

ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS FAKULTETAS
ELEKTRONIKOS KATEDRA

Antanas Paulauskas

Bešepetėlinio nuolatinės srovės variklio valdymas
Bakalauro darbas

Vadovas:
doc. D. Dervinis

ŠIAULIAI, 2013

ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS FAKULTETAS
ELEKTRONIKOS KATEDRA

TVIRTINU
Katedros vedėjas
doc. dr. G. Daunys

2013-06

Bešepetėlinio nuolatinės srovės variklio valdymas
Elektronikos inžinerijos bakalauro darbas

Vadovas:
doc. D. Dervinis
2013-06

Atliko:
NR-8 gr. stud. A. Paulauskas
2013-06

Recenzentas:

lekt. dr. D. Balbonas
2013-06

Paulauskas A. Brushless DC Electric Motor Control. Engineering and research advisor : Dervinis D. Šiauliai University, Faculty of Technology, Department of Electronics. – Šiauliai, 2013. – 29 pp.

Summary

In our times electronics devices changes very fast, some of these devices broke down. In some devices we can use undamaged components, such as brushless DC electric motors. Which can be used in home applications. But there is some problems with these motors controls, because it is driven by commutating three phases from DC power source.

In this thesis there is performed similar devices analysis. Chosen needed technologies and components for designed brushless DC motor controller. Chosen technologies are: Half – H bridge, low side current measurement, motor rotor's position finding and next commutation step setting. There are also explained how such motors work and how to generate steps.

In this thesis was projected electrical and printed circuit boards for controller. In electrical scheme there are all needed technologies for controller and lot of expansion possibilities. There was projected printed circuit board and made prototype. There is written software for controller in C language.

In controller's testing procedure there were got clue parameters of controller. These parameters are working voltage range from 5 to 14 volts with with error tolerance less than 1 percent. Working current is up to 5 amperes continuously or 20 amperes pulse. Main motor commutation steps frequency 400 hertz, which let to drive motor depending of the quantity of the poles up to 400 RPM.

In testing procedure there was problems with current measurement. They were solved fast by using conductive wire instead of resistor. Also there was finded out that for better commutation there is need to change not only the PWM duty, but also frequency.

Turinys

ĮVADAS.....	8
1. RINKOJE ESANČIŲ PANAŠIŲ ĮRENGINIŲ ANALIZĖ.....	9
1.1 Atmel ATAVRMC100.....	9
1.2 Teco electric devices BLH.....	11
1.3 MicroChip PICDEM™ MC LV.....	12
1.4 Rinkos apžvalgos išvados.....	13
2. NAUDOJAMŲ TECHNOLOGIJŲ APŽVALGA.....	17
2.1 Half tiltas (half-bridge).....	17
2.2 Įtampos daliklis.....	18
2.3 Srovės matavimas.....	19
2.4 UART.....	21
3. ĮRENGINIO STRUKTŪRINĖ SCHEMA.....	23
4. ĮRENGINIO ELEKTRINĖ PRINCIPINĖ SCHEMA.....	24
4.1 Mikrovaldiklio grandinė.....	24
4.2 Įtampos stabilizavimas.....	25
4.3 Variklio valdymo tiltas.....	25
4.4 Įtampos dalikliai.....	26
4.5 Srovės matavimas.....	27
4.6 Indikacinio diodo schema.....	28
5. SPAUSDINTO MONTAŽO PLOKŠTĖ.....	29
6. PROGRAMINĖ ĮRANGA.....	30
7. BANDYMŲ REZULTATAI.....	33
7.1 Mikrovaldiklio grandinė.....	33
7.2 Įtampos daliklių grandinės bandymai.....	33
7.3 Srovės matavimo grandinės bandymai.....	34
7.4 Variklio valdymo tilto bandymai.....	34
7.5 Valdiklio bandymų išvados.....	35
IŠVADOS.....	37
LITERATŪRA.....	38

Priedai

1 Priedas. Bešepetėlio nuolatinės srovės variklio valdiklio elektrinė principinė schema.....	40
2 Priedas. Bešepetėlio nuolatinės srovės variklio valdiklio spaudinto montažo plokštė.....	41
3 Priedas. Bešepetėlio nuolatinės srovės variklio valdiklio detalių išdėstymas.....	42
4 Priedas. Bešepetėlio nuolatinės srovės variklio valdiklio detalių sąrašas.....	43
5 Priedas. Programinė įranga BLDC.c.....	44
6 Priedas. Programinė įranga motor.h.....	52

Paveikslai

1.1 pav. Atmel ATAVRMC100 plokštė.....	10
1.2 pav. TECO electric devices BLH150W valdiklis.....	12
1.3 pav. Microchip PICDEM MC LV plokštės vaizdas.....	14
2.1 pav. H tiltas.....	17
2.2 pav. Trijų fazių įtampos keitiklis.....	18
2.3 pav. Įtampos daliklis.....	19
2.4 pav. Srovės matavimo variantai. Figure 1A – aukštos pusės, figure 1B – žemos.....	20
3.1 pav. Projektuojamo valdiklio struktūrinė schema.....	23
4.1 pav. Mikrovaldiklio grandinė.....	24
4.2 pav. Įtampos reguliavimo ir stabilizavimo elektrinė principinė schema.....	25
4.3 pav. Valdymo tilto fragmentas.....	26
4.4 pav. Įtampos daliklis.....	26
4.5 pav. Srovės matavimo elektrinė principinė schema.....	27
4.6 pav. Šviesos diodo schema.....	28
6.1 pav. Pagrindinis algoritmas.....	31
6.2 pav. Pertraukimo algoritmas.....	32
7.1 pav. Dažninė aukštos pusės tilto charakteristika.....	34
7.2 pav. Suformuoti variklio valdymo žingsniai.....	35

Lentelės

1 lentelė. Komerčių valdiklių palyginimo lentelė.....	14
2 lentelė. Srovės matavimo būdų palyginimas.....	20
3 lentelė. Standartinis UART duomenų perdavimas.....	22
4 lentelė. Įtampos daliklių bandymų rezultatai.....	33
5 lentelė Srovės matavimo grandinės bandymų rezultatai.....	34

IVADAS

Elektroniniai įrenginiai nuolat tobulėja, keičiasi ir genda. Kartais iš sugedusių įrenginių galima panaudoti likusius nepažeistus komponentus ir pritaikyti juos savoms reikmėms. Norint tai padaryti reikia išsiaiškinti kaip sudaryti, koku būdu veikia šitie įrenginiai, kad galėtume juos pritaikyti ir valdyti. Vieni tokių komponentų randamų kompiuterinėje technikoje yra bešepetėliniai nuolatinės srovės varikliai.

Kas tai yra bešepetėliniai nuolatinės srovės varikliai? Tai yra varikliai suprojektuoti kaip ir šepetėliniai varikliai, tačiau pagrindinis skirtumas yra elektromagnetinio lauko keitimas ne besisukant rotoriumi, o valdant elektroniškai. Paprastai tokiuose varikliuose elektromagnetinį lauką kuria fiksuotoje padėtyje esantis statorius, ant kurio yra apvijos. Rotorius yra nuolatinis magnetas, kuris sukasi kintant elektromagnetiniam laukui.

Tokie varikliai yra ilgaamžiškesni už šepetėlinius ir reikalauja mažiau priežiūros, tačiau jų valdymas yra komplikuoatas. Kadangi rotorius yra nuolatinis magnetas, nėra besidėvinčių dalių, kurias reiktėtų nuolat prižiūrėti ir keisti. Todėl šio tipo variklius yra patogiu naudoti nuolat veikiančiuose įrenginiuose, kuriuose reikalingas patikimumas, bei ilgas tarnavimo laikas.

Bešepetėliniai nuolatinės srovės varikliai turi tiesioginę priklausomybę tarp srovės stiprumo ir galios, įtampos ir apsisukimų skaičiaus. Jie yra valdomi elektroniškai vietoj mechaninio valdymo, kuris naudojamas šepetėliniuose varikliuose. Paprastai elektromagnetas nejuda, jusa nuolatiniai magnetai, todėl korpusas išlieka stabilus. Susidaro problema kaip perduoti srovę elektromagnetui, todėl naudojami išmanūs elektroniniai valdikliai.

Savo baigiamajame darbe projektuoju nesudėtingą bešepetėlinio variklio valdiklį, kuriuo būtų galima reguliuoti sukimosi kryptį bei apsisukimų skaičių, kuris išvestų reikiamą informaciją, bei stebėtų maitinimo įtampos lygį.

Darbo tikslas: suprojektuoti ir pagaminti bešepetėlinio nuolatinės srovės variklio valdiklį.

Darbo uždaviniai:

1. Išanalizuoti panašius įrenginius;
2. Pasirinkti reikiamus elementus ir technologijas projektuojamam įrenginiui;
3. Suprojektuoti įrenginį;
4. Pagaminti veikiančią prototipą;
5. Atlikti bandymus.

1. RINKOJE ESANČIŲ PANAŠIŲ ĮRENGINIŲ ANALIZĖ

Šiuo metu yra sukurta daug komercinės paskirties bešepetėlinių nuolatinės srovės variklių valdiklių. Šiame skyriuje norėčiau trumpai aptarti komercinius įrenginius skirtus valdyti tokius variklius. Taip pat palyginti jų techninius parametrus, kainą ir kitus aspektus.

1.1 Atmel ATAVRMC100

Atmel korporacija yra pasaulinė lyderė projektuojant ir gaminant mikrovaldiklius, talpuminius lietimui jautrius sprendimus, pažangios logikos, maišytų signalų, netriniosios operatyviosios atminties, radio dažnio (RF) komponentų. Turėdama vieną plačiausių pramonės intelektinės nuosavybės portfelių, Atmel gali elektronikos pramonėje teikti pilnai paruoštus sprendimus, pritaikytus pramonei, vartotojams, apsaugai, komunikacijoms, kompiuteriams ir automobilių pramonei.[1]

Ši kompanija taip pat pasiūlė sprendimus įrenginių valdymui. Vienas tokių įrenginių yra AVR® ATAVRMC100 plokštė. Ši plokštė yra skirta valdyti trijų fazių bešepetėlinius nuolatinės srovės variklius su jutikliminiu ir be jutikliminio būdais. Be jutikliminis būdas padėties nustatymui naudoja grįžtamojo elektromagnetinio lauko jėgą.

Rinkinį sudaro plokštė, trijų fazių bešepetėlis nuolatinės srovės variklis ir demonstracinė programinė įranga. Tai leidžia vartotojams greitai išbandyti mikrovaldiklio suderinamumą valdant didelių apsisukimų turinčius bešepetėlius nuolatinės srovės variklius. Šis rinkinys gali tarnauti kaip bandomoji plokštė. Mažai kainuojantys AVR vystymo įrankiai padaro derinimo darbus lengvesnius, o C kalba parašyti pagrindiniai kodai gali būti lengvai pritaikomi vartotojo projektuojamiems valdymo sprendimams.[2]

ATAVRMC100 pagrindinės funkcijos:

- AT90PWM3-16SQ SO32 mikrovaldiklis (2.7 – 5.5V)
- Plokštėje įmontuotas LIN transyveris Atmel ATA6661
- Jėgos tiltas variklio valdymui
- Holo daviklių arba bejutuminė konfigūracija
- Nulio kirtimo įtampos aptikimas
- Techninės įrangos viršsrovių aptikimas
- Variklio įtampos matavimas

- Įtampos reguliatorius įmontuotas plokštėje (5V)
- AVR Studio® programinė įranga
- Maitinimo įtampos indikavimas žaliu šviesos diodu
- ISP jungtis programavimui ir klaidų aptikimui
- Sistemos laikrodis vidinis RC
- Daug priėjimo taškų testavimui
- Rekomenduojama veikimo įtampa nuo 12 V iki 16V (4A)
- Rekomenduojama veikimo temperatūra nuo 0°C iki 70°C
- Plokštės išmatavimai 75 mm x 55 mm



2 pav. Atmel ATAVRMC100 plokštė

Plokštė yra pristatoma su mikrovaldikliu AT90PWM3 ir jau užprogramuota testine variklio valdymo programine įranga. Tereikia prijungti variklį kaip nurodyta techninėje dokumentacijoje bei nustatyti trumpiklius pagal variklio valdymo tipą. Prijungti maitinimo įtampą kuri turi būti tarp 12 V ir 16 V bent 4A. Tada sukonfigūruoti trumpiklius pagal variklio valdymo tipą, su Holo davikliais ar be jų. Viską padarius teisingai, pasileidžia testinė variklio valdymo programa ir jis pradeda sukstis.

