

Release note  
Linux 5.10.104-tegra  
(R35.1 / Jetpack5.0.2)

# Content

3

4

5

7

8

9

10

**Error! Bookmark not defined.**

14

16

21

22

23

## Revision History

[illegible]

## Release

---

The release file includes:

- 1.1 ReleaseNote.docx
- 1.2 release.zip: Files for Orin, Xavier, and Xavier NX. It includes
  - 1.2.1 Yuan's Driver: driver/LXV4L2D\_\*.ko (Release version)  
driver/debug/ LXV42D\_\*.ko(debug version)
  - 1.2.2 V4L Drivers: driver/V4L/videobuf\*.ko
- 1.3 Firmware of Capture card: files in firmware folder
- 1.4 Scripts file: \*.sh
- 1.5 The release is based on Nvidia **R35.1**  
<https://developer.nvidia.com/embedded/jetson-linux-r351>
- 1.6 Driver version is V1.1.0.120.**1160**\_VB2
- 1.7 Supported capture cards: **SC3x0/ SC4x0/SC5x0/SC7x0/**
- 1.8 Supported Platform: Orin, Xavier, Xavier NX

## Insert Driver and firmware

---

- 1 Power off the Jetson device, and plugin Yuan's capture card.
- 2 Power on the Jetson device.
- 3 Copy the release.zip to home directory(/home/nvidia), and open the terminal.
- 4 Unzip the zip file
  - 4.1 `$ cd ~` (change to home directory -/home/nvidia)
  - 4.2 `$ sudo su` (switch to root – enter password “nvidia”)
  - 4.3 `# unzip release.zip`
  - 4.4 `# cd release` (enter release folder)
- 5 Set execution permission for scripts
  - 5.1 `# chmod +x *.sh`
- 6 Execute the script file
  - 6.1 `# ./setup.sh`
- 7 The script file will do these procedures
  - 7.1 Copy firmware files of capture card to /lib/firmware.
  - 7.2 Copy V4L drivers to /lib/modules/5.10.104-tegra/kernel/drivers/media/v4l2-core and  
/lib/modules/5.10.104-tegra/kernel/drivers/media/common/videobuf2
  - 7.3 Copy Yuan's driver to /lib/modules/5.10.104-tegra/misc
  - 7.4 Install V4L and Yuan's drivers.
- 8 After installing Yuan's driver, driver will upgrade the firmware of capture card.  
Please wait 60 seconds, and then reboot Xavier.
  - 8.1 `# reboot`
- 9 Open the terminal, and change to root.
  - 9.1 `$ cd /home/nvidia/release`
  - 9.2 `$ sudo su`
- 10 Check if the video devices (video0 or video\*) are created by driver.
  - 10.1 `# ls /dev/video*`
  - 10.2 The result should be  
`# /dev/video0`
- 11 Make sure that you can connect to internet, and install the Qt V4L2 test Utility to verify the video input.
  - 11.1 `# ./install.sh`
- 12 Open “Terminal app”, and enter the “v4l2-ctl --list-devices” to check if the video device has been created. (For example)

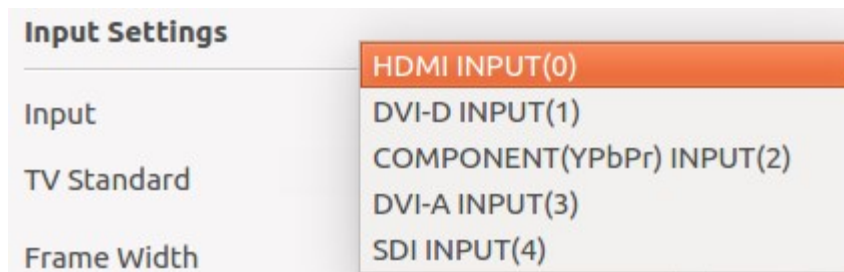
12.1 nvidia@tegra-ubuntu:~\$ **v4l2-ctl --list-devices**

12.2 SC3x0-TWxxxx:RAW 00.00 xxxxxxxx (PCIe: PCI Bus 0000:01 00):

SC5x0-MZxxxx:RAW 00.00 xxxxxxxx (PCIe: PCI Bus 0000:01 00):

SC710-SC710:RAW 00.00 xxxxxxxx (PCIe: PCI Bus 0000:01 00):

12.3 The index of video input, 0 for HDMI, and 4 for SDI.



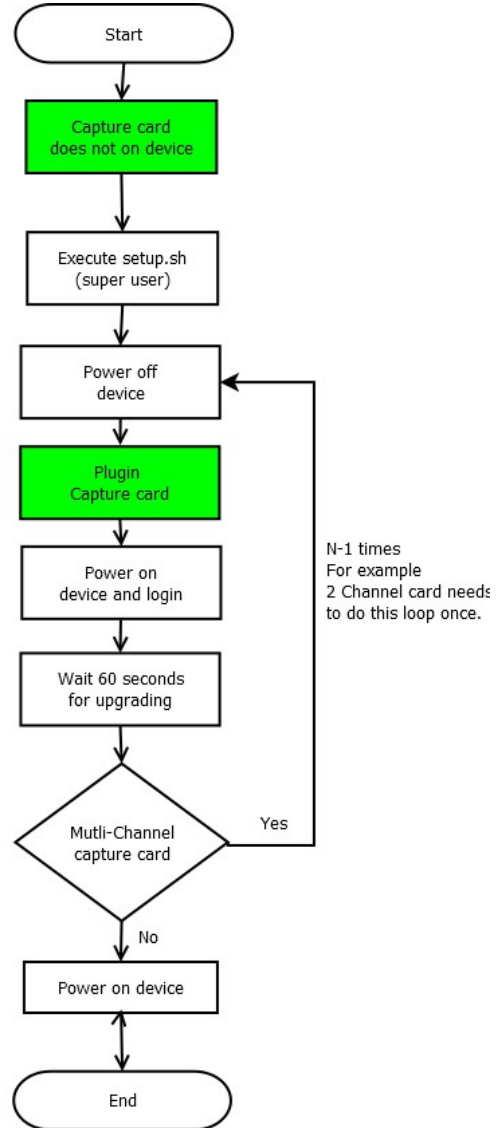
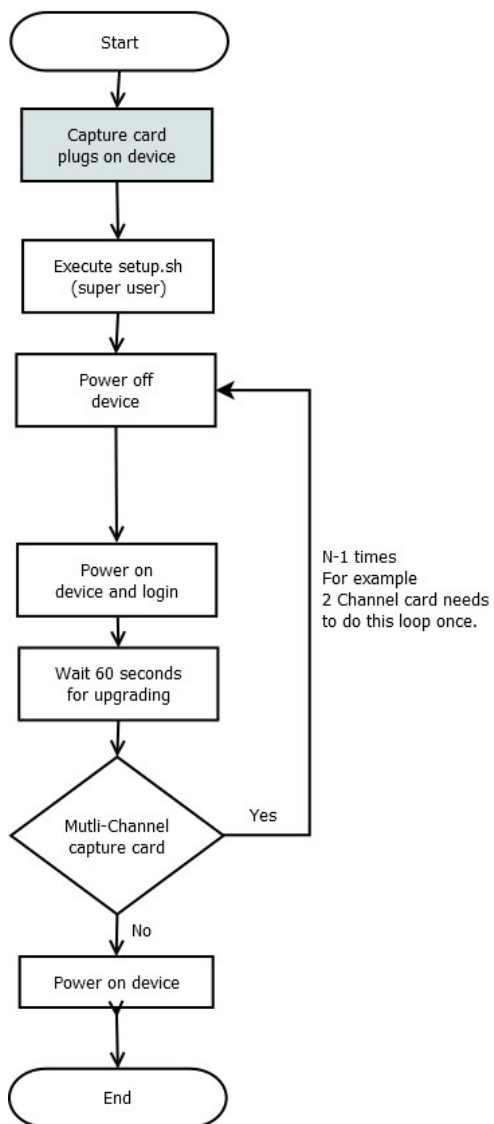
**13 Note: The commands in blue color which you need to execute in the terminal.**

## Debug

---

1. Copy the new release “release.zip” to home directory  
(For example /home/nvidia)
2. Unzip the zip file by command  
\$ `sudo su` (enter password, for example nvidia)  
# `unzip release.zip`
3. Reinstall the debug driver  
# `cd release`  
# `./dsetup.sh`
4. Power off Jetson device. **Remove other PCI devices**, for example “AverMedia CM313B” capture card.
5. Power on Jetson device, enter the debug directory which you unzip  
(For example: \$ `cd /home/nvidia/release/debug`)
6. Generate log\_setup.txt by executing debug.sh, and send the log\_setup.txt for analyzing  
# `sudo ./debug.sh`

## Flow chart of Driver upgrade for SC3C0/SC550/SC5C0





## TVI/AHD/CVI capture card

---

- 1 For the TVI/AHD/CVI capture card, before using the qv4l2 to display the video, you must send the V4L2\_CID\_CAMERA\_TYPE id and type of camera value to driver by ioctl, you can reference the source code in the yuan\_app folder.

```
struct v4l2_control s_ctrl;  
s_ctrl.id = V4L2_CID_CAMERA_TYPE;  
s_ctrl.value = 0; // 0:TVI, 1:AHD, 2:CVI  
V4L2_IOCTL(pDevice, VIDIOC_S_CTRL, &s_ctrl);
```

- 2 You can use the scripts(tvi.sh/ahd.sh/cvi.sh) to change the camera type before using the qv4l2.
- 3 If you want to rebuild the app,
  - 3.1 Install XWindows library libxvmc-dev by [sudo apt install libxvmc-dev](#)
  - 3.2 Type “make clean” and “make” in the yuan\_app folder to rebuild app.
- 4 If you want to set the camera type only, does not want to see the XWindows of app, execute the command below to set video0 to ahd type.

```
./LINUXWINDOW -d 0 -t ahd -o
```

or

```
./ahd.sh 0
```

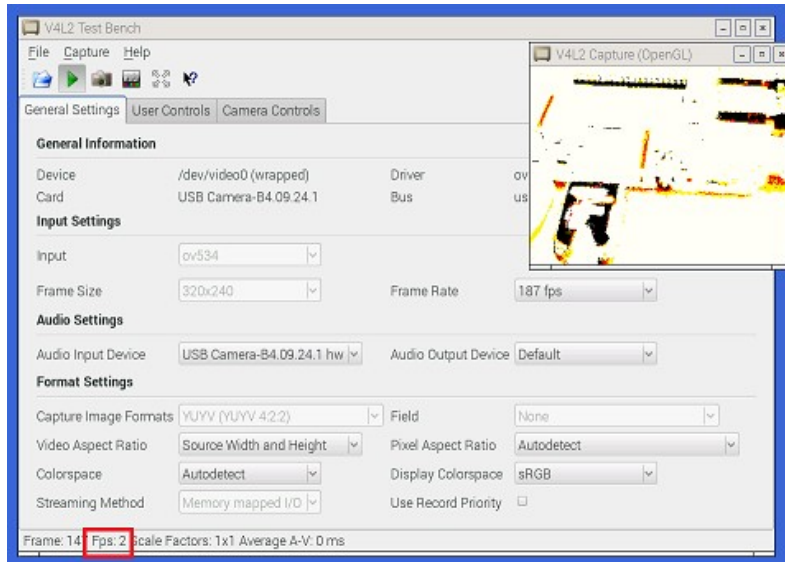
If you want to display the video by XWindow, execute the command below to display video0 which is in ahd type.

```
./LINUXWINDOW -d 0 -t ahd
```

## v4l-utils

- 1 The v4l-utils are a series of packages for handling media devices.
- 2 qv4l2: QT v4l2 control panel application
  - 2.1 It's easy to use, because all the video parameters can be chosen by the GUI.

