INFORMACIJSKA DRUŽBA – IS´2009
16. oktober 2009
VZGOJA IN IZOBRAŽEVANJE V INFORMACIJSKI DRUŽBI

Prednosti blokovnega programiranja robotov v osnovni šoli

Advantages of Function Block Programming

Milan Hlade
OŠ Koroška Bela Cesta talcev 2 4270 Jesenice
milan.hlade@gmail.com

Povzetek
Prispevek opisuje primer uporabne povezave računalnika, vmesnika in stroja. Učenci osvojijo osnove programiranja. Utrdijo pridobljeno znanje o elektriki, računalništvu, tehniki, fiziki, kemiji, biologiji…. Spoznajo dobre in slabe strani naprav, avtomatov. Spoznajo tudi področja uporabe robotskih naprav in avtomatiziranih strojev. Znajo uporabiti pridobljeno znanje na različnih področjih in s tem tudi znajo povezovati različna znanja.
Predmet navaja učence na koristno uporabo računalnika. Spoznajo, da je računalnik mogoče uporabiti za krmiljenje povsem vsakdanjih naprav. Učno uspešnejšim učencem nudi omenjeni program še mnogo več. Vse možnosti in funkcije programa niso opisane zaradi pomanjkanja prostora.

Ključne besede: računalnik, vmesnik, mobilni robot, blokovno programiranje, medpredmetno povezovanje

Abstract
In this contribution I describe an example of useful connection between computer, interface and machine. Pupils get familiar with the basis of programming. They consolidate knowledge about electricity, computer science, technology, physics, chemistry, biology… They get to know good and bad sides of devices, machines. They know how to use the knowledge on different areas and consequently connect different knowledges.
This subject aim is useful use of computer. Pupils learn that the computer can be used for steering entirely common devices. This program offers much more especially to very successful pupils. All possibilities and functions are not decribed due to lack of space.

Keywords: computer, interface, mobile robot, block programming, intersubject connection

1 UVOD

Skupno robotiki je uporaba robota, računalnika in vmesnika med obema. Programiranje je bilo za učence do sedaj najtrši oreh robotike v OŠ delno tudi zaradi zahtevne sintakse v programskih jezikih.

Na OŠ Koroška Bela pri robotiki uporabljamo t.i. blokovno programiranje (del vizualnega programiranja), ki ga učenci veliko bolje razumejo kot klasično programiranje, saj je sestavljen program zelo podoben diagramu poteka. Učenci spoznajo osnovne programske ukaze v obliki blokov ter nato s sestavljanjem blokov izdelajo program.

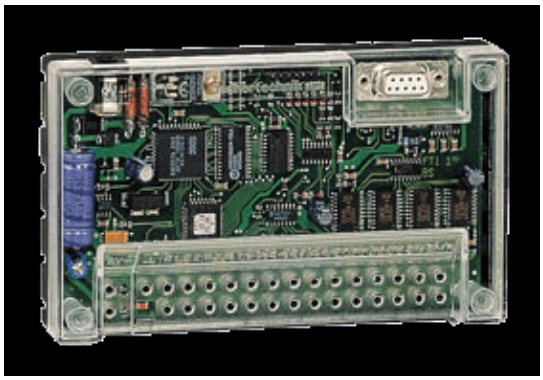
Vmesnik za krmiljenje robotov pa ni uporaben le za robotiko samo, ampak se lahko uporabi za meritve pri različnih predmetih, predvsem pri fiziki.

Programiranje v klasičnih programskih jezikih je za učence osnovnih šol v večini primerov prezahtevno zaradi (za osnovnošolce) zapletene sintakse, manjše preglednosti izvajanja programa, odpravljanja napak pri programiranju, zahtevanega osnovnega znanja o programu in motivacije same. Povezovanje izvajanja programa v računalniku ali vmesniku s praktičnim dogajanjem je za osnovnošolsko znanje s klasičnim programiranjem preveč zahtevno. Tu najdemo razlog za zelo majhno zanimanje za robotiko, poznavanje in uporabo vmesnika tudi za ostala področja (fizika, naravoslovje, kemija, biologija….). Blokovno programiranje, ki je del tako imenovanega vizualnega programiranja z jasnim, preglednim in enostavnim programiranjem odpravi zgoraj omenjene težave. Učenci le povezujejo izbrane bloke, ki predstavljajo klasične programske stavke s puščicami, ki kažejo potek programa.

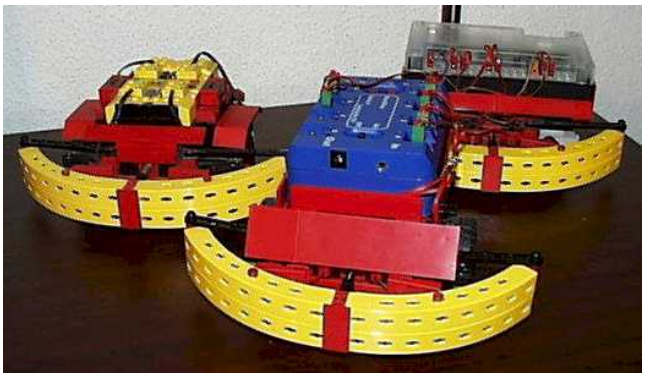
2 BLOKOVNO PROGRAMIRANJE

2.1 Vmesnik in sestavni deli

Za programiranje robotov in tudi ostalih naprav potrebujemo računalnik, vmesnik, napravo – robota in ustrezen računalniški jezik. O računalniku ni treba izgubljati besed, saj je za enostavne primere primeren skoraj vsak ne preveč star osebni računalnik. Vse ostalo smo dobili pri firmi Fischertechnik v kompletu Robo Pro. Dobili smo primeren vmesnik – slika1, sestavne dele za izdelavo mobilnih robotov in ostalih sklopov – slika 2 in programsko opremo za pisanje programov s pomočjo blokov.



