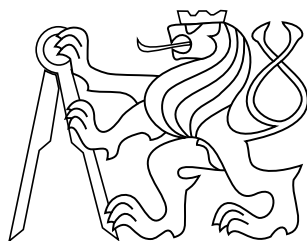


bakalářská práce

# Zpětnovazební regulátory s mikrořadičem pro demonstrační účely

*Jan Světlík*



Květen 2017

vedoucí práce: doc. Ing. Jan Fischer, CSc.

České vysoké učení technické v Praze  
Fakulta elektrotechnická, Katedra měření





## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Jan Světlík**

Studijní program: **Kybernetika a robotika**  
Obor: **Senzory a přístrojová technika**

Název tématu česky: **Zpětnovazební regulátory s mikrořadičem pro demonstrační účely**

Název tématu anglicky: **Microcontroller Based Feedback Controllers for Demonstration Purposes**

### Pokyny pro vypracování:

Pro demonstrační účely navrhnete a realizujete číslicové zpětnovazební regulátory s mikrořadiči řady STM32F042, případně STM32L072. Navrhnete příklady regulátorů: regulátor polohy, impulsní snižující regulátor napětí („step-down“), „lineární“ regulátor napětí, regulátor teploty a další typy regulátorů podle Vašeho výběru. Uveďte potřebné teoretické výpočty související s řešením daných regulátorů. Konkrétní praktickou realizaci řešte tak, aby ji bylo možno podle vzoru vytvořeného v rámci této práce opakovaně sestavit s využitím jednoduchých bloků na nepájivém kontaktním poli.

### Seznam odborné literatury:

- [1] Yiu, J.: The Definitive Guide to ARM® Cortex®-M0 and Cortex®-M0+ Processors
- [2] STMicroelectronics: RM0091: STM32F0x2 Reference manual
- [3] STMicroelectronics: DS10147 STM32F042 Data

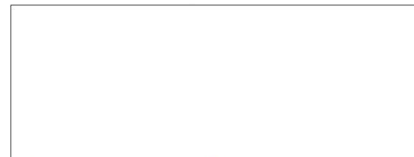
Vedoucí bakalářské práce: doc. Ing. Jan Fischer, CSc.

Datum zadání bakalářské práce: 24. ledna 2017

Platnost zadání do<sup>1</sup>: 30. září 2018



Prof. Ing. Jan Holub, Ph.D.  
vedoucí katedry



Prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 24. 1. 2017

<sup>1</sup> Platnost zadání je omezena na dobu tří následujících semestrů.



## **Poděkování**

Tímto děkuji vedoucímu práce doc. Ing. Janu Fischerovi, CSc. za nebývalou ochotu a čas, který mi věnoval během konzultací.

## **Prohlášení**

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

## **Abstrakt**

Cílem této bakalářské práce je navrhnout a realizovat zpětnovazební regulátory s využitím mikrořadiče řady STM32 pro demonstrační účely. Tyto regulátory jsou navrženy za účelem použití ve výuce. Z tohoto důvodu je při návrhu kladen důraz především na jednoduchost a nízké náklady. Pro tyto účely byly zvoleny čtyři základní demonstrační úlohy. Ke každé regulační úloze je proveden návrh hardwaru, který byl následně otestován a jeho funkčnost ověřena sérií experimentů. Výsledkem práce je kompletní návrh funkčních regulačních úloh současně s demonstračním firmwarem. Dále byla vyvinuta PC aplikace určená pro testování regulátorů a sběr dat z regulace v reálném čase.

## **Klíčová slova**

demonstrační regulátor; mikrořadič; regulace; zpětná vazba

## **Abstrakt**

The aim of this bachelor thesis is to design and realize feedback regulators using microcontroller STM32. These regulators are supposed to be used for evaluation purposes in classes. The emphasis is on simplicity and low cost while proposing such regulation examples. For demonstration purpose, four different kinds of regulators were designed. Each feedback control example is supported with complete design of hardware components. All regulator examples were tested and the functionality was checked using several experiments. The result is complete design of function regulators along with a firmware developed for demonstration purposes. A PC application has been created in purpose of testing, configuring regulators and capturing real-time data during regulation process.

## **Keywords**

evaluation purpose regulator; microcontroller; regulation; feedback

# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Rozbor a stanovení cílů</b>	<b>2</b>
<b>3</b>	<b>Regulátor napětí</b>	<b>3</b>
3.1	Buck konvertor . . . . .	3
3.1.1	Rozbor činnosti buck konvertoru . . . . .	3
3.2	Realizace regulátoru napětí . . . . .	4
3.2.1	Analýza a návrh obvodových prvků . . . . .	6
3.2.2	Snímání výstupního napětí . . . . .	8
3.3	Využití periferie mikrořadiče . . . . .	10
3.3.1	AD převodníky mikrořadičů řady STM32 . . . . .	10
3.3.2	Hardwarové čítače mikrořadiče řady STM32 . . . . .	11
<b>4</b>	<b>Lineární regulátor napětí</b>	<b>13</b>
4.1	Princip lineárního stabilizátoru napětí . . . . .	13
4.2	Realizace regulátoru . . . . .	14
4.2.1	DA převodník v mikrořadiči STM32L072 . . . . .	17
4.2.2	Návrh obvodových prvků . . . . .	17
4.3	Regulátor napětí s integračním RC článkem . . . . .	20
4.3.1	AD převodníky na mikrořadičích STM32L100 . . . . .	21
<b>5</b>	<b>Regulace polohy</b>	<b>22</b>
5.1	Realizace úlohy . . . . .	22
5.1.1	Realizace koncového stupně s obvodem L6206 . . . . .	22
5.1.2	Snímání polohy . . . . .	24
5.1.3	Vyhodnocení kvadratického signálu . . . . .	24
5.1.4	Generování PWM signálu . . . . .	25
5.2	Demo s vývojovým kitem IHM04A1 . . . . .	25
<b>6</b>	<b>Regulace teploty</b>	<b>27</b>
6.1	Realizace úlohy . . . . .	27
6.1.1	Topný přípravek . . . . .	27
6.1.2	Návrh obvodových prvků topného přípravku . . . . .	30
6.1.3	Snímání teploty . . . . .	31
6.1.4	Návrh obvodových prvků pro snímání teploty . . . . .	32
<b>7</b>	<b>Aplikace pro sběr dat během regulace</b>	<b>34</b>
7.1	Přenos dat . . . . .	34
7.2	Datový rámec pro přenos dat . . . . .	35
<b>8</b>	<b>Softwarová implementace regulátoru</b>	<b>37</b>
8.1	Rozbor PID regulátoru . . . . .	37
8.2	Diskrétní aproximace spojitého PID regulátoru . . . . .	38
8.3	Anti-windup . . . . .	40
8.4	Hardwarové aspekty při implementaci regulátoru . . . . .	42



<b>9 Pomocné aplikace</b>	<b>43</b>
9.1 UART-USB převodník . . . . .	43
9.1.1 Konfigurace UART-USB převodníku . . . . .	43
9.1.2 Implementace brány UART-USB převodníku . . . . .	43
9.2 PWM generátor . . . . .	44
9.2.1 Ovládání PWM generátoru . . . . .	44
9.2.2 Princip činnosti PWM generátoru . . . . .	45
9.2.3 Standardní režim PWM generátoru . . . . .	45
9.2.4 Synchronizace kanálů PWM generátoru . . . . .	46
9.2.5 Obecné vztahy . . . . .	47
9.2.6 Nastavení hodin PWM generátoru . . . . .	47
<b>10 Závěr</b>	<b>49</b>
<b>Přílohy</b>	
<b>A Schémata regulačních úloh</b>	<b>50</b>
<b>B Screenshoty z demonstrační aplikace</b>	<b>52</b>
<b>C Fotografie realizovaných úloh</b>	<b>53</b>
<b>Literatura</b>	<b>54</b>



# 1 Úvod

Podmětem ke vzniku této práce byla absence praktických předmětů na oboru Kybernetika a robotika, ve kterých by si studenti vlastnoručně vyzkoušeli realizaci zpětnovazebních regulátorů. Námětem pro takový předmět je syntéza nabytých znalostí z teoretických předmětů s praktickou dovedností. Díky tomu bude student schopen nejen navrhnout, ale i s využitím dostupných prostředků realizovat zpětnovazební regulátor.

Cílem této práce je tedy navrhnout a realizovat demonstrační aplikace zpětnovazební regulace. Součástí těchto aplikací je ukázka použití mikrořadiče jako zpětnovazebního regulátoru. Úlohy budou koncipovány tak, aby je šlo jednoduše realizovat, například na nepájivém kontaktním poli v rámci výuky. Pro demonstrační účely budou vybrány různé úlohy jak z hlediska regulované veličiny, tak z pohledu samotného regulátoru. Část těchto úloh bude zaměřena na tvorbu regulátoru s akčním zásahem v podobě pulzně šířkově modulovaného signálu a některé úlohy pak využívají digitálně analogového převodníku. V každém případě se jedná o řízení v diskrétním čase, a proto se v této práci zaměřím na aspekty diskrétního řízení. Dále zde budou navrženy úlohy zabývající se regulací jak elektrických veličin, tak i těch neelektrických.

## 2 Rozbor a stanovení cílů

Jak již bylo zmíněno v úvodu, pro demonstrační účely jsou vybrány úlohy, které je snadné realizovat a zároveň svým principem zastupují skutečné regulační problémy.

V této práci se tedy zaměřím na návrh a realizaci samotných regulačních úloh. Dále zde budou probrány jednotlivé aspekty důležité pro návrh a především realizaci číslicového regulátoru s využitím mikrořadiče (dále jen MCU). Budou zde vysvětleny způsoby měření regulovaných veličin, případně jejich převod na napětí. S tímto souvisí i výběr samotných snímačů na základě jejich vlastností. K realizaci úloh budou pro měření regulovaných veličin použity především levné senzory. Důvodem je snadná realizace úloh a dostupnost pro studenty.

Dále zde bude probráno zpracování signálů ze snímačů výstupní veličiny. Budou představeny jednotlivé periferie MCU dostupné ke zpracování signálu ať už číslicového, analogového nebo modulovaného. V následující části se budu věnovat implementaci samotného regulátoru. Zmíním nároky na MCU a možnosti výpočtu akčního zásahu regulátoru. Dále představím PC aplikaci, která poslouží pro demonstraci regulace. K této aplikaci navrhnu demonstrační firmaware pro jednotlivé regulační úlohy. Tato aplikace poslouží pro sledování průběhů regulované veličiny a akčního zásahu v reálném čase. Další možností bude ladění vlastního regulátoru a konfigurace parametrů výstupu regulátoru. V poslední části se zaměřím na popis aplikací, které vznikly jako vedlejší produkt při realizaci regulačních úloh. Těmito aplikacemi jsou PWM generátor s MCU řady STM32 a UART-USB převodník.

Jako první úloha byl vybrán regulátor napětí. Tento regulátor bude realizován jako jednoduchý spínaný zdroj řízený MCU. Regulovanou veličinou bude v tomto případě výstupní napětí, které bude snímáno AD převodníkem. MCU bude periodicky přepočítávat akční zásah v podobě střídy PWM signálu, kterým bude řízen výkonový tranzistor.

Další úloha bude též zaměřená na regulaci napětí. V tomto případě se však jedná o obdobu lineárního regulátoru. Aktivní prvek, kterým je bipolární tranzistor, bude řízen analogovým signálem z DA převodníku MCU na základě změřeného výstupního napětí. Cílem regulace bude nastavovat takový proud tranzistorem, aby na výstupu stabilizátoru bylo při různých zátěžích konstantní napětí.

Realizace regulátoru polohy představuje další z vybraných regulačních úloh. V této úloze provedu návrh a realizaci výkonové části regulátoru s využitím integrovaného obvodu od firmy STMicroelectronics. Jedná se o řízení polohy s využitím stejnosměrného motoru. Poloha motoru bude snímána rotačním optickým inkrementálním snímačem. Akčním zásahem bude v tomto případě opět střída PWM signálu, kterým bude řízen koncový H-můstek.

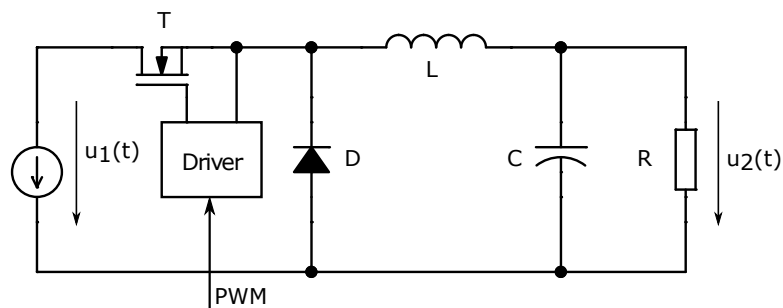
Poslední vybranou úlohou je demonstrace regulace teploty. V úloze bude představena technika snímání teploty odporovým snímačem, kdy je nutné převést měřený odpor na napětí, které lze poté měřit AD převodníkem MCU. V této úloze si student vyzkouší praktickou aplikaci znalostí nabytých v předmětu Senzory a měření. Cílem regulátoru je generovat adekvátní akční zásah pro udržení konstantní teploty na topném prvku.

## 3 Regulátor napětí

V této části práce se budu zabývat realizací jednoduchého regulátoru napětí. Tento regulátor je realizován pomocí spínaného DC/DC měniče, který je řízen mikrořadičem STM32F042. Pro realizaci této úlohy byl zvolen jednočinný propustný měnič. Vybranou topologií je buck konvertor. Jedná se o jednu ze základních topologií spínaných DC/DC měničů.

### 3.1 Buck konvertor

Principiální schéma buck konvertoru je zobrazeno na obrázku 1.

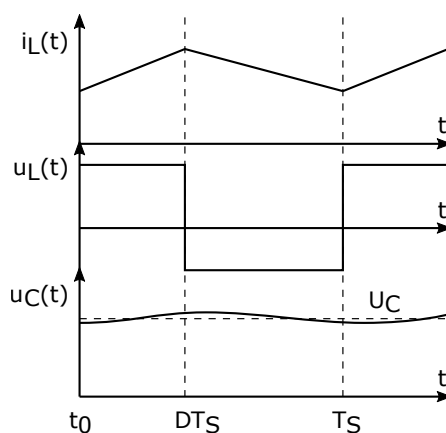


Obrázek 1 Principiální schéma buck konvertoru

Kromě realizace zobrazené na obrázku 1 je možné použít i jiných zapojení. Například místo diody  $D$  může být použit také tranzistor. V takovém případě se jedná o takzvaný synchronní obousměrný měnič.

#### 3.1.1 Rozbor činnosti buck konvertoru

Princip činnosti je znázorněn na obrázku 2, kde je zobrazen průběh proudu cívku  $i_L(t)$ , napětí na cívce  $u_L(t)$  a výstupní napětí kondenzátoru  $u_C(t)$ .

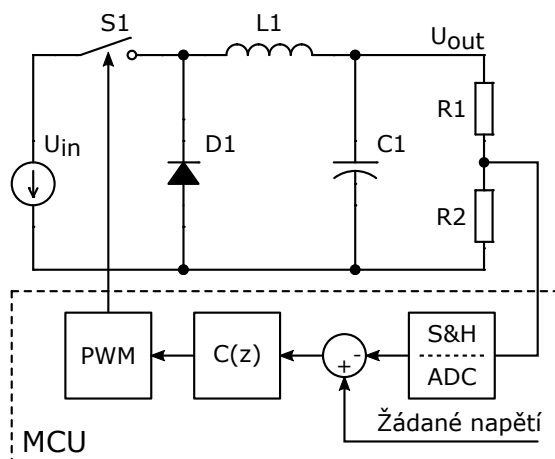


Obrázek 2 Průběhy vybraných veličin v ustáleném stavu

V době sepnutí tranzistoru T protéká indukčností proud do výstupního filtru tvořeného kondenzátorem C a do zátěže R. Pokud zanedbáme zvlnění výstupního napětí, předpokládáme činnost měniče v ustáleném stavu a vstupní napětí je konstantní, je na cívce téměř konstantní napětí. Cívkou protéká lineárně rostoucí proud a v magnetickém obvodu cívky se akumuluje energie. V době rozepnutí tranzistoru se cívka svým indukovaným elektromotorickým napětím snaží udržet proud v obvodu. Tímto dojde k otočení polarity napětí na cívce a otevření diody D, kterou začne protékat proud. V tomto okamžiku, za stejného předpokladu zanedbatelného zvlnění výstupního napětí, protéká indukčností lineárně klesající proud. Akumulovaná energie v jádře cívky je předávána do výstupního filtru a do zátěže. K přenosu energie ze vstupu na výstup měniče dochází jak v době sepnutí, tak v době rozepnutí tranzistoru. Proto se jedná o jednočinný propustný měnič.

## 3.2 Realizace regulátoru napětí

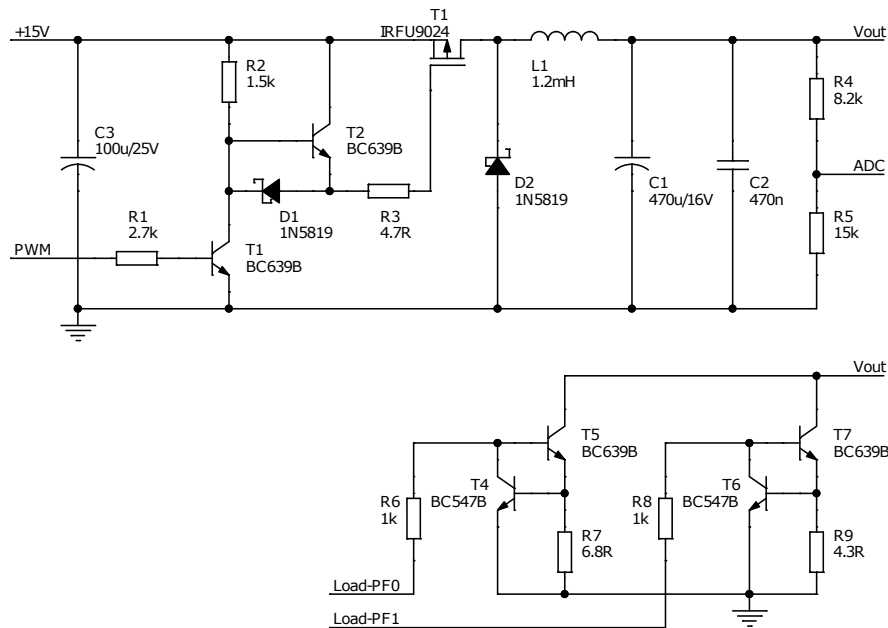
Na obrázku 3 je zobrazeno blokové schéma realizované regulační smyčky. Výstupní napětí měniče je snímáno odporovým děličem. Toto napětí je poté vzorkováno a převáděno interním AD převodníkem mikrořadiče. V následujícím kroku je vypočítána odchylka změřené hodnoty výstupního napětí od požadované hodnoty. Odchylka je použita pro výpočet akčního zásahu diskrétního regulátoru, který je představován blokem  $C(z)$ . Velikost vypočítaného akčního zásahu je hardwarovým čítačem převedena na PWM signál, kterým je řízen výkonový tranzistor.



**Obrázek 3** Blokové schéma regulační smyčky spínaného regulátoru napětí

Jako spínač je použit MOSFET tranzistor. PWM signál generovaný procesorem je nutné ještě výkonově zesílit. Důvodem je, že řízený tranzistor je sice řízen napětím, ale při spínání a vypínání tranzistoru je nutné rychle nabít a vybit vstupní kapacitu gate. Navíc může být prahová úroveň napětí pro sepnutí tranzistoru mnohem vyšší, než je napájecí napětí procesoru. Díky tomu je nutné tranzistoru předřadit budič, který je dimenzován tak, aby byl schopen pokrýt ony proudové špičky při nabíjení a vybíjení tranzistoru. Tento budič tedy poskytuje napětové a proudové zesílení signálu jdoucího z procesoru.

Schéma zapojení buck konvertoru pro realizaci této úlohy je zobrazeno na obrázku 4. Celkové schéma zapojení s mikrořadičem STM32F042 je zobrazeno na obrázku 48 v příloze A.



**Obrázek 4** Schéma zapojení buck konvertoru pro realizaci demonstrační úlohy

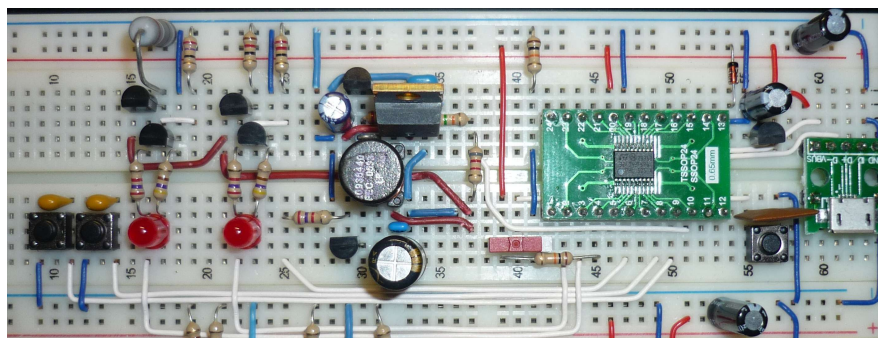
Jako výkonový tranzistor je použit P-kanálový MOSFET tranzistor. Důvodem použití P-kanálového tranzistoru je snadná realizace budicího obvodu tranzistoru. V případě použití N-kanálového MOSFET tranzistoru by bylo nutné použít takzvané plovoucí buzení. Důvodem je, že výstupní cívka a filtr jsou v takovém případě připojeny na source tranzistoru. Řídící signál tranzistoru je tedy nutné posunout o napětí na kterém se nachází source, viz. obrázek 1. K realizaci takového buzení by například v případě synchronního měniče, kde by místo diody  $D_2$  byl další tranzistor, šlo použít buzení pomocí nábojové pumpy. Další možností je pak realizace galvanicky odděleného buzení, kde se nabízí celá řada možných provedení. Tyto techniky buzení tranzistoru jsou však nevhodné pro použití v této úloze, kde je cílem především snadná realizace.

Princip budicího obvodu je velmi snadný. Výstupní signál mikrořadiče je napětově posunut pomocí level-shiffteru, který je tvořen tranzistorem  $T_1$ . Výstup tohoto level-shiffteru je pak vybaven dalším tranzistorem  $T_2$  a rychlou diodou  $D_1$ . V okamžiku, kdy je tranzistor  $T_1$  uzavřený, na vstupu je signál v logické nule, tranzistorem neteče proud, a na jeho kolektoru je plné napájecí napětí. V tomto stavu je dioda  $D_1$  uzavřená a tranzistor  $T_2$  je otevírán odporem  $R_1$ . Na výstupu budiče je tedy napájecí napětí a vstupní kapacita gate tranzistoru je zkratována proti source. Když je tranzistor  $T_1$  otevřen, je na jeho kolektoru minimální napětí  $U_{CEon}$ . V důsledku toho dojde k otevření diody  $D_1$  a zavření tranzistoru  $T_2$ . Tímto je přes diodu  $D_1$  nabíjena kapacita gate tranzistoru  $T_3$  na záporné napětí  $U_{GS}$  a tranzistor  $T_3$  je sepnut. Při použití tohoto zapojení je nutné, aby vstupní napětí konvertoru bylo větší než mezní napětí  $U_{GS}$  tranzistoru, a zároveň nepřesáhlo maximální dovolenou hodnotu. Pro realizaci tohoto měniče bylo zvoleno vstupní napětí 15 V. Zbytek zapojení je typické zapojení buck konvertoru, které je popsáno v sekci 3.1.1.

