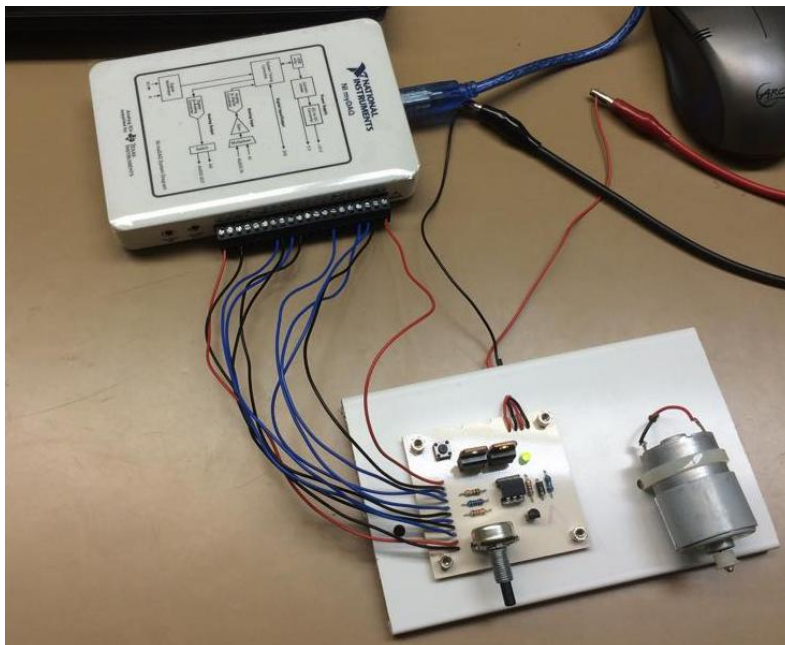


**»Mladi za napredek Maribora 2015«  
32. srečanje**

**Krmiljenje DC motorja s PWM signalom**

Raziskovalno področje: Elektrotehnika, elektronika  
Raziskovalna naloga



Avtor: ŽAN KOLARIČ

Mentor: MILAN IVIČ

Šola: SREDNJA ELEKTRO-RAČUNALNIŠKA ŠOLA MARIBOR

**2015, Maribor**

## 1. KAZALO VSEBINE

1.	KAZALO VSEBINE.....	2
2.	KAZALO SLIK.....	3
3.	POVZETEK .....	4
4.	ZAHVALA.....	4
5.	UVOD .....	4
6.	DC MOTOR.....	5
6.1	Zgradba.....	5
6.2	Delovanje.....	6
7.	PROGRAMSKO OKOLJE LABVIEW.....	6
8.	PROGRAMSKE STRUKTURE .....	9
8.1	Zanka While .....	9
8.2	Prenos podatkov preko roba zanke While .....	10
8.3	Struktura Case.....	10
8.4	Sekvenca.....	11
9.	MERILNA KARTICA NI MYDAQ.....	11
10.	VIRTUALNI INSTRUMENTI PALETE DAQMX .....	15
11.	PWM SIGNAL.....	17
12.	KRMILJENJE DC MOTORJA.....	18
12.1	Električni načrt vmesnika .....	18
12.2	Izračun elementov vmesnika.....	19
12.3	Darlingtonova vezava .....	20
12.4	Programski del, virtualni instrument v okolju LabVIEW .....	21
12.5	Čelna plošča .....	22
12.6	Blok diagram.....	22
12.7	Shema tiskanega vezja .....	24
13.	REALIZACIJA .....	24
14.	ZAKLJUČEK.....	26
15.	VIRI.....	27
16.	SLIKOVNI VIRI.....	27

## 2. KAZALO SLIK

Slika 1: DC motor (vir: Domel d.o.o.) .....	5
Slika 2: Zgradba DC motorja (vir: Kolektorski stroji).....	5
Slika 3: Čelna plošča .....	7
Slika 4: Blok diagram.....	7
Slika 5: Paleta kontrol in indikatorjev .....	8
Slika 6: Objekti v blok diagramu .....	9
Slika 7: Primer uporabe zanke While.....	9
Slika 8: Tuneli na zanki While.....	10
Slika 9: Struktura Case .....	10
Slika 10: Sekvenca Flat, primer uporabe .....	11
Slika 11: Kartica NI myDAQ (vir: National Instruments d.o.o.).....	12
Slika 12: Vhodi in izhodi na kartici NI myDAQ (vir: National Instruments d.o.o.).....	12
Slika 13: Vhodi za meritve z multimetrom (vir: National Instruments d.o.o.) .....	13
Slika 14: NI myDAQ (vir: National Instruments d.o.o.).....	14
Slika 15: Orodja DAQmx knjižice .....	15
Slika 16: DAQmx Create Virtual Channel vi.....	15
Slika 17: DAQmx Timing vi .....	15
Slika 18: DAQmx Start Task vi .....	16
Slika 19: DAQmx Read vi .....	16
Slika 20: DAQmx Write vi.....	16
Slika 21: DAQmx Clear Task vi .....	16
Slika 22: PWM Signal.....	17
Slika 23: Blokovna shema krmilja .....	18
Slika 24: Električni načrt vmesnika .....	18
Slika 25: Optosklopnik (vir: Vishay) .....	20
Slika 26: Darlingtonova vezava .....	20
Slika 27: Krmiljenje DC motorja s PWM signalom, Čelna plošča .....	22
Slika 28: Krmiljenje DC motorja s PWM signalom, Blok diagram.....	22
Slika 29: Shema tiskanega vezja .....	24
Slika 30: Tiskano vezje in kartica NI myDAQ .....	24
Slika 31: Laboratorijski usmernik .....	25
Slika 32: Priključitev vezja na merilno kartico in računalnik .....	25

### **3. POVZETEK**

Za izdelavo raziskovalne naloge sem se odločil, ker me je zanimalo, kako bi lahko krmilili vrtljaje enosmernega motorja. Raziskovalna naloga je pokazala, da se vrtljaji enosmernega motorja lahko krmilijo z PWM signalom. Nalogo sem realiziral z merilno kartico NI myDAQ. Programski del naloge sem naredil v programskem okolju LabVIEW. Na izdelanem modelu se vklop/izklop vrtenja enosmernega motorja krmili s tipko, PWM signal in s tem hitrost vrtenja enosmernega motorja pa se spreminja s potenciometrom.

### **4. ZAHVALA**

Zahvaljujem se vsem, ki so mi omogočili izdelavo te raziskovalne naloge, predvsem mentorju, brez katerega naloga nebi bila takšna, kot je sedaj.

### **5. UVOD**

Za krmiljenje enosmernega motorja s PWM<sup>1</sup> signalom bomo uporabili merilno kartico (modul) NI myDAQ. Hitrost vrtenja DC motorja bomo spreminjali s spreminjanjem napetosti na analognem vhodu merilne kartice. Frekvenca PWM signala bo konstantna 100 Hz. S spreminjanjem napetosti na analognem vhodu bomo spreminjali *Duty Cycle* PWM signala. Med merilno kartico (NI myDAQ) in DC motor bomo vstavili ustrezní vmesnik, da ne pride do preobremenitve in s tem do uničenja merilne kartice.

---

<sup>1</sup> PWM (ang. Pulse Width Modulation) pulzno širinska modulacija.

## 6. DC MOTOR

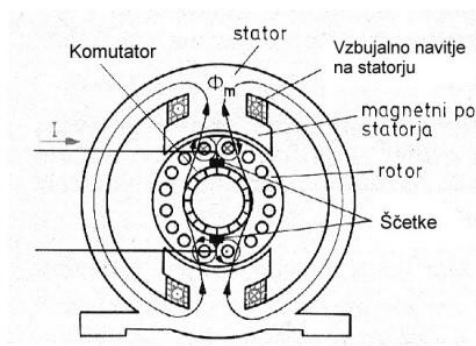
Enosmerni komutatorski stroj, oziroma enosmerni motor s ščetkami večinoma uporabljamo v hitrostno reguliranih pogonih v območju od nekaj W do okrog 10 MW moči. Enosmerni motor je priljubljen zaradi enostavne zgradbe in nizke cene.



Slika 1: DC motor (vir: Domel d.o.o.)

