

Féidearthachtaí as Cuimse  
Infinite Possibilities

# Week 8

## Raspberry Pi

Fundamentals of IoT  
Dr. Eoin Rogers (eoin.rogers@tudublin.ie)



# Lesson Outline

- We're going to introduce the full-size Raspberry Pi this week
- Raspberry Pi programming with gpiozero
- We'll also review MQTT in preparation for this week's lab!
- We should have time to talk a bit about the lab at the end

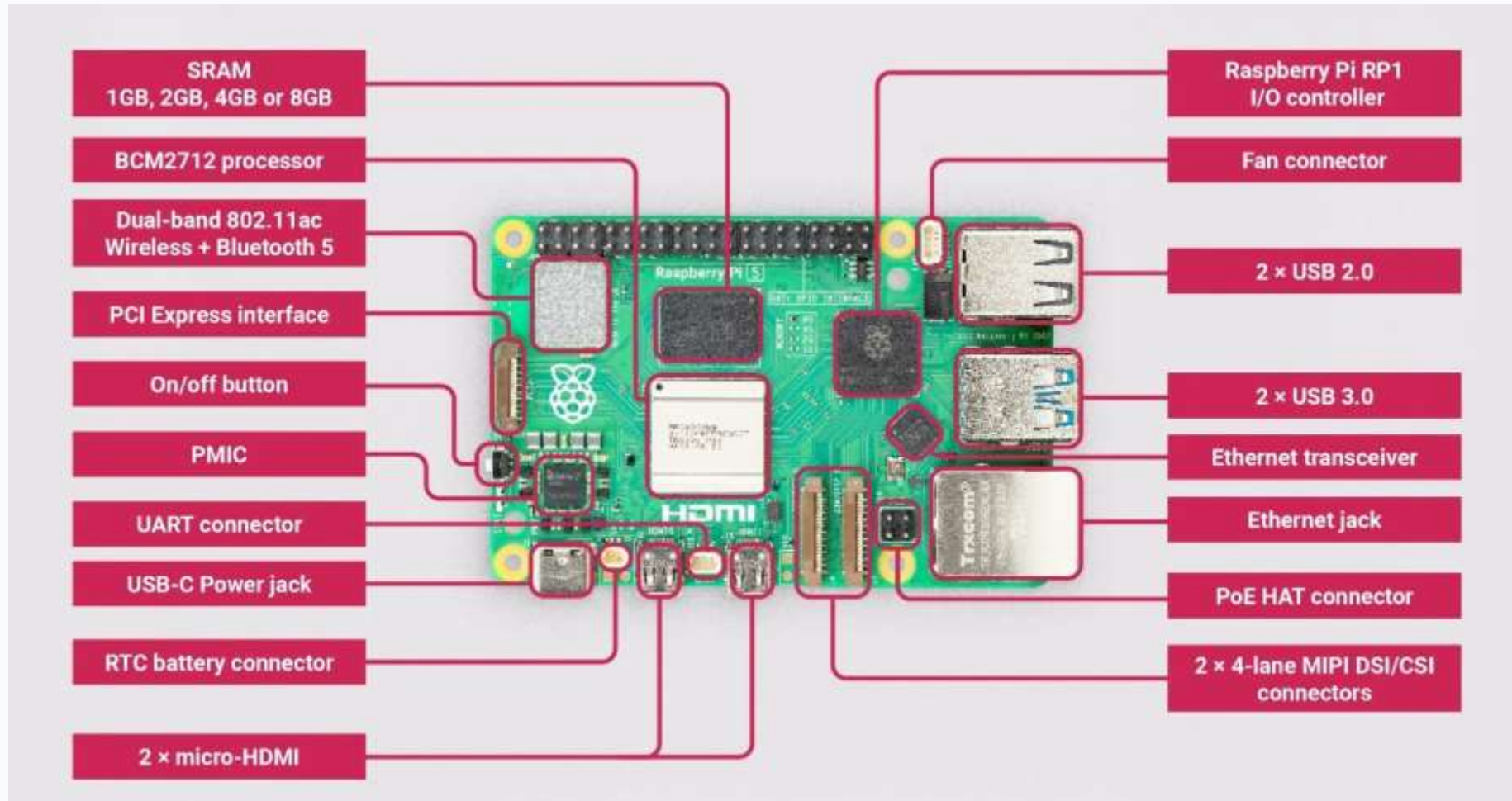
# Raspberry Pi

A single-board computer, but very unlike the Pico!

