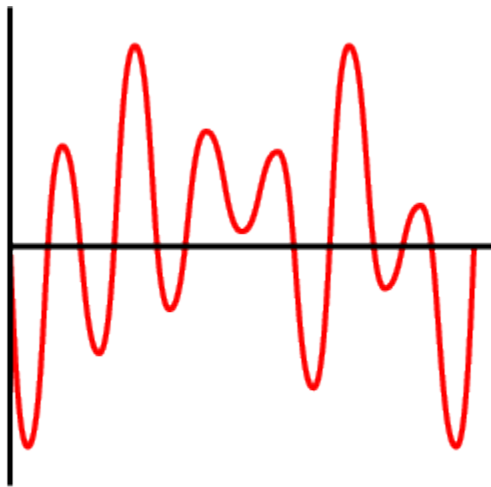


Lesson 5a – Light Dependent Resistor (LDR)

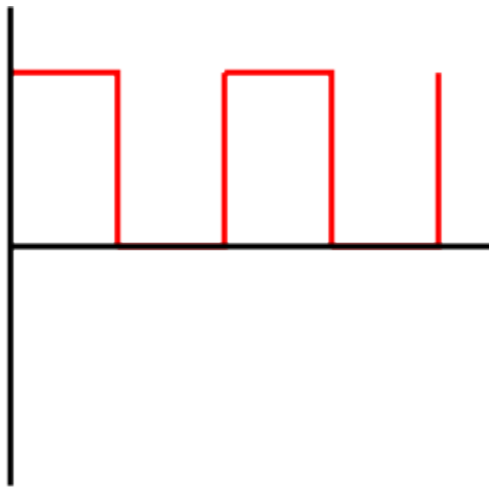
In the world of electronics, there are two types of input and output (I/O): analogue and digital. Digital I/O is fairly easy to understand; it's either *on* or *off*, 1 or 0.

When talking about voltages and the Raspberry Pi, any input that is approximately below 1.8V is considered *off* and anything above 1.8V is considered *on*. For output, 0V is off and 3.3V is on.

Analogue I/O is a little trickier. With an analogue input, we can have a range of voltages from 0V up to 3.3V, and the Raspberry Pi is unable to detect exactly what that voltage is.



Analogue inputs could be given a range of voltages, anywhere from 0V up to 3.3V.



Digital inputs are either on or off. Any voltage above 1.8V is considered on, and below that is considered off.

How, then, can we use a Raspberry Pi to determine the value of an analogue input, if it can only tell when the voltage to a GPIO pin goes above 1.8V?

Using a capacitor for analogue inputs

Capacitors are electrical components that store charge.



When current is fed into a capacitor, it will begin to store charge. The voltage across the capacitor will start off low, and increase as the charge builds up.

By putting a resistor in series with the capacitor, you can slow the speed at which it charges. With a high resistance, the capacitor will charge slowly, whereas a low resistance will let it charge quickly.

If you time how long it takes the capacitor's voltage to get over 1.8V (or be *on*), you can work out the resistance of the component in series with it.

Light-dependent resistors

An LDR (sometimes called a photocell) is a special type of resistor.

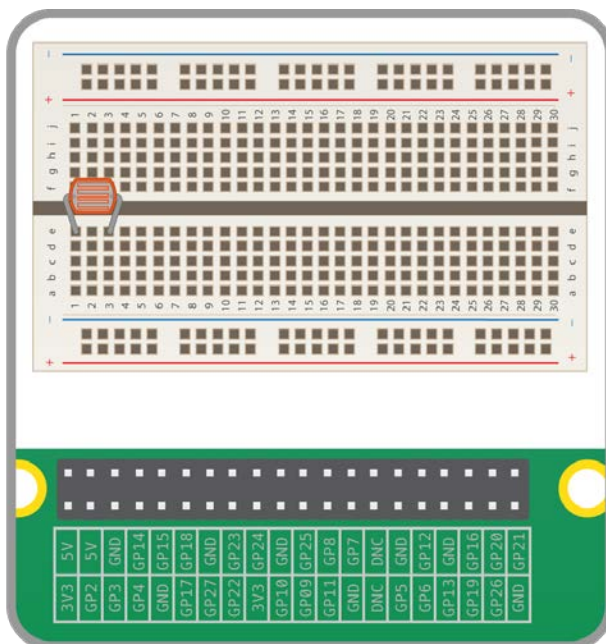


When light hits the LDR, its resistance is very low, but when it's in the dark its resistance is very high.

