CME 466
Design of an Advanced Digital System

Shahim Vedaei and Khan Wahid

Winter 2023

Introduction to Raspberry Pi

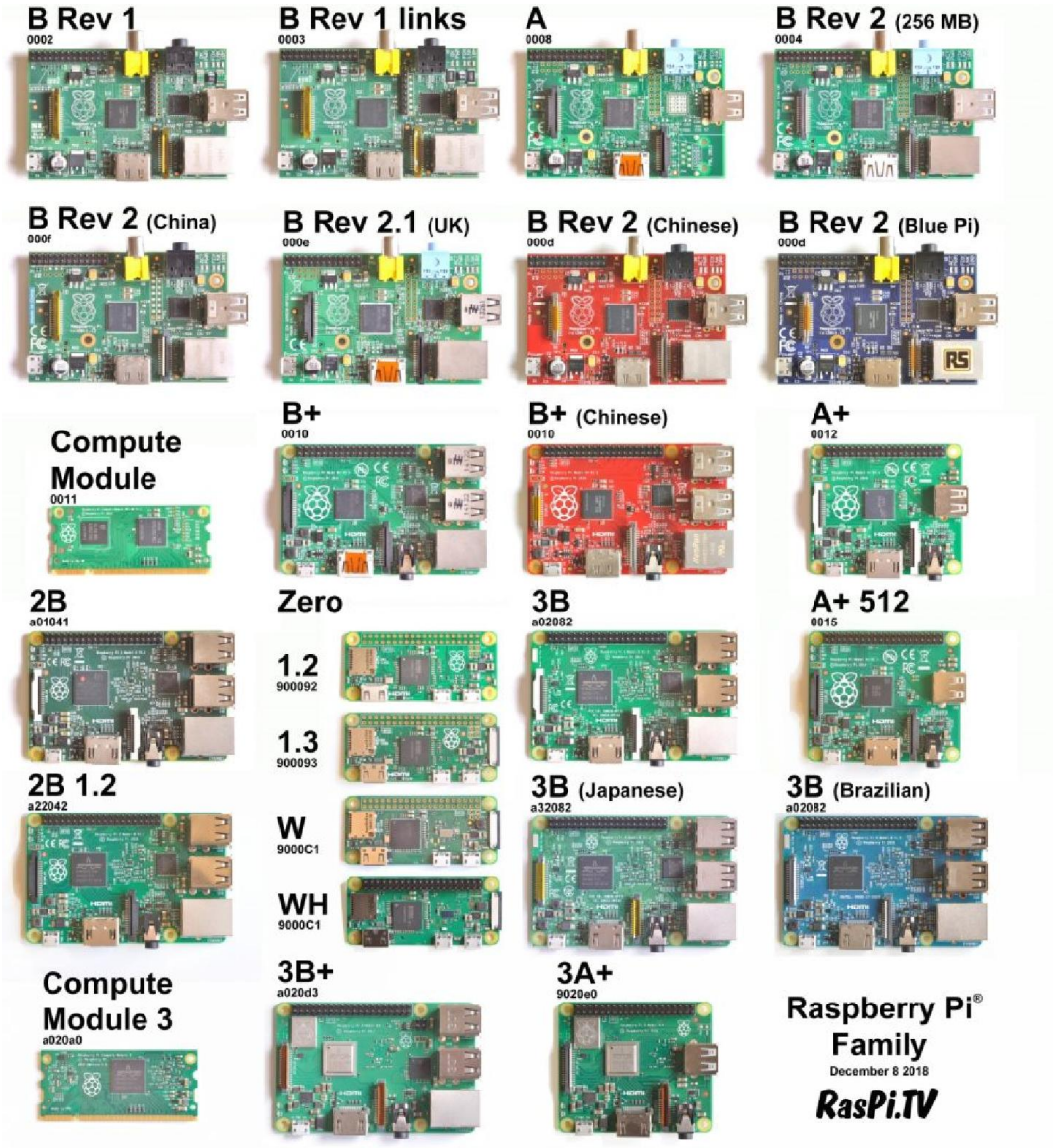
Safety: In this lab, voltages used are less than 15 volts and this is not normally dangerous to humans. However, you should assemble or modify a circuit when power is disconnected and do not touch a live circuit if you have a cut or break in the skin. Preparing for emergencies protects our lives and property. A classroom emergency posting is located in each classroom and lab near the main door of the room. Students are advised to review and be familiar with this classroom emergency posting and the College of Engineering Emergency Response Plan (ERP) The building must be evacuated when an alarm sounds for more than 10 seconds.

