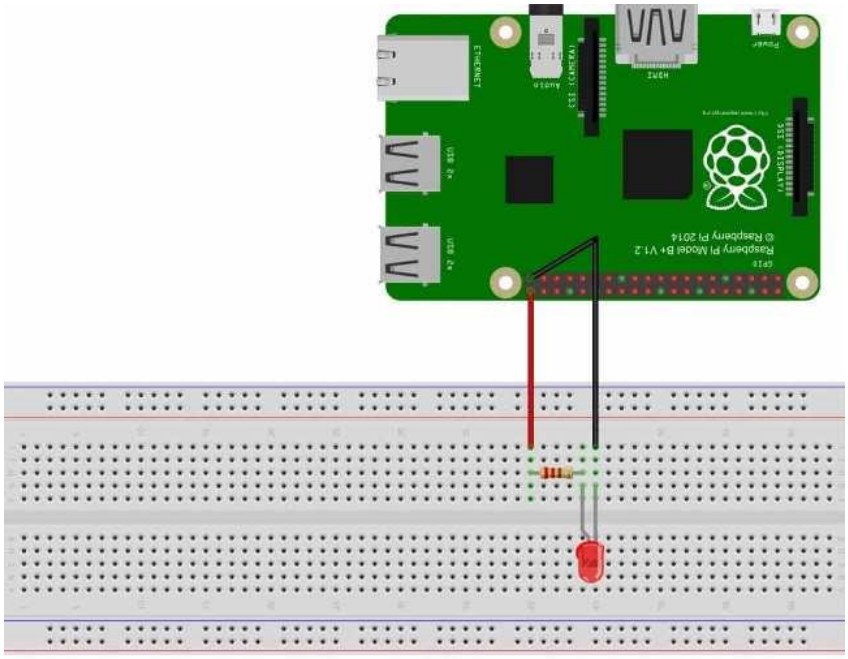
Ex. No : 8 Interface Bluetooth with Raspberry Pi, write a program to turn LED ON/OFF when '1'/'0' is received from smartphone using Bluetooth.

Program:

```
import bluetooth
import RPi.GPIO as GPIO        #calling for header file which helps in using GPIOs of PI

LED=21
GPIO.setmode(GPIO.BCM)     #programming the GPIO by BCM pin numbers. (like PIN40 as GPIO21)
GPIO.setwarnings(False)
GPIO.setup(LED,GPIO.OUT)  #initialize GPIO21 (LED) as an output Pin
GPIO.output(LED,0)

server_socket=bluetooth.BluetoothSocket( bluetooth.RFCOMM )
port = 1
server_socket.bind(("",port))
server_socket.listen(1)

client_socket, address = server_socket.accept()
print("Accepted connection from ",address)

while 1:
        data = client_socket.recv(1024)
        print "Received: %s" % data
        if (data == "0"):    #if '0' is sent from the Android App, turn OFF the LED
                print ("GPIO 21 LOW, LED OFF")
               GPIO.output(LED,0)
        if (data == "1"):    #if '1' is sent from the Android App, turn OFF the LED
                print ("GPIO 21 HIGH, LED ON")
                GPIO.output(LED,1)
        if (data == "q"):
                print ("Quit")
                break
client_socket.close()
server_socket.close()
```

Circuit Diagram:

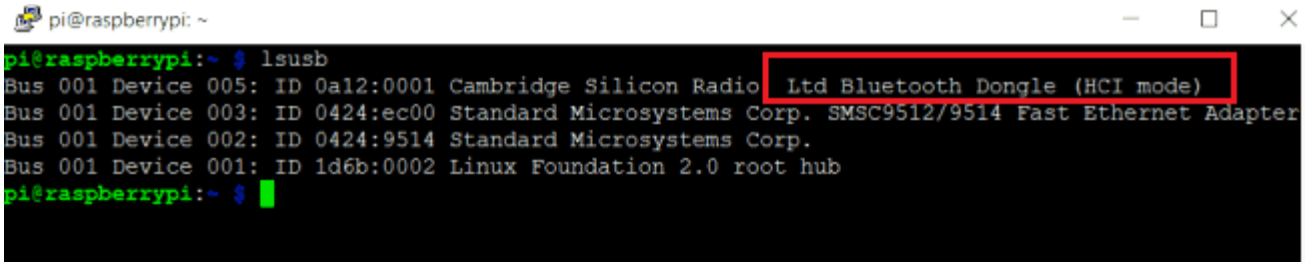


Installing Required Packages for Bluetooth Communication:

