Objective

The objective of this experiment is to introduce and familiarize the user with Raspberry Pi, a small single-board computer. The experiment provides information on the hardware required to set up Raspberry Pi and how to set up the operating system on it. The experiment also provides a basic idea of phython language and how to control an LED using Raspberry Pi and phython programming. The user will learn how to create a circuit using a breadbord and components such as an LED, a resistor, and jumper wires, and how to write phython code to control the LED. The experiment is designed to provide an introduction to Raspberry Pi and its capabilities, and to help users get started with using this powerful tool for various projects.

apparatus:

1. Activated Raspberry Pi
2. LED
3. Resistor (220 ohm)
4. Bread board
5. Jumper wires.

Theory:

The experiment provides information on the components required to set up a Raspberry Pi, including the Raspberry PI board, monitor/display, and connectivity cable, keyboard, mouse, power supply, SD card, Ethernet cable (optional), and audio lead. The experiment also provide the gaudince on setting up the operating system on Raspberry Pi. Raspbian is the official operating system for all models of Raspberry Pi, but there are also third-party operating systems available. It provides an introduction to phython, a popular programming language that can be used to control the GPIO pins of Raspberry Pi. The GPIO pins are used to interface with external components, such as LEDs, buttons, and sensors. The experiment demonstrates how to control an LED using Phython code. Which is executed from a Linux environment using a terminal and shell scripting. It includes the use of a breadboard, an essential component in electronic circuit design, to create a circuit that interfaces with Raspberry Pi. The circuit includes an LED, a resistor, and jumper wires. The experiment demonstrates how to write phython code that interacts with the ~~Gap~~ GPIO pins to control the LED, turning it on and off. overall the theory of the experiment is to provide an introduction to Raspberry Pi, its components, and capabilities. The experiment demonstrates how to set up a Raspberry Pi and provides an Introduction to phython programming and circuit design, highlighting the potential for various projects.

Procedure:

1. Gather all necessary components required for setup, such as Raspberry Pi board, monitor display and connectivity cable, keyboard and mouse, power supply, SD card, Ethernet cable, and audio lead.

2. Install the operating system on the SD card. Raspbian is the official operating system for all models of the Raspberry Pi, but you can also use third-party operating systems such as Ubuntu Mate, Snappy Ubuntu Core, Windows 10 IOT core, OSMC, Libreelec, Pinet, and RISC OS.

3. Connect the Raspberry Pi board to the monitor, keyboard, and mouse. Use the power supply to provide power to the Raspberry Pi board.

4. Boot up the Raspberry Pi and log in with the username and password provided by the operating system.

5. familiarize yourself with the linux environment by using the terminal and shell scripting. You can also use text editors such as Gvim, Nano Editor, Emacs Editor, and Pico Editor.

6. Explore the capabilities of Raspberry Pi by developing small projects such as controlling an LED using phython programming language and GPIO Pins of Raspberry Pi.

7. Continue exploring and developing more complex projects with Raspberry Pi, such as automation data logging, and robotics.

Code:

```
nano blinkLED.py
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(14, GPIO.OUT)
GPIO.output(14, GPIO.HIGH)
print "LED is ON"
time.sleep(2)
GPIO.output(14, GPIO.LOW)
print "LED is OFF"
```

