Lab 4
Installing/configuring/using Raspberry Pi with sensors and actuators

Objective

The aim of this lab is to be able to install the operating system Raspberry PI OS, add the initial configurations, access it remotely over a network and create our first circuit.

What is Raspberry Pi

The Raspberry Pi is a series of small, affordable, and highly versatile single-board computers (SBCs) developed by the Raspberry Pi Foundation, a UK-based charity. Since its introduction in 2012, the Raspberry Pi has become extremely popular in the maker and hobbyist communities, as well as in education and industrial applications. It is designed to promote the teaching of basic computer science and programming concepts in schools and developing countries.

The Raspberry Pi features a Broadcom system-on-a-chip (SoC) with an ARM-based CPU, RAM, USB ports, HDMI output, audio output, and General Purpose Input/Output (GPIO) pins, which allow for interfacing with various sensors, actuators, and other hardware components. The Raspberry Pi runs a variety of operating systems, including Raspberry PI OS (a Debian-based Linux distribution), Ubuntu, Windows 10 IoT Core, and others.

The Raspberry Pi has been used in a wide range of projects, including home automation systems, media centers, retro gaming consoles, robotics, weather stations, and more. Its low cost, small form factor, and broad community support have made it a popular choice for DIY enthusiasts and professionals alike.

Note

In this lab, and to save time, the micro SD cards are supplied with the operating system pre-installed and configured,

Install Raspberry Pi OS using Raspberry Pi Imager

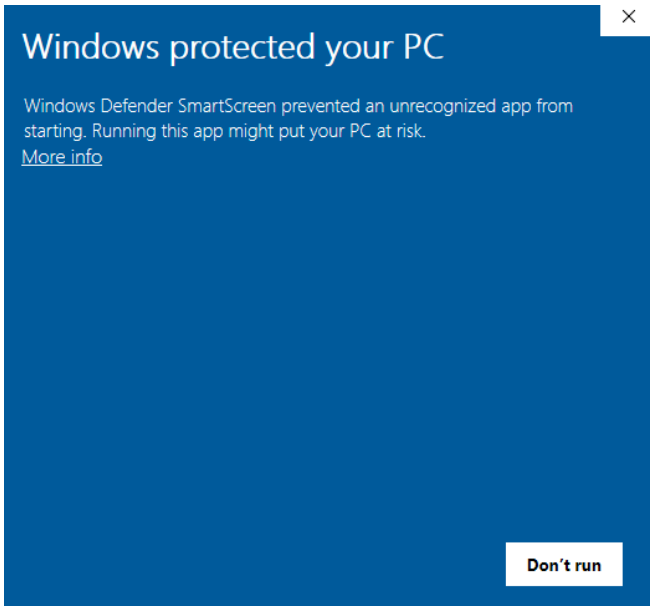
In order to install RaspberryPi OS, you need to use the **Imager app**. **The Raspberry Pi Imager** provides a fast and convenient method for installing Raspberry Pi OS and various other operating systems onto a microSD card, making it immediately compatible with your Raspberry Pi.

To begin, download and install the Raspberry Pi Imager on a computer equipped with an SD card reader. Then, insert the SD card intended for your Raspberry Pi into the reader and launch the Raspberry Pi Imager.

- Raspberry Pi imager for Windows
- Raspberry Pi imager for Mac OS
- Raspberry Pi Imager for Ubuntu x86

