

# Obsah

<b>Úvod</b>	<b>4</b>
<b>1 Základní teorie akvaristiky</b>	<b>5</b>
1.1 Historie . . . . .	5
1.1.1 Počátky . . . . .	5
1.1.2 Věda a technika . . . . .	5
1.2 Rozdelení akvárií . . . . .	6
1.3 Technické vybavení akvária . . . . .	6
1.3.1 Filtrace vody . . . . .	7
1.3.2 Osvětlení . . . . .	7
1.3.3 Ohřev . . . . .	9
1.3.4 Monitorování . . . . .	10
1.3.5 Dostupná komplexní řešení . . . . .	11
<b>2 Systémový návrh</b>	<b>15</b>
2.1 Požadavky . . . . .	15
2.2 Blokové schéma . . . . .	16
2.3 Komunikační rozhraní . . . . .	17
2.3.1 Výběr datové sběrnice . . . . .	18
2.3.2 Sběrnice CAN . . . . .	19
<b>3 Návrh řídicí jednotky</b>	<b>20</b>
3.1 Mikrokontrolér . . . . .	20
3.2 Návrh zapojení a tvorba DPS . . . . .	21
3.2.1 Zapojení ESP32 modulu . . . . .	21
3.2.2 Napájecí obvod . . . . .	22
3.2.3 Deska plošných spojů . . . . .	24
3.3 Konektivita . . . . .	27
<b>4 Obecný modul periferie</b>	<b>29</b>
4.1 Mikrokontrolér . . . . .	29
4.2 Návrh zapojení a tvorba DPS . . . . .	29
<b>5 Volba a návrh periferií</b>	<b>31</b>
5.1 Senzor teploty . . . . .	31
5.1.1 Metody měření teploty . . . . .	31
5.1.2 Realizace sensoru . . . . .	32
5.2 Senzor výšky hladiny . . . . .	33

5.2.1	Popis zvolených sensorů . . . . .	34
5.3	LED osvětlení . . . . .	35
5.3.1	Návrh zapojení . . . . .	35
5.3.2	Tvorba DPS . . . . .	39
5.4	Senzor pH . . . . .	40
5.5	Ovládání 230V periferií . . . . .	40
<b>6</b>	<b>Software</b>	<b>42</b>
6.1	Architektura . . . . .	42
6.2	Popis CAN komunikace . . . . .	43
6.2.1	Adresace . . . . .	44
6.3	Firmware řídicí jednotky . . . . .	44
6.3.1	FreeRTOS . . . . .	45
6.3.2	Indikace stavu zařízení . . . . .	45
6.3.3	Popis jednotlivých tasků . . . . .	46
6.4	Firmware periferií . . . . .	48
6.4.1	Zpracování příkazů CAN . . . . .	49
6.4.2	Smyčka programu . . . . .	50
6.5	Webové rozhraní . . . . .	50
<b>7</b>	<b>Sestavení a testování</b>	<b>52</b>
<b>Literatura</b>		<b>53</b>
<b>Seznam symbolů a zkratek</b>		<b>57</b>
<b>Seznam příloh</b>		<b>59</b>
<b>A Schéma a návrh DPS řídicí jednotky</b>		<b>60</b>
A.1	Blokové schéma . . . . .	60
A.2	Zapojení MCU . . . . .	61
A.3	Napájecí obvod . . . . .	62
A.4	Konektory . . . . .	63
A.5	Sběrnice periferií . . . . .	64
A.6	Vodivé motivy DPS . . . . .	65
<b>B Schéma a návrh DPS modulu periferií</b>		<b>67</b>
B.1	Schéma zapojení . . . . .	67
B.2	Vodivé motivy DPS . . . . .	68

<b>C Schéma a návrh DPS modulu LED osvětlení</b>	<b>70</b>
C.1 Schéma zapojení . . . . .	70
C.2 Vodivé motivy DPS . . . . .	71

# Úvod

V dnešní době, kdy jsou na vzestupu fenomény jako chytrá domácnost, IoT (Internet of Things) nebo Průmysl 4.0, se na trhu objevuje stále více výrobků, jejichž úkolem je automatizovat a zjednodušit různé oblasti našeho života. Tento trend se dnes dotýká nejedné volnočasové aktivity, a to včetně akvaristiky. Tu lze samozřejmě provozovat na různé úrovni, ale i majitelé malých domácích akvárií potřebují k provozu svého koníčku relativně velké množství elektroniky. Běžnou praxí je, že každé z použitých zařízení je ovládáno buďto zcela ručně nebo, pokud disponuje možností vzdáleného přístupu a automatizace, má svou samostanou aplikaci a uživatel tak provoz akvária musí ovládat z několika různých míst, což může být značně nepohodlné a nepřehledné.

Na trhu samozřejmě existují také velmi sofistikované a komplexní systémy, ty ovšem svou cenou vysoce přesahují rozpočet běžného „domácího“ akvaristy. Tato práce se věnuje návrhu a tvorbě zařízení, které má za cíl nabídnout pohodlnou kontrolu a ovládání všech potřebných součástí domácího akvária, a to při zachování jednoduchosti a nízké pořizovací ceny.

# 1 Základní teorie akvaristiky

Tato kapitola je teoretickou částí práce, která předchází návrhu samotného zařízení. Na základě dostupných zdrojů jsou zde rozebrány technické požadavky na provoz akvária a proveden průzkum trhu v oblasti akvaristiky a její automatizace.

## 1.1 Historie

Pro lepší orientaci čtenáře ve trendech vývoje akvaristiky se práce nejprve zaměřuje na krátké shrnutí její historie. Kapitola slouží také pro pochopení motivace akvaristů k rozvoji používané techniky.

### 1.1.1 Počátky

Akvaristika v různých podobách provází lidstvo téměř od prvopočátku. Nejprve se jednalo spíše o chov ryb užitkových, tedy rybářství, ovšem už ve starověké Mezo-potámii docházelo také k chovu ryb okrasných. Počátky akvaristiky byly prováděny spíše metodou pokusů a omylů, protože lidem nebyla známa velká část přírodních zákonitostí – životní potřeby chovaných ryb, způsob jejich rozmnožování a v neposlední řadě také procesy, odehrávající se v přírodním ekosystému, zajišťující jeho rovnováhu. Základem udržení chovaných ryb naživu byla zejména častá výměna vody, ani tak ale dlouho nebylo možné udržet ryby při životě dlouhodobě.

V období středověku se poprvé objevuje také dovoz exotických okrasných rybek z cizích zemí, pro naprostý nedostatek znalostí ale často brzy hynou, např. jen proto, že chovatele nenapadne je nakrmit [1].

### 1.1.2 Věda a technika

Na konci 18. století dochází k rozvoji vědy a několika objevům, které historii akvaristiky zásadně ovlivnily. Poprvé byl izolován kyslík, byl objasněn princip dýchání živočichů a následně také fotosyntéza. Akvaristika, v tehdejší době umělý chov ryb za účelem pozorování a výzkumu, byla provozována zejména na vědecké půdě a byl zde zájem o zdokonalení používaných technik a postupů. V roce 1837 S. H. Ward prakticky prokázal, že osvětlené akvárium obsahující jak rybky, tak i rostliny, vydrží velmi dlouho bez nutnosti výměny vody [1]. Princip výměny plynů byl významným milníkem ve snaze dosáhnout v akváriu rovnováhy podobné přírodnímu prostředí.

Při stále nových poznatcích o životních potřebách ryb a o akvarijní rovnováze bylo nutné přijít s různými technickými řešeními. Akvária 19. a 20. století už byla vytápená a uměle okysličovaná. Původní mechanická řešení a lihové kahany byly

postupně nahrazovány elektrickými přístroji. V pozdějších letech pak přibylo i umělé osvětlení a systémy filtrace vody.

## 1.2 Rozdělení akvárií

Akvária je možné rozdělit na základě mnoha různých parametrů jako je např. velikost, materiál a tvar anebo jejich funkce. Pro účely této práce jsou však relevantní zejména rozdělení, která jsou zásadní pro rozsah použité akvaristické techniky.

V jednoduchosti lze tedy akvária rozdělit podle biotopu [2]:

- Sladkovodní
- Brackická – salinita přibližně 5 až 15 ‰
- Mořská – salinita přibližně 30 až 40 ‰

Asi není potřeba vysvětlovat, že pro akvária mořská a brackická nestačí použít běžnou kohoutkovou vodu, ale je potřeba ji před použitím upravit. Má-li být systém automatizován, je nutné přidat zařízení, které bude salinitu průběžně monitorovat a upravovat. Komplexní profesionální systémy tyto možnosti nabízejí, ale pořizovací cena je relativně vysoká (viz kapitola 1.3.5). Lze tedy konstatovat, že po technické stránce je provoz sladkovodních akvárií jednodušší než provoz akvárií mořských.

Další dělení akvárií je možné z hlediska jejich obsazení:

- Čistě rostlinná akvária
- S běžnými druhy ryb
- Se speciálními druhy – zvýšené nároky na parametry vody

Rozsah použité akvaristické techniky a zejména požadavek na její přesnost je závislý na volbě umístěných druhů rostlin a živočichů. Každý druh má své optimální životní podmínky a zatímco některým živočichům se bude dařit ve vodě o teplotě v rozsahu klidně i  $10^{\circ}\text{C}$ , jiné vyžadují téměř konstantní teplotu v rozsahu třeba jen  $2^{\circ}\text{C}$  [3], to zásadně ovlivní požadavky na přesnost měření teploty i způsob její regulace. Stejně tak je tomu i s dalšími parametry.

## 1.3 Technické vybavení akvária

V této kapitole je uveden výčet základní akvaristické techniky nutné k provozu domácího akvária, rozčleněné podle svého účelu. Ve druhé části se text věnuje přehledu různých dostupných komplexních systémů zaměřujících se na automatizaci provozu akvária. Cílem kapitoly je seznámit čtenáře blíže s problematikou založenou a provozu akvária a různými možnostmi technického zajištění jak domácích, tak i profesionálních akvárií.

### 1.3.1 Filtrace vody

Úkolem filtru je průběžně odstraňovat z vody nečistoty, a to jak mechanické, tak zejména v podobě škodlivých látek vznikajících v akváriu. Filtrační materiál je volen tak, aby tvořil vhodné prostředí pro život filtračních bakterií, které se těmito škodlivými látkami živí [4]. Rozlišujeme tři základní typy akvarijních filtrů – vnější, vnitřní a závěsné. Na obr. 1.1 se nachází ukázka vybraných zástupců jednotlivých typů.

**Vnějším filtrem** se rozumí zařízení umístěné obvykle ve skřínce pod akváriem, mívá připojeny dvě hadice – na vstup a výstup vody. Toto řešení je považováno za nejlepší, protože filtr není omezen rozměry a může tak dosahovat daleko vyššího výkonu a účinnější filtrace díky většímu množství filtračních materiálů.

**Vnitřní filtr** (někdy také ponorný) je levným, ale nepříliš účinným řešením pro malá akvária. Nachází se z velké části v akváriu a za pomocí motorku tlačí vodu přes obvykle molitanovou náplň.

**Závěsný filtr** je kompromisním řešením. Cenou i účinností filtrace se pohybuje mezi oběma zmíněnými typy. Nezabírá prostor uvnitř akvária a může tak využít větší objem filtrační hmoty než filtr vnitřní. Instalace je provedena zavěšením na stěnu akvária, je tedy velmi jednoduchá.



SUNSUN HJ-752

*vnitřní filtr*

SUNSUN CBG-500

*závěsný filtr*

Oase BioMaster 350

*vnější filtr*

Obr. 1.1: Příklad různých typů filtrů. Převzato z [5].

### 1.3.2 Osvětlení

Funkce osvětlení akvária je dvojí. Jednak jde o estetický dojem z pohledu pozorovatele, kdy vhodné nasvícení přidává akváriu na atraktivitě. Druhak se osvětlení

snaží nasimulovat osazenstvu akvária přirozené životní podmínky, aby celý ekosystém mohl fungovat.

Hlavními parametry při výběru svítidla jsou jeho **intenzita**, **spektální charakteristika** a **spotřeba**.

Příliš intenzivní světlo zvyšuje riziko nežádoucí tvorby řas a pro ryby může být stresovým faktorem, nízká intenzita zase může způsobit špatný růst rostlin [6]. Na internetu existuje mnoho návodů a rad na stanovení správné intenzity, ale protože zde hraje roli spousta dalších parametrů jako např. výška hladiny nebo konkrétní typ rostlin, je vhodné tyto hodnoty brát pouze jako orientační a intenzitu osvětlení upravit během provozu podle potřeby. Výpočet se také liší pro jednotlivé typy svítidel.

Spektrum světla hraje roli hned z několika důvodů. Rostliny pro tvorbu chlorofylu a následnou fotosyntézu potřebují světlo zejména vlnových délek 440 nm (modrá barva) a 660 nm (červená barva) [7], pokud by zvolené osvětlení tyto vlnové délky neobsahovalo, nemohou rostliny správně fungovat. Akvárium osvětlené pouze těmito dvěma barvami by ale nevypadalo vizuálně dobře, proto se využívá také širokospetrální bílé světlo, které svým spektrem odpovídá co nejlépe dennímu světlu. Specializovaná svítidla pak nabízejí možnost napodobit světelné spektrum různých vodních prostředí a přizpůsobit se tak i rostlinám a živočichům žijícím ve velkých hloubkách.



Obr. 1.2: Trubice s LED páskem a manuální stmívač a časovač. Převzato z [5].

Na trhu jsou v současné době tři typy akvaristických světel: **zářivky**, **výbojky** a **LED svítidla** [8]. Zářivky jsou považovány za dnes již nepříliš moderní řešení a bývají nahrazovány LED svítidly, ty se vyznačují lepší účinností (tedy nižší spotřebou energie při stejně intenzitě světla), delší životností a širší paletou barev. U zářivek také nebylo možné plynule regulovat intenzitu, jako je tomu u LED, a dosáhnout

tak např. postupného rozsvícení nebo zhasnutí světla simulujícího východ a západ slunce. Skoková změna při zapnutí nebo vypnutí světla je pro ryby také zbytečným stresovým faktorem [3]. Co se týče výbojek, ty nacházejí uplatnění zejména pro hluboké nádrže, protože jejich světlo je bodové a intenzita dostačující k prosvícení velkého objemu vody, spotřeba energie je ale v porovnání s LED vysoká, takže pokud to není nezbytně nutné, je lepší se jim vyhnout.

Typické domácí akvárium je osvětleno jedním nebo několika samostatně stmívatelnými LED svítidly, a to buďto v podobě LED pásků nalepených na hliníkovém profilu anebo hotového svítidla, ve kterém jsou čipy s LED zabudovány napevno. Stmívání je nastavováno buď ručně anebo za pomocí mobilní aplikace dodané výrobcem stmívače. Příklad běžně dostupného výrobku lze vidět na obr. 1.2.

### 1.3.3 Ohřev

Většina okrasných sladkovodních ryb běžně chovaných akvaristy pochází z tropických krajů a vyžaduje teplotu vody v rozmezí 22 – 26 °C [9], to je o něco málo vyšší teplota než bývá v domácnosti typická a proto je nutné zajistit akváriu možnost dodatečného ohřevu. Nejčastějsím řešením je ponorné topné těleso na odporové bázi s vlastní termostatickou regulací, viz obr. 1.3.



Obr. 1.3: SUNSUN topítko 100W s termostatem. Převzato z [5].

Z principu fungování termostatu vyplývá, že výsledná teplota vody není v čase konstantní, ale osciluje okolo nastavené hodnoty. Rozsah kolísání teploty je pak závislý na hysterezi termostatu, obecně lze říci, že to může být i několik stupňů. Pro většinu aplikací to není velký problém, ale některé druhy ryb mohou být na změny

teploty náchylnější, v takovém případě je potřeba buďto vybrat topítko takové, kde výrobce rozsah teplot uvádí, anebo zvolit jiný způsob regulace.

### 1.3.4 Monitorování

Jak již vyplynulo z úvodních kapitol, v akváriu probíhá celá řada procesů ovlivňujících jeho stav. Klíčem k vytvoření prosperujícího akvária je dosažení rovnováhy a stability mezi nimi za pomoci vhodně nastavené akvarijní techniky. Nejen u začínajících akvaristů se mohou vyskytnout problémy s růstem rostlin, zdravím ryb nebo třeba výskytem řasy. Odhalit příčiny těchto problémů může být mnohdy obtížné, ovzálště pokud není k dispozici dostatečné množství informací o tom, co se v akváriu děje.

Existuje několik veličin, které úzce souvisí s procesy v akváriu a které je možné také poměrně jednoduše sledovat. Na trhu je celá řada produktů sloužících k tomuto účelu. Většinou je na výběr možnost analogového nebo čistě mechanického přístroje případně samostatného digitálního čidla, existují ale také komplexní řešení, těm se dále věnuje kapitola 1.3.5.

#### Teplota

Umístěním teploměru (ať už v analogové nebo digitální podobě) do akvária je možné zkontolovat správné nastavení topného tělesa a následně provést jeho úpravu. Také lze včas získat informaci o jeho případné poruše a nebo třeba jen nedostatečném výkonu.

#### pH a CO<sub>2</sub>

Hodnota pH popisuje kyselost resp. zásaditost měřeného vodného roztoku. Běžně se používá logaritmická stupnice s hodnotami 0 až 14, přičemž zcela neutrální voda má pH rovno 7, menší hodnoty mají roztoky kyslé a větší než 7 pak roztoky zásadité. Obecně lze říci, že pro ryby je vyhovující pH v rozsahu 6 až 8 [9].

Důležitým parametrem vody z pohledu rostlin a ryb je koncentrace CO<sub>2</sub>. Přirozeně platí, že rostliny CO<sub>2</sub> spotřebovávají při fotosyntéze a jistá koncentrace je tedy nutná pro jejich prosperitu, naopak příliš vysoká koncentrace může být nebezpečná pro ryby, kterým (obdobně jako např. lidem) komplikuje dýchání. Obsah CO<sub>2</sub> ve vodě je obtížné přímo měřit, jeho měnící se koncentrace má ale vliv právě na hodnoty pH, s rostoucí koncentrací CO<sub>2</sub> se pH vody snižuje a obráceně [10, 6].

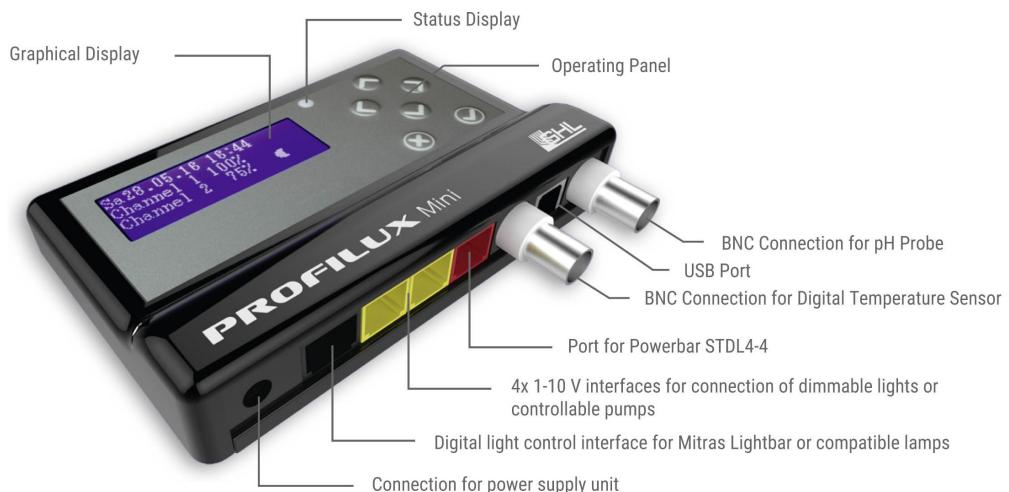
K měření pH vody se používají různé chemické testy (kapkové testy, testovací papírky), které je možné zakoupit v chovatelských potřebách. Z pohledu automatizace je mnohem zajímavějším řešením pH sonda, která umožňuje nepřetržité měření

této veličiny a případnou okamžitou regulaci dávkování CO<sub>2</sub>.

### 1.3.5 Dostupná komplexní řešení

Tato sekce se věnuje porovnání několika nejznámějších systémů v oblasti automatizace akvárií. Je důležité připomenout, že ve všech oblastech elektrotechniky dochází k rychlému rozvoji a každý rok se na trhu objevují nové produkty se stále lepšími parametry a nižší cenou. Informace uvedené v této kapitole, a to zejména cenové údaje, se mohou velmi rychle stát neaktuálními a jsou tedy relevantní pouze v době vzniku této práce.

Při tvorbě této kapitoly byly jako zdroj informací použity jednak oficiální materiály výrobců, ty ovšem samozřejmě obsahují vždy pouze pozitivní informace, dále pak různé uživatelské recenze na platformě YouTube popř. diskuzních fórech, nejedná se o zcela seriózní zdroje a proto je nutné také informace z této kapitoly brát s rezervou.



Obr. 1.4: GHL ProfiLux Mini, nejmenší dostupný kontrolér této firmy. Převzato z [11].

### GHL – ProfiLux

Německá firma GHL se v oblasti akvaristiky pohybuje již přes 20 let a patří nejpochybně ke špičce na trhu z hlediska komplexity a spolehlivosti. Základem jejich systému ProfiLux je kontrolér (např. nejmenší varianta viz obr. 1.4), který je možné konfigurovat z PC za pomocí kabelu anebo vzdáleně s použitím aplikace nebo webového rozhraní. Ke kontroléru lze připojit celou řadu periferií z portfolia firmy, jedná se o různé typy senzorů, dávkovače (pro úpravu parametrů vody), pumpy nebo řidičelný prodlužovací přívod pro síťové zásuvky (laicky řečeno „chytrá prodlužovačka“).

Společnost si zakládá na opravdu vysoké kvalitě a přesnosti svých výrobků, což se ale odráží také na jejich ceně.

Na výběr je z několika variant systému, přičemž ty nejdražší dokážou obsloužit i opravdu rozsáhlé a náročné akvaristické instalace. Cena nejlevnějšího základního setu je přibližně od 10 000 Kč [11, 12].

### **Neptune Systems – Apex**

Systém Apex je nepochybně další ze světových leaderů v této oblasti. Opět je k dispozici několik variant systému podle požadavků a finančních možností uživatele a systém je také velmi modulární. Stejně jako firma GHL, i Neptune Systems je na trhu více než 20 let a jedná se tedy o léty ověřenou značku. Architektura systému je podobná a kromě samotného kontroléru je opět v nabídce celá řada kompatibilních periferií. Dle uživatelských recenzí je konfigurace systému oproti GHL výrazně jednodušší a není nutná znalost programování, navíc systém už od výroby obsahuje přednastavené nejčastější scénáře použití.



Obr. 1.5: Neptune Systems Apex EL, základní set. Převzato z [13].

Cena opět závisí na množství zakoupených modulů, základní set s podobnou výbavou jako u GHL je k dispozici přibližně od 12 000 Kč [14, 13].

### **CoralVue – HYDROS**

Firma CoralVue se svým systémem HYDROS je na trhu oproti svým konkurentům relativně krátce, přibližně 3 roky, svým originálním přístupem a cenově dostupným řešením si ale své zákazníky našla rychle. Systém je svou architekturou ještě více modulární než jeho konkurenti, umožňuje v rámci jedné aplikace spojit i více kontrolérů, které mezi sebou komunikují. Dokonce v případě poruchy hlavního kontroléru dokáže jeho roli převzít jiný připojený kontrolér a systém tak zůstane dál v provozu.

