Ankita Bonde
TE A 19

Assignment : 8

Aim: Write application using Raspberry-Pi/ Beagle board to control the operation of a H/w simulated traffic signal

Theory

① Attaching the Traffic Light

The low Voltage labs traffic lights connect to the Pi using 4 pins. One of these need to be ground the other 3 being actual GPIO pin used to control each of individual LEDS

- Before powering up pi attach the traffic lights so that the pins connect to the GPIO pin highlighted in red

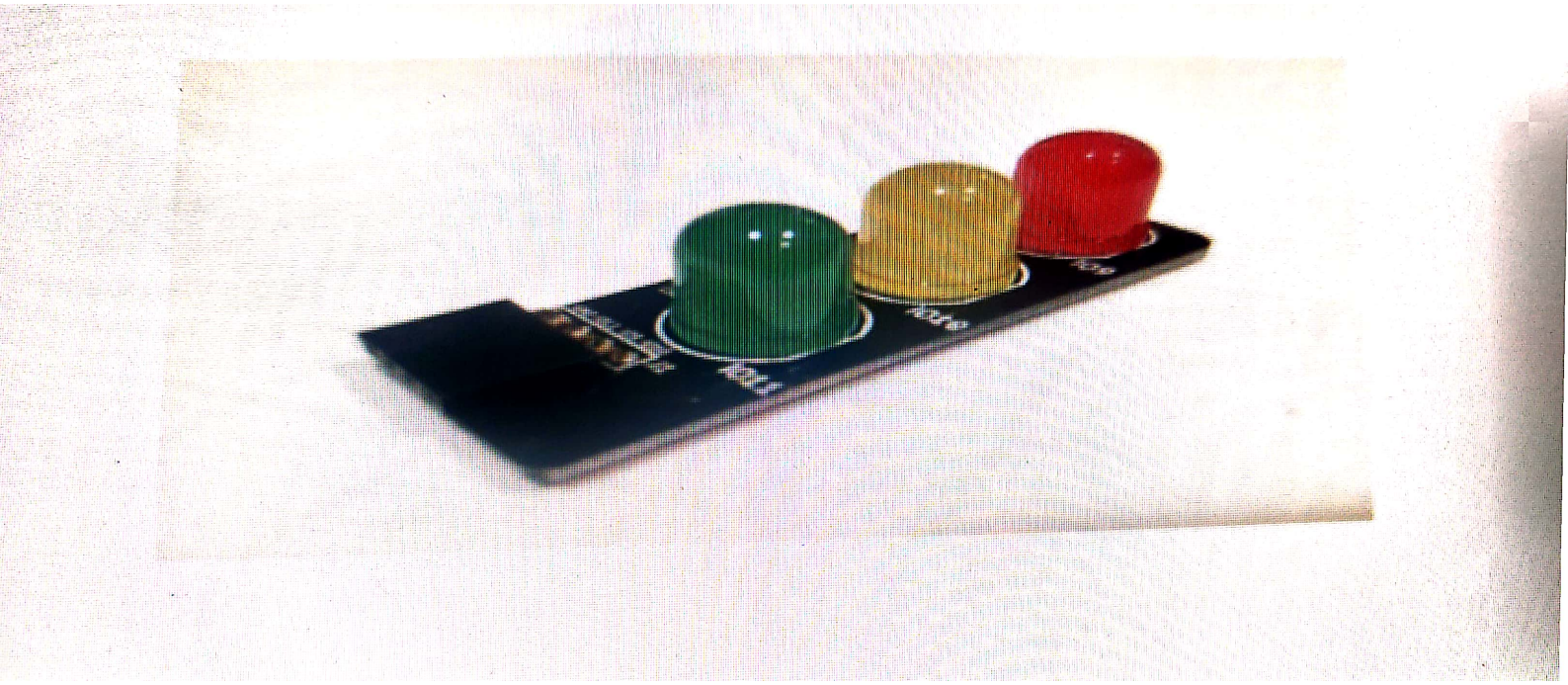
② Programing the Traffic lights

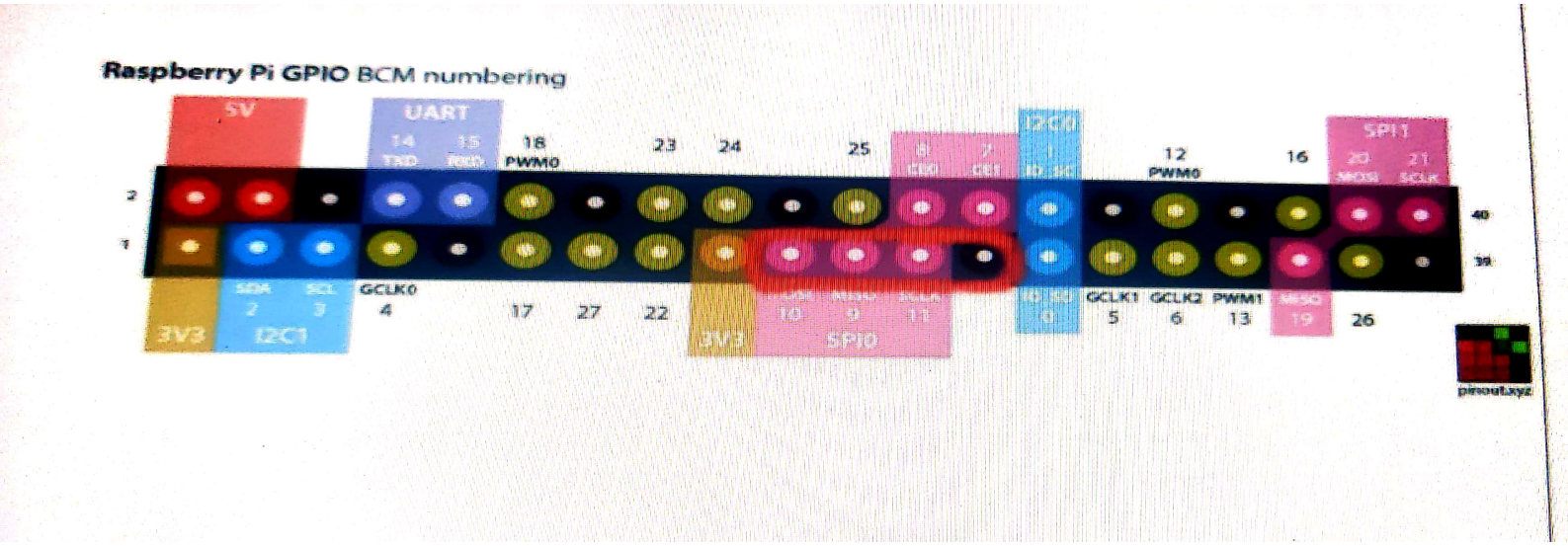
- First you need to install a couple of extra s/w packages needed to allow you to download my sample code and to give Python acess to the GPIO pins on the Pi Enter the following at command line

sudo-apt-get install python-dev python-rpi.gpio.git

③ How it works

