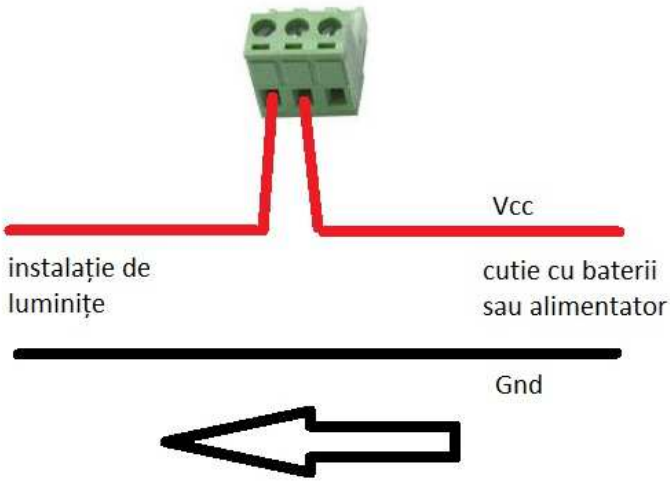
Control WiFi pentru luminițele de Crăciun

Farmecul sărbătorilor de iarnă este dat, printre altele, de instalațiile de lumini. În sute de variante, în zeci de culori, aflate undeva între plăcerea de a decora și obsesie, jucării sau elemente de iluminat, instalațiile de luminițe sunt cu siguranță un element important în cadrul pregătirilor pentru sărbătoarea de Crăciun. Dacă ați experimentat suficiente instalații și simțiți că vreți să treceți la un nivel următor (adică să începeți să le personalizați), în cadrul acestei lecții vă propunem modificarea unei instalații existente prin adăugarea unei comenzi de pornire prin WiFi.

Comanda de pornire se poate realiza simplu prin intermediul unui element de tip releu. Pentru a realiza comanda de la distanță prin intermediul WiFi este necesar să conectăm elementul de tip releu la o placă de dezvoltare ce permite comunicație WiFi. Realizarea unui ansamblu compact, placă de dezvoltare WiFi plus releu de comandă, este însă o sarcină destul de anostă. Din acest motiv vă propunem utilizarea unei platforme integrate: ESP8266-EVB-BAT-BOX (1). Această platformă este echipată cu un microprocesor WiFi ESP8266 și un releu 220VAC/10A. Platforma se poate alimenta de la alimentator de rețea de 5V sau de la un acumulator LiPo de 3.7V. Platforma se comercializează împreună cu o carcasă de protecție HAMMOND și conține un conector WAGO cu șurub de tip mufă (din două părți) pentru integrarea facilă a releului într-un montaj. Datorită facilităților oferite de această platformă este ideală pentru o aplicație de comandă de tip WiFi.

ATENȚIE!!! Chiar dacă releul de pe platforma EVB-BAT-BOX (1) suportă comanda unor circuite conectate la tensiune de rețea (220V), NU este recomandată realizarea de montaje proprii la această tensiune fără supravegherea sau îndrumarea unui specialist. Pericol de electrocutare, moarte, scurtcircuit, incendiu!!! Pentru montaje personale de tip hobby se recomandă comanda unor circuite de luminițe alimentate de la baterii sau cu alimentator de rețea (tensiunea din firele conectate la releu să fie de 3V-12V) eliminând astfel riscul de accindentare.

Conectarea instalației se va face prin secționarea firului de curent (nu cel de masă) și intercalarea releului. Astfel, platforma EVB-BAT-BOX va putea controla alimentarea instalației și vom putea programa când aceasta să se aprindă. Sistemul propus va putea să aprindă instalația la o anumită oră sau în urma unei comenzi directe.



Programarea platformei necesită un programator FTDI de 3.3V (2). Pentru mai multe detalii despre programarea platformei EVB-BAT-BOX este necesară consultarea materialului „HOW TO USE ESP8266 WITH ARDUINO IDE” (3). Programul a fost dezvoltat și testat utilizând Arduino IDE 1.8.3 având instalată extensia ESP8266 Community 2.3.0.

```
#include <ESP8266WiFi.h>
#include <ESP8266mDNS.h>
#include <WiFiUdp.h>
#include <ESP8266WebServer.h>
#include <TimeLib.h>
const int timeZone = 2;
WiFiUDP Udp;
const int NTP_PACKET_SIZE = 48;
byte packetBuffer[NTP_PACKET_SIZE];
unsigned int localPort = 2390;
WiFiServer server(80);
```

