MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
Versova, Andheri, Mumbai – 400053

Department of Instrumentation Engineering

Laboratory Manual

INTERNET OF THINGS

DEPARTMENT OF INSTRUMENTATION ENGINEERING

2020

MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
Versova, Andheri, Mumbai – 400053

Department of Instrumentation Engineering

INTERNET OF THINGS

Subject code	Subject Name	Teaching scheme			Credit assigned			
		Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
ISL803	Internet of Things- Lab Practice	-	02	-	-	1	-	1

Sub Code	Subject Name	Examination scheme							
		Internal Assessment			End Sem Exam	Term work	Pract. And oral	Oral	Total
		Test1	Test2	Avg.					
ISL803	Internet of Things- Lab Practice	-	-	-	-	25	-	25	50

Subject Code	Subject Name	Credits
ISL803	Internet of Things- Lab Practice	1
Course objectives	1. To impart knowledge about fundamentals of IoT 2. To describe data and knowledge management and use of devices in IoT technology. 3. To give knowledge of IoT architecture and Integration of embedded devices with IoT 4. To explain the concept of IIoT. 5. To impart knowledge about designing of industrial internet systems. 6. To describe overview of Android/ IOS app development tools and Internet of Everything	
Course Outcomes	The students will be able to : 1. Use microcontroller based embedded platforms in IOT 2. Use microprocessor based embedded platforms in IOT 3. Use wireless peripherals for exchange of data. 4. Make use of Cloud platform to upload and analyse any sensor data 5. Use of Devices, Gateways and Data Management in IoT. 6. Use the knowledge and skills acquired during the course to build and test a complete, working IoT system involving prototyping, programming and data analysis.	

Course Objectives:

1. To teach fundamentals of IoT.
2. To study data and knowledge management and use of devices in IoT technology.
3. To understand IoT architecture and integration of embedded devices with IoT
4. To understand concept of IoT
5. To learn designing of industrial internet systems
6. To study overview of Android/ IOS app development tools and Internet of everything.

Course Outcomes:

After completion of the course students will be able to:

1. Demonstrate knowledge of operation of IoT architecture.
2. Identify various technologies for implementing IoT
3. Discuss various communication technologies available in IoT
4. Discuss various communication models and protocols used in IoT
5. Discuss role of cloud computing in IoT.
6. Illustrate application of IoT in industrial automation and identify real world design constraints.

Practical/Oral Examination:

Practical/Oral examination will be based on entire syllabus.

Term Work:

Term work shall consist of minimum 08 experiments from the above given list and 02 assignments from imaging techniques module and electrical safety module.

The distribution of marks for term work shall be as follows:

Laboratory work (Experiments)	: 10 Marks
Laboratory work (programs /journal)	: 10 Marks
Attendance	: 5 Marks

The final certification and acceptance of term work ensures the satisfactory performance of laboratory work and minimum passing in the term work.

MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
Versova, Andheri, Mumbai – 400053

Department of Instrumentation Engineering

INTERNET OF THINGS SEM: VIII

LIST OF LABORATORY EXPERIMENTS:

SR.NO	NAME OF EXPERIMENT	PAGE NO.
1	**Introduction to Arduino** platform and programming	
2	Interfacing **LDR sensor and LED** with Arduino	
3	Interfacing **accelerometer** sensor with Arduino	
4	Interfacing **gyroscope** sensor with Arduino	
5	Interfacing Arduino to **Zigbee** module	
6	Interfacing Arduino to **GSM** module	
7	Interfacing Arduino to **Bluetooth** Module	
8	Introduction to **Raspberry PI platform** and python programming	
9	Interfacing **sensors to Raspberry PI**	
10	Setup a **cloud platform** to log the data	
11	Log Data using **Raspberry PI and** upload to the **cloud platform**	
12	**Design an IOT based system**	

AIM: USING THINGSPEAK AS A CLOUD PLATFORM TO LOG DATA FROM RASPBERRY PI

APPARATUS: Hardware

- A Raspberry Pi computer with an SD card or micro SD card
- A monitor with a cable (and, if needed, an HDMI adaptor)
- A USB keyboard and mouse
- A power supply
- Headphones or speakers (optional)
- An ethernet cable (optional)

