Sensing and Actuation Networks and Systems 2022/2023

PL Class 03 – Creating your first IoT program

Vasco Pereira (vasco@dei.uc.pt) & Karima Velasquez (kcastro@dei.uc.pt)
Dep. Eng. Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Updated on
27 February 2023

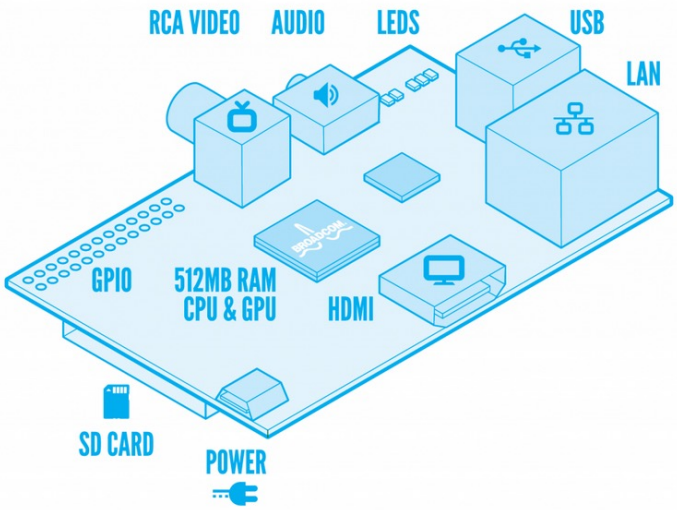
Raspberry Pi introduction

Sensing and Actuation Networks and Systems

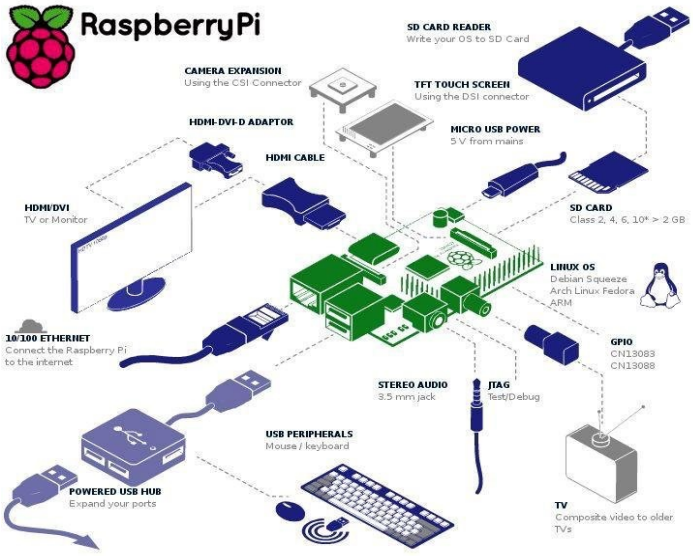
Raspberry Pi

- Tiny, single-board, affordable computer;
- Teaching basic computer science;
- Currently is used for many purposes including weather monitoring and home and industrial automation.

Raspberry Pi



Raspberry Pi

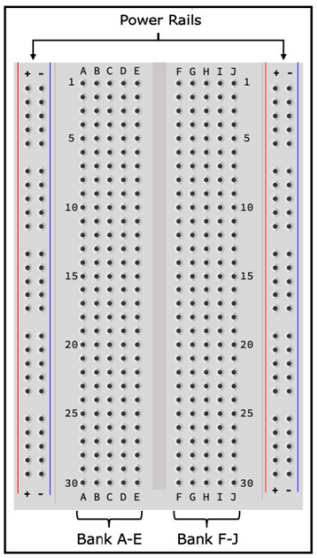


Breadboard

Sensing and Actuation Networks and Systems

Breadboard

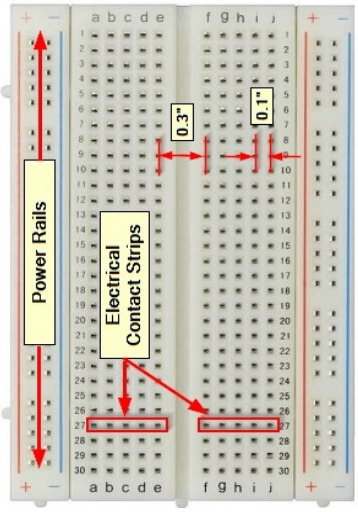
- An electronic breadboard (or protoboard) is a prototyping board that helps to electrically connect components and wires quickly and easily.



Source: Smart, G. (2020). Practical Python Programming for IoT: Build advanced IoT projects using a Raspberry Pi 4, MQTT, RESTful APIs, WebSockets, and Python 3. Packt Publishing.

