

Univerzita Pardubice  
Dopravní fakulta Jana Pernera

Tester akčních členů pro autodiagnostiku

Diplomová práce

2020

Ondřej Kohoutek

Univerzita Pardubice  
Dopravní fakulta Jana Pernera  
Akademický rok: 2019/2020

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Ondřej Kohoutek**  
Osobní číslo: **D18503**  
Studijní program: **N3708 Dopravní inženýrství a spoje**  
Studijní obor: **Elektrotechnické a elektronické systémy v dopravě**  
Téma práce: **Tester akčních členů pro autodiagnostiku**  
Zadávací katedra: **Katedra elektrotechniky, elektroniky a zabezpečovací techniky v dopravě**

### Zásady pro vypracování

Navrhněte režimy testování akčních členů diagnostického testeru určeného pro testování funkcí akčních členů v automobilu a tester naprogramujte. Hardware testeru již existuje, zrevidujte ho a navrhněte úpravy, pokud to bude potřeba.

Doporučený postup:

1. Seznamte se s hardwarem testeru.
2. Navrhněte režimy testování akčních členů, které bude tester umožňovat.
3. Navrhněte úpravu hardware testeru, pokud to bude nutné.
4. Vytvořte program do mikrokontroléru.
5. Prakticky ověřte funkci testeru.

Rozsah pracovní zprávy:  
Rozsah grafických prací:  
Forma zpracování diplomové práce: **tištěná**

Seznam doporučené literatury:

1. Podklady k předmětům Elektrická a elektronická zařízení v konstrukci silničních vozidel, Diagnostika silničních vozidel, Mikroprocesorová řídicí technika, Regulace a automatizace.
2. GREGORA, Stanislav a Zdeněk MAŠEK. Elektronické a mechatronické systémy v konstrukci silničních vozidel. Pardubice: Tiskařské středisko Univerzity Pardubice, 2008. ISBN 9788073950828.
3. GSCHIEDLE, Rolf. Příručka pro automechanika. 2015. ISBN 9783808521632.
4. VLK, František. Automobilová elektronika 3: Systémy řízení motoru a převodů. Brno: František Vlk, 2006, vi, 269 s. ISBN 8023970631.
5. HEROUT, Pavel. Učebnice jazyka C. 4., přeprac. vyd. České Budějovice: Kopp, 2004, 271, viii s. ISBN 8072322206.
6. Podklady poskytnuté vedoucím práce.
7. Datasheety výrobců komponent.

Vedoucí diplomové práce: **Ing. Zdeněk Mašek, Ph.D.**  
Katedra elektrotechniky, elektroniky a zabezpečovací  
techniky v dopravě

Datum zadání diplomové práce: **12. února 2020**  
Termín odevzdání diplomové práce: **1. června 2020**

L.S.

**doc. Ing. Libor Švadlenka, Ph.D.**  
děkan

**Ing. Dušan Čermák, Ph.D.**  
vedoucí katedry

## **Prohlášení autora**

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze 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, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 7/2019 Pravidla pro odevzdávání, zveřejňování a formální úpravu závěrečných prací, ve znění pozdějších dodatků, bude práce zveřejněna prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 19. 7. 2020

Ondřej Kohoutek

## **Poděkování**

Chtěl bych poděkovat mé rodině za trpělivost při studiu a řešení této diplomové práce. Dále pak svému vedoucímu práce panu Ing. Zdeňkovi Maškovi, Ph.D. za trpělivost, poskytnutí neocenitelných odborných rad a materiálů a za pečlivost, se kterou se dané problematice věnoval. V neposlední řadě bych chtěl také poděkovat panu Ing. Jiřímu Blechovi za konzultace a ukázky diagnostiky akčních členů a za poskytnutí materiálního zázemí.

**Anotace**

Diplomová práce se zabývá návrhem softwaru pro tester akčních členů postavený na unifikovaném hardwaru. Obsahuje návrh řešení pro testování stejnosměrných motorů a elektromagnetických ventilů pomocí generátoru s bipolárním PWM výstupem. Dále je navrženo řešení pro testování krokových motorů. Část práce se věnuje modelování škrtkící klapky v Matlabu a zpětnovazebnímu řízení polohy škrtkící klapky pomocí PSD regulátoru.

**Klíčová slova**

Autodiagnostika, tester, akční člen, paralelní diagnostika

**Title**

Design of actuators tester for car diagnostics

**Annotation**

This master thesis deals with the design of software for an actuator tester based on unified hardware. It contains a solution for testing DC motors and solenoid valves using a generator with a bipolar PWM output. There is also a solution for stepper motors testing. One Part of the thesis deals with modeling of electronic throttle in Matlab and feedback control of the throttle position using a PSD controller.

**Keywords**

Car diagnostics, actuator, tester, parallel diagnostics

# Obsah

1	Teoretická část.....	13
1.1	Akční členy používané ve vozidlech .....	13
1.1.1	Škrtící klapka .....	14
1.1.2	EGR ventil.....	16
1.1.3	Krokový motor nastavení světlometu .....	17
1.2	Stejnoseměrné motory s permanentními magnety a jejich řízení .....	18
1.2.1	Obvodové schéma.....	19
1.2.2	Matematický popis [7] .....	19
1.2.3	Stejnoseměrné měniče napětí.....	20
1.3	Krokové motory a jejich řízení .....	24
1.3.1	Vlastnosti krokových motorů.....	24
1.3.2	Konstrukce krokových motorů .....	25
1.3.3	Způsoby krokování .....	29
1.3.4	Způsoby napájení krokových motorů .....	30
2	PWM generátor s bipolárním výstupem.....	32
2.1	Požadavky na tester.....	32
2.2	Popis Hardwaru.....	35
2.2.1	Mikrokontroler .....	36
2.2.2	Expandér vstupů/výstupů.....	38
2.2.3	Externí A/D převodník.....	39
2.2.4	Displej .....	41
2.2.5	H-můstek.....	42
2.2.5.1	Spínací režimy H-můstku .....	45
2.2.5.1.1	Locked anti-phase režim .....	46
2.2.5.1.2	Sign-magnitude režim .....	47
2.2.5.2	Omezení výstupního proudu H-můstku.....	49
2.2.5.3	Nadproudová ochrana H-můstku.....	53
2.2.5.4	Výkonové ztráty H-můstku.....	57
2.3	Lineární model škrtící klapky .....	59

2.3.1	Identifikace parametrů lineárního modelu .....	64
2.3.2	Model regulačního obvodu .....	67
2.4	Řídící software.....	74
2.4.1	Uživatelské funkce.....	74
2.4.2	Použité knihovny.....	76
2.4.3	Použité periférie MCU a jejich parametry .....	77
2.4.3.1	Časovač/čítač TC0 .....	78
2.4.3.2	Časovač/čítač TC2 .....	79
2.4.3.3	Časovač/čítač TC1 .....	82
2.4.3.4	I <sup>2</sup> C Sériové rozhraní .....	83
2.4.3.5	Interní A/D převodník MCU .....	84
2.4.3.6	Použitá přerušení .....	85
2.4.4	Ovládací prvky .....	87
2.4.5	Hlavní smyčka a časování aplikace .....	91
2.4.6	Externí A/D převodník.....	97
2.4.7	Reakce na nadproud.....	98
2.4.8	Generování PWM .....	100
2.4.8.1	Locked anti-phase režim .....	100
2.4.8.2	Sign-magnitude režim.....	101
2.4.8.3	Softwarové řešení .....	102
2.4.9	Obsluha displeje .....	105
2.4.10	Měření proudu.....	106
2.4.11	Měření odporu.....	108
2.4.12	Regulace polohy škrtkící klapky.....	109
2.5	Oživení a naměřené hodnoty.....	112
2.5.1	Měření teploty .....	112
2.5.2	Naměřené průběhy proudu a napětí.....	116
3	Tester krokových motorů .....	119
3.1	Požadavky na tester.....	119
3.2	Popis hardwaru.....	120
3.3	Řídící software.....	120



3.3.1	Použité periférie MCU a jejich parametry .....	121
3.3.2	Časovač/čítač TC0 .....	121
3.3.3	Použitá přerušení.....	121
3.3.4	Hlavní smyčka a časování aplikace .....	122
3.3.5	Ovládací prvky .....	125
3.3.6	Generování spínacích kombinací .....	126
3.3.7	Měření proudu.....	127
3.3.8	Reakce na nadproud.....	128
3.4	Naměřené průběhy proudu a napětí .....	128
4	Závěr.....	130
	Seznam použité literatury.....	132
	Seznam značek a zkratek .....	134
	Seznam obrázků .....	135
	Seznam tabulek .....	138
	Seznam příloh .....	139

## Úvod

Neustálý tlak na zlevňování, zvyšování efektivity a v neposlední řadě i na snižování emisí vede k tomu, že do automobilového průmyslu proniká čím dál tím větší množství elektronických komponent. Již tedy dávno neplatí, že automechanik si vystačí pouze s mechanickými nástroji pro diagnostikování poruch a k jejich opravám. Například seřizování chodu motoru spočívalo v nastavení mechanických částí vozidla jako jsou ventily, svíčky nebo karburátor. Mnohdy se opraváři rozhodovali pouze na základě své zkušenosti s daným typem vozidla, v počátcích neexistovaly na všechny modely ani opravárenské nebo servisní příručky. S vývojem elektroniky byly umožněny první kroky k modernizaci spalovacích jednotek. Hlavní motivací bylo snižování emisí a zvyšování účinnosti. Z tohoto důvodu vznikl v USA systém palubní diagnostiky OBD (On-Board-Diagnostics). Cílem je neustálé řízení a monitorování klíčových komponent, které mají vliv na emise výfukových plynů. Jsou to například Lambda sonda, katalyzátor, systém recirkulace spalin nebo zapalování. Jejich chyby se zaznamenávají do vnitřní paměti a hlásí se řidiči pomocí tzv. MIL kontrolky (Malfunction Indicator Light). Tento standard, dnes již ve verzi OBD II, používáme zejména při sériové neboli vnitřní diagnostice. Využívá se 16 pinová normalizovaná zásuvka, do které se připojí některý z mnoha vyráběných diagnostických nástrojů a je možné vyčíst chybové kódy z jednotlivých řídicích jednotek ve voze, případně i upravovat jejich vnitřní parametry a chybové kódy i mazat.

Se zvyšujícími požadavky zákazníků na komfort a bezpečnost dochází k implementaci dalších systémů. Jsou to například ABS, ESP, klimatizace, airbagy, multimédia systémy atd. Mnoho z těchto řídicích jednotek potřebuje tzv. akční člen – převodník elektrického signálu na mechanickou veličinu, nebo senzor převádějící sledovanou fyzikální veličinu na elektrický signál. Se stoupající složitostí vozů těchto komponent přibývá, a tedy roste i množství závad, které potenciálně může vzniknout, řetězit se a maskovat se navzájem.

Při určování, která z komponent může být vadná, využíváme zpravidla také tzv. paralelní diagnostiku neboli vnější. Jedná se o přímé měření elektrických signálů buď vstupních nebo výstupních za pomoci osciloskopu, multimetru nebo dalších specializovaných přístrojů. Toto zpravidla následuje jako druhý krok po vyčtení chybových kódů pomocí sériové diagnostiky. Podezřelou komponentu potom můžeme demontovat z vozidla a otestovat její činnost. Případně, pokud to je možné, otestovat správnou činnost komponenty přímo na vozidle. K tomuto účelu je nezbytné znát elektromechanické vlastnosti dané komponenty, abychom mohli vhodně zvolit metodu a testovací zařízení. Toto zařízení většinou umožňuje simulovat stejné či velmi podobné signály používané jak pro řízení akčních členů, tak jako

výstup ze senzorů. Můžeme tedy simulovat řízení podezřelé komponenty a vizuálně či měřením vyhodnotit, zda je reakce na patřičný průběh signálů správná. Velkou výhodou je, že nepotřebujeme žádný přístroj pro sériovou diagnostiku. Diagnostickým přístrojem pro sériovou diagnostiku (např. BOSCH KTS 540) bychom mohli test některých akčních členů provést, většinou se však jedná o velmi drahá profesionální zařízení. Dále můžeme použít toto testovací zařízení v procesu čištění, kdy může hýbat po nutnou dobu s mechanickým komponentem a usnadnit tak průnik čistícího prostředku do všech částí. Na druhou stranu je možné též ověřit, zda řídicí jednotka správně vyhodnocuje vstupní signál ze senzoru. Připojíme-li výstup testeru na vstup řídicí jednotky, můžeme sledovat reakci na simulovaný signál a vyvodit závěr, zda je problém se senzorem, nebo s řídicí jednotkou.

Tento způsob testování je velmi efektivní, nedochází při něm k vynaložení zbytečných finančních prostředků kvůli výměně dílu naslepo (to prodražuje opravu), a tím se čas celé opravy zbytečně neprodlužuje. Zákazník tak nečeká příliš dlouho na opravu, a automechanik je schopen zpracovat větší objem zakázek.

Na trhu se momentálně prodávají zařízení např. Pulse Controll od firmy RenéMoto, který umožňuje testovat zapalovací cívky, vstřikovače, ventily klimatizace a turbodmychadla nebo otáčkoměry a rychloměry. Výstupní parametry jsou nastavení střídavosti 1–100 %, nastavení frekvence 20 Hz – 20 kHz, výkonový výstup 12 V při maximálním proudu 10 A. Některé firmy se specializují na úzký okruh testovaných komponent, např. pouze pro škrtící klapky jako výrobek ELECTROTHROTTLER firmy Automotive Diagnostics & Publishing [26] nebo G3-REA-Master, což je tester akčních členů turbodmychadel od firmy G3 Concepts Ltd [27]. Dalším výrobcem je česká firma JB-Elektronik, která má ve svém portfoliu několik zajímavých testerů např. M-PWM2, se kterým je možno testovat velké množství typů ventilů jako např. EGR, regulace turbodmychadla, motor škrtící klapky, měniče tlaku a další. Má zabudované automatické režimy pro usnadnění čištění, ochranu proti zkratu nebo také signálový výstup pro simulaci snímače s frekvenčním výstupem. Střídavost lze nastavit v rozsahu 0,1-99,9 %, frekvence v rozsahu 10 Hz – 10 kHz. Maximální výstupní proud 4 A trvale, 10 A špičkově.

Právě s firmou JB-Elektronik (dále zadavatel práce) byla navázána spolupráce v rámci této diplomové práce. Cílem je vyvinout nástupce současných typů testerů akčních členů, které má JB-Elektronik v nabídce, prioritně pro M-PWM2-A, což je tester – generátor signálových nebo výkonových impulzů s jedním výkonovým a jedním signálovým výstupem, vypínatelnou antiparalelní diodou a šesti variabilními režimy pro automatické testování.

Mým hlavním úkolem bylo vytvořit software pro nový univerzální tester na jedné hardwarové platformě (zejména unifikovaná DPS), která by měla být použita pro všechny typy aktuálních typů testerů, které má zadavatel v nabídce. Bylo také přihlédnuto k budoucím potřebám. To znamená uvažovat rezervu (paměťového prostoru MCU) jak pro případné rozšíření softwaru, tak pro úpravy hardwaru. Zadavatel po konzultaci provedl změny vlastního hardwaru (i s ohledem na cenu komponent) tak, aby zapojení a použité komponenty korespondovaly s požadavky na software. Od zadavatele jsem dostal prototyp osazené DPS, na které jsem odladil software, a otestoval všechny navrhované funkce.

# 1 Teoretická část

## 1.1 Akční členy používané ve vozidlech

Akčním členem rozumíme elektromechanické zařízení, které většinou převádí elektrický signál z řídicího prvku na mechanickou veličinu. Mohou to být ale také zapalovací moduly, relé nebo polovodičové spínací prvky. Z pohledu regulace tedy zasahují do regulované soustavy. Většinou se jedná o elektromagnetické prvky jako jsou solenoidy, krokové nebo stejnosměrné motory ale také piezoelektrické elementy, které jsou součástí vstřikovacích ventilů některých vstřikovačů common-rail. Některé akční členy se používají v podobném nebo stejném provedení napříč všemi výrobci, ale některé jsou osazeny pouze u vozidel určité technologie a nemusí to být jen rozdělení na vznětová a zážehová vozidla. Kompletní seznam by byl velmi obsáhlý, v tabulce níže jsou tedy uvedeny ty nejběžnější akční členy. V zásadě bychom je mohli rozdělit, také podle toho, v jakém subsystému se nachází na oblasti:

- Řízení chodu motoru (vstřikování, příprava směsi, přeplňování, předstih, řízení emisí, chlazení...)
- Komfortní systémy (nastavení volantu, sedaček, klimatizace...)
- Bezpečnostní systémy (osvětlení, brzdový systém)

DC motory	Krokové motory	Solenoidy
palivové čerpadlo	obtokový ventil	vstřikovací ventil
ventilátor chladiče	EGR ventil recirkulace spalin	EGR ventil recirkulace spalin
škrťící klapka	natočení světlometů AFL, AHL	spojka kompresoru klimatizace
stahování oken		variabilní nastavení vačkového hřídele VCT
naklopení světel		ventil vzduchu volnoběhu IAC
		ventil klapky sacího potrubí
		ventil plnicího vzduchu
		obtokový ventil turbodmychadla
		VGT ventil turbodmychadla
		regulace tlaku paliva IMV / DRV

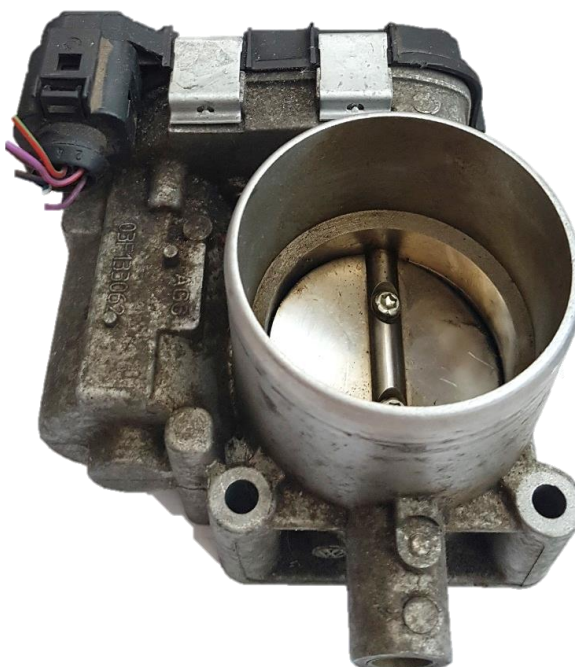
Tab. 1: Přehled typů pohonů pro akční členy

Na následujících stránkách je uveden detailnější popis zástupce každé skupiny akčních členů.

### 1.1.1 Škrťící klapka

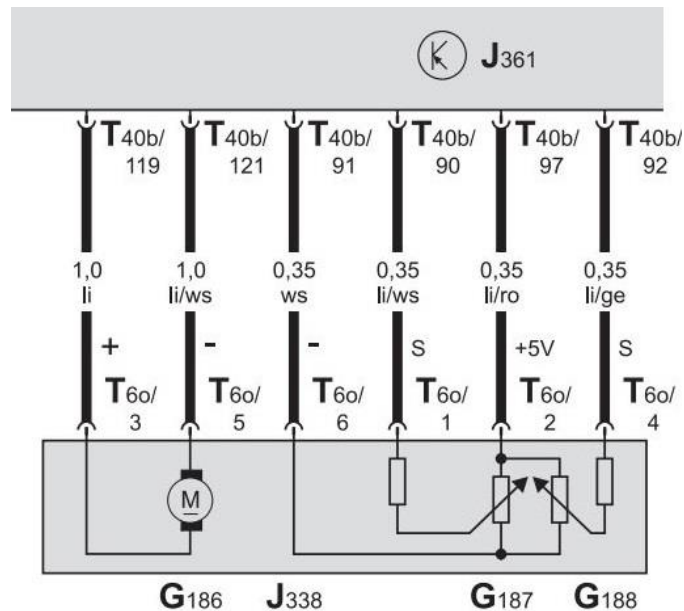
Škrťící klapka se používá u zážehových motoru pro kvantitativní regulaci nasávaného vzduchu do motoru. Může být ovládána buď mechanicky nebo elektricky prostřednictvím plynového pedálu. V současnosti se používá elektrické ovládání, které umožňuje řídit polohu klapky do jisté míry nezávisle na poloze plynového pedálu. Dochází tak ke snížení spotřeby paliva a emisí výfukových plynů a zvyšuje komfort jízdy díky funkci tempomatu. Úhel otevření škrťící klapky se nastavuje na základě polohy plynového pedálu, který v podstatě reprezentuje požadavek na moment. Hmotnost nasávaného vzduchu je potom vypočtena na základě úhlu natočení klapky a hustoty vzduchu závislé na teplotě. Řídící jednotka poté vypočítává potřebné množství vstřikovaného paliva.

Z konstrukčního hlediska se jedná o hliníkový odlitek, ve kterém je prostor pro motor (stejnoseměrný s permanentním magnetem) s převody a elektroniku, která poskytuje buď za pomoci potenciometrů nebo bezkontaktních senzorů údaj o poloze škrťící klapky. Vlastní klapka má elipsovitý tvar a je umístěna ve válci, který je také součástí celého chasis. Při poruše klapky dochází k navrácení klapky pružinou do výchozí polohy, kdy je klapka mírně pootevřena a umožňuje nouzový dojezd. Ta je označována jako limp-home position. Na Obr. 1 je škrťící klapka z vozu Škoda Roomster motorizace 1.4 63 kW.



Obr. 1: Škrťící klapka z vozu Škoda Roomster 1.4 63 kW

Na Obr. 2 níže je připojení klapky k řídicí jednotce vozu. Jedná se o schéma k vozu Škoda Octavia II. U každého vozidla může být zapojení konektoru jiné, je proto nutné jej vyhledat pomocí specializovaného software nebo v online databázi.

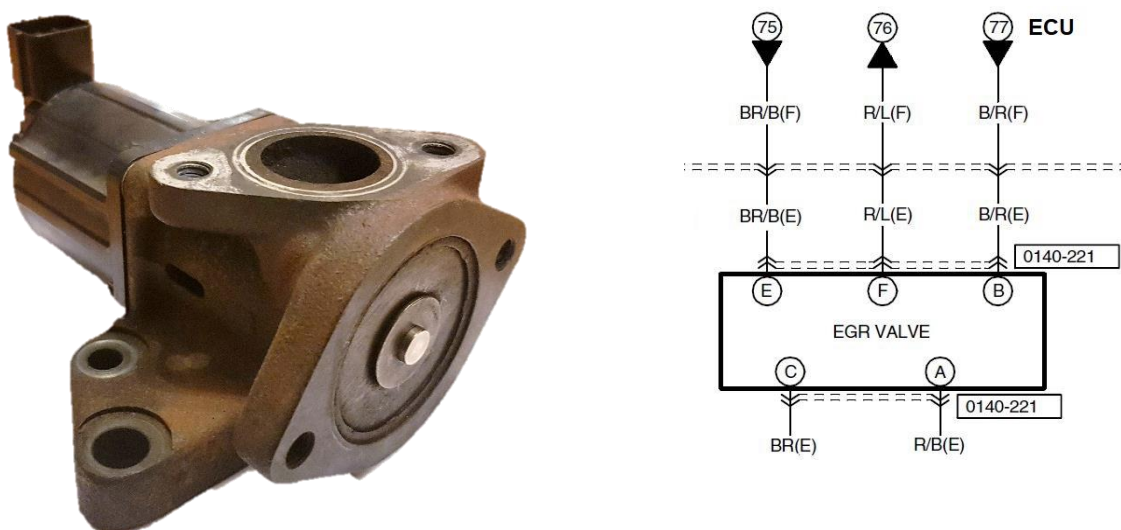


Obr. 2: Připojení škrtky klapky ve voze Škoda Octavia II generace

Z elektrických parametrů bylo změřeno výstupní napětí ze senzorů na pinu 1 a 4, které jsou vůči sobě opačné. Když jeden ze signálů roste, druhý klesá a naopak. Na obou senzorech je rozsah nepatrně jiný. Napájení senzoru 5 V bylo změřeno 4,96 V. Pro pin 4 je to 0,48 – 4,4 V, v limp-home pozici 4,16 V. Pro pin 1 pak 0,56 – 4,56 V, v limp-home pozici 0,96 V. Součet napětí obou snímačů polohy je roven napájecímu napětí 5 V s odchylkou do 0,2 V. Odpor kotvy byl u stojící klapky změřen nepřímou volt-ampérovou metodou pomocí úbytku napětí na svorkách 3,6 V a proudu 1,22 A změřeného proudovou sondou na osciloskopu. Podle Ohmova zákona tedy dostaneme odpor přibližně 2,95  $\Omega$ . Při zahřátí vinutí by odpor vinutí ještě stoupl. Indukčnost kotvy byla změřena 1,38 mH LC metrem UT602.

### 1.1.2 EGR ventil

Jedná se o zařízení pro recirkulaci výfukových plynů, které je zapojeno do sání i do výfuku motoru. Přepouštěním spalín pomocí elektromagnetického ventilu do sání dochází u zážehových motorů k smíchání čerstvé směsi paliva, vzduchu a výfukových plynů, u vznětových motorů se míchá pouze nasávaných vzduch do motoru s výfukovými plyny. Přisáváním spalín se snižuje maximální teplota spalování. Dochází tak ke snižování emisí oxidů dusíku. Recirkulace spalín je nezbytná zejména u motorů s přímým vstřikováním [1]. Samotný ventil může být buď ovládaný pneumaticky podtlakem prostřednictvím EPV ventilu, nebo ovládaný elektricky, kdy je elektromagnetický element přímo spojen s ventilem mezi sacím a výfukovým potrubím. V některých typech vozů se může vyskytovat i krokový motor (např. Mitsubishi Lancer).



Obr. 3: EGR ventil z vozu Mazda 6 2.0 DI a schéma připojení k ECU

Na Obr. 3 je EGR ventil z vozu Mazda 6 motorizace 2.0 DI a jeho zapojení do řídicí jednotky. Oproti jiným modelům z řady 6 má ovládání pomocí elektromagnetu, který vytlačuje kužel z příruby ven. Zpětnou vazbu poskytuje potenciometr skrz 5 pinový konektor. Naměřený odpor vlnutí mezi piny F a B je  $5,9 \Omega$  a indukčnost  $8,08 \text{ mH}$ . Mezi piny A a E je v zavřeném stavu  $6 \text{ k}\Omega$  a v otevřeném  $1,2 \text{ k}\Omega$ . Mezi piny C a E pak  $0,9 \text{ k}\Omega$  a  $5,75 \text{ k}\Omega$ .

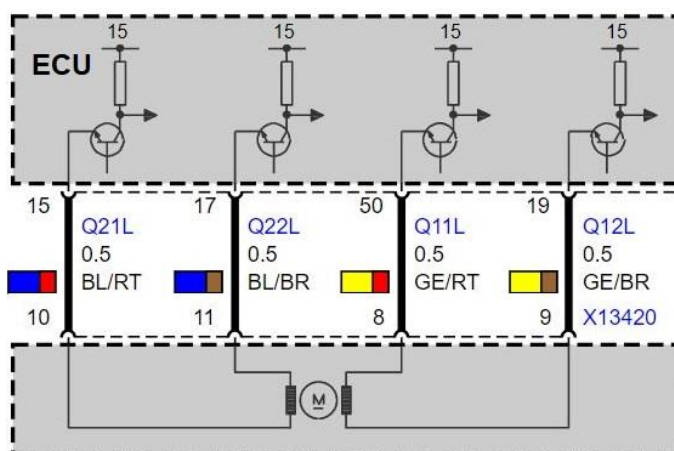


### 1.1.3 Krokový motor nastavení světlometu

V současných moderních světlometech s xenonovými výbojkami nebo LED zdroji jsou použity k nastavení parametrů světelného svazku krokové motory. Například ve světlometech firmy Hella s technologií AFS (Advanced Frontlighting System) otáčí válcem uvnitř optické soustavy mezi zdrojem světla a čočkou. Dokáže tak v řádu milisekund změnit geometrii paprsku a za pomoci dalších mechanismů umožňujících pohyb ve více osách se tento světlomet dokáže přestavit do 7 různých režimů [2]. Dále se používají krokové motory v adaptivních světlometech pro nastavování výšky, například ve voze BMW E90 řady 3. U nejmodernější generace světlometu je obsažen stejný krokový motor, ale je již integrován do tzv. ISM modulu (Intelligent Stepper Motor). Jedná se o mechatronický díl obsahující krokový motor, diagnostický a komunikační hardware komunikující po LIN sběrnici [3]. Důvody pro použití krokových motorů uvádí výrobce hlavně rychlost odezvy, přesnost, rozlišení a nízkou hlučnost [3]. U většiny aplikací se používají bipolární krokové motory.



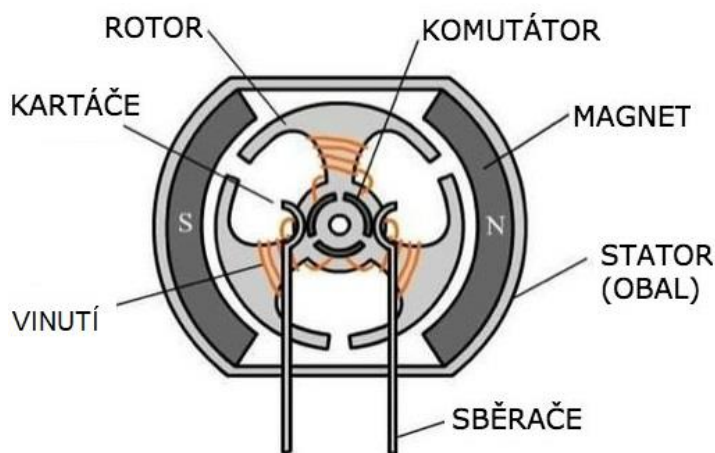
Obr. 4: ISM modul Hella [3]



Obr. 5: Ilustrativní schéma připojení motoru naklápění světlometu k řídicí jednotce ve voze BMW E90 [4]

## 1.2 Stejnosměrné motory s permanentními magnety a jejich řízení

Stejnosměrný motor s permanentními magnety se řadí mezi stroje s cizím buzením, v tomto případě s konstantním buzením. Stator je tvořen permanentními magnety, rotor je tvořen vinutím na železném jádře zpravidla z transformátorových plechů s počtem pólů dle typu konstrukce. Změnu polarity elektrického proudu kotvou a tím změnu magnetického pole zajišťuje komutátor. Na jeho lamely jsou připojeny jednotlivé části rotorového vinutí a na jeho povrch přiléhají kartáče, kterými se teče proud do nebo z vinutí. Středem rotoru pak prochází hřídel uložená na obou koncích motoru v přírubových ložiskách.



Obr. 6: Řez stejnosměrným motorem se třemi póly [5]

Charakteristické vlastnosti stejnosměrných motorů s PM:

### Výhody:

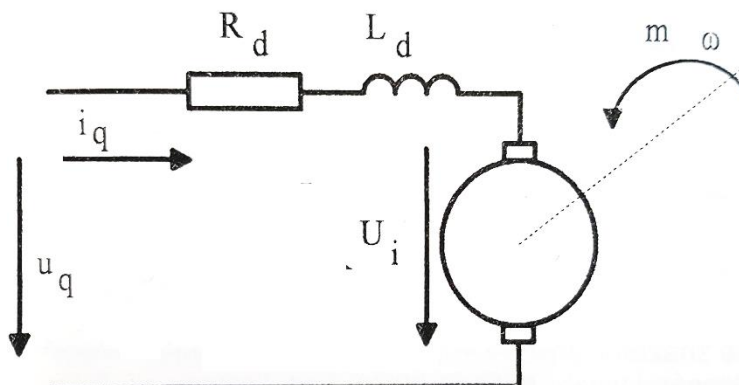
- Vysoký záběrný moment od nízkých otáček
- Jednoduchá změna směru otáčení otočením polarity kotevního napětí
- Řízení otáček regulací velikostí kotevního napětí
- Velký rozsah provozních otáček

### Nevýhody:

- Komutátor snižuje životnost stroje
- Nízko otáčkové stroje s ventilátorem na hřídeli se špatně chladí
- Napájecí napětí je omezeno maximálním lamelovým napětím (přeskoky)
- Větší rozměry oproti asynchronnímu motoru
- Teplotní citlivost permanentních magnetů

### 1.2.1 Obvodové schéma

Zjednodušené elektrické schéma cize buzeného motoru na postačuje pro základní analýzu chování pohonu. Matematický model stejnosměrného cize buzeného motoru tvoří soustava rovnic, která platí pro vztahy mezi elektrickými veličinami, momenty a rychlostí stroje [6].



Obr. 7: Elektrické schéma stejnosměrného motoru s permanentním magnetem, upraveno z [6]

### 1.2.2 Matematický popis [7]

Indukované elektromotorické napětí v kotvě  $U_i$  je dáno:

$$U_i = K\Phi\omega_m \quad [V] \quad (1)$$

Kde  $K$  je konstrukční konstanta stroje [-],  $\Phi$  je magnetický tok [Wb] a  $\omega_m$  úhlová rychlost rotoru [rad/s]. Z rovnice (1) vyplývá, že indukované napětí je úměrné magnetickému toku stroje a rychlosti otáčení. Změna polarity indukovaného napětí je tedy možná změnou smyslu magnetického toku  $\Phi$  nebo rychlosti otáčení  $\omega_m$ .

Elektromagnetický moment stroje  $M_i$  je určen rovnicí:

$$M_i = \frac{2}{a} \cdot \frac{N}{\pi} \cdot \Phi \cdot I = K \cdot \Phi \cdot I \quad [Nm] \quad (2)$$

Kde  $K$  je konstrukční konstanta stroje [-],  $\Phi$  je magnetický tok [Wb], a  $I$  je celkový proud stroje [A]. Podle rovnice je možné dosáhnout opačného smyslu točení motoru změnou polarity magnetického toku  $\Phi$  nebo změnou polarity proudu  $I$ .

Pro obvod kotvy můžeme definovat napěťovou rovnici podle Kirchoffova zákona:

$$u_q = R_d i_q + L_d \frac{di_q}{dt} + U_i \quad [V] \quad (3)$$

Kde  $R_d$  je odpor vinutí kotvy [ $\Omega$ ],  $L_d$  je indukčnost vinutí kotvy [H],  $U_i$  indukované elektromotorické napětí v kotvě [V],  $i_q$  kotevní proud [A],  $u_q$  kotevní napětí [V]. V rovnici (3) je zanedbán úbytek napětí mezi kartáči a komutátorem. Ten se s proudem mění jen velmi málo a uvažuje se přibližně 2 V.

Pro vnitřní moment stroje  $m_i$  platí:

$$m_i = J \frac{d\omega_m}{dt} + m_z \quad [Nm] \quad (4)$$

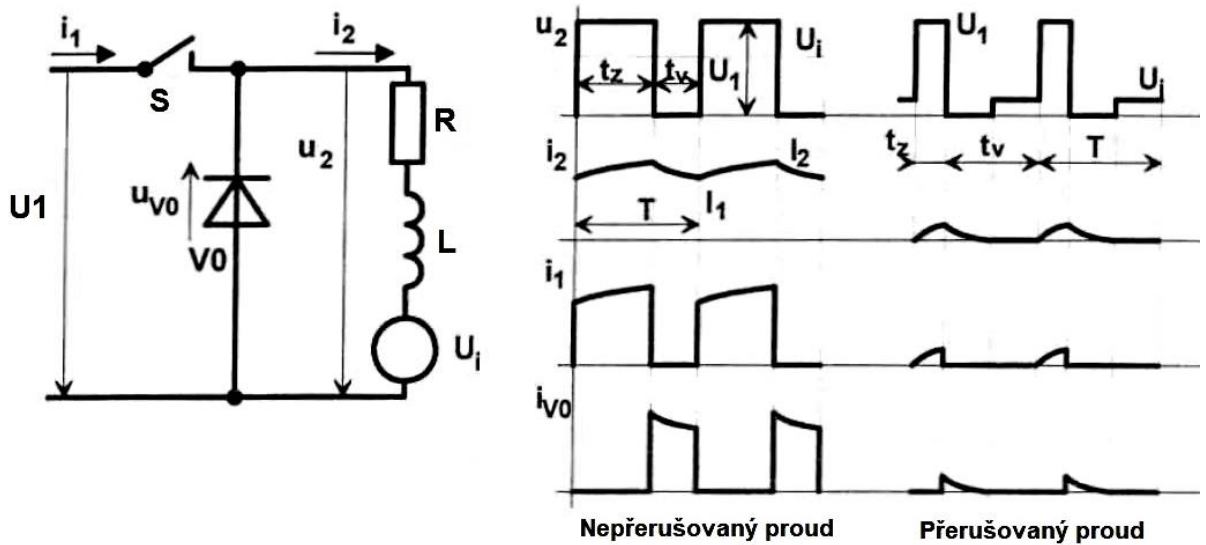
Kde  $m_z$  je mechanický moment zátěže na hřídeli [Nm] a  $J$  moment setrvačnosti rotujících hmot na hřídeli [ $\text{kgm}^2$ ].

Z výše uvedených vztahů vyplývá, že regulace otáček stejnosměrného motoru s permanentními magnety je možná pouze změnou napětí kotvy vzhledem k nemožnosti měnit magnetický tok. Při rozběhu motoru je indukované napětí v kotvě nulové, proud je omezen jen odporem vinutí kotvy. Po roztočení motoru proud i moment klesají. Rychlost motoru se ustálí, jakmile je moment motoru v rovnováze s momentem zátěže a momentem ztrát naprázdno.

### 1.2.3 Stejnosměrné měniče napětí

Tyto měniče převádí vstupní stejnosměrné napětí a proud na výstupní stejnosměrné napětí a proud zpravidla jiné velikosti. Používají se zpravidla v aplikacích se zdrojem stejnosměrného napětí jako jsou vozidla napájená z trakčního vedení nebo z akumulátorů. Existují dva typy měničů, a to přímé a nepřímé. Přímý měnič pracuje pulzním způsobem na bázi periodického přerušování napájecího obvodu. Nepřímý měnič obsahuje střídač, střídavý meziobvod a řízený usměrňovač. Výhodou tohoto měniče je galvanické oddělení [8].

Vzhledem k typu zátěže se dále budu zabývat popisem stejnosměrného měniče s RL zátěží a se zdrojem napětí. Toto napětí reprezentuje indukované napětí  $U_i$  ve stejnosměrném motoru a je přímo úměrné rychlosti otáčení  $\omega_m$ .



Obr. 8: Stejnoseměrný měnič napětí s aktivní zátěží, upraveno z [8]

Na Obr. 8 představuje  $U_i$  indukované napětí v kotvě motoru [V],  $R$  [ $\Omega$ ] a  $L$  [H] odpor a indukčnost kotvy motoru.  $V_0$  představuje nulovou diodu.

Při sepnutí spínače  $S$  platí:

$$u_2 = Ri_2 + L \frac{di_2}{dt} + U_i = U_1 \quad (5)$$

Řešením této diferenciální rovnice s počáteční podmínkou  $i_{2(0)} = I_1$  je:

$$i_2 = \frac{U_1 - U_i}{R} \cdot \left(1 - e^{-\frac{t}{\tau}}\right) + I_1 \cdot e^{-\frac{t}{\tau}} \quad (6)$$

Kde  $\tau = L/R$ .

Při rozepnutí spínače platí:

$$u_2 = Ri_2 + L \frac{di_2}{dt} + U_i = 0 \quad (7)$$

Řešením této diferenciální rovnice s počáteční podmínkou  $i_{2(0)} = I_2$  je:

$$i_2 = \frac{-U_i}{R} \cdot \left(1 - e^{-\frac{t}{\tau}}\right) + I_2 \cdot e^{-\frac{t}{\tau}} \quad (8)$$

Proud  $i_2$  nezaniká exponenciálně k nule, ale k hodnotě  $(-U_i/R)$ . Pokud poklesne proud  $i_2$  v průběhu doby  $t_v$  na nulu, dioda  $V_0$  přestane vést proud a na zátěži zůstane pouze napětí  $U_i$ . Potom je průběh proudu přerušovaný, viz Obr. 8 vpravo.

Pro střední hodnotu napětí na zátěži  $U_{2(AV)}$  při nepřerušovaném proudu platí:

$$U_{2(AV)} = z \cdot U_1 = RI_{2(AV)} + U_i \quad (9)$$

Kde  $z$  je poměrná doba zapnutí a mění se v rozsahu 0 až 1. Výstupní napětí lze tedy pouze snižovat, a to v rozsahu 0 až  $U_1$ .

Střední hodnota proudu tekoucího zátěží  $I_{2(AV)}$  vychází z rovnice (9) jako:

$$I_{2(AV)} = \frac{z \cdot U_1 - U_i}{R} \quad (10)$$

Zvlnění výstupního proudu  $\Delta I_2$  vypočítáme jako:

$$\Delta I_2 = I_2 - I_1 \quad (11)$$

Z rovnic (6) a (8) získáme vztah pro proudy  $I_1$  a  $I_2$  a po úpravě je jejich řešení:

$$\Delta I_2 = \frac{U_1}{f \cdot L} \cdot z \cdot (1 - z) \quad (12)$$

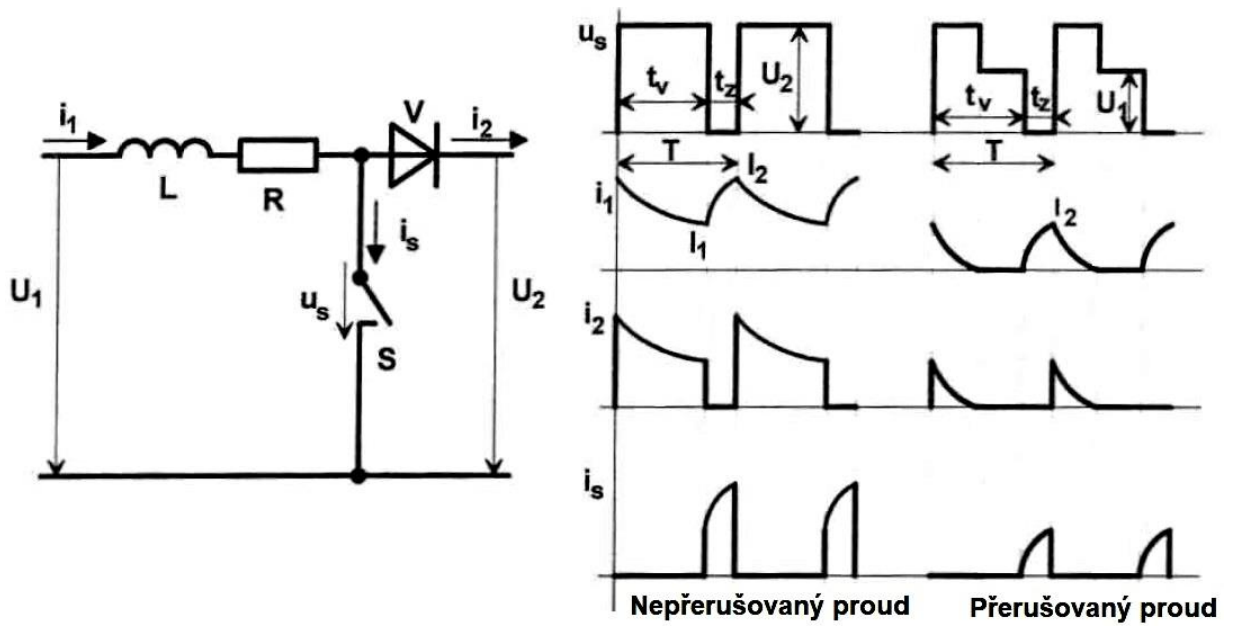
Kde  $f = 1/T$  [Hz],  $z$  je poměrná doba zapnutí (zatěžovatel, střída).

Maximální zvlnění je pro  $z = 0,5$  a má hodnotu:

$$\Delta I_{2max} = \frac{U_1}{4 \cdot f \cdot L} \quad (13)$$

Z rovnice (13) vyplývá, že maximální zvlnění je nepřímě úměrné spínací frekvenci a velikosti indukčnosti  $L$  v obvodu. Zvyšováním spínací frekvence je tedy možné snížit maximální zvlnění proudu  $\Delta I_2$  [8].

Pokud chceme napětí na výstupu měniče zvětšovat, musíme použít jiné zapojení jako je na Obr. 9. Při sepnutí spínače  $S$  narůstá proud  $I_1$  a při vypnutí naopak klesá. V indukčnosti  $L$  se při sepnutí v době  $t_z$  hromadí energie vlivem vysokého proudu  $I_1$ . Při rozepnutí se potom sečte napětí  $U_1$  a napětí indukované na indukčnosti  $L$ . Proud zátěží teče pouze po dobu vypnutí  $t_v$ . Za diodou  $V$  je obvykle připojen kondenzátor proti zemi. I v tomto zapojení může měnič pracovat jak v režimu přerušovaného, tak v režimu nepřerušovaného proudu, viz Obr. 9 vpravo. Při analýze dvou možných stavů měniče budeme postupovat analogicky k předchozímu zapojení.



Obr. 9: Zvyšující stejnosměrný měnič, upraveno z [8]

Na Obr. 9 představuje  $U_1$  indukované napětí v kotvě motoru,  $R$  a  $L$  odpor a indukčnost kotvy motoru.

Při sepnutí spínače  $S$  je  $i_1 = i_s$  a  $i_2 = 0$  a dále platí:

$$u_s = 0 = -U_1 + Ri_1 + L \frac{di_1}{dt} \quad (14)$$

Řešením této diferenciální rovnice s počáteční podmínkou  $i_2(0) = I_1$  je:

$$i_1 = \frac{U_1}{R} \cdot \left(1 - e^{-\frac{t}{\tau}}\right) + I_1 \cdot e^{-\frac{t}{\tau}} \quad (15)$$

Kde  $\tau = L/R$ .

Při rozepnutí spínače je  $i_s = 0$ ,  $i_2 = i_1$ ,  $u_s = U_1$  a dále platí:

$$u_2 = Ri_2 + L \frac{di_2}{dt} + U_1 = 0 \quad (16)$$

Řešením této diferenciální rovnice s počáteční podmínkou  $i_1(0) = I_2$  je:

$$i_1 = \frac{-U_2 - U_1}{R} \cdot \left(1 - e^{-\frac{t}{\tau}}\right) + I_2 \cdot e^{-\frac{t}{\tau}} \quad (17)$$

Střední hodnota nepřerušovaného proudu  $I_{1(AV)}$  je:

$$I_{1(AV)} = \frac{U_1 - U_2 (1 - z)}{R} \quad (18)$$

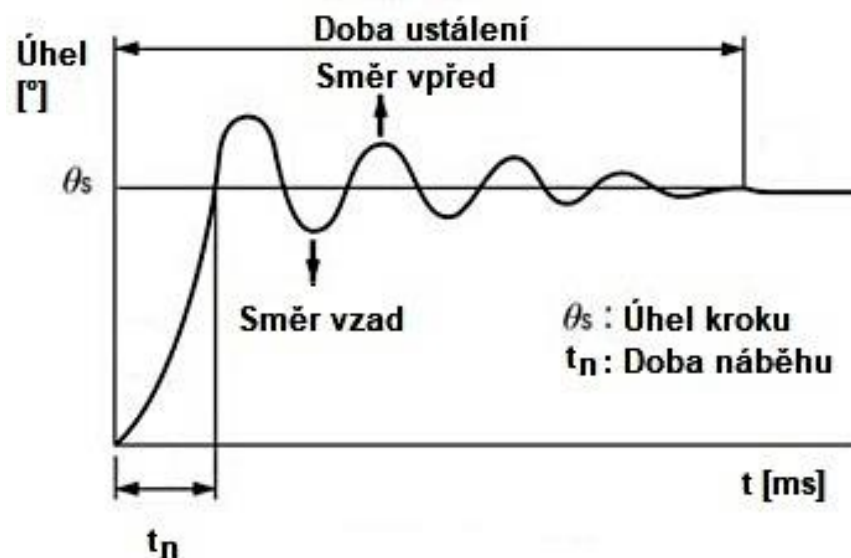
Z rovnice (18) vyplývá, že pomocí poměrné doby zapnutí  $z$  můžeme řídit velikost proudu  $I_{1(AV)}$ .

### 1.3 Krokové motory a jejich řízení

Krokové motory se řídí jiným způsobem než stejnosměrné DC motory. Nelze je roztočit pouhým přiložením stejnosměrného napětí. Jejich pohyb není spojitý, ale probíhá po krocích, jejichž velikost je uvedena ve stupních a je daná konstrukčně počtem pólů rotoru a počtem fází statoru. Pro jednoduchost v našem případě budeme uvažovat dvoufázový motor, tedy takový, který má dvě vinutí posunutá oproti sobě a takový motor připojujeme čtyřmi vodiči. Pro řízení pohybu je potřeba vždy řídicí elektronika, která generuje impulzy v určité kombinaci a posloupnosti tak, aby krokový motor vykonával požadovaný pohyb v daném směru.

