



УНИВЕРЗИТЕТ  
У НОВОМ САДУ



ФАКУЛТЕТ  
ТЕХНИЧКИХ НАУКА

Трг Доситеја Обрадовића 6, 21000 Нови Сад, Република Србија  
Деканат: 021 6350-413; 021 450-810; Централa: 021 485 2000  
Рачуноводство: 021 458-220; Студентска служба: 021 6350-763  
Телефакс: 021 458-133; e-mail: [ftndeans@uns.ac.rs](mailto:ftndeans@uns.ac.rs)

ИНТЕГРИСАНИ  
СИСТЕМ  
МЕНАЏМЕНТА  
СЕРТИФИКОВАН ОД:



## PROJEKAT IZ PREDMETA AUTOELEKTRONIKA

**Naziv projekta:**

Parking senzor

**Mentor projekta:**

Dr Miodrag Brkić

**Studenti:**

Jovan Slavujević EE 239/2015

Katarina Ninković EE 32/2015

Septembar 2019.

## Sadržaj:

1.Uvod .....	3
2.Analiza zadatka .....	4
3. Korištene komponente .....	5
3.1. Mikrokontroler DSPIC30F4013 .....	5
3.2. Ultrazvučni senzor HC-SR04 .....	7
3.3. Can modul MCP2515.....	8
3.4. Arduino.....	10
4. Algoritam rada.....	11
4.1. Algoritam očitavanja senzora .....	11
4.2. Algoritam Arduina .....	12
4.3 Algoritam parking senzora (dsPIC) .....	13
5. MISRA C pravila kodovanja.....	15
6.Zaključak.....	16
7. Literatura.....	17

## 1. Uvod

Tema ovog projekta je uspostavljanje *CAN* komunikacije između *dsPIC*-a i *Arduino Uno*, slanja informacija sa senzora ka *PIC*-u, ispis tih informacija na terminalu preko *UART*-a, kao i slanje naredbi sa *Arduina* ka *PIC*-u preko *Serial* Monitora i slanje informacija sa senzora od *PIC*-a ka *Arduinu*, te ispis na *Serial* Monitoru.

Današnja vozila su veoma kompleksne mašine koje sadrže mehaničke i elektronske delove. Poslednjih godina, broj elektronskih komponenata koje se koriste u vozilima ubrzano raste. Povećanje zahteva za bezbednost, udobnost i performanse dovelo je do toga da se u moderna vozila dodaje mnogo više elektronskih komponenti. Došlo je do povećanja potražnje za povezivanjem ovih elektronskih komponenata na takav način da mogu komunicirati pouzdano, bezbedno i u realnom vremenu. *CAN* komunikacija zadovoljava ove zahteve. Takođe, *CAN* komunikacija lako otkriva neispravi modul u elektronskom sistemu.

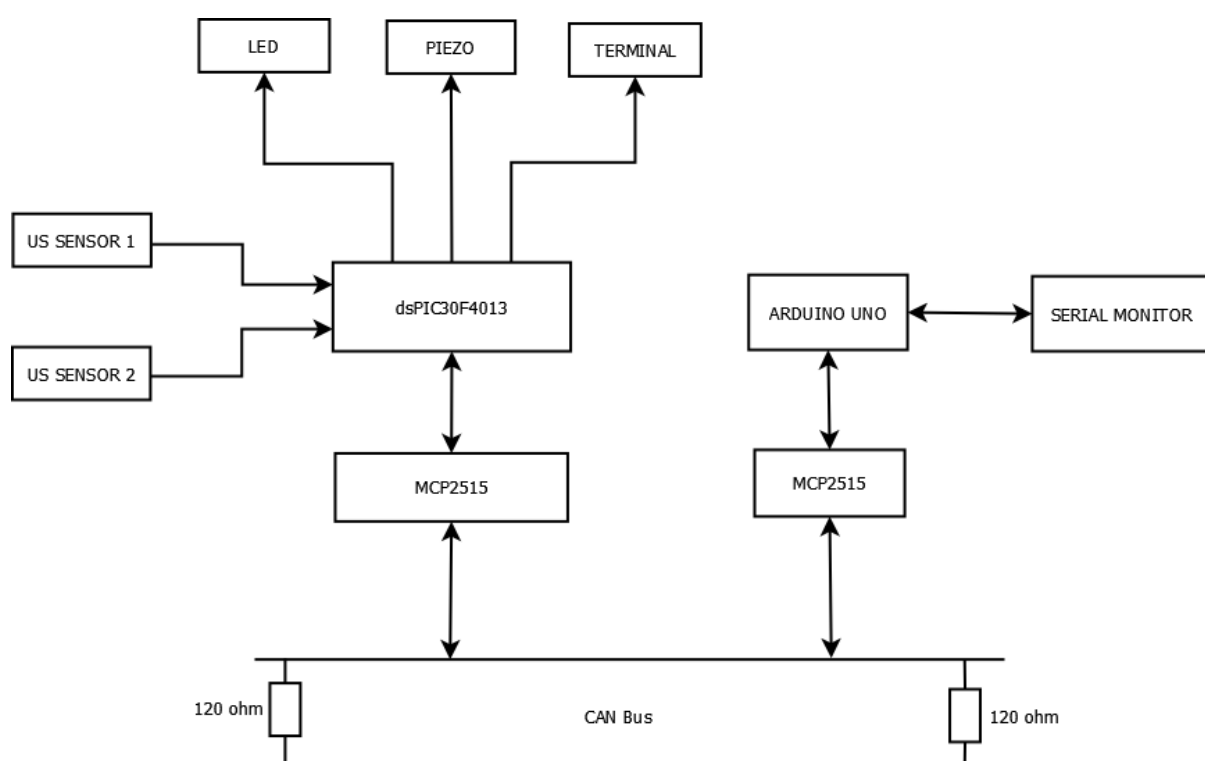
U drugom poglavlju je obrađena analiza zadatka. U trećem poglavlju opisani su detalji predmeta projekta. U četvrtom poglavlju je opisan algoritam rada programa. Poslednja poglavlja sadrže zaključak i literature koju smo koristili, kao i nešto o *MISRA* pravilima kodovanja. U dodatku se nalaze slike na kojima je uputstvo kako se povezuje *dsPIC* i *Arduino* sa modulom *MCP2515*.

