

Boe - Bot

Grupa autora

Upute za sastavljanje i programiranje robota

Zagreb, 2009 god.
radna verzija 2.1.

Sadržaj:

1.0 Boe-Bot.....	
1.1 Opis Boe-Bota.....	
1.2 Potreban alat.....	
1.3 Popis mehaničkih djelova.....	
2.0 Sastavljanje mehanike	
2.1 Postavljanje odstojnika i gumica	
2.2 Pripremanje servo motora	
2.3 Postavljanje servo motora na kućište robota	
2.4 Postavljanje nosača baterija na kućište	
2.5 Postavljanje kotača	
3.0 Sastavljanje elektronike	
3.1 Sastavljanje upravljačke jedinice „Board of education“	
3.2 Postavljanje pločice „Board of education“ na robota	
3.3 Priprema za rad	
4.0 Programiranje.....	
4.1 Kako započeti programirati	
4.1.1 Pisanje programskog koda.....	
4.1.2 Varijable za kontrolu pinova.....	
4.1.3 Definiranje varijabli.....	
4.1.4 Ispisivanje vrijednosti varijabli.....	
4.1.4 Konstante.....	
4.1.5 Brojevi i matematika.....	
4.1.6 Programske petlje.....	
4.1.7 Grananje programa.....	
4.2 Naredba „DEBUG“.....	
4.3 Sastavljanje i programiranje svijetlećih dioda.....	
4.3.1 Kako spojiti LED na robota?.....	
4.3.2 Programiranje LED.....	
4.4 Upravljanje robotom.....	
4.4.1 Spajanje i kalibracija motora.....	
4.4.2 Programiranje motora.....	
4.4.3 Dodavanje Piezzo zvučnika	
5.0 Senzori robota.....	
5.1 Taktični senzori.....	
5.2	

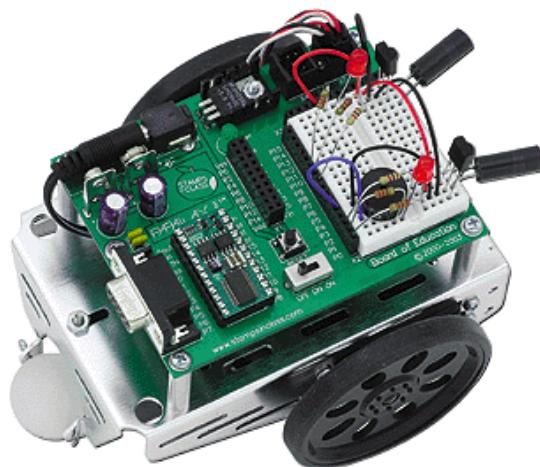
6.0 Popis naredbi Boe-Bota.....

7.0 Popis elemenata.....

1.0 Boe-Bot

1.1 Opis Boe-Bota

Robot Boe-Bot je mobilni robot baziran na Basic Stamp 2 modulu (BS2). Za pokretanje koristi dva servo motora, s kojima mu je ostvaren diferencijalni pogon. Za postavljanje senzora se koristi prototipna pločica na koju se mogu postavljati različiti senzori bez potrebe za lemjeljem. Također robot ima mogućnost nadogradnje pomoću konektora koji prihvataju AppMod module. Slika 1 prikazuje sastavljenog robota.



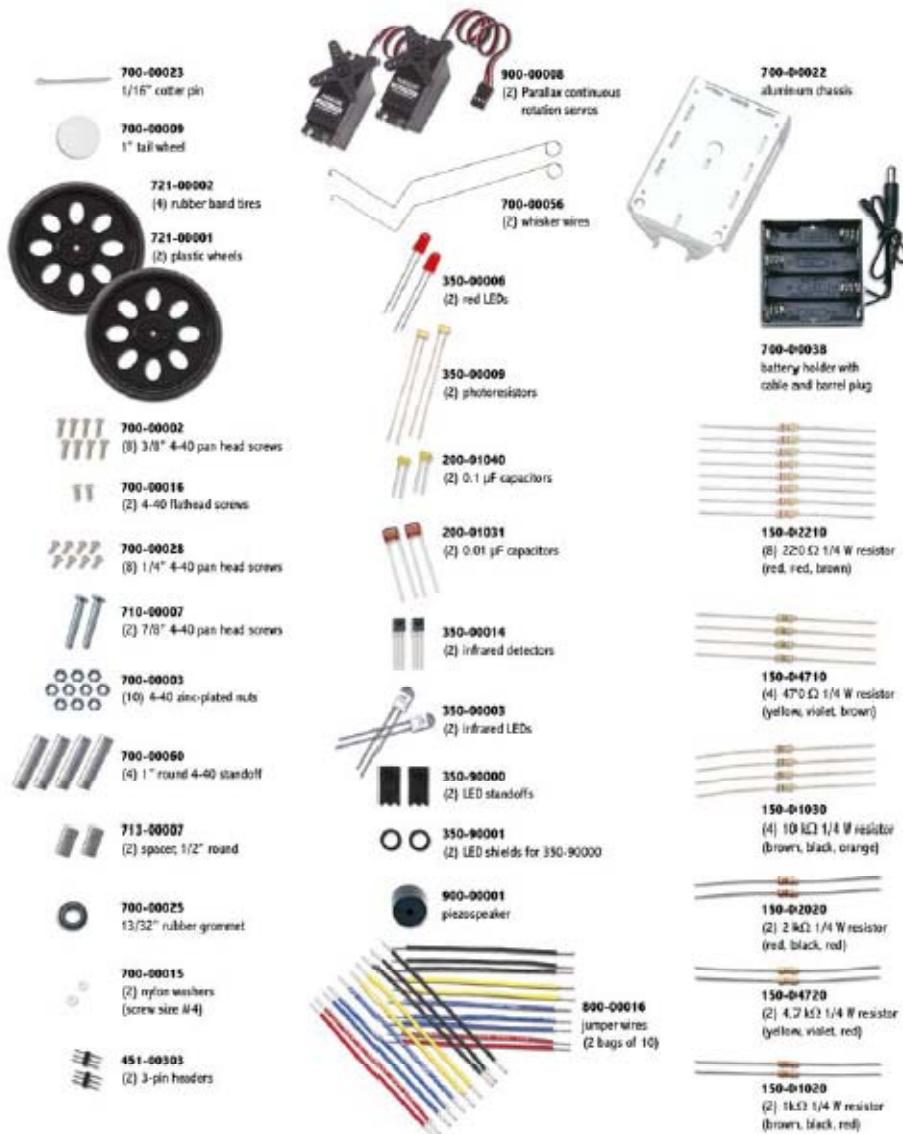
Slika 1

1.2 Potreban alat

Za sastavljanje Boe-Bota potreban je sljedeći alat:

- Sjelača klješta
- Ravna klješta
- Odvijač mali, križni

1.3 Popis mehaničkih djelova

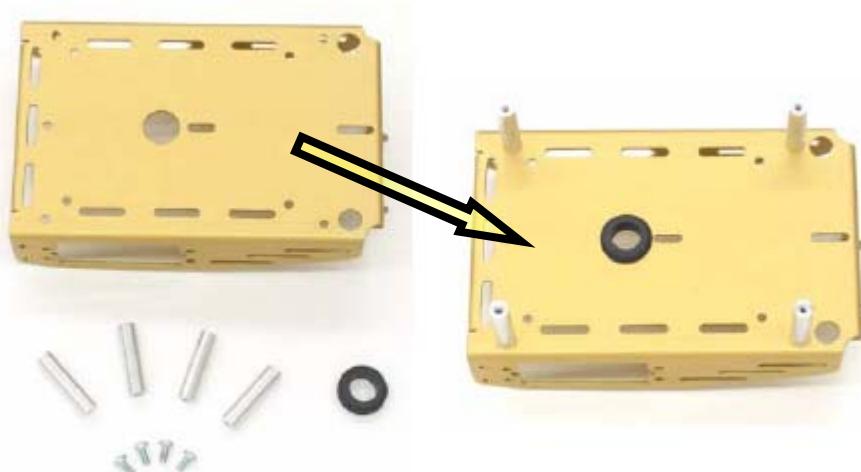


Slika 2

2.0 Sastavljanje mehanike

Sastavljanje mehanike se sastoji od nekoliko koraka. U svakom koraku opisane su potrebne komponente i način na koji se upotrebljavaju.

2.1 Postavljanje odstojnika i gumica



Slika 3

Potrebne komponente:

	Djelovi
	Kućište robota
	$\frac{1}{2}$ " Odstojnici
	$\frac{1}{4}$ " 4-40 vijci
	9/32" gumeni prsteni
	13/32" gumeni prsten

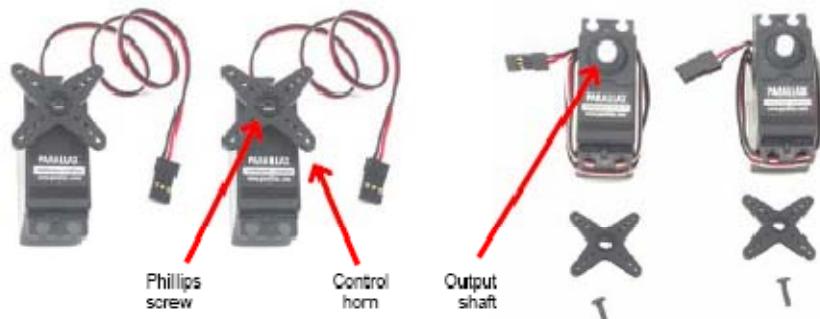
Sastavljanje:

- veći prsten se smješta u srednju rupu na kućištu
- manje prstenje se smješta u rupe na kutevima kućišta
- pomoću 4 vijka postavi odstojnike na strane kućišta

2.2 Pripremanje servo motora

Pripremanje servo motora izvodi se na sljedeći način:

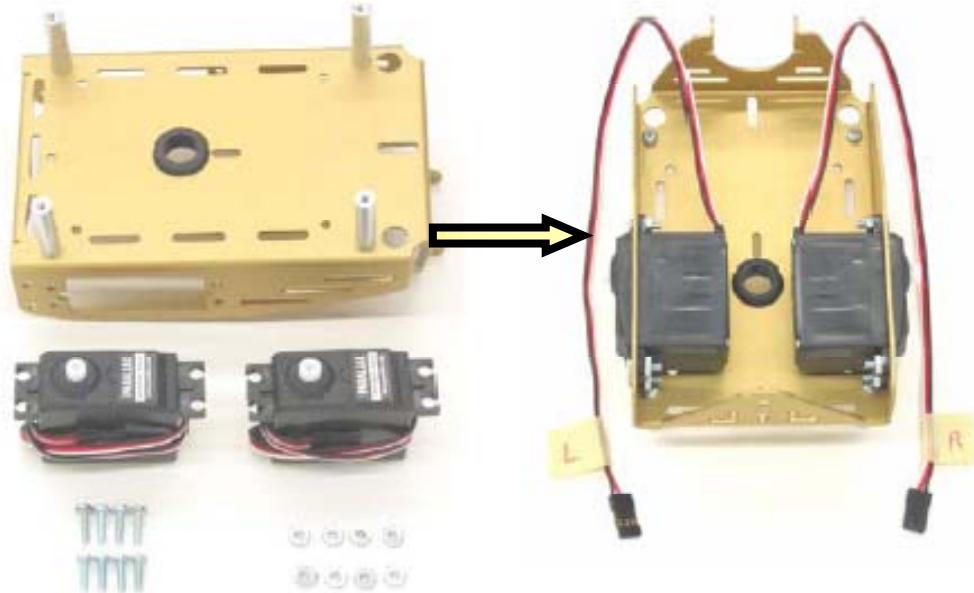
- skinij vijke sa oba servo motora
- potrebno je izvući „rog“ (prikazan je na slici na slici 4)
- vrati vijke na servo motore da se ne izgube



Slika 4

2.3 Postavljanje servo motora na kučište robota

Na slici 5 su prikazani djelovi koji su potrebni i kako bi sklopljeni dio trebao izgledati.



