

# Računalniška merjenja zbirka vaj

**Študijsko gradivo za študente  
Pedagoške fakultete v Ljubljani**

**Janez Jamšek**

**Študijsko leto 2005/2006**



# Kazalo

<b>1. UVOD V RAČUNALNIŠKO KRMILJENJE.....</b>	<b>4</b>
1.1 TESTNI PROGRAM: EPROLAB:.....	4
1.1.1 Digitalni izhod.....	4
1.1.2 Analogni izhod.....	4
1.1.3 Digitalni vhod.....	5
1.1.4 Analogni vhod.....	5
<b>2. MERJENJE TOKOVNO NAPETOSTNIH KARAKTERISTIK.....</b>	<b>5</b>
2.1 MERJENJE KARAKTERISTIKE UPOROV.....	5
2.2 MERJENJE KARAKTERISTIKE DIOD V PREVODNI SMERL.....	6
<b>3. RAČUNALNIŠKI OSCILOSKOP - HISKOP.....</b>	<b>6</b>
<b>4. UPORABA EL. PREGLEDNIC ZA PRIKAZ IN ANALIZO MERITEV.....</b>	<b>7</b>
4.1 PRENOS PODATKOV.....	7
4.1.1 Odložišča (clipboard).....	7
4.1.2 Datoteko.....	7
4.2 OBLIKOVANJE TABEL.....	7
4.2.1 Obrobe.....	7
4.2.2 Številski formati.....	7
4.2.3 Enote.....	7
4.3 RAČUNANJE V EXCELU.....	7
4.3.1 Enačbe.....	7
4.3.2 Funkcije.....	8
4.3.3 Kopiranje.....	9
4.3.4 Tvorba zaporedij.....	9
4.3.5 Nelinearne prilagoditvene funkcije.....	9
4.4 KREIRANJE GRAFA.....	9
4.4.1 Oblikovanje grafikona.....	9
4.4.2 Vstavljanje trenda.....	10
4.4.3 Dodajanje grafa na že obstoječi grafikon.....	10
4.5 SPREMINJANJE DECIMALNEGA SIMBOLA.....	10
4.5.1 V datoteki txt.....	10
4.5.2 V operacijskem sistemu.....	10
<b>5. PROGRAMIRANJE RAČUNALNIŠKO MERILNEGA KRMILNEGA SISTEMA CMC.....</b>	<b>11</b>
5.1 OSNOVNE SESTAVINE.....	11
5.2 SPREMENLJIVKE.....	13
5.3 PRETVARJANJE TIPOV.....	14
5.4 NASLAVLJANJE MERILNO KRMILNEGA VMESNIKA.....	15
5.6 OSNOVNI STAVKI JEZIKA BASIC.....	19
<b>6. PROGRAMIRANJE, ZAJEM IN OBDELAVA MERITEV.....</b>	<b>20</b>
<b>7. EKSPETIMENTI.....</b>	<b>22</b>
7.1 SVETLEČA DIODA.....	22
7.2 PRAZNIJENJE KONDENZATORJA.....	22
7.3 ANALIZA PROSTEGA PADA.....	22

## 1. Uvod v računalniško krmiljenje

Na spletni strani [www.e-prolab.com/comlab](http://www.e-prolab.com/comlab), verzija SI, produkti, tečaji izberete Uvod v računalniško krmiljenje - delavnica. Pogledamo si poglavja:

- 1 Desetiški in dvojiški zapis
- 2 4-bitni digitalni izhod
- 3 Frekvenca utripa
- 4 Pulzno-širinska modulacija
- 5 4-bitni digitalni vhod
- 6 Digitalni vhod kot štoparica, odzivni čas človeka
- 7 Uporaba svetlobnega merilnika povezanega z digitalnim vhodom
- 8 Primer analognega izhoda
- 9 Napetostno območje in ločljivost analognega izhoda
- 10 Primer analognega vhoda
- 11 Analogni vhod, območje in ločljivost
- 12 Umerjanje analognih merilnikov
  - 12.1 Umerjanje potenciometra
  - 12.2 Umerjanje NTC termistorja
  - 12.3 Umerjanje temperaturnega merilnika

Na primerih se seznanite z uporabo merilno krmilnega vmesnika CMC-S3, osnovna izvedba. Uporabljate zbirko za eksperimente iz elektrike imenovano Elektra. Tako spoznate uporabnost digitalnih in analognih vhodov in izhodov.

### 1.1 Testni program: Eprolab:

Pred vsako vajo je smiselno preveriti ali smo pravilno vezali/uporabili digitalne/analogne vhode/izhode. Prav v ta namen je bil izdelan program Eprolab, ki ga najdete na gumbu start-programi-Eprolab-eprolab, s katerim lahko hitro in enostavno preverite delovanje merilno krmilnega vmesnika in priključenega vezja.

Z miško (ali tipkovnico lahko določimo stanja izhodov, stanje vhodov pa je grafično in številsko prikazano na zaslonu ves čas izvajanja programa.

#### 1.1.1 Digitalni izhod

Posamezen bit digitalnega izhoda je vir napetosti z le dvema možnima izhodnima napetostma - nižji napetosti v digitalnem sistemu pripišemo logično 0 in višji napetosti logično 1. Stanje vsakega od osmih bitov je neodvisno od stanj ostalih bitov, zato imamo 256 možnih kombinacij (od 0 do 255). Stanja posameznih bitov digitalnega izhoda spremenimo tako, da kliknemo eno od tipk, nad katerimi je oznaka  $D_0$ , kjer o pomeni output ali izhod. Biti so označeni na samih tipkah -  $D_0$  je najnižji bit do  $D_7$ , ki je najvišji bit. Logično stanje simbolizira slika svetleče diode na tipki. V stanju 1 je rdeča in v stanju logične 0 sive barve. Na levi strani vsakega bita je izpisana desetiška vrednost, ki jo k osem-bitni besedi prispeva posamezen bit. Pod tipkami je izpisana vsota prispevkov posameznih bitov.

Stanje posameznega bita preverimo z voltmetrom (merimo med opazovanim bitom  $D_0$  in ozemljitvijo GND) ali s svetlečimi diodami. V stanju 1 je napetost med priključkom izbranega bita in ozemljitvijo okoli 4,98V, v stanju logične 0 pa med 0V.

**Vaja 1.1:** Poženite testni program za vmesnik in z voltmetrom izmerite napetosti logične 1 in logične 0 na bitih  $D_0$  in  $D_1$ .

**Vaja 1.2:** Sestavite električno vezje s svetlečimi diodami, da bodo kazale logično stanje vseh osmih bitov digitalnega izhoda. ne pozabite na zaščitne upore (okoli  $300\Omega$ ), ki jih vežete zaporedno s svetlečimi diodami. Katere so desetiške vrednosti števil na digitalnem izhodu, da hkrati sveti le ena svetleča dioda.

#### 1.1.2 Analogni izhod

Analogni izhod temelji na osem-bitnem digitalno analognem pretvorniku (DAC). na razpolago imate 2 kanala. Njegova izhodna napetost ima torej 256 nivojev med 0,0 V in 5,0 V, s korakom  $5V/255=19,6$  mV (tudi 0V je stanje, zato  $256-1=255$ ). Kanal 2 pa ima razpon od -9,98 V do +9,98 V. Dvo-kanalni analogni izhod simbolizirata dva navpična drsnika, nad katerima je oznaka  $V_{out}$ . Z nastavitvijo višine drsnika določimo izhodno napetost, ki je zaradi preverjanja tudi izpisana pod drsnikom. Napetost analognega izhoda izmerimo z voltmetrom in izmerjeno vrednost primerjamo s tisto, ki jo izpiše program. Največje pričakovano odstopanje je  $\pm 0,02$  V.

**Vaja 1.3:** Z voltmetrom izmerite napetosti (za različne pozicije drsnikov) obeh analognih kanalov in jo primerjajte s tisto, ki je izpisana pod drsnikom.

