



# **IoT Physical Devices and Raspberry Pi with Python**

**By,**

**Dr. Vidya Rao  
Assistant Professor,  
Dept of DSCA, MIT, MAHE**



# Outline

- Basic building blocks of an IoT Device
- Exemplary Device: Raspberry Pi
- Raspberry Pi interfaces
- Programming Raspberry Pi with Python
- Other IoT devices

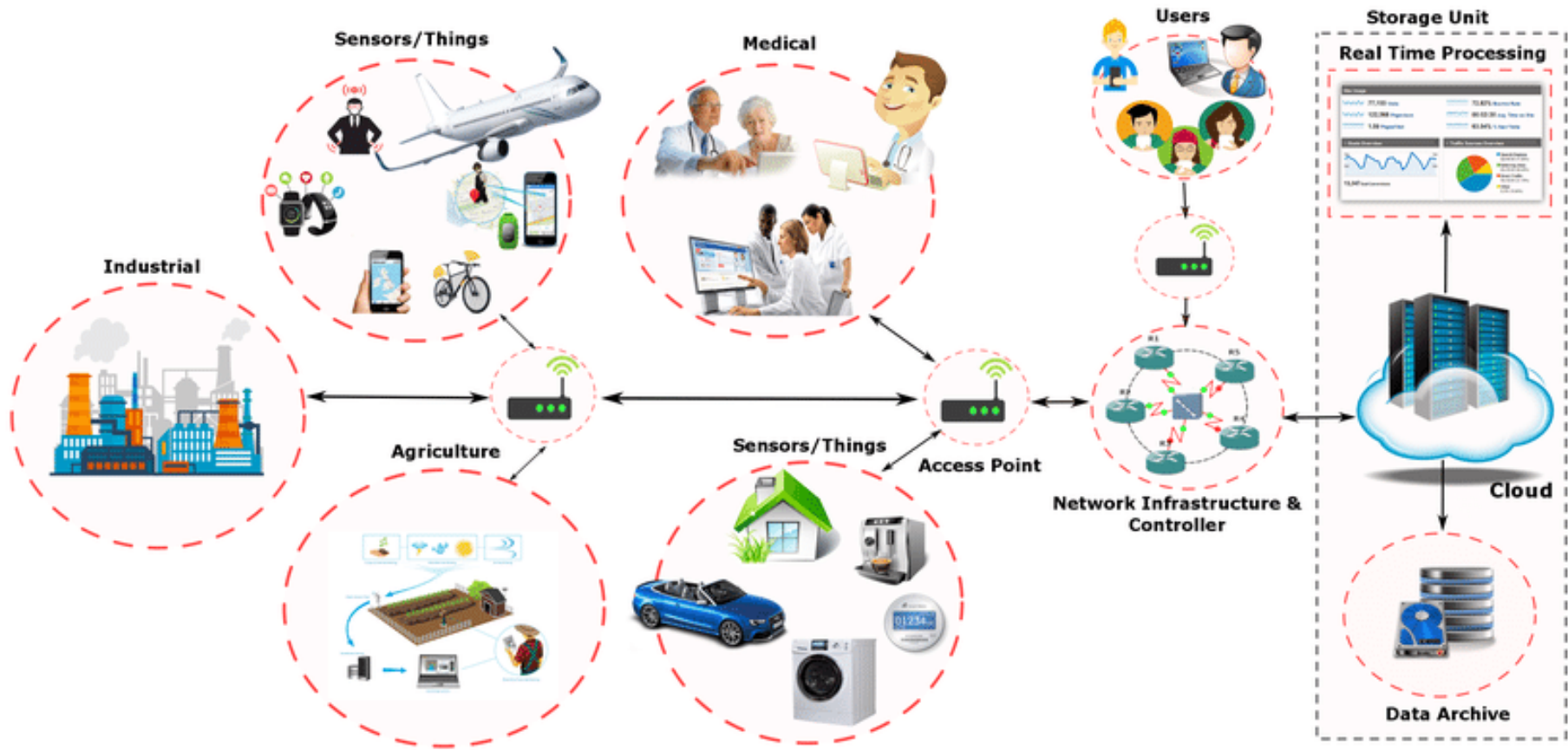


# What is an IoT Devices?

- A "Thing" in the Internet of Things (IoT) can be any object that has a unique identifier and which can send/receive data (including user data) over a network (e.g., smartphone, smart TV, computer, refrigerator, car, etc. ).
- IoT devices are connected to the Internet and send information about themselves or about their surroundings (e.g. information sensed by the connected sensors) over a network (to other devices or servers/storage) or allow actuation upon the physical entities/environment around them remotely.



# IoT Device Examples





# Basic building blocks of an IoT Device

- **Sensing**

- Sensors can be either on board the IoT device or attached to the device.

- **Actuation**

- IoT devices can have various types of actuators attached that allow taking
- actions upon the physical entities in the vicinity of the device.

- **Communication**

- Communication modules are responsible for sending collected data to other devices or cloud-based servers/storage and receiving data from other devices and commands from remote applications.