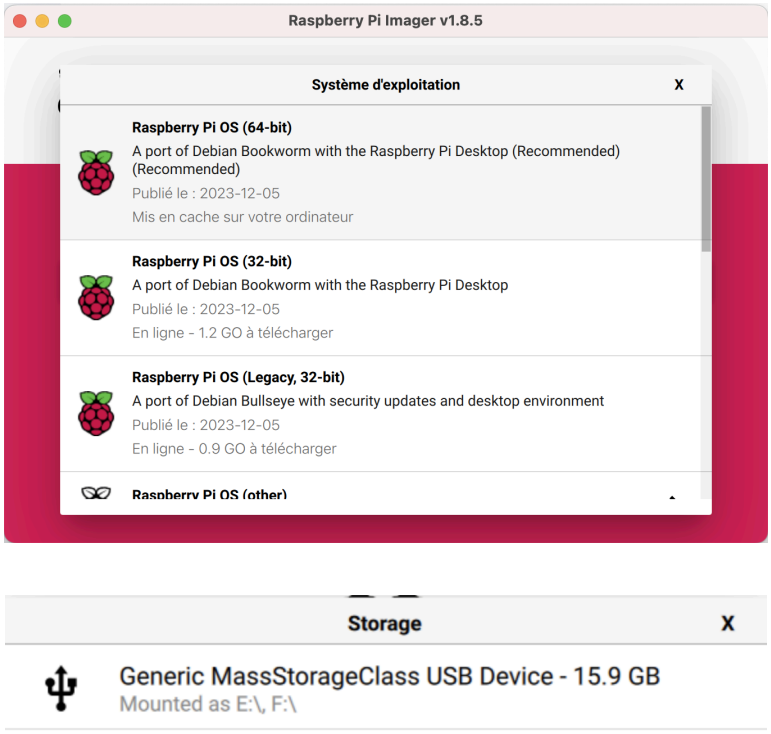
Warning

When launching the installer, your operating system might attempt to prevent its execution. For instance, on Windows, you might encounter the following notification:



- If this message appears, select "**More info**" and proceed by clicking "**Run anyway**".
- Follow the prompts to install and launch the Raspberry Pi Imager.
- Insert your SD card into the SD card slot of your computer or laptop.
- Within the **Raspberry Pi Imager**, choose the operating system you wish to install and the SD card where you want to install it.

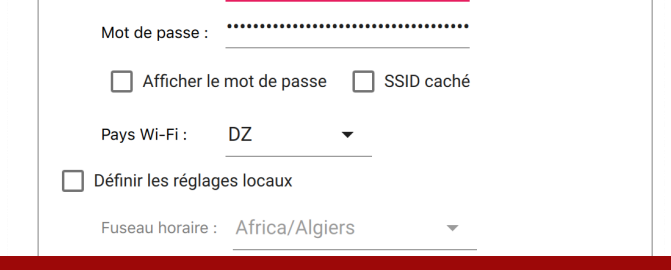
For the initial setup, an internet connection **is required** for the Raspberry Pi Imager to download the chosen operating system. After this initial download, the OS will be cached for future offline installations. If you remain connected to the internet for subsequent uses, the Imager will provide you with the most current version available.

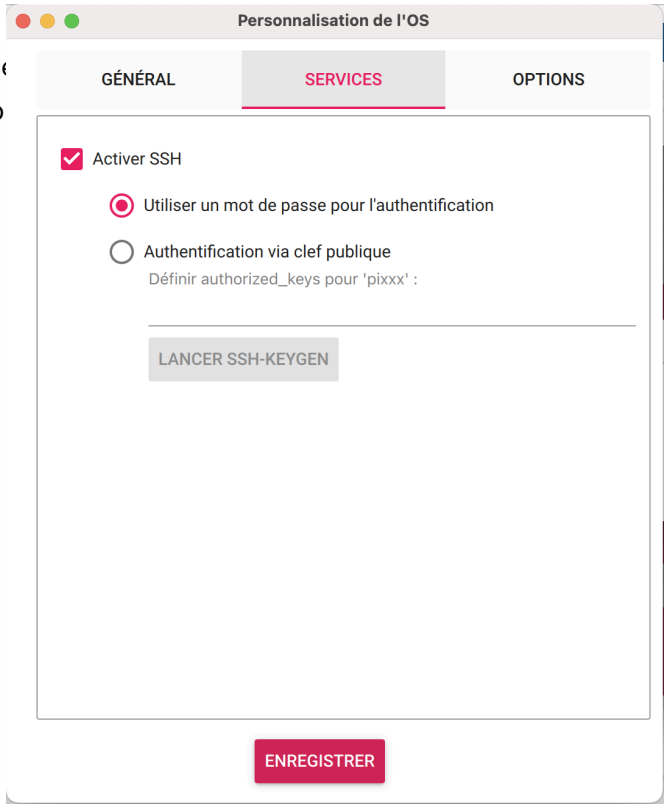


Once you have chosen the model of raspberry pi, the operating system and the destination memory card, click on "Next".

Then enter the basic settings, such as the raspberry pi hostname, username and password, WiFi configuration (SSID and key).

