Raspberry Pi Camera Module

This document describes the use of the four Raspberry Pi camera applications, as of 30 April 2020.

There are four applications provided: `raspistill`, `raspivid`, `raspiyuv` and `raspividyuv`. `raspistill` and `raspiyuv` are very similar and are intended for capturing images; `raspivid` and `raspvidyuv` are for capturing video.

All the applications are driven from the command line, and written to take advantage of the MMAL API which runs over OpenMAX. The MMAL API provides an easier to use system than that presented by OpenMAX. Note that MMAL is a Broadcom-specific API used only on VideoCore 4 systems.

The applications use up to four OpenMAX (MMAL) components: camera, preview, encoder, and null_sink. All applications use the camera component; `raspistill` uses the Image Encode component; `raspivid` uses the Video Encode component; and `raspiyuv` and `raspividyuv` don't use an encoder, and sends their YUV or RGB output directly from the camera component to file.

The preview display is optional, but can be used full-screen or directed to a specific rectangular area on the display. If preview is disabled, the null_sink component is used to 'absorb' the preview frames. The camera must produce preview frames even if these aren't required for display, as they're used for calculating exposure and white balance settings.

In addition, it's possible to omit the filename option (in which case the preview is displayed but no file is written), or to redirect all output to stdout.

Command line help is available by typing just the application name in the command line.

Setting up

See [Camera Setup](#).

Troubleshooting

If the Camera Module isn't working correctly, there are number of things to try:

- Is the ribbon cable attached to the Camera Serial Interface (CSI), not the Display Serial Interface (DSI)? The ribbon connector will fit into either port. The Camera port is located near the HDMI connector.

- Are the ribbon connectors all firmly seated, and are they the right way round? They must be straight in their sockets.

- Is the Camera Module connector, between the smaller black Camera Module itself and the PCB, firmly attached? Sometimes this connection can come loose during transit or when putting the Camera Module in a case. Using a fingernail, flip up the connector on the PCB, then reconnect it with gentle pressure. It engages with a very slight click. Don't force it; if it doesn't engage, it's probably slightly misaligned.

- Have `sudo apt update` and `sudo apt full-upgrade` been run?

- Has `raspi-config` been run and the Camera Module enabled?

- Is your power supply sufficient? The Camera Module adds about 200-250mA to the power requirements of your Raspberry Pi.

If things are still not working, try the following:

- `Error : raspistill/raspivid command not found` . This probably means your update/upgrade failed in some way. Try it again.

- `Error : ENOMEM` . The Camera Module is not starting up. Check all connections again.

- `Error : ENOSPC` . The Camera Module is probably running out of GPU memory. Check `config.txt` in the /boot/ folder. The `gpu_mem` option should be at least 128. Alternatively, use the Memory Split option in the Advanced section of `raspi-config` to set this.

- If you've checked all the above issues and the Camera Module is still not working, try posting on our forums for more help.

Common command line options

Preview window

```
--preview,     -p         Preview window settings <'x,y,w,h'>
```

Allows the user to define the size of the preview window and its location on the screen. Note this will be superimposed over the top of any other windows/graphics.

```
--fullscreen,    -f         Fullscreen preview mode
```

Forces the preview window to use the whole screen. Note that the aspect ratio of the incoming image will be retained, so there may be bars on some edges.

```
--nopreview,     -n         Do not display a preview window
```

Disables the preview window completely. Note that even though the preview is disabled, the camera will still be producing frames, so will be using power.

```
--opacity,     -op         Set preview window opacity
```

Sets the opacity of the preview windows. 0 = invisible, 255 = fully opaque.

