AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Engineering

Lab Report

Experiment # 10

Experiment Title: Familiarization with Raspberry Pi

Course Title:	MICROPROCESSOR AND EMBEDDED SYSTEMS LAB		
Course Code:	COE3104	**Section:**	A
Semester:	Spring 2022-23	**Degree Program:**	BSc in CSE/BSc in EEE
Course Teacher:	**Md. Ali Noor**		

Declaration and Statement of Authorship:

1. I/we hold a copy of this Assignment/Case-Study, which can be produced if the original is lost/damaged.
2. This Assignment/Case-Study is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgment is made.
3. No part of this Assignment/Case-Study has been written for me/us by any other person except where such collaboration has been authorized by the concerned teacher and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared, and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the Faculty Member for review by any internal/external examiners.
7. I/we understand that Plagiarism is the presentation of the work, idea, or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offense that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual forms, including electronic data, and oral presentations. Plagiarism occurs when the origin of the source is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or copy my/our work.

** Student(s) must complete all details except the faculty use part.*
** Please submit all assignments to your course teacher or the office of the concerned teacher.

Group # 06

Sl No	Name	ID	Program
1	Maimona Rahman Farjana	20-42954-1	CSE
2	Md. Sumon	20-42556-1	CSE
3	Md. Sajidul Haque Shohan	20-42022-1	CSE
4	Md. Rashedul Islam	20-43301-1	CSE
5	Syeda Aynul Karim	19-41829-3	CSE
6	Medha Chowdhury	20-41930-1	CSE
7	Syed Aftab Uddin	19-41522-3	EEE

Faculty use only

FACULTY COMMENTS	**Marks Obtained**	
	Total Marks	

Table of Contents

Experiment Title: Familiarization with the Raspberry Pi.

Introduction:

In this experiment, an LED will be controlled by using Raspberry Pi. Python will be used to blink an LED. This experiment will give a basic idea of Python language as well as import GPIO pins of Raspberry Pi. This experiment will be done from a Linux environment thus enabling to use of terminal and shell scripting. As Raspberry Pi runs on a Linux environment, it is always advised to use text editors, like G Vim, Nano Editor, Emacs Editor, and Pico Editor. However, when you installed your Raspbian it comes withIntegrated Development Environment (IDE) for Python.

Objectives:

The objectives of this experiment are to-

1. To be familiar with the hardware and specifications of Raspberry Pi
2. To demonstrate the basic functions of Raspberry Pi

Apparatus:

1) Activated Raspberry pi
2) LED
3) Resistor (220 Ω)
4) Breadboard
5) Jumper wires

Code of Program:

Tool setup:

1. Python is a build language in Linux on Raspberry Pi.
2. To create a Python file. Create a file with an extension (.py).
3. To type Python code, we will use the Linux default terminal here. To use terminal Home> menu> terminal.
4. To write the code.
5. To save the program, press “Ctrl+X” then “Y” then “enter”.
6. The LED will start blinking.

Short descriptions for the necessary functions:

import RPi.GPIO as GPIO _ This line tells the Python interpreter (the thing that runs the Python code) that it will be using a ‘library’ that will tell it how to work with the Raspberry Pi’s GPIO pins.
import time _ Imports the Time library so that we can pause the script later.

GPIO.setmode(GPIO.BCM) _ Each of the pin on the Raspberry Pi has several different names, so you need to tell the program which naming convention is to be used.

print "LED on" _ This line prints some information to the terminal.

GPIO.output(18,GPIO.HIGH) _ This turns the GPIO pin 'on'. What this means is that the pin is made to provide power of 3.3volts. This is enough to turn the LED in our circuit on.

time.sleep(1) _ Pause the Python program for 1 second

print "LED off" _ This line prints some information to the terminal.

GPIO.output(18,GPIO.LOW) _ The line turns the GPIO pin 'off', meaning that the pin is no longer supplying any power.