## 2. Analiza zadatka

Projekat je podeljen u tri zadatka. Prvi zadatak je omogućiti merenje ultrazvučnih senzora – levog i desnog, povezanih na pinove *dsPIC*-a, perioda merenja je pola sekunde. Takođe, omogućiti uvid u vrednosti merenja preko ispisa na terminalu.

Drugi zadatak je uspostaviti *CAN* komunikaciju između *dsPIC*-a i *Arduina*, posredstvom *MCP2515* modula i omogućiti slanje naredbi sa *Serial* monitora do *PIC*-a preko *Arduina*. Naredbe se odnose na kalibraciju oba senzora zasebno, kao i za upravljanje nad piezom (*mute/unmute*), kao i slanje ka *Arduinu*.

Treći zadatak je da se generišu zvučni signali na piezu nakon odrađene kalibracije tako da imamo kontinualan zvuk kada je prepreka ispod donje granice kalibracije, brzo-pulsirajući zvuk kada prepreku imamo između donje granice i sredine, sporo-pulsirajući zvuk kada se prepreka nalazi između sredine i gornje granice, a da nema zvuka kada je prepreka iznad gornje granice.



Slika 2.1 Idejna blok šema

### 3. Korištene komponente

#### 3.1. Mikrokontroler DSPIC30F4013

Mikrokontroler je uređaj opšte namene, koji pribavlja podatke, obavlja ograničenu obradu nad tim podacima, i upravlja svojim okruženjem na osnovu rezultata izračunavanja. Mikrokontroler u toku svog rada koristi fiksni program koji je smešten u *ROM*-u i koji se ne menja u toku životnog veka sistema.

