IoT Security and Privacy

Building IoT devices with Raspberry Pi

YIER JIN

UNIVERSITY OF FLORIDA

EMAIL: YIER.JIN@ECE.UFL.EDU

SLIDES ARE ADAPTED FROM PROF. XINWEN FU @ UCF/UMASS

Learning Outcomes

Upon completion of this unit:

- Students will be able to make the use of Raspberry Pi, which will be used to build secure IoT device prototypes.
- Students will be able to construct basic circuits interacting with Raspberry Pi
- Students will be able to practice programming Raspberry Pi

Prerequisites and Module Time

Prerequisites

- Students should have taken classes on operating system and computer architecture.
- Students should know basic concepts of networking.

Module time

- Two-hour lecture
- One-hour homework

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

Breadboard

GPIO

Sample use: Control LED

Sample use: Read PIR motion sensor

What is a Raspberry Pi?

University of Cambridge's Computer Laboratory

- Decline in skill level
- Designed for education

A credit card sized PC

Plugs into a TV or monitor

Inexpensive <$40 each

Capability:

- Programming
- Electronic Projects
- Office
- Play HD Videos

Different Versions

Raspberry Pi 4

Raspberry Pi 3 model B+

Raspberry Pi 3 model B

Raspberry Pi 2 model B

Raspberry Pi model B+

Raspberry Pi model A+

Raspberry Pi Zero W

Raspberry Pi Zero

Components

Essential components

- Raspberry Pi 3/4 with WiFi (or USB WiFi dongle for old models)
- Prepared Operating System SD Card
- USB keyboard
- USB mouse
- Display (HDMI, DVI, or composite input (TV cable), DSI touch screen)
- Power Supply