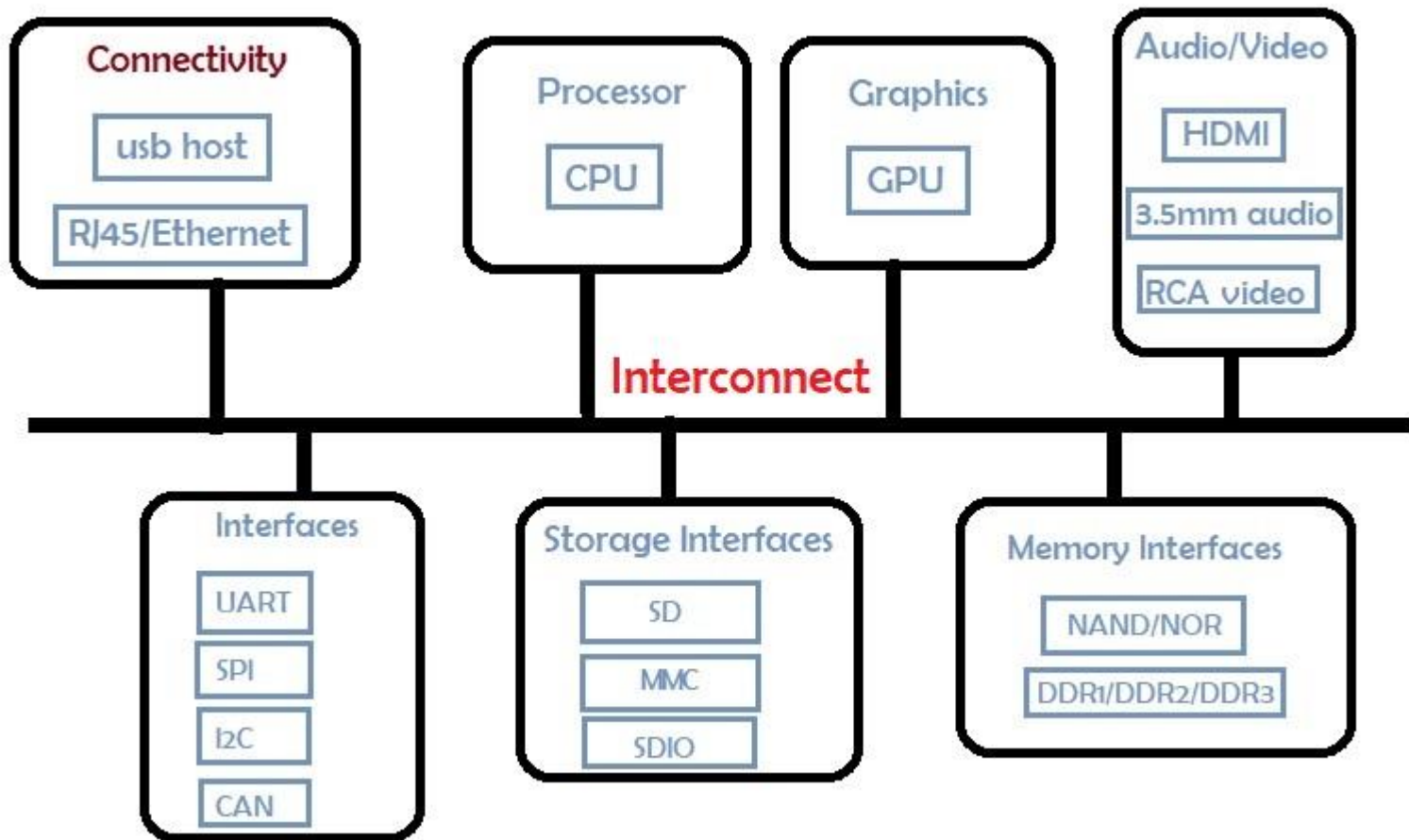
- **Analysis & Processing**

- Analysis and processing modules are responsible for making sense of the collected data.





# Block diagram of an IoT Device





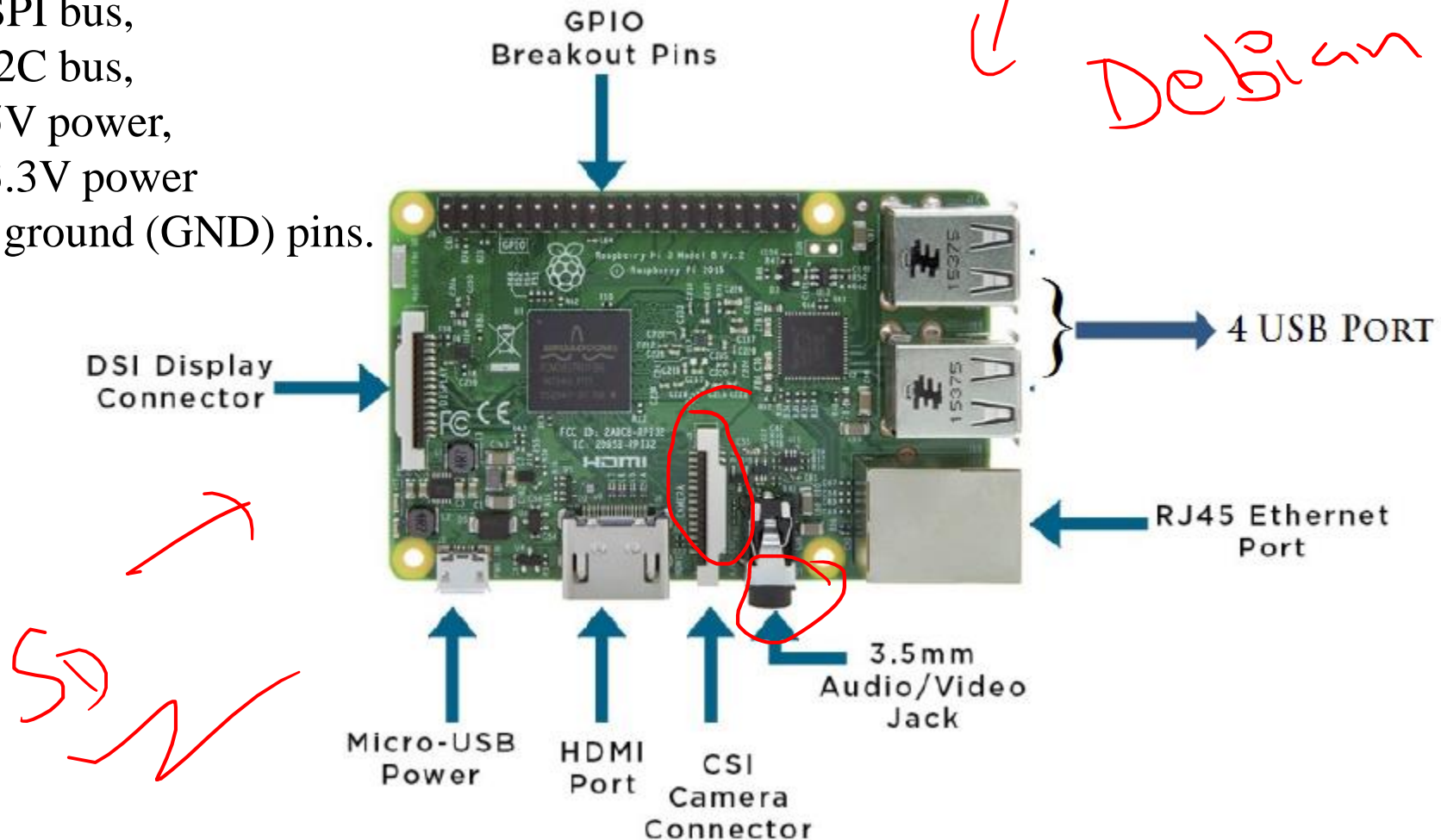
# Exemplary Device: Raspberry Pi

- Raspberry Pi is a low-cost mini-computer with the physical size of a credit card.
- Raspberry Pi runs various flavors of Linux and can perform almost all tasks that a normal desktop computer can do.
- Raspberry Pi also allows interfacing sensors and actuators through the general-purpose I/O pins.
- Since Raspberry Pi runs Linux operating system, it supports Python "out of the box".

