|  |
| --- |
| **Experiment-8**  **Demonstration of Communication Protocol using Bluetooth:**   1. **What are the applications of Bluetooth protocol?** 2. **What are Bluetooth profiles?** 3. **How is Bluetooth security implemented?** 4. **Can we use Bluetooth products on airlines? Justify your answer.** 5. **What are the improvements of BLE 4.0?**   **Lab Experiment:**  **Demonstration of Communication Protocol using Bluetooth**  **Code for Bluetooth Server:**  import Bluetooth  import sys  import time  sys.path.append('/home/pi/Adafruit-Raspberry-Pi-Python-Code-legacy/Adafruit MCP230xx"). server sock-bluetooth Bluetooth Socket(bluetooth RFCOMM)  port=1  server sockbind(["", port)) server sock listen(1)  client sock,address = server sock.accept()  print "Accepted connection from", address  from Adafruit MCP230XX import Adafruit\_MCP230XX  mcp = Adafruit MCP230XX(busnum = 1, address=0x20, num\_gpios = 16)  mcp.config(0, mcp.OUTPUT)  mcp.config(9, mcp.INPUT)  mcp.pullup(9, 1)  while True:  data= client\_sock.recv(1024)  print "received [s]" % data  if data == 'ON:  mcp.output(0,1)  print "LED ON"  if data == 'OFF':  mcp.output(0, 0)  print "LED OFF"  time sleep(0.5)  input= (mcp.input(9))  if input == 0:  text= "ON"  client\_sock.send(text)  if input==512:  text="OFF"  client\_sock.send(text)  time sleep(0.5)  client\_sock.close()  server\_sock.close()  **Code for Bluetooth Client:**  import bluetooth  import sys  sys path.append('/home/pi/Adafruit-Raspberry-Pi-Python-Code-legacy/Adafruit MCP230xx')  bd\_addr="B8:27:EB:87:EB:3E" #server MAC address  port=1  sock=bluetooth BluetoothSocket( bluetooth RFCOMM)  sock.connect((bd\_addr.port))  from Adafruit MCP230XX import Adafruit\_MCP230XX  import time  mcp=Adafruit\_MCP230XX(busnum= 1, address = 0x20, num gpios = 16)  mcp.config(0, mcp.OUTPUT)  mcp config(9, mcp.INPUT)  mcp pullup (9, 1)  while True:  input= (mcp input (9))  if input == 0:  text= "ON"  sock.send (text)  if input == 512:  text="OFF"  sock.send (text)  time.sleep(0.2)  data = sockrecv(1024)  print "received [%]" data  if data == 'ON":  mcp.output(0,1)  print "LED ON"  if data == 'OFF  mcp.output(0,0)  print "LED OFF"  time.sleep(0.2)  sock close() |
|  |
|  |
|  |