OLD DOMINION UNIVERSITY

Computer Science and Cybersecurity Pathways (CTE) 2020

Hands-on Activities with Raspberry Pi

Dr. Murat Kuzlu

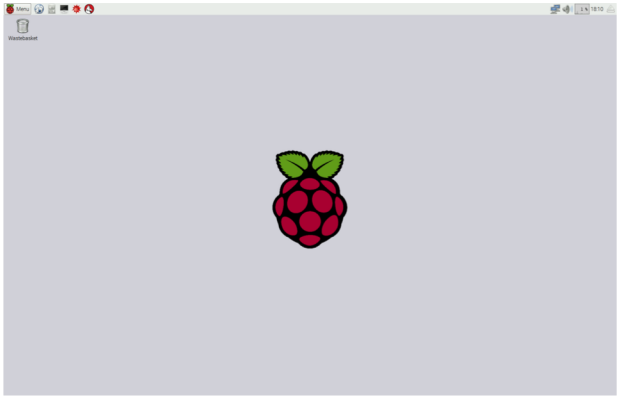
Department of Engineering Technology

What is a Raspberry Pi?

- Single-board computer with special Operating System called Raspbian
- Developed in the UK
- Several models
- Inexpensive ($5 for cheapest model, the Raspberry Pi Zero)
- Credit card size with many IO ports such as USB, ethernet and HDMI
- Can be used with a computer monitor, keyboard, and mouse
- Can be used for web browsing, making spreadsheets, word processing and playing games.

What is Raspbian?

- Free Operating System optimized for the Raspberry Pi
- Based on the Linux kernel
- Can be used like a desktop computer or through the terminal



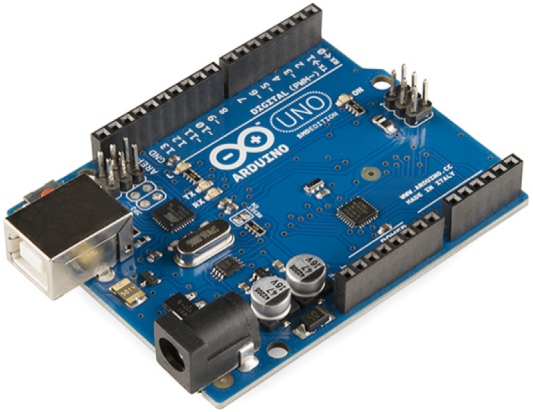
Raspberry Pi vs. Arduino

Raspberry Pi

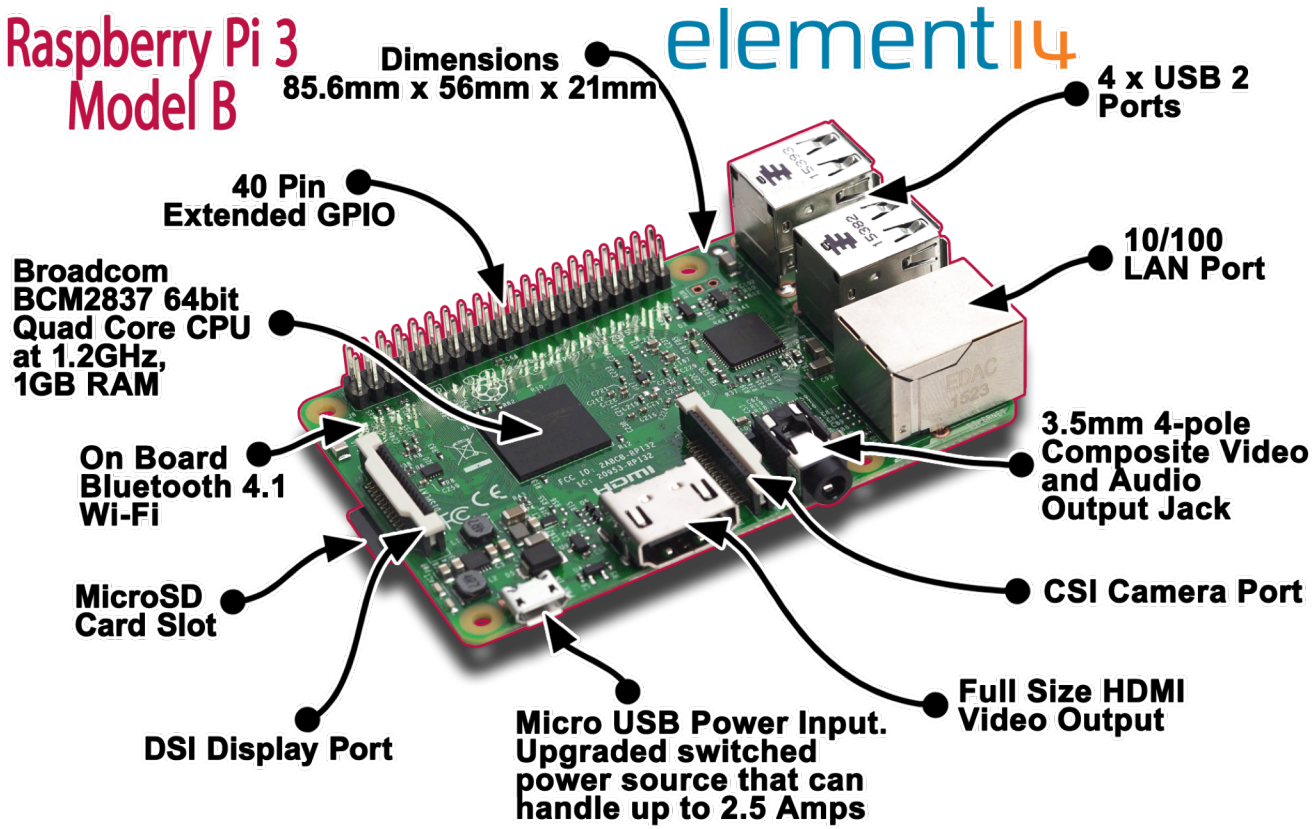
- Allows graphical user interface
- Can be directly connected to Internet
- More powerful and more memory
- Can be used with more programming languages
- Mini computer with Raspbian OS. It can run multiple program at a time.
- It is difficult to power using a battery pack.
- Python is recommended but can use other programming languages such as C,C++ and ruby

Arduino

- Low power consumption
- Can directly read analog inputs
- Requires less hardware (monitor, etc.) to get started
- No operating system needs to be installed
- Microcontroller which is a part of the computer. It can run only one program at a time.
- Arduino can be powered using a battery pack.
- Arduino uses Arduino, C/C++ programming languages