**Vaja 1.4:** Na kanal 1,2 priklopite žarnico. Spreminjajte lego drsnika in jo opazujte.

### 1.1.3 Digitalni vhod

Digitalni vhod ugotovi logično stanje napetosti, ki jo priključimo med izbrani bit in ozemljitev. Praviloma naj bi bila napetost na digitalnem vhodu v digitalni obliki, torej z dvema diskretnima stanjema logične 0 in logične 1. Za vmesnika CMC velja, da napetost nad 2,7 V predstavlja logično 1 in pod 2,3 V logično 0. Uporaba digitalnega vhoda je zanesljivejša, če je razlika med obema napetostnima večja - logična 0 čim bliže 0 V in logična 1 čim bliže 5 V.

Digitalni vhod predstavlja osem krogov, nad katerimi je oznaka  $D_i$ , i kot input ali vhod. Desno so izpisane oznake bitov od najnižjega spodaj ( $D_0$ ) do najvišjega bita na vrhu ( $D_7$ ). Modro obarvan krog pomeni, da je bit v stanju logične 1, sivo obarvan krog pa pomeni stanje 0. Levo je izpisana desetiška vrednost, ki jo k besedi prispeva posamezen bit. Pod krožci je izpisana vsota prispevkov posameznih bitov. Stanje digitalnega vhoda se preverja ves čas.

Najpreprostejši test je povezava izbranega bita digitalnega s priključkom +5V na vmesniku. Lahko bi tudi vezali baterijo za 4.5 V (- pol na GND, + pol na izbrani bit).

**Vaja 1.5:** Povežite vhode za posamezne bite digitalnega vhoda s priključkom +5V ali GND na vmesniku, tako, da bo vsota prispevkov posameznih bitov (izpisana na zaslonu) enaka 16, 32, 64 in 111.

### 1.1.4 Analogni vhod

Analogni vhod pretvori (ADC) zvezno spreminjajočo vhodno napetost v številsko vrednost zapisano v digitalni obliki. Območje kanalov  $V_{in0}$  do  $U_{in5}$  je med 0,0 V in 4,98 V, na kanalih  $U_{in6}$  in  $U_{in7}$  lahko merimo tudi izmenične napetosti (-9,98 V do +9,98V). Vhodno napetost posameznih analognih vhodov ponazarja višina osmih stolpcev zelene barve. Nad stolpci je oznaka vhoda  $V_{in}$ , pod stolpcem pa je izpisana trenutna napetost. Napetost vseh analognih vhodov se stalno preverja (vsake 0,2 s).

**Vaja 1.6:** Preverite analogna vhoda tako, da ju povežite z analognima izhodoma. Spreminjajte analogna izhoda in opazujte kaj izmerite z analognima vhodoma.


**Vaja 1.7:** Vežite vezje s spremenljivim uporom in svetlečo diodo, ter izmerite padec napetosti na svetleči diodi.


## 2. MERJENJE TOKOVNO NAPETOSTNIH KARAKTERISTIK


Za merjenje tokovno napetostnih karakteristik uporov, uporovnih senzorjev in polprevodniških diod uporabite program UI karakteristike.


Elemente vežemo v tri različna vezja, katerih skico vezave priključimo znotraj programa. Karakteristike uporov nad 100  $\Omega$  in polprevodniških diod merimo preko delilnika napetosti. Za merjenje karakteristik uporov z uporabnostjo od 1  $\Omega$  do 100  $\Omega$  uporabimo močnostni ojačevalnik.

### 2.1 Merjenje karakteristike uporov

Potrebujemo referenčni upor in zvežemo napetostni delilnik. Skico vezave dobite v pogovornem oknu, s pritiskom na tipko . Dejansko vrednost referenčnega upora vpišite v ustrezno okence in zaprite pogovorno

okno. Meritev sprožite s pritiskom na tipko . Program med samim merjenjem povečuje napetost analognega izhoda v 40 korakih in sproti vrisuje točke (križci). Vsako merjenje zaznamuje s kratkotrajnim zvočnim signalom. Ko dokonča merjenje lahko spremenimo skalo napetosti in toka. Z izbirno tipko, ki se pojavi desno od grafa, zahtevamo izračun upornosti, čemur sledi risanje prilagoditvene premice izračunane po metodi najmanjših kvadratov. Nad grafom je izbirna tipka, s katero lahko izključimo ali vključimo zvočni signal ali pa zahtevamo povezovanje merjenih točk s črto. Sledenje v primeru uporov ni priporočeno, če pa za dano meritev zahtevamo izračun upornosti pa povezovanja med točkami ne more biti. Z menijskima tipkama lahko določimo količino, ki je na abscisi oz. ordinati. Če imamo napetost na ordinati in tok na abscisi, predstavlja strmina premice upornost.

Na enem grafu lahko narišemo do šest karakteristik, če jih želimo brisati pritisnemo tipko . Pogovorno okno s tabelo vseh meritev odprete z menijsko tipko. Odločite se za prikaz ene ali več krivulj. Vrednosti zapisanih v tabeli ne moremo spreminjati, lahko pa jih posnamemo v odložišče (clipboard), da jih kasneje prilepimo (paste) v kak drug program (Excel). Tabela lahko zapišemo tudi v tekstovno datoteko v kateri ločuje stolpce znak TAB.

**Vaja 2.1:** Izberite enake upore večje od  $300\Omega$ , ter referenčni upor  $1k\Omega$ . Izmerite upornost 1. upora, 2. uporov vezanih zaporedno, 2. uporov vezanih vzporedno, 3. uporov  vezanih zaporedno in 3 uporov vezanih vzporedno. Vse krivulje naj bodo narisane istočasno, uporabite opcijo za prikaz izračunane vrednosti. Meritve shranite v datoteko pod imenom upor.txt v vašo mapo.

**Vaja 2.2:** Izmerite človeško upornost med dvema prstoma, na isti roki, različnih rokah, suhi/mokri prsti. Meritve shranite v datoteko pod imenom CR.txt v vašo mapo.

## 2.2 Merjenje karakteristike diod v prevodni smeri

Postopek merjenja polprevodniških diod je enak kot pri uporih. Program sam ugotovi, da gre za diodo zato tudi ustrezno izračuna prilagoditveno funkcijo in ne izračuna upornosti, temveč določi vrednost kolenske napetosti.

**Vaja 2.3:** Izmerite karakteristiki rdeče in zelene (rumene) svetleče diode. Katera dioda ima večjo kolensko napetost ali oddaja le ta svetlobo s krajšo valovno dolžino? Meritve shranite na odlagališče.

## 3. RAČUNALNIŠKI OSCILOSKOP - HiSkop

Izbiramo lahko ned eno in dvokanalnim merjenjem z nastavitvijo napetostno-časovne skale (0,1 ms/razdelek do 24h/razdelek). Nizi 512, 1024... do 16384 meritev so mogoči z enkratnim in ponavljajočim proženjem. Vzorčenje prožimo s tipko ali kliknemo na umb, lahko pa je zakasnjeno s čakanjem na spremembo podatka na digitalnem vhodu ali na znatno spremembo merjene napetosti.

**Vaja 3.1:** Uporabite Hiskop za opazovanje časovnega poteka ohlajevanja/ogrevanja termistorja. Preskus naredite tako da prestavite termistor iz čaše z vročo vodo v čašo z mrzlo vodo. Izmerjene rezultate shranite v datoteko in jih odprite v Excelu.

**Vaja 3.2:** Na spletni strani [www.e-prolab.com/comlab](http://www.e-prolab.com/comlab) najdete primer za tečaje pod elektrika in RLC vezja: praznjenje kondenzatorja. Uporabite Hiskop za zajem meritve praznjenja kondenzatorja. Izmerjene rezultate shranite v datoteko in jih odprite v Excelu.