- Install some softwares for setting up Bluetooth communication in Raspberry Pi.
- First need to update the Raspbian using below commands:
  sudo apt-get update
  apt-get upgrade
- install few Bluetooth related packages:
  sudo apt-get install bluetooth blueman bluez
- Then reboot the Raspberry Pi:
  sudo reboot
- Finally we need python Library for Bluetooth communication so that we can send and receive data through RFCOMM using Python language:
  sudo apt-get install python-bluetooth
- Also install the GPIO support libraries for Raspberry Pi:
  sudo apt-get install python-rpi.gpio

Pairing Devices with Raspberry Pi over Bluetooth:

- Check **Android Smart phone with Raspberry Pi** is paired or not by using following command **lsusb**



- Next “bluetoothctl” to manage our Bluetooth devices and open the bluetoothctl utility by below command:
  sudo bluetoothctl
- check all the commands of bluetoothctl utility or enter below commands in given order:

  [bluetooth]# power on

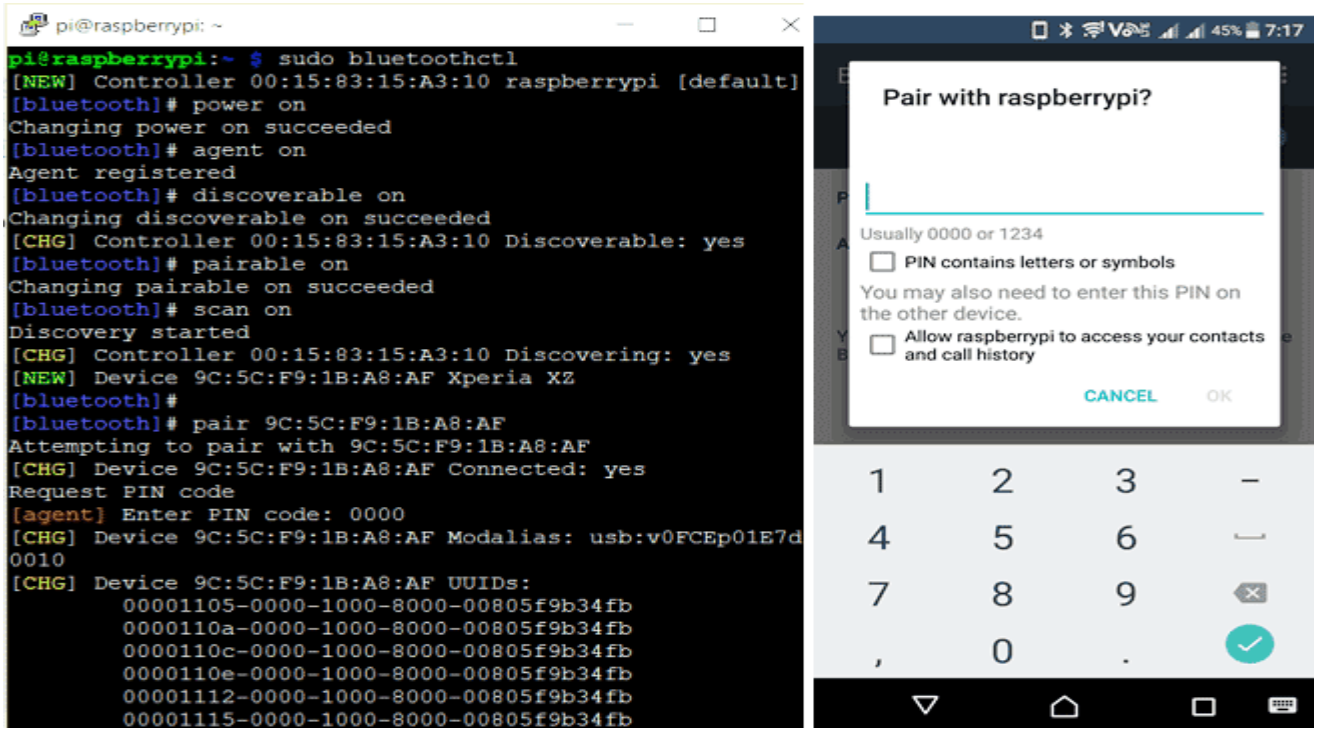
  [bluetooth]# agent on
  [bluetooth]# discoverable on

