

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA ELEKTROMECHANIKY A VÝKONOVÉ ELEKTRONIKY**

## **BAKALÁŘSKÁ PRÁCE**

**Implementace řízení tyristorového usměrňovače do  
TMS320F28335**

ZÁPADOČESKÁ UNIVERZITA V PLZNI  
Fakulta elektrotechnická  
Akademický rok: 2014/2015

**ZADÁNÍ BAKALÁŘSKÉ PRÁCE**  
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Antonín GLAC**  
Osobní číslo: **E12B0020P**  
Studijní program: **B2612 Elektrotechnika a informatika**  
Studijní obor: **Elektrotechnika a energetika**  
Název tématu: **Implementace řízení tyristorového usměrňovače do TMS320F28335**  
Zadávající katedra: **Katedra elektromechaniky a výkonové elektroniky**

**Z á s a d y p r o v y p r a c o v á n í :**

Cílem práce je implementovat řízení tyristorového usměrňovače do číslicového regulátoru MLC interface, který je osazen mikrokontrolérem TMS320F28335. Student by měl v rámci práce navrhnout řízení tyristorového usměrňovače v otevřené smyčce.

1. Zpracujte teorii řízení 3f tyristorového usměrňovače.
2. Navrhněte moduly detekce průchodu nulou a budiče tyristorů, oboje s galvanickým oddělením. Tyto moduly koncipujte tak, aby šly bez úprav připojit k stávajícímu číslicovému regulátoru.
3. Implementujte řízení tyristorového usměrňovače v otevřené smyčce do TMS320F28335.
4. Prakticky ověřte funkčnost celé sestavy na vybrané aplikaci.

Rozsah grafických prací: podle doporučení vedoucího  
Rozsah pracovní zprávy: 20 - 30 stran  
Forma zpracování bakalářské práce: tištěná/elektronická  
Seznam odborné literatury:

1. Vondrášek, F., Výkonová elektronika, svazek 2, Měniče s vnější komutací. Západočeská univerzita v Plzni, 2001

Vedoucí bakalářské práce: Ing. Tomáš Košan  
Regionální inovační centrum elektrotechniky

Datum zadání bakalářské práce: 15. října 2014  
Termín odevzdání bakalářské práce: 8. června 2015

  
Doc. Ing. Jiří Hammerbauer, Ph.D.  
děkan



  
Prof. Ing. Václav Kůs, CSc.  
vedoucí katedry

V Plzni dne 15. října 2014

## **Abstrakt**

Cílem této práce je navrhnout tyristorový usměrňovač a implementovat jeho řízení do číslicového regulátoru MLC interface, který je osazen mikrokontrolérem TMS320F28335. Text práce obsahuje teorii řízení tyristorů a můstkového třífázového usměrňovače. V další části jsou popsány jednotlivé funkční bloky usměrňovače – především budiče tyristorů a detektory průchodu nulou. Obsažen je i základní popis programu, postup práce a kapitola o měření včetně oscilogramů.

DPS modulu usměrňovače byla navržena v programu Altium Designer. Výkresová dokumentace je obsažena v přílohách. Po návrhu následovala výroba (externí firma), osazení a praktické testy. Řízení zajišťuje mikrokontrolér, základem programu je čítač, který je synchronizován s napájecí sítí.

Praktické testy potvrdily funkčnost celého usměrňovače. Program bude možné dále rozšiřovat, např. o regulátor proudu. Výstupem je funkční prototyp usměrňovače (napájení ze sítě 3x400V, maximální proud 10A), který bude umístěn v laboratořích KEV a sloužit pro buzení synchronního stroje.

## **Klíčová slova**

Tyristor, třífázový můstkový usměrňovač, detektor průchodu nulou, pulzní transformátor, DPS, mikrokontrolér, DSP, budič, řídicí úhel

## **Abstract**

The aims of this thesis are design a thyristor rectifier and implement control of rectifier to MLC Interface with DSP TMS320F28335. Theory of SCR and bridge rectifier is described in the first part. In the next parts are described functional blocks of the rectifier, software and measuring. PCB of this rectifier was designed in Altium Designer. The schematic design is included. Driving a rectifier is supported by counter.

The rectifier was tested in a laboratory. It is fully functional. It will be used as a current source for excitation a synchronous machine.

## **Key words**

SCR, three-phase bridge rectifier, zero cross detector, pulse transformer, PCB, microcontroller, DSP, driver, firing angle

## **Prohlášení**

Prohlašuji, že jsem tuto diplomovou/bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské/diplomové práce, je legální.

.....

podpis

V Plzni dne 7.6.2015

Antonín Glac

## **Poděkování**

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Tomáši Košanovi za cenné profesionální rady, připomínky a metodické vedení práce. Děkuji i za poskytnutí potřebného softwaru pro návrh DPS a vyřízení všech potřebných náležitostí pro výrobu desky plošného spoje.

# Obsah

<b>OBSAH .....</b>	<b>8</b>
<b>SEZNAM SYMBOLŮ A ZKRATEK .....</b>	<b>9</b>
<b>ÚVOD .....</b>	<b>10</b>
<b>1 TEORETICKÝ ÚVOD.....</b>	<b>11</b>
1.1 TYRISTOR .....	11
1.2 TŘÍFÁZOVÝ PLNĚ ŘÍZENÝ MŮSTEK.....	13
<b>2 POPIS FUNKČNÍCH BLOKŮ.....</b>	<b>16</b>
2.1 VÝKONOVÝ OBVOD .....	16
2.2 BUDIČE TYRISTORŮ .....	17
2.3 DETEKTORY PRŮCHODU NULOU.....	18
2.4 MĚŘENÍ PROUDU.....	19
<b>3 POPIS PROGRAMU.....</b>	<b>20</b>
3.1 ČÍTAČ .....	20
3.2 SYNCHRONIZACE .....	22
3.3 NASTAVENÍ ŘÍDÍČÍHO ÚHLU.....	22
3.4 SLED FÁZÍ.....	22
<b>4 POSTUP PRÁCE.....</b>	<b>24</b>
4.1 NÁVRH DPS .....	24
4.2 VÝROBA DPS .....	24
4.3 PROGRAMOVÁNÍ.....	24
<b>5 MĚŘENÍ.....</b>	<b>26</b>
5.1 PRŮBĚH MĚŘENÍ .....	26
5.2 OSCIOGRAMY Z MĚŘENÍ.....	28
<b>ZÁVĚR .....</b>	<b>30</b>
<b>SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ .....</b>	<b>31</b>
<b>PŘÍLOHY.....</b>	<b>1</b>



## Seznam symbolů a zkratek

DPS (PCB).....	deska plošného spoje
DSP .....	digitální signálový procesor
$i_D$ .....	blokovací proud tyristoru
$I_G$ .....	zapínací proud tyristoru na gate
$I_H$ .....	vratný proud tyristoru
$I_L$ .....	přidržený proud tyristoru
$i_R$ .....	závěrný proud tyristoru
$m$ .....	počet fází
$p$ .....	pulznost
PWM.....	pulzně šířková modulace
$U_{(B0)}$ .....	maximální blokovací napětí tyristoru
$U_{di}$ .....	střední hodnota usměrněného napětí
$U_{di0}$ .....	střední hodnota usměrněného napětí při $\alpha = 0^\circ$
$U_{diA}$ .....	střední hodnota usměrněného napětí anodové skupiny
$U_{diK}$ .....	střední hodnota usměrněného napětí katodové skupiny
$u_D$ .....	blokovací napětí tyristoru
$u_R$ .....	závěrné napětí tyristoru
$u_T$ .....	napětí na sepnutém tyristoru
$t_q$ .....	doba pro obnovení blokovací schopnosti
$t_v$ .....	doba sepnutí součástky
$\alpha$ .....	řídící úhel

## Úvod

Pro napájení stejnosměrných strojů a buzení synchronních strojů je zapotřebí zdroj stejnosměrného proudu. V minulosti sloužilo nejčastěji jako zdroj dynamo. Dnes je častější realizací zdroje připojení polovodičového usměrňovače ke střídavé napájecí síti. Nejjednodušší variantou je diodový usměrňovač, který ale neumožňuje změnu napětí. Použitím řízeného tyristorového usměrňovače získáme možnost regulovat střední hodnotu výstupního napětí. Nejvýhodnější způsob zapojení na třífázové síti je můstek s 6 součástkami. Má nejnižší zvlnění výstupního proudu a zároveň nevyžaduje speciální transformátor. Alternativou je také pulzní usměrňovač, který je dražší a jeho řízení je složitější. Výhodou je omezení negativního vlivu na síť. Právě vliv na síť (odběr vyšších harmonických) je hlavní nevýhodou diodových i tyristorových usměrňovačů.

Cílem této práce bylo navrhnout a uvést do provozu třífázový řízený tyristorový usměrňovač. Navržený modul usměrňovače má být kompaktní a bezpečný. Silová a řídicí část budou galvanicky odděleny. Vstupy a výstupy budou přímo uzpůsobeny pro připojení k řídicí desce s mikrokontrolérem [1]. Napájení modulu bude možné buď z řídicí desky, nebo z externího zdroje stejnosměrného napětí. Silová část bude uzpůsobena pro provoz na běžné síti 3x400V.

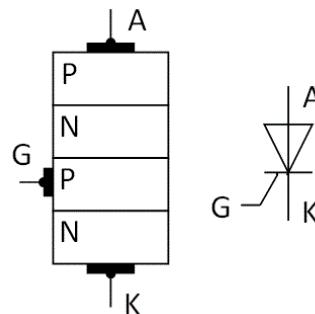
Práce je členěna do několika částí. V první části je popsána teorie o tyristorech a zapojení můstkového usměrňovače, na základě které bude řízení uskutečněno. Druhá část popisuje jednotlivé bloky usměrňovače a vysvětluje jejich princip. Třetí část se věnuje popisu programu pro řízení usměrňovače. Princip běhu programu je popsán pomocí vývojových diagramů. Čtvrtá část obsahuje popis práce na projektu. Poslední část se věnuje měření prototypu v laboratoři. Obsahuje také změřené oscilogramy.

Příloha A obsahuje fotografie realizovaného zařízení, příloha B schéma zapojení, příloha C ukázkou zdrojového kódu programu, příloha D rozmístění součástek na horní vrstvě. V příloze E je obsah příloženého CD.

# 1 Teoretický úvod

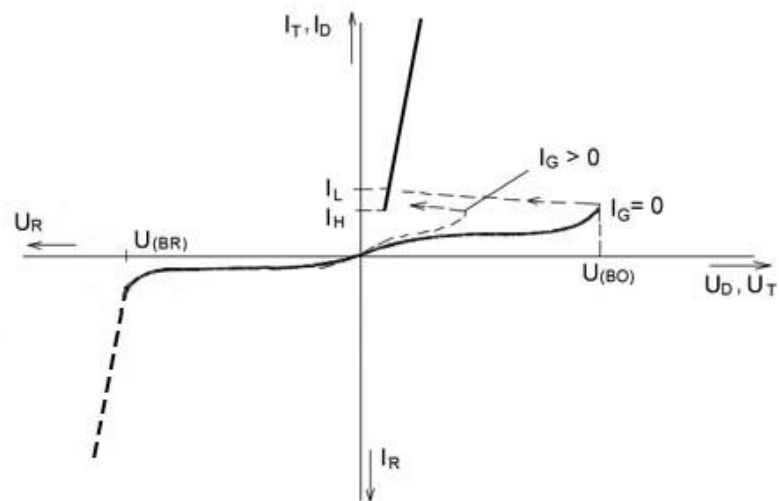
## 1.1 Tyristor

Tyristor je čtyřvrstvá polovodičová spínací součástka se 3 PN přechody. Má vyvedeny 3 vývody – anodu, katodu a gate (viz *Obr. 1*). Tyristor se může nacházet ve 3 stavech vodivosti. Závěrný směr je stejný jako u polovodičové diody. Na tyristoru je závěrné napětí  $u_R$  a protéká jím minimální závěrný proud  $i_R$ . Pokud je potenciál na anodě vyšší než na katodě, je tyristor v blokovacím nebo propustném stavu. V blokovacím stavu je na součástce plné blokovací napětí  $u_D$  a protéká zanedbatelný proud  $i_D$ . Přivedením proudového impulzu předepsaného tvaru a velikosti na gate dojde k přepnutí do propustného stavu (sepnutí). Na součástce je pak úbytek napětí  $u_T$  v řádu jednotek voltů a protéká jí plný proud zátěže. Tyristor je dále možné sepnout překročením blokovacího napětí  $U_{(B0)}$  nebo připojením blokovacího napětí s velkou strmostí  $\frac{du}{dt}$ , kdy vlivem kapacity závěrně polarizovaného přechodu dojde k průchodu nabíjecího proudu a nežádoucímu sepnutí součástky.



*Obr. 1* Struktura a schematická značka tyristoru

Aby se tyristor udržel v propustném stavu, musí jím protékat proud o minimální hodnotě  $I_H$  (vratný (holding) proud). Při zapínání tyristoru musí proud dosáhnout alespoň hodnoty  $I_L$  (přidržený (latching) proud) do doby, než zanikne proud gate. Statickou charakteristiku popisuje *Obr. 2*.



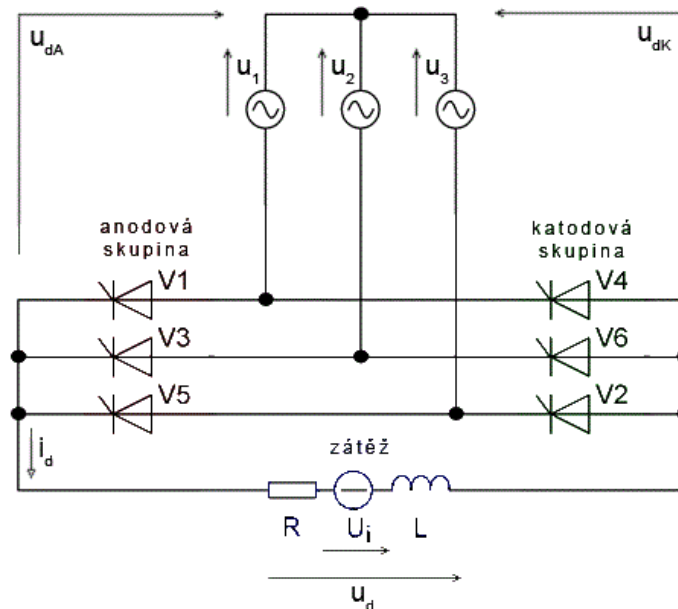
Obr. 2 Statická VA charakteristika tyristoru [2]

Vypnutí tyristoru se provádí přerušením nebo zmenšením proudu pod hodnotu vratného proudu. Speciální typy tyristorů lze vypnout i proudovým impulzem do gate. Dříve než je možné znovu zatížit tyristor blokovacím napětím, je nutné ponechat dobu k obnově blokovací schopnosti  $t_q$ . Velikost se pohybuje v řádu desítek až stovek  $\mu s$ .

Dynamické vlastnosti závisí na kapacitních vazbách mezi PN přechody. Je nutné sledovat strmou náhradu blokovacího napětí  $\frac{du_{AK}}{dt}$ , která může způsobit nežádoucí zapnutí tyristoru. Při spínání tyristoru je dále nutné sledovat strmou náhradu proudu  $\frac{di_{AK}}{dt}$  [3]. Po přivedení spouštěcího impulsu začíná vést proud pouze malá část průřezu, která se postupně rozšiřuje do celé plochy. Pokud by proud narůstal příliš rychle, došlo by k přehřátí a zničení části tyristoru. Rychlost sepnutí tyristoru závisí hlavně na velikosti a strmosti zapínacího proudu  $I_G$ . Hradlové impulzy jsou na gate tyristoru přiváděny přes pulzní transformátor, který má za úkol galvanicky oddělit řídicí a výkonový obvod.

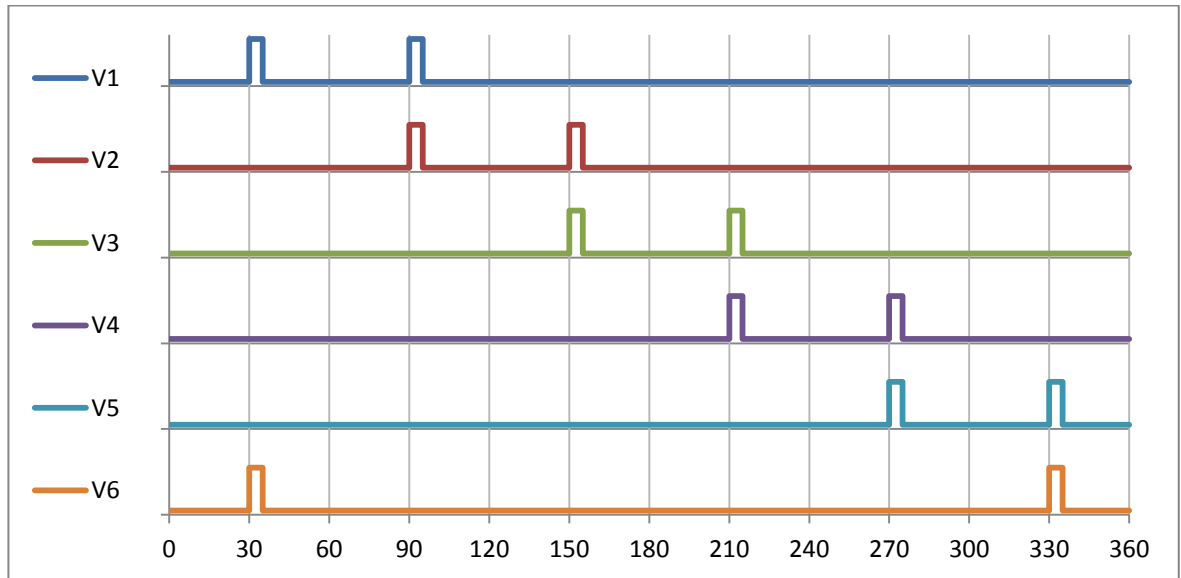
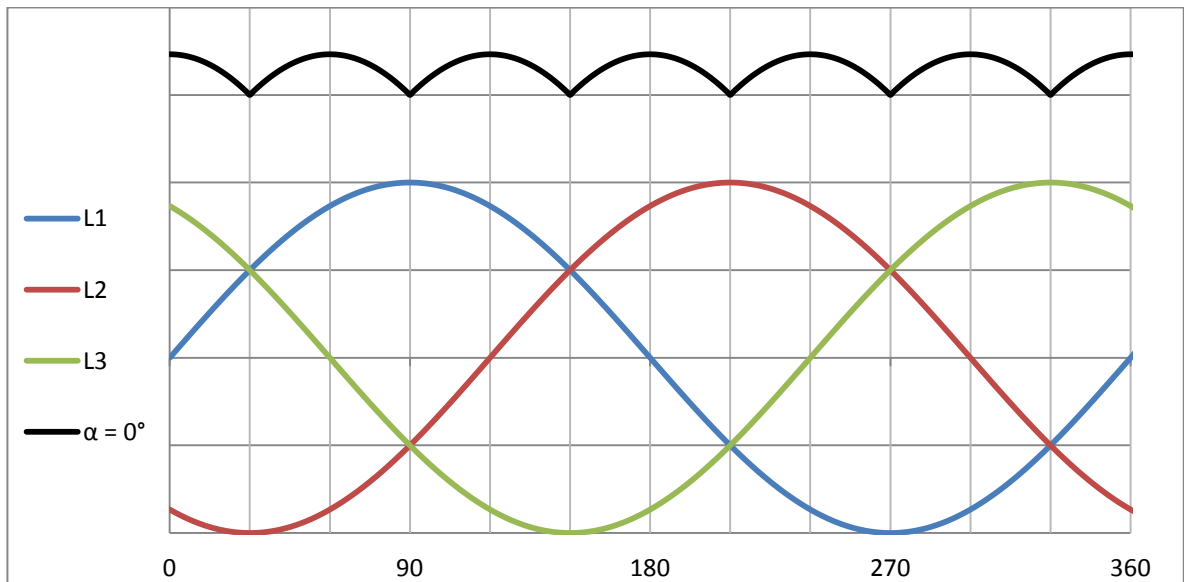
## 1.2 Třífázový plně řízený můstek

Schéma zapojení můstku popisuje *Obr. 3*. Můstek svým zapojením odpovídá neřízenému diodovému usměrňovači, ale všech 6 diod je nahrazeno tyristory [4]. Součástky označené lichými čísly tvoří anodovou skupinu, tyristory s lichými čísly katodovou skupinu. Komutace těchto dvou skupin je navzájem nezávislá. V jeden časový okamžik je sepnut vždy jeden tyristor z anodové a jeden tyristor z katodové skupiny. Zátěž je tak připojena na sdružené napětí mezi příslušnými fázemi. Úhel sepnutí každé součástky bude  $t_v = \frac{2\pi}{m}$ , v našem případě u třífázového zdroje odpovídá  $\frac{2\pi}{3} = 120^\circ$ .



*Obr. 3 Schéma zapojení můstku [5]*

Řízení můstku lze popsat pomocí spínacího diagramu, který zobrazuje proud gate v závislosti na úhlu napětí. Příklady spínacích diagramů jsou uvedeny na *Obr. 4 a Obr. 6*. Příslušné průběhy fázových napětí i usměrněného napětí jsou na *Obr. 5 a Obr. 7*.

Obr. 4 Spínací diagram pro  $\alpha = 0^\circ$ Obr. 5 Průběh napětí pro  $\alpha = 0^\circ$ 

Výsledná střední hodnota napětí usměrňovače se vypočte podle vzorce:

$$U_{di} = U_{diA} - U_{diK}$$

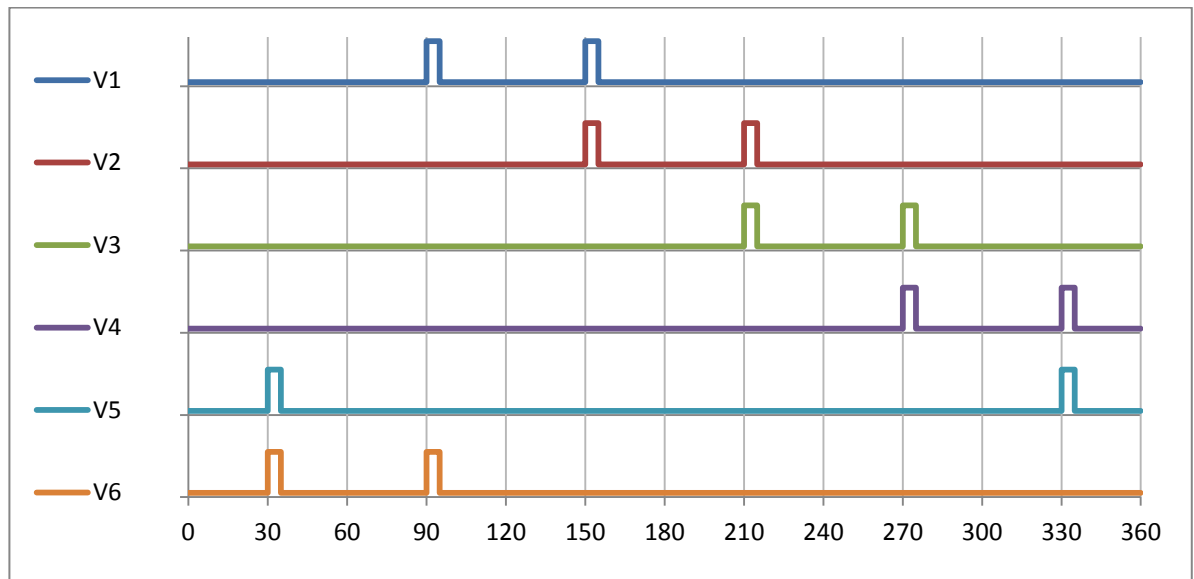
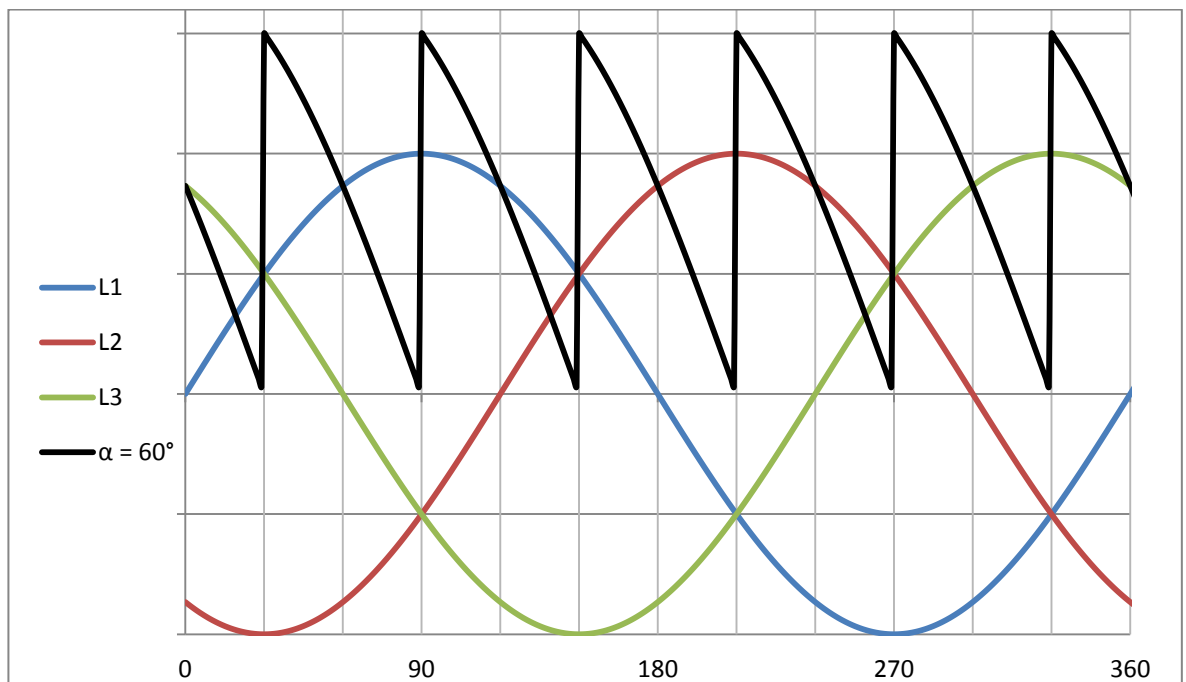
$$U_{diA} = -U_{diK} = \frac{m\sqrt{2} U}{\pi} \sin \frac{\pi}{m} \cos \alpha$$

$U$  je efektivní hodnota fázového napětí,  $m$  udává počet fází – v tomto případě 3 .

$$U_{di} = U_{dio} \cos \alpha$$

$$U_{dio} = \frac{2\sqrt{2} m U}{\pi} \sin \frac{\pi}{m} = \frac{3\sqrt{6}}{\pi} U = \frac{p\sqrt{6} U}{\pi} \sin \frac{\pi}{p}$$

Proměnná  $p$  udává pulznost usměrňovače – trojfázový můstkový usměrňovač patří mezi šestipulzní. Změnou řídicího úhlu  $\alpha$  lze měnit střední hodnotu napětí. Protože výstupní napětí není přímo úměrné řídicímu úhlu  $\alpha$ , je nutné tento úhel přepočítávat podle zadané hodnoty napětí. Protože usměrňovač nemá měření vstupního napětí, jako požadavek se zadává poměrná hodnota napětí  $\frac{U_{di}}{U_{dio}}$ .