Slika 5

Potrebne komponente:

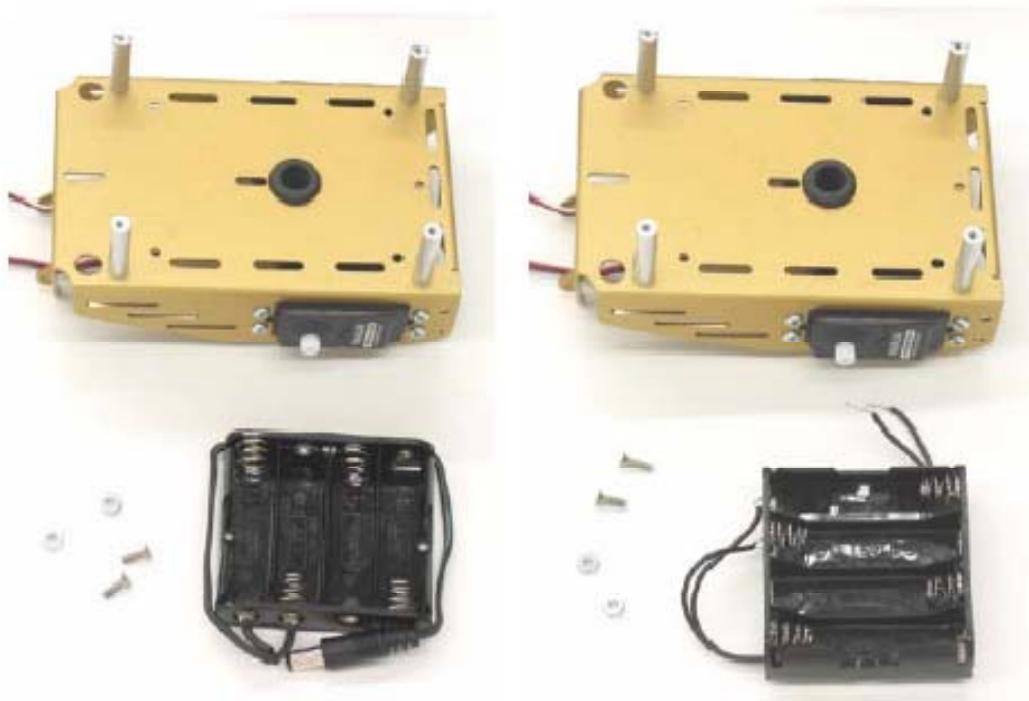
		Opis
		Kučište robota
		Parallaxovi servo motori
		3/8" 4-40 vijci
		4-40 matice

Sastavljanje:

-pomoću vijaka i matica pričvrsti servo motore na kučište, matice trebaju biti na unutarnjoj strani, a servo motori pozicionirani tako da im je kabel okrenut prema sredini kučišta

2.4 Postavljanje nosača baterija na kućište

Na slici 7 prikazani su potrebni djelovi.



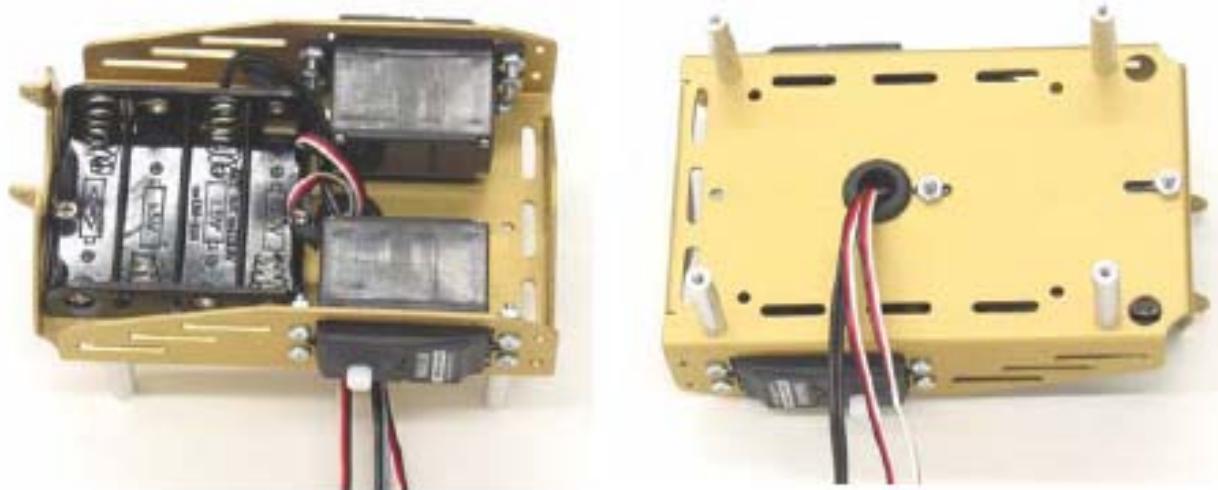
Slika 7

Potrebne komponente:

	Djelovi
	Kućište robota sa dodacima a),b),c)
	Kućište za baterije
	4-40 ravni vijci
	4-40 matice

Sastavljanje:

- pomoću ravnih vijaka i matica pričvrsti kućište za baterije na kućište robota, matice dolaze s gornje strane kućišta
 - povuci kabel od baterija kroz prsten na sredini kućišta
 - povuci kabel od servo motora kroz isti prsten
- Slika 8 prikazuje kako bi sklopljeni dio trebao izgledati.



Slika 8

2.5 Postavljanje kotača

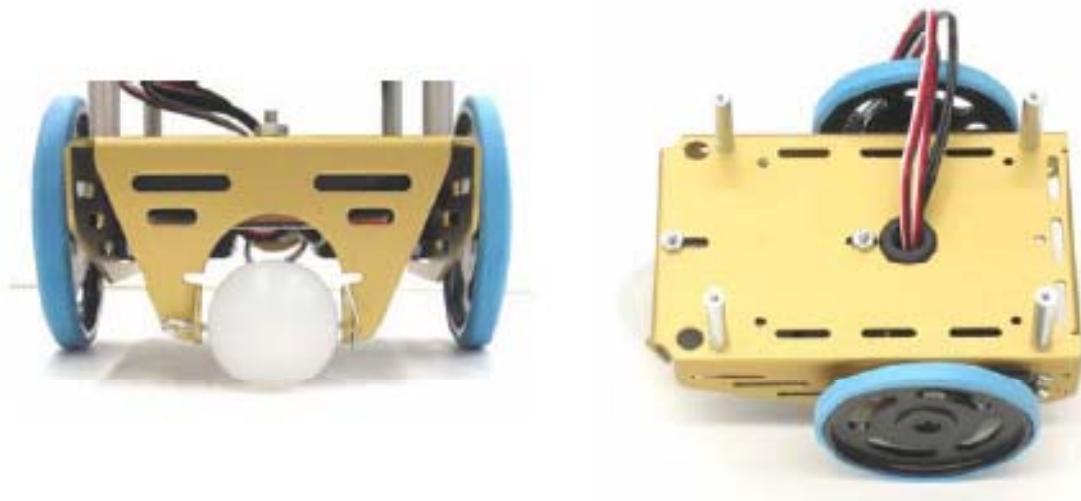


Slika 12

	Djelovi
	Kućište robota
	Plastični kotači
	Plastična kuglica
	Gumene trake za kotače
	Igla za učršćivanje

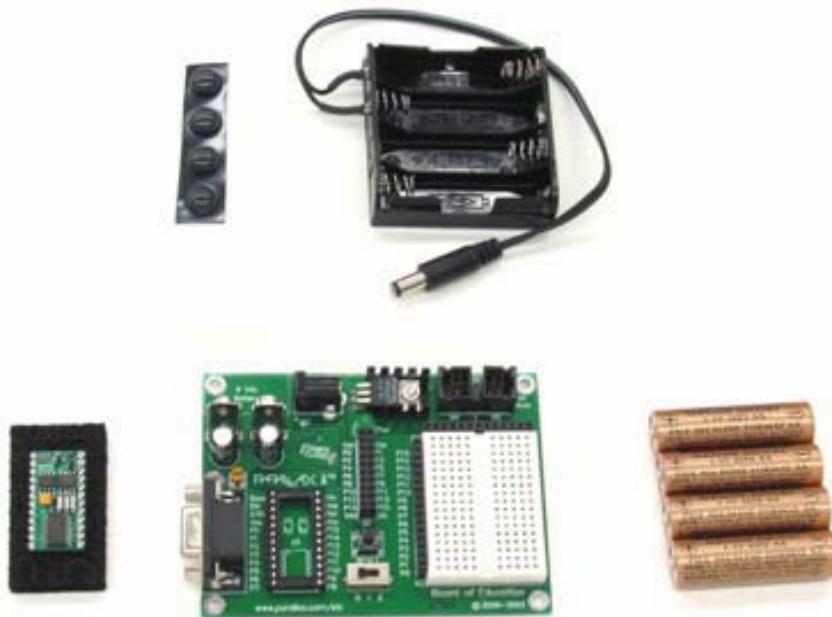
Sastavljanje:

-plastična kuglica koristi se kao stražnji pomoćni kotač; učvršćuje se povlačenjem igle kroz rupa na stražnju djelu kućišta i kuglicu, te savijanjem na kraju kako ne bi ispala
-gumene trake za kotače se navlače preko kotača



3.0 Sastavljanje elektronike

3.1 Sastavljanje upravljačke jedinice „Board of education“



Sastavljanje upravljačke jedinice robota:

Bs2 (C) nalazi se u posebnoj vrećici te ga je potrebno izvaditi iz vrećice.

Potrebno je izvaditi Bs2(C) iz crne spužve te ga umetnuti u pločicu (B), pritom treba paziti da MALI ZELENI POLUKRUG MORA BITI PORED PINOVA **Sout** i **Vin**, također ČIP OZNAKE **PIC16C57C** MORA BITI PORED PINOVA **P7** i **P8**.