Zdroj je doplněn dvěma zátěžemi proudového charakteru tvořenými tranzistory  $T_4$ ,  $T_5$ ,  $T_6$  a  $T_7$ . Díky tomu lze generovat skokovou změnu proudu na výstupu a sledovat schopnosti regulátoru a odezvu výstupního napětí. Výhodou proudové zátěže je nezávislost odebíraného proudu na velikosti výstupního napětí. Princip je vysvětlen v úvodu k lineárnímu regulátoru napětí v kapitole 4.

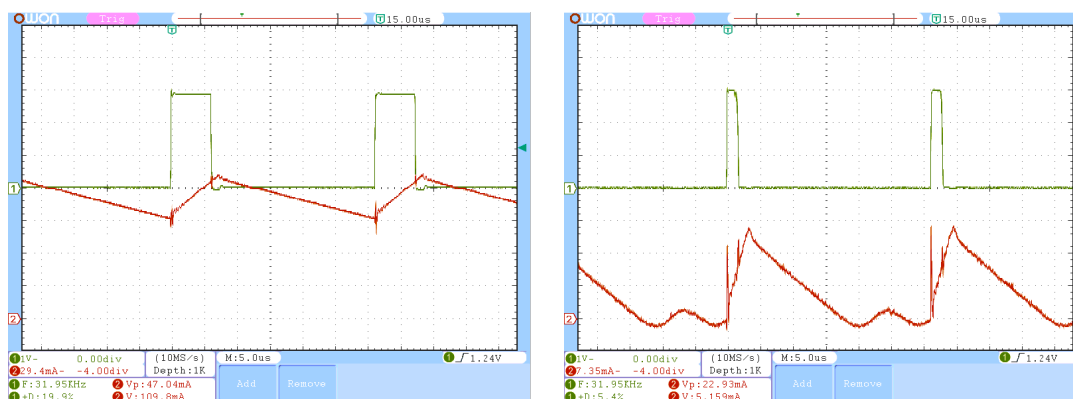
Na obrázku 5 je fotografie zapojení buck konvertoru řízeného mikrořadičem na nepá-

jivém kontaktním poli.



Obrázek 5 Fotografie zapojení buck konvertoru

Na obrázku 6a je zachycen průběh PWM signálu na výstupu MCU (kanál 1) a proudu indukčnosti (kanál 2). Tento průběh je zachycen při výstupním proudu 300 mA, kdy buck konvertor pracuje v CCM (Continuous Conduction Mode) módu. Na obrázku 6b je pak průběh PWM signálu a proud indukčnosti při odlehčení výstupu měniče. V tomto případě je buck konvertor v DCM (Discontinuous Conduction Mode) módu.



a) Průběh řídicího napětí a proudu  $i_L$  v CCM b) Průběh řídicího napětí a proudu  $i_L$  v DCM

Obrázek 6 Průběhy řídicího napětí a proudu  $i_L$  cívky

### 3.2.1 Analýza a návrh obvodových prvků

V této části je provedena podrobnější analýza buck konvertoru. Na obrázku 7 jsou zobrazeny schémata ekvivalentních obvodů v době sepnutí a rozepnutí tranzistoru s vyznačenými obvodovými veličinami.

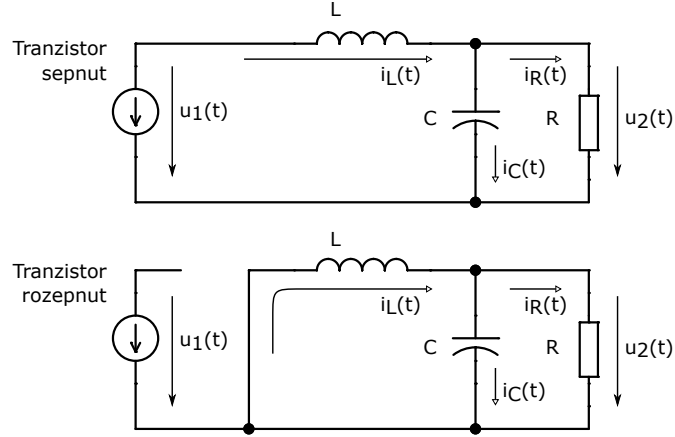
Pokud vyjádříme obvodové veličiny v době sepnutí tranzistoru, dostaneme soustavu dvou lineárních diferenciálních rovnic prvního řádu pro proud cívky  $i_L(t)$  a napětí na kondenzátoru  $u_C(t)$ .

$$u_L(t) = L \frac{di_L(t)}{dt} = u_1(t) - u_C(t) \quad (3.1)$$

$$i_C(t) = C \frac{du_C(t)}{dt} = i_L(t) - \frac{u_C(t)}{R}. \quad (3.2)$$

V době rozepnutí tranzistoru, je konvertor odpojen od vstupního napětí a proud cívky protéká diodou. Cívka je tedy při zanedbání úbytku napětí na diodě připojena





**Obrázek 7** Náhradní schéma v době sepnutí a rozeznutí tranzistoru

mezi svorky kondenzátoru, napětí na cívce je rovno výstupnímu napětí. Na základě toho dostáváme rovnice:

$$u_L(t) = L \frac{di_L(t)}{dt} = -u_C(t) \quad (3.3)$$

$$i_C(t) = C \frac{du_C(t)}{dt} = i_L(t) - \frac{u_C(t)}{R}. \quad (3.4)$$

Výstupní napětí můžeme vyjádřit jako součet střední hodnoty a přidaného zvlnění:

$$u_2(t) = U_2 + \Delta u_2(T). \quad (3.5)$$

Pokud předpokládáme dostatečně velký filtrační kondenzátor, můžeme uvažovat zanedbatelné zvlnění výstupního napětí  $\Delta u_2(T)$ . Výstupní napětí pak lze aproximovat střední hodnotou [20], tedy:

$$u_2(t) \approx U_2 \quad (3.6)$$

Za předpokladu, že měnič pracuje v ustáleném stavu, vyjádříme proud indukčnosti na konci intervalu, kdy je tranzistor sepnut, jako:

$$i_L(t_0 + DT_s) = \frac{1}{L} \int_{t_0}^{t_0 + DT_s} (U_1 - U_2) dt + i_L(t_0) = \frac{1}{L} (U_1 - U_2) DT_s + i_L(t_0), \quad (3.7)$$

kde  $t_0$  je začátek sledovaného intervalu,  $D$  je střída se kterou tranzistor spíná a  $T_s$  je perioda spínání. Stejným postupem určíme proud na konci periody:

$$i_L(t_0 + T_s) = \frac{1}{L} \int_{t_0 + DT_s}^{t_0 + T_s} -U_2 dt + i_L(t_0 + DT_s) = -\frac{1}{L} U_2 (1 - D) T_s + i_L(t_0 + DT_s). \quad (3.8)$$

Dosazením 3.7 do 3.8 dostáváme:

$$i_L(t_0 + T_s) = -\frac{1}{L} U_2 (1 - D) T_s + \frac{1}{L} (U_1 - U_2) DT_s + i_L(t_0). \quad (3.9)$$

Pracuje-li měnič v ustáleném stavu, platí:

$$i_L(t_0 + T_s) = i_L(t_0). \quad (3.10)$$

Řešením 3.9 pak dostáváme známý vztah pro převod mezi vstupním a výstupním napětím u buck konvertoru:

$$U_2 = DU_1. \quad (3.11)$$

Z předešlé rovnice je možné stanovit maximální střihu  $D_{max}$ , které se v praxi volí pro nejmenší přípustné vstupní napětí [26]. V našem případě lze předpokládat konstantní vstupní napětí a proto je volba  $D_{max}$  dána nejvyšším nastavitelným výstupním napětím  $U_{2max}$ . Velikost indukčnosti cívky je zvolena na základě požadované velikosti zvlnění proudu cívkou. To je voleno s ohledem na maximální výstupní proud a tedy i výkon měniče. Pro větší výstupní proudy je zvolen nižší rozkmit proudu cívkou [26]. Na základě průběhů na obrázku 2 můžeme určit celkovou změnu proudu cívkou  $\Delta i_L$  jako:

$$\Delta i_L = i_L(t_0 + DT_s) - i_L(t_0). \quad (3.12)$$

Řešením dostáváme:

$$\Delta i_L = \frac{1}{L} \int_{t_0}^{t_0 + DT_s} (U_1 - U_2) dt + i_L(t_0) - i_L(t_0) = \frac{U_1 - U_2}{L} DT_s. \quad (3.13)$$

Velikost indukčnosti pak můžeme stanovit na základě maximální hodnoty střidy:

$$L = \frac{(U_1 - U_2)}{\Delta i_L} D_{max} T_s = \frac{(U_1 - U_2) U_2}{\Delta i_L U_1} T_s. \quad (3.14)$$

Hodnotu filtračního kondenzátoru je nutno volit na základě požadavku na maximální zvlnění výstupního napětí a uvažovaného zvlnění proudu cívkou [26]. Následující vztah může být použit pro získání minimální kapacity kondenzátoru  $C_1$ :

$$C_{1min} = \frac{\Delta i_L T_s}{8U_2}. \quad (3.15)$$

Volba kondenzátoru se také může opírat o dynamické vlastnosti regulační smyčky. Důvodem je, že odezva výstupního napětí je dána kapacitou a sériovým odporem kondenzátoru. Hodnota kondenzátoru pak může být vybírána s ohledem na maximální velikost překmitu nebo podkmitu výstupního napětí při skokové změně zátěže [26].

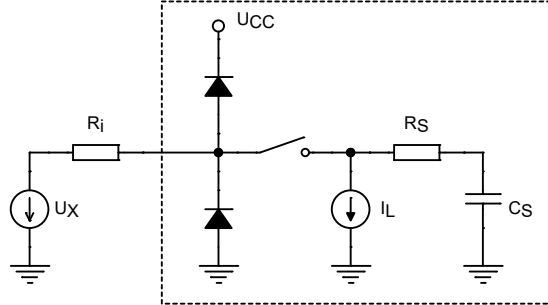
Skutečnosti, které je nutné vzít v potaz při návrhu děliče pro snímání výstupního napětí, jsou probrány v sekci 3.2.2.

### 3.2.2 Snímání výstupního napětí

Výstupní napětí měniče je snímáno odporovým děličem. Odporový dělič je nastaven tak, aby ve zvoleném pracovním bodě bylo na výstupu děliče přibližně poloviční napětí, než je rozsah vstupního napětí AD převodníku. Tímto je pro řízení zaručen dostatečný dynamický rozsah na vstupu převodníku. Dále tento dělič poskytuje ochranný odpor pro omezovač napětí na vstupu MCU. Ten je použit k dodatečné ochraně vstupu MCU, pokud by výstupní napětí měniče přesáhlo maximální stanovené napětí. Omezovač je nastaven tak, aby napětí v takovém případě na pinu mikrořadiče nepřekročilo maximální dovolené napětí, které je dle [2] 4V. Pokud se napětí na výstupu děliče nachází v pracovní oblasti, protéká Zenerovou diodou minimální závěrný proud, který je pro funkci regulační smyčky zanedbatelný.

Dále je nutné uvažovat, že po překročení napájecího napětí AD převodníku, které je rovno napájecímu napětí MCU,  $U_{DD} = 3,3V$ , je výstup převodníku saturován na maximální hodnotu nastaveného bitového rozsahu.

MCU je vybaveno interním vzorkovacím obvodem, který v době sledování nabíjí vnitřní kapacitu. Vzorkovací obvod je po uplynutí doby sledování odpojen od měřeného napětí a zůstává tak po dobu převodu a do dalšího spuštění převodu. Pokud nahradíme měřený obvod vůči svorkám AD převodníku podle Théveninova teorému [2], získáme ekvivalentní schéma zobrazené na obrázku 8.



**Obrázek 8** Ekvivalentní schéma vstupního obvodu AD převodníku

Vzorkovací kapacita je nabíjena přes vnitřní sériový odpor a impedanci měřeného obvodu. Pokud zanedbáme vnitřní svodový proud a další parazitní vlivy jako injektovaný proud z okolních pinů, lze schéma na obrázku 8 popsat, za předpokladu konstantního měřeného napětí, lineární diferenciální rovnicí prvního řádu:

$$(R_S + R_i)C \frac{du_C(t)}{dt} + u_C(t) = U_X, \quad (3.16)$$

kde  $R_i$  je vnitřní odpor měřeného obvodu po náhradě podle Théveninova teorému,  $R_S$  je sériový odpor vzorkovacího obvodu,  $U_X$  je měřené napětí a  $C_S$  je vzorkovací kapacita. Řešením této rovnice za předpokladu nulového počátečního napětí na vzorkovací kapacitě dostáváme:

$$u_C(t) = U_X \left( 1 - e^{-\frac{t}{(R_S + R_i)C}} \right). \quad (3.17)$$

Nyní lze vliv vzorkovací kapacity jednoduše převést na statickou chybu vzniklou konečným vzorkovacím časem  $T_S$ :

$$\Delta U = \lim_{t \rightarrow +\infty} u_C(t) - u_C(T_S) = e^{-\frac{T_S}{(R_S + R_i)C}}. \quad (3.18)$$

Pokud budeme požadovat chybu menší než  $\frac{1}{4}LSB$  pak lze psát:

$$e^{-\frac{T_S}{(R_S + R_i)C}} > \frac{1}{4}2^{-n} \quad (3.19)$$

$$-\frac{T_S}{(R_S + R_i)C} > -\ln(2^{n+2}). \quad (3.20)$$

Z této nerovnosti lze pak stanovit požadavek na maximální velikost vnitřního odporu měřeného zdroje pro stanovenou chybu vzniklou nabíjením vzorkovací kapacity [2]:

$$R_i < \frac{T_S}{C_S \ln(2^{n+2})} - R_S, \quad (3.21)$$

Při zanedbání vnitřního odporu samotného buck konvertoru, který je mnohonásobně menší než vnitřní odpor vzniklý děličem dostáváme:

$$R_i = \frac{R_3 R_2}{R_3 + R_2} < \frac{T_s}{C_S \ln(2^{N+2})}. \quad (3.22)$$

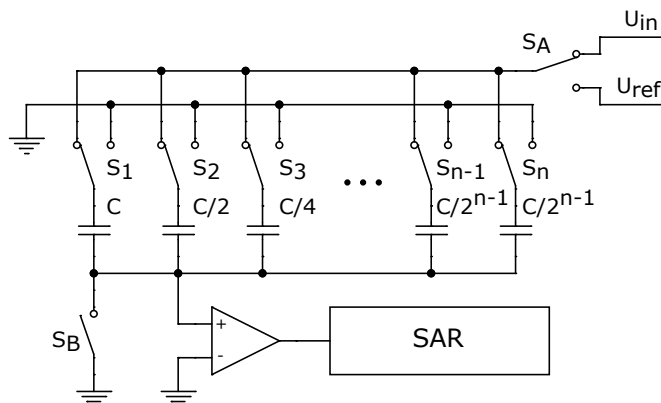
### 3.3 Využití periferie mikrořadiče

Jak již bylo zmíněno, pro měření napětí je použit vestavěný AD převodník mikrořadiče. Pro generování PWM signálu, kterým je řízen výkonový MOSFET tranzistor, je použit hardwarový čítač. Obě zmíněné periferie budou představeny a jejich konfigurace vysvětlena v následujících sekcích.

#### 3.3.1 AD převodníky mikrořadičů řady STM32

Mikrořadič STM32F042 je vybaven interním AD převodníkem s postupnou aproximací založeném na principu nábojové redistribuce. K dispozici je 16 externích multiplexovaných kanálů a 3 interní v případě MCU STM32F042. U tohoto MCU slouží interní kanály pro měření referenčního napětí tohoto převodníku, bateriového napětí a napětí interního teplotního senzoru [1]. V případě MCU STM32L072 jsou k dispozici jen 2 interní kanály pro měření referenčního napětí a napětí teplotního senzoru [3]. Pro konkrétní pouzdra nemusí být všechny externí kanály k dispozici [2], [4].

Principiální schéma AD převodníku s postupnou aproximací pracujícího na principu nábojové redistribuce je na obrázku 9.



**Obrázek 9** Principiální schéma AD převodníku s postupnou aproximací

V době sledování je přepínač  $S_A$  přepnut na vstupní napětí, spínač  $S_B$  je sepnut a přepínače  $S_1$  až  $S_n$  jsou přepnuty tak jak je znázorněno na obrázku 9. V této fázi jsou všechny kapacity o celkové velikosti  $2C$  nabíjeny na vstupní napětí. V době pamatování je spínač  $S_B$  rozpojen a přepínač  $S_A$  je přepnut na referenční napětí. V důsledku toho je na vstupu komparátoru drženo záporné předpětí o velikosti  $U_{Cin} = -U_{in}$ . Poté je spínač  $S_1$  přepnut na zem. Tímto je vytvořen kapacitní dělič, díky kterému je na vstupu komparátoru napětí  $U_{Cin} = -U_{in} + \frac{U_{ref}}{2}$ . Pokud je velikost tohoto napětí menší než 0V, je na výstupu komparátoru logická jednička a přepínač  $S_1$  je do příští konverze ponechán v této poloze. V opačném případě, je-li napětí  $U_{Cin}$  větší než nula, je na výstupu komparátoru logická nula a přepínač  $S_1$  je přepnut zpět na zem. Tímto je získán nejvýznamnější bit datového slova. V dalším kroku je přepnut na referenci přepínač  $S_2$ . Pokud je ve stejné poloze i přepínač  $S_1$  je nové napětí na komparátoru  $U_{Cin} =$

$-U_{in} + \frac{3U_{ref}}{4}$ . V opačném případě je napětí komparátoru  $U_{Cin} = -U_{in} + \frac{1U_{ref}}{4}$ . Tímto je určen další bit datového slova a napětí kapacitního děliče postupně aproximuje vstupní napětí [10].

Délka převodu AD převodníku s postupnou aproximací je tedy závislá nejen na hodinovém kmitočtu AD převodníku, ale také na zvoleném rozlišení [2].

$$T_s = t_s + t_c + t_w = \frac{n_s + n_{SAR}}{f_{CLK}} + t_w, \quad (3.23)$$

kde  $f_{CLK}$  je hodinový kmitočet převodníku,  $n_s$  je počet hodinových cyklů, během kterých AD převodník vzorkuje vstupní napětí viz 3.2.2,  $n_{SAR}$  je počet cyklů potřebných pro převod a  $t_w$  je zpoždění mezi dokončením převodu a zpřístupněním dat v datovém registru AD převodníku. Tento čas je kritický například právě v řízení buck konvertoru, kdy celkový čas převodu nesmí přesáhnout délku periody PWM signálu, kterým je konvertor řízen.

V případě MCU STM32F042 je AD převodník vybaven mechanismem pro automatickou kalibraci. Kalibrace převodníku může být spuštěna uživatelským programem například při spuštění aplikace před použitím AD převodníku. Výsledek kalibrace je poté uložen v kalibračním registru a navíc poskytnut v datovém registru převodníku, ve kterém jsou běžně poskytována převedená data. V případě demonstrační aplikace regulátoru napětí je automatické kalibrace využito.

### 3.3.2 Hardwarové čítače mikrořadiče řady STM32

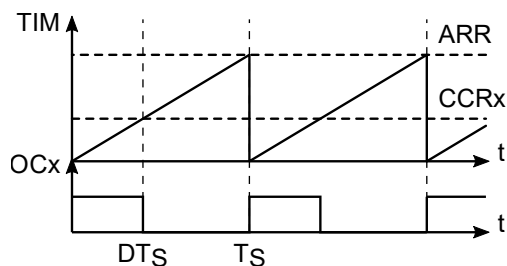
Pro generování PWM signálu, kterým je řízen výkonový tranzistor, je použit hardwarový čítač MCU. Výhodou použití hardwarového čítače je nezávislost generování signálu na uživatelském programu. Mikrořadič STM32F042 je vybaven dvěma základními čítači, jedním rozšířeným čítačem a až šesti standardními čítači. Některé čítače nejsou k dispozici v použité variantě tohoto MCU. V případě mikrořadičů STM32L072 jsou dostupné dva základní a čtyři standardní čítače.

Standardní čítač je vybaven až čtyřmi vstupně/výstupními kanály, které mohou být nakonfigurovány pro generování nebo odchyťování událostí na příslušném pinu. Ke každému kanálu čítače přísluší jeden registr CCR (Capture/Compare Registr). Ve výstupním režimu je hodnota tohoto registru porovnávána číslicovým komparátorem s aktuální hodnotou čítače a na základě výsledku porovnání je generována událost na výstupu kanálu. Tato událost poté může být předána na výstupní pin MCU, nebo může být zpracována interně. V případě použití kanálu pro odchyťování externích událostí, je v registru příslušného kanálu uložena hodnota čítače při zachycení externí události. Externí události mohou být generovány interně a nebo na pinu MCU [1].

Čítače jsou také obecně vybaveny předděličkou PSC (Prescaler), která umožňuje dělení kmitočtu, kterým je čítač taktován. Dále čítač obsahuje registr ARR (Auto Reload Registr) pro předvolbu maximální hodnoty. Pokud je čítač nakonfigurován pro vzestupné čítání, jedná se o hodnotu, při které dochází k přetečení. Při sestupném čítání je tato hodnota výchozí hodnotou čítače při podtečení.

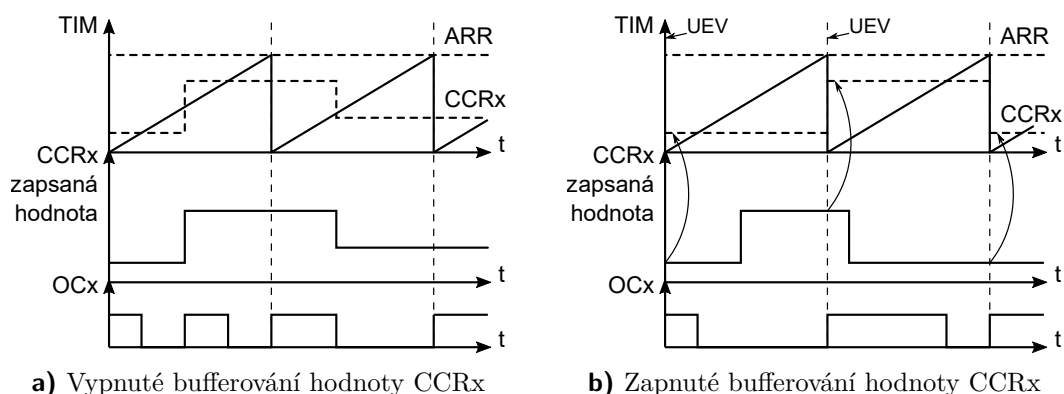
V této úloze je použit kanál 2 čítače TIM2, který je nastaven v režimu generování PWM signálu. Na obrázku 10 je zobrazen princip generování PWM signálu za pomoci čítače, kde  $D$  označuje střídu a  $T_S$  periodu generovaného signálu.