Slika 1: Vmesnik - interface



Slika 2: Sestavljeni mobilni roboti

Vmesnik ima tri možnosti povezave z računalnikom; SR 232, USB priključek in infrardečo povezavo, možnosti električne napeljave direktno iz polnilca ali iz baterije z napetostjo 10

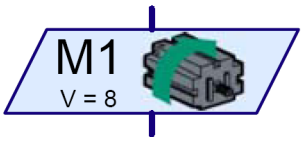
voltov. Za samo upravljanje robota ima na voljo 8 digitalnih vhodov, 4 analogne vhode in 4 izhode, z razširitvenim modulom pa se možnosti še povečajo. Ima možnost nadgradnje za radio frekvenčen prenos podatkov in programov in upravljanje. V vmesnik se lahko naložijo trije programi.

2.2 Blokovno programiranje

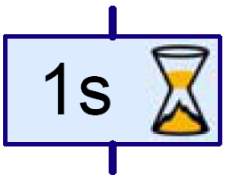
Najpogosteje uporabljeni bloki

Slika 3 prikazuje blok za začetek programa. Vsak blok ima različno število poti, po katerih se lahko program nadaljuje. Blok start ima seveda le eno pot. Bloki morajo biti povezani s potjo; črto, ki se konča s puščico, ki kaže smer izvajanja programa. Naslednji blok na sliki 4 predstavlja ukaz za določen elektrotehniški element, ki je lahko elektromotor, žarnica, elektromagnet, ventil, tuljava…. Vsak blok ima povezavo na tako imenovano nastavitveno okno, kjer se nastavljajo parametri, ki so pomembni za določen blok. Pri bloku na sliki 4 nastavljamo različne lastnosti kot so: izberemo številko izhoda na vmesniku, intenziteti delovanja, smer vrtenja, ustavljanje…. Na sliki 5 pa vidimo blok za časovno programiranje, pri kateri se program ustavi za čas, ki je vpisan v ikoni in katerega lahko spreminjamo.

Slika 3: start blok



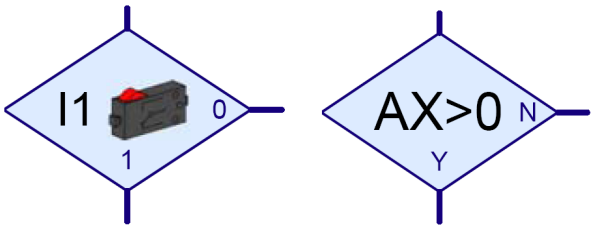
Slika 4 : blok za ukaz



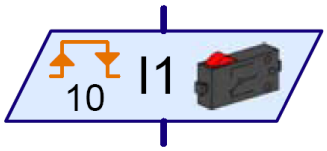
Slika 5 : blok za ukaz

Na sliki 6 vidimo blok za klasični IF stavek. Program lahko poteka po dveh poteh, odvisno od pogoja, ki je izpolnjen v IF stavku. Tako imenovani odločitveni blok lahko nastavljamo na digitalno ali analogno vrednost, lahko pa menjamo tudi izhoda 1 in 0, prav tako pa številko vhoda.

Program pa lahko krmilimo tudi z impulzi, ki jih na stikalu sproži posebno zobato kolo, ki sproži štiri impulze pri enem obratu. Na ta način lahko programiramo mobilnega robota za natančno prevoženo razdaljo. Na sliki 7 vidimo primer štetja impulzov. V programu lahko nastavljamo število impulzov. Z enim impulzom robot prevozi 4 cm poti ne glede na izbrano hitrost motorja ali napetosti baterije.



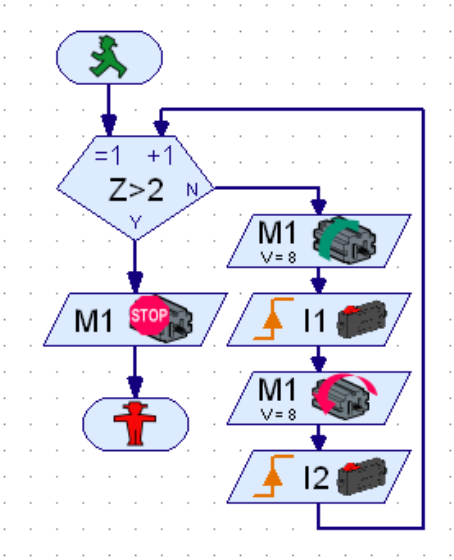
Slika 6: bloka za analogni in digitalni IF stavek



Slika 7: blok za štetje impulzov

Za programiranje večkrat ponavljajočih se operacij uporabljamo ugnezdeno zanko. Tudi pri tem bloku nastavljamo število ponovitev zanke, lahko pa tudi zamenjamo tako imenovana

