

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

WiFi Car Revolutions (partea I)

La fel ca și în cazul filmelor de succes, reluarea repetată a unei idei este destul de periculoasă din cauza epuizării rând pe rând a variantelor interesante ce pot capta atenția. Totuși, în cadrul lecției de față vom relua problematica realizării unei platforme robotice comandate prin WiFi sperând să vă păstrăm în continuarea atenția și interesul prin propunerea unei soluții ”revoluționare” atât ca și componente hardware utilizate cât și ca modalitate de comandă.

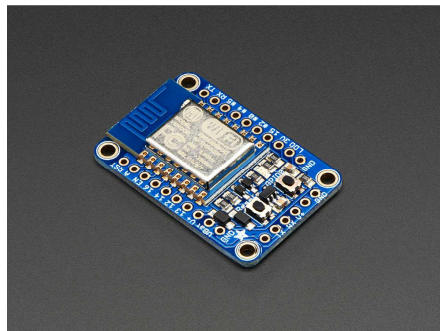


Platforma robotică propusă se bazează pe aceeași platformă FlexyBot utilizată și în precedentele două WiFi Car. Se poate utiliza atât varianta cu două motoare cât și varianta cu patru motoare (comandă unificată pentru motoarele de pe aceeași parte a platformei).



<https://www.robofun.ro/kit-roboti>

Pentru implementarea comenzii vom utiliza o soluție clasică: placă de dezvoltare [Arduino Uno R3](#) și [shield driver de motoare L298](#) (shield utilizat și în prima versiune a WiFi Car). Pentru a realiza conexiunea WiFi vom utiliza o componentă Adafruit HUZAH ESP8266 ce permite o conversie între comunicația de rețea WiFi și comunicația serială.

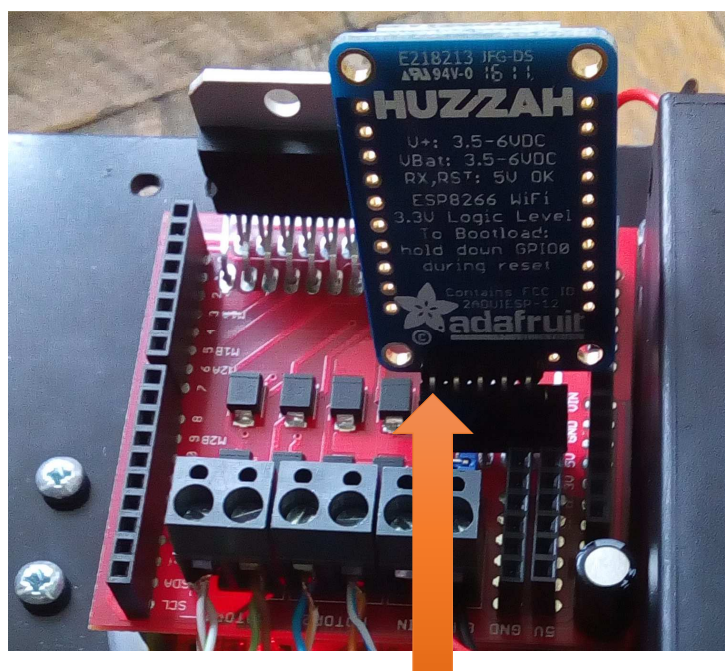


<https://www.robofun.ro/wireless/wireless-wifi/huzzah-esp8266>

<https://www.robofun.ro/forum/>

Noua soluție de implementare a comenzii are următoarele avantaje:

- Un cost mai mic (placa de dezvoltare Arduino Uno + modulul Adafruit HUZZAH sunt mai ieftine împreună decât placa de dezvoltare Arduino Yun);
- Se poate implementa o alimentare comună, placa de dezvoltare se poate alimenta direct prin intermediul shield-ului de motoare – se poate utiliza o singură sursă de alimentare de 7.4V (accumulator LiPo cu două celule) sau 9V ([cutie 6 baterii AA](#)) direct pe pinii de alimentare a shieldului cu jumperul de alimentare pus;
- Soluția este extrem de familiară fiind similară cu soluția propusă de kit-ul [FlexyBot Doua Motoare Controlat prin Bluetooth](#) – doar se înlocuiește modulul bluetooth cu cel WiFi – conectorul este compatibil cu noul modul fiind vorba tot de comunicație serială. Aveți grijă să potriviți pinul de GND al modulului cu pinul de GND al conectorului de pe shield.



