

Rpi Camera Module

From eLinux.org

[Back to the Hub.](#)

Hardware & Peripherals:

Hardware and Hardware History.

Low-level Peripherals and Expansion Boards.

Screens, Cases and Other Peripherals.

Contents

- 1 Introduction
- 2 Technical Parameters
- 3 Sample Programs
 - 3.1 RaspiVid
 - 3.2 RaspiStill
 - 3.3 Motion
 - 3.4 Video4Linux2
 - 3.5 RAW images
 - 3.6 Configuration
- 4 Known Issues
- 5 Troubleshooting
 - 5.1 Camera not working at all
 - 5.2 Colored Banding or Flickering on video
- 6 3rd party software
 - 6.1 BerryCam - iOS
 - 6.2 Pi Sight - Android
 - 6.3 RaspiCam Remote - Android
 - 6.4 Pi Vision - Raspberry Pi
 - 6.5 picamera
- 7 Customizing
 - 7.1 Changing the ribbon cable
 - 7.2 Adjusting Lens Focus
 - 7.3 Macro, Wide, Tele adaptor lenses
 - 7.4 Replacing Stock Lens
 - 7.5 Removing IR Filter
 - 7.6 Housings
- 8 References

Introduction

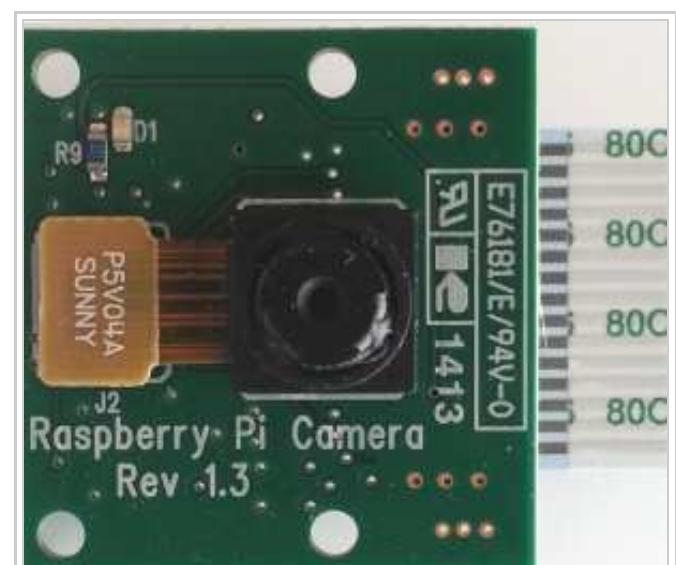
The Raspberry Pi camera board contains a 5 MPixel sensor, and connects via a ribbon cable to the CSI connector on the Raspberry Pi. A User's Guide (<http://www.raspberrypi.org/wp-content/uploads/2013/07/RaspiCam-Documentation.pdf>) describes setup and use. The video and still image quality is better than a USB webcam of similar price.

The Pi camera was released for sale on the 14th of May 2013^[1] and the initial production was 10k units.^[2]

The "Pi NoIR" version of the camera was released on the 28th of October 2013. It has the same sensor with the IR filter removed, and a black PCB.^[3] With no IR filter, it can see near-IR wavelengths (700 - 1000 nm) like a security camera, with the tradeoff of poor color rendition. It is otherwise the same and uses the same software as the normal Pi camera.

Technical Parameters

- Sensor type: OmniVision OV5647^[4] Color CMOS QSXGA (5-megapixel)
- Sensor size: 3.67 x 2.74 mm
- Pixel Count: 2592 x 1944
- Pixel Size: 1.4 x 1.4 um
- Lens: f=3.6 mm, f/2.9
- Angle of View: 54 x 41 degrees
- Field of View: 2.0 x 1.33 m at 2 m
- Full-frame SLR lens equivalent: 35 mm
- Fixed Focus: 1 m to infinity
- Video: 1080p at 30 fps with codec H.264 (AVC)
- Up to 90 fps Video at VGA
- Board size: 25 x 24 mm (not including flex cable)



Raspberry Pi Camera Module

Because the focal length of the lens is roughly the same as the width of the sensor, it is easy to remember the field of view: at x meters away, you can see about x meters horizontally, assuming 4x3 stills mode. Horizontal field of view in 1080p video mode is 75% of that (75% H x 55% V sensor crop for 1:1 pixels at 1920x1080).

