



ŠOLSKI CENTER CELJE
Srednja šola za kemijo, elektrotehniko in računalništvo

RAČUNALIŠKO VODEN AVTO

RAZISKOVALNA NALOGA

Avtor:
Tilen Klinar

Mentor:
Gregor Kramer, u.d.i.e.

Mestna občina Celje, Mladi za Celje
Celje, marec 2014

KAZALO

1. POVZETEK.....	4
2. UVOD IN HIPOTEZE.....	5
2.1 UVOD.....	5
2.2 HIPOTEZE	5
3. OPIS RAZISKOVALNEGA PROBLEMA	5
3.1 OPIS RAZISKOVALNIH METOD.....	5
4. OSREDNJI DEL NALOGE.....	6
4.1 PREDSTAVITEV KOMPONENT	6
4.1.1 ATmega8 – 16PU	7
4.1.2 xBee	9
4.1.3 L298 DUAL FULL-BRIDGE DRIVER.....	11
4.1.4 - Micro Servo	13
4.1.5 RC-280SA.....	14
4.2 PRIPRAVA OGRODJA.....	15
4.3 IZDELAVA VEZJA.....	16
4.4 PISANJE PROGRAMA	21
4.5 RAZPRAVA.....	31
5. ZAKLJUČEK	32
6. Viri.....	33
7. ZAHVALA	34

KAZALO SLIK

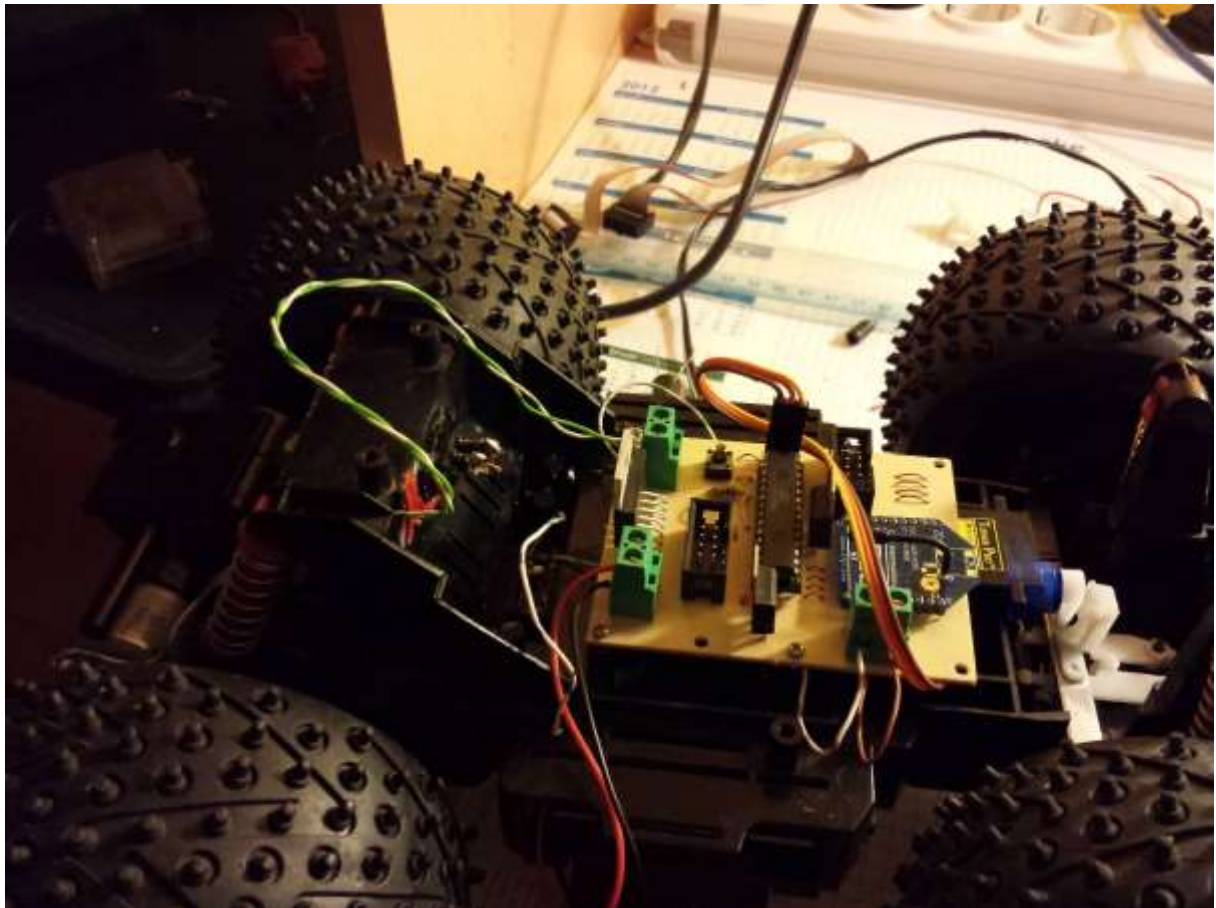
Slika 1: Avto.....	4
Slika 2: Porti in pini ATmega8.....	7
Slika 3: ATmega8-16PU Blokovna Shema.....	8
Slika 4: xBee.....	9
Slika 5: Blokovna shema vezave ATmega8 in xBee.....	10
Slika 6: L298.....	11
Slika 7: Blokovna shema vezave ATmega8 in L298.....	12
Slika 8: TowerPro Micro Servo 9g.....	13
Slika 9: Mere motorja.....	14
Slika 10: Slika popravljenega ogrodja brez gredi in z še vgrajenim motorjem.....	15
Slika 11: Gred.....	15
Slika 12: Skica gredi.....	15
Slika 13: Shema Vezja.....	16
Slika 14: Shema povezave med komponenti.....	17
Slika 15: Prvi poskus tiskanega vezja.....	18
Slika 16: Drugo tiskano vezje.....	19
Slika 17: Primer nadgradnega vezja.....	19
Slika 18: Tiskano Vezje Zgoraj.....	20
Slika 19: Tiskano Vezje Spodaj.....	20
Slika 20: Slika rezkalnika.....	20
Slika 21: Tiskano vezje programatorja v Eaglu.....	21
Slika 22: Programator.....	21
Slika 23: Testni program za krmiljenje L298.....	22
Slika 24: Program za testiranje servo motorja.....	23
Slika 25: Graf pulza.....	24
Slika 26: Primer uravnavanja hitrosti s PWM.....	24
Slika 27: Program za PWM z NI myDAQ.....	25
Slika 28: Nadzorna plošča za NY myDAQ.....	25
Slika 29: Vezje z NI myDAQjem.....	26
Slika 30: Načrt vezja PWM z NI myDAQ.....	26
Slika 31: Program za testiranje PWM z ATmega8.....	27
Slika 32: Celoten program 1.del.....	28
Slika 33: Celoten program 2.del.....	29
Slika 34: Vmesnik za vodenje avtomobila.....	30

KAZALO TABEL

Tabela 1: Opis pinov xBeeja.....	9
Tabela 2: Absolutno Maksimalne Vrednosti L298.....	11
Tabela 3: Vrednosti Motorja.....	14
Tabela 4: Krmilni signali za L298.....	22

1. POVZETEK

V nalogi vam bom predstavil moj projekt, ki ga imenujem računalniško vodeni avto. Zadeva deluje tako, da avto krmili računalnik z uporabo dveh xBee brezžičnih RF modulov v vezju pa podatke sprejme Atmelov mikrokontroler ATmega8. Te podatke sprejme iz sprejemnika vgrajenega v vezju ta pa brezžično sprejme vse podatke iz oddajnika, ki je priključen v računalnik. Na mikrokontroler sta priključena TowerPro SG90 – Micro Servo ter čip L298 in ta z motorjem.



Slika 1: Avto

2. UVOD IN HIPOTEZE

2.1 UVOD

Moja prva naloga je bila izbira komunikacije med vozilom in računalnikom. Z brezžičnim vodenjem se srečujemo pri hobi modelarstvu, kjer gre za vodenje avtomobilov, letal ali čolnov z visokofrekvenčno povezavo. Področje brezžičnega vodenja se še posebej v zadnjem času seli v vojaško industrijo pri vodenju brezpilotnih letal (UAV) in mobilnih robotov, ki opravljajo različne naloge. Vse vodene naprave se lahko znajdejo v območju slabše komunikacije ali lahko komunikacijo celo izgubijo. V takšnih primerih morajo imeti možnost avtonomnega vodenja, kar pomeni, da se bodisi ustavijo ali potujejo v smeri ponovne vzpostavitve komunikacije. Največ časa sem posvetil krmiljenju hitrosti motorja in servo motorja, ki krmili vozilo. V tem primeru so nastale težave z močjo motorja, ki je potrebovalo veliko več toka in napetosti kot ostali del vezja ter s kontroliranjem hitrosti motorja.

2.2 HIPOTEZE

- Avtomobilu lahko obračamo kolesa, nastavljamo hitrost in smer vožnje
- Avtomobil je voden daljinsko z računalnikom ali pametnim telefonom
- Razdalja med oddajnikom in avtomobilom mora biti vsaj 30m

3. OPIS RAZISKOVALNEGA PROBLEMA

Raziskovanje se je začelo novembra, ko smo se odločali, kaj bomo naredili za raziskovalno nalogo. Ob premišljevanju sem se spomnil, da imam doma pokvarjeni daljinsko vodeni avto kateremu se je pokvarilo vezje in krmilje. Nato sem internet prebrskal z namenom, da bi ugotovil ali je možno uporabiti staro ohišje za ustvaritev novega avta. Za raziskavo sem moral vedeti nekaj o brezžični komunikaciji ter mikrokontrolerjih, ki sprejemajo podatke od računalnika, jih obdelati in narediti to, kar podatek od njega zahteva. Program sem pisal v programu imenovanem Bascom AVR, ki služi programiranju AVR mikrokontrolerjev; program s katerim sem napisali program, ki vodil xBee preko računalnika je Microsoft visual studio.

3.1 OPIS RAZISKOVALNIH METOD

Pri raziskovalni nalogi sem najprej na spletu poiskal avtomobile na brezžično vodenje in njihove načine krmiljenja ter vodenja. Na podlagi teh raziskanj sem se odločil, kakšne komponente potrebujem.