- The code for this is very simple. It starts by importing Rpi GPIO library plus time which gives us a timed wait function.





signal that allows us to trap signal sent when the user tries to quit the prog and sys so we can send an appropriate exit signal back to the OS before terminating

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

Next we put the GPIO library into BCM or Broadcom mode and set pins 9, 10 and 11 as o/p

④ Setup

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

The main part of prog will ~~return~~ run in infinite loop until the user exits it by stopping Python with ctrl c. Its a good idea to add a handler function that will run whenever this happens so that we can turn off all the lights prior to exiting.

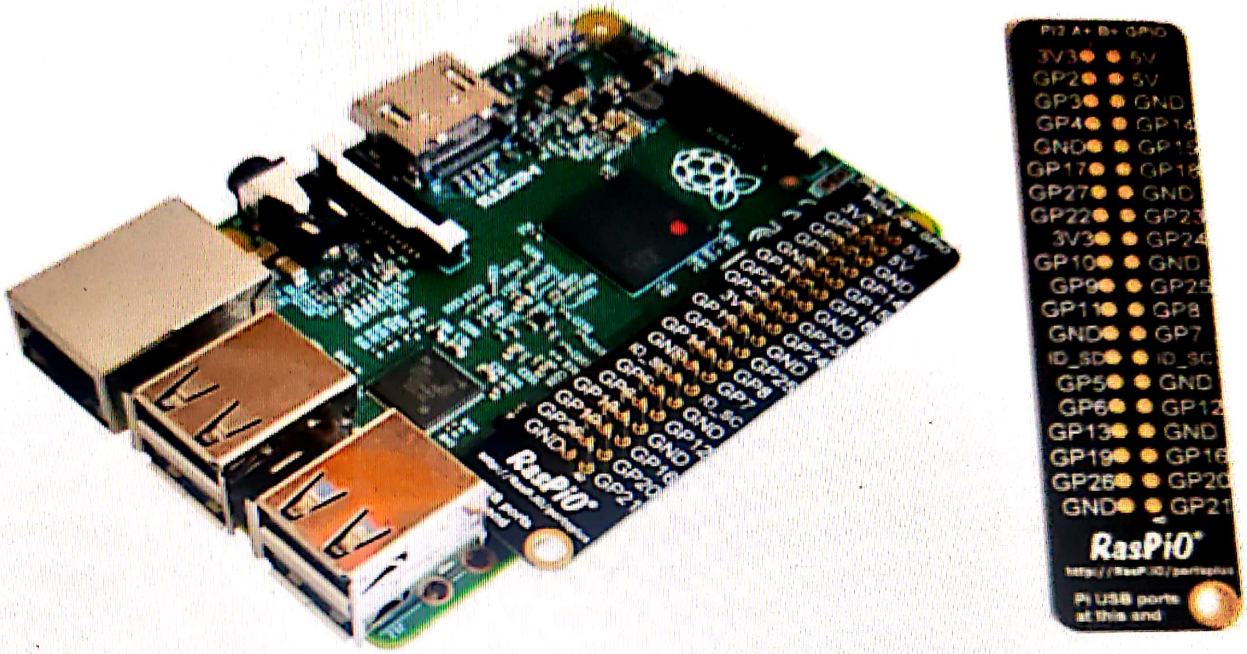
```
# Turn off lights when user ends demo.

def allLightsOFF(signal, frame):
    GPIO.output(9, False)
    GPIO.output(10, False)
    GPIO.output(11, False)
    GPIO.cleanup()
```