### 6.1 Zgradba

Enosmerni motor vsebuje dva glavna dela: stator in rotor. Rotor je lahko cilindričen ali ploščat. Na rotorju so navita navitja, povezana v serijo in priključena na kolektor. Kolektor se imenuje tudi mehanski komutator. Enosmerni tok, ki teče preko ščetk na kolektor, se pretvori v izmenični tok, ko teče skozi posamezna navitja. Komutator je dejansko pretvornik, ki preklaplja s frekvenco. Zunanje elektromagnetno vzbujanje je lahko nadomeščeno z uporabo trajnih magnetov, ki nimajo izgub moči. Ploščati rotor nima železnega jedra, zato je rotorska induktivnost zelo majhna. Tokovna gostota je lahko večja, kot pri rotorjih z železnim jedrom, saj je zaradi zraku izpostavljenega navitja hlajenje boljše.



Slika 2: Zgradba DC motorja (vir: Kolektorski stroji)

## 6.2 Delovanje

Ko motor priključimo na izvor električne napetosti, steče preko ščetk in kolektorja skozi navitji rotorja električni tok. Navitje statorja je lahko vezano zaporedno ali vzporedno z navitjem rotorja, njegova naloga pa je ustvariti magnetno polje, ki se bo zaključevalo preko rotorja. S tem se ustvari pojav sile na tokovodnik. Skozi tuljavo v rotorju teče električni tok. Tuljava se nahaja v magnetnem polju, ki ga povzroča stator. Na rotorsko tuljavo deluje tokovna sila. Smer sile je določena s pravilom leve roke. Na tuljavo bo deloval par sil. Tako se bo ustvaril vrtilni moment  $M$ , ki bo rotor zavrtel. Ko pride tuljava v nevtralno cono, nanjo ne deluje sila, vendar se zaradi vztrajnosti momenta zavrti naprej. Ko tuljava preide nevtralno cono, se zamenjata lameli pod ščetkama in se spremeni smer toka, zato je smer sile in vrtilnega momenta nespremenjena.

## 7. PROGRAMSKO OKOLJE LABVIEW

Programsko okolje LabVIEW je v celoti grafično zasnovano ter omogoča razvoj merilnih in testnih aplikacij. Programi napisani z orodjem LabVIEW se imenujejo virtualni instrumenti (VI). Vsak virtualni instrument je sestavljen iz čelne plošče in blok diagrama.

LabVIEW je blokavno orodje, kar pomeni, da se algoritem v tem programskem jeziku kodira blokavno oziroma z uporabo funkcij v obliki blokov, ki so med seboj povezani s povezavami. Vsebuje tudi *Application Builder*, ki omogoča kreiranje izvršne (exe) verzije, ki se lahko izvaja na računalniku, kjer razvojnega sistema LabVIEW ni nameščenega.

LabVIEW je mrežno zasnovano orodje, kar pomeni, da lahko kjerkoli v omrežju zajamemo signal, ga nekje na drugem računalniku v omrežju analiziramo in nekje na tretjem računalniku v omrežju predstavimo ali shranimo. Lahko pa seveda vse tri sklope opravimo na enem računalniku.

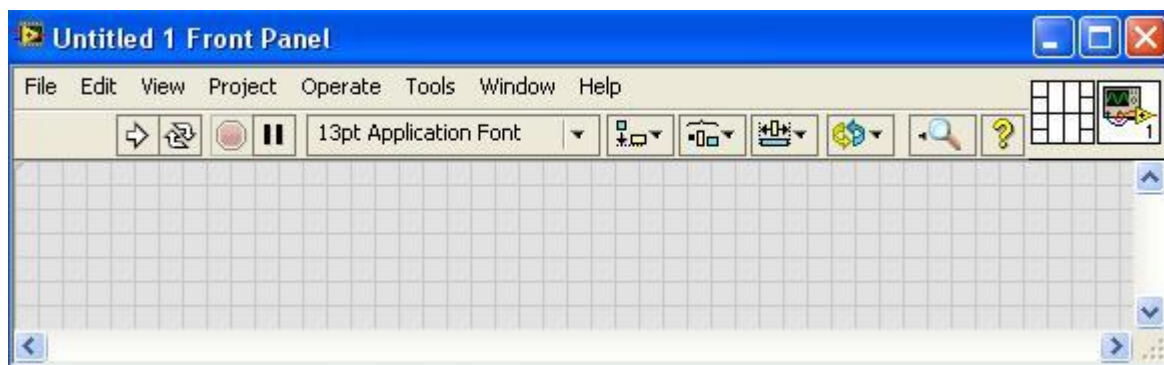
Vsaka merilna aplikacija je sestavljena iz treh segmentov:

- Zajemanje podatkov,
- Analiza,
- Prikaz.

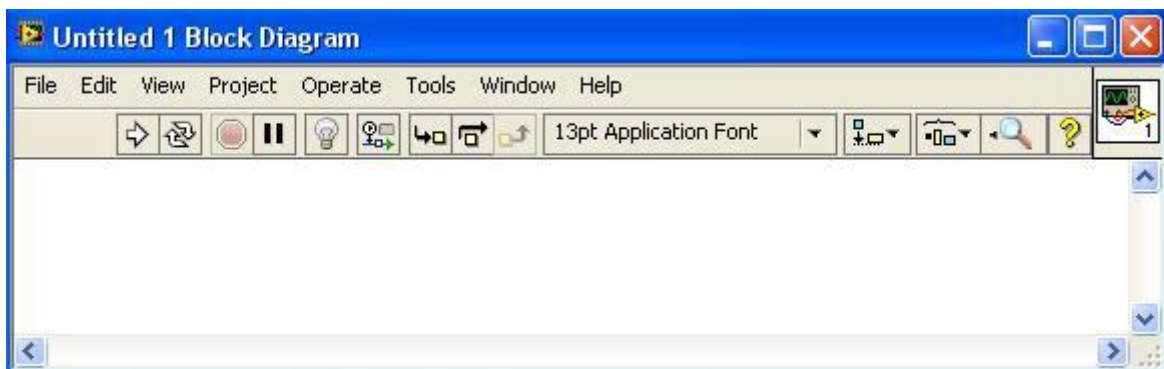
Ob zagonu programa LabVIEW se pojavi okno, kjer lahko izbiramo med kreiranjem novega projekta ali odpiranjem obstoječih, shranjenih projektov. Novi virtualni instrument (VI) kreiramo z izbiro: *File > New VI*. Ob tem se pojavita dve okni:

- Čelna plošča (Front Panel)
- Blok diagram (Block Diagram)

Čelna plošča ima privzeto sivo ozadje in je namenjena kreiranju uporabniškega vmesnika, blok diagram pa kreiranju algoritma virtualnega instrumenta.



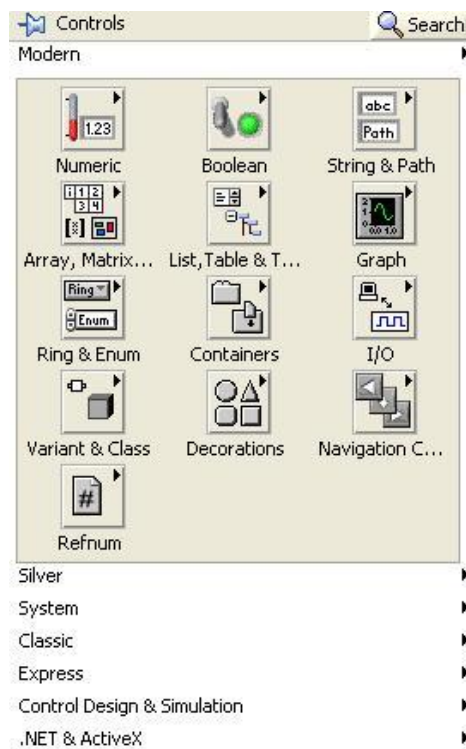
Slika 3: Čelna plošča



Slika 4: Blok diagram

Paleta kontrol in indikatorjev je sestavljena iz več podpalet. Za vsak objekt, ki ga vstavimo na čelno ploščo, se v blok diagramu avtomatsko kreira priključek (Terminal). Priključki služijo za prenos podatkov med čelno ploščo in blok diagramom. Iz oblike priključka lahko razberemo, ali pripada kontroli ali indikatorju.

Vhodni objekti, ki jim pravimo kontrole (Controls), predstavljajo vhodne podatke virtualnega instrumenta. Te objekte lahko med izvajanjem virtualnega instrumenta spreminjamo in s tem vplivamo na njegovo delovanje. Izhodni objekti, ki jim pravimo indikatorji (Indicators), pa služijo za prikaz podatkov. Med izvajanjem virtualnega instrumenta se izračunani podatki posredujejo indikatorjem, ki jih nato prikažejo v tekstovni ali grafični obliki.