Hodnota registru čítače CNT (Counter) je porovnávána s hodnotou registru CCR2, který přísluší druhému kanálu. V případě demonstrační úlohy je použita konfigurace, ve které je výstupní pin kanálu držen v logické jedničce tak dlouho, dokud je hodnota registru čítače menší než hodnota uložená v registru kanálu. Pokud je hodnota čítače



**Obrázek 10** Princip generování PWM signálu s využitím čítače MCU

rovna nebo větší než CCR2, je výstupní pin nastaven do logické nuly. V základním nastavení však může dojít k chybě generovaného PWM signálu, pokud je do CCR registru zapisováno v průběhu periody PWM signálu. Na obrázku 11a je zobrazen průběh při nevhodné změně hodnoty CCR registru.



a) Vypnuté bufferování hodnoty CCRx

b) Zapnuté bufferování hodnoty CCRx

**Obrázek 11** Ukázka průběhů generování PWM a zápisu do CCRx během periody

Z toho důvodu je v konfiguraci pro kanál, který je použit ke generování PWM, nastaveno bufferování hodnoty CCR registru. Díky tomu je hodnota tohoto registru nejprve ukládána do pomocného registru. Teprve při události UEV, tedy při přetečení čítače, jsou hodnoty uloženy do platných registrů. V důsledku toho nemůže dojít k nechtěné změně střídy během periody PWM signálu. Průběh generování PWM při zapnutém bufferování hodnoty CCR registru je naznačen na obrázku 11b.

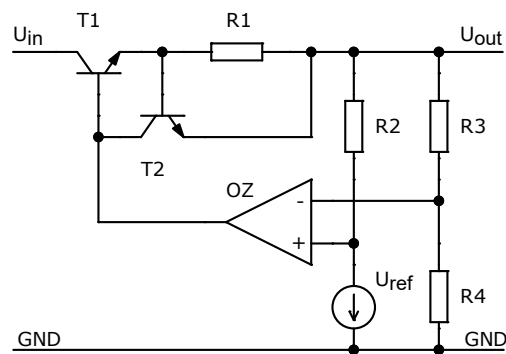
V demonstračním software pro spínaný regulátor napětí je čítač nakonfigurován v režimu master pro generování TRGO signálu, který je použit pro spouštění převodu AD převodníku. Signál TRGO je generován při události, která nastává při shodě hodnoty registru čítače a hodnoty uložené v registru kanálu. Událost tedy nastává v okamžiku rozepnutí tranzistoru a současně je zahájeno vzorkování na AD převodníku. Tímto je docíleno synchronizace AD převodníku a odběru vzorku výstupního napětí s PWM signálem řídicím tranzistor.

## 4 Lineární regulátor napětí

Cílem této úlohy je realizovat číslicovou obdobu lineárního regulátoru napětí. Regulátor je realizován po vzoru funkce klasického lineárního stabilizátoru napětí, tedy s využitím tranzistoru jako regulačního prvku. Jedná se tedy o ztrátovou regulaci. Výstupní napětí je však měřeno mikrořadičem a akční zásah je generován pomocí interního DA převodníku MCU. Akční zásah je přepočítáván a generován v diskretních krocích, jedná se tedy o diskretní řízení.

### 4.1 Princip lineárního stabilizátoru napětí

Na obrázku 12 je zobrazeno principiální schéma lineárního, tří-svorkového stabilizátoru napětí.



Obrázek 12 Principiální schéma lineárního stabilizátoru napětí

Princip činnosti lineárního regulátoru napětí spočívá v řízení toku proudu do zátěže přes aktivní prvek, kterým zpravidla bývá bipolární tranzistor. Regulátor se v tomto případě snaží nastavit takový proud tranzistorem, aby na zátěži byl konstantní úbytek napětí nastavený referencí. Výstupní napětí je snímáno děličem a je porovnáváno s referenčním napětím. Rozdíl těchto napětí je zesílen diferenčním zesilovačem s velkým zesílením. Výstup zesilovače pak ovládá hlavní tranzistor, který je takto řízen v aktivní části jeho výstupní VA charakteristiky. V důsledku průchodu proudu tranzistorem na něm vzniká úbytek napětí a tedy i ztrátový výkon, který je daný rozdílem vstupního a výstupního napětí a protékajícím proudem do zátěže:

$$P_{loss} = (U_{in} - U_{out}) I_{out}. \quad (4.1)$$

Takovéto stabilizátory napětí v sobě integrují jednoduchou ale účinnou ochranu proti přetížení. Ta je realizována odporem  $R_1$  a tranzistorem  $T_2$ . Pokud průchodem proudu do zátěže vznikne na odporu  $R_1$  úbytek napětí vyšší než prahové napětí tranzistoru, tranzistor  $T_2$  se začne otevírat a tím přemostí přechod báze-emitor tranzistoru  $T_1$ . Díky tomu dojde k omezení výstupního proudu.

Výhodou lineárních stabilizátorů napětí je jejich rychlá odezva na změnu zátěže a snadné použití. Další velkou výhodou lineárních stabilizátorů je velmi nízké zvlnění

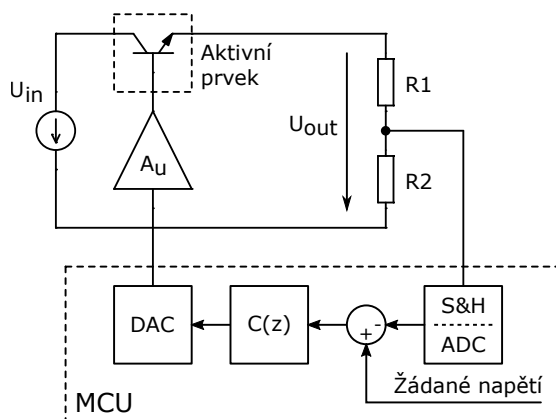
výstupního napětí při zvlněném vstupním napětí. Tuto vlastnost je možné popsat činitelem stabilizace, který je definován jako:

$$k_u = \frac{\frac{\Delta U_{in}}{U_{in}}}{\frac{\Delta U_{out}}{U_{out}}}, \quad (4.2)$$

kde  $\Delta U_{in}$  je velikost zvlnění vstupního a  $\Delta U_{out}$  je velikost zvlnění výstupního napětí,  $U_{in}$  a  $U_{out}$  jsou střední hodnoty vstupního a výstupního napětí. Hodnota činitele stabilizace je součástí katalogových hodnot lineárních regulátorů a pohybuje se v řádu jednotek tisíců. Hodnoty činitele stabilizace jsou často udávány v logaritmickém měřítku. Díky těmto vlastnostem jsou lineární stabilizátory stále používány v aplikacích, ve kterých jsou kladeny vysoké nároky na stabilitu a nízké zvlnění napájecího napětí. Příkladem takovýchto aplikací je například audiotechnika nebo měřicí technika.

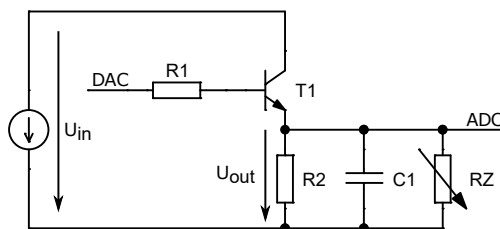
## 4.2 Realizace regulátoru

Na obrázku 13 je zobrazeno blokové schéma regulační smyčky. Výstupní napětí je snímáno odporovým děličem. Blok S&H/ADC představuje vzorkovací obvod a AD převodník MCU. V další části je realizován výpočet odchylky, která je přivedena do bloku  $C(z)$  pro výpočet akčního zásahu diskretního regulátoru. Přenosová funkce regulátoru a výpočet akčního zásahu jsou probrány v kapitole 8. Akční zásah je generován DA převodníkem MCU.



**Obrázek 13** Blokové schéma regulační smyčky pro lineární regulátor napětí

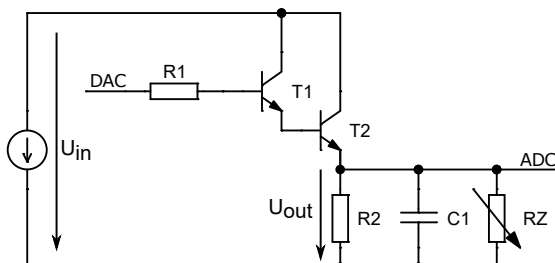
Pro realizaci této úlohy byl zvolen obvod STM32L072, který je vybaven digitálně-analogovým převodníkem. Jednou z nejjednodušších možností realizace této úlohy je použití emitorového sledovače s jedním bipolárním tranzistorem. Schéma zapojení je zobrazeno na obrázku 14.



**Obrázek 14** Schéma zapojení aktivního prvku pro nejjednodušší variantu



Odpor  $R_Z$  představuje proměnnou zátěž. Tranzistor je zapojen se společným kolektorem. Napěťové zesílení emitorového sledovače je přibližně jedna, díky čemuž nebude výstupní napětí vyšší než napětí, kterým je zesilovač buzen. Toto zapojení však disponuje velkým proudovým zesílením a nízkým výstupním odporem. Při použití tohoto zesilovače lze přidáním dalšího tranzistoru do kaskády docílit ještě nižšího výstupního odporu, a především většího proudového zesílení. Toto zapojení je zobrazeno na obrázku 15.

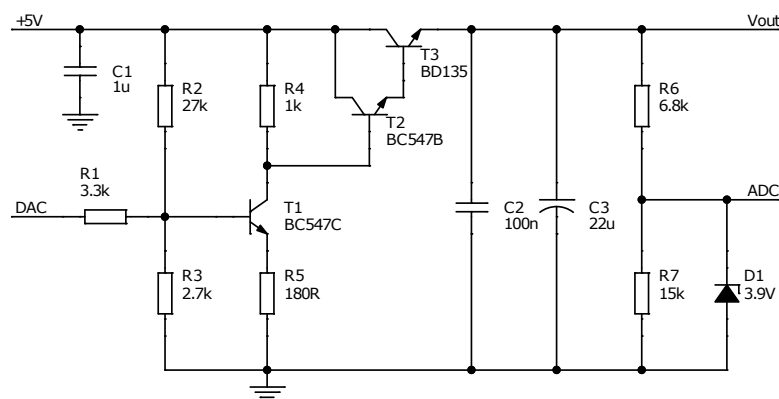


**Obrázek 15** Schéma zapojení pro vyšší výstupní proud

Dvojice výstupních tranzistorů je v takzvaném Darlingtonově zapojení. Tento zesilovač disponuje velkým proudovým zesílením, teoreticky až násobek proudových zisků obou tranzistorů. Tranzistory jsou opět zapojeny jako emitorový sledovač, který má napěťové zesílení blízké jedné a velké proudové zesílení. Další výhodou tohoto zapojení je velmi nízký výstupní a velký vstupní odpor. Maximální výstupní napětí je však omezeno úbytkem napětí na obou přechodech báze emitor  $U_{BE}$  tranzistorů  $T_1$  a  $T_2$ . Toto napětí navíc kolísá v závislosti na kolektorovém proudu. Tyto změny je však možné zanedbat. Nevýhodou emitorového sledovače je, že při řízení DA převodníkem z MCU nelze využít plně celý pracovní rozsah daný napájecím napětím emitorového sledovače. Maximální výstupní napětí  $U_{2max}$  při maximálním vstupním napětí  $U_{1max}$  bude pak:

$$U_{2max} = U_{1max} - 2U_{BE}. \quad (4.3)$$

Z tohoto důvodu je výstup MCU posílen tranzistorovým zesilovačem. Schéma zapojení samotné analogové části je zobrazeno na obrázku 16. Celkové schéma zapojení s mikrořadičem je na obrázku 49 v příloze A.



**Obrázek 16** Schéma zapojení analogové části pro realizaci regulační úlohy

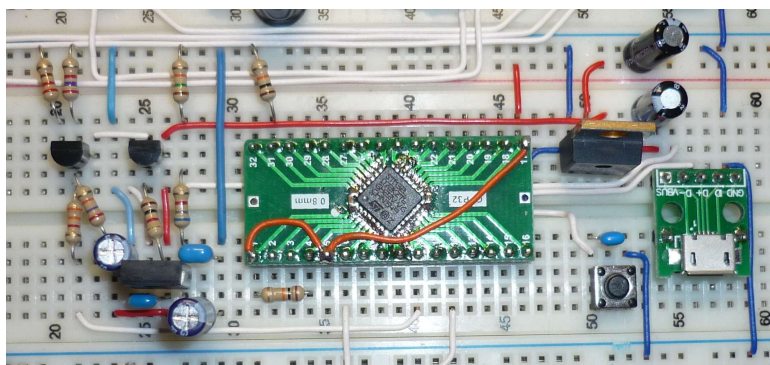
Tranzistorový zesilovač je zapojen se společným emitorem, a je využito plného můstkového zapojení. Toto zapojení disponuje zesílením, které je v případě velkého proudového zesílení tranzistoru přibližně dáno poměrem kolektorového odporu  $R_4$  a emitorového

#### 4 Lineární regulátor napětí

vého odporu  $R_5$ . Emitorový odpor  $R_5$  zavádí zesilovači zápornou zpětnou vazbu, která umožňuje snadnou teplotní stabilizaci pracovního bodu tranzistoru a umožňuje dosažení nízkého zesílení. Velikost zesílení v případě navrhovaného zesilovače se pohybuje v řádu jednotek. Další vlastností tohoto zapojení je relativně nízký vstupní a vysoký výstupní odpor takového zesilovače, jejichž hodnoty se pohybují v řádu jednotek kilohmů. Vysoký výstupní odpor nehraje příliš velkou roli právě díky zapojení výstupních tranzistorů se společným kolektorem. Nižšího vstupního odporu předzesilovače je využito při návrhu zesilovače pro další nastavení napěťového zesílení předřadným odporem  $R_1$ .

Zapojení na obrázku 16 je navrženo tak, aby výstup DA převodníku svým rozsahem pokryl téměř celý pracovní rozsah výkonového tranzistoru, a tím i výstupního napětí. Všechny dříve zmíněné varianty zapojení zobrazené na obrázcích 14 a 15 jsou přípustné, avšak je nutné brát v potaz zmíněná omezení vyplývající z těchto zapojení.

Na obrázku 17 je zobrazena fotografie zapojení “lineárního” regulátoru napětí na nepájivém kontaktním poli.



Obrázek 17 Fotografie zapojení “lineárního” regulátoru napětí

Na obrázku 18a je zobrazen průběh akčního zásahu (kanál 1), kterým je napětí  $U_{DAC}$ , a výstupního napětí (kanál 2) při skokové změně požadované hodnoty. Na obrázku 18b je pak odezva regulátoru při skokové změně zátěže.



- a) Průběh napětí  $U_{DAC}$  a výstupního napětí při skoku reference      b) Průběh napětí  $U_{DAC}$  a výstupního napětí při poruše

Obrázek 18 Průběhy řídicího napětí  $U_{DAC}$  a výstupního napětí “lineárního” regulátoru napětí

### 4.2.1 DA převodník v mikrořadiči STM32L072

Použitý MCU STM32L072 disponuje vestavěným digitálně-analogovým převodníkem. Tento DA převodník obsahuje dva nezávislé výstupní kanály. Tyto kanály mohou být mezi sebou synchronizovány, díky čemu je možné generování například diferenčního signálu. DA převodník implementuje již v hardwaru generování šumu nebo trojúhelníkového průběhu [3]. Oba kanály převodníku umožňují použití výstupního bufferu pro buzení větší zátěže. V základním režimu je vnitřní impedance DA převodníku relativně vysoká, dle [4] až  $20\text{ k}\Omega$ . Při použití výstupního bufferu je výstupní impedance značně nižší. Nevýhodou je ale například vyšší multiplikační chyba nebo významnější teplotní drift.

Při použití v úloze lineární regulátor napětí, je výstup DA převodníku veden na zesilovač, jehož vstupní impedance je v řádu jednotek kilo ohmů, více v sekci 4.2.2. Z toho důvodu je DA převodník výhradně používán se zapnutým výstupním bufferem. Díky tomu je možné zesilovač budit přímo výstupem převodníku bez nutnosti dalšího impedančního oddělení.

### 4.2.2 Návrh obvodových prvků

V této části je naznačen postup návrhu obvodových prvků pro realizaci této úlohy na základě kompletního zapojení zobrazeného na obrázku 16. Analyzujeme-li jen samotný zesilovač zobrazený ve schématu na obrázku 16 bez výstupního emitorového sledovače, můžeme sestavit následující obvodové rovnice:

$$R_4 I_C + U_{CE} + R_5 (I_C + I_B) - U_{CC} = 0 \quad (4.4)$$

$$R_2 (I_B + I_D) + U_{BE} + R_3 (I_C + I_B) - U_{CC} = 0 \quad (4.5)$$

$$(R_1 \parallel R_3) I_D - U_{BE} - R_E (I_C + I_B) = 0, \quad (4.6)$$

kde  $I_B$  je proud do báze ve zvoleném pracovním bodě,  $I_D$  je proud děličem v bázi,  $U_{CE}$  je napětí mezi kolektorem a emitorem,  $U_{BE}$  je napětí mezi bází a emitorem v pracovním bodě.  $U_{CC}$  je napájecí napětí, které je použito z USB, tedy  $U_{CC} = 5V$ .

Na základě těchto rovnic a zvoleném pracovním bodě je možné stanovit odpory  $R_2$ ,  $R_4$  a  $R_5$ . V tomto případě je zesilovač stejnosměrně vázaný a proto je nutné při výpočtu obvodových prvků na tuto skutečnost dbát. Ze schématu je patrné, že napětí v klidovém pracovním bodě bude ovlivňováno napětím DA převodníku. Při návrhu byl pracovní bod zvolen při nulovém vstupním napětí. Tímto je možné v náhradním schématu pro pracovní bod odpor  $R_1$  připojit na zem, čímž vznikne paralelní kombinace  $R_1$  a  $R_3$ . Důvodem volby tohoto pracovního bodu byl snazší odhad kolektorového proudu a napětí mezi kolektorem a emitorem. Hodnoty těchto veličin jsou tímto fixovány v oblasti počátku pracovního rozsahu, zatímco zbytek pracovní oblasti je zajištěn vhodným zesílením. V takto zvoleném pracovním bodě pak bude výstupní napětí zesilovače maximální a bude dané součtem úbytků napětí  $U_{CE}$  a napětí na emitorovém odporu  $R_5$ . Stanovíme-li požadavek na hodnotu maximálního výstupního napětí  $U_2$  v takto zvoleném pracovním bodě, lze při zvoleném minimálním kolektorovém proudu  $I_C$  určit odpor  $R_4$  jako:

$$R_4 = \frac{U_{CC} - U_{2max}}{I_C}. \quad (4.7)$$

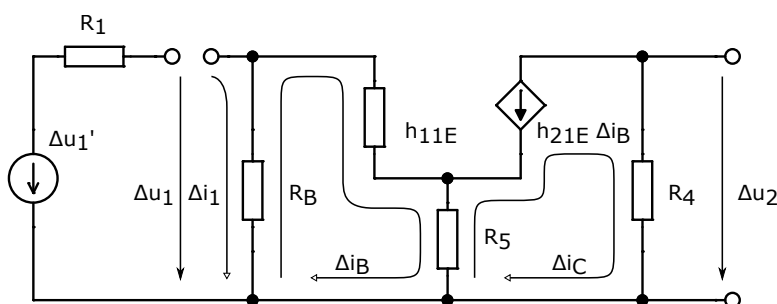
Hodnoty odporu  $R_5$  lze vyjádřit z 4.4:

$$R_5 = \frac{U_{CC} - R_4 I_C - U_{CE}}{I_C + I_B}, \quad (4.8)$$

kde napětí  $U_{CE}$  je zvoleno jako dostatečně velká část výstupního napětí  $U_{2max}$  ve zvoleném pracovním bodě. Další odpor získáme z 4.5:

$$R_2 = \frac{U_{CC} - U_{BE} - R_5 (I_B + I_C)}{I_D + I_B}, \quad (4.9)$$

Proud děliče v bázi  $I_D$  je volen dostatečně velký, tak aby bylo možné uvažovat, že se jedná o tvrdý dělič. Nyní je potřeba stanovit ještě odpory  $R_1$  a  $R_3$ . Z rovnice 4.6 je patrné, že oba odpory budou mít vliv na nastavení pracovního bodu tranzistoru, a proto je nutné již vzít v úvahu zesílení celého zesilovače. Pokud vytvoříme model pro změny signálu, dostaneme ekvivalentní schéma zobrazené na obrázku 19.



**Obrázek 19** Náhradní schéma pro změny signálu

Toto schéma jsme obdrželi odečtením hodnot obvodových veličin od jejich hodnot v pracovním bodě. Při tvorbě náhradního schématu byl zanedbán zpětně převodní přenos  $h_{12E}$ , jehož velikost je velmi malá, v řádech  $10^{-4}$ . Výstupní vodivost  $h_{22E}$  je pro zvolený typ tranzistoru dostatečně nízká a proto byla také zanedbána. Pokud analyzujeme takto vzniklý model metodou smyčkových proudů, dostaneme soustavu dvou rovnic o dvou neznámých:

$$R_B (\Delta i_1 - \Delta i_B) - \Delta u_1 = 0 \quad (4.10)$$

$$R_B (\Delta i_B - \Delta i_1) + h_{11E} \Delta i_B + R_E (\Delta i_B + \Delta i_C) = 0, \quad (4.11)$$

kde  $R_B = R_2 \parallel R_3$ . Dosazením  $\Delta i_C = h_{21E} \Delta i_B$  a řešením soustavy dostaneme:

$$\Delta i_1 = \frac{\Delta u_1 (R_B + h_{11E} + R_E (1 + h_{21E}))}{R_B (h_{11E} + R_E (1 + h_{21E}))} \quad (4.12)$$

$$\Delta i_B = \frac{\Delta u_1}{h_{11E} + R_E (1 + h_{21E})}. \quad (4.13)$$

Nyní můžeme vyjádřit charakteristiky jako  $R_{in}$  a  $H_{Up}$  pro tento zesilovač.

$$R_{in} = \frac{\Delta u_1}{\Delta i_1} = R_B \parallel R_x \quad (4.14)$$

$$H_{Up} = \frac{\Delta u_{2p}}{\Delta u_{1p}} = -\frac{R_C h_{21E}}{R_x}, \quad (4.15)$$

kde pro  $R_x$  platí:

$$R_x = h_{11E} + R_E(1 + h_{21E}). \quad (4.16)$$

Jak je vidět z obrázku 19, toto zesílení platí pouze pro samotný zesilovač bez odporu  $R_1$ . Pokud zanedbáme vnitřní odpor DA převodníku, více v sekci 4.2.1, je pak vstupní napětí zesilovače, které je zesíleno na výstup, dáno napětovým děličem tvořeným sériovým odporem  $R_1$  a vstupním odporem  $R_{in}$  zesilovače. Pro výstupní napětí pak platí:

$$\Delta u_2 = \Delta u_1 \frac{R_{in}}{R_{in} + R_1}. \quad (4.17)$$

Na základě toho je celkové napětové zesílení na prázdko  $H'_{Up}$  dáno vztahem:

$$H'_{Up} = \frac{\Delta u_2}{\Delta u_1} = H_{Up} \frac{R_{in}}{R_{in} + R_1} \quad (4.18)$$

Tímto jsme obdržely druhou rovnici popisující závislost zesílení pro změny signálu nacházející se v lineární oblasti na velikosti vnitřního odporu zesilovače a sériového odporu  $R_1$ . Z rovnice 4.18 nyní vyjádříme odpor  $R_1$ :

$$R_1 = R_{in} \frac{H_{Up} - H'_{Up}}{H'_{Up}}. \quad (4.19)$$

V tuto chvíli lze  $R_1$  dosadit do rovnice 4.6 a vyjádřit hodnotu  $R_3$ . Při bližším prozkoumání obou výrazů je jasné, že takto obdržíme celkem složitý výraz se spoustou složených lomených výrazů. Proto se jeví jako výhodnější převrátit rovnici 4.6 a řešit rovnici pro vodivosti.

$$\frac{1}{R_1} + \frac{1}{R_3} = \frac{I_D}{R_E(I_B + I_C) + U_{BE}} = G_x. \quad (4.20)$$

Dosažením 4.19 do 4.20 dostáváme:

$$\frac{1}{R_{in}} \frac{H'_{Up}}{H_{Up} - H'_{Up}} + \frac{1}{R_3} = G_x. \quad (4.21)$$

Pro vstupní odpor zesilovače pak dostáváme:

$$\frac{1}{R_{in}} = \frac{1}{R_B} + \frac{1}{R_x} = \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_x}. \quad (4.22)$$

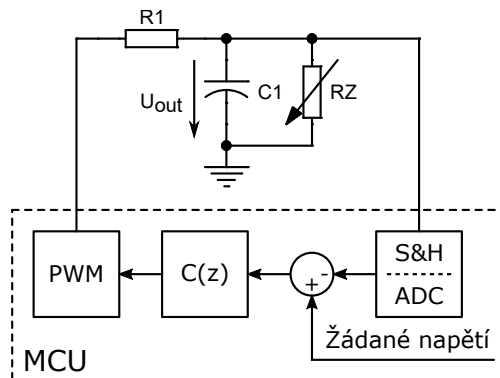
Dosažením a řešením pro  $R_3$  dostáváme:

$$\frac{1}{R_3} = \frac{G_x - \frac{H'_{Up}}{H_{Up} - H'_{Up}} \left( \frac{1}{R_2} + \frac{1}{R_x} \right)}{\frac{H'_{Up}}{H_{Up} - H'_{Up}} + 1}. \quad (4.23)$$

Tímto jsme schopni stanovit hodnotu odporu  $R_3$ , neboť ve vztahu figurují jen známé a již dopočítané, nebo zvolené hodnoty. Hodnotu odporu  $R_1$  jsme pak schopni určit na základě námi stanoveného celkového zesílení  $H'_{Up}$  pomocí vztahu 4.19.