Raspberry Pi 3 Model B



Installation

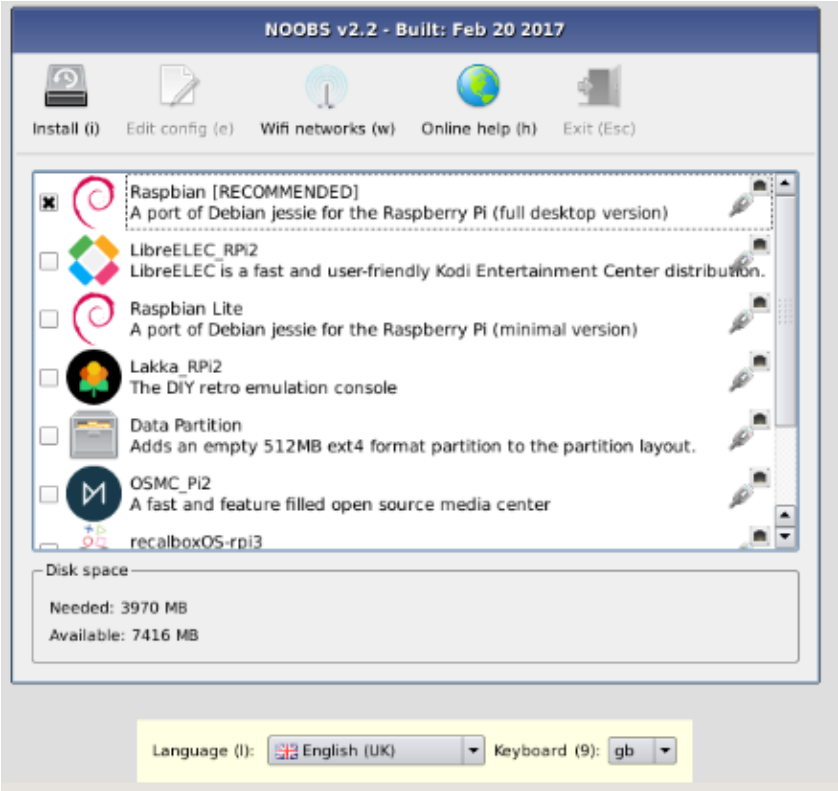
What we need ?

- SD card to install NOOBS or Raspian
- Display module and connectivity cable
- Keyboard and mouse
- Power supply

New Out Of Box Software (NOOBS) is an easy operating system installation manager for the Raspberry Pi.
SD cards with NOOBS preinstalled are available also can be downloaded on the Raspberry Pi website.

To set up a blank SD card with NOOBS:

- Format an SD card as FAT.
- Your SD card will need to be at least 16GB for Full Raspbian, or at least 8GB for all other installs.
- Download and extract the files from the NOOBS zip file.
- Copy the extracted files onto the SD card that you just formatted, so that this file is at the root directory of the SD card.



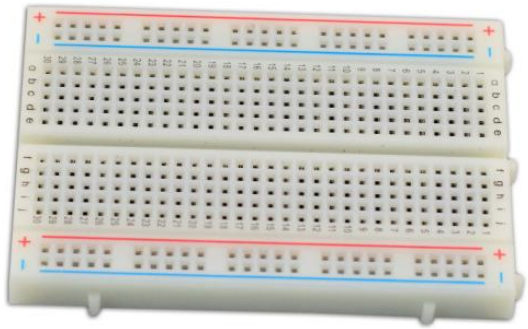
Creating a Basic Python Program

- Open Terminal 
- Type
  nano helloworld.py
  and press ENTER to open a new file in the nano text editor
- Type
  print(“Hello, World!”)
- Use CTRL + O and ENTER to save
- Exit with CTRL + X
- Type
  python helloworld.py
- and press ENTER to run the program
  Hello, World! should appear

COMPONENTS

Jumper Cables

Breadboard

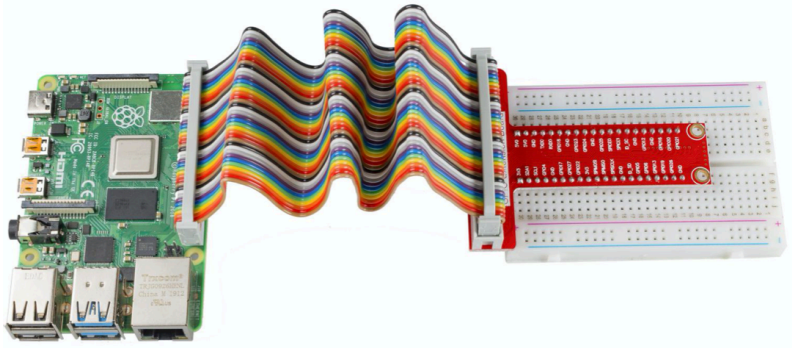


T-Cobbler

40 pin Ribbon cable

Anti reverse cable

GPIO Extension



Hands – on Activity – I

Laser Emitter Module

Laser Emitter Module

- In this tutorial you will learn how to use a Laser (or laser emitter) with Raspberry Pi. A laser is a device that emits light through a process of optical amplification based on the stimulated emission of electromagnetic radiation. Lasers differ from other sources of light because they emit light coherently.
- Laser is widely used in medical treatment, military, and other fields due to its good directivity and energy concentration.
- We can even see the module send Morse signals.
- We need to download the code library with typing below command in the terminal

git clone https://github.com/salihsarp/CTE_RaspberryPi.git

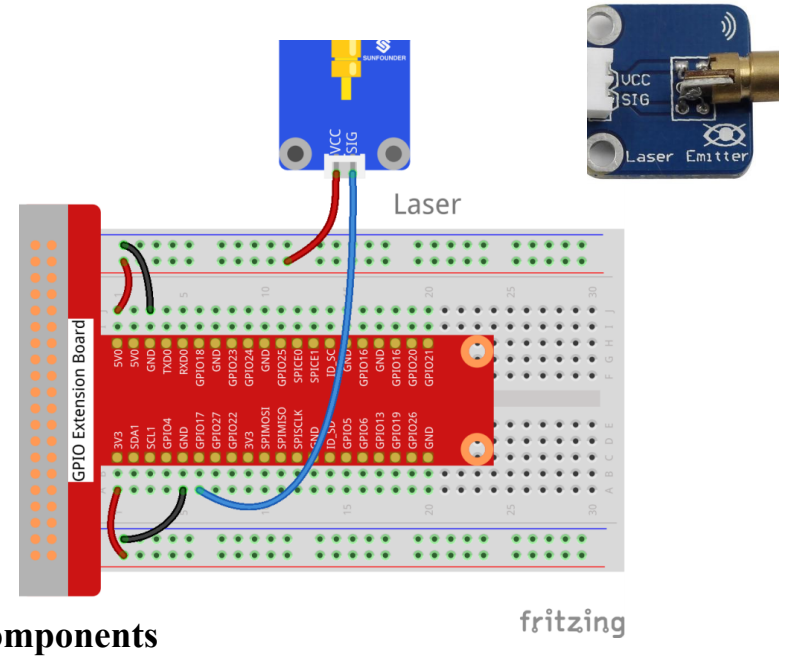
- To run the code after building the circuit.
  - Change directory to