**Slika 5: Paleta kontrol in indikatorjev**

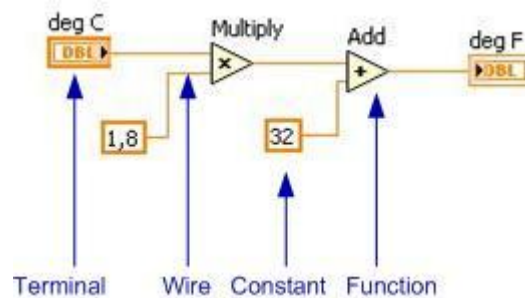
Funkcijo namestimo v blok diagram tako, da jo na paleti funkcij izberemo in povlečemo v blok diagram.

V blok diagramu virtualnega instrumenta se lahko nahajajo naslednji objekti:

- Priključki (Terminals), to so priključni elementi objektov čelne plošče,
- Podprogrami (SubVIs), to so dejansko virtualni instrumenti, ki prav tako vsebujejo čelno ploščo in blok diagram,
- Funkcije (Functions),
- Konstante (Constants), to so objekti v blok diagramu, katerim se vrednost med izvajanjem virtualnega instrumenta ne spreminja,



- Programske strukture (Structures), to so objekti, kot so zanka While, zanka For, pogojni stavki (Case) zaporedje (Flat Sequence) in drugi objekti,
- Povezave (Wires), ki povezujejo posamezne objekte blok diagrama in služijo prenosu podatkov,
- Objekte v blok diagramu, ki imajo vhode in izhode imenujemo Vozlišča (Nodes).

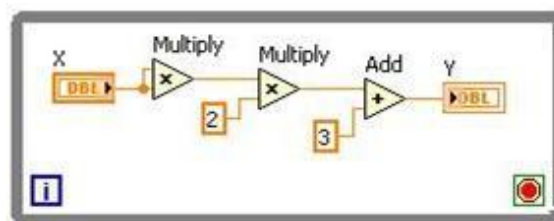


Slika 6: Objekti v blok diagramu

## 8. PROGRAMSKE STRUKTURE

### 8.1 Zanka While

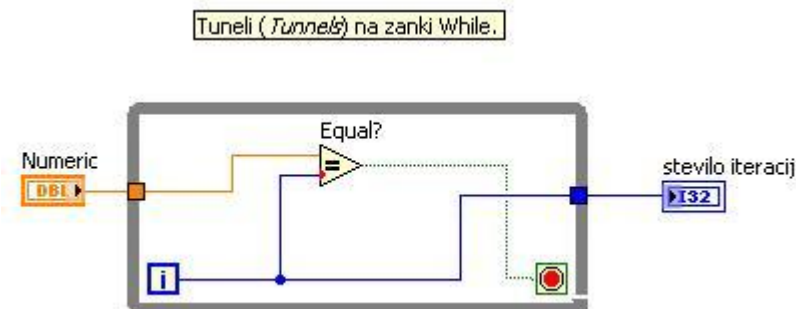
Zanka While vsebuje dva priključka, pogojni priključek (desno spodaj) in števec iteracij (levo spodaj). Na pogojni priključek moramo povezati logično vrednost (*True* ali *False*). S tem podatkom krmilimo izvajanje zanke. Pogojni priključek lahko v njegovem priložnem meniju nastavimo na vrednost *Stop if True* ali na *Continue if True*. Števec iteracij podaja informacijo o številu ponovitev zanke in teče od 0 naprej, kar pomeni, da je vrednost tega števca v prvi izvedbi zanke enaka 0.



Slika 7: Primer uporabe zanke While

## 8.2 Prenos podatkov preko roba zanke While

Pri prenosu podatkov preko roba zanke, se na robu kreira majhen kvadrat s črno obrobo in ozadjem v barvi podatkovnega tipa podatka. Temu kvadratu pravimo tunel (*Tunnel*). Predstavlja neke vrste pomnilniško lokacijo, saj se vanj zapisani podatki v njem začasno shranijo. V primeru, ko je podatek pripeljan v zanko, se podatek shrani v tunel, nato pa se v vsaki iteraciji zanke shranjena vrednost posreduje v notranjost zanke.



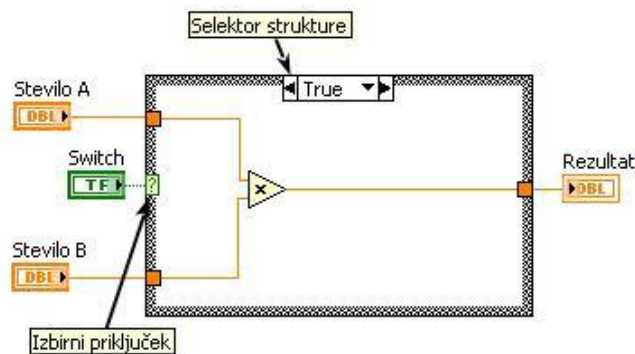
Slika 8: Tuneli na zanki While.

## 8.3 Struktura Case

Programska struktura Case (*Case Structure*) spada v skupino odločitvenih (pogojnih) stavkov. Ti stavki ponujajo možnost izbiranja različnih poti izvajanja. Struktura Case se uporablja, ko je potrebno izvesti določen del kode samo če je izpolnjen pogoj.

Strukturo Case kreiramo z orodjem *Case Structure*, ki se nahaja na paleti *Functions > Programming > Structures > Case Structure*.

Privzeto se ob namestitvi strukture kreirata dva okvirja (*True* in *False*), ki sta nameščena drug nad drugim. Struktura Case vsebuje izbirni priključek (*Selector Terminal*) in selektor strukture (*Case Selector*).



Slika 9: Struktura Case

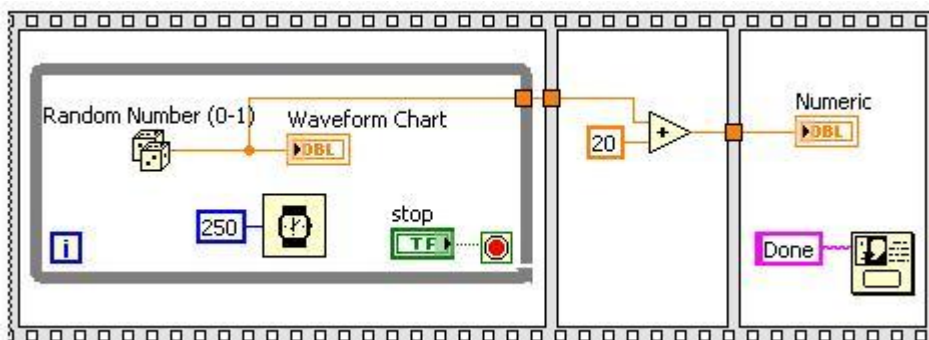
## 8.4 Sekvenca

Sekvenca (*Sequence*) oziroma zaporedje je programska struktura, ki je namenjena določitvi vrstnega reda izvajanja kode. Uporablja se v primerih, ko je pomemben vrstni red izvajanja posameznih segmentov kode. Obstajata dve vrsti sekvenc:

- Ploščata sekvenca (*Flat Sequence*)
- Zložena sekvenca (*Stacked Sequence*)

Sekvenca je sestavljena iz več okvirjev. Pri zloženi sekvenci so okvirji zloženi en nad drugim, tako da je viden samo en okvir, pri ploščati sekvenci pa so okvirji nameščeni drug ob drugem in so tako vidni vsi okvirji.

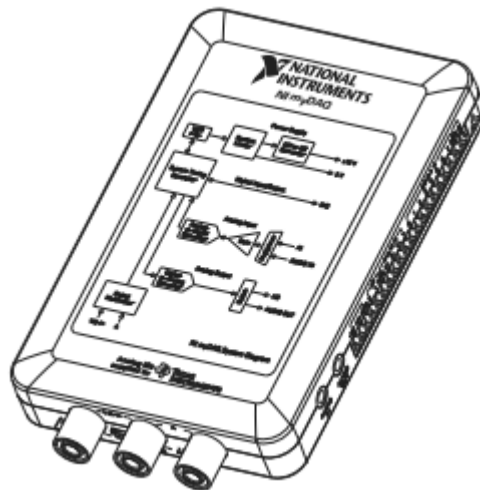
Sekvenca izvaja okvirje zaporedno in je tekom izvajanja ni mogoče prekiniti. Pri uporabi sekvenc je včasih potrebno podatke iz enega okvirja prenesti v drug okvir. Ploščate sekvence omogočajo kreiranje direktne povezave med objekti, ki se nahajajo v različnih okvirjih. Ob tem se na robu med okvirji kreira tunel.



