

Internet of Things

Raspberry Pi & Arduino

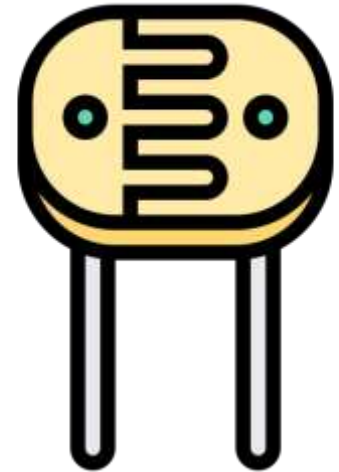
Serial Communication

Abdallah El Ghamry

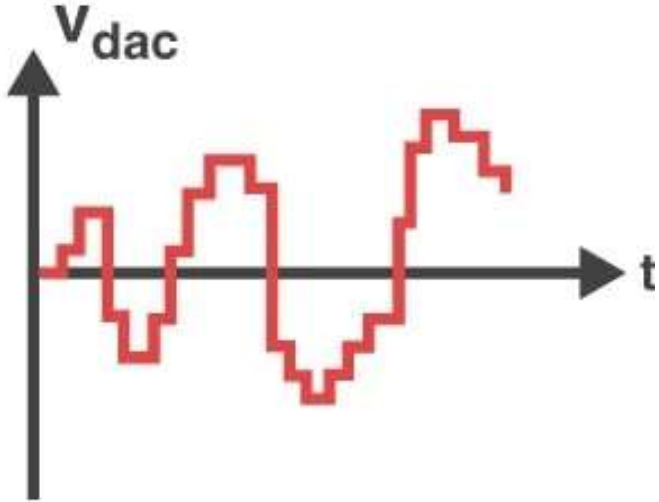
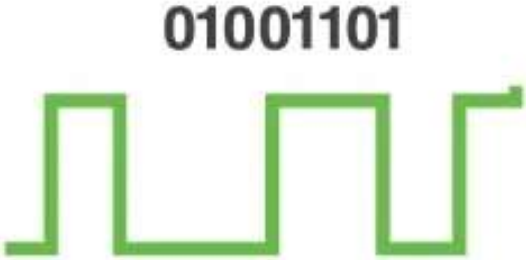
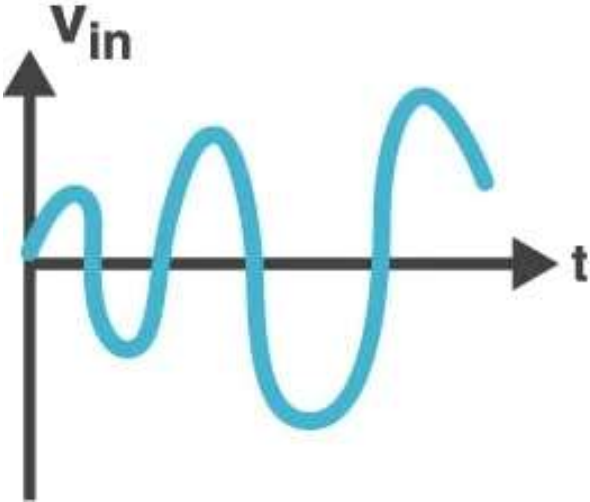
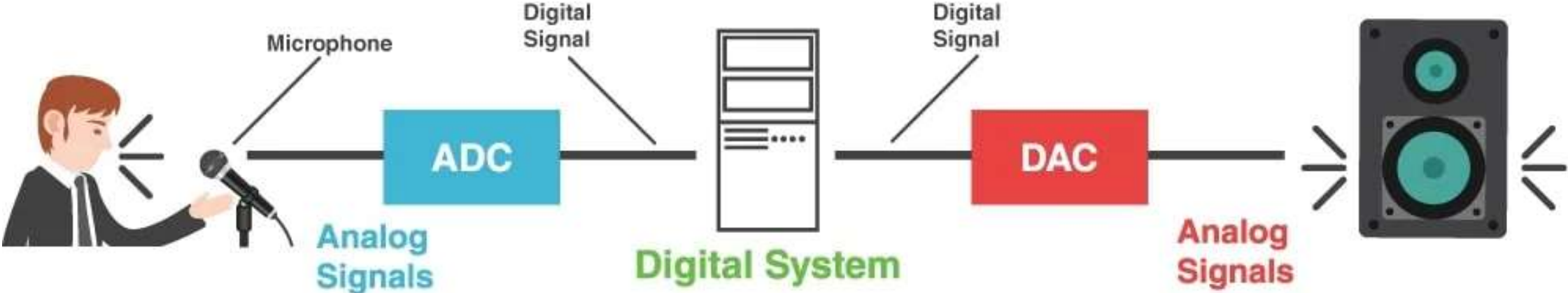


Sensors

- A **sensor** is a device that detects some type of input from the **physical environment**.
- The input can be **light**, **heat**, **motion**, **pressure** or any number of other environmental phenomena.



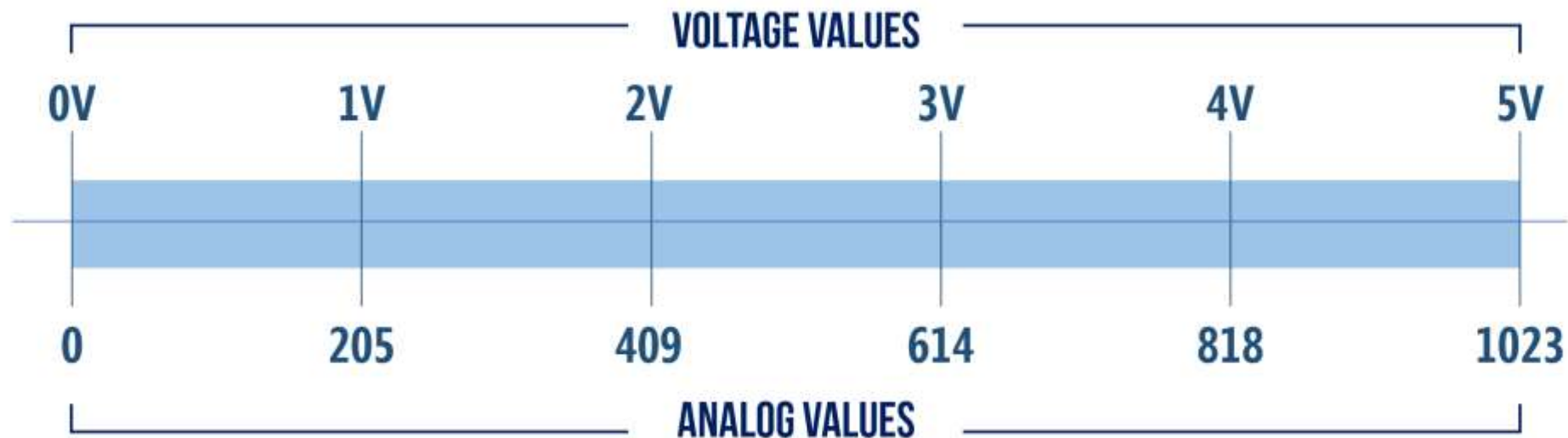
ADC vs. DAC



Read Analog Voltage

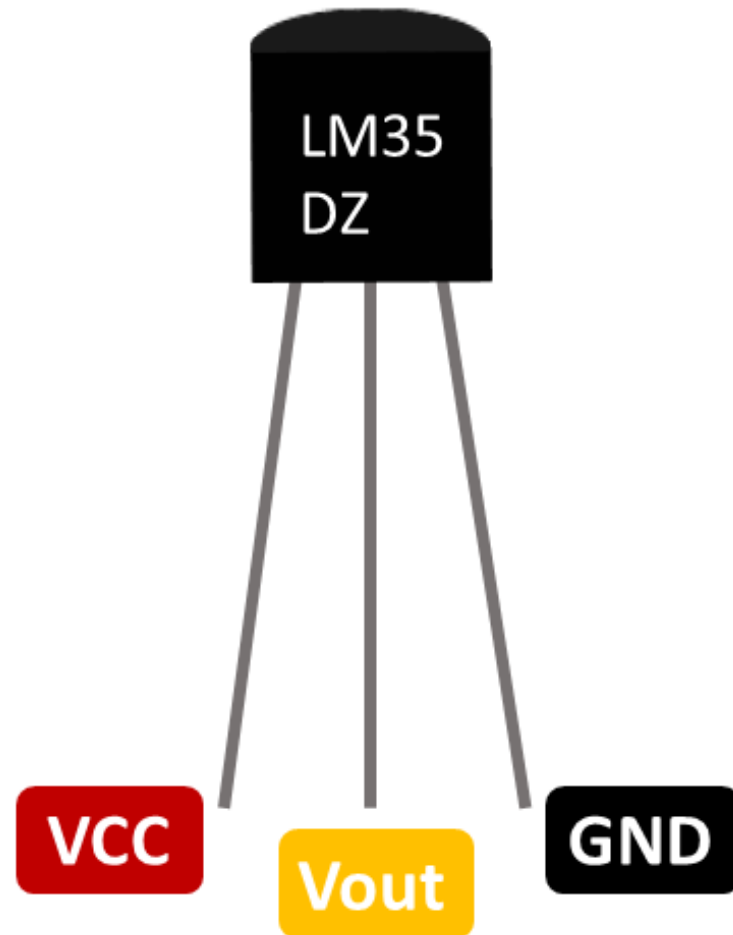
- The `analogRead()` returns a **number between 0 and 1023** that is **proportional to the amount of voltage** being applied to the pin.
- To scale the numbers between 0 and 5, divide 5 by 1023 and multiply that by `sensorValue`:

`voltage = sensorValue * (5.0 / 1023.0);`



LM35 Temperature Sensor

- The LM35 sensor is an **analog temperature sensor**.
- The LM35 can measure temperature in the range of -55°C to 150°C .



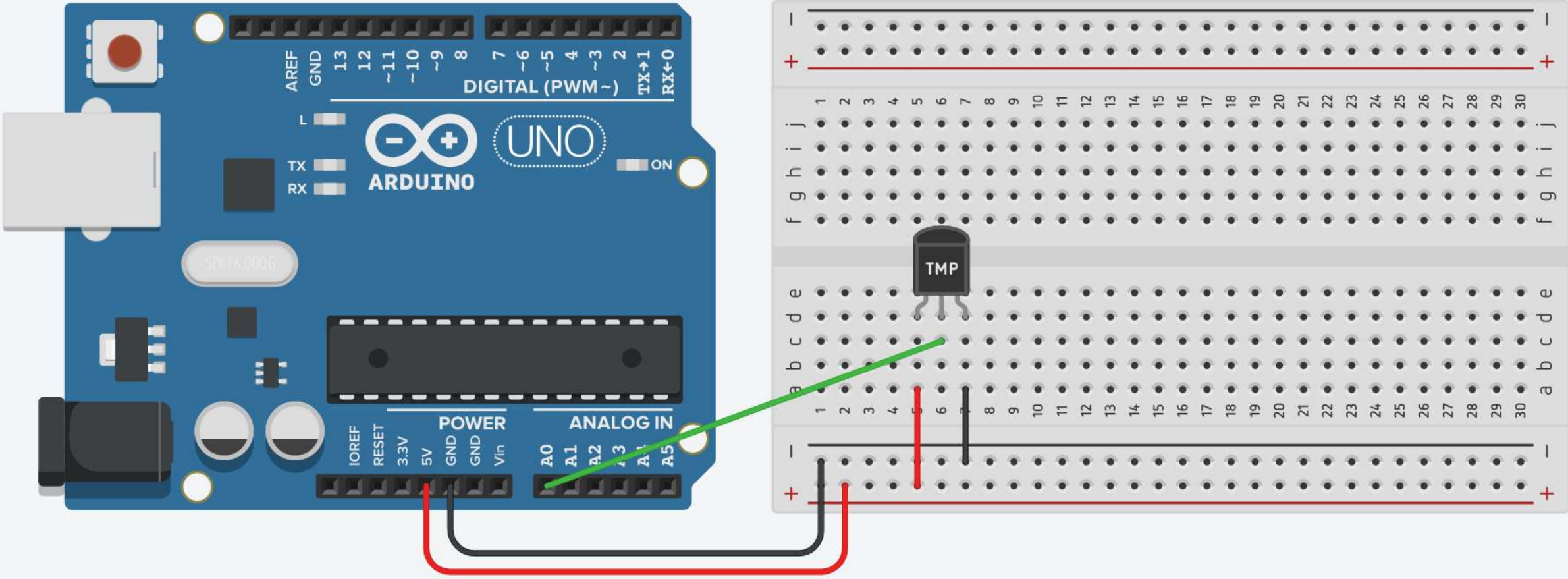
LM35 Temperature Sensor: Specifications

Criteria	Description
Power supply	4V to 30V
Temperature Range	-55°C to +155°C
Accuracy	±0.5°C
Output Scale Factor	10mV/°C
Output at 25°C	250mV

LM35 Temperature Sensor: Components

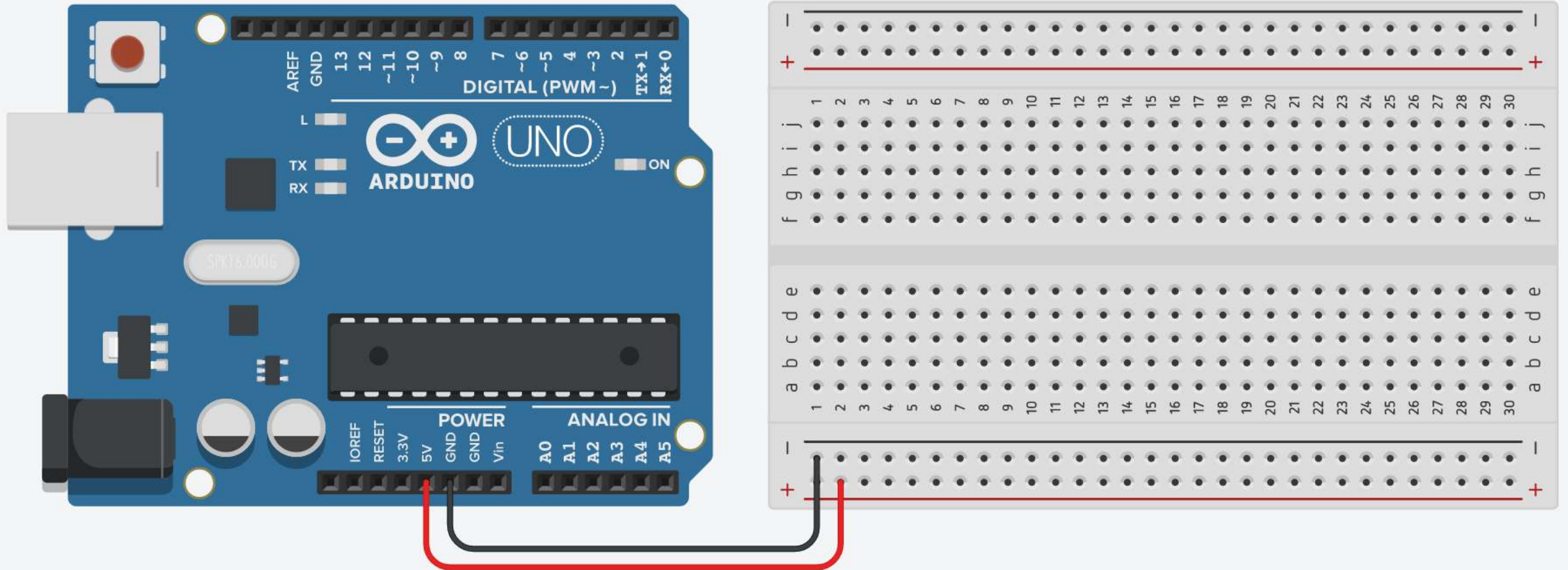
- You need
 - Arduino
 - LM35 Sensor
 - Jumpers
 - Breadboard