# Raspberry Pi 3

- RPi3 has twenty four GPIO pins,
- one serial UART,
- two SPI bus,
- one I2C bus,
- two 5V power,
- two 3.3V power
- eight ground (GND) pins.

Raspbian OS  
↓  
Debian







# Raspberry Pi 3 (contd..)

- **RAM & processor:** RPi3 has 1GB RAM with 700 MHz Low power ARM1176JZ-F processor.
- **USB Ports:** It has four 2.0 USB ports thus providing all the required peripherals like keyboards, monitor and mouse to be connected at a time.
- **Ethernet Port:** It has one Ethernet port that can be connected with the LAN adapter else it can also use in-built WiFi adapter.
- **Display Serial Interface(DSI):** It is used to connect LCD screen to RPi3.
- **Camera Serial Interface(CSI):** It is used to connect camera to RPi3.
- **SD card slot:** A minimum 8GB storage SD is required to store the image of Linux OS that is used to run the RPi3.



# Raspberry Pi 3 (contd..)

- **General Purpose Input-Output (GPIO) Pins:** These pins enable RPi3 to establish connections with various sensors and actuators. It contains true GPIO pins, I2C interface pins, SPI interface pins and one of serial Rx pin for receiving and a Tx pin for transmitting.
- **High-Definition Multimedia Interface (HDMI) output:** It helps RPi3 to provide both video and audio output. Also, RPi3 can be connected to PC monitor using HDMI cable adapter.
- **Composite video output:** This helps RPi3 to connect to composite video output with RCA jack that supports PAL and NTSC video input.
- **Composite video output:** This helps RPi3 to connect to composite video output with RCA jack that supports PAL and NTSC video input.
- **Audio output:** It has 3.5mm audio jack which can be connected to old RCA television to get better quality than the current monitors.



# Raspberry Pi 3 (contd..)

- **Power Input:** A micro USB power input is used for connecting power that takes 5.1V at 2.5A.
- **Status LED:** RPi3 has five status LED with functionalities shown in the Table 1.
- Rest of RPi3 specifications are shown in the Table 2.

Status LED	Function
ACT	SD card access
PWR	3.3V Power
FDX	Full Duplex LAN
LNK	Link/Network activity
100	100 Mbit LAN connected

Table 1: Status information

Table 2: Specification

Board	Specification	Architecture	ARMv8-A
Pi-3			
SoC	Broadcom	CPU	1.2 GHz quad-core
Core frequency	250 MHz	ARM	600 MHz
On-board network	Ethernet, WiFi 802.11 Bluetooth 4.1	Core OS	1.3062 V



# Raspberry Pi 3 Pin Configuration



	3.3V	● ●	5V
GPIO 2	● ●	5V	
GPIO 3	● ●	Ground	
GPIO 4	● ●	GPIO 14	
Ground	● ●	GPIO 15	
GPIO 17	● ●	GPIO 18	
GPIO 27	● ●	Ground	
GPIO 22	● ●	GPIO 23	
3.3V	● ●	GPIO 24	
GPIO 10	● ●	Ground	
GPIO 9	● ●	GPIO 25	
GPIO 11	● ●	GPIO 8	
Ground	● ●	GPIO 7	
---	● ●	---	
GPIO 5	● ●	Ground	
GPIO 6	● ●	GPIO 12	
GPIO 13	● ●	Ground	
GPIO 19	● ●	GPIO 16	
GPIO 26	● ●	GPIO 20	
Ground	● ●	GPIO 21	




# Raspberry Pi Foundations

- Raspberry Pi web page: <https://www.raspberrypi.com/>
- Documentation: <https://www.raspberrypi.com/documentation/>
- Foundation: <https://www.raspberrypi.org/>
- Feature courses: <https://www.raspberrypi.org/courses/featured>

← → ↻ raspberrypi.org

MCNS Workshop on IoT ~... Download Faceboo... Yahoo Mail Computer Science... Certificate in Intern... Hop-distance relati... PDF to Word Conve... PhD Admissions 20...

 **Raspberry Pi**  
Foundation

Learn Teach Research Computers About us

**Donate**

Find out more about our computers  
for home, industry and education at  
**raspberrypi.com**

**Visit raspberrypi.com**





# Linux on Raspberry Pi

*Handwritten red text: 1. Python*

- **Raspbian**

- Raspbian Linux is a Debian Wheezy port optimized for Raspberry Pi.

- **Arch**

- Arch is an Arch Linux port for AMD devices.

- **Pidora**

- Pidora Linux is a Fedora Linux optimized for Raspberry Pi.

- **RaspBMC**

- RaspBMC is an XBMC media-center distribution for Raspberry Pi.

- **OpenELEC**

- OpenELEC is a fast and user-friendly XBMC media-center distribution.