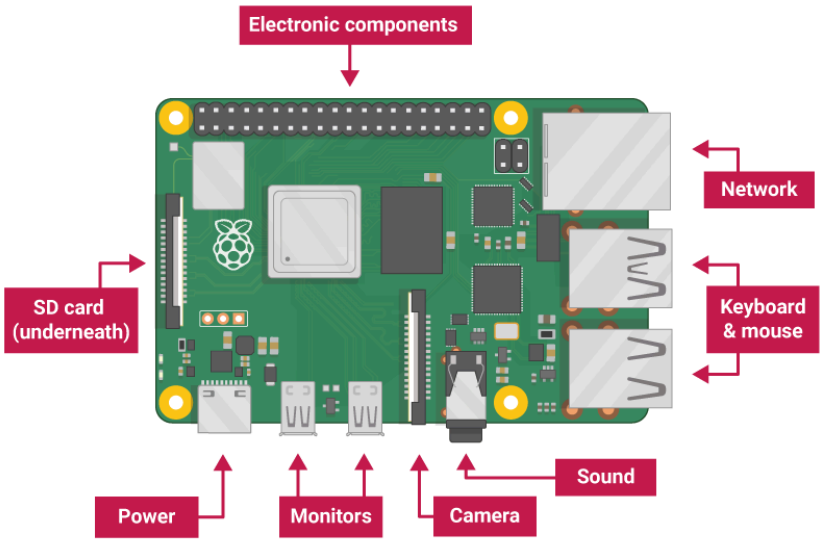
And finally, on the **Services** tab, you need to enable SSH so that you can access the raspberry pi remotely on the network. To do this, click on the tab.



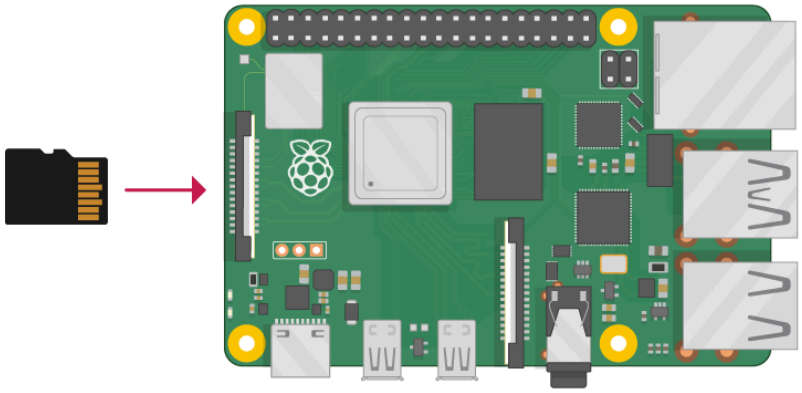


Connect the Raspberry Pi

Now, it's time to connect everything to your Raspberry Pi. It's crucial to do this in the correct sequence to ensure the safety of all your components.



Insert the **microSD card**, which has been prepared with Raspberry Pi OS, into the microSD card slot located on the bottom side of your Raspberry Pi.



If required, a keyboard and USB mouse can be connected, as well as a screen, using a cable with a mini HDMI port (up to two screens can be connected).

How to connect remotely to the Raspberry Pi using SSH

Sometimes, you may require access to a Raspberry Pi without the need to connect it to a monitor. This could be due to the Raspberry Pi being integrated into a device such as a robot, or if you wish to remotely access information from it. Alternatively, you might not have an extra monitor, mouse or a keyboard available.

Every device connected to a Local Area Network (LAN) is allocated an IP address. To connect to your Raspberry Pi from another device using **SSH** or **VNC**, you require the Raspberry Pi's IP address. If you have a display connected, obtaining this information is straightforward. However, various methods exist for remotely locating it from another machine on the network.

SSH from Linux/MacOS

You can use SSH to connect to your Raspberry Pi from a Linux desktop, another Raspberry Pi, or from an Apple Mac without installing additional software.

```
ssh <username>@<hostname>
```

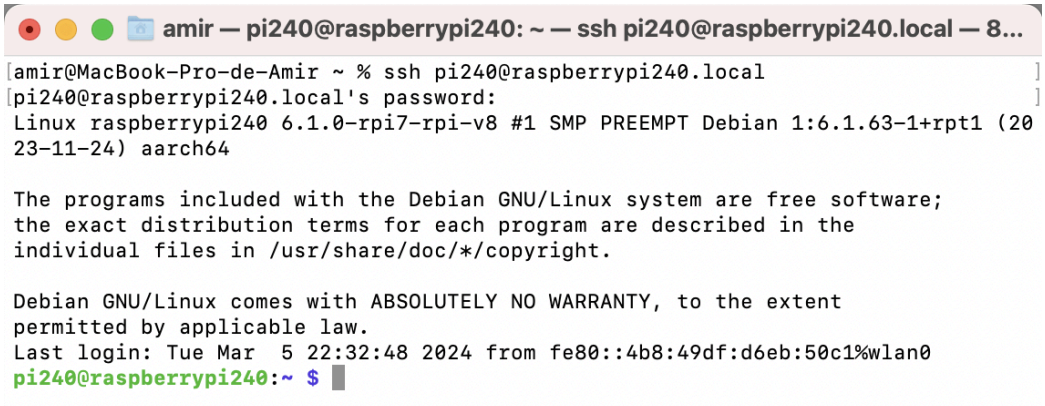
For example: `ssh pixxx@raspberrypixxx.local`

Once the connection is established, a security/authenticity warning will appear. Please type "yes" to proceed. This warning will only appear during the initial connection.