LM35 Temperature Sensor: Circuit



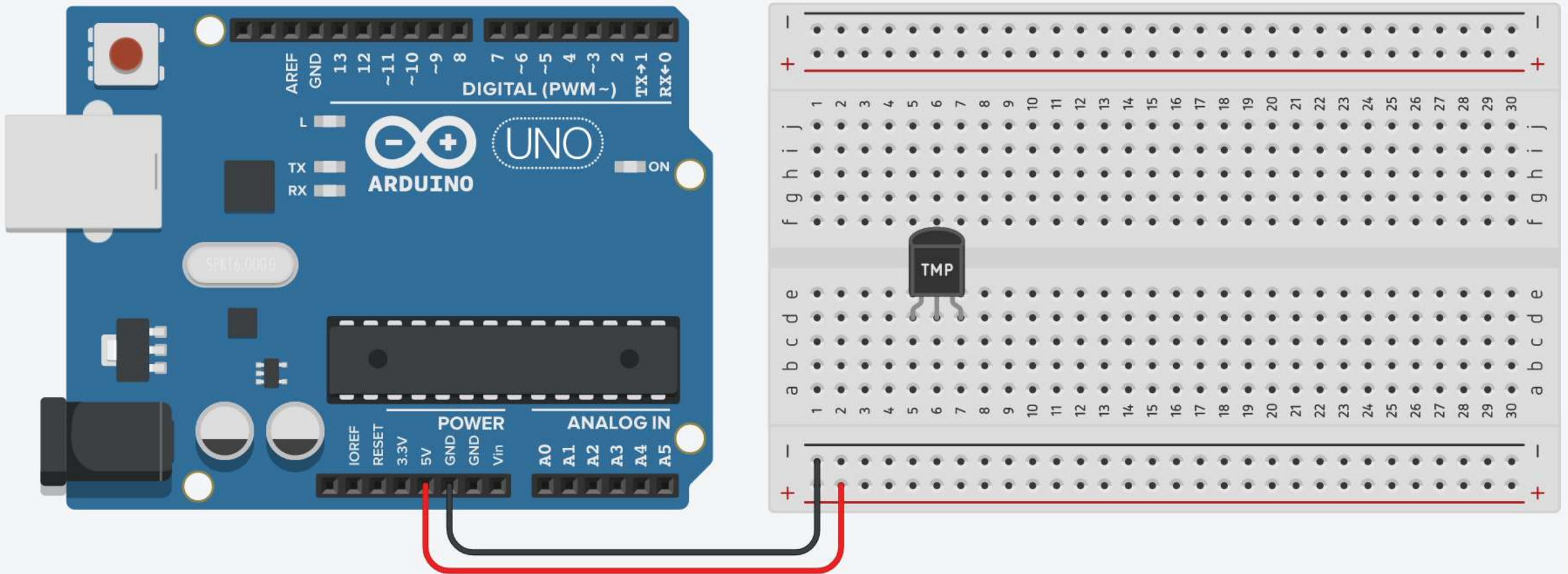
LM35 Temperature Sensor: Steps

1. Connect breadboard **power (+)** and **ground (-)** rails to Arduino **5V** and **ground (GND)**, respectively.



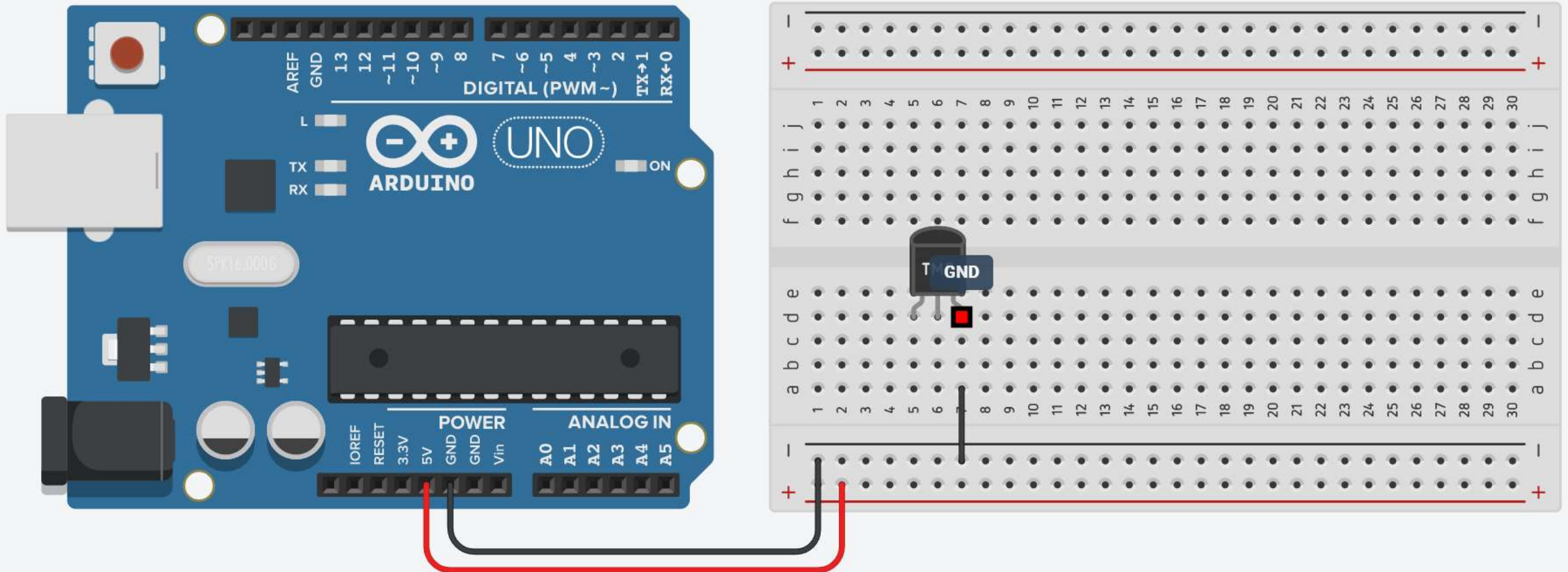
LM35 Temperature Sensor: Steps

2. Plug the **LM35 sensor** into the breadboard.



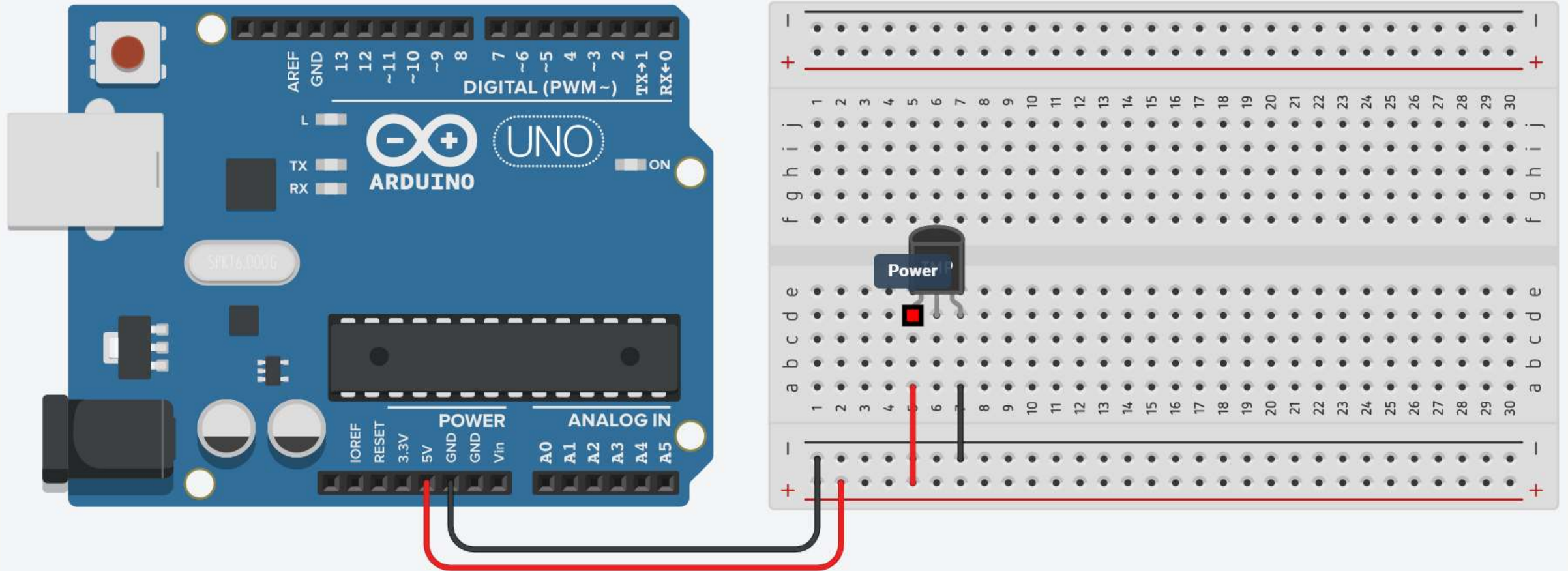
LM35 Temperature Sensor: Steps

3. The sensor **GND** pin connects to the **ground** on Arduino.



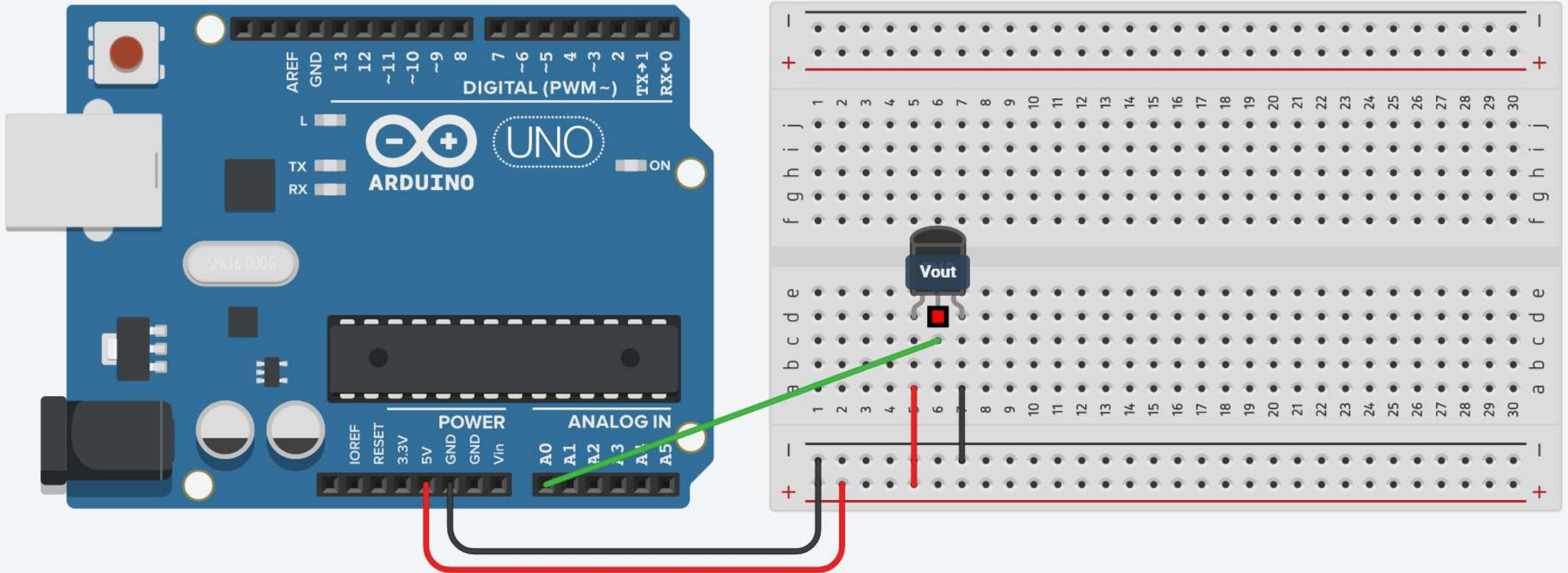
LM35 Temperature Sensor: Steps

- The sensor **Power** pin connects to the **5V** on Arduino.



LM35 Temperature Sensor: Steps

5. Wire up the sensor **Vout** pin to the analog pin **A0** on Arduino.



LM35 Temperature Sensor: Code

```
int sensor_value;           // Variable to store sensor reading
float volt;                 // Variable to store the voltage
float temp;                // Variable to store temperature (Celsius)

void setup() {
  Serial.begin(9600);       // Begin serial communication at 9600 baud rate
}

void loop() {
  sensor_value = analogRead(A0);           // Reading the value from sensor
  volt = sensor_value * (5.0 / 1023.0);    // Convert sensor reading into voltage
  temp = volt * 100;                       // Convert voltage into temperature

  // Print the temperature in Celsius
  Serial.print(temp);
  Serial.println(" C");

  delay(500);                             // Short delay
}
```

LM35 Temperature Sensor: Alternative Code

```
int sensor_value;           // Variable to store sensor reading
float temp;                 // Variable to store temperature (Celsius)

void setup() {
  Serial.begin(9600);       // Begin serial communication at 9600 baud rate
}

void loop() {
  sensor_value = analogRead(A0);           // Reading the value from sensor
  temp = sensor_value * (500.0 / 1023.0); // Convert sensor reading into voltage

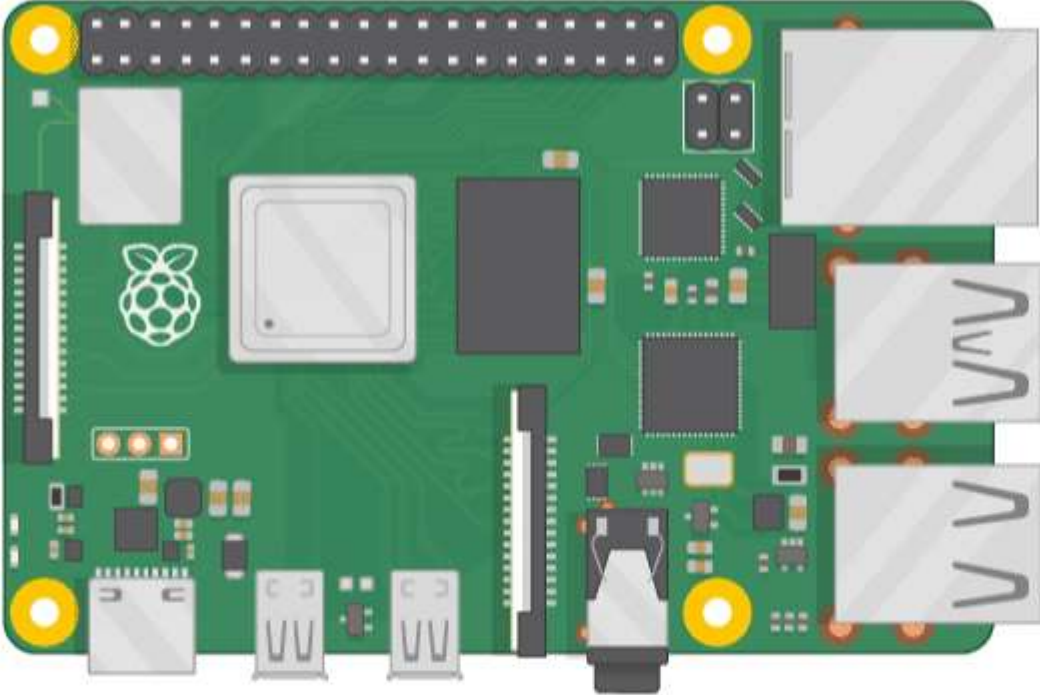
  // Print the temperature in Celsius
  Serial.print(temp);
  Serial.println(" C");

  delay(500);                          // Short delay
}
```

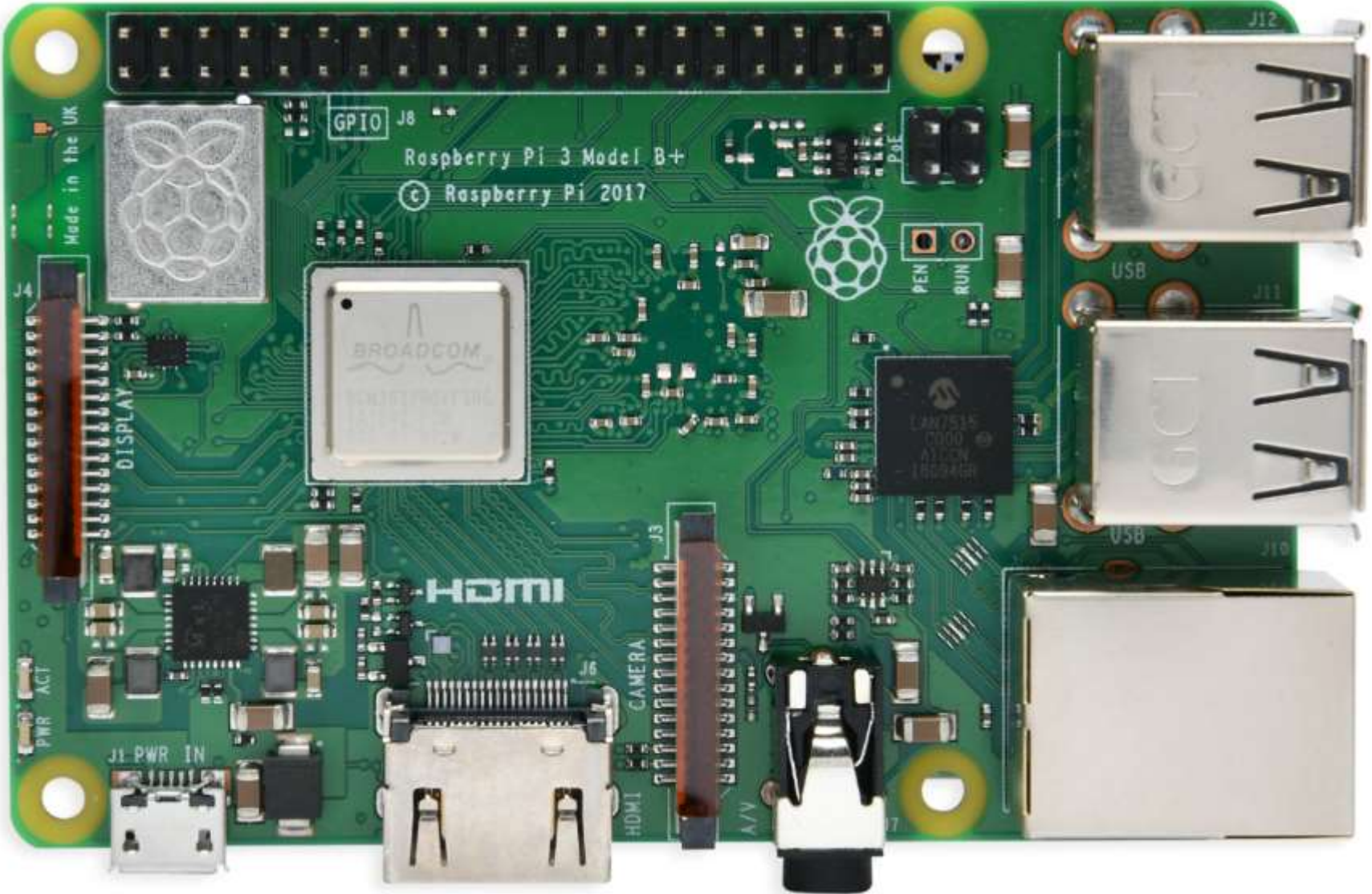

Raspberry Pi

- The Raspberry Pi is capable of **doing all the things you'd expect from a computer.**
- Everything from **browsing the internet and playing games, to watching movies and listening to music.**
- Raspberry Pi is known as a **single-board computer**, but that **doesn't mean it's not powerful.**
- Raspberry Pi can do **anything a bigger computer can do.**
- Over the years, the Raspberry Pi has evolved, **increasing its memory, improving its performance**, and adding features.

Raspberry Pi Ports



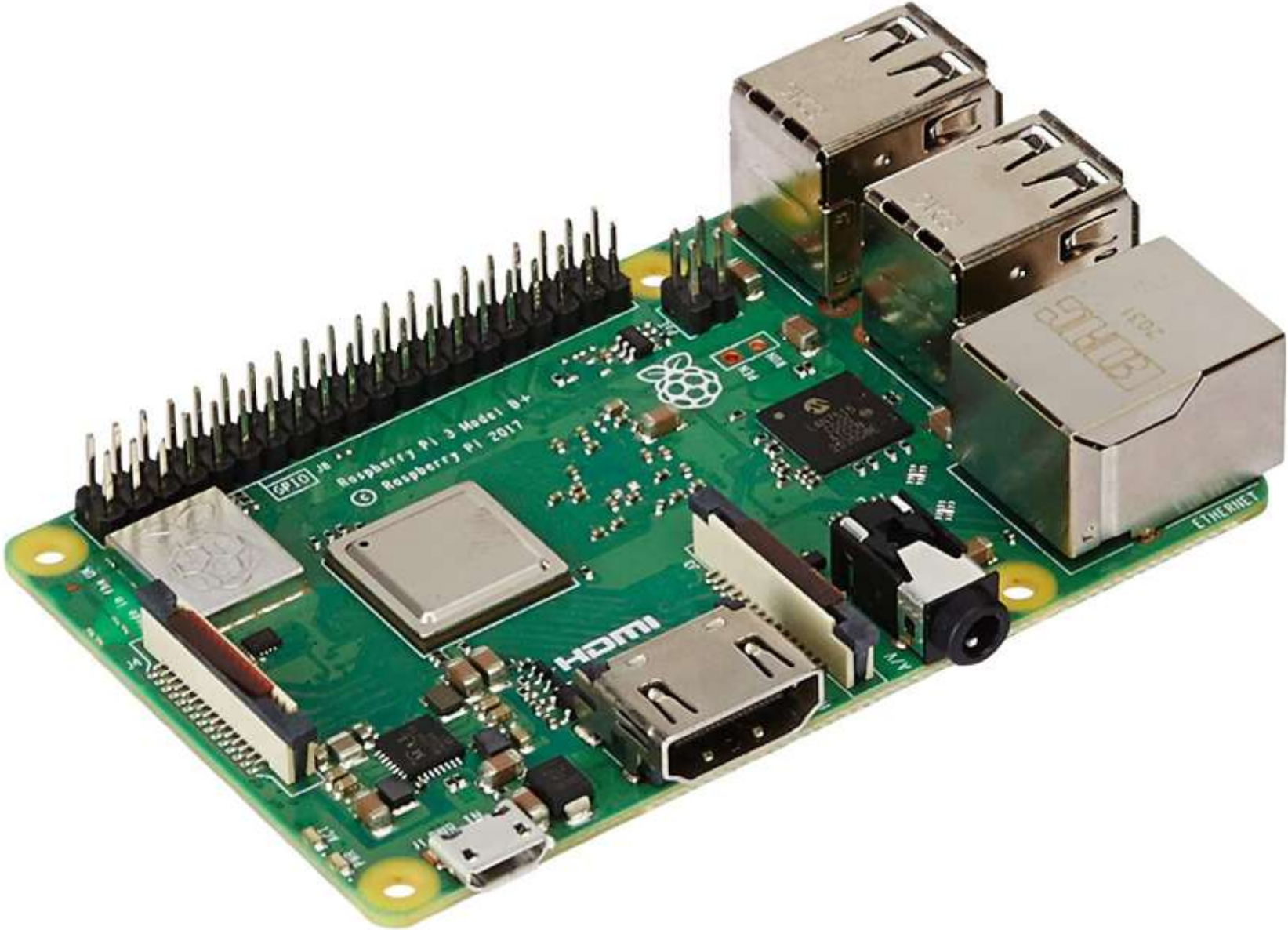
Raspberry Pi 3 Model B+



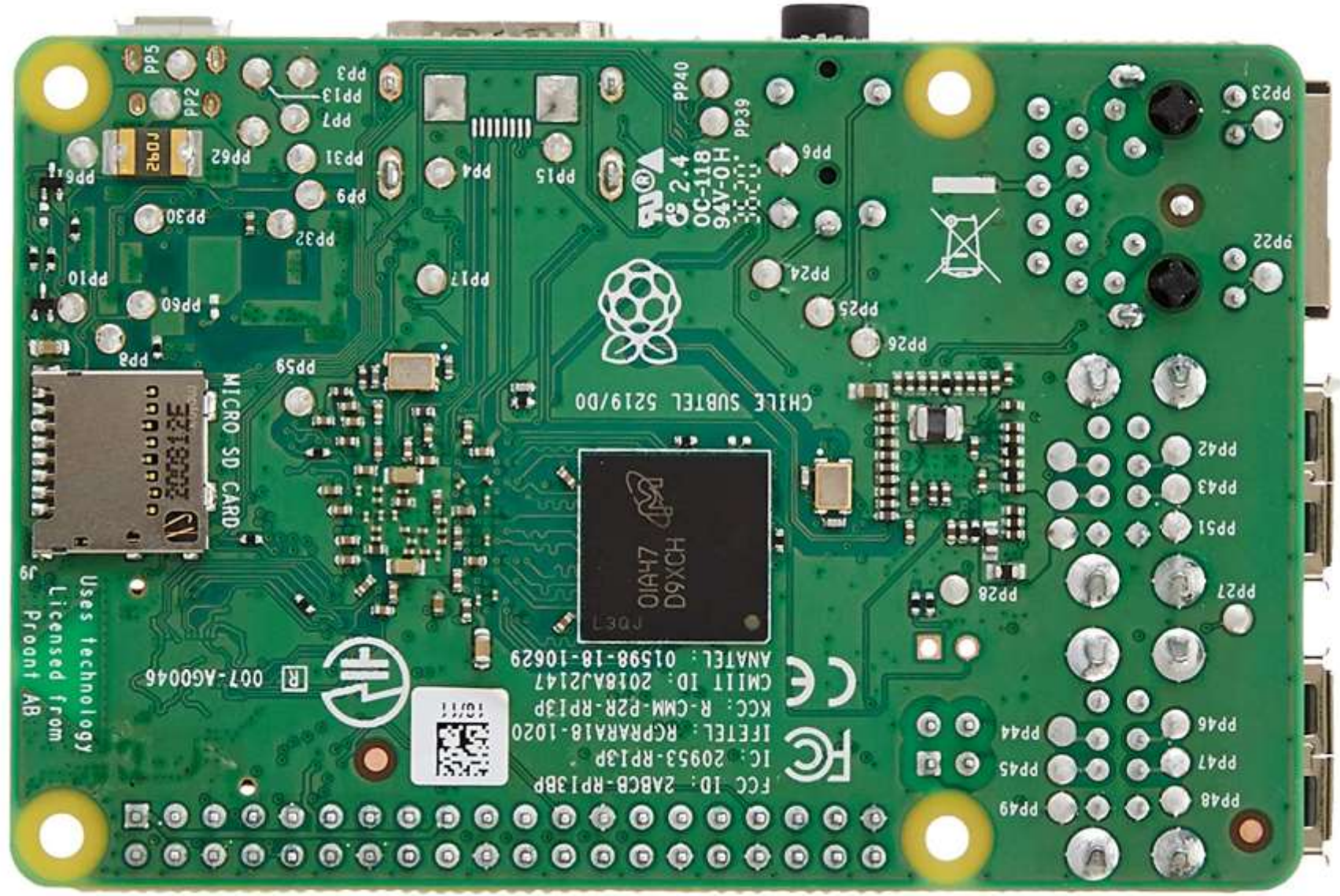
Raspberry Pi 3 Model B+



Raspberry Pi 3 Model B+



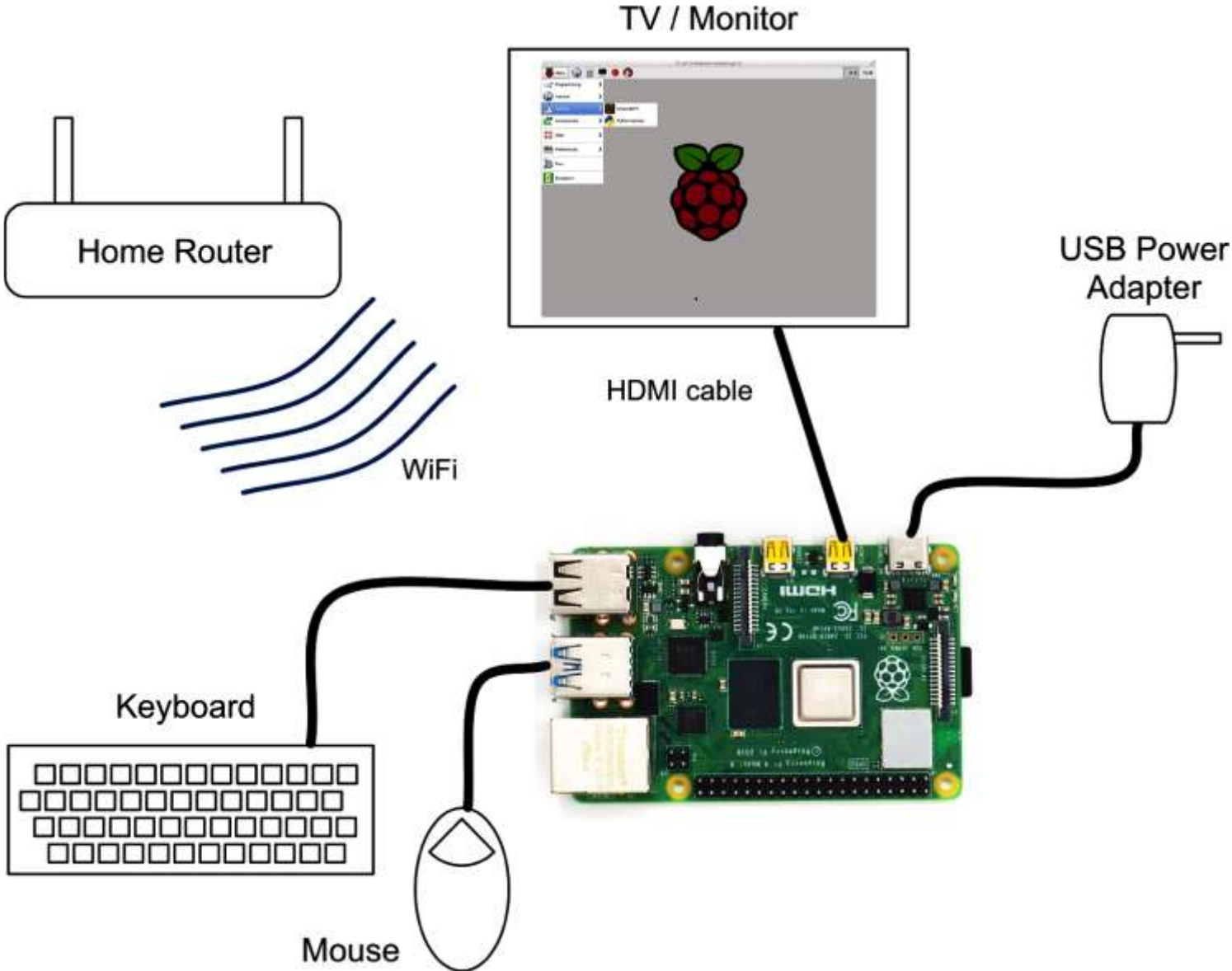
Raspberry Pi 3 Model B+



Raspberry Pi 3 Model B+: Specifications

Processor	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz
Memory	1GB LPDDR2 SDRAM
Connectivity	<ul style="list-style-type: none">▪ 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2, BLE▪ Gigabit Ethernet over USB 2.0▪ 4 × USB 2.0 ports
Access	Extended 40-pin GPIO header
Video & Sound	<ul style="list-style-type: none">▪ 1 × full size HDMI▪ MIPI DSI display port▪ MIPI CSI camera port▪ 4 pole stereo output and composite video port
SD Card Support	Micro SD format for operating system and data storage