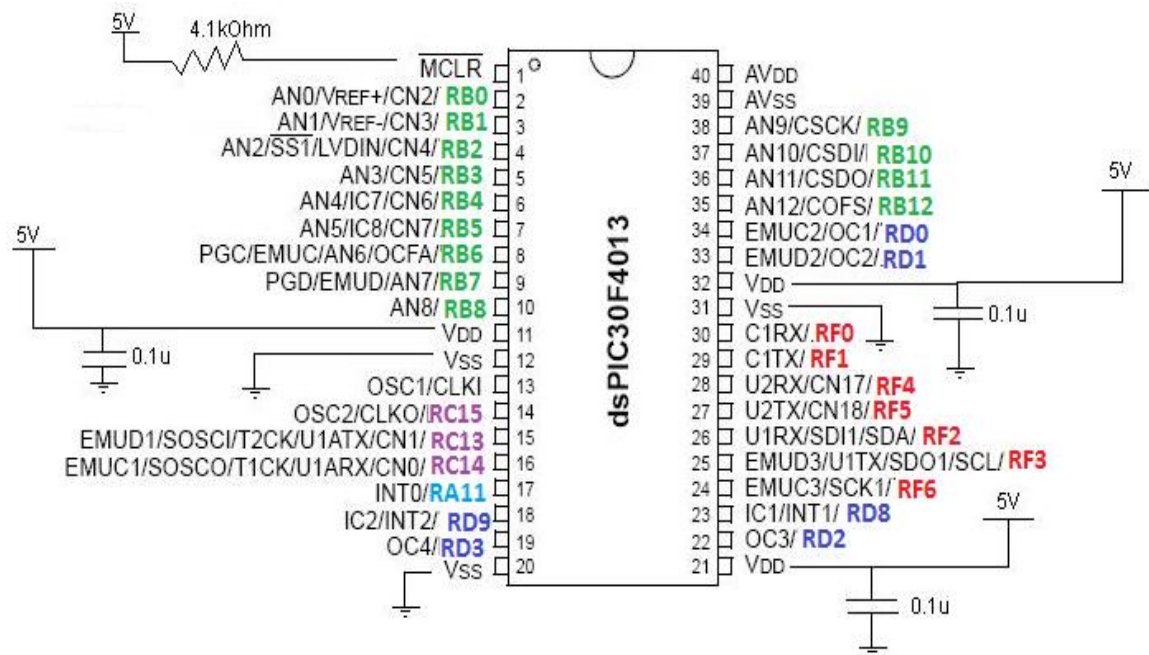


Slika 3.1.1 fizički izgled mikrokontrolera

Osnovne karakteristike mikrokontrolera *dsPIC30F4013*:

- Radni napon od  $2.5\text{ V} \div 5.5\text{ V}$
- Radna frekvencija od  $4\text{ MHz} \div 10\text{ MHz}$  za interni oscilator
- Programska memorija kapaciteta  $48\text{ kb}$
- Memorije za podatke *RAM* kapaciteta  $2\text{ kb}$
- Memorije za podatke *EEPROM* kapaciteta  $1\text{ kb}$
- Prekidi (do 33 izvora prekida)
- U/I portovi: B, C, D, E, F
- Pet tajmera:
  - Timer1: 16-bitni tajmer/brojač događaja
  - Timer2/3: 32-bitni tajmer/brojač događaja
  - Timer4/5: 32-bitni tajmer/brojač događaja
- 12-bitni Analogni-Digitalni (A/D) konvertor
- Komunikacije: *SPI*, *UART*, *I<sup>2</sup>C*
- Mala potrošnja, velika brzina rada

Portovi mikrokontrolera *dsPIC30F4013*:



Slika 3.1.2 pinovi mikrokontrolera

Pinovi mikrokontrolera mogu biti ulazni i izlazni. Takođe postoje i digitalni i analogni pinovi, gde su analogni samo iz porta B.

### 3.2. Ultrazvučni senzor HC-SR04

*HC-SR04* nam omogućava beskontaktno merenje razdaljine uz preciznost merenja do 3mm. Modul se sastoji od predajnika, prijemnika i kontrolne elektronike. Senzor poseduje ukupno 4 pina: *VCC*, *Trig*, *Echo* i *GND*.



Slika 3.2.1 fizički izgled ultrazvučnog senzora

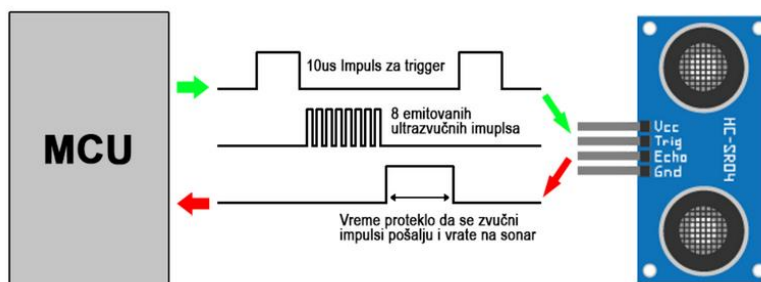
Osnovne karakteristike:

- Radni napon 5 V
- Strujna potrošnja 15 mA
- Radna frekvencija 40 KHz
- Merni opseg od 2 cm ÷ 400 cm
- Ugao merenja 15 stepeni

Način rada:

Na *trigger* ulazu potrebno je generisati impuls trajanja minimum 10  $\mu$ s, nakon čega će modul automatski emitovati osam zvučnih impulsa visoke frekvencije (Slika 3.2.2) i setovati *echo* pin na visok nivo. *Echo* ostaje visok sve dok modul ne primi nazad emitovane impulse, koji se odbijaju od prepreke i vraćaju nazad ka modulu. Rastojanje je direktno proporcionalno vremenu koje je potrebno talasima da se vrate na prijemni deo, odnosno vremenu trajanja visokog nivoa na *echo* pinu. Za računanje rastojanja koristimo sledeću formulu:

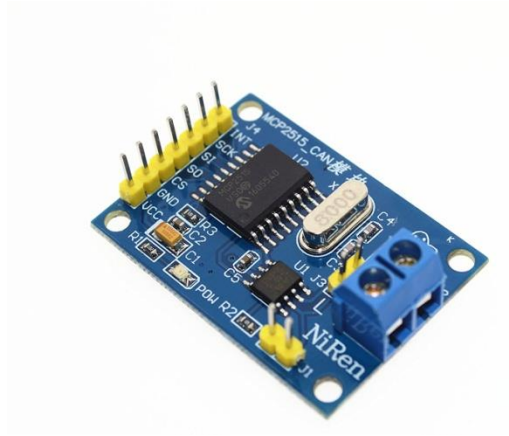
$$Distanca = \frac{Brzina_{ULTRAZVUKA} * Vreme_{PUTOVANJA}}{2} \quad [3.2.1]$$



Slika 3.2.2 generisanje impulse

### 3.3. Can modul MCP2515

*MCP2515* je *Microchip*-ov *CAN* modul. Radi kao *transceiver*, jer je zadužen i za prijem i za emitovanje podataka. Ovaj čip ima dve prihvatne maske i šest filtera koji se koriste za filtriranje neželjenih poruka i na taj način smanjuje troškove matičnog *MCU*-a. Čip *MCP2515* sa mikrokontrolerom komunicira preko *SPI* protokola.



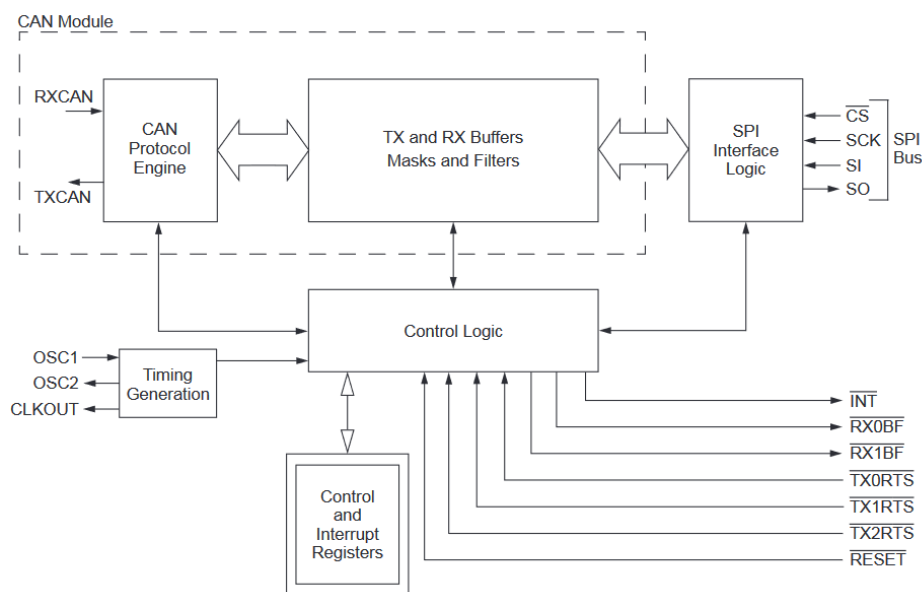
Slika 3.3.1 fizički izgled MCP2515 CAN modula

Karakteristike:

- Ugrađuje *CAN V2.0B* pri brzini od 1 *Mb/s*
- Prijemni baferi, maske i filteri:
  - Dva prijemna bafera sa prioritetskim skladištenjem poruka
  - Šest 29-bitnih filtera
  - Dve 29-bitne maske
- Tri zaštitna odstupanja sa funkcijama prioriteta i prekida
- *SPI* interfejs velike brzine (10 *MHz*)
- *CMOS* tehnologija male snage:
  - Radni napon od 2.7 *V* ÷ 5.5 *V*
  - Aktivna struja 5 *mA*

Jednostavan blok dijagram *MCP2515* sastoji se iz tri glavna bloka kao što je prikazano na sledećoj slici.





