

Izdelava mobilnega robota z razvojnim sistemom roboPIC (7)

Avtor: Silvan Bucik

E-pošta: silvan.bucik@tscng.net

Forum: www.svet-el.si/phpBB2/index.php

Spet je mesec naokoli, z roboPICom smo spet tu. V preteklosti smo spoznali delovanje enosmernih in servomotorjev. Izkušnje, ki smo jih pridobili, bomo uporabili kasneje, ko bomo robota pomikali skozi stene labirinta. Z današnjim prispevkom odpiramo novo poglavje – svet senzorike. Senzorji so ključni elementi robota. Opravljajo podobno funkcijo kot jih pri živih bitjih opravljajo čutila. Preko čutil pridobivamo informacije o prostoru, v katerem se nahajamo, o lastnostih materialov, ki nas obdajajo. Več informacij, ko imamo na razpolago, bolj zanesljivo se počutimo, naši gibi pa so odločni, natančni in hitri. Podobno velja tudi za robotske sisteme. Pravilno delovanje, ustreznost namestitvev in usmeritev senzorjev robotu omogočajo hitro in zanesljivo delovanje. Da bomo dosegli zelene rezultate, nas čaka še nekaj »šolskih ur« in precej eksperimentiranja.

Rešitve nagradne naloge

Preden pričnemo, predlagam, da si pogledamo rešitev nagradne naloge iz 125. številke. Lepo rešitev nam je poslal Jan Jevšovar, kateremu bo podjetje ELBACOMP d.o.o. kot nagrado podarilo mikrokontroler PIC16F877. Sponzorju se za nagrado iskreno zahvaljujemo, nagrajencu pa iskreno čestitamo.

Naloga se je glasila takole:

Napišite program, ki bo motor počasi zaviral od maksimalne hitrosti vrtljajev do popolne zaustavitve. Nato naj motor pospešuje do maksimalnih vrtljajev, pri tem naj se mu zamenja smer vrtenja.

```
// KONFIGURACIJSKE NASTAVITVE

#include <pic.h>
#include <pwm.c>
#include <delay.c>
__CONFIG (HS&WDTDIS&PWRRTEN&BORDIS&LVDPDIS&DUNPROT&WRTDIS&DE-
BUGDIS&UNPROTECT);

// DEKLARACIJE IN DEFINICIJE

#define BITNUM(adr,bit) ((unsigned)(adr)*8+(bit))
#define frekvenca_kHz 1000

int frq=frekvenca_kHz;
int x;
// definiranje spremenljivke
// INICIALIZACIJA

void main (void)

{
TRISA = 0B11111111; // PORTA = vhod, ga ne uporabljamo
ADCON1 = 0x07; // PORTA definiramo kot digitalna
// vhodno-izhodna vrata
TRISB = 0B11111111; // PORTB = vhod, ga ne uporabljamo
TRISC = 0B11110010; // RCO, RC3, RC2 so izhodi
TRISD = 0B11111111; // PORTD = vhod, ga ne uporabljamo
```

```
TRISE = 0B00000111; // PORTE = vhod, ga ne uporabljamo

//GLAVNI PROGRAM

x = 99; // spremenljivka, uporabljena za
// prevajalno razmerje

RC0 = 1; // vrtenje motorja
RC3 = 0; // v izbrano smer

while (x > 0) // dokler x ni nič (motor se
// ustavi), se zanka izvaja
{
pwm1_out(5000,x); // prevajalno razmerje je x,
// frekvenca 200Hz
wait_mili (50); // zakasnitev 50ms
// (motor se ustavi v 50ms * 99
// zank = 5s)
x--; // x se zmanjša za 1
}
RC0 = 0; // motor se ustavi
RC3 = 0;

wait_mili (3000); // in stoji za 3 sekunde

RC0 = 0; // vrtenje motorja
RC3 = 1; // v drugo smer

while (x < 99) // dokler x ni 99 (polna hitrost
// motorja), se zanka izvaja
{
pwm1_out(5000,x); // prevajalno razmerje je x
wait_mili (50); // zakasnitev 50ms
x++; // x se poveča za 1
}

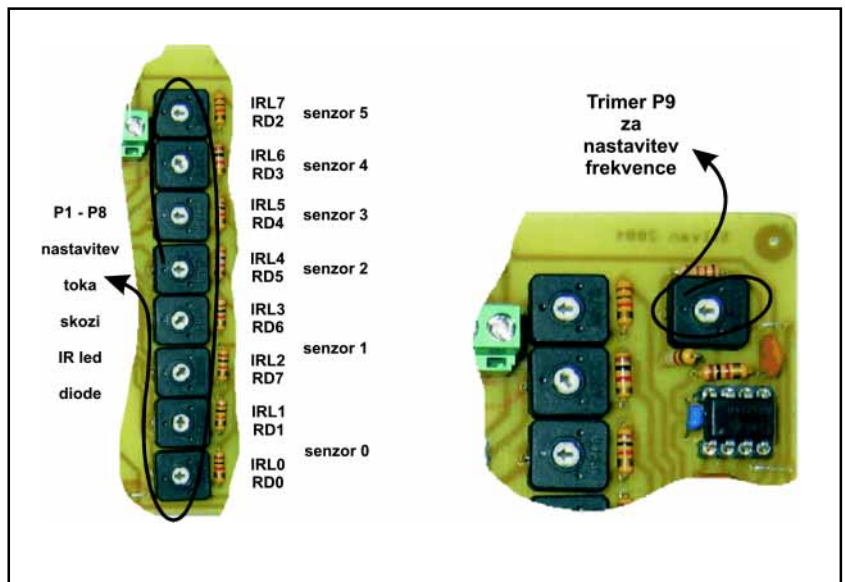
stop:
goto stop;
}
```

