `allocators=1` to `uvcvideo` kernel.

```
#   rmmod uvcvideo
#   insmod /lib/modules/*/kernel/drivers/media/usb/uvc/uvcvideo.ko allocators=1
```

3.3.7 Video Sink Element: `waylandsink`

Table 3-25 `waylandsink` Pad Templates

SINK
Capabilities: video/x-raw format: { BGRx, BGRA, RGBA, RGB16, YUY2, NV12, I420 } width: [1, 4096] height: [1, 4096]

Table 3-26 `waylandsink` Expansion Properties

<table>
<tr><th rowspan="2">Element Name</th><th colspan="2">Expansion Properties</th></tr>
<tr><th>Property</th><th>Property Explanation</th></tr>
<tr><td rowspan="8"><code>waylandsink</code></td><td><code>use-subsurface</code> [1]</td><td>When disabled, a subsurface will not be created from an externally supplied surface
(e.g., needed for scan out when the application's surface is fullscreen)
<table><tr><td>TRUE (default)</td><td>Use subsurface</td></tr><tr><td>FALSE</td><td>NOT use subsurface</td></tr></table></td></tr>
<tr><td><code>suppress-interlace</code></td><td>Suppress the buffer flag of interlace
(e.g., Use if wayland / weston does not support the interlace flag)
<table><tr><td>TRUE (default)</td><td>Suppress the buffer flag of interlace</td></tr><tr><td>FALSE</td><td>Not suppress the buffer flag of interlace</td></tr></table></td></tr>
<tr><td><code>position-x</code></td><td>Wayland Position X value from the application
Range: 0 - 2147483647
Default: 0</td></tr>
<tr><td><code>position-y</code></td><td>Wayland Position Y value from the application
Range: 0 - 2147483647
Default: 0</td></tr>
<tr><td><code>fullscreen</code></td><td>Display fullscreen on top of weston bar
Position will be at center of screen
Default: false</td></tr>
<tr><td><code>out-h</code></td><td>Wayland Height size of application
Integer. Range: 0 - 2147483647
Default: 0</td></tr>
<tr><td><code>out-w</code></td><td>Wayland Width size of application
Integer. Range: 0 - 2147483647
Default: 0</td></tr>
</table>

1. `use-subsurface` is unused and deprecated option in AI SDK v6.00 (Codec v4.3.3.0) and later.

4. Example Pipeline

4.1 `playbin` and `playbin3`

> **NOTE**
>
> The default video filter of the `playbin` for RZ/V2H and RZ/V2N AI SDK(s) is set by `/etc/gstpbfilter.conf`, for changing default video filter please see How to Change the Default Video Filter In `playbin` and `playbin3` Element.

4.1.1 `playbin`

```
# gst-launch-1.0 playbin uri=file:/MP4_file.mp4
```

4.1.2 `playbin3`

```
# gst-launch-1.0 playbin3 uri=file:/MP4_file.mp4
```

4.2 Video Decode Pipeline

4.2.1 `decodebin`

1. `decodebin` H.264 file

```
#  gst-launch-1.0 filesrc location=H264_file.264 ! decodebin ! autovideosink
```

2. `decodebin` H.265 file

```
#  gst-launch-1.0 filesrc location=H265_file.265 ! decodebin ! autovideosink
```

3. `decodebin` MP4 file

```
#  gst-launch-1.0 filesrc location=MP4_file.mp4 ! decodebin ! autovideosink
```

4.2.2 `omxh264dec` (Hardware Accelerated Video Decoder)

1. MP4 container file decode

```
#  gst-launch-1.0 filesrc location=MP4_H264_file.mp4 ! qtdemux ! h264parse ! omxh264dec ! waylandsink
```

2. H.264 elementary stream file decode

```
#  gst-launch-1.0 filesrc location=H264_file.264 ! h264parse ! omxh264dec ! waylandsink
```

3. Multiple MP4 container files decode

```
#  gst-launch-1.0 filesrc location=MP4_file1.mp4 ! qtdemux ! h264parse ! \
   omxh264dec ! waylandsink & \
   gst-launch-1.0 filesrc location=MP4_file2.mp4 ! qtdemux ! h264parse ! \
   omxh264dec ! waylandsink &
```

4. Multiple MP4 container files decode with "max-lateness=-1 qos=FALSE"

```
#  gst-launch-1.0 filesrc location=MP4_file1.mp4 ! qtdemux ! h264parse ! \
   omxh264dec ! waylandsink max-lateness=-1 qos=FALSE & \
   gst-launch-1.0 filesrc location=MP4_file2.mp4 ! qtdemux ! h264parse ! \
   omxh264dec ! waylandsink max-lateness=-1 qos=FALSE &
```

> **NOTE** In order to prevent dropped buffers when running multiple streams, it is recommended to add the `max-lateness=-1` and `qos=FALSE` options after the `waylandsink`. This will ensure that the system is able to handle the incoming streams without any data loss due to buffer overflow.

5. Down scaling MP4 container file decode

```
#  gst-launch-1.0 filesrc location=MP4_FHD_file.mp4 ! qtdemux ! h264parse ! \
   omxh264dec ! video/x-raw,width=1280,height=720 ! waylandsink
```

6. Color conversion H.264 elementary stream file decode

```
# gst-launch-1.0 filesrc location=H264_file.264 ! h264parse ! \
omxh264dec ! video/x-raw,format=RGBA ! waylandsink
```

7. Decode a video file that contains crop information

```
# gst-launch-1.0 filesrc location=CROP_file.264 ! h264parse ! \
omxh264dec ! waylandsink
```

NOTE Crop information can be set via the omxh264enc crop property (see Video H.264 Encode Element).

4.2.3 `avdec_h264` (Software Decoder)

CAUTION In AI SDK v6.00 or later, `avdec_h264` is no longer included in the default installation. Users who wish to use `avdec_h264` must install it manually and assume all associated risks and responsibilities.

1. MP4 container file decode

```
# gst-launch-1.0 filesrc location=MP4_H264_file.mp4 ! qtdemux ! h264parse ! avdec_h264 ! waylandsink
```

2. H.264 elementary stream file decode

```
# gst-launch-1.0 filesrc location=H264_file.264 ! h264parse ! avdec_h264 ! waylandsink
```

4.2.4 `omxh265dec` (Hardware Accelerated Video Decoder)

1. MP4 container file decode

```
# gst-launch-1.0 filesrc location=MP4_H265_file.mp4 ! qtdemux ! h265parse ! omxh265dec ! waylandsink
```

2. H.265 elementary stream file decode

```
# gst-launch-1.0 filesrc location=H265_file.265 ! h265parse ! omxh265dec ! waylandsink
```

3. Multiple MP4 container files decode

```
# gst-launch-1.0 filesrc location=MP4_file1.mp4 ! qtdemux ! h265parse ! \
omxh265dec ! waylandsink & \
gst-launch-1.0 filesrc location=MP4_file2.mp4 ! qtdemux ! h265parse ! \
omxh265dec ! waylandsink &
```

4. Multiple MP4 container files decode with "max-lateness=-1 qos=FALSE"

```
#   gst-launch-1.0 filesrc location=MP4_file1.mp4 ! qtdemux ! h265parse ! \
    omxh265dec ! waylandsink max-lateness=-1 qos=FALSE & \
    gst-launch-1.0 filesrc location=MP4_file2.mp4 ! qtdemux ! h265parse ! \
    omxh265dec ! waylandsink max-lateness=-1 qos=FALSE &
```

NOTE

In order to prevent dropped buffers when running multiple streams, it is recommended to add the `max-lateness=-1` and `qos=FALSE` options after the `waylandsink`. This will ensure that the system is able to handle the incoming streams without any data loss due to buffer overflow.

5. Down scaling MP4 container file decode

```
#   gst-launch-1.0 filesrc location=MP4_FHD_file.mp4 ! qtdemux ! h265parse ! \
    omxh265dec ! video/x-raw,width=1280,height=720 ! waylandsink
```

6. Color conversion H.265 elementary stream file decode

```
#   gst-launch-1.0 filesrc location=H265_file.265 ! h265parse ! \
    omxh265dec ! video/x-raw,format=RGBA ! waylandsink
```

7. Decode a video file that contains crop information

```
#   gst-launch-1.0 filesrc location=CROP_file.265 ! h265parse ! \
    omxh265dec ! waylandsink
```

NOTE

Crop information can be set via the omxh265enc crop property (see Video H.265 Encode Element).

NOTE

As a limitation, `qtdemux` element in GStreamer of the RZ/V2N AI SDK v6.00 cannot be used in combination with `omxh265dec`. Please use other demux plugins.