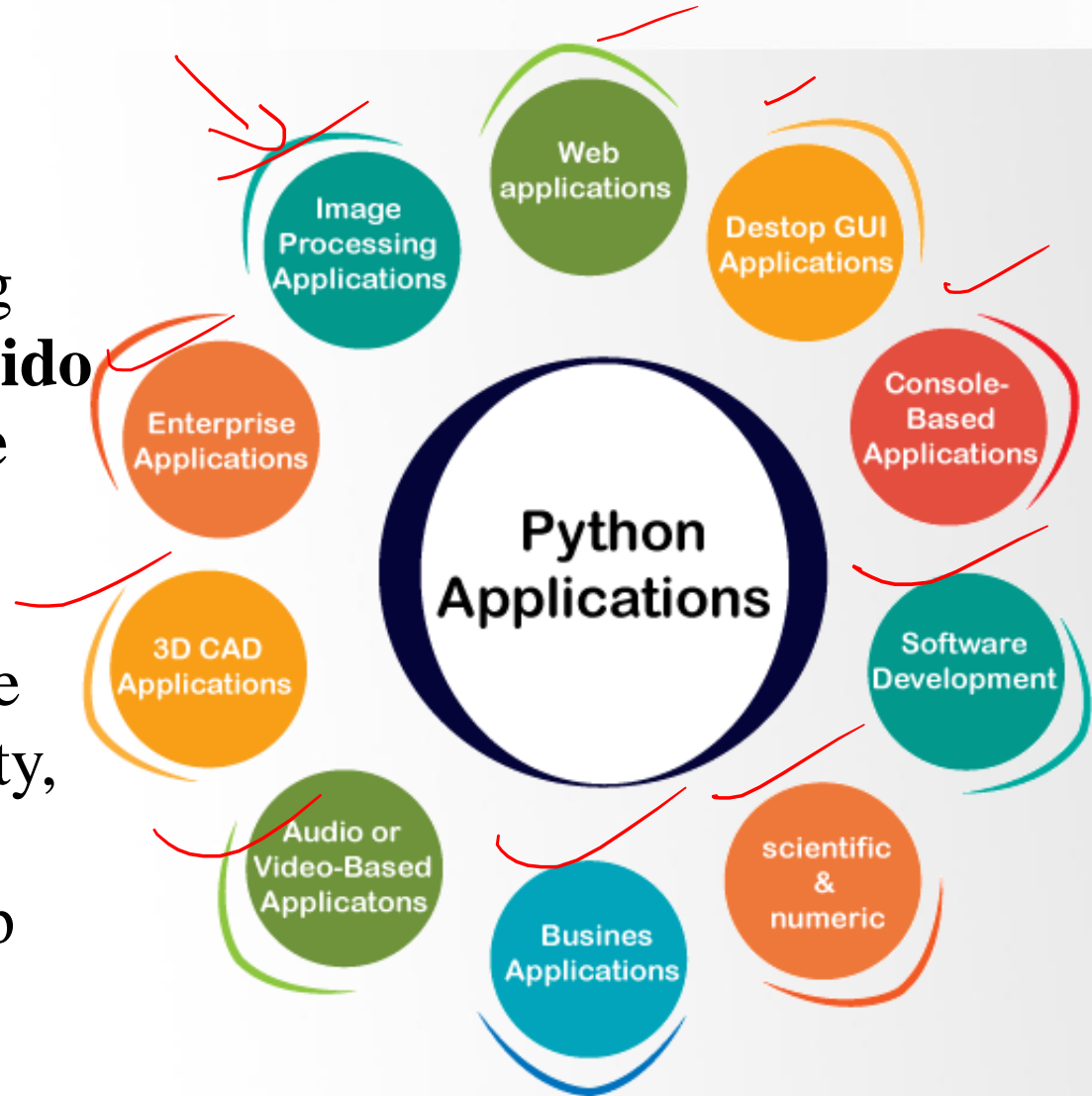
- **RISC OS**

- RISC OS is a very fast and compact operating system.



# Python

- Python is a widely used high-level programming language created by **Guido van Rossum** in the late **1980s**.
- The language places a strong emphasis on code readability and simplicity, making it possible for programmers to develop applications rapidly.



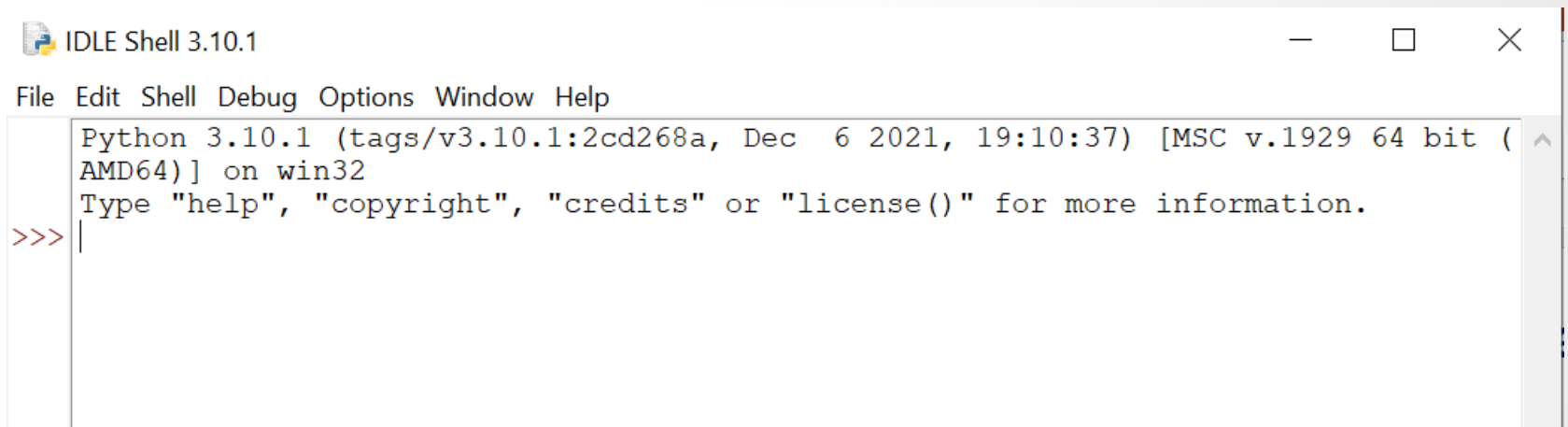


# Installing Python

- To install the interpreter for Python 3, head over to <https://www.python.org/downloads/>



- Python code uses the IDLE program that comes bundled with our Python interpreter.






# Installing PyCharm

- <https://www.jetbrains.com/pycharm/download/#section=windows>

PyCharm

What's New Features Learn Pricing **Download**



Version: 2022.1  
Build: 221.5080.212  
13 April 2022

[System requirements](#)  
[Installation instructions](#)

## Download PyCharm

[Windows](#) [macOS](#) [Linux](#)

### Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

**Download**

Free 30-day trial available

### Community

For pure Python development

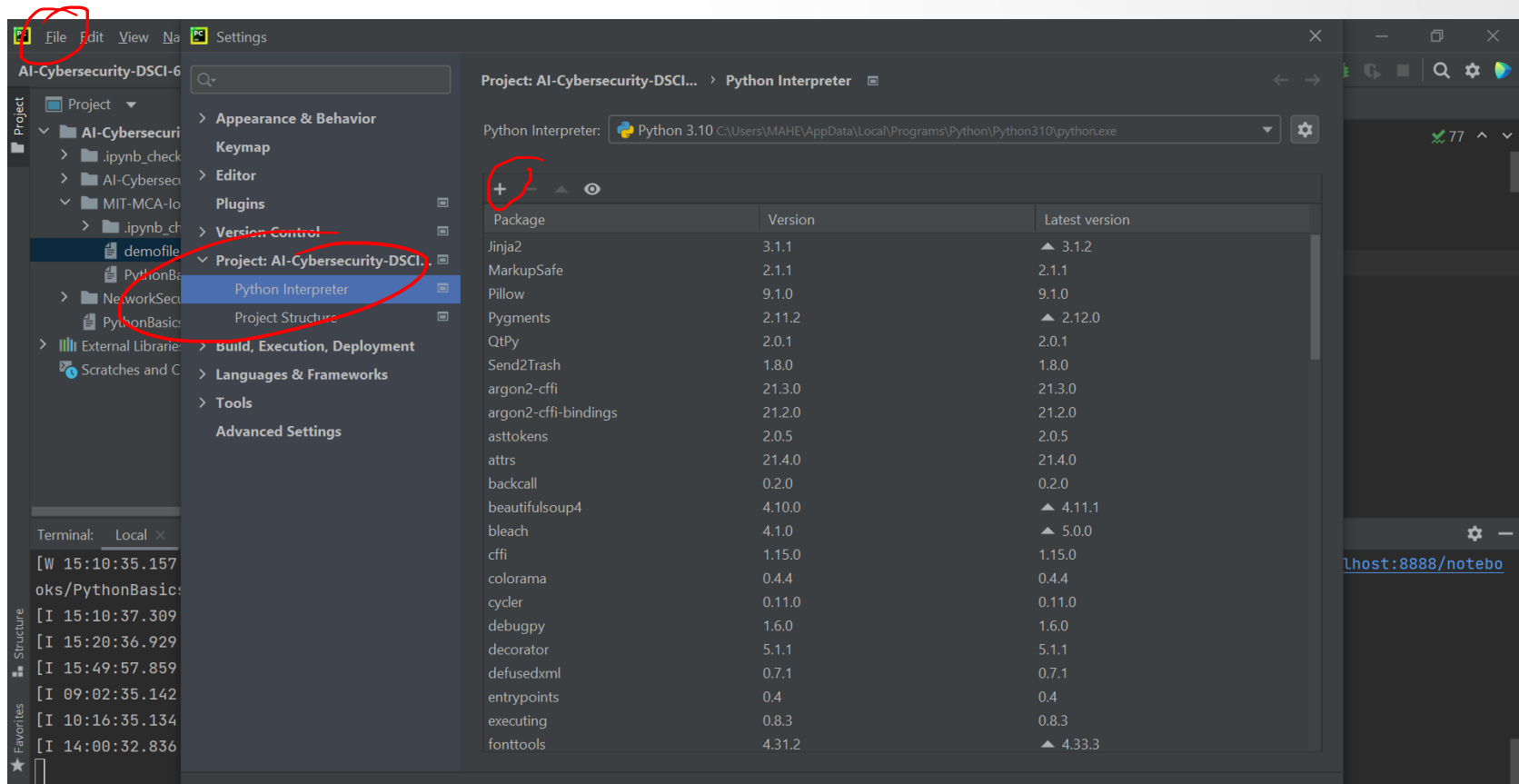
**Download**

Free, built on open-source



# PyCharm Environment

- File → Settings → YourProjectName → Python Interpreter → “+” → search and install packages.



Project: AI-Cybersecurity-DSCI... > Python Interpreter

Python Interpreter: Python 3.10 C:\Users\MAHE\AppData\Local\Programs\Python\Python310\python.exe