Highly suggested extras

- Internet connectivity - LAN cable
- Case
- Breadboard
- Sensors
- Wires

Programming Languages

Any language which will compile for the ARM chip used in the Pi

Python

C

C++

Java

Scratch

Ruby

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

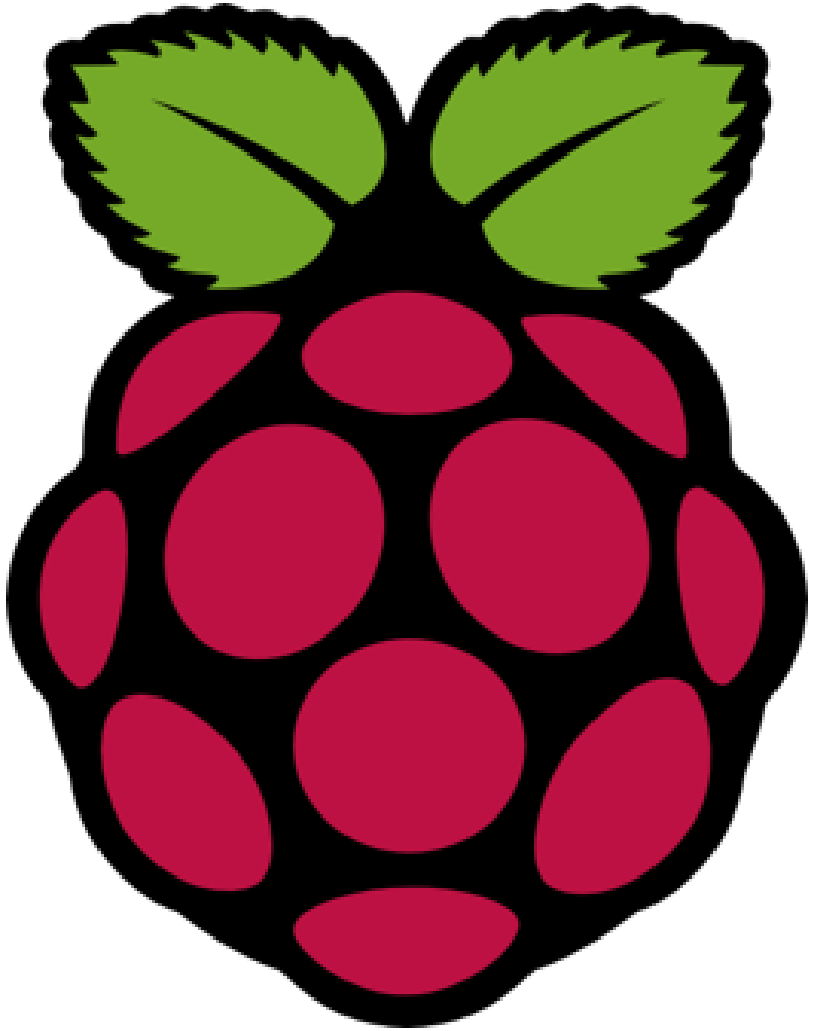
Breadboard

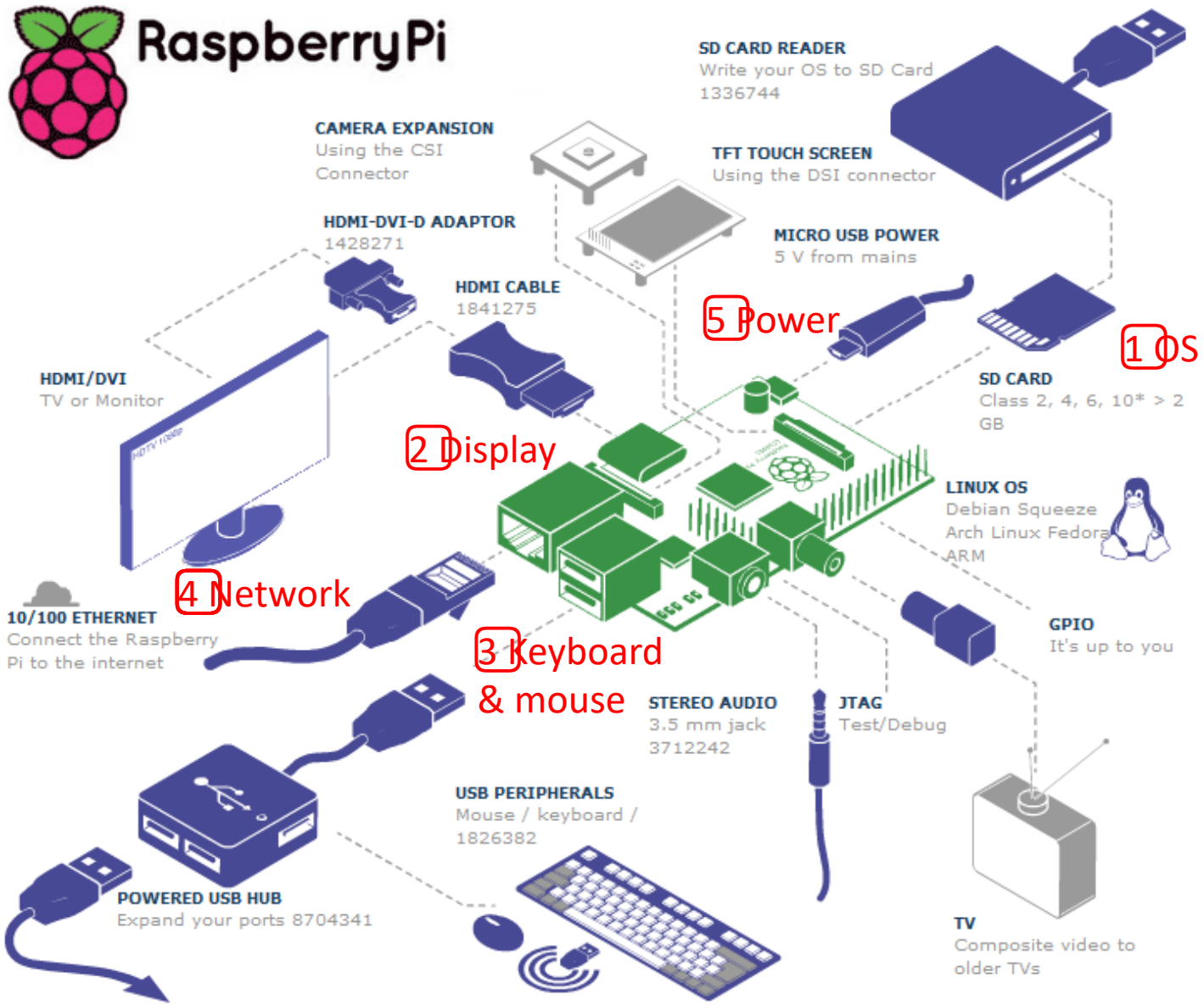
GPIO

Sample use: Control LED

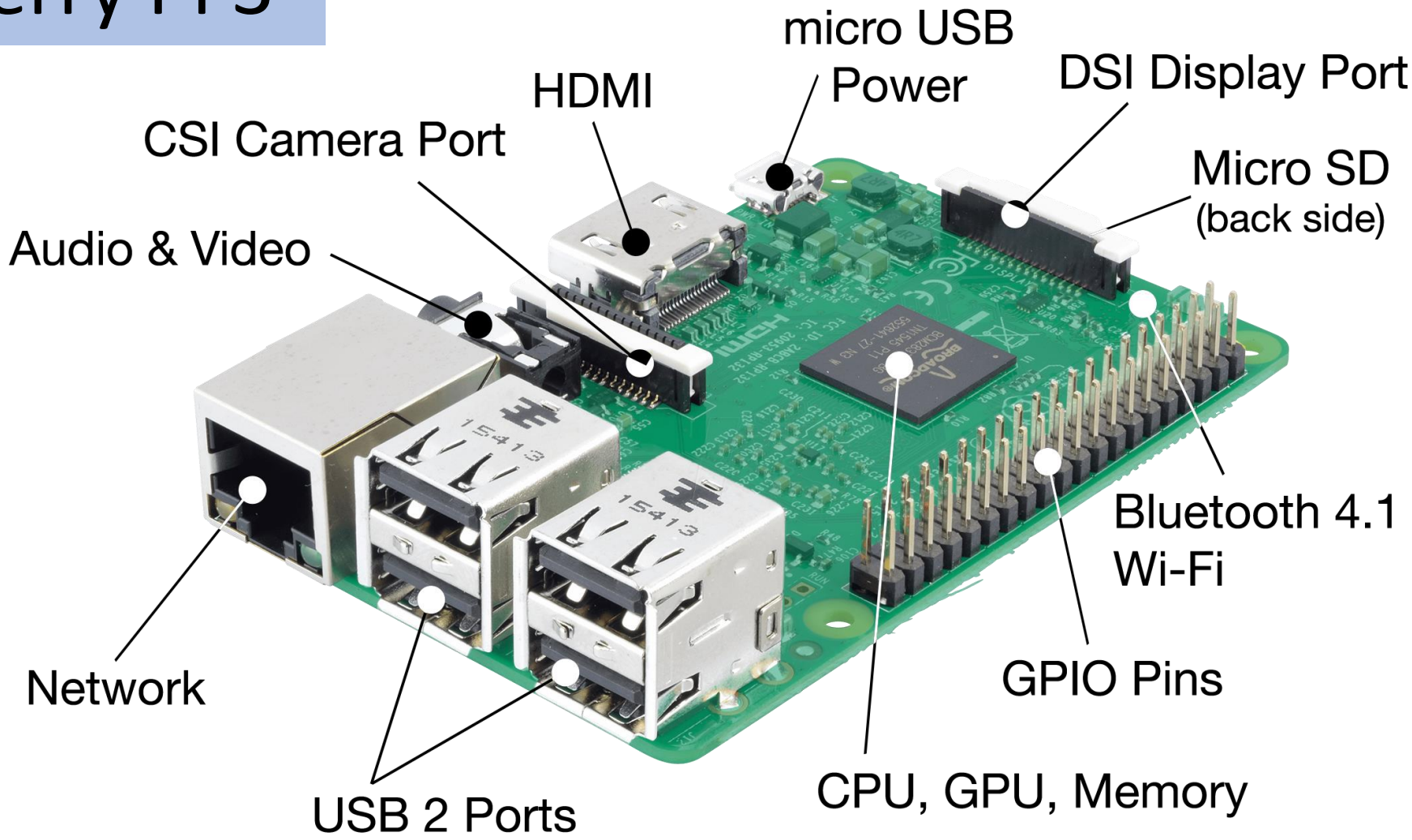
Sample use: Read PIR motion sensor

Setup




RaspberryPi
SD CARD READER
Write your OS to SD Card
1336744
CAMERA EXPANSION
Using the CSI
Connector
TFT TOUCH SCREEN
Using the DSI connector
HDMI-DVI-D ADAPTOR
1428271
MICRO USB POWER
5 V from mains
HDMI CABLE
1841275
5 Power
1 OS
HDMI/DVI
TV or Monitor
SD CARD
Class 2, 4, 6, 10* > 2
GB
2 Display
LINUX OS
Debian Squeeze
Arch Linux Fedora
ARM
4 Network
10/100 ETHERNET
Connect the Raspberry
Pi to the internet
GPIO
It's up to you
3 Keyboard
& mouse
STEREO AUDIO
3.5 mm jack
3712242
JTAG
Test/Debug
USB PERIPHERALS
Mouse / keyboard /
1826382
POWERED USB HUB
Expand your ports 8704341
TV
Composite video to
older TVs

Raspberry Pi 3



Configurations

Booting Up for the First Time

- Choose OS (Raspbian – recommended)
- Configure to use the Whole SD Card
- Change Timezone
- Boot into Desktop

Run the configuration later

- sudo raspi-config
- Or use the GUI

Network setup

- Using a Wired Network
- USB WiFi Adapter/Onboard WiFi