Senzorji z diskretnim (digitalnim) izhodom

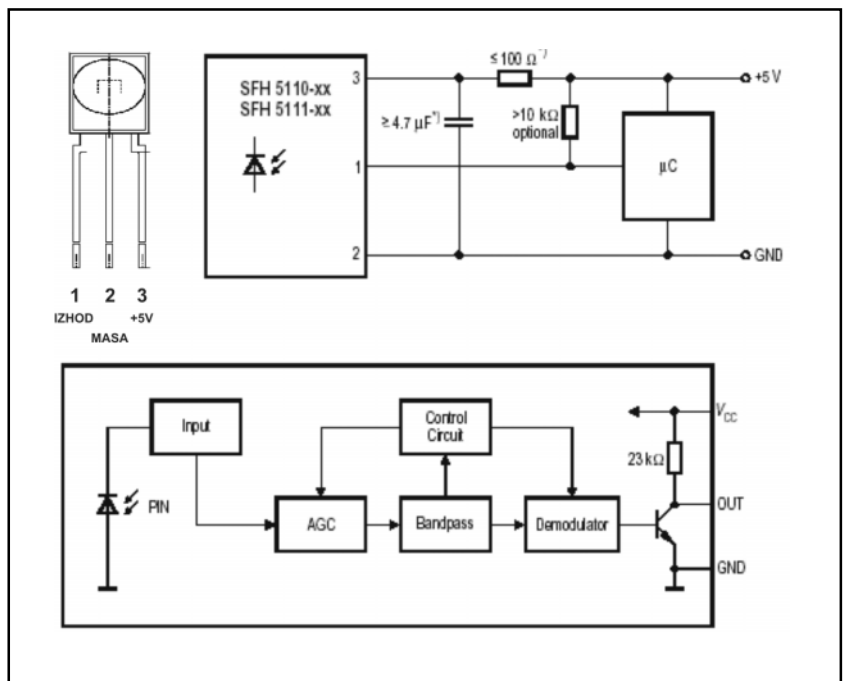
Sedaj smo prišli do uporabe sensorjev. V naših aplikacijah bomo v glavnem uporabljali senzore razdalje, ki delujejo na principu merjenja odboja infrardeče svetlobe. Preden pričnemo z razglabljanjem, bomo najprej spoznali delovanje sensorja, kasneje pa uskladili svoja pričakovanja z možnostmi, ki nam jih posamezni tip sensorja ponuja. Senzor je element, kateremu se spreminja izhodna električna veličina (napetost, tok, upornost...) v odvisnosti od razdalje, ki je meri. V naših primerih bo izhodna veličina napetost. Kot najenostavnejši senzor nam lahko služi preprosto mikrostikalo. Zamislimo si primer, ko namestimo več mikrostikal na stranske robove robota. Med vožnjo po labirintu se bo robot prej ali slej dotaknil stene hodnika. S tem se bo tudi eno izmed mikrostikal sklenilo in robotu sporočilo, da je preblizu stene. Robot se bo nato odmaknil od stene in nadaljeval z vožnjo v drugi smeri.