Înainte de montarea modulului WiFi pe shield el trebuie programat, după montare nu mai este posibilă programarea deoarece pinii de programare sunt utilizați pentru comunicația serială cu placa Arduino Uno.

Pentru programarea și utilizarea modului Adafruit HUZZAH este necesară parcurgerea materialului următor:

Adafruit HUZZAH ESP8266 breakout

<https://learn.adafruit.com/adafruit-huzzah-esp8266-breakout>

Programul ce va rula pe modulul Adafruit este derivat din exemplul WiFiTelnetToSerial al extensiei esp8266 pentru Arduino IDE (codul a fost testat utilizând Arduino IDE 1.8.1 și extensia esp8266 2.3.0):

```
#include <ESP8266WiFi.h>
#define MAX_SRV_CLIENTS 1
const char* ssid = "...";
const char* password = "...";
WiFiServer server(23);
WiFiClient serverClients[MAX_SRV_CLIENTS];
void setup() {
    Serial.begin(9600);
    Serial.println();
    WiFi.begin(ssid, password);
    uint8_t i = 0;
    while (WiFi.status() != WL_CONNECTED && i++ < 20)
        delay(500);
    if(i == 21){ while(1) delay(500); }
    server.setNoDelay(true); }

void loop() {
    uint8_t i;
    if (server.hasClient()){
        for(i = 0; i < MAX_SRV_CLIENTS; i++){
            if (!serverClients[i] ||
                !serverClients[i].connected()){
                if(serverClients[i]) serverClients[i].stop();
```

```

        serverClients[i] = server.available();
        serverClients[i].println(F("=== WiFi Car
                                   Revolutions ==="));

        continue;
    }
}

WiFiClient serverClient = server.available();
serverClient.stop();
}

for(i = 0; i < MAX_SRV_CLIENTS; i++){
    if (serverClients[i] && serverClients[i].connected()){
        if(serverClients[i].available()){
            while(serverClients[i].available())
                Serial.write(serverClients[i].read()); }
        }
    }

    if(Serial.available()){
        size_t len = Serial.available();
        uint8_t sbuf[len];
        Serial.readBytes(sbuf, len);
        for(i = 0; i < MAX_SRV_CLIENTS; i++){
            if (serverClients[i] &&
                serverClients[i].connected()){
                serverClients[i].write(sbuf, len);
                delay(1); }
        }
    }
}
}

```

Pentru încărcarea codului se va utiliza un [cablu USB to TTL](#) sau un [conector FTDI](#). Modulul WiFi va încerca conectarea într-o rețea WiFi locală ce oferă IP-ul dinamic prin DHCP – credențialele de conectare (*ssid* și *password*) trebuie personalizate în program.