But the frame rate is not correct on Jetson devices.



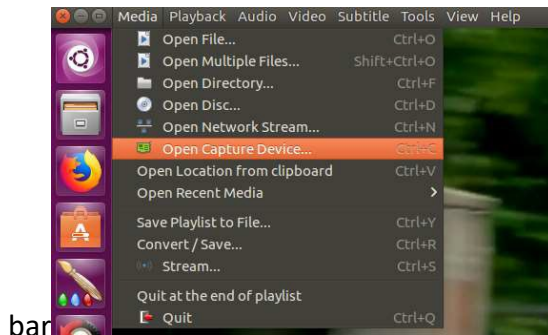
- 2.2 If you are a professional user, you can choose another media app to check the video stream. For example, **gstreamer** or **VLC** ....
- 3 v4l2-ctl: tool to control v4l2 controls from the terminal
  - 3.1 **v4l2-ctl --list-devices**: list all video devices
  - 3.2 **v4l2-d [device index] --all**: display all information available
    - 3.2.1 For example: get /dev/video information by  
**v4l2-ctl -d 0 --all**
  - 3.3 **v4l2 -d [devices index] -i [input index]** : set video input
    - 3.3.1 HDMI: 0
    - 3.3.2 SDI: 4
    - 3.3.3 For example: set /dev/video0 to HDMI input by  
**v4l2-ctl -d 0 -i 0**
- 4 Reference
  - 4.1 V4l-utils: <https://linuxtv.org/wiki/index.php/V4l-utils>

## VLC

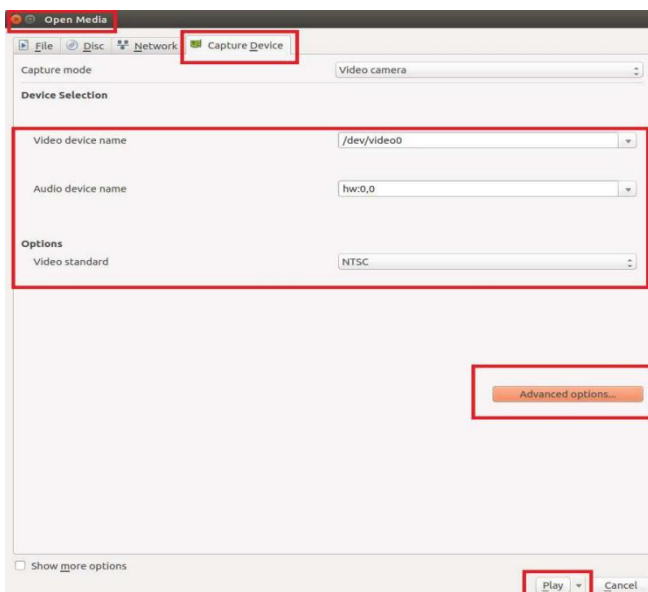
- 1 VLC is a free and open source cross-platform multimedia player and framework that plays most multimedia files as well as DVDs, Audio CDs, VCDs, and various streaming protocols
- 2 Install the VLC
  - 2.1 `$ sudo apt install -y vlc`
- 3 Launch the VLC



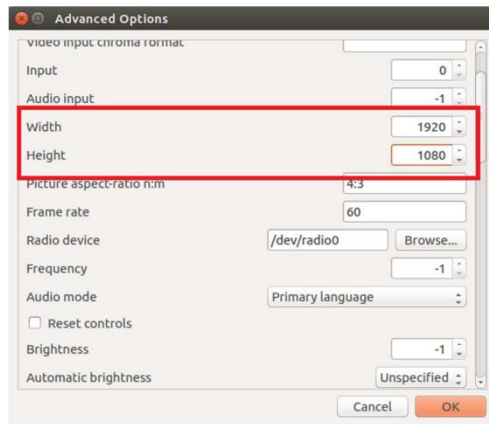
- 4 Select the “Open Capture Device” from menu



- 5 Select the video device name (for example: video0 which get by the v4l-utils command) which you want to open. Select the correct Video standard for your input video source (for example: NTSC).



- 6 Click the “Advance button” to enter the “Advanced Options” dialog, and enter the correct, for example, if your input video is from HDMI, and input is 1080p60, pixel

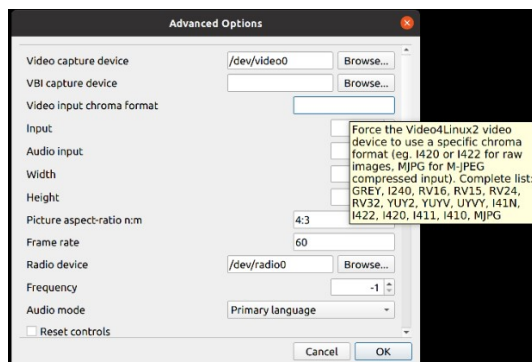


format is YV12, set the options to

- 6.1 Video input chroma format: YV12
- 6.2 Width: 1920
- 6.3 Height: 1080
- 6.4 Input: 0

