

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020

Bc. Roman Sikora



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV RADIOELEKTRONIKY

DEPARTMENT OF RADIOENGINEERING

DC-DC MĚNIČ PRO MATRIX BEAM MODUL

DC-DC CONVERTER FOR MATRIX BEAM MODULE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Roman Sikora

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Lucie Hudcová, Ph.D.

BRNO 2020

Diplomová práce

magisterský navazující studijní obor **Elektronika a sdělovací technika**

Ústav radioelektroniky

Student: Bc. Roman Sikora

ID: 186184

Ročník: 2

Akademický rok: 2019/20

NÁZEV TÉMATU:

DC-DC měnič pro matrix beam modul

POKYNY PRO VYPRACOVÁNÍ:

Navrhněte DC-DC měnič napájející 36-kanálový matrix beam modul. Pro DC-DC měnič a ovládání jednotlivých LED použijte integrované obvody firmy NXP. DC-DC měnič by měl obsahovat dvoufázový vzestupný a tříkanálový sestupný měnič, oba s funkcí Limp Home. Pro řízení měniče a matrix LED controllerů použijte mikroprocesor, případně mikroprocesorový modul.

Na základě návrhu vyrobte prototypový vzorek a realizujte tento DC-DC měnič. Měřením ověřte jeho funkčnost. Součástí výstupu diplomové práce bude naprogramování mikrokontroléru a tvorba vhodného uživatelského rozhraní.

DOPORUČENÁ LITERATURA:

[1] ABOSNINA, Adel Ali a Gerry MOSCHOPOULOS. A Study of Three-Phase DC-DC Boost Converters. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS) [online]. IEEE, 2018, 2018, s. 1-5 [cit. 2018-0-15]. DOI: 10.1109/ISCAS.2018.8350976. ISBN 978-1-5386-4881-0. Dostupné z: <https://ieeexplore.ieee.org/document/8350976/>

[2] MESQUINE, Fouad a Hamza KHALLOUK. Robust control of a DC-DC buck converter. In: 2017 18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA) [online]. IEEE, 2017, 2017, s. 432-437 [cit. 2018-05-15]. DOI: 10.1109/STA.2017.8314873. ISBN 978-1-53-6-1084-8. Dostupné z: <http://ieeexplore.ieee.org/document/8314873/>

Termín zadání: 3.2.2020

Termín odevzdání: 28.5.2020

Vedoucí práce: doc. Ing. Lucie Hudcová, Ph.D.

Konzultant: Ing. Petr Opravil

prof. Ing. Tomáš Kratochvíl, Ph.D.
předseda oborové rady

UPOZORNĚNÍ:

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

ABSTRAKT

Tato diplomová práce se zabývá návrhem buck-boost DC-DC měniče pro napájení matrix beam modulu. Návrh je zaměřen na otestování dvoufázového vztupného měniče a tříkanalového sestupného měniče firmy NXP Semiconductors. Součástí návrhu je implementace mikrokontroléru, kterým je měnič řízen a zároveň je jeho prostřednictvím realizována komunikace s počítačem. Tato práce se rovněž věnuje návrhu uživatelského rozhraní na platformě Windows pro snadnou konfiguraci systému. Součástí návrhu je vytvoření zátěže pro DC-DC měnič. Tuto zátěž lze modifikovat pro dosažení různých proudových odběrů. V diplomové práci je rovněž kladen důraz na vstupní obvody, které zmenšují elektromagnetické rušení do vedení a zvyšují odolnost proti rušení z vedení. Součástí práce je výroba a otestování prototypu.

KLÍČOVÁ SLOVA

zvyšující měnič, snižující měnič, DC-DC, automotive, NXP, ASL2507SHN, ASL3417SHN, LED

ABSTRACT

The master thesis deals with the development of buck-boost DC-DC converter which supplies matrix beam module. The design is focused on testing two-phase boost converter and three channel buck converter manufactured by NXP Semiconductors. Part of the design is implementation of microcontroller for converter control and communication with computer. Part of the thesis is also to design user interface on Windows platform for easy system configuration. Next thing the thesis deals with is designing load for DC-DC converter that is variable and can make different current consumption. One part of this thesis is focused to achieve the lowest conducted emissions and to maximize conducted immunity. Part of this project is production of a prototype and prototype testing.

KEYWORDS

boost converter, buck converter, DC-DC, automotive, NXP, ASL2507SHN, ASL3417SHN, LED

SIKORA, Roman. *DC-DC měnič pro matrix beam modul*. Brno, 2020, 152 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Vedoucí práce: doc. Ing. Lucie Hudcová, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „DC-DC měnič pro matrix beam modul“ jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno

.....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucí práce paní doc. Ing. Lucii Hudcové Ph. D. za trpělivost, aktivitu a cenné rady při realizaci práce. Dále bych rád poděkoval konzultantovi práce panu Ing. Petru Opravilovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora(-ky)

OBSAH

| | |
|--|-----------|
| Úvod | 13 |
| 1 Teorie DC-DC měničů | 15 |
| 1.1 Základní zapojení | 15 |
| 1.1.1 S lineárními obvody | 15 |
| 1.1.2 Bez indukčnosti | 15 |
| 1.1.3 S transformátory | 15 |
| 1.1.4 S cívkami | 15 |
| 1.2 Přehled základních zapojení měničů s cívkami | 16 |
| 1.2.1 Snižující měnič | 16 |
| 1.2.2 Zvyšující měnič | 17 |
| 1.2.3 Zvyšující a snižující měnič | 17 |
| 1.2.4 Invertující měnič | 19 |
| 1.2.5 DCM a CCM | 19 |
| 2 Základní prvky obvodu použité v diplomové práci | 22 |
| 2.1 Popis jednotlivých prvků práce | 22 |
| 2.1.1 ASL2507SHN | 22 |
| 2.1.2 ASL3417SHN | 25 |
| 2.1.3 ATmega128 | 26 |
| 2.1.4 SPI sběrnice | 26 |
| 2.1.5 UART rozhraní | 28 |
| 3 Analýza zapojení | 30 |
| 4 Výpočet jednotlivých částí měniče | 31 |
| 4.1 Snižující měnič | 31 |
| 4.2 Zvyšující měnič | 36 |
| 4.3 Výběr základních součástí | 44 |
| 4.3.1 Tranzistory | 44 |
| 4.3.2 Diody | 48 |
| 4.3.3 Cívky | 48 |
| 5 Simulace jednotlivých stupňů DC-DC měniče | 50 |
| 5.1 Zvyšující měnič | 50 |
| 5.2 Snižující měnič | 53 |

| | | |
|----------|--|-----------|
| 6 | Návrh schématu | 56 |
| 6.1 | Vstupní obvody | 56 |
| 6.1.1 | Vstupní ochrana s filtrací | 56 |
| 6.1.2 | Napájecí napětí digitálních obvodů | 57 |
| 6.1.3 | Převodník mezi USB a UART | 58 |
| 6.2 | Řídící mikrokontroler | 58 |
| 6.3 | Zvyšující měnič | 59 |
| 6.4 | Snižující měnič | 60 |
| 6.5 | Zátěž | 60 |
| 7 | Návrh desky plošného spoje | 62 |
| 7.1 | Zemnicí plocha | 62 |
| 7.2 | Chladicí plochy | 63 |
| 7.3 | Výroba a oživení | 65 |
| 8 | Programové vybavení prototypu | 68 |
| 8.1 | ATmega128 | 69 |
| 8.1.1 | Kooperativní RTOS | 69 |
| 8.1.2 | Hlavní část funkce <i>main()</i> | 71 |
| 8.1.3 | Komunikace s čipy prostřednictvím SPI | 72 |
| 8.1.4 | Ovládání zátěže | 73 |
| 8.1.5 | PWM regulace | 74 |
| 8.1.6 | Sekvence pro ovládání LHM módu | 76 |
| 8.2 | Uživatelské rozhraní | 77 |
| 8.2.1 | Python | 77 |
| 8.2.2 | Základní koncepce | 77 |
| 8.2.3 | Hlavní okno | 78 |
| 8.2.4 | Pokročilé nastavení | 84 |
| 8.2.5 | Řízení zátěže | 86 |
| 8.2.6 | Důležité části kódu | 88 |
| 9 | Měření reálného prototypu | 92 |
| 9.1 | Úvod do měření | 92 |
| 9.1.1 | Řešení počátečních problémů | 92 |
| 9.2 | Konfigurace měničů | 95 |
| 9.3 | Teplotní rozložení | 96 |
| 9.3.1 | Možné příčiny odchylky naměřených hodnot | 98 |
| 9.4 | Měření elektrických veličin | 101 |
| 9.4.1 | Napětí a proudy v obvodu | 101 |
| 9.4.2 | Zvyšující měnič | 103 |

| | | |
|-----------|--|------------|
| 9.4.3 | Snižující měnič | 108 |
| 9.4.4 | Efektivita zapojení | 111 |
| 10 | Závěr | 114 |
| | Literatura | 116 |
| | Seznam zkratk | 119 |
| | Seznam příloh | 120 |
| A | Doplňující výstupy simulací obvodů | 121 |
| A.1 | Zvyšující měnič | 121 |
| A.2 | Snižující měnič | 125 |
| B | Schéma obvodu | 127 |
| B.1 | Převodník mezi USB a UART | 127 |
| B.2 | Zvyšující měnič | 128 |
| B.3 | Snižující měnič | 129 |
| B.4 | Řídící mikrokontroler | 131 |
| B.5 | Zátěž | 132 |
| C | DPS | 134 |
| D | Kritické úseky programu mikrokontroleru ATmega128 | 138 |
| E | Kritické úseky programu uživatelského rozhraní | 144 |
| F | Konfigurace registrů měničů při měření | 147 |
| G | Přiložené soubory | 152 |

SEZNAM OBRÁZKŮ

| | | |
|-----|--|----|
| 1.1 | Schéma snižujícího měniče[3] | 16 |
| 1.2 | Pracovní cykly buck měniče[2] | 16 |
| 1.3 | Schéma zvyšujícího měniče[3] | 17 |
| 1.4 | Pracovní cykly boost měniče[2] | 18 |
| 1.5 | Schéma zvyšujícího a snižujícího měniče[3] | 18 |
| 1.6 | Schéma invertujícího měniče[3] | 19 |
| 1.7 | Schéma zvyšujícího měniče[2] | 19 |
| 1.8 | Porovnání režimů: a)CCM a b)DCM[2] | 20 |
| 2.1 | Pracovní módy a přechody mezi nimi čipu ASL2507SHN[8] | 23 |
| 2.2 | Pracovní módy a přechody mezi nimi čipu ASL3417SHN[9] | 25 |
| 2.3 | Blokové schéma zapojení SPI sběrnice[20] | 27 |
| 2.4 | Vnitřní zapojení SPI sběrnice[20] | 27 |
| 2.5 | Časový diagram SPI komunikace s možnostmi konfigurace[20] | 28 |
| 2.6 | Datový rámeček UART[22] | 28 |
| 3.1 | Blokové schéma DC-DC měniče | 30 |
| 4.1 | Referenční zapojení jednoho výstupního kanálu čipu ASL3417SHN [9] | 31 |
| 4.2 | Limity elektromagnetických emisí do napájecího vedení podle normy CISPR 25, třída 5 [7] | 34 |
| 4.3 | Referenční zapojení jednoho výstupního kanálu čipu ASL2507SHN [8] | 37 |
| 4.4 | Blokové schéma generátoru spínacího kmitočtu vzestupného měniče [8] | 38 |
| 5.1 | Simulační schéma zvyšujícího měniče | 50 |
| 5.2 | Simulace výstupního napětí vzestupného měniče, $U_{IN} = 5,84$ V | 51 |
| 5.3 | Simulace výstupního napětí vzestupného měniče, $U_{IN} = 16,96$ V | 52 |
| 5.4 | Simulační schéma snižujícího měniče | 53 |
| 5.5 | Výstupní napětí snižujícího měniče v intervalu 1 ms po spuštění simulace | 54 |
| 5.6 | Výstupní proud snižujícího měniče v intervalu 1 ms po spuštění simulace | 55 |
| 6.1 | Schéma vstupní ochrany s filtrací | 57 |
| 6.2 | Schéma zapojení lineárního stabilizátoru pro napájení digitálních ob- vodů | 58 |
| 7.1 | Příklad rozložení součástek ve výstupním kanále snižujícího měniče . | 62 |
| 7.2 | Příklad přemostění dlouhých cest dělicími plochami | 63 |
| 7.3 | Příklad chladičích ploch pro součástky | 64 |
| 7.4 | Foto prototypu DC-DC měniče - top | 66 |
| 7.5 | Foto prototypu DC-DC měniče - bottom | 67 |
| 8.1 | Koncepce programového vybavení | 68 |
| 8.2 | Grafické zobrazení zpracování příkazu v mikrokontroleru | 73 |
| 8.3 | Vzhled uživatelského rozhraní - hlavní okno | 78 |

| | | |
|------|---|-----|
| 8.4 | Vzhled uživatelského rozhraní - výřez nastavení BOOST | 79 |
| 8.5 | Vzhled uživatelského rozhraní - výřez nastavení BUCK | 80 |
| 8.6 | Vzhled uživatelského rozhraní - výřez nastavení UART | 82 |
| 8.7 | Vzhled uživatelského rozhraní - pokročilé nastavení | 84 |
| 8.8 | Vzhled uživatelského rozhraní - řízení zátěže | 87 |
| 9.1 | Rozložení teploty na DPS při napájecím napětí 7 V | 96 |
| 9.2 | Rozložení teploty na DPS při napájecím napětí 13,5 V | 96 |
| 9.3 | Detail zúžení přípojného místa cívky k elektrodě drain tranzistoru zvyšujícího měniče | 98 |
| 9.4 | Detail snímku termokamery snižujícího měniče č. 1 pro napájecí na- pětí 13,5 V | 99 |
| 9.5 | Průběh napětí na <i>gate</i> tranzistoru s maximální zátěží, $U_{IN}=6,5$ V . . | 104 |
| 9.6 | Průběh napětí na <i>gate</i> tranzistoru s maximální zátěží, $U_{IN}=13,5$ V . . | 105 |
| 9.7 | Průběh napětí na <i>gate</i> tranzistoru s maximální zátěží, $U_{IN}=17$ V . . . | 105 |
| 9.8 | Průběh napětí na <i>drain</i> tranzistoru s maximální zátěží, $U_{IN}=13,5$ V . | 106 |
| 9.9 | Zvlnění výstupního napětí při maximální zátěží, $U_{IN}=13,5$ V | 107 |
| 9.10 | Náběh výstupního napětí bez zátěže a bez SW náběhu, $U_{IN}=13,5$ V . | 107 |
| 9.11 | Náběh výstupního napětí se zátěží a bez SW náběhu, $U_{IN}=13,5$ V . . | 108 |
| 9.12 | Náběh výstupního napětí se zátěží a se SW náběhem, $U_{IN}=13,5$ V . . | 108 |
| 9.13 | Průběh napětí na elektrodě <i>source</i> tranzistoru kanálu 1, snižujícího měniče 1, při plné zátěži | 108 |
| 9.14 | Průběh napětí na elektrodě <i>source</i> tranzistoru kanálu 2, snižujícího měniče 1, při plné zátěži | 109 |
| 9.15 | Výstupní napětí snižujícího měniče při zvětšování zátěže | 109 |
| 9.16 | Výstupní napětí snižujícího měniče při zmenšování zátěže | 110 |
| 9.17 | Zvlnění výstupního napětí snižujícího měniče | 111 |
| A.1 | Zvlnění napětí na výstupu zvyšujícího měniče, $U_{IN} = 5,84$ V | 121 |
| A.2 | Zvlnění napětí na výstupu zvyšujícího měniče, $U_{IN} = 16,96$ V | 122 |
| A.3 | Výstupní proud vzestupného měniče, $U_{IN} = 5,84$ V | 123 |
| A.4 | Výstupní proud vzestupného měniče, $U_{IN} = 16,96$ V | 123 |
| A.5 | Detail průběhu vstupního proudu vzestupného měniče, $U_{IN} = 5,84$ V | 124 |
| A.6 | Detail průběhu vstupního proudu vzestupného měniče, $U_{IN} = 16,96$ V | 124 |
| A.7 | Výstupní napětí snižujícího měniče v intervalu 10 ms po spuštění simulace | 125 |
| A.8 | Zvlnění napětí na výstupu snižujícího měniče | 126 |
| A.9 | Zvlnění proudu na cívce snižujícího měniče | 126 |
| B.1 | Schéma zapojení převodníku mezi sběrnici USB a rozhraním UART . | 127 |
| B.2 | Schéma zapojení zvyšující části DC-DC měniče | 128 |
| B.3 | Schéma zapojení první ze dvou snižujících částí DC-DC měniče | 129 |

| | | |
|-----|--|-----|
| B.4 | Schéma zapojení druhé ze dvou snižujících částí DC-DC měniče . . . | 130 |
| B.5 | Schéma zapojení řídicího mikrokontroleru <i>ATmega128</i> | 131 |
| B.6 | Schéma zátěže pro první výstupní kanál obou snižujících měničů . . . | 132 |
| B.7 | Schéma zátěže pro druhý výstupní kanál obou snižujících měničů . . | 133 |
| C.1 | DPS prototypu s rozdělením na jednotlivé části | 134 |
| C.2 | DPS vrchní vrstva mědi | 135 |
| C.3 | DPS spodní vrstva mědi | 136 |
| C.4 | DPS rozložení součástek | 137 |
| F.1 | Rozložení NVM paměti zvyšujícího měniče [8] | 149 |
| F.2 | Rozložení NVM paměti snižujícího měniče [9] | 149 |
| G.1 | Stromový diagram obsahu příloženého CD | 152 |

SEZNAM TABULEK

| | | |
|-----|--|-----|
| 4.1 | Shrnutí vypočtených hodnot snižujícího měniče | 36 |
| 4.2 | Shrnutí vypočtených hodnot zvyšujícího měniče | 43 |
| 4.3 | Parametry pro výpočet výkonových ztrát tranzistoru SiR606BDP . . . | 45 |
| 4.4 | Parametry pro výpočet výkonových ztrát tranzistoru Si4056DY . . . | 47 |
| 9.1 | Porovnání naměřených teplot pro různá napájecí napětí | 97 |
| 9.2 | Naměřené napětí na zvyšujícím měniči a vstupních obvodech | 101 |
| 9.3 | Naměřené napětí a proudy na snižujícím měniči č. 1 | 102 |
| 9.4 | Naměřené napětí a proudy na snižujícím měniči č. 2 | 103 |
| 9.5 | Tabulka vypočtených výkonů podle tabulky 9.3 a 9.4 | 104 |
| 9.6 | Tabulka účinností jednotlivých kanálů zvyšujícího měniče | 112 |
| 9.7 | Tabulka účinnosti snižujícího měniče číslo 1, $U_{in}=60,18$ V, $I_{in}=291$ mA | 112 |
| 9.8 | Tabulka účinnosti snižujícího měniče číslo 2 s odpojeným 2. kanálem, $U_{in}=60,22$ V, $I_{in}=183$ mA | 113 |
| 9.9 | Tabulka účinnosti snižujícího měniče číslo 2 s připojeným 2. kanálem, $U_{in}=60,20$ V, $I_{in}=321$ mA | 113 |
| A.1 | Efektivní a střední hodnoty proudu polovodičů, $U_{IN} = 5,84$ V | 121 |
| A.2 | Efektivní a střední hodnoty proudu polovodičů, $U_{IN} = 16,96$ V | 122 |
| A.3 | Efektivní a střední hodnoty proudu polovodičů | 125 |
| F.1 | Konfigurace registrů zvyšujícího měniče při měření - část 1 | 147 |
| F.2 | Konfigurace registrů zvyšujícího měniče při měření - část 2 | 148 |
| F.3 | Konfigurace NVM paměti zvyšujícího měniče | 150 |
| F.4 | Konfigurace registrů snižujícího měniče při měření | 151 |
| F.5 | Konfigurace NVM paměti snižujících měničů | 151 |

ÚVOD

Automobilový průmysl v posledních letech zažívá zlom v oblasti vývoje svítlen. Klasický koncept, kdy byly používány žárovky, se opouští a přechází se na účinnější světelné zdroje. Mezi ně můžeme bezpochyby zahrnout například LED diody (Light Emitting Diode - světlo emitující dioda).

LED světlometry a lampy jsou moderním zdrojem světelného záření s velkou účinností. Jsou tvořeny maticovým zapojením luminiscenčních diod s vysokou svítivostí. Maticové uspořádání umožňuje aktivovat různé kombinace diod, což návrhářům usnadňuje realizovat požadavky zákazníků na vlastnosti svítlen v různých režimech. Tyto systémy mají zvýšené požadavky na napájecí zdroje. Sériové zapojení LED diod zvyšuje požadovanou velikost napájecího napětí svítlny, které je v klasických konceptech automobilového osvětlení nedostupné. Dále zde mohou být různé regulace a proudová omezení napájecích zdrojů pro regulaci intenzity osvětlení. Napájecí zdroje používané v automobilovém průmyslu rovněž umožňují vytvořit na sobě nezávislé napájecí kanály. Ty mohou být snímány a v případě poruchy se svítlna přepne do režimu, kdy fungují pouze její nejdůležitější části. To vše přispívá k bezpečnosti na pozemních komunikacích.

Používání takových světelných zdrojů je opodstatněno několika fakty. Jedním z nich je bezpochyby úspora energie, protože se mnohem menší část elektrické energie mění v teplo. Další předností LED systému, jak již bylo naznačeno, je možnost tvorby různých světelných efektů nebo úpravy vyzařovacího spektra svítlen. To umožňuje v silničním provozu svítit v požadovaných směrech libovolnou intenzitou. Je jasné, že tento systém zvyšuje nároky na napájecí systémy automobilů. Je nutné, aby zdroje elektrické energie pro svítlny byly regulovatelné podle připojené zátěže, byly odolné proti rušení a samy nerušily svoje okolí.

Návrh napájecího zdroje je značnou částí hardwarového vývoje automobilových svítlen. V dnešní době se na trhu objevuje mnoho integrovaných obvodů od různých výrobců, které realizují řízení spínaných zdrojů. Obvody disponují celou řadou diagnostických i řídicích funkcí. Čipy každého výrobce jsou jiné a výrobci svítlen musí otestovat jejich vlastnosti. Otestování integrovaných obvodů společnosti NXP Semiconductors, tedy tvorba demonstračního modelu, je hlavním cílem této diplomové práce. Mělo by být umožněno prostřednictvím uživatelského rozhraní měnit nastavení registrů použitých čipů a umožnit tak sledování chování demonstračního modelu.

Tento projekt je zaměřen na návrh zdroje pro napájení LED modulů. Jedná se o DC-DC (Direct Current-Direct Current - stejnosměrný proud-stejnosměrný proud) měnič typu buck-boost což znamená, že v první fázi se napětí zdroje zvýší a ve druhé se poté sníží na potřebnou hodnotu. Pro realizaci vzestupné i sestupné části mě-

niče jsou použity integrované obvody společnosti NXP Semiconductors. Čipy umožňují řadu funkcí jako jsou řízení parametrů měniče prostřednictvím komunikační sběrnice, možnost regulace výstupních veličin měniče, automatické měření napětí a proudu pro možnou seberegulaci a mnoho dalších.

Cílem této diplomové práce je vytvořit měnič s proměnlivou zátěží z LED diod a tyto funkce demonstrovat.

Vzhledem k faktu, že se jedná o demonstrační model, je důležité, aby řízení registrů integrovaných obvodů měniče bylo uživatelsky přívětivé, a proto je součástí práce tvorba uživatelského rozhraní pro stolní počítač. Vzájemnou komunikaci s počítačem a řízení čipů zajišťuje mikrokontrolér ATmega128.

V současných automobilech je velké množství elektronických zařízení a tento počet bude dále narůstat. Z tohoto důvodu je důležité, aby se zařízení vzájemně nerušily nebo jakkoliv neovlivňovaly. Součástí projektu je proto i návrh vstupních obvodů pro zajištění co možná největší imunity vůči rušení po vedení a zároveň minimalizace emisí do napájecího vedení.

Práce je realizována v následujících krocích:

- První část je věnována seznámení s použitými obvody společnosti NXP. Součástí je i studium technologie DC-DC měničů.
- Další část je věnována výpočtu jednotlivých částí měniče. Zadavatel práce, společnost HELLA AUTOTECHNIK NOVA, definovala požadavky na napájecí zdroj. Podle těchto parametrů jsou provedeny výpočty součástek pro vzestupnou a následně i sestupnou část měniče.
- Následně, podle vypočtených hodnot, jsou vybrány diskrétní součástky, které budou použity pro realizaci demonstračního modelu v diplomové práci. Rovněž je upřesněn způsob řízení měniče a komunikace s počítačem. Na tomto základě je poté navrženo samotné schéma DC-DC měniče.
- V rámci diplomové práce jsou jednotlivé části schématu odsimulovány a zároveň jsou vypočteny teplotní ztráty součástek.
- Dalším krokem je návrh desky plošného spoje s ohledem na pracovní podmínky měniče a výkony vyskytující se v obvodu.
- Součástí diplomové práce je výroba navrženého modelu.
- Následuje část oživení prototypu.
- Další část práce se věnuje programovému vybavení pro prototyp zdroje, jedná se o software pro řídicí mikrokontroler a o uživatelské rozhraní na počítači.
- Závěrem práce je otestování funkčnosti prototypu a proměření základních parametrů.

1 TEORIE DC-DC MĚNIČŮ

Pod pojmem DC-DC měnič je možné si představit obvod využívaný pro regulaci elektrické energie. Tento obvod mění vstupní stejnosměrné napětí o jisté velikosti na jiné výstupní napětí podle typu měniče na vyšší či nižší hodnotu [1].

1.1 Základní zapojení

DC-DC měniče bychom mohli rozdělit na následující čtyři skupiny.

1.1.1 S lineárními obvody

Jedná se o skupinu měničů s lineárními stabilizátory. Může se tedy jednat o parametrické nebo zpětnovazební stabilizátory. Tyto obvody jsou charakterizovány tím, že dokáží na svůj výstup dodat pouze nižší napětí než je na jejich vstupech. U těchto zapojení nedochází ke galvanickému oddělení vstupů a výstupů [1], [2].

1.1.2 Bez indukčnosti

Tuto skupinu by šlo rovněž nazvat jako měniče se spínanými kondenzátory. Používají se zejména jako zdroje pro malé proudy. Opět se zde nachází galvanické spojení vstupu a výstupu. Tyto obvody se rovněž nazývají nábojové pumpy a jsou žádané díky své nízké ceně a jednoduchosti [1], [2].

1.1.3 S transformátory

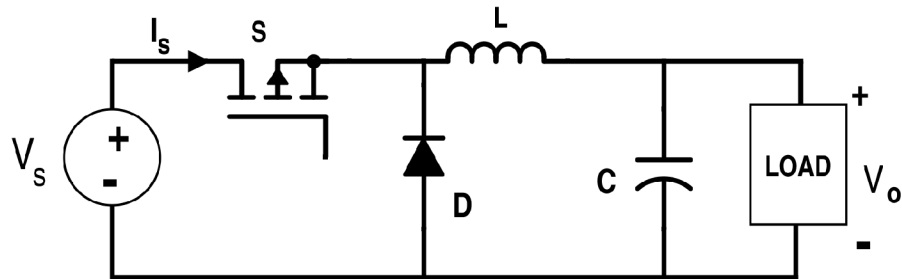
Měniče s transformátory mají jako jediné galvanicky oddělené výstupní napětí od vstupního. Umožňují vytvoření jednoho výstupního napětí nebo dvou symetrických či nesymetrických napětí a podobně [1], [2].

1.1.4 S cívkami

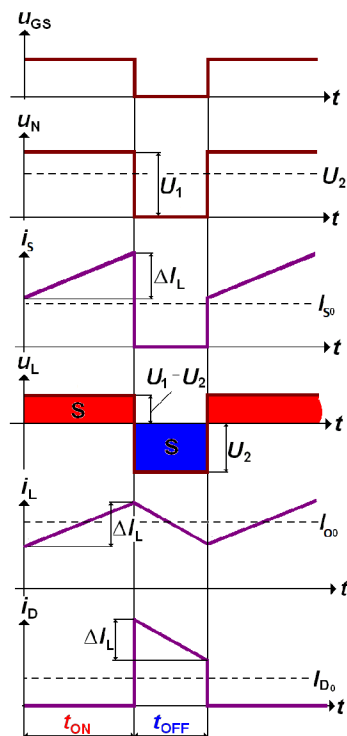
Tato skupina převzala princip klasických spínaných zdrojů, měniče obsahují vždy cívku a mají galvanicky spojený vstup s výstupem. Umožňují zvyšovat (boost), snižovat (buck), invertovat či zvyšovat i snižovat (cuk) výstupní napětí. Díky své poměrně vysoké účinnosti jsou v dnešní době používané pro napájení širokého spektra elektronických zařízení. Měniče s cívkami jsou v této práci rozebrány podrobněji [1], [2].

1.2 Přehled základních zapojení měničů s cívkami

1.2.1 Snižující měnič



Obr. 1.1: Schéma snižujícího měniče[3]



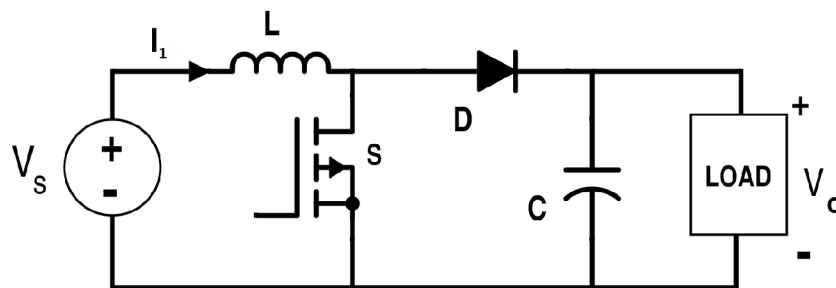
Obr. 1.2: Pracovní cykly buck měniče[2]

Na obrázku 1.1 je zobrazeno základní schéma snižujícího měniče, který bývá v literatuře označen rovněž jako step-down nebo buck. Toto zapojení funguje ve dvou cyklech. V průběhu prvního cyklu je spínač **S** sepnut, proud teče v této fázi ze zdroje **V_S** přímo do zátěže **LOAD**. V cívce **L** se akumuluje energie a stejně tak se nabíjí

kondenzátor **C**. Cívka se při sepnutí spínače chová jako spotřebič a napětí na ní má stejný směr jako proud, který přes ní protéká, v tomto případě I_S .

Do druhého cyklu vstoupí obvod po rozepnutí spínače **S**. Nyní se cívka začne chovat jako zdroj napětí, což znamená, že se polarita napětí obrátí oproti předchozímu případu, ale proud zachová svůj směr. Nyní je tedy zátěž napájena energií z cívky, která je zároveň podpořena energií naakumulovanou v kondenzátoru, jenž se v tomto cyklu do zátěže vybíjí. Z této úvahy je jasné, že zvětšující se kapacitou kondenzátoru **C** se zmenšuje zvlnění výstupního napětí V_O . Vzhledem k tomu, že je spínač **S** rozepnut, je smyčka v této fázi uzavřena přes diodu **D**. Pro lepší názornost jsou pracovní cykly zobrazeny graficky na obrázku 1.2 [1], [2], [3].

1.2.2 Zvyšující měnič

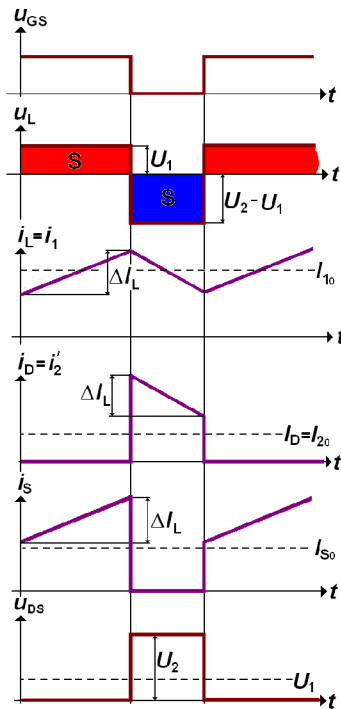


Obr. 1.3: Schéma zvyšujícího měniče[3]

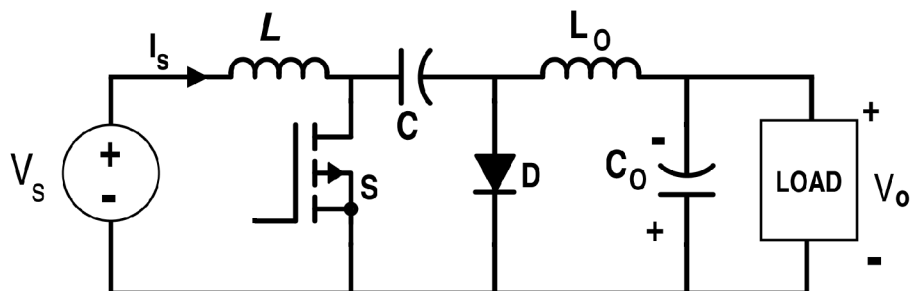
Dalším typickým zapojením je zvyšující měnič nebo taky boost či step-up. Jeho zapojení je možné vidět na obrázku 1.3. Měnič opět funguje ve dvou fázích. Během první je spínač **S** sepnut a proud tedy protéká cívkou **L** a spínačem **S**. Energie se akumuluje v magnetickém poli cívky. Cívka tedy v obvodu vystupuje jako spotřebič. Účelem diody **D** je zabránit vybíjení kondenzátoru **C** přes sepnutý spínač. Kondenzátor je zdrojem energie pro zátěž **LOAD** ve fázi, kdy je spínač **S** sepnut. Po jeho rozepnutí se cívka **L** změní ze zátěže na zdroj energie, který je spojen se zdrojem V_S sériově, což má za následek, že výstupní napětí V_O je vždy vyšší než vstupní napětí V_S . Pro lepší názornost jsou pracovní cykly opět zobrazeny graficky na obrázku 1.4 [1].

1.2.3 Zvyšující a snižující měnič

Tento typ měniče, označovaný také jako cuk, umí napětí zvýšit i snížit. Díky své funkci se používá pro stabilizaci napětí, takže udržuje konstantní výstupní napětí i když na vstupu dochází k výkyvům napětí. Princip funkce se dělí do třech intervalů.



Obr. 1.4: Pracovní cykly boost měniče[2]



Obr. 1.5: Schéma zvyšujícího a snižujícího měniče[3]

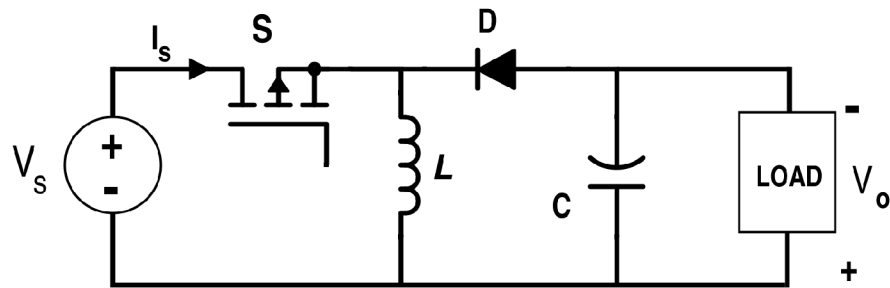
V prvním, po zapnutí napájecího napětí V_S , je spínač S rozepnut, proud teče přes cívku L , kondenzátor C a diodu D . Kondenzátor se tedy nabíjí na napětí dané součtem napětí zdroje V_S a napětí na cívce L , která se chová jako zdroj, nebo na napětí nižší v závislosti na časování spínače S .

Druhý interval je zahájen sepnutím spínače S . Z cívky L se stane spotřebič a začne akumulovat energii do svého magnetického pole. Současně je spojena kladná svorka kondenzátoru C se zemí, a ten se vybíjí do obvodu. Dochází k nabíjení kondenzátoru C_O a hromadění energie v magnetickém poli L_O .

Třetí interval je stejný jako první, nicméně se energie uložená v cívce L_O šíří do obvodu. Proud teče přes diodu D a zátěž $LOAD$.

Je vidět, že výstupní napětí má obrácenou polaritu oproti napětí vstupnímu [1], [3].

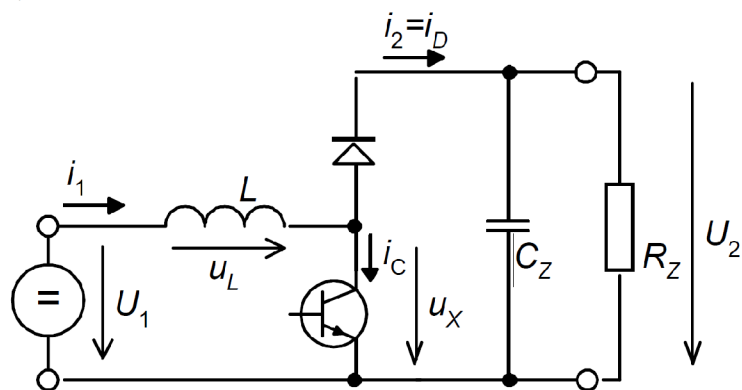
1.2.4 Invertující měnič



Obr. 1.6: Schéma invertujícího měniče[3]

Tento měnič se používá pro změnu polarity napětí. V první fázi kdy je spínač S sepnut, se cívka L chová jako spotřebič. Má tedy směr proudu a napětí shodný a nabíjí se. Když se spínač S rozpojí, tak se cívka L začne chovat v obvodu jako zdroj napětí. To znamená, že se na ní polarita napětí obrátí [1], [2].

1.2.5 DCM a CCM

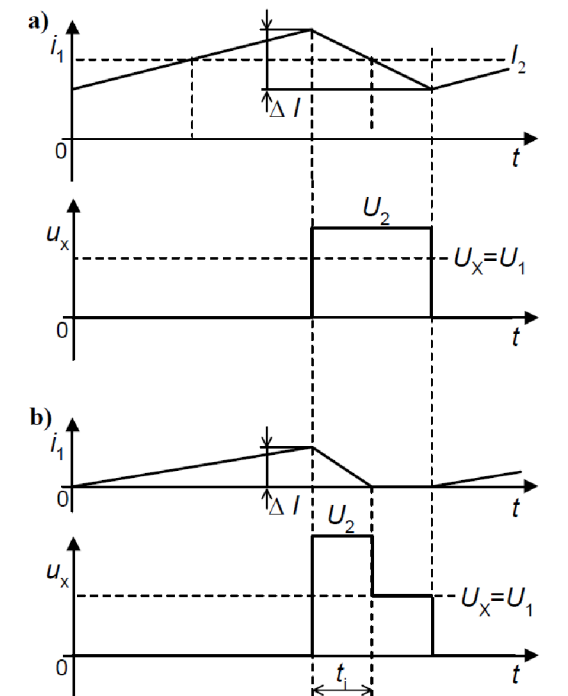


Obr. 1.7: Schéma zvyšujícího měniče[2]

V této sekci jsou popsány dva módy, ve kterých mohou měniče napětí pracovat. Jedná se o módy DCM (Discontinuous Current Mode - režim nespojitých proudů) a režim CCM (Continuous Current Mode - režim spojitých proudů). Režimy se rozlišují podle toho, jakých hodnot dosahuje proud cívky v obvodu. Vzhledem k tomu, že je to proud cívky, bude se jednat u zvyšujícího měniče o proud vstupní a u snižujícího měniče o proud výstupní. Princip módů je vysvětlen na schématu boost měniče z obrázku 1.7.

Pokud je tranzistor **T** rozepnutý, víme, že přes diodu a cívku teče do zátěže proud. Ten klesá a na cívce tvoří úbytek napětí. Z principu funkce boost měniče musí být úbytek záporný, protože napětí U_2 je větší než U_1 . Po sepnutí tranzistoru se na cívce objeví kladné napětí U_1 , proud cívky roste a dioda **D** je uzavřena. Při opětovném rozepnutí tranzistoru se cyklus opakuje.

Oba popisované módy mají totožný princip činnosti. Liší se však středním proudem cívky, kdy u spojitého módu je střední proud cívky I_1 nenulový v průběhu celého cyklu. V režimu přerušovaných proudů však střední hodnota proudu I_1 je tak malá, že v průběhu doby rozepnutí tranzistoru proud i_1 klesne na nulu. V tom okamžiku se zavře i dioda **D**, takže v obvodu nevede ani tranzistor **T** ani dioda **D**. To způsobí, že zdroj je od obvodu zcela odpojen. Na svorkách u_x se objeví napětí U_1 namísto U_2 , protože cívku neteče proud, a jak je známo úbytek napětí na cívce je přímo úměrný časové změně proudu cívky.



Obr. 1.8: Porovnání režimů: a)CCM a b)DCM[2]

Zmíněné napětí u_x musí mít v obou režimech stejnou střední hodnotu. Důvodem je to, že na cívce je nulová střední hodnota úbytku napětí. To směřuje na problém, že U_2 se zvětšuje a v případě zátěže naprázdno by teoreticky rostlo nade všechny meze. Tento problém je názorně zobrazen na obrázku 1.8. Podrobněji jsou tyto režimy popsány v literatuře [2].

2 ZÁKLADNÍ PRVKY OBVODU POUŽITÉ V DIPLOMOVÉ PRÁCI

V současnosti jsou na trhu k dostání integrované obvody, které zajišťují řízení (spínání) měničů. Tyto obvody značně zjednodušují návrh oproti použití diskretních součástek. Integrované obvody zajišťují správné časování, které upravují podle okamžitých naměřených hodnot, ať už výstupních či vstupních napětí nebo jiných veličin, které si samy obstarají. Řízení těchto integrovaných obvodů se poté realizuje pomocí mikrokontroleru prostřednictvím sběrnic, kdy si uživatel nastaví řadu požadovaných parametrů a měnič už poté pracuje sám.

Úkolem je využít dvoufázový vzestupný měnič *ASL2507SHN* a tříkanálový sestupný měnič *ASL3417SHN*. Tyto čipy jsou vyrobeny společností NXP Semiconductors a v diplomové práci jsou využity pro tvorbu měniče s topologií boost-buck.

Řízení je realizováno ze stolního počítače, ten komunikuje s mikrokontrolerem *ATmega128* prostřednictvím sběrnice UART. Mikrokontroler rovněž obstarává komunikaci prostřednictvím SPI sběrnice s integrovanými obvody společnosti NXP. Poslední funkcí ATmegy je řízení zátěže.

2.1 Popis jednotlivých prvků práce

2.1.1 ASL2507SHN

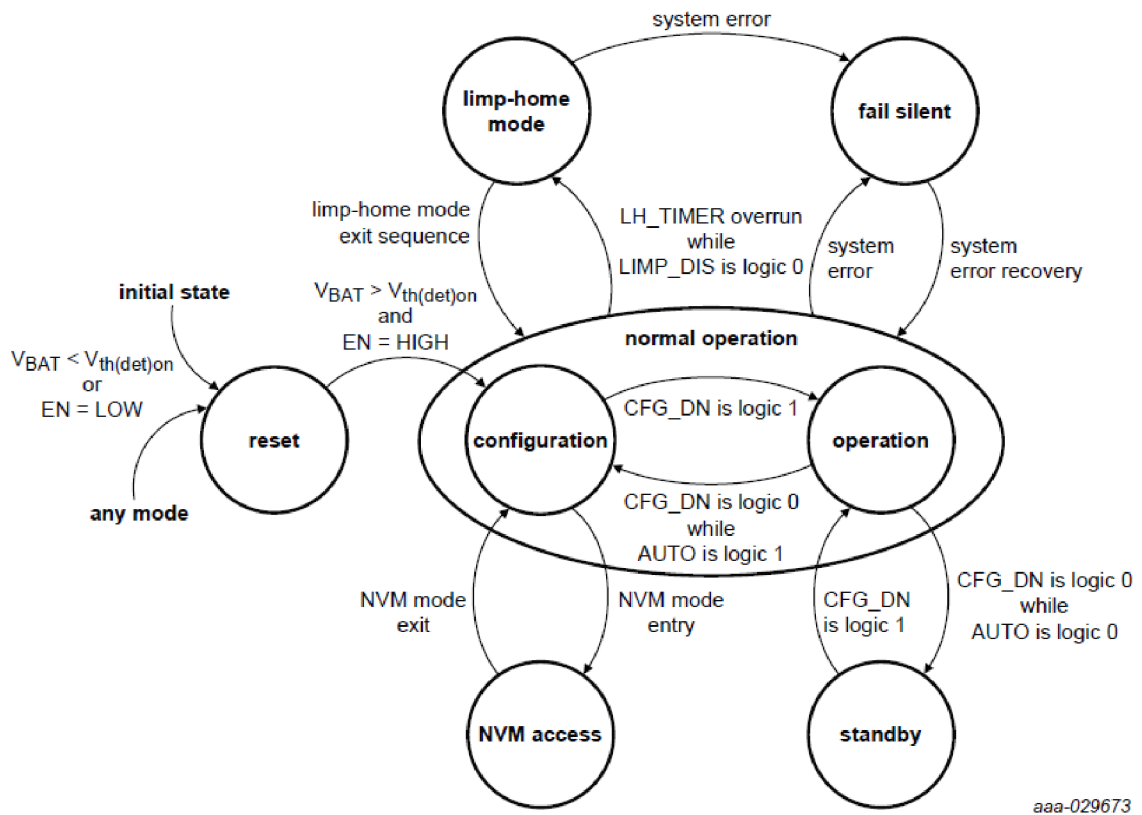
Jedná se o integrovaný, vzestupný, dvoufázový DC-DC (Direct Current-Direct Current - stejnosměrný proud-stejnosměrný proud) měnič. V čipu jsou integrovány dvě nezávislé brány pro ovládání externích MOSFET (Metal Oxide Semiconductor Field Effect Transistor) tranzistorů použitých jako spínače v zapojení vzestupného měniče. Čip obsahuje obvody pro snímání výstupních napětí a dokáže tato napětí nezávisle snímat u obou měničů připojených na zmíněných bránách [8].

Pro řízení a čtení diagnostických informací externím mikrokontrolerem podporuje čip komunikaci prostřednictvím SPI (Serial Peripheral Interface - seriové periferní rozhraní) sběrnice [8].

Měnič pracuje na fixní frekvenci, která může být v rozsahu 125 až 700 kHz. Vstupní napětí může být v rozsahu 5,5 až 40 V, z čehož měnič dokáže vytvořit až 80 V. Čip je možné nakonfigurovat tak, aby vytvořil pouze jedno výstupní napětí s jednou či dvěma fázemi nebo je možné nakonfigurovat dva nezávislé kanály. V případě chyby se čip automaticky přepne do příslušného pracovního módu [8].

Integrovaný obvod má širokou škálu funkcí pro ochranu před abnormálními jevy v obvodu. Lze nastavit přepětovou ochranu napájecího napětí i ochranu před pod-

pětím. Dále jsou zde ochrany pro limitaci proudu cívkou měniče, teplotní ochrany anebo je možné ovlivňovat frekvenční spektrum výstupního napětí [8].



Obr. 2.1: Pracovní módy a přechody mezi nimi čipu ASL2507SHN[8]

Zvláštností tohoto čipu je speciální pracovní mód *Limp-home mode*. Tento mód je zavedený z bezpečnostních důvodů v automobilovém průmyslu. V případě jeho aktivace se čip nastaví do potřebné konfigurace pro základní provoz například svítily [8].

Jak již bylo zmíněno, *ASL2507SHN* dokáže pracovat v několika módech, mezi kterými lze přecházet v průběhu funkce čipu. Grafické znázornění módů a přechodů mezi nimi lze vidět na obrázku 2.1.

Reset mode

Do tohoto módu se obvod dostane v případě, že napájecí napětí obvodu klesne pod rozhodovací úroveň 4,5 V nebo je pin *EN* je v nízké logické úrovni. V *Reset mode* čip odpojí výstup měniče a zároveň přeruší SPI komunikaci. Napětí na pinu *VGG* je vypnuto [8].

Configuration mode

Do *Configuration mode* přejde zařízení z *Reset mode* ihned po překročení rozhodovací úrovně napájecího napětí a po přivedení vysoké logické úrovně na pin *EN*. V tomto módu lze konfigurovat vnitřní registry prostřednictvím SPI. Výstup je odpojen a na pinu *VGG* je napětí v závislosti na předchozím nastavení registrů [8].

Standby mode

Zařízení vstoupí do *Standby mode* z *Operation mode* v případě, že bit *CFG_DN* v interním registru je nastaven na logickou nízku úroveň v době, kdy bit *AUTO* je rovněž v logické nule. Výstupy jsou v tomto módu odpojeny a napětí na pinu *VGG* je podle předchozího nastavení interních registrů [8].

Operation mode

Pokud se nastaví bit *configuration done*, zařízení přejde z *Configuration mode* do *Operation mode*. Poté již není možné měnit důležité konfigurační registry. Napětí na pinu *VGG* je zafixováno a již nelze měnit [8].

Fail silent mode

Do tohoto módu se může čip dostat z *Operation mode*, *Configuration mode*, *Standby mode* nebo *Limp-home mode*. Stane se tak, pokud teplota čipu překročí maximální dovolenou hodnotu. Vznikne chyba na pinu *VGG*, vstupní napětí klesne pod nebo nad nastavené meze. Chování je stejné jak v *Reset mode* až na SPI komunikaci, která funguje bez omezení [8].

Limp-home mode

Tento mód se aktivuje v případě ztráty SPI komunikace. Poté dojde k bezpečnostní konfiguraci zařízení z interní *NVM* paměti čipu. V tomto módu lze konfigurovat pouze *limp-home mode control register*. Nečinnost SPI sběrnice čip odhalí, pokud nadřazený mikrokontroler přestane nulovat interní *LHM* čítač. Vystoupit z tohoto módu lze pouze zadáním speciální sekvence příkazů [8].

