



**УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**



**УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА
НОВИ САД
Департман за рачунарство и аутоматику
Одсек за рачунарску технику и рачунарске комуникације**

ДИПЛОМСКИ – МАСТЕР РАД

Кандидат: Стојан Ковачевић

Број индекса: 11230

Тема рада: Једно решење система за бежично позиционирање рефлекторског места

Ментор рада: др Никола Теслић

Нови Сад, фебруар 2010.



УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА
21000 НОВИ САД, Трг Доситеја Обрадовића 6

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:	
Идентификациони број, ИБР:	
Тип документације, ТД:	Монографска документација
Тип записа, ТЗ:	Текстуални штампани материјал
Врста рада, ВР:	Дипломски – мастер рад
Аутор, АУ:	Стојан Ковачевић
Ментор, МН:	проф. Др Никола Теслић
Наслов рада, НР:	Једно решење система за бежично позиционирање рефлекторског места
Језик публикације, ЈП:	Српски / латиница
Језик извода, ЈИ:	Српски
Земља публиковања, ЗП:	Република Србија
Уже географско подручје, УГП:	Војводина
Година, ГО:	2010.
Издавач, ИЗ:	Ауторски репринт
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6
Физички опис рада, ФО: (поглавља/страна/ цитата/табела/слика/графика/прилога)	5/34/8/2/19/0/0
Научна област, НО:	Електротехника и рачунарство
Научна дисциплина, НД:	Рачунарска техника
Предметна одредница/Кључне речи, ПО:	Аутоматизација, DC мотор, ИШМ (PWM), LIN, Pozicija, Reflektor, Senzor, SimpliciTI
УДК	
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад
Важна напомена, ВН:	
Извод, ИЗ:	У овом раду представљена је реализација система за бежично управљање рефлекторским местима. Бежична комуникација одвија се путем SimpliciTI протокола, између персоналног рачунара и микроконтролера унутар рефлекторског места. Поред микроконтролера који је и примопредајник за бежичну комуникацију, свако рефлекторско место садржи и два мотора једносмерне струје који врше позиционирање, а контролисани су импулсно-ширинском модулацијом. За одређивање тренутне позиције користе се робусни сензори на бази Холовог ефекта, чиме је затворен процес управљања.
Датум прихватања теме, ДП:	25.01.2010.
Датум одбране, ДО:	18.02.2010.
Чланови комисије, КО:	Председник: Др Михајло Катона
	Члан: Др Небојша Пјевалица
	Члан, ментор: Др Никола Теслић
	Потпис ментора



KEY WORDS DOCUMENTATION

Accession number, ANO :	
Identification number, INO :	
Document type, DT :	Monographic publication
Type of record, TR :	Textual printed material
Contents code, CC :	Master Thesis
Author, AU :	Stojan Kovačević
Mentor, MN :	Nikola Teslić, PhD
Title, TI :	Wireless reflector positioning system solution
Language of text, LT :	Serbian
Language of abstract, LA :	Serbian
Country of publication, CP :	Republic of Serbia
Locality of publication, LP :	Vojvodina
Publication year, PY :	2010.
Publisher, PB :	Author's reprint
Publication place, PP :	Novi Sad, Dositeja Obradovica sq. 6
Physical description, PD : <small>(chapters/pages/ref./tables/pictures/graphs/appendixes)</small>	5/34/8/2/19/0/0
Scientific field, SF :	Electrical Engineering
Scientific discipline, SD :	Computer Engineering, Engineering of Computer Based Systems
Subject/Key words, S/KW :	Automation, DC motor, PWM, LIN, Position, Reflector, Sensor, SimpliciTI
UC	
Holding data, HD :	The Library of Faculty of Technical Sciences, Novi Sad, Serbia
Note, N :	
Abstract, AB :	<p>This paper presents an overview of wireless reflector positioning system implementation. It is based on a SimpliciTI wireless protocol interfacing Microcontroller inside reflector to a PC application. Microcontroller is also used to control two DC motors sending them PWM signal. The motors move reflector housing around two perpendicular axes, allowing it to point on a specific spot in the space below. Control process is closed with robust Hall-Effect based sensors mounted close to the shafts, monitoring the position and sending position data back to the Microprocessor via LIN2.0 interface.</p>
Accepted by the Scientific Board on, ASB :	25.01.2010.
Defended on, DE :	18.02.2010.
Defended Board, DB :	President: Mihajlo Katona, PhD
	Member: Nebojša Pjevalica, PhD
	Member, Mentor: Nikola Teslić, PhD
	Menthor's sign

SADRŽAJ

1. Uvod.....	1
2. Fizička struktura sistema.....	3
2.1 Izbor protokola za bežičnu komunikaciju.....	4
2.2 Mikrokontroler	5
2.3 Pogonski motori	7
2.4 Upravljačka elektronika.....	9
2.5 Senzori	11
2.5.1 Unutrašnja struktura senzora	12
2.5.2 Fizička sprega mikrokontrolera sa LIN magistralom.....	13
3. Programska podrška.....	15
3.1 Programska podrška za mikrokontroler.....	16
3.1.1 Realizacija LIN 2.0 protokola.....	17
3.1.2 Bežični protokol SimpliciTI	22
3.1.3 Upravljanje motorima	26
3.2 Programska podrška za personalni računar	28
4. Zaključak.....	33
5. Literatura	34

SPISAK SLIKA

Slika 2.1 – Izgled sistema.....	3
Slika 2.2 – Blok dijagram sistema	4
Slika 2.3 – Razvojni sistem <i>Chipcon SmartRF04EB</i> , modul sa <i>CC2510</i> mikrokontrolerom, USB periferija.....	5
Slika 2.4 – Unutrašnja struktura mikrokontrolera <i>CC2510/2511</i>	6
Slika 2.5 – Motor jednosmerne struje i pužni prenos	8
Slika 2.6 – Struktura integrisanog kola <i>SA-57</i>	9
Slika 2.7 – Montaža hladnjaka na integrisano kolo <i>SA57</i>	10
Slika 2.8 – Postavka magnet-senzor	11
Slika 2.9 – Struktura <i>HAL2810</i> senzora.....	12
Slika 2.10 – LIN magistrala	13
Slika 2.11 – Unutrašnja struktura integrisanog kola <i>TPIC1021</i>	13
Slika 3.1 – Struktura sistema.....	15
Slika 3.2 – Algoritam programske podrške za mikrokontroler	17
Slika 3.3 – Paket podataka	18
Slika 3.4 – Paket podataka snimljen osciloskopom.....	18
Slika 3.5 – Opšta topologija <i>SimpliciTI</i> bežične reže	23
Slika 3.6 – <i>Packet Sniffer</i>	24
Slika 3.7 – Struje motora sa i bez laganog starta (levo) i lagani start (desno).....	27
Slika 3.8 – Korisnička aplikacija.....	28

SPISAK TABELA

Tabela 2.1 – Povezivanje mikrokontrolera i razvojne ploče sa SA57.....	10
Tabela 3.1 – Formiranje <code>directions_table[9]</code>	26

SKRAĆENICE

GND	- Oznaka za signal na nultom potencijalu
LAN	- <i>Local Area Network</i>
PoE	- <i>Power over Ethernet</i>
RAM	- <i>Random Access Memory</i>
USART	- <i>Universal Synchronous/Asynchronous Receiver/Transmitter</i>
IŠM (PWM)	- <i>Impulsno-širinska modulacija (Pulse-Width Modulation)</i>
ISM	- <i>Industrial, Scientific, Medical frequency range</i>
LIN	- <i>Local Interconnect Network</i>
CAN	- <i>Controller Area Network</i>
USB	- <i>Universal Serial Bus</i>
TI	- <i>Texas Instruments</i>

1. Uvod

Postojeća rešenja scenske rasvete sadrže veliki broj kablova, kao i komplikovane i glomazne upravljačke jedinice. Pri dodavanju novih reflektorskih mesta često dolazi do nedostatka slobodnih mesta na kontrolnoj jedinici. To onemogućuje proširenje sistema rasvete, ili ga poskupljuje. Pored toga, rukovanje takvim uređajima je prilično komplikovano.

Rešenje predstavljeno u ovom radu omogućava jednostavno bežično povezivanje više reflektorskih mesta na jedinstvenu kontrolnu jedinicu u vidu personalnog računara. Bežično upravljanje pored smanjenja broja neophodnih kablova za povezivanje donosi i jednostavnost u proširenju sistema, odnosno dodavanja novih reflektorskih mesta. Upotrebom personalnog računara kao centralne upravljačke jedinice i intuitivnih periferija poput kontrolne palice (*joystick-a*) rukovanje se značajno pojednostavljuje. Korišćenjem resursa personalnog računara redosled događaja može biti unapred vremenski definisan, snimljen i kasnije reprodukovano po tačno zadatom vremenskom rasporedu.

Zadatak ovog diplomskog rada je projektovanje i realizacija navedenog sistema za bežično pozicioniranje reflektorskog mesta pomoću personalnog računara, u realnom vremenu. Reflektorsko mesto se montira na plafon ili odgovarajuću noseću konstrukciju, a zahteva se potpuna pokrivenost prostora ispod uređaja, odnosno mogućnost usmeravanja svetlosti u bilo koju tačku prostora ispod uređaja.

Neophodno je najpre odabrati komunikacioni protokol i u skladu sa njim odgovarajuću arhitekturu. Nakon toga dimenzionisati pogonske motore u pogledu snage, prenosnog odnosa, i odabrati odgovarajuću upravljačku elektroniku. U obzir je potrebno uzeti i cenu motora. Sledeće što treba osmisliti je kako rešiti povratnu spregu, odnosno kako u svakom trenutku imati informaciju o poziciji platforme, što se svodi na izbor senzora.

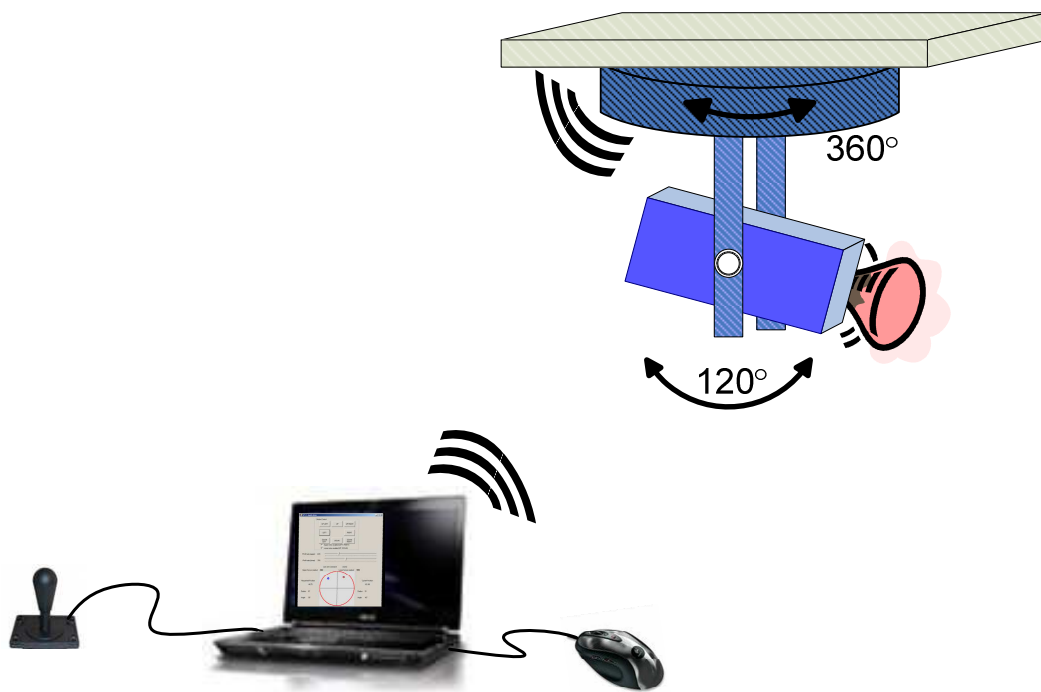
Po završetku projektovanja fizičke strukture sistema, prelazi se na razvoj upravljačkog algoritma, kao i same programske podrške, kako za mikrokontroler, tako i za korisničku

aplikaciju na personalnom računaru. Kao što je zadatkom definisano, upravljački algoritam treba da omogući kontinualno upravljanje u realnom vremenu.

