



# BANGABANDHU SHEIKH MUJIBUR RAHMAN AVIATION AND AEROSPACE UNIVERSITY (BSMRAAU)

## Real Time Embedded System Sessional

### Experiment No. 02: Understand the Python Based Raspberry Pi if & elif Statement and switch based LED control System Design

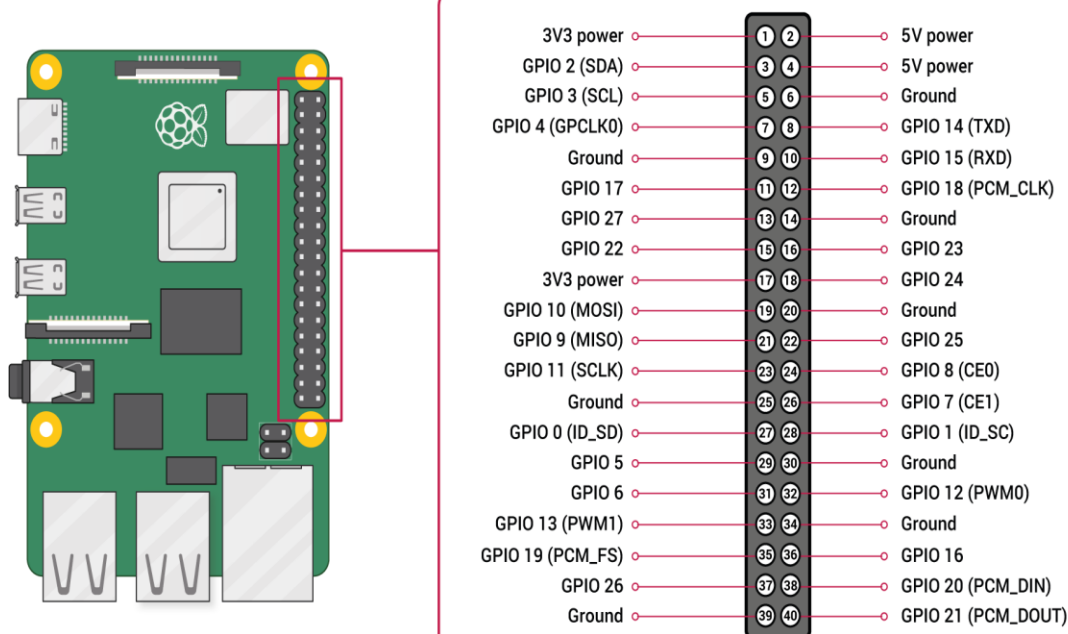
**Objectives:** The objectives of this experiment are-

- Learn how to use if & elif Statement.
- To understand the switch based control system

### Equipment:

- Embedded System module ES-101
- Desktop Computer/ Laptop

### Theory:



In Python, if and elif (short for "else if") are used for conditional statements, allowing to execute different blocks of code based on certain conditions. Here's a breakdown of how they work:

## Basic Structure

1. **if Statement:** The if statement checks a condition and executes a block of code if the condition is True.

```
Python
if condition:

    # Code to execute if condition is True
```

2. **elif Statement:** The elif statement allows you to check multiple conditions. It follows an if statement and runs its block of code if its condition is True, but only if the preceding if (or elif) was False.

```
Python
if condition1:
    # Code to execute if condition1 is True
elif condition2:
    # Code to execute if condition2 is True
```

3. **else Statement:** The else statement can be added at the end of an if/elif chain to execute a block of code if none of the previous conditions were True.

```
Python
if condition1:
    # Code to execute if condition1 is True
elif condition2:
    # Code to execute if condition2 is True
else:
    # Code to execute if both conditions are False
```

## Library Details

The gpiozero library is a Python module designed for controlling GPIO pins on a Raspberry Pi (and other devices) in a simple and intuitive way. The Button class within gpiozero allows work with buttons connected to GPIO pins easily.

1. **Importing the Library:** To use the Button class, you need to import it from the gpiozero module:

```
Python
```

```
from gpiozero import Button
```

2. Creating a Button Instance: You can create an instance of a button by specifying the GPIO pin number it's connected to:

```
Python
```

```
button = Button(2) # GPIO pin 2
```

3. Button Events: The Button class supports various events that you can use to trigger actions. Common events include:

- a) `when_pressed`: A function to call when the button is pressed.
- b) `when_released`: A function to call when the button is released.

```
Python
```

```
def on_button_pressed():  
    print("Button was pressed!")  
  
button.when_pressed = on_button_pressed
```

4. Polling the Button State: You can also check the state of the button programmatically:

```
Python
```

```
if button.is_pressed:  
    print("Button is currently pressed.")
```

## Additional Features

Debouncing: gpiozero automatically handles button debouncing, which means it ignores rapid on/off signals that may occur when the button is pressed or released.

LED Feedback: You can easily pair buttons with other components, like LEDs, to provide feedback when a button is pressed.