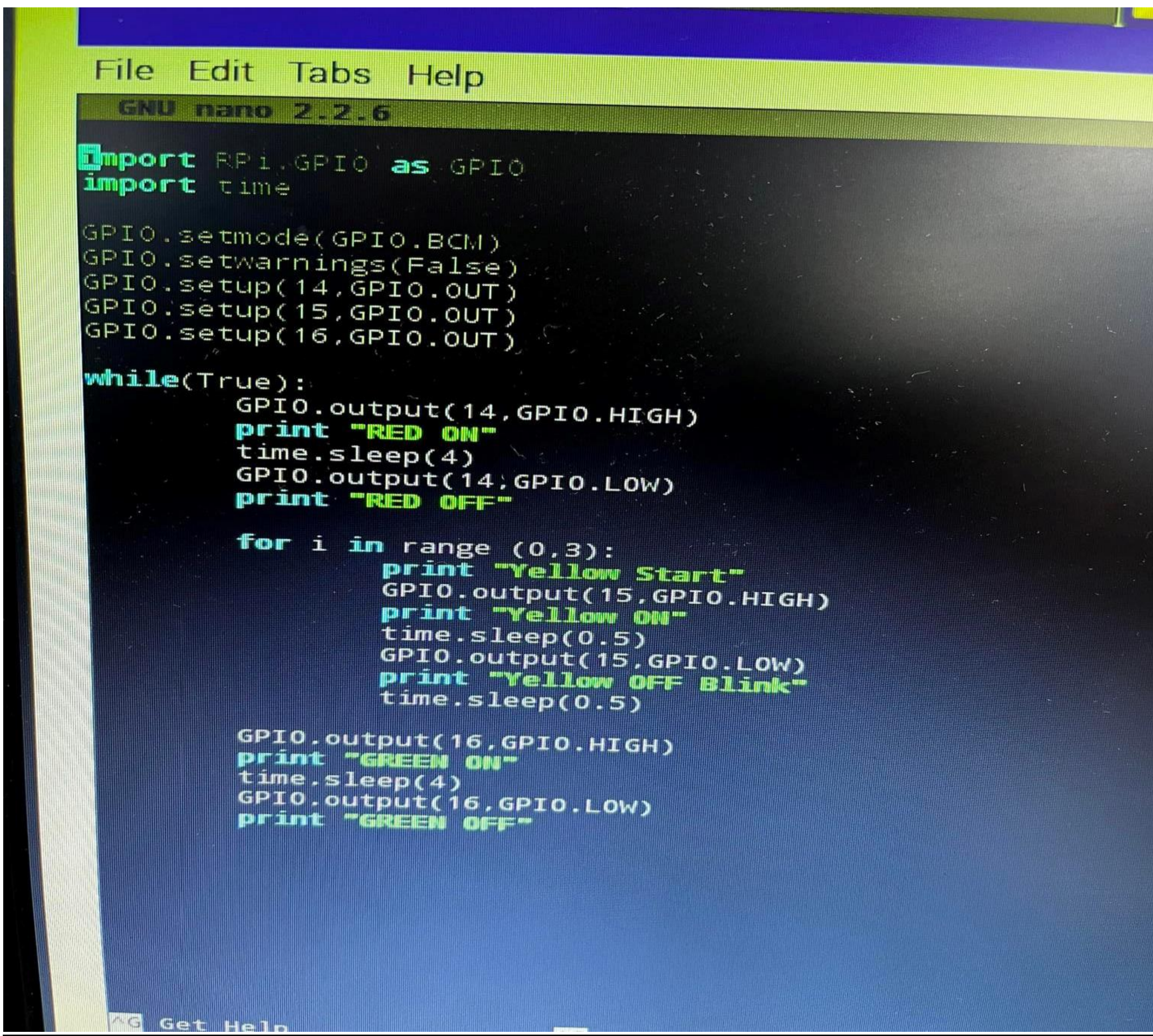

Figure: Code for traffic signal set up in Raspberry Pi