Na slici 10 prikazana je glavna pločica na kojoj se nalazi basic stamp





3.2 Postavljanje pločice „Board of education“ na robota

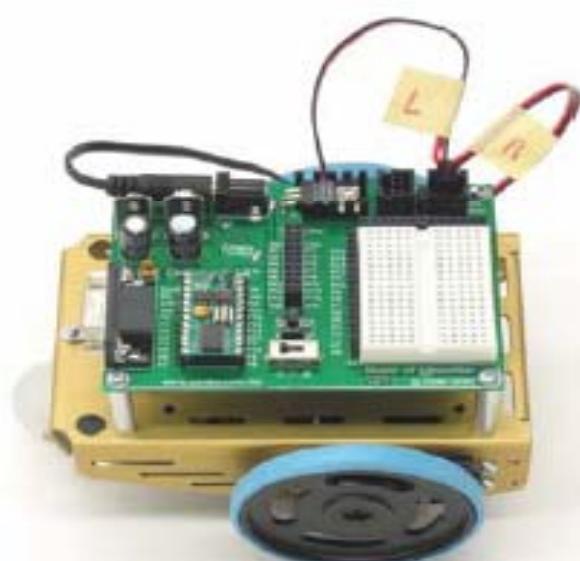
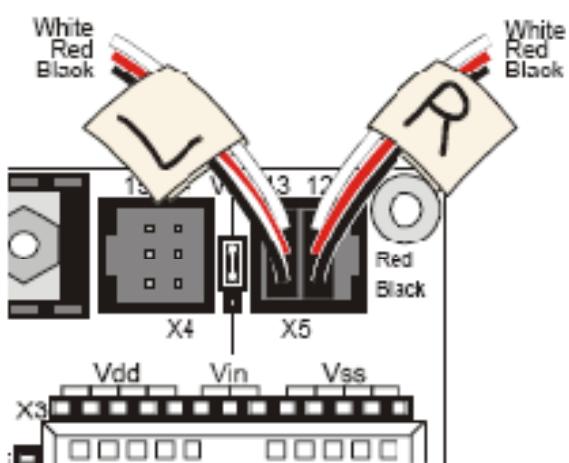


		Djelovi
		Kućište robota
		Pločica BOE sa BS2
		¼" 4-40 vijci

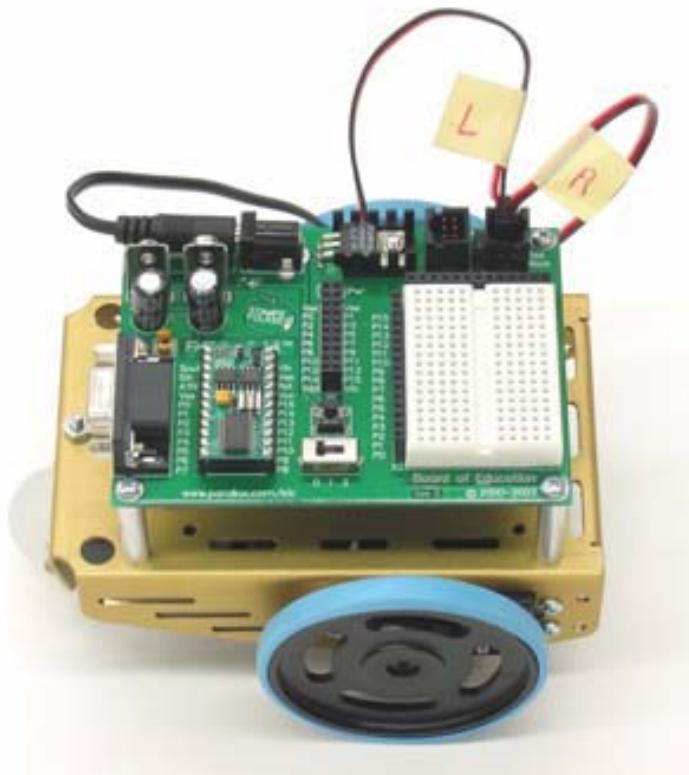
Sastavljanje:

- postavi pločicu na odstojnike na kućištu ,bjela pločica mora biti na strani gdje se nalaze servo motori
- pomoču vijaka učvrsti pločicu
- priključi servo motore na pločicu ,i to oba na konektore X5;potrebno je paiti na polaritet konektora kako je prikazano.

Slika koja prikazuje kako bi trebalo spojiti servo motore



3.3 Priprema za rad

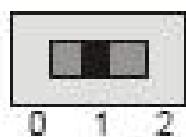


Slika 12

		Djelovi
		Robot
		1.5V AA baterije
		Serijski kabel

Sastavljanje:

- baterije je potrebno staviti u kućište za baterije na robotu, pritom pazite na polaritet
- utaknuti konektor od baterija u konektor na pločici robota
- ako zelena lampica ne svjetli, izvucite konektor i provjerite kabel
-



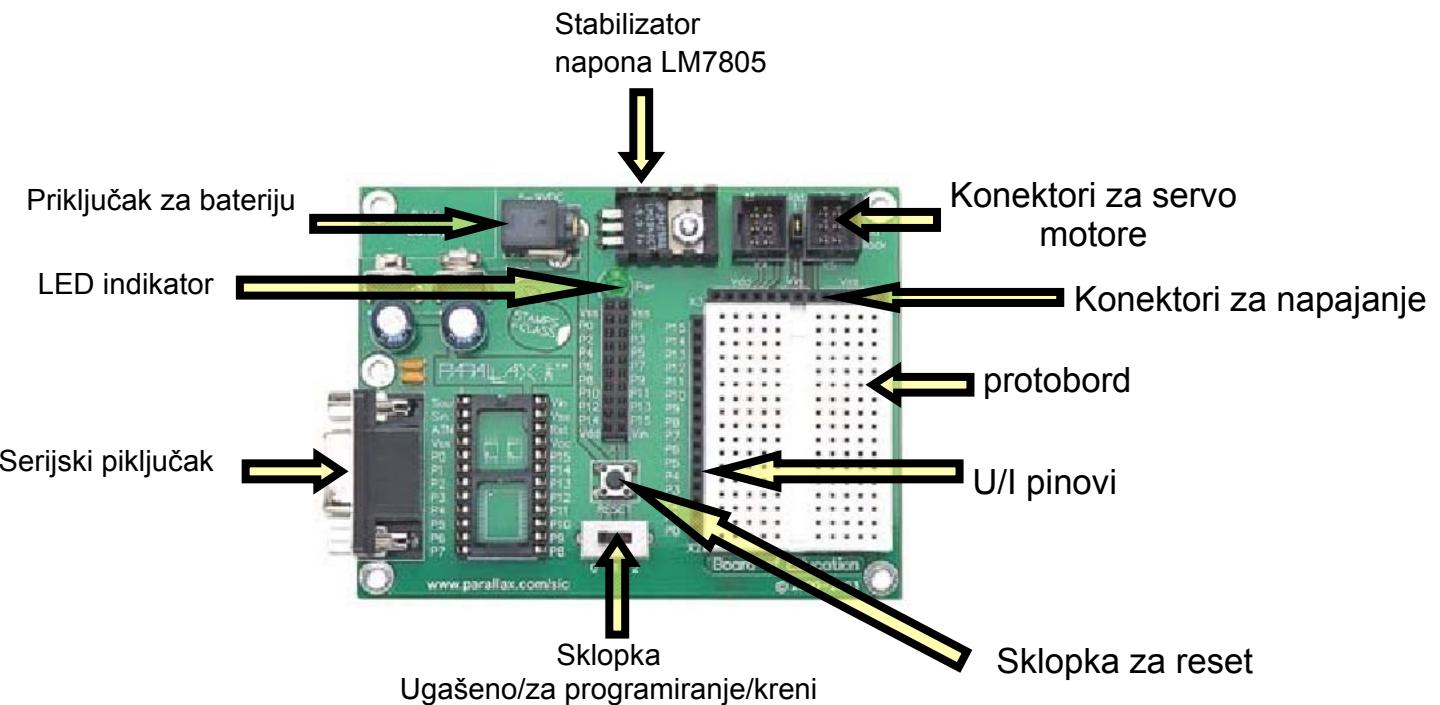
- 0-robot je ugašen
- 1-robot je uključen i spreman za programiranje
- 2-robot je uključen i operativan.

3.4 Opis upravljačke jedinice robota

BASIC Stamp 2 Modul

Upravljačku jedinicu robota proizvela je američka kompanija Parallax.

Upravljačka jedinica sastoji se od integriranog sklopa i glavne elektroničke pločice na kojoj su postavljeni konektori za spajanje senzora i ostalih elemenata na robota.



Integrirani sklop-basic stamp II

sastoji se od elektroničke pločice koja ima 24 pina. Na pločici postavljeni su sklopovi za memoriju, za komunikaciju računala sa izvršnim čipom, stabilizator napona koji osigurava stalan i konstantan napon na izvršnom čipu, sklop za resetiranje

PBASIC2 Interpreter

-Čini ga PIC 16C57 mikrokontroler. Mikrokontroler je trajno programiran sa PBASIC 2 instrukcijskim setom koje mu omogućavanju izvršavanje PBASIC2 instrukcija. Korisnički program se smješta u vanjski EEPROM, dok je mikrokontroler zadužen za čitanje tog programa iz EPROM-a, i njegovo izvršavanje. PIC 16C57 ima 20 ulazno/izlaznih pinova, od kojih se 16 koriste u korisničkim programima. Od preostalih 4, 2 se koriste za serijsku komunikaciju, a ostala dva za komunikaciju sa EPROM-om i ne mogu se koristiti za ništa drugo. Svaki pin je označen sa oznakama P0 do P16.

EEPROM

Korisnički program smješta se u EEPROM, veličine 2 kb. EEPROM je skraćenica od: electrically programmable read only memory, tj. električki programabilna memorija.

Sklop za resetiranje

Kako mikrokontroleri i memorije rade nepredvidljivo pri naponu ispod dozvoljenog, potrebno ih je dovesti u pravo stanje svaki put kada napon postigne dozvoljenu razinu. Sklop se sastoji od čipa S-8045HN i popratnih elemenata koji resetiraju mikrokontroler ako napon padne ispod 4V.

Napajanje

Basic Stamp napaja se naponom od 5V do 15V. Stabilizator napona LM7405 daje na izlazu potrebnih +5V za napajanje PIC-a i EEPROM-a.

Sklop za serijsku komunikaciju

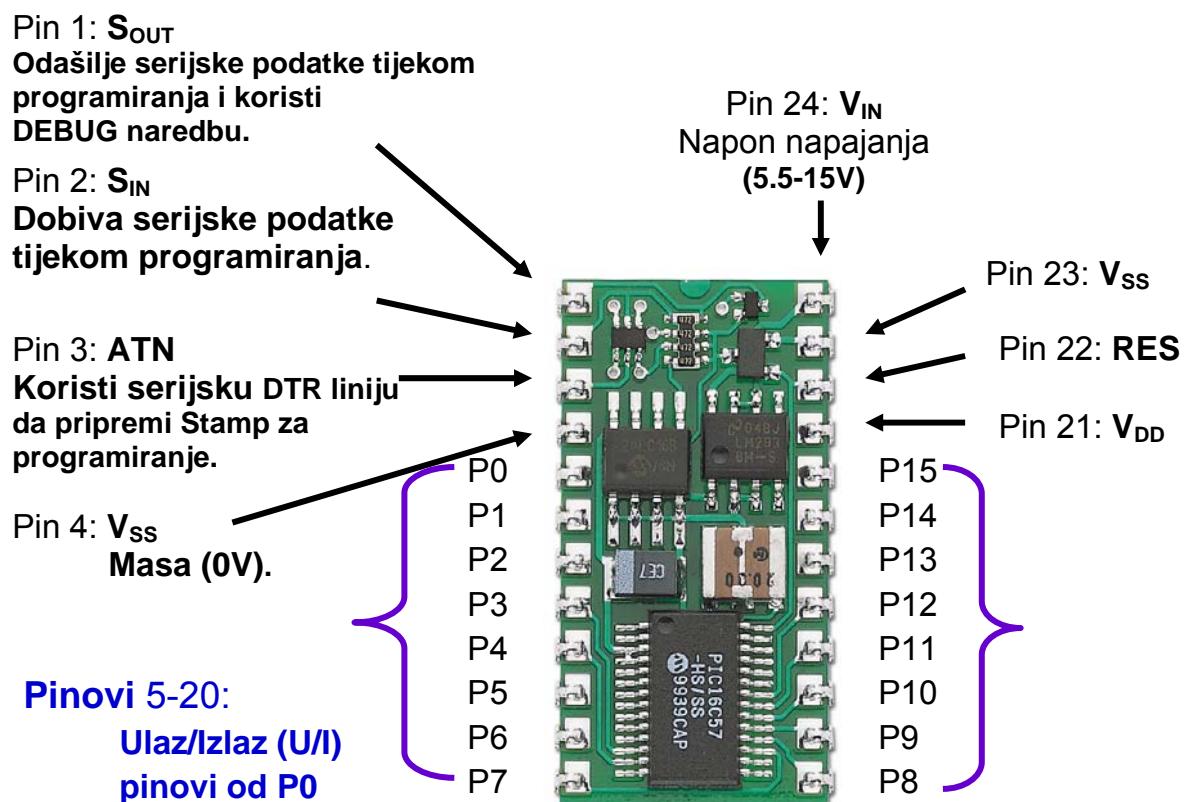
RS-232 koristi dva napona za označavanje logičke 0 i 1: +12V za logičku 0 i -12V za logičku 1. PIC za serijsku komunikaciju koristi TTL/CMOS nivoe, +5V za logičku 0 i 0V za logičku 1, te ovaj sklop vrši u pretvorbu signala iz CMOS u TTL nivo.

