



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## PŘÍPRAVEK PRO ŘÍZENÍ MOTORU VRTACÍ SOUSTAVY MARSOVSKÉHO ROVERU

DC DRILLING MOTOR CONTROL KIT FOR MARS ROVER

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Ondřej Štafa

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Adam Ligocki, Ph.D.

BRNO 2025

# Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

**Student:** Ondřej Štafa

**ID:** 247480

**Ročník:** 3

**Akademický rok:** 2024/25

**NÁZEV TÉMATU:**

## Přípravek pro řízení motoru vrtací soustavy marsovského roveru

**POKYNY PRO VYPRACOVÁNÍ:**

Cílem práce je vytvořit demonstrační přípravek pro vrtací motor marsovského roveru pro soutěž European Rover Challenge (ERC).

1. Seznamte se s problematikou řízení DC motorů.
2. Navrhněte konstrukci a elektrické zapojení přípravku a fyzicky jej realizujte.
3. Navrhněte protokol pro komunikaci s nadřazeným systémem.
4. Vytvořte program pro mikroprocesor, který na základě příkazů z nadřazeného systému bude ovládat testovací soustavu.
5. Přípravek v praxi otestujte.

**DOPORUČENÁ LITERATURA:**

Making Embedded Systems. by Elecia White. Released October 2011. O'Reilly Media, Inc. ISBN: 9781449320584.

**Termín zadání:** 10.2.2025

**Termín odevzdání:** 28.5.2025

**Vedoucí práce:** Ing. Adam Ligocki, Ph.D.

**Ing. Miroslav Jirgl, Ph.D.**  
předseda rady studijního programu

**UPOZORNĚNÍ:**

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Tato práce se zabývá návrhem obvodu pro řízení DC motoru, který bude sloužit k pohonu vrtáku pro odběr vzorků regolitu. Celý vrták je součástí roveru vyvíjeného spolkem Brno Mars Rover za účelem účasti na mezinárodní soutěži European Rover Challenge.

Při řízení motoru bylo dosaženo dobrých výsledků použitím polovodičového H-můstku ovládaného pomocí PWM signálu. Důležitou úlohou je řízení točivého momentu a rychlosti motoru. Z toho důvodu zařízení disponuje snímačem proudu kotvy motoru. Celý obvod je ovládán mikrokontrolérem, který realizuje kaskádní řídicí obvod a zároveň komunikuje s nadřazeným mikrokontrolérem pomocí sběrnice I2C. Pro fyzickou realizaci zařízení byla navržena, osazena a otestována funkční deska plošných spojů.

Práce obsahuje identifikaci parametrů použitého motoru, popis použitých komponent, návrh desky plošných spojů a popis programu mikrokontroléru.

## **KLÍČOVÁ SLOVA**

DC motor, regolit, vrtání, řízení pohonů, kaskádní řízení, točivý moment, H-můstek, PWM, mikrokontrolér, měření proudu, I2C

## **ABSTRACT**

This thesis deals with the design of a circuit for DC motor control, which will be a part of a drilling mechanism used for gathering regolith samples. The whole drilling mechanism will be mounted on a rover developed by Brno Mars Rover team. The aim is to take part in the European Rover Challenge international competition.

As for controlling the motor, good results were achieved utilising a semiconductor based H-bridge, switched by a PWM signal. An important task is the control of the motor's torque and speed. To achieve this, the circuit features a sensor of armature current. The entire circuit is handled by a microcontroller, which implements a cascade control loop and communicates with its master via I2C. A working PCB has been designed, built and tested.

The thesis includes identification of the motor's parameters, description of the used components, the design process of the PCB and description of the microcontroller's software.

## **KEYWORDS**

DC motor, regolith, drilling, electric motor control, cascade control, torque, H-bridge, PWM, microcontroller, current measurement, I2C

ŠTAFKA, Ondřej. *Přípravek pro test motoru vrtací soustavy marsovského roveru*. Bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2025. Vedoucí práce: Ing. Adam Ligocki, Ph.D.

# Prohlášení autora o původnosti díla

**Jméno a příjmení autora:** Ondřej Štafa  
**VUT ID autora:** 247480  
**Typ práce:** Bakalářská práce  
**Akademický rok:** 2024/25  
**Téma závěrečné práce:** Přípravek pro test motoru vrtací soustavy marsovského roveru

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora\*

---

\* Autor podepisuje pouze v tištěné verzi.

## PODĚKOVÁNÍ

Chtěl bych poděkovat vedoucímu práce Ing. Adamovi Ligockimu, Ph.D. za poskytnuté nedocenitelné rady. Dále také patří dík Ing. Miloši Cihlářovi a Ing. Stanislavu Svědihovi za všechny nařezané hliníkové profily.

# Obsah

Úvod	11
<b>1 Teoretická část</b>	<b>12</b>
1.1 Stejnosměrné motory	12
1.1.1 Momentová charakteristika	12
1.1.2 Buzení stejnosměrných motorů	12
1.1.3 Bezkartáčové stejnosměrné motory	13
1.1.4 Řízení cize buzených stejnosměrných motorů	13
1.2 Řízení stejnosměrného motoru pomocí H-můstku	15
1.2.1 Pulzně šířková modulace	15
1.3 Obecný návrh zapojení	18
<b>2 Identifikace parametrů motoru</b>	<b>19</b>
2.1 Převodovka	19
2.2 Jmenovitý proud	19
2.3 Blocked rotor test	20
2.3.1 Odpor vinutí	20
2.3.2 Časová konstanta	20
2.4 Konstanta $C\Phi$	23
2.4.1 První způsob	23
2.4.2 Druhý způsob	25
2.4.3 Třetí způsob	25
<b>3 Použité komponenty</b>	<b>27</b>
3.1 Integrovaný obvod BTS7960	27
3.1.1 H-můstek IBT-2	27
3.1.2 Výstup IS	28
3.1.3 Slew rate	30
3.2 Mikrokontrolér	31
3.2.1 Základní vlastnosti mikrokontroléru Arduino	31
3.2.2 Napájení	32
3.2.3 Číslicové vstupy a výstupy	32
3.2.4 Analogové vstupy	32
3.3 Měření proudu kotvy	32
3.3.1 Integrovaný obvod ACS712	33
3.3.2 PWM a vzorkování	33
3.4 Teploměr DS18B20	35

<b>4</b>	<b>Komunikace</b>	<b>37</b>
4.1	Sběrnice I2C . . . . .	37
4.2	Formát zpráv . . . . .	38
4.2.1	Příchozí zpráva . . . . .	38
4.2.2	Odchozí zpráva . . . . .	38
<b>5</b>	<b>Fyzická realizace zařízení</b>	<b>40</b>
5.1	Testovací zapojení . . . . .	40
5.2	Deska plošných spojů . . . . .	41
5.2.1	Konektory . . . . .	41
5.2.2	IFX007T . . . . .	41
5.2.3	Ostatní součástky . . . . .	42
<b>6</b>	<b>Program mikrokontroléru</b>	<b>44</b>
6.1	Zpracování výstupu snímače proudu . . . . .	44
6.2	Řízení motoru . . . . .	46
6.2.1	Regulátory . . . . .	46
6.3	Bezpečnostní funkce . . . . .	47
	<b>Závěr</b>	<b>49</b>
	<b>Literatura</b>	<b>51</b>
	<b>Seznam symbolů a zkratk</b>	<b>52</b>
<b>A</b>	<b>Obsah elektronické přílohy</b>	<b>53</b>

# Seznam obrázků

1.1	Náhradní schéma stejnosměrného motoru buzeného permanentními magnety. . . . .	14
1.2	Regulační obvod používaný pro řízení pohonů. . . . .	15
1.3	Schéma H-můstku s připojeným zdrojem napětí a DC motorem. Samotný můstek je zvýrazněn červeně. [1] . . . . .	16
1.4	Změna polaroty napětí na zátěži dosažená sepnutím příslušných spínačů H-můstku. Červeně zvýrazněn tok proudu. [2] . . . . .	16
1.5	Obecné schéma zapojení. . . . .	18
2.1	Zapojení pro měření odporu vinutí Ohmovou metodou. . . . .	21
2.2	Jeden z měřených průběhů proudu. . . . .	22
2.3	Zapojení pro měření otáček motoru a proudu a napětí kotvy. . . . .	24
3.1	Schéma zapojení pro měření $k_{ILIS}$ . . . . .	28
3.2	Graf funkce 3.2 . . . . .	31
4.1	Příklad zapojení sběrnice I2C. [3] . . . . .	37
5.1	Schéma zapojení první verze obvodu. . . . .	40
5.2	Schéma zapojení DPS. . . . .	43
5.3	Osazená DPS s připojeným mikrokontrolérem. . . . .	43
6.1	Zjednodušený průběh smyčky programu mikrokontroléru. . . . .	44
6.2	Srovnání výstupu snímače proudu s a bez RC filtru. . . . .	45
6.3	Srovnání vstupu a výstupu číslicového filtru. . . . .	45
6.4	Schéma regulačního obvodu realizovaného v mikrokontroléru. . . . .	46
6.5	Průběh regulovaných veličin po zadání požadované hodnoty 2 ot./s. . . . .	48
6.6	Prezident republiky Petr Pavel testuje ranou, manuálně ovládanou verzi vrtáku. . . . .	50

# Seznam tabulek

2.1	Změřené hodnoty napětí na vinutí motoru a proudu zdroje, ze kterých je vypočítán odpor vinutí. . . . .	21
2.2	Časové konstanty pro skoky napětí o různých velikostech. . . . .	23
2.3	Změřené hodnoty napětí, proudu a otáček pro výpočet konstanty $C\Phi$ . . . . .	24
2.4	Naměřené hodnoty otáček a indukovaného napětí pro výpočet konstanty $C\Phi$ . . . . .	25
3.1	Tabulka závislosti $I_{IS}$ na $I_L$ . . . . .	29
3.2	Tabulka závislosti slew rate na $R_{SR}$ . . . . .	30
4.1	Struktura odchozí zprávy. . . . .	38
4.2	Význam stavového bajtu. . . . .	39

# Úvod

Cílem této práce je vytvořit zařízení pro ovládání DC motoru, které bude komunikovat přes sériovou sběrnici s nadřazeným mikrokontrolérem. Zařízení musí být schopno zejména udržovat konstantní rychlost motoru a chránit ho před přetížením.

Celé zařízení je součástí přípravku pro vrtání a odběr vzorků regolitu z hloubky až 30 cm pod povrchem. DC motor slouží pro pohon samotného vrtáku. Vývoji ostatních komponent vrtacího přípravku se věnují mojí kolegové Martin Kříž [4], Vilém Strachon [5] a Filip Šlíma [6] ve svých pracích.

Vrtací přípravek bude poté součástí roveru vyvíjeného studentským spolkem Brno Mars Rover. Tento rover se bude účastnit soutěže European Rover Challenge, kde je jedním z úkolů právě odběr regolitu ze simulovaného povrchu Marsu.

V kapitole 1 se věnuji teoretickému rozboru úlohy. Kapitola 2 se týká identifikace některých parametrů použitého motoru a kapitola 3 popisuje ostatní komponenty použité pro realizaci zařízení. V kapitole 4 představím protokol pro komunikaci s nadřazeným zařízením. Kapitola 5 se věnuje fyzické realizaci zařízení, včetně návrhu DPS. V poslední kapitole 6 je rozebrán program mikrokontroléru, který ovládá celé zařízení.

# 1 Teoretická část

## 1.1 Stejnosměrné motory

Stejnosměrný, neboli DC (direct current) motor [7], [8], je stroj schopný měnit stejnosměrné napětí na točivý pohyb. Byl prvním široce používaným typem motoru. DC motory se vyrábějí nebo vyráběly v mnoha velikostech pro různé aplikace - od hraček a aku-vrtaček po lokomotivy a průmyslové stroje. Zejména velké DC motory však byly v posledních desetiletích vytlačeny jinými typy motorů.

DC motor se skládá ze statoru a rotoru. Stator vytváří magnetické pole, které nemění svůj směr. Je tvořen buď permanentními magnety nebo cívkami. Rotor je pak tvořen cívkou, někdy označovanou jako kotva. Kotva vytváří točivé magnetické pole, protože polarita napětí na kotvě se periodicky mění. Toho je nejčastěji dosaženo pomocí komutátoru, který je zpravidla realizován jako dvojice kartáčů, které se třou o kovové kontakty.

Mezi hlavní přednosti DC motorů patří jednoduché řízení - tedy ovládání momentu, otáček a změna směru otáčení, neboli reverzace. Jednou z hlavních nevýhod je pak nízká životnost, která je způsobena zejména opotřebením komutátoru.

### 1.1.1 Momentová charakteristika

Momentová charakteristika pohonu vyjadřuje závislost jeho točivého momentu  $M$  na rychlosti otáčení  $\Omega$ . Z charakteristiky lze snadno vyčíst výkon pohonu  $P = M \cdot \Omega$ . DC motory dokáží pracovat ve všech čtyřech kvadrantech momentové charakteristiky.

V I. a III. kvadrantu mají otáčky a točivý moment stejný směr. Výkon pohonu je tedy kladný a pohon spotřebovává elektrickou energii, kterou přeměňuje na energii mechanickou.

Ve II. a IV. kvadrantu působí točivý moment pohonu proti směru otáčení. Výkon je v tomto případě záporný a pohon se chová jako generátor - mechanickou energii přeměňuje na elektrickou. K tomu stavu dochází pokud motorem otáčí cizí zdroj mechanické energie nebo pokud od motoru náhle odpojíme napětí - motorem poté otáčí setrvačnost mechanických součástí, které jsou připojeny k hřídeli.

### 1.1.2 Buzení stejnosměrných motorů

Pojmem buzení označujeme způsob napájení cívky statoru.