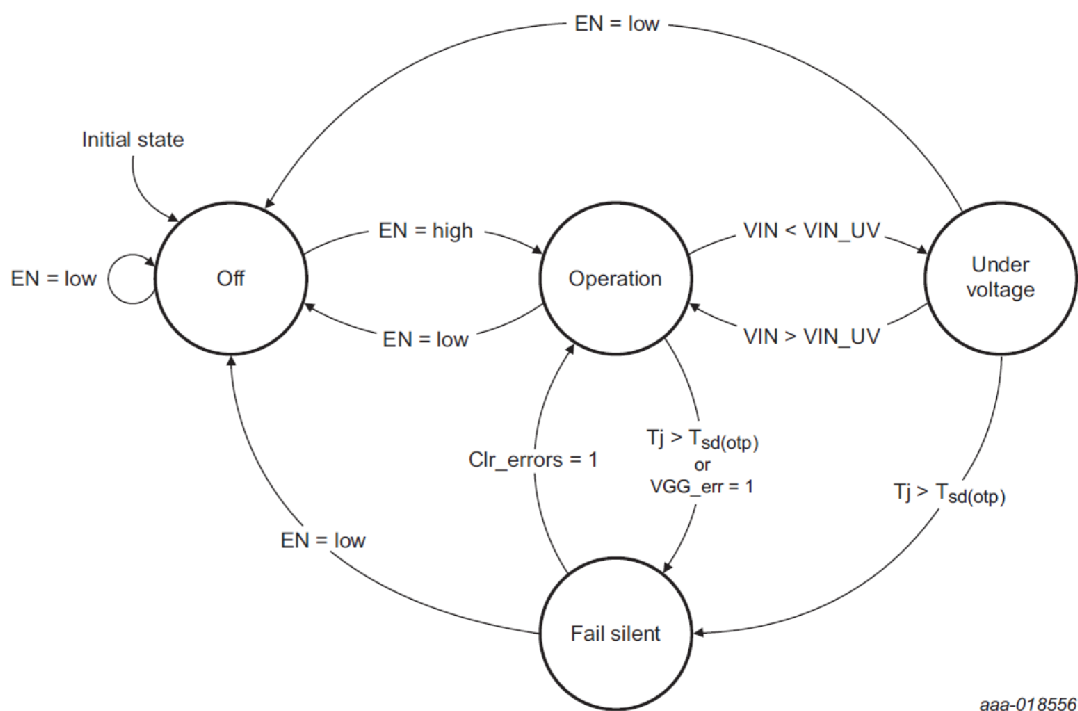
NVM access mode

Do tohoto módu se obvod dostane v případě, že byl zadán speciální příkaz pro přístup. Pouze v tomto módu lze konfigurovat nastavení pro *limp-home mode* v *NVM* paměti [8].

2.1.2 ASL3417SHN

Integrovaný obvod *ASL3417SHN* je tříkanálový, sestupný DC-DC měnič s hysterezí určený pro napájení LED (Light Emitting Diode - světlo emitující dioda) zařízení. Čip dokáže vytvořit tři nezávislé kanály s proudovým odběrem od 30 mA do 1,5 A s tím, že pokud je zapotřebí většího proudu, je nutné spojit dva kanály dohromady. Každý kanál opět nezávisle ovládá vnější MOSFET tranzistor. Velikost ovládacího napětí je možné vnitřně regulovat.

Výstupní proud lze regulovat nastavením vnitřních registrů s přesností pěti procent ve dvou rozsazích. Zde se opět nabízí možnost řízení a čtení čipu prostřednictvím SPI sběrnice.



Obr. 2.2: Pracovní módy a přechody mezi nimi čipu ASL3417SHN[9]

Vstupní napětí obvodu se může pohybovat od 10 do 80 V. Obvod monitoruje napětí na vstupu i na výstupu a opět obsahuje ochrany proti přepětí, podpětí i vysoké teplotě. Na čipu jsou tři PWM (Pulse Width Modulation - pulzně šířková modulace) vstupy pro stmívání jednotlivých kanálů.

Tento obvod může pracovat v několika módech. Jejich grafické znázornění je na obrázku 2.2.

Off mode

Do tohoto módu vstoupí zařízení v případě nízké logické úrovně na pinu *EN*. Všechny výstupy i SPI komunikace jsou poté neaktivní [9].

Operation mode

Zařízení se přepne z *Off mode* do tohoto módu, pokud je napětí na vstupu nad prahovou úrovní, a zároveň pokud je pin *EN* v logické jedničce. V tomto módu jsou všechny výstupy včetně SPI komunikace aktivní [9].

Undervoltage mode

Jedná se o mód, ve kterém se zařízení ocitne v případě, že vstupní napětí klesne pod danou naprogramovanou prahovou úroveň. Pokud se tak stane, jsou poté všechny výstupy vypnuté [9].

Fail silent mode

Do *Fail silent mode* se zařízení dostane, pokud jeho teplota překročí maximální povolenou prahovou teplotu nebo pokud je detekována chyba na pinu *VGG*. V popísaném módu jsou všechny výstupy neaktivní kromě SPI komunikace, která zůstává zachována [9].

2.1.3 ATmega128

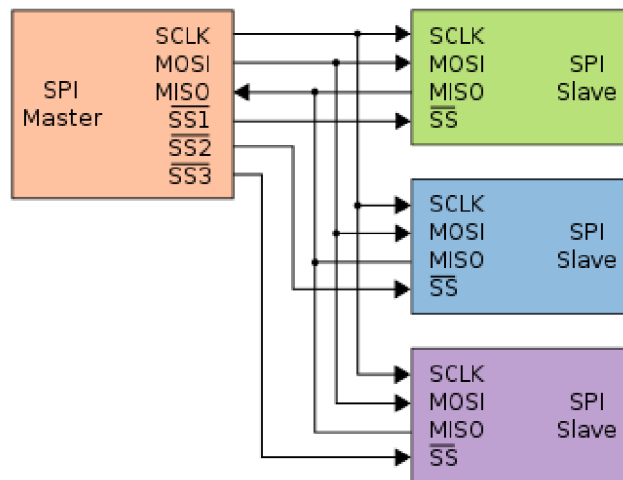
ATmega128 je mikrokontroler z rodiny AVR vyvíjené společností Atmel od roku 1996. Jedná se o mikrokontroler s Harvardskou architekturou, má tedy fyzicky oddělený úložný prostor pro data a instrukce. Obsahuje 128 kB programovou flash paměť, 4 kB paměť SRAM a 4 kB paměť EEPROM. Je možné použít až 53 programovatelných vstupně výstupních pinů. Mikrokontroler disponuje dvěma 8 bitovými čítači a dvěma 16 bitovými [17], [18].

Mikrokontrolery rodiny AVR jsou dnes hojně používány v oblasti studentských projektů a výuky hlavně kvůli jejich jednoduchosti a dostupnosti a zároveň jsou srdcem konceptu Arduino, který má velikou základnu u veřejnosti. Důvodem použití právě tohoto mikrokontroleru v diplomové práci je zejména předešlá zkušenost s jeho programováním, ale zároveň jeho nízká cena, která se pohybuje v řádu stovek korun.

2.1.4 SPI sběrnice

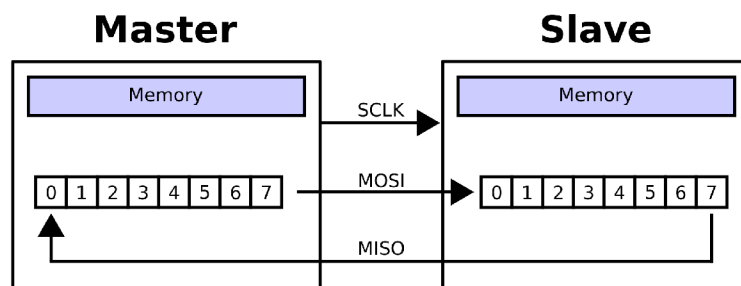
Základním prvkem komunikace v měničích je SPI sběrnice, která byla vyvinutá společností Motorola v osmdesátých letech dvacátého století. Jejím prostřednictvím

jsou propojeny čipy *ASL2507SHN* a *ASL3417SHN* s mikrokontrolerem *ATmega128*. Jedná se o jednoduchou sériovou čtyř-vodičovou synchronní sběrnici. Komunikace probíhá v konfiguraci *Master-Slave*, to znamená, že zde je více zařízení v pozici *slave* a jedno zařízení v pozici *master* (blokové schéma popisuje obrázek 2.3).



Obr. 2.3: Blokové schéma zapojení SPI sběrnice[20]

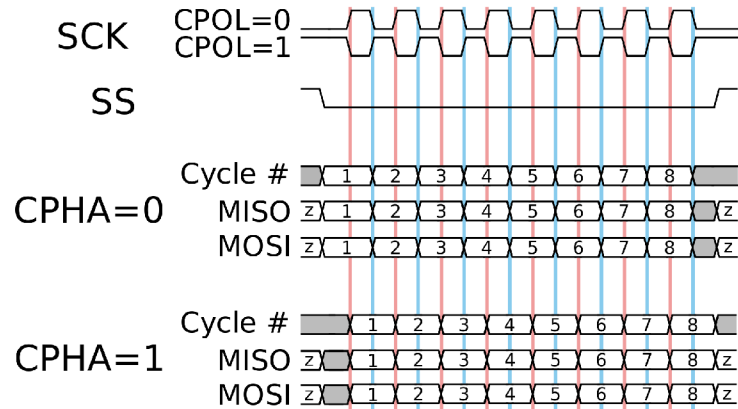
Master řídí komunikaci a určuje signálem SS (Slave Select - výběr podřizovaného zařízení), s kterým podřizovaným zařízením se bude komunikovat, a zároveň určuje frekvenci hodinového signálu komunikace. Komunikace probíhá po vodičích MOSI (Master Output Slave Input - směr dat z nadřizovaného do podřizovaného zařízení) a MISO (Master Input Slave Output - směr dat z podřizovaného do nadřizovaného zařízení) v plném duplexu, tedy oběma směry zároveň. Principiální funkce je velmi jednoduchá, což je zřejmě důvodem velkého využití sběrnice v praxi.



Obr. 2.4: Vnitřní zapojení SPI sběrnice[20]

Jak je vidět na obrázku 2.4, jedná se o dva sériově zapojené posuvné registry.

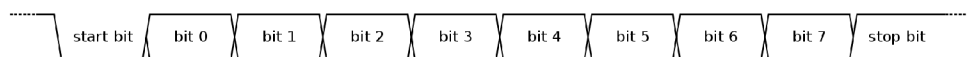
První se nachází v zařízení *master* a druhý v *slave*. Při hraně hodinového signálu dochází tedy k vysouvání jednotlivých bitů z jednoho zařízení do druhého. SPI sběrnice nepoužívá žádná redundantní data, nejsou zde ani kontrolní součty ani potvrzovací, startovací či ukončovací bity. Jediná možná změna průběhu komunikace spočívá v nastavení klidové úrovně a aktivní hrany hodinového signálu. Možné konfigurace jsou patrné z obrázku 2.5 [20], [21].



Obr. 2.5: Časový diagram SPI komunikace s možnostmi konfigurace[20]

2.1.5 UART rozhraní

UART (Universal Asynchronous Receiver and Transmitter - univerzální asynchronní seriové rozhraní) je počítačové komunikační rozhraní. Zpravidla se jedná o asynchronní dvou vodičovou konfiguraci, kdy dvě zařízení spolu mohou komunikovat v plném duplexu. V případě použití modifikované synchronní verze USART probíhá komunikace spolehlivěji, ale je možná jen v polovičním duplexu, tedy v jeden okamžik komunikuje pouze jedno rozhraní.



Obr. 2.6: Datový rámeček UART[22]

Datový rámeček obsahuje start bit, jeden nebo více stop bitů, datové bity a případně paritní bit. Ve většině aplikací se vysílá nejméně významný bit jako první.

Rozhraní je v klidovém stavu ve vysoké logické úrovni. Start bit má nízkou úroveň, dalších pět až devět bitů je odesílaný znak a následuje parita. Další jeden

nebo dva bity jsou ve vysoké logické úrovni a reprezentují konec vysílání. Nízká úroveň start bitu a vysoká úroveň stop bitu zajišťuje minimálně dvě úrovně změny mezi odesílanými znaky. Pokud je na lince nízká úroveň déle než trvá doba odesílání jednoho znaku, je možné detekovat chybu *Break condition*. Na rozhraní UART je možné detekovat ještě následující chyby.

Overrrun error - Pokud přijímač začne přijímat znak, ale ve vstupním přijímacím registru je ještě předchozí nezpracovaný znak, je signalizována tato chyba.

Underrun error - Tato chyba může nastat v případě, že vysílač odeslal celý znak, a jeho výstupní odesílací registr je prázdný. V asynchronním režimu je spíše informační, protože mohou být přidány stop bity a informuje, že není co odesílat. Jako chyba se uplatní spíše v synchronním systému.

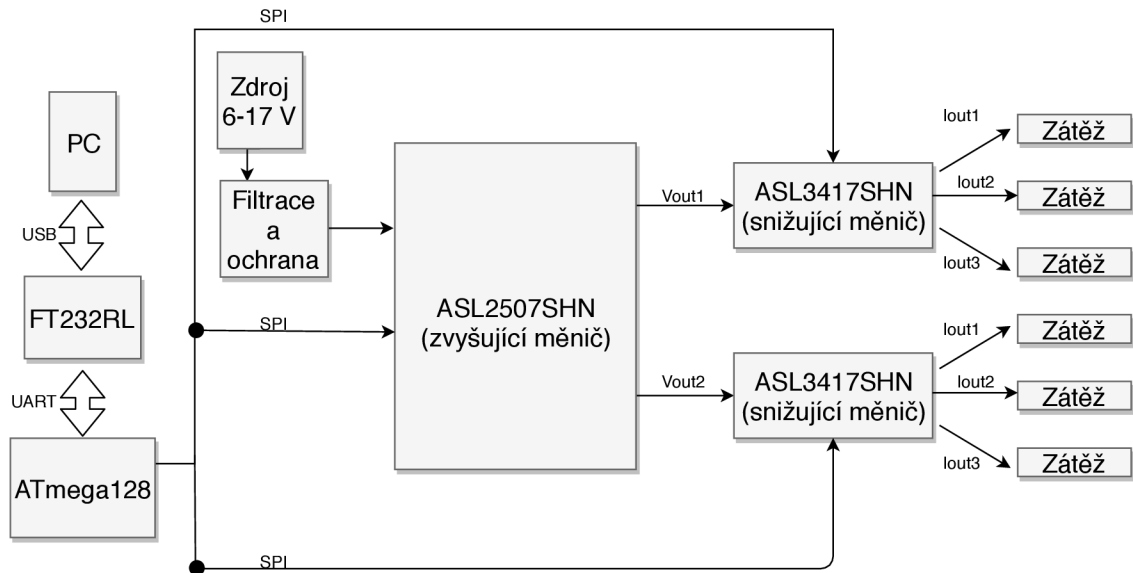
Framing error - V případě, že v očekávaném čase stop bitu je logická úroveň rozhraní v jiné než vysoké úrovni, je signalizována tato chyba. UART signalizuje *Framing error* i v případě *Break condition*.

Parity error - Chyba je signalizována, pokud je v rámci detekována přijímačem jiná parita než ta, která je nastavená pro danou komunikaci.

Rozhraní UART je často používáno pro mikroprocesorovou komunikaci. Samotné mikrokontrolery mají na svých pinech přímo rozhraní UART, avšak externí zařízení často používají komunikační protokoly založené na UARTu, ale s jinými fyzikálními parametry rozhraní. Nejčastěji je možné setkat se s převodem na *RS232*, *RS485* nebo *RS422*. V tomto projektu je použitý převodník *FT232R* převádějící UART na USB (Universal Serial Bus - univerzální sériová sběrnice) [22], [18].

3 ANALÝZA ZAPOJENÍ

Tato kapitola objasňuje principiální zapojení DC-DC měniče. Na obrázku 3.1 je možné vidět blokové schéma obvodu.



Obr. 3.1: Blokové schéma DC-DC měniče

Schéma obsahuje základní bloky výsledného zapojení. Hlavními bloky schématu jsou zvyšující a snižující měniče. Srdcem těchto bloků jsou integrované obvody společnosti NXP Semiconductors *ASL2507SHN* a *ASL3417SHN*. Výpočtům jednotlivých částí je věnovaná následující kapitola. Je možné si všimnout, že zvyšující měnič bude tvořit dvě nezávislá výstupní napětí *Vout1* a *Vout2*. Na každé z nich bude připojen snižující měnič se třemi nezávislými výstupními kanály *Iout1*, *Iout2* a *Iout3*.

Zvyšující měnič bude napájen ze stejnosměrného zdroje napětí. V zapojení budou mezi tento zdroj a měnič vsazeny obvody pro filtraci rušení po napájecím vedení a ochranné obvody proti přepólování.

Řízení obou stupňů měniče bude realizováno mikrokontrolerem ATmega128 prostřednictvím SPI sběrnice. Komunikace mezi mikrokontrolerem a uživatelským rozhraním v počítači bude uskutečněna sériovou linkou. Převod mezi rozhraním UART a sběrnici USB bude realizovat obvod *FT232RL*.

4 VÝPOČET JEDNOTLIVÝCH ČÁSTÍ MĚNIČE

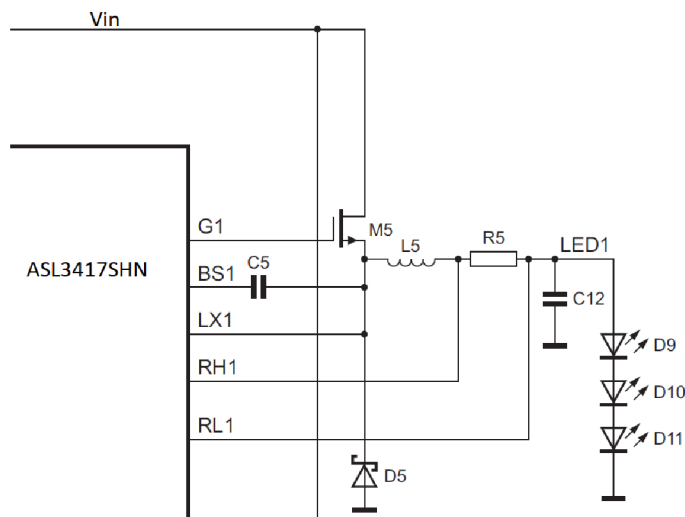
Před samotnými výpočty bylo důležité upřesnit, jaké podmínky má DC-DC měnič splňovat. Zadavatel projektu specifikoval následující:

- Jako zátěž mají být použity LED diody Advanced Power TOPLED LB G6SP.
- Zátěží má protékat proud 140 mA v každém výstupním kanále.
- Maximální příkon jednoho kanálu by měl být 7 Wattů.
- Rozsah napájecího, tedy vstupního napětí, by měl být možný v rozsahu 6-17 V, ale přípravek by měl vydržet až minutové 24 V špičky.
- Frekvence spínání není specifikována a je možné si ji libovolně určit.

4.1 Snižující měnič

Vstupními parametry pro výpočet snižujícího měniče jsou:

- Výstupní proud jedním kanálem má být 140 mA.
- Jeden kanál by měl mít příkon 7 Wattů.
- Frekvence spínání volitelná.
- Mají být použity LED diody LB G6SP s napětím v propustném směru $V_f = 4,1$ V při proudu $I_f = 140$ mA.
- Napájecí napětí snižujícího stupně je volitelné.



Obr. 4.1: Referenční zapojení jednoho výstupního kanálu čipu ASL3417SHN [9]

Cílem výpočtu je určit zejména indukčnost použité cívky, proudové zatížení polovodičů, nutná kapacita výstupního kondenzátoru a velikost rezistoru pro snímání

proudu. Tento rezistor je použit jako snímací rezistor s malým odporem. Podmínky pro určení jeho velikosti jsou uvedeny v datovém listu čipu *ASL3417SHN*.

Prvním možným krokem je určení výstupního napětí jednoho kanálu. Vzhledem k faktu, že proud zátěží a její příkon je daný, není problém napětí spočítat.

$$P_{\text{OUT}} = U_{\text{OUT}} \times I_{\text{OUT}} [\text{W}] \rightarrow U_{\text{OUT}} = \frac{P_{\text{OUT}}}{I_{\text{OUT}}} [\text{V}] = \frac{7}{0,140} = 50 \text{ V} \quad (4.1)$$

Dále lze například určit hodnotu snímacího rezistoru R_{sense} , označeného jako R_5 na obrázku 4.1. Podle datového listu má výběr snímacího rezistoru několik podmínek [9]:

- Pro výpočet je použit základní vztah:

$$R_{\text{sense}} = \frac{U_{\text{sense}}}{I_{\text{sense}}} [\Omega] \quad (4.2)$$

- Napětí na rezistoru se může pohybovat ve dvou rozsazích podle toho, jaký rozsah výstupního proudu je zapotřebí. Rozsahy jsou $U_{\text{sense}} \in <24,120; 120, 300>$ mV
- Proud rezistorem je roven výstupnímu proudu kanálu tedy $I_{\text{sense}} = 140$ mA.

Z důvodu možnosti regulace výstupního proudu ve větším rozsahu a zároveň menší hodnoty výsledného rezistoru a tedy i menších teplotních ztrát na něm byl zvolen první interval napětí.

$$R_{\text{sense}} = \frac{U_{\text{sense}}}{I_{\text{sense}}} = \frac{0,12}{0,14} = 0,857 \Omega \quad (4.3)$$

Nejbližší nižší dostupný rezistor má hodnotu 825 m Ω . To znamená, že výstupní proud bude možné regulovat v následujícím rozsahu:

$$I_{\text{out_per_channel_max}} = \frac{U_{\text{sense_max}}}{R_{\text{sense}}} = \frac{0,12}{0,825} = 145,45 \text{ mA} \quad (4.4)$$

$$I_{\text{out_per_channel_min}} = \frac{U_{\text{sense_min}}}{R_{\text{sense}}} = \frac{0,024}{0,825} = 29,09 \text{ mA} \quad (4.5)$$

Dalším parametrem, který je zapotřebí určit, je střída měniče. Vztah pro výpočet střídý lze odvodit z rovnosti napětově-časového součinu na cívce v čase sepnutí a rozepnutí tranzistoru. Tyto produkty se musí rovnat kvůli zachování energie [4].

$$E_{\text{on}} = (U_{\text{IN}} - U_{\text{T}} - U_{\text{SENSE}} - U_{\text{OUT}}) \times t_{\text{ON}} \quad (4.6)$$

$$E_{\text{off}} = (U_{\text{SENSE}} + U_{\text{OUT}} - U_{\text{D}}) \times t_{\text{OFF}} \quad (4.7)$$

U_{in} je vstupní napětí, U_T je úbytek napětí na otevřeném tranzistoru, U_{SENSE} je úbytek napětí na snímacím rezistoru, U_{OUT} je výstupní napětí a U_D je úbytek napětí v propustném směru na diodě.

$$(U_{IN} - U_T - U_{SENSE} - U_{OUT}) \times t_{ON} = (U_{SENSE} + U_{OUT} - U_D) \times t_{OFF} \quad (4.8)$$

Do vztahu je dále dosazeno následující:

$$t_{OFF} = T - t_{ON} \text{ [s]} \quad (4.9)$$

Po úpravách je vztah pro výpočet střídavy D následující:

$$D = \frac{t_{ON}}{T} = \frac{U_{SENSE} + U_{OUT} + U_D}{U_{IN} - U_T + U_D} \text{ [-]} \quad (4.10)$$

Pro uvedený maximální proud kanálu je úbytek napětí v propustném směru na diodě $V10P10$ $U_D = 0,3$ V a maximální úbytek napětí na snímacím rezistoru je již zmíněných 120 mV. Tranzistor $Si4056DY$ má v otevřeném stavu odpor kanálu $R_{DS(ON)} = 22$ m Ω , při maximálním proudu kanálem, tedy $I_{out_per_channel_max} = 145,45$ mA, je úbytek napětí na otevřeném tranzistoru U_T roven:

$$U_T = R_{DS(ON)} \times I_{out_per_channel_max} = 0,022 \times 0,14545 = 3,2 \text{ mV} \quad (4.11)$$

Měnič jako celek je navrhován tak, aby vstupní napětí snižující části bylo $U_{IN} = 75$ V. Nyní je možné vypočítat maximální střídavu snižujícího stupně dosazením do vzorce 4.10.

$$D = \frac{U_{SENSE} + U_{OUT} + U_D}{U_{IN} - U_T + U_D} = \frac{0,120 + 50 + 0,3}{75 - 0,0032 + 0,3} = 0,6696 \text{ [-]} \quad (4.12)$$

Pro následný výpočet zvyšující části měniče je důležité vypočítat maximální vstupní proud snižující části. Základem tohoto výpočtu je následující vzorec.

$$U_{IN} \times I_{IN_BUCK_max} \times \mu = U_{OUT} \times I_{out_per_channel_max} \times 3 \quad (4.13)$$

Násobení číslem "3" je zde, protože jeden čip snižujícího měniče má 3 výstupní kanály. Účinnost " μ " je zde kvůli ztrátám, které v měniči vzniknou. Ačkoliv účinnost není známá, výpočet s její odhadem bude více směrodatný než bez ní. Při výpočtu je použita hodnota účinnosti 0,9 [5].

$$I_{IN_BUCK_max} = \frac{U_{OUT} \times I_{out_per_channel_max} \times 3}{U_{IN} \times \mu} = \frac{50 \times 0,14545 \times 3}{75 \times 0,9} = 323,2 \text{ mA} \quad (4.14)$$

Nejdůležitější částí výpočtu je určení indukčnosti použité cívky v měniči. Výpočet vychází ze vzorce pro určení frekvence spínání uvedeného v datovém listu [9].

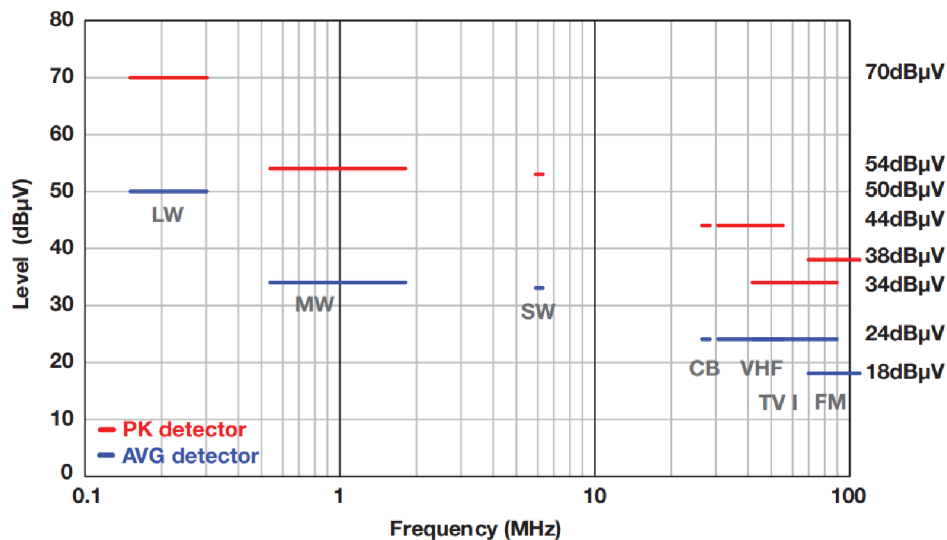
$$f = \frac{U_{IN} \times U_{OUT} - U_{OUT}^2}{I_{Hyst} \times L \times U_{IN}} \text{ [Hz]} \rightarrow L = \frac{U_{IN} \times U_{OUT} - U_{OUT}^2}{I_{Hyst} \times f \times U_{IN}} \text{ [H]} \quad (4.15)$$

Nyní jsou neznámy dvě veličiny. První je zvlnění proudu na cívce I_{Hyst} a druhou je frekvence spínání f .

Zvlnění proudu na cívce je procentuální násobek výstupního proudu DC-DC měniče. Procentuální míra se volí jako kompromis mezi velikostí cívky, její cenou a efektivní hodnotou proudu v měniči. Dobrý kompromis je mezi 20 % a 40 % [5].

$$I_{Hyst} = (0,2 \text{ až } 0,4) \times I_{out_per_channel_max} \text{ [A]} \rightarrow I_{Hyst} = 0,2 \times 0,14545 = 29,1 \text{ mA} \quad (4.16)$$

Ačkoliv frekvence spínání je volitelná, je vhodné její hodnotu určit s ohledem na následné testování elektromagnetických emisí do vedení. Norma "CISPR 25" určuje měřená pásma a jejich limity pro automobilový průmysl. Vyznačení měřených pásem je možné vidět na obrázku 4.2.



Obr. 4.2: Limity elektromagnetických emisí do napájecího vedení podle normy CISPR 25, třída 5 [7]

Je vidět, že měření nepodléhá celé pásmo 100 kHz až 100 MHz, ale pouze určité sub-pásma. Cílem je navrhnout měnič s frekvencí odpovídající neměřené části spektra. Maximální frekvence spínání je omezena čipem *ASL3417SHN* na 2 MHz. Nejbližší nižší volné pásmo, jak je vidět na obrázku 4.2, je 300-500 kHz. Frekvence

snižující části DC-DC měniče byla proto stanovena na 400 kHz. Nyní je možné spočítat hodnotu indukčnosti dosazením do vzorce 4.15.

$$L = \frac{U_{IN} \times U_{OUT} - U_{OUT}^2}{I_{Hyst} \times f \times U_{IN}} = \frac{75 \times 50 - 50^2}{0,0291 \times 400000 \times 75} = 1,4 \text{ mH} \quad (4.17)$$

Další částí výpočtu je proudové dimenzování polovodičů. Vzorce pro výpočet jsou převzaty z literatury [2].

Rovnice 4.18 popisuje výpočet maximálního proudu diodou, 4.19 je střední hodnota proudu diodou a 4.20 je efektivní hodnota proudu diodou. Maximální napětí na diodě v závěrném směru popisuje rovnice 4.21.

$$I_{D_max} = 0,5 \times I_{Hyst} + I_{out_per_channel_max} = 0,5 \times 0,0291 + 0,14545 = 0,1600 \text{ A} \quad (4.18)$$

$$I_{D_stř} = I_{out_per_channel_max} \times (1 - D) = 0,14545 \times (1 - 0,6696) = 0,0481 \text{ A} \quad (4.19)$$

$$I_{D_ef} = I_{out_per_channel_max} \times \sqrt{(1 - D)} = 0,14545 \times \sqrt{(1 - 0,6696)} = 0,0836 \text{ A} \quad (4.20)$$

$$U_{D_max} = U_{IN} = 75 \text{ V} \quad (4.21)$$

Rovnice 4.22 popisuje výpočet maximálního proudu tranzistorem, 4.23 je střední hodnota proudu tranzistorem a 4.24 je efektivní hodnota proudu tranzistorem. Největší napětí, které se může objevit na tranzistoru, popisuje rovnice 4.25.

$$I_{T_max} = 0,5 \times I_{Hyst} + I_{out_per_channel_max} = 0,5 \times 0,0291 + 0,14545 = 0,1600 \text{ A} \quad (4.22)$$

$$I_{T_stř} = I_{out_per_channel_max} \times D = 0,14545 \times 0,6696 = 0,0974 \text{ A} \quad (4.23)$$

$$I_{T_ef} = I_{out_per_channel_max} \times \sqrt{D} = 0,14545 \times \sqrt{0,6696} = 0,1190 \text{ A} \quad (4.24)$$

$$U_{DS_max} = U_{IN} = 75 \text{ V} \quad (4.25)$$

Posledním výpočtem je určení kapacity výstupního filtračního kondenzátoru. Velikost jeho kapacity ovlivňuje zvlnění napětí na výstupu měniče [2]. Pro zvlnění $\Delta U_{OUT} = 0,2 \text{ V}$ je nutná velikost kapacity:

$$C_{OUT_min} = \frac{I_{Hyst}}{8 \times f \times \Delta U_{OUT}} = \frac{0,0291}{8 \times 400000 \times 0,2} = 45,46 \text{ nF} \quad (4.26)$$

Závěrem musí být ověřeno, zda v LC obvodu měniče nedojde k vlastní rezonanci. Podmínka je následující:

$$C_{OUT} \gg \frac{1}{4 \times \pi^2 \times f^2 \times L} = \frac{1}{4 \times \pi^2 \times 400000^2 \times 0,0014} = 110,53 \text{ pF} \quad (4.27)$$

Je ověřeno, že v obvodu nedojde k vlastní rezonanci.

Tab. 4.1: Shrnutí vypočtených hodnot snižujícího měniče

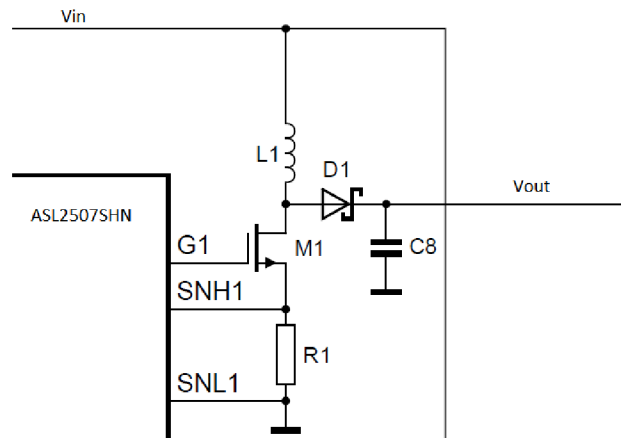
| Veličina | Hodnota | Jednotka | Popis |
|--------------------------------------|---------|----------|--|
| f_{DC-DC} | 400 | kHz | kmitočet spínání |
| U_{IN} | 75 | V | vstupní (napájecí) napětí |
| U_{OUT} | 50 | V | výstupní napětí |
| $I_{IN_BUCK_max}$ | 323,2 | mA | maximální vstupní proud |
| $I_{out_per_channel_max}$ | 145,45 | mA | maximální výstupní proud jednoho kanálu |
| D | 66,96 | % | střída |
| L | 1,4 | mH | indukčnost cívky |
| $I_{T_max}, I_{L_max}, I_{D_max}$ | 0,160 | A | maximální proud tranzistorem, cívkou, diodou |
| U_{D_max}, U_{DS_max} | 75 | V | maximální napětí na diodě, tranzistoru |
| C_{OUT_min} | 45,46 | nF | minimální kapacita výstupního filtračního kondenzátoru |

4.2 Zvyšující měnič

Pro výpočet zvyšující části měniče je nutné brát v potaz následující parametry:

- Vstupní napětí měniče se může pohybovat v rozmezí 6-17 V.
- Výstupní napětí vzhledem k výpočtu snižující části měniče je 75 V.

- Jak bylo vypočteno v rovnici 4.14, proud potřebný pro napájení jedné větve snižující části měniče je 323,2 mA.
- Frekvence spínání je volitelná.



Obr. 4.3: Referenční zapojení jednoho výstupního kanálu čipu ASL2507SHN [8]

Cílem výpočtu je opět určení indukčnosti potřebné pro správnou funkci měniče. Dále je nutné určit proudové zatížení polovodičů. Část výpočtu se rovněž věnuje určení kapacit na vstupu a výstupu měniče.

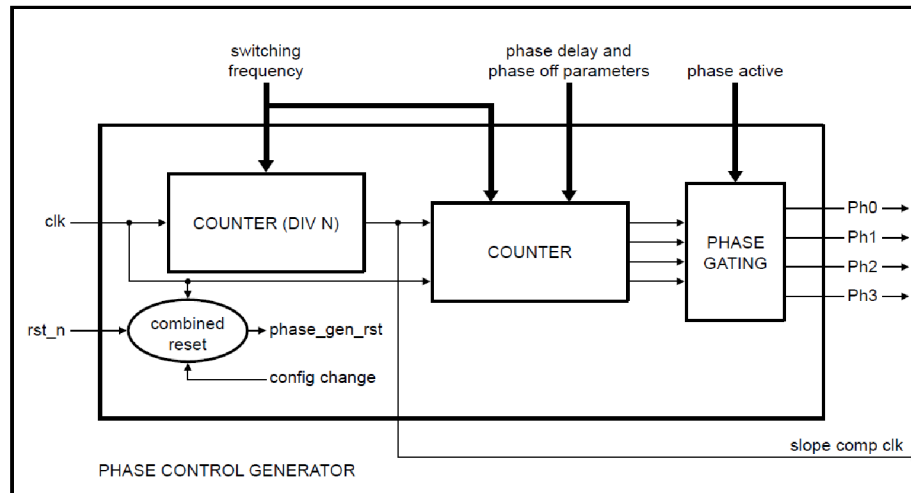
V případě vzestupné části měniče je vhodné jako první určit frekvenci spínání. Opět je zde cílem zasadit kmitočty spínání do pásem, které nepodléhají měření elektromagnetických emisí do vedení. Na obrázku 4.2 jsou tyto limity vyznačeny. Pásmo od 300 kHz do 500 kHz je už obsazeno snižující částí měniče a není vhodné do těchto kmitočtů navrhovat i zvyšující část. Nabízí se tedy pásma 100-150 kHz a pásmo nad 2 MHz. Integrovaný obvod *ASL2507SHN* může spínat v rozsahu kmitočtů $f_{DC-DC} \in \langle 125, 700 \rangle$ kHz. Odtud plyne, proč je pro výpočet zvoleno pásmo 100-150 kHz.

Konfigurace spínacího kmitočtu obnáší nastavení třech registrů, jak je možné vidět na obrázku 4.4.

Prvním stupněm je celočíselná dělička kmitočtu "*Counter (DIV N)*". Úkolem této části je celočíselně vydělit vstupní hodinový signál o frekvenci $f_{osc(int)} = 180$ MHz hodnotou $N \in \langle 2, 256 \rangle$.

Pro generaci a vhodné časování spínání jsou pro každý ze dvou fyzických výstupů v integrovaném obvodu k dispozici čtyři virtuální fáze. Spínání všech fází obou fyzických výstupů je odvozeno od stejného hodinového kmitočtu. To umožňuje definovat přesné fázové zpoždění ve spínání mezi fyzickými výstupy.

Registr "*PHASE GATING*" určuje, kolik virtuálních aktivních fází je pro každý fyzický výstup.



Obr. 4.4: Blokové schéma generátoru spínacího kmitočtu vzestupného měniče [8]

Poslední registr "Counter" definuje zpoždění mezi dvěma aktivními po sobě jdoucími fázemi. Tento registr zároveň definuje minimální dobu, po kterou musí být spínač měniče na výstupu rozepnut. Zpoždění lze nastavit v rozsahu $\langle 1, 32 \rangle$ period již děleného hodinového signálu. Minimální doba rozepnutí je v intervalu $\langle 1, 7 \rangle$ period děleného hodinového signálu.

Z výše zmíněného lze předpokládat, že frekvence spínání se vypočítá podle rovnice 4.28.

$$f_{DC-DC} = \frac{f_{osc(int)}}{N \times PAF \times Z} \text{ [Hz]} \quad (4.28)$$

Proměnná f_{DC-DC} [Hz] představuje výslednou frekvenci spínání měniče, $f_{osc(int)}$ [Hz] je frekvence interního oscilátoru, N [-] je celočíselná hodnota děličky kmitočtu jak bylo zmíněno výše, PAF [-] představuje počet aktivních virtuálních fází pro jeden fyzický výstup a Z [-] je zpoždění mezi fázemi vyjádřené v počtu period hodinového signálu.

Pro výpočet je známo, že frekvence spínání musí být v rozsahu $f_{DC-DC} \in \langle 125, 150 \rangle$ kHz. Podle datového listu součástky je vhodné pro dobrou interní kompenzaci subharmonických oscilací volit hodnotu délky zpoždění (násobek děleného hodinového kmitočtu) mezi fázemi tak, aby součin s počtem aktivních fází byl roven 32. Co se počtu aktivních virtuálních fází týká, vzhledem k faktu, že měnič bude mít dva nezávislé napěťové výstupy s předpokládanou stejnou zátěží, postačí 2 virtuální fáze.

Při určování hodnoty děličky kmitočtu je uvažováno, že interní oscilátor může dosahovat odchylky až $\pm 5\%$.

$$f_{\text{osc(int)}_max} = f_{\text{osc(int)}} + (f_{\text{osc(int)}} \times 0,05) \text{ [Hz]} = 180 \times 10^6 + (180 \times 10^6 \times 0,05) = 189 \text{ MHz} \quad (4.29)$$

$$f_{\text{osc(int)}_min} = f_{\text{osc(int)}} - (f_{\text{osc(int)}} \times 0,05) \text{ [Hz]} = 180 \times 10^6 - (180 \times 10^6 \times 0,05) = 171 \text{ MHz} \quad (4.30)$$

Aby nebyla překročena frekvence spínání 150 kHz, je pro určení N nutné uvažovat $f_{\text{osc(int)}_max}$. Z rovnice 4.28 lze odvodit:

$$N = \frac{f_{\text{osc(int)}}}{f_{\text{DC-DC_max}} \times PAF \times Z} [-] \rightarrow N = \frac{f_{\text{osc(int)}_max}}{f_{\text{DC-DC_max}} \times PAF \times Z} [-] = \frac{189 \times 10^6}{150 \times 10^3 \times 2 \times 16} = 39,375 [-] \rightarrow 40 [-] \quad (4.31)$$

Nyní tedy lze určit nominální frekvenci spínání vzestupné části DC-DC měniče podle rovnice 4.28:

$$f_{\text{DC-DC}} = \frac{f_{\text{osc(int)}}}{N \times PAF \times Z} = \frac{180 \times 10^6}{40 \times 2 \times 16} = 140,625 \text{ kHz} \quad (4.32)$$

Integrovaný obvod *ASL2507SHN* může vytvořit dva nezávislé napěťové výstupy. Toho je v práci využito a následující výpočty jsou shodné pro oba tyto výstupy.

Na začátek je vhodné určit vstupní proud $I_{\text{IN_BOOST}}$ vzestupného měniče. Výpočet je obdobný jako v případě rovnice 4.13. Důvod použití účinnosti μ je zde stejný, rovněž její hodnota je v tomto případě stejná.

$$U_{\text{IN}} \times I_{\text{IN_BOOST}} \times \mu = U_{\text{OUT_BOOST}} \times I_{\text{OUT_BOOST}} \rightarrow I_{\text{IN_BOOST}} = \frac{U_{\text{OUT_BOOST}} \times I_{\text{OUT_BOOST}}}{U_{\text{IN}} \times \mu} \text{ [A]} \quad (4.33)$$

Podle zadání bude měnič napájen napětím v rozsahu $U_{\text{IN}} \in \langle 6, 17 \rangle$ V. Je zapotřebí uvážit úbytek napětí na vstupním obvodu proti přepólování a na vstupním filtru. Tranzistor použitý v obvodu pro ochranu *TPC8128* má pro spodní mez napájecího napětí, a tedy největší proud, odpor kanálu v otevřeném stavu 10 mΩ a pro horní mez 5 mΩ. Stejnoseměrný odpor cívky, záleží na použitém typu, je okolo 6 mΩ. Odhadovaný maximální proud vstupním obvodem (napájené jsou oba výstupní kanály vzestupného měniče) bude 10 A, minimální pak 4 A. Při uvážení tohoto odhadu bude vstupní rozsah napětí pro výpočet následující:

$$U_{\text{IN_min}} = 6 - (10 \times 0,010 + 10 \times 0,006) = 5,84 \text{ V} \quad (4.34)$$

$$U_{IN_max} = 17 - (4 \times 0,005 + 4 \times 0,006) = 16,96 \text{ V} \quad (4.35)$$

Pokud tedy do rovnice 4.33 dosadíme tyto meze, dostaneme maximální a minimální vstupní proud DC-DC měniče. Zároveň uvažujeme fakt, že výstupní proud I_{OUT_BOOST} je roven proudu $I_{IN_BUCK_max}$ vypočítanému v předchozí kapitole.

$$I_{IN_BOOST_max} = \frac{U_{OUT_BOOST} \times I_{OUT_BOOST}}{U_{IN_min} \times \mu} = \frac{75 \times 0,3232}{5,84 \times 0,9} = 4,612 \text{ A} \quad (4.36)$$

$$I_{IN_BOOST_min} = \frac{U_{OUT_BOOST} \times I_{OUT_BOOST}}{U_{IN_max} \times \mu} = \frac{75 \times 0,3232}{16,96 \times 0,9} = 1,588 \text{ A} \quad (4.37)$$

Dále je možné určit například střídu měniče. Obdobně jako v předchozí kapitole věnované snižujícímu měniči lze vztah pro výpočet střídy odvodit z rovnosti napětově-časového součinu na cívce v čase sepnutí a rozepnutí tranzistoru. Tyto produkty se opět musí shodovat kvůli zachování energie [4].

$$E_{on} = (U_{IN} - U_T - U_{SENSE}) \times t_{ON} \quad (4.38)$$

$$E_{off} = (U_D + U_{OUT} - U_{IN}) \times t_{OFF} \quad (4.39)$$

U_{in} je vstupní napětí, U_T je úbytek napětí na otevřeném tranzistoru, U_{SENSE} je úbytek napětí na snímacím rezistoru, U_{OUT} je výstupní napětí, U_D je úbytek napětí v propustném směru na diodě.

$$(U_{IN} - U_T - U_{SENSE}) \times t_{ON} = (U_D + U_{OUT} - U_{IN}) \times t_{OFF} \quad (4.40)$$

Do vztahu je dále dosazeno následující:

$$t_{OFF} = T - t_{ON} \text{ [s]} \quad (4.41)$$

Po úpravách je vztah pro výpočet střídy D následující:

$$D = \frac{t_{ON}}{T} = \frac{U_D + U_{OUT} - U_{IN}}{U_{OUT} - U_T - U_{SENSE} + U_D} \text{ [-]} \quad (4.42)$$

Nastává zde obdobná situace stejně jako u výpočtu vstupního proudu. Pro rozsah vstupního napětí $U_{IN} \in \langle 5,84, 16,96 \rangle \text{ V}$ bude definován i rozsah střídy.

Pro maximální vstupní napětí poteče obvodem proud $I_{IN_BOOST_min}$, naopak pro minimální napětí to bude proud $I_{IN_BOOST_max}$. To ovlivní zejména parametr U_T a U_{SENSE} . Snímací rezistor R_{SENSE} použitý v obvodu má odpor $10 \text{ m}\Omega$, odpor kanálu

R_{DSON} otevřeného tranzistoru *SiR606BDP* je pro danou aplikaci $25 \text{ m}\Omega$. Úbytek napětí U_D na diodě *V10P10* v propustném směru je $0,5 \text{ V}$.

Pro maximální vstupní napětí, a tomu odpovídající minimální proud, bude tedy úbytek na těchto součástkách:

$$U_{T_min} = R_{DSON} \times I_{IN_BOOST_min} = 0,025 \times 1,588 = 0,0397 \text{ mV} \quad (4.43)$$

$$U_{SENSE_min} = R_{SENSE} \times I_{IN_BOOST_min} = 0,010 \times 1,588 = 0,0159 \text{ mV} \quad (4.44)$$

Pro minimální vstupní napětí, a tedy maximální proud, bude úbytek na těchto součástkách:

$$U_{T_max} = R_{DSON} \times I_{IN_BOOST_max} = 0,025 \times 4,612 = 0,1153 \text{ mV} \quad (4.45)$$

$$U_{SENSE_max} = R_{SENSE} \times I_{IN_BOOST_max} = 0,010 \times 4,612 = 0,0461 \text{ mV} \quad (4.46)$$

Následuje výpočet střídny:

$$D_{max} = \frac{U_D + U_{OUT} - U_{IN_min}}{U_{OUT} - U_{T_max} - U_{SENSE_max} + U_D} = \frac{0,5 + 75 - 5,84}{75 - 0,1153 - 0,0461 + 0,5} = 0,9246 \text{ [-]} \quad (4.47)$$

$$D_{min} = \frac{U_D + U_{OUT} - U_{IN_max}}{U_{OUT} - U_{T_min} - U_{SENSE_min} + U_D} = \frac{0,5 + 75 - 16,96}{75 - 0,0397 - 0,0159 + 0,5} = 0,7760 \text{ [-]} \quad (4.48)$$

Střída spínání měniče je tedy v intervalu $D \in \langle 0,7760, 0,9246 \rangle$.

Nyní je možné vypočítat potřebnou indukčnost cívky. Vycházíme ze vztahu pro výpočet zvlnění proudu na cívce [2], [6].

$$\Delta I = \frac{U_{IN} \times D}{L \times f_{DC-DC}} \text{ [A]} \rightarrow L = \frac{U_{IN} \times D}{\Delta I \times f_{DC-DC}} \text{ [H]} \quad (4.49)$$

Zvlnění proudu je procentuální násobek maximálního proudu cívkou. Velikost zvlnění, podobně jak u snižujícího měniče, se opět volí jako kompromis s ohledem na

velikost cívky a její cenu. Vhodný kompromis je opět mezi 20 % a 40 % maximálního proudu cívkou, který odpovídá vstupnímu proudu $I_{IN_BOOST_max}$ [6].

$$\Delta I = (0,2 \text{ až } 0,4) \times I_{IN_BOOST_max} [\text{A}] \rightarrow \Delta I = 0,2 \times 4,612 = 922,4 \text{ mA} \quad (4.50)$$

Maximální proud prochází cívkou v případě, že je na vstupu minimální napětí, to rovněž odpovídá maximální střídě.

$$L = \frac{U_{IN_min} \times D_{max}}{\Delta I \times f_{DC-DC}} [\text{H}] = \frac{5,84 \times 0,9246}{0,9224 \times 140625} = 41,6 \text{ } \mu\text{H} \quad (4.51)$$

Následuje výpočet parametru, který je dobré znát pro pozdější programování vnitřních registrů měniče. Je jím nejkratší možná doba, po kterou je při spínání tranzistor vypnut.

$$\begin{aligned} t_{off_min} = T_{DC-DC} - t_{on_max} &= T_{DC-DC} - (D_{max} \times T_{DC-DC}) = \frac{1}{f_{DC-DC}} \times (1 - D_{max}) = \\ &= \frac{1}{140625} \times (1 - 0,9246) = 536,17 \text{ ns} \end{aligned} \quad (4.52)$$

Následují výpočty pro proudové dimenzování polovodičů. Vztahy pro určení proudů jsou převzaty z literatury [2].