Slika 3.3.2 Blok dijagram mcp2515

1. CAN modul, koji uključuje mehanizam CAN protokola, maske, filtere, prenosne i prijemne bafere. Obrađuje sve funkcije za prijem i prenos poruka na CAN bus. Poruke se prenose prvo učitavanjem odgovarajućih registara i kontrolnih registara a prenos se započinje upotrebom kontrolnih bitova registra preko SPI interfejsa ili upotrebom pinova za omogućavanje prenosa.

2. Logika upravljanja i registri koji se koriste za konfiguriranje uređaja i njegov rad. Kontrolni logički blok kontroliše podešavanje i rad MCP2515 tako što se međusobno povezuju sa ostalim blokovima kako bi prosleđivali informacije i upravljali njima. Interrupt pinovi obezbeđuju veću fleksibilnost sistema. Postoji jedan višenamenski interrupt pin za svaki od registara za prijem, koji se može koristiti za označavanje primljene poruke i učitavanje u jedan od prijemnika.

3. Blok SPI protokola, MCU pristupa uređaju preko SPI interfejsa. Pisanje i čitanje iz svih registara se izvodi pomoću standardnih SPI naredbi za čitanje i pisanje, pored specijalizovanih SPI komandi.

### 3.4. Arduino

*Arduino Uno* je platforma zadužena za izvršavanje elektronike otvorenog koda koji se zasniva na fleksibilnom hardveru i softveru, pritom je jednostavan za upotrebu.

*Arduino Uno* sadrži 14 digitalnih ulazno-izlaznih pinova (od kojih se 6 mogu koristiti kao *PWM* izlazi), 6 analognih ulaza, 16 *MHz* kristalni oscilator, *USB* priključak, priključak za napajanje, *ICSP* zaglavlje i taster za resetovanje. Sadrži sve što je potrebno za podršku mikrokontrolera.



Slika 3.4.1 fizički izgled Arudino Uno

Karakteristike:

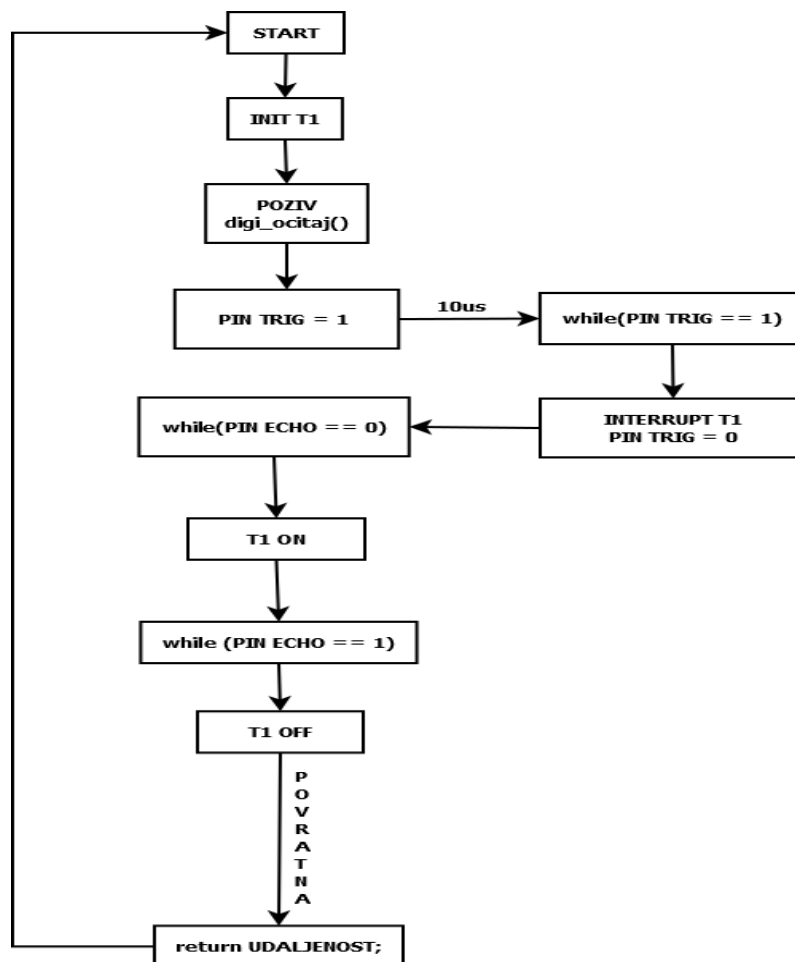
- Mikrokontroler *ATmega328*
- Radni napon 5 V
- Ulazni napon (preporučeno) od 7 V ÷ 12 V
- Ulazni napon (granice) od 6 V ÷ 20 V
- Digitalni ulazni/izlazni pinovi 14 (od kojih 6 obezbeđuje *PWM* izlaz)
- Analogni ulazni pinovi 6
- *Flash* memorija 32 kb
- *SRAM* 2 kb
- *EEPROM* 1kb
- Brzina takta 16 *MHz*

## 4. Algoritam rada

Pre nego što se uđe u “beskonačnu petlju” neophodna je inicijalizacija sistema. U okviru toga imamo inicijalizaciju *SPI*, *UART*, pa i samog *CAN*-a, kao i inicijalizaciju tajmera koje sistem koristi, piezo zvučnika i indikatorske *LED* diode.