Namesto klasičnih stikal lahko uporabimo boljšo rešitev - senzor z diskretnim izhodom. Prednost tovrstnih sensorjev pred mikrostikali je v tem, da so brezkontaktni. To pomeni, da se robotu ni potrebno zaleteti v oviro, da bi jo zaznal. Tako senzor zazna oviro že nekoliko prej in se ji zato lahko izogne. Druga prednost, ki jo omogočajo tovrstni senzori, je tudi, da posamezne senzore lahko uporabimo za merjenje več različnih razdalj, kar poveča zmogljivosti robota.

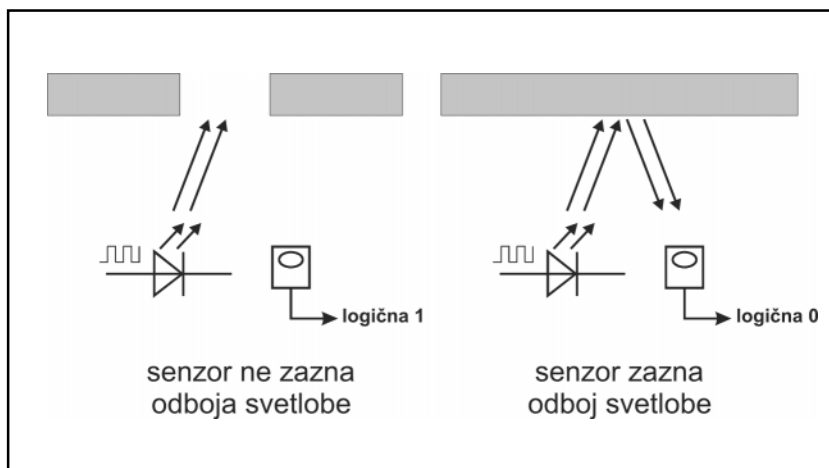
Tipični predstavnik družine sensorjev z diskretnim izhodom je senzor SFH5110 (ali SFH5111). Tovrstne senzore v praksi srečujemo kot sprejemne elemente IR vodenih naprav (primer daljinski upravljevec televizije). SFH5110 spada med infrardeče senzore, ki merijo razdaljo na principu količine odbite svetlobe. Če je senzor osvetljen z zadostno količino infrardeče svetlobe, bo na svojem izhodu spremenil logično stanje (preklopil iz stanja logične 1 v stanje logične 0). Slednje za mikrokontroler pomeni informacijo, da se je robot preveč približal oviri in bo moral ustrezno reagira-



