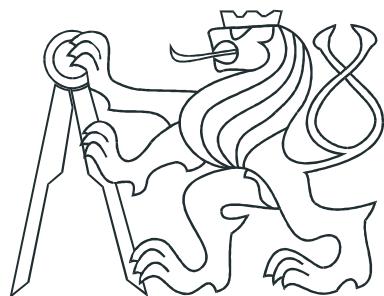


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA ELEKTROTECHNICKÁ



DIPLOMOVÁ PRÁCE

Quadrotor - dálkové řízení

Praha, 2011

Autor: Bc. Miroslav Dušek

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne

podpis

Poděkování

Zejména děkuji vedoucímu diplomové práce Doc. Ing. Jiřímu Bayerovi za cenné rady a optimismus po celou dobu práce. Velké díky patří rodině, která mě po celou dobu studia podporovala.

Abstrakt

Náplní práce je návrh a realizace regulátoru pro bezkartáčové motory experimentálního létajícího stroje Quadrotor, který vznikl v rámci diplomových prací na katedře řídící techniky.

Práce pojednává o problematice bezsenzorového řízení bezkartáčových motorů s lichoběžníkovým polem a sinusovým polem. Obsahuje kompletní hardwarovou a programovou realizaci regulátoru využívající bezsenzorové řízení.

Abstract

This master thesis aims at the design and the realisation of the Controllers for brushless motors of the quadrotor unmanned aerial vehicle, which was developed as part of master thesis on the department of Control Engineering.

The thesis deal with the problem of sensorless control of brushless motors with sinusoidal and trapezoidal field. The main part of this thesis is aimed at the hardware design and software implementation of the controller using sensorless control.

Obsah

Seznam obrázků	vii
Seznam tabulek	ix
1 Úvod	3
1.1 Zadání	3
1.2 Popis stroje Quadrotor	3
1.2.1 Střídač a pohon	4
1.2.2 Vrtule	5
1.2.3 Elektronika	5
1.2.4 Analýza stavu stroje	5
1.3 Cíl práce	6
2 Bezkartáčový motor	7
2.1 Princip Bezkartáčového motoru	7
2.2 Připojení motoru	8
2.3 Teorie řízení BLDC	9
2.3.1 Metoda Back-EMF	9
2.3.2 Rozběh BLDC motoru	12
2.3.3 Navrhované schéma řízení BLDC pomocí Back-EMF	15
2.4 Teorie řízení PMSM motoru	16
3 Realizace měniče pro BLDC motor	19
3.1 Hardware	19
3.1.1 Procesorová deska	20
3.1.1.1 Procesor	20
3.1.1.2 Budiče pro tranzistory	21
3.1.1.3 ADC měření	23

3.1.1.4	Komunikace CAN	24
3.1.1.5	Ostatní HW	25
3.1.2	Výkonová deska	27
3.1.2.1	Trojfázový H-můstek	27
3.1.2.2	Měřící subsystém	29
3.1.2.3	Návrh DPS	30
3.1.3	Zdrojová deska	31
3.2	Software	32
3.2.1	Vývojové nástroje	32
3.2.2	Stavba programu pod GNUARM	32
3.2.3	Konfigurace JTAG pod OPENOCD	35
3.2.4	Inicializace	37
3.2.5	Algoritmus pro BLDC motor	38
3.2.5.1	Start motoru	38
3.2.5.2	Regulace pozice motoru	38
3.2.6	Řídící algoritmus pro PMSM motor	40
3.2.6.1	Nároky na periferie procesoru	40
3.2.6.2	Zavedení desetinného čísla	40
3.2.6.3	Definování funkce sinus	42
3.2.6.4	Vývojový diagram algoritmu	42
3.2.7	CAN komunikace	43
3.2.7.1	Zapojení CAN sběrnice	44
3.2.7.2	nastavení sběrnice CAN	44
3.2.7.3	Zprávy na sběrnici CAN	44
4	Testy realizovaného regulátoru	45
5	Závěr	51
Literatura		54
A	Motivy desek plošných spojů	I
B	Obsah přiloženého CD	V

Seznam obrázků

1.1	Model stroje	4
1.2	Použitý motor MEGA 600/30/7	4
1.3	Střídač TMM4416-3S Expert+ Pro	4
2.1	Základní schéma motoru	7
2.2	Uspořádání motoru s rotačním pláštěm	8
2.3	Struktura trojfázového H-můstku	8
2.4	Princip 6 krokové komutace	9
2.5	Metoda Back-EMF při PWM on	10
2.6	Metoda Back-EMF při PWM off	11
2.7	Schéma pro nabítí bootstrapových kondenzátorů	12
2.8	Směr proudu při stabilizaci rotoru ve známé pozici	13
2.9	Ukázky rampy pro stabilizaci rotoru motoru ve známé pozici	13
2.10	Možnosti prvního kroku komutace dle požadovaného směru otáčení	14
2.11	Ukázky rampy pro stabilizaci rotoru motoru ve známé pozici	14
2.12	Ukázky rampy pro rozběh motoru s měřením Back-EMF	15
2.13	Schéma řízení BLDC pomocí Back-EMF	15
2.14	Grafická reprezentace stavů H-můstku	16
2.15	Generování SVPWM	18
2.16	Volt / Hertz charakteristika	18
3.1	Kompletní měnič motoru	19
3.2	Zapojení procesoru	21
3.3	Modul zajišťující napájení 1.8V a 3.3V	21
3.4	Princip bootstrap budiče pro horní tranzistor	22
3.5	Zapojení bootstrapového budiče	23
3.6	Zapojení neinvertujícího zesilovače	23
3.7	AD823 - vnitřní propojení	24

3.8	Měřící substituční systém	24
3.9	Schéma zapojení obvodu MCP2551	25
3.10	Schéma CAN sběrnice	25
3.11	Zapojení LED signalizace	26
3.12	Konektor JTAG	26
3.13	Parametry tranzistoru SI7336ADP I	28
3.14	Parametry tranzistoru SI7336ADP II	28
3.15	Schéma polovičního H-můstku	29
3.16	Odporový dělič s filtračním kondenzátorem pro fázi A	29
3.17	Proudová zatížitelnost spoje dle normy IPC-D-275	30
3.18	Schéma spínacího zdroje	31
3.19	Vývojové nástroje	32
3.20	Vývojový diagram startu BLDC motoru	38
3.21	Back-EMF Regulátor	38
3.22	Vložení mrtvého pásma do kritických míst	40
3.23	Vývojový diagram algoritmu SVPWM	42
3.24	Vývoj vzestupných a sestupných hran PWM pro horní spínače H-můstku	43
3.25	Zapojení CAN sběrnice na Quadrotoru	44
4.1	Měření na současném měniči	45
4.2	Sestava zkušebního měniče	46
4.3	Optimální rampa pro rozběh motoru RC 600/30/7	47
4.4	Testy měniče I.	47
4.5	Testy měniče II.	48
4.6	Testy měniče III.	48
A.1	Procesorová deska - Top vrstva	I
A.2	Procesorová deska - BOTTOM vrstva	II
A.3	Výkonová deska - TOP vrstva	II
A.4	Výkonová deska - BOTTOM vrstva	III
A.5	Zdrojová deska - TOP vrstva	III
A.6	Zdrojová deska - BOTTOM vrstva	IV

Seznam tabulek

2.1	Stavy přepínačů v trojfázovém H-můstku	16
3.1	Parametry tranzistoru SI7336ADP	27
3.2	Parametry obvodu MIC4680BM	31
3.3	Rozhodovací úrovně signálu Back-EMF I.	39
3.4	Rozhodovací úrovně signálu Back-EMF II.	39
4.1	Parametry zkušebního motoru BL3056	46

Seznam použitých zkratek

PWM	Pulse width modulation
BLDC	BrushLess Direct Current
PMSM	Permanent magnet synchronous electric motors
DPS	deska plošných spojů
Back - EMF	Back electromotoric force
BEMF	Back electromotoric force
SVPWM	space vector pulse width modulation

Kapitola 1

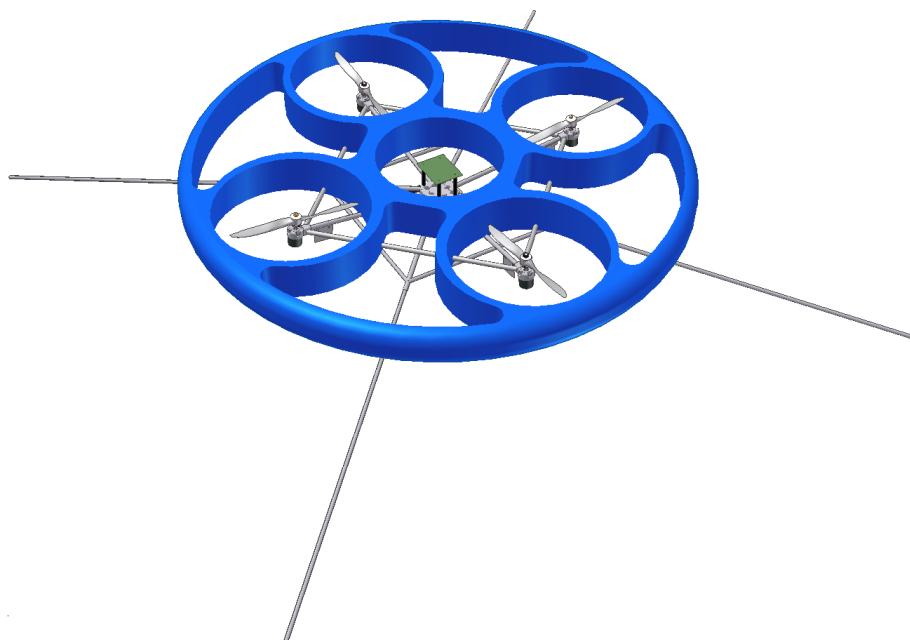
Úvod

1.1 Zadání

Tato práce navazuje na diplomové práce bývalých studentů Ing. J. Marečka a Ing. P. Heinricha, kteří v rámci svých prací vytvořili experimentální létající stroj Quadrotor. Cílem této práce je navrhnout a realizovat úpravu stávajících regulátorů výkonových střídačů pro synchronní motory pohonu vrtulí stroje. Dále reidentifikovat statické a dynamické parametry stroje ve stavech startu, vznášení a při volném letu s upravenými měniči. Na základě této identifikace upravit matematický model stroje a jeho řídící algoritmus.

1.2 Popis stroje Quadrotor

Quadrotor, jak už sám název vypovídá, je elektronický stroj se čtyřmi nezávislými pohonnými jednotkami. Rám stroje je tvořen svařencem z duralových trubek uspořádaných do kříže. V jeho středu je umístěna platforma pro elektroniku a na každém jeho konci je umístěn držák pro motor a pro jeho střídač. Celková velikost stroje je okolo 1,2m a hmotnost stroje 3,5kg. Model stroje lze vidět na následujícím obrázku.



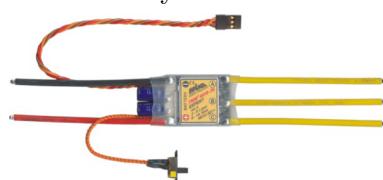
Obrázek 1.1: Model stroje

1.2.1 Střídač a pohon

Stroj je osazen synchronními trojfázovými motory s přímým pohonem vrtule. Řízení těchto motorů obstarává komerční střídač TMM4416-3S Expert+ Pro od firmy MGM-Compro, který se ovládá obdélníkovým signálem s pulzně-šířkovou modulací (PWM). Tento střídač nedává žádné informace o skutečných otáčkách stroje a není schopen žádné regulace otáček. Veškerá regulace je tedy závislá na nadřazené elektronice.



Obrázek 1.2: Použitý motor MEGA 600/30/7



Obrázek 1.3: Střídač TMM4416-3S Expert+ Pro

1.2.2 Vrtule

Quadrotor je v současném stavu vybaven vrtulemi APC 14x7. Tato vrtule je ovšem navržena pro letadla a její využití přináší značné snížení účinnosti pohonu. Použité vrtule mají příliš veliké stoupání a malou plochu listu a tím i značně přispívají do nestability stroje.

1.2.3 Elektronika

Elektronika, kterou je v současné době quadrotor vybaven se skládá

- Inerciální jednotka složená z gyroskopu (jednoosý senzor úhlové rychlosti), z akcelerometrů měřící translační zrychlení a modulu převodníku, který zajišťuje převod analogových měronosných signálů na digitální reprezentaci a komunikaci s řídícím modulem přes SPI sběrnici.
- Hlavní řídící deska - Jádrem quadrotoru je 32 bitový mikroprocesor AT91SAM7X-256, na kterém běží systém FreeRTOS. Deska zpracovává všechna data získaná ze senzorického subsystému, které zpracovává a v závislosti na jejich vyhodnocení provádí regulaci celého stroje.
- WiFi modul - je spojen přes ethernetové rozhraní s hlavní řídící deskou a umožňuje bezdrátové ovládání celého stroje.
- Zdroj napájení - obstarává napětí pro elektroniku (vstupní napětí z baterie)

1.2.4 Analýza stavu stroje

Pro testy byl vyvinut speciální závěs, který umožní quadrotor zavěsit na lano a provést testy bez možnosti zničit stroj. Testy na quadratoru byly provedeny dvojím způsobem

- Regulace s původním regulátorem pro ověření výsledků s předchozích diplomových prací
- Bez regulace. Přímé ovladání motorů z pozemní stanice

Z testů bylo zjištěno, že jakmile stroj jeví známky vzletu, stává se nestabilním a to jak při přímém ovládání motorů tak i regulaci s původním regulátorem.

1.3 Cíl práce

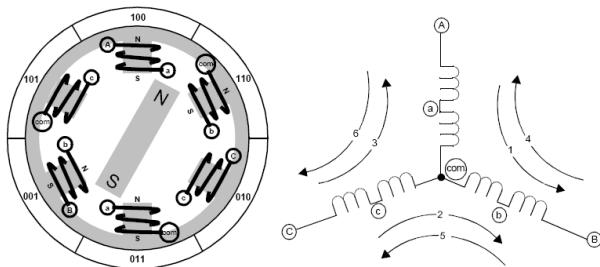
Pro kvalitnější regulaci je tedy velmi důležité mít zpětnou vazbu z měniče o skutečné rychlosti motoru. Dále nahradit analogový komunikační kanál sběrnicí CAN, kde lze zaručit uspěšnou výměnu dat. Jelikož firma MGM-Compro si kladla přehnané podmínky ke spolupráci na vývoji měniče, který by vzešel z jejich stávajícího řešení, zaměřil jsem tuto práci na kompletní vývoj nového měniče, který by splňoval výše zmíněné požadavky.

Kapitola 2

Bezkartáčový motor

2.1 Princip Bezkartáčového motoru

Elektrická energie je přeměňována na mechanickou energii přitažlivými silami mezi permanentními magnety rotoru a rotujícím magnetickým polem indukovaným na vinutí statoru, který je zapojen do hvězdy. Základní schéma motoru lze vidět na následujícím obrázku.