## 4. Uporaba el. preglednic za prikaz in analizo meritev

### 4.1 Prenos podatkov

Podatke meritve lahko prenesemo v Excel preko

#### 4.1.1 Odložišča (*clipboard*)

V programu UI karakteristike kliknemo na gumb tabela, označimo zelene podatke in jih skopiramo (pritisek na gumb kopiraj) v odložišče. V delovnem listu Excela se postavimo v celico v katero se prenese zgornja leva celica tabele iz odložišča, in kliknemo na gumb prilepi (prilepi).

#### 4.1.2 Datoteko

Podatke imamo shranjene kot datoteka txt oblike. Pritisnemo na gumb odpri (open), poiščemo datoteko v ustrezni mapi, in na njej naredimo dvoklik. Če datoteke ne vidimo, ker nima končnice xls ali xlt, moramo spremeniti vrst datoteke (Files of tipe) na VSE (ALL).

### 4.2 Oblikovanje tabel

#### 4.2.1 Obrobe

Kar hočemo obrobiti moramo označiti, nato izberemo zelene obrobe v orodjarni na gumbu obrobe (Borders), če tam ni zelene obrobe, pogledamo še v menu-oblika-obrobe in senčenje.

#### 4.2.2 Številski formati

Da vidimo izmerjene vrednosti na toliko decimalnih mest kolikor je veljavnih uporabimo ustrezni številski format. Označimo celice v katerih želimo prirediti določen format in v menu-oblika-celice-številke določimo število decimalnih mest, pri tem odkljukamo piko (vejico) za tisočine.

#### 4.2.3 Enote

Pri merjenju imamo opravka s fizikalnimi veličinami, zato je meritev veličine enaka številu krat enota. Če v celico napišemo 3m/s, bo Excel to razumel kot tekst, zato dosežemo enote na drugačen način: Označimo, v menu-oblika-celice-števila (menu-format-cells-numbers) izberemo pod kategorijo po meri (custom) in v polje tip (type) vtipkamo številski format, za njem pa v dvojnih narekovajih enoto.

Za	enice uporabimo:	0
	decimalno ločilo:	.
	decimalna mesta:	0
	desetice, stotice in tisočice:	#
	ločilo za tisočine:	,
	poljuben tekst	"poljuben text"

Primer: število 123456.12345 želimo imeti izpisano na 2. decimalki z ločilom za tisočine. V polje za tip bi zapisali: #.##0,00 .

### 4.3 Računanje v Excelu

#### 4.3.1 Enačbe

V Excelu računamo kot s kalkulatorjem, le da začnemo z znakom za enačaj in ne končamo z njim kot na kalkulatorju. Najprej se postavimo v celico v kateri želimo imeti rezultat, vtipkamo enačaj, za katerim sledi

enačba. V enačbi nastopajo, konstante (3, -8, 1.23,...), spremenljivke (to so naslovi posameznih celic A1, D4...), operatorji (+,-,\*,/), funkcije (sin, cos,...) in pomožni operatorji (()).

Primer: 3 krat 5

Kliknemo na celico in vtipkamo: =3\*5 in pritisnemo tipko enter, ali z miško kliknemo na gumb z zeleno kljuko v vnosni vrstici.

### 4.3.2 Funkcije

Postavimo se na celico v kateri želimo imeti rezultat, v orodjarni poiščemo gumb čarodej za funkcije (paste function). V levem delu okna izberemo kategorijo funkcij (matematično trigonometrične, statistične...). Če ne vemo v katero kategorijo spada funkcija, ki jo iščemo, izberemo kategorijo vse (all). V desnem oknu (ime funkcije/function name) so funkcije sortirane po abecedi, izberemo in naredimo dvoklik. Če že poznamo njeno ime, kliknemo v celico, začnemo z enačajem, sledi ime funkcije v oklepajih pa operand.

1) Vsota

Najdemo jo pod imenom **SUM**, še krajše pa je, če kliknemo na gumb za vsoto v orodni vrstici, z miško označimo vse celice, katerih vsebina naj se sešteje.

2) Povprečje

**Average** nam vrne povprečje označenih celic, zato pazimo, da ne označujemo praznih.

3) Izračun koeficientov regresijske premice

Premica, ki se najbolj prilagaja skupini X-Y podatkovnih točk, je tista premica, ki minimizira vsoto kvadratov razdalj od izmerjene točke do premice v vertikalni ali Y smeri. Ta premica je znana kot regresijska premica, njena enačba pa se imenuje regresijska premica:

$$Y = kx + n \quad (4.1)$$

a) Izračun smernega koeficienta, k

$$k = \frac{N \sum XY - \sum X \sum Y}{N \sum X^2 - (\sum X)^2} \quad (4.2)$$

Kjer N pomeni število podatkov. Ta izračun nam v Excelu naredi funkcija:

**SLOPE**(odvisna, neodvisna) nam vrne vrednost smernega koeficienta (k) za linearno enačbo  $Y=kX+n$ . pri tem je X neodvisna spremenljivka in Y odvisna od X. Smerni koeficient lahko tako določamo za vse ali del točk (po prehodnem pojavu).

a) Izračun konstante, n

$$n = \frac{\sum Y}{N} - k \frac{\sum X}{N} \quad (4.3)$$

**INTERCEPT**(odvisna neodvisna) nam vrne vrednost konstante n, vrednost Y v presečišču krivulje z ordinato.

4) Izračun determinacijskega koeficienta

Determinacijski koeficient  $r^2$  je kvadrat korelacijskega koeficienta r in ga izračunamo po sledeči enačbi:

$$r^2 = \frac{n \sum Y + k \sum XY - NY^2}{\sum Y^2 - \bar{Y}^2} \quad (4.4)$$

V Excelu imamo za ta namen predvideno funkcijo **RSQ**(odvisna, neodvisna), ki nam pove, da je  $RSQ*100\%$  variabilnosti Y pojasnjeno z X. Pove nam koliko izmerjene vrednosti odstopajo od regresijske premice. Bliže ko je rezultat 1, manj so točke raztresene okoli prilagoditvene premice.

5) Standardna napaka ocene nam meri standardno vrednost za katero se dejanske vrednosti za Y se razlikujejo od ocenjenih vrednosti, ki jih izračunamo z linearno  $\hat{Y}$  regresijo:

$$s_{Y \cdot X} = \sqrt{\frac{\sum (Y - \hat{Y})^2}{N - 2}} \quad (4.5)$$

Izračunamo jo s pomočjo **STEYX**(odvisna, neodvisna).

### 4.3.3 Kopiranje

Kadar želimo kopirati formulo preko več vrstic v stolpcu, ki se držijo skupaj je najenostavnejši način ta, da se postavimo z miško na skrajni desni spodnji rob celice, kjer je majhen črni kvadrček, v kateri imamo napisano enačbo (seveda moramo biti na celici in ne v celici, ko nam utripa kurzor), takrat se nam spremeni puščica miške v majhen črni plus, v tem trenutku pritisnemo in držimo levi miškin gumb in potegnemo čez vse celice v katerih želimo imeti enako formulo. Ker vlečemo preko vrstic se v celicah spreminja vrstična koordinata celic, katerih naslovi so v enačbi. na ta način lahko z vlačanjem kopiramo, navzgor, navzdol, levo in desno. V primeru, da želimo, da se pri kopiranju formul, določene koordinate celic ne spreminjajo uporabimo absolutno naslavljanje celic, kar pomeni, da pred vrstično ali stolpčno oznako pišemo simbol \$, naprimer =A3\*\$B\$6, pri čemer smo fiksirali koordinato celice B6, ker kopiramo navzdol je dovolj samo simbol pred vrstično oznako.

### 4.3.4 Tvorba zaporedij

Kadar potrebujemo zaporedje števil (celih ali decimalnih) 1,2,3,..., naredimo zaporedje, tako da v celico vpišemo število zaporedja, v sosednjo celico (spodaj, zgoraj, levo ali desno) pa naslednje število zaporedja. Obe celici označimo. Z miško se postavimo na skrajno spodnji desni kot, kjer je majhen črni kvadrček, v tem trenutku se spremeni puščica v črni plus, pritisnemo in držimo levi gumb in potegnemo čez vse celice v katerih želimo imeti zaporedje.