Rovnice 4.53 slouží k výpočtu střední hodnoty proudu diodou, 4.54 popisuje výpočet efektivní hodnoty proudu procházejícího diodou a 4.55 je maximální proud. Maximální napětí v závěrném směru na diodě je popsáno rovnicí 4.56.

$$I_{D_stř} = I_{IN_BOOST_max} \times (1 - D_{max}) = 4,612 \times (1 - 0,9246) = 0,3477 \text{ A} \quad (4.53)$$

$$I_{D_ef} = I_{IN_BOOST_max} \times \sqrt{(1 - D_{max})} = 4,612 \times \sqrt{(1 - 0,9246)} = 1,2664 \text{ A} \quad (4.54)$$

$$I_{D_max} = 0,5 \times \Delta I + I_{IN_BOOST_max} = 0,5 \times 0,9224 + 4,612 = 5,0732 \text{ A} \quad (4.55)$$

$$U_{D_max} = U_{OUT} = 75 \text{ V} \quad (4.56)$$

Následují rovnice pro tranzistor. Jedná se o rovnici 4.57 pro střední hodnotu proudu, 4.58 pro efektivní hodnotu a 4.59 pro maximální hodnotu proudu tranzistorem. Maximální napětí na tranzistoru je popsáno rovnicí 4.60.

$$I_{T_stř} = I_{IN_BOOST_max} \times D_{max} = 4,612 \times 0,9246 = 4,2643 \text{ A} \quad (4.57)$$

$$I_{T_ef} = I_{IN_BOOST_max} \times \sqrt{D_{max}} = 4,612 \times \sqrt{0,9246} = 4,4347 \text{ A} \quad (4.58)$$

$$I_{T_max} = 0,5 \times \Delta I + I_{IN_BOOST_max} = 0,5 \times 0,9224 + 4,612 = 5,0732 \text{ A} \quad (4.59)$$

$$U_{DS_max} = U_{OUT} = 75 \text{ V} \quad (4.60)$$

Tab. 4.2: Shrnutí vypočtených hodnot zvyšujícího měniče

| Veličina | Hodnota | Jednotka | Popis |
|--------------------------------------|-----------------|---------------|--|
| f_{DC-DC} | 140,625 | kHz | kmitočet spínání |
| U_{IN} | <6 , 17> | V | vstupní (napájecí) napětí |
| U_{OUT} | 75 | V | výstupní napětí |
| I_{IN_BOOST} | <1,588 , 4,612> | A | rozsah vstupního proudu |
| I_{OUT_BOOST} | 0,3232 | A | výstupní proud |
| D | <77,60 , 92,46> | % | rozsah střídý |
| L | 41,6 | μH | indukčnost cívky |
| $I_{T_max}, I_{L_max}, I_{D_max}$ | 5,0732 | A | maximální proud tranzistorem, cívkou, diodou |
| U_{D_max}, U_{DS_max} | 75 | V | maximální napětí na diodě, tranzistoru |
| C_{OUT_min} | 10,63 | μF | minimální kapacita výstupního filtračního kondenzátoru |

Pro následné výběry součástek je vhodné podotknout, že maximální proud cívkou bude odpovídat maximálnímu proudu tranzistorem z rovnice 4.59.

Opět je posledním krokem výpočet kapacity výstupního kondenzátoru. Minimální hodnota kapacity závisí na požadovaném zvlnění výstupního napětí (v tomto případě $\Delta U_{OUT} = 0,2 \text{ V}$) a je dána vztahem:

$$C_{OUT_min} = \frac{I_{OUT_BOOST} \times D_{max}}{f_{DC-DC} \times \Delta U_{OUT}} = \frac{0,3232 \times 0,9246}{140625 \times 0,2} = 10,63 \mu\text{F} \quad (4.61)$$

Závěrem musí být ověřeno, zda v LC obvodu měniče nedojde k vlastní rezonanci. Podmínka je následující:

$$C_{\text{OUT}} \gg \frac{1}{4 \times \pi^2 \times f^2 \times L} = \frac{1}{4 \times \pi^2 \times 140625^2 \times 0,0000416} = 30,79 \text{ nF} \quad (4.62)$$

Je ověřeno, že v obvodu nedojde k vlastní rezonanci.

4.3 Výběr základních součástek

Další částí návrhu DC-DC měniče je výběr vhodných součástek pro realizaci. Jedná se zejména o tranzistor pro spínání, diodu a cívku. Vzhledem k faktu, že měnič je navrhován pro praxi, je mimo dodržení vypočtených parametrů důležitá také cena součástky, rozměr a výhodou je rovněž vhodné pouzdro pro snadnou výrobu.

4.3.1 Tranzistory

V zapojení budou použity tranzistory typu MOSFET. Existuje několik požadavků na spínací tranzistory. Kapacita na vstupu a výstupu tranzistoru by neměla překročit 1000 pF. Jeho sériový odpor v otevřeném stavu by měl být co nejmenší, z důvodu, že velikost tohoto odporu je úměrná velikosti stejnosměrných výkonových ztrát. Problémem je, že s klesajícím odporem stoupá vstupní kapacita. Je nutné tedy najít kompromis. Doba přechodu z otevřeného stavu do zavřeného a obráceně by měla být co nejmenší kvůli dynamickým výkonovým ztrátám. Ovládací napětí bude 5,5 V, je nutností tedy najít tranzistory, které budou při této úrovni napětí dostatečně otevřené. Vyhledávání se tedy soustředí na tzv. "logic level MOSFET".

Vzestupný měnič

Největší výkonové požadavky jsou na tranzistor ve zvyšujícím měniči. Maximální proudové špičky na tranzistoru dosahují hodnot více než 5 A, viz. rovnice 4.59. Vypočtené napětí na tranzistoru má hodnotu 75 V.

Mezi mnoha dostupnými tranzistory na běžných tuzemských internetových obchodech byl vybrán tranzistor *SiR606BDP* vyráběný společností Vishay Siliconix [12]. Parametry tohoto tranzistoru byly dostačující a vytvářely dobrý kompromis pro danou aplikaci. Ostatní tranzistory měly příliš vysoké vstupní a výstupní kapacity, příliš vysoký odpor kanálu v otevřeném stavu a dlouhou dobu otevření či zavření. Vznikala tak příliš velká výkonová ztráta. Hlavní výhodou vybraného tranzistoru byl nízký teplotní odpor.

Aby mohl být tranzistor bezpečně použit, je důležité vypočítat, zda při daném zatížení nepřekročí maximální povolenou provozní teplotu, a díky tomu nedojde k

jeho destrukci. V případě světlometů může teplota okolí dosahovat 80 °C. Základní parametry pro výpočet jsou uvedeny v tabulce 4.3.

Tab. 4.3: Parametry pro výpočet výkonových ztrát tranzistoru SiR606BDP

| Veličina | Hodnota | Jednotka | Popis |
|---------------|---------|----------|---|
| f_{DC-DC} | 140,625 | kHz | kmitočet spínání |
| U_{DS_max} | 75 | V | maximální napětí mezi Drain a Source tranzistoru, viz. rovnice 4.60 |
| I_{T_ef} | 4,4347 | A | nejvyšší efektivní hodnota proudu tranzistorem, viz. rovnice 4.58 |
| $I_{T_stř}$ | 4,2643 | A | nejvyšší střední hodnota proudu tranzistorem, viz. rovnice 4.57 |
| R_{DS_on} | 25 | mΩ | odpor kanálu tranzistoru v otevřeném stavu |
| t_{rise} | 12 | ns | doba přechodu tranzistoru do otevřeného stavu |
| t_{fall} | 10 | ns | doba přechodu tranzistoru do zavřeného stavu |
| R_{thJA} | 25 | °C/W | teplotní odpor mezi křemíkem a okolím |

Hlavní složky ztrát na tranzistoru jsou stejnosměrné ztráty a dynamické ztráty způsobené spínáním. S rostoucím kmitočtem spínání se dynamické ztráty stávají dominantními.

Stejnosemné ztráty se v zapojeních s tranzistory MOSFET vypočítají následovně [23]:

$$P_{SS} = R_{DS_on} \times I_{T_ef}^2 \text{ [W]} \quad (4.63)$$

R_{DS_on} je odpor kanálu tranzistoru v otevřeném stavu a I_{T_ef} je efektivní hodnota proudu protékajícího tranzistorem.

Ztráty způsobené spínáním se dělí na ztráty při sepnutí a při rozepnutí [23]:

$$P_{ON} = I_{T_stř} \times U_{DS_max} \times \frac{t_{rise} + t_{fall}}{2} \times f_{DC-DC} \text{ [W]} \quad (4.64)$$

Ztráty způsobené rozepnutím se počítají stejně jako pro sepnutí (podle rovnice 4.64) za předpokladu, že doby přechodů mezi stavy jsou u obou cyklů stejné. Celkové výkonové ztráty se tedy počítají následovně [23]:

$$P_{\text{TOT}} = P_{\text{SS}} + P_{\text{ON}} + P_{\text{OFF}} = P_{\text{SS}} + 2 \times P_{\text{ON}} \quad [\text{W}] \quad (4.65)$$

Ztráty na tranzistoru vzestupného měniče spočítáme dosazením hodnot z tabulky 4.3.

$$\begin{aligned} P_{\text{TOT}} &= P_{\text{SS}} + 2 \times P_{\text{ON}} = R_{\text{DS_on}} \times I_{\text{T_ef}}^2 + 2 \times (I_{\text{T_stř}} \times U_{\text{DS_max}} \times \frac{t_{\text{rise}} + t_{\text{fall}}}{2} \times f_{\text{DC-DC}}) = \\ &= 0,027 \times 4,4347^2 + 2 \times (4,2643 \times 75 \times \frac{(12 + 10) \times 10^{-9}}{2} \times 140625) = 1,5204 \text{ W} \end{aligned} \quad (4.66)$$

Nyní je možné vypočítat změnu teploty, o kolik se oteplí křemík čipu oproti teplotě okolí.

$$\Delta T_{\text{JA}} = R_{\text{thJA}} \times P_{\text{TOT}} = 25 \times 1,5204 = 38,01 \text{ } ^\circ\text{C} \quad (4.67)$$

Maximální pracovní teplota křemíku tranzistoru je $T_{\text{J_max}} = 150 \text{ } ^\circ\text{C}$. Z toho plyne, že maximální teplota okolí, při které je možné provozovat tranzistor při daném zatížení, bude:

$$T_{\text{A_max}} = T_{\text{J_max}} - \Delta T_{\text{JA}} = 150 - 38,01 = 111,99 \text{ } ^\circ\text{C} \quad (4.68)$$

Tranzistor je tedy možné provozovat do poměrně vysoké teploty, aniž by došlo k jeho destrukci.

Sestupný měnič

U sestupného měniče jsou požadavky na tranzistor o něco mírnější zejména kvůli mnohem menšímu proudovému zatížení. Naopak na dynamických ztrátách se projeví vyšší kmitočet spínání.

Jako vhodný byl pro tuto aplikaci zvolen tranzistor Si4056DY vyráběný společností Vishay Siliconix [11]. Splňuje požadavky na malou vstupní a výstupní kapacitu, a má nízký odpor kanálu v otevřeném stavu. Nevýhodou tranzistoru je vyšší teplotní odpor mezi křemíkem a okolím, nicméně vzhledem k malému zatížení by to nemělo představovat velký problém.

Obdobně jako v předchozím případě je dosazeno do rovnic 4.63, 4.64, 4.65:

Tab. 4.4: Parametry pro výpočet výkonových ztrát tranzistoru Si4056DY

| Veličina | Hodnota | Jednotka | Popis |
|----------------------|---------|----------|---|
| $f_{\text{DC-DC}}$ | 400 | kHz | kmitočet spínání |
| $U_{\text{DS_max}}$ | 75 | V | maximální napětí mezi Drain a Source tranzistoru, viz. rovnice 4.60 |
| $I_{\text{T_ef}}$ | 0,1190 | A | nejvyšší efektivní hodnota proudu tranzistorem, viz. rovnice 4.58 |
| $I_{\text{T_stř}}$ | 0,0974 | A | nejvyšší střední hodnota proudu tranzistorem, viz. rovnice 4.57 |
| $R_{\text{DS_on}}$ | 22 | mΩ | odpor kanálu tranzistoru v otevřeném stavu |
| t_{rise} | 28 | ns | doba přechodu tranzistoru do otevřeného stavu |
| t_{fall} | 20 | ns | doba přechodu tranzistoru do zavřeného stavu |
| R_{thJA} | 50 | °C/W | teplotní odpor mezi křemíkem a okolím |

$$\begin{aligned}
 P_{\text{TOT}} &= P_{\text{SS}} + 2 \times P_{\text{ON}} = R_{\text{DS_on}} \times I_{\text{T_ef}}^2 + 2 \times (I_{\text{T_stř}} \times U_{\text{DS_max}} \times \frac{t_{\text{rise}} + t_{\text{fall}}}{2} \times f_{\text{DC-DC}}) = \\
 &= 0,022 \times 0,1190^2 + 2 \times (0,0974 \times 75 \times \frac{(28 + 20) \times 10^{-9}}{2} \times 400000) = 0,1406 \text{ W}
 \end{aligned}
 \tag{4.69}$$

Opět následuje výpočet maximální teploty prostředí, ve kterém lze tranzistor při daném zatížení používat:

$$\Delta T_{\text{JA}} = R_{\text{thJA}} \times P_{\text{TOT}} = 50 \times 0,1406 = 7,03 \text{ °C}
 \tag{4.70}$$

$$T_{\text{A_max}} = T_{\text{J_max}} - \Delta T_{\text{JA}} = 150 - 7,03 = 142,97 \text{ °C}
 \tag{4.71}$$

Vzhledem k dobrým vlastnostem, nízké ceně a dobře pájitelnému pouzdru budou tyto tranzistory, po konzultaci s vedoucím práce, použity pro spínání sestupného

stupně měniče i jako spínače v zátěži. Toto řešení je výhodné zejména s ohledem na případnou výrobu, pokud se použije více totožných součástek.

4.3.2 Diody

Diody v DC-DC měničích by měly splňovat opět několik požadavků. Úbytek napětí v propustném směru by měl být co nejmenší z důvodu nízkých stejnosměrných ztrát. Kapacita mezi elektrodami by měla být rovněž co nejmenší. S rostoucí kapacitou se zmenšuje rychlost spínání diody a dynamické ztráty rostou. Proud v závěrném směru by rovněž měl být co nejmenší opět kvůli minimalizaci ztrát. Těmto požadavkům nejlépe vyhovují rychlé Schottkyho diody.

Mezi mnoha nabízenými diodami byla jako nejvhodnější vybrána dioda s označením V10P10 vyráběná společností Vishay General Semiconductor [10]. U zvyšujícího měniče by neměl úbytek napětí překročit 0,5 V v propustném směru, u snižujícího pak 0,3 V. Závěrný proud by neměl překročit hodnotu 4,5 mA.

Vzhledem k dobrým parametrům diody bude, po konzultaci s vedoucím práce, dioda použita u vzestupného i sestupného měniče. Vzhledem k tomuto faktu postačí vypočítat změnu teploty čipu vůči okolí v nejvíce namáhané části schématu, tedy u vzestupného měniče.

Výrobce udává, že celkové ztráty při průchodu proudu o střední hodnotě 0,3477 A (viz. rovnice 4.54) a střídě signálu 0,9246 (odpovídající spínání vzestupného měniče), nepřekročí celkové výkonové ztráty P_{TOT} hodnotu 0,5 W. Hodnota teplotního odporu je $R_{thJA} = 60 \text{ }^\circ\text{C}$. Je tedy možné přímo vypočítat teplotní parametry diody.

$$\Delta T_{JA} = R_{thJA} \times P_{TOT} = 60 \times 0,5 = 30 \text{ }^\circ\text{C} \quad (4.72)$$

$$T_{A_max} = T_{J_max} - \Delta T_{JA} = 150 - 30 = 120 \text{ }^\circ\text{C} \quad (4.73)$$

Dioda může pracovat při daném zatížení v teplotách okolí do 120 °C.

4.3.3 Cívky

Výběr cívek je poslední částí výběru součástek. Cívky se vyrábějí v mnoha variantách, nevýhodou všech je však jejich velikost, rostoucí indukčnost a velikost protékajícího proudu tento parametr velmi ovlivňují.

Výběr cívek pro tento projekt je obtížný zejména z důvodu poměrně velkých hodnot zmíněných veličin. Navíc nákup součástek bude možný pouze z omezeného množství internetových obchodů.

U vzestupného měniče byla vypočítaná indukčnost cívky 41,6 μH . To není hodnota, která by v SMD (Surface Mount Device - zařízení pro povrchovou montáž)

cívkách byla neobvyklá. Nicméně problémem je, že tato cívka musí být provozovaná při zatížení proudem, kde jeho maximální hodnota překročí 5 A. Cívek, které by měly rozumnou velikost, a byly schopny pracovat v těchto podmínkách, je dostupných poměrně málo. Z těch několika málo dostupných bude pro realizaci zvolena ta cívka, která bude mít nejmenší stejnosměrný odpor, aby se omezily výkonové ztráty.

U sestupného měniče je situace obdobná. Ačkoliv zde není problém s velkým proudem, je zde požadována velká indukčnost, která činí 1,4 mH. Pro výběr budou použita stejná kritéria jako v případě vzestupného měniče. Mimo vypočtených požadavků, které musí cívka splňovat, budou brány v potaz fyzická velikost a stejnosměrný odpor.

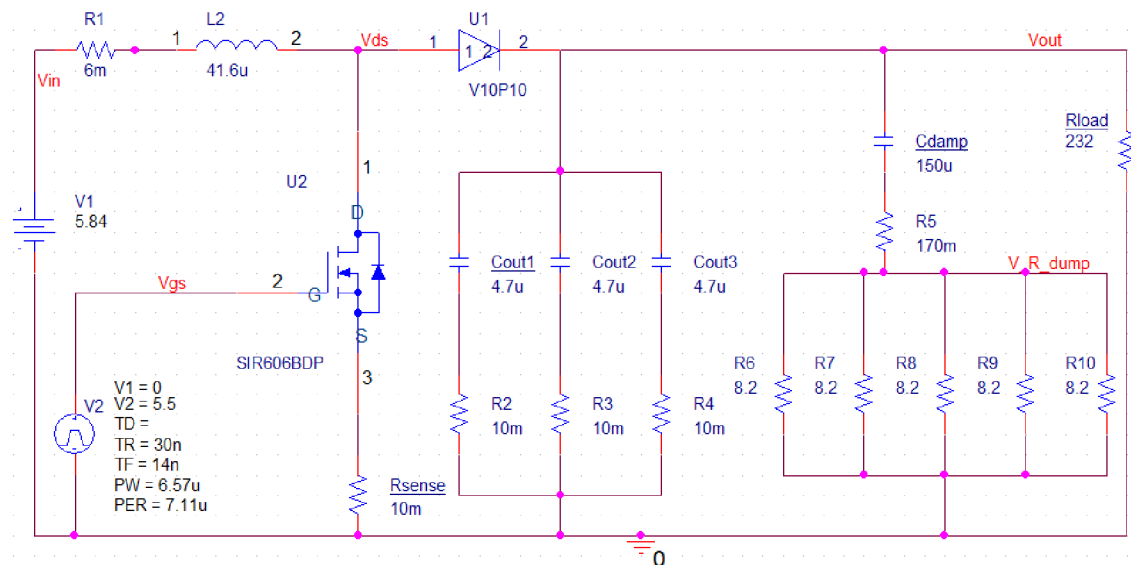
5 SIMULACE JEDNOTLIVÝCH STUPŇŮ DC-DC MĚNIČE

Simulace měniče je realizovaná v programu OrCAD PSpice. Do simulace byly zahrnuty modely polovodičů od výrobce. Modely čipů *ASL3417SHN* a *ASL2507SHN* bohužel k dostání nebyly, spínací napětí je proto nahrazeno zdroji pulzů, kde časování pulzů odpovídá vypočteným frekvencím a střídám. Doby náběžných a sestupných hran jsou nastaveny podle datových listů použitých řídicích čipů. Součástí simulovaného schématu jsou, snímací rezistory a parazitní odpory cívek a kondenzátorů.

Vzhledem k faktu, že modely řídicích čipů nejsou do simulace zahrnuty, jsou v simulaci zkoumány pouze základní parametry. Mezi ně se řadí výstupní napětí, zvlnění výstupního napětí, výstupní a vstupní proud a zvlnění proudu na cívce.

5.1 Zvyšující měnič

Simulace je realizována ve dvou variantách a to pro obě krajní hodnoty napájecího napětí, tedy 5,84 V a 16,96 V. Schéma simulovaného obvodu je na obrázku 5.1.



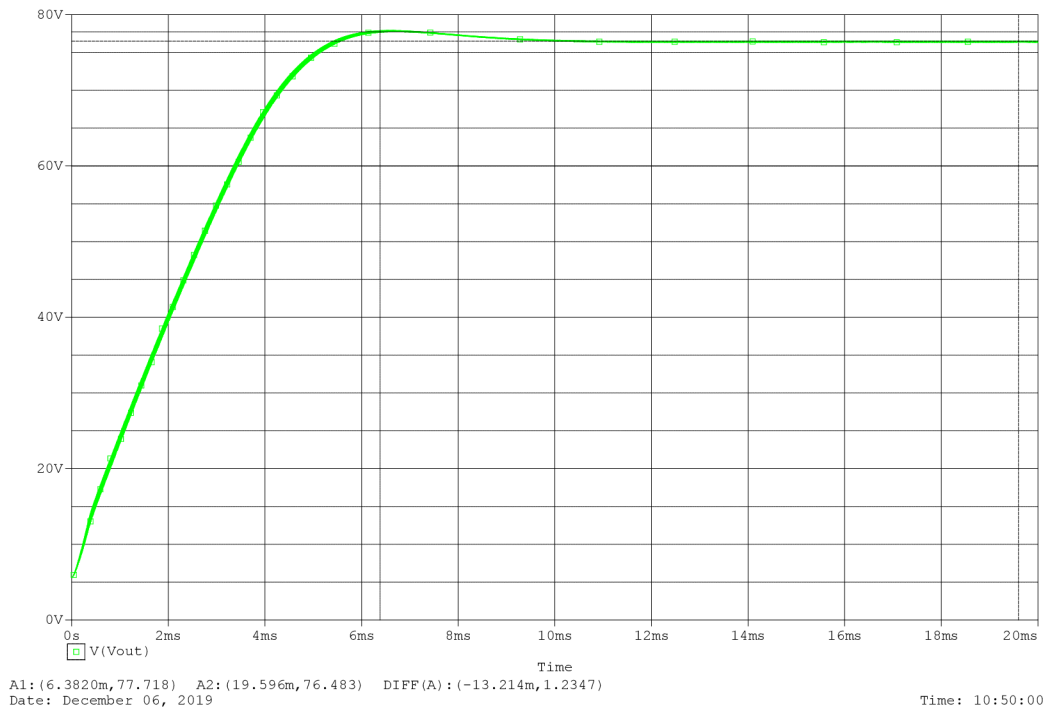
Obr. 5.1: Simulační schéma zvyšujícího měniče

Schéma pro simulaci obsahuje více kondenzátorů než by se dalo podle výpočtu očekávat. Důvodů je několik.

Vypočtená kapacita C_{out_min} je zde rozdělená do tří paralelně zapojených kondenzátorů. Důvodem rozdělení je omezení výkonového zatížení těchto kondenzátorů, kdy jeden SMD kondenzátor by toto zatížení nezvládl. Další důvod rozdělení

je dostupnost vícevrstvých keramických kondenzátorů. Menší kapacity jsou cenově dostupnější.

Ke zmíněným filtračním kondenzátorům je připojen paralelní RC článek. Při simulaci bez článku se ukázalo, že po zapnutí dochází ke kmitům v LC obvodu měniče. Napětí kmitů přesahovalo vypočtené výstupní napětí o několik desítek Voltů. V reálném obvodu by takové přepětí ohrožovalo polovodičové součástky. Lze předpokládat, že čip *ASL2507SHN* by svými regulačními obvody této napěťové špičky měl zabránit. Nicméně vzhledem k nedostatku informací o čipu bylo po konzultaci s vedoucím práce rozhodnuto, že je nutné tyto kmity utlumit vhodným útlumovým článkem a v případě, že čip bude schopen kmity tlumit, nemusí být útlumový článek použit.



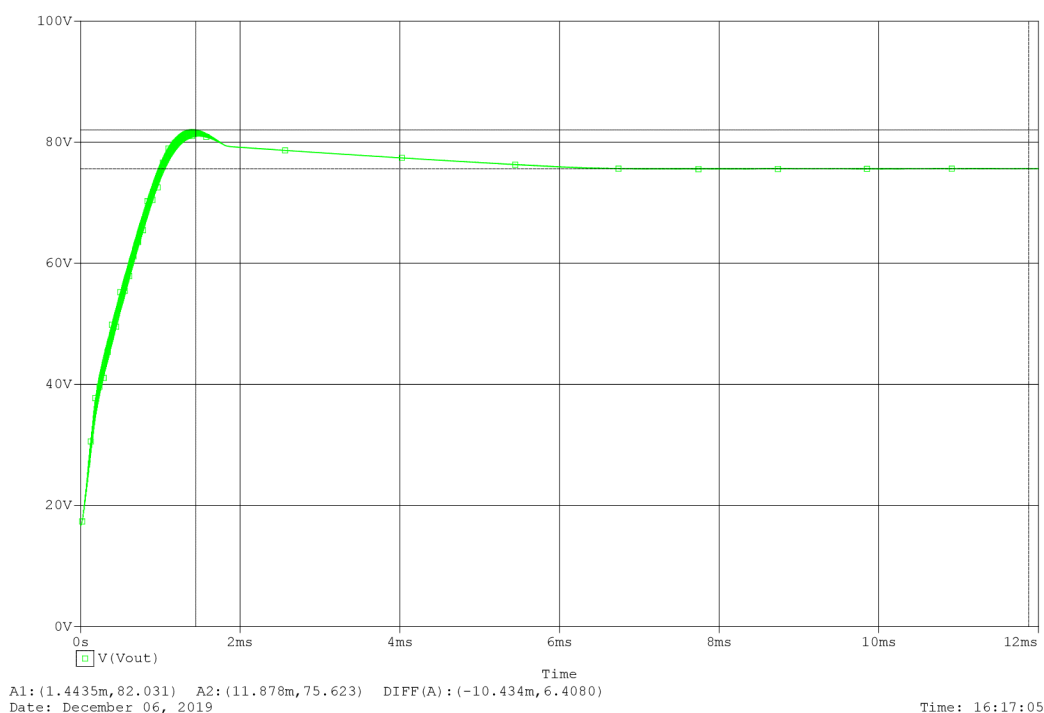
Obr. 5.2: Simulace výstupního napětí vzestupného měniče, $U_{IN} = 5,84 \text{ V}$

Tlumení samozřejmě nelze provést pouhým rezistorem, na kterém by vznikaly velké výkonové ztráty. Jednou z možností je připojit paralelně k výstupním filtračním kondenzátorům další kondenzátor s mnohem větší kapacitou, který má v sérii zapojený rezistor. Kapacita kondenzátoru byla zvolena o řád vyšší než C_{out_min} . Prvotní hodnota odporu R_{damp} byla určena ze vztahu pro činitel jakosti RLC obvodu. Pokud je hodnota činitele jakosti rovná jedné, jsou kmity v obvodu tlumeny [15].

$$Q = \frac{R_{\text{damp}}}{\sqrt{\frac{L}{C_{\text{out1}} + C_{\text{out2}} + C_{\text{out3}}}}} = 1 \rightarrow R_{\text{damp}} = \sqrt{\frac{L}{C_{\text{out1}} + C_{\text{out2}} + C_{\text{out3}}}} =$$

$$= \sqrt{\frac{41,6 \times 10^{-6}}{3 \times 4,7 \times 10^{-6}}} = 1,72 \Omega \quad (5.1)$$

Následně byla velikost odporu v programu PSpice krokována kolem výchozí hodnoty $1,72 \Omega$. Jako dostačující hodnota rezistoru byl tímto způsobem určen odpor $1,64 \Omega$. Patrně kvůli parazitnímu odporu tlumícího kondenzátoru C_{damp} . Opět kvůli rozdělení výkonu je zvolena paralelní kombinace pěti $8,2 \Omega$ rezistorů.



Obr. 5.3: Simulace výstupního napětí vzestupného měniče, $U_{IN} = 16,96 \text{ V}$

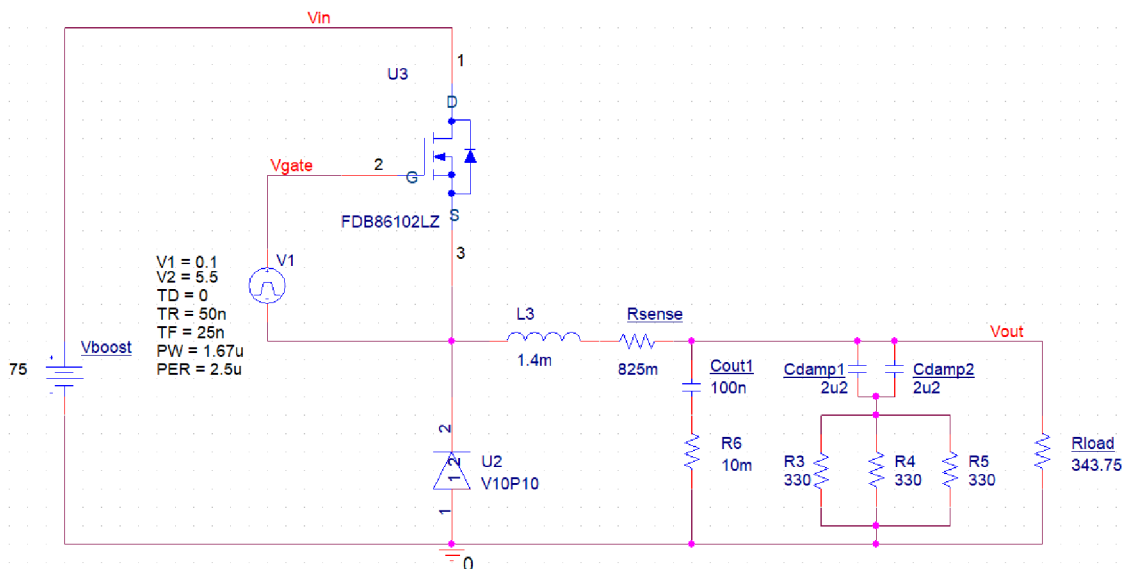
V dalším kroku byla provedena simulace obvodu pro krajní hodnoty intervalu napájecího napětí. Na obrázku 5.2 a 5.3 jsou zobrazeny průběhy výstupního napětí v intervalu prvních 20 ms po zapnutí simulace. Pořád je zde patrný drobný překmit na začátku průběhu, nicméně již v rozsahu několika jednotek Voltů. Je možné si všimnout, že napětí na výstupu přesně neodpovídá vypočteným hodnotám, ale trochu je překračuje. Více simulací ukázalo, že pokud se tranzistor zcela neotevřel, odpor kanálu tranzistoru byl řádově větší než v otevřeném stavu. Obvod LC byl následně natolik tlumen, že nedošlo k jeho rozkmitání. Když tato situace nastala, bylo

výstupní napětí rovno vypočteným hodnotám. Nicméně vzhledem k faktu, že obvod *ASL2507SHN* automaticky mění střidu spínání podle naměřeného výstupního napětí, měl by tento nepatrný rozdíl bez problému kompenzovat.

V příloze A.1 jsou pro doplnění vloženy další průběhy této simulace. Obrázek A.1 a A.2 zobrazuje zvlnění výstupního napětí. Dále je v příloze zobrazen průběh výstupního proudu na obrázku A.3 a A.4. Obrázky A.5 a A.4 popisují zvlnění vstupního proudu, tedy i proudu cívkou. Příloha A.1 rovněž obsahuje dvě tabulky (A.1 a A.2), které obsahují střední a efektivní hodnoty proudu tranzistorem a diodou pro obě napájecí napětí.

5.2 Snižující měnič

Tato část měniče má pevně danou hodnotu vstupního napětí a střidu spínání. Simulační schéma popisuje obrázek 5.4.



Obr. 5.4: Simulační schéma snižujícího měniče

Ve schématu jsou opět patrné nesrovnalosti s předchozím textem. Jedná se o jiný typ tranzistoru a, jako v případě boost měniče, o útlumový člen na výstupu měniče.

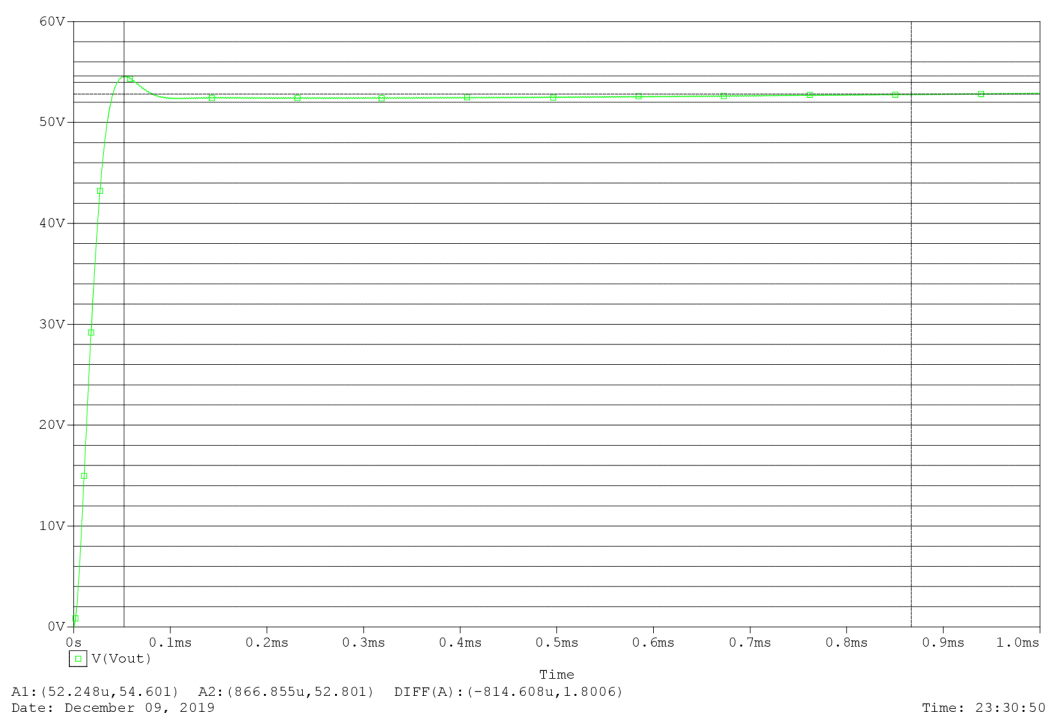
Pro realizaci byl zvolen tranzistor *Si4056DY*. Problém však je, že výrobce neposkytuje simulační model tohoto obvodu. Bylo tedy nutné najít odpovídající náhradu, která modelem disponuje. Velmi podobný parametrům zvolené součástky byl tranzistor *FDB86102LZ* vyráběný společností ON Semiconductor [13]. Tento tranzistor je proto v simulaci použit.

Po začátku simulace se v obvodu opět objevila napěťová špička přesahující výstupní napětí o několik desítek Voltů. Z obdobných důvodů stejně jako ve vzestupném měniči byl použit útlumový RC člen. Kapacita kondenzátoru C_{damp} byla opět zvolena mnohonásobně větší než filtrační kondenzátor C_{out1} . Kvůli rozdělení výkonu, zejména výkonové špičky po zapnutí měniče, jsou zapojeny dva kondenzátory paralelně. Sériový odpor byl opět vypočten ze vztahu pro činitel jakosti RLC obvodu (viz. rovnice 5.1).

$$R_{damp} = \sqrt{\frac{L}{C_{out1}}} = \sqrt{\frac{1,4 \times 10^{-3}}{100 \times 10^{-9}}} = 118,32 \Omega \quad (5.2)$$

Vypočtený R_{damp} byl kvůli rozdělení výkonu rozdělen do třech rezistorů zapojených paralelně.

Průběh výstupního napětí snižujícího měniče v intervalu prvních 1 ms je možné vidět na obrázku 5.5. Průběh napětí v délce 10 ms je možné najít v příloze A.2 na obrázku A.7.

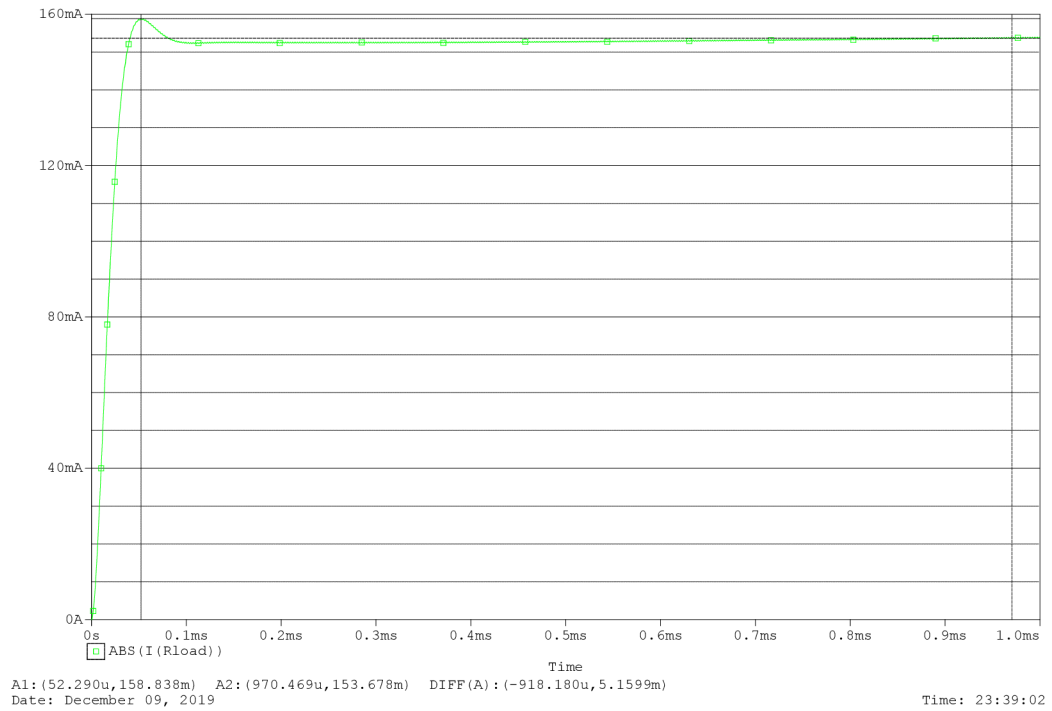


Obr. 5.5: Výstupní napětí snižujícího měniče v intervalu 1 ms po spuštění simulace

Průběh proudu zátěží R_{load} popisuje obrázek 5.6. Výsledné napětí je opět trochu vyšší než bylo vypočítáno. Je to pravděpodobně opět způsobeno kmitáním LC obvodu měniče. Opět je možné předpokládat, že nepatrné navýšení výstupního napětí

bude kompenzováno řídicím čipem *ASL3417SHN*.

V příloze A.2 se nachází doplňující průběhy vybraných veličin. Na obrázku A.8 je možné vidět zvlnění výstupního napětí měniče. Obrázek A.9 ukazuje zvlnění proudu na cívce. Příloha A.2 opět obsahuje tabulku A.3 se středními a efektivními hodnotami proudu polovodiči v obvodu.



Obr. 5.6: Výstupní proud snižujícího měniče v intervalu 1 ms po spuštění simulace

6 NÁVRH SCHÉMATU

Tvorbu schématu lze rozdělit do pěti částí. Toto rozdělení vychází z blokového schématu na obrázku 3.1 na straně 30. Části návrhu jsou následující:

- Vstupní obvody
- Řídící mikrokontroler
- Zvyšující měnič
- Snižující měnič
- Zátěž

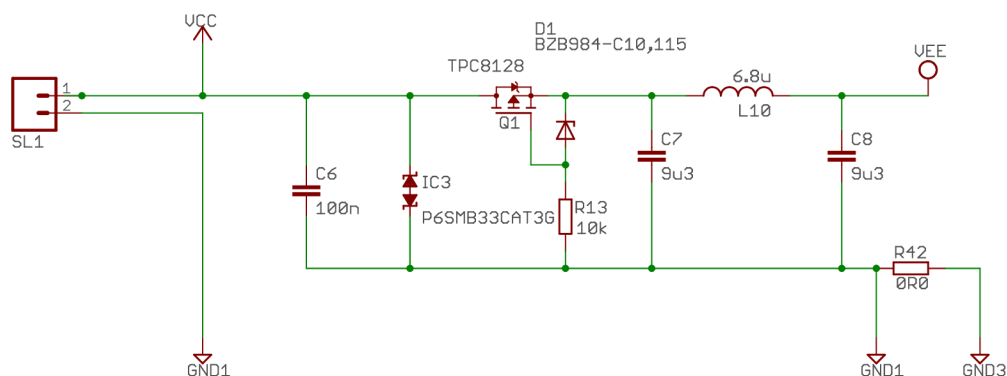
6.1 Vstupní obvody

Mezi vstupní obvody jsou zahrnuty tři části schématu. Jedná se o vstupní ochranu proti přepólování s filtrací, tvorbu napájecího napětí pro digitální obvody a převodník mezi USB (Universal Serial Bus - univerzální sériová sběrnice) a UART (Universal Asynchronous Receiver and Transmitter - univerzální asynchronní sériové rozhraní).

6.1.1 Vstupní ochrana s filtrací

Hlavním úkolem této části schématu je ochránit DC-DC měnič před neúmyslným přepólováním. Dalším úkolem je zajištění ochrany proti neočekávaným výkyvům v napájecím napětí a zlepšení EMC (Electromagnetic Compatibility - elektromagnetická kompatibilita). Elektromagnetická kompatibilita se dělí na odolnost systému proti vnějšímu rušení EMS (Electromagnetic Susceptibility - elektromagnetická odolnost) a schopnost zařízení neovlivňovat okolní elektrické obvody EMI (Electromagnetic Interference - elektromagnetická interference). V tomto případě se jedná o rušení po napájecím vedení. Na obrázku 6.1 se nachází navržené zapojení vstupního obvodu. Kondenzátor C_6 spolu s TVS (Transient Voltage Suppression) diodou tvoří ochranu DC-DC měniče proti napěťovým špičkám, které by se do obvodu mohly dostat z napájecího zdroje.

Dále je zde zařazena ochrana proti přepólování. Ochrana je tvořena tranzistorem MOSFET s kanálem typu P a Zenerovou diodou. V normálním stavu, tedy při správné polaritě vstupního napětí, je tranzistor v otevřeném stavu. Zenerova dioda v obvodu omezuje maximální napětí na hradle tranzistoru, chrání tedy tranzistor před zničením. Při přepólování se na hradle tranzistoru objeví vyšší napětí než na elektrodě "source" a dojde k rozepnutí tranzistoru a přerušení obvodu. Výhodou tohoto zapojení oproti například diodě je řádově menší úbytek napětí v propustném směru.



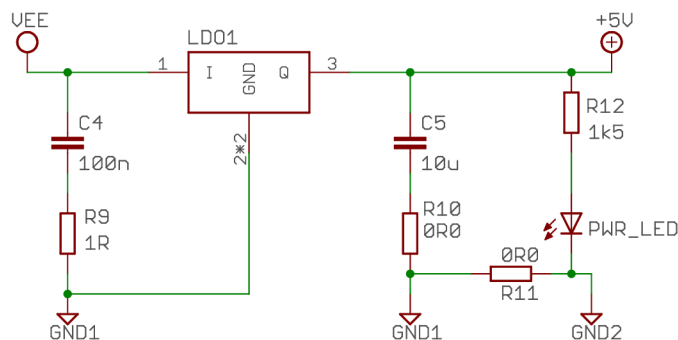
Obr. 6.1: Schéma vstupní ochrany s filtrací

Za ochranou proti přepólování se nachází filtr typu dolní propust. Vzhledem k tomu, že filtr je symetrický dvojbran, zajišťuje filtraci signálu v obou směrech. Cílem návrhu je pokud možno co největší redukce indukčnosti cívky vzhledem k více než 9 A zatížení.

Filtr je navržen na mezní frekvenci 20 kHz. Po konzultaci s vedoucím práce byl návrh realizován ve dvou krocích. Byla zvolena cívka s co možná největší indukčností, která vydrží dané zatížení a je běžně dostupná. Následně je podle Thomsonova vztahu vypočtená potřebná kapacita pro danou frekvenci. Těmto kritériím odpovídá kombinace cívky $L = 6,8 \mu\text{H}$ a kondenzátoru $C = 9,3 \mu\text{F}$. Rezistorem $R42$ je oddělená výkonová zemnicí svorka od společné.

6.1.2 Napájecí napětí digitálních obvodů

Napájení digitálních částí obvodu vyžaduje napájecí napětí 5 V. Při návrhu je třeba uvažovat fakt, že spodní mez napájecího napětí měniče jako celku je 6 V. Z toho důvodu nelze použít klasický lineární regulátor napětí z řady *78xx*. Je nutné použít regulátor s malým úbytkem napětí, tzv. LDO (Low-Dropout regulator). V tomto případě je použitý regulátor *TLE4264-2*. Obvod způsobí maximálně 0,5 V úbytek napětí, takže i po započtení úbytku na vstupních obvodech bude napětí pro regulaci dostatečné.



Obr. 6.2: Schéma zapojení lineárního stabilizátoru pro napájení digitálních obvodů

Zapojení obsahuje filtrační kondenzátory podle doporučení výrobce, odpor pro oddělení digitální zemnicí svorky od společné a LED diodu pro signalizaci připojeného napájení.

6.1.3 Převodník mezi USB a UART

Použitý řídicí mikrokontroler *ATmega128* neumožňuje přímou komunikaci s počítačem prostřednictvím USB. Na jeho pinech je však možné realizovat rozhraní UART. Aby byla možná komunikace mezi mikrokontrolerem a uživatelským rozhraním na počítači je mezi zmíněné části vložen převodník *FT232RL*. Schéma zapojení převodníku vychází z doporučeného zapojení výrobce. V příloze B.1 se nachází na obrázku B.1 schéma zapojení převodníku.

6.2 Řídicí mikrokontroler

Zapojení mikrokontroleru *ATmega128* vychází z doporučení výrobce. Schéma zapojení se nachází v příloze B.4 na obrázku B.5. Napájecí napětí mikrokontroleru je 5 V. Ke všem napájecím pinům je připojen blokovací kondenzátor o hodnotě 100 nF. Celé zapojení řídicí části používá samostatný zemnicí bod, který bude následně v jednom bodě připojen k zemnicímu bodu DC-DC měniče jako celku.

Programování mikrokontroleru bude realizováno prostřednictvím sériového rozhraní ISP využívající piny PE0, PE1 a PB1. Kvůli ochraně pinů jsou mezi konektor ISP rozhraní a mikrokontroler zařazeny rezistory o hodnotě 100 Ω . Signál SCLK je

zároveň připojen přes pull-down rezistor k zemi, aby na něm nevznikaly nežádoucí stavy.

Jako zdroj hodinového signálu slouží externí krystal s frekvencí 7,3728 MHz. Tato konkrétní hodnota je zvolena kvůli použití komunikačního rozhraní UART. Hodnoty symbolové rychlosti rozhraní UART jsou v celočíselném poměru s frekvencí zvoleného krystalu.

Sériová sběrnice SPI je alternativní funkcí pinů PB1 - hodinový signál, PB2 - MOSI (Master Output Slave Input - směr dat z nadřízeného do podřízeného zařízení) a PB3 - MISO (Master Input Slave Output - směr dat z podřízeného do nadřízeného zařízení). Piny jsou opět chráněny rezistorem o hodnotě 100 Ω . Signály SS (Slave Select - výběr podřízeného zařízení) jsou připojeny k pinům PF0-PF2. Zároveň jsou spojeny přes rezistor o hodnotě 10 k Ω k napájecímu napětí pro udržení vysoké logické úrovně na vodičích. V tomto stavu jsou řídicí čipy měniče neaktivní. Naopak komunikační vodiče jsou připojeny k zemi prostřednictvím pull-down rezistorů.

Piny PF3-PF5 jsou signálové piny pro povolovací vstupy řídicích čipů měniče a jsou přes pull-down rezistor připojeny k zemi.

Řídicí obvod obsahuje několik signalizačních LED diod pro snadné signalizování vybraných stavů měniče při provozu. Jsou připojeny k pinům PF6, PF7, PA0-PA2. Diody jsou trvale připojeny k napájecímu napětí a budou aktivovány nízkou logickou úrovní na pinu.

Ve schématu se nachází sběrnice s názvem "IO_PINY". Signály (piny) připojené k této sběrnici jsou použity k ovládání zátěže.

Komunikační rozhraní UART, použité pro komunikaci s uživatelským rozhraním na počítači, je alternativní funkcí pinů PD2-PD5. Pro ochranu pinů je na jednotlivých vodičích opět zařazen rezistor o hodnotě 100 Ω . Navíc, pro filtraci nežádoucích zámků, je mezi vodiče rozhraní a zem připojen kondenzátor 4,7 nF.

Jako poslední alternativní funkce vstupně-výstupních pinů procesoru bude využito tří signálů PWM (Pulse Width Modulation - pulzně šířková modulace). Signály budou přivedeny k jednomu ze dvou čipů *ASL3417SHN* pro ověření funkčnosti regulace výstupního proudu prostřednictvím PWM. V klidovém stavu jsou piny připojeny přes pull-down rezistory k zemi.

6.3 Zvyšující měnič

Schéma zapojení zvyšující části DC-DC měniče vychází ze simulačního schématu. Jsou zde přidány pouze blokové kondenzátory k cívkám a k řídicímu čipu. Zenerova dioda připojena k pinu VGG je zde kvůli ochraně vzhledem k faktu, že napětí na tomto pinu nesmí přesáhnout 10 V. Elektrolytické kondenzátory *C22* a *C29* jsou zde pro případ, kdyby při první realizaci bylo potřeba dodatečné filtrace. Pokud

nebudou nutné, budou následně odstraněny. Schéma zapojení se nachází v příloze B.2 na obrázku B.2.