În cadrul programului trebuie personalizate datele de conectare la rețeaua WiFi (variabilele *ssid* și *pass*).

```
char ssid[] = "…";
```

```
char pass[] = "…";
WiFiClient client;
#define HOSTNAME "ESP8266-"
#define RELAY_PIN 5
int start_hour = 20;
int stop_hour = 23;
boolean automatic = true;
```

În cadrul secțiunii *setup()* se va inițializa conexiunea WiFi și se va raporta în consola serială adresa IP a sistemului. Aceasta trebuie folosită pentru a accesa sistemul dintr-un client web (browser) din aceiași rețea locală. Programul va utiliza biblioteca *Time* și conexiunea de rețea pentru a menține ora exactă (prin sincronizare NTP).

```
void setup() {
  Serial.begin(115200);
  Serial.println();
  Serial.println();
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN,LOW);
  String hostname(HOSTNAME);
  hostname += String(ESP.getChipId(), HEX);
  Serial.println("Hostname:" + hostname);
  WiFi.mode(WIFI_STA);
  WiFi.hostname(hostname);
  WiFi.begin(ssid, pass);
  delay(5000);
  if (WiFi.status() != WL_CONNECTED) {
    delay(10000);
    ESP.restart();
  }
  Serial.print("IP:");
  Serial.println(WiFi.localIP());
  server.begin();
```

```
  Udp.begin(localPort);
  setSyncProvider(getNtpTime);
}
```

Secțiunea *loop()* implementează comunicația HTTP ce ne va permite acționarea releului de la distanță. Interfața de comandă (ce poate fi accesată la adresa IP a sistemului) prezintă ora curentă, modul de funcționare (automat sau manual), ora de pornire și oprire automată precum și comenzile manuale de pornire / oprire. La pornire sistemul este în mod automat și are setate ca oră de pornire ora 20:00 și ca oră de oprire ora 23:00. Prin acționarea semnelor +/- din dreptul celor două ore presetate acestea pot fi modificate.

WiFi Christmas Lights

Time now: 19:13:52
Mode: Automatic
Click here to turn ON the Christmas Lights
Click here to turn OFF the Christmas Lights
Autostart at (+)20(-) hour.
Autostop at (+)23(-) hour.

Prin acționarea comenzilor manuale de oprire sau pornire se va comuta în modul manual de comandă în care intervalul de funcționare automată este ignorat și sistemul răspunde doar la comenzile manuale de pornire și oprire. Reactivarea modului de funcționare automat se poate face apelând orice modificare (+/-) a orelor de pornire / oprire automată.

WiFi Christmas Lights

Time now: 19:17:45
Mode: Manual
Click here to turn ON the Christmas Lights
Click here to turn OFF the Christmas Lights
Autostart at (+)20(-) hour.
Autostop at (+)23(-) hour.

```
void loop() {
  if (automatic)
    if ((hour()>start_hour) && (hour()<stop_hour))
```

```
      digitalWrite(RELAY_PIN, HIGH);
  else digitalWrite(RELAY_PIN, LOW);
WiFiClient client = server.available();
if (!client) return;
Serial.println("new client");
while(!client.available()){ delay(1); }
String request = client.readStringUntil('\r');
Serial.println(request);
client.flush();
if (request.indexOf("/COMMAND=ON") != -1) {
  digitalWrite(RELAY_PIN, HIGH);
  automatic = false;
}
else if (request.indexOf("/COMMAND=OFF") != -1) {
  digitalWrite(RELAY_PIN, LOW);
  automatic = false;
}
else if (request.indexOf("/COMMAND=PLUS_START") != -1) {
  start_hour++;
  if (start_hour>23) start_hour=0;
  automatic = true;
}
else if (request.indexOf("/COMMAND=MIN_START") != -1) {
  start_hour--;
  if (start_hour<0) start_hour=23;
  automatic = true;
}
else if (request.indexOf("/COMMAND=PLUS_STOP") != -1) {
  stop_hour++;
  if (stop_hour>23) stop_hour=23;
  automatic = true;
}
```

```
else if (request.indexOf("/COMMAND=MIN_STOP") != -1) {
  stop_hour--;
  if (stop_hour<0) stop_hour=23;
  automatic = true;
}
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("");
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.println("<h1>WiFi Christmas Lights</h1>");
client.println("<br>");
client.print("Time now: ");
client.print(hour());
client.print(":");
client.print(minute());
client.print(":");
client.println(second());
client.println("<br>");
if (automatic) client.print("Mode: Automatic");
else client.print("Mode: Manual");
client.println("<br>");
client.println("Click <a href=\"/COMMAND=ON\">here</a>
      to turn ON the Christmas Lights<br>");
client.println("Click <a href=\"/COMMAND=OFF\">here</a>
      to turn OFF the Christmas Lights<br>");
client.print("Autostart at <a
      href=\"/COMMAND=PLUS_START\">(+)</a>");
client.print(start_hour);
client.println("<a href=\"/COMMAND=MIN_START\">(-)</a>
      hour.");
client.println("<br>");
```

```
  client.print("Autostop at <a
        href=\"/COMMAND=PLUS_STOP\">(+)</a>");
  client.print(stop_hour);
  client.println("<a href=\"/COMMAND=MIN_STOP\">(-)</a>
        hour.");
  client.println("</html>");
  delay(1);
  Serial.println("Client disonnected");
  Serial.println("");
}
```

