

**CSE446 Lab 1**

**Introduction to Robotics Lab**

**Task 1:** Introduction to the Raspberry Pi GPIO pins, and using push buttons to control LEDs

**Section:** 11

**Group:** 2

| **Name** | **ID** |
| --- | --- |
| Ariful Islam | 21301016 |
| Abrar Faiyaz | 21301152 |
| Md Sayem Mottakee | 21301080 |
| Khan Mohammad Al Kareeb | 21301680 |
| Miskatunnisa Labonno | 20201029 |

**Description:**

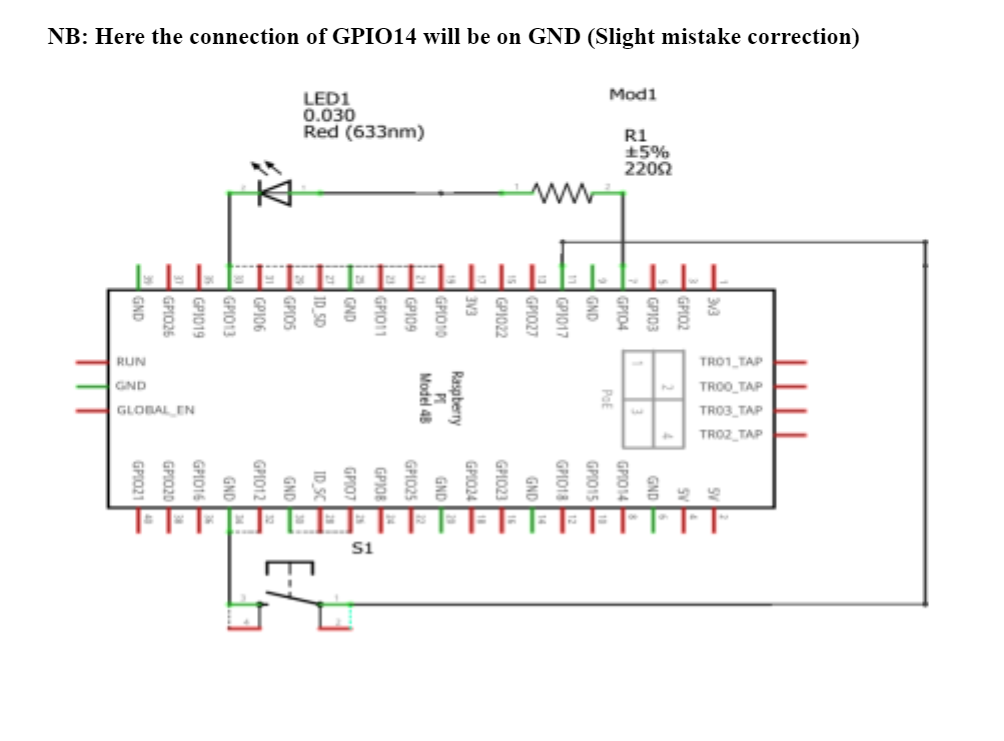
This lab experiment, conducted by Group No: 02, explores the fundamentals of the Raspberry Pi GPIO (General Purpose Input/Output) pins. The primary focus is on understanding the various types of pins on the Raspberry Pi and their potential applications. Additionally, the experiment involves programming the GPIO pins to control various electronic circuits, such as LEDs, motors, and sensors. One notable application involves using push buttons in conjunction with LEDs to create a simple user-controlled circuit. The experiment also delves into the necessity of using a resistor to safeguard the LED when interfacing it with the GPIO pins.

**Component requirement:**

To conduct this experiment, the following components are necessary:

* Raspberry Pi 3
* Breadboard
* Push-Button
* A 220-ohm resistor
* Connecting wires
* LED

**Circuit diagram:**

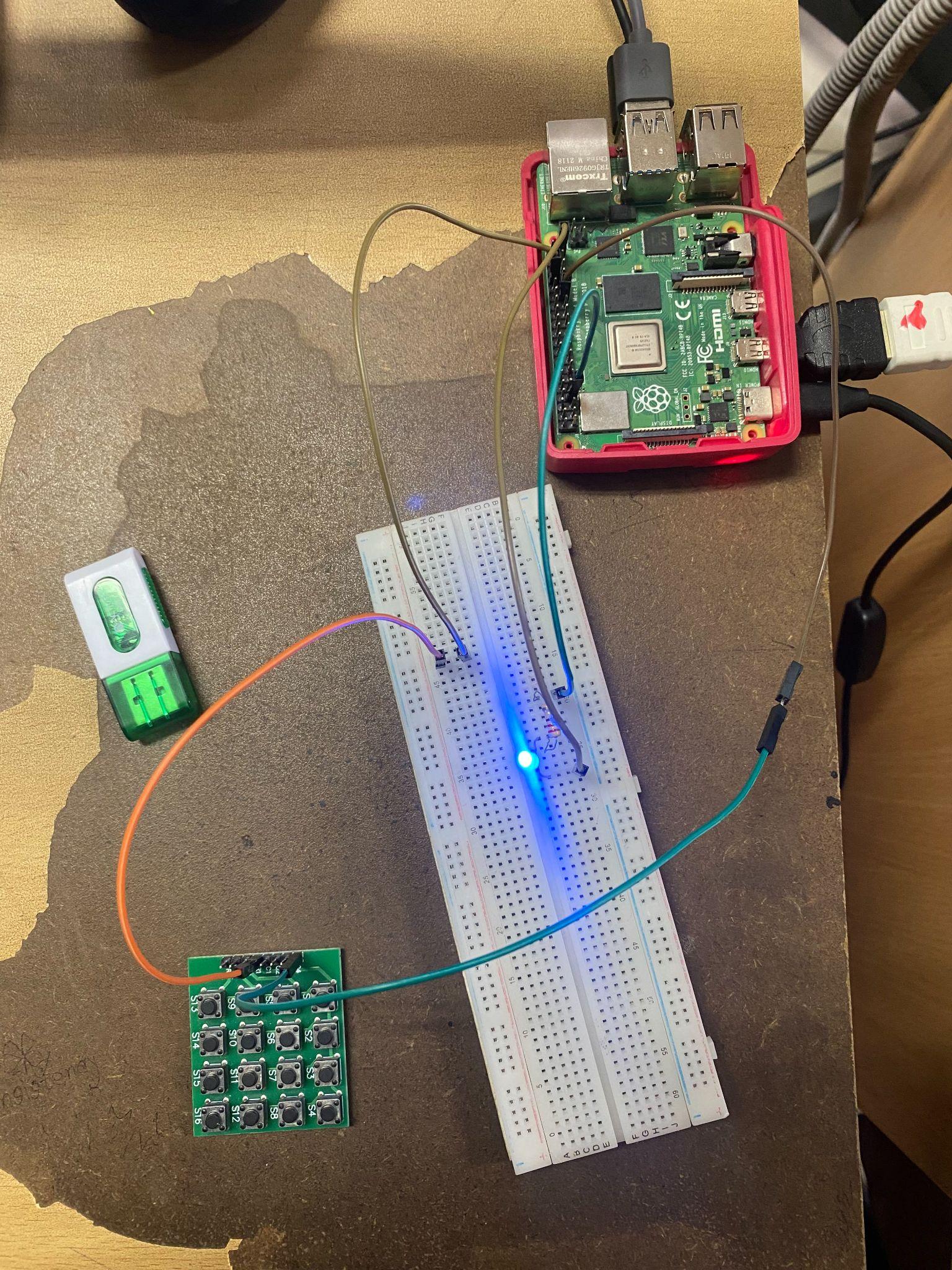
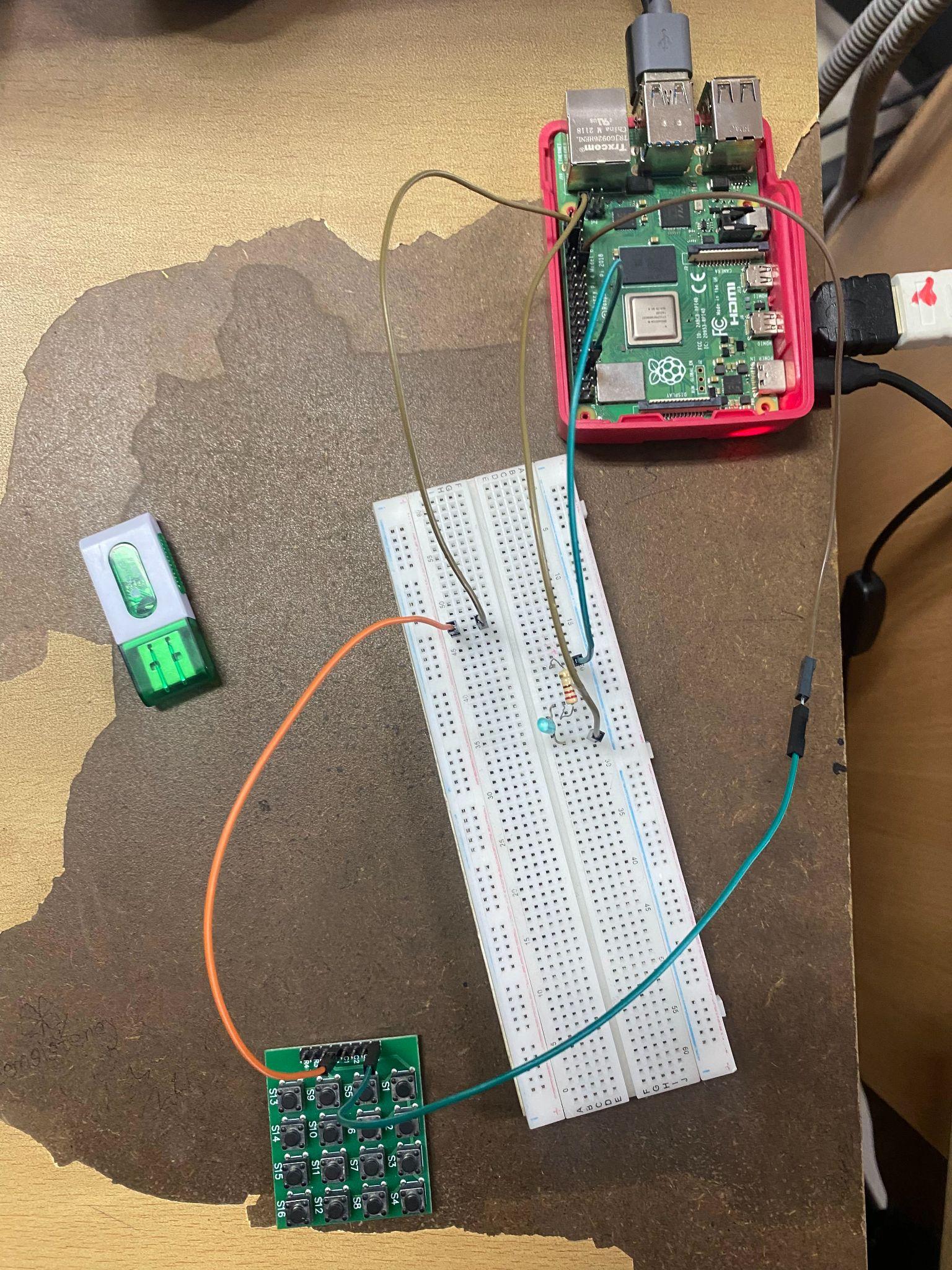
****

**Circuit explanation:**

The purpose of the 220-ohm resistor in series with the LED is to limit the current flowing through the LED to a safe level, preventing it from burning out due to excessive current.

The push button is connected to a GPIO pin to allow the Raspberry Pi to detect when the button is pressed (resulting in a low voltage on the GPIO pin) and when it's released (resulting in a high voltage on the GPIO pin). This configuration enables the Raspberry Pi to control the LED based on the state of the button.

**Circuit setup:**

****

**Code\_01:**

**Code for led on when button is pressed and off when button released:**

import Rpi.GPIO as GPIO

from gpiozero import LED

from gpiozero import Button

from time import sleep

GPIO.setmode(GPIO.BCM)

led = LED(4)

button = Button(5)

while True:

button.wait\_for\_press()

led.on()

print(“Pressed”)

button.wait\_for\_release()

led.off()

print(“Released”)

**Code\_02:**

**Code to maintain the LED on for 10 seconds, followed by a 10-second sleep period.**

import RPi.GPIO as GPIO

from gpiozero import LED

from gpiozero import Button

from time import sleep

GPIO.setmode(GPIO.BCM)

led = LED(4)

button = Button(5)

while True:

led.on() # Turn the LED on

sleep(10) # Keep the LED on for 10 seconds

led.off() # Turn the LED off

sleep(10) # Keep the LED off for 10 seconds

**Code\_03:**

**Code to turn the LED on when the button is pressed and released, and to turn it off when the button is pressed and released again**

import RPi.GPIO as GPIO

from gpiozero import LED

from gpiozero import Button

from time import sleep

GPIO.setmode(GPIO.BCM)

led = LED(4)

button = Button(5)

while True:

button.wait\_for\_press()