Hardware Setup:

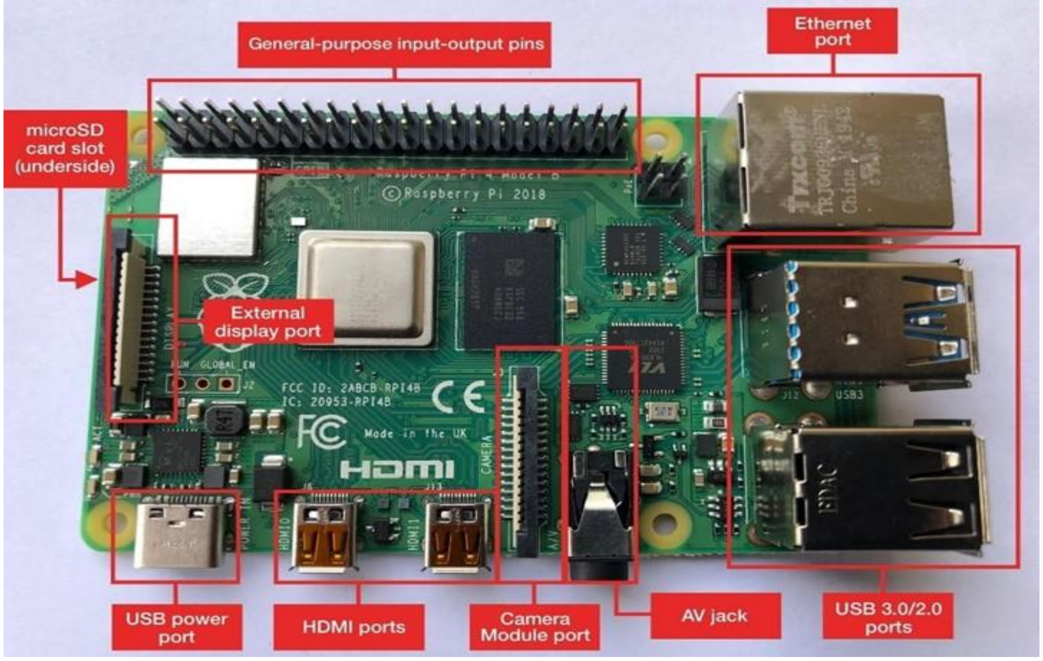


Figure 1: Raspberry Pi 3 - Model B

Circuit Setup:

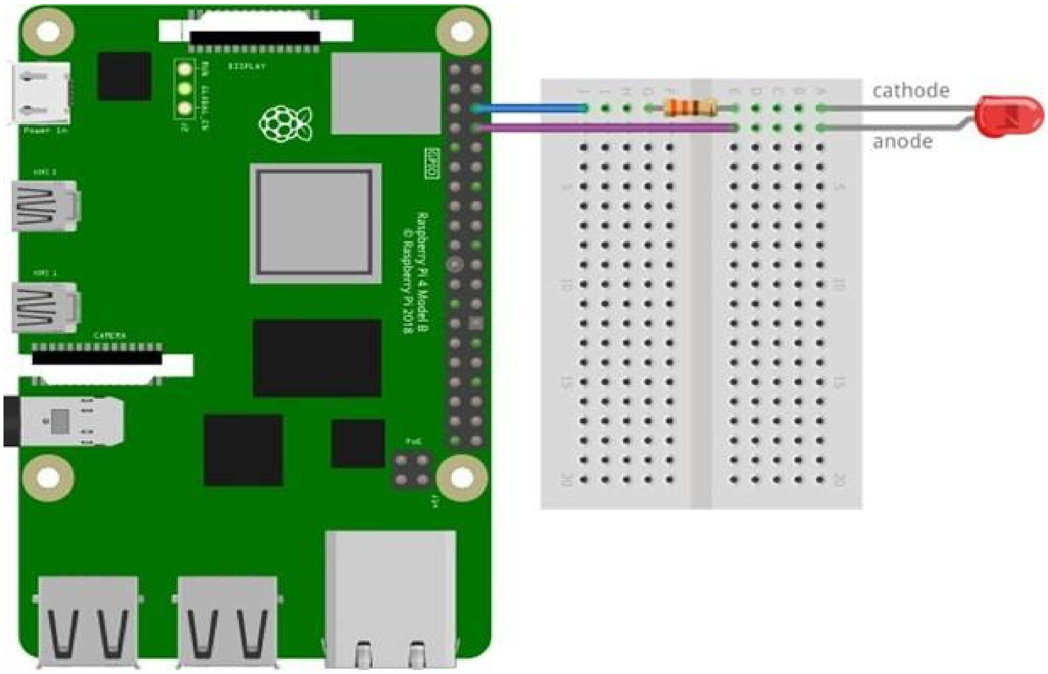
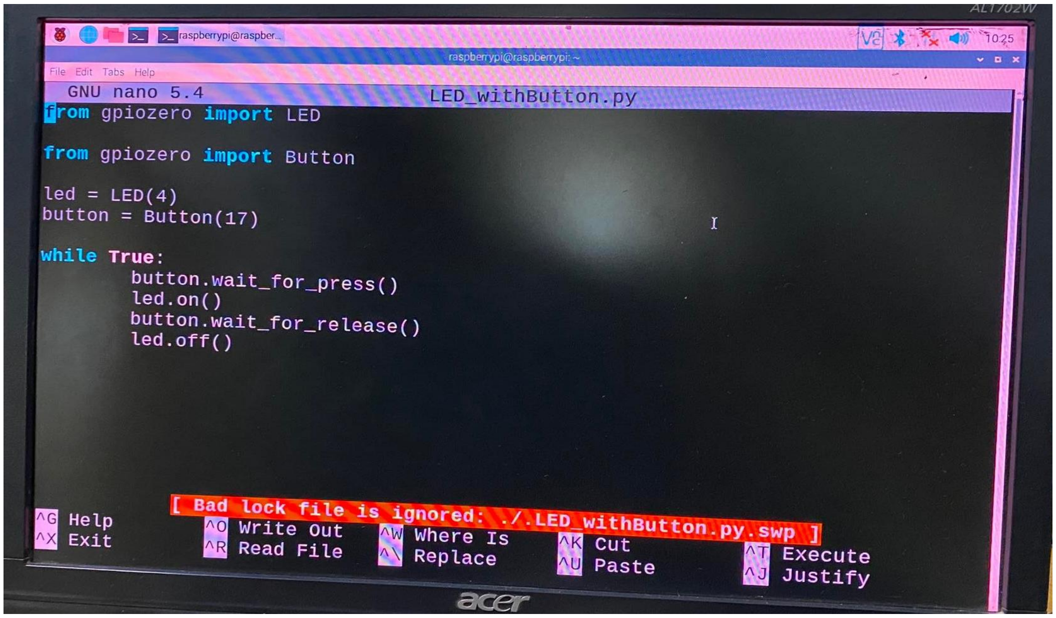


Figure 2: Circuit Setup of Raspberry Pi 3

Source Code for the program:

```
import RPi.GPIO as GPIO

import time



# seting-up the raspberrypi pins

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(14, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)
GPIO.setup(18,  GPIO.OUT)



while (True):

    GPIO.output (14, GPIO.HIGH)
    print("Green  ON")
    time.sleep(6)
    GPIO.output (14, GPIO.LOW)
    print("Green OFF")

for i in range(7):
   GPIO.output (15, GPIO.HIGH)
   Print  ‘Yellow
   Blink’time.sleep(1)
   GPIO.output (15, GPIO.LOW)
   time.sleep(1)

   GPIO.output (18, GPIO.HIGH)
   print ("Red ON")
   time.sleep(3)
   GPIO.output (18, GPIO. LOW)
```



```
  GNU nano 5.4                      LED_withButton.py
from gpiozero import LED

from gpiozero import Button

led = LED(4)
button = Button(17)

while True:
        button.wait_for_press()
        led.on()
        button.wait_for_release()
        led.off()




              [ Bad lock file is ignored: ./.LED_withButton.py.swp ]
^G Help          ^O Write Out    ^W Where Is     ^K Cut          ^T Execute
^X Exit          ^R Read File    ^\ Replace      ^U Paste        ^J Justify
```

