

Model auta riadený pomocou web stránky cez Internet

¹Adam BARÁK, ²Dániel GAJDOŠ, ³Ján MOLNÁR

^{1,2,3} Katedra teoretickej a priemyselnej elektrotechniky, Fakulta elektrotechniky a informatiky, Technická Univerzita Košice, Slovensko

¹adam.barak@student.tuke.sk, ²daniel.gajdos.2@student.tuke.sk, ³jan.molnar@tuke.sk

Abstrakt — Tento článok je zameraný na vytvorenie modelu auta, ktoré je riadené diaľkovo, pomocou internetu. Samotný model obsahuje viaceré separátne časti, ktoré sa starajú o jeho funkčnosť. Najdôležitejšou časťou je mikrokontrolér ESP32, pomocou ktorého je model riadený a tak isto je pomocou neho zabezpečená aj komunikácia pomocou Wi-Fi siete.

Kľúčové slová — auto, ESP32, motor, servomotor, komunikácia, Wi-Fi, web stránka

Car model controlled using the website over the Internet

Abstract — This article is aimed at creating a car model that is controlled by the Internet. The model itself contains several separate parts that take care of its functionality. The most important part is the ESP32 microcontroller, through which the model is controlled, and it is also provided by the WiFi network communication.

Keywords — car, ESP32, engine, servomotor, communication, Wi-Fi, website

I. ÚVOD

Hlavným modulom celého modelu je mikrokontrolér ESP32, v ktorom sa nachádza celý program pre riadenie jednotlivých častí modelu. Jednotlivými časťami, ktoré budú popísané neskôr, sú riadenie smeru natočenia auta, čo je zabezpečené servomotorom, teda môže zatáčať buď doľava alebo doprava. Ďalšou časťou je riadenie smeru pohybu – dozadu a dopredu, pomocou jednosmerného motora. Na aute sa nachádzajú aj “ramená,” ktoré dokážu podržať ľahší predmet zovretý a tým je možné ho preniesť na iné miesto. Taktiež sú tam doplnkové časti, ako napríklad svetlá, ktoré sú ovládané tiež prostredníctvom web-stránky. Všetky elektrické časti sú napájané pomocou batérií, ktoré sú umiestnené na aute, pričom je použitý aj modul na reguláciu napätia, keďže nie všetky časti pracujú na rovnakej napäťovej úrovni.

II. MODEL AUTA

Samotný model auta je vyrobený prevažne z dreva (kolesá, piliere podstavca pre batérie) a konštrukcia (podlaha, ramena) je vyrobená s drevovláknitej dosky. Spojovací materiál tvoria skrutky, matice a lepidlo na drevo. Následne sú použité aj iné materiály ako plast a pod. Zadné kolesa sú pomocou drevenej osky pripevnene k prevodovke, ktorá je priamo prepojená s jednosmerným motorom. Pre predné kolesa sú použité špeciálne skrutky, ktoré majú závit len do polovice tela, pričom časť ktorá neobsahuje závit obsahuje viacero podložiek až po začiatok závitú na skrutke. Spolu s podložkami je táto časť skrutky umiestnená do výrezu kolesa, nachádzajúceho sa v stredovej osi kolesa. Tento systém skrutky a podložiek nám vytvára podmienky pre rotovanie kolesa, ako keby bolo použité ložisko. Keďže model nie je príliš ťažký, bola táto metóda vyhovujúca. Druhá časť skrutky obsahujúca závit, slúži na

pripevnenie kolieska k drevenému ramenu auta pomocou matíc. V strede dreveného ramena je umiestnený hriadeľ, ktorý je prepojený so servomotorom. Pomocou servomotora dochádza k natočeniu hriadeľa spolu s ramenom, čím sa zabezpečuje vychýlenie kolies do oboch strán. Na rovnaké rameno, pomocou ktorého dochádza k vychýľovaniu kolies, je umiestnená aj dvojica ramien ovládaná pomocou druhého servomotora. Tieto ramená sa natáčajú súčasne s natočením prednej nápravy.