Objectives: By the end of this lab, you should have learned the following:

1) Getting know the Raspberry Pi (RPI) board.
2) Hands on python programming and write hello world program.
3) Setup RPI GPIOs.
4) Design a simple program to control the RPI GPIOs.

Code Submission: Always create your projects with your NSID in the project name. Always submit the entire project folder (which should already include your NSID in the folder name) with relevant outputs electronically in a zip file (named <NSID>-<Lab#-Part#>) for each part (if applicable) through Canvas by the deadline. Enhance the readability of your code by adding explanatory comments wherever appropriate. If you are unsure about what is expected, ask your instructor.

1. Raspberry Pi Family



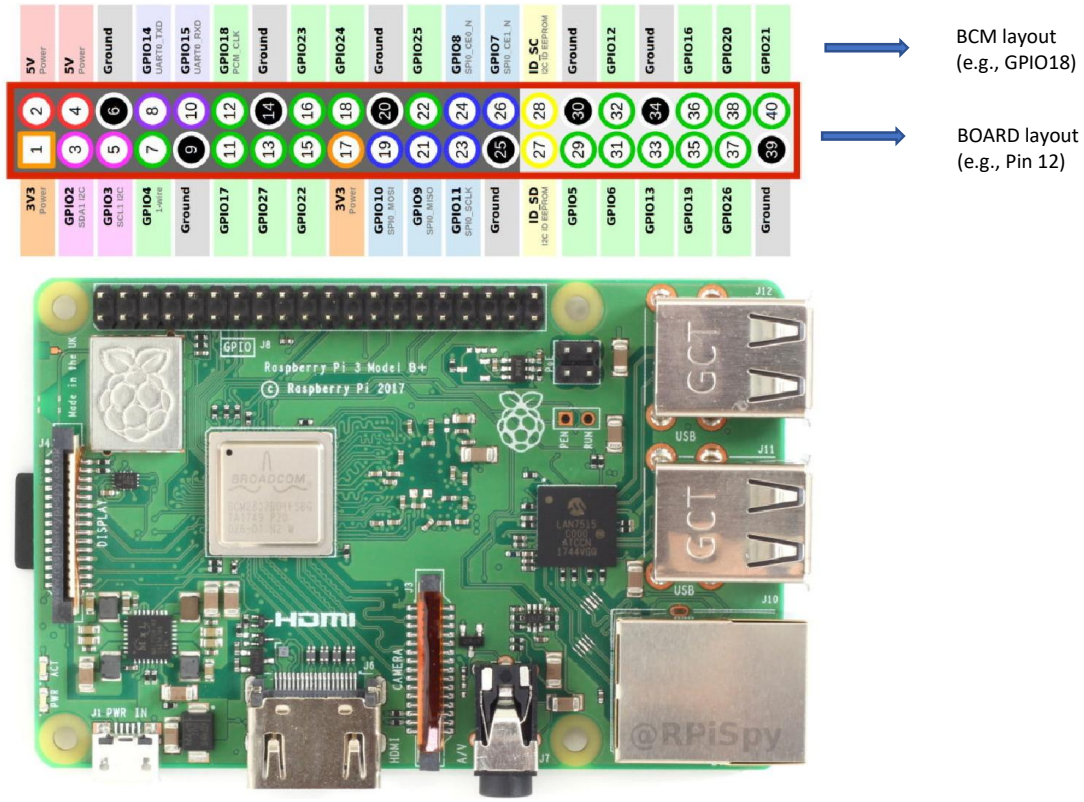
1.1. Raspberry Pi 3 Specifications:

Processor:	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz
Memory:	1GB LPDDR2 SDRAM
Connectivity:	2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE Gigabit Ethernet over USB 2.0 (maximum throughput 300Mbps) 4 × USB 2.0 ports

Access:	Extended 40-pin GPIO header
Video and sound:	1 × full size HDMI MIPI DSI display port MIPI CSI camera port 4 pole stereo output and composite video port
Multimedia:	H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics Micro SD format for loading operating system and data storage
Input power:	5V/2.5A DC via micro-USB connector 5V DC via GPIO header Power over Ethernet (PoE)–enabled (requires separate PoE HAT)
Environment:	Operating temperature, 0–50°C

1.2. GPIO

NOTE: Maximum GPIO input volage is 3.3v

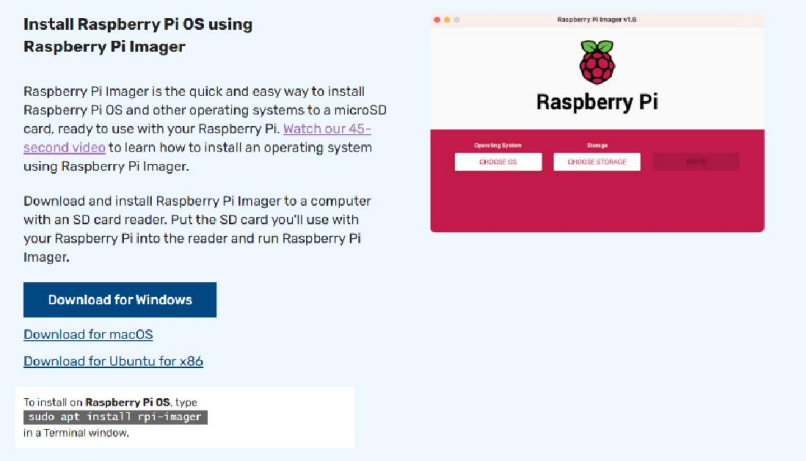


2. How to install the OS

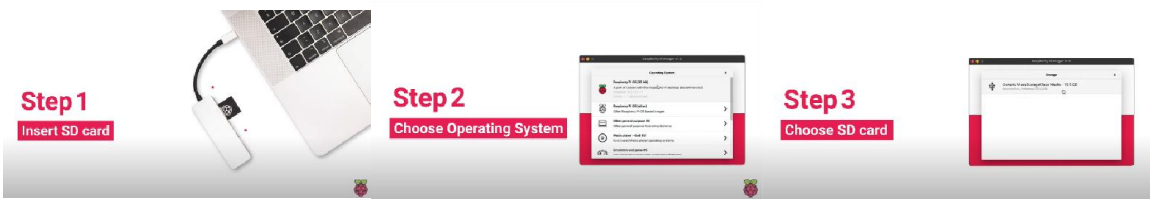
https://www.raspberrypi.com/software/

2.1. Option 1

Download the installer app



Follow the 3 steps:



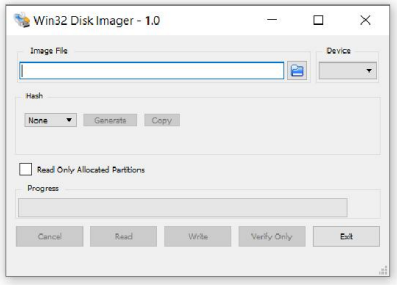
2.2. Option 2

Download the OS image and burn to SD card manually

You need a software to burn the image file to the SD card

On Windows: Win32DiskImages

On Linux: `sudo dd if=/RPI_image of=/dev/sdx`



3. RPI basic configuration

3.1. Basic linux commands

```
cd <directory>    # change directory
cp <file_path_1> <file_path_2>    # copy file
rm <file_path>    # remove file
mkdir <folder_name>    # create folder
nano <text_file>    # terminal-based text editor software
pwd    # Show the current path
uname -a    # print the OS
sudo -i    # super user privilege
sudo apt-get update    # update the package lists
sudo apt-get upgrade    # will fetch new versions of packages existing
sudo apt-get install <my_package>    # download and install desired packages
pip    # is a package-management system written in Python used to install and manage software packages.
pip install <my_python_package>    # install my_python_package
```