Perprogramuoti valdymo plokštei naudojamas standartinis Atmel ISP (In System Programming) programatorius. Atmel taip pat gamina programatorius ir platformas mikrovaldikliams, tokias kaip AVR ISP, STK500, JTAGICE mkll, AVR Dragon ir tt. Iš gamintojos puslapiu užsiregistravus galima parsisiųsti programinę įrangą mikrovaldiklių programavimui AVR Studio.[3]

1.2 Teco electric devices BLH

TED TECO Electric & Machinery Co., Ltd O Electro Devices Co., Ltd yra dukterinė pasaulinio variklių giganto TECO Electric & Machinery Co., Ltd kompanija. Buvo įkurta 1985 m kaip TECO departamentas, gaminant žingsninius variklius, 1998 m buvo pripažinta TECO būti profesionaliu žingsninių variklių gamintoju. Per pastaruosius 24 metus, TED sėkmingai pardavė savo gaminius amerikų ir europų didžiausiems gamintojams.

TED turi stiprias galimybes gaminti aukštos kvalifikacijos reikalaujančius produktus atitinkančius pačius aukščiausius tarptautinius reikalavimus. Remdamasi sukaupta patirtimi kompanija išplėtė savo produkciją, pasiūlydama pirkėjams didelį asortimentą gaminių. Hibridinius žingsninius variklius, servo variklius, servo variklių valdiklius, valdikliai, bešepetėliai varikliai ju valdikliai yra kompanijos pagrindinė produkcija. TECO investavo į variklių industriją ir dabar yra penkta didžiausia pasaulio variklių gamintoja. Siūlyti klientams geriausius produktus ir greita aptarnavimą yra kompanijos tikslas. Kompanijos preciziški varikliai yra pagaminti didelio magnetinio skvarbumo dalių ir unikalios magnetinio pasipriešinimo technologijos. Tai lemia mažą triukšmą, vibracijas, adekvatyvią galią ir sklandų veikimą. Šie aukštos kokybės ir didelio patikimumo produktai žavi klientus. Dauguma kompanijos produktų yra sertifikuoti kaip neturintys defektų, kas dar labiau žavi ir tenkina klientus.[4]

Kai kurios aplikacijos, kurios reikalauja greičio valdymo, dauguma žmonių rinkusių indukcinis, šepetėlinis ar invertuojamčius variklius. Paprastai tokie varikliai yra didelio dydžio, nestabilaus greičio, didelės temperatūros, lėto greičio keitimo, mažo efektyvumo ir reikalauja didelės priežiūros. Sujungiant materialų mokslą ir elektronikos technologijas, atsiranda naujas pasirinkimas bešepetėliai nuolatinės srovės varikliai. Nes jie neturi šepetėlių, priežiūra nereikalauja tiek daug dėmesio, sukimosi momentas nepriklauso nuo greičio, lengva keisti sukimosi kryptį. Tam pačiam galios atidavimui šie varikliai sunaudoja pusę tiek energijos, kiek paprasti varikliai. Net po ilgos veikimo trukmės, varikliai išlaiko žemą temperatūrą ir didelį naudingumą.[5]

TECO kompanija siūlo du valdiklius, kurių pagrindiniai parametrai yra:

- Didelis greičio juostos plotis
- Mažas skleidžiamas triukšmas, ilgaamžiškumas
- Mažas dydis, didelis efektyvumas
- Atitinkantis kainą, lengvai aptarnaujamas
- Lengvas sukimosi krypties keitimas

- Vidinis arba išorinis greičio valdymas



1.2. Pav. TECO electric devices BLH150W valdiklis

Šio gamintojo produktas skirtas valdyti pramonei skirtus variklius. Visiškai paruoštas naudojimui. Greičio reguliavimas galimas įmontuotais plokštėje potenciometrais arba prijungiant išorinį potenciometrą. Paprasta pakeisti variklio sukimosi kryptį. Tai visiškai paruoštas produktas naudojimui, nereikia nieko nustatinti ar programuoti, tik prijungti variklį, maitinimo įtampą ir viskas veikia.

1.3 PICDEM™ MC LV

Microchip Technology Inc. yra lyderiaujanti kompanija gaminanti mikroprocesorius ir analoginius puslaidininkius, palaikanti mažos rizikos produktų bandymus, mažiausią bendrą sistemos kainą ir greičiausią pristatymą į rinką tūkstančiams produktų visame pasaulyje. Įsikūrusi Chandler Arizonos valstijoje, JAV, Microchip siūlo techninę pagalbą, greitą pristatymą ir kokybę.[6]

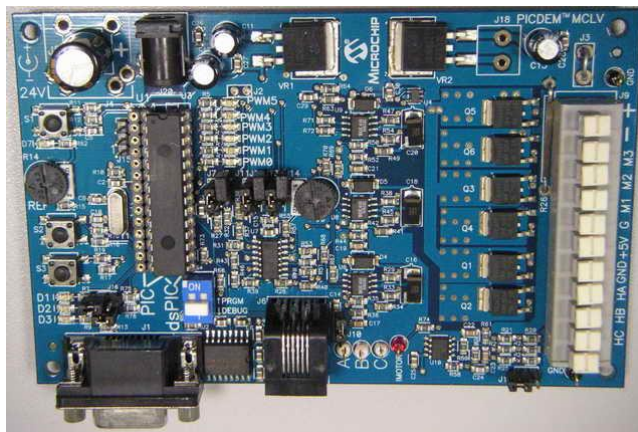
Microchip siūlo sprendimus dideliame galios spektre, tai 8 bitų, 16 bitų bei 32 bitų mikrovaldikliai, su galinga architektūra, lanksčiomis atminties galimybėmis, lengvai naudojamus kūrimo įrankius, pilną techninę dokumentaciją.

The PICDEM MC LV vystymo plokštė paprasta, paruošta naudojimui bešepetėlinių nuolatinės srovės variklių valdiklis, padaryta naudojant dsPIC30F2010, dsPIC30F3010 ir dsPIC30F4012. PICDEM MC LV plokštė valdymo sekciją, naudojančią skaitmeninį signalų valdymą, trijų fazių maitinimo įtampos keitiklį, klaidų stebėjimo schemą, temperatūros jutiklį, indikuojančius šviesos diodus, serijinį prievadą sujungimui su kompiuteriu ir ICSP jungtį programavimui. Plokštė turi integruotus sprendimus besensoriniam arba sensoriniam valdymui. Ši plokštė turi viską ko reikia valdyti variklius nuo 10 V iki 48 V, kai maksimali srovė 2.2 A, naudojant sensorinį arba besensorinį algoritmus.[7]

PICDEM MC LV plokštės parametrai:

- Triejų fazių pagrindinės įtampos keitimo tiltas
- Judesio jutimo įvadai
- Viršsrovių apsaugą, kuri yra programuojama
- Temperatūrinį jutiklį su I²C sąsaja
- Testinius taškus srovės matavimui
- Greičio reguliavimo potenciometrą
- Parodomąją programinę įrangą
- Aktyvų RS-232 portą
- MPLAB jungtis

Ši plokštė pristatoma su jau užprogramuotu mikrovaldikliu PIC18F2431, todėl tereikia tik prijungti variklį, nustatyti plokštės veikimą pagal variklio tipą, sensorinis ar besensorinis, prijungti maitinimą ir variklis pradės sukstis. Variklio greitis keičiamas naudojant integruotą potenciometrą. Ši plokštė turi RS-232 prievadą, kurio pagalba plokštę galima prijungti prie kompiuterio ir naudojant nemokamą programinę įrangą valdyti variklį. Turi integruotą I²C sąsają. Šioje plokštėje taip pat yra integruoti vidiniai maitinimo šaltiniai: vienas 15 V variklio pagrindinei įtampai, kitas 5 V mikrovaldiklio ir kitų elektroninių sistemų darbui užtikrinti.



1.3. pav. Microchip PICDEM MC LV plokštės vaizdas

1.4 Rinkos apžvalgos išvados

Šiame savo darbe paminėjau kelis valdiklius, skirtus valdyti bešepetėlinius nuolatinės srovės variklius. Kiekvienas valdiklių turi savo privalumus ir trūkumus. Rinkos apžvalgos išvadose pateikiu lentelę (1 lentelė), kurioje smulkiau surašyti pagrindiniai šių valdiklių

privalumai ir trūkumai. Šioje lentelėje atsispindi techninės, elektroninės, valdymo, programavimo kalbų galimybės.

1 lentelė.

Komercinių valdiklių palyginimo lentelė.

Valdiklis	ATAVRMC100	PICDEM™ MC LV	Teco BLH150W
Mikrovaldikliai	<p>AT90PWM3</p> <ul style="list-style-type: none"> • 8 bitų; • 16 MHz maksimalus; • 8KB Flash; • 10 bitų ADC; <p>Mikrovaldiklis įlituotas plokštėje todėl jį pakeisti yra gana sudėtinga.</p>	<p>PIC18F2431</p> <ul style="list-style-type: none"> • 8 bitų; • 8 MHz vidinis; • 10 MHz maksimalus; • 16KB Flash; • 10 bitų ADC; <p>Plokštė turi 28 vietų lizdą, todėl mikrovaldiklį paprasta pakeisti arba vietoje jo naudoti DSC.</p>	<p>Gamintojas nepateikia jokių duomenų apie naudojamą mikrovaldiklį.</p>
Valdomų variklių tipai	<ul style="list-style-type: none"> • Su jutikliais; • Be jutiklių; • 12 – 16 V; • 4 A max; 	<ul style="list-style-type: none"> • Su jutikliais; • Be jutiklių; • 10 – 48 V; • 2.2 A max; 	<ul style="list-style-type: none"> • Su jutikliais; • Be jutiklių; • 24 V ± 10%; • 20 A max;
Programavimo galimybės	<p>Pateikiamas su demonstracine programine įranga, galimas programavimas per ISP naudojant gamintojo pateikiamą</p>	<p>Pateikiamas su demonstracine programine įranga, galimas programavimas su bet koku PIC programatoriumi, turi galimybę</p>	<p>Neturi jokių programavimo galimybių, galima reguliuoti tik sukimosi kryptį ir greitį.</p>

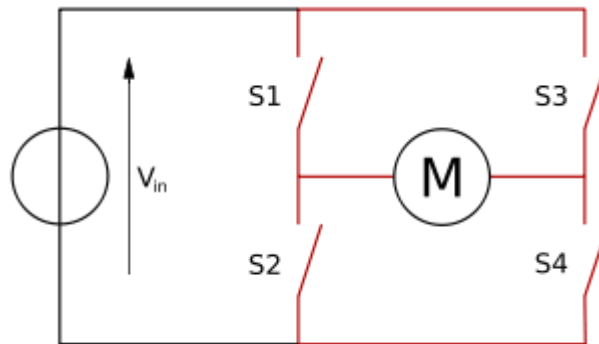
	programinę įrangą.	valdyti per kompiuterį.	
Bendros įpatybės	<ul style="list-style-type: none"> - Valdo sukimosi kryptį ir greitį; - Turi integruotus srovės keitiklius; - Valdo abiejų tipų variklius; - Turi apsaugas nuo viršsrovių; - Turi per žemos įtampos indikacijas; 		
Suderinamumas	Atmel ir Microchip gaminiai yra technologiškai suderinami naudojant I ² C sąsają. BLH150W skirtas veikti kaip autonominis įrenginys nepriklausantis kitiems.		
Privalumai	<ul style="list-style-type: none"> - Paprastas naudoti; - Gamintojas pateikia pagrindinį kodą, tolesniam bandymui; - Lengva integruoti į kitas sistemas; - Programavimui naudojama AVR Studio, su kuria galima parašyti, sukompiliuoti programą ir užprogramuoti mikroprocesorių; - Turi laisvą prievadų valdymo plėtojimui; 	<ul style="list-style-type: none"> - Lengvai suprantamas; - Turi daug šviesos diodų indikacijų apie veikimo būklę; - Turi integruotą greičio ir sukimosi krypties valdymą; - Galima valdyti per kompiuterį su pateikiama nemokama programine įranga; - Turi daug prievadų plėtojimui; - Paprasta integruoti į kuriamas sistemas; - Turi lizdą mikrovaldikliui, todėl sugedus jam paprasta pakeisti; 	<ul style="list-style-type: none"> - Labai paprastas naudoti; - Nereikalauja jokių programavimų ir tt.; - Paprastai valdomas greitis ir sukimosi kryptis; - Visiškai paruoštas sprendimas pramonei;
Trūkumai	<ul style="list-style-type: none"> - Įlituotas mikrovaldiklis, 	<ul style="list-style-type: none"> - Sąlyginai dideli matmenys; 	<ul style="list-style-type: none"> - Visiškai neturi plėtojimo