4.2.5 `avdec_h265` (Software Decoder)

> **CAUTION**
>
> In AI SDK v6.00 or later, `avdec_h265` is no longer included in the default installation. Users who wish to use `avdec_h265` must install it manually and assume all associated risks and responsibilities.

1. MP4 container file decode

```
# gst-launch-1.0 filesrc location=MP4_H265_file.mp4 ! qtdemux ! h265parse ! avdec_h265 ! waylandsink
```

2. H.265 elementary stream file decode

```
# gst-launch-1.0 filesrc location=H265_file.265 ! h265parse ! avdec_h265 ! waylandsink
```

4.3 Video Conversion Pipeline

4.3.1 `vspmfilter`

`outbuf-alloc` and `dmabuf-use` are used in different cases to boost GStreamer pipeline performance.

1. Down scaling video capture

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 io-mode=mmap ! \
video/x-raw, width=1920, height=1080 ! vspmfilter dmabuf-use=TRUE ! \
video/x-raw, width=720, height=480 ! waylandsink
```

2. Format conversion video capture

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 io-mode=mmap ! \
video/x-raw, width=1920, height=1080, format=YUY2 ! vspmfilter dmabuf-use=TRUE ! \
video/x-raw, format=NV12 ! waylandsink
```

3. Down scaling and format conversion video capture

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 io-mode=mmap ! \
video/x-raw, width=1920, height=1080, format=YUY2 ! vspmfilter dmabuf-use=TRUE ! \
video/x-raw, width=720, height=480, format=NV12 ! waylandsink
```

4. Video convert and display

```
# gst-launch-1.0 filesrc location=FHD_H264_file.264 ! h264parse ! omxh264dec ! \
video/x-raw, width=1920, height=1080 ! vspmfilter dmabuf-use=TRUE ! video/x-raw, \
format=BGRA, width=1280, height=720 ! waylandsink
```

4.3.2 `videoconvert`

```
# gst-launch-1.0 filesrc location=FHD_H264_file.264 ! h264parse ! omxh264dec ! \
videoconvert ! video/x-raw, format=BGRA ! waylandsink
```

4.3.3 `videoscale`

```
# gst-launch-1.0 filesrc location=FHD_H264_file.264 ! h264parse ! omxh264dec ! \
video/x-raw, width=1920, height=1080 ! videoscale ! video/x-raw, width=1280, \
height=720 ! waylandsink
```

4.4 Video Encode Pipeline

To avoid unreadable output file when shutting the pipeline down via ^ Ctrl + C, we can use these options below:

`gst-launch-1.0` `-e`, `--eos-on-shutdown`: Force an EOS event on sources before shutting the pipeline down. This option of `gst-launch-1.0` makes sure muxers create readable files when a muxing pipeline is shut down forcefully via ^ Ctrl + C.

`v4l2src` `num-buffers`: specifies the number of buffers to be captured from a `video4linux2` (v4l2) device and then automatically shuts the pipeline down.

NOTE Below example pipelines are tested on **e-CAM22_CURZH** MIPI camera.

4.4.1 `omxh264enc` (Hardware Accelerated Video Encoder)

1. H.264 encode and save to a MP4 container file (format conversion by `vspmfilter`)

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 ! \
video/x-raw, format=UYVY, width=1920, height=1080 ! \
vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
omxh264enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
periodicty-idr=2 use-dmabuf=TRUE ! \
video/x-h264, profile=\(string\)high, level=\(string\)4 ! h264parse ! \
video/x-h264, stream-format=avc, alignment=au ! qtmux ! filesink location=MP4_H264_file.mp4
```

2. H.264 encode and save to an elementary file (format conversion by `vspmfilter`)

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 ! \
video/x-raw, format=UYVY, width=1920, height=1080 ! \
vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
omxh264enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
periodicty-idr=2 use-dmabuf=TRUE ! \
video/x-h264, profile=\(string\)high, level=\(string\)4 ! filesink location=H264_file.264
```

3. H.264 encode 100 frames and save to MP4 container file (format conversion by `vspmfilter`)

```
# gst-launch-1.0 v4l2src num-buffers=100 device=/dev/video0 ! \
video/x-raw, format=UYVY, width=1920, height=1080 ! vspmfilter dmabuf-use=TRUE ! \
video/x-raw, format=NV12 ! omxh264enc control-rate=2 target-bitrate=10485760 \
interval_intraframes=14 periodicty-idr=2 use-dmabuf=TRUE ! \
video/x-h264, profile=\(string\)high, level=\(string\)4 ! h264parse ! \
video/x-h264, stream-format=avc, alignment=au ! qtmux ! filesink location=MP4_H264_file.mp4
```

4. H.264 encode with cropping setting and save to an elementary file (format conversion by `vspmfilter`)

```
# gst-launch-1.0 -e v4l2src device=/dev/video0 num-buffers=300 ! \
video/x-raw, format=UYVY, width=1920, height=1080, framerate=30/1 ! \
vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
omxh264enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
periodicty-idr=2 crop=0:640:0:120 use-dmabuf=TRUE ! \
video/x-h264, profile=\(string\)high, level=\(string\)4 ! \
filesink location=H264_file.264
```

> **NOTE** This pipeline will capture 10 second (300 frame, 30FPS) from camera with resolution FHD (1920x1080). Then, adding crop information to SPS of output video. The resolution after cropping is HD (1280x960) when playback.

4.4.2 `omxh265enc` (Hardware Accelerated Video Encoder)

1. H.265 encode and save to a MP4 container file (format conversion by `vspmfilter`)

```
#   gst-launch-1.0 -e v4l2src device=/dev/video0 ! \
    video/x-raw, format=UYVY, width=1920, height=1080 ! \
    vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
    omxh265enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
    periodicity-idr=2 use-dmabuf=TRUE ! \
    video/x-h265, profile=\(string\)main, level=\(string\)5 ! h265parse ! \
    video/x-h265, alignment=au ! qtmux ! filesink location=MP4_H265_file.mp4
```

2. H.265 encode and save to an elementary file (format conversion by `vspmfilter`)

```
#   gst-launch-1.0 -e v4l2src device=/dev/video0 ! \
    video/x-raw, format=UYVY, width=1920, height=1080 ! \
    vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
    omxh265enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
    periodicity-idr=2 use-dmabuf=TRUE ! \
    video/x-h265, profile=\(string\)main, level=\(string\)5 ! filesink location=H265_file.265
```

3. H.265 encode 100 frames and save to MP4 container file (format conversion by `vspmfilter`)

```
#   gst-launch-1.0 v4l2src num-buffers=100 device=/dev/video0 ! \
    video/x-raw, format=UYVY, width=1920, height=1080 ! vspmfilter dmabuf-use=TRUE ! \
    video/x-raw, format=NV12 ! omxh265enc control-rate=2 target-bitrate=10485760 \
    interval_intraframes=14 periodicity-idr=2 use-dmabuf=TRUE ! \
    video/x-h265, profile=\(string\)main, level=\(string\)5 ! h265parse ! \
    video/x-h265, alignment=au ! qtmux ! filesink location=MP4_H265_file.mp4
```

4. H.265 encode with cropping setting and save to an elementary file (format conversion by `vspmfilter`)

```
#   gst-launch-1.0 -e v4l2src device=/dev/video0 num-buffers=300 ! \
    video/x-raw, format=UYVY, width=1920, height=1080, framerate=30/1 ! \
    vspmfilter dmabuf-use=TRUE ! video/x-raw, format=NV12 ! \
    omxh265enc control-rate=2 target-bitrate=10485760 interval_intraframes=14 \
    periodicity-idr=2 crop=0:640:0:120 use-dmabuf=TRUE ! \
    video/x-h265, profile=\(string\)main, level=\(string\)5 ! \
    filesink location=H265_file.265
```

> **NOTE** This pipeline will capture 10 second (300 frame, 30FPS) from camera with resolution FHD (1920x1080). Then, adding crop information to SPS of output video. The resolution after cropping is HD (1280x960) when playback.

> **NOTE** Property `periodicity-idr` of `omxh265enc` is supported by Codec version 3.2.0 or later.

NOTE

As a limitation, `qtmux` element in GStreamer of the RZ/V2N AI SDK v6.00 cannot be used in combination with `omxh265enc` to pack the H.265 encoded stream to Media Container file. Please store the H.265 encoded stream with other container.

4.5 Video Sink Pipeline

4.5.1 `waylandsink`

1. Display `videotestsrc` with setting position – Example set `position-x=100` and `position-y=100`

