Experimenting with teensy 4.0 and Wifi

Author:- Alok Kumar Mishra
(akmishra_99@yahoo.com)
Thursday, October 21, 2021

we are going to use

Arduino-Teensy4 - Teensy 4.0 Expansion Board

it is available at and for details of available features please see below

https://www.tindie.com/products/cburgess129/arduino-teensy4-teensy-40-expansion-board/

it has teensy 4.0 and esp based wifi. we are going to use ESPwifi library , and esp (ESP-12e) based wifi is connected to teensy 4.0 serial port 1 , we are going to use (following sketch is WebClientREpeating) which initializes esp based wifi on teensy 4.0 serial port 1 and fetches web page from server arduino.cc , edit following sketch/program and put your wifi SSID and passphrase and upload to teensy4.0 and after every ten second following (Figure 1) will be displayed on serial window.

sketch WebClientRepeating taken from arduino ide ESPwifi example

```
/*
 WiFiEsp example: WebClientRepeating

 This sketch connects to a web server and makes an HTTP request
 using an Arduino ESP8266 module.
 It repeats the HTTP call each 10 seconds.

 For more details see: http://yaab-arduino.blogspot.com/p/wifiesp.html
*/

#include "WiFiEsp.h"

// Emulate Serial1 on pins 6/7 if not present
#ifndef HAVE_HWSERIAL1
// #include "SoftwareSerial.h"
// SoftwareSerial Serial1(6, 7); // RX, TX
#endif

char ssid[] = "<SSID>";            // your network SSID (name)
char pass[] = "<PASSPHRASE>";        // your network password
int status = WL_IDLE_STATUS;     // the Wifi radio's status

char server[] = "arduino.cc";

unsigned long lastConnectionTime = 0;         // last time you connected to the server, in
milliseconds
```

```
const unsigned long postingInterval = 10000L; // delay between updates, in milliseconds

// Initialize the Ethernet client object
WiFiEspClient client;

void setup()
{
  // initialize serial for debugging
  Serial.begin(115200);
  // initialize serial for ESP module
  Serial1.begin(115200);
  // initialize ESP module
  WiFi.init(&Serial1);

  // check for the presence of the shield
  if (WiFi.status() == WL_NO_SHIELD) {
    Serial.println("WiFi shield not present");
    // don't continue
    while (true);
  }

  // attempt to connect to WiFi network
  while ( status != WL_CONNECTED) {
    Serial.print("Attempting to connect to WPA SSID: ");
    Serial.println(ssid);
    // Connect to WPA/WPA2 network
    status = WiFi.begin(ssid, pass);
  }

  Serial.println("You're connected to the network");

  printWifiStatus();
}

void loop()
{
  // if there's incoming data from the net connection send it out the serial port
  // this is for debugging purposes only
  while (client.available()) {
    char c = client.read();
    Serial.write(c);
  }

  // if 10 seconds have passed since your last connection,
  // then connect again and send data
  if (millis() - lastConnectionTime > postingInterval) {
    httpRequest();
  }
}

// this method makes a HTTP connection to the server
void httpRequest()
{
  Serial.println();
```

```
  // close any connection before send a new request
  // this will free the socket on the WiFi shield
  client.stop();

  // if there's a successful connection
  if (client.connect(server, 80)) {
    Serial.println("Connecting...");

    // send the HTTP PUT request
    client.println(F("GET /asciilogo.txt HTTP/1.1"));
    client.println(F("Host: arduino.cc"));
    client.println("Connection: close");
    client.println();

    // note the time that the connection was made
    lastConnectionTime = millis();
  }
  else {
    // if you couldn't make a connection
    Serial.println("Connection failed");
  }
}


void printWifiStatus()
{
  // print the SSID of the network you're attached to
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your WiFi shield's IP address
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);

  // print the received signal strength
  long rssi = WiFi.RSSI();
  Serial.print("Signal strength (RSSI):");
  Serial.print(rssi);
  Serial.println(" dBm");
}
```

Figure 1 output of connecting to/fetching web page using wifi