	<p>todėl jam sugedus sudėtinga pakeisti kitu;</p> <ul style="list-style-type: none">- Neturi tiesioginės jungties su kompiuteriu;- Mažas valdomų variklių spektras;	<ul style="list-style-type: none">- Mažas valdomų variklių spektras;	<p>galimybių;</p> <ul style="list-style-type: none">- Dideli matmenys;- Aukšta kaina;- Nesuderinamas su kitais įrenginiais;
--	--	--	---

2. NAUDOJAMŲ TECHNOLOGIJŲ APŽVALGA

2.1. Half tiltas (half-bridge).

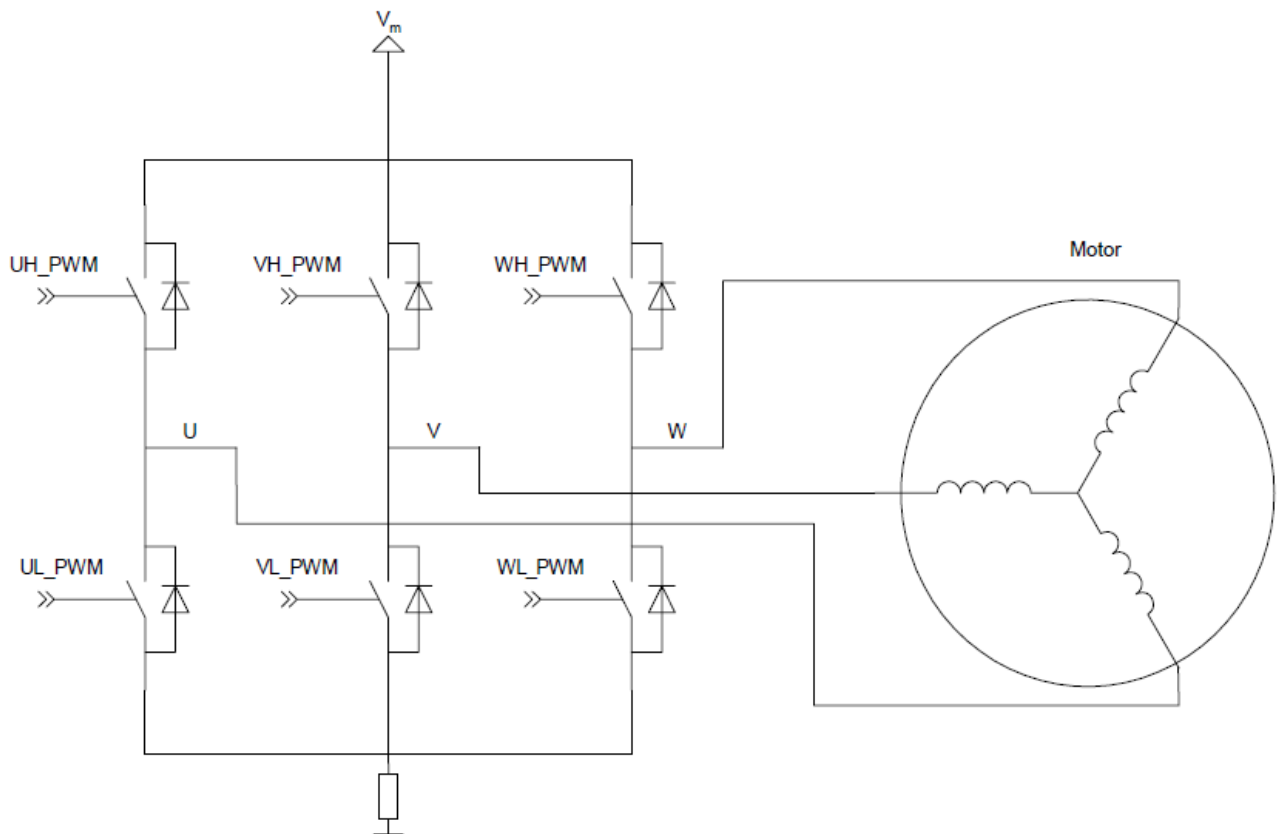
H tiltas yra elektroninis įrenginys, kuris leidžia įtampą paduoti abiejomis kryptimis. H tiltai prieinami kaip mikroschemos arba gali būti surenkami iš diskrečiųjų komponentų. Half tiltas yra dalis H tilto, tai viena atšaka kuria galima keisti įtampą (2.1. pav.). Kaip matome paveikslėlyje, H tiltas turi dvi puses (S1, S2 ir S3, S4). Viena H tilto pusė yra half tiltas. Naudojant Half tiltą galima keisti įtampą vienam apkrovos kontaktui.



2.1. pav. H tiltas

Terminas H tiltas kilo pagal grandinės formą, ji primena H raidę. Tipiškas H tiltas sudarytas iš keturių jungiklių (Half tiltas iš dviejų jungiklių). Kai įjungiami pirmas ir ketvirtas jungikliai (2.1.pav.) varikliui paduodama įtampa ir jis sukasi pagal laikrodžio rodyklę. Kai įjungiami trečias ir antras jungikliai, įtampa pakeičia tekėjimo kryptį ir variklis pradeda sukis prieš laikrodžio rodyklę. Svarbu paminėti tai, kad pirmas ir antras arba trečias ir ketvirtas jungikliai negali būti įjungiami tuo pačiu metu. Tai sukeltų trumpą jungimą.

Kitas šios schemos variantas naudoja tik du tranzistorius vienoje apkrovos pusėje. Tokia konfigūracija vadinama Half tiltu. Half tiltas naudojamas kai kuriuose įtampos keitikliuose ir stiprintuvuose. Half tiltas yra vadinamas kaip pusė pilno H tilto. Naudojant half tiltą galima padaryti trijų fazių keitiklį. Trijų fazių keitiklis yra pagrindinė kintamos srovės variklių dalis. Half tiltuose naudojami diodai, kurie sumažina trumpo jungimo pavojų. Tokios half tilto modifikacijos paprastai naudojamos valdyti varikliams (2.2. pav.).



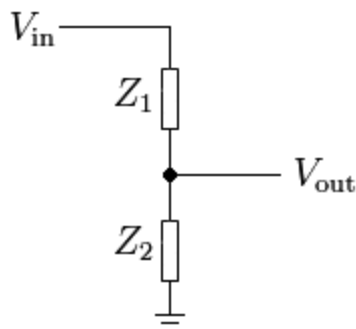
2.2. Pav. Triejų fazių įtampos keitiklis

2.2 Įtampos daliklis.

Elektronikoje įtampos daliklis (taip pat žinomas kaip potencialų daliklis) yra linijinė schema, kuri sukuria išėjimo įtampą, kuri yra dalis įėjimo įtampos. Įtampos dalinimas priklauso nuo komponentų reikšmių, per kurą teka įtampa.

Kaip pavyzdys, įtampos daliklį sudaro du rezistoriai (arba potenciometras) sujungti nuosekliai. Taip yra daroma siekiant gauti nurodytą įtampą arba gauti mažesnę įtampą, kuri gali būti išmatuota, taip pat gali būti naudojamas kaip signalo slopintuvas kai vyrauja žemi dažniai. Kai yra nuolatinė įtampa ir žemi dažniai, įtampos daliklis tiksliai padalina įtampą ir turi gerą tikslumą. Kai reikalingas įtampos dalinimas platesnėje dažnių juostoje, lygiagrečiai įtampos daliklio varžai gali būti prijungtas kondensatorius, tam, kad kompensuotų apkrovos talpą, elektros įtampos perdavime, talpuminis įtampos daliklis yra naudojamas tam, kad matuoti aukštą įtampą.

Įtampos daliklis, priklausantis nuo žemės, yra sukuriamas sujungiant du rezistorius nuosekliai, kaip parodyta 2.3 paveiksle. Įėjimo įtampa pereina per nuosekliai sujungtus Z1 ir Z2 rezistorius, o išėjimo įtampa yra įtampa per Z2. Z1 ir Z2 gali būti sukuriama kombinuojant įvairius elementus, tokius kaip rezistoriai, induktoriai ir kondensatoriai[9].



3 Pav. Įtampos daliklis

Taikant Omo dėsnį, įėjimo ir išėjimo įtampų santykis randamas pagal sekančias formules:

$$V_{out} = \frac{Z_2}{Z_1 + Z_2} * V_{in}; \quad (2.1)$$

Įrodymas:

$$V_{in} = I * (Z_1 + Z_2); \quad (2.2)$$

$$V_{out} = I * Z_2; \quad (2.3)$$

$$I = \frac{V_{in}}{Z_1 + Z_2}; \quad (2.4)$$

$$V_{out} = V_{in} * \frac{Z_2}{Z_1 + Z_2}; \quad (2.5)$$

Šios schemos perdavimo funkcija (taip pat žinoma kaip perdavimo santykis) yra:

$$H = \frac{V_{out}}{V_{in}} = \frac{Z_2}{Z_1 + Z_2}; \quad (2.6)$$

3.2 Srovės matavimas

Paprastai yra du būdai matuoti tekančią srovę, tai yra srovės matavimas aukštojoje pusėje, arba srovės matavimas žemojoje pusėje. Šie du būdai turi savų privalumų ir trūkumų. Dažnai yra svarstoma, kuris būdas geresnis, tačiau sprendimas dažniausiai yra priimamas įvertinus reikalingą sistemą ir pasirenkant kainos ir tikslumo atžvilgiu geriausią variantą.

Pargindinis skirtumas tarp šių dviejų būdų yra rezistoriaus jungimas grandinėje, kurioje matuojama srovė. Abu (aukštos ir žemos pusės) variantai pateikti 2.4 paveiksle. Kaip matome skirtumas nedidelis, vienu atveju rezistorius jungiamas tarp maitinimo šaltinio ir apkrovos, antru atveju – tarp apkrovos ir žemės kontaktų[10].

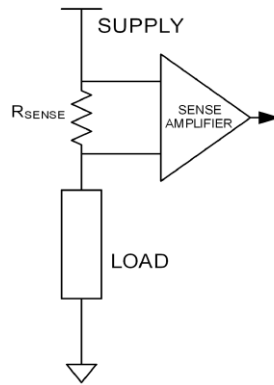


FIGURE 1A

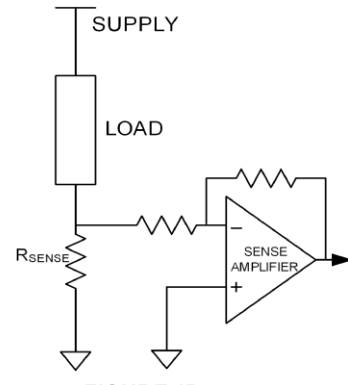


FIGURE 1B

4 Pav. Srovės matavimo variantai. Figure 1A – aukštos pusės, figure 1B – žemos.

Kaip matome pateiktame paveiksle, įtampos kritimas per rezistorių yra matuojamas panaudojant operacinius stiprintuvus, tačiau jie nevisada yra reikalingi.

Pagrindiniai aukštos ir žemos pusių srovės matavimų privalumai ir trūkumai[11][12] pateikti 2 lentelėje.

2 lentelė.

Srovės matavimo būdų palyginimas

	Aukštos pusės srovės matavimas	Žemos pusės srovės matavimas
Privalumai	<ul style="list-style-type: none"> - Aptinka betkokį gedimą ir gali atlikti reikiamus veiksmus. - Nesukelia apkrovos žemės pusėje. 	<ul style="list-style-type: none"> - Nesudėtinga konstrukcija. - Didelis tikslumas. - Maža kaina.
Trūkumai	<ul style="list-style-type: none"> - Reikia labai tikslaus rezistoriaus. - Turi atlaikyti dideles, greitai kintančias įtampas. 	<ul style="list-style-type: none"> - Atsiranda nepageidaujama apkrova žemės grandinėje. - Kartais prireikia atskiro prijungimo kelio.
Kada rinktis	Kai nėra pageidaujama papildoma apkrova žemės pusėje. Taupant kainą, eliminuojant kelią iki žemės.	Šis matavimo būdas yra geriausias tada, kai sistema toleruoja papildomą apkrovą žemės grandinėje.

4.2 UART.

Universalus nesinchroninis siūstuvus/imtuvus (UART) yra dalis kompiuterių elektronikos, kuri siunčia ir priima duomenis lygiagrečiais ir nuosekliais prievadais. UART dažnai naudojama komunikacijų standartuose, tokiuose kaip: EIA, RS – 232, RS – 422 arba RS – 485. Žodis universalus reiškia, kad duomenų formatas ir perdavimo greičiai yra konfigūruojami. Elektrinių signalų lygiai ir metodai paprastai yra apdorojami išorinių įrenginių.

UART paprastai yra individuali integruota schema, naudojama nuosekliai komunikacijai tarp kompiuterio ir kito įrenginio. Dabar UART yra integruotas į daugelį mikrovaldiklių. Dvigubas UART arba DUART sujungia du atskitus UART į vieną schemą. Dauguma šiuolaikinių mikroschemų gaminamos su integruotu sinchroniniu UART. Tokie įrenginiai turi USART(universalus sinchroninis/nesinchroninis siūstuvus/imtuvus).

UART paima atskirus bitus ir siunčia juos individualiai nustatyta tvarka. Priėmimo įtaisas surenka bitus ir sujungia juos baitus. Kiekvienas UART turi shift registrus, kurie yra pagrindinis metodas, verčiant tarp lygiagrečiųjų ir nuosekliųjų prievadų. Nuoseklusis siuntimas yra, kai duomenų perdavimas vyksta vienu laidu, siunčiant pitus nustatytu formatu. Lygiagretusis siuntimas yra, kai bitai yra siunčiami laidais, priklausančiais kiekvienam bitui.

UART paprastai nepriima ir nesiunčia duomenų tarp skirtingų įrenginių tiesiogiai. Atskiri komunikavimo įrenginiai yra naudojami signalų lygių keitimui. Išoriniai signalai gali būti įvairių formų. Kaip pavyzdys yra standartizuoti įtampų lygiai, tokie kaip: RS – 232, RS – 422, RS – 485 iš EIA. Istoriskai, srovė buvo naudojama telegrafų tinkluose. Kai kurios sistemos nenaudoja laidų, kaip pavyzdžiai: optiniai tinklai, IrDA ir Bluetooth. Kai kurie tinklai naudoja nešančiojo signalo moduliaciją, su ir be laidų. Kaip pavyzdžiai: garso signalų moduliacija telefoniniuose vertikliuose, RF moduliacija, DC – LIN, kuri naudojama įtampos linijų komunikacijai.

UART duomenų perdavimas gali būti įgyvendinamas naudojant tik vieną laidą, be jokio priimančio įrenginio duomenų siuntimo siunčiamajam įrenginiui. Full duplex naudojamas kai įrenginiai vienu metu gali siųsti ir priimti informaciją. Half duplex – kai įrenginiai paeiliui keičia siuntimo ir priėmimo signalus[13].

Dešinysis duomenų bitas visada siunčiamas pirmas. Jei naudojama parity, parity bitas siunčiamas po duomenų bitų, bet anksčiau stop bito. Kai įrenginiai nesiunčia duomenų, įtampos lygis yra aukštas. Šis būdas istoriskai atkeliavo iš telegrafų tinklų, kur buvo laikomas aukštas įtampos lygis, rodantis, kad nepažeista perdavimo linija. Duomenų siuntimas pradedamas žemu loginiu start bitu, tada siunčiami duomenų bitai (paprastai 8, tačiau šiuolaikinės sistemos

leidžia pasirinkti bitų skaičių), tada siunčiamas pasirenkamas parity bitas. Duomenų siuntimas baigiamas vienu ar daugiau loginių aukštų stop bitų.

Start bitas parodo priimančiajam įrenginiui, kad yra pradedami siųsti duomenys. Tada seka nuo penkių iki aštuonių duomenų bitų. Tada seka, jei pasirinktas parity bitas. Sekantys vienas arba du bitai (aukšto loginio lygio) parodo, kad duomenų siuntimas yra baigtas. Kadangi start bitas yra žemo loginio lygio, o stop bitas – aukšto, visada yra du garantuoti signalo pasikeitimai.

Jeigu linija yra laikoma žemame loginiame lygyje ilgiau, nei laikas skirtas bitui persiųsti, yra pertraukimo sąlyga, kuri yra aptinkama UART. Standartinį UART duomenų perdavimą parodo 3 lentelė.

5 lentelė.

Standartinis UART duomenų perdavimas.

Bito numeris	1	2	3	4	5	6	7	8	9	10	11
	Start bitas.	5 – 8 duomenų bitai.								Stop bitai.	
	Start	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Stop	

3. ĮRENGINIO STRUKTŪRINĖ SCHEMA

Šiame skyriuje pateikiama projektuojamo įrenginio struktūrinė schema, pateikiama 3.1 paveiksle ir jos paaiškinimai.



3.1 pav. Projektuojamo valdiklio struktūrinė schema.

Visą variklio valdymą atlieka mikrovaldiklis. Mikrovaldiklis stebi maitinimo šaltinio įtampą, kad ji nebūtų per aukšta arba per žema. Taip pat mikrovaldiklis atlieka variklio valdymą, pagal gautus įtampų lygius iš variklio valdymo grandinės. Mikrovaldiklis apdoroja srovės lygį, kurį sunaudoja variklis. Duomenis mikrovaldiklis siunčia į kompiuterį RS – 232 sąsaja.

Variklio valdymo grandinė valdo variklio sukimosi kryptį ir greitį, priklausomai nuo gautų signalų iš mikrovaldiklio.

Įtampų ir srovės matavimo grandinė matuoja maitinimo šaltinio įtampą. Ši grandinė taip pat matuoja įtampų lygius, kurie naudojami variklio valdymui. Grandinė taip pat turi matuoti variklio naudojamą srovę, kad ji nebūtų per didelė.

Mikrovaldiklio maitinimui naudojamas paprastas įtampos keitiklis, kuris paverstų maitinimo šaltinio įtampą į reikalingą mikrovaldikliui.

Duomenų išvedimas vyksta panaudojus mikrovaldiklio UART sąsają, taip pat reikalingas TTL lygių keitiklis, kad mikrovaldiklis „susišnekėtų“ su kompiuteriu. Šis keitiklis panaudojus MAX232 ar panašią mikroschemą, bus įgyvendintas atskirai. Taip pat numatyta perspektyva prijungti kitus išorinius įvedimo ar išvedimo įrenginius.

Visas projektuojamas valdiklis bus vienoje spausdinto montažo plokštėje.

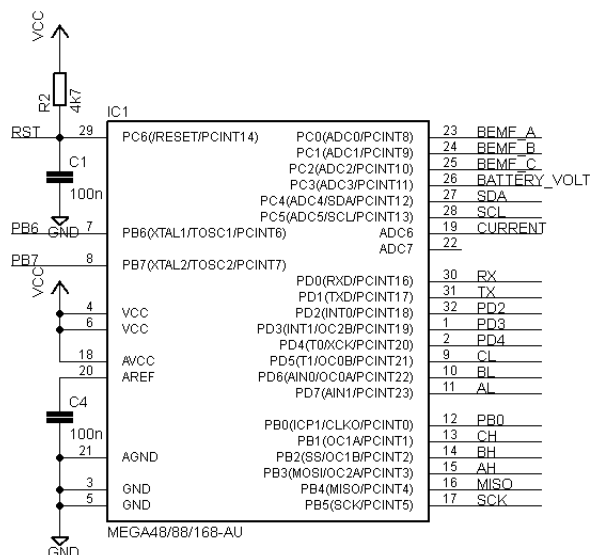
4. ĮRENGINIO ELEKTRINĖ PRINCIPINĖ SCHEMA

Vadovaujantis anksčiau pateikta įrenginio struktūrine schema suprojektuota elektrinė principinė schema. Schema susideda iš kelių dalių: įtampos valdymo grandinės, mikrovaldiklio grandinės, variklio valdymo grandinės ir įtampų ir srovės stebėjimo grandinės. Visa suprojektuota valdiklio principinė elektrinė schema pavaizduota 1 priede.

4.1 Mikrovaldiklio grandinė

Visą valdiklio darbą atliks mikrovaldiklis, todėl teisingas mikrovaldiklio pasirinkimas turi didelę įtaką valdiklio veikimui. Įvertinus reikalingus parametrus valdikliui, tokius kaip: mikrovaldiklio dažnis, palaikomos sąsajos, skaitliukų skaičius ir kojelių skaičius pasirinktas Atmel Atmega 168 mikrovaldiklis. Šis mikrovaldiklis turi 8 MHz vidinį rezonatorių, kurio visiškai pakanka valdikliui, todėl nereikia naudoti išorinio. Taip pat mikrovaldiklis palaiko UART, I²C sąsajas, kurios išplečia valdiklio duomenų perdavimo galimybes. Pasirinktas mikrovaldiklis turi 6 PWM išvadus, nors nuspręsta naudoti tik 3. Mikrovaldiklis turi tris skaitliukus, kurie bus išnaudojami variklio valdymui. Mikrovaldiklis turi 8 kanalų analoginį skaitmeninį keitiklį, iš kurių bus naudojami 5, įtampoms ir srovei matuoti. Šis mikrovaldiklis turi 32 kojeles, iš kurių 23 yra programuojami įėjimai/išėjimai.

Mikrovaldiklio grandinės dalis pavaizduota 4.1 paveiksle.



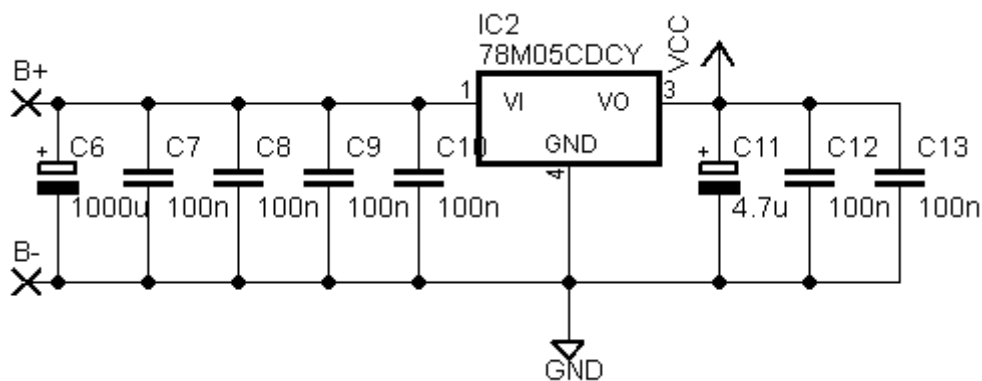
4.1 pav. Mikrovaldiklio grandinė.

Kaip matome 4.1 paveiksle mikrovaldiklio reset signalas laikomas aukštame loginiame lygyje. Taip pat matome du 100 nF kondensatorius, kurie naudojami parazitiniams reiškiniams slopinti.

4.2. Įtampos stabilizavimas.