Kromě bezdrátově řízeného modulu se čtyřmi síťovými zásuvkami nově firma nabízí také modul Control XP8, který krom zásuvek obsahuje i vlastní kontrolér, může

tak fungovat zcela samostatně, stále však umožňuje také drátové spojení s dalšími kontroléry nebo bezdrátové připojení k dalším zásuvkám. Toto může sloužit jako jednoduché univerzální řešení pro menší akvária s možností budoucího rozšíření.



Obr. 1.6: CoralVue HYDROS Control X2 Starter pack. Převzato z [15].

Základní minimální sada je dostupná již od přibližně 4500 Kč, aby byla ale výbava stejná jako u výše zmíněných konkurentů je potřeba dokoupit ještě pH sondu za přibližně 800 Kč [16, 15].

### **Seneye**

Společnost Seneye nenabízí komplexní řešení pro automatizaci, ale i přesto jsou její produkty zajímavé a pro mnoho akvaristů mohou být skutečně užitečné. Místo pokročilého ovládání akvarijní techniky se výrobky zaměřují pouze na monitorování parametrů vody (popř. dalších veličin), důraz je kladen na maximální jednoduchost použití. Vnitřním sladkovodním akváriím je věnována řada Seneye Home a veškeré monitorování je zajištěno jedním malým zařízením, které uživatel přímo ponoří do vody a pomocí kabelu připojí k počítači ze kterého se zařízení napájí a zároveň do něj odesílá data. Alternativně lze příkoupit také krabičku, která slouží jako webserver, do ní se zařízení připojí namísto počítače a data jsou rovnou zálohována do cloudu, odkud jsou uživateli dostupná v mobilní aplikaci.

Zařízení monitoruje teplotu, pH, úroveň škodlivého amoniaku, osvětlení a hladinu vody. Umožňuje také odesílat oznámení při překročení nastavené meze některého z parametrů.

Cena samotného monitorovacího zařízení je přibližně 3000 Kč a podobná je také cena zmíněného webserveru. Pro automatické monitorování se zálohováním na cloud je tedy potřeba počítat s investicí okolo 6000 Kč [17].



Obr. 1.7: Seneye Home a Seneye Web Server. Převzato z [17].

## 2 Systémový návrh

Tato část práce popisuje proces návrhu vlastního zařízení, které by mělo být výstupem této práce. Věnuje se konkretizaci požadavků na zařízení a koncepčnímu návrhu na systémové úrovni, který je zde podpořen blokovým schématem. Po celou dobu tvorby zařízení je kladen důraz na požadavky stanovené v této kapitole a na jejich základě jsou tvořena vhodná technická řešení. Detailně se jednotlivým blokům a jejich návrhu věnují následující kapitoly.

Zařízení vytvořené v rámci této práce bude určeno pro použití v menším sladkovodním akváriu osazeném běžnými druhy rostlin a živočichů bez speciálních životních potřeb – tedy scénář běžného domácího akvaristy s omezeným rozpočtem. Není ale vyloučeno jeho budoucí rozšíření i pro náročnejší aplikace.

### 2.1 Požadavky

Cílem je vytvořit zařízení, které umožní co nejvíce automatizovat provoz akvária. Hlavním aspektem by měla být jednoduchost použití pro koncového uživatele, vše by mělo být nanejvýš intuitivní a přehledné. Zařízení musí mít možnost připojení k internetu prostřednictvím sítě Wi-Fi, uživatel tak bude moci zařízení konfigurovat a sledovat z libovolného místa za pomoci webové stránky popř. mobilní aplikace.

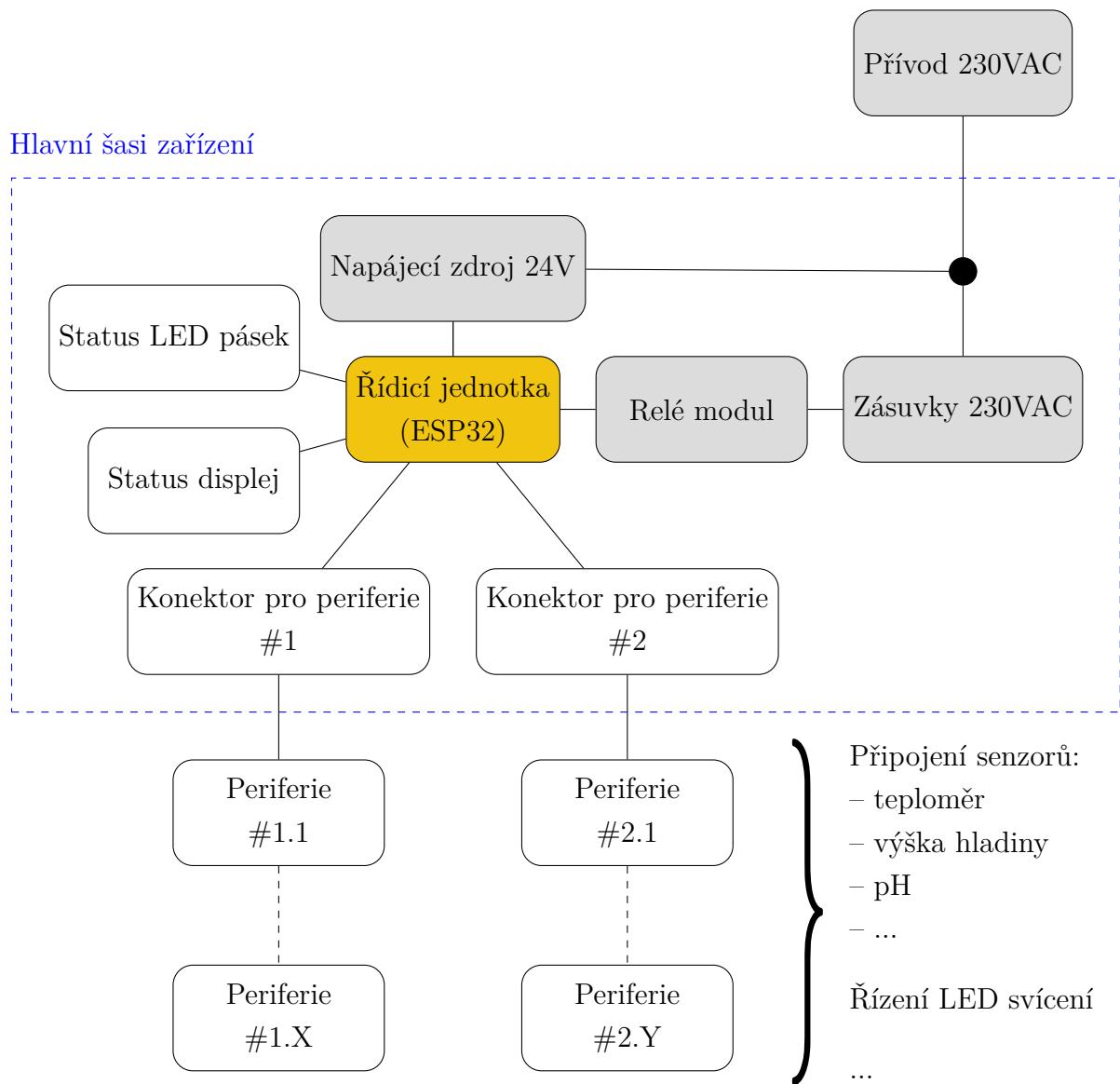
Požadavky jednotlivých akvaristů se mohou lišit a zároveň se v čase měnit. Vytvoření dokonalého a všestaranného zařízení, které vyhoví všem účelům použití není v časových ani finančních možnostech bakalářské práce, proto byl stanoven požadavek, aby bylo zařízení co nejvíce modulární a rozšiřitelné. Musí být zvolena taková architektura, aby bylo možné v budoucnu přidat další funkce a periferie bez nutnosti modifikovat stávající hardware.

Výstupem bakalářské práce by mělo být zařízení schopné monitorovat některé akvaristické veličiny a na základě jejich hodnoty informovat uživatele a ovládat akvárium. Zařízení bude přímo řídit LED páskové osvětlení na 12 V a spínat popř. vypínat již existující akvaristické přístroje pracující se síťovým napětím 230 V.

Jelikož modulární architektura bude nepochyběně vyžadovat použití více než jednoho mikrokontroléru a tedy také více různých firmwarů, je potřeba zajistit jejich vzájemnou kompatibilitu a stabilitu celého systému. Veškerý firmware tak musí být verzovaný a po připojení nové periferie musí řídicí jednotka rozpozнат, o jakou periferii se jedná. V případě připojení nekompatibilní periferie (např. z důvodu zastaralého firmwaru řídicí jednotky) musí být uživatel upozorněn a nesmí být nijak narušena funkce zbytku systému. Aby bylo možné těmto situacím předejít, musí mít řídicí jednotka možnost vzdálené aktualizace firmwaru.

## 2.2 Blokové schéma

Blokové schéma zařízení se nachází na obr. 2.1. Pro pohodlné použití je hlavní část zařízení soustředěna do jedné krabičky napájené přívodním síťovým kabelem. Uživatel pak dle potřeby připojí příslušenství pracující s napětím 230 V do integrovaných síťových zásuvek a veškeré další periferie za pomocí jednoho z univerzálních konektorů. O stavu zařízení bude uživatel informován sérií notifikačních LED a malým displejem.



Obr. 2.1: Blokové schéma systému.

Pro napájení vlastní elektroniky zařízení je v šasi umístěn hotový modul spínáního zdroje převádějící síťové napětí 230 V na stejnosměrných 24 V se kterými pak

zařízení dále pracuje (viz kapitola 3.2.2).

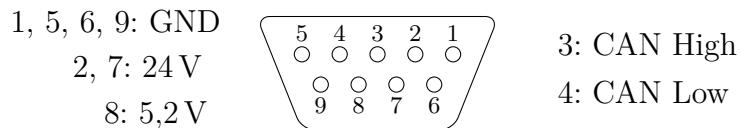
Z hlediska bezpečnosti je potřeba zajistit, aby se uživatel ani samotná nízkonapěťová část obvodu nemohli dostat do kontaktu s nebezpečným napětím. Toho je dosaženo galvanickým oddělením částí zařízení pracujících se síťovým napětím. V blokovém schématu (obr. 2.1) jsou všechny tyto části podbarveny šedou barvou. Pro spínaní síťových zásuvek jsou použita relé, která sama o sobě tvoří galvanickou izolaci, pro ještě lepší ochranu mikrokontroléru je pak zvolena varianta modulu obsahující také optočleny. Pro napájecí zdroj s výstupem 24 V je přítomnost galvanického oddělení kontrolována v dokumentaci výrobce.

## 2.3 Komunikační rozhraní

Před návrhem jednotlivých částí zařízení je zapotřebí definovat komunikační rozhraní mezi řídicím modulem a periferiemi, protože právě od jeho specifikace se následně odvíjí tvorba zbytku zařízení.

Úkolem rozhraní je obousměrně komunikovat s periferiemi, tedy např. stahovat data z připojených sensorů a zároveň za pomocí příkazů periferie řídit. Kromě datové komunikace musí rozhraní periferie také napájet, a to i v případě energeticky náročnějších obvodů jako např. osvětlení.

Pro připojení periferií byl zvolen konektor typu D-sub 9, který je cenově dostupný, disponuje dostatečným množstvím pinů a umožňuje montáž jak do panelu, tak i jako zakončení kabelu. Přiřazení a funkce jednotlivých vodičů jsou vyobrazeny na obr. 2.2. Konektor disponuje dvěma úrovněmi napájení, 5 V slouží k napájení mikrokontrolérů modulů periferií a k nim připojených senzorů nebo jiné nenáročné elektroniky. Je pravděpodobné, že s připojením více periferií za sebe dojde k úbytku napětí v důsledku ztrát na vedení. Aby bylo toto částečně kompenzováno, napětí vystupující z měniče řídicí jednotky je přibližně o 0,2 V vyšší. Druhou napájecí linkou je výstup přímo z externího spínaného zdroje, tedy s napětím 24 V, ten slouží pro výkonově náročnější periferie, které si již napětí dále upraví podle potřeby a nebudou neúměrně zatěžovat první zmíněnou napájecí linku.



Obr. 2.2: Přiřazení pinů konektorů D-sub 9 pro připojení periferií.

### 2.3.1 Výběr datové sběrnice

Existuje celá řada datových sběrnic, které jsou v elektrotechnice hojně využívány. Každá z nich má své výhody a nevýhody, stejně tak jako jisté limitace použití. V tab. 2.1 se nachází výčet různých sběrnic, které byly při výběru uvažovány.

Tab. 2.1: Datové sběrnice, porovnání [18].

Typ	Výhody	Nevýhody	Limitace
SPI	Více zařízení na sběrnici Vysoká rychlosť přenosu dat Jednoduchý protokol	Nutný CS pin pro každé zařízení	Určeno na krátkou vzdálenost
I <sup>2</sup> C	Pouze 2 piny Více zařízení – 128 adres	Riziko kolize adres Nižší rychlosť přenosu dat proti SPI	Určeno na krátkou vzdálenost
CAN	Vysoká spolehlivost Dlouhé propojení	Vyšší náklady na implementaci Nižší rychlosť přenosu dat	Nepodporovano běžnými MCU – nutný externí řadič
UART	Jednoduchá implementace Možnost asynchronní komunikace	Nižší rychlosť přenosu dat proti SPI Pouze 2 zařízení	Pouze 2 zařízení Určeno na krátkou vzdálenost

Hlavní šasi zařízení nabízí dva konektory. Během provozu je ale žádoucí připojit větší, předem nedefinovaný počet periferií. Proto je potřeba, aby zvolená sběrnice umožnila připojení více zařízení současně. Obecný problém všech sběrnic je omezení jejich maximální délky. S rostoucí délkou se sběrnice snáze zaruší, navíc z důvodu parazitních vlastností vedení dochází k zaoblení ostrých hran signálu, dlouhé vedení se chová jako filtr typu dolní propust. V důsledku toho se snižuje maximální rychlosť sběrnice.

Sběrnice SPI nebo I<sup>2</sup>C je obecně doporučeno používat pouze v rámci DPS, tedy na krátké vzdálenosti. Při snížení rychlosti je možné je používat i na větší vzdálenost, ovšem modulární scénář vytvářeného systému teoreticky nestanovuje žádný délkový limit a bylo by velmi obtížné spolehlivě určit, kolik periferií uživatel může za sebe zapojit při zachování spolehlivé komunikace.

UART je výhodný svou jednoduchou implementací a umožňuje obousměrnou asynchronní komunikaci. Nevýhodou je, že funguje pouze pro dvě zařízení. Jednou z

možností, jak tuto limitaci obejít, by bylo zavedení řetězového způsobu komunikace, kdy by každé zařízení komunikovalo se dvěmi sousedními a informace by se postupně předávala dále až k cílovému zařízení. Tento systém je relativně jednoduchý, ale například v případě poruchy jednoho zařízení se odpojí všechna následující zařízení, což může mít neočekávané následky.

Sběrnice CAN je určena pro provoz v průmyslovém prostředí (zejména je používána v automobilovém průmyslu) a díky své robustnější konstrukci ji lze bez problému použít i na delší vzdálenosti a pro více zařízení. Při komunikaci je používán diferenční pár vodičů, takže i odolnost proti rušení je výrazně lepší. Nevýhodou je ale její o něco složitější a dražší implementace. Většina běžných mikrokontrolérů nemá pro CAN vestavěnou periferii a je tak potřeba buďto zvolit dražší mikrokontrolér nebo připojit externí ovladač (řízený např. přes SPI). Dále je nutné přidat i řadič, který převede signál na diferenční a zároveň umožní zvýšit provozní napětí na 12 nebo 24 V, čímž dojde k ještě lepšímu potlačení šumu.

### 2.3.2 Sběrnice CAN

Po důkladné rešerši a zvážení zmíněných kladů a záporů byla zvolena sběrnice CAN. ESP32 jakožto již zvolený mikrokontrolér řídicí jednotky obsahuje vestavený CAN kontrolér a pro moduly periferií byl na základě tohoto rozhodnutí zvolen také vhodný mikrokontrolér. Co se týče nutnosti přidání řadiče, jedná se sice o další součástku, která na první pohled navýšuje cenu zařízení, kromě převodu signálu na diferenční ale zajišťuje také ochranu konektorů proti mnoha nežádoucím jevům jako je zkrat, ESD výboj nebo přepětí. Tímto se ve výsledku celé zapojení zlevní a zjednoduší.

### 3 Návrh řídicí jednotky

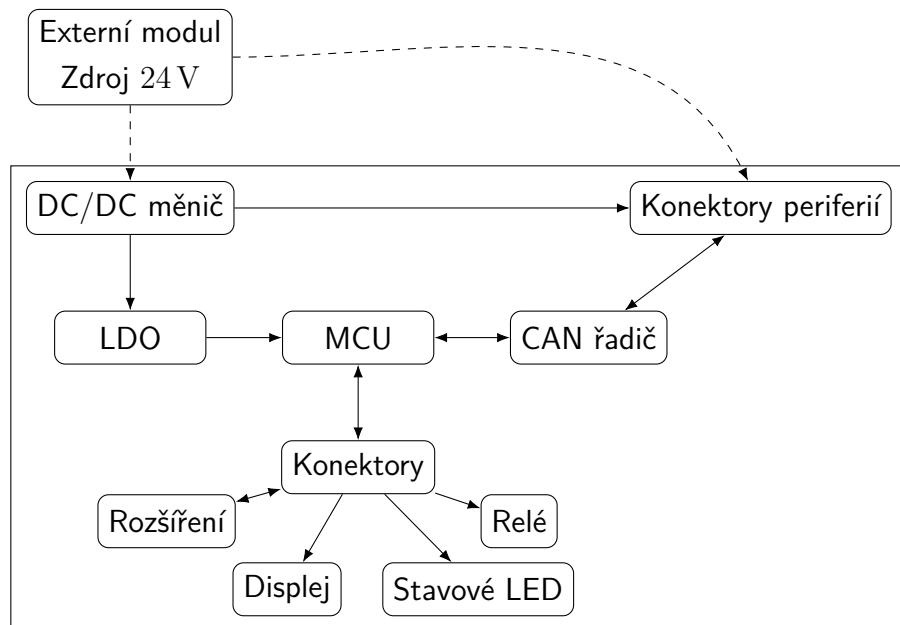
Řídicí jednotka je jádrem celého zařízení. Její funkcí je řízení systému a zároveň komunikace s uživatelem za pomocí Wi-Fi. Musí v sobě nést informaci o konfiguraci systému a na jejím základě zpracovávat data z jednotlivých připojených periferií. Podle uživatelem nastavených scénářů pak dynamicky reaguje na změny hodnot měřených akvaristických veličin a ovládá akční členy (osvětlení, ohřev, filtr vody). Za pomocí displeje a LED pásku také informuje uživatele o momentálním stavu zařízení.

#### 3.1 Mikrokontrolér

Při výběru vhodného mikrokontroléra bylo potřeba zohlednit výše zmíněné požadavky, tedy zejména Wi-Fi konektivitu a dostatečný výkon k její obsluze, periferii CAN a dostatek GPIO pinů pro připojení zbylých modulů v hlavním šasi (viz obr. 2.1). Na trhu existuje více výrobců nabízejících mikrokontroléry s vhodnými parametry, z důvodu jednoduchosti použití a nízké ceny byl nakonec zvolen model ESP32 od firmy Espressif, konkrétně modul WROOM-32E [19] s čipem ESP32-D0WDR2-V3 [20]. Tento modul je často využíván v různých hobby projektech, ale také v komerčních aplikacích zejména v oblasti chytré domácnosti. Z tohoto důvodu k němu existuje velká škála softwarových knihoven a v rámci komunity uživatelů je také sdíleno mnoho projektů, kterými je možné se inspirovat.

## 3.2 Návrh zapojení a tvorba DPS

Řídicí jednotka je tvořena jednou DPS, která kromě samotného mikrokontroléru obsahuje také měnič napětí typu buck ke snížení napájecího napětí externího zdroje na hodnotu 5,2 V. Toto napětí je pak dále používáno pro napájení samotného mikrokontroléru řídicí jednotky a zároveň je vyvedeno na konektor pro připojení periferií. Blokové schéma na úrovni logických bloků v rámci jedné DPS je na obr. 3.1, jednotlivým částem se blíže věnují další kapitoly. Celé schéma je k dispozici v příloze A.



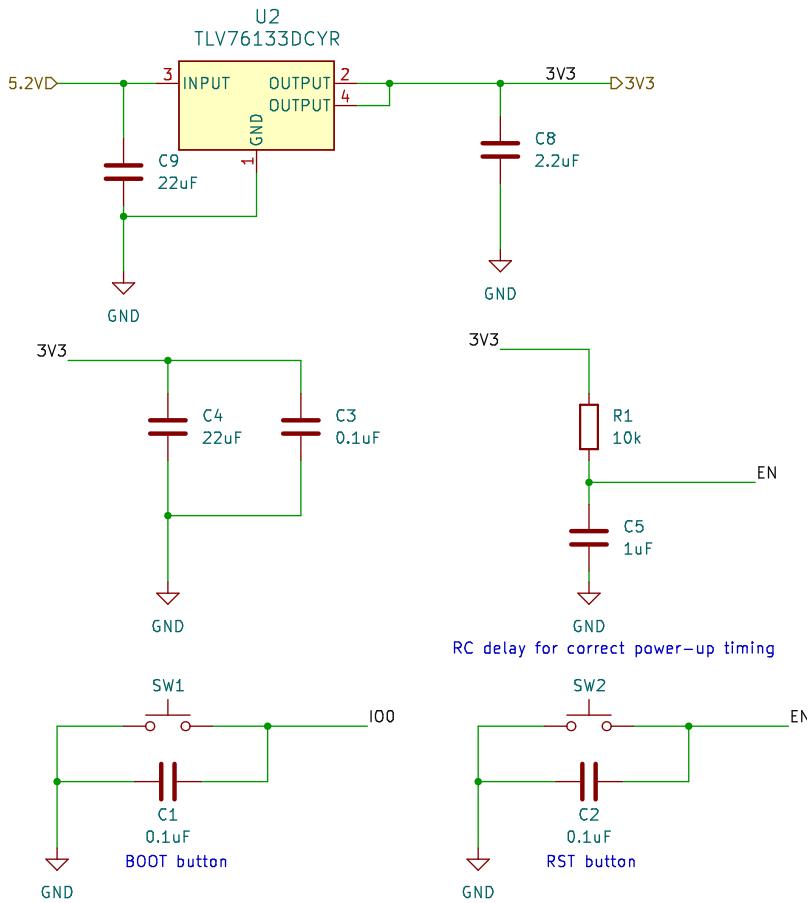
Obr. 3.1: Blokové schéma řídicí jednotky.

### 3.2.1 Zapojení ESP32 modulu

Při tvorbě schématu bylo vycházeno z dokumentace výrobce [19] a také ze schémat různých existujících vývojových desek. K zajištění správné a spolehlivé funkce modulu je potřeba dodržet několik věcí. Výřez schématu obsahující potřebné doplňující obvody pro ESP32 modul je na obr. 3.2.

Na napájecí pin (3V3) je třeba přivést stabilní napětí a opatřit ho blokovacími kondenzátory (C3, C4). Ke snížení napětí z původních 5,2 V na požadovaných 3,3 V je použit lineární regulátor TLV76133 (U2).

Dále je potřeba přivést kladné napětí na povolovací pin (EN). Z dokumentace vyplývá, že by mělo být přivedeno až po ustálení napájecí linky. Uvedený čas nutný ke stabilizaci je roven  $t_{STBL} = 50 \mu\text{s}$  [20]. Požadované zpoždění zajistí RC článek



Obr. 3.2: Podpůrné obvody pro modul ESP32-WROOM-E. Vytvořeno v KiCad 7.0.

$(R_1, C_5)$  s časovou konstantou  $\tau$ :

$$\tau = R_1 C_5 = 10 \text{ k}\Omega \cdot 1 \mu\text{F} = 10 \text{ ms} \quad (3.1)$$

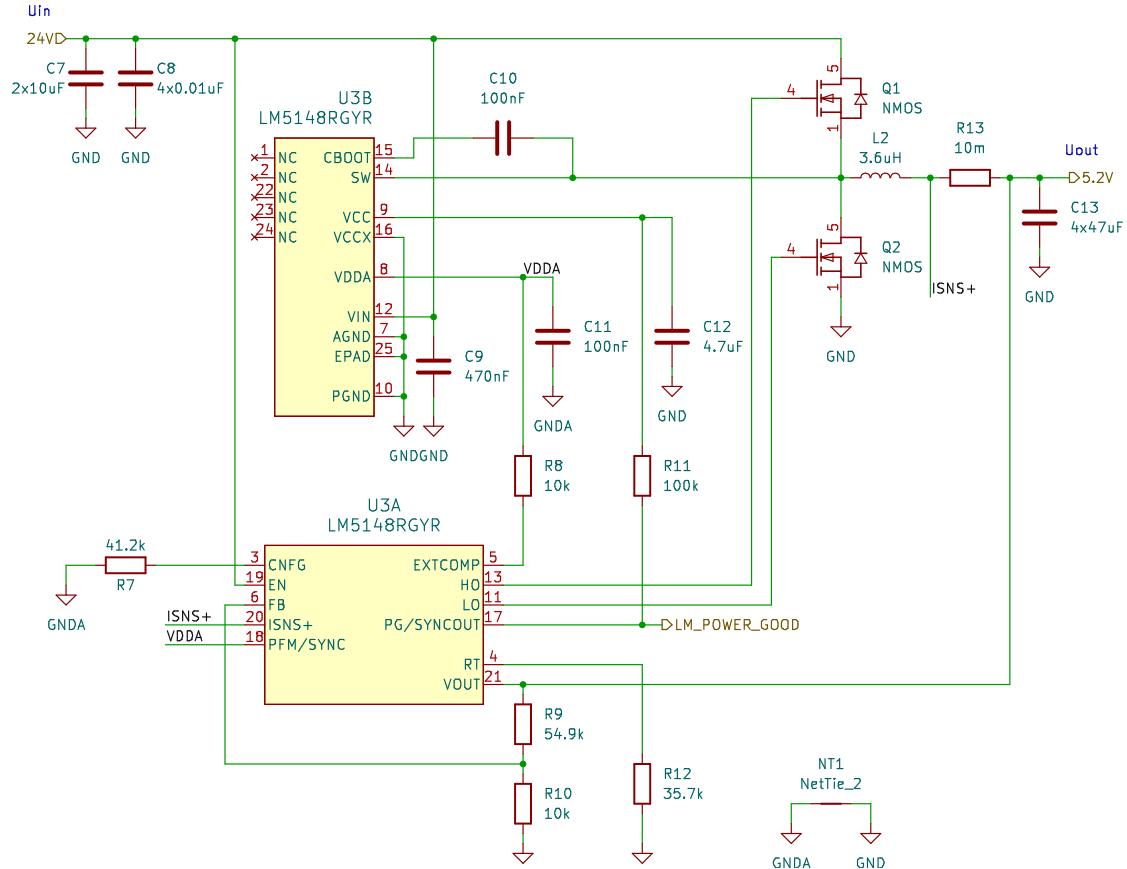
Jak je vidět, byla zvolena dostatečná návrhová rezerva.

Pro možnost resetu zařízení a vstupu do bootloaderu byla doplněna také dvě tlačítka (SW1, SW2).

### 3.2.2 Napájecí obvod

Pro napájení celého zařízení je použit externí zdroj stejnosměrného napětí 24 V a toto napětí je také rozvedeno všem připojeným periferiím. Pro většinu komponent je ale nutné napětí snížit. K tomuto účelu byl navržen DC/DC měnič typu buck s požadovaným výstupním napětím 5,2 V. Existuje celá řada čipů vyvinutých pro tento účel. Aplikace v tomto zařízení je specifická svými požadavky na výstupní proud. Zatímco samotná řídicí jednotka nebude odebírat velký proud, není jasné dané, kolik periferí a s jakými výkonovými požadavky uživatel k systému připojí.

Navržený měnič tak musí fungovat v širším rozsahu proudů (řádově od desítek mA po jednotky A), a to s co nejlepší účinností.



Obr. 3.3: Napájecí obvod řídicí jednotky. Vytvořeno v KiCad 7.0.

Aby bylo vyhověno zmíněným požadavkům a zachována návrhová rezerva, byl jako základ buck měniče zvolen čip LM5148 [21]. Jedná se o moderní součástku firmy Texas Instruments s velkou výkonovou rezervou. Tento čip funguje pouze jako buck kontrolér a zapojení je potřeba doplnit dvěma externími MOSFET tranzistory. Většina tepelných ztrát vzniká právě na nich, čímž se sníží ohřev samotného čipu a generované teplo se lépe rozloží. Na volbě tranzistorů závisí také výsledná účinnost měniče. Při návrhu zapojení této součástky byl použit nástroj Webench Power Designer [22], který podle zadaných parametrů navrhne konkrétní schéma zapojení, provede simulaci a zobrazí grafy upravené na míru zadaným hodnotám. Tento nástroj uvádí přibližnou účinnost zapojení jako 88 %. V navrženém schématu bylo posléze provedeno několik změn, aby vše odpovídalo požadavkům uvedeným v katalogovém listu součástky [21]. Kompletní schéma zapojení spolu s odkazy k relevantním kapitolám katalogového listu se nachází v příloze A.3, pro přibližnou představu pak postačí zjednodušené schéma na obr. 3.3.

### Hodnoty součástek

Označení součástek používané v této kapitole je vztaženo ke zjednodušenému schématu na obr. 3.3. Součástky okolo čipu LM5148 lze rozdělit na dvě skupiny. V prvé řadě ty součástky, které spolu s čipem tvoří jádro buck měniče. Jsou jimi externí tranzistory Q1 (high-side) a Q2 (low-side), induktor L2, vstupní kapacitory (C7,C8) a výstupní kapacitory (C13). Také sem lze zaředit snímací rezistor R13. Pro tyto součástky byly použity hodnoty doporučené výrobcem v katalogovém listu pro danou aplikaci [21]. Druhou skupinu tvoří součástky nutné ke správnému nastavení buck kontroléru. Jejich možné hodnoty pro různá nastavení jsou opět specifikovány v katalogovém listu.

Rezistor R12 slouží k nastavení frekvence spínání, ze zvolené hodnoty vyplývá frekvence:

$$f_{SW}[\text{kHz}] = \frac{10 \cdot 10^6}{45 \cdot R_{12}[\text{k}\Omega] + 53} \quad (3.2)$$

$$f_{SW} = \frac{10 \cdot 10^6}{45 \cdot 35,7 + 53} = 602,59 \text{ kHz} \quad (3.3)$$

Výstupní napětí je dáné poměrem odporového děliče R9 a R10. Na pinu FB je udržováno konstantní napětí  $U_{FB} = 0,8 \text{ V}$ , z této podmínky pak vyplývá vztah pro výstupní napětí:

$$U_{OUT} = U_{FB} \cdot \frac{R_9 + R_{10}}{R_{10}} \quad (3.4)$$

$$U_{OUT} = 0,8 \cdot \frac{54,9 \cdot 10^3 + 10 \cdot 10^3}{10 \cdot 10^3} = 5,192 \text{ V} \quad (3.5)$$

Připojením pull-up rezistoru R8 mezi piny EXTCOMP a VDDA dojde k nastavení interní kompenzace, která je v tuto chvíli dostatečná a není potřeba tvořit kompenzační síť. Pro jistotu byl ale na výsledné DPS vyhrazen prostor i pro součástky kompenzační sítě, viz úplné schéma v příloze A.3.

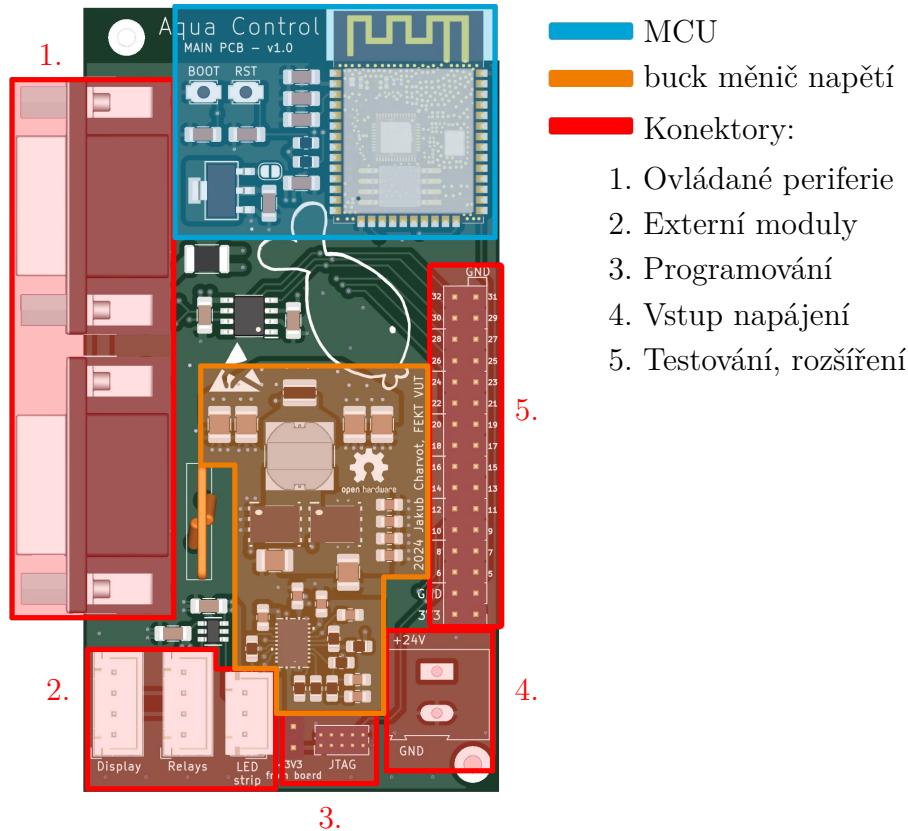
Čip LM5148 nabízí možnost pracovat paralelně v synchronizaci s druhým stejným čipem. Tato funkce zde nebude potřeba, tak stačí synchronizační pin PG/SYNCOUT pull-up rezistorem R11 spojit s napětím VCC, tím se přepne do módu indikátoru správnosti výstupního napětí (PG = Power Good).

Poslední důležitou součástkou je rezistor R7 sloužící pro zbylá nastavení. Hodnota  $R_7 = 41,2 \text{ k}\Omega$  indikuje nastavení módu primárního kontroléru (viz předešlý odstavec). Dále zapříčinuje vypnutí módu Dual Phase a zapnutí použití rozprostřeného spektra (Spread spectrum).

### 3.2.3 Deska plošných spojů

Ačkoliv se jedná o relativně jednoduchou DPS, je potřeba při návrhu dbát jistých pravidel a doporučení. Modul ESP32 je vybaven anténou a volba jeho umístění na

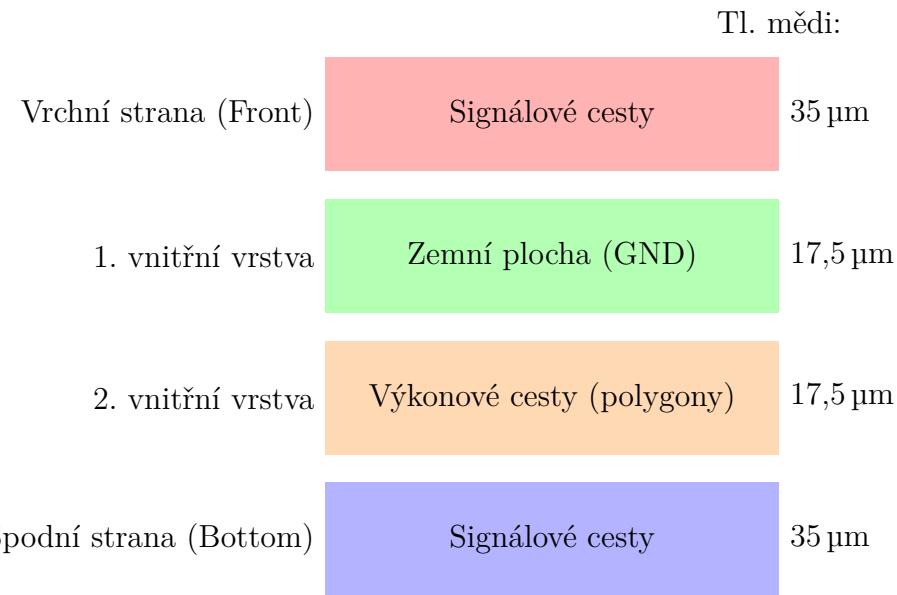
DPS je rozhodujícím faktorem pro správnou funkci antény. Další částí vyžadující optimální návrh rozložení a propojení součástek je pak buck měnič.



Obr. 3.4: Vizualizace DPS řídicí jednotky s vyznačením jednotlivých částí.

Prvním krokem návrhu je volba počtu vrstev a jejich funkce. Vyšší počet vrstev nabízí více prostoru pro vedení cest a také umožňuje vedení napájecích napětí pomocí rozlitých měděných polygonů, čímž se zároveň zlepší vlastnosti zařízení z hlediska EMC!. Zvolený výrobce (JLC PCB [23]) nabízí výrobu desek s jednou až dvaceti vrstvami mědi. Byla zvolena čtyvrstvá deska, která je pro danou aplikaci dostatečná a stále se nachází v přijatelné cenové skupině výrobce. Rozložení a funkce vrstev jsou vyobrazeny na obr. 3.5.

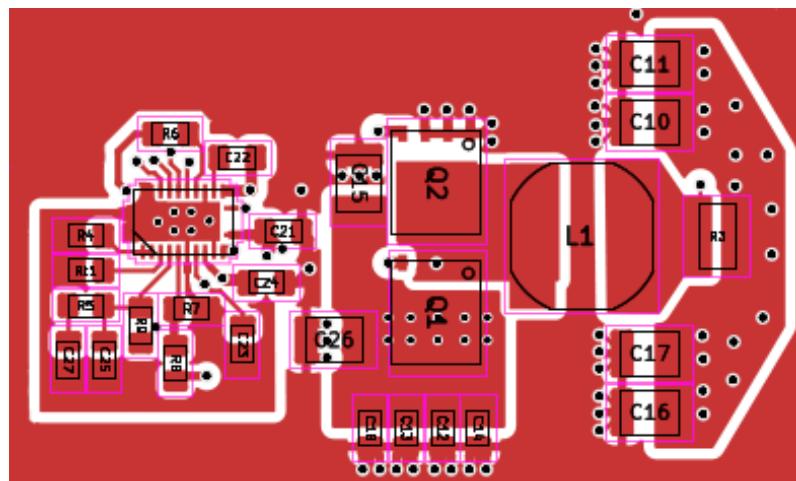
Pro optimální fukci Wi-Fi antény výrobce doporučuje umístit ESP32 modul do pravého horního rohu DPS tak, aby se pod anténou nenacházela vrsta mědi a nejlépe ani samotná deska [24]. Na obr. 3.4 je zobrazen výsledný návrh DPS, kde v modré vyznačené oblasti lze vidět, že tyto požadavky byly splněny. Vedle ESP32 modulu se nachází související součástky popsané v kapitole 3.2.1.



Obr. 3.5: Rozložení vrstev DPS řídicí jednotky.

### Měnič napětí

Při tvorbě části desky s měničem napětí typu buck je potřeba dodržet obecná doporučení týkající se návrhu DPS se spínánými zdroji. Tato doporučení jsou spolu s příklady detailně popsána v katalogovém listu součástky [21] a dalších návrhových příručkách firmy Texas Instruments (výrobce čipu LM5148). Pohled na výsledný návrh propojení součástek na DPS je zobrazen na obr. 3.6. Zobrazena je zde pouze strana součástek (vrchní strana DPS).



Obr. 3.6: Návrh rozložení součástek buck měniče, vrchní strana DPS. Exportováno z programu Kicad 7.0.

V levém dolním rohu obrázku lze vidět oddělení signálové země za pomocí vlast-

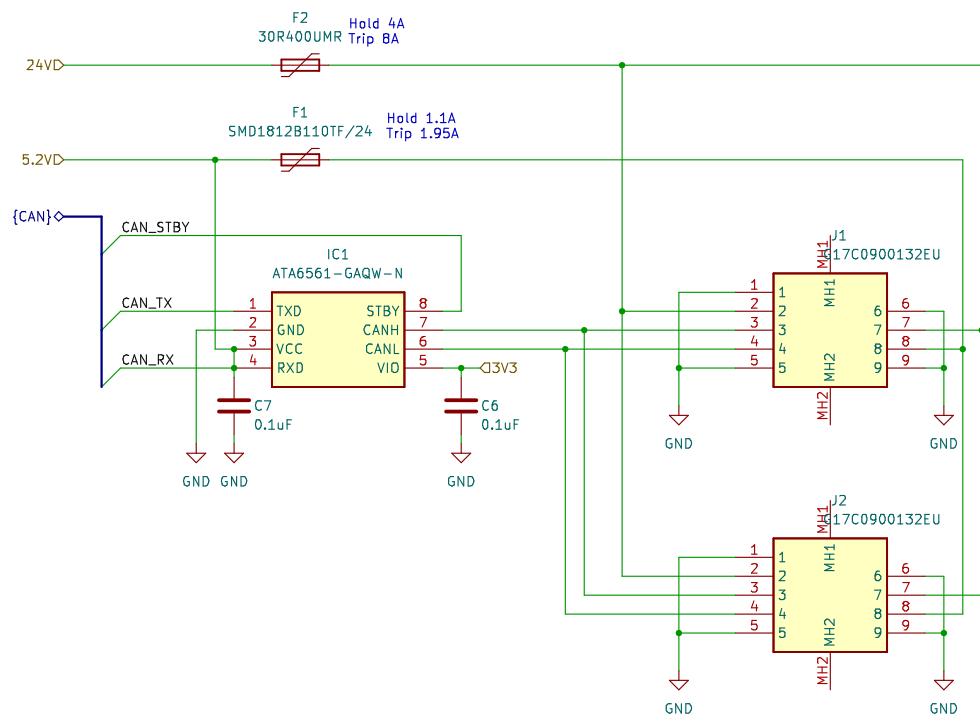
ního ohraničeného polygonu. Tento krok zajistí lepší stabilitu konfiguračních obvodů buck kontroléru. Umístěním tranzistorů a induktoru blízko sebe dojde k minimalizaci plochy spínacího uzlu. Vstupní kapacitory jsou umístěny co nejblíže high-side tranzistoru, výstupní kapacitory pak blízko výstupního uzlu. Vstupní a výstupní napětí jsou přivedeny resp. odvedeny pomocí měděných polygonů na druhé vnitřní vrstvě. První vnitřní plocha je pak celá provedena jako zemníci, zkracuje tak délku proudových smyček. Zbylé signálové cesty jsou vedeny na spodní vrstvě DPS.

### 3.3 Konektivita

Řídicí jednotka je vybavena několika konektory, jak je opět možno vidět na obr. 3.4. Na levé straně DPS (značeno 1.) se nachází konektory pro připojení periferií připravené pro montáž do panelu, kde pak budou přístupné uživateli. Ostatní vyznačené konektory slouží pro připojení externích modulů v rámci hlavního šasi (viz blokové schéma na obr. 2.1) popř. pro programování a testování, nebudou tedy volně přístupné uživateli.

Abychom předešli poškození zařízení při nevhodném zacházení uživatelem, je potřeba pro volně dostupné konektory přidat dodatečnou ochranu [25]. Jak je vidět na obr. 2.2, konektor pro periferie sdružuje jak datovou komunikaci, tak i napájení. Ochrana diferenční datové linky zajistí samotný CAN řadič, který je určen pro průmyslové použití a obsahuje zabudovanou ochranu jak proti zkratu datové linky s napájením či zemí, tak proti ESD [26].

Co se týče napájecích vodičů, každý z nich je ošetřen vratnou pojistkou (ang. polyfuse) dimenzovanou podle předpokládaného maximálního odběru zařízení. Při překročení tohoto proudu, např. z důvodu zkratu v některé z periferií, pojistka sepne a proud v obvodu omezí na minimum. Schéma zapojení konektorů spolu s hodnotami vratných pojistek se nachází na obr. 3.7.



Obr. 3.7: Zapojení a ochrana konektorů. Vytvořeno v KiCad 7.0.

## 4 Obecný modul periferie

Díky zvolené koncepci systému je možné za periferii považovat jakékoli zařízení schopné obousměrně komunikovat po navržené sběrnici. Není vyloučeno, aby byla každá periferie navržena zcela odlišně na základě svých vlastních požadavků na výkon, počet pinů nebo dostupná rozhraní daného MCU. Hlavní výhodou této koncepce je to, že periferie mohou být vyvíjeny postupně a přidávány do již funkčního a odladěného systému bez nutnosti modifikovat stávající hardware. V případě chyby v návrhu periferie je také oprava méně náročná, než by tomu bylo v případě zabudování veškeré funkcionality přímo do řídicí jednotky.

Nicméně pokud by byl pro každou periferii zvolen zcela jiný mikrokontrolér a vytvořen vlastní návrh DPS, vývoj více periferií by byl zbytečně drahý a časově náročný. Proto byl zvolen koncept „obecného modulu periferie“, tedy jedné DPS s konkrétním mikrokontrolérem zajišťující připojení ke komunikačnímu rozhraní, napájení periferie a rozhraní pro programování. Kromě toho budou na DPS dvě dutinkové lišty, do kterých bude možné vsadit druhou DPS (popř. během vývoje pouze prototypovací desku) ve funkci dceřinné desky (ang. daughterboard). Vložená deska pak bude obsahovat obvody nutné přímo pro danou konkrétní periferii.

### 4.1 Mikrokontrolér

Kritéria pro výběr mikrokontroléru byla následující:

- Musí nutně splňovat:
  - CAN periferie – pro komunikaci po sběrnici
  - PWM výstup – řízení LED, popř. jiné
  - ADC – pro práci s analogovými sensory
  - Nízká cena
- Je výhodou:
  - Dobrá dokumentace, komunita uživatelů
  - Zkušenosť autora s danou platformou
  - Další periferie (I<sup>2</sup>C, SPI, ...)

Na základě těchto kritérií byl vybrán mikrokontrolér **PIC18F26Q83** od firmy Microchip, ten splňuje všechna kritéria a disponuje také množstvím dalších periferií, které by mohly být v budoucnu užitečné [27].