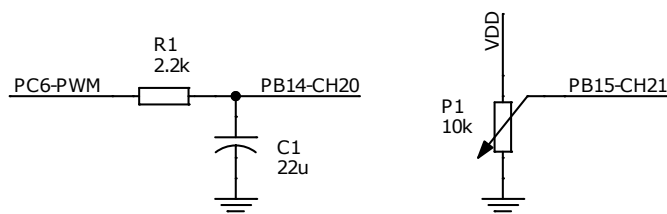
### 4.3 Regulátor napětí s integračním RC článkem

Pro účely předmětu Mikroprocesory a mikrořadiče v elektronice byl navržen software v assembleru pro regulátor napětí s integračním RC článkem. Tato regulační úloha byla již použita ve výuce. Na obrázku 20 je zobrazeno blokové schéma regulátoru napětí s dolnopropustním filtrem.



**Obrázek 20** Blokové schéma regulátoru napětí s integračním RC článkem

Regulátor je realizován jednoduchým RC článkem zapojeným jako integrační článek. Akčním zásahem je střída PWM signálu, který je generován pomocí hardwarových čítačů. Napětí na výstupu RC článku je přímo snímáno interním AD převodníkem. Poté je vypočítána odchylka od požadované hodnoty, která je použita k výpočtu akčního zásahu regulátoru. Požadovaná hodnota výstupního napětí je nastavována potenciometrem, jehož napětí je měřené na druhém kanálu AD převodníku. Tato regulační úloha je navržena pro vývojový kit STM32L100-Discovery. Na obrázku 21 je zobrazeno připojení dolnopropustního filtru k vývojovému kitu.



**Obrázek 21** Schéma připojení RC článku a potenciometru k vývojovému kitu

Hodnota odporu  $R_1$  je volena tak, aby nedošlo k přetížení pinu procesoru při zkratování výstupu filtru. Hodnota kondenzátoru je pak určena na základě frekvence generovaného PWM signálu tak, aby časová konstanta RC článku byla mnohonásobně větší než perioda PWM signálu. Pro časovou konstantu lze psát:

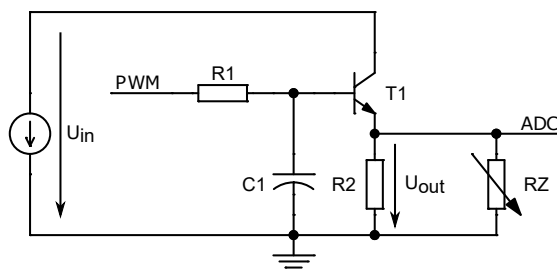
$$\tau = R_1 C_1 = kT_S, \quad (4.24)$$

kde  $T_S$  je perioda PWM signálu a  $k$  je zvoleno v řádu desítek až stovek.

Pro generování PWM signálu je použit první kanál čítače TIM3, který je nastaven v režimu generování PWM, více v 3.3.2.

S dolnopropustním filtrem je možné realizovat lineární regulátor napětí i na mikrořadiči, který nedisponuje interním DA převodníkem. Napětí pro řízení aktivního prvku může být takto generováno pomocí PWM signálu, který je posléze filtrován. Na obrázku 22 je znázorněno schéma s emitorovým sledovačem, který je řízen filtrovaným

PWM signálem. Jedná se o obdobu nejjednodušší varianty lineárního stabilizátoru napětí představené na obrázku 14.



**Obrázek 22** Schéma regulátoru napětí s integračním RC článkem a emitorovým sledovačem

Vstupní napětí  $U_{in}$  je stejné jako napájecí napětí procesoru.

### 4.3.1 AD převodníky na mikrořadičích STM32L100

AD převodníky MCU řady STM32L100RC jsou vybaveny až 42 multiplexovanými kanály. Tento převodník umožňuje jednorázové převody jednotlivých kanálů, jednorázový převod sekvence kanálů, nebo cyklický převod jednoho či více kanálů. Pro konfiguraci více převodů tento převodník obsahuje sekvencer, díky kterému je možné nastavit libovolné sekvence převodů. Kanály tohoto AD převodníku lze přiřadit do dvou tříd, regulární kanály a injektované kanály. Regulární kanály mají jeden společný datový registr. Použití těchto kanálů vyžaduje při softwarovém čtení převedených dat příslušné nastavení, které povolí generování příznaku EOC (End of Conversion) po každém převodu, nikoliv po skončení celé sekvence. Pro čtení dat z těchto kanálů je tedy optimálnější použít DMA kontrolér. Sekvencer regulárních kanálů umožňuje nastavení posloupnosti až 28 kanálů.

Injektované kanály mají každý příslušný datový registr, ve kterém jsou po skončení konverze k dispozici převedená data. Sekvencer injektovaných kanálů umožňuje aktivaci až 4 kanálů. Hlavní vlastností injektovaných kanálů je, že mohou být spuštěny během převodu regulárních kanálů. V takovém případě dochází k přerušování konverze na regulárních kanálech. Ta je odložena do té doby, než jsou převedeny všechny kanály nastavené v sekvenceru pro injektované kanály. Oba sekvencery umožňují téměř libovolnou konfiguraci. Navíc je možné tentýž kanál nastavit jak pro regulární převod, tak pro injektovaný.

Při řešení této demonstrační úlohy, byly nejprve pro měření výstupního napětí RC článku a požadované hodnoty, zadané potenciometrem, použity regulární kanály. Požadavkem na řešení úlohy bylo softwarové čtení dat, které však nebylo možné díky chybě příznaku EOC. Tento příznak je nastavován hardwarem a resetován čtením z datového registru, nebo zápisem logické nuly na bit EOC. Po skončení konverze je tento příznak nastaven, ale není zajištěn, a proto v dalším strojovém cyklu dochází k jeho vynulování. Z důvodu této chyby byly nakonec použity injektované kanály, jejichž nastavení a softwarová obsluha je snazší.

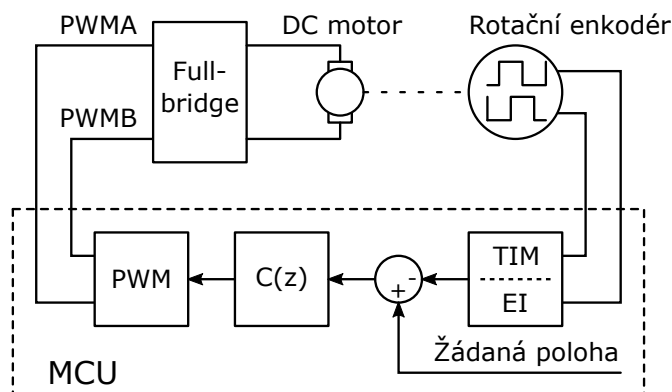
Převod injektovaných kanálů je spuštěn softwarově. Po spuštění se čeká do skončení převodu obou zvolených kanálů. Poté je proveden výpočet akční zásahu diskrétní aproximace PI regulátoru, který je pro jednoduchost počítán jen pomocí celých čísel. Každá iterace je zahájena poté, co je čítačem vygenerován příznak UEV (Update Event). Akční zásah je tedy přepočítáván po skončení periody PWM signálu. Nastavení čítače pro generování PWM je až na použití jiného čítače identické jako v 3.3.2.

## 5 Regulace polohy

Pro realizaci této regulační úlohy byl použit stejnosměrný motor osazený rotačním optickým inkrementálním snímačem polohy. Na hřídeli motoru je připevněné kyvadlo, které je vyvažováno v dané poloze. Jedná se o snadnou úlohu z hlediska regulace, je zde pouze jeden stupeň volnosti, avšak cílem je realizace samotného regulátoru s využitím dostupných prostředků.

### 5.1 Realizace úlohy

Na obrázku 23 je zobrazeno schéma regulační smyčky pro realizaci regulátoru polohy. Kvadrurní signál z rotačního inkrementálního snímače je vyhodnocován čítačem mikrořadiče. Na základě dekódovaného údaje o relativní poloze hřídele motoru je vypočítána odchylka od referenční polohy. Tato odchylka je přivedena do bloku  $C(z)$ , který představuje výpočet akčního zásahu regulátoru. Poté je vypočítaný akční zásah převeden na dvoukanalový PWM signál, kterým je ovládán koncový stupeň pro buzení motoru.



**Obrázek 23** Blokové schéma regulační smyčky pro realizaci regulátoru polohy

Na obrázku 24 je zobrazeno kompletní schéma zapojení pro realizaci této regulační úlohy. Jako řídicí obvod je použit MCU STM32F042F6P6.

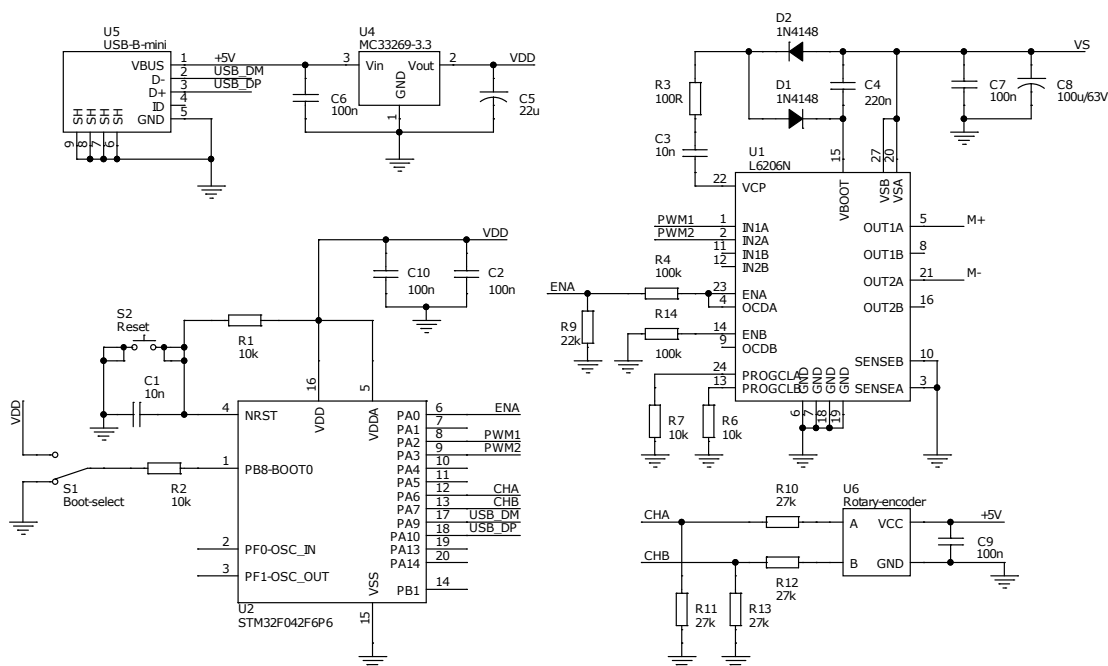
MCU je napájeno přes lineární stabilizátor z USB. USB je využito také pro nahrávání firmwaru do MCU pomocí DFU (Direct Firmware Upgrade) protokolu. Ten je spuštěn připojením logické '1' na pin BOOT0 a restartováním MCU. Poté je MCU přepnuto do režimu bootování z USB. Výhodou je snadné nahrání firmwaru bez nutnosti dalšího programátoru.

Napájecí napětí pro H-můstek, ve schématu označeno jako VS, je v případě použitého motoru 12 V. K napájení H-můstku je použit externí zdroj.

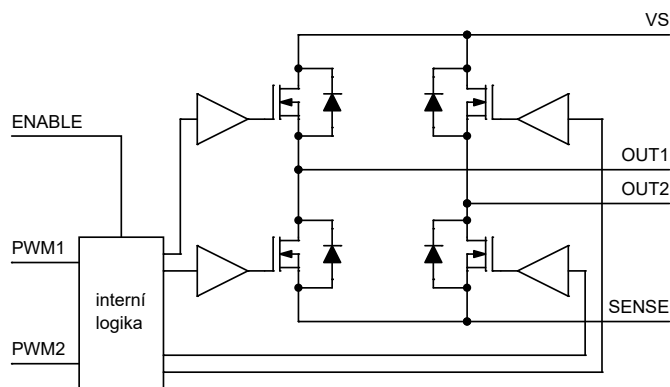
#### 5.1.1 Realizace koncového stupně s obvodem L6206

Pro řízení motoru je použit integrovaný obvod L6206. Na obrázku 25 je zobrazeno zjednodušené blokové schéma plného můstku. Kompletní blokové schéma tohoto obvodu lze najít v [16].





**Obrázek 24** Schéma zapojení regulátoru polohy s obvodem L6206



**Obrázek 25** Zjednodušené blokové schéma jednoho H-můstku obvodu L6206

Obvod L6206 implementuje dva oddělené full-bridge drivery s integrovanou nadproudovou ochranou. Interní nadproudová ochrana je realizována odběrem přesně definovaného zlomku proudu tekoucím výkonovým MOSFET tranzistorem. Tento proud je poté porovnán s vnitřní proudovou referencí, která je programovatelná externím rezistorem. Pokud dojde k překročení maximálního proudu, je automaticky stažen výstup OCD (Overcurrent Detection) do logické nuly.

Každý kanál je vybaven řídicím vstupem pro aktivaci tohoto kanálu. Řídicí PWM signál se přivádí zvlášť pro každý half-bridge. Tento integrovaný obvod v sobě integruje nábojovou pumpu pro řízení horních MOSFET tranzistorů.

S využitím OCD výstupu lze realizovat triviální nadproudovou ochranu již v hardwaru, která při překročení nastaveného proudového limitu deaktivuje H-můstek. Tato ochrana je realizována oddělením řídicího pinu MCU odporem  $R_4$  a propojením OCD výstupu s řídicím vstupem daného H-můstku. Při překročení proudového limitu je řídicí vstup H-můstku stažen na zem a tím dojde k deaktivaci výkonové části. Pokud je vstup pro řízení H-můstku v logické nule, je H-můstek ve stavu vysoké impedance, tedy všechny tranzistory jsou zavřeny.

Výhodou tohoto obvodu je velmi snadné použití a možnost paralelizovat výstupní kanály obvodu pro dosažení vyššího výstupního proudu.

### 5.1.2 Snímání polohy

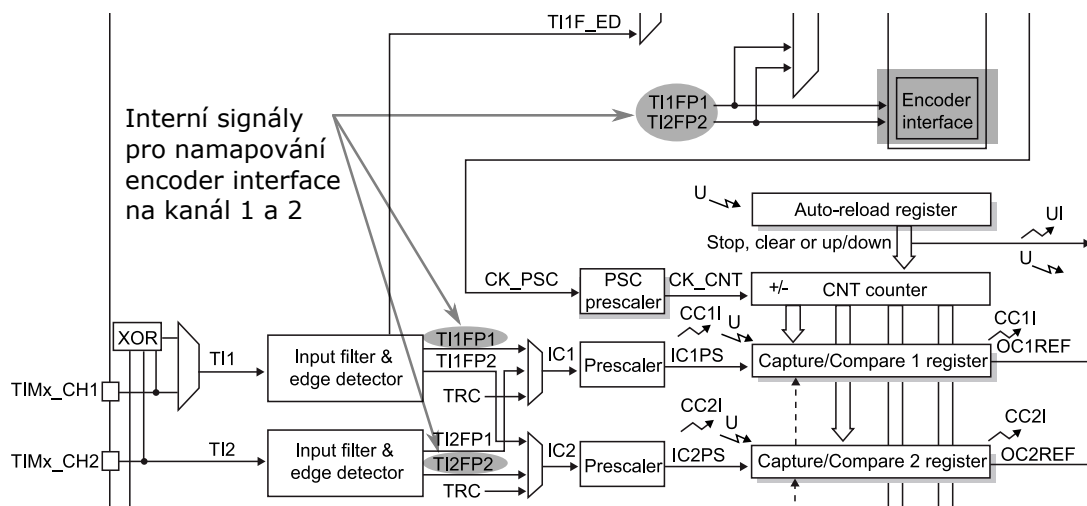
Pro snímání polohy je použit optický inkrementální snímač. Použitý snímač poskytuje kvadrurní signál na dvou kanálech v TTL napěťových úrovních bez referenční značky. Jedná se tedy o čistě relativní údaj o poloze, a proto je regulace polohy kyvadla závislá na nastavení počátku. Kvadrurní signál je zpracováván interním čítačem mikrořadiče. Výstup rotačního inkrementálního snímače je připojen na piny procesoru přes napěťový dělič tvořený odpory  $R_{10}$ ,  $R_{11}$ ,  $R_{12}$  a  $R_{13}$ . Důvodem je, že použité piny procesoru nejsou 5V tolerantní piny. Tyto piny byly ovšem vybrány kvůli možnosti interního namapování na vstupy čítače, který je využit pro dekódování kvadrurního signálu, více v následující sekci.

### 5.1.3 Vyhodnocení kvadrurního signálu

Pro vyhodnocení kvadrurního signálu snímače polohy je využit hardwarový čítač TIM3. Tento čítač umožňuje na prvních dvou kanálech vyhodnocovat kvadrurní signál v různých režimech. Čítač může být nastaven pro čítání na:

- hranu na prvním kanálu
- hranu na druhém kanálu
- obě hrany na obou kanálech

Na obrázku 26 je zobrazeno blokové schéma prvních dvou kanálů použitého čítače [1] s vyznačenými signály pro namapování prvních dvou kanálů na rozhraní dekodéru kvadrurního signálu.



**Obrázek 26** Blokové schéma kanálů čítače použitých jako rozhraní k enkodéru

Čítač je inkrementován při náběžné a dekrementován při sestupné hraně na zvoleném kanálu. Polaritu interního signálu lze navíc ještě invertovat nastavením příslušných bitů v konfiguračním registru, a tím změnit směr čítání. Během činnosti je hardwarově nastavován bit DIR v registru TIMx\_CR1, který udává aktuální směr čítání.

Hodnota v registru TIMx\_CNT pak přímo udává relativní polohu. Maximální hodnota čítače i v tomto režimu může být konfigurována pomocí auto-reload registru. Při

jejím překročení je čítač resetován do nuly, v opačném případě je tato hodnota nastavena jako výchozí při podtečení čítače.

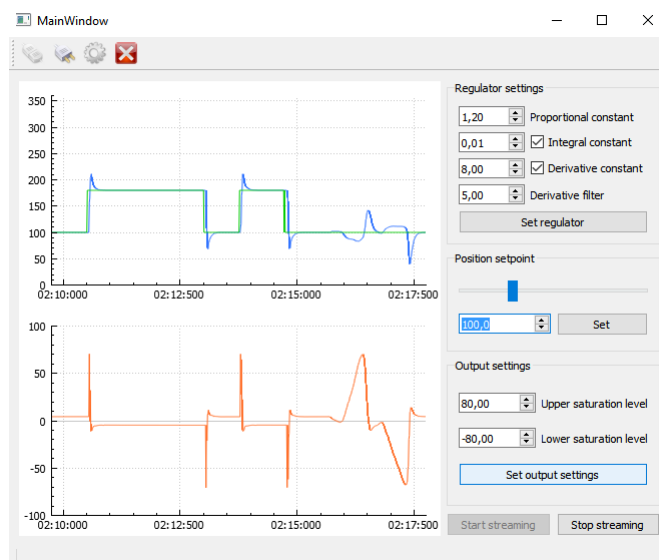
### 5.1.4 Generování PWM signálu

Pro generování PWM signálu musí být použit jiný čítač než pro dekodování kvadraturního signálu. Pro řízení motoru jsou využity dva kanály čítače, z nichž jeden je vždy v režimu, kdy je na výstupu držena logická nula, a druhý je nastaven v režimu generování PWM signálu, viz sekce 3.3.2. Tímto je jeden polomost držen ve stavu, kdy je trvale sepnut dolní tranzistor, a druhý polomost přepíná mezi horním a dolním tranzistorem. V důsledku toho motorem protéká proud, jehož střední hodnota je řízena střídou PWM signálu na druhém polomostu. Pro otáčení motoru v opačném směru je nastavení opačné. Druhý kanál čítače je nastaven v režimu, kdy je na daném výstupu držena logická nula a první kanál je konfigurován pro generování PWM.