#### 1.3.1 Vlastnosti krokových motorů

Odezva na posun o jeden krok má charakter kmitavého přechodného jevu. Frekvence těchto tlumených kmitů je vlastní frekvencí pohonu. Pokud je tato vlastní frekvence blízká krokovací frekvenci, dojde k rezonanci, která může vést ke ztrátě kroku a k výrazné ztrátě momentu. Na průběh oscilací má vliv charakter připojené zátěže na hřídeli motoru. Tyto oscilace lze utlumit zvýšením tření v ložiscích, tím se doba ustálení zkrátí [9].



Obr. 10: Odezva motoru na jeden krok, upraveno z [10]

Maximální rychlost krokování je závislá na průběhu odezvy ovlivněném připojenou zátěží a parametry konkrétního typu motoru. Čím větší má zátěž moment setrvačnosti, tím menšího počtu kroků za sekundu je možno dosáhnout [9]. Rychlost otáčení neboli krokování, se udává v počtech kroků nebo pulzů za sekundu [krok/s] nebo jako úhel natočení za sekundu [°/s].

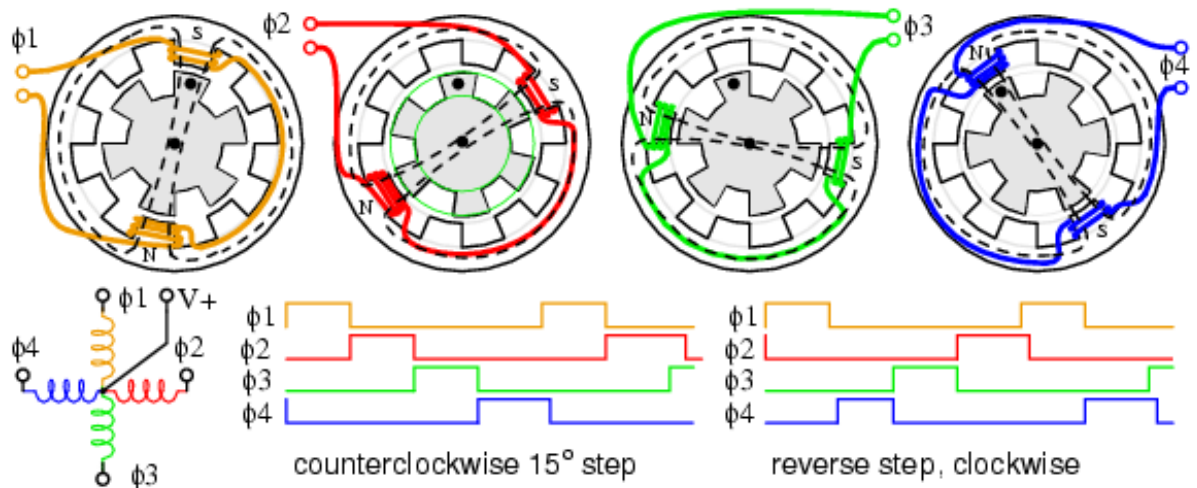


### 1.3.2 Konstrukce krokových motorů

Krokové motory dělíme na typy s pasivním rotorem – reluktanční a s aktivním rotorem – hybridní, nebo s permanentními magnety.

#### Reluktanční krokový motor

Jedná se o stroj jednoduché a historicky nejstarší konstrukce, který nepotřebuje buzení. Běžně má 20 až 200 kroků na otáčku. Přídržný moment je malý, reziduální moment téměř nulový. Magnetický obvod rotoru i statoru je složen z plechů. Rotor s vyniklými póly je vyroben z magneticky měkkého materiálu. Stator má též vyniklé póly s vinutími jednotlivých fází, které pracují jako elektromagnety. Rotor má však vždy jiný počet (obvykle menší) pólů než stator. Na pólech rotoru i statoru jsou zuby pro dosažení jemnějšího kroku motoru. Činnost krokového motoru je založena na změnách reluktance magnetického obvodu stroje při otáčení rotoru. Pokud zanedbáme reluktanci železa vůči vzduchu, rozptylu a zakřivení magnetických indukčních čar ve vzduchové mezeře, je reluktance obvodu fáze statoru lineární funkcí překrytí zubů statoru a rotoru. Indukčnost cívky fáze je tedy největší, jsou-li osy zubů rotoru a statoru proti sobě. Minimální je v poloze posunutě o polovinu zubové rozteče [11]. Rotor se vždy nastaví do pozice, kde je minimální hodnota magnetického odporu  $R_m$  a magnetický tok  $\Phi$  se tedy uzavírá cestou nejmenšího odporu. Rozdíl úhlového rozestupu pólů rotoru a statoru udává velikost kroku motoru.

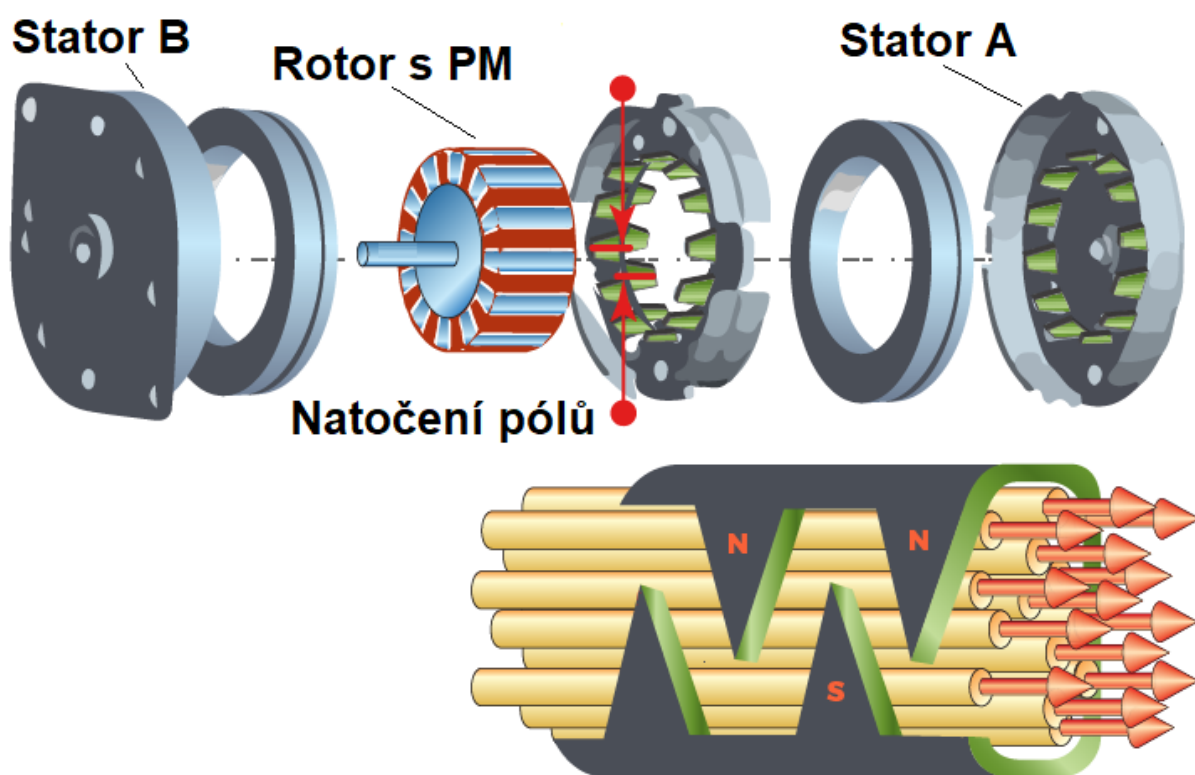


Obr. 11: Řez krokovým motorkem s proměnnou reluktancí, uspořádání fází a spínací diagram, převzato z [9].

Na Obr. 11 je zobrazen čtyř fázový reluktanční krokový motor s osmi póly na statoru a šesti póly na rotou. Při spínání fází podle pořadí na Obr. 11 vlevo se bude motor otáčet proti směru hodinových ručiček. Při změně spínacích kombinací dle Obr. 11 vpravo se bude motor otáčet po směru hodinových ručiček.

### Krokový motor s permanentními magnety

Jedná se o krokový motor s aktivním rotorem, jehož nevyjádřené póly jsou tvořeny permanentní magnety, jejichž polarita se střídá po obvodu rotoru. Statorové vinutí je obvykle dvoufázové buď s vyvedeným středem (unipolární) nebo bez vyvedeného středu (bipolární). Při přepínání fází je nutno měnit také polaritu napětí. Obvyklá velikost kroku se u těchto motorů pohybuje od  $3,6^\circ$  do  $18^\circ$ . Zvýšeného rozlišení je dosaženo buď zvýšením počtu pólů (permanentních magnetů) rotoru, zvýšením počtu fází statoru, nebo způsobem krokování. Na Obr. 12 je zobrazen krokový motor konstrukce can-stack. Jedná se o dvoufázový krokový motor. Rotor je složen ze stejného počtu pólů párů jako má stator. Stator se skládá ze dvou „misk“ které mají dokola póly ve tvaru zubu. Tyto póly jsou mezi sebou posunuty o vzdálenost půl úhlové rozteče mezi miskami. Mezi těmito póly jsou vinutí obou fází. Jakmile vinutím začne téct proud, vytvořené magnetické pole zmagnetuje střídavě póly statoru, rotor se natočí o jeden krok dle polaroty proudu po směru nebo proti směru hodinových ručiček [9].

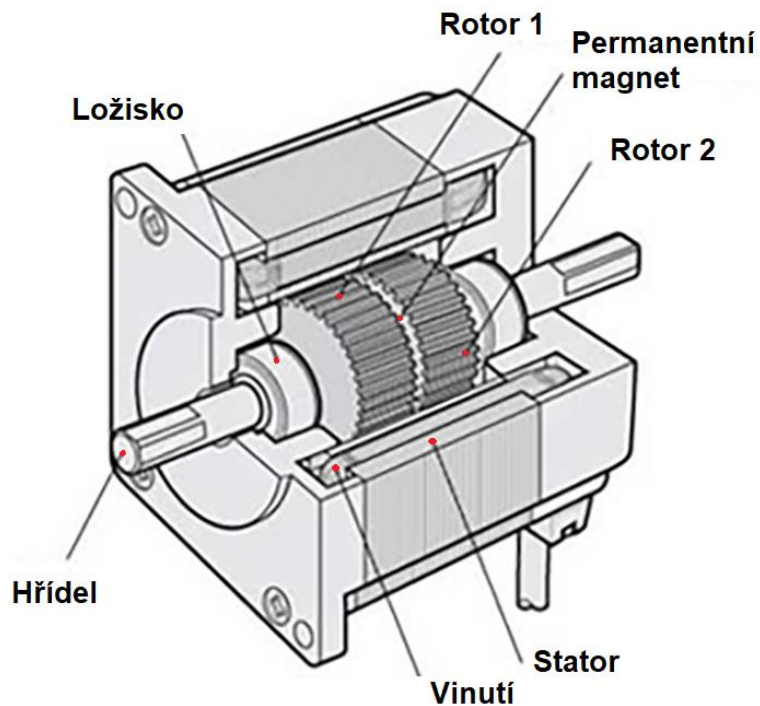


Obr. 12: Konstrukce krokového motoru s permanentními magnety typu can-stack [12]

Krokový motor na Obr. 12 je složen z 24 pólů. Úhlová rozteč je tedy  $360^\circ/24 = 15^\circ$ . Jeden krok má potom velikost poloviny úhlové rozteče, to je  $7,5^\circ$ .

## Hybridní krokový motor

Jedná se o kombinaci reluktančního krokového motoru a krokového motoru s permanentními magnety. Rotor motoru tvoří permanentní magnet a je rozdělen na dvě části ve vertikální rovině. Obě části rotoru mají po obvodu pólové nástavce vzájemně posunuté o polovinu úhlové vzdálenosti pólů. Obě části jsou také opačně zmagnetovány permanentním magnetem. Tyto motory bývají obvykle dvoufázové, ale i tří nebo pěti fázové. Ovládání je stejné jako u krokového motoru s permanentními magnety. Krok je jemnější v rozsahu asi  $0,9 - 3,6^\circ$ . Je možné dosáhnout většího momentu a vyšší provozní frekvence. Jedná se o nejrozšířenější typ krokového motoru [9].



Obr. 13: Řez hybridním krokovým motorem, upraveno z [13]

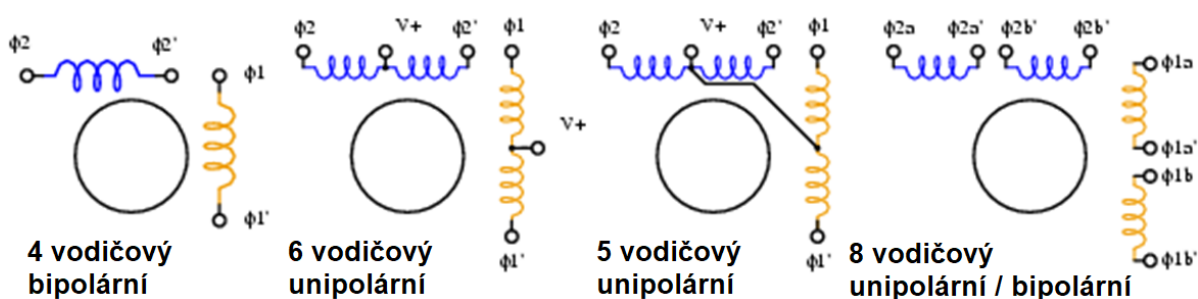
## Uspořádání vinutí

V konstrukci krokových motorů se používají dvě možnosti zapojení vinutí, a to bipolární a unipolární. Je to proto, že u krokových motorů s aktivním rotorem se mění směr proudu ve fázích v průběhu otáčení.

V případě bipolárního vinutí má každá fáze pouze jedno vinutí na jednom nebo více pól párech a oba konce jsou vyvedeny ven z motoru. Změna směru proudu se provádí změnou polarity napájecího napětí vinutí. Z pohledu řízení to znamená, že pro každou fázi krokového motoru potřebujeme jeden H-můstek.

Oproti tomu v případě unipolárního vinutí má každá fáze vinutí dvě, navinuta bifilárně. Konec jednoho vinutí je pak spojen se začátkem druhého a vyveden spolu se zbylými dvěma konci vinutí. Tento společný vývod je připojen k napájecímu napětí a zbylé konce vinutí jsou pak spínány k zemi. Dle toho, která polovina vinutí je aktivní, se mění směr magnetického toku. Proud ovšem protéká vždy jen jedním směrem – proto unipolární vinutí. Z pohledu elektrického zapojení je obvod řízení unipolárního motoru jednodušší, potřebujeme pouze dva tranzistory na jednu fázi namísto čtyřech v případě bipolárního vinutí.

Existují i typy krokových motorů s vyvedenými středy vinutí, případně oběma konci vinutí v případě osmi vývodového typu, které je možno zapojit jak bipolárně, tak unipolárně.

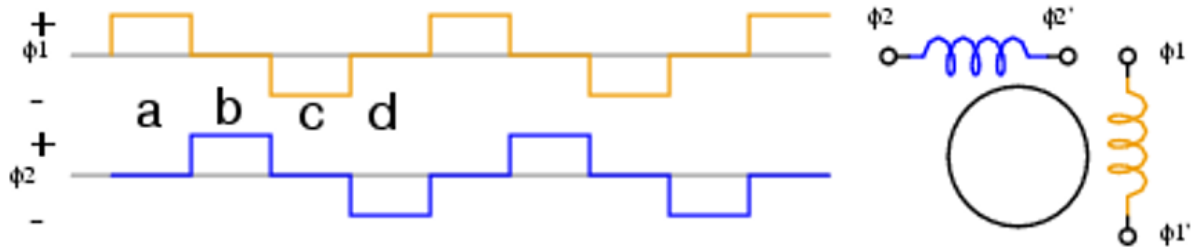


Obr. 14: Typy vinutí krokových motorů, převzato z [9]

Unipolární krokové motory jsou nevýhodné kvůli většímu odporu vinutí, které je navinuto tenčím vodičem, protože jsou navinuta dvě vinutí a ne jedno. Krokový motor bude mít menší moment při stejném napájecím napětí, protože fázemi protéká menší proud. Proud nelze navýšit z důvodu výkonové ztráty [9].

### 1.3.3 Způsoby krokování

Existuje více typů krokování. Plný krok, jednofázové řízení, kdy je v jednom okamžiku aktivní pouze jedna fáze a během jednoho elektrického cyklu se vystřídají 4 spínací kombinace, viz Obr. 15. Nevýhodou tohoto řízení je menší moment oproti dvoufázovému řízení.



<b>Polarita fáze <math>\phi_1</math></b>	+		-	
<b>Polarita fáze <math>\phi_2</math></b>		+		-
<b>Spínací kombinace</b>	1	2	3	4

Obr. 15: Průběh proudu fázemi a tabulka spínacích kombinací pro plný krok, jednofázové řízení, převzato z [9]

Dalším typem je dvoufázové řízení, kdy jsou v každém okamžiku aktivní obě fáze. Celkový moment je větší díky vektorovému součtu magnetických toků obou fází. Velikost kroku se nemění [9]. V Tab. 2 níže jsou uvedeny spínací kombinace.

<b>Polarita fáze <math>\phi_1</math></b>	+	+	-	-
<b>Polarita fáze <math>\phi_2</math></b>	-	+	+	-
<b>Spínací kombinace</b>	1	2	3	4

Tab. 2: Tabulka spínacích kombinací plný krok, dvoufázové řízení [9]

Druhá možnost je poloviční krok, což je kombinace jednofázového a dvoufázového řízení s plným krokem. Takto lze dosáhnout zdvojnásobení počtu kroků na otáčku a lepšího rozlišení v úhlu [9]. Spínací kombinace jsou v Tab. 3 níže.

<b>Polarita fáze <math>\phi_1</math></b>	+	+		-	-	-		+
<b>Polarita fáze <math>\phi_2</math></b>		+	+	+		-	-	-
<b>Spínací kombinace</b>	1	2	3	4	5	6	7	8

Tab. 3: Tabulka spínacích kombinací poloviční krok [9]

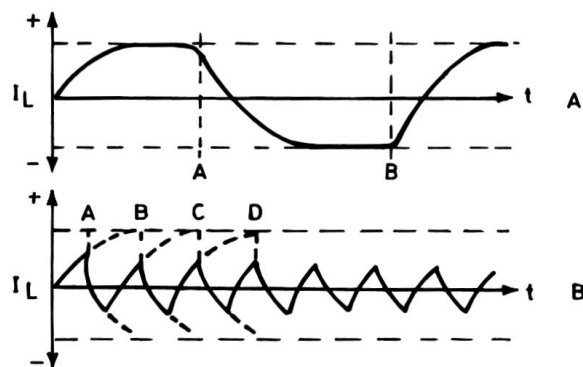
Reverzace otáčení motoru ve všech režimech se provede procházením spínacích kombinací v opačném pořadí.

### 1.3.4 Způsoby napájení krokových motorů

Existují dvě možnosti, jak napájet krokový motor. První možností je u malých krokových motorů ze zdroje napětí, druhou pak ze zdroje proudu s PWM regulací u krokových motorů větších.

#### Napájení konstantním napětím

Maximální hodnota proudu vinutím motoru je omezena ohmickým odporem vinutí  $R_L$ . Čas, za který proud vinutím dosáhne svoji maximální hodnotu, je určen časovou konstantou vinutí  $\tau = L/R_L$ , kde  $L$  je indukčnost vinutí. Pokud je frekvence řídicích pulzů příliš vysoká, proud vinutím nemůže dosáhnout své maximální hodnoty kvůli časové konstantě  $\tau$ . Tím pádem dochází ke zmenšení momentu krokového motoru. Tento problém lze řešit zvýšením nárůstu proudu přidáním rezistoru do série s vinutím krokového motoru a zvýšením napájecího napětí. Tím se časová konstanta  $\tau$  zmenší. Nevýhodou jsou ztráty na předřadném odporu, lze použít jen u motorů s malým výkonem [9].



Obr. 16: Průběh proudu bipolárním vinutím s nízkou (A) a vysokou (B) frekvencí krokování [9]

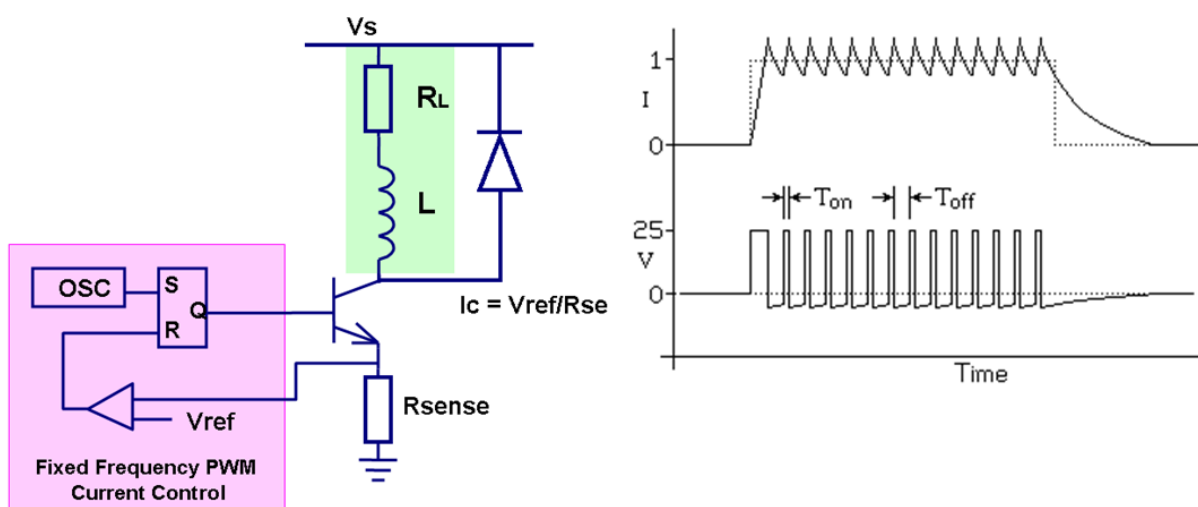
#### Napájení konstantním proudem

Při tomto způsobu napájení je krokový motor napájen budičem, který pracuje s konstantní střední hodnotou proudu udržovanou obvykle pomocí PWM regulace. Výhodou toho řešení jsou menší ztráty, které vznikají pouze na spínacích tranzistorech. Budiče je možné realizovat buď s pevnou spínací frekvencí, kde se střední hodnota proudu reguluje střídou, nebo s proměnnou frekvencí a pevnou dobou vypnutí  $t_{off}$  kde se střední hodnota proudu udržuje konstantní pomocí změny frekvence spínání.

V případě budiče s pevnou spínací frekvencí je krokový motor napájen z vyššího napětí, než je jeho jmenovité napětí. Tím je docíleno rychlejšího nárůstu proudu vinutím při sepnutí tranzistoru.

Při překročení nastaveného proudu vinutím se pomocí úbytku na snímacím rezistoru tranzistor uzavře. K sepnutí tranzistoru dojde při další periodě oscilátoru OSC a cyklus se opakuje.

Velikost zvlnění proudu závisí na indukčnosti a odporu vinutí, na frekvenci PWM, na střídě spínání a na rychlosti poklesu proudu ve fázi vypnutí. Ani při tomto způsobu řízení však nelze zvyšovat spínací kmitočet nad určitou mez kvůli zvyšujícím se spínacím ztrátám tranzistorů. Od určité frekvence již také proud nestihne narůst na maximální hodnotu. Oproti napěťovému řízení se ale zvýší frekvence krokování, kdy ještě nedochází k poklesu momentu krokového motoru [9].



Obr. 17: Budič s PWM regulací proudu a průběhy proudu vinutím, napětí na vinutí [9]

## 2 PWM generátor s bipolárním výstupem

### 2.1 Požadavky na tester

Funkční požadavky vycházejí zejména ze stávajících produktů zadavatele práce. Univerzální hardware by měl fungovat s různými variantami software. Mohou existovat tzv. low-cost varianty podle výbavy testeru, aby byla pokryta co největší část poptávky na trhu. Jedna DPS by měla být navržena tak, aby bylo možné měnit typ testeru podle osazení potřebných komponent. Dalším aspektem návrhu je možnost rozšíření do budoucna například moduly připojenými na I<sup>2</sup>C sběrnici nebo využití navrhovaného externího A/D převodníku pro měření dalších neelektrických veličin jako je tlak, teplota, průtok atp. Pro řízení byla vybrána platforma mikroprocesorů AVR od firmy Microchip Technology.

Tester umožňuje připojit všechny typy akčních členů s DC motorem a solenoidy uvedené v Tab. 1 s výjimkou vstřikovacích ventilů common rail, které vyžadují vysoký budicí proud. Také prozatím nelze testovat na tomto typu testeru benzinové a CNG vstřikovací ventily, u kterých je použito impulzní řízení (frekvence a doba pulzu, nikoliv střída). Testování indukčních cívek, zapalovacích modulů je možné jen za určitých podmínek.

#### Technické parametry

- Napájecí napětí max. 18 V / 28 V
- Maximální trvalý výstupní proud<sup>1</sup> 4 A
- Signálový výstup, obdélníkový průběh 5 V / 12 V max. 50 mA  
(pro simulaci výstupu z akčního členu)
- Rozsah střídy 0-100 %
- Rozsah frekvence 10 Hz – 10 kHz
- Výkonový unipolární výstup (otevřený kolektor)
  - Vypínatelná antiparalelní dioda
- Bipolární generátor PWM se silovým výstupem (H-můstek)
- Unipolární generátor PWM se silovým výstupem (H-můstek)
- Nadproudová ochrana proti zkratu
- Ochrana proti přehřátí

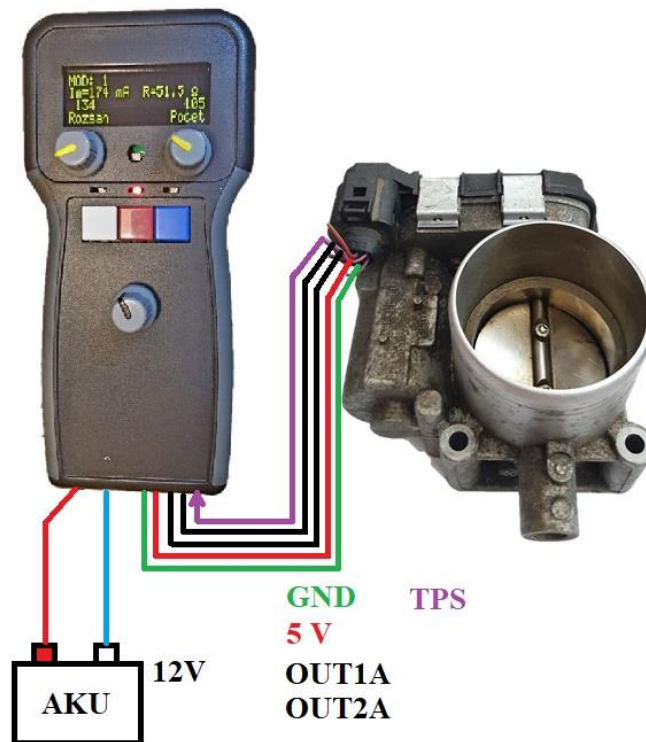
---

<sup>1</sup> Omezeno tavnou pojistkou 4 A

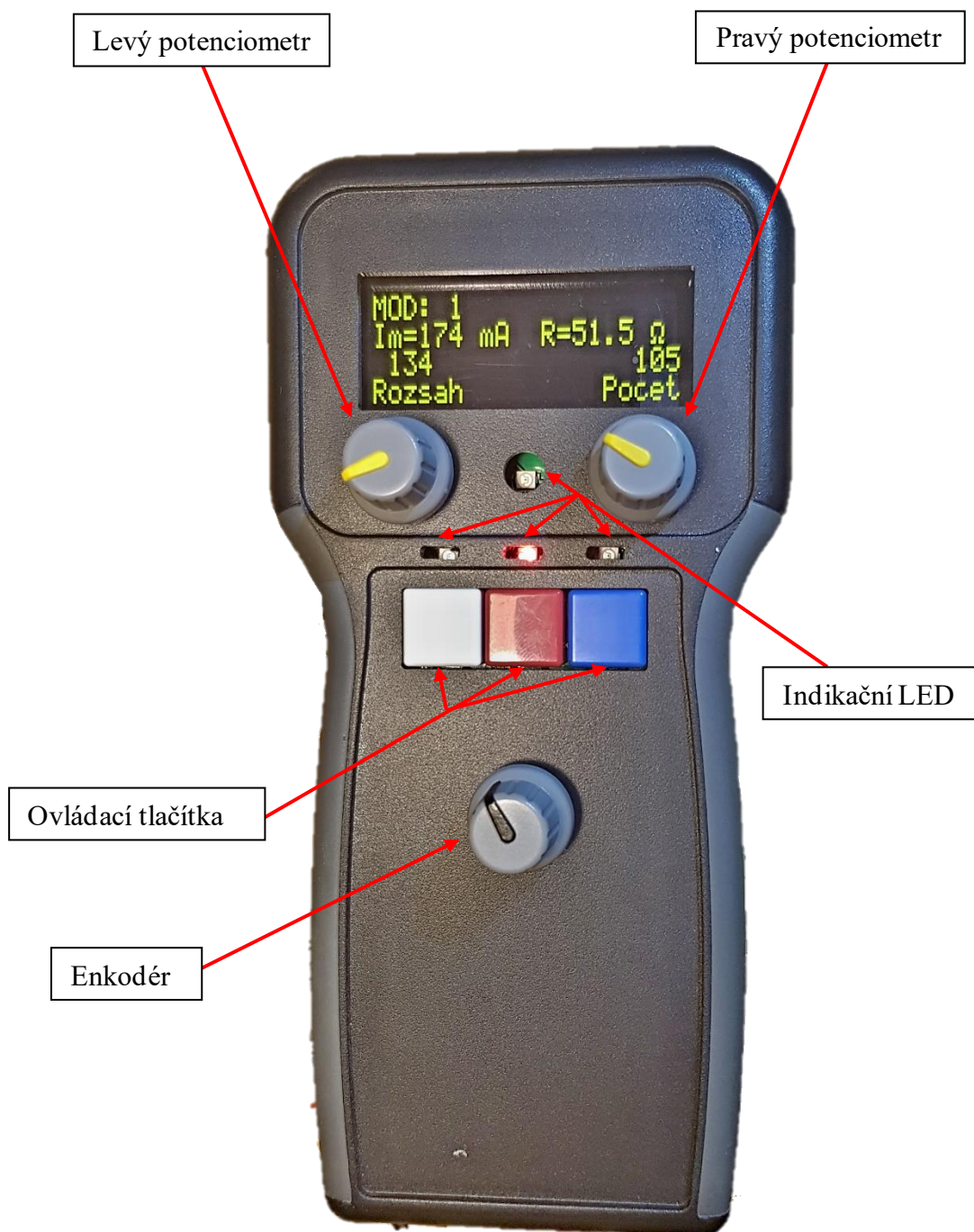


### Uživatelské funkce:

- Obsluha tlačítka, potenciometry a enkodérem
- Uživatelské rozhraní na displeji se zobrazením relevantních hodnot
- Parametrizace nastavení v menu přístroje
- Měření napětí ze snímače polohy akčního členu
- Nastavení omezení proudu zátěží
- Měření proudu zátěží
- Automatické režimy testování s proměnlivou střídou, frekvencí
- Zpětnovazební regulace polohy akčního členu



Obr. 18: Připojení škrtící klapky k testeru

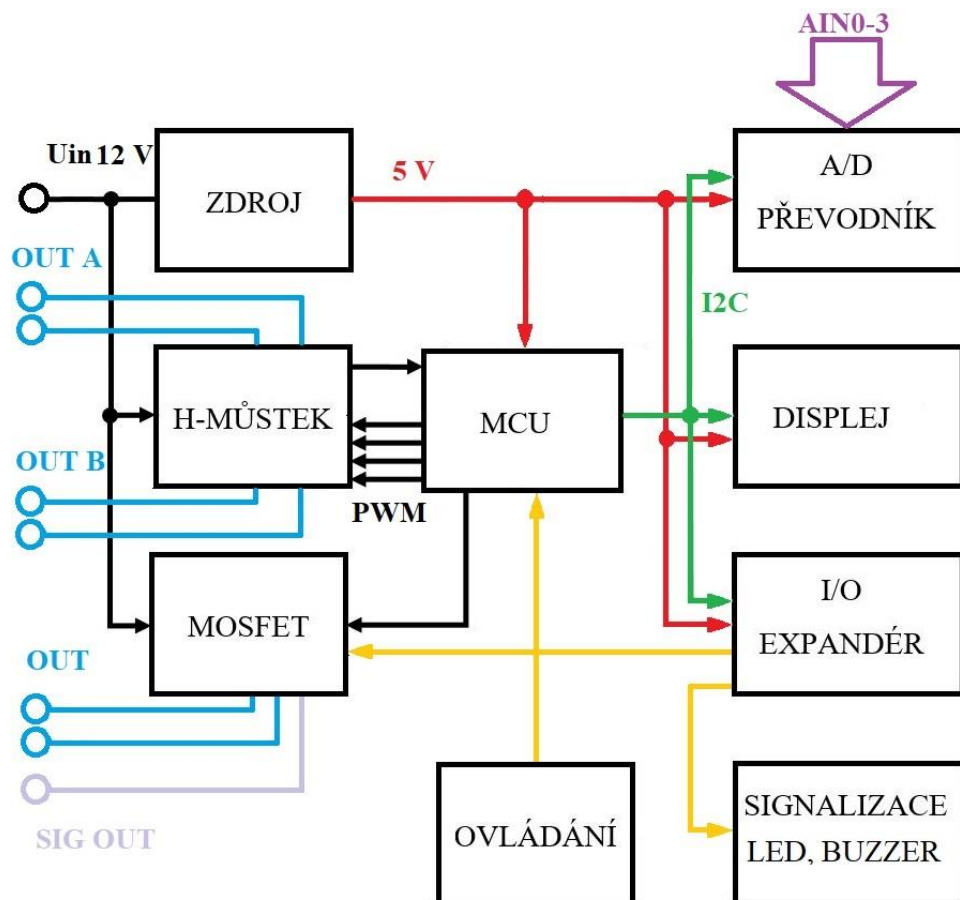


Obr. 19: Vzhled testeru

Bližší popis funkcí ovládacích prvků je v kapitole 2.4.4 Ovládací prvky. Připojovací zdířky budou nejspíše umístěny v horní části i dolní části testeru.

## 2.2 Popis Hardwaru

Na počátku řešení diplomové práce existoval hardware, který bohužel nevyhovoval požadavkům na nový tester. Proto byly nutná inovace testeru, která si vyžádala několik změn na úrovni hardwaru. Zadavatel práce následně na základě společných konzultací s vedoucím práce a mnou navrhl nový hardware. Kvůli rozsáhlým změnám bylo nutné také změnit chassis testeru, ve stávajícím již nebyl prostor pro displej. Jako náhrada dvou stávajících mikrokontrolerů byl vybrán MCU ATmega328. Dále byl nahrazen dvojitý H-můstek z typu L298 na L6207 a v návrhu desky plošných spojů je počítáno i s dvojitým H-můstkem DRV8874<sup>2</sup>. Dále byl přidán alfanumerický OLED displej 4x20 řádků, rotační enkodér a počet ovládacích tlačítek byl navýšen na tři. Kvůli nedostatku volných pinů v mikrokontroleru je tester osazen expandérem osmi vstupů/výstupů PCF8574. Nově je také použit externí čtyřkanálový 12bitový A/D převodník ADS1015. Displej, A/D převodník a expandér komunikují s MCU po sběrnici I<sup>2</sup>C. Zdrojová část je nezměněna. Diplomová práce neobsahuje detailní elektrické schéma, jelikož si jej zadavatel nepřije zveřejnit, protože obsahuje jeho technické know-how. Proto je níže uvedeno alespoň blokové schéma.



Obr. 20: Blokové schéma generátoru PWM s bipolárním výstupem

<sup>2</sup> H-můstek DRV8874 nebyl v rámci této práce implementován.

### **2.2.1 Mikrokontroler**

Byl použit 8bitový mikrokontroler ATmega328P z rodiny AVR od výrobce Atmel. V současné době by bylo možné použít ve stejné cenové hladině mnohem výkonnější MCU s více periferiemi (např. čítače/časovače) a větší pamětí. Tento mikrokontroler je však široce používaný (např. na platformě Arduino). To umožňuje použít mnoho knihoven a ušetřit tak někdy čas potřebný na vývoj. Dalším důvodem, proč byl zvolen právě tento MCU, jsou znalosti nabyté při studiu v rámci předmětu Mikroprocesorová technika. Níže jsou uvedeny některé základní parametry.

#### **Paměťový prostor**

- 32 KB FLASH pro program
- 2 KB SRAM interní paměť dat
- 1 KB EEPROM

#### **Periferie**

- 10bitový osmi kanálový A/D převodník
- TWI (I<sup>2</sup>C) komunikační sběrnice
- 2 8bitové časovače/čítače, 1 16bitový časovač/čítač
- 6 PWM kanálů
- SPI sériová sběrnice
- Sériové rozhraní USART

#### **Vstupy/Výstupy**

- 23 programovatelných vstupů/výstupů
- 32 vývodové pouzdro TQFP

#### **Elektrické parametry**

- Provozní napětí 2,7 – 5,5 V, maximální napětí 6 V
- Maximální stejnosměrný proud jednoho pinu 40 mA
- Maximální stejnosměrný proud celého pouzdra 200 mA

<b>Pin MCU</b>	<b>Číslo pinu MCU</b>	<b>Funkce pinu MCU</b>	<b>Funkce</b>
PB0	12		N/A
PB1 (OC1A)	13	Výstup PWM	H-můstek IN1A
PB2 (OC1B)	14	Výstup PWM	H-můstek IN2A
PB3 (OC2A)/ MOSI	15	Výstup PWM / Sběrnice SPI	H-můstek V <sub>REF</sub>
PB4 (MISO)	16	Sběrnice SPI	MISO
PB5 (SCK)	17	Vstup pro tlačítko 1/ Sběrnice SPI	SCK / Tlačítko 1
PC0	23	Výstup	H-můstek EN <sub>A</sub> , EN <sub>B</sub>
PC1	24	Vstup pro tlačítko 2	Tlačítko 2
PC2 (ADC2)	25	Vstup převodníku kanál 2	H-můstek SENSE A
PC3 (ADC3)	26	Vstup převodníku kanál 3	H-můstek SENSE B
PC4 (SDA)	27	Sběrnice I <sup>2</sup> C	SDA
PC5 (SCL)	28	Sběrnice I <sup>2</sup> C	SCL
ADC6	19	Vstup převodníku kanál 6	Potenciometr 2
ADC7	22	Vstup převodníku kanál 7	Potenciometr 1
PD0 (RXD)	30	Výstup PWM / Komunikace UART	H-můstek IN2B / UART
PD1 (TXD)	31	Výstup PWM / Komunikace UART	H-můstek IN1B / UART
PD2 (INT0)	32	Vstup /přerušení	H-můstek OCDA
PD3 (INT1)	1	Vstup /přerušení	H-můstek OCDB
PD4	2	Vstup pro tlačítko enkodéru	ENCODER SW
PD5	9	Vstup pro tlačítko 3	Tlačítko 3
PD6 (PCINT22)	10	Vstup stopa A enkodéru	ENKODER vlevo
PD7 (PCINT23)	11	Vstup stopa B enkodéru	ENKODER vpravo

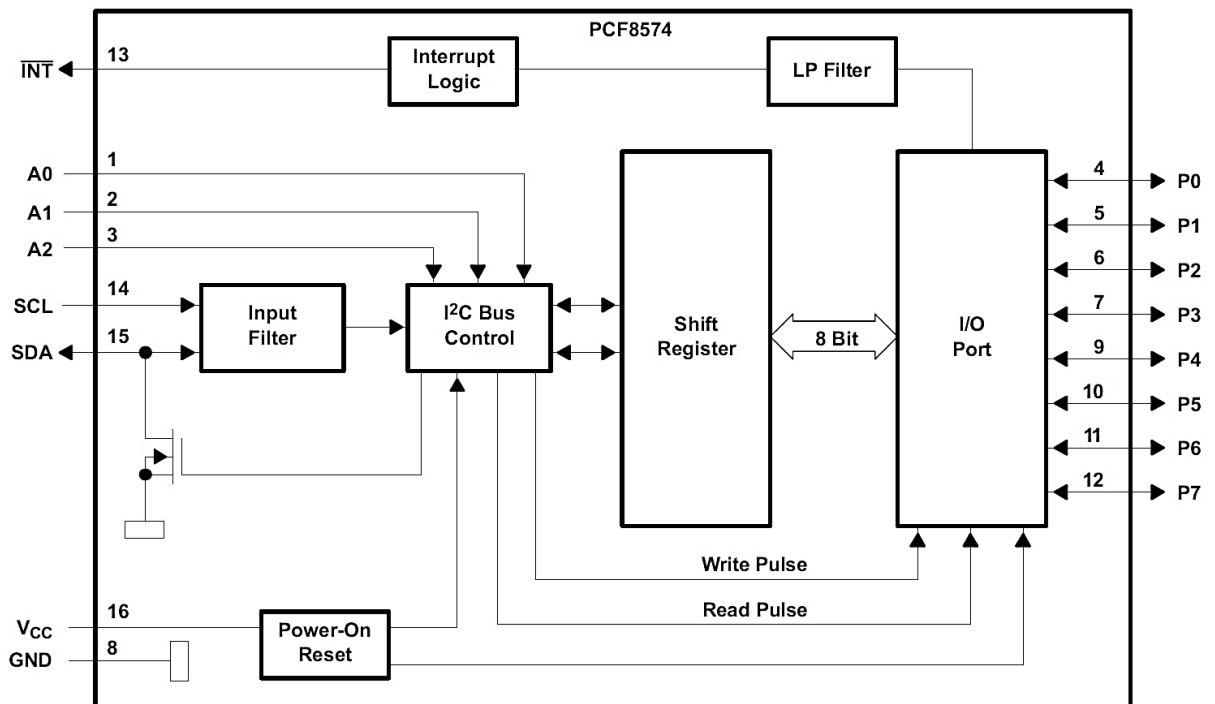
Tab. 4: Použité piny MCU a jejich funkce

## 2.2.2 Expandér vstupů/výstupů

Protože MCU nemá dostatek výstupních pinů pro další použité funkce, byl do zařízení implementován 8bitový I/O expandér PCF8574 komunikující po sběrnici I<sup>2</sup>C. Je možno využít každý z osmi pinů buď jako vstupní, nebo jak jako výstupní. Výstupní piny jsou spínané proti zemi, ve vypnutém stavu je na pinu kladné napětí přes interní pull-up rezistor. Rozdíl je jen v možném odebíraném proudu. Při spínání zátěže proti zemi je možno odebírat až 25 mA na pin, v případě spínání proti kladnému napětí ovšem pouze 1 mA. V testeru je využito všech 8 pinů jako výstupy. Adresa po I2C sběrnici se nastavuje na vstupech A0, A1, A2 a uzemněním všech těchto vstupů se nastaví na 0x40 hexadecimálně [22].

P0	P1	P2	P3	P4	P5	P6	P7
Piezoměnič	LED3	Signál výstup	Relé antiparalelní dioda	LED1 zelená	LED 1 červená	LED2	LED4

Tab. 5: Zapojení výstupních pinů expandéru PCF8574



Obr. 21: Zjednodušené blokové schéma I/O expandéru PCF8574 [22]

### 2.2.3 Externí A/D převodník

Kvůli nedostatku analogových vstupů interního A/D převodníku MCU bylo nutno přidat externí A/D převodník. Dalším důvodem je také vyšší přesnost oproti internímu A/D převodníku MCU. Použitý delta-sigma A/D převodník ADS1015 je 12bitový, má čtyři kanály a je obsluhován po sběrnici I<sup>2</sup>C. Maximální rychlost vzorkování je 3300 SPS. Umožňuje diferenciální měření, kontinuální měření a samostatný převod. V testeru je použit pouze režim samostatného převodu (single ended). A/D převodník má programovatelný rozsah (zesílení) vstupních napětí, viz Tab. 6, která se nesmí překročit, jinak hrozí zničení obvodu. V Tab. 6 je zeleně vyznačen použitý rozsah. V programu je použita vzorkovací rychlost 1600 sps. A/D převodníku. Tento typ A/D převodníku byl vybrán kvůli dostupnosti knihovny pro obsluhu, nebylo nutno napsat vlastní knihovnu a uspořil se tak čas pro vývoj software. Dva vstupy AIN0 a AIN1 A/D převodníku jsou použity pro měření zpětné vazby z akčního členu (např. ze škrtkící klapky). Vstup AIN2 je použit pro měření napájecího napětí testeru, vstup AIN3 je rezervní, hardwarově připravený pro doplňkové měření odporu, v současné verzi softwaru není použitý. Další detaily jsou uvedeny v kapitole 2.4.6 Externí A/D převodník.

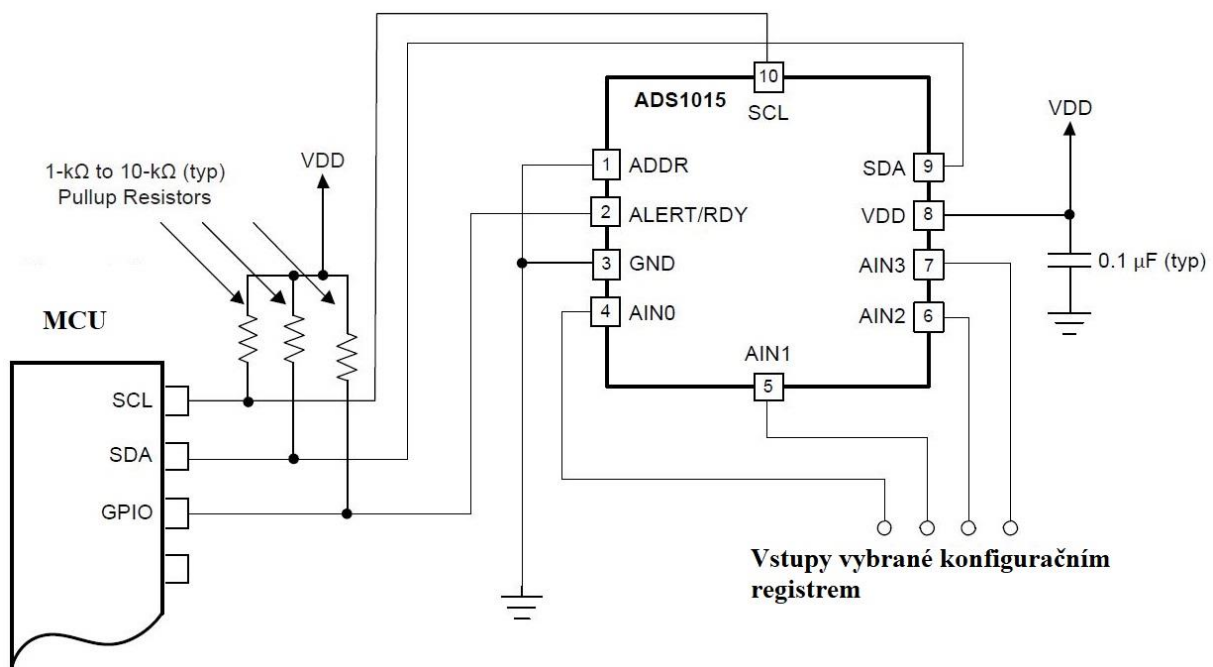
Rozsah	rozlišení na bit
±6,144 V	3 mV
±4,096 V	2 mV
±2,048 V	1 mV
±1,024 V	0,5 mV
±0,512 V	0,25 mV
±0,256 V	0,12 mV

Tab. 6: Rozsahy vstupních napětí převodníku ADS1015 a odpovídající rozlišení<sup>3</sup>

Parametr	Min.	Max.
Napájecí napětí [V]	-0,3	7
Napětí na vstupech [V]	GND - 0,3	V <sub>cc</sub> + 0,3
Napětí digitálních vstupů [V]	GND - 0,3	5,5
Vstupní proud [mA]	-10	10
Rozsah vstupních napětí	±0,256	±6,144
Vstupní impedance [MΩ]	3	100

Tab. 7: Vybrané elektrické parametry A/D převodníku ADS1015

<sup>3</sup> Záporná napětí lze měřit pouze v diferenciálním měření mezi dvěma vstupy



Obr. 22: Typické zapojení A/D převodníku ADS1015



## 2.2.4 Displej

V testeru je použit alfanumerický OLED displej EA OLEDM204, který má 4x20 znaků. Displej má integrovaný řadič SSD1311, který umožňuje komunikaci přes I<sup>2</sup>C nebo SPI rozhraní. Vzhledem k tomu, že sběrnice SPI potřebuje více pinů MCU, a displej není jedinou periférií připojenou k MCU, bylo zvoleno rozhraní I<sup>2</sup>C. Displej potřebuje dvě rozdílná napájecí napětí 12 V a 5 V. Po komunikaci s výrobcem displeje bylo možné nakonec úspěšně propojit displej s MCU na sběrnici I<sup>2</sup>C s 5 V logikou. Displej má totiž nativně vnitřní logiku 3,3 V. Z důvodu striktního maxima napájecího napětí displeje 13 V (v palubní síti vozidla může být až 15 V) je pro napájení displeje použit lineární stabilizátor 10 V. Ukázalo se, že pro displej to není problém, a jeho čitelnost je přesto dobrá. Rozměry aktivní plochy displeje jsou 47x17 mm. Displej byl vybrán hlavně kvůli velmi malé výšce 4,7 mm i s vývody. Velikost jednoho znaku je 5x8 bodů, velikost jednoho bodu je 0,5x0,5 mm.

