Name :- Fokane Sakshi Anil
Class :- TE - A
Roll No :- 42

Assignment No :- 8

Aim :-

Write an application using Raspbery-Pi/ Beagle board to control the operation of a hardware simulated traffic signal.

Theory :-

Attaching the traffic lights :-

The low voltage labs traffics lights connect to the Pi using four Pins. One of these needs to be ground, the other three being actual GPIO pins used to control each of the individual LEDs.

Before powering up the Pi, attach the traffic lights so that the pins connect to the GPIO pins highlighted in red:

Programming the Traffic lights :-

First, you need to install a couple of extra software packages needed to allow you to download my sample code, & to give python access to the GPIO pins on the Pi. Enter the following at the command line :

```
Sudo apt-get install python-dev
python-rpi.gpio git
```

How it works:

The code for this is very simple. It starts by importing the Rpi. GPIO library, plus time which gives us a timed wait function, signal that allows us to trap the signal sent when the user tries to quit the program & sys so we can send an appropriate exit signal back to O.S before terminating.

```
import RPi.GPIO as GPIO
import time
import signal
import sys.
```

Next we put the GPIO library into "BCM" or "Broadcom" mode & sets pins 9, 10 & 11 to be used as outputs:

```
#setup
GPIO.setmode(GPIO.BCM)
GPIO.setup(9, GPIO.OUT)
GPIO.setup(10, GPIO.OUT)
GPIO.setup(11, GPIO.OUT)
```

The main part of the program will run in infinite loop until the user exists it by stopping Python with Ctrl C. It's a good idea to add a handler function that will run whenever that happens, so that we can turn off all the lights prior to exiting:

```
#Turn off all lights when user
ends demo
def allLightsOff(signal, frame):
 GPIO.output(9, False)
 GPIO.output(10, False)
 GPIO.output(11, False)
  GPIO.cleanup()
  Sys.exit(0)
signal.signal(signal.SIGINT, allLig-
htsoff)
```

The main body of the code then consists of an infinite while loop that turns on the red light, waits, turns on the amber light, waits, then cycles through the rest of the traffic light pattern by turning the appropriate LED's on & off.

When Control-C is pressed an interrupt signal. SIGINT is sent This is handled by the all lights Off function that switches all the lights off, tidies up the GPIO library state & exits cleanly back to the operating System.

Conclusion :-

Thus, we have implemented the appl^n for traffic Signals using RaspberryPi.