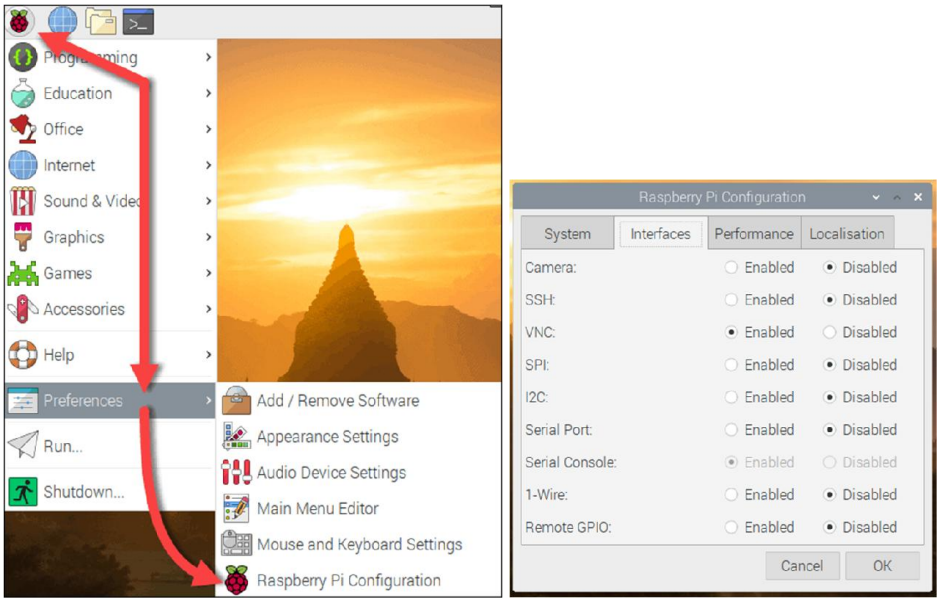
3.2. Check the internet connection and IP address.

3.2.1. Using GUI by clicking on wireless connectivity button on the menu bar.

3.2.2.

```
ifconfig
```

3.3. VNC and SSH

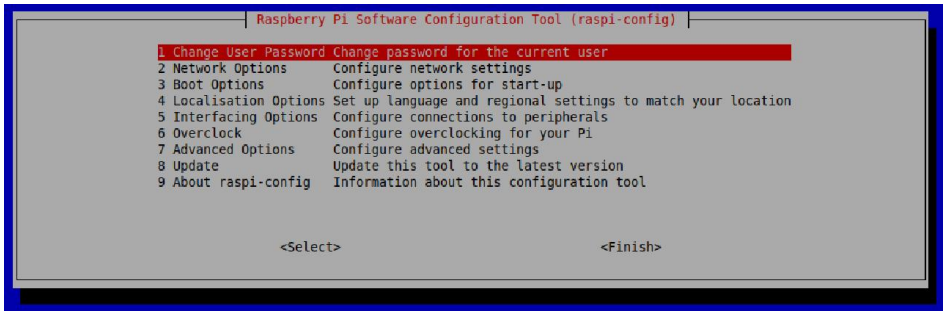


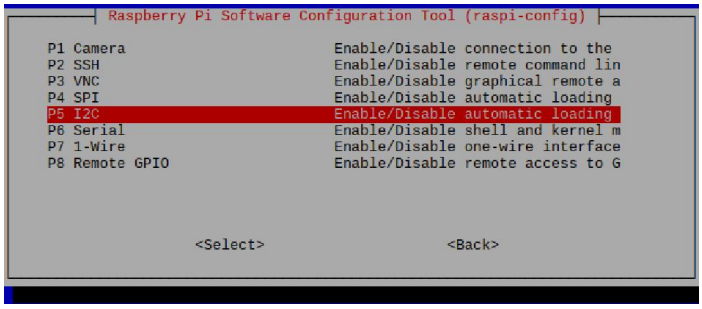
Or using terminal by:

```
sudo raspi-config
```