References: Focal length and f-stop measurements [1] (<http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=46150>), [2] (<http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=32605&start=325#p357018>) depth of field & angle of view calculators[3] (<http://www.tawbaware.com/maxlyons/calc.htm>) sensor info[4] (<http://ovt.com/products/sensor.php?id=66>) f-stop measurement[5] (<http://www.raspberrypi.org/phpBB3/viewtopic.php?t=32605&p=357246#p357135>)

Sample Programs

Sample programs using the camera for taking stills and video have been developed for the Foundation. There is a User's Guide (<http://www.raspberrypi.org/wp-content/uploads/2013/07/RaspiCam-Documentation.pdf>) and the source for these programs is available on github (https://github.com/raspberrypi/userland/tree/master/host_applications/linux/apps/raspicam).

Forum user M1M1 published instructions on compiling the sample programs here (<http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43990#p350970>)

The command line parameters for the programs are as follows:

RaspiVid

```
RaspiVid Camera App v1.3.9
=====
Display camera output to display, and optionally saves an H264 capture at requested bitrate

usage: raspivid [options]

Image parameter commands

-, --help      : This help information
-w, --width    : Set image width <size>. Default 1920
-h, --height   : Set image height <size>. Default 1080
-b, --bitrate  : Set bitrate. Use bits per second (e.g. 10MBits/s would be -b 10000000)
-o, --output   : Output filename <filename> (to write to stdout, use '-o -')
-v, --verbose   : Output verbose information during run
-t, --timeout   : Time (in ms) to capture for. If not specified, set to 5s. Zero to disable
-d, --demo     : Run a demo mode (cycle through range of camera options, no capture)
-fps, --framerate : Specify the frames per second to record
-e, --penc     : Display preview image *after* encoding (shows compression artifacts)
-g, --intra    : Specify the intra refresh period (key frame rate/GoP size)
-pf, --profile  : Specify H264 profile to use for encoding
-td, --timed   : Cycle between capture and pause. -cycle on,off where on is record time and off is pause time
-s, --signal   : Cycle between capture and pause on Signal
-k, --keypress  : Cycle between capture and pause on ENTER
-i, --initial  : Initial state. Use 'record' or 'pause'. Default 'record'
-qp, --qp      : Quantisation parameter. Use approximately 10-40. Default 0 (off)
-ih, --inline   : Insert inline headers (SPS, PPS) to stream
-sg, --segment  : Segment output file in to multiple files at specified interval <ms>
-wr, --wrap    : In segment mode, wrap any numbered filename back to 1 when reach number
-sn, --start   : In segment mode, start with specified segment number
-sp, --split   : In wait mode, create new output file for each start event

H264 Profile options :
baseline,main,high

Preview parameter commands

-p, --preview   : Preview window settings <'x,y,w,h'>
-f, --fullscreen : Fullscreen preview mode
-op, --opacity   : Preview window opacity (0-255)
-n, --nopreview : Do not display a preview window

Image parameter commands

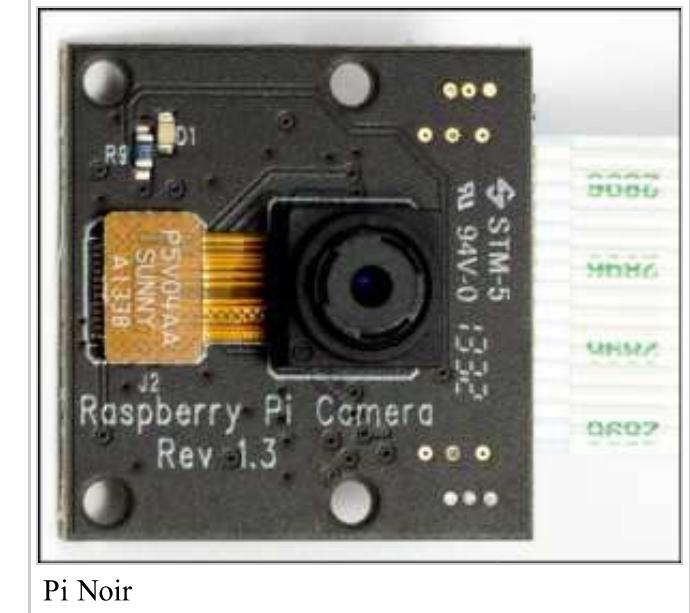
-sh, --sharpness : Set image sharpness (-100 to 100)
-co, --contrast  : Set image contrast (-100 to 100)
-br, --brightness : Set image brightness (0 to 100)
-sa, --saturation : Set image saturation (-100 to 100)
-ISO, --ISO      : Set capture ISO
-vs, --vstab    : Turn on video stabilisation
-ev, --ev       : Set EV compensation (-25 to 25)
-ex, --exposure : Set exposure mode (see Notes)
-awb, --awb    : Set AWB mode (see Notes)
-ifx, --imfix   : Set image effect (see Notes)
-cfx, --colfx   : Set colour effect (U:V)
-mm, --metering : Set metering mode (see Notes)
-rot, --rotation : Set image rotation (90,180,270)
-hf, --hflip    : Set horizontal flip
-vf, --vflip    : Set vertical flip
-roi, --roi     : Set region of interest (x,y,w,d as normalised coordinates [0.0-1.0])
-ss, --shutter   : Set shutter speed in microseconds

Notes

Exposure mode options :
auto,night,nightpreview,backlight,spotlight,sports,snow,beach,verylong,fixedfps,antishake,fireworks

AWB mode options :
off,auto,sun,cloud,shade,tungsten,fluorescent,incandescent,flash,horizontal

Image Effect mode options :
```