cd /home/pi/CTE_RaspberryPi/Projects/1_Laser

  - Run the code

sudo python Laser_Module.py

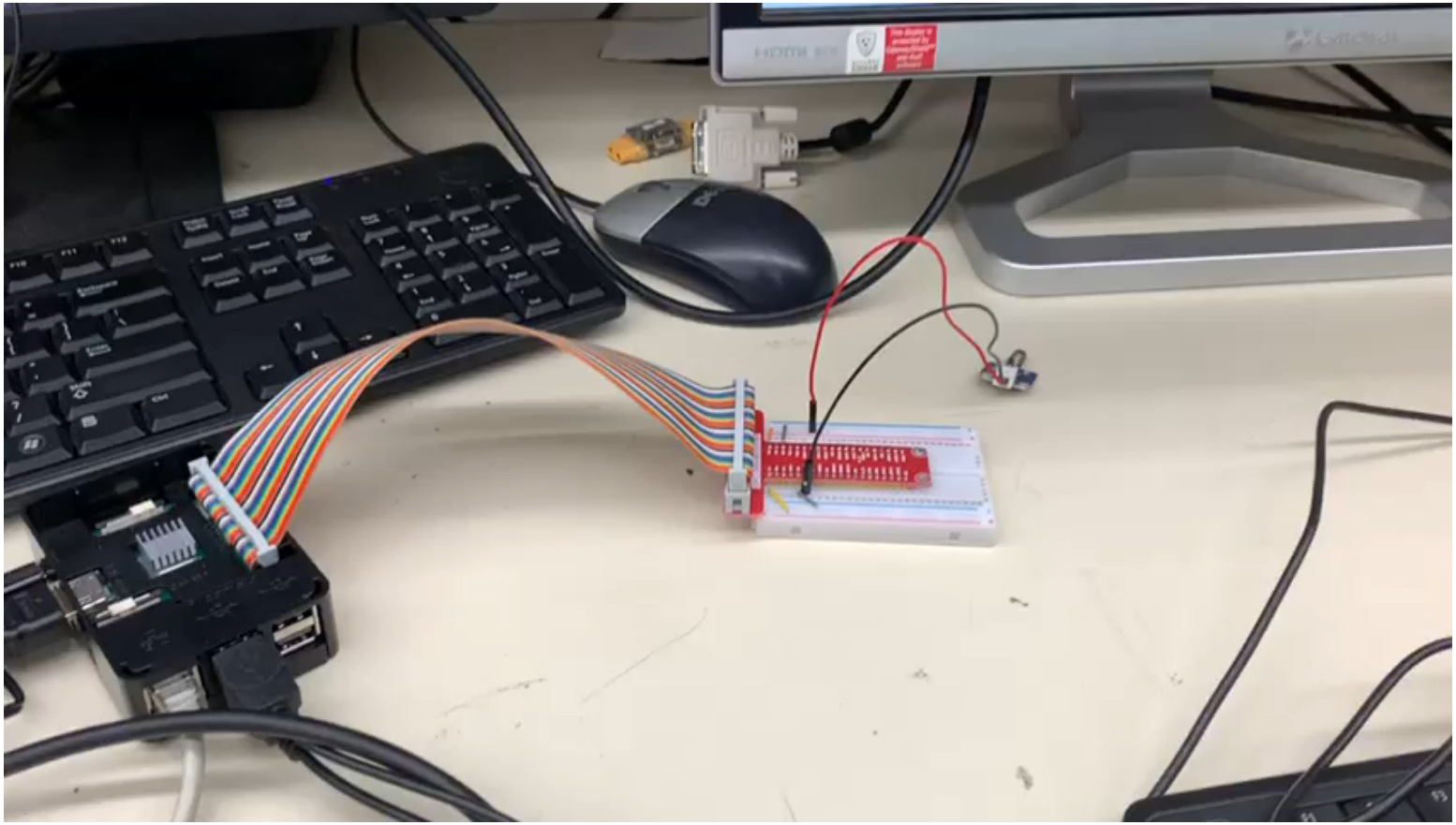
```
def loop():
    while True:
        print '...Laser on'
        GPIO.output(LedPin, GPIO.LOW)  # led on
        time.sleep(0.5)
        print 'Laser off...'
        GPIO.output(LedPin, GPIO.HIGH) # led off
        time.sleep(0.5)
```



Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 4 * Jumper wires (Male to Male, 2 red and 2 black)
- 1 * Network cable (or USB wireless network adapter)
- 1 * Laser Emitter module
- 1 * 2-Pin anti-reverse cable

Note: DO NOT look directly at the laser head. It can cause great harm to your eyes. You can point the laser beam to the table and see the light spot flashing on the table.



- Your laser should start blinking as seen in the video!

Hands – on Activity – II

RGB LED Module

RGB LED Module

- RGB LED modules can emit various colors of light. Three LEDs of red, green, and blue are packaged into a transparent or semitransparent plastic shell with four pins led out.
- The three primary colors of red, green, and blue can be mixed and compose all kinds of colors by brightness, so you can make an RGB LED emit colorful light by controlling the circuit.
- We need to download the code library with typing below command in the terminal

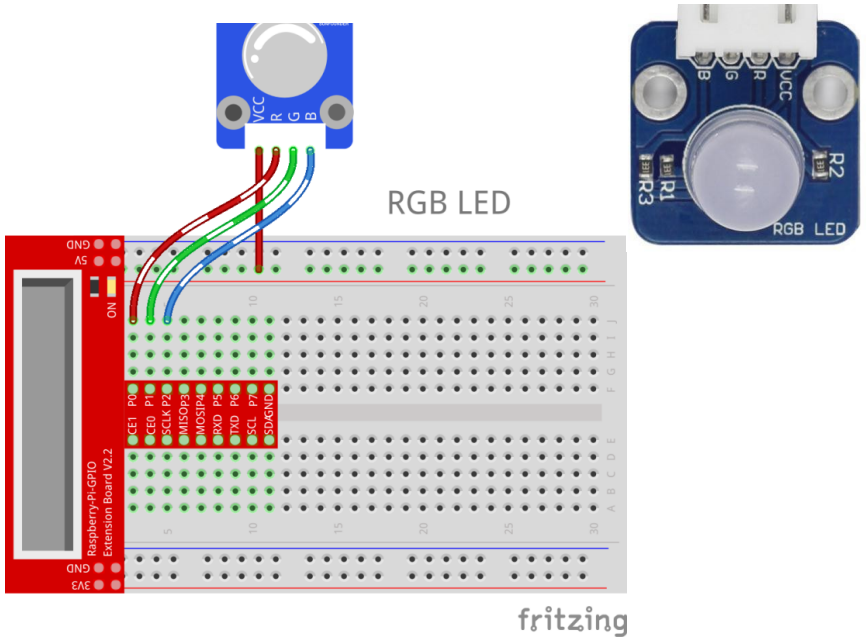
git clone https://github.com/salihsarp/CTE_RaspberryPi.git