Camera control options

```
--sharpness,     -sh         Set image sharpness (-100 - 100)
```

Sets the sharpness of the image. 0 is the default.

```
--contrast,     -co         Set image contrast (-100 - 100)
```

Sets the contrast of the image. 0 is the default.

```
    --brightness,    -br         Set image brightness (0 - 100)
```

Sets the brightness of the image. 50 is the default. 0 is black, 100 is white.

```
    --saturation,    -sa         Set image saturation (-100 - 100)
```

Sets the colour saturation of the image. 0 is the default.

```
    --ISO,    -ISO         Set capture ISO (100 - 800)
```

Sets the ISO to be used for captures.

```
    --vstab,    -vs         Turn on video stabilisation
```

In video mode only, turns on video stabilisation.

```
    --ev,    -ev         Set EV compensation (-10 - 10)
```

Sets the EV compensation of the image. Default is 0.

```
    --exposure,    -ex         Set exposure mode
```

Possible options are:

- auto: use automatic exposure mode
- night: select setting for night shooting
- nightpreview:
- backlight: select setting for backlit subject
- spotlight:
- sports: select setting for sports (fast shutter etc.)
- snow: select setting optimised for snowy scenery
- beach: select setting optimised for beach
- verylong: select setting for long exposures
- fixedfps: constrain fps to a fixed value
- antishake: antishake mode

- fireworks: select setting optimised for fireworks

Note that not all of these settings may be implemented, depending on camera tuning.

```
    --flicker, -fli        Set flicker avoidance mode
```

Set a mode to compensate for lights flickering at the mains frequency, which can be seen as a dark horizontal band across an image. Flicker avoidance locks the exposure time to a multiple of the mains flicker frequency (8.33ms for 60Hz, or 10ms for 50Hz). This means that images can be noisier as the control algorithm has to increase the gain instead of exposure time should it wish for an intermediate exposure value. `auto` can be confused by external factors, therefore it is preferable to leave this setting off unless actually required.

Possible options are:

- off: turn off flicker avoidance
- auto: automatically detect mains frequency
- 50hz: set avoidance at 50Hz
- 60hz: set avoidance at 60Hz

```
    --awb,    -awb         Set Automatic White Balance (AWB) mode
```

Modes for which colour temperature ranges (K) are available have these settings in brackets.

- off: turn off white balance calculation
- auto: automatic mode (default)
- sun: sunny mode (between 5000K and 6500K)
- cloud: cloudy mode (between 6500K and 12000K)
- shade: shade mode
- tungsten: tungsten lighting mode (between 2500K and 3500K)
- fluorescent: fluorescent lighting mode (between 2500K and 4500K)
- incandescent: incandescent lighting mode
- flash: flash mode
- horizon: horizon mode
- greyworld: Use this on the NoIR camera to fix incorrect AWB results due to the lack of the IR filter.

Note that not all of these settings may be implemented, depending on camera type.

```
    --imxfx,    -ifx        Set image effect
```

Set an effect to be applied to the image:

- none: no effect (default)
- negative: invert the image colours
- solarise: solarise the image
- posterise: posterise the image
- whiteboard: whiteboard effect
- blackboard: blackboard effect
- sketch: sketch effect
- denoise: denoise the image
- emboss: emboss the image
- oilpaint: oil paint effect
- hatch: hatch sketch effect
- gpen: graphite sketch effect
- pastel: pastel effect
- watercolour: watercolour effect
- film: film grain effect
- blur: blur the image
- saturation: colour saturate the image
- colourswap: not fully implemented
- washedout: not fully implemented
- colourpoint: not fully implemented
- colourbalance: not fully implemented
- cartoon: not fully implemented

Note that not all of these settings may be available in all circumstances.

```
    --colfx,    -cfx        Set colour effect <U:V>
```

The supplied U and V parameters (range 0 - 255) are applied to the U and Y channels of the image. For example, --colfx 128:128 should result in a monochrome image.

```
    --metering,    -mm        Set metering mode
```

Specify the metering mode used for the preview and capture:

- average: average the whole frame for metering
- spot: spot metering
- backlit: assume a backlit image
- matrix: matrix metering

```
--rotation,    -rot           Set image rotation (0 - 359)
```

Sets the rotation of the image in the viewfinder and resulting image. This can take any value from 0 upwards, but due to hardware constraints only 0, 90, 180, and 270 degree rotations are supported.

```
--hflip,    -hf          Set horizontal flip
```

Flips the preview and saved image horizontally.

```
--vflip,    -vf          Set vertical flip
```

Flips the preview and saved image vertically.

```
--roi,    -roi           Set sensor region of interest
```