Software Packages to Use I/O (Input/Output)

GPIO setup: install Python library Rpi.GPIO

```
sudo apt-get update
sudo apt-get install python-dev
sudo apt-get install python-rpi.gpio
```

I2C: A standard for chips talking to each other

- Multiple devices to Raspberry Pi through I2C bus
- Unique address for each device through jumper settings
- To use with Pi: Enable kernel support from rasp-config

Install the i2c-tools utility

```
sudo apt-get update
sudo apt-get install python-smbus
sudo apt-get install i2c-tools
```

- Test I2C devices: sudo i2cdetect -y 1

Sample Starter Kits

CanaKit Raspberry Pi 4 4GB Starter Kit with WiFi $99.99

CanaKit Raspberry Pi 2 Ultimate Starter Kit with WiFi $84.99

Raspberry Pi 2 Model B Starter Pack - Includes a Raspberry Pi 2 $99.95

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

Breadboard

GPIO

Sample use: Control LED

Sample use: Read PIR motion sensor

Example Use

Office: LibreOffice

Programming: Python, Scratch

Game console

Web server: Apache

Tor router

HTPC - A home theater PC (HTPC) or media center computer

Bird house

Super computer

Clock

PiBot

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

Breadboard

GPIO

Sample use: Control LED

Sample use: Read PIR motion sensor

How A Breadboard Works

A grid of holes on a plastic board

Internal metal strips underlining the holes as jumper wires

- Connecting specific rows or columns of holes.