Detaljan opis fizičke strukture sistema, kao i komponenti korišćenih u realizaciji dat je u drugom poglavlju. U trećem poglavlju opisan je upravljački algoritam, kao i detalji realizacije programske podrške za mikrokontroler i personalni računar.

2. Fizička struktura sistema

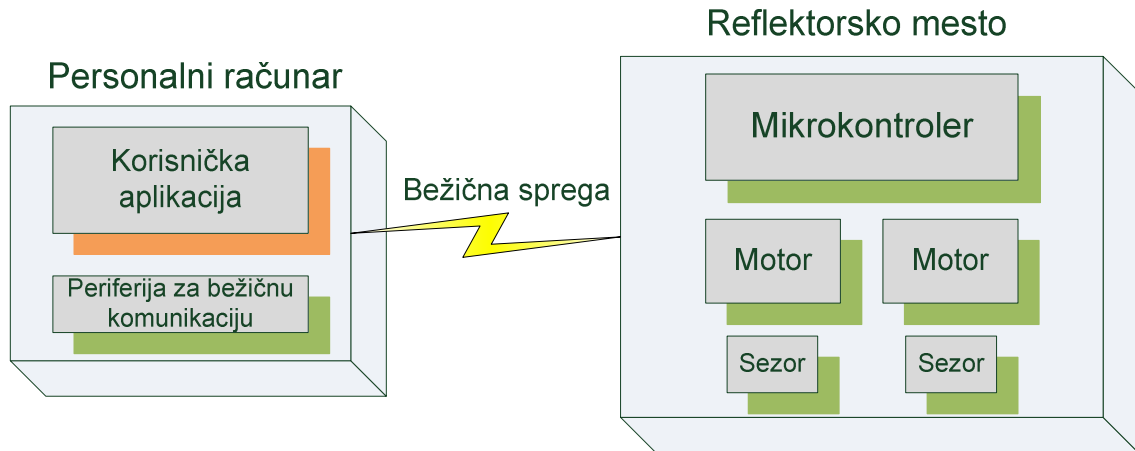
Sistem je koncipiran tako da personalni računar sa odgovarajućim periferijama za bežičnu komunikaciju i intuitivni unos podataka (mišem, upravljačkom palicom), čini centralnu upravljačku jedinicu. Izgled sistema prikazan je na slici 2.1.



Slika 2.1 – Izgled sistema

Na reflektorskom mestu se pored pogonskih motora nalazi pripadajuća upravljačka elektronika, senzori pozicije, kao i mikrokontroler koji je zadužen za pozicioniranje i komunikaciju sa personalnim računarom. Komunikacija treba da bude omogućena bežičnom

spregom, koja će pri tom zadovoljavati uslov zadatka da se omogući upravljanje u realnom vremenu. Struktura sistema prikazana je na slici 2.2.



Slika 2.2 – Blok dijagram sistema

2.1 Izbor protokola za bežičnu komunikaciju

Obzirom na činjenicu da je upravljanje bazirano na bežičnoj komunikaciji, jedan od najvažnijih zadataka prilikom projektovanja sistema je izbor odgovarajućeg komunikacionog protokola. U zavisnosti od izbora protokola dalje se odabira adekvatna arhitektura i razvojno okruženje koji će podržati taj protokol.

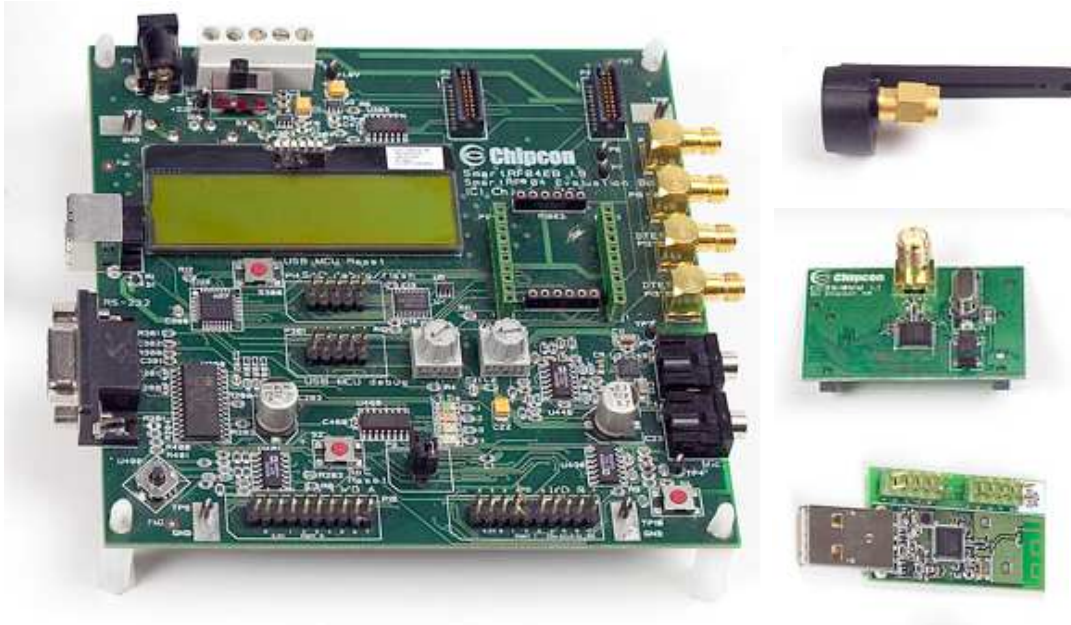
U konkretnom slučaju nema potrebe za velikim protokom podataka na velikim razdaljinama već za što jednostavnijim sistemom sa malom potrošnjom.

Na osnovu postavljenih kriterijuma izbor je sužen na dva protokola, koji su razvijeni za primene automatizacije i upravljanja uređajima za industrijske i potrošačke sisteme i komercijalno su dostupni na tržištu. To su *ZigBee IEEE 802.15.4* i *SimpliciTI*. Oba navedena protokola omogućavaju bežičnu radio komunikaciju na 2.4 GHz, u okviru ISM opsega. Protokol stek *ZigBee* je uniforman za sve proizvođače, ali za njegovo korišćenje treba izdvojiti određenu svotu novca kao naknadu za korišćenje intelektualne svojine, odnosno dobijanja prava na korišćenje. Sa druge strane *SimpliciTI* protokol je razvio *Texas Instruments* i daje besplatno kompletan protokol stek na korišćenje. Pored toga što je besplatan za korišćenje, *SimpliciTI* protokol se može koristiti i u pojednostavljenoj varijanti.

Ključne osobine *SimpliciTI* protokola:

- Mala potrošnja
- Mali protok podataka
- Domet 10-15 metara
- Jednostavan za korišćenje
- Besplatan

Jedan od mikrokontrolera proizvođača *TI* koji podržavaju navedeni protokol je *CC2510f32* sa kojim je kompatibilan razvojni sistem *Smart RF04EB*. Takođe, tu je i USB periferija za personalni računar, na kojoj se nalazi *CC2511* mikrokontroler, koji se od *CC2510* razlikuje jedino po tome što pored ostalog ima podršku za USB komunikaciju. Na slici 2.3 prikazan je razvojni sistem sa USB periferijom.

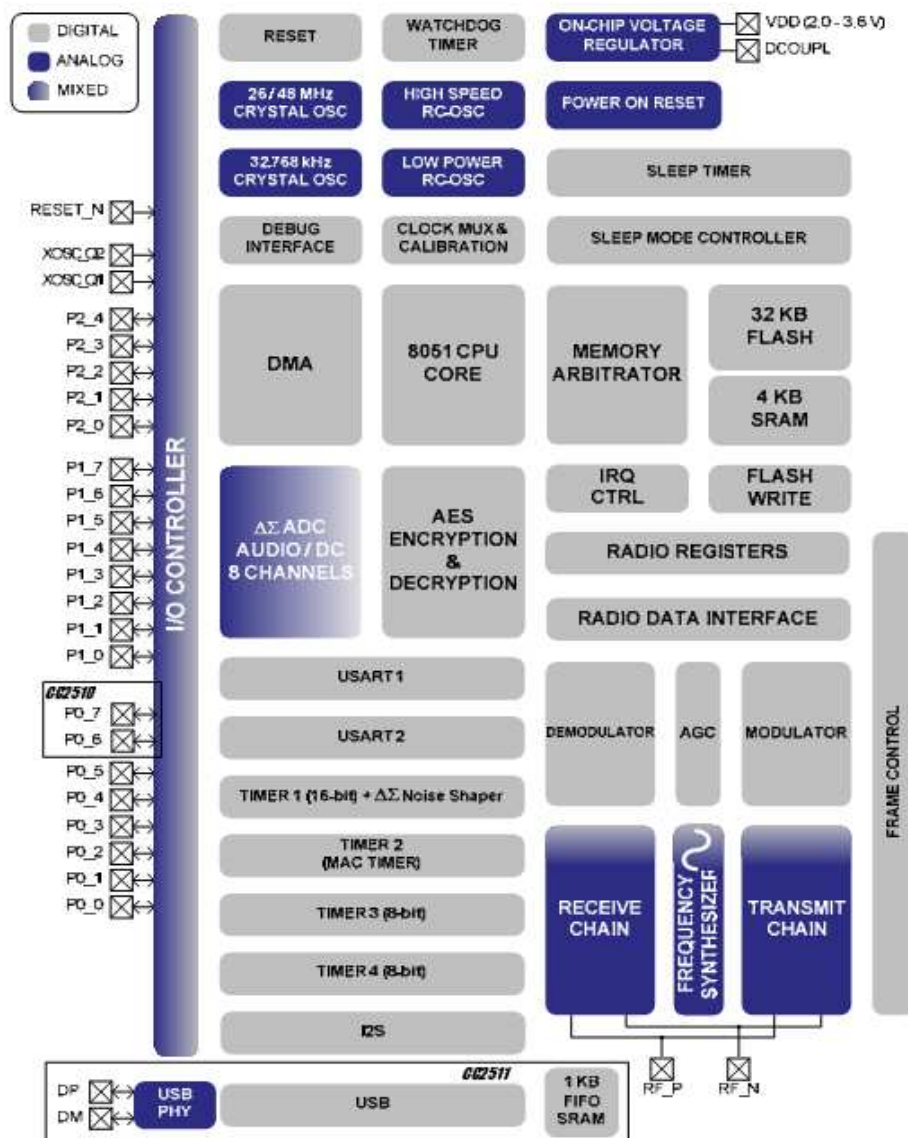


Slika 2.3 – Razvojni sistem *Chipcon SmartRF04EB*, modul sa *CC2510* mikrokontrolerom, USB periferija

2.2 Mikrokontroler

Kao što je navedeno u prethodnom odeljku biće korišćen model *CC2510f32*. Pored osnovnih funkcionalnosti on ima ugrađen primopredajnik za bežičnu komunikaciju, što značajno olakšava projektovanje sistema i smanjuje broj neophodnih komponenti. Posebno je pogodno to što proizvođač uz njega daje kompletno razvojno okruženje što je posebno pogodno imati kod ovakvog razvoja prototipa uređaja. Unutrašnja struktura mikrokontrolera data je na slici 2.4.

Na dnu slike 2.4 vidimo i podršku za USB spregu koja je ugrađena u model *CC2511*, koji se samo po tome i razlikuje od modela *CC2510*. Model *CC2511* nalazi se na primopredajničkoj USB periferiji za personalni računar. Time je omogućena bežična komunikacija sa mikrokontrolerom na platformi. Proizvođač uz navedenu USB primopredajnu periferiju daje i rukovaoce uređajem za *Windows* i *Linux* operativne sisteme.



Slika 2.4 – Unutrašnja struktura mikrokontrolera CC2510/2511

Ovaj mikrokontroler zahteva stabilisano napajanje u opsegu od 2V do 3.6V. Njegova potrošnja varira u zavisnosti od aktivnih periferija. Samo jezgro troši manje od 5mA, dok najveću potrošnju pravi upravo periferija za bežičnu komunikaciju. U najgorem slučaju, tj kada je aktivan radio prijemnik, odnosno predajnik, ukupna potrošnja mikrokontrolera je oko 30mA.

Ključne osobine *CC2510F32* su sledeće:

- Ugrađena periferija za bežičnu komunikaciju na 2.4 GHz
- Poboljšano *8051* jezgro, po idustrijskim standardima
- Mala potrošnja
- 32 kB programabilne memorije (*flash*)
- 4 kB RAM memorije
- Jedno 16-bitno kolo vremenske kontrole i tri 8-bitna
- Dva kanala za serijsku komunikaciju
- IŠM (PWM) generatori

Na razvojnoj ploči realizovan je i modul za bežičnu komunikaciju sa antenom. Od dodatne periferije tu je LCD displej, LED indikacija, tasteri, potenciometri. Dodatne periferije su naročito korisne u fazi razvoja, prilikom traženja i otklanjanja grešaka (*debuging*), za signalizaciju i indikaciju određenih stanja uređaja.

2.3 Pogonski motori

Za pokretanje kućišta reflektora oko dve ose koriste se dva motora. Za precizno pozicioniranje se uobičajeno koriste koračni (*stepper*) motori. Ukoliko želimo koristiti motore jednosmerne struje (*DC*) za ovakvu primenu, tu je neophodno napraviti odgovarajući prenosni odnos, obzirom na činjenicu da ovi motori rade u glavnom na velikom broju obrtaja, reda veličine preko deset hiljada obrtaja u minutu. Kao mogućnost su razmatrani i motori jednosmerne struje bez četkica (*BLDC*). Njihove karakteristike i načini upravljanja su daleko bolji u poređenju sa konvencionalnim motorima jednosmerne struje i koračnim motorima. Međutim, uzimajući u obzir cene motora, kao i odnose mase i snage za navedene tipove motora, opredelili smo se da kao pogonske koristimo klasične motore jednosmerne struje.

Odabrani motor je model *RE385* proizvođača *MFA Como Drills* i korišćen je za pokretanje kućišta po obe ose u sistemu. Nominalni radni napon ovog motora je u opsegu od 6V do 15V. Motori su stigli sa već ugrađenim mehaničkim prenosom u odnosu 11:1. Iz kataloških podataka motora vidimo da je broj obrtaja pri datom prenosnom odnosu i 12V napajanju 1145 obrt/min, što je previše za naš sistem, pa taj prenosni odnos treba dalje razdeliti. Usvojeno je da pri maksimalnoj pobudi motora pun krug rotacije po jednoj osi kućište napravi za 2s. To znači da treba napraviti mehaniku u prenosnom odnosu 30:1. Na taj način definisana je maksimalna brzina kojom se sistem može pozicionirati.

Problematika koja se prva javila bila je kako po prestanku pobude motora zadržati sistem u trenutnom položaju najefikasnije. Po prestanku pobude osovina motora jednosmerne struje može slobodno da se okreće. To se ne sme dozvoliti, pa se to može rešiti na dva načina:

- Upravljanjem (moment držanja)
- Mehanički (izborom prenosa)

U prvoj varijanti, generisanjem momenta držanja na motoru, drastično bi se povećala potrošnja sistema, posebno kada nema kretanja, što bi uticalo i na smanjenje radnog veka motora.

Druga varijanta podrazumeva upotrebu puž prenosa, koji je prikazan na slici 2.5. Spiralni zupčanik se montira na osovinu motora, dok se drugi zupčanik pričvršćuje na osovinu sistema. Pored toga što onemogućava pokretanje osovine motora od strane kućišta, takav mehanički prenos jako je efikasan, kada se u obzir uzme razmera prostora koji zauzima i prenosni odnos koji ima.



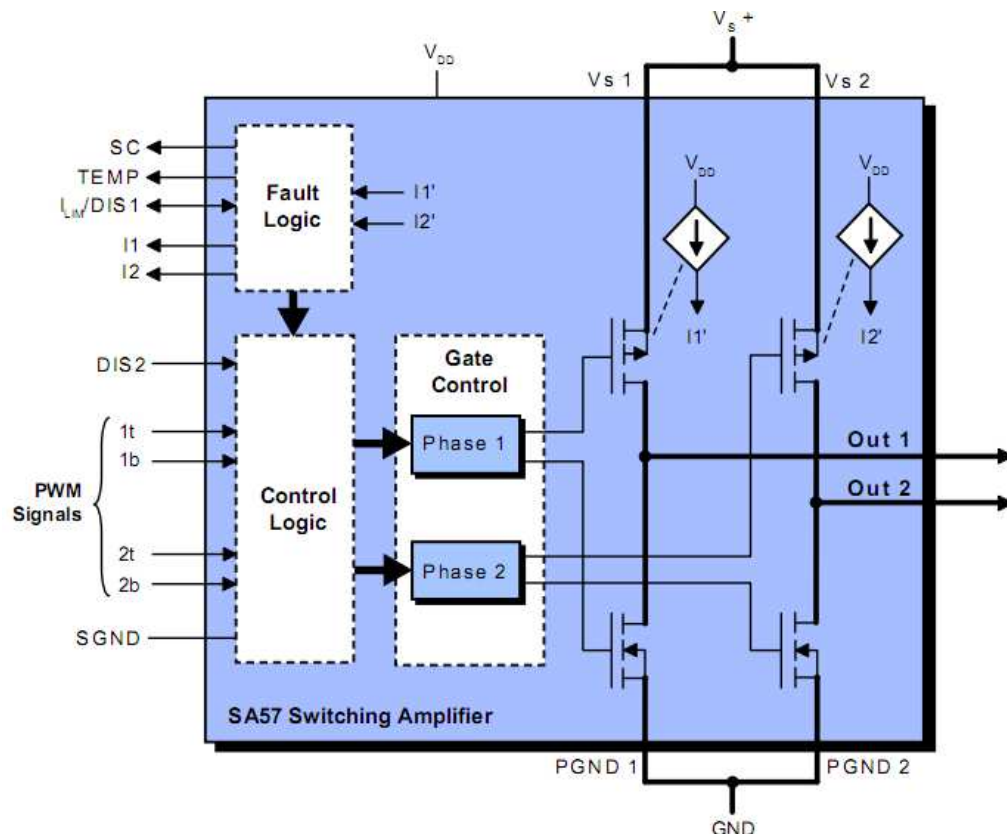
Slika 2.5 – Motor jednosmerne struje i pužni prenos

Izmerena struja gornjeg motora, koji je zadužen za pokretanje oko vertikalne ose, pri 63% maksimalne pobude iznosi 0.2A, što na 12V iznosi 2,4W. Donji motor, koji je pod znatno većim opterećenjem, troši na 70% pobude 0.57A, što je 6.84W. Zbirna potrošnja oba motora ne prelazi 10W, što može biti od značaja za dalja unapređenja. Upravljačka elektronika za motore biće opisana u sledećem odeljku.

2.4 Upravljačka elektronika

Upravljačka elektronika može se realizovati pomoću diskretnih komponenti, ili se mogu koristiti gotove komponente, koje su specijalizovane za takve primene. Poredeći pouzdanost, cenu i jednostavnost realizacije logičan izbor bio je upotreba gotove, integrisane komponente.

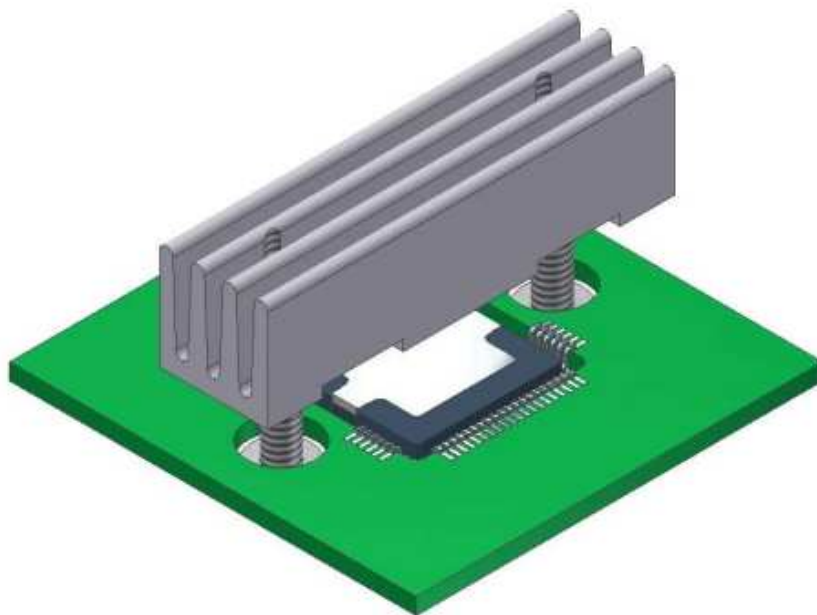
Korišćeno je integrisano kolo *SA-57* proizvođača *Cirrus Logic*. Struktura tog kola data je na slici 2.6.



Slika 2.6 – Struktura integrisanog kola *SA-57*

Glavni deo ovog integrisanog kola čini H-most (desno na slici), koji čine četiri snažna MOSFET-a, koji su sposobni da daju struju jačine 8A u kontinualnom režimu rada, odnosno 15A u impulsnom režimu rada. Tu je i prateća kontrolna logika, koja obezbeđuje da se ne desi da dva tranzistora iz iste grane istovremeno vode, što bi dovelo do spaljivanja istih. Takođe, u ovom integrisanom kolu ugrađena je termička zaštita, a postoji i mogućnost kontinualnog praćenja i ograničavanja struje, za obe grane H-mosta. Ugrađene zaštite su fizičkog tipa, što znači da su nezavisne od mikrokontrolera, samim tim smanjeno je vreme reakcije na neželjene događaje, a istovremeno se smanjuje zauzeće resursa mikrokontrolera.

Takođe je omogućeno direktno sprezanje integrisanog kola sa mikrokontrolerom, bez potrebe za dodatnim komponentama, pulap otpornicima, prilagođenjima i slično. Za ispravno funkcionisanje ovog integrisanog kola treba obezbediti stabilisanih 12V, odnosno +/-12 ili +/-6V, u zavisnosti od toga kakvim napajanjem raspolažemo.



Slika 2.7 – Montaža hladnjaka na integrisano kolo SA57

Proizvođač, *Cirrus Logic*, za isprobavanje i ispitivanje svojstava ovog integrisanog kola ima razvojnu ploču, na koju je montirano integrisano kolo i ostavljene su nožice za sprezanje sa mikrokontrolerom. Montiran je i hladnjak kako bi se poboljšalo odvođenje toplote sa kućišta.

	<i>Mikrokontroler</i>	<i>Razvojna ploča za SA57</i>	
Gornji Motor	P1_6	J5_5	ENABLE
	P1_1	J4_2, J4_5	PWM Input
	P1_0	J4_1, J4_6	PWM Input
	GND	J5_2	GND
Donji Motor	P1_5	J5_5	ENABLE
	P1_7	J4_2, J4_5	PWM Input
	P1_4	J4_1, J4_6	PWM Input
	GND	J5_2	GND

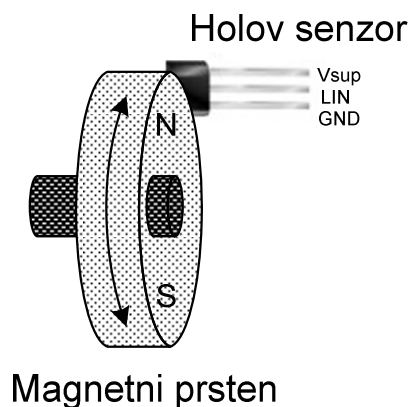
Tabela 2.1 – Povezivanje mikrokontrolera i razvojne ploče sa SA57

Regulacija brzine motora, odnosno redukovane brzine kojom se kućište reflektora pomera, vrši se impulsno-širinskom modulacijom (*PWM*), menjanjem faktora ispune IŠM signala. IŠM signal generisan je unutar mikrokontrolera brojačima visoke rezolucije.

Kao što se može primetiti, ovo integrisano kolo ima daleko veće mogućnosti nego što to sistem zahteva. Upravljačka elektronika je namerno predimenzionisana kako bi se omogućilo što lakše skaliranje sistema u zavisnosti od potrebe. To znači da se u pogledu fizičke strukture ne treba ništa menjati, kao ni programske podrške. U slučaju potrebe za povećanjem dimenzija reflektorskog mesta, promene namene sistema ili povećanja mase uređaja koji se montira umesto reflektora, sve što treba uraditi je pravilno dimenzionisati pogonske motore koji će zadovoljiti potrebe novog okruženja.

2.5 Senzori

Da bi proces upravljanja bio zatvoren mora se kontinualno pratiti pozicija kućišta reflektora, odnosno znati njegov položaj u svakom trenutku. U tom cilju korišćen je senzor koji radi na bazi Holovog efekta, odnosno meri jačinu magnetnog polja koje na njega deluje. Obzirom na to da su kretanja u sistemu kružnog tipa, potrebno je izvršiti konverziju kružnog u pravolinijsko kretanje, jer je senzor za takva merenja i namenjen. To je urađeno upotrebom prstenastog magneta i postavkom senzora kao na slici 2.8.

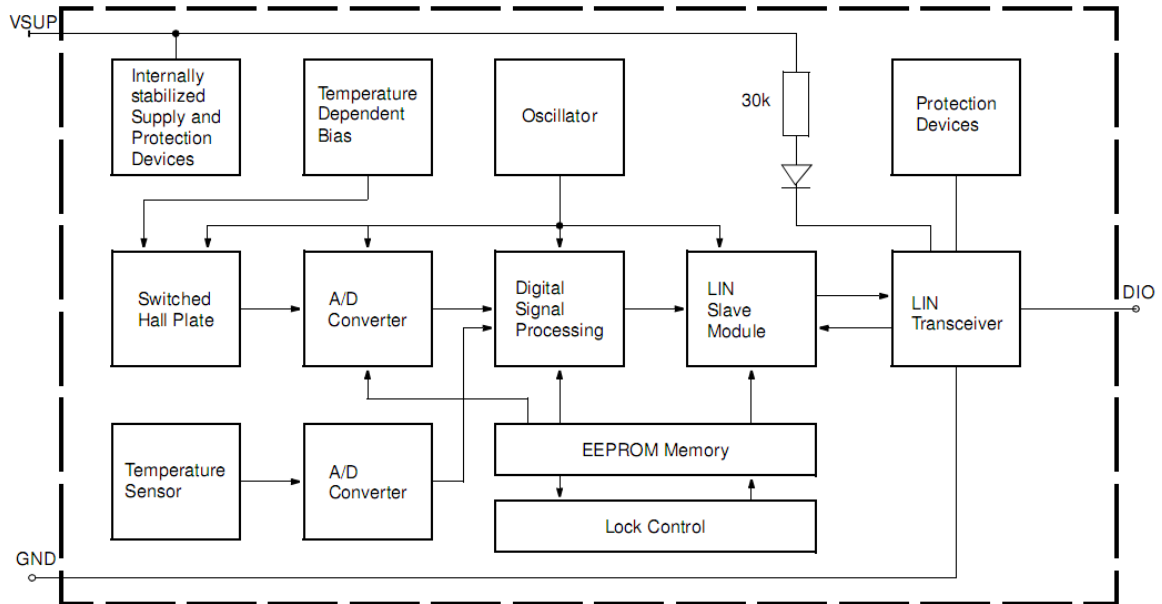


Slika 2.8 – Postavka magnet-senzor

Senzor je pričvršćen za kućište, dok je magnet pričvršćen za osovinu. Pri rotaciji osovine zajedno sa njom rotira se i magnet, čime se menja jačina magnetnog polja kojim magnet deluje na senzor. Očitavanjem jačine magnetnog polja na senzoru dobijamo informaciju o poziciji kućišta za odgovarajući motor i osu rotacije.

2.5.1 Unutrašnja struktura senzora

Upotrebljen je senzor *HAL2810* proizvođača *Micronas*. Pored osnovnog Holovog elementa ovaj senzor sadrži i modul za digitalnu obradu signala, koji je zadužen za temperaturnu kompenzaciju Holovog elementa. Da bi se izvršila temperaturna kompenzacija tu je i temperaturni senzor, A/D pretvarači, unutrašnja stabilizacija, kao i blok zadužen za komunikaciju putem *LIN* sprege.



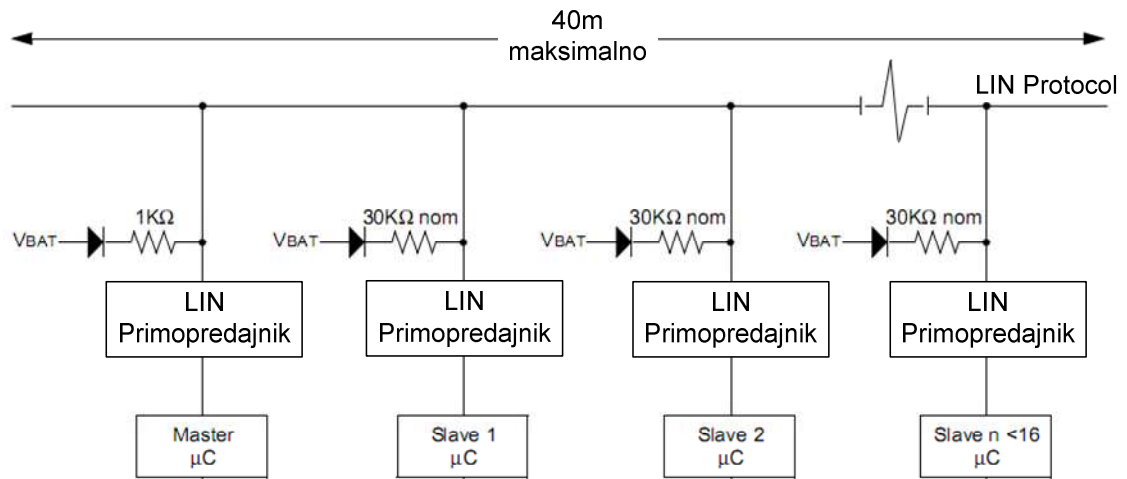
Slika 2.9 – Struktura *HAL2810* senzora

Kao što se sa slike 2.9 vidi, senzor ima 3 nožice: VSUP, GND i DIO. Komunikacija odvija se preko DIO nožice. Radni opseg napajanja ovog senzora je 9V do 40V. Senzor je namenjen za korišćenje u automobilskoj industriji, pa je zbog toga veoma robustan, otporan na šumove, prenapone, kao i temperaturne varijacije. Tipičan napon napajanja ovih senzora je 12V, a takođe i nominalan radni napon LIN magistrale. To znači da naponski nivo mikrokontrolera od 3.3V treba prilagoditi naponskom nivou LIN magistrale.

LIN protokol predstavlja nadogradnju na RS-232 (serijsku) komunikaciju, tako da je dovoljno napraviti fizičku spregu i pravilno podesiti generator takta za serijsku komunikaciju unutar mikrokontrolera i tako se mogu iskoristiti postojeći resursi za realizaciju. Magistrala za *LIN* može da podrži do šesnaest senzora koji su povezani u paraleli. Za ovu konkretnu primenu dovoljna su nam dva senzora. Za svaku osu rotacije po jedan. Tipičan izgled magistrale, kao i fizička sprega sa mikrokontrolerom biće prikazani u sledećem odeljku.

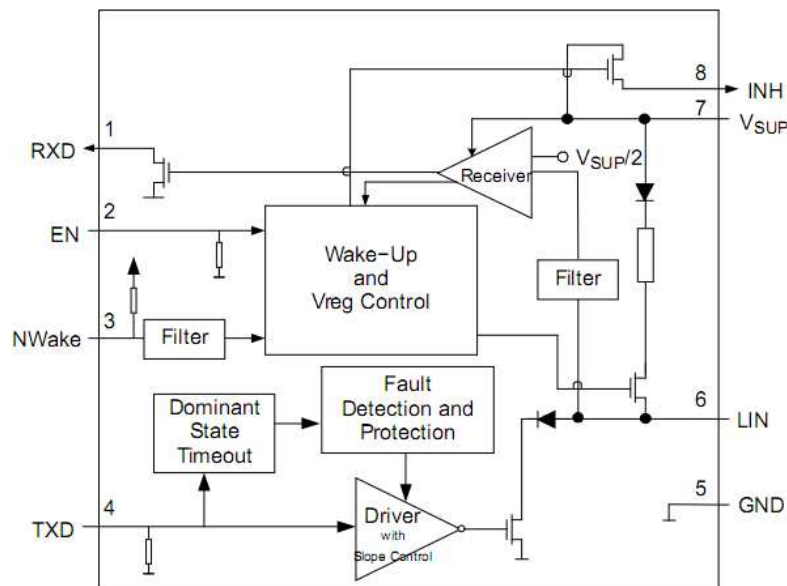
2.5.2 Fizička sprega mikrokontrolera sa LIN magistralom

Na slici 2.10 prikazana je LIN magistrala i kako se na nju u paraleli povezuju senzori.



Slika 2.10 – LIN magistrala

Kao primopredajnik RS-232 na LIN magistralu korišćeno je integrisano kolo *TPIC1021* proizvođača *Texas Instruments*. Funkcionalni blok dijagram ovog integrisanog kola je prikazan na slici 2.11. Slika je preuzeta iz kataloških podataka proizvođača. Na stranu mikrokontrolera povezuju se RXD i TXD nožice i to na odgovarajuće nožice mikrokontrolera za RX i TX ta serijsku komunikaciju, dok EN treba da bude na logički visokom nivou da bi integrisano kolo funkcionisalo. Na LIN magistralu povezuju se nožice Vsup, LIN i GND.