## Pin configuration

LED				SWITCH			
LED 1	LED 2	LED 3	LED 4	SW 1	SW 2	SW 3	SW 4
23	24	25	1	4	17	27	22

## Circuit Diagram

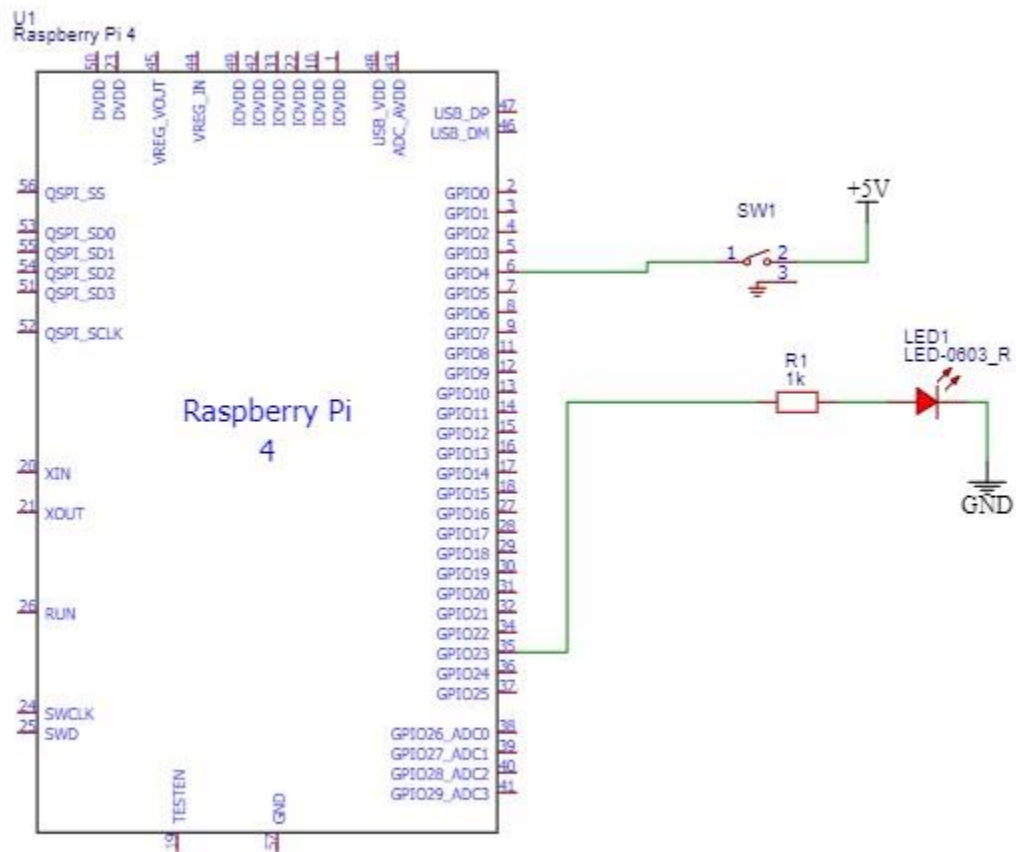


Figure 2: Circuit Diagram

### Code:

```
from gpiozero import Button
from gpiozero import LED
from time import sleep

led1 = LED(23)
button1 = Button(4, None, True)
```

```
button2 = Button(17, None, True)
```

```
while True:
```

```
    if button1.is_pressed:
```

```
        print("Button 1 is pressed")
```

```
        led1.on()
```

```
    elif button2.is_pressed:
```

```
        print("Button 1 is pressed")
```

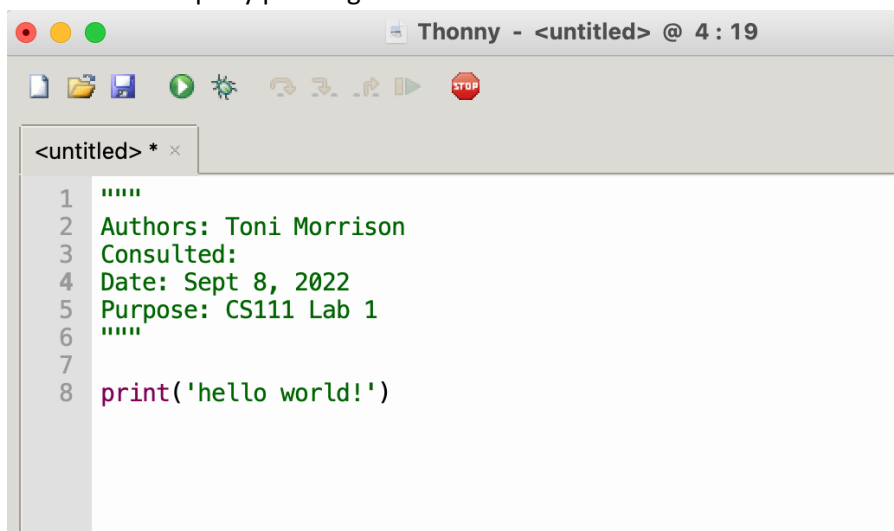
```
        led1.off()
```

## Working Procedure:

1. First turn on the power switch on the ES-101 module. After the module is turned on, the PuTTY software should be turned on from the PC and communication between the PuTTY software and VNC software should be established as per Lab Manual 1(A).
2. Install the “gpiozero” Library\*\* (if not already installed):  
Open a terminal on your Raspberry Pi and install the “gpiozero” library with:

```
sh
pip install gpiozero
```

3. Create a Python script to make the switch based LED control . Open a text editor on your ES-101 and create a file named “switch\_led.py”.
4. Execute the script by pressing RUN Button



5. Now when switch 1 is press the LEDs 1 will turn on and when switch 2 is press LED 1 is turn off

6. Press Ctrl + C on the computer keyboard to close the program. After that write exit () command and press enter button.

### **Reference Book**

1. Raspberry Pi Cookbook: Software and Hardware Problems and Solutions  
by Simon Monk
2. Python Programming for Raspberry Pi, Sams Teach Yourself in 24 Hours  
by Richard Blum & Christine Bresnahan