Allows the specification of the area of the sensor to be used as the source for the preview and capture. This is defined as x,y for the top-left corner, and a width and height, with all values in normalised coordinates (0.0 - 1.0). So, to set a ROI at halfway across and down the sensor, and a width and height of a quarter of the sensor, use:

```
-roi 0.5,0.5,0.25,0.25
```

```
--shutter,    -ss           Set shutter speed/time
```

Sets the shutter open time to the specified value (in microseconds). Shutter speed limits are as follows:

Camera Version	Max (microseconds)
V1 (OV5647)	6000000 (i.e. 6s)

Camera Version	Max (microseconds)
V2 (IMX219)	10000000 (i.e. 10s)
HQ (IMX477)	200000000 (i.e. 200s)

Using values above these maximums will result in undefined behaviour.

```
    --drc,    -drc           Enable/disable dynamic range compression
```

DRC changes the images by increasing the range of dark areas, and decreasing the brighter areas. This can improve the image in low light areas.

- off
- low
- med
- high

By default, DRC is off.

```
    --stats,    -st           Use stills capture frame for image statistics
```

Force recomputation of statistics on stills capture pass. Digital gain and AWB are recomputed based on the actual capture frame statistics, rather than the preceding preview frame.

```
    --awbgains,    -awbg
```

Sets blue and red gains (as floating point numbers) to be applied when `-awb off` is set e.g. -awbg 1.5,1.2

```
    --analoggain,    -ag
```

Sets the analog gain value directly on the sensor (floating point value from 1.0 to 8.0 for the OV5647 sensor on Camera Module V1, and 1.0 to 12.0 for the IMX219 sensor on Camera Module V2 and the IMX447 on the HQ Camera).

```
    --digitalgain,    -dg
```

Sets the digital gain value applied by the ISP (floating point value from 1.0 to 64.0, but values over about 4.0 will produce overexposed images)

```
    --mode,    -md
```

Sets a specified sensor mode, disabling the automatic selection. Possible values depend on the version of the Camera Module being used:

Version 1.x (OV5647)

Mode	Size	Aspect Ratio	Frame rates	FOV	Binning
0	automatic selection				
1	1920x1080	16:9	1-30fps	Partial	None
2	2592x1944	4:3	1-15fps	Full	None
3	2592x1944	4:3	0.1666-1fps	Full	None
4	1296x972	4:3	1-42fps	Full	2x2
5	1296x730	16:9	1-49fps	Full	2x2
6	640x480	4:3	42.1-60fps	Full	2x2 plus skip
7	640x480	4:3	60.1-90fps	Full	2x2 plus skip

Version 2.x (IMX219)

Mode	Size	Aspect Ratio	Frame rates	FOV	Binning
0	automatic selection				
1	1920x1080	16:9	0.1-30fps	Partial	None
2	3280x2464	4:3	0.1-15fps	Full	None
3	3280x2464	4:3	0.1-15fps	Full	None
4	1640x1232	4:3	0.1-40fps	Full	2x2
5	1640x922	16:9	0.1-40fps	Full	2x2

Mode	Size	Aspect Ratio	Frame rates	FOV	Binning
6	1280x720	16:9	40-90fps	Partial	2x2
7	640x480	4:3	40-200fps[1]	Partial	2x2

[1]For frame rates over 120fps, it is necessary to turn off automatic exposure and gain control using `-ex off`. Doing so should achieve the higher frame rates, but exposure time and gains will need to be set to fixed values supplied by the user.

HQ Camera

Mode	Size	Aspect Ratio	Frame rates	FOV	Binning/Scaling
0	automatic selection				
1	2028x1080	169:90	0.1-50fps	Partial	2x2 binned
2	2028x1520	4:3	0.1-50fps	Full	2x2 binned
3	4056x3040	4:3	0.005-10fps	Full	None
4	1012x760	4:3	50.1-120fps	Full	4x4 Scaled

```
    --camselect,    -cs
```

Selects which camera to use on a multi-camera system. Use 0 or 1.

```
    --annotate,    -a        Enable/set annotate flags or text
```

Adds some text and/or metadata to the picture.

Metadata is indicated using a bitmask notation, so add them together to show multiple parameters. For example, 12 will show time(4) and date(8), since 4+8=12.