Software

- Raspbian, installed via NOOBS
- **ThingSpeak**
- WiFi or Internet

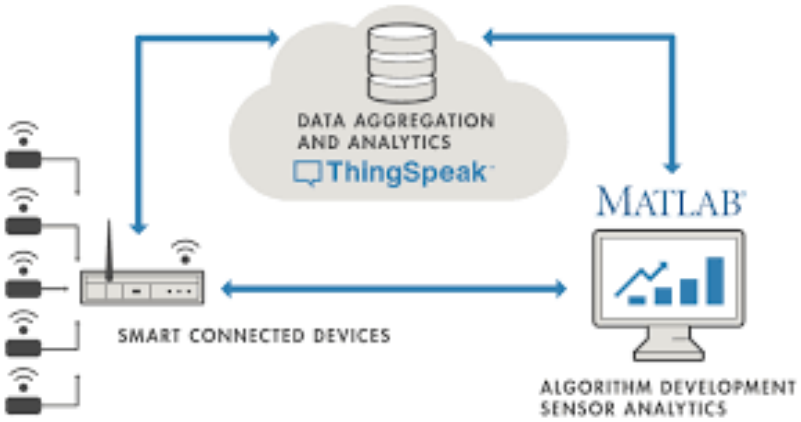
THEORY:

Connect your Raspberry Pi

- Check the slot on the underside of your Raspberry Pi to see whether an SD card is inside. If no SD card is there, then insert an SD card with Raspbian installed (via NOOBS).
- Find the USB connector end of your mouse's cable, and connect the mouse to a USB port on your Raspberry Pi (it doesn't matter which port you use).
- Connect the keyboard in the same way.
- Make sure your screen is plugged into a wall socket and switched on.
- Look at the HDMI port(s) on your Raspberry Pi — notice that they have a flat side on top.
- Use a cable to connect the screen to the Raspberry Pi's HDMI port — use an adapter if necessary.
- If you want to connect the Pi to the internet via Ethernet, use an Ethernet cable to connect the Ethernet port on the Raspberry Pi to an Ethernet socket on the wall or on your internet router. You don't need to do this if you want to use wireless connectivity, or if you don't want to connect to the internet.
-
- If your screen has speakers, your Raspberry Pi can play sound through these. Or you could connect headphones or speakers to the audio port.
- Plug the power supply into a socket and then connect it to your Raspberry Pi's USB power port.
- You should see a red light on your Raspberry Pi and raspberries on the monitor.
- Your Raspberry Pi then boots up into a graphical desktop.

ThingSpeak for IoT

ThingSpeak™ is an IoT analytics platform service from MathWorks®, the makers of MATLAB® and Simulink®. ThingSpeak allows you to aggregate, visualize, and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices or equipment. Execute MATLAB code in ThingSpeak, and perform online analysis and processing of the data as it comes in. ThingSpeak accelerates the development of proof-of-concept IoT systems, especially those that require analytics. You can build IoT systems without ThingSpeak provides a hosted solution that can be used in production.



In this simplest **Raspberry Pi IOT project**, we will use ThingSpeak as cloud server to store the data. Here Raspberry Pi will read its CPU temperature and send it to ThingSpeak, and it can be monitored from anywhere in the world using internet. This will be useful if you are running the Pi for long time for some application at some remote place and need monitor its CPU temperature.

ThingSpeak is an open IoT platform for monitoring your data online. In ThingSpeak channel you can set the data as private or public according to your choice. ThingSpeak takes minimum of 15 seconds to update your readings. Its a great and very easy to use platform for building IOT projects.

some advantages of ThingSpeak over other platforms is, it is very easy to set up and it also has many options to plot graph.

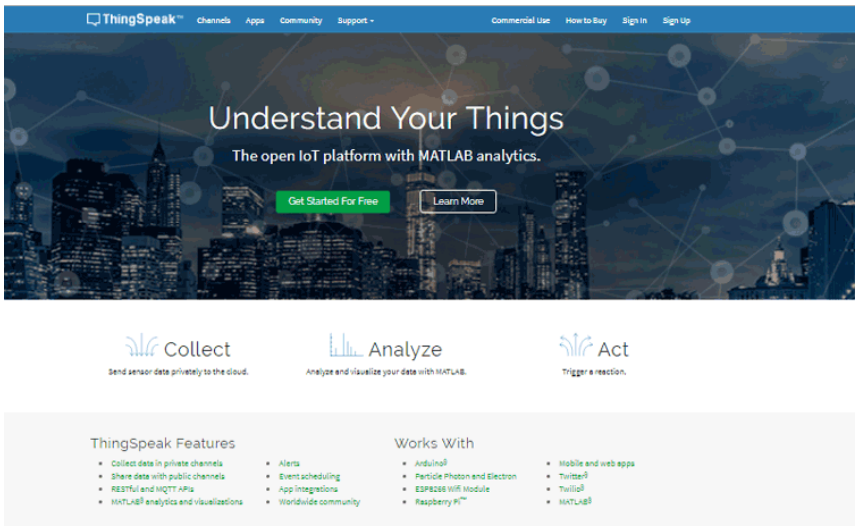
PROCEDURE:

Steps for building Raspberry Pi Data Logger on Cloud

Step 1: Signup for ThingSpeak

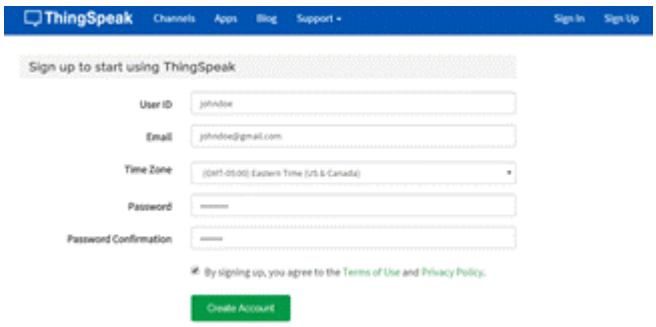
For creating your channel on ThingSpeak you first need to sign up on ThingSpeak. In case if you already have account on ThingSpeak just sign in using your id and password.

For creating your account go to www.thinspeak.com



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

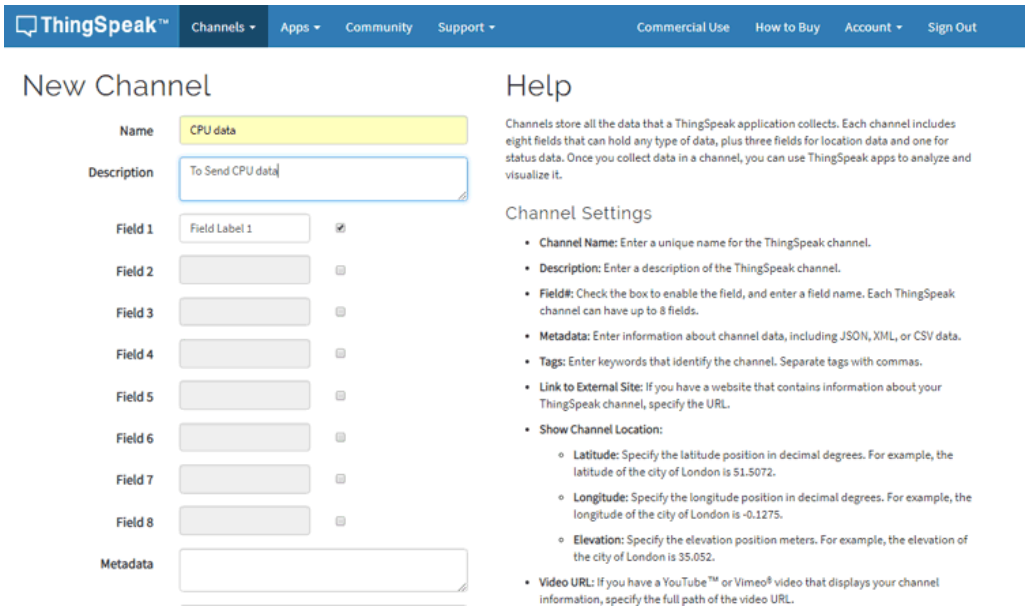
After clicking on signup fill your details.