### 4.3.5 Nelinearne prilagoditvene funkcije

Kadar je zveza med odvisno in neodvisno spremenljivko nelinearna, jo dostikrat lahko z uvedbo nove spremenljivke lineariziramo. S naj bo pot, ki jo v času t prepotuje enakomerno pospešeno gibajoče telo, z začetno hitrostjo enako nič:

$$s = \frac{at^2}{2} \quad (4.6)$$

Uvesti moramo tako novo spremenljivko, da bo odvisnost nove spremenljivke od časa linearna ne pa kvadratična. To dosežemo, če korenimo levo in desno stran enačbe:

$$\sqrt{s} = \sqrt{\frac{a}{2}}t \quad (4.7)$$

in uvedemo novo spremenljivko  $x = \sqrt{s}$ . Na ta način dosežemo linearno odvisnost med novo odvisno spremenljivko x in časom t. Meritve poti v določenih časovnih trenutkih so obremenjene z motnjami iz okolja in nepravilnostmi merilne metode. Zato je težko določiti pospešek. V primeru linearizacije, lahko ugotovimo linearno zvezo z regresijsko premico tako določimo pospešek z najmanjšo napako.

## 4.4 Kreiranje grafa

Označimo podatke za X in Y os (če se podatki ne držijo skupaj uporabimo tipko CTRL, tako, da označimo podatke za x os, pritisnemo tipko CTRL in jo držimo ter označimo še podatke za y os), nato kliknemo na gumb Čarodej za grafikone (Chart wizard) in sledimo korakom:

Izberemo zvrst grafikona, podzvrst, če v tretjem koraku vidimo pričakovani graf gremo naprej, sicer preverimo kako je Excel predpostavil nastavitve za naše podatke. V 4 koraku imenujemo graf, osi,... V zadnjem koraku izberemo kot objekt na ta list ali kot nov list in pritisnemo gumb finish.

### 4.4.1 Oblikovanje grafikona

Na splošno se da vsak element (naslov, os, mrežne črte...) spreminjati (oblika, barva, pozicija...). Pravilo je: element označimo (seveda smo znotraj grafikona in ne na listu), puščico (miškino) imamo na označenem

elementu in kliknemo desni miškin gumb, odpre se nam priročni menu, na katerem najdemo vse v zvezi z označenim elementom. Še krajša pot je dvoklik na izbrani element.

#### **4.4.2 Vstavljanje trenda**

V grafu lahko skozi izmerjene točke potegnemo aproksimacijsko prilagoditveno funkcijo. V primeru da smo odvisnost spremenljivk linearizirali, lahko v graf dodamo linearno prilagoditveno funkcijo ali trend. V grafu označimo (kliknemo na točko) točke, pritisnemo desni miškin gumb in iz priročnega menija izberemo: dodaj trend... (add trendline...). na lističu tip (type) izberemo linearni, na lističu opcije (options) pa lahko določimo še izpis na grafu za enačbo za trenda in determinacijskega koeficienta.

#### **4.4.3 Dodajanje grafa na že obstoječi grafikon**

Kadar želimo v že obstoječi grafikon dodati še eno krivuljo to naredimo, tako da se z miško postavimo na grafikon (element je področje grafikona) oziroma, da je označen celoten grafikon, s klikom na desni miškin gumb izberemo iz priročnega menija izvorni podatki (source data) in kliknemo na listič serije (series). Tu s klikom na gumb dodaj (add), nato pa določimo obseg celic za odvisno (Y os) in neodvisno (X os) spremenljivko ter s pritiskom na gumb v redu dodamo novo krivuljo.

### **4.5 Spreminjanje decimalnega simbola**

#### **4.5.1 V datoteki txt**

Če imamo nastavljeno decimalno piko (vejico), kot decimalni simbol v Excelu, v tekstovni datoteki pa so zapisana števila z decimalno vejico(piko), moramo preden odpremo datoteko v Excelu spremeniti v datoteki decimalna ločila. To storimo s pomočjo Worda. V njem odpremo datoteko, v meniju izberemo ukaz Spremeni (Replace) in vtipkamo v polje najdi vejico (piko), v polje zamenjaj z pa piko (vejico), nato pritisnemo na gumb zamenjaj vse. Datoteko shranimo in jo nato odpremo v Excelu. Pozor, če so števila zapisana tudi s piko (vejico) za tisočine, potem dobimo pri taki zamenjavi popolno neumnost. V tem primeru naredite najprej zamenjavo pike (vejice) z neko črko, nato vejice (pike) s pikami (vejicami), in nazadnje še črko, ki ste si jo izbrali za vejico (piko).

#### **4.5.2 V operacijskem sistemu**

Decimalno ločilo, ki je nastavljeno v operacijskem sistemu, ponavadi uporabljajo vsi programi, ki so v njem zagnani (vsaj Microsoftovi). Tako lahko v primeru, ko imamo v operacijskem sistemu nastavljen decimalni simbol piko (vejico), v tekstovni datoteki pa decimalni simbol vejico(piko) naredimo spremembo decimalnega simbola v operacijskem sistemu. Kliknemo na gumb START, izberemo nastavitve (Settings), kontrolna plošča (Control Panel), področje nastavitvev (Regional Settings), izberemo listič števila (Numbers) in kliknemo v polje decimalni simbol, zberemo piko(vejico) in vtipkamo vejico(piko), ter potrdimo vnos s pritiskom na gumb V redu (OK). Če imate Excel zagnan ga zaprite in ponovno zaženite, kajti šele sedaj bo začel uporabljati nastavljen simbol.

## 5. PROGRAMIRANJE računalniško merilnega krmilnega sistema cmc

Programsko orodje Visual basic najdete na gumbu start-programi-visual basic. Ko odprete program si najprej izberete v oknu New project na lističu New opcijo Standard EXE in odpre se vam Visual basic. Zgoraj najdete menijsko vrstico in orodjarno. Na levi strani se nahaja paleta splošnih sestavin ali obrazcev ali komponent. Programirati začnete tako, da nanašate sestavine na obrazec Project1-Form1. Lastnosti sestavin vidite v oknu Properties-form na vaši desni strani. Drugo okno Project1-Code, v katero vpisujete kodo, dosežete, če izberete v zgornjem oknu na desni strani imenovanem Project, gumb View code. Nazaj na prvo okno Form pridete s klikom na sosednji gumb View Object.

**A)** Predno začnete programirati vedno najprej shranite vaš program v neko vašo mapo (meni-file-save project). Visual basic namreč ustvari kar nekaj datotek.

**B)** Za programiranje merilno krmilnega vmesnika so pripravljene procedure in funkcije, ki jih dosežete tako, da v vaš program dodate modul (meni-project-add module): kliknite na listič Existing in na m disku v mapi ORobotika\VB poiščite datoteko SysIOLib1.bas.

**C)** Še predno začnete s programiranjem poskrbite za inicializacijo merilno krmilnega vmesnika. To storite tako, da v oknu Form1 naredite kjerkoli na sivi površini (to je sestavina Form1) dvoklik z miško. Odpre se kodno okno in generira začetek in konec procedure Form\_Load:

```
Private Sub Form_Load()
```

```
End Sub
```

Ko zaženete program je to prva procedura, ki se izvrši. V njej zato vedno določite izhodiščne vrednosti spremenljivk, postavite vse analogne na 0 V in digitalne izhode na logično 0. Med obe vrstici tudi vtipkajte InitIOSys. Ta procedura prepozna tip vhodno/izhodnega sistema in opravi nekatere druge potrebne postopke. Tip vmesnika zapiše v spremenljivko IOSys, ki tako dobi eno od treh možnih oznak CMC\_S3, H\_bridge ali None.

### 5.1 Osnovne sestavine

Vsaka sestavina ima lastnosti in dogodke, ki jih lahko kličemo v zvezi z njo. V nadaljevanju si pogledjmo si najbolj osnovne sestavine in njihove najbolj osnovne lastnosti in dogodke.