- To run the code after building the circuit.
- Change directory to

cd /home/pi/ CTE_RaspberryPi/Projects /2_RGB_LED

- Run the code

sudo python RGB.py

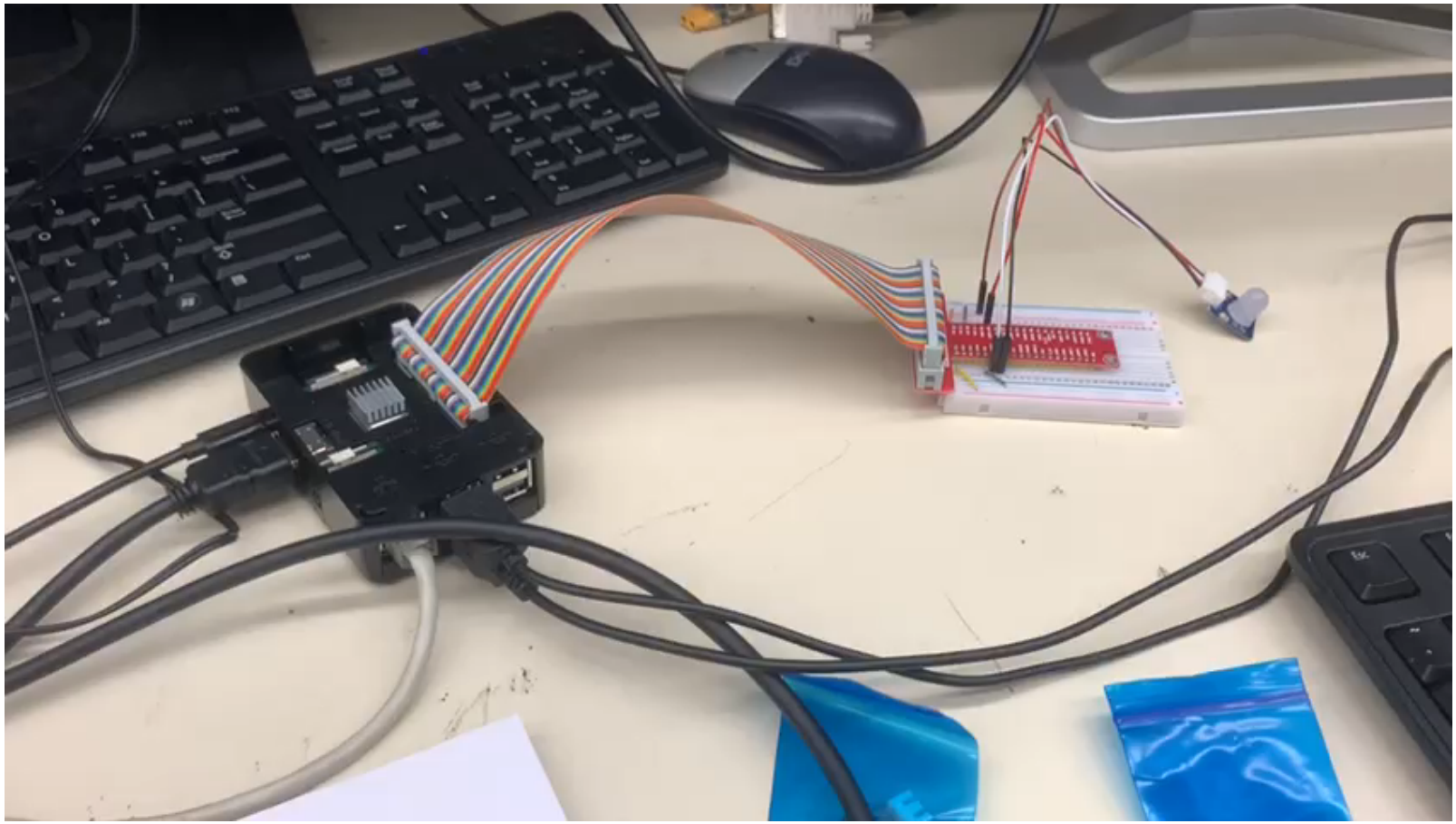


Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 1 * Network cable (or USB wireless network adapter)
- 1 * RGB LED module
- 1 * 4-Pin anti-reverse cable

You will see the RGB LED light up and display different colors in turn.

```
def setColor(col):   # For example : col = 0x112233
    R_val = (col & 0xff0000) >> 16
    G_val = (col & 0x00ff00) >> 8
    B_val = (col & 0x0000ff) >> 0

    R_val = map(R_val, 0, 255, 0, 100)
    G_val = map(G_val, 0, 255, 0, 100)
    B_val = map(B_val, 0, 255, 0, 100)

    p_R.ChangeDutyCycle(100-R_val)     # Change duty cycle
    p_G.ChangeDutyCycle(100-G_val)
    p_B.ChangeDutyCycle(100-B_val)

def loop():
    while True:
        for col in colors:
            setColor(col)
            time.sleep(1)
```



- Your RGB LED should start changing colors!

Hands – on Activity – III

Touch Switch Module

Touch Switch Module

- In this experiment, touch the base electrode of the transistor by fingers to make it conduct as human body itself is a kind of conductor and an antenna that can receive electromagnetic waves in the air.
- These electromagnetic wave signals collected from the human body are amplified by the transistor and processed by the comparator on the module to output steady signals.
- We need to download the code library with typing below command in the terminal

git clone https://github.com/salihsarp/CTE_RaspberryPi.git

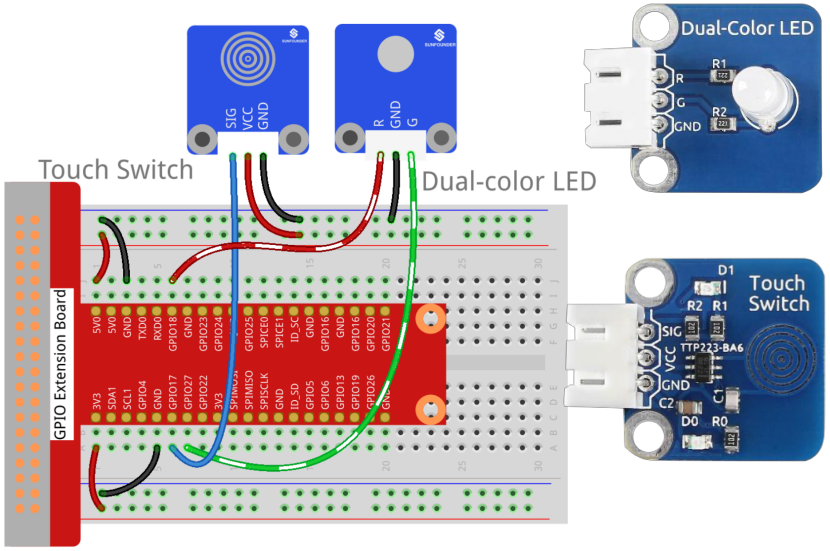
- To run the code after building the circuit.
  - Change directory to