Slika 2: Nastavitev modulirne frekvence led diode



Slika 3: Zgradba digitalnega sensorja SFH5110

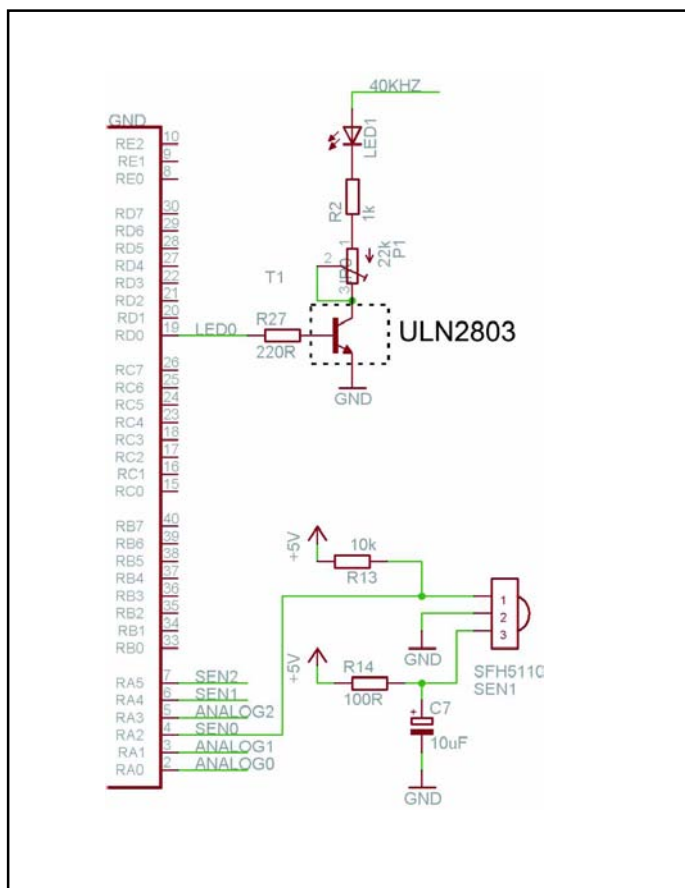


Slika 1: Princip delovanja digitalnega sensorja

ti. Element SFH5110 predstavlja le sprejemni del sensorja, celotno podobo sensorja dopolnjuje še infrardeča led dioda (na primer LD274). Ta opravlja funkcijo svetlobnega reflektorja, s katerim robot osvetljuje okolico.

Poglejmo si delovanje sensorja bolj podrobno. Led dioda oddaja v okolico modulirano infrardečo svetlobo, senzor pa odbito svetlobo meri. Odana svetloba ni »enosmerna«, ampak je utripajoča. Frekvenca utripanja je natančno določena in je odvisna od izbire tipa sensorja. Biti mora sinhronizirana s sprejemnim sensorjem, ki je uglasen le na določeno frekvenco. Ta vrednost se giblje med 30 in 40 kHz; standardne vrednosti za SFH5110 pa so 30, 33, 36, 38 in 40 kHz.

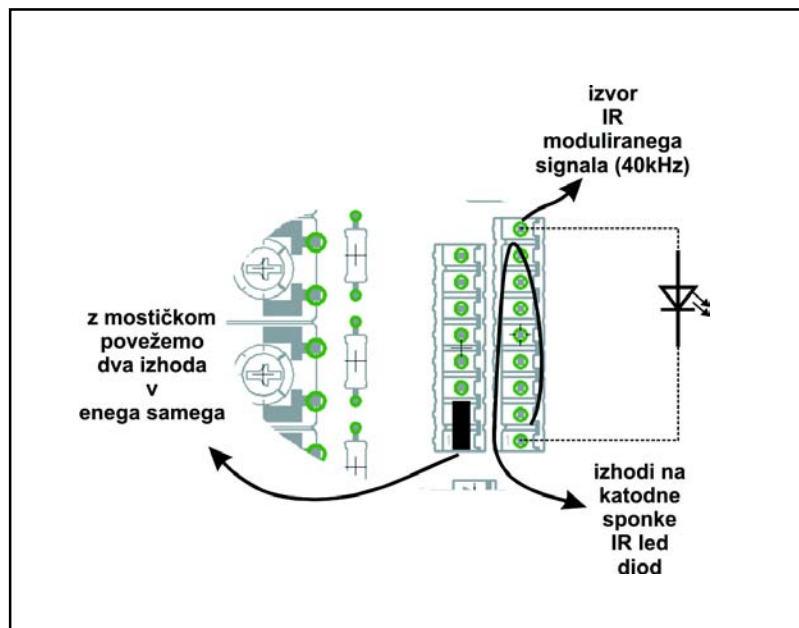
Uporaba modulirane svetlobe omogoča večjo ro-



Slika 4: Povezava digitalnega senzora z mikrokontrolerjem

bustnost, to je manjšo občutljivost senzorjev na zunanje motnje. Kot motnja na primer močno vplivajo okoliška razsvetljava, bliskavice fotoaparatorov in avtomatski merilniki razdalje (autofocus) kamer oziroma fotoaparatorov. Na tekmovanjih je potrebno računati tudi z novinarji in drugimi gosti, katerih fotonaprave za robota ne smejo biti moteče.