Raspberry Pi System



Raspberry Pi 4 Model B

More powerful processor

Choice of RAM

- 1GB
- 2GB
- 4GB
- 8GB

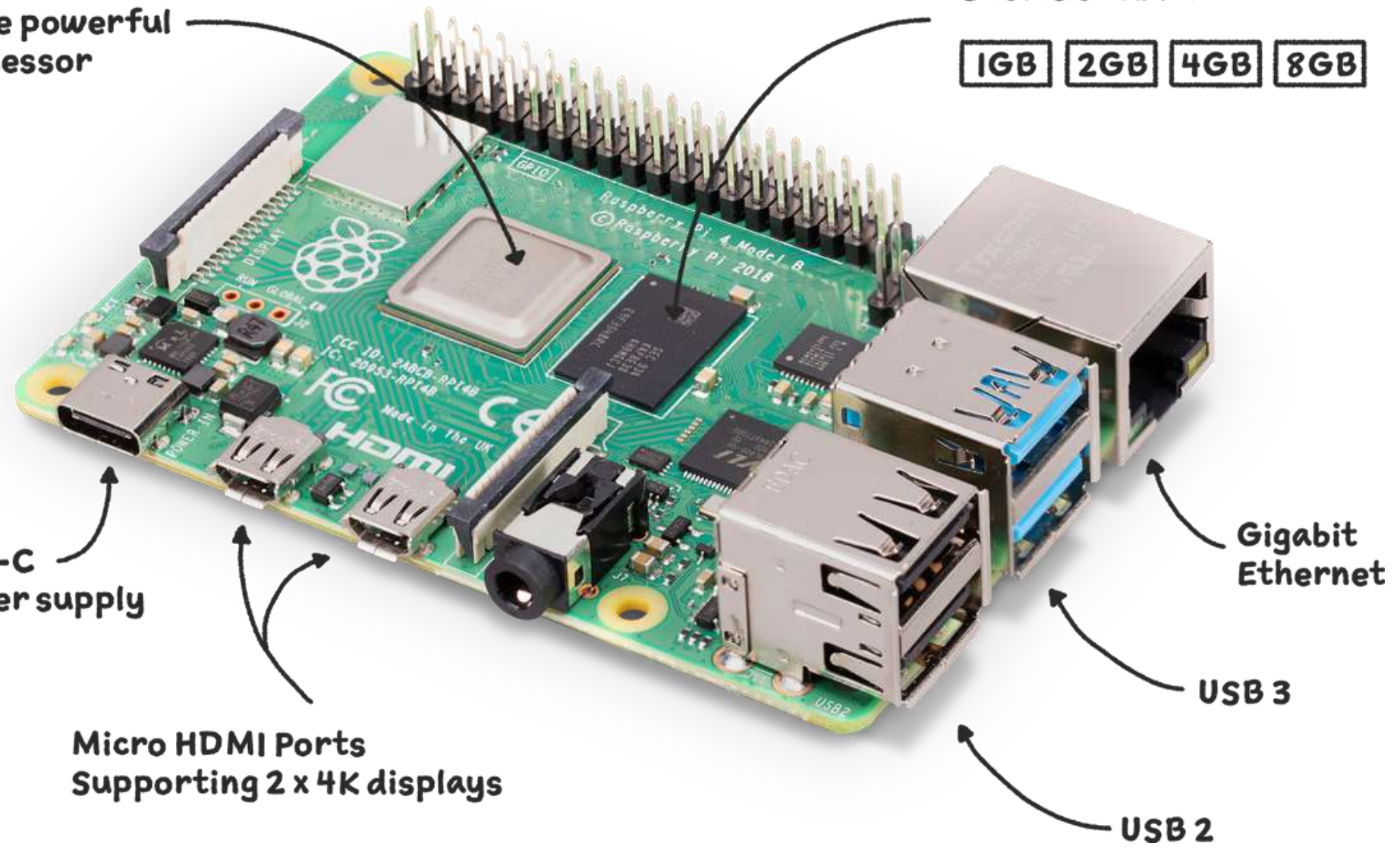
USB-C Power supply

Micro HDMI Ports Supporting 2 x 4K displays

Gigabit Ethernet

USB 3

USB 2

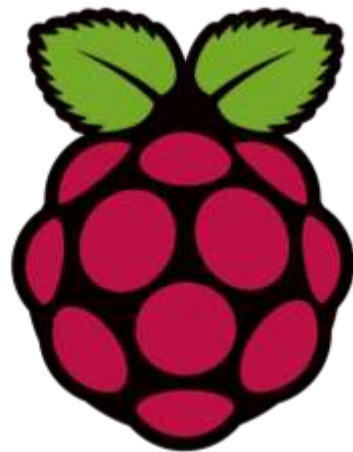


Raspberry Pi Models

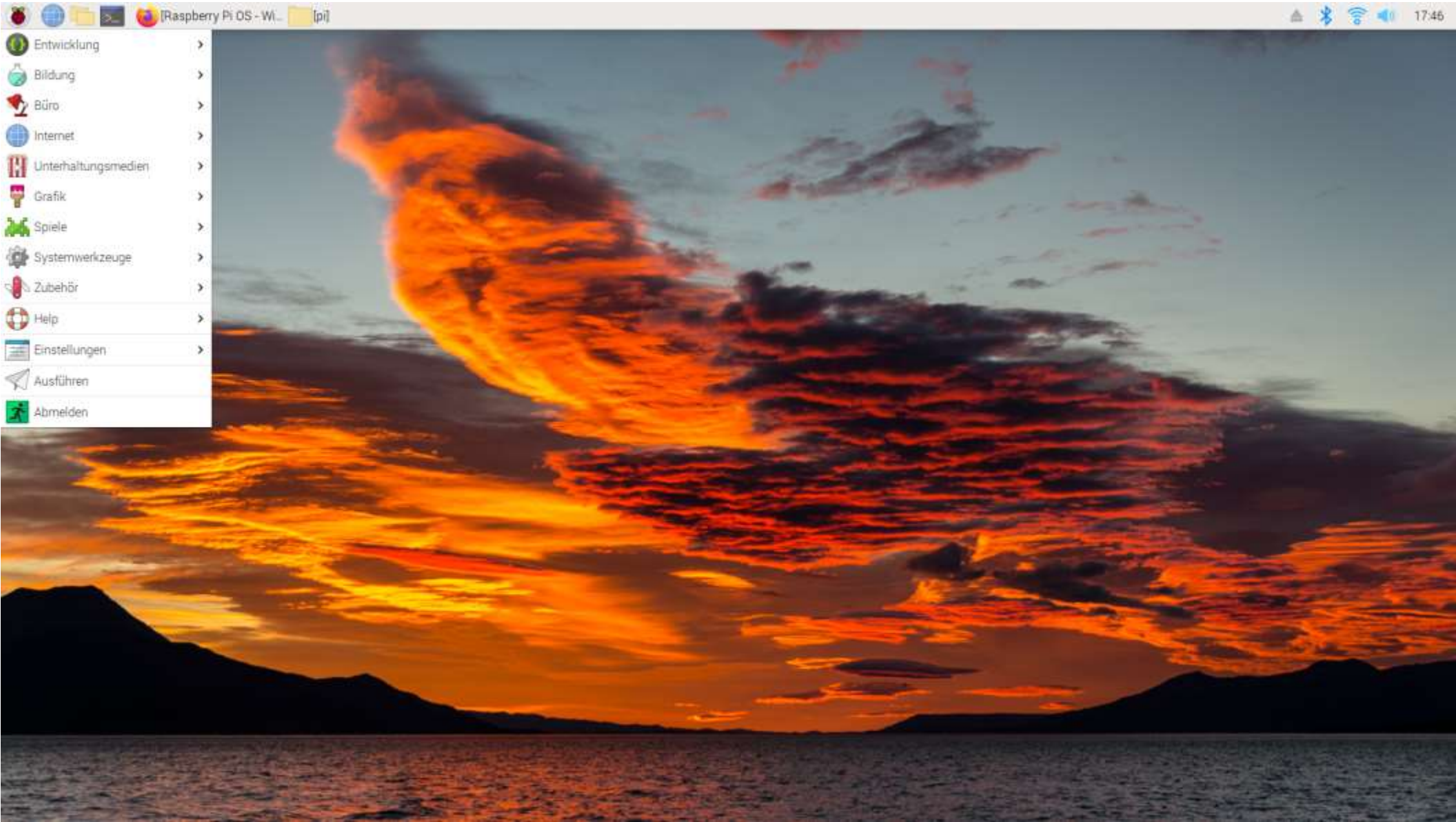


Raspberry Pi OS

- Raspberry Pi is able to run a wide range of software, including a number of different operating systems – the core software that makes a computer run.
- Raspberry Pi OS (previously called Raspbian) is our official supported operating system.
- The Raspberry Pi OS is based on Linux.



Raspberry Pi OS



Install Raspberry Pi OS

- Download **Raspberry Pi OS**.

<https://www.raspberrypi.com/software/operating-systems/>

Raspberry Pi OS

Our recommended operating system for most users.

Compatible with:

All Raspberry Pi models

Raspberry Pi OS with desktop

Release date: February 21st 2023

System: 32-bit

Kernel version: 5.15

Debian version: 11 (bullseye)

Size: 924MB

Show SHA256 file integrity hash:

[Release notes](#)

Download



[Download current](#)

[Archive](#)

Install Raspberry Pi OS

- Download **Pi Imager** to install the OS.
<https://www.raspberrypi.com/software/>

Install Raspberry Pi OS using Raspberry Pi Imager

Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi. [Watch our 45-second video](#) to learn how to install an operating system using Raspberry Pi Imager.

Download and install Raspberry Pi Imager to a computer with an SD card reader. Put the SD card you'll use with your Raspberry Pi into the reader and run Raspberry Pi Imager.

Download for Windows



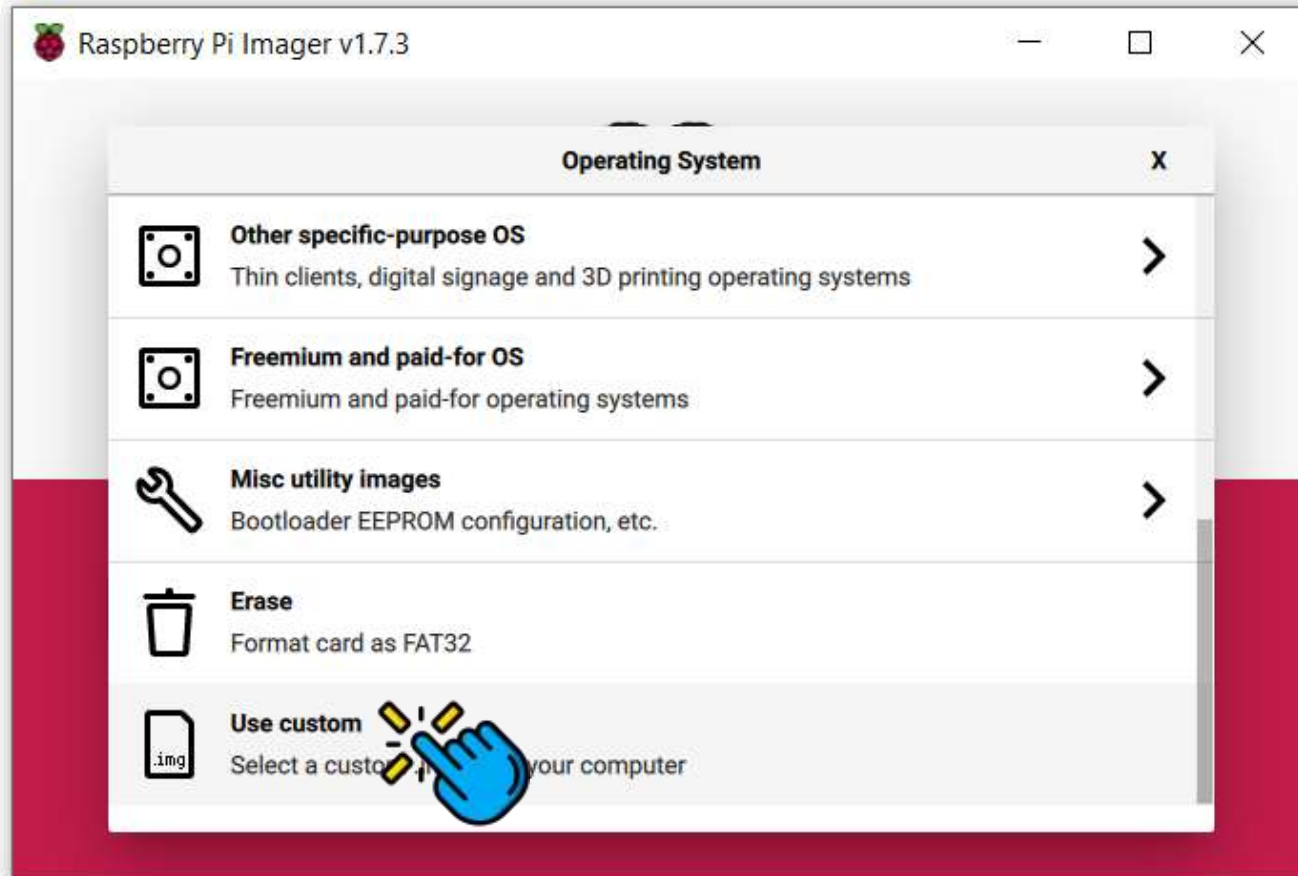
Install Raspberry Pi OS

- Click **Choose OS**.



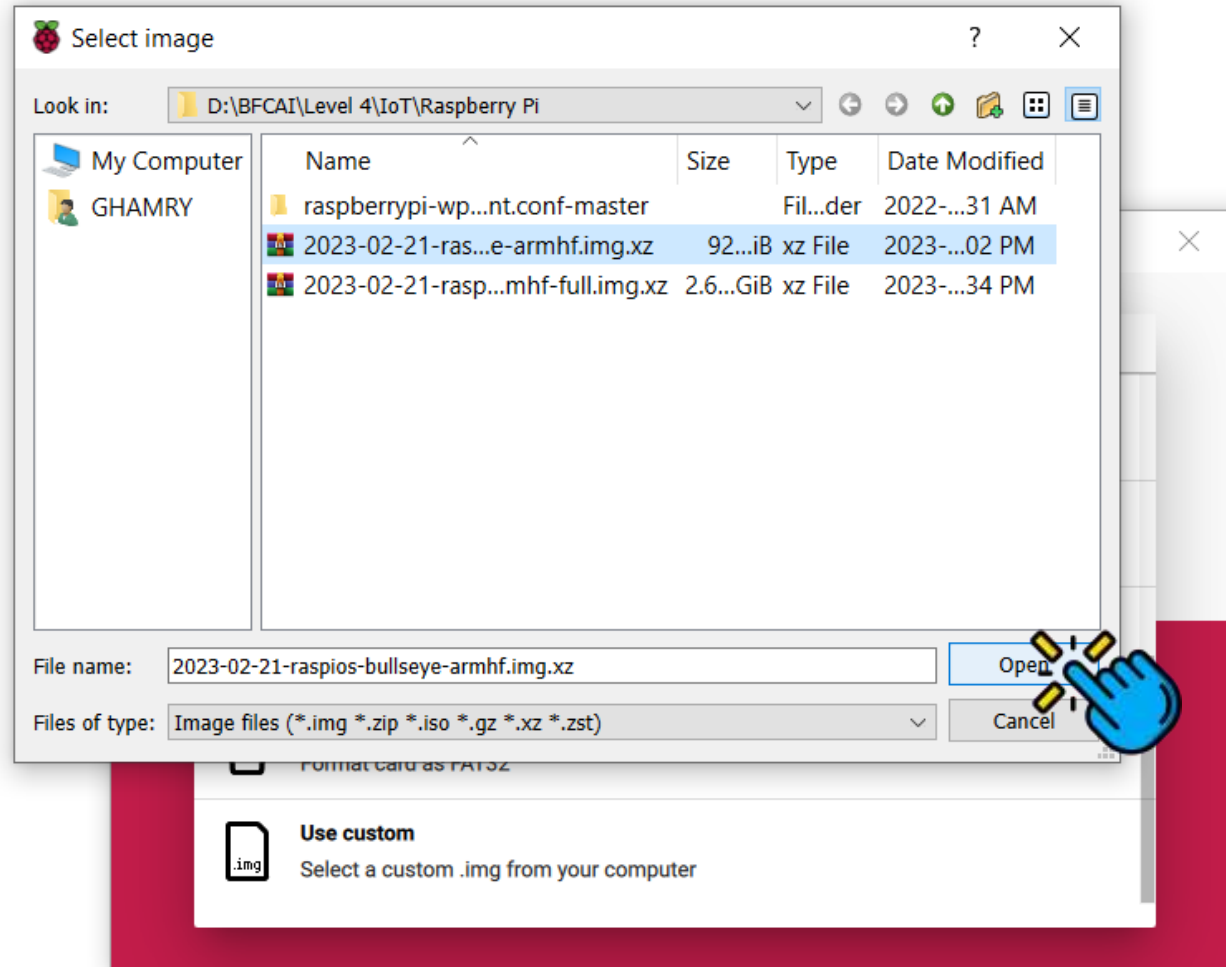
Install Raspberry Pi OS

- Click **Use Custom**.



Install Raspberry Pi OS

- Choose the **Raspberry Pi OS** file, and Click **Open**.



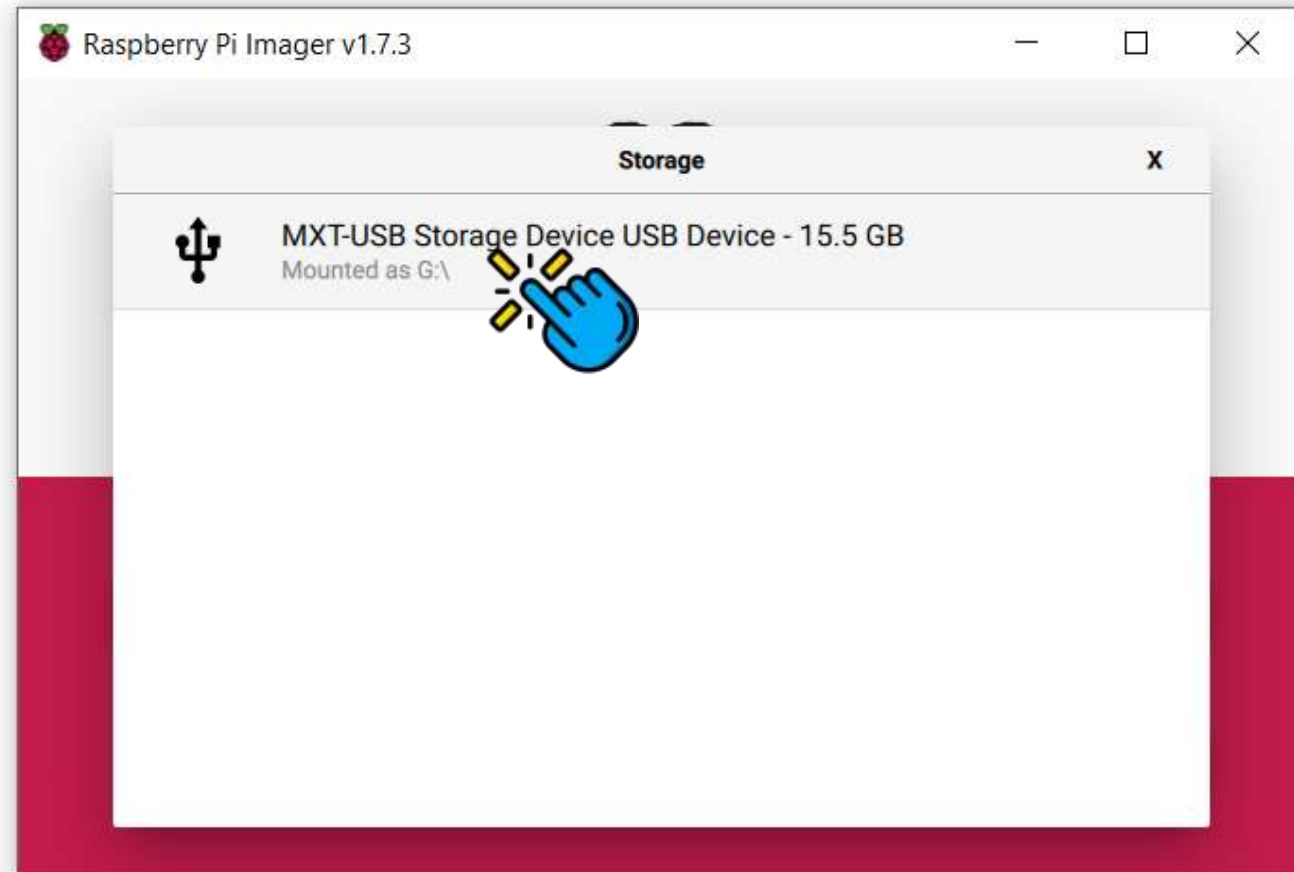
Install Raspberry Pi OS

- Insert the **SD Card** into you computer, and Click **Choose Storage**.



Install Raspberry Pi OS

- Choose your **SD Card**.



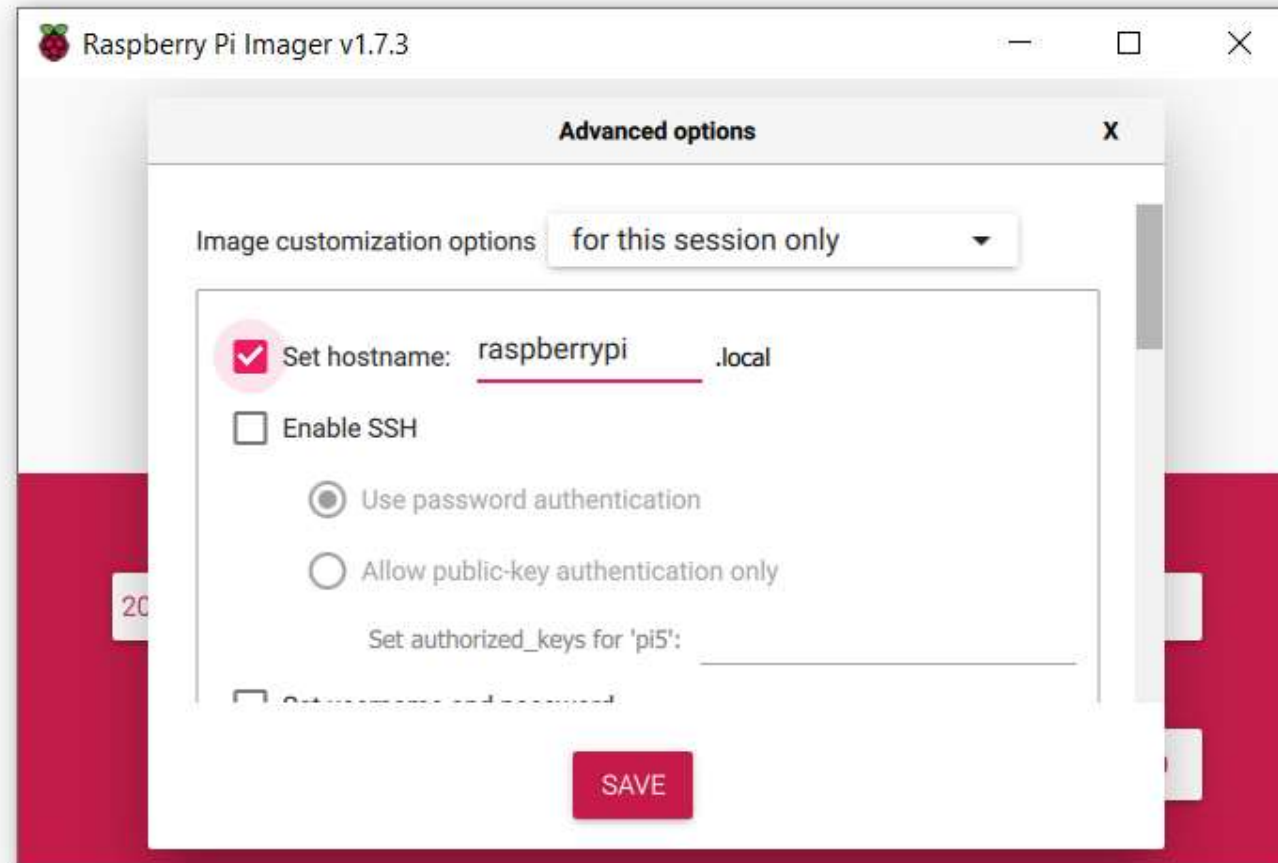
Install Raspberry Pi OS

- Click on **Settings** button.