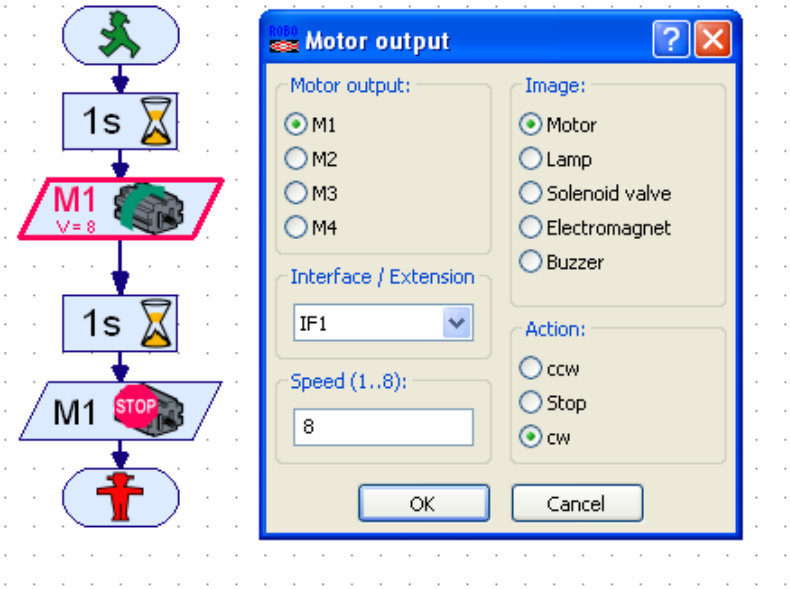
YES in NO izhoda. Blok vidimo na sliki 8. Obstaja tudi poseben blok za ugnezdeno zanko, kjer se vrednost ponovitev določa s spremenljivko, katere vrednost določimo med programiranjem.



Slika 8: blok za ugnezdeno zanko

Nastavitveno okno

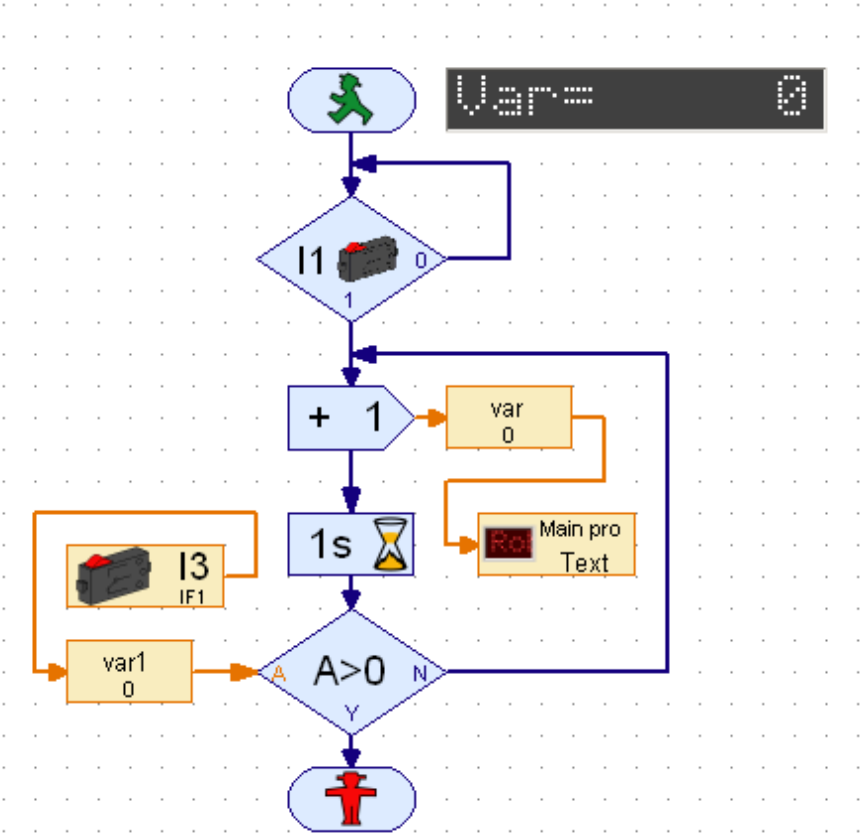
Vsakemu bloku razen bloku start in stop lahko nastavljamo razne parametre. Parametri so prikazano številsko ali grafično v nastavitvenem oknu in bloku samem. Na sliki 9 vidimo nastavitveni blok za nastavitve elektromotorja. Blok, kateremu nastavljamo parametre se obarva rdeče, kar preprečuje nastavljanje napačnega bloka. V nastavitvenem oknu za elektromotor nastavljamo številko izhoda iz vmesnika, vrsto električnega elementa, hitrost motorja ter smer vrtenja in ustavitev. Kot vidimo na bloku se parametri grafično ali številsko pojavijo v bloku. Parametre prikazane v posameznem bloku je veliko lažje spremljati, nastavljati, spreminjati z nastavitvenem oknu na podlagi grafičnih znakov, kot pa vrednosti parametrov iskati v vrsticah programov.



Slika 9: Nastavitveno okno izbranega bloka

Sestavljanje programa

Kot smo že dejali, programiranje poteka v tako imenovanem blokovnem sistemu. Na sliki 10 vidimo sestavljen program. Posamezni bloki so povezane s črtami, potmi programa, po katerih se program tudi izvaja. Program na sliki št. 9 meri čas v sekundah. Čas začnemo meriti z enim, končamo pa z drugim stikalom. Čas v sekundah je prikazan na posebnem mestu, tako imenovanem izpisu.