[bluetooth]# pairable on
[bluetooth]# scan on

- After the last command *"scan on",* Bluetooth device (Mobile phone) will be displayed in the list. Then copy the MAC address of the device and pair it by using given command:

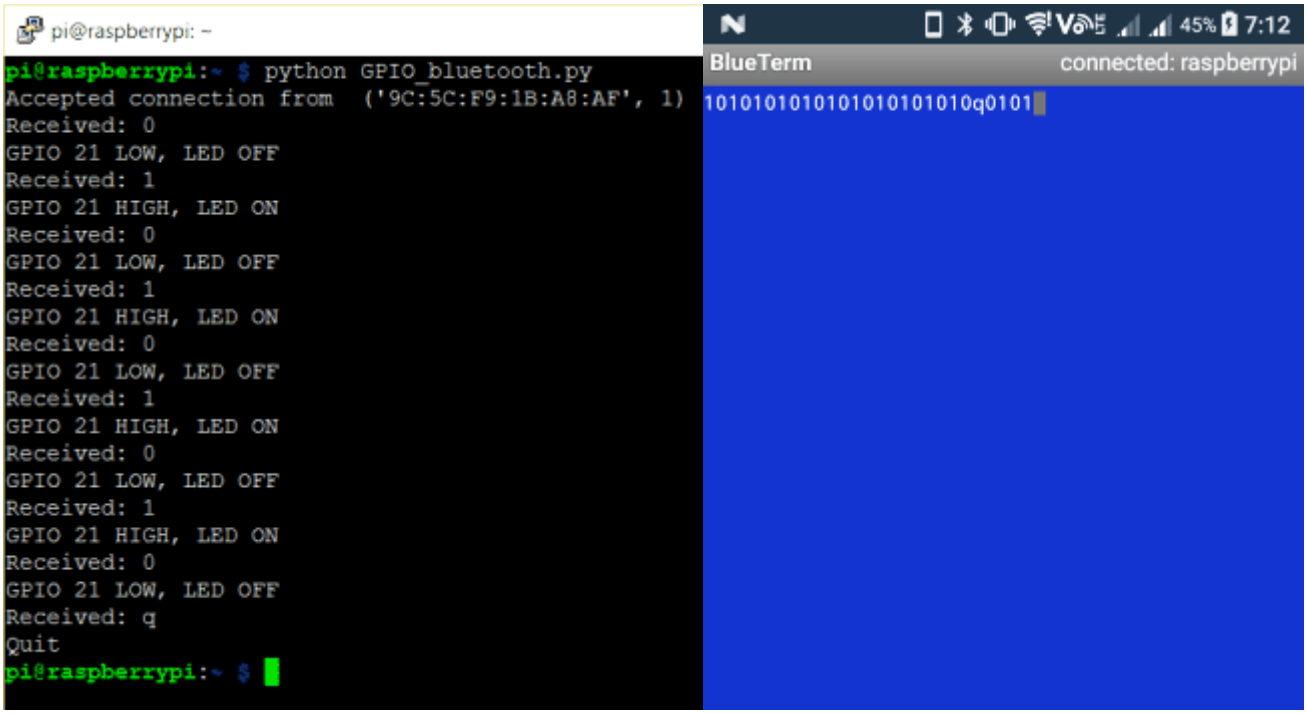
  pair <address of your phone>

- It will be prompted for Passcode or Pin in the Terminal console then type passcode there and press enter. Then type the same passcode in the mobile phone when prompted and now successfully paired with Raspberry Pi.



Controlling LED with Android App BlueTerm:

- Install a Android App for communicating with Raspberry Pi using a Bluetooth Serial Adapter. RFCOMM/SPP protocol emulates serial communication over Bluetooth, BlueTerm App which supports this protocol.
- Also use any other Bluetooth Terminal App which supports communication via RFCOMM socket.
- Now after downloading and installing the BlueTerm App, run the below given Python Program from the terminal and connect the paired raspberrypi device from the BlueTerm App at the same time.
- After successful connection you will see connected:raspberrypi at the top right corner of the App as shown below:



- Now you can just **enter '1' or '0' from the BlueTerm app to make the GPIO pin HIGH and LOW** respectively, which in turns switch ON and OFF the LED connected to this pin. Press 'q' to exit the program.