After this verify your E-mail id and click on continue.

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



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.

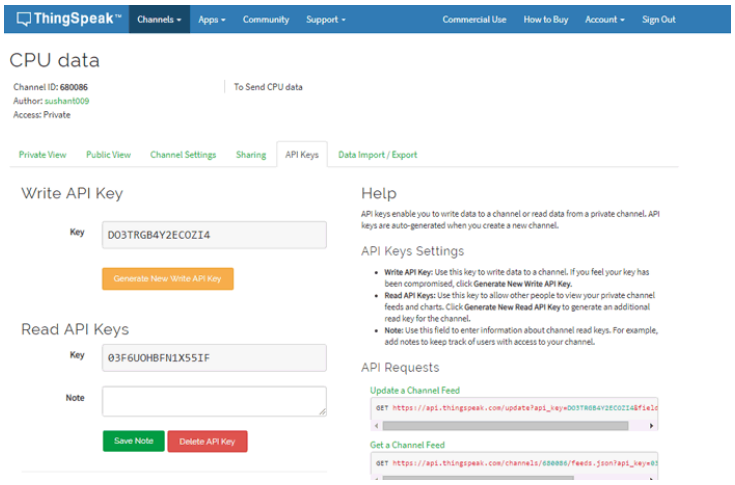
Now enter the name of your data (like Temperature or pressure) in Field1. If you want to use more than one Field you can check the box next to Field option and enter the name and description of your data.

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.



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

Step 4: Python Code for Raspberry Pi

Complete code is given at the end of this tutorial, just make a file with any name and .py extension and copy-paste the code and save the file. Don't forget to replace the API key with yours. You can run the python file any time using below command:

```
python /path/filename.py
```

Assuming you already installed python in Raspberry pi using this command

```
sudo apt-get install python
```

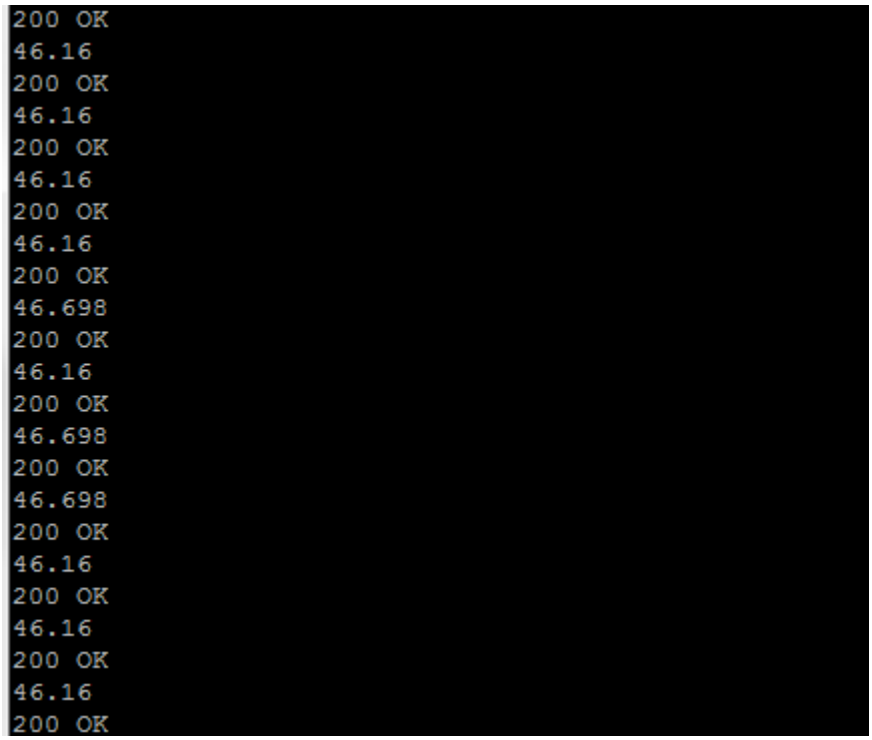
Case 1: If you are using monitor screen then just use the given code.