#### 5.1.1 Command ali Gumb



Uporaba: za nadzor nad potekom programa.

Lastnosti:

Name ali naslov je ime sestavine s katerim se nanjo sklicujemo. V osnovi se vedno imenuje Command1, pri čemer se število na koncu poveča za eno za vsak novi gumb na obrazcu ali formi. Ime sestavine spremenite v oknu Properties-name.

Caption ali naslov je besedilo, ki se izpiše na gumbu. Caption določimo tako, da označimo dotični gumb, v oknu Properties najdemo lastnost caption in kliknemo v sosednje, desno polje ter vnesemo novo naslov. V programu lahko kadarkoli spremenimo napis na gumbu imenovanem command1 z ukazom: command1.caption="nov naslov".

Dogodki:

Click je najbolj pogost dogodek v zvezi z gumbom. Zgodi se ko 1x kliknemo na gumb z levim miškinim gumbom. Dogodek dosežemo najenostavneje tako, da naredimo dvoklik na gumb. Odpre se kodno okno in v njem generira koda

```
Private Sub Command1_Click()
```

```
End Sub
```

Vse ukazne vrstice znotraj tega podprograma (Subrutine) se izvršijo ob kliku na dotični gumb.

### 5.1.2 Label ali nalepka



Uporaba: za izpisovanje besedila ali vrednosti spremenljivk na zaslonu.

Lastnosti:

Name je v osnovi Label1.

Caption ali naslov je besedilo, ki se izpiše v nalepki. Caption določimo tako, da označimo dotično nalepko, v oknu Properties najdemo lastnost caption in kliknemo v sosednje, desno polje ter vnesemo novo naslov. V programu lahko kadarkoli spremenimo besedilo nalepke imenovane label1 z ukazom: `label1.caption="novo besedilo"`.

Dogodki:

V zvezi z nalepko obstaja več dogodkov, vendar pa nas nobeden od njih za naše namene ne zanima.

### 5.1.3 TextBox ali vnosno polje



Uporaba: za vnos besedila ali številčnih vrednosti tekom izvajanja programa

Lastnosti:

Name je v osnovi TextBox1.

Text ali besedilo je besedilo, ki ga lahko uporabnik tekom izvajanja vpiše v vnosno polje. Na ta račun je omogočena interakcija z uporabnikom, ki lahko določi naprimer nove vrednosti spremenljivk... V programu lahko kadarkoli beremo kar je zapisano v vnosnem polju z ukazom: `A=textBox1.text`, pri čemer je A spremenljivka tipa string.

Dogodki:

V zvezi z vnosnim poljem obstaja več dogodkov, vendar pa nas nobeden od njih za naše namene ne zanima.

### 5.1.4 Timer ali časovnik



Uporaba: za ponavljajoče izvajanje določenega dela programa.

Lastnosti:

Name je v osnovi Timer1.

Enabled ali omogočen je lastnost časovnika s katero ga lahko omogočimo ali izklopimo. V oknu Properties lahko izberete dve možnosti True ali False. Če bo nastavljena na True pomeni, da bo časovnik omogočen in se bodo vse ukazne vrstice zapisane v njegovem dogodku imenovanem Timer izvrševale na vsakih Interval ms. V programu ga lahko kadarkoli izklopimo ali vklopimo z ukazom: `Timer1.enabled=False` ali `Timer1.enabled=True`.

Interval je vrednost v mili sekundah, ki določa na vsake koliko časa se izvrši dogodek Timer, če je le ta omogočen. Določimo ga v oknu Properties ali v programu z ukazom `Timer1.interval=1000`.

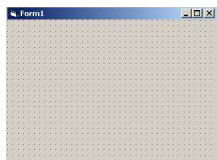
Dogodki:

Timer je edini dogodek v zvezi s časovnikom. Vse kodne vrstice zapisane v proceduri Timer1\_Timer se izvršijo na vsake Interval ms, če je časovnik omogočen.

```
Private Sub Timer1_Timer()
```

```
End Sub
```

### 5.1.5 Form ali obrazec



Uporaba: je temelj za naš program in je obvezen.

Lastnosti:

Name je v osnovi Form1.

Dogodki:

Load je prvi dogodek, ki se izvrši ko zaženete program, zato v njem določite vse začetne nastavitve, kot so vrednosti spremenljivk, digitalnih izhodov, analognih vhodov...

```
Private Sub Form_Load()
```

```
End Sub
```

Terminate je dogodek, ko kliknete na x zgoraj desno v oknu Form in ga s tem zaprete. Takrat je priporočljivo postaviti vse digitalne izhode na logično 0 in analogne izhode na 0V.

```
Private Sub Form_Terminate()
```

```
End Sub
```

## 5.2 Spremenljivke

Spremenljivka je referenca na določeno mesto pomnilnika, kjer se nahaja njena vrednost. Ta vrednost se lahko spreminja tekom izvajanja programa zato tudi takšno ime. Vse spremenljivke je potrebno pred uporabo deklarirati kar pomeni določiti njeno ime in tip. Ime je lahko poljubno. Izogibamo se šumnikov in sičnikov, prepovedani znaki so %,;, in podobni. Prav tako ne uporabljamo za imena spremenljivk rezervirana imena kot naprimer Command1.

Osnovni tipi spremenljivk:

### 5.2.1 Byte

Spremenljivka tipa byte lahko zavzame pozitivna, cela števila v rangu [0, 255].

Deklaracija:

```
Dim A as Byte
```

Primer:

A=12

### 5.2.2 Integer

Spremenljivka tipa Integer lahko zavzame pozitivna, cela števila v rangu [-32768, 32767].

Deklaracija:

Dim A as Integer

Primer:

A=-12

### 5.2.3 Single

Spremenljivka tipa Single lahko zavzame realna števila v rangu [-3.402823E38, 3.402823E38].

Deklaracija:

Dim A as Single

Primer:

A=-12.1234

### 5.2.4 String

Spremenljivka tipa String je znakovna spremenljivka in ji lahko priredimo besedilo (največ 64K).

Deklaracija:

Dim A as String

Primer:

A="poljubno besedilo"

### 5.2.5 Boolean

Spremenljivka tipa Boolean je logična spremenljivka in lahko zavzame le dve možni vrednosti True ali False.

Deklaracija:

Dim A as Boolean

Primer:

A=True

### 5.2.6 Deklaracija spremenljivk

Spremenljivke deklariramo čisto na začetku, na vrhu v kodnem oknu. Kadar deklariramo več spremenljivk ne podvajamo besede Dim.

Primer:

Dim A as Integer, B, C as Single, D, E, F as Boolean

## 5.3 Pretvarjanje tipov

Za izpis vedno uporabljamo sestavino Label, le ta je tipa string. Poglejmo si kako izpišemo različne tipe spremenljivk v Label.

### 5.3.1 Celo število v string

A naj bo celo-številskega tipa (npr. Dim A as Byte) in naj ima vrednost A=55.

Primer:

```
Label1.caption=str(A)
```

Načeloma deluje tudi brez funkcije STR, ki vam pretvori število v besedilo.

### 5.3.2 Realno število v string

A naj bo realno število (Dim A as Single) in naj ima vrednost A=55.123456789.

Primer:

```
Label1.caption=str(A)
```

Če bi želeli izpis samo na določeno število decimalnih mest uporabimo funkcijo FORMAT().

Primer:

```
Dim A as Single, B as string  
A=55.123456789
```

```
B=format(A, "0.00")
```

Na zaslonu se bo izpisalo 55.12.

### 5.3.3 Boolean v string

A naj bo logičnega tipa ( Dim A as Boolean) in naj ima vrednost A=true.

Primer:

```
Label1.caption=str(A)
```

Na zaslonu se bo izpisalo True.