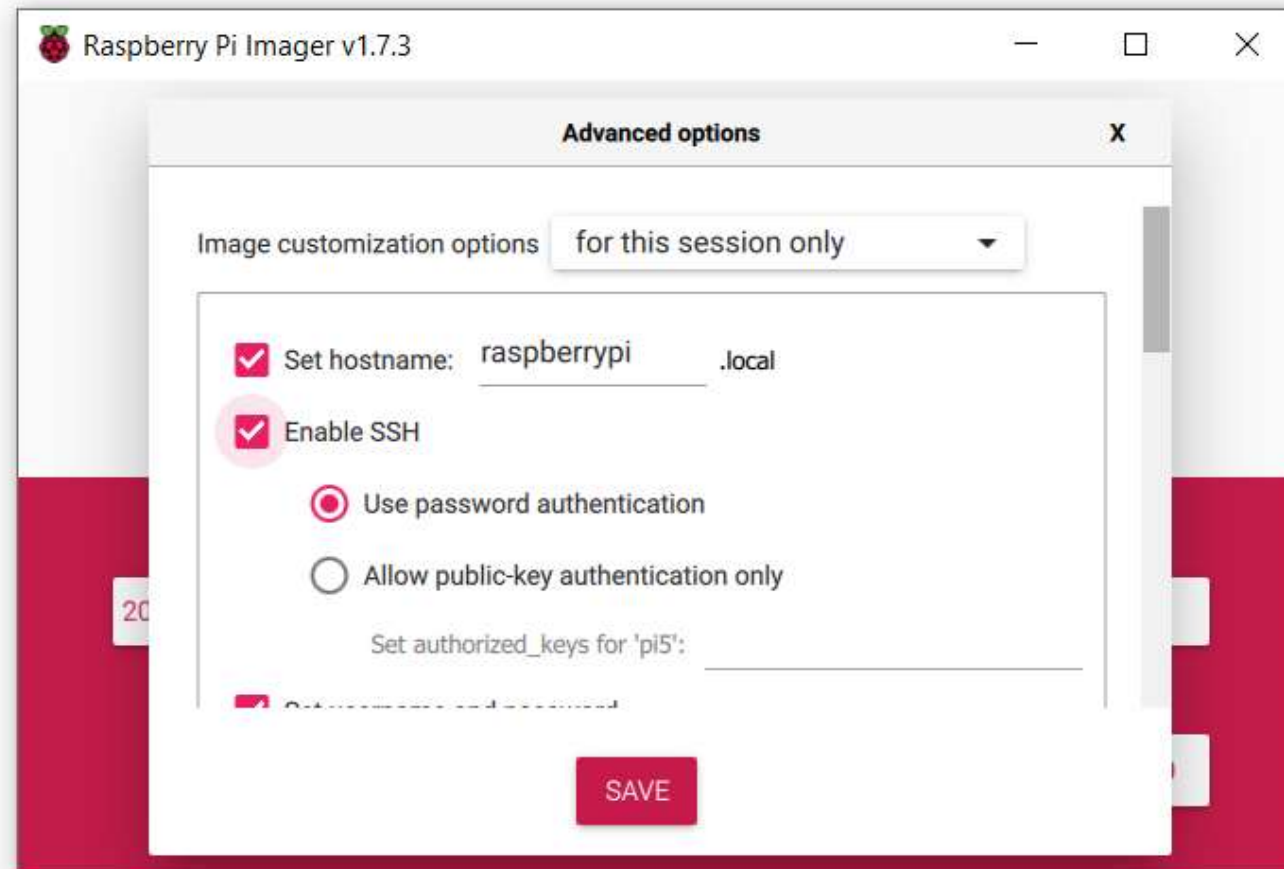
Install Raspberry Pi OS

- Check **Set hostname**.



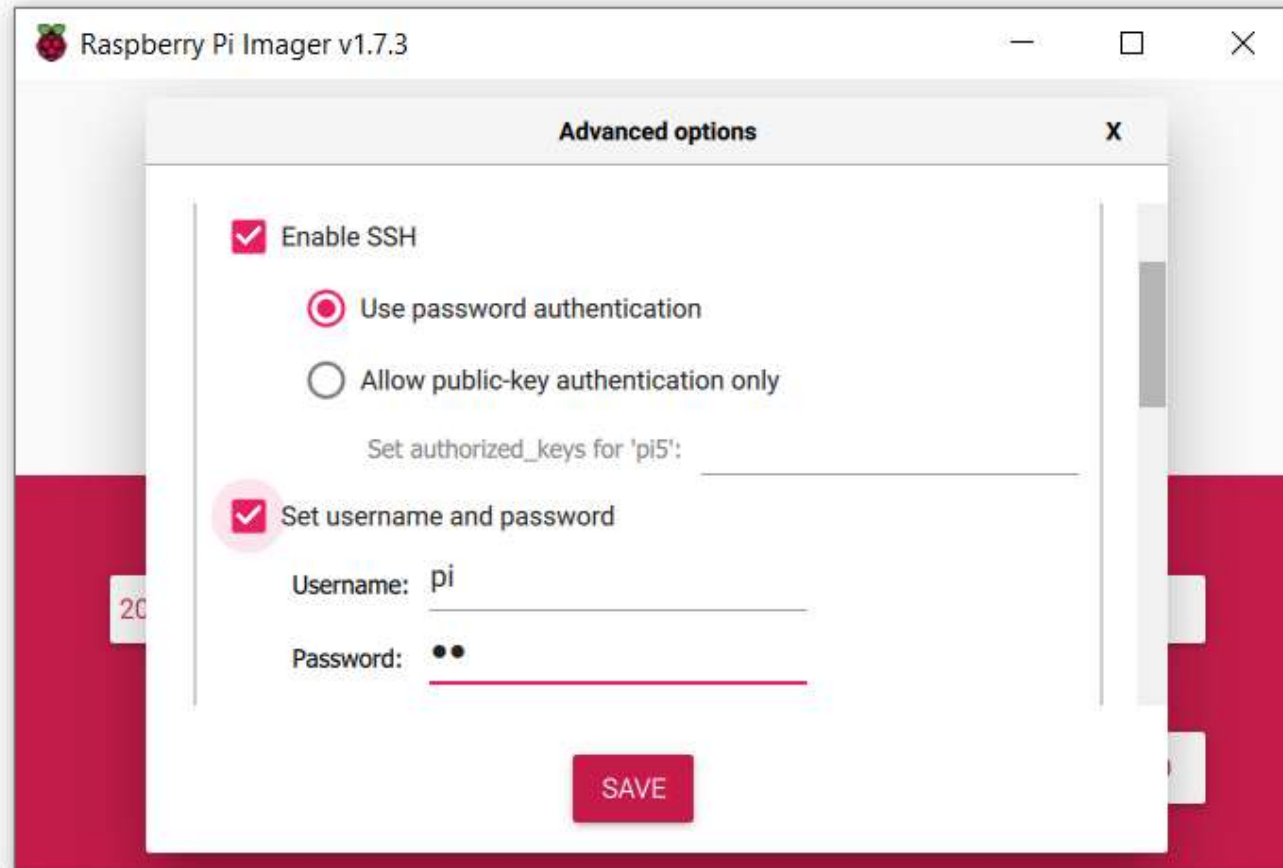
Install Raspberry Pi OS

- Check **Enable SSH**, as we will access the Raspberry Pi remotely.



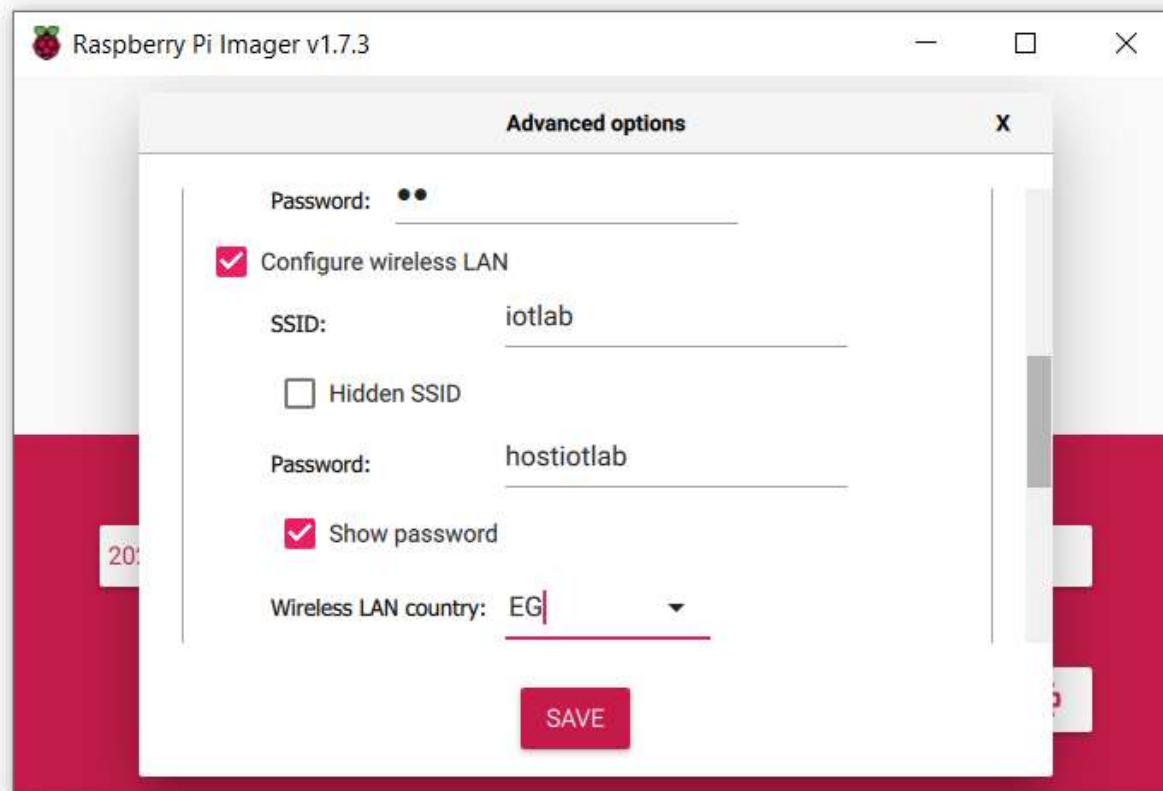
Install Raspberry Pi OS

- Set your **username** and **password**.



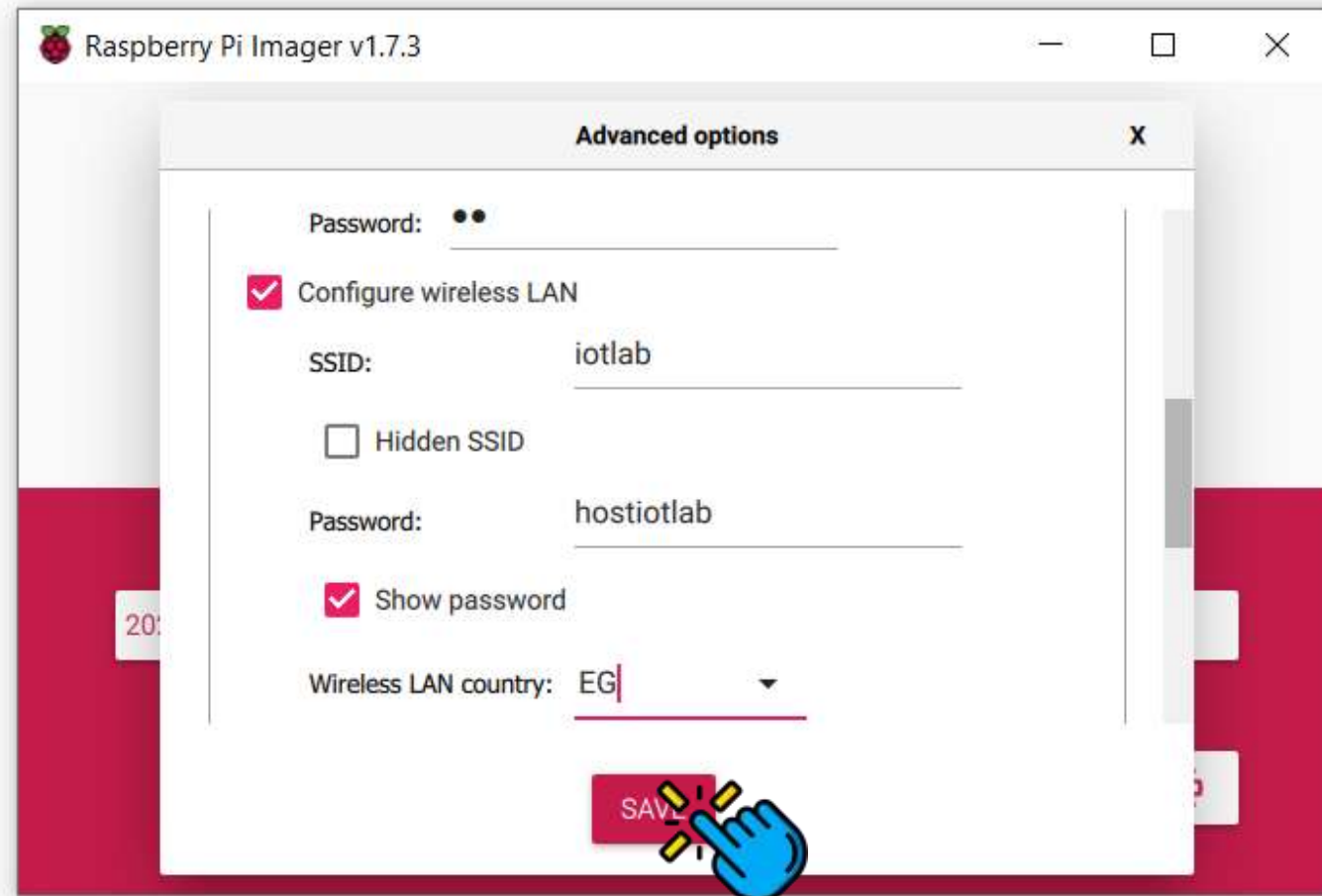
Install Raspberry Pi OS

- We will access the Raspberry Pi using Wi-Fi, so check **Configure Wireless LAN** and choose your **SSID** and **Password**, and choose the country **EG**.
- We will use “**iotlab**” and “**hostiotlab**” as SSID and password.



Install Raspberry Pi OS

- Click **Save** button.



Install Raspberry Pi OS

- Click **Write**.



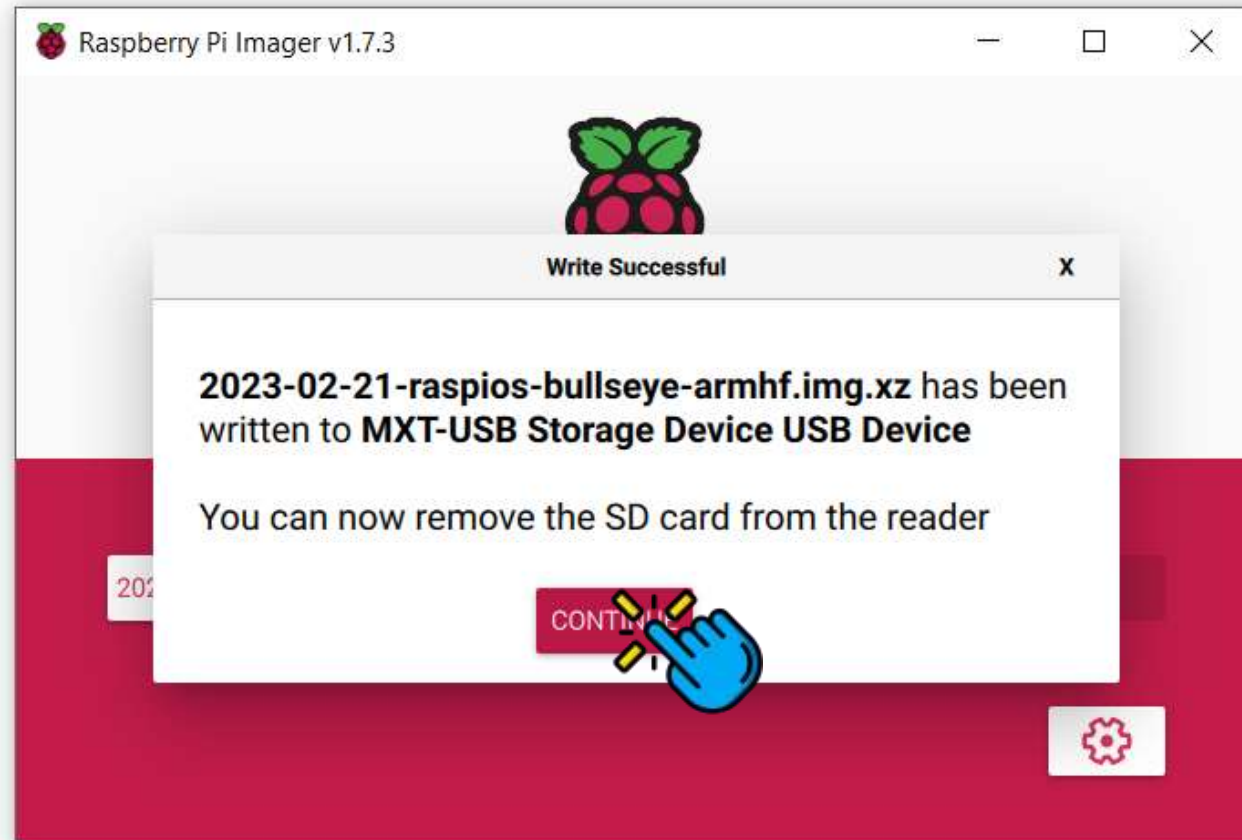
Install Raspberry Pi OS

- Wait until **writing is done**.



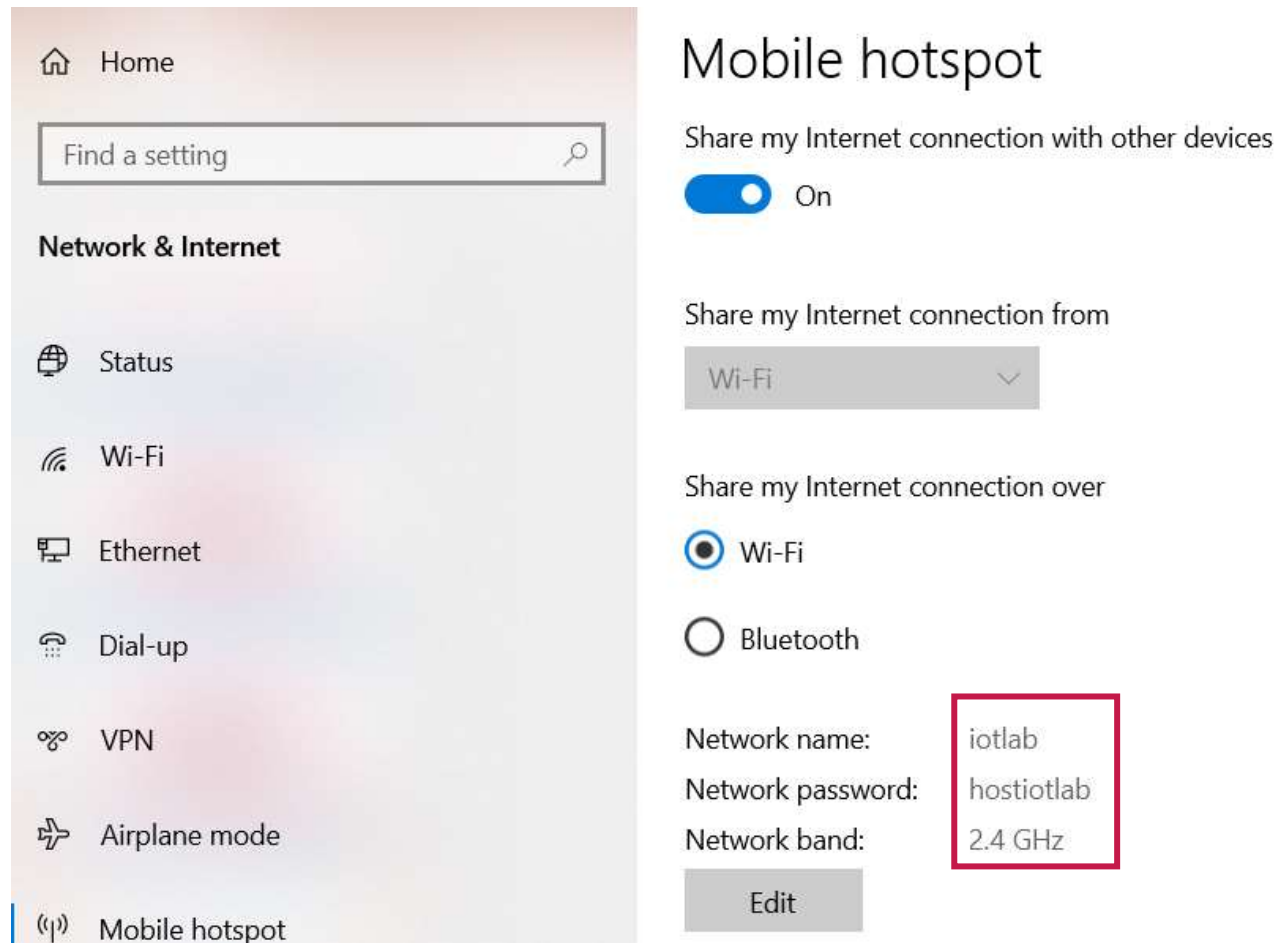
Install Raspberry Pi OS

- After writing is done, the SD Card will be **ejected automatically**, so **reinsert the SD Card** to complete the remaining steps.



Install Raspberry Pi OS

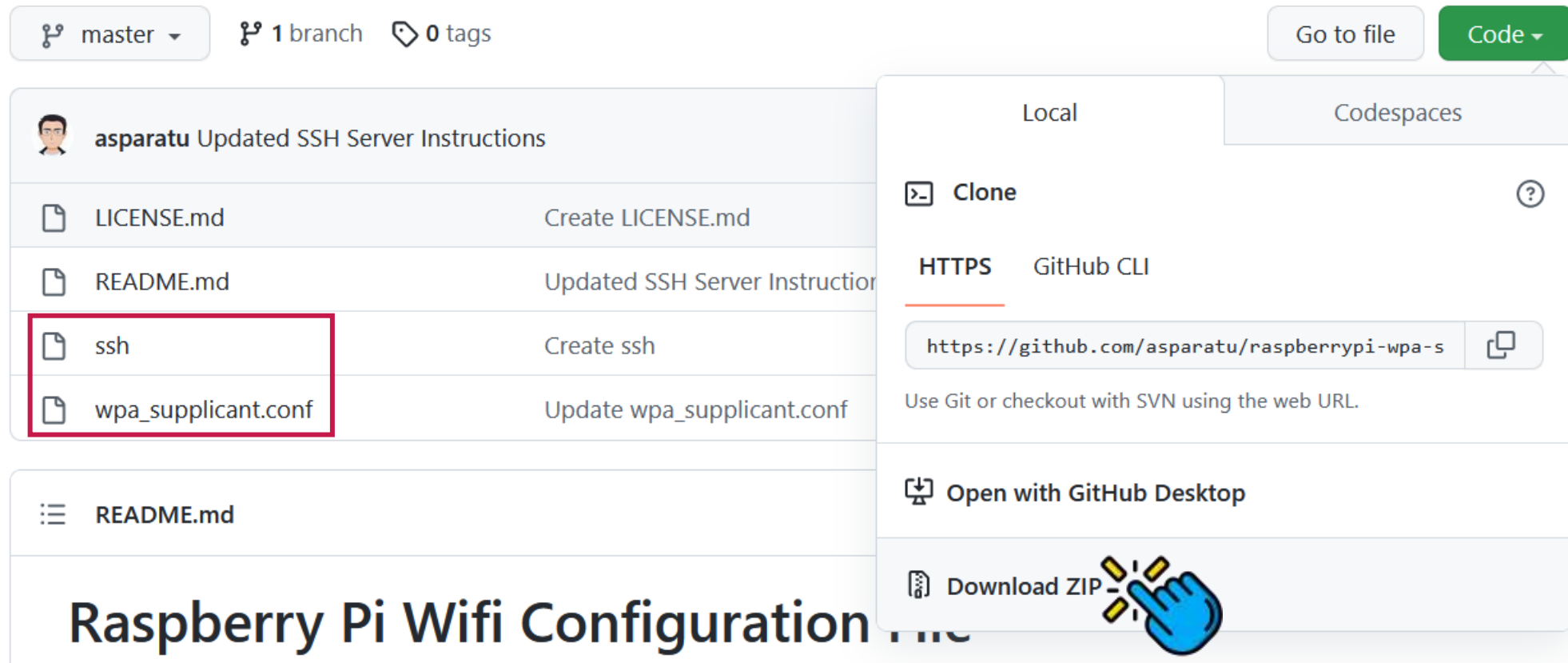
- Open **Mobile hotspot**, and make sure your Wi-Fi info is correct.
- Make sure to **turn on the hotspot**.