Electric component leads and wires are inserted into the holes

- Wires connect electric component leads

Building circuits with a breadboard: stick components and wires into the holes.

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

Breadboard

GPIO (General-purpose input/output)

Sample use: Control LED

Sample use: Read PIR motion sensor

GPIO: MODELS A+, B+ AND RASPBERRY PI 2, 3, 4

Physical interface between the Pi and the outside world.

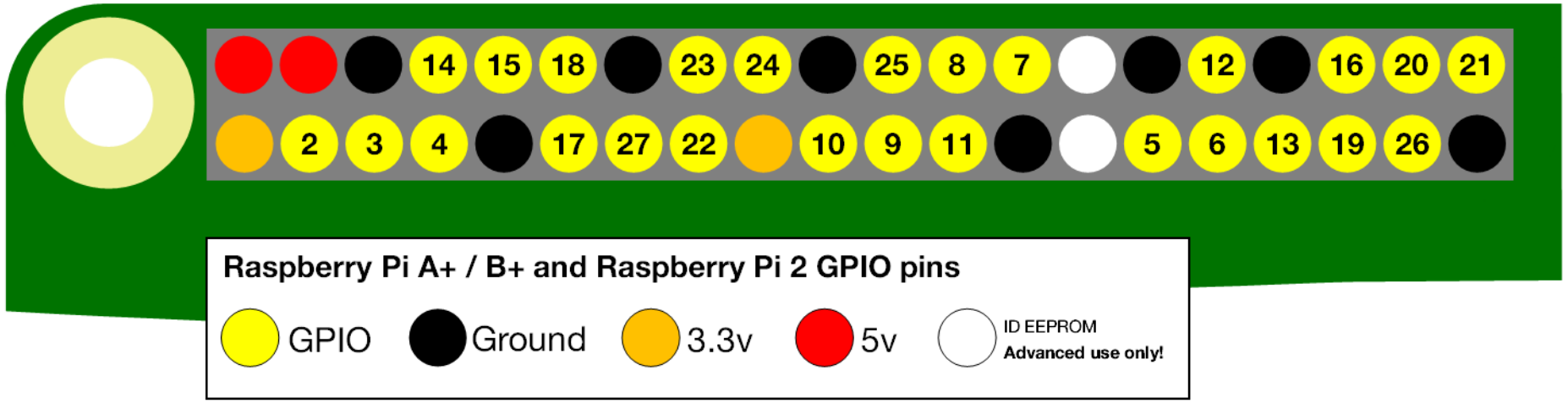
Can be viewed as switches

- That you can turn on or off (input)
- Or that the Pi can turn on or off (output).

GPIO Pins

40 pins on the board

- 26 GPIO pins
- Others are power or ground pins



Use of GPIO Pins

Program the pins to interact in various ways

Input to a pin from a sensor, another computer or device

- For example, from a motion sensor

Output from a pin can do

- Turn on an LED
- Send a signal or data to another device.

A networked Raspberry Pi

- Remote control of attached physical devices
- Receiving data from those devices

Pin Numbering

GPIO NUMBERING

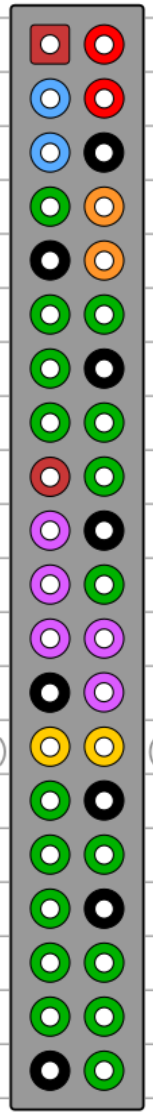
- Seen by the onboard computer

PHYSICAL NUMBERING

- Counting across and down from pin 1 at the top left

Raspberry Pi 3 GPIO Header

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I²C)		DC Power 5v	04
05	GPIO03 (SCL1 , I²C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)		(I²C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40



