**Practical - 8**

**Aim:**

Simulate COAP protocols

**Components:** LED, Temperature sensor, Jumper wires, ESP 8266 NodeMCU

**Procedure:**

* Copper Plugin (Install on old Firefox version)
* Copper(Cu) plugin is not compatible with new Firefox.
* Install any old version of Firefox version 45 to 50
* Install CoAP Library (<https://github.com/automote/ESP-CoAP>)
* Connect Temperature sensor with ESP8266 (Diagram is same as previous practical)
* Code for Copper url access: (For CoAP Server)

In Arduino IDE Go to file -> examples -> ESP-CoAP simple library -> CoapServer

We run above code to get the IP address of ESP

Now, Run the URI coap://IPaddress:default port number/resource to connect to ESP.

* Code: (For GET and POST method)

Below two functions are used to control the actuators. The callback\_temp will be used with GET method to get temperature The callback\_LED will be used with POST method to Turn on or Turn off led

void callback\_temp(coapPacket \*packet, IPAddress ip, int port,int obs) {

Serial.println("temperature");

byte temperature;

byte humidity;

int chk = dht11.read(DHT11\_PIN, &temperature, &humidity, NULL);

String msg="real time temperature: ";

msg = msg + temperature;

msg = msg+" C ; real time Humidity: " ;

msg = msg + humidity ;

msg = msg + "%";

char message[58];

msg.toCharArray(message,58);

//coap.sendResponse(packet, ip, port, light);

if(obs==1)

coap.sendResponse(message);

else

coap.sendResponse(ip,port,message);

}

void callback\_LED(coapPacket \*packet, IPAddress ip, int port,int obs) {

char p[packet->payloadlen + 1];

memcpy(p, packet->payload, packet->payloadlen);

p[packet->payloadlen] = NULL;

Serial.println(p);

String message(p);

if (message.equals("0"))

{

digitalWrite(D5,LOW);

Serial.println("if loop");

}

else if (message.equals("1"))

{

digitalWrite(D5,HIGH);

Serial.println("else loop");

}

char \*light = (digitalRead(D5) > 0)? ((char \*) "1") :((char \*) "0");

//coap.sendResponse(packet, ip, port, light);

if(obs==1)

coap.sendResponse(light);

else

coap.sendResponse(ip,port,light);

}

void setup() {

coap.server(callback\_LED, "LED");

coap.server(callback\_temp, "Coap\_Temp");

coap.start();

}

void loop() {

coap.loop();

delay(1000);

}