Valdymo grandinės įtampai nustatyti ir stabilizuoti naudojamas teigiamos įtampos reguliatorius 78M05CDCY. Šis reguliatorius pasirinktas, nes naudojant jį nereikia papildomų komponentų, kas sumažina ne tik išlaidas, bet ir sutaupo vietą spausdinto montažo plokštėje [2 priedas]. Įtampos stabilizavimui naudojami kondensatoriai. Maitinimo šaltinio įtampos stabilizavimui naudojamas C6 1000 uF kondensatorius, nes formuojant signalus varikliui valdyti, pasireiškia įtampos netolydumas. Mikrovaldiklio įtampos stabilizavimui naudojamas C11 4.7 uF kondensatorius. Kaip matome 4.2 paveiksle visi kiti kondensatoriai 100 nF, jie yra skirti parazitiniams reiškiniams, kurie atsiranda prie aukštesnių dažnių slopinimui.

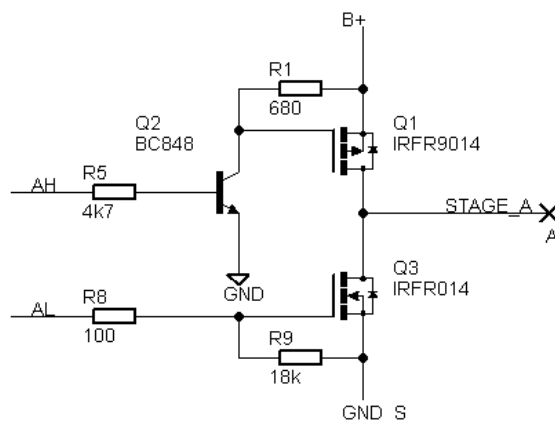
Įtampos stabilizavimo elektrinė principinė schema pavaizduota 4.2 paveiksle.



4.2 pav. Įtampos reguliavimo ir stabilizavimo elektrinė principinė schema.

4.3. Variklio valdymo tiltas.

Variklio valdymui naudojami trys identiški half H tiltai. Būtent šie tiltai formuoja reikiamus signalus varikliui. Tiltai yra trys, jie yra identiški, nes nepriklausomai nuo variklio tipo (star arba delta variantai), naudojama trijų laidų kombinacija. 4.3 paveiksle pateiktas valdymo tilto fragmentas.



4.3 pav. Valdymo tilto fragmentas.

Kaip matome 4.3 paveiksle pagrindinį valdymą atlieka tranzistoriai Q1 ir Q3. Šie tranzistoriai pasirinkti įvertinus reikalingus parametrus valdikliui, taipogi tranzistoriai turi atsargą, todėl esant būtinybei galima prijungti ir galingesnius variklius, kurie naudoja daugiau srovės.

Kaip matyti iš 4.3 paveikslo, aukštojoje tilto pusėje valdymui naudojamas papildomas tranzistorius Q2. Jis veikia kaip raktas. Šiuo būdu tranzistorius Q1 yra nuolat uždarytas, o initializavus PWM signalą, kuriuo valdomos visų tiltų aukštosios pusės tranzistorius atsidaro. Žemosios pusės valdymui naudojamas rezistorius R8, jis apriboja srovę, tekančią į tranzistorių Q3. Pagal Omo dėsnį:

$$I = \frac{U}{R}; \quad (4.1)$$

$$R = \frac{U}{I}; \quad (4.2)$$

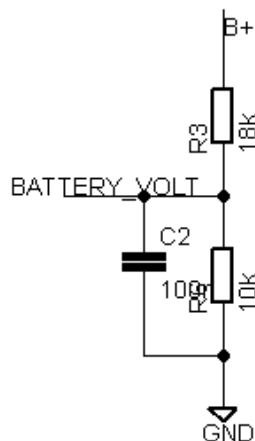
Čia I – tekanti srovė, U – įtampa, R – varža, tada:

$$R = \frac{5}{0.05} = 100\Omega; \quad (4.3)$$

Rezistorius R9 naudojamas palaikyti žemą įtampos lygį, kai nėra valdymo signalo.

4.4 Įtampos dalikliai.

Valdiklyje įtampos dalikliai matuoja variklio fazių įtampą ir maitinimo šaltinio įtampą. Įtampos dalikliai yra keturi, jie visi identiškų parametrų. Kadangi buvo nuspręsta, jos maksimali maitinimo šaltinio įtampa gali būti 14 voltų, pagal tokius parametrus paskaičiuoti įtampos daliklių parametrai, kad tenkintų mikrovaldiklio analoginių skaitmeninių įėjimų parametrus. 4.4 paveiksle pateiktas vienas įtampos daliklis, kaip matoma šuntuojantis kondensatorius panaudotas siekiant išvengti nepageidaujamų reiškinių, kintant dažniui.



4.4 pav. Įtampos daliklis.

Kadangi maksimali maitinimo šaltinio įtampa 14 voltų, o mikrovaldiklio palaikoma – 5 voltai, įtampos dalinimo koeficientas parinktas artimesnis 3, pagal tai apskaičiuoti komponentų parametrai.

$$V_{out} = \frac{R_2}{R_1 + R_2} * V_{in}; \quad (4.4)$$

$$\frac{V_{out}}{V_{in}} = \frac{R_2}{R_1 + R_2}; \quad (4.5)$$

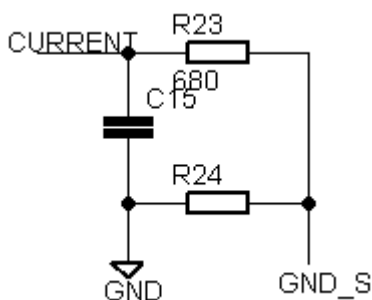
Pasirinkta R2 – 10 kΩ, pagal tai apskaičiuota R1;

$$\frac{5}{14} = \frac{10k}{10k + R_1}; \quad (4.6)$$

$$R_1 = \frac{14 * 10k}{5} - 10k = 18k; \quad (4.7)$$

4.5 Srovės matavimas

Variklio valdiklyje svarbu matuoti variklio vartojamą srovę, todėl suprojektuotas žemosios pusės srovės matavimas. Šį būdą nuspręsta pasirinkti dėl nesudėtingos konstrukcijos ir plataus pritaikymo galimybių. 4.5 paveiksle pavaizduota srovės matavimo elektrinė principinė schema. Kaip matome, paveiksle panaudotas šuntuojantis kondensatorius ir papildomas 680 Ω rezistorius, tokiu būdu siekiama įgyvendinti idėją, kad srovė būtų matuojama tiesiogiai ir jos reikšmės nebereikėtų programiškai keisti.

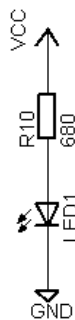


4.5 pav. Srovės matavimo elektrinė principinė schema.

Kaip matome 4.5 paveiksle R24 turi būti kuo artimesnė 0 ty, <0,01 Ω. Įgyvendinus šią sąlygą, įtampos kritimas turėtų būti kaip ir analoginio skaitmeninio keitiklio gaunama reikšmė.

4.6 Indikacinio diodo schema

Šiame variklio valdiklyje panaudotas vienas šviesos diodas (LED). Diodas tik parodo ar yra maitinimo įtampa mikrovaldiklio grandinei. Šviesos diodo elektrinė principinė schema pateikta 4.6 paveiksle.



4.6 pav. Šviesos diodo schema

$$I = \frac{U}{R}; \quad (4.8)$$

$$R = \frac{U}{I}; \quad (4.9)$$

$$R = \frac{U_{in} - U_{led}}{I_{led}}; \quad (5.0)$$

Čia R – reikalingas rezistorius, U_{in} – esanti įtampa, U_{led} – šviesos diodui reikalinga įtampa, I_{led} – šviesos diodui reikalinga srovė.

SMD 0805 šviesos diodui reikalinga įtampa ~2 voltai, per diodą tekančią srovę pasirenkame kuo mažesnę, nes diodas tik indikuoja įtampos buvimą. Tada,

$$R = \frac{5-2}{0,005} = 600 \Omega; \quad (5.1)$$

Kadangi turime likusių 680Ω rezistorių iš valdymo tiltų, juos ir panaudojame.

5. SPAUSDINTO MONTAŽO PLOKŠTĖ

Pagal suprojektuotą elektrinę principinę schemą, suprojektuota ir pagaminta spausdinto montažo ploštė. Plokštė dvipusė, takeliai plokštėje vyrauja nuo 0.3 milimetrų iki 0.8 milimetrų. Plokštėje panaudoti trys žemės signalo užpylimai. Laidai (maitinimo šaltinio, variklio) lituojami, padidinti patikimumui padarytos skylutės papildomam laidų fiksavimui. Projektuojant plokštę palikta kontaktų grupė programatoriaus prijungimui, taip pat palikta kontaktų grupė leidžianti išplėsti valdiklio galimybes. Visi komponentai, išskyrus 100uF kondensatorių panaudoti SMD korpusuose. Rezistoriai ir neelektrolitiniai kondensatoriai 0805 korpusuose, mikrovaldiklis TQFP 32 korpuse, tranzistoriai TO 252 ir SOT 23, įtampos reguliatorius SOT 223 korpuse. Suprojektuotos spausdinto montažo plokštės dydis yra trys centimetrai ant 6 centimetrų. Suprojektuotos valdiklio spausdinto montažo plokštės takelių išdėstymas pateikiamas antrame priede. Spausdinto montažo plokštės detalių išdėstymas pateiktas trečiame priede, panaudotų detalių sąrašas pateikiamas 4 priede.

6. PROGRAMINĖ ĮRANGA

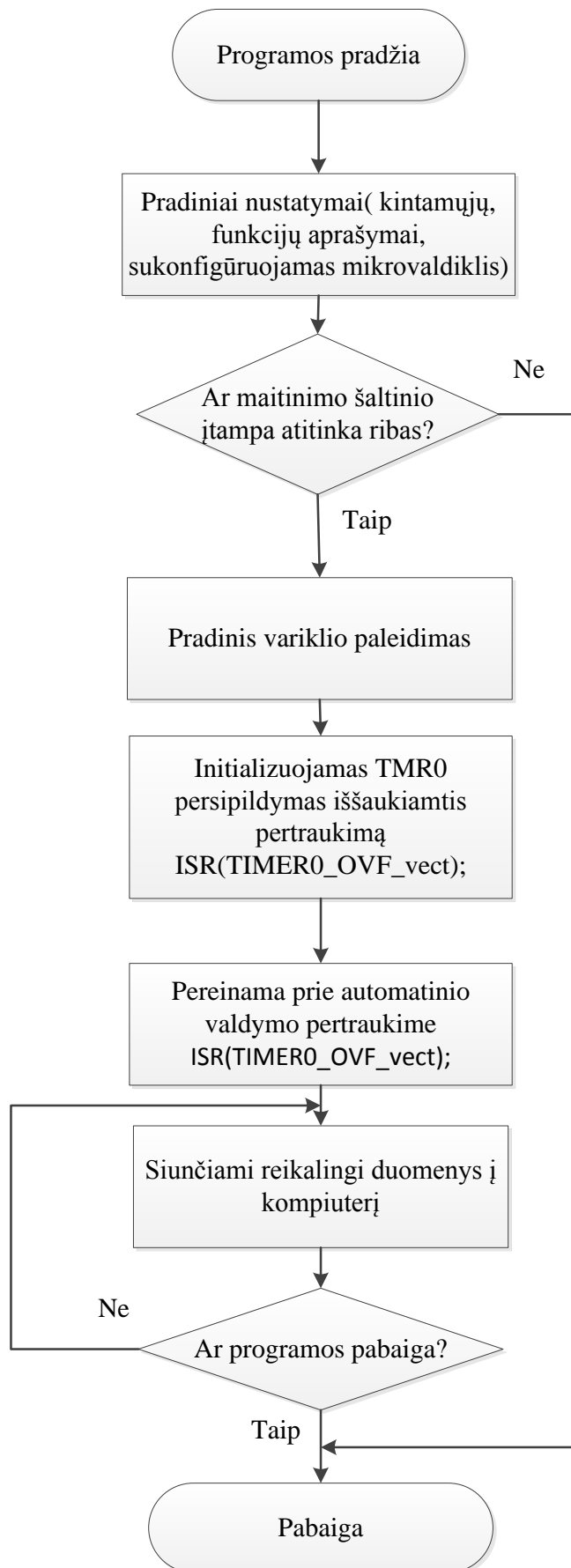
Suprojektuotam variklio valdikliui parašyta programinė įranga. Visa programinė įranga parašyta C kalba, panaudojant Atmel AVR Studio 4 programą. Siekiant palengvinti ir supaprastinti pagrindinės programinės įrangos tekstą sukurtas mail.h failas, kuriame aprašomos ir nustatomos naudojamos reikšmės. Šiame faile aprašomi priskyrimai, maitinimo šaltinio minimalios ir maksimalios reikšmės, srovės maksimali reikšmė. Tai leidžia lengviau koreguoti valdiklio darbą nekeičiant ir neieškant reikšmių pagrindiniame faile. Programinės įrangos algoritmai pavaizduoti 6.1 ir 6.2 paveiksluose. Programinės įrangos kodas C kalba pateikiamas penktame ir šeštame prieduose.