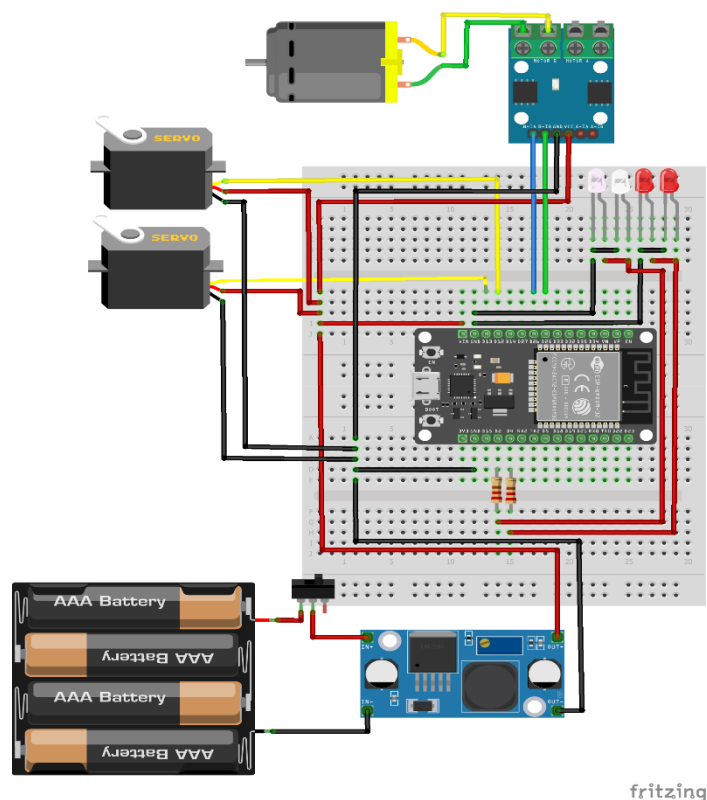
III. ELEKTRONIKA

Tak ako bolo spomínané vyššie, celý projekt je riadený pomocou mikrokontroléra ESP32 DevKit V1, ktorý v sebe zahŕňa aj samotný Wi-Fi modul na komunikáciu cez internet.

Ďalej obsahuje 2 servomotory. Jeden z nich slúži na riadenie otáčania predných kolies a druhý sa stará o pohyb ramien, pomocou ktorých je možné uchopiť predmet do určitej váhy.

O pohon zadných kolies sa stará jednosmerný motorček s napájaním 5 V, na ktorého riadenie je použitý externý modul L9110, ktorý predstavuje H-mosť.

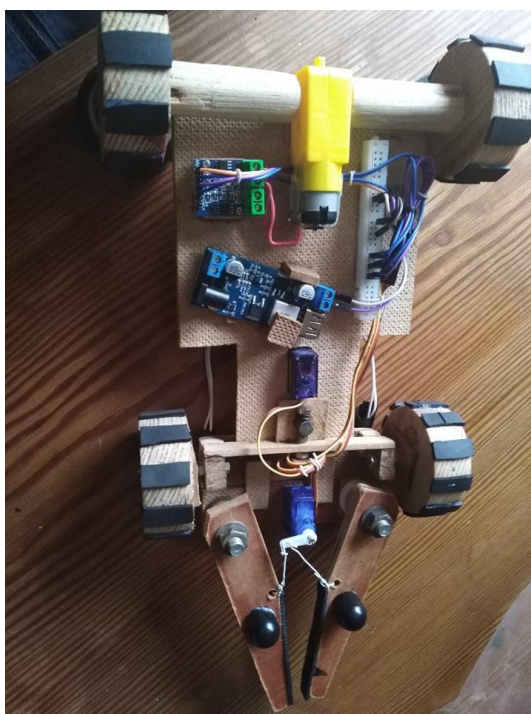
Na modeli sa nachádzajú doplnky a to predné a zadné svetlá, ktoré sú tvorené pomocou paralelne zapojených LED diód – to znamená, že 2 paralelné (biele) diódy pre predné svetlá a ďalšie 2 (červené) diódy pre zadné svetlá.



Obr. 1 Schéma zapojenia

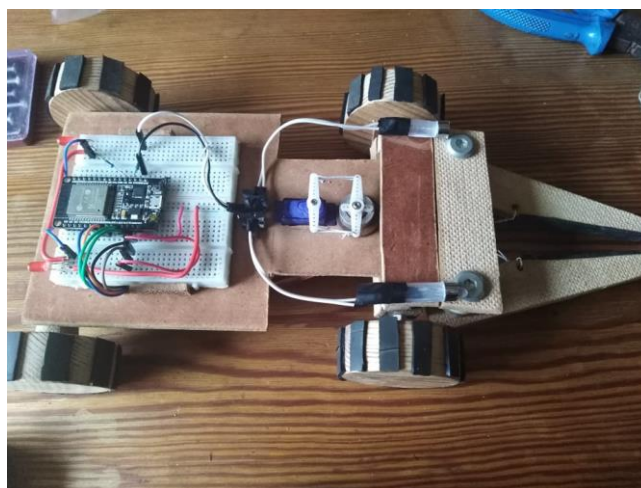
Na obrázku Obr. 1 je možné vidieť, že H-mosť je pripojený k mikrokontroléru pomocou dvojice riadiacich vodičov (modrý, zelený), ktorými sa posielajú digitálne signály vo forme log. 1 a log. 0. Ak chceme, aby sa autíčko pohlo, musíme priviesť na H-mosť signál log. 1, prostredníctvom jedného vodiča a signál log. 0, pomocou druhého vodiča. Smer, ktorým sa model auta bude pohybovať závisí od kombinácie digitálnych signálov privedených na H-mosť. V tomto prípade sú prípustne dve kombinácie. Napr. riadiacim vodičom č.1 príde log. 1 a riadiacim vodičom č.2 log. 0 alebo naopak. Zároveň samotný H-mosť je napájaný zo znižovacieho DC-DC meniča hodnotou 5 V. DC-DC menič bol použitý s dôvodu potreby napájania všetkých komponentov napájacím napätím 5 V. Na vstup meniča bol pripojený napájací zdroj zložený z batérií, pričom jeho hodnota je 11,1V. Servomotory sú, ako bolo spomenuté, napájané tiež z DC-DC meniča, ale ich riadiaci vodič je pripojený priamo na mikrokontrolér pomocou pinu D13 a D12. Zároveň všetky komponenty majú spoločné uzemnenie. Na obrázku Obr. 2 je možné vidieť, ako sú jednotlivé komponenty (H-mosť, DC-DC menič, DC motor,

...) prichytené a zapojené priamo na spodnej časti modelu auta.



Obr. 2 Zapojenie komponentov na aute (pohľad zo spodnej strany auta)

Na obrázku Obr. 3 je možné vidieť samotný mikrokontrolér a taktiež zapojenie predných a zadných svetiel a servomotora pre riadenie predných kolies.



Obr. 3 Zapojenie komponentov na aute (pohľad z zhora)

Napájanie modelu je tvorené pomocou 3 akumulátorov o kapacite 1300 mAh a maximálnym vybíjacím prúdom 13 A, keďže sú typu „High Discharge,“ a to znamená, že vedia dať na výstup až 10-násobok svojej kapacity. Výstupné napätie týchto do série zapojených jednotlivých akumulátorov (jeden akumulátor má napätie 3,7 V) je 11,1 V (teoretická hodnota). Toto napájanie je pripojené na vstup DC-DC meniča, ktorého napájacie napätie je prijateľné od 9 do 36 V.

IV. SOFTVÉR

Softvér je tvorený vo vývojovom prostredí Arduino IDE a Visual Studio Code. Arduino IDE obsahuje program pre mikrokontrolér ESP 32 a Visual Studio Code zase program webovej stránky, prostredníctvom ktorej je samotné auto ovládané. Na obr.4 sú zobrazené knižnice, ktorých funkcie sú

využívane priamo v programe pre mikrokontrolér.

```
#include <WiFi.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>
#include <Servo.h>
#include "SPIFFS.h"

AsyncWebServer server(80);

Servo myServo, myServo2;
int pos = 60;
```

Obr. 4 Knižnice zahrnuté v programe

Knižnica WiFi.h slúži na pripojenie mikrokontroléra k AP, prostredníctvom autentifikačných údajov. AsyncTCP.h, ESPAsyncWebServer slúžia na možnosť vytvorenie asynchrónneho servera a zabezpečenie komunikácie s ním. V komunikácii medzi klientom a serverom bude práve mikrokontrolér vystupovať ako server, ktorý bude spracovávať dopyty od klientov. To, že je asynchrónny znamená, že dokáže obsluhovať viac ako jedno pripojenie súčasne. Na mikrokontroléry je server vytvorený pomocou príkazu AsyncWebServer server(80), kde 80 je číslo portu pre protokol HTTP. Ďalej je pridaná knižnica Servo.h, ktorá, ako už je z názvu možné pochopiť, zabezpečuje prácu so servomotormi. Keďže používame 2 servomotory, pomocou príkazu Servo myServo, myServo2, si oba z nich inicializujeme. Posledný príkaz z obrázku Obr.4 je int pos=60, ktorý taktiež slúži pre servomotory a nastavuje ich východziu pozíciu. Ako posledná je použitá knižnica SPIFFS.h, ktorá zabezpečuje prístup do pamäte mikrokontroléra, kde sa nachádzajú nahraté súbory pre Web stránku.

Na obrázku Obr. 5 je možné vidieť funkcie pre Wi-Fi komunikáciu a taktiež aj samotný Setup, v ktorom sa nachádza v podstate inicializácia všetkých komponentov, s ktorými mikrokontrolér komunikuje. Nachádza sa tu aj samotné pripojenie Wi-Fi siete.

```
void setup() {
  lcd.init();
  lcd.backlight();
  myServo.attach(13);
  myServo2.attach(12);

  myServo.write(pos);
  myServo2.write(95);

  WiFi.disconnect(true);

  delay(1000);

  WiFi.onEvent(WiFiStationConnected, SYSTEM_EVENT_STA_CONNECTED);
  WiFi.onEvent(WiFiGotIP, SYSTEM_EVENT_STA_GOT_IP);
  WiFi.onEvent(WiFiStationDisconnected, SYSTEM_EVENT_STA_DISCONNECTED);

  WiFi.begin("123", "321dano321");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(1000);
  }
  Serial.println("");
  Serial.println("Pripojene IP adresa:");
  Serial.print(WiFi.localIP());

  pinMode(2, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(25, OUTPUT);
  pinMode(26, OUTPUT);
}
```

Obr. 5 Inicializácia servomotorov a Wi-Fi komunikácia

Popis funkcií na obrázku Obr 5:

- ✓ void WiFiStationConnected(); - informácia o úspešnom pripojení k Access Pointu;
- ✓ void WiFiGotIP(); - Vypíše IP adresu Wi-Fi, keď sa pripojí.
- ✓ void WiFiStationDisconnected(); - Ak sa preruší spojenie, táto funkcia vypíše, že sa pokúša o obnovenie pripojenia;
- ✓ WiFi.begin(); - autorizácia a pokus o pripojenie ku Wi-Fi sieti;

- ✓ WiFi.onEvent; - príkazy, ktoré sa vykonávajú, ak príde k nejakej udalosti spojenej s Wi-Fi prepojením.

Na Obr. 6 sa nachádza časť kódu, ktorá zabezpečuje samotné pripojenie k časti pamäti, kde sa nachádzajú jednotlivé súbory.

```
if(!SPIFFS.begin(true)){
    Serial.println("An Error has occurred while mounting SPIFFS");
    return;
}

server.on("/", HTTP_GET, [] (AsyncWebServerRequest *request){
    request->send(SPIFFS, "/ovladanie.html", "text/html");
});

server.on("/ovladanie.css", HTTP_GET, [] (AsyncWebServerRequest *request){
    request->send(SPIFFS, "/ovladanie.css", "text/css");
});

server.on("/frontlight", HTTP_GET, [] (AsyncWebServerRequest *request) {
    String zap = request->getParam("output")->value();
    digitalWrite(2, zap.toInt());
    Serial.print("GPIO: ");
    Serial.print(zap.toInt());
    request->send(200, "text/plain", "OK");
});

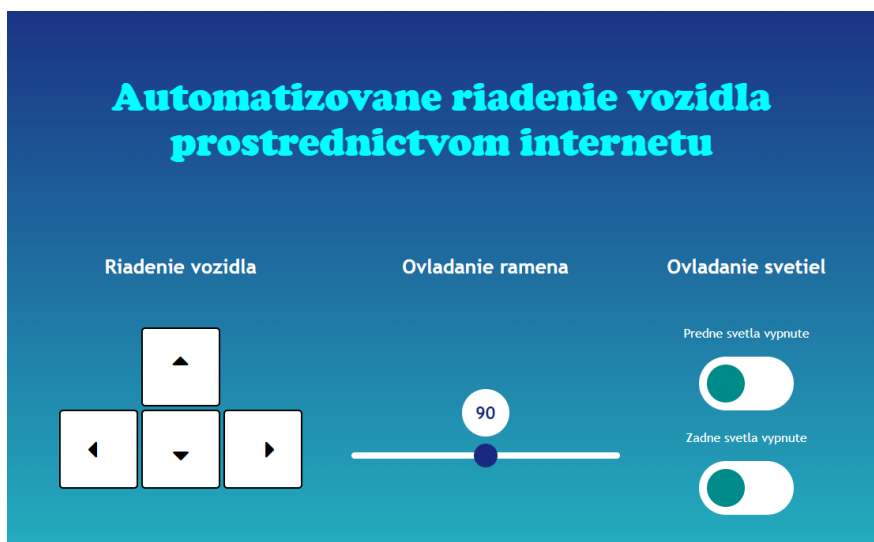
server.on("/bottomlight", HTTP_GET, [] (AsyncWebServerRequest *request) {
    String zap = request->getParam("output")->value();
    digitalWrite(4, zap.toInt());
    Serial.print("GPIO: ");
    Serial.print(zap.toInt());
    request->send(200, "text/plain", "OK");
});
```

Obr. 6 Funkcie zahŕňajúce odpoveď na požiadavky od klienta

Ďalej sa tu nachádzajú odpovede na požiadavky zaslané klientom (v tomto prípade, keďže ide o aplikáciu pre počítač, bude klientom práve ten). V jednotlivých odpovediach sa zároveň nachádzajú časti kódov, ktoré sa vykonávajú, ak k danej požiadavke príde. Cez funkciu server.on (), sú vytvorené jednotlivé odpovede na požiadavky, ktoré budú prijaté od klienta. Na základe toho je v jednotlivých odpovediach zahrnutá aj funkcionálna pre komponenty, ktoré majú byť ovplyvnené.

V. APLIKÁCIA

Aplikácia bola vytvorená vo vývojovom prostredí Visual Studio Code. Jedná sa vlastne o webovú stránku, ku ktorej klient pristúpi po zadaní IP adresy mikrokontroléra. Je tvorená jazykmi HTML, CSS a JAVASCRIPT. Prvé dva spomínané jazyky slúžia na vytvorenie štruktúry a dizajnu webovej aplikácie. Jazyk JAVASCRIPT má za úlohu spracovávať udalosti užívateľa na stránke a s potrebnými údajmi ich poslať prostredníctvom HTTP požiadavky na spracovanie do mikrokontroléra. Na stránke, ako je možné vidieť na Obr. 7, sa nachádzajú tlačidlá s ukazovateľmi smeru, pomocou ktorých sa riadi samotné autičko. Tieto tlačidlá sa aktivujú stlačením kláves W,A,S,D na klávesnici, čím vyvolajú požiadavku na server. Podobne funguje aj posúvač rozsahu, z angl. „range sliders“, ktorým sa riadi pohyb ramien. Zmenou rozsahu nastáva udalosť, ktorá pošle požiadavku spolu s hodnotou z rozsahu. A nakoniec tu máme ešte prepínacie tlačidlá, na zapínanie, resp. vypínanie predných alebo zadných svetiel. Meniť rozsah je možné pomocou kláves pravej, resp. ľavej šípky, ale taktiež pomocou myši. Prepínacie tlačidlá reagujú len na klik myši.



Obr. 7 Aplikácia pre riadenia vozidla

VI. ZÁVER

V tomto článku sme zaoberali návrhom modelu auta riadneho cez internet. Pri návrhu bolo potrebné vyhotoviť model auta ako po hardvérovej stránke, tak aj po softvérovej. Z hardvérového hľadiska bolo potrebné navrhnuť podvozok, pohon modelu a napájanie. Zo softvérového hľadiska ovládanie modelu a taktiež zabezpečiť komunikáciu vo Wi-Fi sieti. Pre lepšie ovládanie modelu bola vytvorená aj aplikácia, ktorá užívateľovi umožňuje daný model ovládať a taktiež vykonávať základné užívateľské nastavenia.

LITERATÚRA

- [1] ESPRESSIF SYSTEMS. 2016. ESP32 Datasheet. [Online]. [cit. 2021-9-27]. Dostupné na: https://cdn.sparkfun.com/datasheets/IoT/esp32_datasheet_en.pdf
- [2] Datasheet pre H-mostík L9110. [Online]. [cit. 2021-11-20]. Dostupné na: <https://html.alldatasheet.com/html-pdf/1133007/ETC2/L9110/109/1/L9110.html>
- [3] TEXAS INSTRUMENTS. 2021. Datasheet pre menič LM2596. [Online]. [cit. 2021-11-20]. Dostupné na: <https://www.ti.com/lit/ds/symlink/lm2596.pdf>
- [4] MDM WEB DOCS. Getting Started – AJAX. [Online]. Developer.mozilla.org. [cit. 2021-10-14]. Dostupné na: https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started
- [5] SANTOS, Rui. ESP32-WebServer. [Online]. Random Nerd Tutorials. [cit. 2021-10-25]. Dostupné na: <https://randomnerdtutorials.com/esp32-web-server-arduino-ide/>
- [6] SANTOS, Rui. ESP32-AsyncWebServer: [Online]. Random Nerd Tutorials. [cit. 2021-10-25]. Dostupné na: <https://randomnerdtutorials.com/esp32-async-web-server-espasyncwebserver-library/>
- [7] SANTOS, Rui. ESP32-SPI Flash File System. [Online]. Random Nerd Tutorials. [cit. 2021-11-12]. Dostupné na: <https://randomnerdtutorials.com/esp32-web-server-spiffs-spi-flash-file-system/>
- [8] TECHTUTORIALSX. ESP32-Connecting to WiFi network. [Online]. Techtutorialsx.com. [cit. 2021-10-8]. Dostupné na: <https://techtutorialsx.com/2017/04/24/esp32-connecting-to-a-wifi-network/>