Next, enter your account password when prompted.

You should now see the Raspberry Pi prompt, which mirrors the one found on the Raspberry Pi itself.

```
pixxx@raspberrypixxx: ~ $
```



You are now remotely connected to the **Raspberry Pi** and you can execute commands.

SSH from Windows

You can use SSH to connect to your Raspberry Pi from a Windows 10 computer with the October 2018 Update or later installed, without needing third-party clients.

Launch a terminal window on your computer and replace "<IP>" with the IP address of the Raspberry Pi you wish to connect to.

```
ssh <username>@<hostname>
```

If the SSH command doesn't work on Windows command prompt, you can download the Putty application

How to retrieve an IP address without using the hostname or a monitor

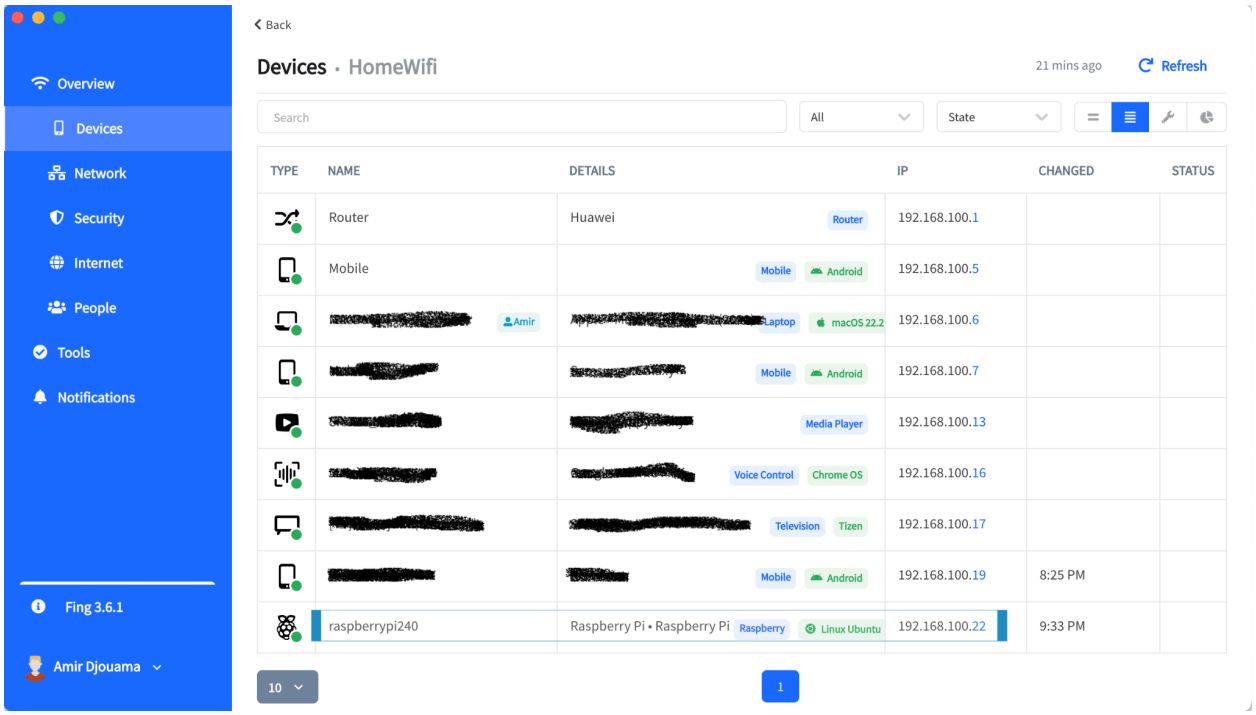
Start by plugging in your raspberry pi and connecting it to the network. Once connected, you can use applications such as Angry IP scanner (for Windows, MacOS and Linux) or Fing app (for Windows, MacOS, iOS and Android).