Install Raspberry Pi OS

- Download the following files.

<https://github.com/asparatu/raspberrypi-wpa-supPLICANT.conf>



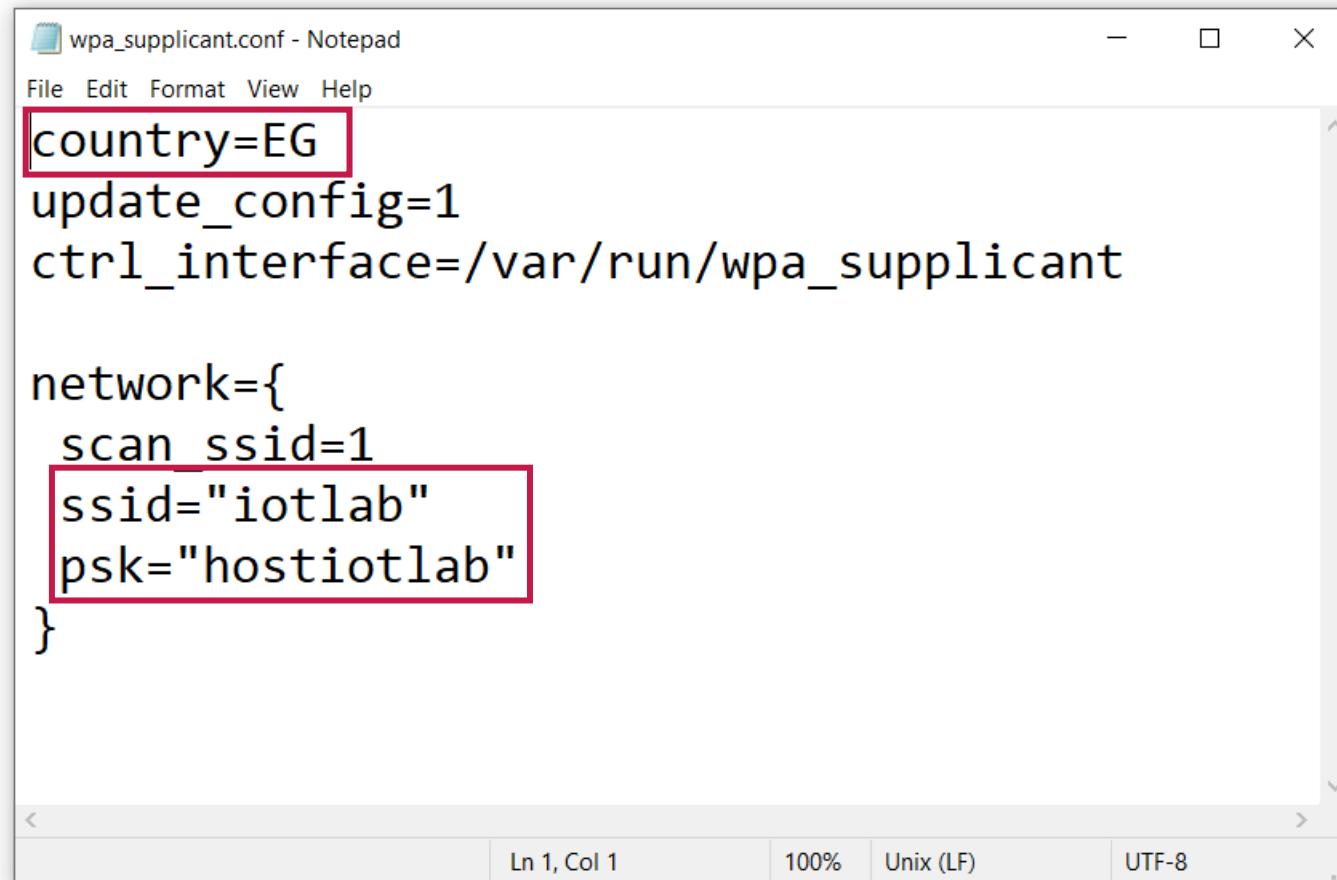
The screenshot shows a GitHub repository page for the user 'asparatu'. The repository name is 'raspberrypi-wpa-supPLICANT.conf'. The current branch is 'master', and there is 1 branch and 0 tags. The file list includes:

File Name	Change
LICENSE.md	Create LICENSE.md
README.md	Updated SSH Server Instruction
ssh	Create ssh
wpa_supPLICANT.conf	Update wpa_supPLICANT.conf

The 'ssh' file is highlighted with a red box. The 'Download ZIP' button is highlighted with a blue hand icon. The repository title is 'Raspberry Pi Wifi Configuration'.

Install Raspberry Pi OS

- Open the file `wpa_supplicant.conf`, and edit it to your **Wi-Fi** info.
- Make sure to save changes.



```
wpa_supplicant.conf - Notepad
File Edit Format View Help
country=EG
update_config=1
ctrl_interface=/var/run/wpa_supplicant

network={
  scan_ssid=1
  ssid="iotlab"
  psk="hostiotlab"
}
```

The screenshot shows a Notepad window titled "wpa_supplicant.conf - Notepad". The menu bar includes "File", "Edit", "Format", "View", and "Help". The text content is as follows:

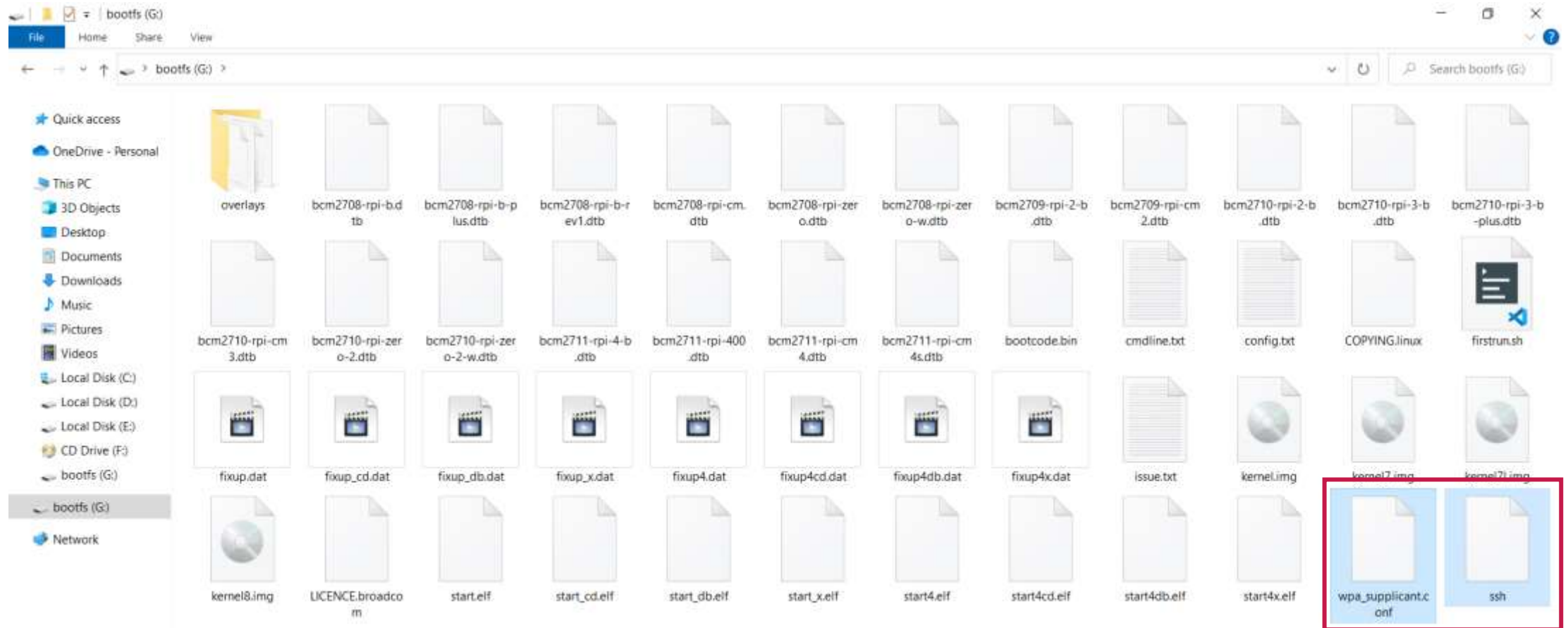
```
country=EG
update_config=1
ctrl_interface=/var/run/wpa_supplicant

network={
  scan_ssid=1
  ssid="iotlab"
  psk="hostiotlab"
}
```

The status bar at the bottom indicates "Ln 1, Col 1", "100%", "Unix (LF)", and "UTF-8".

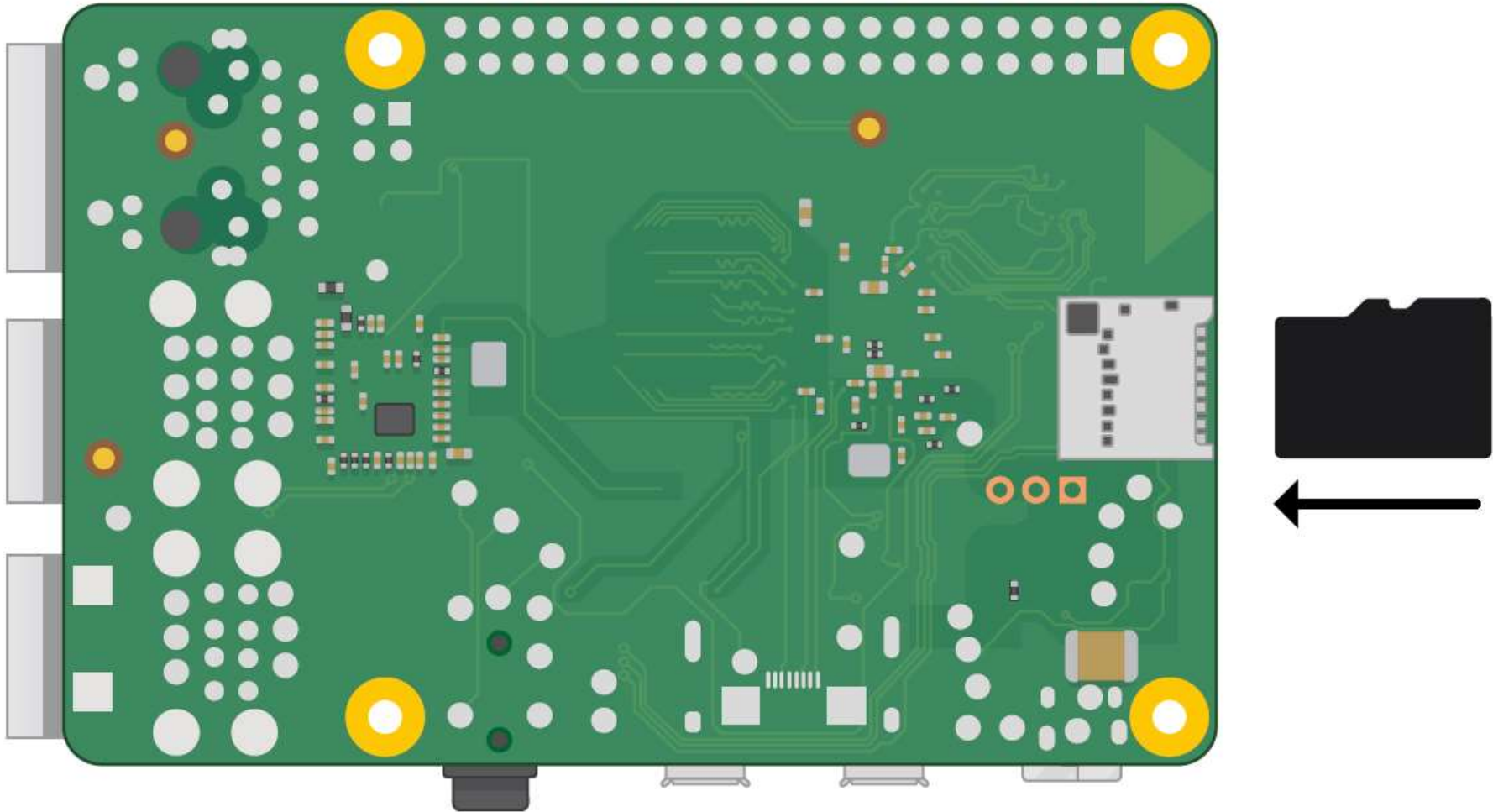
Install Raspberry Pi OS

- Copy the files `wpa_supplicant.conf` and `ssh` to your SD Card.



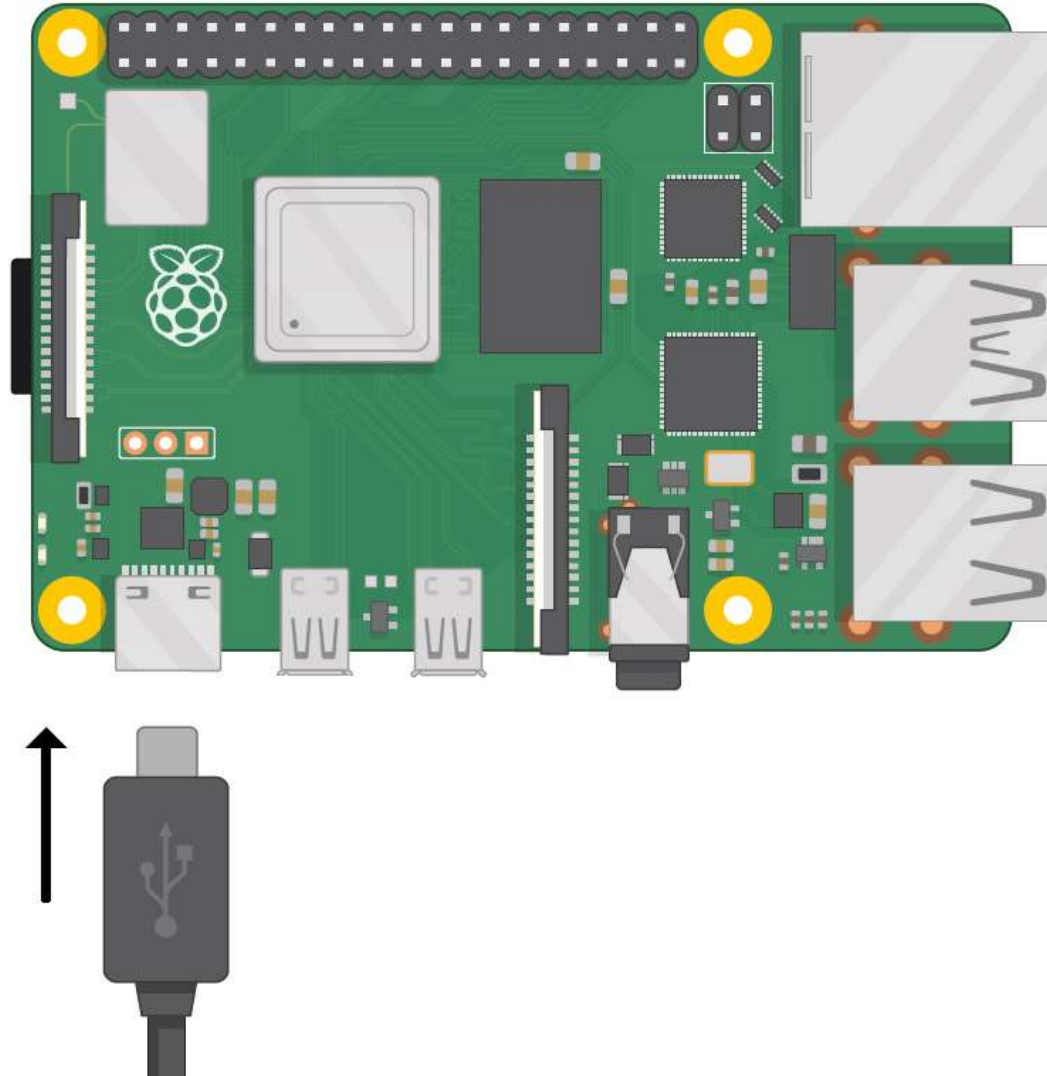
Install Raspberry Pi OS

- Eject the **SD Card**, and **insert it into your Raspberry Pi**.



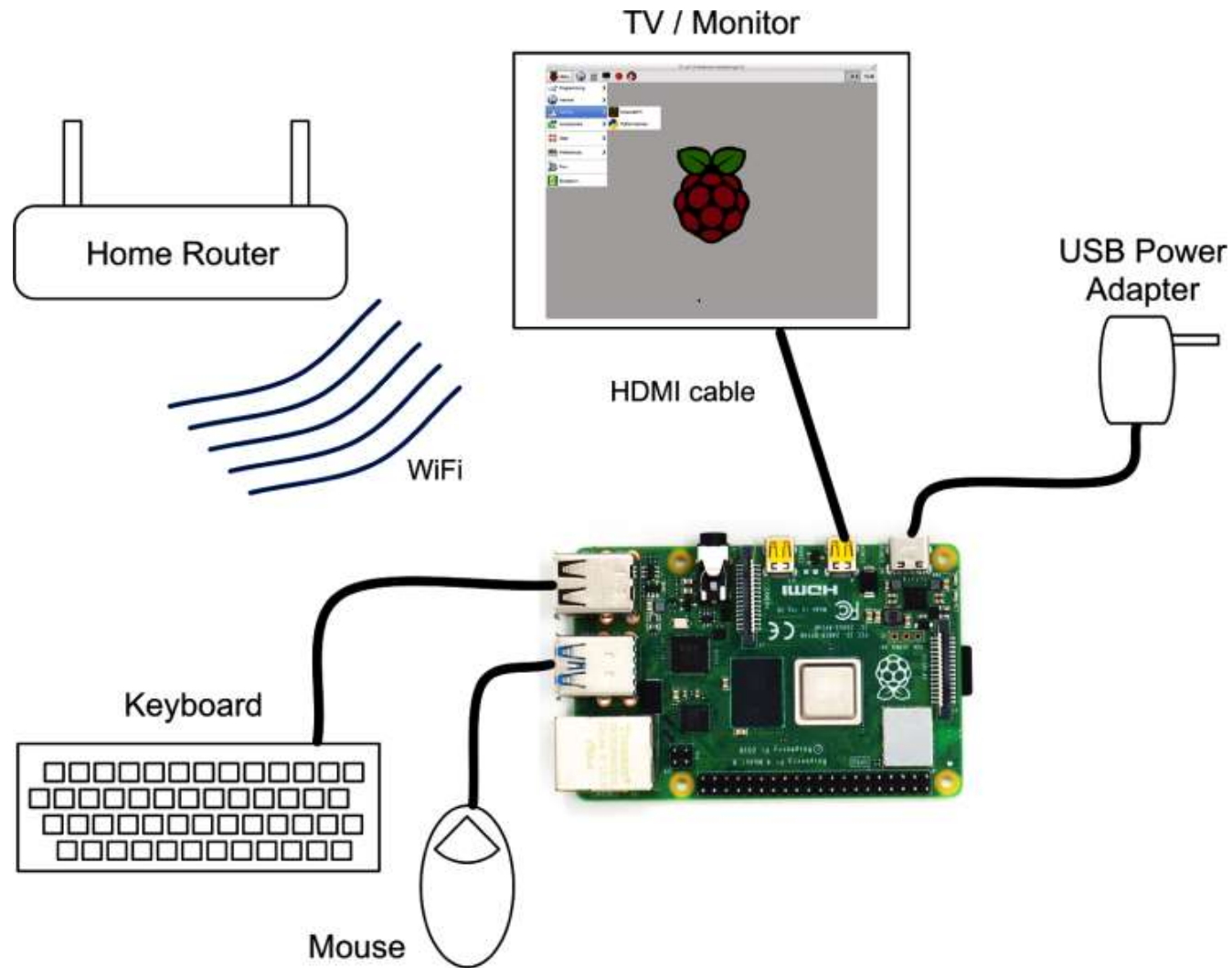
Install Raspberry Pi OS

- Connect your Raspberry Pi to the **power supply**.