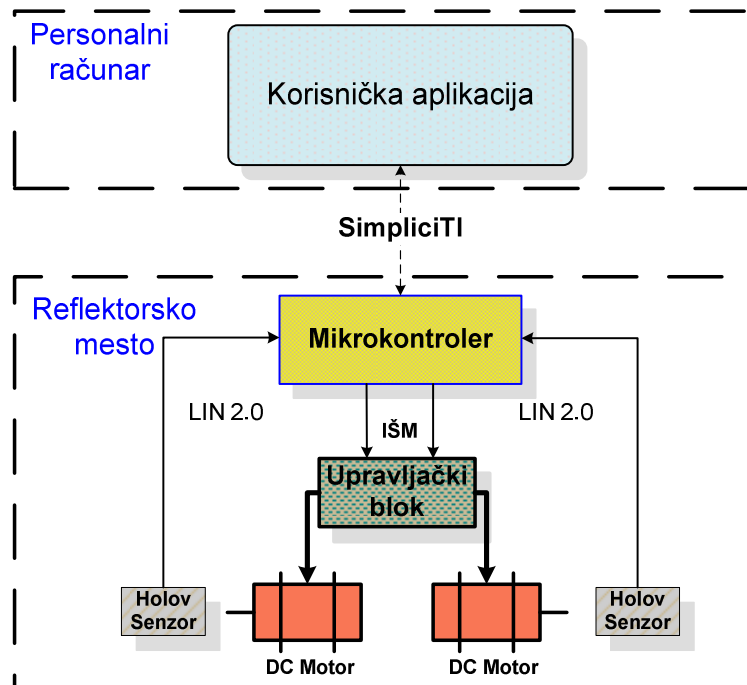
Slika 2.11 – Unutrašnja struktura integrisanog kola *TPIC1021*

Gledajući od mikrokontrolera ka LIN magistrali, ovo kolo vrši podizanje i prilagođenje naponskih nivoa. Istovremeno se u obrnutom smeru vrši spuštanje naponskih nivoa i prenošenje svega što se dešava na LIN magistrali na prijemnu nožicu mikrokontrolera, u opsegu od 0 do 3.3V. Zbog toga treba u realizaciji programske podrške za mikrokontroler obratiti pažnju da se onemogući prijem dok traje emitovanje podataka sa mikrokontrolera, kako ne bi dolazilo do konflikta.

3. Programska podrška

Sledeće pitanje koje se postavlja je šta se u stvari podrazumeva pod upravljanjem u realnom vremenu, odnosno o kom se redu veličine vremenskih intervala zapravo radi. Podaci iz jedne studije Vašingtonskog univerziteta govore da je vreme reakcije uvežbanog rukovaoca uređajima poput kamera ili scenske rasvete oko 240 ms. To znači da ukupan odziv sistema mora biti manji od ovog intervala, idealno bi bilo da je trenutani, što je naravno nemoguće postići.

Sistem ima strukturu sa slike 3.1. Centralna upravljačka jedinica je personalni računar zajedno sa korisničkom aplikacijom instaliranom na njemu.



Slika 3.1 – Struktura sistema

Željena pozicija reflektorskog mesta se zadaje pomoću korisničke aplikacije koja se izvršava na personalnom računaru. Zadana pozicija se bežično prenosi ka adresiranom

reflektorskom mestu. Izvršna aplikacija na mikrokontroleru unutar reflektorskog mesta kojem je poslati paket namenjen prihvata poruke. Po utvrđivanju validnosti poruke poredi se trenutna pozicija sa zadatom. Nakon toga, ukoliko je potrebno, mikrokontroler izdaje komande motorima za kretanje u odgovarajućem smeru u vidu IŠM signala koje šalje na upravljačku elektroniku za motore motore jednosmerne struje. U međuvremenu Holovi senzori naizmenično očitavaju trenutnu poziciju i šalju je mikrokontroleru putem LIN 2.0 protokola. Mikrokontroler na osnovu pozicije odlučuje da li se nastavlja ili obustavlja kretanje.

Detalniji opis programske podrške podeljen je u dve celine: programska podrška za mikrokontroler i za personalni računar – korisnička aplikacija.

3.1 Programska podrška za mikrokontroler

Razvoj programske podrške za mikrokontroler urađen je korišćenjem programskog paketa *IAR Embedded Workbench* i razvojnog sistema *Chipcon SmartRF04EB*. Programski jezik je jezik *C* za mikrokontrolere (*Embedded C*).

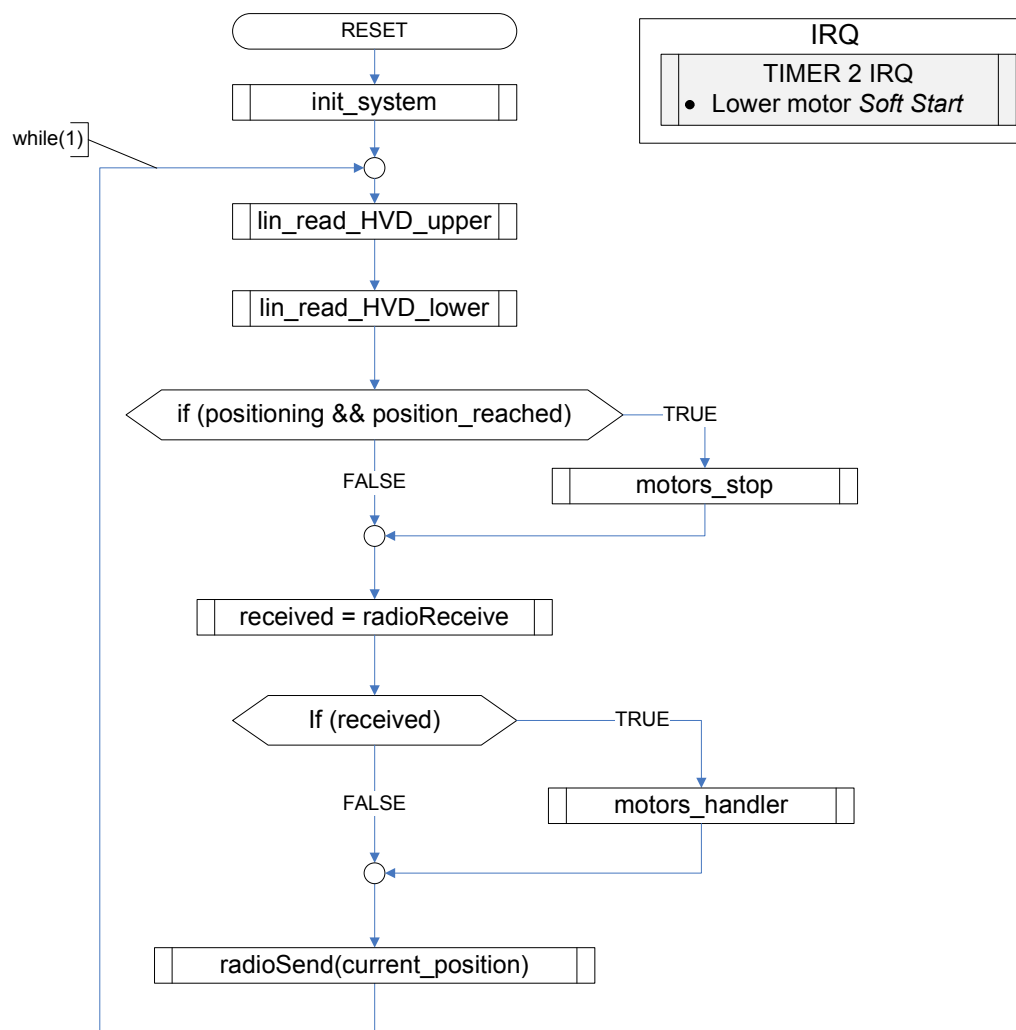
U okviru programskog paketa *IAR Embedded Workbench* integrisan je skup alata za generisanje izvrsne datoteke:

- Prevodilac: *8051 IAR C/C++ Compiler*
- Povezivač: *IAR XLINK Linker*
- Asembler: *8051 IAR Assembler*

Programska podrška za mikrokontroler treba da omogućiti:

- Bežičnu komunikaciju sa personalnim računarom
- Očitavanje vrednosti sa senzora
- Upravljanje motorima jednosmerne struje
- Slanje informacija o trenutnoj poziciji na personalni računar

Na slici 3.2 je prikazan algoritam izvrsavanja programske podrške za mikrokontroler. Po pokretanju sistema vrši se inicijalizacija bežične komunikacije, ulazno izlaznih prolaza, kola vremenske kontrole, kao i otpočinjanje LIN komunikacije sa sensorima.