Pi Noir

```
none,negative,solarise,sketch,denoise,emboss,oilpaint,hatch,gpen,pastel,watercolour,film,blur,saturation,colou
Metering Mode options :
average,spot,backlit,matrix
```

RaspiStill

```
RaspiStill Camera App v1.3.6
=====
Runs camera for specific time, and take JPG capture at end if requested

usage: raspistill [options]

Image parameter commands

-, --help      : This help information
-w, --width    : Set image width <size>
-h, --height   : Set image height <size>
-q, --quality  : Set jpeg quality <0 to 100>
-r, --raw       : Add raw bayer data to jpeg metadata
-o, --output   : Output filename <filename> (to write to stdout, use '-o -'). If not specified, no file is saved
-l, --latest    : Link latest complete image to filename <filename>
-v, --verbose   : Output verbose information during run
-t, --timeout   : Time (in ms) before takes picture and shuts down (if not specified, set to 5s)
               minimum 30ms, setting to 0 waits forever.
-th, --thumb    : Set thumbnail parameters (x:y:quality) or none
-d, --demo      : Run a demo mode (cycle through range of camera options, no capture)
-e, --encoding  : Encoding to use for output file (jpg, bmp, gif, png)
-x, --exif      : EXIF tag to apply to captures (format as 'key=value') or none
-tl, --timelapse : Timelapse mode. Takes a picture every <t>ms
-fp, --fullpreview : Run the preview using the still capture resolution (may reduce preview fps)
-k, --keypress   : Wait between captures for a ENTER, X then ENTER to exit
-s, --signal     : Wait between captures for a SIGUSR1 from another process
-g, --gl         : Draw preview to texture instead of using video render component
-gc, --glcapture : Capture the GL frame-buffer instead of the camera image

Preview parameter commands

-p, --preview   : Preview window settings <'x,y,w,h'>
-f, --fullscreen : Fullscreen preview mode
-op, --opacity   : Preview window opacity (0-255)
-n, --nopreview : Do not display a preview window

Image parameter commands

-sh, --sharpness : Set image sharpness (-100 to 100)
-co, --contrast  : Set image contrast (-100 to 100)
-br, --brightness : Set image brightness (0 to 100)
-sa, --saturation : Set image saturation (-100 to 100)
-ISO, --ISO      : Set capture ISO
-vs, --vstab     : Turn on video stabilisation
-ev, --ev        : Set EV compensation (-25 to 25)
-ex, --exposure  : Set exposure mode (see Notes)
-awb, --awb      : Set AWB mode (see Notes)
-ifx, --imxffx   : Set image effect (see Notes)
-cfx, --colfx    : Set colour effect (U:V)
-mm, --metering  : Set metering mode (see Notes)
-rot, --rotation  : Set image rotation (90,180,270)
-hf, --hflip     : Set horizontal flip
-vf, --vflip     : Set vertical flip
-roi, --roi      : Set region of interest (x,y,w,d as normalised coordinates [0.0-1.0])
-ss, --shutter   : Set shutter speed in microseconds

Notes

Exposure mode options :
auto,night,nightpreview,backlight,spotlight,sports,snow,beach,verylong,fixedfps,antishake,fireworks

AWB mode options :
off,auto,sun,cloud,shade,tungsten,fluorescent,incandescent,flash,horizon

Image Effect mode options :
none,negative,solarise,sketch,denoise,emboss,oilpaint,hatch,gpen,pastel,watercolour,
film,blur,saturation,colourswap,washedout,posterise,colourpoint,colourbalance,cartoon

Metering Mode options :
average,spot,backlit,matrix

Preview parameter commands

-gs, --glscene   : GL scene square,teapot,mirror,yuv,sobel
-gw, --glwin     : GL window settings <'x,y,w,h'>
```