### 4.1. Algoritam očitavanja senzora

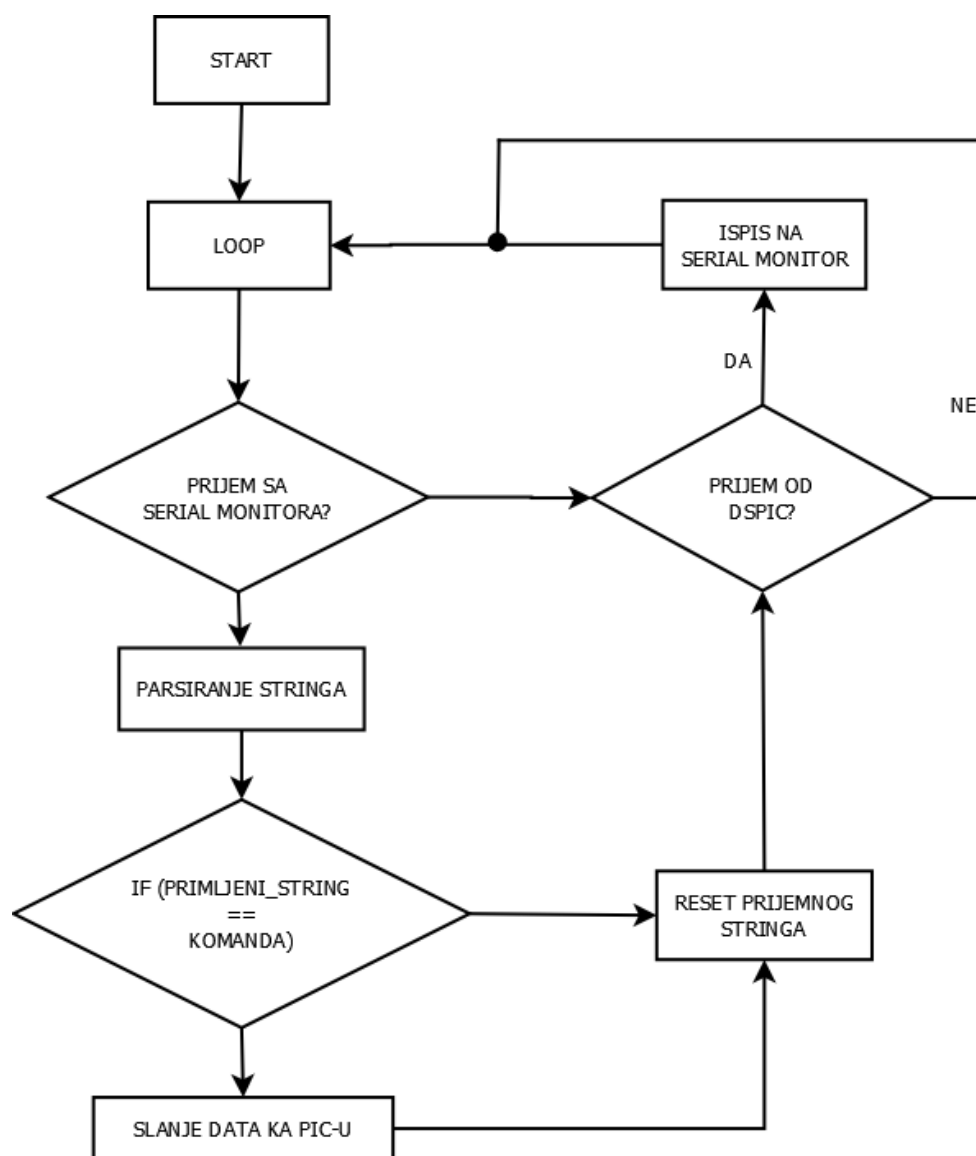
Dva brojača se koriste za očitavanje senzora, jedan za levi, a drugi za desni. Isti brojač se koristi i za generisanje impulsa trajanja od  $10\ \mu\text{s}$  i za prijem o informaciji razdaljine od strane senzora. Na sledećoj slici je predstavljen algoritam rada očitavanja senzora:



Slika 4.1.1 algoritam rada senzora

## 4.2. Algoritam Arduina

Uloga *Arduino* sistema je da komandu unetu sa *serial* monitora primi kao string, zatim da ga isparsira i proveri da li je poslata postojeća komanda ili “smeće”. Ukoliko primeljeni string odgovara nekoj od postojećih komandi (*START*, *STOP*, *CALMINI*/2, *CALMAXI*/2) onda šalje odgovarajuću kombinaciju bajtova ka *dsPIC*-u, a zatim briše *buffer* string i čeka novu komandu.