Pagrindinis programos tikslas valdyti variklį, nustatant sekantį žingsnį pagal reikšmes gautas iš įtampos daliklių. Taip pat matuoti maitinimo šaltinio įtampos lygį, kuris negali nukristi žemiau nustatytos reikšmės. Taip pat programinė įranga stebi variklio naudojamą srovę, kuri neturi pakilti aukščiau nustatytos reikšmės. Pagrindiniai duomenys išvedami į kompiuterį naudojant RS – 232 sąsają.

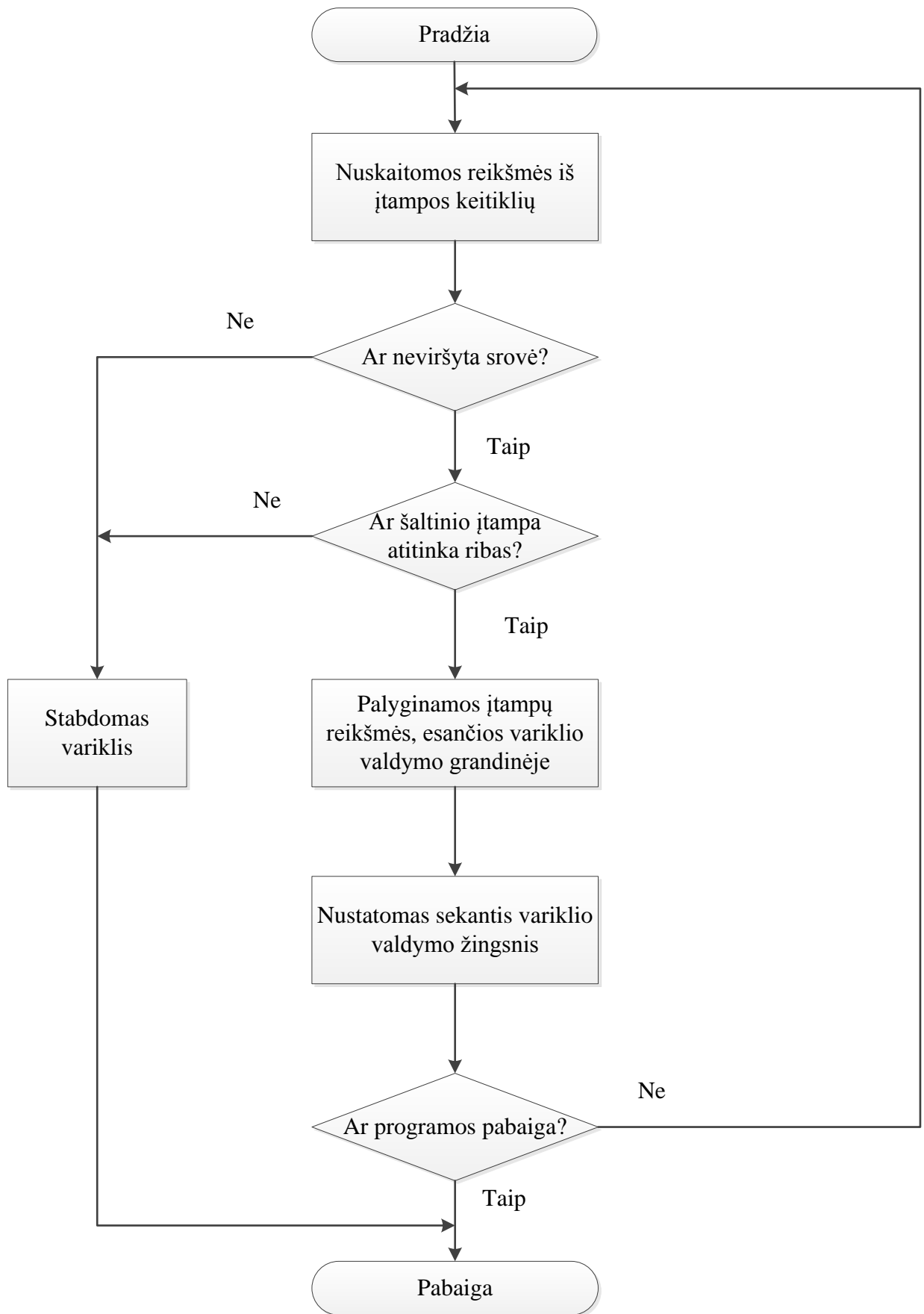
Programos pradžioje aprašomi naudojami kintamieji, funkcijos, sukonfigūruojami mikrovaldiklio registrai, įjungiamos reikiamos funkcijos (analoginiai skaitmeniniai keitikliai, impulso pločio moduliacija, UART). Tada patikrinama maitinimo šaltinio įtampa, ar ji yra aprašytų normų ribose, jei maitinimo šaltinio įtampa yra nustatytų normų ribose vykdomos tolimesnės procedūros. Pradiniam variklio startavimui paleidžiama žingsnių seka, reikalinga, kad indukuotusi įtampa ir būtų galima nustatyti sekančius žingsnius pagal gautas reikšmes. Reikšmių nuskaitymui ir sekančio žingsnio nustatymui naudojamas pertraukimas, kuris įvyksta persipildžius skaitikliui. Taipogi nuolat yra matuojama maitinimo šaltinio įtampa ir vartojama srovė, esant neatitikimams variklis yra stabdomas.

Pertraukimo metu nustatoma kuri reikšmė lygi pusei maitinimo įtampos. Kadangi per šešių žingsnių ciklą įvyksta du tokie pusės įtampos kirtimai, papildomai palyginama su esančia kito išvado įtampa ir nustatomas sekantis reikalingas žingsnis. Pertraukimo metu taip pat stebima naudojama srovė ir maitinimo šaltinio įtampa, jei reikia variklis yra stabdomas.

Variklio apsisukimų skaičius valdomas į variklį paduodama įtampa, todėl naudojama impulso pločio moduliacija įtampos keitimui, kuri taip pat keičia ir variklio apsisukimų skaičių. Dėl šios variklio savybės ir yra reikalingas dinaminis sekančių žingsnių nustatymas.



6.1 pav. Pagrindinis algoritmas



6.2 pav. Pertraukimo algoritmas

7. BANDYMŲ REZULTATAI

Suprojektuoto ir pagaminto variklio valdiklio prototipo bandymai buvo atliekami tikrinant kiekvieną jį sudarančios grandinės dalį atskirai, kol buvo prieita prie galutinio pilno valdiklio bandymo. Bandymų metu duomenys buvo siunčiami į kompiuterį per RS – 232 sąsają.

7.1 Mikrovaldiklio grandinės bandymas

Mikrovaldiklio grandinės bandymai buvo atlikti naudojant skirtingą programinę įrangą. Bandymų metu siekta įsitikinti, kad nėra padaryta klaidų valdiklio projektavimo metu. Bandymų metu buvo patikrinti visi mikrovaldiklio kontaktai, buvo patikrinta ar nepadaryta klaidų projektavimo metu ir visi signalai nukeliauja ten kur reikia. Taipogi bandymų metu patikrinta UART ir kiti išvadai, kurie buvo palikti papildomiems įrenginiams. Bandymų metu sėkmingai patikrinti programavimo kontaktai. Mikrovaldiklio grandinėje nerasta jokių klaidų.

7.2 Įtampos daliklių grandinės bandymai

Šiais bandymais buvo siekta įsitikinti, kad įtampos dalikliai suprojektuoti teisingai ir mikrovaldiklis gauna tikslią informaciją. Bandymų metu naudota programinė įranga nuskaitė įtampų reikšmes, programiškai jas pakeitė į įtampos lygį ir išvedė į kompiuterį per RS – 232 sąsają. Papildomam bandymų tikslumui buvo naudojami du skaitmeniniai multimetrai. Bandymų rezultatai pavaizduoti 4 lentelėje.

6 lentelė

Įtampos daliklių bandymų rezultatai

Įtampa rodoma ant maitinimo šaltinio	Maitinimo įtampos matavimo daliklis	A įtampos matavimo daliklis	B įtampos matavimo daliklis	C įtampos matavimo daliklis	Pirmo multimetromatavimai	Antro multimetromatavimai	Paklaida
5 V	610 mV	605 mV	612 mV	608 mV	605 mV	608 mV	<1%
7.5 V	803 mV	803 mV	808 mV	806 mV	804 mV	802 mV	<1%
12 V	1401 mV	1405 mV	1406 mV	1405 mV	1403 mV	1402 mV	<1%

Kaip matyti aukščiau pateiktoje lentelėje įtampos daliklių paklaida yra mažesnė už vieną procentą. Įtampos reikšmės skiriasi dėl mikrovaldiklio darbo. Kadangi paklaida yra labai maža, tai tenkina suprojektuoto valdiklio reikalavimus.

7.3 Srovės matavimo grandinės bandymai

Bandymu metu buvo matuojama srovė ir lyginama su mikrovaldiklio pateikiama reikšme. Bandymų metu buvo naudojama papildyta programinė įranga, kuri buvo naudojama įtampos daliklių bandymuose. Bandymų rezultatai pateikiami 5 lentelėje.

7 lentelė

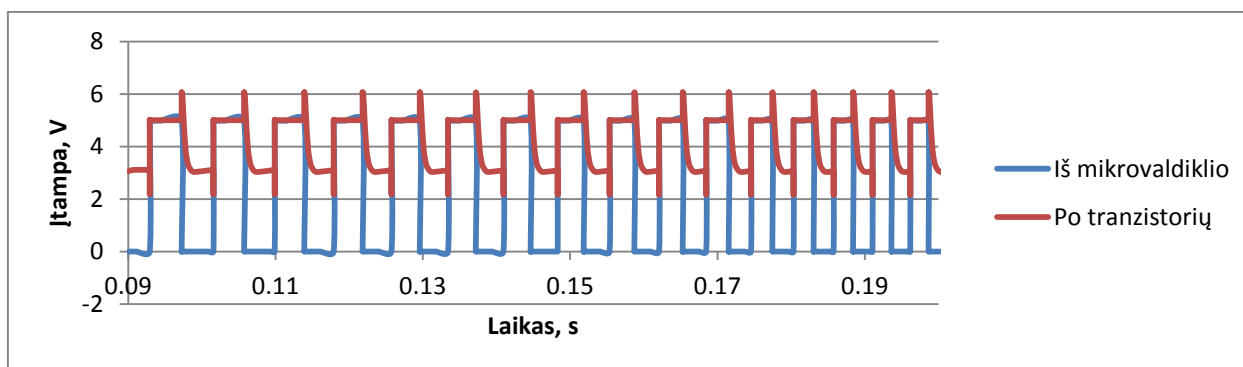
Srovės matavimo grandinės bandymų rezultatai

Mikrovaldiklio rodoma srovė	Pirmo multimetrom rodoma srovė	Antro multimetrom rodoma srovė	Paklaida
300 mA	330 mA	331 mA	10 %
1000 mA	1030 mA	1031 mA	3 %