Slika 10: Sekvenca Flat, primer uporabe

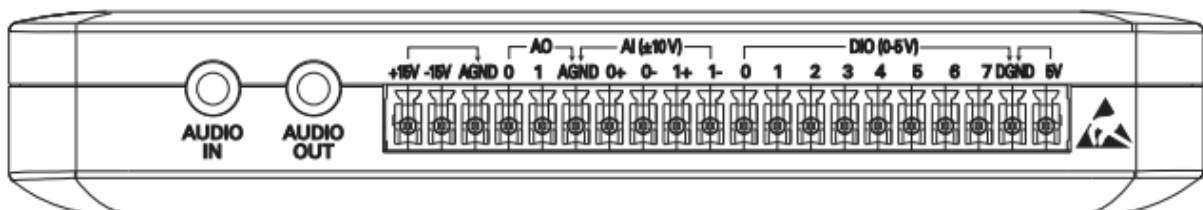
## 9. MERILNA KARTICA NI MYDAQ

Kartica NI myDAQ je nizkocenovna prenosna USB-kartica za zajemanje podatkov, ki uporablja programsko orodje LabVIEW in njegove komponente za delovanje. Izdelana je za merjenje in analiziranje signalov. Njen namen je zlasti uporaba za študijsko izobraževanje in je idealna za spoznavanje elektronskih elementov ter merjenje signalov iz senzorjev.

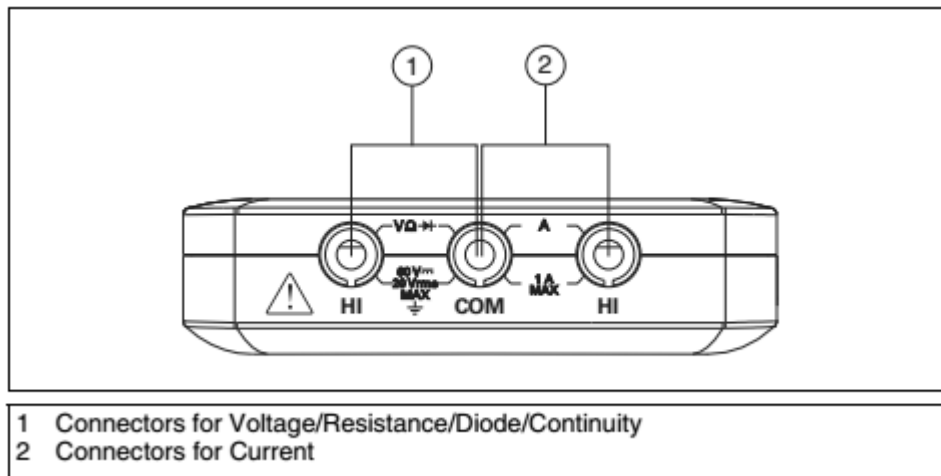


Slika 11: Kartica NI myDAQ (vir: National Instruments d.o.o.)

Na kartici najdemo 2 analogna vhoda z maso, 2 analogna izhoda z maso, 8 digitalnih vhodov in izhodov, glavno maso, 1 izhod za enosmerno napajanje zunanjih modulov z napetostjo 5 V in izhode za napetostno napajanje -15 V in +15 V. Ob tem moramo biti pozorni če uporabljamo napajanje iz kartice, da ne prekoračimo njene izhodne moči, ki znaša 500 mW. Na kartici najdemo tudi avdio vhod in en izhod ter vhode za digitalni multimeter. Na kartici so še števeci in časovniki za generiranje signalov. Z računalnikom je povezana preko USB vodila.



Slika 12: Vhodi in izhodi na kartici NI myDAQ (vir: National Instruments d.o.o.)



Slika 13: Vhodi za meritve z multimetrom (vir: National Instruments d.o.o.)

### Glavne karakteristike:

#### USB:

- USB port

#### Napetostno napajanje:

- +15 V
- -15 V
- +5 V

#### Audio:

- Audio vhod
- Audio izhod

#### Analogni izhodi:

- AO0 Analogni izhod 0
- AO1 Analogni izhod 1
- AGND Masa za analogne vhode/izhode

#### Analogni vhodi:

- AI0+ Analogni vhod 0, pozitivni priključek
- AI0- Analogni vhod 0, negativni priključek

- AI1+ Analogni vhod 1, pozitivni priključek
- AI1- Analogni vhod 1, negativni priključek

Digitalni vhodu/izhodi:

- DIO Digitalni vhodi/izhodi (0-7). Digitalni izhodi: logična 1 => 3,3 V
- DGND Masa za digitalne vhode/izhode

Digitalni multimeter:

- HI Pozitivni priključek za V, OHM, Diode
- COM Skupna masa za multimeter
- HI Pozitivni priključek za A (max 1 A)

Merilna kartica vsebuje še paleto NI ELVISmx:

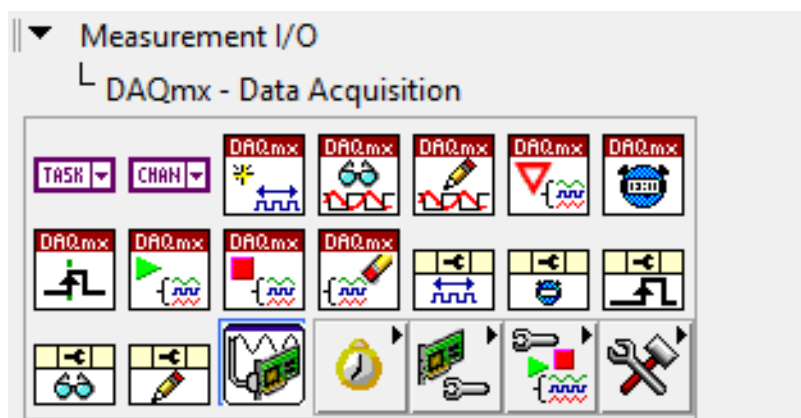
- DMM: Digitalni multimeter
- SCOPE: Osciloskop
- FGEN: Funkcijski generator
- BODE: Bodejev analizator
- DSA: Dinamični signalni analizator
- DigIn: Digitalni čitalec
- DigOut: Digitalni zapisovalnik



Slika 14: NI myDAQ (vir: National Instruments d.o.o.)

## 10. VIRTUALNI INSTRUMENTI PALETE DAQMX

Nabor virtualnih instrumentov za delo z merilnimi karticami se nahaja v paleti *Functions > Measurements I/O > NI-DAQmx*. V njej so konstante za naloge in kanale, ki smo jih kreirali s pripomočkom *DAQ Assistant*, virtualni instrumenti, vozlišča za nastavljanje lastnosti *Property Nodes* in ostale palete.



Slika 15: Orodja DAQmx knjižice

*DAQmx Create Virtual Channel vi* - Kreira virtualne kanale in jih doda k nalogi. Ta polimorfčni VI ustreza vhodno/izhodnemu tipu kanala kot je npr. analogni vhod/izhod, digitalni vhod/izhod ali vhod/izhod za števec. Pri tem izberemo tudi vrsto meritve oziroma generiranja kot sta npr. merjenje temperature in generiranje napetosti ter v določenih primerih tudi uporabljen senzor kot je npr. termočlen.



Slika 16: DAQmx Create Virtual Channel vi

*DAQmx Timing vi* - Ta polimorfčni vi nastavi število in frekvenco otipkov, ki jih je potrebno zmeriti ali generirati ter po potrebi kreira vmesni pomnilnik.



Slika 17: DAQmx Timing vi

*DAQmx Start Task vi* - Spremeni stanje naloge v zagon meritve oziroma generiranja.



Slika 18: DAQmx Start Task vi

*DAQmx Read vi* - Bere vrednosti kanalov, ki smo jih navedli v nalogi. Temu polimorfičnemu VI nastavimo obliko in način branja podatkov.