Povezivanje sa računalom

- Sout –serijski izlaz mikrokontrolera
- Sin-serijski ulaz mikrokontrolera
- Atn-ulaz kojime računalo priprema mikrokontroler za programiranje (Memory enable-omogući upis podataka u memoriju).
- Vss-masa

Povezivanje sa računalom

Robot se sa računalom povezuje preko 9 pinskog serijskog ili USB kabela.

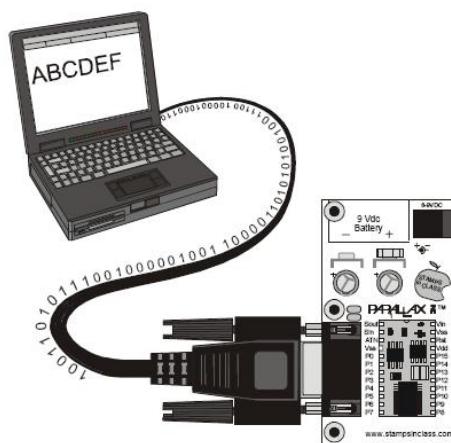


4.0 Programiranje robota

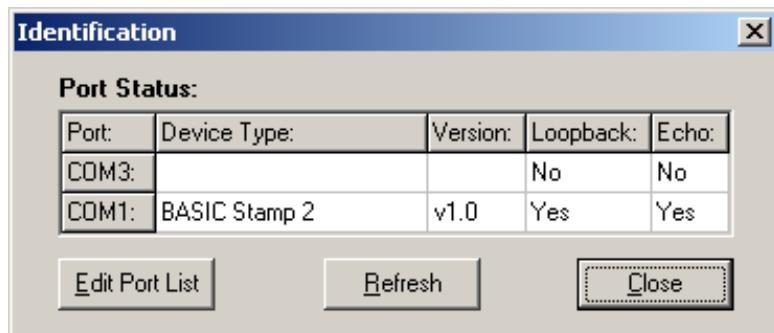
4.1 Pisanje programskog koda

Program se pokreće dvostrukim klikom na stampw.exe ikonu. Program se upisuje u bijeli prozor (E), a pohranjuje se pritiskom na ikonu diskete (F). Prije pisanja programa treba definirati vrstu čipa (B) i verziju PBASIC jezika (C). Napisani program pokreće se pritiskom na mali plavi trokutić (RUN-D).

Prije nego što počnemo programirati robota trebamo isprobati da li robot komunicira sa računalom. Robota treba spojiti s računalom preko RS 232 kabela i u programu testirati komunikaciju računala i robota-RUN/IDENTIFY.



Slika 13 prikazuje ispravno stanje komunikacije računala sa Bs2 čipom



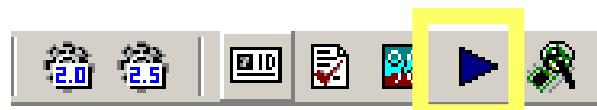
slika 13

Definiranje vrste čipa i Pbasic jezika vrši se odabirom ikone čipa i vrste jezika prema slici 14



Slika 14

- Nakon što ste napisali program pokrenite ga tipkom RUN



Nakon odabira čipa i pbasic jezika možemo započeti pisati program.

U prozoru za programiranje pojavit će se dvije početne naredbe bez kojih program neće raditi.

**'{\$STAMP BS2}' - vrsta čipa
'{\$PBASIC 2.5}' -vrsta Pbasic jezika**

!! Kod krivo unesene naredbe program će nas upozoriti sa prozorom u kojem će pisati tip pogriješke!!



4.2 Varijable za kontrolu pinova

4.3 Definiranje varijabli

4.4 Ispisivanje vrijednosti varijabli

4.5 Konstante

4.6 Brojevi i matematika

4.7 Programske petlje

4.8 Grananje programa

4.9 Naredba „DEBUG“

- koristi se za slanje podataka serijskom vezom iz Basic Stampa u računalo, kada u programu pozovemo DEBUG, program otvara DEBUG prozor i podaci sa upravljačke jedinice roobta šalju se natrag na računalo, te se ispisuju na ekranu.
- pritiskom prekidača RESET dolazi do ponovnog prikazivanja podataka u DEBUG prozoru.

Primjer 1 govori o ..

Primjer 1

```
' {$STAMP BS2}
' {$PBASIC 2.5}

DEBUG 66,65,83,73,67,32,83,116,97,109,112,32,50

END
```

Uz DEBUG možemo dodati i naredbu DEC koja označava decimalni sustav.

Primjer 2

```
' {$STAMP BS2}
' {$PBASIC 2.5}

DEBUG DEC 7 * 11

END
```

Ako pored DEBUG prozora ne napišemo DEC, DEBUG će broj 77 prepoznavati kao ASCII kod.

Primjer 3

```
' {$STAMP BS2}  
' {$PBASIC 2.5}
```

```
DEBUG 7 * 11
```

```
END
```

- još neke naredbe za DEBUG.

CR - započivanje novog reda

HOME - započinje tekst od početka

CLS - briše DEBUG prozor

END - koristi se kada želimo završiti program

Primjer 4

```
' {$STAMP BS2}  
' {$PBASIC 2.5}
```

```
DEBUG "pozdrav ovo je poruka sa Vašeg Boe-Bot-a", CR
```

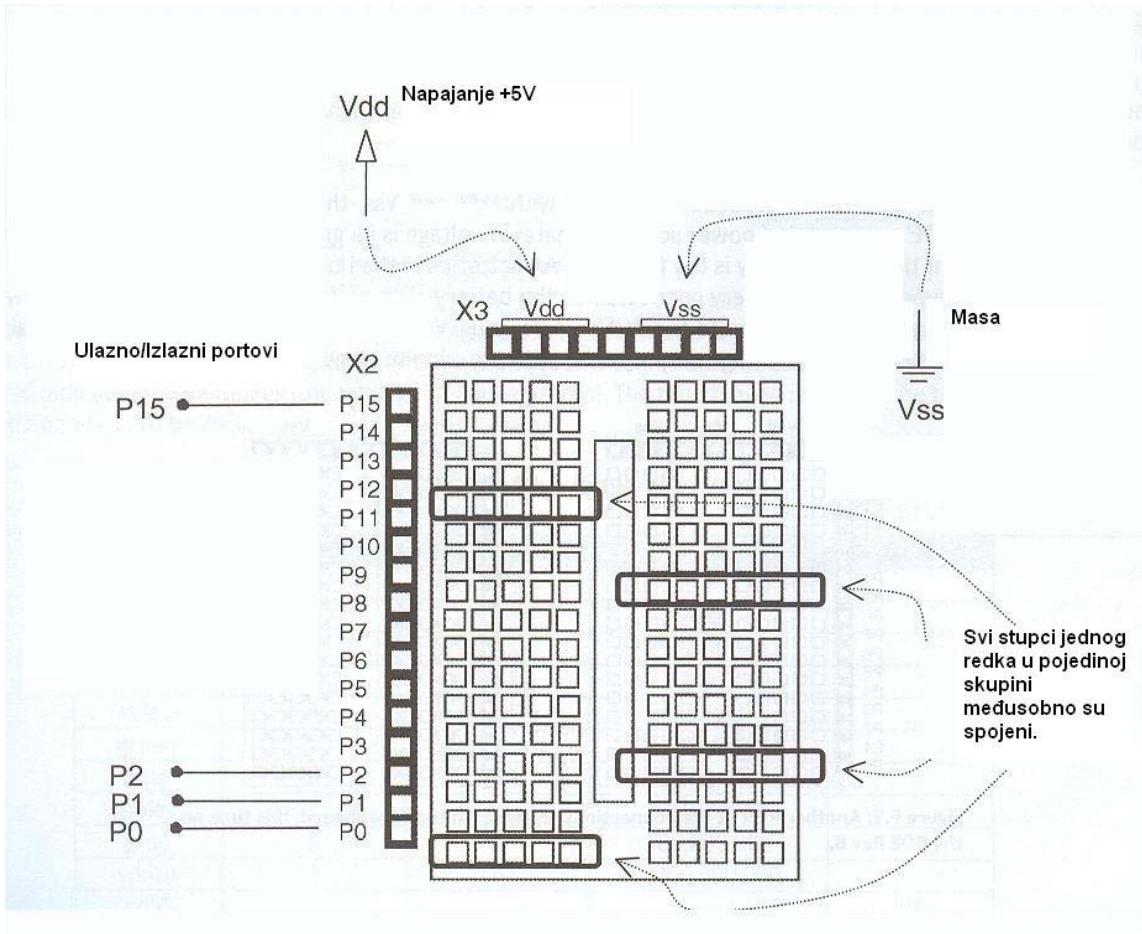
```
DEBUG "Koliko je 7 X 11?", CR, "Odgovor je: ", DEC 7 * 11
```

```
END
```

4.3 Sastavljanje i programiranje svijetlećih dioda

4.3.1 Kako spojiti LED na robota?

Postavljanje elektronike vrši se pomoću prototipne pločice (protoboard), tako da se el. komponente mogu jednostavno izmjenjivati. Protoboard je prikazan na slici 16.

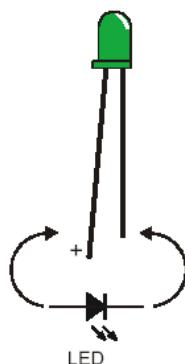


slika 16

Protoboard pločica sastoji se od dvije skupine rupa. Svaka skupina sastoji se od 17 redova i 5 stupaca. Svi stupci jednog redka u pojedinoj skupini su međusobno spojeni, kao što se vidi na slici. Shema se spaja tako da se nožice komponenata umetnu u pojedine rupe. Sve nožice koje se nalaze u istom redu, u istoj skupini bit će međusobno spojene. Pinovi P0 do P15 spojeni su na BS2 i mogu se koristiti kao ulazno/izlazni portovi. Portovi označeni oznakom Vdd spojeni su na napon od +5V i koriste se za napajanje senzora i ostalih elemenata, portovi označeni oznakom Vin spojeni su na napon napajanja (napon baterije +6V ili +9V). Portovi označeni oznakom Vss spojeni su na masu (0V).