Rev. 2
29/02/2016

www.element14.com/RaspberryPi

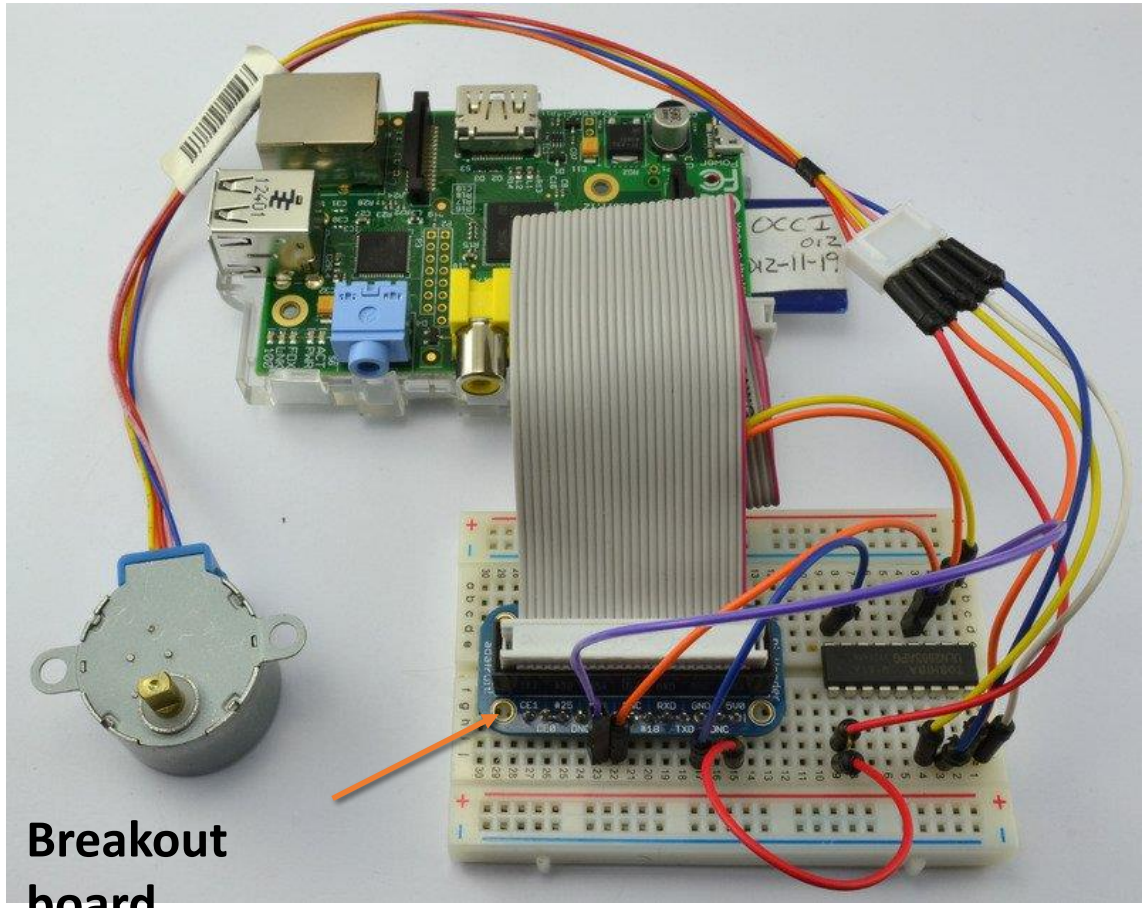
Warning

Unknowingly plugging wires and power sources may damage Pi when connecting power hungry things to a Pi

- LEDs are fine
- Motors are not

For caution and ease of use, use a breakout board like Pibrella Until you become familiar with Pi