6.4 Snižující měnič

Schéma snižující části měniče je rozděleno do dvou částí. Obě se nachází v příloze B.3. První na obrázku B.3 a druhá na obrázku B.4. Rozdíl mezi těmito částmi je pouze v zapojení pinů pro PWM regulaci. Zatím co čip v prvním schématu je spojen s mikrokontrolerem, druhý čip má PWM piny připojeny pull-down rezistorem k zemi. Na vodičích PWM regulace, mezi mikrokontrolerem a čipem *ASL3417SHN* jsou připojeny kondenzátory pro omezení záskmitů. V ničem jiném se tato schémata neliší a následný popis platí pro obě.

Základní části obvodu vychází ze simulačního schématu. Jsou zde doplněny opět blokující kondenzátory připojené mezi elektrodu "drain" spínacích tranzistorů a zem. Vzhledem ke spínané energii je jejich kapacita řádově vyšší než u kondenzátorů blokujících napájení řídicího čipu. Opět, jako v případě zvyšujícího měniče, je zde ochranná Zenerova dioda na pinu VGG. Součástky připojené na piny BSx a LXx jsou doporučené výrobcem.

Ve schématu jsou doplněny součástky, které nebudou pravděpodobně osazeny. Jsou zde pro případné ladění obvodu. Mezi nimi jsou například RC články připojené paralelně ke Schottkyho diodám. Ty by byly osazeny v případě, že by spínání bylo příliš rychlé a vytvářelo rušení na vysokých frekvencích. RC článek toto spínání zpomalí. Dále jsou zde přidány rezistory mezi řídicí čip a hradla tranzistorů pro případné zpomalení spínání, pokud by bylo nutné. Rovněž na vstupu snižující části je přidán kondenzátor, který může být osazen za účelem zmenšení rušení anebo z důvodu dodatečné stabilizace vstupního napětí.

6.5 Zátěž

Původním záměrem bylo realizovat zátěž z LED diod s využitím řídicího obvodu z rodiny *ASL5xxxSHN* vyráběné společností NXP Semiconductors. Při návrhu se však ukázalo, že informace o tomto čipu vede výrobce jako NDA (Non-Disclosure Agreement - dohoda o mlčenlivosti). Po konzultaci s vedoucím práce bylo rozhodnuto, že zátěž nebude realizována s využitím těchto obvodů, ale bude nahrazena sériovým zapojením LED diod s možností změny odběru zátěže pomocí tranzistorů ovládaných řídicím mikrokontrolerem.

V obvodu jsou použity LED diody *LW G6CP* výrobce OSRAM Opto Semiconductors [14]. Proud v propustném směru touto diodou je $I_f = 140$ mA při napětí

$V_f = 4,1$ V. Vzhledem k vypočteným hodnotám pro snižující měnič je těchto diod zapojeno 12 v sérii.

Změna odběru zátěže bude realizována odpojováním částí diod pomocí MOSFET tranzistoru *Si4056DY*. Aby byla možnost tvořit různé kombinace zátěže, bude počet a pozice tranzistorů pro každý ze dvou výstupních kanálů rozdílný. Třetí kanál bude ukončen svorkami, aby bylo možné pro testování obvodu použít i jiné typy zátěže. Tranzistory budou řízeny vstupně-výstupními piny mikrokontroleru *ATmega128* použitého zároveň pro ovládání řídicích čipů měniče. Na začátku série LED diod je zařazen rezistor o hodnotě 1Ω pro měření proudu.

Kompletní schéma zátěže je možné vidět na obrázku B.6 a B.7 v příloze B.5.

7 NÁVRH DESKY PLOŠNÉHO SPOJE

Návrh DPS (Deska Plošného Spoju) je realizován v prostředí Eagle. Tento program byl zvolen zejména z důvodu jeho dostupnosti a kvůli předešlé zkušenosti s jeho použitím.

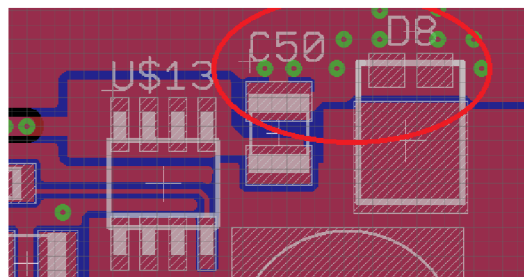
Deska je dělená do částí, které odpovídají dělení schématu v předchozí kapitole. Celé zapojení se nachází na jedné desce, zejména z důvodu kompaktnosti prototypu. Měníč obsahuje pouze konektor pro připojení napájecího napětí a konektor USB pro komunikaci s počítačem. Zátěž z LED diod je rovněž implementována přímo na desce prototypu, pouze dva kanály jsou vyvedeny na konektory pro případné testování i jiných typů zátěže. Zapojení dále obsahuje konektor pro programovací rozhraní ISP mikrokontroleru ATmega a několik testovacích bodů pro snímání digitálních komunikačních signálů.

Plošný spoj je dvouvrstvý, důvodem je zejména nižší výrobní cena. Šířka mědi je $90\ \mu\text{m}$, větší šířka mědi je zvolena zejména z důvodu velkého proudového zatížení spojů a zároveň kvůli nutnosti dobrého chlazení. Rozměr výsledného prototypu je 257mm na délku a 180mm na šířku.

Pro realizaci jsou použity součástky v provedení SMD (Surface Mount Device - zařízení pro povrchovou montáž). Ve většině se jedná o velikosti součástek 0805, 1206 a 1210, kde větší pouzdra jsou použity pro více namáhané spoje nebo v případech kde součástky, zejména kapacitory, nejsou dostupné v malých pouzdech kvůli velké kapacitě či velkému průraznému napětí.

Výslednou desku plošného spoje je možné si prohlédnout v příloze C.

7.1 Zemnicí plocha

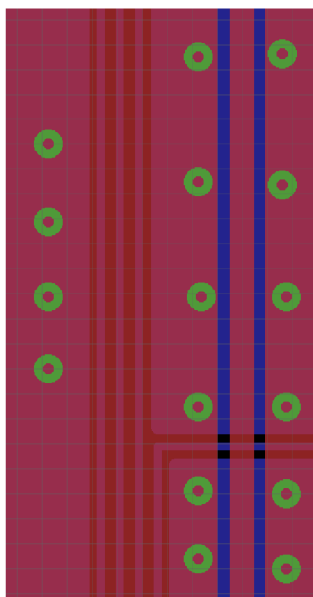


Obr. 7.1: Příklad rozložení součástek ve výstupním kanále snižujícího měniče

Dobře navržená zemnicí plocha je základem dobré funkce obvodu. Je důležité zajistit, aby na desce nevznikaly smyčky na zemnicí ploše, a aby nevznikaly rozdíly potenciálu zemnicí plochy mezi kritickými částmi obvodu. Jedná se o místa připojení

snímacích rezistorů nebo jiných částí spínacích smyček. Obecně lze říct, že vzdálenost mezi spínacími prvky, cívkami a filtračními prvky by měla být co nejmenší. To je vyobrazeno na obrázku 7.1. Mezi filtračním kondenzátorem C50 a diodou D8 je velmi malá vzdálenost, navíc je zde dostatek prokůvů což rovněž omezí lokální změnu potenciálu zemnicí plochy.

Pokud se v některé z vrstev vyskytnou dlouhé cesty, které dělí rozlitou zemnicí plochu, je vhodné tyto cesty přemostit pomocí druhé vrstvy. To je možné vidět na obrázku 7.2.



Obr. 7.2: Příklad přemostění dlouhých cest dělících zemnicí plochu

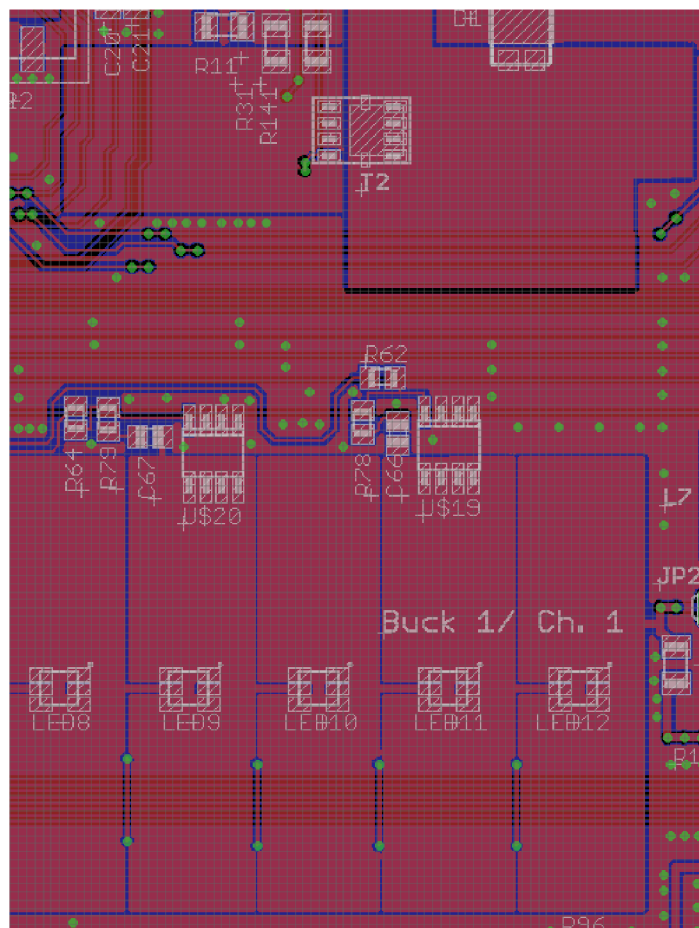
Otázkou zůstává, zda je vhodné dělit zemnicí plochu na tu pro digitální signály a pro výkonové signály. V tomto případě bylo rozhodnuto vytvořit jednu společnou zem. Toto řešení bylo zvoleno hned z několika důvodů. V praxi jedna zem pozitivně přispívá k dosažení dobrých výsledků při měření elektromagnetické kompatibility. Dalším, hlavním důvodem je to, že obvody ASL2507SHN a ASL3417SHN nemají odděleny piny pro digitální a analogovou zem. Zde vzniká problém s rozhodnutím, na jakou zem celý obvod připojit vzhledem k faktu, že obvod nepracuje pouze s výkonovými částmi a spínacími prvky, ale je zde rovněž digitální komunikace.

7.2 Chladicí plochy

Velkou část desky plošného spoje tvoří chladicí plochy. Největší nároky na chlazení jsou v obvodu zvyšujícího měniče, kde podle parametrů výrobce tranzistorů by měl být dodržen rozměr jednoho palce čtverečního chladicí plochy, stejně tak pro diody.

Další velké chladicí plochy se nachází pod LED diodami. Je zapotřebí si uvědomit, že při proudu 140 mA a úbytku napětí 4,1 V je ztrátový výkon více jak 0,5 W na diodu.

Vzhledem k tomu, že se teplo roztéká rovnoměrně po chladicí ploše na všechny strany, je dost důležité dodržet základní pravidla. Plocha by měla být co nejvíce obdélníková, aby se teplo mohlo roztékat na všechny strany. Není vhodné pokud je na chladicí ploše příliš záhybů či lomů. V žádném případě by chladicí plocha neměla být tenká a táhnout se příliš daleko po plošném spoji. Vhodnější je pokud chladicí plocha obtéká součástku na stejné vrstvě, prokovení a přenášení tepla na druhou vrstvu není tak efektivní. Příklad chladicí plochy je na obrázku 7.3. Tranzistor T2 je součástí zvyšujícího měniče. LED diody ve spodní části obrázku představují část jednoho výstupního kanálu celého prototypu měniče. Pro představu rozměr jedné LED diody je 3,2x2,8 mm.



Obr. 7.3: Příklad chladících ploch pro součástky

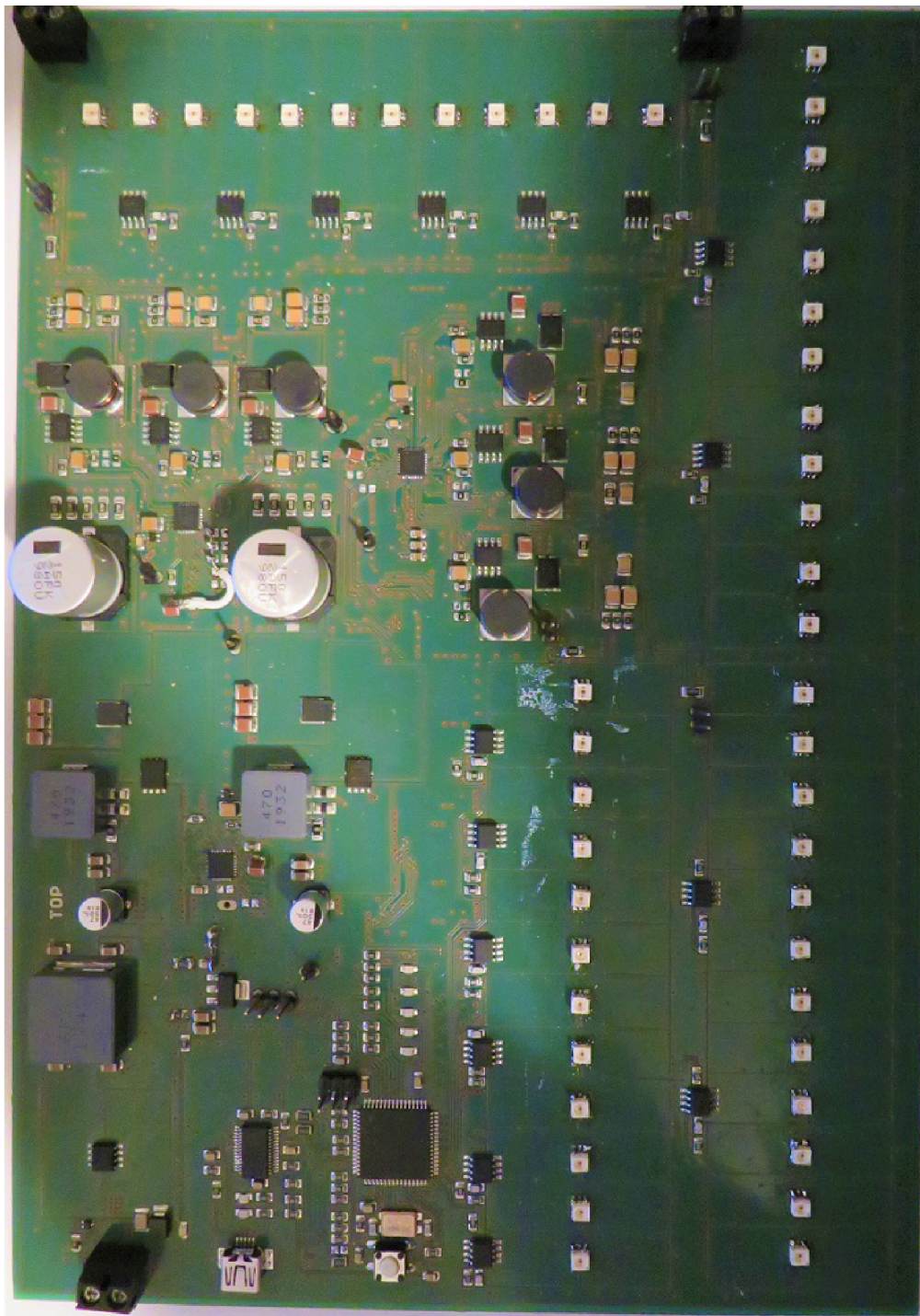
7.3 Výroba a oživení

Deska byla vyrobena společností SAFIRAL podle požadavků. Firma rovněž osadila vyrobenou DPS. Fotografie vyrobeného prototypu jsou na obrázku 7.4 a 7.5. Zadaatel projektu rozhodl, že deska bude vyrobena pouze v jednom kusu. Z toho vyšla některá omezení pro následující práci, mezi které se počítá následující: oživení prototypu je nutné provádět se zvýšenou opatrností, aby nedošlo k nevratnému zničení desky plošného spoje, případné ladění funkčnosti prototypu je nutné provést pouze na úrovni změny součástek, změna chladicích ploch desky není možná.

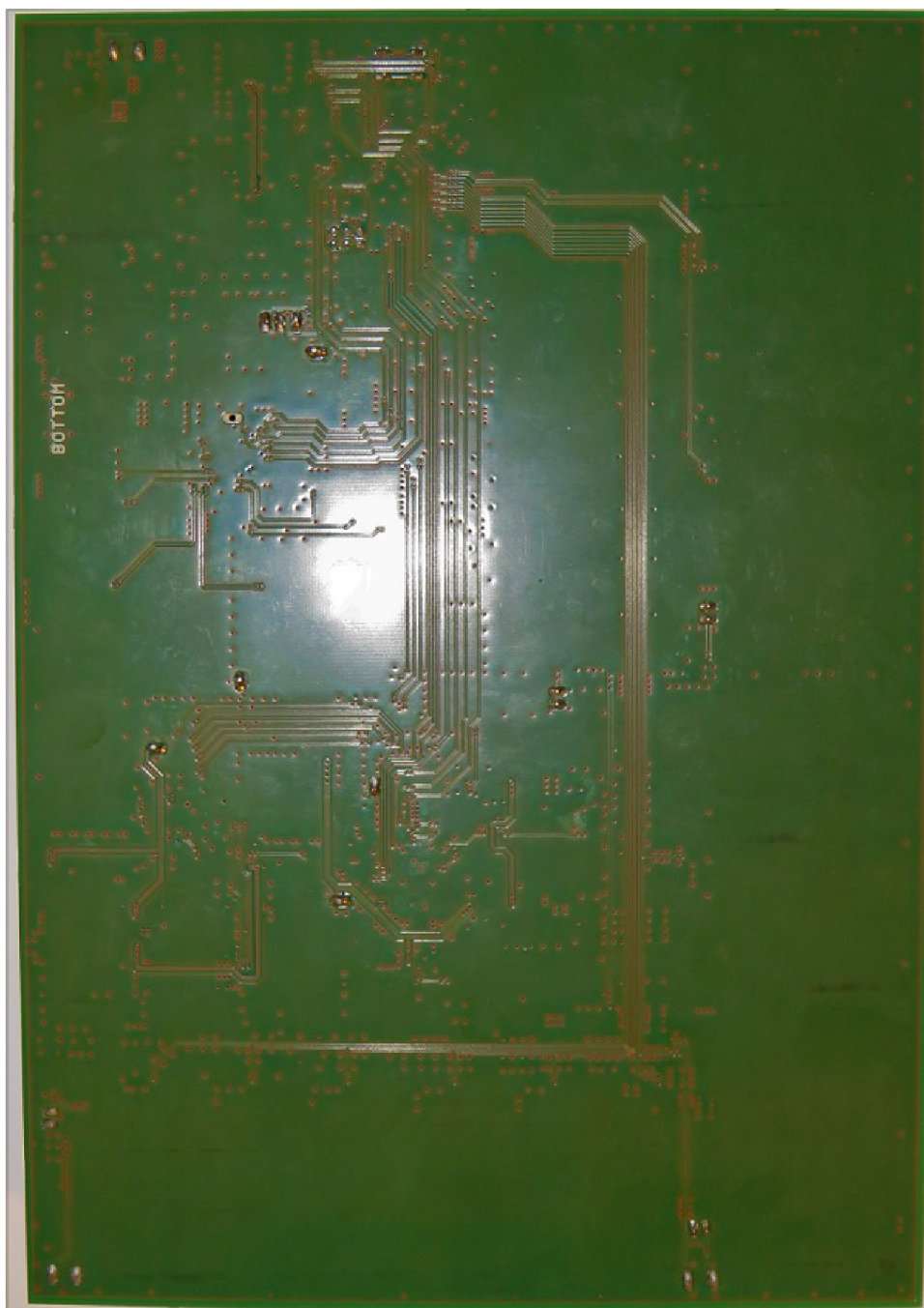
Při oživení se postupovalo v několika krocích. V prvním kroku byl obvod připojen na stabilizovaný zdroj s napětím 5 V a proudovým omezením na několik málo desítek mA. Následně se napětí po malých krocích zvyšovalo a zároveň bylo měřeno výstupní napětí LDO, kterým je napájena digitální část. Tím se ověřila funkčnost regulátoru pro digitální napětí.

Dalším nezbytným krokem bylo ověření funkčnosti mikrokontroleru a komunikace s PC prostřednictvím převodníku FT232. Dále byla ověřena správnost připojení vstupně - výstupních pinů na prototypu. Zároveň byla odzkoušena SPI komunikace. U sběrnice SPI se ukázalo, že kondenzátory 150 pF připojené ke každému vodiči vůči zemi jsou příliš velké a neumožňují komunikační rychlost větší než několik desítek kHz. Byly proto odstraněny. Tímto krokem byla umožněna komunikace s obvodem na hodinovém kmitočtu více jak 1 MHz.

Výše zmíněnými kroky bylo vyzkoušeno, že digitální část návrhu je plně funkční a bylo tak možné přistoupit ke tvorbě uživatelského rozhraní měniče, respektive jeho programové části.



Obr. 7.4: Foto prototypu DC-DC měniče - top



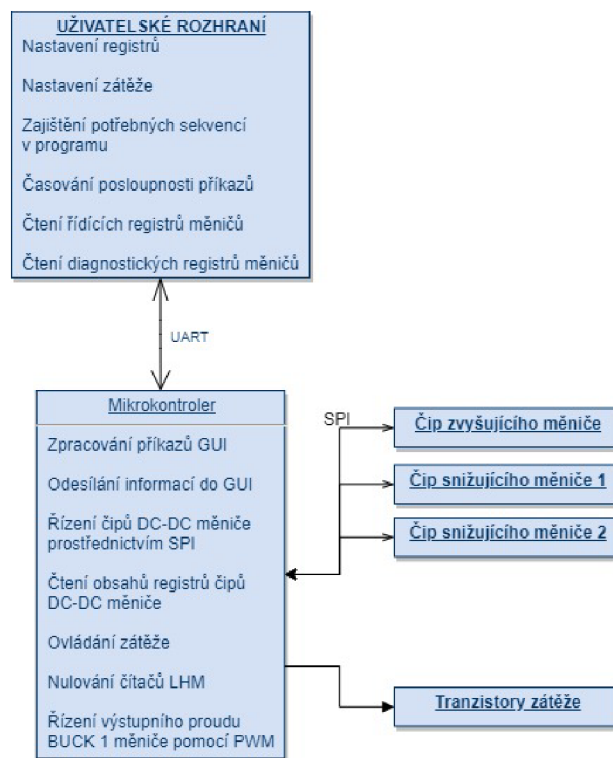
Obr. 7.5: Foto prototypu DC-DC měniče - bottom

8 PROGRAMOVÉ VYBAVENÍ PROTOTYPU

Programové vybavení prototypu se skládá ze dvou hlavních částí. První částí je program pro řídicí mikrokontroler ATmega128 osazený na prototypu DC-DC měniče. Druhou část tvoří grafické uživatelské rozhraní na platformě Windows.

Celek je koncipován následujícím způsobem. Celá logika systému je obsažena v programu uživatelského rozhraní, který realizuje veškeré potřebné operace. To znamená, že veškeré programové sekvence příkazů, které mají daný postup, jsou realizovány v grafickém rozhraní (např. inicializační sekvence měničů, posloupnosti pro programování NVM (Non-Volatile Memory - energeticky nezávislá paměť) paměti měniče apod.).

Z výše uvedeného vyplývá, že hlavním úkolem mikrokontroleru je zpracování přijatých příkazů z rozhraní UART aniž by tyto příkazy jakkoliv modifikoval, či jinak upravil. Dále mikrokontroler zajišťuje nulování časovačů v integrovaných obvodech měniče pro zamezení vstupu do LHM módu. Jeho úkolem je také ovládání tranzistorů zátěže.



Obr. 8.1: Koncepce programového vybavení

Pro přehlednost je koncepce programového vybavení zobrazena graficky na obrázku 8.1. Podrobnější popis SW (Software - programové vybavení) je v následujících podkapitolách.

8.1 ATmega128

Program pro řídicí mikrokontroler je psán v programovacím jazyce C. Pro vývoj programu bylo použito prostředí AtmelStudio. Jak již bylo zmíněno, cílem bylo vytvořit program pro mikrokontroler tak, aby byl pokud možno co nejjednodušší pro zamezení jeho zamrznutí v jakékoli části.

Integrované obvody měniče by měly být programovány s určitou posloupností, nicméně z důvodu omezených paměťových možností mikrokontroleru ATmega128 bylo rozhodnuto, že se hodnoty řídicích registrů nebudou ukládat do např. kruhové paměti a následně se ve správném pořadí programovat. Místo toho je zvolen přístup, že mikrokontroler přijme příkaz po rozhraní UART, ten okamžitě zpracuje a odešle zpět odpověď. Tím se navíc zajistí, že neselhala komunikace SPI na prototypu měniče a požadovaný registr je naprogramován správně.

Mimo tyto hlavní funkce mikrokontroleru jsou zde úseky programu, které zajišťují jiné řídicí funkce.

Jedná se o nulování čítačů v čípech měniče, které v případě přetečení signalizují chybu SPI komunikace, a obvody vstoupí do LHM (Limp-Home Mode). Dále zajišťuje sekvence příkazu pro vystoupení z LHM. Je zde rovněž úsek programu věnovaný ovládání zátěže, ale i řízení PWM (Pulse Width Modulation - pulzně šířková modulace) pro regulaci výstupních kanálů snižujícího měniče č. 1.

Program je rozdělen do několika souborů:

- **uart.h/.c** - převzatá knihovna jejíž autorem je Peter Fleury zajišťující základní funkce k ovládání a nastavení rozhraní UART
- **scheduler.h/.c** - převzatá knihovna z předmětu MMIA použitá pro časování kooperativního systému reálného času
- **sikora_DCDC.h/.c** - knihovna, která obsahuje veškeré ostatní funkce použité v projektu pro řízení prototypu
- **main.c** - hlavní soubor projektu obsahující hlavní funkci *main()*

8.1.1 Kooperativní RTOS

Program mikrokontroleru je koncipován jako kooperativní systém reálného času (RTOS (Real Time Operating System - operační systém reálného času)).

Jedná se o nejjednodušší variantu RTOS. Tvoří ho "supersmyčka" ve funkci *main()*. Jednotlivé úlohy jsou zde ve formě funkcí volané v nekonečné smyčce. Při tomto systému je sdílená paměť mezi úlohami, což vede k nižším požadavkům na paměť RAM oproti preemptivnímu multitaskingu. Problémem je, že systém jako takový nemůže odejmout úloze výpočetní prostředky úloze, to vede na problém, že pokud by nějaká z úloh byla blokující, blokuje se celý systém [19].

O plánování úloh se zde nedá příliš hovořit. Každá funkce (úloha) dostane tolik času, kolik potřebuje, aby mohla skončit. Výhodou je, že vnitřní funkce jsou snadno pochopitelné [19].

Pro realizaci RTOS je použita knihovna *scheduler.h* používaná v předmětu Mikrokontroléry pro pokročilé aplikace. Knihovnu tvoří dvě základní funkce.

```
extern void ResetDelay(uint16_t* const DelayCounter);
extern bool HasDelayElapsed(const uint16_t Delay,
                             uint16_t* const DelayCounter, bool Reset);
```

Následuje zjednodušený příklad použití. Program bliká s LED diodou na pinu PORTA0 s frekvencí 1 sekundy.

```
#define LED_REZ (_BV(PA0))

ISR(TIMER3_COMPA_vect)
{
    TickCounter++;
}

int main (void)
{
    uint16_t delay1=0;
    while(1)
    {
        if(HasDelayElapsed(1000, &delay1, true))
        {
            PORTA ^= LED_REZ;
        }
    }
}
```

Pro správnou funkci systému je třeba využít jednoho časovače mikrokontroleru. V případě tohoto projektu je to 16 bitový Timer/Counter 3. Ten je nastaven tak, aby vyvolal přerušování programu každou 1 milisekundu. To vytváří milisekundový "tick" operačního systému. Funkce *HasDelayElapsed()* následně kontroluje, zda od posledního volání této funkce uběhl zadaný počet milisekund (první vstupní parametr funkce). Druhý vstupní parametr funkce slouží k uložení stavu *TickCounteru* v posledním volání funkce. Poslední parametr povoluje přepsání druhé vstupní proměnné aktuálním stavem *TickCounteru* na konci funkce pomocí zmíněné funkce *ResetDelay()*.

Tímto způsobem je v tomto projektu dosaženo nulování časovačů pro kontrolu správné funkce SPI každých 325 ms (v případě snižujících měničů) a 10 ms (v případě zvyšujících měničů).

```
void TASK_BUCKS_LHM_timer_refresh(uint8_t
```

```

        LHM_refresh_timer_block);
void TASK_BOOST_LHM_timer_refresh(uint8_t
        LHM_refresh_timer_block);

```

Dále tento přístup umožňuje každých 100 ms číst diagnostické registry čipů měničů, a vyhodnocovat informace. Funkce je součástí přílohy D.

```

void TASK_evaluate_diagnostic_registers(void);

```

8.1.2 Hlavní část funkce *main()*

Jak již bylo zmíněno funkce *main()* obsahuje nekonečnou smyčku *while(1)*, ve které je spouštěno několik funkcí v pravidelných časových intervalech. Mimo to je zde nejdůležitější úsek celého systému.

Jedná se o příjem a zpracování příkazů z rozhraní UART. Tato část je stále aktivní a nepodléhá časování kooperativního RTOS. Program na začátku nekonečné smyčky kontroluje, zda byl přijat po rozhraní UART nějaký příkaz. Pokud ne, nevykonává se nic, a systém pokračuje na ostatní úlohy. Pokud však byl přijat příkaz, okamžitě se zpracuje a odešle se odpověď. Tím je zamezeno zahlcení vstupní paměti pro rozhraní UART, protože uživatelské rozhraní nepošle další příkaz, dokud neobdrží odpověď. Úsek kódu zajišťující příjem příkazu je v příloze D. Jedná se o funkci:

```

int8_t ReceiveUartCommand(TPRIKAZ *prikaz);

```

Řetězec znaků, kterým se komunikuje na rozhraní UART má pevně danou délku a formát: "#X!YY&ZZ=DD\$"

- X - je to jeden znak, který určuje zda je to příkaz pro čtení ('R') nebo zápis ('W')
- YY - jsou to dva znaky reprezentující hexadecimální hodnotu. Tato hodnota odlišuje zda se jedná o příkaz pro obvody měniče, nebo interní nastavení mikrokontroleru atd...
- ZZ - je adresa cílového registru, který se má nastavit, nebo konkrétní zátěže která se má ovládat. Opět se jedná o číslo v hexadecimální podobě se dvěma znaky
- DD - jsou data, která je třeba odeslat. Opět se jedná o číslo v hexadecimální podobě se dvěma znaky.

Je vidět, že množství odlišných příkazů, které je možné poslat je velmi velké.

Struktura *TPRIKAZ* obsahuje dílčí části příkazu z GUI rozdělené do jednotlivých proměnných uložených v dekadické podobě.

```

typedef struct {
    uint8_t RW;

```

```

uint8_t TYPE;
uint8_t REG_ADDRESS;
uint8_t VALUE;
}TPRIKAZ;

```

Význam jednotlivých proměnných:

- RW - informace zda přijatý příkaz slouží pro čtení dat či pro zápis
'W' - zápis
'R' - čtení
- TYPE - informace o jaký druh dat se jedná, zda je to příkaz pro čipy měniče, pro ovládání zátěže, LHM mód nebo PWM. Jednotlivé typy jsou definovány v *sikora_DCDC.h*.

```

#define BOOST 0 /*(HEX_ARRAY2DEC("00"))*/
#define BUCK1 16/*(HEX_ARRAY2DEC("10"))*/
#define BUCK2 32/*(HEX_ARRAY2DEC("20"))*/
#define ENABLE_PINS 48 /*(HEX_ARRAY2DEC("30"))*/
#define TRANSISTORS 64 /*(HEX_ARRAY2DEC("40"))*/
#define LHM 80 /*(HEX_ARRAY2DEC("50"))*/
#define PWM 96 /*(HEX_ARRAY2DEC("60"))*/

```

- REG_ADDRESS - obsahuje adresu registru, se kterým se má pracovat, nebo specifikuje, se kterou zátěží se má pracovat, či který kanál PWM se upravuje
- VALUE - číselná hodnota nastavovaného registru, či jiná číselná data

Program zpracovává příkaz v hlavní funkci *main()* zřetězeně. Pro přehlednost je zpracování zjednodušeně vyobrazeno graficky na obrázku 8.2. Schéma neobsahuje podrobné kroky zpracování. Na konci každé větve zmíněného zpracování je znovuvytvoření příkazu pro rozhraní UART a odeslání dat do GUI v počítači.

8.1.3 Komunikace s čipy prostřednictvím SPI

Rozhraní SPI používané pro komunikaci je nastaveno tak, že úroveň hodinového signálu v klidovém stavu je nízká. Data se načítají na sestupnou hranu hodinového signálu. Frekvence hodinového signálu je 1,8432 MHz.

V programu je možné si všimnout, že každý příkaz se na SPI sběrnici pošle vícekrát. Důvodem je to, že bylo zjištěno, že nově nastavená hodnota jednotlivých registru se při čtení spolehlivě odešle na mikrokontroler až při druhém čtení. Proto je čtení po SPI sběrnici zdvojené. Při zápisu do registrů měničů, se jejich nová hodnota vždy přečte a tato hodnota se následně odešle do uživatelského rozhraní. Tím je zajištěno, že byl registr nastaven správně a nedošlo tak k chybě komunikace.

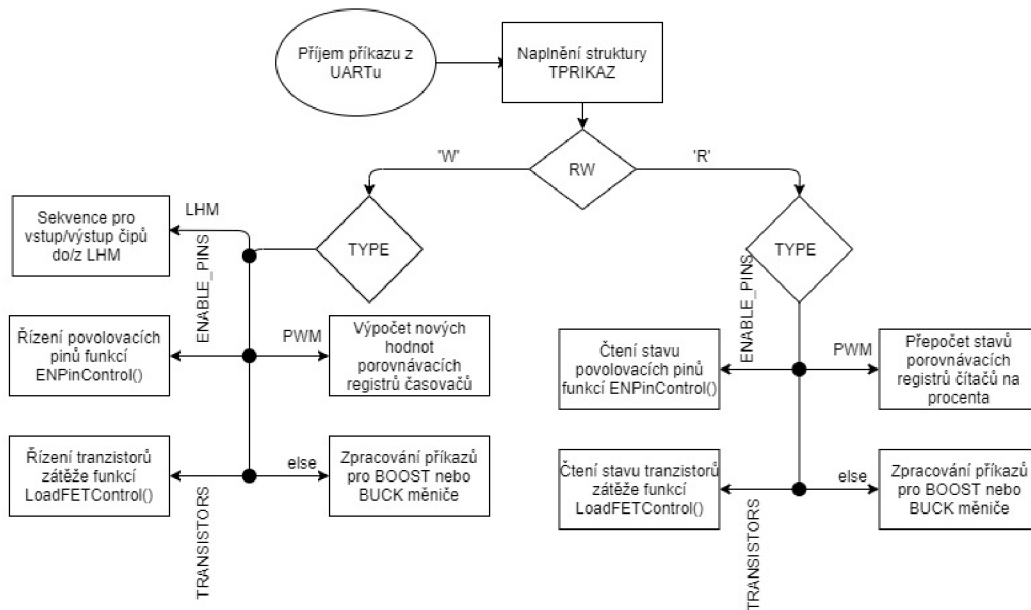
Základní funkcí pro komunikaci na SPI sběrnici je:

```

int8_t SPI_tx_rx(uint8_t type, uint8_t acces,

```

```
uint8_t address, uint8_t value,
uint8_t* returned_value);
```



Obr. 8.2: Grafické zobrazení zpracování příkazu v mikrokontroleru

Celou funkci je možné spatřit v příloze D. Komunikace probíhá v 16 bitových blocích. Nejvýznamnější bit bloku je bit určující zda se jedná o čtení (log. 1) či zápis (log. 0), za ním následuje 7 bitová adresa registru a následně 8 bitů dat.

Vstupními parametry funkce jsou:

- *type* - definuje, který signál "chip select" bude používán. Jsou zde 3 možnosti s použitím již zmíněných definovaných symbolů (BOOST, BUCK1, BUCK2).
- *acces* - 'R' je čtení, 'W' je zápis
- *address* - adresa registru
- *value* - data
- *returned_value* - ukazatel na proměnnou kde se ukládá odpověď čipu

8.1.4 Ovládání zátěže

Jak je patrné ze schématu obvodu, ovládání zátěže je realizováno několika tranzistory, které jsou připojeny k pinům mikrokontroleru. Jedná se o piny PA3-PA7, PC0-PC7 a PG0-PG2. Regulace funguje tak, že pouze jeden tranzistor v příslušné zátěži může být otevřen. Data z uživatelského rozhraní se tedy posílají tak, aby část příkazu *TYPE* určovala, že se pracuje se zátěží. V položce *REG_ADDRESS* je určen kanál se kterým se pracuje. Položka *VALUE* pak obsahuje na některé své pozici log. 1. Funkce *LoadFETControl()* tento příkaz následně zpracuje. Pro představu jak

zmíněná funkce pracuje, je její část zobrazena v příloze D. Funkce je deklarována následovně:

```
uint8_t LoadFETControl(uint8_t acces, TPRIKAZ prikaz);
```

8.1.5 PWM regulace

Další úlohou mikrokontroleru je PWM (Pulse Width Modulation - pulzně šířková modulace) regulace tří ze šesti výstupních kanálů. Vodiče pro PWM jsou připojeny na piny procesoru, na kterých je možné vytvořit hardwerovou PWM regulaci. Ta se realizuje vnitřními časovači/čítači. ATmega128 disponuje čtyřmi těmito hardwarovými prvky.

- **PWM1** - pin PB4 - OC0 pro časovač/čítač 0
- **PWM2** - pin PB5 - OC1A pro časovač/čítač 1
- **PWM3** - pin PB6 - OC1B pro časovač/čítač 1

Je vidět, že jedna z PWM regulací je realizována 8 bitovým časovačem/čítačem 0, který nabízí mnohem méně možností než 16 bitový časovač/čítač 1. Aby se všechny tři signály chovaly stejně, bylo zapotřebí zvolit takový způsob generace PWM signálu, který umožní oba zmíněné časovače.

Byl zvolen tzv. *Phase Correct PWM* režim. Ten se vyznačuje tím, že umožňuje 0 % i 100 % střidu a má větší rozlišení než FastPWM mód, takže nastavení střidy je přesnější.

Funkce generátoru je následující. Čítač se opakovaně inkrementuje od své minimální hodnoty (0x00) do maximální (0xFF) a dekrementuje zpět k minimu. Každý čítač má pro příslušný kanál tzv. porovnávací registr (Output Compare Reg.). Pokud hodnota čítače dosáhne hodnoty porovnávacího registru při inkrementaci tak se nastaví příslušný výstupní pin, pokud při dekrementaci, tak se tento pin vynuluje.

Z toho plyne, že regulace frekvence PWM signálu není příliš jemná, a je dána vztahem:

$$f_{\text{PWM}} = \frac{f_{\text{OSC}}}{2 \times N \times \text{TOP}} [\text{Hz}] \quad (8.1)$$

Hodnoty *TOP* a *f_{OSC}* jsou dány (0xFF a 7372800 Hz). Jedinou možností, jak ovlivnit frekvenci PWM, je dělička *N*, která v případě ATmega128 je 1, 8, 64, 256, nebo 1024. Vzhledem k tomu, že frekvence PWM signálu pro čip ASL3417SHN musí být v rozmezí 1Hz - 1kHz, byla zvolena dělička *N* = 64. Což dává výslednou frekvenci 226 Hz.

Oproti tomu regulace střidy může být mnohem jemnější. Je však nutné počítat hodnoty porovnávacích registrů. Aby byl její výpočet bez zaokrouhlovacích chyb mezivýsledku, je pro tento výpočet jako jediný použitý formát s pohyblivou řádovou čárkou. Nicméně vzhledem k velmi ojedinelým nutnostem výpočtu to není problém. Příklad výpočtu se nachází níže.

```

//default setting PWM for PWM3
inter_res_pwm_calc=PWM3_value*TOP_PWM_COUNTER / 100;
inter_res_pwm_calc =TOP_PWM_COUNTER - inter_res_pwm_calc;
OCR1B=((uint8_t)round(inter_res_pwm_calc)) & 0xFF;

```

Hodnota *PWM3_value* je hodnota střídy (0-100). *TOP_PWM_COUNTER* je hodnota vrcholu čítače (0xFF).

Další důležitou schopností časovače/čítače je, že při dosažení minimální hodnoty při dekrementaci může vyvolat přerušeni. Je samozřejmé, že přerušeni je ideální dobou pro změnu porovnávacích registrů. Jenže zde vznikl problém. Dva čítače vyvolávaly přerušeni tak často, že docházelo k pádům systému kvůli neúspěšné UART komunikaci. Řešením bylo, aby se přerušeni vyvolalo pouze v případě, kdy je nutné aktualizovat porovnávací registr. Proto program pracuje i s registrem TIMSK, který povoluje a zakazuje přerušeni za běhu programu. Hodnota porovnávacího registru se počítá ve funkci *main()* z důvodu, že přerušeni má trvat jen velmi krátkou dobu a výpočty s pohyblivou řádovou čárkou jsou poněkud náročné. V přerušeni dochází pouze k aktualizaci porovnávacího registru a k zakázání přerušeni. Níže je ukázka vektoru přerušeni jednoho kanálu.

```

/*vektor preruseni pro citac0*/
ISR(TIMERO_OVF_vect)
{
    OCR0=OCR0_new_content;
    PWM0_change=0; //pomocna promenna
    TIMSK &= ~_BV(TOIE0); //zakazani preruseni
}

```

Následuje ukázka nastavení střídy PWM jednoho kanálu.

```

else if(rx_message.TYPE == PWM)
{
    if (rx_message.VALUE <=100 && rx_message.VALUE >=0)
    {
        /*Osetreni aby nedoslo k vypoctum dokud se
        predchozi prijata hodnota nenastavi*/
        if (PWM0_change == 0 && PWM1_change == 0)
        {
            if (rx_message.REG_ADDRESS == 1)
            {
                PWM1_value = (float)rx_message.VALUE;
                inter_res_pwm_calc=PWM1_value*TOP_PWM_COUNTER / 100;
                inter_res_pwm_calc=TOP_PWM_COUNTER-inter_res_pwm_calc;
                OCR0_new_content=((uint8_t)round(inter_res_pwm_calc))&0xFF;
                PWM0_change=1;
                TIMSK |= _BV(TOIE0); //povoleni preruseni Timer/Counter0
            }else if (rx_message.REG_ADDRESS == 2) . . .
        }
    }
}

```

8.1.6 Sekvence pro ovládání LHM módu

Limp-home mode je stav integrovaných obvodů měniče, kdy se odhalí chyba SPI komunikace. Pokud taková chyba nastane, obvod automaticky načte nastavení všech registrů z energeticky nezávislé NVM paměti a pracuje v jakémsi omezeném režimu.

Funkčnost komunikace SPI se kontroluje tak, že v určitých časových intervalech se odesílá čipům měniče příkaz pro vynulování interních časovačů. Pokud tyto časovače z nějakého důvodu přetečou, je vyhodnocena chyba SPI. Nulování zajišťují funkce:

```
void TASK_BUCKS_LHM_timer_refresh(uint8_t
                                   LHM_refresh_timer_block);
void TASK_BOOST_LHM_timer_refresh(uint8_t
                                   LHM_refresh_timer_block);
```

Vstupní proměnná *LHM_refresh_timer_block* je používána právě pro možnost vstoupit do LHM módu. Pokud je nastavená na log. 1 prostřednictvím GUI, tak se obnovení časovačů v čípech nekoná, a obvod vstoupí do LHM.

Naopak pokud je třeba z LHM vystoupit, je třeba odeslat určitou sekvenci příkazů. V případě zvyšujícího měniče se jedná o registr na adrese 0x3C a v případě snižujícího pak 0x33.

Příklad sekvence pro výstup z LHM módu u zvyšujícího měniče se nachází níže:

```
else if (rx_message.TYPE == LHM)
{
    if(rx_message.VALUE == 0)
    {
        LHM_refresh_timer_block=0;
        //boost exit LHM seq
        SPI_tx_rx(BOOST, 'W', HEX_ARRAY2DEC("3C"),HEX_ARRAY2DEC
                ("16"), &SPI_received_value);
        SPI_tx_rx(BOOST, 'W', HEX_ARRAY2DEC("3C"),HEX_ARRAY2DEC
                ("26"), &SPI_received_value);
        SPI_tx_rx(BOOST, 'W', HEX_ARRAY2DEC("3C"),HEX_ARRAY2DEC
                ("46"), &SPI_received_value);
        SPI_tx_rx(BOOST, 'W', HEX_ARRAY2DEC("3C"),HEX_ARRAY2DEC
                ("76"), &SPI_received_value);
        .
        .
        .
    }
    .
    .
    .
}
```

8.2 Uživatelské rozhraní

Uživatelské rozhraní neboli GUI (Graphic User Interface - grafické uživatelské rozhraní) je důležitou součástí práce. Jedná se o základní část projektu, bez které by nebylo možné měnič jakýmkoliv způsobem řídit.

Požadavek je, aby bylo uživatelské rozhraní jednoduché na ovládání, mělo by umožňovat libovolné nastavení vnitřních registrů integrovaných obvodů měničů. GUI by dále mělo umožňovat čtení a výpis diagnostických registrů, možnost nastavení PWM regulace či řízení zátěže.

8.2.1 Python

Existuje mnoho programovacích jazyků, ve kterých lze tvořit aplikace. Pro tento projekt byl vybrán vysokoúrovňový, dynamicky interpretovaný (skriptovací) jazyk Python. Jedná se o multiplatformní jazyk umožňující širokou škálu možností např. při práci s datovými typy.

Tento jazyk vyniká svou jednoduchostí z hlediska učení. Například stručností zápisu, tvorbou bloků pouhým odsazováním, velkou dostupností knihovních modulů. Značnou výhodou je velká základna příznivců Pythonu, není proto problém se získáváním informací.

V posledních letech zažívá programovací jazyk Python nárůst popularity a je využíván mnoha vývojáři. To je hlavním důvodem využití tohoto jazyku v diplomové práci. Ačkoliv nemám s tímto jazykem předešlé zkušenosti myslím, že je vhodné se s ním seznámit. Další výhodou je, že se jedná o open-source projekt, takže veškeré potřebné nástroje pro vývoj jsou dostupné zdarma.

V práci je použita verze 3 tohoto jazyka. Grafická nadstavba je realizována pomocí *Tkinter* modulu.

8.2.2 Základní koncepce

Uživatelské rozhraní zajišťuje veškeré řídicí funkce. Je proto poměrně rozsáhlé. Aby bylo ovládání přehledné, je rozdělené do tří hlavních oken.

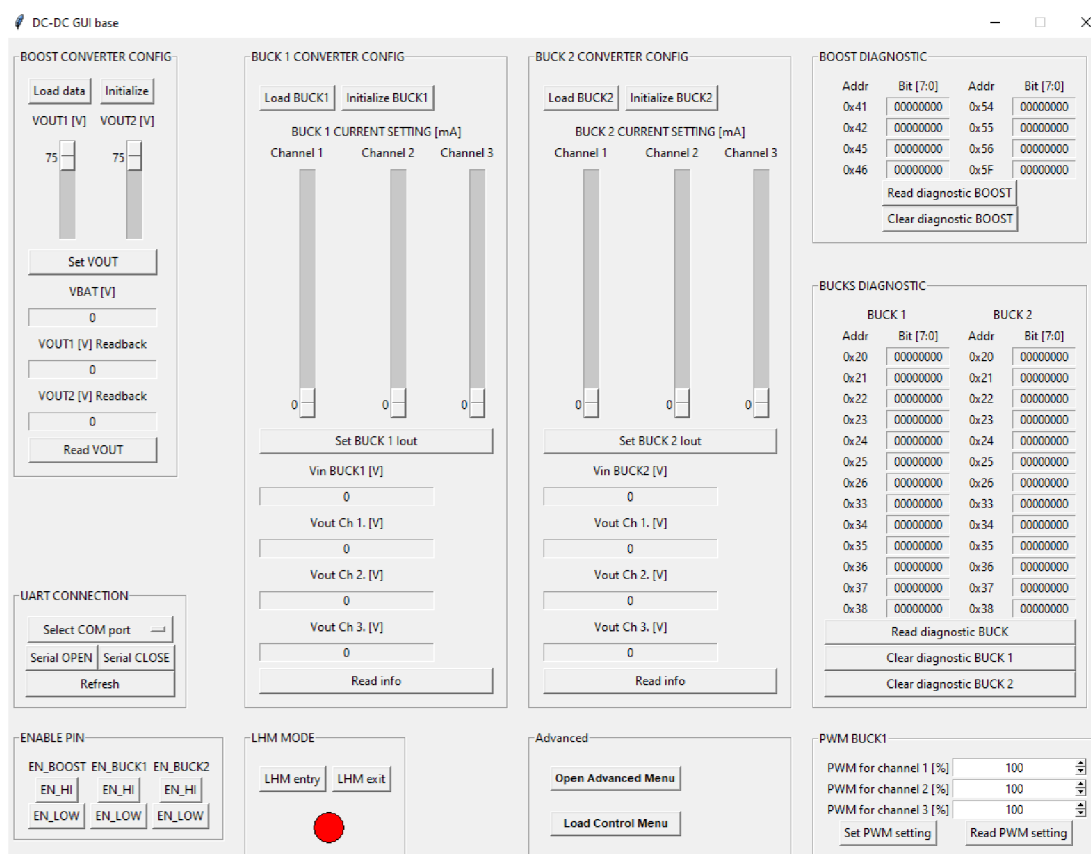
- **Hlavní okno**
- **Pokročilé nastavení**
- **Řízení zátěže**

Pro základní ovládání slouží hlavní okno, které je možné spatřit po spuštění programu. Zde lze pohodlně načíst ze souboru hodnoty registrů čipů měniče a následně čipy inicializovat. Následně lze pohodlně regulovat výstupní napětí zvyšujícího měniče a výstupní proudy jednotlivých kanálů snižujících měničů. Rovněž jsou v hlavním okně části pro čtení diagnostických registrů.