Obr. 6 Spínací diagram pro  $\alpha = 60^\circ$ Obr. 7 Průběh napětí pro  $\alpha = 60^\circ$

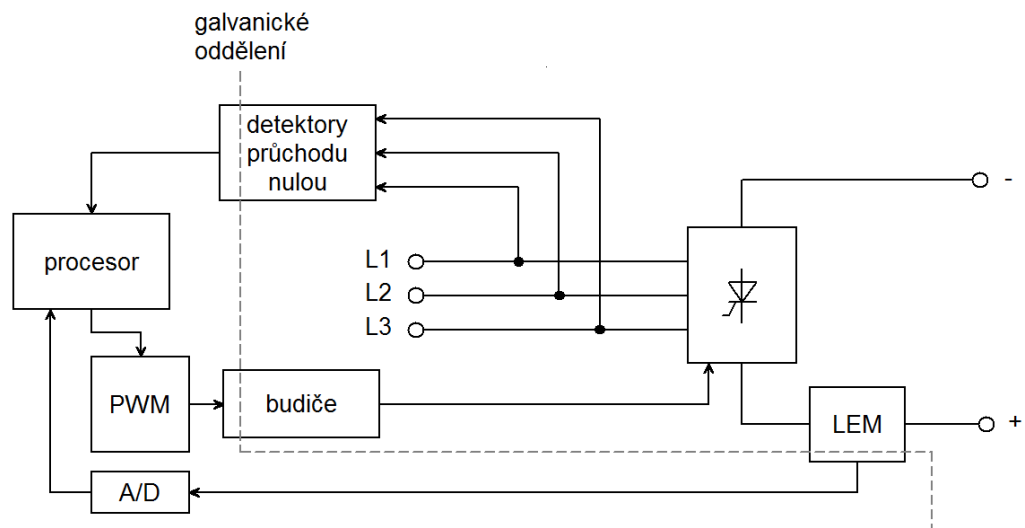
## 2 Popis funkčních bloků

Celkové schéma zapojení je obsaženo v příloze B. Rozvržení součástek na horní straně DPS je v příloze D. Blokové schéma celého usměrňovače je na *Obr. 8*.

### 2.1 Výkonový obvod

Základem usměrňovače je kompaktní modul Semikron SK 40 DT (SK1) se 6 tyristory, zapojenými jako třífázový můstek. Oproti použití samostatných tyristorů se tím zjednodušil návrh plošného spoje i volba chladiče. Vzhledem k ostatním součástkám je předimenzován z důvodu nedostupnosti třífázových modulů pro nižší jmenovité proudy. Každý tyristor byl doplněn RC ochranou (např. kombinace R11, C7), která chrání součástku proti přepětí při komutaci. Při volbě součástek byla využita práce [6].

Ve stejnosměrném obvodu je zařazena proudová sonda LEM CAS25-NP pro měření výstupního proudu. Na vstupu (H2) i výstupu (H7) výkonového obvodu byly použity konektory Wago 236.



*Obr. 8 Blokové schéma celého usměrňovače*

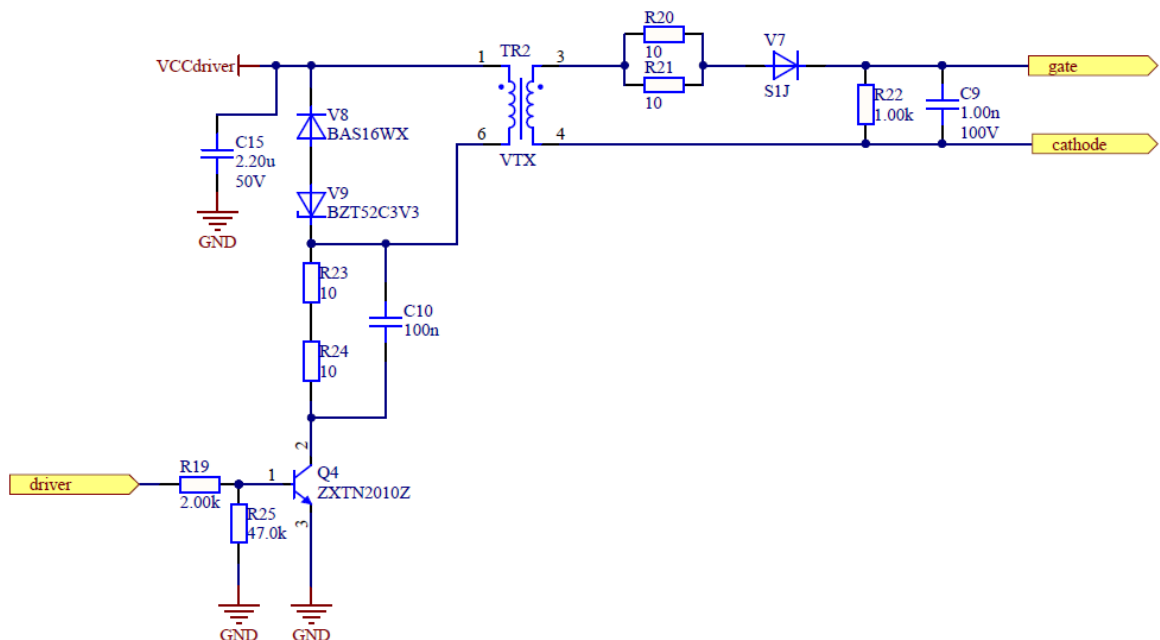


## 2.2 Budiče tyristorů

Hlavním úkolem budiče (schéma viz *Obr. 9*) je přivést spouštěcí impulzy správného tvaru a velikosti na gate tyristoru. Zároveň budič zajišťuje galvanické oddělení řídicí a výkonové části obvodu. Galvanické oddělení je zajištěno pomocí pulzního transformátoru. Byl použit model Vigortronix VTX-111-010 s převodem 2:1 a izolační pevností 2,8kV.

Budiče je možné napájet buď z řídicí desky přes konektor H6 (viz příloha D), nebo z externího zdroje stejnosměrného napětí. Výběr napájecího zdroje se volí pomocí jumperu na konektoru H3. Maximální napětí budiče je omezeno maximálním proudem použitého pulzního transformátoru (dle katalogového listu je maximální proud 200mA). Pro spolehlivou funkci budiče postačuje napětí 5V, jak bylo ověřeno při testování.

Sepnutím tranzistoru Q4 proudem z PWM modulátoru začne celým obvodem procházet proud. Proudový impulz je přenesen přes transformátor na gate tyristoru. Dioda V7 na sekundární straně zajistí, aby do tyristoru šly jen pulzy správné polarity. Diody V8 a V9 zajistí omezení přechodného děje při vypnutí spínacího tranzistoru. Je možné programově ovlivnit, zda bude na výstupu budiče souvislá řada pulzů, nebo jen jeden pulz pro každou periodu. Podobně je možné nastavit délku pulzu.

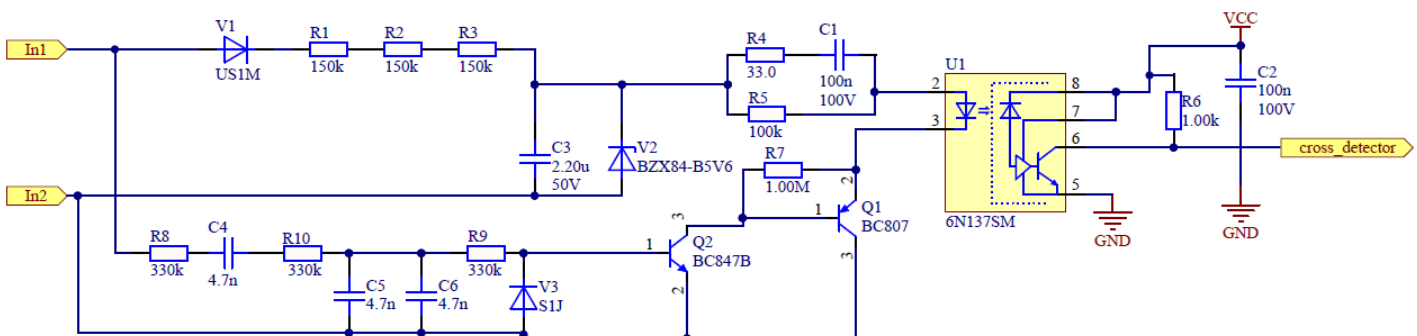


*Obr. 9 Schéma budiče tyristoru*

## 2.3 Detektory průchodu nulou

Úkolem detektoru průchodu nulou je zajistit synchronizaci řídicí jednotky s jednotlivými fázemi. Má také za úkol zjistit sled fází. V tomto projektu je detektor připojen na sdružené napětí. Neměří tedy průchody jednotlivých fází nulou, ale koincidenci dvou fází.

Schéma zapojení je uvedeno na *Obr. 10*. Pokud hodnota napětí na vstupu 1 přesáhne hodnotu napětí na vstupu 2, dojde k sepnutí tranzistorů Q1 a Q2 a proud začne procházet přes rezistor R4, kondenzátor C1 a vstupní diodu optočlenu U1. Než dojde k nabití kondenzátoru C1 a přerušení proudu, proudový impuls o délce přibližně 20 $\mu$ s se přenesse na výstup optočlenu. Optočlen používá zapojení s otevřeným kolektorem. Na výstup je připojen pull-up rezistor R6 a aktivní výstup je v nízké úrovni napětí. Pokud hodnota napětí na vstupu 1 klesne zpět pod hodnotu napětí na vstupu 2, tranzistory se uzavrou a kondenzátor C1 se vybijí přes rezistory R4 a R5, aby mohl v další periodě opět přenést impuls.



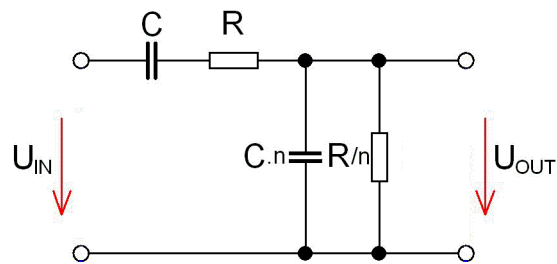
*Obr. 10 Schéma detektoru průchodu nulou*

Při tomto pulzu se vyvolá v programu přerušení, které synchronizuje čítač pro spínání jednotlivých tyristorů. Po zapnutí usměrňovače program nejprve zjistí sled fází. Při výpadku synchronizačních impulsů (např. při odpojení napájení) se opět spustí funkce pro zjištění sledu fází, která čeká, dokud synchronizace od všech fází nebude v pořádku. Vývojový diagram programu pro obsluhu detektoru je uveden v kapitole 3 na *Obr. 13*.

Schéma detektoru průchodu nulou bylo částečně převzato z práce [7]. Podstatnou změnou je doplnění vstupního filtru. Ten je zapojen jako Wienův článek (schéma viz *Obr. 11*) a rezonanční frekvence je 50Hz. Obvykle se volí hodnota  $n = 1$ , kdy hodnota odporů i kapacit je

v obou větvích stejná. Zde bylo zvoleno  $n = 2$ , protože v původním návrhu byly osazeny 3 rezistory. Hodnota  $R$  v podélné větvi tedy odpovídá dvojnásobku odporu jednoho z nich –  $R = 660\text{k}\Omega$ . Na základě následující rovnice byla dopočítána hodnota kapacity  $4,7\text{nF}$ . Filtr omezuje vliv komutace na činnost detektoru.

$$f = \frac{1}{2\pi RC} \quad C = \frac{1}{2\pi Rf} = \frac{1}{2\pi * 660 * 10^3 * 50} = 4,82 * 10^{-9}\text{F} \cong 4,7\text{nF}$$



Obr. 11 Schéma Wienova článku

## 2.4 Měření proudu

Modul usměrňovače je osazen proudovou sondou LEM CAS25-NP. Analogový výstup napětí je úměrný procházejícímu proudu. Napájecí napětí sondy je  $5\text{V}$ , nulový proud odpovídá polovině této hodnoty. Výstup je zpracován A/D převodníkem na řídicí desce a může sloužit jako zpětná vazba pro regulátor proudu.