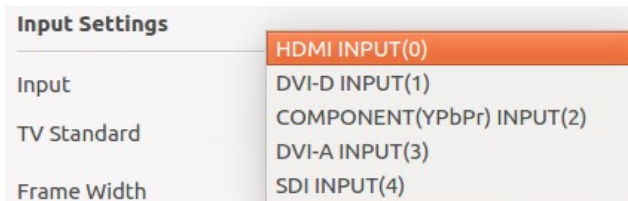
#### 6.4.1 Index of input

- 7 Click “OK” button to finish the configuration
- 8 Click “Play” button of “Open Media” dialog to start video capture, and you should be able to see the video after few seconds.
- 9 If you want to open the second VLC, you can use the terminal to launch the app  
\$ vlc&
- 10 Then select the correct “video device name” (for example video1) to open the second channel of video.
- 11 When the mouse waits few seconds in the edit box of “Video Input chroma format”, a supported tip will show. For example, you can type YV12, YUYV, YUY2, RV24, I420, NV12 ...  
(RV24 is for BGR3 pixel)

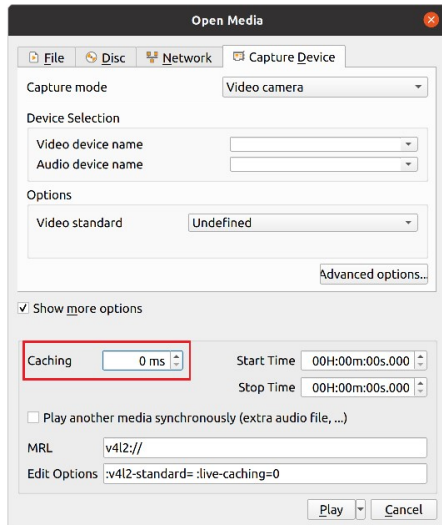


format)

12 index of video input, 0 for HDMI, and 4 for SDI

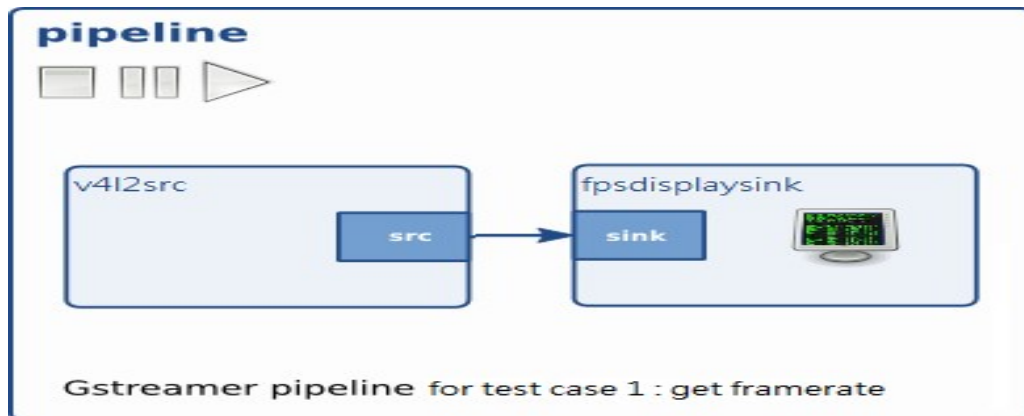


13 If you want to test the latency by vlc, to get real time video stream, set the Caching time to “0 ms”

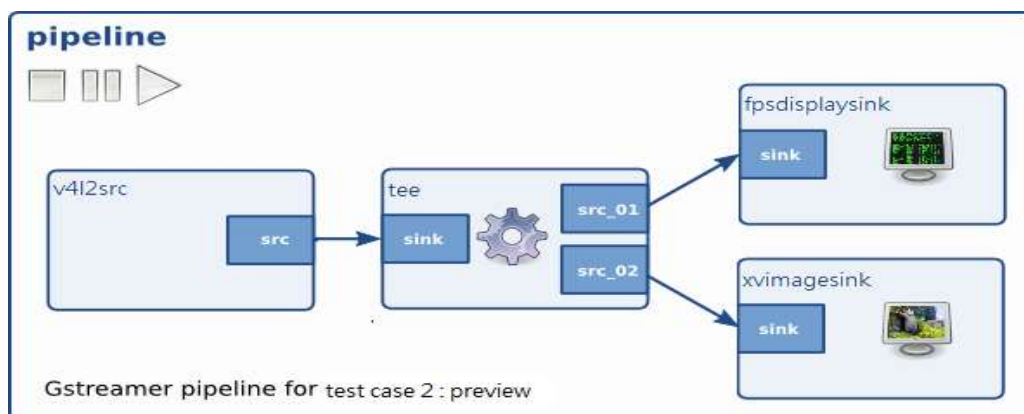


## Internal Frame Rate test cases

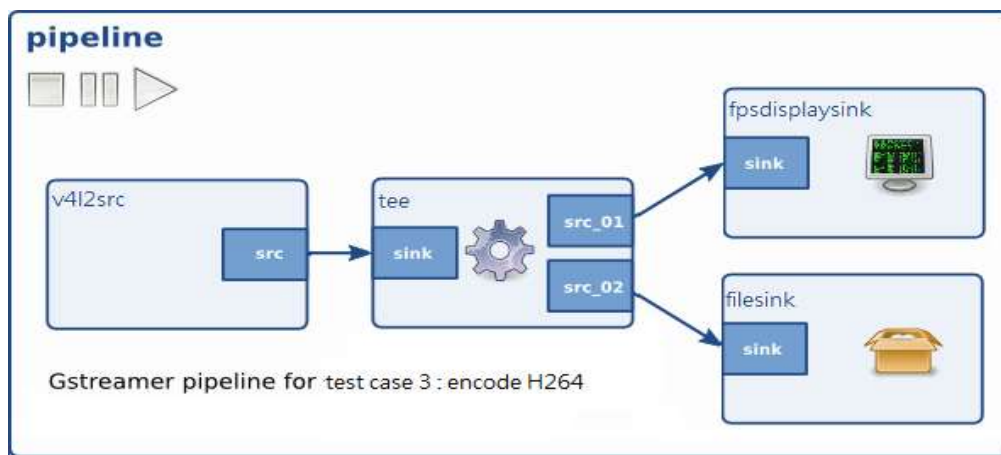
- 0 Baseline – Execute tegrastats to get system loading.
- 1 Get raw data – Use gstreamer command to get video buffer from driver