## **Cizí buzení**

Cívka statoru je napájena z jiného zdroje než cívka rotoru. Motory buzené permanentními magnety se chovají stejně jako motory cize buzené [7]. Dlouhou dobu bylo buzení permanentními magnety problematické kvůli nemožnosti výroby dostatečně silných magnetů. V posledních letech se však tento problém daří lépe překonat s použitím prvků vzácných zemin [8].

## **Další způsoby buzení**

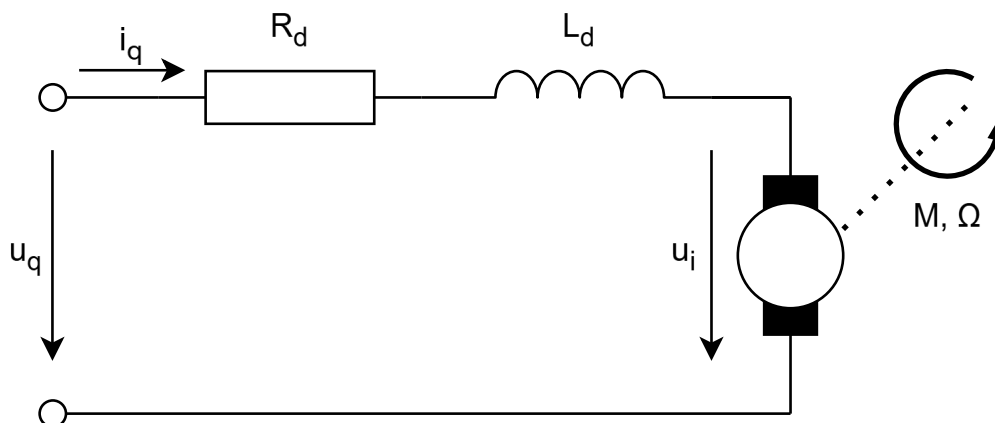
Cívky statoru a rotoru mohou být připojeny ke společnému zdroji, a to paralelně, sériově, nebo kombinací obou zapojení. Mluvíme pak o buzení sériovém, derivačním, nebo kompaundním. Tato zapojení jsou však využívána spíše u velkých motorů a tato práce se jimi nebude dále zabývat.

### **1.1.3 Bezkartáčové stejnosměrné motory**

Jedná se o jeden z nejnovějších typů stejnosměrných motorů. V angličtině jsou označovány jako brushless DC, zkráceně BLDC motory. Rotor BLDC motoru je tvořen permanentním magnetem. Řídicí jednotka sleduje natočení rotoru a na základě této informace elektronicky řídí proud cívkami statoru. Dochází tak k elektronické komutaci, což, jak název napovídá, umožňuje odstranění mechanického komutátoru s kartáči, který je nejporuchovější součástí ostatních typů stejnosměrných motorů.

### **1.1.4 Řízení cize buzených stejnosměrných motorů**

Cize buzený stejnosměrný motor můžeme popsat náhradním schématem na obrázku 1.1.



Obr. 1.1: Náhradní schéma stejnosměrného motoru buzeného permanentními magnety.

Zde jsou uvedeny významy jednotlivých výrazů použitých ve schématu 1.1:

$u_q$  ... napětí kotvy

$i_q$  ... proud kotvy

$u_i$  ... indukované napětí

$R_d$  ... odpor vinutí kotvy

$L_d$  ... indukčnost vinutí kotvy

Dále je motor popsán těmito veličinami:

$C$  ... konstanta motoru

$\Phi$  ... magnetický tok, v případě buzení permanentními magnety také konstantní

Chování motoru poté charakterizuje několik rovnic. Pro nás jsou podstatné dvě:

$$u_i = C \cdot \Phi \cdot \Omega \quad (1.1a)$$

$$M = C \cdot \Phi \cdot i_q \quad (1.1b)$$

### Řízení otáček

Z rovnic 1.1a a 1.1b vyplývá že, při nulovém zatížení motoru - tedy nulovém momentu - je proud kotvy  $i_q$  teoreticky taktéž nulový. Poté platí  $u_q = u_i$  a otáčky jsou tedy přímo úměrné napájecímu napětí. Hodí se podotknout, že kvůli tření mechanických součástí musí hřídel při otáčení vždy působit nenulovým momentem,  $i_q$  tedy při chodu motoru ve skutečnosti nebude přesně nulový.

S rostoucím momentem ale otáčky klesají, protože roste  $i_q$ , tím vzniká vyšší úbytek napětí na  $R_d$ , tedy klesá  $u_i$ .

Zpětnou vazbu - tedy informaci o rychlosti otáčení - můžeme získat poměrně snadno, například pomocí rotačního enkodéru. Řízení otáček ale není pro celé vrtací zařízení příliš důležité.

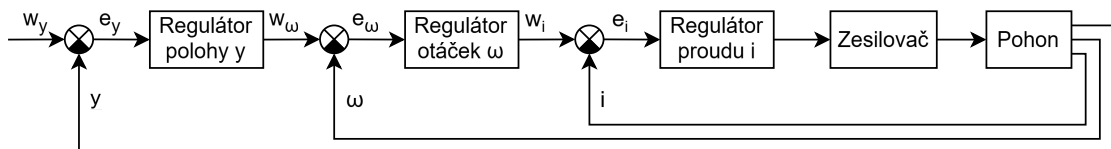
## Řízení momentu

Z rovnice 1.1b je patrné, že točivý moment je vždy přímo úměrný proudu kotvy  $i_q$  s konstantou úměrnosti  $C\Phi$ . Proud a moment můžeme samozřejmě zvětšit zvýšením napájecího napětí  $u_q$ .

Přímé měření momentu točícího se motoru je náročné a nepraktické. V praxi proto raději nejprve zjistíme konstantu úměrnosti  $C\Phi$  a poté jako zpětnou vazbu měříme pouze  $i_q$ .

## Regulační obvod

Pro řízení pohonů se často využívá rozvětvený regulační obvod s pomocnou regulovanou veličinou. Jde o kaskádní zapojení několik regulátorů, jak lze vidět na obrázku 1.2. V této úloze se budu věnovat pouze řízení otáček a točivého momentu, takže lze zanedbat vnější vrstvu regulátoru, která se stará o udržení konkrétní polohy.



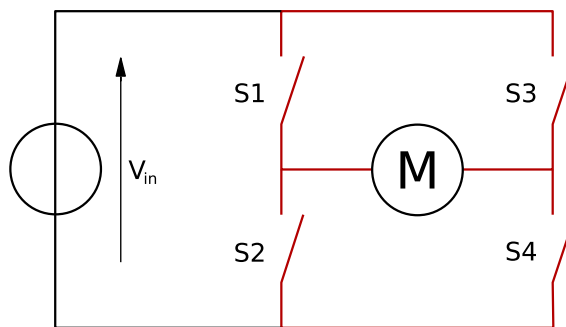
Obr. 1.2: Regulační obvod používaný pro řízení pohonů.

## 1.2 Řízení stejnosměrného motoru pomocí H-můstku

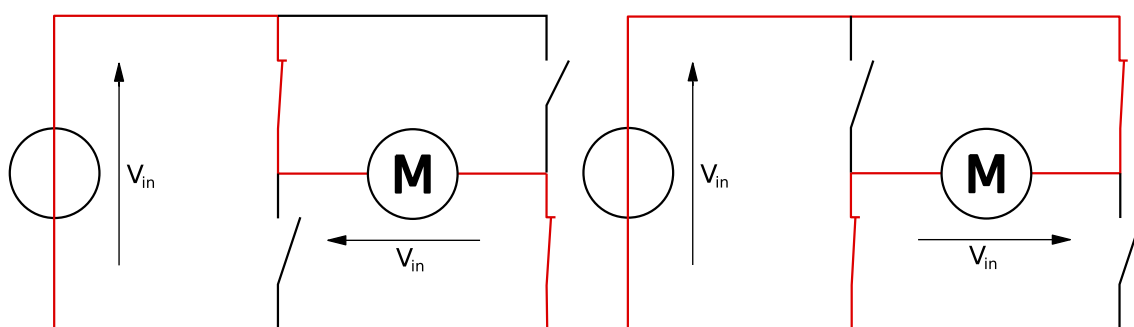
H-můstek sestává ze čtyř spínačů, jak je naznačeno na obrázku 1.3. Pokud sepneme spínače S1 a S4, můžeme přivést na motor napětí ze zdroje s kladnou polaritou. Sepnutím spínačů S3 a S2 polaritu změním. To je znázorněno na obrázku 1.4.

### 1.2.1 Pulzně šířková modulace

Pro změnu výkonu motoru je nutné měnit napětí přiváděné na jeho svorky. Napětí zdroje ale v praxi většinou není možné regulovat. Jsem proto nucen napětí na motoru měnit rychlým spínáním a rozpínáním některých spínačů můstku - tedy pulzně šířkovou modulací, zkráceně PWM.



Obr. 1.3: Schéma H-můstku s připojeným zdrojem napětí a DC motorem. Samotný můstek je zvýrazněn červeně. [1]



Obr. 1.4: Změna polarity napětí na zátěži dosažená sepnutím příslušných spínačů H-můstku. Červeně zvýrazněn tok proudu. [2]

### Střída

Výkon PWM signálu určuje střída, anglicky *duty cycle*. Často je vyjádřena v procentech a udává, jakou dobu z celé periody je spínač sepnut - rovnice 1.2. Efektivní hodnota výstupního napětí je pak přímo úměrná střídě a vstupnímu napětí. Platí rovnice 1.3

$$D = \frac{t}{T} \quad (1.2)$$

$D$  ... střída

$t$  ... doba, po kterou je spínač sepnut

$T$  ... perioda PWM

$$U_{ef} = D \cdot U_{max} \quad (1.3)$$

$U_{ef}$  ... efektivní hodnota napětí na výstupu můstku

$U_{max}$  ... napájecí napětí můstku

## Frekvence a strmost hran

Je zřejmé, že můstek realizovaný pomocí mechanických spínačů nebo relé nelze nijak sepnout částečně - spínač je vždy buď sepnutý nebo rozepnutý. Použití PWM v tomto případě ale není možné, protože by častým spínáním velmi rychle došlo k opotřebením spínačů. Musíme proto použít můstek realizovaný pomocí polovodičových spínačů, kterým časté spínání nevádí.

Polovodičové spínače, zejména tranzistory, dovolují částečné otevření. V tomto stavu na nich ale stále vzniká vysoký úbytek napětí, který vede k přeměně velké části spínaného výkonu na teplo. Polovodičové můstky je tedy taktéž výhodné spínat pomocí PWM.

K částečnému otevření přechodu bude stále docházet, ovšem vždy pouze na krátký moment, při náběžné a sestupné hraně vstupního obdélníkového signálu. Z energetického hlediska tedy chceme signál s co nejstrmějšími hranami a s co nejnižší frekvencí, aby spínače nemusely často měnit svůj stav. V opačném případě by totiž docházelo jak k plýtvání energií, tak k silnému zahřívání součástek. To by poté vyžadovalo implementaci velkých chladičů.

Zároveň ale musíme mít na paměti, že změny napětí na svorkách motoru vyvolají kolísání proudu kotvy a tedy i momentu motoru se stejnou frekvencí. Platí, že čím nižší je frekvence, tím silnější je střídavá složka proudu kotvy. Při velmi nízkých frekvencích může tento jev vyústit v nerovnoměrný chod motoru. Je empiricky ověřeno, že jedna perioda PWM by měla být alespoň čtyřikrát kratší než časová konstanta motoru, tedy:

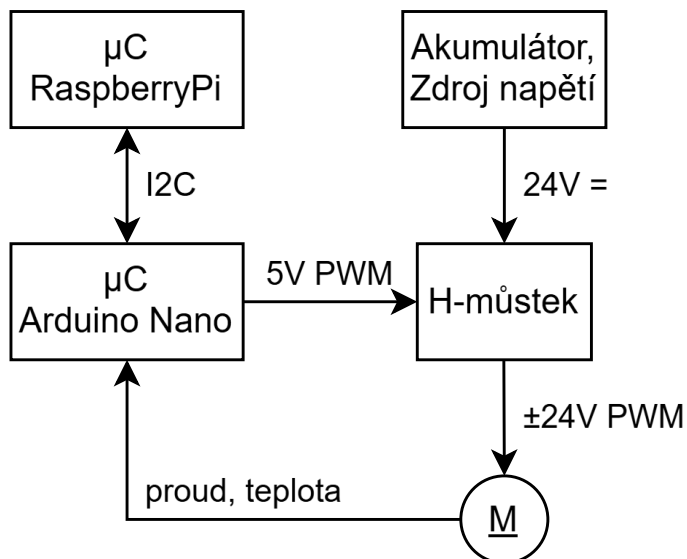
$$4 \cdot T_{PWM} < \tau_M \quad (1.4)$$

$T_{PWM}$  ... perioda PWM

$\tau_M$  ... časová konstanta motoru

Při zvolení dostatečně vysoké frekvence je střídavá složka proudu kotvy zanedbatelná.

Pokud navíc provedeme frekvenční analýzu reálného obdélníkového signálu, zjistíme, že čím jsou strmější hrany tohoto signálu, tím silněji jsou ve frekvenčním spektru zastoupeny vysoké frekvence. To v praxi znamená, že příliš strmě spínaný obvod by generoval vysokofrekvenční rušení, které by vysílal jak do rozvodné sítě, tak do okolního prostoru ve formě elektromagnetických vln. Tím se zhoršuje tzv. elektromagnetická kompatibilita, zkráceně EMC. V této úloze je třeba brát v potaz, že ze zdroje napětí pro motor vrtáku jsou napájeny i další komponenty vrtáku. Moje zařízení proto nesmí příliš ovlivňovat toto napájecí napětí.



Obr. 1.5: Obecné schéma zapojení.

Pokud tedy shrnu poslední čtyři odstavce - čím nižší frekvenci a čím strmější hrany PWM zvolíme, tím nižší budou ztráty v polovodičových spínačích H-můstku. Další faktory nás ale nutí zvolit frekvenci naopak co nejvyšší a zajistit pozvolné přechody mezi úrovněmi výstupního napětí. V následujících kapitolách se tedy mimo jiné pokusím identifikovat vhodnou frekvenci PWM.