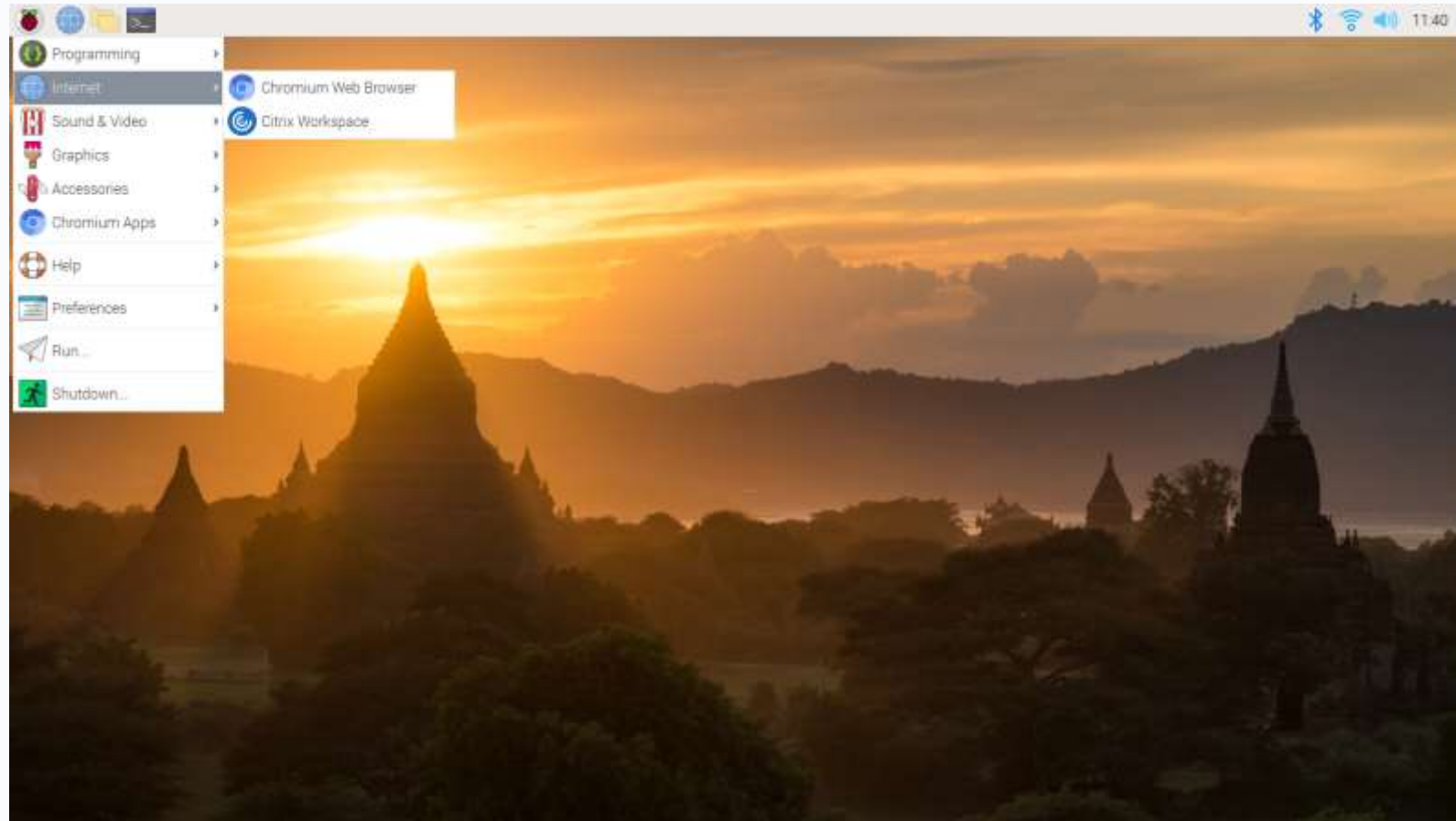
# Raspberry Pi 5 specs

- 2.4 GHz 64-bit Arm processor (Broadcom SoC with 4 Cortex-A76 cores)
- 800 MHz GPU
- 4 or 8GB of RAM
- USB, Wifi, Gigabit Ethernet, support for OpenGL and Vulkan
- But no built-in secondary storage (we have SD cards, can also boot over the network or USB)