### 4.2 Návrh zapojení a tvorba DPS

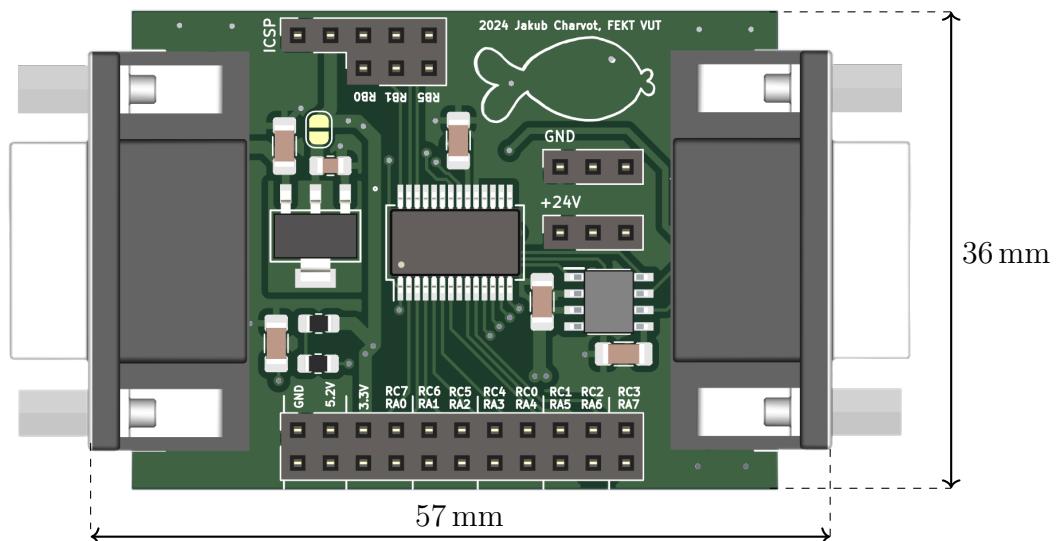
Celé schéma pro obecný modul periferie se nachází v příloze B.1. Návrh zapojení lze rozdělit na tři části. V prvním kroku je zapojen mikrokontrolér tak, aby byly

opět splněny všechny požadavky výrobce. Mikrokontroléry řady PIC se vyznačují značnou jednoduchostí použití a ke správnému chodu jim stačí jen minimum dalších součástek. V případě PIC18F26Q83 postačí připojit blokovací kondenzátor (C6) na napájení (pin VCC) a přivést kladné napětí na resetovací pin (MCLR). MCU je dále doplněn resetovacím tlačítkem pro pohodlnější práci při vývoji a testování firmwaru a regulátorem napětí (U1) s výstupem 3,3 V.

Ve druhém kroku je přidána dvojice D-sub konektorů a CAN řadič ATA6561 obdobně jako u řídicí jednotky popsané v kapitole 3.2. Na závěr jsou všechny dosud nevyužité piny vyvedeny na konektor (dutinkovou lištu), aby byly jednoduše dostupné pro připojenou dceřinnou desku. Pro napájení výkonově náročnejších periferií (např. ovladač LED osvětlení) jsou zvlášť vyvedeny ještě dvě trojice pinů připojené na 24 V a zem (GND).

## Tvorba DPS

Hlavním cílem bylo vytvořit DPS co nejmenší. Jedná se o samostatné moduly, kterých bude v systému zapojeno několik a pro manipulaci a rozmístění modulů okolo akvária je menší rozměr výhodou. Hlavní limitací v této snaze není počet součástek, ale spíše rozměry mechanických prvků, zejména konektorů D-sub. Na obr. 4.1 je vidět výsledný návrh DPS s osazenými součástkami a dosaženými rozměry. Je zřejmé, že jiným rozložením dutinkových lišť by se návrh dal ještě drobně zmenšit, ale je potřeba ponechat jistou flexibilitu pro tvorbu dceřinných desek.



Obr. 4.1: Vizualizace DPS obecného modulu periferie.

Co se týče rozložení vrstev, byla opět použita čtyřvrstvá deska se střejnou strukturou jako je na obr. 3.5 pro DPS řídicí jednotky. Celý návrh DPS se nachází v příloze B.2.

# 5 Volba a návrh periferií

Tato kapitola se již věnuje návrhu konkrétních periferií, tedy jednotlivých senzorů a akčních členů. Po technické stránce jsou všechny zmíněné moduly nadstavbou pro „obecný modul periferie“ popsaný detailně v předešlé kapitole. Jelikož je celý systém modulární, je pravděpodobné, že postupem času bude dále rozšiřován o nové typy periferií a i v současné chvíli je jich v plánu více, než je v možnostech této práce. Pro lepší přehlednost se v tab. 5.1 nachází přehled všech realizovaných i v tuto chvíli pouze plánovaných periferií.

Tab. 5.1: Přehled periferií.

Název	Typ	Napájení	Funkce	Realizováno
Sensor teploty	S	5 V	Teplota vody	Ano, kap. 5.1
Sensor hladiny	S	5 V	Výška hladiny (spojitě + skokově)	Ano, kap. 5.2
LED osvětlení	A	24 V	Intenzita osvětlení	Ano, kap. 5.3
Sensor pH	S	5 V	pH vody	Ano, kap. 5.4
Topné těleso	A	230 V	Ohřev vody	Ano, kap. 5.5
Filtr vody	A	230 V	Filtrace vody	Ano, kap. 5.5
Krmítka	A	24 V	Dávkování krmiva	Ne
Sensor průtoku	S	-	Voda tekoucí filtrem	Ne

S = sensor, A = akční člen

## 5.1 Senzor teploty

Cílem tohoto modulu je kontinuálně měřit teplotu vody a naměřená data poskytovat řídicí jednotce skrze sběrnici CAN. Při volbě konkrétního teplotního čidla je potřeba vzít v potaz několik faktorů:

- Přesnost a rozsah
- Časová stálost
- Složitost implementace
- Pouzdro určené pro ponoření do vody
- Cena

### 5.1.1 Metody měření teploty

Nejčastěji používanými součástkami určenými k měření teploty jsou nepochybňě termistory a termočlánky [28]. Termistor je rezistor vytvořen z materiálu, který mění svůj odpor v závislosti na teplotě přičemž rozlišujeme dva základní typy termistorů

podle toho, zda s rostoucí teplotou jejich odpor roste (PTC termistor) anebo klesá (NTC termistor). U obou typů lze obecně říci, že závislost odporu na teplotě je značně nelineární, pro zjištění přesné teploty je tedy potřeba buďto měřená data dále zpracovat (např. mikrokontrolérem) anebo využít speciální integrovaný obvod, který výstup z připojeného termistoru linearizuje a dále propaguje buďto v analogové nebo i digitální podobě. Jelikož odpor termistoru a stejně tak i dalších součástek, potřebných k jeho zapojení, má jistou výrobní toleranci, je vhodné sensor před použitím kalibrovat.

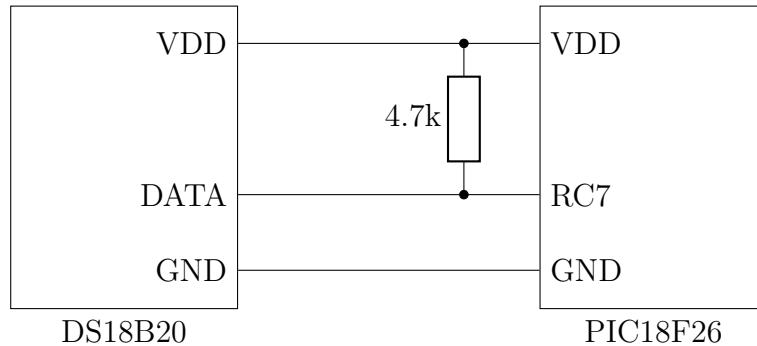
Princip termočlánku je odlišný, jedná se o vodivé spojení dvou kovů na kterém díky Seebeckově jevu vzniká termoelektrické napětí. Velikost tohoto napětí je daná použitými materiály a je také teplotně závislá. V praxi se používá nejčastěji několik dvojic materiálů, které svými vlastnostmi a cenou nejvíce vyhovují běžným požadavkům, ty pak získaly také své označení jako termočlánky typu J, K, T nebo E (typů existuje více, uvedeny jsou nejčastěji používané [29]). Termočlánky pracují oproti ostatním senzorům s výrazně větším rozsahem teplot a mohou měřit také teploty velmi vysoké. Nevýhodou je nízké výstupní napětí, které musí být spolehlivě měřeno, tedy ideálně porovnáno s přesnou napěťovou referencí a také je potřeba, aby část zařízení, ke kterému je termočlánek připojen (tzv. studený konec), byla udržována při konstantní referenční teplotě a nebo případnou změnu teploty měřila jiným způsobem a kompenzovala výpočtem [28, 29].

Z hlediska praxe je další často využívanou možností použití zcela integrovaného sensoru s digitálním výstupem. Pro zpracování je sice potřeba mikrokontrolér, ale tyto sensory bývají od výroby kalibrovány a také jejich zapojení je velmi jednoduché, což je výhodou.

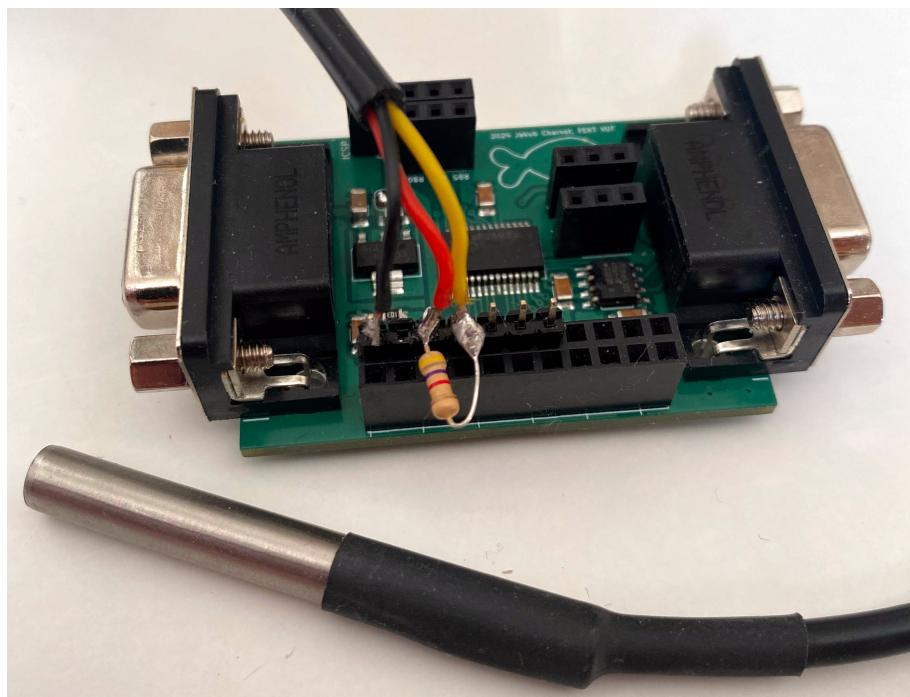
### 5.1.2 Realizace sensoru

Při porovnání uvedených metod se použití termočlánku jeví jako nevhodné, zejména kvůli náročnosti implementace, která současně navýšuje také cenu. Zbývá tedy rozhodnutí mezi termistorem a digitálním čidlem. Ve voděodolném pouzdře lze zakoupit jak několik variant termistorů, tak i digitální čidlo (zde DS18B20). Nejlevněji vychází termistor typu NTC, ale v porovnání s cenou celého zařízení je rozdíl v ceně zanedbatelný.

Pro realizaci sensoru bylo zvoleno digitální čidlo DS18B20, které narozdíl od termistoru není potřeba kalibrovat a výrobce garantuje přesnost  $\pm 0,5^\circ\text{C}$  na celém teplotním rozsahu od  $-55^\circ\text{C}$  do  $125^\circ\text{C}$ . Rozlišení sensoru je až 12 bitů, tedy minimální měřitelná změna teploty odpovídá  $0,0625^\circ\text{C}$ . Pro komunikaci s čidlem se využívá protokol 1-Wire, kdy datový vodič funguje obousměrně, pro propojení čidla s mikrokontrolérem tedy stačí využít tři vodiče a jeden pull-up rezistor, viz obr. 5.1.



Obr. 5.1: Připojení čidla DS18B20 k MCU.



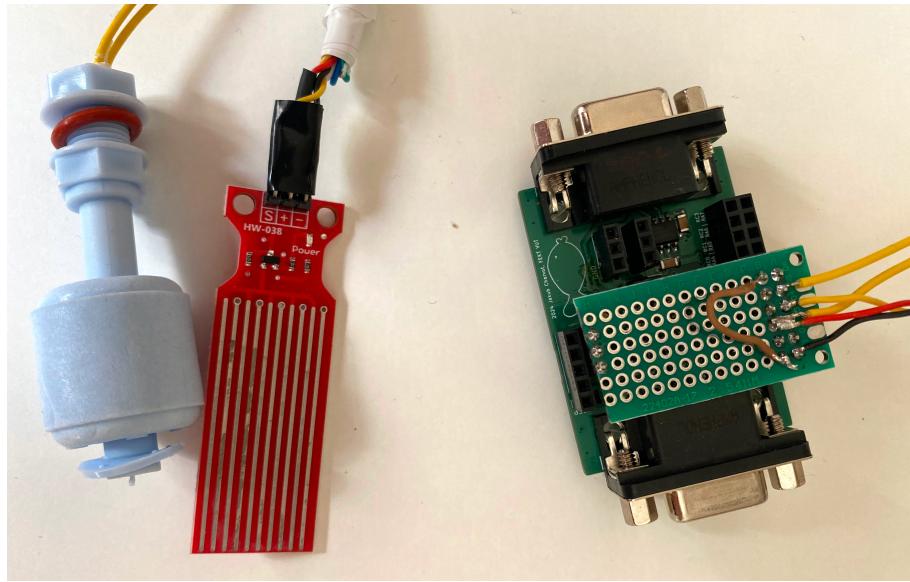
Obr. 5.2: Realizovaný senzor teploty s čidlem DS18B20.

Fyzické připojení čidla k obecnému module periferie bylo provedeno za pomocí jedné pinové lišty, ke které jsou připájeny vývody čidla a stejně tak i pull-up rezistor. Výsledná podoba realizovaného senzoru je vidět na obr. 5.2. V budoucnu bude modul vsazen ještě do vlastní krabičky.

## 5.2 Senzor výšky hladiny

Voda v akváriu se průběžně odpařuje a je potřeba ji doplňovat. Účelem této periferie je průběžné monitorování hladiny akvária a upozornění uživatele na nutnost doplnění vody. Také může uživatele varovat v případě poškození nádrže a nežádoucího úniku

vody do okolí.



Obr. 5.3: Realizovaný sensor výšky hladiny.

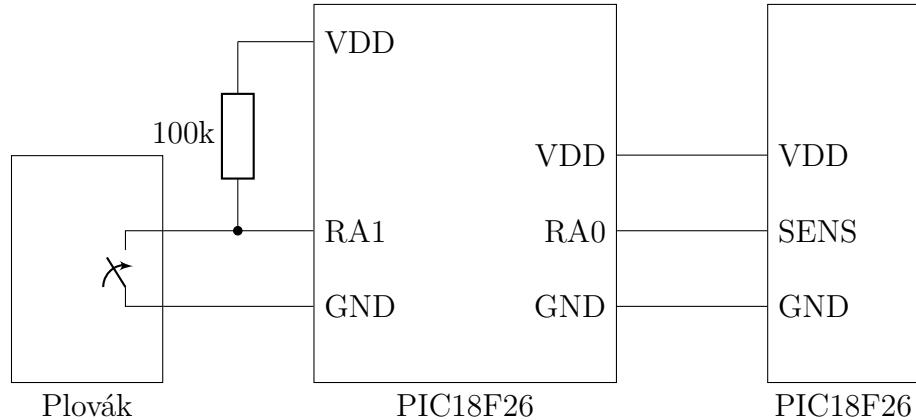
K realizaci tohoto modulu jsou použity dva jednoduché sensory. Každý funguje na jiném principu a má tedy také odlišné přednosti a nedostatky, v kombinaci tak zvyšují celkovou spolehlivost modulu. Oba sensory se nachází na obr. 5.3, který zároveň zobrazuje realizaci jejich připojení k obecnému modulu periferií za pomocí prototypové desky.

### 5.2.1 Popis zvolených sensorů

První ze sensorů využívá k určení výšky hladiny vodivost (potažmo odpor) vody. Obsahuje dva sety vodivých plošek, které nejsou vodivě spojeny. Při ponoření měřící části do vody začne mezi ploškami procházet slabý proud, který je přibližně úměrný velikosti ponořené části. Tento proud otevří tranzistor, na jehož výstupu pak vzniká stejnosměrné napětí v rozsahu přiloženého napájení (zde 0 až 3,3 V). Tento signál je přiveden na pin mikrokontroléru a následně zpracován vestaveným převodníkem. PIC18F26 obsahuje integrovanou periferii ADC s rozlišením 12 bitů, teoreticky lze tedy rozlišit 4296 úrovní [27]. Pro převod měřené hodnoty na výšku hladiny je potřeba sensor nejprve nakalibrovat. Byla tedy změřena přibližná výstupní hodnota pro minimální a maximální měřitelné ponoření sensoru a údaj je následně mikroprocesorem převeden na procenta. Údaj v procentech ponořené části je pro uživatele univerzálním ukazatelem nezávislým na umístění sensoru. Pro měření absolutní výšky hladiny by musel uživatel v systému nastavit výšku umístění sensoru a také ji pokaždé měnit v případě změny jeho pozice.

Druhým senzorem je jednoduchý plovák obsahující mechanický spínač, který je při ponoření do vody rozepnut. V případě poklesu hladiny pod zvolenou úroveň je pak spínač opět sepnut.

Propojení mikrokontroléru s oběma sensory se nachází na obr. 5.4.



Obr. 5.4: Připojení sensorů hladiny k MCU.

## 5.3 LED osvětlení

Úkolem této periferie je zajistit osvětlení akvária a umožnit jeho ovládání. V porovnání s ostatními moduly je zapojení relativně komplexní a proto byla pro tuto periferii navržena a zhotovena vlastní DPS (fungující jako dceřinná deska, viz. 4). Jako typ svítidla byly zvoleny LED pásky pracující s napětím 12 V. Modul musí být schopen samostatně ovládat 2 LED pásky, kdy za pomocí regulace proudu do LED pásku nastaví intenzitu osvětlení.

### 5.3.1 Návrh zapojení

Na začátku návrhu je potřeba specifikovat si požadavky na elektrické parametry zapojení. Uvažujme délku každého pásku  $l = 1 \text{ m}$ , vstupní napětí získané z konektoru D-sub  $U_{in} = 24 \text{ V}$  a výstupní napětí pro které je pásek určen  $U_{out} = 12 \text{ V}$ . Pro stanovení maximálního proudu bylo vycházeno z údajů na e-shopu LED Solution [7], kdy nejvýkonější nabízený LED pásek pro dané napětí má příkon  $P_i = 20 \text{ W/m}$ . Pak každý kanál musí být schopen dodat proud odpovídající nejnáročnějšímu scénáři:

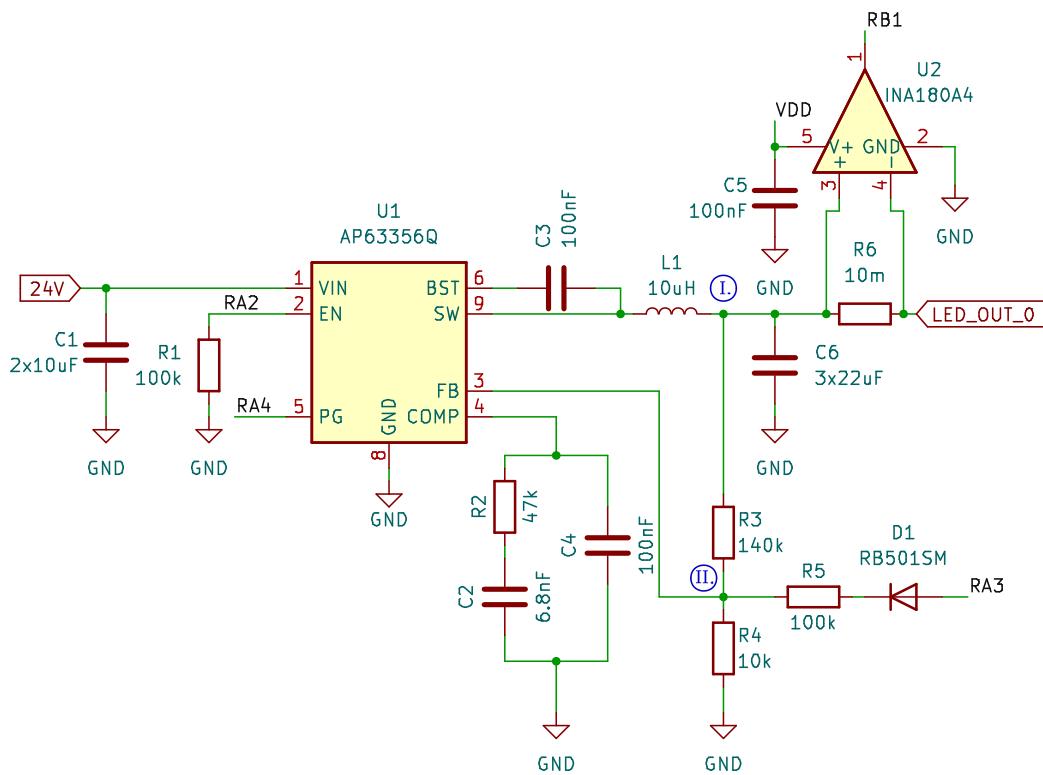
$$I_{max} = \frac{P_i \cdot l}{U_{out}} = \frac{20 \cdot 1}{12} = 1,66 \text{ A} \quad (5.1)$$

Jelikož se jedná o dceřinnou desku pro obecný modul, rozměr výsledné DPS je omezen a část plochy je navíc využita konektory pro vsazení do obecného modulu.

Je tedy potřeba zvolit co nejvíce integrované řešení, které současně slibuje dobrou účinnost a tedy co nejnižší ohřev zařízení během provozu.

Pro řízení LED osvětlení je často používán proudový zdroj, který umožňuje lineárně regulovat výstupní proud a tím i intenzitu osvětlení. Na trhu existuje opět celá řada čipů určena přímo k ovládání LED pásků [[TI\\\_acs {led} Drivers](#)], problém je zde ale v tom, že uživatel velmi pravděpodobně připojí pásek s nižším, předem neznámým příkonem. Maximální proud je tedy specifický danému LED pásku a proudový zdroj by musel zároveň spolehlivě zaručit, že nebude překročeno napětí  $U_{out} = 12 \text{ V}$ .

Tento funkci většina čipů nedisponuje a pokud ano, nejsou dostatečně integrované pro použití v této aplikaci. Po důkladné rešerši a několika iteracích návrhu se nakonec jeví jako nejlepší možnost použítí napěťového měniče typu buck spolu se zesilovačem pro snímání proudu. Snímaný proud je následně zpracován mikrokontrolérem a jsou upraveny poměry ve zpětné vazbě měniče, aby napětí odpovídalo požadovanému proudu.



Obr. 5.5: Zjednodušené schéma ovladače LED. Vytvořeno v KiCad 7.0.

## Popis schématu a výpočty hodnot součástek

Zjednodušené schéma pro jeden ovládaný kanál se nachází na obr. 5.5, při výpočtech bude použito označení součástek z tohoto schématu. Úplné schéma modulu pak lze nalézt v příloze C.1.

Jako buck kontrolér je zvolen čip AP63356Q vyvinutý společností Diodes incorporated, jedná se o úspornou a velmi malou součástku, která v sobě integruje oba potřebné MOSFET tranzistory a spíná s pevně danou frekvencí  $f_{SW} = 450 \text{ kHz}$  [30]. Pro snímání proudu poslouží čip INA180A4 firmy Texas Instruments [31].

Z obrázku je vidět, že pro ovládání jednoho kanálu LED pásků jsou využity 4 piny MCU. RA4 a RB1 fungují jako vstupy, RA2 a RA3 pak jako výstupy. Rezistor  $R_1$  drží buck měnič ve vypnutém stavu, dokud mikrokontrolér nenastaví hodnotu pinu RA2 na logickou 1. Pin RA4 je pak čipem (U1) stažen k zemi vždy, když na výstupu není odpovídající nastavené napětí.

Nastavení výstupního napětí měniče je dosaženo za pomoci zpětnovazební smyčky mezi výstupním uzlem (I.) a zpětnovazebním pinem FB (uzel II.). V uzlu II. je drženo konstantní napětí 0,8 V [30], poměrem rezistorů R3 a R4 je pak definováno výstupní napětí. Zvolíme hodnotu odporu  $R_4 = 10 \text{ k}\Omega$ , pro maximální požadované napětí  $U_{outMAX} = 12 \text{ V}$  platí:

$$R_3 = R_4 \cdot \left( \frac{U_{outMAX}}{0,8} - 1 \right) = 10 \text{ k} \cdot \left( \frac{12}{0,8} - 1 \right) = 140 \text{ k}\Omega \quad (5.2)$$

Na pin RA3 mikrokontroléru je přivedeno analogové napětí z periferie DAC popř. PWM signál. Skrze rezistor R5 (a ochrannou diodu) pak teče proud do rezistoru R4, úbytek napětí na tomto rezistoru je ale konstantní (0,8 V) a tedy je konstantní i proud rezistorem. Z prvního Kirchhoffova zákona pak víme, že proud rezistorem R3 klesne o hodnotu proudu dodanou z pinu RA3, tím klesne také napětí na výstupu měniče a dojde ke ztlumení jasu LED pásku. Citlivost (nebo také rozsah) změny je definována hodnotou R5, snížením jeho hodnoty lze dosáhnout na výstupu ještě nižšího napětí. Pro hodnotu  $R_5 = 100 \text{ k}\Omega$  zobrazenou ve schématu lze nejnižší možné napětí vypočítat následovně:

$$U_{outMIN} = 0,8 + R_3 \cdot I_3 = 0,8 + R_3 \cdot \left( \frac{0,8}{R_4} - \frac{U_{VDD} - U_{D1}}{R_4 + R_5} \right) \quad (5.3)$$

Kdy  $U_{VDD} = 3,3 \text{ V}$  je maximální napětí pinu RA3 a  $U_{D1} = 0,35 \text{ V}$  je prahové napětí zvolené diody. Po dosazení získáme:

$$U_{outMIN} = 0,8 + 140 \text{ k} \cdot \left( \frac{0,8}{10 \text{ k}} - \frac{3,3 - 0,35}{10 \text{ k} + 100 \text{ k}} \right) = 8,25 \text{ V} \quad (5.4)$$

Toto napětí by mělo být dostatečně nízké k úplnému zhasnutí LED pásku.

Pro zajištění stability obvodu je navržena externí kompenzační síť. Je složena ze tří součástek připojených mezi pin COMP a zem. Tyto síť se navrhují z důvodů

zlepšení odezvy systému na rychlé změny výstupního proudu. V ideálním případě chceme, aby systém zareagoval co nejrychleji, ale současně nedošlo k jeho rozkmitání. Vzhledem ke scénáři, kdy MCU uměle vstupuje do zpětnovazební smyčky, je riskantní pohybovat se na zmíněné hranici. Pro zachování stability byla proto zvolena velmi pomalá odezva systému. Požadovaného zpomalení je dosaženo vysokou hodnotou kapacity kompenzačního kondenzátoru,  $C_4 = 100 \text{ nF}$ .

V dalším kroku je stanovena hodnota induktoru L1. Výrobce doporučuje zvolit zvlnění proudu induktorem (ripple)  $\Delta I_L$  jako 30 až 50 % maximálního odběru. Pro výpočet induktoru je uvažován maximální povolený proud čipem, tím je zajištěno, že čip bude fungovat i po překročení maximálního očekávaného proudu  $I_{max} = 1,66 \text{ A}$ . Při zvolení střední hodnoty 40 % získáme:

$$\Delta I_L = 0,4 \cdot I_{IC-max} = 0,4 \cdot 3,5 = 1,4 \text{ A} \quad (5.5)$$

Odpovídající hodnota indukčnosti je vypočtena následujícím vztahem:

$$L_1 = \frac{U_{outMAX} \cdot (U_{in} - U_{outMAX})}{U_{in} \cdot \Delta I_L \cdot f_{SW}} \quad (5.6)$$

Po dosazení získáme:

$$L_1 = \frac{12 \cdot (24 - 12)}{24 \cdot 1,4 \cdot 450 \text{ k}} = 9,52 \mu\text{H} \quad (5.7)$$

Zvolíme nejbližší běžně používanou hodnotu  $L_1 = 10 \mu\text{H}$ .

Pro vstupní (C1) a výstupní (C6) kapacitu použijeme hodnoty doporučené výrobcem, stejně tak pro bootstrap kondenzátor C3.

Poslední součástkou zůstává měřicí rezistor R6. Tímto rezistorem protéká celý výstupní proud měniče, v rámci minimalizace ztrátového výkonu by měl mít tedy co nejmenší odpor. Musíme ovšem také brát v potaz rozsah měřicího zesilovače INA180A4. Tato součástka se vyrábí v několika variantách, byla zvolena varianta s nejvyšším ziskem  $G_{INA} = 200$  pro zachování co nejnižší hodnoty rezistoru, výstupní napětí zesilovače je v rozsahu 0 až 3,3 V (VDD mikrokontroléru). Při maximálním očekávaném proudu chceme dosáhnout horní hranice rozsahu zesilovače, z této podmínky vyplývá vztah pro výpočet odporu rezistoru R6:

$$R_6 = \frac{U_{VDD}}{I_{max} \cdot G_{INA}} = \frac{3,3}{1,66 \cdot 200} = 9,94 \text{ m}\Omega \quad (5.8)$$

Zvolíme blízkou hodnotu  $R_6 = 10 \text{ m}\Omega$ .

## Očekávané parametry

Pro výpočet výstupního zvlnění (v uzlu I.) chybí údaj o ekvivalentním sériovém odporu (ESR) výstupních kondenzátorů (C6), který výrobce neuvádí. Vyjdeme tedy

z typické hodnoty pro keramický kondenzátor  $ESR = 15 \text{ m}\Omega$  [32], kdy počítáme s paralelní kombinací tří kondenzátorů. Očekávané výstupní zvlnění je tedy přibližně:

$$\Delta U_{out} = \Delta I_L \cdot \left( \frac{ESR}{3} + \frac{1}{8 \cdot f_{SW} \cdot C_6} \right) \quad (5.9)$$

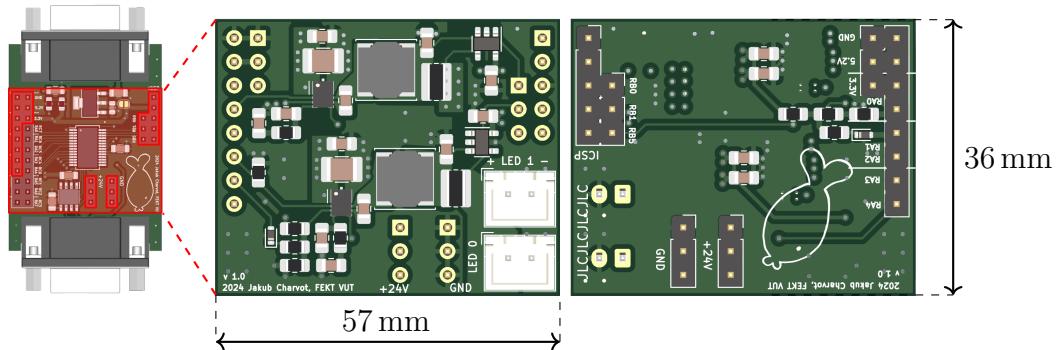
$$\Delta U_{out} = 1,4 \cdot \left( \frac{15 \text{ m}}{3} + \frac{1}{8 \cdot 450 \text{ k} \cdot 3 \cdot 22 \mu} \right) = 13 \text{ mV} \quad (5.10)$$

Výpočet výsledné účinnosti synchronního buck měniče je poměrně komplexním problémem [33]. K jejímu odhadu je potřeba stanovit hodnotu ztrátového výkonu jednotlivých částí měniče. U integrovaného řešení, jako je i použitý čip AP63356Q, výrobce často uvádí pouze hodnoty základních parametrů. Chybí detailly zejména o integrovaných tranzistorech, které jsou klíčové pro výpočet spínacích ztrát. Jako možnost tedy zbývá přibližný odečet z grafů katalogového listu nebo využití kalkulačky dodané výrobcem [34], která v sobě obsahuje přesnější výpočetní rovnice.

Po dosazení odpovídajících hodnot do kalkulačky výrobce je odhadovaná účinnost celého obvodu (při maximálním zatížení jednoho kanálu)  $\eta = 96\%$  a ztrátový výkon  $P_Z = 0,84 \text{ W}$ .

### 5.3.2 Tvorba DPS

Jak již bylo zmíněno v úvodu kapitoly, jedná se o dceřinnou desku pro obecný modul periferie, čímž jsou jasně určeny její maximální rozměry. Stejně jako u předešlých návrhů, i zde je použita čtyřvrstvá struktura viz obr. 3.5. Vizualizace návrhu se nachází na obr. 5.6 spolu s ukázkou sesazení s obecným modulem periferie.



Obr. 5.6: Vizualizace DPS periferie LED osvětlení.

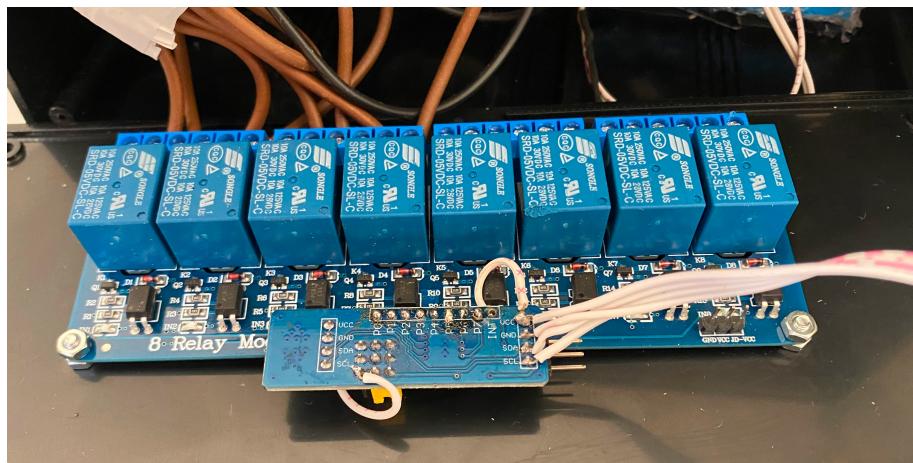
Za účelem zvětšení plochy pro umístění součástek byly z konektorů obecného modulu vyvedeny pouze některé piny, i přesto bylo nakonec potřeba umístit komponenty také na spodní stranu DPS. Rozmístění součástek je obdobné pro oba měniče napětí a respektuje doporučení výrobce a tedy i obecná pravidla pro návrh měničů

napětí [30]. Je kladen důraz na to, aby smyčka ze spínacího uzlu přes výstupní kapacitu a zem zpět do kontroléru byla co nejkratší a vedena za pomocí polygonů. Stejně tak vstupní kapacitor je umístěn přímo vedle vstupních pinů kontroléru.

## 5.4 Senzor pH

## 5.5 Ovládání 230V periferií

Jak vyplývá z požadavků zařízení a přehledu používané akvaristické techniky, pro automatizovaný provoz akvária je nutné umožnit řídicí jednotce ovládat několik okruhů se síťovým napětím a spínat tak zvlášť zakoupené hotové spotřebiče pracující s tímto napětím. Jedná se typicky o ohřev vody, filtrace, popř. některé druhy osvětlení.

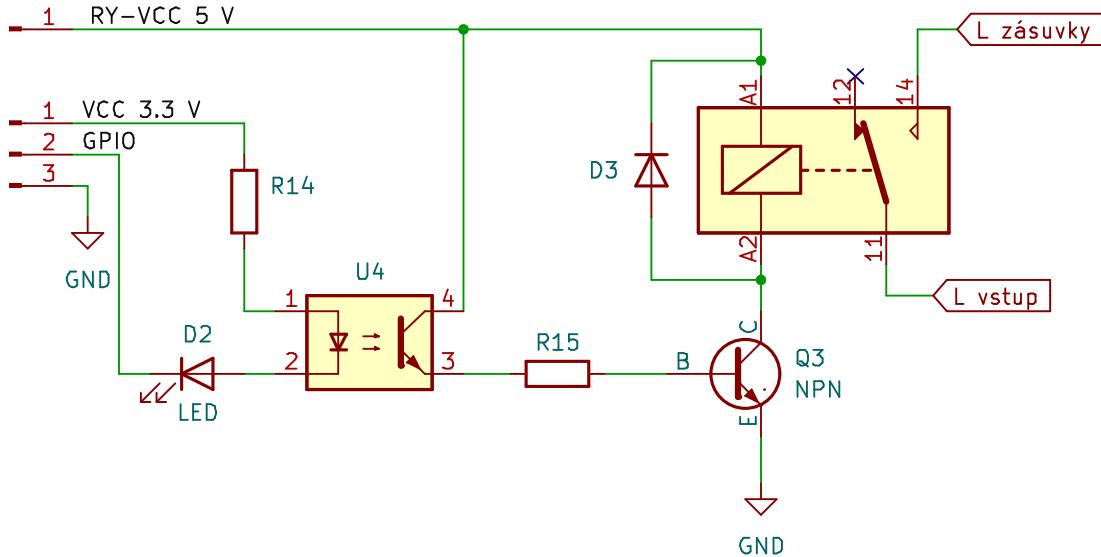


Obr. 5.7: Relé modul spolu s expandérem pinů.

Aby uživatel mohl spínaná zařízení bezpečně připojit bez nutnosti odborné způsobilosti, nachází se na hlavním šasi zařízení čtyři standartní zásuvky (typ E) s jednofázovým napětím 230 V. Fázové vodiče jsou uvnitř zařízení přerušeny spínacími relé. Je použit předpřipravený modul disponující osmi relé [35], čtyři z nich tedy zůstanou nevyužité a slouží jako rezerva pro případ poškození některého z používaných relé nebo při potřebě budoucího rozšíření o další zásuvky.

Z důvodu nedostatku pinů na mikrokontroléru řídicí jednotky (ESP32) je k relé modulu připojen ještě jeden externí modul, a to expandér GPIO pinů komunikující přes sběrnici I<sup>2</sup>C [36]. Z pohledu mikrokontroléru jsou tak všechny zásuvky ovládány pomocí dvou GPIO pinů (SDA, SCL). Ty je navíc možné dále využít pro připojení jiných periferií jako např. OLED displeje pro zobrazení stavu zařízení.

Relé na použitém modulu potřebuje pro spolehlivé sepnutí napětí alespoň 5 V, logické signály řídicí jednotky ale pracují s napětím pouze 3,3 V. Ze schématu na obr. 5.8 je vidět, že použitý relé modul je spínán signálem logické nuly, tímto způsobem je problém s rozdílnou úrovní napájení elegantně vyřešen.



Obr. 5.8: Schéma jednoho kanálu relé modulu. Vytvořeno v KiCad 7.0.

Do budoucna by bylo možným zlepšením a rozšířením této práce zahrnutí obou zmíněných modulů přímo na DPS řídicí jednotky.

# 6 Software

Tato kapitola se věnuje popisu návrhu softwaru pro jednotlivé části systému řízení akvária. Jak již bylo zmíněno v předchozích kapitolách, systém se skládá z řídicí jednotky, ke které jsou připojeny jednotlivé periferie a která komunikuje s webovým serverem za pomoci WiFi. Každá ze zmíněných částí pak potřebuje vlastní zdrojový kód.

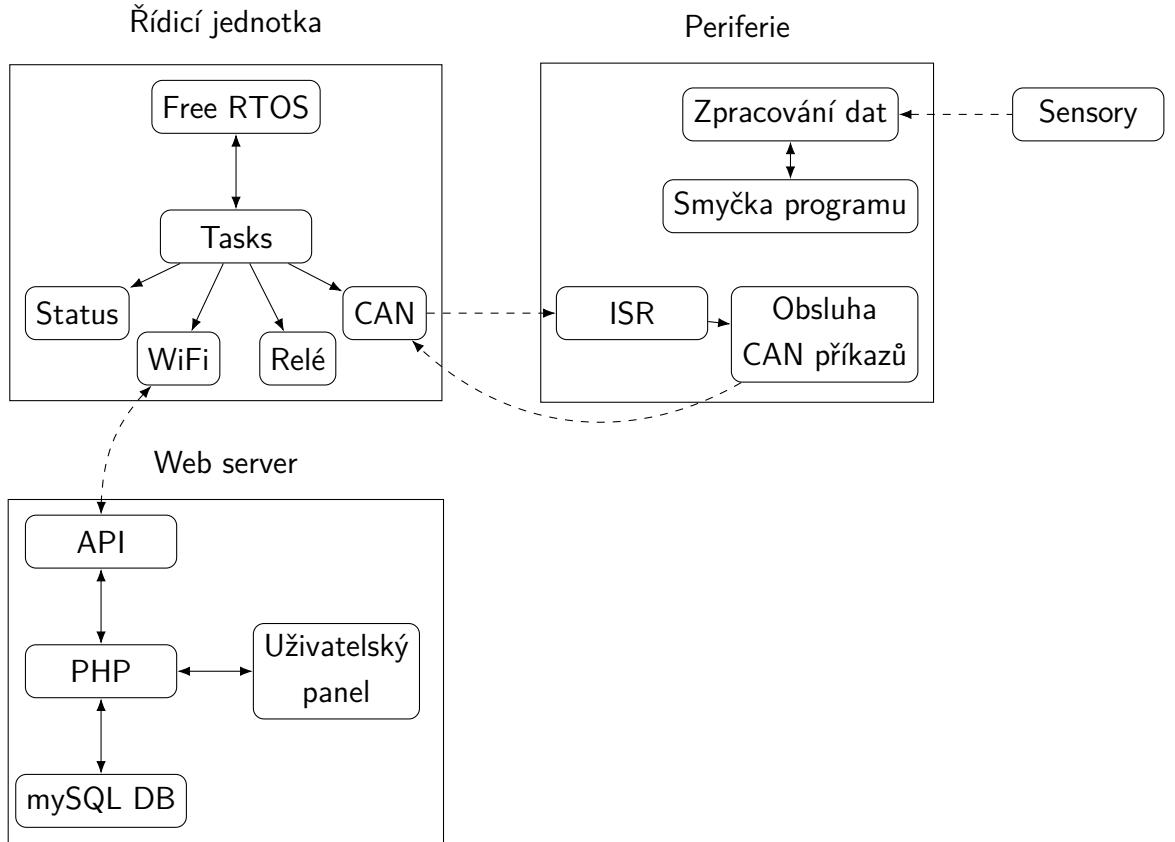
K programování a testování byla použita dvě vývojová prostředí. Visual Studio Code je open source řešení spravované společností Microsoft a díky široké škále doplňků umožňuje velmi univerzální použití. S přidaným rozšířením ESP-IDF je také preferovaným prostředím společnosti Espressif, bylo tedy využito pro tvorbu kódu řídicí jednotky, stejně tak i pro webové rozhraní. Pro programování periferií bylo zvoleno prostředí MPLAB X. Jedná se o řešení společnosti Microchip určené speciálně pro programování mikrokontrolérů této firmy. Součástí je kromě samotného editoru také kompilátor, možnost debugování kódu nebo modul zvaný Code Configurator sloužící pro generování jednoduchého HAL (Hardware Abstraction Level) kódu.

V této chvíli software odpovídá podobě zbytku zařízení – tedy jedná se o prototyp určený primárně k demonstraci funkce zařízení. Aby byl kód použitelný v reálné aplikaci a choval se zde spolehlivě, je potřeba podrobit jej rozsáhlejšímu testování a také lépe ošetřit chování zařízení v různých nežádoucích stavech.

## 6.1 Architektura

Na obr. 6.1 se nachází blokové schéma systému z pohledu softwaru. Obrázek slouží primárně pro lepší orientaci čtenáře v této kapitole, obsahuje pouze klíčové části a některé věci zjednodušuje. Podrobněji se jednotlivým blokům věnují další podkapitoly. Obdélník popsaný v obrázku jako „Periferie“ popisuje strukturu kódu platnou pro všechny periferie, je ale samozřejmé, že periferií této struktury bude v systému připojeno více.

Propojení přerušovanými šipkami v obrázku značí komunikaci mezi dvěma částmi s odlišným programem. Z hlediska internetové komunikace se navržené zařízení chová jako klient, tedy neposlouchá na žádném portu a z vnější sítě není nijak dostupné. Webový server disponuje datovým rozhraním (API), kterého se zařízení v pravidelných intervalech dotazuje na případné změny konfigurace a prostřednictvím kterého zasílá na server data ze svého běhu. Při komunikaci mezi řídicí jednotkou a periferiemi pak řídicí jednotka funguje jako „master“ a periferie odpovídají pouze v reakci na dotaz z její strany.



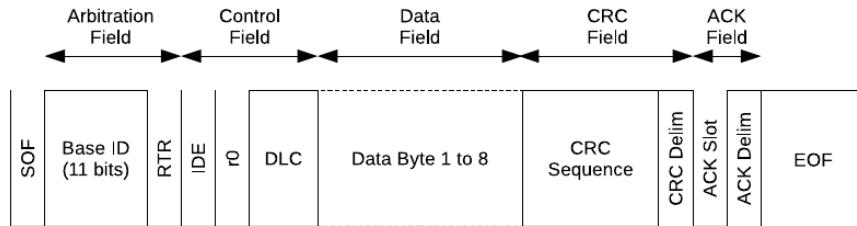
Obr. 6.1: Zjednodušená architektura softwaru systému.

Jelikož jsou řídicí jednotka i periferie programovány ve stejném jazyce (C/C++), lze mezi nimi část kódu sdílet. Tímto způsobem lze částečně předejít chybám v komunikaci modulů mezi sebou. Společná část kódu obsahuje definice datových typů a konstant používaných při komunikaci po sběrnici CAN.

## 6.2 Popis CAN komunikace

Protokol CAN je poměrně rozsáhlý a velká část organizace komunikace je řešena přímo hardwarovou periferií mikrokontroléru. Pro úspěšnou a efektivní komunikaci je potřeba nastavit oba typy mikrokontrolérů stejně a stanovit společný standart komunikace. Systém popsaný v této práci pracuje s frekvencí 125 kHz a používá standartní strukturu rámců s identifikátorem zprávy dlouhým 11 bitů (standart CAN 2.0B umožňuje také délku 29 bitů). Struktura datového rámce je zobrazena na obr. 6.2. Sběrnice CAN má implementovaný princip arbitrace, pokud začne komunikovat více zařízení současně, odešle se zpráva mající identifikátor s nejvyšší prioritou. Logická nula se jeví na sběrnici jako dominantní, jednička naopak jako recessivní. Pokud zařízení odesílá recessivní signál a zároveň čte ze sběrnice signál do-

minantní, znamená to pro něj ztrátu arbitace a přestává vysílat, jelikož na sběrnici je v danou chvíli vysílaná zpráva s vyšší prioritou. Po skončení vysílání pak přerušené zařízení pokus opakuje.



Obr. 6.2: Struktura datového rámce sběrnice CAN. Převzato z [20]

V navrženém systému nese identifikátor zprávy dvě informace. První tři odesílané bity značí typ zprávy. Rozlišena je zpráva určené všem jednotkám (BR – Broadcast), zpráva od řídicí jednotky k periferiím (TS – master To Slave), od periferie zpět k řídicí jednotce (TM – slave To Master) a debug zpráva sloužící k odeslání diagnostických dat nezávisle na ostatní komunikaci. Zbylých 8 bitů pak tvoří adresu jednotky odesílající zprávu (v případě TM) resp. zprávu přijímající (v případě TS).

### 6.2.1 Adresace

Adresy jednotlivých modulů by mely být po startu systému nebo připojení nové jednotky automaticky přiděleny tak, aby nedošlo ke kolizi adres ani v případě připojení několika sensorů stejného typu. Princip adresace spočívá v sérii několika BR zpráv. Po startu systému pošle řídicí jednotka požadavek na adresaci, jako odpověď odešlou jednotky periferií své unikátní sériové číslo přičemž pouze jedna ze zpráv vyhraje arbitraci. Řídicí jednotka odpoví zprávou, která obsahuje sériové číslo úspěšné jednotky a přidělenou osmibitovou adresu. Následně zopakuje požadavek adresace a odpoví již pouze jednotky bez adresy, po dokončení adresaci pak neodpoví žádná jednotka. Pokud je do běžícího systému připojena nová periferie, odešle sama BR zprávu s požadavkem na přidělení adresy.

Ačkoliv je tento princip vymyšlen, v rámci prototypu prozatím není implementován a otestován a každý typ periferie má pevně přidělenou adresu, lze tedy připojit pouze jednu periferii stejného typu. V současné chvíli je toto řešení dostačující.

## 6.3 Firmware řídicí jednotky

Firma Espressif nabízí pro své mikrokontroléry dva základní frameworky. Oba jsou vyvíjeny jako open-source a jsou tedy také volně dostupné pro jakékoliv použití.

Univerzálním řešením vhodným i pro komerční aplikace je ESP-IDF (Espressif Integrated Development Framework). Pro hobby projekty lze využít také Arduino framework, který je taktéž oficiálně podporovaný. Poslední novinkou je pak možnost programování v jazyce Rust namísto klasického C/C++, tento projekt je vytvářen komunitou uživatelů za podpory společnosti Espressif, prozatím ale nebyla oficiálně vydána stabilní verze.

V rámci této práce je využit framework ESP-IDF spolu s několika volně dostupnými knihovnami. Toto řešení již obsahuje HAL kód pro práci s periferiemi MCU, není tedy potřeba pracovat přímo s procesorovými registry. Framework má také podporu pro jednoduchý operační systém FreeRTOS, který je rovněž v této práci využit [37].

### 6.3.1 FreeRTOS

Systém FreeRTOS umožňuje vytvářet tzv. tasky neboli samostatné procesy které běží z pohledu uživatele paralelně. Jelikož má zvolený mikroprocesor dvě jádra, mohou dva tasky běžet skutečně paralelně, více procesů se pak ve svém běhu střídá a běží tak přerušovaně, dá se říci pseudoparalelně. O tuto režii se stará samotný operační systém a vývojář má několik možností, jak tento proces ovlivnit. V případě vytížení procesoru systém přiděluje čas na základě nastavených priorit a dá přednost tasku s vyšší prioritou, může se tak stát, že některý proces zůstane pozastaven na dlouhou dobu. Při tvorbě kódu je potřeba mít toto na paměti, vhodně zvolit priority tasků a také průběžně sledovat vytížení procesoru jednotlivými tasky.

Aby byl kód tzv. thread-safe, tedy bezpečný pro přístup z více vláken, je potřeba ošetřit případy, kdy více tasků pracuje se stejnými daty nebo přistupuje ke stejné periferii mikrokontroléru. K tomuto účelu FreeRTOS nabízí synchronizační objekty jako jsou mutexy, semafory případně fronty.

### 6.3.2 Indikace stavu zařízení

Šasi řídicí jednotky je vybaveno adresovatelným barevným LED páskem sestávajícím ze čtyř diod jejichž úkolem je signalizovat uživateli stav zařízení. Jednotlivé stavы spolu s popisem diod jsou zobrazeny na obr. 6.3. Každý task, který je součástí procesu diagnostiky má svou vlastní globální proměnnou do které ukládá svůj stav. Dvakrát za vteřinu se pak spustí jednoduchá diagnostická funkce (běží v rámci vlastního tasku), která jednotlivé stavové proměnné přečte, vyhodnotí celkový stav zařízení a aktualizuje barvu stavových LED.

Stav zařízení	Stav WiFi	Alarm 1	Alarm 2
Nedefinováno	Nedefinováno	Nedefinováno	Nedefinováno
Inicializace	Inicializace	Upozornění	Upozornění
Ok	Připojeno	Alarm	Alarm
Chyba	Online	OK	OK
	Chyba		

Obr. 6.3: Stavové LED řídicí jednotky.

### 6.3.3 Popis jednotlivých tasků

Řídicí jednotka vykonává současně několik funkcí. Komunikuje s periferiemi skrze sběrnici CAN, ovládá připojená relé a stavové diody. Kromě toho také komunikuje s webovým serverem prostřednictvím WiFi. Jednotlivé úlohy je třeba vykonávat periodicky, ovšem každou s jinou prioritou. Kód je proto rozdělen do několika tasků podle své funkce a priority.

K synchronizaci tasků jsou využity binární semafory. K převzetí semaforu ve FreeRTOS slouží funkce `xSemaphoreTake( xSemaphore, xBlockTime )`. Na řádku s touto funkcí kód čeká až do přijetí semaforu odeslaného jiným taskem nebo do vypršení času stanoveného druhým parametrem.

Každý z tasků začíná inicializací, kdy se jednorázově nastaví potřebné proměnné. Dále pokračuje nekonečnou smyčkou přerušenou vždy čekáním na semafor nebo neblokujícím zpožděním za pomocí funkce `vTaskDelay( const TickType_t xTicksToDelay )`.

#### **Kontrolní proces – control\_task**

Jedná se o hlavní task programu co se týče samotného algoritmu. Na začátku běhu ve vhodném pořadí spustí ostatní tasky a provede vlastní inicializaci. Je spuštěna adresace zařízení na sběrnici CAN a od připojených periferií následně vyžádána in-

formace o jejich typu, sériovém čísle a současném stavu. Z perzistentní paměti flash je na základě sériového čísla načtena konfigurace jednotlivých modulů a samotného systému.

Ve smyčce je následně pravidelně prováděn sled několika operací. Pokud je to potřeba, dojde ke stažení nové konfigurace ze serveru, aktualizaci odpovídajících proměnných i paměti flash. O nutnosti této operace rozhoduje `wifi_task`.

Dále jsou obsluženy periferie na sběrnici. Je vyžádán jejich status a dle typu také senzorická a provozní data.

Na základě konfigurace systému, získaných dat a reálného času je odeslán požadavek odpovídajícím akčím členům a aktualizován stav relé. Také je zpracován případný alarm, který je následně jiným taskem signalizován prostřednictvím stavových diod.

Dle nastavené periody jsou data získaná ze senzorů také odeslána na webový server.

### **Obsluha WiFi – wifi\_task**

V inicializační fázi tohoto tasku je vzněšen požadavek na připojení k síti WiFi. V případě selhání nebo pozdějšího odpojení je pokus automaticky opakován.

V nekonečné smyčce pak task periodicky kontroluje dostupnost internetového připojení, dotazuje se serveru na poslední dostupnou verzi konfigurace a aktualizuje údaj o reálném čase (ten je následně uchován vnitřním RTC časovačem mikrokontroléru).

### **Obsluha sběrnice CAN – can\_bus\_task**

Inicializuje potřebné GPIO piny a připraví potřebné proměnné. Ve smyčce pak pravidelně kontroluje příchozí zprávy na sběrnici (zejména typu Broadcast, které mohou být odeslány periferiemi bez vyzvání). Příchozí zprávy filtruje a pomocí front předává dalším taskům. Na začátku vždy čeká na kontrolní semafor značící požadavek na vyčtení sběrnice. Pokud není vyslán požadavek, čekání na semafor vyprší a sběrnice je zkontovalována s danou periodou.

### **Obsluha WiFi – wifi\_task**

V inicializační fázi tohoto tasku je vzněšen požadavek na připojení k síti WiFi. V případě selhání nebo pozdějšího odpojení je pokus automaticky opakován.

V nekonečné smyčce pak task periodicky kontroluje dostupnost internetového připojení, dotazuje se serveru na poslední dostupnou verzi konfigurace a aktualizuje údaj o reálném čase (ten je následně uchován vnitřním RTC časovačem mikrokontroléru).

### **Indikace stavu – status\_update\_task**

Zpracovává stavové proměnné ostatních tasků a na základě těchto dat vyhodnocuje stav zařízení. Tento stav následně indikuje za pomoci stavových diod, jak bylo popsáno v kapitole 6.3.2.

## **6.4 Firmware periferií**

Jak bylo popsáno v kapitole 4, všechny periferie mají stejné jádro. Ve všech pracuje stejný mikrokontrolér (PIC18F26), stejným způsobem pracují se sběrnicí CAN a také část příkazů na sběrnici je všem periferiím společná. Rozdíl mezi periferiemi je daný pouze připojenými sensory. Ty vyžadují vždy specifické nastavení vstupních a výstupních pinů, zpracování dat a jejich odeslání skrze sběrnicu CAN.

Tato struktura je replikována také ve zdrojovém kódu. Pro všechny periferie existuje společný projekt, kdy základní kód zůstává vždy stejný. Pro sestavení a nahrání programu do konkrétního modulu je potřeba upravit hlavičkový soubor `device_type.h`. Tento soubor nastaví typ zařízení a následně za pomocí preprocessorových direktiv vloží do projektu pouze odpovídající knihovny. Výnatek z tohoto souboru je zde uveden jako výpis 6.1. Stejným principem je kód větven na všech místech, kde je toto potřeba.

Program sestává ze dvou hlavních částí. Nejprve je to fáze inicializace, kdy dojde k nastavení potřebných registrů a proměnných. Poté zařízení vstupuje do nekonečné smyčky. V té jsou periodicky vyčítána data z připojených senzorů resp. ovládány akční členy. Příchozí zprávy na sběrniči CAN vyvolají vždy přerušení, zpracovávají se tedy mimo hlavní smyčku programu.

Výpis 6.1: Část souboru `device_type.h` sloužící k výběru typu cílené periferie.

```
#ifndef DEVICE_TYPE_H
#define DEVICE_TYPE_H

// soubor procesoru PIC
#include <xc.h>

// knihovna sdílená s řídicí jednotkou
#include "../shared/common_types.h"

// Výběr jedné z hodnot definovaných v common_types.h
#define DEVICE_TYPE      DEVICE_TYPE_WATER_LEVEL_SENSOR

...

// Hlavičkové soubory specifické dle typu periferie
#if DEVICE_TYPE == DEVICE_TYPE_LED_BOARD
    #include "led_board_driver/led_board_driver.h"
#elif DEVICE_TYPE == DEVICE_TYPE_TEMP_SENSOR
    #include "temp_sensor_driver/temp_sensor_driver.h"
#elif DEVICE_TYPE == DEVICE_TYPE_WATER_LEVEL_SENSOR
    #include "water_level_driver/water_level_driver.h"
...
#else
    #error "Not supported DEVICE_TYPE"
#endif
#endif /* DEVICE_TYPE_H */
```

#### 6.4.1 Zpracování příkazů CAN

Jako základ je využit kód vytvořený modulem Microchip Code Configurator. Jedná se o užitečný nástroj, který předpřipraví kód nutný k nastavení konkrétních registrů. Také obsahuje funkce úrovně HAL, aby nebylo nutné dále pracovat přímo s registry. Vygenerovaý kód je přitom krátký a snadno čitelný. Veškeré generované nastavení bylo následně kontrolováno s katalogovým listem procesoru a v případě potřeby upraveno.

Na rozdíl od procesoru řídicí jednotky, PIC18F26 obsahuje v periferii CAN až 11 různých filtrů příchozích zpráv [27]. Již na úrovni hardwaru jsou tak zprávy rozděleny dle typu. Zprávy určené periferiím (TS - To Slave) jsou řazeny do jedné fronty, zprávy určené všem (BR - Broadcast) pak do druhé fronty. Ostatní zprávy nejsou

zpracovány vůbec. Při přijetí zprávy je vyvoláno přerušení a kód přečte zprávu z dané fronty. Pokud je to potřeba, odešle vlastní zprávu jako odpověď. První bajt datového pole zprávy vždy identifikuje daný příkaz.

### 6.4.2 Smyčka programu

Jelikož zpracování příkazů ze sběrnice CAN probíhá vždy jako obsluha přerušení, samotná smyčka programu je velmi jednoduchá. V případě sensorů ji lze rozdělit do tří kroků. Nejprve jsou vyčtena surová data ze sensorů a uložena do vhodných proměnných. Ve druhém kroku jsou data zpracována – je vyhodnocena jejich správnost, provedena korekce nebo přepočet, případně jsou data průměrována z více opakování měření. Ve třetím kroku jsou zpracovaná data uložena do globální proměnné. Pokud přišel po sběrnici CAN požadavek na odeslání dat, jsou mu předána tato data.

Na konci smyčky je pak vždy vyhodnocen stav zařízení a opět uložen do globální proměnné.

Přesný průběh jednotlivých kroků programové smyčky se liší v závislosti na typu periferie a byl upravován průběžně k dosažení optimální funkce jednotlivých sensorů.

## 6.5 Webové rozhraní

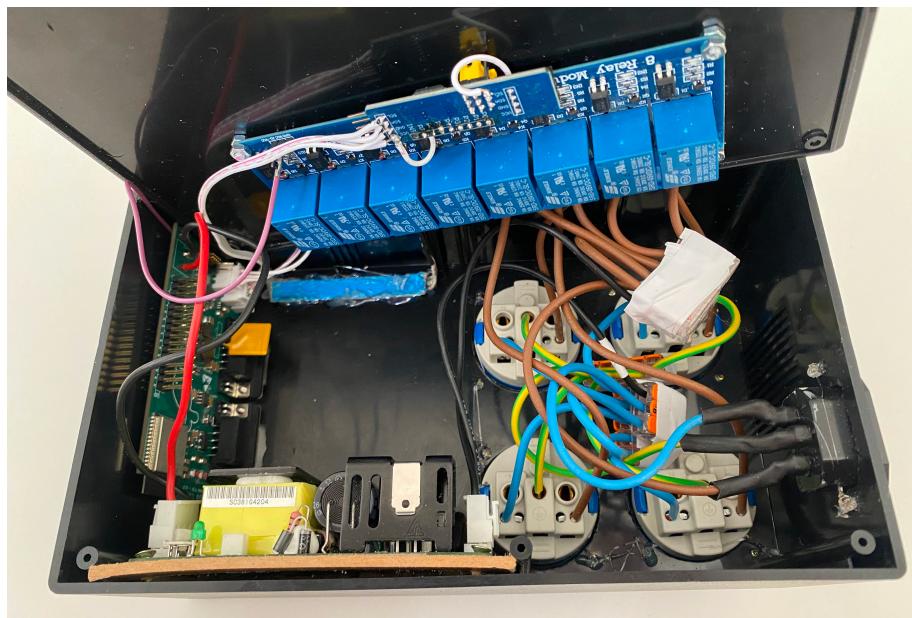
Aby bylo možné systém konfigurovat a monitorovat, bylo zapotřebí navrhnout a vytvořit webové rozhraní. Důležitým krokem v rozhodování byla volba, zda bude MCU řídicí jednotky sloužit přímo jako webový server nebo pouze jako klient. První scénář klade podstatně vyšší nároky na zatížení a paměť MCU, výhodou je ale velmi jednoduchý systém, který funguje samostatně bez nutnosti externího serveru případně také bez připojení k internetu (ESP32 může sloužit přímo jako přístupový bod). Výhodou druhé varianty je větší flexibilita, externí server má nesrovnatelně vyšší výkon a kapacitu úložiště a umožní tak tvorbu mnohem komplexnější webové stránky, která bude (v případě připojení serveru do internetu) dostupná odkudkoliv. Server zároveň může ukládat měřená data a ta tedy budou dostupná i v případě, že samotné zařízení je mimo provoz nebo není připojeno k síti.

V rámci realizace této práce byla zvolena varianta externího serveru. Jádro vytvořené webové aplikace tvoří program v jazyce PHP, který zpracovává jak požadavky uživatele, tak i samotného zařízení. Na pozadí dále běží databázový server s otevřeným systémem MySQL, sloužící k uchování provozních dat. V databázi jsou uloženy údaje o uživatelích, systémech akvárií (tedy řídicí jednotka a k ní připojené periferie) a jejich konfigurace a data získaná ze senzorů. Struktura tabulek databáze je zobrazena na obr. ???. V záznamu odpovídajícímu danému systému akvária je uložen číselný údaj o aktuální verzi konfigurace. Pokud uživatel konfiguraci modifikuje,

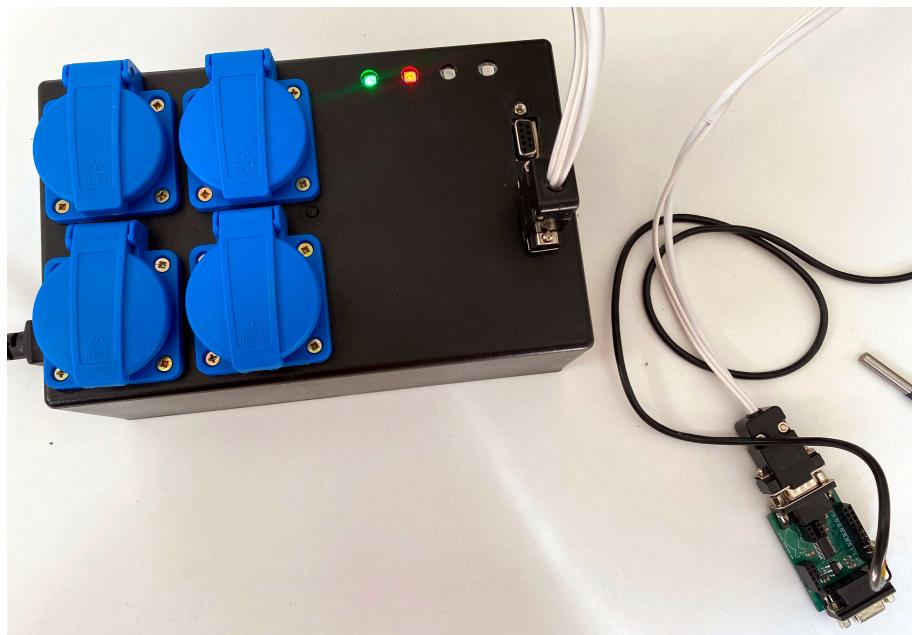
toto číslo se inkrementuje, na což následně zareaguje zařízení a vyžádá si ze serveru novou verzi konfigurace.

Zařízení komunikuje s webem pomocí aplikačního rozhraní (API), které tvoří nenáročný způsob komunikace využívající formát JSON. Jednotlivé adresy API rozhraní jsou přehledně popsány v tab. ??.

## 7 Sestavení a testování



Obr. 7.1: Pohled dovnitř šasi řídicí jednotky.



Obr. 7.2: Foto šasi řídicí jednotky v běhu.

# Literatura

- [1] VÍTEK, Jiří. *Akvaristika včera, dnes a zítra* [online]. [cit. 2023-12-03]. Dostupné z: [https://www.akvarijni.cz/texty/historie\\_akvaristiky.htm](https://www.akvarijni.cz/texty/historie_akvaristiky.htm).
- [2] HÁSKOVÁ, Barbora. *Společenstva pravoků a bezobratlých živočichů ve sladkovodních akváriích a možnosti jejich využití ve výuce*. 2011. Dostupné také z: [https://dspace.cuni.cz/bitstream/handle/20.500.11956/39224/BPTX\\_2010\\_2\\_\\_0\\_258158\\_0\\_107901.pdf?sequence=1&isAllowed=y](https://dspace.cuni.cz/bitstream/handle/20.500.11956/39224/BPTX_2010_2__0_258158_0_107901.pdf?sequence=1&isAllowed=y). Bakalářská práce. Univerzita Karlova v Praze, Pedagogická Fakulta.
- [3] MUSIL, Libor. *Inteligentní systém pro správu akvária*. 2018. Bakalářská práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií.
- [4] ROSTLINNÁ AKVARIA | INVITAL. *Akvarijní filtrace* [online]. YouTube, 2023-06-28. [cit. 2023-12-05]. Dostupné z: <https://www.youtube.com/watch?v=XRoeJDoC10k>.
- [5] ROSTLINNÁ AKVÁRIA. *E-shop* [online]. 2023. [cit. 2023-12-03]. Dostupné z: <https://www.rostlinna-akvaria.cz/eshop>.
- [6] KEJZLAR, Radim. *Automatizované řízení provozu akvária*. 2022. Diplomová práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií.
- [7] E-SHOP LED SOLUTION. *Jak osvětlit akvárium?* [online]. [cit. 2023-12-03]. Dostupné z: <https://eshop.ledsolution.cz/jak-osvetlit-akvarium-led-paskem/>.
- [8] ROSTLINNÁ AKVÁRIA. *Jaké osvětlení vybrat pro Vaše akvárium* [online]. [cit. 2023-12-03]. Dostupné z: <https://www.rostlinna-akvaria.cz/jak-vybrat-osvetleni-pro-vase-akvarium-a-usetrit>.
- [9] SLAVOTÍNEK, Jan. *Systém podpory akvarijních životních podmínek*. 2014. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Vedoucí práce Petr PETYOVSKÝ.
- [10] DVORÁK, Jan. *Regulátor PH pro akvárium*. 2014. Bakalářská práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií.
- [11] GHL. *ProfiLux Aquarium Controller* [online]. [cit. 2023-12-12]. Dostupné z: <https://www.aquariumcomputer.com/products/profilux-aquarium-controller/>.
- [12] *ProfiLux Sets* [online]. E-shop GHL Store. [cit. 2023-12-12]. Dostupné z: <https://store.aquariumcomputer.com/profilux-sets-2/>.