### 1.3 Obecný návrh zapojení

Jak již bylo naznačeno, cílem této práce je řízení DC motoru. To je realizováno pomocí H-můstku. Jako zpětná vazba je použito měření proudu kotvou motoru a teplota vinutí motoru. Zpětnou vazbu vyhodnocuje mikrokontrolér, který na jejím základě řídí H-můstek a zároveň komunikuje s nadřazeným zařízením. Zapojení je naznačeno na obrázku 1.5.

V následujících kapitolách budou blíže popsány konkrétní komponenty, které byly pro realizaci zařízení vybrány.

## 2 Identifikace parametrů motoru

V tomto projektu je použita lehce upravená vrtačka Parkside PABS 20-Li G8. Disponuje stejnosměrným motorem napájeným akumulátorem o napětí 20 V. Datasheet udává maximálního točivý moment 45 N m. Vrtačka je zakončena sklíčidlem, ve kterém je upevněn vrták. Mezi sklíčidlem a motorem se nachází převodovka a mechanismus pro limitaci točivého momentu, který ale není využit.

Podrobnější technické specifikace bohužel nejsou k dispozici. Po rozebrání vrtačky jsem zjistil, že samotný motor nese označení *RS775 20V DC*. Informace přímo k tomuto motoru ale také nejsou dostupné, nezbývá tedy jiná možnost, než potřebné parametry motoru změřit.

Identifikace parametrů motoru je pouze malou částí celé práce, takže se při měření nezabývám nejistotami měření.

Motor přitahuje feromagnetické předměty, je tedy patrně buzený permanentními magnety, což je pro motory vrtaček typické. Budeme k němu tedy přistupovat jako k motoru cize buzenému.

### 2.1 Převodovka

Převodovka vrtačky je dvoustupňová. Dle datasheetu vrtačka na nižším převodu dosahuje 430 ot./s, na vyšším pak 1500 ot./s. S rostoucím mechanickým zatížením se ale otáčky samozřejmě snižují.

Rozebráním vrtačky bylo zjištěno, že převodovka je planetová. Planetová převodovka vždy snižuje úhlovou rychlost a zvyšuje moment. Samotný motor má tedy oproti výstupní hřídeli vrtačky vysokou rychlost a nízký moment. Vzhledem k tomu, že nepotřebuji dosáhnout vysokých otáček, budu využívat pouze nižší stupeň převodovky.

Pro zjištění převodového poměru bylo provedeno měření, kdy bylo otáčeno vstupní hřídelí převodovky, dokud se výstupní hřídel neotočila o jednu otáčku. Pro nižší převod tak byl zjištěn převodový poměr 1:57, pro vyšší převod je to 1:17.

### 2.2 Jmenovitý proud

Je velmi problematické, že neznám jmenovitý proud motoru. Nelze jej totiž nijak změřit, ale při jeho překročení může dojít k přehřátí vinutí a zničení motoru. Vzhledem k tomu, že ve svém zařízení budu používat jiný než původní způsob napájení, je třeba stanovit proud, který by neměl být překročen.

Na svorky motoru bylo přivedeno napětí 20 V, tedy stejné jako napětí původního akumulátoru. Následně byla výstupní hřídel vrtačky mechanicky zatížena. Se

zvyšující se zátěží dochází ke snižování otáček a vyššímu odběru proudu. Lze předpokládat, že motor není konstruován tak, aby dlouhodobě vydržel zkratový proud. Při zatížení, které vyústilo v přibližně jednu otáčku za sekundu, se odběr proudu pohyboval kolem 5 A. Vyšší zatížení už by vedlo k zastavení motoru. Jmenovitý proud tedy stanovím na 5 A.

## 2.3 Blocked rotor test

Pro zjištění časové konstanty a odporu vinutí motoru slouží tzv. *blocked rotor test*, tedy test se zablokovaným rotorem. Zablokováním rotoru zajistíme, že otáčky motoru budou vždy nulové. Z rovnice 1.1a tak vyplývá, že indukované napětí  $u_i$  bude také nulové. Z obrázku 1.1 si tak můžeme odmyslet kotvu motoru a zůstane nám pouze sériový RL článek.

Po přivedení skokové změny napětí  $u_q$  se bude proud  $i_q$  exponenciálně blížit určité ustálené hodnotě s určitou časovou konstantou. To je právě časová konstanta motoru zmiňovaná v části 1.2.1. Z ustálené hodnoty  $i_q$  a napětí  $u_q$  je poté možné vypočítat odpor vinutí  $R_d$ .

### 2.3.1 Odpor vinutí

Odpor vinutí byl změřen Ohmovou metodou. Bylo využito zapojení na obrázku 2.1. Výsledky jsou zaznamenány v tabulce 2.1. Odpor vinutí byl vypočítán podle Ohmova zákona, podle rovnice:

$$R_d = \frac{U_M}{I_Z} = \frac{0,085 \text{ V}}{0,317 \text{ A}} = 0,2681 \Omega \quad (2.1)$$

$R_d$  ... odpor vinutí

$U_M$  ... napětí na svorkách motoru

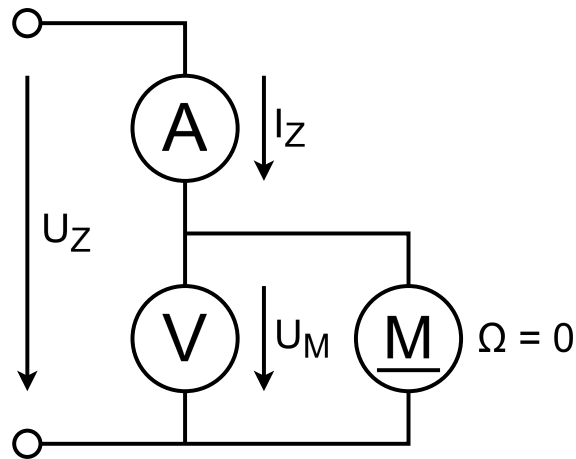
$I_Z$  ... proud ze zdroje

Průměrná hodnota odporu vinutí ze všech měření je 0,2608  $\Omega$ .

### 2.3.2 Časová konstanta

Pro změření časové konstanty je třeba osciloskopem zachytit časový průběh proudu. Osciloskopy ale dokáží měřit pouze napětí. Měření můžeme realizovat více způsoby.

Sériově k motoru můžeme připojit bočník, tedy rezistor, jehož odpor bude mnohem nižší než  $R_d$ , aby významně neovlivnil měření. Osciloskopem pak můžeme sledovat úbytek napětí na bočníku, který bude přímo úměrný  $i_q$ . Úbytek napětí na



Obr. 2.1: Zapojení pro měření odporu vinutí Ohmovou metodou.

$U_M$ [V]	$I_Z$ [A]	$R_d$ [ $\Omega$ ]
0.085	0.317	0.2681
0.127	0.473	0.2685
0.256	1.027	0.2493
0.279	1.071	0.2605
0.324	1.248	0.2596
0.425	1.646	0.2582
0.483	1.881	0.2568
0.537	2.101	0.2556
0.585	2.273	0.2574
0.638	2.485	0.2567
0.747	2.706	0.2761
0.797	2.899	0.2749
0.860	3.284	0.2619
0.893	3.605	0.2477

Tab. 2.1: Změřené hodnoty napětí na vinutí motoru a proudu zdroje, ze kterých je vypočítán odpor vinutí.

rezistoru ani není třeba přepočítávat na proud, protože nás zajímá pouze časová konstanta.

Alternativně můžeme použít proudovou sondu s napěťovým výstupem. V mém případě byla použita proudová sonda HP34134A, nastavená na citlivost 1 V/A. Citlivost je ale opět pro samotný výsledek irelevantní.

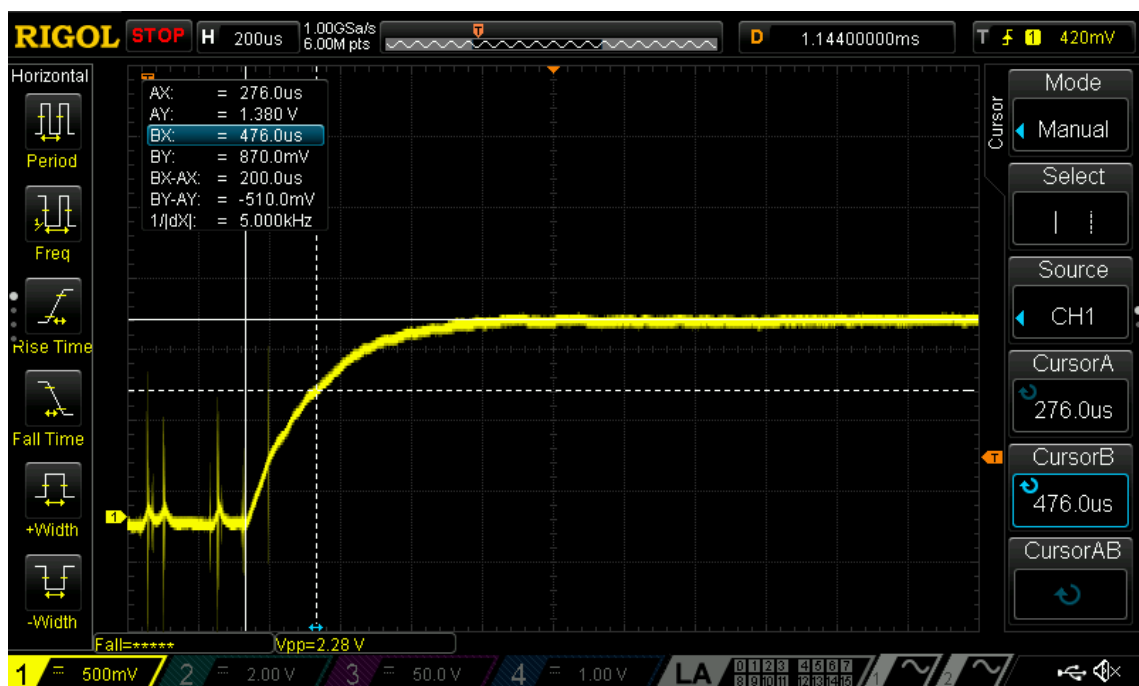
Bylo změřeno šest jednotlivých skoků napětí o různých amplitudách. Lepších

výsledků by mohlo být dosaženo použitím periodického obdélníkového signálu, ale standardní funkční generátor by pravděpodobně nebyl schopen dodat dostatečný výkon. Všechna měření jsou zapsána v tabulce 2.2. Jedno z těchto měření je pak zachyceno na obrázku 2.2. Průměrná hodnota časové konstanty je 233  $\mu$ s.

V části 1.2.1 bylo uvedeno, že jedna perioda PWM by měla být alespoň čtyřikrát kratší než časová konstanta motoru. Z rovnice 1.4 tedy vyjádříme

$$T_{PWM} < \frac{\tau_M}{4} = \frac{233 \mu\text{s}}{4} = 58 \mu\text{s} \quad (2.2)$$

$$f_{PWM} > \frac{1}{T_{PWM}} = \frac{1}{58 \mu\text{s}} = 17\,241 \text{ Hz} \quad (2.3)$$



Obr. 2.2: Jeden z měřených průběhů proudu.

Pro zajímavost můžeme vypočítat také indukčnost kotvy. Pro časovou konstantu  $\tau$  sériového RL článku platí

$$\tau = \frac{L}{R} \quad (2.4)$$

$$L = \tau \cdot R = 233 \cdot 10^{-6} \cdot 0,2608 = 60,77 \mu\text{H} \quad (2.5)$$

$U_{max}$ [V]	$\tau_M$ [ $\mu$ s]
2,07	240
1,78	280
1,62	230
1,48	250
1,42	200
1,38	200

Tab. 2.2: Časové konstanty pro skoky napětí o různých velikostech.

## 2.4 Konstanta $C\Phi$

Jak bylo řečeno v části 1.1.4, konstanta  $C\Phi$  udává vztah mezi proudem kotvy a točivým momentem. Můžeme ji určit několika způsoby, které budou popsány v následujících částech. Každá z těchto metod je zatížena jinou chybou měření, takže se jednotlivé výsledky mírně liší.

### 2.4.1 První způsob

Z rovnice 1.1a můžeme vyjádřit:

$$C\Phi = \frac{u_i}{\Omega} = \frac{u_q - i_q \cdot R_d}{\Omega} \quad (2.6)$$

Pokud tedy na svorky motoru přivedeme určitou hodnotu napětí, necháme motor otáčet a poté změříme odebíraný proud a rychlost otáčení, jsme schopni konstantu vypočítat. Měření je vhodné provést pro více různých hodnot napájecího napětí.

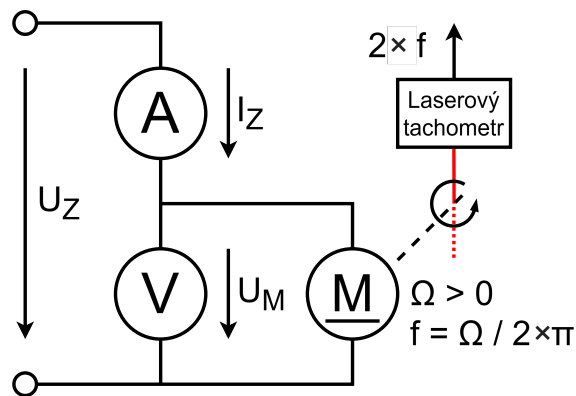
### Měření

Měření bylo provedeno podle schématu na obrázku 2.3. Naměřené hodnoty jsou zaznamenány v tabulce 2.3. Níže příklad výpočtu:

$$C\Phi = \frac{u_q - i_q \cdot R_d}{\Omega} = \frac{u_q - i_q \cdot R_d}{\frac{2 \cdot n}{2 \cdot 60} \cdot 2 \cdot \pi} = \frac{1,015 - 0,984 \cdot 0,2608}{\frac{1179}{2 \cdot 60} \cdot 2 \cdot \pi} = 0,012\,298 \text{ N m/A} \quad (2.7)$$

$n$  ... Rychlost otáčení v otáčkách za minutu

Pro měření otáček byl využit laserový tachometr MM-0360. Laserový tachometr vysílá laserový paprsek a snímá jeho odraz zpět od otáčející se osy. Pokud osa mění svoji odrazivost v závislosti na úhlu natočení, můžeme softwarově detekovat pulsy na výstupu tachometru. Frekvence těchto pulsů poté odpovídá určitému násobku



Obr. 2.3: Zapojení pro měření otáček motoru a proudu a napětí kotvy.