Breadboard

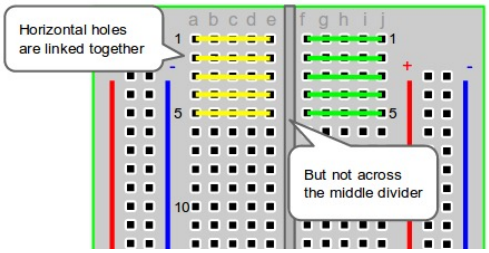
- The holes in the breadboard are where electrical components and wires are placed to electrically connect them. The holes are electrically connected;
- The two outer columns of holes are commonly referred to as *power rails*. There is a positive (+) column and a negative (-) column on either side of the breadboard;
- They do not provide power themselves. They need a power source such as a power supply or battery connected to them to provide power.



Source: Smart, G. (2020). Practical Python Programming for IoT: Build advanced IoT projects using a Raspberry Pi 4, MQTT, RESTful APIs, WebSockets, and Python 3. Packt Publishing.

Breadboard

- The centre of the breadboard has two banks of holes, which have been labelled Bank A-E and Bank F-J. Each row of holes in a bank is electrically connected. For example, holes A1 through to E1 are electrically connected, as are holes F1 through to J1. However, A1-E1 are not electrically connected to F1-J1 because they are on a separate bank.



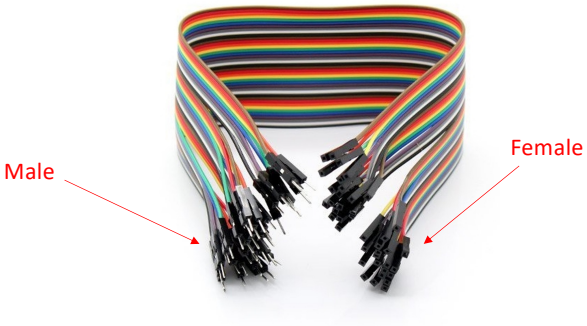
Source: https://computers.tutsplus.com/tutorials/how-to-use-a-breadboard-and-build-a-led-circuit--mac-54746

Jumper Wires

Sensing and Actuation Networks and Systems

Jumper Wires

- Jumper wires are used to make it easier to manage circuits built on a breadboard;
- The wires are usually cut to lengths such that they fit neatly between two holes in the breadboard, but it is easy enough to create your own using a length of wire and some wire strippers;

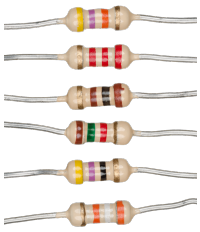


Resistors

Sensing and Actuation Networks and Systems

Resistors

- A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element;
- Used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses;
- Resistors are common elements of electrical networks and electronic circuits and are ubiquitous in electronic equipment.

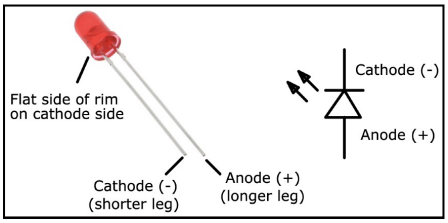


LED Light

Sensing and Actuation Networks and Systems

Led Light

- A LED (*Light-Emitting Diode*) is a small, yet bright, light made of a tiny crystal that emits a colour when electricity is connected to it;
- The left-hand side of the diagram shows a physical representation of a LED, while the right-hand side shows the schematic symbol for a LED.



Source: Smart, G. (2020). Practical Python Programming for IoT: Build advanced IoT projects using a Raspberry Pi 4, MQTT, RESTful APIs, WebSockets, and Python 3. Packt Publishing.

Led Light

- LEDs need to be connected the correct way around into a circuit, otherwise, they will not work;
- When looking closely at a LED, it is possible to notice a flat side on the LED casing;
- The leg on this side is the **cathode**, which connects to the **negative** or ground side of a power source;
- The cathode leg will also be the shorter of the LED's legs;
- The other leg is known as the **anode** and connects to the **positive** side of a power source.

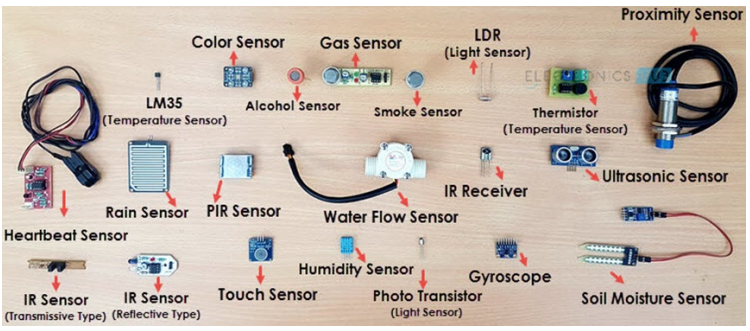
Source: Smart, G. (2020). Practical Python Programming for IoT: Build advanced IoT projects using a Raspberry Pi 4, MQTT, RESTful APIs, WebSockets, and Python 3. Packt Publishing.