- [13] *Apex EL Controller System* [online]. E-shop Neptune Systems Store. [cit. 2023-12-12]. Dostupné z: <https://shop.neptunesystems.com/products/apex-el-controller-system?variant=14414158921780>.
- [14] NEPTUNE SYSTEMS. *Why Apex?* [online]. [cit. 2023-12-12]. Dostupné z: <https://www.neptunesystems.com/products/apex-controllers/why-apex/>.
- [15] *HYDROS* [online]. E-shop CoralVue Aquarium Products. [cit. 2023-12-12]. Dostupné z: <https://www.coralvue.com/hydros>.
- [16] CORALVUE. *CoralVue Hydros* [online]. [cit. 2023-12-12]. Dostupné z: <https://www.coralvuehydros.com/>.
- [17] SENEYE. *Seneye Home* [online]. 2023. [cit. 2023-12-12]. Dostupné z: <https://www.seneye.cz/seneye-home/>.
- [18] PRODIGY TECHNOVATIONS. *I2C vs SPI: Difference Between I2C and SPI* [online]. 2021. [cit. 2023-12-07]. Dostupné z: <https://prodigytechno.com/i2c-vs-spi/>.
- [19] ESPRESSIF SYSTEMS. *ESP32-WROOM-32E/ESP32-WROOM-32UE Datasheet*. 2023. Ver. 1.6. Dostupné také z: [https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e\\_esp32-wroom-32ue\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf).
- [20] ESPRESSIF SYSTEMS. *ESP32 Series Datasheet*. 2023. Ver. 4.3. Dostupné také z: [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf).
- [21] TEXAS INSTRUMENTS. *LM5148 Datasheet*. 2023. SNVSC01. Dostupné také z: <https://www.ti.com/lit/ds/symlink/lm5148.pdf>.
- [22] *Webench Power Designer* [online]. Texas Instruments. [cit. 2023-12-10]. Dostupné z: <https://webench.ti.com/power-designer/>.
- [23] JLCPCB. *JLCPCB* [online]. JLCPCB. [cit. 2024-04-28]. Dostupné z: <https://jlcpcb.com/>.
- [24] ESPRESSIF SYSTEMS. *ESP32 Hardware Design Guidelines*. 2022. Ver. 3.3. Dostupné také z: <https://docs.espressif.com/projects/esp-hardware-design-guidelines/en/latest/esp32/esp-hardware-design-guidelines-en-master-esp32.pdf>.
- [25] HARRIS, Mark. *Methods of Circuit Protection* [online]. Altium Resources, 2021. [cit. 2023-12-10]. Dostupné z: <https://resources.altium.com/p/methods-protect-your-circuit>. Aktualizováno 29.11. 2023.

- [26] TI. *ATA6560 Datasheet*. 2022. Ver. 3.3. Dostupné také z: <https://docs.espressif.com/projects/esp-hardware-design-guidelines/en/latest/esp32/esp-hardware-design-guidelines-en-master-esp32.pdf>.
- [27] MICROCHIP. *PIC18F26Q83* [online]. [B.r.]. [cit. 2023-12-13]. Dostupné z: <https://www.microchip.com/en-us/product/pic18f26q83>.
- [28] DAVIS, Nick. Introduction to Temperature Sensors: Thermistors, Thermo-couples, and Thermometer ICs. *All About Circuits*. 2022. Dostupné také z: <https://www.allaboutcircuits.com/technical-articles/introduction-temperature-sensors-thermistors-thermocouples-thermometer-ic/>.
- [29] SCIENCE, Techie. *Thermocouples and Temperature Measurement*. 2023. Dostupné také z: <https://techiescience.com/thermocouples-and-temperature/>. Accessed: 2024-05-15.
- [30] INCORPORATED, Diodes. *AP63356Q 3.5A, 36V Synchronous Buck Converter*. 2020. Dostupné také z: [https://www.diodes.com/assets/Datasheets/AP63356Q\\_AP63357Q.pdf](https://www.diodes.com/assets/Datasheets/AP63356Q_AP63357Q.pdf). Accessed: 2024-05-15.
- [31] INSTRUMENTS, Texas. *INAx180 Low- or High-Side, Voltage Output, Current-Sense Amplifier*. 2022. Dostupné také z: <https://www.ti.com/lit/ds/symlink/ina180.pdf>. Accessed: 2024-05-15.
- [32] CONTRIBUTORS, Wikipedia. *Equivalent Series Resistance* [[https://en.wikipedia.org/wiki/Equivalent\\_series\\_resistance](https://en.wikipedia.org/wiki/Equivalent_series_resistance)]. 2024. Accessed: 2024-05-15.
- [33] XING HE Daniel Li, Vincent Zhang. An Accurate Approach for Calculating the Efficiency of a Synchronous Buck Converter Using the MOSFET Plateau Voltage. *Texas Instruments*. 2020. Dostupné také z: <https://www.ti.com/lit/pdf/slvaeq9>.
- [34] INCORPORATED, Diodes. *AP6335x-Q Power Calculation*. 2024. Dostupné také z: [https://www.diodes.com/assets/Calculators/AP6335x-Q\\_Power\\_Calculation.xlsx](https://www.diodes.com/assets/Calculators/AP6335x-Q_Power_Calculation.xlsx). Accessed: 2024-05-25.
- [35] 8-kanálu relé modul 5VDC 250VAC 10A [online]. E-shop LaskaKit.cz. [cit. 2023-12-11]. Dostupné z: <https://www.laskakit.cz/8-kanalu-rele-modul-5vdc-250vac-10a/>.
- [36] *PCF8574 I2C 8bit I/O Expandér* [online]. E-shop LaskaKit.cz. [cit. 2023-12-11]. Dostupné z: <https://www.laskakit.cz/pcf8574-i2c-8bit-i-o-expander/>.
- [37] ESPRESSIF. *ESP-IDF* [online]. [cit. 2024-05-25]. Dostupné z: <https://www.espressif.com/en/products/sdks/esp-idf>.

- [38] PETERSON, Zachariah. *Beginner's Guide to ESD Protection Circuit Design for PCBs* [online]. Altium Resources, 2021. [cit. 2023-12-10]. Dostupné z: <https://resources.altium.com/p/beginners-guide-esd-protection-circuit-design-pcb>. Aktualizováno 13.6. 2023.
- [39] HEDVÍČEK, Michal. *Návrh komplexního automatizovaného systému pro akvaristy*. 2017. Diplomová práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií.
- [40] ESPRESSIF SYSTEMS. *ESP32 Technical Reference Manual*. 2023. Ver. 5.0. Dostupné také z: [https://www.espressif.com/sites/default/files/documentation/esp32\\_technical\\_reference\\_manual\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf).
- [41] INSTRUMENTS, Texas. *LED Drivers Overview*. 2024. Dostupné také z: <https://www.ti.com/power-management/led-drivers/overview.html>. Accessed: 2024-05-15.

# Seznam symbolů a zkratek

<b>IoT</b>	Internet of Things – internet věcí
<b>LED</b>	Light-Emitting Diode – dioda emitující světlo
<b>pH</b>	Potential of Hydrogen – potenciál vodíku
<b>CO<sub>2</sub></b>	Carbon Dioxide – oxid uhličitý
<b>PC</b>	Personal Computer – osobní počítač
<b>Wi-Fi</b>	Wireless Fidelity – známá bezdrátová síť
<b>SPI</b>	Serial Peripheral Interface – typ sběrnice
<b>I<sup>2</sup>C</b>	Inter-Integrated Circuit – typ sběrnice
<b>DPS</b>	deska plošných spojů
<b>CAN</b>	Controller Area Network – typ sběrnice
<b>CS</b>	Chip Select – signál patřící k SPI sběrnici
<b>UART</b>	Universal Asynchronous Receiver-Transmitter – typ sběrnice
<b>MCU</b>	Microcontroller Unit – mikrokontrolér
<b>DC</b>	Direct Current – stejnosměrný proud
<b>MOSFET</b>	Metal-Oxide-Semiconductor Field-Effect – typ unipolárního tranzistoru
<b>ESD</b>	Electrostatic Discharge – elektrostatický výboj
<b>VCC</b>	Voltage at the Common Collector – symbol pro kladné napájecí napětí
<b>VDD</b>	Voltage Drain Drain – symbol pro kladné napětí zařízení
<b>PPTC</b>	Polymeric Positive Temperature Coefficient – také značí typ vratné pojistky
<b>GPIO</b>	General Purpose Input/Output – označení vstupně výstupních pinů MCU
<b>SDA</b>	Serial Data Line – datový signál pro I <sup>2</sup> C

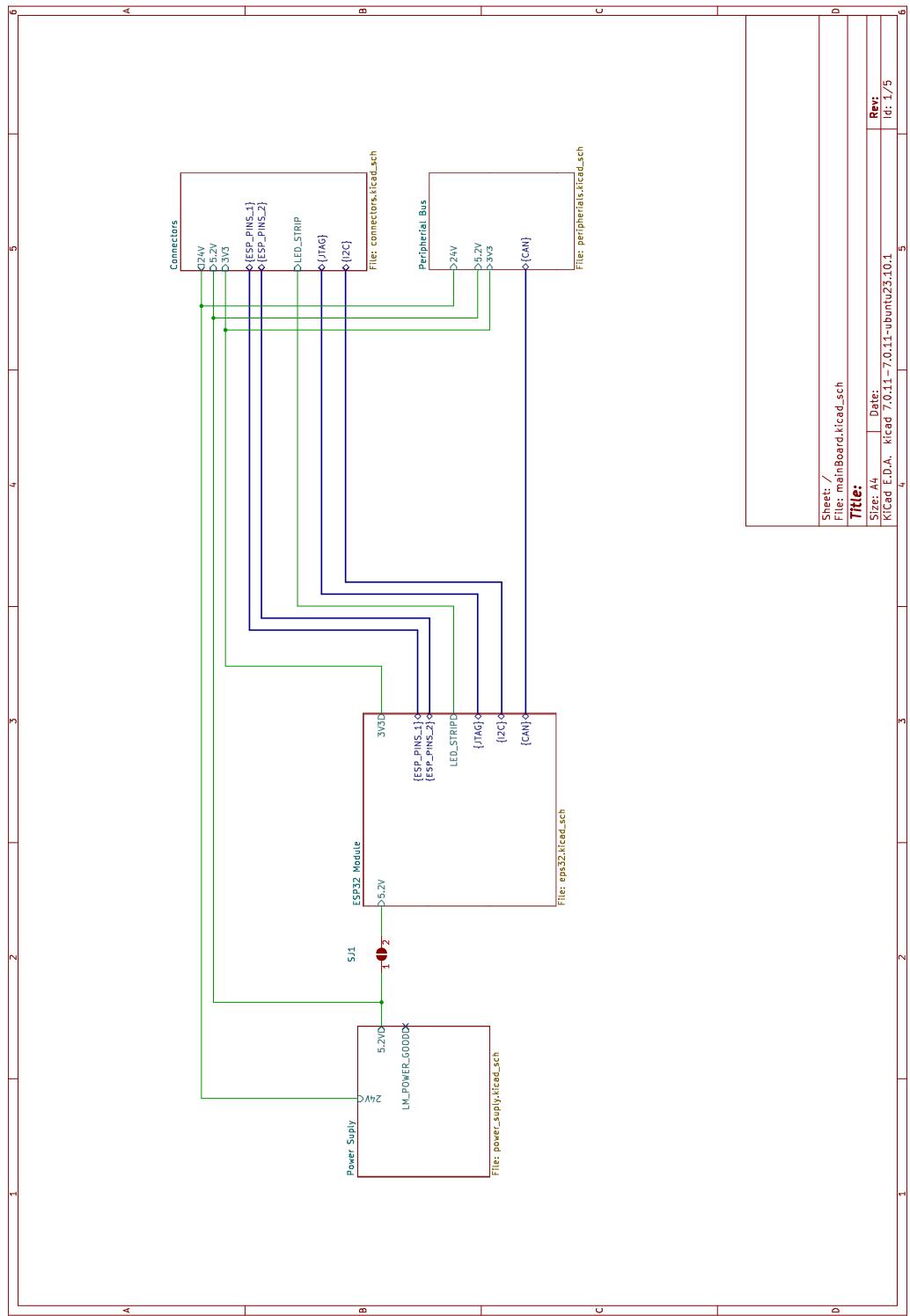
<b>SCL</b>	Serial Clock Line – hodinový signál pro I <sup>2</sup> C
<b>OLED</b>	Organic Light-Emitting Diode – často používaný typ displeje
<b>PWM</b>	Pulse-Width Modulation – pulzně šířková modulace
<b>ADC</b>	Analog to Digital Convertor – Analogově-digitální převodník
<b>DAC</b>	Digital to Analog Convertor – Digitálně-analogový převodník
<b>VCC</b>	Voltage Common Collector – napájecí napětí
<b>GND</b>	Ground – zem, zemnicí vodič
<b>ESR</b>	Equivalent Series Resistance – ekvivalentní sériový odpor
<b>NTC</b>	Negative Temperature Coefficient – záporný teplotní koeficient
<b>PTC</b>	Positive Temperature Coefficient – kladný teplotní koeficient
<b>ESP-IDF</b>	Espressif IoT Development Framework – vývojový framework firmy Espressif
<b>HAL</b>	Hardware Abstraction Layer – abstrakční vrstva ve zdrojovém kódu
<b>API</b>	Application Programming Interface – aplikační rozhraní
<b>RTOS</b>	Real-Time Operating System – operační systém reálného času
<b>PHP</b>	PHP Hypertext Preprocessor – programovací jazyk
<b>ISR</b>	Interrupt Service Routine – obsluha přerušení
<b>DB</b>	Database – databáze
<b>EMC</b>	Elektro magnetická kompatibilita
<b>JSON</b>	JavaScript Object Notation – formát ukládání a přenosu strukturovaných dat
<b>RTC</b>	Real Time Clock – označení reálného času

# **Seznam příloh**

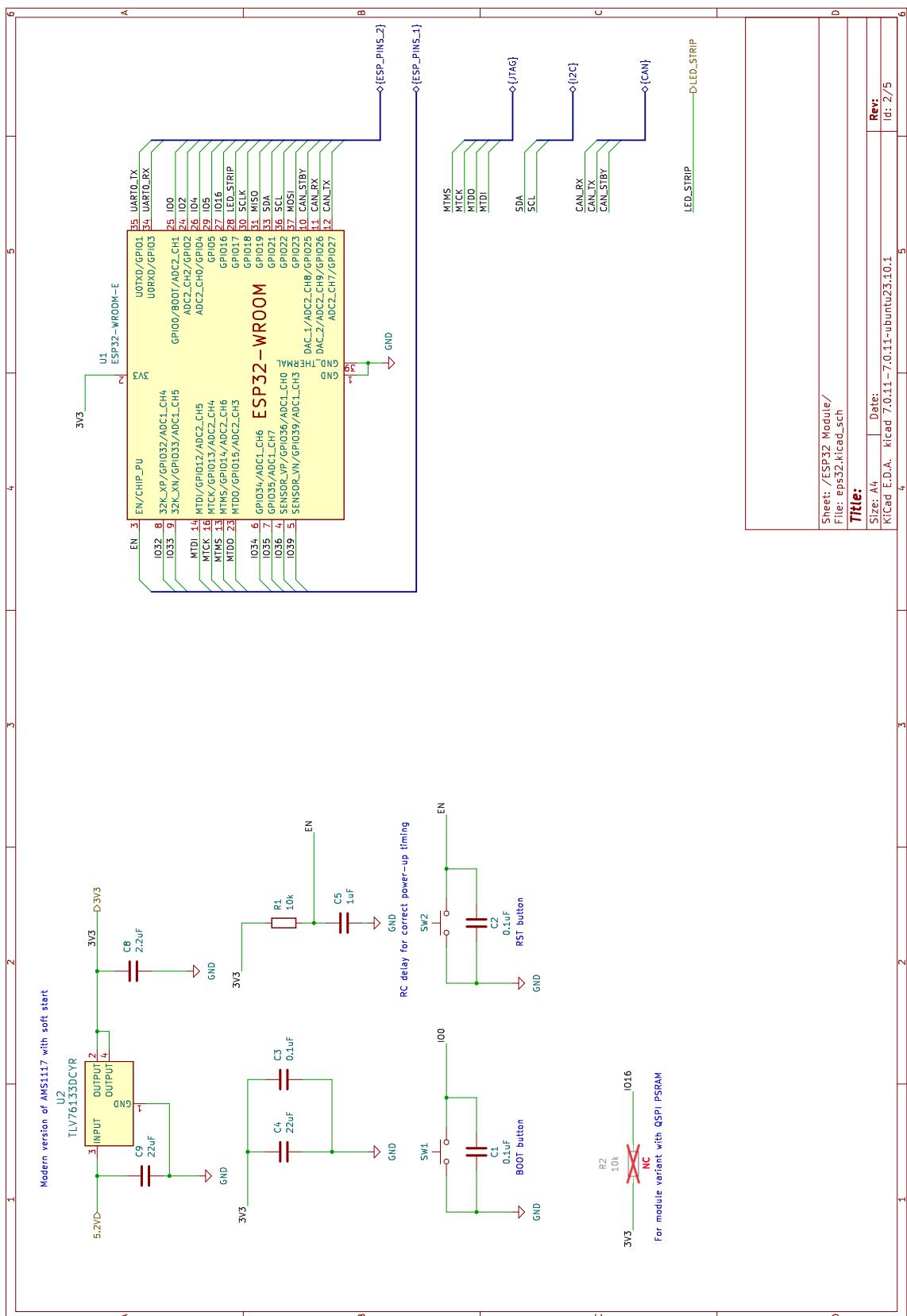
<b>A Schéma a návrh DPS řídicí jednotky</b>	<b>60</b>
A.1 Blokové schéma . . . . .	60
A.2 Zapojení MCU . . . . .	61
A.3 Napájecí obvod . . . . .	62
A.4 Konektory . . . . .	63
A.5 Sběrnice periferií . . . . .	64
A.6 Vodivé motivy DPS . . . . .	65
<b>B Schéma a návrh DPS modulu periferií</b>	<b>67</b>
B.1 Schéma zapojení . . . . .	67
B.2 Vodivé motivy DPS . . . . .	68
<b>C Schéma a návrh DPS modulu LED osvětlení</b>	<b>70</b>
C.1 Schéma zapojení . . . . .	70
C.2 Vodivé motivy DPS . . . . .	71