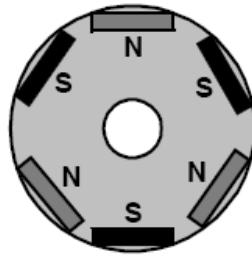


Obrázek 2.1: Základní schéma motoru

V základu rozlišují dva druhy bezkartáčových motorů

- - motor s lichoběžníkovým polem (BLDC)
- - motor se sinusovým polem (PMSM)

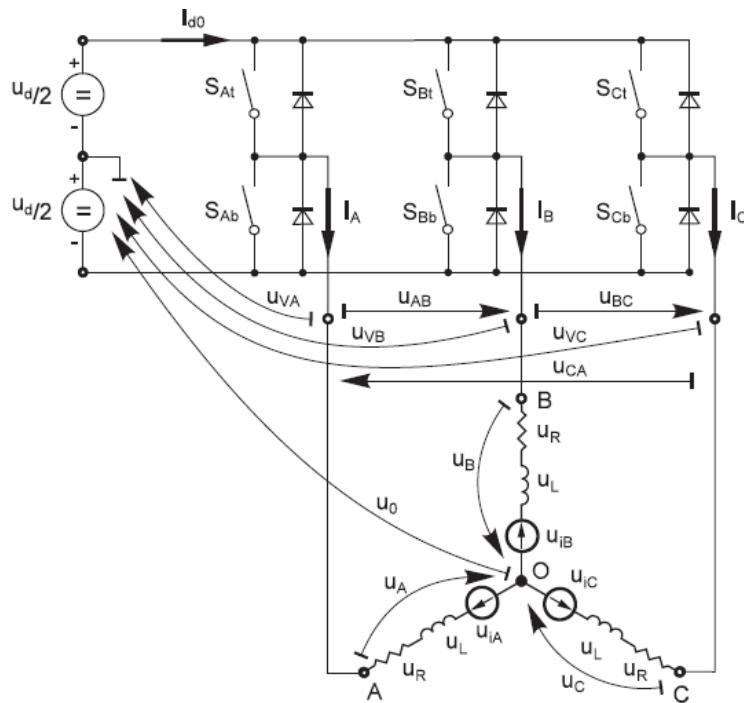
V našem případě je použit motor (RC 600/30/7) firmy Megamotor, který má rotační plášt. Mechanická konstrukce motoru se pak změní tak, že uvnitř motoru je statorové vinutí a rotor je tvořen permanentními magnety umístěny na rotačním plášti motoru způsobem uvedeným na tomto obrázku.



Obrázek 2.2: Uspořádání motoru s rotačním pláštěm

2.2 Připojení motoru

K řízení bezkartáčových motorů se využívá systému tvořeného 3 polovičními H-můstky, kde každý řídí průběh budícího signálu pro jedno vinutí(jednu fázi). Pro řízení je tedy nutné mít 6 kanálů, které přes budiče hradel řídí 6 spínacích tranzistorů (IGBT, MOSFET). Přibližné schéma může vypadat takto:



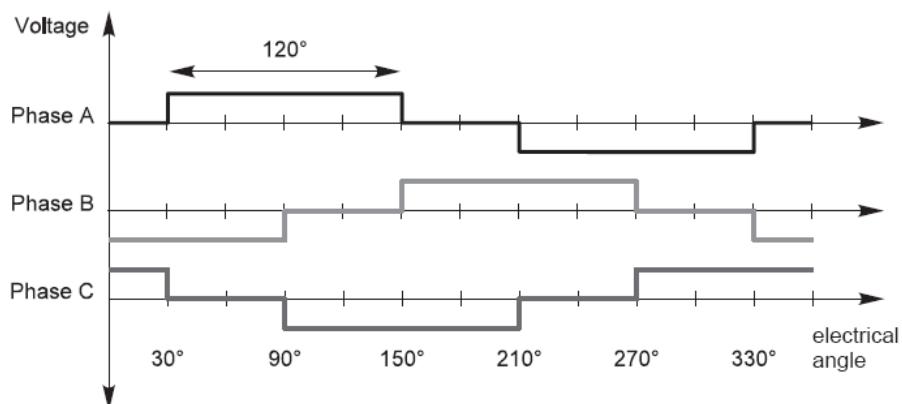
Obrázek 2.3: Struktura trojfázového H-můstku

- u_{VA}, u_{VB}, u_{VC} - "branch" napětí, napětí jednotlivých fází proti zemi

- u_A, u_B, u_C - fázová napětí na vinutí
- u_{iA}, u_{iB}, u_{iC} - fázová napětí indukovaná na statorovym vinutí(Back-EMF)

2.3 Teorie řízení BLDC

K řízení spínacích tranzistorů se používá pulsně-šířková modulace(PWM), což je metoda řízení úrovně stejnosměrného napětí. Algoritmus spínání, dle které se rotor motoru otáčí se označuje jako 6 kroková komutace. Princip této metody je znázorněn na následujícím obrázku.



Obrázek 2.4: Princip 6 krokové komutace

V každém kroku komutace jsou aktivní jen dvě fáze. Sled fázové komutace je AB - AC - BC - BA - CA - CB. Pro maximální točivý moment motoru je třeba, aby komutace byla provedena vždy po 60° el. úhlu. Čas komutace je určen pozicí rotoru, který je detekován buď za pomocí senzorů nebo sledování back-EMF napětí na plovoucí cívce motoru. V našem případě je umístění senzoru, tak aby se dala přesně odhadnout poloha rotoru, nemožné, proto musíme použít bezsenzorové snímání v podobě metody Back-EMF.

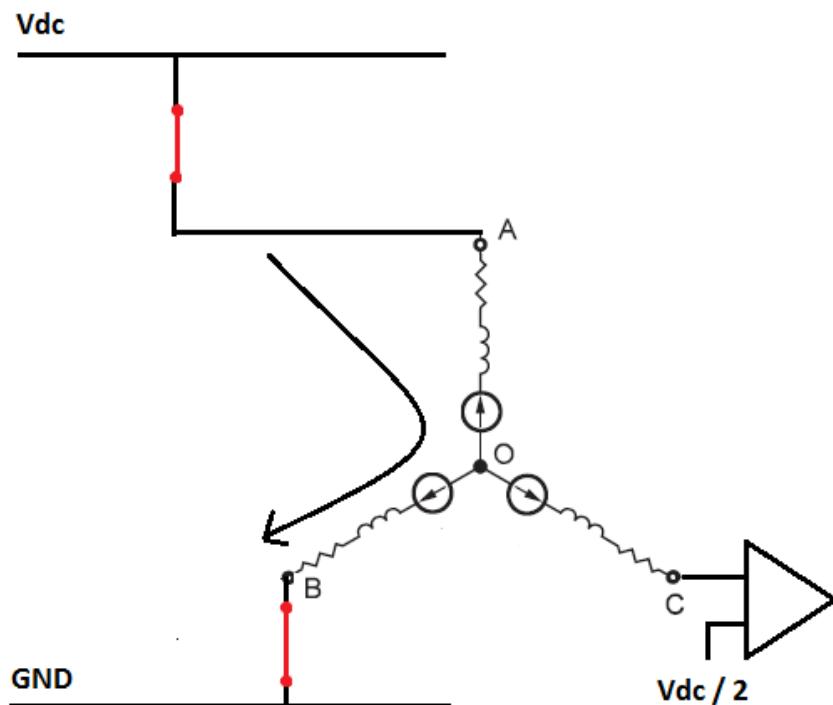
2.3.1 Metoda Back-EMF

Back-EMF je napětí indukované v zrovna nenapájených vinutích, které se v tu chvíli chovají jako generátor, to znamená, točivé magnetické pole v nich indukuje napětí úměrné

jeho velikosti, které je tak úměrné rychlosti otáčení motoru. Back-EMF lze měřit dvěma způsoby.

- dolní tranzistor jedné fáze a horní tranzistor druhé fáze jsou sepnuty
- dolní tranzistor jedné fáze sepnut, horní tranzistor druhé fáze rozepnut

V prvním případě si představme situaci, že fáze A a fáze B jsou sepnuty. Potom situace na motoru vypadá takto :



Obrázek 2.5: Metoda Back-EMF při PWM on

- Z fáze A, pak máme

$$v_n = v_{dc} - v_{mos} - ri - L \frac{di}{dt} - e_a$$

- Z fáze B, pak máme

$$v_n = v_{mos} + ri + L \frac{di}{dt} - e_b$$

- Pak platí vztah pro v_n definován jako

$$v_n = \frac{v_{dc}}{2} - \frac{e_a + e_b}{2}$$

- Vzniklý vztah použitím rovnice ($e_a + e_b + e_c = 0$) lze upravit na

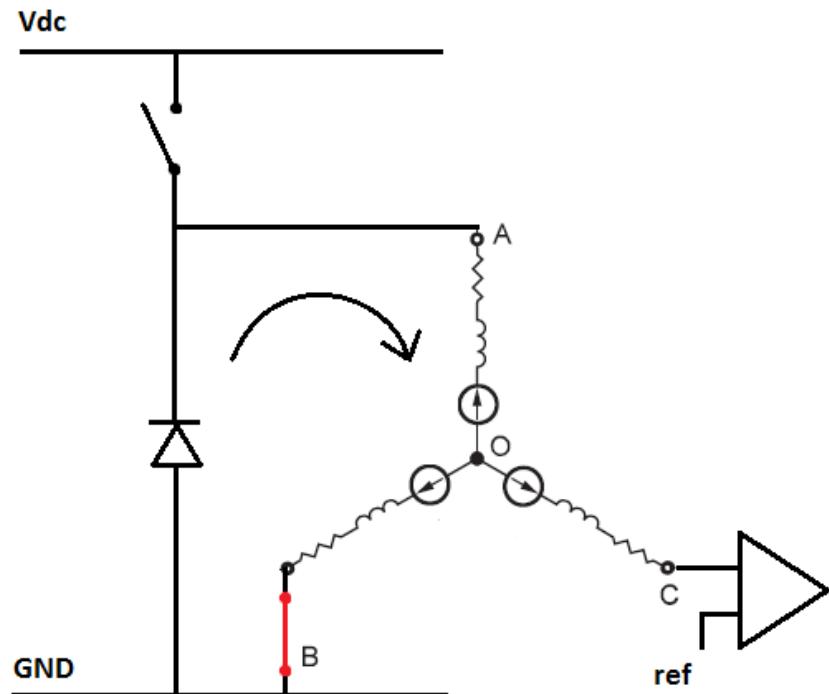
$$v_n = \frac{v_{dc}}{2} + \frac{e_c}{2}$$

- Potom vztah pro fázové napětí plovoucí fáze C je definován jako

$$v_c = e_c + v_n = \frac{v_{dc}}{2} + \frac{3}{2} e_c$$

Výsledný vztah pak platí pro všechny fáze motoru, které jsou v daném kroku komutace plovoucí. Na tomto vztahu pracuje metoda Back-EMF. Pomocí AD převodníku kontroluje průchod nulou rozdílu poloviny svorkového napětí a napětí naměřeného na plovoucí fázi.

V druhém případě dolní tranzistor fáze B je sepnut a horní tranzistor fáze A je řízen PWM o definované střídě. Potom situace na motoru v době rozepnutého tranzistoru na fázi A vypadá takto :



Obrázek 2.6: Metoda Back-EMF při PWM off

- Z fáze A, pak máme

$$v_n = 0 - v_d - ri - L \frac{di}{dt} - e_a$$

- Z fáze B, pak máme

$$v_n = v_{mos} + ri + L \frac{di}{dt} - e_b$$

- Pak lze najít vztah pro v_n definován jako

$$v_n = \frac{v_{mos} - v_d}{2} - \frac{e_a + e_b}{2}$$

- Vzniklý vztah použitím rovnice ($e_a + e_b + e_c = 0$) lze upravit na

$$v_n = \frac{v_{mos} - v_d}{2} + \frac{e_c}{2}$$

- Potom vztah pro fázové napětí plovoucí fáze C je definován jako

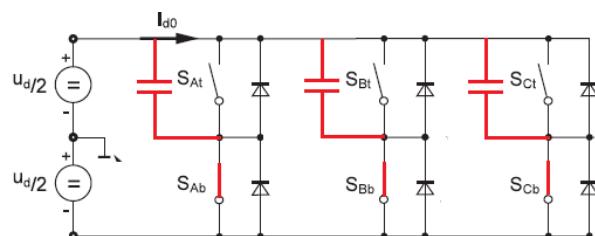
$$v_c = e_c + v_n = \frac{v_{mos} - v_d}{2} + \frac{3}{2} e_c$$

Pro nízké rychlosti bude použita druhá metoda čili dolní tranzistor sepnut v jedné fázi a horní tranzistor v druhé fázi v době měření rozepnut. Při vysokých rychlostech, kdy střída PWM se bude blížit k 1 a nebude dostatečný prostor pro A/D měření, lze využít první metodu.

2.3.2 Rozběh BLDC motoru

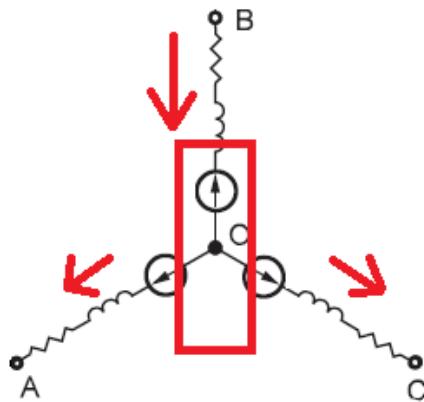
Start motoru v bezsenzorovém řízení lze rozdělit do několika kroků.

- Nabití bootstrapových kondenzátorů - tento krok je dán výběrem budičů H-můstku a je nutný pro bezproblémové spínání horních tranzistorů. Dolní tranzistory ve všech větví H-můstku jsou sepnuty na dobu potřebnou pro nabítí bootstrapových kondenzátorů. Tato doba je určena velikostí kapacity použitých kondenzátorů a byla stanovena na 3ms.



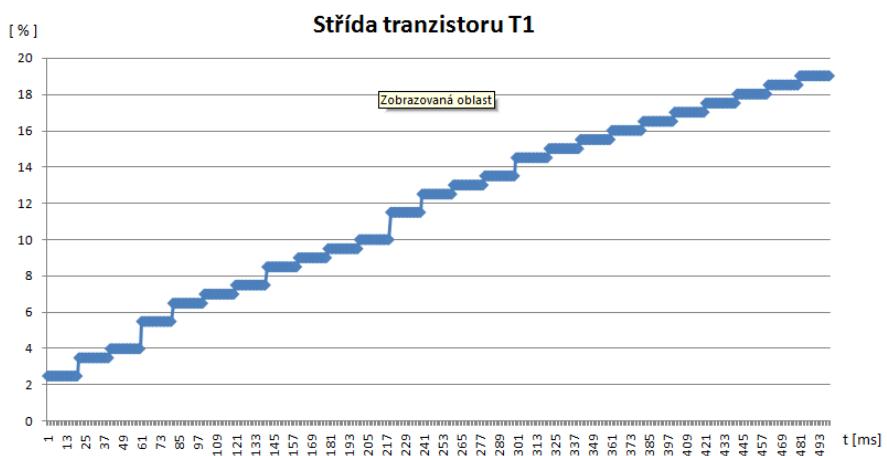
Obrázek 2.7: Schéma pro nabítí bootstrapových kondenzátorů

- Určení pozice rotoru - Jelikož neznáme pozici rotoru, je nejdříve zapotřebí dostat rotor do požadované pozice. Toho se docílí postupným zvyšováním proudu protékající statorovým vinutím ve směru, který je zobrazen na následujícím obrázku.



Obrázek 2.8: Směr proudu při stabilizaci rotoru ve známé pozici

Dolní tranzistory S_{Bb} , S_{Cb} ve fázích B a C se sepnou střídou budícího signálu 100%. Horní tranzistor S_{At} se spíná se vzrůstající střídou, která se řídí následujícím grafem. Jak lze vidět Graf je složen z 25 kroků, kde každý krok trvá 20ms. Po 500ms (20*25ms) rotor kmitá v požadované pozici.