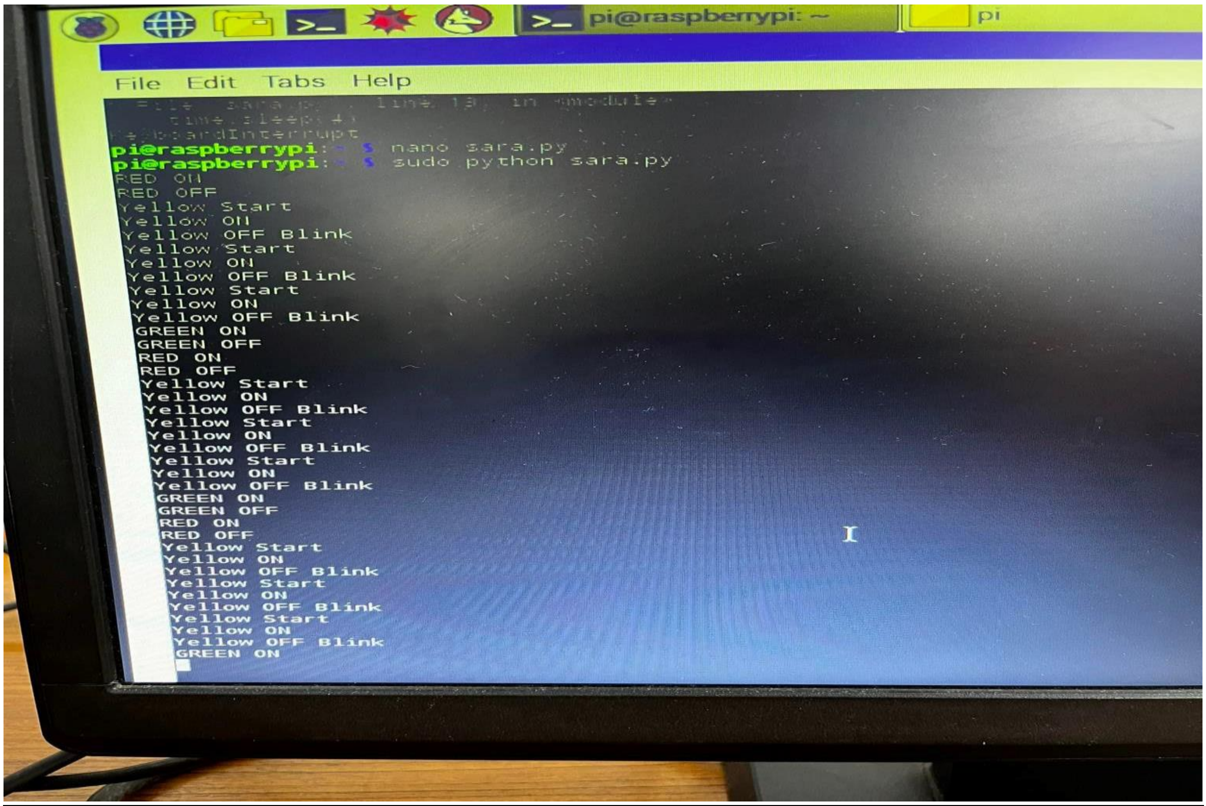

Figure: Code for blink LED

<u>Simulation:</u>

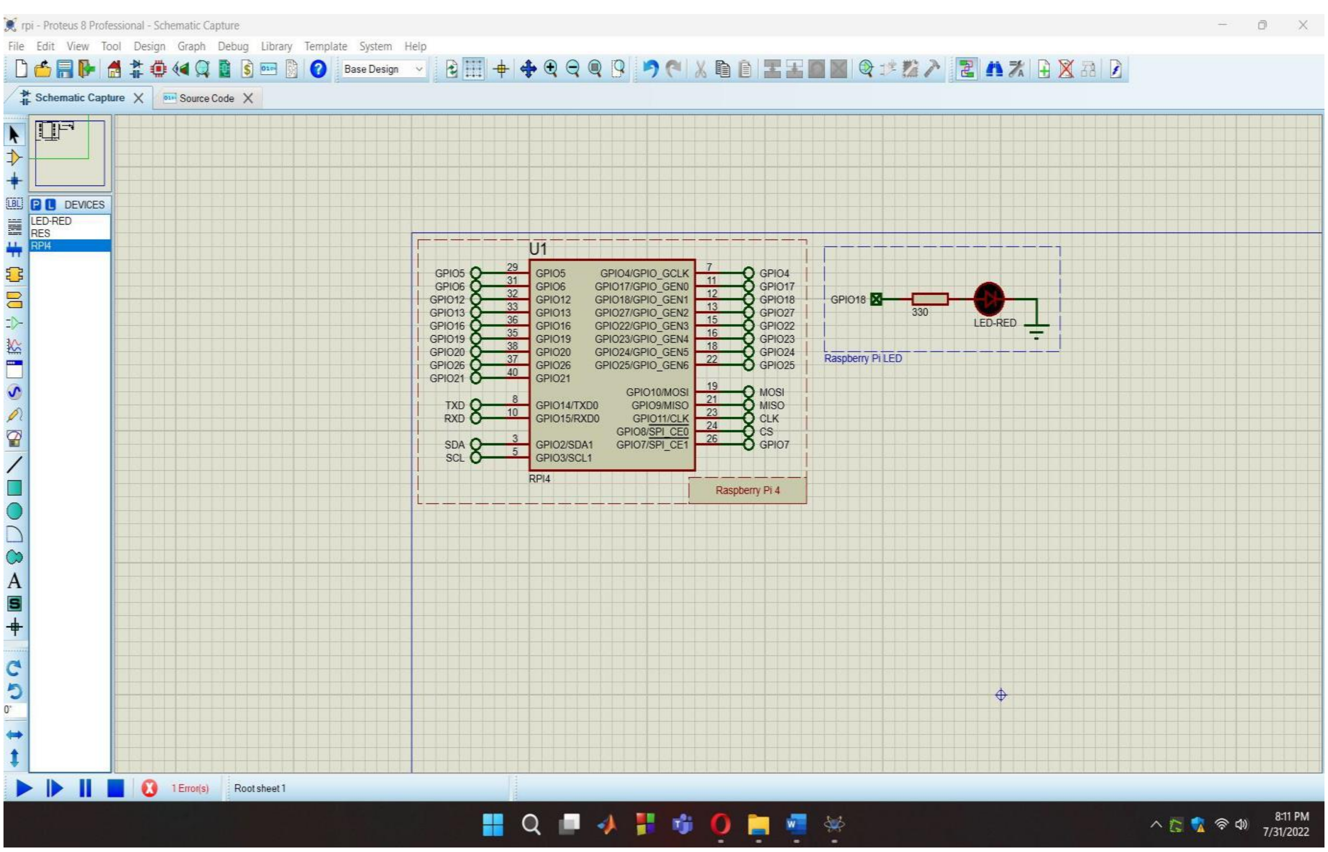