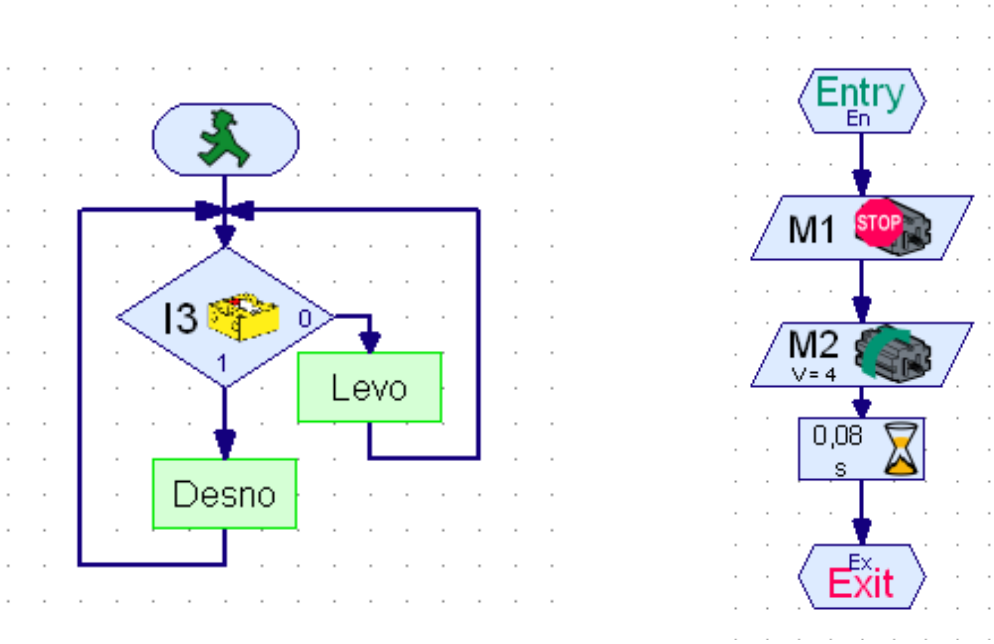


Slika 10: Program napisan s pomočjo blokov

Podprogrami

Za večkrat ponavljajoče enake operacije pa lahko izdelamo tako imenovane podprograme, ki jih uporabimo v različnih glavnih programih. Na sliki 11 vidimo podprograma: Levo in Desno, ki opravita določene funkcije, ki se večkrat ponavljajo. Na desni strani slike pa vidimo podprogram Levo. Vsak podprogram je prikazan kot blok posebne oblike.

Slika 11 predstavlja program, s katerim mobilni robot vozi po črni črti na beli podlagi. Fotocelica vseskozi nadzira položaj robota in »izbira« kateri podprogram se bo izvedel v odvisnosti kje se fotocelica nahaja.



Slika 11: Program z bloki podprograma in podprogramom

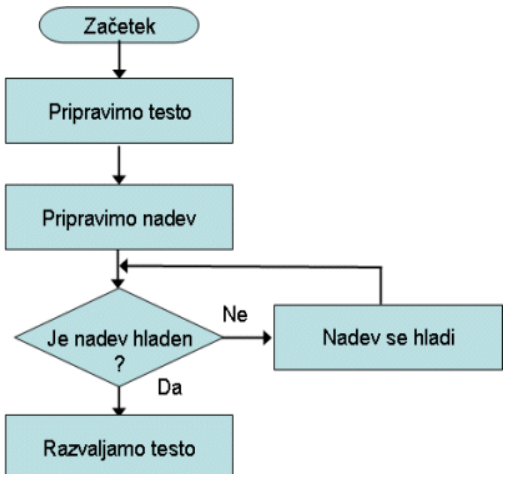
Spremenljivke

V program lahko vključimo bloke, ki nam shranjujejo določene številčne vrednosti kot spremenljivke. Primer programa s spremenljivkami vidimo na sliki 10 vedno, ko program naleti na določen blok, ki omogoča povezavo do bloka s spremenljivko, se številčna vrednost shrani v blok s spremenljivko. Spremenljivke lahko uporabimo za krmiljenje programa ali kot zapis izmerjene vrednosti. Zadnja možnost je še posebej uporabna pri raznih meritvah, kjer se fizikalna meritev shrani kot spremenljivka v analogni vrednosti običajno električne napetosti. Vrednosti spremenljivk je mogoče izvoziti v druge programe npr. MS Excel za kasnejšo obdelavo.

2.3 Prednosti blokovnega programiranja

Velika grafična predstavljivost programa

Kot sem dejal že v uvodu, je sam program zelo podoben t.i. diagramu poteka. Še posebej je to uporabno pri obsežnejših programih, kjer osnovnošolci hitro izgubijo predstavo in povezavo med diagramom poteka in napisanim programom. Pred programiranjem je za učence koristno narisati diagram poteka zaradi logičnih rešitev pri reševanju določene naloge. Tudi parametre posameznega bloka si lahko napišejo poleg vsakega bloka že pri sestavljanju diagrama poteka. Slika 12 predstavlja diagram poteka preprostega problema iz vsakdanjega življenja. Če program lahko napišemo s podobnimi znaki, je programiranje veliko lažje.