Important

You should be connected to the **same network** as the **Raspberry Pi**, in our case, you should be connected to the **ENSIA** WiFi network.

The rest of this tutorial will use the **Fing application**.

To retrieve the IP address of your raspberry pi, launch the Fing application and click on Devices. After that, the list of devices connected to the network will be displayed on the application, so all you have to do is find your raspberry by seeing the IP address to its right.



Expand file system

The Raspberry Pi Foundation's system images are crafted to be as compact as possible to accommodate small SD cards. However, when applied to larger SD cards (usually exceeding 4GB), a substantial portion of the card remains unused. By expanding the primary partition to utilize this unallocated space, the full storage capacity of the device becomes accessible.

Once connected to the Raspberry Pi, type the following commands:

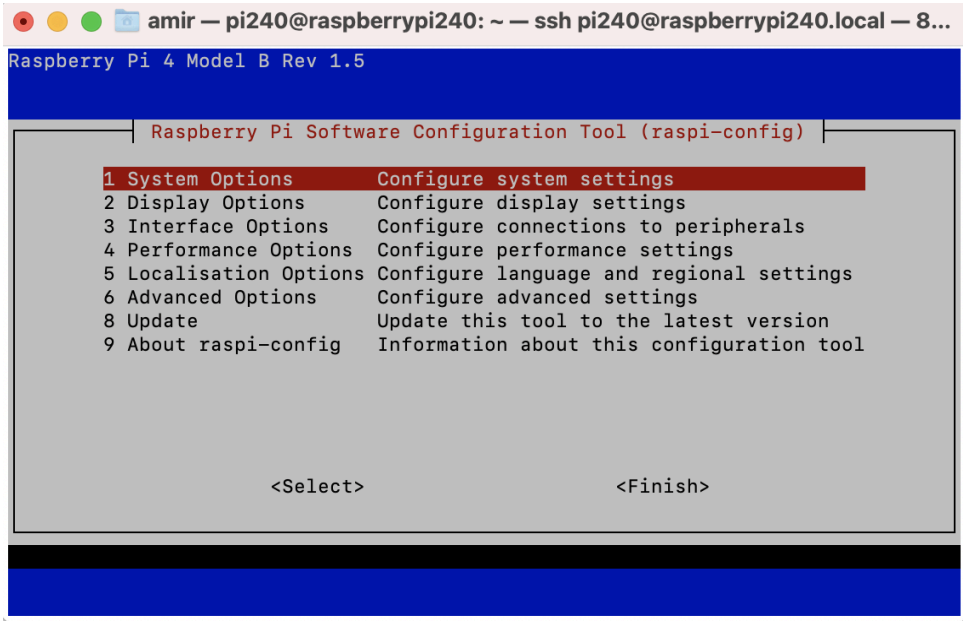
`pixxx@raspberrypixxx: ~ $ sudo apt-get update && apt-get upgrade` (to update raspberry pi and the installation repositories),

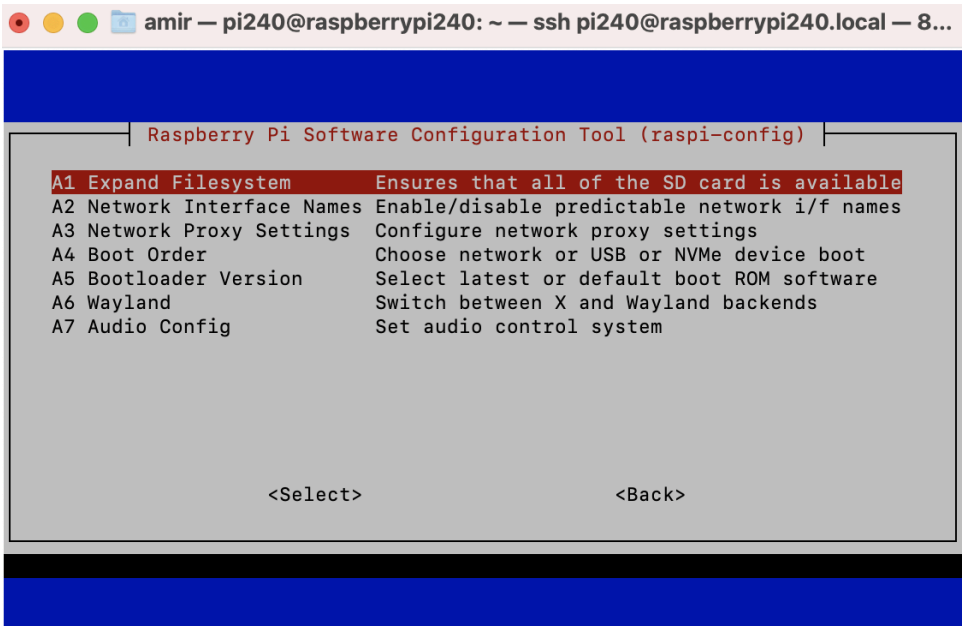
Once you have finished, run the following command:

```
pixxx@raspberrypixxx: ~ $ sudo raspi-config
```