- Labeled pins (otherwise needs a diagram)
- Maybe protection circuit



Breakout board

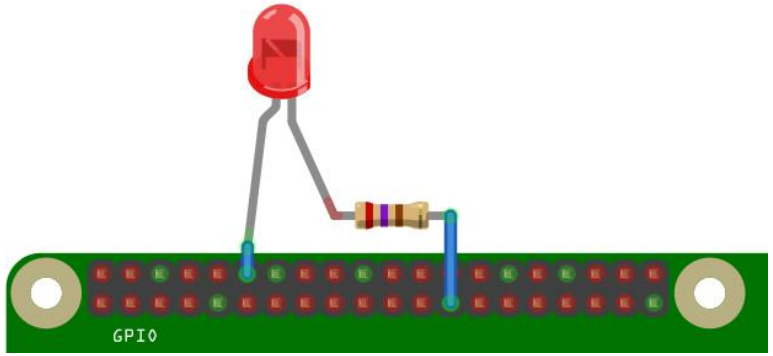
Output

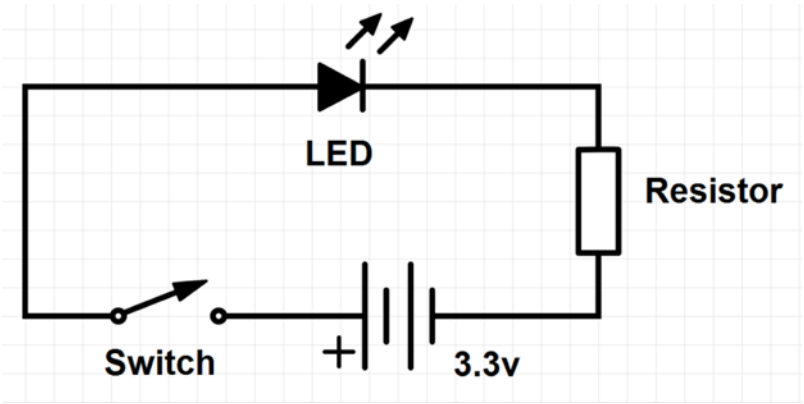
GPIO pin **outputs** are 3v3 or 0v

- Can be called **on** or **off**, **HIGH** or **LOW**
- Each pin can turn on or off

Example use

- Raspberry Pi as **the switch and the battery** in the left diagram.





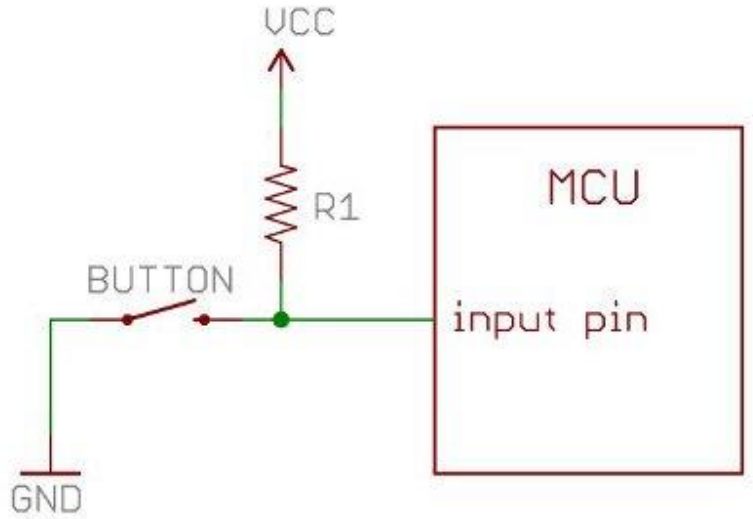
INPUT

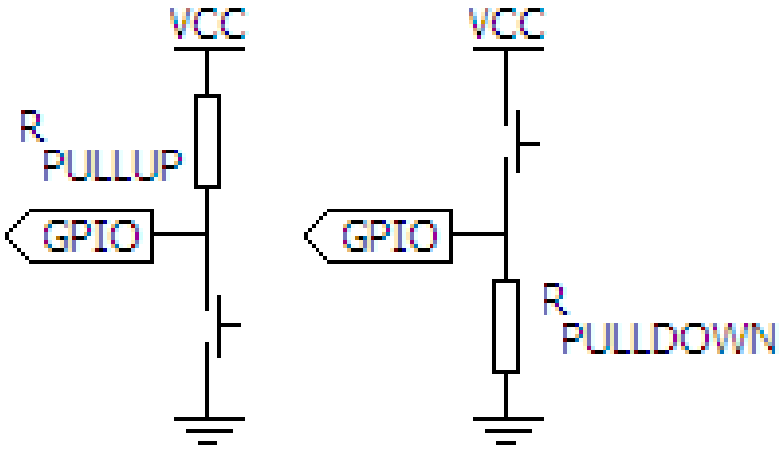
GPIO pins are the simplest I/O on microcontrollers

"pull up" and "pull down" circuits give an input pin a reference

- In a case that the switch is off
- To differentiate noise from signal

The input mode of a GPIO pin has high impedance





Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

Breadboard

GPIO

Sample use: Control LED

Sample use: Read PIR motion sensor

Program with Python to Control LED

Open **Terminal** application

- Click the terminal icon on the taskbar or
- **Main Menu** -> **Accessories** -> **Terminal**.

Open editor ***idle3***: sudo idle3 & and press

Create a Python file within idle3

- **File** -> **new window**.
- File -> Save As **reaction.py**
- Type the code
- Save the file: File -> Save.

Run the code: **Run** -> **Run Module** or by pressing F5

Run Python File from the Command Line

Run Python 2

- python reaction.py

Run python3

- Python3 reaction.py

Python [7]

Hello world

- print("Hello world")

Indentation

- Indicate a block of code
- 4 blank spaces in tradition

Variables

- Assignment like c with no need of type definition
- Type can be changed

Comments

- Single line #
- Multi-line: triple quotes “”” comments “””

Python [7] (Cont'd)

Lists (like array)

- Mylist=[7, 8, 9]

Iteration

- for

```
N= [1, 2, 3]
for n in N:
    print(n)
```

Range

```
for i in range(5):
    print(i)
```

if statement

```
name = "Joe"
if len(name) > 3:
    print("Nice name,")
    print(name)
else:
    print("That's a short name,")
    print(name)
```