U [V]	I [A]	2 · n [ot./s]	$\Omega$ [rad/s]	$C\Phi$ [Nm/A]
1.015	0.984	1179	61.73	0.012285
1.418	1.073	1962	102.7	0.011079
1.915	1.105	2987	156.4	0.010402
2.365	1.142	3980	208.4	0.009920
2.940	1.403	5259	275.4	0.009348
3.402	1.510	6333	331.6	0.009072
3.865	1.575	7572	396.5	0.008713
4.895	1.705	9906	518.7	0.008580
5.895	1.774	12242	641.0	0.008475
6.850	1.825	14568	762.8	0.008356
7.944	1.890	17197	900.4	0.008275

Tab. 2.3: Změřené hodnoty napětí, proudu a otáček pro výpočet konstanty  $C\Phi$ .

frekvence otáčení osy. Pro zajištění spolehlivosti odrazů byla na osu mého motoru upevněna koncovka, jejíž tvar zapříčinil, že každá otáčka způsobila dva pulsy na výstupu snímače. Software použitý pro zpracování signálu ze snímače zobrazoval rychlost otáčení v otáčkách za sekundu. Tato veličina je zde značena písmenem  $n$ .

Můžeme si všimnout, že s rostoucími otáčkami se konstanta snižuje. Dále budu jako hodnotu konstanty uvažovat průměr z posledních pěti hodnot, tedy  $C\Phi = 0,008480 \text{ N m/A}$ .

Pokud nás zajímá točivý moment na výstupu převodovky, vynásobíme konstantu příslušným převodovým poměrem - v mém případě 57, takže  $C\Phi = 0,4834 \text{ N m/A}$ .

## 2.4.2 Druhý způsob

Opět z rovnice 1.1a vyjádříme:

$$C\Phi = \frac{u_i}{\Omega} \quad (2.8)$$

Nyní ovšem budeme motorem otáčet pomocí jiného motoru se známými otáčkami. Na svorkách měřeného motoru se indukuje napětí  $u_i$ . Pokud ho změříme voltmetrem s dostatečně vysokým vnitřním odporem, neprojeví se odpor vinutí  $R_d$ .

Jako zdroj otáček byla využita druhá identická vrtačka. Měření tak bylo možno provést pro dvě různé hodnoty úhlové rychlosti. Tato rychlost ale nebyla změřena a jsou uvažovány přibližné hodnoty udávané výrobcem. Toto měření tak slouží pouze k potvrzení výsledků předchozího měření.

Naměřené hodnoty jsou zaznamenány v tabulce 2.4 a lze vidět, že hodnota konstanty vyšla o 20 % nižší, než při předchozím měření. Měřený motor se totiž chová jako mechanická zátěž, otáčky napájeného motoru tak byly ve skutečnosti nižší, než udává výrobce pro chod bez zatížení. Měření ale potvrzuje, že konstanta byla určena alespoň řádově správně. Opět příklad výpočtu:

$$C\Phi = \frac{u_i}{\Omega} = \frac{u_i}{\frac{n}{60} \cdot 2 \cdot \pi} = \frac{0,298}{\frac{430}{60} \cdot 2 \cdot \pi} = 0,006\,618 \text{ N m/A} \quad (2.9)$$

$n[\text{ot./s}]$	$\Omega[\text{rad/s}]$	$u_i[\text{V}]$	$C\Phi[\text{N m/A}]$
430	45.03	0.298	0.006618
1500	157.08	1.030	0.006557

Tab. 2.4: Naměřené hodnoty otáček a indukovaného napětí pro výpočet konstanty  $C\Phi$ .

## 2.4.3 Třetí způsob

Z rovnice 1.1b vyjádříme:

$$C\Phi = \frac{M}{i_q} \quad (2.10)$$

Zde nám stačí měřit pouze odebíraný proud a výsledný moment. Měření bychom opět pro minimalizaci chyby provedli pro více hodnot proudu. Jak ale bylo zmíněno, měření momentu není úplně jednoduché. Za tímto účelem existují přístroje zvané dynamometry. Ty jsou ale většinou určeny pro větší motory. Uchycení mého motoru do dynamometru by tedy bylo obtížné.

Existuje ale i jiný způsob měření momentu. Pokud kolmo k hřídeli motoru připevníme rameno známé délky  $l$ , konec tohoto ramene bude působit silou  $F = M/l$ .

Sílu  $F$  pak můžeme měřit například pomocí siloměru nebo váhy. Konstantu poté vypočítáme podle následující rovnice:

$$C\Phi = \frac{M}{i_q} = \frac{F \cdot l}{i_q} = \frac{m \cdot g \cdot l}{i_q} \quad (2.11)$$

Dalo by se využít uspořádání, kdy na váze leží závaží známé hmotnosti  $m$ , které je nadlehčováno již zmíněným ramenem motoru. Na váze pak odečítáme nižší hmotnost  $m'$  a můžeme vyjádřit:

$$C\Phi = \frac{(m - m') \cdot g \cdot l}{i_q} \quad (2.12)$$

Tento způsob měření ale nebyl v praxi realizován.

## 3 Použité komponenty

### 3.1 Integrovaný obvod BTS7960

Půlmůstek BTS7960 tvoří dva tranzistory, které na základě vstupu IN připojují silový výstup OUT buď k napětí  $V_S$ , nebo k zemi GND. Pokud tedy připojíme ke každé svorce motoru jeden z těchto obvodů, můžeme tím realizovat H-můstek. Paralelně k tranzistorům jsou připojeny diody, které v případě náhlého odpojení napětí dovolí proudu v cívkách motoru protéct zpět do zdroje a zabrání tak proražení tranzistorů. Vstupem INH můžeme obvod uspat a výstup připojit k zemi. Pomocí vstupu SR můžeme nastavit rychlost přeběhu - anglicky *slew rate*. Výstupu IS bude věnována zvláštní pozornost níže.

BTS7960 také realizuje další velmi užitečné funkce jako ochranu před příliš nízkým nebo příliš vysokým napětím, před nadproudem a přehřátím a generování deadtime - tedy prodlevy, mezi okamžiky sepnutí jednotlivých spínačů pro prevenci zkratu.

Tato součástka již není ve výrobě a byla nahrazena integrovaným obvodem IFX007T od stejného výrobce, tedy Infineon Technologies. Všechny důležité parametry a funkce ale zůstaly zachovány.

#### 3.1.1 H-můstek IBT-2

Deska IBT-2 je dostupný a jednoduše ovladatelný H-můstek, který splňuje výkonové požadavky použitého motoru. Jeho hlavní součásti jsou dva půlmůstky BTS7960. Ty jsou na desce doplněny několika komponentami, zejména integrovaným obvodem 74HC244D, elektrolytickým kondenzátorem, chladičem, konektory a několika SMD rezistory.

Pravděpodobně neoficiální dokumentace poskytnutá k tomuto můstku je však bohužel velmi stručná a místy chybná. Některé detailnější informace o fungování můstku bylo tedy potřeba zjistit pomocí reverzního inženýrství. Elektrolytický kondenzátor je připojen mezi napájecí napětí a zem. Slouží k filtraci špiček, které jsou mimo jiné způsobovány již zmíněným PWM ovládáním. Dále lze dohledat, že integrovaný obvod 74HC244D je buffer. Na desce slouží k oddělení vstupů BTS7960 od vstupních pinů IBT-2.

Vstupy IN obvodů BTS7960 jsou v můstku IBT-2 připojeny k pull-down rezistorům a ke vstupům můstku LPWM/RPWM. Pokud je na oba PWM vstupy přivedena logická nula, svorky motoru jsou připojeny k zemi. Pokud přivedeme na jeden z PWM vstupů logickou jedničku, příslušná svorka bude připojena k napájecímu napětí, zatímco druhá svorka bude stále připojena k zemi. Motorem tedy

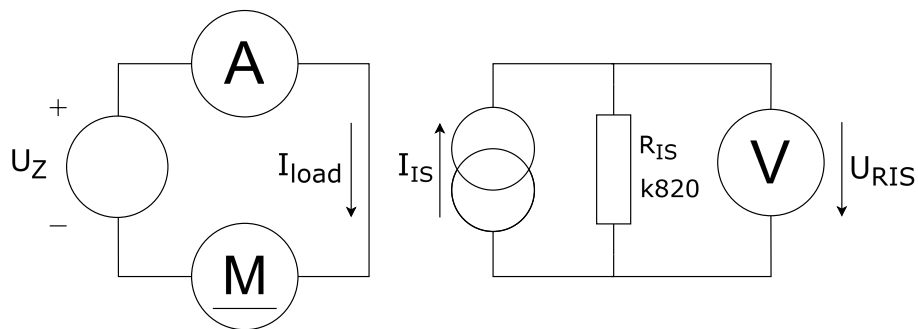
začne protékat proud. Pokud na oba PWM vstupy přivedeme logickou jedničku, budou obě svorky motoru připojeny k napájecímu napětí. Mezi nimi tak bude opět nulové napětí.

Výstupy IS obvodů BTS7960 jsou připojeny přímo ke konektorům můstku L\_IS a R\_IS. Vstupy SR jsou připojeny k zemi pomocí rezistorů o hodnotě 10 kΩ.

Bylo by možné použít dva samostatné obvody BTS7960 a ovládat je přímo z mikrokontroléru. Zejména obvod 74HC244D je pro mou aplikaci spíše nadbytečný, protože použitý mikrokontrolér bude dodávat dostatečný proud i správné napětí. Velkou výhodou IBT-2 jsou však již zmíněné konektory, chladič, a fakt, že je vše spojeno do kompaktního celku. V raných fázích vývoje je tedy použití IBT-2 opodstatněné.

### 3.1.2 Výstup IS

Na výstup IS je za normálního chodu připojen zdroj proudu  $I_{IS}$ , který je  $k_{ILIS}$ -krát nižší, než proud do zátěže  $I_L$ . Nadřazený mikrokontrolér disponuje pouze vestavěnými snímači napětí a nikoliv proudu.  $I_{IS}$  tedy budeme muset převést na napětí svedením tohoto proudu na zem přes vhodný rezistor  $R_{IS}$ . Získáme tak na tomto rezistoru napětí, které bude přímo úměrné proudu  $I_L$ . Typickou hodnotu  $k_{ILIS}$  můžeme vyčíst z datasheetu, kde zjistíme, že má poměrně vysokou toleranci - 3000 - 14000, s typickou hodnotou 8500. Provedl jsem měření  $k_{ILIS}$  pro různé hodnoty napětí na motoru  $U_Z$  podle schématu 3.1. Výsledky měření a vypočtené hodnoty jsou zapsány v tabulce 3.1. Můžeme si všimnout, že hodnota  $k_{ILIS}$  značně kolísá. Výstup IS tedy není vhodný pro přesné měření proudu motorem.



Obr. 3.1: Schéma zapojení pro měření  $k_{ILIS}$ .

Výstup IS má ale další funkci. Pokud je překročena bezpečná hodnota napětí, proudu nebo teploty, teče do výstupu IS proud  $I_{IS(lim)}$ , který má hodnotu kolem 4,5 mA. Pokud tedy opět IS spojíme vhodným rezistorem se zemí, můžeme měřením napětí na rezistoru detekovat poruchu.

$I_{load}[A]$	$U_{R_{IS}}[V]$	$I_{IS}[mA]$	$k_{ILIS}[-]$
0,63	0,070	0,0854	7380
0,96	0,122	0,1488	6452
2,02	0,261	0,3183	6346
2,91	0,370	0,4512	6449
3,24	0,435	0,5305	6108
3,97	0,526	0,6415	6189
3,94	0,530	0,6463	6096
3,64	0,558	0,6805	5349
2,44	0,323	0,3939	6194

Tab. 3.1: Tabulka závislosti  $I_{IS}$  na  $I_L$ .

Pokud bych chtěl výstup IS používat pro měření proudu i pro detekci poruchy, vyvstává v mém případě zajímavý problém. Proud  $I_{IS}$  za normálního stavu bude mnohem nižší, než  $I_{IS(lim)}$  při poruše. Napětí na připojeném rezistoru ale nesmí přesáhnout maximální napětí A/D převodníku, kterým budeme toto napětí snímat. Rezistor  $R_{IS}$  tedy musíme dimenzovat tak, aby při protékajícím  $I_{IS(lim)}$  vzniknul úbytek napětí o hodnotě horní hranice rozsahu A/D převodníku, typicky 5 V. Negativním důsledkem tohoto zapojení je, že pro měření  $I_L$  nám zůstane pouze malá část rozsahu A/D převodníku. To lze vyřešit tak, že použijeme vyšší hodnotu  $R_{IS}$ . Dejme tomu takovou, aby při nejvyšším předpokládaném  $I_L$  vzniknul úbytek napětí například 3 V. Pokud by nyní došlo k poruše a začal by protékat  $I_{IS(lim)}$ , překročili bychom rozsah A/D převodníku. Napětí na výstupu IS může podle datasheetu dosáhnout až 45 V. Připojili bychom proto paralelně k tomuto odporu Zenerovu diodu, která napětí na rezistoru omezí na 5 V. Získali bychom tak vyšší citlivost při měření  $I_L$  a zároveň zachovali dostatečnou mezeru pro rozpoznání poruchy.