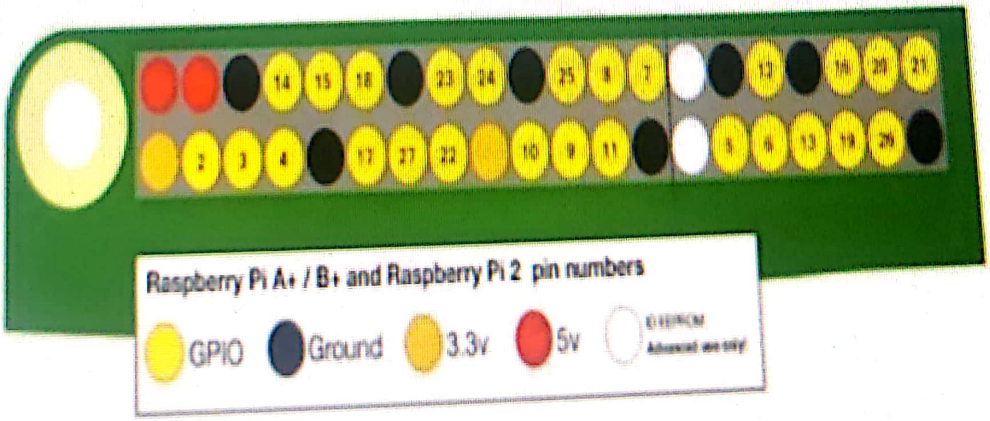
sys exit (0)
signal. signal (signal SIGINT, allLightsOff)

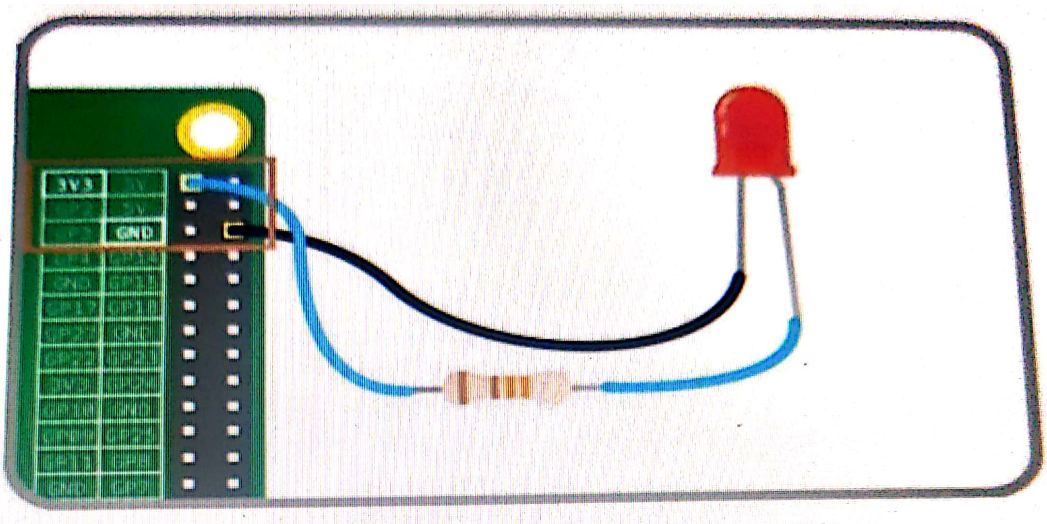
the main body of the code then consists of an infinite while loop that turns on the red light waits and turns on the pin10 waits then cycles and through the rest of the traffic light pattern by turning the appropriate LEDS on and off
when Ctrl c pressed an interrupt signal SIGINT is sent
This is handled by the all Lights Off function that switches all the light off tidies up the GPIO library state and exists cleanly back the OS

Conclusion
Thus we have implemented the application for traffic signals using Raspberry Pi



If you don't have a pin label, then this can help you to identufy the pin numbers





f you connect special purpose(GPIO) pin are connect to long leg, make sure to write the im