Code

```
1. import RPi.GPIO as GPIO  # Import GPIO library
2. import time # Import time library
3. GPIO.setmode(GPIO.BCM) #Use BCM pin numbering
4. GPIO.setwarnings(False) #Ignore GPIO warning messages
5. led = 4 # Assign 4 to variable led
6. GPIO.setup(led, GPIO.OUT) # Set pin 4 for outputting information
7. GPIO.output(led, 1) # Turns the GPIO pin ‘on’ ( i.e., outputs 3.3v)
8. time.sleep(5) # Pause the prorgam for 5 second
9. GPIO.output(led, 0) # Turns the GPIO pin ‘off’ ( i.e., outputs 0v)
10. # Clean up all the used ports in the program. Resets any ports you have used in this program back to input mode
11. GPIO.cleanup()
```

Outline

Overview

Setup and configurations

10 Use of Raspberry Pi

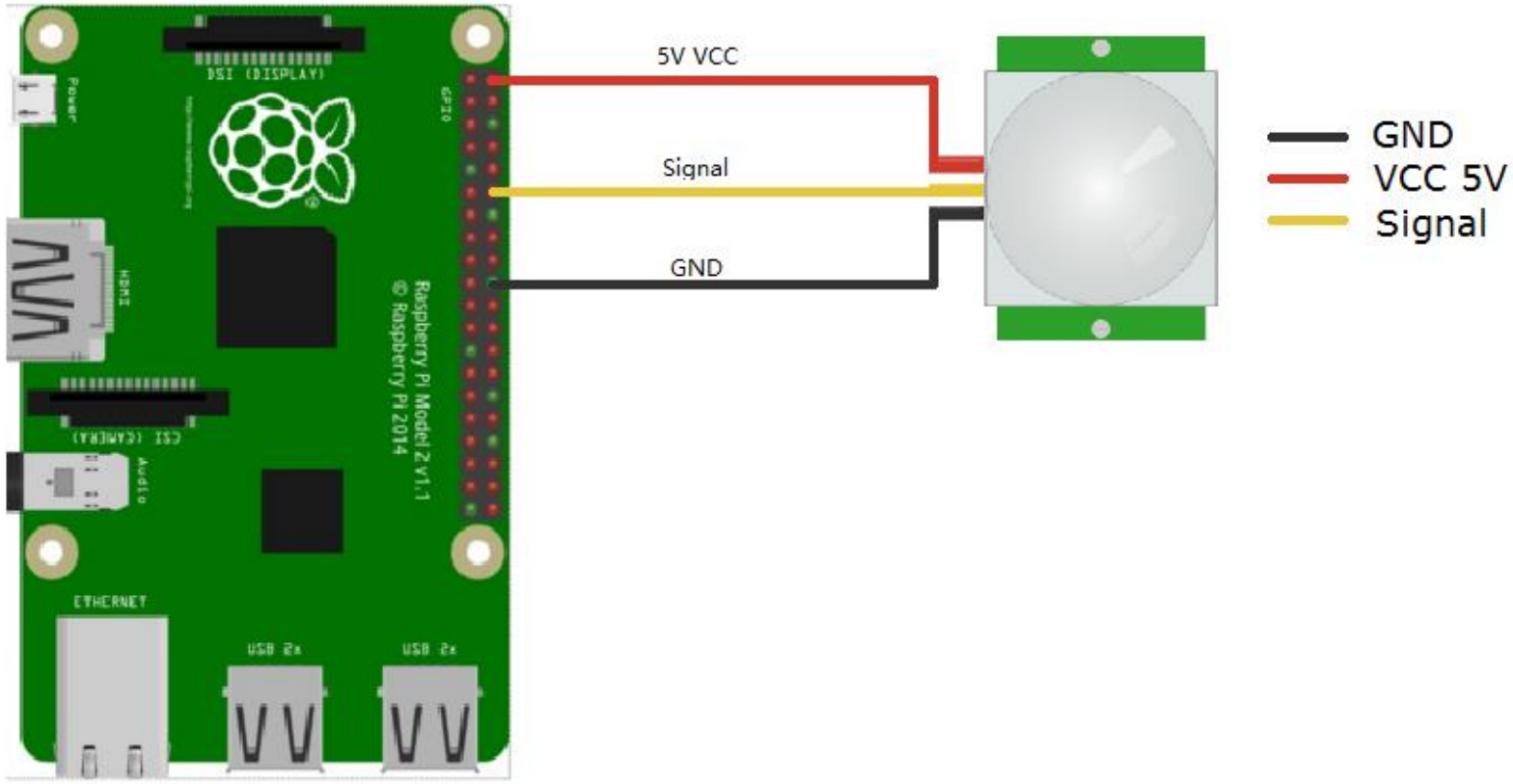
Sample starter kit

GPIO

Sample use: Control LED

Sample use: Read PIR motion sensor

Motion Sensor



PIR Sensor Principle

The PIR sensor is made of IR sensitive material

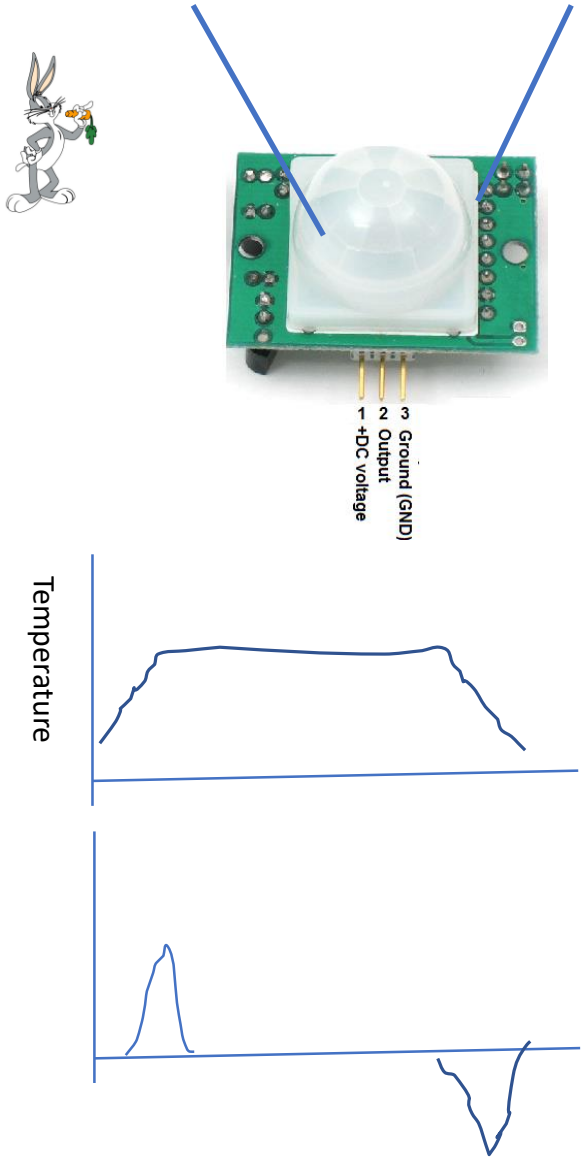
The lens over the sensor can change

- Breadth, range, sensing pattern

An idle sensor detect the same amount of IR from the environment.

Motion detection

- A warm entity entering the sensing zone causes a positive differential change