cd /home/pi/CTE_RaspberryPi/Projects/3_Touch_Switch

  - Run the code

sudo python Touch_Switch.py

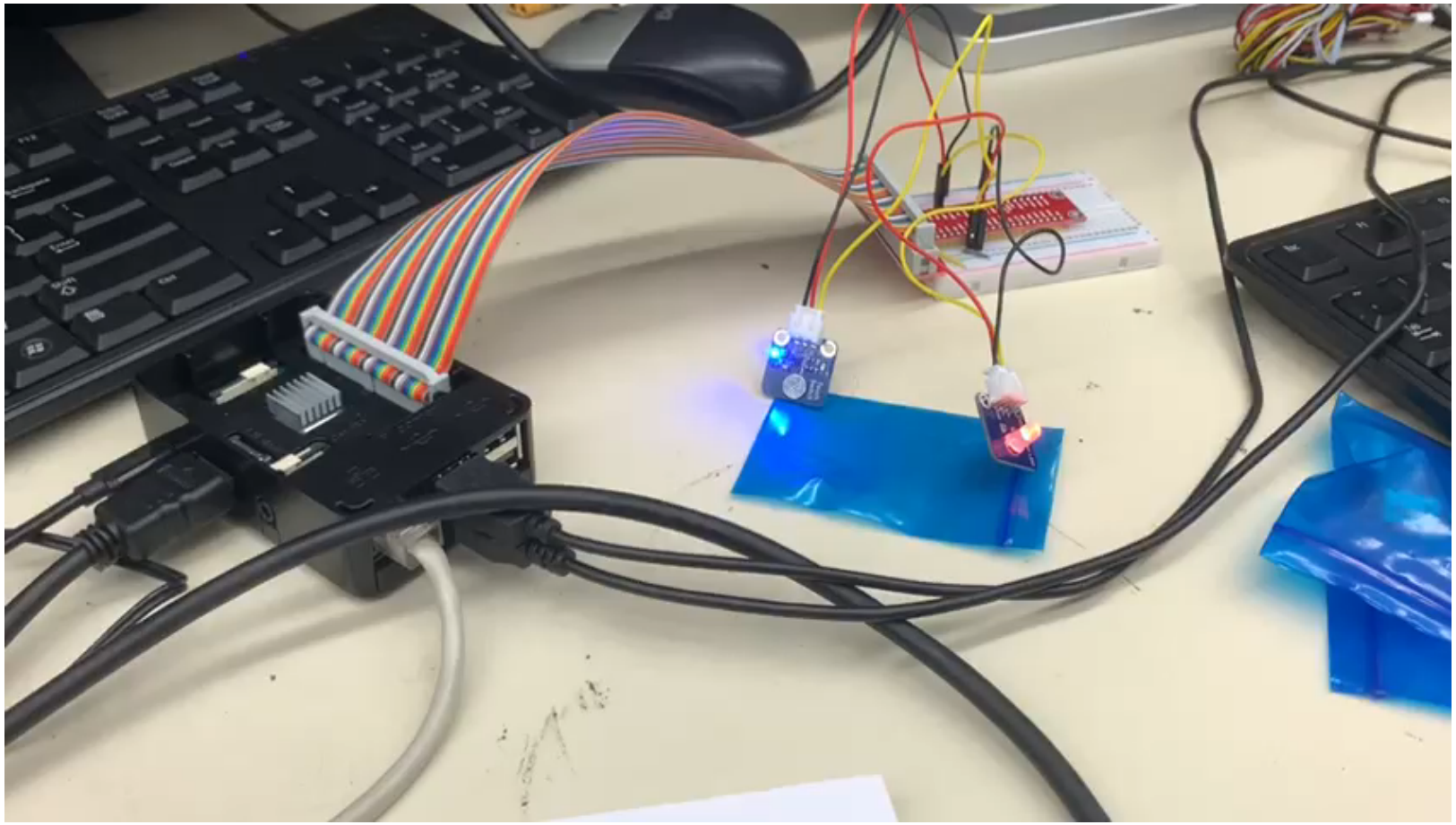
```
def loop():
    while True:
        Led(GPIO.input(TouchPin))
        Print(GPIO.input(TouchPin))

def destroy():
    GPIO.output(Gpin, GPIO.HIGH)       # Green led off
    GPIO.output(Rpin, GPIO.HIGH)       # Red led off
    GPIO.cleanup()                     # Release resource
```



Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 1 * Network cable (or USB wireless network adapter)
- 1 * Touch Switch module
- 1 * Dual-Color LED module
- 2 * 3-Pin anti-reverse cable

Touch the metal disk, you can see the LED change its colors and "ON" and "OFF" printed on the screen.



- Your LED should switch colors each time you touch the touch sensor!

Hands – on Activity – IV

Active Buzzer Module

Active Buzzer

- In this tutorial you will learn how to use a buzzer (or piezo speaker) with Raspberry Pi. Buzzers can be found in alarm devices, computers, timers and confirmation of user input such as a mouse click or keystroke.
- Two types of Buzzer , Active and Passive.
- We can produce many different rhythms even Morse Code.
- We need to download the code library with typing below command in the terminal

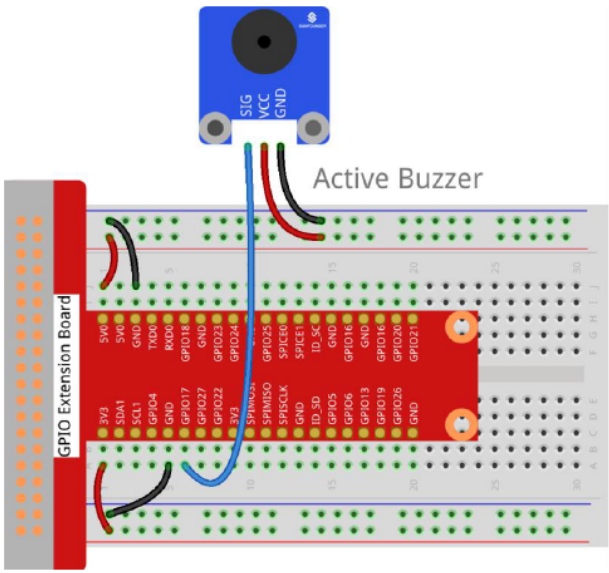
git clone https://github.com/salihsarp/CTE_RaspberryPi.git

- To run the code after building the circuit.
  - Change directory to

cd /home/pi/CTE_RaspberryPi/Projects/4_Active_Buzzer

  - Run the code

sudo python Active_Buzzer.py




```
def on():
    GPIO.output(BuzzerPin, GPIO.LOW)

def off():
    GPIO.output(BuzzerPin, GPIO.HIGH)

def beep(x):
    on()
    time.sleep(x)
    off()
    time.sleep(x)

def loop():
    while True:
        beep(0.5)
```

Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 4 * Jumper wires (Male to Male, 2 red and 2 black)
- 1 * Network cable (or USB wireless network adapter)
- 1 * Passive buzzer module
- 1 * Active buzzer module
- 1 * 3-Pin anti-reverse cable

The active buzzer has built-in oscillating source, so it will beep as long as it is wired up, but it can only beep with fixed frequency. Passive buzzer can be programmed when to beep.

Hands – on Activity – V

Sound Sensor Module

Sound Sensor

- In this application you will learn how to use a sound sensor and conversion of analog signal to digital signal with Raspberry Pi.
- Sound sensor is a component that receives sound waves and converts them into electrical signal. It detects the sound intensity in ambient environment like a microphone.
- Then we will convert analog quantity into digital quantity by PCF8591 and transfer them to MCU.
- We need to download the code library with typing below command in the terminal

git clone https://github.com/salihsarp/CTE_RaspberryPi.git

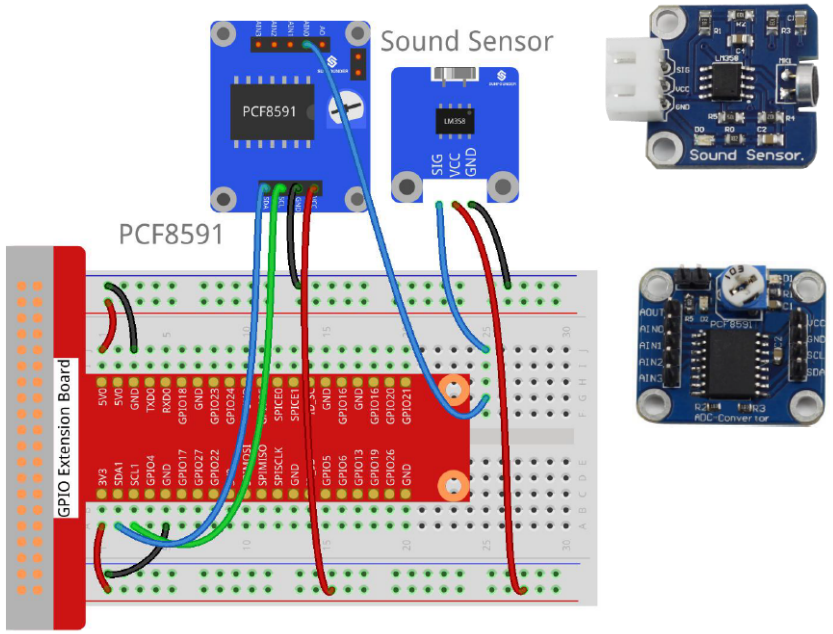
- To run the code after building the circuit.
  - Change directory to

cd /home/pi/CTE_RaspberryPi/Projects/ 5_Sound_Sensor

  - Run the code

sudo python Sound_Sensor.py

```
def loop():
    count = 0
    while True:
        voiceValue = ADC.read(0)
        if voiceValue:
            print 'Value:', voiceValue
            if voiceValue < 50:
                print "Voice detected! ", count
                count += 1
            time.sleep(0.2)
```



Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 4 * Jumper wires (Male to Male, 2 red and 2 black)
- 1 * Network cable (or USB wireless network adapter)
- 1 * PCF8591
- 1 * Sound sensor module
- 1 * 3-Pin anti-reverse cable
- Several Jumper wires (Male to Female)

Speak close to or blow to the microphone, and you can see “Voice In!! ***” printed on the screen.

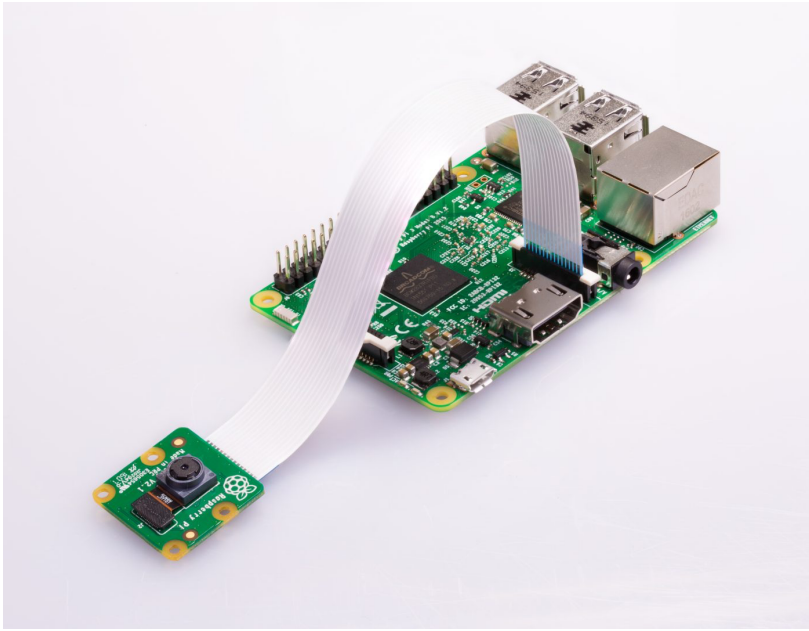
Hands – on Activity – VI

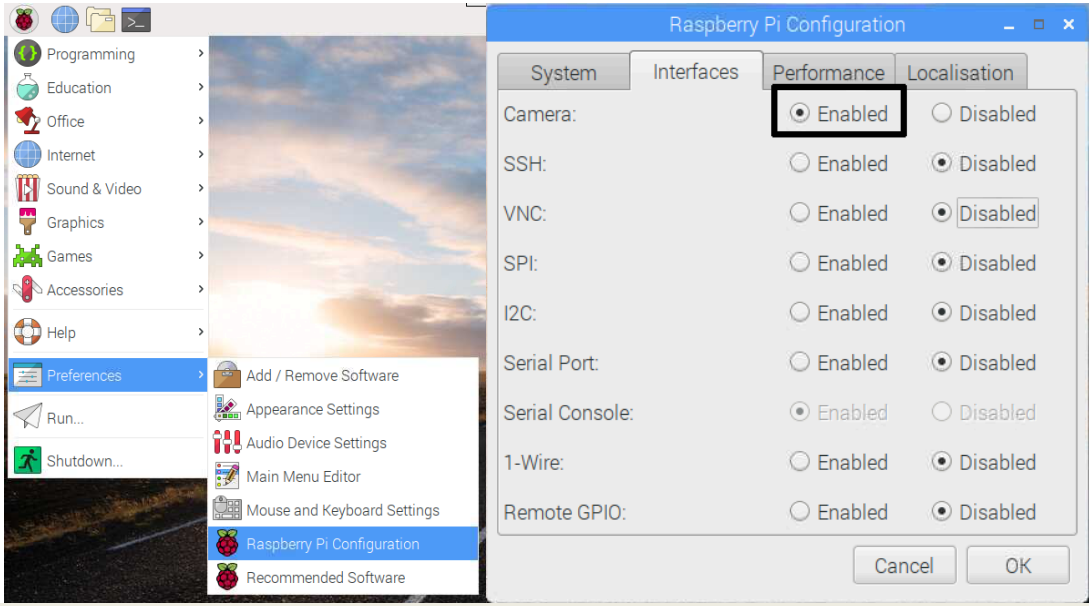
Camera IoT Application

Raspberry Pi Camera Setup

- **Step 1:-** First, turn off your Pi and then locate the camera connector, a long black connector between the HDMI port and Audio Jack. The connector is marked with 'CAMERA' on the board.
- **Step 2:-** To connect camera pull up on the ends of the black plastic connector and insert camera cable as shown in the image. Camera cable's metal pads should face the metal pads inside the connector. Now press down on the ends of the black connector.
- **Step 3:-** We need to enable the camera. Go to the 'Preferences' and 'Raspberry Pi Configuration' tab.
- **Step 4:-** Then, a screen will open, and on that screen click on 'Interfaces'. Click 'Enabled' on the 'Camera' row.
- **Step 5:-** After this, reboot your Raspberry Pi.
- **Step 6:-** After the reboot, test the camera to check whether its working or not by using the following command:

  raspistill -o cam_test.jpg
- **Step 7:-** If the camera works properly, the on-board red LED of the camera will glow whenever you click as photo.
- **Step 8:-** If you see the command fail error, go back and check that you have enabled the camera or not and make sure the camera cable is connected firmly with the board.