Figure 3: Code on the terminal

Simulation Output Results:

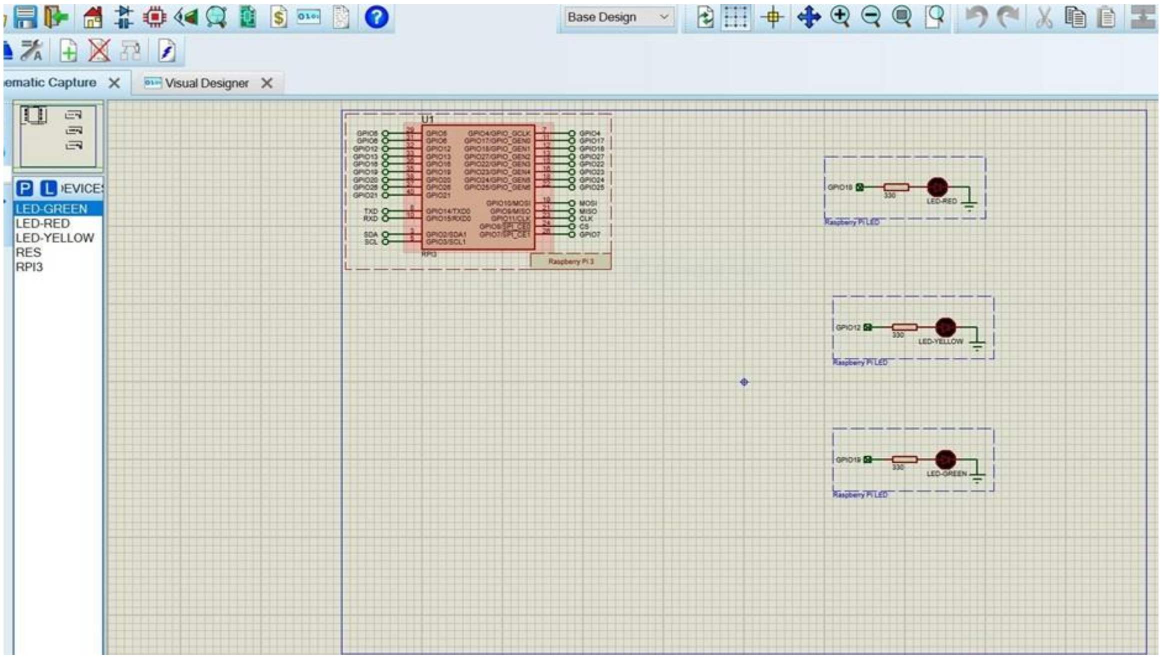


Figure 4: Traffic light management system (All LED off)

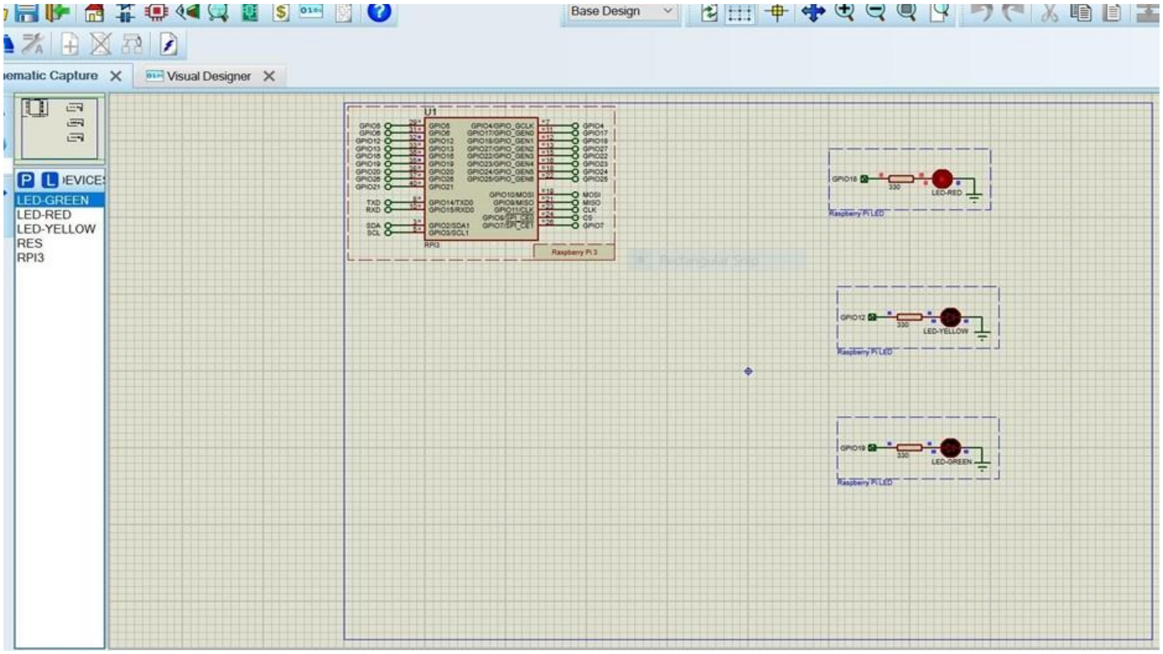


Figure 5: Traffic light management system (RED LED turn on)