- The warm entity leaving the sensing zone generates a negative differential change



PIR Sensor Code Example

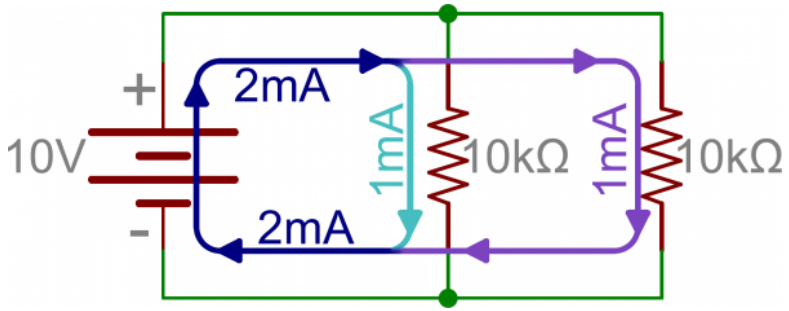
```
1. import RPi.GPIO as GPIO
2. import time


3. GPIO.setmode(GPIO.BCM)
4. pir= 7
5. GPIO.setup(pir, GPIO.IN)


6. try:
7.     print “PIR Module Test (CTRL+C to exit)”
8.     time.sleep(10)
9.     print “Ready”
10. while True:
11.     # read status of pin pir
12.         if GPIO.input(pir): #
13.             print “Motion Detected!”
14.         time.sleep(1)


15. except KeyboardInterrupt:
16.     print “ Quit”


17. GPIO.cleanup()
```

Circuits

Equivalent Resistances in Series Circuits

Equivalent Resistances in Parallel Circuits

$$R_{tot} = R_1 + R_2 + ... + R_{N-1} + R_N$$



$$\frac{1}{R_{tot}} = \frac{1}{R_1} + \frac{1}{R_2} + ... + \frac{1}{R_{N-1}} + \frac{1}{R_N}$$

References

[1] CTAYLOR, Voltage, Current, Resistance, and Ohm's Law, 2016

[2] Smon Monk, LEARN RASPBERRY PI 13 GUIDES, 2015

[3] Mtaylor, Raspberry gPIo, 2016

[4] How PIRs work, 2015

[5] PETE-O, Series and Parallel Circuits, 2016

[6] RPi.GPIO basics 6 – Using inputs and outputs together with RPi.GPIO – pull-ups and pull-downs, Jul 172013

[7] Python, 2017