Jak jsem ale již zmínil, měření proudu touto metodou je nepřesné. Výstup IS tedy budeme využívat pouze pro detekci přetížení nebo poruchy H-můstku. Hodnotu  $R_{IS}$  tedy můžeme zvolit maximálně:

$$R_{IS} = \frac{U_{ADmax}}{I_{IS(lim)}} = \frac{5}{0,0045} = 1111 \Omega \quad (3.1)$$

$U_{ADmax}$  ... maximální napětí A/D převodníku

V praxi jsem zvolil hodnotu 820  $\Omega$ .

### 3.1.3 Slew rate

Jak již bylo zmíněno, integrovaný obvod BTS7960 dovoluje nastavení slew rate, tedy strmost hran spínaného napětí. To je provedeno připojením rezistoru  $R_{SR}$  mezi vstup SR a zem. Slew rate poté závisí na velikosti odporu tohoto rezistoru. Datasheet uvádí následující hodnoty:

$R_{SR}$ [k $\Omega$ ]	$\Delta V_{OUT}$ [V/ $\mu$ s]
0	11
5,1	6
51	1,6

Tab. 3.2: Tabulka závislosti slew rate na  $R_{SR}$ .

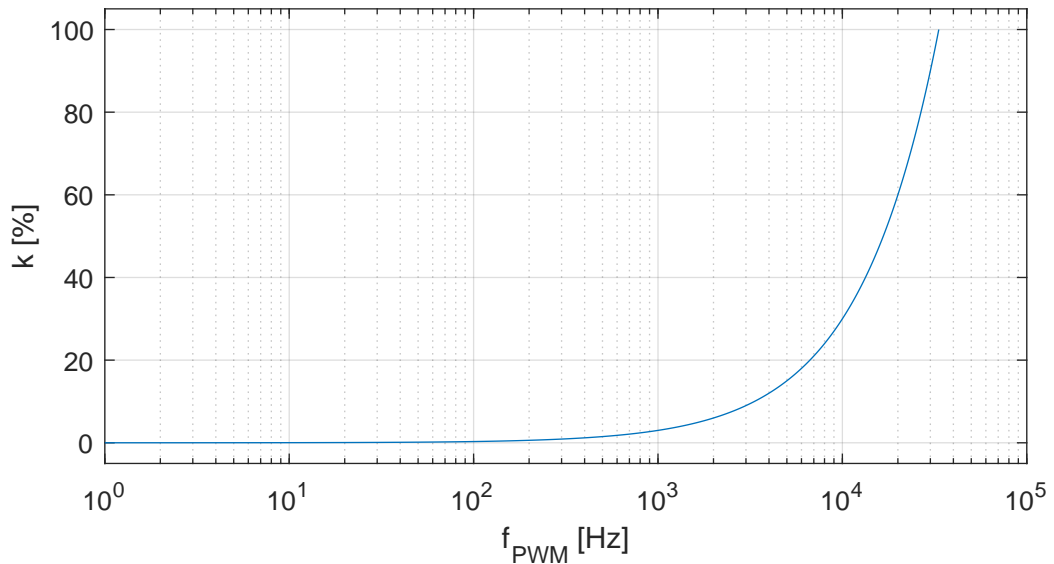
Pro zvolení vhodné frekvence PWM bude zajímavé zjistit vztah mezi touto frekvencí a dobou trvání přechodných dějů v poměru k době trvání jedné periody PWM. Jinými slovy - jakou část každé periody zaberou náběžné a sestupné hrany. Tento poměr označíme  $k$ . Měl by velmi hrubě vyjadřovat horní hranici energie vyplývané přepínáním spínače. Při  $k = 1$  platí, že spínač nikdy není plně otevřený ani zavřený. Při hodnotách  $k$  blízkých se k jedné se dá předpokládat, že na spínači budou vznikat vysoké ztráty. Na můstku IBT-2 je ke vstupům SR připojen rezistor o hodnotě 10 k $\Omega$ . Budeme tedy počítat se slew rate 1,6 V/ $\mu$ s. Jednoduše pak sestavíme následující rovnici, kterou budeme chápat jako funkci závislou na frekvenci PWM.

$$k = \frac{2 \cdot \frac{V_S}{\Delta V_{OUT}}}{T_{PWM}} = 2 \cdot \frac{V_S}{\Delta V_{OUT}} \cdot f_{PWM} = 2 \cdot \frac{24}{1,6 \cdot 10^6} \cdot f_{PWM} \quad (3.2)$$

$k$  ... poměr doby trvání nástupných a sestupných hran ku jedné periodě PWM  
 $\Delta V_{OUT}$  ... slew rate  
 $T_{PWM}$  ... perioda PWM  
 $f_{PWM}$  ... frekvence PWM

Graf této funkce je na obrázku 3.2. Můžeme si všimnout, že na frekvencích pod 100 Hz je hodnota  $k$  zanedbatelná. Kolem 1 kHz se stále pohybuje pod 5 %.

Datasheet zároveň uvádí maximální frekvenci PWM. Zde se datasheety k BTS7960 a IFX007T liší, oba ale uvádějí nejvyšší bezpečnou frekvenci pro nejnižší slew rate 25 kHz. K IFX007 už ale není uveden slew rate, ale pouze délky náběžných a sestupných hran, které vychází nižší, než pro BTS7960. Po nastavení kratších hran by tak s IFX007T bylo možno dosáhnout i vyšší spínací frekvence.



Obr. 3.2: Graf funkce 3.2

## 3.2 Mikrokontrolér

Pro řízení celého systému byl zvolen mikrokontrolér Arduino Nano v3.0. Mezi jeho přednosti patří především jednoduchost zapojení a programování. Platforma Arduino je zároveň vysoce rozšířená a velmi dobře zdokumentovaná. Nevýhodou pak může být poměrně nízká taktovací frekvence a nízká vzorkovací frekvence vestavěného A/D převodníku, což je jediný parametr, který mě do nějaké míry v této aplikaci limituje. Níže bude ukázáno, jak se tento problém dá vyřešit.

### 3.2.1 Základní vlastnosti mikrokontroléru Arduino

Verze Nano v3.0 je vybavena čipem ATmega328, taktovaným frekvencí 16 MHz. K dispozici je vlastní vývojové prostředí s vlastním jazykem založeným na C++. Arduino při každém spuštění jednou vykoná určitou posloupnost příkazů - setup. Poté vejde do nekonečné smyčky - loop. Jeden běh smyčky samozřejmě zabere mnoho hodinových cyklů procesoru a jeho délka trvání závisí na prováděném programu. Délka trvání jedné smyčky je důležitá z hlediska pozdější implementace regulátoru, protože se vlastně jedná o vzorkovací frekvenci. Její hodnotu nejlépe zjistíme experimentálně - buď necháme mikrokontrolér změřit čas, který uběhne mezi začátky po sobě jdoucích cyklů, nebo budeme měřit počet vykonaných cyklů za jednotku času.

### 3.2.2 Napájení

Na desce mikrokontroléru je regulátor, který převádí napětí ze vstupu  $V_{in}$  (7 V – 12 V) na napětí 5 V a 3,3 V, která najdeme na vývodech +5V a 3V3.

Alternativně je možné přivádět napětí 5 V z externího stabilizátoru na vývod +5V.

### 3.2.3 Číslicové vstupy a výstupy

Arduino Nano disponuje 22 vývody, které mohou být nastaveny buď jako vstup nebo výstup 0-5 V. Pro vstupy je pak k dispozici také vnitřní pull-up obvod.

Každý z těchto vývodů může být teoreticky použit pro generování PWM signálu, pokud budeme výstup v kódu manuálně přepínat s vhodnou prodlevou. Takový přístup je ale ze softwarového hlediska nepraktický a lze jím dosáhnout pouze omezené frekvence. Naštěstí je 6 vývodů vybaveno vlastním generátorem PWM, který v programu můžeme jednoduše ovládat funkcí `analogWrite()`. Argumenty funkce jsou číslo vývodu a duty cycle vyjádřený jako celé číslo od 0 do 255. Čtyři z těchto vývodů pracují na frekvenci 490 Hz, zbylé dva na frekvenci 980 Hz.

### 3.2.4 Analogové vstupy

Dále je k dispozici osm 10-bitových A/D převodníků, opět pro napětí 0-5 V. Můžeme vypočítat rozlišení:

$$R = \frac{U}{2^N - 1} = \frac{5}{2^{10} - 1} = 4,888 \text{ mV} \quad (3.3)$$

R ... rozlišení

U ... maximální napětí

N ... počet bitů A/D převodníku

Maximální vzorkovací frekvence těchto převodníků se pohybuje kolem 10 kHz. Mějme však na paměti, že pro dosažení této frekvence by musela být funkce pro čtení analogového vstupu volána v programu vícekrát za sebou, bez jakéhokoliv příkazu mezi nimi. Pokud tuto funkci budeme volat v každém cyklu pouze jednou, skutečná vzorkovací frekvence bude nižší, a bude se odvíjet od počtu cyklů vykonaných za sekundu.

## 3.3 Měření proudu kotvy

Jak bylo zmíněno v kapitole 1.1.4, úlohu regulace točivého momentu je výhodné převést na regulaci velikosti proudu kotvou motoru. Měření velikosti proudu lze

realizovat více způsoby.

Snímač proudu s digitálním výstupem může komunikovat buď paralelně, nebo sériově. Paralelní komunikace zabere příliš velký počet digitálních vstupů. Sériová komunikace je naopak příliš pomalá.

Lepší tedy bude převést proud na napětí a poté využít jeden z analogových napěťových vstupů mikrokontroléru. K převodu proudu na napětí by bylo možné použít bočník zapojený sériově s kotvou motoru, na kterém budeme měřit úbytek napětí. Proud ale může protékat kotvou v obou směrech a použitý A/D převodník nedokáže zpracovat záporné napětí. Bylo by tedy ještě potřeba mezi snímací rezistor a A/D převodník připojit analogový obvod, který by vstupní signál převáděl na jeho absolutní hodnotu. Výstup takového zapojení by ale pravděpodobně byl silně zatížen parazitními vlivy, zejména okolní teplotou.

### 3.3.1 Integrovaný obvod ACS712

Jednodušší alternativou je použít již hotový, komerčně dostupný snímač proudu s napěťovým výstupem. Jednou z možností je integrovaný obvod ACS712. Pracuje na principu Hallova jevu a je dostupný ve více variantách pro různé rozsahy proudu. Nejnižší rozsah je  $\pm 5$  A, což je pro tento projekt nedostačující. Byla proto využita varianta s rozsahem  $\pm 20$  A. Snímač tento proud převede na napětí o rozsahu 0,5 V – 4,5 V. Snímač má tedy citlivost 100 mV/A.

Šířka pásma snímače je 80 kHz. Dá se ale snížit vestavěným RC článkem. Uvnitř integrovaného obvodu se nachází rezistor o hodnotě 1700  $\Omega$ . Mezi vývod *FILTER* a zem poté můžeme připojit kondenzátor a vytvořit tím dolní propust. Toto zapojení by podle datasheetu mělo mít lepší šumové vlastnosti než externí filtr.

### 3.3.2 PWM a vzorkování

Jak již bylo uvedeno, motor bude ovládán PWM signálem. V části 1.2.1 bylo popsáno, jak frekvence tohoto signálu ovlivňuje chování motoru, a že je pro každý typ motoru žádoucí dosáhnout určité minimální frekvence. V části 2.3.2 pak byla tato frekvence stanovena na přibližně 17 kHz. Použitý mikrokontrolér ale dokáže generovat PWM signál s frekvencí pouze 490 Hz nebo 980 Hz. Experimentálně bylo zjištěno, že mechanické vlastnosti použitého motoru nejsou takto nízkou frekvencí negativně ovlivněny.

Proud kotvy motoru  $i_q$  ale silně kolísá. Je tedy velmi důležité se ujistit, jestli při vzorkování výstupu snímače proudu nedojde k chybě vzorkování. Při nejrychlejším možném vzorkování a nejnižší možné frekvenci PWM můžeme dosáhnout počtu vzorků na jednu periodu PWM

$$n = \frac{f_{vz}}{f_{PWM}} = \frac{10000}{490} = 20,4 \quad (3.4)$$

$n$  ... počet vzorků na jednu periodu PWM  
 $f_{vz}$  ... vzorkovací frekvence  
 $f_{PWM}$  ... frekvence PWM

Takový počet vzorků by měl stačit pro věrnou rekonstrukci měřeného signálu. Nižší počet vzorků by však již byl problematický. Je zde ale stále určité riziko úplného vynechání krátkých peaků. Pro dosažení tohoto počtu vzorků je také potřeba v každé smyčce mikrokontroléru provést 20 měření. V každém smyčce tak mikrokontrolér měřením proudu stráví tuto dobu:

$$t = N \cdot \frac{1}{f_{vz}} = 20 \cdot \frac{1}{10000} = 2 \text{ ms} \quad (3.5)$$

$t$  ... čas potřebný k měření  
 $N$  ... počet měření  
 $f_{vz}$  ... vzorkovací frekvence

Alternativou k tomuto přístupu je analogově odfiltrovat střídavou složku napětového signálu na výstupu snímače proudu. Zůstane nám tak pouze signál úměrný stejnosměrné složce  $i_q$ .

Filtr musí ideálně zajistit, aby výsledné zvlnění nepřesáhlo rozlišovací schopnost A/D převodníku. Pro tento účel si vystačíme i s jednoduchým RC článkem. Nejnižší harmonická složka, která se bude v signálu vyskytovat, bude patrně samotná frekvence  $f_{PWM}$ . Časovou konstantu RC článku musíme určit tak, aby byla tato složka dostatečně utlumená. Všechny ostatní složky budou mít vyšší frekvenci a budou tedy utlumeny více. Mějme však na paměti, že zvýšením časové konstanty zároveň prodloužíme dobu, než se na výstupu filtru projeví změna stejnosměrné složky  $I_M$ . Abychom si vystačili s nižší časovou konstantou, použijeme vývody mikrokontroléru s vyšší  $f_{PWM}$  - 980 Hz. V podkapitole 3.1.3 bylo zjištěno, že použitý H-můstek by měl takovou frekvenci pohodlně zvládnout.