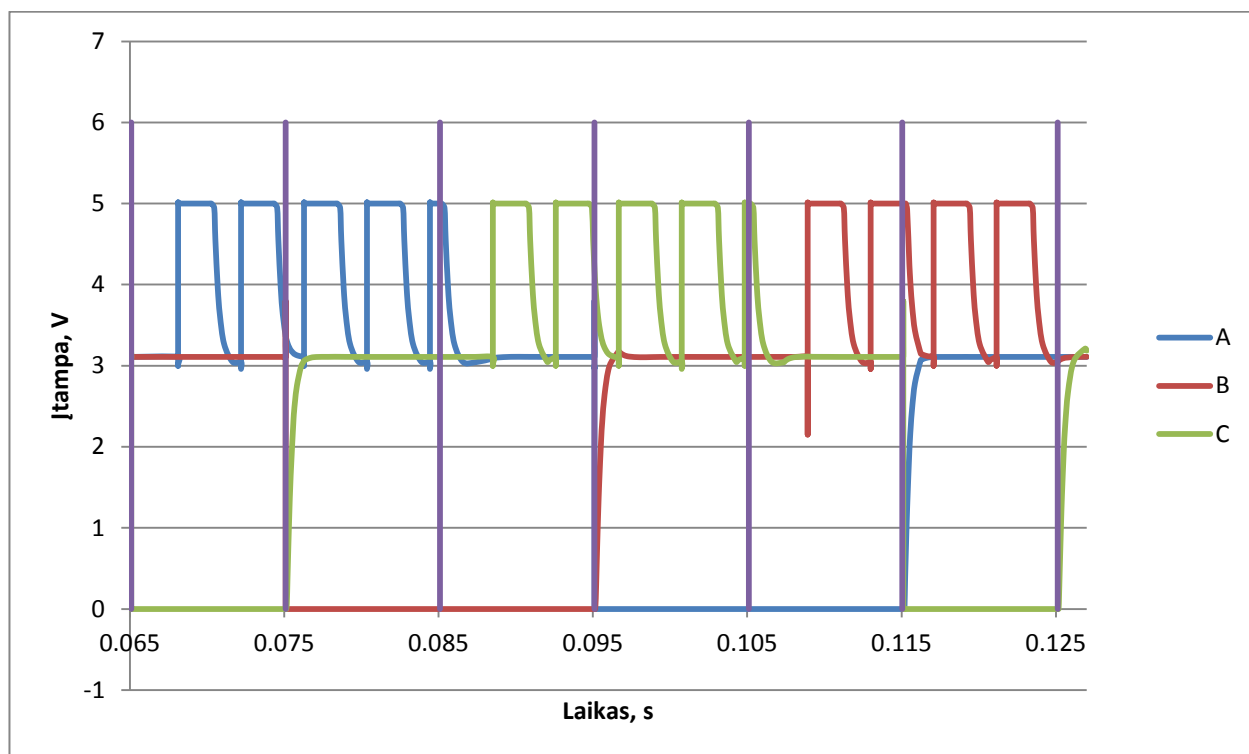
Kaip matyti aukščiau pateiktoje lentelėje mikrovaldiklio matuojamos srovės paklaida skiriasi, tai įvyksta todėl, nes mikrovaldiklis matuoja srovę šimtų miliamperų tikslumu. Įvertinus paklaidą ir tai, jog valdiklio apsaugai reikalingas srovės matavimas amperais, pasiektas tikslumas tenkina išsikeltus tikslus projektuojant valdiklį.

7.4 Variklio valdymo tilto bandymai

Bandant variklio valdymo tiltą, pirmiausia buvo tikrinamos dažninės savybės. Naudojant trumpą programos kodą buvo didinamas ir matuojamas dažnis. Šio bandymo tikslas išsiaiškinti maksimalų perjungimo dažnį. Bandymų rezultatai pateikiami 7.1, 7.2, 7.3 paveiksluose.



7.1 pav. Dažninė aukštos pusės tilto charakteristika



7.2 pav. Suformuoti variklio valdymo žingsniai.

Kaip matyti aukščiau pateiktuose grafikuose, didėjant perjungimo dažniui pastebimi labai dideli iškraipymai, kurių nėra mažesniuose dažniuose. Atlikus valdymo grandinės bandymus išsiaiškinta, kad optimaliausi rezultatai pasiekiami, kai perjungimo dažnis nėra didesnis už 400 Hz, ty, vieną perjungimą galima atlikti kas 2.5 mili sekundės.

8 Valdiklio bandymų išvados

Atliekant suprojektuoto valdiklio bandymus, buvo stengiamasi patikrinti ar nėra padaryta klaidų, projektuojant elektrinę principinę ir spausdinto montažo plokštę. Neradus klaidų projektavime ir gamybos procese, buvo tikrinamos valdiklį sudarančios pagrindinės grandinės įtakančios valdiklio darbą. Atliekant atskirų grandinių bandymus, buvo pastebėta, jog būtų greičiau ir efektyviau atlikti bandymus, jei būtų numatyti testavimo taškai spausdinto montažo plokštės projektavimo metu. Tai ne tik palengvintų bandymų atlikimą, bet ir supaprastintų tolimesnės eksploatacijos metu iškilusių problemų paiešką.

Atliekant tolimesnį pilną valdiklio bandymą iškilo problemų su srovės matavimo grandine. Tikriausiai dėl per didelės tekančios srovės sugedo srovės matavimo grandinė. Pradėjus ieškoti problemos nustatyta, kad perdegė matavimui naudotas rezistorius. Šią problemą nuspręsta išspręsti vietoje rezistoriaus prijungus 0.22 AWG skersmens laidininką. Laidininko ilgis buvo parenkamas bandymų keliu, kad nereikėtų programiškai koreguoti srovės reikšmės ir taip apsunkinti

mikrovaldiklio darbą. Bandynų metu rastas tinkamas laidininko ilgis, kuris yra 66 centimetrai. Po šio incidento kilo minčių srovės matavimą įgyvendinti vietoje rezistoriaus ar pašalinio laidininko padarant reikiamos varžos apkrovą spausdinto montažo plokštėje. Tokiu atveju reiktų parinkti tinkamą takelio plotį atsižvelgiant į apkrovą. Bandymų metu nustatyti pagrindiniai valdiklio parametrai: matuojamos įtampos paklaida $< 1 \%$, matuojamos srovės paklaida $< 10 \%$, variklio komutacijos dažnis 400 Hz, kas leidžia pasiekti didelį apsisukimų skaičių.

IŠVADOS

Darbo atlikimo metu buvo surasti ir išanalizuoti panašūs įrenginiai. Atliekant panašių įrenginių analizę, kai kurie sprendimai buvo pritaikyti projektuojant valdiklį. Analizuojant panašius įrenginius pastebėta, kad jie turi ribotas išplėtimo galimybes. Analizuoti įrenginiai paprastai buvo didelių išmatavimų ir sąlyginai didelės kainos. Išanalizavus panašius įrenginius buvo pasirinktos reikiamos technologijos optimaliam projektuojamo įrenginio įgyvendinimui.

Analizuojant panašius įrenginius pasirinktos ir pritaikytos pagrindinės technologijos, tokios kaip: įtampos ir naudojamos srovės matavimas bei pagrindinis valdymo tiltas. Analizuojant panašius įrenginius atsisakyta pradinės idėjos aukštus ir žemus valdymo signalus veikti impulso pločio moduliacija ir nuspresta, kad užteks tik aukštus reikalingus signalus veikti impulso pločio moduliacija.

Išanalizavus ir pasirinkus reikiamas technologijas buvo atliktas elektrinės principinės ir spaudinto montažo plokščių projektavimas naudojant Cadsoft EAGLE 5.11.0 programą. Projektuojant įrenginį reikėjo susikurti kelių komponentų bibliotekas, nes standartinėse jų nebuvo. Projektuojant spaudinto montažo plokštę buvo stengiamasi laikantis visų projektavimo taisyklių optimaliai išnaudoti vietą ir suprojektuoti įrenginį kiek imanoma mažesnių gabaritų.

Suprojektavus įrenginį buvo pagamintas prototipas ir atlikti jo bandymai. Prototipo bandymų metu nustatytos pagrindinės valdiklio charakteristikos. Valdiklio maitinimo įtampa gali būti nuo 5 iki 14 voltų, įtampos matavimo paklaida nustatyta < 1 %. Valdiklis valdo variklį naudodamas ilgalaikę 5 amperų apkrovą, nustatytas srovės matavimo tikslumas < 10 %, didėjant srovės dydžiui paklaida mažėja. Nustatytas komutacijos žingsnių dažnis 400 Hz, kas leidžia valdyti variklį didelėmis apsukomis.

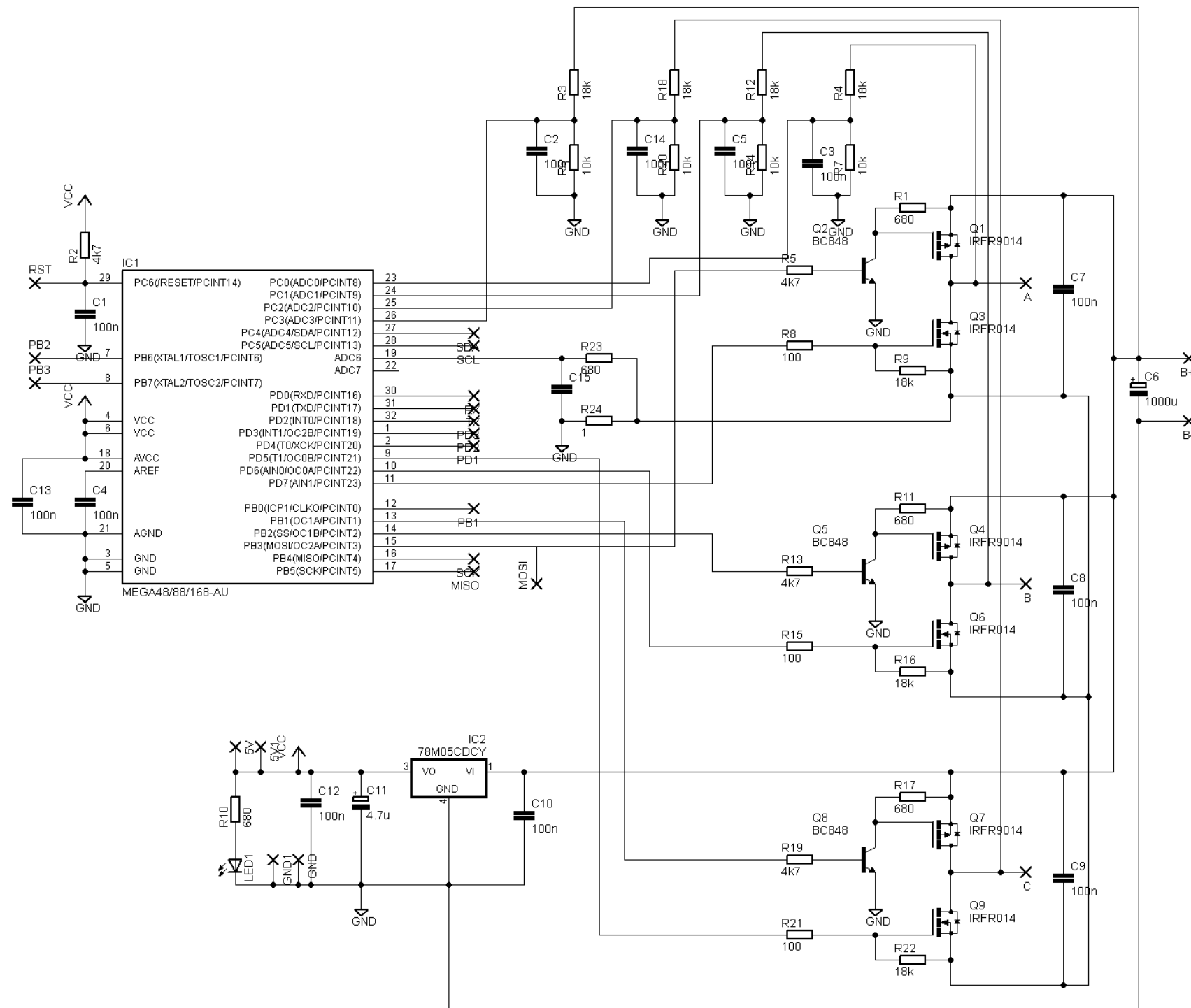
Bandymų metu nustatyta, jog variklio apsisukimų skaičius priklauso nuo variklio polių skaičiaus. Nustatyta, jog pasirinkus per didelę PWM signalo skverbę variklio apsisukimų skaičius artėja prie maksimalios ribos ir variklis sustoja. Atliekant bandymus pastebėta, jog norint efektyviau valdyti variklio apsisukimų skaičių reikia keisti ne tik PWM signalo skverbę, bet ir dažnį.