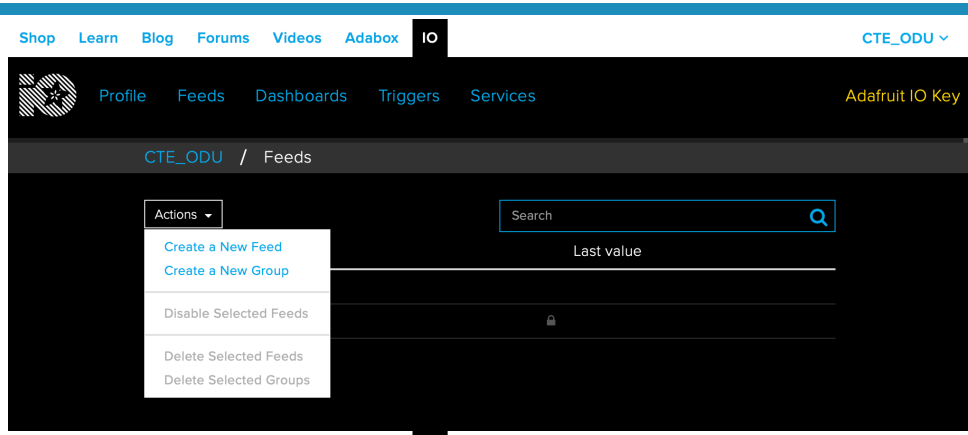


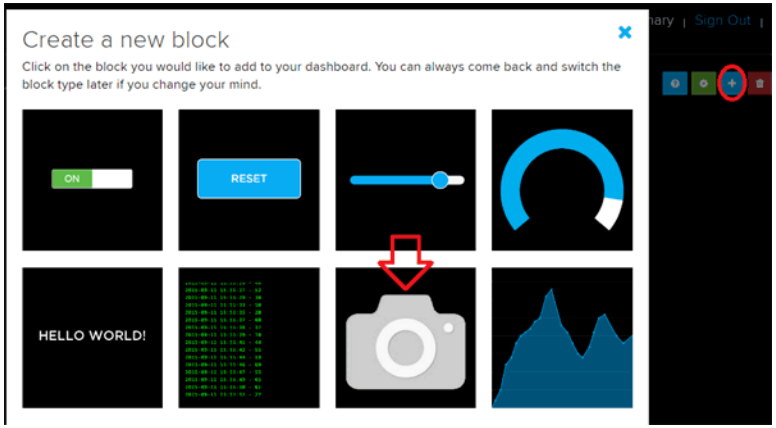


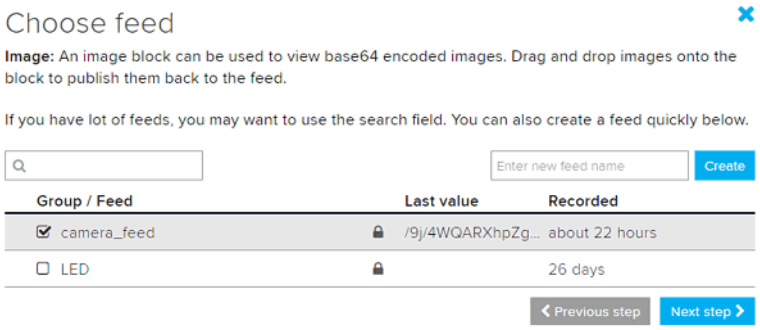
Stream video to Adafruit IO dashboard

Step 1 Adafruit IO Setup

- For Adafruit IO setup, first sign up on **Adafruit IO** website (https://io.adafruit.com) by clicking on *'Get started for Free'* at the top right corner of the screen.
- Now after this you need to create a feed, which you can do by clicking on *'Feed'* . Click on 'Actions' to proceed and then click on 'Create a New Feed'.
- After creating a feed now you need to create a 'Dashboard'. To create dashboard click on the "Create Dashboard" button in the upper right corner and give name to the new dashboard like Pi Camera Dashboard.
- Now to add block click on '+ sign' on the top right corner of the screen and scroll down to the image block and click on Create.
- In the 'Choose feed' window, click on the feed named camera_feed and click on 'Next step'.
- After that click on Create Block to create the image block. Also, you can use Adafruit IO for developing various IoT projects.







Stream video to Adafruit IO dashboard

Step 2 Raspberry Pi Setup

- To setup the Pi, you need to install the Node.js on your Raspberry Pi. For installing the Node.js go to home directory using *'cd ~'* command and then download a pre-built Node.js package from the node-arm project and install it on the terminal.

```
cd ~
wget http://node-arm.herokuapp.com/node_latest_armhf.deb
sudo dpkg -i node_latest_armhf.deb
```

- To check, whether the Node.js is installed or not use the below commands:

```
node –v
npm –v
```

- This will show you the current version of Node.js.
- After installing the Node.js, now you need to install 'adafruit-io-camera' on your Pi. This tool is used to monitor the Pi camera and send pictures to an Adafruit IO feed.

```
sudo apt-get update
sudo apt-get install -y imagemagick
sudo npm install --global --no-optional forever forever-service adafruit-io-camera
```

- After this use your 'Username' and 'AIO Key' (password) to provide a path to your Adafruit IO account.

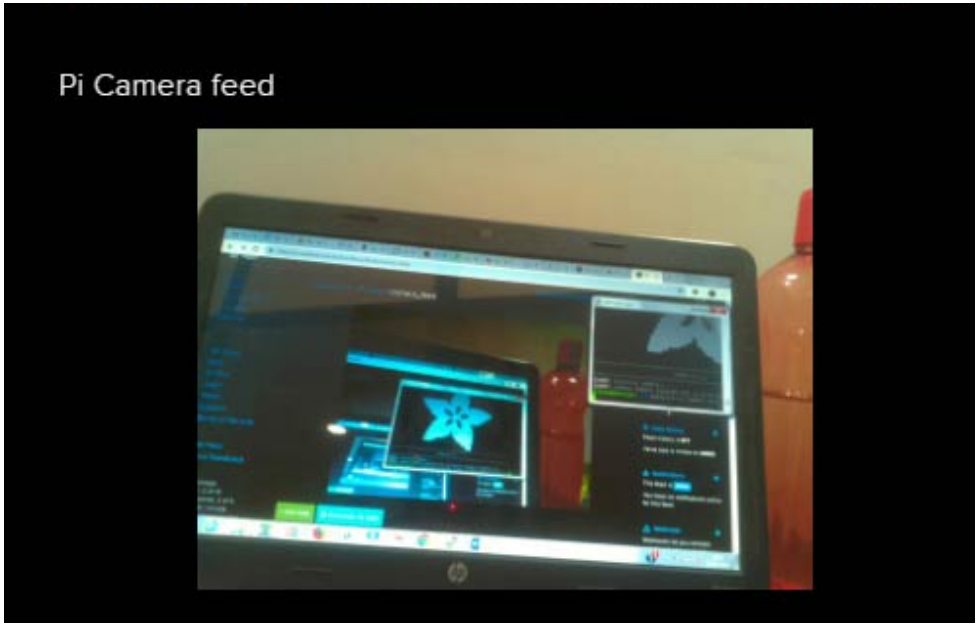
```
adafruit-io camera config --username YOUR USERNAME --key AIO KEY
```

- Enter your username and AIO key on the place of 'YOUR USERNAME' and 'AIO KEY' respectively.

Stream video to Adafruit IO dashboard

- Now start your camera using the following command:
  adafruit-io camera start -f camera_feed -m false -r 2
- Here ‘camera_feed’ is the name of feed that you have created earlier. You can change it according to your feed name.

- Now navigate to your Adafruit IO dashboard. It takes some time for the camera to start and upload images on the dashboard.



