## Internet of Things Laboratory

## Assignment 10


## Prajakta Deokule

## C22019221332

## 4329


**Problem Statement: Understanding and connectivity of Raspberry Pi board with the camera. Write an application to capture and store the image.**

```
from time import sleep
import picamera
camera=picamera.PiCamera()
print(camera.revision)
try:
        camera.rotation=180
        camera.resolution=(640,480)
        camera.start_preview()
        for  i in range(2):
           camera.capture('/home/pi/'+str(i)+'.jpg')
           print('photo '+str(i))
           sleep(2)
        camera.stop_preview()
        print('Done')
finally:
   camera.close()
```

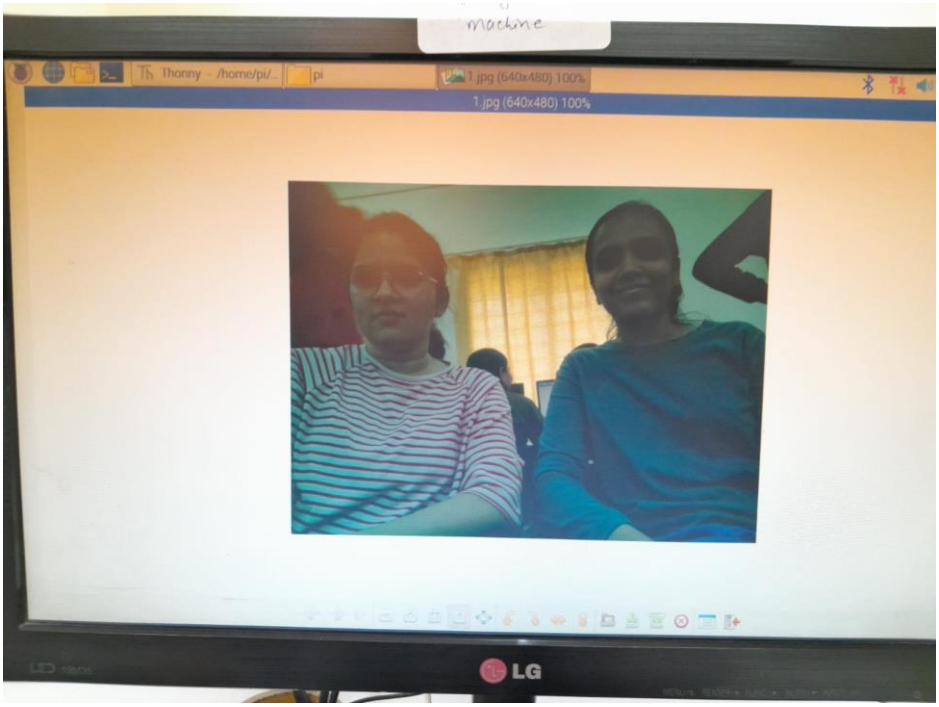