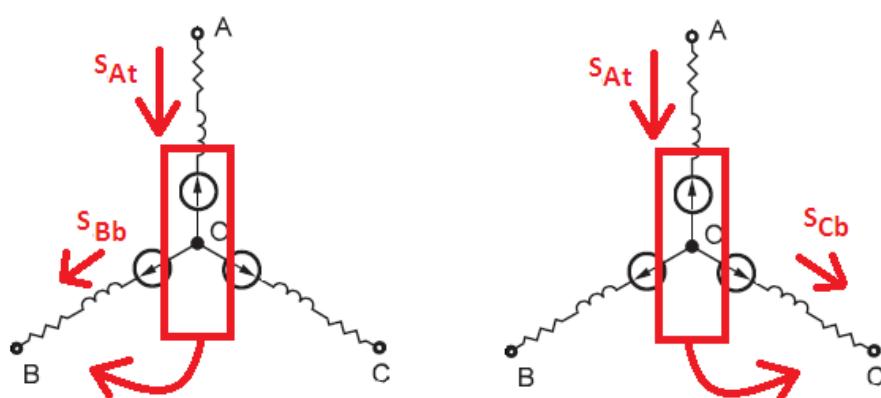


Obrázek 2.9: Ukázky rampy pro stabilizaci rotoru motoru ve známé pozici

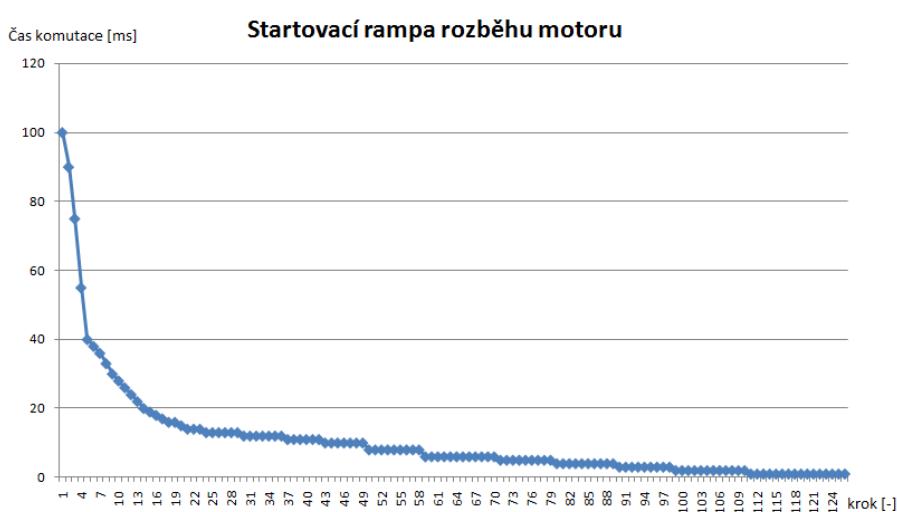
- Rozběh motoru v otevřené smyčce - Jakmile je rotor v ustálené pozici lze motor rozeběhnout v požadovaném směru. Rozběh motoru se řídí dle rampy po dobu 3 mechanických otáček. Mechanická otáčka se rovná p elektrickým otáčkám, kde p je

počet pólových dvojic. Elektrická otáčka se rovná 6 krokům komutace. Počet kroků komutace, který je řízen dle této rampy je $(6 \cdot 7 \cdot 3) = 126$. V koncových bodech této rampy se motor otáčí rychlostí úměrné napětí na motoru, které je dáno střídou stanovenou při stabilizaci rotoru. Vztah mezi rychlostí komutace a napětím na motoru je dáno konstantou známou od výrobce motoru. V našem případě je tato konstanta rovna

$$K_m = 640 \frac{RPM}{V}$$

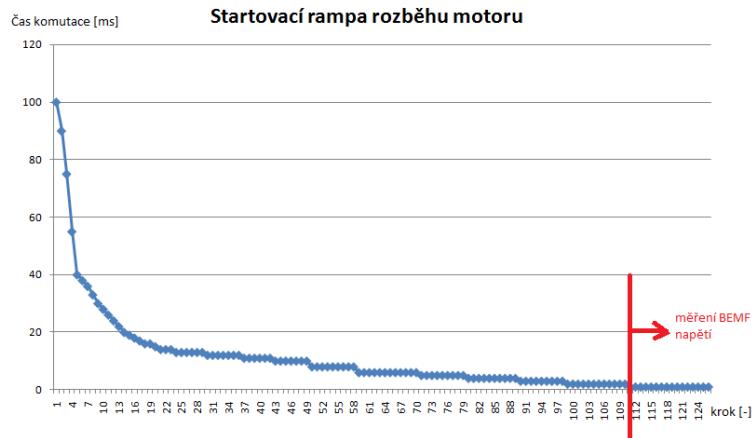


Obrázek 2.10: Možnosti prvního kroku komutace dle požadovaného směru otáčení



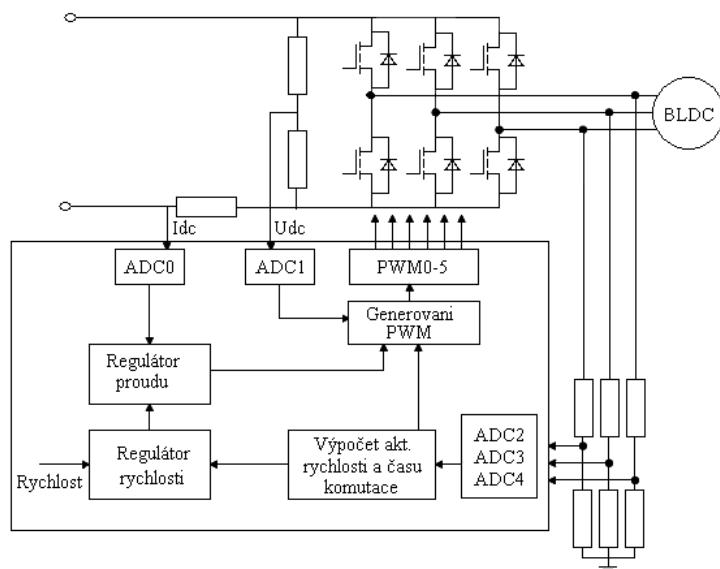
Obrázek 2.11: Ukázky rampy pro stabilizaci rotoru motoru ve známé pozici

- Přechod do bezsenzorového řízení - Tento bod provádíme paralelně s koncem předchozího kroku. Během čekání na další komutaci měříme AD převodníkem napětí na právě nebuzené fázi a zjišťujeme zda napětí Back-EMF je již přítomné. Pokud toto Back-EMF napětí AD převodník naměří na nejméně třech po sobě jdoucích krocích, po doběhnutí startovací rampy se doba komutace již řídí nově vzniklou zpětnou vazbou.



Obrázek 2.12: Ukázky rampy pro rozběh motoru s měřením Back-EMF

2.3.3 Navrhované schéma řízení BLDC pomocí Back-EMF



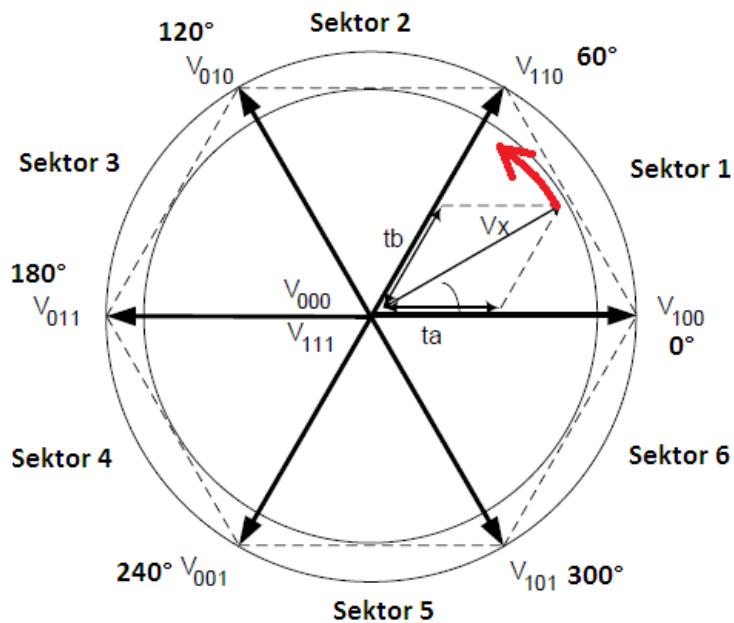
Obrázek 2.13: Schéma řízení BLDC pomocí Back-EMF

2.4 Teorie řízení PMSM motoru

PMSM motor lze řídit sinusovou pulzně šířkovou modulací (PWM) jako je SVPWM (Space vector pulse with modulation). H-můstek má 6 stavů, kde napětí je přivedeno na motor a 2 stavy, kdy jsou seplé jen horní tranzistory nebo dolní tranzistory. Všechny tyto stavy jsou popsány v následující tabulce. Pro odvození vzorců, které stanový časy spínání PWM, uvedeme i grafickou reprezentaci.

stav	kombinace	Napětí Van	Napětí Vbn	Napětí Vcn	prostorový vektor
0	$S_{Ab} S_{Bb} S_{Cb}$	0	0	0	$V_0(000)$
1	$S_{At} S_{Bb} S_{Cb}$	$\frac{2}{3} V_{dc}$	$-\frac{1}{3} V_{dc}$	$-\frac{1}{3} V_{dc}$	$V_1(100)$
2	$S_{At} S_{Bt} S_{Cb}$	$\frac{1}{3} V_{dc}$	$\frac{1}{3} V_{dc}$	$-\frac{2}{3} V_{dc}$	$V_2(110)$
3	$S_{Ab} S_{Bt} S_{Cb}$	$-\frac{1}{3} V_{dc}$	$\frac{2}{3} V_{dc}$	$-\frac{1}{3} V_{dc}$	$V_3(010)$
4	$S_{Ab} S_{Bt} S_{Ct}$	$-\frac{2}{3} V_{dc}$	$\frac{1}{3} V_{dc}$	$\frac{1}{3} V_{dc}$	$V_4(011)$
5	$S_{Ab} S_{Bb} S_{Ct}$	$-\frac{1}{3} V_{dc}$	$-\frac{1}{3} V_{dc}$	$\frac{2}{3} V_{dc}$	$V_5(001)$
6	$S_{At} S_{Bb} S_{Ct}$	$\frac{1}{3} V_{dc}$	$-\frac{2}{3} V_{dc}$	$\frac{1}{3} V_{dc}$	$V_6(101)$
7	$S_{At} S_{Bt} S_{Ct}$	0	0	0	$V_7(111)$

Tabulka 2.1: Stavy přepínačů v trojfázovém H-můstku



Obrázek 2.14: Grafická reprezentace stavů H-můstku

Uvažujme sektor 1, omezený vektory $V_1(100), V_2(110)$ a nulovými vektory $V_0(000)$ a $V_7(111)$. Vektor V_x uvnitř tohoto sektoru se rovná

$$V_x * \sin\left(\frac{\pi}{3} - \alpha\right) = V_a * \sin(\alpha)$$

a

$$V_x \sin(\alpha) = V_b * \sin\left(\frac{\pi}{3}\right)$$

Proto

$$V_a = \frac{2}{\sqrt{3}} * V_x * \sin\left(\frac{\pi}{3} - \alpha\right)$$

a

$$V_b = \frac{2}{\sqrt{3}} * V_x * \sin(\alpha)$$

V_a, V_b jsou složky V_x ve směru vektorů $V_1(100), V_2(110)$ a lze je approximovat procentuální hodnotou z $V_1(100), V_2(110)$ časy t_a, t_b přes periodu PWM T, čili platí:

$$V_x = V_a + V_b = V_1(100) * \frac{t_a}{T} + V_2(110) * \frac{t_b}{T} + \frac{t_0}{T} * (V_{000} nebo V_{111})$$

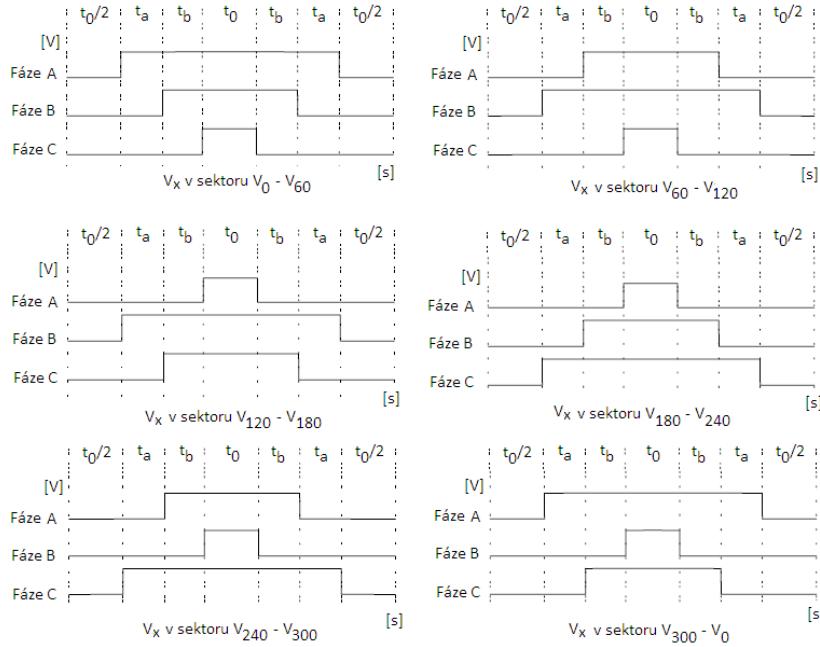
Z této rovnice již získáme potřebné časy t_a, t_b, t_0

$$t_a = \frac{V_x}{V_1(100)} * \sin\left(\frac{\pi}{3} - \alpha\right) * T \quad 0 \leq \alpha \leq \frac{\pi}{3}$$

$$t_b = \frac{V_x}{V_2(110)} * \sin(\alpha) * T \quad 0 \leq \alpha \leq \frac{\pi}{3}$$

$$t_0 = T - t_a - t_b$$

Průběhy napětí aplikováním těchto časů vypadají v jednotlivých sektorech takto:

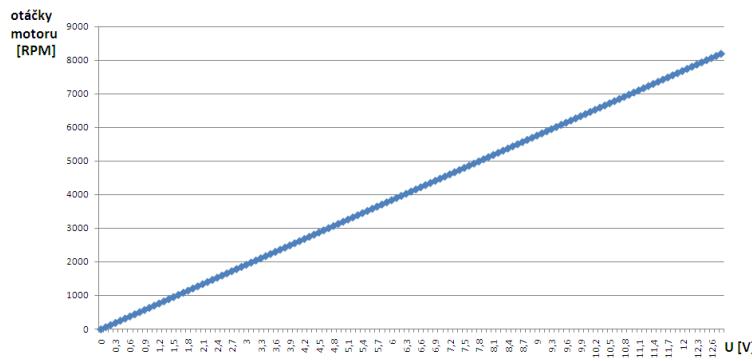


Obrázek 2.15: Generování SVPWM

Pro řízení otáček využijeme metodu Volt / Hertz. Katalogová hodnota motoru, která udává vztah mezi napětím motoru a frekvencí je

$$K_m = 640 \frac{\text{RPM}}{V} .$$

Pokud konstantu K_m aplikujeme na napětový rozsah 0 az maximální napájecí napětí měniče, získáme křivku, po které lze regulovat řízení otáček motoru.



Obrázek 2.16: Volt / Hertz charakteristika

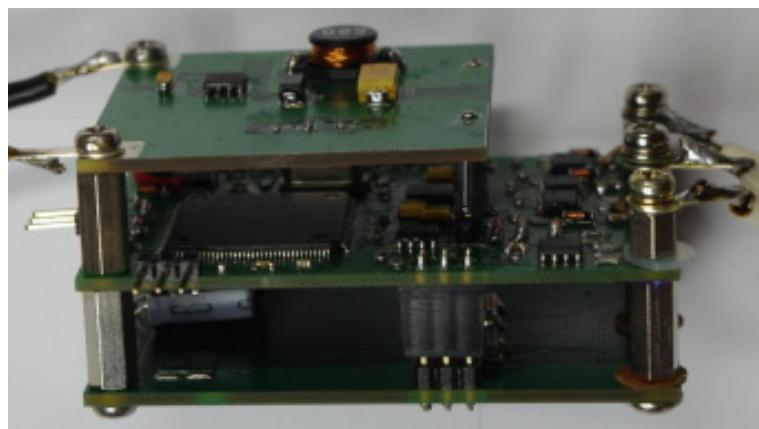
Parametrem regulace se testy nalezne optimální hodnota akcelerace čili změna napětí ΔU v časovém intervalu Δt .

Kapitola 3

Realizace měniče pro BLDC motor

3.1 Hardware

Hardware měniče se skládá ze tří plošných spojů. Spojení těchto desek je realizováno distančními sloupky a piny (samec / samice) tak, aby bylo rozebíratelné. Rozměr destiček byl limitován tvarem držáku na quadrotoru, a i když je větší než má starý komerční měnič, měnič lze bez problémů za pomoci stahovacích pásků na držák quadrotoru upevnit.