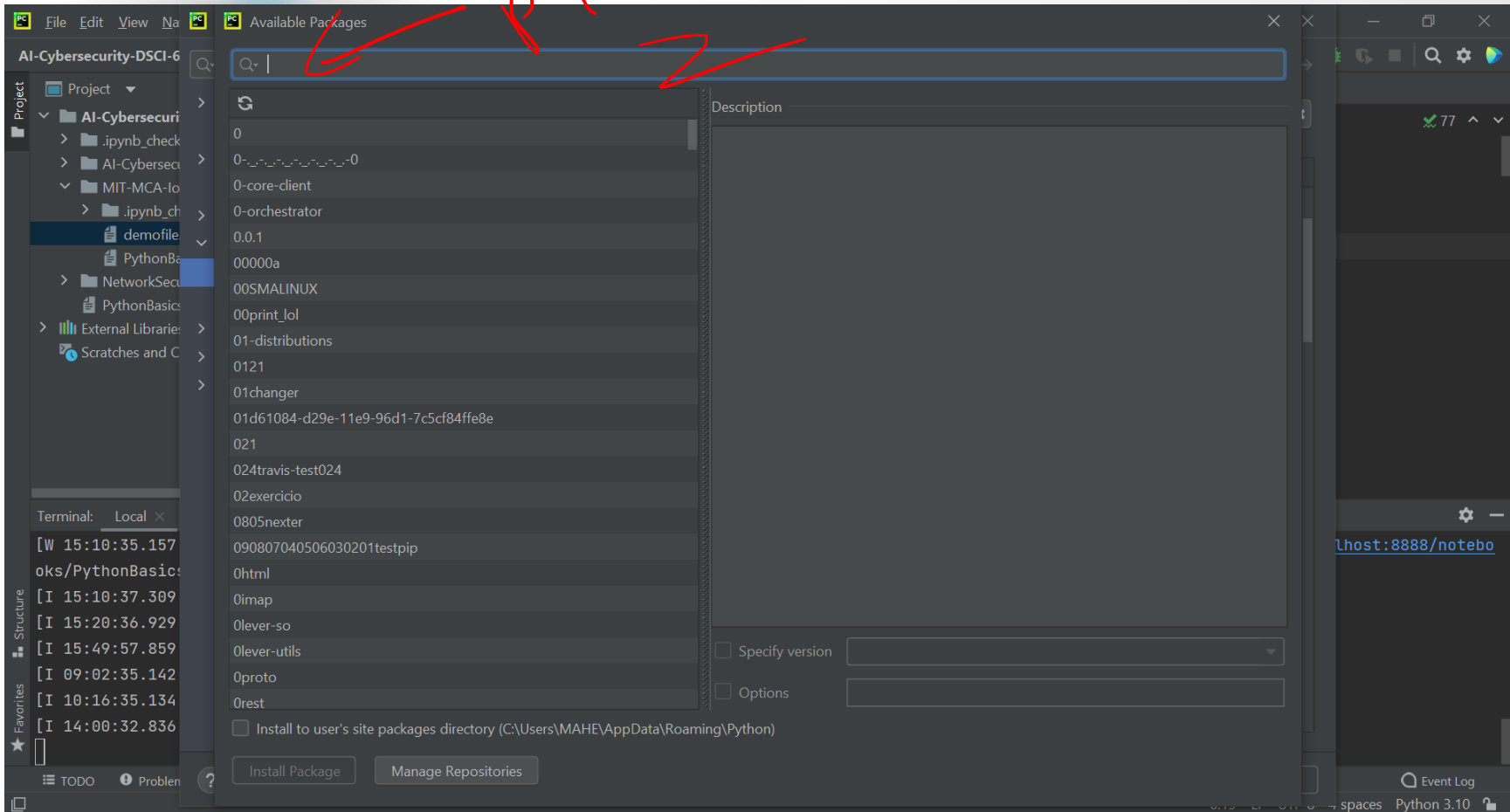
+ - ▲ ▼

Package	Version	Latest version
Jinja2	3.1.1	▲ 3.1.2
MarkupSafe	2.1.1	2.1.1
Pillow	9.1.0	9.1.0
Pygments	2.11.2	▲ 2.12.0
QtPy	2.0.1	2.0.1
Send2Trash	1.8.0	1.8.0
argon2-cffi	21.3.0	21.3.0
argon2-cffi-bindings	21.2.0	21.2.0
asttokens	2.0.5	2.0.5
attrs	21.4.0	21.4.0
backcall	0.2.0	0.2.0
beautifulsoup4	4.10.0	▲ 4.11.1
bleach	4.1.0	▲ 5.0.0
cffi	1.15.0	1.15.0
colorama	0.4.4	0.4.4
cycler	0.11.0	0.11.0
debugpy	1.6.0	1.6.0
decorator	5.1.1	5.1.1
defusedxml	0.7.1	0.7.1
entrypoints	0.4	0.4
executing	0.8.3	0.8.3
fonttools	4.31.2	▲ 4.33.3



# PyCharm Environment

(contd..)





# Installing Jupyter Notebook on PyCharm

- Jupyter Notebook on PyCharm----

<https://jupyter.org/install>

- Install Jupyter-lab →
- Install notebook →
- Run Jupyter Notebook →

`pip install jupyter-lab`

`jupyter install notebook`

`python -m notebook`

```
Terminal: Local x + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\ML and AI in CyberSecurity books codes\AI-Cybersecurity-DSCI-6672-master> cd MIT-MCA-IoT-Class
PS D:\ML and AI in CyberSecurity books codes\AI-Cybersecurity-DSCI-6672-master\MIT-MCA-IoT-Class> python -m notebook
```

```
file:///C:/Users/MAHE/AppData/Local/Packages/PythonSoftwareFoundation.Python.3.10_qbz5n2kfra
pen.html

Or copy and paste one of these URLs:
http://localhost:8888/?token=de2e5fd5d28a8b2b63b384775031cf787ffa08034e3711e0
or http://127.0.0.1:8888/?token=de2e5fd5d28a8b2b63b384775031cf787ffa08034e3711e0
[I 15:08:17.817 NotebookApp] Kernel started: f3199600-efbc-46bf-9c4e-e779743687aa, name: python3
[W 15:08:17.941 NotebookApp] 404 GET /nbextensions/widgets/notebook/js/extension.js?v=20220502150750
aka/PythonSoftwareFoundation.Python.3.10_qbz5n2kfra
```



# Virtual Home Lab

## Virtual Box

← → ↻ virtualbox.org/wiki/Downloads

Apps MCNS Workshop on IoT ... Download Facebook... Yahoo Mail Computer Science... Certificate in Intern... AMC ENGINEERING... Reading list

**VirtualBox**  
Download VirtualBox

search...  
Login Preferences

About  
Screenshots  
Downloads  
Documentation  
End-user docs  
Technical docs  
Contribute  
Community

Here you will find links to VirtualBox binaries and its source code.

**VirtualBox binaries**

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.0 packages, see [VirtualBox 6.0 builds](#). Please also use version 6.0 if you need to run VMs with software virtualization, as this has been discontinued in 6.1. Version 6.0 will remain supported until July 2020.

If you're looking for the latest VirtualBox 5.2 packages, see [VirtualBox 5.2 builds](#). Please also use version 5.2 if you still need support for 32-bit hosts, as this has been discontinued in 6.0. Version 5.2 will remain supported until July 2020.

**VirtualBox 6.1.32 platform packages**

- Windows hosts
- OS X hosts
- Linux distributions
- Solaris hosts
- Solaris 11 IPS hosts

The binaries are released under the terms of the GPL version 2.

See the [changelog](#) for what has changed.

You might want to compare the checksums to verify the integrity of downloaded packages. *The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!*

- [SHA256 checksums](#). [MD5 checksums](#)

<https://www.virtualbox.org/wiki/Downloads>



# Virtual Home Lab

## Raspberry Pi for virtual machine

The screenshot shows a web browser displaying the Robotics Back-End website. The URL in the address bar is [roboticsbackend.com/install-raspbian-desktop-on-a-virtual-machine-virtualbox/](https://roboticsbackend.com/install-raspbian-desktop-on-a-virtual-machine-virtualbox/). The website header includes the Robotics Back-End logo and the tagline "Program Robots Like a Boss". A navigation menu at the top lists categories: Latest, ROS, ROS2, Raspberry Pi, Arduino, Courses, Youtube Tutorials, and Contact. The main content area features a large article titled "Install Raspberry Pi OS Desktop on a Virtual Machine (VirtualBox)". Below the title is a screenshot of the Raspbian Desktop running in an Oracle VM VirtualBox window. The desktop shows a welcome message, a taskbar with icons for the Raspberry Pi logo, a globe, a folder, a terminal, and a Raspberry Pi icon. A tooltip says "Click here to open applications menu". The bottom of the window shows a "Welcome to Raspberry Pi" message. To the right of the main article is a sidebar with a section titled "Want to learn Raspberry Pi from scratch?" and a link to "Raspberry Pi For Beginners" with an accompanying image of a Raspberry Pi board and various components.

<https://roboticsbackend.com/install-raspbian-desktop-on-a-virtual-machine-virtualbox/>

# Installing Jupyter Notebook on PyCharm



localhost:8888/tree

Quit Logout

Files Running Clusters

Select items to perform actions on them.

