Overview

- This is a raspberry pi HDMI to CSI-2 Module with Toshiba TC358743XBG chip, HDMI input supports up to 1080p25fps.

According to the customer feedback, this module does not support **OctoPi**.

Packing List

- 1 x Raspberry Pi HDMI to CSI-2 Module
- 1 x FFC Cable(14cm/5.51inch length) for Pi4/Pi3b+/Pi3b
- 1X FFC Cable(14cm/5.51inch length) for Pi Zero
- 1x Heatsink Sets

Features

- Input signal: HDMI;
- Output signal: CSI;
- HDMI Input: 720p50\720p60\1080i50\1080p25
- Function: HDMI to CSI-2
- Limitation: HDMI input supports up to 1080p25fps
- Usage: Same as standard Raspberry Pi camera
- Chip: Toshiba TC358743XBG
- Compatible with: Raspberry Pi 4B/3B+/3B/2B/B+/3A+/Pi Zero/Zero W

How to check whether this module is driven correctly?

step1. Check this module if be driven?

After connecting all the cables, power on the Raspberry Pi, the C779 indicator light is normally green, and after opening the Raspberry Pi terminal, enter the following command:

```
ls /dev
```

then check if video0 appears. If it appears correctly, it means that the module has been successfully driven and is working normally.
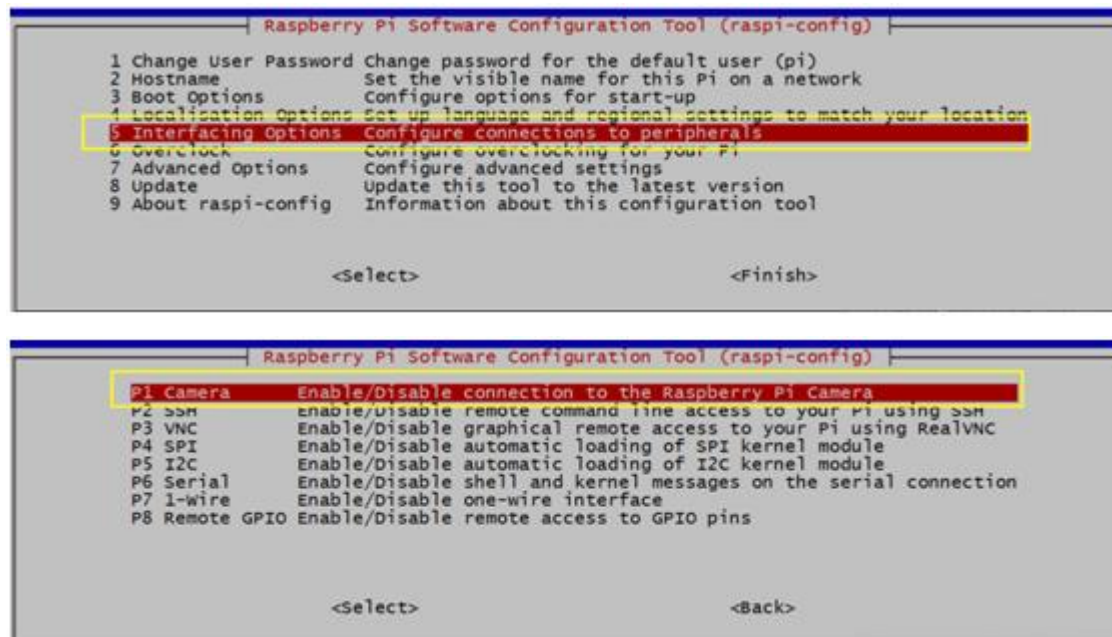
Please try the following oreration if you can't find 'video0'

step2. update & upgrade the raspberry pi system (It may be necessary to update the software source according to your country, and this will take a long time)

```
sudo apt-get update
sudo apt-get upgrade
```

step3. Open camera module via raspi-config command;

```
sudo raspi-config
```


Enable camera & reboot the raspberry pi.

step4: goto step1

step5: if you still can't find 'video0' file in the /dev path, please try the following methods:

A: Confirm whether the HDMI input device has a signal, (you can test whether it is displayed normally by connecting to the screen)

B. Confirm whether the resolution and frame rate of the HDMI input device are below the maximum input resolution and frame rate(720p50\720p60\1080i50\1080p25);

C: We recommend that you download the lastest official image of Raspberry pi if you still don't solve your questions. Download URL: https://www.raspberrypi.org/downloads/raspberry-pi-os/

Driver configuration

1、edit / boot / config.txt, and add the line to the end of the file. Sudo is required.

dtoverlay=tc358743

2、Use "dmesg | grep CMA" to check the amount of memory allocated to the CMA heap.

pi@raspberrypi:~ $ dmesg | grep cma

[0.000000] cma: Reserved 256 MiB at 0x000000001ec00000

If the report has less than 96MB of files assigned to CMA, edit / boot / config.txt and add.

dtoverlay=cma,cma-128

3、Restart raspberry pi.If all goes well, you should get a / dev / video0 device.Using "arecord - L" will tell you whether the sound card of tc358743 is loaded correctly.

