Pin Objects Teacher Guide

Pin Objects Teacher Guide

When you create (instantiate) a **Pin** object in MicroPython on the Raspberry Pi Pico, you assign it to a variable to make it easier to work with.

The Pin object can be customised with different parameters, such as the pin number, mode (input or output), and pull resistor configuration.

Let’s say you want to create a Pin object for a button connected to GPIO pin 14 and assign it to a variable named button\_pin. Additionally, you want to create a Pin object for an LED connected to GPIO pin 25 and assign it to a variable named led\_pin.

A screenshot of a computer program

Description automatically generated

**button\_pin** is a variable that now holds a Pin object representing the button. The Pin constructor is called with three arguments: the pin number (14), the pin mode (Pin.IN for input), and the pull resistor configuration (Pin.PULL\_UP for a pullup resistor).

**led\_pin** is a variable that now holds a Pin object representing the LED. The Pin constructor is called with two arguments: the pin number (25) and the pin mode (Pin.OUT for output).

Once you have these variables, you can use them to interact with the corresponding pins. For example:

A screenshot of a computer program

Description automatically generated

Now, **button\_pin** and **led\_pin** act as convenient references to the specific GPIO pins you’ve defined, making it easier to read and understand your code.

**To PULL\_UP or PULL\_DOWN?**

Whether to use a pull-up or pull-down resistor on a GPIO (General Purpose Input/Output) pin of the Raspberry Pi Pico is a personal choice. The Pico has internal pull-up and pull-down resistors. The Pin.PULL\_UP or Pin.PULL\_DOWN argument in the `Pin` constructor activates the internal resistors.

**1. Pull-Up Resistor:**

A pull-up resistor connects the GPIO pin to the voltage supply (usually 3.3V on the Pico).

It ensures that the GPIO pin reads a high state (1) when it's not actively driven low.

**2. Pull-Down Resistor:**

A pull-down resistor connects the GPIO pin to ground.

It ensures that the GPIO pin reads a low state (0) when it's not actively driven high.

It is easier for students starting out to understand the pull down as the read values are 0 for off and 1 for on. This is the same for LED so I tend to stick with this option to begin with and later introduce the pull up resistor.

It is worth noting that students who undertake independent research may come across the pull up method in tutorials. This often leads to problems with the readings not given the expected result.