The raspivid utility saves .h264 format files. Some programs need this in a .mp4 wrapper to use, which you can do in several ways. One is to use "MP4Box" which is in the 'gpac' package.

```
sudo apt-get update
sudo apt-get install gpac

MP4Box -add filename.h264 filename.mp4
```

Motion

There is a version of the security cam / motion-detection program 'motion' adapted for the R-Pi camera (motion-MMAL), written by user 'dozencrows'.^[5] Motion detection with still and video capture at a resolution of 1024x576 at 1 fps is possible, and several fps at lower resolutions. There is a blog post explaining how to set it up.^[6] It is also possible to use the stock motion program (sudo apt-get install motion) with either of the two available V4L2 drivers.

Video4Linux2

There is an unofficial Userspace Video4Linux2 (V4L2) driver.^{[7][8][9]} This allows use of the R-Pi camera with many standard Linux applications developed for USB webcams. Due to the Pi's ARMv6 cpu, lower resolutions and framerates are typical when compared with a modern desktop system.

There is an official kernel-space V4L2 driver in development, with a "beta" version available for testing from 2 Dec 2013.^[10]

RAW images

The **raspistill --raw** option will append RAW data to the JPEG. The RAW data contains the image direct from the sensor, before debayer and color processing. A simple tool called **raspi_dng**^[11] is being developed^[12] to convert the JPEG+RAW file into a standard Adobe DNG file.

Configuration

The cameras red LED lights up when recording video or taking a still picture, this can be turned off by adding the following line to config.txt (RPiconfig)

```
-----  
disable_camera_led=1  
-----
```

It's possible to turn it back on at runtime by using GPIO5^[13]

Known Issues

- ISO can go to 1600 in 'night' or 'sports' mode, but can only be directly set via -ISO from 100 to 800.
- exposure times longer than 1 second are not supported

Prior to March 2014, video mode did not use the full sensor area (only a 1920x1080 center crop), and there was no video mode faster than 30 fps. With new firmware as of March 18 2014^[14] the camera has a full-sensor 2x2 binned video mode, and also can do 640 x 480 up to 90 fps.

Troubleshooting

Camera not working at all

Example error message: "mmal: mmal_vc_component_create: failed to create component 'vc.ril.camera' (1:ENOMEM)"

Make sure the flex cable is inserted the correct way on both ends, and each connector has the locking tab engaged. The RPi + camera draws about 260 mA more current when recording video, than without the camera. The Model B is about 550 mA by itself, so camera use pushes it over 800 mA. Some power supplies, cables, and polyfuses may not maintain 5V onboard at this current load well enough for reliable operation. You can check this with a voltmeter across TP1 and TP2.^{[15][16]}

In some cases, the micro-connector from the camera module to the camera board PCB has been loose. This can be fixed by removing and re-seating the small (tiny!) yellow flex cable connector labelled "P5V04A SUNNY" on the camera board. It makes a click when it is fully seated.^[17] Note: before touching the camera board, make sure you are grounded, as the board is ESD sensitive (can be damaged by static shock).

This error message also appear if you forgot to configure the configuration file of the Raspberry. Just type: sudo raspi-config and choose "5- Enable camera" option then "Enable"

As of august 2014, with 2014-06-20-wheezy-raspbian.img, the firmware of a RPi B+ model needs to be updated :

```
-----  
$ sudo rpi-update  
-----
```

Here is a working version :

```
-----  
$ vcgencmd version  
Aug 12 2014 18:21:28  
Copyright (c) 2012 Broadcom  
version f32b2bbfdea55d48c9a52b92e5c798f9aa5f47bc (tainted) (release)  
  
$ uname -a  
Linux poopdeck 3.12.26+ #703 PREEMPT Tue Aug 12 18:13:52 BST 2014 armv6l GNU/Linux  
  
$ cat /proc/cpuinfo | grep Revision  
Revision : 0010  
-----
```

Colored Banding or Flickering on video

This can be caused by a poor power supply, or bad power cable. Several people have fixed this by replacing the power supply, or the microUSB power cable.^{[18][19]}

It can also be caused by indoor fluorescent lighting. The camera defaults to video mode at 30 fps, which works well in the USA with 60 Hz power. If you are in Europe or elsewhere with 50 Hz power, this may give you flicker under fluorescent lighting.^[20] Try using the option **-fps 25** to change the frame rate to 25 frames per second.