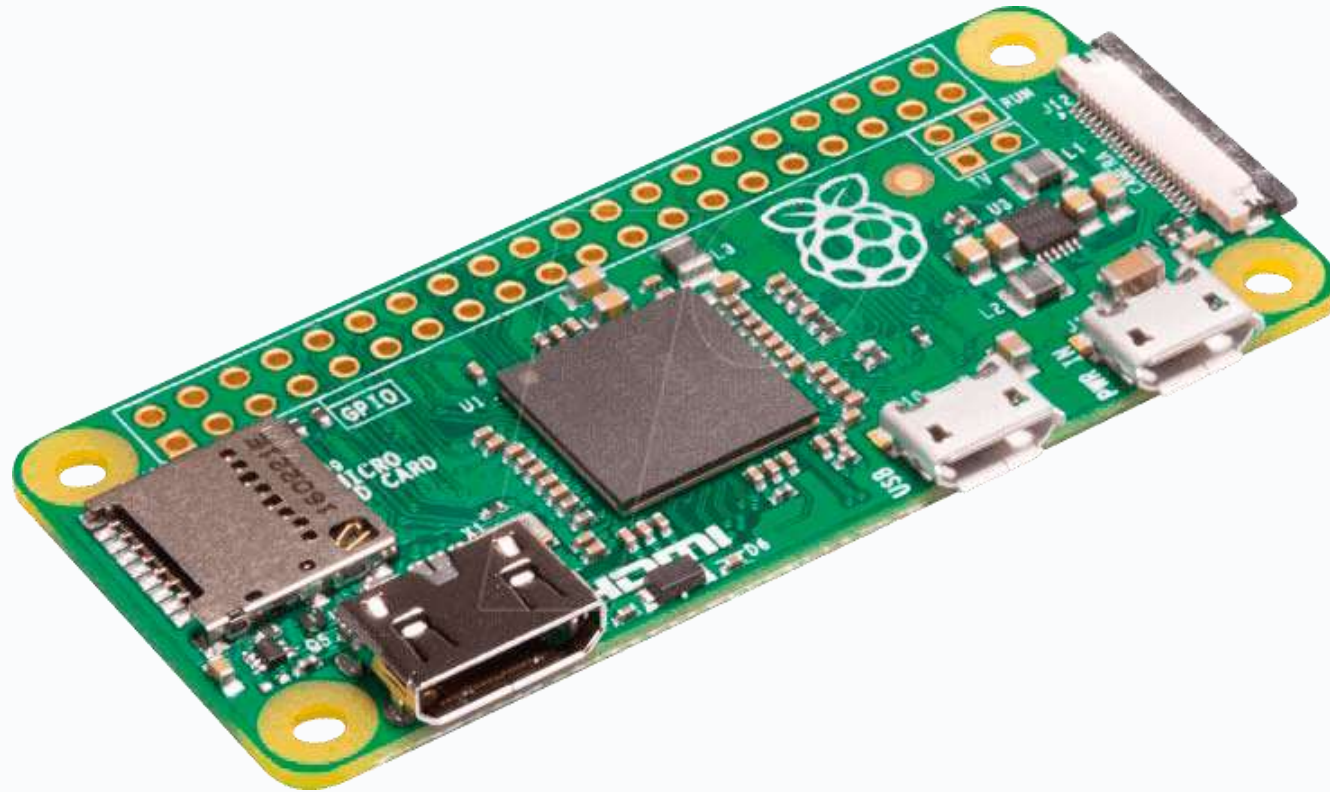
# Raspberry Pi 5 diagram



# It runs Linux!



# Raspberry Pi Zero





# Raspberry Pi 400

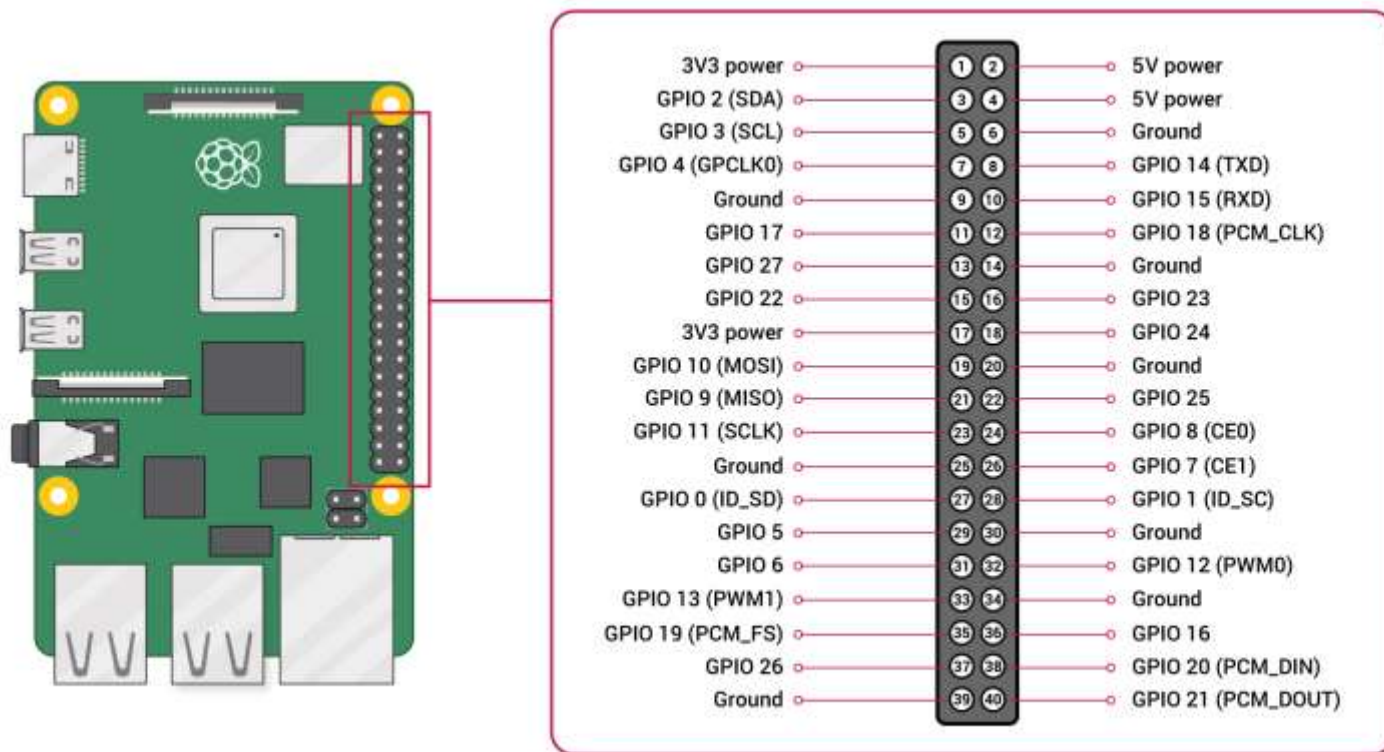




# GPIO on the Raspberry Pi

The Pi's GPIO facilities are similar to the Pico's

# Raspberry Pi GPIO pinout



Note the distinction between the **Broadcom** and **Board** naming schemes – this is important!

# Libraries

- A number of GPIO libraries exist for the Pi, and support a wide range of languages
- The actual GPIO is controller from userspace via ioctls into a kernel module
- Both low-level and high-level APIs have been implemented on top of this for a range of languages

# gpiozero

Very easy to use Python GPIO library

# Controlling an LED

```
import gpiozero  
  
my_led = gpiozero.LED(17)  
  
my_led.on()  
my_led.off()
```

# Blinking an LED

```
import gpiozero, time

my_led = gpiozero.LED(17)

while True:
    my_led.on()
    time.sleep(1)
    my_led.off()
    time.sleep(1)
```



# PWM LED control

```
import gpiozero  
  
led = gpiozero.PWMLED(12)  
  
led.value = 0.5
```

# Toggle an LED with a button

```
import gpiozero

button = gpiozero.Button(17)
led = gpiozero.LED(2)

while True:
    button.wait_for_press()
    led.toggle()
```