Parametr	Min.	Max.
Napájecí napětí OLED [V]	12	13
Napětí pro logiku 5 V [V]	4,4	5,5
Pracovní teplota [°C]	-40	80
Pozorovací úhel [°]	160	-
Svítivost [cd/m <sup>2</sup> ]	140	-
Kontrast [-]	2000:1	-
Životnost [h]	50000	-

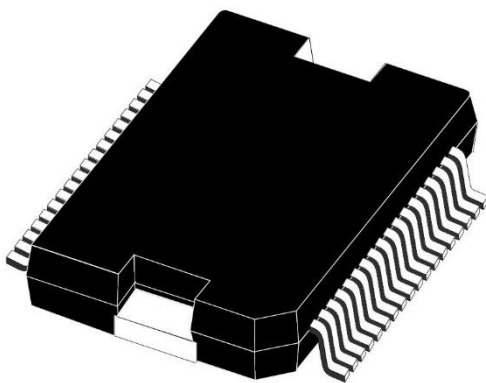
Tab. 8: Vybrané parametry displeje EA OLEDM204 [16]



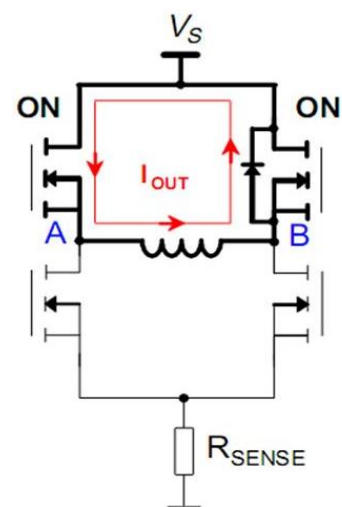
Obr. 23: Modul alfanumerického displeje EA OLEDM204 [16]

## 2.2.5 H-můstek

V diskuzi, jaký H-můstek bude použitý, byly navrženy v první řadě obvody dvojitých H-můstků, u kterých byl předpoklad pro splnění proudových nároků. Zároveň byly upřednostňovány obvody s nízkým  $R_{DSON}$  a výhodným řízením z pohledu MCU. Kvůli preferenci nižší ceny H-můstku byl však vybrán obvod L6207 v pouzdře PowerSO36, viz Obr. 24. Ten se ukázal jako nevhodný z důvodu velkých výkonových ztrát a omezených možností jeho chlazení. Z pohledu technického řešení je mnou preferován obvod DRV8874. Obvod L6207 obsahuje dva plnohodnotné H-můstky, které lze použít buď pro řízení jednoho či dvou DC motorků, nebo jednoho krokového motorku. Obvod obsahuje mimo těchto můstků ještě obvody pro omezení výstupního proudu a detekci maximálního nadproudu, teplotní pojistku. Omezení proudu do zátěže lze nastavit na pinu  $VREF_A$  a  $VREF_B$  přivedením externího napětí, kde interní komparátor porovnává napětí mezi těmito vstupy a pinem  $SENSE_A$ ,  $SENSE_B$ , na které je připojen externí bočník pro měření proudu. Jakmile je napětí na bočníku větší než  $VREF$ , jsou vypnuty spodní tranzistory v H-můstku a proud protéká jen mezi zátěží a vrchními tranzistory v takzvaném slow decay módu. V tu chvíli není možné měřit proud zátěží z úbytku na bočníku, protože tam žádný neteče, viz Obr. 25 a Obr. 71. Nevýhodou tohoto omezení je, že informace o aktivaci omezování proudu není nijak vyvedena z pouzdra. Nelze na ni tedy v MCU reagovat. Nadproudová a teplotní ochrana  $I_{MAX}$ , resp.  $T_{JMAX}$  je nastavena pevně a nelze ji změnit. Obvod L6207 nemá dedikované vývody s příznaky vybavení nadproudové a teplotní ochrany. O vybavení nadproudové nebo teplotní ochrany H-můstku A nebo H-můstku B je informováno na pinu  $EN_A$  nebo  $EN_B$ , který se při aktivaci uzemní, viz Obr. 27. Mezi MCU a piny  $EN_A$ ,  $EN_B$  je nutno vložit do série rezistor a MCU musí porovnávat požadovaný stav vstupů  $EN_A$ ,  $EN_B$  s jejich skutečným stavem.



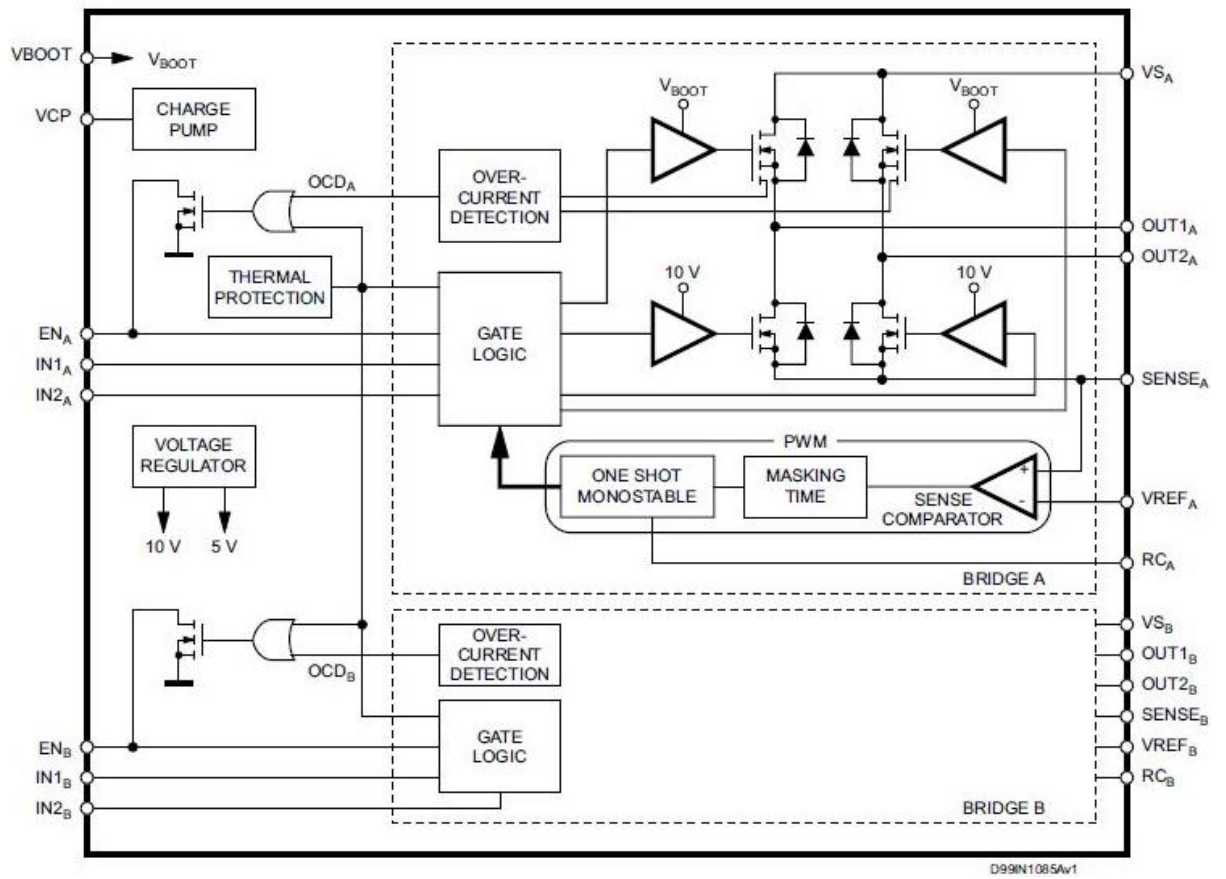
Obr. 24: Tvar pouzdra PowerSO36 IO L6207 [20]



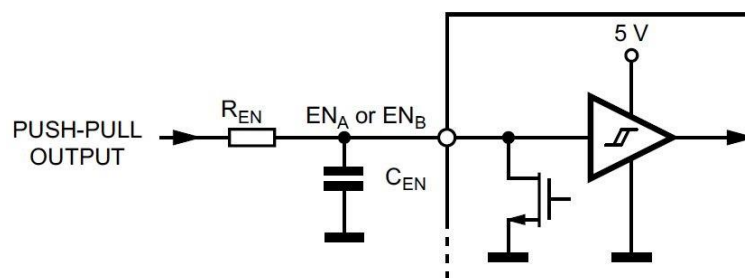
Obr. 25: Slow decay recirkulace proudu [9]

Symbol	Parametr	Min.	Typ.	Max.
V <sub>S</sub> [V]	Napájecí napětí	8	-	52
R <sub>DSON</sub> [Ω]	Odpor Drain-Source v sepnutém stavu [T <sub>J</sub> = 25 °C]	-	0,34	0,4
I <sub>OUT</sub> [A]	Výstupní proud RMS	-	-	2,8
I <sub>MAX</sub> [A]	Špičkový proud (limit nadproudové ochrany)	4	5,6	7,1
f <sub>SW</sub> [kHz]	Spínací frekvence	-	-	100
V <sub>REF</sub> [V]	Referenční napětí pro regulaci proudu zátěží	-0,1	-	5
V <sub>SENSE</sub> [V]	Rozsah napětí na pinech SENSE	-1	-	1
V <sub>EN</sub> [V]	Rozsah napětí vstupy EN <sub>A</sub> , EN <sub>B</sub>	-0,3	-	7
T <sub>JMAX</sub> [°C]	Maximální teplota přechodu	-	165	-
t <sub>RISE</sub> [ns]	Doba sepnutí při 2,8 A, odporová zátěž	40	-	250
t <sub>FALL</sub> [ns]	Doba vypnutí při 2,8 A, odporová zátěž	40	-	250
R <sub>THJC</sub> [°C/W]	Tepelný odpor přechod-pouzdro	-	-	1
R <sub>THJA</sub> [°C/W]	Tepelný odpor přechod-prostředí pro oboustrannou DPS FR4 tloušťky Cu vrstvy 35 μm a 6 cm <sup>2</sup> chladič plochy s 16 prokvy	-	-	15

Tab. 9: Vybrané parametry H-můstku L6207 v pouzdře PowerSO36 [20]



Obr. 26 : Vnitřní blokové schéma H-můstku L6207 [20]



Obr. 27: Způsob připojení vstupů  $EN_A$ ,  $EN_B$  k MCU [20]

## 2.2.5.1 Spínací režimy H-můstku

Tab. 10: Pravdivostní tabulka spínacích kombinací H-můstku [20]

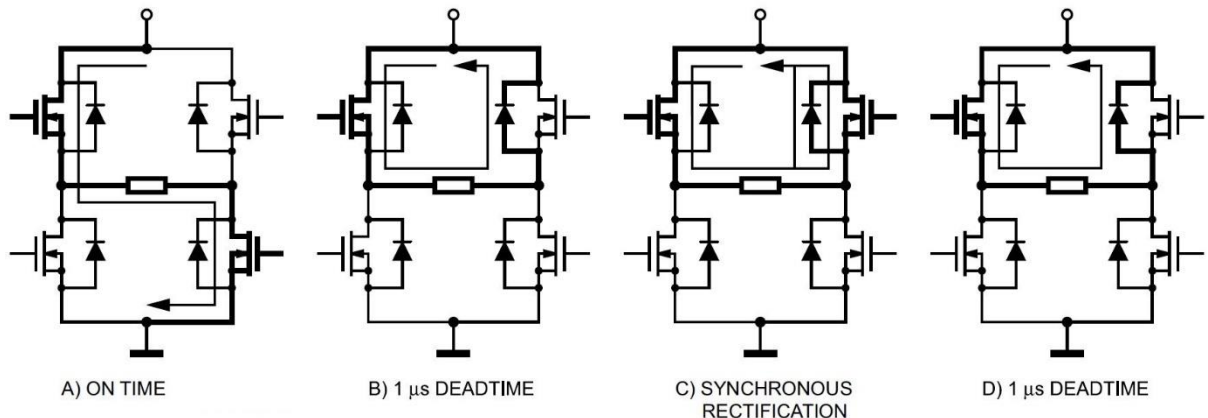
Inputs			Outputs		Description <sup>(1)</sup>
EN	IN1	IN2	OUT1	OUT2	
L	X <sup>(2)</sup>	X <sup>(2)</sup>	High Z <sup>(3)</sup>	High Z <sup>(3)</sup>	Disable
H	L	L	GND	GND	Brake mode (lower path)
H	H	L	Vs	GND (Vs) <sup>(4)</sup>	Forward
H	L	H	GND (Vs) <sup>(4)</sup>	Vs	Reverse
H	H	H	Vs	Vs	Brake mode (upper path)

<sup>1</sup> Platí pouze v případě, že je připojena zátěž na výstupy OUT1 a OUT2

<sup>2</sup> nezáleží na stavu

<sup>3</sup> Výstup ve vysoké impedanci

<sup>4</sup> Při  $t_{ON}$  je na výstup uzemněn, při  $t_{OFF}$  je na výstupu Vs (kladné napětí)

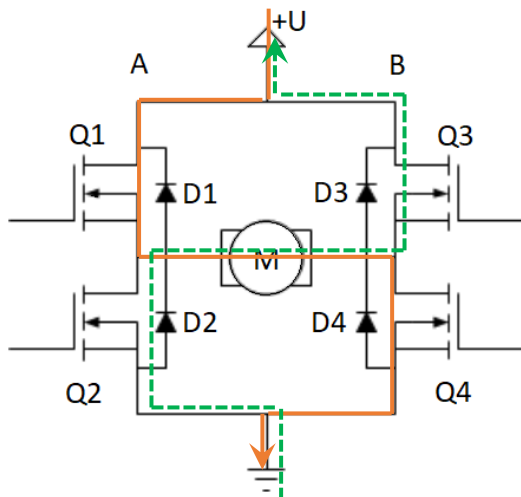


Obr. 28: Stav H-můstku v režimu slow decay

Na Obr. 28 Jsou zobrazeny stavy H-můstku ve slow decay režimu. V následujících dvou kapitolách budou popsány použité režimy spínání H-můstku z pohledu hardwaru.

### 2.2.5.1.1 Locked anti-phase režim

Tento režim je nastavený jako výchozí po zapnutí testeru. Na oba dva vstupy H-můstku IN1A a IN2A je přiváděn vzájemně inverzní signál PWM. Vstup EN<sub>A</sub>, resp. EN<sub>B</sub> je trvale zapnutý log. úrovní H. Protože jsou oba signály PWM z výstupu OC1A a OC1B vůči sobě inverzní, dochází střídavě ke spínání obou polovin H-můstku v diagonále. Jsou možné dva způsoby spínání tranzistorů, viz Tab. 11 níže.

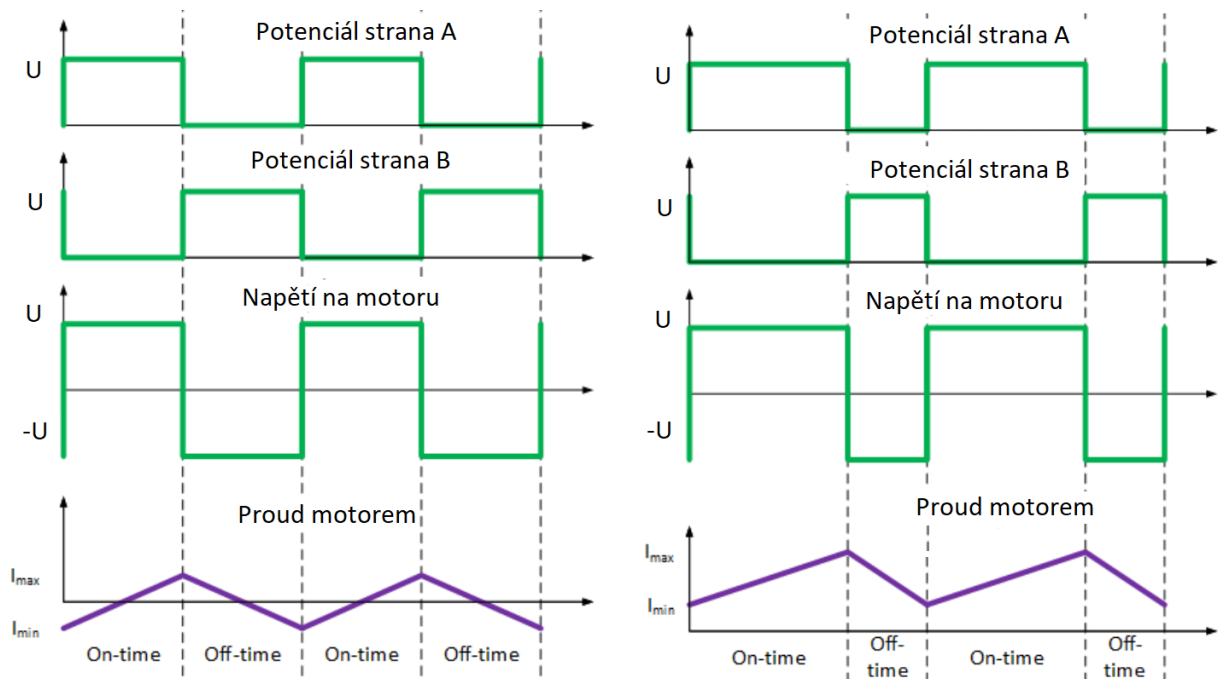


Obr. 29: Spínací režim locked anti-phase H-můstku a toky proudu [24]

Kombinace 1				
	Q1	Q2	Q3	Q4
Stav zapnuto	1	0	0	1
Stav vypnuto	0	1	1	0
Kombinace 2				
Stav zapnuto	0	1	1	0
Stav vypnuto	1	0	0	1

Tab. 11: Spínací kombinace H-můstku v režimu locked anti-phase

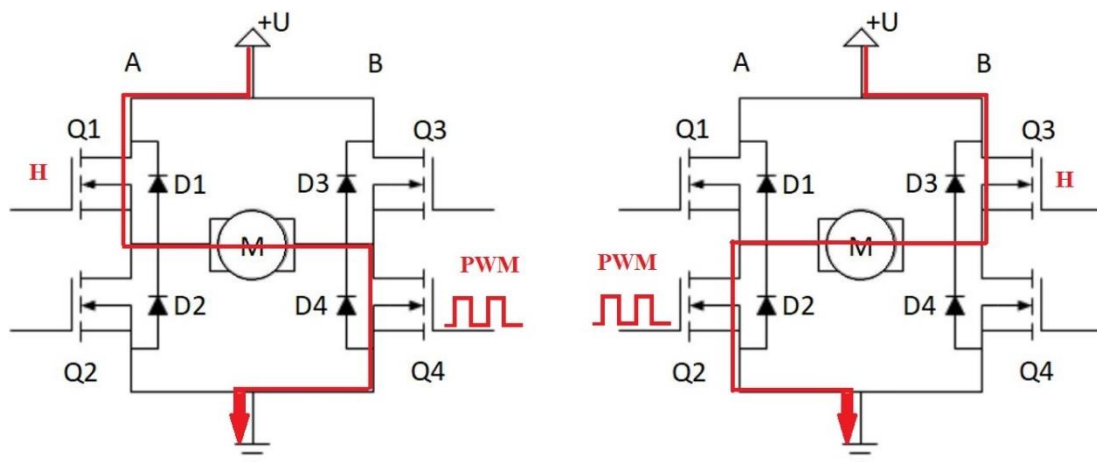
Pokud je střída frekvence PWM nastavena na 50 %, výsledná střední hodnota napětí na motoru bude nulová, motor se netočí. Jakmile se střída PWM zvýší nad 50 %, motor se začne otáčet na jednu stranu a při snížení střidy PWM pod 50 % se motor bude točit na stranu druhou. Nevýhodou tohoto režimu je to, že pokud se motor netočí, nejedná se ve skutečnosti o statický stav. Pokud by došlo k poruše na jednom z řídicích kanálů PWM, motor by se mohl nekontrolovatelně roztočit na jednu nebo na druhou stranu. Na Obr. 30 jsou zobrazeny průběhy proudu a napětí při střídě PWM 50 % a při střídě větší jak 50 %.



Obr. 30: Vlevo průběh proudu a napětí na motoru střída PWM 50 %, vpravo střída PWM > 50 %. Locked anti-phase řízení. [24]

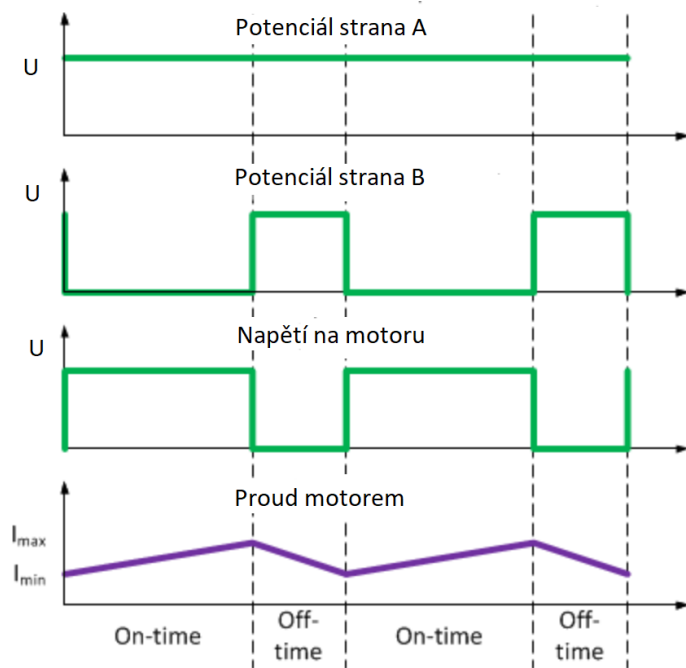
### 2.2.5.1.2 Sign-magnitude režim

V dalších dvou režimech je buď vstup IN1A stále zapnutý v log. úrovni H a na IN2A je přiveden signál PWM nebo naopak IN1A je stále zapnutý v log. úrovni H a na IN2A je přiveden signál PWM podle požadované polaroty napětí na výstupu H-můstku. To odpovídá řádkům 3 a 4 v Tab. 10. H-můstek je v zapnutém stavu udržován log. úrovní H na vstupu EN<sub>A</sub>, resp. EN<sub>B</sub>. Ke spínání tranzistorů dochází tak pouze v jedné polovině H-můstku, druhá polovina je stále zapnutá. V anglické literatuře se tento režim nazývá Sign-magnitude<sup>4</sup>.



Obr. 31: Spínací režim sign-magnitude H-můstku [24] pro směr vpřed a vzad

<sup>4</sup> Volně přeloženo z angličtiny jako znaménko-velikost, nebo také směr-velikost

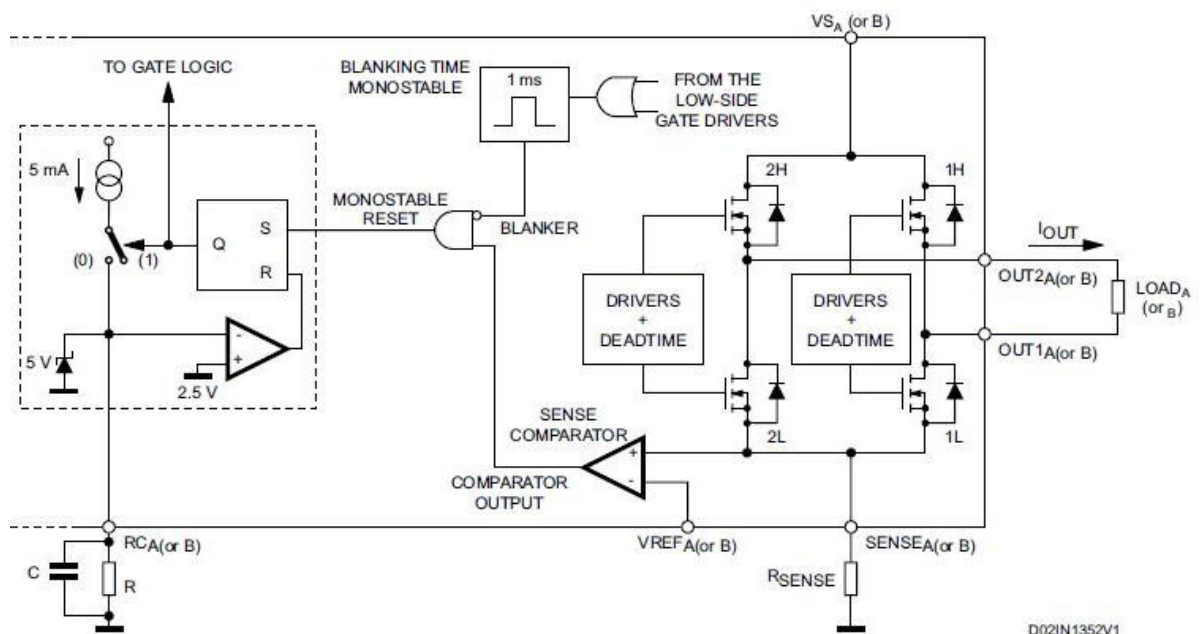


Obr. 32: Proud a napětí na motoru při řízení PWM typu sign-magnitude



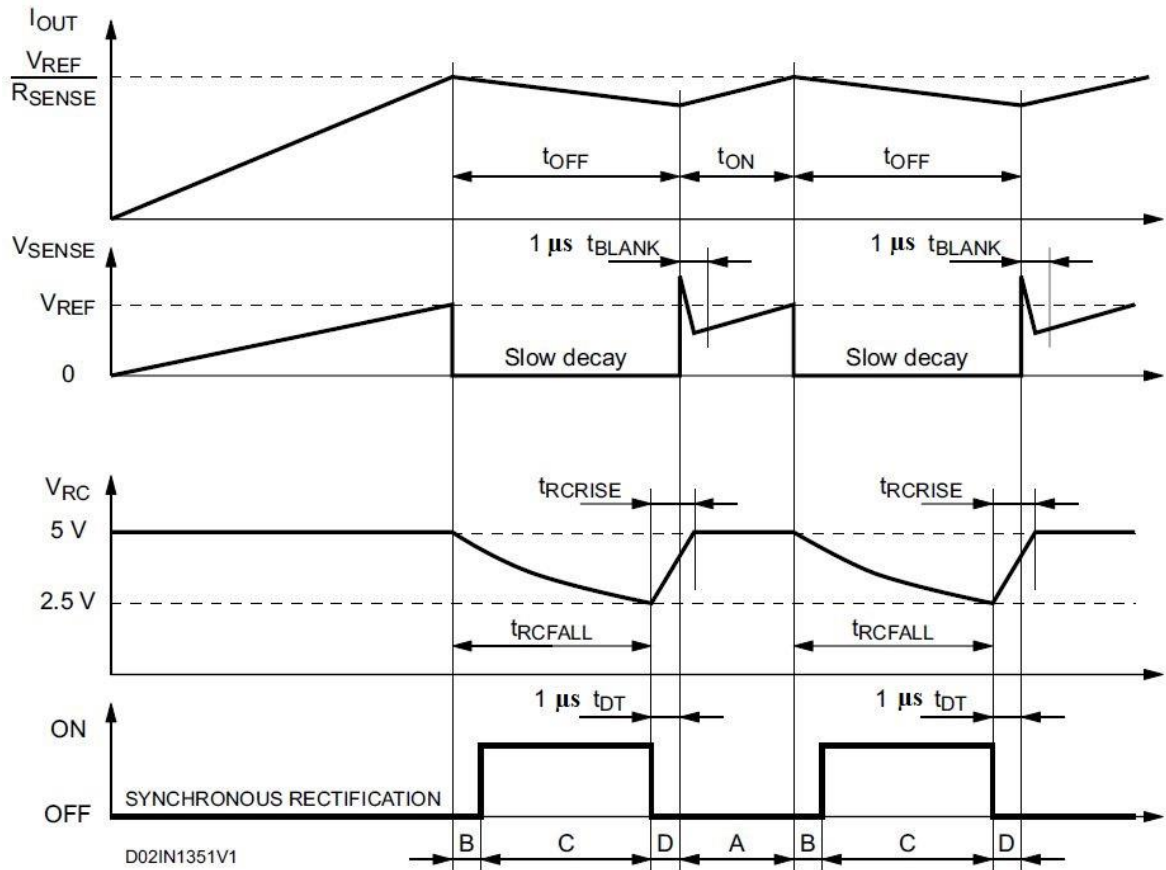
### 2.2.5.2 Omezení výstupního proudu H-můstku

Tato funkce bude pravděpodobně velmi často využita. Připojované akční členy mají velký odběr proudu a bez omezení výstupního proudu H-můstku by nebylo možné vůbec v některých režimech IO L6207 uchládit. Detaily viz kapitola 2.5.1 Měření teploty. Dvojitý H-můstek L6207 obsahuje vnitřní regulátor proudu s konstantní dobou vypnutí PWM. Délka této doby vypnutí se nastavuje pomocí externího RC článku. Na Obr. 33 je zjednodušené schéma bloku pro omezení výstupního proudu. Pin  $R_{SENSE}$  slouží k připojení externího bočnicku, na kterém lze měřit úbytek napětí, a vypočítat tak proud, který zátěží teče. Výkonové rezistory na tento účel použité by měly mít co nejmenší indukčnost a být co nejbližší pouzdru H-můstku pro omezení přechodových jevů. V testeru byla použita paralelní kombinace rezistorů  $2 \times 0,39 \Omega$ , tudíž je výsledná hodnota  $0,195 \Omega$ . Povolený ztrátový výkon rezistorů v pouzdře R2512 je 1 W.



Obr. 33: Schéma bloku proudové regulace L6207 [20]

K pinu  $RC_A$ , respektive  $RC_B$  je nutno připojit RC článek, kterým se nastavuje  $t_{OFF}$  a  $t_{ON}$  PWM regulace proudu. Je vhodné nastavit časovou konstantu tak, aby byl proud co nejvíce vyhlazený. Hodnota kondenzátoru  $C_{OFF}$  byla použita 820 pF a odpor  $R_{OFF}$  má hodnotu 39 k $\Omega$ . Parametr  $t_{OFF}$  je tedy podle vzorce (21) 20,188  $\mu$ s. Hodnota  $t_{DT}$  je typicky 1  $\mu$ s.



Obr. 34: Průběh omezení výstupního proudu H-můstku [20]

Na Obr. 34 jsou průběhy proudového omezení s vyznačenými časy  $t_{OFF}$  a  $t_{ON}$ . Jakmile výstupní proud H-můstku dosáhne úrovně dané poměrem  $V_{REF}$  a  $R_{SENSE}$ , dojde k zásahu komparátoru, který porovnává napětí na  $R_{SENSE}$  a vstupu  $V_{REF}$ . Jakmile je napětí na  $R_{SENSE}$  vyšší než  $V_{REF_A}$ , nebo  $V_{REF_B}$ , monostabilní klopný obvod vypne spodní tranzistor v H-můstku. Ten zůstane vypnutý po dobu  $t_{OFF}$ . Proud zátěži mezitím protéká v horní polovině H-můstku, viz Obr. 28 C. Jakmile uplyne doba  $t_{OFF}$ , monostabilní obvod se opět překlopí a opět zapne spodní tranzistor v H-můstku. Celková doba  $t_{OFF}$  je součtem  $t_{RCFALL}$  a interně generovaného ochranného intervalu  $t_{DT}$ .

Doba náběhu  $t_{RCRISE}$  je daná kapacitou kondenzátoru  $C_{OFF}$  podle vzorce:

$$t_{RCRISE} = 600 \cdot C_{OFF} = 600 \cdot 820 \cdot 10^{-12} = 0,492 \mu\text{s} \quad (19)$$

Doba vypnutí  $t_{OFF}$  se spočítá podle vzorce (21) jako:

$$t_{RCFALL} = 0,6 \cdot R_{OFF} \cdot C_{OFF} \quad (20)$$

$$t_{OFF} = t_{RCFALL} + t_{DT} = 0,6 \cdot R_{OFF} \cdot C_{OFF} + t_{DT} \quad (21)$$

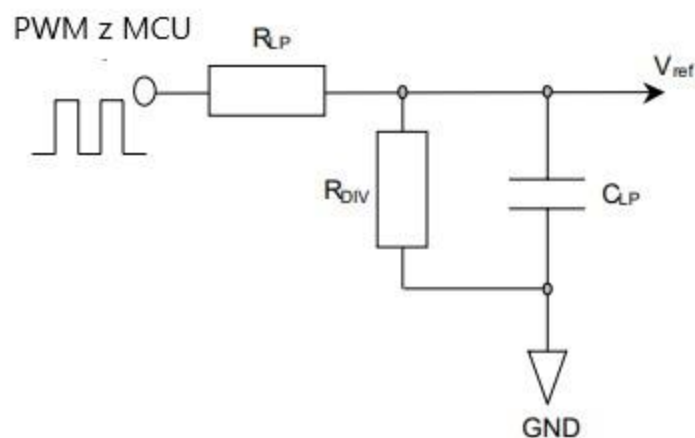
$$t_{OFF} = t_{RCFALL} + t_{DT} = 0,6 \cdot 39 \cdot 10^3 \cdot 820 \cdot 10^{-12} + 1 \cdot 10^{-6}$$

$$t_{OFF} = 20,188 \mu S$$

Doba  $t_{BLANK}$  zabraňuje opětovnému přepnutí monostabilního klopného obvodu a tím nechtěnému vybavení proudového omezení v důsledku proudové špičky tekoucí rezistorem  $R_{SENSE}$ , která vzniká při znovu zapnutí spodního tranzistoru v H-můstku.

Proudové omezení funguje nezávisle na stavu vstupů IN1 a IN2. Tím je umožněna i střída PWM 100 %, protože obvod nečeká na začátek další periody PWM, aby sepnul tranzistory v diagonále a proud zátěží začal opět růst. Proudové omezení bude udržovat nastavenou hodnotu proudu, dokud komparátor nevyhodnotí pokles úbytku napětí na bočníku  $R_{SENSE}$  a jeho výstup nepřeklopí monostabilní klopný obvod.

Jak již bylo zmíněno, lze za použití vstupů  $V_{REFA}$ ,  $V_{REFB}$  nastavit maximální špičkový proud zátěží. Z praktických důvodů regulace obou H-můstků na stejnou hodnotu omezení proudu zátěží se tyto vstupy spojují paralelně. Analogové napětí pro nastavení proudového omezení se vytváří jednoduchým D/A převodníkem tvořeným PWM z časovače/čítače TC2 na výstupu MCU (viz kapitola 2.4.3.2) a RC filtrem typu dolní propust na vstupu H-můstku, viz Obr. 35. Toto řešení bylo nutné, protože MCU nemá interní D/A převodník.



Obr. 35: D/A převodník pro nastavení proudového omezení v H-můstku [20]

Velikost referenčního napětí  $V_{ref}$  se vypočte podle vzorce (22) níže, kde  $D_{\mu C}$  je velikost střídy PWM v rozsahu 0 až 1 z mikrokontroleru a  $R_{DIV}$ ,  $R_{LP}$  a  $C_{DIV}$  jsou hodnoty odporu a kapacity dolnofrekvenčního filtru.

$$V_{ref} = \frac{5V * D_{\mu C} * R_{DIV}}{R_{DIV} + R_{LP}} \quad (22)$$

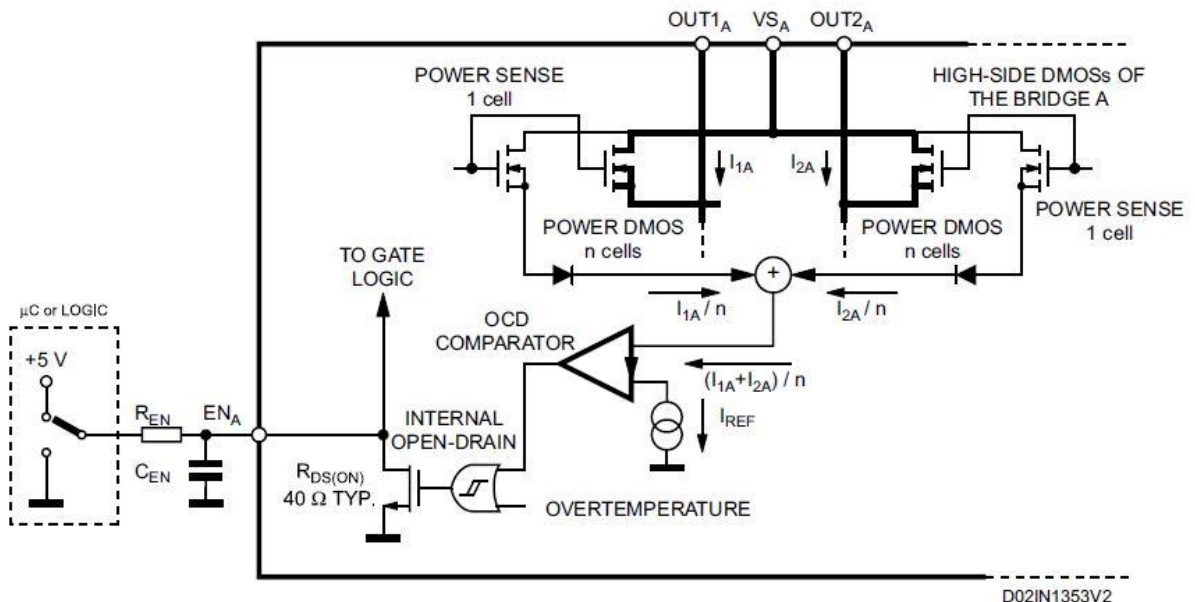
$$V_{refmin} = \frac{5 * 0,01 * 1000}{1000 + 4700} = 0,0088 V \quad (23)$$

$$V_{refmax} = \frac{5 * 1 * 1000}{1000 + 4700} = 0,877 V \quad (24)$$

V zapojení jsou použity hodnoty  $R_{DIV} = 1 \text{ k}\Omega$ ,  $R_{LP} = 4,7 \text{ k}\Omega$  a  $C_{DIV} = 220 \text{ nF}$ . Je nutné volit parametry komponent tak, aby bylo zvlnění co nejmenší s ohledem na použitou frekvenci PWM z mikrokontroleru. Frekvence PWM je 31,372 kHz, viz kapitola 2.4.3.2 Časovač/čítač TC2, to odpovídá periodě  $T_{PWM} = 31,8 \mu\text{s}$ . Časová konstanta  $\tau_{DP}$  dolnofrekvenčního filtru je 180  $\mu\text{s}$ . V případě regulace střídy 1-100 % se pohybuje velikost  $V_{ref}$  dle (36) od 0,0088 V až 0,877 V. Při použití  $R_{SENSE} = 0,195 \Omega$  to odpovídá rozsahu proudu 45 mA až 4,5 A. Hodnotu proudu [A] je možno programově měnit v menu. Lze i nastavit fixně rozsah proudu tak, aby nebylo možné zvolit velikost proudu, kdy je již problematické chlazení H-můstku. Dá se tedy říci, že nezáleží na tom, jaký ohmický odpor mají vinutí motorů, uživatel má možnost nastavit požadovaný proud odebíraný akčním členem.

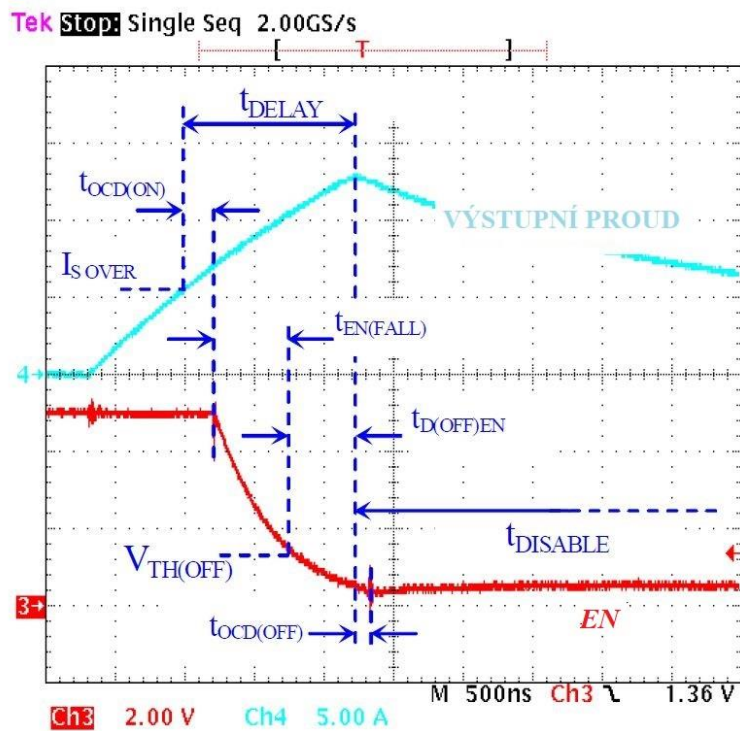
### 2.2.5.3 Nadproudová ochrana H-můstku

Tato ochrana vypíná H-můstek při zkratu výstupů mezi sebou nebo při zkratu výstupů proti zemi. Zabraňuje tedy poškození H-můstku v případě závady na akčním členu, případně chrání proti chybám obsluhy. Interní OCD komparátor porovnává proud protékající měřícím elementem v horní polovině H-můstku s interním referenčním zdrojem proudu. Jakmile hodnota proudu tekoucí H-můstkem dosáhne hranice typicky 5,6 A, tranzistor MOSFET s otevřeným drainem se otevře a uzemní vstup  $EN_A$  resp.  $EN_B$  a zároveň vypne všechny čtyři spínací tranzistory v dotčeném H-můstku. Proud doznívá nulovými diodami. Žádný proud neteče spínacími tranzistory, tím pádem nadproudová ochrana nemá jak měřit protékající proud. Po uplynutí zpoždění  $t_{OCD(OFF)}$  začne narůstat napětí na pinu EN a po dosažení prahového zapínacího napětí se opět sepnou spínací tranzistory a proud jimi začne znovu protékat. V případě přetrvávajícího zkratu výstupu proti zemi nebo mezi výstupními piny  $OUT1_A$ ,  $OUT2_A$ , resp.  $OUT1_B$ ,  $OUT2_B$  RC článek na vstupu pinu  $EN_A$  resp.  $EN_B$  redukuje efektivní hodnotu výstupního proudu nuceným vypnutím H-můstku po dobu fixního intervalu  $t_{DISABLE}$  při každém vybavení nadproudové ochrany. Maximální hodnota zkratového proudu závisí na rychlosti přeběhu, na povaze elektrického obvodu ve zkratu, na napětí zdroje a na zpoždění  $t_{DELAY}$ . Detaily průběhů proudu jsou na Obr. 37 a Obr. 39.

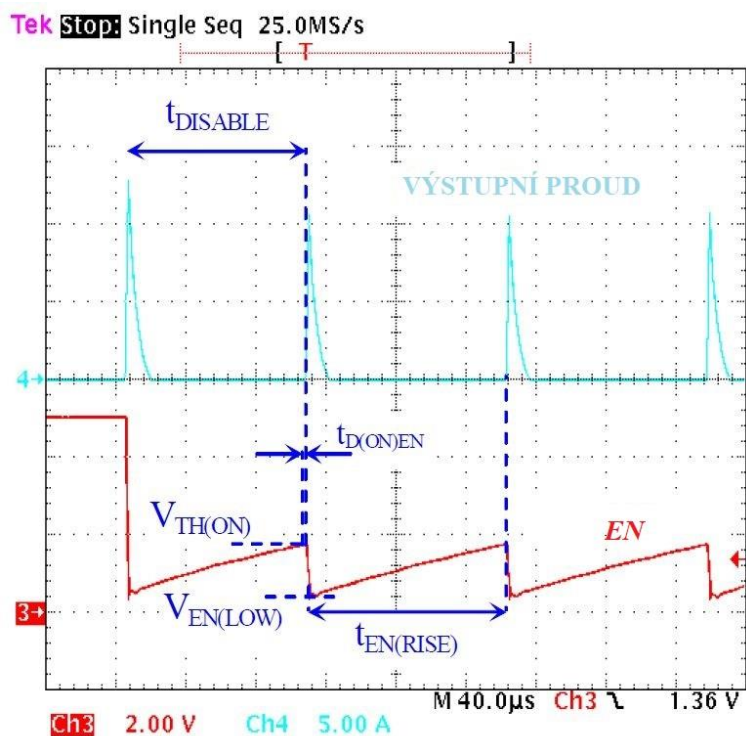


Obr. 36: Zjednodušené schéma nadproudové ochrany L6207 [20]

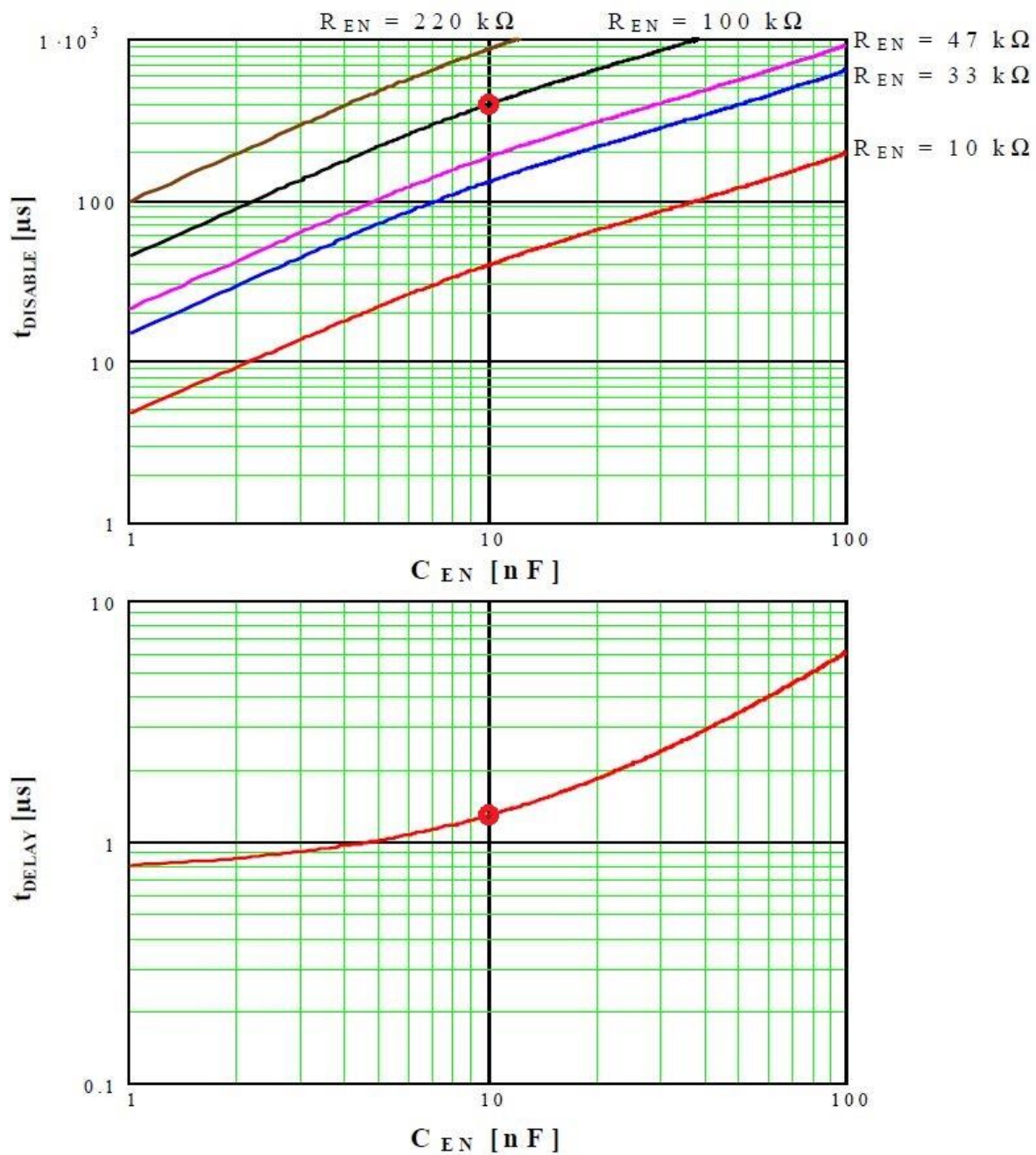
RC článkem na vstupech  $EN_A$ ,  $EN_B$  nastavujeme parametry  $t_{DELAY}$  a  $t_{DISABLE}$ . Hodnotu  $C_{EN}$  volíme tak, aby byla odezva na nadproud co nejkratší a hodnotu  $R_{EN}$  tak, aby byl čas vypnutí dostatečně dlouhý, viz aplikační list AN1762 [21]. Na DPS je osazen  $C_{EN} = 10 \text{ nF}$ ,  $R_{EN} = 100 \text{ k}\Omega$  a s těmito hodnotami dosáhneme  $t_{DISABLE} 400 \mu\text{s}$  a  $t_{DELAY}$  přibližně  $1,4 \mu\text{s}$  [21], viz Obr. 39. Tyto hodnoty splňují doporučení výrobce v aplikační poznámce [21]. Minimální hodnota  $t_{DISABLE}$  je doporučena  $100 \mu\text{s}$ , hodnota  $t_{DELAY}$  by se měla pohybovat mezi  $1$  až  $2 \mu\text{s}$ . Nastavení těchto hodnot zajistí dostatečně rychlý zásah nadproudové ochrany a dostatečný čas vypnutí H-můstku, aby nedošlo k nárůstu efektivní hodnoty proudu nad nepřijatelnou mez. V testeru ovšem není spoléháno pouze na logiku interní nadproudové ochrany. Na uzemnění pinu  $EN_A$ , resp.  $EN_B$  reaguje MCU vypnutím (log. úroveň L) výstupu  $EN_A$ , resp.  $EN_B$ . Ze schématu na Obr. 36 je patrné, že vstup  $EN_A$ , resp.  $EN_B$  bude uzemněn i v případě vybavení teplotní ochrany. H-můstek se vypne při přesáhnutí teploty na přechodu  $165 \text{ }^\circ\text{C}$  s  $15 \text{ }^\circ\text{C}$  hysterezí.