Jak již bylo zmíněno, snímač ACS712 disponuje vestavěným filtrem. Pokud ale realizujeme vlastní filtr, můžeme číst jak filtrovaný, tak nefiltrovaný signál snímače. To nám dovolí rychleji zareagovat na případnou proudovou špičku, která se v nefiltrovaném signálu projeví dříve.

AD převodník mikrokontroléru ale způsobí další šum, v mém případě přibližně na úrovni nejnižších dvou bitů. V programu mikrokontroléru tak bude navíc realizováno průměrování naměřených hodnot proudu.

## Výpočet parametrů RC článku

Bylo změřeno, že při spínání motoru se střídou 50 % se na výstupu snímače proudu objevuje napěťový signál s mezivrcholovým napětím přibližně  $U_{pp} = 0,8 \text{ V}$ . V kapitole 3.2.4 bylo určeno, že chceme toto napětí snížit na hodnotu 5 mV. Potřebujeme tedy dosáhnout zesílení:

$$A = \frac{0,005}{0,8} = 0,00625 \approx -44 \text{ dB} \quad (3.6)$$

Pro zesílení RC článku platí následující rovnice:

$$A = \frac{1}{\omega \cdot \tau} \quad (3.7)$$

$$\tau = R \cdot C \quad (3.8)$$

Odtud vyjádříme časovou konstantu a dosadíme známé hodnoty.

$$\tau = \frac{1}{\omega \cdot A} = \frac{1}{2 \cdot \pi \cdot f_{PWM} \cdot A} = \frac{1}{2 \cdot \pi \cdot 980 \cdot 0,00625} \approx 0,02598 \text{ s} \quad (3.9)$$

Se známou časovou konstantou stále zbývá zvolit konkrétní hodnoty kapacity a odporu. Příliš nízká kapacita by znamenala vysoký šum na výstupu článku. Příliš nízký odpor by zase na vstupu článku vyžadoval vysoký proud, který snímač nebude schopen dodat.

Z datasheetu snímače proudu můžeme vyčíst, že maximální proud na výstupu je 3 mA. Maximální napětí na výstupu je 4,5 V. Z toho můžeme jednoduše vypočítat, že rezistor RC článku musí mít hodnotu minimálně 1500  $\Omega$ .

Co se kondenzátoru týče, byla zvolena hodnota 100 nF, což by mělo být dostatečné pro potlačení šumu. Můžeme poté vypočítat odpor:

$$R = \frac{\tau}{c} = \frac{0,02598}{100 \cdot 10^{-9}} = 259800 \Omega \quad (3.10)$$

V praxi byl tedy zvolen rezistor 270K. Při čtení výstupu snímače proudu mikrokontrolérem bylo zvlnění signálu stále viditelné, hodnota odporu byla proto zvýšena na 330K.

## 3.4 Teploměr DS18B20

DS18B20 je digitální snímač teploty. Pracuje v rozsahu  $-55^\circ\text{C} - 125^\circ\text{C}$  s programovatelným rozlišením 9 - 12 bitů.

Výstup je zasílán prostřednictvím sériové sběrnice 1-Wire. Jak název napovídá, tato sběrnice je specifická tím, že komunikace probíhá pouze pomocí jednoho vodiče.

Je ale zapotřebí zajistit také společnou zem. Pro správné fungování musí být ke sběrnici připojen pull-up rezistor o hodnotě  $4700\ \Omega$ .

Každý teploměr má vlastní 64-bitovou adresu. Je tedy možné k jedné sběrnici připojit více teploměrů a komunikovat s každým zvlášť.

Napájení může být zajištěno parazitně, z vodiče používaného pro komunikaci. Teploměr tak může být k mikrokontroléru připojen pouze dvěma vodiči - jeden pro zem, druhý pro komunikaci a napájení. Je ale možné pro napájení využít samostatný vodič, čímž se jejich celkový počet zvýší na tři.

Pro mikrokontroléry Arduino je k dispozici hotová knihovna umožňující jednoduché vyčítání měřených hodnot.

Pro měření teploty vinutí motoru je ideální, když je teploměr umístěn co nejbližší tomuto vinutí, ideálně přímo uvnitř něj. Použitý motor bohužel vestavěným teploměrem nedisponuje, takže je můj teploměr umístěn zvenku, na kryt motoru. Experimentálně bylo zjištěno, že časová konstanta přenosu tepla z vinutí motoru do teploměru se pohybuje v rámci vyšších jednotek sekund. Takto vysoká časová konstanta není dostatečná pro spolehlivou ochranu motoru před přehřátím, ale přinejmenším slouží pro demonstraci funkčnosti tohoto zapojení.

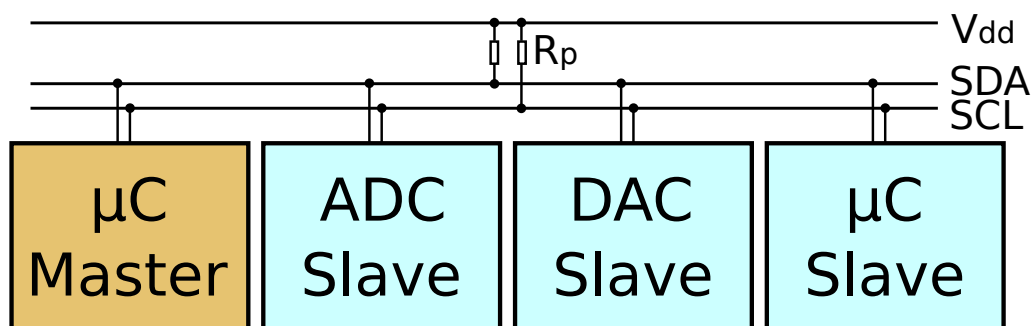
## 4 Komunikace

Dalším z požadavků kladených na mé zařízení je schopnost komunikovat s nadřazeným mikrokontrolérem. Specificky se jedná o příjem požadavků na směr a rychlost otáčení motoru a odesílání případných chybových zpráv a informací o nynějším stavu. Objem přenášených zpráv se tedy bude pohybovat v jednotkách bajtů za sekundu. Nadřazený mikrokontrolér potřebuje po stejné sběrnici komunikovat s dalšími zařízeními a všechna tato zařízení jsou fyzicky umístěna blízko u sebe, případně i na společné desce.

### 4.1 Sběrnice I2C

Vhodnou sběrnicí pro výše uvedené podmínky je I2C. Příklad zapojení je na obrázku 4.1. Sběrnice se tedy skládá ze dvou vodičů - SCL pro obdélníkový hodinový signál a SDA pro sériová data. Tyto vodiče jsou připojeny k napájecímu napětí pomocí pull-up rezistorů a jednotlivá zařízení je při komunikaci připojují k zemi. Ke sběrnici může být obecně připojeno více zařízení typu master i slave. Každé zařízení ale musí mít unikátní adresu. Adresy jsou typicky sedmibitové, což limituje počet připojených zařízení na 128. Komunikaci zahajuje vždy master, který na určitou adresu buď zasílá data nebo si od ní naopak vyžádá zprávu. Po případném odeslání této zprávy je sběrnice opět uvolněna pro další komunikaci. Komunikace tedy může v jeden moment probíhat pouze mezi dvěma zařízeními a jedním směrem.

Data jsou přenášena po bajtech. Pokud tedy chceme přenášet vícebajtové datové typy, např. float, je před odesláním třeba k proměnné přistupovat jako k poli bajtů. Po přijetí tohoto pole druhým mikrokontrolérem se přijaté pole opět interpretuje jako původní datový typ.



Obr. 4.1: Příklad zapojení sběrnice I2C. [3]

V tomto případě bude ke sběrnici připojen nadřazený mikrokontrolér jako jediný master a další tři mikrokontroléry jako slave. Jedním z nich je můj mikrokontrolér

Arduino Nano. Ten disponuje hardwarovým modulem a softwarovými knihovnami pro velmi snadnou práci s I2C sběrnici. Pro SDA je určen pin A4, pro SCL A5.

## 4.2 Formát zpráv

### 4.2.1 Příchozí zpráva

Příchozí zprávu může tvořit jediné znaménkové číslo, tedy *signed int* nebo *float*, přičemž velikost čísla určuje požadovanou rychlost motoru v otáčkách za sekundu a znaménko určuje směr otáčení. Původně byl využit *float* pro svoji verzatilitu. Rychlost ale není třeba ovládat příliš přesně, takže jsem pro zrychlení komunikace začal využívat *int8\_t*, přičemž je platný pouze rozsah  $-100 \dots 100$ , který odpovídá  $-3 \text{ Hz} \dots 3 \text{ Hz}$ . Požadovaná hodnota tedy musí být před odesláním vydělena číslem 0,03 a přijaté číslo je poté opět číslem 0,03 vynásobeno.

### 4.2.2 Odchozí zpráva

Struktura odchozí zprávy je znázorněna v tabulce 4.1. Hodnoty rychlosti a točivého momentu jsou přepočítány stejně jako v předchozím případě. Význam stavového bajtu je v tabulce 4.2.

Pořadí bajtu	Datový typ	Význam
0	uint8_t	Stav
1	int8_t	Rychlost
2	int8_t	Točivý moment
3	uint8_t	Teplota motoru

Tab. 4.1: Struktura odchozí zprávy.

Bit	Význam
0	00...motor je vypnutý 01...motor se točí
1	10...motor se zaseknul 11...nevyužito
2	0...H-můstek je OK 1...chyba H-můstku
3	nevyužito
4	
5	
6	
7	

Tab. 4.2: Význam stavového bajtu.

# 5 Fyzická realizace zařízení

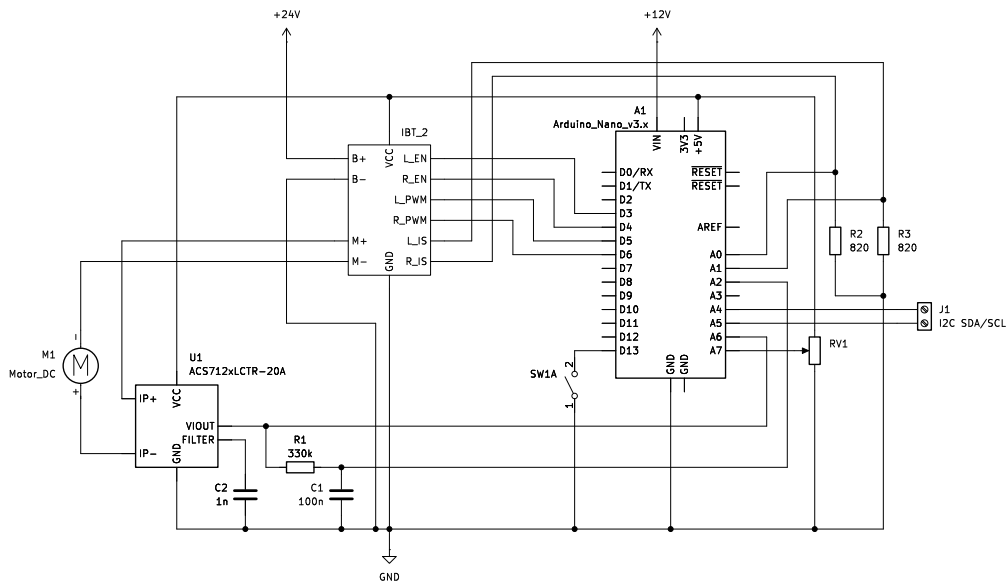
Nyní tedy máme všechny potřebné komponenty a znalosti k fyzické realizaci obvodu. Zapojení prošlo více vývojovými etapami, které jsou popsány níže.

## 5.1 Testovací zapojení

Pro prvotní testování funkčnosti všech komponent bylo realizováno zapojení na nepájivém kontaktním poli podle schématu na obrázku 5.1.

RC článek popsany v podkapitole 3.3.2 je realizován součástkami  $R1$  a  $C1$ . Vnitřní filtr snímače proudu díky nízké kapacitě kondenzátoru  $C2$  potlačí pouze frekvence nad 100 kHz.

Obvod je dále doplněn o potenciometr  $RV1$  a spínač  $SW1A$ . Ty slouží k nastavení žádaného točivého momentu a směru otáčení motoru přímo v mém obvodu, takže se při testování této funkcionality nemusím zabývat komunikací s nadřazeným mikrokontrolérem.



Obr. 5.1: Schéma zapojení první verze obvodu.

## 5.2 Deska plošných spojů

Provedení na nepájivém kontaktním poli se ukázalo jako velmi vhodné pro experimentaci. Při propojování s ostatními částmi elektronického vybavení vrtáku se však projevila nízká spolehlivost plynoucí z absence pevných, pájených spojů. Bylo proto nutné zapojení realizovat na desce plošných spojů. Toto provedení je zároveň mnohem kompaktnější. Deska byla navržena v programu KiCad 8 a celý projekt je dostupný v příloze. Nové schéma je na obrázku 5.2 a fotografie osazené desky je poté na obrázku 5.3.

Připojení celého H-můstku IBT-2 k jiné desce by bylo velmi nepraktické. Místo toho jsou použity přímo integrované obvody IFX007T ( $U2$  a  $U3$ ) pro realizace H-můstku na jedné desce spolu s ostatními součástkami. Zároveň došlo k odstranění RV1 a SW1A.

### 5.2.1 Konektory

Mikrokontrolér Arduino Nano není k desce napevno připájen, ale zasouvá se do připájených lišt s konektory typu DuPont. Pro připojení k I2C sběrnici slouží konektor *JST1*. Došlo také k přidání konektoru pro připojení teploměru, *JST2*.