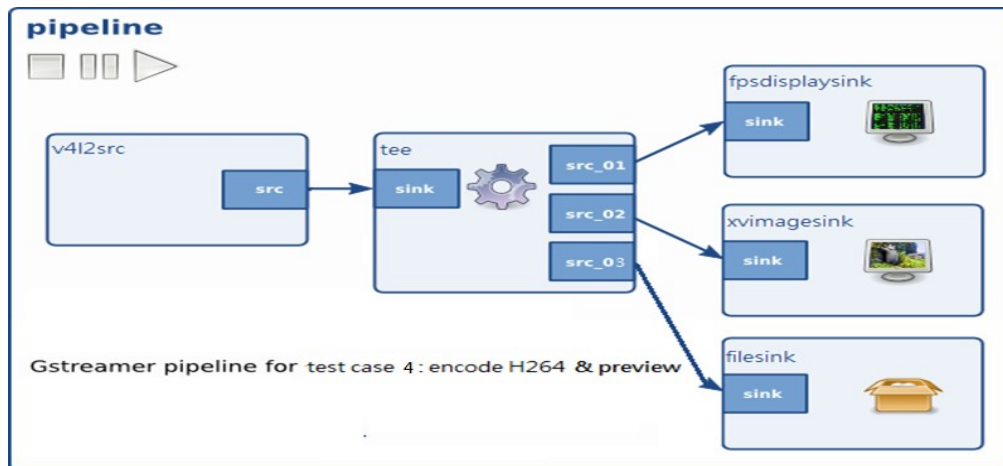
- 2 Preview– Use gstreamer command to get video buffer from driver, and display on screen.



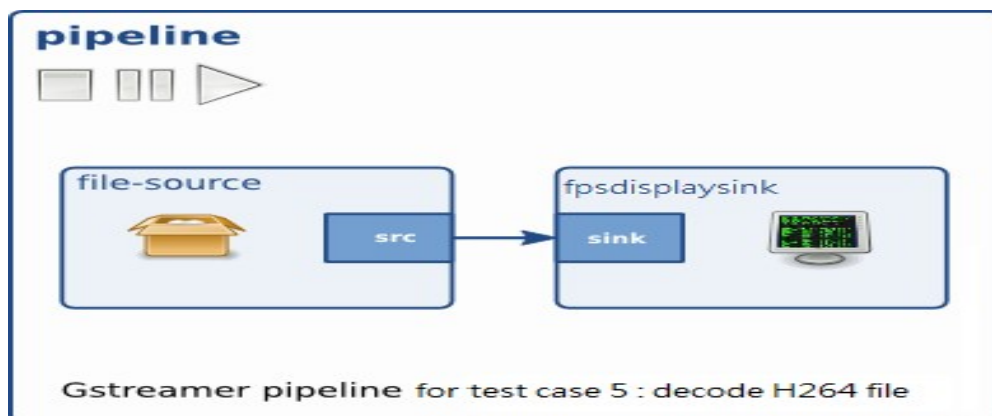
- 3 Encode264 to file: Use gstreamer command to get video buffer from driver, encode to h264 by GPU, and save to file.



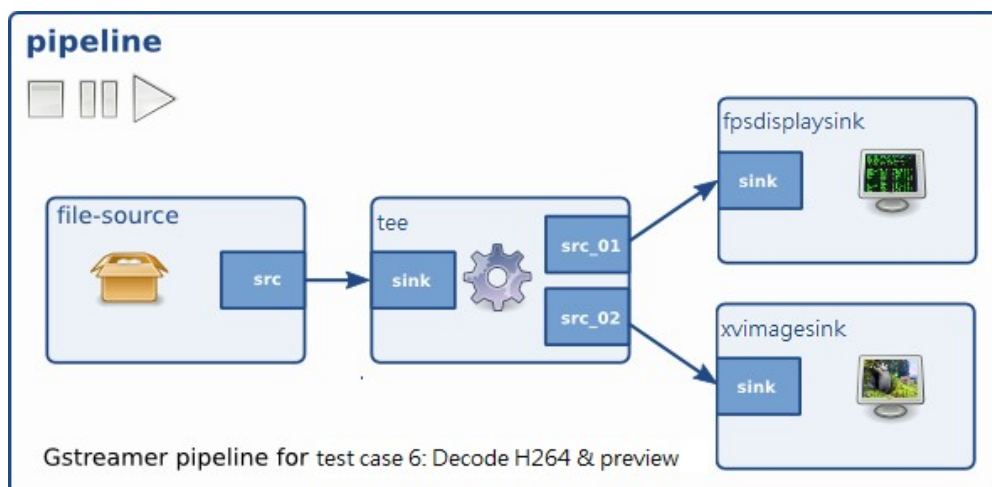
- 4 Preview and Encode 264 to file: Use gstreamer command to get video buffer from driver, encode to h264 by GPU, save to file, and display on screen.