```
#   gst-launch-1.0 videotestsrc ! video/x-raw, format=NV12, width=640, height=480 ! \
    waylandsink position-x=100 position-y=100
```

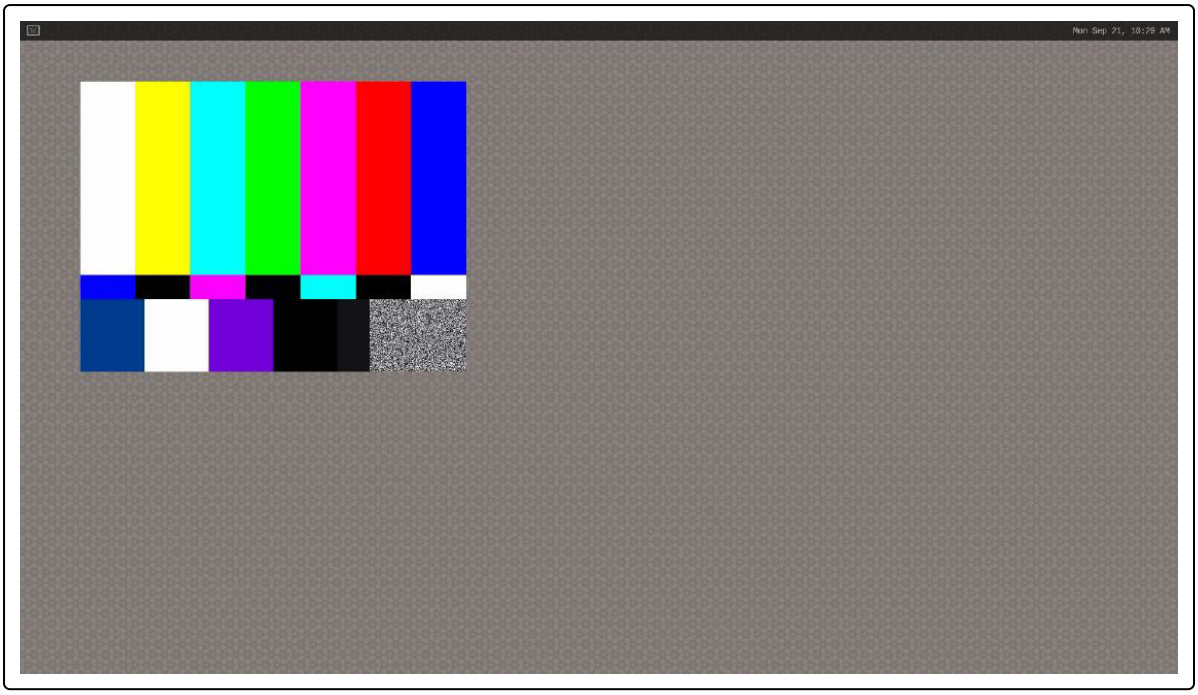


Figure 4-1: Video on Weston Screen With Setting Position

2. Display `videotestsrc` fullscreen

```
#   gst-launch-1.0 videotestsrc ! video/x-raw, format=NV12, width=640, height=480 ! \
    waylandsink fullscreen=TRUE
```

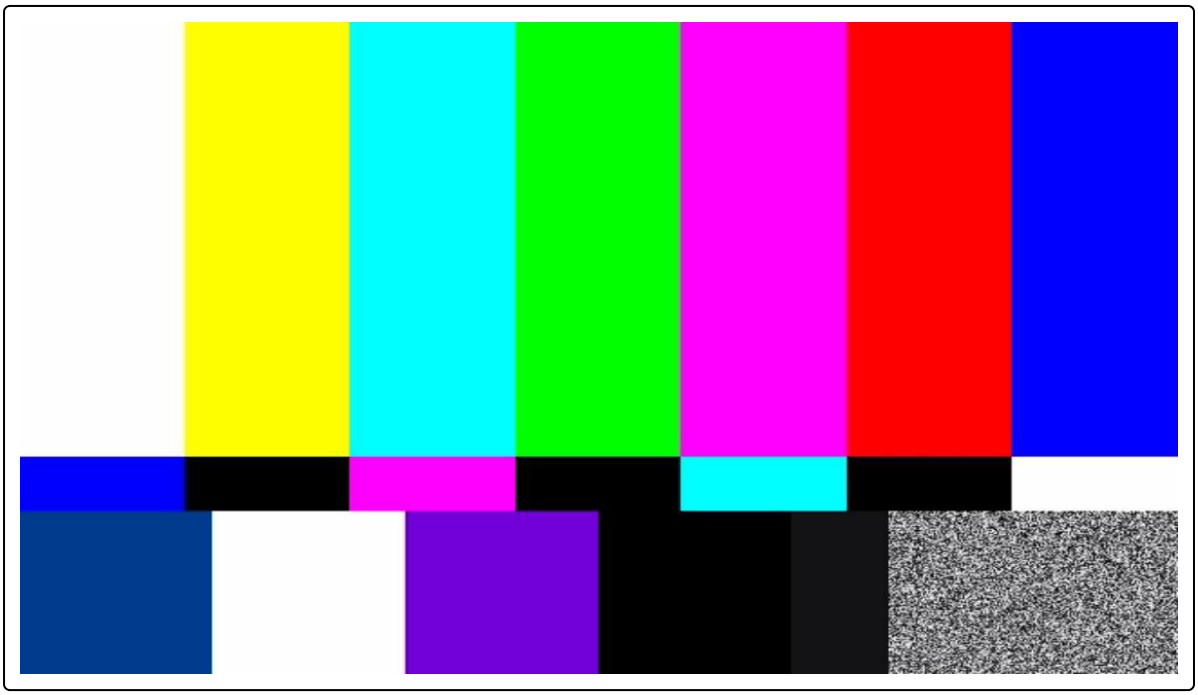


Figure 4-2: Fullscreen Video on Weston Screen

3. Display `videotestsrc` with setting wayland size of application – Example set `out-h=1000` `out-w=1500`

```
#    gst-launch-1.0 videotestsrc ! video/x-raw, format=NV12, width=640, height=480 ! \
     waylandsink out-h=1000 out-w=1500
```

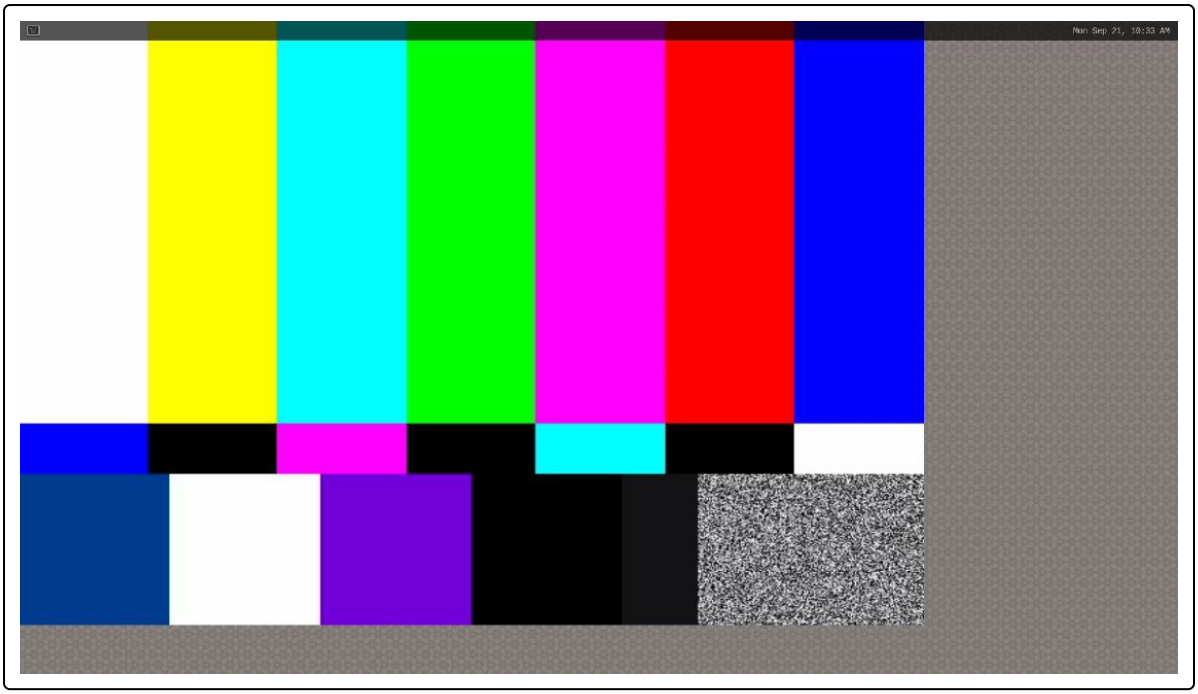


Figure 4-3: Video With Setting Size on Weston Screen

4.5.2 `fbdevsink`

- In case RZ/V2H AI SDK v5.20 (or older) and/or RZ/V2N AI SDK v5.00 (or older) (*Kernel 5.10-CIP, Yocto-3.1 (Dunfell)*):
  - Before using `fbdevsink`, stop Weston interface by `systemctl stop weston@root`
  - After using `fbdevsink`, start Weston interface by `systemctl start weston@root`
- In case RZ/V2H AI SDK v6.00 (or later) and/or RZ/V2N AI SDK v6.00 (or later) (*Kernel 6.1-CIP, Yocto-5.0 (Scarthgap)*):
  - Before using `fbdevsink`, stop Weston interface by `systemctl stop weston`
  - After using `fbdevsink`, start Weston interface by `systemctl start weston`