## 5.2 Demo s vývojovým kitem IHM04A1

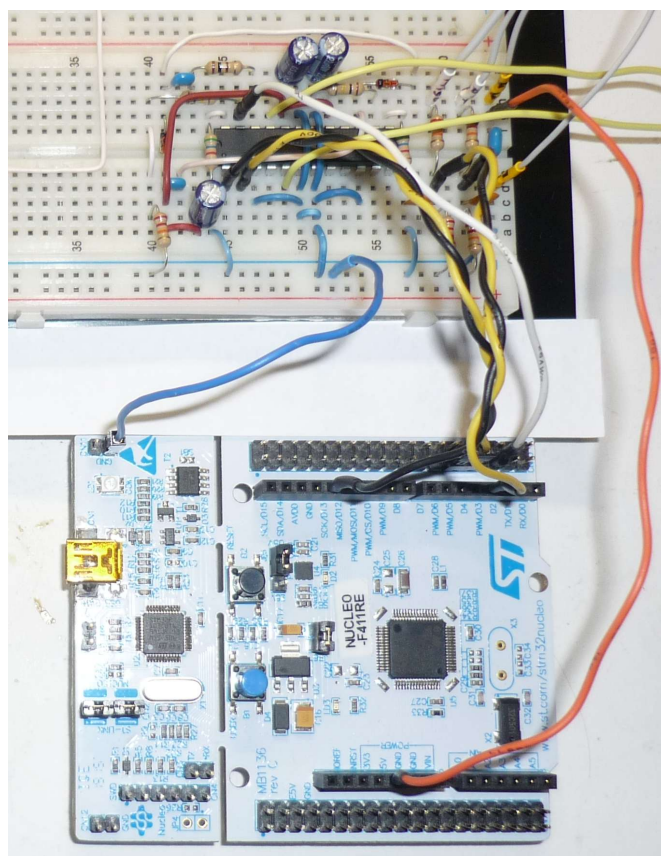
Tato úloha byla odladěna na vývojovém kitu STM32 Nucleo-F411RE a vývojovém modulu IHM04A1, který je osazen právě obvodem L6206. Pro ukázkou byl navržen firmware pro mikrořadič STM32F411, který implementuje PID regulátor s filtrovanou derivační složkou, více v 8.4, a umožňuje nastavení parametrů samotného regulátoru. Součástí je PC aplikace, která je navržena pro komunikaci se zmíněným firmwarem a vytváří uživatelské rozhraní spolu s možností vykreslování dat během regulace, více v kapitole 7.

Na obrázku 27 je zobrazena odezva na skok požadované hodnoty a odezvy při působení poruchy. K záznamu dat byla použita demonstrační PC aplikace pro sběr dat.



**Obrázek 27** Ukázka demonstrační aplikace pro regulátor polohy

Na obrázku 28 je fotografie připojení obvodu L6206 k vývojovému kitu Nucleo. Po pravé straně je přes dělič připojen rotační inkrementální snímač.



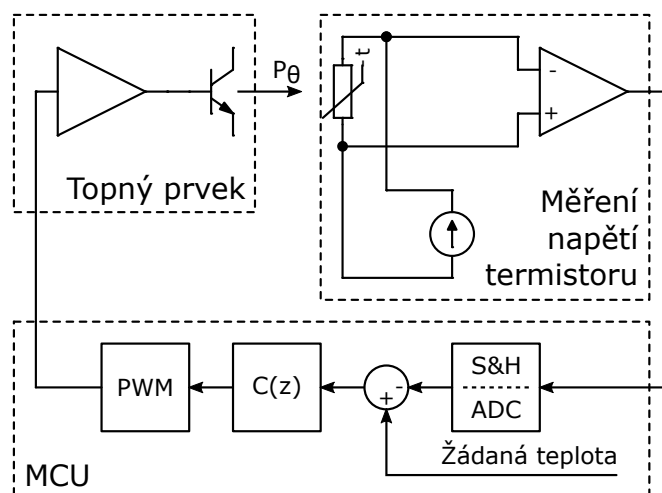
**Obrázek 28** Fotografie připojení obvodu L6206 k vývojovému kitu Nucleo

## 6 Regulace teploty

Účelem této demonstrační úlohy je, aby se studenti seznámily s chováním systémů s pomalejší dynamikou než v případě lineárního regulátoru napětí nebo spínaného zdroje. Úloha je založena na ohřívání výkonového polovodičového prvku na požadovanou teplotu. Ta je nastavována zpětnovazebním regulátorem implementovaným na mikrořadiči. Teplota je snímána pomocí NTC termistoru.

### 6.1 Realizace úlohy

Na obrázku 29 je zobrazeno blokové schéma regulační smyčky pro regulaci teploty. Odpor termistoru je měřen pomocí čtyřvodičové metody. Změřené a zesílené napětí na termistoru je převedeno interním AD převodníkem mikrořadiče. Na základě této hodnoty je stanovena odchylka od požadované teploty. Velikost odchylky je poté přepočítána algoritmem regulátoru na akční zásah. Ten je v tomto případě pomocí interního čítače MCU převeden na střihu PWM signálu, kterým je řízen průtok proudu topným prvkem.



Obrázek 29 Blokové schéma regulační smyčky pro regulaci teploty

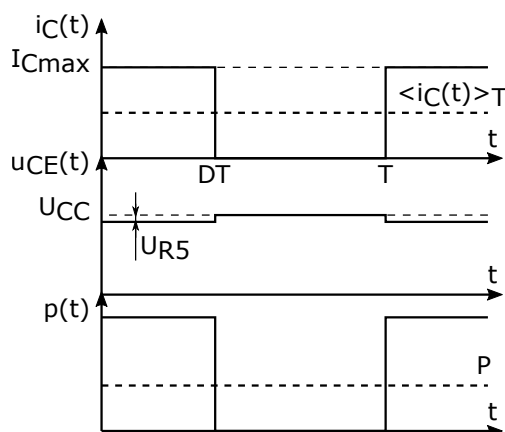
#### 6.1.1 Topný přípravek

Jako topný prvek je použit výkonový bipolární NPN tranzistor, který je provozován výhradně v lineárním režimu. Tento tranzistor má na sobě plně napájecí napětí a je nastaven tak, aby kolektorem tekla konstantní proud. Tímto na tranzistoru vzniká výkonová ztráta daná úbytkem napětí mezi kolektorem a emitorem a střední hodnotou proudu do kolektoru. Proud kolektoru je regulován pulzně. Ztrátový výkon je tedy dán

střídou PWM signálu, kterým je tranzistor zavírán a otevírán:

$$P = \frac{1}{T} \int_{t_0}^{t_0+T} u(t)i(t) dt = \frac{1}{T} \left( \int_{t_0}^{t_0+DT} U_{CE} I_{Cmax} dt + 0 \right) = D U_{CE} I_{Cmax}, \quad (6.1)$$

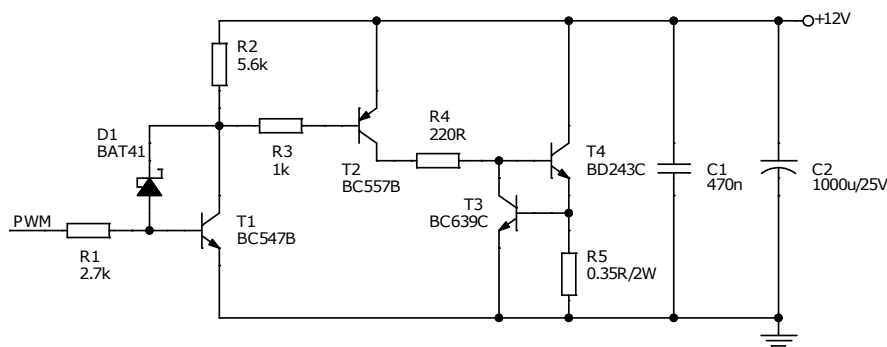
kde  $D$  je střída PWM signálu,  $U_{CE}$  je napětí mezi kolektorem a emitorem tranzistoru v době otevření tranzistoru a  $I_{Cmax}$  je maximální kolektorový proud. Časový průběh kolektorového proudu, napětí mezi kolektorem a emitorem a celkové výkonové ztráty na tranzistoru  $p(t)$  je zobrazen na obrázku 30.



**Obrázek 30** Průběh proudu, napětí a výkonu na topném tranzistoru

Pokles napětí  $u_{CE}(t)$  při otevření tranzistoru je dán průtokem proudu přes stabilizační odpor, více v následující části.

Na obrázku 31 je zobrazeno schéma zapojení pro vyhřívání výkonového tranzistoru. Kompletní schéma zapojení s mikrořadičem a obvodem pro měření teploty je zobrazeno na obrázku 51 v příloze A.



**Obrázek 31** Schéma zapojení topného prvku

PWM signál generovaný mikrořadičem je s využitím level-shiffteru posunut na úroveň napájení přípravku pro vyhřívání. Tento signál je poté invertován dalším stupněm s komplementárním tranzistorem. Další invertor je použit z toho důvodu, aby v klidovém stavu byl výkonový tranzistor zavřen. Tedy pokud bude MCU v resetu, přípravek nebude topit.

Maximální kolektorový proud je stanoven odporem do báze výkonového tranzistoru. Bohužel proudový zesilovací činitel tranzistoru má velký rozptyl. Dále je velikost  $h_{21e}$  závislá na kolektorovém proudu tranzistoru. Navíc s teplotou kolísá napětí báze emitor,

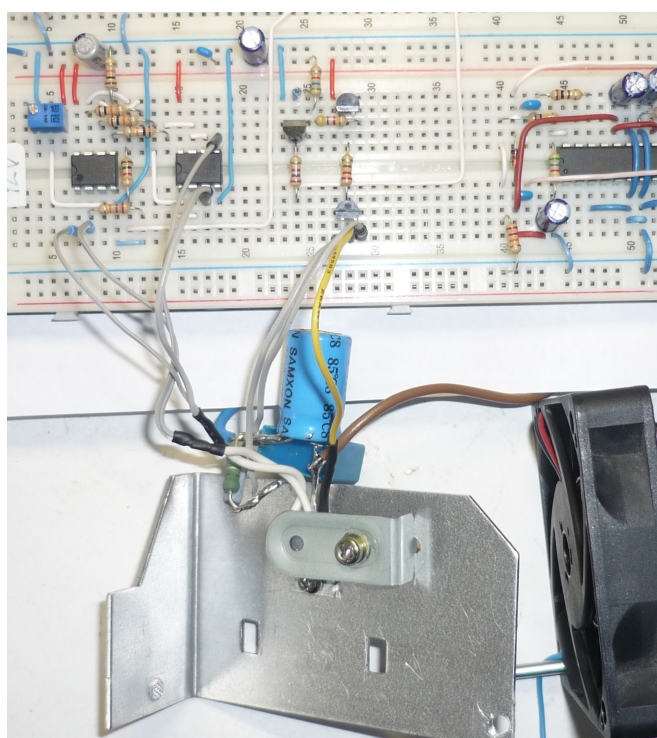
což má za následek změny proudu do báze. Tento jev je ovšem nežádoucí, a proto je nutné jej kompenzovat. Z pohledu regulace bychom mohly uvažovat, že při použití robustního regulátoru bude regulace na konstantní teplotu fungovat s minimálním vlivem kolísání  $h_{21e}$ , avšak z důvodu připojení tranzistoru mezi svorky napájecího zdroje je nutné striktně definovat pracovní režim tranzistoru v saturační oblasti, jinak by mohlo dojít ke zkratování zdroje.

Z tohoto důvodu je tranzistor  $T_4$  doplněn o emitorový odpor a tranzistor  $T_3$ . Úbytek napětí na emitorovém odporu otevírá tranzistor  $T_3$ , který poté svým kolektorovým proudem ubírá proud báze výkonovému tranzistoru. Tímto je vytvořena zpětnovazební smyčka, která stabilizuje kolektorový proud výkonového tranzistoru. Díky tomu je výkonový tranzistor bezpečně držen v saturační oblasti. Maximální kolektorový proud lze pak snadno určit z úbytku napětí na odporu  $R_5$ , který způsobí otevření tranzistoru  $T_3$ .

$$I_C \approx \frac{0,7}{R_5}. \quad (6.2)$$

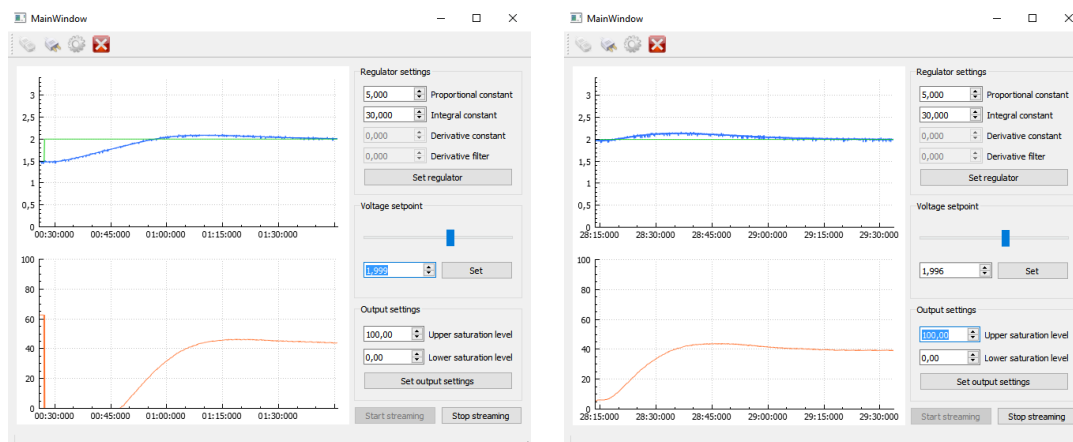
Odpor do báze  $R_4$  je pak možné zvolit menší, pro větší bázový proud, než by bylo nutné na základě proudového zesilovacího činitele získaného pro zvolený  $I_C$  z datasheetu výrobce. Tímto je stanovena rezerva, pokud by měl tranzistor za určitých provozních podmínek nižší proudový zesilovací činitel. Zbytek je zajištěn stabilizátorem proudu.

Na obrázku 32 je fotografie zapojení analogové části regulátoru teploty na nepájivém kontaktním poli.



**Obrázek 32** Fotografie zapojení regulátoru teploty

Na obrázku 33 je zobrazen přechodový děj při skoku požadované hodnoty a při skokové změně vnější poruchy. Vnější porucha byla v tomto případě realizována přidáním aktivního chlazení. K zaznamenání přechodových dějů byla použita demonstrační aplikace vyvinutá pro tyto účely, více v 7.

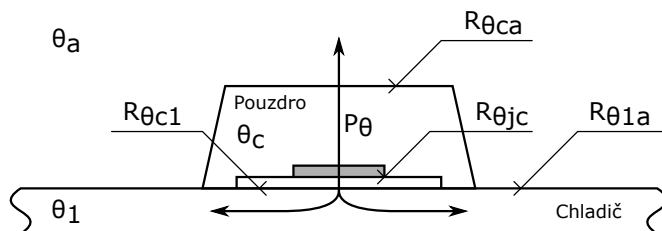


a) Odezva regulátoru teploty na skok reference      b) Odezva regulátoru teploty na poruchu

**Obrázek 33** Ukázka demonstrační aplikace pro regulátor teploty

### 6.1.2 Návrh obvodových prvků topného přípravku

V této části je proveden návrh prvků pro realizaci topné části regulační úlohy. Pro tuto úlohu byl zvolen bipolární tranzistor BD243C pro jeho snadnou dostupnost a nízkou cenu. Tento tranzistor byl zvolen s ohledem na očekávaný ztrátový výkon který na něm bude mařen v teplo. Na základě katalogových hodnot tranzistoru lze určit minimální kolektorový proud  $I_C$ , který bude potřeba pro ohřátí tranzistoru o  $\Delta T$ . Pro odvod tepla z pouzdra tranzistoru je použit hliníkový chladič. Tranzistor je připevněn na chladič bez izolační podložky s použitím teplo vodivé pasty pro snížení tepelného odporu mezi pouzdrem tranzistoru a chladičem. Snímací termistor je připojen na svrchní straně tranzistoru pomocí spony. Na obrázku 34 je naznačena situace při odvodu tepla z tranzistoru.



**Obrázek 34** Odvod tepla z čipu tranzistoru

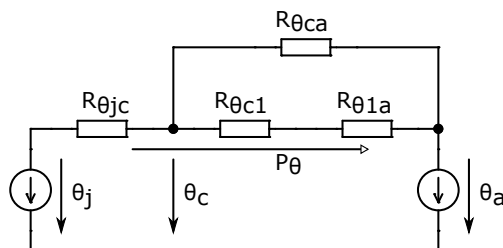
kde  $P_\theta$  představuje tepelný tok,  $\theta_a$  je teplota okolí,  $\theta_1$  je teplota chladiče a  $\theta_c$  je teplota pouzdra. Odváděné teplo z čipu tedy musí projít přes tepelný odpor mezi čipem a pouzdrem  $R_{\theta_{jc}}$ , tepelný odpor mezi pouzdrem tranzistoru a chladičem  $R_{\theta_{c1}}$  a přes odpor mezi chladičem a okolím  $R_{\theta_{1a}}$ . Navíc je zde tepelný odpor přechodu pouzdra a okolí  $R_{\theta_{ca}}$ , který se řadí paralelně k sériové kombinaci odporů  $R_{\theta_{c1}}$  a  $R_{\theta_{1a}}$ . Na obrázku 35 je zobrazeno ekvivalentní schéma k tomuto tepelnému obvodu.

Řešení tohoto tepelného obvodu je analogické k řešení elektrického obvodu. Pro oteplení tranzistoru bude platit:

$$\Delta\theta = \theta_j - \theta_a = P (R_{\theta_{jc}} + R_{\theta_{ca}} \parallel (R_{\theta_{c1}} + R_{\theta_{1a}})), \quad (6.3)$$

kde  $\theta_j$  je teplota čipu a  $P$  je výkon mařený na tranzistoru. Pro tepelný odpor mezi





Obrázek 35 Ekvivalentní schéma tepelného obvodu

chladičem a okolním prostředím lze použít přibližný vztah [24]:

$$R_{\theta_{1a}} \approx \frac{3,3}{\sqrt{d\lambda}} C^{0,25} + \frac{650}{S} C, \quad (6.4)$$

kde  $d$  je tloušťka použitého plechu,  $S$  je plocha chladiče,  $\lambda$  je tepelná vodivost materiálu chladiče a  $C$  je korekční faktor, který je dán například geometrií umístění chladičícího plechu nebo povrchovou úpravou. Tento vztah navíc platí pouze za omezujících podmínek kterými jsou například teplotní rozsah a přítomnost jediného topného zdroje na chladiči [24]. Nicméně pro přibližný odhad hodnoty tepelného odporu mezi chladičem a okolím je vyhovující. Přibližné hodnoty korekčního faktoru a tepelného odporu na styku tranzistoru s chladičem při použití teplo vodivé pasty lze dohledat v tabulkách.

Teplota pouzdra tranzistoru  $\theta_c$  je pak dána vztahem:

$$\theta_c = P (R_{\theta_{ca}} \parallel (R_{\theta_{c1}} + R_{\theta_{1a}})) + \theta_a. \quad (6.5)$$

Na základě vztahu 6.3 je možné určit maximální velikost kolektorového proudu pro maximální zvolenou teplotu tranzistoru:

$$I_C = \frac{\theta_j - \theta_a}{(R_{\theta_{jc}} + R_{\theta_{ca}} \parallel (R_{\theta_{c1}} + R_{\theta_{1a}})) (U_{CC} - 0,7)}. \quad (6.6)$$

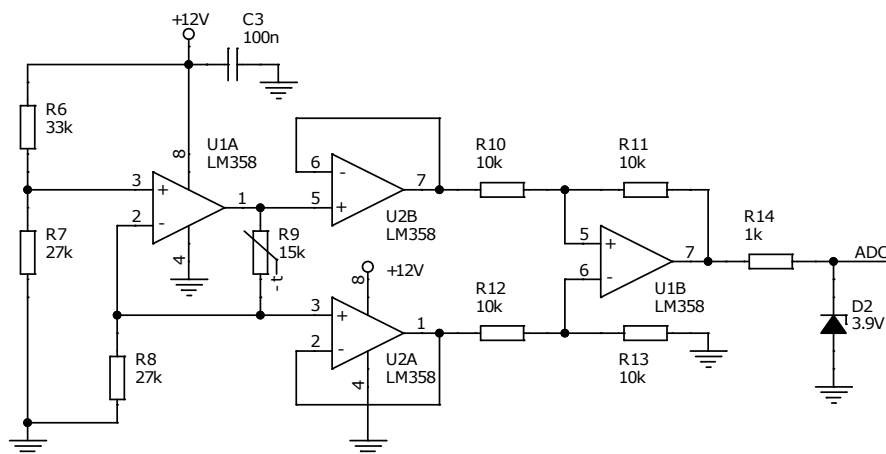
Hodnota odporu  $R_5$  pro stabilizaci  $I_C$  je určena podle vztahu 6.2. Odpor  $R_1$ ,  $R_2$  a  $R_3$  jsou určeny tak, aby tranzistory  $T_1$  a  $T_2$  pracovaly ve spínacím režimu. Jejich výpočet je velmi snadný.

### 6.1.3 Snímání teploty

Teplota tranzistoru je snímána pomocí NTC termistoru. Důvodem výběru NTC termistoru je jeho velmi nízká cena a snadná dostupnost. Nevýhodou je však nelineární závislost odporu na teplotě, tedy teplotní koeficient  $\alpha_{\theta}$  není konstantní, ale je funkcí teploty. V malém rozsahu pracovních teplot je charakteristika téměř lineární, a lze ji snadno linearizovat.

Odpor termistoru je snímán pomocí čtyřvodičové metody. Termistor je napájen zdrojem konstantního proudu realizovaného operačním zesilovačem. Úbytek napětí je poté zesílen diferenčním zesilovačem. Napětí na výstupu diferenčního zesilovače je snímáno AD převodníkem. Schéma měřicí části je zobrazeno na obrázku 36.

Napěťové svorky jsou přivedeny na diferenční zesilovač. Tímto je eliminován přídavný úbytek napětí na odporu  $R_1$ , který by se přičítal k měřenému napětí, neboť zdroj proudu v zapojení na obrázku 36 vyžaduje plovoucí zátěž. Vstup diferenčního zesilovače je navíc impedančně oddělen pomocí dvou operačních zesilovačů, které jsou zapojeny jako sledovače. Bez impedančního oddělení by bylo nutné vzít v úvahu vliv vstupního odporu rozdílového zesilovače, který je z pohledu svorek zdroje proudu připojen paralelně



Obrázek 36 Schéma zapojení pro měření odporu termistoru

k odporu termistoru. Proud zdroje konstantního proudu je pak rozdělen mezi odpor termistoru a vstupní odpor zesilovače. Tímto by vznikla chyba měření přímo úměrná poměru odporu termistoru ke vstupnímu odporu. Diferenční napětí zesilovače, které bylo zesíleno na výstup, by v takovém případě bylo:

$$U_d = I \frac{R_9 R_{in}}{R_9 + R_{in}}, \quad (6.7)$$

kde  $R_9$  je odpor termistoru,  $R_{in}$  je vstupní odpor zesilovače a  $I$  je proud zdroje. Pokud uvážíme, že diferenční zesilovač má zesílení rovno jedné, je napětí  $U_d$  rovno výstupnímu napětí zesilovače. Pro vstupní odpor diferenční zesilovač platí:

$$R_{in} = R_{10} + R_{12}. \quad (6.8)$$

Pokud vyjádříme absolutní chybu měření jako rozdíl napětí při nekonečném vstupním odporu zesilovače a napětí s uvážením reálného diferenčního zesilovače dostáváme:

$$\Delta U = -I \frac{R_9}{R_{in}}. \quad (6.9)$$

Chyba měření by byla silně závislá na teplotě a byla by největší pro nízké teploty, kdy je odpor termistoru největší. Tuto chybu lze snadno korigovat výpočtem, kdy

$$R_9 = \frac{U_d R_{in}}{I R_{in} - U_d}. \quad (6.10)$$

#### 6.1.4 Návrh obvodových prvků pro snímání teploty

Zdroj konstantního proudu tvořený operačním zesilovačem  $U_{1A}$  je nastaven tak, aby úbytek na odporu termistoru  $R_9$  nepřesáhl při pokojové teplotě maximální napětí pro vstup AD převodníku. Maximální velikost proudu, který bude protékat termistorem lze určit z odporu termistoru při pokojové teplotě  $R_{9m}$ .

$$I_2 = \frac{U_{R_{9m}}}{R_{9m}}, \quad (6.11)$$

kde  $I_2$  je výstupní proud zesilovače a  $U_{R_{9m}}$  je maximální napětí na termistoru. Označíme-li napětí na neinvertujícím vstupu OZ jako  $U_1$  a napětí na výstupu jako  $U_2$ ,

lze na základě požadavku maximálního výstupního napětí OZ stanovit velikost odporu  $R_8$  jako:

$$R_8 = \frac{U_{2m} - U_{R_8m}}{I_2}, \quad (6.12)$$

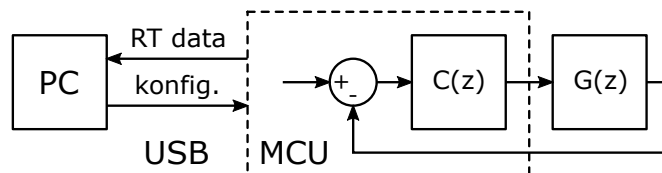
kde  $U_{2m}$  je maximální výstupní napětí zesilovače. Napětí na neinvertujícím vstupu zesilovače je pak dáno vztahem:

$$U_1 = R_8 I_2. \quad (6.13)$$

Velikosti odporů ve zpětné vazbě rozdílového zesilovače jsou zvoleny pro celkové zesílení  $H_{Up} = 1$ . Navíc velikosti odporů nemusí být příliš velké díky impedančnímu oddělení sledovači, viz sekce 6.1.3.