## 3 Popis programu

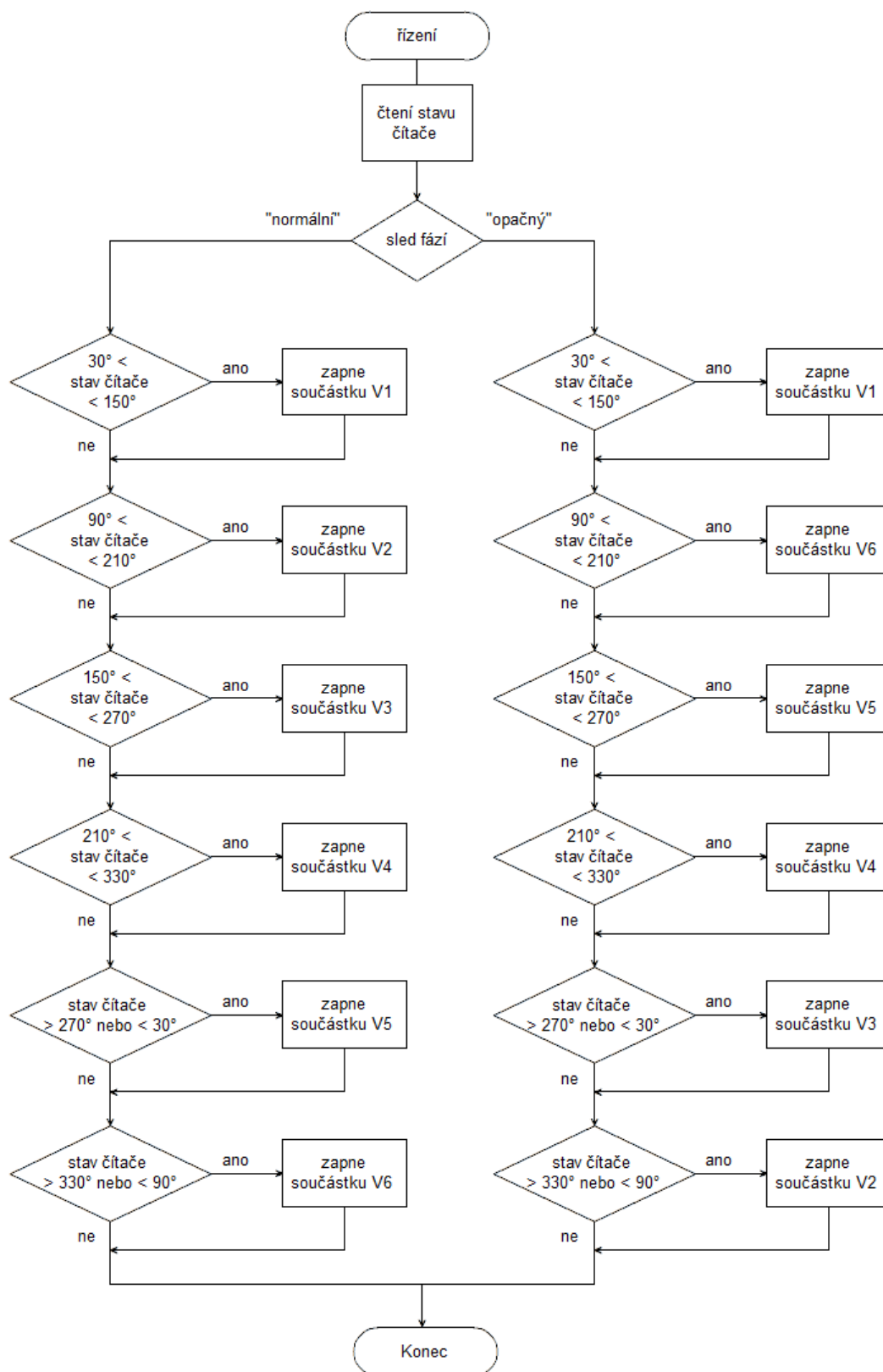
### 3.1 Čítač

Základem programu pro řízení usměrňovače je čítač. Použitý procesor obsahuje 32bitový čítač, který je dekrementován při každém taktu hodin. Takt hodin pro čítač je možné nastavit pomocí děličky, v tomto případě je použito dělení 50 při taktu procesoru 150MHz. Při podtečení čítače (dosažení nuly) je do čítače nahrána hodnota nastavené periody (v tomto případě 60000). Jeden cyklus čítače odpovídá času 20ms, a tedy frekvenci sítě 50Hz. Program podle aktuální hodnoty čítače spíná a vypíná výstup PWM modulátoru, který generuje zapínací pulzy pro jednotlivé tyristory. Funkce pro zjištění stavu čítače (a sepnutí tyristorů) je volána pomocí přerušení každých 20 $\mu$ s. Aktuální úhel je přímo úměrný hodnotě čítače, úhel 0° byl zvolen do poloviny čítače. Základní princip programu pro řízení usměrňovače vychází z prací [7] a [8].

Pro celý rozsah řídicího úhlu  $\alpha$  se nemění ani pořadí spínání, ani doba sepnutí jednotlivých součástek. Řídicí úhel  $\alpha$  znamená zpoždění komutace proti okamžiku, kdy by došlo ke komutaci u diodového usměrňovače. Čítač je programově zpožděn oproti úhlu napětí sítě. Hodnota čítače včetně tohoto zpoždění je nastavena při synchronizaci.

Při podtečení vyvolá čítač přerušení, ve kterém kontroluje počet průchodů nulou od poslední synchronizace. V bezchybném stavu k tomu dojde jednou. Pokud počet period bez synchronizace přesáhne nastavenou hodnotu, spustí se funkce pro zjištění sledu fází a spínání tyristorů se zastaví.

Funkce pro spínání jednotlivých součástek je zjednodušeně popsána vývojovým diagramem na *Obr. 12*. Podle sledu fází (hodnoty globální proměnné *reverse*) se zvolí pořadí spínání. Program při spínání změní hodnotu řídicího registru PWM modulátoru, který začne generovat obdélkový průběh na příslušném kanálu. Ten se pomocí budiče zesílí, přenesení na gate tyristoru a dojde k sepnutí součástky. Zároveň je v řídicím registru deaktivováno spínání součástek, které nemají být sepnuty. Délku pulzu i mezery lze nastavit pomocí dalších řídicích registrů. V programu je perioda nastavena jako konstanta `PWM_PERIOD`. Porovnávací registr obsahuje poloviční hodnotu, aby byl zachován poměr 1:1 pro pulz a mezeru. Při testování byla použita perioda 100 $\mu$ s, při které spínání fungovalo spolehlivě.



Obr. 12 Vývojový diagram programu pro spínání součástek

## 3.2 Synchronizace

Synchronizační pulz na detektoru L1/L2 vyvolá v programu přerušení, které nastaví hodnotu čítače. Nulový úhel odpovídá polovině používaného rozsahu čítače (30000 kroků). Zapojením detektorů průchodu nulou na sdružené napětí dojde k posunu o  $30^\circ$  oproti průchodu nulou samotné fáze (5000 kroků). Detektor má určité zpoždění přenosu, v programu nastavené jako konstanta OFFSET (-830 kroků). Nastavená hodnota je součtem všech těchto hodnot a aktuálního řídicího úhlu  $\alpha$ , přepočteného na rozsah čítače (60000 kroků =  $360^\circ$ ). Přepočtený řídicí úhel je uložen v globální proměnné *alfa6*. Při synchronizaci se zároveň nuluje hodnota počtu chybějících synchronizací. Zpoždění přenosu není podle měření závislé na napětí, a lze ho kompenzovat přičtením konstanty.

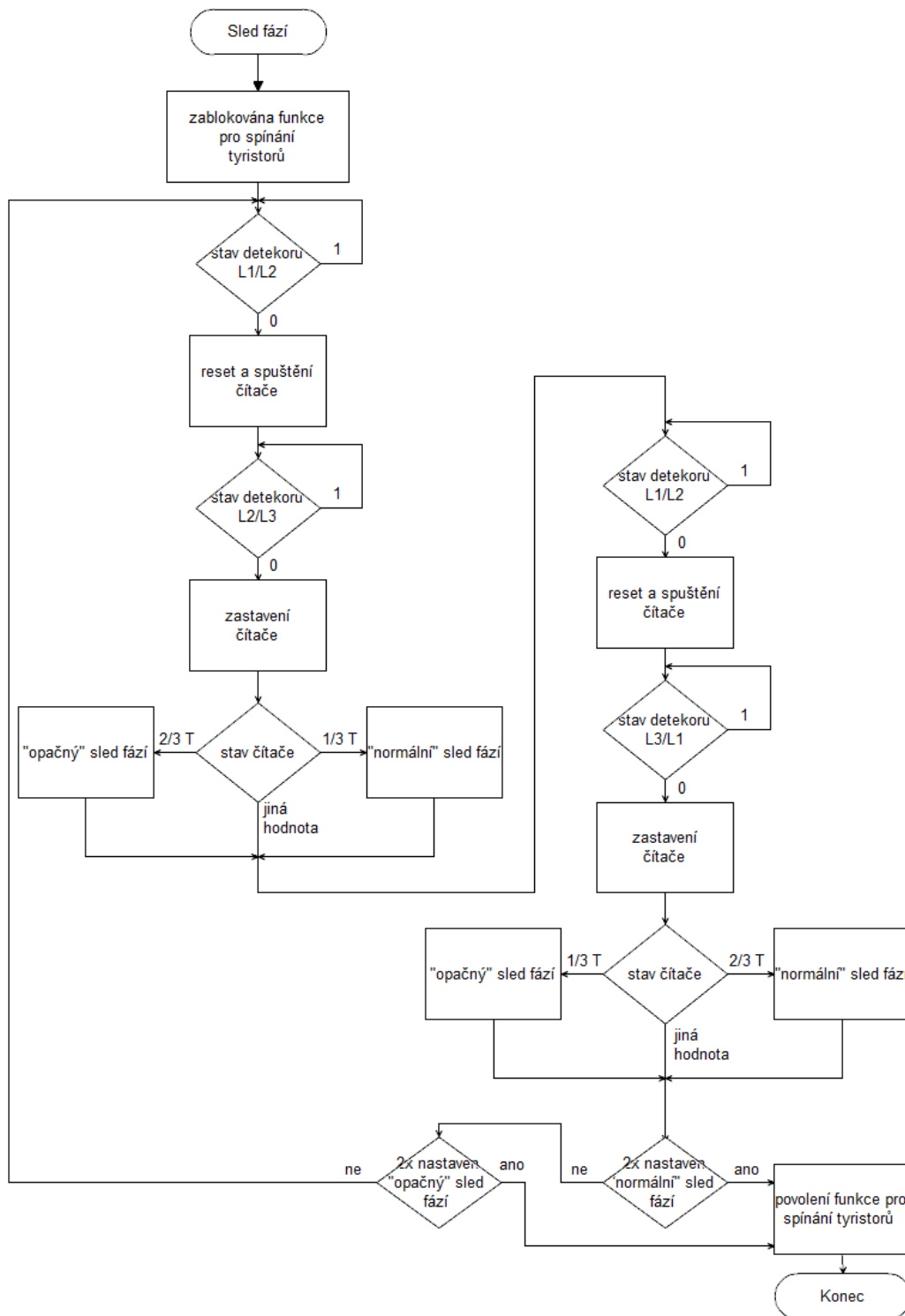
## 3.3 Nastavení řídicího úhlu

Řídicí úhel alfa se nastavuje pomocí funkce *setAlfa()*, která má jako vstupní parametr požadovaný řídicí úhel ve stupních. Je možné ho zadat i jako desetinné číslo, funkce požadavek přepočítá na krok čítače. Pokud je požadovaný úhel větší než  $170^\circ$ , nebo je menší než  $0^\circ$ , je omezen na tyto hodnoty.

Řídicí úhel může být nastavován ručně při debug režimu nebo pomocí naprogramovaného regulátoru.

## 3.4 Sled fází

Zjišťování sledu fází řeší program na základě informací z detektorů průchodu nulou. Program rozlišuje 2 stavy – „normální“ a „opačný“ sled fází. Stav odpovídá směru otáčení točivého pole. V programu je stav zaznamenán jako globální proměnná *reverse*. Nulová hodnota značí „normální“ sled fází (L1-L2-L3), nenulová hodnota nastává při prohození 2 fází, které otočí směr točivého pole. Podle sledu fází se řídí pořadí spínání jednotlivých tyristorů. Algoritmus zjištění sledu fází a jejich funkčnosti je uveden na *Obr. 13*.



Obr. 13 Vývojový diagram programu pro zjištění sledu fází

Program při kontrole sledu fází kontroluje funkčnost všech 3 fází. Pro samotné měření sledu fází by teoreticky stačily jen 2 detektory průchodu nulou. Nebyla by ale možná kontrola zbývajících fází, ani potenciální provoz na 2 fáze při poruše (jako jednofázový můstek). Kontrola sledu fází je spuštěna při startu programu a při delším výpadku synchronizace.