Text may include date/time placeholders by using the '%' character, as used by strftime.

Value	Meaning	Example Output

Value	Meaning	Example Output
-a 4	Time	20:09:33
-a 8	Date	10/28/15
-a 12	4+8=12 Show the date(4) and time(8)	20:09:33 10/28/15
-a 16	Shutter Settings	
-a 32	CAF Settings	
-a 64	Gain Settings	
-a 128	Lens Settings	
-a 256	Motion Settings	
-a 512	Frame Number	
-a 1024	Black Background	
-a "ABC %Y-%m-%d %X"	Show some text	ABC %Y-%m-%d %X
-a 4 -a "ABC %Y-%m-%d %X"	Show custom formatted date/time	ABC 2015-10-28 20:09:33
-a 8 -a "ABC %Y-%m-%d %X"	Show custom formatted date/time	ABC 2015-10-28 20:09:33

```
    --annotateex,    -ae        Set extra annotation parameters
```

Specifies annotation size, text colour, and background colour. Colours are in hex YUV format.

Size ranges from 6 - 160; default is 32. Asking for an invalid size should give you the default.

Example	Explanation
-ae 32,0xff,0x808000 -a "Annotation text"	gives size 32 white text on black background
-ae 10,0x00,0x8080FF -a "Annotation text"	gives size 10 black text on white background

```
    --stereo,    -3d
```

Select the specified stereo imaging mode; `sbs` selects side-by-side mode, `tb` selects top/bottom mode; `off` turns off stereo mode (the default).

```
    --decimate,    -dec
```

Halves the width and height of the stereo image.

```
    --3dswap,    -3dswap
```

Swaps the camera order used in stereoscopic imaging; NOTE: currently not working.

```
    --settings,    -set
```

Retrieves the current camera settings and writes them to stdout.

Application-specific settings

raspistill

```
    --width,    -w          Set image width <size>

    --height,    -h          Set image height <size>

    --quality,    -q          Set JPEG quality <0 to 100>
```

Quality 100 is almost completely uncompressed. 75 is a good all-round value.

```
    --raw,    -r          Add raw Bayer data to JPEG metadata
```

This option inserts the raw Bayer data from the camera into the JPEG metadata.

```
    --output,    -o          Output filename <filename>
```

Specifies the output filename. If not specified, no file is saved. If the filename is '-', then all output is sent to stdout.

```
    --latest,    -l          Link latest frame to filename <filename>
```

Makes a file system link under this name to the latest frame.

```
    --verbose,     -v           Output verbose information during run
```

Outputs debugging/information messages during the program run.

```
    --timeout,     -t           Time before the camera takes picture and shuts down
```

The program will run for the specified length of time, entered in milliseconds. It then takes the capture and saves it if an output is specified. If a timeout value is not specified, then it is set to 5 seconds (-t 5000). Note that low values (less than 500ms, although it can depend on other settings) may not give enough time for the camera to start up and provide enough frames for the automatic algorithms like AWB and AGC to provide accurate results.

If set to 0, the preview will run indefinitely, until stopped with CTRL-C. In this case no capture is made.

```
    --timelapse,    -tl         time-lapse mode
```

The specific value is the time between shots in milliseconds. Note that you should specify `%04d` at the point in the filename where you want a frame count number to appear. So, for example, the code below will produce a capture every 2 seconds, over a total period of 30s, named `image0001.jpg`, `image0002.jpg` and so on, through to `image0015.jpg`.

```
 -t 30000 -tl 2000 -o image%04d.jpg
```

Note that the `%04d` indicates a 4-digit number, with leading zeroes added to make the required number of digits. So, for example, `%08d` would result in an 8-digit number.