### 5.3.4 String v število

A naj bo znakovnega tipa (Dim A as String) in naj ima vrednost A="123.12345", b pa številkega tipa real (Dima B as single). Ko uporabljate sestavino TextBox in vanj vpišete 123.12345 je to besedilo in ne številka, zato lahko priredite vrednost TextBox.text samo spremenljivki tipa string: A= TextBox.text. Ko želite uporabiti to kot številsko vrednost jo morate predhodno pretvoriti v število. To naredimo z uporabo funkcije VAL()

Primer:

```
B=VAL(A)
```

Vrednost spremenljivke B je enaka 123.12345.

## 5.4 Naslavljanje merilno krmilnega vmesnika

Kot ste spoznali ima merilno krmilni vmesnik digitalne in analogne vhode in izhode. Vseh se lahko programsko poslužujemo. V nadaljevanju si pogledjmo naslavljanje posameznih vhodov in izhodov s praktičnimi primeri.

### 5.4.1 Digitalni izhod

Na razpolago imamo več možnosti.

#### a) DigOut digX, N

Kjer je  $X = \{A, B, C, D\}$  in določa katere izhode bomo naslavljali.,  $N$  pa je pozitivno celo število od 0 do 255 (imamo 8 bitni digitalni izhod  $2^8=256$ ). S tem ukazom naslavljamo vse bite digitalnega izhoda digA (digB, digC ali digD) hkrati.

Primer:

Prve štiri bite (Do(0), Do(1), Do(2) in Do(3)) digC želimo imeti v logični ničli, višje 4 (Do(4), Do(5), Do(6) in Do(7)) pa v logični enici. Nasloviti jih moramo z binarnim številom 11110000. Desetiška vrednost je enka 240, tako je ukaz enak:

DigOut digC, 240

#### b) BitOut DigX, DM, Stanje

Kjer je  $X = \{A, B, C, D\}$  in določa katere izhode bomo naslavljali.,  $M$  določa bit stanje katerega želimo določiti in je lahko število od 0 do 7 (imamo 8 bitni digitalni izhod ). Stanje = {BitOn, BitOff}. BitOn pomeni, da je v logični enici, BitOff pa v logični ničli. S tem ukazom naslavljamo le en bit digitalnega izhoda digA (digB, digC ali digD).

Primer:

Peti bit Do(4) (štejemo od nič navzgor, zato 4), digitalnih izhodov DigD želimo postaviti v logično enico. Ukaz je enak:

BitOut digC, 4, BitOn

#### c) DirOut DigX, DN, DM, Stanje

Kjer je  $X = \{A, B, C, D\}$  in določa katere izhode bomo naslavljali.,  $N$  in  $M$  sta bita na katera je vezan naprimer enosmerni električni motor  $d$  in sta števili od 0 do 7 (imamo 8 bitni digitalni izhod ). Stanje = {DirPlus, DirMinus in DirOff}. DirPlus pomeni, da je bit  $N$  v logični ničli, bit  $M$  pa v logični enici. DirMinus pomeni, da je bit  $N$  v logični enici, bit  $M$  pa v logični ničli. DirOff pa postavi oba ( $N$  in  $M$ ) bita v stanje 0. Ukaz uporabljamo predvsem za krmiljenje enosmerne motorjev; zavrtimo ga lahko v eno stran v drugo stran ali pa zaustavimo.

Primer:

Enosmerni motor vezemo na prva dva bita (0 in 1) digitalnih izhodov digC. Želimo ga zavrteti levo. Ukaz je enak:

DirOut DigC, D0, D1, DirPlus

Če se motor zavrti v desno stran, potem zamenjamo priključka na motorju ali pa uporabimo stanje DirMinus.

**Vaja 5.1:** Sestavite program, ki bo na vrhu ekrana napisal DIGITALNI IZHOD. V polje zraven katerega bo pisalo: Podatek, boste vpisali desetiški podatek za 8 bitno besedo (0-255). S pritiskom na gumb IZHOD naj računalnik pošlje podatek digitalnemu izhodu.

**Vaja 5.2:** Uporabite digitalni izhod za demonstracijo dvojiškega štetja od 0 do 15. Na digitalnem izhodu naj se vrednosti menjajo vsake pol sekunde. Dogajanje opazujte na digitalnih izhodih  $D_0(0)$  do  $D_0(3)$ , na katere vežite svetleče diode.

**Vaja 5.3:** Vpišite mesec svojega rojstva (številke od 0 do 12). Če ste rojeni med januarjem in junijem, naj se prižge bit  $D_0(1)$ , če med julijem in decembrom pa bit  $D_0(1)$ .

**Vaja 5.4:** S tremi svetlečimi diodami ponazorite delovanje prometnega semaforja. Bit  $D_0(0)$  naj predstavlja rdečo, bit  $D_0(1)$  oranžno in bit  $D_0(2)$  zeleno luč. Obrnite vezje tako, da bodo diode v navpični smeri in napišite program, ki ponavlja sledeče zaporedje:

- Ⓐ pet sekund naj gori rdeča luč
- Ⓑ pet sekund naj gori zelena luč
- Ⓒ pol sekunde naj gori zelena in oranžna luč,

dokler ne boste pritisnili gumba konec.

#### 5.4.1 Digitalni vhod

Na razpolago imamo dve možnosti.

##### a) DigIn(digX)

Kjer je  $X = \{A, B, C, D\}$  in določa katere vhode bomo brali. S tem ukazom beremo vse štiri bite digitalnega vhoda digA (digB, digC ali digD) hkrati. Ukaz je funkcija in vrne število med 0 in 15 (imamo 4 bitni digitalni vhod  $2^4=16$ ), zato je potrebno vrednost prirediti neki spremenljivki.

Primer:

Berimo digitalne vhode digC. Predhodno deklarirajmo še spremenljivko A tipa byte, ki ji bomo priredili desetiško vrednost digitalnih vhodov. Ukaz je enak:

Dim A as byte

A=DigIn(digC)

Če sta prva dva bita 0 in 1 digitalnih vhodov digC ozemljena, bita 2 in 3 pa povezana na napetost +5V, so vhodi v binarnem zapisu enaki 1100, desetiško pa 12, torej  $A=12$ .

##### b) BitIn(digX, DN)

Kjer je  $X = \{A, B, C, D\}$  in določa katere vhode bomo brali, N pa je pozitivno celo število med 0 in 3 (imamo 4 bitni digitalni vhod). S tem ukazom beremo le stanje enega bita digitalnih vhodov digA (digB, digC ali digD). Ukaz je funkcija in vrne vrednost logične spremenljivke True ali False. True pomeni, da je vhod v logični enici. Vrednost funkcije lahko priredimo neki spremenljivki.

Primer:

Berimo le drugi bit (1, ker štejemo od nič navzgor) vhodov digC. Predhodno deklarirajmo še spremenljivko B tipa boolean, ki ji bomo priredili stanje bita. Ukaz je enak:

Dim B as boolean

B=BitIn(digC,D1)

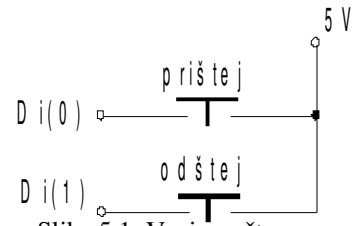
Če je bit 1 digitalnih vhodov ozemljen je  $B=False$ . Ponavadi želimo, da se glede na stanje bita (naprimer, da je vezano na ta bit stikalo) nekaj izvrši, izpiše... Uporabimo If Then stavek:  
If B=True Then label1.caption="logična enica"

Zapišemo lahko tudi:

If BitIn(digC,D1)=True Then label1.caption="logična enica"

Na zaslonu se izpiše logična enica, če je bit v logični enici (stikalo je sklenjeno).

**Vaja 5.5:** Zvežite vezje na sliki 5.1 in sestavite program, ki bo na zaslonu izpisalo število (začetno naj bo 0), če boste pritisnili na tipko prištej, se naj število poveča za eno, če pa boste pritisnili na tipko odštej, naj se število zmanjša za eno.



Slika 5.1: Vezje za števec.