## 4 Postup práce

### 4.1 Návrh DPS

Návrh plošného spoje probíhal v programu Altium Designer. Jedná se o velmi rozsáhlý program pro návrh i kontrolu DPS a generování výstupních dat pro výrobu. Součástky byly vybrány z knihovny vytvořené na FEL ZČU. K některým složitějším součástkám, které nebyly dostupné z knihovny, bylo nutné footprinty vytvořit. Šířka spojů a izolačních vzdáleností byla zvolena při konzultacích. Izolační vzdálenost mezi silovou a řídicí částí byla zvolena minimálně 5mm, což by mělo odpovídat platným normám. Podobně byly řešeny i popisky a nutné podklady pro výrobu DPS [9].

### 4.2 Výroba DPS

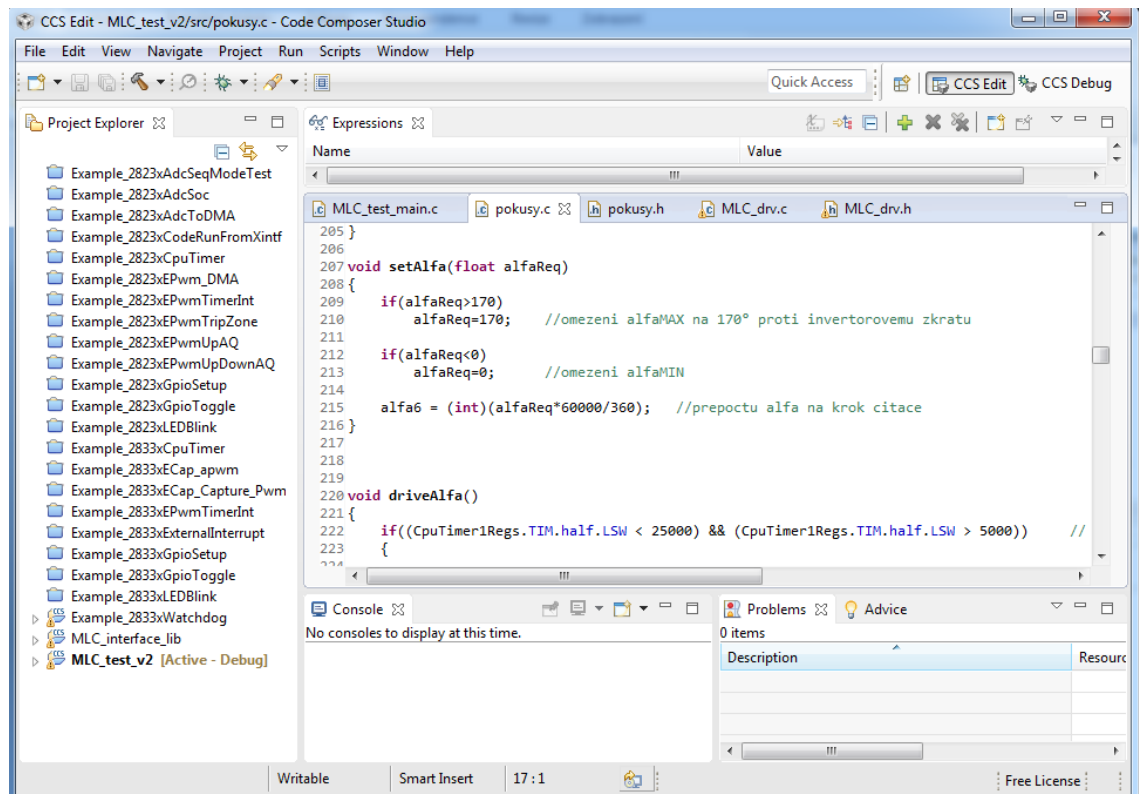
Po opravení zjištěných chyb byla DPS vyrobena ve firmě Pragoboard. Osazení součástek bylo provedeno ručně v laboratoři KEV. Přes 90% součástek bylo typu SMD, což usnadnilo osazování. Vývodové součástky byly pouze konektory, transformátory, tyristorový modul a proudová sonda.

### 4.3 Programování

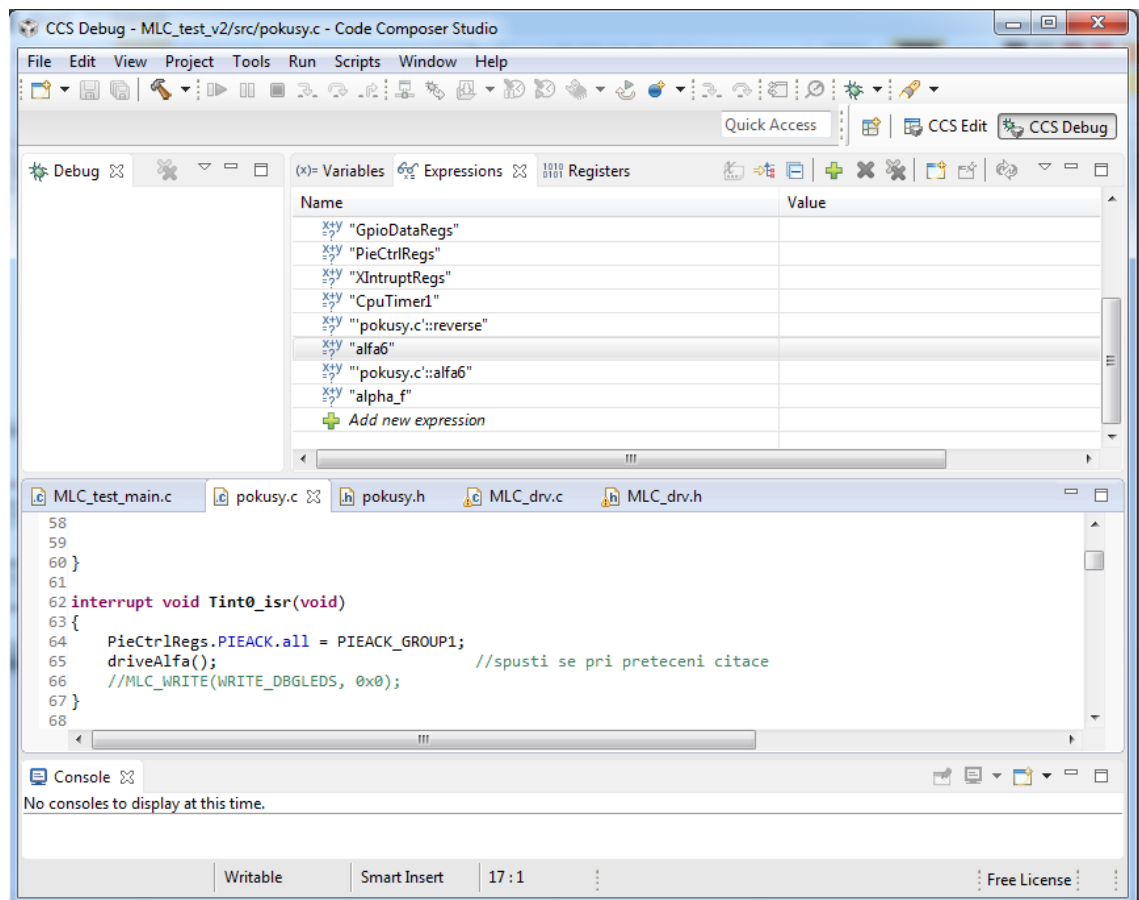
Programování probíhalo v prostředí Code Composer Studio, dodávaném výrobcem procesoru. Před samotným programováním bylo nutné nastudovat katalogový list procesoru a využívaných periférií [10]. Pro testování byla zapůjčena řídicí deska MLC Interface. Ověřování funkčnosti jednotlivých částí kódu bylo prováděno v laboratoři FEL.

Prostředí Code Composer Studio nabízí 2 režimy – Edit (*Obr. 14*) a Debug (*Obr. 15*). První je určen pro psaní kódu, druhý pro ladění programu. Přímo za běhu je možné číst a přepisovat obsah jednotlivých registrů i proměnných.





Obr. 14 Prostředí Code Composer Studio – režim Edit



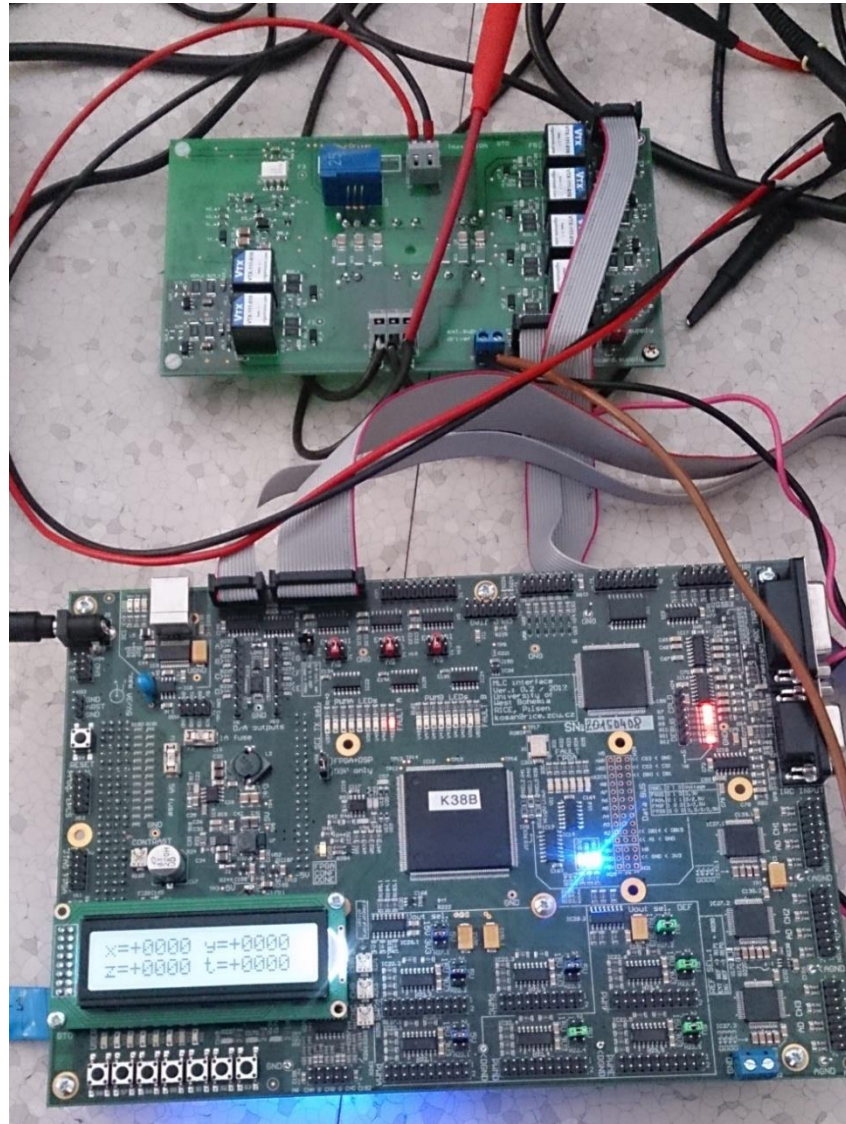
Obr. 15 Prostředí Code Composer Studio – režim Debug

## 5 Měření

### 5.1 Průběh měření

Po dokončení osazování proběhly testy v laboratoři (viz *Obr. 16*). Nejprve úspěšně proběhl test, zda nedošlo při osazování ke zkratu mezi spoji. Byly odhaleny drobné odchylky v popisu, oproti původnímu návrhu. Popisky fází U, V a W jsou v opačném pořadí než fáze L1,L2,L3 v návrhu DPS. Podobně je otočeno i připojení fází L1,L2,L3 na jednotlivé větve můstku. Fáze L1, podle které probíhá synchronizace programu, je připojena na větev se součástkami V2 a V5. Bylo tedy nutné upravit program pro řízení. Než byla tato chyba odhalena, usměrňovač pracoval při pořadí fází U, V, W jako jednofázový můstek, při opačném sledu fází nefungoval vůbec. Při ladění bylo také nutné správně nastavit synchronizaci. Protože je detektor průchodu nulou zapojen na sdružené napětí, pulz z něj přichází o 30° dříve. Podobně bylo nutné vykompenzovat i drobné zpoždění detektoru. Při měření se prokázalo, že toto zpoždění není závislé na napájecím napětí.