După programarea modulului WiFi el poate fi conectat la platforma mobilă și se poate trece la programarea plăcii Arduino Uno. Codul pentru componenta Arduino este similar cu codul utilizat în cadrul variantei WiFi Car Reloaded pentru componenta 32U4 – programul va avea rolul de a transpune comenzile primite serial în comenzi către motoare (apar deosebiri la pinii de comandă ai motoarelor și la portul serial folosit, în cazul de față vom utiliza un port serial software - toate specifice shield-ului de motoare folosit).

```
#define MOTOR2_PIN1 3
#define MOTOR2_PIN2 5
#define MOTOR1_PIN1 6
#define MOTOR1_PIN2 9

#include "SoftwareSerial.h"
SoftwareSerial mySerial(2,4);

int vs = 0;
int vd = 0;

void setup()
{ mySerial.begin(9600);
  pinMode(MOTOR1_PIN1, OUTPUT);
  pinMode(MOTOR1_PIN2, OUTPUT);
  pinMode(MOTOR2_PIN1, OUTPUT);
  pinMode(MOTOR2_PIN2, OUTPUT); }

void loop()
{ if (mySerial.available()) {
  char c = (char)mySerial.read();
  switch (c) {
    case 'i':
      if (vs<245) vs=vs+10;
      if (vd<245) vd=vd+10;
```

```
mySerial.print("Viteza: ");
mySerial.print(vs);
mySerial.print("/");
mySerial.println(vd);
break;
case 'b':
    if (vs>-245) vs=vs-10;
    if (vd>-245) vd=vd-10;
    mySerial.print("Viteza: ");
    mySerial.print(vs);
    mySerial.print("/");
    mySerial.println(vd);
    break;
case 's':
    if (vs<245) vs=vs+10;
    if (vd>-245) vd=vd-10;
    mySerial.print("Viteza: ");
    mySerial.print(vs);
    mySerial.print("/");
    mySerial.println(vd);
    break;
case 'd':
    if (vs>-245) vs=vs-10;
    if (vd<245) vd=vd+10;
    mySerial.print("Viteza: ");
    mySerial.print(vs);
    mySerial.print("/");
    mySerial.println(vd);
    break;
case 'x':
    vs=0;
    vd=0;
```

```

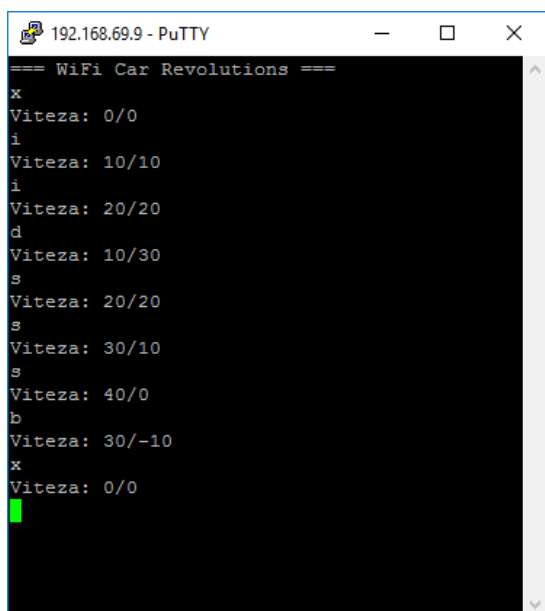
        mySerial.print("Viteza: ");
        mySerial.print(vs);
        mySerial.print("/");
        mySerial.println(vd);
        break;
    }
}
mySerial.flush();
go(vs, vd);
delay(50);
}

void go(int speedLeft, int speedRight) {
    if (speedLeft > 0) {
        analogWrite(MOTOR1_PIN1, speedLeft);
        analogWrite(MOTOR1_PIN2, 0);
    }
    else {
        analogWrite(MOTOR1_PIN1, 0);
        analogWrite(MOTOR1_PIN2, -speedLeft);
    }

    if (speedRight > 0) {
        analogWrite(MOTOR2_PIN1, speedRight);
        analogWrite(MOTOR2_PIN2, 0);
    }else {
        analogWrite(MOTOR2_PIN1, 0);
        analogWrite(MOTOR2_PIN2, -speedRight);
    }
}
}

```


Comanda platformei robotice rezultate se poate face, la fel ca și în cazul WiFi Car Reloaded, utilizând un client telnet (adresa de conectare este adresa IP alocată modulului WiFi prin DHCP) și comenzile i, d, s, b, x:

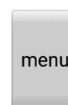


Sau utilizând aplicația Android RoboRemoFree - Arduino control:

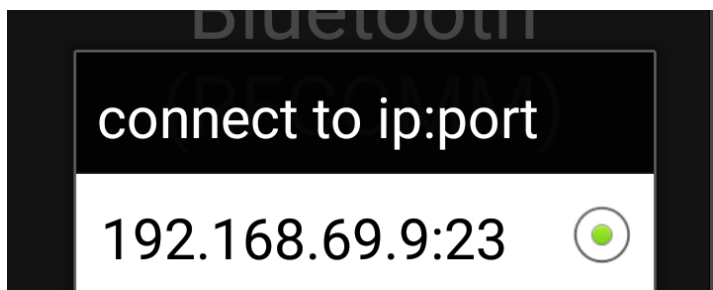


<https://play.google.com/store/apps/details?id=com.hardcodedjoy.roboremofree>

și configurând o interfață cu cinci butoane pentru comenzile i, d, s, b, x.



Adresa de conectare va fi adresa IP a sistemului și portul va fi portul 23 (specific protocolului telnet):



<https://www.robofun.ro/forum/>