LAB – 7 Programming on Switch

Aim:

To write a program to utilize a push switch to control an LED.

Task:

1. To turn an LED on and off with a push switch so that it toggles between on and off each time switch is pressed.

Algorithm:

1. Set one pin as input and another pin as output in ‘Board’ mode.
2. Initialize the state of LED as False and state of input state as True.
3. Read the input pin and store the Boolean value in a variable (say new_input_state)
4. Check whether new_input_state is False and the old input state is True. If yes then change the state of the LED.
5. Wait for a while.
6. Repeat the steps 4 and 5 continuously.

Pin & Circuit Diagram:

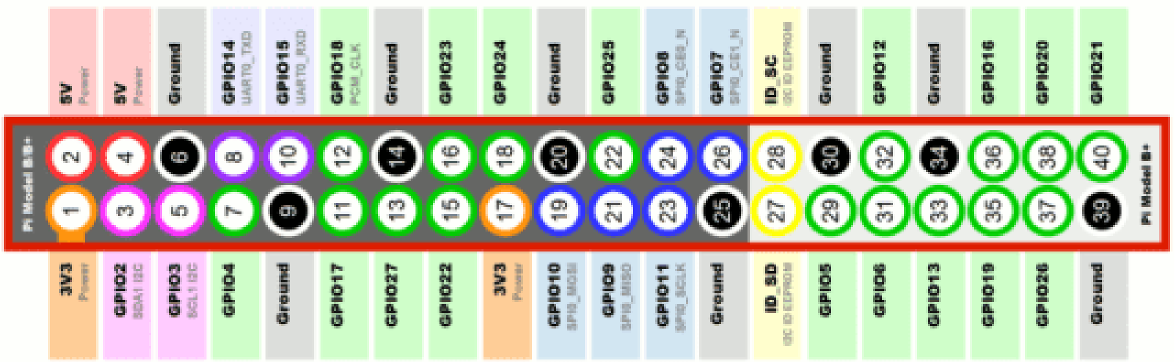


Figure 1. Pin Configuration of Board

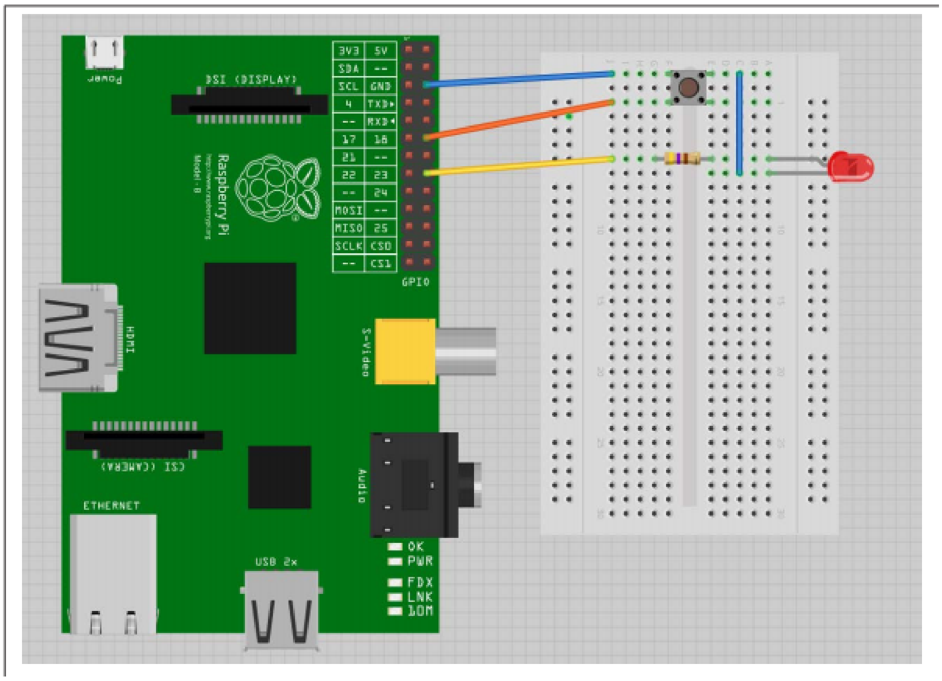
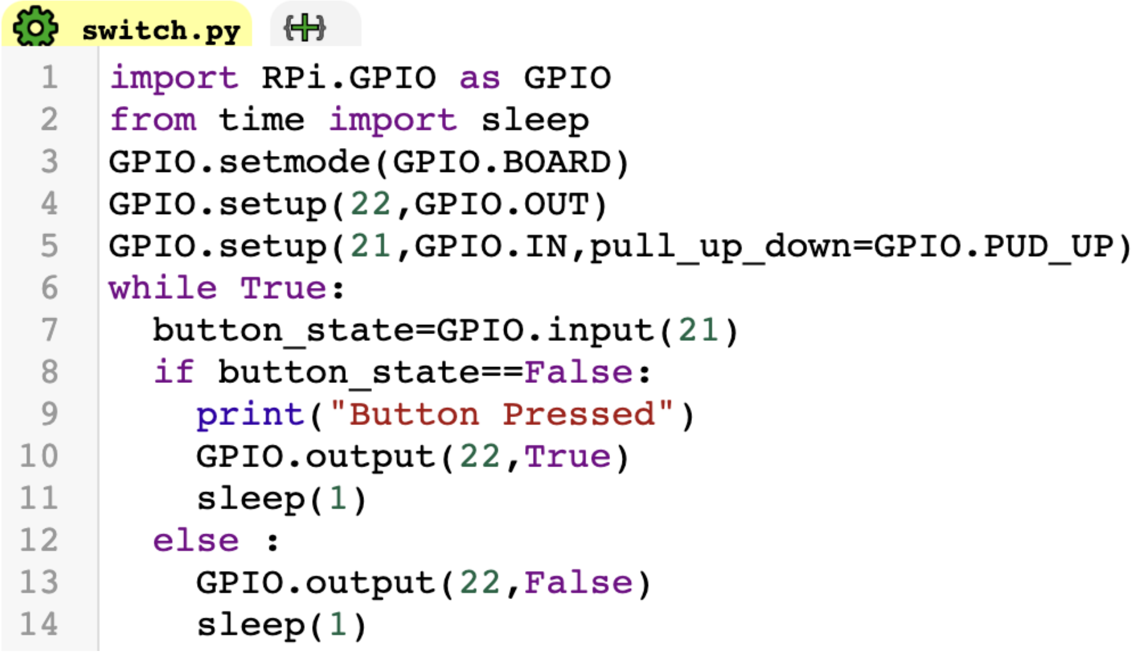


Figure 2. Raspberry Pi interfaced with a Switch and LED

Program:



```python
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BOARD)
GPIO.setup(22,GPIO.OUT)
GPIO.setup(21,GPIO.IN,pull_up_down=GPIO.PUD_UP)
while True:
  button_state=GPIO.input(21)
  if button_state==False:
    print("Button Pressed")
    GPIO.output(22,True)
    sleep(1)
  else :
    GPIO.output(22,False)
    sleep(1)
```

Pre-Lab Questions:

1. Draw a schematic of a push switch.
2. What is meant by switch bouncing? How to overcome this problem?
3. What is polling? Why is it not preferred?

Post Lab Questions:

1. How to build a rotary encoder using simple switches.

2. Write the functions of a keyboard interface available in a computer keyboard controller.

3. What is the ghost key problem? How is it minimized? (Hint:https://cdn.thomasnet.com/ccp/10110462/133780.pdf)

Result:

Thus,the python code to turn an LED on and off with a push switch so that it toggles between on and off each time the switch is pressed in Raspberry Pi was written and successfully tested.