Obr. 37: Detail průběhu výstupního proudu při zásahu nadproudové ochrany, vliv  $t_{DELAY}$  a  $t_{DISABLE}$  [21]



Obr. 38: Detail průběhu výstupního proudu při zásahu nadproudové ochrany, vliv  $t_{DISABLE}$  [21]



Obr. 39: Vliv velikosti  $C_{\text{EN}}$   $R_{\text{EN}}$  na délku  $t_{\text{DISABLE}}$  a  $t_{\text{DELAY}}$  [21]



#### 2.2.5.4 Výkonové ztráty H-můstku

Vzhledem k použití testeru v poměrně nepříznivých podmínkách je nutno uvažovat i aspekt chlazení a výkonových ztrát dvojitého H-můstku, který je pouzdru PowerSO36. Celé zařízení je umístěno na DPS, kde není příliš prostoru na rozlité plochy mědi, ani na rozměrnější chladič. Investor chce primárně chladit tester pasivně. Tepelný odpor  $R_{THJA}$  přechod-prostředí je velmi závislý na návrhu DPS. Pouzdro IO L6207 je připájeno na oboustranné desce plošných spojů s prokovy pod pouzdrům. Vyzařované teplo je odváděno do okolního prostředí také pomocí malého chladiče s tepelným odporem  $19,5 \text{ }^\circ\text{C/W}$ , který je umístěn shora pouzdra, avšak na pouzdro samotné nedoléhá a odvádí teplo jen z DPS. Následující výpočet maximální efektivní hodnoty proudu jedním H-můstkem (druhý není použit) pro teplotu přechodu  $125 \text{ }^\circ\text{C}$  však platí pouze bez tohoto přidaného chladiče. Maximální bezpečná teplota přechodu je dle [20]  $125 \text{ }^\circ\text{C}$ . Pro montáž na dvouvrstvou DPS s prokovy do spodní vrstvy a při chladící ploše  $6 \text{ cm}^2$  je tepelný odpor přechod-prostředí  $R_{THJA} = 15 \text{ }^\circ\text{C/W}$ .

Tomu odpovídá povolený ztrátový výkon  $P_{TOT}$  celého L6207 (oba H-můstky) při teplotě okolí  $30 \text{ }^\circ\text{C}$ :

$$P_{TOT} = \frac{t_J - t_{AMB}}{R_{THJA}} = \frac{125 - 30}{15} = 6,33 \text{ W} \quad (25)$$

Kde  $t_J$  [ $^\circ\text{C}$ ] je teplota přechodu,  $t_{AMB}$  [ $^\circ\text{C}$ ] teplota prostředí a  $R_{THJA}$  [ $^\circ\text{C/W}$ ] tepelný odpor přechod-prostředí.

Vlastní příkon L6207 při napájecím napětí  $V_{CC} = 14 \text{ V}$  a vlastním spotřebě proudu  $I_H = 0,005 \text{ A}$  je:

$$P_V = V_{CC} \cdot I_H = 14 \cdot 0,005 = 0,07 \text{ W} \quad (26)$$

Spínací ztráty  $P_{SW}$  při  $t_{RISE} = t_{FALL} = 250 \text{ ns}$ , spínací frekvenci  $f_{PWM} = 10 \text{ kHz}$  a maximálním efektivním proudem  $I_{RMSMAX}$  jsou:

$$P_{SW} = I_{RMSMAX} \cdot V_{CC} \cdot (t_{RISE} + t_{FALL}) \cdot f_{PWM} \quad (27)$$

$$P_{SW} = 2,334 \cdot 14 \cdot 250 \cdot 10^{-9} \cdot 10 \cdot 10^3 = 0,163 \text{ W} \quad (28)$$

Kde  $V_{CC}$  [V] je napájecí napětí,  $t_{RISE}$  doba je sepnutí [s] a  $t_{FALL}$  je doba vypnutí [s]. Pro výpočet  $I_{RMSMAX}$ , který je vlastně v tuto dobu neznámý, byla použita funkce řešitel v Excelu. Ta dopočítala spínací ztráty  $P_{SW}$  na základě zadané teploty přechodu  $125 \text{ }^\circ\text{C}$ . V průběhu měření teploty, viz kapitola 2.5.1, bylo zjištěno, že spínací ztráty jsou oproti vodivostním ztrátám opravdu zanedbatelné. Frekvence  $10 \text{ kHz}$  byla zvolena proto, že je maximální možná a spínací ztráty jsou přímo úměrné spínacímu kmitočtu.

Vodivostní ztráty  $P_{RDS}$  v sepnutém stavu jsou:

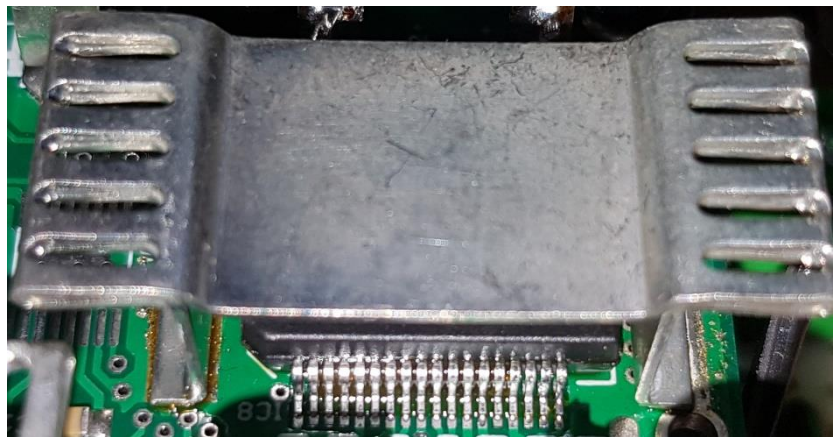
$$P_{RDS} = P_{TOT} - P_V - P_{SW} = 6,33 - 0,07 - 0,163 = 6,1 \text{ W} \quad (29)$$

Efektivní hodnota proudu  $I_{RMSMAX}$  je při teplotě přechodu  $t_j=125 \text{ °C}$ :

$$I_{RMSMAX} = \sqrt{\frac{P_{RDS}}{R_{DSONL} + R_{DSONH}}} = \sqrt{\frac{6,1}{0,59 + 0,53}} = 2,334 \text{ A} \quad (30)$$

Kde  $R_{DSONL}$  a  $R_{DSONH}$  je odpor spodního a vrchního spínacího tranzistoru v sepnutém stavu při dané teplotě přechodu  $t_j=125 \text{ °C}$ .

Maximální efektivní hodnota proudu 2,334 A je poměrně malá vzhledem k velikosti odebíraného proudu některých akčních členů. Například škrťací klapka z kapitoly 1.1.1 z vozu Škoda Roomster kód motoru BXW má při napájecím napětí 12 V odběr asi 6,5 A při plném otevření na dorazu. EGR ventil z vozu Mazda popsany v kapitole 1.1.2 má odběr 6 A v plně otevřeném stavu. Motory používané v automobilech mohou mít ale mnohem větší odběr, například ventilátor topení může mít odběr při jmenovitém napětí palubní sítě 14 V až 16 A. Je tedy nevyhnutelné většinu akčních členů testovat s aktivním proudovým omezením.



Obr. 40: Detail chladiče dvojitého H-můstku z boku

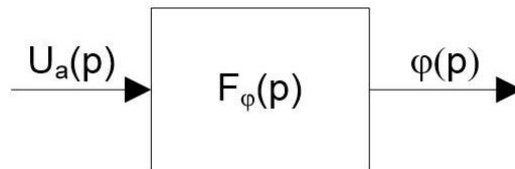


Obr. 41: Pohled shora na chladič dvojitého H-můstku a na chladičí plochu na DPS

### 2.3 Lineární model škrťící klapky

Aby bylo možné najít vhodné konstanty číslicového PID regulátoru, který používá jeden z režimů testování škrťící klapky pomocí zpětnovazební regulace, je nutné nejprve sestavit model regulačního obvodu. Následně lze za použití nástroje PID tuner v Matlabu naladit PID regulátor na požadované vlastnosti. Škrťící klapka je elektromechanická soustava, která je poháněna stejnosměrným motorem s permanentním magnetem. Moment motoru působí přes převody na klapku a zajišťuje její otevírání a zavírání. Průběh zatěžovacího momentu silně ovlivňuje pružina, která zajišťuje návrat klapky do výchozí tzv. limp home pozice. Zatěžovací moment pružiny se mění s velikostí výchylky klapky. Dále je nutno do celkového zatěžovacího momentu připočítat třecí ztráty například v převodech a ložiskách. Zatěžovací moment způsobený tímto třením se s rostoucími otáčkami lineárně zvětšuje. Nelinearity způsobené pružinou a třením značně komplikují regulaci polohy škrťící klapky [15].

Pro odvození matematického modelu potřebujeme rovnici popisující obvod kotvy motoru (3), a pohybovou rovnici (4) popisující rovnost momentu stroje a zatěžovacího momentu. Po úpravě rovnic podle postupu uvedeném v [14] uvažujeme jako vstupní veličinu napětí  $U_a$  a výstupní veličinu úhel natočení hřídele  $\varphi$ . Na celou soustavu nahlížíme jako na systém, viz Obr. 42, který má jeden vstup a jeden výstup a jehož vnitřní vlastnosti jsou popsány přenosovou funkcí pomocí Laplaceovy transformace. Následující postup je převzat z [14].



Obr. 42: Regulovaná soustava jako systém s jedním vstupem a výstupem [14]

Otáčkový přenos soustavy je:

$$F_{\omega}(p) = \frac{\Omega(p)}{U_a(p)} = \frac{k_m}{(p \cdot L_a + R_a) \cdot (p \cdot J_c + b) + k_m \cdot k_u} \quad (31)$$

Pokud je elektrická konstanta  $\tau_e$  alespoň o řád menší oproti mechanické konstantě, můžeme zanedbat indukčnost kotvy  $L_a$  a systém se zjednoduší o řád a má pouze jednu časovou konstantu

$$F_{\omega}(p) = \frac{\Omega(p)}{U_a(p)} = \frac{k_m}{p \cdot R_a \cdot J_c + b \cdot R_a + k_m \cdot k_u} = \frac{k_0}{p \cdot \tau_m + 1} \quad (32)$$

Kde:

$$\tau_m = \frac{R_a \cdot J_c}{b \cdot R_a + k_m \cdot k_u} \quad \text{je elektromechanická časová konstanta [s]} \quad (33)$$

a kde:

$$k_0 = \frac{k_m}{b \cdot R_a + k_m \cdot k_u} \quad \text{je zesílení systému [rad/s/V]} \quad (34)$$

Vztah mezi okamžitou úhlovou rychlostí hřídele  $\omega$  a úhlovým natočením hřídele je [14]:

$$\omega(t) = \frac{d\varphi(t)}{dt} \quad (35)$$

Vztah (35) po aplikaci Laplaceovy transformace (36) dosadíme do rovnice přenosu systému (31), kde jsou výstupní veličinou otáčky hřídele.

$$\varphi(p) = \frac{\Omega(p)}{p} \quad (36)$$

Dosadíme za  $\Omega(p) = p \cdot \varphi(p)$  do (31) a dostaneme rovnici přenosu soustavy na Obr. 42:

$$F_\varphi(p) = \frac{\varphi_{motor}(p)}{U_a(p)} = \frac{k_m}{p \cdot [p^2 \cdot L_a \cdot J_c + p \cdot (L_a \cdot b + R_a \cdot J_c)] + b \cdot R_a + k_m \cdot k_u} \quad (37)$$

Kde:

$\varphi_{motor}(p)$  – Laplaceův obraz polohy rotoru [rad]

$U_a(p)$  – Laplaceův obraz kotvy motoru [V]

$R_a$  – Odpor kotvy [ $\Omega$ ]

$L_a$  – Indukčnost kotvy [H]

$J_c$  – Celkový moment setrvačnosti rotujících částí [ $\text{kgm}^2$ ]

$b$  – Součinitel viskózního tření [Nm/rad/s]

$k_u$  – Napěťová konstanta [V/rad/s]

$k_m$  – momentová konstanta [Nm/A]

Podle rovnice (37) vidíme, že se jedná o integrační soustavu třetího řádu. Opět zanedbáme elektrickou konstantu  $\tau_e$  a řád soustavy se sníží ze třetího na druhý řád. V kapitole 1.1.1 jsou uvedeny naměřené hodnoty testované škrtkící klapky, můžeme tedy určit elektrickou konstantu  $\tau_e$  jako:

$$\tau_e = \frac{L_a}{R_a} = \frac{0,00138}{2,95} = 0,47 \text{ ms} \quad (38)$$

Tato hodnota bude jistě o řád nižší než elektromechanická časová konstanta  $\tau_m$ .

Rovnici (37) tedy můžeme zjednodušit do tvaru:

$$F_{\varphi_{motor}}(p) = \frac{\varphi_{motor}(p)}{U_a(p)} = \frac{k_m}{p(p \cdot R_a \cdot J_c + b \cdot R_a + k_m \cdot k_u)} = \frac{k_1}{p(p \cdot \tau_m + 1)} \quad (39)$$

Tento přenos platí pro osu motoru, kde:

$$k_1 = \frac{k_m}{b \cdot R_a + k_m \cdot k_u} \quad \text{je zesílení systému [rad/s/V]} \quad (40)$$

$$\tau_m = \frac{R_a \cdot J_c}{b \cdot R_a + k_m \cdot k_u} \quad \text{je elektromechanická časová konstanta [s]} \quad (41)$$

Přenos je uveden pro osu motoru, klapka je připojena přes převod. To znamená pouze násobení konstantou. Výsledný vztah pro přenos klapky bude:

$$F_{\varphi_{klapky}}(p) = \frac{F_{\varphi_{motor}}(p)}{\text{převod}} = \frac{\frac{\varphi_{motor}(p)}{\text{převod}}}{U_a(p)} = \frac{\varphi_{klapky}(p)}{U_a(p)} = \frac{k_{oklapky}}{p(p \cdot \tau_m + 1)} \quad (42)$$

Model klapky dle (42)(39) neobsahuje žádné nelinearity, které má skutečná škrťící klapka. Tento lineární model platí jen mimo limp home oblast, ve které klapku drží pružina. Pro identifikování parametrů lineárního modelu použijeme přechodovou charakteristiku. Přechodová charakteristika neboli v našem případě odezva na skok střídě, vyjadřuje závislost polohy klapky na střídě (tedy napětí) na kotvě motoru. Vztah mezi střídou PWM a napětím kotvy motoru  $U_a$  uvažujeme jako lineární, při zanedbání úbytků napětí na spínacích tranzistorech H-můstku.

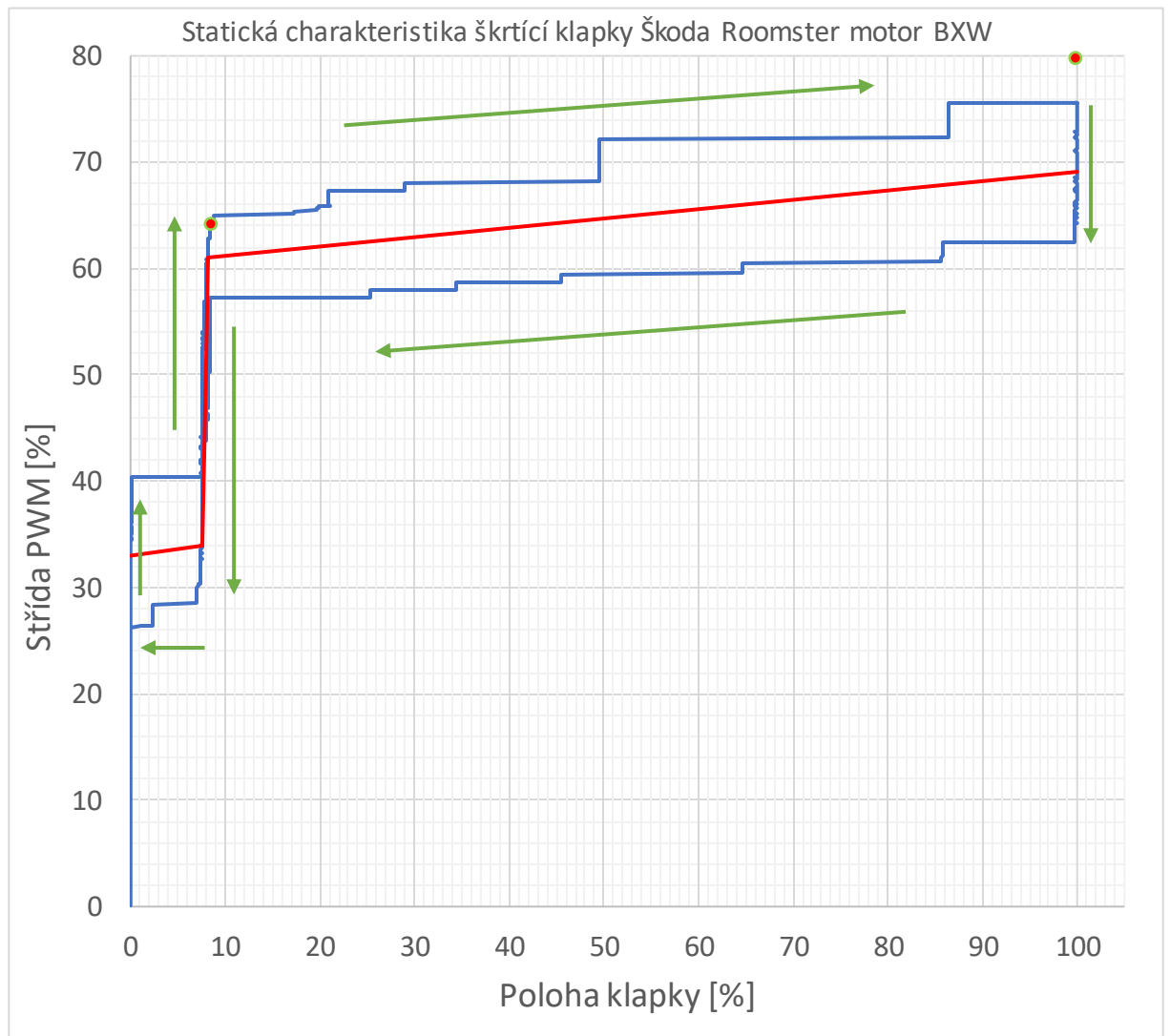
Abychom mohli správně zvolit rozmezí pro skok střídě, tedy počáteční hodnotu střídě, potřebujeme změřit statickou charakteristiku škrťící klapky. Průběh statické charakteristiky by měl ukázat, kde začíná lineární oblast, ve které budeme identifikovat parametry klapky při odezvě na skok střídě pomocí programu Matlab.

V modelu je zaveden offset střídě, při požadavku na střídě 0 není na svorkách nulové napětí ale plné záporné napětí. Střída 512 (50 %) je nulové napětí a střída 1023 (100 %) znamená plné kladné napětí na výstupu. To je dané režimem řízení H-můstku, který byl nastaven na locked anti-phase. Tento režim spínání je blíže vysvětlen v kapitole 2.4.8 Generování PWM

Samotné měření statické charakteristiky u vybrané škrtkové klapky (Škoda Roomster 1.4 l kód motoru BXW) bylo provedeno formou cyklu velmi pomalu rostoucí a klesající rampy střídání PWM. Frekvence PWM byla nastavena na 30 kHz pro zajištění linearity závislosti mezi napětím kotvy  $U_a$  a střídáním PWM, aby motor pracoval s nepřerušovaným proudem. Rozsah střídání je v podstatě  $\pm 100\%$ . Algoritmus v MCU byl nastaven tak, aby při spuštění každé 2 sekundy zvýšil střídání z 0 o jeden krok (0,195 %) až na hodnotu střídání 75,58 % (horní doraz klapky). Při dosažení horního dorazu klapky (poloha plně otevřeno) program střídání opět začal snižovat až do 0 (plné záporné napětí na kotvě motoru). Po dosažení spodního dorazu (klapka plně zavřena) program čekal 2 sekundy a opět začal střídání navyšovat, tj. opakovat celý cyklus znovu. Tento proces měření je převzat z [15].

Na Obr. 43 je graf sestavený z přijatých hodnot polohy klapky a střídání PWM. Při nulové střídání PWM a použitím locked anti-phase režimu řízení H-můstku je na klapce záporné napětí, které ji drží plně zavřenou. Jak střídání stoupá, klapka se dostane až do limp-home pozice, kdy je částečně otevřena asi na 8 %. V okolí limp-home pozice je patrný vliv pružiny, jejíž moment musí motor překonat, aby se klapka začala otevírat. Z grafu na Obr. 43 je patrné také malé zesílení soustavy, kdy je potřebná velká změna střídání pro malou změnu polohy. Z této polohy je nutný velký nárůst střídání až přibližně na 65 %, potom se již začne klapka více otevírat. V další části statické charakteristiky má soustava znatelně vyšší zesílení, malá změna střídání způsobí velkou změnu polohy. Střídání dále roste od 65 % až po přibližně 75 %, kdy klapka již narazí na horní doraz. Zřejmě vlivem tření a zřejmě i vlivem pružiny dochází během této fáze ještě k několika skokům jak při zvyšování, tak při snižování střídání PWM, resp. při otevírání a zavírání klapky. Statická charakteristika má poměrně velkou hysterizi a velmi proměnlivý sklon. Pružina má v celém rozsahu dostatečný moment pro překonání tření pro návrat do limp-home polohy.

Na Obr. 43 je zelenými šipkami znázorněn směr pohybu klapky. Červená křivka zobrazuje po částech lineární funkci polohy škrtící klapky v závislosti na střídě PWM neboli vliv pružiny. Ta byla sestrojena z průměrných hodnot mezi pohybem z plně zavřené do plně otevřené klapky a zpět. Červené body vyznačují počáteční a koncovou hodnotu při skoku střídy (podrobnosti jsou uvedeny v další kapitole).



Obr. 43: Statická charakteristika škrtící klapky

### 2.3.1 Identifikace parametrů lineárního modelu

Parametry lineárního modelu byly identifikovány z průběhu přechodové charakteristiky pomocí metody nejmenších čtverců za použití nástroje Systém Identification Tool v Matlabu. Vstupem je střída PWM v rozsahu [0-1023], což odpovídá 0-100 %. Výstupem je pak poloha klapky také v rozsahu [0-1023], to odpovídá poloze klapky 0-100 %.

Přenos modelu bude tedy určen jako:

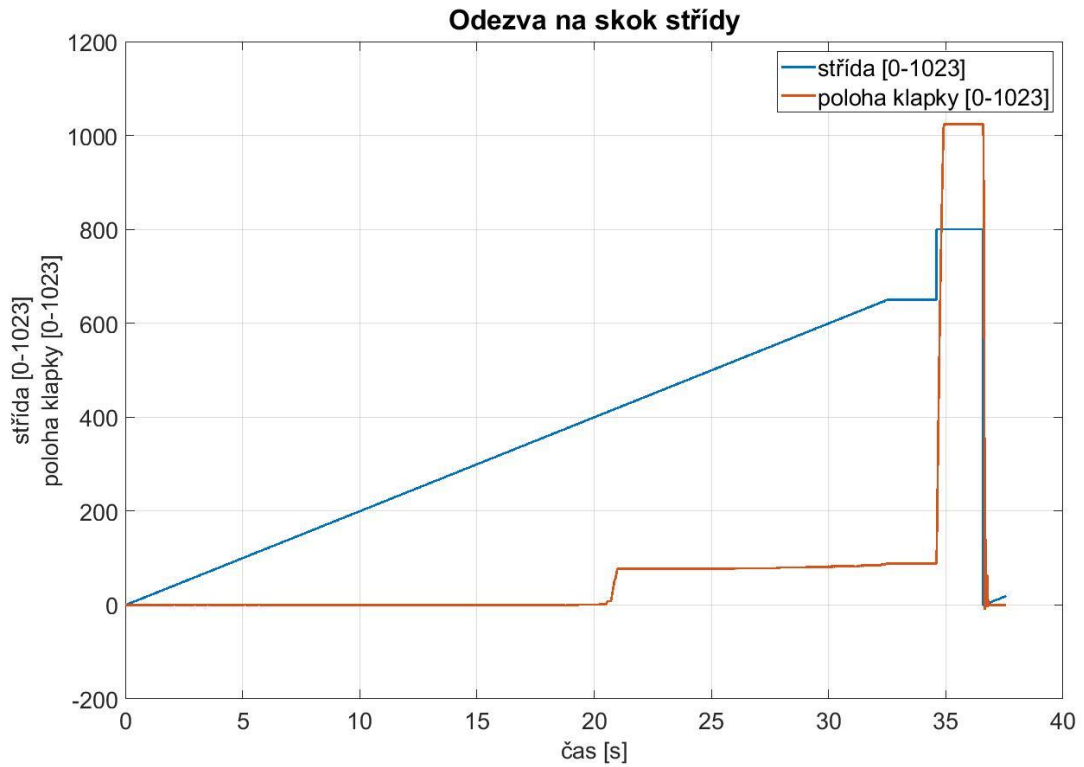
$$F_{\varphi_{\text{klapky}}}(p) = \frac{\varphi_{\text{klapky}}(p)}{\text{střída}(p)} = \frac{k_{\text{oklapky}}}{p(p \cdot \tau_m + 1)} \quad (43)$$

Pomocí Matlabu budeme hledat parametry  $\tau_m$ , což je elektromechanická konstanta [s] a  $k_{\text{oklapky}}$ , to je zesílení systému [% polohy/s/% střídý], viz (40) a (47). Bylo provedeno několik skoků střídý (viz Tab. 12) z hodnot těsně před začátkem lineární oblasti, v lineární oblasti i v oblasti, kde je ještě vliv pružiny markantní. Nejprve byla nastavena počáteční hodnota střídý pomocí pomalého nárůstu střídý, po ustálení (2 sekundy) byla skokově střída zvýšena na koncovou hodnotu. Perioda vzorkování hodnot byla zvolena 2 ms. Červeně je v Tab. 12 vyznačen řádek s použitými hodnotami skoku střídý a identifikované hodnoty elektromechanické konstanty a zesílení.

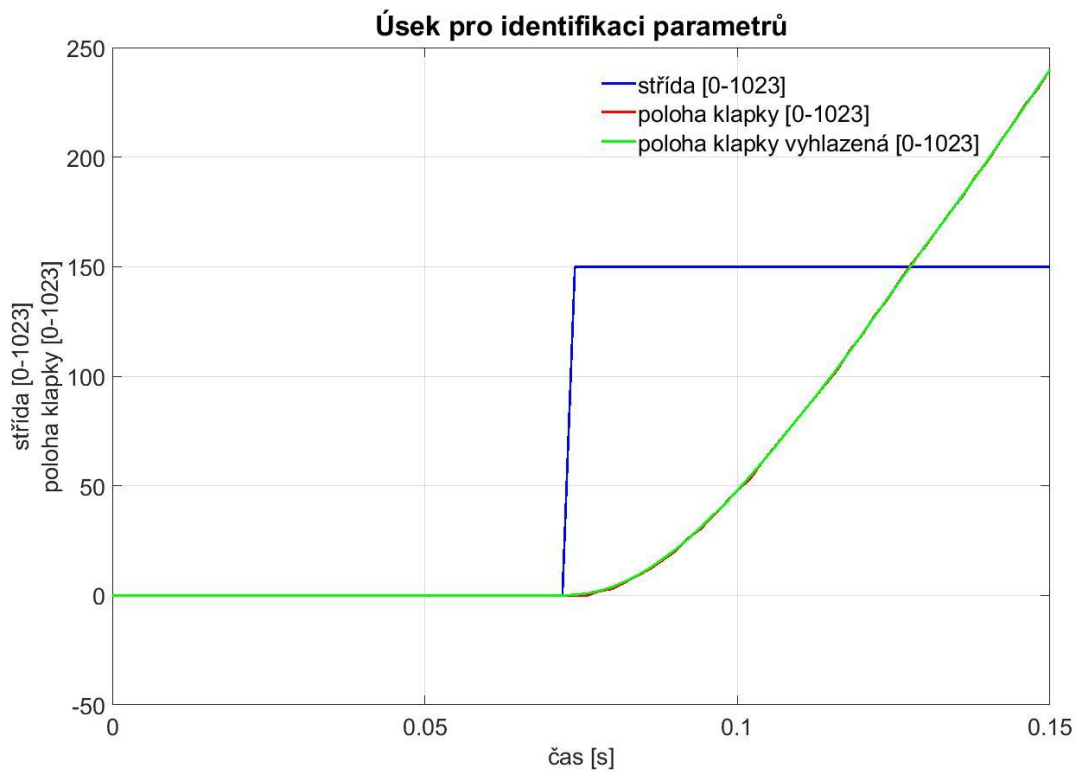
počáteční hodnota střídý [%]	koncová hodnota střídý [%]	$k_{\text{oklapky}}$ [rad/s/V]	$\tau_m$ [s]
70,31	78,13	30,163	0,015717
<b>63,48</b>	78,13	<b>28,083</b>	<b>0,018963</b>
63,48	68,36	23,511	0,018676
70,31	99,90	35,236	0,096661
65,43	68,36	29,42	0,036462
48,83	78,13	12,154	0,010347

Tab. 12: Provedené skoky střídý a identifikované hodnoty  $k_{\text{oklapky}}$  a  $\tau_m$  z Matlabu



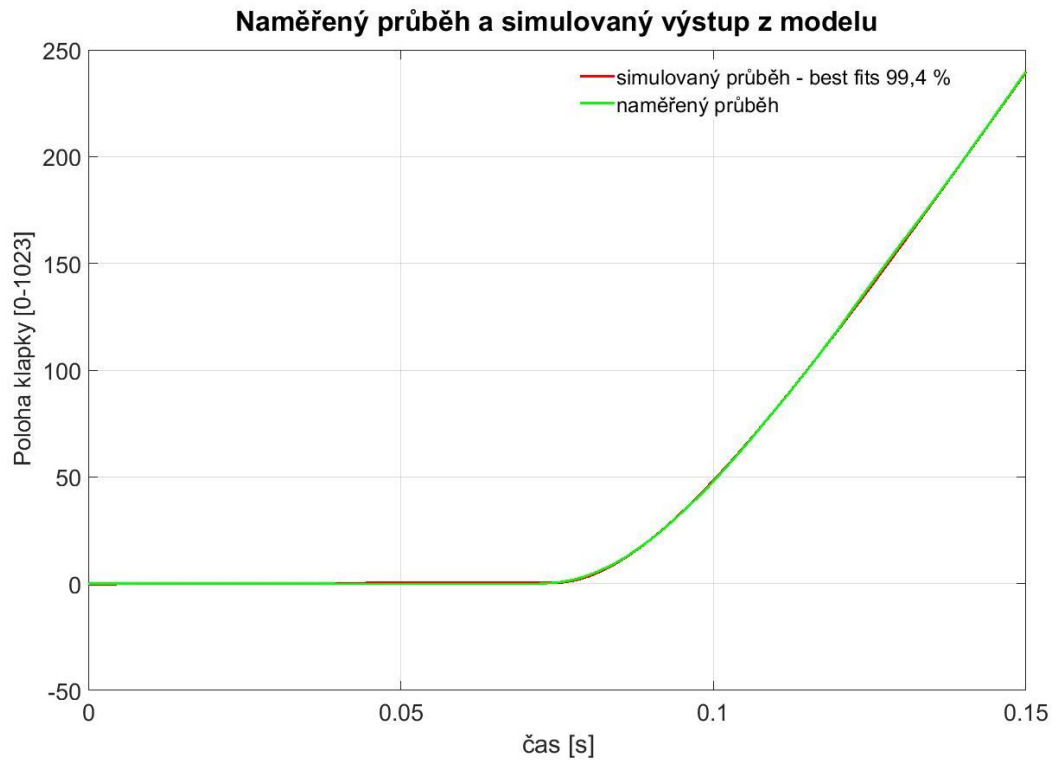


Obr. 44: Odezva na skok střidy 63,48 % až 78,13 %



Obr. 45: Vybraný úsek odezvy na skok střidy pro identifikaci parametrů

Vybraný úsek dat z odezvy na skok střídý na Obr. 45 dále použijeme pro identifikaci parametrů lineárního modelu. Při měření polohy klapky nebyl použit filtr pro vyhlazení signálu klapky, signál byl čistý a bez rušení. Pomocí System Identification Toolu byly vypočteny hodnoty  $\tau_m$  a  $k_{0klapky}$  uvedené v Tab. 12. Na Obr. 46 je porovnání průběhu modelu a skutečného naměřeného průběhu. Podle shody 99,4 % je vidět, že nalezené parametry lineárního modelu se shodují s naměřenými hodnotami.



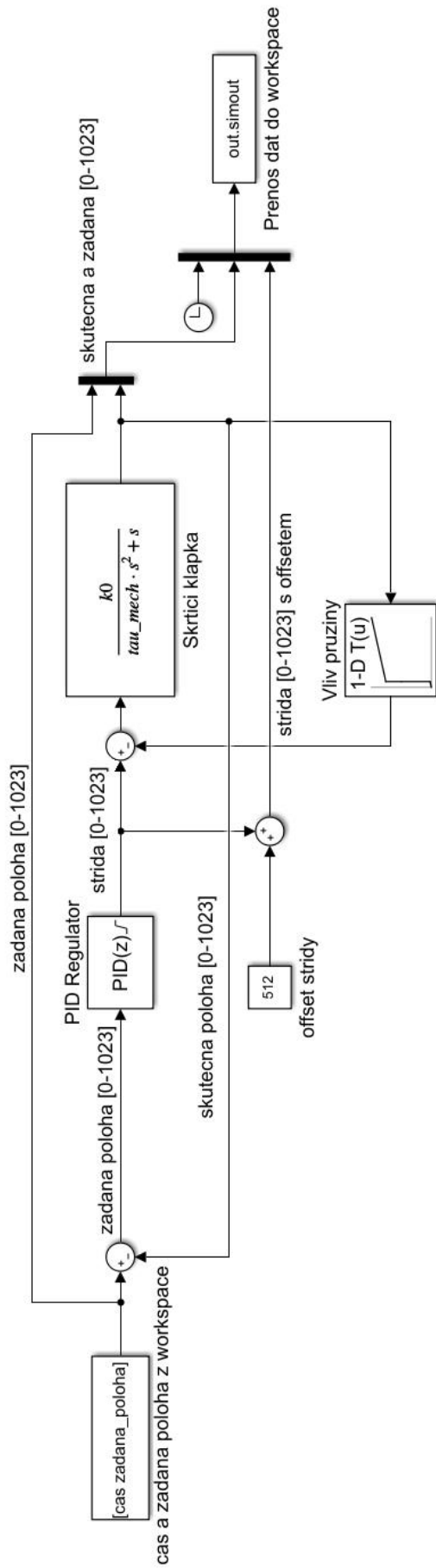
Obr. 46: Porovnání naměřené hodnoty a výstupu z modelu v Matlabu

Výsledný přenos soustavy je tedy:

$$F_{\varphi_{klapky}}(p) = \frac{k_{0klapky}}{p(p \cdot \tau_m + 1)} = \frac{28,083}{p(0,018963p + 1)} \quad (44)$$

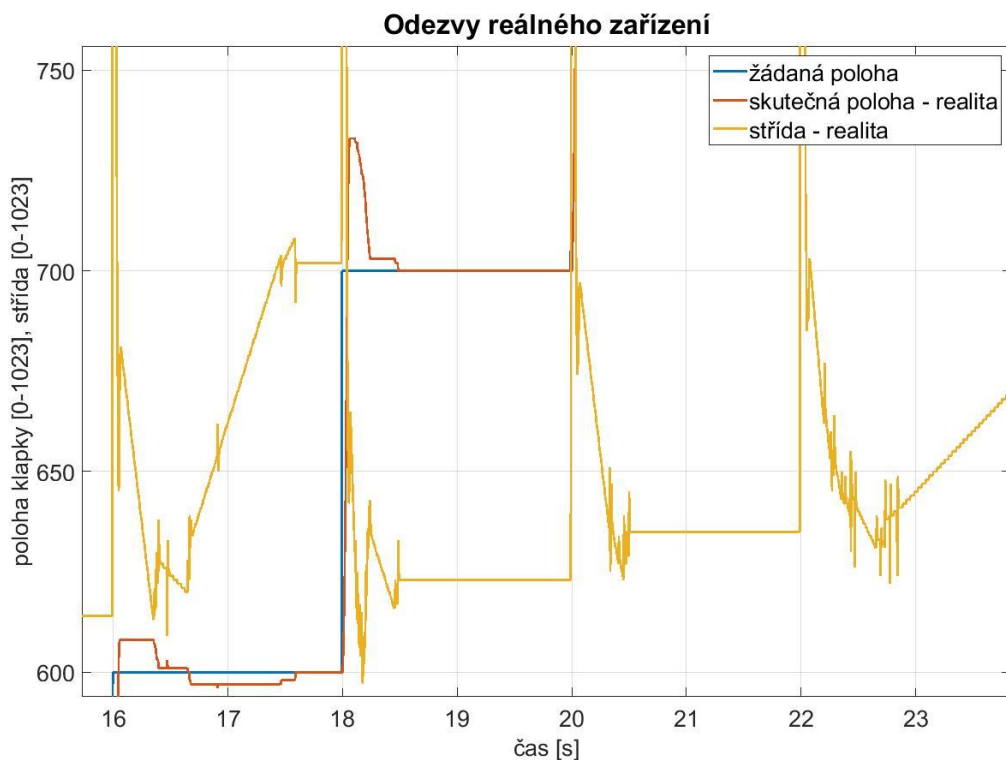
### 2.3.2 Model regulačního obvodu

Vedoucí práce sestavil v Simulinku model regulačního obvodu a pomocí PID tuneru našel vhodné konstanty diskretního PID regulátoru, viz Obr. 47. Tyto konstanty byly otestovány na reálném zařízení a ručně doladěny. Při regulaci byly pohledem zjištěny drobné překmity při dojezdu na žádanou polohu. Pro porovnání lineárního modelu a reálného zařízení bylo provedeno měření s několika skoky žádané polohy, viz Obr. 48. Na každé žádané hodnotě algoritmus čekal dvě sekundy na ustálení. Počáteční žádaná hodnota byla limp-home, vrcholová hodnota byla 1000 (97,66 %). Při nárůstu program vykonal 12 skoků a při snižování také. Data byla opět odesílána přes sériové rozhraní do počítače, odesílané parametry byly čas, žádaná hodnota, skutečná hodnota a velikost střídý. Perioda regulátoru byla nastavena na 5 ms, žádaná a skutečná hodnota a velikost střídý v bezrozměrných jednotkách [0-1023]. Do modelu nebyly zaneseny kompenzátory statického tření. Statické tření způsobuje hysterezi statické charakteristiky. V důsledku tohoto tření se klapka zavírá při jiné střídě, než při které se otevírá. Celá soustava je velice komplikovaná a přesné řízení škrťící klapky vyžaduje kompenzaci všech popsaných nelinearit. Regulátor v automobilu si musí poradit s proměnlivým zesílením, které je jiné při různých úhlech otevření klapky ale také s mechanickými odchylkami, které vznikají sériovou výrobou a opotřebením. Téma řízení škrťící klapky by jistě vydalo na celou diplomovou práci. Vzhledem k tomu, že v testeru je zpětnovazební řízení klapky pouze jedna z mnoha funkcí a neslouží k řízení chodu motoru, byl model sestaven pouze se základními prvky a za předpokladu zjednodušení mnoha dalších nelineárních jevů, které bychom museli řešit, pokud bychom se zabývali pouze samotnou regulací škrťící klapky. V modelu je zaveden offset střídý, při požadavku na střídý 0 není na svorkách nulové napětí, ale plné záporné napětí. Střídý 512 (50 %) je nulové napětí a střídý 1023 (100 %) znamená plné kladné napětí na výstupu. Použitý model škrťící klapky je v podstatě lineární. Aby byl model alespoň částečně reflektoval nelinearity škrťící klapky, je v modelu také zabudován blok obsahující statickou charakteristiku klapky (vliv pružiny). Tím by se měl model alespoň částečně přibližovat realitě.

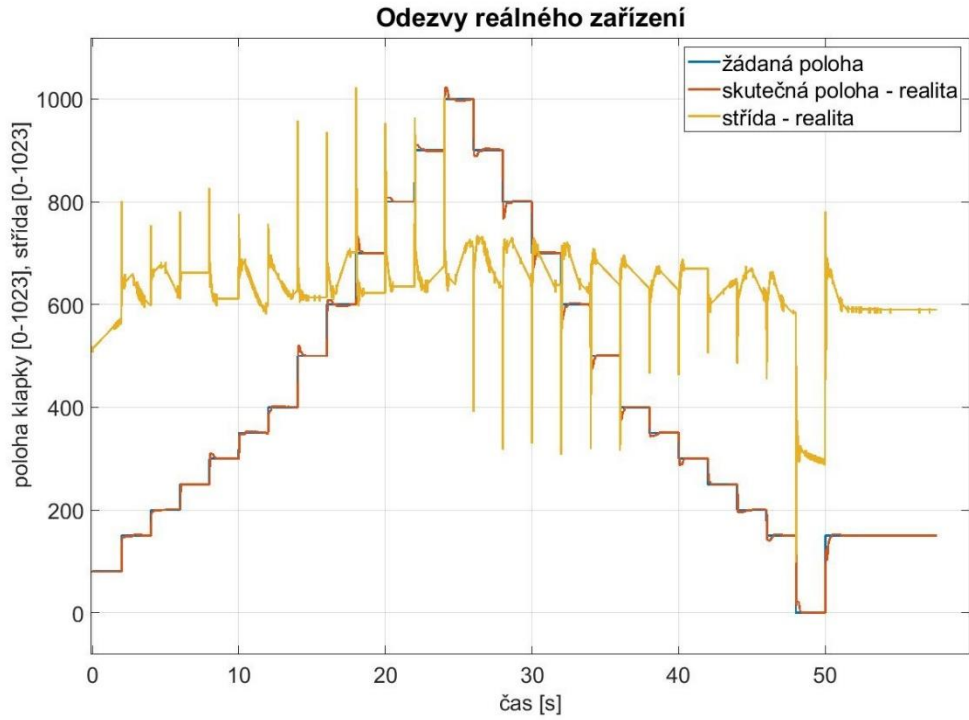


Obr. 47: Model soustavy v Simulinku

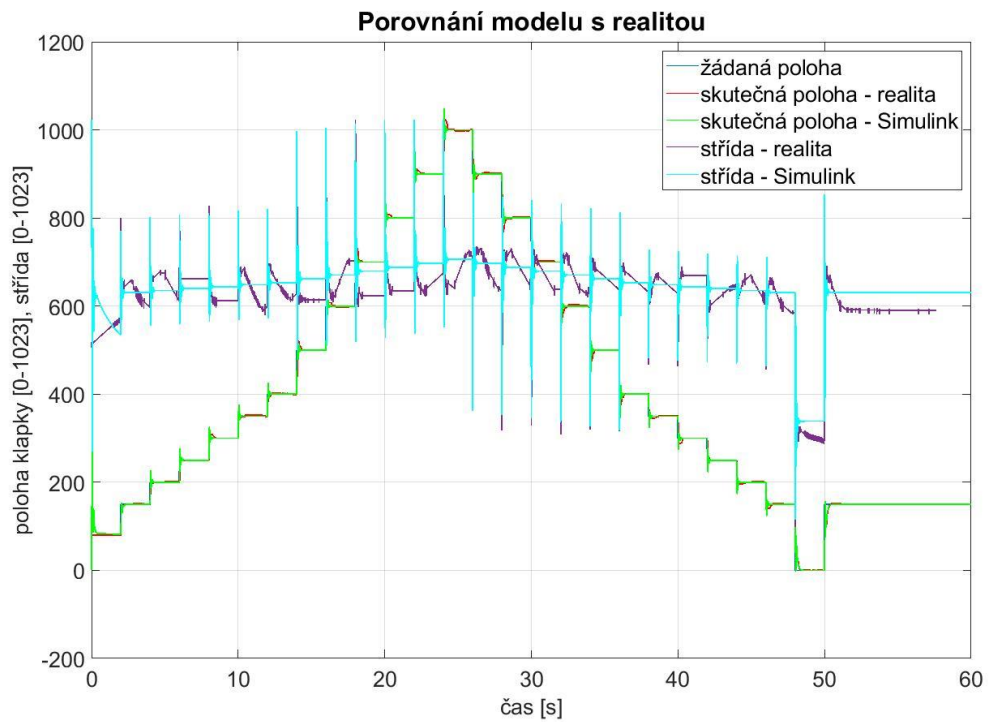
Na Obr. 48 je vidět vliv pružiny v okolí limp-home polohy, což je asi 8 % otevření klapky. Než motor překoná zátěžný moment pružiny, střída se mění zpočátku výrazně. Ke zvýšení střídy dojde až po nějakém čase, kdy se zvýší integrační složka a motor překoná zátěžný moment pružiny. Tento jev je doprovázen znatelným překmitem v oblasti zlomu statické charakteristiky z nelineární do lineární části. Je zde i patrné, že se zesílení v lineární části statické charakteristiky výrazně zvětšuje. V důsledku proměnlivého zesílení jsou patrné překmity při změně žádané hodnoty. Nejcitlivější poloha klapky je v malých úhlech otevření, kde se mění výrazně průtok vzduchu. Při velkých úhlech otevření již nedochází k tak markantním změnám v průtoku vzduchu. Řešení proměnlivého zesílení spočívá v implementaci adaptivního regulátoru s kompenzací nelinearit. Využití v testeru, který má pracovat s různými typy klapek je ale problematické. Je velmi pravděpodobné, že různé klapky budou mít různé parametry, tudíž by regulátor stejně nepracoval optimálně.