Hands – on Activity – VII

Temperature and Humidity IoT Application

(ThinkSpeak)

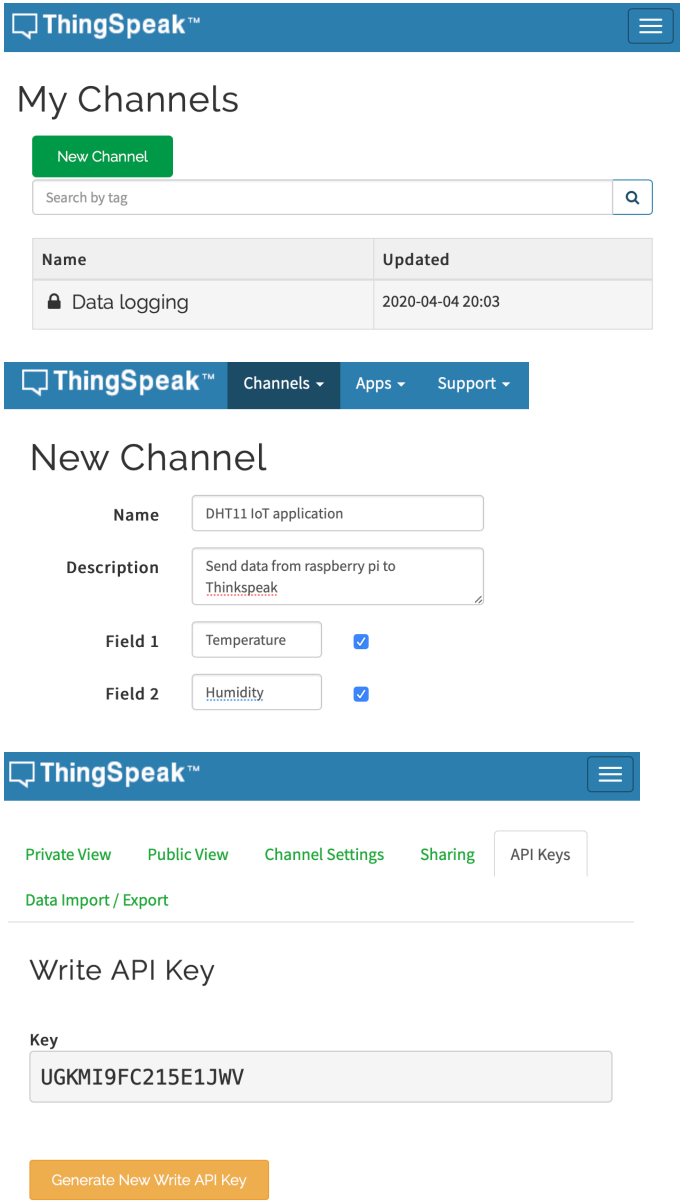
ThinkSpeak

- ThingSpeak is a Web Service (REST API) and IoT analytics platform service that lets you collect and store sensor data in the cloud and develop Internet of Things applications.
- Thingspeak channel is the core element of the thingspeak platform. This channel is used to store the real-time data, or the data transferred through various sensors and embedded systems. Data stored at the channel is further used for analysis and visualization.

ThinkSpeak Setup

- First sign up on **ThinkSpeak** website (https://thinkspeak.com) by clicking on *'Get started for Free'* at the middle of the screen.
- Now after this you need to create a new channel, which you can do by clicking on *'New Channel'* and select field1 and field2, then hit save button at the end.
- After creating a channel, select DHT11 IoT application and copy the API key which is under *"API Keys"* tab. If there is no key, hit the *"Generate*
- *New Write API Key" We are going to use this API key in our code.*
- After that open, the IoT_Temp_Humi.py file and replace the star string with your API key number.

```
import httplib, urllib
import time
import Adafruit_DHT
sleep = 30 # how many seconds to sleep between posts to the channel
key = '****************'  # Write API key

humidity, temperature = Adafruit_DHT.read_retry(11, 27)  # GPIO27 (BCM nota
```



Raspberry Pi DHT11 Setup

- The digital temperature and humidity sensor DHT11 is a composite sensor that contains a calibrated digital signal output of temperature and humidity.

To run the code after building the circuit.

- We need to download the Adafruit DHT11 library with typing below command in the terminal

git clone https://github.com/salihsarp/Adafruit_Python_DHT.git

- Change directory to

cd Adafruit_Python_DHT

- Then type below commands

sudo apt-get install build-essential python-dev # python2
sudo apt-get install build-essential python3-dev # python3

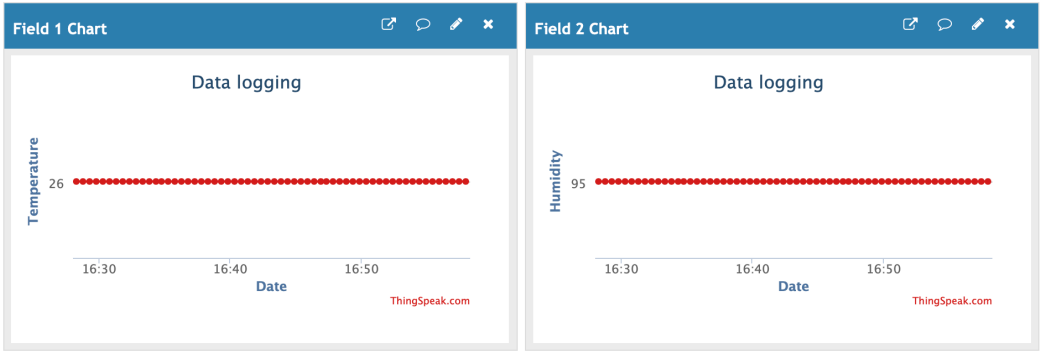
- To install the library, in the terminal, type the following.

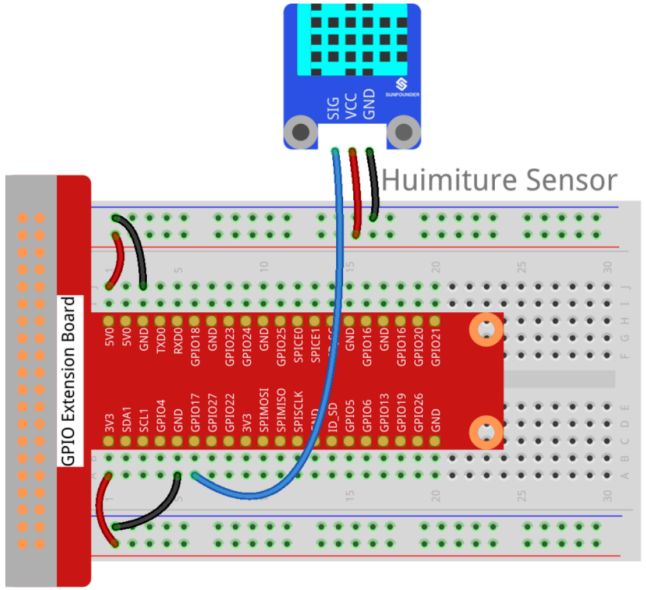
sudo python setup.py install # python2
sudo python3 setup.py install # python3

- Run the code

sudo python IoT_Temp_Humi.py

- Temperature and Humidity charts on ThinkSpeak website will be updated as shown below.





Components

- 1 * Raspberry Pi
- 1 * Breadboard
- 4 * Jumper wires (Male to Male, 2 red and 2 black)
- 1 * Humiture module (DHT11)
- 1 * 3-Pin anti-reverse cable

More Things to Try

- https://pimylifeup.com/raspberry-pi-weather-station/
  - See “Improving your weather station – Utilizing the LED Matrix”
- https://pimylifeup.com/raspberry-pi-twitter-bot/
  - See “Tweeting Webcam Photos via your Raspberry Pi Twitter Bot”