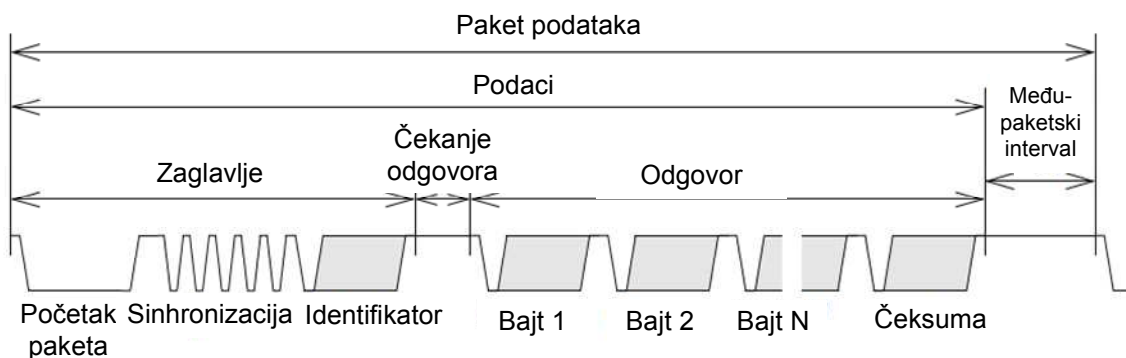


Slika 3.2 – Algoritam programske podrške za mikrokontroler

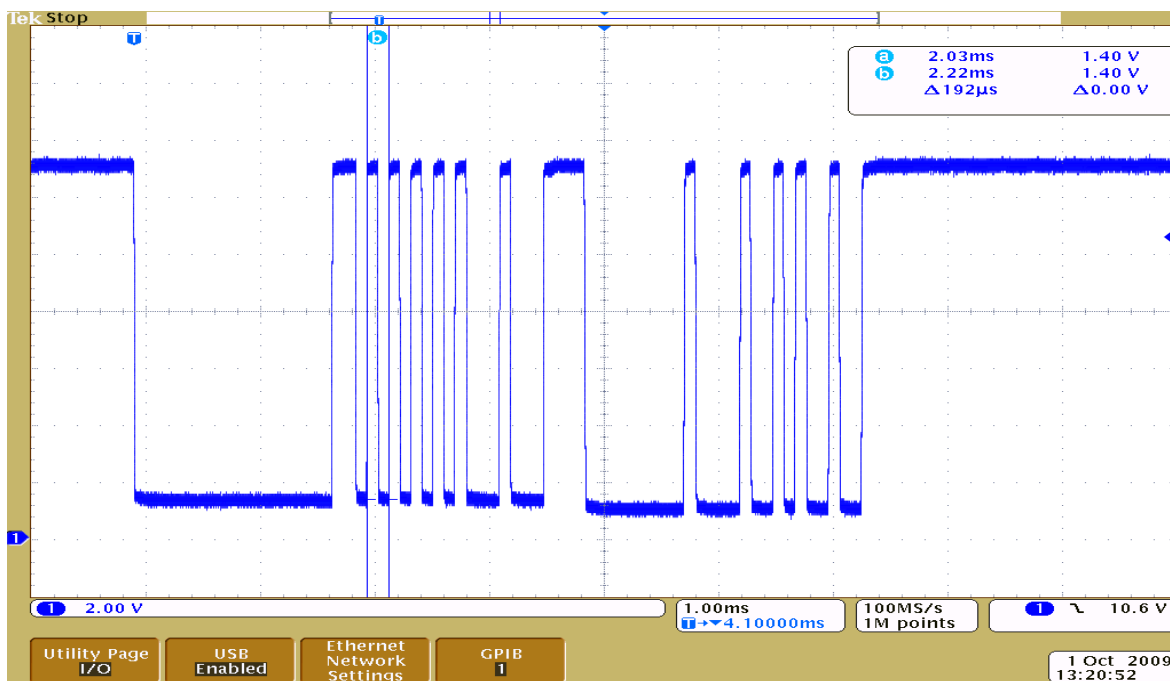
3.1.1 Realizacija LIN 2.0 protokola

Evropski konzorcijum proizvođača automobila razvio je ovaj serijski komunikacioni protokol i predstavio ga kao jeftino rešenje za kratka rastojanja i male brzine. Za takve primene je već razvijeni, pouzdani i brzi CAN protokol bio skup i predimensionisan. Konfiguracija magistrale je tipa jedan vodeći (*master*) kontroler, centralna jedinica, i jedan ili više pratećih, do 16, (*slave*) jedinica-čvorova. Vodeći kontroler proziva čvorišta da šalju svoje podatke na magistralu, koji su pri tom vidljivi svim ostalim uređajima na njoj. Na taj način se obavlja dvosmerna komunikacija, i nije potrebna dalja arbitraža. Pod kratkim rastojanjima podrazumeva se da magistrala nije duža od 40m. Maksimalna brzina protoka koja se može postići je 20 kbit/s. Minimalno vreme odziva je reda veličine 50ms, što je dovoljno za spore procese. LIN protokol podržava i slanje ureñaja koji se ne koriste u stanje mirovanja (*sleep mode*) u cilju smanjenja potrošnje, tačnije, ako se senzor ne očitava 4s, automatski odlazi u stanje mirovanja. Takođe, neophodno je i njihovo vraćanje u operativno stanje (*wake up*).

Paket podataka ima izgled prikazan na slici 3.3. Zaglavlje sadrži naznaku početka paketa, sekvencu za sinhronizaciju i identifikator. Prva dva dela su identična za svaki paket LIN protokola. Identifikator je različit u zavisnosti od senzora na magistrali kome je namenjen i paketa koji mu se šalje, a uključena je i provera parnosti. Potom se čeka na odgovor. Na ispravno zaglavlje koje mu je namenjeno senzor odgovara slanjem podataka bajt po bajt. Pri tom se u zavisnosti od zaglavlja zna tačno koliko bajtova se očekuje od senzora. Na kraju stiže provera ispravnosti pristiglih podataka. U okvir paketa podataka ulazi i međupaketski interval.



Slika 3.3 – Paket podataka



Slika 3.4 – Paket podataka snimljen osciloskopom

Da su specifikacije zadovoljene, odnosno da je protokol ispravno realizovan pokazuje se snimkom sa osciloskopa, na slici 3.4. U gornjem desnom uglu snimka vidi se trajanje dva bitska intervala koje iznosi 192us. To odgovara prenosnoj brzini od 10.4kbit/s.

Zaglavlja i kratki opisi funkcija koje služe za realizaciju komunikacije sa senzorom, kao i definicije konstanti se nalaze unutar *lin.h* datoteke. Realizacija je smeštena u *lin.c* datoteku. U nastavku će biti data zaglavlja funkcija sa kratkim opisima, ulaznim i izlaznim parametrima.

Funkcija	Opis
<code>lin_init</code>	Vrši inicijalizaciju komunikacije sa senzorima na celoj LIN magistrali
Parametar	Opis
<code>void</code>	-
Povratna vrednost	Opis
<code>void</code>	-
Poziva funkcije	<code>lin_wake</code>

Funkcija	Opis
<code>lin_wake</code>	Obezbeđuje nizak naponski nivo na magistrali u trajanju od 1ms, koliko je potrebno da bi se senzori doveli u operativno stanje. Nakon ove komande potrebno je sačekati bar 100ms kako bi se izvršile inicijalizacije unutar senzora, pre nego što se pošalje bilo kakva naredna komanda.
Parametar	Opis
<code>void</code>	-
Povratna vrednost	Opis
<code>void</code>	-
Poziva funkcije	-

Funkcija	Opis
<code>lin_set_addr</code>	Postavlja adresu registra određenog senzora za očitavanje, tako što šalje kompletan paket podataka na LIN magistralu. Nephodno je postaviti adresu registra pre očitavanja
Parametar	Opis
BYTE <code>addr</code>	Adresa registra unutar senzora čija se vrednost očitava
BYTE <code>set_addr_pid</code>	Zaštićeni identifikator paketa (<i>Protected Identifier</i>) u kom se nalazi informacija o kom se senzoru radi
Povratna vrednost	Opis
void	-
Poziva funkcije	<code>lin_break</code> , <code>lin_synch</code> , <code>lin_send_byte</code>

Funkcija	Opis
<code>lin_set_HVD_addr_upper_sensor</code>	Postavlja adresu za očitavanje HVD registra gornjeg senzora, ista funkcija postoji i za donji senzor
Parametar	Opis
void	-
Povratna vrednost	Opis
void	-
Poziva funkcije	<code>lin_set_addr</code>

Funkcija	Opis
<code>lin_read_HVD</code>	Očitava dvobajtnu vrednost iz HVD registra senzora, ukoliko je prethodno izvršeno ispravno postavljanje adrese
Parametar	Opis
BYTE <code>read2b_pid</code>	Zaštićeni identifikator paketa, odnosno zaglavlja za očitavanje dvobajtnu vrednosti iz prethodno adresiranog registra odgovarajućeg senzora
Povratna vrednost	Opis
INT16	Dvobajtna vrednost iz HVD registra
Poziva funkcije	<code>lin_read2_header</code>

Funkcija	Opis
<code>lin_read2_header</code>	Na LIN magistralu šalje zaglavlje paketa za očitavanje dvobajtna vrednosti iz prethodno adresiranog registra senzora
Parametar	Opis
<code>BYTE read2b_pid</code>	Zaštićeni identifikator paketa, odnosno zaglavlja za očitavanje dvobajtna vrednosti iz prethodno adresiranog registra odgovarajućeg senzora
Povratna vrednost	Opis
<code>void</code>	-
Poziva funkcije	<code>lin_break, lin_synch, lin_send_byte</code>

Funkcija	Opis
<code>lin_read_HVD_lower</code>	Vrši očitavanje vrednosti HVD registra senzora montiranog uz donji motor, postoji ekvivalentna funkcija za gornji
Parametar	Opis
<code>void</code>	-
Povratna vrednost	Opis
<code>INT16</code>	Dvobajtna vrednost iz HVD registra
Poziva funkcije	<code>lin_read_HVD</code>

Funkcija	Opis
<code>lin_read_HVD_upper_float_value</code>	Vrši očitavanje vrednosti sa senzora i izračunava vrednost u [mT] koja je reprezentovana pokretnim zarezom
Parametar	Opis
<code>void</code>	-
Povratna vrednost	Opis
<code>float</code>	Jačina magnetnog polja u [mT]
Poziva funkcije	<code>lin_read_HVD</code>

Funkcija	Opis
lin_go2sleep	Funkcija šalje sve senzore na magistrali (<i>cluster-u</i>) u stanje mirovanja (<i>sleep mode</i>)
Parametar	Opis
BYTE master_rq_pid	Zaštićeni identifikator paketa podataka, koji ima vrednost 0x3C, što je definisano u specifikacijama za LIN protokol
Povratna vrednost	Opis
Void	-
Poziva funkcije	-

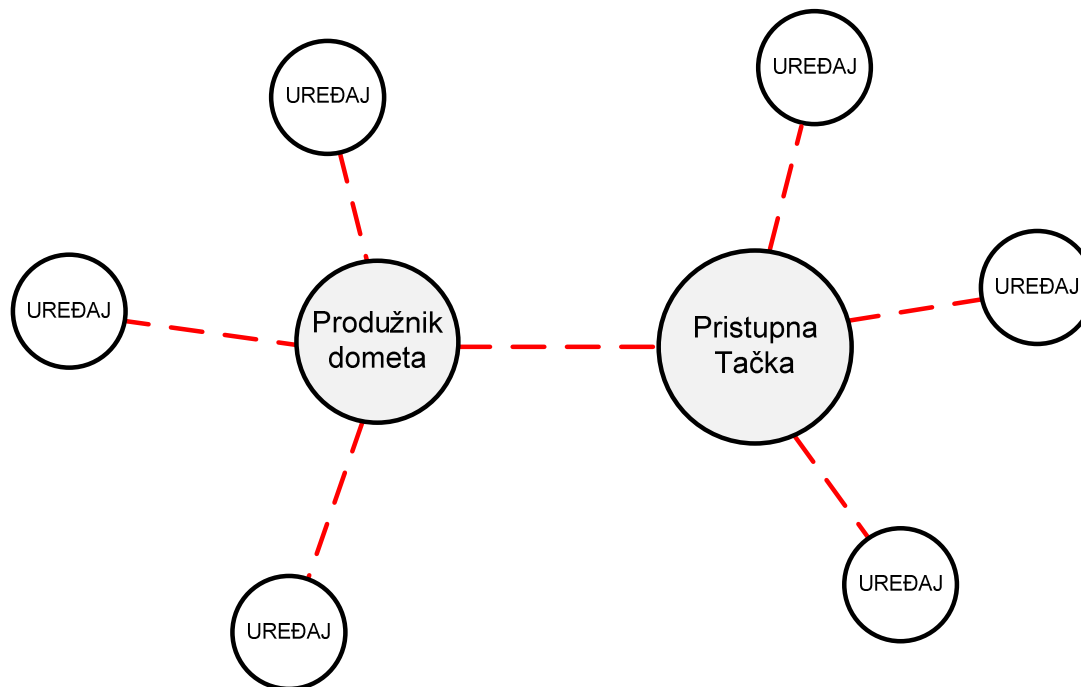
Funkcija	Opis
checksum_enh	Funkcija šalje sve senzore na magistrali (<i>cluster-u</i>) u stanje mirovanja (<i>sleep mode</i>)
Parametar	Opis
UINT8 pid	Zaštićeni identifikator paketa podataka
UINT8 *data	Pokazivač na niz podataka koji će biti poslat
UINT8 num_of_bytes	Broj bajtova za koji treba uraditi računanje, ne računajući zaštićeni identifikator paketa podataka
Povratna vrednost	Opis
UINT8	
Poziva funkcije	-

3.1.2 Bežični protokol SimpliciTI

Pod *SimpliciTI* bežičnim protokolom podrazumeva se radio komunikacija na 2.4 GHz. Proizvođač, *Texas Instruments*, uz mikrokontroler, razvojni sistem i USB periferiju za personalni računar daje i gotov programski modul, sa već realizovanim protokolom. Time se dobija zaokružena celina i u mnogome olakšava razvoj prototipova. Bitno je napomenuti da za korišćenje ovog protokola *Texas Instruments* ne naplaćuje pravo korišćenja intelektualne svojine, i na taj način ne pravi nikakve dodatne troškove firmama koje se bave razvojem, dok istovremeno podstiče na korišćenje svojih rešenja. Ovaj protokol je, dakle, namenjen razvoju bežičnih uređaja sa podrškom za jednostavno umrežavanje, uz malu potrošnju energije, što je