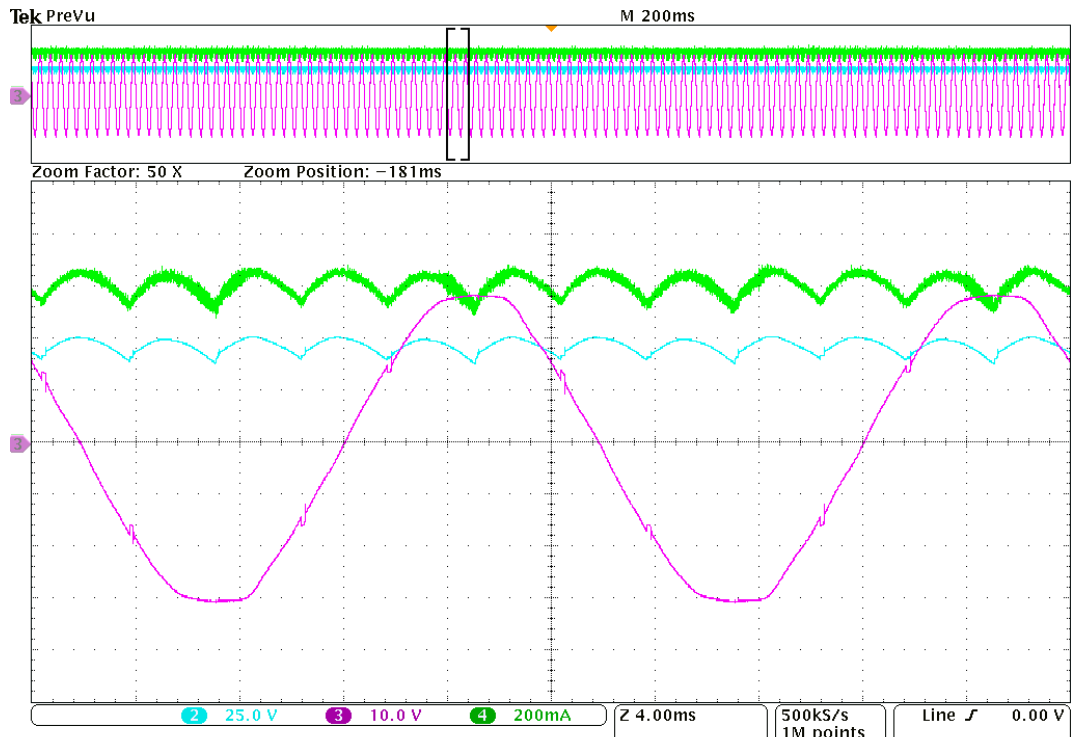
Po těchto úpravách programu byl usměrňovač shledán funkčním.



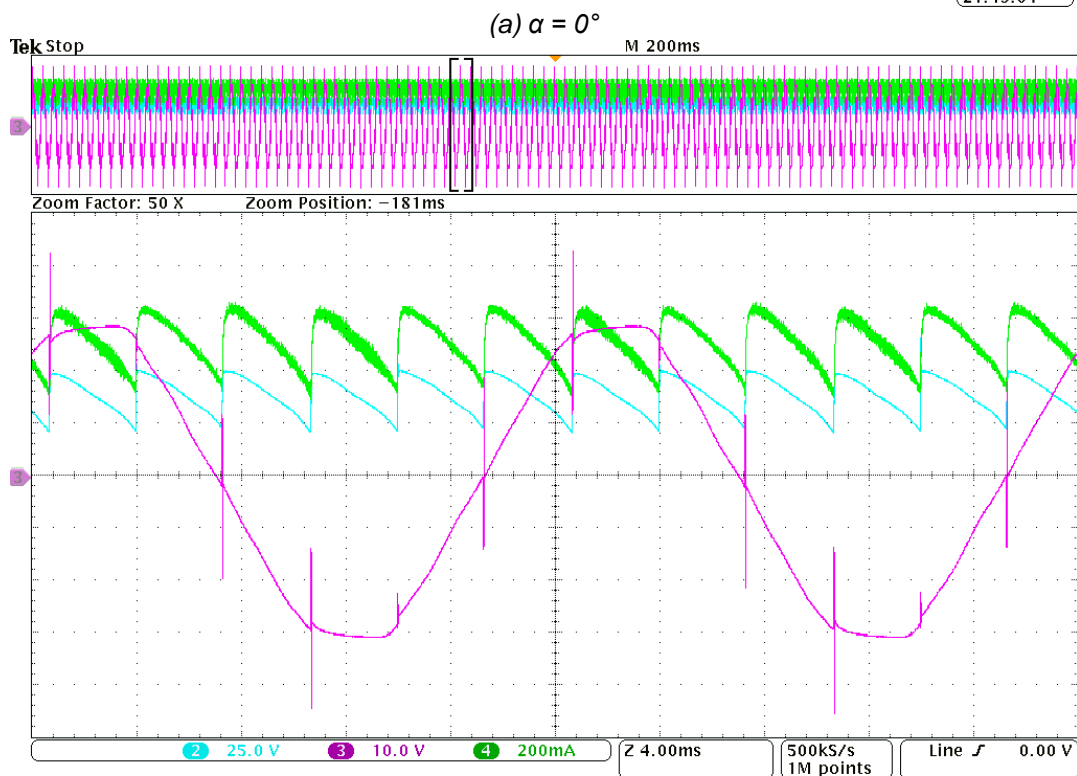
Obr. 16 Sestava při měření

Testování proběhlo nejprve s R zátěží, pak s RL zátěží. Indukčnost použité cívky byla ale vzhledem k odporu rezistoru malá a v měření se induktivní charakter příliš neprojevil. Naměřené oscilogramy pro různá nastavení řídicího úhlu  $\alpha$  jsou na Obr. 17 a Obr. 18. Usměrňovač je schopen spolehlivého provozu od napětí 50V.

## 5.2 Oscilogramy z měření



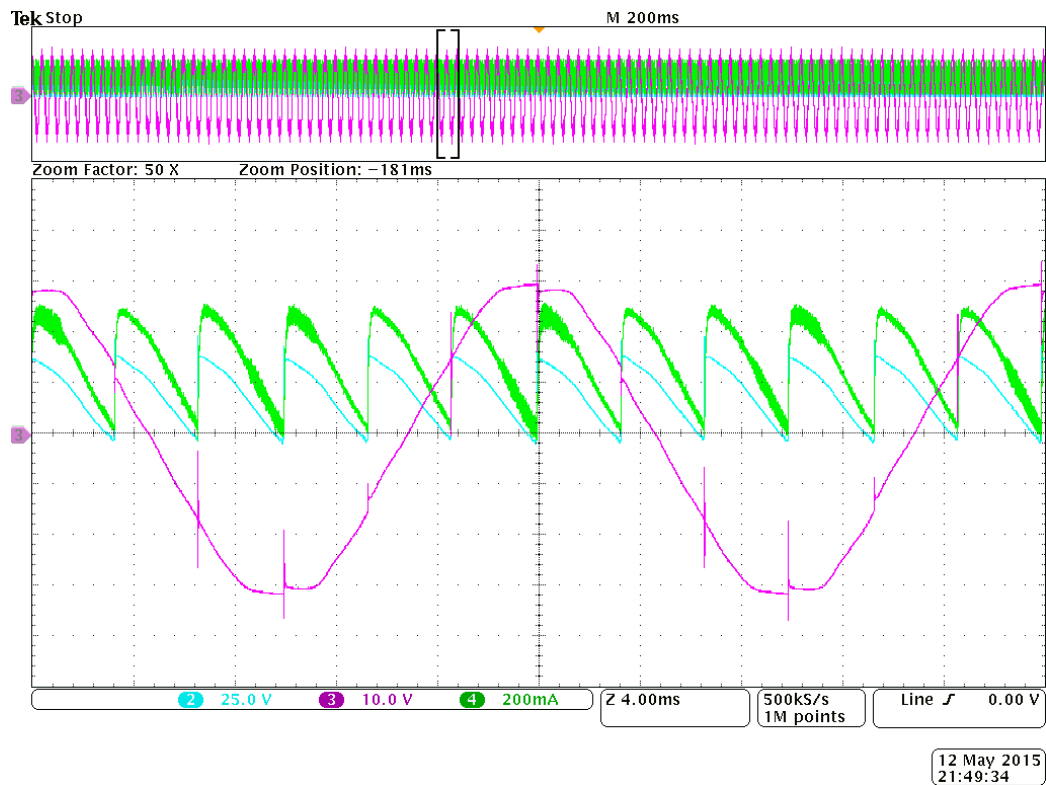
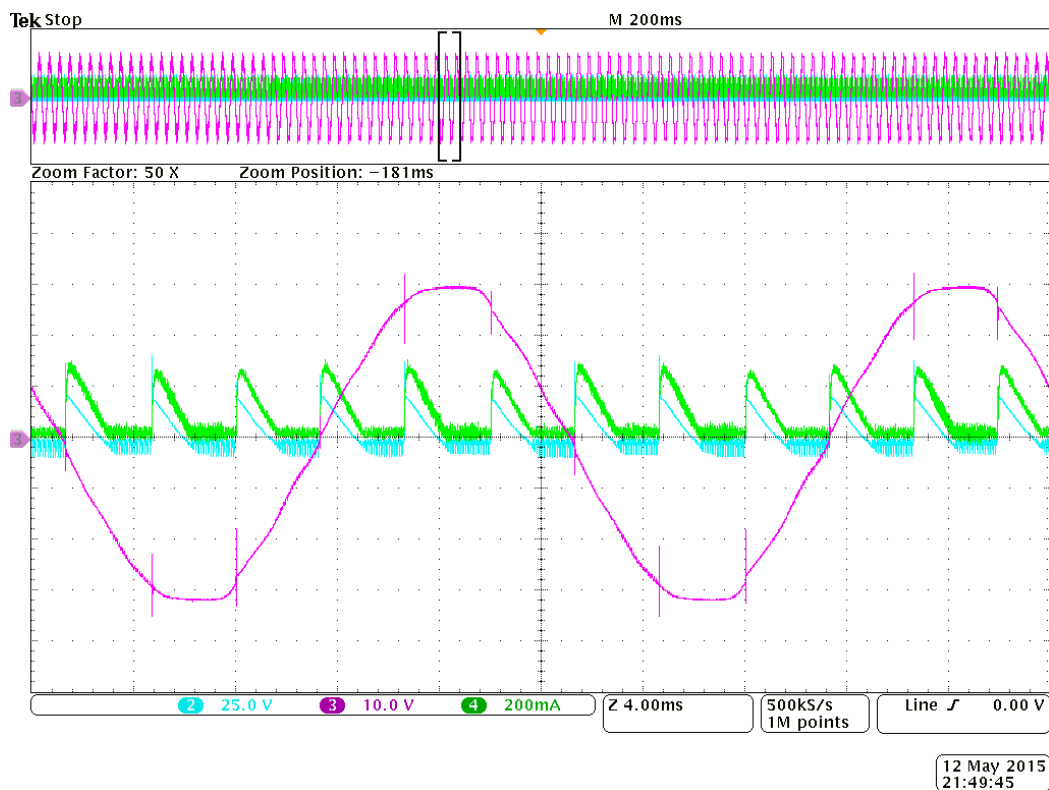
12 May 2015  
21:49:04



12 May 2015  
21:49:22

(b)  $\alpha = 30^\circ$

Obr. 17 Průběhy proudu a napětí, CH2 (světle modrá): výstupní napětí usměrňovače  $u_D$  (25V/d), CH3 (fialová): fázové napětí sítě (10V/d), CH4 (zelená): výstupní proud usměrňovače při RL zátěži  $i_D$  (200mA/d)

(a)  $\alpha = 60^\circ$ (b)  $\alpha = 90^\circ$ 

Obr. 18 Průběhy proudu a napětí, CH2 (světle modrá): výstupní napětí usměrňovače  $u_D$  (25V/d), CH3 (fialová): fázové napětí sítě (10V/d), CH4 (zelená): výstupní proud usměrňovače  $i_D$  při RL zátěži (200mA/d)

## Závěr

Cílem této práce bylo navrhnout kompaktní desku plošného spoje s tyristorovým usměrňovačem, kterou bude možné ovládat pomocí řídicí desky MLC interface.

Během přibližně půl roku byla navržena, vyrobena a osazena deska plošného spoje. Vytvořený řízený třífázový můstkový usměrňovač byl otestován a změřen. Výsledný prototyp je plně funkční.

Program pro řízení usměrňovače byl vytvořen pro otestování funkčnosti usměrňovače. V souladu se zadáním probíhá řízení pouze v otevřené smyčce. Použitý procesor i deska MLC interface umožňují rozšíření programu o další možnosti, jako je např. regulace proudu nebo řízení dalšího měniče. Zpětnou vazbu je možné doplnit připojením výstupu proudové sondy na A/D převodník řídicí desky a úpravou programu.

Až v průběhu osazování a testování byly odhaleny některé chyby v návrhu. Převrácené popisky fází a součástek ve větvích usměrňovače byly vyřešeny úpravou programu. Další zjištěnou chybou bylo prohození napájecího a výstupního pinu u proudové sondy. Tato chyba vznikla již při tvorbě footprintu z důvodu špatné přehlednosti katalogového listu. U prototypu byla chyba opravena přerušením plošného spoje a připájením drátků. V návrhu je možné tuto chybu snadno opravit.