```
# gst-launch-1.0 videotestsrc ! video/x-raw, width=640, height=480 ! fbdevsink
```

4.5.3 `fpsdisplaysink`

```
# gst-launch-1.0 videotestsrc ! video/x-raw, width=1280, height=720, framerate=30/1 ! fpsdisplaysink
```

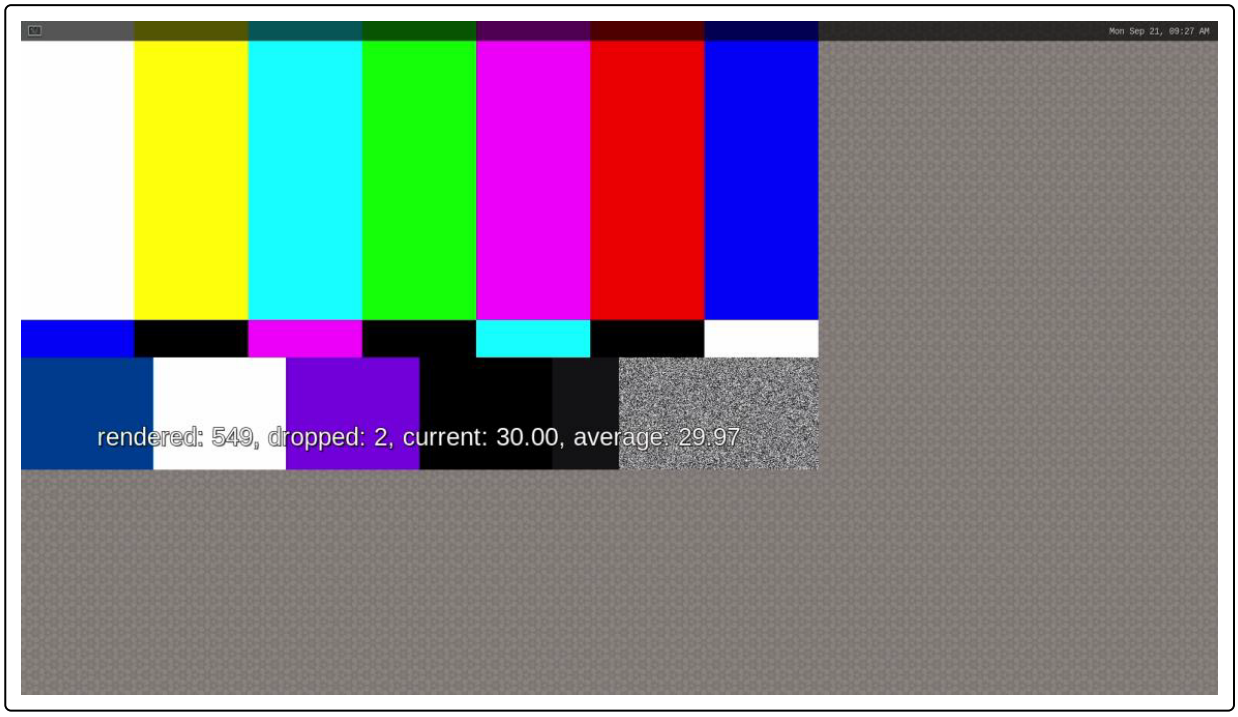


Figure 4-4: `fpsdisplaysink`

4.5.4 `glimagesink`

1. Display scaling video

```
# gst-launch-1.0 videotestsrc ! video/x-raw, width=640, height=480, framerate=30/1 ! \
glimagesink max-lateness=-1 render-rectangle='<0, 0, 720, 480>'
```

2. Rotation video

```
# gst-launch-1.0 videotestsrc ! video/x-raw, width=640, height=480, framerate=30/1 ! \
glimagesink max-lateness=-1 rotate-method=1
```

4.6 Audio Pipeline

> **NOTE** To utilize an audio device, refer to Section How To Configure and Utilize Audio.

4.6.1 Decode MP3

```
# gst-launch-1.0 filesrc location=MP3_file.mp3 ! mpegaudioparse ! mpg123audiodec ! \
audioconvert ! audioresample ! alsasink device=plughw:0,0
```

4.6.2 Decode Vorbis stream

```
# gst-launch-1.0 -v filesrc location=OGG_file.ogg ! oggdemux ! vorbisdec ! audioconvert ! \
audioresample ! alsasink device=plughw:0,0
```

4.6.3 Decode AAC

```
# gst-launch-1.0 filesrc location=ACC_file.aac ! faad ! audioconvert ! audioresample ! \
alsasink device=plughw:0,0
```

4.6.4 Encode Vorbis stream

```
# gst-launch-1.0 -e audiotestsrc wave=sine num-buffers=100 ! audioconvert ! \
vorbisenc ! oggmux ! filesink location=OGG_file.ogg
```

4.7 Network Protocol Pipeline

4.7.1 UDP

1. Receiver

```
#  gst-launch-1.0 udpsrc port=5000 buffer-size=100000000 ! application/x-rtp ! \
   rtph264depay ! h264parse ! omxh264dec ! waylandsink
```

> **NOTE**
> In order to prevent dropped buffers when streaming over a network, it is recommended to add the `max-lateness=-1` and `qos=FALSE` options after the `waylandsink`. This will ensure that the system is able to handle the incoming streams without any data loss due to buffer overflow.

2. Sender

```
#  gst-launch-1.0 filesrc location=MP4_file.mp4 ! qtdemux ! h264parse ! \
   video/x-h264, stream-format=avc, alignment=au ! rtph264pay config-interval=1 ! \
   udpsink host=RECEIVER_IP port=5000
```

> **NOTE**
> A receiver needs to run first.

4.7.2 TCP (`tcpserversink` to `tcpclientsrc`)

1. Receiver

```
#  gst-launch-1.0 tcpclientsrc host=SENDER_IP ! h264parse ! omxh264dec ! waylandsink
```

2. Sender

```
#  gst-launch-1.0 filesrc location=MP4_file.mp4 ! qtdemux ! h264parse config-interval=1 ! \
   video/x-h264, stream-format=byte-stream ! tcpserversink host=SENDER_IP
```

> **NOTE**
> A sender needs to run first.

4.7.3 TCP (`tcpclientsink` to `tcpserversrc`)

1. Receiver

```
#  gst-launch-1.0 tcpserversrc host=RECEIVER_IP port=5000 ! h264parse ! \
   omxh264dec ! waylandsink
```

2. Sender

```
#  gst-launch-1.0 filesrc location=MP4_file.mp4 ! qtdemux ! h264parse config-interval=1 ! \
   video/x-h264,stream-format=byte-stream ! tcpclientsink host=RECEIVER_IP port=5000
```

NOTE A receiver needs to run first.

5. Debugging Tool

5.1 GST_DEBUG

Use `gst-launch-1.0 --gst-debug-help` to show the list of all registered categories.

Add `GST_DEBUG=<category>:<log level>` to enable debug log of category. For example:

```
# GST_DEBUG=videotestsrc:5 gst-launch-1.0 videotestsrc ! fakesink
```

Table 5-1 Debug Levels

Level	Name	Description
0	none	No debug information is output.
1	ERROR	Logs all fatal errors. These are errors that do not allow the core or elements to perform the requested action. The application can still recover if programmed to handle the conditions that triggered the error.
2	WARNING	Logs all warnings. Typically, these are non-fatal, but user-visible problems are expected to happen.
3	FIXME	Logs all "fixme" messages. Those typically that a code path that is known to be incomplete has been triggered. It may work in most cases but may cause problems in specific instances.
4	INFO	Logs all informational messages. These are typically used for events in the system that only happen once or are important and rare enough to be logged at this level.
5	DEBUG	Logs all debug messages. These are general debug messages for events that happen only a limited number of times during an object's lifetime; these include setup, teardown, change of parameters, etc.
6	LOG	Logs all log messages. These are messages for events that happen repeatedly during an object's lifetime; these include streaming and steady-state conditions. This is used for log messages that happen on every buffer in an element for example.
7	TRACE	Logs all trace messages. Those are message that happen very very often. This is for example is each time the reference count of a GstMiniObject, such as a GstBuffer or GstEvent, is modified.
9	MEMDUMP	Logs all memory dump messages. This is the heaviest logging and may include dumping the content of blocks of memory.

5.2 Pipeline Graph

For those cases where your pipeline starts to grow too large and you lose track of what is connected with what, GStreamer has the capability to output graph files. These are **.dot** files, readable with free programs like **GraphViz**, that describe the topology of your pipeline, along with the caps negotiated in each link.

Requirements:

- Ubuntu 16.04 or later.
- **GraphViz** (detailed information at https://graphviz.org/)

Steps to use it:

1. Define output location for the generated pipeline graphs.

   Export:

   ```
   # export GST_DEBUG_DUMP_DOT_DIR=/output
   ```

   While running application or pipeline:

   ```
   # GST_DEBUG_DUMP_DOT_DIR=/output gst-launch-1.0 [OPTIONS] PIPELINE-DESCRIPTION
   ```

   or

   ```
   # GST_DEBUG_DUMP_DOT_DIR=/output ./gst_application
   ```

2. .dot files are generated in output location

   ```
   0.00.00.078803791-gst-launch.NULL_READY.dot
   0.00.00.087083833-gst-launch.READY_PAUSED.dot
   0.00.00.096721166-gst-launch.PAUSED_PLAYING.dot
   0.00.01.097021958-gst-launch.PLAYING_PAUSED.dot
   0.00.01.100463125-gst-launch.PAUSED_READY.dot
   ```

3. Converting Pipeline dot Files to various format such as PDF, PNG, JPEG, etc.

   This step should be done on Ubuntu PC which has **GraphViz**.

   ```
   $ dot -T{format} INPUT_FILE > OUTPUT_FILE
   ```

   We can use **0.00.00.096721166-gst-launch.PAUSED_PLAYING.dot** as input file.

   Example for pipeline:

   ```
   # GST_DEBUG_DUMP_DOT_DIR=/output gst-launch-1.0 videotestsrc num-buffers=30 ! waylandsink
   ```

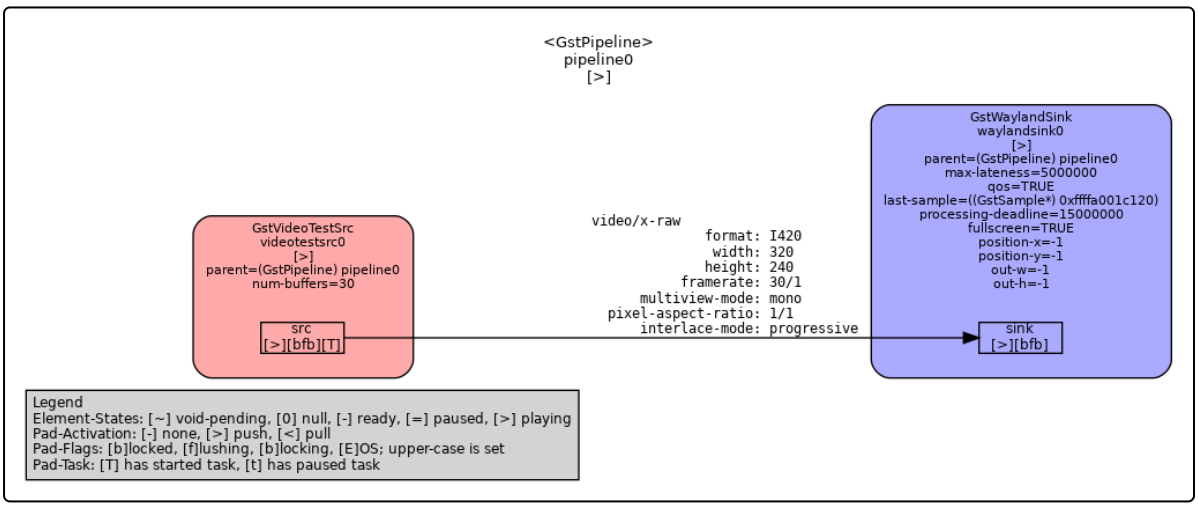


Figure 5-1: Pipeline Graph

6. Performance Comparison

This section provides a detailed analysis of the performance and efficiency of various plugins, along with their properties, to assist you in selecting the options that best meet your requirements.

Method to measure performance:

- Video Framerate: using pad probe tool to measure fps.
- CPU Load: using top command.

Performance levels:

- Video Framerate: Best > Good > Poor
- CPU Load: Mild > Moderate > Heavy

6.1 Renesas Hardware Accelerated Decoder vs Software Decoder

6.1.1 `omxh265dec` vs `avdec_h265`

File: bbb_sunflower_h265_2160p_30fps_30s.mp4

can be downloaded from: https://github.com/renesas-rz/media/blob/main/Big_Buck_Bunny/bbb_sunflower_h265_2160p_30fps_30s.mp4

File information:

Table 6-1 Decode H.265 Input File Information

Video		Audio	
Format	HEVC	Format	AAC
Format profile	Main@L5@Main	Format profile	LC
Codec ID	hev1	Codec ID	40
Duration	30s 67ms	Duration	30s 38ms
Bit rate	4,025 Kbps	Bit rate mode	Constant
Width	3,840 pixels	Bit rate	395 Kbps
Height	2,160 pixels	Channel(s)	2 channels
Display aspect ratio	16:09	Channel(s)_Original	6 channels
Frame rate mode	Constant	Sampling rate	48.0 KHz (7%)
Frame rate	15.000 fps	Compression mode	Lossy
Color space	YUV	Stream size	1.41 MiB (9%)
Chroma subsampling	4:2:0		
Bit depth	8 bits		

Video		Audio	
Scan type	Progressive		

Pipeline:

- `omxh265dec`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_h265_2160p_30fps_30s.mp4 ! qtdemux ! \
h265parse ! omxh265dec ! waylandsink -rp waylandsink0:sink
```

NOTE

Add the `qos=FALSE` option after the `waylandsink` element. This ensures the system can handle incoming streams without data loss caused by buffer overflow. (The video will play at 30 FPS.)

- `avdec_h265`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_h265_2160p_30fps_30s.mp4 ! qtdemux ! \
h265parse ! avdec_h265 ! waylandsink -rp waylandsink0:sink
```

Table 6-2 `omxh265dec` and `avdec_h265` Performance Comparison for RZ/V2H

	`omxh265dec` (`use-dmabuf`)	`avdec_h265`
Video Framerate	Best	Poor
CPU Load	Mild	Heavy
Dropped Frame	Best	Poor

6.1.2 `omxh264dec` vs `avdec_h264`

File: bbb_sunflower_1080p_60fps_normal.mp4

can be downloaded from: https://download.blender.org/demo/movies/BBB/bbb_sunflower_1080p_60fps_normal.mp4.zip

File information:

Table 6-3 Decode H.264 Input File Information

Video		Audio	
Format	AVC	Format	MPEG Audio
Format profile	High@L4.2	Format version	Version 1
Format settings, CABAC	Yes	Format profile	Layer 3
Format settings, ReFrames	4 frames	Mode	Joint Strereo
Codec ID	avc1	Mode extension	MS Stereo

Video		Audio	
Codec ID/Info	Advanced Video Coding	Codec ID	6B
Duration	10mn 34s	Duration	10mn 34s
Bit rate	4,000 Kbps	Bit rate mode	Constant
Maximum bit rate	19.7 Mbps	Bit rate	160 Kbps
Width	1,920 pixels	Maximum bit rate	165 Kbps
Height	1,080 pixels	Channel(s)	2 channels
Display aspect ratio	16:09	Sampling rate	48.0 KHz
Frame rate mode	Constant	Compression mode	Lossy
Frame rate	60.000 fps	Stream size	12.1 MiB (4%)
Color space	YUV	Writing library	LAME3.99r
Chroma subsampling	4:2:0	Language	English
Bit depth	8 bits		
Scan type	Progressive		

Pipeline:

- `omxh264dec`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_1080p_60fps_normal.mp4 ! qtdemux ! \
h264parse ! omxh264dec ! waylandsink -rp waylandsink0:sink & \
gst-launch-1.0 filesrc location=bbb_sunflower_1080p_60fps_normal.mp4 ! qtdemux ! \
h264parse ! omxh264dec ! waylandsink -rp waylandsink0:sink
```

NOTE

Add the `qos=FALSE` option after the `waylandsink` element. This ensures the system can handle incoming streams without data loss caused by buffer overflow. (The video will play at 60 FPS.)