**Vaja 5.6:**

Sestavite program, ki bo preverjal, če so logična stanja bitov  $D_0$ ,  $D_1$ ,  $D_2$  in  $D_3$ , vsa v stanju logične 1, vsa v stanju logične 0 ali so v različnih stanjih. Na zaslonu se naj izpiše:

- vsa tri stanja so 1
- vsa tri stanja so 0
- logična stanja so različna

### 5.4.1 Analogni izhod

Na razpolago imamo dve možnosti.

#### a) AnOutI K, N

Kjer je  $K = \{1, 2\}$  in določa kateri analogni kanal naslavljamo  $V_{out1}$  ali  $V_{out2}$ ,  $N$  pa je celo število od 0 do 255 (8 bitni DAC). Napetost na  $V_{out1}$  je linearno odvisna od števila  $N$  in sicer od 0 do 5 V,  $V_{out2}$  pa od -9.8 V do +9.8 V. S tem ukazom naslavljamo napetost na analognih izhodih v korakih. Za prvi kanal je 1 korak enak  $1 * 5 V / 255 = 16,6 mV$ .

Primer:

Na analognem izhodu, kanal 1 želimo dobiti 3 V. Ukaz je enak:

AnOutI 1, 153

Ker je  $153 * 5 V / 255 = 3 V$ .

#### b) AnOut K, R

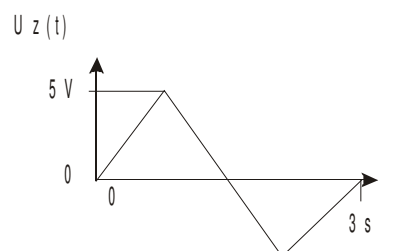
Kjer je  $K = \{1, 2\}$  in določa kateri analogni kanal naslavljamo  $V_{out1}$  ali  $V_{out2}$ ,  $R$  pa je napetost med 0 V in 5 V (realno število) za kanal 1 in med -9,8 V in 9,8 V za kanal 2.

Primer:

Na analognem izhodu, kanal 1 želimo dobiti 3,1 V. Ukaz je enak:

AnOut 1, 3.1

**Vaja 5.7:** Sestavite vezje na sliki 5.2, in programirajte alarmno napravo na sledeč način: Alarm vklopite s klikom na gumb vklop alarma. Če pride do sprožitve alarma (prekinitev infrardečega snopa), naj začne 1 minuto utripati žarnica, in sicer naj utripa žagasto, s periodo 3s. Da je prišlo do vloma se naj izpiše tudi na zaslonu. Po preteku 1 minute naj se ugasne žarnica in alarma naprava pa naj se postavi v začetno stanje. Če hočemo izklopiti alarm kliknemo na gumb izklop alarma.



Slika 5.2: Napajalna napetost žarnice.

### 5.4.1 Analogni vhod

Na razpolago imamo dve možnosti.

#### a) AnInI (K)

Kjer je  $K = \{0, 1, 2, 3, 4, 5, 6, 7\}$  in določa kateri analogni kanal beremo. Funkcija vrne število od 0 do 4095 (12 bitni ADC). Število je linearno odvisno od vhodne napetosti. Pri vhodih od 0 do 5 je območje med 0 in 5V, pri vhodih 6 in 7 pa od -9,8 do +9,8 V. Ukaz je funkcija in vrne število, zato je potrebno vrednost prirediti neki spremenljivki.

Primer:

Berimo analogni vhod, kanal 4. Predhodno deklarirajmo še spremenljivko N tipa integer, ki ji bomo priredili vrednost analognega vhoda. Ukaz je enak:

Dim N as integer

$N = \text{AnInI}(4)$

Če je na analognem vhodu, kanal 4 napetost 1 V, potem je  $N = 1 \text{ V} * 4095 / 5 \text{ V} = 819$ .

#### b) AnIn (K)

Kjer je  $K = \{0, 1, 2, 3, 4, 5, 6, 7\}$  in določa kateri analogni kanal beremo. Funkcija vrne napetost od 0 V do 5 V za vhode od 0 do 5 in od -9,8 do +9,8 V za vhoda 6 in 7. Ukaz je funkcija in vrne število, zato je potrebno vrednost prirediti neki spremenljivki.

Primer:

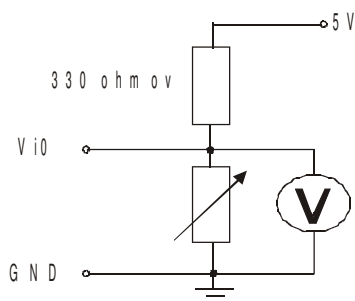
Berimo analogni vhod, kanal 4. Predhodno deklarirajmo še spremenljivko U tipa single, ki ji bomo priredili vrednost analognega vhoda. Ukaz je enak:

Dim U as single

$U = \text{AnIn}(4)$

Če je na analognem vhodu, kanal 4 napetost 1,1 V, potem je  $U = 1.1$ .

**Vaja 5.8:** Sestavimo program, ki bo oponašal digitalni voltmeter enosmerne napetosti. Na zaslonu naj se izpiše pritisnjena napetost med sponkama V<sub>i0</sub> in GND. Sestavite vezje na sliki 5.3, spreminjajte upornost dekad, napetost pomerite tudi z voltmetrom.



Slika 5.3: Vezje za digitalni voltmeter.

## 5.6 Osnovni stavki jezika Basic

### 5.6.1 Stavki IF THEN

S tem stavkom se sprašujemo ali je nek logični pogoj (LP) izpolnjen ali ne. Naprimer, da imamo dve spremenljivki tipa byte  $A=2$  in  $B=3$ , potem ju lahko med seboj primerjamo

$A>B$ ; LP ni izpolnjen

$A<B$ ; LP je izpolnjen

$A=B$ ; LP ni izpolnjen

V Basicu to naredimo z osnovnim IF THEN stavkom:

```
IF LP THEN s1
```

Kjer je s1 nek ukazni stavek, naprimer zagon motorja, izpis na zaslonu, izračun...

Primer:

```
IF A>B THEN A=A+1
```

Ker je  $A=2$  in  $B=3$ , logični pogoj ni izpolnjen in  $A=A+1$  se ne izvrši.

Logičnih pogojev je lahko več:

```
IF LP1 AND LP2 AND LP2 THEN s1
```

Ukaznih stavkov je lahko več:

```
IF LP THEN
  s1
  s2
  ...
  sN
End IF
```

### 5.6.2 Stavki IF THEN ELSE

If Then stavka:

```
IF A>B THEN A=A+1
```

```
IF A<B THEN B=B+1
```

Lahko nadomestimo z enim If Then Else stavkom:

```
IF LP1 THEN
  S1
ELSE
  S2
End IF
```

## 6. PROGRAMIRANJE, ZAJEM IN OBDELAVA MERITEV

**Vaja 6.1:** Merjenje karakteristike upora:

S pritiskom na gumb start sprožimo merjenje uporovne  $U(I)$  karakteristike. Merimo v 32 točkah s povečevanjem napetosti od 0V do 5 V s korakom  $5/32 \text{ V} \cong 0,16 \text{ V}$ , v časovnih presledkih 0,2 s. Ko zajamemo vse vrednosti, številka tekoče meritve se izpisuje na zaslonu, shranimo tabelo meritev v datoteko podatki.dat na M disk v mapo RM v podmapo skupina 1(2).