Slika 12: Primer preprostega diagrama poteka

Hitro odkrivanje napak pri sestavljanju programa

Programsko orodje samo opozori sestavljavca programa na napako pri programiranju. Največkrat so to nepovezani bloki ali veje programa. Vsak blok, pri katerem se je pojavila napaka se pred nalaganjem programa v vmesnik obarva rdeče. Na ekranu pa se pojavi tudi opis napake. Še posebej je opisana pomoč dobrodošla v večjih skupinah ali pri učencih z določenimi učnimi težavami; hiperaktivnost, disleksija….

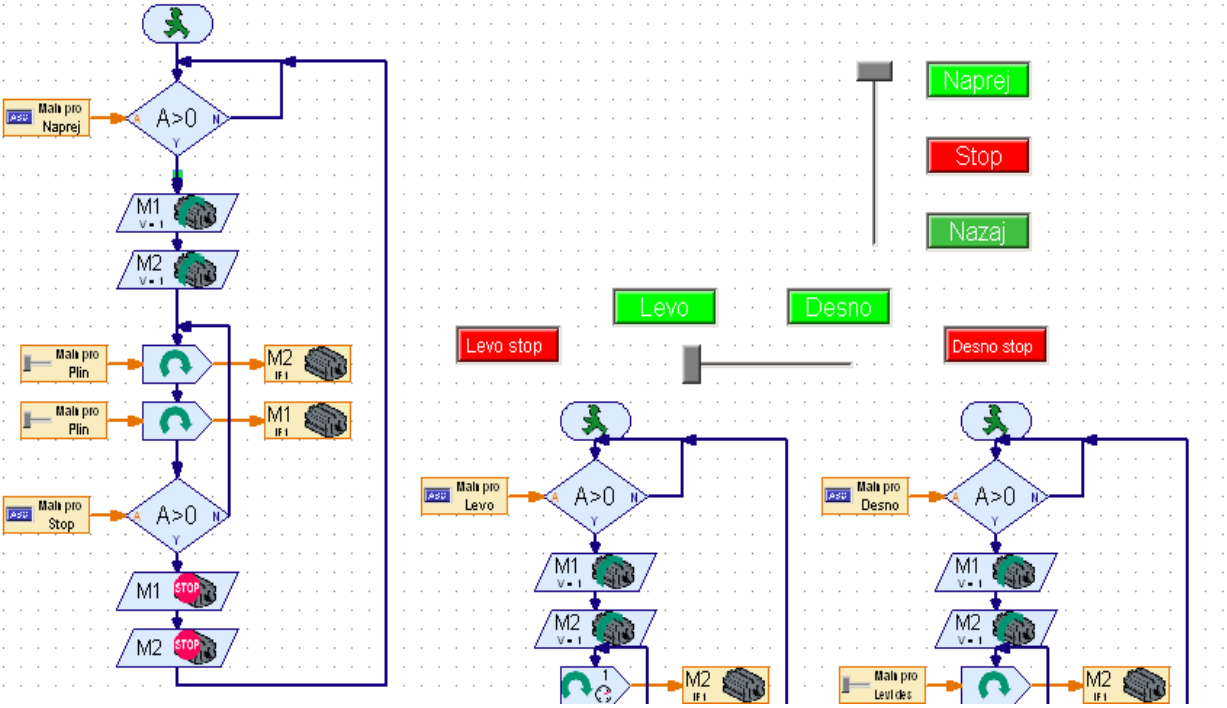
ON LINE delovanje programa

Napisan program lahko poženemo in s tem testiramo v tako imenovanem ON LINE načinu. Delovanje oz. dogajanje med programom je prikazano na ekranu tako, da je blok, kjer se trenutno nahaja program obarvan rdeče. Seveda je pogoj ON LINE delovanje konstantna povezava vmesnika in računalnika. Z infra rdeči ali radio frekvenčno povezavo odpade večkrat moteča in nadležna fizična povezava vmesnika in računalnika.

Interaktivni gumbi

Interaktivni gumbi na ekranu omogočajo krmiljenje mobilnega robota ali poljubne naprave preko ekrana. Na zaslonu izdelamo interaktivne gumbe s katerimi vključujemo ali izključujemo razne električne naprave, ali jih krmilimo z drsniki. Vrednost drsnika se shrani v spremenljivki, ki nato krmili električni element.

Na sliki 12 vidimo program, s katerim krmilimo mobilnega robota preko interaktivnih gumbov. Z gumbi vključujemo posamezen ukaz, z drsniki pa nastavljamo intenzivnost ukaza. S tem programom robota lahko vozimo naprej in nazaj ali zavijamo levo ali desno.

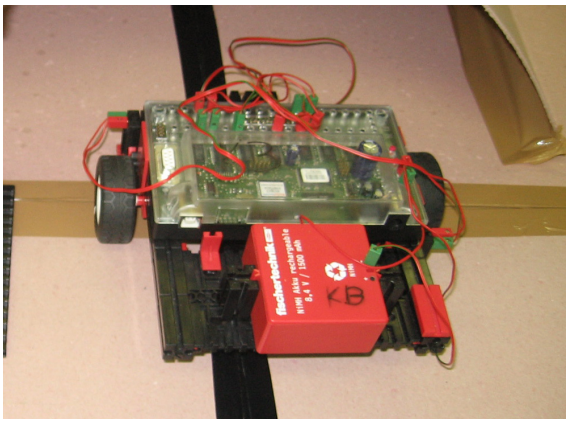


Slika 12: Primer programa z interaktivnimi gumbi.