- `avdec_h264`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_1080p_60fps_normal.mp4 ! qtdemux ! \
h264parse ! avdec_h264 ! waylandsink -rp waylandsink0:sink & \
gst-launch-1.0 filesrc location=bbb_sunflower_1080p_60fps_normal.mp4 ! qtdemux ! \
h264parse ! avdec_h264 ! waylandsink -rp waylandsink0:sink
```

Table 6-4 `omxh264dec` and `avdec_h264` Performance Comparison for RZ/V2H

	`omxh264dec` (`use-dmabuf`)	**`avdec_h264`**
Video Framerate	Best	Poor
CPU Load	Mild	Heavy

	omxh264dec (use-dmabuf)	avdec_h264
Dropped Frame	Best	Poor

6.2 Running Mode Comparison: `use-dmabuf` vs `no-copy`

File: bbb_sunflower_h265_2160p_30fps_30s.mp4

File information is described in Table 6-1

Pipeline:

- `use-dmabuf=TRUE`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_h265_2160p_30fps_30s.mp4 ! qtdemux ! \
h265parse ! omxh265dec use-dmabuf=TRUE ! waylandsink -rp waylandsink0:sink
```

- `no-copy=TRUE`

```
# gst-launch-1.0 filesrc location=bbb_sunflower_h265_2160p_30fps_30s.mp4 ! qtdemux ! \
h265parse ! omxh265dec no-copy=TRUE ! waylandsink -rp waylandsink0:sink
```

Table 6-5 `use-dmabuf` and `no-copy` Performance Comparison for RZ/V2H

	`omxh265dec` (`use-dmabuf`)	`omxh265dec` (`no-copy`)
Video Framerate	Best	Poor
CPU Load	Mild	Moderate
Dropped Frame	Best	Poor

7. Limitation

7.1 `vspmfilter`

- `vspmfilter` doesn't work with `fakesink`, `filesink` and `waylandsink` on the default mode.
- `vspmfilter` doesn't support scale up due to hardware limitation of ISU. ISU only supports scale down function with bilinear interpolation.
- `vspmfilter` works with USB cameras, as long as the format and framerate match the selected resolution.

Appendix

A How to Write Output to Raw File Instead of Displaying Using `filesink`

```
#    gst-launch-1.0 filesrc location=FHD_MP4_file.mp4 ! qtdemux ! h264parse ! \
     omxh264dec ! filesink location=FDH_RAW_file.raw
```

B How to Change the Default Video Filter In `playbin` and `playbin3` Elements

`playbin` and `playbin3` provides a stand-alone everything-in-one abstraction for an audio and/or video player.

The default video filter for video chain in **playbin** and **playbin3** is defined in `/etc/gstpbfilter.conf`.

If we want to use `vspmfilter` or other video filter, we can change this default.

Here is how to change the default video filter in **playbin** and **playbin3**:

Please edit `/etc/gstpbfilter.conf` to change the default video filter.

In case of using `vspmfilter` as default video filter.

gstpbfilter.conf

```
video-filter=vspmfilter
```

In case of using `videoconvert` as default video filter.

gstpbfilter.conf

```
video-filter=videoconvert
```

C How to Change the Default Mode In OMX Video Decoder Element

When we specify neither `no-copy` nor `use-dmabuf` option. The default mode of omx video decoder element is `use-dmabuf=TRUE`.

Here is how to change the default mode of omx video decoder element.

Please edit `/etc/xdg/gstomx.conf` to change the default mode.

In case of using `no-copy` mode as default, add the following definition to the “hacks”.

gstomx.conf

```
hacks=use-no-copy-mode-as-default
```

D How to Detect Active Camera

List all connected cameras:

```
#    v4l2-ctl --list-devices
```

e.g.: Renesas RZ/V AI SDK for RZ/V2H with RZ/V2H EVK

```
root@rzv2h-evk-ver1:~# v4l2-ctl --list-devices
RZG2L_CRU (platform:16000000.cru0):
        /dev/video0
        /dev/media0

renesas,cru-r9a09g057 (platform:16010000.cru1):
        /dev/media1

renesas,cru-r9a09g057 (platform:16020000.cru2):
        /dev/media2

renesas,cru-r9a09g057 (platform:16030000.cru3):
        /dev/media3

Rapoo Camera: Rapoo Camera (usb-15810100.usb-1):
        /dev/video1
        /dev/video2
        /dev/media4
```

List all supported formats for selected camera

```
#    v4l2-ctl -d <path-to-device;e.g.:/dev/video1> --list-formats-ext
```

e.g.: Device: `/dev/video1` (`Rapoo Camera (usb-15810100.usb-1)`) The terminal output shown here has been selectively truncated.

```
root@rzv2h-evk-ver1:~# v4l2-ctl -d /dev/video1 --list-formats-ext
ioctl: VIDIOC_ENUM_FMT
        Type: Video Capture
        [0]: 'MJPG' (Motion-JPEG, compressed)
                Size: Discrete 1920x1080
                        Interval: Discrete 0.033s (30.000 fps)
                        Interval: Discrete 0.050s (20.000 fps)
```

Test with Gstreamer Pipeline:

```
root@rzv2h-evk-ver1:~# gst-launch-1.0 v4l2src device=/dev/video1 io-mode=mmap num-buffers=30 ! video/x-
raw,format=YUY2,width=640,height=480 ! waylandsink
Setting pipeline to PAUSED ...
Pipeline is live and does not need PREROLL ...
Setting pipeline to PLAYING ...
New clock: GstSystemClock
Got EOS from element "pipeline0".
Execution ended after 0:00:01.492581917
Setting pipeline to NULL ...
Total time: 1.492584 seconds
Freeing pipeline ...
```

E How to Configure and Utilize MIPI Camera

To utilize a MIPI camera, it is necessary to perform configuration settings prior to its operation.

Example configuration for integrating the **e-CAM22_CURZH** MIPI camera with the **"Renesas RZ/V AI SDK for RZ/V2H with RZ/V2H EVK"** (also applicable for **"Renesas RZ/V AI SDK for RZ/V2N with RZ/V2N EVK"**):

Detect `/dev/video0` and `/dev/media0`

E.1 Setup MIPI Camera

NOTE

Please confirm whether your system can detect the device IP (CRU-IP) sub-device or not.

If you can see **cru-ip-16000000.cru0** as below, please go ahead.

```
root@rzv2n-evk:~# cat /sys/class/video4linux/v4l-subdev*/name | grep "cru-ip"
cru-ip-16000000.cru0
```

If you cannot find CRU IP, please use the steps in **Setup MIPI Camera without Device IP**.

Firstly, to determine the initial CSI-2 (Camera Serial Interface 2) and device IP (CRU-IP) sub-device recognized by the system, execute the following command, and assign its output to the variable `csi2` and `ip`:

```
#  export csi2=$(cat /sys/class/video4linux/v4l-subdev*/name | grep "csi2" | head -n 1)
#  export ip=$(cat /sys/class/video4linux/v4l-subdev*/name | grep "cru-ip" | head -n 1)
```

In this example `csi2` is **csi-16000400.csi20** and `ip` is **cru-ip-16000000.cru0**.

Below command resets the media controller state. It clears all existing links and formats on the media device represented by `/dev/media0`.

```
#  media-ctl -d /dev/media0 -r
```

Below command sets up a link between two entities in the media device. It links the output pad 1 of the **csi-16000400.csi20** entity to the input pad 0 of the **cru-ip-16000000.cru0** entity and activates the link.

```
#  media-ctl -d /dev/media0 -l "'${csi2}':1 -> '${ip}':0 [1]"
```

Below command sets the video format for a specific pad (pad 1) of the **csi-16000400.csi20** entity. The format specified here is **UYVY8_2X8** with a resolution of 1920x1080 and no interlaced field.

```
#  media-ctl -d /dev/media0 -V "'${csi2}':1 [fmt:UYVY8_2X8/1920x1080 field:none]"
```

Below command sets the video format for pad 0 of the **imx462 0-001f** entity to **UYVY8_2X8** at 1920x1080 resolution with no interlaced field.

```
#  media-ctl -d /dev/media0 -V "'imx462 0-001f':0 [fmt:UYVY8_2X8/1920x1080 field:none]"
```

Below command sets the video format for pad 0 of the **cru-ip-16000000.cru0** entity to **UYVY8_2X8** at 1920x1080 resolution with no interlaced field.

```
#    media-ctl -d /dev/media0 -V "'${ip}':0 [fmt:UYVY8_2X8/1920x1080 field:none]"
```

E.2 Setup MIPI Camera without Device IP

> **NOTE** All steps here are basically for AI SDK v5.20 or older.

Firstly, to determine the initial CSI-2 (Camera Serial Interface 2) sub-device recognized by the system, execute the following command, and assign its output to the variable `csi2`:

```
#    csi2=$(cat /sys/class/video4linux/v4l-subdev*/name | grep "csi2" | head -n 1)
```

In this example `csi2` is **rzg2l_csi2 16000400.csi20**.