Senzor deluje zelo preprosto: če je osvetljen z zadostno količino



Slika 5: Priključitev IR led diod na razvojno ploščo

modulirane IR svetlobe ustrezne valovne dolžine in frekvence, bo dal na izhodu logično vrednost 0 (0V); v nasprotnem primeru bo ostal v stanju logične 1 (+5 V).

Izhod sprejemnika je izveden v izvedbi OC (open collector), zato je na izhodu potreben pull up upor vrednosti približno 10 kΩ. Poleg izhodne sponke sta na senzorju le še napajalni sponki (+5 V in GND). Senzorji so občutljivi na električne motnje, ki se širijo po napajalnih vodih, zato je napajalno napetost potrebno dodatno filtrirati z RC členom (uporom 100 Ω in kondenzatorjem 10 mF).

Spodnja slika prikazuje vezavo oddajne IR led diode in senzora. Pri montaži elementov bodimo pozorni, da sta oba pravilno usmerjena - v isto smer. Ob namestitvi pazimo, da IR led dioda ne osvetljuje sprejemnika neposredno. Oddajanje led diode omogočimo preko baze NPN tranzistorja (integrirano vezje ULN2803) z napetostnim potencialom +5 V (stanjem logične 1).

Praktični primer uporabe

Sedaj pa k nalogi. Poglejmo si program, ki bo prižgal led diodo na izhodu RB0, če bo senzor na priključku RA2 zaznal bližino predmeta.

Poglejmo si, kako bomo nalogo izvedli:

- najprej s pomočjo trimerja P9 nastavimo ustrezno frekvenco oscilatorja (časovno vezje NE555). Ta je odvisna od prej opisanih značilnosti senzora. Torej preverite kataloške podatke;
- priključka mikrokontrolerja RB0 (led signalizacija) in RD0 (IR led dioda) definiramo kot izhoda, vsi ostali priključki so vhodi. Ob tem zagotovimo, da so vse IR led diode, vključno s signalno led diodo na priključku RB0, ugasnjene;
- vklopimo NPN tranzistor, ob tem steče pulzirajoči tok skozi IR led diodo;
- počakamo 1 ms, da se stanje na senzorju stabilizira (uporabimo knjižnično funkcijo wait_mili (zakasnitev_v_ms));
- preverimo izhodno stanje senzora in z led diodo na priključku RB0 signaliziramo stanje senzora (če senzor zazna pred seboj oviro, se led dioda prižge, v nasprotnem primeru ostane ugasnjena);
 - izklopimo NPN tranzistor, s tem prekinemo tok skozi IR led diodo;
 - akcijo ves čas ponavljamo, tako da bomo lahko stanje senzora večkrat preverili (uporabimo neskončno zanko while(1>0)).

Rešitev problema v programskem jeziku C:

```
// INICIALIZACIJA

void main (void) // IME GLAVNE FUNKCIJE

{
  TRISA = 0B11111111; // PORTA definiramo kot vhod
  ADCON1 = 0x07; // PORTA definiramo kot digitalna
  vhodno-izhodna vrata
  TRISB = 0B11111110; // RB0 definiramo kot izhod
  TRISC = 0B11111111; // PORTC definiramo kot vhod, ga
  ne uporabljamo
  TRISD = 0B11111110; // RD0(IRLED0) definiramo kot
  izhod (bodite pozorni,
  // da sta IR led dioda in senzor v
```

```

                isti ravnini
                // in sta enako usmerjena
TRISE = 0B00000111; // PORTE definiramo kot vhod, ga ne
                    // uporabljamo
PORTD = 0B00000000; // vse IR led diode so ugasnjene
RBO = 1;             // dioda na RBO je ugasnjena

//GLAVNI PROGRAM

while (1>0)
{
RDO = 1;           // vklop 40kHz na
                  // IRLEDO
wait_mili(1);     // počakamo, da se stanje
                  // na senzorju stabilizira
if (RA2 == 0)    // senzor vidi predmet ob aktivni
                  // logični 0
{
RBO = 0;         // če senzor zazna oviro, se indika
                  // cijska led dioda prižge
}
else
{
RBO = 1;         // če senzor ne vidi ničesar, signalna
                  // led dioda ugasne
}
RDO = 0;         // izklopimo 40kHz na
                  // IRLEDO
}
}