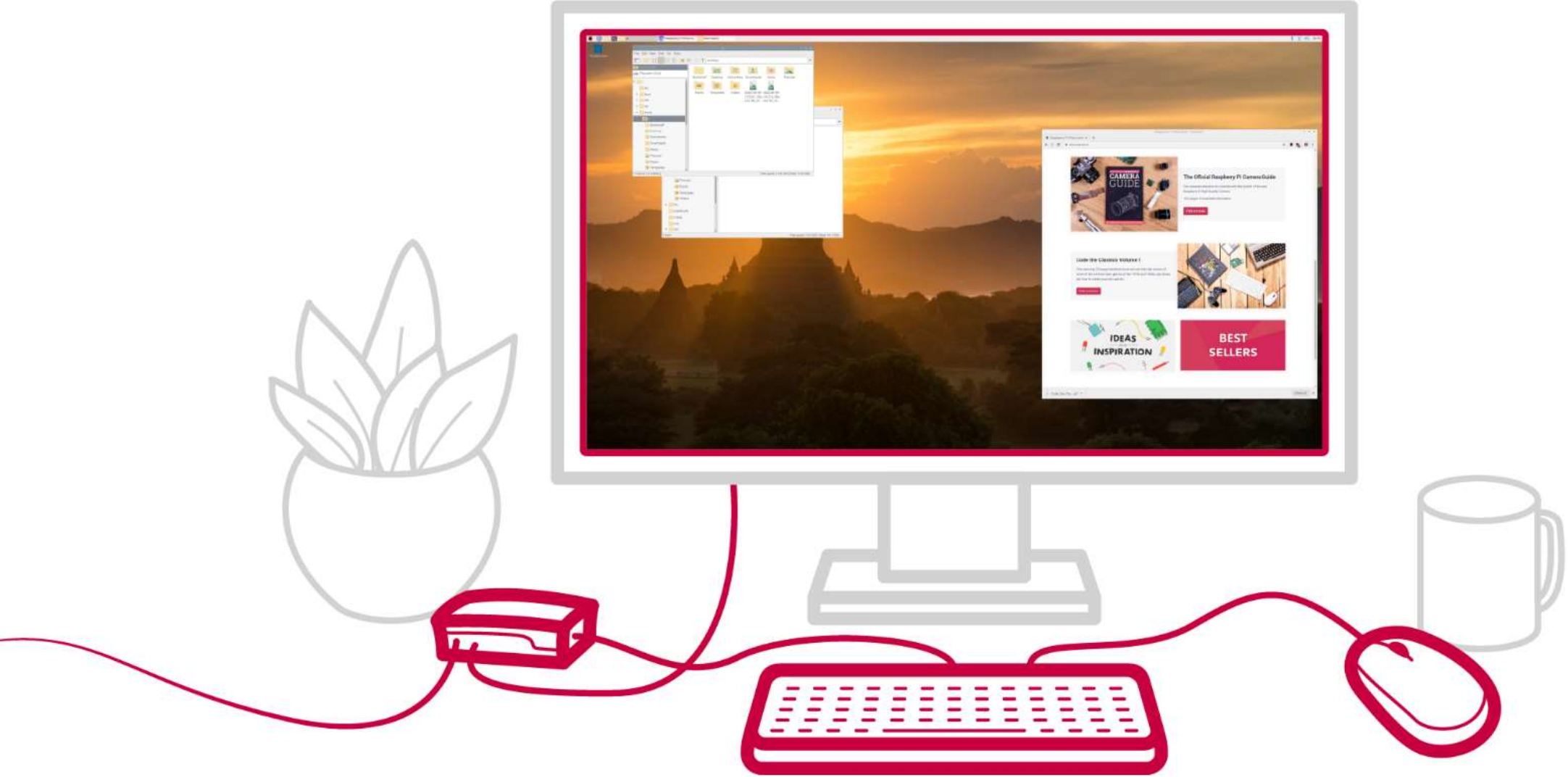


Raspberry Pi as PC

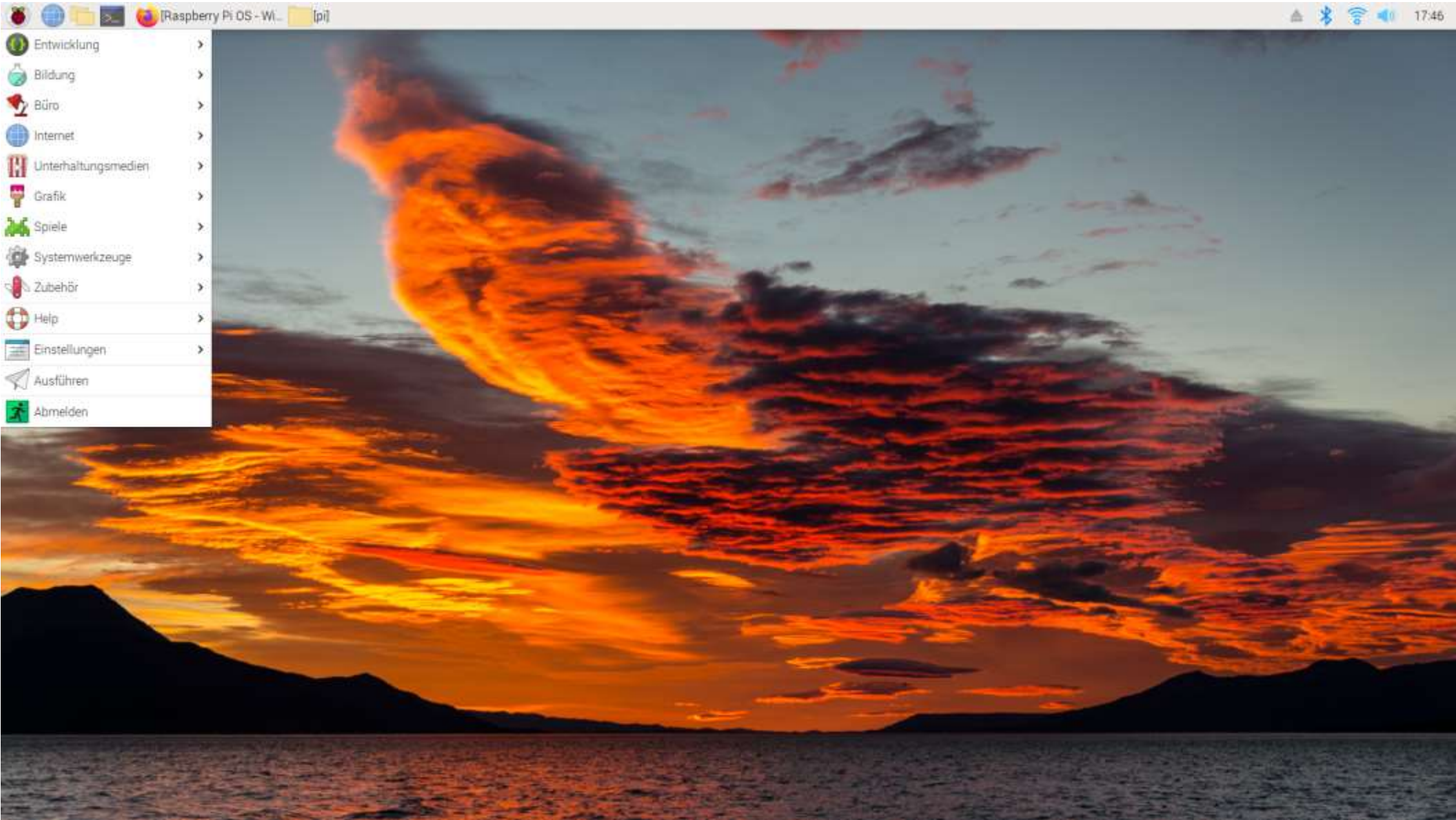
- You can connect mouse, keyboard and monitor as you use your PC.



Raspberry Pi as PC



Raspberry Pi as PC



Accessing Raspberry Pi via SSH

- After connecting your Raspberry Pi to power, it will be connected to your **Wi-Fi** automatically and have an **IP address**.
- Open **Mobile hotspot**, and copy that IP address.

Network name: iotlab

Network password: hostiotlab

Network band: 2.4 GHz

Edit

Devices connected: 1 of 8

Device name	IP address	Physical address (MAC)
-------------	------------	------------------------

raspberrypi	192.168.137.145	b8:27:eb:fd:ac:ef
-------------	-----------------	-------------------

Accessing Raspberry Pi via SSH

- The Raspian OS on the Raspberry Pi allows for **remote login and control via SSH** (Secure Shell).
- Download & install the **PuTTY** application.

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

MSI ('Windows Installer')

64-bit x86: [putty-64bit-0.78-installer.msi](#) (signature)

64-bit Arm: [putty-arm64-0.78-installer.msi](#) (signature)

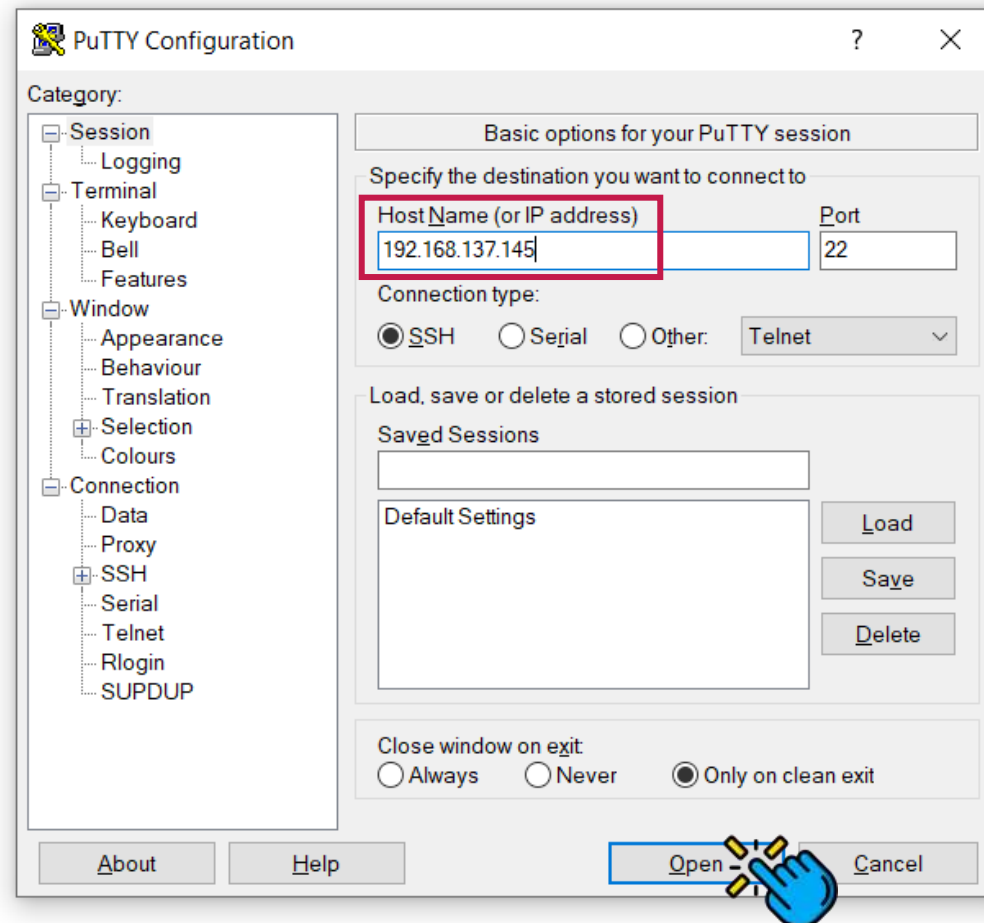
32-bit x86: [putty-0.78-installer.msi](#) (signature)

Unix source archive

.tar.gz: [putty-0.78.tar.gz](#) (signature)

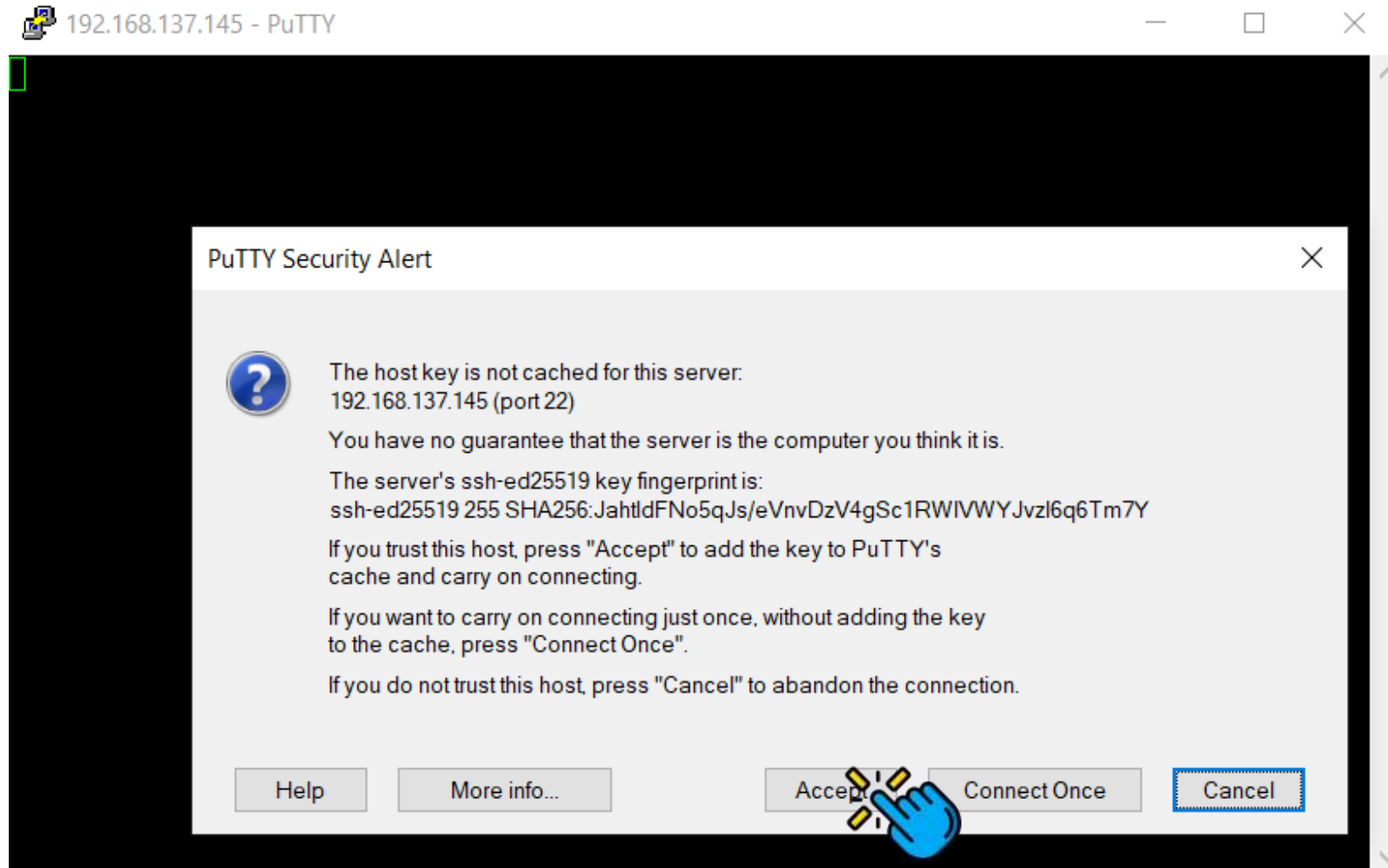
Accessing Raspberry Pi via SSH

- Open **PuTTY**, and enter the Raspberry Pi **IP address** in the **Host Name**.
- Click **Open** button.



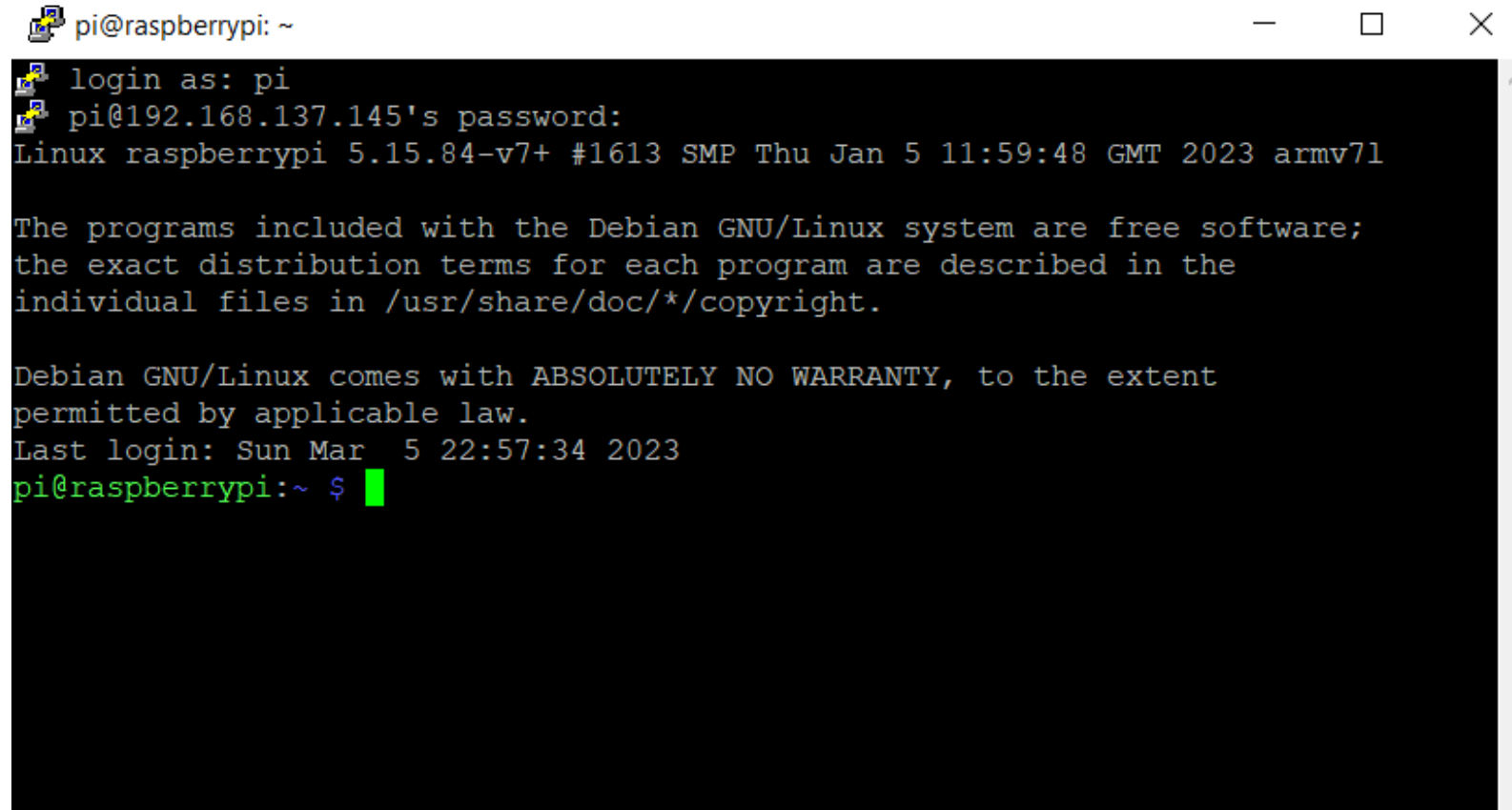
Accessing Raspberry Pi via SSH

- Click **Accept** button.



Accessing Raspberry Pi via SSH

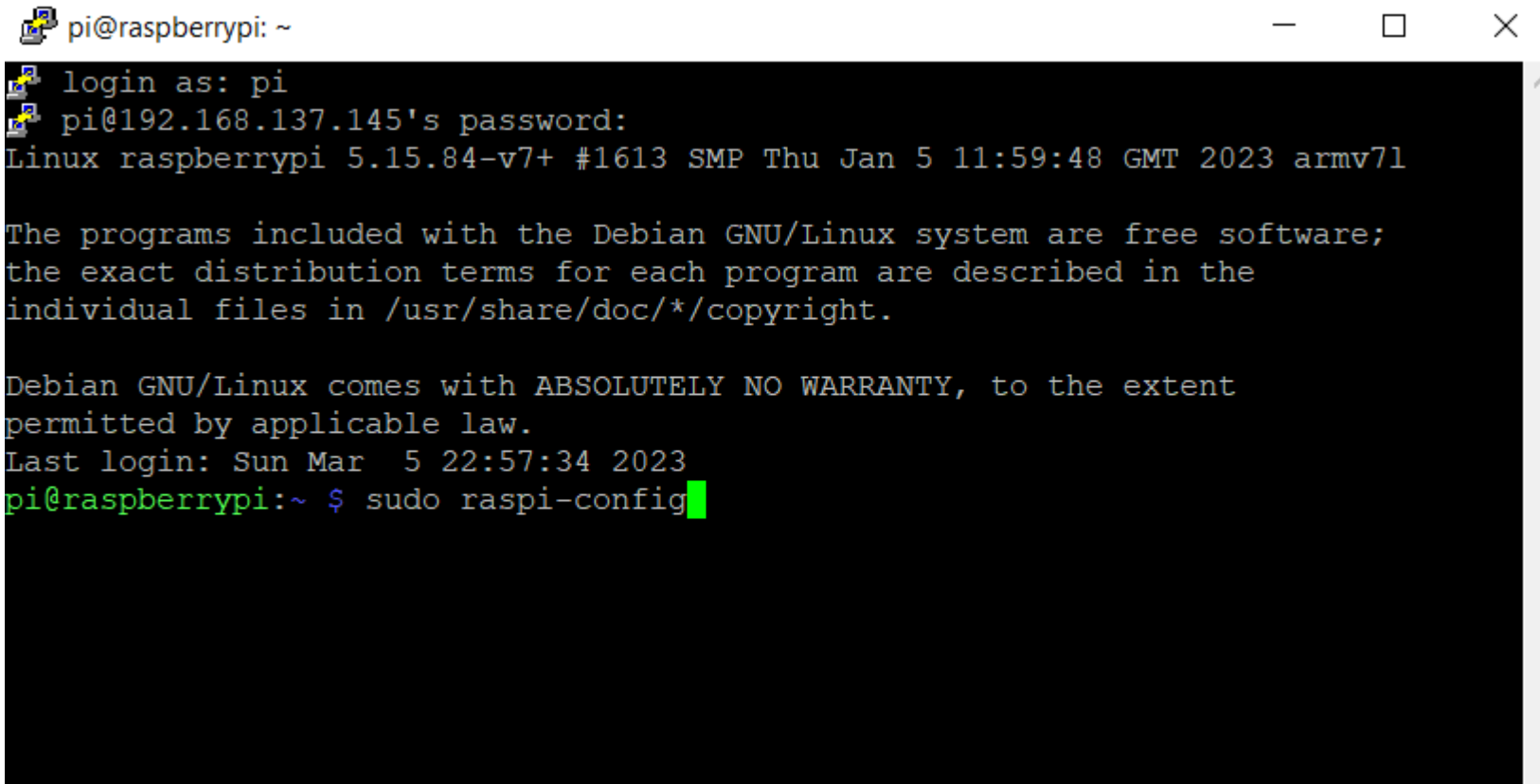
- Enter your **username** and **password**, and click **Enter**.
- Now, you can access your Raspberry Pi via **Secure Shell**.



```
pi@raspberrypi: ~  
login as: pi  
pi@192.168.137.145's password:  
Linux raspberrypi 5.15.84-v7+ #1613 SMP Thu Jan 5 11:59:48 GMT 2023 armv7l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Sun Mar 5 22:57:34 2023  
pi@raspberrypi:~ $
```

Accessing Raspberry Pi via SSH

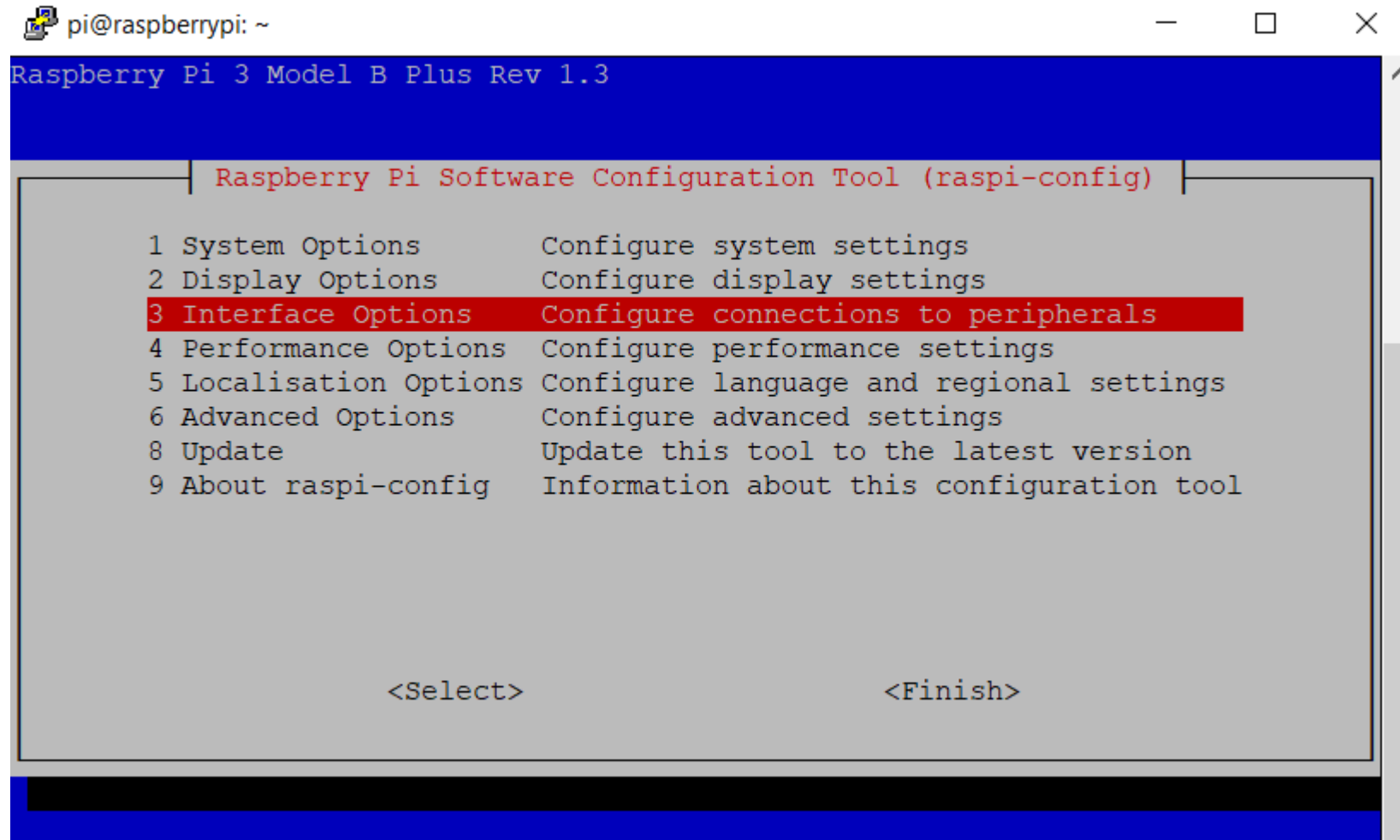
- To access your Raspberry Pi configurations, Enter the following command
`sudo raspi-config`

A terminal window titled 'pi@raspberrypi: ~' with standard window controls. The terminal shows a successful SSH login for the 'pi' user. The output includes the system version 'Linux raspberrypi 5.15.84-v7+ #1613 SMP Thu Jan 5 11:59:48 GMT 2023 armv7l', a copyright notice for Debian GNU/Linux, and the last login time 'Sun Mar 5 22:57:34 2023'. The prompt is now 'pi@raspberrypi:~' and the command 'sudo raspi-config' has been entered, with a green cursor at the end.

```
pi@raspberrypi: ~  
login as: pi  
pi@192.168.137.145's password:  
Linux raspberrypi 5.15.84-v7+ #1613 SMP Thu Jan 5 11:59:48 GMT 2023 armv7l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Sun Mar 5 22:57:34 2023  
pi@raspberrypi:~ $ sudo raspi-config
```

Accessing Raspberry Pi via SSH

- For example, to enable/disable **Interface Options**, select it and click Enter.