Below command resets the media controller state. It clears all existing links and formats on the media device represented by `/dev/media0`.

```
#    media-ctl -d /dev/media0 -r
```

Below command sets up a link between two entities in the media device. It links the output pad 1 of the **rzg2l_csi2 16000400.csi20** entity to the input pad 0 of the **CRU output** entity and activates the link.

```
#    media-ctl -d /dev/media0 -l "'$csi2':1 -> 'CRU output':0 [1]"
```

Below command sets the video format for a specific pad (pad 1) of the **rzg2l_csi2 16000400.csi20** entity. The format specified here is **UYVY8_2X8** with a resolution of 1920x1080 and no interlaced field.

```
#    media-ctl -d /dev/media0 -V "'$csi2':1 [fmt:UYVY8_2X8/1920x1080 field:none]"
```

Below command sets the video format for pad 0 of the **imx462 0-001f** entity to **UYVY8_2X8** at 1920x1080 resolution with no interlaced field.

```
#    media-ctl -d /dev/media0 -V "'imx462 0-001f':0 [fmt:UYVY8_2X8/1920x1080field:none]"
```

E.3 Configuring e-CAM22_CURZH Camera Capture FPS

By default, the **e-CAM22_CURZH** camera is configured at 60fps. To use 30fps, please set it as below (in this case, the device id for camera is 0):

```
#    v4l2-ctl -d 0 -c framerate=30
```

F How to Configure and Utilize Audio

To utilize an audio device, it is necessary to perform configuration settings prior to its operation.

First, enable Dip Switch DSW2 as below table.

Table 8-1 Dip Switch 2 (DSW2) for RZ/V2H EVK and RZ/V2N EVK v1.0

DSW2-1	DSW2-2	DSW2-3	DSW2-4	DSW2-5	DSW2-6
ON	ON	ON	ON	OFF	OFF

Table 8-2 Dip Switch 2 (DSW2) for RZ/V2N EVK v2.0

DSW2-1	DSW2-2
ON	ON

Second, setup audio route using `amixer` as below.

```
#     amixer -q cset name='Aux Switch' on
#     amixer -q cset name='Mixin Left Aux Left Switch' on
#     amixer -q cset name='Mixin Right Aux Right Switch' on
#     amixer -q cset name='ADC Switch' on
#     amixer -q cset name='Mixout Right Mixin Right Switch' off
#     amixer -q cset name='Mixout Left Mixin Left Switch' off
#     amixer -q cset name='Headphone Volume' 100%
#     amixer -q cset name='Headphone Switch' on
#     amixer -q cset name='Mixout Left DAC Left Switch' on
#     amixer -q cset name='Mixout Right DAC Right Switch' on
#     amixer -q cset name='DAC Left Source MUX' 'DAI Input Left'
#     amixer -q cset name='DAC Right Source MUX' 'DAI Input Right'
#     amixer -q sset 'Mic 1 Amp Source MUX' 'MIC_P'
#     amixer -q sset 'Mic 2 Amp Source MUX' 'MIC_P'
#     amixer -q sset 'Mixin Left Mic 1' on
#     amixer -q sset 'Mixin Right Mic 2' on
#     amixer -q sset 'Mic 1' 100% on
#     amixer -q sset 'Mic 2' 100% on
#     amixer -q sset 'Lineout' 100% on
#     amixer -q set "Headphone" 100% on
#     amixer -q set 'DVC In',0 100%
#     amixer -q cset name='Mixin PGA Volume' 2
#     amixer -q cset name='Mixin PGA Switch' on
#     amixer -q cset name='ADC Volume' 67%
#     amixer -q set 'DVC Out',0 100%
```

Then, test playback using GStreamer pipeline as below.

```
#     gst-launch-1.0 filesrc location=/usr/share/sounds/alsa/Front_Right.wav ! wavparse ! \
      audioconvert ! alsasink
```

Index

R

Renesas Extended Element

`omxh264dec`
5-6, 8-10, 15-17, 37-38, 41, 50-51, 56-58, 61

`omxh264enc` 7, 9, 11, 16, 21-24, 38, 42

`omxh265dec` 5-6, 9-10, 12, 18-20, 38-39, 55-56, 59

`omxh265enc` 7, 9, 11-12, 19, 25-28, 39, 43-44

`v4l2src` 6-7, 9, 11, 29-30, 41-43, 64

`vspmfilter` 6-7, 9, 11, 32-33, 41-43, 60, 62

`waylandsink`
5-7, 9, 11-12, 34-35, 37-41, 45-46, 50-51, 54, 56-57, 59-60, 64

Revision History	Linux Interface Specification GStreamer Software Manual

Rev.	Date	Description	
		Page	Summary
1.00	May 28, 2024	-	Initial revision of RZ/V2H Group GStreamer User's Manual: Software
1.01	Oct. 04, 2024	17	Adding NOTE for case using `vspmfilter` with USB Camera
		22	Update "4.3 Video Conversion Pipeline" • Introducing `vspmfilter`
		23 – 24	Update "4.4 Video Encode Pipeline" • Introducing `vspmfilter` • Remove `framerate` property in pipeline
		35 – 37	Add "6. Performance Comparison"
		38	Add "7. Limitation"
		41	Add "8.3 How to Change the Default Mode In OMX Video Decoder Element" into "8. Appendix" ("6. Appendix" on previous revision)
1.02	Feb. 28, 2025	-	Add RZ/V2N Initial revision of RZ/V2H Group and RZ/V2N Group GStreamer User's Manual: Software
		21	Update table `vspmfilter` Expansion Properties • In older revision, property `dmabuf-use`, was incorrectly written as `use-dmabuf`
		47	Add "F How to Configure and Utilize Audio" into "Appendix" ("8. Appendix" on previous revision)
1.03	Apr. 25, 2025	-	Update Revision number to 1.03
		4 – 6	Update "2.3 Module Configuration" figures • Introducing `omxh265dec` and `omxh265enc` into existing figures • Replace part of Hardware area for "2.3.3 Video Capture and Encode" figure from (mistaken written) VCPL4 to VCD
		14, 16	Update NOTE of `omxh264dec` and `omxh265dec` following RZ/V2H AI SDK 5.20

Rev.	Date	Description	
		Page	Summary
1.03	Apr. 25, 2025	18, 20	Update NOTE of `omxh264enc` and `omxh265enc` following RZ/V2H AI SDK 5.20
		26 – 27	Update decoding example pipelines for "4.2 Decode Pipeline" • Add H265 file decode example pipeline for "4.2.1 `decodebin` " • Add multiple MP4 files decode example pipelines for "4.2.2 `omxh264dec` (Hardware Accelerated Video Decoder)" • Add multiple MP4 files decode example pipelines for "4.2.4 `omxh265dec` (Hardware Accelerated Video Decoder)"
		39 – 42	Update "6. Performance Comparison" • Add H.265 performance comparison • Replace H.264 performance comparison with multiple decoding usecase
		50	Update audio setting
2.00	Jun. 30, 2025	-	Update Revision number to 2.00 Initial revision for RZ/V2N AI SDK v6.00 (*Kernel 6.1-CIP, Yocto-5.0 (Scarthgap)*)

		1, 3, 9, 11, 24, 27, 29 – 30, 33	Update and add information regarding RZ/V2N AI SDK v6.00
			• Update and/or add **Note**/**Caution** regarding to operating environment
			• Update tables of:
			  • "1.1 Related Documents"
			  • "2.1 Hardware and Software Environment" ("2.1 Hardware Environment" on previous revision)
			  • "3.2 GStreamer Plug-in and Element"

		48	Update Appendix "B How to Change the Default Video Filter In `playbin` Element"
		51 – 52	Update Appendix "E How to Configure and Utilize MIPI Camera"
2.01	Dec. 26, 2025	-	Update Revision number to 2.01 Initial revision for RZ/V2H AI SDK v6.00 (*Kernel 6.1-CIP, Yocto-5.0 (Scarthgap)*)

Rev.	Date	Description	
		Page	Summary

2.01	Dec. 26, 2025	1, 3	Update and add information regarding RZ/V2H AI SDK v6.00
			• Update and/or add **Note**/**Caution** regarding to operating environment
			• Update tables of:
			  • "1.1 Related Documents"
			  • "2.1 Hardware and Software Environment" ("2.1 Hardware Environment" on previous revision)

		8	Update table of "3.2 GStreamer Plug-in and Element"
			• Update row of "License" and "Version" following new column of "Yocto Version" (Dunfell, Scarthgap) due to different licenses. In previous revisions was accidentally written as:
			  • LGPL ver2 (from revisions 1.00 to 2.00):
			  • "1. GStreamer core plugins"
			  • GPL ver2 (from revisions 1.00 to 2.00):
			  • "2. GStreamer Base plugins"
			  • LGPL ver2 (in revisions 1.02, 1.03 and 2.00):
			  • "3. GStreamer Good plugins"
			  • "4. GStreamer Bad plugins"
			  • "5. GStreamer Ugly plugins"
			  • "6. GStreamer gst-omx plugins"
			  • LGPL ver2.1 (from revisions 1.00 to 2.00):
			  • "7. gst-plugin-vspmfilter"
			• Update row of "Remarks" for "7. gst-plugin-vspmfilter"