3rd party software

BerryCam - iOS

BerryCam enables you to use your iOS device to control the Raspberry Pi camera module.

A Python handler script is installed on the raspberry pi.

Control exposure modes, special effects for still images with the simple interface - no need to learn complex SSH commands. Preview your images on the iOS device and save to your camera roll. The same images are also saved to your Raspberry Pi.^[6] (<http://www.fotosyn.com/berrycam-support/>)

Pi Sight - Android

Pi Sight is an Android application which allows you to control all the aspects of your Raspberry Pi camera.

It was inspired by BerryCam for iOS and has much of the same functionality. You can control exposure modes, image effects, white balance, image size and quality and pretty much every other parameter that can be accessed when using the camera on the command line. It also uses the same Python handler script on the Raspberry Pi as BerryCam ^[7] (<http://www.pi-supply.com/pi-sight>)

RaspiCam Remote - Android

RaspiCam Remote is an app to view the Raspberry Pi camera module on your Android Device.

Unlike other apps, no software needs to be installed or configured on the Raspberry Pi - simply type in the ip address of the Pi. It also allows use of USB cameras.

Continuous 30fps h264 streaming for camera monitoring on your android device, and live view of the various image effects. ^[8] (https://play.google.com/store/apps/details?id=com.pibits.raspberrypiremote&hl=en_GB)

Pi Vision - Raspberry Pi

Pi Vision is a GUI based controller for the Raspberry Pi camera module which runs on your Raspberry Pi. ^[9] (<http://heywhatsthebigidea.net/projects/pi-vision-a-raspberry-pi-camera-controller/>)

picamera

The picamera library is a pure Python library for controlling the Raspberry Pi camera module. It is available for installation from the Raspbian repository (as python-picamera) or from PyPI (<http://pypi.python.org/pypi/picamera/>) (for non-Raspbian distros). ^[10] (<http://picamera.readthedocs.org/>)

Customizing

Changing the ribbon cable

The ribbon cable that connects the camera board to the Pi is 15-core 1mm pitch ribbon cable, with alternate end contacts (top side / bottom side). ModMyPi stock Replacement Raspberry Pi Camera Cables (<https://www.modmypi.com/raspberry-pi-camera/replacement-raspberry-pi-camera-cable-150mm>) for £1.99 and have worldwide shipping. In the UK they can also be sourced from Toby Electronics (<http://www.toby.co.uk/content/catalogue/products.aspx?series=FFCx-xx-A-xx-xx>) where the required type is FFC1-15-B150105160MM^[21]. Other suppliers have been noted on eBay.

The Foundation has noted that the camera only passed EMI emissions testing with ribbon cable lengths of 15cm, although longer cables may also work. ^[22] One user 'rew' reports a 4 m (13 foot) extension cable is working.^{[23][24]}

A quick guide on how to replace the raspberry pi camera cable (<https://www.modmypi.com/blog/how-to-replace-the-raspberry-pi-camera-cable>) is available on the ModMyPi tutorials.

Adjusting Lens Focus

The camera is fixed-focus (at infinity) as delivered. The thread-locking glue dots can be scraped off^[25] allowing manual lens focus, as close as 2.5 cm. To change the focus to **FL** cm, unscrew the lens (turn counter-clockwise) by **D** degrees, where $D = 360 * (3.5 \text{ cm} / \text{FL})$. For example, to focus at 10 cm, unscrew lens $360 * (3.5 / 10) = 126$ degrees. Values are approximate^[26]. The closest achievable focus is about 25 mm, after unscrewing the lens roughly 2 full turns.^[27] At this point it is loose and must be held in place by other means.

Unscrewing more than 2 turns results in removal of the lens^[28] allowing use of other optical systems. The lens thread size is believed to be M6 x 0.35^[29].

Macro, Wide, Tele adaptor lenses

The near focus point of the camera is about 1 meter as delivered. Without modification to the camera itself, it is possible to put a simple external lens in front such as reading glasses, SLR diopter ("close-up filter") or loupe to enable closer focus. For best results, put the additional lens very close to the camera lens.

