

More Input

What is a potentiometer?

- A potentiometer is an analog device that lets you vary electrical potential (voltage) between 2 values.
- Normally those 2 values are **ground** and **power**, but don't have to be.
- Inside a potentiometer when turned, a wiper moves across a resistive strip changing the proportion of resistance of **ground** to **output** and **power** to **output**.
- This causes the voltage on the **output** pin to vary.



What is a photo-resistor?

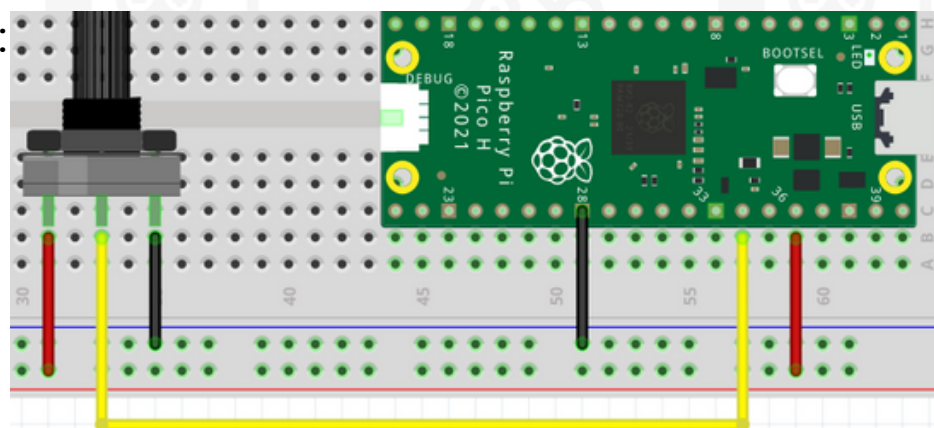
- A photo-resistor, sometimes called a Light Dependent Resistor (LDR) simply varies its resistance based on the intensity of light.
- Like a resistor, its direction in a circuit doesn't matter.
- In order to use one to get an analog voltage output, dependent on light level, we need to incorporate it into a similar form as the potentiometer.



Getting a reading from the potentiometer:

Hook up the pi pico as shown:

Note that the Pi Pico has special pins for Analog to Digital Conversion. These ADC pins are 26, 27, 28 and we are just using 28



Now for the code:

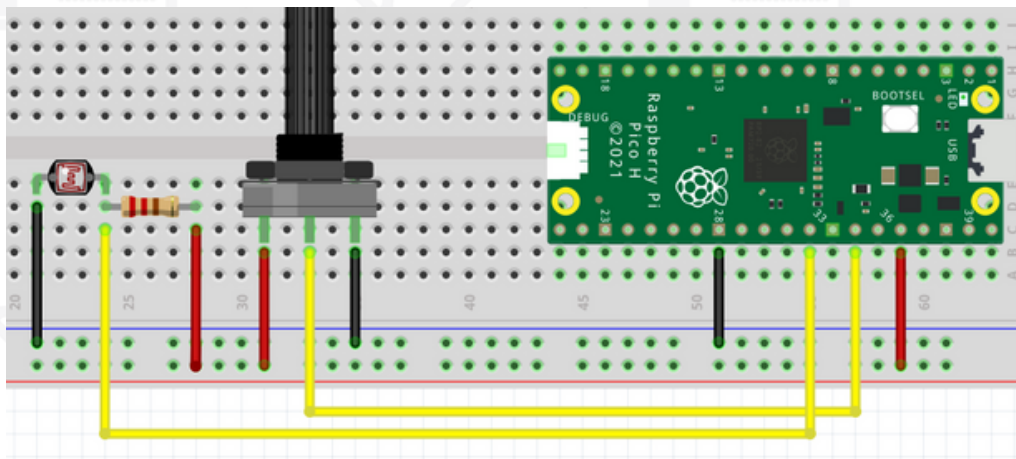
```
1 from machine import ADC
2 import time
3
4 potOut = ADC(28) # Initialize the potentiometer pin
5
6 while True:
7     reading = potOut.read_u16() #read value as 16 bit integer(0 - 65535)
8     print("ADC: ",reading)
9     time.sleep(0.2)
```

We have to import a new module from machine for the ADC readings.

When you run the code, you should get a bunch of numbers in the console between 0 and 65535, these represent voltages 0v and 3.3v respectively.

Getting a reading from the photo-resistor:

Now you can hookup the photoresistor as a voltage divider (like how the potentiometer functions), well be using a GL5516 photo resistor.