Obrázek 3.1: Kompletní měnič motoru

- Procesorová deska - obsahuje veškerou řídící elektroniku, měření a obvody pro komunikaci (CAN)
- Výkonová deska - obsahuje výkonový H-můstek, měřící výkonové odpory pro měření proudu
- Zdrojová deska - zajišťuje napájení procesorové desky.

3.1.1 Procesorová deska

Procesorová deska má za úkol nezávislou regulaci H-můstku, který ovládá samotný BLDC motor a plnění příkazů, které přijme od nadřazené elektroniky přes komunikační protokol CAN.

3.1.1.1 Procesor

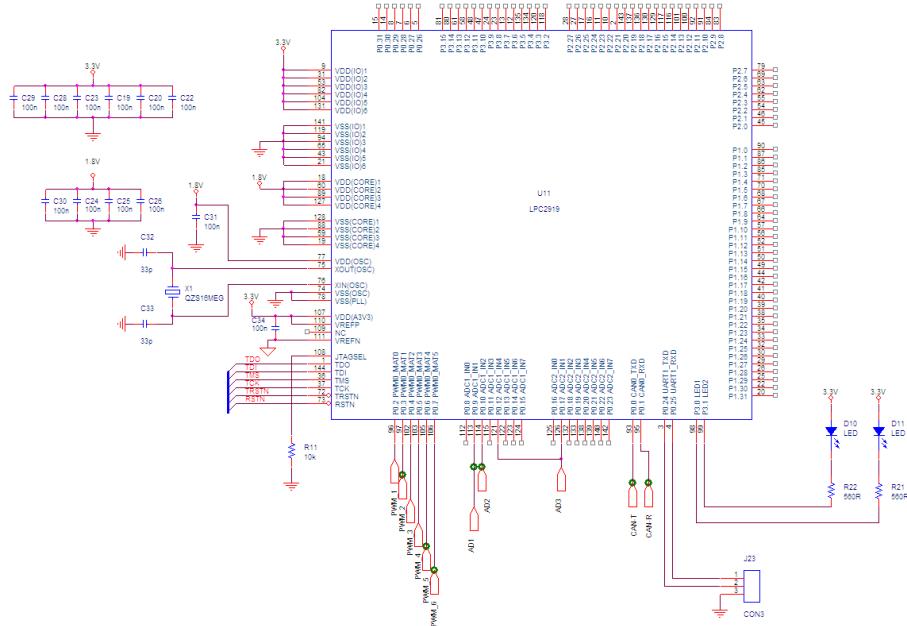
Jádrem procesorové desky je procesor firmy NXP LPC2919//01. Tento procesor je postaven na architektuře ARM968E-S. Pro tuto aplikaci byl tento procesor vybrán kvůli jeho periferiím.

- maximální frekvence procesoru je 125MHz
- 2x 10bitové osmi kanálové AD převodníky s měřícím rozsahem 3,3V a časem převodu pod 2,44us
- 4x šesti kanálová PWM
- synchronizace PWM a ADC
- 2x CAN 2.0B

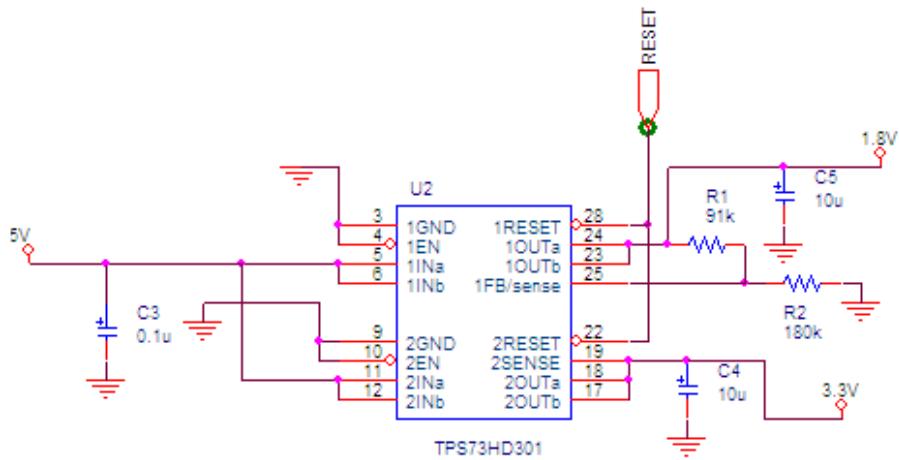
Částečná nevýhoda tohoto procesoru je velikost patice LQFP144, což při rozměrech desky dělá problémy při návrhu plošného spoje. Napájení pro procesor zajišťuje dvojitý LDO regulátor napětí TPS73HD301, který je napájen 5V ze zdrojové desky. LDO regulátor pro tyto účely plně postačuje, jelikož úbytek napětí je poměrně malý a při průměrném odběru procesoru 100mA je ztrátový výkon pro tento obvod přijatelný.

- 1.8V pro jádro
- 3.3V pro I/O porty

Procesor pro svoji funkčnost potřebuje krystal, který byl vybrán o hodnotě 16MHz. Blokovací kondenzátory zapojené co nejbliže k napájecím pinům procesoru. Přivedené napájecí napětí 1,8V a 3,3V a v neposlední řadě vyvedené programátorské a ladící rozhraní JTAG. Na následujících obrázkách lze vidět mnou navržené zapojení, které je použito na desce.



Obrázek 3.2: Zapojení procesoru

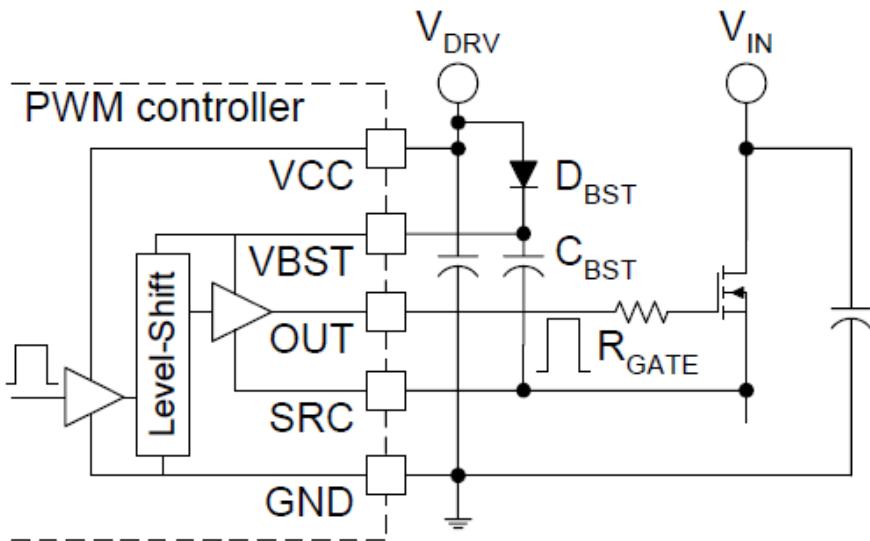


Obrázek 3.3: Modul zajišťující napájení 1.8V a 3.3V

3.1.1.2 Budiče pro tranzistory

Pro sepnutí N-MOSFET tranzistoru je zapotřebí výrazně vyšší napětí než je V_{th} ($V_{GS} > V_{th}$). N-MOSFET tranzistor potřebuje mít napětí vyšší než V_{dc} , aby došlo k nasycení tranzistoru a zabránilo se tak k výrazným tepelným ztrátám. Spínání polovičního H-můstku, kde horní i dolní tranzistor jsou typu N, je možné použitím speciálního bootstrap budiče, který umožňuje sepnutí horního tranzistoru v době, kdy je dolní tranzistor rozep-

nut.



Obrázek 3.4: Princip bootstrap budiče pro horní tranzistor

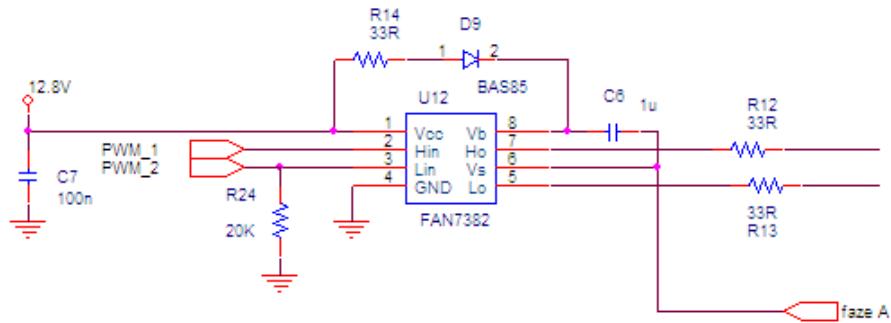
Volba bootstrapového kondenzátoru vychází z dynamických vlastností buzeného tranzistoru a obecně platí, že

$$C_{EXT} = \frac{Q_{GATE}}{V_{GATE}}$$

$$C_{BOOST} >>> C_{EXT}$$

Uvážením předchozí podmínky byla hodnota bootstrapového kondenzátoru stanovena na 1uF. Na volbu diody, přes kterou je nabíjen bootstrapový kondenzátor, je kladen důraz na její rychlosť, proto byla zvolena schottkyho dioda.

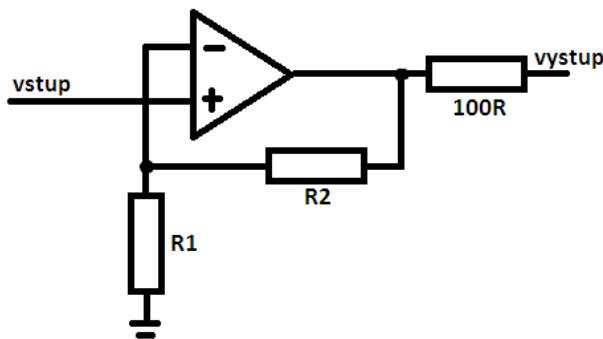
Bootstrapový obvod byl zvolen FAN7382 od firmy fairchild. Jelikož vyžaduje pro správnou funkčnost napětí přes 10V (při testech se ukázalo, že spolehlivě funguje při 10,2V), je tento obvod napájen přímo s baterie quadrotoru. Je tedy důležité hlídat napětí na baterii. Pokud je obvod napájen pod 10V, jeho nesprávná funkčnost se projeví zejména na větvi pro horní tranzistor. Špatné spínání tranzistorů v jedné větvi může vést ke zkratovým proudům a poškodit tak měnič.



Obrázek 3.5: Zapojení bootstrapového budiče

3.1.1.3 ADC měření

Měřený signál je nejdříve přiveden do rychlého operačního zesilovače, který je zapojen v neinvertivním zapojení dle následujícího obrázku.

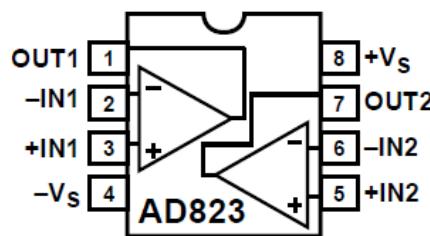


Obrázek 3.6: Zapojení neinvertujícího zesilovače

Volba hodnot R1 a R2 zavisí na tom, pokud toto měření využíváme jako měření proudu (vyšší zesílení) pro PMSM motor, či jako měření BEMF napětí pro BLDC motor (menší zesílení // zapojení jako sledovač napětí). V každém případě se výstupní napětí operačního zasilovače se rovná

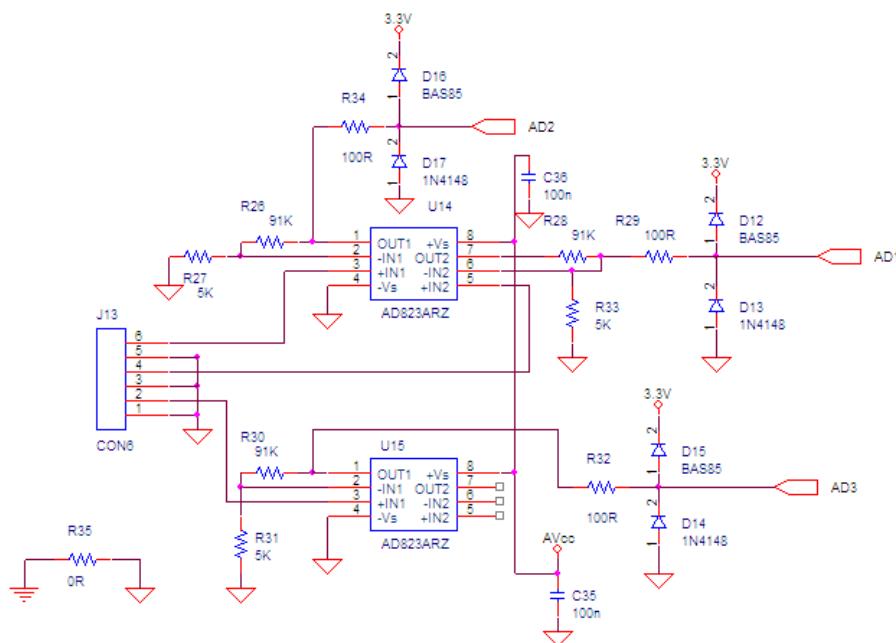
$$U_{vyst} = U_{vst} \left(1 + \frac{R2}{R1}\right)$$

Jako obvod rychlého operačního zesilovače byl použit AD823ARZ od firmy Analog Devices. Tento obvod obsahuje dva rychlé operační zesilovače s frekvencí 16MHz. Vnitřní struktura tohoto obvodu je provedena takto:



Obrázek 3.7: AD823 - vnitřní propojení

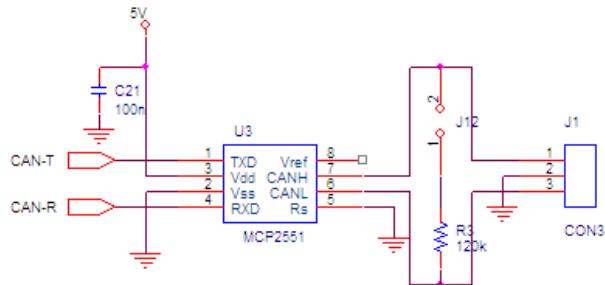
Výstup z každého zesilovače je opatřen diodovým omezovačem. Dioda 1N4148 slouží k potlačení případné záporné úrovně signálu a schottkyho dioda BAS85 omezuje napětí na hodnotu 3,3V + její úbytek napětí, který v případě schotkyho diody je okolo 0,3V, což je ještě přípustná mezní hodnota pro AD převodník. Výsledné schéma použité na procesorové desce je vidět na následujícím obrázku:



Obrázek 3.8: Měřící subsystém

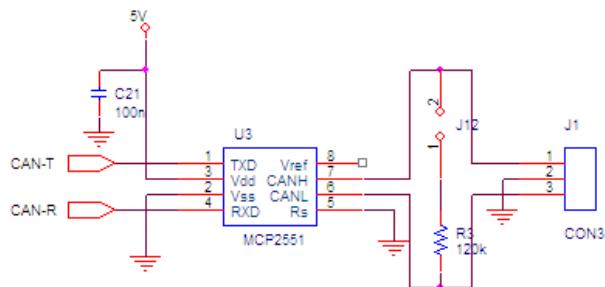
3.1.1.4 Komunikace CAN

CAN řadič je přímo součástí procesoru, proto ke komunikaci po sběrnici je již potřeba jen budič sběrnice. Pro tento účel byl vybrán obvod MCP2551, který podporuje rychlosť dat až 1Mb/s. Zapojení tohoto obvodu :



Obrázek 3.9: Schéma zapojení obvodu MCP2551

Odpor 120 ohmů, který je připojen v závislosti na jumperu k němu přiřazeným, vychází z topologie sběrnice CAN. Jak lze vidět z obrázku



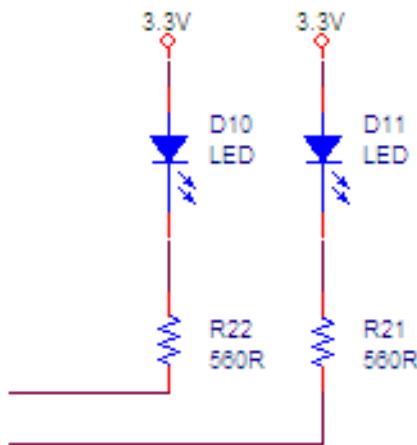
Obrázek 3.10: Schéma CAN sběrnice

odpor 120 ohmů ukončuje na obou stranách sběrnici CAN, proto jen u koncových zařízení se odpor 120 ohmů zapojením Jumperu začlení do obvodu. Napětí 5V pro tento obvod zajišťuje zdrojová deska

3.1.1.5 Ostatní HW

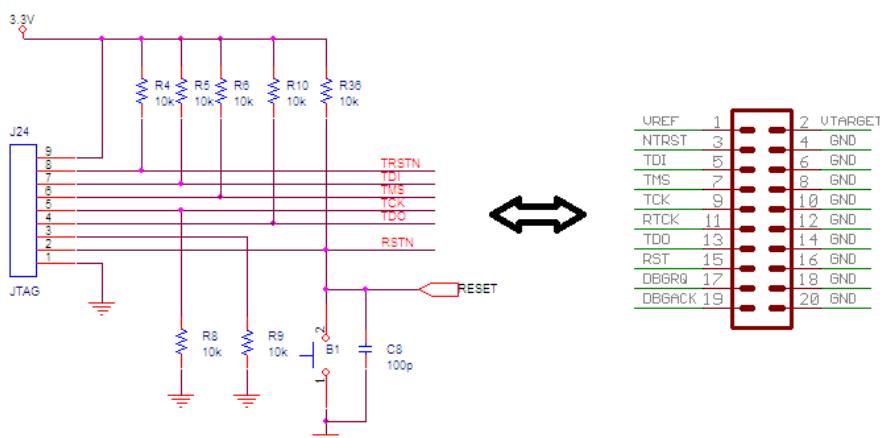
Mezi další periferie procesorove desky patří

- Signalizace - pro signalizaci o poruše a činnosti měniče jsou k dispozici dvě LED diody. (červená, zelená). Zapojeny jsou takovým způsobem, že pokud na pinu procesoru je logická 1., LED dioda nesvítí. V opačném případě, čili pin procesoru v logické 0, LED dioda svítí. Optimální svítivost diody je nastavena odporem.



