

Univerzitet u Sarajevu
Elektrotehnički fakultet
Ugradbeni sistemi 2022/23

Dokumentacija imlementacije i korisnička uputstva

Mentori: R. prof. dr. Samim Konjicija,
Mag. ing Selmir Gajip

Ime i prezime: Adna Herak,
Nerma Kadrić,
Azra Žunić

30.6.2023.

Sadržaj

1	Dokumentacija implementacije	3
2	Korisnička uputstva	4

1. Dokumentacija implementacije

Projekat "**Smart parking system**" je zasnovan na ideji koja podržava dva sistema koji su povezani.

Prvo smo osmislile parkiranje pomoću Mbot robota koji se programira u Arduino Uno, u programskom jeziku C.

Naime, prvo smo uključile potrebne biblioteke za rad s robotom. Zatim, inicijalizirale smo na pinove 9 i 10 motore koji su sastavni dio robota, kao i dva senzora:

- ultrasonični senzor koji se nalazi na bočnoj strani robota i služi za mjerenje distance i pomaže robotu da uoči prazan prostor na parking mjestu (pin 3);

- senzor za praćenje linije koji se nalazi na prednjoj strani robota (pin 2).

Pomoćna funkcija **move** prima dva parametra: smjer i brzinu, te se motori postavljaju na određene vrijednosti u zavisnosti od toga kako želimo da se robot kreće.

Zatim definišemo threshold (prag) na 500, brzinu kretanja na 150 i distancu na 15. Pomoćna funkcija **setup** predstavlja očitavanje senzora za praćenje linije i u zavisnosti od slučaja određujemo kretanje. Pratilac linije ima 2 senzora koji mogu detektovati bijelu površinu (u rasponu od 1-2 cm).

Radi tako što emituje IR (Infracrveno) svjetlo i bilježi koliko se reflektuje natrag.

Ako se mnogo reflektuje nazad, može se zaključiti da je blizu bijele površine.

Ako se malo reflektuje nazad, može se zaključiti da je površina crna ili senzor nije blizu površine.

Na primjer, ukoliko je **linefollower_2.readSensors()==3.00000** znači da je očitana **bijela boja**, a ukoliko je **linefollower_2.readSensors()<=2.00000** očitana je **crna**.

Brojač i broji koliko puta je robot skrenuo kako bismo pravilno parkirale isti.

Funkcije **loop** i **_loop** utiču na kontinualni rad samog programa.

Također ćemo opisati kod u Micropython-u koji upravlja radom sa Raspberry Pi PicoETF. Dva ultrasonična senzora kao i display Banggood su povezani na sam sistem PicoETF.

Naime, prvo smo importovale određene biblioteke potrebne za rad sa displayem i senzorima. Dio za display smo iskoristile s prethodnih vježbi, tj. dio za ispis poruke.

Zatim smo pomoću dokumentacije za servo-motor uključile klasu servo kako bismo motor inicijalizirale i postavile na pin 28.

Pomoćna funkcija **ultra** postavlja trigger i echo na određene vrijednosti, te ukoliko je robot primijećen na distanci **većoj od 30cm a manjoj od 45cm**, dolazi do otvaranja rampe (pod određenim uglom) koja je postavljena na ulazu na parking.

Zatim, imamo i pomoćnu funkciju **slobodnomjesto** koja u zavisnosti od distance očitane pomoću senzora na displeju ispisuje status parkinga (prazan/pun).

Na kraju, tu je i main funkcija koja omogućava kontinualni rad programa.

Na opisan način je povezana komunikacija između robota senzora, motora i sistema.

Detaljniji rad sistema je opisan u samoj specifikaciji.

2. Korisnička uputstva

Da bi potencijalni korisnik koristio naš projekat, potrebno je, naravno, da ima odgovarajuće komponente. Korištenje je poprilično jednostavno: potrebno je robota postaviti na početak crne linije (možemo je nazvati cestom), zatim žičano povezati robot na računar te ga jednostavno uključiti na dugme koje se nalazi s njegove desne strane. Razvojni sistem (picoETF) također je potrebno žičano spojiti s računarom, a komponente: servo motor, display te senzor potrebno je spojiti prema šemi koja se nalazi u dokumentaciji projekta. Kod je pisan u dva okruženja: Thonny i Arduino Uno te su skripte postavljenje u prilogu također. Potrebno je pokrenuti kod pritiskom na dugme Run odnosno Upload u okruženjima. Kada se upload završi, potrebno je pokrenuti robota na dugme s njegove prednje strane. Sve ostalo će robot sam uraditi u saradnji s ostalim komponentama. Kada se parkira, sam će se zaustaviti.