Slika 19: DAQmx Read vi

*DAQmx Write vi* - Zapiše vrednosti kanalov, ki smo jih navedli v nalogi. Temu polimorfičnemu VI nastavimo obliko in način pisanja podatkov.



Slika 20: DAQmx Write vi

*DAQmx Clear Task vi* - Pobriše nalogo, ki jo pred tem po potrebi zaustavi in hkrati sprosti rezervirane vire naloge.

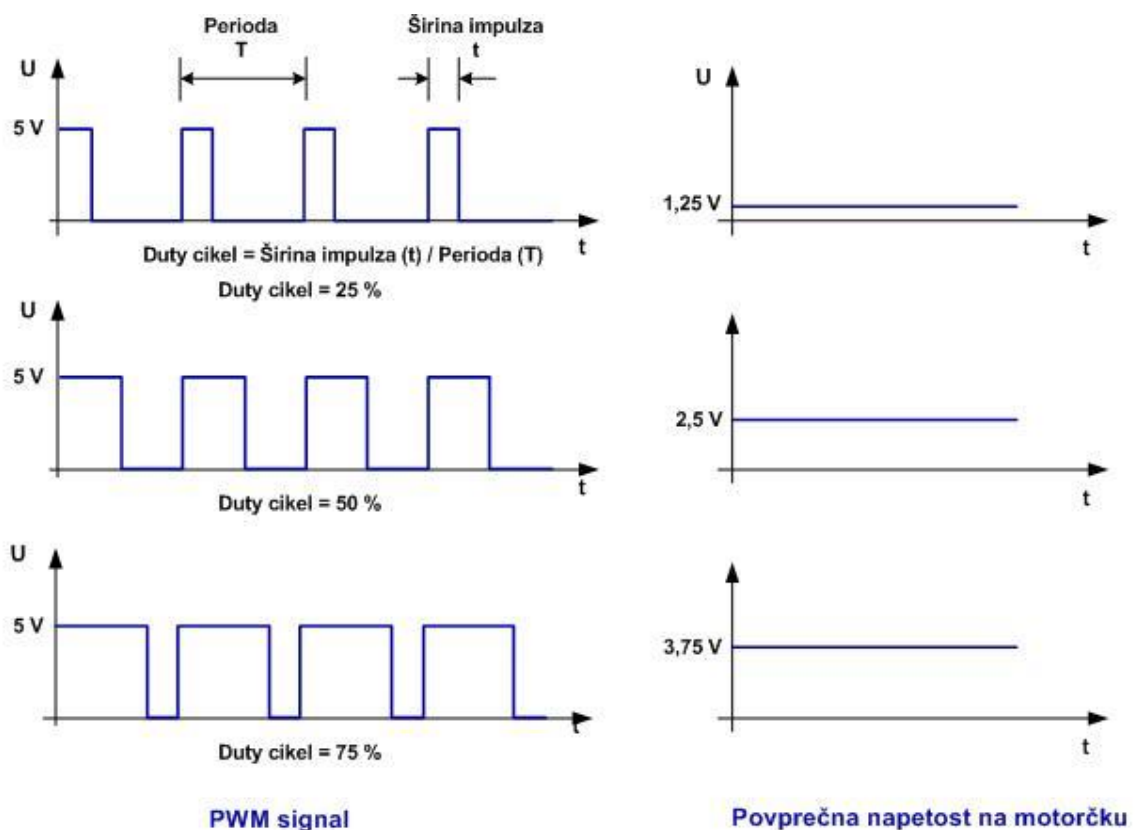


Slika 21: DAQmx Clear Task vi



## 11. PWM SIGNAL

Pulzno širinska modulacija (PWM) je tehnika nadzora oziroma krmiljenja energije, ki jo pošiljamo el. porabnikom z uporabo on-off digitalnega signala. Delu periode, ko je digitalni signal na on pravimo *duty cycle*. Povprečna vrednost enosmernega signala se lahko spreminja s spreminjanjem duty cikla. Duty cikel se lahko spreminja od 0 (signal je na off) do 1 (signal je konstantno na on). Če ima signal vrednost 5 V kadar je na on in vrednost 0 V kadar je na off, potem lahko s spreminjanjem duty cikla dosežemo katero koli vrednost med 0 V in 5 V. S pomočjo te metode lahko krmilimo npr. hitrost vrtenja DC motorjev ali krmilimo svetilnost svetlobnih teles. Če na enosmerni motor priključimo *PWM* signal, lahko reguliramo njegovo hitrost vrtenja. Pri različnem *Duty Cycle* bo hitrost vrtenja enosmernega motorja različna.

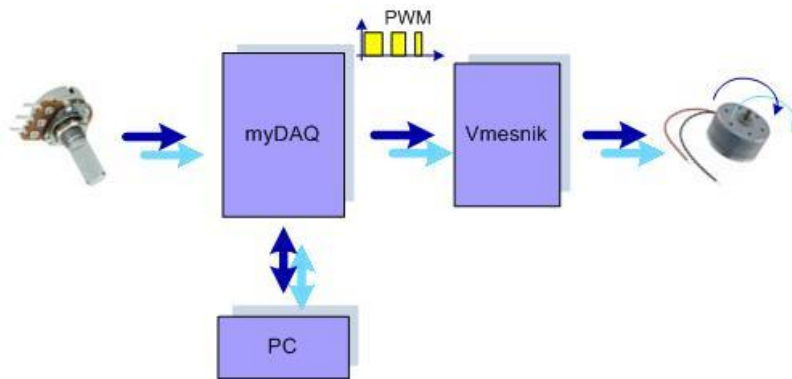


Slika 22: PWM Signal

## 12. KRMILJENJE DC MOTORJA

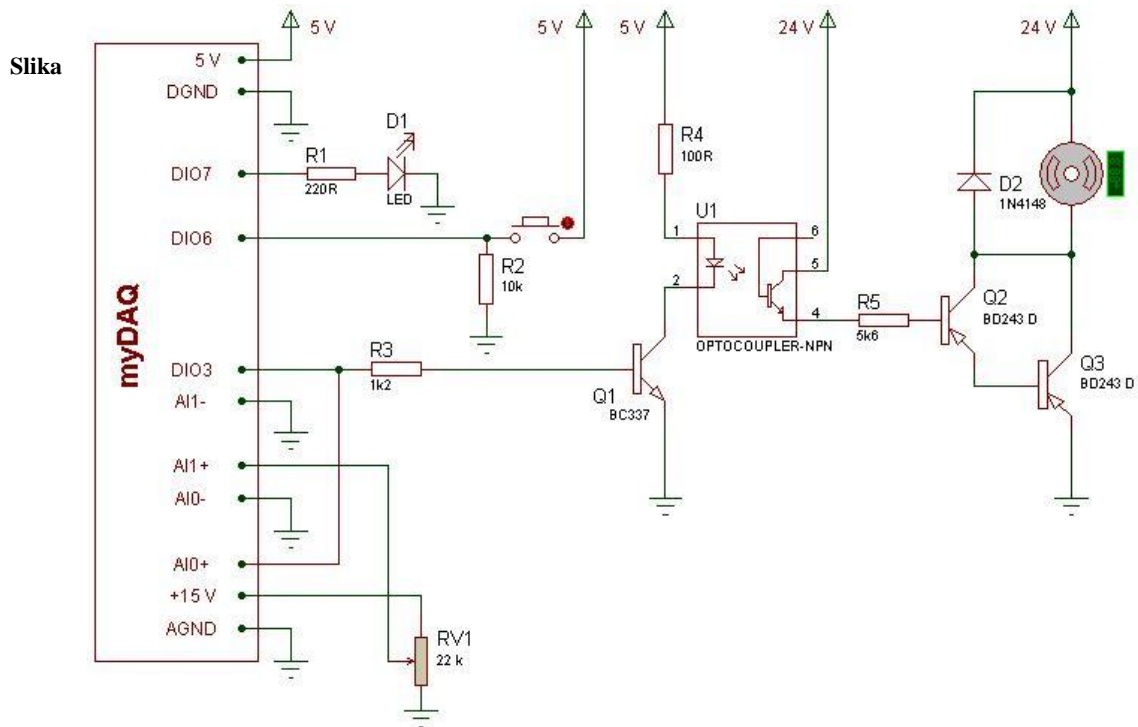
Naloga je razdeljena na dva dela:

- Programski del v okolju LabVIEW.
- Načrtovanje vmesnika, ki bo poskrbel za zaščito modula myDAQ in omogočil ustrezno moč za krmiljenje DC motorja.