Obrázek 3.11: Zapojení LED signalizace

- Testovaní I/O piny Procesorová deska má k dispozici 2 I/O piny, které mohou být využity dvojím způsobem, dle nastavení v procesoru.
 - klasické I/O piny - při zkouškách např. vhodné pro tlačítko
 - Sériová linka - výpis dat
- Programovací a ladící rozhraní Jako programovací rozhraní je vyveden JTAG, avšak oproti klasickému 20 pinovému konektoru, má speciální konektor o menším počtu pinů. Toto lze uskutečnit bez jakéhokoliv omezení, pouhým vyvedením aktivních pinů. Speciální konektor byl nutný vytvořit kvůli požadované velikosti DPS.



Obrázek 3.12: Konektor JTAG

3.1.2 Výkonová deska

Výkonová deska je složena ze dvou částí

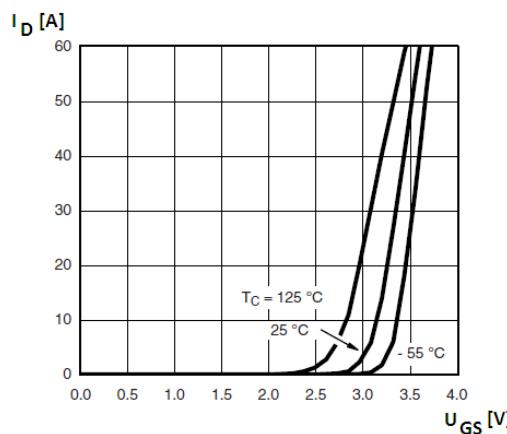
- Trojfázový H-můstek
- Měřící subsystém

3.1.2.1 Trojfázový H-můstek

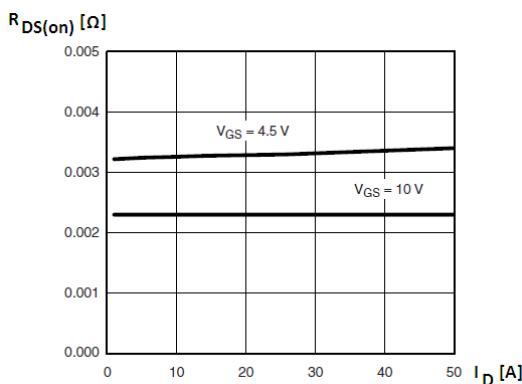
Trojfázový H-můstek se skládá ze 3 polovičních H-můstku. Tranzistor, který byl použit je typu MOSFET s kanálem N a má název SI7336ADP. Tento tranzistor má patici Powerpack SO-8, která je typu SMD. Důležité vlastnosti tranzistoru jsou vyvedeny v následující tabulce a grafech.

Název parametru	Hodnota
maximální napětí U_{DS}	30V
maximální napětí U_{GS}	+/- 20V
Kontinuální proud I_D	25A
$U_{GS(th)}$	(1 - 3) V
Celkový náboj Gate Q_g	36nC
$t_{d(on)}$	24ns
t_r	16ns
$t_{d(off)}$	90ns
t_f	32ns

Tabulka 3.1: Parametry tranzistoru SI7336ADP



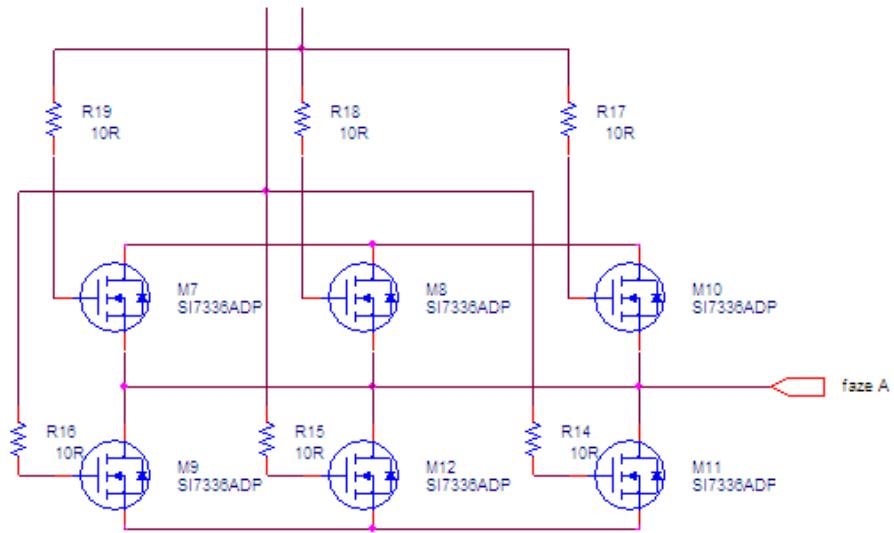
Obrázek 3.13: Parametry tranzistoru SI7336ADP I.



Obrázek 3.14: Parametry tranzistoru SI7336ADP II.

Při spínání induktivní zátěže musí být zabezpečen průchod proudu v době kdy tranzistor není sepnut. Pokud by nahromaděná energie v induktivní zátěži neměla kam proudit, mohlo by dojít k proražení tranzistoru. Potřebná dioda, která chrání tranzistor před proražením a zabezpečuje průchod proudu v době, kdy tranzistor není sepnut je již přímo v pouzdře tranzistoru.

Na každé místo v H můstku byly navrženy tři tyto paralelně spojené tranzistory. Proud protékající větví H-můstku se pak mezi tyto tranzistory rozdělí a zbytečně tak nezatěžuje jednotlivý tranzistor. Výsledný poloviční H-můstek pro jednu fázi motoru pak vypadá takto:

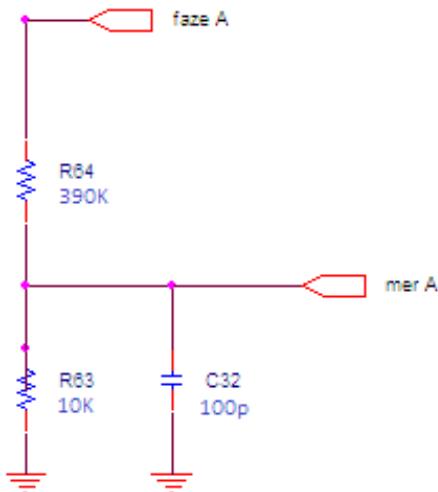


Obrázek 3.15: Schéma polovičního H-můstku

3.1.2.2 Měřící substitutivní systém

Zpracování měřeného signálu obstarává procesorová deska, avšak odpory, na kterých probíhá měření, jsou umístěny na výkonové desce. Pro měření elektromotorických napětí na fázích motoru je na každé fázi umístěn odporový dělič s filtračním kondenzátorem. Pomocí odporového děliče omezíme signál na hodnoty přijatelné pro AD převodník. Filtrační kondenzátor má pak za úkol odfiltrovat rušení od vysokých kmitočtů.

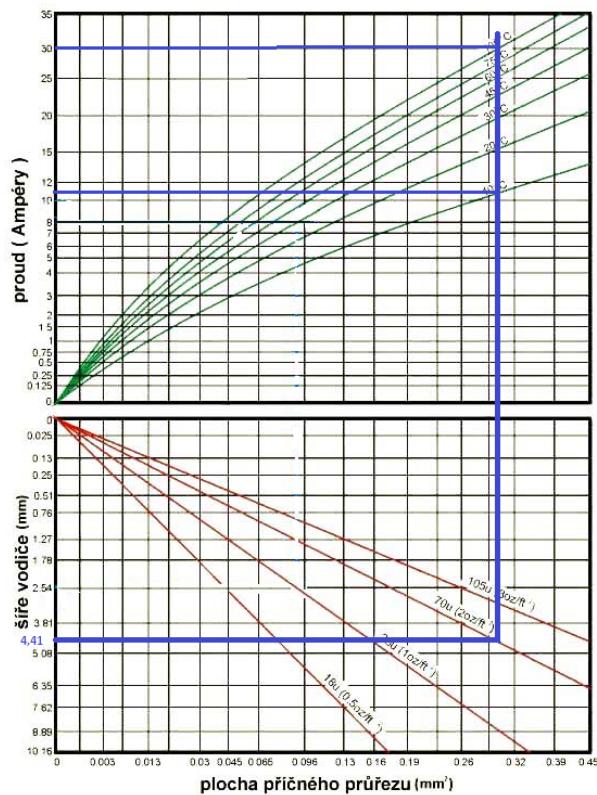
$$U_{vyst} = \frac{R_{63}}{R_{63}+R_{64}} =$$



Obrázek 3.16: Odporový dělič s filtračním kondenzátorem pro fázi A

3.1.2.3 Návrh DPS

Při návrhu DPS výkonové desky byl kladen veliký důraz na proudovou zatížitelnost. Vezmeme-li úvahu maximální proud motoru, který se rovná 30A, musí navržený plošný spoj tento proud přenést. Šířku spoje se návrhem podařilo stanovit na hodnotu v rozmezí 4,4 - 5 mm a aby se dosáhlo větší proudové zatížitelnosti, využilo se možností výrobce plošných spojů a použila se Cu fólie o šířce 70 um. Proudová zatížitelnost se řídí dle normy IPC-D-275. Vynesením do grafu stanoveným touto normou, získáme dovolený proudový rozsah v závislosti na teplotě.



Obrázek 3.17: Proudová zatížitelnost spoje dle normy IPC-D-275

Jak lze vidět z grafu proudová zatížitelnost DPS je dostatečná. Výkonová deska je navržena ve 4. třídě přesnosti a skládá se ze dvou vrstev TOP a BOTTOM. Na TOP vrstvě je trojfázový H-můstek a na BOTTOM vrstvě jsou měřící odpory a zbytek vrstvy tvoří zem.

3.1.3 Zdrojová deska

Zdrojovou desku tvoří spínaný zdroj, který vytváří z napětí baterie 12,8V při plném nabití, 5V pro procesorovou desku. Spínaný zdroj byl zvolen proto, že napětový rozdíl vstupního napětí a výstupního napětí je značně veliký a při zvážení celkového odběru měniče, který se pohybuje okolo 200 mA, by při použití lineárních napětových regulátorů docházelo k velkým ztrátám.

$$P_{ztrat} = \Delta U * I = (12,8 - 5)V * 200mA = 1,56W$$

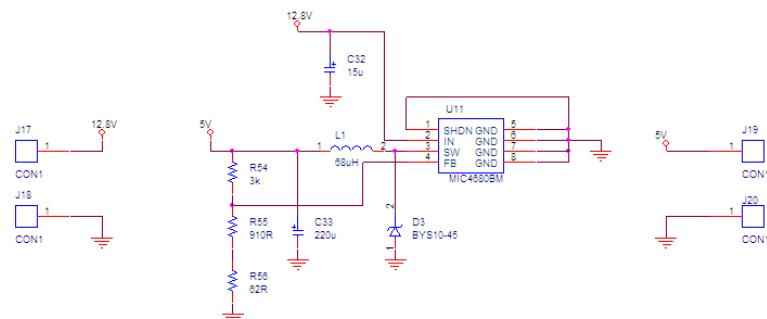
Nevýhodou spínaného zdroje je rušení, které znemožňuje vést v jeho blízkosti datové spoje. Při rozměrech procesorové desky a snahy umístit všechny součástky pouze na jedné vrstvě DPS, byl spínaný zdroj umístěn na tuto desku.

Hlavní částí spínaného zdroje je obvod MIC4680BM od firmy Micrel, což je regulovatelný spínaný napětový regulátor typu step-down. Jeho hlavní parametry jsou

Název parametru	Hodnota
Spínací frekvence f	200KHz
Maximální proudový odběr I	1,3A
Rozsah vstupního napětí	4 - 34V

Tabulka 3.2: Parametry obvodu MIC4680BM

Dioda, cívka a kondenzátor spolu odpory, který nastavují hodnotu výstupního napětí byly zvoleny dle optimálních hodnot pro tento obvod uvedených v jeho dokumentaci. Úplné schéma spínačního zdroje je na tomto obrázku.



Obrázek 3.18: Schéma spínačního zdroje

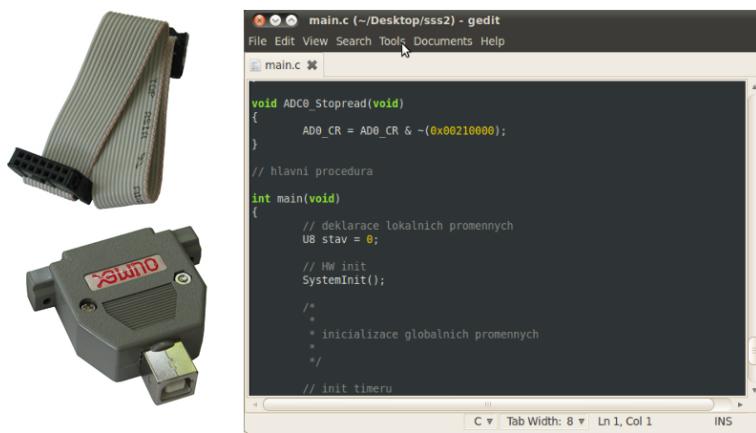
3.2 Software

3.2.1 Vývojové nástroje

Základní kritérium pro volbu vývojových nástrojů pro tuto aplikaci bylo

- volná dostupnost
- plná verze, bez omezení velikosti projektu
- dostupná podpora na internetu

Všechny tyto kritéria splňuje GNUARM ve spojení s OPENOCD. GNUARM je využit jako C/C++ a assemblerovský překladač a linker. OPENOCD je využit jako JTAG debugger a programátor. Oba tyto programy se ovládají z příkazové řádky, proto je ještě zapotřebí grafické prostředí pro psaní programu. Pro tento účel stačí využít gedit, který je přímo součástí systému linux a nabízí zvýraznění kódu a možnost otevřít dva panely v jednom okně. Hardwarovou komunikaci s procesorem obstarává pak ARM-USB-TINY, což je JTAG debbuger, který je plně podporován programem OPENOCD.