- 5 Decode264 from file: Use gstreamer command to get video from h264 file, and decodes to h264 by GPU.



- 6 Preview and decode264 from file: Use gstreamer command to get video from h264 file, decodes to h264 by GPU, and display on screen.

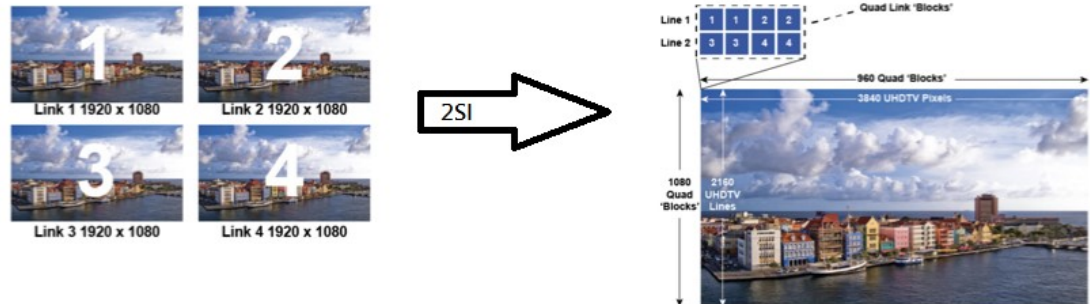


## SC710N1 12G-SDI Capture card

- 1 SC710 SDI supports two quad link: Square Division Quad Split (SQD) and 2 Sample Interleave(2SI). Please reference to <https://blog.rossvideo.com/how-to-go-uhd-with-your-3g-equipment>

- 2 Use v4l2-ctl command to set the quad link

2.1 Set video0 to 2SI : `$ v4l2-ctl -d 0 -i 0x80000004`

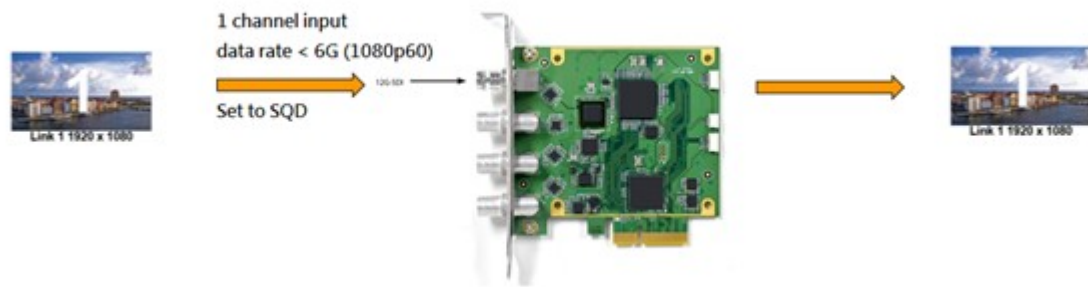


2.2 Set video0 to SQD : `$ v4l2-ctl -d 0 -i 0x80000104`



- 3 When data rate is **lower** than 6Gbps (resolution lower than 1080p60), set the quad link to SQD.  
When data rate is **higher** than 6G, you can set to 2SI or SQD as your requirement.
- 4 [1Ch < 6G]  
When the video is input 1 Channel, and data rate is **lower** than 6G. Only SQD mode is supported.





## 5 [1Ch >= 6G]

When the video input is 1 Channel, and data rate is **higher** than 6G. Set the input to SQD or 2SI mode.

### 5.1 SQD mode

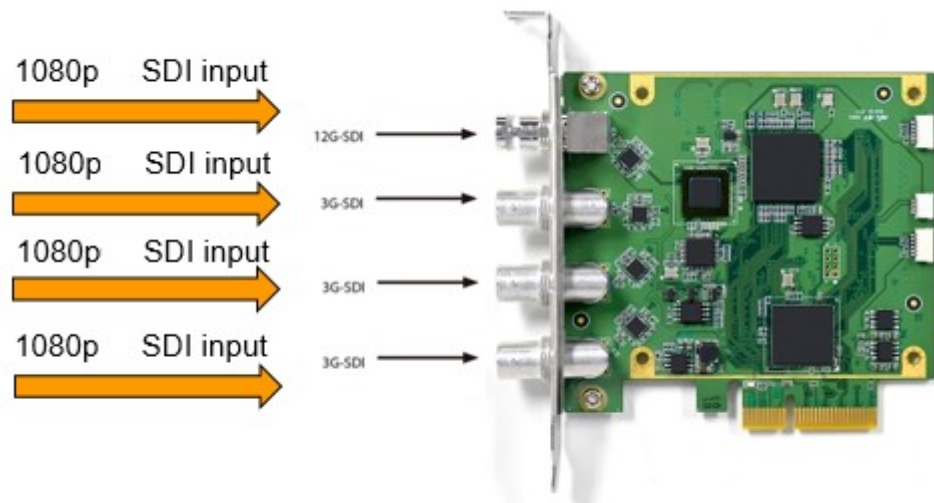


### 5.2 2SI mode



## 6 [4Ch 1080p]

6.1 Driver only support 1080**p** (progressive scan, for example : 1080p60, 1080p50, 1080p30) resolution when the video input is 4 Channels.



6.2 If the resolution is not 1080p, system will reboot when you use vlc to open video device.

6.3 If one of the channels lost signal/connection, the capture image will freeze.

6.4 SQD and 2SI are supported when input is 4 channels with 1080p.

7 vlc\_sc710sdi.sh script sets the video input and then call the vlc to display the video.

7.1 For example:

```
$ ./vlc_sc710sdi.sh 0 sqd
```

This will open the video0, and uses SQD to display the input video.