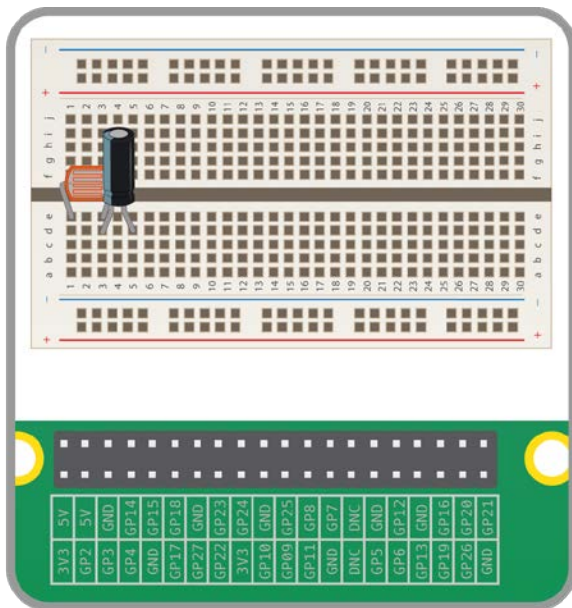
By placing a capacitor in series with an LDR, the capacitor will charge at different speeds depending on whether it's light or dark.

Creating a light-sensing circuit

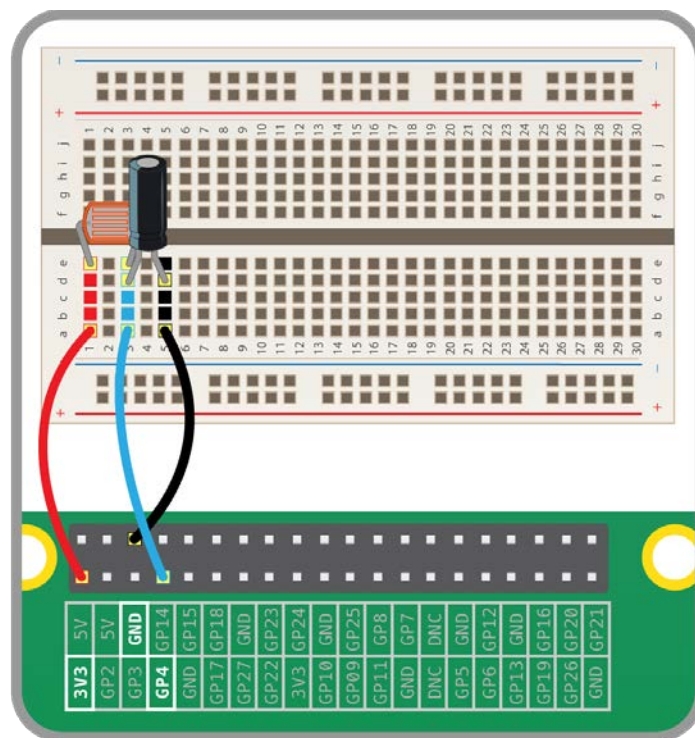
1. Place an LDR into your breadboard, as shown below:



1. Now place a capacitor in series with the LDR. As the capacitor is a polar component, you must make sure the long leg is on the same track as the LDR leg.



1. Finally, add jumper leads to connect the two components to your Raspberry Pi.



Coding a light sensor

Luckily, most of the complicated code you would have to write to detect the light levels received by the LDR has been abstracted away by the `gpiozero` library. This library will handle the timing of the capacitor's charging and discharging for you.

Use the following code to set up the light sensor. Call your program **ldrTimer.py**

```
from gpiozero import LightSensor, Buzzer
```

```

ldr = LightSensor(4) # alter if using a different pin

while True:

    print(ldr.value)

```

Run this code, then cover the LDR with your hand and watch the value change. Try shining a strong light onto the LDR.

Alternate code IF the above does not work:

Be careful to add the 4 spaces for indents, or you will get an error!

```

import RPi.GPIO as GPIO, time # Get all the libraries we need
GPIO.setmode(GPIO.BOARD) # Set the GPIO library to use pin numbers from the Pi Board

def timer (pin): # Create a new function
    reading = 0 # Create our counter and set it to zero
    GPIO.setup(pin, GPIO.OUT) # Set the pin to output
    GPIO.output(pin, GPIO.LOW) # Set the pin to low to discharge the capacitor
    time.sleep(0.1) # wait for 100ms whilst the capacitor discharges
    GPIO.setup(pin, GPIO.IN) # Set the pin to input
    while (GPIO.input(pin) == GPIO.LOW): # keep looping until the capacitor is charged and
the input hits high
        reading += 1 # add one to our counter each loop
    return reading # when the loop finishes, return the reading
while True:
    result = timer(7)
    print result

```