LITERATŪRA

1. About Atmel Corporation [Žiūrėta 2012-10-21] Prieiga per internetą:
<<http://www.atmel.com/About/corporate/factsheet.aspx>>
2. ATAVRMC 100 [Žiūrėta 2012-10-21] Prieiga per internetą:
<<http://www.atmel.com/Images/doc7551.pdf>>
3. Atmel AVR Studio [Žiūrėta 2012-10-21] Prieiga per internetą:
<http://www.atmel.com/microsite/atmel_studio6/>
4. Ted Motors Company [Žiūrėta 2012-10-21] Prieiga per internetą:
<<http://www.tedmotors.com/company-profile.htm>>
5. Brushless DC Driver [Žiūrėta 2012-10-21] Prieiga per internetą:
<<http://www.tedmotors.com/brushless-dc-driver.htm>>
6. Microchip Technology Inc. [Žiūrėta 2012-10-21] Prieiga per internetą:
<http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=72>
7. PICDEM [Žiūrėta 2012-10-21] Prieiga per internetą:
<<http://ww1.microchip.com/downloads/en/DeviceDoc/01033B%2013.pdf>>
8. Review of Circuits [Žiūrėta 2013-01-20] Prieiga per internetą:
<http://www.tedpavlic.com/teaching/osu/ece209/support/circuits_sys_review.pdf>
9. Current sensing [Žiūrėta 2013-01-20] Prieiga per internetą:
<http://www.planetanalog.com/document.asp?doc_id=527795&site=planetanalog>
10. Low side current sensing [Žiūrėta 2013-03-20] Prieiga per internetą:
<<http://www.ti.com/analog/docs/microsite.tsp?sectionId=560&tabId=2181µsiteId=7>>
11. High side current sensing [Žiūrėta 2013-03-20] Prieiga per internetą:
<<http://www.ti.com/analog/docs/microsite.tsp?sectionId=560&tabId=2182µsiteId=7>>
12. Adam Osborne, *An Introduction to Microcomputers Volume 1: Basic Concepts*, Osborne-McGraw Hill Berkeley California USA, 1980 ISBN 0-931988-34-9 pp. 116-126

PRIEDAI

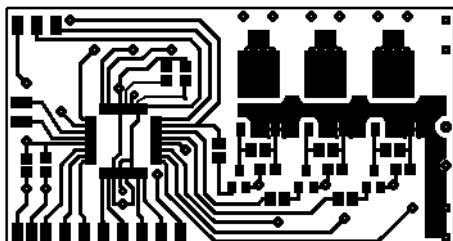
1 PRIEDAS



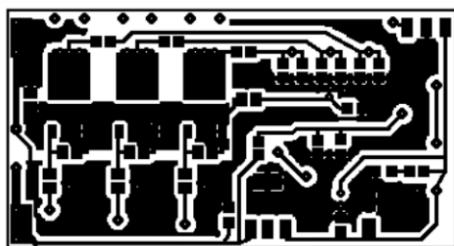
	Mastelis	Raide		Mase	
TECHNOLOGIJOS	Diplomantas	A. Paulauskas		Besepetelinio nuolatinės srovės	
SU FAKULTETAS	Vadovas	D. Dervinis	Parasas	Data	variklio valdiklis
2013	Lapas	Lapu	Konsultantas		
	1	4	Recenzentas	D. Balbonas	
			Kat. vedejas	G. Daunys	

Bešepetėlinio nuolatinės srovės variklio valdymas, spausdinto montažo plokštė

Viršutinis sluoksnis



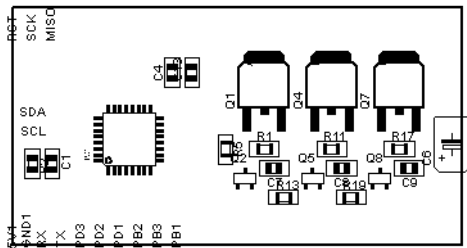
Apatinis sluoksnis



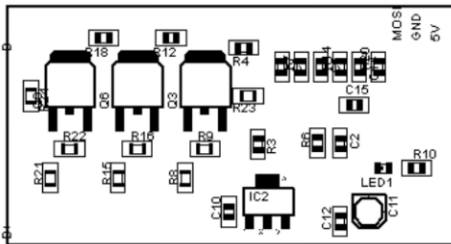
		Mastelis	Raidė			Masė			
TECHNOLOGIJOS ŠU FAKULTETAS					Parašas	Data	Bešepetėlinio nuolatinės srovės variklio valdymas		
		Diplomantas	A. Paulauskas						
		Vadovas	D. Dervinis						
2013	Lapas	Lapų	Konsultantas						
	2	4	Recenzentas	D. Balbonas					
			Kat. Vedėjas	G. Daunys					

Bešepetėlinio nuolatinės srovės variklio valdymas, detalių išdėstymas

Viršutinis sluoksnis



Apatinis sluoksnis



		Mastelis		Raidė		Masė	
TECHNOLOGIJOS						Parašas	Data
ŠU FAKULTETAS		Diplomantas		A. Paulauskas			
		Vadovas		D. Dervinis			
2013	Lapas	Lapų	Konsultantas				Bešepetėlinio nuolatinės srovės variklio valdymas
	3	4	Recenzentas	D. Balbonas			
			Kat. Vedėjas	G. Daunys			

Programinė įranga BLDC.c

```

#define F_CPU 8000000
#include <avr/io.h>
#include <avr/pgmspace.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#define BAUDRATE 19200
#define BAUD_PRESCALLER (((F_CPU / (BAUDRATE * 16UL))) - 1)
#include „motor.h“

int a;
int b;
int c;
int volt;
int curr;

int volt_min = VOLTAGE_MIN;
int volt_max = VOLTAGE_MAX;
int curr_max = CURRENT_MAX;
char buffer[5];

void adc_init(void){
    ADMUX = 0b01000000;           //Avcc(+5v) as voltage reference
    ADCSRB =0b00000000;         //ADC in free-running mode
    ADCSRA =0b11100110;         //Start converting
}

uint16_t adc_read(uint8_t ch){
    ch &= 0b00000111;           // AND operation with 7
    ADMUX = (ADMUX & 0xF8)|ch; // clears the bottom 3 bits before Oring
    ADCSRA |= (1<<ADSC);
    _delay_ms (1);
    while(!(ADCSRA & (1<<ADIF))); //Some time for conversation to be done
    ADCSRA|=(1<<ADIF);
}

```

```

return (ADC);
}

void USART_init(void){           //Set UART speed, enable TX and RX lines
    UBRRH = (uint8_t)(BAUD_PRESCALLER>>8);
    UBRRL = (uint8_t)(BAUD_PRESCALLER);
    UCSRB = (1<<RXEN0)|(1<<TXEN0);
    UCSRC = (1<<USBS0)|(3<<UCSZ00);
}

void USART_send( unsigned char data){//Send data to UART
    while(!(UCSR0A & (1<<UDRE0)));
        UDR0 = data;
}

void USART_putstring(char* StringPtr){//Send string to UART
    while(*StringPtr != 0x00){
        USART_send(*StringPtr);
        StringPtr++;
    }
}

void timer(void){               //Enable interrupts caused by TMO0 overflow
    SREG=0b10000000;
    TCCR0A=0b00000000;
    TCCR0B=0b00000101;
    TIMSK0=0b00000001;
}

void Start_motor(void){        //Generating start commutation
    //1 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b00100001;

```

```

PORTD=0b10000000;
    _delay_ms (250);
    //2 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b10000001;
    PORTD=0b10000000;
    _delay_ms (200);
    //3 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b10000001;
    PORTD=0b01000000;
    _delay_ms (150);
    //4 STEP
    TCCR2A=0b10000001;
    TCCR1A=0b00000001;
    PORTD=0b01000000;
    _delay_ms (100);
    // 5 STEP
    TCCR2A=0b10000001;
    TCCR1A=0b00000001;
    PORTD=0b00100000;
    _delay_ms (75);
    //6 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b00100001;
    PORTD=0b00100000;
    _delay_ms (50);
}

void Stop_motor(void){                                //Stop motor and turn off all timers and interrupts
    // 1 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b00100001;
    PORTD=0b10000000;

```

```

_delay_ms (300);
TCCR2A=0b00000001;
TCCR1A=0b00000001;
PORTD=0b00000000;

SREG = 0b00000000;
TCCR0A = 0b00000000;
TCCR0B = 0b00000000;
TIMSK0 = 0b00000000;
TCCR1A=0b00000000;
TCCR1B=0b00000000;
TCCR2A=0b00000000;
TCCR2B=0b00000000;
OCR1A = 0;
OCR1B = 0;
OCR2A = 0;
OCR2B = 0;
DDRD=0b00000000;
PORTD=0b00000000;
DDRB=0b00000000;
PORTB = 0b00000000;
}

ISR(TIMER0_OVF_vect){

a=STAGE_A; //Read ADC values
b=STAGE_B;
c=STAGE_C;
volt=VOLTAGE;
curr=CURRENT;

if (volt < volt_min){ //Check low voltage
USART_putstring(„Voltage too low“);
}
}

```

```

    USART_send(,r');
    USART_send(,n');
    _delay_ms (1000);
}

if (volt > volt_max){           //Check high voltage
    USART_putstring(,Voltage too high");
    USART_send(,r');
    USART_send(,n');
    _delay_ms (1000);
}                               //Check max current
if (curr > curr_max){
    Stop_motor();
}

//Find rotors position and set next commutation step
if (a == volt/2){
    if (c > volt/2){
        //4 STEP
        TCCR2A=0b10000001;
        TCCR1A=0b00000001;
        PORTD=0b01000000;
    }
    if (c < volt/2){
        // 1 STEP
        TCCR2A=0b00000001;
        TCCR1A=0b00100001;
        PORTD=0b10000000;
    }
}

if (b == volt/2){
    if (a > volt/2){
        //6 STEP
        TCCR2A=0b00000001;

```

```
TCCR1A=0b00100001;
PORTD=0b00100000;
}
if (a < volt/2){
    //3 STEP
    TCCR2A=0b00000001;
    TCCR1A=0b10000001;
    PORTD=0b01000000;
}
}

if (c == volt/2){
    if(b > volt/2){
        //2 STEP
        TCCR2A=0b00000001;
        TCCR1A=0b10000001;
        PORTD=0b10000000;
    }
    if(b < volt/2){
        // 5 STEP
        TCCR2A=0b10000001;
        TCCR1A=0b00000001;
        PORTD=0b00100000;
    }
}

}

int main()
{
    USART_init();
    adc_init();
    //Initialize PWM
    TCCR1A=0b00000001;
```

```
TCCR1B=0b00000011;
TCCR2A=0b00000001;
TCCR2B=0b00000100;
OCR1A = 255;
OCR1B = 255;
OCR2A = 255;
OCR2B = 255;
DDRD=0b11100000;
PORTD=0b00000000;
DDRB=0b00001110;
PORTB = 0b00000000;
Start_motor();
timer();
```

```
while(1){
    //Send data to UART
    USART_putstr(„Voltage „);
    itoa(volt, buffer, 10);
    USART_putstr(buffer);
    USART_putstr(„ „);

    _delay_ms(50);

    USART_send(„r“);
    USART_send(„n“);

    USART_putstr(„Current „);
    itoa(curr, buffer, 10);
    USART_putstr(buffer);
    USART_putstr(„ „);

    _delay_ms(50);
```

```
    USART_send(\\r\\);  
    USART_send(\\n\\);  
    _delay_ms (1000);  
  
    }  
}
```

```
#define STAGE_A          adc_read(0);
#define STAGE_B          adc_read(1);
#define STAGE_C          adc_read(2);
#define VOLTAGE          adc_read(3);
#define CURRENT          adc_read(6);

#define VOLTAGE_MIN      400;
#define VOLTAGE_MAX      900;
#define CURRENT_MAX      100;
```