Na formo vstavimo sledeče objekte:

timer  
command

izvajanje meritev na 0,2 s  
za prožitve meritev

label	za izpis zaporedne meritve
command	za shranitev tabele v datoteko

Definiramo globalno spremenljivko index, referenčni upor Rref in tabelo za shranjevanje meritev:

**Dim index as integer, Rref as integer, tabela(2,32) as string**

Ko pritisnemo na gumb START naj se začne zajemanje meritev: Napetost  $V_o(1)$  povečujemo od 0V v korakih po 0,16V do 5V. Pri vsakem povečanju merimo napetost na referenčnem uporu Rref, to je vhod  $V_i(0)$ . Iz znane pritisnjene napetosti  $V_o(1)$  in znanega padca napetosti na referenčnem uporu  $V_i(0)$ , lahko izračunamo tok in napetost na neznanem, oziroma merjenem uporu:

$I=V_i(0)/R_{ref}$  'Rref je znana upornost,  
 $U=V_o(1)-V_i(0)$

**Dogodek:** Form\_Load()

Tabelo za vpis meritev opremimo z naslovno vrstico (vrstica nič) indeks postavimo na prvo vrstico in vklopimo časovnik.

```
Private Sub Form_Load()  
tabela(0, 0) = "Št. meritve"  
tabela(1, 0) = "U/V"  
tabela(2, 0) = "I/A"  
index = 0  
timer1.enabled=true  
End Sub
```

Da se meritve izvajajo na vsake 0,2 s, je potrebno nastaviti lastnost interval timer1 na 200.

**Dogodek:** Izvajanje meritev - Timer1\_Timer()

```
Private Sub Timer1_Timer()  
dim uu,ii as string  
AnOut 1,5  
index=index+1  
label1.caption:=str(index)  
tabela(0, index)=str(index)  
uu=format(5-Anin(0), "0.000")  
tabela(1,index)= uu  
ii=format(Anin(0)/Rref, "0.000")  
tabela(2,index)=ii  
index=index+1  
if index=33 then timer1.enabled=false  
End Sub
```

**Dogodek** klik na gumb Shrani

```
Private Sub Command2_Click()  
label1.caption:="shanjujem datoteko "  
Open "c:\meritve.txt" For Output As #1  
for i=1 to index  
write #1, tabela(0,i); " ";tabela(1,i);" ";tabela(2,i)  
next i  
Close #1  
End Sub
```

Ne pozabite deklarirati tudi vse ostale pomožne spremenljivke.

### Vaja 6.2:

Zajemi nelinearno karakteristiko svetleče diode (LED), 32 točk, čas vzorčenja 250ms. Meritve shrani v datoteko dioda.dat na M disk v mapo RM v vašo podmapo (skupina 1(2)). Podatke odprite v Excelu. Podatki naj imajo

svojo enoto (določite format). Narišite graf, dodajte mu trend (ustrezni). V preglednici izračunajte diferenčno upornost svetleče diode od kolena naprej. Za izmerjene podatke izračunaj  $R^2$  in standardno napako ocene.

## 7. EKSPETIMENTI

### 7.1 Svetleča dioda

Zajemi nelinearno karakteristiko svetleče diode (LED), 32 točk, čas vzorčenja 250ms. Meritve shrani v datoteko dioda.dat na C disk v mapo student v svojo podmapo (skupina1 ali skupina2). Podatke odpri v Excelu. Podatki naj imajo svojo enoto (določiš format). Narišite graf in mu dodajte trend (ustrezni). V preglednici izračunajte diferenčno upornost svetleče diode od kolena naprej. Za izmerjene podatke izračunajte  $R^2$  in standardno napako.

### 7.2 Praznjenje kondenzatorja

Naredite program v Visual basicu, ki bo zajel karakteristiko polnjenja kondenzatorja v 100 točkah, vezje na sliki 7.1. V Excelu nariši graf, dodaj mu trend (ustrezni). V preglednici izračunaj ohmsko upornost kondenzatorja za polnjenje do 63% svoje maksimalne vrednosti. Za iste podatke izračunaj še presečišče s vertikalno osjo,  $R^2$  in standardno napako. Izberite tako elementa R in C, da bo produkt med 1 in 4 s. Točke zajemate v času, ki je 3 krat večji od RC konstante.

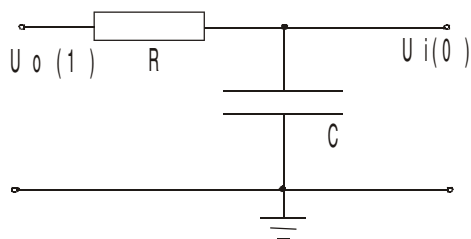
Kadar je zveza nelinearna, jo dostikrat lahko z uvedbo nove spremenljivke lineariziramo. Pri praznjenju kondenzatorja je napetost eksponentna funkcija časa.

$$U_c = U_0 \cdot e^{-\frac{t}{RC}} \quad (7.1)$$

kjer je  $U_0$  začetna napetost na kondenzatorju in  $U_c$ , napetost na kondenzatorju. Program Excel omogoča določitev eksponentne aproksimacijske funkcije znotraj grafa. Jasnejšo sliko o postopku aproksimacije dobimo, če enačbo praznjenja logaritmiramo:

$$\ln U_c = \ln U_0 - \frac{1}{RC} t \quad (7.2)$$

Če uvedemo novo odvisno spremenljivko  $\ln U_c$ , dobimo linearen graf, katerega konstanta je  $\ln U_0$  in smerni koeficient  $-1/RC$ .



Slika 7.1: Vezje za merjenje karakteristike kondenzatorja

### 7.3 Analiza prostega pada

Meritev opravimo tako, da spustimo merilno letev skozi merilna vrata. Merilna letev pri tem prosto pada, in prekinja svetlobni snop. Vsakič ko je svetlobni snop prekinjen je merilna letev padla za širino pasu na merilni letvi. V tabelo zapisujemo čas prekinitve in pot ki jo je do takrat opravila. Meritev opravimo s programom kinematika, zaženemo ga v DOS-u. Rezultat je datoteka meritev shranjena pod imenom s(t) v mapi c:\student\kinematika.

Datoteko odpremo v Excelu. V prvem stolpcu naj bo zaporedna številka, drugem stolpcu imejmo čas ( $t$ ), enota je s, v tretjem pa pot ( $s$ ), enota je m. Ker imamo enakomerno pospešeno gibanje je zveza med potjo in časom enaka:

$$s(t) = \frac{at^2}{2} \quad (7.3)$$

Če bi narisali graf, bi imeli parabolično odvisnost poti od časa. Enačbo korenimo in dobimo:

$$\sqrt{s} = \sqrt{\frac{a}{2}} t \quad (7.4)$$

Če sedaj narišemo graf  $\sqrt{s}(t)$ , potem je naklon premice enak  $\sqrt{\frac{a}{2}}$ . Tako lahko izračunamo pospešek.

Narišite graf v katerem bomo imeli točke za odvisnost  $s(t)$  in  $\sqrt{s}(t)$ . V obe krivulji dodajte ustrezn trend, in izpišite vrednosti trendnih linij ob grafih. V preglednici tudi izračunajte naklon in iz njega pospešek.

Hitrost prosto padajočega predmeta je enaka:

$$v = a \cdot t \quad (7.5)$$

Hitrost dobimo iz podatkov za pot z odvajanjem:

$$v(t) = \frac{ds(t)}{dt} \approx \frac{\Delta s(t)}{\Delta t} = \frac{s_{t+1} - s_t}{t_{t+1} - t_t} \quad (7.6)$$

Narišimo graf  $v(t)$ , v graf dodajmo ustrezn trend in ob njem izpišimo podatke trenda. Iz naklona dobimo pospešek. Tudi v preglednici izračunajmo pospešek s pomočjo ustrezne funkcije. Izračunate tudi začetno hitrost, korelacijski koeficient in standardno deviacijo hitrosti.

Pospešek lahko izračunamo iz podatkov za hitrost, tako, da naredimo numerično odvajanje:

$$a(t) = \frac{dv(t)}{dt} \approx \frac{\Delta v(t)}{\Delta t} = \frac{v_{t+1} - v_t}{t_{t+1} - t_t} \quad (7.7)$$

Narišimo graf  $a(t)$ , in graf povprečnega pospeška, dodajmo ustrezn trend in ob njem izpišimo podatke trenda. Iz naklona dobimo pospešek. Tudi v preglednici izračunajmo pospešek s pomočjo ustrezne funkcije. Izračunate tudi korelacijski koeficient in standardno deviacijo pospeška. Izračunajte kolikšna bi bila hitrost po 5s.