Choose: `6 Advanced Options > A1 Expand Filesystem`

Select `<Finish>` and reboot the Raspberry Pi





Setup the VNC server

When working with Linux platforms, **Virtual Network Computing (VNC)** emerges as the primary choice. It serves as a cross-platform remote desktop technology compatible with most operating systems. There exists a plethora of products in the market, some free and others commercial. However, many of the latest Raspberry Pi distributions come equipped with RealVNC, which seamlessly integrates cloud-based connections. This feature enables connectivity not only within local networks but also over the internet, granting users the freedom to access their Pi from anywhere. Such flexibility renders VNC a more prominent option compared to others. Let's delve into setting up VNC on the Raspberry Pi.

In order to enable the VNC server, you need to open the `raspi-config` window:

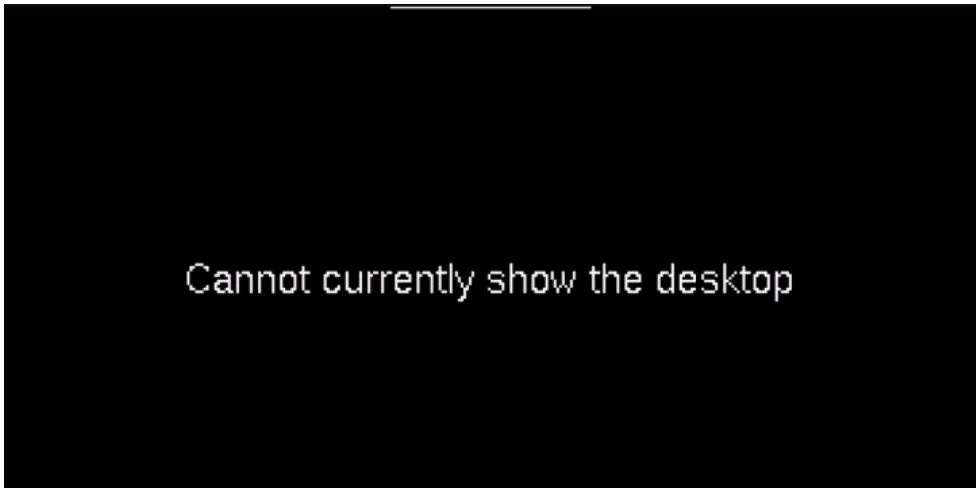
Choose: `3 Interfaces Options > I2 VNC >` Select `<Yes>`. Select Finish to quit (or ESC)

Once enabled, you have to install the Real VNC server application

```
pixxx@raspberrypixxx: ~ $ sudo apt-get install realvnc-vnc-server
```

You need to download the VNC viewer and download the client for your operating system and install it. Once completed, launch the VNC viewer and enter the Raspberry Pi's IP address, then press enter.

Note: You ended up in some sort of problem if you unplug the monitor, mouse, and keyboard from the Pi and try accessing its GUI desktop over VNC




```
pixxx@raspberrypixxx: ~ $ sudo raspi-config
```

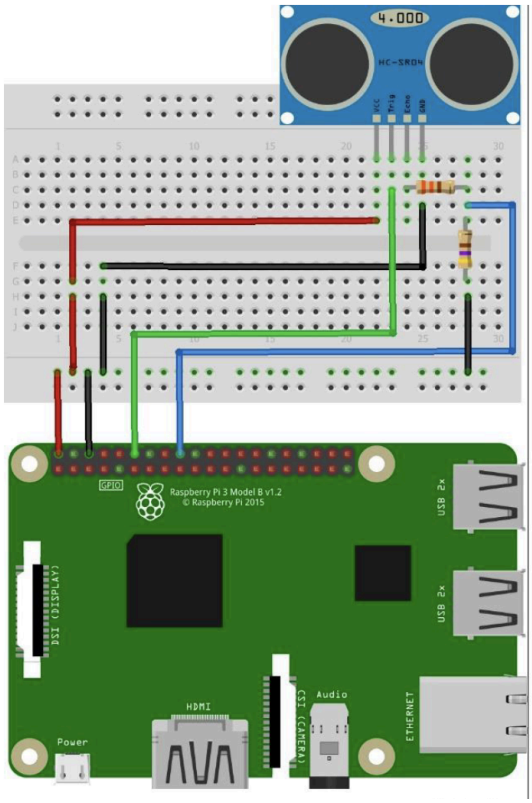
Choose: 7 `Advanced Options` > Set the resolution as per your screen size. We recommend setting DMT Mode 82 1920 X 1080 69 Hz 16:9.