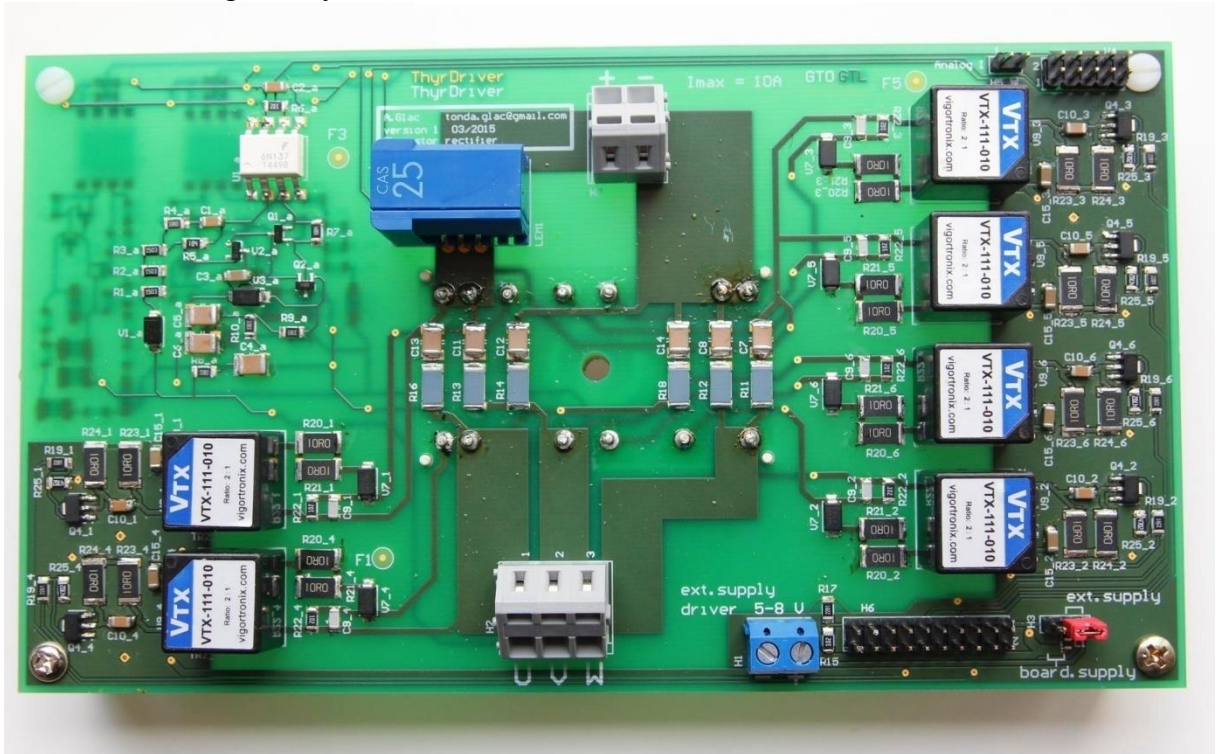
Vyrobený prototyp bude sloužit jako zdroj stejnosměrného napětí v laboratoři KEV.

## Seznam literatury a informačních zdrojů

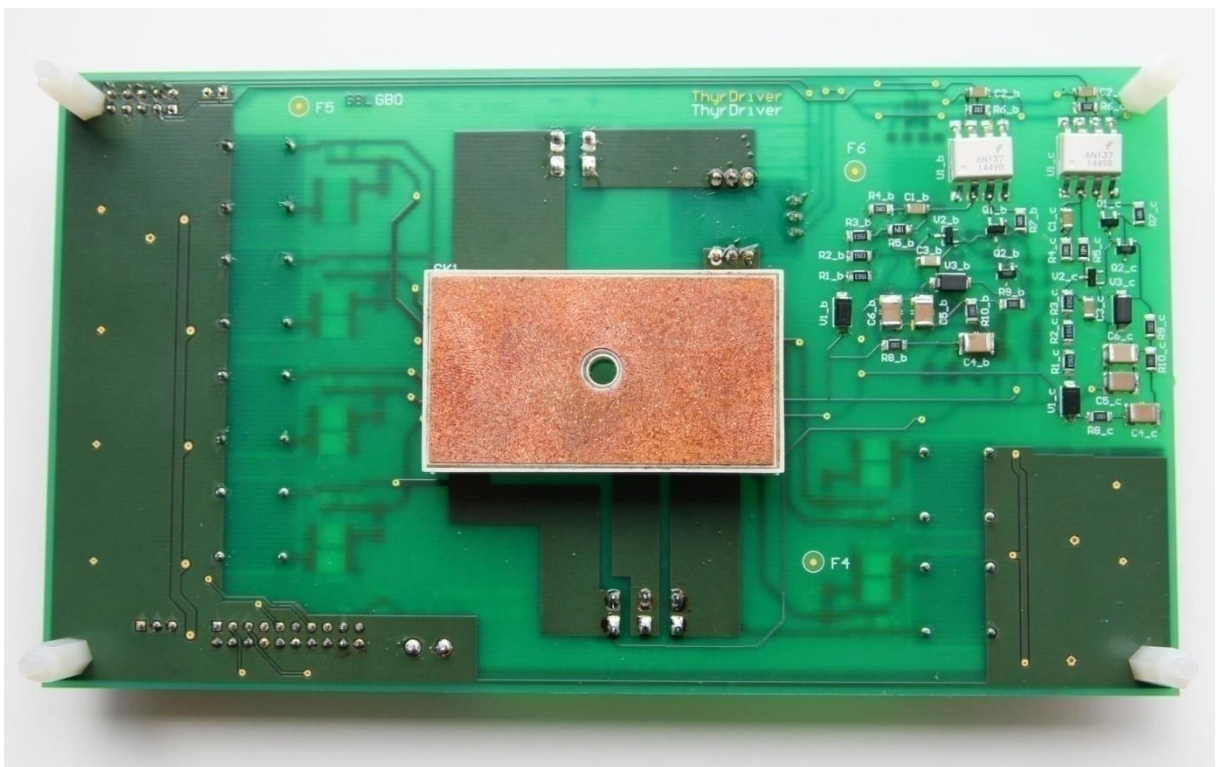
- [1] KOŠAN, Tomáš. *MLC interface - vývojový kit pro víceúrovňové měniče s procesorem a FPGA*. Plzeň, 2013. Výzkumná zpráva č. 22190-010-2012. Fakulta elektrotechnická. Regionální inovační centrum elektrotechniky.
- [2] TONHAUSEROVÁ, Markéta. *Mikroelektronické systémy* [online]. 2006. VŠB-TU Ostrava. [cit.5.2.2015]  
Dostupné z: <http://352lab.vsb.cz/ServerFinalVer/Tonhauser/spin.html>
- [3] VONDRÁŠEK, František. *Výkonová elektronika - svazek I. Západočeská univerzita v Plzni*, 1994. ISBN 80-7082-136-1
- [4] VONDRÁŠEK, František. *Výkonová elektronika - svazek II. Západočeská univerzita v Plzni*, 1994. ISBN 80-7082-137-X
- [5] PAVLÍČEK, Jiří. *EDUCON* [online]. 2005. ZČU FEL KEV. [cit. 5.2.2015]  
Dostupné z: <http://educon.zcu.cz/rservice.php?akce=tisk&cislomodulu=2005022204>
- [6] ROBBINS P. William. *Snubber circuits* [online]. 1997. University of Minnesota. Dept. of Electrical Engineering. [cit. 20.2.2015]  
Dostupné z: <http://space.ednchina.com/Upload/2009/10/27/3707851d-d135-4b92-81c4-1b2635d16c60.pdf>
- [7] KOŠAN, Tomáš. *Implementace algoritmu řízení tyristorového usměrňovače do mikropočítače*. Plzeň, 2008. Diplomová práce. Západočeská univerzita. Fakulta elektrotechnická. Katedra energetiky a ekologie. Vedoucí práce Jan Molnár.
- [8] MISSIRLIU, P., *Universal thyristor driving circuit using Atmel AVR microcontroller*, 2002
- [9] ZÁHLAVA, V.: *Metodika návrhu plošných spojů*, ČVUT Praha, 2000. ISBN 8001021939
- [10] TEXAS INSTRUMENTS. *TMS320F28335 digital signal controller (Rev M.)* [online], 2012. Dostupné z: <http://www.ti.com/lit/ds/symlink/tms320f28335.pdf>

