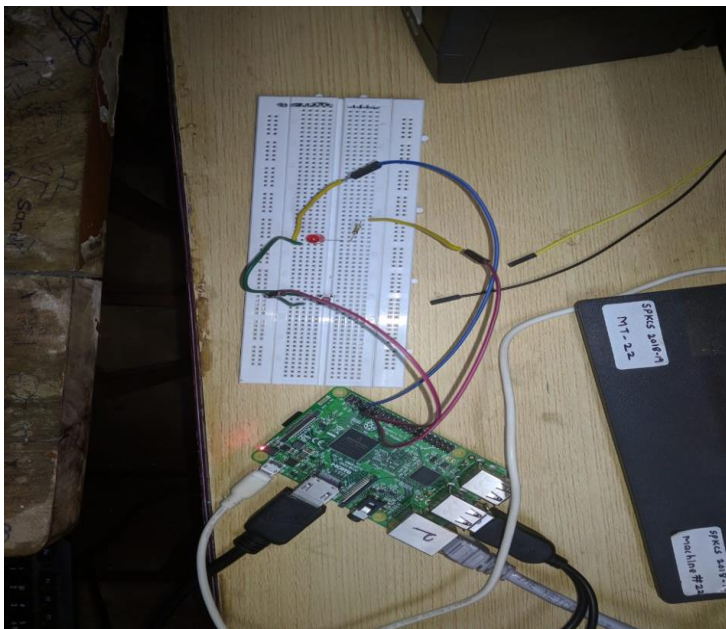# Practical No. 6

Aim: Setup a TCP server and client on a raspberry pi using Python modules to send messages and execute shell commands from within python such as starting another application

Step 1: Install socket

Step 2: connect led to breadboard and boot raspberry

Step 3: Setup TCP server and run the script

```
*tcp_server.py - I:\pract6\tcp_server.py (3.7.3)*
File  Edit  Format  Run  Options  Window  Help
import socket
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(11,GPIO.OUT)
GPIO.setwarnings(False)

UDP_IP='192.168.0.80'#server ip address
UDP_PORT=5007
sock = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
sock.bind((UDP_IP,UDP_PORT))

while True:
    data,addr = sock.recvfrom(1024)
    d = int(data)
    if d==1:
        GPIO.output(11,GPIO.HIGH)
    elif d==0:
        GPIO.output(11,GPIO.LOW)

    print("Receiver Message : ",data)
```

Step 4: Setup TCP client and run the script

```
*tcp_client.py - I:\pract6\tcp_client.py (3.7.3)*                    —   □   ×
File  Edit  Format  Run  Options  Window  Help
import socket
sock=socket.socket(socket.AF_INET,socket.SOCK_DGRAM)

UDP_IP='192.168.0.80'#server ip address
UDP_PORT=5007

while True:
    number=int(input("Enter num :"))
    sock.sendto(str(number).encode('UTF-8'),(UDP_IP,UDP_PORT))
    if number>1:
        break
```

Step 5: send input to the server

```
Shell

Python 3.7.3 (/usr/bin/python3)
>>> %cd /home/pi/Downloads
>>> %Run tcp_client.py

  Enter num :1
  Enter num :0
  Enter num :1
  Enter num :0
  Enter num :
```

Step 6: view output