Obr. 48: Detail odezvy klapky z vozu Škoda Roomster BXW na skok žádané polohy

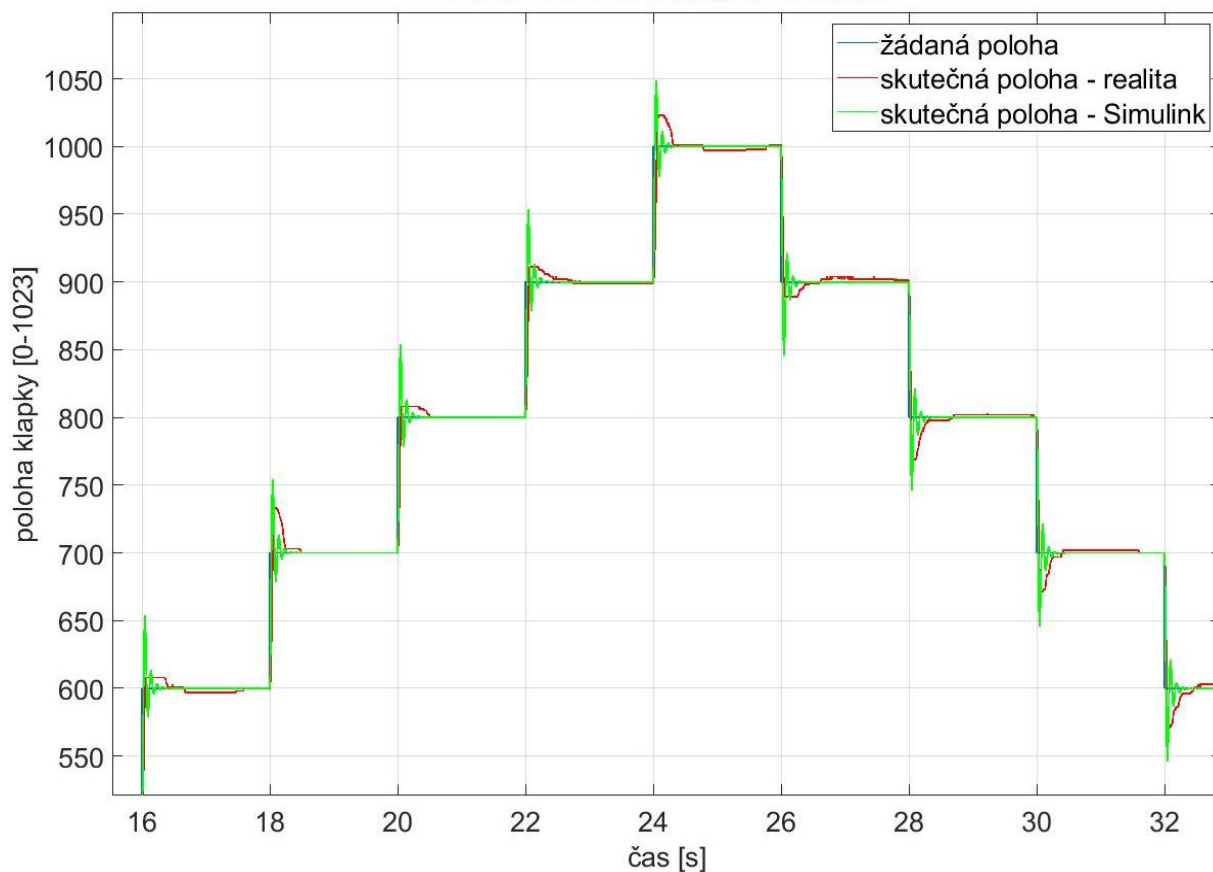


Obr. 49: Celkový průběh odezvy klapky z vozu Roomster BXW na žádanou polohu



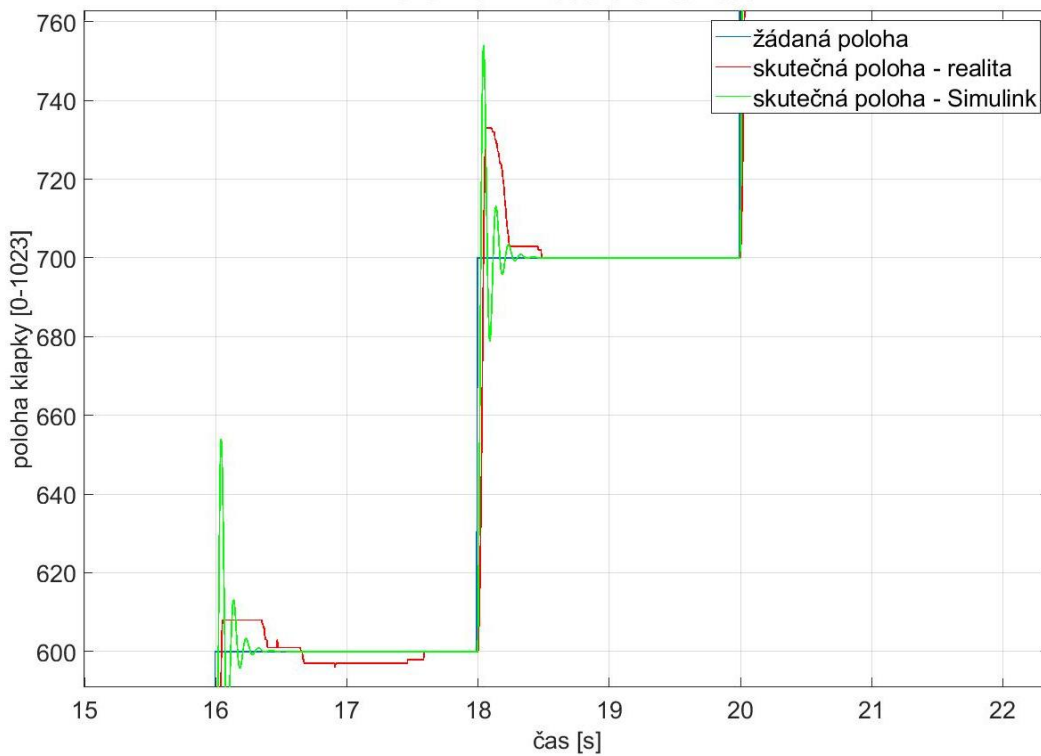
Obr. 50: Porovnání průběhu střídání a polohy klapky z vozu Roomster BXW s modelem v Simulinku

### Porovnání modelu s realitou



Obr. 51: Detail porovnání průběhu žádané a skutečné polohy klapky z vozu Roomster BXW s modelem v Simulinku

### Porovnání modelu s realitou



Obr. 52: Detail 2 porovnání průběhu žádané a skutečné polohy klapky z vozu Skoda Roomster BXW s modelem v Simulinku

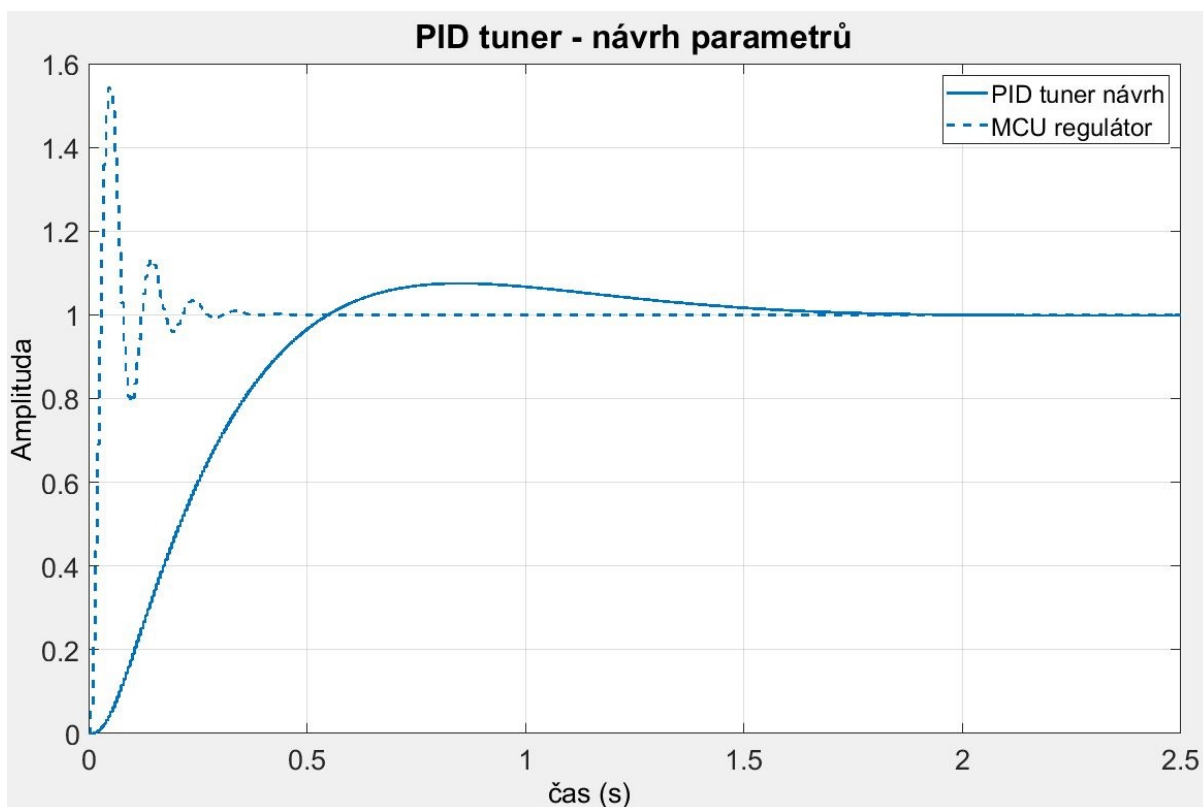
Regulátor v MCU má řešenou jinak derivační složku oproti modelu v Simulinku. Algoritmus v MCU počítá diferenci regulované veličiny, Simulink počítá diferenci regulační odchylky. Výhoda regulátoru implementovaného v MCU je ta, že nereaguje na skokové změny žádané hodnoty a tím by se měl zlepšit průběh regulace, náběhy na požadovanou hodnotu však budou o něco pomalejší. Pokud bychom chtěli ještě více omezit rychlé změny žádané hodnoty, můžeme v regulátoru zapnout filtr žádané hodnoty za předpokladu, že regulátor obsahuje integrační složku. Filtr žádané hodnoty omezuje spektrum přenášených frekvencí, a tedy zpomaluje reakci na žádanou hodnotu. Je nutno zvážit, co bude výhodnější vzhledem k režimu testování škrťících klapek.

Pro naladění dle modelu lze využít nástroj PID Tuner v Matlabu, který umí na základě požadavků na regulátor vypočítat parametry a porovnat je se stávajícím nastavením. Tyto nalezené hodnoty byly ručně doladěny na klapce z vozu Škoda Roomster kód motoru BXW, viz kapitola 1.1.1. V Tab. 13 je porovnání nastavení regulátoru v MCU a modelu v Simulinku. Nejedná se o hodnoty naměřené na reálném zařízení, doba náběhu, doba ustálení a překmit platí pouze pro model v Simulinku. Nalezené hodnoty PID tunerem mají výrazně delší dobu náběhu a dobu ustálení, avšak značně menší překmit, to je patrné na Obr. 53. Jak již bylo zmíněno v předchozím textu, model není dokonalý a regulátor neobsahuje kompenzaci nelinearity, který je v praxi nutný. Dále byly zanedbány některé odlišnosti v regulátoru MCU a Simulinku, i vlastnosti regulované soustavy. I přes tato zjednodušení však regulátor funguje poměrně dobře a bude zřejmě vyhovovat navrženému účelu.

Parametr	Návrh PID tuner	MCU regulátor
P	0,10333	3
I	0,36432	30
D	-0,0026247	0,03
N	44,0944	10
Doba náběhu [s]	0,36	0,015
Doba ustálení [s]	1,46	0,255
Překmit [%]	7,51	54,2

Tab. 13: Porovnání parametrů regulátoru MCU a výstupu z PID tuneru





Obr. 53: Porovnání aktuálního nastavení regulátoru v MCU s navrženými hodnotami PID tuneru

## 2.4 Řídící software

### 2.4.1 Uživatelské funkce

Tester s bipolární PWM je určen pro testování akčních členů jako jsou magnetické a proporcionální ventily, EGR ventily. Dále prvky obsahující stejnosměrné motory jako jsou škrtkové klapky, ventilátory, pohony stahování oken, palivová čerpadla. S omezením lze také testovat solenoidové vstřikovače v benzínových motorech nebo LPG vstřikovače. Případně také indukční cívky. Tester však nemá režim s parametry určenými přímo pro impulzní vládání těchto akčních členů. Uživatel má k dispozici buď manuální režim, kde si volí pomocí levého potenciometru frekvenci PWM od 10 Hz do 10 kHz a pravým potenciometrem je nastavena střída 0 až 100 %, nebo je možné zvolit některý z automatických režimů 2 až 8, viz Tab. 14. Tyto režimy mohou sloužit buď k ověření funkce, nebo například k rozhybání zatuhlého akčního členu v ultrazvukové čistící lázni. Levý a pravý potenciometr slouží k nastavení rychlosti změny, a to buď střídly, nebo frekvence, případně poměru sepnutí a vypnutí dle zvoleného automatického režimu. Funkce režimů 9 až 13 jsou buď informativní (9 a 10), nebo pro nastavení parametrů jako je velikost proudového omezení, zapnutí antiparalelní diody (pro širokou proporcionalitu u akčních členů) a režimy spínání H-můstku. Všechny režimy lze vybrat z menu, které se vyvolá stiskem tlačítka enkodéru. Po zapnutí je automaticky navolen režim 1, to je plně manuální ovládání frekvence a střídly PWM.

S testerem lze také emulovat signály čidel, například frekvenční signál váhy vzduchu (MAF), nebo ze snímače tlaku chladiva klimatizace. Dále je možno testovat automatické nastavení sklonu xenonových světlometů, správnou funkci tachometru atp.

<b>Režim</b>	<b>levý potenciometr</b>	<b>pravý potenciometr</b>	<b>Funkce</b>
<b>1</b>	Frekvence	Střída	Manuální režim zadávání střídání a frekvence
<b>2</b>	Rychlost změny frekvence	Rychlost změny střídání	Automatický režim Střída se mění v rozsahu 0-100 % Frekvence se mění v rozsahu 10 Hz-10 kHz Průběh pila
<b>3</b>	Frekvence	Rychlost změny střídání	Poloautomatický režim Střída se mění v rozsahu 0-100 % Frekvence manuálně
<b>4</b>	Rychlost změny frekvence	Střída	Poloautomatický režim Frekvence se mění v rozsahu 10 Hz-10 kHz
<b>5</b>	Frekvence	Rychlost změny střídání	Poloautomatický režim Střída se mění v rozsahu 0-100 % Průběh trojúhelník
<b>6</b>	perioda	poměr zapnutí/vypnutí	Zapíná a vypíná výstup v nastavených intervalech s měnitelným poměrem zapnutí a vypnutí
<b>7</b>	Rychlost změny žádané hodnoty polohy	-	Automaticky mění žádanou polohu klapky 0-100 % dle nastavení rychlosti Průběh pila
<b>8</b>	-	Žádaná hodnota polohy	Manuálně nastaví žádanou hodnotu polohy 0-100 %
<b>9</b>	-	-	Zobrazí hodnotu napájecího napětí
<b>10</b>	-	-	Detekuje dorazy a zobrazí napětí na horním a dolním dorazu škrtkovací klapky
<b>11</b>	-	Maximální hodnota proudu	V menu nastavuje velikost proudového omezení H-můstku
<b>12</b>	-	-	Zapne/Vypne antiparalelní diodu na unipolárním výstupu
<b>13</b>	-	-	Nastavuje režim spínání tranzistorů v H-můstku (vpřed, vzad, bipolárně)

Tab. 14: Režimy a funkce implementované v testeru

## 2.4.2 Použité knihovny

Řídící software pro tester byl napsán v Atmel studiu verze 7.0 jako projekt v jazyce C++ pro MCU ATmega328PB. Níže je seznam hlavičkových souborů použitých knihoven. Standardní knihovny jsou součástí instalace programu Atmel studio. [Autorem knihoven pro SMA filtr, obsluhu enkodéru a PSD regulátor je vedoucí práce pan Ing. Zdeněk Mašek, Ph.D.](#)

### Standardní knihovny GCC

- stdlib.h – standardní knihovna, deklaráce C maker a rozšíření pro AVR
- io.h – standardní knihovna, IO definice pro MCU
- delay.h – obsahuje funkce pro zpoždění
- interrupt.h – obsahuje funkce pro práci s přerušením
- pgmspace.h – obsahuje funkce pro přístup k datům v paměti flash
- inttypes.h – obsahuje makra pro konverze datového typu integer
- stdio.h – obsahuje funkce pro standardní vstup/výstup

### Externí knihovny třetích stran

- SBIT.h – obsahuje makro pro přehlednější práci s porty
- OLED.h – obsahuje funkce pro obsluhu OLED displeje
- I2C.h – obsahuje funkce pro obsluhu sběrnice I<sup>2</sup>C
- PCF8574.h – obsahuje funkce pro obsluhu I/O expandéru PCF8574
- pins\_arduino.h – obsahuje definice vstupů a výstupu mikrokontroleru
- sma.h – obsahuje funkce pro Simple Moving Average filtr
- Arduino.h – obsahuje funkce pro zjednodušení obsluhy I/O a čítačů/časovačů
- Adafruit\_ADS1015.h – obsahuje funkce pro obsluhu externího převodníku ADS1015
- uart.h – obsahuje funkce pro odesílání dat přes UART
- stdinout\_to\_uart.h – obsahuje funkce pro přeměrování stdin a stdout na UART
- PSD\_polohovy.h – obsahuje funkce pro obsluhu číslicového PID regulátoru v polohovém tvaru
- mechEncoder.h – obsahuje funkce pro obsluhu mechanického enkodéru

### Speciální nastavení projektu

- Symbol  $F\_CPU=16000000$  nastaven pro C i C++ kompilátor
- Použití vprintf knihovny – nastavení linkeru
- Použití knihoven v linkeru libprintf\_flt, libscanf\_flt
- Optimalizace C++ kompilátoru nastavena na úroveň –Og
- Optimalizace C kompilátoru nastavena na úroveň –O1

### 2.4.3 Použité periférie MCU a jejich parametry

Použité periférie byly definovány na základě potřeb pro řízení H-můstku, pak podle použitých ovládacích prvků na stávajícím testeru akčních členů i nově přidaných. Počty tlačítek i periférií mimo MCU přesahuje potřebu aktuálního využití v programu. Je to rezerva pro funkce, které mohou být implementovány v budoucnu při dalším vývoji testeru. Protože nebylo dostatek vstupů/výstupů na MCU z důvodu použití speciálních nebo alternativních funkcí řady pinů, byl použit obvod PCF8574, který komunikuje po I<sup>2</sup>C sběrnici a umožňuje přidat dalších 8 vstupů/výstupů. Přes sběrnici I<sup>2</sup>C komunikuje i OLED displej a externí AD převodník ADS1015. Omezujícím faktorem bylo použití výstupních pinů pro 16bitový časovač/čítač TC1, který je použitý pro generování frekvence PWM. Dalším dva 8bitové časovače/čítače TC0 a TC2 jsou také použity. V podstatě všechny periférie MCU jsou použity.

Periférie	Funkce
8bitový časovač/čítač TC0	Časování aplikace, základní jednotka času je 4,992 ms
16bitový časovač/čítač TC1	Generování PWM 10 Hz až 10 kHz pro řízení H-můstku
8bitový časovač/čítač TC2	Generování PWM pro D/A převodník proudovou regulací H-můstku (referenční napětí $V_{REF}$ )
A/D převodník	Měření proudu H-můstkem, čtení hodnot ovládacích potenciometrů
I <sup>2</sup> C sériové rozhraní (TWI)	Komunikace s displejem, I/O expandérem a externím A/D převodníkem

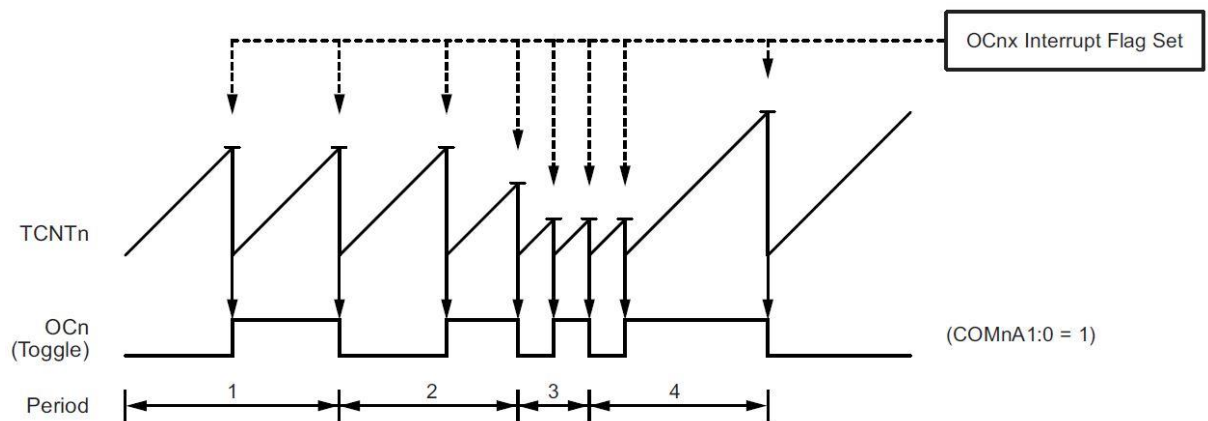
Tab. 15: Seznam použitých periférií MCU

Přerušení	Funkce
INT0	Detekce nadproudu H-můstku A
INT1	Detekce nadproudu H-můstku B
PCINT2	Vyhodnocení směru otáčení a počtu kroků mechanického enkodéru
TIMER0 COMPA	K přerušení dochází každých 4,992 ms, inkrementuje proměnnou pro časování aplikace v násobcích 4,992 ms. Vykonává PID regulátor.
TIMER1 OVF	V obsluze přerušení časovače/čítače TC1 dochází k aktualizaci frekvence PWM. Dále řeší odebírání vzorků pro měření proudu.

Tab. 16: Seznam použitých přerušení MCU

### 2.4.3.1 Časovač/čítač TC0

Tento 8bitový časovač/čítač je použit pro časování celé aplikace. Pracuje v režimu CTC, komparační (top) hodnota je nastavena v komparačním registru OCR0A. K vynulování čítače dochází při shodě TCNT0 s komparační hodnotou OCR0A, viz Obr. 54. Na rozdíl od zbylých dvou časovačů zde není potřeba výstup na pin mikrokontroleru, zpracovává se pouze přetečení časovače pomocí přerušování při komparaci.



Obr. 54: Časový diagram časovače/čítače TC0 v režimu CTC [18]

Původně bylo zamýšleno použít jako základní periodu časování aplikace 1 ms. Nicméně kvůli problémům s periodou regulátoru byla změněna základní jednotka času na 5 ms. Hodnota není přesná, dle výpočtu podle vzorce (46) je to 4,992 ms. Kvůli použité děličce a krystalu 16 MHz nevychází komparační hodnota jako celé číslo.

Komparační hodnotu pro OCR0A registr vypočteme podle vzorce (44)(45) jako:

$$OCR0A = t * \frac{f_{CLK}}{N} - 1 = 0,005 * \frac{16 \cdot 10^6}{1024} - 1 = 77,125 \quad (45)$$

Kde  $t$  je požadovaný časový interval [s],  $f_{CLK}$  taktovací frekvence mikrokontroleru [Hz],  $N$  je hodnota předděličky časovače [-].

Protože výsledek není celočíselný, přesnou hodnotu časového intervalu časovače pro komparační hodnotu  $OCR0A = 77$  vypočteme jako:

$$t = N * \frac{OCR0A + 1}{f_{CLK}} = 1024 * \frac{77 + 1}{16 \cdot 10^6} = 0,004992 \text{ s} \quad (46)$$

Dělička 1024 byla zvolena z důvodu jediné možnosti dosažení 5 ms intervalu, resp. 4,992 ms na 8-bit časovači (čítací rozsah 0 až 255) při taktovací frekvenci MCU 16 MHz.

<b>N</b>	1	8	64	256	1024
<b>OCR0A</b>	79999	9999	1249	311,5	77,125

Tab. 17: Závislost komparační hodnoty na dělicím poměru předděličky pro 5 ms periodu

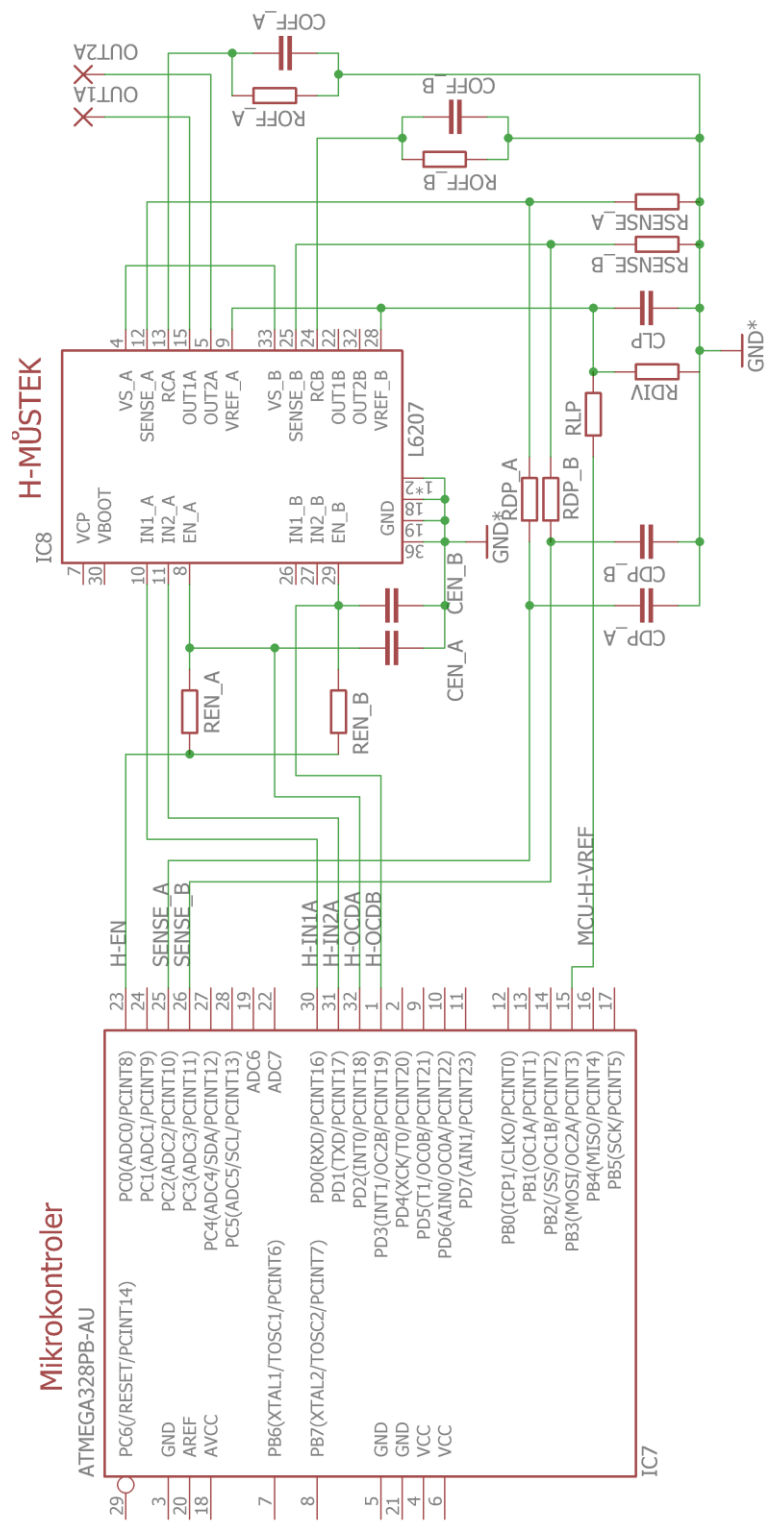
Nastavení časovače TC0 zajišťuje funkce `ms_timer_init()`, jejíž kód je níže.

```
TCCR0A = (1<<WGM01);           // CTC
OCR0A = 77;                     // komparační hodnota pro 4,992 ms
TIMSK0 = (1<<OCIE0A);         // povolení přerušení
TCCR0B = (1<<CS02)|(1<<CS00);  // fclk/1024 spuštění časovače
```

### 2.4.3.2 Časovač/čítač TC2

Protože MCU nemá interní D/A převodník, bylo nutno realizovat jednoduchý D/A převodník pomocí tohoto 8bitového čítače/časovače a RC článku, který tvoří filtr typu dolní propust. Ten zajistí dostatečné vyhlazení referenčního napětí pro regulaci proudu H-můstku. Detail připojení H-můstků k mikrokontroleru je na Obr. 35 a Obr. 55. Výsledná velikost referenčního napětí je dána vzorcem (22).

Jedná se o druhý 8bitový časovač/čítač, který je použit pro generování PWM pro vytvoření jednoduchého D/A převodníku pro nastavení proudového omezení H-můstku. V tomto případě potřebujeme pevnou frekvenci a budeme měnit pouze střidu. Pro tuto funkci byl vybrán režim 1 – PWM Phase Correct. TOP hodnota je pevně daná 0xFF, střida se určuje nastavením registru OCR2A.



Obr. 55: Zjednodušené schéma připojení IO L6207 k MCU



Dle datového listu [18] použijeme vzorec pro výpočet frekvence PWM:

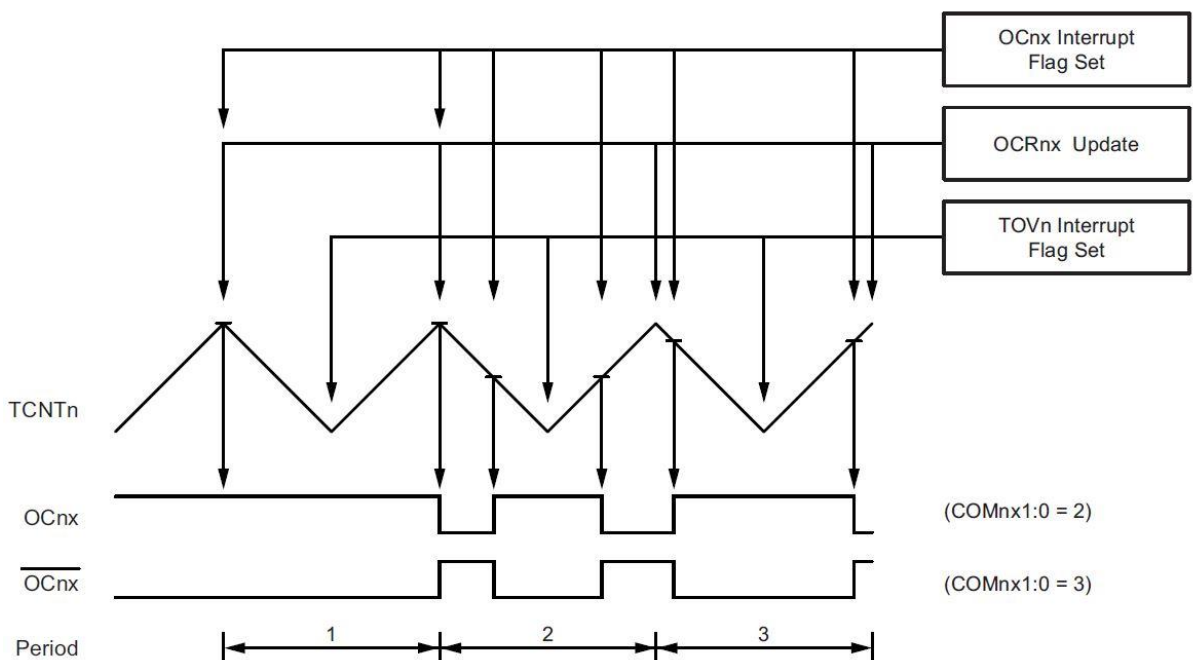
$$f_{PWM} = \frac{f_{CLK}}{N * 510} \quad (47)$$

Kde  $f_{PWM}$  je frekvence PWM signálu [Hz],  $f_{CLK}$  taktovací frekvence mikrokontroleru [Hz],  $N$  dělicí poměr předděličky čítače/časovače [-]. Nejvyšší frekvence v tomto režimu je tedy možno dosáhnout nastavením dělicího poměru 1.

Výsledná frekvence potom bude:

$$f_{PWM} = \frac{16 * 10^6}{1 * 510} = 31,372 \text{ kHz} \quad (48)$$

Na základě simulace v programu LTspice má RC filtr na vstupu H-můstku časovou konstantu  $\tau_{DP} 0,18 \text{ ms}$ . Této časové konstantě odpovídá zlomová frekvence  $f_0=885 \text{ Hz}$ . S těmito parametry se zvlnění na vstupu H-můstku pohybuje okolo 40 mV, to odpovídá proudu asi 0,2 A. Tato hodnota je poměrně velká vzhledem k rozsahu nastavení proudového omezení. Nižšího zvlnění referenčního napětí lze dosáhnout buď zvýšením časové konstanty RC filtru, nebo použitím režimu Fast PWM, se kterým je možné dosáhnout dvojnásobné frekvence  $f_{PWM}=62,5 \text{ kHz}$ . Průběh na Obr. 56 ukazuje časový diagram průběhů čítače/časovače v režimu Phase Correct PWM.



Obr. 56: Časový diagram časovače/čítače TC2 v režimu Phase Correct PWM [18]

### 2.4.3.3 Časovač/čítač TC1

Časovač/čítač TC1 je jediný 16bitový, kterým MCU disponuje. Tento časovač/čítač je použitý na generování spínací frekvence PWM pro H-můstek. Zadáním je určen rozsah frekvence PWM 10 Hz až 10 kHz. Protože je potřeba měnit jak střídu, tak frekvenci, zužuje to výběr z možných režimů časovače. Datasheet MCU [18] doporučuje pro řízení motorů použít Phase and Frequency Correct PWM režim 8. Ten jako jediný umožňuje měnit současně střídu i frekvenci bez negativních projevů v generování PWM. U tohoto módu je TOP hodnota, která určuje výstupní frekvenci, nastavena registrem ICR1. Střída se nastavuje pomocí komparačních registrů OCR1A, respektive OCR1B. Compare událost nastává při čítání nahoru i dolů, jedná se o tzv. dual slope režim. Jsou použity oba výstupy OC1A i OC1B, které jsou vůči sobě inverzní. Detail připojení H-můstku k mikrokontroleru je na Obr. 55 Bližší vysvětlení nutnosti dvou výstupů je uvedeno v kapitole 2.4.8 Generování PWM.

Dle datového listu [18] použijeme vzorec pro výpočet frekvence PWM:

$$f_{PWM} = \frac{f_{CLK}}{2 * N * TOP} \quad (49)$$

Kde  $f_{PWM}$  je frekvence PWM signálu [Hz],  $f_{CLK}$  taktovací frekvence mikrokontroleru 16 MHz, N dělicí poměr předděličky čítače/časovače [-], TOP je top hodnota časovače/čítače [-].

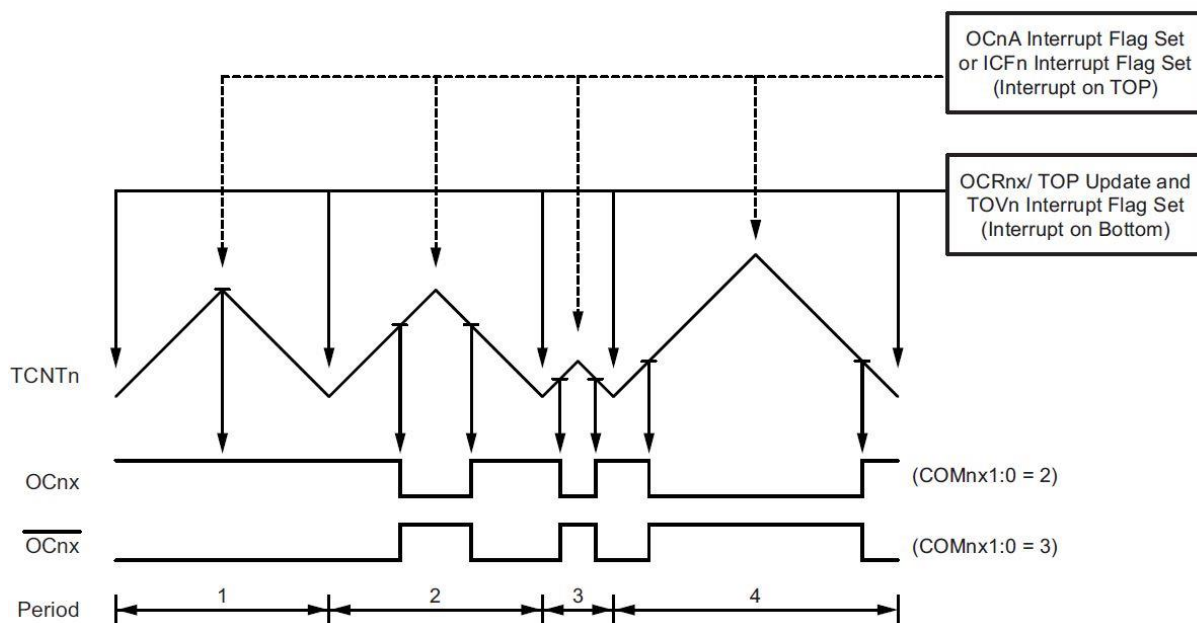
Hledáme tedy vhodné nastavení předděličky a TOP hodnoty:

$$TOP = \frac{f_{CLK}}{2 * N * f_{PWM}} \quad (50)$$

N	TOP 10 KHz	TOP 100 Hz	TOP 10 Hz
1	800	80000	800000
8	100	10000	100000
64	12,5	1250	12500
256	3,125	312,5	3125
1024	0,78125	78,125	781,25

Tab. 18: Závislost TOP hodnoty na dělicím poměru předděličky

Z Tab. 18 výše je patrné, že rozsah frekvence PWM 10 Hz až 10 kHz nelze dosáhnout s jednou hodnotou předděličky. Proto je programově nutno přepínat hodnotu předděličky N = 8 a N = 64 na přelomu první a druhé dekády. Časovač/čítač TC1 má povolenou reakci na přerušení přetečení, jehož význam bude vysvětlen později.



Obr. 57: Časový diagram časovače/čítače TC1 v režimu Phase and Frequency Correct PWM [18]

#### 2.4.3.4 I<sup>2</sup>C Sériové rozhraní

Data na této sběrnici jsou přenášena sériově 8bitovými slovy, používá se pro komunikaci mezi mikrokontrolerem a podřízenými obvody. Všechny obvody jsou připojeny na jedné sběrnici, mají svou unikátní 7bitovou nebo 10bitovou adresu přenášenou společně s daty a jsou v topologii Master/Slave nebo Multimaster. Maximální počet zařízení je omezen pouze celkovou maximální kapacitou sběrnice 400 pF. Sběrnice je dvou vodičová, první vodič je označený jako SDA = serial data, druhý jako SCL = serial clock.

Adresa každého obvodu je složena z pevné části určené výrobcem a volitelné části určené logickými úrovněmi na adresních pinech konkrétní součástky. Maximální komunikační rychlost sběrnice se pohybuje od 100 kbps až po 3,4 Mbps. [19]

Výrobce mikrokontroleru Atmel označuje tuto sběrnici jako TWI – Two Wire Interface. U AVR je možno zvolit, zda se má chovat jako master nebo slave. Adresa je 7bitová a maximální podporovaná přenosová rychlost je 400 kbps. ATmega328P má HW řadič I<sup>2</sup>C, není nutno tedy protokol I<sup>2</sup>C emulovat softwarově. Piny PC4 a PC5 mají přidruženou funkci SDA a SCL. Pro obsluhu komunikace po sběrnici I<sup>2</sup>C existuje mnoho knihoven, je nutno pouze vybrat nejvhodnější a nastavit základní parametry, jako je rychlost sběrnice.

Na tuto sběrnici je připojen OLED displej, externí A/D převodník ADS1015 a integrovaný obvod PCF8574T, který slouží jako expandér vstupů a výstupů. Všechna zařízení komunikují rychlostí 400 kbps.

### 2.4.3.5 Interní A/D převodník MCU

ATmega328 má 9 multiplexovaných kanálů ADC0 až ADC8, kde poslední z nich je interně připojený k teplotnímu senzoru uvnitř pouzdra. Vstupy ADC2 a ADC3 jsou použity pro měření úbytku napětí na bočnicích H-můstku (měření proudu). Kanály ADC4 a ADC5 mají alternativní funkci jako SDA a SCL, takže je nelze pro A/D převod použít. Pro čtení hodnot ovládacích potenciometrů slouží vstupy ADC6 a ADC7. Jako zdroj referenčního napětí je možno vybrat buď interní zdroj 1,1 V, nebo externí  $0 \text{ V} - V_{cc}$ . Při maximálním rozlišení 10 bitů je nejvyšší vzorkovací frekvence jednoho kanálu 15 ksp/s. Tato rychlost s počtem připojených kanálů úměrně klesá, při dvou to je polovina, při třech třetina atd. Maximální taktovací frekvence převodníku by se měla pohybovat mezi 50 až 200 kHz v případě potřeby největšího rozlišení. Pokud postačí menší rozlišení, je možno taktovací frekvencí zvýšit nad 200 kHz. Tím lze zvýšit rychlost převodu, a tedy snížit potřebný čas pro převod. První konverze trvá 25 cyklů, další pak 13.

Napětí z převedené hodnoty vypočítáme dle vzorce (51) níže [18]:

$$U_{IN} = \frac{ADC * U_{REF}}{2^N} \quad (51)$$

Kde  $U_{IN}$  je napětí [V] na analogovém vstupu A/D převodníku, ADC je napětí převedené na číslo [-] v rozsahu  $2^N-1$ ,  $U_{REF}$  je referenční napětí [V] A/D převodníku, N je počet bitů A/D převodníku.

Volba referenčního napětí určuje rozsah A/D převodníku, napětí na analogovém vstupu musí mít rozsah  $0-U_{REF}$  [V]. Referenční napětí by tedy mělo pro plné využití rozsahu A/D převodníku odpovídat velikosti měřeného napětí na analogovém vstupu. Výběr zdroje napěťové reference se provádí v registru ADMUX nastavením bitů REFS0 a REFS1. Další parametry A/D převodníku se nastavují v registru ADCSRA.

Pro čtení hodnot ovládacích potenciometrů na vstupech ADC6 a ADC7 můžeme využít plné 10bitové rozlišení A/D převodníku a jako referenci  $V_{cc}$ , což je 5 V. Využijeme režim single conversion, což znamená ruční spuštění převodu a vyčkání na výsledek. Délka převodu není v tomto případě kritická, dělička je nastavena na 64, při taktovací frekvenci MCU 16 MHz je A/D převodník taktován na 250 kHz.

Délku převodu potom určíme jako:

$$t = \frac{1}{f_{CLKADC}} * K \quad (52)$$

Kde  $t$  je délka převodu [s],  $f_{CLKADC}$  je taktovací frekvence A/D převodníku [Hz],  $K$  je počet cyklů na převod.

Pro nejkritičtější případ, kdy převod trvá 25 cyklů je tedy délka jednoho převodu:

$$t = \frac{1}{250 * 10^3} * 25 = 0,1 \text{ ms} \quad (53)$$

Zpočátku bylo plánováno, že se bude měřit střední hodnota proudu. To klade vysoké nároky na rychlost vzorkování zejména při vysokých frekvencích PWM. Při vyšších taktovacích frekvencích převodníku již nelze používat 10bitové rozlišení z důvodu nepřesnosti, ale jen 8bitové rozlišení. V průběhu vývoje byl změněn požadavek a měří se zatím jen maximální hodnota proudu, a to synchronizovaně s periodou PWM (detaily jsou uvedeny v kapitole 2.4.10). Pro tyto účely je převodník taktován na 1 MHz. Převod potom trvá podle (53) 25  $\mu$ s. Při maximální frekvenci PWM 10 kHz je možno změřit asi 8 vzorků v jedné periodě PWM, při frekvenci 100 Hz je to pak 800 vzorků. Pod 100 Hz se proud neměří.

V programu nastavuje A/D převodník funkce `void SetupADC(uint8_t prescaler, uint8_t lowRes)` a o vlastní čtení analogových vstupů se pak stará funkce `uint16_t Read_adc(uint8_t channel)`.

### 2.4.3.6 Použitá přerušení

#### INT0

Externí přerušení INT0 reaguje na sestupnou hranu na pinu PD2. Tento port je připojen na EN\_A vstup H-můstku A. Interní logika v H-můstku A ho uzemní v případě, že detekuje nadproud nebo přehřátí v H-můstku A.

#### INT1

Externí přerušení INT1 reaguje na sestupnou hranu na pinu PD3. Tento pin je připojen na EN\_B vstup H-můstku B. Interní logika v H-můstku B ho uzemní v případě, že detekuje nadproud nebo přehřátí v H-můstku B.

## **PCINT2**

Externí přerušení PCINT2 reaguje na změnu logické úrovně na pinech PD6 a PD7. U tohoto přerušení je třeba specifikovat v registru PCMSK2 konkrétní piny, na které má přerušení reagovat. Konkrétně se jedná o bity PCINT22 a PCINT23. Na tyto piny je připojen enkodér a v obsluze přerušení je vyhodnocován směr otáčení a počet kroků.

## **TIMER0 COMPA**

Přerušení je generováno časovačem/čítačem TC0, jakmile je hodnota v registru TCNT0 shodná s hodnotou v komparačním registru OCR0A (Compare match A). V obsluze tohoto přerušení je inkrementována každých 4,992 ms proměnná *t\_ms* použitá pro časování v programu. Dále je v obsluze přerušení vykonáván kód pro PID regulátor kvůli zajištění přesné periody vykonávání PID regulátoru 4,992 ms, viz kapitola 2.4.3.1 Časovač/čítač TC0.

## **TIMER1 OVF**

Přerušení generuje časovač/čítač TC1 při přetečení (Overflow). TC1 vytváří spínací frekvenci PWM pro řízení H-můstku V obsluze přerušení dochází k aktualizaci ICR1 (Input Capture Register) registru 16bitového čítače/časovače TC1, pomocí kterého se nastavuje výstupní frekvence PWM pro H-můstek. Vzhledem k tomu, že registr ICR1 není double buffered, došlo by při jeho nastavení ke změně frekvence okamžitě. Proto je nutno čekat až na konec periody, při kterém je vykonáno přerušení při přetečení, viz Obr. 57. Krátce po přerušení nastane na počátku další periody změna TOP hodnoty v registru ICR1. Tu je potřeba nastavit co nejrychleji, aby čítač udělal co nejméně tiků. Přerušení se po vykonání samo zakáže, je povolováno cyklicky v hlavní smyčce programu dle času určeném parametrem *PWM\_UPDATE\_PERIOD\_MS*. Tento parametr je nastaven tak, že momentálně dochází k přerušení přibližně každých 80 ms. V obsluze přerušení se také opět povolí přerušení *TIMER0\_COMPA* pro časovač/čítač TC0, které se zakazuje před aktualizací střídý a frekvence prostřednictvím hodnot v registrech OCR1A, OCR1B a ICR1, aby časovač/čítač TC0 pokračoval v inkrementování proměnné *t\_ms*, viz odstavec výše.

Poslední část kódu v obsluze přerušení *TIMER1\_OVF\_vect* řeší odebírání vzorků napětí v průběhu jedné periody PWM na bočníku H-můstku pro měření proudu, případně nastaví proměnnou *mer\_proud\_v\_mainu* na 1, pomocí které se spustí odebírání vzorků napětí na bočníku v hlavní smyčce programu v případě, že by trval odběr všech vzorků v přerušení příliš dlouho, více informací o způsobu měření proudu H-můstkem je uvedeno v kapitole 2.4.10 Měření proudu.

## 2.4.4 Ovládací prvky

Původní verze testeru měla pro ovládání jen dva potenciometry a jedno tlačítko. Použití jiného procesoru a přidání displeje umožnilo rozšíření možností ovládání testeru. Rozmístění ovládacích prvků je patrné na Obr. 19. Na piny PB5, PC1 a PD5 jsou připojena tlačítka. Na piny PD6, PD7 a PD4 je připojen enkodér s tlačítkem. Potenciometry jsou připojeny k analogovým vstupům ADC6 a ADC7. Jsou připojeny na 5 V, tudíž je nutno mít referenční napětí A/D převodníku nastaveno na  $V_{cc}^5$ , viz kapitola 2.4.3.5 Interní A/D převodník MCU. Potenciometry jsou použité pro nastavení frekvence a střidy PWM signálu pro ovládání H-můstku a v poloautomatických režimech se pomocí nich nastavují rychlosti změny dané veličiny, a to buď střidy, frekvence, nebo poměru zapnutí a vypnutí H-můstku. Dále se pomocí potenciometru nastavuje v menu velikost proudového omezení a žádaná hodnota [0-100 %] v případě zpětnovazebního řízení škrťací klapky v manuálním režimu. Čtení hodnot z potenciometrů zajišťuje funkce `uint16_t Read_adc(uint8_t channel)`. Výstup je použitý většinou 10bitový, pouze v případě nastavení proudového omezení je použito 8bitové rozlišení kvůli vyšší rychlosti A/D převodu. Hodnota 0-1023 je v některých případech v poloautomatických režimech interpolována do potřebného rozsahu funkcí `Map`, viz ukázka kódu níže:

Vstup: `x, in_min, in_max, out_min, out_max`

Výstup: Přemapovaná hodnota `x` z rozsahu `in_min` a `in_max` do rozsahu `out_min, out_max`

```
long Map(long x, long in_min, long in_max, long out_min, long
out_max
{
    return (x - in_min) * (out_max - out_min) / (in_max - in_min) +
out_min;
}
```

```
//Čtení hodnot kanál 6 ADC potenciometr vpravo-střída
```

```
strida_adc = Read_adc(POT_DUTY);
```

```
// interpolace strida_adc z převodníku [0-1023] do hodnoty [0-top]
```

```
ocr = Map(strida_adc,0,1023,0,top);
```

---

<sup>5</sup> Bylo by možné použít dělič a referenční napětí 1,1 V. A/D převodník by byl lépe využit při měření proudu.