2.4 Možnosti programiranja v osnovni šoli

Možnosti programiranja v osnovni šoli je veliko. Na sliki 13 vidimo portalni žerjav, ki je krmiljen z vmesnikom. Na sliki 14 pa vidimo sestavljenega mobilnega robota, ki je primeren za naloge, ki se lahko izvajajo s preprostimi vozili.

Slika 13: Krmiljenje portalnega žerjava http://www.jojos-homepage.de/ft/portalkran/PortalkranobenM.jpg 23. avgust 2002 14:08:20



Slika 14: Mobilni robot

V nadaljevanju so opisani primeri nalog rešenih z opisano opremo. Naloge se ločijo po vrsti naprave, ki jo programiramo.

Naloge za mobilnega robota:

- Robot naj se pet sekund pelje naravnost, nato pa naj se ustavi.
- Robot naj se vrti v smeri urinega kazalca okoli desnega kolesa. Vrtenje sprožimo in ustavimo s stikalom.
- Robot naj prevozi točno 1 meter dolgo razdaljo in se vrne na začetni položaj
- Robot naj se ustavi pred steno in naj vseskozi išče prosto pot. Pri tem naj sveti luč na robotu.
- Z robotom in fotoaparatom posnemi film.
- Robot naj se vozi po črni črti
- Robot naj se ogne postavljenim oviram.
- Robot naj se počasi vozi po mizi. Postavi stikala tako, da z uporabo ustreznega programa ne pade z mize.
- Krmili robota s pomočjo interaktivne table

Naloge ki rešujejo probleme na raznih napravah:

- Žerjav naj prenese preko ovire deset bremen
- Pri gibanju predmetov po tekočem traku naj se po desetem predmetu trak ustavi. Po desetih sekundah, ko je nameščen nov zaboj za predmete naj se trak znova zažene.
- Program naj zapiše vrednost osvetlitve fotocelice na ekran.
- Program naj zapiše koliko napetosti ima še izbrana baterija.
- Program naj premakne predmet, ki se premika z vijakom za 20 milimetrov.
- Program naj nariše enostavno črko na modelu tiskalnika.
- Napiši program, ki šteje učence, ki so kosili v šoli.

3 MEDPREDMETNE POVEZAVE

Vmesnik s priloženo programsko opremo je zelo primeren za uporabo, simulacijo oziroma za merjenje raznih fizikalnih količin predvsem pri naravoslovnih predmetih. S priloženimi električnimi elementi lahko merimo osvetlitev, prevodnost, čas, električno napetost in tok, temperaturo….

V nadaljevanju so opisani nekateri primeri uporabe vmesnika v medpredmetnih povezavah, predvsem pri Fiziki in Tehniki in tehnologiji…

3.1 Fizika – enakomerno gibanje

Z mobilnim robotom lahko demonstriramo enakomerno gibanje. Program nam omogoča osem različnih hitrosti. Izmerimo pot, ki jo mobilni robot prevozi v določenem času. Iz izmerjenih podatkov lahko narišemo diagram pot v odvisnosti od časa. Naj poudarim, da je v spodnji tabeli navedena hitrost mobilnega robota le s številčno vrednostjo, kar ni pravilno. To je posledica parametra v nastavitvenem oknu in bloku samem s katerim krmilimo elektromotor. Kasneje lahko iz podatkov izračunamo pravo hitrost. Na omenjeno vrednotenje hitrosti moramo še posebej opozoriti učence v devetem razredu, da nebi prihajalo do napačnih predstav v povezavi s fiziko.

Pri merjenju razdalje prevožene v določenem času moramo biti pozorni na polnost baterije. Pri mejni napetosti, ki še zagotavlja normalno delovanje mobilnega robota je lahko prevožena pot v predpisanem času pri manj polni bateriji precej manjša.

V tabeli 1 izmerimo pot, ki jo robot opravi v določenem času pri določeni hitrosti motorja.

	Čas gibanja robota (s)		Izračunaj pot v 15, 20 s		Kolikšno pot je opravil v 1 s
»Hitrost« motorja	5	10	15	20	
1.					
2.					
3.					
4.					
5.					
6.					
7.					
8.					

Tabela 1: Prevožena pot mobilnega robota pri različnih hitrostih in časih

V diagram 1 narišemo lahko hitrost v odvisnosti od časa. Iz strmine premice bodo učenci hitro odčitali hitrost mobilnega robota pri različnih nastavitvah. Mobilnega robota lahko uporabimo tudi kot demonstracijski pripomoček pri pouku fizike.