Obrázek 3.19: Vývojové nástroje

3.2.2 Stavba programu pod GNUARM

Program psaný pod vývojovým nástrojem GNUARM obsahuje tři základní části

- Assemblerovský zavaděč, který je umístěn na začátku programu a definuje nezbytné nastavení pro správný běh procesoru

- nastavení základních vektorů přerušení, procesor je očekává na prvních 8 pozicích od adresy 0 (reset, nedefinovaná instrukce, prefetch abort, data abort, IRQ, FIQ)
 - nastavení PLL a DIV pro určení taktu jádra procesoru
 - Určení ukazetele zásobníku pro každý mód procesoru (Uživatelský, Systémový, Supervisor, IRQ, FIQ, Abort, Nedefinovaný)
 - Kopírování sekce .data (inicjalizované proměnné) do RAM
 - Vynulování .bss (neinicjalizované proměnné) sekce
 - Skok do hlavního programu
- Skript pro sestavení programu do paměti procesoru - Po kompliaci programu má program 4 hlavní sekce :
 - .text - Spustitelný kód
 - .rodata - proměnné určené pouze ke čtení (konstanty)
 - .data - inicializované proměnné
 - .bss - neinicjalizované proměnné

Všechny tyto sekce je třeba umístit na pamětová místa procesoru. Nejdříve skript nadefinuje dostupnou paměť procesoru. Pro procesor LPC2919 je tato sekce definovaná

```

/* org - počátek */
/* len - délka */
/* r - čtení */
/* x - spustitelný */
/* w - zápis */
MEMORY
{
  flash (rx) : org = 0x20000000, len = 0xC0000
  sram0(rw) : org = 0x80000000, len = 0x8000
  sram1(rw) : org = 0x80008000, len = 0x4000
}

```

Poté již do těchto míst přiřadit dříve zmíněné sekce programu.

```

SECTIONS
{
    .text :
    {
        *start.o (.text) /* nahráváme do flash na adresu 0 startup musí mít ve slově
        start */

        *(.text) /* nahráváme do flash na adresu 0 + sizeof(start.o) kod všech
        zdrojových kódů */

        } > flash /* tato sekce se nahraje do flash paměti */
        . = ALIGN(4); /* zarovnání konce do paměti 32bitů */

        .rodata :
        {
            *(.rodata) /* zápis souboru read only (const) do flash */
            } > flash /* tato sekce se nahraje do flash paměti */
            . = ALIGN(4);

        _etext = . ; /* uložení do proměnné adresa konce sekce .text */

        .data : AT (_etext) /* na adresu e_text nahrát všechny rw inicializované
        proměnné */
        {
            _data = . ;
            *(.data)
            } > sram0 /* program je čeká v sram, proto v assemblerovském zavaděče nutno
            zajistit zkopírování do sram0 */
            . = ALIGN(4);

        _edata = . ;

        PROVIDE ( edata = . );

        .bss :
        {
            _bss = . ;

```

```

*( .bss )
*(COMMON)
} > sram0 /* Nedefinování prostoru v paměti pro neinicializované proměnné */
. = ALIGN (4);

_ebss = . ;

_end = . ;

PROVIDE (_end = .);

}

```

- Hlavní program (ASM,C,C++) - Zaváděč ke konci jeho běhu přepíná do hlavního programu, většinou do funkce main().

3.2.3 Konfigurace JTAG pod OPENOCD

Konfigurační skript, který nastaví OPENOCD pro spojení s procesorem přes hardwarový JTAG debugger v naší aplikaci vypadá takto:

```

#LPC-2919 CPU

telnet_port 4444 # přístup k procesoru lze realizovat přes telnet

#define JTAG HW debuggeru
interface ft2232
ft2232_device_desc "Olimex OpenOCD JTAG ARM-USB-TINY-H"
ft2232_layout olimex-jtag
ft2232_vid_pid 0x15ba 0x002a

#rychlost komunikace JTAG
jtag_khz 2000

#nedefinování procesoru
set _CHIPNAME lpc2919

```

```

set _ENDIAN little
set _CPUTAPID 0x0596802b
set _ETBTAPID 0x1B900F0F

#reset_config trst_and_srst srst_pulls_trst
# typ resetu procesoru
reset_config trst_and_srst combined

# reset zpoždění
jtag_nsrst_delay 200
jtag_ntrst_delay 200

#nedefinování procesoru
set $_TARGETNAME [format "%s.cpu" $_CHIPNAME]

jtag newtap $_CHIPNAME etb -irlen 4 -ircapture 0x1 -irmask 0xf -expected-id $_ETBTAPID
jtag newtap $_CHIPNAME cpu -irlen 4 -ircapture 0x1 -irmask 0xf -expected-id $_CPUTAPID

# Create the ".cpu" target
target create $_TARGETNAME arm966e -endian little -chain-position $_TARGETNAME
-variant arm966e

etm config $_TARGETNAME 8 normal full etb
etb config $_TARGETNAME $_CHIPNAME.etb

$_TARGETNAME configure -work-area-virt 0 -work-area-phys 0x80000000
-work-area-size 0xC000 -work-area-backup 0

#nedefinování flash paměti
#flash bank <driver> <base> <size> <chip_width> <bus_width>
flash bank lpc2900.flash lpc2900 0x20000000 0xC0000 0 0 $_TARGETNAME 120000
#flash bank $_FLASHNAME lpc2900 0 0 0 0 $_TARGETNAME 125000
arm7_9 dbgrq enable
arm7_9 dcc_downloads enable

```

```
$_TARGETNAME configure -event reset-init {  
    flash probe 0 # inicializace flash  
  
    flash erase_sector 0 0 18 # smazani celé flash pameti po sektorech  
  
    flash write_image firmware.bin 0x20000000 # nahranie aplikacie do flash  
}  
  
init  
  
reset init # reaguje na udalosti drive nadefinovanou  
  
reset run # po nahraní aplikacie udalostí na reset init provede reset a spustí  
nahraný kód
```

3.2.4 Inicializace

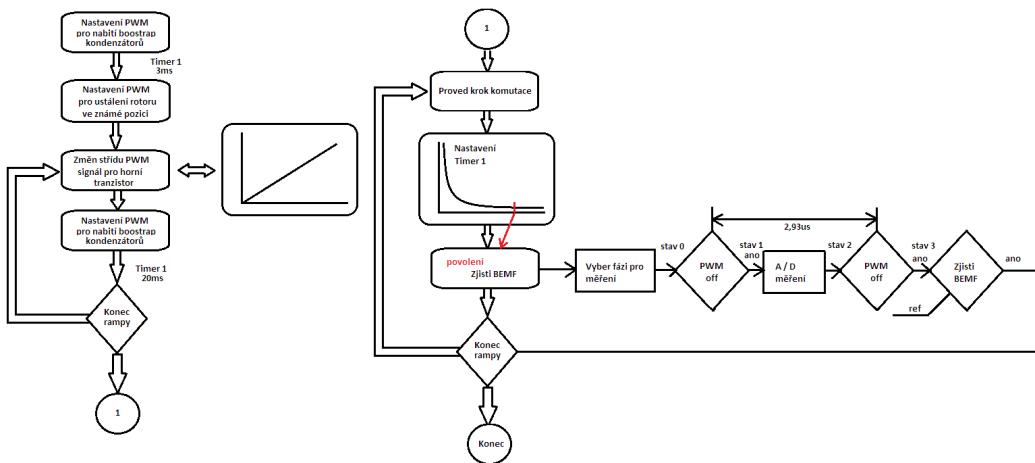
Nejdříve než se samotný algoritmus bude vykonávat je třeba inicializovat všechny potřebné periferie procesoru.

- Hodiny procesoru nastaveny 120MHz
- PWM - hodiny pro blok PWM jsou nastaveny na 120MHz, perioda PWM při testech se ukázala být nejvhodnější 30KHz. Při inicializaci je nutné na všechny PWM signály připojené na gate tranzistorů H-můstku přivést střídu 0, čili 0V.
- ADC - ADC převodník je nastaven na přesnost 10bitů a na maximální rychlosť A/D převodníku 4,5MHz (2,22us).
- CAN - Hodiny CANu nastaveny na 60MHz, z které již lze nastavit přenosová rychlosť na 1Mb/s
- Časovače - Perioda časovače je nastavena na 120MHz.

3.2.5 Algoritmus pro BLDC motor

3.2.5.1 Start motoru

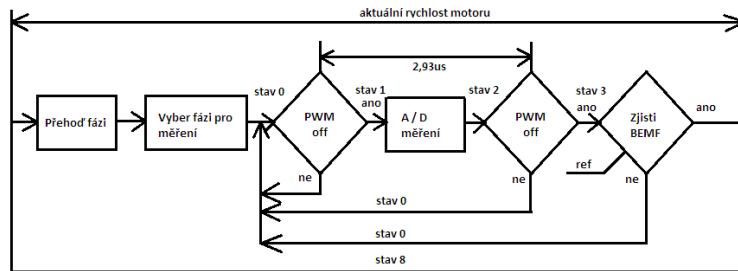
Start motoru, který již byl detailně popsán v kapitole 2. pracuje dle tohoto vývojového diagramu. Startovací rampy, které byly použity jsou vygenerovány externím algoritmem a do programu byly vloženy jako statická pole. Pro testy bylo výhodná grafická p



Obrázek 3.20: Vývojový diagram startu BLDC motoru

3.2.5.2 Regulace pozice motoru

Stavový automat, který zajišťuje správnou komutaci motoru podle měření elektromotorických napětí jednotlivých fází motoru popisuje následující schéma:



Obrázek 3.21: Back-EMF Regulátor

- Blok Přehod' fázi - obsahuje 6 stavový automat, který nastavuje hodnoty PWM signálů podle 6 krokové komutace.
- Blok Vyber fázi pro měření - Analyzuje nový stav motoru, který nastavil předchozí blok. Vybere zrovna nebuzenou fázi motoru a nastaví její příslušný kanál A/D

převodníku pro blok A/D měření.

- 1. Podmínka PWM off - Kontrolní blok, který sleduje průběh PWM signálu horního tranzistoru buzené fáze motoru. Pokud je tranzistor rozepnut povolí algoritmu, přejít do bloku A/D měření. Pokud je tranzistor sepnut, stavový automat vyčká v tomto bloku do doby než bude opět tranzistor rozepnut.
- blok A/D měření - Aktivuje kanál A/D převodníku definovaný blokem Vyber fázi pro měření. Vyčká na dokončení převodu a získá měřenou hodnotu z registru A/D převodníka.
- 2. Podmínka PWM off - se nyní používá pro ověření pravosti měřené hodnoty. Pokud je stále horní tranzistor rozepnut hodnota se nastaví jako platná, v opačném případě je hodnota neplatná a algoritmus se vrací k 1. podmínce PWM off.
- Blok zjistí Back-EMF - Přebírá platnou hodnotu získanou z měření A/D převodníku. A podle směru proudu a referenční hodnoty Back-EMF napětí rozhodne, zda je čas pro přenutí do dalšího stavu. Rozhodnutí probíhá dle následující tabulky :

Směr proudu	A do B	A do C	B do C
Měřící fáze motoru	C	B	A
Charakteristika BEMF napětí	0V do BEMF	Vcc do BEMF	0V do BEMF
Rozhodovací podmínka	$hod. \geq BEMF$	$hod. \leq BEMF$	$hod. \geq BEMF$

Tabulka 3.3: Rozhodovací úrovně signálu Back-EMF I.

Směr proudu	B do A	C do A	C do B
Měřící fáze motoru	C	B	A
Charakteristika BEMF napětí	Vcc do BEMF	0V do BEMF	Vcc do BEMF
Rozhodovací podmínka	$hod. \leq BEMF$	$hod. \geq BEMF$	$hod. \leq BEMF$

Tabulka 3.4: Rozhodovací úrovně signálu Back-EMF II.

Reference Back-EMF napětí byla nastavena testováním. Hodnota, která se jevila jako nejlepší pro daný motor je 0,5V.

Z doby jednoho úspěšného průchodu tímto algoritmem(stav 8) se dá vypočítat skutečná rychlosť motoru, jelikož délka áprůchodu je přímo závislá na měření elektromotorických napětí jednotlivých fzí motoru. Zavedeme-li

Δt [s] ... úspěšný průchod algoritmem (stav 8)

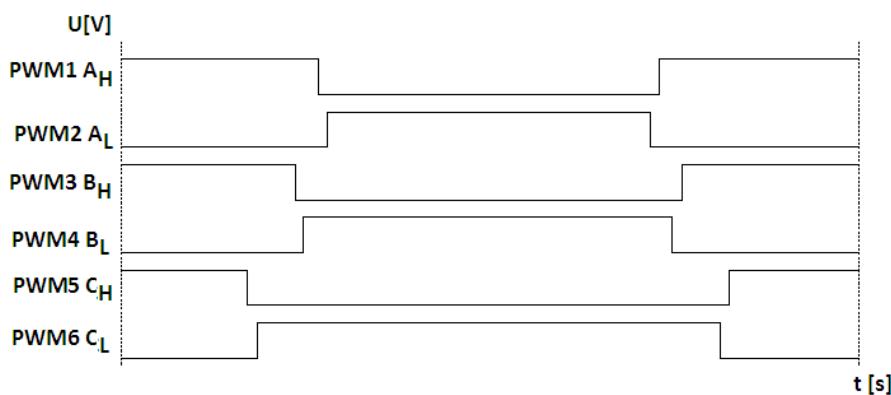
, pak skutečná rychlosť motoru je rovna

$$v_{motor} = \frac{6 * \Delta t * p}{60} \left[\frac{ot}{min} \right].$$

3.2.6 Řídící algoritmus pro PMSM motor

3.2.6.1 Nároky na periferie procesoru

Tento algoritmus klade velké nároky na PWM modul procesoru. Jelikož algoritmus generuje centrovany signál, musí PWM modul mít nejméně 6 kanálů PWM, u kterých se dá nastavit jak vzestupná tak sestupná hrana. Spínání dolních tranzistorů je komplementární ke spínání horních tranzistorů. Z toho plyne další důležitý problém, který je třeba uvážit ke správné funkci algoritmu. Sepnutí a rozepnutí tranzistoru je dlouhé řádově desítky nanosekund a doba sepnutí se nemusí rovnat době rozepnutí tranzistoru. Při komplementárním spínání horního a dolního tranzistoru v jedné fázi tedy mohou vznikat zktatové proudy, z důvodu toho, že dolní tranzistor již je sepnut a horní tranzistor ještě z daleka uplně není rozepnut. Proto se v tomto místě vloží takzvané mrtvé pásmo, kdy po určitou dobu jsou oba tranzistory rozepnutý.



Obrázek 3.22: Vložení mrtvého pásmá do kritických míst

3.2.6.2 Zavedení desetinného čísla

Jelikož výpočet časů t_a, t_b, t_c pro generování vektorové PWM je matematicky náročné a je využita goniometrická funkce sinus, je třeba zavést desetinný typ proměnné. Použitý procesor LPC2919/01 nemá hardwarovou podporu pro práci s desetinnými čísly, a tak

zavedeme speciální strukturu typu union, která bude s určitou přesností reprezentovat desetinné číslo.

```
typedef union
{
    signed short celek;
    struct
    {
        unsigned short des: 10;
        signed short cely: 6;
    }part;
}Q6_10;

typedef union
{
    signed long celek;
    struct
    {
        unsigned long des: 16;
        signed long cely: 16;
    }part;
}Q16_16;
```

V případě struktury Q6_10, je pro celé číslo vyhrazeno 6bitů (-32 až 31) a pro desetinné číslo 0 až 1023 ($1023/1024 = 0.9990234$). Přesnost pak je $1/1024 = 0.0009765$. Podobným způsobem lze definovat i strukturu Q16_16. Pro tyto speciální struktury je nutné nadefinovat základní matematické operace jako sčítání, odčítání, dělení a násobení.