Aktuálně se v programu používá pouze jedno tlačítko, a to na spuštění a zastavení aktuálního režimu. Byla prozatím zachována i funkce hold, při které je možné zvolit režim tím samým tlačítkem, respektive s každým stiskem tlačítka se číslo režimu zvyšuje o jedna. Po vypršení časové lhůty 2 s, která se zvyšuje s každým s tiskem tlačítka o 500 ms se aktivuje vybraný režim. Tato funkcionalita bude pravděpodobně vhodná pro low-end verzi testeru bez displeje po další konzultaci se zadavatelem projektu.

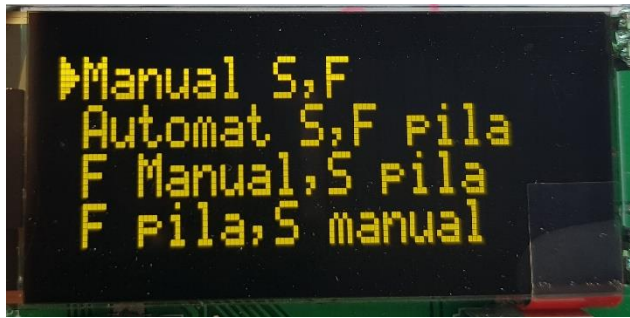
Enkodér byl přidán jako ovládací prvek vhodný pro pohyb v menu na displeji, případně pro další specifické režimy, například krokování. Použitý mechanický enkodér má tři výstupy (stopa A, stopa B, tlačítko). Z pohledu zpracování v programu je nutné detekovat změnu logické úrovně na obou stopách enkodéru a zjistit, na kterou stranu se otáčí. Tato funkce je umístěna v obsluze přerušení `PCINT2`. Každá změna na pinu PD6 nebo PD7, kam je enkodér připojen, vyvolá toto přerušení, ve kterém se čte stav enkodéru (využívá se knihovna `mechEncoder`) a aktualizuje se globální proměnná `steps`, která v sobě nese číselnou informaci o počtu vykonaných kroků vlevo nebo vpravo. Globální proměnná `steps` pak vrací hodnotu o počtu vykonaných kroků vlevo či vpravo. Prostřední tlačítko enkodéru slouží jako potvrzovací v menu a jako ukončení menu. Rozhraní pro nastavení parametrů z pohledu uživatele zajišťuje funkce `void Menu(void)`. Ta je vyvolána stiskem tlačítka enkodéru. Tato funkce zobrazí nabídku, ve které je možno se pohybovat vpřed nebo vzad otočením enkodéru. Šipka na první pozici řádku se přesouvá po řádcích s každým otočením enkodéru vlevo nebo vpravo a tím označuje aktuálně vybranou položku v menu. To je vícestránkové a umožňuje vnoření více úrovní, což je využito aktuálně pro nastavení typu PWM

Obsah funkce obsluhy přerušení `PCINT2_vect` použitého pro čtení stavu enkodéru:

```
uint8_t pindNewState = PIND;
uint8_t pindChanged = pindNewState^pindOldState; // který pin na
portu se změnil
pindChanged      &= PCMSK2;      //který pin s funkcí PCINT způsobil
přerušení PCINT2
pindOldState = pindNewState;    // stav na portu pro příští detekci
změny

// pokud přerušení způsobil mech. enkodér, tak je obslouženo, reakce
na obě hrany na obou stopách
if ((pindChanged == (1<<ENC_A_PIN) || pindChanged == (1<<ENC_B_PIN))
{
    pindNewState >>= ((ENC_A_PIN < ENC_B_PIN) ? ENC_A_PIN :
ENC_B_PIN);
    steps += readMechEncoder(pindNewState);
}.
```





Obr. 58: Ukázka první strany menu na displeji

U všech mechanických spínačů je potřebné ošetřit zákmity, které vznikají při rozpínání a spínání kontaktů a způsobují falešné impulzy pro MCU. Zvolená metoda řešení ukládá čas stisku tlačítka a jeho poslední stav do proměnných. Při dalším průchodu programem se otestuje znovu stav tlačítka, zda bylo uvolněno a zda předchozí stav bylo stisknuto. Následně se od aktuálního času odečte čas stisknutí, výsledný čas musí být větší než nastavený čas zákmitu. Poté dojde k nastavení proměnné *single* signalizující krátký stisk tlačítka. V případě dlouhého stisku tlačítka je pak testováno, zda aktuální čas mínus čas zapnutí je větší než definovaný čas pro dlouhý stisk tlačítka. V tom případě se nastaví proměnná *hold* nesoucí informaci o dlouhém stisku. Celou obsluhu čtení tlačítek má na starosti funkce `void ReadBTN(uint8_t counter)` volaná s periodou přibližně 10 ms, viz kapitola 2.4.5 Hlavní smyčka a časování aplikace.

Níže je ukázka části kódu pro ošetření zákmitu tlačítka BTN1 připojeného na pin PB5.

```
// Tlačítko stisknuto a poslední stav byl klidový
if (BTN1 == 0 && lastReading == 1)
{
    onTime = millis(); //Ulož čas stisknutí tlačítka
}

// Tlačítko uvolněno a předtím bylo stisknuto
if (BTN1 == 1 && lastReading == 0)
{
    // Aktuální čas-čas stisknutí>zákmit filtr a není hold
    if (((millis() - onTime) > bounceTime) && hold != 1)
    {
        single = 1;

        // Čas Posledního stisku = aktuální čas
        lastSwTime = millis();
    }
}
```

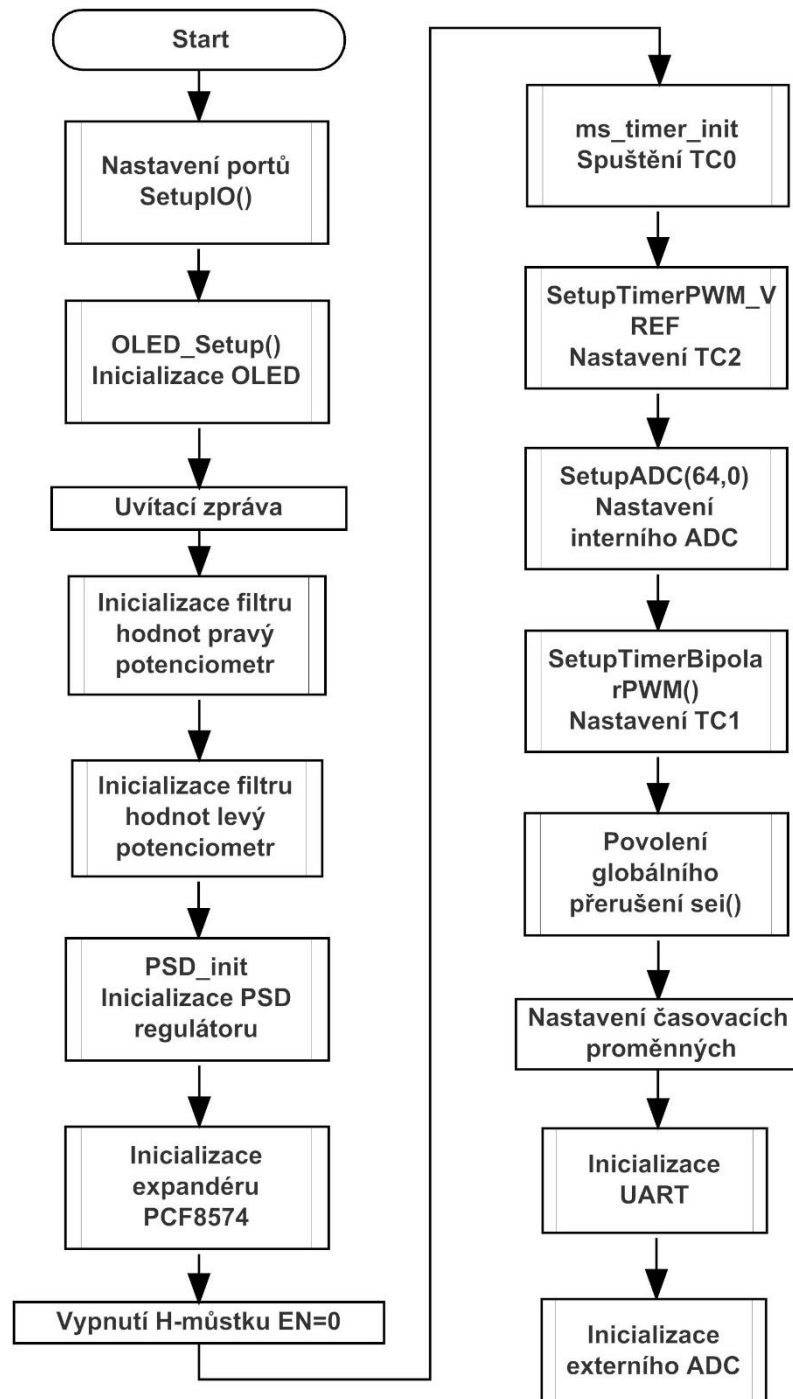
Následující tabulka popisuje režimy a funkce aktuálně implementované v testeru. V případě potřeby lze v kódu poměrně jednoduše další funkce přidat. Omezením je pouze velikost paměti MCU.

Režim	levý potenciometr	pravý potenciometr	Funkce
1	Frekvence	Střída	Manuální režim zadávání střídy a frekvence
2	Rychlost změny frekvence	Rychlost změny střídy	Automatický režim Střída se mění v rozsahu 0-100 % Frekvence se mění v rozsahu 10 Hz-10 kHz Průběh pila
3	Frekvence	Rychlost změny střídy	Poloautomatický režim Střída se mění v rozsahu 0-100 % Frekvence manuálně
4	Rychlost změny frekvence	Střída	Poloautomatický režim Frekvence se mění v rozsahu 10 Hz-10 kHz
5	Frekvence	Rychlost změny střídy	Poloautomatický režim Střída se mění v rozsahu 0-100 % Průběh trojúhelník
6	perioda	poměr zapnutí/vypnutí	Zapíná a vypíná výstup v nastavených intervalech s měnitelným poměrem zapnutí a vypnutí
7	Rychlost změny žádané hodnoty polohy	-	Automaticky mění žádanou polohu klapky 0-100 % dle nastavení rychlosti Průběh pila
8	-	Žádaná hodnota polohy	Manuálně nastaví žádanou hodnotu polohy 0-100 %
9	-	-	Zobrazí hodnotu napájecího napětí
10	-	-	Detekuje dorazy a zobrazí napětí na horním a dolním dorazu škrťací klapky
11	-	Maximální hodnota proudu	V menu nastavuje velikost proudového omezení H-můstku
12	-	-	Zapne/Vypne antiparalelní diodu na unipolárním výstupu
13	-	-	Nastavuje režim spínání tranzistorů v H-můstku (vpřed, vzad, bipolárně)

Tab. 19: Režimy a funkce implementované v testeru

## 2.4.5 Hlavní smyčka a časování aplikace

Ve funkci `Main()` je nutno ještě před vykonáváním nekonečné smyčky provést inicializaci periférií a proměnných. Jedná se o nastavení portů MCU a vnitřních periférií, inicializaci OLED displeje, zobrazení startovací zprávy na displeji, inicializaci filtrů napětí z potenciometrů, a dalších externích periférií, viz níže. Po vykonání této sekvence již následuje nekonečná smyčka programu



Obr. 59: Vývojový diagram funkce `Main()` testeru PWM s bipolárním výstupem

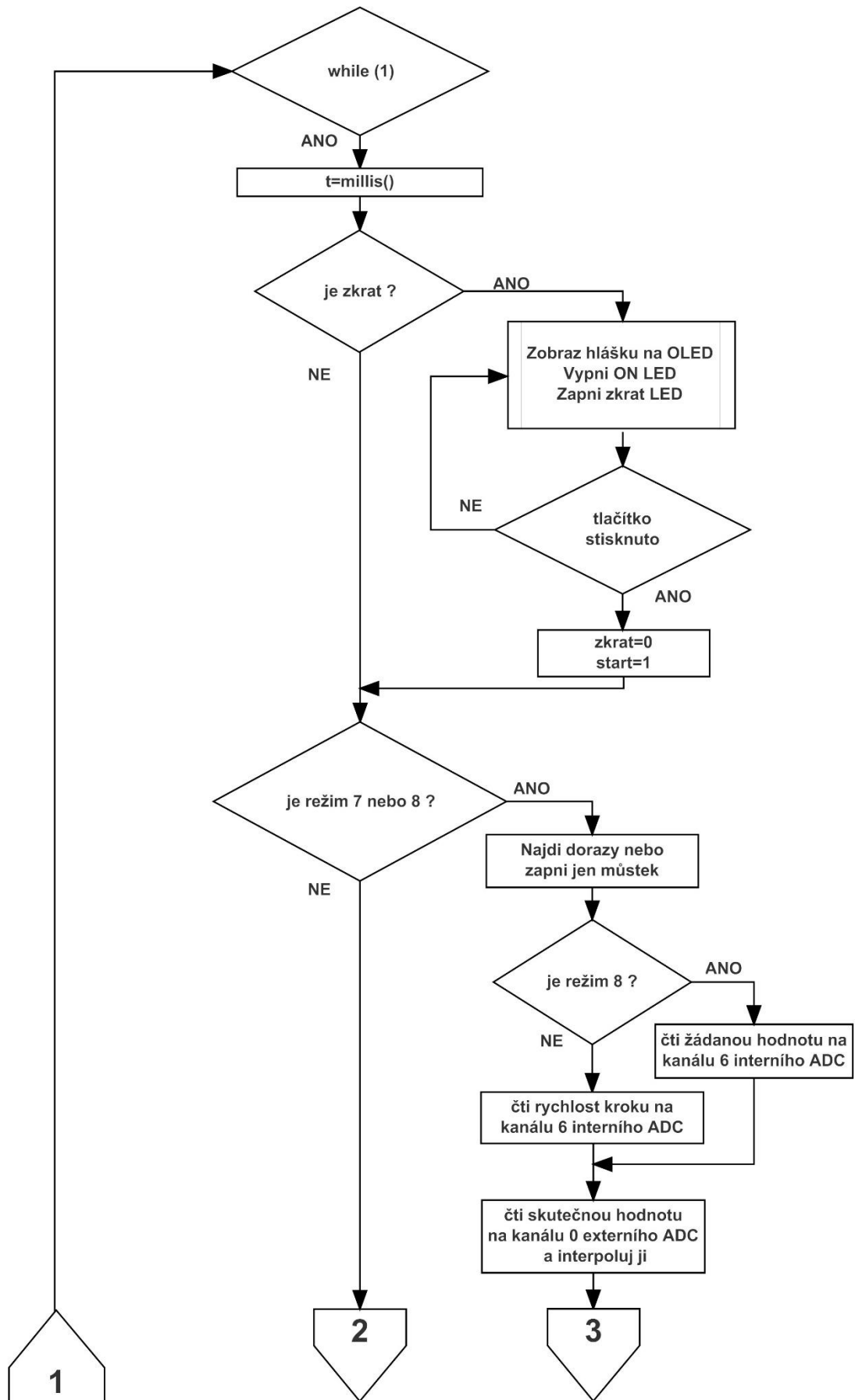
Jak již bylo zmíněno v kapitole 2.4.3.2 Časovač/čítač TC2 a 2.4.3.6 Použitá přerušení, o časování aplikace se stará časovač/čítač TC0. V obsluze přerušení `TIMER0_COMPA_vect` je inkrementována proměnná `t_ms`, která je násobkem 4,992 ms. Při každém průchodu hlavní smyčky je zavolána funkce `uint32_t millis(void)`, která vrací čas od zapnutí, jejíž návratová hodnota je uložena do proměnné `t`. Tato proměnná je použita ve vyhodnocení podmínky, zda již uplynul čas pro opětovné vykonání definované části kódu, viz příklad níže.

```
While (1)
{
    t = millis();
    // další kód

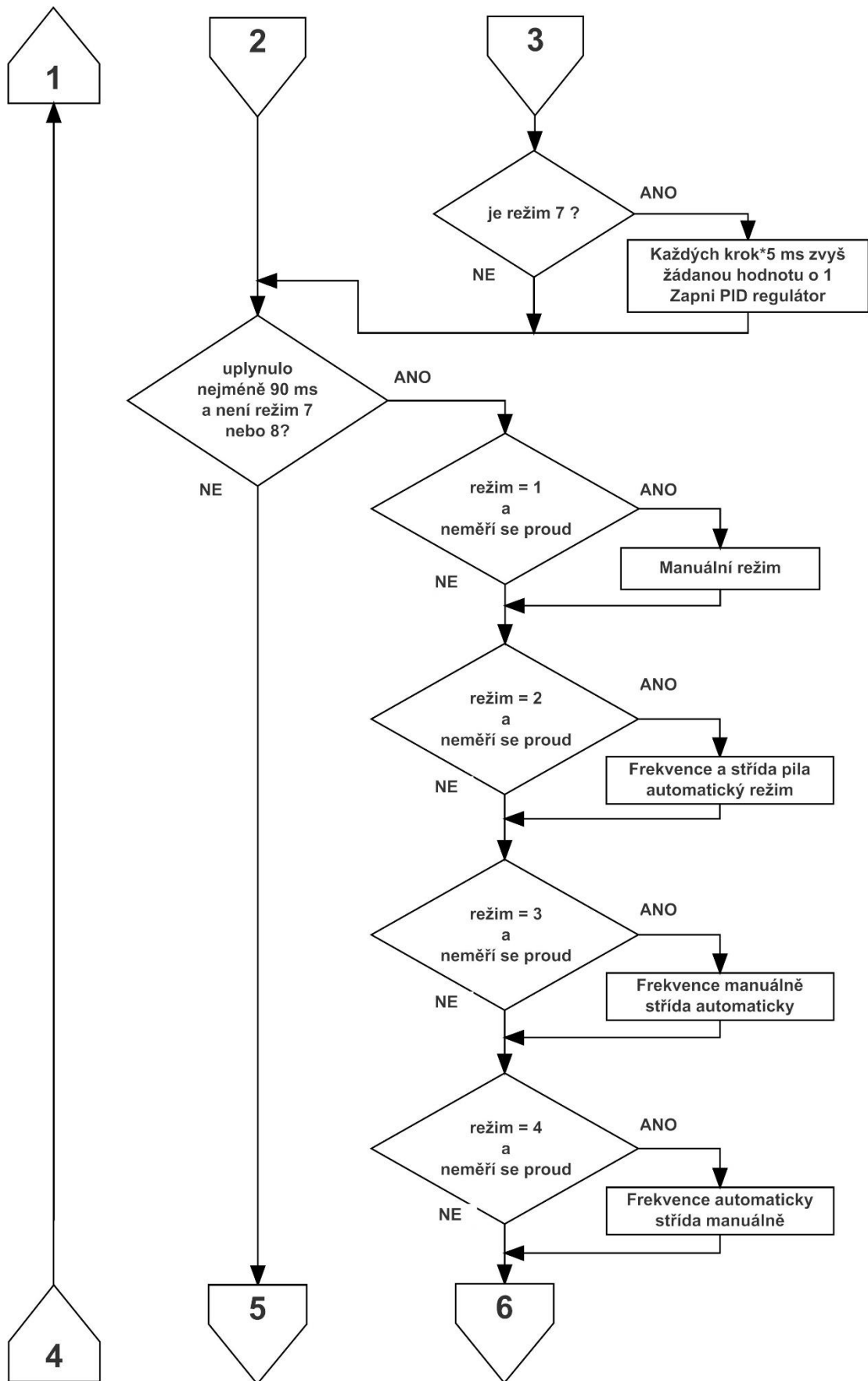
    if (( t>= tTarget4)&&(rezim == 7))
    {
        tTarget4 = t + (uint32_t)T;
        // vykonání časovaného kódu
    }
}
```

Na začátku hlavní smyčky programu dojde k nastavení proměnné `t`, poté se kontroluje hodnota proměnné `zkrat`, která je nastavena na 1 v případě volání funkce přerušení `INT0`. Podrobný popis funkce je v kapitole 2.4.7. Dále následuje v případě režimu 7 a 8, což jsou režimy zpětnovazební regulace polohy pomocí PID regulátoru, nastavení žádané a skutečné hodnoty a nastaví se hodnota proměnné `PID_regulator_run` na 1. To zajistí vykonání kódu regulátoru v přerušení `TIMER0_COMPA`. Dále následují již časované bloky 1, 2 a 3 popsané níže. Tento algoritmus je popsán vývojovým diagramem na následujících stranách.

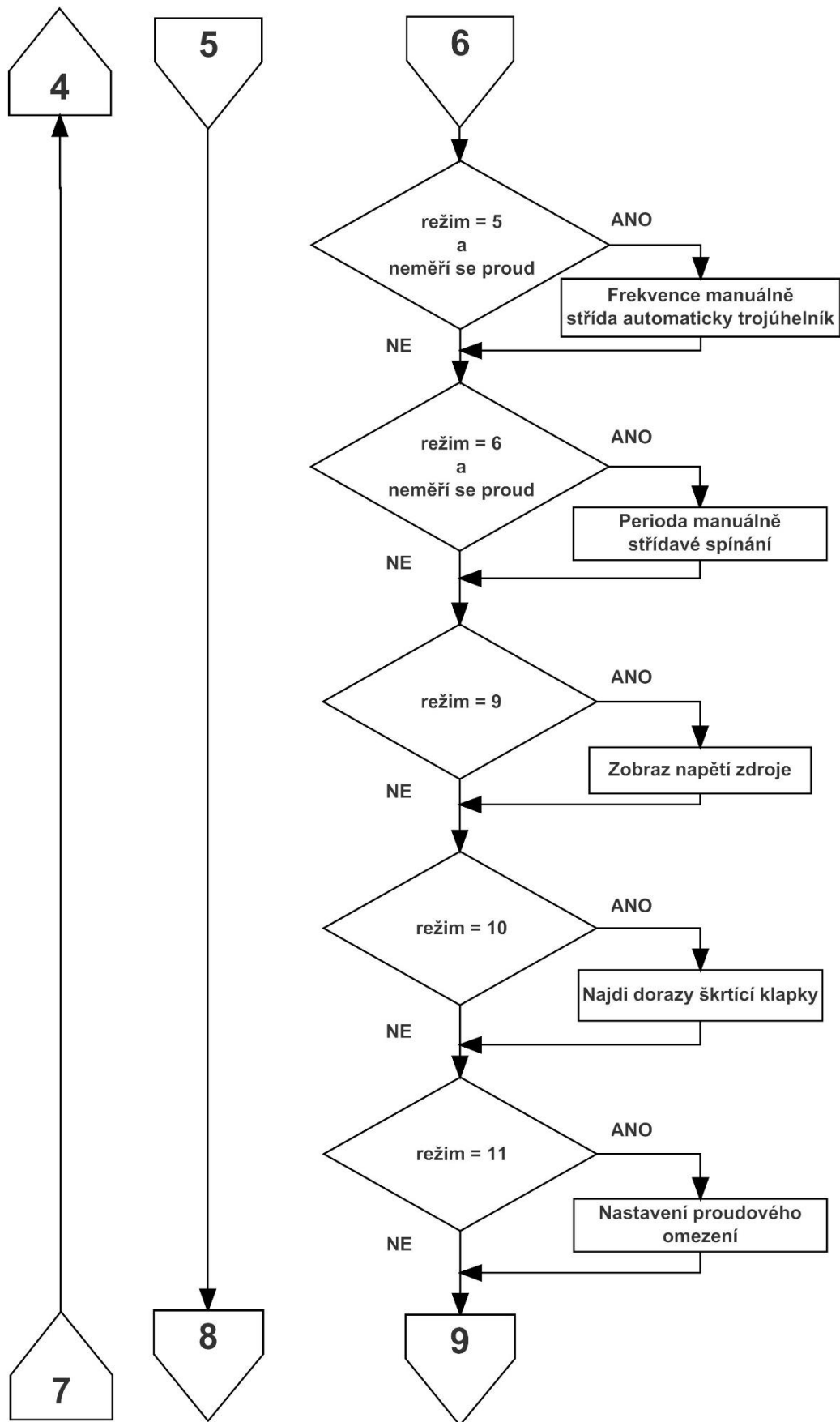
V hlavní smyčce programu jsou časovány tři hlavní bloky. Prvním je Vykonání kódu režimů 1 až 15, což jsou v podstatě položky, které je možné vybrat z menu přístroje. Například automatické režimy, nebo nastavení testeru. V tomto časovaném bloku dochází také k aktualizaci frekvence a střídání PWM, která je realizována funkcemi `void ReadOcr(void)` a `void SetOcr(void)`. A také měření proudu H-můstkem při nízkých frekvencích PWM. Periodu opakování tohoto bloku nastavuje parametr `PWM_UPDATE_PERIOD_MS`. Ten je aktuálně nastaven na  $16 \cdot t\_ms$ , tedy přibližně 90 ms. Druhý blok, jehož periodu nastavuje parametr `BTN_READ_PERIOD_MS`, vykonává čtení stavu tlačítek. To se děje aktuálně s periodou  $2 \cdot t\_ms$ , tzn. přibližně 10 ms. Poslední třetí blok vykonává odesílání dat pro OLED displej a měření napájecího napětí zdroje s periodou  $20 \cdot t\_ms$ , což je přibližně 100 ms. Tuto periodu nastavuje parametr `OLED_REFRESH_TIME`.



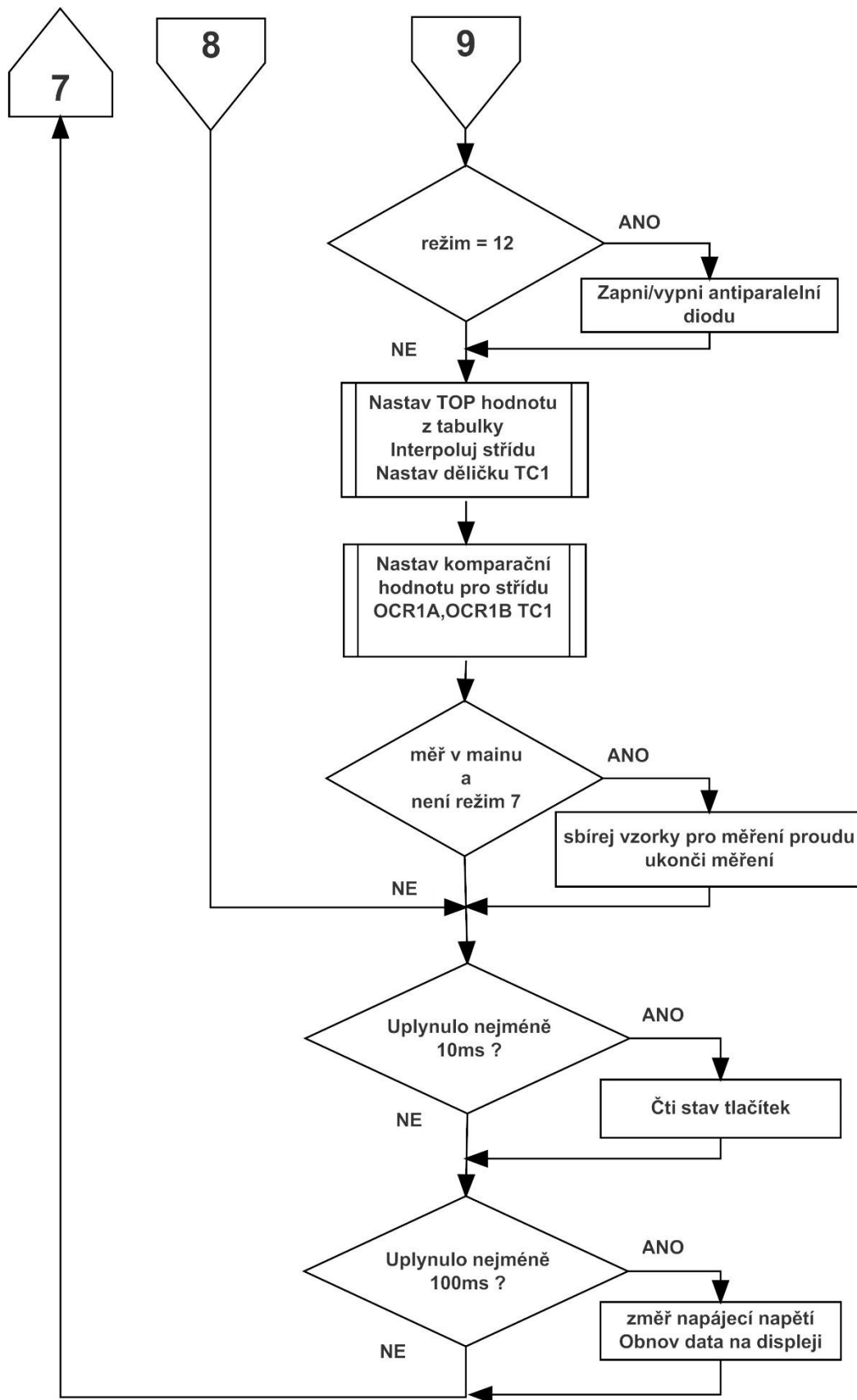
Obr. 60: Zjednodušený vývojový diagram hlavní smyčky programu 1.část



Obr. 61: Zjednodušený vývojový diagram hlavní smyčky programu 2.část



Obr. 62: Zjednodušený vývojový diagram hlavní smyčky programu 3.část



Obr. 63: Zjednodušený vývojový diagram hlavní smyčky programu 4.část



## 2.4.6 Externí A/D převodník

Na vstupy AIN0 a AIN1 externího A/D převodníku jsou připojeny analogové vstupy testeru. AIN0 se používá pro zpětnovazební řízení škrťací klapky pro čtení napětí senzoru polohy v klapce. Vstup AIN1 bude použit pro režim automatického testu senzorů polohy škrťací klapky. Na stupech AIN0 a AIN1 je dělič 1/2, takže se programově násobí 2. Vstup AIN2 je použit pro měření velikosti napájecího napětí zdroje přes dělič 1/3. Vstup AIN3 je rezervní, hardwarově připravený pro doplňkové měření odporu, v současné verzi softwaru není použitý. Adresa pro komunikaci na I<sup>2</sup>C sběrnici je nastavena na 0x48 hexadecimálně.

Kanál	Funkce
AIN0	Čtení senzoru skutečné polohy škrťací klapky pro PSD regulátor
AIN1	Čtení druhého senzoru skutečné polohy škrťací klapky
AIN2	Měření napájecího napětí testeru
AIN3	Doplňkové měření odporu (budoucí funkce)

Tab. 20: Použití analogových vstupů externího A/D převodníku ADS1015

Pro účely testeru je použit režim samostatného převodu (single ended). V tomto režimu je rozlišení pouze 11bitové<sup>6</sup>, takže výstup je v rozsahu 0-2047. V programu je nastaven zisk interního zesilovače na 2/3, to je pro maximální rozsah vstupního napětí 0 až 6,144 V. Odpovídající rozlišení je tedy 3 mV na krok (LSB). Tento A/D převodník má také vlastní interní zdroj referenčního napětí, bez přepočtu tedy nelze porovnávat naměřené hodnoty s interním A/D převodníkem MCU ATmega328. V programu je použita vzorkovací rychlost 1600 sps. Podle datasheetu [23] výrobce A/D převodníku by tedy měla být doba převodu při této vzorkovací frekvenci 0,625 ms. To je zřejmě uvažováno pro režim kontinuálního převodu jednoho ze vstupů. V praxi trvá jeden převod při přepínání vstupů 1,625 ms. To je oproti A/D převodníku v MCU ATmega328 velmi dlouho. Nicméně čtení polohy škrťací klapky je jedná funkce, kde je doba převodu důležitá. Perioda vykonávání regulátoru je přibližně 5 ms, takže by neměl být problém ani s takto dlouhou dobou převodu. Čtení ostatních analogových vstupů nemá takovou prioritu, nedochází tak k častému přepínání vstupů A/D převodníku, které by zdržovalo měření skutečné polohy škrťací klapky v průběhu regulace polohy škrťací klapky. V případě potřeby lze zvýšit vzorkovací rychlost na 3300 sps.

<sup>6</sup> V režimu diferenciálního převodu mezi dvěma kanály je rozlišení 12bitové, výstup je v rozsahu -2048 až 2047

Použitá knihovna s funkcemi pro obsluhu A/D převodníku ADS1015 od Adafruit Industries bohužel neumožňuje kontinuální převod. V programu je nejprve třeba vytvořit instanci *ads* třídy *Adafruit\_ADS1015*, abychom mohli používat metody a členy této třídy. Dále je třeba inicializovat TWI (I<sup>2</sup>C) sběrnici voláním *ads.begin()*. Potom již stačí zavolat funkci *ads.readADC\_SingleEnded(číslo kanálu)* na potřebném místě.

Ukázka kódu obsluhy převodníku a čtení hodnoty z kanálu 0:

```
Adafruit_ADS1015 ads; // Instance ads
ads.begin(); // Inicializace TWI
adc_open = 2*(ads.readADC_SingleEnded(0)); // Čtení hodnoty na vstupu 0 ADC
```

#### 2.4.7 Reakce na nadproud

Jak již bylo nastíněno v kapitole 2.4.3.6 Použitá přerušeni, použitý H-můstek nemá samostatný výstup poskytující informaci o zkratu, nebo vybavení interní tepelné pojistky. Tato informace je poskytována na pinu EN<sub>A</sub>, případně EN<sub>B</sub>, které zároveň slouží pro ovládání H-můstek z MCU. Tento pin je uzemněn v případě, že vnitřní logika příslušného H-můstku vyhodnotí, že je proud větší než typicky 5,6 A. Tento stav je nutné zachytit, to řeší externí přerušeni *INT0* a *INT1*, které je nastaveno tak, aby reagovalo na sestupnou hranu. Protože sestupná hrana se objeví i při chtěném vypínání H-můstku, program před vypnutím deaktivuje v registru *EIMSK* reakci na externí přerušeni *INT0* případně *INT1*. Po zapnutí H-můstku přivedením log.1 na vstup EN<sub>A</sub> nebo EN<sub>B</sub> se vynulují žádosti o přerušeni v registru *EIFR* a po krátké prodlevě 100 μs je přerušeni v registru *EIMSK* opět povoleno. Tím je zajištěno, že nebude docházet k falešné reakci na změnu vstupu EN<sub>A</sub> a EN<sub>B</sub>. V samotné obsluze přerušeni se ověří, zda je můstek zapnutý, tzn. pin PC0 je v log. úrovni H a v případě že je podmínka splněna, vypne se H-můstek a proměnná *zkrat* se nastaví na 1. Tato globální proměnná se vyhodnocuje v hlavní smyčce programu, viz Obr. 60. Zde je vypnuta LED signalizující aktivní výstup a zapnuta LED signalizující zkrat. Dále je zobrazena hláška na displeji informující obsluhu, že byl detekován zkrat na výstupu H-můstku. Pokud obsluha zmáčkne prostřední tlačítko BTN1, H-můstek se opět aktivuje a displej začne zobrazovat provozní informace v případě, že byl zkrat odstraněn. Reakce MCU na zkrat není z pohledu času kritická. I v případě, že MCU má na výstupním pinu PC0 log. úroveň H, při aktivaci nadproudové ochrany v H-můstku protéká proud přes odpor R<sub>EN</sub> a otevřený tranzistor na vstupu H-můstku do země.

Tím je udržován H-můstek vypnutý bez ohledu na log. úroveň na vstupech EN<sub>A</sub>, resp. EN<sub>B</sub>.  
Detaily jsou uvedeny v kapitole 2.2.5.3 Nadproudová ochrana H-můstku.

Makra pro zapnutí a vypnutí externího přerušení INT0:

```
#define ENABLE_INT0 (EIMSK |= (1<<INT0)) //Zapne přerušení INT0
#define DISABLE_INT0 (EIMSK &= ~(1<<INT0)) //Vypne přerušení INT0
```

Obsah obsluhy přerušení INT0\_vect:

```
if ((PORTC&(1<<PORTC0))) //Vybaví zkrat pouze když je H-můstek zapnutý
{
    H_BRIDGE_DISABLE; //vypne PC0 výstup EN H-můstek
    zkrat = 1; // proměnná se vyhodnotí v hlavní smyčce
}
```

Sekvence zapínání H-můstku:

```
H_BRIDGE_ENABLE; //Makro nastaví log. úroveň H na pinu PC0
EIFR = (1<<INTF0); //Vynuluje žádosti o přerušení z INT0
_delay_us(100); // Čeká 100 μs
ENABLE_INT0 //Makro zapne reakci na přerušení INT0
```

Sekvence vypínání H-můstku:

```
DISABLE_INT0; //Makro vypne reakci na přerušení INT0

H_BRIDGE_DISABLE; //Makro nastaví log. úroveň L na pinu PC0
EIFR = (1<<INTF0); //Vynuluje žádosti o přerušení z INT0
```

## 2.4.8 Generování PWM

Pro generování frekvence PWM je použitý 16bitový časovač/čítač TC1, jak již bylo zmíněno v kapitole 2.4.3.3. Ten pracuje v režimu 8 Phase and Frequency Correct PWM. Na Obr. 57 je znázorněn průběh, kde je z nuly inkrementována hodnota v registru TCNT a po dosažení TOP hodnoty stanovené hodnotou v registru ICR opět dekrementována až na nulu, resp. bottom. Tento režim se jmenuje dual-slope<sup>7</sup>. Z tohoto mechanismu vychází dále popisované principy pro aktualizaci komparační a TOP hodnoty, bez kterých nelze za chodu měnit střidu a frekvenci. Jak je také patrné z Obr. 57, jsou výstupy OC1A a OC1B z čítače/časovače vůči sobě inverzní. Použití dvou výstupů PWM umožňuje změnu směru otáčení nebo pohybu akčního členu. V testeru jsou implementovány tři režimy řízení můstku, popsané v následujících kapitolách.

### 2.4.8.1 Locked anti-phase režim

V prvním, takzvaném locked anti-phase režimu řízení H-můstku, je na oba dva vstupy H-můstku IN1A a IN2A přiváděn vzájemně inverzní signál PWM. Pro tento režim řeší nastavení časovače/čítače TC1 funkce `void SetupTimerBipolarPWM(void)`. Tato funkce je volána ve funkci `Main()`, režim locked anti-phase je tedy výchozím režimem po zapnutí testeru.

Kód funkce `SetupTimerBipolarPWM()` nastavující časovač/čítač TC1

```
TimerMode = 1;
OCR1A = OCR1B = 512; //počáteční střída
ICR1 = top = 1000; //počáteční frekvence
TCCR1A = 0;
TCCR1C = 0;
TIMSK1 = 0; // bez přerušení

//set OC1A/1B on compare match when up counting,clear when down
counting
TCCR1A |= (1<<COM1A1)|(1<<COM1B1)|(1<<COM1B0);
//spouští timer s 16MHz / CS11= DIV 8, mod 8 - Phase and Frequency
correct, TOP = ICR1
TCCR1B |= (1<<WGM13)|(1<<CS11);
```

Proměnná `TimerMode` slouží pro identifikaci aktuálního režimu na displeji. Pro tento režim se zobrazí na prvním řádku displeje zkratka “BI”.

---

<sup>7</sup> Z angličtiny dvojitý sklon

### 2.4.8.2 Sign-magnitude režim

Tento způsob řízení H-můstku používají dva pracovní režimy testeru. U těchto dvou režimů se liší směr otáčení motoru (resp. polarita napětí na výstupu H-můstku) podle toho, jaké dva tranzistory v diagonále H-můstku jsou právě sepnuté. Buď je vstup IN1A stále zapnutý v log. úrovni H a na IN2A je přiveden signál PWM nebo naopak IN1A je stále zapnutý v log. úrovni H a na IN2A je přiveden signál PWM, podle požadované polaroty napětí na výstupu H-můstku. To odpovídá řádkům 3 a 4 v Tab. 10. H-můstek je v zapnutém stavu udržován log. úrovní H na vstupu EN<sub>A</sub>, resp. EN<sub>B</sub>. Ke spínání tranzistorů dochází tak pouze v jedné polovině H-můstku, druhá polovina je stále zapnutá. Pro tento režim se zobrazí na prvním řádku displeje zkratka “FW”, nebo “RE” dle směru otáčení.

Níže je kód funkce `void SetupTimerForwardPWM(void)` pro nastavení otáčení vpřed dle Tab. 10. Pro opačný směr je nastavení analogické, a nastavení realizuje funkce `void SetupTimerReversePWM(void)`.

```
TimerMode = 2;
ICR1 = top = 1000;           // počáteční frekvence

TCCR1A = 0;
//Clear OC1B on compare match when up-counting. Set OC1B on compare match
when down counting, IN1A = PWM
TCCR1A |= (1<<COM1B1)|(1<<COM1B0);
// spouští timer s 16MHz / CS11= DIV 8, mod 8 - Phase and frequency
correct, TOP = ICR1
TCCR1B |= (1<<WGM13)|(1<<CS11);
PORTB |= (1<<PORTB1); //Nastav IN2A=PB1 do log "1" OC1A
```

### 2.4.8.3 Softwarové řešení

V kapitole 2.4.4 je popsáno ovládání testeru a princip čtení hodnot obou potenciometrů pro zadávání střídy a frekvence PWM. To probíhá v hlavní smyčce s periodou natavenou parametrem `PWM_UPDATE_PERIOD_MS`. Ten je aktuálně nastaven na  $16 \cdot t_{ms}$ , tedy přibližně 90 ms. Střída se zadává potenciometrem vpravo (vstup interního A/D převodníku ADC6), frekvence potenciometrem vlevo (vstup interního A/D převodníku ADC7). Hodnoty jsou čteny v 10bitovém rozlišení. Následně jsou vyfiltrovány filtrem s klouzavým průměrem SMA se šířkou průměrovacího okna 3 vzorky.

```
SetupADC(64,0); // Nastav 10bit rozlišení a děličku 64
strida_adc = Read_adc(POT_DUTY); //Čtení hodnot z POT střídy
frekvence_adc = Read_adc(POT_FREQ); //Čtení hodnot z POT frekvence

avg1 = sma(strida_adc,&filterSMA1,buffer1,K1); // filtrace 1. signálu
avg2 = sma(frekvence_adc,&filterSMA2,buffer2,K2); // filtrace 2. signálu
```

Následně je zavolána funkce `ReadOcr()`. Ta přepíná děličku pro časovač/čítač TC1 mezi 8 a 64 na rozhraní druhé a třetí dekády, to je frekvence 100 Hz. Tomu odpovídá hodnota z A/D převodníku 349. Nad touto hodnotou se používá dělička pro čítač/časovač TC1 64, pod touto hodnotou potom 8. Celkový rozsah frekvence je rozdělen logaritmičticky do tří dekád 10-100 Hz, 100-1000 Hz a 1000-10000 Hz. Poloha obou potenciometrů je přečtena v 10bitovém rozlišení A/D převodníkem a výstup je v rozsahu hodnot [0-1023]. Pro převod rozsahu [0-1023] na hodnotu TOP, která určuje frekvenci PWM, slouží tabulka obsahující 1024 TOP hodnot čítače/časovače TC1, která je uložena v paměti FLASH.

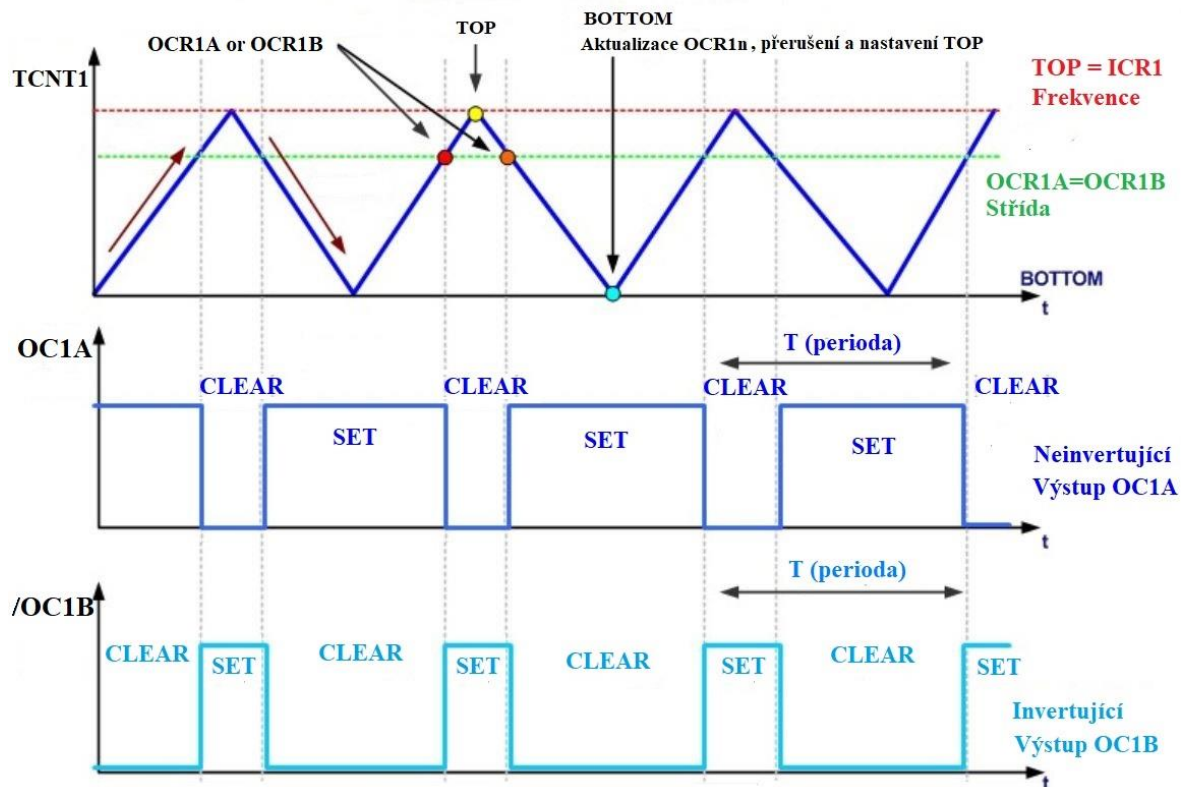
```
FrequencyTable[] PROGMEM = {12500,12416,12332,12249,...,100};
```

```
if (frekvence_adc < 341)
{
    TCCR1B |= (1<<CS10); //Nastav děličku 64
}
else
{
    TCCR1B &= ~(1<<CS10); //Nastav děličku 8
}
//[0-1023]=frekvence_adc z potu, index podle ADCW, vrací TOP z tabulky TOP
hodnot
top = pgm_read_word(&(FrequencyTable[frekvence_adc]));
// interpolace [0-1023] z hodnoty ADC-střída do hodnoty [0-top]
ocr = Map(strida_adc,0,1023,0,top);
if (ocr < 0) ocr = 0;
if (ocr > top) ocr = top;
```

Následně je volána funkce `SetOcr()`, jejíž obsah je níže. V ní je zakázáno přerušení pro časovač/čítač TC0 generující časovací pulzy pro aplikaci o délce 4,992 ms do doby, než se nastaví nová TOP hodnota pro časovač/čítač TC1 v přerušení `TIMER1_OVF_vect`. Podmínka `TCNT1 > 10` slouží pro zmenšení rizika, že se jeden z komparačních registru nestihne aktualizovat, pokud se kód pro aktualizaci komparačních registrů OCR1A a OCR1B vykoná v blízkosti konce periody PWM. Je důležité také smazat žádost o přerušení, jinak by se vyvolalo přerušení ihned. TOP hodnota by se tak nastavila před koncem periody, a nikoliv na začátku další periody tak, jak je to potřeba. Pokud by žádost o přerušení nesmazala, byla by v danou chvíli TOP hodnota nižší, než aktuální hodnota v registru TCNT1 a aktuální perioda PWM by nebyla ukončena. Časovač/čítač TC1 by tak tikal až do přetečení do hodnoty 65535. V generovaném průběhu PWM by tak vznikla chyba v podobě mezery dlouhé 32,768 ms. Detaily natavení registrů ICR1 a OCR1A, OCR1B spolu s průběhem výstupní PWM jsou na Obr. 64

```
TIMSK0 &= ~(1<<OCIE0A); //zakázání přerušení pro TC0

if (TCNT1 > 10)
{
    OCR1A = OCR1B = ocr; //nastaví komparační hodnotu pro střídu
    TIFR1 = 0xFF; //smazání zadosti o přerušení
    TIMSK1 |= (1<<TOIE1); //povolení přerušení při přetečení OVF_vect
TC1
}
else
{
    // Povol přerušení zakázaná před změnou TOP a OCR pro TC0
    TIMSK0 |= (1<<OCIE0A);
}
}
```



Obr. 64: Časový diagram nastavení registrů ICR1 a OCR1n, průběh výstupního signálu PWM [28]

Při přetečení čítače/časovače TC1 (bottom) je vlajkou *TOV1* spuštěno přerušení přetečení, viz Obr. 57. Obsluha přerušení je ve funkci *TIMER1\_OVF\_vect*. V ní se aktualizuje registr *ICR1* časovače/čítače TC1. Tím se nastaví nová *TOP* hodnota, a tedy frekvence PWM. Přerušení se samo zakáže a dále povolí přerušení pro časovač/čítač TC0 (aby mohl opět inkrementovat proměnnou *t\_ms* pro časování aplikace), které bylo zakázáno ve funkci *SetOcr()* v hlavní smyčce. Tím je dokončen jeden cyklus nastavení frekvence a střídání PWM. Při automatických režimech jsou nahrazeny hodnoty přečtené z potenciometrů vypočtenými hodnotami v dané části programu, nastavení *TOP* a *OCR* je ale jinak shodné.

```
ICR1 = top; //Nastaví novou TOP hodnotu
TIMSK1 &= ~(1<<TOIE1); //Zakáže přerušení TIMER_OVF_vect
TIMSK0 |= (1<<OCIE0A); //Povolí přerušení zakázaná před změnou TOP a OCR pro TC0
```



## 2.4.9 Obsluha displeje

Pro obsluhu alfanumerického OLED displeje jsem napsal knihovnu skládající se ze souborů OLEDM.c a OLEDM.h. V datasheetu [16] publikovaném výrobcem je uvedena základní tabulka příkazů pro obsluhu displeje. Kompletní seznam příkazů je pak obsažen v datasheetu [17] integrovaného řadiče SSD1311. Obsluha sběrnice I<sup>2</sup>C je zajištěna knihovnou I<sup>2</sup>C.c, ve které jsou základní funkce pracující přímo s TWI registry v MCU. Jedná se o začátek a konec komunikace, zápis a čtení dat. V hlavičkovém souboru OLEDM.h jsou nadefinována makra pro nastavení konfiguračních registrů displeje v hexadecimálním formátu. Jedná se o doporučená nastavení od výrobce a některá uživatelská nastavení jako orientace displeje, počet řádků, aktivace kurzoru nebo úroveň kontrastu a také adresa displeje na I<sup>2</sup>C sběrnici, což je 0x3C.