Svjetleće diode su one diode koje emitiraju svjetlost kada su propusno polarizirane. Za izradu svjetlećih dioda primjenjuju se spojevi galija sa arsenom i fosforom. Ovisno o sastavu, svjetleće diode mogu dati infracrveno, crveno, žuto, zeleno ili palvo svjetlo. Intenzitet svjetlosti ovisi o jakosti struje koja prolazi kroz diodu i smjeru iz kojeg se dioda promatra. Svjetleće diode upotrebljevaju se kao signalni i kontrolni elementi te kao izvori svjetlosti u različitim uređajima. Osnovni spoj svjetleće diode prikazan je na slici 17 gdje otpornik R sprječava porast struje kroz diodu iznad dopuštene vrijednosti. U usporedbi sa žaruljicom svjetleća dioda ima znatno bolja svojstva. (dulji vijek trajanja, mogućnost rada sa malim naponima, mali utrošak snage.) Nedostatak im je mali dopušteni rezervni napon, od svega nekoliko volta.

Svjetleća dioda spaja se u strujni krug tako da se anoda spaja na (+) pol napajanja, a katoda se spaja na (-) pol napajanja. Ako polove napajanja spojimo suprotno dioda će biti nepropusno polarizirana što znači da dioda neće svjetliti. Svjetleće diode izvode se tako da anoda ima dulju nožicu, a katoda kraću nožicu.

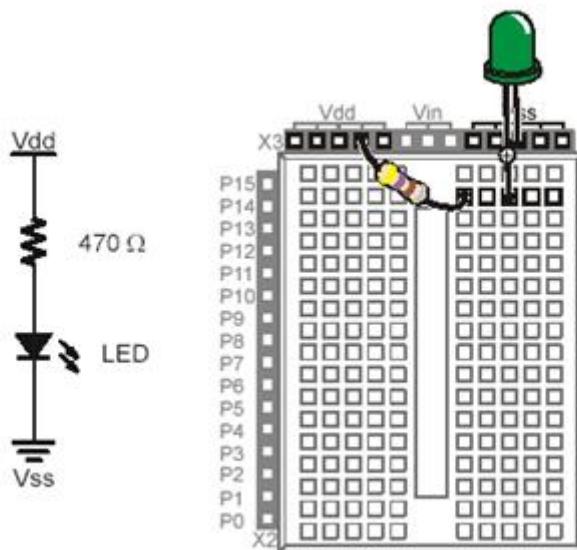
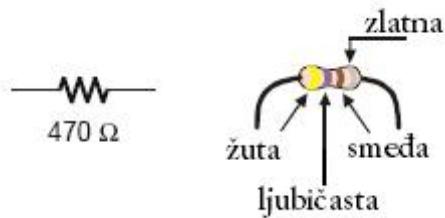


Slika prikazuje izgled LED diode i njezin shematski prikaz.

Primjer 1

na protobord potrebno je spojiti shemu prema slici 17, te na otporniku i diodi voltmetrom izmjeriti napone.

Naziv komponente	Količina
LED	1 kom.
Otpornik 470 ohm (žuta, ljubičasta, smeđa, zlatna)	1 kom.

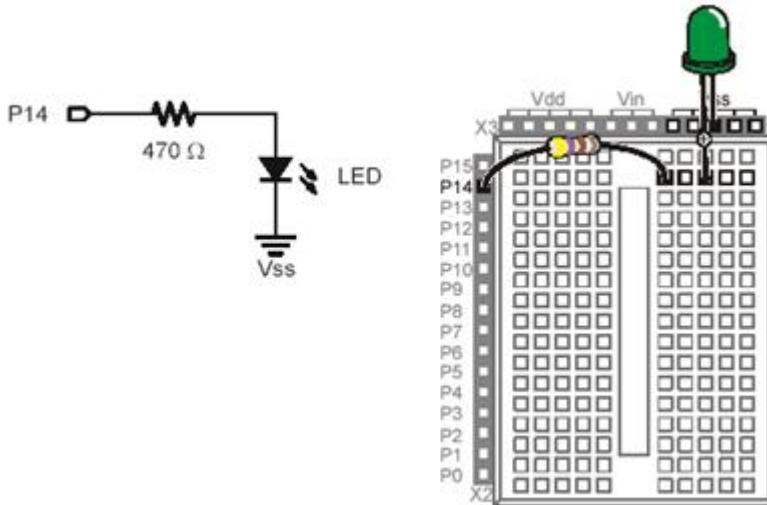


Oprez: treba pripaziti da je LED pravilno polarizirana

Zaključak

Primjer 2

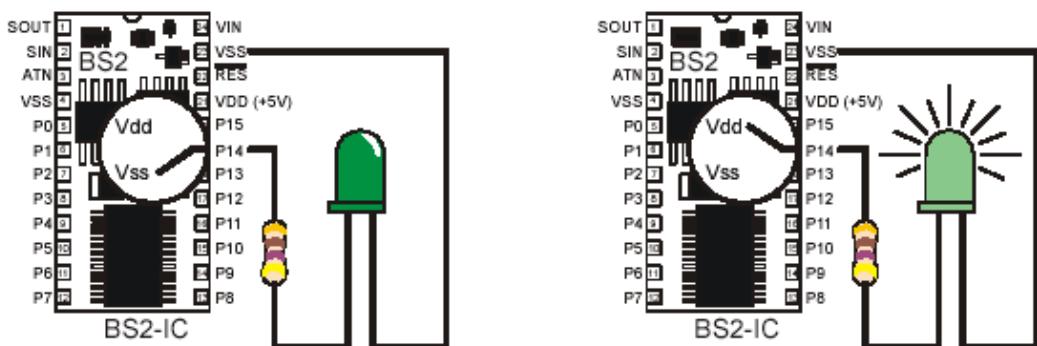
Upravljanje radom svjetleće diode vrši se tako da se otpornik i dioda spoje prema shemi sa



4.3.2 Programiranje LED

U ovom slučaju (slika...) svjetleća dioda spojena je na mikroprocesor (BS2) koji upravlja sa njezinim radom.

Basic Stamp ponaša se kao sklopka i strujni krug spaja na Vss potencijal (0V) ili na Vdd potencijal (+5V).



Ovako priključena svjetleća dioda na Basic Stamp, omogućava početak programiranja rada svjetleće diode.

Primjer 3

```
' {$STAMP BS2}  
' {$PBASIC 2.5}
```

DO

```
HIGH 13  
PAUSE 500  
LOW 13  
PAUSE 500
```

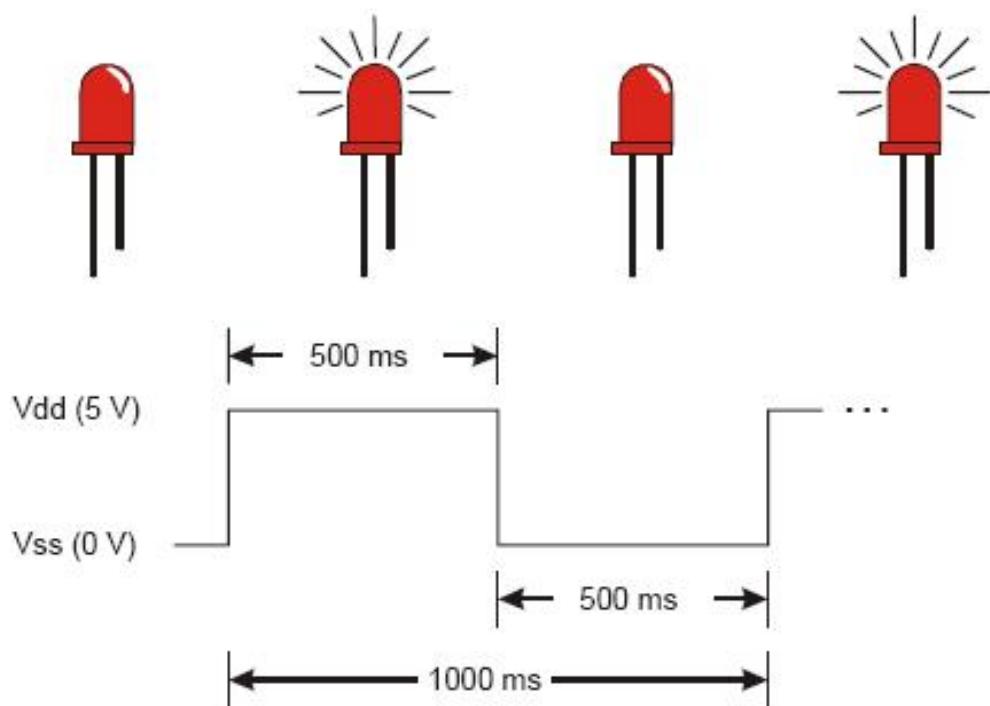
LOOP

-kada smo upisali program, robot se spaja s računalom i upisuje se program u robota pomoću naredbe RUN.

-Uključivanjem prekidača dioda će svijetliti u zadanom intervalu.

Program koji se nalazi u memoriji Basic Stampa izvršava instrukcije tako da započinje od početka, a završetak je uvjetovan zadanim petljom.

U primjeru programa koristi se DO-LOOP petlja koja nam omogućuje beskonačno izvršavanje programa.



Što kada želimo da LED blinka 10 puta?

U našem programu koristili smo DO-LOOP WHILE petlju koja nam omogućuje uvjetovano izvršavanje programa. (ako je brojač ≤ 10 LOOP petlja se završava.)

Primjer 4

```
' {$STAMP BS2}
' {$PBASIC 2.5}

Counter VAR Byte
Counter = 1
DO
    HIGH 13
    PAUSE 500
    LOW 13
    PAUSE 500
Counter = Counter + 1
LOOP WHILE (Counter <= 10)
END
```

- kada smo upisali program, robot se spaja s računalom i upisuje se program u robota pomoću naredbe RUN.
- Uključivanjem prekidača dioda će svijetliti u zadanim intervalima 10 puta.

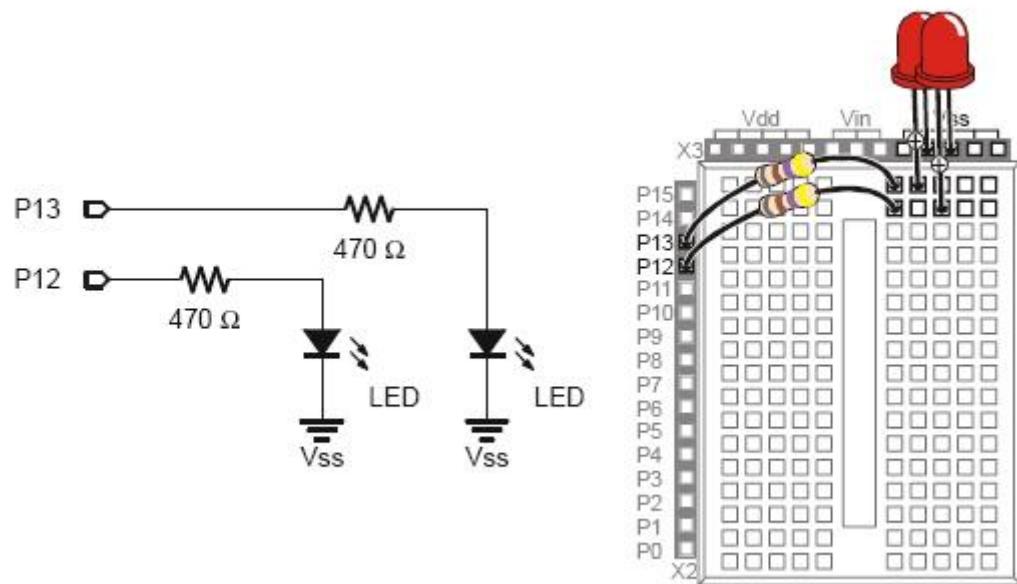
Naredba PULSOUT

Primjer 5

```
' {$STAMP BS2}
' {$PBASIC 2.5}

DO
PULSOUT 13, 65000
PAUSE 2000
LOOP
```

Zadatak 1:



4.4 Upravljanje robotom

Robot „Boe-Bot“ koristi servo motore koji su preuređeni tako da se mogu kontinuirano rotirati. Servo motor ima tri žice koje se priključuju na robota i sastoji se od više cjelina. Prva cjelina servo motora je istosmjerni motor-reduktor, prekidač krajnjeg položaja, potenciometar za pozicionu povratnu vezu i integrirani elektronički sklop za upravljanje. Od tri žice koje izlaze iz kućišta motora, jedna je energetska (+5V do +9V), jedna za uzemljenje, a jedna je upravljački ulaz. Signal kojim se upravlja sa motorom mora biti moduliran pulsno širinskom modulacijom jer na temelju širine impulsa motor zauzima željenu poziciju. Kada govorimo o servo pozicioniranju, tada podrazumjevamo da elektronički sklop usmjerava motor na poziciju i zadržava ga u njoj. Ako zadržimo osovinu motora dok servo petlja radi i prisilno pokušamo odvući u drugi položaj, elektronički sklop očitati će kut potenciometra utvrđujući da osovina nije u traženoj poziciji i povećava struju na motoru jer ona će povećati moment motora i motor će se vraćati nasuprot momenatu koji smo izveli rukom. Servo motor nastaviti će sa djelovanjem petlje sve dok se osovina ne vrati u zadnu poziciju. Servo motori su izuzetno kompaktni i laki za upravljanje što ih čini veoma primjenjivim u robotskim konstrukcijama.

4.4.1 Spajanje i kalibracija motora

Za pokretanje robota koriste se servo motori. Motori su postavljeni tako da robot ima diferencijalan pogon.

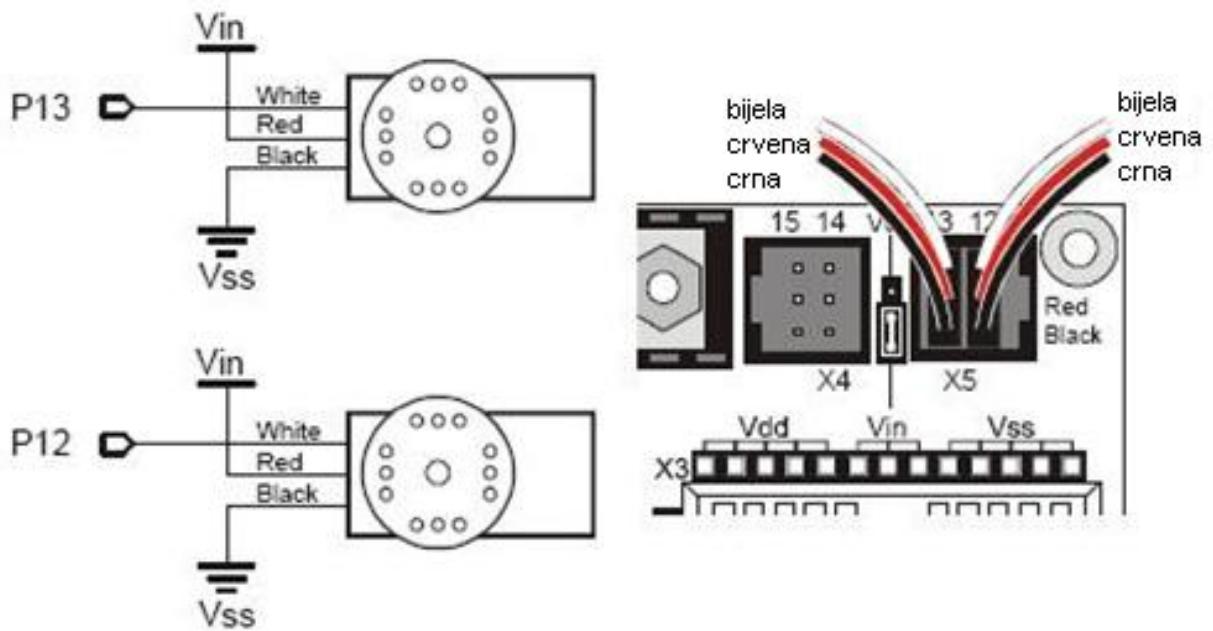
Motot ima tri žice, svaka žica ima svoju ulogu u pokretanju i kontroli motora.



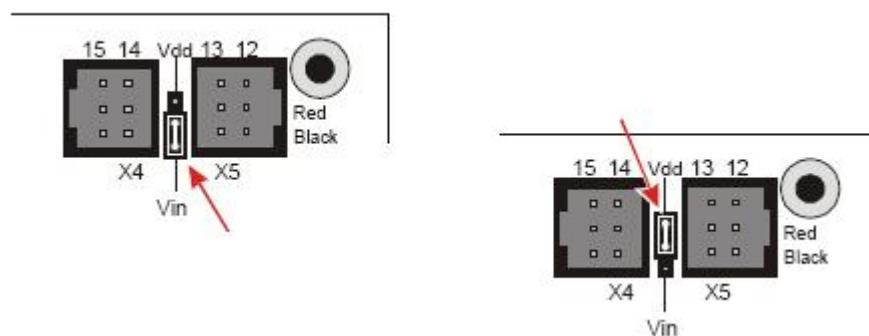
- **bijela žica** - ta žica ima ulogu za prijenos digitalnog signala.

- **crvena žica** - ta žica služi za napajanje motora(+5V)

- **crna žica** - masa(-).



Prebacivanje konektora!!!!!!!!!!!!!!



Formula $750 \times 2 \mu\text{s} = 1500 \mu\text{s}$ or 1.5mS

$$\text{Duration argument} = \frac{\text{Pulse duration}}{2\mu\text{s}} = \frac{0.0015\text{s}}{0.000002\text{s}} = 750$$

Osnovna

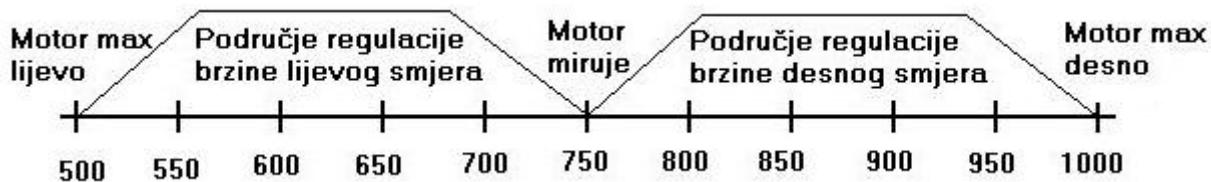
Primjer 1

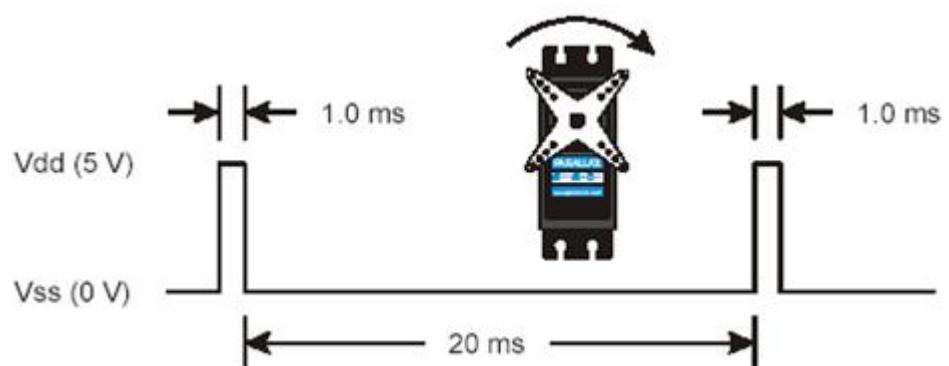
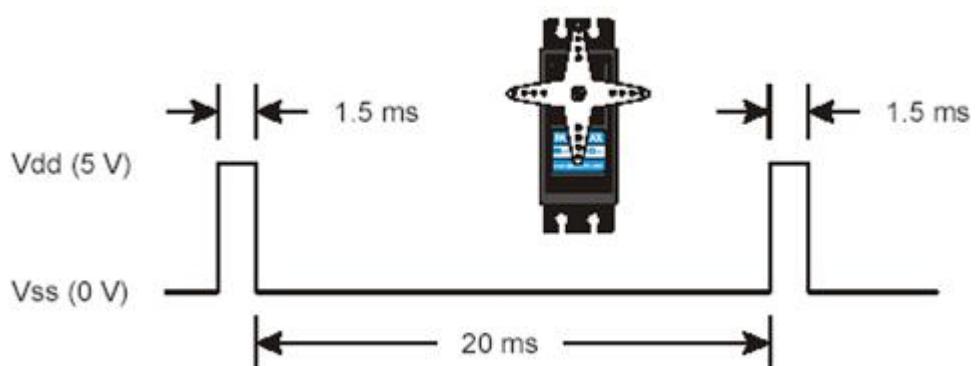
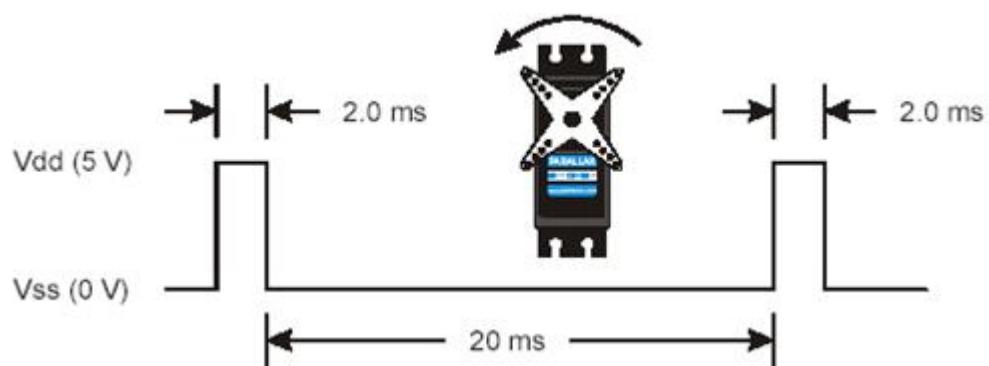
```
' {$STAMP BS2}
' {$PBASIC 2.5}

DO
PULSOUT 12, 750
PAUSE 20
LOOP
```

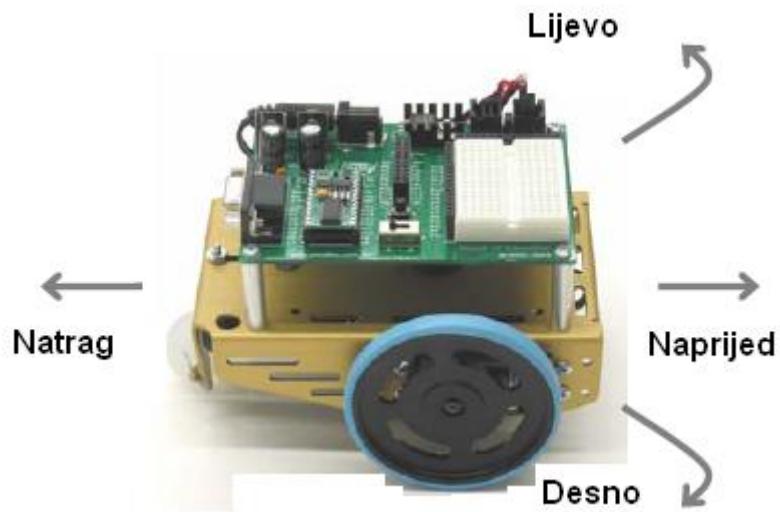
Druga slika

-Za pokretanje motora potrebno je svakih 20ms generirati impuls na upravljačkom pinu motora. Dužina tog impulsa određuje brzinu vrtnje i smjer motora. Pri impulsima u trajanju od 1.5 ms motor miruje.Za duže impulse 1.5-2 ms motor će se okretati u jednu stranu, a za kraće impulse 1-1.5 ms motor će se okretati u drugu stranu.