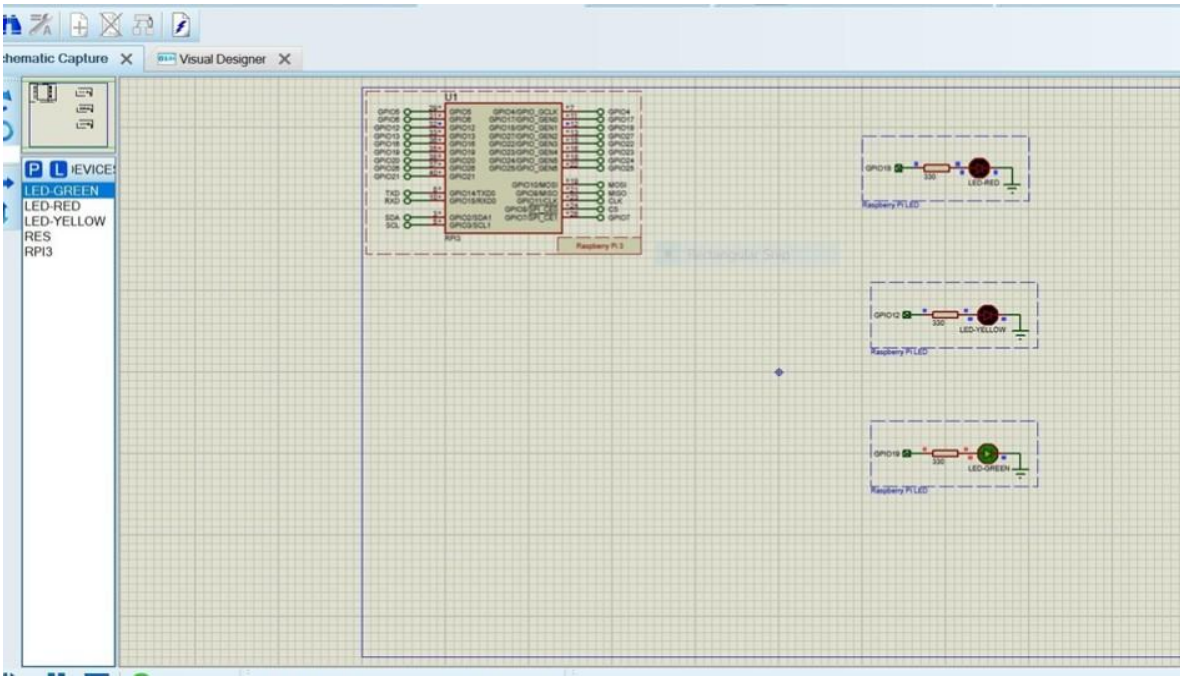


Figure 6: Traffic light management system (Green LED turn on)

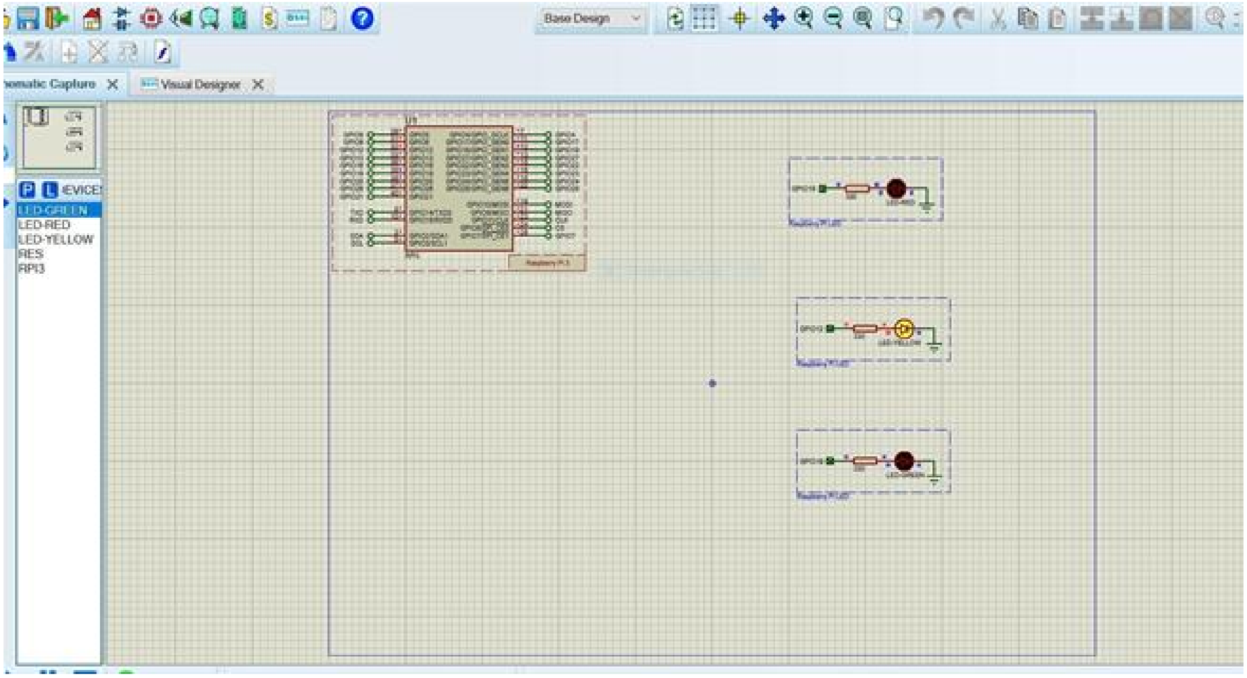


Figure 7: Traffic light management system (Yellow led turned on)

Hardware Output Results:

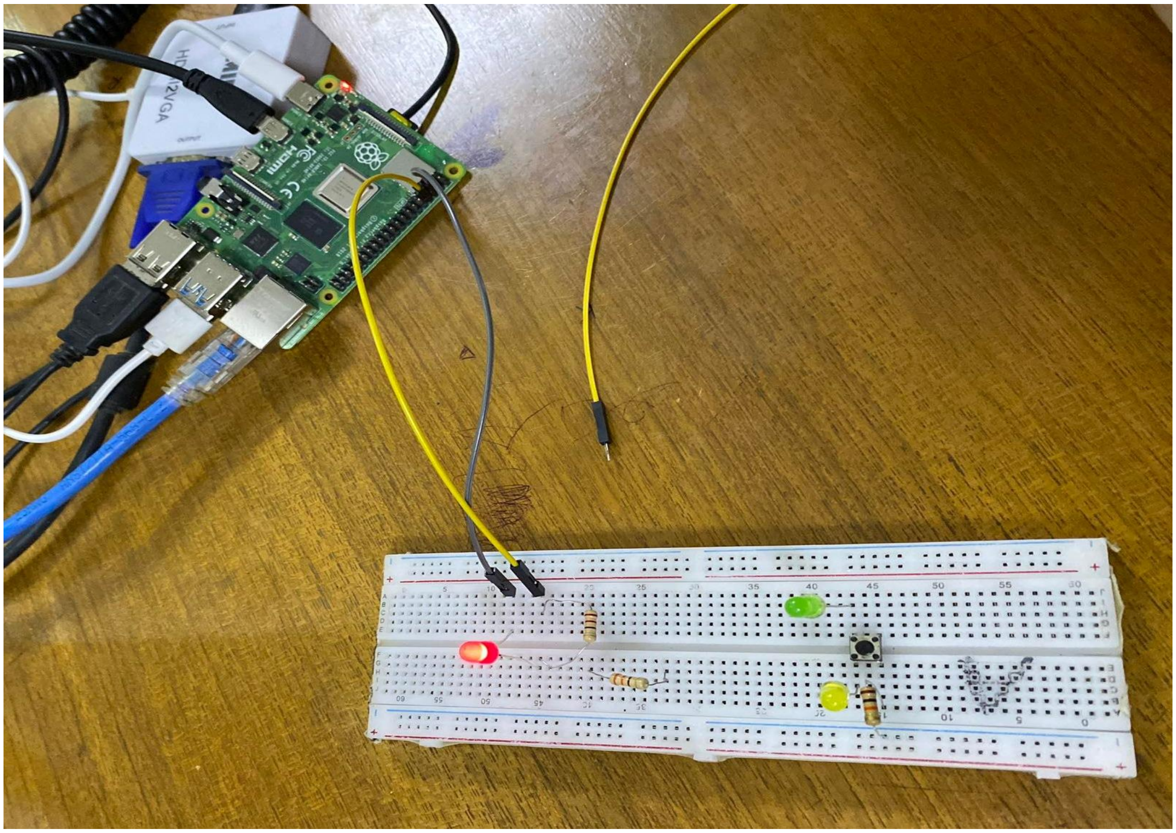


Figure 8: Traffic light management system

Discussion & Conclusion:

This experiment was done in two ways. First, using three colored animated LED lights (Red, Yellow, and Green) three resistors' breadboards, and connecting wires with a Raspberry Pi. Python is used to blink the LED. After that, the LED lights (Red, Yellow, and Green) were connected to GPIO14, GPIO15, *and* GPIO18 from the Raspberry Pi board. Then some code was written for Python, we will use the Linux default terminal here. Then the Raspberry pi board was connected to the computer and the code was run to get blinking the lights. This experiment provides a basic introduction to the Python language and how to import GPIO pins from a Raspberry Pi. This experiment is performed in a Linux environment with terminal and shell scripting available. Another way this experiment was done was with the help of Proteus software. At the time of doing this experiment, some issues were faced. The issues were some pin configuration mistakesand some errors in code. But these issues were solved with the help of a lab manual and the internet. In bothways, the result of the experiment was obtained successfully.

References:

1) Raspberry pi datasheet.