Na konektor *Dean1* je nutno přivést napětí 24 V pro napájení H-můstku. Ke konektoru *Dean2* se poté připojují svorky motoru. Konektor XT60 vyžaduje napětí 12 V pro napájení mikrokontroléru.

### 5.2.2 IFX007T

Datasheet k tomuto integrovanému obvodu nabízí několik doporučení k návrhu.

Co nejbližší k napájecím vývodům mají být přítomny kondenzátory pro vykrytí špiček ve spotřebě. K tomu slouží  $C4$  a  $C5$ . Pro vykrytí větších výkyvů je připojen společný elektrolytický kondenzátor  $C3$ .

Kvůli indukovaným proudům by se na logických vstupech IN a INH mohlo objevit vysoké napětí. Tomu se předchází sériovým připojením odporů  $R7$ ,  $R8$ ,  $R9$ ,  $R10$ .

Slew rate je nastaven hodnotami odporů  $R2$  a  $R3$ .  $C6$  a  $C7$  slouží k odstranění případného šumu.

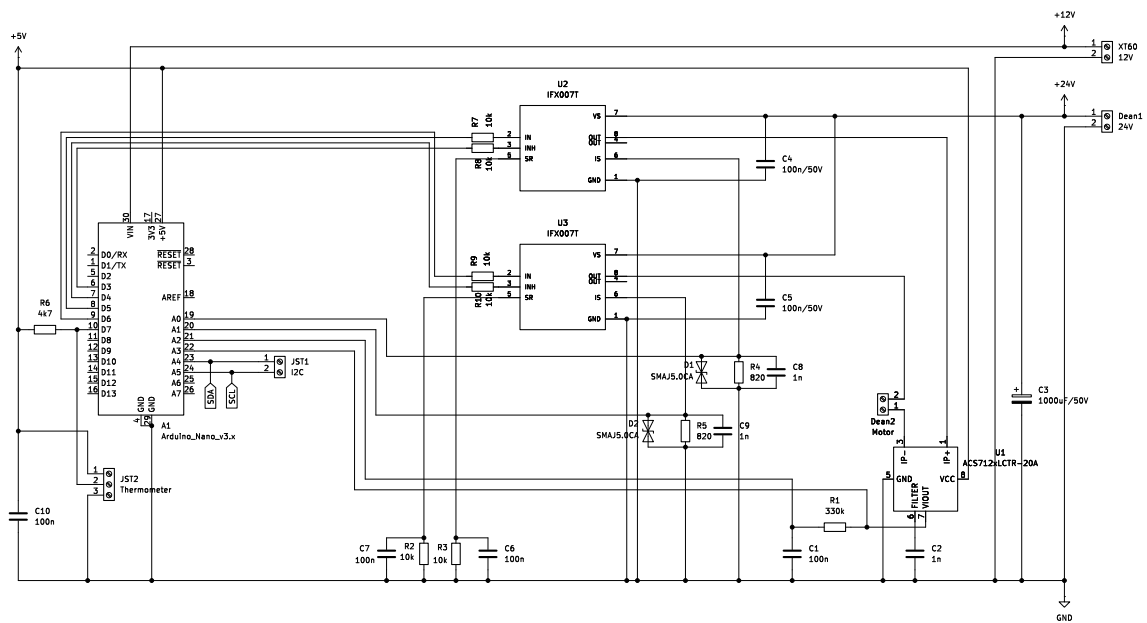
Pro převod proudu z výstupů IS na napětí slouží odpory  $R4$  a  $R5$ . Toto napětí by při zvolené hodnotě odporů nemělo překročit 5 V. Pokud by ale došlo k chybě a na těchto vývodech se objevilo vyšší napětí, mohlo by dojít ke zničení vstupů mikrokontroléru. Z toho důvodu jsou paralelně připojeny transily D1 a D2, které se při vyšším napětí otevřou. Kondenzátory  $C8$  a  $C9$  zabraňují šumu.

### 5.2.3 Ostatní součástky

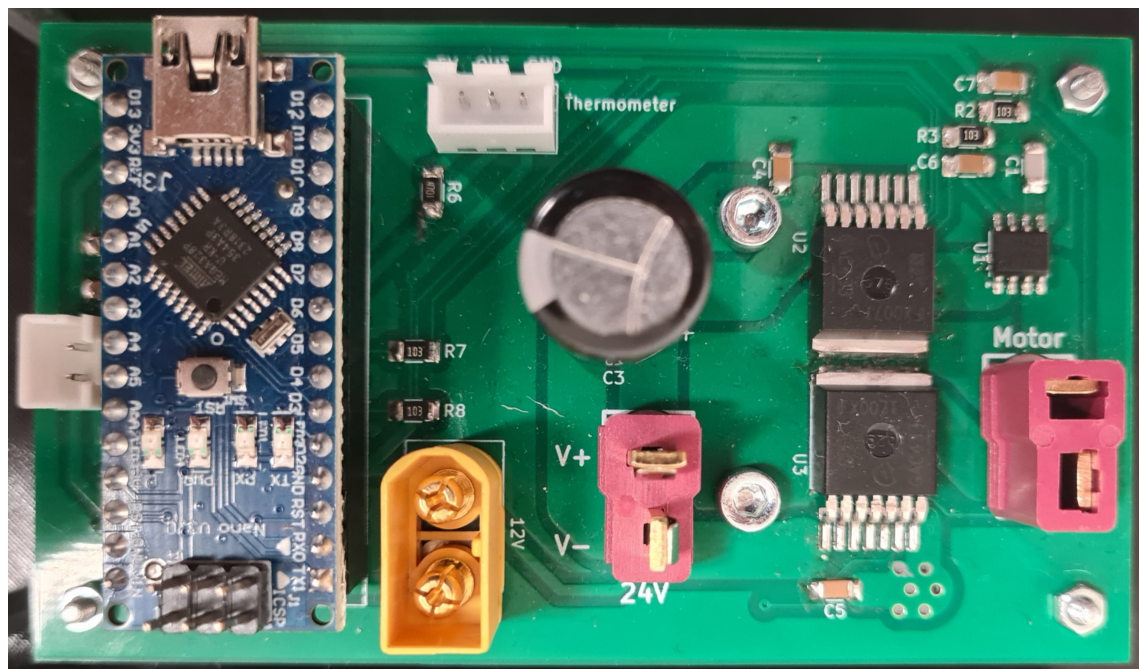
Součástky  $R1$ ,  $C1$ ,  $C2$  slouží k filtraci výstupu ze snímače proudu, stejně jako v předchozím návrhu.

Kondenzátor  $C10$  stabilizuje výstupní napětí zdroje 5 V.

$R6$  je pull-up rezistor pro výstup teploměru.



Obr. 5.2: Schéma zapojení DPS.

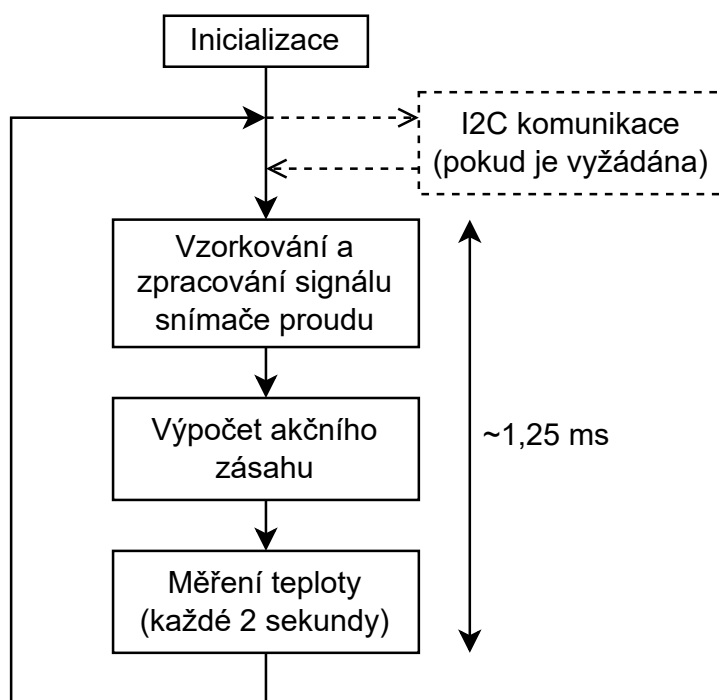


Obr. 5.3: Osazená DPS s připojeným mikrokontrolérem.

## 6 Program mikrokontroléru

Program mikrokontroléru byl vyvíjen v prostředí Arduino IDE. Prostředí využívá vlastní jazyk, který je ale pouze rozšířením C++. Kód je zálohován a verzován na Githubu a lze ho také najít v elektronické příloze.

Smyčka programu se nevykonává v pevných intervalech - při vzorkování vstupu a při regulaci je uplatněna proměnlivá vzorkovací perioda. Ta se pohybuje kolem 1,25 ms. Na obrázku 6.1 je znázorněn průběh hlavních funkcí programu.



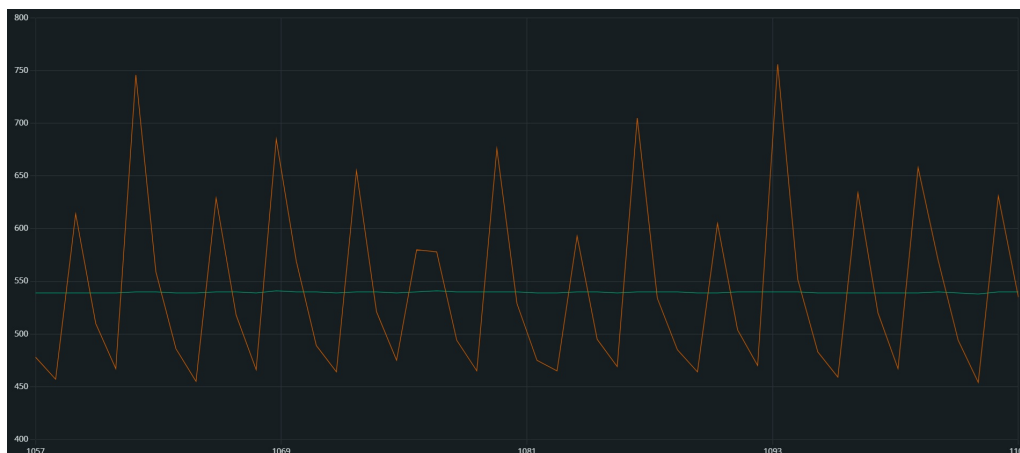
Obr. 6.1: Zjednodušený průběh smyčky programu mikrokontroléru.

### 6.1 Zpracování výstupu snímače proudu

V části 3.3 byl popsán hardware použitý ke snímání proudu kotvy.

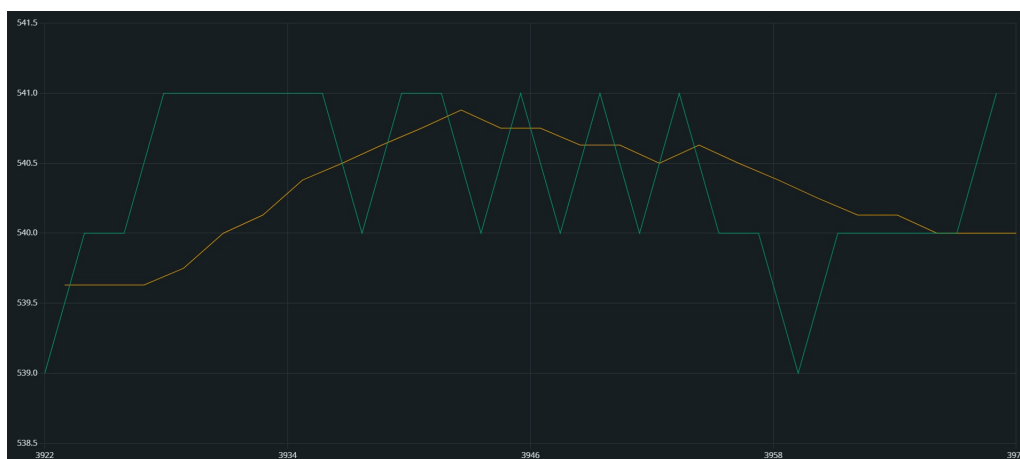
Na obrázku 6.2 je srovnán vzorkovaný výstup snímače proudu. Na ose y je výstup A/D převodníku, na ose x pak pořadí vzorku. Vzorkovací perioda byla při tomto měření 18 ms. Zeleně je znázorněn výstup snímače proudu filtrovaný RC filtrem.

Červeně je znázorněn výstup snímače bez RC filtru, který je navíc zatížen chybou vzorkování, protože skutečná perioda spojitého signálu se pohybuje kolem 1 ms.



Obr. 6.2: Srovnání výstupu snímače proudu s a bez RC filtru.

Bylo řečeno, že A/D převodník mikrokontroléru do signálu ze snímače zanesení šum. Z toho důvodu byl softwarově implementován filtr, který realizuje plovoucí průměr posledních 10 naměřených hodnot. Na obrázku 6.3 lze vidět zeleně vzorkovaný signál na výstupu RC filtru, tedy vstup číslicového filtru. Oranžově pak výstup číslicového filtru.



Obr. 6.3: Srovnání vstupu a výstupu číslicového filtru.

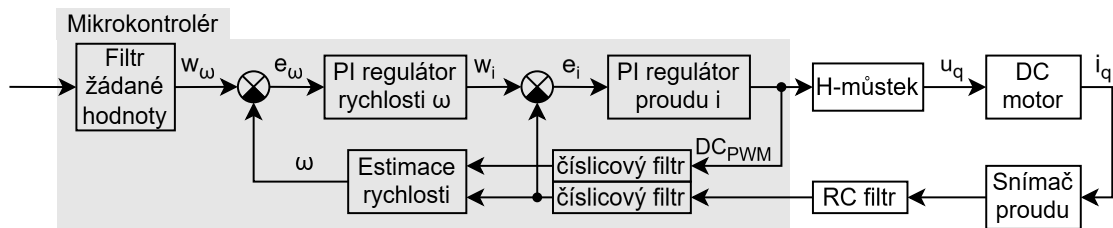
Pro realizaci číslicového filtru byla použita komunitní knihovna FIR-Filter, v programu FIR.h.