Note: if you are using TigerVNC from the lab computer to remotely connect to the RPi for the first time, make sure to do the following:

- Connect RPi's HDMI video output to a monitor using a HDMI cable (may need micro HDMI)
- Connect a keyboard to the USB port of the RPi
- Open VNC Server from the upper right corner and go to Options
- Change Encryption to "Prefer off" and Authentication to "VNC password"
- Use a password "raspberry"





4. Hello world program

1. Open a text editor and type the following

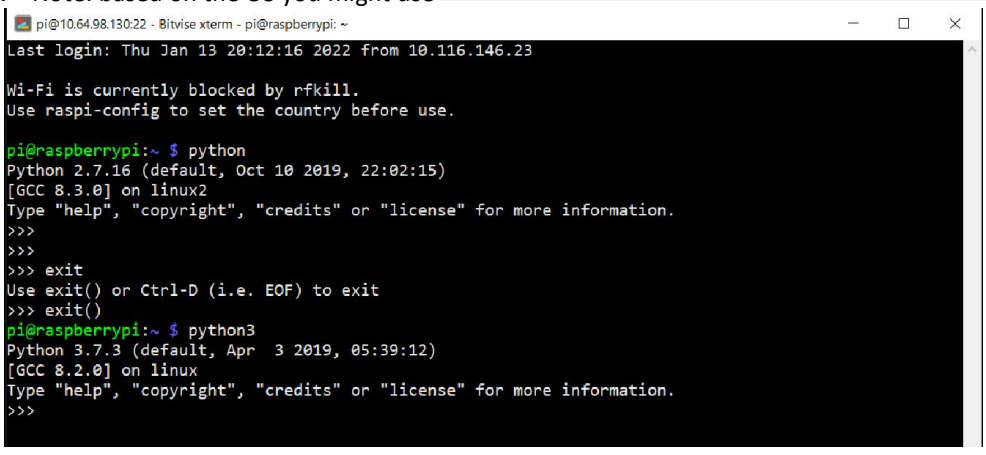
```
print("Hello world")
```

2. Save the file with *.py* format. (ex. Hello.py)
3. Open a terminal and cd to the file path.
4. Run the code by typing in the terminal:

```
python Hello.py
```

   a. Note: based on the OS you might use

```
python3 Hello.py
```



5. Control GPIO

1. First you need RPI GPIO python package
   In a terminal:

```
pip3 install RPi.GPIO
```

2. Write the python program

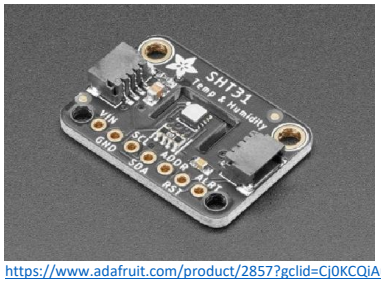
```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)

GPIO.setup(11, GPIO.IN)
GPIO.setup(12, GPIO.OUT)

while True:
        GPIO.output(12, True)
        time.sleep(1)
        GPIO.output(12, False)
        time.sleep(1)
        a = GPIO.input(11)
```

6. GPIO Peripherals

Ex. 1

```
import board
import busio


i2c = busio.I2C(board.SCL, board.SDA)

while not i2c.try_lock():
    pass

i2c.writeto(0x18, bytes([0x05]), stop=False)
result = bytearray(2)
i2c.readfrom_into(0x18, result)

i2c.unlock()
```

Ex. 2



https://www.adafruit.com/product/2857?gclid=Cj0KCQjAuP-OBhDqARIsAD4XHpczB-dEB0szRYOjEp27m8iaYOd7y5LvkaOIqOAiEGZPidkfOmaU3WcaAop7EALw_wcB

```
import board
import busio
import adafruit_sht31d

i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_sht31d.SHT31D(i2c)

sensor.temperature
sensor.relative_humidity
```