Slika 23: Blokovna shema krmilja

### 12.1 Električni načrt vmesnika



Električni načrt vmesnika

24:

## 12.2 Izračun elementov vmesnika

- Izračun upora za omejitev izhodnega toka na priključku  $DIO3$ :

$$R_3 = \frac{U_{DIO3} - U_{BE}}{I_{DIO3}} = \frac{3,3 \text{ V} - 0,7 \text{ V}}{2 \cdot 10^{-3} \text{ A}} = 1,3 \text{ k}\Omega \Rightarrow R_3 = 1,2 \text{ k}\Omega$$

- Izračun toka optosklopnika z uporom  $R4$ :

$$IF = 30 \text{ mA} \Rightarrow \text{Katalog (priloga spodaj)}$$

$$UF = 1,1 \text{ V} \Rightarrow \text{Katalog (priloga spodaj)}$$

$$R_4 = \frac{U_S - U_F - U_{CEsat}}{I_F} = \frac{5 \text{ V} - 1,1 \text{ V} - 0,7 \text{ V}}{30 \cdot 10^{-3} \text{ A}} = 106 \Omega \Rightarrow R_4 = 100 \Omega$$

- Določitev toka izhodnega tranzistorja optosklopnika z uporom  $R5$

$$P_{dis_{max}} = 200 \text{ mW} \Rightarrow \text{Katalog (priloga spodaj)}$$

$$U_{ce_{sat}} = 0,4 \text{ V} \Rightarrow \text{Katalog (priloga spodaj)}$$

$$P_{dis} = \frac{P_{dis_{max}}}{2} = \frac{200 \text{ mW}}{2} = 100 \text{ mW}$$

$$I_{C_{opto}} = \frac{P_{dis_{max}}}{U_M} = \frac{100 \text{ mW}}{24 \text{ V}} = 4,2 \text{ mA}$$

$$R_5 = \frac{U_M - 2 \cdot U_{BE} - U_{CEsat}}{I_{C_{opto}}} = \frac{24 \text{ V} - 1,4 \text{ V} - 0,4 \text{ V}}{4,2 \cdot 10^{-3} \text{ A}} = 5,33 \text{ k}\Omega \Rightarrow R_5 = 5,6 \text{ k}\Omega$$

- Ali potrebujemo Darlingtonovo vezavo?

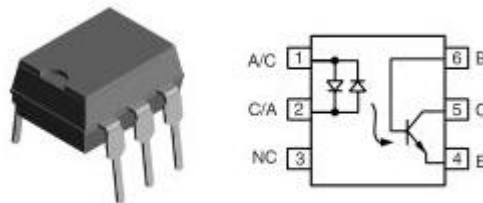
$$h_{FE} = 15 \Rightarrow \text{Katalog}$$

$$I_C = h_{FE} \cdot I_{C_{opto}} = 15 \cdot 4,2 \text{ mA} = 0,063 \text{ A} \Rightarrow \text{Izhod z enim tranzistorjem}$$

$$I_C = h_{FE} \cdot h_{FE} \cdot I_{C_{opto}} = 15 \cdot 15 \cdot 4,2 \text{ mA} = 0,945 \text{ A} \Rightarrow \text{Izhod z Darlingtonovo vezavo}$$

Skozi neobremenjen DC motor po podatkih proizvajalca teče tok  $I = 0,58 \text{ A}$ , zato smo morali na izhodu vezati dva tranzistorja BD 243 D v Darlingtonovo vezavo.

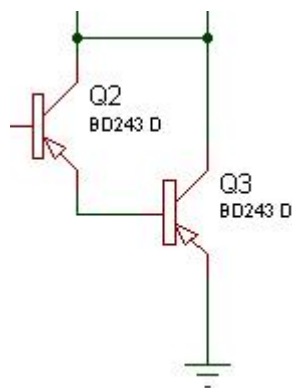
Optosklopnik galvansko loči modul myDAQ z izhodnim delom, kamor je priključen motor. S tem smo zaščitili modul myDAQ pred uničenjem v primeru okvare. Optosklopniki v enem ohišju vsebujejo tako oddajno diodo kot galvansko ločeni sprejemni tranzistor.



Slika 25: Optosklopnik (vir: Vishay)

### 12.3 Darlingtonova vezava

Darlingtonovo vezavo sestavljata dva tranzistorja povezana tako da se ojačen tok prvega ponovno ojači skozi drugega. Darlingtonova vezava se obnaša kot en tranzistor z zelo velikim tokovnim ojačenjem. Ima tri zunanje priključke, ki so ekvivalentni priključkom standardnega tranzistorja. Da je tranzistor odprt mora biti napetost 0,7 V na obeh bazno emitorskih spojih, torej skupaj 1,4 V. Dobavljivi so kot komponente, lahko pa ga sestavimo s pomočjo dveh tranzistorjev.