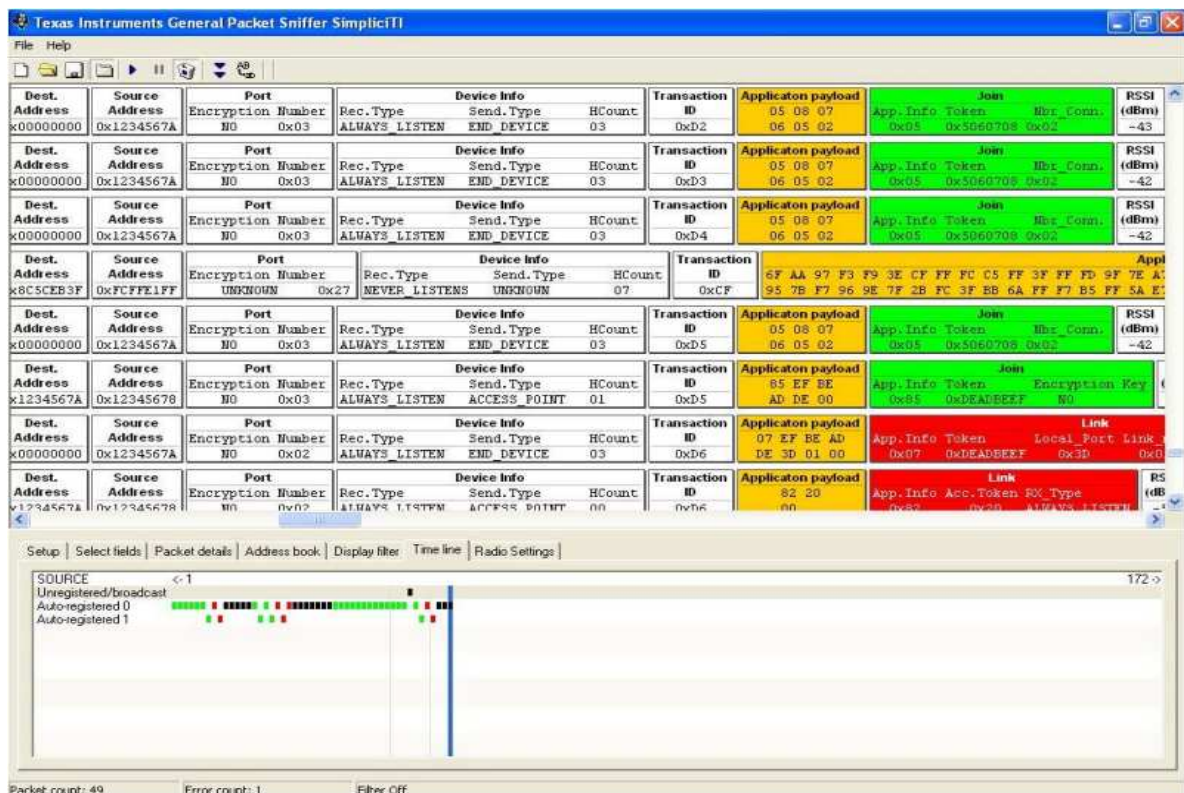
posebno pogodno za baterijski napajane uređaje. Opšta topologija *SimpliciTI* mreže je tipa zvezda i prikazana je na slici 3.5.

Bez korišćenja produžnika dometa moguće je ostvariti komunikaciju na rastojanju 10-15 metara u zavisnosti od tipa antene, pod uslovom da postoji fizička vidljivost između prijemnika i predajnika. Za potrebe razvoja ovakvog jednog sistema produžnik dometa nije morao biti korišćen, a pošto je samo jedan uređaj u pitanju, bilo je dovoljno koristiti povezivanje tipa tačka-tačka (*peer-to-peer*).



Slika 3.5 – Opšta topologija *SimpliciTI* bežične reže

Ispravnost funkcionisanja *SimpliciTI* radio komunikacije verifikuje se programskim alatom *Packet Sniffer* koji je prikazan na slici 3.6.



Slika 3.6 – Packet Sniffer

Prototipovi funkcija koje daje proizvođač nalaze se u biblioteci *cul.h*, dok je realizacija najbitnijih funkcija unutar *radio.c* datoteke. U nastavku će biti navedeni prototipovi funkcija sa pojašnjenjima.

Funkcija	Opis
<code>radioInit</code>	Inicijalizacija registara i periferije mikrokontrolera za bežičnu komunikaciju
Parametar	Opis
UINT32 frequency	Radio frekvencija koja će biti korišćena
BYTE localAddress	Adresa mrežnog čvora, odnosno reflektorskog mesta
Povratna vrednost	Opis
Void	-

Funkcija	Opis
radioSend	Funkcija šalje podatke date dužine pojedinačnom primaocu ili emituje svim slušaocima korišćenjem radio sprege
Parametar	Opis
BYTE* transmitData	Pokazivač na niz podataka koji treba da budu poslani
WORD dataLength	Broj bajtova koje treba poslati (emitovati)
BYTE doAck	Da li želimo da dobijemo potvrdu o prijemu poslanog paketa ili ne
Povratna vrednost	Opis
BOOL	Vrednost koju vraća funkcija je tipa BOOL i to TRUE ako je potvrđen prijem paketa od strane prijemnika, a FALSE ukoliko nije bilo potvrde

Funkcija	Opis
radioReceive	Funkcija šalje podatke date dužine pojedinačnom primaocu ili emituje svim slušaocima korišćenjem radio sprege
Parametar	Opis
BYTE** receiveData	Dvostruki pokazivač na primljeni paket podataka. Na taj način smanjuje se zauzeće RAM memorije unutar mikrokontrolera
BYTE* length	Pokazivač na podatak o dužini pristiglog paketa podataka
WORD timeout	Vrednost koja govori koliko će najduže prijemnik čekati na paket u milisekundama
BYTE* sender	Na ovu adresu funkcija će upisati adresu pošiljaoca paketa
Povratna vrednost	Opis
BOOL	Funkcija vraća vrednost tipa BOOL, i to TRUE ako je paket primljen, a FALSE ako nije primljen paket, odnosno istekao je timeout [ms] interval čekanja

3.1.3 Upravljanje motorima

Motorima se upravlja slanjem IŠM signala i signala dozvole (*ENABLE*) na upravljačku, snažnu elektroniku koja pogoni motore. U zavisnosti od željenog smera kretanja motora za emitovanje IŠM signala se aktivira odgovarajući kanal, odnosno nožica mikrokontrolera. IŠM signali se generišu korišćenjem kola vremenske kontrole 1 i 3 mikrokontrolera. Arhitektura mikrokontrolera je takva da omogućava generisanje IŠM signala korišćenjem kola vremenske kontrole, bez potrebe za korišćenjem prekidnih rutina. Dovoljno je podesiti odgovarajuće registre tako da se dobije odabrani period signala, kao i faktor ispune. Poželjno je da frekvencija IŠM signala bude iznad 20 kHz da ne bi bilo neprijatnih zvukova.

Datoteka u kojoj su definisane sve konstante i nožice koje se koriste pri upravljanju motorima je *motor.h*. Funkcija za rukovanje motorima realizovana je u okviru *main.c* datoteke.

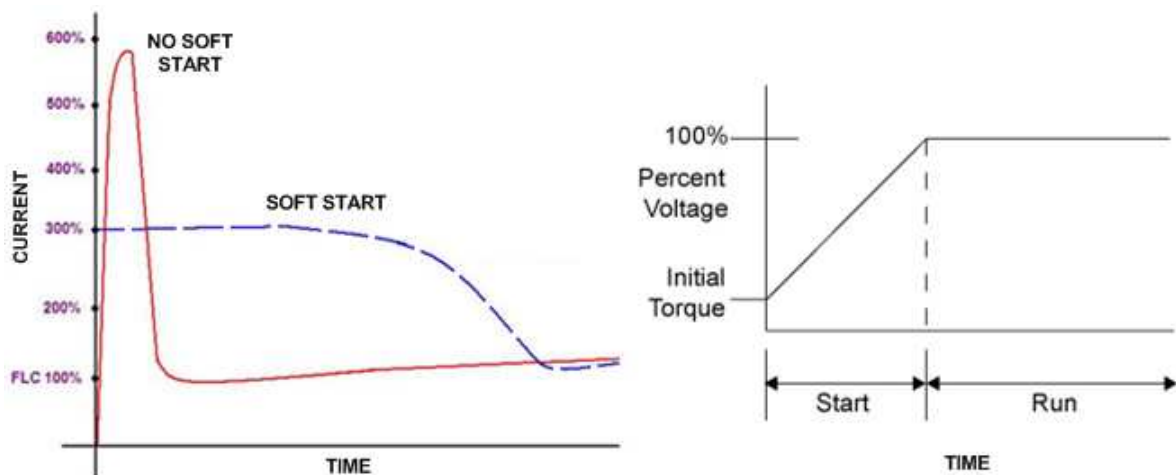
	Dozvola rada motora		Smer motora		Vrednost parametra
	gornji	donji	gornji	donji	
STOP_ALL	0	0	0	0	0x00
DIR_LEFT	1	0	0	0	0x08
DIR_RIGHT	1	0	1	0	0x0A
DIR_UP	0	1	0	0	0x04
DIR_DOWN	0	1	0	1	0x05
DIR_UP_RIGHT	1	1	1	0	0x0C
DIR_UP_LEFT	1	1	0	0	0x0D
DIR_DOWN_LEFT	1	1	0	1	0x0E
DIR_DOWN_RIGHT	1	1	1	1	0x0F

Tabela 3.1 – Formiranje `directions_table[9]`

Za svaki od četiri donja bita podataka iz tabele 3.1 postoji odgovarajuća maska i na taj način se vrlo lako pokreću ili zaustavljaju motori. Konstante su definisane unutar *motor.c* datoteke.

Funkcija	Opis
motors_handler	Rukuje motorima, u zavisnosti od vrednosti prosleđenog parametra rade oba ili samo jedan od motora, u smeru takođe zavisnom od parametra
Parametar	Opis
BYTE state	Sadrži informaciju o tome koji motor treba da se kreće i u kom smeru, parametar dobija jednu od vrednosti iz tabele 3.1
Povratna vrednost	Opis
void	-

Donji motor ima znatno veće opterećenje nego gornji, zbog mehaničkog prenosa i pokretanja kućišta po horizontalnoj osi. Njemu je zbog toga potrebna veća struja za rad, odnosno veći faktor ispunje IŠM signala. Ugao zakrenutosti u odnosu na horizontalnu osu ima značajan uticaj na opterećenje motora. Što je veći ugao, veće je i opterećenje. Kako izgledaju struje motora u slučaju kada imamo usporeni start i kada ga nema može se videti na slici 3.6. Na toj slici vidimo i kako izgleda osnovni model usporenog starta, po linearnoj funkciji. Složeniji modeli koriste aproksimaciju sinusnih, kvadratnih ili drugih složenijih funkcija kako bi se dobio što mekši i laganiji start motora. Bez korišćenja usporenog starta pored toga što se može degradirati radni vek motora, imamo trzanje kućišta. To je nepoželjno kako sa stanovišta mehaničkih delova sistema, tako i same krajnje funkcionalnosti uređaja.