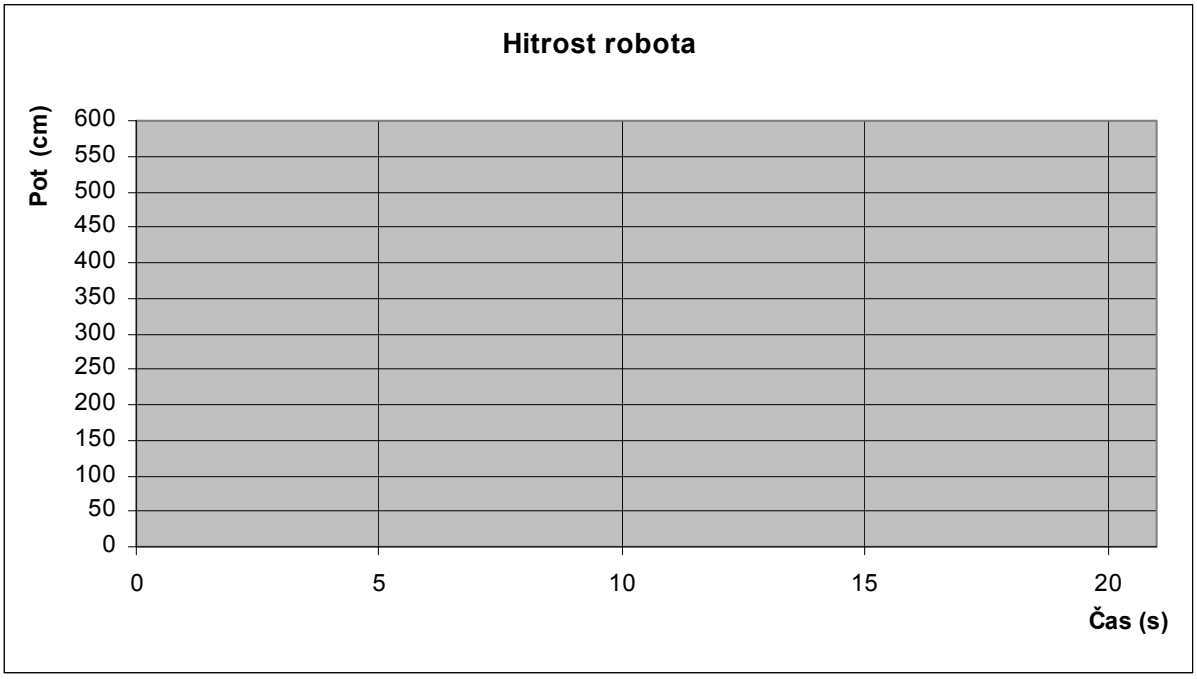


Diagram 1: Pot v odvisnosti od časa

Izračunamo lahko dejansko hitrost robota za vsako »hitrost« motorja iz programa v tabeli 2

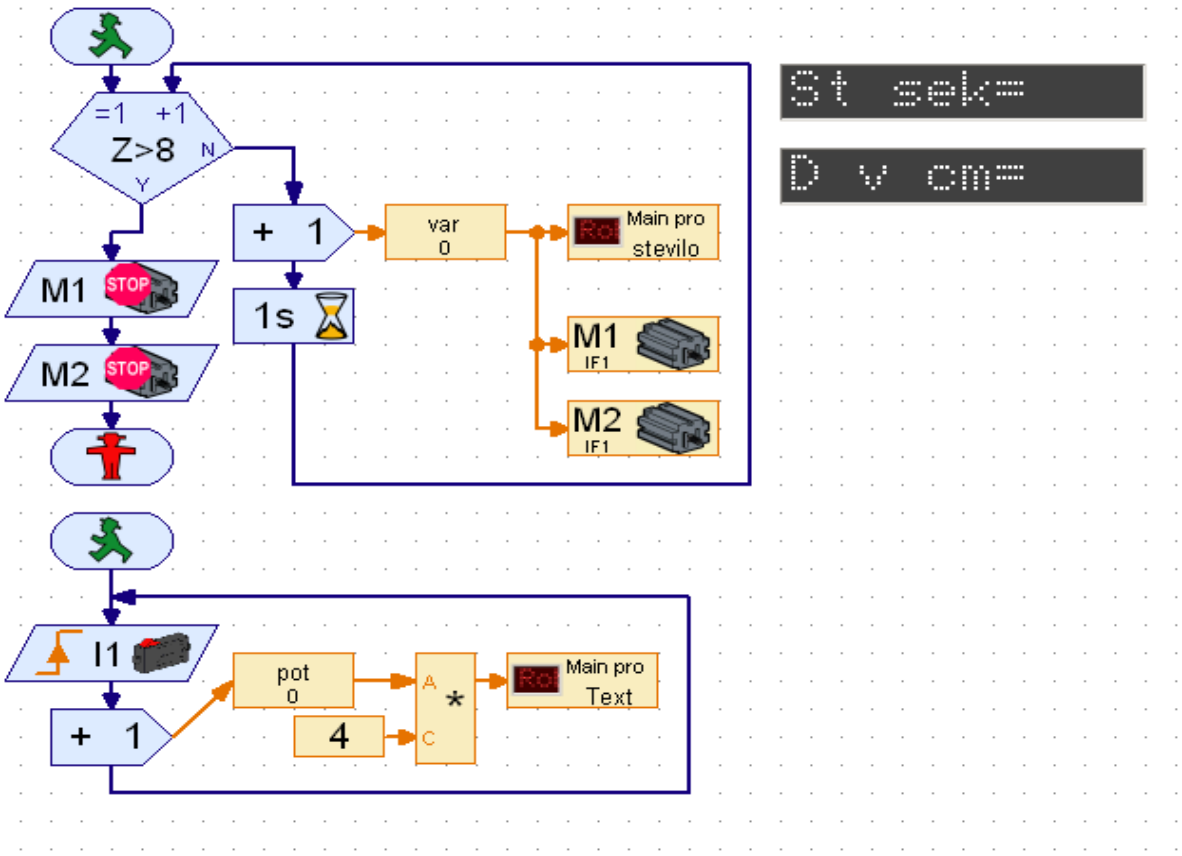
»hitrost« motorja	1	2	3	4	5	6	7	8
Hitrost robota v (cm/s)								
Hitrost robota v (m/s)								
Hitrost robota v (km/h)								

