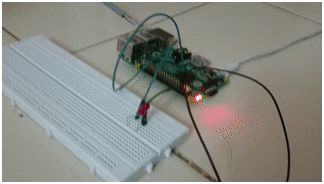
**LED BLINKING USING RASPBERRY PI**

**PYTHON CODE:**

|  |
| --- |
|  |
|  | import RPi.GPIO as GPIO # RPi.GPIO can be referred to as GPIO from now |
|  | import time |
|  |  |
|  | def setup(): |
|  | GPIO.setmode(GPIO.BOARD) # GPIO Numbering of Pins |
|  | GPIO.setup(ledPin, GPIO.OUT) # Set ledPin as output |
|  | GPIO.output(ledPin, GPIO.LOW) # Set ledPinto LOW to turn Off the LED |
|  |  |
|  | def loop(): |
|  | while True: |
|  | print 'LED on' |
|  | GPIO.output(ledPin, GPIO.HIGH) # LED On |
|  | time.sleep(1.0) # wait 1 sec |
|  | print 'LED off' |
|  | GPIO.output(ledPin, GPIO.LOW) # LED Off |
|  | time.sleep(1.0) # wait 1 sec |
|  | def endprogram(): |
|  |  |
|  | GPIO.output(ledPin, GPIO.LOW) # LED Off |
|  | GPIO.cleanup() # Release resources |
|  |  |
|  | if \_\_name\_\_ == '\_\_main\_\_': # Program starts from here |
|  | setup() |
|  | try: |
|  | loop() |
|  | except KeyboardInterrupt: # When 'Ctrl+C' is pressed, the destroy() will be executed. |
|  | endprogram() |
|  |  |
|  |  |



**TRAFFIC LIGHT using RASPBERRY PI**

**PYTHON CODE:**

fromgpiozero import LED

from time import sleep

red=LED(22)

orange=LED(27)

green=LED(17)

while True:

red.on()

sleep(1)

orange.on(1)

sleep(1)

green.on()

sleep(1)

red.off()

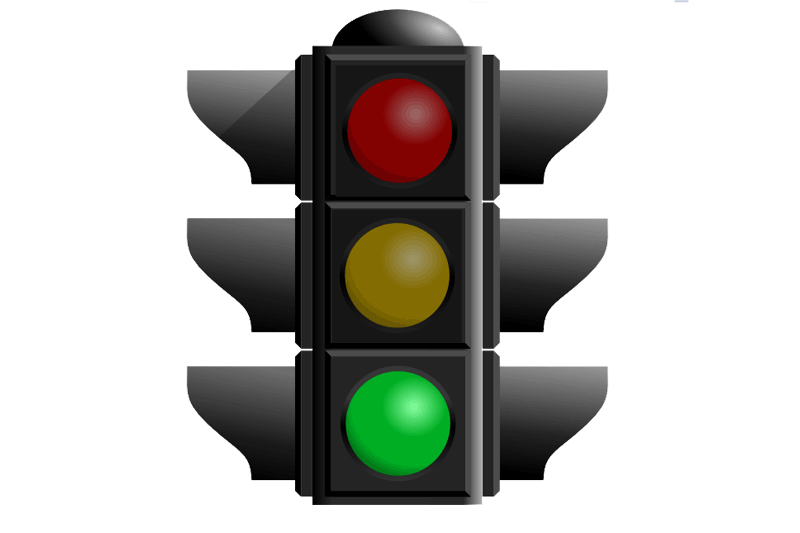
sleep(1)

orange.off()

sleep(1)

green.off()

.



SUBMITTED BY DURGA DEVI.T