Během řízení se vzorkovací perioda pohybuje v jednotkách milisekund. Kvůli implementaci filtru posledních desítek hodnot samozřejmě dochází k prodloužení časové

konstanty snímače, což zhoršuje kvalitu regulace. Bez číslicové filtrace by ale šum A/D převodníku významně ovlivňoval akční veličinu. Řízení motoru v rámci desítek až stovek milisekund je pro tuto aplikaci stále akceptovatelné, takže si tuto filtraci mohou dovolit implementovat.

## 6.2 Řízení motoru

Hlavní úlohou mikrokontroléru je regulace rychlosti motoru. Implementovaný regulátor popisuje schéma na obrázku 6.4. Jde o úpravu schématu z obrázku 1.2.



Obr. 6.4: Schéma regulačního obvodu realizovaného v mikrokontroléru.

Protože není k dispozici zpětná vazba o skutečné rychlosti motoru, je tato rychlost dopočítávána z ostatních veličin podle vzorce 6.1.

$$\omega = \frac{u_i}{C\Phi} = \frac{u_q - R_d \cdot i_q}{C\Phi} = \frac{DC_{PWM} \cdot U_Z - R_d \cdot i_q}{C\Phi} \quad (6.1)$$

$DC_{PWM}$  ... střída výstupního PWM signálu, v rozsahu 0 - 1

$U_Z$  ... Napětí spínané H-můstkem, v tomto případě 24 V

$R_d$  ... odpor vinutí motoru

$i_q$  ... proud kotvy

$C\Phi$  ... konstanta motoru

Jak lze vidět ve schématu na obrázku 6.4, ve výpočtu je použita filtrovaná hodnota proudu popsaná v předchozí části. Výpočet s aktuální střídou způsoboval nestabilitu, takže je střída pro tento výpočet také filtrována, a to stejným způsobem jako proud.

Prudké změny střídou spínacího signálu H-můstkem způsobovaly výkyvy v napětí zdroje. Proto byl na vstupu realizován filtr žádané hodnoty typu rampa.

### 6.2.1 Regulátory

Pro samotnou regulaci rychlosti a proudu jsou použity dva PS regulátory, tedy diskretizované varianty PI regulátoru. Pro jejich implementaci jsem vytvořil vlastní

knihovnu PIController.h. Knihovna je poměrně jednoduchá. Pracuje s čísly typu float, implementuje omezení akčního zásahu a anti-windup ve formě omezení sumy odchylky.

V konstruktoru je nutné pro každou instanci zvolit parametry  $K_i$  a  $K_p$  a limity akčního zásahu a sumy odchylky. Tyto hodnoty se uloží jako soukromé proměnné. Jedinou veřejnou proměnnou je požadované hodnota regulované veličiny.

V každém průchodu smyčkou programu se pro každý regulátor volá funkce compute, které je nutno pomocí dvou argumentů předat aktuální stav regulované veličiny a čas od předchozího zavolání funkce. Ve funkci compute se provede výpočet odchylky, její přičtení k předchozí sumě a výpočet akčního zásahu podle dříve zadaných parametrů  $K_i$  a  $K_p$ .

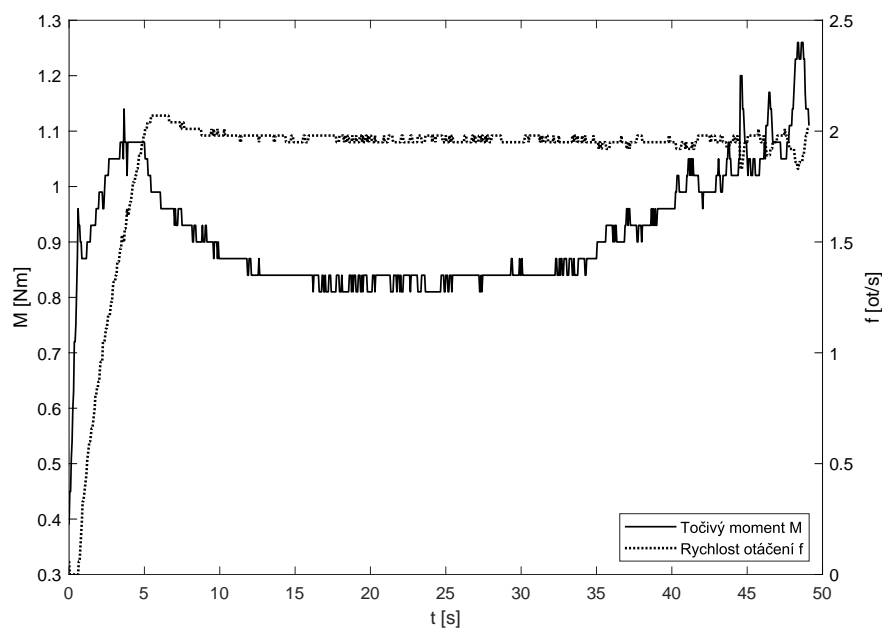
Bylo by možné upravit třídu tak, aby funkce compute nevyžadovala žádné argumenty. Každá instance by mohla držet ukazatel, odkud si přečte aktuální hodnotu regulované veličiny. Také čas předchozího výpočtu by si každá instance mohla hlídat sama, což by ale bylo v tomto případě výpočetně náročnější, protože funkce se vždy volají hned po sobě, takže je tento čas prakticky stejný.

Co se týče návrhu regulátorů z hlediska řízení, jejich parametry byly zvoleny heuristicky, na základě známých rozsahů požadovaných hodnot a akčních zásahů. Maximální požadovaná rychlost otáčení je 3 ot./s. Motor při této rychlosti bez zátěže odebírá kolem 2 A. Při zatížení je ale odběr vyšší, takže byly počáteční konstanty regulátoru zvoleny  $K_p = K_i = 1$ . Výsledným akčním zásahem je střída výstupního PWM signálu. Ta se v programu nastavuje pomocí čísla 0 – 255, přičemž v mojí aplikaci se pohybují do hodnoty 150. Maximální střída je tedy asi 60%. Počáteční parametry regulátoru proudové smyčky se tak pohybovaly v řádu desítek. Parametry byly poté experimentálně doladěny. Rigoróznější návrh regulátoru je nad rámec této práce.

Výsledný průběh regulačního děje lze vidět v grafu 6.5. Po zadání požadované hodnoty rychlosti byl vrták připojený k motoru zasouván do písku. Lze si všimnout, že rychlost zůstává konstantní, zatímco točivý moment - a tedy i proud kotvou - roste následkem zvyšování mechanického zatížení. Data byla zaznamenávána nadřazeným mikrokontrolérem, kterému byla zasílána přes sběrnici I2C. Muselo proto dojít k viditelnému amplitudovému kvantování. Vzorkovací frekvence je 50 ms.

## 6.3 Bezpečnostní funkce

V každém průchodu smyčkou se kontroluje několik parametrů nutných pro správný chod zařízení. Pokud dojde k chybě, požadovaná rychlost motoru se nastaví na nulu, připraví se zpráva pro nadřazený mikrokontrolér a rychlost není možné nastavit na jinou hodnotu, dokud chybový stav trvá.



Obr. 6.5: Průběh regulovaných veličin po zadání požadované hodnoty 2 ot./s.

Zpráva o požadované rychlosti od nadřazeného mikrokontroléru by měla přicházet každých 100 ms. Za chybový stav se pokládá, když čas od poslední zprávy překročí 500 ms.

Dále se kontroluje chybový výstup integrovaných obvodů IFX007T. Tato chyba může znamenat příliš vysoké nebo nízké napětí na jejich vstupu, nebo jejich přehřátí.

Komunikace s teploměrem na samotném motoru je pomalá, takže probíhá pouze každé dvě sekundy. To ale není limitujícím faktorem, jak bylo popsáno v kapitole 3.4. Překročení teploty 60 °C je opět vyhodnoceno jako chybový stav. O teplotě je ale nadřazený mikrokontrolér dopředu informován, jak je popsáno v tabulce 4.1.

## Závěr

Navržené zařízení je funkční a plní svoji úlohu v rámci vrtacího přípravku popsáno v úvodu. V prvních fázích návrhu byl kladen důraz na co nejrychlejší zprovoznění celého zařízení, aby byla otestována proveditelnost zvolených postupů. Byla tedy upřednostněna rychlost nad rigorózností návrhu. Záběr projektu je zároveň poměrně široký a tyto faktory spolu s nedostatkem zkušeností vyústily v některé nedostatky, které zde popíši.

Jak bylo uvedeno v kapitole 2, jako motor byla použita aku-vrtačka. Při měření parametrů se motor choval nekonzistentně - například při napájení konstantním napětím kolísala jeho rychlost. Lepších výsledků by tak mohlo být dosaženo použitím kvalitnějšího motoru. Od výrobců bývají standardně k dispozici parametry jejich motorů, takže by byla eliminována potřeba tyto parametry měřit. Také odezva snímače teploty vinutí by byla výrazně zrychlena, pokud by byl použit motor, ve kterém je snímač již zabudován.

Zpočátku návrh vůbec nepočítal s měřením rychlosti motoru. Když jsem rozpoznal, že měření této veličiny by bylo užitečné, bylo již pozdě zakomponovat do návrhu snímač rychlosti. To je důvod, proč byla implementována softwarová estimace rychlosti. Samotná estimace podává přesné výsledky, ale způsobuje problémy při řízení, protože do smyčky zavádí vazby, se kterými se těžko pracuje. Z hlediska řízení by tedy bylo vhodné do návrhu zakomponovat i měření otáček motoru.

Kvůli nedostatečné frekvenci PWM signálu, který spíná napájení motoru, značně periodicky kolísá proud kotvou. Nebylo zjištěno, že by toto kolísání způsobovalo mechanické problémy. Klade ale vyšší nároky na zdroj napájecího napětí a je kvůli němu zapotřebí filtrovat výstup snímače proudu, což zpomaluje proudovou regulační smyčku. Frekvence PWM je ale omezena použitým mikrokontrolérem. Bylo by tedy vhodné použít výkonnější mikrokontrolér, který by zároveň dovolil rychlejší vykonávání výpočtů regulátoru, nebo specializovaný obvod pro generování PWM signálu o vyšší frekvenci.

Závěrem bych ale chtěl zopakovat, že navzdory všem těmto nedostatkům finální zařízení splňuje všechny požadavky, které jsou na něj kladeny v rámci vrtacího přípravku. Ostatní subsystémy vrtacího přípravku také jeví určité problémy - mnou identifikované nedostatky tak mohou být adresovány v budoucnu, pokud dojde k celkové přestavbě vrtacího přípravku.



Obr. 6.6: Prezident republiky Petr Pavel testuje ranou, manuálně ovládanou verzi vrtáku.

# Literatura

- [1] Cyril Buttay. Structure of an h-bridge, 2006. [cit. 2024-10-12]. URL: [https://commons.wikimedia.org/wiki/File:H\\_bridge.svg](https://commons.wikimedia.org/wiki/File:H_bridge.svg).
- [2] Cyril Buttay. The two basic states of an h-bridge, 2006. [cit. 2024-10-12]. URL: [https://commons.wikimedia.org/wiki/File:H\\_bridge\\_operating.svg](https://commons.wikimedia.org/wiki/File:H_bridge_operating.svg).
- [3] Colin M.L. Burnett. Sample inter-integrated circuit (i<sup>2</sup>c) schematic, 2006. [cit. 2024-12-4]. URL: <https://commons.wikimedia.org/wiki/File:I2C.svg>.
- [4] KŘÍŽ, Martin. Řídicí systém vrtné soustavy pro marsovský rover. Online, bakalářská práce. Adam LIGOCKI (vedoucí práce). Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2025. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/168010>. [cit. 2025-05-08].
- [5] STRACHOŇ, Vilém. Přípravek zásobníku regolitu pro marsovský rover. Online, bakalářská práce. Adam LIGOCKI (vedoucí práce). Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2025. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/168012>. [cit. 2025-05-08].
- [6] ŠLÍMA, Filip. Přípravek lineárního pohonu pro marsovský rover. Online, bakalářská práce. Adam LIGOCKI (vedoucí práce). Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2025. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/168013>. [cit. 2025-05-08].
- [7] Václav Kůs. *Elektrické pohony a výkonová elektronika*. Západočeská univerzita v Plzni, 2016.
- [8] Pavel Koblík and Jiří Pavelka. *Elektrické pohony a jejich řízení*. České vysoké učení technické, 2016.

## Seznam symbolů a zkratek

<b>DC</b>	Direct current – stejnosměrný proud NEBO Duty cycle – střída
<b>DPS</b>	Deska plošných spojů
<b>PWM</b>	Pulse width modulation – pulzně šířková modulace

## A Obsah elektronické přílohy

```
/ ..... kořenový adresář přiloženého archivu
├── pcb ..... Soubory návrhu PCB. Vytvořeno v KiCad 8.0
│   ├── libs ..... Přídavné knihovny s použitými součástkami
│   └── pcb.kicad_pro ..... Soubor pro oteření celého projektu
├── software ..... Program mikrokontroléru. Vytvořeno v Arduino IDE 2.3.6
│   ├── hbridge_control
│   │   ├── hbridge_control.ino ..... Hlavní program
│   │   ├── PIController.cpp ..... Zdrojový soubor knihovny PIController
│   │   └── PIController.h ..... Hlavičkový soubor knihovny PIController
│   └── i2c_controller
│       └── i2c_controller.ino ... Program pro druhý, testovací mikrokontrolér,
│           který simuluje I2C mastera a může zasílat příkazy pro mé zařízení
```