If a time-lapse value of 0 is entered, the application will take pictures as fast as possible. Note that there's an minimum enforced pause of 30ms between captures to ensure that exposure calculations can be made.

```
    --framestart,    -fs
```

Specifies the first frame number in the timelapse. Useful if you have already saved a number of frames, and want to start again at the next frame.

```
--datetime,    -dt
```

Instead of a simple frame number, the timelapse file names will use a date/time value of the format `aabbccddee`, where `aa` is the month, `bb` is the day of the month, `cc` is the hour, `dd` is the minute, and `ee` is the second.

```
--timestamp,   -ts
```

Instead of a simple frame number, the timelapse file names will use a single number which is the Unix timestamp, i.e. the seconds since 1970.

```
--thumb,    -th          Set thumbnail parameters (x:y:quality)
```

Allows specification of the thumbnail image inserted into the JPEG file. If not specified, defaults are a size of 64x48 at quality 35.

if `--thumb none` is specified, no thumbnail information will be placed in the file. This reduces the file size slightly.

```
--demo,    -d           Run a demo mode <milliseconds>
```

This options cycles through the range of camera options. No capture is taken, and the demo will end at the end of the timeout period, irrespective of whether all the options have been cycled. The time between cycles should be specified as a millisecond value.

```
--encoding,    -e          Encoding to use for output file
```

Valid options are `jpg`, `bmp`, `gif`, and `png`. Note that unaccelerated image types (GIF, PNG, BMP) will take much longer to save than jpg, which is hardware accelerated. Also note that the filename suffix is completely ignored when deciding the encoding of a file.

```
--restart,    -rs
```

Sets the JPEG restart marker interval to a specific value. Can be useful for lossy transport streams because it allows a broken JPEG file to still be partially displayed.

```
    --exif,    -x          EXIF tag to apply to captures (format as 'key=value')
```

Allows the insertion of specific EXIF tags into the JPEG image. You can have up to 32 EXIF tag entries. This is useful for tasks like adding GPS metadata. For example, to set the longitude:

```
--exif GPS.GPSLongitude=5/1,10/1,15/1
```

would set the longitude to 5 degs, 10 minutes, 15 seconds. See EXIF documentation for more details on the range of tags available; the supported tags are as follows:

```
IFD0.<   or
IFD1.<
ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, I

EXIF.<
ExposureTime, FNumber, ExposureProgram, SpectralSensitivity, ISOSpeedRatings, OEC

GPS.<
GPSVersionID, GPSLatitudeRef, GPSLatitude, GPSLongitudeRef, GPSLongitude, GPSAlti

EINT.<
InteroperabilityIndex, InteroperabilityVersion, RelatedImageFileFormat, RelatedIma
```

Note that a small subset of these tags will be set automatically by the camera system, but will be overridden by any EXIF options on the command line.

Setting `--exif none` will prevent any EXIF information being stored in the file. This reduces the file size slightly.

```
    --gpsdexif,    -gps
```

Applies real-time EXIF information from any attached GPS dongle (using GSPD) to the image; requires `libgps.so` to be installed.

```
    --fullpreview,    -fp          Full preview mode
```

This runs the preview window using the full resolution capture mode. Maximum frames per second in this mode is 15fps, and the preview will have the same field of view as the capture. Captures should happen more quickly, as no mode change should be required. This feature is currently under development.

```
    --keypress,     -k          Keypress mode
```

The camera is run for the requested time (`-t`), and a capture can be initiated throughout that time by pressing the Enter key. Pressing X then Enter will exit the application before the timeout is reached. If the timeout is set to 0, the camera will run indefinitely until the user presses X then Enter. Using the verbose option (`-v`) will display a prompt asking for user input, otherwise no prompt is displayed.

```
    --signal,     -s          Signal mode
```

The camera is run for the requested time (`-t`), and a capture can be initiated throughout that time by sending a `USR1` signal to the camera process. This can be done using the `kill` command. You can find the camera process ID using the `pgrep raspistill` command.

```
kill -USR1 <process id of raspistill>
```

```
    --burst,     -bm
```

Sets burst capture mode. This prevents the camera from returning to preview mode in between captures, meaning that captures can be taken closer together.

raspiyuv

Many of the options for `raspiyuv` are the same as those for `raspistill` . This section shows the differences.

Unsupported options:

```
    --exif, --encoding, --thumb, --raw, --quality
```

Extra options :

```
    --rgb,    -rgb        Save uncompressed data as RGB888
```

This option forces the image to be saved as RGB data with 8 bits per channel, rather than YUV420.