```

Uporaba prireditvenih imen

Ta razdelek sicer bolj sodi v poglavje o uporabi programskega jezika C. Vendar se mi zdi pomembno, da uporabo prireditvenih imen obravnavamo na tem mestu, kjer je moč uporabe tovrstnega pristopa najbolj občutna.

V programu bomo uporabili makroukaz BITNUM, s katerim posameznim bitom v registrih priredimo nova imena. Makroukaz bomo definirali v razdelku DEKLARACIJE IN DEFINICIJE z izrazom:

```
#define BITNUM(adr,bit) ((unsigned)&adr)*8+(bit))
```

Vrnimo se na programsko rešitev prejšnje naloge. V programu smo direktno naslavljali tri bite:

- bit RA2, ki je vezan izhod diskretnega senzorja SEN0,
- bit RBO, na tem priključku je vezana indikacijska led dioda in
- bit RD0, s katerim smo omogočili tok skozi oddajno IR led diodo IRLEDOA.

Naštetim bitom bomo priredili nova imena, ki bodo takšna, da nam bo pisanje programa čim lažje.

```

bit
irled0a @ BITNUM(PORTD,0), // bitu 0 na PORTD (RDO) priredimo
                           // ime irled0a
senzor0 @ BITNUM(PORTA,2), // bitu 2 na PORTA (RA2) priredimo
                           // ime senzor0
lucka0 @ BITNUM(PORTB,0); // bitu 0 na PORTB (RBO) priredimo
                           // ime lucka0

```

RAZPIS

Odrpno državno študentsko in dijaško tekmovanje v konstrukciji in vožnji z mobilnimi roboti RoboT 2006

- Sedmo odrpno državno študentsko tekmovanje z mobilnimi roboti
- Šesto odrpno državno dijaško tekmovanje z mobilnimi roboti
- Drugo finale tekmovanja Slovenske robotske lige RoboLiga 2006
- Tekmovanje za osnovnošolce v kategoriji Lego Mindstorms LegoBum 2006



Prijave zbiramo do 31. 3. 2006; finalno tekmovanje bo 11. 5. 2006 na UM-FERI. Najuspešnejšim trem tekmovalcem iz vsake kategorije bodo v finalu podeljene nagrade:

- ⇒ Nagradni sklad v višini najmanj 100.000 SIT;
- ⇒ Denarni sklad prispevajo: podjetje MOTOMAN Robotec, d.o.o., Elektrotehniško društvo Maribor ter Sekcija elektronikov pri OZS in Območni obrtni zbornici Maribor.
- ⇒ Podjetje COMTRON prispeva praktične nagrade za A) kategorijo (računalniške komponente);
- ⇒ Izdajatelj revije Svet elektronike prispeva praktične nagrade za B) kategorijo.

Podrobnejše informacije: <http://www.ro.feri.uni-mb.si/tekma/>, janez.pogorelc@uni-mb.si, 02 220 7304

Organizira: Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Inštitut za robotiko, Smetanova 17, 2000 Maribor v sodelovanju z Zvezo za tehnično kulturo Slovenije, revijo Svet elektronike ter sponzorji: Elektrotehniško društvo, Obrtna zbornica Slovenije, MOTOMAN Robotec in COMTRON.

Opisi najuspešnejših robotov preteklih tekmovanj (zgradba, el. načrt, program za vodenje po labirintu) so na voljo v reviji Svet elektronike št. 69 (okt. 2000), št. 78 (julij 2001), št. 90 (sept. 2002), št. 116 (jan. 2005) in št. 125 (nov. 2005).