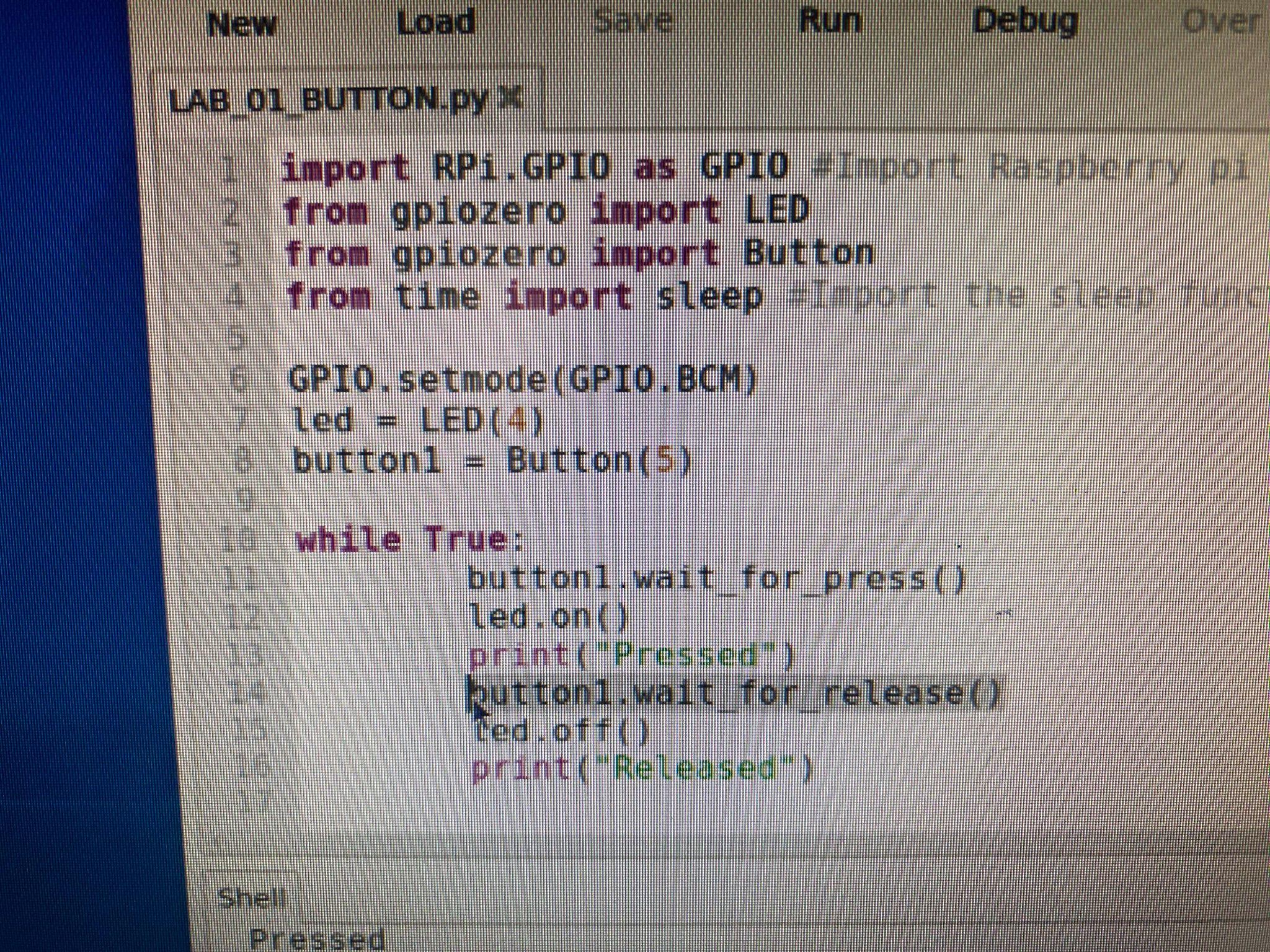
button.wait\_for\_release()

led.on()

button.wait\_for\_press()

button.wait\_for\_release()

led.off()

****

**Discussion:**

1. **Why is there a 220-ohm resistor in series with the LED?**

**Answer:** LEDs are semiconductor devices that emit light when an electric current flows through them. LEDs are also very sensitive to current, and too much current can cause them to burn out. The resistor helps to limit the current flowing through the LED to a safe level. Without the resistor, the LED could draw too much current and burn out. The value of the resistor is chosen to allow the LED to shine brightly without burning out.

1. **Why is the push button connected from a GPIO pin on the RPI to the GND pin of the RPI instead of being connected directly to the LED and the resistor combination?**

**Answer:** The push button is connected to a GPIO pin on the Raspberry Pi so that the Raspberry Pi can detect when the button is pressed. When the button is not pressed, the Raspberry Pi reads a high voltage on the GPIO pin. When the button is pressed, the Raspberry Pi reads a low voltage on the GPIO pin. The Raspberry Pi can use this information to control the LED or perform other actions, such as playing a sound or sending a notification. If the push button were connected directly to the LED, the Raspberry Pi would not be able to detect when the button is pressed. Consequently, the Raspberry Pi could not control the LED based on the state of the button.

1. **What would happen if the 220-ohm resistor was replaced with a 1k-ohm resistor? What visual change would you observe?**

**Answer:** If the 220-ohm resistor were replaced with a 1k-ohm resistor, the LED would appear dimmer. This is because the 1k-ohm resistor limits the amount of current passing through the LED. In essence, the LED's brightness would decrease. It's crucial to note that using a 1k-ohm resistor may not be safe for all LEDs. Different LEDs have varying maximum safe current draws. The LED could potentially burn out if the 1k-ohm resistor is unable to maintain the current within a safe range.