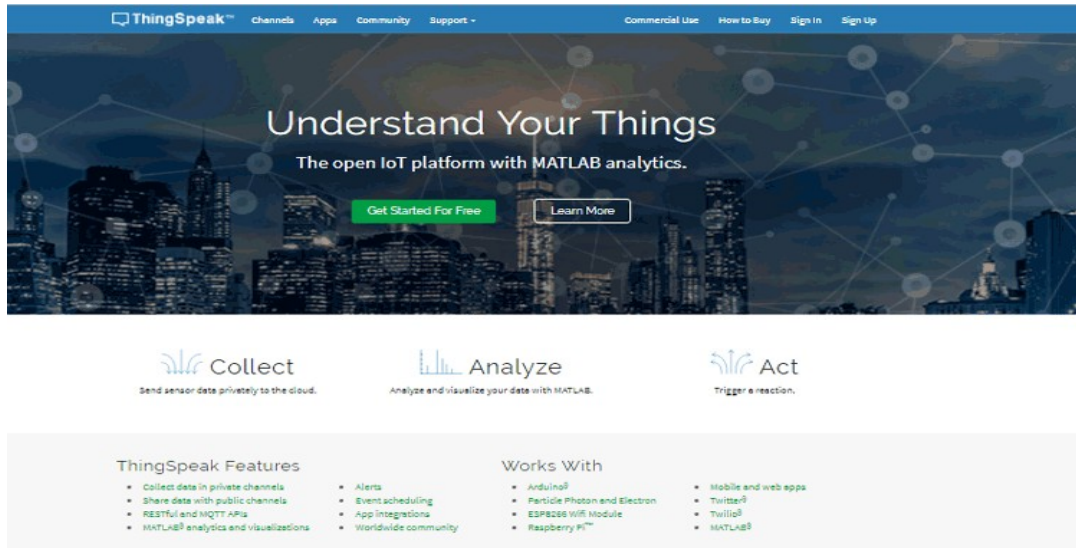


Steps for building Raspberry Pi Data Logger on Cloud

Step 1: Signup for ThingSpeak

For creating your channel on ThingSpeak you first need to sign up on ThingSpeak. In case if you already have account on ThingSpeak just sign in using your id and password.

For creating your account go to www.thingspeak.com



Click on signup if you don't have account and if you already have account click on sign in.

After clicking on signup fill your details.

Step 2: Create a Channel for Your Data

Once you Sign in after your account verification, Create a new channel by clicking “New Channel” button

New Channel

Name: CPU data

Description: To Send CPU data

Field 1: Field Label 1

Field 2:

Field 3:

Field 4:

Field 5:

Field 6:

Field 7:

Field 8:

Metadata:

Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- Channel Name:** Enter a unique name for the ThingSpeak channel.
- Description:** Enter a description of the ThingSpeak channel.
- Field#:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- Tags:** Enter keywords that identify the channel. Separate tags with commas.
- Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- Show Channel Location:**
 - Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.
 - Longitude:** Specify the longitude position in decimal degrees. For example, the longitude of the city of London is -0.1275.
 - Elevation:** Specify the elevation position meters. For example, the elevation of the city of London is 35.052.
- Video URL:** If you have a YouTube™ or Vimeo® video that displays your channel information, specify the full path of the video URL.

After clicking on “New Channel”, enter the Name and Description of the data you want to upload on this channel. For example I am sending my CPU data (temperature), so I named it as CPU data.

Now enter the name of your data (like Temperature or pressure) in Field1. If you want to use more than one Field you can check the box next to Field option and enter the name and description of your data.