Slika 4.2.1 algoritam rada Arduino sistema

### 4.3 Algoritam parking senzora (dsPIC)

Algoritam parking senzora se svodi na izvršavanje beskonačne petlje koja se sastoji od periodičnog očitavanja informacija sa levog i desnog senzora, te njihovog ispisa na terminal, što izvršava *senzori()* funkcija, slanja ka *Arduinu*, što izvršava *sendMsgBuf()*, kao i na čekanju prijema nekih komandi sa *CAN*-a od *Arduina* (*CAN\_receive()* funkcija).

Kada se prijem podataka od *Arduina* dogodi postavlja se pitanje o kakvoj komandi je zapravo reč, to se proverava na osnovu primljenog niza bajtova, njihovo parsiranje se izvršava uz pomoć funkcije *komanda()*, koja je celobrojnog tipa, a broj koji vrati predstavlja neku komandu koju treba sistem da izvršava. Komande *CALMIN* i *CALMAX* odnose se na kalibraciju senzora. Kalibracija se izvršava posebno za svaki senzor, preko funkcije *kalibracija()*, a suština je u tome da pri slanju *CALMIN*-a za jedan senzor trenutno očitana vrednost sa tog senzora bude postavljena kao donja granica, dok *CALMAX*-a postavlja gornju granicu. Kalibrisana vrednost se dobija po sledećoj formuli:

$$Cal = \frac{digi_{trenutni} - donja_{gr}}{gornja_{gr} - donja_{gr}} * 100 \quad [4.3.1]$$

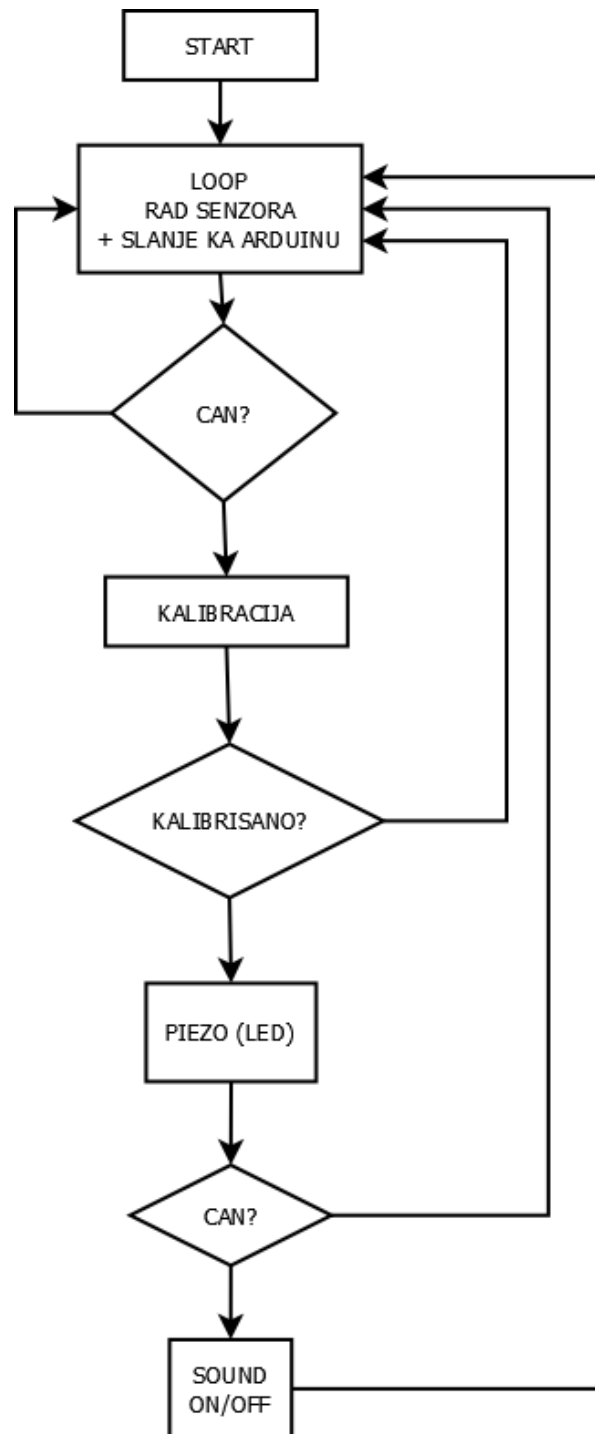
Ukoliko je donja granica veća od gornje, dolazi do greške, dodeljene granice se brišu i traži se ponavljanje kalibracije. Ukoliko pak nije došlo do takve greške, smatra se da je kalibracija uspešno izvršena za jedan senzor, što je dovoljno da započne rad piezo-*LED* programa.