Před hlavní smyčkou programu se ve funkci `main()` zavolá funkce `void OLED_setup(void)`, která provede nastavení komunikační rychlosti I<sup>2</sup>C sběrnice 400 kbps a nastaví veškeré parametry, které jsou nutné před odesláním dat do displeje. Před odesláním dat do displeje je nutné zavolat funkci `void OLED_setCursor(uint8_t col, uint8_t row)`, kde `row` je číslo řádku [0-3] a `col` pozice v řádku [0-19]. Po zavolání funkce nastavující začátek zobrazované informace následuje odeslání samotných dat. K tomu je majoritně v programu použita funkce `void OLED_sendString(const char *String)`. Vstupní proměnná je řetězec ASCII znaků. Odesílání dat po sběrnici I<sup>2</sup>C probíhá cyklicky po bytech za pomoci funkce `void oled204m_data_write(uint8_t data)`. Pro potřebu vypisovat speciální znaky byla vytvořena funkce `void OLED_sendChar(uint8_t character)`, kde vstupní proměnná představuje adresu znaku v CGROM (Character Generator ROM). Vymazání displeje zajišťuje funkce `void OLED_clearOLED(void)`, která na displej v podstatě vypíše znak 0x20 – mezeru do všech 80 znaků.

O zobrazení aktuálních dat na displeji se stará funkce `void DisplayValues()`, která se volá v hlavní smyčce programu s periodou přibližně 100 ms nastavenou parametrem `OLED_REFRESH_TIME`. Viz kapitola 2.4.5 Hlavní smyčka a časování aplikace a vývojový diagram na Obr. 63.

Ukázka kódu:

```
OLED_setup();
OLED_setCursor(0,0); OLED_sendString("TESTER AKCNICH CLENU"); //1.raw,1.col
OLED_setCursor(4,1); OLED_sendString("VERZE SW "); //2.raw,5.col
itoa(SW_VERSION,buffer,10);
OLED_setCursor(10,1); OLED_sendString(buffer); //2.raw,11.col
```

### 2.4.10 Měření proudu

Pro měření proudu zátěží je potřeba měřit úbytek napětí na bočniku H-můstku. Tento bočník má velikost  $0,195 \Omega$ , viz kapitola 2.2.5. Rozsah napětí na tomto bočniku je v řádu  $0,0088 \text{ V}$  až  $0,877 \text{ V}$  (22). To odpovídá proudu  $45 \text{ mA}$  až  $4,5 \text{ A}$ . Napětí na bočniku H-můstku je měřeno interním A/D převodníkem na pinu ADC2. Zpočátku byla myšlenka měřit střední hodnotu proudu, v průběhu návrhu a po konzultaci se zadavatelem projektu bylo rozhodnuto spíše měřit momentálně maximální hodnotu proudu, protože není zatím ověřeno, jakou hodnotu měří vlastně sériová diagnostika. Proto byla v programu zachována možnost do budoucna střední hodnotu proudu měřit. Tato funkce je spíše informativní. Většina postupů pro diagnostiku akčních členů uvádí spíše rozmezí hodnoty odporu vinutí akčního členu než proudu. Byla zvolena metoda odebrání maximálního počtu vzorků v průběhu jedné periody. Pro střední hodnotu proudu by pak bylo možné všechny vzorky sečíst a vydělit počtem odebraných vzorků. Pro maximální hodnotu proudu postačí výpočet největšího vzorku z odebraných. Počátek a konec vzorkování je potřeba synchronizovat s generováním PWM, aby byl proud měřen ve správné části periody, kdy zátěží přes snímací rezistor teče proud. Pro tento účel je využito přerušení při přetečení `TIMER1_OVF_vect` čítače/časovače TC1, které nastává na začátku každé periody PWM. Při vyšších frekvencích PWM je nutno vzorkovat velmi rychle. Převodník je tedy nastaven na 8bitové rozlišení a frekvenci  $4 \text{ MHz}$ . Při frekvenci PWM  $10 \text{ kHz}$  je možno odebrat asi 8 vzorků v jedné periodě PWM. Malý počet vzorků by při malém zvlnění proudu při vysoké frekvenci PWM neměl být problém. S klesající frekvencí zvlnění proudu stoupá, to je ale vykompenzováno větším počtem vzorků na periodu PWM. Hlavním cílem je, aby odběr všech vzorků proudu během celé periody PWM, realizovaný v obsluze přerušení, netrval moc dlouho, aby přerušení zbytečně neblokovalo vykonávání jiných částí kódu v MCU. Proto byla zvolena hranice frekvence PWM  $5 \text{ kHz}$ . Pro vysoké frekvence PWM (nad  $5 \text{ kHz}$ ) se všechny vzorky proudu za celou periodu PWM změní v obsluze přerušení, které se vyvolá na začátku každé periody PWM. Pro frekvence PWM pod  $5 \text{ kHz}$  se vzorky proudu odebírají v nekonečné smyčce v `main()`.

Výpočet střední hodnoty proudu:

```
if (top < 200)    // od frekvence PWM 5 kHz
{
    // odebírá vzorky a čeká na skončení periody PWM
    while (!(TIFR1 & (1<<TOV1)))
    {
        sum += Read_adc(SENSE_A);
        proud_pocet_vzorku++;
        if (proud_pocet_vzorku >= 255) break;    // pro jistotu
    }
    proud_adc_sum = sum;
    mereni_proudu_spusteno = 0;                // konec měření
}
else if (frekvence_adc > 340)    //100 Hz
{
    mer_proud_v_mainu = 1; // vzorky se odeberou v nekonečné smyčce v
    main()
}
}
```

Pro nižší frekvence PWM by bylo zbytečně dlouho blokováno vykonávání celého programu, proto se nastaví proměnná *mer\_proud\_v\_mainu* na 1 a odebrání vzorků proběhne v hlavní smyčce v *main()*.

```
if ((mer_proud_v_mainu)&&(rezim !=7))
{
    // odebírá vzorky a čeká na skončení periody PWM
    while (!(TIFR1 & (1<<TOV1)))
    {
        proud_adc_sum += Read_adc(SENSE_A); //vzorkuje na vstupu ADC6
    }
    proud_pocet_vzorku++;
    if (proud_pocet_vzorku >= MAX_POCET_VZORKU_PROUDU) break;
}
mereni_proudu_spusteno = 0;    // konec měření
mer_proud_v_mainu = 0;
}
```

Maximální hodnota vzorku napětí na bočnicku je použita na výpočet odporu připojené zátěže. Ta je měřena přímo ve funkci *Read\_adc()* a uložena do proměnné *max\_sample*.

```
if (channel == 2)
{
    if (ADCH > max_sample )
    {
        max_sample = ADCH;
    }
}
}
```

Proměnná `max_sample` je využita následně při výpočtu maximálního napětí na bočníku  $R_{SENSE}$  a následně max. proudu zátěží. Z těchto veličin spolu s odporem  $R_{DSON}$  spínacích tranzistorů H-můstku je vypočten odpor zátěže, viz následující kapitola. Tato funkcionality bude dále konzultována se zadavatelem, pravděpodobně dojde k úpravě poskytovaných informací obsluze testeru.

```
max_napeti = (uint16_t)((ADC_VREF*(uint32_t)max_sample)>>8); // výpočet
max. napětí
max_proud = max_napeti/0.195; // 2xparalelně 0,39 Ω
```

### 2.4.11 Měření odporu

Protože všechny dílenské příručky uvádějí pro ověření funkčnosti akčního členu měření odporu vinutí, byla přidána i tato funkce do testeru. Hodnoty odporu vinutí akčních členů se pohybují zpravidla v jednotkách ohmů, u krokových motorů to mohou být desítky ohmů, viz kapitola 1.1 Akční členy používané ve vozidlech. Přesnost měření odporu závisí na více faktorech. Cílem není měřit co nejpřesněji odpor vinutí, pro tyto účely existují lepší metody. Přesnost měření odporu vinutí závisí na přesnosti měření proudu. V sérii s akčním členem (jeho vinutím) je i H-můstek a bočník. Je nutno tedy uvažovat i s odporem  $R_{SENSE}$  a  $R_{DSON}$ . Tyto hodnoty ovšem nejsou konstanty, jak by se mohlo zdát, ale závisí na teplotě, což ovlivňuje přesnost měření zejména malých odporů (odporů o srovnatelné nebo menší velikosti, než je  $R_{SENSE} + R_{DSON}$ ). Respektive pak bude přesnost měření hodně záviset i na přesnosti stanovení  $R_{SENSE}$  a  $R_{DSON}$ . Odpor vinutí  $R$  [ $\Omega$ ] pak spočítáme podle vzorce (54):

$$R = \frac{V_{CC}}{I} - R_{DSON} - R_{SENSE} \quad [\Omega] \quad (54)$$

Kde  $V_{CC}$  je velikost napájecího napětí [V],  $I$  proud (max. hodnota) [A] vypočtený z úbytku napětí na bočníku  $R_{SENSE}$  [ $\Omega$ ] a  $R_{DSON}$  je odpor tranzistorů H-můstku v sepnutém stavu [ $\Omega$ ]. Hodnota odporu je prozatím stále zobrazována na OLED displeji spolu s dalšími údaji (kromě režimů zpětnovazební regulace škrtkové klapky). Vzhledem k omezeným rozměrům displeje bude tato funkce nejspíše přesunuta do některého z automatických režimů. Hodnotu odporu není třeba zobrazovat neustále a zabírat tak místo na displeji.

Pro doplňkové měření odporu akčních členů je připraven na DPS proudový zdroj s IO LM334. Pro měření odporu jsou vyčleněny samostatné svorky, měření bude provádět externí A/D převodník na vstupu AIN3, viz kapitola 2.2.3 Externí A/D převodník.

### 2.4.12 Regulace polohy škrťící klapky

Pro testování akčních členů se snímačem polohy (zejména škrťících klapkek) byl implementován režim zpětnovazební regulace polohy. Teoreticky je možné řídit i jiný akční člen než škrťící klapku, to v rámci práce však nebylo otestováno. Pro tuto funkci byl implementován číslicový PID (neboli PSD) regulátor, jehož funkce a struktura proměnných jsou definovány v hlavičkovém souboru *PSD\_polohovy.h*. Pro správnou funkci číslicového PID regulátoru je nutno nastavit všechny jeho konstanty. To provedeme pomocí maker:

```
#define T 5U // perioda vzorkování (ms)
#define Ti 100 // integrační časová konstanta (ms)
#define Td 1*10 // derivační časová konstanta (ms)
#define N 10U // koeficient filtru derivační složky.
#define Kp 3 // proporcionální zesílení (závislé na rozměru akční a
reg. veličiny)
#define OUT_MIN 0 // dolní mez akční veličiny regulátoru
#define OUT_MAX 1023 // horní mez akční veličiny regulátoru
#define SETPOINT_FILTER 1U // filtr žádané hodnoty, 1 - zapnuto, 0 -
vypnuto
```

Následně je nutno regulátor inicializovat. To je provedeno ve funkci *Main()*, kde proběhne i reset regulátoru:

```
PSD_param_t psd1;
//Inicializace PSD regulátoru
PSD_init(&psd1,Kp,Ti,Td,N,(float)T,SETPOINT_FILTER,OUT_MIN,OUT_MAX)
//Reset regulátoru, integrační složka, akční veličina => požadovaná
hodnota.
PSD_reset(&psd1,0,0);
```

Regulátor je nutno vykonávat s přesnou periodou. Proto bylo potřeba vložit kód vykonávání regulátoru do funkce obsluhy přerušení `TIMER0_COMPA_vect` čítače/časovače TC0. Přerušení nastává každých 4,992 ms. Vzniká tedy chyba -0,008 ms oproti nastavené periodě vzorkování PSD regulátoru  $T = 5$  ms, která by neměla být nijak závažná. Aby nedocházelo k vykonávání kódu PSD regulátoru zbytečně, když není režim zpětnovazební regulace aktivován, je použita proměnná `PID_regulator_run`. Ta se nastavuje na 1 pouze v případě, že je v menu vybrán režim 7 nebo 8, viz Tab. 19, tedy manuální nebo automatické zpětnovazební řízení škrťací klapky.

Kód funkce obsluhy `TIMER0_COMPA_vect` přerušení časovače/čítače TC0:

```
t_ms++;
if (PID_regulator_run == 1)
{
    // spuštění regulátoru, výstup je střída PWM v rozsahu 0 až 1023,
    // vykonání funkce trvá cca 108 us na AVR fcpu = 16 MHz
    strida_adc = PSD_calcul(&psd1,skutecna1,zadana);
    ReadOcr(); //Interpolace střídy [0-1023] do [0-TOP] z tabulky => ocr

    if (TCNT1 > 10)
    {
        OCR1A = OCR1B = ocr;           //nastav komparační hodnotu pro
        střídu
        TIFR1 = 0xFF;                 //smazání žádosti o přerušení
        TMSK1 |= (1<<TOIE1);         //povolení přerušení při přetečení
        OVF_vect čítač/časovač 1
    }
}
```

Funkce `int32_t PSD_calcul(PSD_param_t *Parametry, uint16_t Aktualni, uint16_t Zadana)` přebírá jako vstupní parametry strukturu s nastavením regulátoru, aktuální a žádanou hodnotu regulované veličiny [0-1023]. Návrátová hodnota je komparační hodnota do PWM časovače [0-1023].

Čtení žádané hodnoty polohy v manuálním režimu probíhá v hlavní smyčce programu, viz Obr. 60. Do proměnné `zadana` je uložena hodnota [0-1023] z interního A/D převodníku, který měří napětí na potenciometru vpravo připojeném na pin ADC6. Skutečná hodnota ze senzoru škrťací klapky je čtena externím A/D převodníkem na kanálu AIN0 v 11bitovém rozlišení a vynásobena dvěma kvůli děličí 1/2 na vstupu externího A/D převodníku. Skutečná hodnota je dále přemapována do stejného rozsahu žádané hodnoty vstupující do PSD regulátoru [0-1023] pomocí funkce `Map` a jako hodnota proměnné `skutecna1` použita jako vstupní parametr funkce `PSD_calcul`.

Pro vstupní parametry funkce *Map* je nutno znát hodnoty horního a dolního dorazu klapky. Ty jsou zjištěny při spuštění režimu 7 nebo 8 zavoláním funkce `void NajdiDorazy(void)`. Tato funkce naplno otevře a poté úplně zavře škrtící klapku. Při každém z těchto kroků se uloží naměřená hodnota polohy klapky do proměnné *adc\_open* a *adc\_closed*. Při vypnutém H-můstku je následně změřena i hodnota limp home polohy. Ta je použita následně k výpočtu offsetu pro číslicový PID regulátor. Integrační složka nezačíná od nuly, ale už má naintegrované. To je dáno použitým režimem spínání locked anti-phase H-můstku. Nulovému napětí na kotvě motoru odpovídá střída PWM 50 % Regulátor je následně nastaven funkcí *PSD\_reset*.

Ukázka čtení žádané a skutečné hodnoty, přemapování skutečné hodnoty do rozsahu [0-1023]:

```
#define SKUTECNA_MIN 0
#define SKUTECNA_MAX 1023

zadana = Read_adc(POT_DUTY);
// Kanal 0, 3mV/bit, +- 6.144V, dělič 1/2
skutecna_adc = 2*ads.readADC_SingleEnded(0);
// Interpolace do rozsahu [0-1023]
skutecna1=Map(skutecna_adc,adc_closed,adc_open,SKUTECNA_MIN,SKUTECNA_MAX)
```

V automatickém režimu není potenciometrem vpravo zadávána žádaná hodnota ale rychlost změny žádané hodnoty. Rozsah potenciometru [0-1023] je přepočítán na kroky [1-102], kde každý krok je násobkem základní jednotky 4,992 ms. Například pro maximální hodnotu 102 se mění žádaná hodnota jednou za 510 ms. Změna žádané hodnoty je o 1, to je přibližně 0,098 % z celého rozsahu úhlu otevření škrtící klapky. Průběh žádané hodnoty je trojúhelník, klapka se otevírá ze zavřeného stavu do otevřeného a zpět, po vykonání jednoho cyklu zůstane na v zavřené poloze přibližně dvě sekundy a poté se cyklus opakuje znovu. Tento automatický režim bude ještě pravděpodobně zrevidován a přizpůsoben individuálním potřebám. Nabízí se také možnost parametrizace prostřednictvím menu. Jednalo by se zřejmě o základní vstupy jako je rychlost a čas, resp. násobky základního časového intervalu.

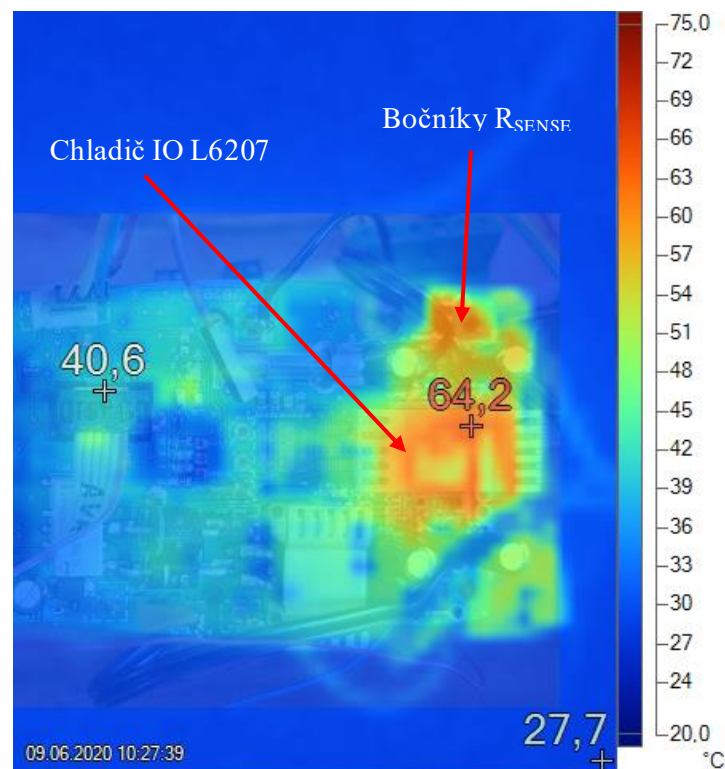
## 2.5 Oživení a naměřené hodnoty

### 2.5.1 Měření teploty

Pro ověření účinnosti chlazení byl tester (výstup H-můstku A) připojen k umělé RL zátěži, napájecí napětí nastaveno na 12 V a střída na 100 %. Při prvním měření byl proud nastaven na 1,43 A. Teplota byla průběžně měřena termo kamerou až do ustálené hodnoty. Pro zjištění spínacích ztrát byla střída pokusně snížena na 99 %. Ke zvýšení teploty nedošlo, naopak teplota mírně klesla. To znamená, že spínací ztráty jsou zanedbatelné. Parametry obou testů jsou v Tab. 21 níže, teplota IO L6207 byla změřena na povrchu pouzdra, které je na vrchní vrstvě DPS. Největší teplota je dle předpokladu v okolí chladiče a na jeho povrchu. Také oba odpory bočnicku H-můstku se zahřívají procházejícím proudem. Bohužel vzhledem k nízkému rozlišení termo kamery nelze dobře rozeznat, zda se zahřívají ještě některé další prvky v okolí H-můstku. Lze ale předpokládat, že nejvíce tepla vyzařují zmíněné komponenty.

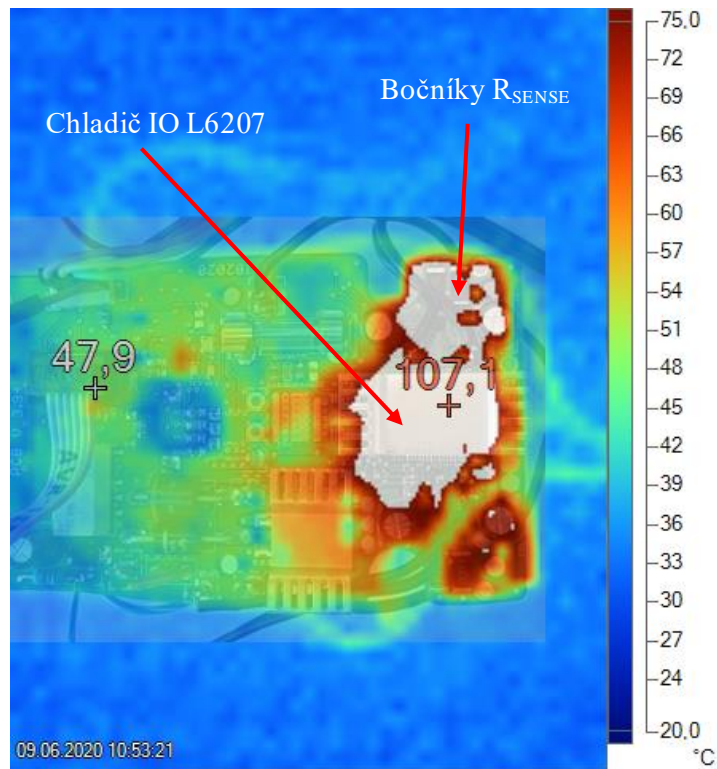
Veličina	1.Měření	2.Meření
Napájecí napětí [V]	12	12
Proud zátěží [A]	1,43	2
Střída PWM [%]	100	100
Teplota L6207 [°C]	69,3	110,8

Tab. 21: Parametry testů teploty testeru



Obr. 65: Teplota testeru při proudu zátěží 1,43 A





Obr. 66: Teplota testeru při proudu zátěži 2 A

Byl proveden výpočet teploty přechodu pro proud H-můstkem  $I_{RMS} = 2$  A a teplotě okolí  $22$  °C. Na základě tohoto výpočtu byla nalezena hodnota tepelného odporu přechod-prostředí  $R_{THJA} = 20$  °C/W. Pro výpočet tepelného odporu  $R_{THJA}$  byl použit Excel. Ze známé teploty naměřené termo kamerou a parametrů H-můstku byla opakovaným dosazováním různých hodnot tepelného odporu  $R_{THJA}$  nalezena výsledná hodnota tepelného odporu  $R_{THJA}$ , která odpovídá naměřené teplotě H-můstku termokamerou. Parametry odporu tranzistorů v sepnutém stavu  $R_{DSON}$  byly uvažovány stejně jako v případě vzorce (30)  $1,12$  Ω.

Spínací ztráty  $P_{RDS}$  v sepnutém stavu jsou:

$$P_{RDS} = I_{RMS}^2 \cdot R_{DSON} = 4 \cdot 1,12 = 4,48 \text{ W} \quad (55)$$

Vlastní příkon L6207 při napájecím napětí  $V_{CC} = 12$  V a vlastním spotřebě proudu  $I_H 0,005$  A je:

$$P_V = V_{CC} \cdot I_H = 12 \cdot 0,005 = 0,06 \text{ W} \quad (56)$$

Celkový ztrátový výkon  $P_{TOT}$  při teplotě okolí  $22$  °C:

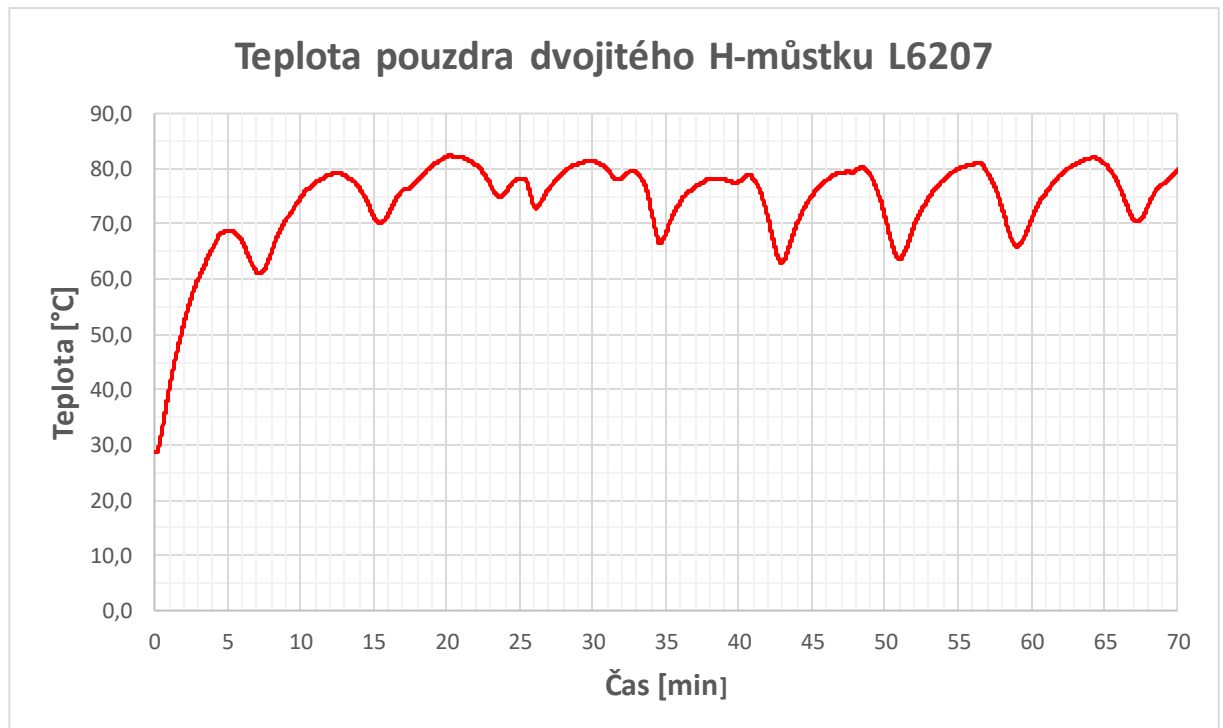
$$P_{TOT} = P_V + P_{RDS} = 0,06 + 4,48 = 4,54 \text{ W} \quad (57)$$

Pro kontrolu správnosti nalezeného tepelného odporu  $R_{THJA}$  spočítáme teplotu přechodu  $t_j$  pro ztrátový výkon  $P_{TOT} = 4,54 \text{ W}$ , teplotu okolí  $t_{AMB} = 22 \text{ °C}$  a tepelného odporu přechod-prostředí  $R_{THJA} = 20 \text{ °C/W}$ :

$$t_j = t_{AMB} + R_{THJA} \cdot P_{TOT} = 22 + 20 \cdot 4,54 = 112,8 \text{ °C} \quad (58)$$

Tato vypočtená hodnota koresponduje s naměřenou hodnotou  $110,8 \text{ °C}$ . Je tedy zřejmé, že navížený způsob chlazení není pro plné zatížení optimální. Tepelný odpor přechod-prostředí  $R_{THJA}$  je vyšší téměř o  $5 \text{ °C/W}$  oproti výpočtu maximálního proudu v kapitole 2.2.5. Neznáme však skutečnou teplotu čipu, ale pouze teplotu na povrchu obvodu.

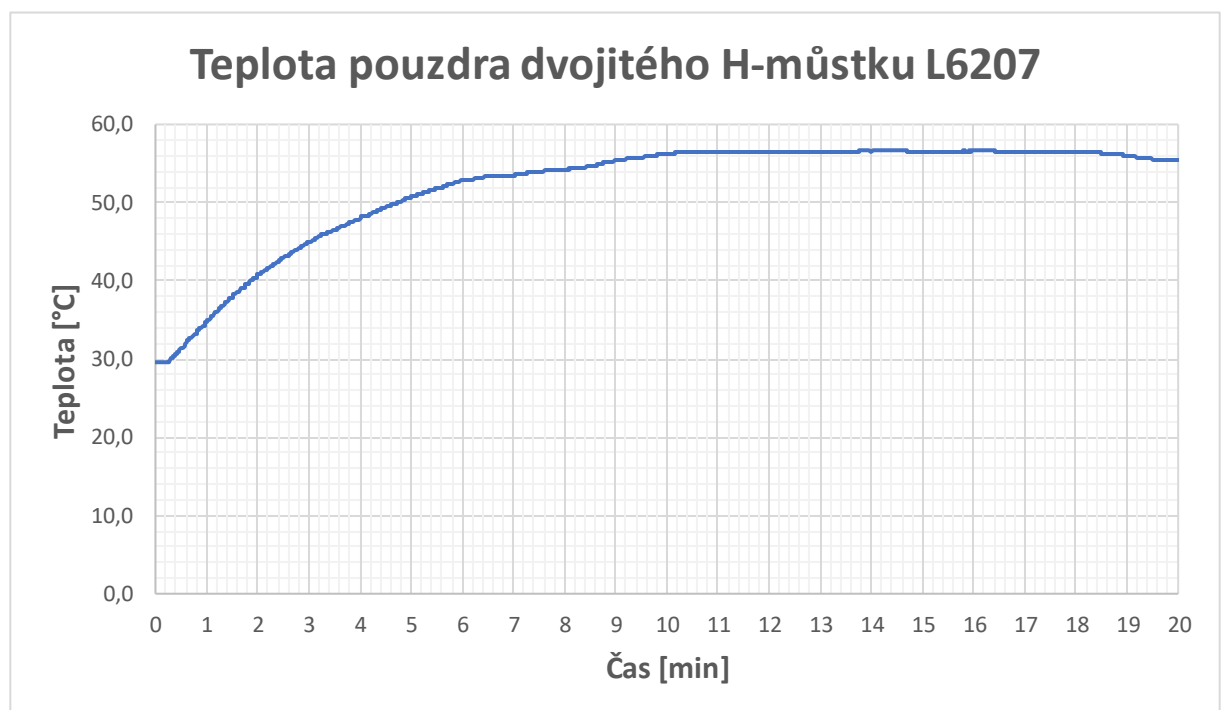
Pro simulaci reálného použití testeru byl spuštěn test v automatickém módu 2, kdy se mění frekvence i střída PWM automaticky rychlostí danou potenciometry. Střída PWM se mění o dvojnásobnou velikost než frekvence PWM. Perioda změny střídavy byla přibližně  $0,5 \text{ s}$ , perioda změny frekvence PWM přibližně  $1,4 \text{ s}$ . Výsledek měření teploty je na Obr. 67, maximální teplota pouzdra H-můstku je  $82,3 \text{ °C}$  při teplotě okolí  $28,5 \text{ °C}$ . Teplotní fluktuace jsou způsobené proměnlivou zátěží, v okolí 50% střídavy PWM proud klesá, proto se pouzdro stihne ochladit.



Obr. 67: Teplota pouzdra H-můstku během automatického testu proměnlivé střídavy a frekvence PWM

Další zatěžovací test byl proveden v režimu automatické zpětnovazební regulace polohy. Zde již nedochází k tak velkým skokům, střída PWM, a tedy i proud zátěží, nedosahuje maximálních hodnot. Teplota se ustálí okolo 56 °C, naměřené maximum bylo 56,7 °C při teplotě okolí 28,5 °C. Je zřejmé, že teplota pouzdra dvojitého H-můstku L6207 bude velmi záviset mimo povahy a velikosti zátěže také na zvoleném režimu a rychlosti změny frekvence a střídání PWM.

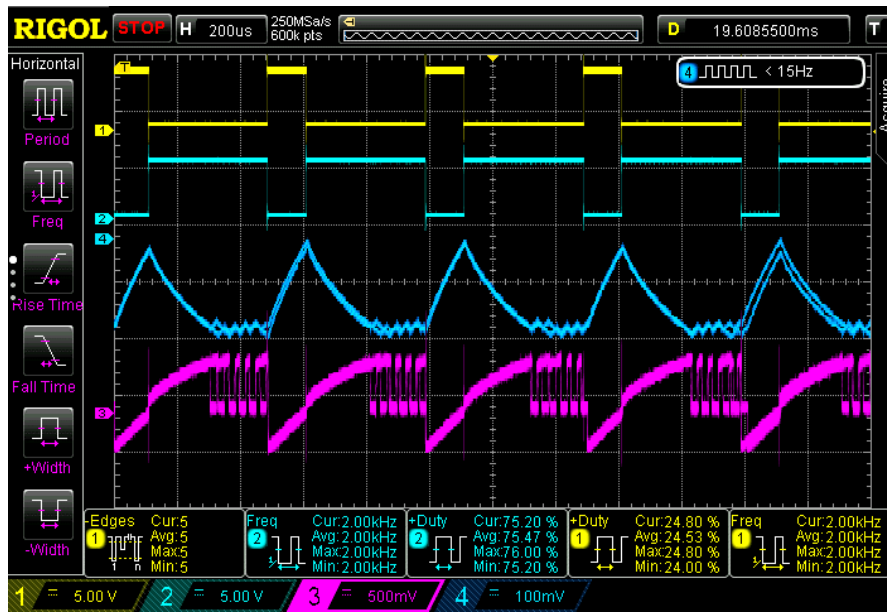
Měření byla provedena multifunkčním přístrojem Testo 480 s připojeným termočlánkem, který byl zafixován pomocí teplovodivé podložky v mezeře mezi chladičem a pouzdem dvojitého H-můstku L6207 (Chladič IO L6207 neodvádí teplo přímo z pouzdra IO L6207, ale je připojen na vrchní vrstvě DPS, která slouží pro chlazení).



Obr. 68: Teplota pouzdra H-můstku během automatického testu zpětnovazební regulace polohy

## 2.5.2 Naměřené průběhy proudu a napětí

Na Obr. 69 níže je oscilogram zobrazující průběh vstupního PWM signálu pro řízení H-můstku na kanálech 1 (výstup OC1B) a 2 (výstup OC1A). Kanál 3 zobrazuje průběh napětí na rezistoru  $R_{SENSE}$  pro měření proudu H-můstku a kanál 4 výstup z proudové sondy s převodem 100 mV/A, tedy proud zátěži (škrtkící klapka z vozu Škoda Roomster kód motoru BXW). Je zde dobře patrný zásah interní regulace proudu, který nenaroste na maximální hodnotu danou impedancí obvodu. Tento průběh je pro frekvenci PWM 2 kHz, střídu 25 % a pro nastavení výstupu bipolární PWM, tedy locked anti-phase řízení.



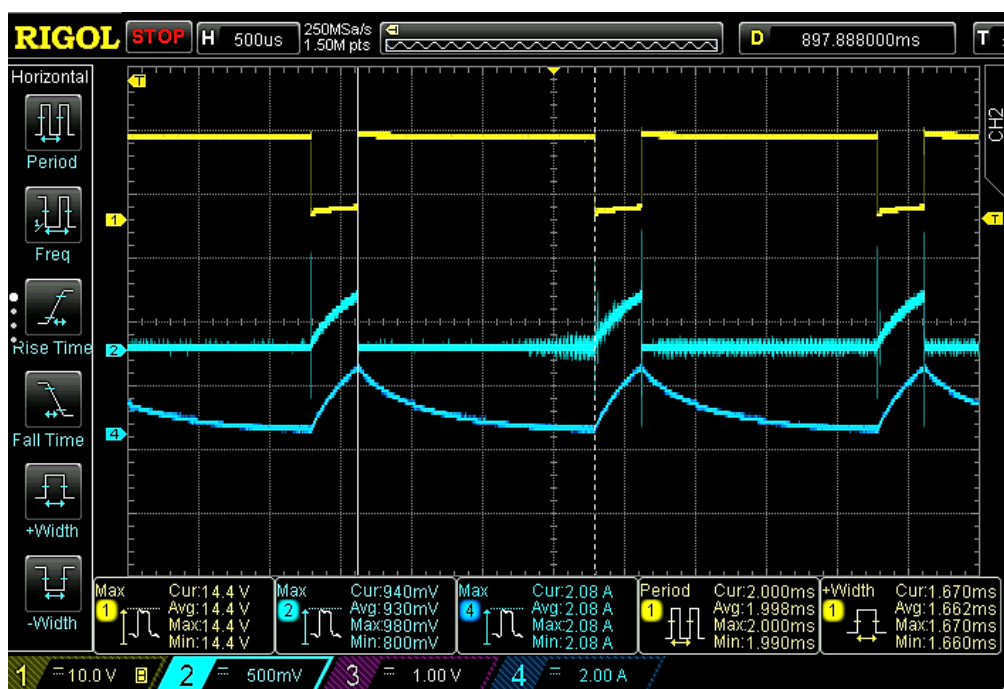
Obr. 69: Průběh PWM signálu, proudu zátěži a napětí na bočniku H-můstku



Obr. 70: Průběh proudu zátěži a napětí na bočniku H-můstku bez proudového omezení

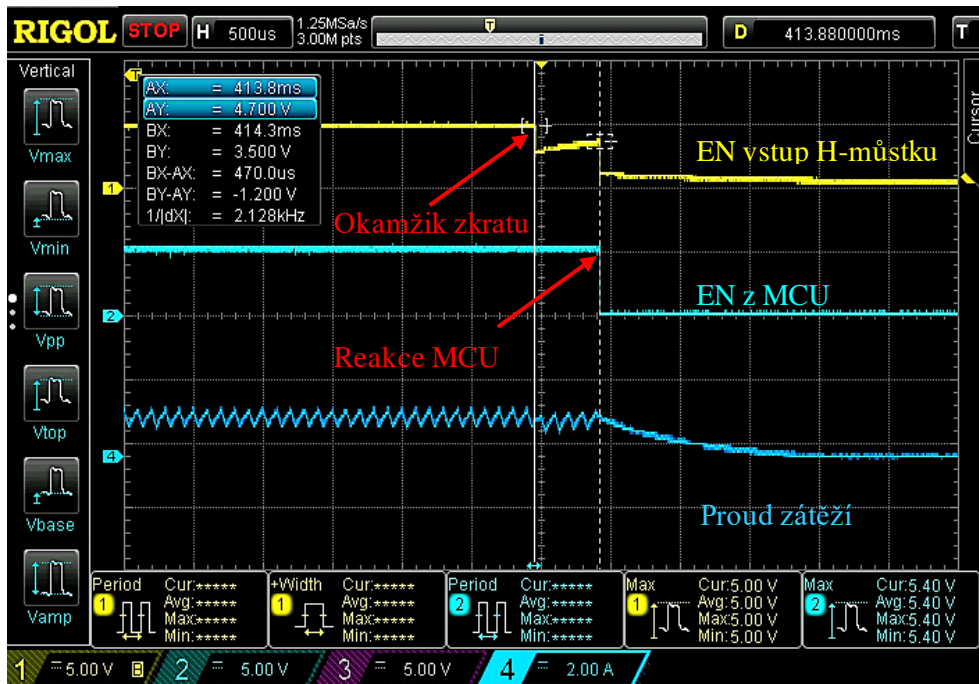
Na Obr. 70 lze popsat jevy při řízení metodou locked anti-phase. Při přepínání diagonální dvojice tranzistorů v H-můstku proud zátěží (kanál 4) ještě chvíli protéká stejným směrem. Jakmile klesne na nulu, tak se proud zátěží otočí a napětí na bočniku H-můstku (kanál 2) opět začne narůstat v kladných hodnotách. Parametry použitého signálu PWM byly: frekvence PWM 500 Hz, střída 40 %. Jak je vidět na oscilogramu, proud nebyl interně omezován, v menu byla zvolen maximální proud 4 A. Jako zátěž byla použita škrtkící klapka popsána v kapitole 1.1.1.

Na dalším obrázku byl již nastaven jiný režim spínání H-můstku, a to unipolární PWM. Detaily jsou uvedeny v řádku 3 v Tab. 10. V tomto režimu dochází při vypnutí H-můstku k postupnému poklesu proudu, který cirkuluje přes horní dva tranzistory H-můstku, viz Obr. 28. V tuto chvíli tedy neteče proud přes bočník H-můstku a není možné měřit proud zátěží. To je patrné na oscilogramu na Obr. 71 níže. Na kanálu 1 je měřené napětí na zátěži (strana spínaná PWM), kanál 2 zobrazuje napětí na bočniku H-můstku, kanál 3 zobrazuje proud zátěží měřený proudovou sondou s převodem 100 mV/A. Měření proběhlo s nastavenou frekvencí PWM 500 Hz a střídou 17 % na stejné škrtkící klapce, jako v předchozím měření.

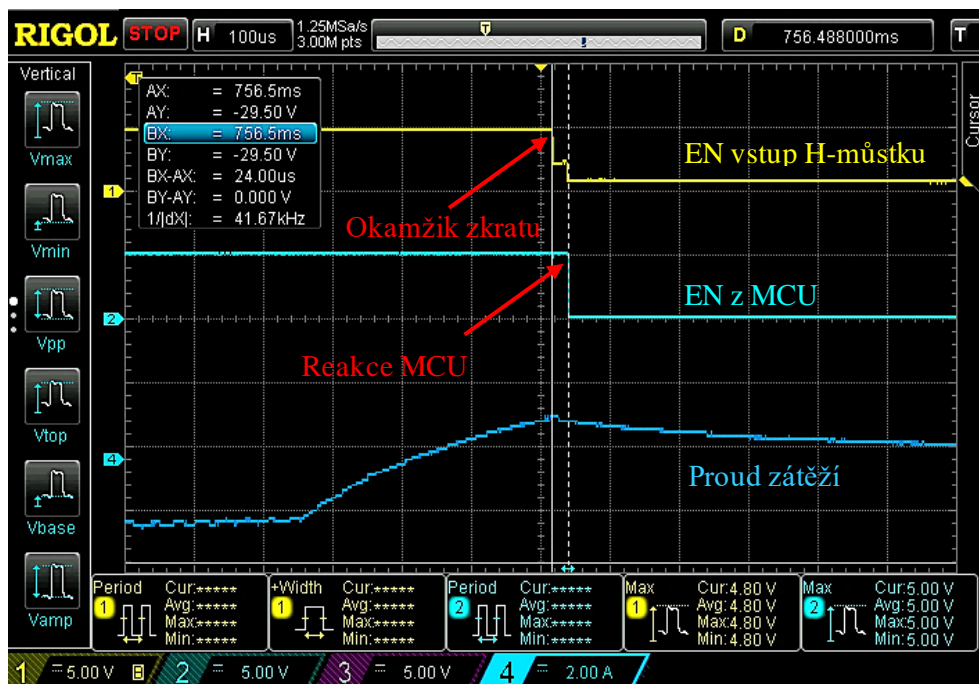


Obr. 71: Průběh proudu a napětí na zátěži a průběh napětí na bočniku H-můstku:

V rámci testování byla vyzkoušena i reakce na zkrat na výstupu H-můstku. Sondy osciloskopu byly připojeny na oba konce odporu  $R_{EN}$ , viz Obr. 27 Naměřený průběh napětí na vstupu  $EN_A$  odpovídá popsanému průběhu na Obr. 37 a Obr. 38 z aplikační poznámky [21]. Z Obr. 72 je patrné, že MCU zareaguje na zkrat uzemněním výstupu EN z MCU za asi 24  $\mu$ s. Kanál 1 zobrazuje vstup  $EN_A$  do H-můstku, kanál 2 zobrazuje výstup  $EN_A$  z MCU. Kanál 4 je proud zátěží při střídě PWM 68 %.



Obr. 72: Průběh napětí na pinu  $EN_A$  a průběh proudu zátěží při zkratu frekvence PWM 10 kHz



Obr. 73: Průběh proudu a napětí na zátěži a průběh napětí na bočniku H-můstku:

## 3 Tester krokových motorů

### 3.1 Požadavky na tester

Vzhledem k tomu, že obě verze testerů sdílejí stejný hardware, mají i některé společné požadavky z pohledu obsluhy a funkčnosti. Z principu však chybí některé funkce využívající PWM, například signálový výstup pro emulaci senzorů, nebo výkonový výstup s MOSFET tranzistorem. Tento tester je spíše jednoúčelově zaměřen. Je určen pouze pro testování dvoufázových krokových motorů. Krokové motory nejsou příliš v akčních členech automobilů rozšířeny.

#### Technické parametry

- Napájecí napětí max. 18 V / 28 V
- Maximální trvalý výstupní proud<sup>8</sup> 4 A
- Rozsah periody kroku 10 ms – 2 s
- Rozsah kroků 0-1023
- Počet kroků v jednom směru 0-1023
- Nadproudová ochrana proti zkratu
- Ochrana proti přehřátí

#### Požadavky na výstupy:

- Výstup pro řízení bipolárních dvoufázových krokových motorů
- Možnost nastavení full step / half step krokování

#### Uživatelské funkce:

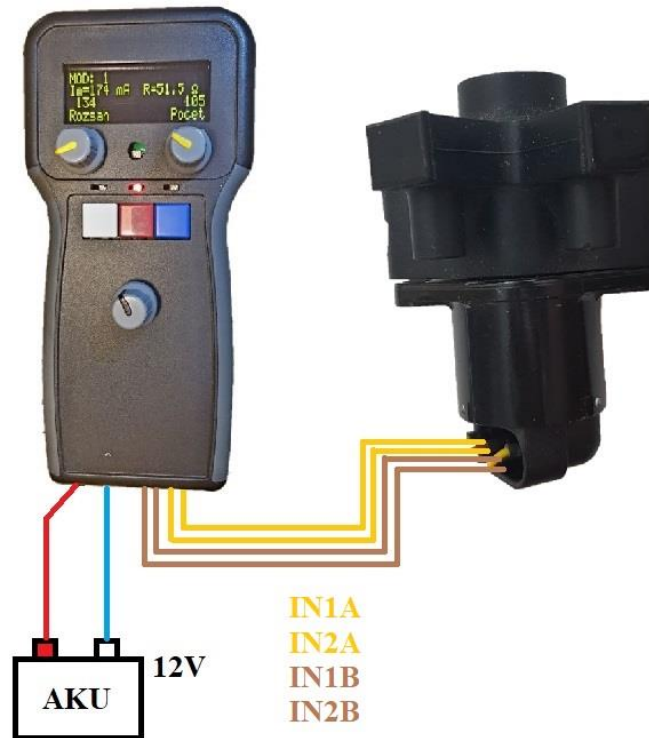
- Obsluha tlačítka a potenciometry, případně enkodér
- Uživatelské rozhraní na displeji se zobrazením relevantních hodnot
- Nastavení omezení proudu ve fázích krokového motoru
- Měření proudu ve fázích krokového motoru
- Automatický režim testování, nastavení rozsahu a počtu kroků
  - Krokují tam a zpět, počet kroků nastavuje pravý potenciometr, rozsah kroků (omezení) nastavuje potenciometr levý

---

<sup>8</sup> Omezeno tavnou pojistkou 4 A

### 3.2 Popis hardwaru

Hardware je totožný s PWM generátorem s bipolárním výstupem, viz kapitola 2.2. Jediným rozdílem je to, že jsou využity namísto jednoho oba H-můstky v pouzdře IO L6207. Na Obr. 74 je pak typické připojení akčního členu s krokovým motorkem. Tento akční člen je konkrétně obtokový ventil pro regulaci volnoběhu z motocyklu.