Ultrasonic sensor and Raspberry Pi

The components

- Raspberry Pi
- Micro SD card with Raspberry PI OS
- USB Type-C cable
- Breadboard
- Ultrasonic sensor
- Resistors

Wiring scheme



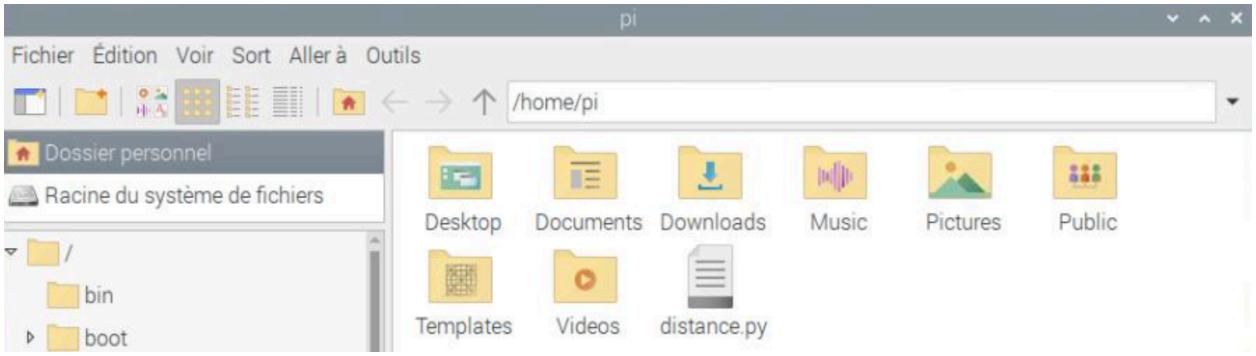
Raspberry Pi	Ultrasonic sensor
5V	VCC
GND	Gnd
GPIO 24	Echo*
GPIO 18	Trig

* The Echo output features a voltage divider based on a 330 Ω resistor (orange, orange, brown and gold) and one of 470 Ω (yellow, violet, brown and gold).

This divider bridge attenuates the 5 Vdc voltage at the sensor output so as not to damage the digital input digital input of the Raspberry Pi board (3.3 Vdc compatible only).

The detailed Python code example can be downloaded directly from this address. Save or move the code to your `/home/pi` folder.

You can use a VNC client to remotely access the RPi desktop or using the terminal (SSH) or using code editor over ssh.