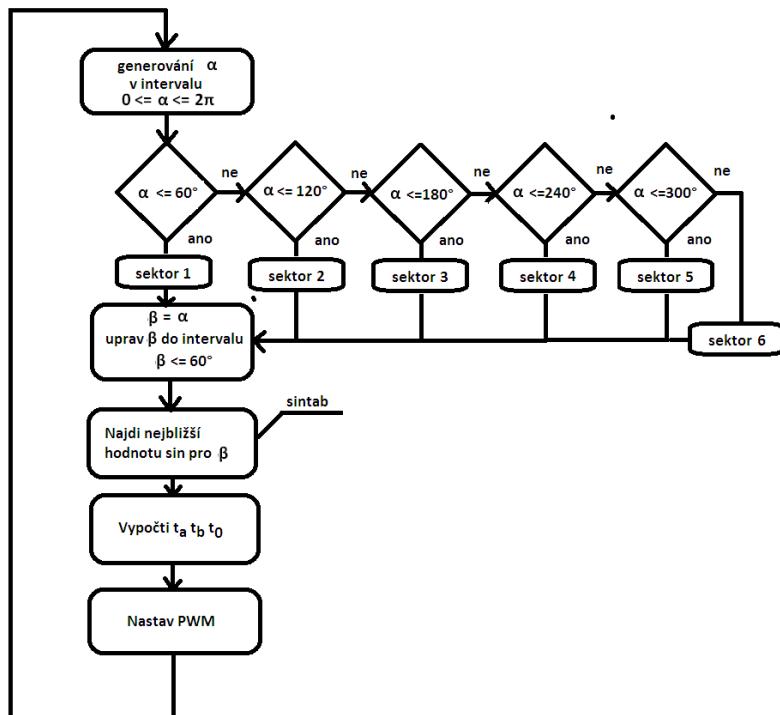
- sčítání : výsledek.celek = a.celek + b.celek
- odčítání : výsledek.celek = a.celek - b.celek
- dělení : výsledek.celek = (((*signedlong*)a.full << 11)/b.full) + 1)/2
- násobení : výsledek.celek = ((*signedlong*)a.full * b.full + 512) >> 10

3.2.6.3 Definování funkce sinus

Jelikož výpočet sínu je pro procesor časově náročný, využijeme statické pole a ve formátu Q6_10 zapíšeme hodnoty sinu pro $0 \text{--} \frac{\pi}{2}$ s krokem 0.0157rad. Tato approximace sinů pro aplikaci plně postačuje.

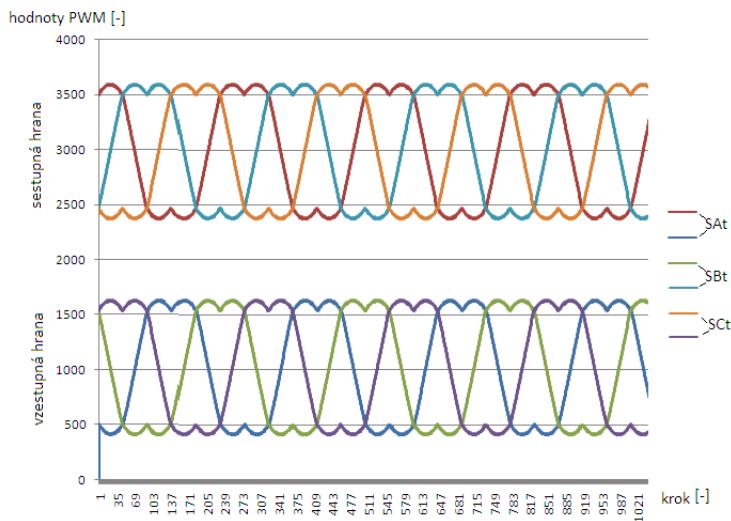
```
const signed short sintab[101]=
{
0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 223, 239, 255,
270, 286, 301, 316, 332, 347, 362, 377, 392, 406, 421, 436, 450, 465, 479, 493,
507, 521, 535, 548, 562, 575, 589, 602, 615, 627, 640, 652, 665, 677, 689, 701,
712, 724, 735, 746, 757, 768, 778, 789, 799, 809, 819, 828, 837, 847, 856, 864,
873, 881, 889, 897, 905, 912, 919, 926, 933, 940, 946, 952, 958, 963, 969, 974,
979, 983, 988, 992, 996, 999, 1003, 1006, 1009, 1011, 1014, 1016, 1018, 1019,
1021, 1022, 1023, 1023, 1024, 1024 };
```

3.2.6.4 Vývojový diagram algoritmu



Obrázek 3.23: Vývojový diagram algoritmu SVPWM

Výstupem algoritmu jsou časy definování vzestupných hran a sestupních hran PWM signálů. Na následujícím obrázku lze vidět, jak se mění hrany PWM signálů pro horní spínače H-můstku po 1000 krocích algoritmu. PWM signály pro dolní spínače jsou k těmto komplementární.



Obrázek 3.24: Vývoj vzestupných a sestupných hran PWM pro horní spínače H-můstku

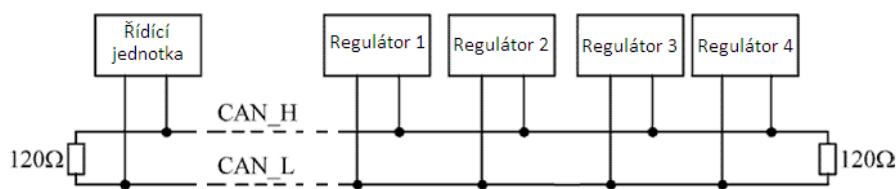
3.2.7 CAN komunikace

Tato sběrnice byla vybrána pro přenos řídících a stavových dat díky těmto vlastnostem

- Vysoká rychlosť přenosu dat 1Mbit/s při délce sběrnice do 40m
- Rozlišení zpráv identifikátorem
- Selekce přijímaných identifikátorů zpráv
- Prioritní přístup zabezpečující urychljené doručení významných zpráv
- Diagnostika sběrnice např.: chyba doručení zprávy
- Vysoká provozní spolehlivost

3.2.7.1 Zapojení CAN sběrnice

CAN sběrnice je zapojena na quadrotoru následujícím způsobem.



Obrázek 3.25: Zapojení CAN sběrnice na Quadrotoru

3.2.7.2 nastavení sběrnice CAN

- Jelikož je CAN broadcastová sběrnice a každý účastník si přečte zprávu, která je v danou chvíli na sběrnici. Jelikož regulátor nezajímá zpráva o dané rychlosti jiného regulátoru ale zajímá ho zpráva od řídící elektroniky o změně referenční rychlosti otáčení. Každý regulátor má tedy nastavený filtr, který propustí jen příchozí zprávy, které ho opravdu zajímají. Zabrání se tak zbytečnému přerušení činnosti procesoru
- CAN sběrnice má prioritní rozhodování. Tedy zpráva s nejnižším identifikátorem má přednost, pokud budou dva účastníci na sběrnici chtít poslat zprávu ve stejný okamžik.

3.2.7.3 Zprávy na sběrnici CAN

identifikátor zprávy	popis
10	chyba, zastav motor
11	změna reference 1byte - číslo motoru, 2byte - referenční rychlosť
12	aktuální rychlosť 1byte - číslo motoru, 2byte - aktuální rychlosť

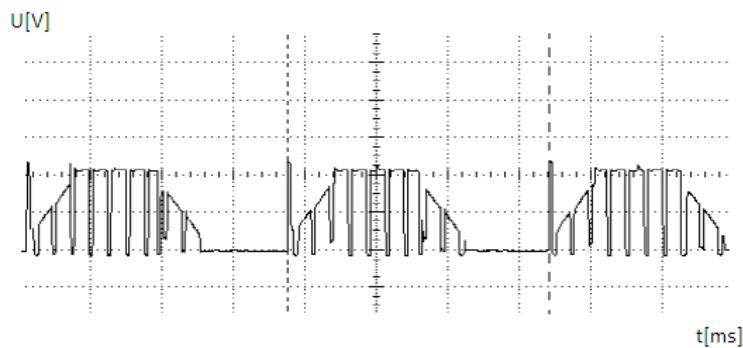
Kapitola 4

Testy realizovaného regulátoru

Synchronní 14 pólový motor RC 600/30/7 od firmy Megamotor, který je použit v Quadra-toru má tyto parametry

otáčky na volt [RPM/1V]	maximální proud [A]	vnitřní fázový odporník [$m\Omega$]	počet závitů
640	30	23,5	7

Analýzou funkce komerčního měniče, kde měření na jedné fázi střídače ukázalo lichoběžníkový průběh jsem předpokládal, že dáný bezkartáčový motor má lichoběžníkové vinutí a tudíž se jedná o BLDC motor.



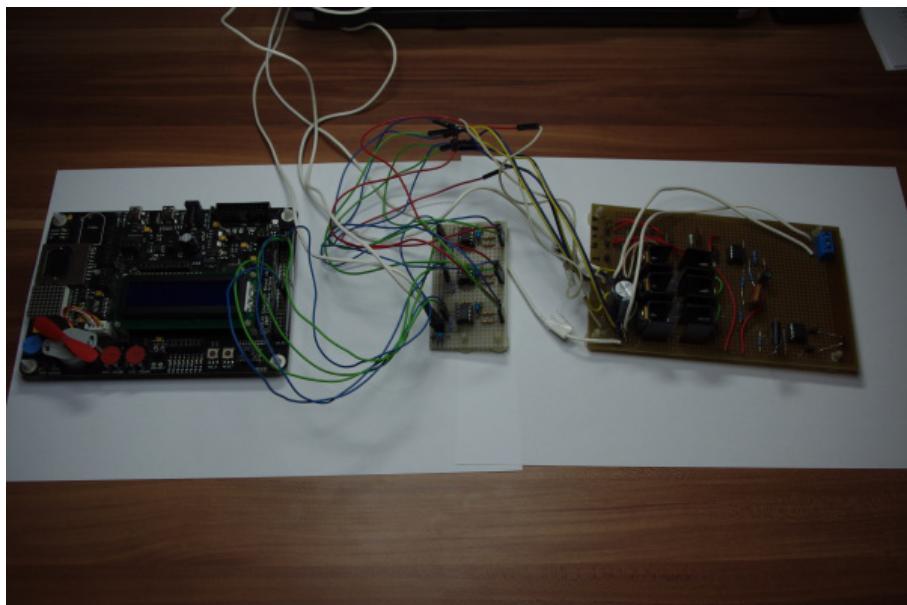
Obrázek 4.1: Měření na současném měniči

Pro lepší odladění algoritmu pro regulaci BLDC motoru byl zakoupen méně výkoný 2 pólový motor BL3056, jehož známé parametry jsou uvedeny v tabulce

Napětí [V]	Maximální výkon [W]	Maximální rychlosť [RPM]	Točivý moment [Nm]
24	60	10800	0,227

Tabulka 4.1: Parametry zkušebního motoru BL3056

Dále byl postaven zkušební měnič složený z procesorového kitu, kde byly postupně použity kity LPC-P2129, LPC-P2919 a LPC2148 (v3) Edu board, z modulu, který obsahoval budiče H-můstku(nejdříve L6385/7, poté současné FAN7382) a nakonec z trojfázového H-můstku, který byl složen z tranzistorů STP55NF06 Tento tranzistor má obdobné vlastnosti jako Si7336ADP, až na maximální proud I_D , který je dvojnásobný (50A). Sestavení zkušebního měniče z těchto komponent je pak vidět na následujícím obrázku

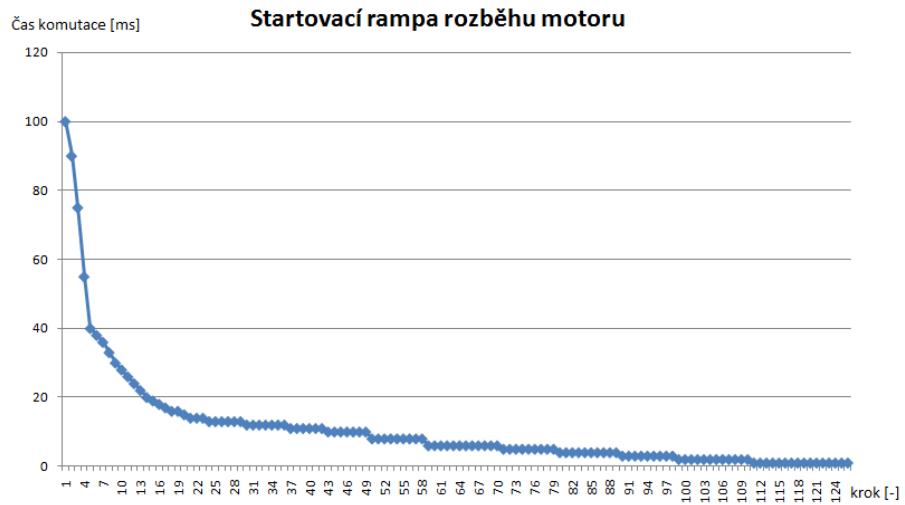


Obrázek 4.2: Sestava zkušebního měniče

Algoritmus BLDC byl odladěn na tomto zkušebním měniči a motor BL3056 se dle něj rozeběhl v otevřené smyčce k požadovaným otáčkám. V těchto otáčkách zjistil v požadovaném počtu BEMF napětí a přepnul se do regulovatelné smyčky, která zajišťovala automatickou komutaci.

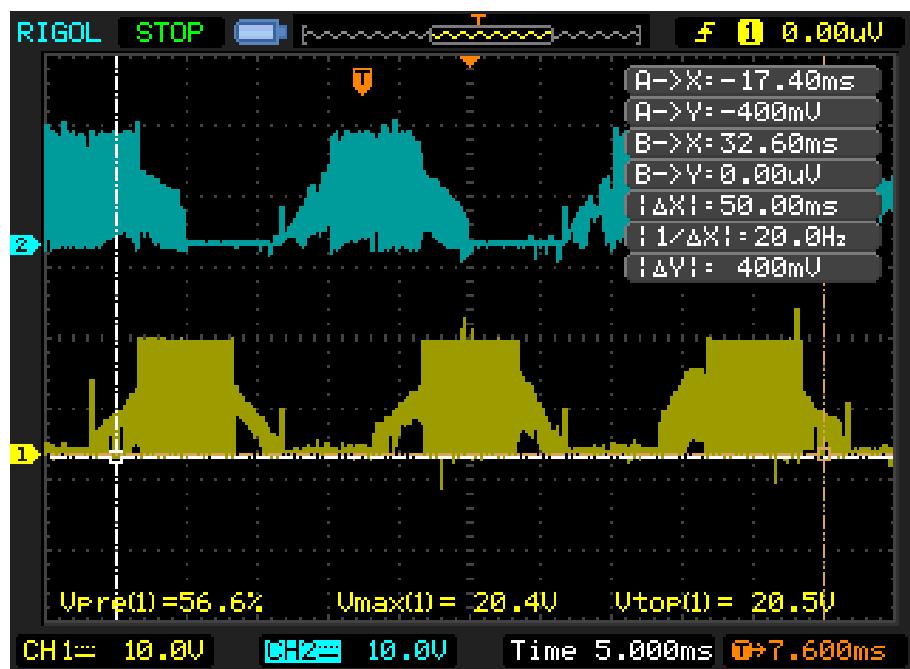
Po tomto úspěchu byl algoritmus přenesen do navrhovaného měniče, kde se choval stejným způsobem. Tím se zkontoval program pro BLDC motor a zároveň i funkčnost hardware navrhovaného měniče po několika drobných úpravách HW.