## 7 Aplikace pro sběr dat během regulace

Pro demonstrační účely byla vytvořena PC aplikace pro sběr dat v reálném čase během procesu regulace. Sledovanými veličinami jsou požadovaná hodnota, hodnota výstupní veličiny a akční zásah regulátoru. Tato PC aplikace spolu s demonstračním firmwarem umožňuje konfiguraci a ladění samotného regulátoru. Díky transparentnímu uživatelskému rozhraní lze snadno nastavovat konstanty regulátoru určeného pro daný typ demonstrační úlohy. Navíc je možné nastavovat požadovanou hodnotu a parametry výstupu regulátoru, díky tomu je možné sledovat odezvy regulační smyčky na změny reference nebo případné vnější poruchy. Na obrázku 37 je zobrazeno blokové schéma pro sběr dat během regulace a konfiguraci regulátoru.



Obrázek 37 Blokové schéma přenosu dat během regulace

### 7.1 Přenos dat

Pro přenos dat z regulace v případě MCU STM32F042 a STM32L072 je využito USB a emulace sériové linky VCP (Virtual Com Port). K implementaci na straně MCU jsou s výhodou využity CDC (Communication Device Class) ovladače od STMicroelectronics, které umožňují snadné zprovoznění VCP přes USB. Tyto ovladače jsou součástí balíku HAL (Hardware Abstraction Layer) ovladačů. V případě použití jiných MCU, které nemají USB, nebo vývojového kitu Nucleo, který nemá vyvedené USB na externí konektor, je k přenosu dat použita standardní sériová linka. V takovém případě je nutné použít externí UART-USB převodník, nebo převodník který je součástí STLink programátoru. Tento programátor je k dispozici na každém vývojovém kitu Nucleo, kde je MCU přímo spojeno s STLink programátorem.

Nutnou součástí pro funkčnost této aplikace jsou ovladače od STMicroelectronics na straně PC. Tyto ovladače jsou volně ke stažení na webových stránkách STMicroelectronics.

Protože z regulace je přenášen větší objem dat, je přenos pomocí formátovaných textových zpráv nevhodným řešením. Z tohoto důvodu jsou data posílána v binárním tvaru.

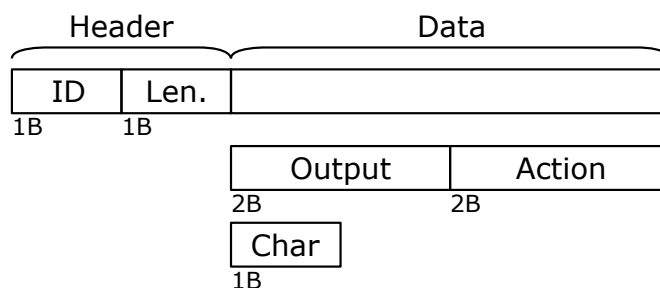
Pro přenos dat byl navržen jednoduchý datový rámec, který umožňuje detekovat velikost, začátek a konec příchozích dat. Díky tomu je pak možné snadno identifikovat fragmentované rámce, nebo po sobě jdoucí rámce které byly nežádoucím způsobem připojeny za sebe. K tomu může zpravidla dojít v případě, kdy jsou data přenášena ještě přes další prvek, kterým je například zmiňovaný UART-USB převodník. V případě převodníku vestaveném ve vývojovém kitu Nucleo jsou přenášena data ukládána

v mezipaměti, a poté v daných okamžicích odesílána přes USB. V důsledku toho dochází ke fragmentaci dat a náhodnému slučování fragmentů z po sobě jdoucích zpráv.

Tato aplikace je napsána v Qt. Qt je multiplatformní framework pro vývoj aplikací jak pro embeded systémy, tak i desktop aplikace. Qt je založen na jazyce C++.

## 7.2 Datový rámec pro přenos dat

Na obrázku 38 je zobrazen formát datového rámce použitého pro komunikaci s regulátorem. Rámec se skládá z hlavičky a datové části. Součástí hlavičky je identifikátor přenášených dat. Dále je zde pole udávající počet následujících datových polí. Velikost datových polí, se však liší podle typu přenášených informací. V případě přenosu konfiguračních údajů, jako jsou například konstanty regulátoru, jsou data přenášena v podobě formátovaného textu. V takovém případě pole velikost udává počet ASCII znaků do konce rámce. Při přenosu binárních dat se jedná o počet vzorků v rámci. Každý vzorek má pak vyhrazeny čtyři bajty, dva pro přenos akčního zásahu a dva pro hodnotu výstupní veličiny. Tyto velikosti byly uváženy jako dostatečné pro přenos informací ze všech navržených regulačních úloh.

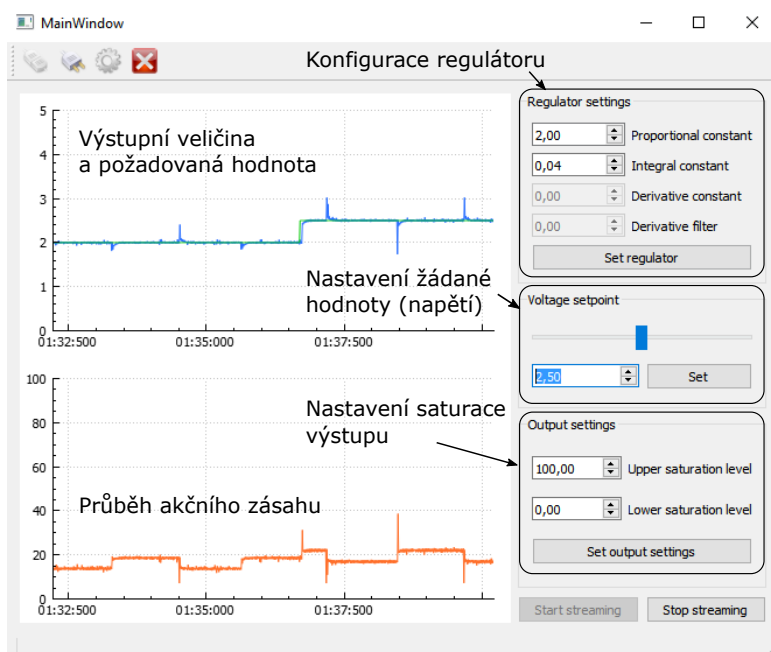


**Obrázek 38** Formát datového rámce pro přenos dat z regulace

Aktuální požadovaná hodnota je odesílána formou konfiguračních dat, tedy jako formátovaný text. Tato hodnota je posílána jen jednou za čas, protože k její změně dochází jen v PC aplikaci. Tímto je datový přenos během sledování regulace odlehčen, a lze přenášet o to více vzorků za vteřinu. Propustnost dat je nakonec omezena jen PC aplikací, respektive její schopností překreslování grafu, a nikoli přenosovou kapacitou USB.

Na obrázku 39 je zobrazeno uživatelské rozhraní PC aplikace vyvinuté pro sběr dat během regulace.

V horní části grafu jsou současně vykreslovány hodnoty výstupní veličiny a aktuální požadované hodnoty. Ve spodním grafu je průběh akčního zásahu regulátoru. Po pravé straně jsou prvky umožňující konfiguraci regulátoru, nastavení požadované hodnoty a saturačních úrovní výstupu.



Obrázek 39 Uživatelské rozhraní aplikace pro sběr dat

## 8 Softwarová implementace regulátoru

V této části práce se zaměřím na softwarovou implementaci regulátoru na mikrořadičích použitých ve všech regulačních úlohách. V úvodu bude proveden rozbor PID regulátoru, ke kterému bude v další části odvozena diskrétní aproximace. Nakonec budou v souvislosti s použitými mikrořadiči probrány aspekty týkající se výpočetní náročnosti algoritmu regulátoru.

### 8.1 Rozbor PID regulátoru

PID regulátor je jeden z nejpoužívanějších regulátorů v průmyslu. Tento regulátor se skládá ze tří složek, proporcionální, integrační a derivační. Přenosová funkce PID regulátoru je pak dána součtem těchto složek.

Proporcionální regulátor je regulátor, jehož akční zásah je dán pouze zesílenou odchylkou od reference (požadované hodnoty). Pro akční zásah platí:

$$u(s) = K_P e(s), \quad (8.1)$$

kde  $K_P$  je proporcionální konstanta, nebo také konstanta zesílení. Realizace takového regulátoru je velice snadná. U systémů, které nejsou astatické, tento regulátor generuje takový akční zásah, jehož výsledkem je trvalá odchylka výstupní hodnoty od reference. Tato trvalá odchylka je tím větší, čím je menší zesílení regulátoru. Při zvyšování zesílení lze regulační odchylku na skok reference snižovat, avšak roste strmota náběžné hrany a amplituda překmitů. V případě astatických systémů, tj. systémů, jejichž přenosová funkce má pól v nule a systém je tedy sám o sobě integračního charakteru, tento regulátor poskytuje přesnou regulaci bez odchylky od reference.

Integrační složka regulátoru je složka, jejíž příspěvek k akčnímu zásahu se obvykle vyjadřuje jako:

$$u(s) = K_I \frac{e(s)}{s}, \quad (8.2)$$

kde  $K_I$  je integrační konstanta. Tento regulátor integruje odchylku od referenční hodnoty. Tím je do systému zaveden astatismus prvního řádu. Díky tomu je možné dosáhnout nulové odchylky výstupní hodnoty od reference při skokové změně  $u$  v systému, které nejsou astatické. Tato složka však může již destabilizovat uzavřenou regulační smyčku, která by se samotným P regulátorem byla stabilní i pro všechny hodnoty zesílení P složky. Integrační složka se téměř nepoužívá samostatně, neboť odezva na skok reference takového regulátoru může být příliš pomalá.

Akční zásah derivační složky se standardně vyjadřuje jako:

$$u(s) = K_D s e(s), \quad (8.3)$$

kde  $K_D$  je derivační konstanta. Tato složka naopak reaguje velmi rychle. Její uplatnění je jen při přechodovém ději, neboť při ustálené velikosti regulační odchylky je její akční zásah nulový. V důsledku toho se derivační složka téměř výhradně používá pouze v kombinaci s nějakou dříve zmíněnou složkou.

V praxi je nemožné použít derivační složku ve formě ideální derivace. Prvním důvodem je, že žádný fyzikální systém nemá nekonečné zesílení na nekonečných frekvencích. Druhým důvodem je fakt, že systém s takovým chováním by velmi zesiloval šum, který se právě vyskytuje spíše na vyšších kmitočtech. Z toho důvodu se derivační složka filtruje. Jednoduchý filtr může být realizován zavedením pólu do přenosové funkce derivační složky regulátoru. Přenosovou funkci této složky lze pak vyjádřit jako:

$$u(s) = K_D \frac{se(s)}{1 + \frac{K_D}{N}s}, \quad (8.4)$$

kde  $N$  je činitel filtrace.

Pokud sečteme všechny složky, dostáváme pro akční zásah spojitého PID regulátoru:

$$u(s) = K_P e(s) + K_I \frac{e(s)}{s} + K_D \frac{se(s)}{1 + \frac{K_D}{N}s}, \quad (8.5)$$

## 8.2 Diskrétní aproximace spojitého PID regulátoru

Rovnice 8.5 vyjadřuje akční zásah spojitého PID regulátoru s filtrovanou derivační složkou. Pro softwarovou implementaci je nutné najít diskrétní aproximaci této rovnice.

Pokud budeme chtít takový regulátor aproximovat diskrétním regulátorem, je nejjednodušší postup aproximovat derivační a integrační složku akčního zásahu zvlášť, tak jak jsou probrány v sekci 8.1. Pro celkový akční zásah můžeme psát:

$$u(s) = u_P(s) + u_I(s) + u_D(s) = C_P(s)e(s) + C_I(s)e(s) + C_D(s)e(s), \quad (8.6)$$

kde  $C_P(s)$ ,  $C_I(s)$  a  $C_D(s)$  jsou přenosy jednotlivých složek regulátoru. Pro náhradu spojitě integrační a derivační složky použijeme aproximaci zpětnou diferencí:

$$\frac{dx(t)}{dt} \approx \frac{x[k] - x[k-1]}{T_S}, \quad (8.7)$$

kde  $T_S$  je perioda vzorkování. Uplatněním Z transformace dostáváme:

$$sx(s) \approx \frac{1 - z^{-1}}{T_S} x(z). \quad (8.8)$$

Aproximací integrační složky akčního zásahu regulátoru dostáváme:

$$u_I(z) = K_I \frac{T_S}{1 - z^{-1}} e(z). \quad (8.9)$$

Úpravou pak obdržíme vyjádření v záporných mocninách  $z$ , které je intuitivnější, neboť  $z^{-1}$  reprezentuje přímo jednotkové zpoždění:

$$u_I(z) = z^{-1} u_I(z) + K_I T_S e(z). \quad (8.10)$$

Uplatněním zpětné Z transformace dostáváme rovnici pro výpočet integračního příspěvku diskrétní aproximace regulátoru:

$$u_I[k] = u_I[k-1] + K_I e[k], \quad (8.11)$$



kde  $\acute{K}_I = K_I T_S$  je nová integrační konstanta přepočítaná pro diskrétní regulátor. Stejným postupem vyjádříme příspěvek nefiltrované derivační složky. Po aproximaci akčního zásahu derivační složky dostáváme:

$$u_D(z) = K_D \frac{1 - z^{-1}}{T_S} e(z). \quad (8.12)$$

Úpravou této rovnice a vyjádřením v záporných mocninách  $z$  obdržíme:

$$u_D(z) = \frac{K_D}{T_S} (e(z) - z^{-1}e(z)). \quad (8.13)$$

Nyní opět provedeme zpětnou Z transformaci, čímž získáme:

$$u_D[k] = \acute{K}_D (e[k] - e[k - 1]), \quad (8.14)$$

kde  $\acute{K}_D = \frac{K_D}{T_S}$  je nová derivační konstanta. Pro filtraci derivační složky lze vybrat z celé řady filtrů, nicméně s ohledem na výpočetní výkon mikrořadičů s jádrem ARM-Cortex M0, více v 8.4, je vhodné použití například IIR (Infinite Impulse Response) filtr s exponenciálním zapomínáním, jehož výpočetní algoritmus není příliš náročný. Přenosovou rovnici takového filtru lze vyjádřit jako:

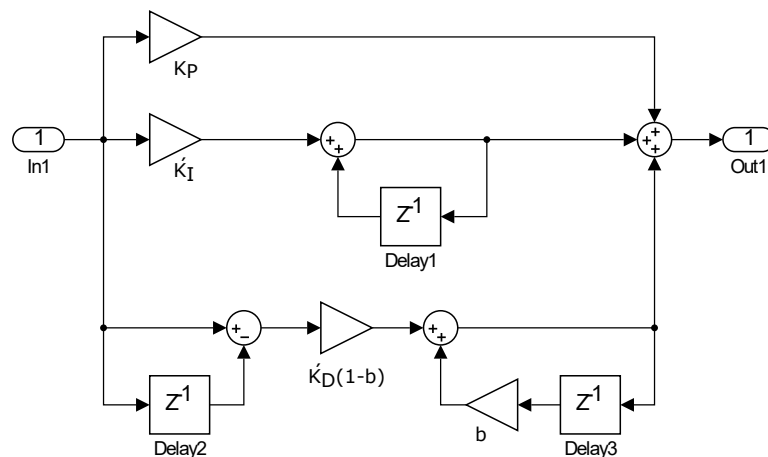
$$F(z) = \frac{1 - b}{1 - bz^{-1}} = \frac{u_{Df}(z)}{u_D(z)}, \quad (8.15)$$

kde  $u_{Df}(z)$  je akční zásah filtrované derivační složky a  $b$  je koeficient filtru. Vyjádřením  $u_{Df}(z)$  z 8.15, dosazením 8.14 a uplatněním zpětné Z transformace, dostáváme pro akční zásah filtrované derivační složky:

$$u_{Df}[k] = bu_{Df}[k - 1] + (1 - b)\acute{K}_D (e[k] - e[k - 1]). \quad (8.16)$$

Sečtením všech složek akčního zásahu, dostaneme akční zásah diskrétní aproximace spojitého PID regulátoru i se zařazeným IIR filtrem pro derivační složku:

$$u[k] = K_P e[k] + u_I[k - 1] + \acute{K}_I e[k] + bu_{Df}[k - 1] + (1 - b)\acute{K}_D (e[k] - e[k - 1]). \quad (8.17)$$



**Obrázek 40** Schéma diskrétního regulátoru

K této rovnici diskretizovaného PID regulátoru je možné sestavit schéma, které je zobrazeno na obrázku 40.

Za zmínku stojí, že s ohledem na optimalitu výpočetního algoritmu jsou všechny koeficienty v rovnici 8.17 vypočítány předem, nebo jsou přepočítány při změně konstant spojitého regulátoru. Během samotného výpočtu akčního zásahu se počítá s již určenými koeficienty.

Další alternativou je provést diskretní aproximaci filtrované derivační složky již se zahrnutou filtrací pomocí přidaného pólu, tak jak je uvedeno v rovnici 8.4. Postup aproximace je naprosto stejný, a proto jej nebudu znova popisovat. S použitím aproximace dopřednou diferencí, vyjádřením rovnice v záporných mocninách  $z$  a uplatněním zpětné  $Z$  transformace dostáváme:

$$u_{Df}[k] = \tilde{K}_D (u_D[k-1] + N(e[k] - e[k-1])), \quad (8.18)$$

kde  $\tilde{K}_D = \frac{K_D}{NT_S + K_D}$  je přepočítaná derivační konstanta. Pokud vyjádříme rovnici celkového akčního zásahu PID regulátoru s takto filtrovanou derivační složkou, dostáváme:

$$u[k] = K_P e[k] + u_I[k-1] + \acute{K}_I e[k] + \tilde{K}_D (u_D[k-1] + N(e[k] - e[k-1])). \quad (8.19)$$

Pokud se podíváme blíže na derivační složku akčního zásahu zjistíme, že rovnice 8.17 a 8.19 jsou ekvivalentní. V obou případech jde o jeden a ten samý regulátor pouze s jinak vyjádřenou filtrací derivační složky, neboť na základě porovnání obou rovnic můžeme psát:

$$b = \frac{K_D}{NT_S + K_D}. \quad (8.20)$$

Dále pokud porovnáme zbylý člen u výrazu  $(e[k] - e[k-1])$ , dostáváme:

$$\frac{(1-b)K_D}{T_S} = \frac{NK_D}{NT_S + K_D}. \quad (8.21)$$

Pokud dosadíme za  $b$  a provedeme úpravu, tak tato rovnice vede na:

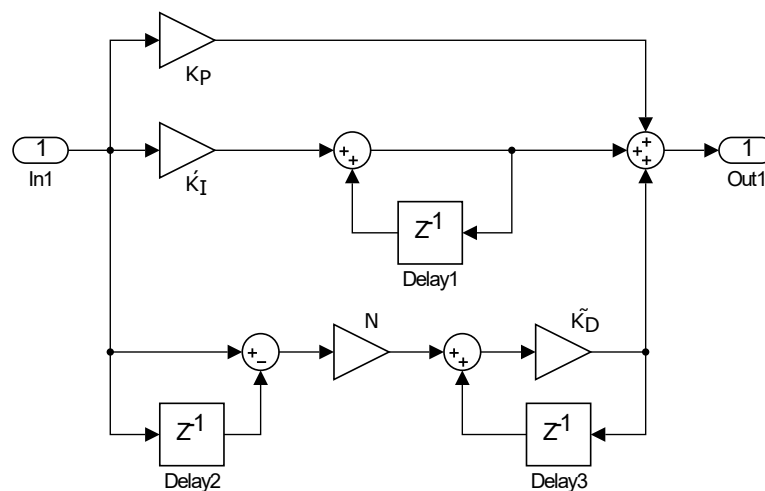
$$\frac{N}{NT_S + K_D} = \frac{N}{NT_S + K_D}. \quad (8.22)$$

Jedná se tedy o stejné regulátory, což také vyplývá ze skutečnosti, že IIR filtry jsou filtry, ke kterým existují ekvivalentní spojité filtry. Pokud sestavíme schéma podle rovnice 8.19, dostaneme tak jiné vyjádření téhož regulátoru.

Na obrázku 41 je zachyceno toto schéma. Dále je ještě možné diskutovat o umístění konstant  $\acute{K}_I$  a  $\acute{K}_D$  pro regulátor vyjádřený rovnicí 8.19. Zmíněné konstanty lze přemístit, pokud celou rovnici pro akční zásah dané složky vydělíme  $\acute{K}_I$  případně  $\acute{K}_D$ . Konstanta  $\acute{K}_D$  může být takto přesunuta na výstup celé derivační cesty před poslední sčítačku. Stejně tak může být přemístěna i integrační konstanta. Toto přemístění bude mít zejména vliv na velikosti přírůstků ukládaných v integrátoru.