Macro, Wide-Angle, and Telephoto adapter lenses designed for the iPhone 4/4s and similar devices will also work on the R-Pi camera. Depending on your configuration, properly mounting the accessory lens in front of the camera module may require some ingenuity. Focus becomes critical with a telephoto adapter, so success there may also require the manual focus modification.

Replacing Stock Lens

Changing the lens is a more significant and risky modification than simply adding an accessory lens. Richard Kinch has made an adaptor to mount M12 thread lenses on the R-Pi camera board, after unscrewing and removing the stock lens.^{[30][31]} Martin Miller has posted a guide on raspberrytorte.com describing complete camera module disassembly and replacing lens holder with a stock M12 unit.^[32] Various M12 x 0.5 (S-mount) lenses intended for board-level cameras are available at focal lengths from fisheye to long telephoto.^{[33][34][35]}

Removing IR Filter

The IR filter is rigid and glued to the bottom inside of the black plastic lens housing, requiring complete disassembly to remove. Two early attempts by RPi forum users M33P^[36] and scorp^[37] were initially successful, but in both cases the camera died soon afterwards from blunt trauma to the fragile bond wires on the die.^[38]

A RPi blog post (27 May 2013) has more description of this process with a video of the modification, and video from the resulting IR-enabled camera.^[39]

Housings

To help building your own camera board case or enclosure, a hand-measured mechanical drawing^[40] of the camera module is available, thanks to Gert van Loo. There is another drawing^[41] from raspberrypi-spy. There are some CAD files for housings and camera holders available on Thingiverse^[42].

References

1. ↑ <http://www.raspberrypi.org/archives/3890>
2. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=32605&start=350#p358580>
3. ↑ <http://www.raspberrypi.org/archives/5128>
4. ↑ <http://www.ovt.com/products/sensor.php?id=66>
5. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44966>
6. ↑ <http://rbnrpi.wordpress.com/project-list/setting-up-wireless-motion-detect-cam/>
7. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=50639>
8. ↑ <http://www.linux-projects.org/modules/sections/index.php?op=viewarticle&artid=14>
9. ↑ <http://www.linux-projects.org/modules/sections/index.php?op=viewarticle&artid=16>
10. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=62364>
11. ↑ <https://github.com/illes/raspiraw>
12. ↑ <http://bealecorner.org/best/RPi/>
13. ↑ <http://www.raspberrypi-spy.co.uk/2013/05/how-to-disable-the-red-led-on-the-pi-camera-module/>
14. ↑ <http://www.raspberrypi.org/new-camera-mode-released/>
15. ↑ http://elinux.org/R-Pi_Troubleshooting#Troubleshooting_power_problems
16. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43998#p350528>
17. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=39996&start=150#p373336>
18. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44863>
19. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43982>
20. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=43982&p=351090#p350434>
21. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43737#p352401>
22. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43737#p348917>
23. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=43737#p362714>
24. ↑ http://www.bitwizard.nl/catalog/product_info.php?products_id=146
25. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=32605&start=225#p346170>
26. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=46637#p366719>
27. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=60828&p=454021#p454021>
28. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=32605&start=325#p352613>
29. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=46637&p=367053#p367053>
30. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=45887>
31. ↑ http://www.truetex.com/raspberry_pi_m12_lens_adapter.pdf
32. ↑ http://wiki.raspberrytorte.com/index.php?title=Camera_Module_Lens_Modification
33. ↑ http://www.solino.com/cms/index.php?article_id=193&clang=0
34. ↑ <http://www.m12lenses.com/M12-Mount-Mega-Pixel-Lenses-s/21.htm>
35. ↑ <http://www.net-gmbh.com/en/s-mount.html>
36. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44339#p352586>
37. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44339#p352902>
38. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44339>
39. ↑ <http://www.raspberrypi.org/archives/4088>
40. ↑ <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=44466>

41. ↑ <http://www.raspberrypi-spy.co.uk/2013/05/pi-camera-module-mechanical-dimensions/>
42. ↑ <http://www.thingiverse.com/search?q=raspberry+pi+camera&sa=>

Retrieved from "http://elinux.org/index.php?title=Rpi_Camera_Module&oldid=346190"

Category: RaspberryPi

-
- This page was last modified on 17 August 2014, at 12:06.
 - Content is available under a Creative Commons Attribution-ShareAlike 3.0 Unported License unless otherwise noted.