4.4.2 Programiranje motora



```
' {$STAMP BS2}  
' {$PBASIC 2.5}  
  
counter VAR Byte  
FOR counter = 1 TO 100  
PULSOUT 13, 850  
PAUSE 20  
NEXT  
  
FOR counter = 1 TO 200  
PULSOUT 12, 850  
PAUSE 20  
NEXT  
  
END
```

```

' {$STAMP BS2}
' {$PBASIC 2.5}

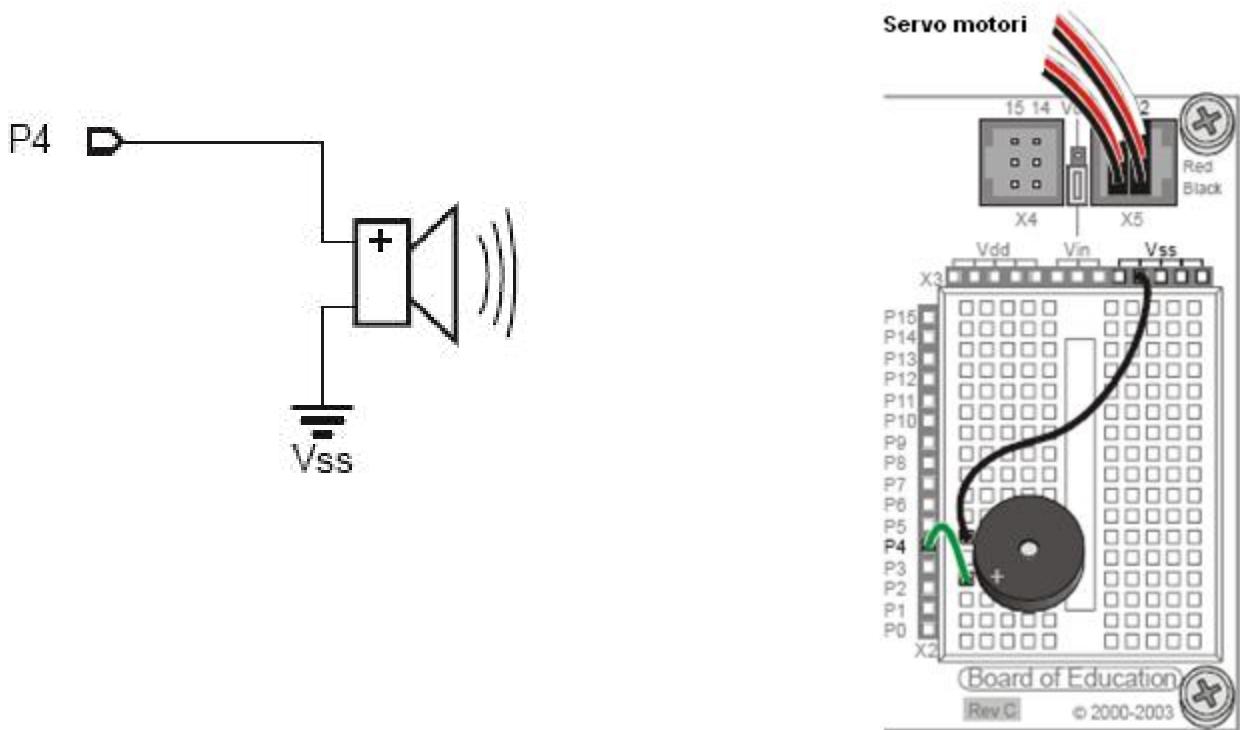
DEBUG "Program running!"
counter VAR Nib ' Triangle has 3 sides
pulseCount VAR Word ' Pulse count to servos
FREQOUT 4, 2000, 3000 ' Signal program start/reset.

Main:
FOR counter = 1 TO 3 ' Repeat 3 times for triangle
GOSUB Forward
GOSUB Right_Rotate120
NEXT
END
Forward:
FOR pulseCount = 1 TO 163 ' Forward 1 yard
PULSOUT 13, 850
PULSOUT 12, 650
PAUSE 20
NEXT
RETURN
Right_Rotate120:
FOR pulseCount = 1 TO 21 ' Rotate right 120 degrees
PULSOUT 13, 850
PULSOUT 12, 850
PAUSE 20
NEXT
RETURN

```

4.4.3 Dodavanje piezzo zvučnika

Na slici 14 je prikazana shema spoja. Također je prikazana i shema spajanja servo motora, iako njih nije potrebno spajati jer su već spojeni.



U ploču kvarca sa suprotnih strana pustimo struju, kvarc će početi vibrirati, a to zovemo "**Piezo učinak**". Upravo se taj učinak koristi u izradi mikrofona, tranzistora....

Programming the Start/Reset Indicator

The next example program tests the piezospeaker. It uses the **FREQOUT** command to send precisely timed high/low signals to a speaker. Here is the **FREQOUT** command's syntax:

FREQOUT Pin, Duration, Freq1 {,Freq2}

Here's an example of a **FREQOUT** command that's used in the next example program.

FREQOUT 4, 2000, 3000

The **Pin** argument is 4, meaning that the high/low signals will be sent to I/O pin P4. The **Duration** argument, which is how long the high/low signals will last, is 2000, which is 2000 ms or 2 seconds. The **Freq1** argument is the frequency of the high/low signals. In this example, the high/low signals will make a 3000 hertz, or 3 kHz, tone.

```

' {$STAMP BS2}           ' Stamp directive.
' {$PBASIC 2.5}          ' PBASIC directive.
DEBUG CLS, "Beep!!!"      ' Display while speaker beeps.
FREQOUT 4, 2000, 3000      ' Signal program start/reset.
DO ' DO...LOOP
DEBUG CR, "Waiting for reset..."   ' Display message
PAUSE 500                   ' every 0.5 seconds
LOOP                         ' until hardware reset.

```

```

' {$STAMP BS2}
' {$PBASIC 2.5}

counter VAR Word
pulseWidth VAR Word
pulseWidthComp VAR Word
FREQOUT 4, 2000, 3000          ' Signal program start/reset.
DO
DEBUG "Enter pulse width: "
DEBUGIN DEC pulseWidth
pulseWidthComp = 1500 - pulseWidth
FOR counter = 1 TO 244
PULSOUT 12, pulseWidth
PULSOUT 13, pulseWidthComp
PAUSE 20
NEXT
LOOP

```

```

' {$STAMP BS2}
' {$PBASIC 2.5}

DEBUG "Program Running!"
counter VAR Word
FREQOUT 4, 2000, 3000           ' Signal start of program.
FOR counter = 1 TO 122          ' Clockwise just under 3 seconds.
PULSOUT 12, 650
PAUSE 20
NEXT
FOR counter = 1 TO 40           ' Stop one second.
PULSOUT 12, 750
PAUSE 20
NEXT
FOR counter = 1 TO 122          ' Counterclockwise three seconds.
PULSOUT 12, 850
PAUSE 20
NEXT
FREQOUT 4, 500, 3500 ' Signal end of program
END

```

```

' {$STAMP BS2}
' {$PBASIC 2.5}

DEBUG "Program Running!"
counter VAR Word
FREQOUT 4, 2000, 3000 ' Signal program start/reset.
FOR counter = 1 TO 64 ' Forward
PULSOUT 13, 850
PULSOUT 12, 650
PAUSE 20
NEXT
PAUSE 200
FOR counter = 1 TO 24 ' Rotate left - about 1/4 turn
PULSOUT 13, 650
PULSOUT 12, 650
PAUSE 20
NEXT
PAUSE 200
FOR counter = 1 TO 24 ' Rotate right - about 1/4 turn
PULSOUT 13, 850
PULSOUT 12, 850
PAUSE 20
NEXT
PAUSE 200

```

```
FOR counter = 1 TO 64 ' Backward
PULSOUT 13, 650
PULSOUT 12, 850
PAUSE 20
NEXT
END
```

5.0 Senzori robota

Pri konstruiranju robota težimo da naš robot razumije i da je svijestan okoline, u stvarnosti robot je ograničen senzorima koje mu dodajemo i software-om koji za njega napišemo. Detektiranje nije opažanje. Senzori su samo pretvarači koji pretvaraju određene fizikalne pojave u električni signal koji mikroprocesor može čitati. Najčešće, za postizanje takve pretvorbe koriste se analogno-digitalni pretvarači koji se povezuju sa mikroprocesorom, čitanjem vrijednosti ulazno/izlaznog porta ili korištenjem vanjskog prekida. Software nam omogućuje stvaranje različitih razina abstrakcije, koje nam kao programerima pomažu da sagledamo podatke sa senzora na različite načine. Sustav mora imati na raspolaganju neke varijable kako bi mogao provjeriti: Je li u ovoj sobi tamno? Da li je upravo ušla osoba? Postoji li zid s leve strane? Međutim, jedina pitanja koja je robot u stanju postaviti su oblika: Da li se smanjio otpor fotootpornika? Da li je napon sa piroelektričkog senzora povezanog na četvrti kanal A/D pretvarača prešao zadani prag? Da li se izlaz infracrvenog detektora prepreke promjenio iz niskog u visoko stanje?

Osjetljivost i doseg

Pri analizi bilo kojeg senzora potrebno je razumjeti dva značajna pojma, a to su osjetljivost i doseg. Osjetljivost je mjera promjene izlaznog signala s promjenom mjerene veličine. Označimo izlaz senzora sa r , a mjerenu fizikalnu veličinu sa x . Osjetljivost senzora dana je izrazom:

Npr.

Fotodetektor može dati izlazni napon od 0,87 volti (r), kada u njega ulazi $2,3 \cdot 10^{13}$ fotona u sekundi (x).

$$\frac{\Delta r}{r} = S \frac{\Delta x}{x}$$

Senzor reagira na različite vrijednosti određenog fizikalnog podražaja davanjem određenog napona, struje ili frekvencije. Tipično, sklopovlje pridruženo senzoru pojačava ili na neki drugi način pretvara signal sa senzora i dovodi ga na analogno-digitalni pretvarač povezan na mikroprocesor. A/D pretvarač je osjetljiv na ograničeni raspon napona, često u rasponu od 0 do 5 V. Prilikom pretvaranja signala potrebno je pažljivo razmotriti kako se fizikalna veličina pretvara u digitalnu vrijednost dostupnu mikroprocesoru, postoje dvije mogućnosti a to su: linearan i logaritamski odnos između napona i brojeva koje daje A/D pretvarač.