And also modify your code to read both signals:

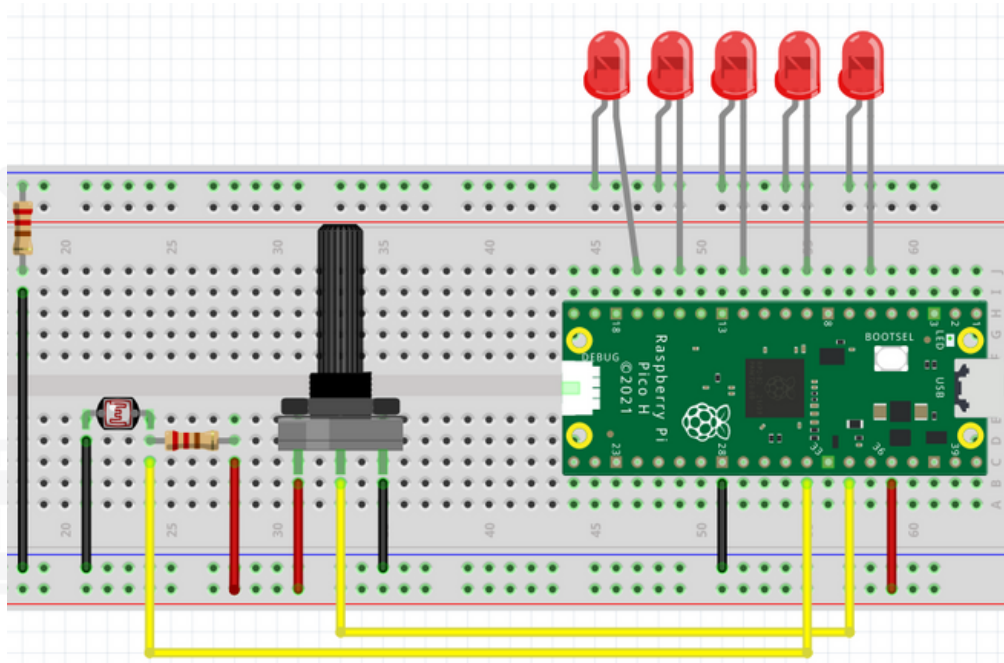
```
1 from machine import ADC
2 import time
3
4 potOut = ADC(28)
5 LightOut = ADC(27)
6
7 while True:
8     reading1 = potOut.read_u16()
9     reading2 = LightOut.read_u16()
10
11     print("Pot: ", reading1)
12     print("LDR: ", reading2)
13
14     time.sleep(0.2)
```

Once you run your code, test the photo-resistor using your phone torch

Controlling the speed of trailing LED's:

In the first workshop you went over how to power a single LED, now lets see how we can power and control 5 different LED's.

Hookup some more wires and LED's as shown bellow:



The LEDs are connected to GPIO pins 4, 6, 9, 11, 13

Once all wired up import Pin from machine and add the following lines of code before while loop:

```
6
7 # defining all LEDs in an array to be able to iterate through them with a loop:
8 leds = [Pin(4, Pin.OUT), Pin(6, Pin.OUT), Pin(9, Pin.OUT), Pin(11, Pin.OUT), Pin(13, Pin.OUT)]
9
```

And at the end of the while loop:

```
16
17     delay = reading1/65535 # making the delay between 0 and 1 second
18
19     for led in leds: # iterating through LED's
20         led.on()
21         time.sleep(delay)
22         led.off()
23
```

When you run your code, the speed of the flashing LED's should change depending on the potentiometer setting

Review and extra challenges:

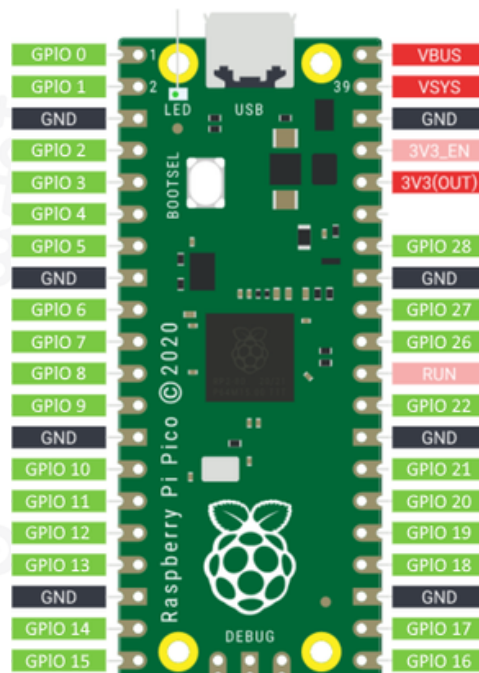
Today we have expanded on the basics of LED's and incorporated a potentiometer to control thier speed. We only briefly covered the process of reading from Photo-resistors because they will be used in a more complex system in a later workshop.

Challenge 1: Make the delay between LED's update from the potentiometer at every LED instead of after the cycle of 5.

Challenge 2: Add a buzzer to your circuit and make it trigger when the photo-resistor is bellow a certain value (given enough light)

Thinking point: What happens when you power all 5 LED's at the same time? Can you explain this behaviour?

Pinout Reference



Pi Pico Code Docs