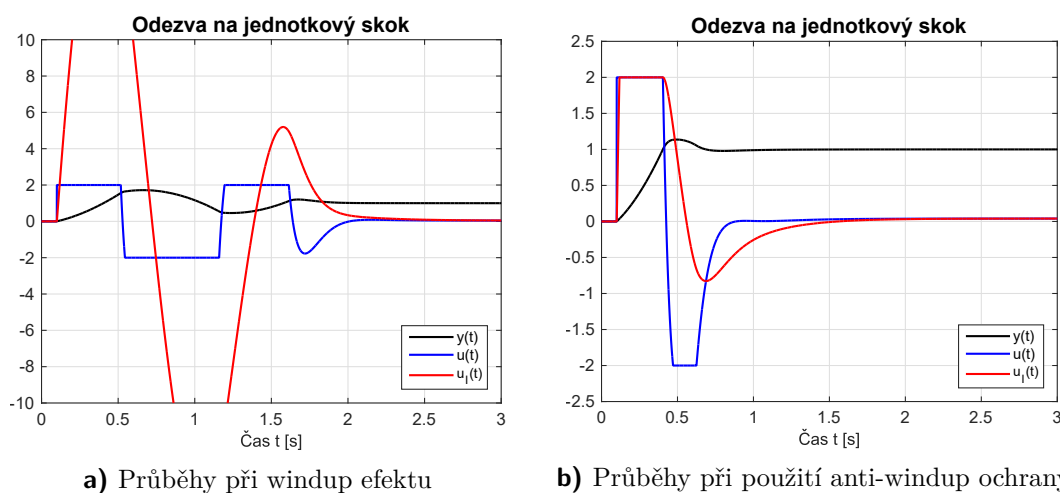
### 8.3 Anti-windup

Windup efekt je efekt, ke kterému může dojít, pokud je výstup regulátoru amplitudově omezen a regulátor obsahuje akumulární prvek jakým je integrátor. Během windup efektu integrátor stále akumuluje nenulovou odchylku, a generuje tak akční zásah, který je už za hranicí saturace výstupu regulátoru nebo akčního členu. Pokud se regulovaná hodnota opět přiblíží referenci, je generovaný akční zásah podpořen veškerou odchylkou



Obrázek 41 Schéma diskrétního regulátoru

naakumulovanou i v době saturace výstupu. Pokud regulovaná veličina překročí referenční hodnotu a vznikne tak odchylka s opačným znaménkem, výstup regulátoru je stále satureován na předešlou hodnotu vlivem akumulované odchylky. Než regulátor začne opět reagovat, musí nejprve dojít k regeneraci akumulované odchylky zpět do nuly. Tímto může dojít k nestabilitě zpětnovazební smyčky i přesto, že regulátor je robustní a regulační smyčka je v lineární oblasti akčního zásahu stabilní. Z toho důvodu je nutné regulátor doplnit takzvanou anti-windup ochranu, která zamezí takovému chování. Na obrázku 42 je porovnání odezvy regulační smyčky na jednotkový skok v čase  $t = 0, 1$  s jak pro systém s implementovanou antiwindup ochranou, tak pro systém bez ní. Oba průběhy přísluší stejnému stabilnímu systému druhého řádu, se stejně nastaveným PI regulátorem.



a) Průběhy při windup efektu

b) Průběhy při použití anti-windup ochrany

Obrázek 42 Ukázka průběhu výstupní veličiny, akčního zásahu a zásahu integrátoru

Průběh  $y(t)$  představuje výstupní veličinu,  $u(t)$  je akční zásah regulátoru a  $u_I(t)$  je akční zásah integrační složky PI regulátoru. Z obrázku 42a je patrné, že bez antiwindup ochrany je ustálení výstupní hodnoty mnohem delší. Důvodem je právě zpožděná reakce regulátoru ve chvíli, kdy odchylka od referenční hodnoty protíná nulovou hodnotu. Tato zpoždění může, jak již bylo zmíněno, vést až k nestabilitě.

Anti-windup ochrana je snadná na implementaci a zaručuje bezchybné chování re-

gulátoru při omezeném akčním zásahu. Stačí již při výpočtu akčního zásahu integrační složky kontrolovat překročení saturačních mezí a saturovat samotnou hodnotu integrátoru, tj. složku  $u_I[k]$  v rovnici 8.17 a 8.19.

## 8.4 Hardwarové aspekty při implementaci regulátoru

V případě úloh regulace napětí a teploty, je použit mikrořadič STM32F042, nebo STM32L072. V obou případech se jedná o MCU s jádrem ARM Cortex M0, které neobsahuje FPU (Floating Point Unit), a tedy neimplementuje operace s čísly s plovoucí desetinnou čárkou přímo v hardwaru [25]. Toto má za následek, že operace s čísly typu float jsou překladačem implicitně převáděny na operace s celými čísly. Z tohoto důvodu jsou tyto výpočetní operace velmi časově náročné. Délka výpočtu jedné aritmetické operace pak může trvat až desítky strojových cyklů a navíc ještě závisí na velikosti obou operandů. Při implementaci algoritmu regulátoru je nutné na tento fakt brát zřetel, a proto je vhodné vybírat velikosti konstant z oboru kladných, nebo záporných mocnin 2. Další optimalizací výpočtu může být snaha použít pouze celočíselné výpočty. V demonstračním softwaru je i přesto v případě regulátoru napětí s buck konvertorem a “lineárního” regulátoru použito floatové aritmetiky. Důvodem použití floatové aritmetiky je, že demonstrační firmware dovoluje konfiguraci samotného regulátoru. Z tohoto důvodu je nutné konstanty diskrétní aproximace spojitého regulátoru přepočítávat na základě uživatelem stanovených hodnot a zde je použití reálných čísel výhodné.

V případě regulátoru polohy byl pro testování regulátoru použit MCU STM32F411RE na vývojovém kitu Nucleo. Toto MCU obsahuje jádro ARM Cortex-M4 a již disponuje FPU jednotkou. Díky tomu MCU implementuje aritmetické operace s čísly s plovoucí desetinnou čárkou již v hardwaru. Jediným omezením je šířka registrů v FPU, která je 32 bitů, a proto lze počítat s hardwarovou podporou aritmetických operací jen v jednoduše přesnosti, tedy s čísly typu float. Aritmetické operace s čísly typu double jsou opět překladačem převáděny na sady celočíselných operací a pak je výpočet opět mnohem více časově náročný. Z tohoto důvodu je nutné při použití floatové aritmetiky striktně dbát na explicitní ošetření typů zejména u reálných konstant, kde by vlivem implicitní typové konverze mohlo snadno dojít k překlopení do aritmetiky s čísly typu double.

## 9 Pomocné aplikace

V této části jsou popsány pomocné aplikace, které vznikly jako vedlejší produkt při návrhu regulačních úloh. Jenou z těchto aplikací je UART-USB převodník realizovaný na MCU STM32F042. Další aplikací je PWM generátor, který vznikl za účelem testování stejnosměrných a krokových motorů. Tento generátor je navržen pro MCU STM32F042 a STM32F411.

### 9.1 UART-USB převodník

Výhodou tohoto UART-USB převodníku je jeho snadná realizace a s již hotovým firmwarem si jej může postavit každý student.

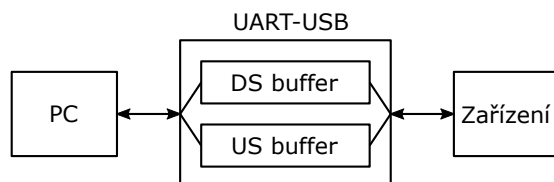
Převodník je konfigurovatelný v celé škále možností, které umožňuje VCP ovladač od STMicroelectronics. Převodník používá vestavěnou UART periférii, která je konfigurována vždy při připojení k VCP pomocí terminálu na straně PC. Převodník je schopen správně fungovat od 4800 baudu/s až do 460800 baudu/s. Dále je možno přenášet bloky dat o velikosti do 64B.

#### 9.1.1 Konfigurace UART-USB převodníku

Po power-down restartu MCU je inicializována USB periférie. V době inicializace také proběhne enumerace zařízení. Po enumeraci je MCU v PC viditelný jako zařízení CDC a je možné se k němu připojit pomocí jakékoliv terminálové aplikace. Před připojením jsou v terminálové aplikaci zpravidla nastaveny výchozí parametry sériové linky, nebo je uživatel zadá sám. Tyto parametry COM portu nemají žádný vliv pro komunikaci se samotným MCU, neboť se jedná pouze o emulaci sériové linky. Komunikace pak probíhá pomocí roury pro blokový přenos. Při připojení ke COM portu jsou zmíněné konfigurační údaje odeslány paketem na konfigurační rouře. Tyto data je pak možné v MCU zpracovat a na základě hodnot jednotlivých datových polí je možné podle nich nastavit UART periférii, která tvoří druhou bránu UART-USB převodníku. K implementaci je s výhodou použít ovladače pro USB zařízení typu CDC.

#### 9.1.2 Implementace brány UART-USB převodníku

Přenášená data lze rozdělit do dvou toků, upstream a downstream. Příchozí data jsou ukládána do kruhových bufferů, zvláště pro oba toky. Při naplnění každého bufferu dochází k odeslání dat na opačnou bránu. Navíc je použit hardwarový čítač, který pravidelně generuje přerušování, během kterého jsou dočasně uložená data odeslána. Tím dochází k periodickému čištění cyklických bufferů jak na downstreamu tak upstreamu. Pro přenos dat na downstream je využit UART2. Na obrázku 43 je naznačeno blokové schéma UART-USB převodníku.



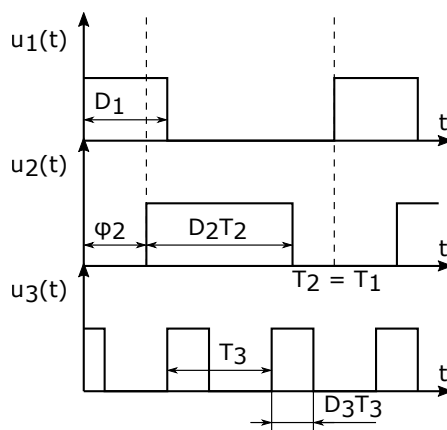
Obrázek 43 Blokové schéma UART-USB převodníku

## 9.2 PWM generátor

PWM generátor navržený především pro testovací účely disponuje třemi (v případě mikrořadiče STM32F042) až čtyřmi (v případě STM32F411) kanály. U každého kanálu generátoru je možné nezávislé nastavení střídy a frekvence generovaného signálu. Užitečnější funkcí je však možnost synchronizace všech kanálů, díky čemu je možné generování sekvence obdélníkových pulzů, jejichž vzájemný fázový posun je nastavitelný. Možnosti generátoru jsou následující:

- nastavení frekvence od jednotek Hz do 50kHz v případě generátoru pro vývojový kit Nucleo F411RE a do 24kHz v případě MCU STM32F042
- nastavení střídy signálu s minimálním krokem  $\frac{1}{2000}$
- 4 nezávislé kanály v případě MCU STM32F411, 3 v případě STM32F042
- možnost synchronizace kanálů s kanálem master
- nastavení fázového posunu mezi synchronizovanými kanály
- zapnutí/vypnutí výstupu
- invertování generovaného signálu

Na obrázku 44 je zobrazen příklad konfigurace kanálů PWM generátoru. První dva kanály jsou synchronizovány. Třetí kanál synchronizován není, a je nastaven nezávisle na předešlých kanálech.



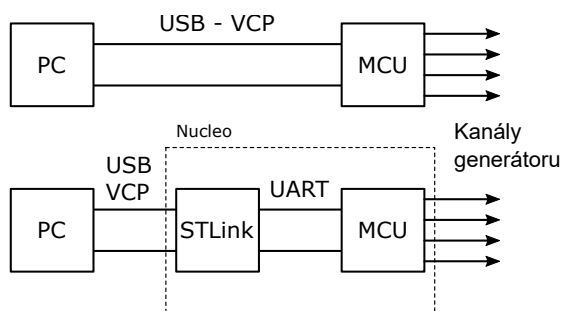
Obrázek 44 Příklad konfigurace kanálů PWM generátoru

kde  $\varphi_2$  je fázový posun druhého kanálu vůči prvnímu,  $T_1$  až  $T_3$  jsou periody kanálů a  $D_1$  až  $D_3$  jsou střídy kanálů. První kanál je v roli mastera.

### 9.2.1 Ovládání PWM generátoru

Na obrázku 45 je zobrazeno připojení PWM generátoru k PC.

PWM generátor se ovládá pomocí emulace sériové linky na USB. Ke konfiguraci generátoru je použita sada jednoduchých jednoznakových příkazů, které jsou popsány v tabulce 1.



Obrázek 45 Blokové schéma připojení PWM generátoru k PC

příkaz	parametry	popis
f	<kanál> <hodnota>	nastavení frekvence na daném kanálu
d	<kanál> <hodnota>	nastavení střídání na daném kanálu
t	<kanál> <hodnota>	nastavení délky periody signálu
p	<kanál> <hodnota>	nastavení fáze mezi daným slavem a masterem
o	<kanál> <hodnota>	zapnutí nebo vypnutí výstupu daného kanálu
i	<kanál>	invertování signálu na daném kanálu
s	<kanál>	synchronizace daného kanálu
r	<kanál>	uvolnění daného kanálu ze synchronizace

Tabulka 1 Tabulka příkazů podporovaných PWM generátorem

Pro PWM generátor bylo také implementováno SCPI rozhraní. K tomuto účelu byl využit open-source SCPI parser. Jedná se o parser textových příkazů se syntaxí dle SCPI standardu. Tento parser implementuje základní Common Commands standardu a současně umožňuje implementaci vlastní hierarchie příkazů.

### 9.2.2 Princip činnosti PWM generátoru

Funkce generátoru je založena na hardwarových čítačích dostupných v mikrořadičích STM32. Pro každý kanál PWM generátoru připadá jeden samostatný čítač. Čítače jsou vybrány tak, aby je bylo možné mezi sebou propojit pomocí interních multiplexorů. Vždy je jeden čítač v roli mastera. Čítač master má přednastaveno generování signálu TRGO (Trigger Output). Tento signál je pak interně propojen pomocí multiplexorů na spouštěcí vstupy ostatních čítačů. Díky tomu je možné čítače mezi sebou synchronizovat a generovat signály s přesně stanoveným fázovým posunem.

### 9.2.3 Standardní režim PWM generátoru

Každý čítač je vybaven předděličkou (registr PSC), jejíž hodnotu označíme jako  $p_x$ . Dále je k dispozici auto-reload registr (ARR) uchovávající maximální hodnotu čítače, při jejímž překročení dochází k nulování čítače. Tuto hodnotu označíme jako  $r_x$ . Každý z použitých čítačů disponuje čtyřmi vstupně výstupními kanály. Pro generátor je vždy použit jeden z těchto kanálů, který je nastaven v režimu generování PWM signálu. Pokud označíme hodnotu CCR registru (Capture/Compare Register), který přísluší danému kanálu čítače, jako  $c_x$ , lze pro parametry výstupního signálu odvodit následující vztahy:

$$f_{out} = \frac{f_0}{r_x p_x} \quad (9.1)$$

$$D = \frac{c_x}{r_x}. \quad (9.2)$$

Výpočet hodnot registrů pro požadovanou frekvenci  $f_{out}$  je následující: Nejprve je vypočítána hodnota předděličky  $p_x$  na základě požadované minimální hodnoty  $r_x$ . Tato hodnota omezuje maximální kmitočet generátoru a také stanoví nejhrubší krok, se kterým je možno nastavovat střihu výstupního signálu.

$$p_x = \frac{f_0}{r_{min} f_{out}}. \quad (9.3)$$

Poté je přepočítána hodnota  $r_x$ , čímž dojde ke zpřesnění kmitočtu v rozsahu prvně stanoveném hodnotou  $p_x$ . Hodnota  $r_x$  je vždy větší nebo rovna minimální hodnotě ARR. Tímto je zaručen maximální velikost kroku, s kterým lze nastavit střihu.

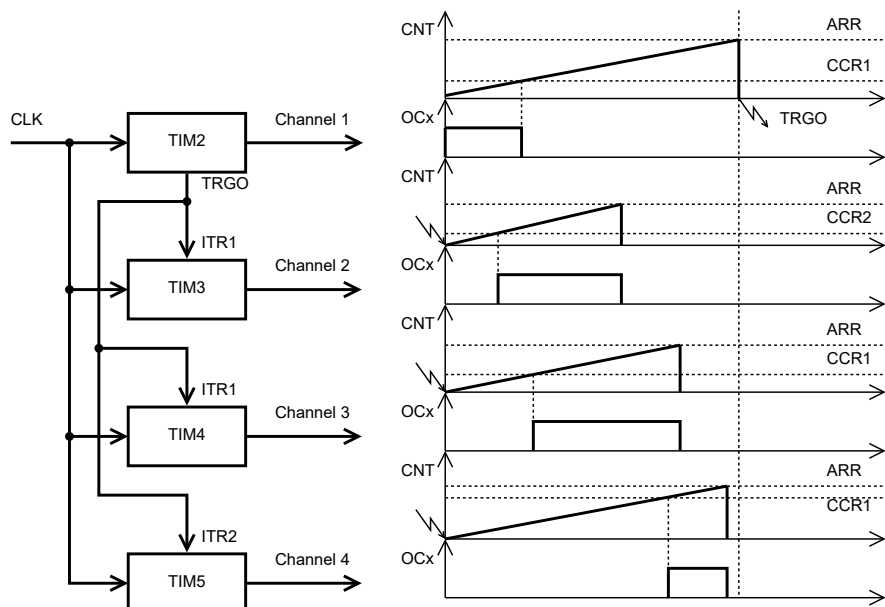
$$r_x = \frac{f_0}{p_x f_{out}}. \quad (9.4)$$

Hodnota  $c_x$  je pak jednoduše stanovena na základě požadované střidy  $D$ .

$$c_x = D r_x \quad (9.5)$$

### 9.2.4 Synchronizace kanálů PWM generátoru

Vlastní synchronizace je realizována čítačem master, který obsluhuje první kanál PWM generátoru. Tento čítač je nakonfigurován tak, že signál TRGO je propojen s událostí čítače Update Event (UEV). Pokud tedy dojde k přetečení čítače, je generován signál TRGO. Na tento signál jsou poté připojeny ostatní čítače svým příslušným vstupem pro spouštění, signály Internal Trigger (ITRx).

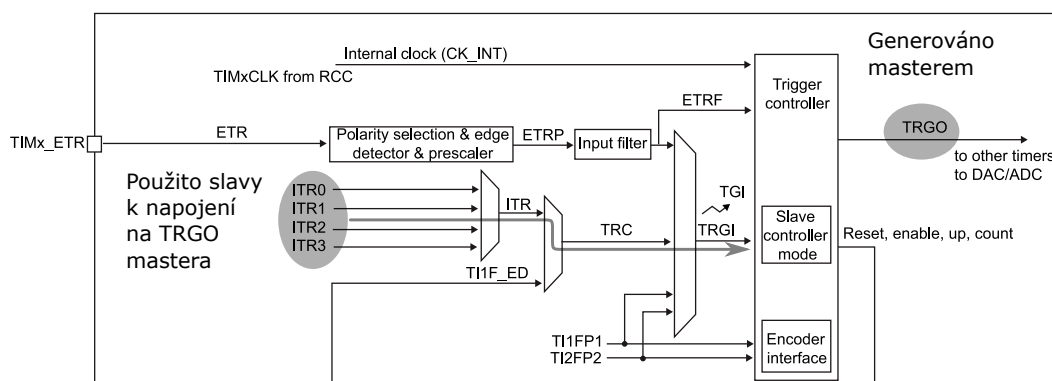


**Obrázek 46** Příklad časových průběhů během synchronizace kanálů



Tyto čítače jsou nastaveny v režimu One Pulse Mode, tedy pokud dojde k přetečení čítače, je čítač vypnut a čeká na další spouštěcí signál od čítače typu master. Princip činnosti je zachycen na obrázku 46. Z obrázku je vidět, že hodnota CCR registru u čítačů typu slave odpovídá fázovému posunu výstupního pulzu vůči výstupu mastera. Rozdíl hodnot ARR a CCR registrů pak odpovídá šířce pulzu. Dále je patrné, že generování výstupního pulzu probíhá inverzně vůči masterovi. Tato inverze je prováděna implicitně při přechodu na tento režim. Hodnota registru CCR mastera udává délku pulzu jako v normálním režimu.

Důležitým bodem je nastavení zpoždění při zachycení TRGO u všech slavů. Tímto je zaručena perfektní synchronizace mezi masterem a slavy, které začínají čítat přesně v okamžiku generování TRGO signálu. Na obrázku 47 je část blokového schématu čítače, kde jsou vyznačeny interní signály použité k synchronizaci čítačů.



**Obrázek 47** Interní signály čítače použité k propojení kanálu master se slavy

### 9.2.5 Obecné vztahy

V této části jsou vztahy popisující konfiguraci čítačů v případě synchronizace pro požadovaná nastavení.

Označíme-li  $c_s$  jako hodnotu registru CCR kanálu typu slave,  $r_s$  hodnotu ARR registru kanálu typu slave a  $r_m$  hodnotu registru ARR kanálu typu master, lze odvodit následující vztahy pro výpočet hodnot registrů ARR a CCR pro kanál typu slave. Hodnoty jsou počítány na základě požadované šířky pulzu a fázového posunu ve stupních.

$$c_s = \frac{\varphi_d r_m}{360}, \quad (9.6)$$

$$r_s = D_s r_m + c_s. \quad (9.7)$$

Hodnota předděličky  $p_s$  kanálu slave je rovna v této konfiguraci hodnotě předděličky kanálu master  $p_m$ .

Signály jsou tedy generovány jen a pouze samotnými čítači bez jakéhokoliv zásahu programu v průběhu generování signálu. Díky tomuto lze spoléhat na precizní nastavení parametrů výstupního signálu, které jsou dány parametry čítačů a jejich konfigurací.

### 9.2.6 Nastavení hodin PWM generátoru

Protože jsou parametry generovaného signálu přímo závislé na parametrech hodinového signálu čítače, je nutné jako zdroj hodin použít kalibrovaný zdroj, který má definovatelné parametry. V případě MCU STM32F042 je použit interní RC oscilátor HSI48.

Tento oscilátor je prioritně určen jako zdroj hodin pro USB periférii. Oscilátor HSI48 lze ale také použít jako zdroj pro ostatní periférie a pro jádro MCU. Vlastností tohoto oscilátoru je, že je pravidelně kalibrován z USB sběrnice. Díky tomuto je zajištěna větší kmitočtová stabilita než v případě jakéhokoliv jiného RC oscilátoru.

V případě PWM generátoru implementovaného na vývojovém kitu Nucleo s mikrořadičem STM32F411 je využít externí hodinový signál. Tento signál je přiveden přes bypass z programátoru STLink dostupného na vývojovém kitu. Tento zdroj hodin je pak použit i pro ostatní periférie a samotné jádro MCU.

## 10 Závěr

V této práci byl proveden návrh několika regulačních úloh pro demonstrační účely. Tyto úlohy mají sloužit pro výukové účely, a při jejich návrhu byl na tuto skutečnost brán zřetel. Ke každé z vybraných úloh byl proveden kompletní návrh, včetně oživení a ověření funkčnosti. Pro testovací a demonstrační účely byl ke každé úloze vytvořen firmware pro mikrořadič, který implementuje samotný regulátor. Zdrojové kódy k demonstračním regulátorům jsou v dostatečné míře okomentované a mohou sloužit jako příklady ve výuce. V této práci byly zdokumentovány všechny navržené regulační úlohy. Současně byl ke každé úloze nastíněn postup návrhu hodnot použitých obvodových prvků. Dále zde byly probrány některé aspekty implementace regulátorů na mikrořadičích z rodiny STM32. Byl zde popsán princip funkce a některé možné konfigurace použitých periférií mikrořadičů. Navržené regulační úlohy se liší jak podle principu regulace tak podle obtížnosti jejich implementace.

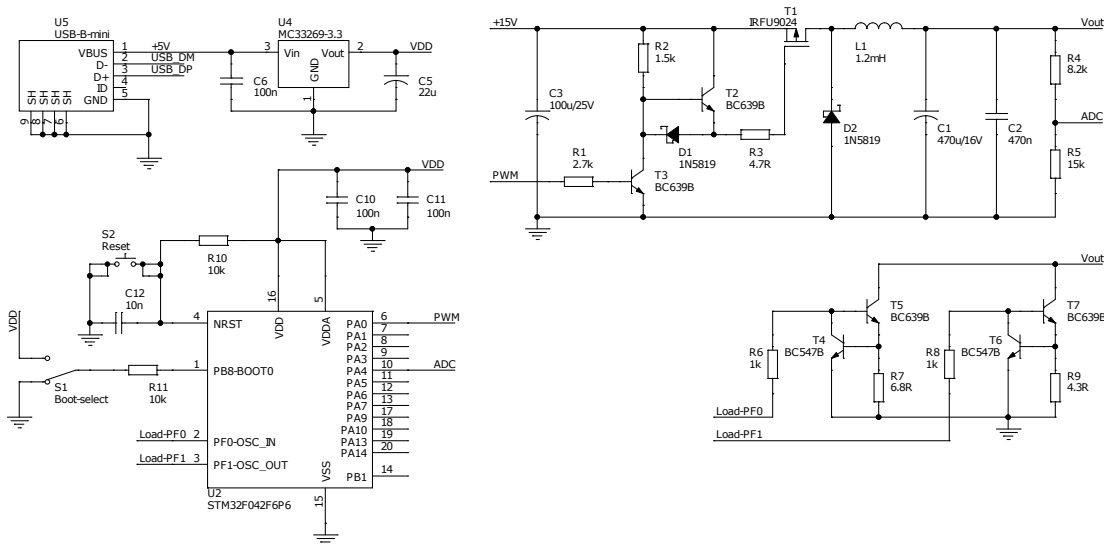
Navržený firmware pro mikrořadiče navíc umožňuje komunikaci po sériové lince nebo po USB s využitím VCP. Součástí je také možnost konfigurovat a ladit regulátor implementovaný na MCU. Demonstrační firmware navíc umožňuje odesílání regulačních dat v reálném čase. Tyto data mohou být vykreslována pomocí Matlabu nebo jiných softwarů pro sběr dat. Pro použití s navrženými regulačními úlohami byla vyvinuta PC aplikace, která umožňuje sběr a vykreslování dat během regulace. Tato aplikace navíc umožňuje konfiguraci samotného regulátoru pomocí jednoduchého uživatelského rozhraní. Jelikož tato aplikace byla nadstavbou oproti zadání této práce, nezbylo k jejímu zdokonalení tolik času. Z toho důvodu je tato PC aplikace stále ještě vyvíjena, nicméně první verze je již otestovaná a plně funkční.

Při ožívování a testování demonstračních úloh regulace byly také vyvinuty dvě pomocné aplikace, UART-USB převodník a PWM generátor. UART-USB převodník byl vyvinut jako alternativa pro komunikaci s PC při paměťově vytíženém procesoru, kde by použití CDC ovladačů nebylo možné. PWM generátor byl použit pro ožívování a testování některých regulačních úloh. Obě zmíněné aplikace jsou již plně funkční, nicméně jsou stále vyvíjeny a zdokonalovány.

# Příloha A

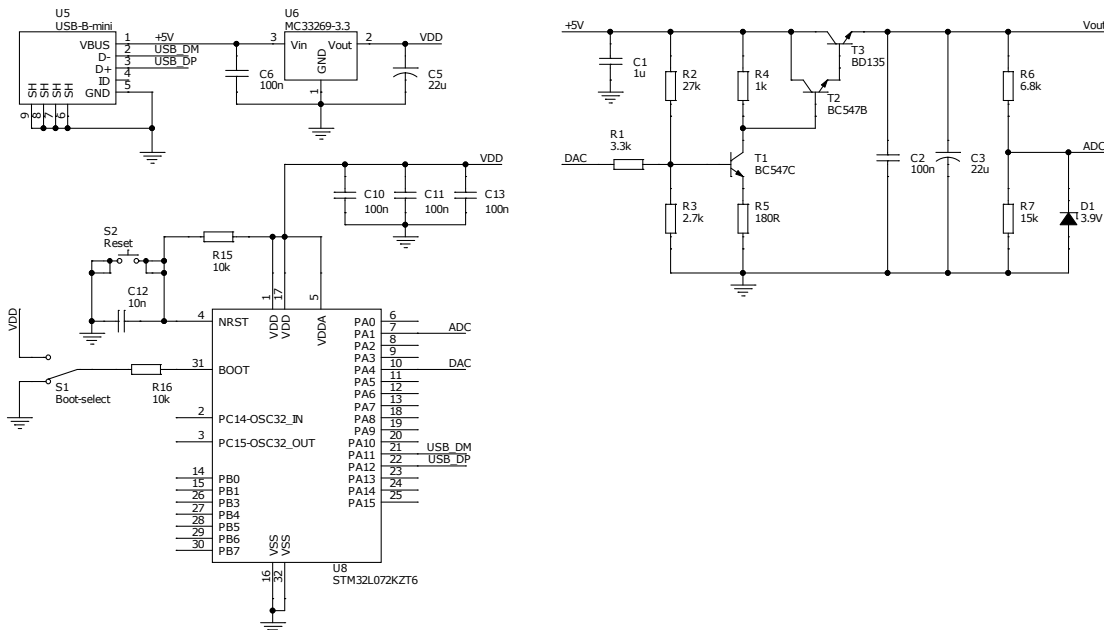
## Schémata regulačních úloh

Na obrázku 48 je schéma zapojení regulátoru napětí se spínaným měničem.



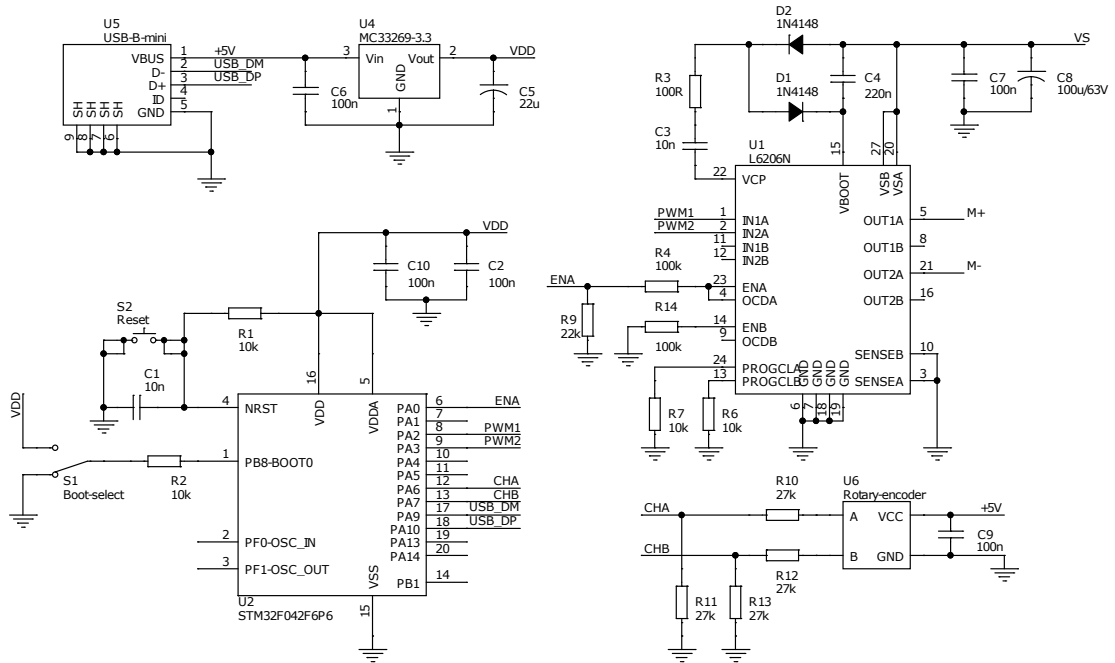
**Obrázek 48** Kompletní schéma zapojení spínaného regulátoru napětí

Na obrázku 49 je schéma zapojení “lineárního” regulátoru napětí, více v kapitole 4.



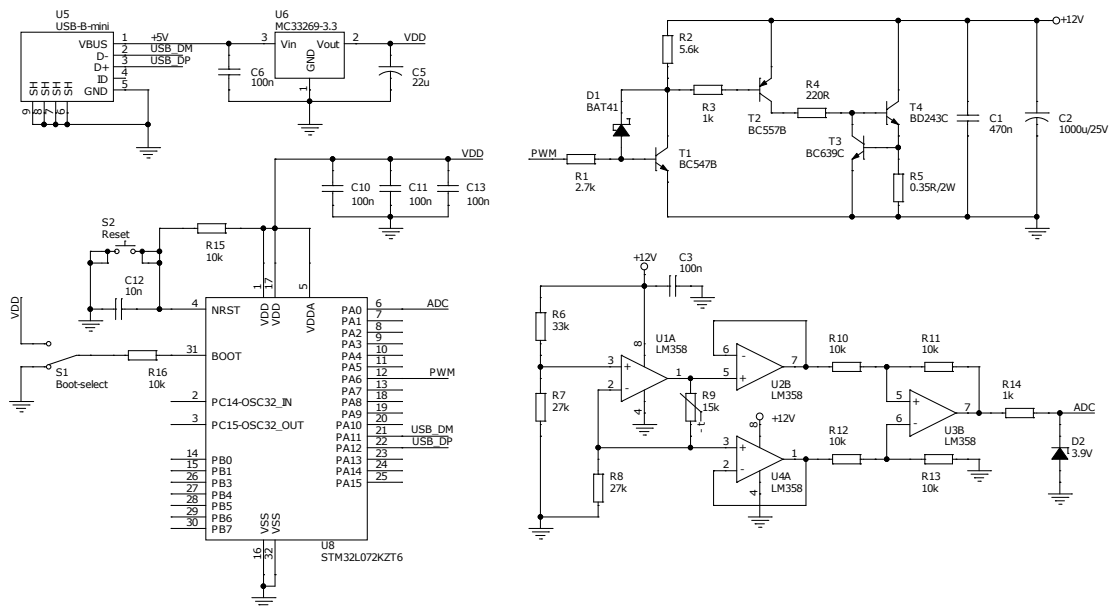
**Obrázek 49** Kompletní schéma zapojení “lineárního” regulátoru napětí

Na obrázku 50 je kompletní schéma zapojení regulátoru polohy představeného v kapitole 5.



**Obrázek 50** Kompletní schéma zapojení regulátoru polohy

Na obrázku 51 je zobrazeno schéma zapojení regulátoru teploty, více v kapitole 6.

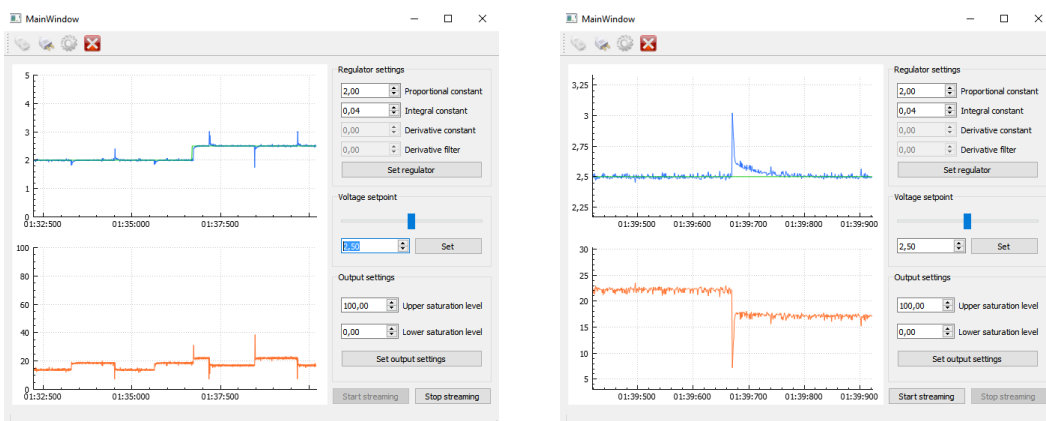


**Obrázek 51** Kompletní schéma zapojení regulátoru teploty

## Příloha B

### Screenshots z demonstrační aplikace

V této příloze jsou k dispozici screenshots z demonstrační PC aplikace. Na obrázku 52 je příklad zaznamenaných průběhů pro regulátor napětí se spínaným měničem.

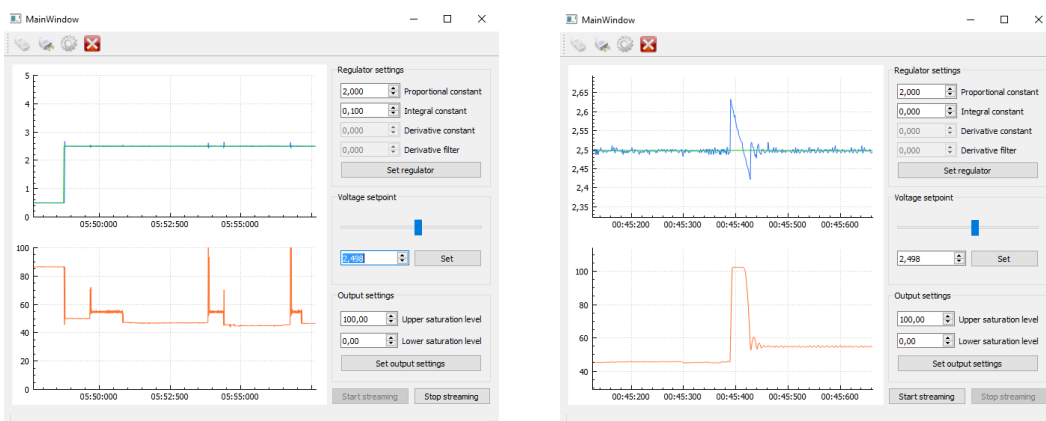


a) Zaznamenaná data z regulace

b) Detail zaznamenaných dat

**Obrázek 52** Ukázka použití demonstrační aplikace pro spínaný regulátor napětí

Na obrázku 53 je záznam průběhů výstupního napětí a akčního zásahu “lineárního” regulátoru napětí.



a) Zaznamenaná data z regulace

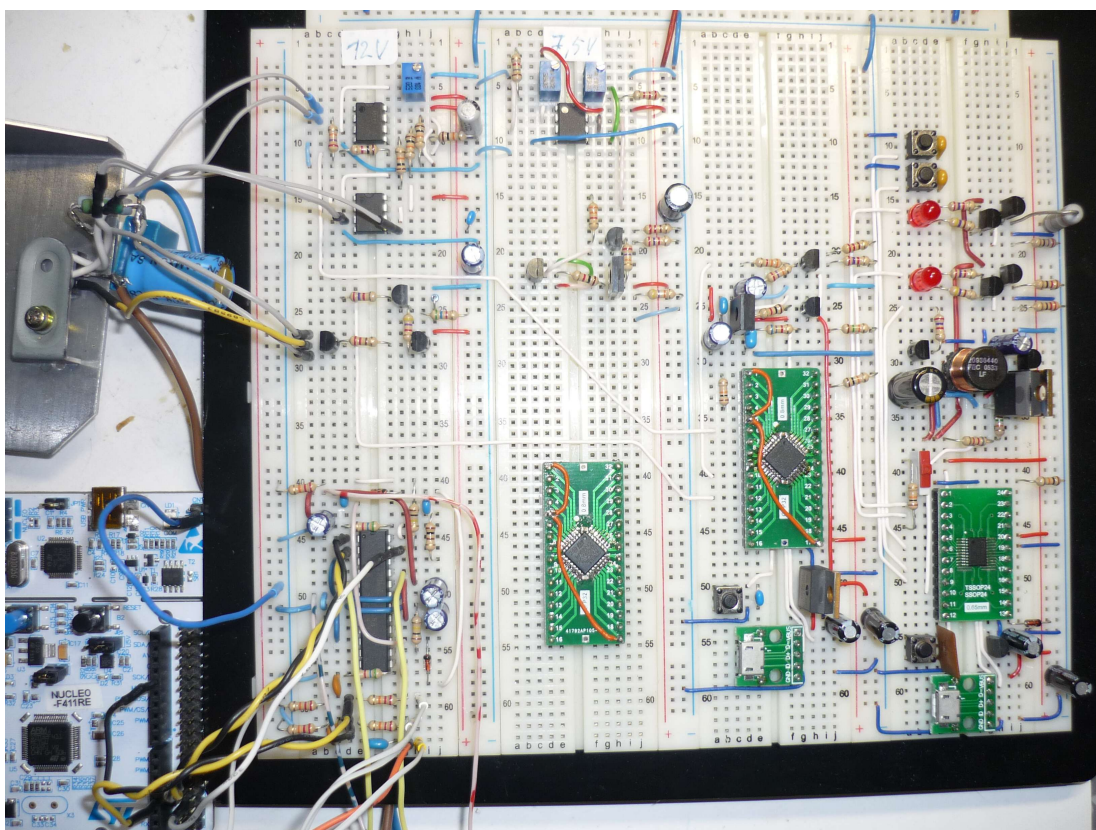
b) Detail zaznamenaných dat

**Obrázek 53** Ukázka použití demonstrační aplikace pro “lineární” regulátor napětí

## Příloha C

### Fotografie realizovaných úloh

Na obrázku 5 je fotografie zapojení všech regulačních úloh na nepájivém kontaktním poli.



Obrázek 54 Fotografie zapojení všech regulačních úloh

## Literatura

- [1] STMicroelectronics. *RM0091. STM32F0x1/STM32F0x2/STM32F0x8 advanced ARM<sup>®</sup>-based 32-bit MCUs*. Čvc 2015. URL: [http://www.st.com/resource/en/reference\\_manual/dm00031936.pdf](http://www.st.com/resource/en/reference_manual/dm00031936.pdf).
- [2] STMicroelectronics. *DS10147. STM32F042x4 STM32F042x6*. Pros. 2015. URL: <http://www.st.com/resource/en/datasheet/stm32f042c4.pdf>.
- [3] STMicroelectronics. *RM0376. Ultra-low-power STM32L0x2 advanced ARM<sup>®</sup>-based 32-bit MCUs*. Lis. 2016. URL: [http://www.st.com/resource/en/reference\\_manual/dm00108281.pdf](http://www.st.com/resource/en/reference_manual/dm00108281.pdf).
- [4] STMicroelectronics. *DS10689. STM32L072x8 STM32L072xB STM32L072xZ*. Břez. 2016. URL: <http://www.st.com/resource/en/datasheet/stm32l072v8.pdf>.
- [5] STMicroelectronics. *RM0383. STM32F411xC/E advanced ARM<sup>®</sup>-based 32-bit MCUs*. Čvc 2014. URL: [http://www.st.com/resource/en/reference\\_manual/dm00119316.pdf](http://www.st.com/resource/en/reference_manual/dm00119316.pdf).
- [6] STMicroelectronics. *DS10314. STM32F411xC STM32F411xE*. Ún. 2015. URL: <http://www.st.com/resource/en/datasheet/stm32f411ce.pdf>.
- [7] STMicroelectronics. *AN4013. STM32 cross-series timer overview*. Čvc 2016. URL: [http://www.st.com/resource/en/application\\_note/dm00042534.pdf](http://www.st.com/resource/en/application_note/dm00042534.pdf).
- [8] STMicroelectronics. *AN4776. General-purpose timer cookbook*. Červ. 2016. URL: [http://www.st.com/resource/en/application\\_note/dm00236305.pdf](http://www.st.com/resource/en/application_note/dm00236305.pdf).
- [9] STMicroelectronics. *AN3116. STM32<sup>TM</sup>'s ADC modes and their applications*. Břez. 2010. URL: [http://www.st.com/resource/en/application\\_note/cd00258017.pdf](http://www.st.com/resource/en/application_note/cd00258017.pdf).
- [10] Thomas Kugelstadt. *The operation of the SAR-ADC based on charge redistribution*. 2005. URL: <http://www.ti.com.cn/cn/lit/an/slyt176/slyt176.pdf>.
- [11] STMicroelectronics. *AN2834. How to get the best ADC accuracy in STM32 micro-controllers*. Ún. 2017. URL: [http://www.st.com/resource/en/application\\_note/CD00211314.pdf](http://www.st.com/resource/en/application_note/CD00211314.pdf).
- [12] STMicroelectronics. *AN4566. Extending the DAC performance of STM32 micro-controllers*. Srp. 2015. URL: [http://www.st.com/resource/en/application\\_note/dm00129215.pdf](http://www.st.com/resource/en/application_note/dm00129215.pdf).
- [13] STMicroelectronics. *UM1724. STM32 Nucleo-64 boards*. Lis. 2015. URL: [http://www.st.com/resource/en/user\\_manual/dm00105823.pdf](http://www.st.com/resource/en/user_manual/dm00105823.pdf).
- [14] STMicroelectronics. *DB2633. Dual brush DC motor driver expansion board based on L6206 for STM32 Nucleo*. Čvc 2015. URL: [http://www.st.com/resource/en/data\\_brief/x-nucleo-ihm04a1.pdf](http://www.st.com/resource/en/data_brief/x-nucleo-ihm04a1.pdf).
- [15] STMicroelectronics. *UM1925. Getting started with the X-NUCLEO-IHM04A1 dual brush DC motor driver expansion board for STM32 Nucleo*. Srp. 2015. URL: [http://www.st.com/resource/en/user\\_manual/dm00215639.pdf](http://www.st.com/resource/en/user_manual/dm00215639.pdf).



- [16] STMicroelectronics. *DS2188. DMOS dual full bridge driver*. Ún. 2014. URL: <http://www.st.com/resource/en/datasheet/16206.pdf>.
- [17] P. Ripka a V. Haasz. *Senzory a měření. Přednášky*. 2015.
- [18] I. Zemanek. *Electronické obvody a prvky. Přednášky*. 2015.
- [19] M. Šebek. *Automatické řízení. Přednášky*. 2015.
- [20] Robert W. Erickson a Dragan Maksimovic. *Fundamentals of Power Electronics. Lectures*.
- [21] J. Fischer. *Mikroprocesory a mikrořadiče v přístrojové technice. Přednášky*. 2015.
- [22] J. Novák a J. Holub. *Distribuované systémy a sítě. Přednášky*. 2015.
- [23] Šimon Fibír. “Programovatelná indikační jednotka pro inkrementální snímače”. Bakalářská práce. ČVUT v Praze, 2016.
- [24] Fukátko Jaroslav, Fukátko Tomáš a Šindelka Jiří. *Teplo a chlazení v elektronice*. 2006. ISBN: 80-7300-199-3.
- [25] STMicroelectronics. *AN4044. Floating point unit demonstration on STM32 micro-controllers*. Květ. 2016. URL: [http://www.st.com/resource/en/application\\_note/DM00047230.pdf](http://www.st.com/resource/en/application_note/DM00047230.pdf).
- [26] Brigitte Hauke. *Basic Calculation of a Buck Converter's Power Stage*. Srp. 2015. URL: <http://www.ti.com/lit/an/slva477b/slva477b.pdf>.
- [27] Christophe Vaucourt. *Choosing Inductors and Capacitors for DC/DC Converters*. Ún. 2004. URL: <http://www.ti.com/lit/an/slva157/slva157.pdf>.
- [28] *Qt Documentation*. 2017. URL: <http://doc.qt.io/>.