Pro pokročilejší uživatele, kteří znají jak čipy pracují, je možné vejít do pokročilého menu tlačítkem *Open Advanced Menu*. Otevře se okno kde je možné vidět jak jsou nastavené jednotlivé registry, toto nastavení upravit, naprogramovat energeticky nezávislé paměti, vytvářet soubory s nastavením registrů.

Poslední okno slouží pouze k řízení zátěže. V něm je zobrazeno aktuální nastavení výstupních řetězců LED diod.

8.2.3 Hlavní okno



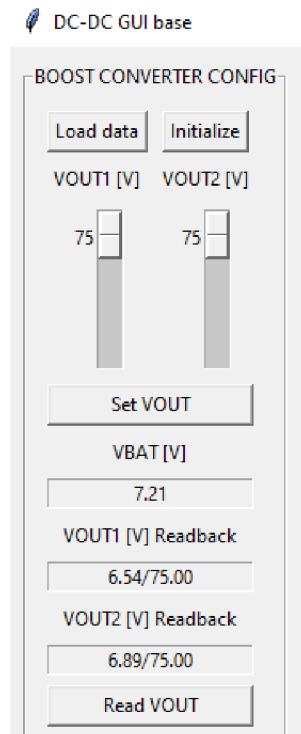
Obr. 8.3: Vzhled uživatelského rozhraní - hlavní okno

Na obrázku 8.3 je možné vidět vzhled hlavního okna uživatelského rozhraní. Jak je možné vidět, ovládání jednotlivých částí je rozděleno do sekcí. V dalším textu jsou jednotlivé sekce popsány.

BOOST CONVERTER CONFIG

Jedná se o část GUI, která zajišťuje řízení zvyšujícího měniče. Pomocí tlačítka *Load data* je možné naimportovat inicializační data pro zvyšující měnič ze souboru. Sou-

bory pro inicializaci jsou součástí programu. Inicializační data jsou uloženy v textovém souboru. Více informací o tvorbě souborů je v textu dále.



Obr. 8.4: Vzhled uživatelského rozhraní - výřez nastavení BOOST

Samotné importování dat nemá na měnič vliv. Čip měniče se nakonfiguruje až po zmáčknutí tlačítka *Initialize*. Po stisku tohoto tlačítka se jako první zavolá funkce, která aktivuje čipy signálem *Enable*. Důvodem je zjednodušení, aby uživatel nemusel při každém zapnutí měniče ručně aktivovat tyto signály. Po této aktivaci jsou nakonfigurovány všechny registry měniče v doporučeném pořadí, kromě těch, které uchovávají informaci o výstupním napětí a o výstupní přepětové ochraně.

Dále jsou zde dva posuvníky, kterými lze nastavit hodnotu zmíněných registrů. Podle doporučení výrobce se po zmáčknutí tlačítka *Set VOUT* odesílá informace do měničů ve smyčce. Při každém oběhu smyčky se inkrementuje hodnota nastaveného výstupního napětí o 2, až se dosáhne požadované hodnoty, smyčka končí. Tím se zajistí, že v obvodu měniče nedochází k velkým proudovým špičkám. Navíc se jako první konfiguruje výstupní kanál 1 a až když dosáhne požadovaného napětí, konfiguruje se kanál 2. Opět pro zamezení proudových špiček.

Následně jsou v této části GUI 3 informační boxy, které se naplní po zmáčknutí tlačítka *Read VOUT*. Pokud není funkční komunikace, tak tlačítko nereaguje.

- **VBAT[V]** - Informace, kterou poskytuje čip zvyšujícího měniče. Jedná se o naměřenou velikost napájecího napětí.

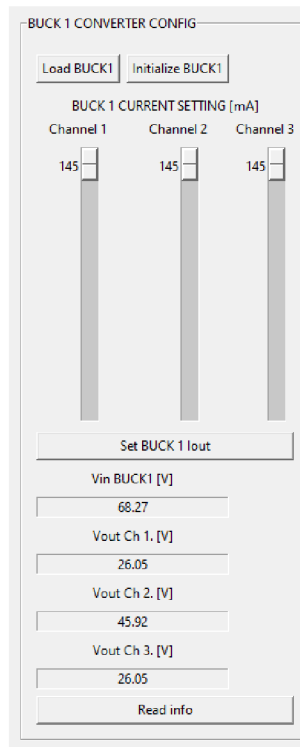
- **VOUTx Readback[V]** - Zde se hodnota zobrazuje ve formátu Y/Z, kde Y je reálně naměřená hodnota ve Voltech, kterou poskytují čipy snižujícího měniče a Z je velikost napětí ve Voltech, jaká je nastavená v registru zvyšujícího měniče. Samotný zvyšující měnič neposkytuje reálně naměřené hodnoty napětí na svém výstupu.

Funkce, které se volají v této sekci programu jsou následující:

```
def func_boost_button_load_settings(self):
def func_boost_button_initialize_io(self):
def func_boost_button_set_vout(self):
def func_boost_button_read_vout_readback(self):
```

Všechny funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

BUCKx CONVERTER CONFIG



Obr. 8.5: Vzhled uživatelského rozhraní - výřez nastavení BUCK

Tato sekce zajišťuje řízení snižujících měničů. Opět je možné importovat data pro inicializaci čipů pomocí tlačítka *Load BUCKx*. Způsob načítání je obdobný jako u zvyšujícího měniče.

Po zmáčknutí tlačítka *Initialize BUCKx* se opět čipy resetují pinem Enable, dále se nakonfigurují potřebné registry. Na konci funkce se výstupní kanály deaktivují v registru na adrese 00h. Zároveň se aktualizují diagnostické registry.

Další části této sekce jsou posuvníky, které umožňují nastavení proudu výstupními kanály. Těmito posuvníky lze nastavit výstupní proud kanálem v rozsahu 0-145 mA. Je ale nutné vědět, že minimální výstupní proud kanálem se rovná nastavené hysterezi výstupního proudu, která je v inicializačních souborech nastavená na přibližně 30 mA. To znamená, že pokud je nastavená hodnota nižší, tak je proud v kanále odlišný od nastaveného.

Nastavení se provede stisknutím tlačítka *Set BUCK x Iout*. Tím se zavolá funkce, která nastaví příslušné registry na požadovanou hodnotu. Pokud je nastaven posuvník na 0, příslušný kanál je ve funkčním registru odpojen.

V další části této sekce jsou opět informační boxy, které zobrazují (po stisku tlačítka *Read info*) naměřené hodnoty vstupního a výstupních napětí. Tyto hodnoty měří čipy snižujícího měniče. Je proto opět nutný přepočítání z hodnoty registru na hodnotu ve Voltech.

Funkce, které se volají v této sekci programu jsou následující:

```
def func_buck1_button_load_settings(self):
def func_buck2_button_load_settings(self):

def func_buck1_button_initialize_io(self):
def func_buck2_button_initialize_io(self):

def func_buck1_button_set_iout(self):
def func_buck2_button_set_iout(self):

def func_buck1_button_read_info_readback(self):
def func_buck2_button_read_info_readback(self):
```

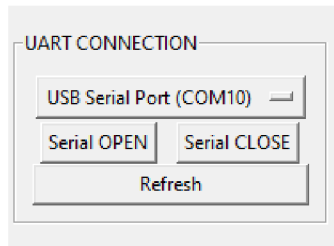
Všechny funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

UART CONNECTION

V této části GUI se ovládá připojení k sériové lince. Nachází se zde rozbalovací menu, ve kterém jsou vypsané připojené (dostupné) COM porty. Je zde tlačítko Refresh, které tento seznam aktualizuje. Dále jsou zde tlačítka pro navázání spojení (*Serial OPEN*) i pro ukončení spojení (*Serial CLOSE*). Tlačítka volají funkce:

```
def serialOpen(self):
def serialRefresh(self):
def closePort(self):
```


První dvě zmíněné funkce jsou součástí třídy *App(object)* v souboru *GUI.py*. Poslední funkce, ale i vnitřní funkce dvou zmíněných jsou součástí třídy *RS232(object)* v souboru *rs232.py*.



Obr. 8.6: Vzhled uživatelského rozhraní - výřez nastavení UART

V tomto souboru (třídě) jsou definovány veškeré funkce nutné k ovládání sériové linky. Je zde její inicializace, veškerá nastavení i všechny funkce pro odesílání a příjem dat. Funkce jsou okomentovány v kódu. Nejdůležitější funkce,

```
def uartTX_RX(self, NUM_OF_TRY, RW, TYPE, ADRESS, VALUE):
```

která odesílá příkazy je rovněž v příloze E. Vysvětlení její funkce je v kapitole 8.2.6.

ENABLE PIN

V této sekci je možné přímo ovládat piny procesoru, které zajišťují povolování integrovaných obvodů měniče (tzv. Enable signals).

Po zmačknutí je volaná jedna z následujících funkcí:

```
def func_enable_pin_HI(self, TYPE):
def func_enable_pin_LOW(self, TYPE):
```

Kromě nastavení příslušného pinu, je po aktivaci čipu vysokou úrovní volaná funkce, která zajistí výstup obvodu z LHM módu.

Vzhledem k tomu, že funkce jsou používány i v jiných částech kódu, nemá každé tlačítko svou vlastní funkci, ale úkony, které vykoná jedna ze zmíněných funkcí se liší podle vstupního parametru *TYPE*. Význam parametru je stejný jako popsaná proměnná *TYPE* v sekci 8.1.2. Obě funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

LHM MODE

Tato jednoduchá část umožňuje uměle simulovat nefunkční SPI komunikaci. Při aktivaci tlačítkem *LHM entry* se volá funkce:

```
def func_boost_button_LHM_entry(self):
```

Funkce zablokuje resetování interních časovačů čipů měniče (popsáno v sekci 8.1.6) a tím se aktivuje LHM mód.

Zmáčknutím tlačítka *LHM exit* je volaná funkce:

```
def func_boost_button_LHM_exit(self):
```

Následně se odešle do mikrokontroleru příkaz pro spuštění deaktivační sekvence LHM (popsáno v sekci 8.1.6). Po vystoupení z LHM je uživatel dotázán zda chce resetovat zařízení pomocí *Enable* pinů a docílit tak výchozího stavu, nebo chce ponechat nastavení z LHM, které zde po výstupu z LHM přetrvává. Obě funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

PWM BUCK1

Sekce programu vytvořena pro ovládání střidy PWM signálu. Jsou zde 3 tzv. "spin-boxy" pro zadávání celočíselné střidy v rozmezí 0-100. Tento vstup je ošetřen proti zadání nesprávných hodnot.

Tlačítkem *Set PWM setting* se volá funkce,

```
def func_set_PWM_setting(self):
```

kteřá odešle nastavení všech 3 PWM kanálu. Po sériové lince se posílá hodnotu 0-100 převedenou do hexadecimální podoby, kterou mikrokontroler zpracuje podle potřeby.

Rovněž je zde tlačítko *Read PWM setting*, které volá funkci:

```
def func_read_PWM_setting(self):
```

Ta přečte aktuální stav nastavení PWM v mikrokontroleru. Použije se v případě, kdyby došlo ke znovu spuštění aplikace, bez restartu desky do výchozí polohy.

Obě funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

BOOST and BUCKS DIAGNOSTIC

Tyto důležité sekce zajišťují uživateli možnost přečíst diagnostické registry čipů na prototypu. Tlačítkem *Read* se volá jedna z funkcí

```
def func_diagnostic_button_buck_read(self):
def func_diagnostic_button_boost_read(self):
```

Po přečtení se vypisují data v binární podobě, aby bylo pro uživatele jednodušší vyhodnocovat konkrétní bity s datovým listem součástky. Součástí zmíněných funkcí je volání funkce:

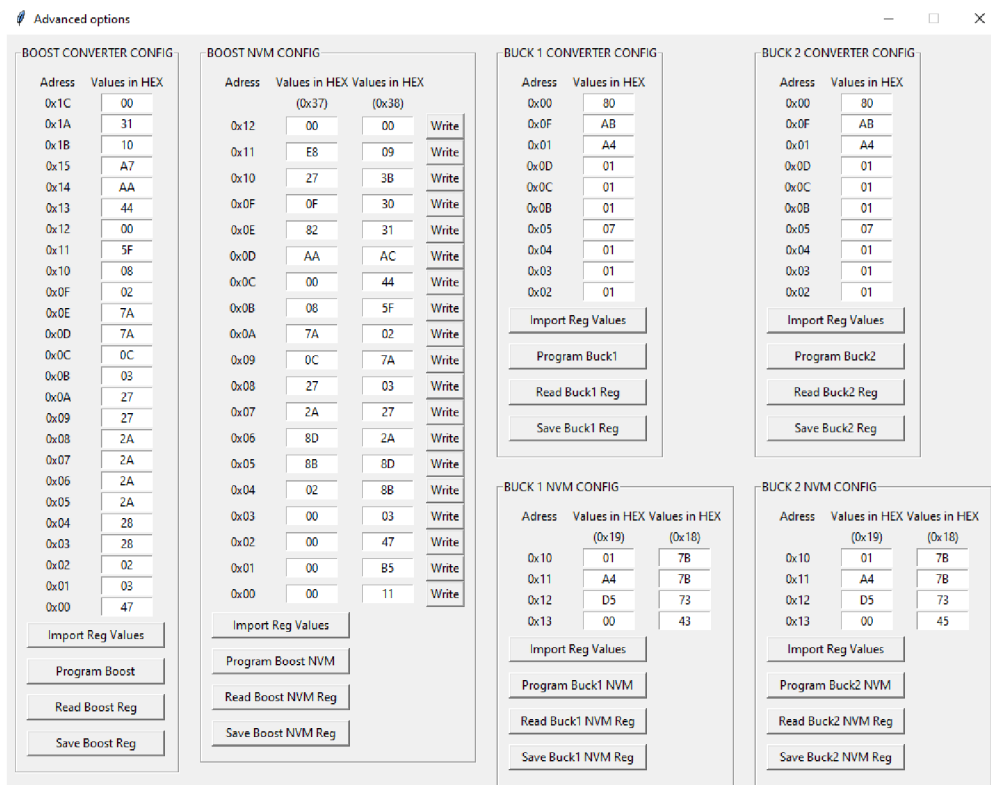
```
def actualizeDiagnosticEntry(self):
```

Tato funkce zajišťuje převod hexadecimálních hodnot uložených v paměti programu do binární podoby a jejich následný výpis.

Dále tato sekce umožňuje nulování (refresh) diagnostických registrů tlačítkem *Clear*. Pokud vznikne v čipu chybový stav, je nutné, pro správnou funkci, tyto registry zresetovat. Všechny funkce jsou součástí třídy *App(object)* v souboru *GUI.py* v příloze.

8.2.4 Pokročilé nastavení

Pokročilé nastavení je určeno pro uživatele, kteří jsou seznámeni s funkcí čipů, významem jednotlivých registrů a principem fungování. Vzhled pokročilého nastavení



Obr. 8.7: Vzhled uživatelského rozhraní - pokročilé nastavení

je možné vidět na obrázku 8.7.

Okno pokročilého nastavení je vytvořeno jako samostatná třída *APP_REG_SET (object)* v souboru *register_settings_gui.py*. Tento soubor tedy obsahuje všechny funkce využívané tímto oknem.

Pokročilé nastavení je opět rozděleno na několik částí reprezentujících ovládání zvyšujícího či snižujícího měničů. Toto okno by se dalo rozdělit na dvě základní části:

- BOOST/BUCK_x CONVERTER CONFIG - přístup k řídicím registrům příslušného integrovaného obvodu

- BOOST/BUCKx NVM CONFIG - přístup k energeticky nezávislé paměti příslušného integrovaného obvodu, která obsahuje konfiguraci pro případ vstupu do LHM módu

Tyto části fungují podobně, proto je v následujícím textu popsána pouze konfigurace zvyšujícího měniče.

BOOST CONVERTER CONFIG

V levém sloupci jsou vypsány adresy jednotlivých registrů a v pravém sloupci jsou obsahy registrů spojené s těmito adresami. Uživatel může zadávat hexadecimální hodnoty, které se následně zapíše do odpovídajících registrů. Zadávání je ošetřeno proti zápisu jiných než hexadecimálních hodnot v délce dvou znaků.

Zadané hodnoty je možné uložit tlačítkem *Save Boost Reg* a vytvořit tak inicializační soubor. Rovněž je možné takový soubor načíst tlačítkem *Import Reg Value*.

Je důležité vědět, že pokud se inicializuje kterýkoliv měnič nebo jakkoliv programuje, posílají se tam data vypsané právě v těchto zadávacích boxech. Jediné co neodpovídá hodnotám ve zmíněných boxech jsou hodnoty výstupních napětí a přepětových ochran (zvyšující měnič), výstupních proudu (snižující měniče). Pokud chce uživatel znát reálné hodnoty právě těchto registrů, je nutné použít tlačítko *Read Boost Reg*. Tím se zavolá funkce pro přečtení registrů čipu a přečtené hodnoty se následně uloží do interní paměti uživatelského rozhraní. Po tomto kroku jsou všechny hodnoty v zadávacích boxech aktuální.

Pokud uživatel použije tlačítko *Program Boost* musí vědět, že funkce, která je následně volaná nakonfiguruje měnič (odešle hodnoty pro zápis) bez ohledu na to v jakém stavu se měnič nachází. V tom se liší od "Inicializace" v hlavním okně programu, která na začátku resetuje čipy do výchozího stavu. Pokud je v módu ve kterém by registry nebyly přístupné pro zápis, tak se nenakonfigurují. Proto je vhodné po naprogramování přečíst hodnoty zpětně a zkontrolovat obsah.

Funkce volané tlačítka v této sekci, a sekcích pro snižující měnič jsou:

```
def func_read_boost_registers(self):
def func_read_buck1_registers(self):
def func_read_buck2_registers(self):

def func_program_boost_reg(self):
def func_program_buck1_reg(self):
def func_program_buck2_reg(self):

def saveRegValue(self, type_of_data):
def importRegValue(self, type_of_data):
```

A funkce ošetřující vstupy je:

```
def callback(self):
```

BOOST NVM CONFIG

Tato část je z pohledu kódu o něco komplikovanější. Práce s NVM (Non-Volatile Memory - energeticky nezávislá paměť) paměti je poněkud složitější. K této paměti lze přistupovat pouze pokud se čip nachází v *NVM acces mode*. Problémem je, že programování této paměti má jiný postup v případě zvyšujícího měniče i v případě snižujících. Faktem je, že z uživatelského hlediska je tento rozdíl bezvýznamný. V kódu jsou pro práci s NVM paměti následující funkce:

```
def func_boost_entry_NVM_acces_mode(self):
def func_boost_exit_NVM_acces_mode(self):
def func_program_one_boost_NVM(self, address_of_register_in_JSON_dic):
def func_program_all_boost_NVM(self):
def func_read_one_boost_NVM(self, address_of_register_in_JSON_dic):
def func_read_all_boost_NVM(self):
def func_read_buck_nvme(self, TYPE):
def func_write_buck_nvme(self, TYPE):
```

Tyto funkce jsou poměrně dlouhé. Podstatou ale je, že dodržují doporučené postupy pro zápis do NVM paměti, či její čtení, zveřejněné v datových listech součástek. V souboru *register_settings_gui.py* je postup pro dobré pochopení okomentován.

Co se týká rozdílu oproti nastavování klasických registrů, tak při čtení či programování NVM paměti je uživatel dotazován na případnou nutnost resetovat čipy. To je nutné pro správné naprogramování paměti.

Větší rozdíl je možnost, v případě zvyšujícího měniče, programovat registry zvlášť (po jednom). To má jeden hlavní důvod. Energeticky nezávislá paměť zajišťuje svou funkčnost do 200 zápisů. Vzhledem k tomu, že registrů pro zápis je 19, kompletní zápis by bylo možné provést pouze 10 krát. Na druhé straně, pokud zapisuje uživatel registry po jednom, každý zápis registru znamená reálně 2 zápisy do NVM. Musí se totiž zapsat i novou hodnotu do paměti, kde je uložen počet zápisů. U snižujících měničů je registrů k zápisu tak málo, že nemá význam způsob zápisu po jednom registru realizovat.

Z pohledu uživatele je ještě rozdíl v zadávání hodnot. Je zde horní a spodní byte dat. Zapsání obou bytů dat se bere jako jeden zápis do paměti NVM.

8.2.5 Řízení zátěže

Na obrázku 8.8 je možné vidět vzhled okna pro ovládání zátěže. Zátěž lze regulovat pomocí tranzistorů připojených na piny procesoru. Jedná se tedy o regulaci LED diod osazených na desce. Okno je rozděleno na dvě části. Jedna je pro snižující měnič 1 a druhá pro zbývající.



Obr. 8.8: Vzhled uživatelského rozhraní - řízení zátěže

Regulaci se provádí změnou polohy posuvníků. Změna polohy okamžitě mění nastavení v mikrokontroleru. Zelené značky symbolizují rozsvícené LED diody, naopak šedé značí odpojenou LED diodu.

Z principu zapojení je jasné, že přípravek umožňuje odpojovat LED diody jen určitým způsobem. Tomu odpovídají i pozice, do jakých jde posuvníky nastavit. Zátěž 1 obou měničů lze zmenšovat po dvou diodách mimo první tranzistor, který odpojí pouze jednu LED diodu. Zátěž dva má pouze 2 tranzistory, a proto lze nastavit pouze tři způsoby svícení LED řetězce. Zátěž 1 s 6 tranzistory dokáže vytvořit sedm různě dlouhých LED řetězců.

Okno pro řízení zátěže je opět vytvořeno jako samostatná třída *LOAD_CONTROL* (*object*) v souboru *load_control.py*. Zde jsou i potřebné funkce.

Okno nabízí rovněž možnost přečíst stav zátěže v případě znovu-spuštění programu bez restartu zařízení.

Základními funkcemi volanými po změně polohy posuvníku či zmáčknutí tlačítka jsou:

```
def func_change_led(self, var):
def func_read_load(self):
def func_send_load(self):
```

8.2.6 Důležité části kódu

Program uživatelského rozhraní je celkově intuitivní. Každé okno je reprezentováno jedním souborem projektu (jednou třídou).

Vzhledem ke stylu psaní kódu uživatelského rozhraní, je v první části každého souboru vytvořeno mnoho grafických prvků (tlačítka, popisky, posuvníky, ...). Ty následně při určité akci volají funkce uložené v tomtéž souboru ke konci. Tyto funkce se vykonají. To znamená, že program vždy vykonává nějaké krátké úseky kódů.

Součástí projektu jsou dva soubory, ve kterých nejsou řešeny grafické prvky. Jsou to soubory *rs232.py* a *json_dic.py*.

Soubor *rs232.py*

Tento soubor obsahuje definici třídy *RS232(object)*. Ta zajišťuje kompletní práci se sériovou linkou. Jsou zde funkce pro její inicializaci, otevření, odeslání příkazů a jejich přijetí ve správném formátu. Formát příkazů, kterými se komunikuje je popsán v kapitole 8.1.2.

Základní komunikační funkcí tohoto souboru je:

```
def uartTX_RX(self, NUM_OF_TRY, RW, TYPE, ADRESS, VALUE):
```

Vstupními parametry funkce jsou opět součástí komunikačního příkazu (viz. 8.1.2). Navíc je zde parametr *NUM_OF_TRY*, ten udává počet pokusů, kolikrát má být příkaz odeslán, pokud nepříjde správná odpověď.

Funkce je dělená na dvě základní části. Pro čtení (paramter *RW* je 'R') a pro zápis (parametr *RW* je 'W'). Obě části se liší pouze částečně (rozdíl bude zdůrazněn v následujícím textu).

Po zavolání funkce se vytvoří ze vstupních parametrů řetězec (příkaz) k odeslání za pomoci funkce:

```
def makeUartMessage(self, RW, TYPE, ADRESS, VALUE):
```

V dalším kroku se vyčistí vstupní i výstupní paměti. Potom je odeslán jeden znak *ACK* (6h). Následně program čeká na odpověď mikrokontroleru. Čtení ze sériové linky je čekající funkce, to znamená, že dokud nedorazí odpověď, nebo do uplynutí nastaveného *TIMEOUT* (v tomto případě 2 sekundy), je program zablokován.

Po přijetí odpovědi, tedy stejného znaku *ACK* je jasné, že komunikace funguje, a je možné pokračovat dále. Pokud ne funkce skončí a vrací hodnotu -1.

V případě úspěchu dojde opět k pročištění vstupních i výstupních pamětí a je odeslán příkaz funkcí:

```
def write(self, data)
```

Po odeslání je jasné, že mikrokontroler odpoví. Proto je volaná funkce:

```
def uartReceiveOneMessage(self):
```

Funkce obsahuje části, které pomocí smyčky čtou sériovou linku takovým způsobem, aby chybný (nežádoucí) řetězec nebyl vyhodnocen jako příkaz. Drží se tedy oddělovacích znaků v řetězci. Po přečtení celého příkazu (začíná znakem # a končí \$) se vytvoří pole, kde jsou přijaté hodnoty uloženy a toto pole se vrací. V případě neúspěchu se vrací -1.

Po návratu je ověřeno, zda návratová hodnota je datový typ *list*. Tím se ověří, že příkaz byl přijat.

Následuje kontrola správnosti přijatých dat. Zde je zmíněný rozdíl mezi zápisem a čtením. Pro zápis se kontroluje zda přijatá data odpovídají odeslaným, včetně parametru *VALUE*, tedy například hodnoty registru, kterou chceme nastavit. Mikrokontroler tuto hodnotu nastaví, následně obsah registru přečte a pošle přečtenou hodnotu zpět. Tím je docíleno, že při zápisu nedojde k chybě a registr je naprogramován správně. U příkazu pro čtení se tento parametr nekontroluje.

Pokud kontrola správnosti dopadne dobře, odeslání příkazu je dokončeno a funkce vrací pole hodnot, kde jsou jednotlivé části komunikačního řetězce. Pokud dopadne špatně, tak se odeslání a příjem zkusí *NUM_OF_TRY* krát. Pokud komunikace neproběhne ani po těchto pokusech, funkce vrací hodnotu -1.

Soubor `json_dic.py`

Velmi důležitý soubor projektu. Třída, která je zde obsažená, zastupuje vnitřní paměť programu. Dále jsou zde definovány konstanty používané pro rozlišení dat v programu. Důležitou součástí tohoto souboru je práce se soubory. V následujícím textu je popsána každá část.

Jsou zde definovány datové typy *list*. Ty zastupují interní paměť programu, a jsou v nich vždy nejaktuálnější data. Pokud se něco posílá do prototypu měniče, pracuje se s daty právě v těchto polích. To znamená, že i každá úprava zadávacích boxů v grafickém rozhraní okamžitě ukládá nová data do těchto paměťových polí.

V programu je takových paměťových prvků celkem 6. Tuto šestici tvoří dvě trojice stejně dlouhých polí. Jedna je pro diagnostické informace, druhá pro řídicí registry. Každá trojice obsahuje stejně dlouhé pole pro uložení **typu** dat, **adresy** registru a jeho **hodnoty**. Po zapnutí programu jsou tyto pole inicializovány na výchozí délky, a jsou do nich uložena výchozí data s tím, že pole typů a pole adres se v programu nesmí měnit, pouze obsahy (hodnoty) registrů. Inicializace je provedena funkcí:

```
def firstInitializationOfLists(self):
```

Co se týká definování konstant důležitých pro správnou funkci GUI, je zde 14 takových symbolů. Všechny je možné vidět v příloze E. Tyto symboly se nejčastěji používají k rozlišení příkazů na sériové lince, nicméně mají značné využití i při

rozlišení dat v interní paměti programu. Příklad uchovávání dat ve zmíněných polích s využitím definovaných symbolů:

```
self.addDataToList(self.BOOST, '1C', '00')
```

Pokud je volaná tato funkce, je na první volné místo v poli (na konec pole) uloženo následující:

```
self.list_of_types[konec_seznamu] = self.BOOST
self.list_of_adress[konec_seznamu] = '1C'
self.list_of_values[konec_seznamu] = '00'
```

Další velká úloha této třídy je práce se soubory. Je důležité aby program umožňoval ukládání konfigurace registrů integrovaných obvodů měniče do souborů.

Pro práci s daty a soubory je použit modul *JSON*. Ten umožní poměrně jednoduché a přehledné zapisování a čtení ze souborů. Pro přehlednou práci je v souboru *json_dic.py* vytvořen datový typ *dictionary* (proměnná *python_dict*). Ten se následně už jednoduše převádí na typ *JSON* pomocí funkce:

```
def pythonDict2JSON(self):
```

Opačný převod realizuje:

```
def JSON2pythonDict(self):
```

Zápis do souboru je realizován funkcí, jejíž obsah je možné vidět v příloze E:

```
def saveJSONtofile(self, type_of_data):
```

Po zavolání zmíněné funkce se otevře dialogové okno pro výběr umístění souboru. Následně je ošetřeno aby název obsahoval alespoň nějaké znaky, zároveň je ošetřeno, aby přípona na konci názvu byla přidána pokud by obsluha zapoměla. Dále se soubor otevře. Následně je volaná funkce:

```
self.addData2dictionary()
```

Ta zajistí, že se data z paměťových polí, zmíněných na začátku této podkapitoly, uloží do datového typu *dictionary*. Ten se následně převede na datový typ *JSON*. Do souboru se uloží na první řádek zmíněné definované symboly, aby bylo jasné o jaká data se jedná. Na nový řádek se poté uloží slovník v *JSON* formátu. Na konci se soubor zavře.

Načítání ze souboru funguje podobně jak uložení. Je realizováno funkcí:

```
def loadJSONfromfile(self, type_of_data):
```

Tato funkce načte ze souboru všechny řádky. Následně zkontroluje zda je načítán správný soubor. Pokud je vše v pořádku uloží obsah souboru do slovníku ve formátu *JSON*. Ten se následně převede na datový typ pythonu *dictionary* a je zavolána funkce:

```
def readDatafromDictionary(self, type_of_data):
```

Tato funkce prochází načtená data, a podle typu načítaných dat přepíše interní paměť programu, tedy na počátku zmíněná pole dat. Pro představu je v příloze E zobrazen kousek této funkce.

9 MĚŘENÍ REÁLNÉHO PROTOTYPU

9.1 Úvod do měření

Poté co byl měnič vyroben, nakonfigurován a otestován, bylo třeba změřit jeho parametry. Jednalo se o základní elektrické veličiny obvodu. Mezi ně se počítá frekvence spínání měničů, zvlnění napětí, střída spínání a podobně.

Návrh DPS byl rovněž zaměřen na návrh dostatečného chlazení vzhledem k velkým proudům v obvodu. Proto byl prototyp snímán i pod termokamerou.

Zátěž byla poněkud odlišná oproti původně navrhovaným parametrům. To bylo způsobeno dvěma důvody. Při výrobě došlo k záměně původně plánovaných LED diod, které měly mít úbytek napětí 4,1 V při proudu 140 mA na diody s úbytkem 2,15 V při protékajícím proudu o velikosti 145 mA. To znamená, že 4 ze 6 výstupních kanálů pracovaly na skoro poloviční výkon. Nicméně díky volným svorkám na dvou kanálech bylo možné připojit zátěž, která se více blíží výpočtům. Jednalo se o nastavitelné výkonové odpory, které byly seřizeny na odpor 330 Ω . To umožnilo dosáhnout napětí 46 V při proudu 140 mA.

9.1.1 Řešení počátečních problémů

Rušení na SPI

Při testování SPI komunikace se ukázalo, že na vodičích sériové linky jsou patrné poměrně velké zákmity, které v krajních případech způsobovaly špatné vyhodnocení odesílaných dat. Nabízelo se několik možností jak tyto zákmity vyřešit.

V první fázi bylo zkoumáno, zda napájecí napětí logických obvodů, tedy 5 V za LDO regulátorem není z nějakého důvodu zarušeno. Při jeho změření se ukázalo, že se stejné zákmity objevují i na něm. Proto byla v této fázi snaha o tlumení vstupu LDO regulátoru pro případ, že by se díky nežádoucím zámkům v napájecím napětí rozkmitával. Nicméně toto tlumení nemělo vliv.

V další fázi se LDO regulátor odstavil a napájení logických obvodů bylo realizováno externím zdrojem. Nicméně i po tomto kroku kmity neustály. Tím byla eliminována možnost, že se rušení šíří po napájení.

Byly proto změřeny úrovně zámků na všech vodičích SPI sběrnice i na napájení logických obvodů. To prokázalo, že nejsilnější zákmity jsou na dlouhých vodičích, které byly taženy nejbližší kolem zvyšujícího měniče. Zároveň byla změřena i frekvence zámků. Ta se pohybovala v okolí 3 MHz. Zákmity se opakovaly s frekvencí, která byla totožná s frekvencí spínání zvyšujícího měniče. Z toho důvodu byl nejdříve zvětšen blokovací kondenzátor u pinu integrovaného obvodu zvyšujícího měniče, na

který je přivedeno napájení logických obvodů (tedy 5 V). To však na zákmity nemělo nejmenší vliv.

Došel jsem tedy k závěru, že se rušení indukuje na zmíněné dlouhé cesty vlivem kapacity mezi ovlivněnými vodiči a výkonovými cestami zvyšujícího měniče. Řešení tohoto stavu spočívalo v novém návrhu desky plošného spoje (které ale nebylo možné) nebo v omezení rychlosti spínání tranzistorů měniče. Při návrhu DPS byl před každý tranzistor zařazen rezistor o hodnotě 0Ω , právě pro umožnění tohoto zpomalení.

Mezi hradla tranzistorů a čipy měniče byly tedy vloženy rezistory o hodnotě 10Ω . Tím se zákmity zmenšily na minimum a komunikace SPI začala být spolehlivá. Na druhé straně, tyto rezistory budou mít ve výsledku vliv na více parametrů. Zejména sníží nepatrně účinnost měniče vlivem pomalejšího spínání.

Příliš velká zátěž

Při testování měniče s plnou zátěží, tedy s výkonovými rezistory připojenými ke kanálům č. 2 obou snižujících měničů začalo docházet k chybám. Zatím co snižující měnič č. 1 (v dolní části DPS) fungoval správně, měnič č. 2 odpojoval druhý kanál (tedy výkonový rezistor) již po chvilce činnosti. Pokud byl rezistor odpojen, měnič fungoval bez problému. Pokud byl připojen pouze rezistor a řetězce LED diod byly odpojeny, měnič opět fungoval správně.

Chyba vznikala v ovládání hradel spínacích tranzistorů při velké zátěži. Konkrétně se jednalo o příliš nízké napětí mezi piny BSx a LXx. To je signalizováno nastaveným bitem v registru na adrese 34h. Celkově se jedná o funkci, která zajistí, aby nemohl být tranzistor nechtěně sepnut vlivem nízkého napětí na elektrodě *source* tranzistoru [9]. Bylo provedeno mnoho měření, jednalo se o stabilitu napájecího napětí, velikost napětí na elektrodě *gate* tranzistoru a podobně, nicméně nic neukazovalo na problém.

Po dalším testování se začal objevovat další problém s výkyvy v měření napětí i ve zvyšujícím měniči. Tento fakt je popsán v následujících dvou podkapitolách spolu s pravděpodobným vysvětlením problému.

Vzhledem k faktu, že se problém nepodařilo vyřešit, bylo rozhodnuto, že měnič bude proměřen s funkční konfigurací výstupních kanálů. Tedy s aktivním kanálem 1 a 3 snižujícího měniče číslo 2 a se všemi aktivními kanály měniče číslo 1. Vzhledem k odděleným napájením jednotlivých snižujících měničů, tedy dvěma nezávislým výstupním napětím zvyšujícího měniče bylo možné prozkoumat chování alespoň jedné funkční větve DC-DC měniče.

Výkyvy v měření napětí

Při testování možného rozsahu vstupního napájecího napětí prototypu měniče se vyskytl problém při krajních mezích napájecího napětí. Jednalo se o úroveň napájení nižší než 6,5 V a vyšší než 16,5 V.

V obou případech došlo po náhodně dlouhé době k odpojení výstupních kanálů zvyšujícího měniče. Chyba, kterou signalizovaly diagnostické registry měniče, byla buď podpětí, či přepětí na napájecím napětí. Hodnoty napětí, při kterých se objeví tato chyba, jsou nastavitelné. V případě přepětí bylo možné tuto hranici nastavit na vyšší hodnotu, tedy přibližně 18,4 V (nastavení registru na adrese 1Ah). Vzhledem k dimenzování obvodu nepředstavovala tato změna žádné riziko. Po tomto nastavení byla umožněná funkčnost při napájecím napětí 17 V.

V případě spodní hranice však byla situace jiná. Zde dochází ke kolizí s nastaveným spínacím napětím hradel externích tranzistorů měniče. To je tvořeno LDO (Low-Dropout regulator) regulátorem. Už z principu činnosti tohoto regulátoru je nepřípustné, aby se na jeho vstupu objevilo příliš nízké napětí. Proto musela být úroveň detekce podpětí nastavena na vyšší hodnotu než je spínací napětí. Napětí spínání bylo nastaveno na hodnotu 5,52 V (registr na adrese 15h) a podpěťová úroveň na hodnotu 5,77 V (registr na adrese 1Bh). Nižší úroveň už by nebyla zřejmě rozumná jelikož podle datového listu tranzistoru *SiR606BDP*, použitého pro spínání, se odpor kanálu v otevřeném stavu velmi rychle zvyšuje po poklesu napětí na hradle tranzistoru k 5 voltům.

V obou případech se evidentně vyskytovala chyba špatného měření napětí, jelikož i při napájecím napětí 6 V by úbytek napětí na vstupních obvodech nedosáhl takové hodnoty, aby došlo k přerušování činnosti.

S nesprávným měřením se je možné setkat i v případě, že si obsluha v GUI měří napětí na napájecím napětí prototypu. Při opakovaném měření je toto napětí nestálé, a občas se odchyluje od reálné hodnoty až o 0,5 V.

Při objasňování tohoto problému byly opět proměřeny průběhy různých napětí s čipem souvisejících, nicméně se neukázalo, že by se někde objevovaly jakékoliv výkyvy, které by to způsobovaly. Navíc je v návrhu rozmístěno mnoho blokových kondenzátorů a filtračních prvků, takže by případné výkyvy na vodičích měly být těmito prvky odstraněny.

Pravděpodobná příčina

Objasnění zmíněných problémů je velmi problematické a časově náročné. Nicméně byl určen pravděpodobný závěr, že tyto výkyvy jsou spojeny s plovoucí zemí. Tím je myšleno, že při spínání výkonových prvků dochází k nepatrnému lokálnímu navýšení napěťové úrovně zemního vodiče.

K tomuto závěru vedlo několik důvodů. Při zvyšující se zátěži bylo interní měření vstupního napětí zvyšujícího měniče méně stále. Dále se problém odpojování výstupního kanálu objevoval pouze ve snižujícím měniči č. 2. Vzhledem k faktu, že zapojení měničů je totožné a rozložení součástek vůči měniči je velmi podobné, vypadá to, že chyba je v desce plošného spoje, konkrétně v zemnicí ploše.

Objasnění takového tvrzení je však poměrně obtížné a z časových důvodů není tento jev zkoumán do podrobnosti. Součástí řešení tohoto problému by mohl být i návrh dalšího prototypu měniče. V tomto případě není bohužel možné vyřešit zmíněné problémy další filtrací na vodičích.

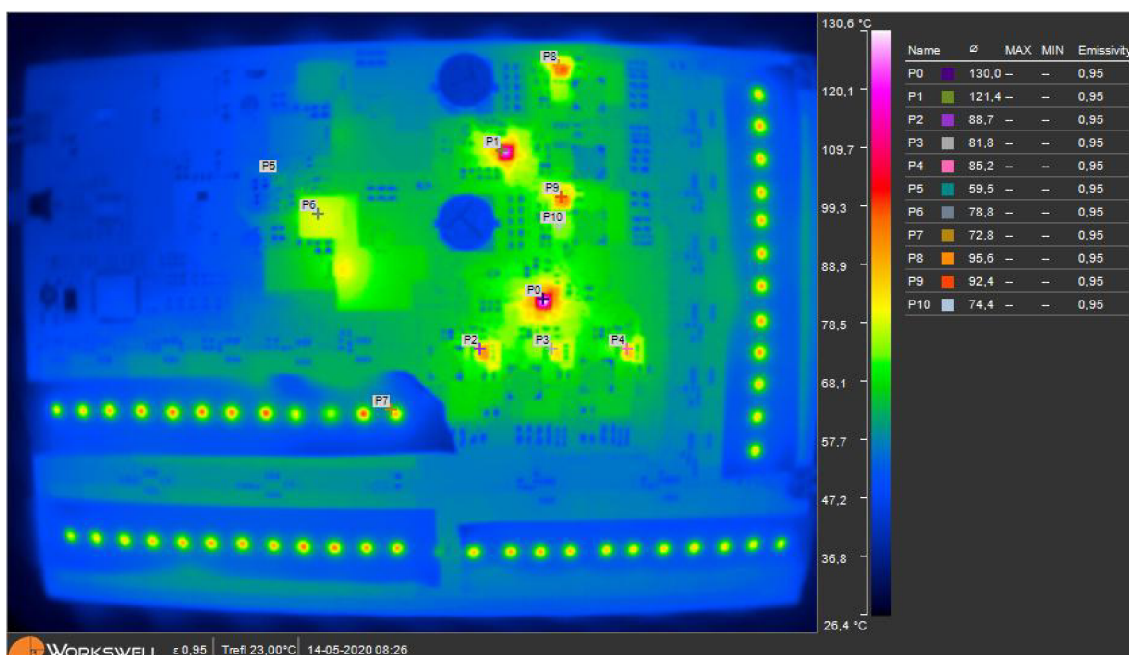
9.2 Konfigurace měničů

Z teorie o čípech použitých v práci je jasné, že měnič umožňuje široké možnosti konfigurace. Aby bylo měření směrodatné, je v této kapitole popsáno, jak jsou jednotlivé registry nakonfigurovány. Měření probíhalo v normálním pracovním režimu měniče. Nastavení energeticky nezávislé paměti je zde popsáno taky, nicméně v LHM módu měření neprobíhalo. LHM mód je specifikován tím, že po jeho aktivaci se omezí proudy a napětí v obvodu a odpojí se některé výstupní kanály. Konfigurace LHM módu není v zadání ničím specifikována a jeho nastavení je takové, aby bylo obsluze zřejmé, že se v tomto módu obvod nachází.

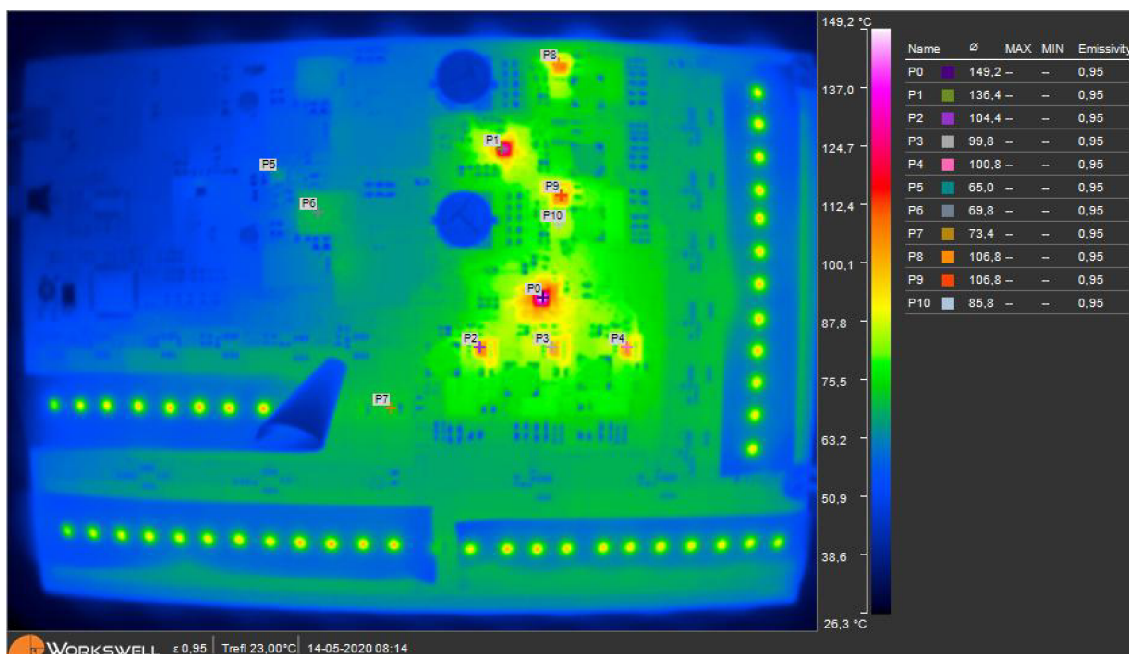
V tabulkách s konfigurací je pouze naznačen význam registrů. Pro podrobnější informace je nutné nastudovat datový list (viz elektronické zdroje: [8] a [9]).

- Tabulka s konfigurací zvyšujícího měniče, která byla použita při měření se nachází v příloze F, jedná se o tabulky F.1 a F.2
- Tabulka s konfigurací energeticky nezávislé paměti pro LHM mód se nachází v příloze F, jedná se o tabulku F.3. V této příloze je na obrázku F.1 naznačeno rozložení této paměti.
- Tabulka s konfigurací snižujících měničů se nachází v příloze F, jedná se o tabulku F.4. Je nutné říci, že se konfigurace mohla lišit při různých nastaveních výstupu v průběhu měření.
- Konfigurace energeticky nezávislé paměti je v případě obou snižujících měničů podobná. Na obrázku F.2 v příloze F je opět možné vidět rozložení NVM paměti snižujícího měniče. Tabulka F.5 ve stejné příloze pak představuje nastavení snižujícího měniče číslo 1. V LHM módu je měnič nastaven tak, že výstupní proud kanálem 1 a 3 je 70,16 mA, kanál 2 je odpojen. Dále kanál 1 reaguje na vstupní PWM regulaci a kanál 3 je spuštěn bez ohledu na vstupní PWM signál. Rozdíl ve snižujícím měniči č. 2 je pouze v tom, že ani kanál 1 nereaguje na vstupní PWM signál a je trvale spuštěn.

9.3 Teplotní rozložení



Obr. 9.1: Rozložení teploty na DPS při napájecím napětí 7 V



Obr. 9.2: Rozložení teploty na DPS při napájecím napětí 13,5 V

Při návrhu bylo rovněž nutné zaměřit se na chladicí plochy na desce plošného spoje z důvodu velkých proudů vyskytujících se v obvodu.

Snímání termokamerou bylo realizováno pro minimální napájecí napětí 6,5 V a pro nominální, tedy 13,5 V. Snímky z termokamery je možné vidět na obrázku 9.2 a 9.1.

Při měření bylo na desce vytvořeno několik měřících bodů. Měření bylo prováděno při teplotě okolí 23 °C.

Výsledky měření byly ve značném rozporu s vypočtenými hodnotami v podkapitolách sekce 4.3. Nejvíce se vypočteným hodnotám blížily součástky zvyšujícího měniče. Pro přehlednost jsou naměřené hodnoty shrnuty v tabulce 9.1.

Tab. 9.1: Porovnání naměřených teplot pro různá napájecí napětí

| | | Teplota ve °C pro napájecí napětí | |
|------------|---|-----------------------------------|--------|
| Měřicí bod | Popis | 7 V | 13,5 V |
| P0 | Čip buck měniče č. 1 | 130,0 | 149,2 |
| P1 | Čip buck měniče č. 2 | 121,4 | 136,4 |
| P2 | Tranzistor kanálu 3, buck č. 1 | 88,7 | 104,4 |
| P3 | Tranzistor kanálu 2, buck č. 1 | 81,8 | 99,8 |
| P4 | Tranzistor kanálu 1, buck č. 1 | 85,2 | 100,8 |
| P5 | Čip boost měniče | 59,5 | 65,0 |
| P6 | Cívka boost měniče napájející buck č. 1 | 78,8 | 69,8 |
| P7 | LED dioda výstupu při plném zatížení | 72,8 | 73,4 |
| P8 | Tranzistor kanálu 1, buck č. 2 | 95,6 | 106,8 |
| P9 | Tranzistor kanálu 3, buck č. 2 | 92,4 | 106,8 |
| P10 | Cívka kanálu 3, buck č. 2 | 74,4 | 85,8 |

Podle výpočtu by měl tranzistor zvyšujícího měniče zvýšit svoji teplotu při plné zátěži o 38 °C. Dioda zvyšujícího měniče by měla zvýšit svojí teplotu o maximálně 30 °C. Tranzistory snižujícího měniče pak o pouhých 7 °C. Realita je však velmi odlišná.

Co se týká vstupních obvodů, tedy přepětové ochrany a vstupního filtru, byla změna teploty minimální a pro případný další návrh by bez problému mohla být chladicí plocha této části zmenšena. Zároveň se ukázalo, že výkonové cesty na DPS se při velkých proudech neohřívají a jejich dimenzování je dostatečné.