Připojen byl proto motor RC 600/30/7. Start motoru v otevřené smyčce po nalezení optimální rampy rozeběhu, kterou lze vidět na následujícím obrázku

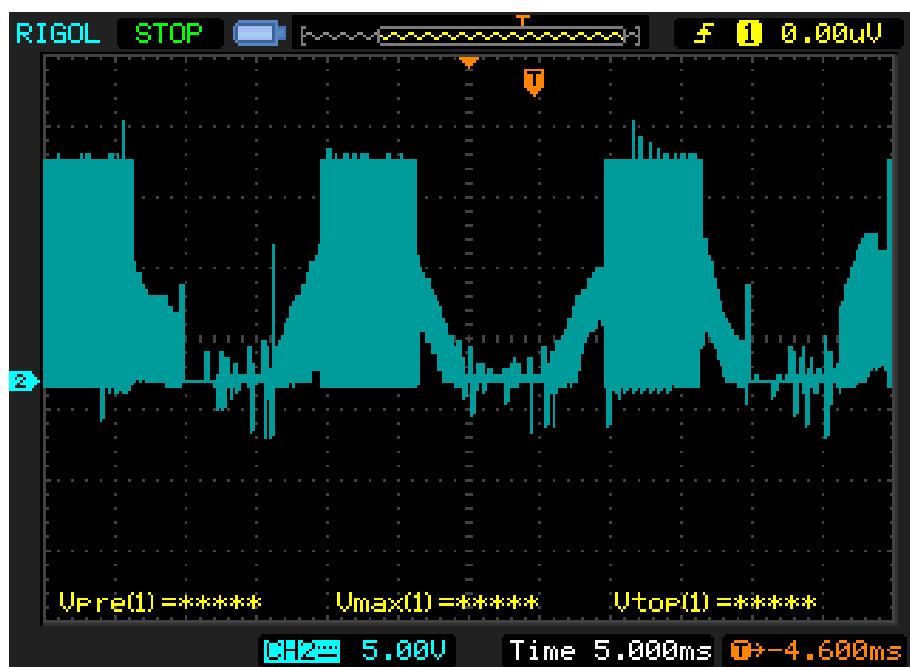


Obrázek 4.3: Optimální rampa pro rozběh motoru RC 600/30/7

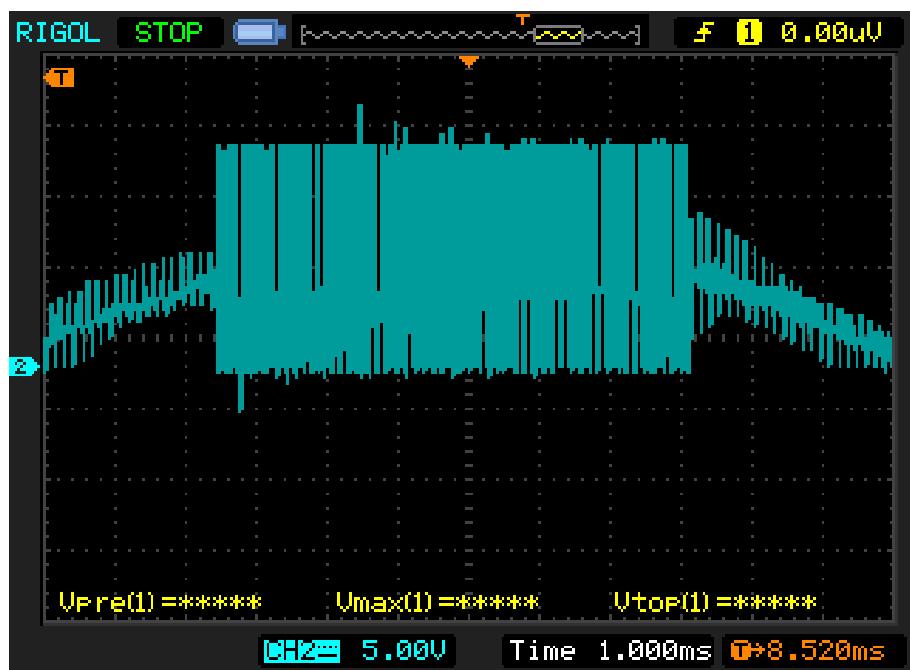
se rozeběhl a dokázal běžet v otevřené smyčce ve stanovených otáčkách, avšak sebe-menší odpor, který byl kladen na rotor ho zastavil a do regulační smyčky se ani nedostal. Proud protékající motorem se pohyboval okolo 3A. Měření napětí na fázích motoru RC 600/30/7 lze vidět na následujících obrázcích.



Obrázek 4.4: Testy měniče I.



Obrázek 4.5: Testy měniče II.



Obrázek 4.6: Testy měniče III.

Po těchto nezdarech jsem se rozhodl implementovat algoritmus pro bezkartáčový motor se sinusovým polem čili PMSM. To ale znamenalo změnit hardware navrhovaného měniče na současnou verzi kvůli použitému procesoru. Jak jsem se již zmínil v rozboru

tohoto algoritmu, jsou zde kladený větší požadavky na PWM modul, který procesor LPC2129 nesplňoval. V současné verzi měniče byl procesor vybrán tak, aby mohl řídit oba druhy bezkartáčových motorů a zároveň měl dostatečnou výkonovou rezervu pro více sofistikované řízení.

Po vyzkoušení správné činnosti algoritmu SVPWM, byl aplikován na motor RC 600/30/7. Rozběh motoru tímto algoritmem se sice podařil, ale motorem při rozběhu, tedy rychlostech blížících se k nule v nezatíženém stavu protékal kontinuální proud okolo 23 A, což se blíží maximálnímu proudu motoru. Tento proud navrhovaný měnič nevydržel, i když byl i na tento proud navrhnut a po pár minutách zatížení tímto proudem měnič vyhořel. Proto jsem možnost řízení motoru 600/30/7 tímto algoritmem zavrhl a vrátil se zpět k BLDC motoru. Nyní již byl k dispozici procesor s lepším PWM modulem, který navíc hardwarově synchronizoval start AD převodníku s PWM signálem. Aplikování algoritmu BLDC na novém procesoru zlepšila o trochu zmíněné nedostatky, avšak pro rychlé změny referenční hodnoty či pevné zachycení regulační smyčky, nadále nebylo možné.

Kapitola 5

Závěr

Výsledkem této práce je navržený a realizovaný regulátor pro bezkartáčové motory zejména pro 600/30/7, který je použit ve stroji quadrotor. Regulátor je schopen rozeběhnout tento motor v otevřené smyčce do požadovaných otáček ne větších než 2500ot/min. Zároveň je schopen reagovat na změnu referenčních otáček, kterou přijme po CAN sběrnici, pomalým poklesem či nárůstem na požadované otáčky. Bohužel kvůli špatně chycené automatické komutaci je toto chování nestabilní a při zatížení rotoru přestává motor pracovat. Na zkušebním motoru BL3056 se tento regulátor chová korektně. Při rozběhu se motor dostane na požadované otáčky a přejde do regulační smyčky pro automatickou komutaci v závislosti na měření BEMF napětí.

Kvůli těmto problémům není možné zatím zařadit tento regulátor do quadrotoru a tím i znemožňují identifikovat nové statické a dynamické parametry stroje.

Další prací na regulátoru by měla být lepší analýza měřených BEMF napětí, tak aby byl motor schopný přejít do regulační smyčky automatické komutace a návrh nového H-můstku pro lepší proudovou zatížitelnost.

Literatura

HEINRICH, P. (2008), *Experimentální létající stroj - Základní řízení*, Diplomová práce ČVUT.

MAREČEK, J. (2008), *Experimentální létající stroj - Orientace v prostředí*, Diplomová práce ČVUT.

HAVLENA, V.; ŠTECHA, J. (2000), *Moderní teorie řízení*, Praha: Vydavatelství ČVUT.

ROUBAL, J., PEKAŘ, J., PACHNER, D. a HAVLENA, V.; (2005), *Moderní teorie řízení – Cvičení*, Praha: Vydavatelství ČVUT.

MICROCONTROLLER DIVISION APPLICATIONS (2000), *ST72141 BLDC MOTOR CONTROL SOFTWARE AND FLOWCHART EXAMPLE*

MICROCONTROLLER DIVISION APPLICATIONS (2000), *PWM MANAGEMENT FOR BLDC MOTOR DRIVES USING THE ST72141*

MICROCONTROLLER DIVISION APPLICATIONS (2000), *BLDC motor start routine for the ST72141 microcontroller*

INFEON TECHNOLOGIES (2000), *Sensorless Field Oriented Control for PMSM motors*

INFEON TECHNOLOGIES (2000), *Sensorless Field Oriented Control for PMSM motors*

MICROCONTROLLER DIVISION APPLICATIONS (2007), *Space vector modulation using 8bit ST7MC microcontroller*

MICROCONTROLLER DIVISION APPLICATIONS (2000), *An introduction to sensorless brushless DC motor drive application with the ST72141*

FREESCALE SEMICONDUCTOR (2005), *PMSM and BLDC Sensorless Motor Control using the 56F8013 Device*

NXP SEMICONDUCTOR (2009), *Sensorless single-shunt FOC on LPC2900*

JIANWEN SHAO (2003), *Direct Back EMF Detection Method for sensorless brushless DC (BLDC) Mtor Drives*, Blacksburg, Virginia.

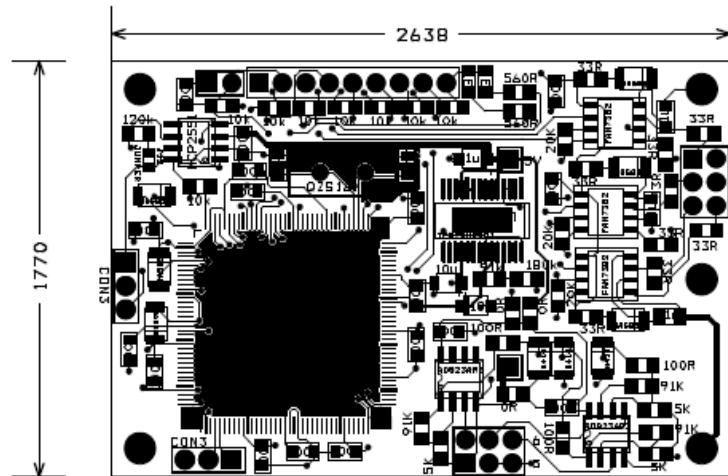
RANDY YATES (2009), *Fixed-Point Arithmetic: An Introduction*

D.W. HAWKINS (2006), *Real-time processing with the Philips LPC ARM microcontroller; using GCC and the MircoC/OS-II RTOS*

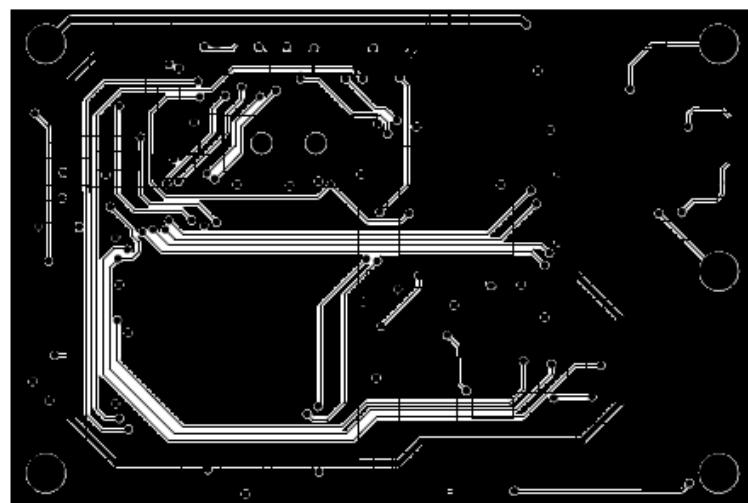
NXP SEMICONDUCTOR (2007), *Brushless DC motor control using the LPC2141*

Příloha A

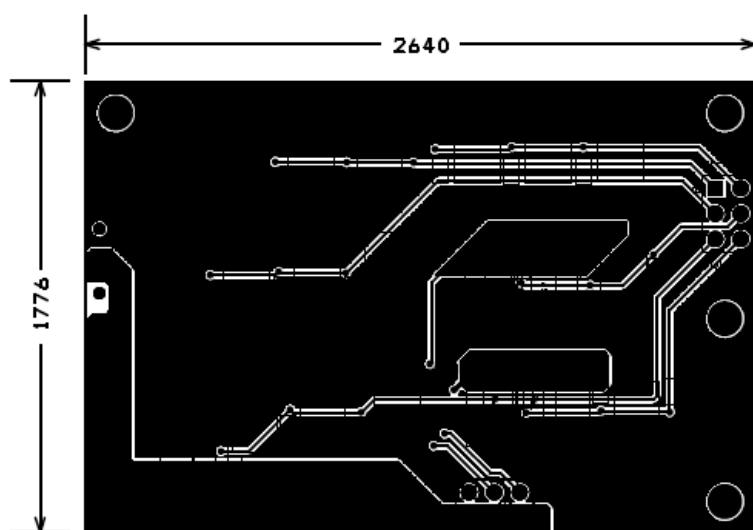
Motivy desek plošných spojů



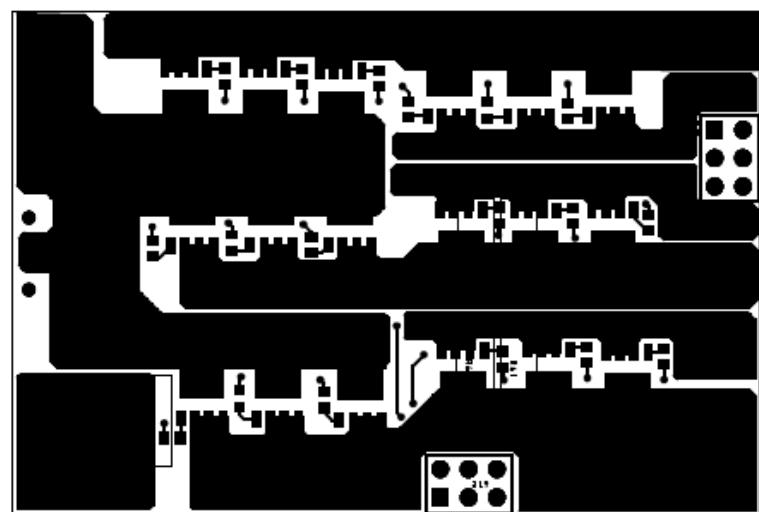
Obrázek A.1: Procesorová deska - Top vrstva



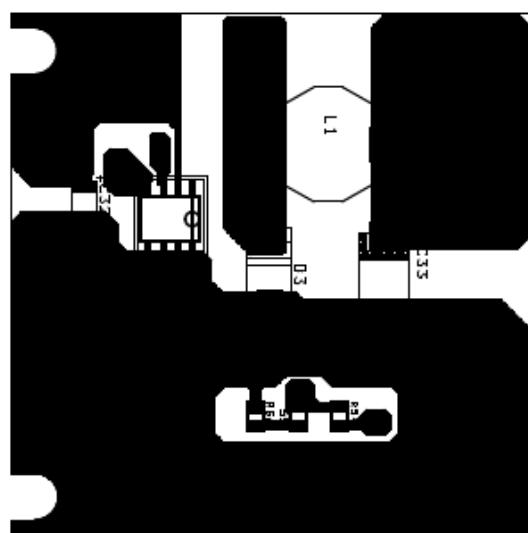
Obrázek A.2: Procesorová deska - BOTTOM vrstva



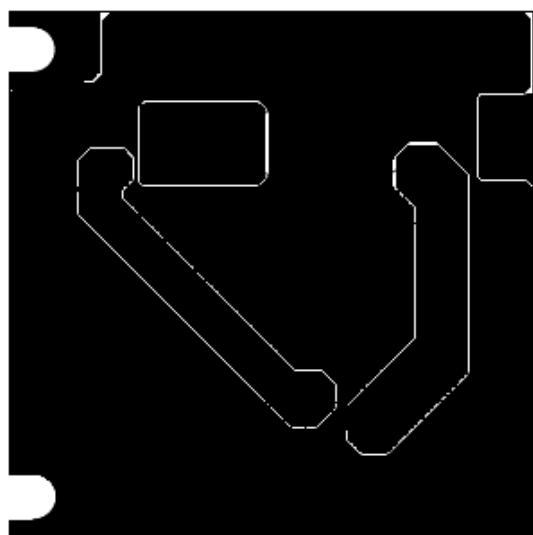
Obrázek A.3: Výkonová deska - TOP vrstva



Obrázek A.4: Výkonová deska - BOTTOM vrstva



Obrázek A.5: Zdrojová deska - TOP vrstva



Obrázek A.6: Zdrojová deska - BOTTOM vrstva

Příloha B

Obsah přiloženého CD

K této práci je přiloženo CD, na kterém jsou uloženy zdrojové kódy.

- DPS: - schéma, DPS a výrobní data pro Orcad
- Datasheet: - datasheets k součástkám použitých v měniči
- Code: - adresář obsahuje veškeré kódy programů pro měnič a hlavní řídící desku
 - PMSM
 - BLDC
 - Řídící elektronika
- DP: obsahuje elektronickou verzi DP