Funcțiile *getNtpTime()* și *sendNTPpacket()* sunt apelate periodic automat de biblioteca Time pentru sincronizarea timpului.

```
IPAddress timeServerIP;
const char* ntpServerName = "time.nist.gov";
time_t lastsyncr;

time_t getNtpTime() {
  WiFi.hostByName(ntpServerName, timeServerIP);
  while (Udp.parsePacket() > 0) ;
  sendNTPpacket(timeServerIP);
  uint32_t beginWait = millis();
  while (millis() - beginWait < 2500) {
    int size = Udp.parsePacket();
    if (size >= NTP_PACKET_SIZE) {
      Udp.read(packetBuffer, NTP_PACKET_SIZE);
      unsigned long secsSince1900;
      secsSince1900 =  (unsigned long)packetBuffer[40] <<
        24;
      secsSince1900 |= (unsigned long)packetBuffer[41] <<
        16;
```

```
      secsSince1900 |= (unsigned long)packetBuffer[42] <<
        8;
      secsSince1900 |= (unsigned long)packetBuffer[43];
      lastsyncr = (time_t) (secsSince1900 - 2208988800UL +
        timeZone * SECS_PER_HOUR);
      return lastsyncr;
    }
  }
 return 0;
}

void sendNTPpacket(IPAddress &address) {
  memset(packetBuffer, 0, NTP_PACKET_SIZE);
  packetBuffer[0] = 0b11100011;
  packetBuffer[1] = 0;
  packetBuffer[2] = 6;
  packetBuffer[3] = 0xEC;
  packetBuffer[12]  = 49;
  packetBuffer[13]  = 0x4E;
  packetBuffer[14]  = 49;
  packetBuffer[15]  = 52;
  Udp.beginPacket(address, 123);
  Udp.write(packetBuffer, NTP_PACKET_SIZE);
  Udp.endPacket();
}
```

Pentru alte variante de realizare puteți consulta și proiectele:

- ESP8266 Module – WiFi Christmas Lights (4)
- Wi-Fi christmas lights (5)
- ESP8266 IoT Holiday Light Controller (6)
- Node JS / ESP8266 WIFI Controlled Christmas Lights Class (7)

Referințe on-line

(1) ESP8266-EVB-BAT-BOX

https://www.robofun.ro/esp266-evb-bat-box?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(2) Conector FTDI 3.3 V

https://www.robofun.ro/conector-ftdi-3V3?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(3) HOW TO USE ESP8266 WITH ARDUINO IDE

https://www.olimex.com/Products/IoT/ESP8266-EVB/resources/ESP8266-EVB-how-to-use-Arduino.pdf

(4) ESP8266 Module – WiFi Christmas Lights

http://www.mrhobbytronics.com/esp8266-module-wifi-christmas-lights/

(5) Wi-Fi christmas lights

https://hackaday.io/project/18816-wi-fi-christmas-lights

(6) ESP8266 IoT Holiday Light Controller

https://www.hackster.io/wilfrednilsen/esp8266-iot-holiday-light-controller-2081cc

(7) Node JS / ESP8266 WIFI Controlled Christmas Lights Class

https://www.meetup.com/Austin-Web-Mobile-Electronic-Club-For-Beginners-Experts/events/243160186/