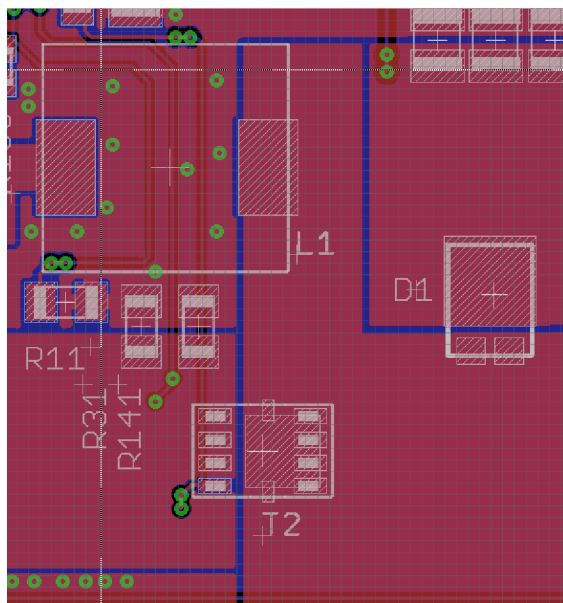
Chlazení LED diod je z mého hlediska ideální. Jejich teplota je kolem 70 °C a mohou pracovat až do teploty čipu kolem 150 °C. Navíc měřicí bod *P7* je ovlivněn velmi vysokou teplotou snižujícího měniče, takže pokud by se snižující měnič podařilo více ochladit, byla by jejich teplota ještě o něco nižší.

9.3.1 Možné příčiny odchylky naměřených hodnot

Zvyšující měnič

Celkově se realitě blížily výpočty zvyšujícího měniče. Při plném zatížení se tranzistor stejně jako cívka oteplil o 55,8 °C. Při vyšším napájecím napětí se ohřál o 46,8 stupňů. V případě diody bylo zvýšení teploty kolem 40 °C.

Ačkoliv jsou tyto hodnoty vyšší než očekávané, jsou pořád dostatečně nízké, aby obvod mohl pracovat i v teplotách okolo 80 °C. I když je chlazení dostatečné, na obrázcích z termokamery je možné si všimnout, že největší tepelné vyzařování v případě tranzistoru je na elektrodě *drain* (v okolí měřícího bodu *P6*). Bylo by tak možné zmenšit chladící plochu elektrody *source* a tuto plochu přidat na "druhou" stranu tranzistoru. Dále by bylo vhodné rozšířit zúžení chladící plochy v místě kde se připojuje cívka k chladící ploše *drainu* tranzistoru (viz obrázek 9.3).



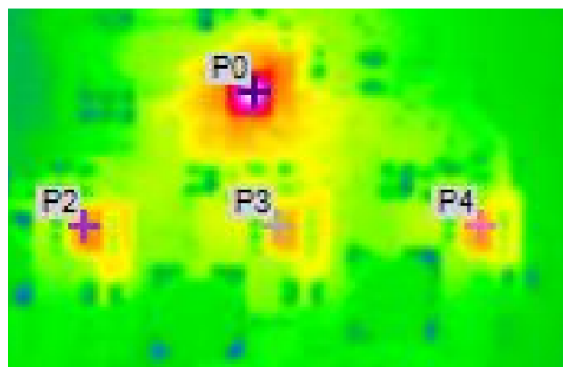
Obr. 9.3: Detail zúžení připojného místa cívky k elektrodě drain tranzistoru zvyšujícího měniče

Dalším důvodem odchylky od vypočtené hodnoty bude bezpochyby také zpomalení tranzistorů rezistory na elektrodách *gate*. Tím se zvednou dynamické ztráty tranzistoru.

Poslední možnou příčinou může být poměrně velké ohřívání samotného čipu zvyšujícího měniče. Ten ve svém blízkém okolí zvýší teplotu plošného spoje a tím ovlivní i oteplení ostatních součástek. Čip má při zatížení teplotu okolo 60 °C. To je poměrně velká hodnota vzhledem k faktu, že výkonové spínací prvky jsou externí. Zajímavé je, že s rostoucím napájecím napětím zdatelně roste i teplota čipu. Je tedy vidět, že ovládaný proud nemá na změnu teploty nijak závažný vliv. Spíše to ukazuje na možnost, že oteplení je spojeno s vnitřní regulací napětí pro ovládání elektrody *gate* tranzistoru. Řešení, v případě návrhu nové desky, by spočívalo v promyšlenějším návrhu chladících ploch čipu. Faktem je, že se jedná o pouzdro HVQFN32, takže největším problémem je odvod tepla ze spodní (chladící) strany součástky pryč. V tomto případě není teplota až tak vysoká, takže by pravděpodobně postačilo přidat prokovy pod chladící plochu čipu. Možností by rovněž bylo udělat spoje s okolní zemnicí plochou o něco širší vzhledem k faktu, že velká část pinů čipu je na stejném potenciálu jak jeho chladící plocha.

Snižující měniče

Zde je situace zcela odlišná. Na obrázku 9.4 je možné vidět detail záběru termokamery. Na tomto výřezu se nachází čip, tranzistory a cívky snižujícího měniče číslo 1.



Obr. 9.4: Detail snímku termokamery snižujícího měniče č. 1 pro napájecí napětí 13,5 V

Situace je velmi znepokojivá. Teploty čipu dosahují 149 °C, u tranzistoru je to 100 °C a u cívek se teplota pohybuje kolem 75 °C. Otázka zní, kde vzniká chyba.

Při analýze chyby je dobré podívat se na obrázek 9.2 do míst snižujícího měniče č. 2, kde je cívka a tranzistor kanálu 2 odpojen. Je možné vidět, že teplota DPS v těchto místech dosahuje přibližně 75 °C. Pokud by k této teplotě bylo přihlédnuto jako k výchozí teplotě desky, byl by ohřev aktivních tranzistorů 25-30 °C a cívky by

se neohřívaly téměř vůbec. Přesto je zmíněných 25-30 °C celkem dost s přihlédnutím k velmi malým proudům, které tranzistory protékají. Na druhé straně je spínací kmitočet poměrně vysoký, takže dynamické ztráty tranzistoru jsou z tohoto důvodu vyšší.

Při prozkoumání termosnímku je vidět, že opět většina tepla se vyzařuje do elektrody *drain*, avšak při návrhu byla chybně udělaná velká chladicí plocha při elektrodě *source*. Z toho plyne, že pro případný další návrh by bylo nutné tyto chladicí plochy optimalizovat obdobně jako tomu je u zvyšujícího měniče.

To však neřeší enormní teploty v oblasti samotného čipu snižujícího měniče. Dle mého názoru právě velký ohřev čipů ASL3417SHN vede k ohřevu jeho okolí, a tím se velmi zvedne i teplota ostatních komponent. Z toho plyne, že zásadní úlohou případného nového návrhu desky plošného by bylo řešení teplot integrovaných obvodů snižujícího měniče, se kterými se při prvním návrhu příliš nepočítalo.

Je velmi obtížné určit důvod, proč dochází k tak velkým teplotním výkyvům. Opět se zde vyskytuje jev stejně jako v případě zvyšujícího měniče. Při klesajícím vstupním napětí teplota čipů prudce klesá. Naopak je zvláštní, že s rostoucí zátěží měniče spíše dochází k jeho ochlazování. Jedná se však pouze například o 10 °C při poloviční zátěži. To však bude pravděpodobně spojeno se změnou spínací frekvence. Obecně se prozatím nepodařilo z časových důvodů objasnit, kde tak velký ohřev vzniká. Je pravděpodobnější, že chyba je někde v návrhu DPS, nikoliv v programovém vybavení, jelikož možnosti nastavení snižujících měničů není tolik. V případě nového návrhu by bylo dle mého názoru třeba realizovat následující kroky:

- Projít ještě jednou všechna zapojení a zkontrolovat, zda není někde chyba například ve špatně vytvořeném pouzdru. Tento krok by bylo nutné provést i pro vyhodnocení zda extrémní ohřev nesouvisí s nízkou účinností snižujícího měniče.
- Kontaktovat výrobce čipů a určit možné příčiny takového ohřevu.
- Při návrhu není možné odvést teplo mimo chladicí plochu (pod čipem) v jedné vrstvě mědi, jelikož pouzdro neobsahuje dostatek pinů, přes které by bylo možné ze spodní chladicí plochy čipu projít. Navíc je využita většina pinů pouzdra HVQFN32, takže blízko čipu není žádná velká plocha mědi, která by mohla chladit.
- Řešením by bylo přidání prokovů do chladicí plochy a odvedení tepla na jinou vrstvu DPS. To by však problém asi nevyřešilo úplně, jelikož již v aktuálním provedení jsou pod čipem 2 prokovy, a víc jak další dva by se tam pravděpodobně nevešly.
- Dalším účinnějším způsobem by bylo použití DPS s izolovaným kovovým substrátem, nicméně toto řešení by bylo velmi drahé.

Výhodou je, že se na prototypu nachází 2 identicky zapojené čipy ASL3417SHN.

To značně redukuje pravděpodobnost chyby v osazení plošného spoje při jeho výrobě.

9.4 Měření elektrických veličin

V této kapitole jsou vyobrazeny naměřené základní parametry a průběhy prototypu měniče. Kapitola je rozdělená na část pro napětí a proudy v obvodu, samostatná sekce pro časové průběhy zvyšujícího měniče, snižujícího měniče a v poslední části je diskutována efektivita zapojení.

9.4.1 Napětí a proudy v obvodu

Výsledky měření napětí a proudu v různých místech obvodu jsou rozděleny do několika tabulek. Společné pro všechny tabulky jsou první dva sloupce, které prezentují napájecí napětí a proud. Tabulky jsou barevně rozděleny na zelenou a modrou část. Zelená část prezentuje naměřené výsledky při plné zátěži (maximální proud protékající zátěží), modrá pak prezentuje případ kdy proud zátěže je poloviční (přibližně 78 mA).

Tab. 9.2: Naměřené napětí na zvyšujícím měniči a vstupních obvodech

| Napájecí napětí | | Zvyšující měnič | | Vstupní obvody | |
|-----------------|--------------|-----------------|----------------|----------------|------------|
| V_{in} [V] | I_{in} [A] | V_{out1} [V] | V_{out2} [V] | V_{out} [V] | Úbytek [V] |
| 6,5 | 5,66 | 68,3 | 71 | 6,376 | 0,124 |
| 13,5 | 2,60 | 73,7 | 73,8 | 13,44 | 0,060 |
| 17,0 | 2,08 | 73,8 | 73,6 | 16,96 | 0,040 |
| 6,5 | 3,10 | 71,2 | 72,5 | 6,43 | 0,070 |
| 13,5 | 1,50 | 73,8 | 73,6 | 13,46 | 0,040 |
| 17,0 | 1,19 | 73,8 | 73,6 | 16,96 | 0,040 |

Tabulka 9.2 ukazuje naměřené hodnoty výstupního napětí zvyšujícího měniče na obou výstupních kanálech. Dále se zde nachází naměřené hodnoty napětí na výstupu vstupních filtračních obvodů a zároveň úbytek napětí na těchto obvodech. Pro srovnání s výpočtem je třeba nahlédnout do kapitoly 4, kde se nachází podrobné výpočty. Podle této kapitoly by měl být maximální vstupní proud 9,2 A při plném zatížení a nejnižším napájecím napětí, naopak pro nejvyšší napájecí napětí by tento proud měl nabývat hodnot maximálně 3,2 A. Co se týká úbytku napětí na vstupních obvodech, měl by se pohybovat v intervalu $\langle 40, 160 \rangle$ mV.

Při porovnání je třeba uvážit, že nejnižší napájecí napětí bylo 6,5 V, nikoliv plánovaných 6 V, což by proudy navýšilo. Naměřené proudy se pohybují v intervalu $\langle 2,08, 5,66 \rangle$ A. Jsou to nižší hodnoty než vypočtené. Nicméně je zde fakt, že měnič nepracuje na navrženém maximálním výkonu, ale kvůli jiným osazeným LED diodám a jednomu odpojenému výstupnímu kanálu pracuje na skoro polovině výkonu. Z toho plyne že by se vstupní proud pravděpodobně zvýšil k hranici 11 A. Nicméně nárůst by nejspíše nebyl lineární hlavně z důvodu, že podle měření, zvedá zvyšující se zátěž poměrně značně efektivitu zapojení viz tabulka 9.5. Co se týká úbytku na vstupních obvodech, ten by předpokládanou hranici překročil téměř jistě. Nicméně by pravděpodobně nepřesáhl 0,5 V. S tímto úbytkem by měnič stále mohl pracovat. Záleželo by tedy pouze na ohřevu součástek, zda měnič podmínky zvládne, či nikoliv. Vzhledem k faktu, že při dosažených proudech v průběhu měření k přílišnému ohřevu nedocházelo, měnič by pravděpodobně fungoval správně.

Co se týká výstupního napětí měniče, napětí kolísalo nejvíce při nejnižším napájecím napětí. Zde byl pokles napětí přibližně 6 V, nicméně i přesto bylo napětí stabilní bez zákmitů. Při ostatních napájecích napětích bylo výstupní napětí konstantní. Úrovně výstupů by šlo s největší pravděpodobností ještě ladit změnou nastavení parametrů zpětné vazby v měniči. Nicméně by se jednalo o dlouhý proces, ke kterému z časových důvodů nedošlo. Obecně lze říci, že zvyšující měnič plní svou úlohu v zapojení dobře, zejména z pohledu změny výstupního napětí při změně vstupního napětí. Jeho efektivita je rovněž vysoká, o tom více v další kapitole.

Tab. 9.3: Naměřené napětí a proudy na snižujícím měniči č. 1

| Napájecí napětí | | Kanál 1 | | Kanál 2 | | Kanál 3 | |
|-----------------|--------------|---------------|----------------|---------------|----------------|---------------|----------------|
| V_{in} [V] | I_{in} [A] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] |
| 6,5 | 5,66 | 25,80 | 147,00 | 45,82 | 139,39 | 25,74 | 149,00 |
| 13,5 | 2,60 | 25,53 | 148,00 | 45,95 | 139,39 | 25,55 | 150,00 |
| 17,0 | 2,08 | 25,68 | 148,00 | 46,07 | 139,39 | 25,66 | 150,00 |
| 6,5 | 3,10 | 24,03 | 79,00 | 25,86 | 78,79 | 24,05 | 79,00 |
| 13,5 | 1,50 | 24,02 | 79,00 | 25,99 | 78,79 | 24,05 | 80,00 |
| 17,0 | 1,19 | 23,98 | 79,00 | 25,94 | 78,79 | 24,01 | 80,00 |

U snižujících měničů byla zkoumána hlavně stabilita výstupu. Právě díky zvyšujícímu měniči na vstupu by měl měnič pracovat s velmi malými výkyvy vstupního napětí. Tabulka 9.3 a 9.4 ukazuje měřené hodnoty proudu a napětí na výstupních kanálech. Pro měření zelené části tabulky byl nastaven výstupní proud na hodnotu

Tab. 9.4: Naměřené napětí a proudy na snižujícím měniči č. 2

| Napájecí napětí | | Kanál 1 | | Kanál 3 | |
|-----------------|--------------|---------------|----------------|---------------|----------------|
| V_{in} [V] | I_{in} [A] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] |
| 6,5 | 5,66 | 25,62 | 146,00 | 25,63 | 149,00 |
| 13,5 | 2,60 | 25,42 | 146,00 | 25,46 | 149,00 |
| 17,0 | 2,08 | 25,52 | 146,00 | 25,54 | 149,00 |
| 6,5 | 3,10 | 23,92 | 77,00 | 23,99 | 78,00 |
| 13,5 | 1,50 | 23,91 | 77,00 | 23,97 | 79,00 |
| 17,0 | 1,19 | 23,90 | 77,00 | 23,96 | 79,00 |

145 mA. Pro modrou část tabulky to bylo 78 mA.

Je vidět, že i pro maximální zátěž na kanálu 2 měniče č. 1 je odchylka proudu minimální (5,61 mA). Podle datového listu výrobce je odchylka měření napětí na snímacím rezistoru $\pm 4\%$. Snímací rezistor má hodnotu $0,825\ \Omega$. To znamená, že pro proud 145 mA může být odchylka napětí $\pm 4,79\text{ mV}$, a to je po přepočtu $\pm 5,8\text{ mA}$. Takže závěr je, že přesnost řízení proudu je v celém rozsahu zátěží velmi dobrá. Měnič velmi dobře reguloval i v případě změny zátěže (řetězců LED diod). Přesný proud byl dosažen i v případě připojení pouze jedné LED diody jako zátěže. Problém snižujícího měniče je ale jeho nízká účinnost, která bude diskutována dále a velmi vysoké teploty dosahované v obvodu.

Co se týká změny výstupních veličin pomocí PWM regulace, je tato možnost plně funkční a je možné jemně regulovat výstupní proud. Výhodou PWM regulace je, že lze dosáhnout i o něco nižších proudů než je nastavená velikost hystereze měničů (to v případě regulace čipem nelze). Co se týká celého měření, tak omezování zátěže bylo realizováno pouze nastavením registrů čipů.

Poslední tabulka v této sekci, tedy tabulka 9.5, prezentuje vypočtené výkony v obvodu. Základ pro výpočet tvořily tabulky 9.3 a 9.4. V tabulce je vypočítán příkon prototypu, součet výkonů na jednotlivých kanálech snižujícího měniče 1 a 2, celkový výstupní výkon, a účinnost zapojení. Účinnosti je věnována jedna z následujících podkapitol.

9.4.2 Zvyšující měnič

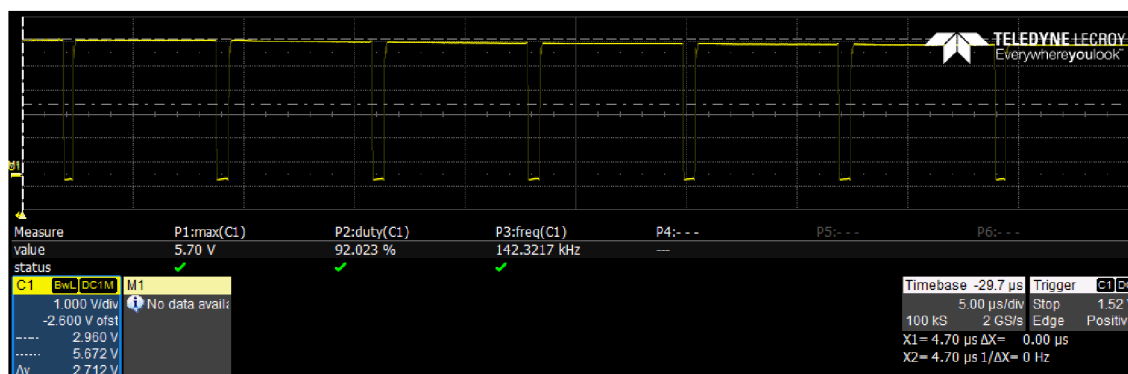
V této podkapitole jsou zobrazeny naměřené časové průběhy napětí, které souvisí se zvyšujícím měničem. Parametry jsou porovnávány s vypočtenými hodnotami v kapitole 4.

Tab. 9.5: Tabulka vypočtených výkonů podle tabulky 9.3 a 9.4

| Napájecí napětí | | | | | | |
|-----------------|--------------|--------------|-----------------|-----------------|---------------|--------------|
| V_{in} [V] | I_{in} [A] | P_{in} [W] | P_{buck1} [W] | P_{buck2} [W] | P_{out} [W] | Účinnost [%] |
| 6,5 | 5,66 | 36,79 | 14,01 | 7,56 | 21,57 | 58,6 |
| 13,5 | 2,60 | 35,10 | 14,02 | 7,50 | 21,52 | 61,3 |
| 17,0 | 2,08 | 35,39 | 14,07 | 7,53 | 21,60 | 61,0 |
| 6,5 | 3,10 | 20,17 | 5,84 | 3,71 | 9,55 | 47,4 |
| 13,5 | 1,50 | 20,25 | 5,87 | 3,73 | 9,60 | 47,4 |
| 17,0 | 1,19 | 20,23 | 5,86 | 3,73 | 9,59 | 47,4 |

Průběhy spínání externího tranzistoru

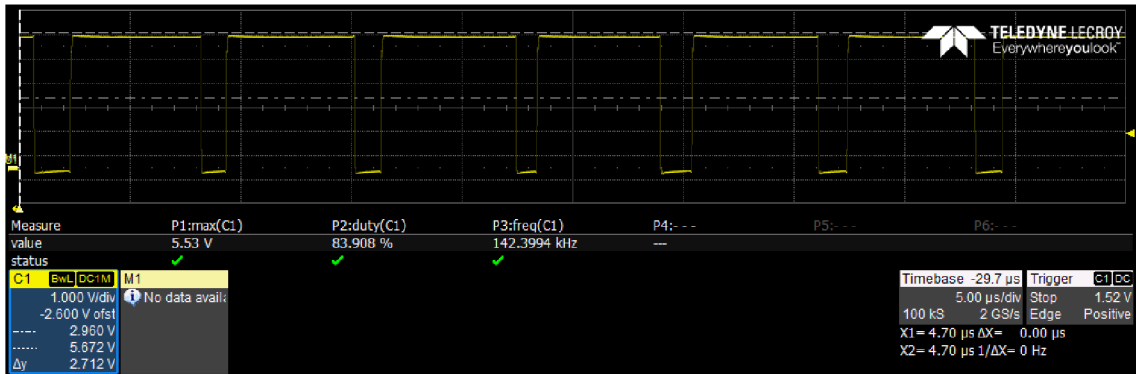
V této části je vyobrazen tvar spínacího napětí tranzistoru zvyšujícího měniče. Snímán byl výstup napájecí snižující měnič číslo 1, jelikož se zatížení tohoto kanálu více blížilo předpokládaným hodnotám. Měření probíhalo se všemi připojenými výstupními kanály. Viz zelená část tabulky 9.3.



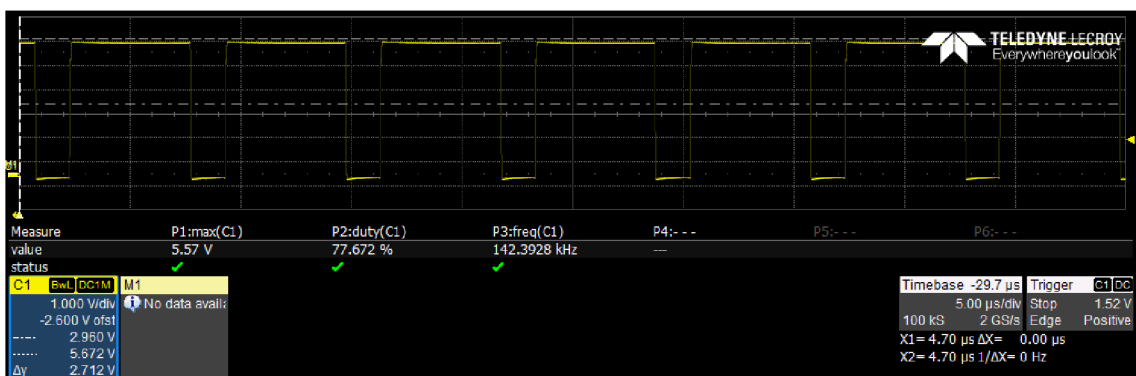
Obr. 9.5: Průběh napětí na *gate* tranzistoru s maximální zátěží, $U_{IN}=6,5$ V

Na obrázku 9.5, 9.6 a 9.7 jsou průběhy pro různá napájecí napětí. Podle teoretického výpočtu by měla být frekvence spínání 140,625 kHz, střída by se měla pohybovat v rozsahu od 77,6 % pro nejvyšší napájecí napětí, po 92,46 % po napětí nejnižší.

Frekvence spínání je fixní a je nastavitelná interními registry. Její reálná hodnota je tedy v očekávaných mezích. Naměřená hodnota spínací frekvence je přibližně 142,3 kHz. Jak je uvedeno při výpočtech v kapitole 4, výchozí oscilátor měniče má přesnost ± 5 %. To je důvodem odchylky od výpočtu.



Obr. 9.6: Průběh napětí na *gate* tranzistoru s maximální zátěží, $U_{IN}=13,5$ V



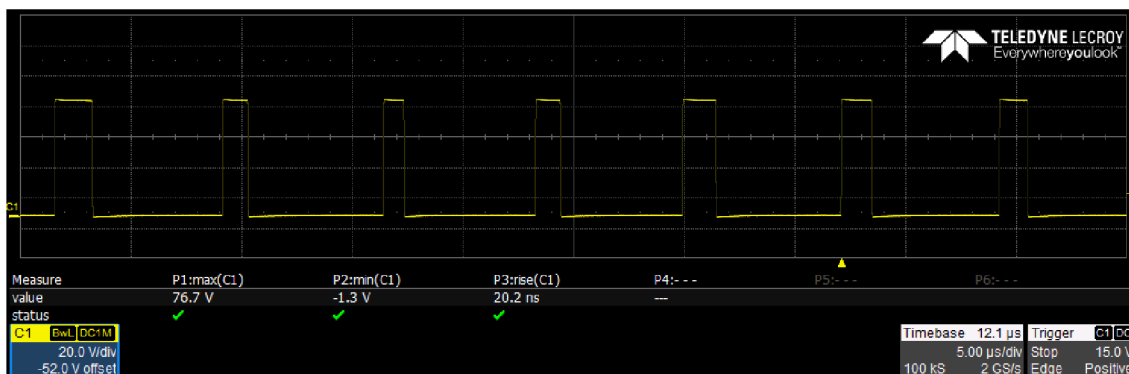
Obr. 9.7: Průběh napětí na *gate* tranzistoru s maximální zátěží, $U_{IN}=17$ V

Střída signálu je nastavena čipem automaticky. Naměřená hodnota střídý pro nejnižší vstupní napětí je 92,023 % a pro nejvyšší napětí se rovná 77,67 %. Vzhledem k faktu, že střída je závislá na napěťových poměrech v obvodu, tak zvýšení zátěže na původně předpokládanou hodnotu by ji příliš neovlivnilo. Tento výsledek je tedy velmi dobrý. Pro horní mez napájecího napětí je střída téměř stejná jak u výpočtu. Pro nejnižší mez napájení je střída o trochu menší než vypočtená. Z toho plyne, že další snížení vstupního napětí by vedlo k dosažení vypočtené hodnoty.

Napětí spínacího signálu bylo nastaveno na 5,52 V. Naměřené hodnoty odpovídaly tomuto nastavení, pouze u nejnižší napájecí úrovně bylo napětí o 0,2 V vyšší. To je ještě v mezích přesnosti udávaných výrobcem v datovém listu součástky (± 5 %, tedy 0,276 V).

Obecně je řídicí napětí tranzistorů velmi hladké, bez rušení či zákmitů s pěkným obdélníkovým průběhem. Dle mého názoru je tedy v tomto směru návrh DPS v pořádku.

Tvar signálu za cívkou měniče



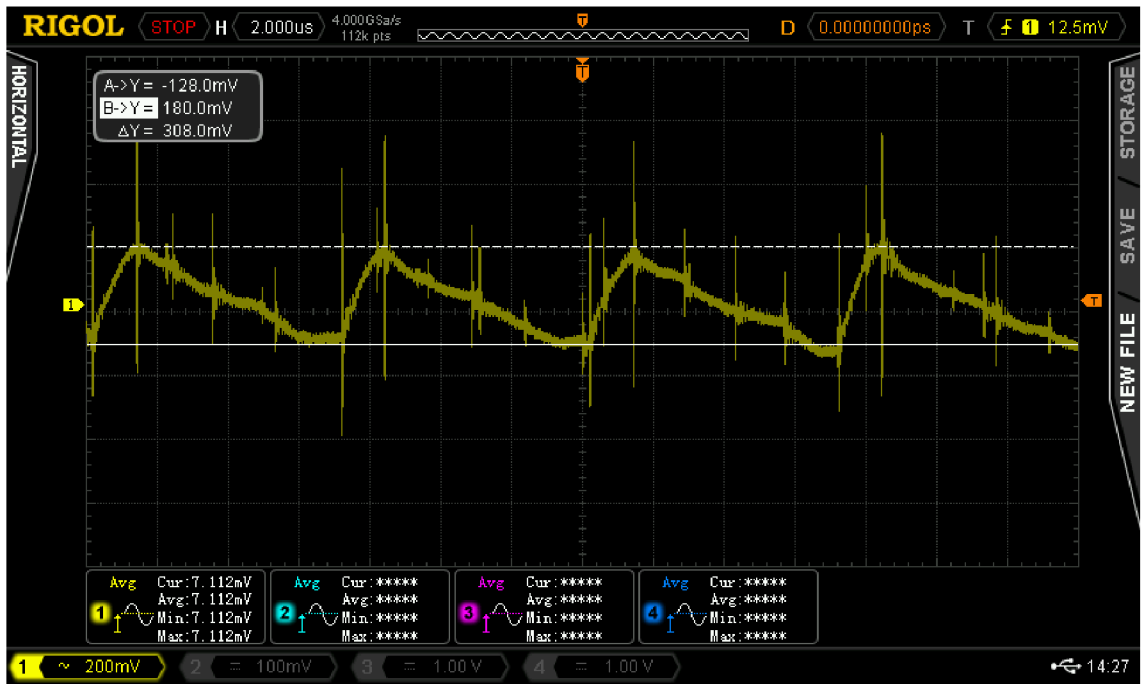
Obr. 9.8: Průběh napětí na *drain* tranzistoru s maximální zátěží, $U_{IN}=13,5$ V

V této podkapitole je vyobrazen na obrázku 9.8 průběh napětí na společném bodě mezi výstupem cívky, anodou diody a elektrodou *drain* tranzistoru. Opět se jedná o výstup pro snižující měnič č. 1. Průběh je měřen pro maximální zátěž a pro nominální napájecí napětí 13,5 V. Opět je velmi dobré, že je průběh hladký, bez zákmitu či rušení. Neobjevují se zde ani žádné velké napěťové překmity, které by se daly očekávat vzhledem ke spínání indukční součástky. Je proto jasné, že filtrace navržené DPS je dostatečná. Vzhledem k absenci napěťových překmitů není nebezpečí zničení polovodičových součástek. Maximální napětí naměřené v tomto bodě je 76,7 V, minimální pak -1,3 V. Náběžná hrana pulsů je 20,2 ns.

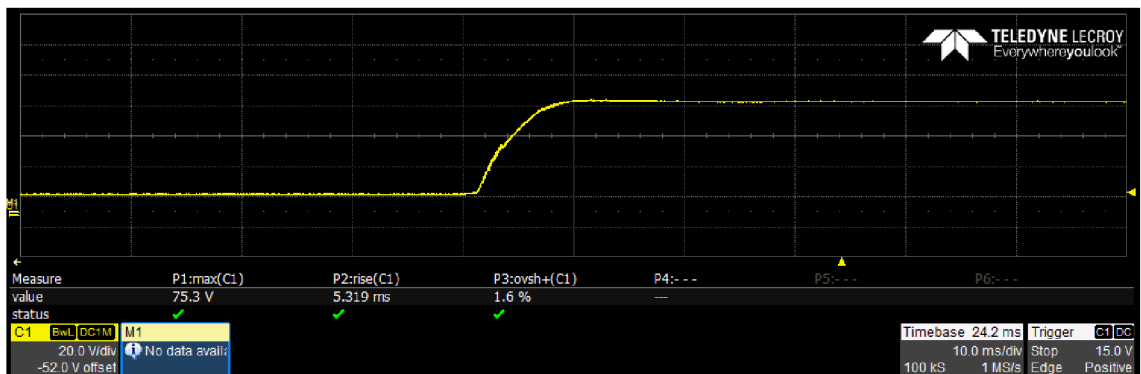
Náběh výstupního napětí a jeho zvlnění

Tato sekce obsahuje několik průběhů. Všechny jsou měřeny na výstupu měniče pro napájení snižujícího měniče č. 1, pro nominální napájecí napětí 13,5 V s maximální zátěží. Na obrázku 9.9 je zobrazen detail výstupního napětí pro určení zvlnění. Podle návrhu nemělo překročit zvlnění 0,2 V. Zvlnění zobrazeného průběhu je maximálně 180 mV, což je v pořádku.

Na obrázku 9.10 je zobrazen náběh výstupního napětí bez připojené zátěže. Zároveň zde není aplikován doporučený softwarový náběh výstupního napětí. Na obrázku 9.11 je vyobrazen stejný náběh s připojenou maximální zátěží. Je vidět, že průběh výstupního napětí je velmi hladký, neobjevují se zde překmity, které by se daly očekávat. Maximální napětí na výstupu je 75,3 V v případě bez zátěže, se zátěží je to 74,7 V. Doba náběhu výstupního napětí je v obou případech kolem 5,4 ms. Změřený překmit napětí je 1,6 % v případě bez zátěže, se zátěží je to pouze 0,5 %. Z průběhu je patrné, že filtrace navržená na DPS je dostatečná, a nedochází zde k nežádoucím jevům.

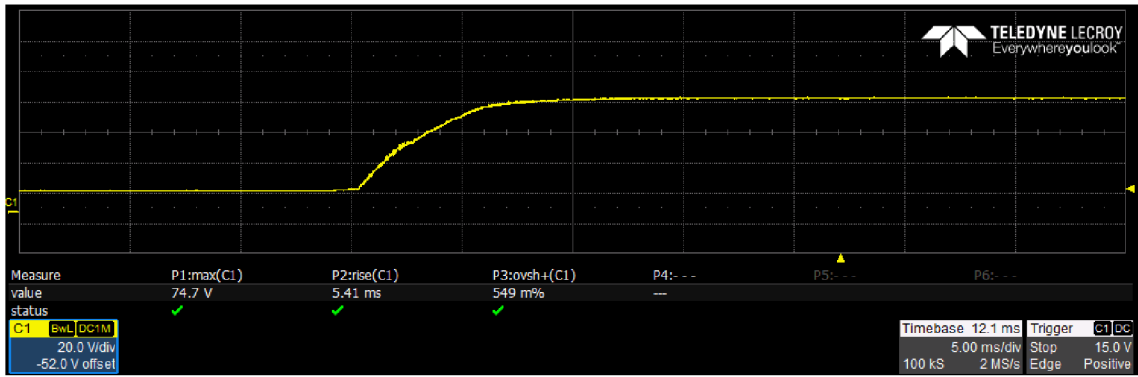


Obr. 9.9: Zvlnění výstupního napětí při maximální zátěži, $U_{IN}=13,5$ V

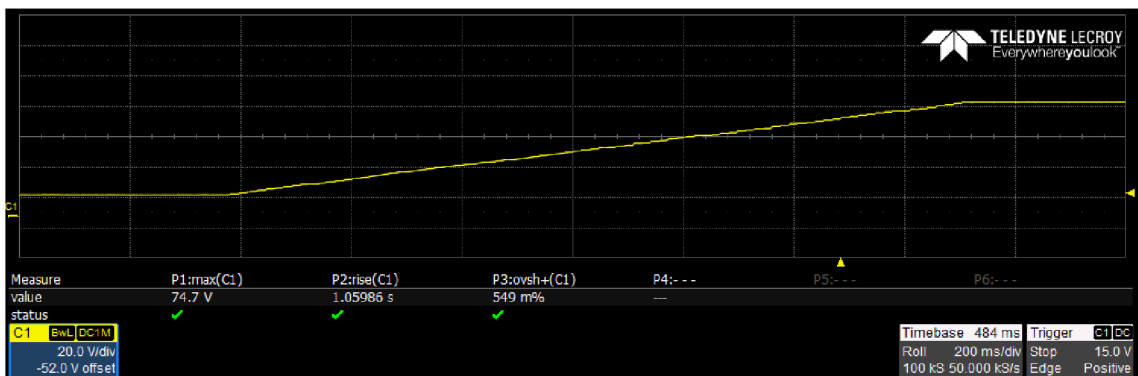


Obr. 9.10: Náběh výstupního napětí bez zátěže a bez SW náběhu, $U_{IN}=13,5$ V

Výrobce doporučuje vytvořit v ovládacím SW umělý náběh napětí. Proto je na obrázku 9.12 zobrazena i tato varianta. Doba, než měnič v tomto případě dosáhne požadované výstupní napětí, je 1,06 s. Překmit je 0,5 % a maximum napětí je 74,7 V.

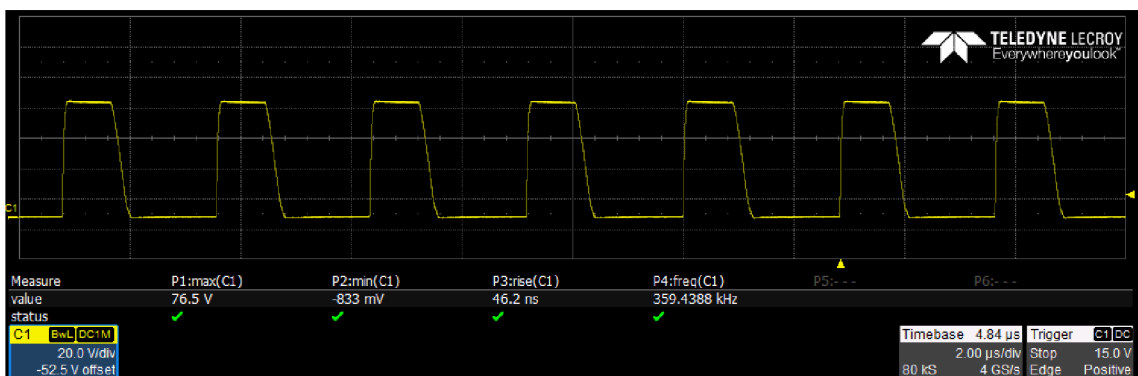


Obr. 9.11: Náběh výstupního napětí se zátěží a bez SW náběhu, $U_{IN}=13,5$ V

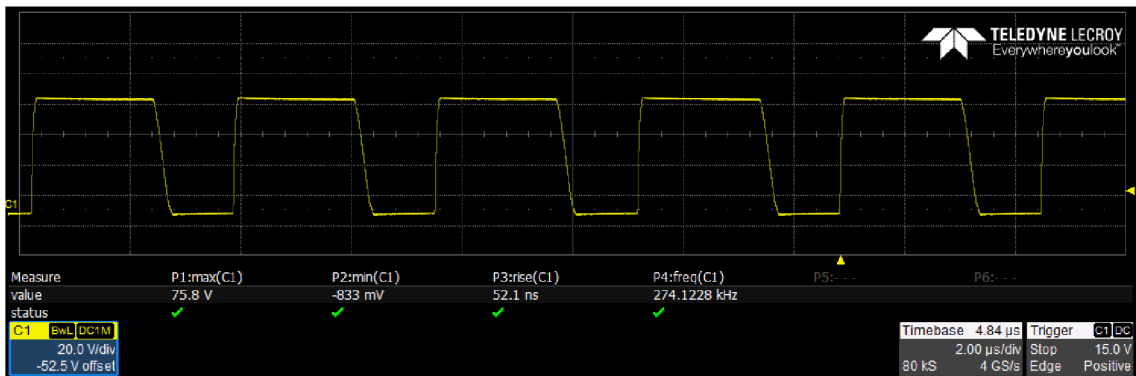


Obr. 9.12: Náběh výstupního napětí se zátěží a se SW náběhem, $U_{IN}=13,5$ V

9.4.3 Sníživí měnič



Obr. 9.13: Průběh napětí na elektrodě *source* tranzistoru kanálu 1, sníživího měniče 1, při plné zátěži

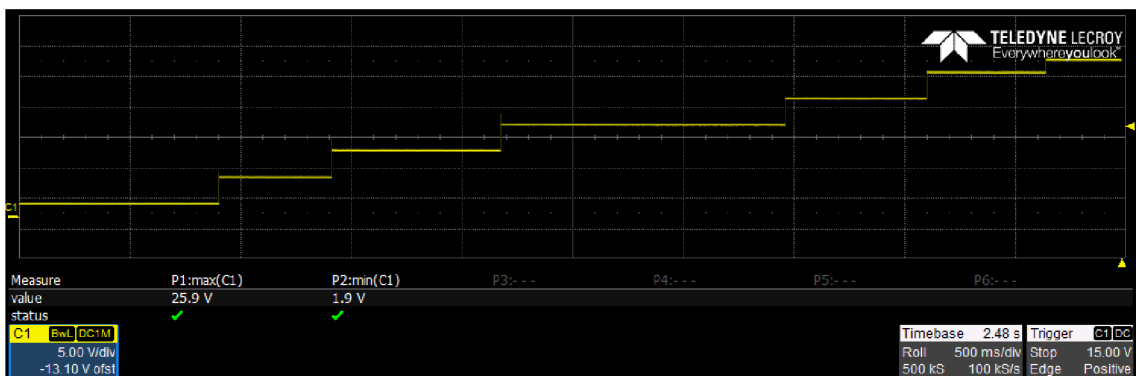


Obr. 9.14: Průběh napětí na elektrodě *source* tranzistoru kanálu 2, snižujícího měniče 1, při plné zátěži

V této podkapitole jsou zobrazeny průběhy napětí na elektrodě *source* tranzistorů ve snižujícím měniči č. 1. Měření probíhalo v obou případech s maximální zátěží. Jsou zde zobrazeny zmíněné průběhy kanálu 1 (obrázek 9.13) a kanálu 2 (obrázek 9.14).

V kapitole 4 je vypočtená frekvence, na které by měl měnič pracovat. Problém je, že vstupní parametry pro výpočet jsou: napětí na vstupu měniče, napětí na výstupu měniče a velikost zvlnění proudu na cívce. Výsledná frekvence je proto velmi odlišná při změně jakéhokoliv parametru.

V tomto případě je situace taková, že indukčnost cívky není vypočtených 1,4 mH ale 1,5 mH, Už tato změna, v případě konstantních zbylých parametrů, vyvolá snížení frekvence o 30 kHz.



Obr. 9.15: Výstupní napětí snižujícího měniče při zvětšování zátěže

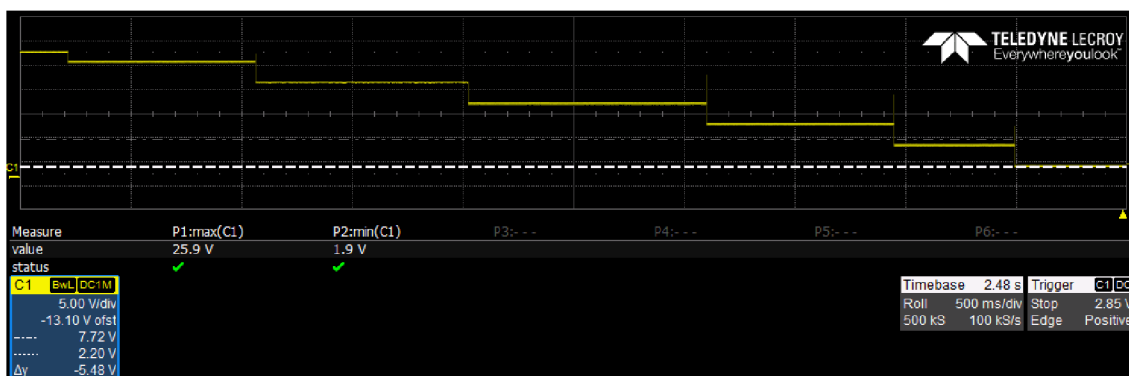
Samozřejmě jsou zde i další proměnné. V čípech je nastavená velikost zvlnění proudu na cívce na hodnotu 15,75 mA až 25,45 mA. Přesnější nastavení není možné,

ale jeho rozptyl opět způsobí velké výkyvy frekvence. Další odchylky frekvence způsobí výstupní napětí, které se liší podle připojené zátěže a podle nastaveného proudu. O střídě spínání se výrobce v datovém listu nezmiňuje, její výpočet je tedy bezpředmětný.

Popsané problémy jsou na průbězích znát. Frekvence signálu, v případě největší zátěže na kanálu 2, je pouhých 274 kHz. V případě maximální zátěže na kanálu 1 je to ale 359 kHz vzhledem k nižšímu výstupnímu napětí. Při tolika proměnných, které obsahuje tento prototyp měniče (nastavitelné proudy, napětí, zvlnění proudu) by bylo jen obtížné udržet frekvenci v požadovaných mezích.

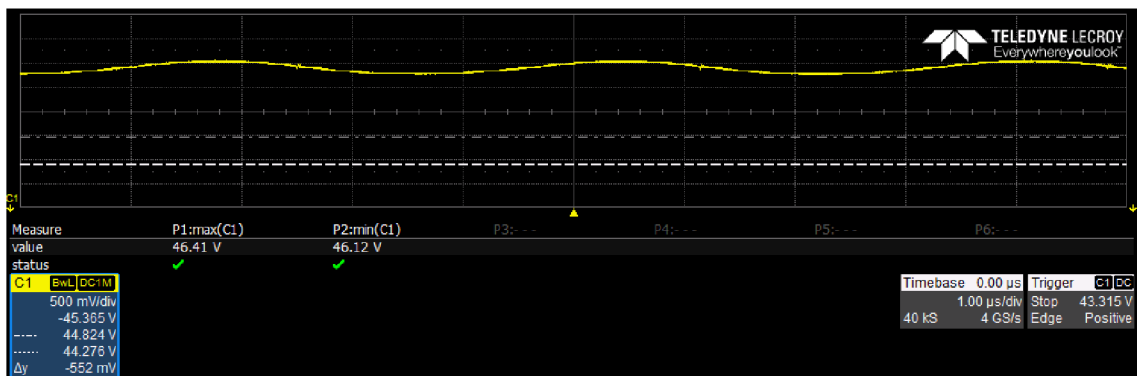
Co se týká ostatních parametrů signálu, opět zde nejsou žádné zákmity či rušení, maximální napětí nepřesahuje hodnotu 76,5 V. Co se týká filtrace, lze ji u snižujícího měniče považovat za dostačující.

Pro zajímavost jsou na obrázku 9.15 a 9.16 naznačeny průběhy výstupního napětí při změně délky výstupního LED řetězce.



Obr. 9.16: Výstupní napětí snižujícího měniče při zmenšování zátěže

Posledním změřeným průběhem je detail napětí na výstupním kanále 2, snižujícího měniče č. 1, při plném zatížení. Průběh se nachází na obrázku 9.17. Zvlnění napětí činí přibližně 145 mV, což odpovídá výpočtu.



Obr. 9.17: Zvlnění výstupního napětí snižujícího měniče

9.4.4 Efektivita zapojení

Efektivitě zapojení je věnována samostatná podkapitola. Důvodem jsou nepříliš uspokojivé výsledky naměřené účinnosti. V kapitole 9.4.1, v tabulce 9.5 se nachází vypočtené hodnoty účinnosti pro několik různých podmínek.

Na první pohled je účinnost velmi nízká. V nejlepším případě je účinnost 61,3 %. Otázkou tedy je, co způsobuje nízkou účinnost. Fakt, že účinnost pro poloviční zatížení je ještě menší, byl celkem očekávaný vzhledem k tomu, že měnič byl navržen na násobně větší zátěž.

Už podle snímání termokamerou (viz kapitola 9.3) bylo podle teplotních ztrát jasné, že účinnost nebude příliš velká.

Bylo tedy třeba určit, ve které části obvodu ke ztrátě účinnosti dochází. Při návrhu byla DPS osazena měřícími body na výstupech všech kanálů jak u zvyšujícího tak snižujícího měniče. To umožnilo zatížit výkonovým rezistorem samostatně zvyšující měnič a změřit jeho účinnost, dále pak napájet snižující měnič externím zdrojem, vynechat tak zvyšující měnič a změřit samostatně účinnost snižujícího měniče.

Účinnost zvyšujícího měniče

V tabulce 9.6 je vypočtená účinnost obou kanálů zvyšujícího měniče. Každý kanál byl měřen samostatně. Účinnost zvyšujícího měniče je měřena včetně ztráty na vstupních obvodech. Měření bylo prováděno na rezistoru takové hodnoty, aby se výkon na něm spotřebovaný blížil spotřebě jednoho snižujícího měniče. Bohužel nebyl k dispozici rezistor s takovým odporem a dostatečným výkonem, aby bylo možné připojit na něj 75 V, proto je měření prováděno při napětí výstupu měniče přibližně 51 V. I přes tyto nedostatky je účinnost každého kanálu skoro 91 %. Je třeba zdůraznit, že je to hodnota včetně vstupních obvodů, signalizačních diod a

Tab. 9.6: Tabulka účinností jednotlivých kanálů zvyšujícího měniče

| Napájecí napětí | | Výstupní kanál č. 1 | | | | |
|-----------------|--------------|---------------------|---------------|--------------|---------------|--------------|
| V_{in} [V] | I_{in} [A] | V_{out} [V] | I_{out} [A] | P_{in} [W] | P_{out} [W] | Účinnost [%] |
| 13,44 | 1,43 | 51,95 | 0,336 | 19,19 | 17,46 | 90,95 |
| | | Výstupní kanál č. 2 | | | | |
| V_{in} [V] | I_{in} [A] | V_{out} [V] | I_{out} [A] | P_{in} [W] | P_{out} [W] | Účinnost [%] |
| 13,45 | 1,36 | 50,7 | 0,328 | 18,28 | 16,63 | 90,98 |

mikrokontroleru. Je tedy zřejmé, že ve zvyšujícím měniči problém s velkou ztrátou účinností není a celkově se dá jeho návrh považovat za zdařilý.

Účinnost snižujícího měniče

Měření účinnosti snižujícího měniče probíhalo následovně: na vstup snižujícího měniče číslo 1 bylo přivedeno napětí 60 V a následně byly spuštěny všechny výstupní kanály na maximální výkon. Výsledky měření jsou v tabulce 9.7. Druhý měnič byl proměřen pro stav, kdy je 2. kanál odpojen, stejně jako tomu bylo u všech předchozích měření. Toto měření je zaneseno v tabulce 9.8. Napájení ze zdroje umožnilo, že bylo možné spustit i 2. kanál. Měření je proto provedeno i se spuštěným 2. kanálem (výsledky jsou v tabulce 9.9). Funkčnost byla umožněna pravděpodobně jiným přípojným místem napájecího napětí.