Upload New

localhost:8888/notebooks/Untitled.ipynb?kernel\_name=python3

MCNS Workshop on IoT -... Download Faceboo... Yahoo Mail Computer Science... Certificate in Intern... Moodle2Word: Wor... Hop-distance relati... PDF to Word Conve...

jupyter Untitled Last Checkpoint: a few seconds ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Help

Trusted Python 3 (ipykernel)

In [ ]: |





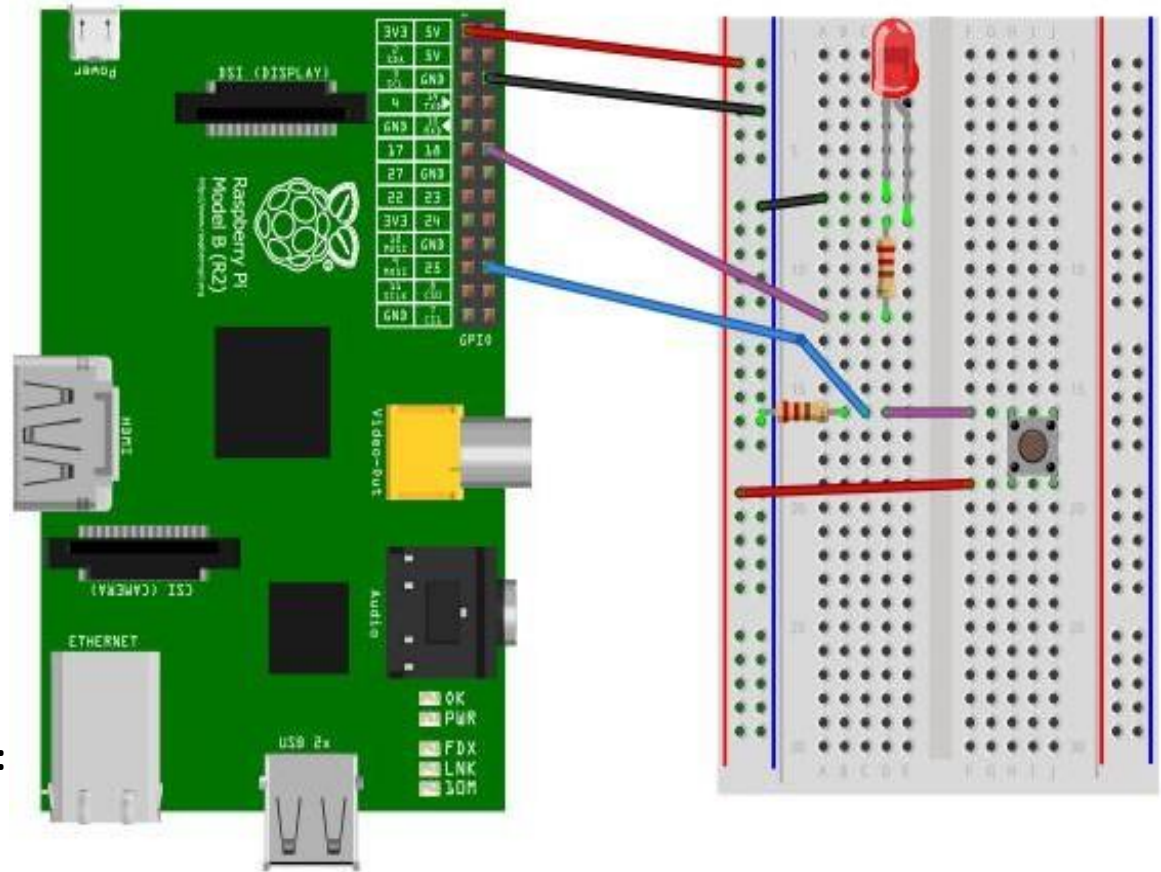
# Raspberry Pi Example: Interfacing LED and switch with Raspberry Pi

```
from time import sleep
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
```

```
#Switch Pin GPIO.setup(25,
GPIO.IN) #LED Pin
GPIO.setup(18, GPIO.OUT)
state=False
```

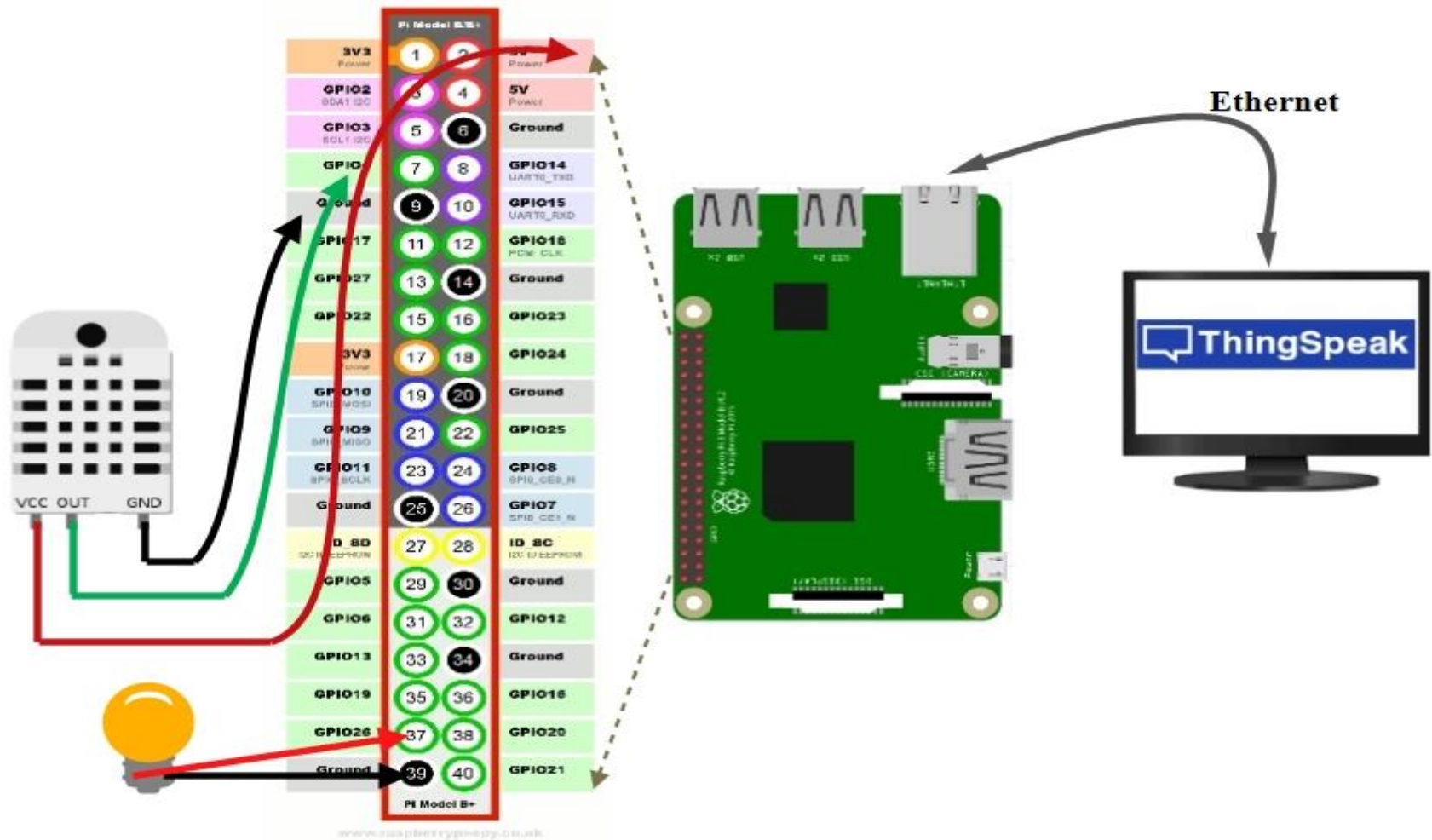
```
def toggleLED(pin):
    state = not state
    GPIO.output(pin, state)
```

```
while True:
    try:
        if (GPIO.input(25) == True):
            toggleLED(pin)
            sleep(.01)
    except KeyboardInterrupt:
        exit()
```