Figure: LED OFF

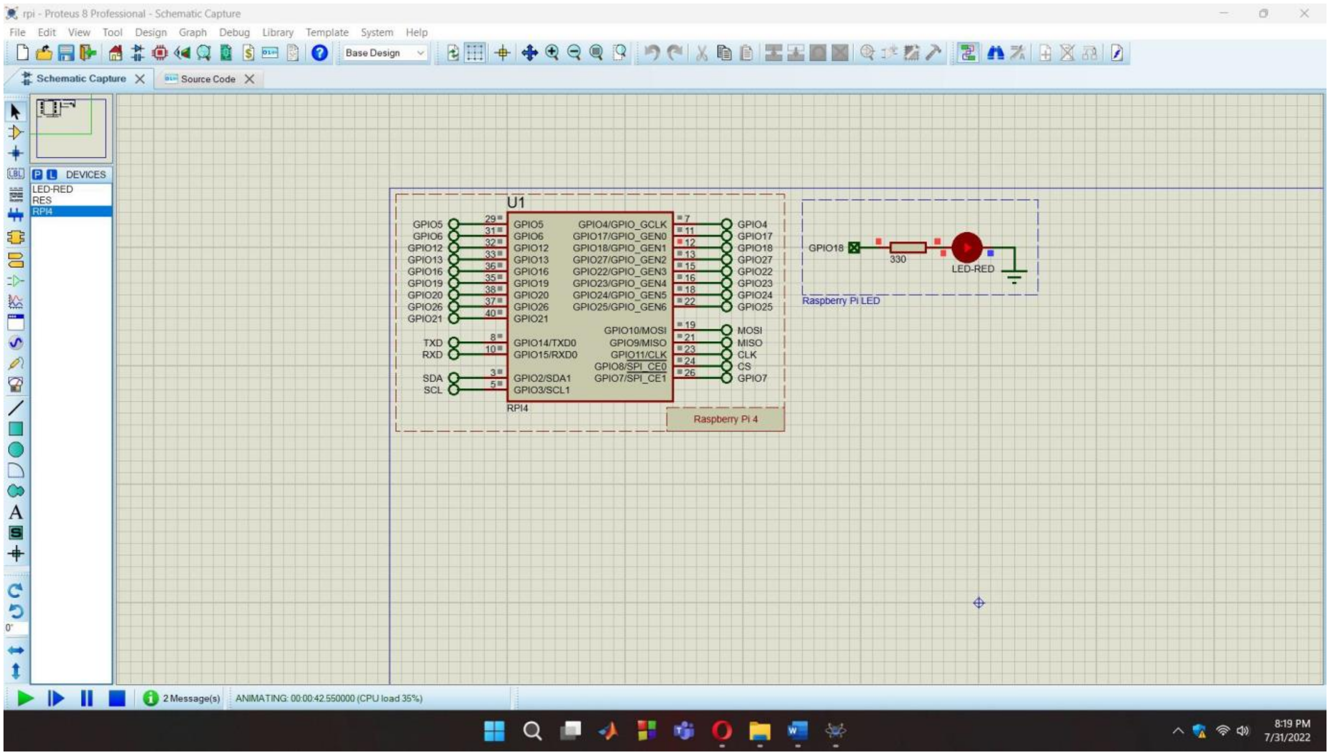

Figure: LED ON

Lab Simulation:

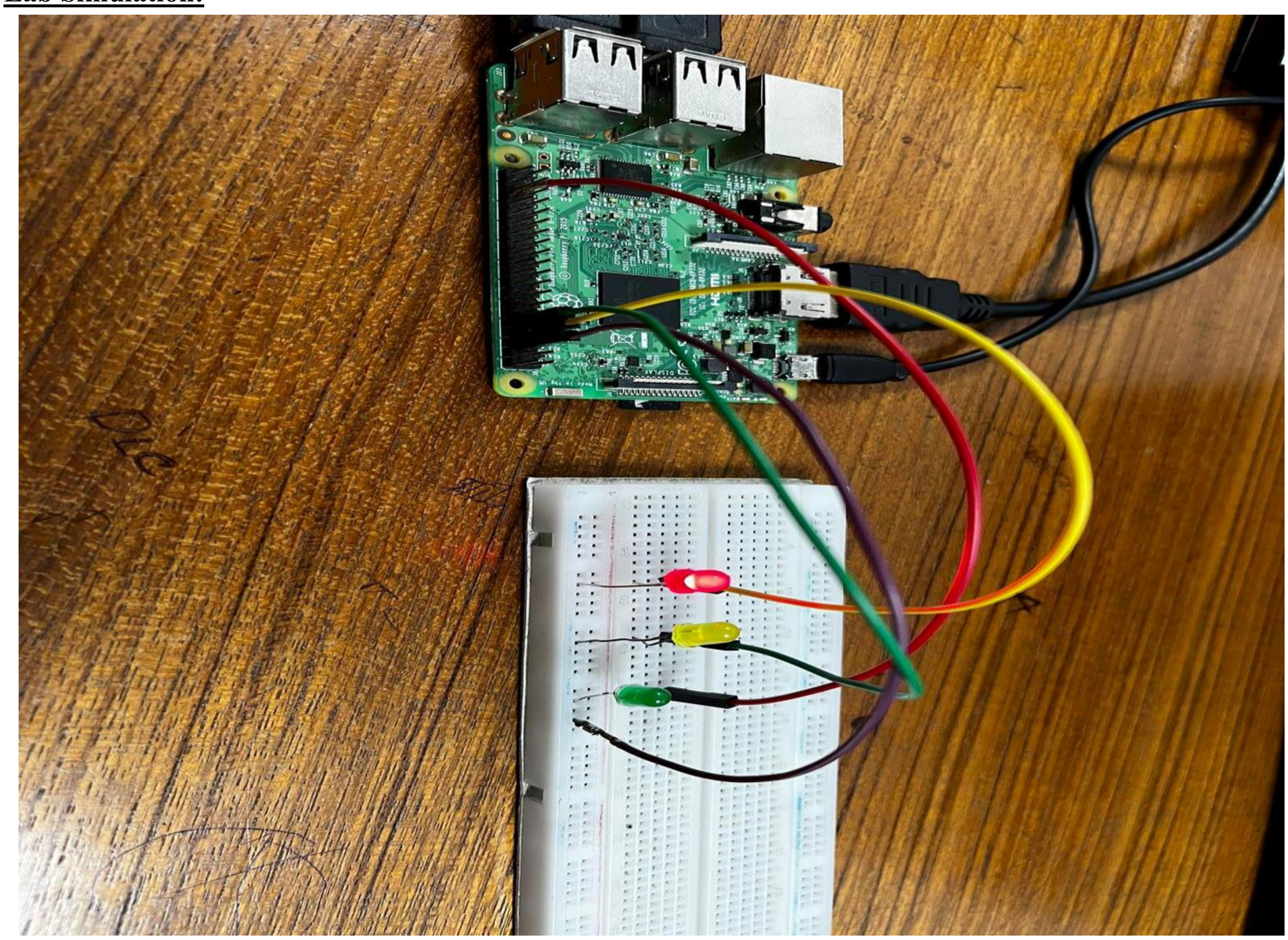

Figure: Red light is ON

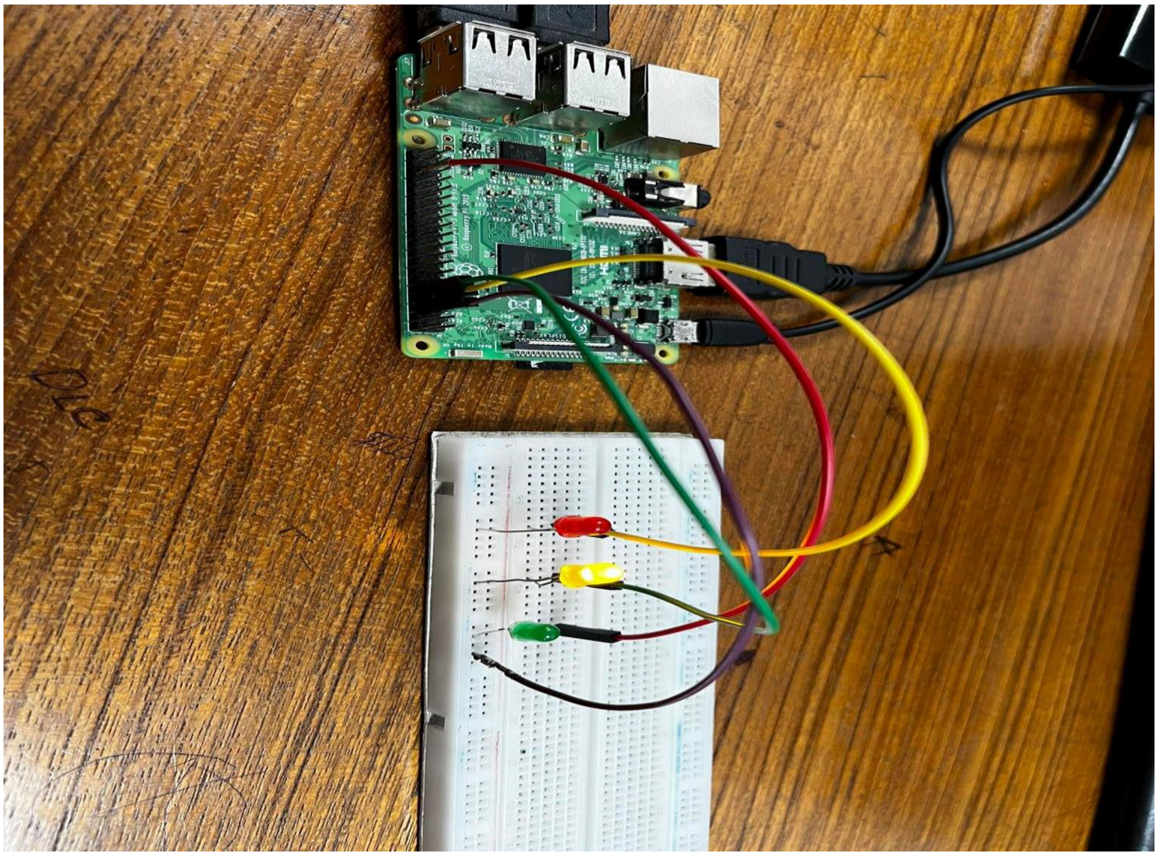

Figure: Yellow light is ON

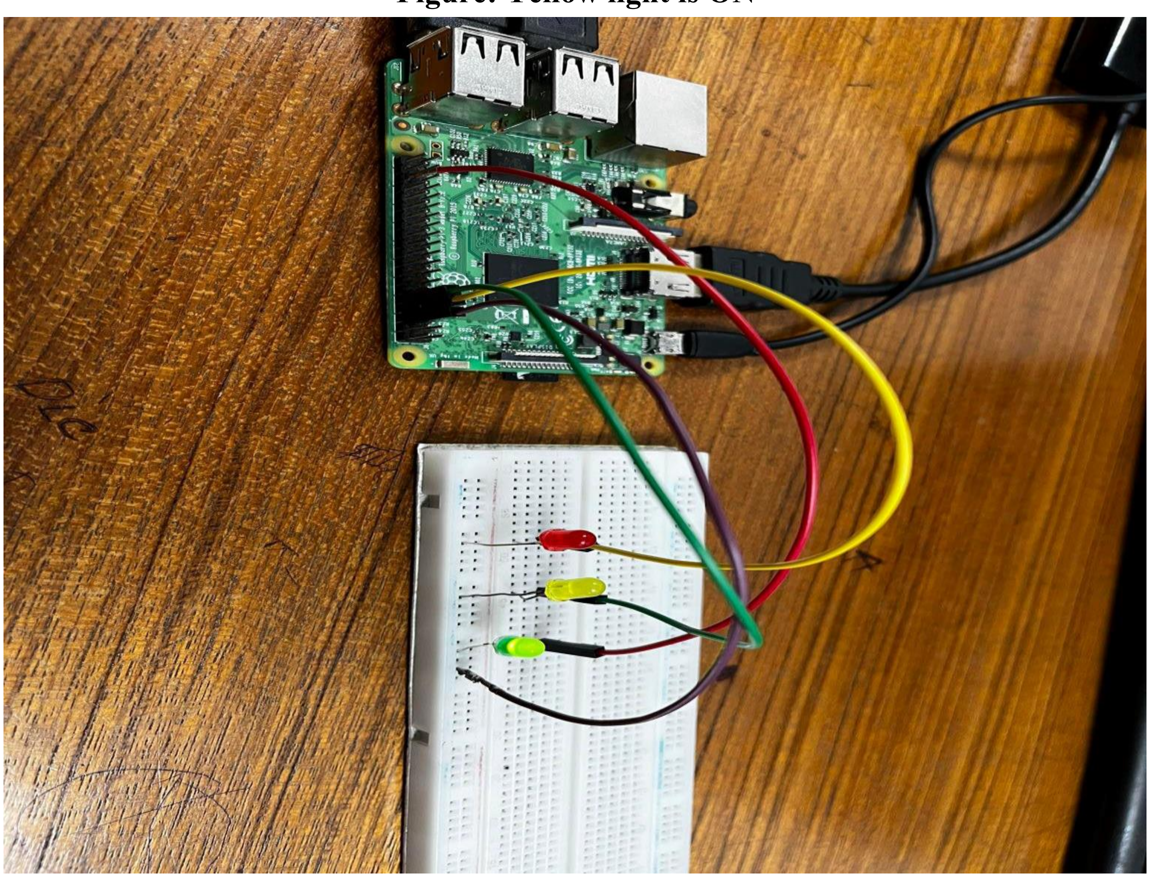

Figure: Green light is ON

Report:

All codes, scripts and proteus simulation of the blink program and traffic light system is attached above.

Discussion:

Some advanced features in proteus proffessional 8.9 may not be included in earlier versions. Therefore, in order to use all the functions, we must download and install the 8.9 version.. Developing a new project is crucial. If we don't set up the ideal conditions for our desired endeavor, we won't ever receive the results we want. phython code is used to develop every project on the Raspberry Pi, So we must be careful while using certain sytaxes. Because proteus 8.9 does not check this, we must determine whether any pins have been issued more than once after opening a new component. The same pin may be assigned to two separate components. Therefore, before simulating the project, we must consider that for the temporary delay.

Conclusion:

This experiment provided a basic understanding of Raspberry Pi, its components, and how to set up with an operating system. Additionally the experiment demonstrated how to control an LED using phython programming language with the GPIO Pin of the Raspberry Pi. It is important to note that Raspberry Pi can be used for various applications such as automation, data logging, and robotics. with the knowledge gained from this experiment, one can explore and develop more complex projects with Raspberry Pi.