Kada se prepreka ispred kalibrisanog senzora nalazi „iznad opsega“, odnosno iznad gornje granice udaljenosti, ne dolazi do generisanje zvuka, kao ni do indikacije *LED* diode. Za opseg od (50% ÷ 100%) dolazi do generisanja sporo-pulsirajućeg zvuka i svetljenja diode. Za opseg od (0% ÷ 50%) dolazi do brzo-pulsirajućeg, a „ispod opsega“ imamo indikaciju diode i kontinualan zvuk piezo zvučnika.

Treba napomenuti da dok se ovo izvršava može se raditi i kalibracija drugog senzora, kao i ponovno kalibrisanje već kalibrisanog senzora, menjajući njegove granice, ali i utišavanje zvučnika preko komandi *STOP/START*, što izvršava funkcija *zvuk()*.

Takođe, kako bi *CAN* prijem bio moguć u bilo kom trenutku bez smetnji, generisanje isprekidanog zvuka, odnosno naizmenično paljenje i gašenje

zvučnika, kao i LED diode, ali i periodično očitavanje senzora i ispis na terminalu realizovani su bez softverskog kašnjenja, uz pomoć *interrupt*-a tajmera tako što koristimo promenljivu koja se inkrementira unutar njega. Promenljiva neće čekati da prođe vreme, već izvršava komande kada dođe vreme za to, odnosno kada bude inkrementirana dovoljno da prođe *if*( ), za razliku od *delay*( ) funkcije.



Slika 4.3.1 Algoritam rada parking senzora

## 5. MISRA C pravila kodovanja

Kod u programskom jeziku je zadovoljio *MISRA (Motor Industry Software Reliability Association)* 2012 pravila. *MISRA* pravila predstavljaju set pravila koja kada se zadovolje omogućavaju da kod napisan u programskom jeziku C bude bezbedan, prenosiv i pouzdan za *embedded* sisteme. *MISRA* razlikuje *mandatory*, *required* i *advisory* pravila. *Mandatory* pravila moraju biti zadovoljena, inače neće proći kompajliranje, bez zadovoljenih *required* pravila kod će proći kompajliranje ali ona ipak moraju biti ispoštovana i poštovanje *advisory* pravila predstavlja dobru praksu ali može i bez njih.

## 6.Zaključak

Projekat je uspešno realizovan. Uspostavljena je komunikacija između dva različita razvojna sistema – *dsPIC*-a i *Arduino*-a posredstvom *CAN*-a. Omogućen je veoma brz i efikasan odziv i prenos željenih informacija među njima.

Takođe, ultrazvučni senzori su zadovoljili očekivane rezultate, te dobro očitavaju merene udaljenosti prepreka. Pri izvršenoj kalibraciji dobija se dobar odziv piezo-*LED* programa.

Ostaje žal što sistem nije realizovan uz pomoć *FreeRTOS*-a, odnosno nije realizovan rad u realnom vremenu što bi poboljšalo performanse i rad samog parking senzora.



## 7. Literatura

- [1] Microchip Technology Inc. “dsPIC30F3014, dsPIC30F4013“ „Data Sheet“. [Online].  
Dostupno: <http://ww1.microchip.com/downloads/en/devicedoc/70138c.pdf>.  
[Pristup: 10-Avg-2019].
  
- [2] Katedra za elektroniku. “Primenjena elektronika 2018/19“. [Online].  
Dostupno: <http://ellab.ftn.uns.ac.rs/moodle/course/view.php?id=713>.  
[Pristup: 25-Avg-2019].
  
- [3] Microchip Technology Inc “MCP2515” Data Sheet, [Online].  
Dostupno: <https://www.mouser.com/datasheet/2/268/20001801H-708845.pdf>.  
[Pristup: 16-Sep-2019].