Use raspistill to take photos

Raspistil function: run the camera at a specified time, and capture jpg images if necessary.

Format:raspistill [options]

Main image parameters and commands:

-?, --Help: help document

-w. -- width: set the image width < size > width

-h. -- height: set image height < size > height

-q. -- Quality: set JPEG quality < 0 to 100 >

-r. -- raw: add raw raw Bayer data to JPEG metadata

-o. -- output: output file name < file name >,

-l. -- Latest: link the latest complete image to the specified file < file name >

-v. -- verb: output details when running the camera

-t. -- timeout: the time delay is specified when photographing and closing. If it is not specified, the default is 5S

-e. -- encoding: encode to output the specified format file (jpg, BMP, GIF, PNG)

-TL, - timelapse: take one picture every < MS > at intervals

-k. -- keypress: press the key to trigger, press' Enter 'to take a picture, press' x' and then 'Enter' to exit

-s. - signal: signal triggered, waiting for another process signal to take a picture

-GC, - glcapture: capture GL frame buffer instead of camera image

-Set, - settings: retrieve the camera settings and write them to stdout

-CS, - camselect: select camera device < digital >, default 0

The above is just a list of some common commands. More detailed commands can be used by Baidu, Google or viewing help documents. Here are some specific examples:

After a delay of two seconds (in milliseconds), take a picture and save it as

image.jpg.raspistill -t 2000 -o image.jpg

Take a custom size photo.raspistill -t 2000 -o image.jpg -w 640 -h 480

Record a 5 second video clip (1080p25) using the default settings.raspivid -t 5000 -o

video.h264

Save to file a 5 second encoded camera stream image raspivid - t 5000 - O - >

My_file.h264

FAQ

Q1: What to do if the module can't work normally?

A1:

- 1. Please use monitor to test, DO NOT use VNC.(Customer Feedback: During the VNC it spend some of gpu and there is not enough for camera.Based on RPi 4 + Hawkeye Firefly Mini)

- 2. First the HDMI device should be plugged in and have signal output before the Raspberry Pi is started.

- 3. Please check if there is a video related file in the /dev file.

- 4. Please provide us more details for us to confirm the issue.

a. First please send us your **order number** and tell us "Call command" "Input Device" and "Using Device".

b. What is the HDMI input device, resolution and frequency?

c. Which version of Raspberry Pi you use?

d. What is the specific calling command?

e. What is the terminal error notification?

Q2: Some python sample code

A2: The HDMI source supported by Raspberry Pi with the module is 720p/50fps,720p/60fps,1080i/50fps,1080p/24fps,1080p/25fps. Lower resolution is also working.

This is the python code used in my video.The Pi uses an official image with no other changes.

```
from picamera import PiCamera
from time import
camera = PiCamera()
camera.start_preview()
sleep(1000)
camera.stop_preview()
```

Q3:Customer Feedback 1 for your reference(https://www.amazon.com/gp/product/B0899L6ZXZ#customerReviews)

A3: If you want to use your Raspberry Pi for HDMI capture, this is the only device I'm aware of that will do it. Furthermore you can do some powerful things that would normally require equipment costing many hundreds of dollars. For example, using always-on camera preview and a few lines of Python code you can easily do image flipping, rotation, and rudimentary scaling. Note that you can't adjust color or exposure, and audio is not passed through.

I've tried this with a variety of HDMI devices; half of them work perfectly, half of them don't work at all.

Things that worked just fine:

- GoPro Hero2

- Generic no-name HDMI camera

- OREI HD-102 1x2 HDMI splitter with a Google Chromecast attached to it (but see below)

There's a downside, though. I could not get any of these to work:

- Canon 6D, which causes a "PiCameraMMALError: Failed to enable connection: Out of resources" error

- Blackmagic ATEM Mini, which produces a scrambled picture

- Google Chromecast, because I don't think this device supports HDCP; but it works fine if you strip off the HDCP.

This HDMI input module does what I care about (capturing my generic HDMI camera), but it failed at some things that thankfully I didn't need it to do. Your use case may vary so don't be surprised if some HDMI devices don't work with it.

Q4: Will this work with i2s hats such as hifiberry amp2?

A4: Since this item don't use any GPIO, so we think that it can work with I2S hats.