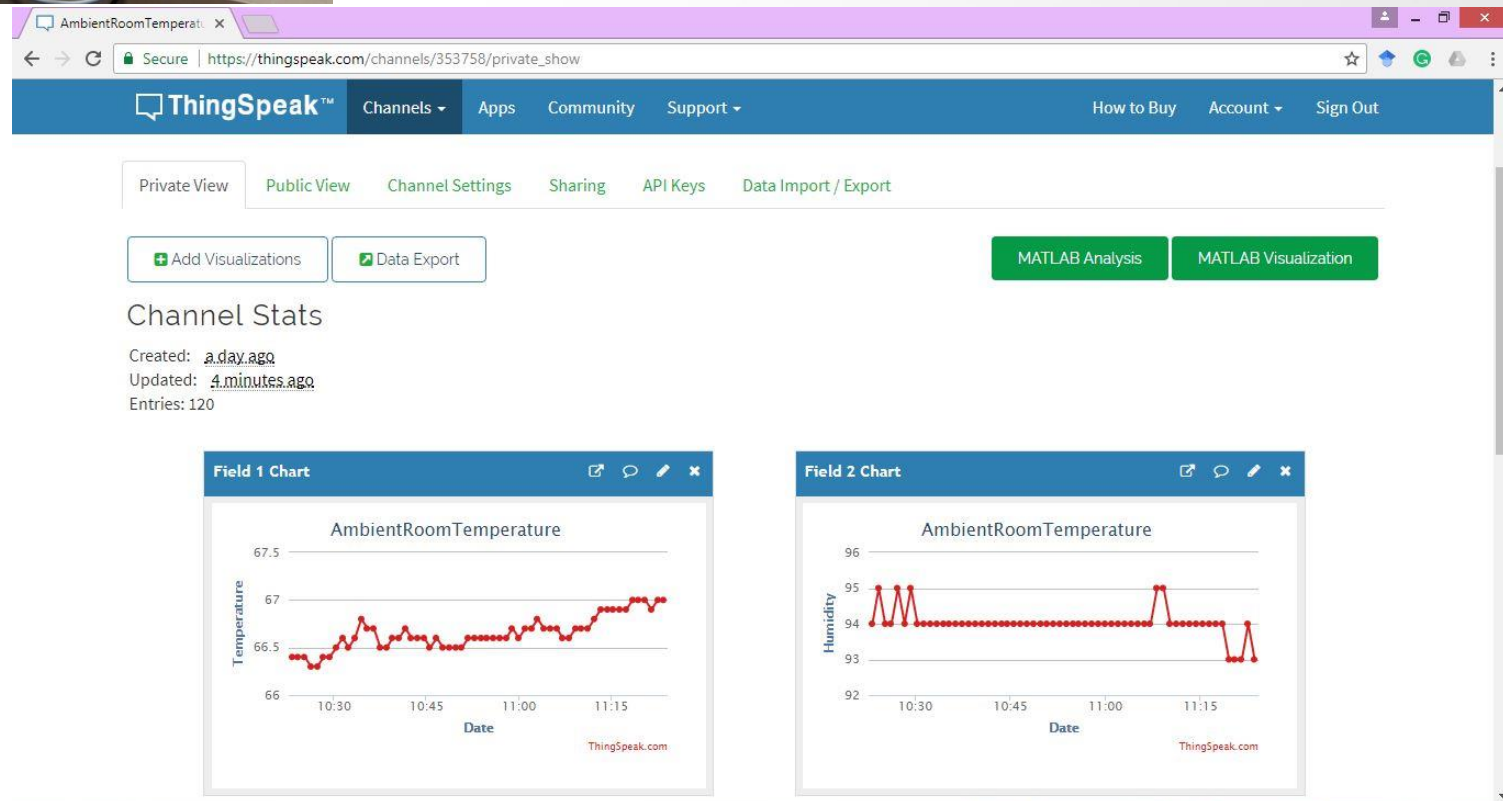




# Temperature Monitoring using RPi



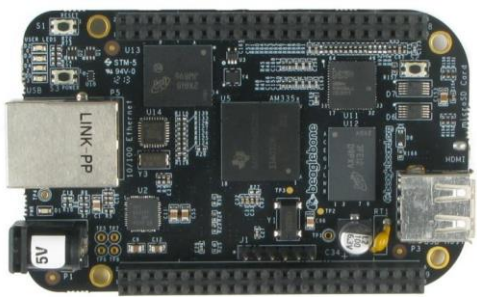
# Temperature Monitoring using RPi







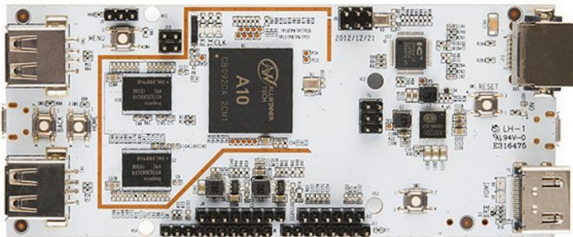
# Other Devices



pcDuino



Cubieboard



BeagleBone Black



Intel Edison



WHAT  
NEXT?