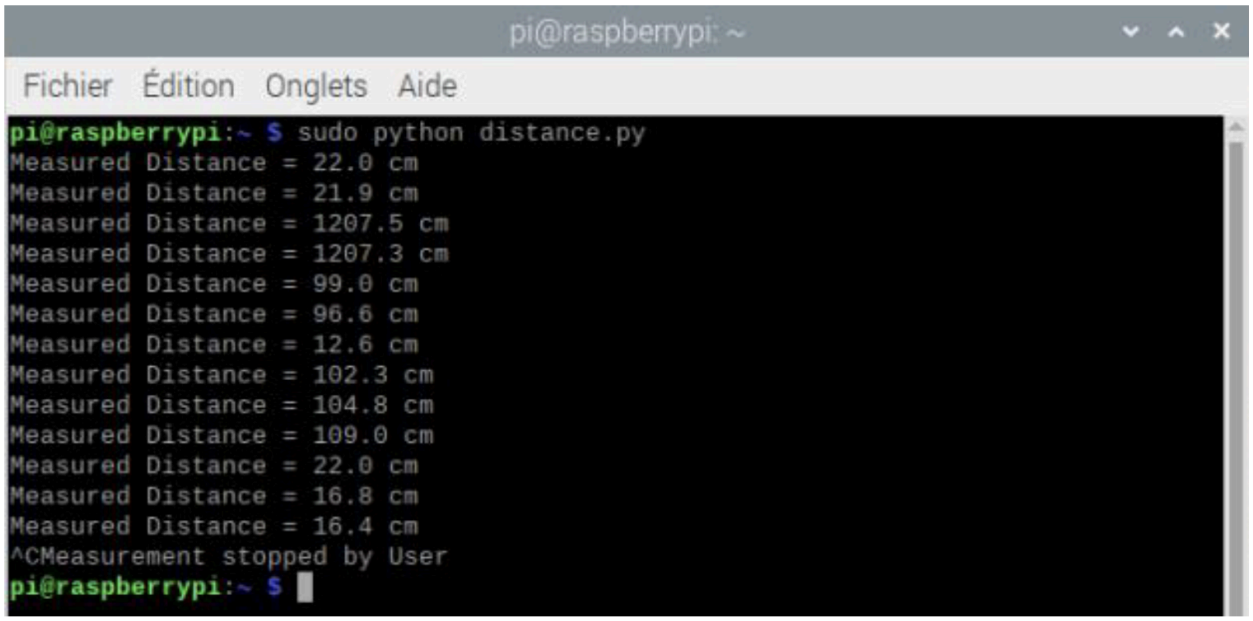


Now you can run the code in a terminal:

```
pixxx@raspberrypixxx: ~ $ sudo python distance.py
```

This program measures the distance between your **ultrasonic sensor** and an object, wall, etc (as seen in the lab with Arduino).

This distance is displayed directly in the terminal.



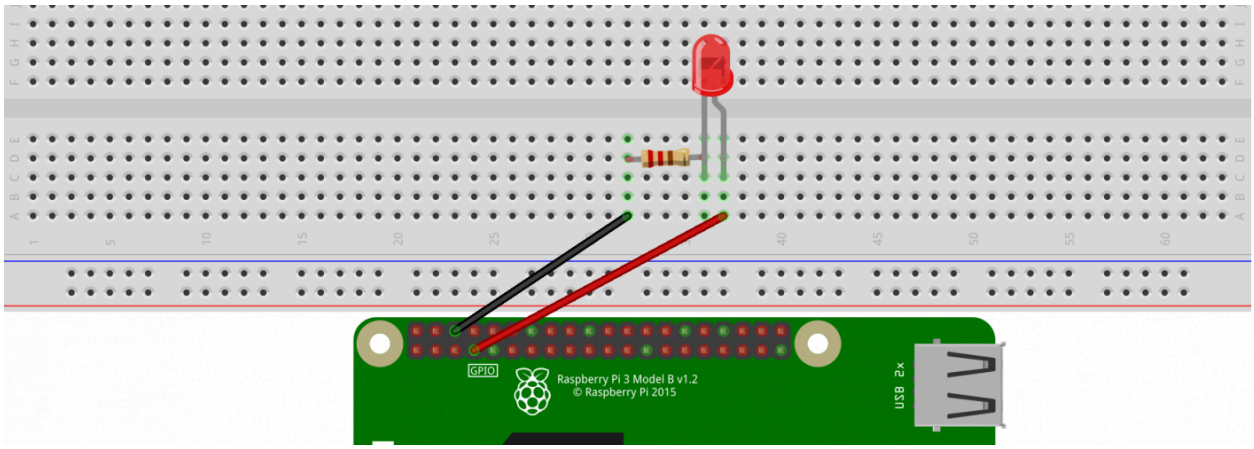
To stop the measurement, press **CTRL+C**.

LEDs and Raspberry Pi

The components

- Raspberry Pi
- Micro SD card with Raspberry PI OS
- USB Type-C cable
- Breadboard
- Resistors
- Set of LEDs

Wiring scheme



Connect the **red wire** to **GPIO 7**, and the **black wire** to **GPIO 6**. Connect the two wires to your breadboard a few rows apart. Connect the red wire to the anode of your LED (the long leg), one leg of the resistor to the cathode (the other leg of the LED, the short leg), and the other leg of the resistor to the black cable.

Once the circuit is complete, simply download the source code from this address.

Once that's done, we'll make the program executable. To do this, simply run the following command:

```
pixxx@raspberrypixxx: ~ $ chmod +x /home/pi/led.py
```

All that's left is to run the script by calling it like this:

```
pixxx@raspberrypixxx: ~ $ /home/pi/led.py
```

Each time you run the script, the LED will light up if it is off, and go out if it is on.

Challenge: LED and Ultrasonic

Now that we know how to read and display the distance with an ultrasonic sensor, and we know how to switch a LED on and off, all we have to do is plug the unit in and display a red LED when we reach a threshold distance and a green LED when we move away.