7.2 Since the vlc can't execute with root account, executes the script with user account.

```
$ ./vlc_sc710sdi.sh 1 2si
```

This will open the video1, and uses 2SI to display the input video.

8 Pixel format supported

8.1 Test environment

8.1.1 Test device: Xavier

8.1.2 Kernel: 5.10.65-tegra

8.1.3 Xilinx: 0x20190816

8.1.4 Lattice: 0x220111

8.1.5 MCU: 0d201125

## 8.2 MMAP

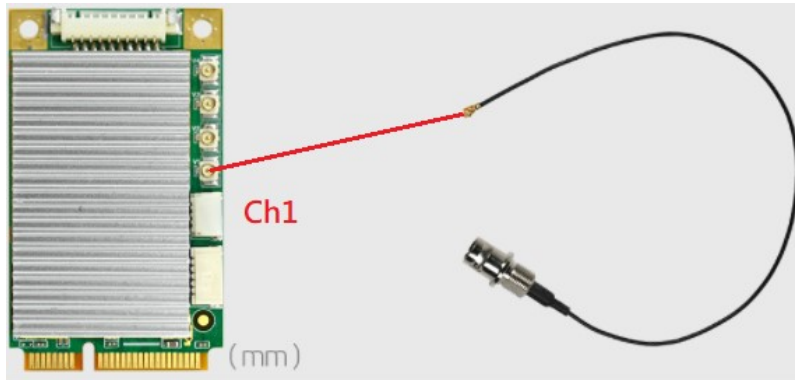
No	Resolution	YUYV	NV12	YV12	YU12
1	480i60	0	0	0	0
2	576i50	0	0	0	0
3	720p50	0	0	0	0
4	720p60	0	0	0	0
5	1080i50	0	0	0	0
6	1080i60	0	0	0	0
7	1080p24	0	0	0	0
8	1080p30	0	0	0	0
9	1080p50	0	0	0	0
10	1080p60	0	0	0	0
11	4K UHD-1p24	0	0	0	0
12	4K UHD-1p25	0	0	0	0
13	4K UHD-1p30	0	0	0	0
14	4K UHD-1p50	0	0	0	0
15	4K UHD-1p60	0	0	0	0
16	DCI 4Kp24	0	0	0	0
17	DCI 4Kp25	0	0	0	0
18	DCI 4Kp30	0	0	0	0
19	DCI 4Kp50	0	0	0	0
20	DCI 4Kp60	0	0	0	0

### 8.3 USERPTR

No	Resolution	NV12
1	480i60	X
2	576i50	X
3	720p50	O
4	720p60	O
5	1080i50	X
6	1080i60	X
7	1080p24	O
8	1080p30	O
9	1080p50	O
10	1080p60	O
11	4K UHD-1p24	O
12	4K UHD-1p25	O
13	4K UHD-1p30	O
14	4K UHD-1p50	O
15	4K UHD-1p60	O
16	DCI 4Kp24	O
17	DCI 4Kp25	O
18	DCI 4Kp30	O
19	DCI 4Kp50	O
20	DCI 4Kp60	O

## SC400N4 MC TVI Capture card

- SC400N4 MC TVI is a 4CH TVI 1080p30 capture card, the following output pixel formats are tested with driver v1058.



No	Resolution	NV12	YV12	YU12
1	720p25	O	O	O
2	720p30	O	O	O
3	1080p25	O	O	O
4	1080p30	O	O	O

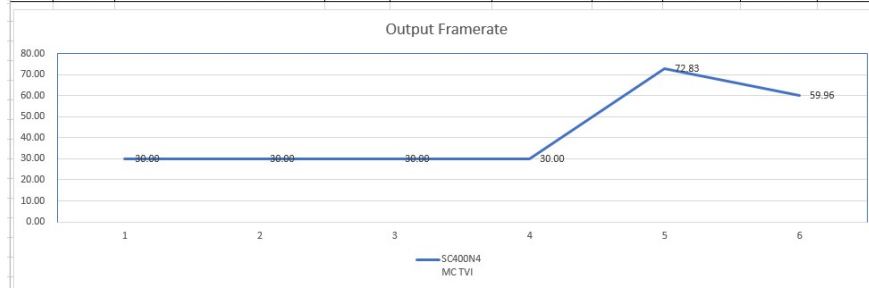
- The result of internal frame rate test

### 2.1 Test environment

- Video input :1080p30x4(channels)
- Video output format: NV12(4:2:0)
- Nvidia release version : L4T R32.4.4
- Driver version: V1.1.0.120.926\_VB2
- Xavier NX

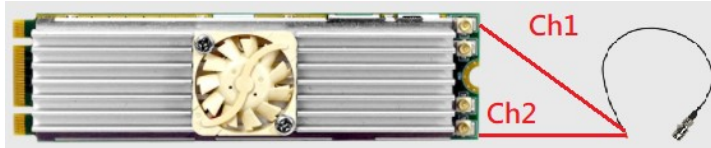
### 2.2 Result

Card Index	Test case Index	Item	Capture Card	Input/Format/Channels	L4T Version Drive Version nvpmodel	RAM used	CPU used	GPU Used	Framerate
6	0	Baseline	SC400N4 MC TVI	1080p30 NV12 4	R32.4.4 v926 2(15W/6Ccore)	33.70	1.81	0.15	0.00
	1	Get raw data				34.63	12.20	19.51	30.00
	2	Preview				35.58	69.87	38.33	30.00
	3	Encode264 to file				41.59	82.97	29.95	30.00
	4	Preview and Encode 264 to file				37.93	46.71	34.64	30.00
	5	Decode264 from file				37.42	28.58	19.11	72.83
	6	Preview and decode264 from file				38.83	36.84	39.93	59.96



## SC400N2 M2 SDI Capture card

- 1 SC400N2 M2 SDI is a 2CH 1080p60 capture card, the following output pixel formats are tested with driver v1058.