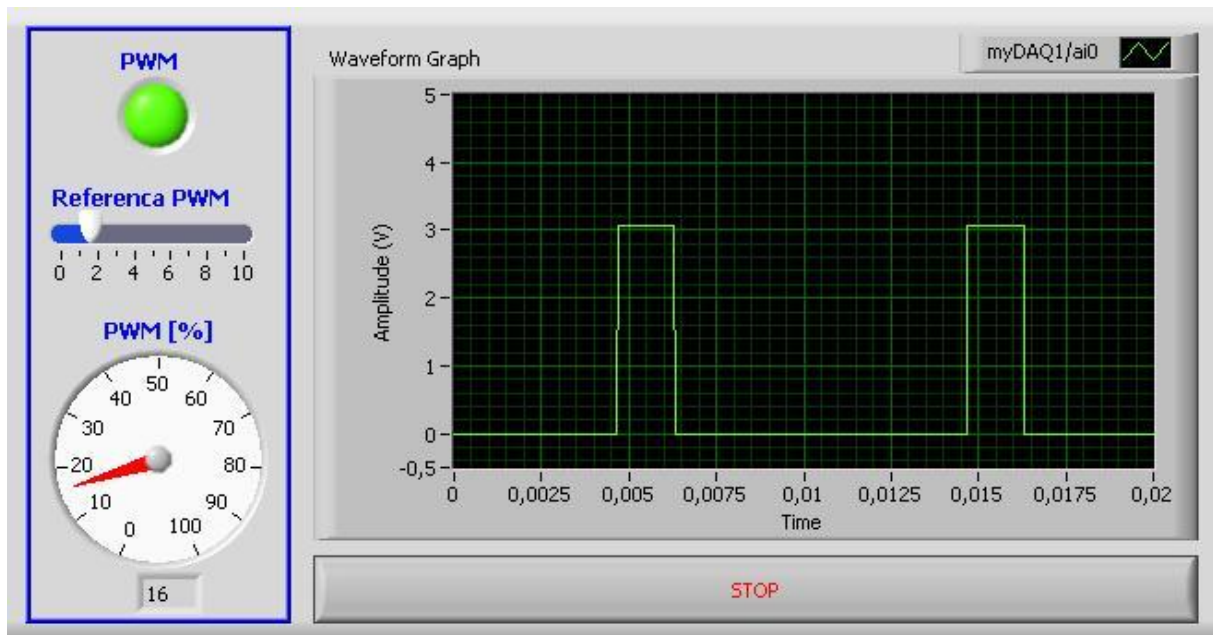


Slika 26: Darlingtonova vezava

## 12.4 Programski del, virtualni instrument v okolju LabVIEW

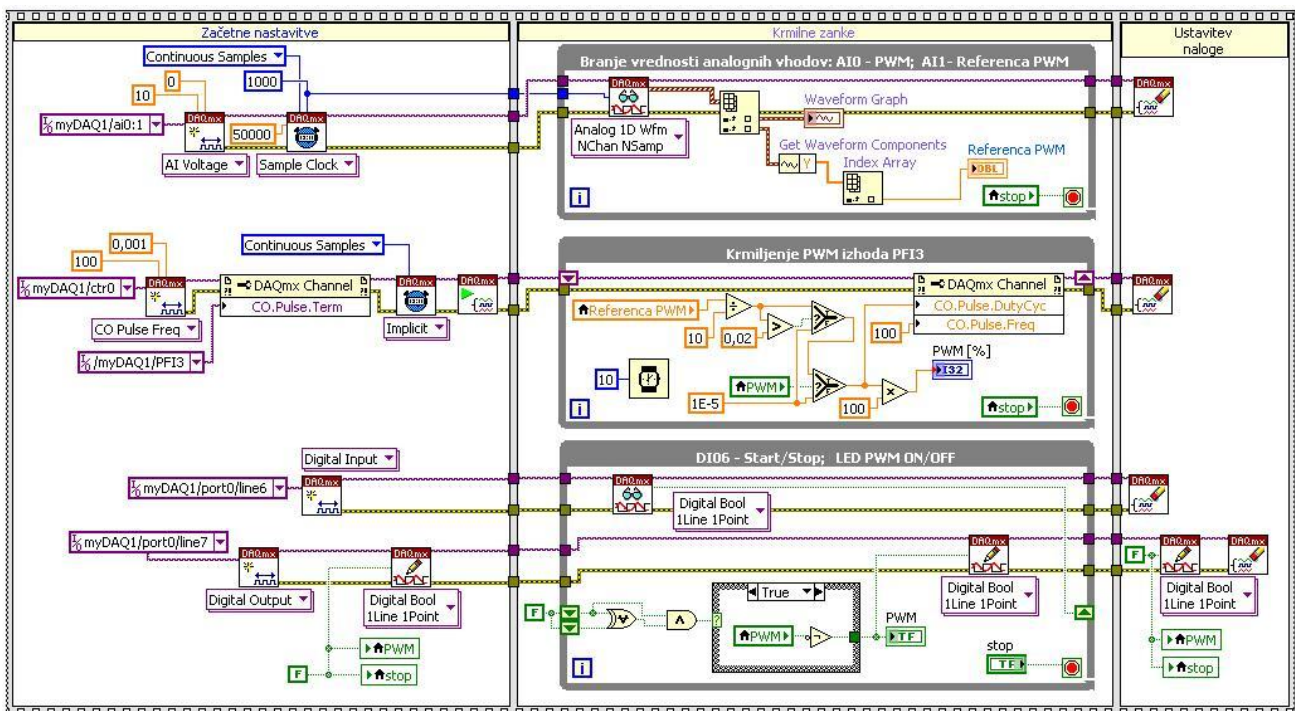
- Na digitalni vhod DIO3 je povezana tipka, na digitalni izhod DIO7 pa LED dioda. Ob vsakem pozitivnem prehodu stanja na DIO3, se spremeni izhodno stanje (DIO7). *Alternate* način delovanja tipke. Za nadzor delovanja izhoda mora biti na čelni plošči LED indikator z imenom PWM. Pri zagonu programa morata biti izhod DIO7 in indikator PWM v vsakem primeru izključeni. To velja tudi ob zaustavitvi programa, s klikom na gumb Stop.
- Na analogni vhod AI1 je priključen potenciometer, s katerim spreminjamo *Duty Cycle* PWM signala. Analogni vhod AI0 uporabimo za merjenje oblike PWM signala, zato mora virtualni instrument brati oba analogna vhoda, AI0 in AI1. S spreminjanjem napetosti na analognem vhodu AI1 (potenciometer) se mora na čelni plošči prikazovati velikost napetosti (0 V – 10 V) na indikatorju (*Horizontal Pointer Slide*) z imenom *Referenca PWM*.
- Za oblikovanje PWM signala na izhodu DIO3 (*PF13*) moramo uporabiti števec (*Counter*). Frekvenca PWM signala mora biti konstantna 100 Hz, začetna vrednost *Duty Cycle* pa 0,001. Preoblikovana merjena vrednost signala *Referenca PWM* (0 V - 10 V) krmili digitalni izhod DIO3, ko sveti LED indikator PWM. Za prikazovanje vrednosti *Duty Cycle* uporabimo indikator (*Gauge*) z imenom *PWM [%]*.
- PWM signal ki ga dobimo na digitalnem izhodu *DIO3*, merimo na vhodu AI0 in ga prikazujemo v grafikonu (*Waveform Graph*).

## 12.5 Čelna plošča



Slika 27: Krmiljenje DC motorja s PWM signalom, Čelna plošča

## 12.6 Blok diagram



Slika 28: Krmiljenje DC motorja s PWM signalom, Blok diagram

Prvi okvir strukture Flat predstavlja začetne nastavitve. Z virtualnimi instrumenti iz palete DAQmx so kreirani kanali, analogna vhoda AI0 (za merjenje oblike PWM signala) in AI1 (potenciometer za spreminjanje PWM) in digitalna kanala, DIO6 (priključena tipka) in DIO7 (priključena LED dioda). Kreiramo še števec (*Counter*), kjer nastavimo frekvenco 100 Hz in začetno vrednost duty cycle 0,001.

Nastavili smo kontinuirano zajemanje obeh analognih signalov z uporabo strojne ure. Delovanje strojne ure smo nastavili s pomočjo funkcije *DAQmx Timing*, kjer smo poleg kontinuiranega zajemanja določili še število otipanih vrednosti po kanalu (1000) in frekvenco tipanja (50 000).

Števec ima na merilni kartici myDAQ izhod za PWM signal na DIO3, zato smo nastavili izhod števca (*Counter Output:Pulse:OutputTerminal*) na PF13. Nastavili smo še kontinuirano generiranje PWM signala, ki se začne v prvem okvirju sekvence Flat s funkcijo DAQmx Start Task VI.

V prvem okvirju strukture Flat upoštevamo še zahtevo, da morata biti pri zagonu programa izhod DIO7 (LED dioda) in indikator PWM izključena.

Drugi okvir strukture Flat predstavlja krmilne zanke. V krožnem pomnilniku v zanki While s funkcijo za branje *DAQmx Read* sproti beremo vrednosti na analognem vhodu AI0 in AI1. Referenco PWM (napetost 0 V – 10 V) prikazujemo na indikatorju *Horizontal Pointer Slide*, PWM signal pa na grafikonu *Waveform Graph*.

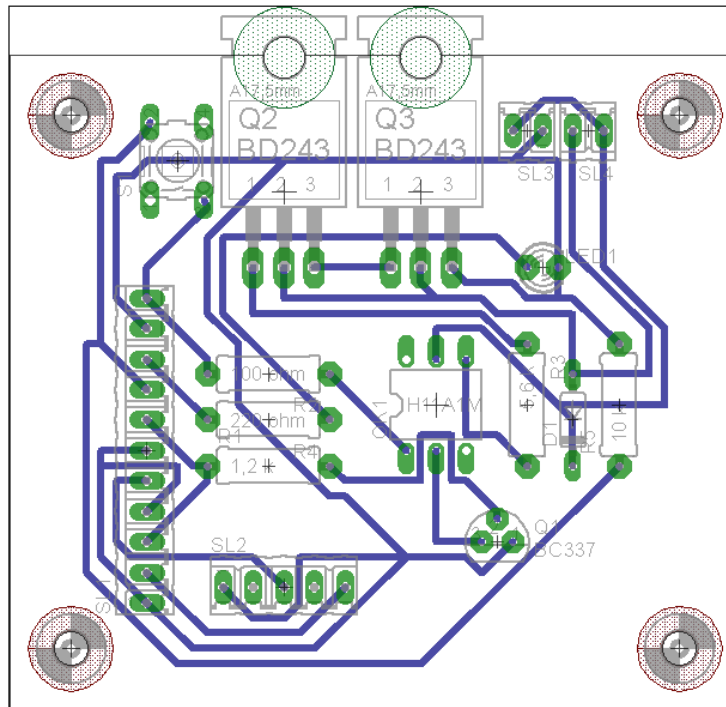
PWM signal v % prikazujemo na indikatorju *Gauge*. Prikazuje se v območju od 0 do 100 %. Uredili smo tudi delovanje tipke, s katero omogočamo PWM signal ter LED indikatorja za nadzor delovanja izhoda.

Za istočasno zaustavitev vseh zank While smo uporabili lokalno spremenljivko. To in lokalno spremenljivko za LED indikator PWM izhoda smo uporabili tudi v zadnjem okvirju sekvence Flat, da se je naloga ustavila.