Rešitev bi v programskem jeziku C sedaj izgledala takole:

```
// INICIALIZACIJA
void main (void)      // IME GLAVNE FUNKCIJE

{
  TRISA = 0B11111111; // PORTA definiramo kot vhod
  ADCON1 = 0x07;      // PORTA definiramo kot digitalna
                      // vhodno-izhodna vrata
  TRISB = 0B11111110; // RBO definiramo kot izhod
  TRISC = 0B11111111; // PORTC definiramo kot vhod, ga ne
                      // uporabljamo
  TRISD = 0B11111110; // RDO(IRLEDOa) definiramo kot izhod
                      // (bodite pozorni,
                      // da sta IR led dioda in senzor v
                      // isti ravnini in sta enako usmerjena
  TRISE = 0B00000111; // PORTE definiramo kot vhod, ga ne
                      // uporabljamo
  PORTD = 0B00000000; // vse IR led diode so ugasnjene
  lucka0 = 1;         // dioda je ugasnjena

//GLAVNI PROGRAM

while (1>0)
{
  irled0a = 1;        // vklop 40kHz na
                      // IRLEDO
  wait_mili(1);      // počakamo, da se stanje
                      // na senzorju stabilizira
  if (senzor0 == 0)  // senzor vidi predmet ob aktivni
                      // logični 0
  {
    lucka0 = 0;      // če senzor vidi, se dioda prižge
  }
  else
  {
    lucka0 = 1;      // če senzor ne vidi, dioda ugasne
  }
  irled0a = 0;      // izklop 40kHz na
                      // IRLEDO
}
}
```

Vidimo, da je tako napisan program preglednejši in bolj razumljiv. Odslej se bomo uporabe prireditvenih imen pogosto posluževali.

Samostojno rešite sledeči nalogi:

1. Napišite program, ki bo postavil bit 2 v pomožnem registru `senzor_stat` na vrednost 1, če bo senzor na priključku RA5 zaznal bližino predmeta.
2. Napišite program, ki bo pognal enosmerni motor na izhodu X3, v primeru, da bo senzor na priključku RE0 zaznal bližino predmeta.

Nalogi rešite tako, da boste za posamezne bite v registrih uporabili prireditvena imena.

Do prihodnje številke

V tej številki smo spoznali delovanje in uporabo senzorjev z diskretnim izhodom. Tudi v bodoče bomo ostajali na senzoriki. Prihodnjič nas čakajo senzori z analognim napetostnim izhodom. ●

Literatura:

- [1] Microchip, PIC16F87x data sheet, Microchip Technology Incorporated, 2001, <http://ww1.microchip.com/downloads/en/DeviceDoc/30292c.pdf>;
- [2] B. Peršič, Gradnja mikroprocesorskih sistemov, Ljubljana: Fakulteta za elektrotehniko, 1998,
- [3] HI-TECH, PICC Lite C Manual, HI-TECH Software, 2002, <http://www.htsoft.com/downloads/manuals.php>;
- [4] B.W. Kernighan, D.M. Ritchie, Programski jezik »C«, Ljubljana: Fakulteta za elektrotehniko, 1991,
- [5] STMicroelectronics, L293 datasheet, push-pull four channel drivers, <http://www.st.com/stonline/books/pdf/docs/1328.pdf>;

Programska oprema:

- [1] Microchip, MPLAB IDE v6.61, <http://ww1.microchip.com/downloads/en/DeviceDoc/mp661.zip>;
- [2] HI-TECH software, PICC lite COMPILER v8.05PL2, <http://www.htsoft.com/products/PICClite.php>;
- [3] Microchip, PIC18F/PIC16F Quick Programmer, http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1824&appnote=en012031;
- [4] Petr Kolomaznik, EHL elektronika, PIC downloader v1.08, <http://www.ehl.cz>;
- [5] Herman Aartsen, TNO - The Netherlands, PIC Bootloader +, <http://www.microchipc.com/>

AX ELEKTRONIKA svet **ELEKTRONIKE**

080 1741

BREZPLAČNA ŠTEVILKA ZA VSE INFORMACIJE