No	Resolution	YUYV	NV12	YV12	YU12
1	720p50	O	O	O	O
2	720p60	O	O	O	O
3	1080i50	O	O	O	O
4	1080i60	O	O	O	O
5	1080p24	O	O	O	O
6	1080p50	O	O	O	O
7	1080p60	O	O	O	O

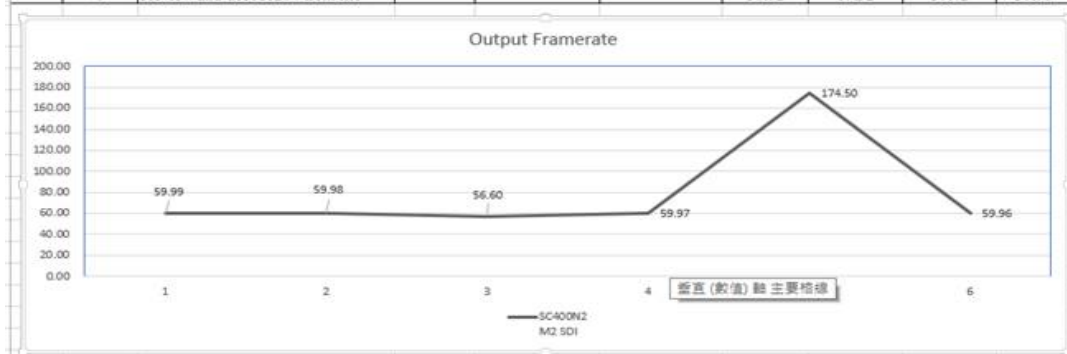
- 3 The result of internal frame rate test

### 3.1 Test environment

- 3.1.1 Video input :1080p60x2(channels)
- 3.1.2 Video output format: YUYV(4:2:2)
- 3.1.3 Nvidia release version: L4T R32.6.1
- 3.1.4 Driver version: V1.1.0.120.948\_VB2
- 3.1.5 Xavier NX

## 2 Result

Card Index	Test case Index	Item	Capture Card	Input/Format/Channels	L4T Version Drive Version nvpmode1	RAM used	CPU used	GPU Used	Framerate
8	0	Baseline	SC400N2 M2 SDI	1080p60 YUYV 2	R32.6.1 v948 8(QW/6Ccore)	30.51	1.29	0.75	0.00
	1	Get raw data				31.12	9.49	12.22	59.99
	2	Preview				32.26	23.31	35.58	59.98
	3	Encode264 to file				36.67	20.84	4.76	56.60
	4	Preview and Encode 264 to file				39.06	40.52	35.6	59.97
	5	Decode264 from file				36.99	18.93	6.84	174.50
	6	Preview and decode264 from file				38.82	19.52	39.42	59.96



## SC710N1 M2 HDMI Capture card

1. SC710N1 M2 HDMI is a 4K capture card, the following output pixel formats are tested with driver v1104

No	Resolution	YUYV	NV12	YV12	YU12	RGB3	BGR3	UYVY
1	480p60	O	O	O	O	O	O	O
2	480i59	O	O	O	O	X	X	O
3	576p50	O	O	O	O	O	O	O
4	576i50	O	O	O	O	X	X	O
5	720p50	O	O	O	O	O	O	O
6	720p60	O	O	O	O	O	O	O
7	1080i50	O	O	O	O	X	X	O
8	1080i60	O	O	O	O	X	X	O
9	1080p24	O	O	O	O	O	O	O
10	1080p25	O	O	O	O	O	O	O
11	1080p30	O	O	O	O	O	O	O
12	1080p50	O	O	O	O	O	O	O
13	1080p60	O	O	O	O	O	O	O
14	4K UHD-1p24	O	O	O	O	O	O	O
15	4K UHD-1p25	O	O	O	O	O	O	O
16	4K UHD-1p30	O	O	O	O	O	O	O
17	4K UHD-1p50	O	O	O	O	O	O	O
18	4K UHD-1p60	O	O	O	O	O	O	O
19	DCI 4Kp24	O	O	O	O	O	O	O
20	DCI 4Kp25	O	O	O	O	O	O	O
21	DCI 4Kp30	O	O	O	O	O	O	O
22	DCI 4Kp50	O	O	O	O	O	O	O
23	DCI 4Kp60	O	O	O	O	O	O	O
24	1280x2880p33	O	X	X	X	O	O	O

- 2 The result of internal frame rate test

### 2.1 Test environment

2.1.1 Video input :3840x2160p60

2.1.2 Video output format: YUYV (4:2:2)

2.1.3 Nvidia release version: L4T R32.4.4

2.1.4 Driver version: V1.1.0.120.926\_VB2

2.1.5 Xavier NX

## 2. Result

Item	Capture Card	Input/Format/Channels	L4T Version Drive Version nvpmmodel	RAM used	CPU used	GPU Used	Framerate
Baseline	SC710N1 M2 HDV	4Kp60 YUYV 1	R32.4.4 v926 2(15W/6Ccore)	28.80	0.62	0.22	0.00
Get raw data				29.52	5.16	5.84	60.00
Preview				30.15	30.11	33.06	60.00
Encode264 to file				39.93	18.96	2.00	58.17
Preview and Encode 264 to file				39.44	29.81	50.15	57.45
Decode264 from file				37.46	7.26	5.90	81.28
Preview and decode264 from file				37.92	13.49	44.66	59.97