Tab. 9.7: Tabulka účinnosti snižujícího měniče číslo 1, $U_{in}=60,18$ V, $I_{in}=291$ mA

| Kanál 1 | | Kanál 2 | | Kanál 3 | | | | Účinnost [%] |
|---------------|----------------|---------------|----------------|---------------|----------------|--------------|---------------|--------------|
| V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | P_{in} [W] | P_{out} [W] | |
| 25,66 | 146,00 | 45,44 | 138,18 | 25,66 | 149,00 | 17,51 | 13,85 | 79,10 |

Opět se ukázalo, v případě snižujícího měniče číslo 2, že s rostoucí zátěží roste i účinnost zapojení. Nicméně, faktem zůstává, že celková účinnost zapojení je nízká. Dosahuje maximálně 79,1 %. Dá se předpokládat, že pokud by byla zátěž větší, účinnost by ještě vzrostla. Na druhé straně, při zvyšujícím se napětí na vstupu snižujícího měniče prudce rostla teplota obvodů na desce. Vzhledem k tomu, že měření je provedeno pro napájecí napětí 60 V, je jasné, že při napájení 75 V by účinnost o něco klesla.

Tab. 9.8: Tabulka účinnosti snižujícího měniče číslo 2 s odpojeným 2. kanálem, $U_{in}=60,22$ V, $I_{in}=183$ mA

| Kanál 1 | | Kanál 2 | | Kanál 3 | | P_{in} [W] | P_{out} [W] | Účinnost [%] |
|------------------|-------------------|------------------|-------------------|------------------|-------------------|-----------------|------------------|--------------|
| V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | | | |
| 25,79 | 145,00 | - | - | 25,76 | 147 | 11,02 | 7,53 | 68,29 |

Tab. 9.9: Tabulka účinnosti snižujícího měniče číslo 2 s připojeným 2. kanálem, $U_{in}=60,20$ V, $I_{in}=321$ mA

| Kanál 1 | | Kanál 2 | | Kanál 3 | | P_{in} [W] | P_{out} [W] | Účinnost [%] |
|------------------|-------------------|------------------|-------------------|------------------|-------------------|-----------------|------------------|--------------|
| V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | V_{out} [V] | I_{out} [mA] | | | |
| 25,79 | 145,00 | 49,21 | 140,61 | 25,76 | 147 | 19,32 | 14,45 | 74,75 |

Další věc, kterou je třeba uvážit je to, že při měření účinnosti zvyšujícího měniče byl měřen vždy pouze jeden kanál. Pokud by tedy byly připojeny oba kanály zároveň, pravděpodobně by vzrostly i ztráty na vstupních filtračních obvodech vlivem většího proudu v obvodech.

Zvýšení účinnosti by šlo, při návrhu další verze prototypu, dosáhnout několika způsoby. Nejdůležitějším krokem by bylo omezit přehřívání obvodu tak, jak to bylo popsáno v kapitole 9.3. Další z možností by zahrnovala zrychlení spínání výstupních tranzistorů zmenšením rezistoru na elektrodě *gate* tranzistorů. Součástí nového řešení by rovněž mělo být optimalizování výpočtů součástí obvodu na použitou zátěž.

10 ZÁVĚR

Cílem diplomové práce bylo navrhnout DC-DC měnič pro napájení matrix beam modulu. Úkolem bylo využít čipy *ASL2507SHN* a *ASL3417SHN* a demonstrovat jejich funkci. Z těchto čipů bylo cílem navrhnout měnič s topologií boost-buck, tedy obsahující vzestupnou a sestupnou část. Prototyp by měl být jednoduše říditelný z grafického uživatelského rozhraní na počítači. Zadavatel práce definoval základní požadavky na obvod jako je napájecí napětí, počet výstupních kanálů a výkon zátěže. Součástí požadavků byly rovněž podmínky, ve kterých by měl měnič fungovat. Součástí návrhu by měla být i základní filtrace napájecího napětí a ochranné obvody.

První část práce se věnuje seznámení s použitými obvody a popisu jejich vlastností. Velkou částí teoretického rozboru práce je studium technologie stejnosměrných měničů napětí, zejména popis jejich funkce.

Praktická část diplomové práce je rozdělena do více částí popisujících jednotlivé kroky v realizaci prototypu stejnosměrného měniče:

- První krátká část obsahuje rozbor projektu. Je zde blokové schéma a návrh, jakým způsobem bude projekt realizován.
- Další část se věnuje výpočtům základních parametrů měniče a tím pádem se stává podstatnou částí práce. Vzhledem k topologii měniče je i výpočet celku rozdělen do dvou hlavních částí. Jsou jimi výpočty zvyšující i snižující části měniče. V této části se úspěšně podařilo vypočítat parametry obvodů včetně dodržení všech zadaných požadavků.
- Následující úsek práce je věnovaný zejména výběru polovodičových součástek pro realizaci projektu. Obsahem je nejen samotný výběr součástek, ale i výpočty výkonových ztrát, ke kterým bude na reálném prototypu docházet. Toto zamyšlení je, vzhledem k předpokládaným proudovým odběrům v obvodu, nutné, aby mohly být navrženy dostatečné chladičí plochy.
- Další kapitola práce se zabývá simulacemi vypočtených obvodů. Jeho součástí je zejména ověření základních parametrů vypočtených v práci, ale taky řešení případných nečekaných jevů, které v obvodu mohou vzniknout. Tímto jevem je myšleno kmitání (překmity při náběhu měniče), které vznikalo v LC obvodu obou stupňů měniče. Bylo tedy nutné navrhnout tlumící členy pro zmíněné části obvodu. Upravené schéma měniče bylo znovu odsimulováno, výsledky dopadly uspokojivě a bylo ověřeno, že měnič je s vypočtenými parametry obvodu realizovatelný.
- Po zmíněných krocích bylo možné přistoupit k návrhu celého schéma prototypu měniče. Návrh je rozdělen do pěti základních částí. Jsou v něm realizovány návrhy vstupních obvodů, zátěže, řídicí digitální logiky a obou stupňů měniče.
- Po úspěšném návrhu schématu bylo možné vytvořit desku plošného spoje. V

diplomové práci jsou popsány základní kroky návrhu s objasněním kritických částí tvorby desky plošného spoje prototypu DC-DC měniče. Po kontrole a odsouhlasení návrhu konzultantem práce a zadavatelem projektu byl prototyp měniče vyroben.

- Vyrobený vzorek bylo následně možné oživit a otestovat základní funkce. První spuštění dopadlo úspěšně, byla otestována komunikace a mohlo se přistoupit k tvorbě programového vybavení. Tvorba software je velmi obsáhlá a je jí věnována značná část diplomové práce. Úspěšně se podařilo vytvořit software pro obsluhu prototypu měniče umožňující jeho libovolnou konfiguraci. Uživatelské rozhraní je jednoduše ovladatelné a přehledné.
- Závěrečná část diplomové práce je věnována měření základních parametrů prototypu DC-DC měniče. Jsou zde porovnány naměřené hodnoty s vypočtenými, dále se zde nachází rozbor vzniklých problémů a návrhy jejich řešení. V závěru práce je diskutována efektivita zapojení.

V práci se podařilo realizovat návrh DC-DC měniče. Jsou splněny základní body zadání. Funkčnost měniče byla ověřena měřením. Součástí práce jsou návrhy kroků, pro případný návrh další verze prototypu, za účelem zlepšení jeho parametrů. Výstupem projektu je vzorek DC-DC měniče s grafickým uživatelským rozhraním a programovým vybavením pro mikrokontroler osazený na desce plošného spoje. Výstup rovněž obsahuje základní konfigurační soubory pro snadné spuštění měniče uživatelem.

Pro případný další vývoj a rozšíření prototypu měniče by bylo možné realizovat několik kroků. Bylo by vhodné vyřešit přehřívání obvodů snižujícího měniče. Dále by bylo dobré optimalizovat účinnost snižujících částí měniče. Nedílnou součástí nového návrhu desky plošného spoje by byla optimalizace chladících ploch stávajícího návrhu, zejména v místech, kde je chlazení předimenzováno, a tyto plochy zmenšit pro úsporu místa na plošném spoji. Další vývoj by se měl rovněž zaměřit na efektivnější rozložení zemnicí plochy, které zajistí vylepšení funkcí všech čipů.

LITERATURA

- [1] KREJČIŘÍK, Alexandr. *DC/DC měniče*. Praha: BEN, 2001. ISBN 80-730-0045-8.
- [2] NOVOTNÝ, Vlastislav, Pavel VOREL a Miroslav PATOČKA. *Návrh elektronických zařízení: přednášky*. VUT Brno, UREL, 2000.
- [3] *Analysis of Four DC-DC Converters in Equilibrium. All about circuits* [online]. 6. června 2015 [cit. 2019-09-02]. Dostupné z: <https://www.allaboutcircuits.com/technical-articles/analysis-of-four-dc-dc-converters-in-equilibrium/>
- [4] ROBERTS, Steve. *DC/DC BOOK OF KNOWLEDGE: Practical tips for the User*. Second Edition. Gmunden, Austria, 2015.
- [5] HAUKE, Brigitte. *Basic Calculation of a Buck Converter's Power Stage, application report* [online]. Texas Instruments, 2015 [cit. 2019-10-23]. Dostupné z: <http://www.ti.com/lit/an/slva477b/slva477b.pdf>
- [6] HAUKE, Brigitte. *Basic Calculation of a Boost Converter's Power Stage, application report* [online]. Texas Instruments, 2014 [cit. 2019-11-23]. Dostupné z: <http://www.ti.com/lit/an/slva372c/slva372c.pdf>
- [7] HEGARTY, Timothy. *An overview of conducted EMI specifications for power supplies* [online]. Texas Instruments, 2018 [cit. 2019-10-28]. Dostupné z: <http://www.ti.com/lit/wp/slyy136/slyy136.pdf>
- [8] *ASL1507SHN; ASL2507SHN: Single-phase and two-phase boost converter with limp-home mode, product datasheet* [online]. NXP Semiconductors, 2018 [cit. 2019-09-04]. Dostupné z: https://www.nxp.com/docs/en/data-sheet/ASL1507_AS2507.pdf
- [9] *ASL3417SHN: Enhanced three channel LED buck driver with limp-home, product datasheet* [online]. NXP Semiconductors, 2017 [cit. 2019-09-04]. Dostupné z: <https://www.nxp.com/docs/en/data-sheet/ASL3417SHN.pdf>
- [10] *V10P10: High Current Density Surface Mount TMBS® (Trench MOS Barrier Schottky) Rectifier Ultra Low, product datasheet* [online]. Vishay General Semiconductor, 2019 [cit. 2019-12-02]. Dostupné z: <http://www.vishay.com/docs/89006/v10p10.pdf>

- [11] *Si4056DY: N-Channel 100 V (D-S) MOSFET, product datasheet* [online]. Vishay Siliconix, 2012 [cit. 2019-12-02]. Dostupné z: <http://www.vishay.com/docs/62662/si4056dy.pdf>
- [12] *SiR606BDP: N-Channel 100 V (D-S) MOSFET, product datasheet* [online]. Vishay Siliconix, 2017 [cit. 2019-12-02]. Dostupné z: <http://www.vishay.com/docs/75576/sir606bdp.pdf>
- [13] *FDB86102LZ: N-Channel PowerTrench® MOSFET, product datasheet* [online]. ON Semiconductor, 2011 [cit. 2019-12-09]. Dostupné z: <https://www.onsemi.com/pub/Collateral/FDB86102LZ-D.pdf>
- [14] *LW G6CP: Advanced Power TOPLED®, product datasheet* [online]. OSRAM Opto Semiconductors, 2019 [cit. 2019-12-10]. Dostupné z: https://dammedia.osram.info/media/resource/hires/osram-dam-6030638/LW%20G6CP_EN.pdf
- [15] Kolka, Zdeněk - ústní sdělení (profesor VUT FEKT UREL, Technická 3082/12, Královo Pole, 61600, Brno, Česká republika) dne 4. prosince 2019.
- [16] Opravil, Petr - ústní sdělení (HW developer HELLA AUTOTECHNIK NOVA, Technologická 372/2 Ostrava (Poruba), 70800, Ostrava, Česká republika)
- [17] *8-bit Atmel Microcontroller with 128KBytes In-SystemProgrammable Flash: ATmega128, product datasheet* [online]. Atmel Corporation, 2016 [cit. 2019-09-04]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/doc2467.pdf>
- [18] Povalač, Aleš. *AVR jádro a periferie: Mikrokontroléry pro pokročilé aplikace. Brno*, 2016. Prezentace. UREL, VUT v BRNĚ.
- [19] Povalač, Aleš. *RTOS a zásady návrhu embedded systémů: Mikrokontroléry pro pokročilé aplikace. Brno*, 2018. Prezentace. UREL, VUT v BRNĚ.
- [20] Serial Peripheral Interface Bus. *Wikipedia: The free encyclopedia* [online]. San Francisco: Wikimedia Foundation, 10.08.2019 [cit. 2019-09-15]. Dostupné z: https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus
- [21] FRÝZA, Tomáš. *Řízení sériové komunikace: Mikroprocesorová technika a embedded systémy. Brno*, 2017. Prezentace. UREL, VUT v BRNĚ.
- [22] Universal asynchronous receiver-transmitter. *Wikipedia: The free encyclopedia* [online]. San Francisco: Wikimedia Foundation, 10.08.2019 [cit. 2019-09-18].

Dostupné z: https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter

- [23] ORDONEZ, Martin. *Power Electronics - MOSFET Power Losses*. In: YouTube [online]. 25.9.2017 [cit. 2019-11-23]. Dostupné z: <https://www.youtube.com/watch?v=KSOHwVoxpzg>

SEZNAM ZKRATEK

| | |
|--------|--|
| DC-DC | Direct Current-Direct Current - stejnosměrný proud-stejnosměrný proud |
| MOSFET | Metal Oxide Semiconductor Field Effect Transistor |
| DCM | Discontinuous Current Mode - režim nespojitých proudů |
| CCM | Continuous Current Mode - režim nespojitých proudů |
| LED | Light Emitting Diode - světlo emitující dioda |
| SMD | Surface Mount Device - zařízení pro povrchovou montáž |
| UART | Universal Asynchronous Receiver and Transmitter - univerzální asynchronní seriové rozhraní |
| USB | Universal Serial Bus - univerzální sériová sběrnice |
| NDA | Non-Disclosure Agreement - dohoda o mlčenlivosti |
| SPI | Serial Peripheral Interface - seriové periferní rozhraní |
| SS | Slave Select - výběr podřízeného zařízení |
| MOSI | Master Output Slave Input - směr dat z nadřízeného do podřízeného zařízení |
| MISO | Master Input Slave Output - směr dat z podřízeného do nadřízeného zařízení |
| PWM | Pulse Width Modulation - pulzně šířková modulace |
| EMC | Electromagnetic Compatibility - elektromagnetická kompatibilita |
| EMI | Electromagnetic Interference - elektromagnetická interference |
| EMS | Electromagnetic Susceptibility - elektromagnetická odolnost |
| TVS | Transient Voltage Suppression |
| LDO | Low-Dropout regulator |
| DPS | Deska Plošného Spoje |
| LHM | Limp-Home Mode |
| NVM | Non-Volatile Memory - energeticky nezávislá paměť |
| SW | Software - programové vybavení |
| RTOS | Real Time Operating System - operační systém reálného času |
| GUI | Graphic User Inteface - grafické uživatelské rozhraní |

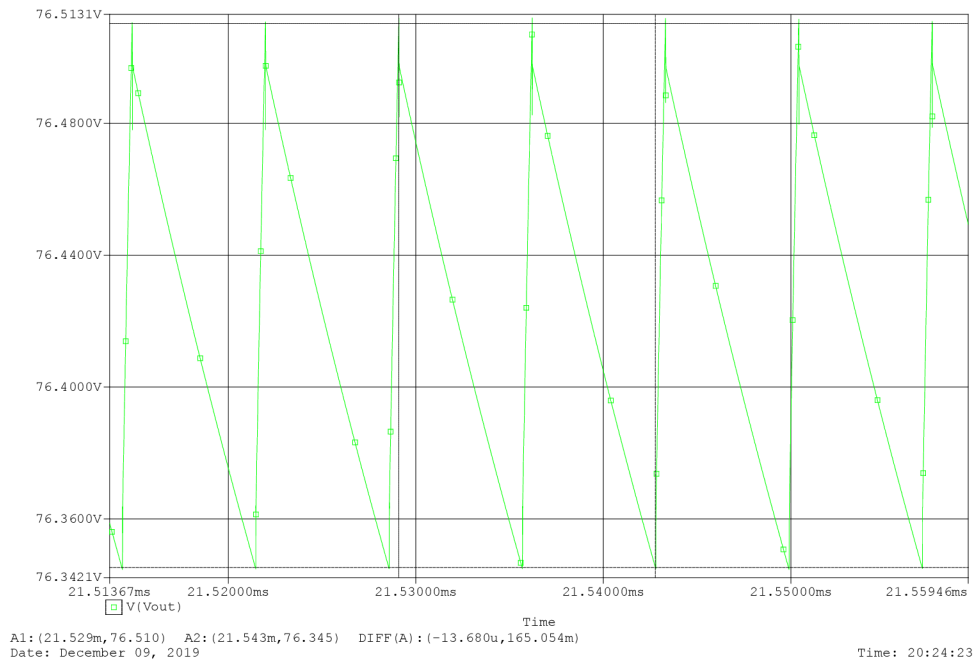
SEZNAM PŘÍLOH

| | | |
|----------|--|------------|
| A | Doplňující výstupy simulací obvodů | 121 |
| A.1 | Zvyšující měnič | 121 |
| A.2 | Snižující měnič | 125 |
| B | Schéma obvodu | 127 |
| B.1 | Převodník mezi USB a UART | 127 |
| B.2 | Zvyšující měnič | 128 |
| B.3 | Snižující měnič | 129 |
| B.4 | Řídící mikrokontroler | 131 |
| B.5 | Zátěž | 132 |
| C | DPS | 134 |
| D | Kritické úseky programu mikrokontroleru ATmega128 | 138 |
| E | Kritické úseky programu uživatelského rozhraní | 144 |
| F | Konfigurace registrů měničů při měření | 147 |
| G | Přiložené soubory | 152 |

A DOPLŇUJÍCÍ VÝSTUPY SIMULACÍ OBVODŮ

Příloha obsahuje doplňující grafy simulací jednotlivých částí měniče. Simulace jsou realizovány v programu PSpice.

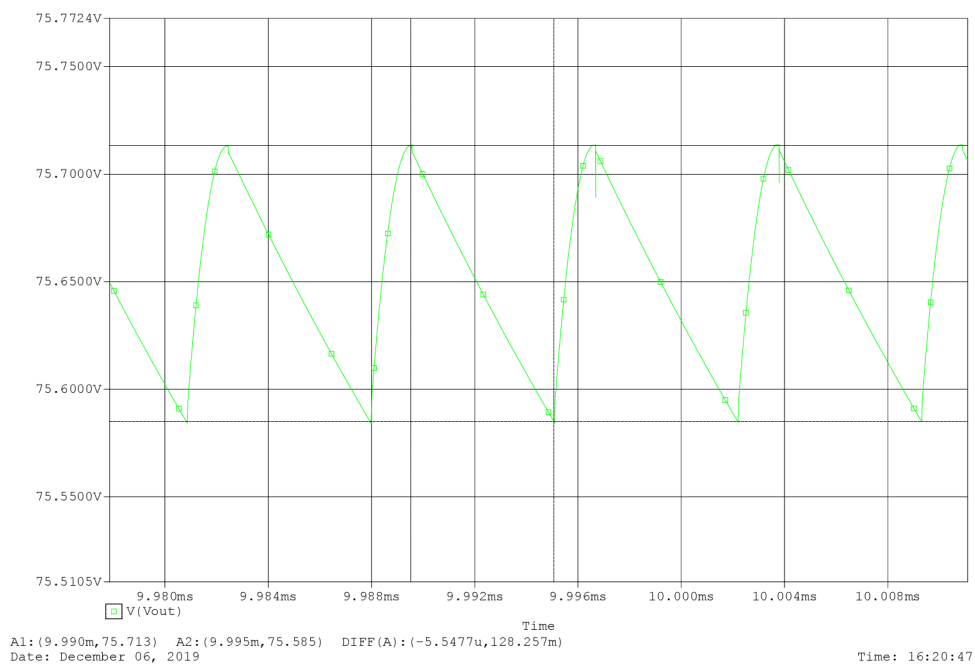
A.1 Zvyšující měnič



Obr. A.1: Zvlnění napětí na výstupu zvyšujícího měniče, $U_{IN} = 5,84 \text{ V}$

Tab. A.1: Efektivní a střední hodnoty proudu polovodičů, $U_{IN} = 5,84 \text{ V}$

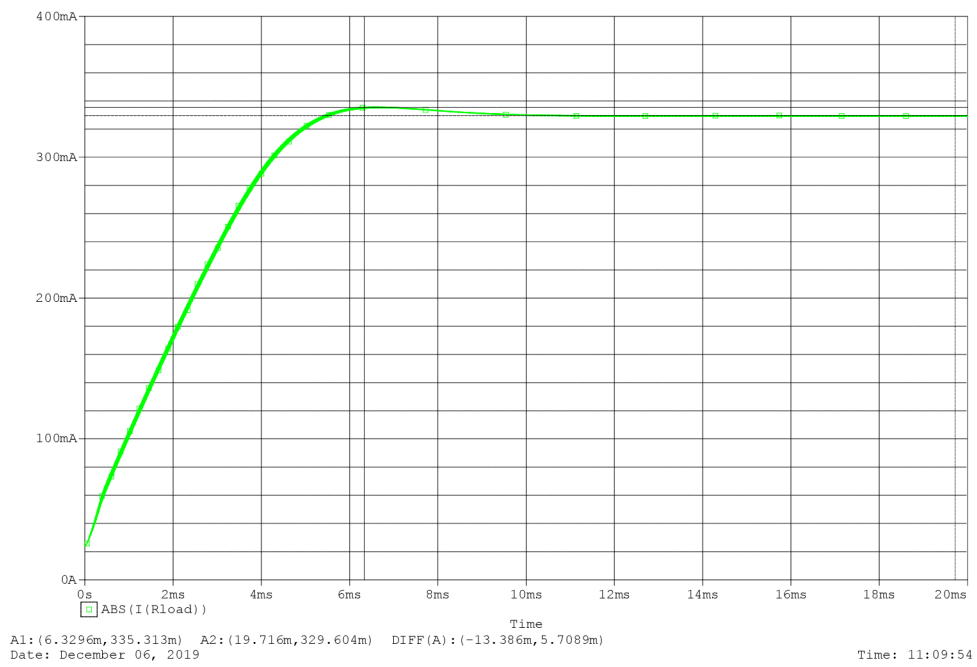
| Veličina | Hodnota | Jednotka |
|--------------|---------|----------|
| AVG(I(U1:1)) | 329,86 | mA |
| RMS(I(U1:1)) | 1,232 | A |
| AVG(I(U2:D)) | 4,185 | A |
| RMS(I(U2:1)) | 4,353 | A |



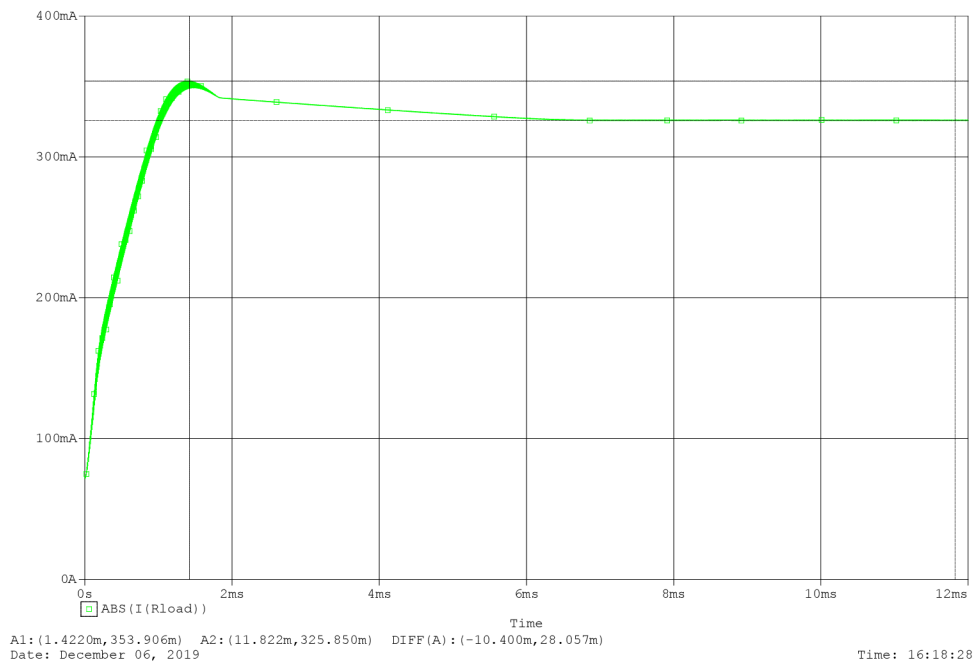
Obr. A.2: Zvlnění napětí na výstupu zvyšujícího měniče, $U_{IN} = 16,96$ V

Tab. A.2: Efektivní a střední hodnoty proudu polovodičů, $U_{IN} = 16,96$ V

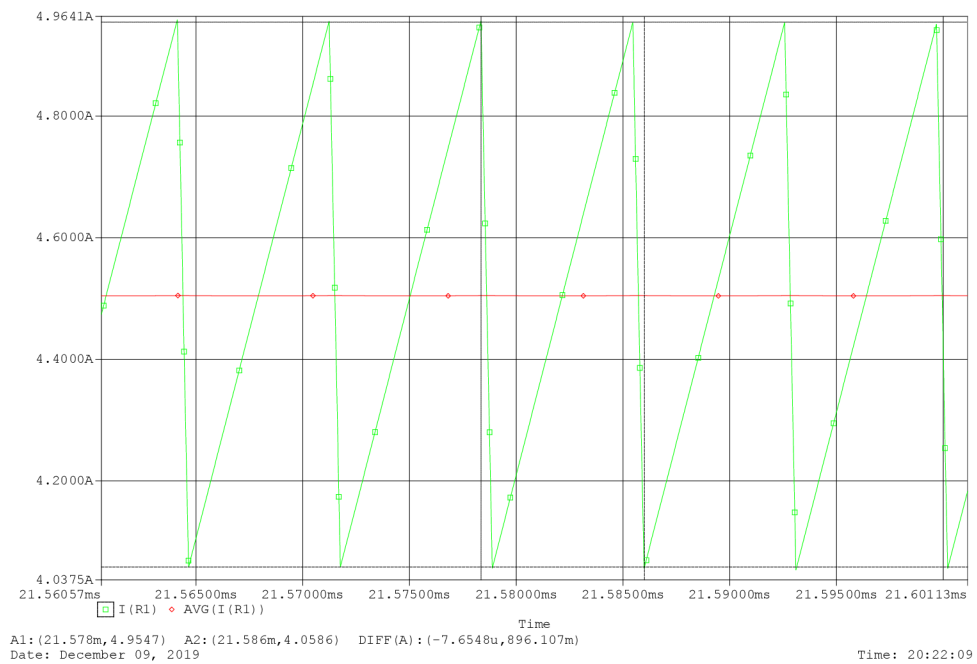
| Veličina | Hodnota | Jednotka |
|--------------|---------|----------|
| AVG(I(U1:1)) | 326,24 | mA |
| RMS(I(U1:1)) | 769,45 | mA |
| AVG(I(U2:D)) | 1,155 | A |
| RMS(I(U2:D)) | 1,430 | A |



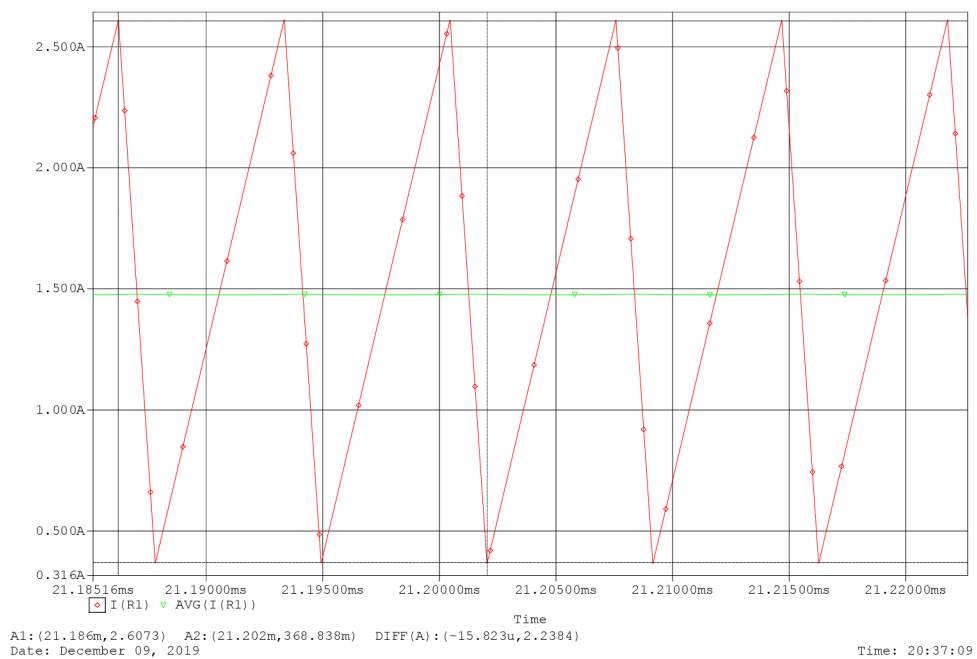
Obr. A.3: Výstupní proud vzestupného měniče, $U_{IN} = 5,84 \text{ V}$



Obr. A.4: Výstupní proud vzestupného měniče, $U_{IN} = 16,96 \text{ V}$

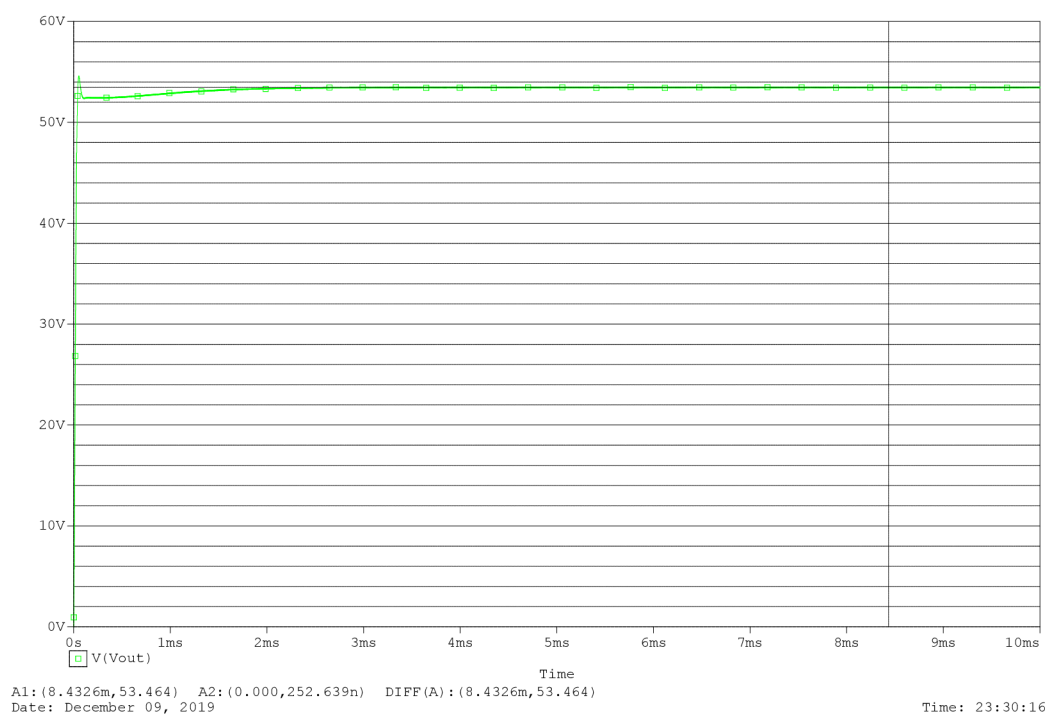


Obr. A.5: Detail průběhu vstupního proudu vzestupného měniče, $U_{IN} = 5,84 \text{ V}$



Obr. A.6: Detail průběhu vstupního proudu vzestupného měniče, $U_{IN} = 16,96 \text{ V}$

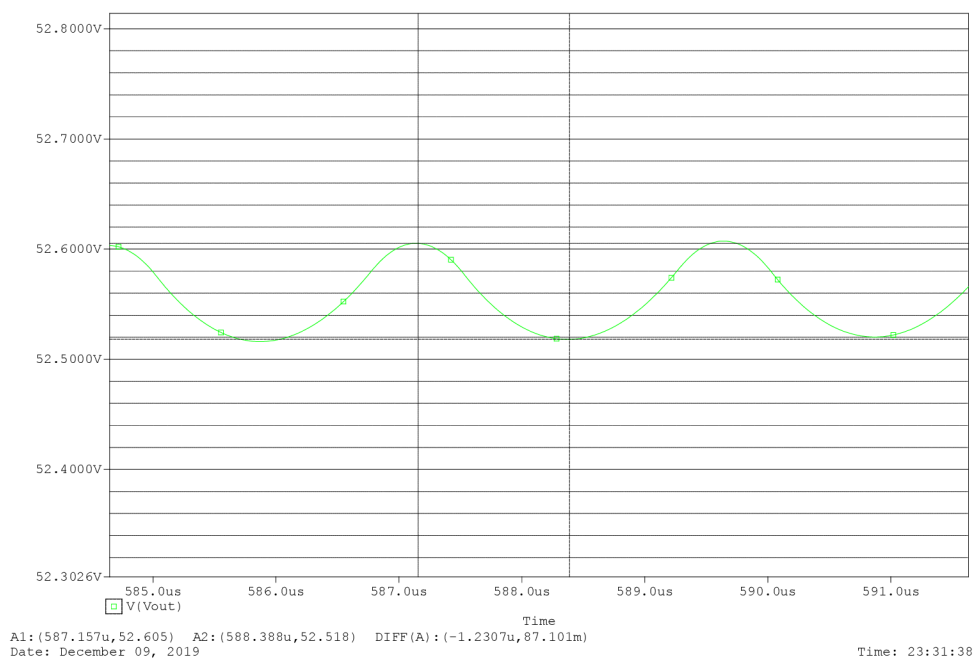
A.2 Snižující měnič



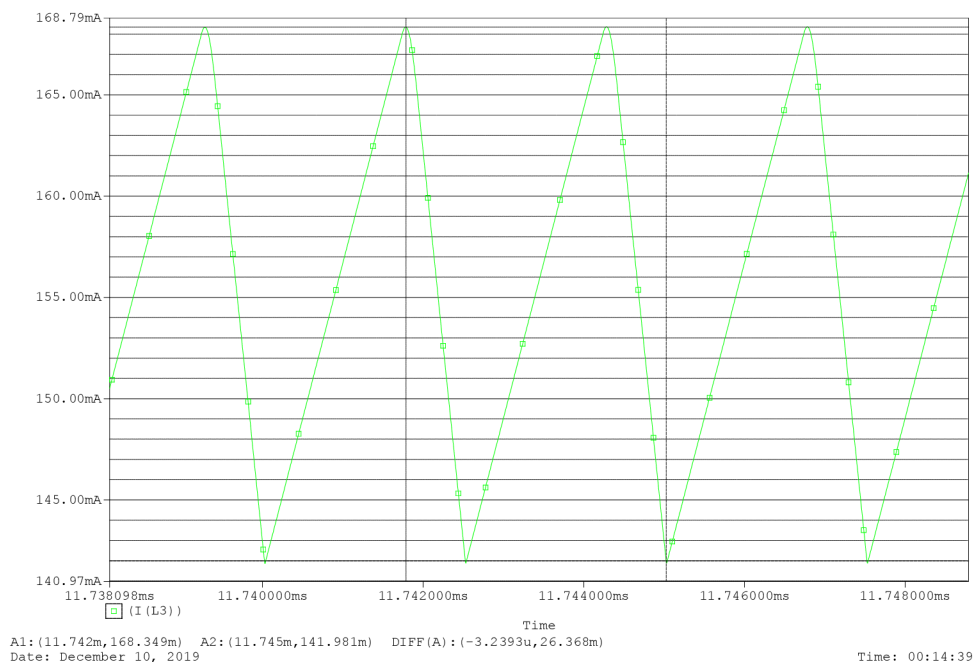
Obr. A.7: Výstupní napětí snižujícího měniče v intervalu 10 ms po spuštění simulace

Tab. A.3: Efektivní a střední hodnoty proudu polovodičů

| Veličina | Hodnota | Jednotka |
|--------------|---------|----------|
| AVG(I(U2:1)) | 35,83 | mA |
| RMS(I(U2:1)) | 136,01 | mA |
| AVG(I(U3:D)) | 119,68 | mA |
| RMS(I(U3:D)) | 177,06 | mA |

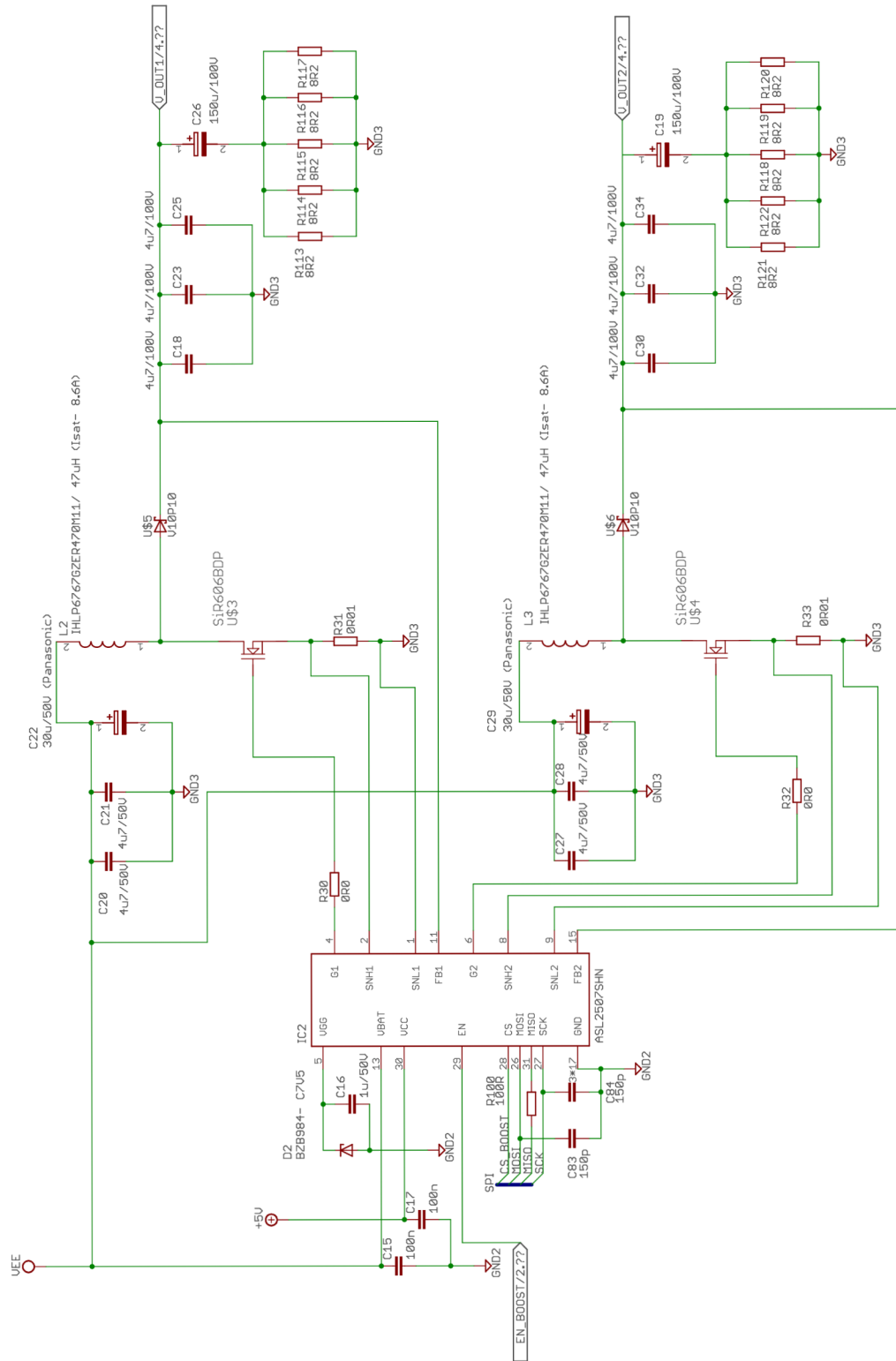


Obr. A.8: Zvlnění napětí na výstupu snižujícího měniče



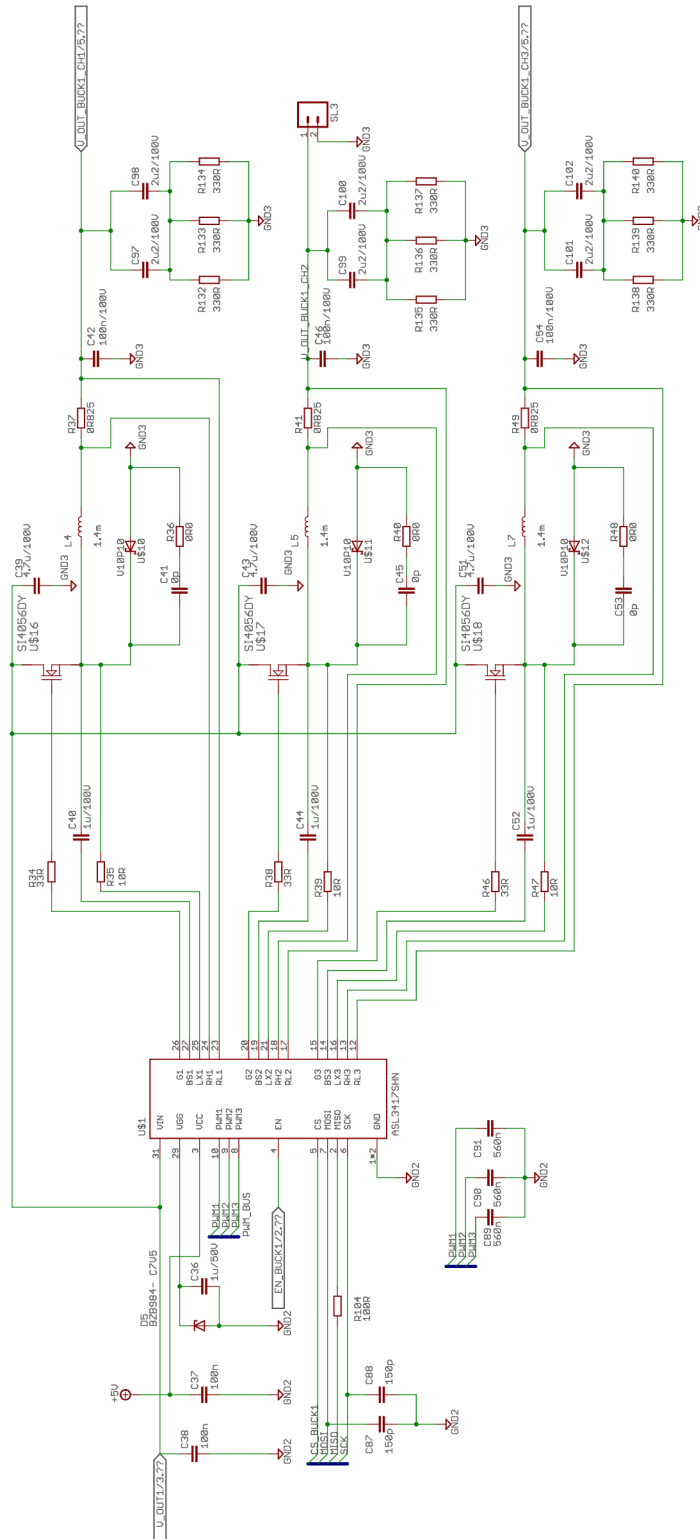
Obr. A.9: Zvlnění proudu na cínce snižujícího měniče

B.2 Zvyšující měnič

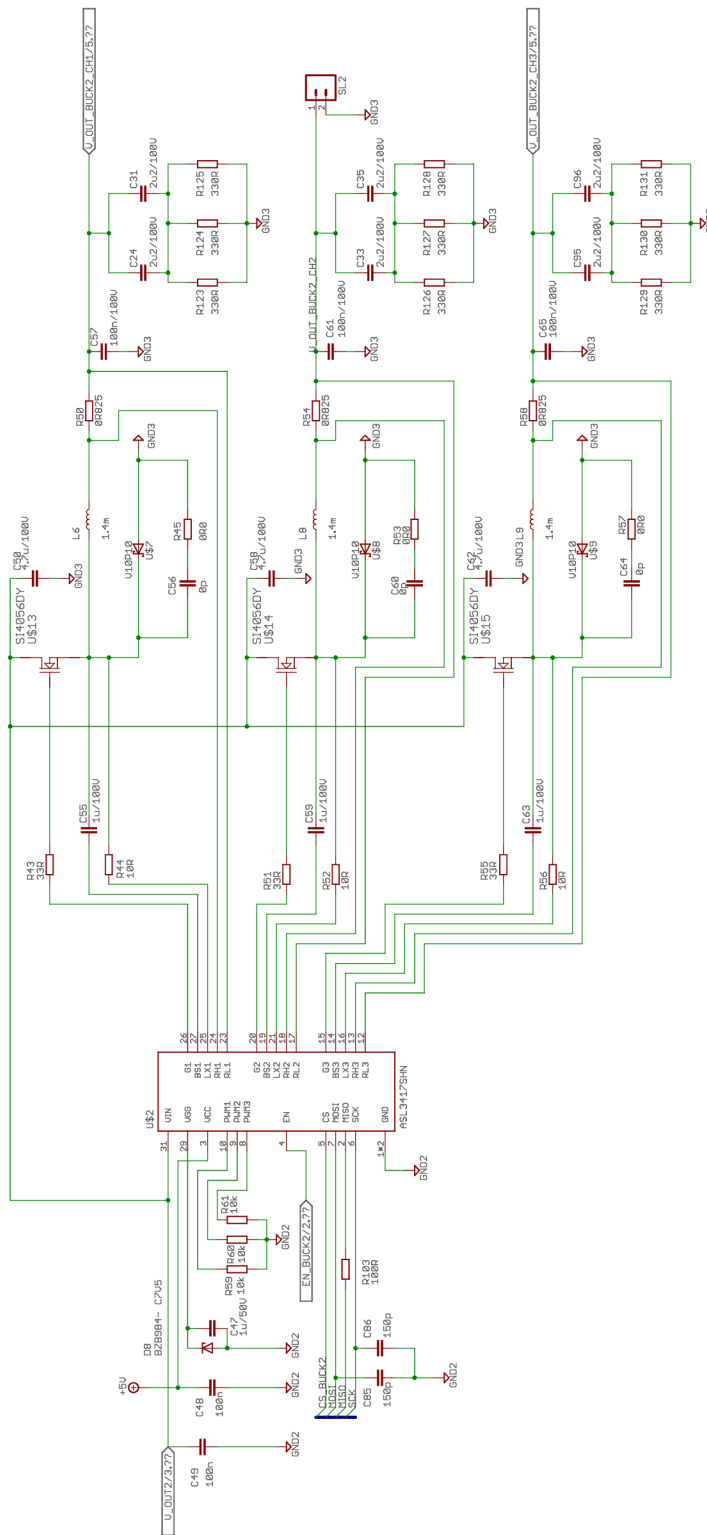


Obř. B.2: Schéma zapojení zvyšující části DC-DC měniče

B.3 Snižující měnič

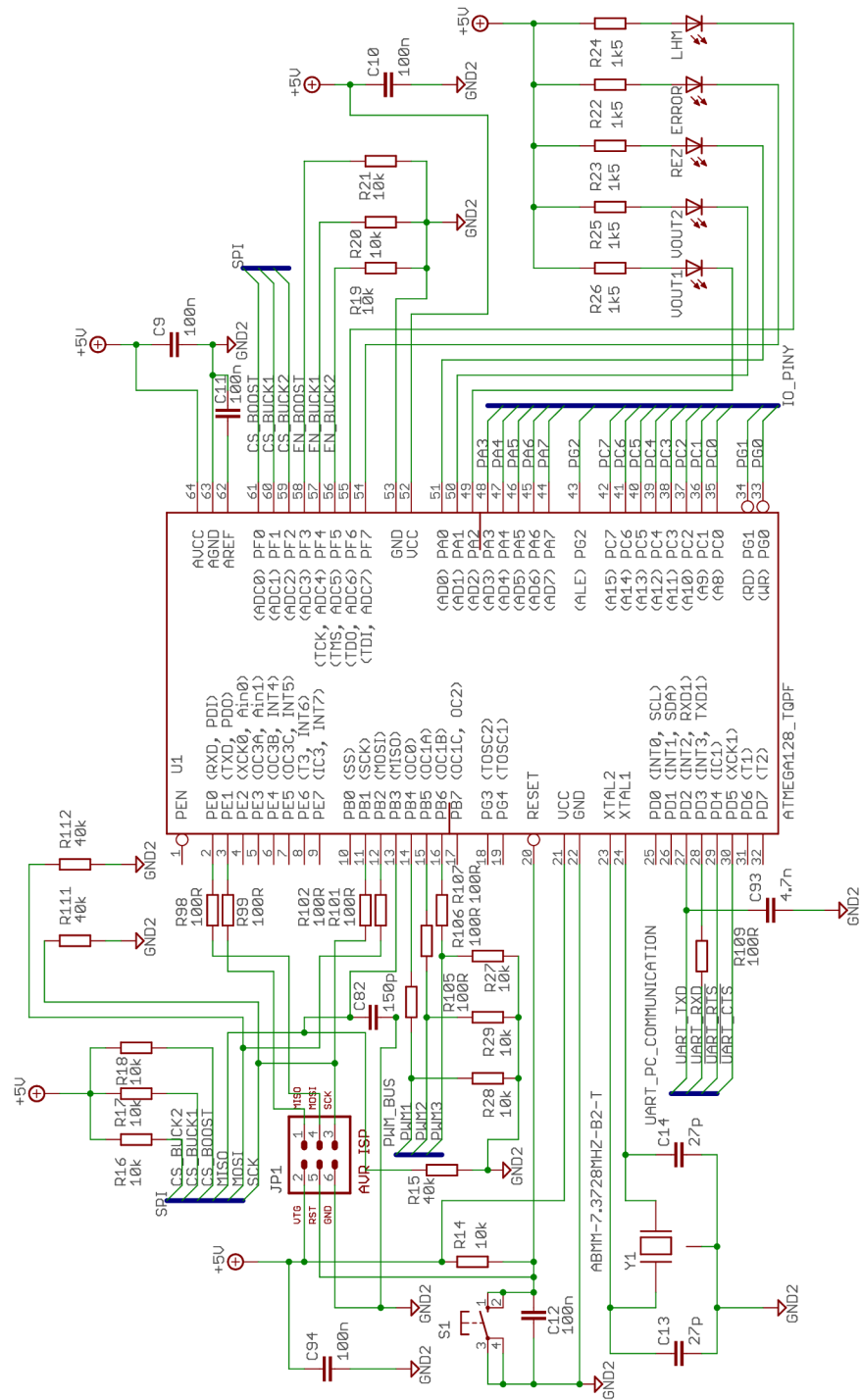


Obr. B.3: Schéma zapojení první ze dvou snižujících částí DC-DC měniče



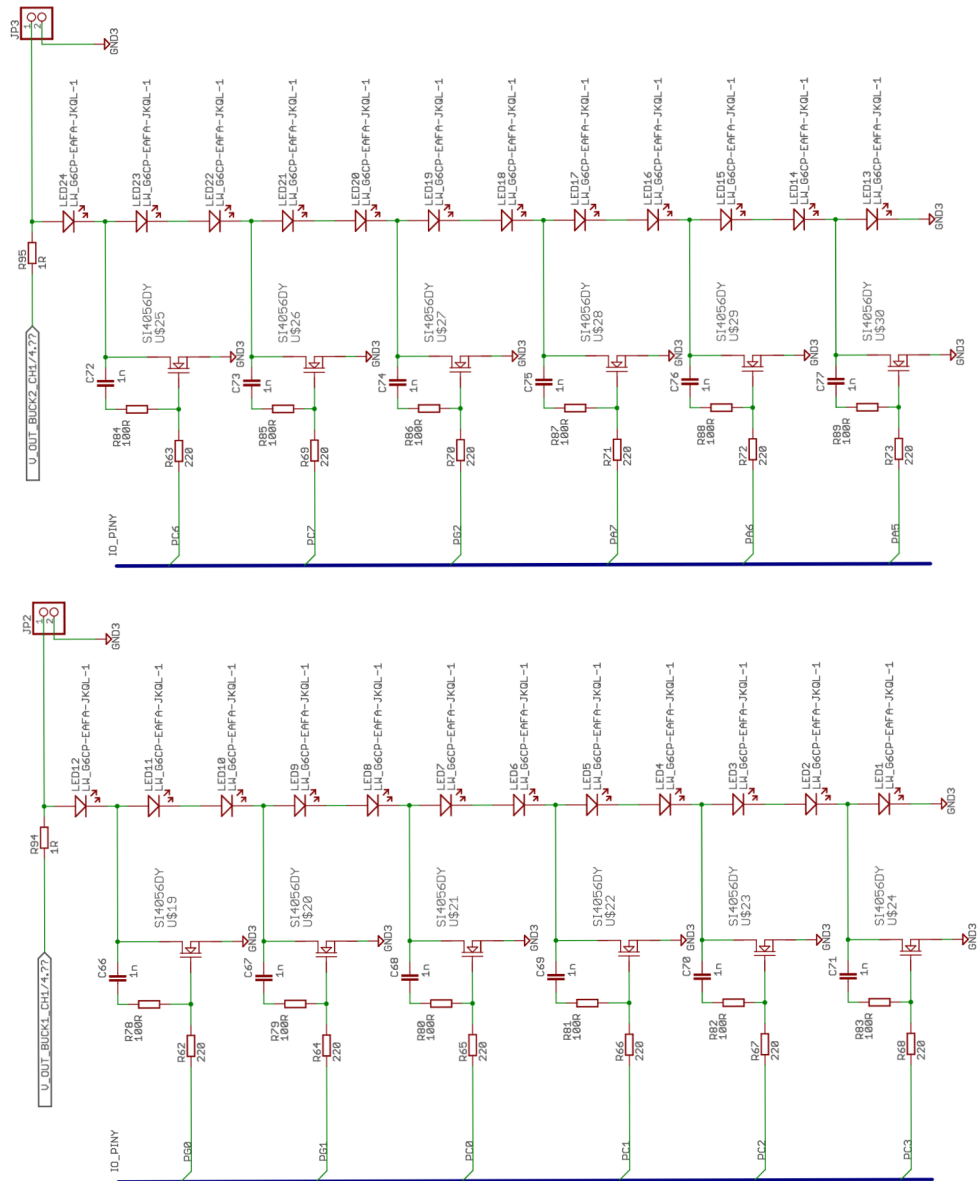
Obr. B.4: Schéma zapojení druhé ze dvou snižujících částí DC-DC měniče

