Diagnostic Camera on I2C Bus

When the camera is not detected, check for a loosen cable or any hardware failure. Or diagnostic the camera module by checking responses on the I2C bus.

#pi #camera #i2c

Last update: April 23, 2021

Table of Content

Camera I2C Address responses

Scan on I2C Bus 0, the camera module will response on two addresses `0x10` (camera sensor) and `0x64` (camera board).

Missing one of two above addresses means that there is an issue happened.

1. Hardware check

Run the command:

```
vcgencmd get_camera
```

which should print out `supported=1 detected=1` with a working camera.

If the output shows `detected=0`, check the ribbon cable first. If other camera still works after swapping the camera module, then do a software check.

2. Software check

An interesting topic: Camera not detected despite being plugged in on official Raspberry forum shows a method to check the connection of the camera board and the camera sensor on a I2C interface.

2.1. Install I2C tools

Install `i2c-tools`:

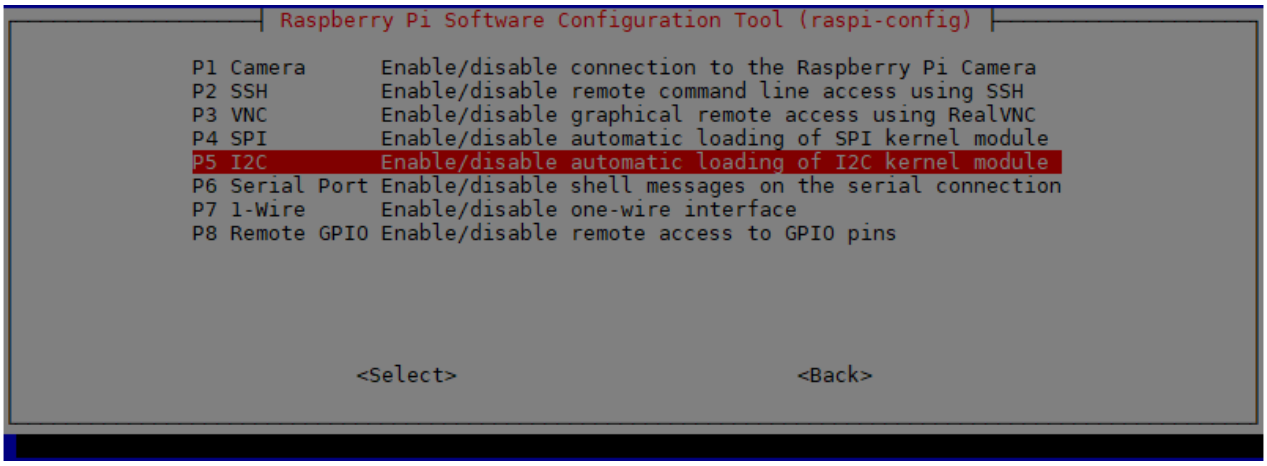
```
sudo apt-get install i2c-tools
```

This package contains a set of I2C tools for Linux: a bus probing tool, a chip dumper, register-level access helpers, EEPROM decoding scripts, and more.

```
i2cdetect       detect I2C chips
i2cdump         examine I2C registers
i2cget          read from I2C/SMBus chip registers
i2cset          set I2C registers
i2ctransfer     send user-defined I2C messages in one transfer
```

2.2. Load I2C driver

Permanently enable I2C interface by running `sudo raspi-config` or adding `i2c-dev` to */etc/modules*:



Enable I2C Interface via `raspi-config`

Or just load driver temporarily for a quick check:

```
sudo modprobe i2c-dev
```

2.3. Config GPIOs

GPIO pin number

Please look at PI GPIO document for more information about setting GPIOs.

Note that GPIO number is defined in BCM2835 ARM processor, not the number printed on Pi boards. Read more at BCM2835 Peripherals.

Use `raspi-gpio get` to get the current status of GPIOs.

1. Change **GPIO0** and **GPIO1** to input /* by default they are set to SDA0 and SCL0 */

```
raspi-gpio set 0 ip
raspi-gpio set 1 ip
```

2. Change the function of **GPIO28**, **GPIO29** to I2C pins out **SDA0** and **SCL0** by setting *Alternate Function 0 (A0)* on those pins.

```
raspi-gpio set 28 a0
raspi-gpio set 29 a0
```

3. Power on Camera by setting High on output pin **GPIO44** and **GPIO45**

```
raspi-gpio set 44 dh
raspi-gpio set 40 dh
```

2.4. Scan I2C bus

Run `i2cdetect` on I2C BUS0 at */dev/i2c-0*:

```
i2cdetect 0
```

press `Y` to continue and it should print out:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: 10 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- 64 -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

Consider that `i2cdetect` tries to ping every address on a bus and reports whether an address responds. If any number shows up in report, it means there is a working device at that address.

> **Camera's I2C addresses**
>
> If having `0x64`, the camera board is connected properly, there is no problem with cable and connectors.
>
> If having `0x10`, it means the camera sensor has responded. there is no problem with sensor and sensor connection.