4. OSREDNJI DEL NALOGE

4.1 PREDSTAVITEV KOMPONENT

V naslednjih straneh vam bom podrobno predstavil naslednje komponente, ki sem jih uporabil pri izdelavi:

- ATmega8 – 16PU
- xBee
- L298 DUAL FULL-BRIDGE DRIVER
- TowerPro SG90 - Micro Servo
- RC-280SA 12V DC Motor

4.1.1 ATmega8 – 16PU

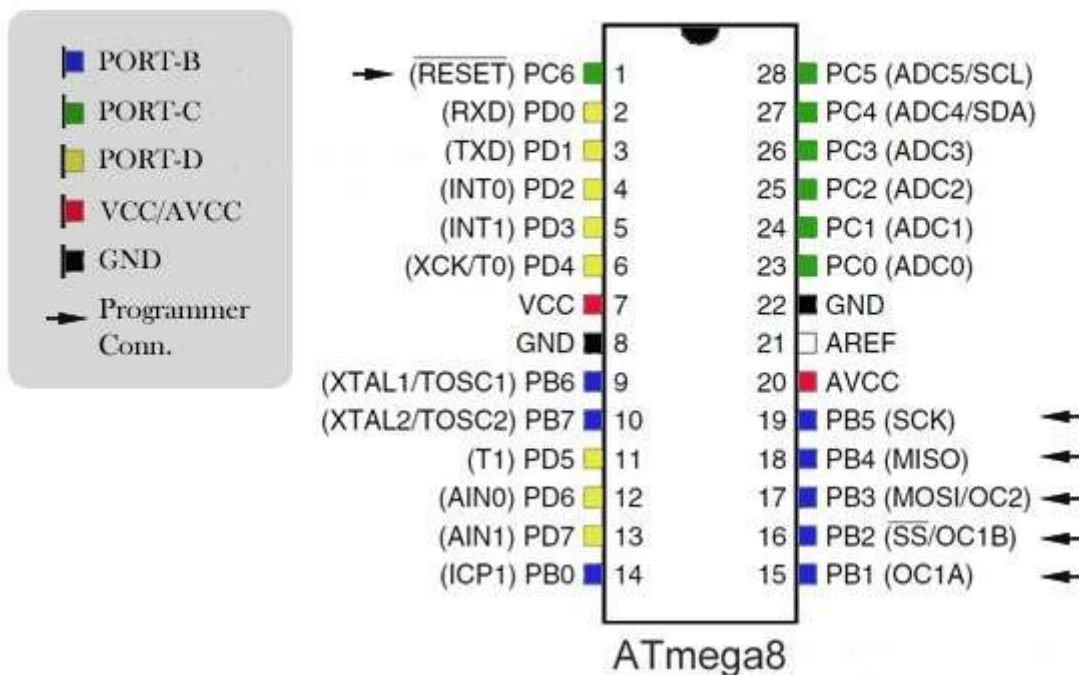
ATmega8 – 16PU je mikrokrmilnik iz družine ATmega. Mikrokrmilnik ima 3 porte in vsak port ima 8 pinov, skupaj je to 24 logičnih vhodiv/izhodov, ki jih lahko nastavimo po želji.

Mikrokrmilnik lahko programiramo z več različnimi računalniškimi programi, kot na primer Bascom AVR, AVR Studio..., nato pa program zapišemo na mikrokrmilnik preko teh ali drugih programov, ki se vklopijo v LPT največkrat pa USB vhod.

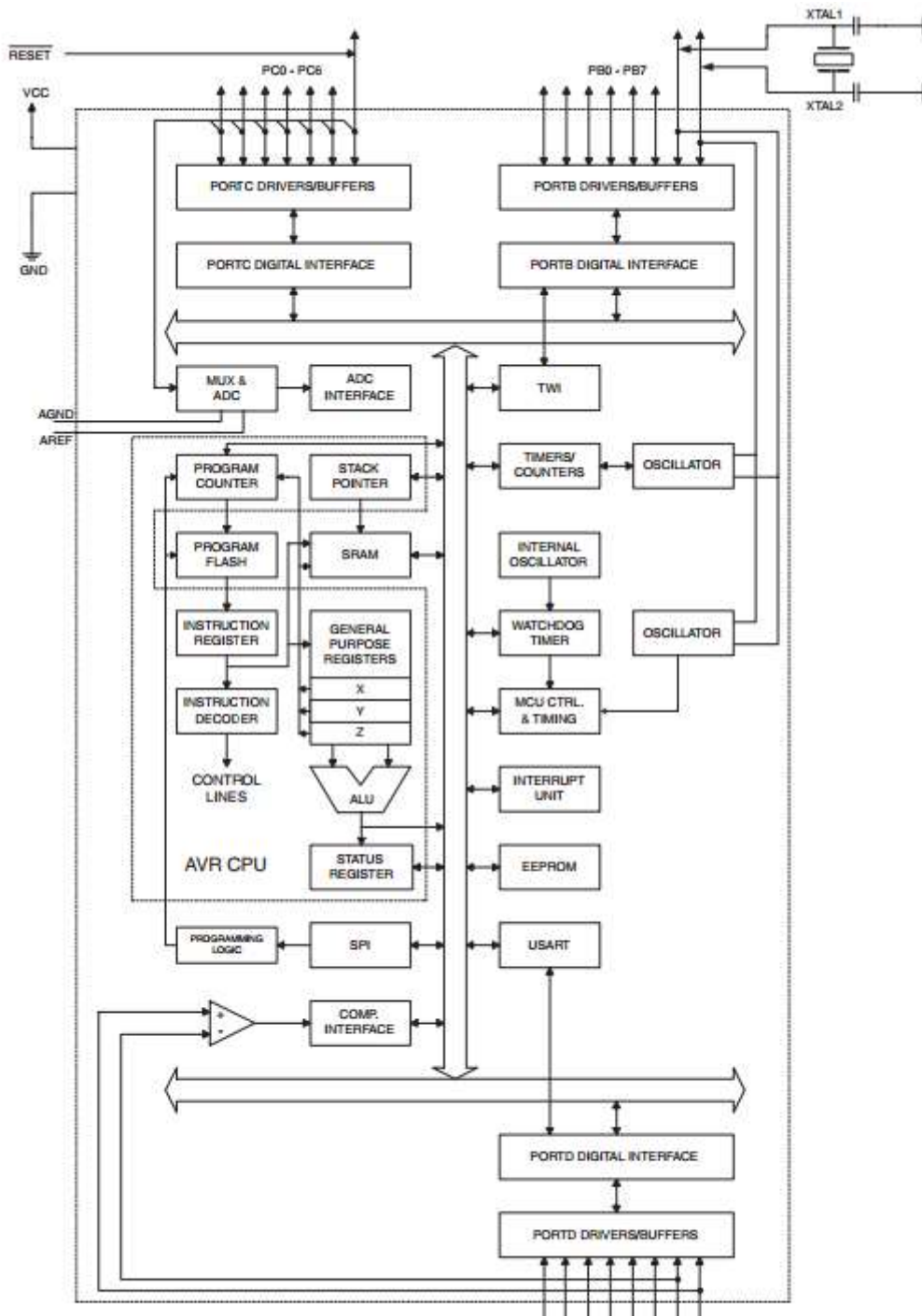
Lastnosti:

- visoka zmogljivost in nizka poraba,
- 8KB spomina v sistemu za program,
- 512B EEPROM-a,
- 1KB internega SRAM-a,
- dva 8-bitna časovnika z ločenim prescalerjem,
- 16-bitni časovnik,
- trije PWM kanali,
- 6 kanalni ADC z 10 bitno natančnostjo,

Hitrost (MHz)	Napajanje (V)	Mikrokontroler	Ohišje	Območje delovanja
16	4.5-5.5	ATmega8 – 16PU	28P3	Industrijsko (-40°C do 85°C)



Slika 2: Porti in pini ATmega8



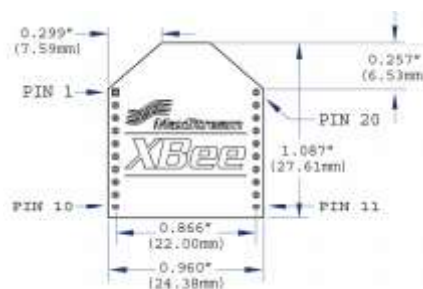
Slika 3: ATmega8-16PU Blokovna Shema

4.1.2 xBee

Z pomočjo xBeeja je možna vzpostavitev brezžične povezave med računalnikom in ATmega8 mikrokontrolerjem.

Lastnosti:

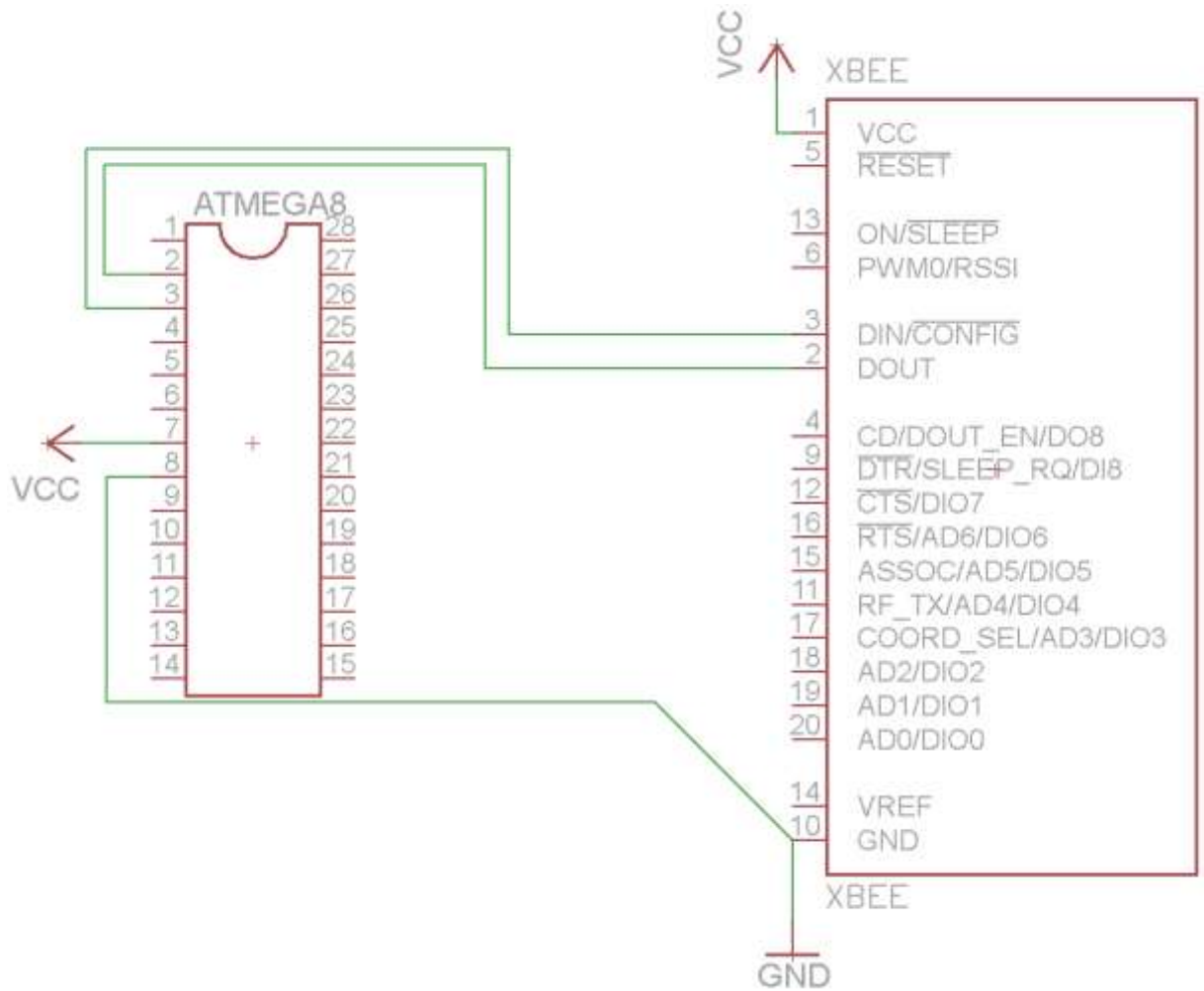
- Napetost delovanja: 2,8-3,4V
- Tok delovanja: 50mA (3,3V)
- Delovanje v zaprtih prostorih: do 30m
- Delovanje na prostem: do 90m
- Moč oddajanja: 1mW (0 dBm)



Slika 4: xBee

Pin #	Ime	Smer	Opis
1	VCC	-	Napajanje
2	DOUT	Izhod	UART Podatkovni Izhod
3	DIN/CONFIG	Vhod	UART Podatkovni Vhod
4	DO8*	Izhod	Digitalni Izhod 8
5	RESET	Vhod	Reset (najmanj 200ns puls)
6	PWM0/RSSI	Izhod	PWM Izhod 0/Indikator moči RX signala
7	PWM1	Izhod	PWM Izhod 1
8	(reserved)	-	Ne povezati
9	DTR / SLEEP_RQ / DI8	Vhod	Pin za kontrolo spanja ali Digitalni Vhod 8
10	GND	-	GND
11	AD4 / DIO4	Vhod/Izhod	Analogni vhod 4 ali digitalni vhod/izhod 4
12	CTS / DIO7	Vhod/Izhod	Clear-to-Send Flow Control ali Digitalni vhod/izhod 7
13	ON / SLEEP	Izhod	Indikator statusa za modul
14	VREF	Vhod/Izhod	Napetostna Referenca za analogne/digitalne vhode
15	Associate / AD5 / DIO5	Vhod/Izhod	Analogni vhod 5 ali digitalni vhod/izhod 5
16	RTS / AD6 / DIO6	Vhod/Izhod	Request-to-Send Flow Control, Analogni Vhod 6 ali Digitalni vhod/izhod 6
17	AD3 / DIO3	Vhod/Izhod	Analogni Vhod 3 ali Digitalni vhod/izhod 3
18	AD2 / DIO2	Vhod/Izhod	Analogni Vhod 2 ali Digitalni vhod/izhod 2
19	AD1 / DIO1	Vhod/Izhod	Analogni Vhod 1 ali Digitalni vhod/izhod 1
20	AD0 / DIO0	Vhod/Izhod	Analogni Vhod 0 ali Digitalni vhod/izhod 0

Tabela 1: Opis pinov xBeeja



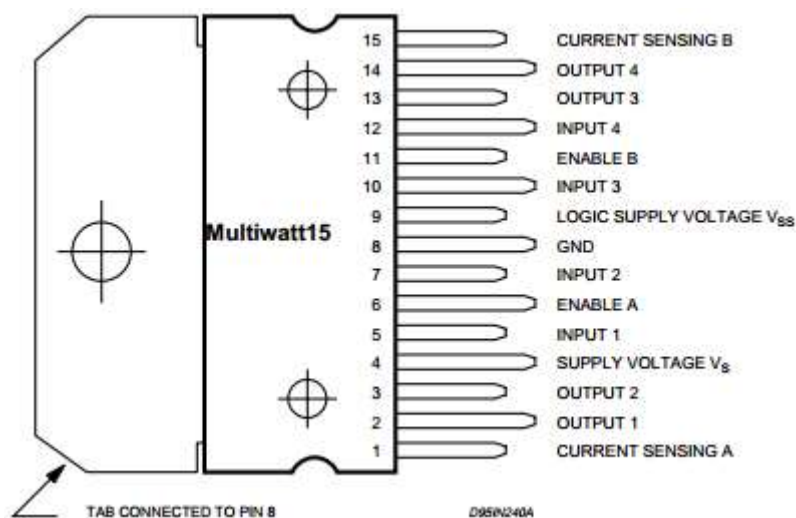
Slika 5: Blokovna shema vezave ATmega8 in xBee

4.1.3 L298 DUAL FULL-BRIDGE DRIVER

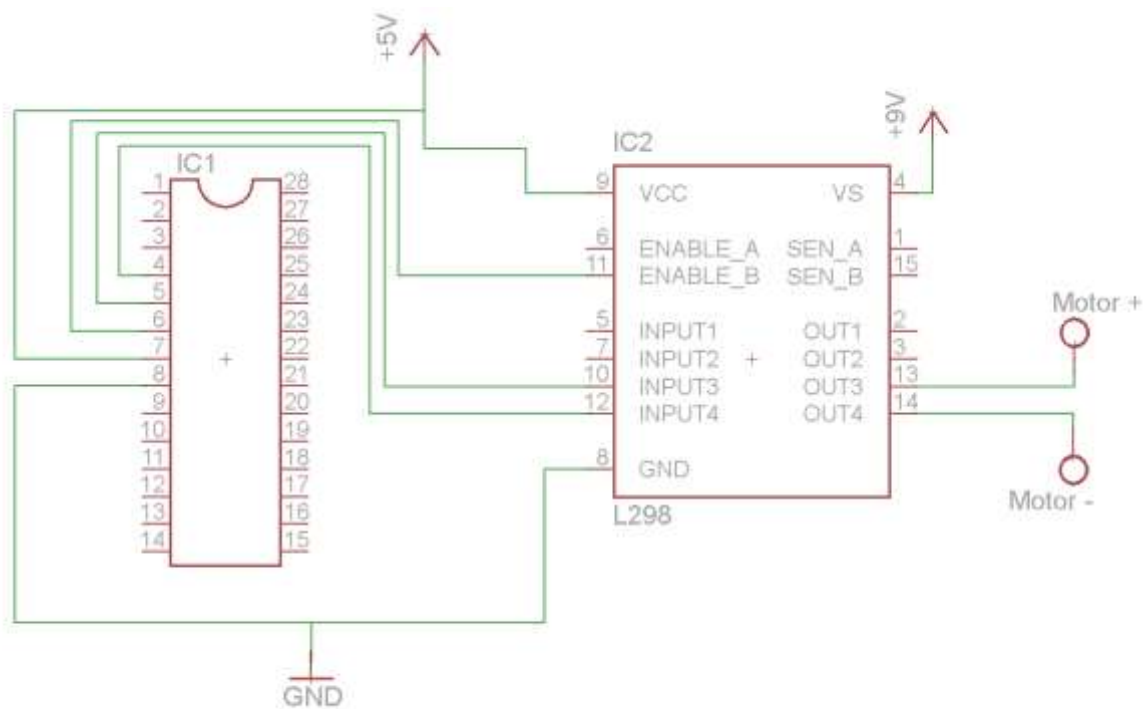
L298 je visokonapetostni in visoko tokovni driver za motorje narejen za krmiljenje induktivnih bremen, kot so releji, elektromagneti, DC in koračni motorji. Na L298 se lahko priključita, do dva induktivna bremena. Z uporabo dveh enable vhodov lahko krmiljeni napravi krmilimo neodvisno od ostalih logičnih vhodov (pomembno za PWM – Pulse Width Modulation), ki je uporabljen za nadzor napetosti na bremenu.

Absolutno Maksimalne Vrednosti			
Symbol	Parameter	Vrednost	Enota
V_s	Napajanje	50	V
V_{ss}	Logično Napajanje	7	V
V_i, V_{en}	Vhodna in Enable Napetost	-0.3 do 7	V
I_o	Maksimalni Izhodni tok (vsak kanal) - Ne Ponavljivi ($t = 100\mu s$) -Ponavljajoči (80% ON, -20% OFF; $t_{on} = 10ms$) -DC Operacija	3 3,5 2	A A A
V_{sens}	Napetost Zaznavanja	-1 do 2,3	V
P_{tot}	Skupna Oddana Moč ($T_{case} = 75^\circ C$)	25	W
T_{op}	Junction Operating Temperature	-25 do 130	$^\circ C$
T_{stg}, T_j	Storage and Junction Temperature	-40 to 150	$^\circ C$

Tabela 2: Absolutno Maksimalne Vrednosti L298



Slika 6: L298



Slika 7: Blokovna shema vezave ATmega8 in L298

4.1.4 - Micro Servo

TowerPro SG90 je zanesljiv in ugoden analogni servo motor. Rotacijsko območje motorja je od 0° do 180° z 60° v 0.1s pri 4.8V. Servo motor je majhen in lahek (9g) ampak ima relativno veliki navor za njegovo velikost (pri 4.8V ima 1.8kg-cm). Motor krmilimo s pomočjo PWM (spreminjanje DutyCycla).



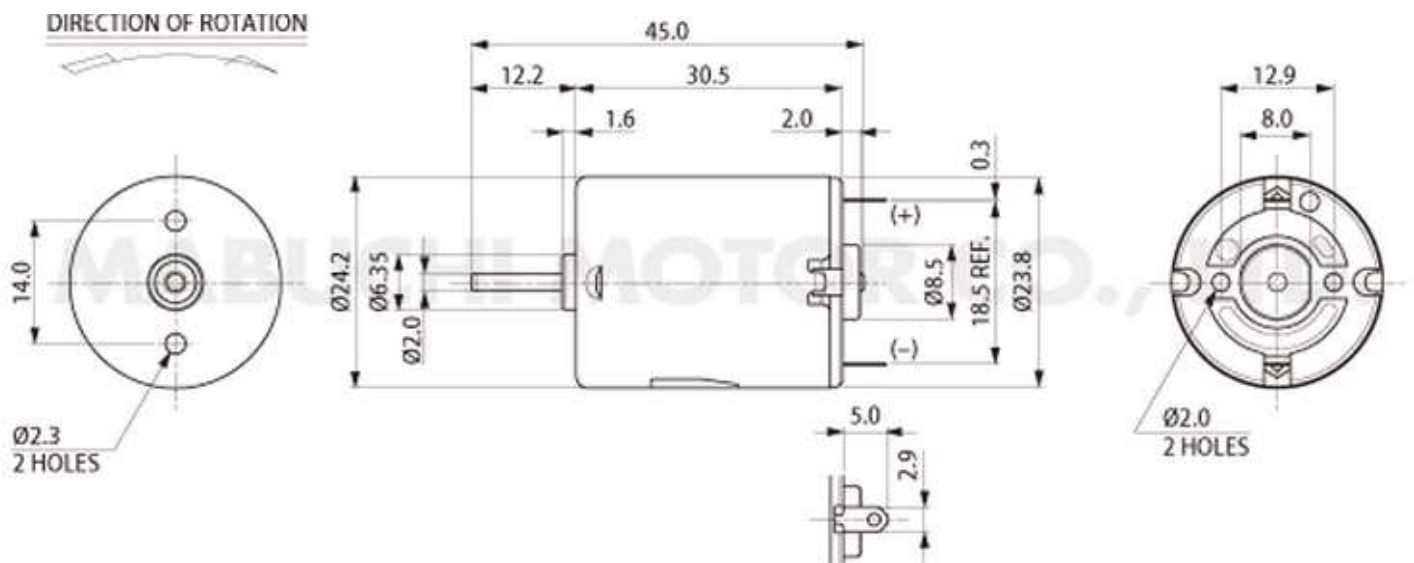
Slika 8: TowerPro Micro Servo 9g

4.1.5 RC-280SA

Ugodni in močni 12V DC motor z velikim navorom. Uporablja se v igračah, hobi modelarstvu, robotih, avtomatizaciji in še mnogo več. Na zadnji strani motorja ima dva stika preko katerih ga priključimo na priključek ali neposredno prispajkamo žico.

Model	Napajanje [V]		Brez Obremenitve		Pri Maksimalni Učinkovitosti					
	Operacijsko Območje V	Nominalna V	Hitrost vrtljaj/min	Tok A	Hitrost vrtljaj/min	Tok A	Navor		Izhodna Moč W	
							mN·m	g·cm		
RC-280SA	20120	4,5-12	9	1130	0.14	9540	0.76	4.06	41.3	4.04

Tabela 3: Vrednosti Motorja



Slika 9: Mere motorja

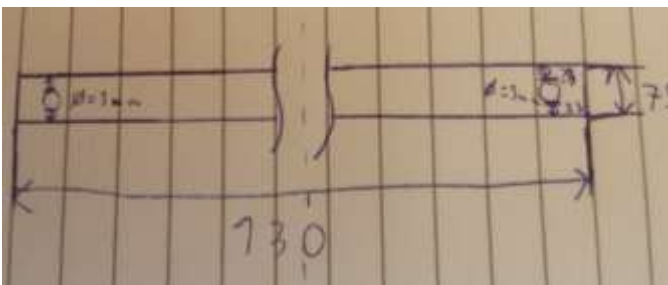
4.2 PRIPRAVA OGRODJA

Zaradi poškodb ogrodja in krmilnega sistema sem najprej moral popraviti ogrodje preden sem se odpravil na izdelavo vezja. Veliki del popravil je bil preprost, kot so čiščenje, odstranitev starega vezja, zamenjati motor od krmilja, ... Najtežji del opravil je bil popraviti zlomljeno krmilno gred, ki se je poškodovala.



Slika 10: Slika popravljenega ogrodja brez gredi in z še vgrajenim motorjem

Novo gred sem naredil iz nerjavečega jekla.



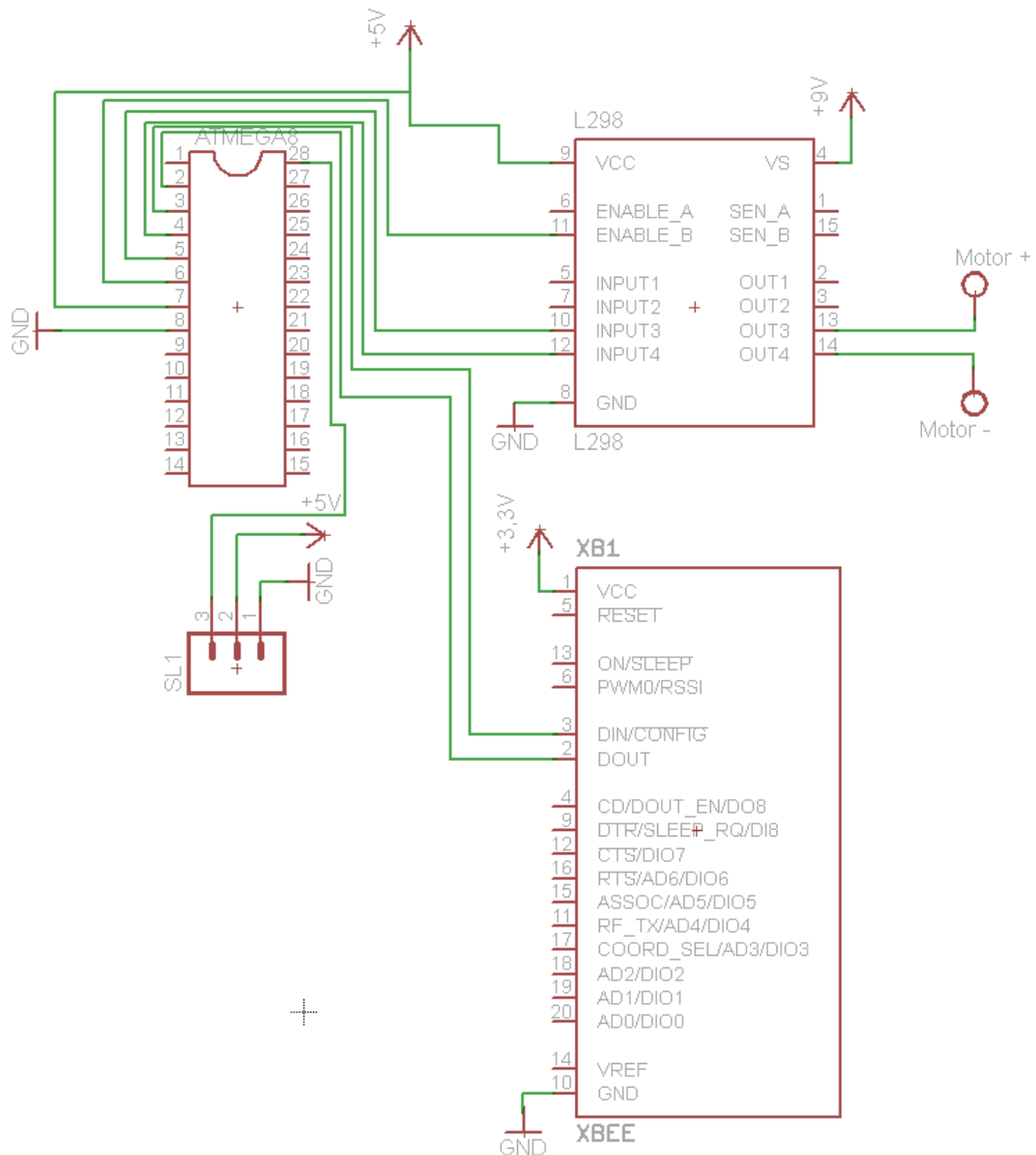
Slika 12: Skica gredi



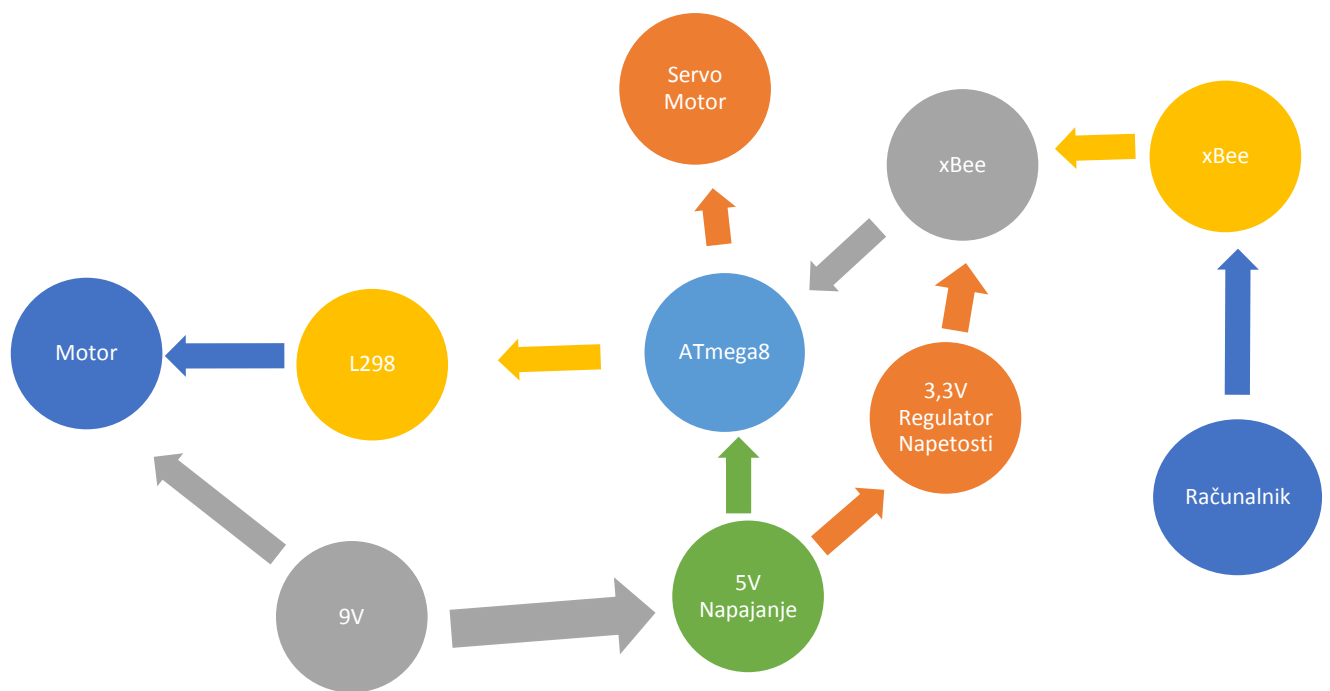
Slika 11: Gred

4.3 IZDELAVA VEZJA

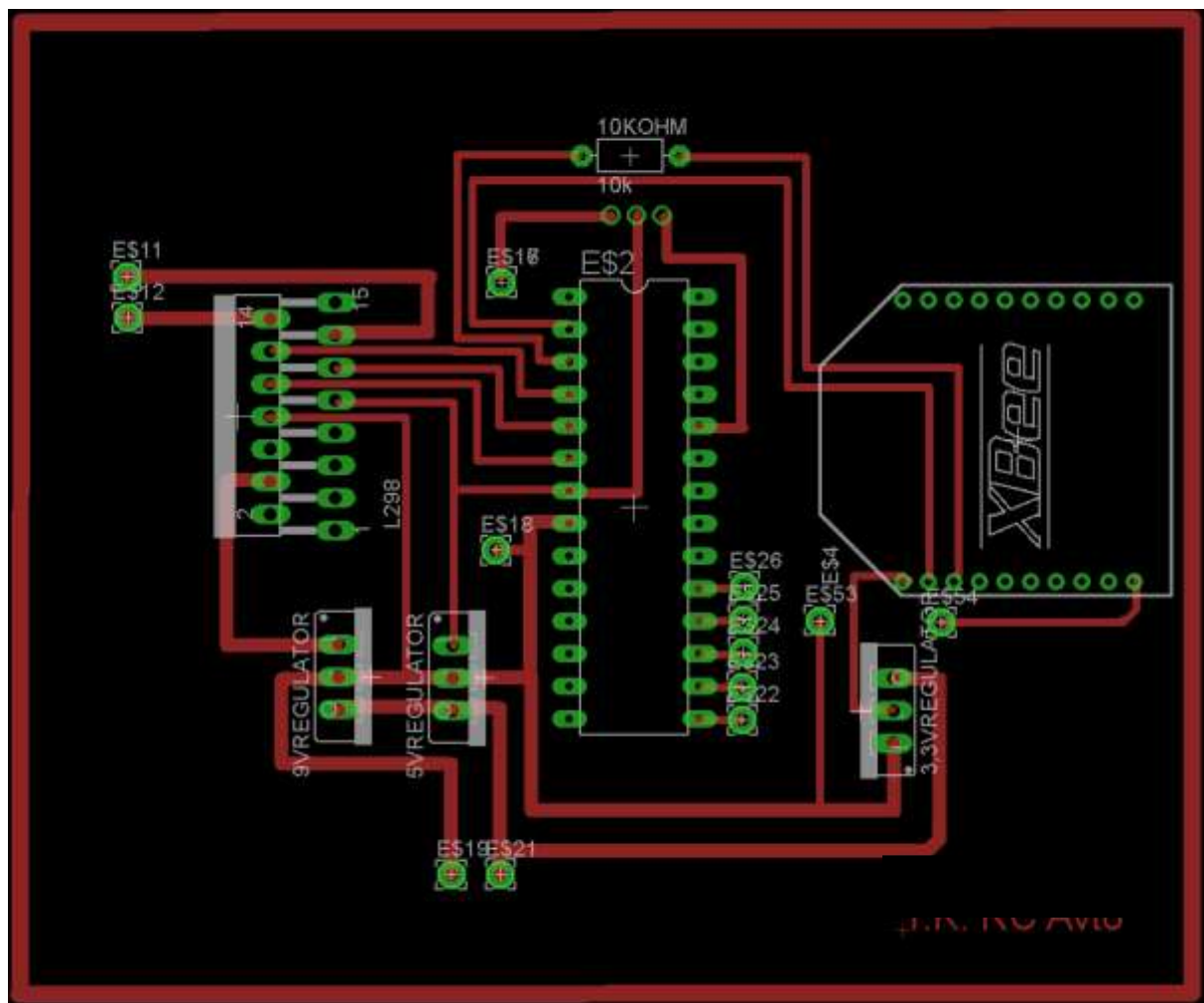
Po končanih popravilih in po končanem brskanju po internetu za vse komponente, ki sem jih potreboval je nastopil čas za izdelavo vezja. Ampak pred izdelavo vezja je potrebno narediti načrt. Preprosti načrt sem naredil v programu imenovanem Eagle prav tako je tiskano vezje narisano v Eaglu.



Slika 13: Shema Vezja



Slika 14: Shema povezave med komponenti

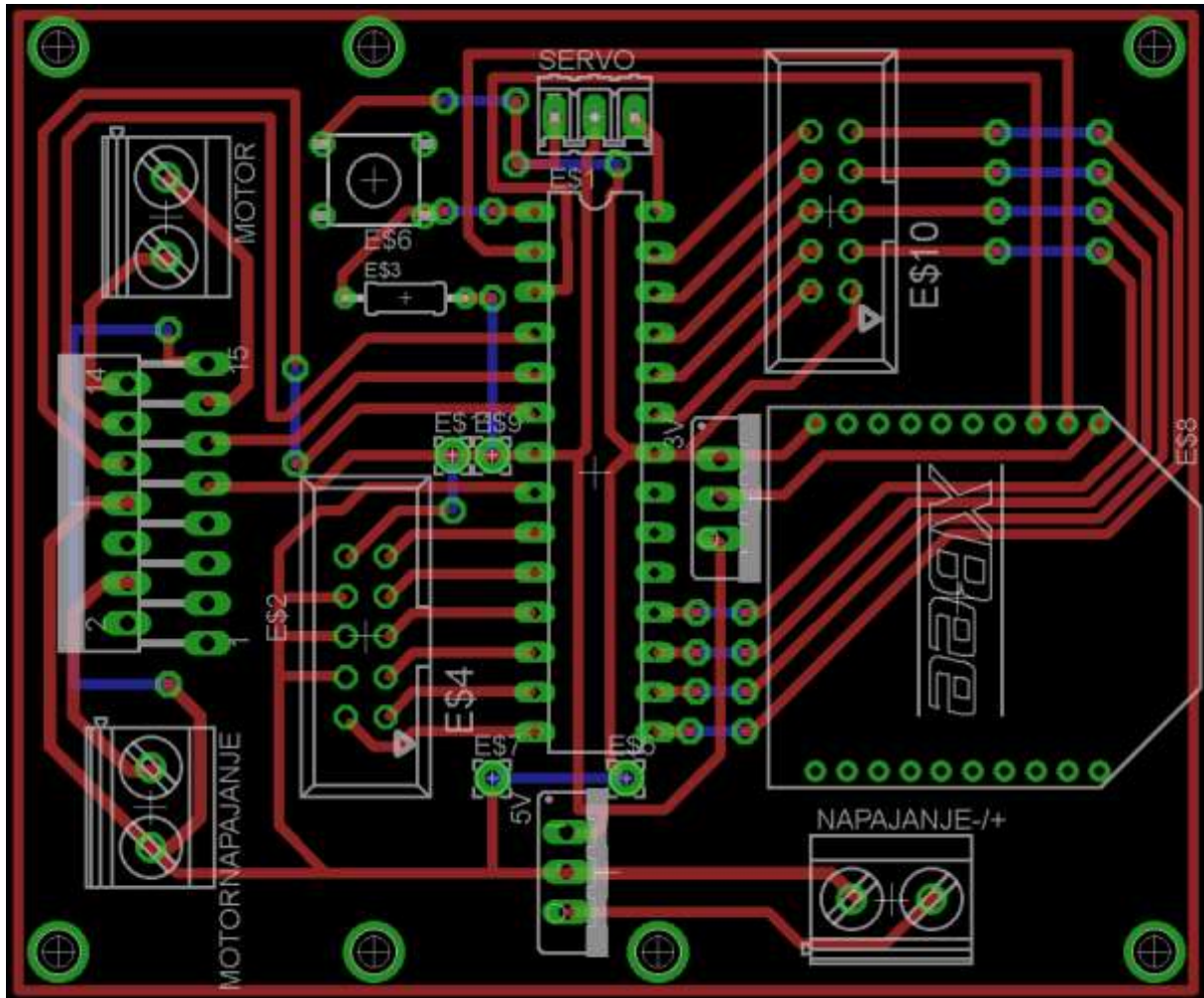


Slika 15: Prvi poskus tiskanega vezja

Prvi poskus izdelave vezja je bil neuspešen zaradi par kritičnih napak kot so: nepravilno povezani regulatorji zaradi tega razloga ni do motorja prišla zadostna količina napetosti za uspešni zagon. Med napakami je tudi to, da so pri rezkanju povezave, ki so bile med dvema pinoma, nenamerno stanjšale zaradi velikosti rezkalne glave pri obrobjanju pinov.

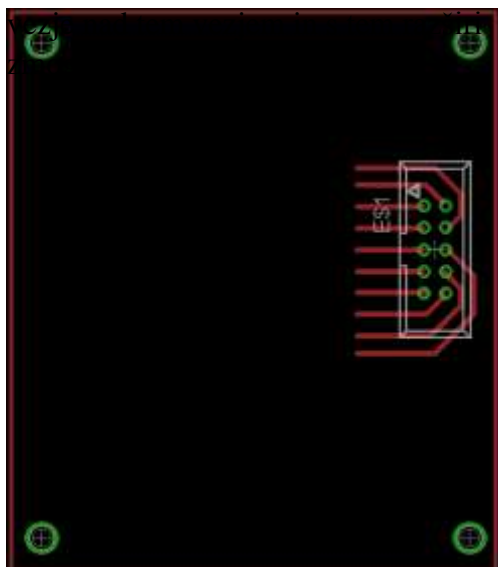
Po dolgem razmisleku in ponovnem pregledu vezja sem se odločil, da bom ustvaril čisto novega ter odpravil vse napake prejšnjega. Po bolj podrobnejšem pregledu tiskanega vezja sem se tudi odločil, da bom dal motor na drugačno napajanje kot preostali del vezja.

Iz te napake sem se naučil nove reči na katere predtem nisem pazil, kot vpliv velikosti rezkalne glave na ostale povezave v območju pinov in drugih vezav. Zaradi tega sem pri naslednjem tiskanem vezju pazil na to, da nebi bilo ponovnega konflikta med pini in vezmi.



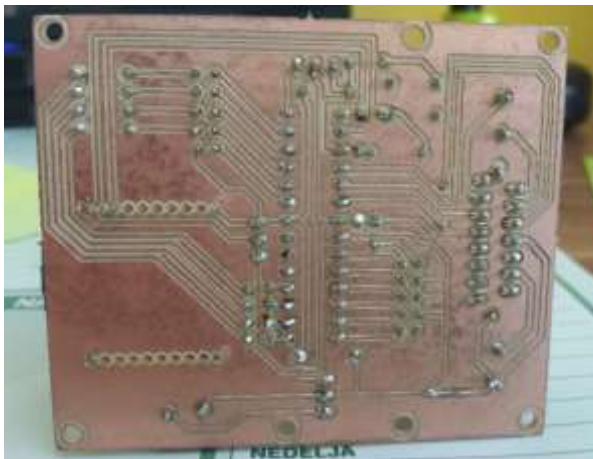
Slika 16: Drugo tiskano vezje

V drugem poskusu sem spremenil veliko v vezju. Kot že prej omenjeno sem dodal posebej napajanje za motor, to vezje tudi vsebuje tipko za reset ATmega8 mikrokontrolerja. Dodal sem tudi deset pinske priključke s katerim omogočam lažjo nadgradnjo vezja in boljšim izkoristkom mikrokontrolerja. Pripravil sem tudi luknje za pritrditev vezja na avto. Vezje bo pritrjeno s tremi vijaki, preostale štiri luknje so pa namenjene nadgradnji vezja z uporabo distančnikov se lahko namesti še eno



Slika 17: Primer nadgradnega vezja

V tem primeru nadgradnje lahko povežemo deset pinski priključek z deset pinskim priključkom na glavnem vezju in s tem dodam katerokoli drugo napravo kot mikrokrmilnik, rele, LED diode,...



Slika 19: Tiskano Vezje Spodaj



Slika 18: Tiskano Vezje Zgoraj

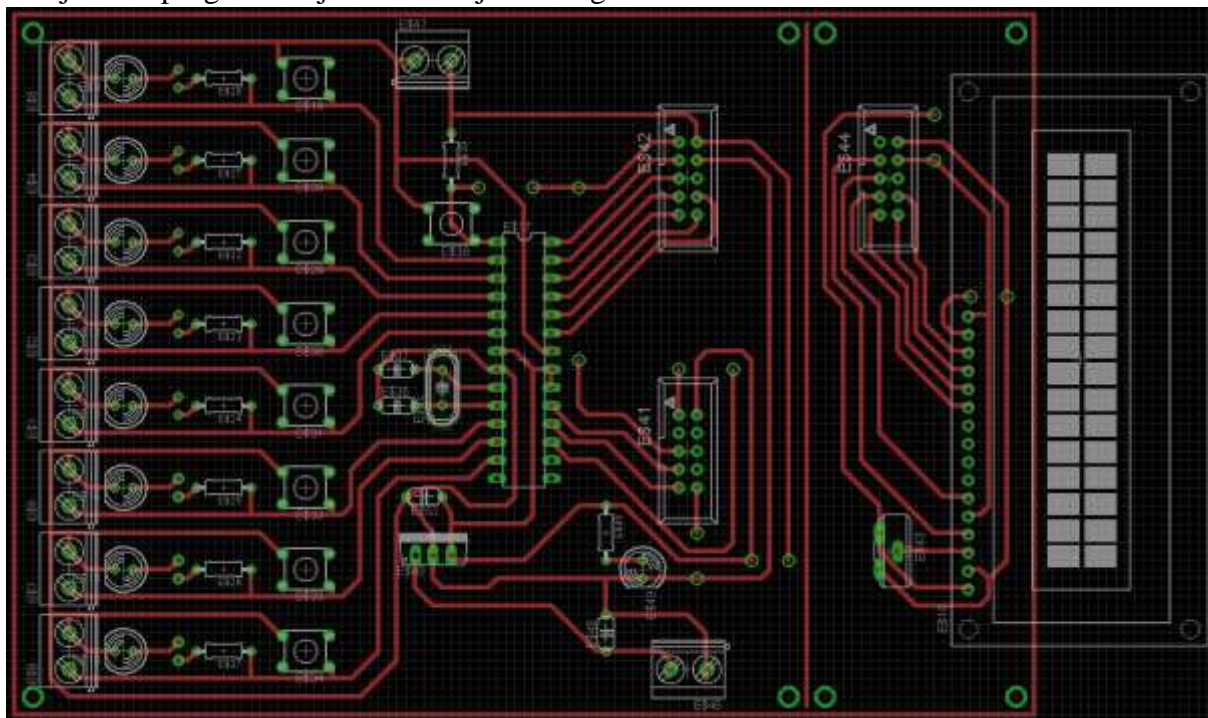
Vezje sem rezkal v šoli z CNC rezkalnikom LPKF ProtoMat S62. Rezkalnik je narejen za fino obdelavo, uporablja se za graviranje, rezkanje in vrtanje.



Slika 20: Slika rezkalnika

4.4 PISANJE PROGRAMA

Po končani gradnji in merjenju napetosti na tiskanem vezju sem se odpravil pisanju testnih programov za testiranje delovanja motorja in servomotorja. Za pisanje tega programa sem uporabil BASCOM – AVR in za zapisovanje na mikrokrmilnik sem uporabil eXtreme Burner – AVR, vezje s katerim sem prenesel podatke iz računalnika v mikrokrmilnik je posebej narejeno za programiranje in testiranje ATmega8 krmilnika.



Slika 21: Tiskano vezje programatorja v Eaglu



Slika 22: Programator

To vezje za programiranje ATmega8 sem sestavljal posebej z namenom za programiranje in testiranje ATmega8 serije čipov. Vezje ima deset pinskih priključek ki služi priključitvi 2x16 digitalnem zaslону ali katerikoli drugo vezje. To vezje mi je tudi služilo pri testiranju prejšnjega tiskanega vezja. Z priključitvijo E\$42 deset pinskega priključka z tem, ki je na vezju za avto lahko z uporabo tipk pošiljam signale na drugo vezje in tako neposredno kontroliram drugo vezje.

Sprva sem sprogramiral testni program za L298, ki vodi motor.

```
'ATmega8
$regfile = "m8def.dat"
'crystal = 8MHz
$crystal = 8000000
'na portu D so priključki ki so povezani na L298
'potrebna je oznacitev vsakega posebjaj saj pina
'PORTD.0 in PORTD.1 sta povezana z xBeejem in
'PIND.1 je input
Config PORTD.2 = Output
Config PORTD.3 = Output
Config PORTD.4 = Output
'test vrtenje motorja naprej nazaj za 10s
Do
  Wait 2
  'Input 1
  PORTD.2 = 1
  'Enable
  PORTD.3 = 1
  'Input 2
  PORTD.4 = 0
  Wait 10
  'ustavitev za 2s
  PORTD.3 = 0
  Wait 2
  'druga smer
  PORTD.2 = 0
  PORTD.3 = 1
  PORTD.4 = 1
  Wait 10
  PORTD.3 = 0
Loop
```

Slika 23: Testni program za krmiljenje L298

Input 1	Enable	Niput 2	Ukrep
0	0	0	Motor nima napajanja
0	0	1	Motor nima napajanja
0	1	0	Motor zabremza in se ustavi
0	1	1	Motor se vrti nazaj
1	0	0	Motor nima napajanja
1	0	1	Motor nima napajanja
1	1	0	Motor se vrti naprej
1	1	1	Motor zabremza in se ustavi

Tabela 4:Krmilni signali za L298

Nato sem še napisal program s katerim sem vodil servo motor. Ta program je bil veliko težje za napisati, saj uporablja časovnike in potrebni so bili izračuni za razmerja časa.

```
'ATmega8
$regfile = "m8def.dat"
'crystal = 8MHz
$crystal = 8000000

Dim Krmilo_vred As Word
Dim I As Byte
Dim X As Byte
'casovnik0 perioda stetja = 0,1us, perioda za servo = 100us
Config TIMER0 = Timer , Prescale = 8
'casovnik0 zacne steti pri 155 (od 255) 0,1*100 = 100us
TCNT0 = 155
'zacne steti
Enable TIMER0
'omogocimo prekinitve
Enable Interrupts
On TIMER0 , Isr_timer0_krmilo

Config PORTC.5 = Output
|
Do
'MIN 10 MAX 20!!! Vec pa je nevarnost da se
'Zobniki v servomotorju zataknejo

For X = 10 To 20 Step 1
    Krmilo_vred = X
    Waitms 500
Next
x = 10

Loop

'*****
'podprogram za vodenje servomehanizma za krmilo
Isr_timer0_krmilo:
If I = 0 Then
    PORTC.5 = 1
End If

I = I + 1

'servo krmilo
If I = Krmilo_vred Then
    PORTC.5 = 0
End If

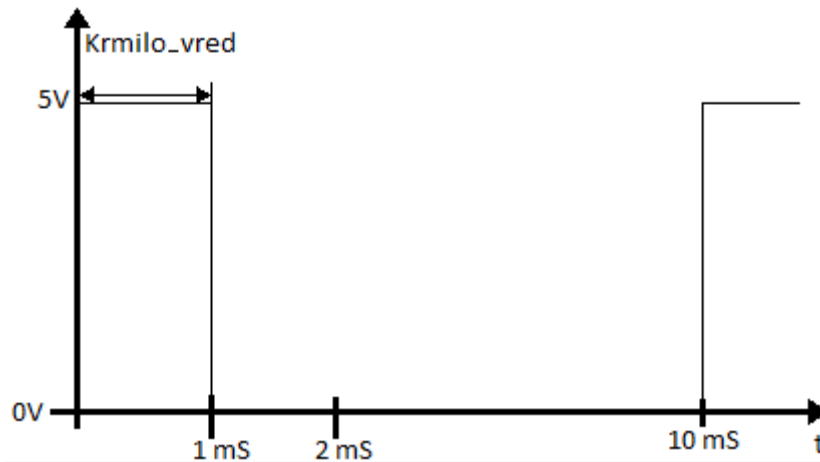
If I = 200 Then
    I = 0
End If

    TCNT0 = 155
Return
'*****
```

Slika 24: Program za testiranje servo motorja

Ta program deluje tako, da uporabi časovnik (TIMER0) z namenom, da vsakih 100 μ S prekine delovanje loopa in skoči v Isr_timer0_krmilo in izvede ta podprogram.

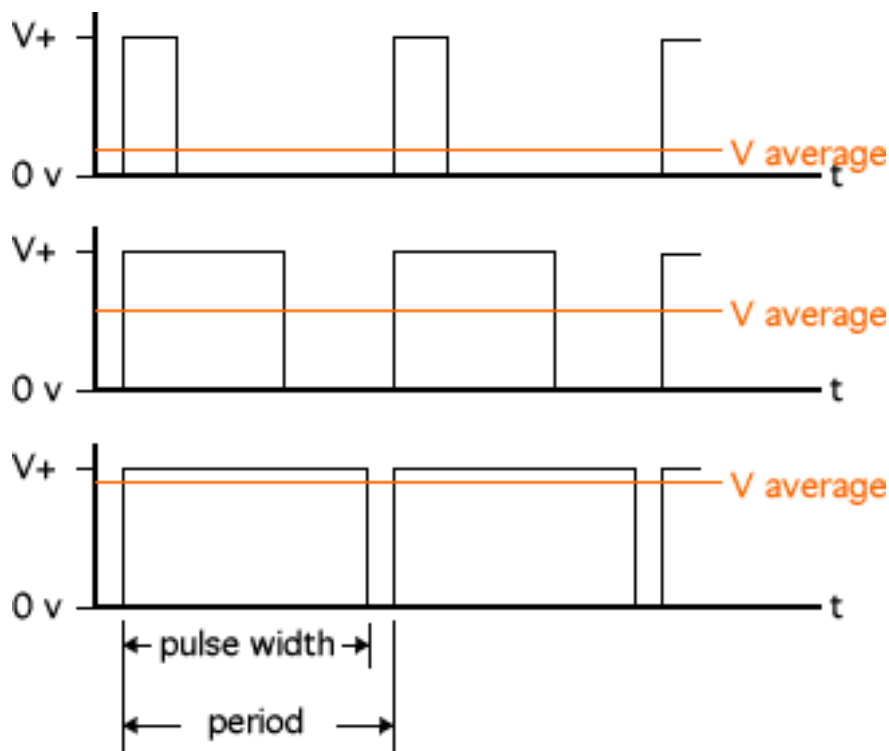
Podprogram deluje tako, da predčasno prekine signal iz PORTC.5. Temu pravimo Pulse Width Modulation (PWM).



Slika 25: Graf pulza

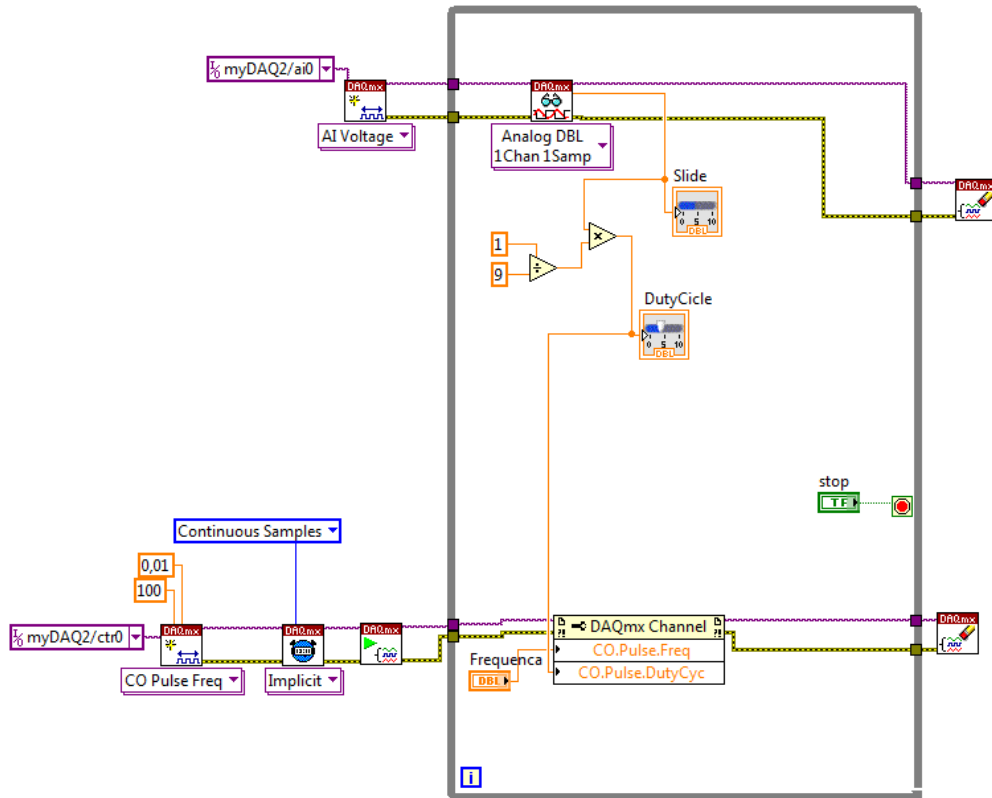
Celotna dolžina periode je 20ms. Minimalna dolžina za pulz je 1ms, maksimalna dolžina za pulz pa je 2ms, v primeru da je pulz daljši ali krajši servo motor izgubi komunikacijo.

Po uspešnem testiranju motorja in servo motorja sem šel pisati program za uravnavanje hitrosti motorja. Program temelji na enakih principih kot program, ki ga uporabljam za servo motor ampak tokrat bodo pulzi daljši, frekvenca pa bo nižja zaradi problemov z inducirano napetostjo. Perioda je v tem primeru 100ms, prevajalni cikel pa se nastavlja v 10 korakih, torej po 10ms/korak.

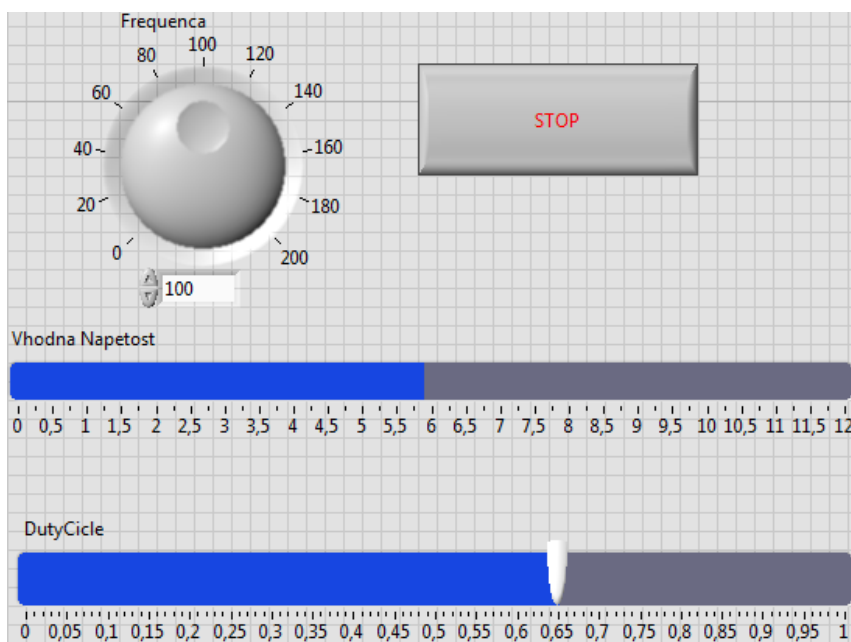


Slika 26: Primer uravnavanja hitrosti s PWM

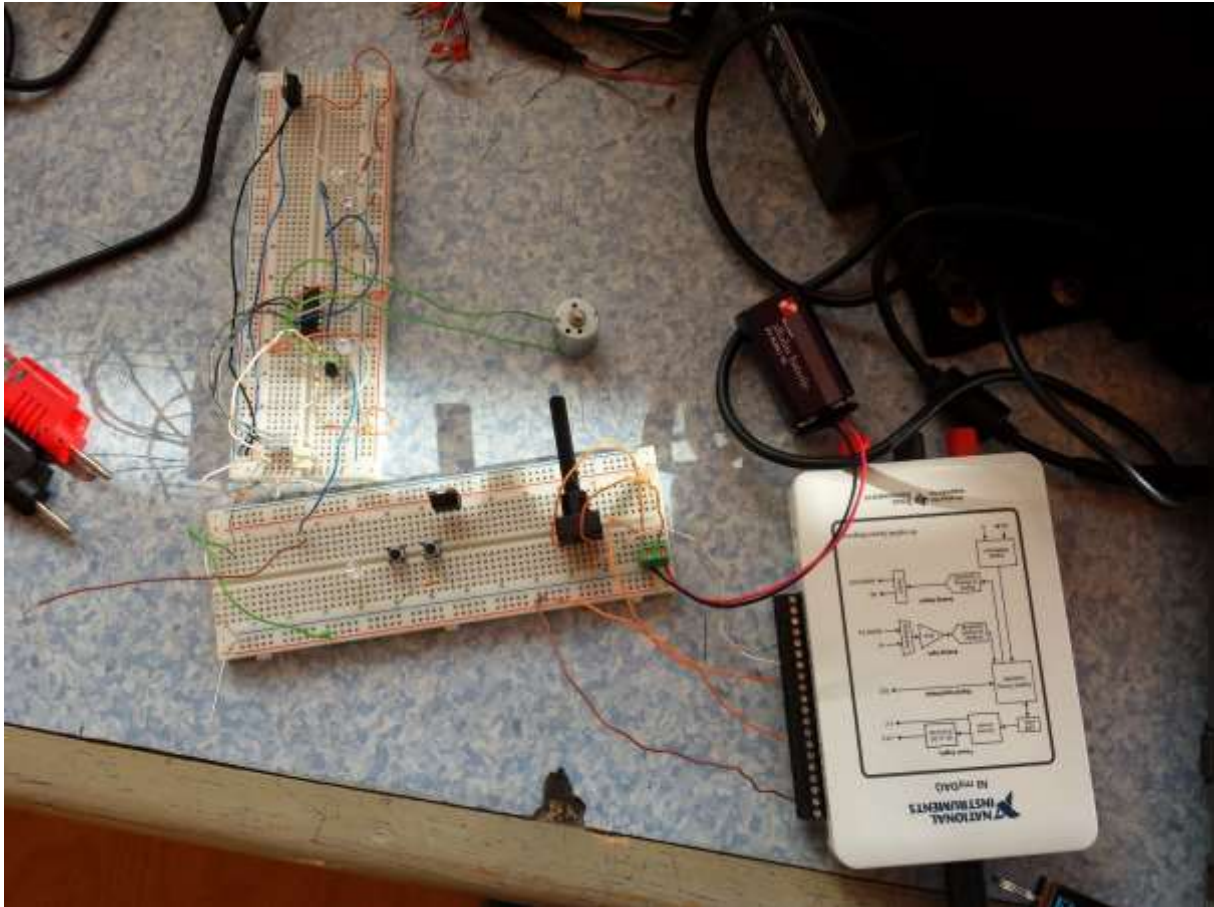
Prve poskuse s PWM na motorju sem opravil z NI myDAQ (National Instruments my Data Acquisition Device), ki lahko z uporabo counterja (primerljivo s timerjem v ATmega8) ustvarim pulze različnih dolžin. Program je napisan v programu LabVIEW 2012.



Slika 27: Program za PWM z NI myDAQ

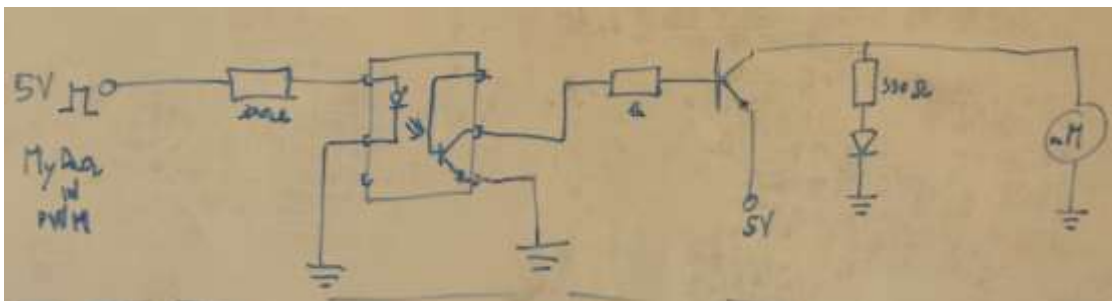


Slika 28: Nadzorna plošča za NY myDAQ



Slika 29: Vežje z NI myDAQjem

Z uporabo analognih vhodov na myDAQ sem lahko meril napetost, ki izhaja iz 9V baterije vezane na potenciometer, kar mi omogoča spreminjanje napetosti. Količina napetosti pa vpliva na hitrost motorja tako, da spreminja vrednost DutyCycla, ki je lahko število med 0,01 in 1. Ampak zaradi tega ker so vrednosti baterije med 0-9V sem jih moral pomnožiti z $\frac{1}{9}$ ($DutyCycle = V_{\text{vhodnaNapetost}}(V) \times \frac{1}{9}$), da lahko DutyCycle z večjo natančnostjo nadziram ($Hitrost(\%) = DutyCycle \times 100$).



Slika 30: Načrt vezja PWM z NI myDAQ

Po uspešnem testiranju PWMja z myDAQjem sem se odpravil na izdelavo programa v bascomu. Tokrat sem ponovno uporabil timer ampak zaradi uporabe TIMER0 za servo motor sem se odločil uporabiti TIMER1 saj prej ali slej bom moral programa združiti v enega samega.

```
'ATmega8
$regfile = "m8def.dat"
'crystal = 8MHz
$crystal = 8000000

Dim X As Byte
Dim I As Byte
Dim Hitrost_vred As Word
'perioda je lus
Config TIMER1 = Timer , Prescale = 8
'TIMER1 je 16bitni s tem razlogom ima veliko večjo število kot TIMER0
TCNT1 = 55535

On TIMER1 , Isr_timer1_hitrost
Enable Interrupts

Config PORTD.2 = Output
Config PORTD.3 = Output
Config PORTD.4 = Output

Do
'hitrost_vred je od 0 do 10  0=0% ; 10 = 100%
PORTD.2 = 1
PORTD.4 = 0
For I = 0 To 10 Step 1
Wait 1
Hitrost_vred = I
Next
I = 0
Wait 1
PORTD.2 = 0
PORTD.4 = 1
For I = 0 To 10 Step 1
Wait 1
Hitrost_vred = I
Next
I = 0
Loop

'*****

Isr_timer1_hitrost:
If X = 0 Then
PORTD.3 = 1
End If

X = X + 1

'hitrost
If X = Hitrost_vred Then
PORTD.3 = 0
End If

If X = 10 Then
X = 0
End If

TCNT1 = 55535
Return
```

Slika 31: Program za testiranje PWM z ATmega8

Po končanem testiranju programov je napočil čas, da združimo vse programe in dodamo še program za xBee s katerim krmilimo ATmega8.

```

Sregfile = "m8def.dat"
Scrystal = 8000000
'Ssim

' casovnik0 perioda stetja = 0,1us, perioda za servo = 100us
Config TIMER0 = Timer , Prescale = 8
TCNT0 = 155

Config TIMER1 = Timer , Prescale = 8           'perioda je 1us
TCNT1 = 55535

On TIMER0 , Isr_timer0_krmilo
On TIMER1 , Isr_timer1_hitrost

DDRC.5 = 1
PORTC.5 = 0

'motor - izhodi
DDRD.2 = 1
DDRD.3 = 1
DDRD.4 = 1

Dim Hitrost_vred As Word , Krmilo_vred As Word , Kot_avtomatsko As Word
Dim Kot_cas As Byte , Sprejem As String * 1 , Zacetna_hitrost As Byte
Dim I As Word
Dim X As Byte
Dim J As Byte

Enable Urxc
On Urxc , Getchar

Waitms 100

PORTD.3 = 0
' Config Servos = 1 , Servo1 = Portc.5 , Mode = Servo , Timer = Timer1
Enable TIMER0
Enable TIMER1
Enable Interrupts

Waitms 50

Do
Loop

'.....
'podprogram za vodenje servomehanizma za krmilo
Isr_timer0_krmilo:
If I = 0 Then
PORTC.5 = 1
End If

I = I + 1

'servo krmilo
If I = Krmilo_vred Then
PORTC.5 = 0
End If

'pri fosc 4MHz => I=200, pri 8MHz => I=400
If I = 200 Then
I = 0
End If

TCNT0 = 155
Return
'.....

Isr_timer1_hitrost:
If X = 0 Then
PORTD.3 = 1
End If

X = X + 1

```

Slika 32: Celoten program 1.del

```
' hitrost
If X = Hitrost_vred Then
PORTD.3 = 0
End If

If X = 10 Then
X = 0
End If

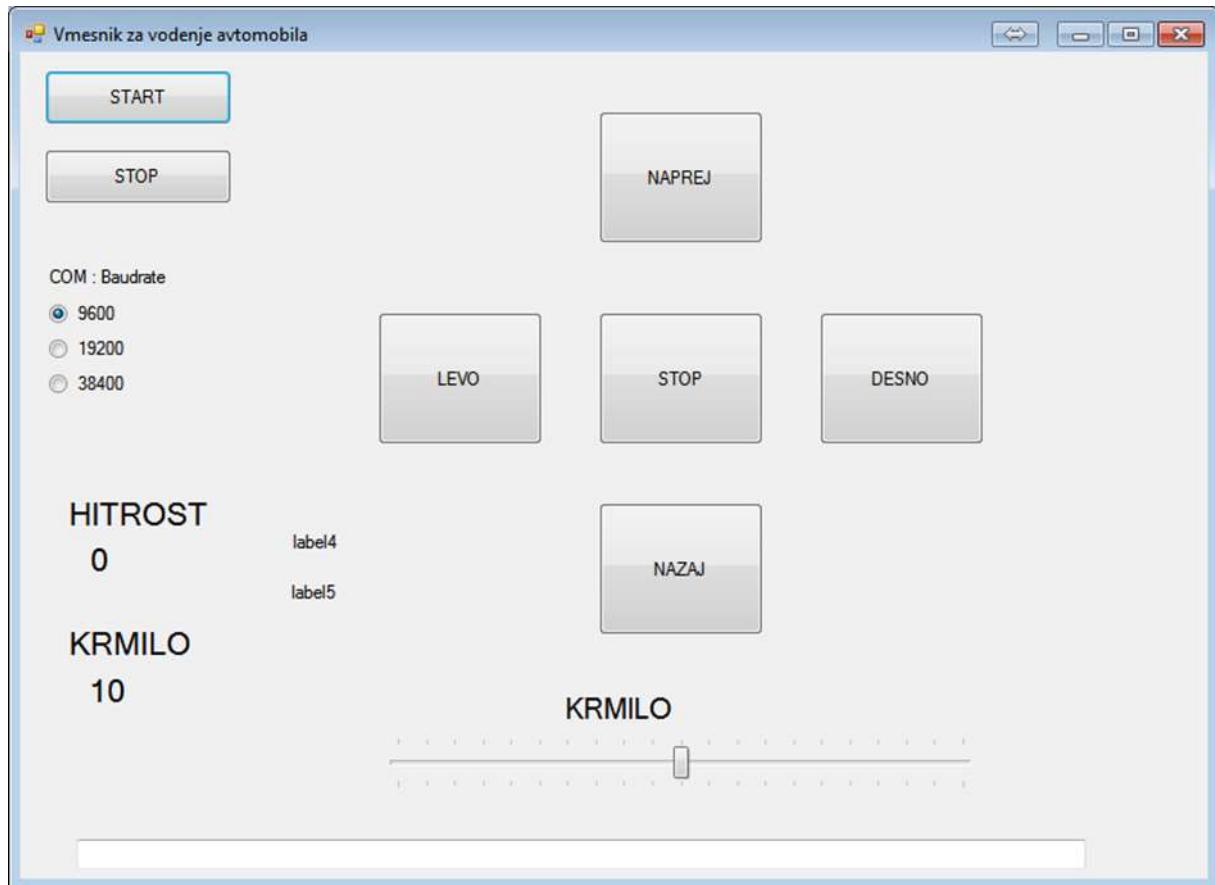
TCNT1 = 55535
Return

.....

Get char:
Sprejem = Inkey()
Select Case Sprejem
Case "1" : Krmilo_vred = Krmilo_vred + 1 ' povečaj za en korak
Case "2" : Krmilo_vred = Krmilo_vred - 1 ' zmanjšaj za en korak
Case "3" : Hitrost_vred = Hitrost_vred + 1 ' povečaj hitrost za en korak
Case "4" : Hitrost_vred = Hitrost_vred - 1 ' zmanjšaj hitrost za en korak
Case "5" : Hitrost_vred = 0 ' postavi hitrost na 0 - hitri STOP
Case "6" : Krmilo_vred = 15 ' postavi krmilo v nevtralo
End Select
Return

.....
```

Slika 33: Celoten program 2.del



Slika 34: Vmesnik za vodenje avtomobila

Program za vmesnik je napisan v Microsoft Visual Studio 2010 C# programskem okolju.

Oddajni modul xBeeja se nastavi kot serijski COM port, izbiramo lahko tudi različne hitrosti prenosa podatkov (baudrate). Z vsakim pritiskom na tipko se pošlje koda (številka 1,2...6), ki v mikrokrmilniku povzroči spremembo koraka v razdelku getchar.

4.5 RAZPRAVA

Hipoteze/Teze:

- Avtomobilu lahko obračamo kolesa, nastavljamo hitrost in smer vožnje
Avtomobil je zgrajen tako da se mu lahko obračajo kolesa z uporabo TowerPro servo motorja, nastavlja hitrost motorja z uporabo PWMja na enable pinu od čipa L298.

- Avtomobil je voden daljinsko z računalnikom ali pametnim telefonom
Avtomobil je voden preko uporabniškega vmesnika, ki se namesti na osebni računalnik. Komunikacija med računalnikom in avtomobilom je vzpostavljena brezžično s pomočjo dveh xBeejev. Problem lahko nastane pri večji hitrosti vožnje zaradi zakasnitev prenosa podatkov za krmilo in hitrost. Hitrost mora biti zato nastavljena bolj pazljivo. Delno rešitev predstavlja sredinski gumb, ki hkrati postavi hitrost na nič in krmilo v nevtralo. Za učinkovito in varno vodenje bi bilo potrebno dodelati boljši uporabniški vmesnik.

- Razdalja med oddajnikom in avtomobilom mora biti vsaj 30m
Z uporabo xBeeja, ki ima signalno razdaljo v zaprtih prostorih 30m in na odprtem 90m je možna vzpostavitev razdalje z najmanj 30m.

5. ZAKLJUČEK

Potrdil sem vse hipoteze. Z samim izgledom in delovanjem izdelka sem zelo zadovoljen.

Avto še ima veliko prostora za nadgrajevanje, kot dodajanje senzorjev, LED diod, daljinsko upravljanje z daljincem...

Pri tej raziskovalni nalogi sem se naučil veliko stvari in se seznanil z kar nekaj novimi stvarmi, ki mi bodo v življenju koristile. Naučil sem se tudi kako se lotiti izdelave kakšnega projekta ali reševanje problemov. Izkoristil sem tudi znanje pridobljeno v šoli, saj sem ga pri tej raziskovalni nalogi še kako potreboval.

6. Viri

ATMEL ATmega8 lastnosti. Dostopno na:

<http://pdf1.alldatasheet.com/datasheet-pdf/view/171443/ATMEL/ATMEGA8-16PU.html>

L298 DUAL FULL-BRIDGE DRIVER lastnosti. Dostopno na:

<https://www.sparkfun.com/datasheets/Components/General/L298N.pdf>

TowerPro SG90 - Micro Servo lastnosti. Dostopno na:

<http://www.servodatabase.com/servo/towerpro/sg90>

RC-280SA 12V DC Motor lastnosti. Dostopno na:

http://www.mabuchi-motor.co.jp/cgi-bin/catalog/e_catalog.cgi?CAT_ID=rc_280sa

Bascom. Dostopno na:

www.mcselec.com

Eagle. Dostopno na:

<http://www.cadsoftusa.com/>

7. ZAHVALA

Raziskovalno nalogo mi je uspelo narediti v celoti, zanjo sem porabil veliko časa, ampak še več bi ga porabil, če ne bi poznal profesorja oz. mojega mentorja Gregorja Kramerja, kateremu bi se rad iskreno zahvalil za ves čas in trud, ki si ga je vzel za pomoč pri vseh problemih na katere sem naletel.

Zahvalil bi se rad tudi Janku Holobarju, ki si je vzel čas za izdelavo vezij in strokovne pomoči pri izdelavi tiskanega vezja.