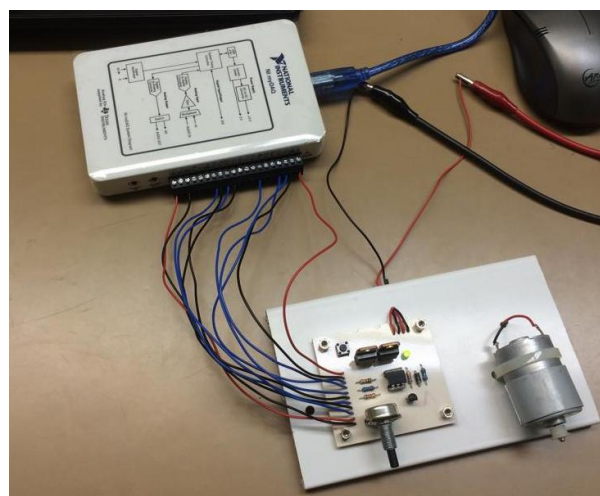
## 12.7 Shema tiskanega vezja

Tiskano vezje smo izdelali v okolju Eagle. Po načrtu smo povezali priključke na tiskanini z ustreznimi priključki na merilni kartici myDAQ, za napetost DC motorja pa smo uporabili usmernik.



Slika 29: Shema tiskanega vezja

## 13. REALIZACIJA

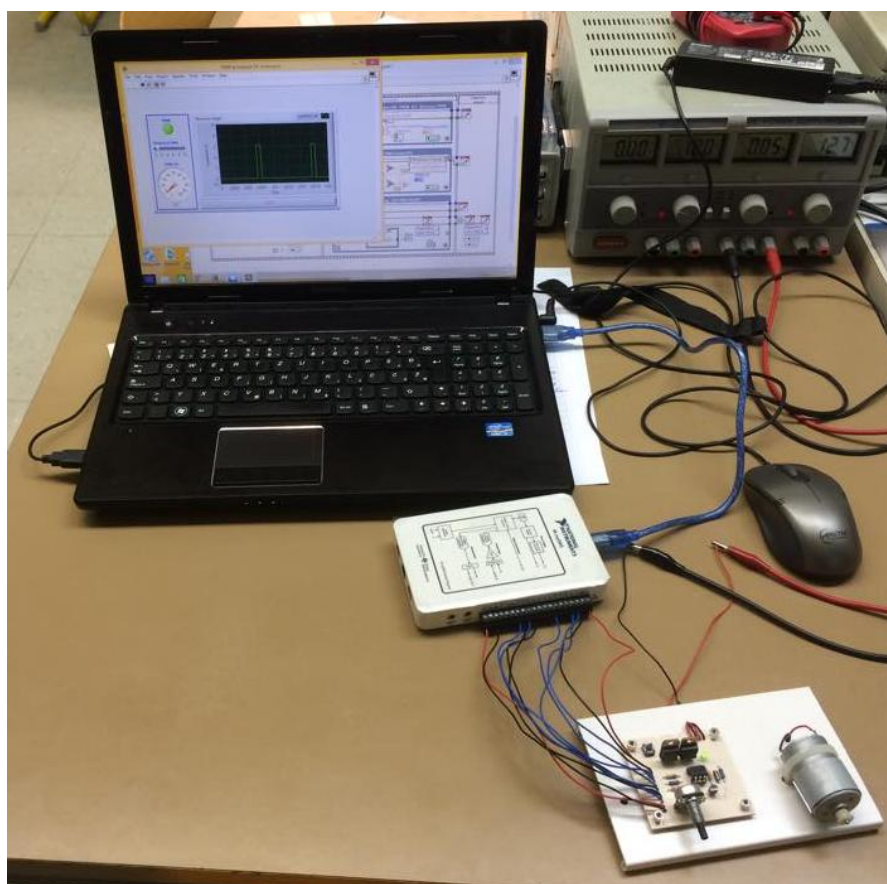


Slika 30: Tiskano vezje in kartica NI myDAQ





Slika 31: Laboratorijski usmernik



Slika 32: Priklučitev vezja na merilno kartico in računalnik

## **14. ZAKLJUČEK**

V raziskovalni nalogi smo praktično prikazali kako lahko krmilimo hitrost vrtenja enosmernega motorja s PWM signalom. Na začetku nismo vedeli kako uporabljati programsko okolje LabVIEW in merilno kartico NI myDAQ. Z njim smo se prvič srečali pri praktičnem pouku v šoli. S pomočjo profesorja pa smo se naučili še veliko drugih stvari, ki se jih pri praktičnem pouku nismo naučili. Naučili smo se tudi konstruiranja tiskanih vezij v okolju Eagle, spoznali foto postopek in postopek jedkanja ter še veliko drugih stvari, ki jih bomo lahko uporabil pri svojem nadaljnjem delu.

Naloga nem je na začetku delala nekaj preglavic, vendar smo z vztrajnostjo in potrpežljivostjo reševali problem za problemom in nalogo pripeljali do dokončne rešitve pravilnega delovanja po zahtevah, ki smo si jih zadali na začetku.

## 15. VIRI

- DC Motor [spletni vir] [ftp://ftp.scv.si/pters/simon\\_konecnik/ESN/4%20-%20Kolektorski%20stroji.pdf](ftp://ftp.scv.si/pters/simon_konecnik/ESN/4%20-%20Kolektorski%20stroji.pdf) (12.1.2015)
- NI myDAQ [spletni vir] [http://sukjaro.eu/ELFT-NI-palyazat/myDAQ\\_Manual.pdf](http://sukjaro.eu/ELFT-NI-palyazat/myDAQ_Manual.pdf) (13.1.2015)
- LabVIEW [spletni vir] [ftp://ftp.scv.si/vss/joze\\_lukanc/LabVIEW/LabVIEW\\_2010.pdf](ftp://ftp.scv.si/vss/joze_lukanc/LabVIEW/LabVIEW_2010.pdf) (15.1.2015)
- LabVIEW [spletni vir] [http://sukjaro.eu/ELFT-NI-palyazat/LabView\\_Manual.pdf](http://sukjaro.eu/ELFT-NI-palyazat/LabView_Manual.pdf) (16.1.2015)
- PWM Signal [spletni vir] <http://www.svet-el.si/o-reviji/samogradnje/659-pwm-elektronsko-krmiljen-usmernik-napajalnik?showall=1> (17.1.2015)
- Darlingtonova vezava [spletni vir] [http://en.wikipedia.org/wiki/Darlington\\_transistor](http://en.wikipedia.org/wiki/Darlington_transistor) (19.1.2015)
- Optosklopnik H11AA1 [spletni vir] <http://www.vishay.com/docs/83608/h11aa1.pdf> (13.1.2015)
- Tranzistor BD243D [spletni vir] <http://www.micropik.com/PDF/BD243%5B1%5D.pdf> (13.1.2015)
- Tranzistor BC337 [spletni vir] [http://www.onsemi.com/pub\\_link/Collateral/BC337-D.PDF](http://www.onsemi.com/pub_link/Collateral/BC337-D.PDF) (13.1.2015)

## 16. SLIKOVNI VIRI

Slika 33: DC motor. Dostopno na naslovu:

[http://www.domel.com/resources/files/galerije/26/mtn\\_4823714.jpg](http://www.domel.com/resources/files/galerije/26/mtn_4823714.jpg) (12.1.2015)

Slika 34: Zgradba DC motorja.

[ftp://ftp.scv.si/pters/simon\\_konecnik/ESN/4%20-%20Kolektorski%20stroji.pdf](ftp://ftp.scv.si/pters/simon_konecnik/ESN/4%20-%20Kolektorski%20stroji.pdf) (12.1.2015)

Slika 35: Kartica NI myDAQ.

[http://sukjaro.eu/ELFT-NI-palyazat/myDAQ\\_Manual.pdf](http://sukjaro.eu/ELFT-NI-palyazat/myDAQ_Manual.pdf) (13.1.2015)

Slika 36: Vhodi in izhodi na kartici NI myDAQ.

[http://sukjaro.eu/ELFT-NI-palyazat/myDAQ\\_Manual.pdf](http://sukjaro.eu/ELFT-NI-palyazat/myDAQ_Manual.pdf) (13.1.2015)

Slika 37: Vhodi za meritve z multimetrom.

[http://sukjaro.eu/ELFT-NI-palyazat/myDAQ\\_Manual.pdf](http://sukjaro.eu/ELFT-NI-palyazat/myDAQ_Manual.pdf) (13.1.2015)

Slika 38: NI myDAQ.

<http://lab.fs.uni-lj.si/ltt/images/stories/mydaq-large.png> (14.1.2015)

Slika 39: Optosklopnik.

<http://pinout-circuits-images.dz863.com/4/H11AA1.jpg> (16.1.2015)