Note that the image buffers saved in `raspiyuv` are padded to a horizontal size divisible by 32, so there may be unused bytes at the end of each line. Buffers are also padded vertically to be divisible by 16, and in the YUV mode, each plane of Y,U,V is padded in this way.

```
    --luma,    -y
```

Only outputs the luma (Y) channel of the YUV image. This is effectively the black and white, or intensity, part of the image.

```
    --bgr,    -bgr
```

Saves the image data as BGR data rather than YUV.

raspivid

```
    --width,    -w        Set image width <size>
```

Width of resulting video. This should be between 64 and 1920.

```
    --height,    -h        Set image height <size>
```

Height of resulting video. This should be between 64 and 1080.

```
    --bitrate,    -b        Set bitrate
```

Use bits per second, so 10Mbits/s would be `-b 10000000`. For H264, 1080p30 a high quality bitrate would be 15Mbits/s or more. Maximum bitrate is 25Mbits/s (`-b 25000000`), but much over 17Mbits/s won't show noticeable improvement at 1080p30.

```
    --output,    -o        Output filename <filename>
```

Specify the output filename. If not specified, no file is saved. If the filename is '-', then all output is sent to stdout.

To connect to a remote IPv4 host, use `tcp` or `udp` followed by the required IP Address. e.g. `tcp://192.168.1.2:1234` or `udp://192.168.1.2:1234` .

To listen on a TCP port (IPv4) and wait for an incoming connection use `--listen (-l)` option, e.g. `raspivid -l -o tcp://0.0.0.0:3333` will bind to all network interfaces, `raspivid -l -o tcp://192.168.1.1:3333` will bind to a local IPv4.

```
    --listen,    -l
```

When using a network connection as the data sink, this option will make the sytem wait for a connection from the remote system before sending data.

```
    --verbose,    -v          Output verbose information during run
```

Outputs debugging/information messages during the program run.

```
    --timeout,    -t          Time before the camera takes picture and shuts down
```

The total length of time that the program will run for. If not specified, the default is 5000ms (5 seconds). If set to 0, the application will run indefinitely until stopped with Ctrl-C.

```
    --demo,    -d          Run a demo mode <milliseconds>
```

This options cycles through the range of camera options. No recording is done, and the demo will end at the end of the timeout period, irrespective of whether all the options have been cycled. The time between cycles should be specified as a millisecond value.

```
    --framerate,    -fps          Specify the frames per second to record
```

At present, the minimum frame rate allowed is 2fps, and the maximum is 30fps. This is likely to change in the future.

```
    --penc,    -e          Display preview image after encoding
```

Switch on an option to display the preview after compression. This will show any compression artefacts in the preview window. In normal operation, the preview will show the camera output prior to being compressed. This option is not guaranteed to work in future releases.

```
    --intra,    -g          Specify the intra refresh period (key frame rate/GoP)
```

Sets the intra refresh period (GoP) rate for the recorded video. H264 video uses a complete frame (I-frame) every intra refresh period, from which subsequent frames are based. This option specifies the number of frames between each I-frame. Larger numbers here will reduce the size of the resulting video, and smaller numbers make the stream less error-prone.

```
    --qp,    -qp          Set quantisation parameter
```

Sets the initial quantisation parameter for the stream. Varies from approximately 10 to 40, and will greatly affect the quality of the recording. Higher values reduce quality and decrease file size. Combine this setting with a bitrate of 0 to set a completely variable bitrate.

```
    --profile,    -pf          Specify H264 profile to use for encoding
```

Sets the H264 profile to be used for the encoding. Options are:

- baseline
- main
- high

```
    --level,    -lev
```

Specifies the H264 encoder level to use for encoding. Options are `4` , `4.1` , and `4.2` .

```
    --irefresh,    -if
```

Sets the H264 intra-refresh type. Possible options are `cyclic`, `adaptive`, `both`, and `cyclicrows`.

```
    --inline,     -ih          Insert PPS, SPS headers
```

Forces the stream to include PPS and SPS headers on every I-frame. Needed for certain streaming cases e.g. Apple HLS. These headers are small, so don't greatly increase the file size.

```
    --spstimings,    -stm
```

Insert timing information into the SPS block.

```
    --timed,     -td          Do timed switches between capture and pause
```

This options allows the video capture to be paused and restarted at particular time intervals. Two values are required: the on time and the off time. On time is the amount of time the video is captured, and off time is the amount it is paused. The total time of the recording is defined by the `timeout` option. Note that the recording may take slightly over the timeout setting depending on the on and off times.