Tabela 2: Dejanska hitrost mobilnega robota

3.2 Fizika – enakomerno pospešeno gibanje

Z mobilnim robotom pa lahko demonstriramo tudi enakomerno pospešeno gibanje. Na sliki 15 vidimo program, ki ga lahko uporabimo za demonstracijo enakomerno pospešenega gibanja. Število sekund nastavljamo v bloku za ugnezdene zanke in se nam prikažejo na izpisu. Spodnji program nam pa na izpisu izpiše prevoženo razdaljo v centimetrih.

Mobilni robot nam poveča hitrost vsako sekundo za eno vrednost v programu. To vrednost že imamo iz prejšnje naloge, kjer smo računali hitrosti mobilnega robota. Ko prevoženo razdaljo vnesemo v diagram, se pokaže skoraj prava parabola.



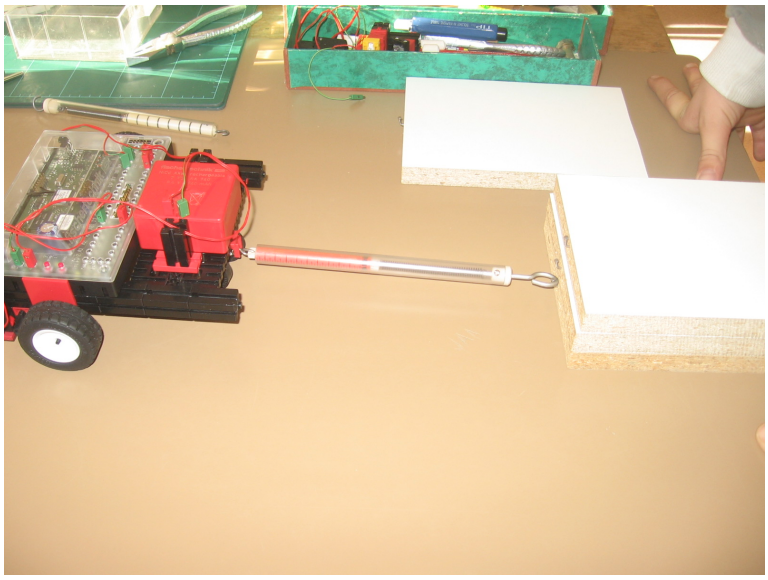
Slika 15: Program za demonstracijo enakomerno pospešenega gibanja

3.3 Fizika, Tehnika in tehnologija – merjenje napetosti toka

Z vmesnikom lahko izmerimo, kolikšno električno napetost nam daje posamezen električni vir. Vrednost lahko prikažemo na izpisu. Vrednost lahko merimo vsako desetinko sekunde, kar je uporabno pri spremenljivih virih. Napetost moramo izmeriti pri analognem vhodu. Prepričati se moramo, da napetost ni večja od 10 voltov. V nasprotnem primeru uporabimo pred upor. Na izpisu lahko izberemo analogni izpis, ki je podoben klasičnemu voltmetru.
Način merjenja električnega toka izvedemo klasično z ampermetrom.

3.4 Fizika – merjenje sil

Na mobilnega robota z vrvico pritrdimo silomer in preko njega vlečemo lesen kvader mase 0,2 kg. Izmerimo silo, s katero robot vleče kvader. Gre za demonstracijski poizkus, s katerim pokažemo velikost sil, vpliv podlage na velikost sile trenja in vpliv sile na podlago na silo trenja. Na sliki 16 vidimo merjenje sile s pomočjo mobilnega robota.



Slika 16: Primer merjenja sile s pomočjo mobilnega robota

Literatura:

Gerlič, I. (1995): Osnove elektronike. Založba obzorja Maribor .

Kocijančič, S., Sušnik, B., Hajdinjak, L. (2002): Tehnika in tehnologija 7 in 8. TZS, Ljubljana

Kralj, S., Karič, E., Zidanšek, E., Slavinec, M.(2003): Fizika 8 in 9. DZS, Ljubljana

Učni načrt za izbirna predmeta RVT in EIR
http://www2.pef.uni-lj.si/jj/Razno/Robotika_v_tehniki_izbirni.pdf 19. maj 2009 16:27:55

http://www.mss.gov.si/fileadmin/mss.gov.si/pageuploads/podrocje/os/devetletka/predmeti_izbirni/Elektrotehnika_izbirni.pdf 2. junij 2009 10:42:48

Konstrukcijski kompleti fischertechnik http://www.fischertechnik.si/prog.php?skup=70-05
8. september 2009 21:17:58

Uradna stran podjetja Fischertechnik http://www.fischertechnik.de
8. september 2009 21:24:41

Peter E. King: Procon Technology sells fischertechnik robots, robotic kits, hobby kits, interface units, PLC software, Programmable Control software
http://users.tpg.com.au/users/p8king/releas15.htm 6. september 2009 12:51:34