)Sastavljanje taktilnih senzora

Potrebne komponente:

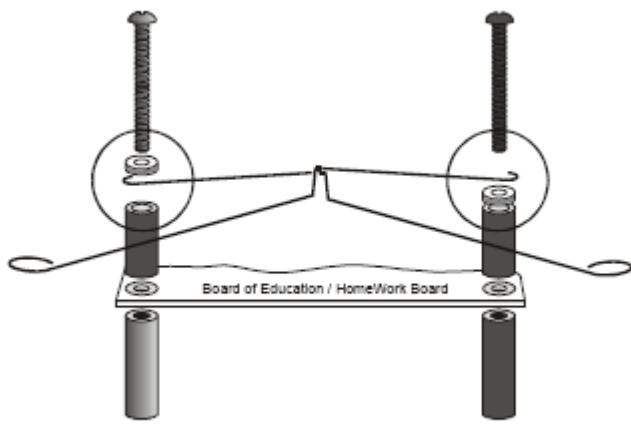
Potrebno	Djelovi
1	Sastavljeni robot
2	10K otpornici
2	3 pinski konektori
2	3/8" 4/40 odstojnici
2	½" 4/40 vijci
2	Ticala
2	Plastična pločica
4	Vodovi



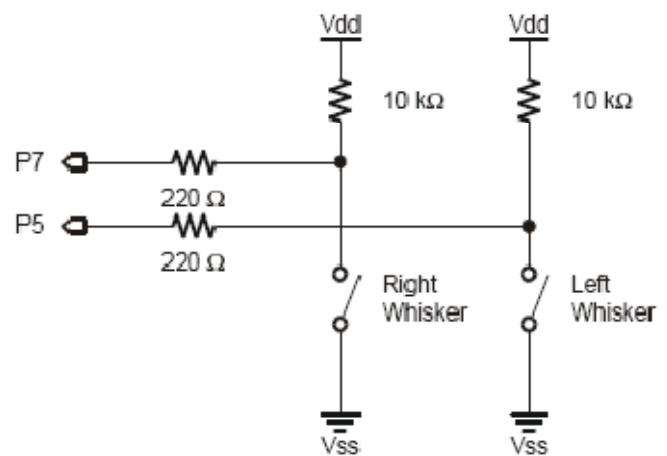
Slika 15

Prvo je potrebno postaviti ticala:

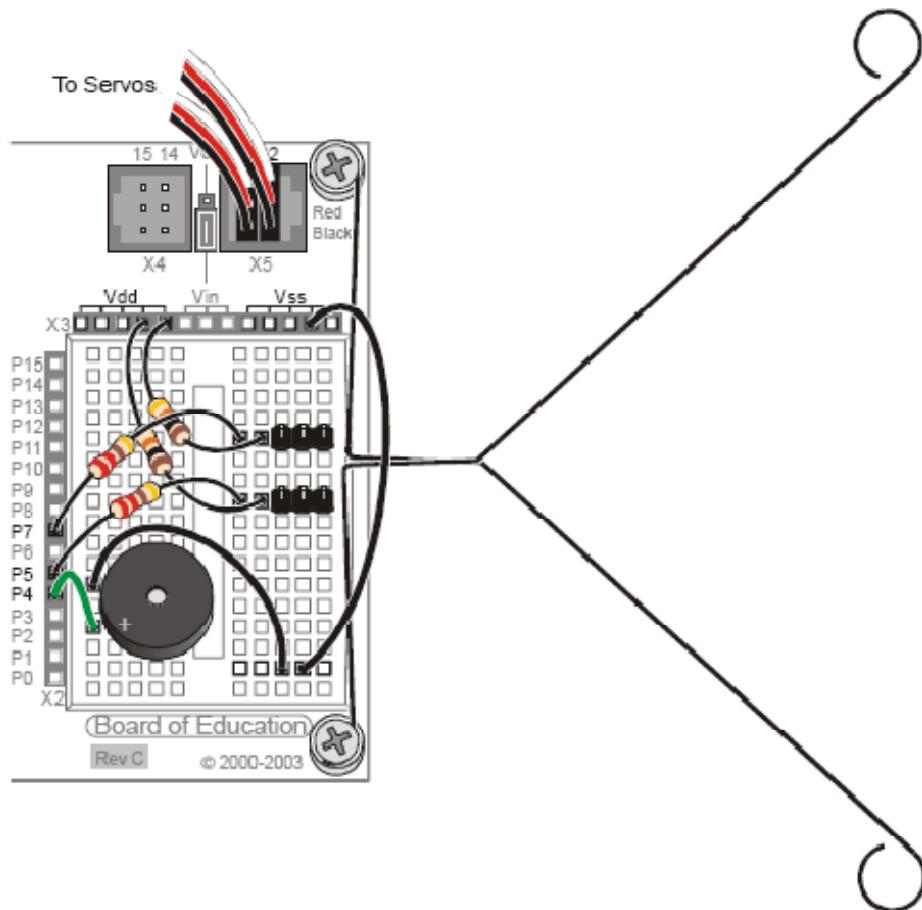
- Skinji dva prednja vijka koja osiguravaju Board of education pločicu
- umjesto njih postavi odstojnike
- na odstojnike postavi plastičnu pločicu
- postavi osiguraj pločicu sa vijcima
- vrati vijke sa ticalima na odstojnike ,tako da se rupa od ticala nalazi između glave vijka i plastične pločice.
- Postupak je prikazan na slici 16



Slika 16



Slika 17

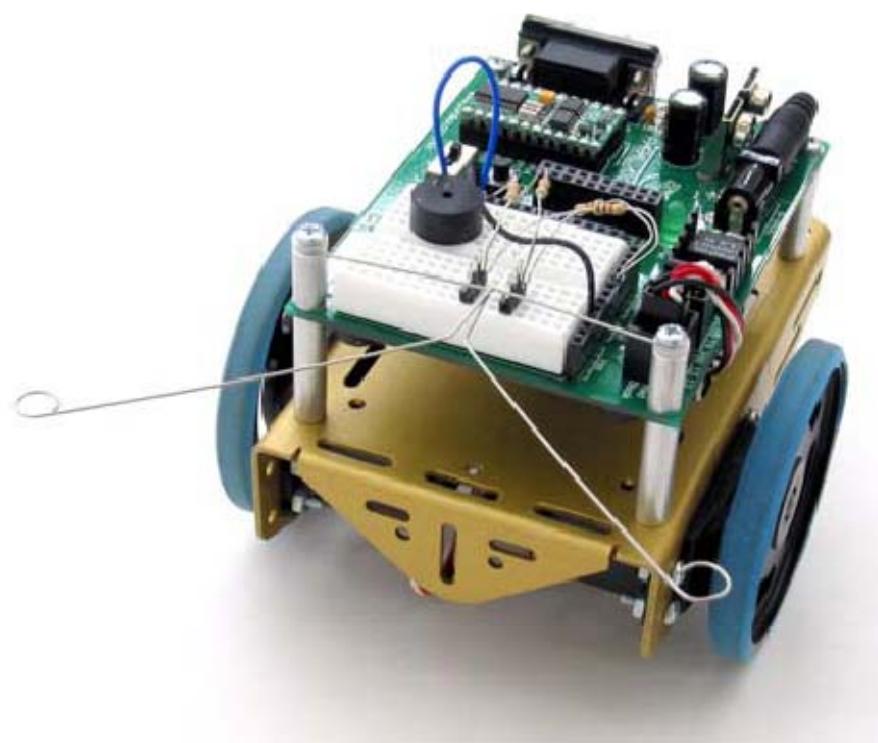


Slika 18

Elektricna shema na slici 17 , sastavljena shema na slici 18.

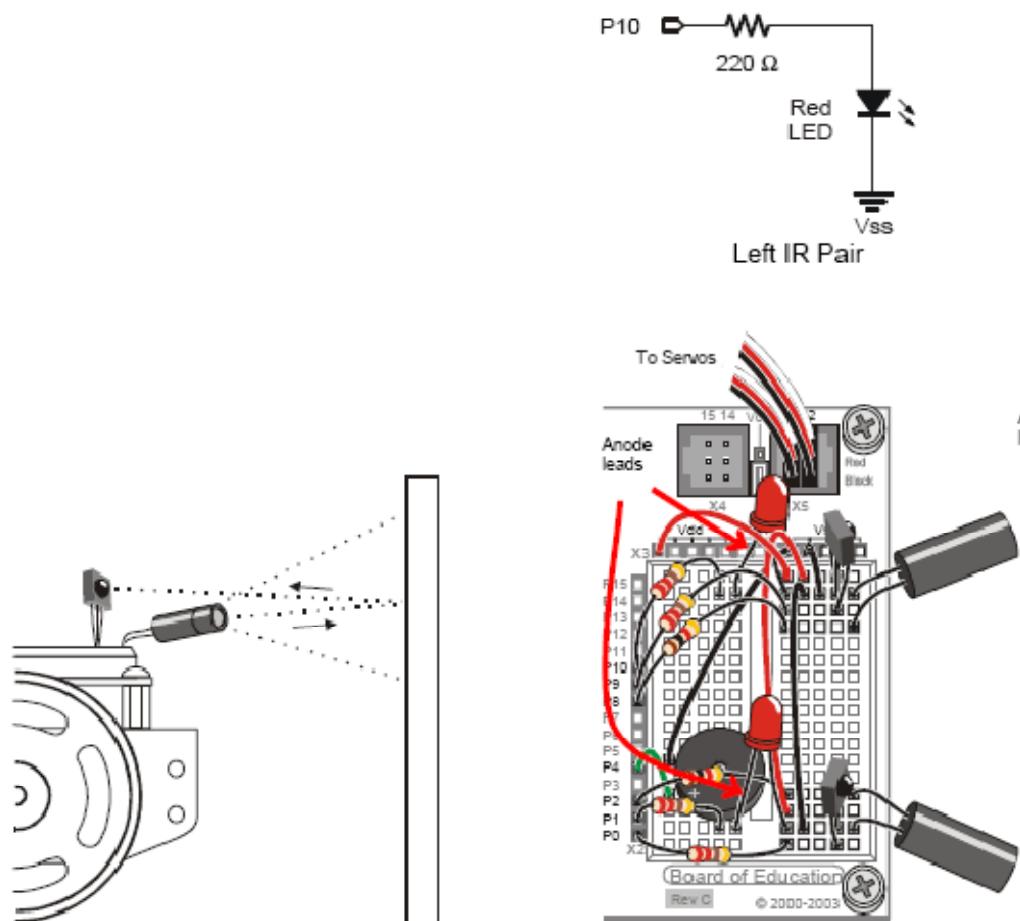
Ticala nam služe za navigaciju Boe-Bot robota ,prilikom kretanja ticalo je njegov senzor nakon što ticalo udari u neku površinu ono djeluje kao sklopka i stvara impuls

koji je jednak +5V ili log 1.Tako mikrokontroler dobiva informaciju o poziciji ticala.O programiranju ćemo govoriti u sljedećem poglavlju.



Spajanje IR senzora

Ako nečemo koristiti ticala koja djeluju na maloj udaljenosti ,moraju fizički dodirnuti prepreku. IR senzori ili Infracrveni senzori su precizniji djeluju na veću udaljenost i ne trebaju fizički dodirivati predmete kako bi se robot mogao pozicionirati.IR senzori odašilju svjetlost koja se odbija od prepreke i zatim tu odbijenu svjetlost primaju fototranzistori koji pretvaraju tu zraku svjetlosti u napon od 0 ili 5 V odnosno log 0 i log 1.(Slika 21.)



Slika 19

Slika 20

Potrebni djelovi:

Potrebno	Djelovi
2	Fototranzistor
2	Kućište infracrvene diode
2	Infracrvena dioda
2	Otpornik 220k
2	Otpornik 1k