For example:

```
 raspivid -o test.h264 -t 25000 -timed 2500,5000
```

will record for a period of 25 seconds. The recording will be over a timeframe consisting of 2500ms (2.5s) segments with 5000ms (5s) gaps, repeating over the 20s. So the entire recording will actually be only 10s long, since 4 segments of 2.5s = 10s separated by 5s gaps. So:

2.5 record – 5 pause - 2.5 record – 5 pause - 2.5 record – 5 pause – 2.5 record

gives a total recording period of 25s, but only 10s of actual recorded footage.

```
    --keypress,    -k         Toggle between record and pause on Enter keypress
```

On each press of the Enter key, the recording will be paused or restarted. Pressing X then Enter will stop recording and close the application. Note that the timeout value will be used to signal the end of recording, but is only checked after each Enter keypress; so if the system is waiting for a keypress, even if the timeout has expired, it will still wait for the keypress before exiting.

```
    --signal,    -s          Toggle between record and pause according to SIGUSR1
```

Sending a `USR1` signal to the `raspivid` process will toggle between recording and paused. This can be done using the `kill` command, as below. You can find the `raspivid` process ID using `pgrep raspivid`.

```
kill -USR1 <process id of raspivid>
```

Note that the timeout value will be used to indicate the end of recording, but is only checked after each receipt of the `SIGUSR1` signal; so if the system is waiting for a signal, even if the timeout has expired, it will still wait for the signal before exiting.

```
    --split,    -sp
```

When in a signal or keypress mode, each time recording is restarted, a new file is created.

```
    --circular,    -c
```

Select circular buffer mode. All encoded data is stored in a circular buffer until a trigger is activated, then the buffer is saved.

```
    --vectors,    -x
```

Turns on output of motion vectors from the H264 encoder to the specified file name.

```
    --flush,    -fl
```

Forces a flush of output data buffers as soon as video data is written. This bypasses any OS caching of written data, and can decrease latency.

```
    --save-pts,    -pts
```

Saves timestamp information to the specified file. Useful as an imput file to `mkvmerge`.

```
    --codec,    -cd
```

Specifies the encoder codec to use. Options are `H264` and `MJPEG`. H264 can encode up to 1080p, whereas MJPEG can encode upto the sensor size, but at decreased framerates due to the higher processing and storage requirements.

```
    --initial,    -i          Define initial state on startup
```

Define whether the camera will start paused or will immediately start recording. Options are `record` or `pause`. Note that if you are using a simple timeout, and `initial` is set to `pause`, no output will be recorded.

```
    --segment,    -sg         Segment the stream into multiple files
```

Rather than creating a single file, the file is split into segments of approximately the number of milliseconds specified. In order to provide different filenames, you should add `%04d` or similar at the point in the filename where you want a segment count number to appear e.g:

```
--segment 3000 -o video%04d.h264
```

will produce video clips of approximately 3000ms (3s) long, named `video0001.h264`, `video0002.h264` etc. The clips should be seamless (no frame drops between clips), but the accuracy of each clip length will depend on the intraframe period, as the segments will always start on an I-frame. They will therefore always be equal or longer to the specified period.

The most recent version of Raspivid will also allow the file name to be time-based, rather than using a segment number. For example:

```
--segment 3000 -o video_%c.h264
```

will produce file names formatted like so:

```
video_Fri Jul 20 16:23:48 2018.h264
```

There are many different formatting options available — see here for a full list. Note than the `%d` and `%u` options are not available, as they are used for the segment number formatting, and that some combinations may produce invalid file names.

```
--wrap,    -wr          Set the maximum value for segment number
```

When outputting segments, this is the maximum the segment number can reach before it's reset to 1, giving the ability to keep recording segments, but overwriting the oldest one. So if set to 4, in the segment example above, the files produced will be `video0001.h264`, `video0002.h264`, `video0003.h264`, and `video0004.h264`. Once `video0004.h264` is recorded, the count will reset to 1, and `video0001.h264` will be overwritten.

```
--start,    -sn          Set the initial segment number
```

When outputting segments, this is the initial segment number, giving the ability to resume a previous recording from a given segment. The default value is 1.

```
--raw,     -r
```

Specify the output file name for any raw data files requested.

```
--raw-format,     -rf
```

Specify the raw format to be used if raw output requested. Options as `yuv`, `rgb`, and `grey`. `grey` simply saves the Y channel of the YUV image.