Sensors

Sensing and Actuation Networks and Systems

Sensors

- A sensor is a device that converts signals from one energy domain to an electrical domain;
- Sensors are usually a part of a bigger system, and can be analog or digital, according to the output they produce.



Source: https://www.electronicshub.org/different-types-sensors/#What_is_a_Sensor

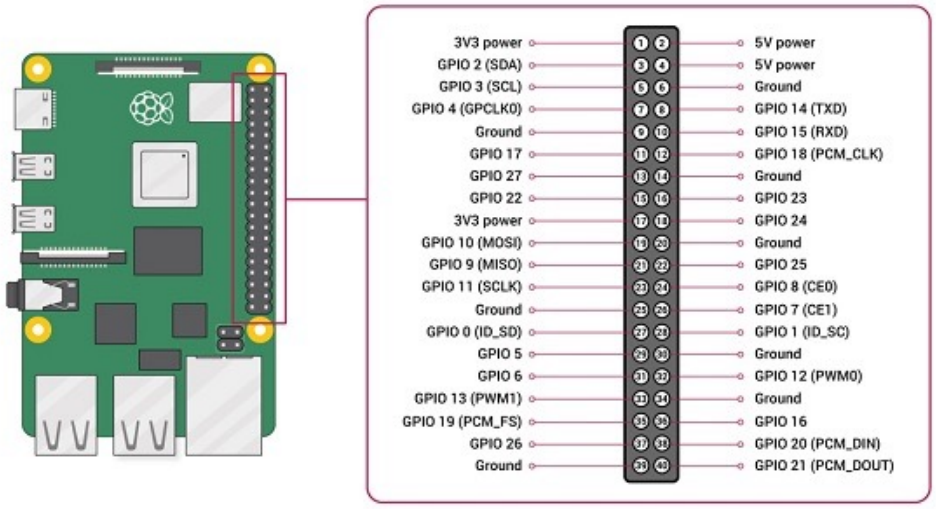
GPIO interface

Sensing and Actuation Networks and Systems

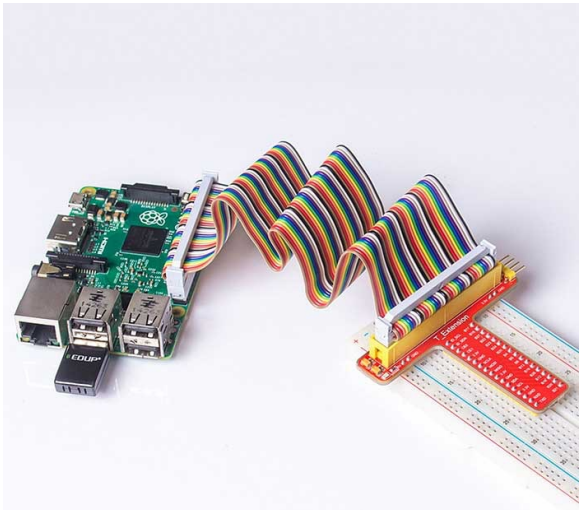
GPIO interface

- *General-Purpose Input/Output*;
- Standard interface used to connect microcontrollers to other electronic devices;
- It can be used with sensors, diodes, displays, and System-on-Chip modules;
- The GPIO can be used in three modes:
  - Input: default, receive input from connected device (e.g., button);
  - Output: deliver data to connected device (e.g., led lamp);
  - UART interface: *Universal Asynchronous Receiver-Transmitter,* enables definition of custom advertising packets.

GPIO interface



GPIO interface



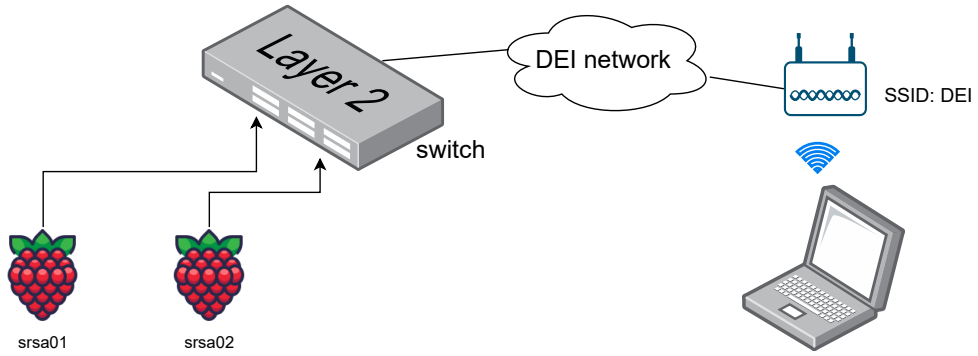
GPIO extension board

It is now time to get your Raspberry!