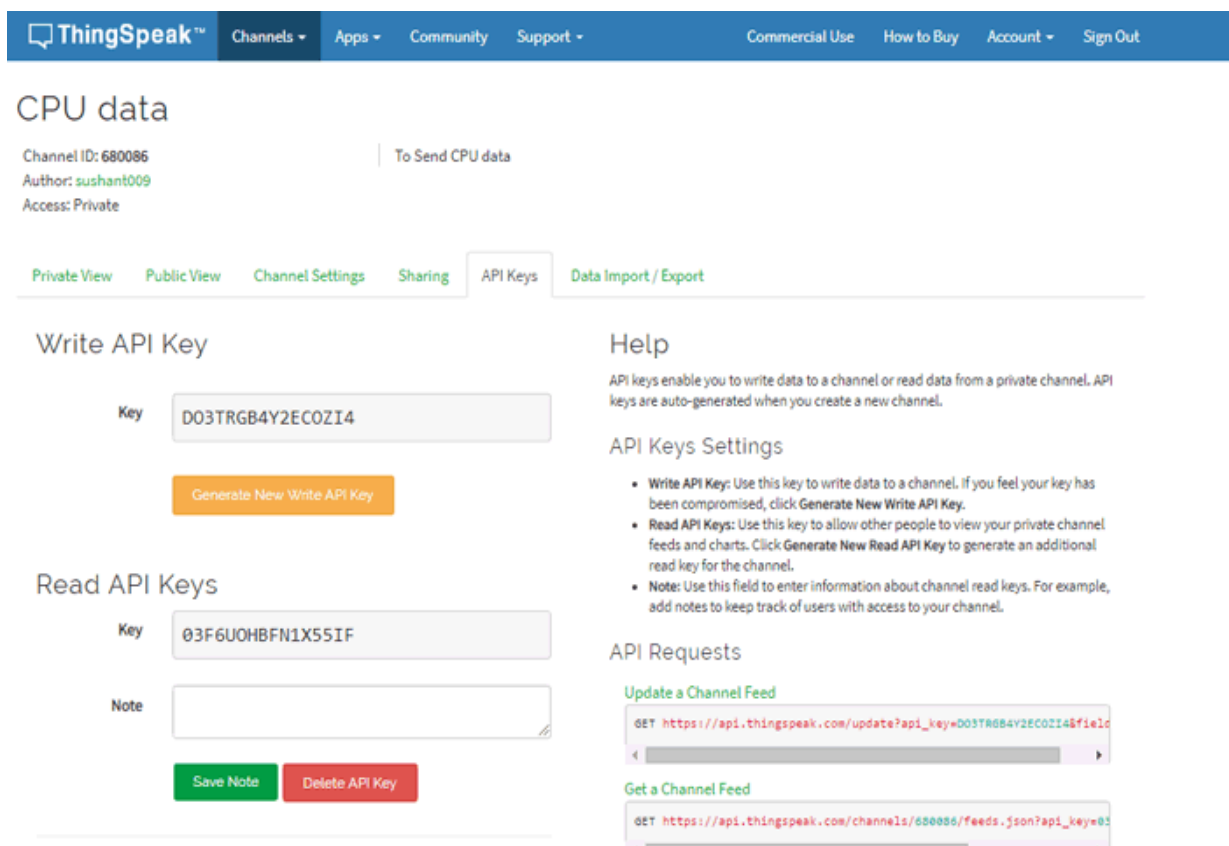
After this click on save channel button to save your details.

Step 3: Getting API Key in ThingSpeak

To send data to ThingSpeak, we need an unique API key, which we will use later in our python code to upload our CPU data to ThingSpeak Website.

Click on “API Keys” button to get your unique API key for uploading your CPU data.

Getting API Key in ThingSpeak



The screenshot shows the ThingSpeak website interface for a channel named "CPU data". The channel ID is 680086, the author is sushant009, and the access is Private. The "API Keys" tab is selected, showing two sections: "Write API Key" and "Read API Keys".

Write API Key: A text box contains the key "D03TRGB4Y2ECOZI4". Below it is a button labeled "Generate New Write API Key".

Read API Keys: A text box contains the key "03F6UOHBFN1X55IF". Below it is a text area for a note, and two buttons: "Save Note" and "Delete API Key".

Help: A section explaining that API keys enable writing data to a channel or reading data from a private channel. It lists "API Keys Settings" with instructions for Write API Key, Read API Keys, and a Note field.

API Requests: A section showing two example API requests:

```
Update a Channel Feed
GET https://api.thingspeak.com/update?api_key=D03TRGB4Y2ECOZI4&field1=

Get a Channel Feed
GET https://api.thingspeak.com/channels/680086/feeds.json?api_key=03
```

Now copy your “Write API Key”. We will use this API key in our code.

Step 4: Python Code for Raspberry Pi

```
import httplib
import urllib
import time
key = "4QWRBTCPMOJDPSF" # Put your API Key here
def thermometer():
    while True:
        #Calculate CPU temperature of Raspberry Pi in Degrees C
        temp = int(open('/sys/class/thermal/thermal_zone0/temp').read()) / 1e3

        # Get Raspberry Pi CPU temp
        params = urllib.urlencode({'field1': temp, 'key':key })
        headers = {"Content-type": "application/x-www-form-urlencoded", "Accept": "text/plain"}
        conn = httplib.HTTPConnection("api.thingspeak.com:80")
        try:
            conn.request("POST", "/update", params, headers)
            response = conn.getresponse()
            print temp
            print response.status, response.reason
            data = response.read()
            conn.close()
        except:
            print "connection failed"
            break
# Boiler plate code

if __name__ == "__main__":
    while True:
        thermometer()
```

Step 5 : Execute the Python code on Raspberry Pi

- i. Assume you have booted the Raspberry Pi with the Raspbian OS and it is up and running
- ii. Open a terminal
- iii. Assuming you already installed python in Raspberry pi using this command, if not then

sudo apt-get install python

Case 1: If you are using monitor screen then just use the given code.

Now install all libraries:

```
sudo apt-get install httplib
```

```
sudo apt-get install urllib
```

iv. After installing libraries run your python code at the terminal

```
python /path/cpu.py
```

If the code runs properly you will see some CPU temperature values as shown in below image.

```
200 OK
46.16
200 OK
46.16
200 OK
46.16
200 OK
46.16
200 OK
46.16
200 OK
46.698
200 OK
46.16
200 OK
46.698
200 OK
46.698
200 OK
46.16
200 OK
46.16
200 OK
46.16
200 OK
46.16
200 OK
```

Step 6: Check ThingSpeak site for Data Logging

After completing these steps open your channel and you will see the CPU temperature data is updating into ThingSpeak website. Check ThingSpeak site for Data Logging

Channel Stats

Created: [about an hour ago](#)

Entries: 49