Examples

Still captures

By default, captures are done at the highest resolution supported by the sensor. This can be changed using the `-w` and `-h` command line options.

Take a default capture after 2s (times are specified in milliseconds) on the viewfinder, saving in `image.jpg`:

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

Take a capture at a different resolution:

```
raspistill -t 2000 -o image.jpg -w 640 -h 480
```

Reduce the quality considerably to reduce file size:

```
raspistill -t 2000 -o image.jpg -q 5
```

Force the preview to appear at coordinate 100,100, with width 300 pixels and height 200 pixels:

```
raspistill -t 2000 -o image.jpg -p 100,100,300,200
```

Disable preview entirely:

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

Save the image as a PNG file (lossless compression, but slower than JPEG). Note that the filename suffix is ignored when choosing the image encoding:

```
raspistill -t 2000 -o image.png –e png
```

Add some EXIF information to the JPEG. This sets the Artist tag name to Boris, and the GPS altitude to 123.5m. Note that if setting GPS tags you should set as a minimum GPSLatitude, GPSLatitudeRef, GPSLongitude, GPSLongitudeRef, GPSAltitude, and GPSAltitudeRef:

```
raspistill -t 2000 -o image.jpg -x IFD0.Artist=Boris -x GPS.GPSA
```

Set an emboss image effect:

```
raspistill -t 2000 -o image.jpg -ifx emboss
```

Set the U and V channels of the YUV image to specific values (128:128 produces a greyscale image):

```
raspistill -t 2000 -o image.jpg -cfx 128:128
```

Run preview for 2s, with no saved image:

```
raspistill -t 2000
```

Take a time-lapse picture, every 10 seconds for 10 minutes (10 minutes = 600000ms), naming the files `image_num_001_today.jpg`, `image_num_002_today.jpg` and so on, with the latest picture also available under the name `latest.jpg`:

```
raspistill -t 600000 -tl 10000 -o image_num_%03d_today.jpg -l la
```

Take a picture and send the image data to stdout:

```
raspistill -t 2000 -o -
```

Take a picture and send the image data to a file:

```
raspistill -t 2000 -o - > my_file.jpg
```

Run the camera forever, taking a picture when Enter is pressed:

```
raspistill -t 0 -k -o my_pics%02d.jpg
```

Video captures

Image size and preview settings are the same as for stills capture. Default size for video recording is 1080p (1920x1080).

Record a 5s clip with default settings (1080p30):

```
raspivid -t 5000 -o video.h264
```

Record a 5s clip at a specified bitrate (3.5Mbits/s):

```
raspivid -t 5000 -o video.h264 -b 3500000
```

Record a 5s clip at a specified framerate (5fps):

```
raspivid -t 5000 -o video.h264 -f 5
```

Encode a 5s camera stream and send the image data to stdout:

```
raspivid -t 5000 -o -
```

Encode a 5s camera stream and send the image data to a file:

```
raspivid -t 5000 -o - > my_file.h264
```

Shell error codes

The applications described here will return a standard error code to the shell on completion. Possible error codes are:

C Define	Code	Description
EX_OK	0	Application ran successfully
EX_USAGE	64	Bad command line parameter
EX_SOFTWARE	70	Software or camera error
	130	Application terminated by Ctrl-C

ABOUT US

About us

Our team

Governance

Safeguarding

Our supporters

Jobs

Contact us

SUPPORT

Help

Documentation

Projects

Training

Downloads

Research

FAQ

SIGN UP TO OUR NEWSLETTER

Your email here

SUBSCRIBE

RASPBERRY PI FOUNDATION
UK REGISTERED CHARITY 1129409

Privacy   Cookies   Trademark rules and brand guidelines