## Přílohy

### Příloha A – Fotografie vytvořeného usměrňovače



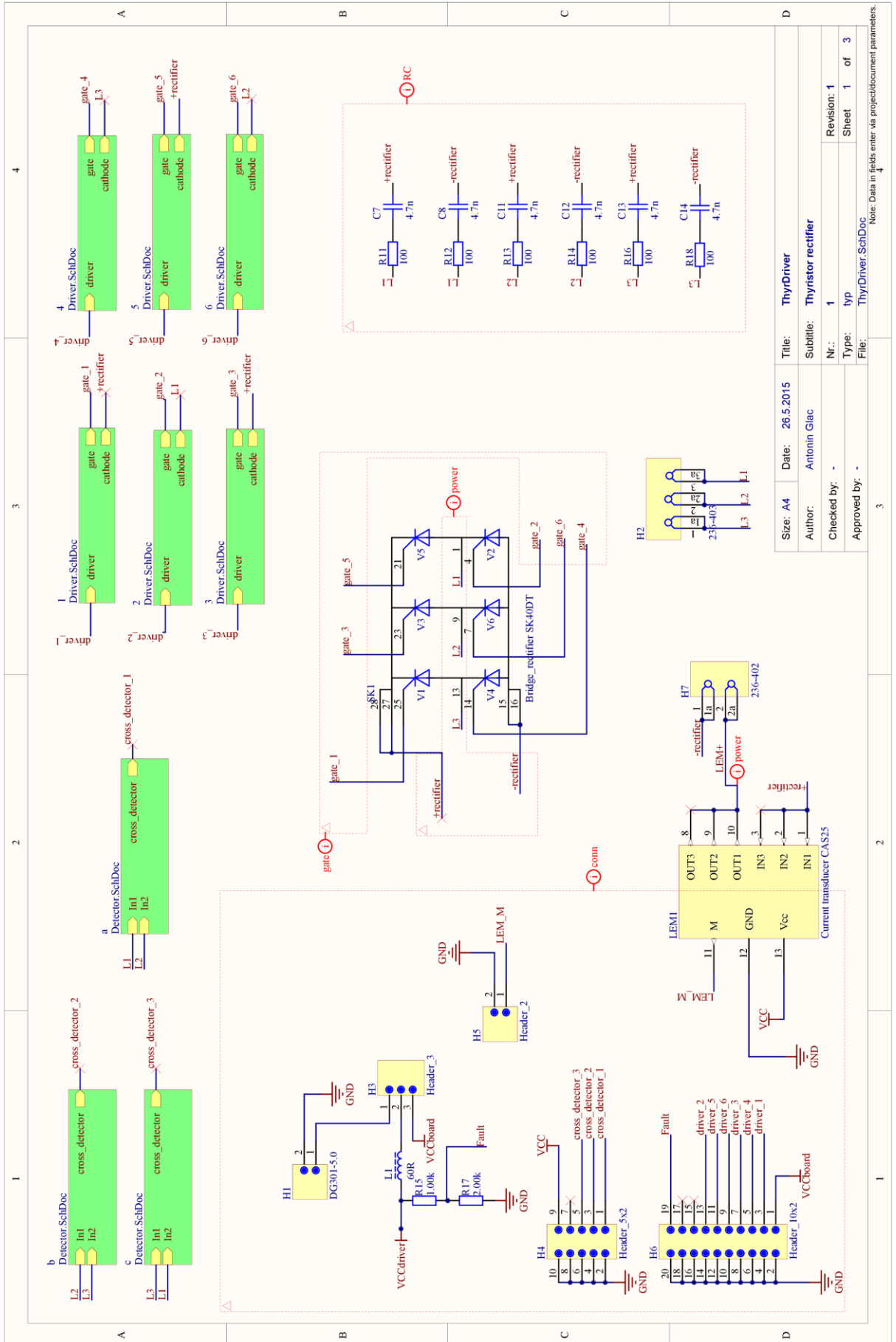
Obr. 1 Horní strana modulu usměrňovače

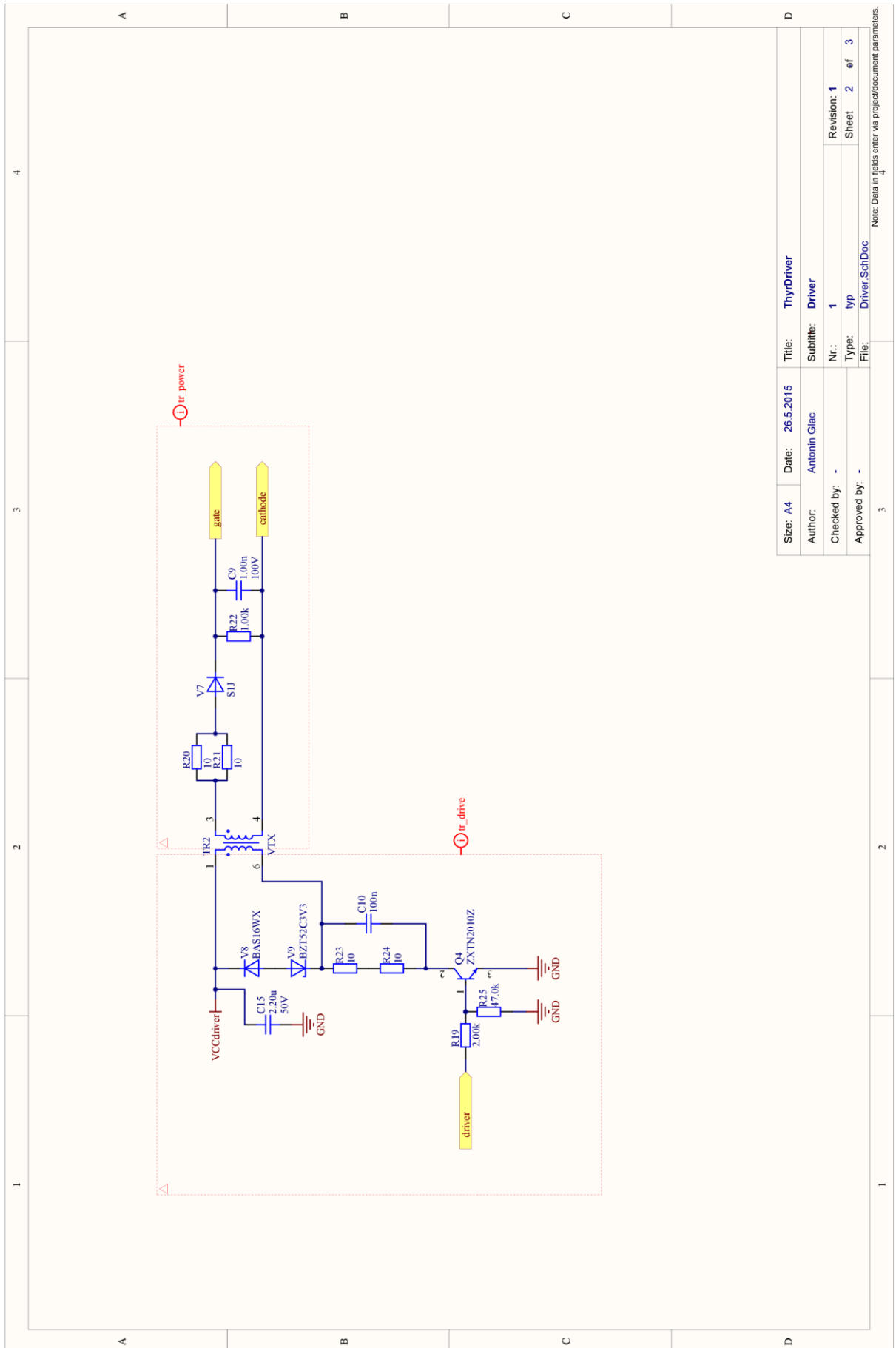


Obr. 2 Spodní strana modulu usměrňovače

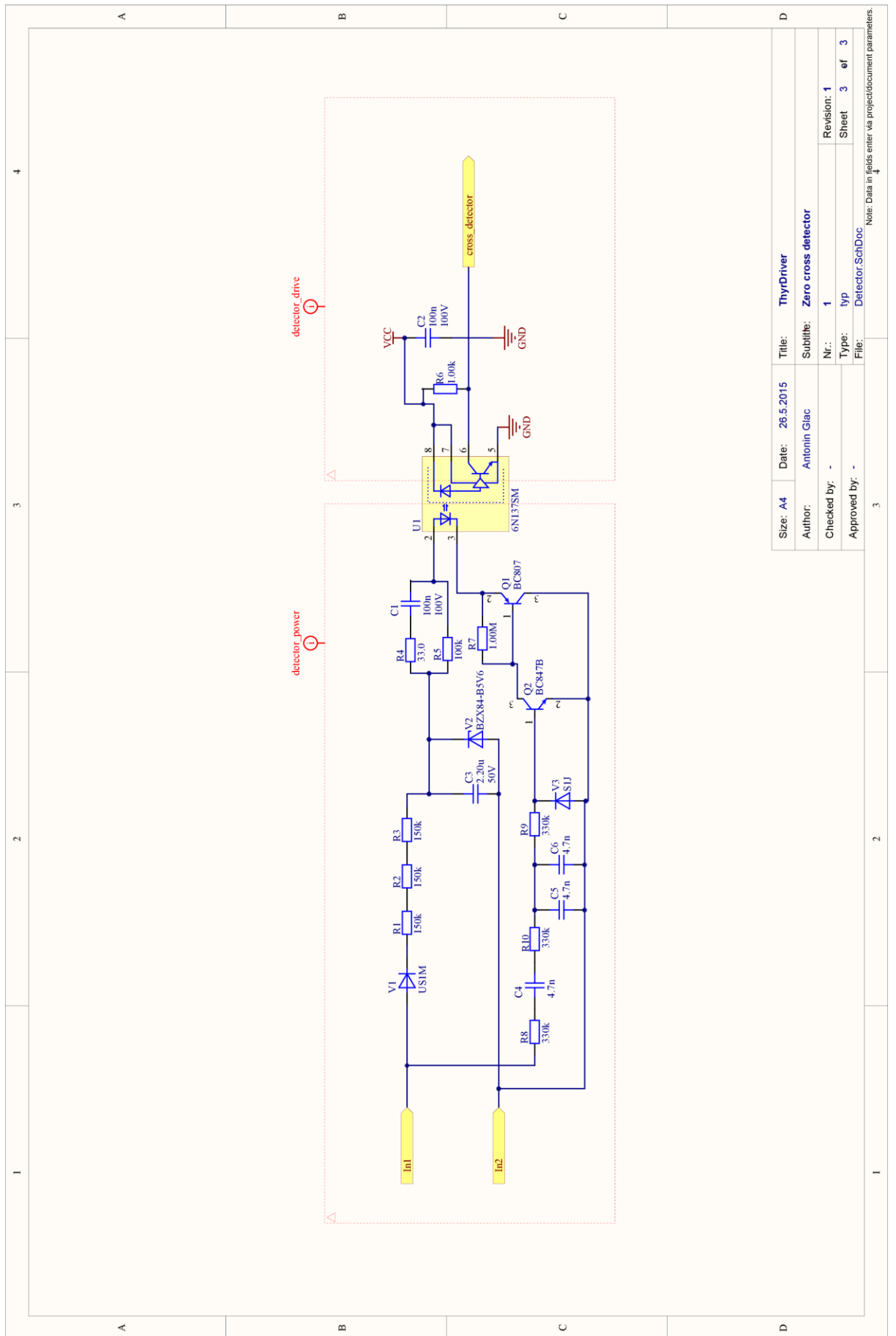


**Příloha B – Výkresová dokumentace**





Size: A4	Date: 26.5.2015	Title: ThyDriver
Author: Antonín Glac	Subtitle: Driver	
Checked by: -	Nr.: 1	Revision: 1
Approved by: -	Type: typ	Sheet 2 of 3
	File: Driver.SchDoc	Note: Data in fields enter via project/document parameters.



Size: A4	Date: 26.5.2015	Title: ThyDriver
Author: Antonin Glac	Subitle: Zero cross detector	
Checked by: -	Nr.: 1	Revision: 1
Approved by: -	Type: typ	Sheet 3 of 3
	File: Detector.SchDoc	

Note: Data in fields enter via project/document parameters.

## Příloha C – Ukázka programu

```

void sledFazi()
{
    PieCtrlRegs.PIEIER1.bit.INTx7 = 0; //vypne interrupt od Timer0 - zablokuje funkci
driveAlfa()
    int OKNorm;
    int OKRev;

    while(1) //smyčka dokud nezjistit správný sled fazi
    {
        OKNorm = 0;
        OKRev = 0;

        while(GpioDataRegs.GPBDAT.bit.GPIO52) //čeká na výstup detektoru L1=0
        ;
        CpuTimer2Regs.TCR.bit.TRB = 1; //reset citace
        CpuTimer2Regs.TCR.bit.TSS = 0; //start citace

        while(GpioDataRegs.GPBDAT.bit.GPIO55) //čeká na výstup detektoru L2=0
        ;
        CpuTimer2Regs.TCR.bit.TSS = 1; //stop citace

        if((CpuTimer2Regs.TIM.half.LSW < 41000 ) && (CpuTimer2Regs.TIM.half.LSW >
39000)) //timer value == 20/3 ms
        {
            reverse = 0; OKNorm++;
        }

        if((CpuTimer2Regs.TIM.half.LSW < 21000 ) && (CpuTimer2Regs.TIM.half.LSW >
19000)) //timer value == 20*2/3 ms
        {
            reverse = 1; OKRev++;
        }

        while(GpioDataRegs.GPBDAT.bit.GPIO52) //čeká na výstup detektoru L1=0
        ;
        CpuTimer2Regs.TCR.bit.TRB = 1; //reset citace
        CpuTimer2Regs.TCR.bit.TSS = 0; //start citace

        while(GpioDataRegs.GPBDAT.bit.GPIO58) //čeká na výstup detektoru L3=0
        ;
        CpuTimer2Regs.TCR.bit.TSS = 1; //stop citace

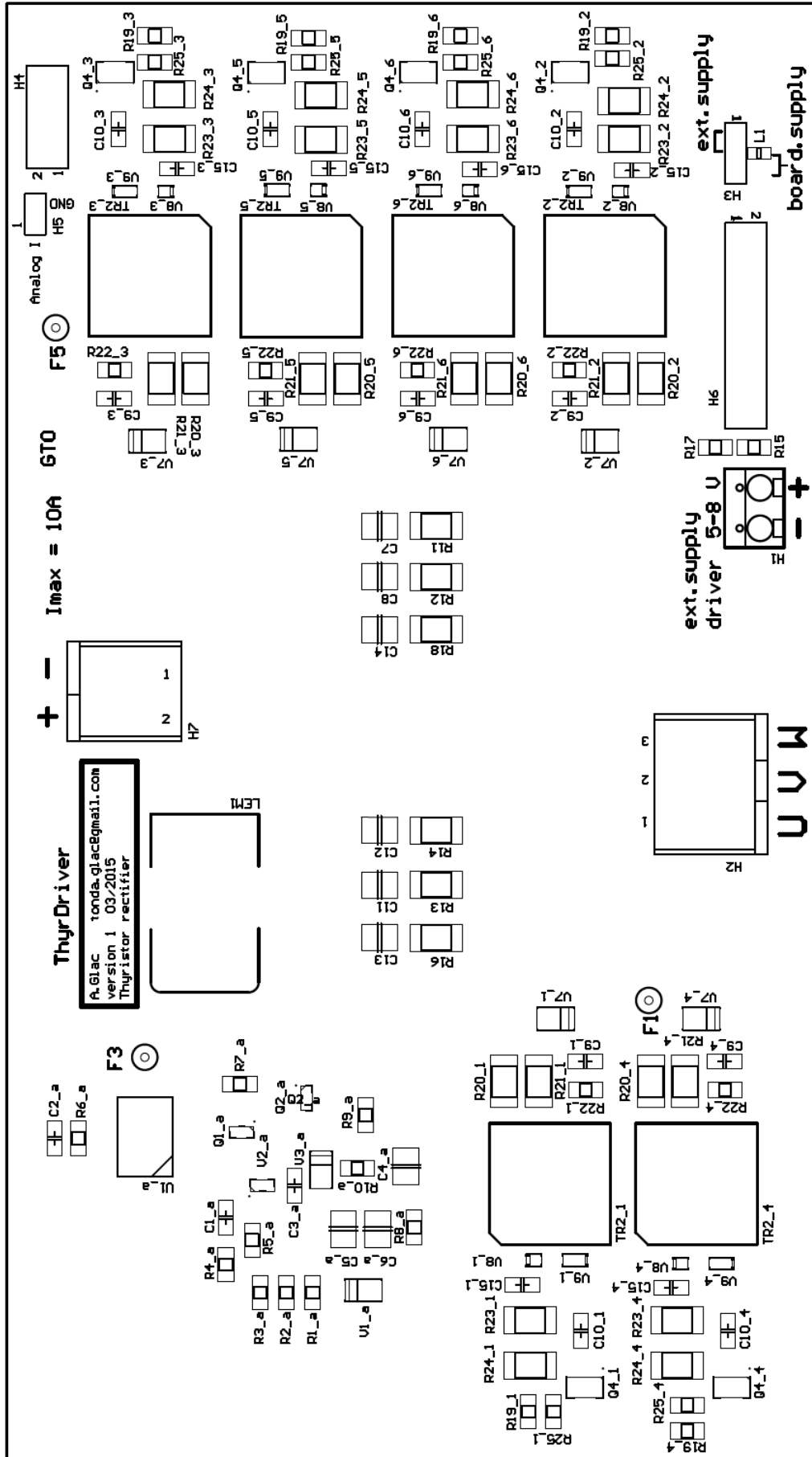
        if((CpuTimer2Regs.TIM.half.LSW < 41000 ) && (CpuTimer2Regs.TIM.half.LSW >
39000)) //timer value == 20/3 ms
        {
            reverse = 1; OKRev++;
        }

        if((CpuTimer2Regs.TIM.half.LSW < 21000 ) && (CpuTimer2Regs.TIM.half.LSW >
19000)) //timer value == 20*2/3 ms
        {
            reverse = 0; OKNorm++;
        }

        if((OKNorm == 2) && (reverse == 0)) //zjistěn normalní sled fazi
            break;
        if((OKRev == 2) && (reverse == 1)) //zjistěn opačný sled fazi
            break;
    }
    CpuTimer1.InterruptCount = 0; //vynuluje citac synchro
    PieCtrlRegs.PIEIER1.bit.INTx7 = 1; //zapne interrupt Timer0 -odblokuje driveAlfa()
}

```

**Příloha D – rozmístění součástek na horní straně DPS**



## **Příloha E – Obsah příloženého CD**

- text bakalářské práce ve formátu PDF a DOCX
  
- zdrojové kódy
  
- výkresová dokumentace
  
- návrh DPS v Altium Designer