Slika 3.7 – Struje motora sa i bez laganog starta (levo) i lagani start (desno)

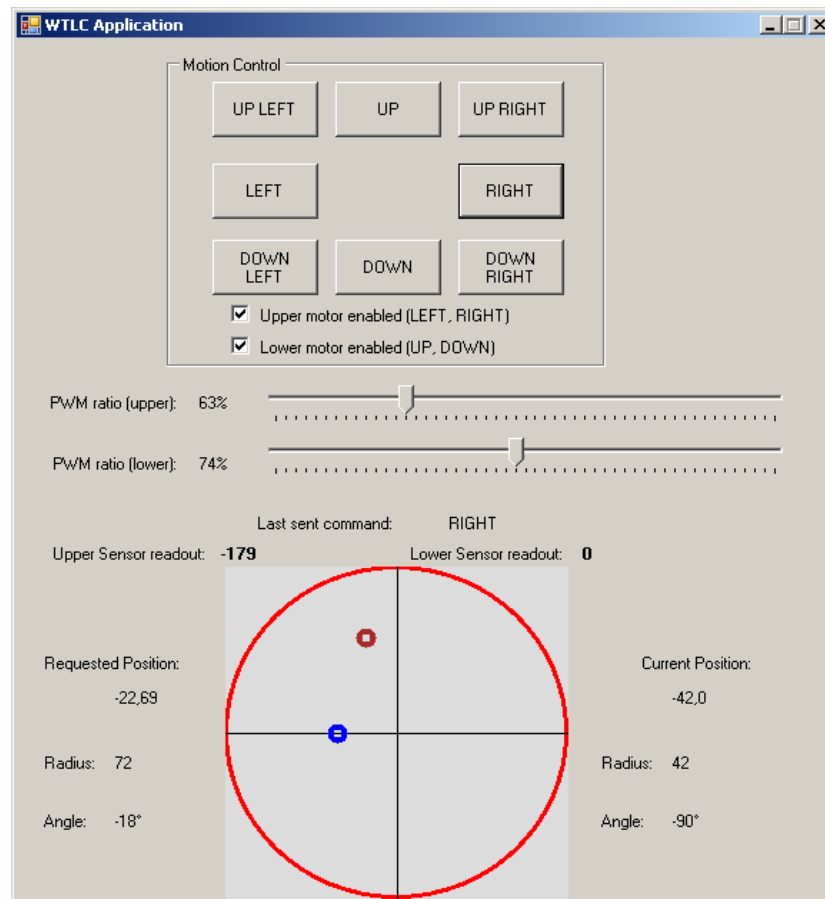
Lagani start motora može biti realizovan na dva načina:

- Programski
- Fizički, korišćenjem elektronskih komponenti.

U realizaciji ovog sistema to je urađeno programski, unutar programske podrške za mikrokontroler. Korišćenjem prekidne rutine kola vremenske kontrole 2 mikrokontrolera. Ukoliko je omogućeno usporeno startovanje, u toku svake prekidne rutine povećava se faktor ispunje za donji motor. Nagib funkcije za usporeni start definiše se vremenskim intervalom prekida kola vremenske kontrole 2, kao i time za koliko se uvećava faktor ispunje.

3.2 Programska podrška za personalni računar

Razvoj korisničke aplikacije za personalni računar urađen je korišćenjem *Microsoft Visual Studio 2005* programskog paketa, i programskog jezika *C#*. Periferija za bežičnu komunikaciju se ponaša kao kolo za serijsku komunikaciju koje se nalazi na prolazu *COM3*, pa je dovoljno u aplikaciju ubaciti serijski prolaz (*Serial Port*) i podesiti ga po specifikacijama za *SimpliciTI* protokol.



Slika 3.8 – Korisnička aplikacija

Korisnička aplikacija sastoji se iz jednog oblika (*Form*) kao na slici 3.8. U gornjem delu forme nalazi se 8 dugmića (*buttons-a*) kojima se može ručno odabrati bilo koja kombinacija rada motora, odnosno pokretanja kućišta. Na sredini oblika nalaze se dva klizača kojima se menja faktor ispunje za svaki motor posebno. U donjem delu forme nalazi se slika. Na slici veliki krug predstavlja prostor, gledano od gore, iz „ptičije“ perspektive.

Plavi kružić predstavlja trenutnu poziciju sistema, dok braon kružić predstavlja zadatu poziciju, i on se pojavljuje tek nakon duplog klika na unutrašnjost velikog crvenog kruga, a nestaje po dostizanju zadate pozicije. Sve dešavanja u korisničkoj aplikaciji se pokreću na događaje (*events*). To su događaji „klik mišem na dugme“, „pomeranje klizača“, „dvoklik na sliku“.

U nastavku će biti navedene funkcije koje pokreću odgovarajući događaji, sa kratkim opisom funkcionalnosti.

Funkcija	Opis
<code>buttonUP_MouseDown</code>	Nakon događaja „dugme miša pritisnuto“ šalje na USB periferiju <i>ASCII</i> kod (karakter) indeksa kojim se određuje u kom smeru će se sistem kretati. Periferija emituje radio poruku u odgovarajućem formatu i po prijemu poruke mikrokontroler na osnovu primljene poruke iz tabele 3.1 uzima vrednost parametra koji se prosleđuje funkciji <code>motors_handler</code> programske podrške mikrokontrolera
Parametar	Opis
<code>object sender</code>	Informacija o tome ko je izazvao događaj
<code>MouseEventArgs e</code>	Informacija o kom se događaju radi
Povratna vrednost	Opis
<code>void</code>	-
Poziva metode	<code>usbDongle.Write</code>

Funkcija	Opis
buttonUP_MouseUp	Funkcija nakon događaja „dugme miša otpušteno“ šalje na USB periferiju ASCII kod (karakter) indeksa kojim se prekida kretanje sistema, odnosno uzima vrednos 0x00 iz tabele 3.1 i taj parametar prosleđuje funkciji <code>motors_handler</code> programske podrške mikrokontrolera.
Parametar	Opis
<code>object sender</code>	Informacija o tome ko je izazvao događaj
<code>MouseEventArgs e</code>	Informacija o kom se događaju radi
Povratna vrednost	Opis
<code>void</code>	-
Poziva metode	<code>usbDongle.Write</code>

Ekvivalentni par funkcija za događaje „dugme miša pritisnuto“ i „dugme miša otpušteno“ postoji i za ostalih sedam dugmića, u nastavku neće biti navođeni.

Funkcija	Opis
trackBarPwmUpper_MouseUp	Funkcija nakon događaja „dugme miša otpušteno“ na klizaču kojim se podešava faktor ispunje za gornji motor, informaciju sa klizača pretvara u tekstualni zapis koji šalje na USB periferiju na osnovu koga će programska podrška za mikrokontroler postaviti novi faktor ispunje za gornji motor. Ekvivalentna funkcija postoji i za donji motor.
Parametar	Opis
<code>object sender</code>	Informacija o tome ko je izazvao događaj
<code>MouseEventArgs e</code>	Informacija o kom se događaju radi
Povratna vrednost	Opis
<code>void</code>	-
Poziva metode	<code>usbDongle.Write</code>

Funkcija	Opis
usbDongle_DataReceived	Funkcija nakon događaja „podaci primljeni na USB periferiju“ primljene podatke pretvara iz tekstualnog zapisa u brojevi, kako bi se moglo vršiti iscrtavanje trenutne pozicije na slici
Parametar	Opis
object sender	Informacija o tome ko je izazvao događaj
SerialDataReceivedEventArgs e	Informacija o kom se događaju radi
Povratna vrednost	Opis
void	-
Poziva metode	usbDongle.Read

Funkcija	Opis
timer1_Tick	Funkcija nakon isteka intervala kola vremenske kontrole 1, ukoliko su pristigli novi podaci osvežava sliku sa novom pozicijom sistema
Parametar	Opis
object sender	Informacija o tome ko je izazvao događaj
EventArgs e	Informacija o kom se događaju radi
Povratna vrednost	Opis
void	-
Poziva metode	Poziva metode za rad sa grafikom

Funkcija	Opis
pictureBox1_DoubleClick	Nakon dvoklika na polje sa slikom vrši se osvežavanje slike unutar polja, sa dodatkom braon kružića koji će biti postavljen na mestu dvoklika na slici i koji predstavlja zadatu poziciju na koju se treba pozicionirati. Dalje se zadata pozicija sa slike se pretvara u bročane podatke na osnovu kojih se određuje u kom smeru treba da se kreće sistem kako bi najbrže došao do željne pozicije, i sve to se šalje na USB periferiju koja te podatke prosleđuje na reflektorsko mesto
Parametar	Opis
object sender	Informacija o tome ko je izazvao događaj
EventArgs e	Informacija o kom se događaju radi
Povratna vrednost	Opis
void	-
Poziva metode	Poziva metode za rad sa grafikom, <code>usbDongle.Write</code>

Funkcija	Opis
checkBoxUpper_CheckedChanged	Funkcija dozvoljava ili zabranjuje rad dugmićima koji imaju veze sa gornjim motorom, i na taj način dozvoljava dozvoljava ili zabranjuje rad gornjem motoru. Ekvivalentna funkcija postoji i za donji motor
Parametar	Opis
object sender	Informacija o tome ko je izazvao događaj
EventArgs e	Informacija o kom se događaju radi
Povratna vrednost	Opis
void	-
Poziva metode	Poziva metode za podešavanja osobina dugmića

4. Zaključak

U prikazanom rešenju akcenat je stavljen na spregu mehanike i elektronike, koji su namenski konstruisani i napravljeni. Tu su se javljali različiti problemi mehaničke prirode, usled nesavršenosti izrađenih komponenti, kao i manjim odstupanjima pri njihovoj montaži. Sve to uzrokovalo je povećani utrošak energije za pokretanje motora, zbog preskakanja i gubitaka na mehaničkim prenosima. Realizacija mehaničkog dela platforme nije bila predmet razmatranja u ovom diplomskom radu.

Jedan od problema bio je taj da su motori stvarali previše buke što svakako nije poželjno bez obzira u kakvoj će primeni biti korišćen sistem. Problem je rešen zamenom metalnih zupčanika zupčanicima od fiberglasa koji ne prenose zvuk. U narednoj fazi biće pažljivije razmotren izbor pogonskih motora, kao i mehaničkih prenosnika. Rešavanje tih problema bilo je dragoceno za sticanja iskustva i novih saznanja iz oblasti mehanike, odnosno mehatronike.

Logičan korak napred bio bi upuštanje u problematiku vezanu za uključivanje i isključivanje, odnosno dimovanje svetala. Ta problematika je izostavljena sa ciljem da se napravi funkcionalna višenamenska pokretna platforma, koja bi se kasnije mogla koristiti u različite svrhe.

Mehanička koncepcija sistema je takva da se umesto reflektora na pokretnu platformu može montirati kamera za video-nadzor ili scensko snimanje, laser za diskoteke, kao i druge naprave sličnih gabarita i funkcionalnosti. Time je ostavljen širok dijapazon mogućih unapređenja i različitih primena. Tako se kasnije više pažnje može obratiti na specifičnosti primene, dok su problemi vezani za mehaničku konstrukciju uređaja dobro poznati i, što je najvažnije, rešeni.

5. Literatura

- [1] Vladimir Kovačević: *Logičko projektovanje računarskih sistema I –projektovanje digitalnih sistema*, Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, 2001
- [2] L. Nađ: *Mikrosenzori i aktuatori*, Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, 2003
- [3] Texas Instruments: *CC2510/CC2511 Datasheet*,
<http://focus.ti.com/docs/prod/folders/print/cc2510f32.html>
- [4] Texas Instruments: *Network Protocol for Sub-1 GHz, 2.4 GHz and IEEE 802.15.4 RF ICs*, <http://www.ti.com/simpliciti>
- [5] Cirrus Logic: *SA-571HZ Datasheet*
<http://www.cirrus.com/en/press/kits/SA306-SA57.html>
- [6] Honeywell: *Hall Effect Sensing and Application*,
<http://content.honeywell.com/sensing/prodinfo/solidstate/technical/hallbook.pdf>
- [7] LIN Consortium: *LIN Specification Package Revision 2.0*, <http://www.lin-subbus.org>
- [8] Christian Gross: *Beginning C# 2008 From Novice to Professional*, APRESS, 2008