Class setup

Sensing and Actuation Networks and Systems

Class setup



Raspberry Pi Access Info

Sensing and Actuation Networks and Systems

Raspberry Pi Access Info

- Access credentials:
  - User: `user`
  - Password: `password`

- IP addresses:

Raspberry Pi	IP address
srsa-pi-1	10.6.1.2
srsa-pi-2	10.6.1.3
srsa-pi-3	10.6.1.4
srsa-pi-6	10.6.1.7
srsa-pi-8	10.6.1.9
srsa-pi-9	10.6.1.10

Accessing Raspberry Pi

Sensing and Actuation Networks and Systems

Accessing Raspberry Pi

- Connect to Wi-Fi with SSID = DEI
  - The username/password are the same of your student.dei.uc.pt account
- Check connectivity to your Raspberry Pi:

```
ping <ip_address>
```

Accessing Raspberry Pi

- Connect to your Raspberry Pi using *ssh*:

```
ssh user@<ip_address>
```

Students with Windows systems can use Putty

- After entering, check your directory using command *pwd*:

```
pwd
```

  - The result should be like: `/home/user`

Accessing Raspberry Pi

- From your base directory, create the directory where you will store your code:

```
$mkdir <your_directory>
```

  - **Note:** Use a distinctive name in order to differentiate your directory from other students using the same device.

- Change to the new directory created:

```
$cd <your_directory>
```


Python Virtual Environments

Sensing and Actuation Networks and Systems

Python Virtual Environments

- The Python virtual environment is such that the Python interpreter, libraries and scripts installed into it are isolated from those installed in other virtual environments;
  - This enable sharing the same Raspberry between groups
- It contains a directory tree with Python executable files and other files, which indicate that it is a virtual environment;
- All Python related activity is sandboxed to your virtual environment.

Python Virtual Environments

- Execute the following command to create your environment:

```
$python -m venv <your_venv>
```

  - This command creates a new Python virtual environment using the `venv` tool.
  - The `-m venv` tells Python that we are going to run the `venv` module
  - The `<your_venv>` parameter is the name of the folder where the virtual environment will be created.

Python Virtual Environments

- Activate the virtual environment from your new directory:

```
$ source <your_venv>/bin/activate
(venv) $
```

  - When the terminal has a Python virtual environment activated, all Python related activity is sandboxed to it.
  - If the Raspberry Pi is disconnected from power, after the Python virtual environment is created, it is not erased. However, it will have to be activated again to be used.

GPIO Zero library

Sensing and Actuation Networks and Systems

GPIO Zero library

- An entry-level and easy to use GPIO library for controlling simple electronics;
- Component interfaces are provided to allow a frictionless way to get started with physical computing;
- With very little code, you can quickly connect your components together;
- GPIO Zero builds on a number of underlying pin libraries, including RPi.GPIO and pigpio, each with their own benefits.

GPIO Zero library

- Use the pip command (Python Install Packages) to install the packages. As first step, upgrade the **pip** tool, using the following command:

```
(venv) $ pip install --upgrade pip
```

  - This command might take some time to complete, and potentially output a lot of text to the Terminal;
- With pip upgraded, confirm which Python packages are already installed. For this, use the following command:

```
(venv) $ pip list
```

- **Note:** This commands must be executed inside the virtual environment

GPIO Zero library

- Install the GPIO packages using the pip install command as follows:

```
(venv) $ pip install gpiozero pigpio
```

- Execute the `pip list` command again and notice if there is any difference. Take a snapshot of the packages you have previously installed using the `pip freeze` command:

```
(venv) $ pip freeze > requirements.txt
```

  - This command freezes all installed packages into a file named `requirements.txt`, which is a common filename to use for this purpose.

GPIO Zero library

- Look inside your `requirements.txt` file and notice its content. You can use the following command:

```
(venv) $ cat requirements.txt
```

- This is a good practice in case if you move your Python project to another machine or a new virtual environment;
- You can use the `requirements.txt file` to install all required packages for your project to work;
- Notice that whenever you install new packages with pip install you also will need to re-run `pip freeze > requirements.txt` to capture new packages and their dependencies.

And now, some exercises…
Go to Assignment 03!