# A Schéma a návrh DPS řídící jednotky

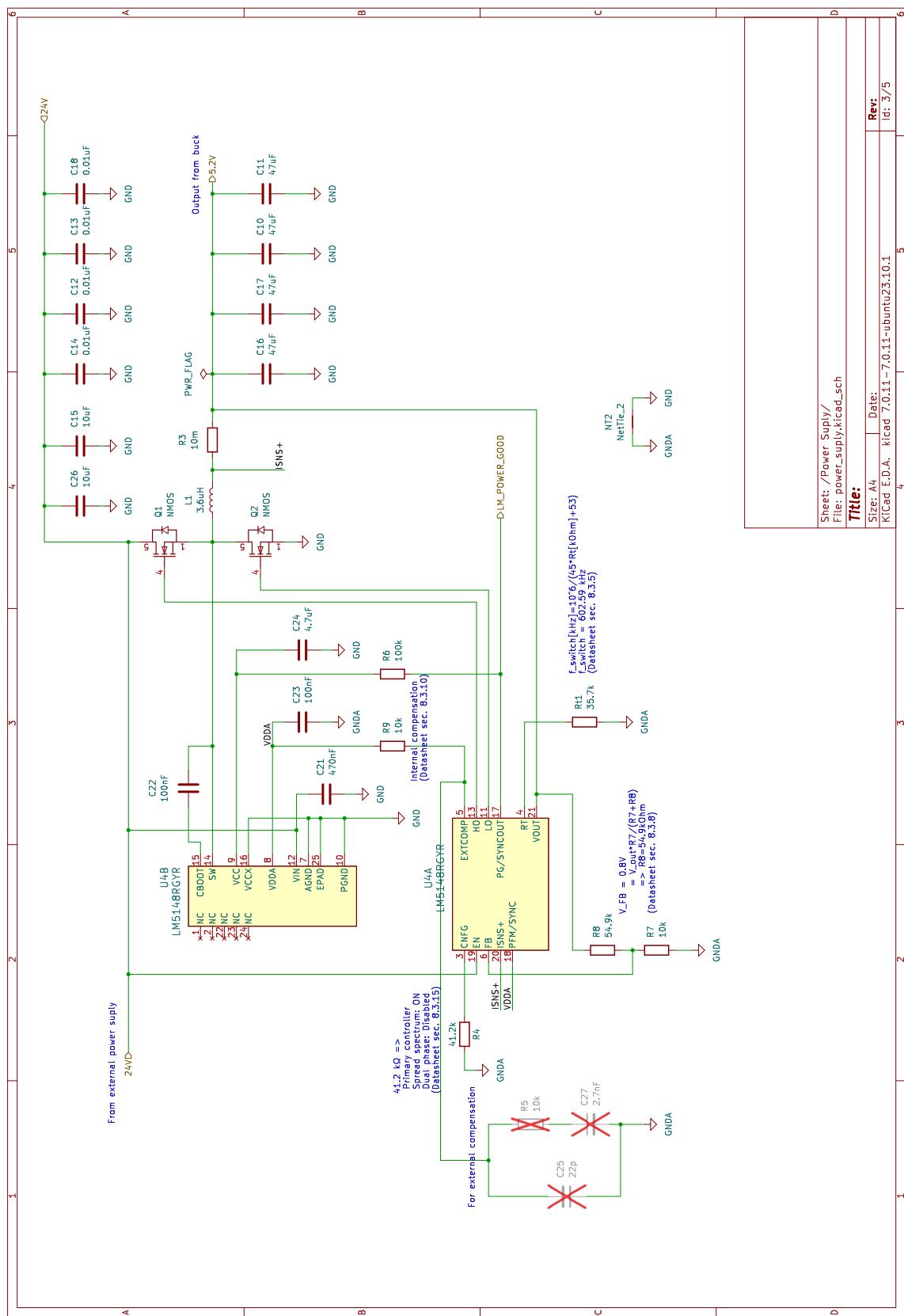
## A.1 Blokové schéma



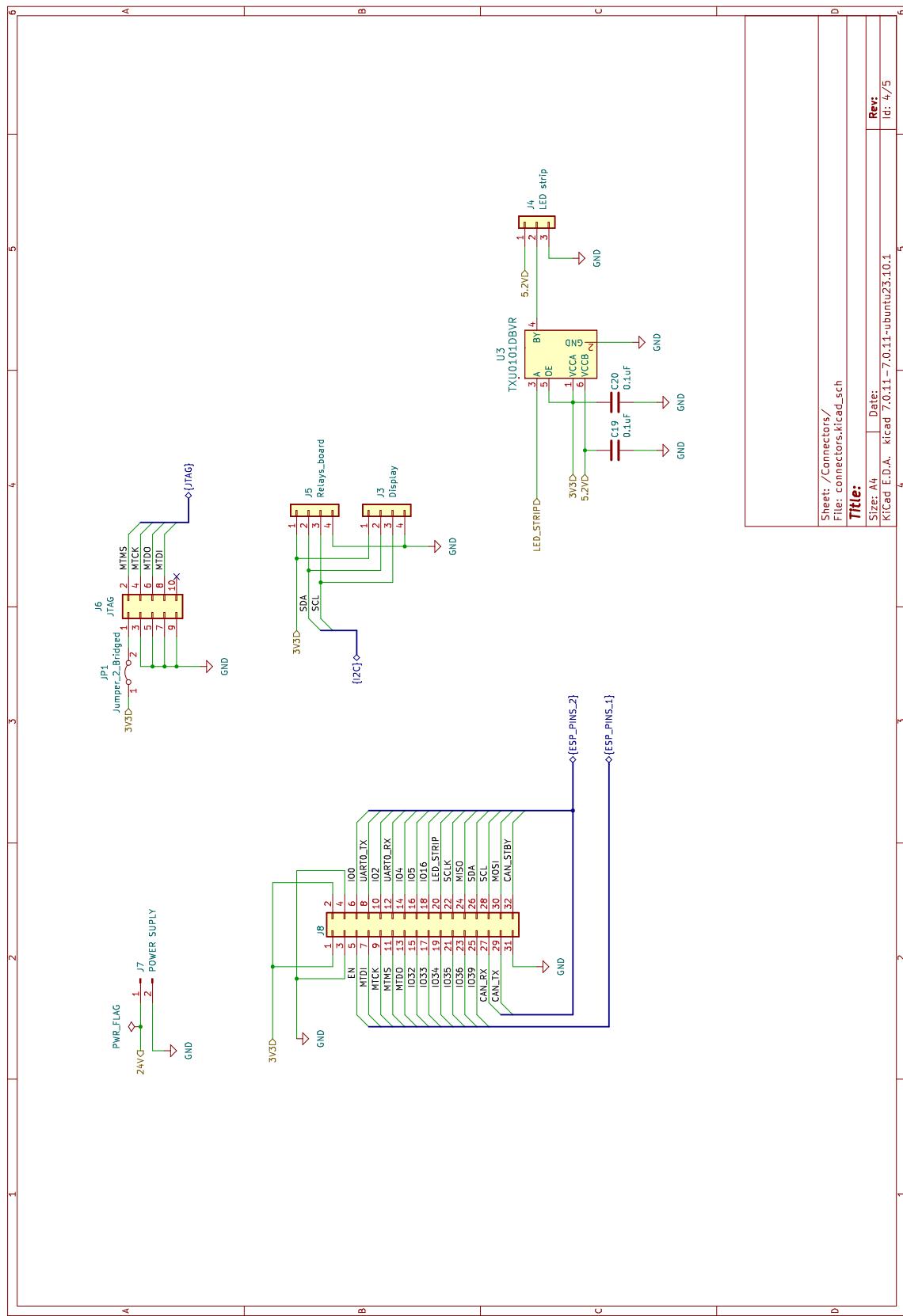
## A.2 Zapojení MCU



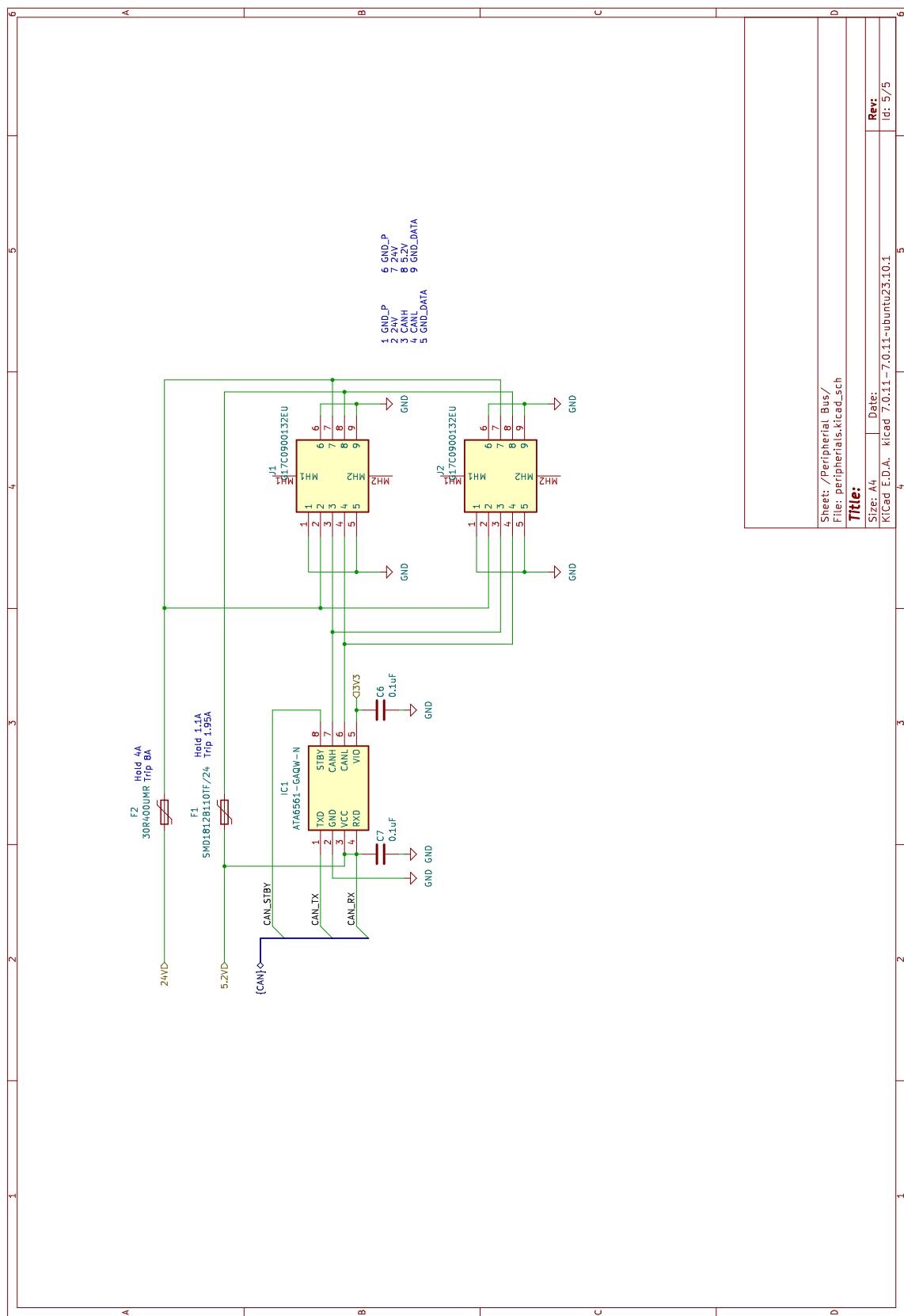
### A.3 Napájecí obvod



## A.4 Konektory

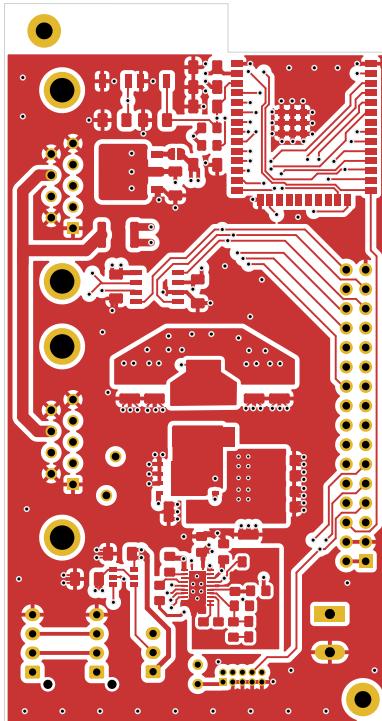


## A.5 Sběrnice periferií

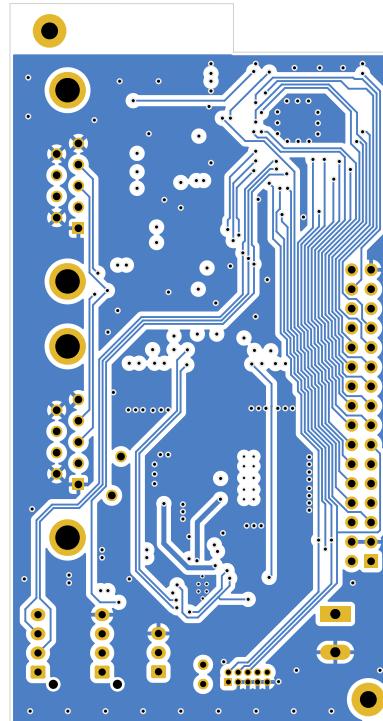


## A.6 Vodivé motivy DPS

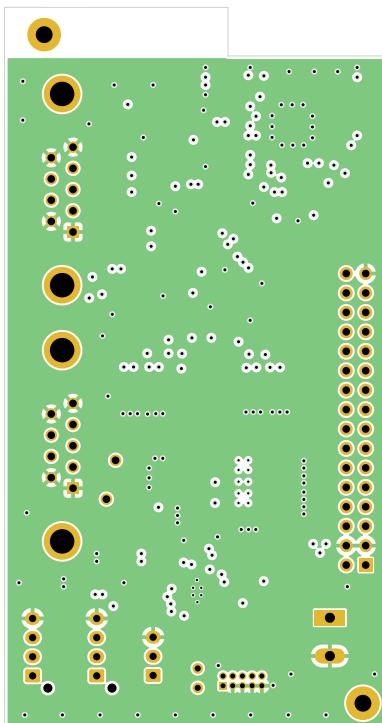
Vodivý motiv horní strany  
(F.Cu)



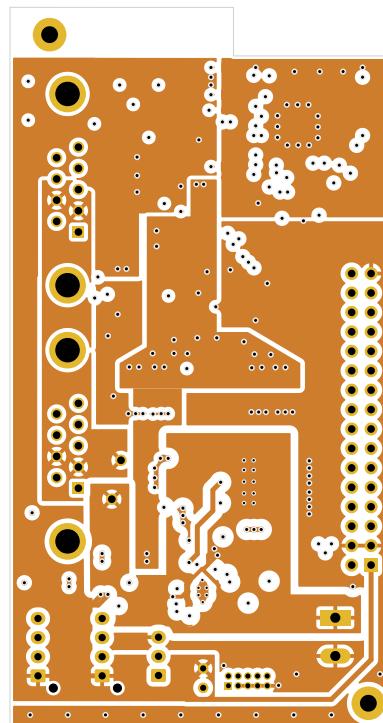
Vodivý motiv spodní strany  
(B.Cu)



Vodivý motiv 1. vnitřní vrstvy  
(In1.Cu)



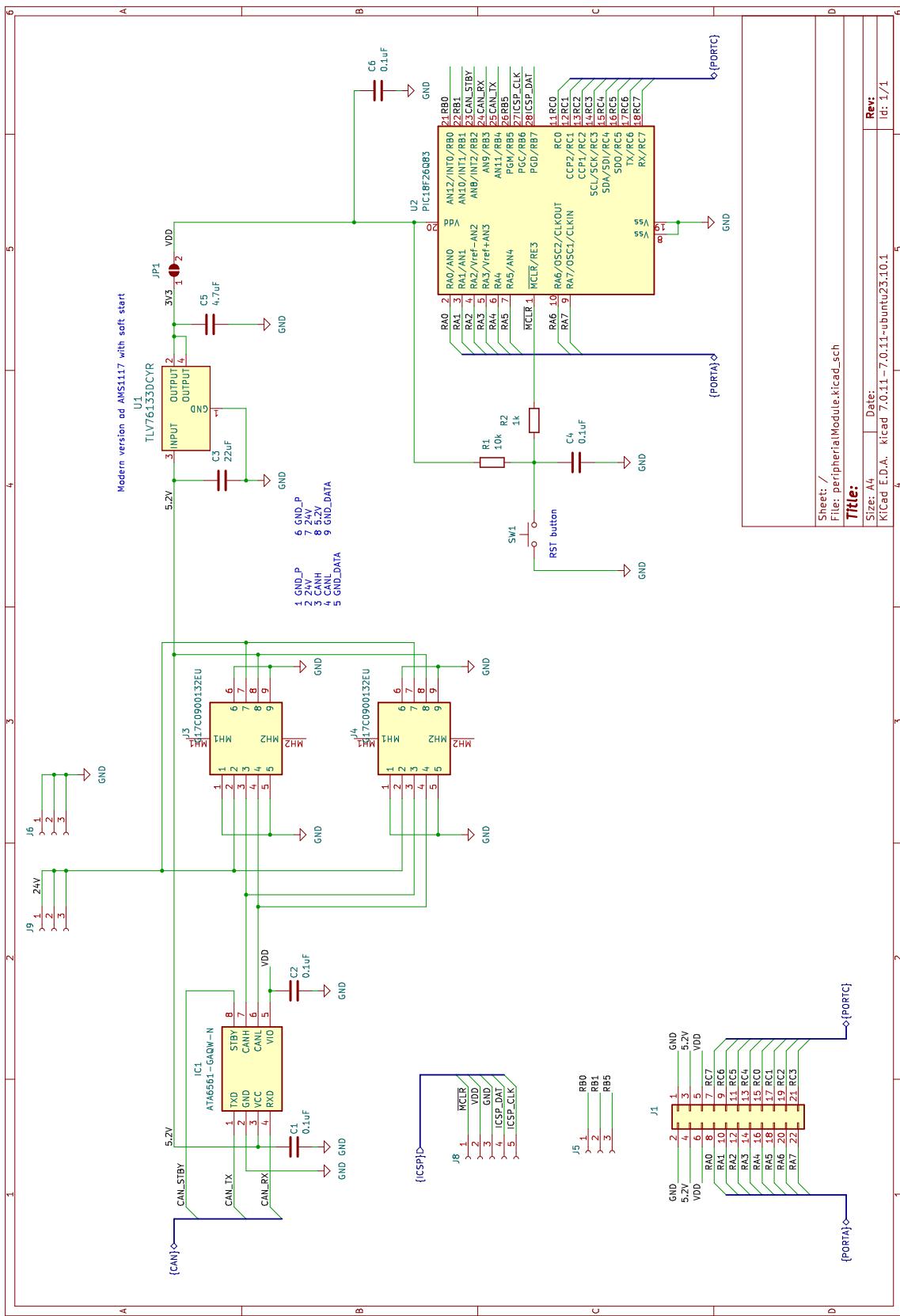
Vodivý motiv 2. vnitřní vrstvy  
(In2.Cu)





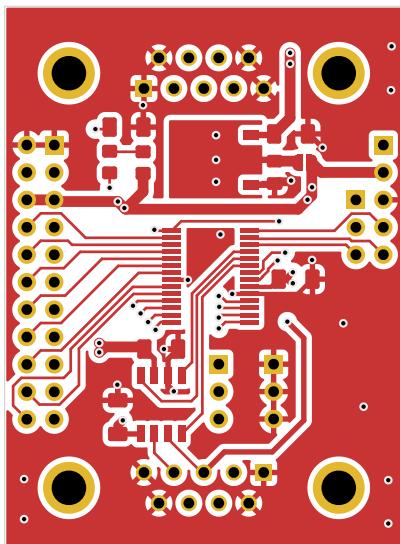
# B Schéma a návrh DPS modulu periferií

## B.1 Schéma zapojení

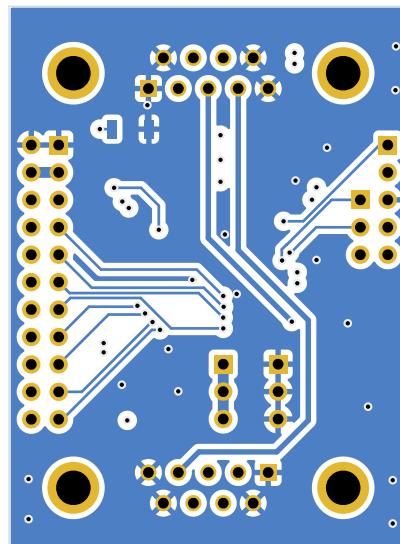


## B.2 Vodivé motivy DPS

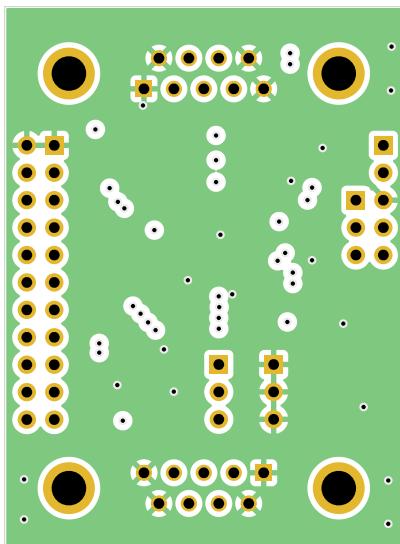
Vodivý motiv horní strany  
(F.Cu)



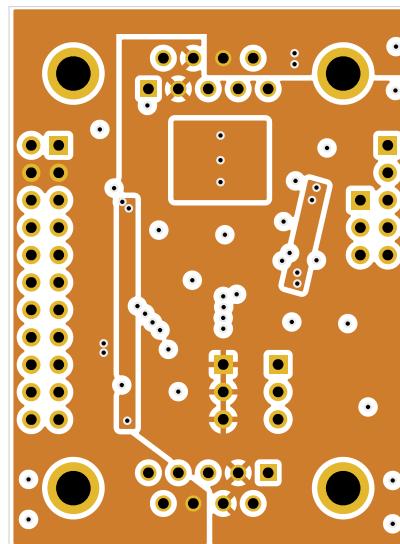
Vodivý motiv spodní strany  
(B.Cu)



Vodivý motiv 1. vnitřní vrstvy  
(In1.Cu)



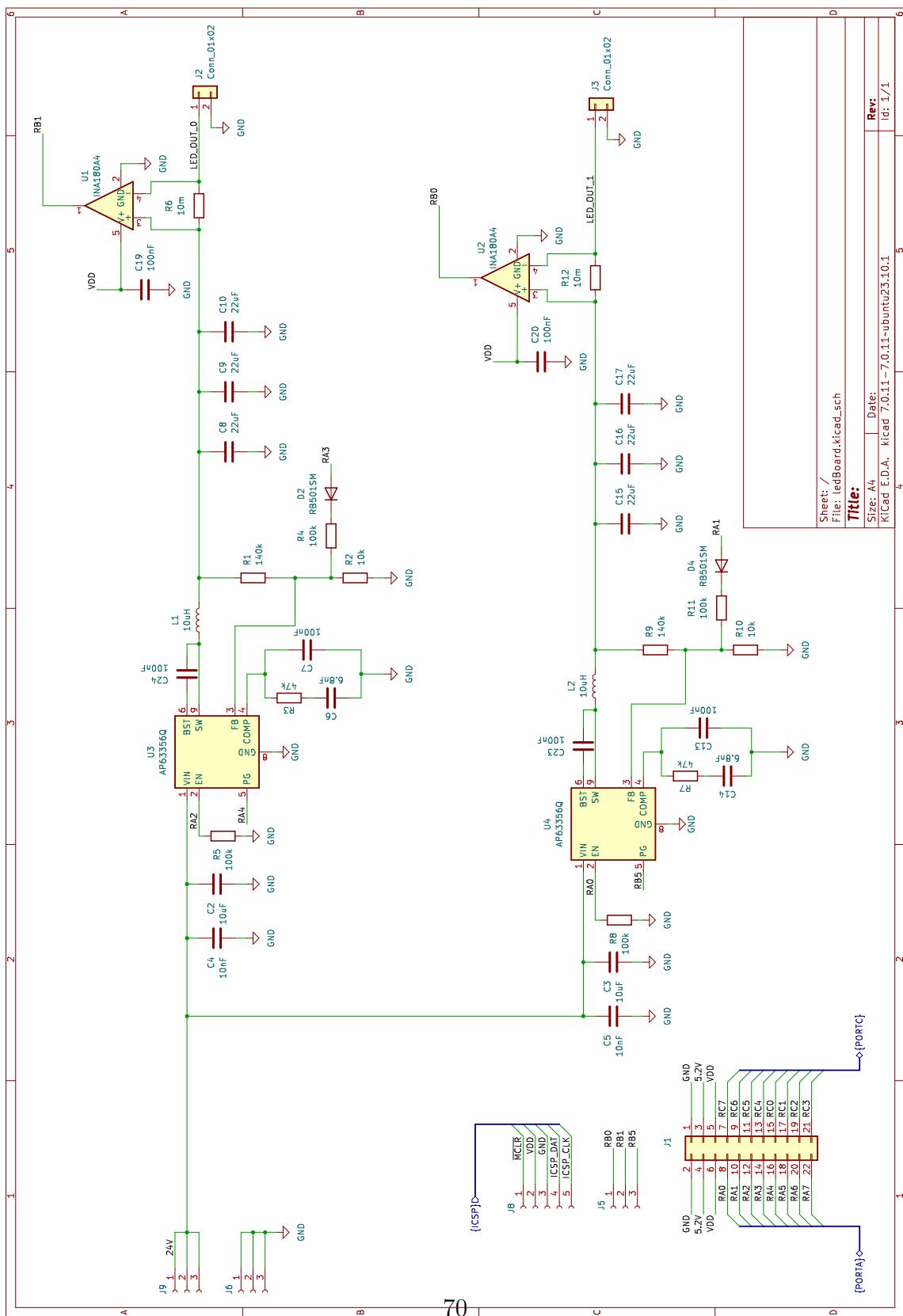
Vodivý motiv 2. vnitřní vrstvy  
(In2.Cu)





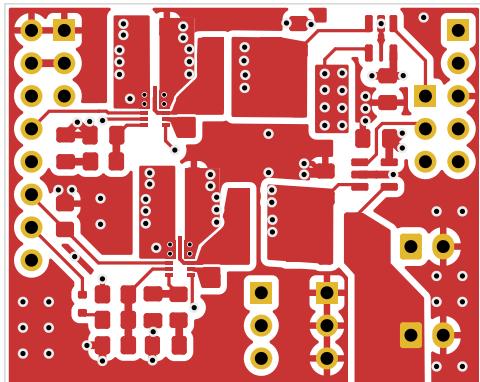
# C Schéma a návrh DPS modulu LED osvětlení

## C.1 Schéma zapojení

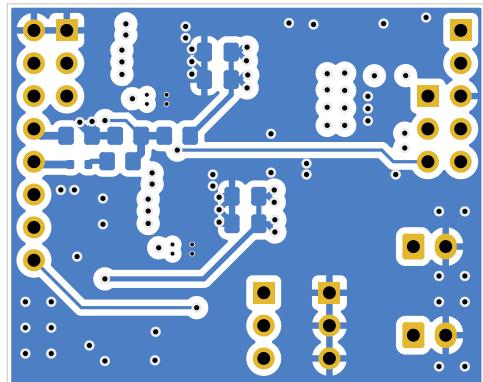


## C.2 Vodivé motivy DPS

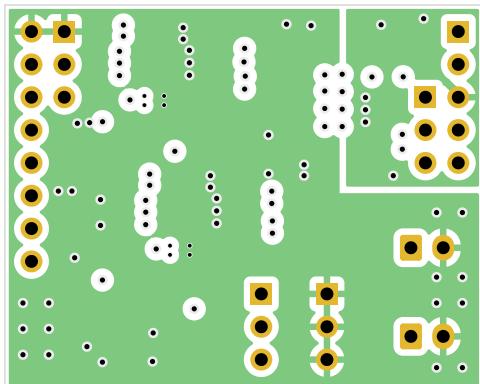
Vodivý motiv horní strany  
(F.Cu)



Vodivý motiv spodní strany  
(B.Cu)



Vodivý motiv 1. vnitřní vrstvy  
(In1.Cu)



Vodivý motiv 2. vnitřní vrstvy  
(In2.Cu)