Now install all libraries:

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

After installing libraries run your python code (*python /path/filename.py*)

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



If there are any errors uploading the data, you will receive "connection failed" message.

Case 2: If you are using "Putty" then you should follow these commands:

First update your pi using:

```
sudo apt-get update
```

After this make a file cpu.py using:

```
nano cpu.py
```

After creating this file copy your code to this file and save it using CTRL + X and then 'y' and Enter.

After this install all libraries using:

```
sudo apt-get install httplib

sudo apt-get install urllib
```

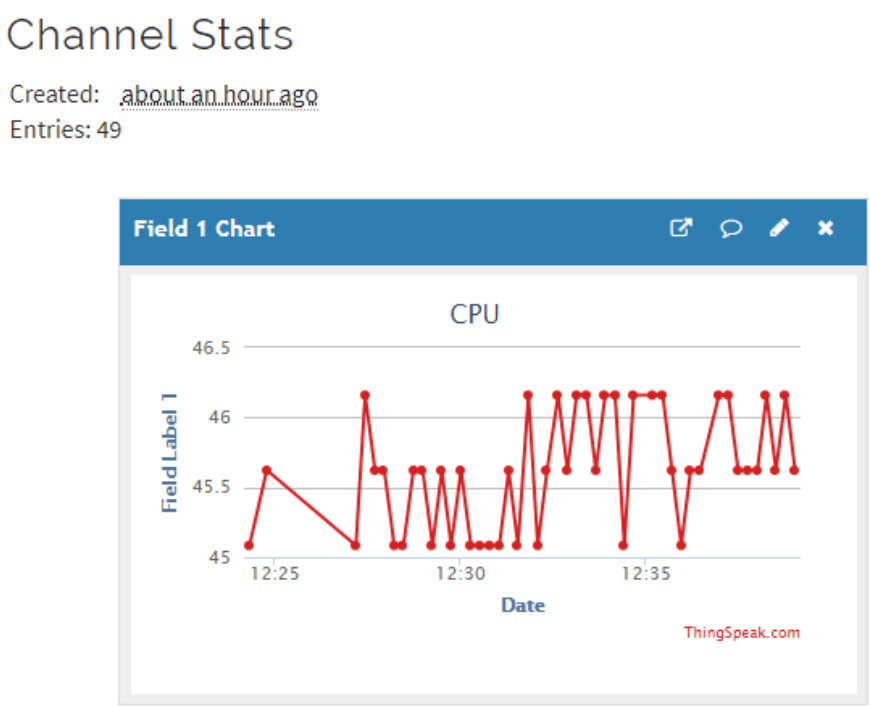
After installing libraries run your python code using:

```
python cpu.py
```

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

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.



Like this you can send any sensor data connected with Raspberry pi to the ThingSpeak Cloud. In next article we will connect LM35 temperature sensor with Raspberry Pi and send the temperature data to ThingSpeak, which can be monitored from anywhere.

Complete **Python code** for this **Raspberry Pi Cloud Server** is given below. Code is easy and self-explanatory to understand.

Code

```
import httplib
import urllib
import time
key = "ABCD"  # 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-typZZe": "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
if __name__ == "__main__":
      while True:
            thermometer()
```

Conclusion: ThingSpeak allows you to aggregate, visualize and analyze live data streams in the cloud. With ThingSpeak, your data is stored in channels. Each channel stores up to 8 fields of data. You can create as many channels as you need for your application. ThingSpeak stores all the information you send it in one central location in the cloud, so you can easily access your data for online or offline analysis. Your private data is protected with an API key that you control. When you are logged in to your ThingSpeak account, you can use the web to securely download the data stored in the cloud. You can also programmatically read your data in CSV or JSON formats using a REST API call and the appropriate API key. Your devices can also read data from a ThingSpeak channel by subscribing to an MQTT topic. Import data from third-party web services including climate data from NOAA, public utility data from local utility providers, and stock and pricing data from financial providers. You can use that data together with the data you are collecting from your devices and equipment to investigate correlations and develop predictive algorithms.