GPIO Controlled LEDs

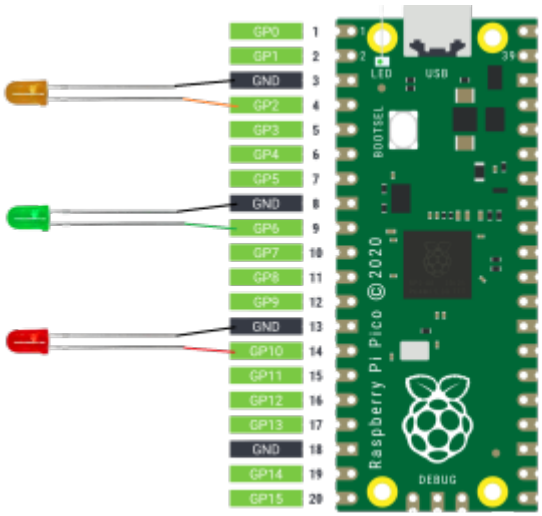
Jun 2026

MicroPython

Introduction

For the first three LED examples we have used the onboard LED on the Pico (pin25). But if we want to control other LEDs then we need to start to the use the GPIO pins available on the Pico.

Here is a wiring diagram for the LED setup. This is what we will use when we run the software below.



We have wired the LEDs to GPIO pins 2, 6 & 10. These are the GPIO Pin numbers, not the actual Pi Pico Pin numbers.

Below is the software being used to drive the LEDs

download

```
# Start by importing lib files "machine" & "time"
# Imports the machine module, which provides access to hardware features such as GPIO pins, ADCs,
PWM, etc.
import machine
# Imports the time module, which provides delay functions like sleep().
import time

#we're initalizing pin 25 (which is the onboard LED) & pins 2, 6 and 6 as outputs
# Creates a pin object called led1.
# Uses GPIO pin 25.
# Sets the pin mode to output (Pin.OUT).
# On a Raspberry Pi Pico, GPIO 25 is usually connected to the onboard LED.
led1 = machine.Pin(25, machine.Pin.OUT)
# Creates an output pin object for GPIO 2.
led2 = machine.Pin(2, machine.Pin.OUT)
# Creates an output pin object for GPIO 6.
led3 = machine.Pin(6, machine.Pin.OUT)
# Creates an output pin object for GPIO 10.
led4 = machine.Pin(10, machine.Pin.OUT)
```

```
#Run a loop 5 times
# Starts a loop that repeats 5 times.
# The variable i takes values from 0 to 5.
for i in range(5):
# Sets GPIO 2 HIGH (3.3V).
# Turns LED 2 on (assuming the LED is wired active-high).
    led2.value(1)
# Sets GPIO 6 & 10 LOW.
# Turns LED 3 and 4 off.
    led3.value(0)
    led4.value(0)
    #Use time to wait for 0.5 second
    time.sleep(0.2)

# The rest of the code is a repetition of what has been done above.
    #Turn the onbaord LED off
    led1.value(0)
    #Turn external LED on pin 2 (only) on
    #led1.value(0)
    led2.value(1)
    led3.value(0)
    led4.value(0)
    #Use time to wait for 1 second
    time.sleep(0.11)
    #Turn external LED on pin 3 (only) on
    #led1.value(0)
    led2.value(0)
    led3.value(1)
    led4.value(0)
    #Use time to wait for 1 second
    time.sleep(0.2)
    #Turn external LED on pin 4 (only) on
    #led1.value(0)
    led2.value(0)
    led3.value(0)
    led4.value(1)
    #Use time to wait for 1 second
    time.sleep(0.2)
    #Turn the external LED on pin 4 off
    led4.value(0)
```

The software works by performing the following tasks.

- Import Libraries (Machine and Time)
- * Set GPIO Pins to output
- Start a loop that will loop 5 times
- Turn on the first LED and turn off all other LEDs
- Turn on the second LED and turn off all other LEDs
- Turn on the third LED and turn off all other LEDs

That's it. Pretty simple. Remember, you can control other devices using the GPIO pins other than LEDs, but the current output is limited to around 20mA. If you try to pull more than this, you could damage that GPIO Pin.

From:
https://wiki.alanwalker.uk/ - **WalkerWiki - wiki.alanwalker.uk**

Permanent link:
https://wiki.alanwalker.uk/doku.php?id=gpio_controlled_leds&rev=1781713930

Last update: **2026/06/17 16:32**