# Callbacks for hardware events

```
import gpiozero

def pressed():
    print('You pressed the button')

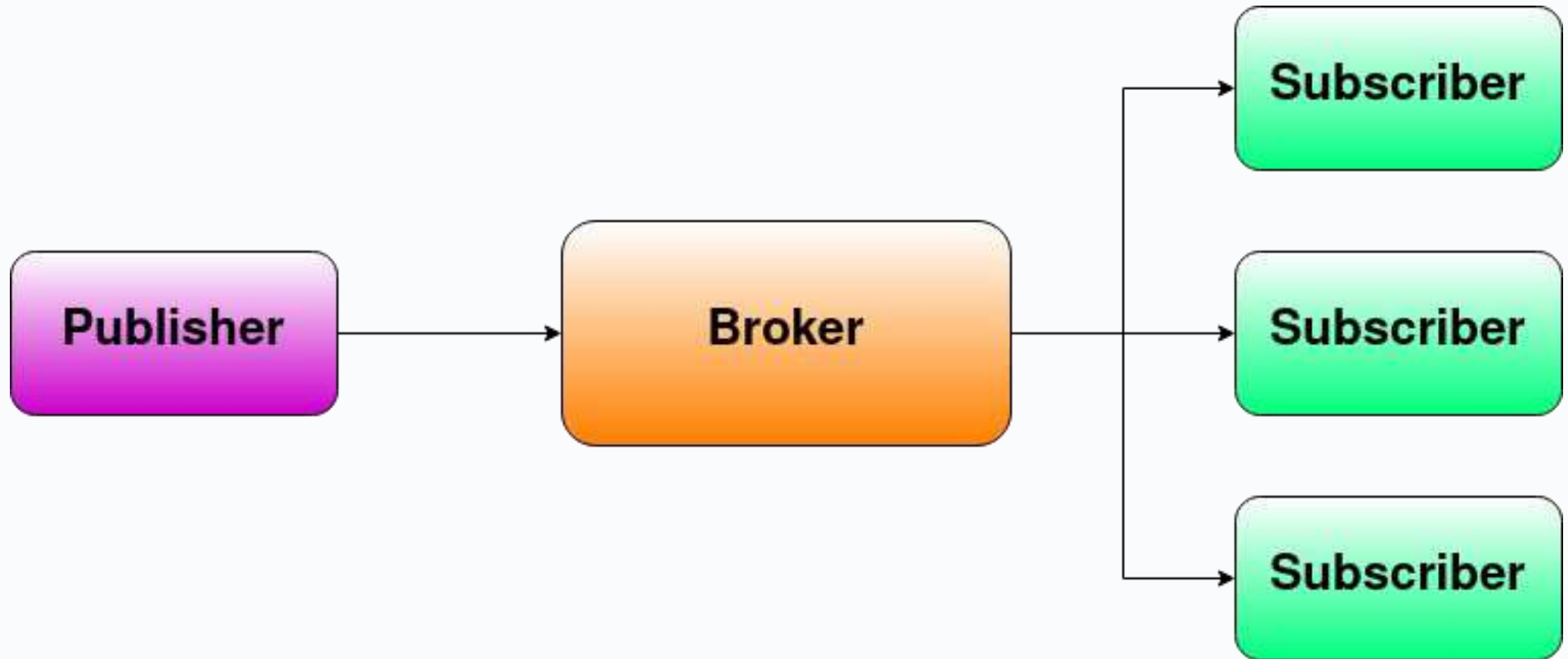
def released():
    print('You released the button')

button = gpiozero.Button(10)
button.when_pressed = pressed
button.when_released = released
```

# MQTT overview

You'll need to understand MQTT for the lab, so let's review it!

# MQTT



# Motor controller

A device used to control an electric motor



# Motor controller



A device to control an electric motor. For example, the one shown to the left is used to control the lens of a digital camera. Why might we need such a device? Why not wire the motor directly to the GPIO pins? Where have we seen a motor controller before?

# Summary

- Introduced the Raspberry Pi
- Gave you a taste of programming the GPIO facilities. All other types of programming are the same as on a normal PC or laptop: it is just an ordinary computer running Linux!
- Reviewed about MQTT
- Introduced motor controllers, a new type of component, which is a device that controls one or more electric motors

That's all for this week

Thanks for your attention!