Image captured using Raspberry pi

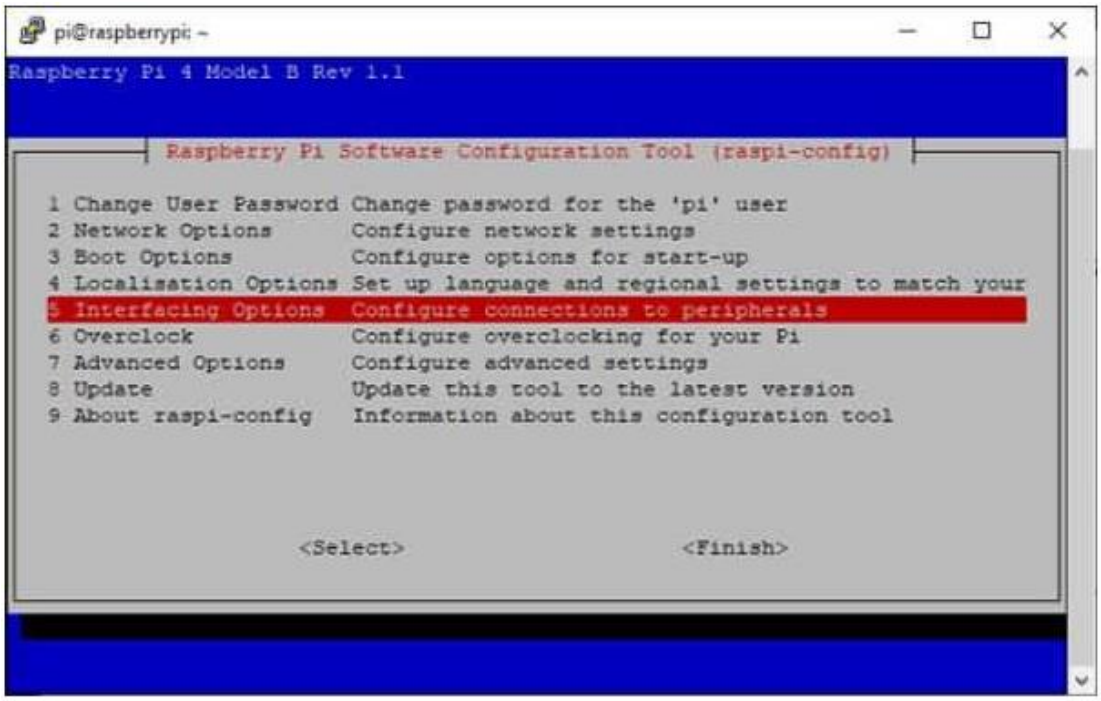
## 1. How to connect a camera with Rpi?

First, plug the ribbon connector of the camera module into the connector on the Raspberry Pi. The white connector closer to the USB and Ethernet ports is the one for the camera. The other port, located on the other side of the single-board computer, is meant for connecting a display.

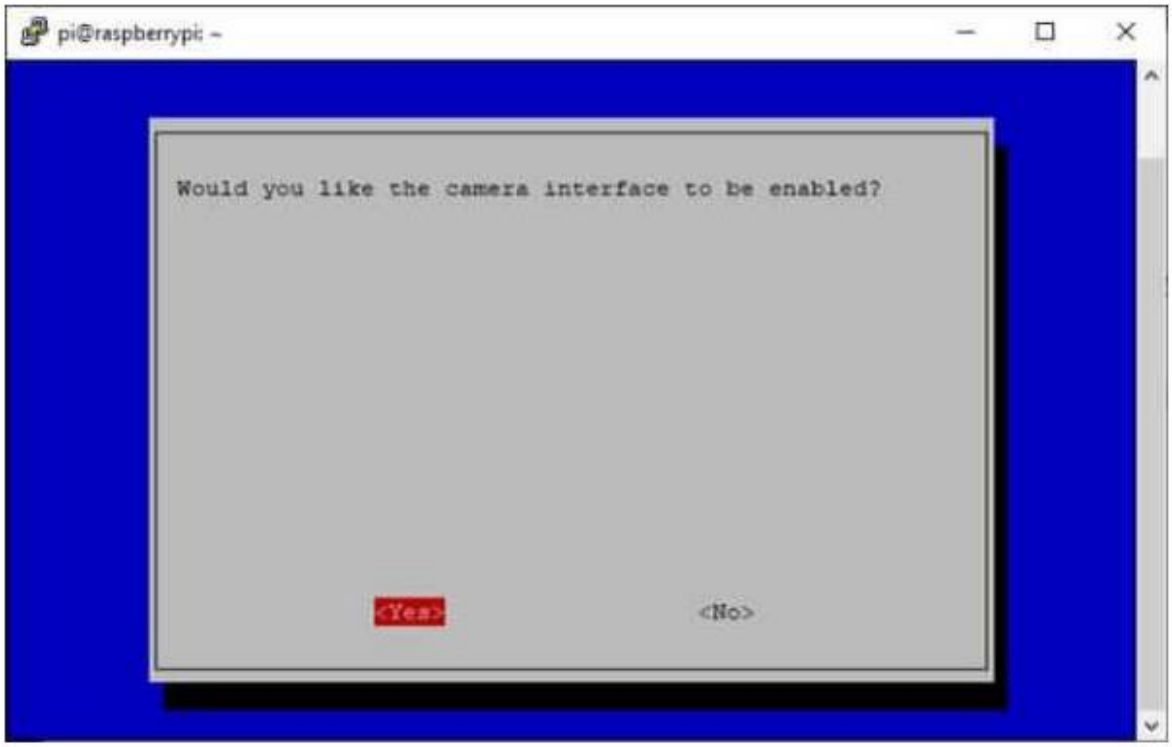
Connect the camera module while the Raspberry Pi is still powered off. Note that the shiny contacts on the ribbon cable should face away from the USB ports. Then, turn the Raspberry Pi on and launch the Raspberry Pi Software Configuration tool by typing the following command:

sudo raspi-config

In the graphical user interface, choose the fifth option in the list:




Then, select the ‘Camera’ option from the list and enable it:




When done, use the tab key in the main menu to select the ‘Finish’ option and then hit the enter key. After a reboot, the Raspberry Pi will be ready to interface with a connected camera module.

Once the computer is finished rebooting, you can check the status of the camera using the following command:

```
vcgencmd get_camera
```

**2. Write a program to record video with a Rpi camera**

```
import time

import picamera

camera=picamera.PiCamera()

camera.start_preview()

try:

        camera.rotation=180

        camera.resolution=(1280,720)

        print('Started Recording')

        name='/home/pi/Videos/video_raspberrypi.h264'
```

```
        camera.start_recording(name)
        camera.wait_recording(10)
        camera.stop_recording()
        print('Done')
        camera.stop_preview()
finally:
    camera.close()
```

### 3. How many cameras can Rpi handle?

One adapter board can connect **FOUR cameras** on a single Raspberry Pi board, and includes support for the High-Quality Camera.

### 4. How can we take two consecutive pictures with equivalent settings?

You may wish to capture a sequence of images all of which look the same in terms of brightness, color, and contrast (this can be useful in timelapse photography, for example). Various attributes need to be used in order to ensure consistency across multiple shots. Specifically, you need to ensure that the camera's exposure time, white balance, and gains are all fixed:

- To fix exposure time, set the `shutter_speed` attribute to a reasonable value.
- Optionally, set `iso` to a fixed value.
- To fix exposure gains, let `analog_gain` and `digital_gain` settle on reasonable values, then set `exposure_mode` to `'off'`.
- To fix white balance, set the `awb_mode` to `'off'`, then set `awb_gains` to a (red, blue) tuple of gains.

The following script provides a brief example of configuring these settings:

```
from time import sleep
from picamera import PiCamera

camera = PiCamera(resolution=(1280, 720), framerate=30)
# Set ISO to the desired value
camera.iso = 100
# Wait for the automatic gain control to settle
sleep(2)
```

```
# Now fix the values
camera.shutter_speed = camera.exposure_speed
camera.exposure_mode = 'off'
g = camera.awb_gains
camera.awb_mode = 'off'
camera.awb_gains = g
# Finally, take several photos with the fixed settings
camera.capture_sequence(['image%02d.jpg' % i for i in range(10)])
```

**5. Write a command to check the version of picamera installed with Rpi.**

The picamera library relies on the setuptools package for installation services. The setuptools pkg_resources API can be used to query which version of picamera is available in our Python environment like so:

```
>>> from pkg_resources import require
>>> require('picamera')
[picamera 1.2 (/usr/local/lib/python2.7/dist-packages)]
>>> require('picamera')[0].version
'1.2'
```

If multiple versions are installed (e.g. from pip and apt-get) they will not show up in the list returned by the require method. However, the first entry in the list will be the version that import picamera will import.

If the “No module named pkg_resources” error is received, you need to install the pip utility. This can be done with the following command in Raspbian:

```
$ sudo apt-get install python-pip
```