Obr. 74: Připojení akčního členu s krokovým motorkem k testeru

### 3.3 Řídící software

Obslužný software pro tester byl napsán v Atmel studiu verze 7.0 jako projekt v jazyce C++. Níže je seznam hlavičkových souborů použitých knihoven. Standartní knihovny jsou součástí instalace programu Atmel studio. Nastavení projektu je shodné s popisem v kapitole 2.4.2 Použité knihovny. Seznam hlavičkových souborů použitých knihoven je rovněž shodný, pouze následující hlavičkové soubory nebyly použity:

- pgmspace.h – obsahuje funkce pro přístup k datům v paměti flash
- PSD\_polohovy.h – PSD\_polohovy.h – obsahuje funkce pro obsluhu číslicového PID regulátoru v polohovém tvaru



### 3.3.1 Použité periférie MCU a jejich parametry

Pro řízení byly použity stejné periférie, jako v případě PWM generátoru s bipolárním výstupem, viz kapitola 2.4.3. Pouze 16bitový časovač/čítač TC1 nebyl vůbec použit, spínací sekvence pro H-můstky jsou generovány softwarově. Pouze je odlišné nastavení 8bitového čítače/časovače TC0, který řídí časování celé aplikace. Režim čítače/časovače TC0 je již vysvětlen v kapitole 2.4.3.1 Časovač/čítač TC0, rozdíl je v odlišné době periody přetečení.

### 3.3.2 Časovač/čítač TC0

Základní jednotka časování aplikace je 1 ms. Proto je nutné přepočítat komparační hodnotu čítače/časovače TC0, která je jiná než u PWM generátoru s bipolárním výstupem.

Komparační hodnotu pro OCR0A registr vypočteme podle vzorce (59) jako:

$$OCR0A = t * \frac{f_{CLK}}{N} - 1 = 0,001 * \frac{16 \cdot 10^6}{64} - 1 = 249 \quad (59)$$

Kde  $t$  je požadovaný časový interval [s],  $f_{CLK}$  taktovací frekvence mikrokontroleru [Hz],  $N$  je hodnota předděličky časovače [-].

Dělička 64 byla zvolena z důvodu jediné možnosti dosažení 1 ms intervalu na 8-bit časovači (čítací rozsah 0 až 255) při taktovací frekvenci MCU 16 MHz.

<b>N</b>	1	8	64	256	1024
<b>OCR0A</b>	15999	1999	249	61,5	14,625

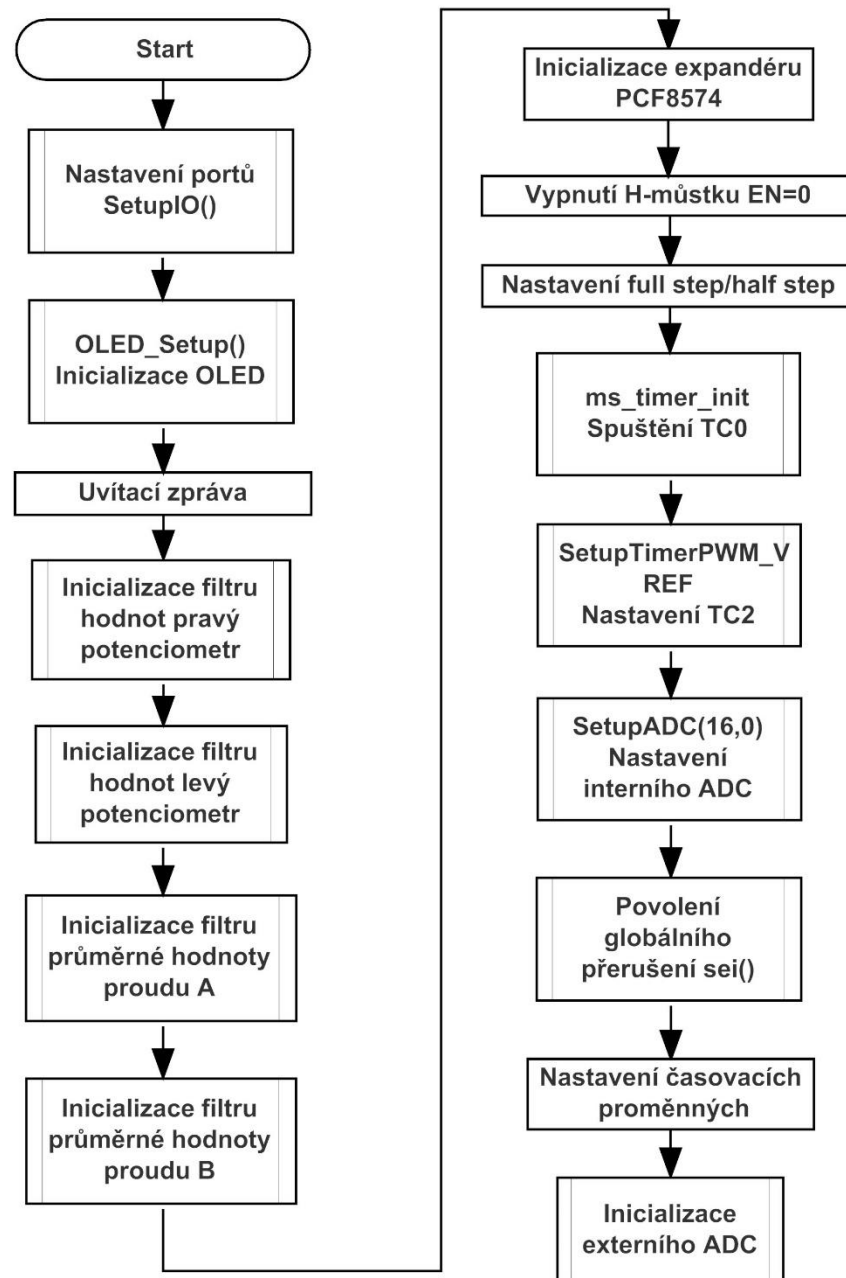
Nastavení časovače TC0 zajišťuje také funkce `ms_timer_init()`, jako u předchozího testeru s tím rozdílem, že komparační hodnota je 249.

### 3.3.3 Použitá přerušení

Použitá přerušení jsou shodná s předchozím testerem a jsou popsána v kapitole 2.4.3.6 Použitá přerušení. Pouze přerušení `TIMER1_OVF` čítače/časovače TC1 není použito vzhledem k tomu, že ani samotný časovač/čítač TC1 není použitý (u krokového motoru není potřeba generovat PWM z MCU). Nově je použito externí přerušení `INT1` pro vyhodnocení zkratu na druhém H-můstku IO L6207, který ovládá druhou fázi krokového motorku.

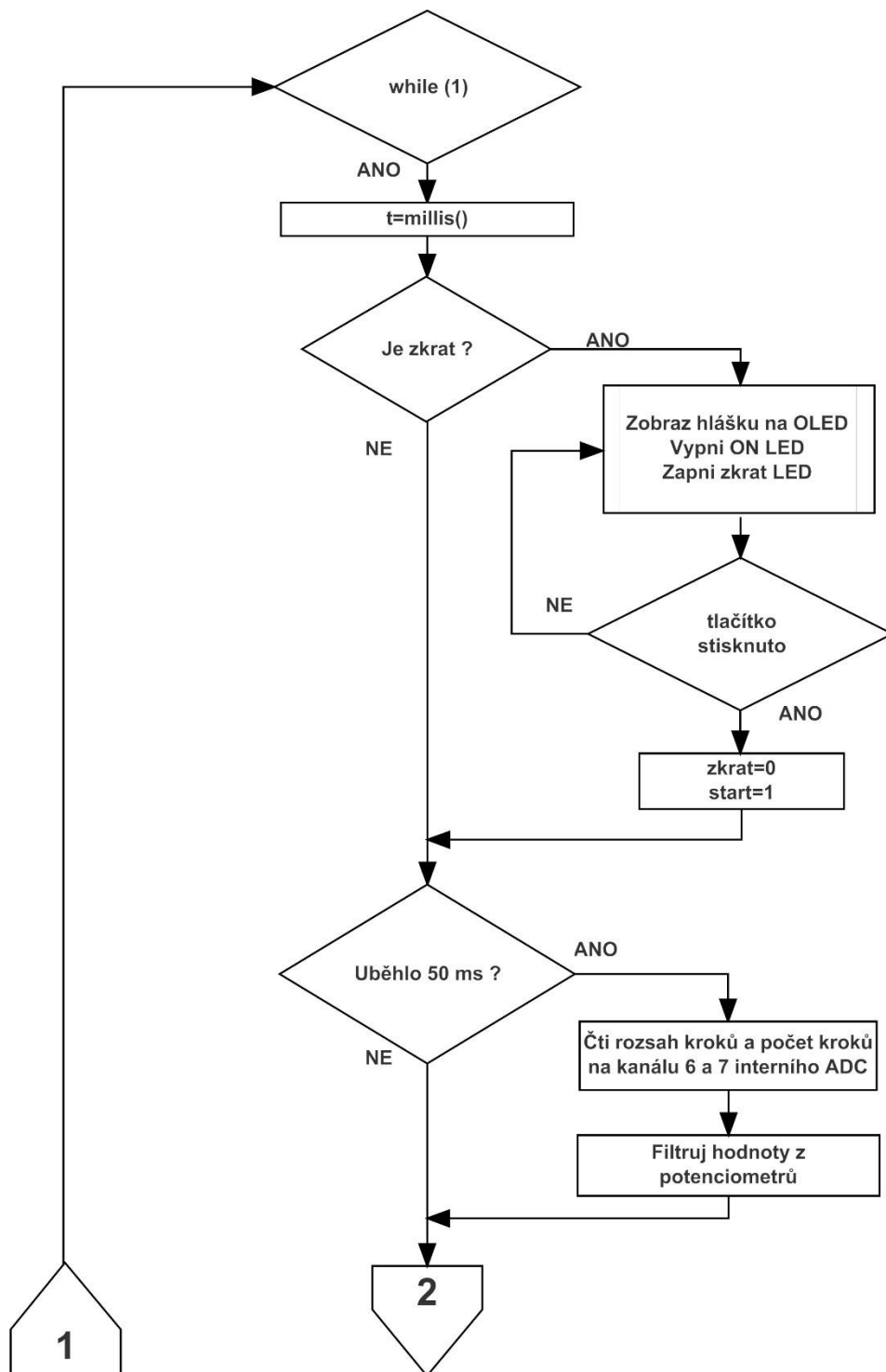
### 3.3.4 Hlavní smyčka a časování aplikace

Ve funkci *Main()* je nutno ještě před vykonáváním nekonečné smyčky nastavit potřebné proměnné. Jedná se o nastavení portů MCU a vnitřních periferií, inicializaci OLED displeje, zobrazení startovací zprávy, inicializaci filtrů, a dalších externích periferií, viz níže. Po vykonání této sekvence již následuje nekonečná smyčka programu.

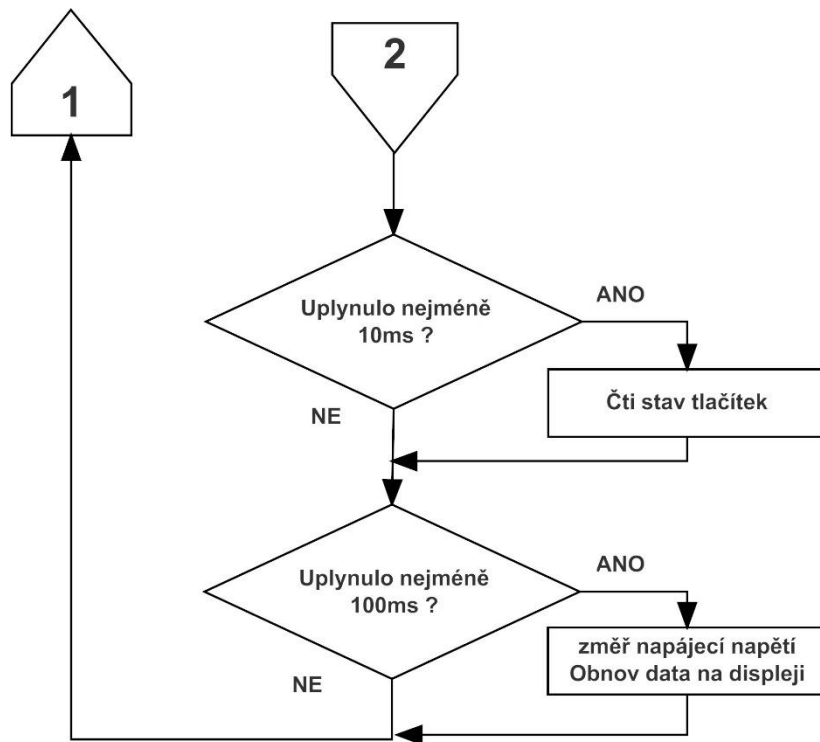


Obr. 75: Vývojový diagram funkce *Main()* testeru krokových motorů

O časování aplikace se stará opět časovač/čítač TC0. Ten ale v tomto případě přetéká po 1 ms. V obsluze přerušení `TIMER0_COMPA_vect` je inkrementována proměnná `t_ms`, která je násobkem 1 ms. Při každém průchodu hlavní smyčky je zavolána funkce `millis()`, která vrací čas od zapnutí testeru a dojde k inkrementaci proměnné `t`. Tato proměnná je použita ve vyhodnocení podmínky, zda již uplynul čas pro opětovné vykonání definované části kódu. Na začátku hlavní smyčky programu dojde k nastavení proměnné `t`, poté se kontroluje hodnota proměnné `zkrat`, která je nastavena na 1 v případě volání funkce přerušení `INT0` nebo `INT1`. Nekonečná smyčka obsahuje tři časované bloky kódu. První z nich řeší čtení hodnot ovládacích potenciometrů každých 50 ms, to je nastaveno makrem `POT_READ_PERIOD_MS`. Druhý blok řeší čtení tlačítek s periodou 10 ms nastavenou makrem `BTN_READ_PERIOD_MS` a poslední blok zobrazení aktuálních dat na displeji každých 50 ms nastavených makrem `OLED_REFRESH_TIME_MS`. Zkrat se zde vyhodnocuje stejně jako v případě testeru PWM s bipolárním výstupem principem popsáním v kapitole 2.4.4 Ovládací prvky.



Obr. 76: Zjednodušený vývojový diagram hlavní smyčky programu testeru pro krokové motory 1.část



Obr. 77: Zjednodušený vývojový diagram hlavní smyčky programu testeru pro krokové motory 2.část

### 3.3.5 Ovládací prvky

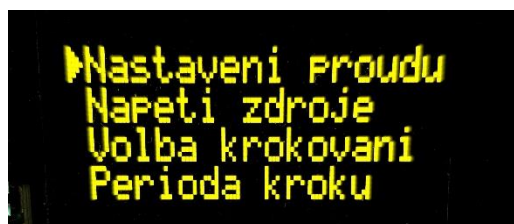
Zpracování signálu z ovládacích prvků a hardwarové osazení je shodné s popisem v kapitole 2.4.4 předchozího testeru. Rozdíl je pouze ve funkci potenciometrů. Levým potenciometrem se nastavuje rozsah kroků a pravým potenciometrem počet kroků, programově je pak signál filtrován SMA filtrem s průměrovacím oknem o šířce 5 vzorků.

```

rozsah_kroku = Read_adc(POT_RANGE);           //čti hodnoty POT rozsah
pocet_kroku  = Read_adc(POT_COUNT);          //čti hodnoty POT počet

avg1 = sma(rozsah_kroku,&filterSMA1,buffer1,K1); // filtrace 1. signálu
avg2 = sma(pocet_kroku,&filterSMA2,buffer2,K2); // filtrace 2. signálu
  
```

Menu je ovládáno enkodérem, stejně jako u předchozího testeru, pouze zatím nenabízí takovou škálu automatických režimů. Po zapnutí krokuje automaticky tam a zpět počet kroků daným potenciometry.



Obr. 78: Obsah menu testeru pro krokové motory

### 3.3.6 Generování spínacích kombinací

Tuto funkci obstarává mikrokontroler, nebylo použito žádného speciálního obvodu pro generování spínacích kombinací. Ty jsou generovány softwarově. Při tomto způsobu řízení je nutno zajistit stálou periodu vystavení logických úrovní na výstupní porty MCU pro řízení obou H-můstků. Vzhledem k delší časové obsluze zejména OLED displeje po I<sup>2</sup>C sběrnici byl proto kód pro nastavení spínací kombinace umístěn do obsluhy přerušení `TIMER0_COMPA_vect` čítače/časovače TC0.

Samotné spínací kombinace vychází z kapitoly 1.3.3 Způsoby krokování. Je možné je dohledat v datových listech speciálních IO pro generování právě těchto spínacích kombinací např. L297 od firmy STMicroelectronics [25]. Tyto spínací kombinace pro obě fáze jsou uloženy ve dvou tabulkách pro full step a half step:

```
//2 fázové řízení, Normal drive, obě fáze vždy sepnuté, 0b 0 0 0 0 IN2A
IN1A IN1B IN2B
full_step_table[] = {0b00001001,0b00001010,0b00000110,0b00000101};

//2 fázové řízení, Half step, 0b 0 0 0 0 IN2A IN1A IN1B IN2B
half_step_table[] =
{0b00000001,0b00000101,0b00000100,0b00000110,0b00000010, 0b00001010,
0b00001000, 0b00001001};
```

Tato tabulka je čtena s periodou, která je uložena v proměnné `step_update_period_ms`. Tato proměnná má po zapnutí hodnotu 30, základní perioda kroku je tedy 30 ms. Uživatel si může v menu nastavit periodu kroku od 10 ms do 2 s. Podle typu zvoleného krokování je po vymaskování nastavena logická hodnota na výstupech PD0, PD1, PB1 a PB2. Tyto výstupní piny jsou připojeny na vstupy IN1B, IN2B a IN1A, IN2A obou H-můstků.

```
// PORTB: x x x x x 1 1 x    // PB2 - H-IN2A, PB1 - H-IN1A
// PORTD: x x x x x x 1 1    // PD1 - H-IN1B, PD0 - H-IN2B
PORTB = (PORTB & 0b11111001) | ((val & 0b00000011)<<1);
PORTD = (PORTD & 0b11111100) | ((val & 0b00001100)>>2);
```

Dále je vykonán kód, který řídí čtení tabulky zleva nebo zprava o počet kroků daný ovládacími potenciometry. Tím se motor točí na jednu nebo na druhou stranu.

### 3.3.7 Měření proudu

Měření proudu je poměrně problematické. Po otestování několika více či méně složitých algoritmů, po diskuzi se zadavatelem projektu jsem dospěl k řešení, které považuji prozatím za vyhovující pro tuto aplikaci. Měří se proudy v obou fázích krokového motoru, odběr vzorků je synchronizován s krokováním (přesněji na spínací kombinaci vyčtenou z tabulky). V přerušení časovače/čítače TC0 `ISR(TIMER0_COMPA_vect)` probíhá měření napětí na bočniku H-můstku A, pokud je pořadí spínací kombinace 0. Měření napětí na bočniku H-můstku B probíhá, když je pořadí spínací kombinace 2. Obsah tabulky `full_step_table[]` je v předchozí kapitole 3.3.6 Generování spínacích kombinací. Pořadí spínací kombinace pro měření proudu bylo zjištěno experimentálně pomocí osciloskopu a proudové sondy. Odběr vzorků nastává, ve chvíli, kdy je předpoklad, že hodnota proudu v dané fázi krokového motoru je nejvyšší a kladná z pohledu napětí na měřícím bočniku.

```
if (PORTC & (1<<PORTC0)) //Jen pokud jsou H-můstky zapnuté
{
    if (poradi == 2)
    {
        Read_adc(SENSE_B); //Čtení kanal ADC3
    }
    if (poradi == 0)
    {
        Read_adc(SENSE_A); //Čtení kanal ADC2
    }
}
```

Naměřená hodnota z A/D převodníku je poté filtrována SMA filtrem o šířce průměrovacího okna 20 vzorků. Ukázka kódu pro měření proudu ve funkci `uint16_t Read_adc(uint8_t channel)` jednou z fází H-můstku A (pro druhý můstek je analogické):

```
if (channel == 2)
{
    avg3 = sma(ADC,&filterSMA3,buffer3,K3); //Filtr SMA
    avg_max_sample_A = avg3;
}
```

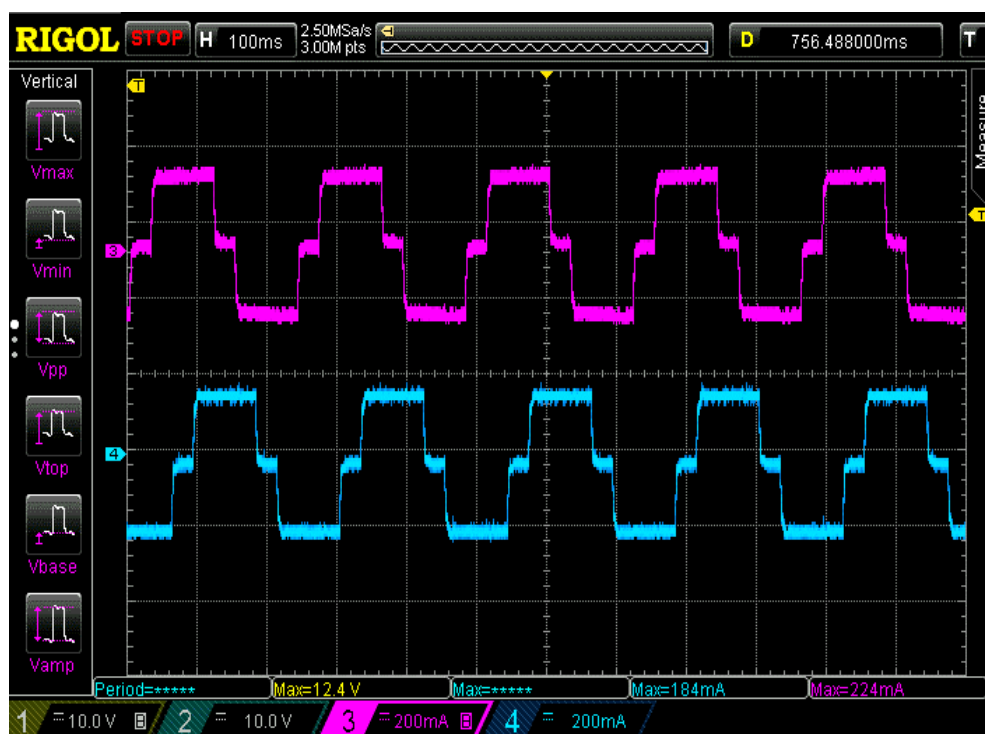
Po přepočtu se potom zobrazí proud v mA na OLED displeji. Vzhledem k tomu, že proud krokovým motorkem ve většině aplikací je poměrně malý, není zde využito dobře rozlišení A/D převodníku. Bylo by vhodné snížit referenční napětí interního A/D převodníku MCU na 1,1 V a dělič napětí s ovládacími potenciometry, jak je zmíněno v kapitole 2.4.4. V programu je řešeno měření proudu obou fází krokového motoru pro diagnostiku, zda není některá z fází krokového motoru vadná.

### 3.3.8 Reakce na nadproud

Je shodná jako u předchozího testeru a všechny mechanismy jsou popsány v kapitole 2.4.7 Reakce na nadproud. Rozdíl je pouze ve vyhodnocování obou externích přerušení *INT0* a *INT1* místo jednoho. Aktivace alespoň jedno z nich vyvolá reakci na zkrat.

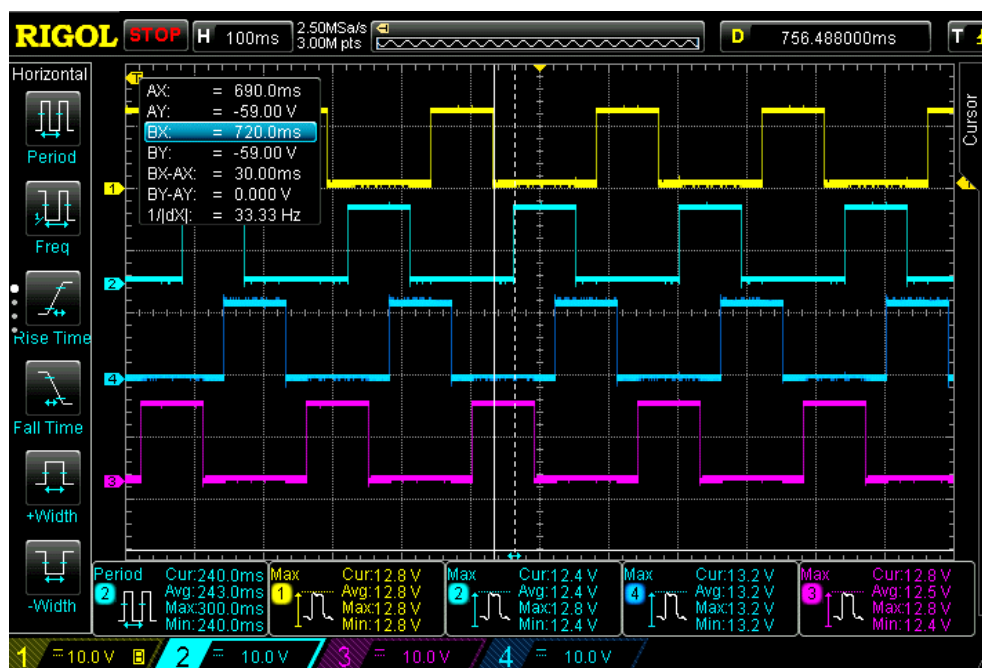
## 3.4 Naměřené průběhy proudu a napětí

Na oscilogramu na jsou průběhy napětí na jednotlivých fázích krokového motoru, resp. jednotlivých výstupech obou H-můstků. Je zde patrná perioda kroku 30 ms. Na Obr. 79 je oscilogram průběhu proudu oběma fázemi při krokování half step. Obě měření byla provedena s obtokovým ventilem z motocyklu z Obr. 74.



Obr. 79: Proud oběma fázemi krokového motorku při krokování half step





Obr. 80: Napětí na výstupech obou H-můstků – spínací kombinace

Na Obr. 80 výše jsou na kanály osciloskopu CH1 až CH4 připojeny výstupy z H-můstků. Pořadí je následující: CH1 je OUT1<sub>A</sub>, CH2 je OUT2<sub>A</sub>, CH3 je OUT1<sub>B</sub>, CH4 je OUT2<sub>B</sub>.

## 4 Závěr

Výsledkem této práce měl být software pro řízení různých typů testerů akčních členů na jedné hardwarové platformě. V důsledku se jednalo v podstatě o návrh celého nového zařízení, které má být konkurenceschopné na trhu. Výsledkem této práce je koncept, na kterém je možno dále stavět. Určitě však ještě není vhodný k prodeji. Univerzálnost řešení představuje konstruktérský a programátorský problém, kdy je nutné vybrat optimální řešení ovšem za cenu kompromisu. Vybraný mikrokontroler má určitá omezení ve velikosti paměti, rychlosti a počtu vnitřních periférií. Tato omezení vyžadují zkušeného programátora, který je bude schopen překonat. Protože je toto můj první rozsáhlý projekt na platformě AVR mimo Arduina, bylo pro mne skloubení všech funkcí největším úskalím diplomové práce.

Za nejslabší článek návrhu považuji vybraný H-můstek L6207. První slabina tohoto H-můstku spočívá v absenci informace o interním omezování proudu. Uživatel není nijak informován, že dochází k omezení proudu zátěží. Dále měření proudu je možné pouze na externích bočnicích, je tedy nutno měřit poměrně malá napětí v rozsahu daném velikostí bočniců H-můstku. Proudový výstup (proudové zrcadlo) můstku DRV8874 je výhodnější, není potřeba mít externí bočník na měření proudu, postačí jen rezistor proti zemi na výstupu H-můstku, na kterém se posléze měří napětí. Odpor v sepnutém stavu  $R_{\text{DSON}}$  tranzistorů v H-můstku L6207 je velký, to způsobuje značné oteplení pouzdra H-můstku při větších prouděch. Například H-mústek DRV8874 má při teplotě 80 °C  $R_{\text{DSON}}$  pouze asi 0,2  $\Omega$ , což je pětina oproti H-můstku L6207. Jak bylo uvedeno v kapitole 2.5.1, dosahované teploty při provozu jsou značné. Po uzavření do krabičky bude nutné zřejmě chladit H-mústek aktivně. Proud bude nutné také softwarově omezit do bezpečných hodnot. Řešením je také vyměnit H-mústek za jiný.

Implementovaný PSD regulátor polohy škrťací klapky pracuje poměrně dobře, otázkou však je, jak bude fungovat se škrťacími klapkami jiných typů. Jiné škrťací klapky mohou mít jiné mechanické vlastnosti, vliv pružiny, tření a hystereze budou zřejmě odlišné. Nastavení jedné sady parametrů nemusí vyhovovat jiným škrťacím klapkám. Řešením by bylo přidat do uživatelského menu různé sady nastavení parametrů PSD regulátoru, což však klade nároky na kvalifikovanost obsluhy a celkově zesložituje použití testeru. Při identifikaci parametrů škrťací klapky se také ukázalo, jak obtížné je v praxi rychle a přesně klapku řídit kvůli nelinearitám vlivem tření a pružiny. V praxi bude důležité spíše vyhodnotit, zda jsou výstupní napětí senzorů polohy stabilní a v rozmezí daném servisním manuálem.

Tester pro krokové motory funguje spolehlivě, ale s použitým hardwarem nebylo možné dosáhnout automatické detekce dorazů. Tato nastavitelná funkce je například ve dvojitém H-můstku L9942 od firmy STMicroelectronics. Ten měří rychlost nárustu proudu motorem a dokáže vyhodnotit zastavení krokového motoru. DPS testeru je připravena na osazení dvojitého H-můstku DRV8874, který by měl být pro tento tester vhodnější.

Použití displeje se ukázalo být jako velmi dobrá volba pro zvýšení poměru cena/výkon. Je dobře čitelný i na slunci. Použití grafického menu umožňuje vytvořit sadu testů s již přednastavenými parametry pro konkrétní druh diagnostické operace. Obsluha se tedy nemusí zabývat s technickými detaily. Na některé ovládací funkce by bylo lepší místo potenciometrů použít spíše enkodér, nebo kombinaci tlačítek. Ovládání potenciometry je dosti citlivé.

Další možný rozvoj obou verzí testeru je závislý na požadavcích trhu, pro jaké uživatele bude primárně určen. Pro techniky v autoservisech je potřeba zařízení, které bude mechanicky a teplotně odolné a nebude vyžadovat takovou míru technických znalostí. Funkcionalitu a možné režimy zařízení je vhodné diskutovat s uživateli a stanovit na základě praktických zkušeností s poruchami na akčních členech. Zatímco pro laboratorní využití nebo jiné specializované využití se nabízí další možnosti vývoje, například ovládání přes PC prostřednictvím rozhraní UART, které by umožňovalo detailnější nastavení parametrů, logování dat, tvorbu grafů.

## Seznam použité literatury

- [1] GREGORA, Stanislav a Zdeněk MAŠEK. *Elektronické a mechatronické systémy v konstrukci silničních vozidel*. Pardubice: Univerzita Pardubice, 2008. ISBN 978-80-7395-082-8.
- [2] Adaptive Headlights. *Hella* [online]. [cit. 2020-05-20]. Dostupné z: <https://www.hella.com/techworld/uk/Technical/Automotive-lighting/Adaptive-headlights-663/#>
- [3] Headlamp levelling system. *Hella* [online]. [cit. 2020-05-20]. Dostupné z: <https://www.hella.com/techworld/ae/Technical/Automotive-lighting/Headlamp-levelling-system-838/>
- [4] BMW E90 335i wiring diagram. *NewTIS* [online]. [cit. 2020-05-20]. Dostupné z: <https://www.newtis.info/tisv2/a/en/e90-335i-lim/wiring-functional-info/body/lighting/automatic-headlight-beam-throw-adjustment/pf9cQN2>
- [5] Elon Musk vysvětlil co znamená logo Tesla. *Hybrid.cz* [online]. 5.2.2017 [cit. 2020-05-20]. Dostupné z: <http://www.hybrid.cz/elon-musk-vysvetlil-co-znamenalo-logo-tesla>
- [6] PAVELKA, Jiří, Jiří JAVŮREK a Zdeněk ČEŘOVSKÝ. *Elektrické pohony*. Praha: České vysoké učení technické, 1996. ISBN 80-01-01411-8.
- [7] MĚŘIČKA, Jiří, Václav HAMATA a Petr VOŽENÍLEK. *Elektrické stroje*. Vyd. 2. Praha: Vydavatelství ČVUT, 2000. ISBN 80-01-02109-2.
- [8] PAVELKA, Jiří a Zdeněk ČEŘOVSKÝ. *Výkonová elektronika*. Vyd. 2. přeprac. Praha: Vydavatelství ČVUT, 2000. ISBN 80-01-02094-0.
- [9] MAŠEK, Zdeněk. *Řízení krokových motorků* [Word dokument]. Pardubice, 2019 [cit. 2020-04-01]. Dostupné z: [www.stag.upce.cz](http://www.stag.upce.cz). Studijní text. Univerzita Pardubice.
- [10] Structure of Motores de Passo. *Oriental motor* [online]. [cit. 2020-05-20]. Dostupné z: <https://www.orientalmotor.com.br/motores-de-passo/technology/stepper-motor-overview.html>
- [11] NOVÁK, Jaroslav. *Elektrické Pohony* [PDF dokument]. 2011 [cit. 2020-05-24]. Dostupné z: [www.stag.upce.cz](http://www.stag.upce.cz). Studijní opora. Univerzita Pardubice, Dopravní fakulta Jana Pernera.
- [12] EITEL, Lisa. Can-stack linear actuators: How the motor and screw components work. *Linear Motion Tips* [online]. JULY 24, 2017 [cit. 2020-05-20]. Dostupné z: <https://www.linearmotiontips.com/can-stack-linear-actuators-motor-screw-component-work/>
- [13] COLINS, Danielle. What stepper motor type is best for high torque? *Linear Motion Tips* [online]. MAY 6, 2017 [cit. 2020-05-20]. Dostupné z: <https://www.motioncontroltips.com/what-stepper-motor-type-is-best-for-high-torque/>
- [14] MAŠEK, Zdeněk. *Matematické modely vybraných systémů* [Word dokument]. Pardubice, 2013 [cit. 2020-06-17]. Dostupné z: [www.stag.upce.cz](http://www.stag.upce.cz). Studijní text. Univerzita Pardubice.

- [15] THOMASSON, A. a L. ERIKSSON. *Model-Based Throttle Control using Static Compensators and Pole Placement*. 2011, **66**(4), 717-727. DOI: 10.2516/ogst/2011137. ISSN 1294-4475. Dostupné také z: <http://ogst.ifpenergiesnouvelles.fr/10.2516/ogst/2011137>
- [16] EA OLEDM204. *Display Visions Electronic Assembly* [online]. Munich: ELECTRONIC ASSEMBLY, 2020 [cit. 2020-06-21]. Dostupné z: <https://www.lcd-module.com/fileadmin/eng/pdf/doma/oledm204-ae.pdf>
- [17] SSD1311. *SOLOMON SYSTECH* [online]. Hong Kong: SOLOMON SYSTECH, 2020 [cit. 2020-06-21]. Dostupné z: <http://www.solomon-systech.com/en/product/display-ic/oled-driver-controller/ssd1311/>
- [18] *ATmega328P datasheet* [online]. [cit. 2020-06-22]. Dostupné z: [http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P\\_Datasheet.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf)
- [19] MAŠEK, Zdeněk. *PMRTP-Přednáška 8* [Word dokument]. Pardubice, 2008 [cit. 2020-06-23]. Dostupné z: [www.stag.upce.cz](http://www.stag.upce.cz). Studijní text. Univerzita Pardubice.
- [20] *L6207 datasheet: Rev 3* [online]. STMicroelectronics, October 2018 [cit. 2020-06-26]. Dostupné z: <https://www.st.com/resource/en/datasheet/l6207.pdf>
- [21] *AN1762 Application note* [online]. STMicroelectronics, 2003 [cit. 2020-06-26]. Dostupné z: [https://www.st.com/resource/en/application\\_note/cd00004482-l6205-l6206-l6207-dual-full-bridge-drivers-stmicroelectronics.pdf](https://www.st.com/resource/en/application_note/cd00004482-l6205-l6206-l6207-dual-full-bridge-drivers-stmicroelectronics.pdf)
- [22] *PCF8574 datasheet* [online]. Texas Instruments, March 2015 [cit. 2020-06-26]. Dostupné z: <https://www.ti.com/lit/ds/symlink/pcf8574.pdf>
- [23] *ADS101x datasheet* [online]. Texas Instruments, January 2018 [cit. 2020-06-26]. Dostupné z: <https://www.ti.com/lit/ds/symlink/ads1015.pdf>
- [24] TANTOS, Andras. *Lock Anti-Phase Drive* [online]. 2011 [cit. 2020-06-30]. Dostupné z: <https://www.modularcircuits.com/blog/articles/h-bridge-secrets/lock-anti-phase-drive/>
- [25] *L297 datasheet* [online]. STMicroelectronics, 2001 [cit. 2020-07-6]. Dostupné z: <https://www.st.com/resource/en/datasheet/l297.pdf>
- [26] ETC Electrothrottle Electronic Throttle Control Actuator and Solenoid Tester. *Automotive Diagnostics and Publishing* [online]. [cit. 2020-07-06]. Dostupné z: [http://www.autodiagnosicsandpublishing.com/etc\\_electrothrottle\\_electronic\\_throttle\\_tester.html](http://www.autodiagnosicsandpublishing.com/etc_electrothrottle_electronic_throttle_tester.html)
- [27] *G3-REA-Master (Touch) Advanced Actuator Tester/Programmer* [online]. 2019 [cit. 2020-07-06]. Dostupné z: <https://www.g3concepts.com/g3-rea-master>
- [28] *Working with Atmel AVR Microcontroller Basic Pulse Width Modulation (PWM) Peripheral* [online]. [cit. 2020-07-15]. Dostupné z: <http://www.ermicro.com/blog/?p=1971>

## Seznam značek a zkratek

Značka	Význam	Jednotka
I	Proud	A
U	Napětí	V
t	Čas	s
f	Frekvence	Hz
C	Kapacita	F
R	Odpor	$\Omega$
L	Indukčnost	H
$\tau$	Časová konstanta	s
$\Phi$	Magnetický tok	Wb
$\omega$	Úhlová rychlost	rad/s
M	Moment	Nm
T	Teplota	$^{\circ}\text{C}$

Zkratka	Význam
MCU	Mikrokontroler
PWM	Pulzně šířková modulace
CGROM	Grafická paměť znaků jen pro čtení
DPS	Deska plošných spojů
PCB	Deska tištěných spojů
MOSFET	Tranzistor ovládaný polem
LED	Dioda emitující světlo
IO	Integrovaný obvod
SMA	Filtr s klouzavým průměrem
A/D	Analogově digitální
TC	Časovač / Čítač
EGR	Recirkulace výfukových plynů
OLED	Organická dioda emitující světlo
AVR	Alf Vegard RISC procesor
USART	Univerzální synchronní/asynchronní přijímač a vysílač
VREF	Referenční napětí

## Seznam obrázků

Obr. 1: Škrtkící klapka z vozu Škoda Roomster 1.4 63 kW .....	14
Obr. 2: Připojení škrtkící klapky ve voze Škoda Octavia II generace .....	15
Obr. 3: EGR ventil z vozu Mazda 6 2.0 DI a schéma připojení k ECU .....	16
Obr. 4: ISM modul Hella [3].....	17
Obr. 5: Ilustrativní schéma připojení motoru naklápění světlometu k řídicí jednotce ve voze BMW E90 [4].....	17
Obr. 6: Řez stejnosměrným motorem se třemi póly [5].....	18
Obr. 7: Elektrické schéma stejnosměrného motoru s permanentním magnetem, upraveno z [6] .....	19
Obr. 8: Stejnosměrný měnič napětí s aktivní zátěží, upraveno z [8] .....	21
Obr. 9: Zvyšující stejnosměrný měnič, upraveno z [8].....	23
Obr. 10: Odezva motoru na jeden krok, upraveno z [10] .....	24
Obr. 11: Řez krokovým motorkem s proměnnou reluktancí, uspořádání fází a spínací diagram, převzato z [9]. .....	25
Obr. 12: Konstrukce krokového motoru s permanentními magnety typu can-stack [12].....	26
Obr. 13: Řez hybridním krokovým motorem, upraveno z [13] .....	27
Obr. 14: Typy vinutí krokových motorů, převzato z [9].....	28
Obr. 15: Průběh proudu fázemi a tabulka spínacích kombinací pro plný krok, jednofázové řízení, převzato z [9] .....	29
Obr. 16: Průběh proudu bipolárním vinutím s nízkou (A) a vysokou (B) frekvencí krokování [9].....	30
Obr. 17: Budič s PWM regulací proudu a průběhy proudu vinutím, napětí na vinutí [9] .....	31
Obr. 18: Připojení škrtkící klapky k testeru .....	33
Obr. 19: Vzhled testeru.....	34
Obr. 20: Blokové schéma generátoru PWM s bipolárním výstupem .....	35
Obr. 21: Zjednodušené blokové schéma I/O expandéru PCF8574 [22] .....	38
Obr. 22: Typické zapojení A/D převodníku ADS1015 .....	40
Obr. 23: Modul alfanumerického displeje EA OLEDM204 [16].....	41
Obr. 24: Tvar pouzdra PowerSO36 IO L6207 [20] .....	42
Obr. 25: Slow decay recirkulace proudu [9].....	42
Obr. 26 : Vnitřní blokové schéma H-můstku L6207 [20].....	44
Obr. 27: Způsob připojení vstupů EN <sub>A</sub> , EN <sub>B</sub> k MCU [20].....	44

Obr. 28: Stavby H-můstku v režimu slow decay .....	45
Obr. 29: Spínací režim locked anti-phase H-můstku a toky proudu [24] .....	46
Obr. 30: Vlevo průběh proudu a napětí na motoru střída PWM 50 %, vpravo střída PWM > 50 % . Locked anti-phase řízení. [24] .....	47
Obr. 31: Spínací režim sign-magnitude H-můstku [24] pro směr vpřed a vzad .....	47
Obr. 32: Proud a napětí na motoru při řízení PWM typu sign-magnitude.....	48
Obr. 33: Schéma bloku proudové regulace L6207 [20].....	49
Obr. 34: Průběh omezení výstupního proudu H-můstku [20].....	50
Obr. 35: D/A převodník pro nastavení proudového omezení v H-můstku [20] .....	51
Obr. 36: Zjednodušené schéma nadproudové ochrany L6207 [20].....	53
Obr. 37: Detail průběhu výstupního proudu při zásahu nadproudové ochrany, vliv $t_{DELAY}$ a $t_{DISABLE}$ [21].....	55
Obr. 38: Detail průběhu výstupního proudu při zásahu nadproudové ochrany, vliv $t_{DISABLE}$ [21] .....	55
Obr. 39: Vliv velikosti $C_{EN}$ $R_{EN}$ na délku $t_{DISABLE}$ a $t_{DELAY}$ [21].....	56
Obr. 40: Detail chladiče dvojitého H-můstku z boku .....	58
Obr. 41: Pohled shora na chladič dvojitého H-můstku a na chladičí plochu na DPS.....	58
Obr. 42: Regulovaná soustava jako systém s jedním vstupem a výstupem [14] .....	59
Obr. 43: Statická charakteristika škrtkovací klapky .....	63
Obr. 44: Odezva na skok střídání 63,48 % až 78,13 % .....	65
Obr. 45: Vybraný úsek odezvy na skok střídání pro identifikaci parametrů .....	65
Obr. 46: Porovnání naměřené hodnoty a výstupu z modelu v Matlabu.....	66
Obr. 47: Model soustavy v Simulinku .....	68
Obr. 48: Detail odezvy klapky z vozu Škoda Roomster BXW na skok žádané polohy.....	69
Obr. 49: Celkový průběh odezvy klapky z vozu Roomster BXW na žádanou polohu .....	70
Obr. 50: Porovnání průběhu střídání a polohy klapky z vozu Roomster BXW s modelem v Simulinku .....	70
Obr. 51: Detail porovnání průběhu žádané a skutečné polohy klapky z vozu Roomster BXW .....	71
Obr. 52: Detail 2 porovnání průběhu žádané a skutečné polohy klapky z vozu Škoda Roomster BXW s modelem v Simulinku .....	71
Obr. 53: Porovnání aktuálního nastavení regulátoru v MCU s navrženými hodnotami PID tuneru .....	73
Obr. 54: Časový diagram časovače/čítače TC0 v režimu CTC [18].....	78



Obr. 55: Zjednodušené schéma připojení IO L6207 k MCU.....	80
Obr. 56: Časový diagram časovače/čítače TC2 v režimu Phase Correct PWM [18] .....	81
Obr. 57: Časový diagram časovače/čítače TC1 v režimu Phase and Frequency Correct PWM [18].....	83
Obr. 58: Ukázka první strany menu na displeji .....	89
Obr. 59: Vývojový diagram funkce Main() testeru PWM s bipolárním výstupem .....	91
Obr. 60: Zjednodušený vývojový diagram hlavní smyčky programu 1.část .....	93
Obr. 61: Zjednodušený vývojový diagram hlavní smyčky programu 2.část .....	94
Obr. 62: Zjednodušený vývojový diagram hlavní smyčky programu 3.část .....	95
Obr. 63: Zjednodušený vývojový diagram hlavní smyčky programu 4.část .....	96
Obr. 64: Časový diagram nastavení registrů ICR1 a OCR1n, průběh výstupního signálu PWM [28].....	104
Obr. 65: Teplota testeru při proudu zátěži 1,43 A .....	112
Obr. 66: Teplota testeru při proudu zátěži 2 A .....	113
Obr. 67: Teplota pouzdra H-můstku během automatického testu proměnlivé střídy a frekvence PWM.....	114
Obr. 68: Teplota pouzdra H-můstku během automatického testu zpětnovazební regulace polohy .....	115
Obr. 69: Průběh PWM signálu, proudu zátěži a napětí na bočníku H-můstku .....	116
Obr. 70: Průběh proudu zátěži a napětí na bočníku H-můstku bez proudového omezení .....	116
Obr. 71: Průběh proudu a napětí na zátěži a průběh napětí na bočníku H-můstku: .....	117
Obr. 72: Průběh napětí na pinu EN <sub>A</sub> a průběh proudu zátěži při zkratu frekvence PWM 10 kHz .....	118
Obr. 73: Průběh proudu a napětí na zátěži a průběh napětí na bočníku H-můstku: .....	118
Obr. 74: Připojení akčního členu s krokovým motorkem k testeru .....	120
Obr. 75: Vývojový diagram funkce Main() testeru krokových motorů .....	122
Obr. 76: Zjednodušený vývojový diagram hlavní smyčky programu testeru pro krokové motory 1.část .....	124
Obr. 77: Zjednodušený vývojový diagram hlavní smyčky programu testeru pro krokové motory 2.část .....	125
Obr. 78: Obsah menu testeru pro krokové motory .....	125
Obr. 79: Proud oběma fázemi krokového motorku při krokování half step .....	128
Obr. 80: Napětí na výstupech obou H-můstků – spínací kombinace.....	129

## Seznam tabulek

Tab. 1: Přehled typů pohonů pro akční členy .....	13
Tab. 2: Tabulka spínacích kombinací plný krok, dvoufázové řízení [9] .....	29
Tab. 3: Tabulka spínacích kombinací poloviční krok [9] .....	29
Tab. 4: Použité piny MCU a jejich funkce .....	37
Tab. 5: Zapojení výstupních pinů expandéru PCF8574.....	38
Tab. 6: Rozsahy vstupních napětí převodníku ADS1015 a odpovídající rozlišení .....	39
Tab. 7: Vybrané elektrické parametry A/D převodníku ADS1015 .....	39
Tab. 8: Vybrané parametry displeje EA OLEDM204 [16].....	41
Tab. 9: Vybrané parametry H-můstku L6207 v pouzdře PowerSO36 [20].....	43
Tab. 10: Pravdivostní tabulka spínacích kombinací H-můstku [20].....	45
Tab. 11: Spínací kombinace H-můstku v režimu.....	46
Tab. 12: Provedené skoky střídáky a identifikované hodnoty $k_{0klapky}$ a $\tau_m$ z Matlabu.....	64
Tab. 13: Porovnání parametrů regulátoru MCU a výstupu z PID tuneru .....	72
Tab. 14: Režimy a funkce implementované v testeru .....	75
Tab. 15: Seznam použitých periférií MCU .....	77
Tab. 16: Seznam použitých přerušení MCU .....	77
Tab. 17: Závislost komparační hodnoty na dělicím poměru předděličky pro 5 ms periodu ....	79
Tab. 18: Závislost TOP hodnoty na dělicím poměru předděličky .....	82
Tab. 19: Režimy a funkce implementované v testeru .....	90
Tab. 20: Použití analogových vstupů externího A/D převodníku ADS1015.....	97
Tab. 21: Parametry testů teploty testeru.....	112

## **Seznam příloh**

Přílohou práce je CD s programem pro řízení PWM generátoru s bipolárním výstupem a pro řízení testeru krokových motorů. Dále skripty a model v Matlabu, datasheety a obrazový materiál použitý v diplomové práci.