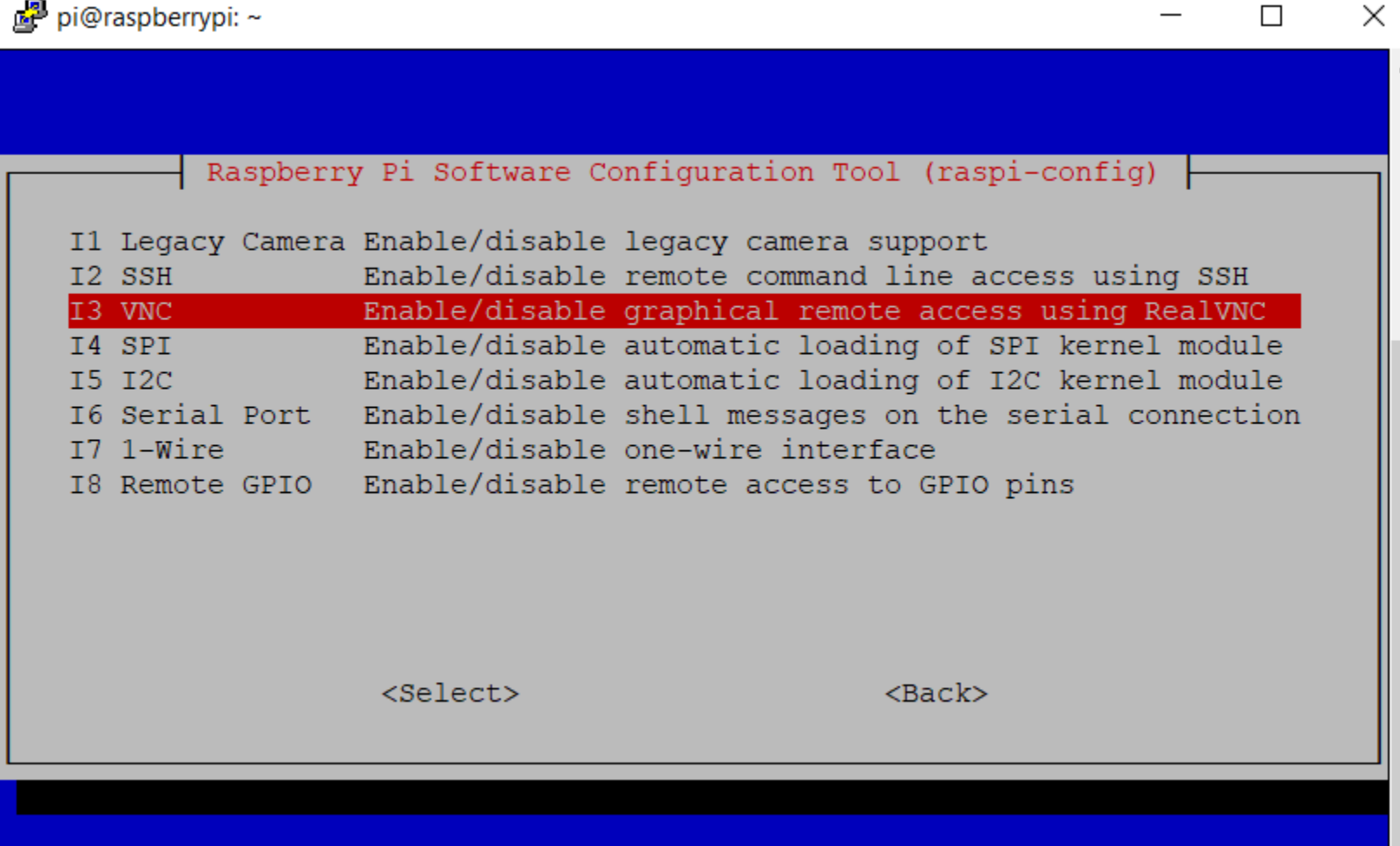


```
pi@raspberrypi: ~
Raspberry Pi 3 Model B Plus Rev 1.3
Raspberry Pi Software Configuration Tool (raspi-config)
1 System Options          Configure system settings
2 Display Options        Configure display settings
3 Interface Options      Configure connections to peripherals
4 Performance Options    Configure performance settings
5 Localisation Options   Configure language and regional settings
6 Advanced Options       Configure advanced settings
8 Update                 Update this tool to the latest version
9 About raspi-config     Information about this configuration tool

<Select>                <Finish>
```


Accessing Raspberry Pi via SSH

- Select **VNC**, and click **Enter**.



```
pi@raspberrypi: ~  
Raspberry Pi Software Configuration Tool (raspi-config)  
I1 Legacy Camera Enable/disable legacy camera support  
I2 SSH Enable/disable remote command line access using SSH  
I3 VNC Enable/disable graphical remote access using RealVNC  
I4 SPI Enable/disable automatic loading of SPI kernel module  
I5 I2C Enable/disable automatic loading of I2C kernel module  
I6 Serial Port Enable/disable shell messages on the serial connection  
I7 1-Wire Enable/disable one-wire interface  
I8 Remote GPIO Enable/disable remote access to GPIO pins  
  
<Select> <Back>
```

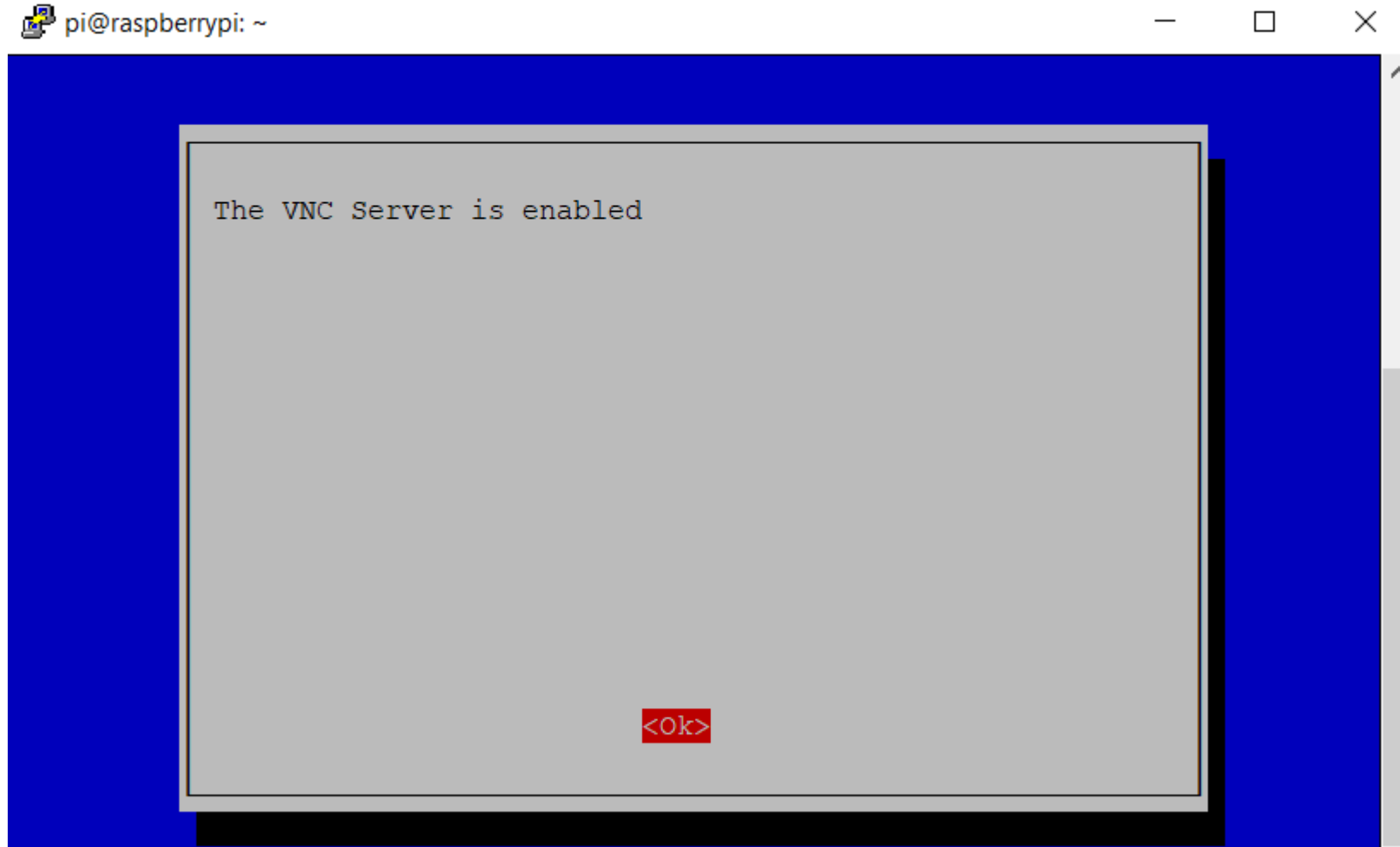
Accessing Raspberry Pi via SSH

- Select **Yes**, and click **Enter**.



Accessing Raspberry Pi via SSH

- Click **Ok**.



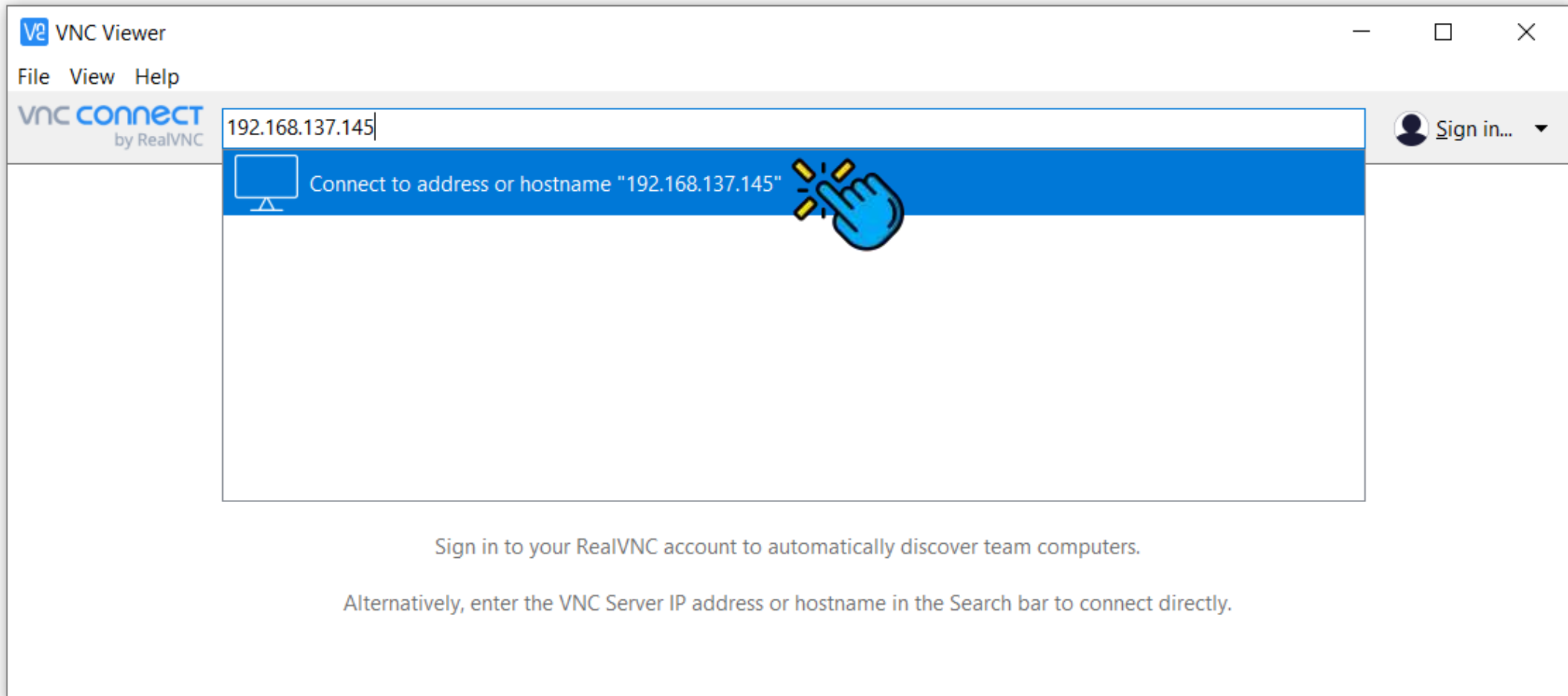
Accessing Raspberry Pi via VNC

- The Raspberry Pi can be controlled like any other desktop computer using a keyboard, mouse, and monitor.
- The VNC (Virtual Network Computing) allows you to remotely control the desktop interface of the Raspberry Pi from another computer or mobile device without the need for a monitor.
- The SSH only provides terminal access.



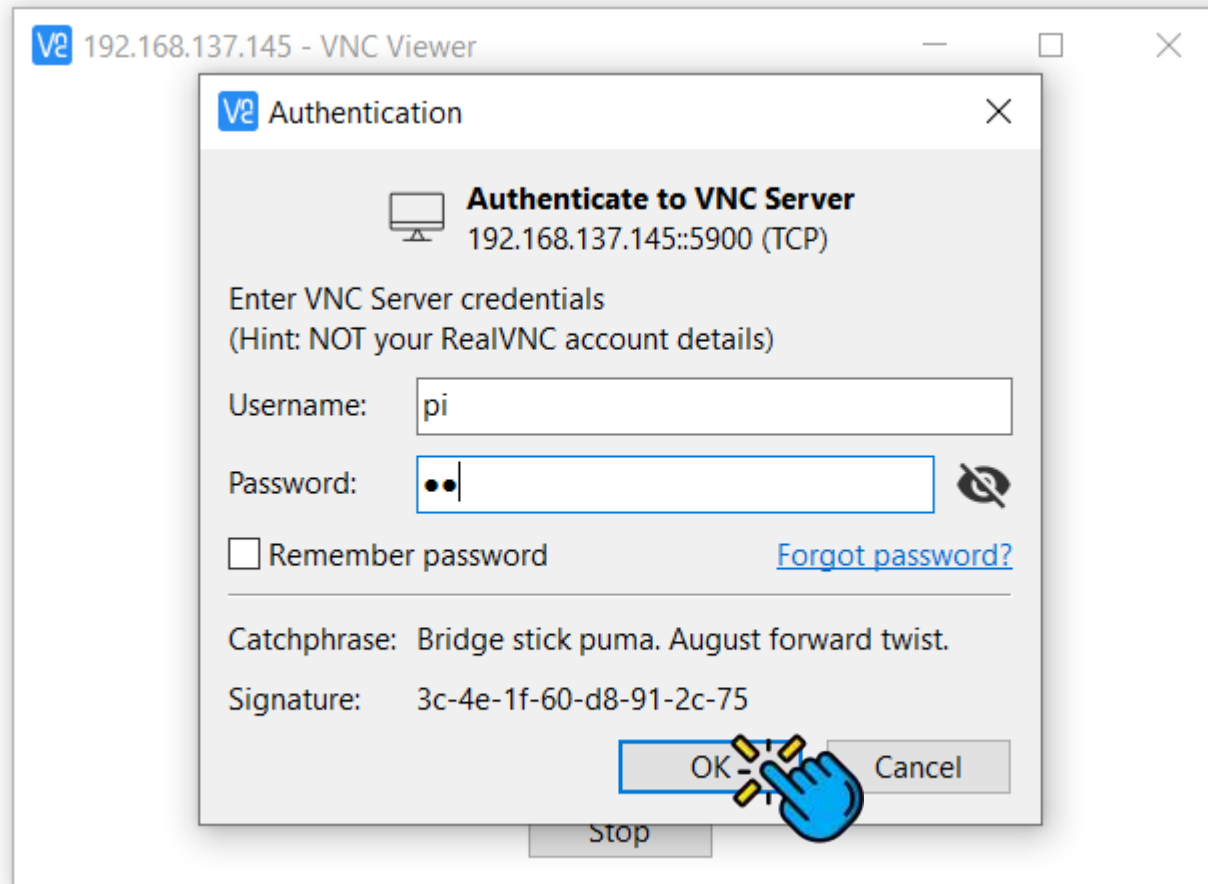
Accessing Raspberry Pi via VNC

- Download and Open **VNC Viewer**, and Enter the **IP address** of your Pi.



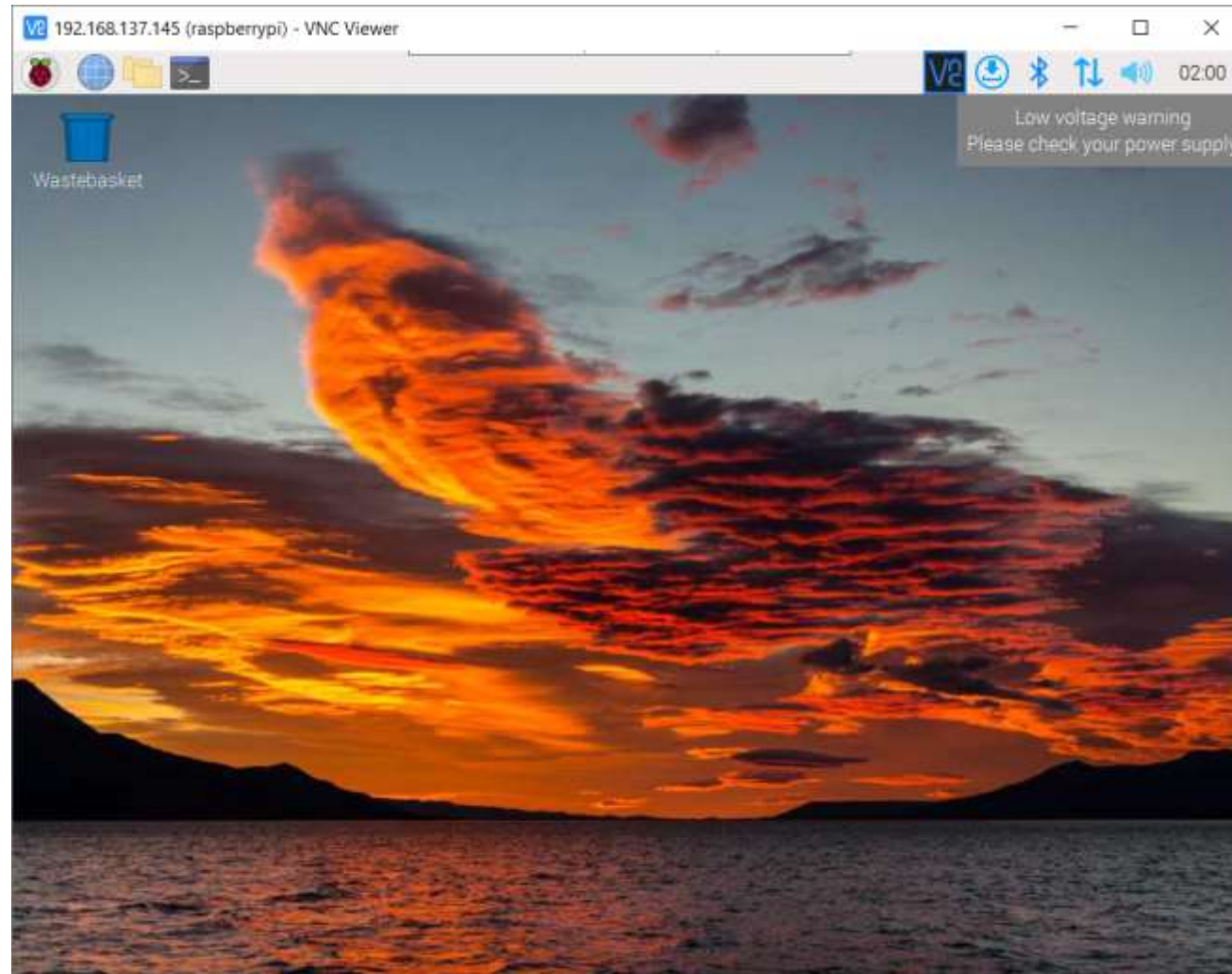
Accessing Raspberry Pi via VNC

- Enter your **username** and **password**, and click **Ok**.



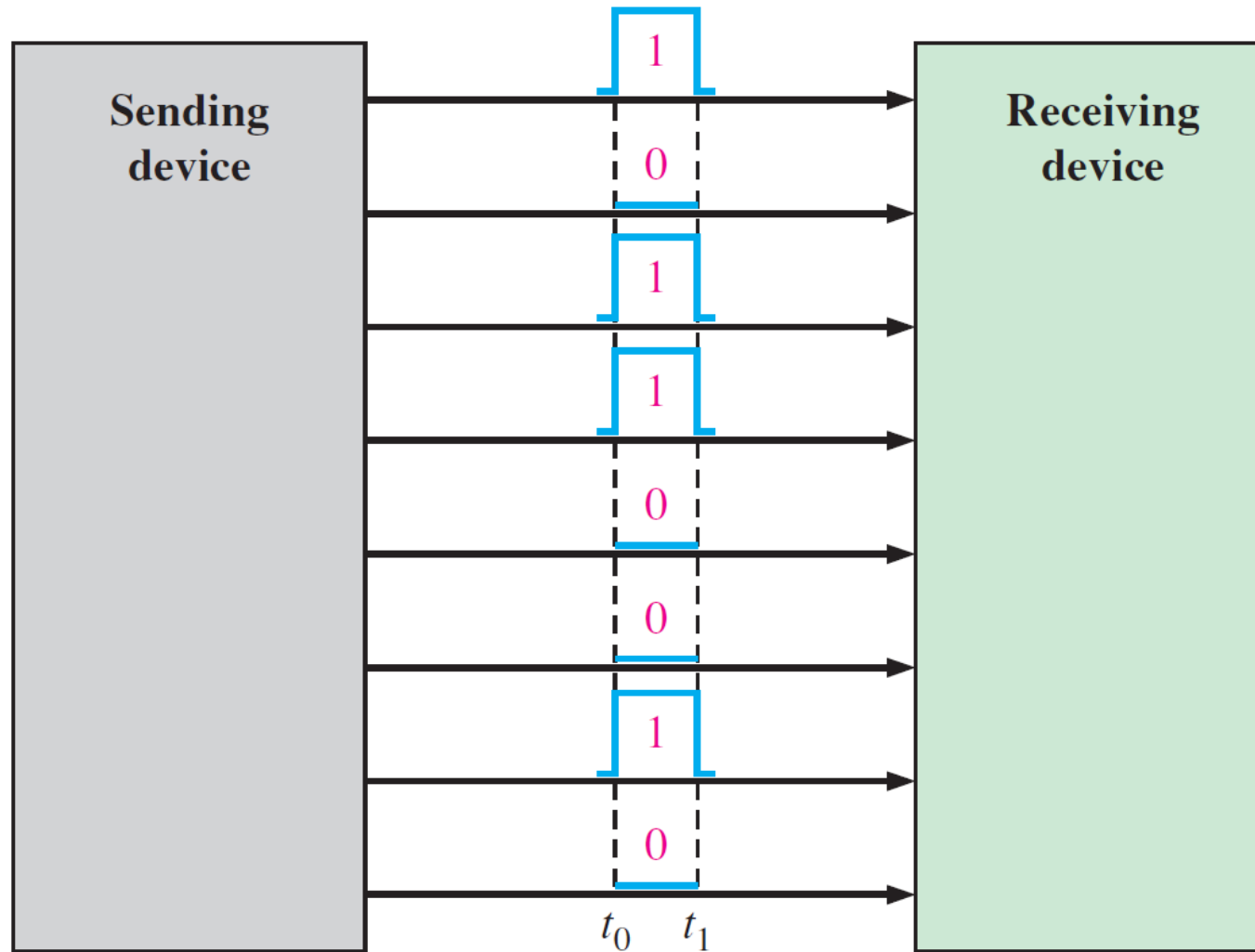
Accessing Raspberry Pi via VNC

- Now, you can **graphically** access your Raspberry Pi from anywhere.



