CHAPTER:15 WEBSERVER

PRACTICAL: 15A

AIM: Using ESP as Access Point Mode.

ARDUINO CODE :

```
/**********************
 * Author: Shreejicharan
 * Title: Using ESP as Access Point:
 * Date: 28/05/2017
 * Time: 7:00
 * Email: shreejicharanelectronics@gmail.com
 **********************/

/*Using ESP as Access Point: Use ESP module as access point and connect other device to it using WiFi */
#include <ESP8266WiFi.h>

const char* ssid     = "ESP8266";
const char* password = "12345678";

void setup()
{
  Serial.begin(9600);
  delay(3000);
  Serial.println("ESP As an Access Point Mode");
  WiFi.mode(WIFI_AP);
  WiFi.disconnect();
  delay(100);
  Serial.println("Configuring access point...");
  WiFi.softAP(ssid,password);
  IPAddress myIP = WiFi.softAPIP();
  Serial.println("AP IP address: ");
  Serial.println(myIP);
}

void loop() {
}
```

SIMULATION:

CHAPTER:15 WEBSERVER

PRACTICAL: 15B

AIM: Using ESP as Station Point Mode.

ARDUINO CODE :

```
/**********************
 * Author: Shreejicharan
 * Title: Using ESP as Station:
 * Date: 28/05/2017
 * Time: 7:00
 * Email: shreejicharanelectronics@gmail.com
 **********************/

/*Using ESP as Station: Use ESP module as station and connect to a router (or access point).
* station = client    */
#include <ESP8266WiFi.h>

const char* ssid     = "ssid";
const char* password = "password";

void setup()
{
  Serial.begin(9600);
  delay(3000);
  Serial.println("ESP As a Station Point Mode");
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(100);
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
            while (WiFi.status() != WL_CONNECTED) {
              delay(500);
              Serial.print(".");   }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
```

```
void loop()
{
}
```

SIMULATION:

CHAPTER:15 WEBSERVER

PRACTICAL: 15C

AIM: Design Webserver using IOT.

```
/**********************
 * Author: Shreejicharan
 * Title: Virtual Local Host based LED On Off and Sensor
 * Date: 27/05/2017
 * Time: 6:00
 * Email: shreejicharanelectronics@gmail.com
 **********************/

#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266mDNS.h>

MDNSResponder mdns;

// Replace with your network credentials
const char* ssid = "wifichat";
const char* password = "12345678";

ESP8266WebServer server(80);

String webPage = "";
```

```
int gpio0_pin = 0;

int gpio2_pin = 2;


void setup(void){

  webPage += "<h1>ESP8266 Web Server</h1><p>Socket #1 <a
href=\"socket1On\"><button>ON</button></a> <a
href=\"socket1Off\"><button>OFF</button></a></p>";

  webPage += "<p>Socket #2 <a href=\"socket2On\"><button>ON</button></a> <a
href=\"socket2Off\"><button>OFF</button></a></p>";


  // preparing GPIOs

  pinMode(gpio0_pin, OUTPUT);

  digitalWrite(gpio0_pin, LOW);

  pinMode(gpio2_pin, OUTPUT);

  digitalWrite(gpio2_pin, LOW);


  delay(1000);

  Serial.begin(115200);

  WiFi.begin(ssid, password);

  Serial.println("");


  // Wait for connection

  while (WiFi.status() != WL_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

  Serial.println("");
```

```
 Serial.print("Connected to ");
 Serial.println(ssid);
 Serial.print("IP address: ");
 Serial.println(WiFi.localIP());

 if (mdns.begin("esp8266", WiFi.localIP())) {
  Serial.println("MDNS responder started");
 }

 server.on("/", []{
  server.send(200, "text/html", webPage);
 });
 server.on("/socket1On", []{
  server.send(200, "text/html", webPage);
  digitalWrite(gpio0_pin, HIGH);
  delay(1000);
 });
 server.on("/socket1Off", []{
  server.send(200, "text/html", webPage);
  digitalWrite(gpio0_pin, LOW);
  delay(1000);
 });
 server.on("/socket2On", []{
  server.send(200, "text/html", webPage);
  digitalWrite(gpio2_pin, HIGH);
  delay(1000);
 });
```

```
  server.on("/socket2Off", [](){
    server.send(200, "text/html", webPage);
    digitalWrite(gpio2_pin, LOW);
    delay(1000);
  });
  server.begin();
  Serial.println("HTTP server started");
}

void loop(void){
  server.handleClient();
}
```

SIMULATION: