

**SVEUČILIŠTE U SPLITU
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I
BRODOGRADNJE**

**PROGRAMIRANJE MOBILNIH ROBOTA I
LETJELICA - SEMINAR**

**ROBOT UPRAVLJAN POKRETIMA RUKE
TEMELJEN NA ARDUINO PLATFORMI**

**Petra Šteko
Klara Pehar
Josip Nigojević**

Split, veljača 2022.

SADRŽAJ

1.	UVOD	1
2.	ARDUINO	2
2.1.	Arduino IDE	2
2.1.1.	Programiranje u Arduino IDE	3
3.	KORIŠTENE KOMPONENTE	5
3.1.	Linearni pozicijski senzor	5
3.2.	NODEMCU DEVKIT V1.0 (ESP8266)	6
3.3.	ESP32-WROOM-32C (ESP32)	7
3.4.	Servo motor SG90 Micro Servo	7
4.	REALIZACIJA SUSTAVA	10
4.1.	ESP načini rada	12
4.2.	ESP-NOW protokol	14
5.	ZAKLJUČAK	16
6.	LITERATURA	17

1. UVOD

U okviru ovog seminarskog rada potrebno je izraditi mobilnog robota s diferencijalnim pogonom temeljen na Arduino platformi koji se upravlja pokretima ruke.

Logika je razdvojena na klijentski i serverski dio. Na klijentskoj stani se nalaze linearni pozicijski senzori koji prate položaj prstiju na ruci te na temelju njihovog pomicanja na svom izlazu daje signal koji se čita pomoću ESP8266 razvojne pločice. Podatak o položaju prstiju se šalje bežičnom WiFi vezom na serverski dio gdje se nalazi ESP32 koji onda na temelju tih podataka pokreće motore robota na odgovarajući način. Komunikacija između klijenta i servera se odvija po ESP-NOW protokolu.

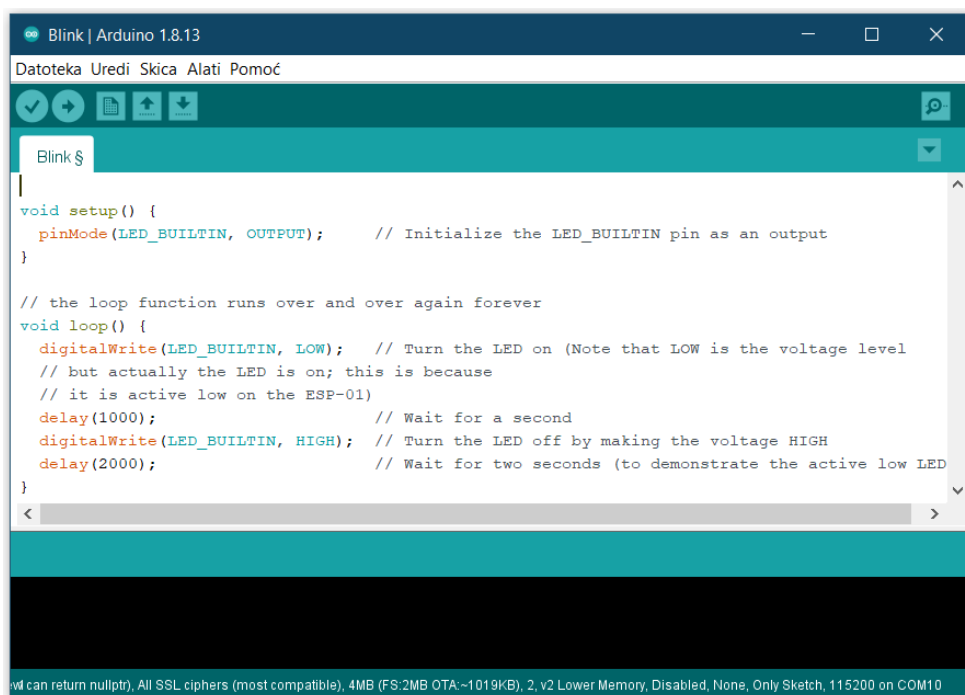
U nastavku će biti objašnjene korištene komponente, princip rada te način komunikacije.

2. ARDUINO

Arduino je računalna i programska platforma namjenjena kreiranju elektroničkih projekata. Sastoji se od hardvera (elektroničkog programibilnog sklopa kojeg čine mikrokontroler, ulazno/izlazni pinovi, USB priključak i druge slične komponente integrirane na jednu pločicu) i softvera kojeg čini besplatna Arduino Software IDE razvojna okolina [1]. Za Arduino postoje razni dodaci kao što su senzori, dodatni sklopovi koji proširuju njegove mogućnosti (engl. *shield*) i Wi-Fi moduli koji omogućuju izradu različitih projekata. U nastavku je opisan Arduino IDE koji je korišten za programiranje NodeMCU razvojne pločice.

2.1. Arduino IDE

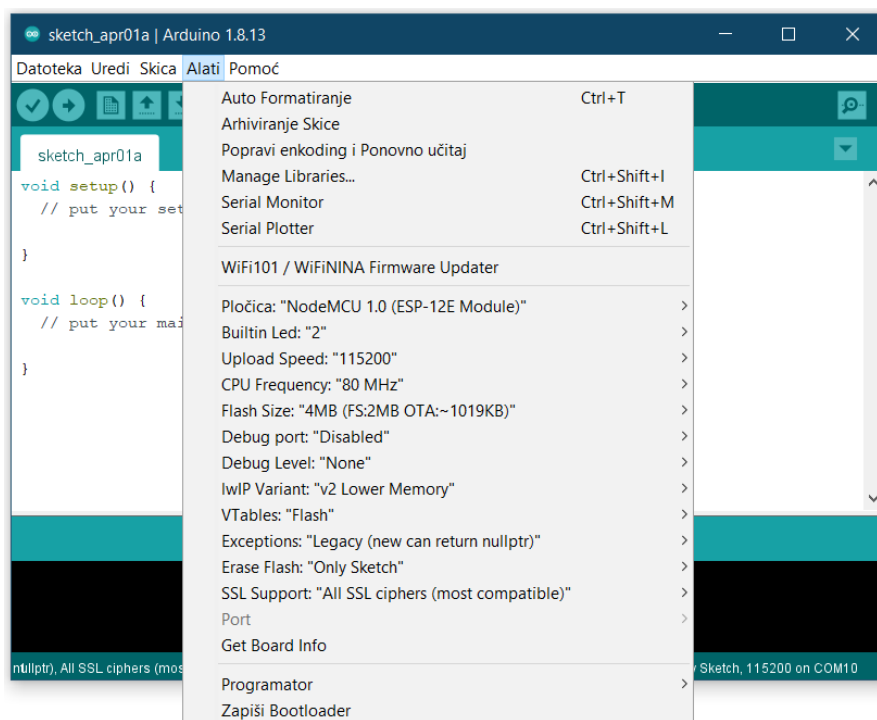
Arduino IDE je razvojna okolina otvorenog koda koja služi za pisanje koda i prijenos na odgovarajuću razvojnu pločicu. Softver otvorenog koda (engl. *open-source*) se odnosi na softver čiji je izvorni kod dostupan svim korisnicima koji mogu mijenjati, unaprijeđivati i prepravljati njegov sadržaj. Vrlo je jednostavan za korištenje, a sam IDE se sastoji od nekoliko dijelova kojeg čine tekstualni editor, konzola, izbornik i trake za brzi pristup kao osnovne alate potrebne programerima tijekom pisanja programa [2]. Napisani programi (tzv. skice, engl. *sketch*) se pohranjuju s ekstenzijom *.ino*. Primjer skice prikazan je na slici 2.1.



Slika 2.1. Arduino IDE

2.1.1. Programiranje u Arduino IDE

Arduino programski jezik je vrlo sličan C-u i C++ i omogućava jednostavno pisanje programa, prijenos i izvođenje programskog koda na razvojnoj pločici. Bogati izbor biblioteka (engl. *libraries*) nudi razne mogućnosti i povezivanje s različitim vrstama senzora i dodataka za izradu programskog koda za razne vrste projekata. Prvi korak pri izradi skice je odabir odgovarajuće razvojne pločice koja se koristi. To je moguće postaviti u izborniku *Alati*. Isto tako moguće je postaviti i neke druge opcije kao što su brzina prijenosa, frekvencija procesora itd. Osim izbornika *Alati* postoje i izbornici: *Datoteka* (za pohranu, otvaranje, zatvaranje skica i sl.), *Uredi* (za promjenu veličine fonta, razmaka, kopiranje i sl.), *Skica* (prijenos, kompajliranje i sl.) i *Pomoć* (za upoznavanje s okolinom).

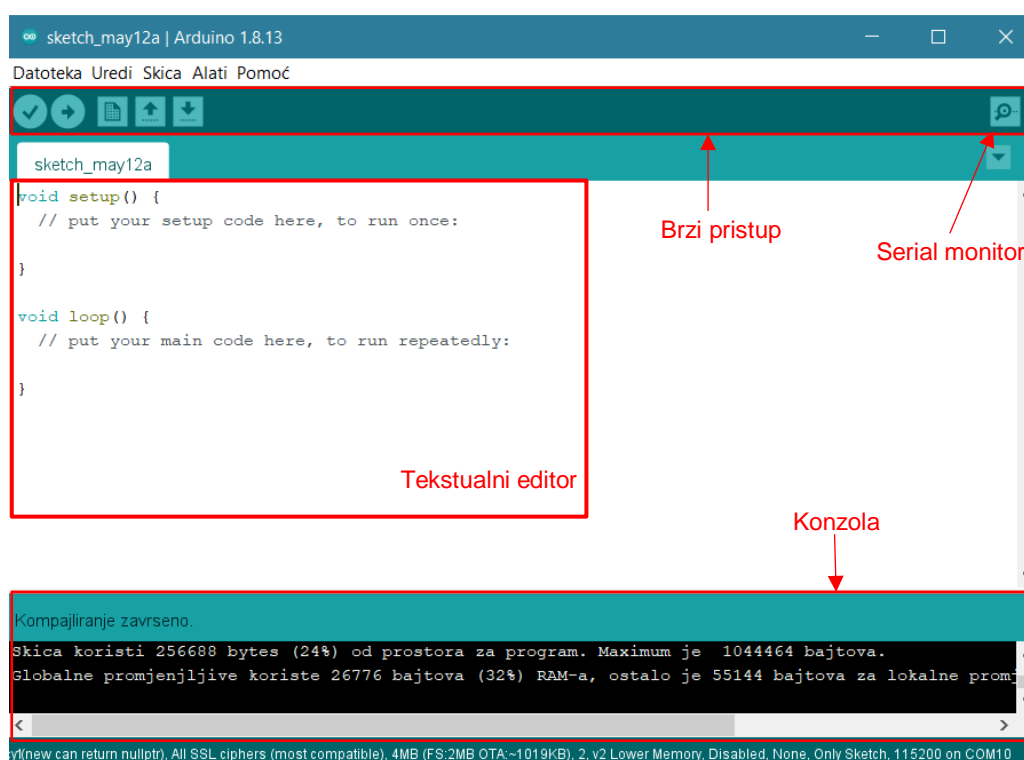


Slika 2.2. Prikaz izbornika alati

Arduino IDE se sastoji od brzog izbornika i tekstualnog editora (slika 2.3.). U brzom izborniku s lijeve strane postoje opcije (redom): *provjeri*, *prenesi*, *novi*, *otvori*, *pohrani*. Opcija *provjeri* kompajlira kod i provjerava je li kod ispravan i spreman za prijenos na pločicu. Ako postoje pogreške, one se ispisuju u konzoli. U konzoli se ispisuju i neke osnovne informacije o napisanom programskom kodu kao što su zauzeće dinamički alocirane memorije na pločici, preostali memorijski prostor i sl. Opcija *prenesi* služi za prijenos ispravnoga koda na razvojnu

pločicu. Opcije *novi*, *otvori* i *pohrani* služe za kreiranje nove skice, otvaranje postojećih skica i pohranjivanje skice. Na kraju brzog izbornika nalazi se gumb za otvaranje prozora za debugiranje (engl. *Serial monitor*). U njega je moguće ispisivati sadržaj naredbom *Serial.print()* i pomaže pri detektiranju logičkih pogrešaka u napisanom kodu.

Dio prozora u kojemu se uređuje i dodaje programski kod je tekstualni editor. Programski kod se može podijeliti u dvije cijeline: *setup* i *loop* funkciju. *Setup* funkcija se poziva kada skica započne i izvršava se samo jednom. Služi za inicijalizaciju varijabli, pinova, postavljanje početnih uvjeta rada i sl. *Loop* funkcija se, kao što joj samo ime kaže, izvršava više puta.



Slika 2.3. Prikaz brzog izbornika i tekstualnog editora

Nakon dovršetka pisanja programskog koda, ispravan kod se kompajlira i spreman je za prijenos na razvojnu pločicu.

3. KORIŠTENE KOMPONENTE

U nastavku je dan pregled svih korištenih komponenata za izradu robota.

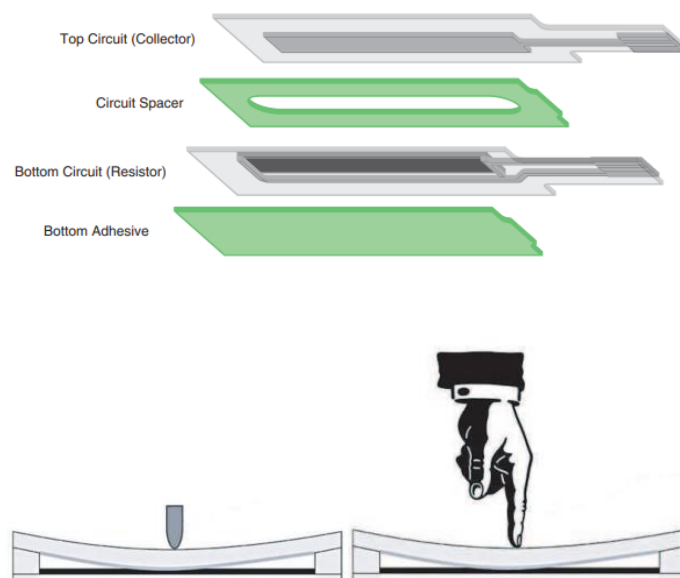
3.1. Linearni pozicijski senzor

Na slici 3.1. je prikazan korišteni linearni pozicijski senzor. Konkretni korišteni model je TSP-L-0012-103-1%-ST.



Slika 3.1. Linearni pozicijski senzor [3]

Senzor je otporni membranski potencimetar koji se sastoji od otpornika, kućišta i tzv. *wiper* sklopa. Na senzoru se nalaze tri pina: dva izlazna otporna kanala te kolektorski kanal. Pritiskom na senzor, generira se izlazni napon. *Wiper* je nevodljivi mehanizam koji pritišće gornji strujni krug koji aktivira potencijometar s vanjske strane. Gornji i donji strujni krug su odvojeni 0,15 mm odstoynog ljepljiva i kontakt između krugova nastane pritiskom (uobičajeni iznos sile je 0,7 - 1,8 N) na gornjem krugu. Tako se gornji krug gura prema dolje sve dok ne dođe do dodira između krugova kada se generira izlazni signal [4].

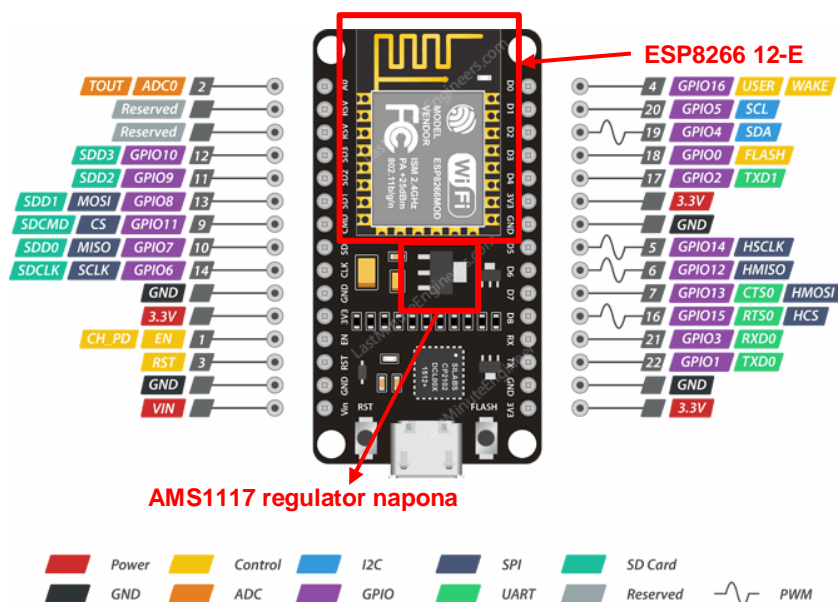


Slika 3.2. Prikaz slojeva senzora [4]

3.2. NODEMCU DEVKIT V1.0 (ESP8266)

ESP8266 je SoC (engl. *System on a Chip*) integrirani sklop temeljen na ESP-12E modulu koji je član serije ESP-XXX modula koji se razlikuju po izlaznim pinovima, flash memoriji i tipu antene. ESP8266 ima ugrađenu mogućnost Wi-Fi veze koja omogućava mikrokontroleru pristup Wi-Fi mreži i uspostave jednostavne TCP/IP veze korištenjem AT naredbi ili može djelovati kao samostalna Wi-Fi stanica (engl. *Access Point*). ESP-12E verzija ima 32-bitni mikrokontroler, 4 MB memorije za pohranu i 128 kB radne memorije. Često se koristi za razvoj IoT sustava i aplikacija. IoT (engl. *Internet of Things*) je mogućnost povezivanja putem interneta i omogućuje uređajima svih vrsta međusobnu komunikaciju i integraciju [5]. Često se koristi i za sustave kućne automatizacije i bežične upravljačke sustave.

NodeMCU je razvojna pločica s ugrađenim ESP8266 Wi-Fi modulom. U ovom radu korištena je v1.0 razvojna pločica. Sadrži mikro USB konektor koji omogućuje povezivanje na računalo i prijenos koda na pločicu. NodeMCU je moguće napajati preko USB konektora (napon 5 V) ili Vin pina (7 – 12 V) jer ima ugrađen AMS1117 regulator napona koji pretvara ulazni napon u 3.3 V potrebnih za napajanje. Sadrži deset digitalnih ulazno/izlaznih pinova (četiri od njih imaju PWM mogućnosti), jedan analogni pin, dva SPI sučelja, dva UART sučelja, jedno I2C sučelje, dva tipkala (engl. *flash*, *reset*) i ostale komponente koje omogućuju razvoj i testiranje [6]. Pregled i funkcije pinova su dani na slici 3.3.

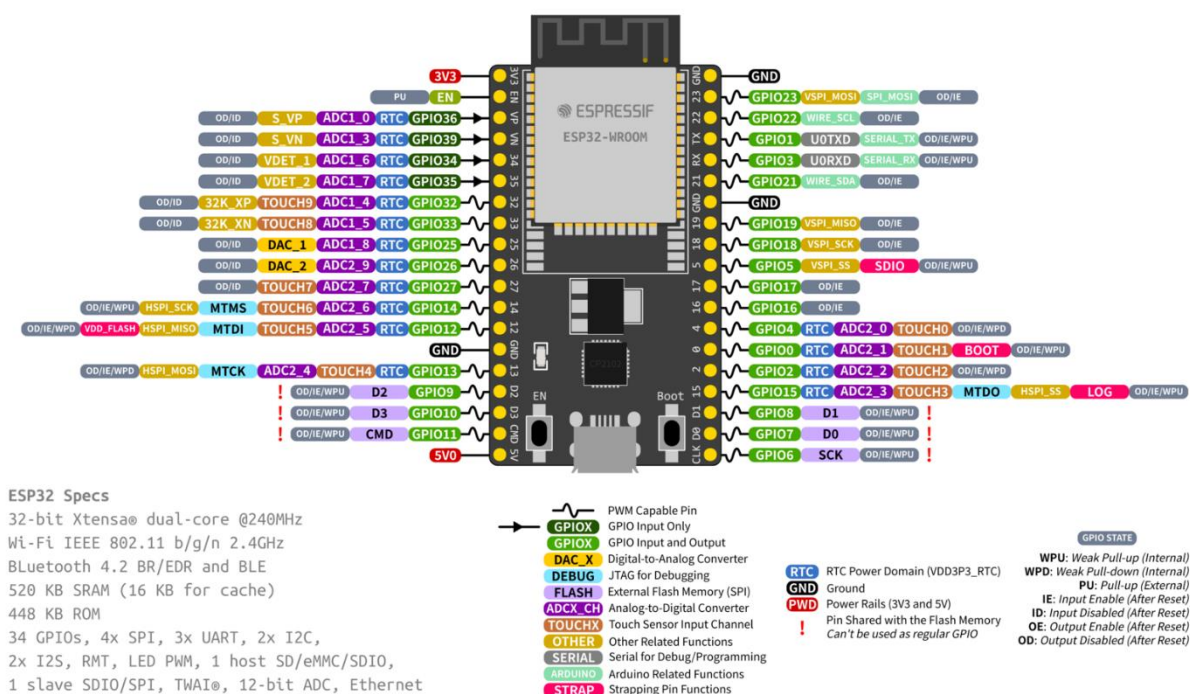


Slika 3.3. NodeMCU razvojna pločica [7]

3.3. ESP32-WROOM-32C (ESP32)

ESP32 je isto tako SoC koji može funkcionirati kao samostalan mikrokontroler, a u sebi ima ugrađeno sve što je potrebno za WiFi te za razliku od ESP8266 podržava i Bluetooth 4.0 komunikaciju. Kao i ESP8266, koristi se često u IoT aplikacijama. Sadrži 32-bitni mikrokontroler, 4 MB flash memorije, 520 kB RAM-a. Također sadrži napajanje od 5 V te 3.3 V. Sadrži 34 GPIO pina, četiri SPI, tri UART i dva I2C sučelja te preostale komponente koje se mogu koristiti za razvoj. Pregled pinova i njihove funkcije su dane na slici 3.4.

ESP32-DevKitC



Slika 3.4. ESP32-DevKit [8]

3.4. Servo motor SG90 Micro Servo

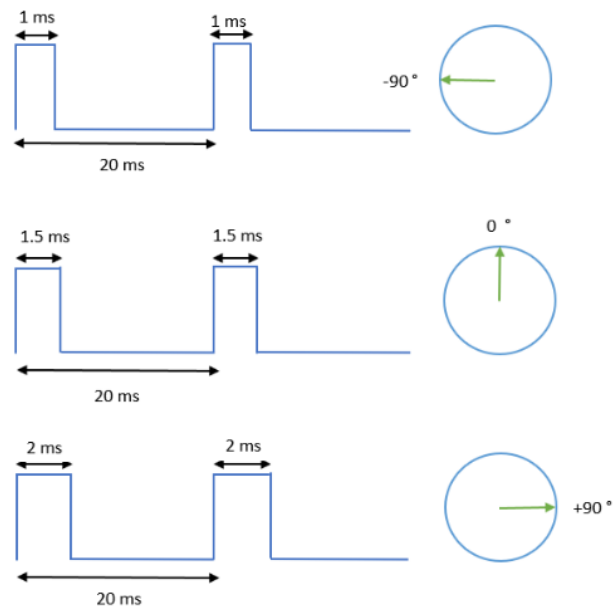
Postoje dvije vrste servo motora: servo motori s kontinuiranom rotacijom i pozicijski servo motori. Pozicijski rotacijski servo motori se okreću za 180° (često i za 210°). Također imaju graničnike u mehanizmu zupčanika za zaštitu izlazne osovine od pretjeranog okretanja. Servo motor s kontinuiranom rotacijom je servo koji nema ograničenja u rasponu kretanja. Umjesto da ulazni signal određuje u koji se položaj servo treba okretati, servo kontinuirane rotacije povezuje ulaz s brzinom izlaza i smjerom. Neograničeno kretanje ovih motora omogućuje im kretanje i u smjeru desno i lijevo [9].

Za pogon robota, korišteni su SG90 Micro Servo servo motori s mogućnošću rotacije za 360° (kontinuirana rotacija). SG90 ima tri pina: PWM (engl. *Pulse Width Modulation*) pomoću kojega se upravlja rotacijom motora, napajane (5 V) te uzemljenje.



Slika 3.5. SG90 Micro Servo motor [10]

Modulacija širine impulsa (PWM) šalje motoru električni impuls promjenjive širine. S PWM-om postoji minimalni puls, maksimalni puls i brzina ponavljanja. Kod pozicijskih servo motora, rotor će se okrenuti u željeni položaj na temelju trajanja impulsa. Kada se servo motoru naredi da se pomakne, on se pomakne na položaj i drži ga. Npr. za pozicijski SG90, PWM signal s trajanjem impulsa 1 ms i periodom signala 20 ms pomiče osovinu motora na položaj -90° , a impuls koji traje 2 ms s periodom 20 ms pomiče osovinu na položaj $+90^\circ$. Za impuls koji traje 1,5 ms motor miruje [11].

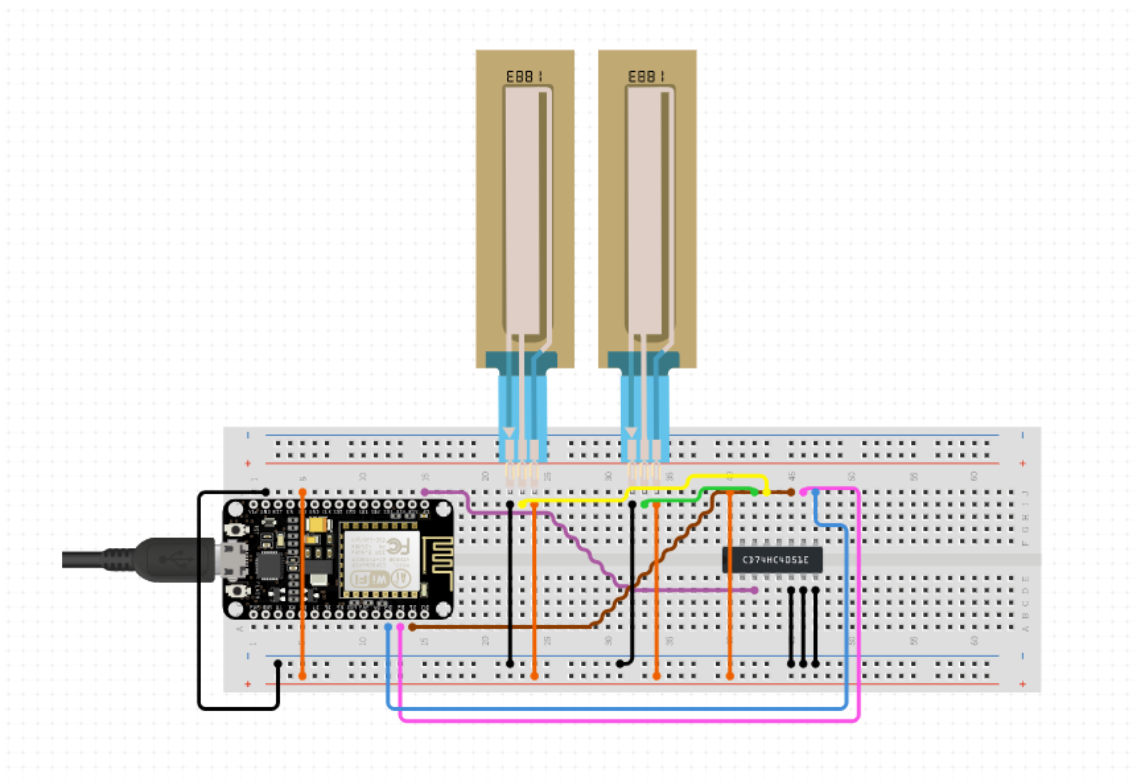


Slika 3.6. Upravljanje položajem servoa bez kontinuirane rotacije pomoću PWM signala [11]

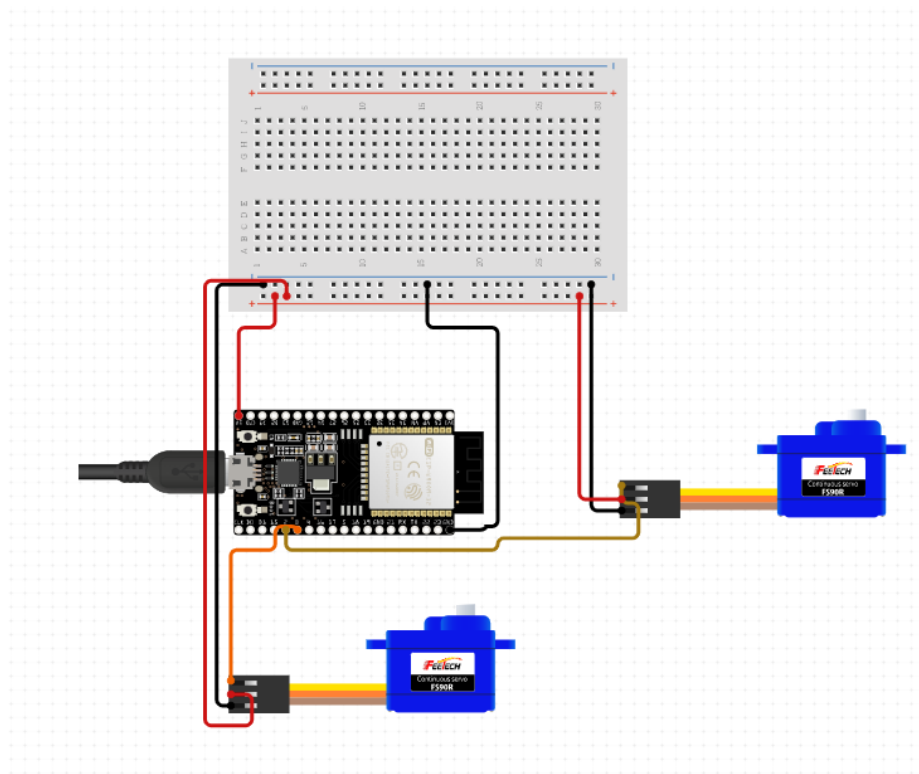
Za servo s kontinuiranom rotacijom vrijedi sličan princip. Takav servo motor upravljamo zadavanjem kuta. Kut od 90° predstavlja mirovanje dok kut manji od 90° (minimalno 0°) predstavlja jedan smjer vrtnje, a veći od 90° (maksimalno 180°) drugi smjer vrtnje uz razliku da npr. kut od 45° predstavlja veću brzinu vrtnje nego kut od 30° .

4. REALIZACIJA SUSTAVA

Sklopovski dio sustava je razdvojen na dvije cjeline: klijentski dio i serverski dio. Serverski dio se sastoji od linearnih pozicijskih senzora pomoću kojih se određuje položaj prsti ruke te ESP8266 pločice koja očitava podatak sa senzora. Klijentski dio se sastoji od ESP32 pločice koja podatak o položaju prstiju ruke prima bežično od ESP8266 pločice i na temelju tog podatka rotira servo motore u odgovarajućem smjeru. Na slikama 4.1 i 4.2 prikazane su montažne sheme oba dijela.

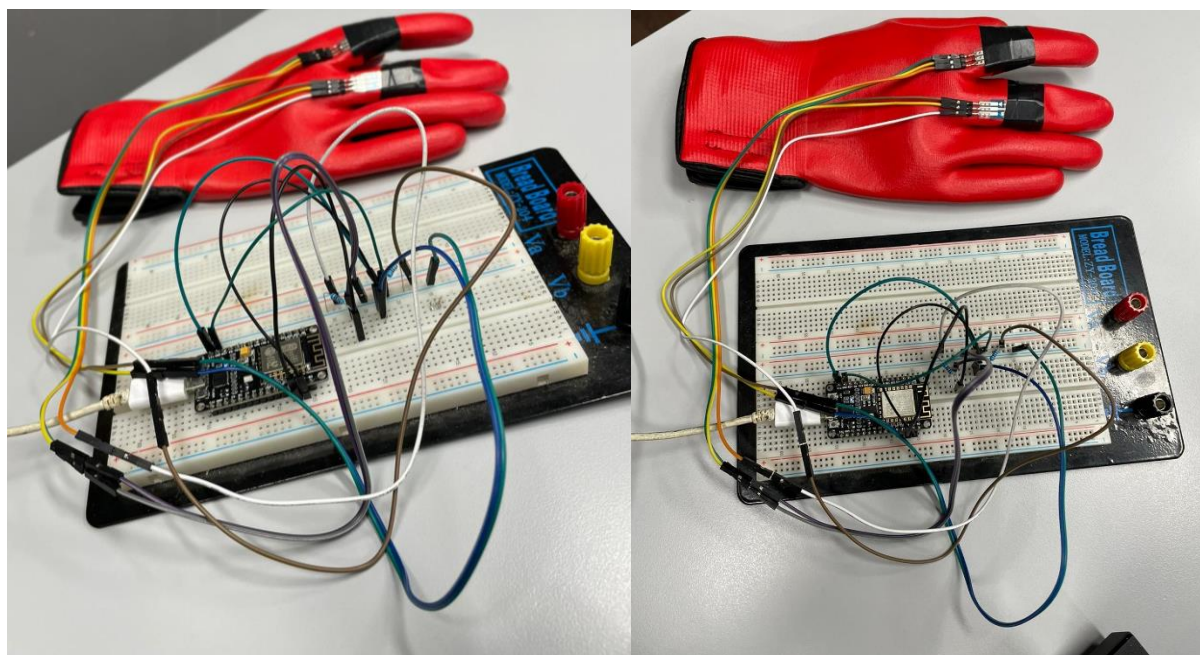


Slika 4.1. Pojednostavljena montažna shema serverskog dijela



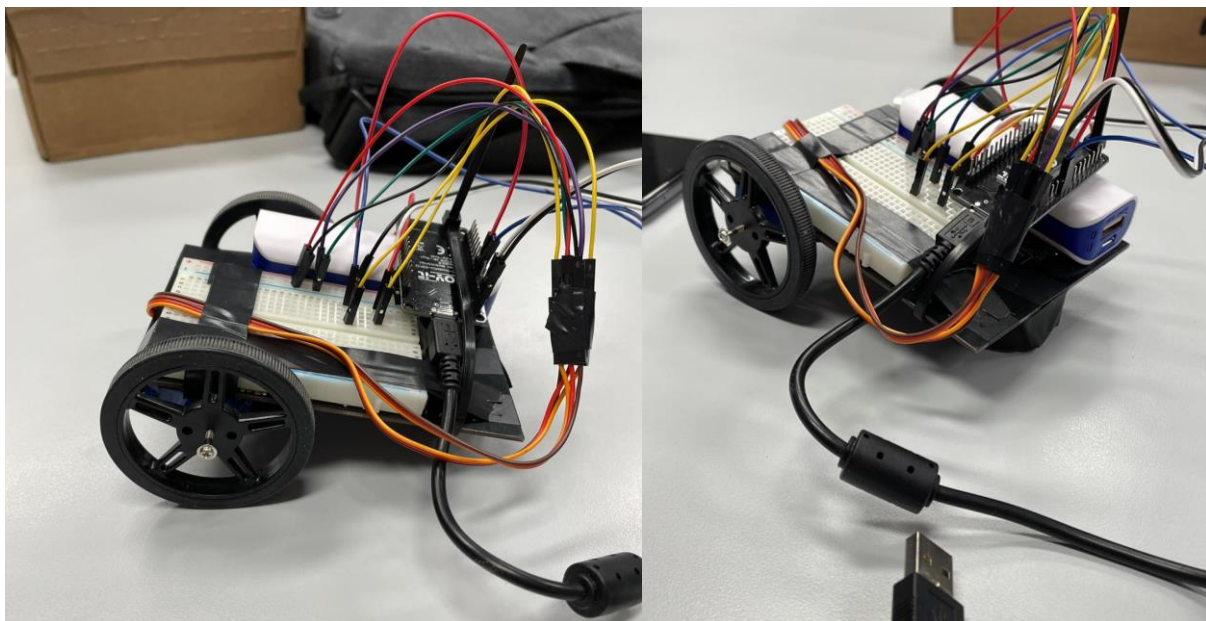
Slika 4.2. Pojednostavljena montažna shema klijentskog dijela

Da bi se lakše očitao pokret prsta ruke, senzori su postavljeni na rukavicu (slika 4.3). Senzori su korišteni na način da se čita naponska razina koja se klasificira kao *boolean* varijabla (0 ili 1) te odgovarajuća binarna kombinacija različito pokreće robota (naprijed, nazad, lijevo ili desno). Podatak o očitavanju se šalje bežično putem ESP-NOW protokola.



Slika 4.3. Slika serverskog dijela

Sve su komponente postavljene i pričvršćene na jednu pločicu. Osim motora i ESP32, na klijentskoj strani postavljeno je i napajanje.



Slika 4.4. Slika robota (klijentski dio)

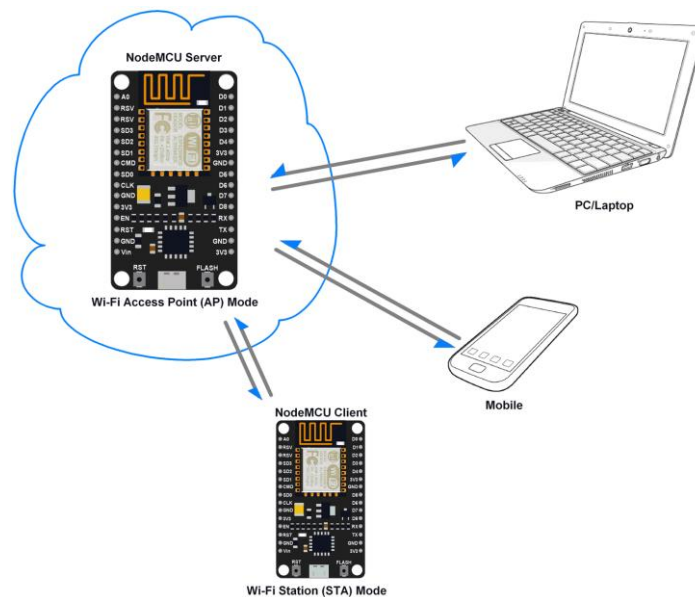
Kako bi sve uspješno komuniciralo i radilo potrebno je koristiti bežičnu vrstu komunikacije. U okviru ovog seminara, korišten je ESP-NOW protokol koji koristi WiFi komunikaciju za koji posjeduju podršku obje razvojne pločice (ESP8266 i ESP32).

4.1. ESP načini rada

ESP8266 i ESP32 podržavaju WiFi te se tako mogu postaviti u nekoliko različitih načina rada:

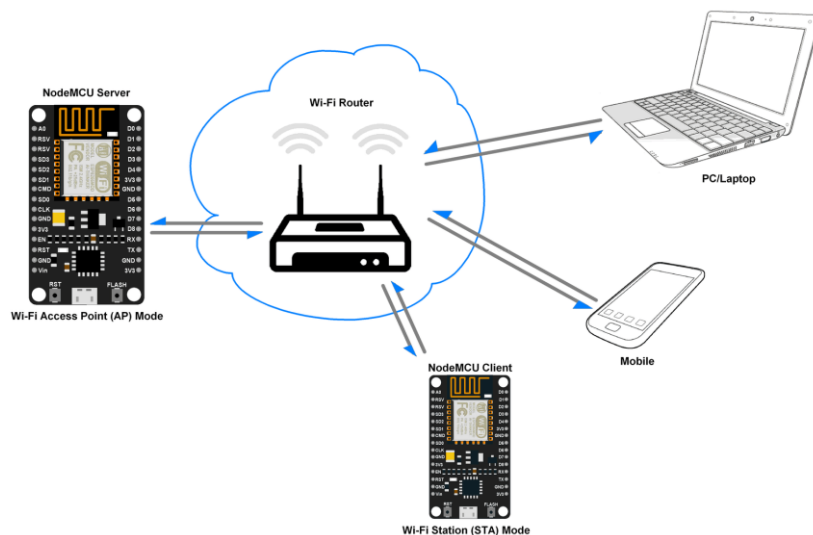
- *Access Point (AP)*
- *Station Mode (STA)*
- *Kombinirani način rada (AP i STA)*

Kada ESP radi kao AP, drugi uređaji (mobiteli, laptopi i sl.) mogu se povezati na pristupnu točku koju je postavio ESP. U ovom režimu rada, ESP se ne može povezati s vanjskom Wi-Fi mrežom koju nudi neki drugi uređaj (npr. usmjerivač, eng. *router*). Možemo postaviti SSID i lozinku za AP način koji će se koristiti za provjeru autentičnosti drugih uređaja tijekom povezivanja s njim.



Slika 4.2. ESP kao AP [12]

Kada ESP radi u STA načinu, može se povezati s Wi-Fi mrežom usmjerivača ili žarišne točke (npr. mobitela). Nijedan drugi uređaj ne može se povezati s ESP. ESP dobiva IP adresu od Wi-Fi usmjerivača na koji je povezan. S ovom IP adresom može djelovati kao HTTP poslužitelj na koji se može povezati bilo koji Wi-Fi uređaj.



Slika 4.3. ESP u STA načinu rada [12]

Zadnji način rada je da se ESP nalazi istovremeno u AP i STA načinu rada. To je zadani način rada koji je korišten u izradi bežične meteorološke postaje u sklopu ovog rada. U ovom načinu, ESP se može povezati s vanjskim Wi-Fi usmjerivačem i istovremeno postaviti pristupnu točku

za povezivanje drugih uređaja s njim. AP i STA su izolirani, npr. mobitel je povezan s pristupnom točkom koju postavlja ESP, a ESP je u STA načinu rada povezan s Wi-Fi mrežom usmjerivača. Tada samo ESP može pristupiti Internetu, a mobitel ne može [12].

4.2. ESP-NOW protokol

ESP-NOW je protkol kreiran od strane Espressifa kako bi omogućio ESP32 i ESP8266 pločicama da kreiraju privatnu *peer-to-peer* bežičnu mrežu. Protokol koristi njihove WiFi mogućnosti kako bi slali i primali podatke ali ESP32 i ESP8266 ne moraju biti povezani na Internet. ESP-NOW podržava *one-to-many* i *many-to-many* komunikaciju.

Kako bi uređaji uspješno komunicirali, potrebno ih je prethodno upariti. ESP-NOW podržava šifriranu i nešifriranu *unicast* komunikaciju, mješovite šifrirane i nešifrirane *peer* uređaje, slanje do 250 bajtova podataka te slanje *callbackova* koji mogu informirati aplikacijski sloj o (ne)uspješnom prijenosu podataka. Mane ESP-NOW protokola su ograničeni broj *peer* uređaja (do 20) te ograničenje podataka na 250 bajtova. Podržava jednosmjernu i dvosmjernu komunikaciju. U sklopu seminara korištena je jednosmjerna komunikacija budući da je potrebno da samo jedna pločica šalje podatke očitavanja, dok ih druga prima i koristi te podatke za pokretanje motora.



Slika 4.4. Jednosmjerna komunikacija [13]

Kako bi uspješno povezali uređaje, potrebno je poznavati MAC adresu na koju se šalju podaci. U našem slučaju je to MAC adresa klijenta (ESP32) koji prima podatke. Isto tako kako bi uopće mogli koristiti ESP-NOW komunikaciju, potrebno je postaviti ESP32 i ESP8266 u STA način rada [13][14].

Postoje odgovarajuće „uloge“ (eng. *roles*) koje se moraju postaviti prilikom korištenja ESP-NOW protokola:

- ESP_NOW_ROLE_IDLE – bez funkcije
- ESP_NOW_ROLE_CONTROLLER – postavljanje ESP-a kao *master*
- ESP_NOW_ROLE_SLAVE – postavljanje ESP-a kao *slave*
- ESP_NOW_ROLE_COMBO – kombinirano (*master* + *slave*)

5. ZAKLJUČAK

Kroz ovaj seminarski rad uspješno je realiziran robot s diferencijalnim pogonom upravljan pokretima ruke. Neki od problema ovako realiziranog sustava su: spora reakcija motora usprkos brznoj bežičnoj komunikaciji, neprecizno pozicionirani motori koji ne uzrokuju očekivano ponašanje robota (manja razlika između kretnje unaprijed i blagog skretanja), sporo očitavanje sa senzora. Cijeli projekt bi se mogao nadograditi izradom bolje konstrukcije koja bi riješila problem s pozicioniranjem motora te korištenje drugih, bržih senzora. Također za preciznije upravljanje mogu se koristiti druge vrste razvojnih pločica kao i druge vrste komunikacije (npr. Bluetooth komunikacija).

6. LITERATURA

- [1] „Software“, s Interneta, <https://www.arduino.cc/en/software>, 07.02.2023.
- [2] „Arduino IDE i prvi kod“, s Interneta, <https://e-radionica.com/hr/blog/2015/08/19/arduino-ide-i-prvi-kod/>, 07.02.2023.
- [3] „TSP-L-0012-103-1%-ST“, <https://hr.mouser.com/ProductDetail/Spectra-Symbol/TSP-L-0012-103-1-ST?qs=rVuZtzXoae7NeF2tMYDafg%3D%3D>, s Interneta, 07.02.2023.
- [4] „THINPOT DATASHEET“, https://hr.mouser.com/datasheet/2/381/THINPOT_DATA_SHEET_Rev_B-2949406.pdf, s Interneta, 07.02.2023.
- [5] „Internet stvari“, s Interneta https://hr.wikipedia.org/wiki/Internet_stvari, 07.02.2023.
- [6] „ESP8266 Pinout Overview [ESP-01, NodeMCU, WeMos D1 Mini], s Interneta, <https://diyi0t.com/what-is-the-esp8266-pinout-for-different-boards/>, 07.02.2023.
- [7] „NodeMCU razvojna pločica“, s Interneta, <https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/>, 07.02.2023.
- [8] „ESP32-DevKitC V4 Getting Started Guide“, <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html>, s Interneta, 07.02.2023.
- [9] „What is a Servo Motor? – Understanding the basics of Servo Motor Working“, <https://circuitdigest.com/article/servo-motor-working-and-basics>, s Interneta, 07.02.2023.
- [10] „SG90 Micro Servo Motor 9G for RC Robot Helicopter Aircraft“, <https://www.az-delivery.de/en/products/az-delivery-micro-servo-sg90>, s Interneta, 07.02.2023.
- [11] „SERVO MOTOR | BASICS, WORKING PRINCIPLE, THEORY, AND MORE!“, <https://www.solomotorcontrollers.com/blog/servo-motor/>, s Interneta, 07.02.2023.
- [12] „HTTP server on NodeMCU with Arduino IDE“, s Interneta, <https://www.electronicwings.com/nodemcu/http-server-on-nodemcu-with-arduino-ide>, 07.02.2023.
- [13] „Getting Started with ESP-NOW (ESP32 with Arduino IDE)“, <https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/>, s Interneta, 07.02.2023.
- [14] „ESP-NOW – Peer to Peer ESP32 Communications“, <https://dronebotworkshop.com/esp-now/>, s Interneta, 07.02.2023.