B.4 Řídící mikrokontroler

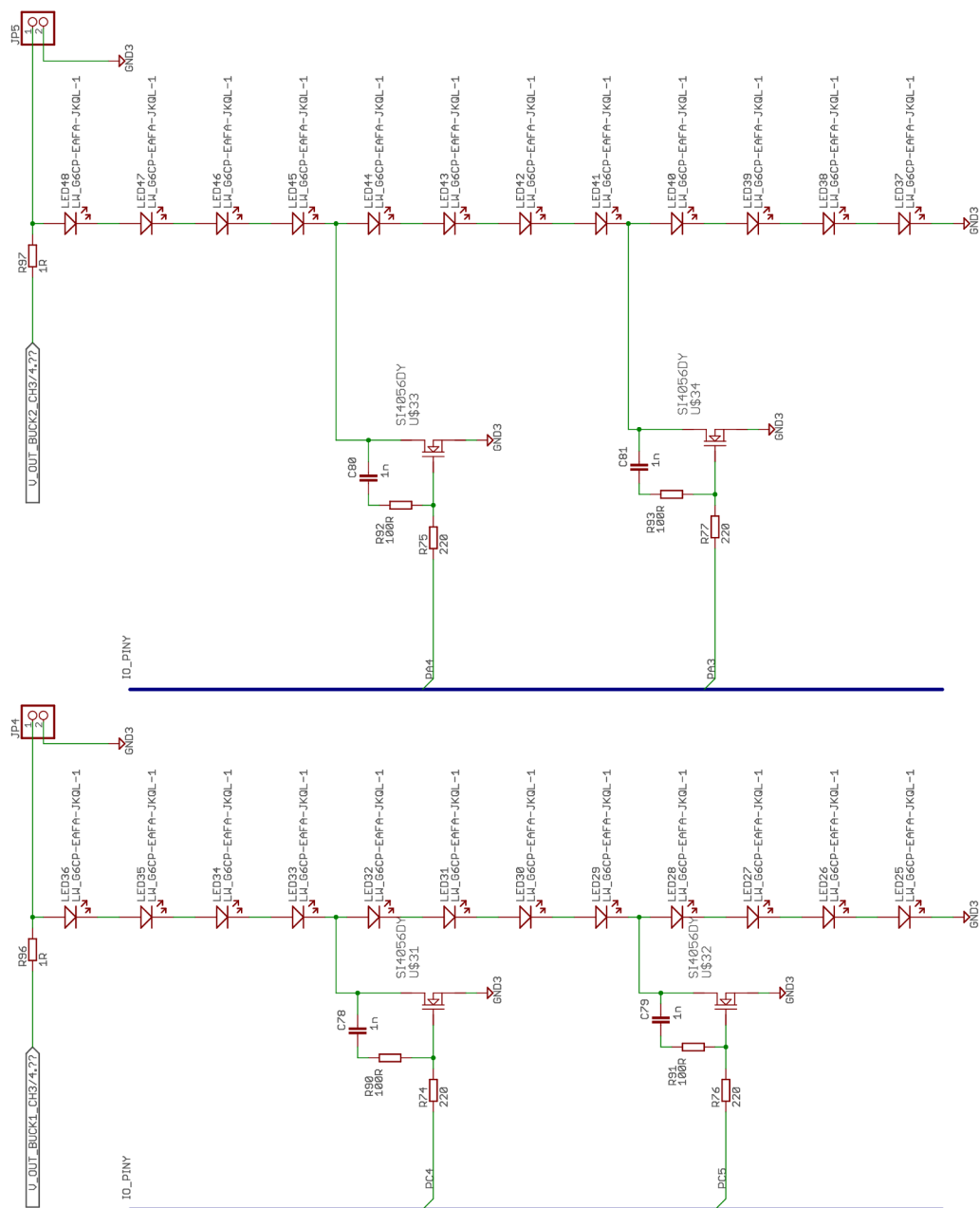


Obr. B.5: Schéma zapojení řídicího mikrokontroleru ATmega128

B.5 Zátěž



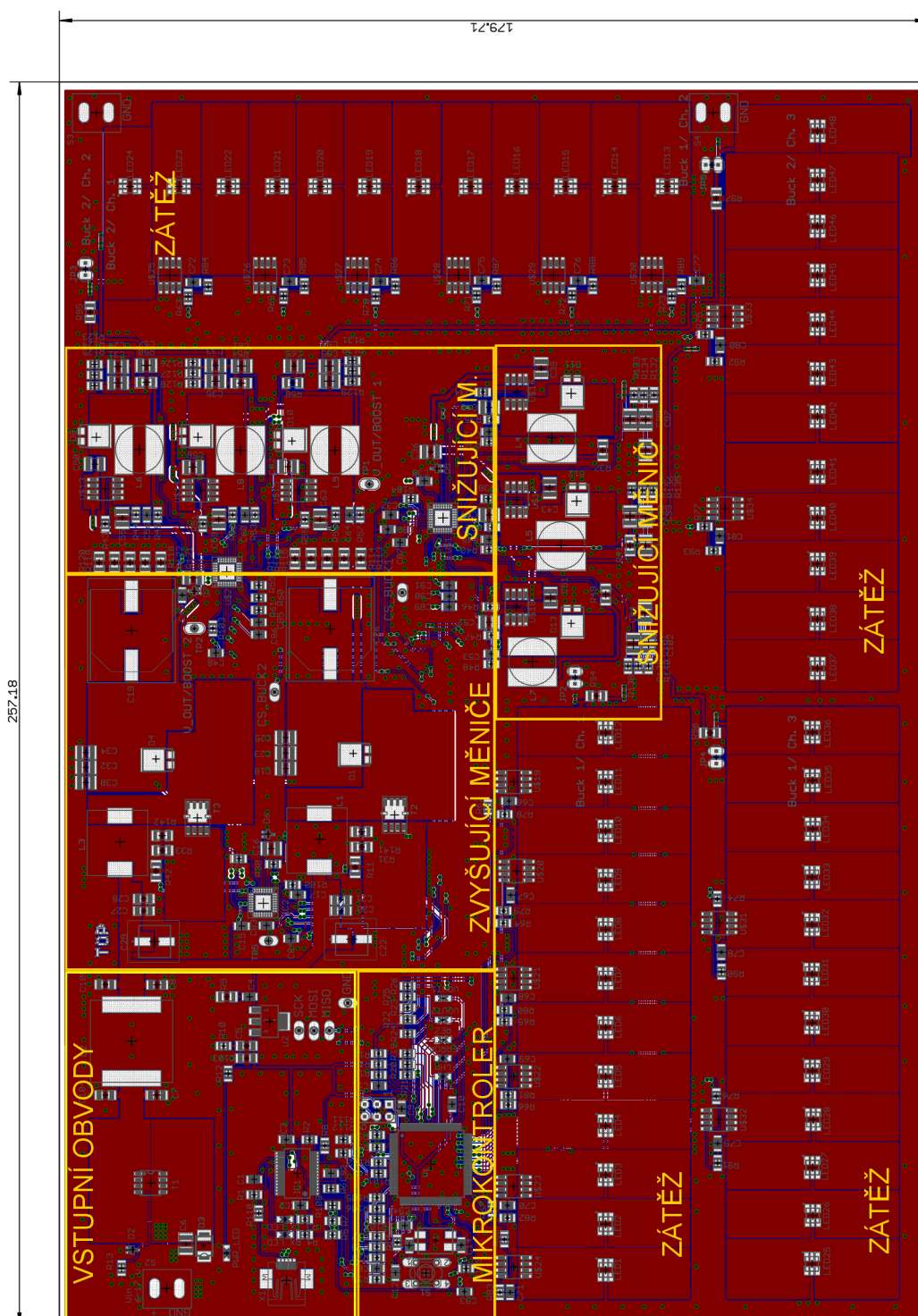
Obr. B.6: Schéma zátěže pro první výstupní kanál obou snižujících měničů



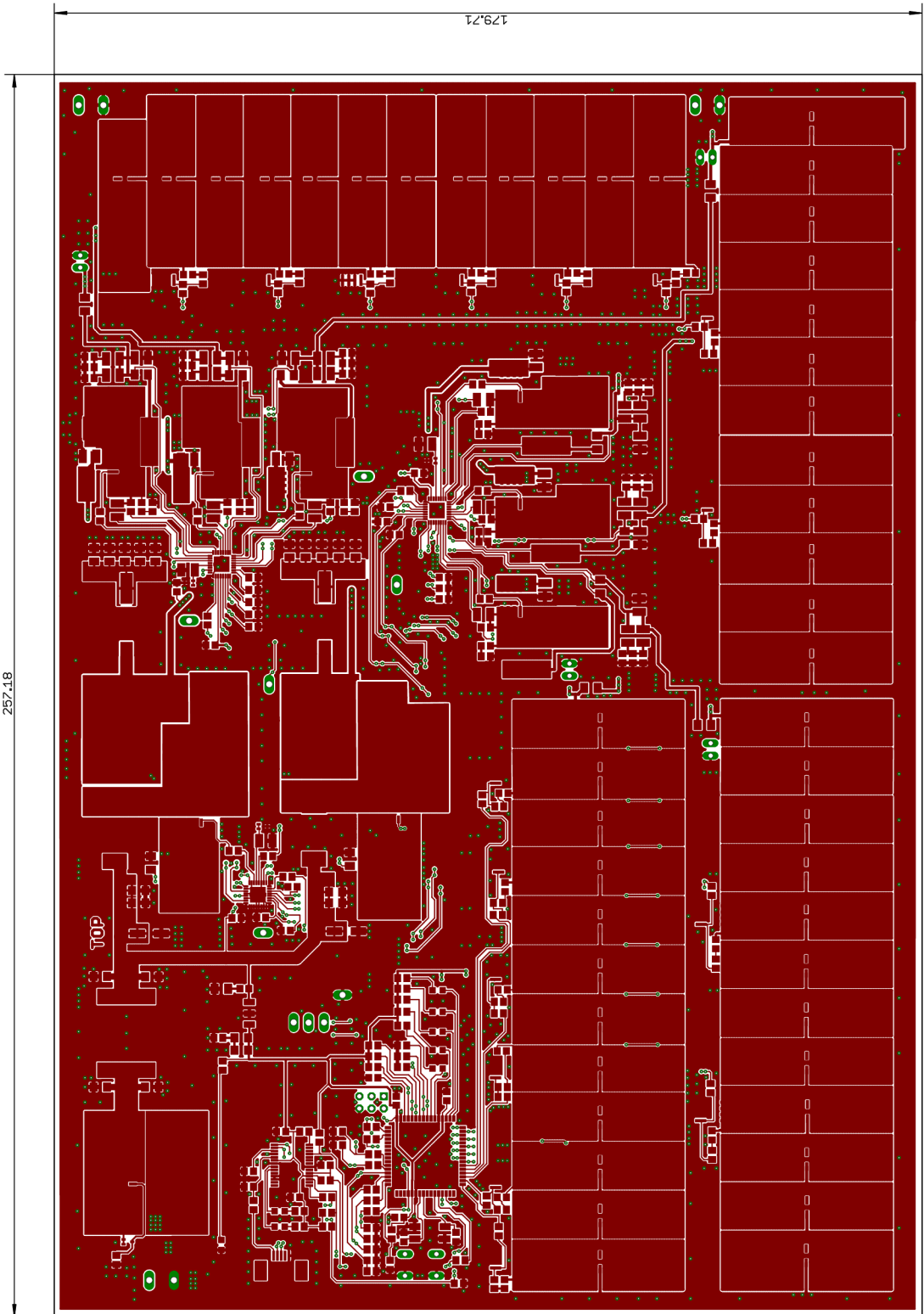
Obr. B.7: Schéma zátěže pro druhý výstupní kanál obou snižujících měničů

C DPS

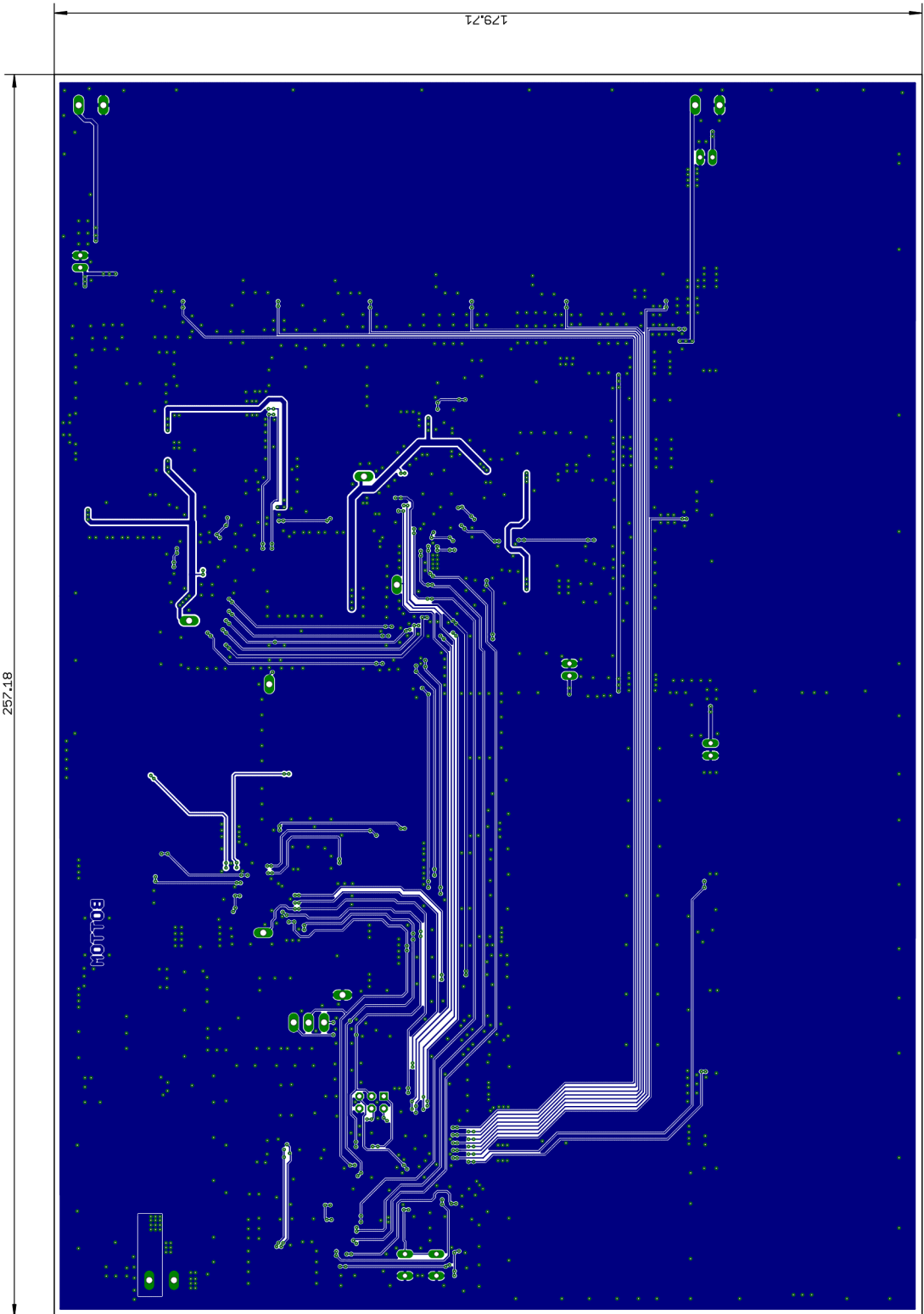
V této kapitole jsou zobrazeny různé zobrazení DPS (Deska Plošného Spoje).



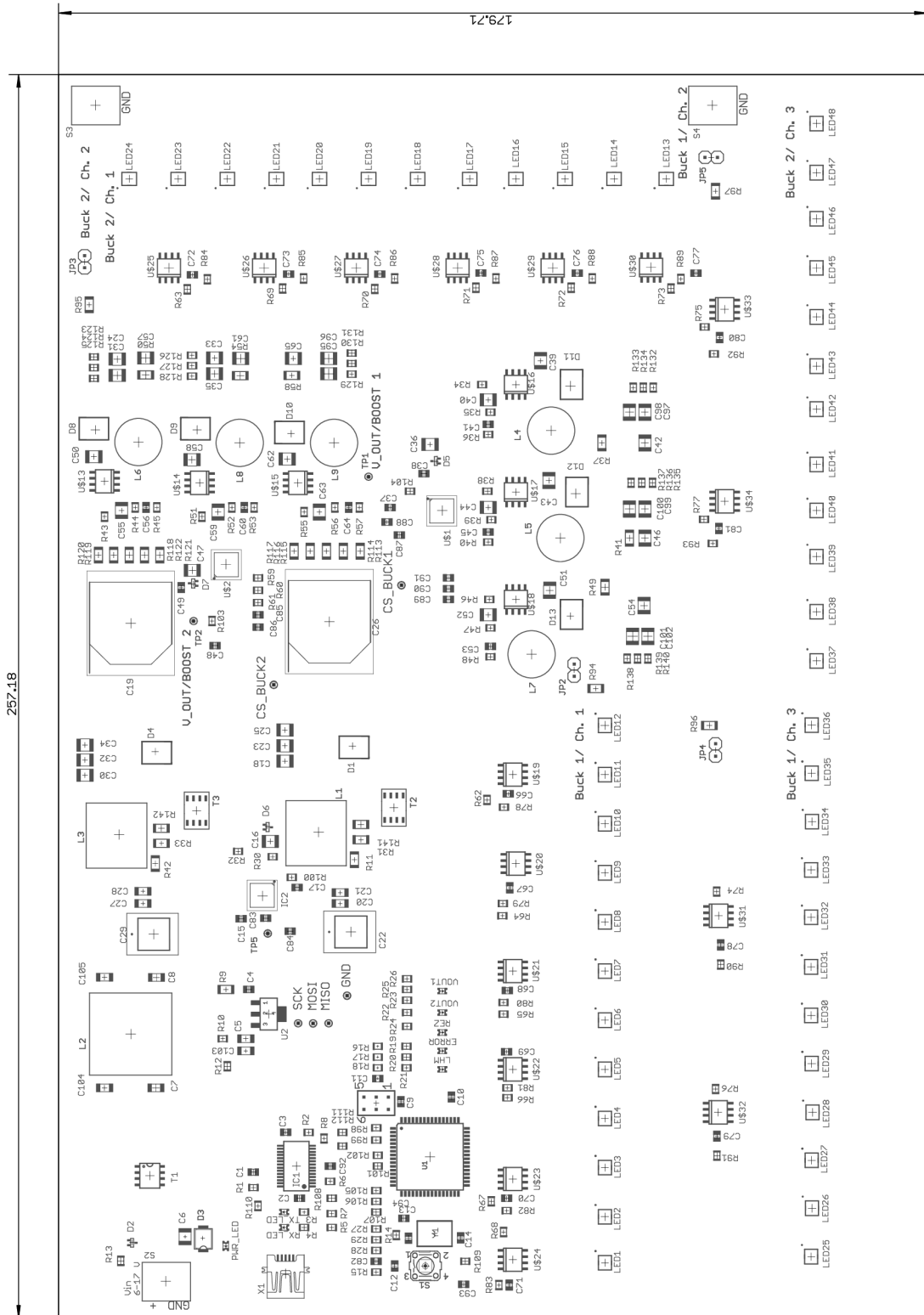
Obr. C.1: DPS prototypu s rozdělením na jednotlivé části



Obr. C.2: DPS vrchní vrstva mědi



Obr. C.3: DPS spodní vrstva mědi



Obr. C.4: DPS rozložení součástek

D KRITICKÉ ÚSEKY PROGRAMU MIKROKONTROLERU ATMEGA128

V této příloze se nachází podstatné částí SW pro mikrokontroler ATmega128 osazený na prototypové desce.

Funkce pro příjem příkazu z rozhraní UART

```
int8_t ReceiveUartCommand(TPRIKAZ *prikaz)
{
    static uint16_t prijem_UART;
    static uint8_t rx[3], i;

    rx[2]='\0';
    prijem_UART = UART_NO_DATA;
    /*funkce uart1_getc() je necekajici, pokud nejsou k
    dispozici data vraci UART_NO_DATA=0x0100*/
    prijem_UART = uart1_getc();

    if (prijem_UART < UART_NO_DATA)
    {
        /*Na zacatku komunikace odesila GUI mikrokontroleru symbol
        ACK (0x6), tim se overi ze komunikace je mozna,
        mikrokontroler odpovi stejnym znakem*/
        if (prijem_UART == ACK)
        {
            uart1_putc(ACK);
            return 1;
        }else if (prijem_UART == '#')
        {
            /******
            /* prikaz je ve formatu #X!YY&ZZ=DD$          */
            /* X - 'R' nebo 'W' - cteni nebo zapis
            Y - TYP DAT - data pro boost, buck, PWM, LHM atd..
            Z - adresa registru v pripade boost a buck typu, jinak
            pomocna data
            D - data */
            /*Nasleduje rozdeleni prikazu a prevod do dekadicke podoby
            ze znaku*/
            /******
            prijem_UART = UART_NO_DATA;

            while(prijem_UART != '!'){
                prijem_UART=uart1_getc();
                if (prijem_UART < UART_NO_DATA && prijem_UART != '!')
```

```

{
    prikaz->RW=prijem_UART-'0';
}
}

prijem_UART = UART_NO_DATA;

i=0;
while(prijem_UART != '&'){
    prijem_UART=uart1_getc();
    if (prijem_UART < UART_NO_DATA && prijem_UART != '&')
    {
        rx[i]=prijem_UART;
        i++;
    }
}
prikaz->TYPE = HEX_ARRAY2DEC(rx);
prijem_UART = UART_NO_DATA;

i=0;
while(prijem_UART != '='){
    prijem_UART=uart1_getc();
    if (prijem_UART < UART_NO_DATA && prijem_UART != '=')
    {
        rx[i]=prijem_UART;
        i++;
    }
}
prikaz->REG_ADDRESS=HEX_ARRAY2DEC(rx);
prijem_UART = UART_NO_DATA;
i=0;
while(prijem_UART != '$'){
    prijem_UART=uart1_getc();
    if (prijem_UART < UART_NO_DATA && prijem_UART != '$')
    {
        rx[i]=prijem_UART;
        i++;
    }
}
prikaz->VALUE=HEX_ARRAY2DEC(rx);
prijem_UART = UART_NO_DATA;

return 0;
}else{
    return -1;
}
}else{

```

```

    return -1;
}
}

```

Funkce pro odeslání a příjem dat z SPI sběrnice

```

int8_t SPI_tx_rx(uint8_t type, uint8_t acces, uint8_t adres,
                uint8_t value, uint8_t* returned_value)
{
    static uint8_t SPI_byte1, SPI_byte2;
    SPI_byte1=0;
    SPI_byte2=0;
    /*access je bud zapis ('W') nebo cteni ('R')*/
    if (acces == 'R' || acces == 'r')
    {
        SPI_byte1 = (1 << 7); //data jsou read (prvni bit adresy)
    }
    else if (acces == 'W' || acces == 'w')
    {
        SPI_byte1 = 0; //data jsou write (prvni bit adresy)
    }
    else
    {
        return -1;
    }

    SPI_byte1 |= adres & (0x7F); //adresa se vymaskuje
    SPI_byte2 = value;

    /*funkce CS_L(type) zajisti nizkou uroven na
    pozadovanem pinu procesoru */
    /*type - podle definovanych symbolu BOOST, BUCK1
    nebo BUCK2*/
    CS_L(type);

    _delay_us(1);

    /*Zapisek do registru SPDR zahajeni prenosu*/
    SPDR = SPI_byte1;
    /*odeslana data jsou signalizovana v registru SPSR*/
    while(!(SPSR & (1<<SPIF)));
    *returned_value = SPDR;

    SPDR = SPI_byte2;
    while(!(SPSR & (1<<SPIF)));
    *returned_value = SPDR;
}

```

```

_delay_us(1);

/*funkce CS_H(type) zajisti vysokou uroven na
pozadovanem pinu procesoru */
/*type - podle definovanych symbolu BOOST, BUCK1
nebo BUCK2*/
CS_H(type);

return 0;
}

```

Funkce pro práci se zátěží

```

uint8_t LoadFETControl(uint8_t acces, TPRIKAZ prikaz)
{
    switch (acces)
    {
        case 'R':
            switch(prikaz.REG_ADDRESS)
            {
                case T_BUCK1_CH1:
                    /*Kazdy bit v navratove hodnote symbolizuje jeden*/
                    /*tranzistor v dane zatezi*/
                    return ((PORTC >> PC3) & 0b1) |
                        (((PORTC >> PC2) & 0b1) <<1) |
                        (((PORTC >> PC1) & 0b1) << 2) |
                        (((PORTC >> PC0) & 0b1) << 3) |
                        (((PORTG >> PG1) & 0b1) << 4) |
                        (((PORTG >> PG0) & 0b1) << 5);

                    break;

                case T_BUCK1_CH3:
                    return ((PORTC >> PC5) & 0b1) |
                        (((PORTC >> PC4) & 0b1) <<1);

                    break;

                .
                .
                .
            }
            break;
        case 'W':
            switch(prikaz.REG_ADDRESS)
            {
                case T_BUCK1_CH1:
                    /*Kazdy bit v navratove hodnote symbolizuje jeden*/
                    /*tranzistor v dane zatezi*/
                    PORTC &= ~( _BV(PC3) | _BV(PC2) | _BV(PC1) | _BV(PC0));

```

```

    PORTG &= ~(_BV(PG1) | _BV(PG0));

    PORTC |= ((prikaz.VALUE & 0b1) << PC3) |
              (((prikaz.VALUE >> 1) & 0b1) << PC2) |
              (((prikaz.VALUE >> 2) & 0b1) << PC1) |
              (((prikaz.VALUE >> 3) & 0b1) << PC0);

    PORTG |= (((prikaz.VALUE >> 4) & 0b1) << PG1) |
              (((prikaz.VALUE >> 5) & 0b1) << PG0);
    return prikaz.VALUE;
break;

case T_BUCK1_CH3:
    PORTC &= ~(_BV(PC5) | _BV(PC4));
    PORTC |= (((prikaz.VALUE & 0b1) << PC5) |
              (((prikaz.VALUE >> 1) & 0b1) << PC4));
    return prikaz.VALUE;
break;
.
.
.
}
break;
default:
    PORTC &= ~(_BV(PC3) | _BV(PC2) | _BV(PC1) | _BV(PC0));
    PORTG &= ~(_BV(PG1) | _BV(PG0));
    PORTC &= ~(_BV(PC5) | _BV(PC4));
    PORTA &= ~(_BV(PA3) | _BV(PA4) | _BV(PA5) |
              _BV(PA6) | _BV(PA7));

    PORTG &= ~_BV(PG2);
    PORTC &= ~(_BV(PC7) | _BV(PC6));
    return 0;
break;
}
}

```

Funkce pro vyhodnocení diagnostických informací

```

void TASK_evaluate_diagnostic_registers(void)
{
    /* ***** */
    /* Vyhodnoceni diagnostiky menicu */
    /* ***** */
    static uint16_t delay=0;
    static TDIAGNOSTIC diagnostic_reg;

    if(HasDelayElapsed(100, &delay, true))

```

```

{
    //cteni registru, ulozeni stavu do struktury
    SPI_read_diagnostic(&diagnostic_reg);
    EvaluationDiagReg(&diagnostic_reg);
    /***/
    /* EvaluationDiagReg() vyhodnoti diagnosticke registry, */
    /*v pripade chyby v diagnostickem registru ktorehokoliv */
    /*cipu je aktivovana LED dioda na pinu PF7, v pripade */
    /*aktivace LHM modu je aktivovana LED dioda na pinu PF6,*/
    /*v pripade dosazeni vystupniho napeti boostu je */
    /*aktivovana LED dioda na pinu PA2 pro Vout1 a PA1 pro Vout2*/
    /***/
}

return;
}

```

E KRITICKÉ ÚSEKY PROGRAMU UŽIVATELSKÉHO ROZHŘANÍ

V této příloze se nachází podstatné části SW uživatelského rozhraní pro ovládání DC-DC měniče.

Funkce pro odeslání a příjem příkazů po rozhraní UART

```
def uartTX_RX(self, NUM_OF_TRY, RW, TYPE, ADRESS, VALUE):
    rx_ack = 0
    #vola se funkce ktera ze vstupnich parametru vytvori string pro
    #odeslani

    tx_uart_command = self.makeUartMessage(RW, TYPE, ADRESS, VALUE)
    rx_uart_command=[]

    if RW == 'r' or RW == 'R': #cteni registru z DC menice
        self.flushBuffers() #vycistení IO bufferu
        self.write(ACK) #odeslani ACK do menice
        rx_ack = self.read(1) #funkce read ceka po dobu TIMEOUT
            #na odpoved
        self.flushBuffers() #vycistení IO bufferu
        if rx_ack == ACK: #pokud uC odpovedel ACK
            for i in range(NUM_OF_TRY):# prikaz se zkusi poslat
                # az 3 krat, nebo dokud neprijde spravna odpoved
                rx_uart_command = []
                self.write(tx_uart_command) #odeslani prikazu
                rx_uart_command = self.uartReceiveOneMessage() #prijem prikazu
                self.flushBuffers()#vycistení IO bufferu
                if type(rx_uart_command) is list: #pokud UART nefunguje,
                    #funkce vraci -1, to neni datovy typ list, osetreni
                    #proti chybe
                    if (rx_uart_command[0] == RW) and
                        (rx_uart_command[1] == TYPE) and
                        (rx_uart_command[2] == ADRESS):
                        #kontrola zda nedoslo k precteni spatneho registru
                        return rx_uart_command
            return -1
        else:
            self.flushBuffers()
            return -1

    #zapis do prototypu DCDC
    elif RW == 'w' or RW == 'W':
        self.flushBuffers()
```

```

self.write(ACK) #kontrola spojeni ACK
rx_uart_command = self.read(1)
self.flushBuffers()
if rx_uart_command == ACK:
    for i in range(NUM_OF_TRY): #pokus zapsat az 3 krat
        rx_uart_command = []
        self.write(tx_uart_command)
        rx_uart_command = self._uartReceiveOneMessage()
        self.flushBuffers()
        if (rx_uart_command[0] == RW) and (rx_uart_command[1] == TYPE)
            and(rx_uart_command[2] == ADDRESS) and
                (rx_uart_command[3] == VALUE):
            #rozdil oproti variante read, ze realne prectena
            #hodnota v registru se musi schodovat s tim co posilame
            return rx_uart_command
    return -1
else:
    self.flushBuffers()
    return -1
else:
    return -1

```

Výpis definovaných konstant v uživatelském rozhraní

```

# types
self.BOOST = '00'
self.BOOST_NVM_37 = '0F'
self.BOOST_NVM_38 = '1F'
self.BUCK1 = '10'
self.BUCK2 = '20'
self.BUCK1_NVM_19 = '2F'
self.BUCK1_NVM_18 = '3F'
self.BUCK2_NVM_19 = '4F'
self.BUCK2_NVM_18 = '5F'
self.ENABLE_PINS = '30'
self.TRANSISTORS = '40'
self.LHM = '50'
self.PWM = '60'

```

Úsek kódu realizující zápis do souboru

```

def saveJSONtofile(self, type_of_data):
    self.filename = filedialog.asksaveasfilename(initialdir="C:/
    \PyCharmWORK/DCDC_CONVERTER_GUI2/Soubory/", title="Select file",
    filetypes=(( 'All Files', '*.*' ),))
    # vychazi cesta a typy souboru, FUNKCE VRACI CESTU K SOUBORU

```



```

if self.filename != '': #ošetření, pokud je název souboru prázdný
    if not('.txt' in self.filename): #pokud obsluha nenapíše .txt
        self.filename=self.filename + '.txt'

with open(self.filename, 'w') as fp:
    self.addData2dictionary() #přidání dat do slovníku
    self.pythonDict2JSON() #převod slovníku pythonu do JSON
    json.dump(json.dumps(type_of_data),fp) #uložení TYPU do souboru,
        #pro rozlišení jednotlivých souborů

    fp.write('\n')
    json.dump(self.JSON_dict, fp)#uložení slovníku do souboru
    fp.close()

```

Úsek funkce pro načítání dat ze slovníku Pythonu

```

def readDatafromDictionary(self, type_of_data):
    pole_hodnot=self.python_dict['values']
    start_index=0

    if type_of_data == 'all':
        self.list_of_types = self.python_dict['types']
        self.list_of_adress = self.python_dict['adress']
        self.list_of_values = self.python_dict['values']
    else:
        if type_of_data == self.BOOST: #boost
            start_index = self.python_dict['types'].index(self.BOOST)
                #start_index je pozice 1. vyskytu self.BOOST
            for i in range(start_index, len(self.list_of_values)):
                if self.list_of_types[i] != self.BOOST:
                    break
                self.list_of_values[i] = pole_hodnot[i]

        elif type_of_data == self.BUCK1: #buck1
            start_index = self.python_dict['types'].index(self.BUCK1)
            for i in range(start_index, len(self.list_of_values)):
                if self.list_of_types[i] != self.BUCK1:
                    break
                self.list_of_values[i] = pole_hodnot[i]

        .
        .
        .
    return

```

F KONFIGURACE REGISTRŮ MĚNIČŮ PŘI MĚŘENÍ

Tab. F.1: Konfigurace registrů zvyšujícího měniče při měření - část 1

| Název | Adresa | Hodnota | Poznámka |
|--|--------|---------|--|
| Spread spectrum enable | 0x1C | 0x00 | rozprostřené spektrum zakázáno |
| Overvoltage treshold | 0x1A | 0x33 | přepětová ochrana 18,39 V |
| Undervoltage treshold | 0x1B | 0x10 | podpětová ochrana 5,77 V |
| VGG control | 0x15 | 0xA7 | napětí ovládání tranzistoru 5,52 V |
| Current sense slope config | 0x14 | 0xAA | nastavení zpětné vazby podle doporučení výrobce |
| Slope compensation config | 0x13 | 0x44 | nastavení zpětné vazby podle doporučení výrobce |
| Loop filter integral config | 0x12 | 0x00 | nastavení zpětné vazby podle doporučení výrobce |
| Loop filter proportional config | 0x11 | 0x5F | nastavení zpětné vazby podle doporučení výrobce |
| Phase select config | 0x10 | 0x08 | výstup 2 je ovládán 3. virtuální fází, výstup 1 první fází |
| Gate driver phase | 0x0F | 0x02 | výstup 2 je připojen k interní logice 2, výstup 1 k logice 1 |
| Phase-off time and phase delay of output 2 | 0x0E | 0x78 | nastavení časování spínání výstupu 2 |
| Phase-off time and phase delay of output 1 | 0x0D | 0x78 | nastavení časování spínání výstupu 1 |
| SS scenario logic 2 | 0x0C | 0x0C | nastavení rozprostřeného spektra + povolení virtuálních fází 2. interní logiky |
| SS scenario logic 1 | 0x0B | 0x03 | nastavení rozprostřeného spektra + povolení virtuálních fází 1. interní logiky |

Tab. F.2: Konfigurace registrů zvyšujícího měniče při měření - část 2

| Název | Adresa | Hodnota | Poznámka |
|-------------------------------|--------|---------|--|
| Clock divider for output 2 | 0x0A | 0x27 | přímá dělička 40 hodinového kmitočtu |
| Clock divider for output 1 | 0x09 | 0x27 | přímá dělička 40 hodinového kmitočtu |
| Maximum coil current output 2 | 0x08 | 0x2A | maximální proud (5,53 A) cívkou na výstupu 2 |
| Maximum coil current output 1 | 0x07 | 0x2A | maximální proud (5,53 A) cívkou na výstupu 1 |
| Limit voltage output 2 | 0x06 | 0xD5 | výstupní přepětová ochrana kanál 2 (76,8 V) |
| Limit voltage output 1 | 0x05 | 0xD5 | výstupní přepětová ochrana kanál 1 (76,8 V) |
| Output voltage 2 | 0x04 | 0xD0 | Výstupní napětí 75 V |
| Output voltage 1 | 0x03 | 0xD0 | Výstupní napětí 75 V |
| Gate driver output | 0x02 | 0x02 | Přiřazení výstupu k vnitřní výstupní logice |
| Gate driver enable | 0x01 | 0x02 | Povolení výstupní logiky |
| Function control register | 0x00 | 0x47 | Hlavní řídicí registr, povolení výstupních napětí, signalizace, že je konfigurace hotová, povolení měření vstupního napětí |

| NVM address linked to SPI register 36h[4:0] | NVM_DATA[15:8] linked to SPI register 37h and 59h | NVM_DATA[7:0] linked to SPI register 38h and 5Ah |
|---|---|--|
| 00h | n.a. | multi-content NVM data fields |
| 01h | NVM write count bit 15 to bit 8 | NVM write count bit 7 to bit 0 |
| 02h | n.a. | function control |
| 03h | n.a. | gate driver enable |
| 04h | gate driver output | output voltage 1 |
| 05h | output voltage 2 | limit voltage output 1 |
| 06h | limit voltage output 2 | maximum coil current V_{O1} |
| 07h | maximum coil current V_{O2} | clock divider for V_{O1} |
| 08h | clock divider for V_{O2} | SS scenario logic 1 |
| 09h | SS scenario logic 2 | phase-off time and phase delay of output 1 |
| 0Ah | phase-off time and phase delay of output 2 | gate driver phase |
| 0Bh | phase select configuration | loop filter proportional configuration |
| 0Ch | loop filter integral configuration | slope compensation configuration |
| 0Dh | current sense slope resistor configuration | VGG control |
| 0Eh | internal 1 | overvoltage threshold |
| 0Fh | undervoltage threshold | spread spectrum enable |
| 10h | internal 2 | internal 3 |
| 11h | internal 4 | internal 5 |
| 12h | ramp up output 1 | ramp up output 2 |

Obr. F.1: Rozložení NVM paměti zvyšujícího měniče [8]

| NVM address linked to SPI register 0x17h [4:0] | NVM data linked to SPI register 0x19h [15:8] ^[9] | NVM data linked to SPI register 0x18h [7:0] ^[9] |
|--|---|---|
| 0x10h | LED current channel 2 (SPI register address 0x03h) default: 0x00h | LED current channel 1 (SPI Register Address 0x02h) default: 0x00h |
| 0x11h | VGG control (SPI register address 0x01h), default: 0x96h | LED current channel 3 (SPI Register Address 0x04h) default: 0x00h |
| 0x12h | NVM_Hyst, default: 0x00h (see Table 32) | Under voltage threshold (SPI Register Address 0x0Fh) default: 0x00h |
| 0x13h | - | NVM_PWM_ctrl, default: 0x2Ah (see Table 32) |

Obr. F.2: Rozložení NVM paměti snižujícího měniče [9]

Tab. F.3: Konfigurace NVM paměti zvyšujícího měniče

| Adresa v NVM | Hodnota[15:8] | Poznámka k [15:8] | Hodnota[7:0] | Poznámka k [7:0] |
|--------------|---------------|--------------------------------------|--------------|--------------------------------------|
| 0x00 | 0x00 | n.a. | 0x11 | speciální registr povolující LHM mód |
| 0x01 | 0x00 | MSB čítače zápisů do NVM | 0xB5 | LSB čítače zápisu do NVM |
| 0x02 | 0x00 | n.a. | 0x47 | POZN |
| 0x03 | 0x00 | n.a. | 0x03 | POZN |
| 0x04 | 0x02 | POZN | 0x8B | POZN, (50,12 V) |
| 0x05 | 0x8B | POZN, (50,12 V) | 0x8D | POZN, (50,84 V) |
| 0x06 | 0x8D | POZN, (50,84 V) | 0x2A | POZN |
| 0x07 | 0x2A | POZN | 0x27 | POZN |
| 0x08 | 0x27 | POZN | 0x03 | POZN |
| 0x09 | 0x0C | POZN | 0x7A | POZN |
| 0x0A | 0x7A | POZN | 0x02 | POZN |
| 0x0B | 0x08 | POZN | 0x5F | POZN |
| 0x0C | 0x00 | POZN | 0x44 | POZN |
| 0x0D | 0xAA | POZN | 0xAC | POZN, (5,19 V) |
| 0x0E | 0x82 | internal 1 (NEMĚNIT) | 0x31 | POZN |
| 0x0F | 0x0F | POZN | 0x30 | POZN |
| 0x10 | 0x27 | internal 2 (NEMĚNIT) | 0x3B | internal 3 (NEMĚNIT) |
| 0x11 | 0xE8 | internal 4 (NEMĚNIT) | 0x09 | internal 5 (NEMĚNIT) |
| 0x12 | 0x00 | nastavení náběhu výstupního napětí 1 | 0x00 | nastavení náběhu výstupního napětí 2 |

POZN = význam odpovídá stejnojmennému registru v tabulkách F.1 a F.2

Tab. F.4: Konfigurace registrů snižujícího měniče při měření

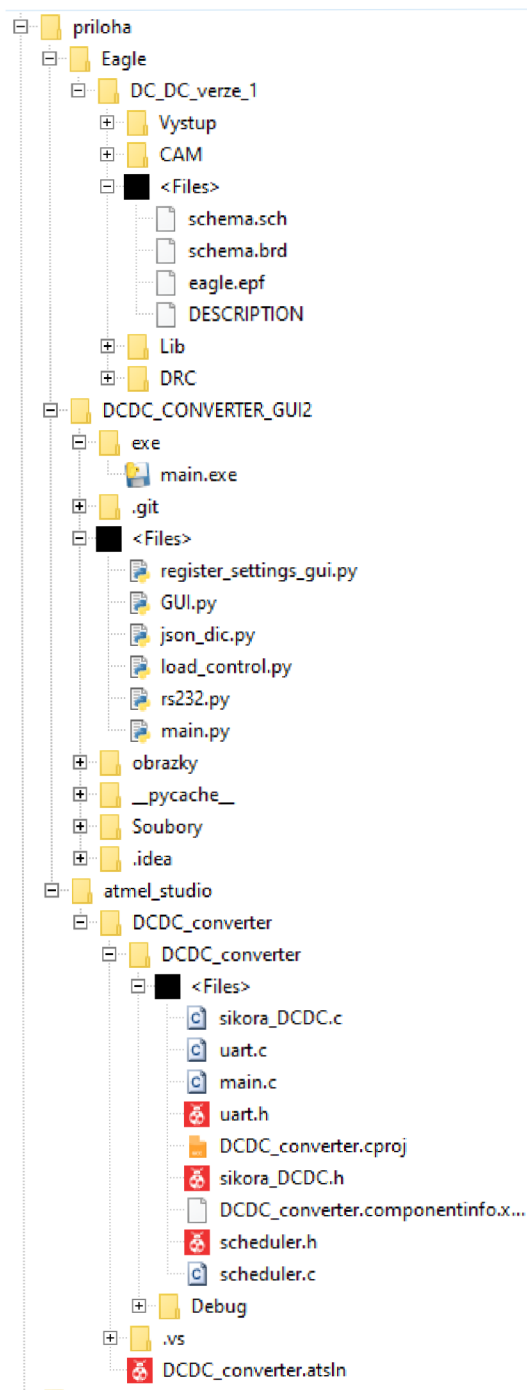
| Název | Adresa | Hodnota | Poznámka |
|---------------------------|--------|---------|---|
| Function control register | 0x00 | 0xCE | Hlavní řídicí registr, povoluje výstupní kanály |
| Undervoltage treshold | 0x0F | 0xAB | Práh podpětí na vstupu měniče (60,11 V) |
| VGG control | 0x01 | 0xA4 | Napětí spínání externích tranzistorů (5,64 V) |
| Hysteresis channel 3 | 0x0D | 0x01 | nastavení 1 (cca 20 mA) |
| Hysteresis channel 2 | 0x0C | 0x01 | nastavení 1 (cca 20 mA) |
| Hysteresis channel 1 | 0x0B | 0x01 | nastavení 1 (cca 20 mA) |
| LED current range | 0x05 | 0x07 | maximální napětí na snímá-cím rezistoru je 120 mV |
| LED current channel 3 | 0x04 | 0xFE | proud výstupním kanálem 3 (145 mA) |
| LED current channel 2 | 0x03 | 0xFE | proud výstupním kanálem 2 (145 mA) |
| LED current channel 1 | 0x02 | 0xFE | proud výstupním kanálem 1 (145 mA) |

Tab. F.5: Konfigurace NVM paměti snižujících měničů

| Adresa v NVM | Hodnota[15:8] | Poznámka k [15:8] | Hodnota[7:0] | Poznámka k [7:0] |
|--|---------------|--|--------------|--|
| 0x10 | 0x01 | POZN | 0x7B | POZN |
| 0x11 | 0xA4 | POZN | 0x7B | POZN |
| 0x12 | 0xD5 | speciální registr (hystereze + rozsah) | 0x73 | POZN |
| 0x13 | 0x00 | n.a. | 0x43 | speciální registr (rozsah + nastavení PWM) |
| POZN = význam odpovídá stejnojmennému registru v tabulce F.4 | | | | |

G PŘILOŽENÉ SOUBORY

V této příloze se nachází stromový diagram příloženého CD.



Obr. G.1: Stromový diagram obsahu příloženého CD