		9 – 11, 33, 56	Add information regarding `playbin3` • Add `playbin3` to "3.2.1 `playbin` and `playbin3` " and Table 3-2 `playbin` and `playbin3` and Note • Add `playbin3` to NOTE of "3.2.6 Video Sink Plugin" • Add `playbin3` element example for "4.1 `playbin` and `playbin3` " • Add `playbin3` to Appendix "B How to Change the Default Video Filter In `playbin` and `playbin3` Elements"

Rev.	Date	Description	
		Page	Summary
2.01	Dec. 26, 2025	14 - 30,	Update formats, property add update and/or add NOTE of "3.3 Renesas Extended Element" • Update table of: • "3.3.5 Source Element: `v4l2src`" • Update tables and update and/or add NOTE of: • "3.3.1 Video H.264 Decode Element: `omxh264dec`" • "3.3.2 Video H.265 Decode Element: `omxh265dec`" • "3.3.3 Video H.264 Encode Element: `omxh264enc`" • "3.3.4 Video H.265 Encode Element: `omxh265enc`" • "3.3.6 Video Conversion Element: `vspmfilter`"
		34 - 35, 37 - 39, 42	Update example pipeline add update and/or add NOTE for "4. Example Pipeline" • Update and/or add new example pipeline with NOTE for: • "4.3.1 `vspmfilter`" of "4. Video convert and display" • "4.3.3 `videoscale`" • "4.7.2 TCP (`tcpserversink` to `tcpclientsrc`)" • "4.7.3 TCP (`tcpclientsink` to `tcpserversrc`)" • Add NOTE regarding `qos=false` option for: • "4.2.2 `omxh264dec` (Hardware Accelerated Video Decoder)" • "4.2.4 `omxh265dec` (Hardware Accelerated Video Decoder)" • Add NOTE regarding `interval_intraframes` option for: • "4.4.1 `omxh264enc` (Hardware Accelerated Video Encoder)" • "4.4.2 `omxh265enc` (Hardware Accelerated Video Encoder)" • Add instruction for Scarthgap before and after using for "4.5.2 `fbdevsink`"
		54	Update "7.1 `vspmfilter`"
2.02	Mar. 27, 2026	-	Update Revision number to 2.02

Rev.	Date	Description	
		Page	Summary

2.02	Mar. 27, 2026	1, 3	Update and add information regarding RZ/V2N AI SDK v6.30
			• Update tables of:
			  • Add RZ/V2N AI SDK v6.30 repository to "1.1 Related Documents"
			  • Update row of "Remarks" for "2.1 Hardware and Software Environment"

		4 - 6	Update software configuration diagrams for GStreamer
			• Update diagrams of:
			  • "2.3.1 Video Decode and Display"
			  • "2.3.2 Video Capture and Display"
			  • "2.3.3 Video Capture and Encode"

		11	Replace "videosink" with "video-sink" in NOTE of 3.2.6 Video Sink Plugin

		14 - 28,
		31 - 32
			Update and/or add new formats, properties, resolution; update and/or add NOTE of "3.3 Renesas Extended Elements
			• Update table of:
			  • "3.3.5 Source Element: `v4l2src`"
			• Update tables; update and/or add NOTE of:
			  • "3.3.1 Video H.264 Decode Element: `omxh264dec`"
			  • "3.3.2 Video H.265 Decode Element: `omxh265dec`"
			  • "3.3.3 Video H.264 Encode Element: `omxh264enc`"
			  • "3.3.4 Video H.265 Encode Element: `omxh265enc`"
			  • "3.3.7 Video Conversion Element: `waylandsink`"
			• Update NOTE of: 3.3.6 Video Conversion Element: `vspmfilter`

Rev.	Date	Description	
		Page	Summary
2.02	Mar. 27, 2026	34 - 36, 39 - 40, 47	Update example pipeline add update and/or add NOTE for "4. Example Pipeline" • Add new example pipeline with NOTE for down scaling, color conversion and decode a crop video to: • "4.2.2 `omxh264dec` (Hardware Accelerated Video Decoder)" • "4.2.4 `omxh265dec` (Hardware Accelerated Video Decoder)" • Add new example pipeline with NOTE for `crop` property to: • "4.4.1 `omxh264enc` (Hardware Accelerated Video Encoder)" • "4.4.2 `omxh265enc` (Hardware Accelerated Video Encoder)" • Remove NOTE regarding `interval_intraframes` option due to duplication: • "4.4.1 `omxh264enc` (Hardware Accelerated Video Encoder)" • "4.4.2 `omxh265enc` (Hardware Accelerated Video Encoder)" • Add a NOTE to prevent dropped buffer when streaming to "4.7.1 `UDP`"
		64	Add `amixer set 'DVC Out',0 100%` to "F How to Configure and Utilize Audio"
2.03	Oct. 30, 2026	-	Update Revision number to 2.03
		-	Rename Document titles • Change from "User's Manual: Software" to "Software Manual" • Change from "User's Manual: Hardware" to "Hardware Manual"
		3	Update table of "2.1 Hardware and Software Environment" • Add supported versions (v1.0 and v2.0) of RZ/V2N Evaluation Board Kit • Add a reference to the RZ/V2N EVK web page
		9	Update table of "3.2 GStreamer Plug-in and Element" • Remove version information of "7. gst-plugin-vspmfilter" since it is not separated by GStreamer version
		10	Improve phrasing and capitalize feature descriptions of "3.2.1 `playbin` and `playbin3`"

Rev.	Date	Description	
		Page	Summary
2.03	Oct. 30, 2026	16, 19	Update footnotes of "3.3.1 Video H.264 Decode Element: `omxh264dec` " and "3.3.2 Video H.265 Decode Element: `omxh265dec` " • Reorder footnotes • Add a reference to Appendix C to the footnote of default value of `no-copy` and `use-dmabuf`
		22 - 23	Update expansion properties table of "3.3.3 Video H.264 Encode Element: `omxh264enc` " • Add footnote for `no-copy` and `use-dmabuf` • Add values 4 and 5 for `scan-type` • Add NOTE indicating that `send-eos` and `ref-frames` properties are not supported • Add `b-frames` property • Update description and clarify default value of `interval_intraframes` • Correct descriptions of `control-rate` values 3 and 4 • Clarify default value of `target-bitrate` • Correct property name from `periodicty-idr` to `periodicity-idr` ; update its description and clarify its default value • Update footnote numbering
		26 - 27	Update expansion properties table of "3.3.4 Video H.265 Encode Element: `omxh265enc` " • Add footnote for `no-copy` and `use-dmabuf` • Add values 4 and 5 for `scan-type` • Remove `send-eos` and `ref-frames` properties • Add `b-frames` property • Update description and clarify default value of `interval_intraframes` • Correct descriptions of `control-rate` values 3 and 4 • Clarify default value of `target-bitrate` • Update description and clarify default value of `periodicity-idr` • Update footnote numbering

Rev.	Date	Description	
		Page	Summary
2.03	Oct. 30, 2026	30 - 31	Update footnotes of "3.3.5 Source Element: `v4l2src`" • Reorder footnotes of `mmap` and `dmabuf` • Add footnote for `userptr` (unsupported mode) • Update footnote numbering of `no-resurect-buf`
		37, 39, 41 - 43, 45, 50, 56 - 57, 59	Use `TRUE` / `FALSE` consistently for boolean properties in pipelines and NOTEs of "4. Example Pipeline" and "6. Performance Comparison" • "4.2 Video Decode Pipeline" • "4.3 Video Conversion Pipeline" • "4.4 Video Encode Pipeline" • "4.5 Video Sink Pipeline" • "4.7 Network Protocol Pipeline" • "6.1 Renesas Hardware Accelerated Decoder vs Software Decoder" • "6.2 Running Mode Comparison: `use-dmabuf` vs `no-copy`"
		63	Correct `usedmabuf=true` to `use-dmabuf=TRUE` in Appendix "C How to Change the Default Mode In OMX Video Decoder Element"
		67	Add Dip Switch 2 (DSW2) configuration for RZ/V2N EVK v2.0 to Appendix "F How to Configure and Utilize Audio"

Linux Interface Specification GStreamer Software Manual

Publication Date: Rev.2.03 Oct. 30, 2026

Publication by: Renesas Electronics Corporation

Linux Interface Specification GStreamer

Renesas Electronics Corporation

R01US0653EJ0203