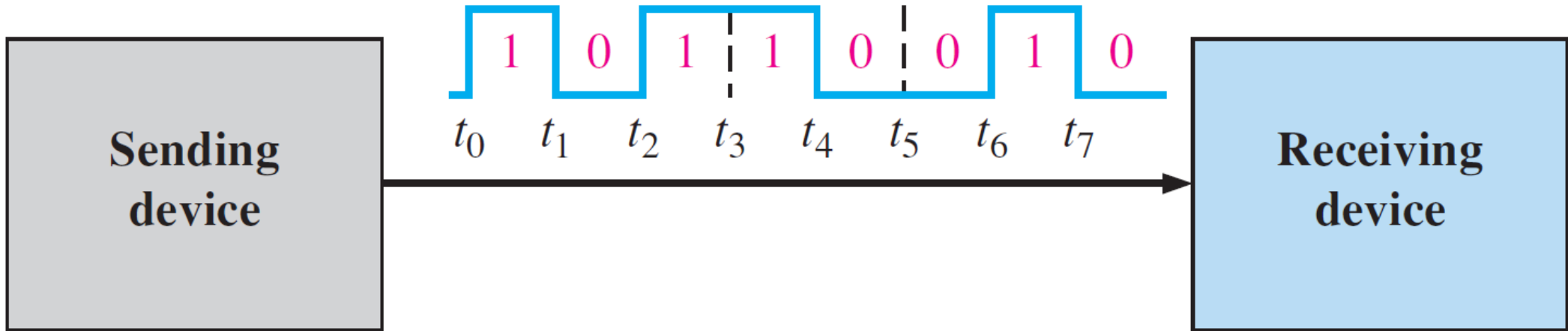
Parallel Communication

- In **parallel communication**, where many bits are sent at the same time.



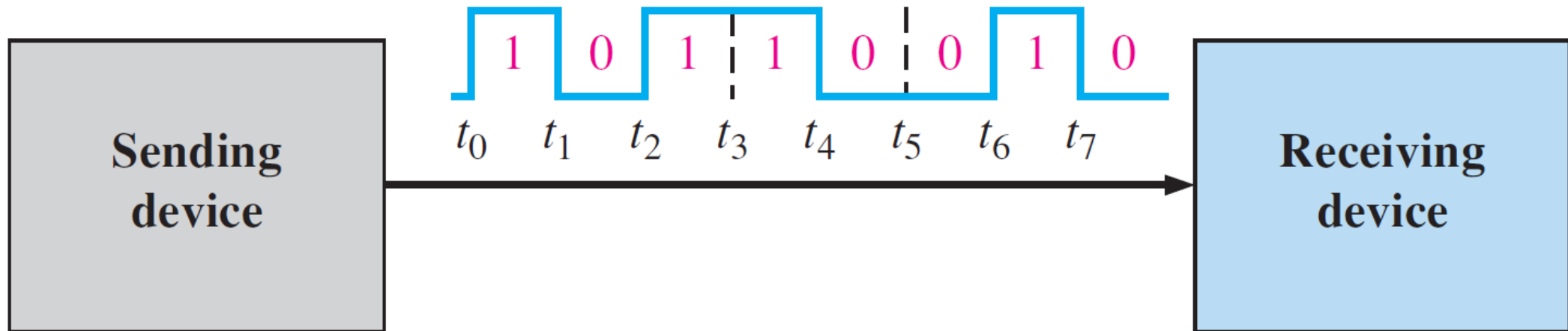
Serial Communication

- Serial communication is simply **a way to transfer data**.
- The data will be sent **sequentially**, one bit at a time.



Serial Communication: UART Protocol

- UART means “Universal Asynchronous Receiver Transmitter”.
- When you use **serial communication** between **Arduino** and **Raspberry Pi**, you’re using the **UART protocol**.
- The **UART protocol** allows you to **communicate** between the 2 boards.

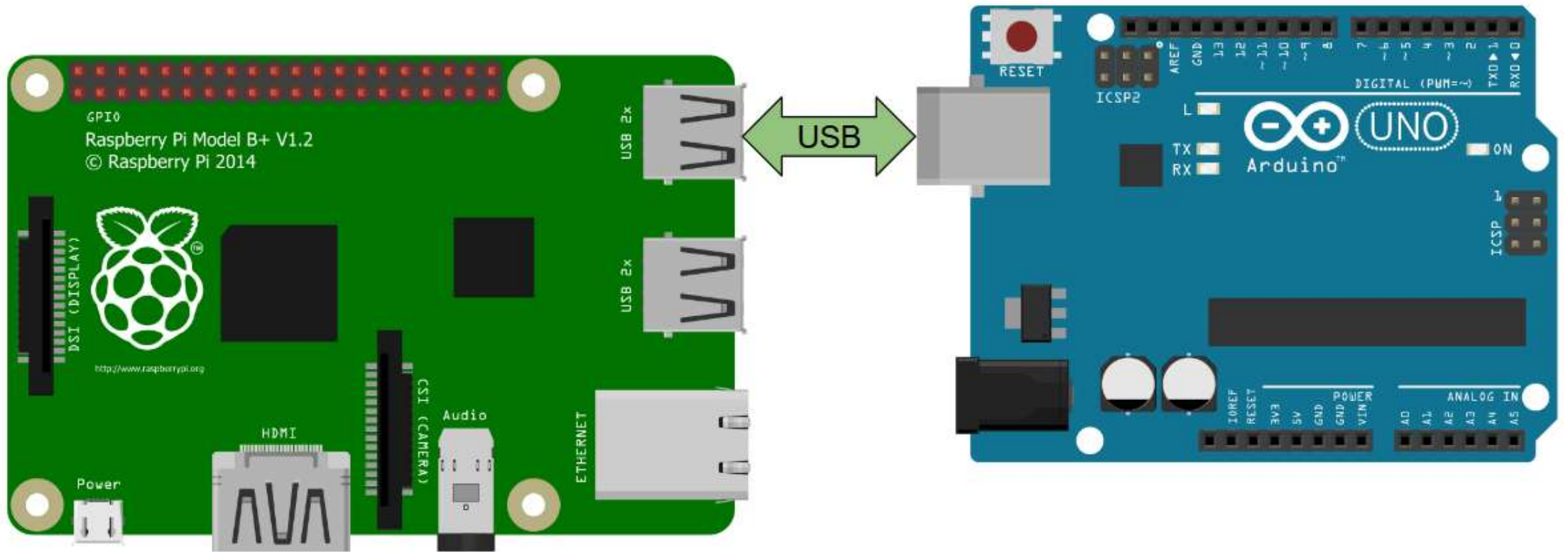


Raspberry Pi & Arduino Serial Communication

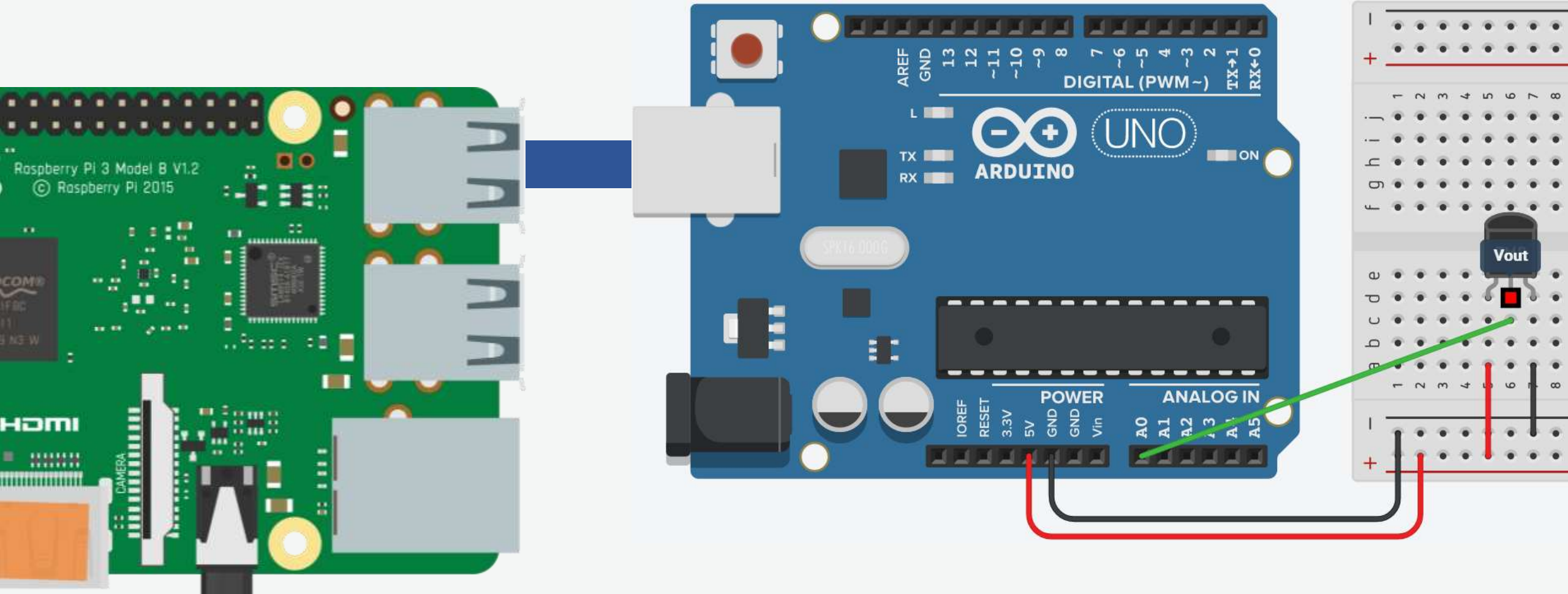


Raspberry Pi & Arduino Serial Communication

- The easiest way is to use a **USB** cable between both board.

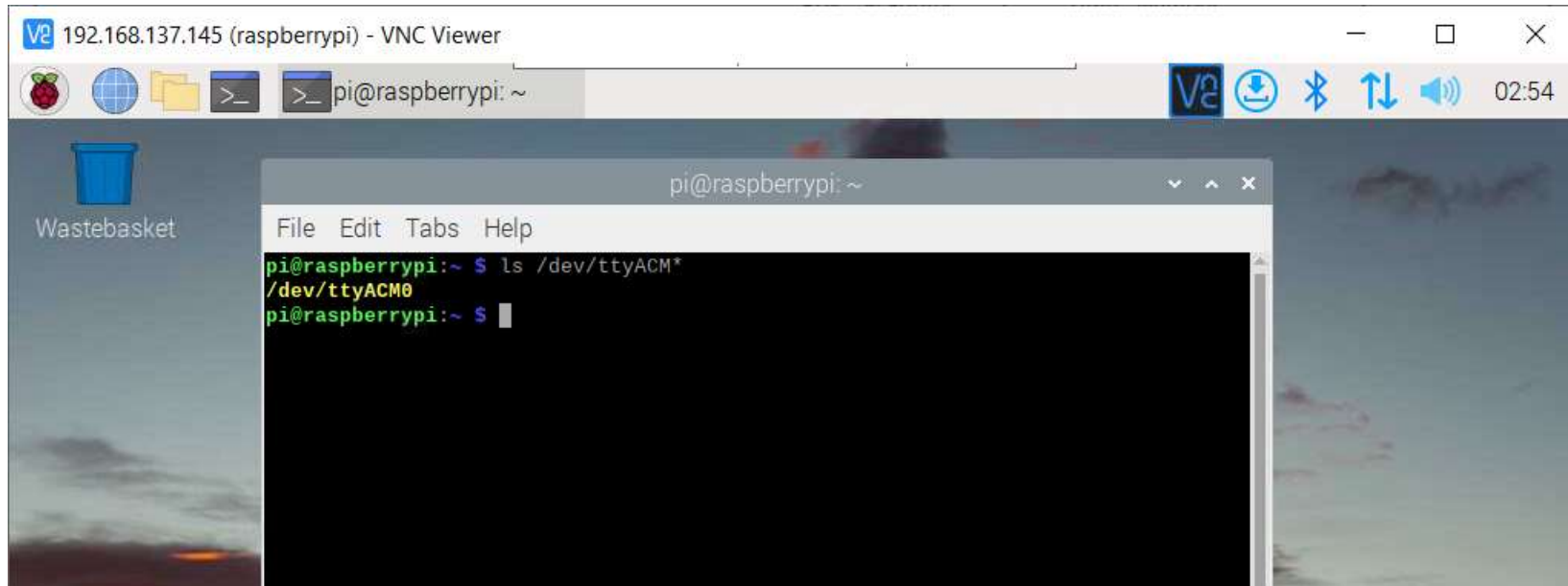


Raspberry Pi & Arduino Serial Communication



Raspberry Pi & Arduino Serial Communication

- When connecting the Arduino with a USB cable, you should see it appear as `/dev/ttyACM0`, or `/dev/ttyUSB0`.
- Sometimes the number can be different, for example `/dev/ttyACM1`.
- Simply run `ls /dev/ttyACM*` and you should see it.

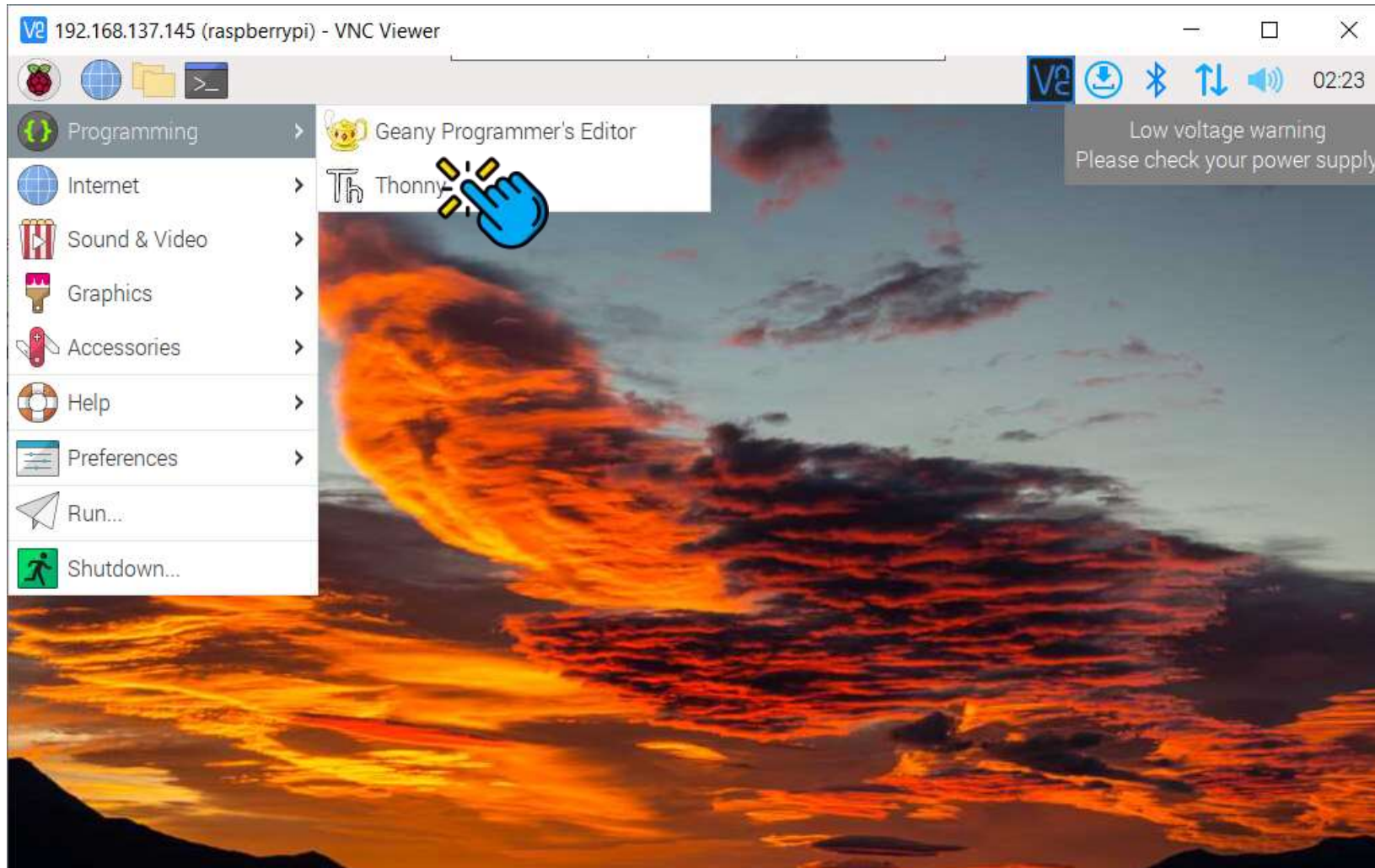


The screenshot shows a VNC Viewer window titled "192.168.137.145 (raspberrypi) - VNC Viewer". The desktop background is a sunset over water. A terminal window is open, displaying the following text:

```
pi@raspberrypi:~  
File Edit Tabs Help  
pi@raspberrypi:~ $ ls /dev/ttyACM*  
/dev/ttyACM0  
pi@raspberrypi:~ $
```

Raspberry Pi & Arduino Serial Communication

- Open **Thonny** Python IDE.



Raspberry Pi & Arduino Serial Communication

```
# Import the serial library
import serial

# Initialize serial communication
ser = serial.Serial('/dev/ttyACM0', 9600)

# Flush any bytes that could already be in the input buffer
ser.flush()

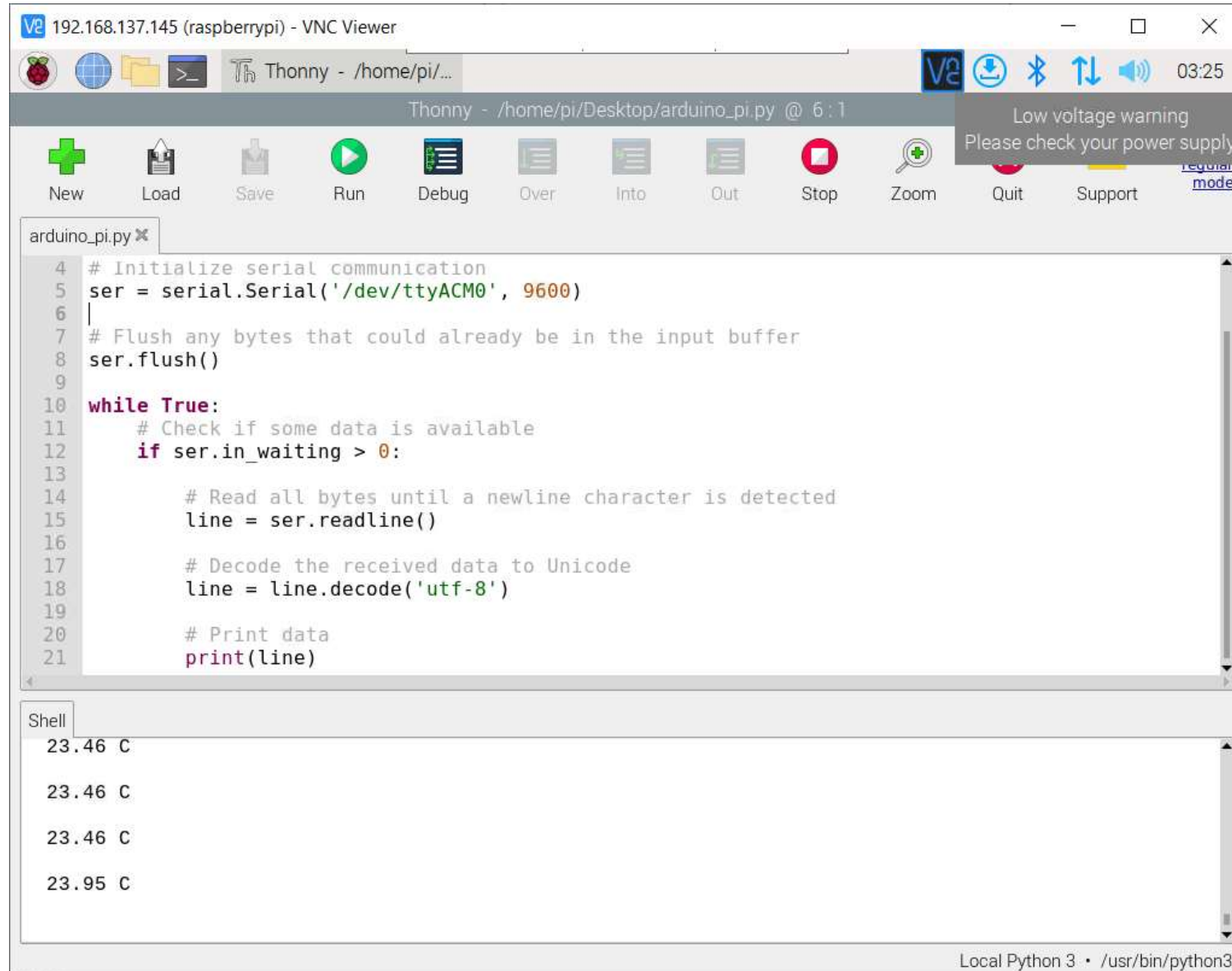
while True:
    # Check if some data is available
    if ser.in_waiting > 0:

        # Read all bytes until a newline character is detected
        line = ser.readline()

        # Decode the received data to Unicode
        line = line.decode('utf-8')

        # Print data
        print(line)
```

Raspberry Pi & Arduino Serial Communication



The screenshot displays a Raspberry Pi desktop environment accessed via VNC. The window title is "192.168.137.145 (raspberrypi) - VNC Viewer". The desktop features a Thonny IDE window titled "Thonny - /home/pi/Desktop/arduino_pi.py @ 6:1". The IDE's menu bar includes options: New, Load, Save, Run, Debug, Over, Into, Out, Stop, Zoom, Quit, and Support. A "Low voltage warning" notification is present, stating "Please check your power supply". The code in the IDE is as follows:

```
4 # Initialize serial communication
5 ser = serial.Serial('/dev/ttyACM0', 9600)
6
7 # Flush any bytes that could already be in the input buffer
8 ser.flush()
9
10 while True:
11     # Check if some data is available
12     if ser.in_waiting > 0:
13
14         # Read all bytes until a newline character is detected
15         line = ser.readline()
16
17         # Decode the received data to Unicode
18         line = line.decode('utf-8')
19
20         # Print data
21         print(line)
```

Below the IDE is a terminal window titled "Shell" showing the output of the program:

```
23.46 C
23.46 C
23.46 C
23.95 C
```

The bottom right corner of the desktop shows the system tray with icons for VNC, network, Bluetooth, and volume, along with the time "03:25". The status bar at the bottom of the window indicates "Local Python 3 • /usr/bin/python3".

References

- [Interfacing LM35 Temperature Sensor with Arduino](#)
- [Install Raspbian OS and Connect to Hotspot](#)
- [Raspberry Pi Arduino Serial Communication](#)